Right choice for ultimate yield

LSIS strives to maximize customers' profit in gratitude of choosing us for your partner.

Human Machine Interface

XGT InfoU

XGT HMI Software

User's Manual



Safety Instructions

- Read this manual carefully before installing, wiring, operating, servicing or inspecting this equipment.
- Keep this manual within easy reach for quick reference.

http://www.lsis.com

Revision History

Version	Date	Major Change	Revised Page
V 1.0	‘07.07	The first edition is published	-
V 1.1	‘08.06	Revised to version 1.1	-
V 1.5	‘09.06	Revised to version 1.5	Full revision
V 1.9	‘15.02	Revised to version 1.7	Full revision

※ The manual number is marked on the right side of the manual back cover

ⓒ 2015 LSIS Co., Ltd All rights reserved.

Chapter 1 XGT InfoU 1-1

1.1 Special Feature 1-1
1.1.1 Open Structure 1-1
1.1.2 Efficient Project Management 1-1
1.1.3 Tag Engineering 1-1
1.1.4 I/O Driver Convenience 1-2
1.1.5 Better Alarm Function 1-2
1.1.6 Convenient Logging Management 1-3
1.1.7 Define VBS and Action Trigger 1-4
1.1.8 Robust Graphic Engine 1-4
1.1.9 Cross Reference 1-5
1.1.10 Web Server 1-5
1.1.11 Mobile Monitoring System 1-6

1.2 System Requirement 1-7

1.3 License 1-7

1.4 Product Composition 1-8

1.5 Caution 1-9

Chapter 2 System Composition 2-1

2.1 Software Component 2-1

2.2 Stand Alone System 2-3

2.3 Client/Server System 2-3

2.4 Line Redundancy 2-4

2.5 Redundancy System 2-5

2.6 Web Server 2-6

2.7 Mobile Monitoring System 2-7

Chapter 3 Install and Start 3-1

3.1 Installation 3-1
3.1.1 Arrange Installation 3-1
3.1.2 Select a Package to Install 3-8
3.1.3 Start Installation 3-9
3.1.4 Complete Installation 3-10
3.1.5 Menu and Shortcut 3-10

3.2 Start and Stop 3-11
3.2.1 Start InfoU 3-11
3.2.2 Stop InfoU 3-12

3.3 Removal and Reinstallation of Programs 3-13
3.3.1 Removal of the InfoU 3-13
3.3.2 Reinstallation 3-16

Chapter 4 Engineering 4-1

4.1 Screen Composition 4-2
4.1.1 Main Screen Composition 4-2

4.2 Menu and InfoU Explorer 4-3
4.2.1 File 4-3
4.2.2 Tool 4-5
4.2.3 Help 4-9
4.2.4 Tool Bar 4-10
4.2.5 Pop-up Window 4-10

Chapter 5 Run Time 5-1

5.1 Start InfoU Run-Time 5-1

5.2 Shutdown InfoU Run-Time 5-3

5.3 Runtime Menu 5-6
5.3.1 Menu Item 5-6

5.4 Device Status 5-7
5.4.1 Controls 5-7
5.4.2 Station Staus Information Interface 5-7

5.5 Simulation settings 5-8
5.5.1 Controls 5-8
5.5.2 Simulation Information Interface 5-8

5.6 Device Data 5-10
5.6.1 Controls 5-10
5.6.2 Raw Data Information Interface 5-10

Chapter 6 System Settings 6-1

6.1 Project Property 6-1
6.1.1 Create New Project 6-1
6.1.2 Project Settings 6-4

6.2 Graphic Run-Time Settings 6-9

6.3 Alarm Property 6-11

6.4 Time Synchronization 6-18
6.4.1 Default Screen 6-18
6.4.2 Scenario for the Whole System 6-20
6.4.3 Time Synchronization Settings 6-22

Chapter 7 I/O Driver 7-1

7.1 I/O Driver Setting 7-1
7.1.1 Communication Channel 7-1
7.1.2 Station 7-3

Chapter 8 Tag Management 8-1

8.1 Overview 8-1
8.1.1 Definitions of Terms 8-1
8.1.2 Execution and Main Screen 8-2
8.1.3 Conversion of Tag View 8-4

8.2 Configuration of Communication Equipment 8-6
8.2.1 Communication Channel 8-6
8.2.2 Station 8-9

8.3 Tag Property 8-12
8.3.1 Common Items 8-13
8.3.2 General Items 8-15
8.3.3 Details – Analog 8-17
8.3.4 Details – Digital 8-20
8.3.5 Details – String 8-21
8.3.6 Alarm – Analog 8-22
8.3.7 Alarm – Digital 8-24
8.3.8 Calculations 8-24

8.4 Tag Edition 8-28
8.4.1 Addition of Tags 8-28
8.4.2 Storage of Tags 8-29
8.4.3 Deletion of Tags 8-30

8.5 CSV Tag I/O 8-31
8.5.1 Export to CSV File 8-31
8.5.2 Import from CSV File 8-32
8.5.3 Tag Items of the CSV File 8-34

8.6 Cop/Paste/Cut of Tags 8-38
8.6.1 Copy 8-38
8.6.2 Paste 8-40
8.6.3 Cut 8-41

8.7 Sturcture Tag 8-42
8.7.1 Main Screen and Explanation of Terms 8-42
8.7.2 Addition of Structure of Tag Templates 8-43
8.7.3 Addition of Structure of Tag Items 8-44
8.7.4 Addition of Structure of Tag (Creating the instance of structure tags) 8-45

8.8 Others 8-47
8.8.1 Find/Replace 8-47
8.8.2 Configuration 8-49
8.8.3 License 8-49

Chapter 9 Graphic Editor 9-1

9.1 Configuration of Graphic Editor 9-2

9.2 Menu Configuration 9-3
9.2.1 File 9-5

9.2.2 Edit 9-8
9.2.3 View 9-10
9.2.4 Insert 9-14
9.2.5 Draw 9-16
9.2.6 Tool 9-17
9.2.7 Window 9-21
9.2.8 Help 9-21

9.3 Insert Image 9-22
9.3.1 Insert Image 9-22
9.3.2 Difference between Bitmap and Vector 9-22
9.3.3 Characteristics of each Graphic File Format 9-23

9.4 Tag Link 9-25
9.4.1 Using Location 9-25
9.4.2 Find/Replace 9-26

9.5 Library 9-27
9.5.1 Use Symbol Objects 9-27
9.5.2 Register Symbol Objects 9-27
9.5.3 Library Dialog Box 9-27

9.6 Layer 9-29
9.6.1 Layer Settings 9-29
9.6.2 Layer Lock/Unlock 9-29
9.6.3 Layer Show/Hide 9-29

9.7 Object Property 9-30
9.7.1 General 9-30
9.7.2 Event Property 9-33

9.8 Define Dynamic Properties 9-34
9.8.1 Horizontal Size 9-34
9.8.2 Vertical Size 9-35
9.8.3 Horizontal Move 9-36
9.8.4 Vertical Move 9-37
9.8.5 Horizontal Fill 9-38
9.8.6 Vertical Fill 9-39
9.8.7 Color 9-40
9.8.8 Blink 9-41
9.8.9 Visible 9-42
9.8.10 Rotate 9-43

9.8.11 Display Value 9-44
9.8.12 Display String 9-46
9.8.13 GIF Animation 9-47
9.8.14 GIF Play Time 9-48
9.8.15 GIF Tag Image 9-49
9.8.16 Lamp 9-49
9.8.17 Click 9-50
9.8.18 Input Value 9-69
9.8.19 Horizontal Drag 9-70
9.8.20 Vertical Drag 9-71

9.9 Runtime Settings 9-72
9.9.1 Graphic Runtime Settings 9-72
9.9.2 Page Settings 9-74

9.10 Graphic Script 9-76
9.10.1 InfoU Graphic Function List 9-76
9.10.2 Explanation of InfoU Graphic Function 9-79

9.11 Tag Variable 9-101
9.11.1 Example Used by the Dynamic Property Settings for Graphic 9-101
9.11.2 Example of Using in Graphic Script 9-101

9.12 Objects Using Images 9-106
9.12.1 Button Image 9-106
9.12.2 Lamp Image 9-105

9.13 Smart Symbol 9-108
9.13.1 Creating Smart Symbol 9-108
9.13.2 Registering and Importing the Smart Symbol in the Library 9-109
9.13.3 Smart Symbol Instance Settings 9-110

9.14 Animation Editor 9-111
9.14.1 Execution and Composition of the Animation Editor 9-111
9.14.2 Creation of Animation 9-111

9.15 Lighting Function 9-115
9.15.1 Object Selection Settings in the Graphic Runtime 9-115
9.15.2 Object Highlight Settings in the Graphic Runtime 9-116

Chapter 10 Graphic Runtime 10-1

10.1 Run 10-1

10.2 Menu Composition 10-2
10.2.1 File 10-2
10.2.2 View 10-4
10.2.3 Operation 10-5
10.2.4 Help 10-9

10.3 Popup Menu 10-10
10.3.1 Detailed Realtime Trend 10-10
10.3.2 Realtime Trend 10-11
10.3.3 Realtime List Trend 10-11
10.3.4 Alarm Viewer 10-12
10.3.5 Input Value 10-12
10.3.6 Cross Reference 10-12

10.4 Cross Reference Function 10-13
10.4.1 Execution on the Plant Screen 10-13
10.4.2 Execution on the Tag View 10-14
10.4.3 Execution on the Alarm Pop-up Screen 10-15
10.4.4 Execution on the Graphic Object 10-15

Chapter 11 Graphic Object 11-1

11.1 How to Use Graphic Object 11-1
11.1.1 Object Insert 11-1
11.1.2 Script Use 11-3

11.2 Alarm Viewer 11-4
11.2.1 Prerequisite and Environment 11-4
11.2.2 Screen Configuration 11-4
11.2.3 Real-time Function 11-6
11.2.4 Settings 11-11
11.2.5 Filtering 11-14
11.2.6 Other Functions 11-16
11.2.7 How to Use Script 11-19

11.3 Trend Viewer 11-25
11.3.1 Prerequisite and Environment 11-25

11.3.2 Screen Configuration 11-25
11.3.3 Settings 11-27
11.3.4 Logging Search 11-45
11.3.5 Others 11-46
11.3.6 How to Use Script 11-47

11.4 List Trend Viewer 11-60
11.4.1 Prerequisite and Environment 11-60
11.4.2 Screen Configuration 11-60
11.4.3 Settings 11-61
11.4.4 How to Use Script 11-71

11.5 Recipe Viewer 11-79
11.5.1 Prerequisite and Environment 11-79
11.5.2 Screen Configuration 11-79
11.5.3 Real-Time Function 11-80
11.5.4 How to Use Script 11-85

11.6 Gauge Control 11-88
11.6.1 Prerequisite and Environment 11-88
11.6.2 Screen Configuration 11-88
11.6.3 Property Settings 11-89

11.7 How to Use External ActiveX 11-91
11.7.1 How to Use 11-91

Chapter 12 Alarm Management 12-1

12.1 Alarm Kind and Rules 12-1
12.1.1 Analog Alarm 12-2
12.1.2 Digital Alarm 12-4
12.1.3 Alarm Delay 12-5

12.2 Alarm Group Settings 12-7
12.2.1 Alarm Group Management 12-7
12.2.2 Alarm Group Settings 12-9

12.3 Alarm Service 12-12
12.3.1 Alarm Sound 12-12
12.3.2 Common Items for Alarm Service 12-13
12.3.3 Automatic Printout of the Line Printer 12-14
12.3.4 Mail Sending 12-15

12.3.5 SMS Transmission 12-20

Chapter 13 Logging Management 13-1

13.1 Logging Type 13-1
13.1.1 Logging Type 13-1
13.1.2 Calculation Method for Logging Data 13-2

13.2 Logging Settings 13-3
13.2.1 Configuration Interface 13-3
13.2.2 Add, Edit and Delete 13-4
13.2.3 General 13-5
13.2.4 Cycle 13-7
13.2.5 Statistics 13-7
13.2.6 Accumulation 13-8
13.2.7 String 13-9
13.2.8 Event 13-10
13.2.9 Selective Log 13-11
13.2.10 Event String 13-12

Chapter 14 Script 14-1

14.1 Start 14-1

14.2 Editor Screen 14-2

14.3 How to Use 14-5
14.3.1 Use of Internal Fuctions 14-5
14.3.2 Tag Name Delivery 14-6
14.3.3 Editor Debugging 14-8
14.3.4 How to use DLL 14-13
14.3.5 How to use OLE object 14-13
14.3.6 Error Handling 14-14

14.4 Runtime Debug 14-15

Chapter 15 Function 15-1

15.1 Funtion Inquire 15-1

Chapter 16 Action Trigger 16-1

16.1 Start 16-1

16.2 Configuration 16-2

16.2.1 General Information 16-2

16.2.2 Schedule 16-7

16.2.3 Tag Value Change 16-10

16.2.4 Alarm Status 16-12

16.3 Holiday Settings 16-14

Chapter 17 User Settings 17-1

17.1 Setting Screen 17-1

Chapter 18 Recipe 18-1

18.1 Component 18-1

18.1.1 Model 18-1

18.1.2 Unit 18-1

18.1.3 Data 18-1

18.2 Settings 18-2

18.2.1 Model Information 18-3

18.2.2 Unit Information 18-4

18.2.3 Data Information 18-6

18.3 Recipe Runtime 18-7

18.3.1 Function 18-7

18.3.2 Recipe Viewer OCX 18-7

18.3.3 Internal Function 18-7

Chapter 19 Database 19-1

19.1 Database Connection 19-1

19.1.1 Demo 19-1

19.1.2 Database Connection Settings 19-2

19.2 Database Operation 19-4

19.2.1 Operation Type 19-4
19.2.2 Demo 19-4
19.2.3 Database Operation Settings 19-5

19.3 Database Runtime 19-10
19.3.1 Function 19-11
19.3.2 Internal Function 19-11

Chapter 20 Report 20-1

20.1 Start 20-1
20.1.1 Report Menu 20-2
20.1.2 New 20-5
20.1.3 Report Property Settings 20-7

20.2 Layout 20-11
20.2.1 Report Layout 20-11
20.2.2 Sheet Property 20-12

20.3 Report Open 20-20
20.3.1 Edit 20-20
20.3.2 Report View 20-21
20.3.3 Refresh Auto Output Information 20-25
20.3.4 View History Information 20-25
20.3.5 How to Use during the Runtime 20-26

Chapter 21 Utility 21-1

21.1 Unit 21-1

21.2 Alarm Message Registration 21-3

21.3 On/Off Display Message Registration 21-5

21.4 LSIS Program Registration 21-7

21.5 LS PLC Tag Import 21-9

21.6 Application Registration 21-14

Chapter 22 Cross Reference 22-1

22.1 Function and Property 22-1

22.2 Cross Reference DB Creation 22-2

22.3 Engineering Function 22-4

22.3.1 Search by Tag 22-4
22.3.2 Search by Module 22-4
22.3.3 Filtering 22-5
22.3.4 Error Tag Search 22-6
22.3.5 Jump Function 22-7
22.3.6 Save File 22-9

22.4 Runtime Function 22-11

Chapter 23 Web 23-1

23.1 InfoU Web Configuration 23-1

23.1.1 Prerequisite 23-1
23.1.2 Configuration Procedures 23-2

23.2 Creating InfoU Web Site 23-2

23.2.1 Execution of the Navigation Management 23-2
23.2.2 Creating New Website 23-5
23.2.3 Get Project Information 23-12
23.2.4 Adding the Virtual Folder on the Existing Website 23-14
23.2.5 Creation of a Website in Addition to the Existing One 23-19

23.3 InfoU Website Configuration 23-25

23.3.1 Overview 23-25
23.3.2 Prerequisite 23-25
23.3.3 Login Screen 23-25
23.3.4 System Management 23-33
23.3.5 Monitoring Web User 23-41
23.3.6 Main Screen 23-45

23.4 Web HMI Server 23-51

23.4.1 Operation Settings 23-51
23.4.2 Server Settings 23-52

Chapter 24 Redundancy and Client Server 24-1

24.1 System Environment Settings 24-1
24.1.1 Cancel Network Security 24-1

24.2 Redundancy Settings 24-16
24.2.1 Procedure 24-16
24.2.2 Create Server Project 24-16
24.2.3 Project Property Settings 24-17
24.2.4 Redundancy Settings 24-18
24.2.5 RunTime Operation 24-24

24.3 Client-Server Settings 24-26
24.3.1 Procedure 24-26
24.3.2 Server Project Settings 24-26
24.3.3 Client Project Settings 24-31

Chapter 25 Mobile 25-1

25.1 Mobile Designer 25-1
25.1.1 Startup 25-1
25.1.2 Mobile Server Project 25-5
25.1.3 Mobile APP Page 25-8
25.1.4 Resolution Document 25-11
25.1.5 Object 25-16
25.1.6 Dynamic Properties 25-28
25.1.7 Service Settings 25-38

25.2 Mobile Server 25-42

25.3 Mobile Client 25-45

25.4 Mobile Network Settings 25-59

Appendix 1 Modbus Server A1-1

1.1 Modbus Server Editor A1-1
1.1.1 Execute Editer A1-1
1.1.2 Registered Information A1-2
1.1.3 Tag Setting A1-4
1.1.4 Copy / Paste Clipboard A1-8

1.1.5 Export to CSV file A1-11

1.2 Modbus Server Runtime A1-14

1.2.1 Execute Runtime A1-14

1.2.2 Runtime Menu A1-15

Appendix 2 LSIS InfoU OPC Server A2-1

2.1 Introduction A2-1

2.2 Install Program A2-1

2.2.1 Installation A2-1

2.2.2 Modify/Repair/Remove Program A2-5

2.3 Configuration of Default Screen A2-10

2.3.1 Configuration A2-10

2.3.2 Monitoring A2-13

2.4 OPC Project Configuration A2-15

2.4.1 Create New OPC Project A2-15

2.4.2 Open or Modify OPC Project A2-17

2.5 Run Monitoring A2-18

2.5.1 Connect LS InfoU OPC Server / Start Monitoring A2-18

2.5.2 Disconnect LS InfoU OPC Server / Stop Monitoring A2-20

2.6 DCOM Configuration A2-20

Appendix 3 OPC DCOM Configuration A3-1

3.1 Windows 7 Configuration A3-1

3.1.1 Common(Client&Server) Settings A3-1

3.1.2 Client Settings A3-11

3.1.3 Server Settings A3-15

3.2 Windows XP Configuration A3-23

3.2.1 Client Settings A3-23

3.2.2 Server Settings A3-33

Appendix 4 Visual Basic Script Help A4-1

4.1 Script Function of Microsoft Script Engine A4-1

Appendix 5 Function A5-1

5.1 Function Classification A5-1
5.1.1 List by Item A5-1
5.1.2 List by Alphabetic Order A5-6

5.2 Function Descriptions and Examples A5-11
5.2.1 Project Function A5-11
5.2.2 Screen Function A5-14
5.2.3 Tag Function A5-17
5.2.4 Script Function A5-22
5.2.5 Event Function A5-23
5.2.6 Log Function A5-25
5.2.7 Network Function A5-28
5.2.8 System Function A5-31
5.2.9 Recipe Function A5-35
5.2.10 Database Function A5-42
5.2.11 Alarm Function A5-42
5.2.12 External Device Function A5-52
5.2.13 User Function A5-53
5.2.14 Report Function A5-55
5.2.15 History Data Function A5-57
5.2.16 I/O Driver Function A5-63

Appendix 6 InfoU Web Q&A Collection A6-1

6.1 Q&A for Windows XP A6-1

6.2 Q&A for Windows 7 ..... A6-14

Chapter 1 XGT InfoU

As a PC HMI(Human Machine Interface) software package for collecting real-time data on the device, setting up a data transmission system and playing a role of a central observer, InfoU allows the operator to monitor/analyze the overall operational status of the device in real time.

This software provides various useful functions that can run in the latest operation systems such as Windows XP, Windows 2000, Windows Vista, Windows7 and enhances the reliability, fast processing speed and convenient use of the automatic system.

Data management function of this monitoring software helps the user to collect, store and calculate data on the device's operation status and to respond to requests for data. Its monitoring function also provides real-time data along with graphs configured by the operator on the display screen.

1.1 Special Feature

1.1.1 Open Structure

Its open structure with excellent extension and compatibility makes it possible to provide stable communication and effective interface with other systems.

● Supports industrial standard interface such as 1OLE DB, OPC Server/Client

● Easy access to DBMS (Data Base Management System)

Its database connection or work function ensures easy input or output by using tag values of universal databases such as MS, SQL or Oracle and efficient data sharing with MES or ERP system.

● It can be referred by other applications with OLE automation.

● It supports edition and compatibility of MS office software especially PowerPoint of the graphic editor for easy graphic works

● It also provides ActiveX component graphic objects and allows the user to use other company's components for his/her graphic works.

1.1.2 Efficient Project Management

InfoU project is treated as the unit subjected to engineering and real-time operation management and under each project unit, sub-paths such as tag, logging, alarm, graph, report and script are created and managed.

Therefore, it becomes much easier to move or copy a certain project to other system, drive or path.

The user can easily manage numerous projects while easily copying and transferring contents between projects and efficiently conducting engineering works.

In addition, the user can selectively execute only the modules he/she wants since each function works individually in its own execution unit,

1.1.3 Tag Engineering

It provides efficient engineering and convenient operation by providing two perspectives of tag management criteria.

(1) View by device
It is very convenient to use since device set-up and tag work can be executed at the same time as a tree-type tag explorer according to device configuration.

(2) View by group
Logical tag groups are displayed in a window explorer type to allow the user to manage tags according to their purposes from the point of operator's view.

(3) Import/Export CSV
The user can output, add, edit or delete a large amount of tags by using CSV (Comma Separated Values)-type files editable with Excel or general text editors (notepad)

(4) Expression
Not only numeric expression but also logical expression can be used with memory tag values

(5) LS PLC Tag Sharing
GLOFA-LSIS's PLC-shares its program tool GMWIN and engineering files to enable automatic input to InfoU tags.

1.1.4 I/O Driver Convenience

(1) Tag Simulator
Even though there is no device in place actually or it is before communication service is open, it can provide various rules of data to tags and significantly reduce debugging time since it can run the same applications with those in the site.

- Analog Simulator: Random, sign wave, chopping wave, saw tooth wave, control function
- Digital Simulator: Random, change, control function
- String Simulator: Random, control function
- XG5000 Simulator: For XGT PLC's tags, you can use the simulation function of XG5000 that is the program tool.

(2) Create auto block
Even though the user does not set up additional communication blocks, it automatically creates blocks according to tag's I/O address setting when runtime runs.

(3) Automatically register system tag when adding a station
When registering a station, it also registers its communication status automatically to allow the user to check the station status and its communication performance.

1.1.5 Upgraded Alarm Function

Its improved alarm function using alarm group delivers more efficient problem solving.

(1) Manage alarm group
It can manage alarm groups of the tag designated by the user to execute alarm sound, printing, sending mails and SMS by the level of the alarm generated in the relevant tag.

(2) Batch assignment of alarm rank
If any alarm rank is not assigned in the tag setting, the user can assign a certain rank uniformly to all of the

alarms in a group by using this batch assignment function.

(3) Various analog alarms

Since Limit alarm, Gap alarm and Variation alarm occur for one tag at the same time, each process can be closely and thoroughly monitored.

(4) Change dynamic setting

Dynamic change in alarm's limit value, gap value, variation value and other setting values enables more flexible action processing.

(5) Convenience of alarm check

You can check alarms on the graphics screen using the Alarm viewer OCX. You can use mini alarm, backup alarm check, alarm filtering function, etc. for convenient and efficient management.

(6) Alarm-Linked to Action Trigger

You can realize various functions by specifying trigger operations linked with alarm-action trigger-script.

1.1.6 Convenience of Logging Management

The user can create and manage a data logging group for major data that need history management according to his/her purpose and assign one tag data to multiple logging groups.

(1) Type of logging group

Divided into Cycle, Statistics, Accumulation and String

(2) Save in excel file

Logging data inquired from trend and list trend can be immediately saved in excel files.

(3) Inquire logging data

Data can be inquired from trend, report and list trend and API, which can be inquired from script, is provided

(4) Automatically calculated statistics/integrated data

Statistics/integrated data are created without the use of additional script or application.

(5) Data to inquire according to type of history model

- Cycle: Analog (current value), Digital (current status), String
- Statistics:
  - Analog (current value, min value, min time, max value, max time, average value)
  - Digital (ON count, running time, change count)
- Accumulation: Analog (change amount, current value)

(6) Report

Daily report, monthly report, temporary report and etc can be generated. The user can easily select an Excel form and register data in the form. Data of logging group can be inquired and automatically printed from Report and this Report function provides copy of report model, drag & drop report tag, clipboard copy/paste,

(7) Backup

It is possible to backup all of the collected groups and backup files can be inquired from trend and list trend.

1.1.7 Define VBS and Action Trigger

It supports Action Trigger and VB Script for preparing robust and flexible applications

(1) Script Editor

It supports VB script, uses external OLE individuals such as ActiveX and registers and uses DLL's API.

(2) Action Trigger

It integrates execution action conditions of script, recipe and database work together to manage them and trigger actions. Changing tag value, generating alarm and action by schedule can be set up.

Also efficient management is possible by assigning one action to multiple tasks.

(3) Provide recipe

Recipe is used when the same production/process line generates various products according to the prescribed vales and this function is applicable to various fields including flexible manufacturing system and lighting control.

(4) Execute recipe

A recipe is provided by its exclusive graphic object and can be executed and controlled on the run time graphic screen. In addition, execution can be defined from Action Trigger. Providing API available in Script ensures control at more diverse situations.

(5) Transmit tag name with parameter value.

It is possible to implement effective applications by assigning various action triggers for script execution and transmitted tag name to script function with parameter value.

1.1.8 Robust Graphic Engine

(1) Symbol Library

It supports more than 1000 symbol library images and its drag & drop provides a more convenient usage way..

(2) Support various types of graphic files

BMP, GIF, JPEG, PNG, TIFF, ICON, WMF, EMF, Animation GIF and other various graphic files are supported.

(3) Engineering Template

A certain screen can be more frequently and conveniently reused with its template object and plant screens and transmitted parameter values.

(4) Provide exclusive graphic objects for stable extension:

Alarm view (real-time, history), trend (real-time, history), recipe view, list trend (real-time, history) and gauge control are provided for convenient use.

(5) Provide accurate and fast screen mobility

The user can display part of the screen or the entire screen or move to the work area he/she want with Zoom, panning or navigation function to more accurately and fast identify the relevant process.

(6) Layer

Hide/UnHide and Lock/UnLock functions can be used after selecting a graphic object.

(7) Built-in Graphic Script

When it is insufficient with the screen settings only because of complicate dynamic components of the graphic screen, various functions can be used from the built-in scripts and internal functions of the graphic editor.

(8) ActiveX Control

Direct control over the Company's unique control functions such as Alarm Viewer and Trend and other company's ActiveX enables efficient and high-functional graphic works. It supports virtual keyboard and window to input various values for panel PC.

(9) Use the prepared drawing screen.

It provides the template screens for frequently used functions to improve engineering workability remarkably. When clicking the mouse, it can display the prepared real-time/history trend, list, report retrieval screen, etc. without engineering works.

(10) Template screen

You can make the customized template screen; namely, you can draw the graphic works of the standardized device through Drag & Drop to improve productivity.

(11) Added convenient functions

Batch change of tag link and string, digital tag control (pulse control, momentary control), image button and image lamp functions are added.

1.1.9 Cross Reference

(1) Simple correction

With Cross Reference, the user can trace the interrelation and immediately correct engineering error without making unnecessary effort to find each error in the tag and the drawing to change or delete it.

(2) Accident Analysis during the run time

When alarm occurs during the run time, the user can use Cross Reference in the alarm screen and the alarm pop-up screen to move to the relevant plant screen or Trend and analyze the accident fast and conveniently.

1.1.10 Web Server

(1) Simple Web Conversion

In the InfoU Web server, existing projects can be automatically converted to the web through the Wizard function so that even a user who does not have any expertise on web server or website configuration can easily configure a website.

(2) Monitoring system without restriction to place

InfoU is possible to control monitoring only with web browser and monitor history data through Trend as well as real-time tag data without additional installation or restriction to place.

(3) Configuration with various types

The user can create a new web site by configuring a site he/she wants and form a link connection by adding virtual folders to existing web sites.

1.1.11 Mobile Monitoring System

(1) Graphic editor for mobile devices

Through the graphic editor for mobile devices, a user can make the plant screen that can be monitored in the smartphone. It also provides various dynamic properties and graphic library.

(2) Various functions

It provides various functions; monitoring through a plant screen, monitoring for registered tags, alarm monitoring, trend viewer, etc.

(3) Mobile app

It provides the client operating in the smartphone environment.

1.2 System Requirement

At least the following environment is required to execute this program;

- IBM-PC compatible computer with Intel Pentium 4 or higher
- Memory of 1GB or higher
- VGA graphic card and monitor with basic resolution 1024 X 768 or higher
- Hard disc space required of 10GB or more
- Port of USB 1.0 or higher
- Microsoft Windows XP/2003/Vista/Win7
- Microsoft Excel 32bit Verstion (MS Office 2003/2007/2010/2013)
- Ethernet Card 100Mbps higher
- Standard Serial Port

1.3 License

(1) Using USB lock key

- The use of the project may be restircted depending on the following conditions (1-hour use)
- Restriction to the number of I/O tags: When more number of I/O tags is using than the designated number to the key
- Project option: When using general local license key for web option project
- Web version: Restrict to the number of concurrent users

(2) Demo Mode

- If the user uses excessive functions than the assigned ones to the lock key, demo mode will run for one hour.
- Example of use 1: If you run more than 500 I/O tags with the lock key for 500 tags, the program will end after one hour of run time and engineering tool execution.
- Example of use 2: If you set up the web project with the lock key for a signal system, the program will end after one-hour run time.

1.4 Product Composition

Classification	Production Name	I/O Tag	Other
Developer Version (Engineering+Real-time operation)	InfoU-RC-75	75 tags	
	InfoU-RC-150	150 tags	
	InfoU-RC-500	500 tags	
	InfoU-RC-1K	1000 tags	
	InfoU-RC-2K	2000 tags	
	InfoU-RC-MAX	64000 tags	
Executor Version (Real-time operation)	InfoU-RT-75	75 tags	
	InfoU-RT-150	150 tags	
	InfoU-RT-500	500 tags	
	InfoU-RT-1K	1000 tags	
	InfoU-RT-2K	2000 tags	
	InfoU-RT-MAX	64000 tags	
Web Version (Engineering + Real-time operation + Web)	InfoU-OP-WEB2	Unlimited tags	Concurrent users: 2 Users
	InfoU-OP-WEB5	Unlimited tags	Concurrent users: 5 Users
	InfoU-OP-WEB10	Unlimited tags	Concurrent users: 10 Users
	InfoU-OP-WEBMAX	Unlimited tags	Concurrent users: Unlimited Users
	InfoU-OP-WEB5/150	150 tags	Concurrent users: 5 Users
	InfoU-OP-WEBMAX/150	150 tags	Concurrent users: Unlimited Users
	InfoU-OP-WEB5/500	500 tags	Concurrent users: 5 Users
	InfoU-OP-WEBMAX/500	500 tags	Concurrent users: Unlimited Users
	InfoU-OP-WEB5/1K	1000 tags	Concurrent users: 5 Users
	InfoU-OP-WEBMAX/1K	1000 tags	Concurrent users: Unlimited Users
	InfoU-OP-WEB5/2K	2000 tags	Concurrent users: 5 Users
	InfoU-OP-WEBMAX/2K	2000 tags	Concurrent users: Unlimited Users

1.5 Caution

(1) Please, be cautious to keep and retain the enclosed license key.
Any lost license key will not be reissued and some program operations and functions may be restricted depending on the license key.

(2) This software shall be operated only in the authorized OS and installation environment.
The entire or part of the program may generate some errors or malfunctions if it is installed and used in any unauthorized OS.

(3) Checklist when any error occurs during the program installation or the initial execution
Please, be aware that some program errors may occur because the vaccine program in your computer may block the program's communication port or service.

(4) Check your history data volume
Accumulation of history data such as alarm history and data history may cause insufficient storage space in the system and lead to the occurrence of error.

(5) When carrying out engineering works, take into account the system performance
If you register and use too many tags or log in excessive history data over the system performance, the program may become significantly slower or generate malfunctions during the engineering or run-time operation.

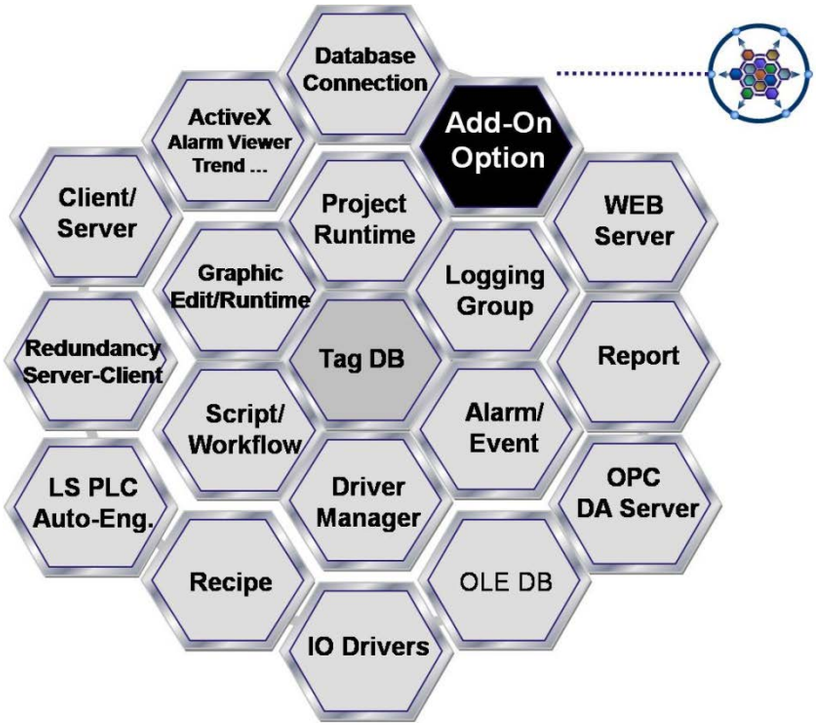
(6) Be cautious when preparing a script
When preparing a script, excessive system load or malfunction may occur due to the script which is omitted from unlimited loop by logical error or the one that requires long processing time.

(7) Be cautious about logging data loss when adding a tag after deleting it
There is no problem with the logging setting even though you delete a tag, which is set up in the logging group, from the tag database and then add it again with the same name, however, you may loss some logging data.
It is better to work with tag edition or CSV file update function rather than adding a tag after deleting it when changing tag attributes,

(8) Instruction for use in Windows
Common (Windows XP, 2003, Vista, Win7): Installed and executed by the administrator
Windows Vista, Win7: Cancelation of user-defined options

Chapter 2 System Composition

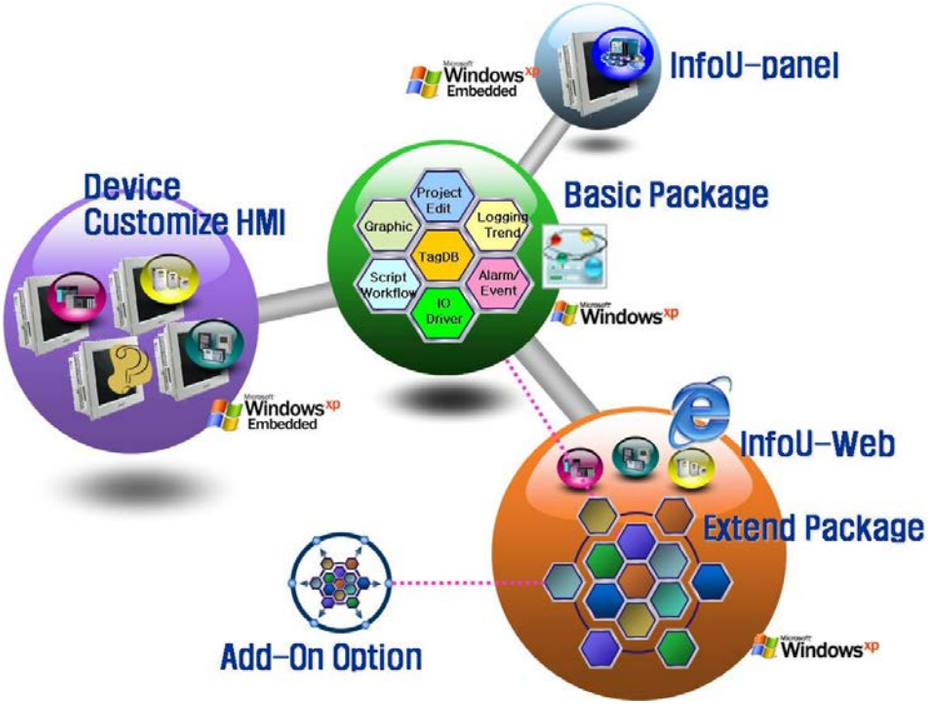
2.1 Software Component



InfoU software is made up of the components as seen in the figure above and the structure of this software allows the user to add other various components to its basic ones and those newly added components are internally connected through data communication and API to act dependently on each other. .

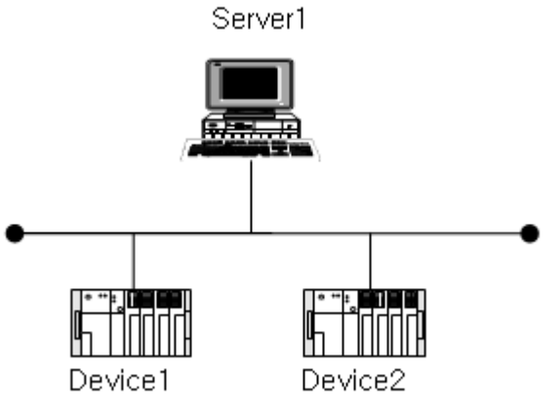
In the engineering environment, integral engineering tool InfoUD and graphic editor InfoUG are provided in additional execution files and Report is provided as an additional Excel function.

In the run time environment, the components above and I/O drivers are activated by additional execution files and play a each one's role depending on the engineering setting environment.



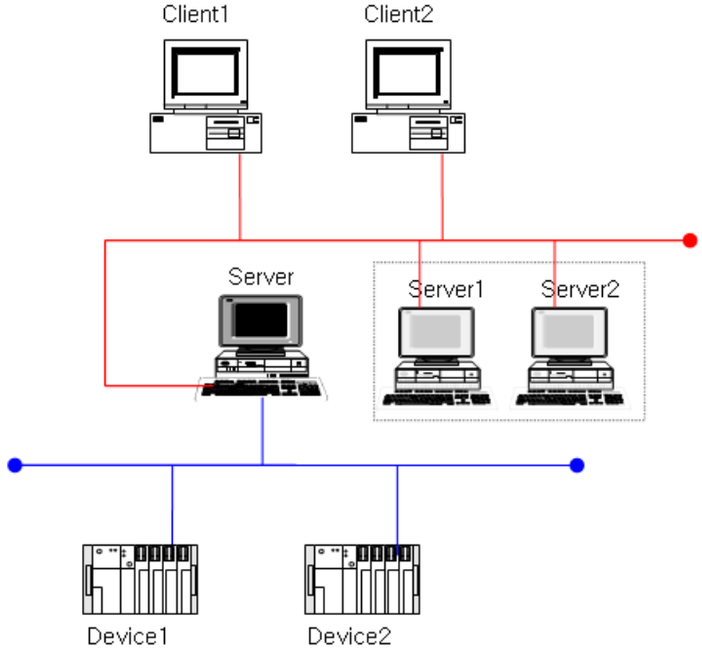
Various types of software packages can be provided by re-composing InfoU's functions and options as seen in the figure above.

2.2 Stand Alone System



● Only with this stand alone system, the user can monitor and control the data status of multiple devices.

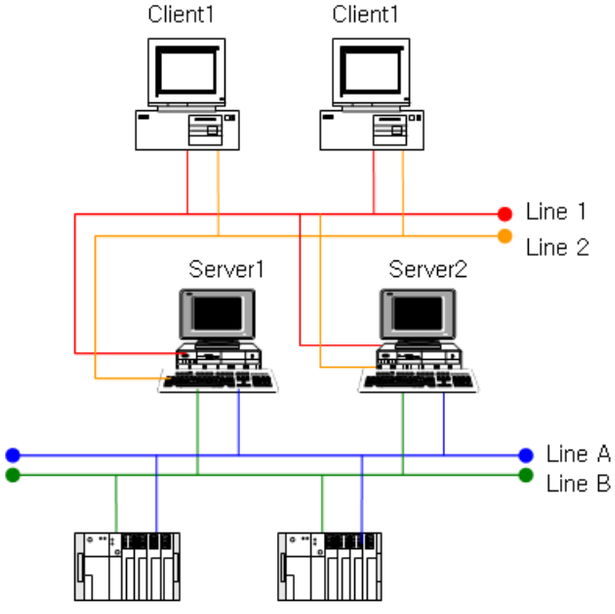
2.3 Client/Server System



● Remote/multiple monitoring and controlling the system is available.

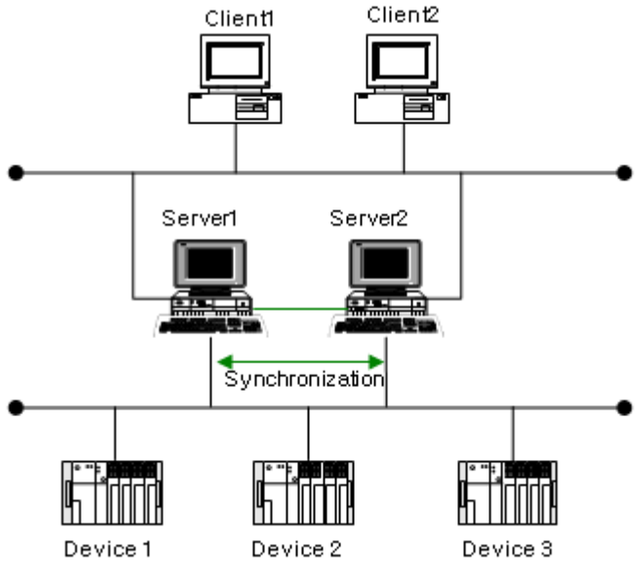
● Time synchronization between server and client is provided.

2.4 Line Redundancy



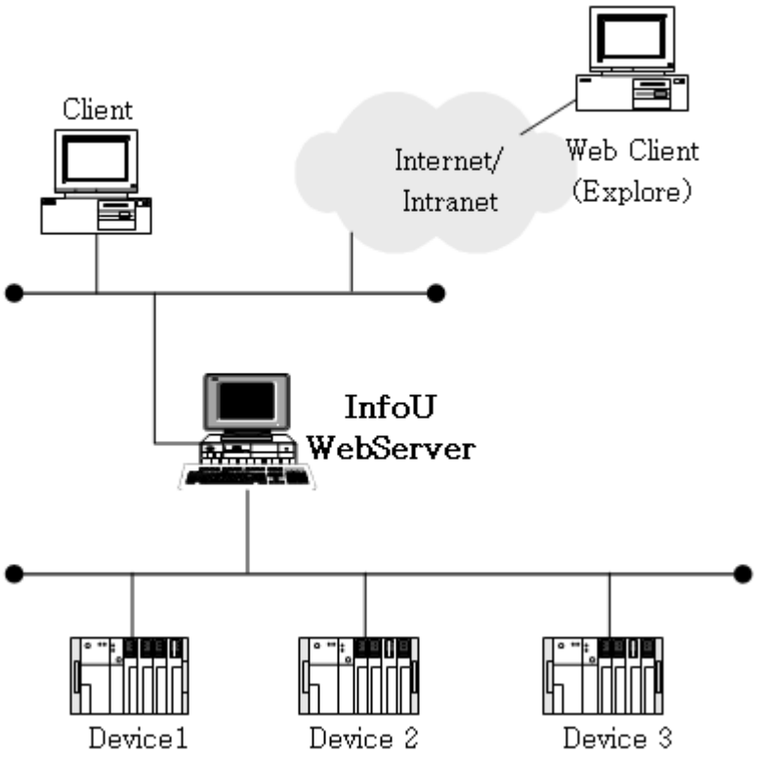
- Line redundancy with the device port such as PLC is available.
- Ethernet port redundancy of Server 1, Server 2 and Client
- An additional switcher is needed for signal communication between system and device,

2.5 Redundancy System



- When the Run Server is failed, the Standby Server will be activated through auto switching to provide a reliable service.
- Manual switching: The manual switching is avaiable in menu operations and script.
- Time synchronization can be made among Server 1, Server 2 and Client.
- Monitoring to check for any error of the partner server is available through RS-232C communication line.
- Project duplication: A redundancy project can be automatically duplicated to prevent error and to provide convenience to the user.
- For signal communication between system and device, an additional switcher is needed.

2.6 Web Server

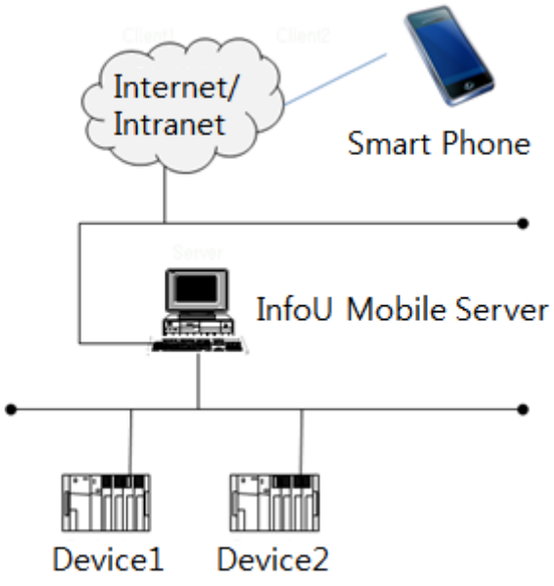


● Without any need to conduct additional engineering works for the projects in the InfoU server, real-time monitoring and controlling is available in the Web Client.

● Without installing any additional client program, monitoring is available only through MS Explore web browser.

● Data such as Trend (real-time history), Alarm (real-time history) can be monitored simultaneously as if the user is in the site.

2.7 Mobile Monitoring System



- It provides the graphic editor with which you can make a plant screen for the smartphone.
- Through a plant screen, monitoring, tag monitoring, alarm monitoring, real-time trend monitoring are available in the smartphone.

Chapter 3 Install and Start

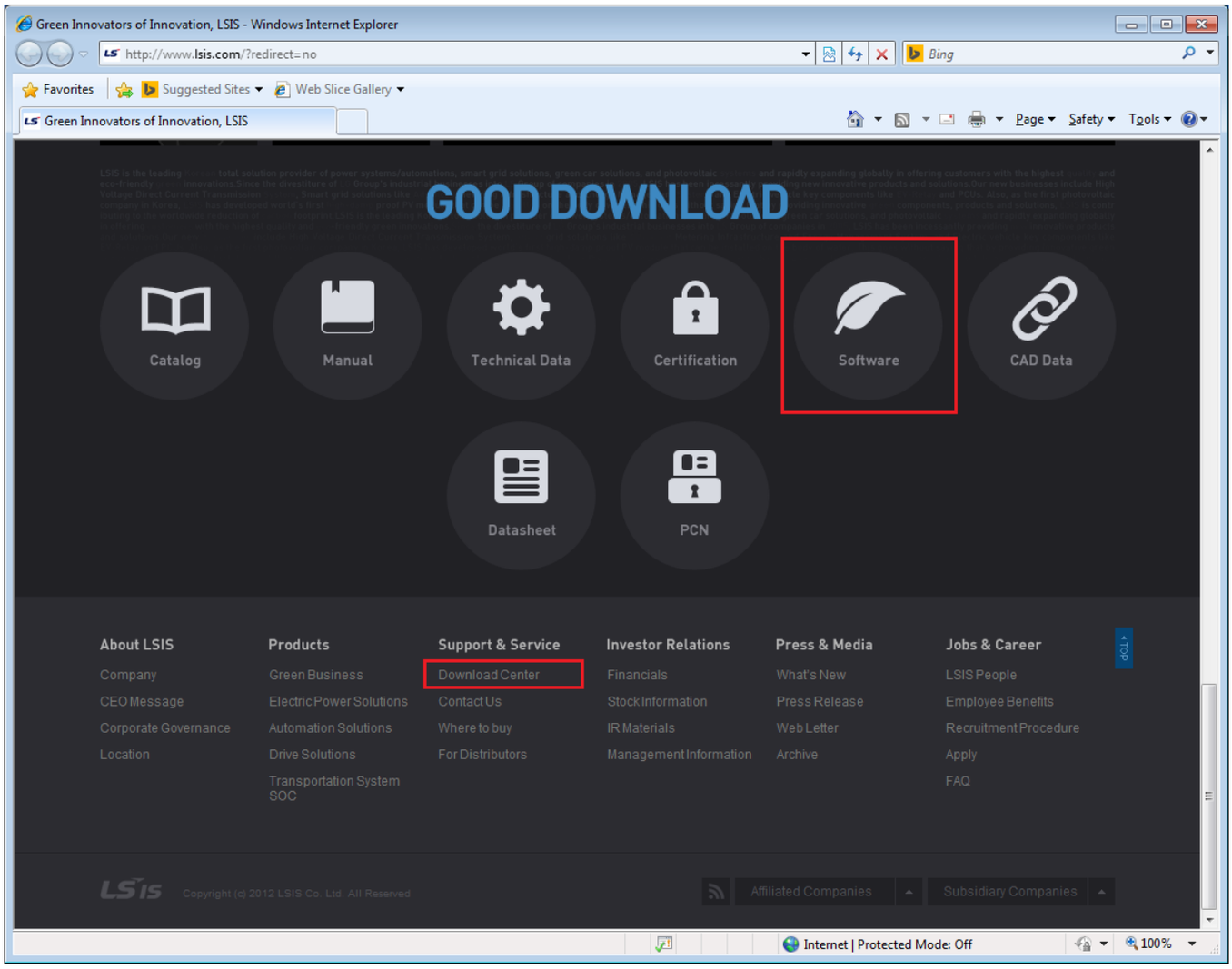
3.1 Installallation

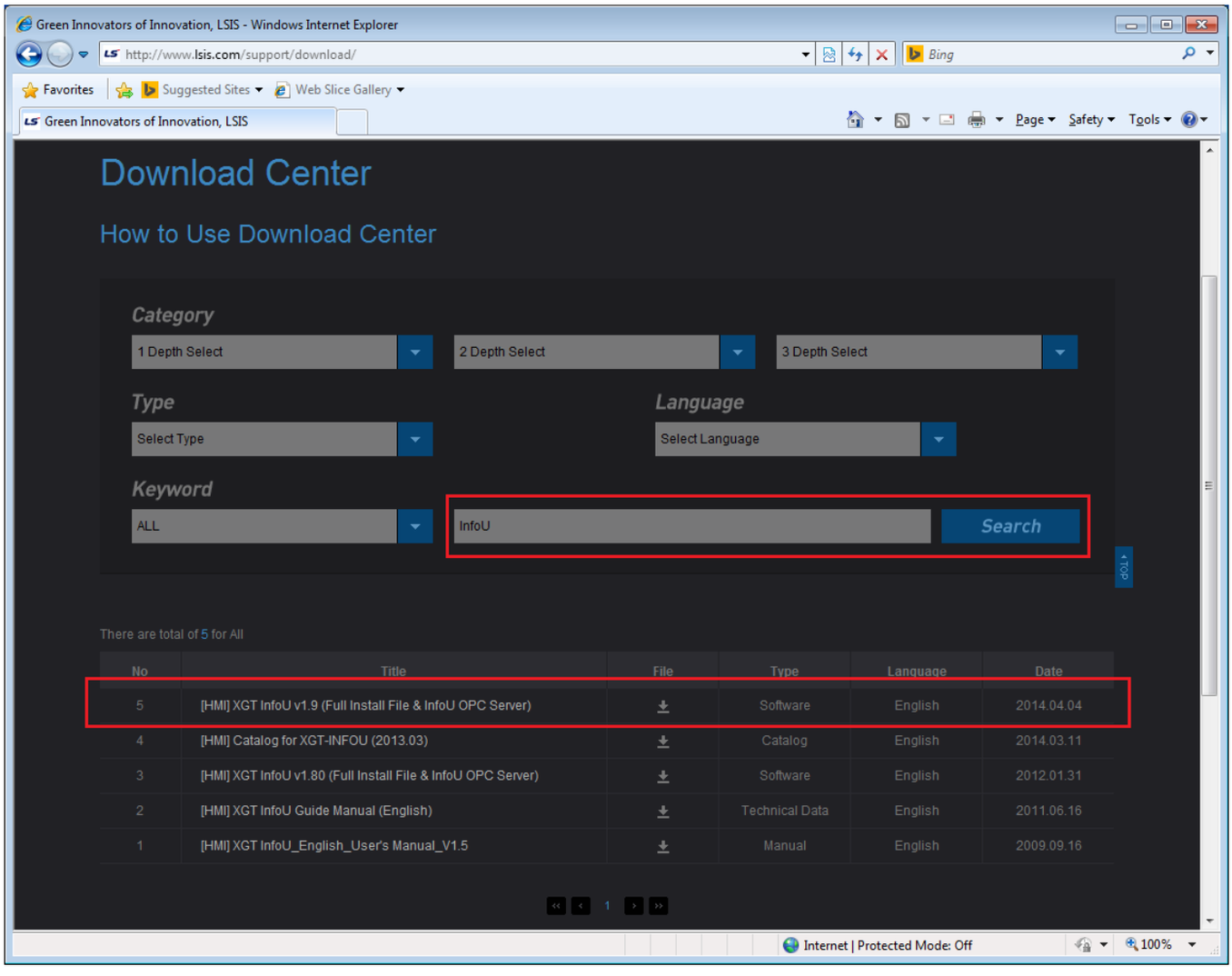
3.1.1 Arrange Installation

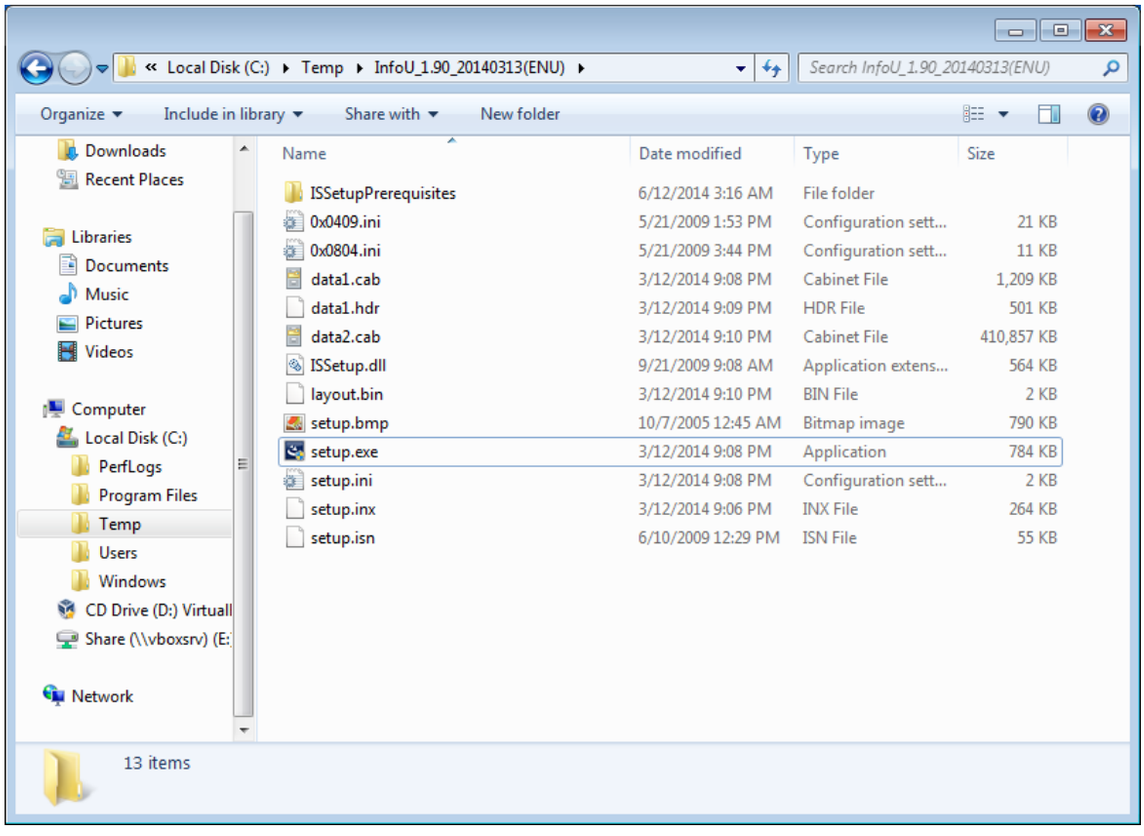
Once the setup CD are inserted in the computer's CD/DVD input device, the installation screen automatically appears in the user's computer as seen in the splash screen below.

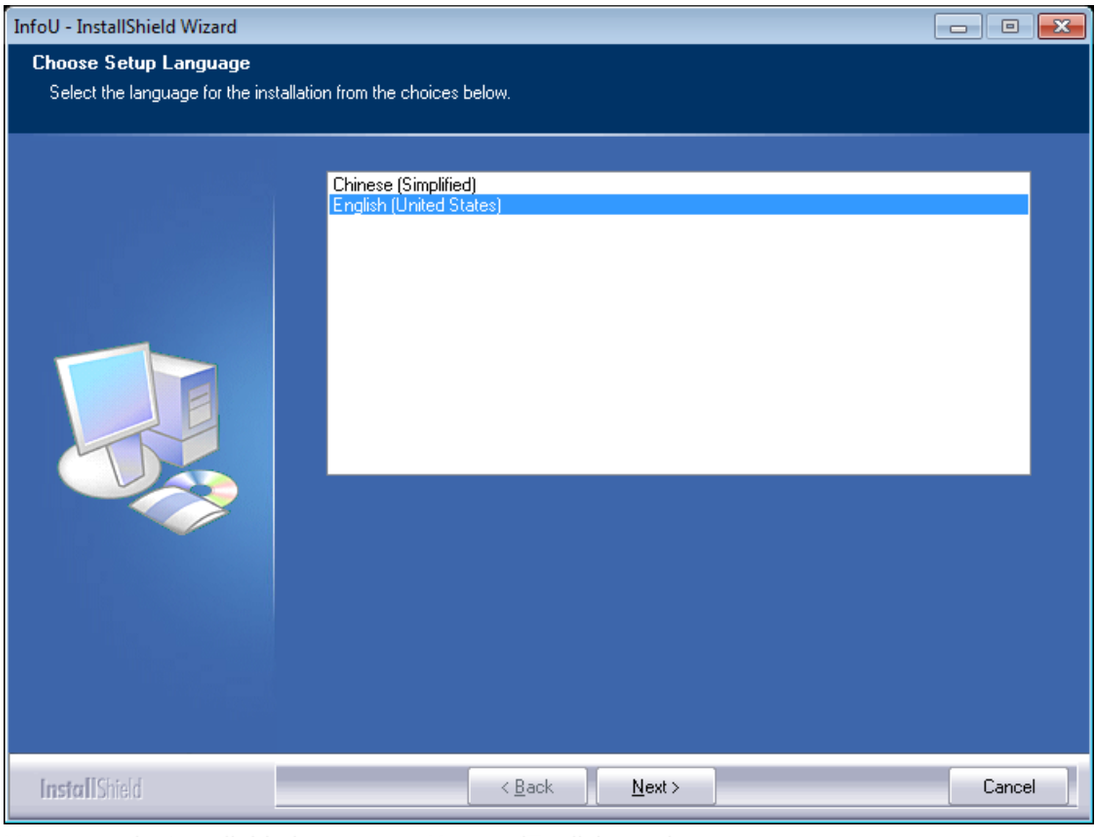
If it is not automatically executed, execute the CD's setup.exe file in the explorer.

Otherwise, the latest version of InfoU is posted on our homepage (www.lsis.com) so feel free to download it and unzip a file for use.

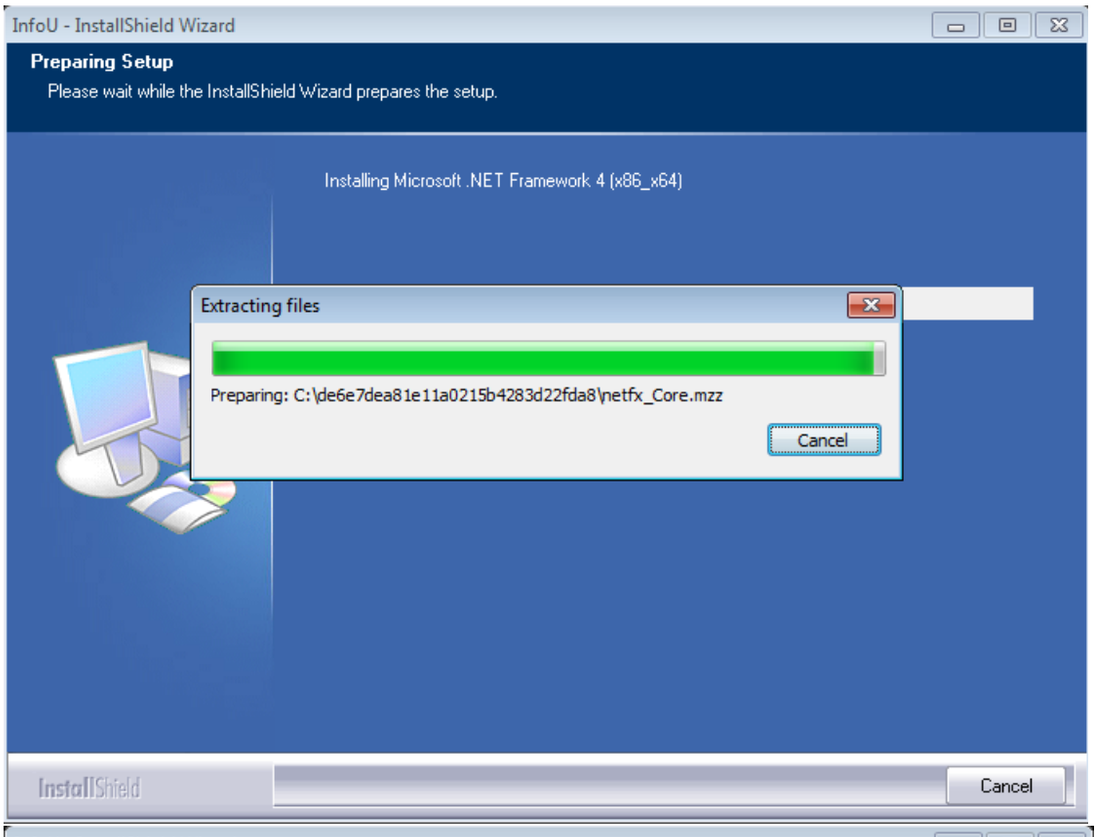


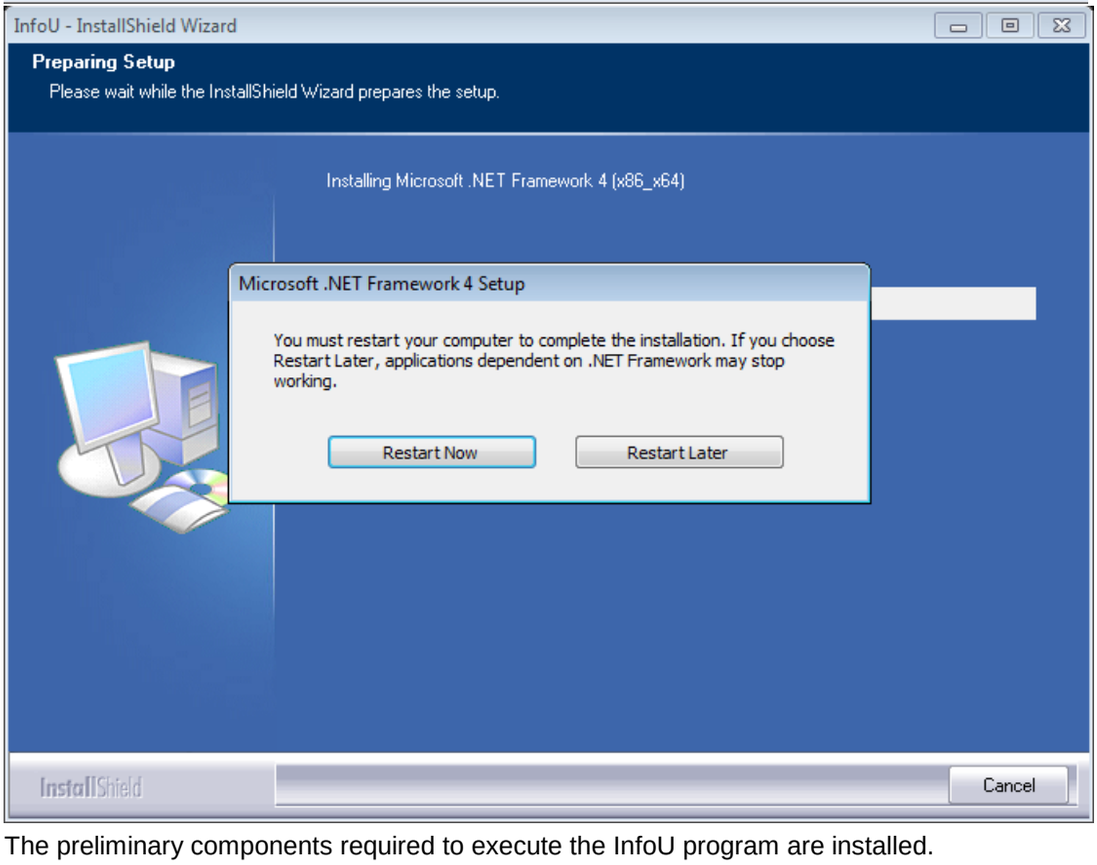

Green Innovators of Innovation, LSIS - Windows Internet Explorer
http://www.lsis.com/support/download/
Bing
Favorites
Suggested Sites
Web Slice Gallery
Green Innovators of Innovation, LSIS
Page
Safety
Tools
Download Center
How to Use Download Center
Category
1 Depth Select
2 Depth Select
3 Depth Select
Type
Select Type
Language
Select Language
Keyword
ALL
InfoU
Search
TOP
There are total of 5 for All
No
Title
File
Type
Language
Date
5
[HMI] XGT InfoU v1.9 (Full Install File & InfoU OPC Server)
Software
English
2014.04.04
4
[HMI] Catalog for XGT-INFOU (2013.03)
Catalog
English
2014.03.11
3
[HMI] XGT InfoU v1.80 (Full Install File & InfoU OPC Server)
Software
English
2012.01.31
2
[HMI] XGT InfoU Guide Manual (English)
Technical Data
English
2011.06.16
1
[HMI] XGT InfoU_English_User's Manual_V1.5
Manual
English
2009.09.16
1
Internet | Protected Mode: Off
100%





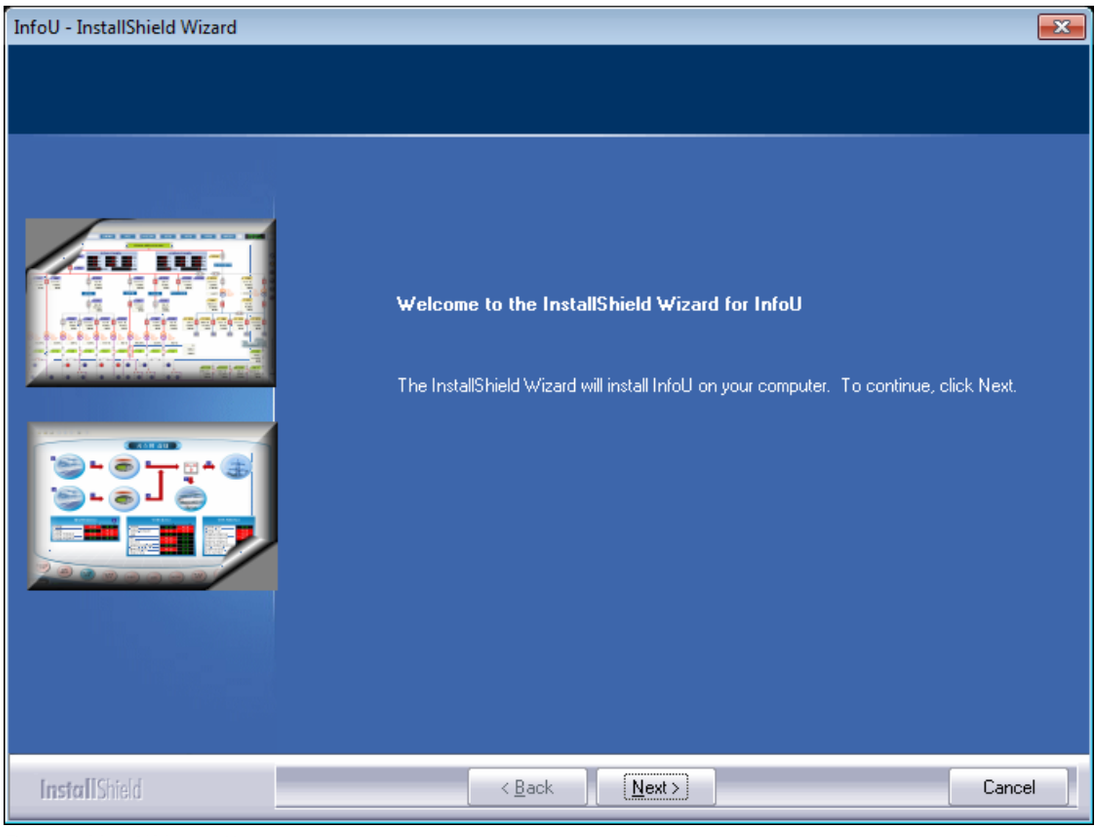
You can select available languages; Korean /English version.





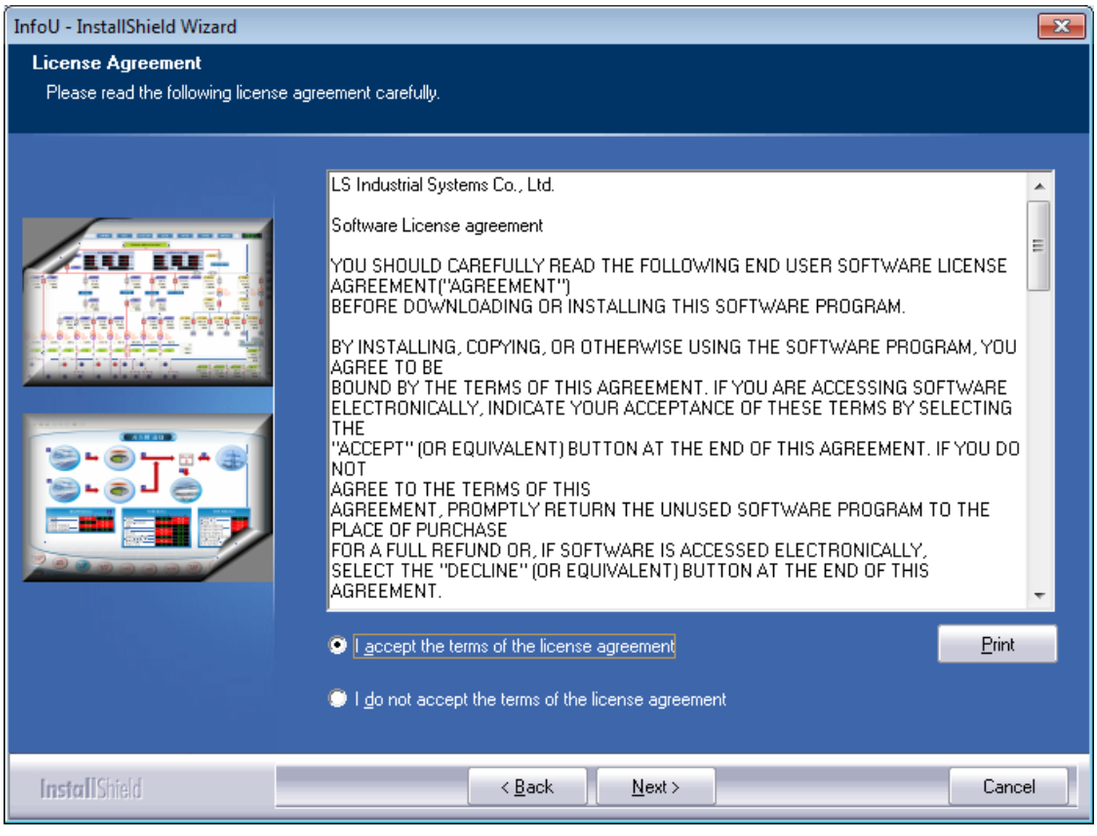
The preliminary components required to execute the InfoU program are installed.

The installed preliminary components are “Microsoft Visual C++ 2008 Redistributable Package” and “HASP License Key”, “Microsoft .NET Framework 4” installation program.



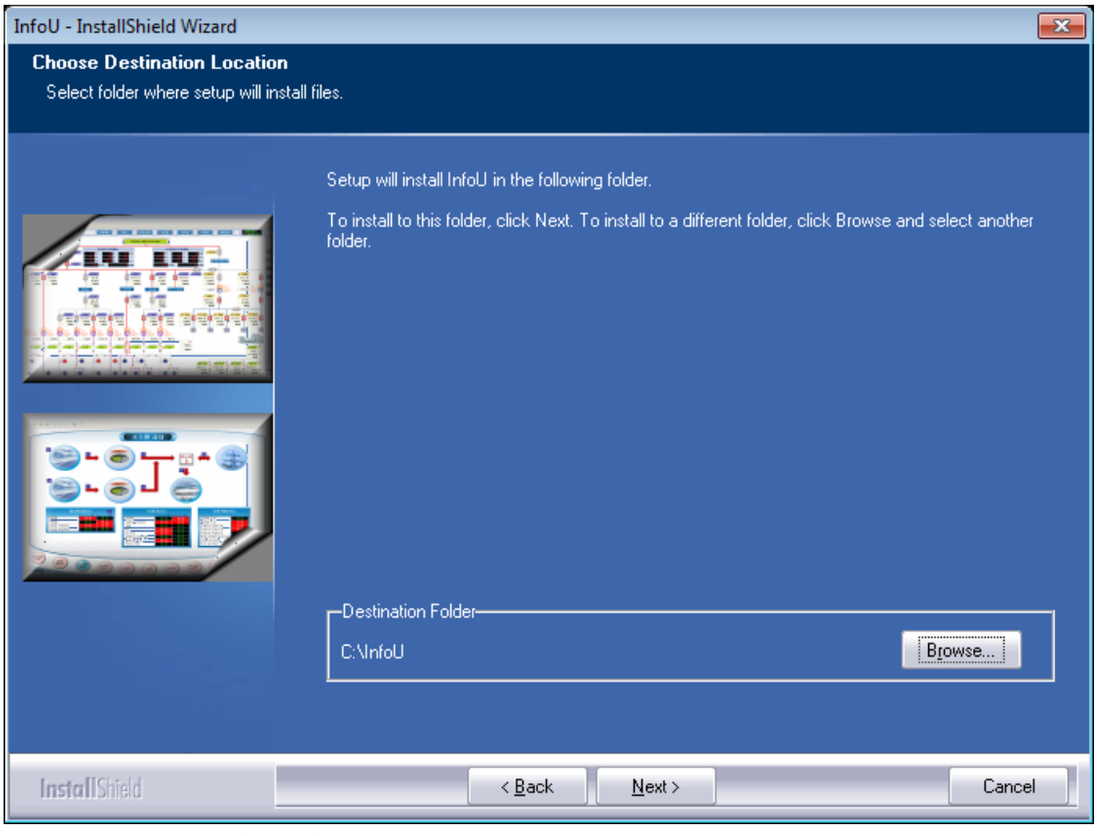
Press **[Next(N)]** button

If you want to change the previous screen's setting, press **[<Back (B)]** button or, if you stop installation, press **[Cancel]** button. Then, the installation work will stop.



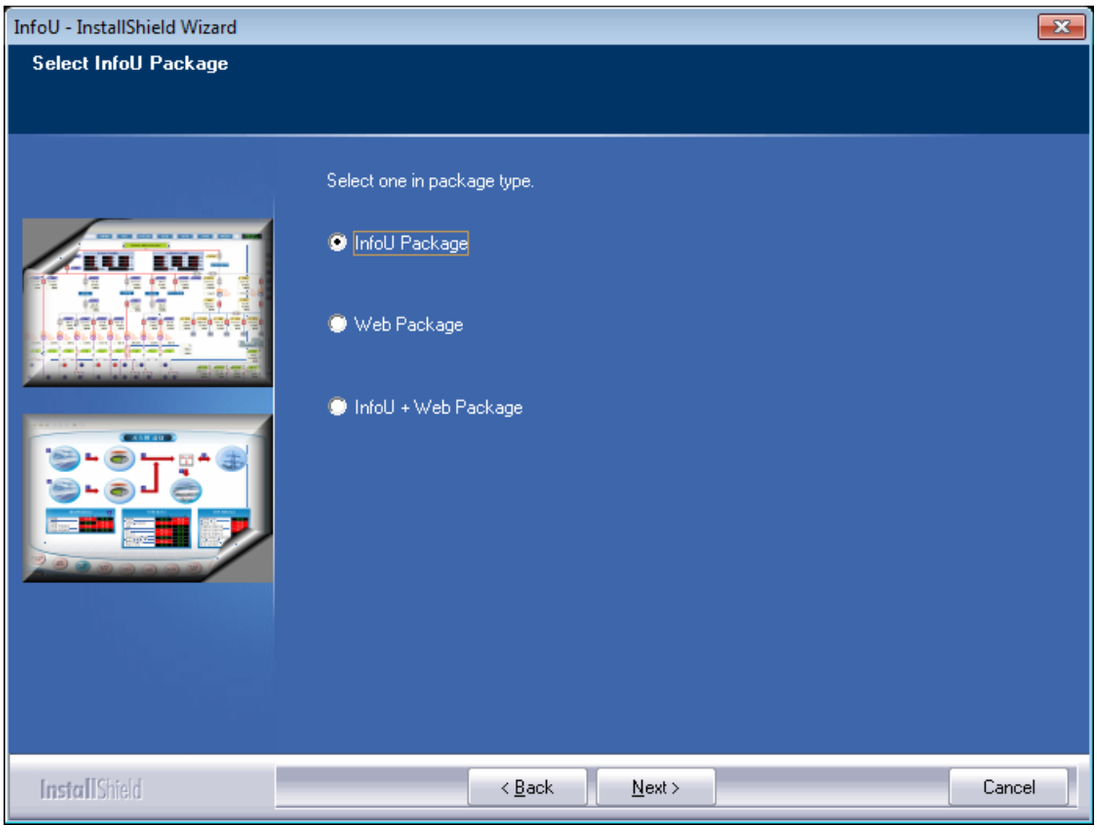
This is License Contract.

In case that the printer is already set up in the system, the terms and conditions of the license is printed upon the click on **[Print (P)].** Only after you press the radio buttone of "I agree on the terms and condtions of this license", **[Next (N)>]** button becomes active and it moves to the next screen.



Then, you may select a path to install.

3.1.2 Select a Package to Install

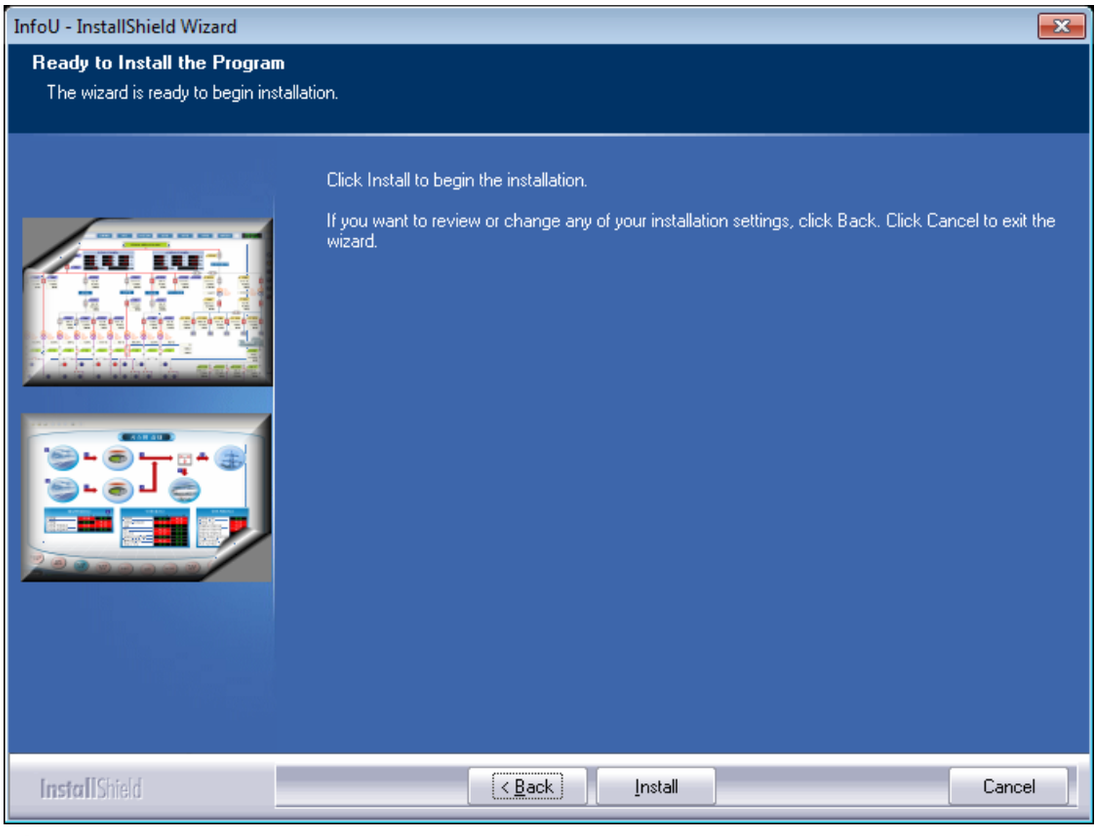


There are three types of packages you may install.

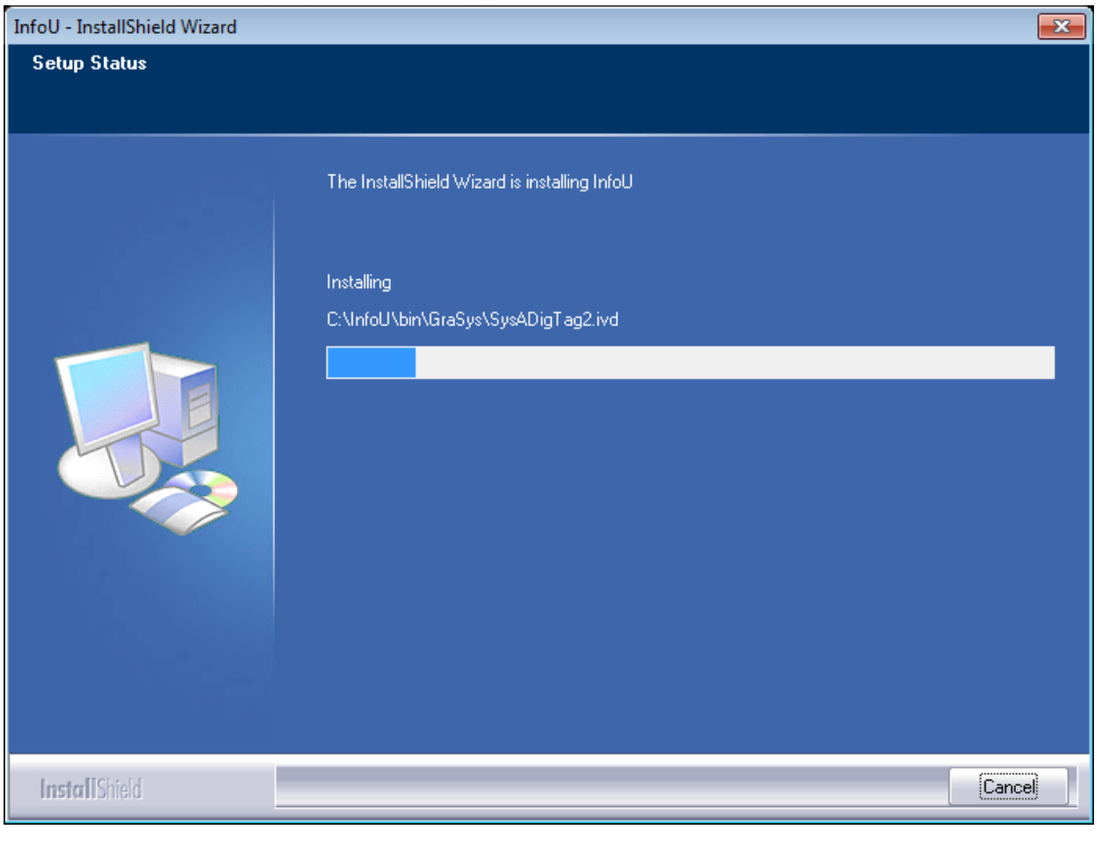
- InfoU Package - Package that consists of a wider range of applications by using InfoU - LS Industrial System's universal HMI software package.
- Web Package – InfoU's extend package that allows the user to monitor and operate the screen functions presented by InfoU with Internet browser.
- InfoU + Web Package - Package that allows the user to configure No 2 and No 3 packages in one computer.

Click **[Next (N)>]** button after selecting one,

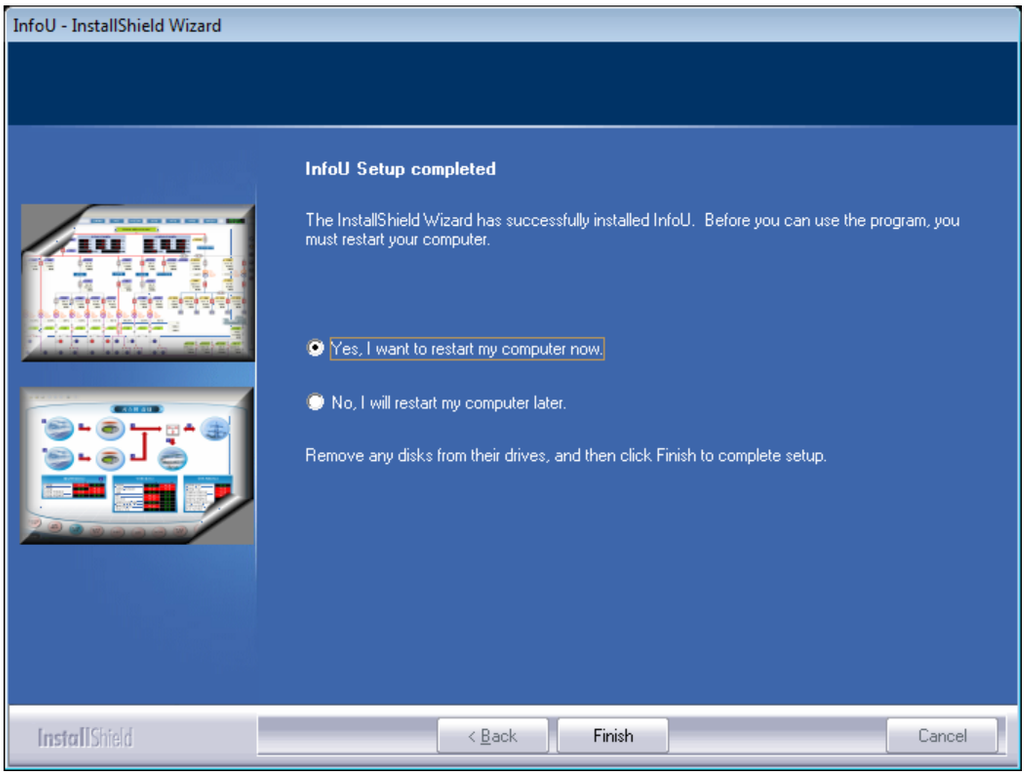
3.1.3 Start Installation



After completing settings, press **[Install (I)]** button to start installation.

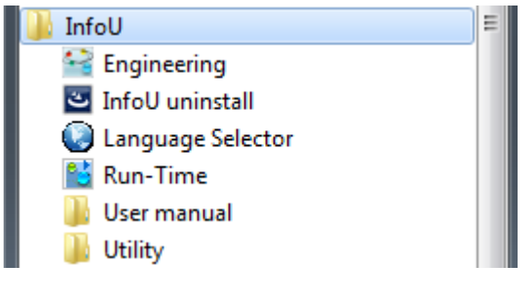


3.1.4 Complete Installation

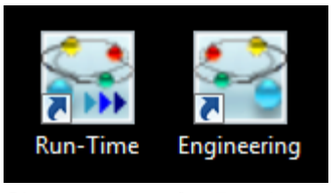


Once installation is completed, the computer shall be restarted for the program's normal execution.
Select the radio button "Yes, I will restart the computer now" and press [Complete] button, the system will restart after few minutes. Once the system restarts, all of the installation works are completed.

3.1.5 Menu and Shortcut



If installation is normally completed, [User Manual] [Run-Time] [Engineering] and [Language Selector]menus are created in the new "InfoU" program list.



Shortcut icons for Runtime and Engineering are created in the desk top.

3.2 Start and Stop

3.2.1 Start InfoU

(1) How to execute engineering (InfoUD)

Select [Engineering] from the program menu to execute InfoUD- InfoU's integral engineering tool.

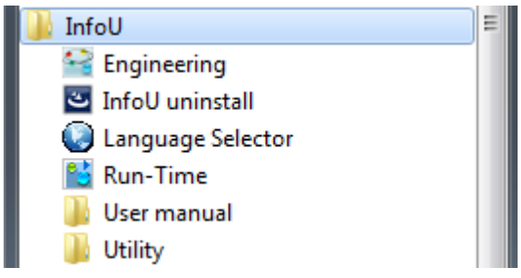
You may also click and execute the shortcut icons generated during the program installation.

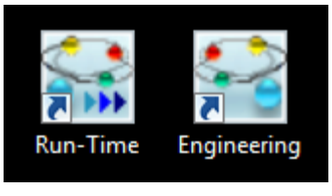
(2) How to execute run time (InfoUD)

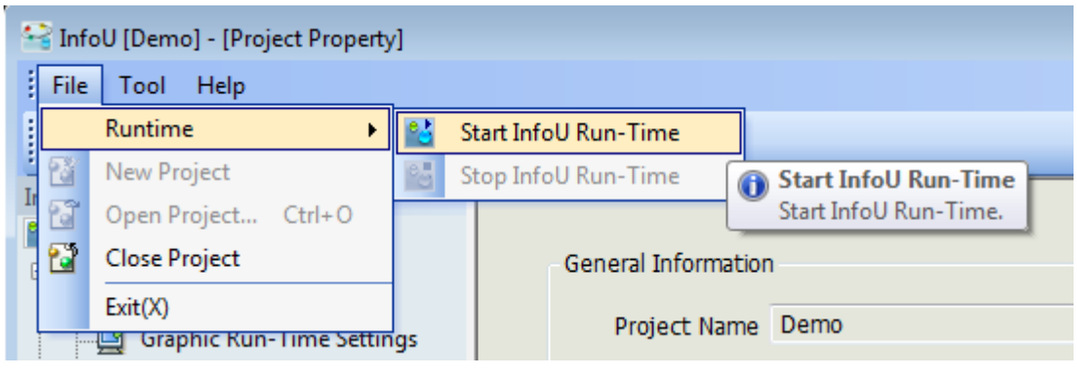
Select [File]-[Runtime]-[Start InfoU Run-Time] from the engineering (InfoUD) program menu or select the tool bar icon to execute InfoUR, InfoU's Run-Time, along with other Runtime component programs.

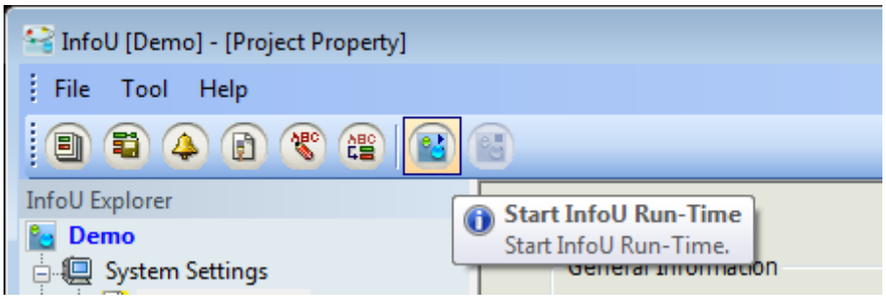
At this time, the default project that starts to run is the project on which the user worked most recently or the demo project included when installing.

You may also click and execute the shortcut icons generated during the program installation.









3.2.2 Stop InfoU

(1) How to stop engineering (InfoUD)

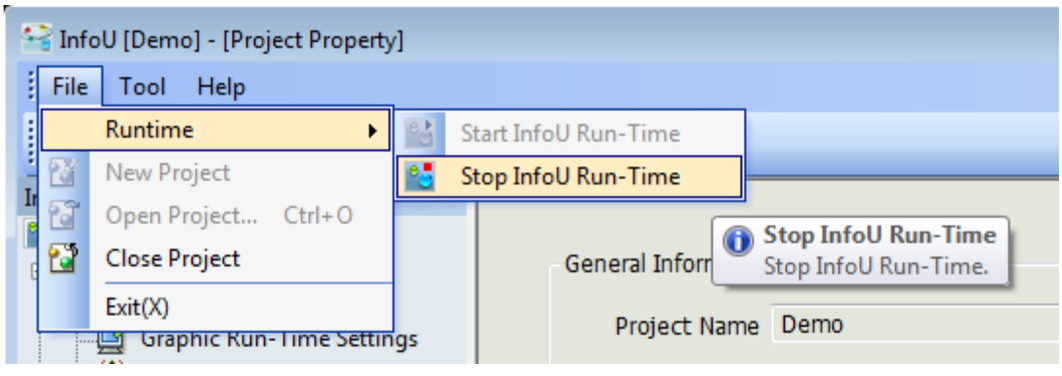
Select [File]-[Exit] from the menu to end InfoU.

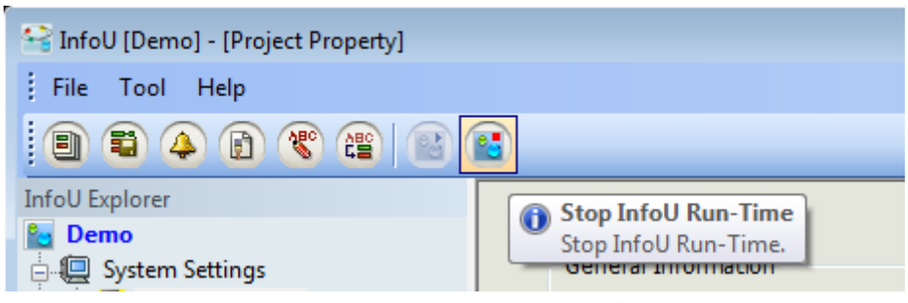
At this time, only the engineering (InfoUD) program will end without having influence on the run time in effect.

(2) How to stop run time (InfoUD)

At this time, only the engineering (InfoUD) program will end without having influence on the run time in effect.

Select [File]-[Runtime]-[Stop InfoU Run-Time] from the engineering (InfoUD) program menu or select the tool bar icon to end Runtime.





When the engineering (InfoUD) program is not in effect, you can also end the project when ending the graphic runtime.

You may select project end menu from the tray icons to end.

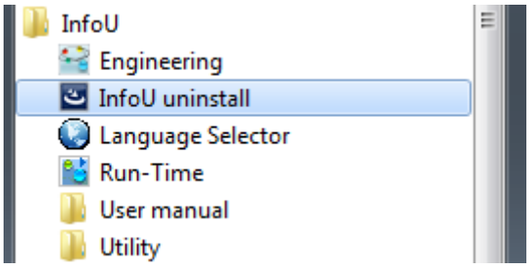
[See Chapter 5: Runtime]

3.3 Removal and Reinstallation of Programs

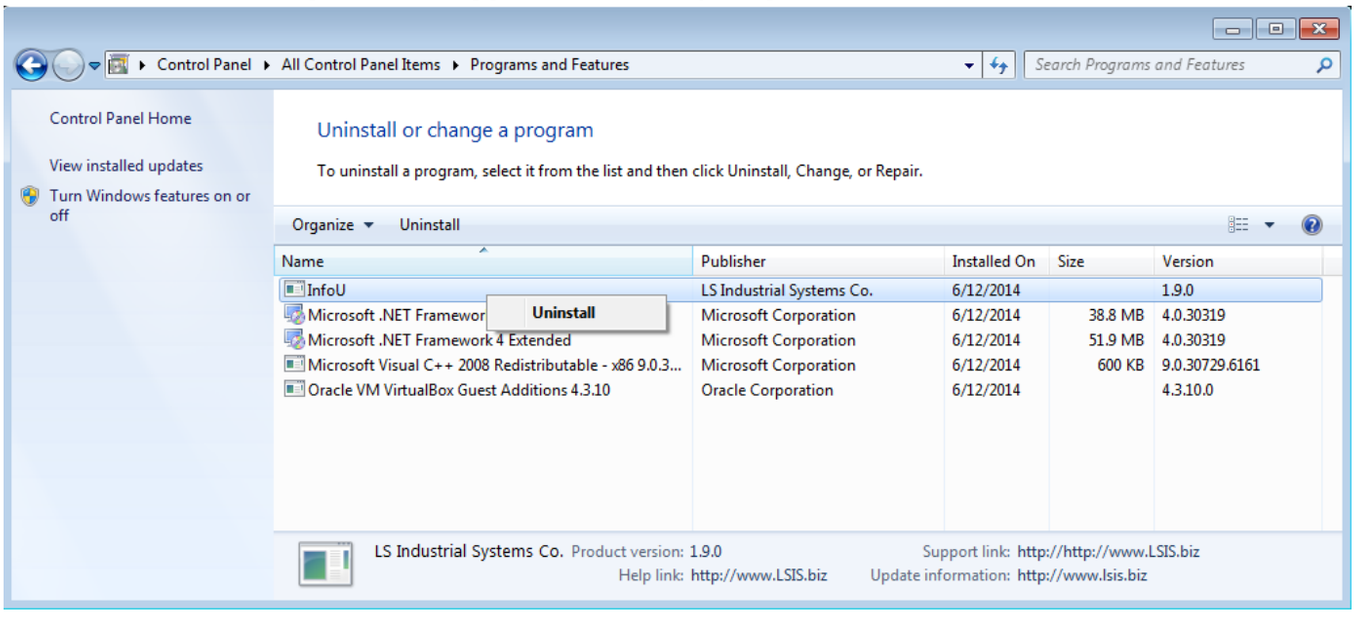
3.3.1 Removal of the InfoU

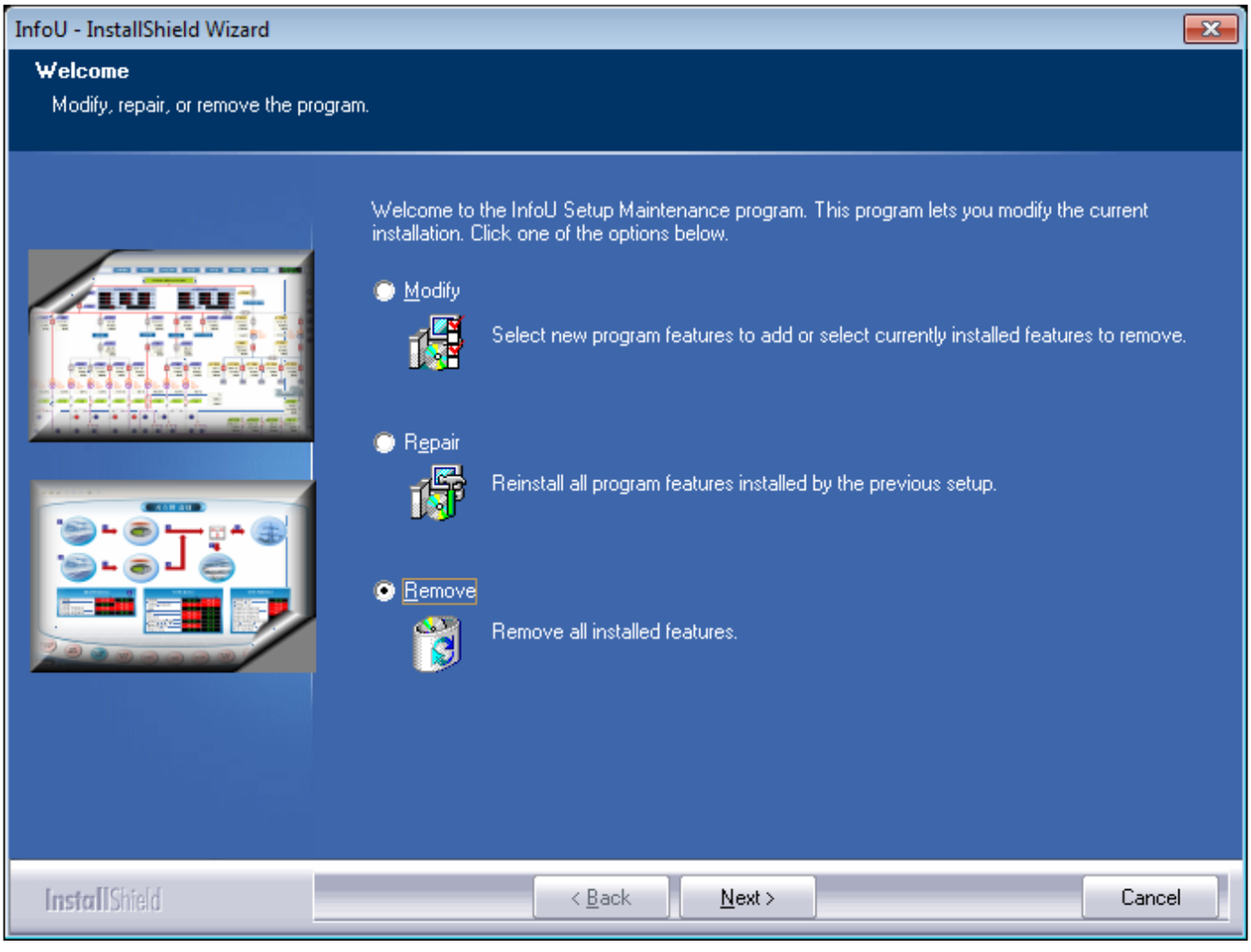
To remove the InfoU program installed in the system, exit the running InfoU runtime program and InfoU engineering program.

Select the [Delete InfoU] menu in the "InfoU" program list.

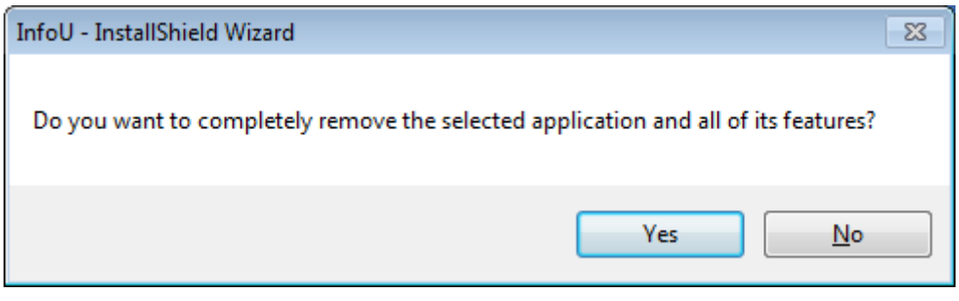


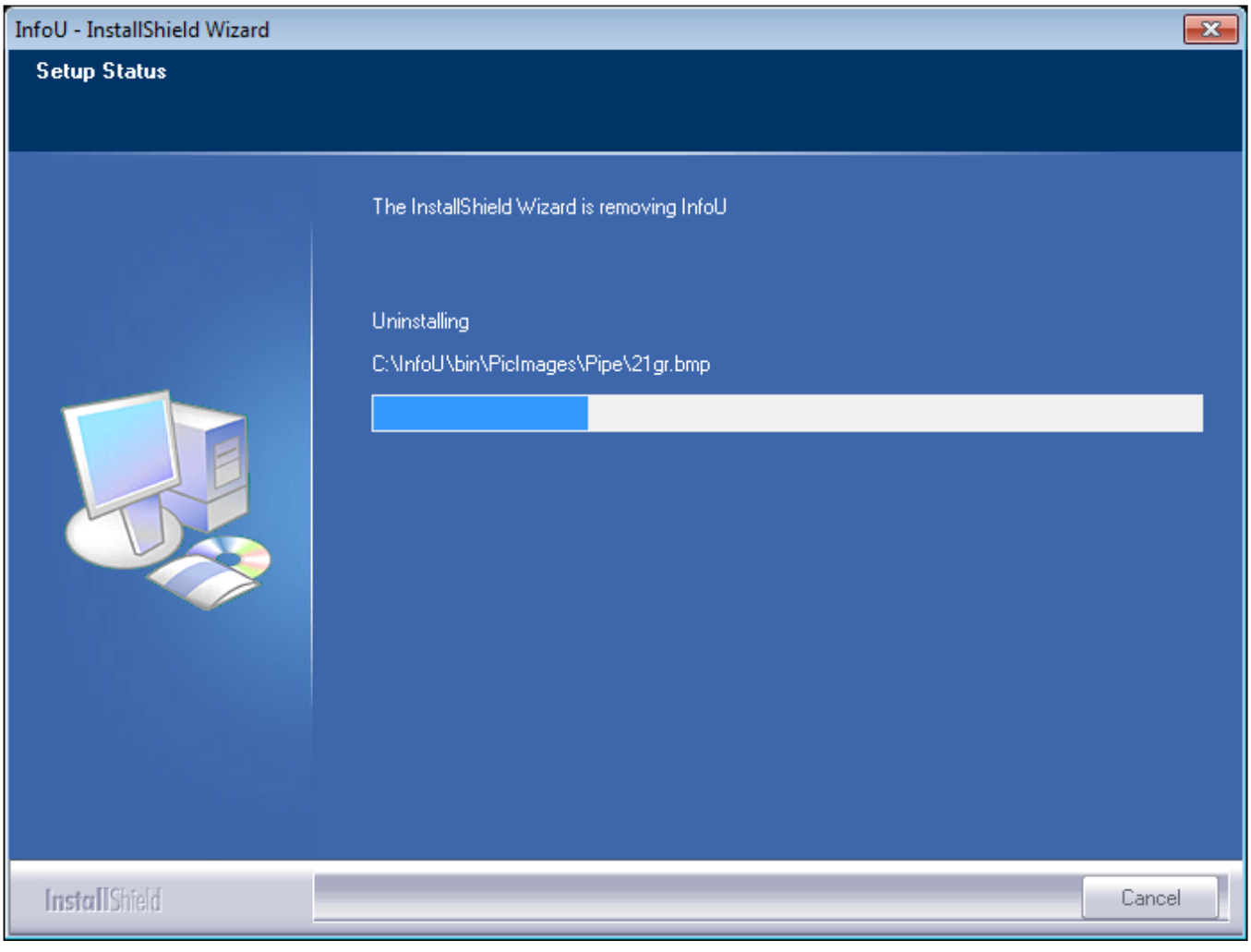
Otherwise, select InfoU in [Control Panel]-[Add/Remove Programs] and click the ‘Remove’ button.



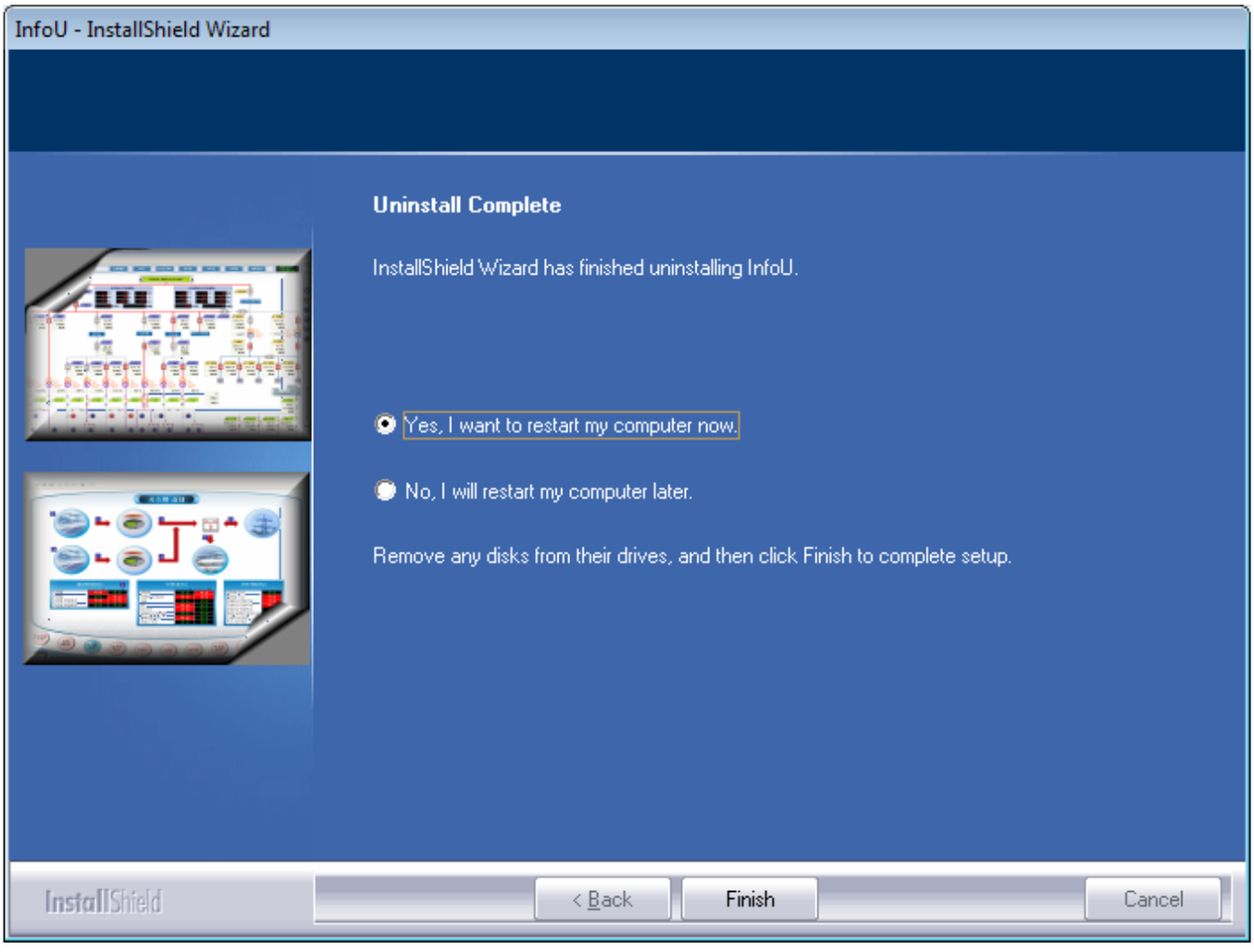


In the above screen, after selecting [Remove], click the [Next] button.





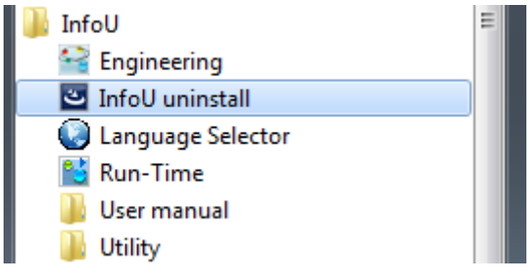
Program removal is in process.

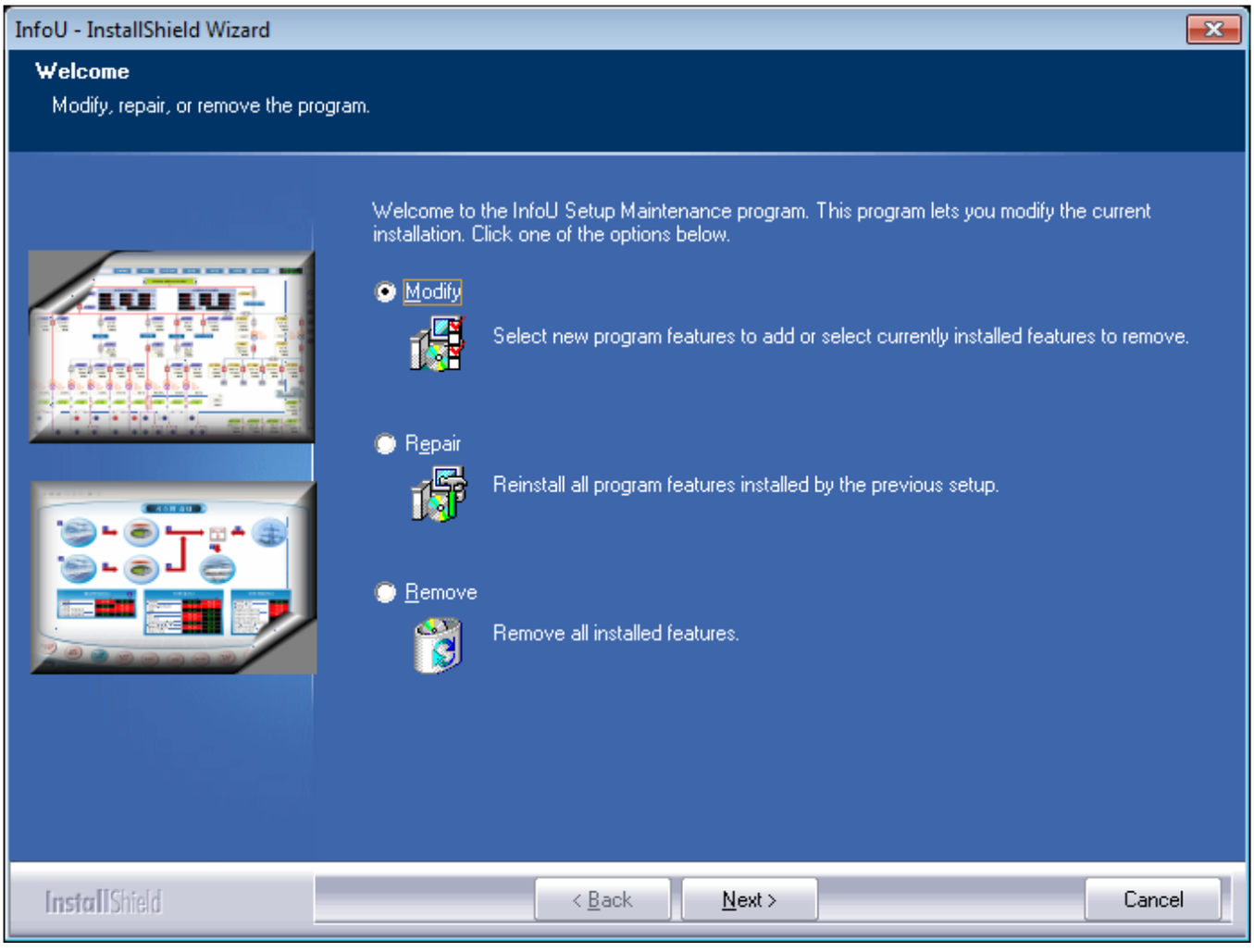


When you restart the computer, program removal will be completed.

3.3.2 Reinstallation

To modify or repair, remove the installed program, as shown in the fig., select [Delete InfoU].





- Modify: Change the installed package.
- Restore: Reinstall the previously installed package.
- Remove: Remove the installed InfoU.

Chapter 4 Engineering

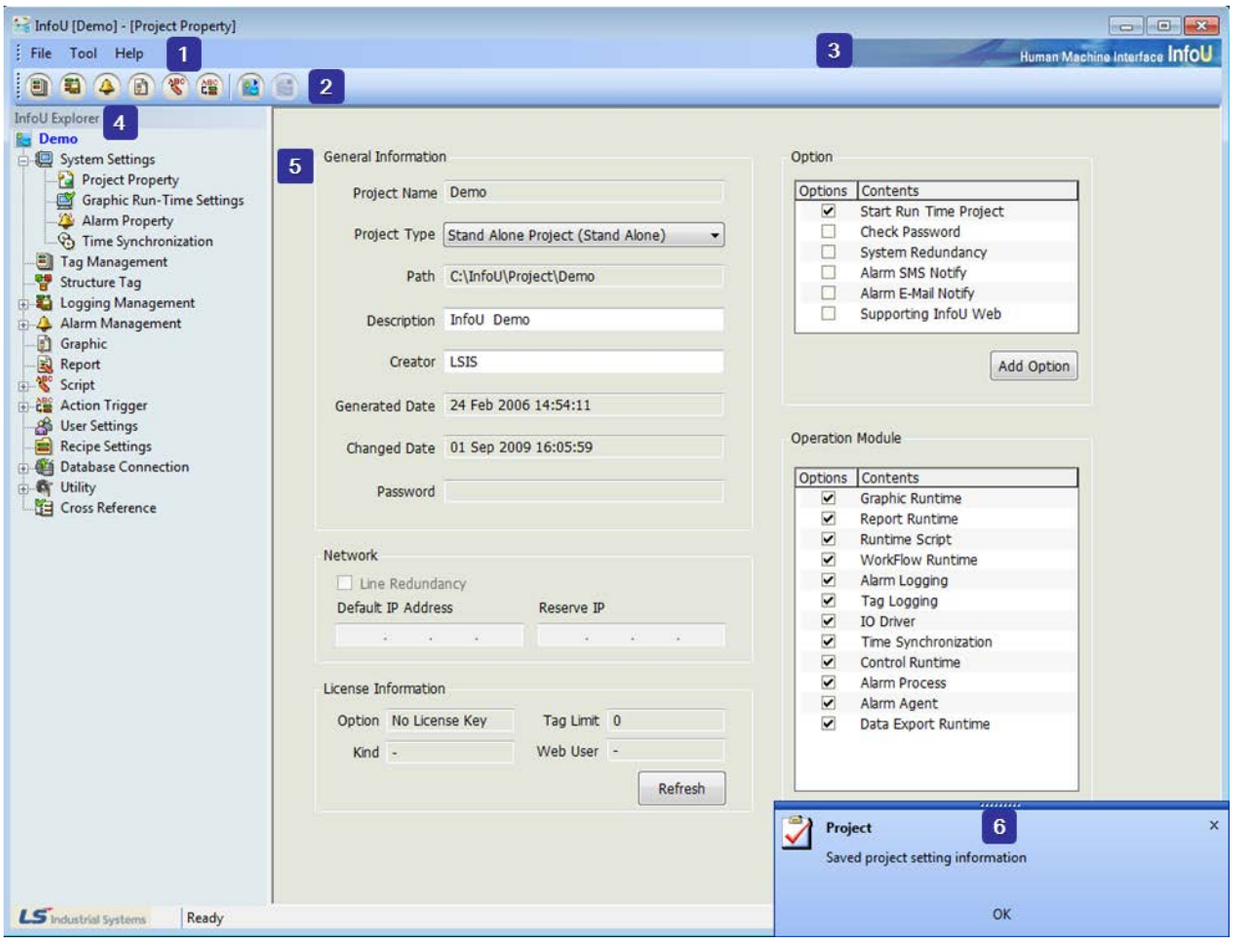
As a tool used for configuring a monitoring and control system and developing an application, InfoU Engineering (InfoUD) allows the user to define system environments needed for executing Run-Time and set up device configuration, tag, logging and alarm. Engineering settings are managed on a project basis.

The screen below is the one that appears when the program starts executing and it is made up of menu, tool bar, tool (application) explorer and splash screen. If the program is executed, the default project will be loaded and the project information screen will appear. The project on which the user worked most recently or the project that is currently running is designated as the default project.
If Close Project is executed or Open Project fails, the project list screen will appear.

Since this chapter covers only the default screen, menu and tool bar, please refer to [Help] if you need more information on each application function.

4.1 Screen Composition

The default screen of engineering is as below.



4.1.1 Main Screen Composition

The main screen consists of the following;

(1) Menu bar

The user can select a menu from the menu bar made up of functions that can be used for the application program.

(2) Tool bar

The user can execute a frequently used application program with one click on the tool bar.

(3) Identity Zone

A zone to display the identity of the InfoU program.

(4) InfoU Explorer

Icons displayed in a tree type to select an application program.

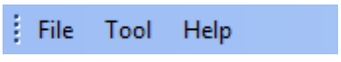
(5) Main View

An area in which the application program is executed.

(6) Pop-up Window

Notice is displayed on the system's status or the user event for application programs.

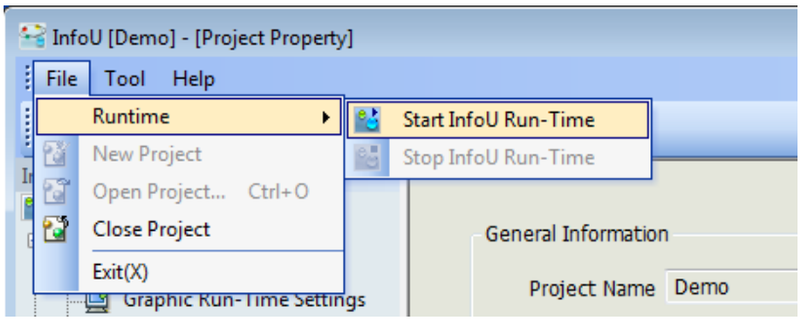
4.2 Menu and InfoU Explorer



Function menus to be used in InfoUD Engineering are provided.

4.2.1 File

Basic menus such as Start InfoU Run-Time/Stop InfoU Run-Time, Open Project/Close Project, Print and other menu are provided.



(1) Runtime

The user can start or stop InfoU Run-Time.

(2) New Project

The user can create a new InfoU Project.

1) Project Folder and File

Once a new project is create in InfoU, the following folders and files will be managed.

The following folders are created having the same names with projects.

Folder Name	Description
Archive-AlarmLog	Path to save alarm event logging files
Archive-TagLog	Path to save data logging files
Config	Path for configuration files needed for running
CrossRef	Path for files related to cross reference
GraLS	Path for graphic screen files
IO Driver	Path for driver configuration files
Log	Path to save application logs
Pacakage	Path for server package files
Report	Path for configuration files related to report
Report-Form	Path for report form files
Report-Data	Path to save report creation files
Script	Path for script files
TagDB	Path to save all of the engineered setup information database
Wav	Path for sound file to generate salarm sound

2) File type related to project

Item	Extension	Description
Project	.lsp	Project information files
Alarm Logging	.adb .rec	Alarm event logging database files Real-time alarm saving files
Tag Logging	.db	Logging database
Config	.ini	Configuration files needed for running
Database	.dbf .cdx	Engineered setup database files
Graphic	.ivd	Graphic file
Serverdata	.sdp	Server data file where the server configuration information is saved
Report	.stg .crf	Report settings information file
Script	.sba	Script file
Sound	.wav	Wave sound file

(3) Open Project

The user can open an InfoU project.

(4) Close Project

The user can close the current project.

(5) Exit

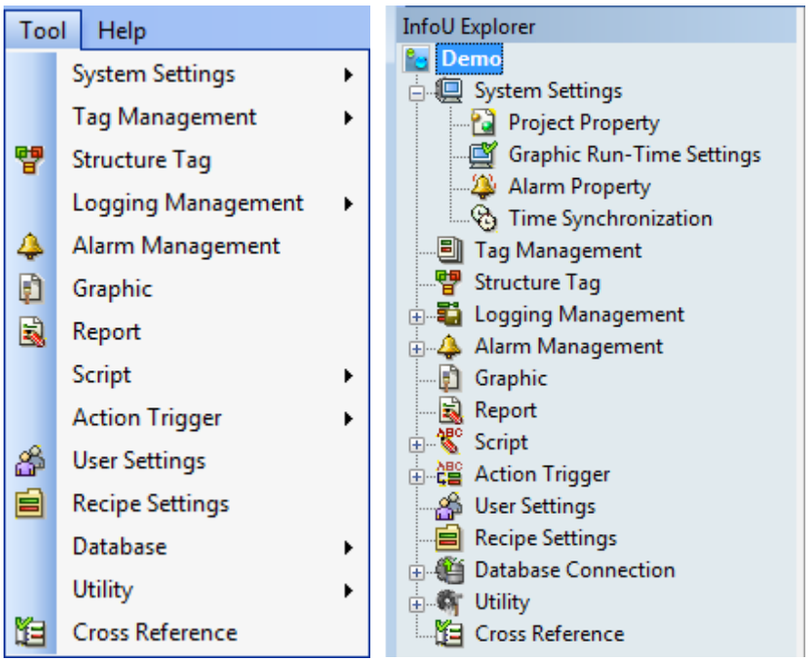
The screen displayed in the current menu view will end and a new empty screen will be displayed.

Notice

If any project is running in real-time, other project can not be selected even after the project is closed in the engineering.

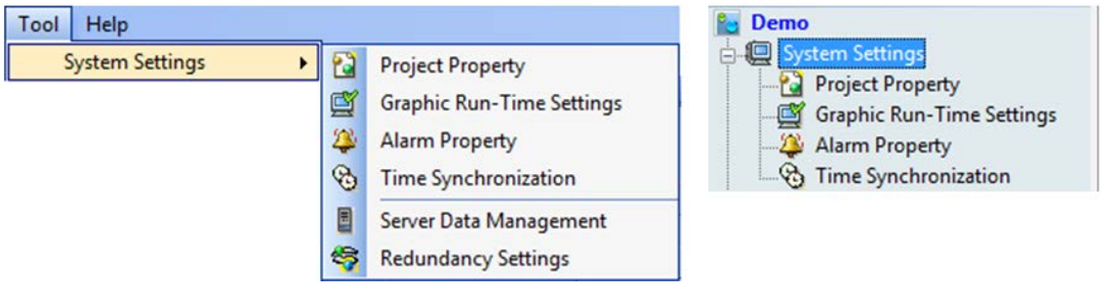
4.2.2 Tool

You can execute various functions that need for your engineering works. Tool menu and InfoU Explorer perform the same functions.



(1) System Settings

You can set up system settings needed for Runtime.



1) Project Property

The user can modify project information and options.

2) Graphic Run-Time Settings

The user can set up the splash screen and style of the graphic runtime screen.

3) Alarm Property

The user can set up overall settings related to activating an alarm.

4) Time Synchronization

The user can set up time synchronization property between system and device.

5) Server Data Management

Set the server data management.

6) Redundancy Settings
Select the tasks related to redundancy.

(2) Tag Management
You can organize devices to be used for the projects, input, edit or delete tags.

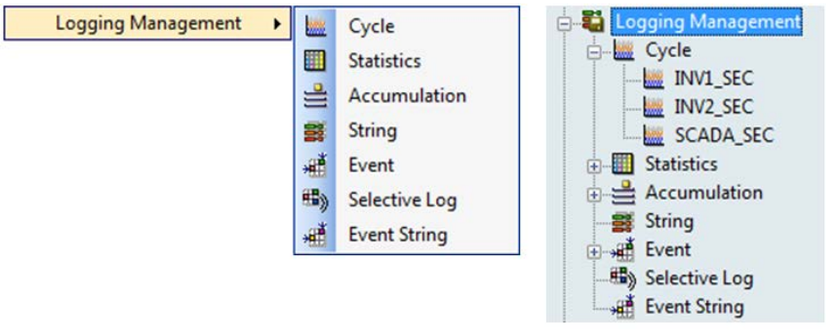


(3) Structure Tag
A Structure Tag template and item list is displayed.



1) Structure Tag
The user can input, edit or delete structure tags.

(4) Logging Management
You can set up and manage logging groups of the tag data needed for the user's history management.
Logging groups can be inquired from Trend, Report, List Trend and Script.



1) Cycle
The user can set up regular cycle logging groups.

2) Statistics
The user can set up statistics logging groups.

3) Accumulation
The user can set up accumulation logging groups.

4) String
The user can set up string logging groups.

5) Event
Set the event logging group.

6) Selective Log
Set the logging group for selective storage.

7) Event String
Set the event string logging group.

(5) Alarm Management

You can manage alarm event information by setting up alarm groups.



1) Alarm Management

The user can input, edit or delete alarm groups.

(6) Graphic

A list of graphic files currently used for the project is displayed.



1) Graphic

The user can click a file from the InfoU Graphic file list in the project “GraLS” folder and edit it in the graphic editor.

(7) Report

A report list is displayed.

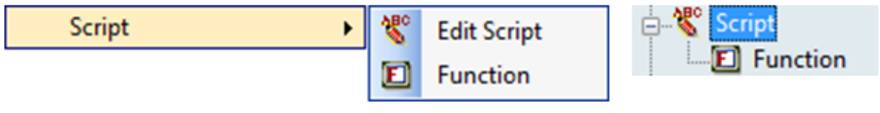


1) Graphic

The user can click a file from the InfoU Graphic file list in the project “GraLS” folder and edit it in the graphic editor.

(8) Script

You can edit a script and inquire an internal function list.



1) Edit Script

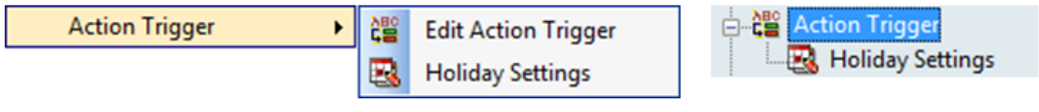
The user can edit a script document

2) Function List

A list of the internal functions presented in InfoU is displayed.

(9) Action Trigger

With this function, you can execute Edit Action Trigger and Holiday Settings for executing script, recipe and database.



1) Edit Action Trigger
The user can add, modify or delete action trigger.

2) Holiday Settings
The user can add, modify or delete holiday settings.

(10) User Settings

The user can set up user-related information and use these settings when receiving the authorization to control them in the alarm action and graphic



1) User Settings
The user can set up his/her personal information such as his/her authorization to control or contact information.

(11) Recipe Settings

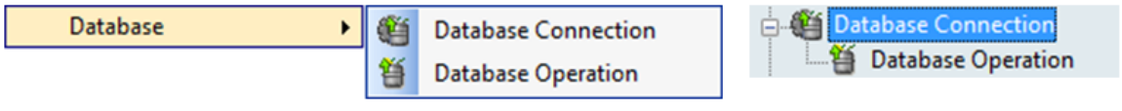
The user can set up control over designated patterns by inputting items, units and data of a recipe.



1) Recipe Settings
The user can add, modify or delete a recipe from the displayed recipe list.

(12) Database

Sharing data with other systems is possible by setting up Database Connection and Database Operation.

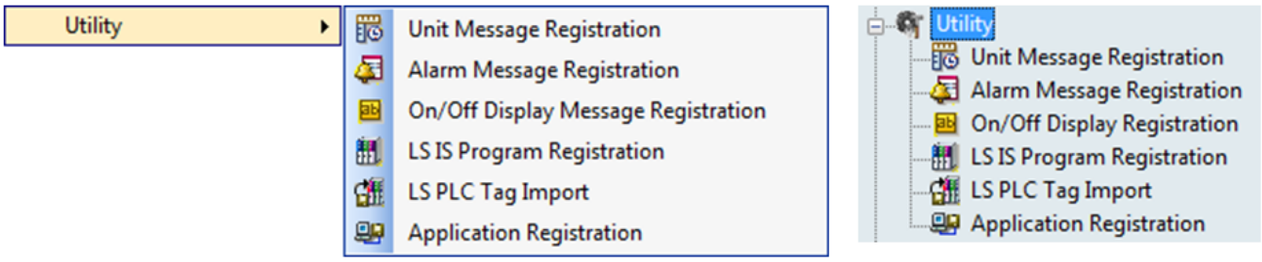


1) Database Connection
The user can manage settings to connect with external database

2) Database Operation
The user can define the data input and output relationship with external database

(13) Utility

It supports convenient engineering works by registering and setting up various components.



1) Unit Message Registration
Various industrial units registered in this menu are very helpful to select a unit easily during the tag edit.

2) Alarm Message Registration
If the user registers alarm messages that will be displayed in case of any alarm event and select those during the tag edit, he/she can inquire those messages when alarm actually takes place.

3) On/Off Display Registration
The user can register display methods according to the status of digital value to use them during the tag edit.

4) LS IS Program Registration
The user can register LS Industrial System's PLC Programs and execute them in the engineering screen and also input a tag automatically

5) LS PLC Tag Import
You can import the LS PLC tag for use.

6) Application Registration
The user can register external programs and execute them in the engineering screen.

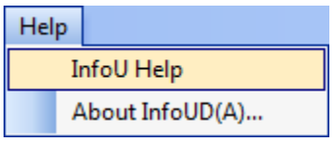
(14) Cross Reference
The Cross reference relationship in the project between tag and object is displayed to allow the user to identify any engineering error.



1) Cross Reference
The cross reference screen is displayed.

4.2.3 Help

Displays InfoU Engineering information



(1) InfoU Help
The Help screen is displayed

(2) InfoU Information
Information on the InfoU version is displayed.

4.2.4 Tool Bar

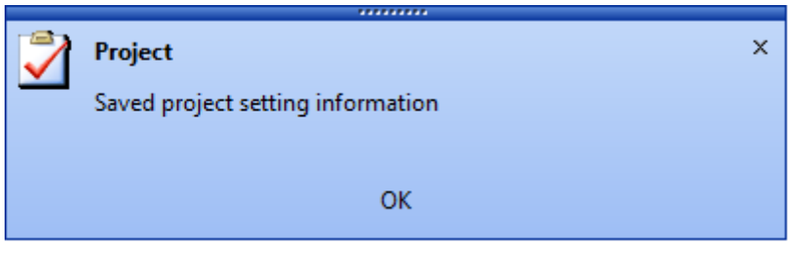
It displays the toolbar menu for each image.



Tool Bar	Name
	Tag Management
	Logging Management
	Alarm Management
	Graphic
	Script
	Action Trigger
	Start InfoU Run-Time
	Stop InfoU Run-Time

4.2.5 Pop-up Window

It provides the function that gives a user the event information.



(1) Notice to inform the system status and user event for application programs is displayed.

(2) Notice is displayed in the messenger-style window.

(3) This window disappears few minutes later after it pops up.

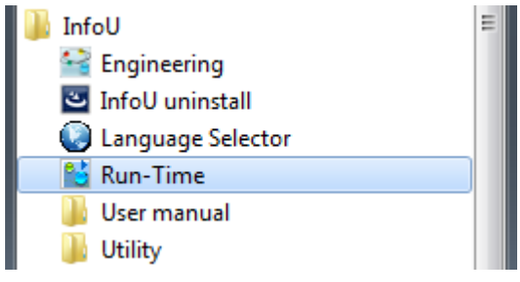
Chapter 5 Run Time

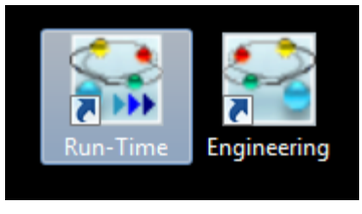
With this function, the user can execute the project registered for engineering settings.
After establishing a communication link between system and device according to the setting information on the channel and station, the user can monitor the real-time site status with the attained data in the plant screen. Also, the user can perform various functions such as history logging and report printing by generating alarms of the major monitoring points.

5.1 Start InfoU Run-Time

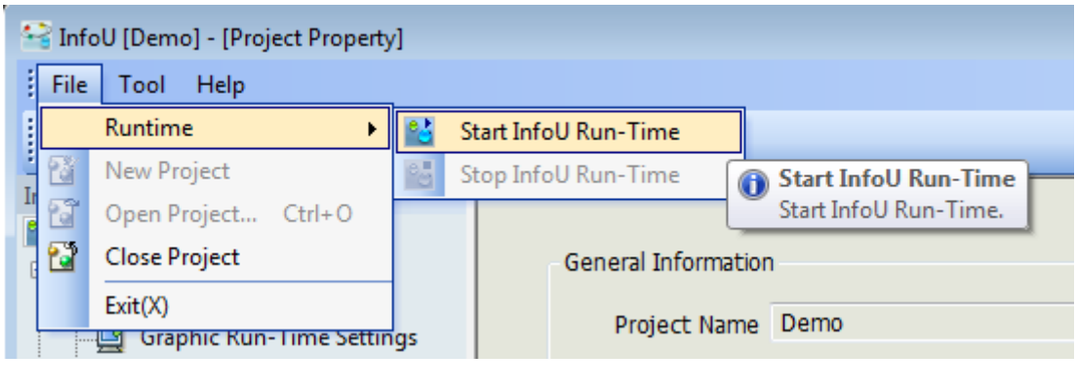
To start InfoU Run Time, click Start in the window screen to find a program directory or click the Run Time icon in the desk top..

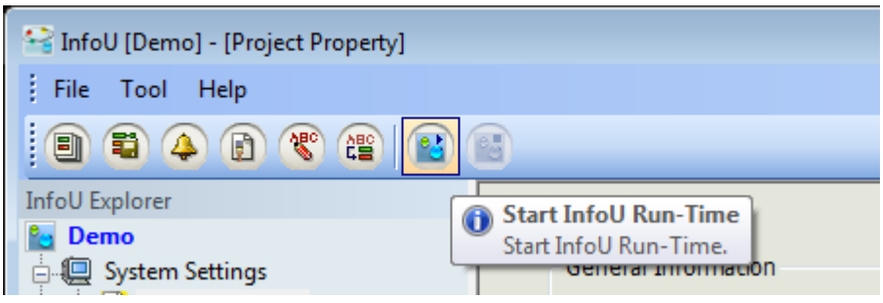
(1) Start with the Shortcut icon.





(2) Start in the InfoU engineering menu.

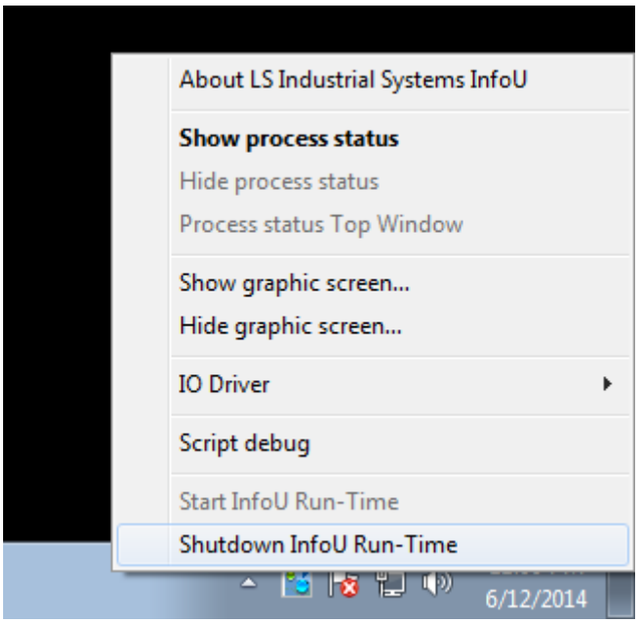



InfoU [Demo] - [Project Property]
File
Tool
Help
InfoU Explorer
Demo
System Settings
Start InfoU Run-Time
Start InfoU Run-Time.

5.2 Shutdown InfoU Run-Time

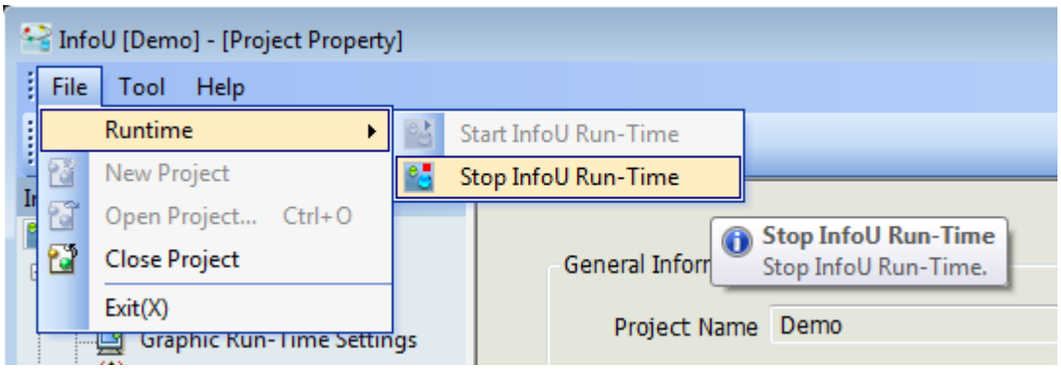
All of the functions operated during Runtime will stop when this function is selected. This function is executed as follow.

(1) Shutdown InfoU Run-Time at Tray icon

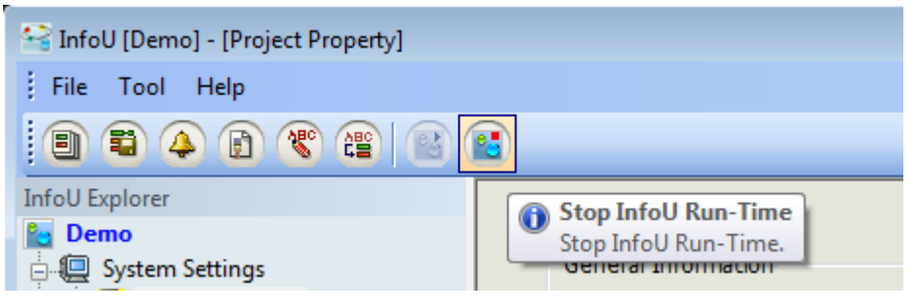


Click [InfoU Tray Icon] → Mouse right button click → [Shutdown InfoU Run-time]

(2) Stop InfoU Run-Time at engineering menu



Click [File] → [Runtime] → [Stop InfoU Run-Time]

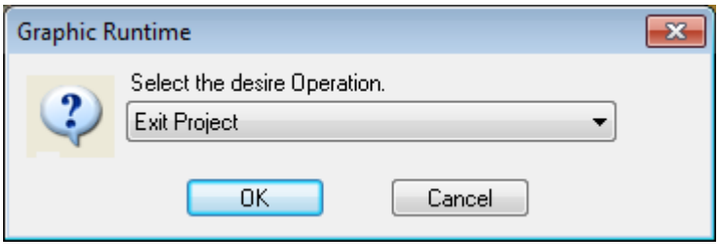


[InfoU Toolbar] → [Stop InfoU Run-Time]

(3) Exit the InfoU Run-Time



Click the menu [File] → [Exit] or click the Windows system menu [X] button.



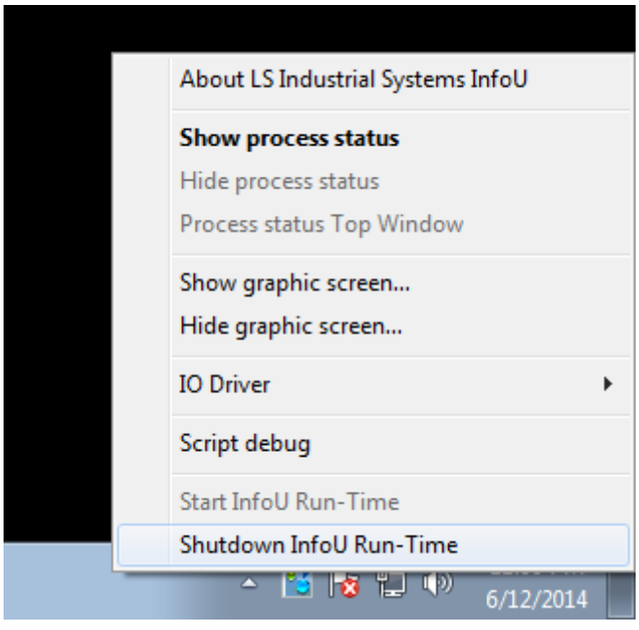
If you click the 'OK' button, InfoU runtime will end.



Once InfoU Run-time ends, the following splash screen appears to show the process status.

5.3 Runtime Menu

5.3.1 Menu Item



Tray icons of the system are displayed as seen in the figure above and the user may use the following menus by clicking one of them.

- Shutdown InfoU Run-Time: The project being operated in real time ends.
- Show process status: The monitoring screen for process status is displayed.
- Show graphic viewer: The graphic monitoring screen is displayed as an upper-level screen.
- Hide process status: The graphic monitoring screen is hidden. Even though it is not displayed, it is being executed in the background.
- I/O Driver --> Device Status: The status of a device connected and set up is displayed.
- I/O Driver -->Simulation settings: The simulation setting screen for the set tag is displayed to generate data on the device values.
- Device Data: A screen is displayed to allow the user to inquire device data.

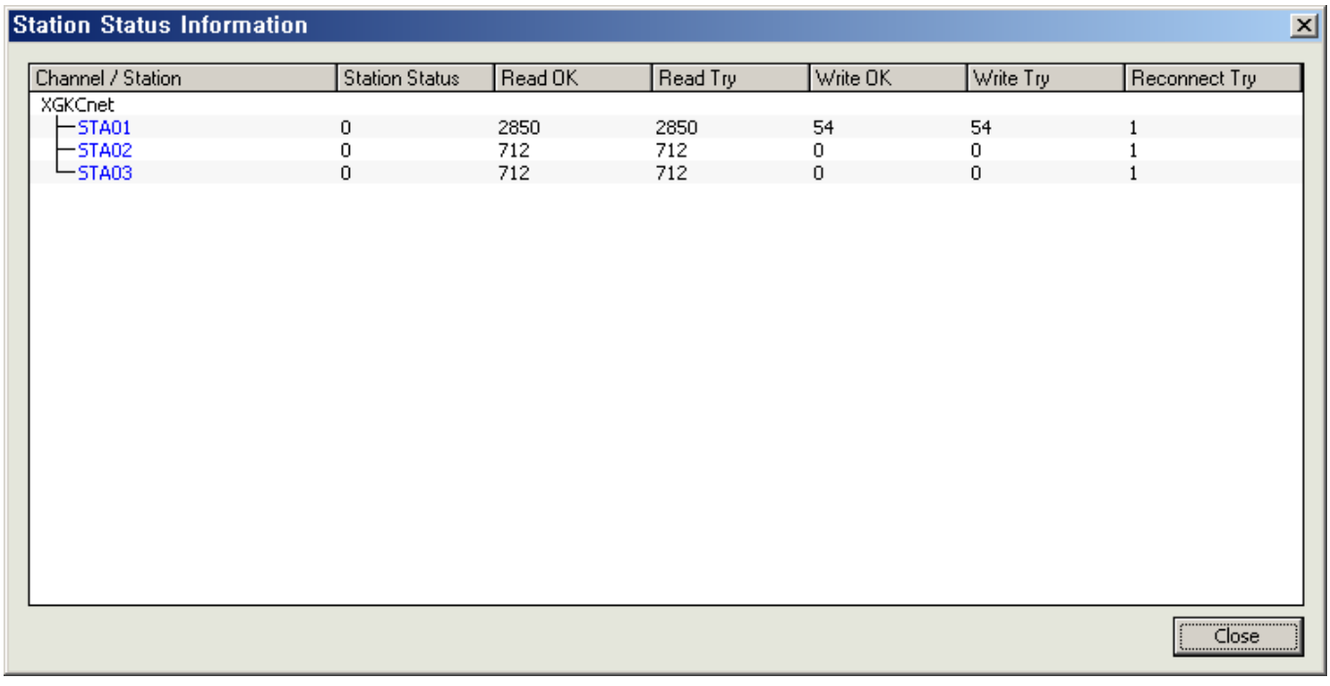
5.4 Device Status

The status screen of the connected device is displayed.

5.4.1 Controls

- Select [IO Driver] --> [Device Status] from Run-Time Tray icon Menu to execute.
- Select [Work] --> [IO Station Status] from Graphic Run-Time Program Menu to execute.
- The user may use ShowNetworkStatus() from Graphic Script Function to execute.

5.4.2 Station Staus Information Interface



- Channel/Station: The communication configuration and status of the device is displayed.
- Station OK: It shows the communication status of the device and 0 means normal, 1 means abnormal.
- Read OK: The number of cases in which device values have been successfully read since the project started is displayed.
- Read Try: The number of cases in which reading device values has been attempted since the project started is displayed.
- Write OK: The number of cases in which device values has been successfully written since t he project started is displayed.
- Write Try: The number of cases in which writing device values has been attempted since the project started is displayed.
- Reconnect Try: The number of cases in which connecting with the device has been attempted is displayed.

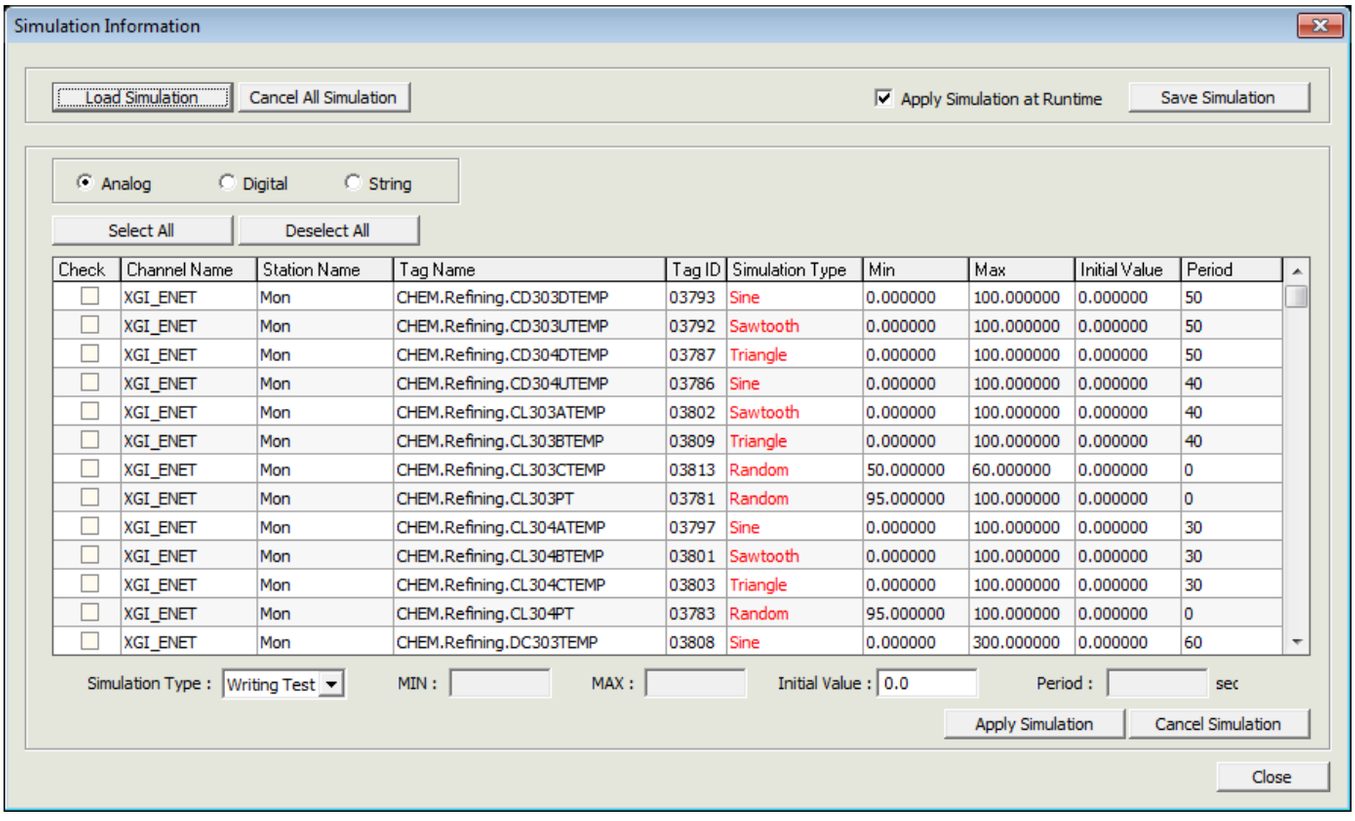
5.5 Simulation settings

As a function to generate virtual device values, the user can simulates virtual device values with this function before connecting a device to identify any error of the engineering work in advance.

5.5.1 Controls

Select [IO Driver] --> [Simulation settings] from Run-Time Tray icon Menu to execute.

5.5.2 Simulation Information Interface



- Load Simulation: The saved simulation files are loaded to set up simulation.
- Cancel All Simulation: All of the simulation settings that have been set up until now are cancelled.
- Apply Simulation at Runtime: When starting Runtime, the user can apply simulation immediately to set up.
- Save Simulation: The current simulation information is saved in a file.
- Select All: All of the tags displayed in the tag list window are selected.
- Deselect All: All of the selected tags in the tag list window are cancelled.
- Tags are displayed depending on the selection either analog or digital. Multiple selections are available.
- Simulation Type: Select a rule to generate simulation data.
  - Analog: The user can select one among Writing Test, Random, Sine, Sawtooth and Triangle,
  - Digital: The user can select one among Writing Test, Random and Change,
  - String: You can choose one; controllable, random
  - When selecting items of XG5000, in case of communication channel/station of LSIS XGT PLC series, you can use the simulation function of XG5000.

- MIN: The user can set up the lowest value of the range of change when generating data.
- MAX: The user can set up the highest value of the range of change when generating data.
- Initial Value: The user can input the initial value when applying the simulation rule.
- Period: Input a period of one cycle of the simulation type that has a certain cycle (Sine, Sawtooth and Triangle).
- Apply Simulation: The user can apply simulation to the selected tags.
- Cancel Simulation: The user can cancel applying simulation to the selected tags.
- Close: Close the current dialog box. Even after the screen ends, simulation data keep operating.

5.6 Device Data

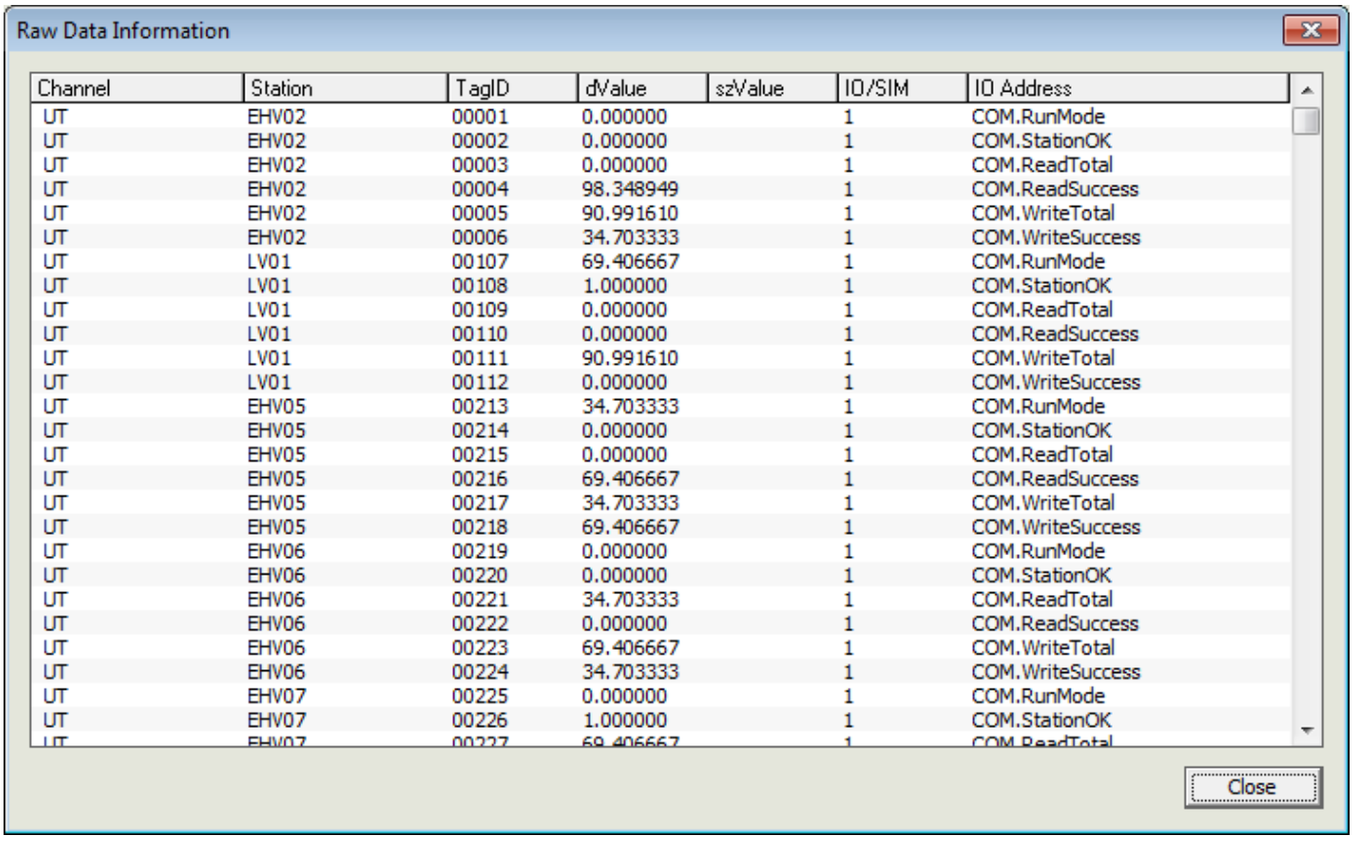
The user may monitor raw data values attained from the communication with the device.

Simulation data are displayed when device values are simulated.

5.6.1 Controls

Select [IO Driver]→[Device data] in the real time operation tray icon menu and run it.

5.6.2 Raw Data Information Interface



Chapter 6 System Settings

In this step, you may create a project and manage basic environment settings for Runtime.

6.1 Project Property

As the first step needed for Runtime, the user can create a project and manage configuration settings and options in this step.

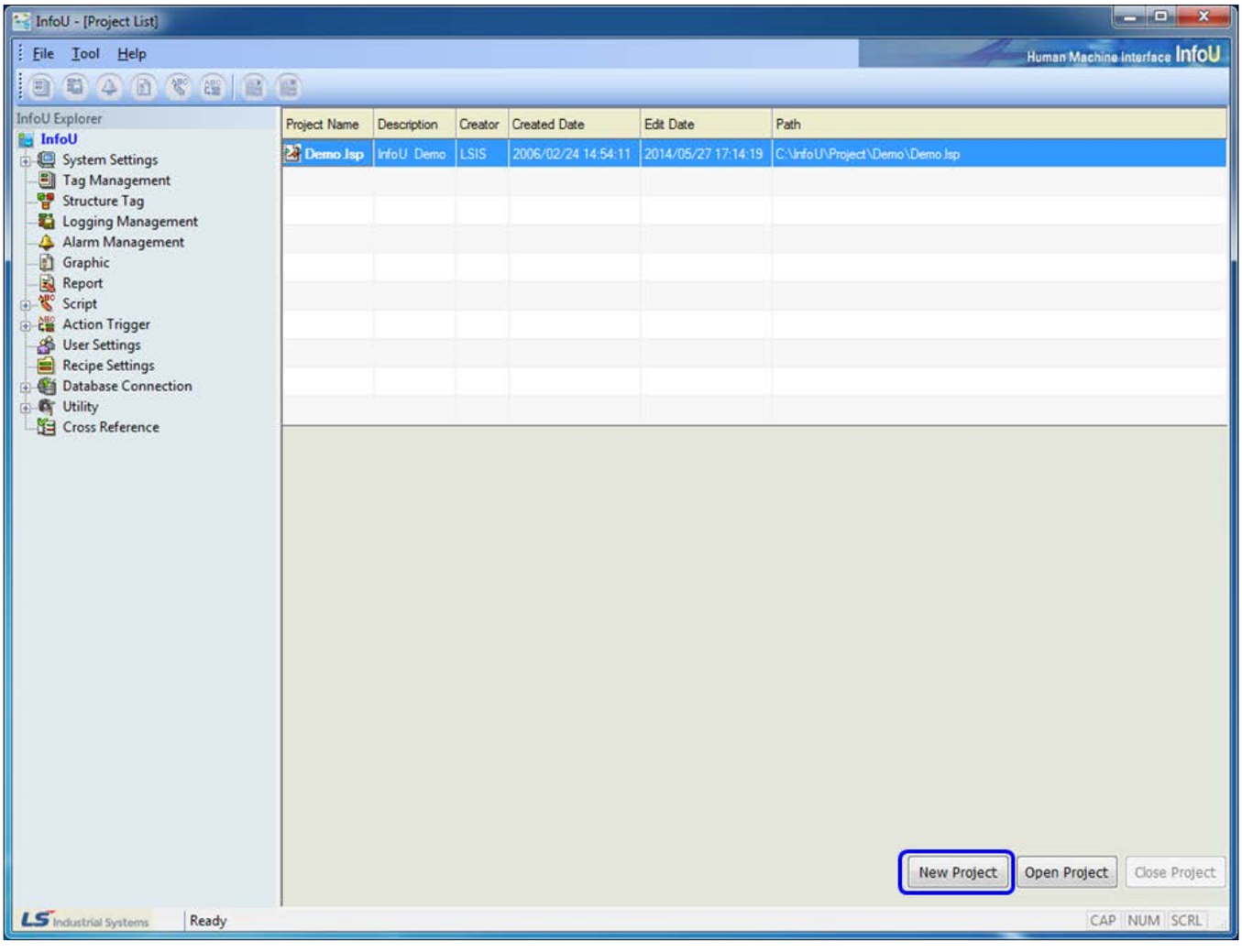
Each project becomes an unit subjected to Engineering and Run-time management and the user can move, copy or use projects to other system, drive or path since they are created and managed as a batch and under each project unit, sub-paths such as tag, logging, alarm, graph, report and script are created and managed.

It can easily manage numerous projects while providing easy copying and transferring contents between projects and efficient engineering works.

6.1.1 Create New Project

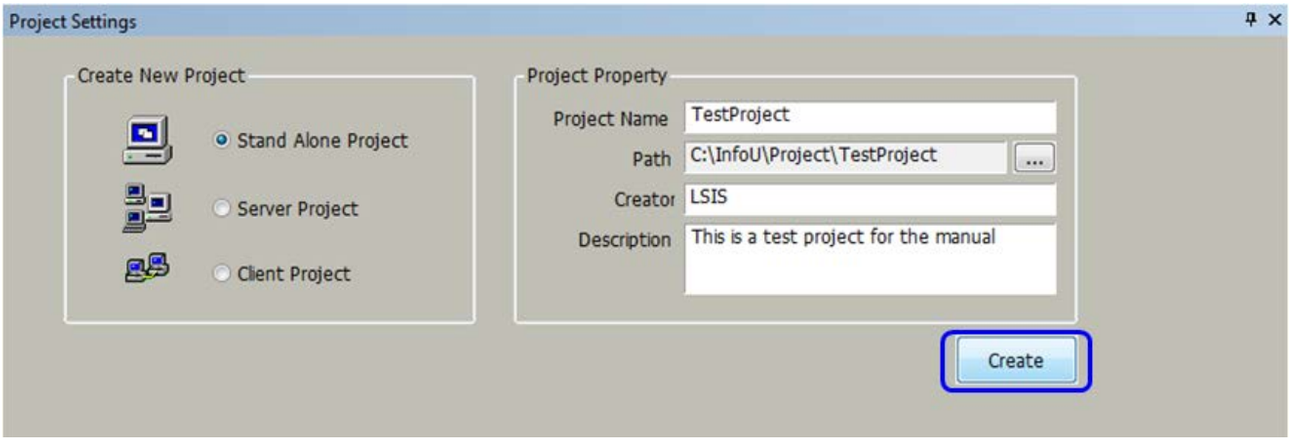
A function New Project can be executed after closing the previous project.

Once the project is successfully closed, the latest project list is displayed. Click [New Project] button to perform the work of creating a new project.



Notice

If any project is currently operated, a new project is not created.



(1) Project Type

1) Stand Alone Project
A stand alone project is created

2) Client -> Server Project
A project that is activated by the server mode of the client to server system is created

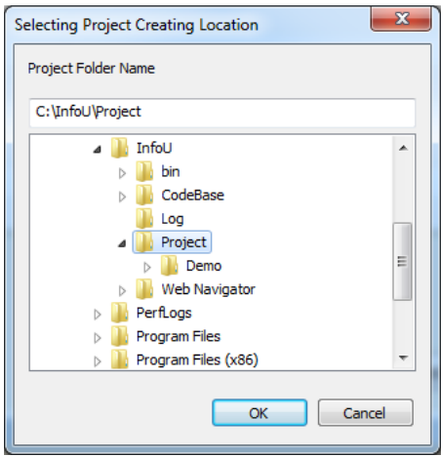
3) Client -> Client Project
A project that is activated by the client mode of the client to server system is created

(2) Project Name
Input a project name

(3) Project Path

1) It refers to a path where a new project is created.If […] button is selected, the following screen will appear to allow the user to select a path

2) A new folder with the same name as the project is created in the subfolder of the folder designated by the user to create a new project



(4) Creator

Input a name or a department of the person who creates the project.

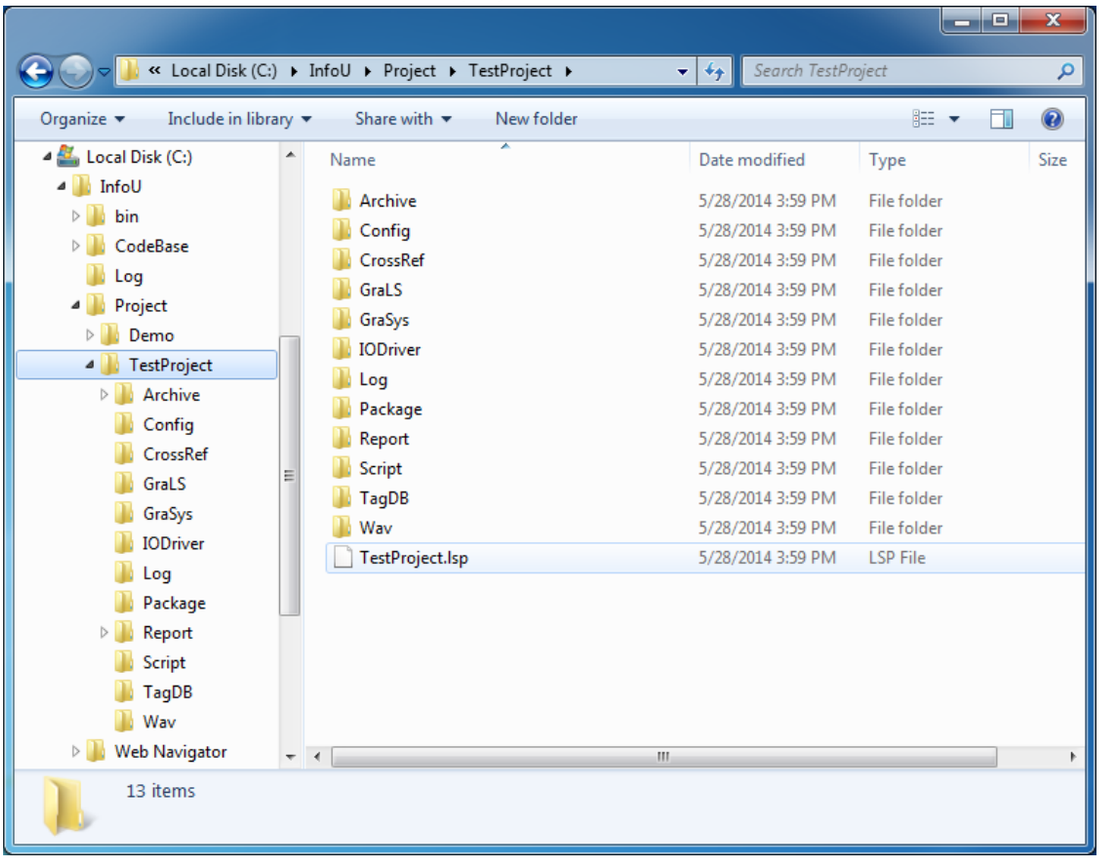
(5) Description

Input detailed information

(6) Create

A new project is created. After the new project is created, the project will open

(7) Once the new project is created, a new project file with a project folder and extension ".lsp" will be created



6.1.2 Project Settings

You may set up general properties of the project and basic configurations of the network. Also, you may set up project options and execution modules suitable for your purpose.

Notice

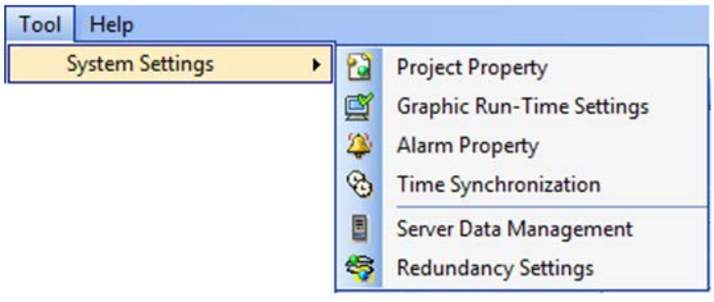
Project options can be set up depending on the authorized scope by the system license. Refer to the following example:

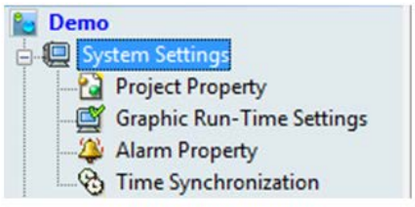
If the license key is of developer or executor version and “Web Function Support” is selected as an option, the demo mode will operated for one hour resulted from the violation of license.

(1) How to display the screen

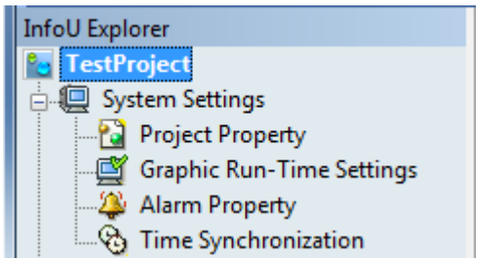
●If the engineering program is executed, the Project Property screen of the basic project is displayed

● Select the ‘Project Property’ in the menu and tool explorer.



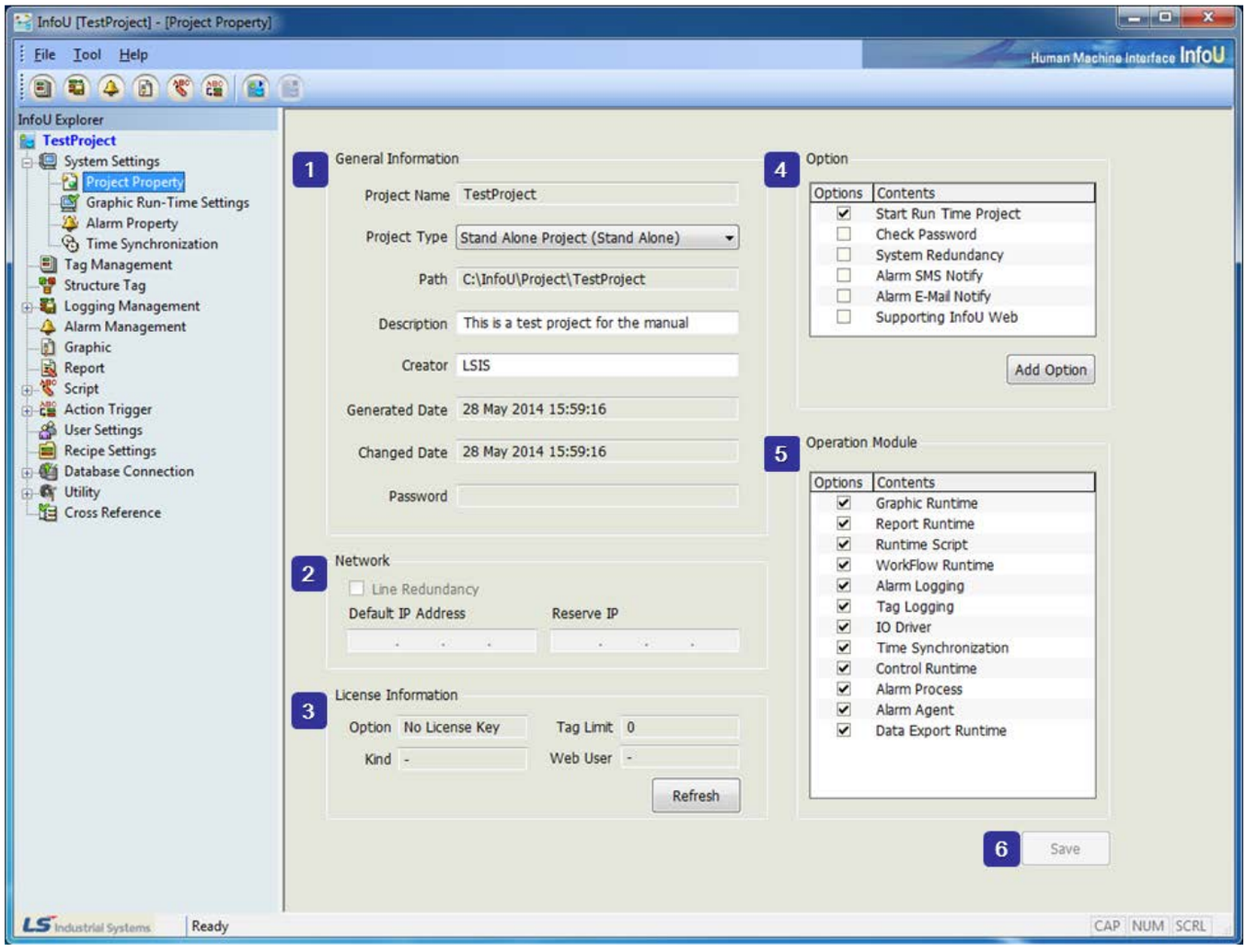


●Select the route item (project name) from InfoU Explorer.



(2) Project Property Interface

The screen configuration of ‘Project Property’ is as below.



1) General Information

a. Project Name

The name the user gives when creating a project is displayed and it is not editable

b. Project Type

The selected project type is editable

c. Path

The location of the project is displayed

d. Description

The user can give additional information on the project name

e. Creator

The name of the first creator is displayed

f. Generated Date

The date when the project is created for the first time is displayed

g. Changed Date

The date when the project is latestly changed is displayed

h. Password

If the option “Check password when the program opens” is selected, the user needs to input thepassword.

2) Network

a. Line Redundancy

If Line Redundancy was set up in the client and server project, this function is selected.

b. Default IP Address

The user needs to write the IP Address he/she will primarily use

c. Reserve IP Address

The user needs to write the IP Address he/she will use in emergency.

d. This setting is deactivated in a single project (Stand Alone).

3) Information of license

a. The information of the license key is displayed.

b. Option

Engineering, runtime-only, web server options are displayed.

c. Tag Limit

Number of available I/O Tag is displayed.

d. Kind

XGT-InfoU

e. Web User

When using a web license, the number of concurrent connected users.

4) Option

a. Start Run Time Project

It is automatically executed in a runtime mode while InfoUR is executed.If it is not set up, the standby mode is executed.When creating a project, the user may select default.

b. Check Password

The user needs to input the password to begin the engineering work when opening the project.

c. System Redundancy

When starting the server, the user needs to set up whether to start it as redundancy.

d. Alarm SMS Notify

The user needs to set up whether to provide the alarm message service by SMS or not when any alarm occurs.

e. Alarm E-Mail Notify

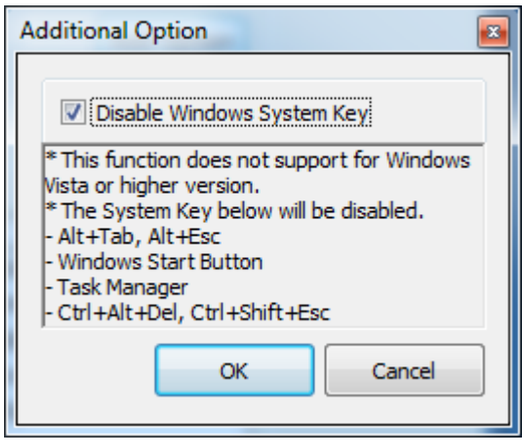
The user needs to set up whether to provide the alarm message service by e-mail or not whenany alarm occurs.

f. Supporting InfoU Web

As an option, the user needs to decide whether to provide the Web service support or not.

g. Additional Option

If you click the [Additional Options] button, the below screen will show up.



- When using this option, the Windows system key will be deactivated.
- Alt-Tab, Alt+Esc, Windows Start button, Task Manager, Ctrl+Alt+Del, Ctrl+Shift+Esc keys are not available.

Notice

This function is available for Windows XP only. It cannot be used for Windows Vista or Windows7.

5) Operation Module

a. The user can limit program functions by selecting functions to execute from the process list

b. This function is intended to restrict unnecessary functions for tests when using a low-capacityhardware

Notice

Select All is a default setting and if the user limits functions voluntarily, unexpected results may be prompted because the program settings may effect on the interrelation between functions

c. Graphic Runtime

If it is not selected, only data service is executed as background without the plant monitoring control.

d. Report Runtime

The function to automatically output or save the report is restricted.

e. Runtime Script

The function to execute the scrip is restricted.

f. WorkFlow Runtime

The function to designate the scrip operation is restricted.

g. Alarm Logging

The function to save alarm event history is restricted.

h. Tag Logging

The function to save history data is restricted.

i. IO Driver

The function to communicate with the device is restricted.

j. Time Synchronization

The function for time synchronization is restricted.

k. Control Manager

All of the functions set up not only in the screen but also in the scrip are restricted.

l. Alarm Process

Limitation on operational functions such as automatic printout of alarm sound, automatic printout, SMS, E-Mail and alarm logging, etc. when an alarm occurs.

m. Alarm Agent

Limitation of the alarm sound

n. Data Export Runtime

The function for data export is restricted.

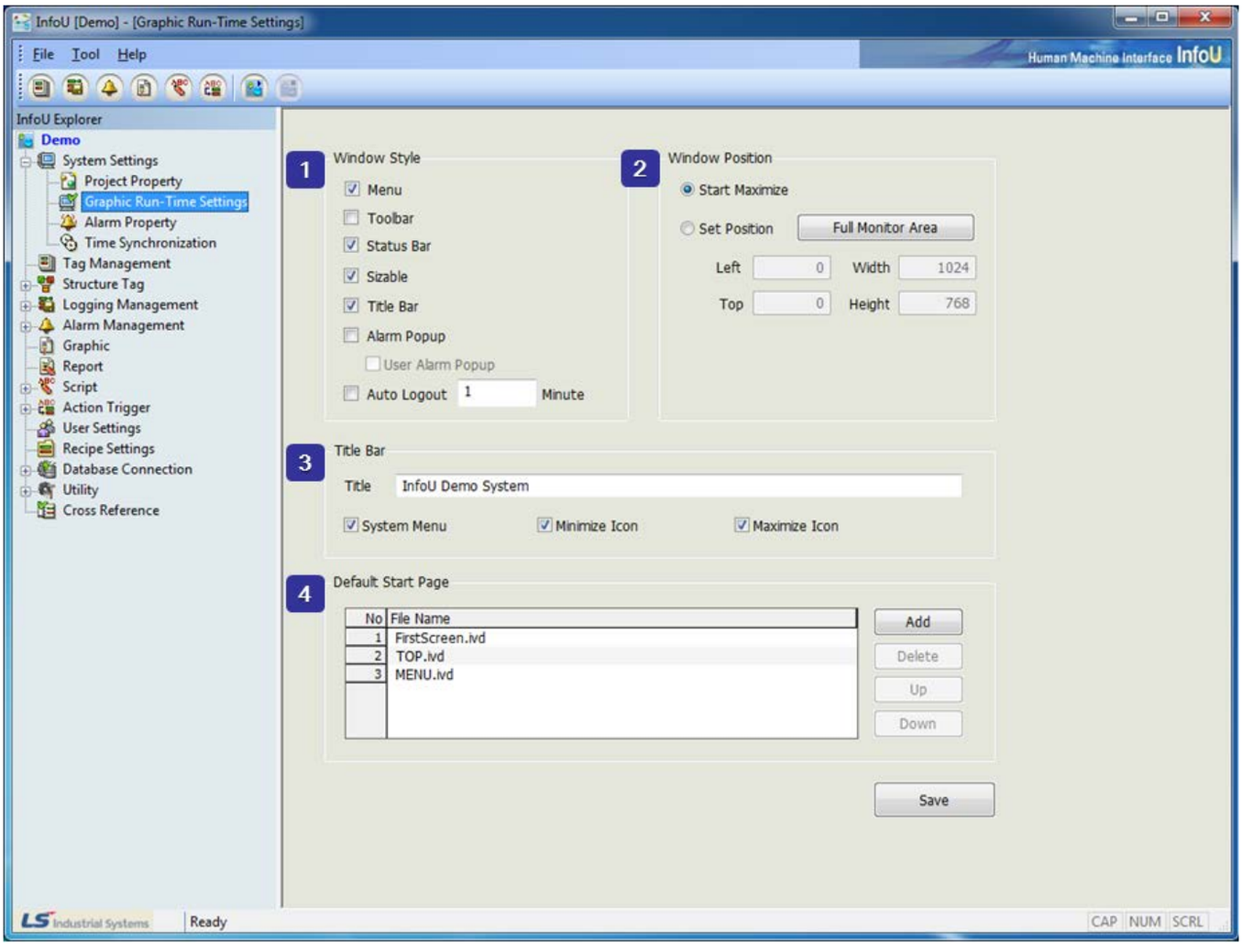
6) Save

a. The selected options are saved.

b. The options selected during Runtime are not applied; therefore, the user needs to restart the program after closing it to apply the selected options.

6.2 Graphic Run-Time Settings

In the graphic execution screen, the user can set up environment for the graphic run-time (InfoUX.exe).
For environment settings, the user can set up menu, view/hide toolbar, initial execution position, name to be displayed in the title bar, view/hide window control buttons and other options and also set up a default page to open as soon as it is executed.



(1) Window Style

1) Menu
Select whether to view the basic menu.

2) Toolbar
Select whether to view the toolbar.

3) Status Bar
Select whether to view the status bar(d) Sizable: Select whether to allow the operator to change thescreen size.

4) Screen size adjustment
The operator can change the screen size.

5) Title Bar
Select whether to display the title window located in the top of the window.

6) Alarm Popup
Select whether to display the latest alarm in the pop-up window when any alarm occurs during RunTime
a. User Alarm Popup
The alarm popup screen of a user is displayed.

7) Auto Logout
In case there are no key inputs or mouse operations until the set time passes after login, it will be automatically logged out.

(2) Window Position

1) Start Maximize
Select whether to execute RunTime in the maximum screen.

2) Set Position
The user can designate the location and size of the screen during Run Time.
Click “Full Monitor Area” button, the area coordinates for the multiple monitors are displayed. The coordinates reference (0, 0) is for the main monitor.

(3) Title Bar

1) Title
Input the title of the screen.

2) System Menu
Select whether to display system menus in icon types in the left top of the screen.

3) Minimize Icon
Select whether to display minimized buttons in the right top of the screen.

4) Maximize Icon
Select whether to display maximized buttons in the right top of the screen

(4) Default Start Page

1) This is for selecting screens to be displayed at first when executing Run Time.

2) Add
All pages contained in the current project will be listed and among them, you can select one for the initial screen.

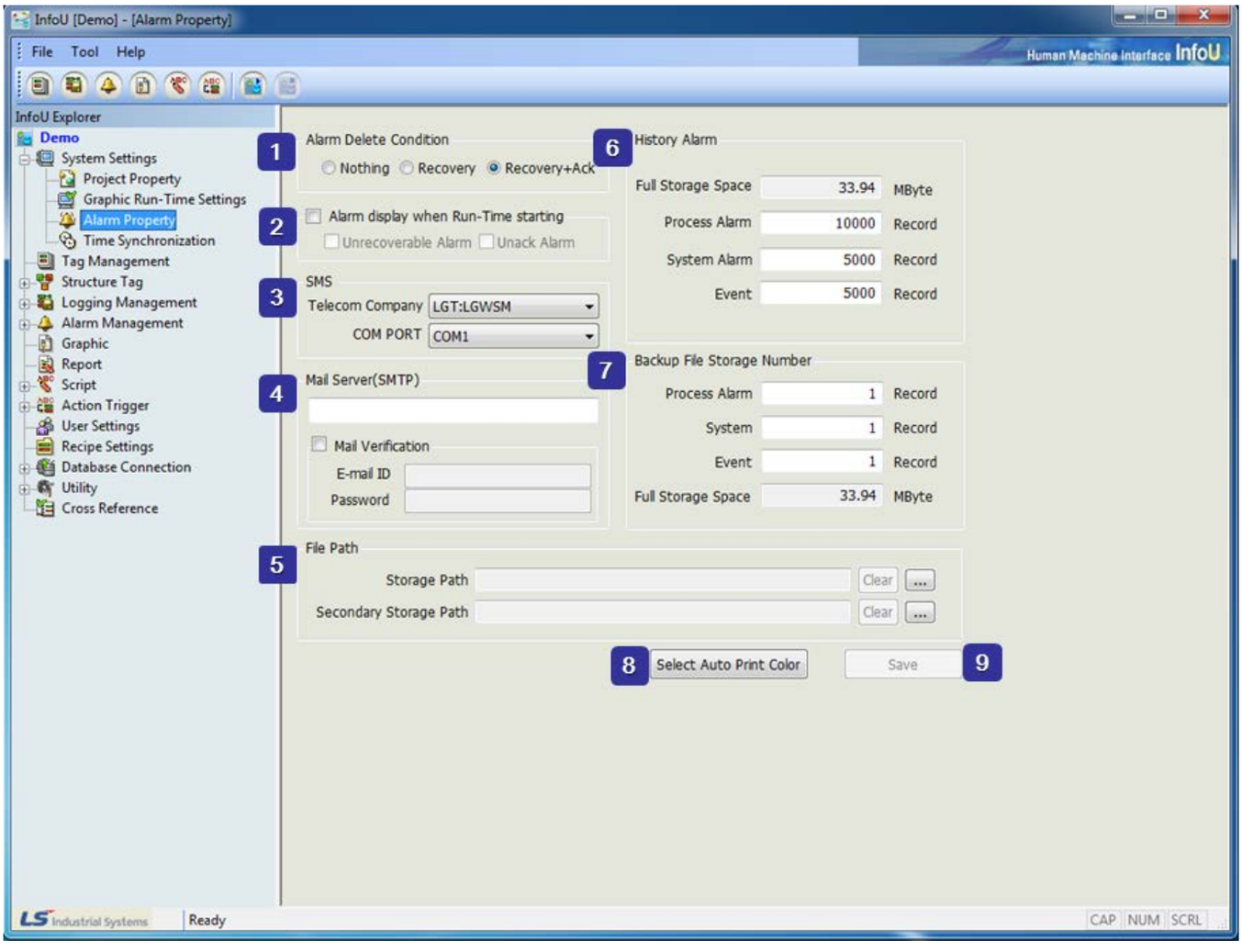
3) Delete
The selected screen file is deleted from the initial screen list.

4) Up/Down
You can specify the order of opening the initial screen by moving the selected file up/down.

6.3 Alarm Property

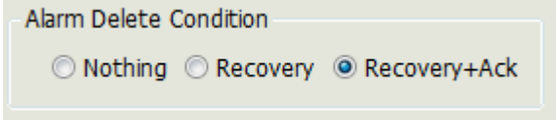
You may set up items needed for handling and operating alarms. You may set up the number of alarm events to save and the criteria to handle real-time alarms (Criteria to delete or designate auto printing and color, etc) and also set up common options (SMS, mail) needed to handle alarms.

Select Alarm Property from InfoU Explorer.



(1) Alarm Delete Condition

Set up a time to delete alarm events from the real-time alarm memory.



1) Recovery

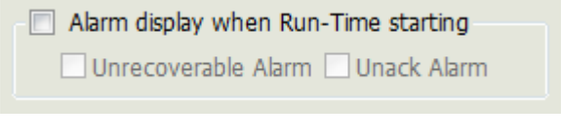
When the alarm is recovered, it is deleted form the real-time alarm memory (Default)

2) Ack.
You can delete the alarm only when it is acknowledged.

3) Recovery and Ack.
The alarm can be deleted only when it is recovered and acknowledged.

(2) Alarm display when Run-Time Starting

Set up whether to cancel the real-time alarm when restarting. If not set up, the real-time alarm is reset when the system restarts and the number of alarm begins from 0.



1) Unrecoverable Alarm
When starting the program, information on the unrecovered alarm is loaded in the memory.

2) Unack Alarm
When starting the program, information on the unacknowledged alarm is loaded in the memory.

(3) SMS

Set up details on the SMS message service to send alarm information when it occurs.



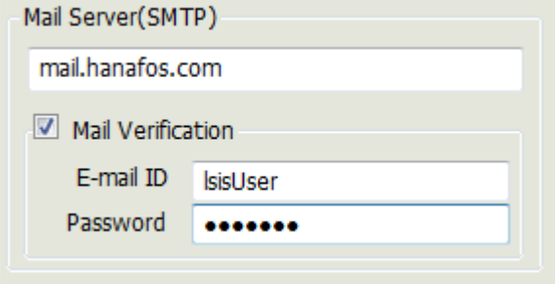
1) Telecom Company
Select a telecom company used for sending SMS

2) SMS COM Port
If the SMS sending terminal is installed, set up COM Port.

(4) Mail Server (SMTP)

A mail server (SMTP) shall be set up to send a mail related to alarm information to the user.

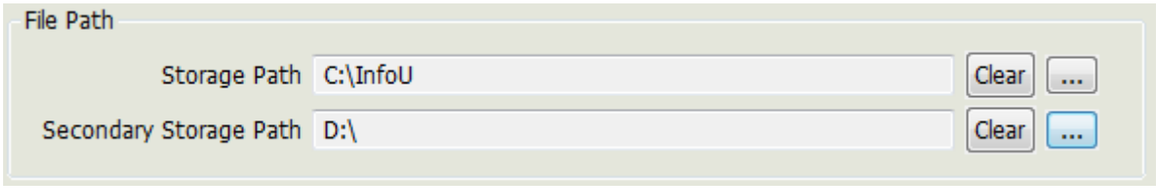
Set up a mail server address.



1) Mail Verification
If an account is needed to connect with the mail server, input the mail ID and password.

(5) Backup File Path

You may a path to save backup files.



1) Storage Path
   A path to save backup files is set up.

2) Secondary Storage Path
   To be prepared for the event that it fails to access to the storage path, set up a secondary storage path to access.

3) Delete button

   It deletes the input path.

4) Path Search
   A button to search folders to input each storage path and when this button is pressed, a folder search window will appear.

(6) History Alarm

The number of alarms for each alarm type is set up to save them in the history DB.

1) Total storage space
   It calculates the storage space depending on the number of alarms set by a user and represents it to the user.

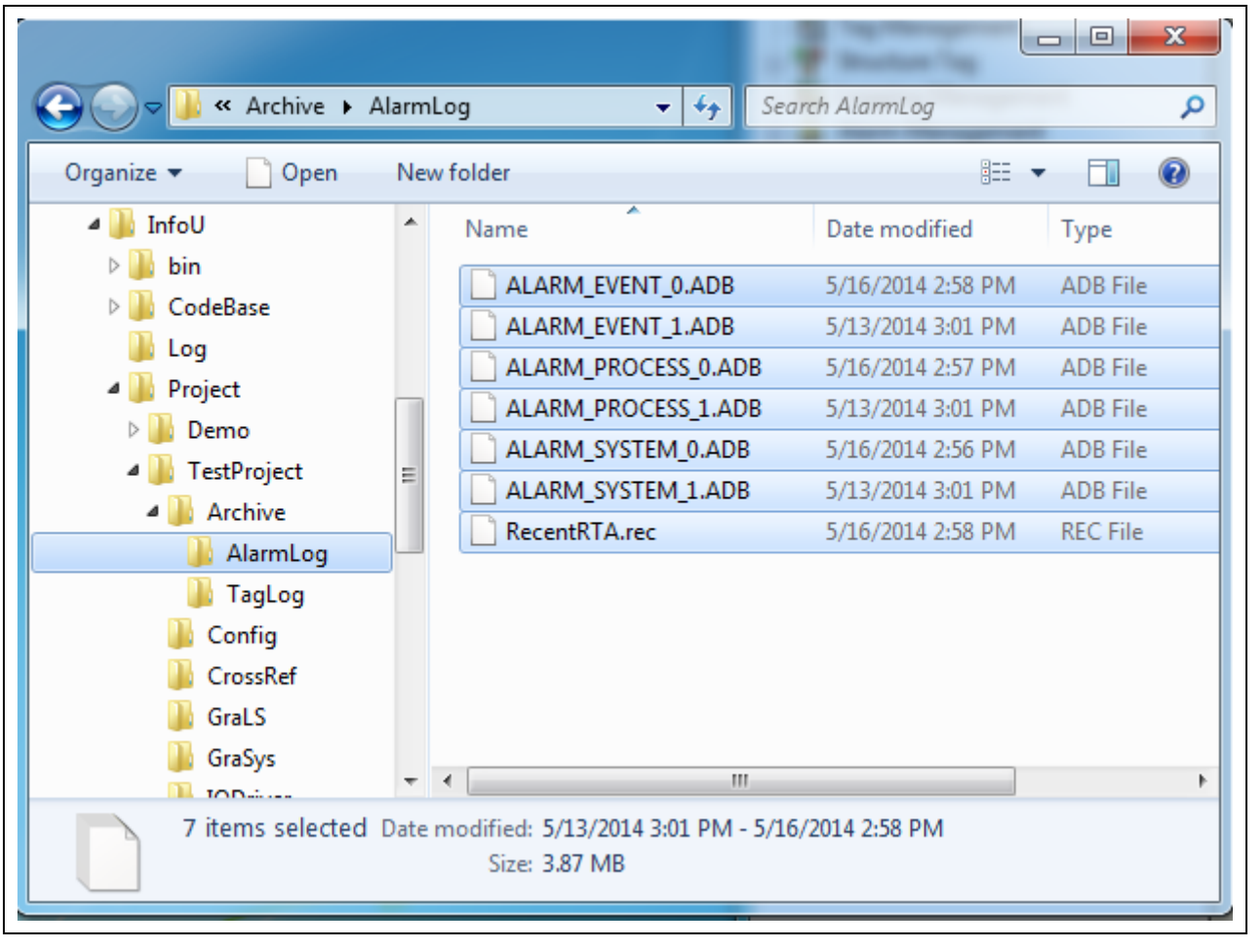
2) Process Alarm
   Input the number of process history alarms to be saved.

3) System Alarm
   Input the number of system history alarms to be saved.

4) Event
   Input the number of event history alarms to be saved.

Notice

If DB files already exist, editing is not available. To reset the storage period or the alarm number, it is needed to delete the existing alarm logging file.



(7) Backup File Storage Number

You may set up the number of backup DB files on history alarms.

1) Process Alarm

The number to backup DB files on process alarms

2) System Alarm

The number to backup DB files on system alarms

3) Event

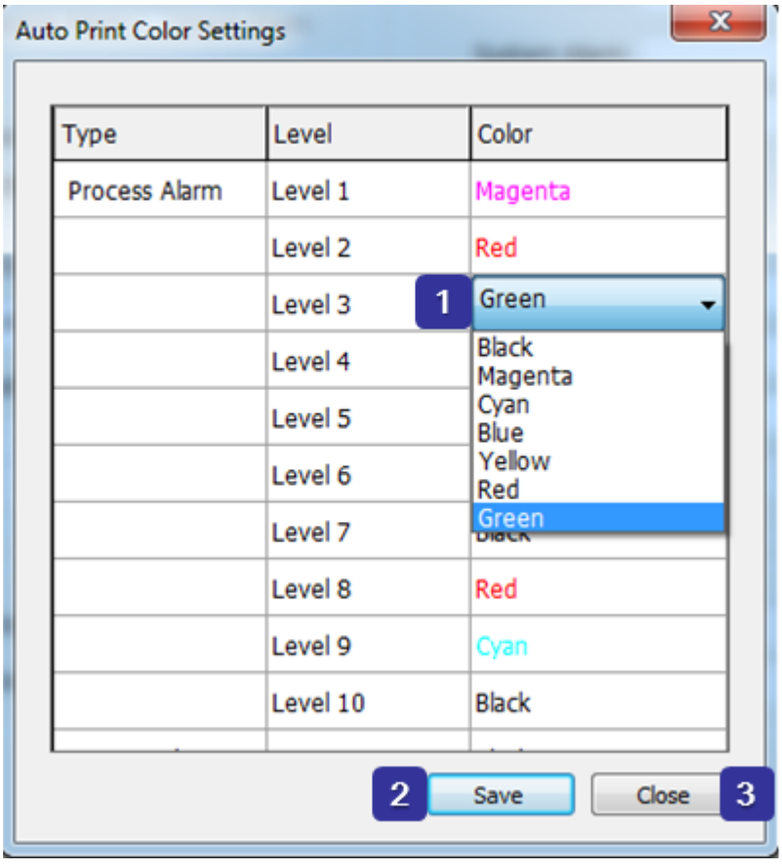
The number to backup DB files on event alarms

4) Full Storage Space

The storage space needed to maintain the selected number of backup files is displayed.

(8) Select Auto Print Color

You may set up print colors by levels and types when auto printing alarms. Click “Select Auto Print Color” button.



1) Auto Print Color Settings
Black is the default color and six colors such as pink, dark green, blue, yellow, red and green are available.

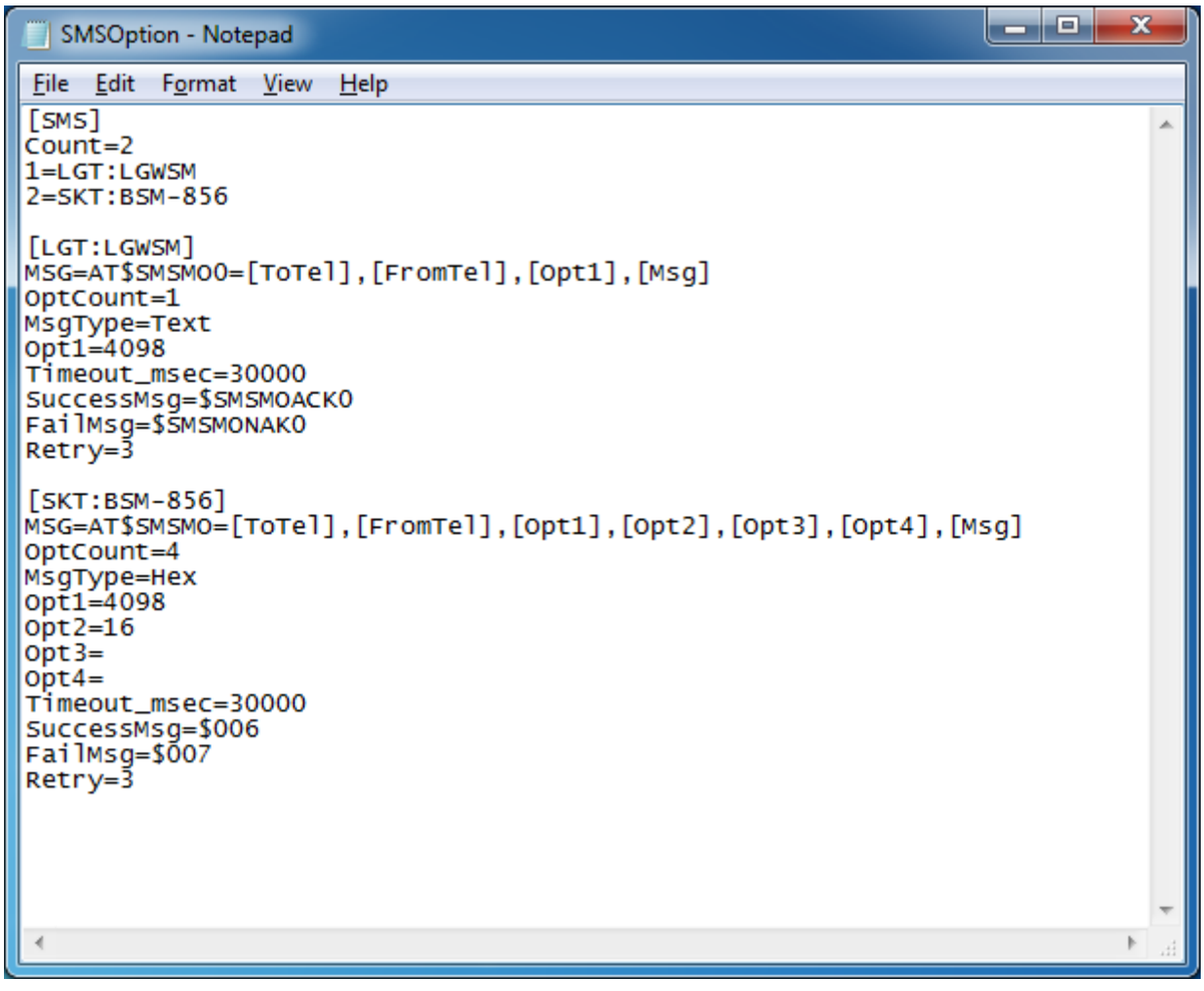
2) Save
Once color setting is completed, click 'Save' button to save the selected colors.

3) Close
The window for Auto Print Color Settings is closed.

(9) Save
Alarm settings are saved.

(10) SMSOption-Notepad
If any new modem control device is used in addition to the mandatory SMS terminal, additional instruction for users is presented.

1) SMSOption.ini File
There is SMSOption.ini file in the bin folder (ex>C:\InfoU\bin) of the InfoU installation folder (If there is no corresponding file, the file is automaticalled created when InfoU starts)
The file contains the following contents:



2) SMSOption.ini Edit

a. [SMS] Item

●Count: Specify the count number when adding the SMS terminal number to be displayed in the setting screen.

●1=LGT:LGWSM (Exist as many as the above count from 1. When setting SMS telecom company terminal, contents to be displayed in the combo box Count)

●2=SKT:BSM-856

●3= Telecom company: Terminal (write SMS terminal number and the telecom company name and terminal name to be newly added.)

b. [Telecom Company: Terminal]: Set up sending and receiving options for the telecom company and terminal model. (see the terminal manual)

●MSG: Write message types to send/receive to/from the SMS terminal.

[ToTel]: Receiver's telephone number

[FromTel]: Sender's telephone number

[Opt1]: If there is any option, write it.

●OptCount: Write the number of options

●MsgType: Select either Text or Hex (Be caution because specifications are different between

terminals)

●Opt1,…:Option character to be entered in each option part.

●Timeout_msec: Time to wait response from the telecom company

●SuccessMsg: Message code to receive from the terminal when sending is successful.

●FailMsg: Message code to receive from the terminal when sending is failed.

●Retry: Retry number when sending is failed.

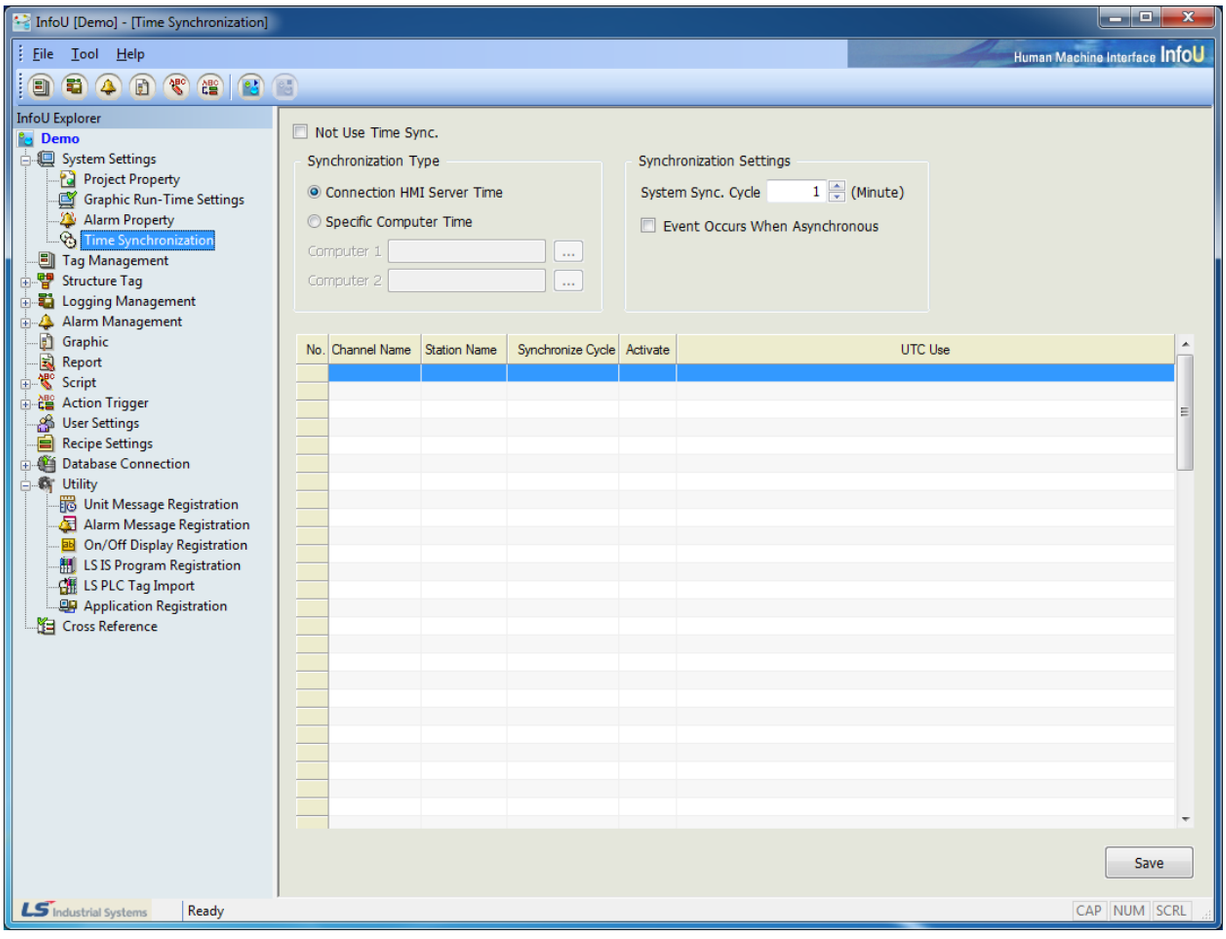
6.4 Time Synchronization

To prevent any improper time from being applied to all of the processes used in the system due to any wrong computer time during the real-time operation of the current project, time synchronization is executed by periodically obtaining time from the computer on which the project is executed or other computer.

This can be applied to the stand alone system provided in the InfoU system and other server/client systems and it also provides time synchronization with the I/O device station connected with the InfoU system.

For time in the system, basically UTC (Universal Time Coordinated) is used.

6.4.1 Default Screen



In the engineering main screen, select Time Synchronization from subclauses under System Settings of InfoU Explorer to display the setting screen above.Time Synchronization is selected as a default value he initial setting screen.

(1) General Settings

1) Not Use Time Sync

a. If selected

Time synchronization is not used between the computer in which InfoU system is installed and I/O device.

2) Synchronization Type
   a. Connection HMI Server Time
      The computer in which the current InfoU project is operated is designated
   b. Specific Computer Time
      The computer on the network is designated.

3) Synchronizing cycle setting service by a user (Unit: Min.)
   a. You can synchronize the computer time with the targeted subject in seconds set by the cycle.

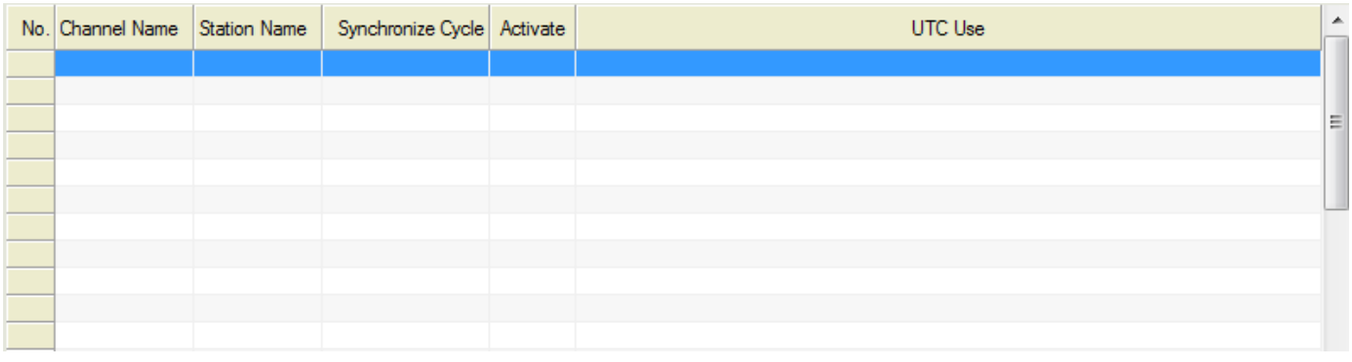
4) Event Occurs When Asynchronous
   a. If synchronization fails, it generates this information as an event after it carries out time synchronization with a local system.

5) System Sync. Cycle (Unit: Min.)
   a. The interval of the computer's time synchronization can be set up by the user.

6) Synchronization of the device that provides time synchronization
   a. Whether time synchronization is activated or not
   b. Even though a device provides time synchronization, it does not perform it if it is not activated.
      Time synchronization cycle (Unit: sec), Time unit (UTC/ Local Time)
   c. For time synchronization, UTC is used as basic time and if not specified, local time can be used.

(2) Device's Time Synchronization

Device's time synchronization is available only if full time synchronization is activated. Device's time synchronization does not apply to the entire devices but only to the stations that provide time synchronization while providing the following list to allow the user to select.



Also, for device's time synchronization, it can be set up and executed separately depending on whether each device is activated or not even through the user selects a certain time. The user can select either UTC or local time since a certain time display way exist for each station of the device rather than using UTC in the entire stations such as in full time synchronization

6.4.2 Scenario for the Whole System

(1) Standalone

A time synchronization way used for projects that are executed independently.

1) Connection HMI Time

Time is synchronized with the time of its own local system.

2) Specific Computer Time

a. Time synchronization is performed with the computer assigned as Computer 1.

b. If time synchronization fails in Computer 1, it is carried out with the time of Computer 2 after an event occurs (if set up).

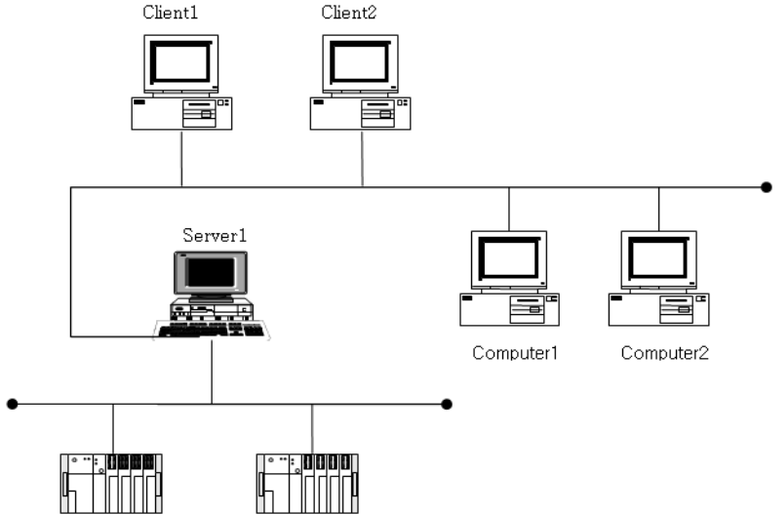
c. If time synchronization fails both in Computer 1 and Computer 2, it is carried out with the time of local system.

3) Notice

If time synchronization is carried out with local system due to the failure in time synchronization, time synchronization is attempted with the very first setting information.

(2) Server/Client-Single System

A time synchronization way used for projects that form a server and client relation to execute.



1) Connection HMI Time

a. Server

Time is synchronized with the time of its own local system.

b. Client

Time is synchronized with the time of the server connected with the project that is being executed.

2) Specific Computer Time

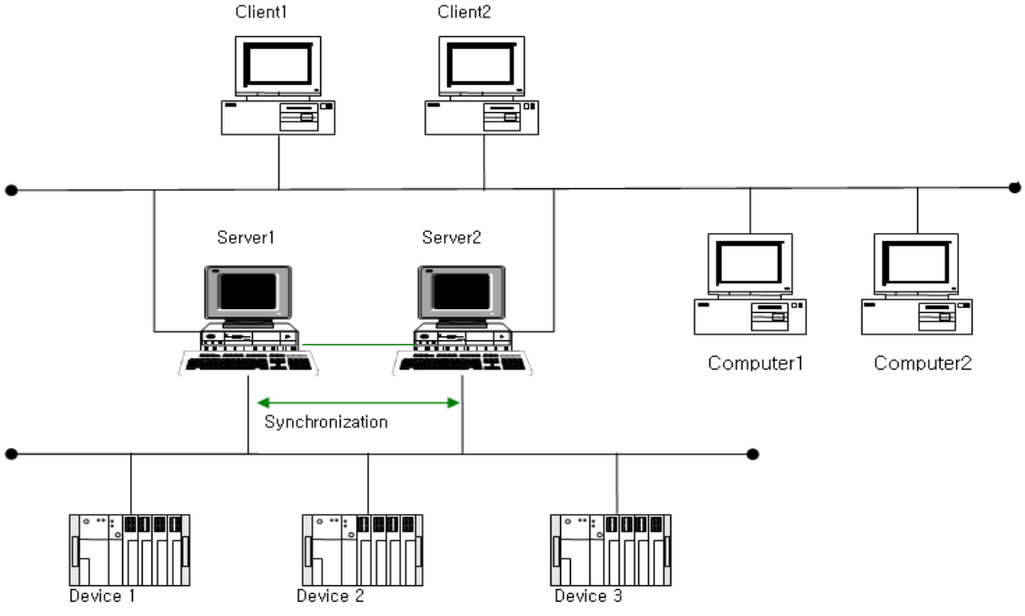
a. If all of the time synchronization fails, it is carried out with the time of the local system.

3) Notice

If time synchronization is carried out with local system due to the failure in time synchronization, time synchronization is attempted with the very first setting information.

(3) Server/Client-Redundancy System

A time synchronization way for projects that form a server and client relation and server redundancy



1) Connection HMI Time

a. Server (Running)

●Server 1 brings the information on whether it is in running or standby when it runs first and also brings the information on its partner-Server 2.

●If Server 1 is running, Server 1 uses its local time and records this information on the log.

●If Server 1 is in standby, Server 1 synchronizes its time with that of Server2,

b. Server (If redundancy is abstained)

●If Server 1 is running and Server 2 is in standby and Server 2's running is abstained by Server 1, Serve 2 uses local time and Server1 obtains time information from Server 2 to time synchronize after receiving information on the change in the system's running status

c. Client

●The client synchronizes time between the project under executing and the currently connected Server 1.

●If Server 1 is abstained to Server 2, redundancy system, due to any failure in the network or the system, the client that receives this information changes automatically from Server 1 to Server 2 for time synchronization and an event for the changed information on synchronization occurs.

2) Specific Computer Time

a. Server/Client

●It is synchronized with the time of Computer 1 connected with the network.

●If time synchronization fails in Computer 1, it is carried out with the time of Computer 2 after an event occurs (if set up).

●If time synchronization fails both in Computer 1 and Computer 2, it is carried out with the time of local system.

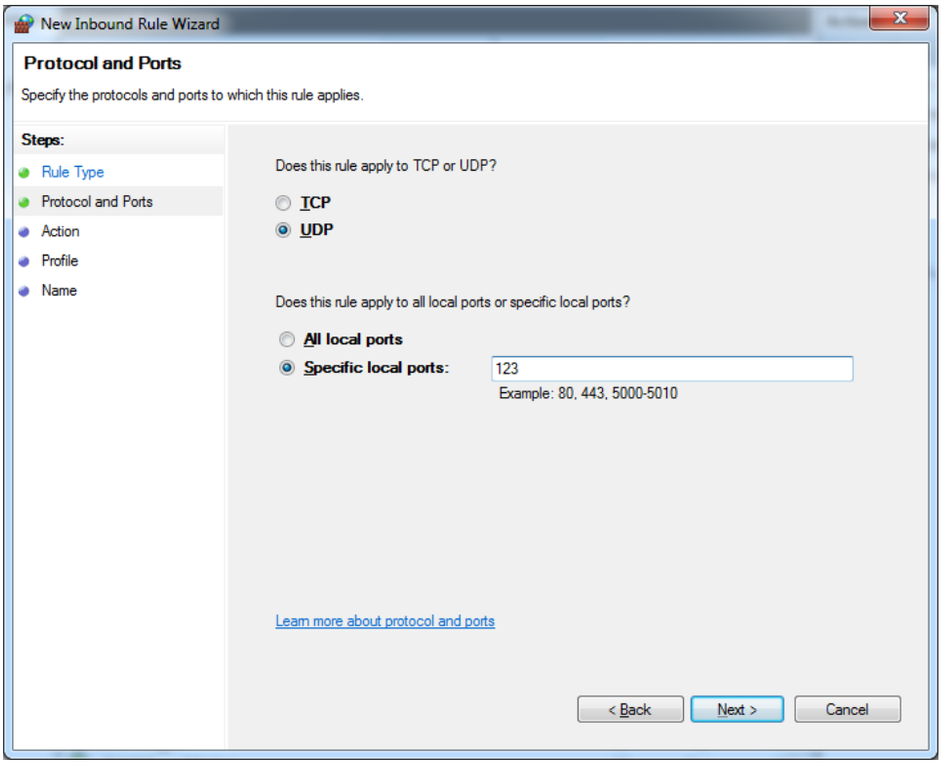
3) Notice

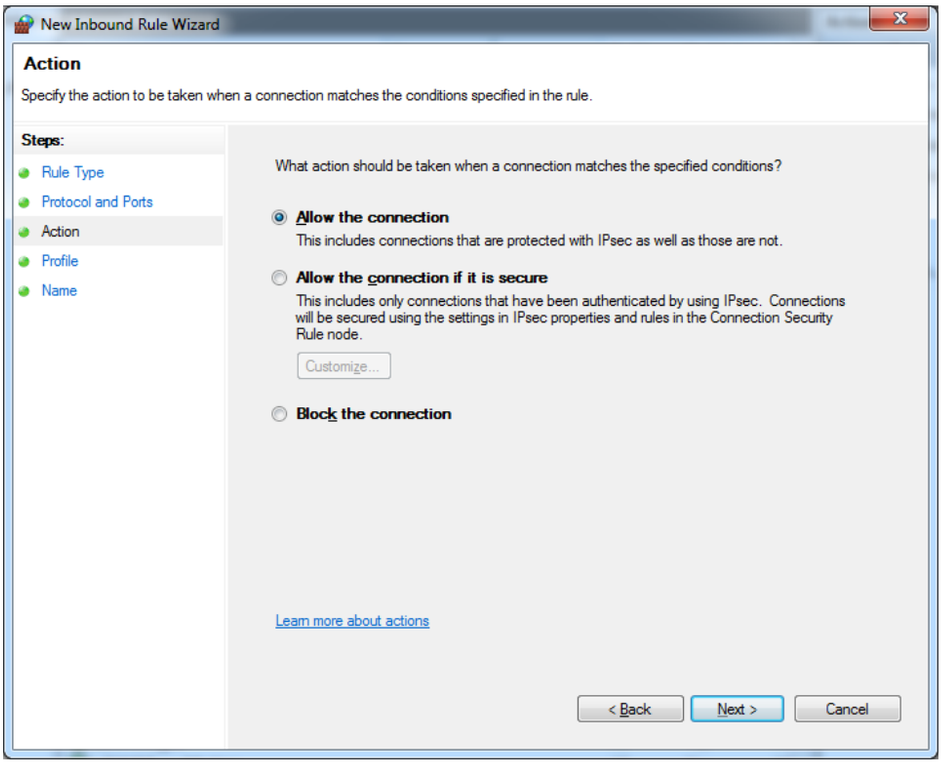
If time synchronization is carried out with local system due to the failure in time synchronization, time synchronization is attempted with the very first setting information.

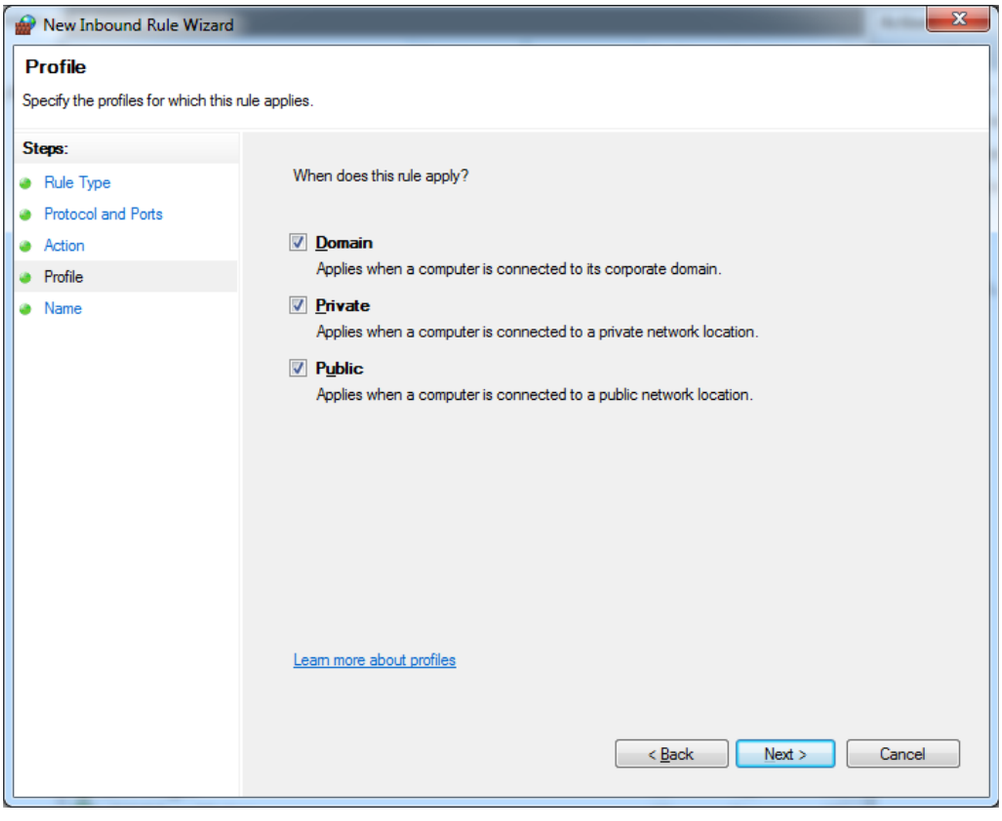
6.4.3 Time Synchronization Settings

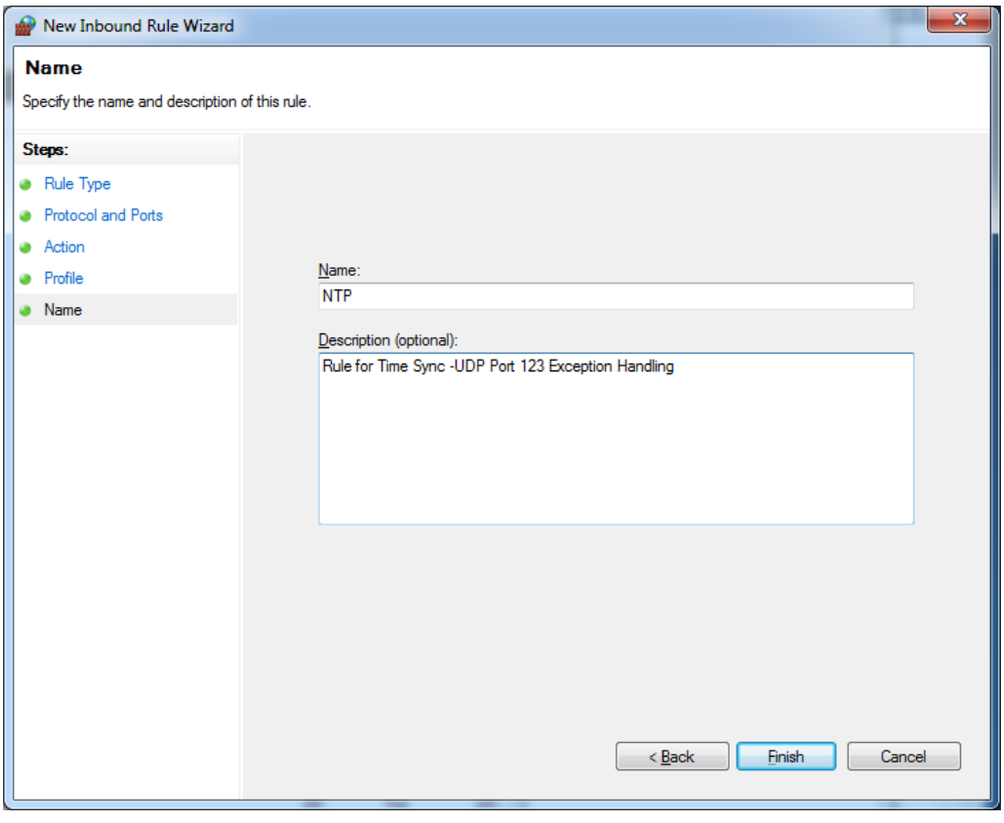
(1) Exceptional settings of firewall [server]

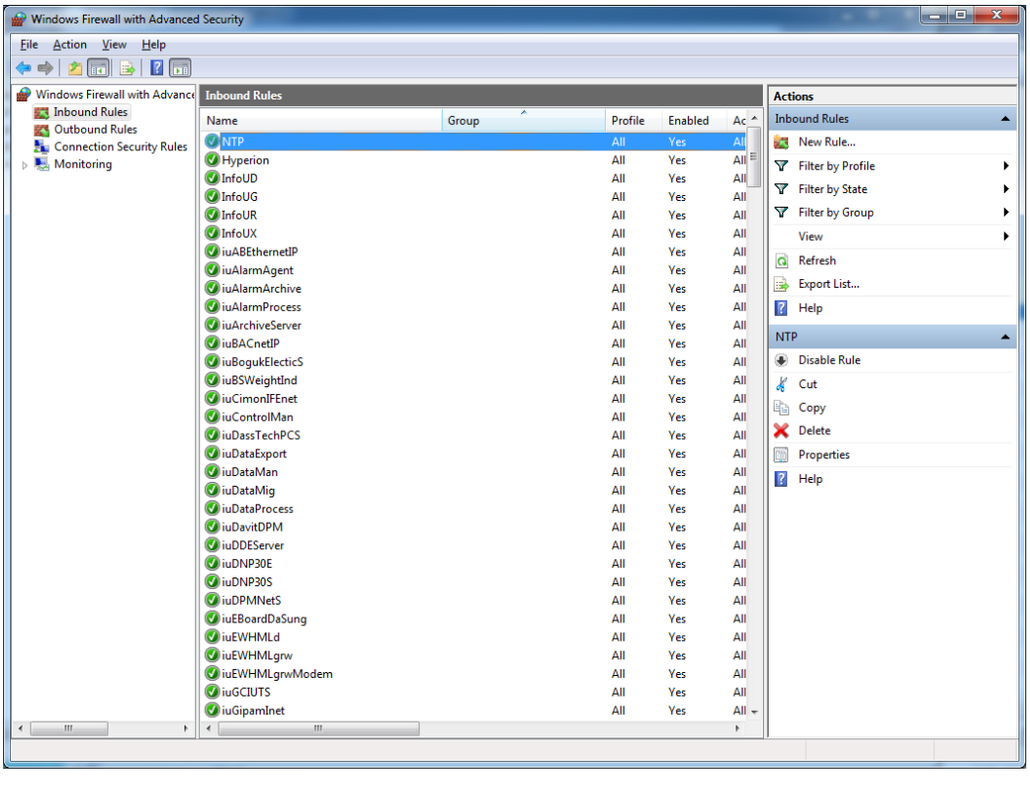
UDP port 123 is handled exceptionally. Set (Inbound) or Firewall Disable.

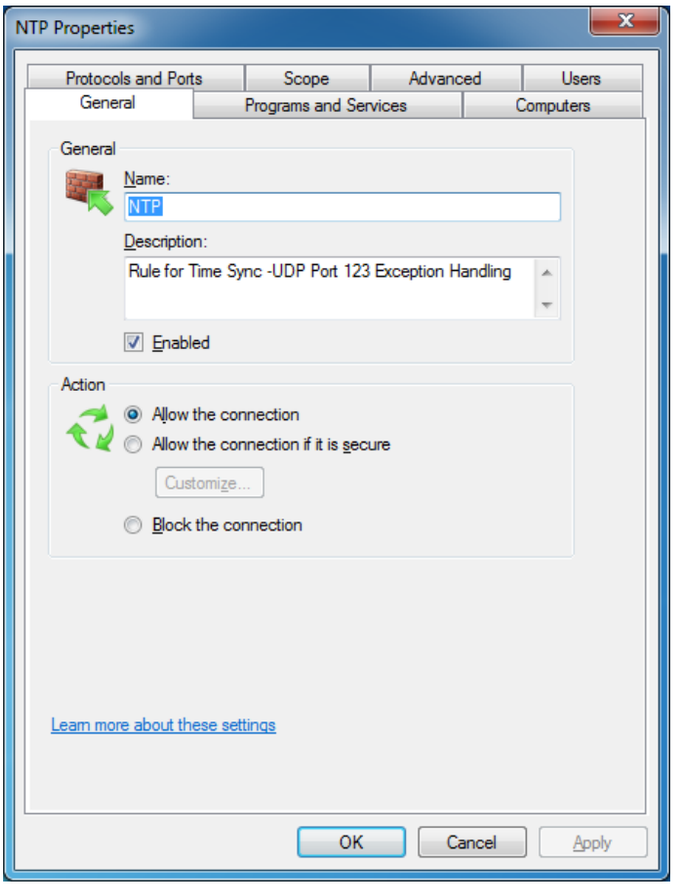



New Inbound Rule Wizard
Action
Specify the action to be taken when a connection matches the conditions specified in the rule.
Steps:
Rule Type
Protocol and Ports
Action
Profile
Name
What action should be taken when a connection matches the specified conditions?
Allow the connection
This includes connections that are protected with IPsec as well as those are not.
Allow the connection if it is secure
This includes only connections that have been authenticated by using IPsec. Connections will be secured using the settings in IPsec properties and rules in the Connection Security Rule node.
Customize...
Block the connection
Learn more about actions
< Back
Next >
Cancel


New Inbound Rule Wizard
Profile
Specify the profiles for which this rule applies.
Steps:
Rule Type
Protocol and Ports
Action
Profile
Name
When does this rule apply?
Domain
Applies when a computer is connected to its corporate domain.
Private
Applies when a computer is connected to a private network location.
Public
Applies when a computer is connected to a public network location.
Learn more about profiles
< Back
Next >
Cancel

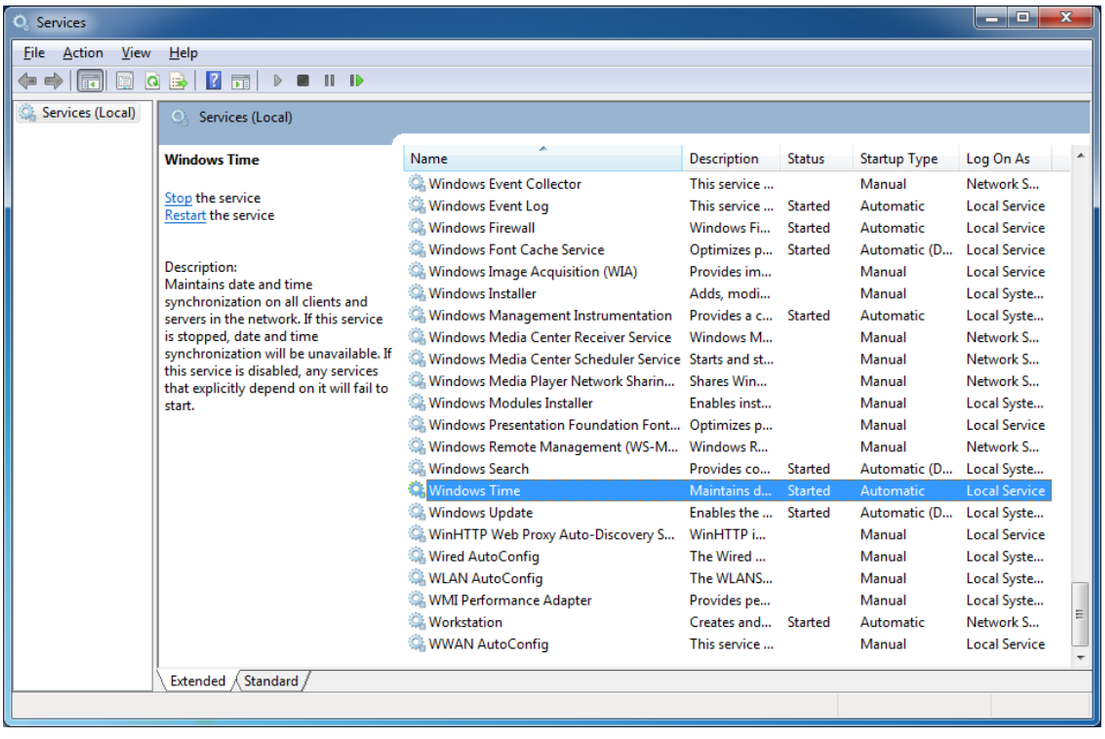

New Inbound Rule Wizard
Name
Specify the name and description of this rule.
Steps:
Rule Type
Protocol and Ports
Action
Profile
Name
Name:
NTP
Description (optional):
Rule for Time Sync -UDP Port 123 Exception Handling
< Back
Finish
Cancel

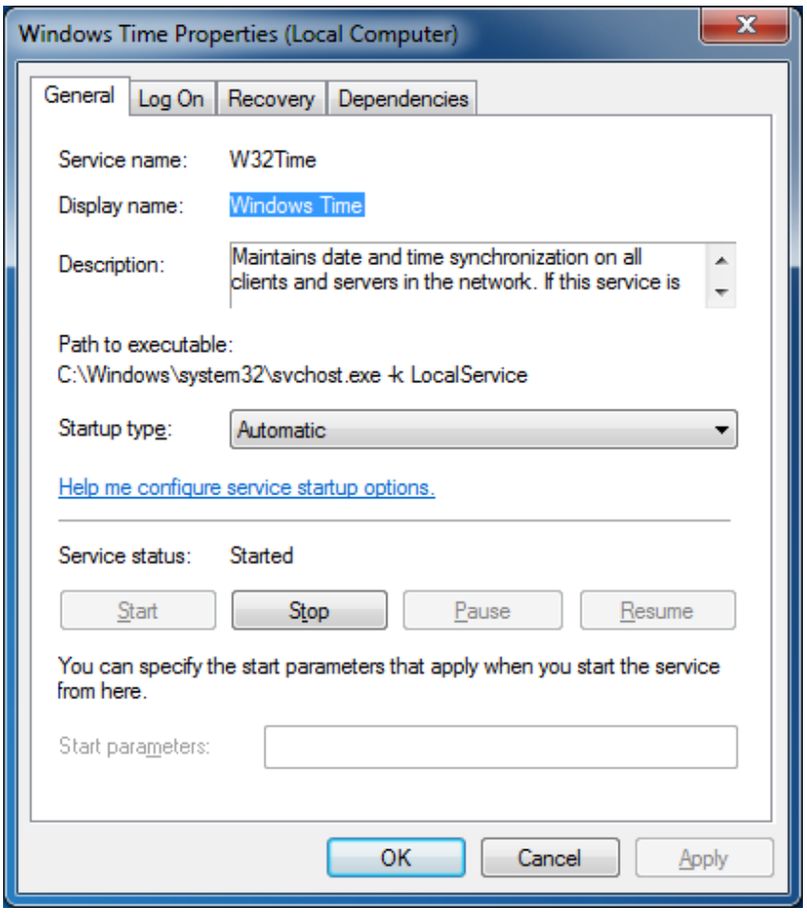

Windows Firewall with Advanced Security
File Action View Help
Windows Firewall with Advanc
Inbound Rules
Outbound Rules
Connection Security Rules
Monitoring
Inbound Rules
Name Group Profile Enabled
NTP All Yes
Hyperion All Yes
InfoUD All Yes
InfoUG All Yes
InfoUR All Yes
InfoUX All Yes
iuABEthernetIP All Yes
iuAlarmAgent All Yes
iuAlarmArchive All Yes
iuAlarmProcess All Yes
iuArchiveServer All Yes
iuBACnetIP All Yes
iuBogukElecticS All Yes
iuBSWeightInd All Yes
iuCimonIFEnet All Yes
iuControlMan All Yes
iuDassTechPCS All Yes
iuDataExport All Yes
iuDataMan All Yes
iuDataMig All Yes
iuDataProcess All Yes
iuDavitDPM All Yes
iuDDEServer All Yes
iuDNP30E All Yes
iuDNP30S All Yes
iuDPMNetS All Yes
iuEBoardDaSung All Yes
iuEWHMLd All Yes
iuEWHMLgrw All Yes
iuEWHMLgrwModem All Yes
iuGCIUTS All Yes
iuGipamInet All Yes
Actions
Inbound Rules
New Rule...
Filter by Profile
Filter by State
Filter by Group
View
Refresh
Export List...
Help
NTP
Disable Rule
Cut
Copy
Delete
Properties
Help


NTP Properties
Protocols and Ports
Scope
Advanced
Users
General
Programs and Services
Computers
General
Name:
NTP
Description:
Rule for Time Sync -UDP Port 123 Exception Handling
Enabled
Action
Allow the connection
Allow the connection if it is secure
Customize...
Block the connection
Learn more about these settings
OK
Cancel
Apply

(2) Services running [Server]

Among the service items, check whether the Windows Time item has been started.



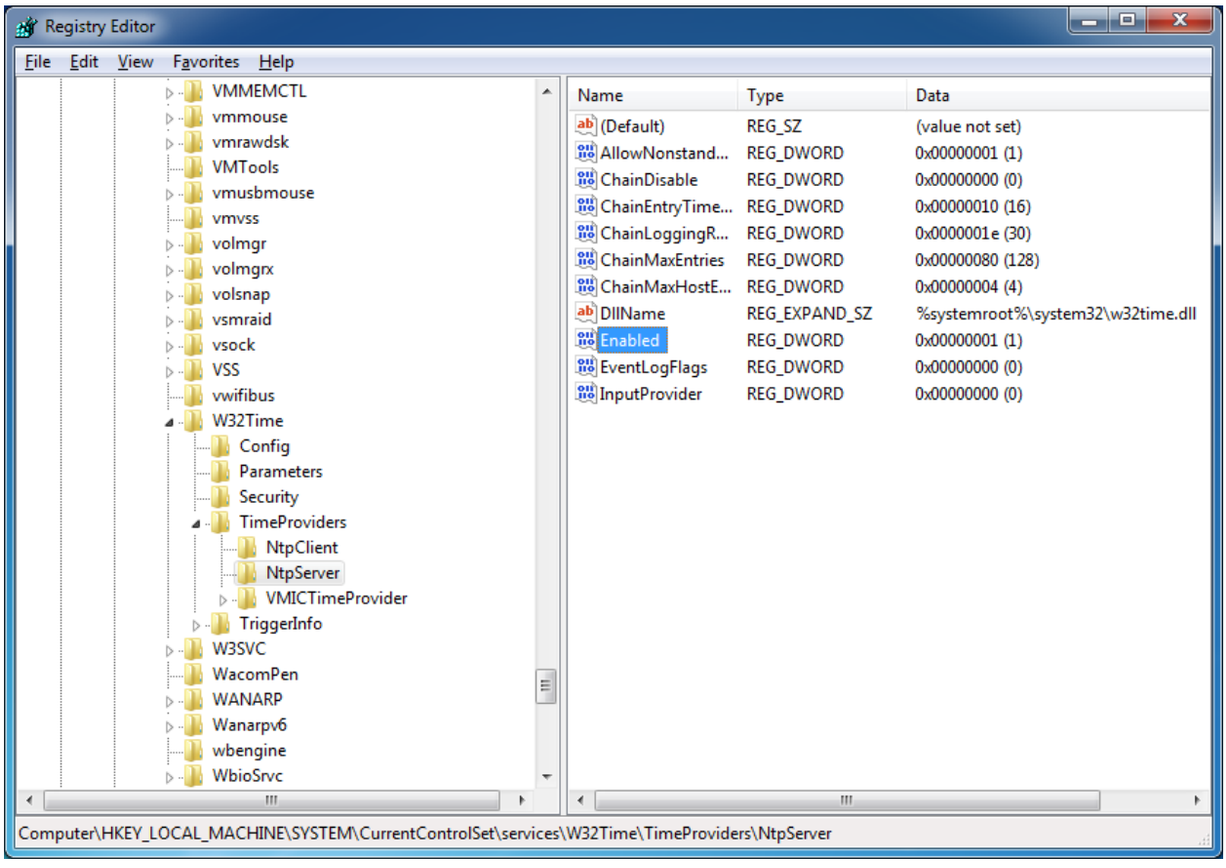


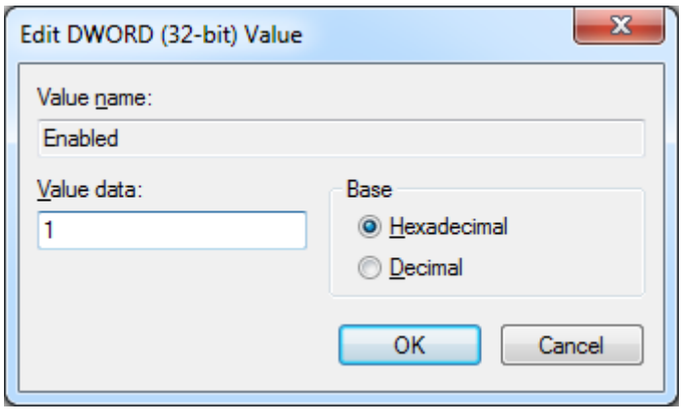
(3) Registry settings [Server, Client]

1) Server

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W32Time\TimeProviders\NtpServer

Enabled value data set to 1

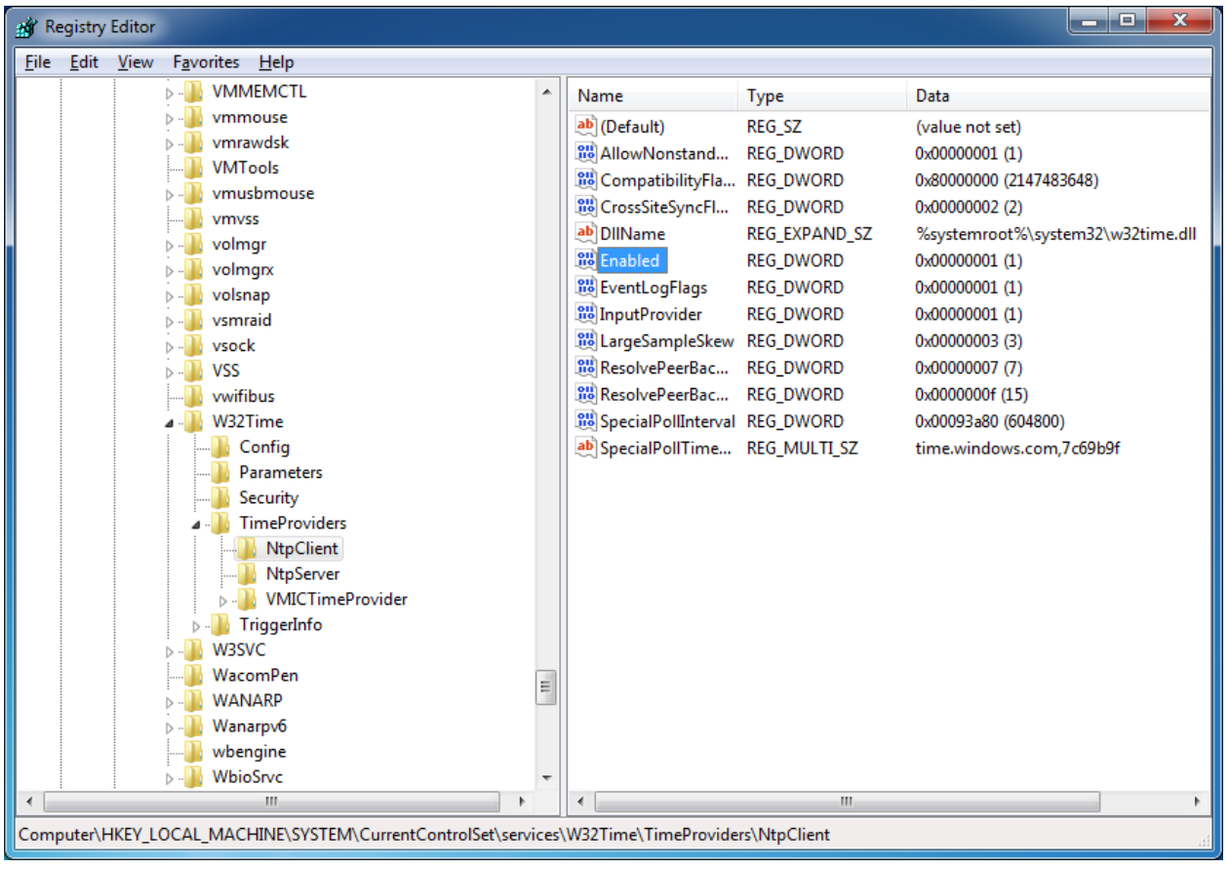




2) Client

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W32Time\TimeProviders\NtpClient

Enabled value data set to 1



Chapter 7 I/O Driver

This software is to collect data from a device and transfer to the upper process.
The user can add or set up Channel and Station in Tag Management.

7.1 I/O Driver Setting

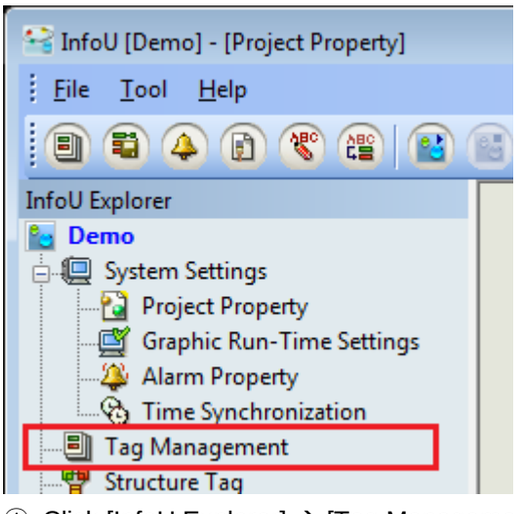
The user can add, delete or modify Channel and Station through pop-up menu if each device is displayed in a tree structure. This section briefly explains how to edit each item.

7.1.1 Communication Channel

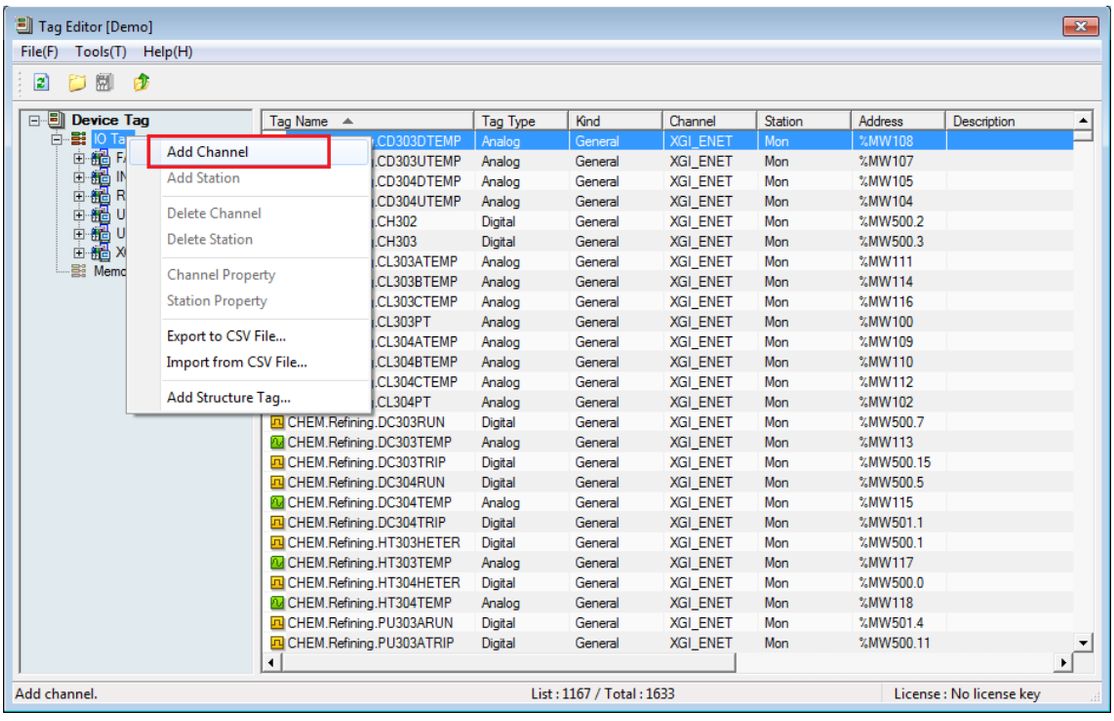
Channel refers to a channel to be communicated and selected among supporting I/O Drivers. Also, it basically provides a “memory” type of channels to classify input/out tags and memory tags in InfoU.

(1) Configuration

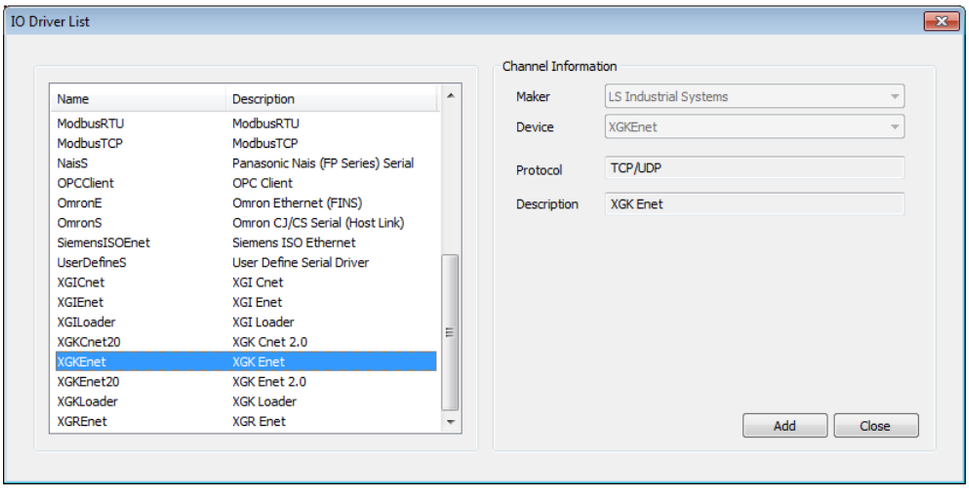
To add communication channels, run the menu based on the below procedures.



① Click [InfoU Explorer] → [Tag Management]



② Run [Device Tag] → [I/O Tag] → [Add Channel].



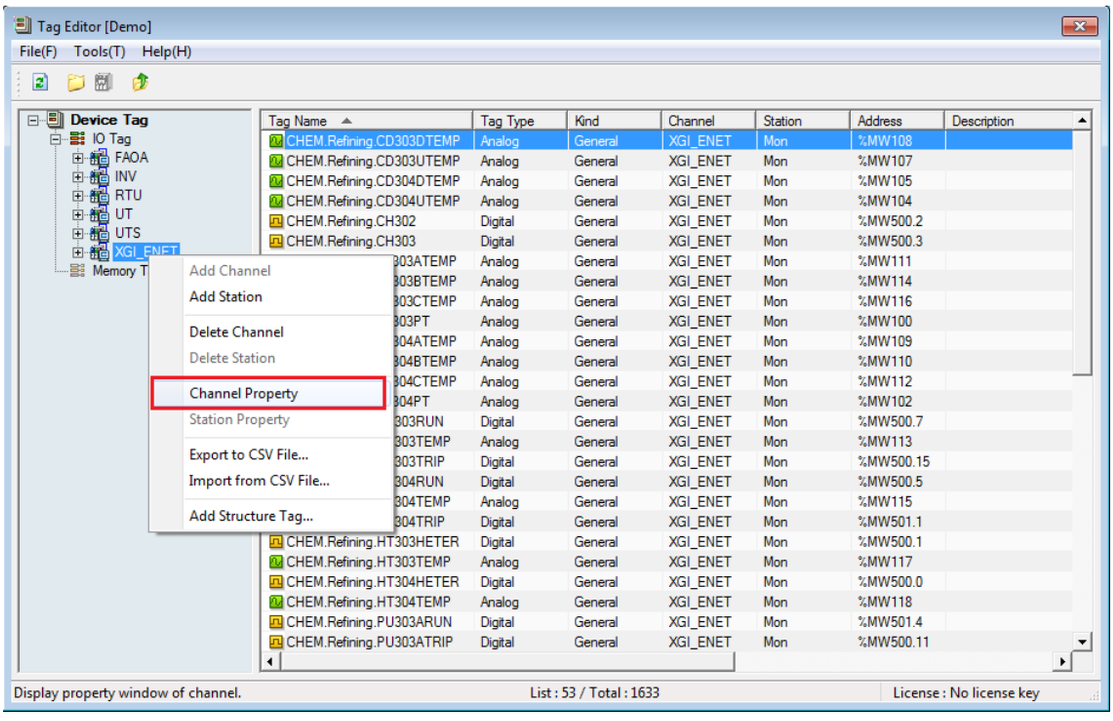
③ Select the desired communication driver in the communication driver list and click the [OK] button.

Notice

☞ The setting method may be different depending on the communication driver.

☞ For settings of each communication driver, refer to [Communication Driver] of the appendix.

(2) Edit



To edit the communication channel, double-click the relevant communication channel or click with the right mouse button then run [Communication Channel Property] in the popup menu.

7.1.2 Station

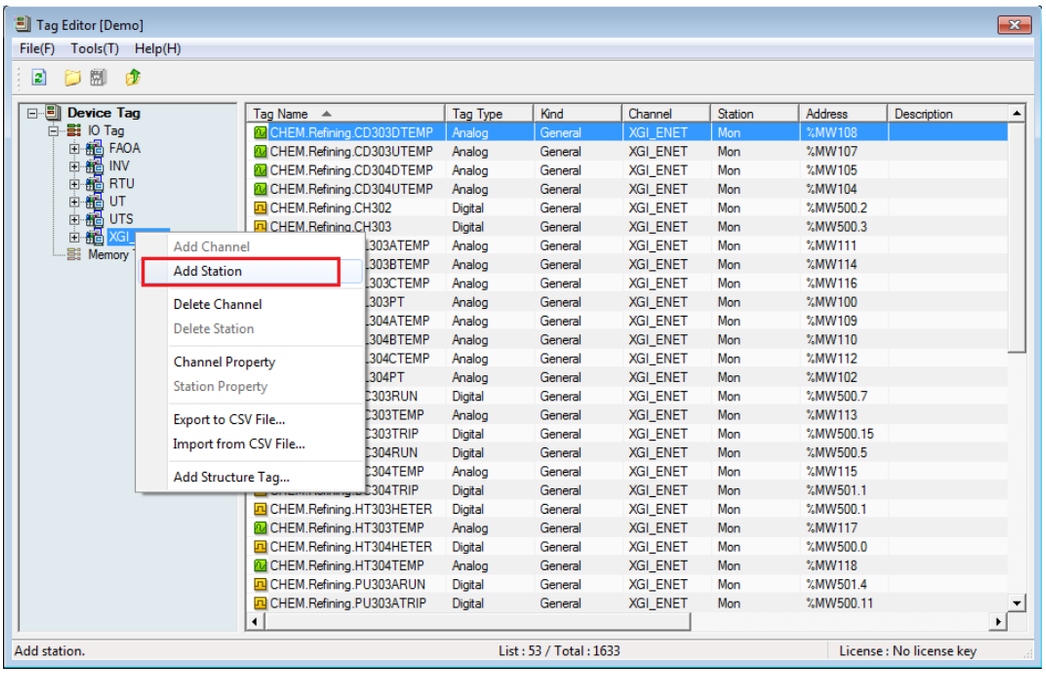
The station means the actual device belonging to the upper communication channel such as PLC or Inverter.

Notice

☞ The station cannot exist by itself so it should be added and subordinated to the communication channel.

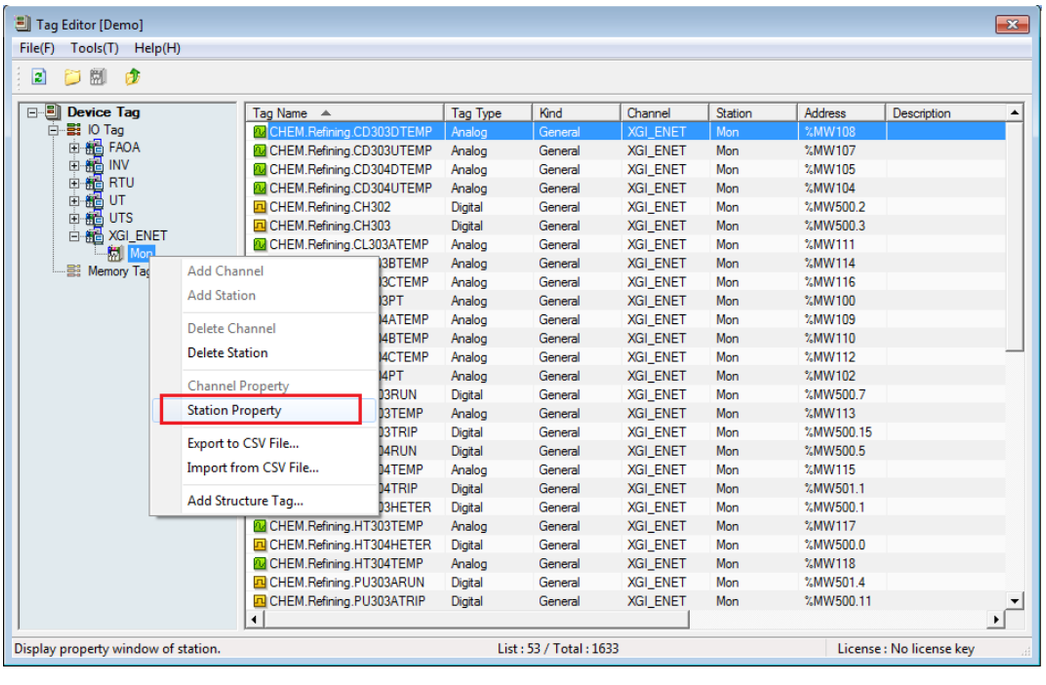
☞ For some communication drivers, you can add the station even in the communication driver setting window.

(1) Add



After clicking with the right mouse button on [Device Tag] → [I/O Tag] → [“Communication Channel Name”], run [Add Station] in the popup menu.

(2) Edit



In [Device Tag] → [I/O Tag] → [“communication channel name”] → [“station name”], double-click the station name that you want to edit or after clicking with the right mouse button, run [Station Property] in the popup menu.

Chapter 8 Tag Management

8.1 Overview

The tag editor is the tool to register and manage the tags that are the basis of the HMI system. All functions supported and operated by the HMI system runs based on the tags registered in the tag DB. Tags can be largely divided into I/O tags that receive signal from the on-site equipment and memory tags registered virtually by necessity of the upper system.

The InfoU shows the tag information obviously and neatly through the property window and a user can easily perform the tag engineering works through the InfoU. In addition, it supports the clipboard and CSV file's I/O so you can register and manage mass tag data easily and promptly.

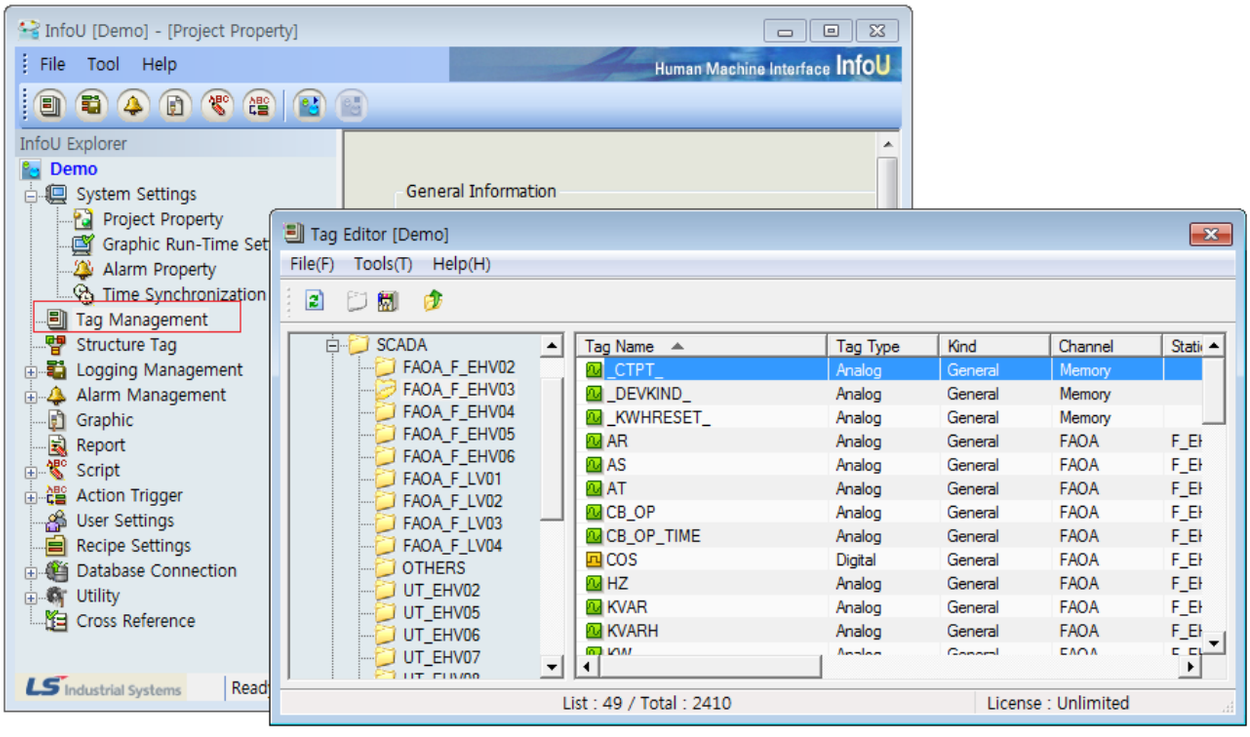
To execute the tag editor, click [Tag Management] from the menu tree items on the left side of the InfoU engineering module or select [Tag Management] in the main menu or toolbar of the InfoU engineering module.

8.1.1 Definitions of Terms

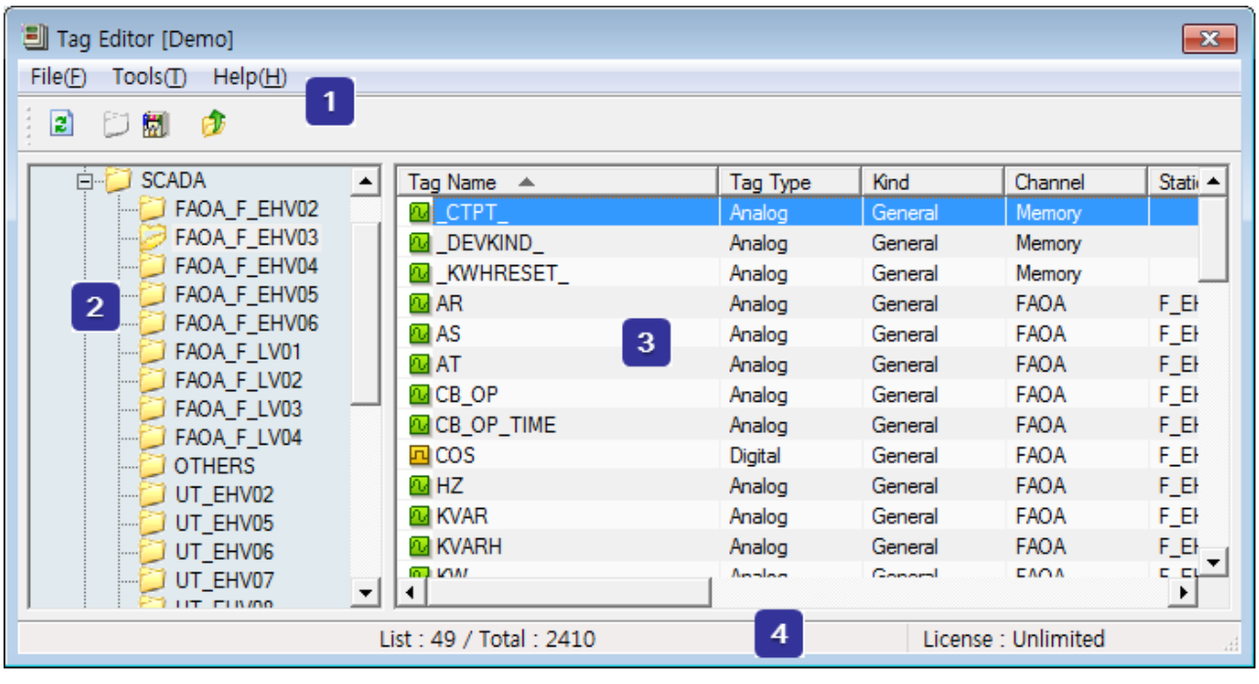
Terms	Description
Tag editor	It is the module of InfoU engineering program, which performs a variety of editions for tags.
Analog tag	It means the analog type of tags that can be largely divided into the integer and real number type.
Digital tag	It has two tag values; 0 or 1 (Open/Close, ON/OFF).
Group tag	It is the virtual tag to constitute the tag with logical layer information. For example, it acts as a folder of the Explorer.
General tag	It means any tag created by a user's direct engineering.
System tag	The tag is automatically created when registering the system. It is mainly related to the unique status and operations of the equipment such as the conditions, the number of reading, etc.
Analog alarm	The alarm occurs for the analog tag. When the analog tag values exceed the limits of the range, variations, etc., the alarm will occur. After that, if the values come into the normal range, the alarm will be cleared.
Digital alarm	The alarm occurs for the digital tag. When the digital tag value is 0 or 1, or it varies, the alarm is set up for processing.
CSV file	It is the text format file and each item is separated by a comma (,). It can be compatible with Excel and used to manage the mass tag in the tag editor.
Structure tag template	In case there is the tag set of the specific device or the tag that should be created repetitively since its structure is fixed, this template defines the name of the patterns of these tags as a bunch.
Structure tag item	It means the common prototype of each tag forming the structure tag template.

8.1.2 Execution and Main Screen

To execute the tag editor, click [Tag Management] from the menu tree on the left side of the InfoU engineering program or select [Tool] → [Tag Management] in the menu.



The screen of the executed tag editor is as shown below.



The composition of the tag editor is as shown below table.

(1) Menu / toolbar

It is composed of the basic menu and toolbar buttons such as Refresh, View Transition, Move to upper group, etc.

Toolbar icon	Description
	The icon indicates Refresh. It refreshes the details of tag database.
	The icon indicates Group Tag View. It converts the deployment scheme of the tag into Group View.
	The icon indicates Device Tag View. It converts the deployment scheme of the tag into Device View.
	The icon indicates Move to Upper Group. It moves the tag to the upper tree items in the tree structure.

(2) Tag structure tree

It deploys the tag composition in a tree image to display many tags in an efficient manner. The deployment of the tag composition has two types; [Device View] deploying tags through the physical structure that is actually connected to devices; [Group View] composing the tag structure with logical groups. A user can select the desired tag structure to perform tag editions effectively.

(3) Tag list

It shows the list of the tags belonging to the items selected from the tag structure tree. The properties shown in the list are the typical items of the tags. A user can select the shown column or change the position of the column or specify the order of showing tags.

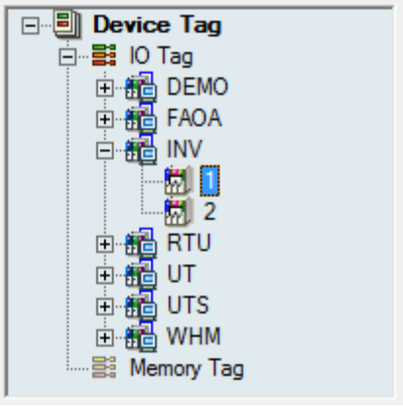
(4) License information

It shows the InfoU's license information simply. It also shows the validity of the license and the number of tags.

8.1.3 Conversion of Tag View

The conversion of tag view is the function to manage tags easier by converting the deployment method of the tag tree into a user's desired one. You can convert the deployment into [Device View] and [Group View] by clicking the appropriate button of the toolbar menu.

(1) Device view tree structure



It can be used when you want to view tags in the way physically connected to devices. The group is not visible in the device view. The device view tree structure is composed of the following tree items.

1) Tag management(fixed item)
   It shows the whole tags, however, the group tag is not visible.

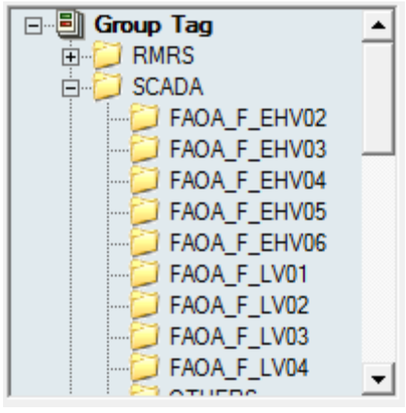
2) I/O tag (fixed item)
   It is the tag that is actually connected to the I/O point of the on-site device. There are communication channels for its sub-item. When selecting this, all I/O tag lists will be shown.

3) Communication channel (variable item)
   It means the type of the channel to communicate and you can see the channel selected among available I/O Drivers. Namely, the channels only registered during tag engineering can be shown. When selected from the tree, the tag list belonging to the relevant communication channel is visible.

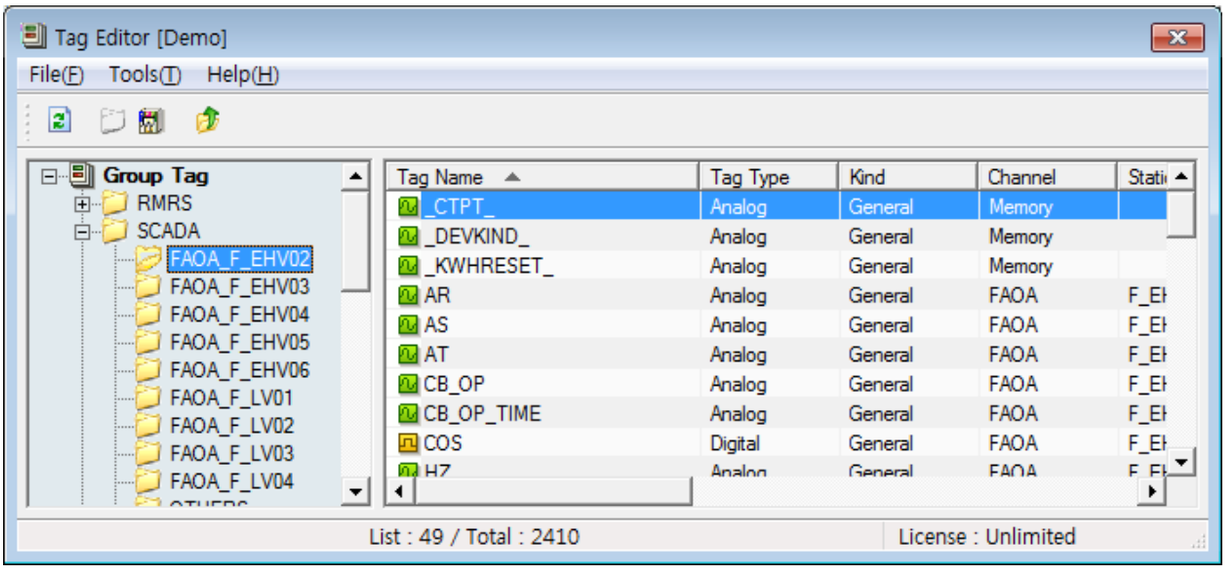
4) Station (variable item)
   It is the sub-item of the communication channel and means the actual devices such as the PLC, inverter. When selecting this, the tag list belonging to the relevant station will be shown.

5) Memory tag (fixed item)
   It means the virtual memory tag for operations in HMI system. When selecting this, all memory tags except the group tags will be shown.

(2) Group view tree structure



In contrast with the above, the group view tree structure shows tags by composing logical groups. The creation of each group leads to the creation of group tags and it includes the sub-tag. Each group can have the sub-group. When selecting one group, the tag belonging to the selected group will be shown with the sub-group.



8.2 Configuration of Communication Equipment

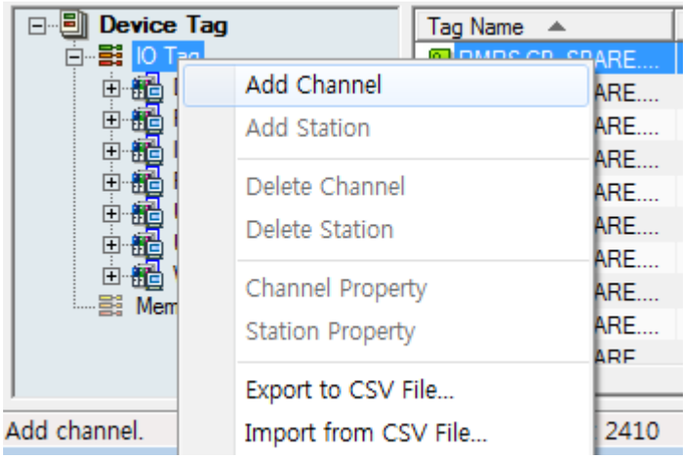
In the case of the device view tree structure, you can add, delete, and change the communication channels and stations through popup menus. This section briefly describes how to edit each time. For each property of the communication channels, stations with detailed usage, refer to the “I/O driver” manual.

8.2.1 Communication Channel

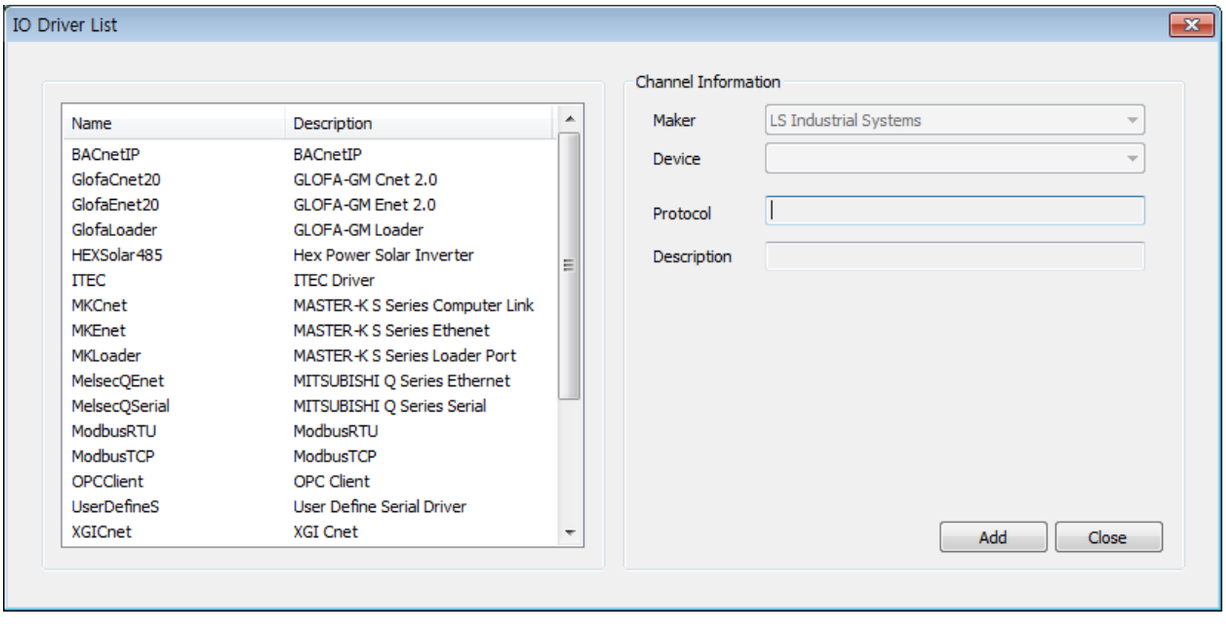
It means the type of the channel to communicate and can be selected among the available I/O drivers

(1) Addition

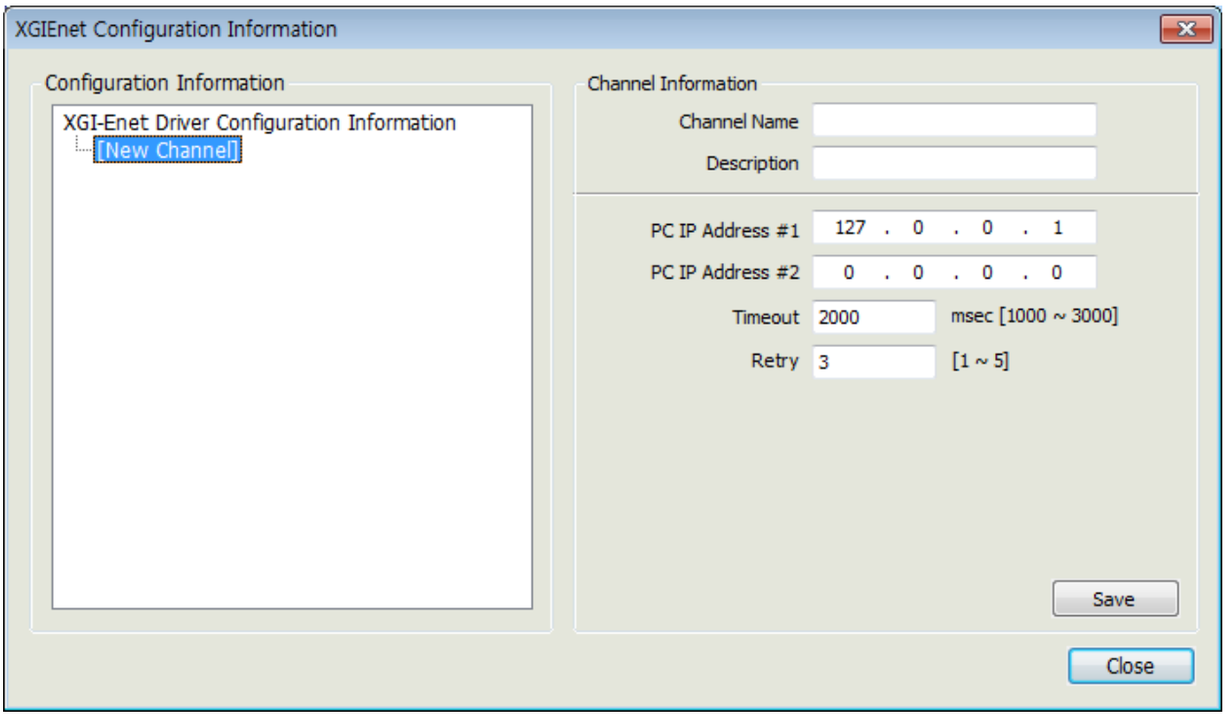
To add communication channels, after clicking with the right mouse button on the I/O tag items, select [Add Channel] menu.



If you select the menu, the available I/O diver lists will be shown.



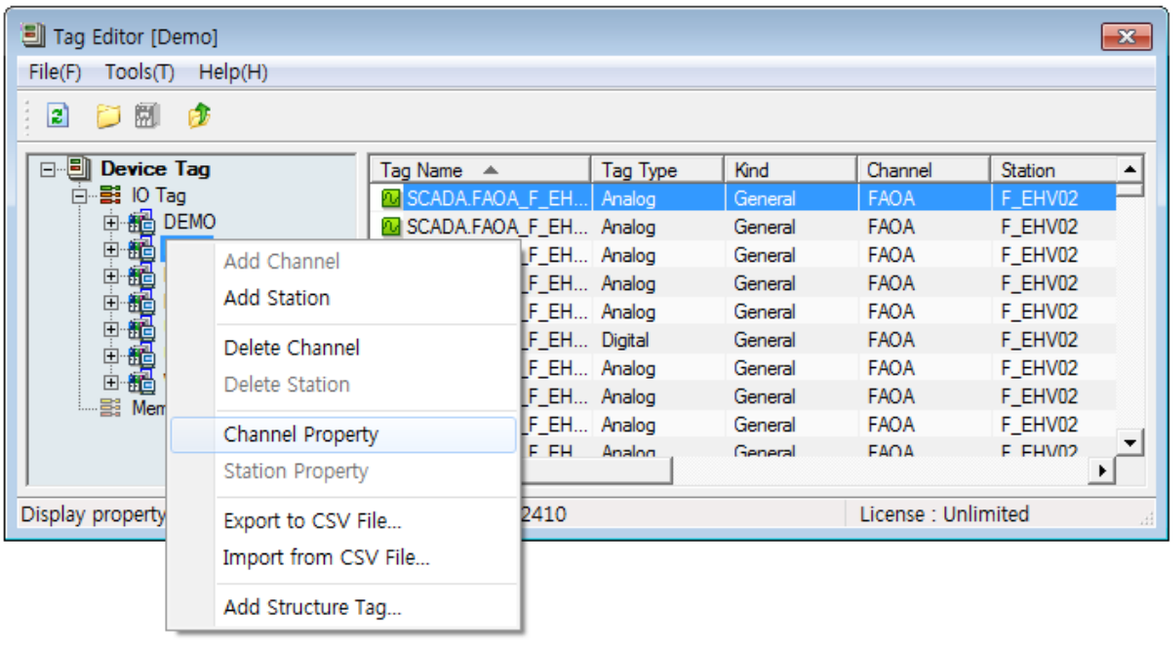
If you select the item to add in the communication driver list and click [Add], the window to edit the properties of the relevant communication channel will be shown.



For more details, refer to the "I/O Driver" manual.

(2) Edition of Properties

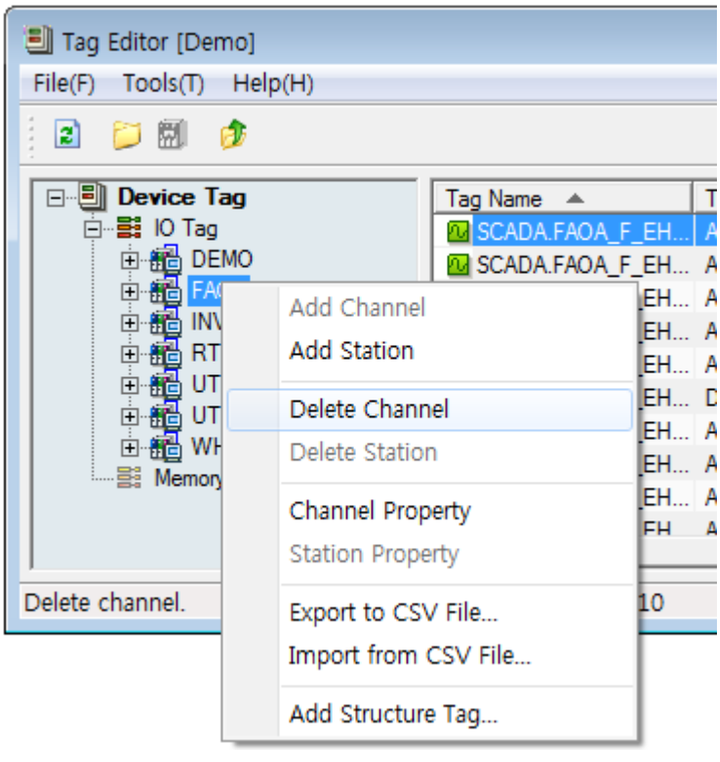
To edit the properties of the registered communication channel, click with the right mouse button on the relevant communication channel and select [Channel Property]. (Or you can double-click the tree item)



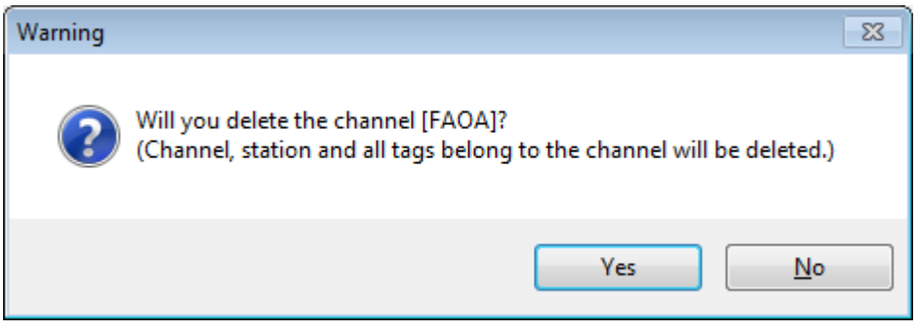
If you select [Channel Property], the window displaying the properties for the registered communication channel will show up. Through this window, a user can change the properties of the communication channel and save them.

(3) Deletion

It is the function to delete the registered communication channel with the stations and tags belonging to the registered one. Select the communication channel to delete and click with the right mouse button and then, select [Delete Channel] in the popup menu.



If you select [Delete Channel], the following warning message will show up.



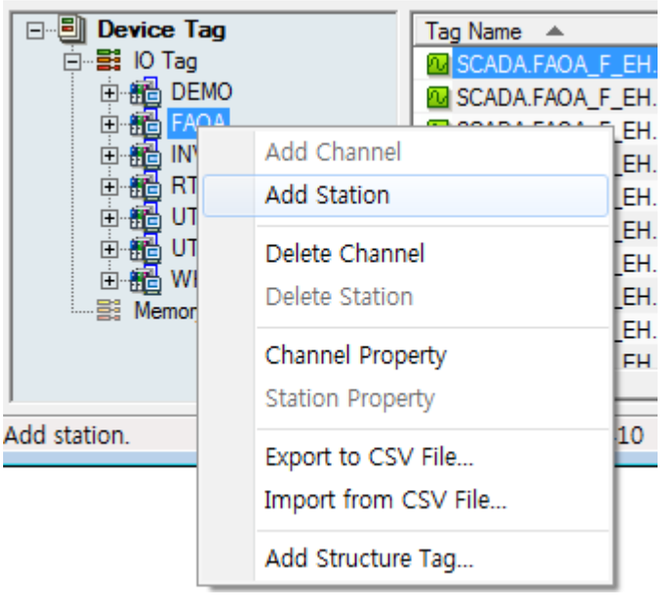
If you select [Yes], the 'Delete' function will be executed; if you select [No], the function will be canceled. When you delete the data once, it will be deleted permanently so be careful of this.

8.2.2 Station

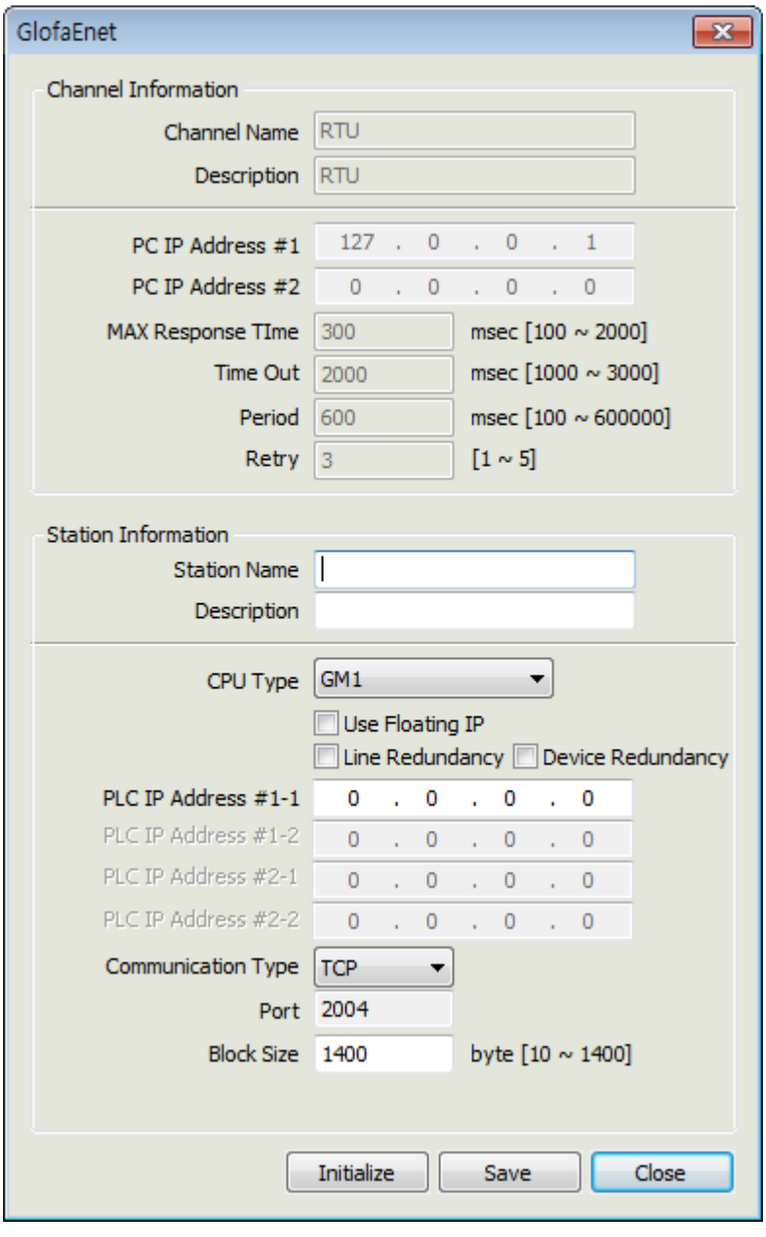
The station means the actual devices such as the PLC, inverter acting as the communication channel, which is the type of communication channels. Its menu deployment method is similar to the communication channels.
For more details on each station, refer to the "I/O Driver" manual.

(1) Addition

To add stations, after selecting the type of the relevant communication channels, click with the right mouse button and execute [Add Station].



Then, you can see the screen for the relevant station.
If you input the details and save them, the relevant station will be added to the sub-item of the communication channel.



(2) Edition of Properties

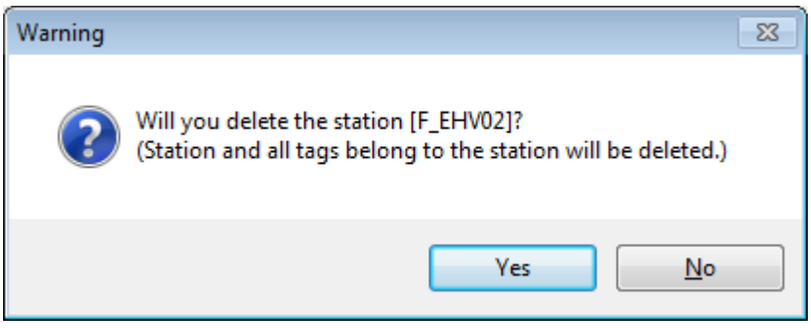
Then, you can see the screen for the relevant station.

If you input the details and save them, the relevant station will be added to the sub-item of the communication channel.

(3) Deletion

It is the function to delete the registered station with the tags belonging to the registered one. Select the station to delete and click with the right mouse button and then, select [Delete Station] in the menu.

Then, the following warning message will show up.



If you select [Yes], the 'Delete' function will be executed; if you select [No], the function will be canceled.
In common with the communication channel, when you delete the data once, it will be deleted permanently so be careful of this.

Notice

You should pay special attention to delete communication channels and stations.

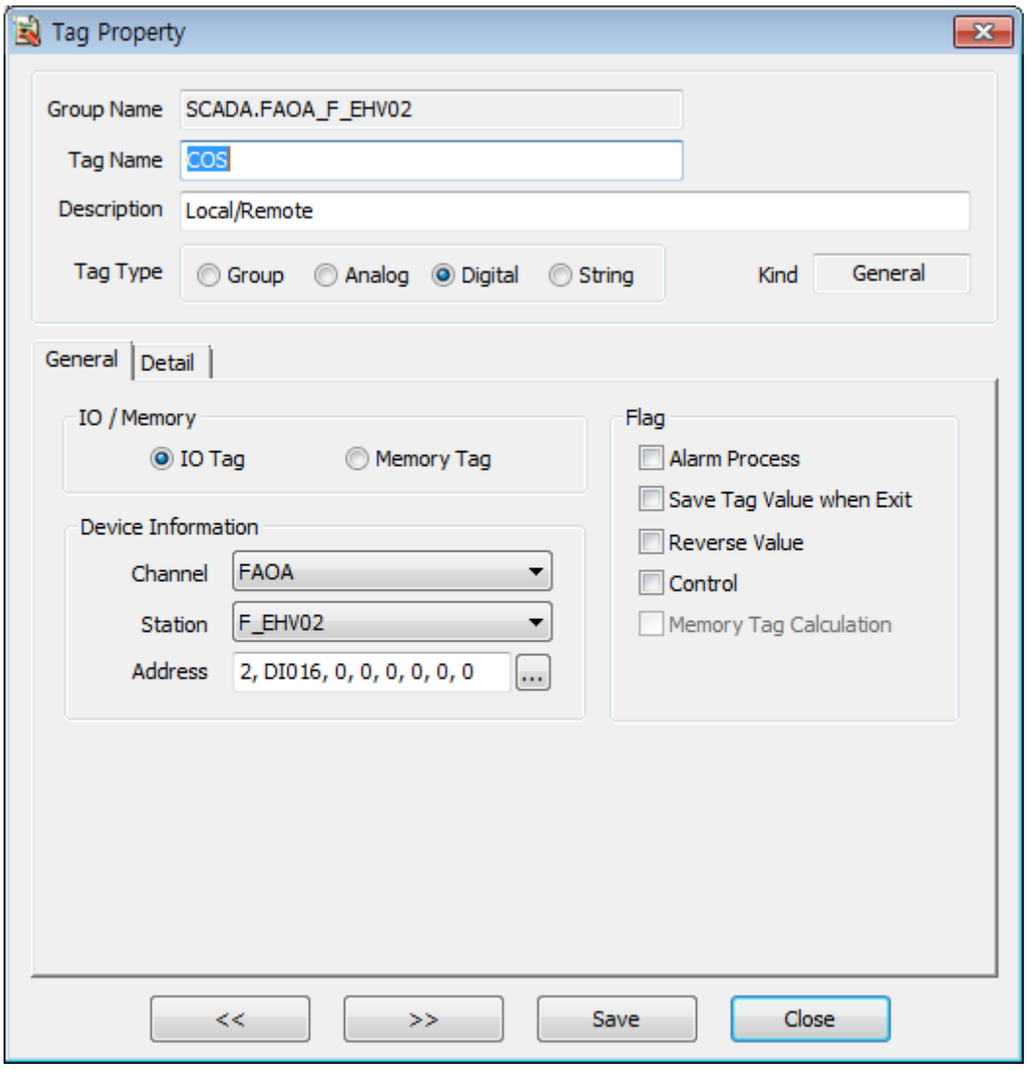
☞ Deleting communication channels means the relevant communication channel with its all stations and tags will be removed.

☞ Deleting stations means the relevant station with its all stations and tags will be removed.

8.3 Tag Property

The tag is the basic unit that has the field or virtual value in the upper HMI system as well as the InfoU. The can be divided into the actual I/O tag connected to the I/O devices; the memory tag created by necessity. All tags can have their own values except group tags and they can be divided into Analog, Digital, String tags depending on the type of values.

The tag has several properties and the definable items are limited depending on each property. In this section, the properties of the tag are classified and the meaning of each property is defined. The figure shown below is an example of the tag property window.



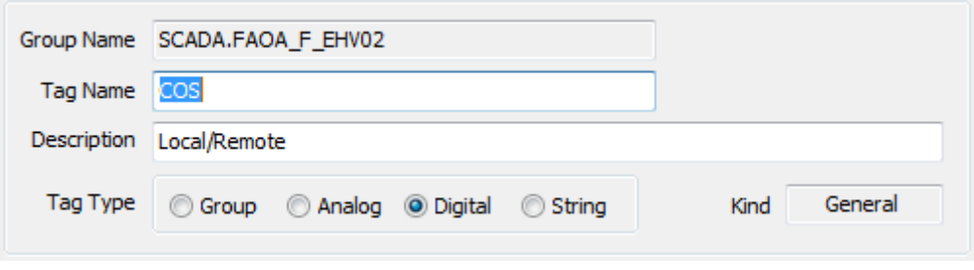
The tag properties are divided into 5; common, general items, details, alarms, operation expressions. In the case of the detailed tab and alarm tab, the definable properties may be different depending on the tag types. As shown in the Fig., unsettable items are deactivated and depending on the tag types and the relevant property group tab is visible and invisible over and over again.

The following is the description on each property group tab.

Properties	Description
Description items	Property that the tags should have commonly
General items	I/O information of I/O and various flags
Details - Analog	The property that is related analog values can be settable when the tag is an analog type.
Details - Digital	The property that is related to digital values can be settable when the tag is a digital type.
Details - String	The property that is related to string values can be settable when the tag is a string type.
Alarm - Analog	The alarm property can be settable when the tag is an analog type.
Alarm - Digital	The alarm property can be settable when the tag is digital type.
Operation expression	Property related to operation expressions

8.3.1 Common Items

They are the most basic property sets of the tag that are commonly used for all types of tags.



(1) Group Name

Properties name	Description
Group name	Name to manage the tag in a logical group

(2) Tag Name

Properties name	Description
Tag name	It is the name of the tag and has the eigen value in the group. The length of the tag name cannot exceed 95 characters of Korean/ 190 characters of English. In this context, the length of the tag name means the full name including the group name. The rules applied to define the tag name are as shown below. ● The length of the name is limited to 95 characters of Korean/ 190 characters of English, ● It must not begin with a number and cannot contain spaces. ● Except some cases, special characters are not available. ● Available characters : # $ [ ] _ { }

(3) Description

Properties name	Description
Description	It is the comment on the tag. The string length is limited to 127 characters of Korean/ 254 characters of English.

(4) Tag Type

Properties name	Description
Tag type	It classifies the type of tag values. They can be divided into Group / Analog / Digital / String depending on the characteristics of values. ● Group : The value does not exist and it is deactivated in the device view so you cannot select it. ● Analog : Select this if you have the analog value. ● Digital : Select this if you have the digital value of 0 or 1. ● String : Select this if you have the string value. If the group tag has the tag as its sub-item, changing into the general tag is not allowed.

(5) Kind

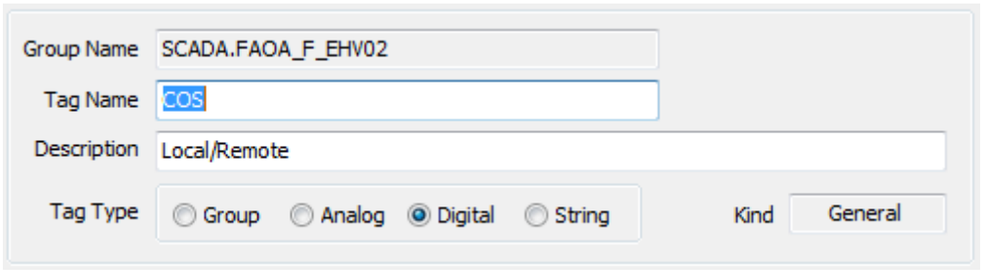
Properties name	Description
Kind	It is the kind of tags. It is read-only property so cannot be changed by a user. ● General : Tags created by a user ● System : Tags to manage the status of the station. They are automatically created when adding stations.

Notice

Among common items, the group name and tag name may work differently when executing the tag properties in the device view and group view.
In the group view, the full name of the tag is divided into the group name and tag name but in the device view, the group name column is empty and the full name of tags will be displayed in the tag name column.

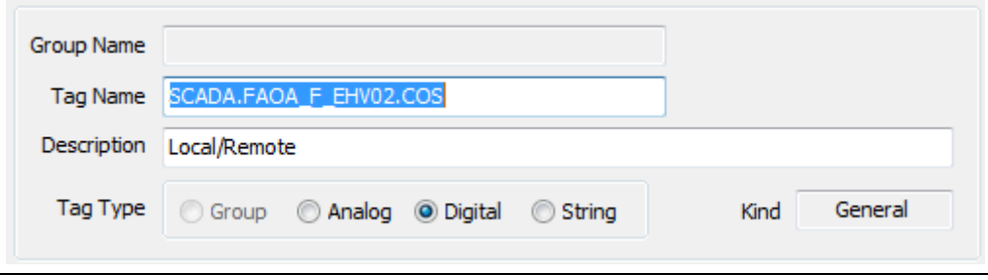
☞ **Name property in the group view**

The full name of the tag is divided into the [group name] and [tag name]. In this case, the group name is read only so it cannot be changed by a user and the [tag name] should be the only one in the relevant group.



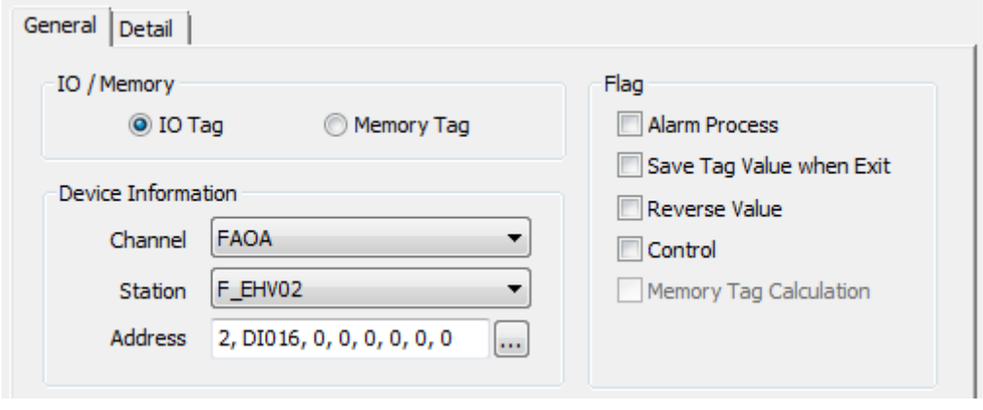
☞ **Name property in the device view**

There is no group in the device view so the [Group Name] item does not exist. Instead, the full tag name combining the [group name] with the tag name is displayed in the [Tag Name]. In practice, the external/internal communication is made with the full tag name and the full name has the eigen value in the project.



8.3.2 General Items

They are the items to set up the tag properties related to I/O and various flags.



(1) Separation

Properties name	Description
Separation	It distinguishes whether the tag is collected from the connected actual devices; or the tag is virtually created for engineering and convenience. ● I/O tag : Select this if the tag is connected to the actual devices. ● Memory tag: Select this if the tag is logically used only.

(2) Device Information

The item is activated only in case of I/O tag. You can set up the communication channel, station and I/O address of the tag.

Properties name	Description
Communication channel	Select the channel for communication of the tag. It can be selected only when the communication channel is registered in the tree items of the tag editor in advance.
Station	It is a subordinate concept belonging to the communication channel and means selecting the actual devices such the PLC, Inverter, etc. It can be selected only when the station is registered in the tree items of the tag editor in advance.
I/O address	Input the communication address of the actual tag of the relevant station. Depending on the types of communication channels, you can also select it from the list by pressing the button. The string length is limited to 127 characters of Korean/ 254 characters of English. The setup method may be different for each device. For more details, refer to the "I/O Driver" manual.

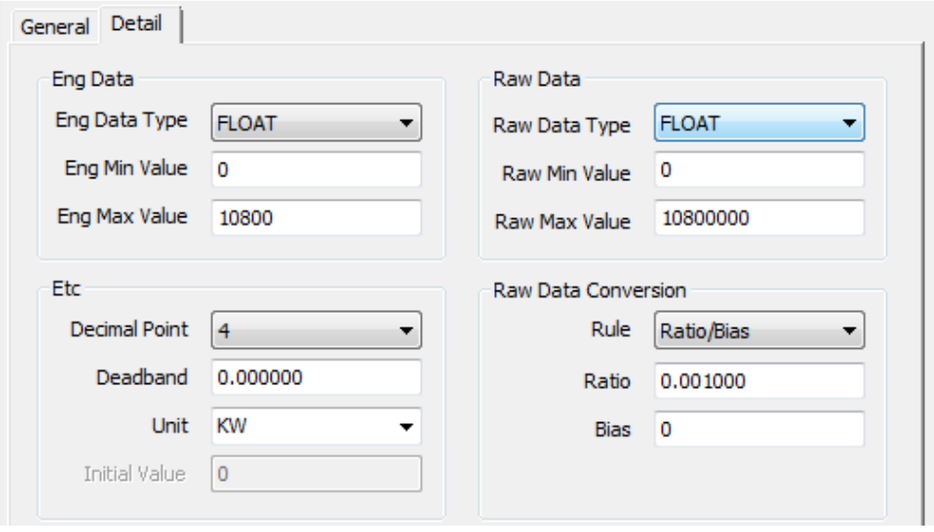
(3) Flag

It defines various settings for tags and operations of tag values during runtime

Properties name	Description
Alarm Process	It sets whether applying alarm to tag values or not. It can be selected only in case of the analog / digital tag. If this property is selected, the property tab is created to set up the detailed alarms.
Save Tag Value When Exit	It determines whether recording the final value of the tag in the database when ending the real-time operation. If this property is activated, the tag value will be saved when the program ends. The saved value will be used as the initial value of the tag when starting the project.
Write Tag Value	It determines whether controlling tags is allowed or not. If this property is canceled, you cannot control the tag even though the control command is given.
Reverse Value	It determines whether reversing the digital tag value obtained from the field. This property can be applied when the tag is I/O and digital type.
Memory Tag Calculation	It determines whether performing the operation expression in case the value is not the group tag but the memory tag. If this property is selected, the operation expression tab that can set up the operation expression will be created.

8.3.3 Details – Analog

When the tag is an analog type, it defines the properties related to the analog type.



(1) Engineering Data

It means the properties related to the data that a user recognize in the upper system.

Property Name	Description
Eng. Data Type	It is the type of the engineering value of the tag in the upper system. There might be a little difference from the pure data type due to the processing of tag values. The types of available values are as follows. ● INT8: Signed 8 bit integer (-128 ~ 127) ● INT16: Signed 16 bit integer (-32.768 ~ 32.767) ● INT32: Signed 32 bit integer (-2,147,438,648 ~ 2,147,483,647) ● UINT8: Unsigned 8 bit integer (0 ~ 255) ● UINT16: Unsigned 16 bit integer (0 ~ 65535) ● UINT32: Unsigned 32 bit integer (0 ~ 4,294,967,295) ● FLOAT: 4 byte real number (-3.4028e+38 ~ 3.4028e+38) ● DOUBLE: 8 byte real number (-1.797693e+308 ~ 1.797693e+308) [NOTE] When using the real number type of the tag in the upper system, it is recommended to apply the DOUBLE type rather than the FLOAT type to increase accuracy of the value and reduce the number of errors.
Eng. Min. Value	It is the minimum engineering value of the tag. If the engineering value of the tag is less than this, the value will be treated as ‘Bad’.
Eng. Max. Value	It is the maximum engineering value of the tag. If the engineering value of the tag is greater than this, the value will be treated as ‘Bad’.

(2) Raw Data

It means the properties related to the tag data used for the equipment.

The property becomes active only in case of I/O tag so it cannot be settable for the memory tag.

Property Name	Description
Raw Data Type	It is the type of the tags collected from the equipment. There might be a little difference from the pure data type due to the processing of tag values. The types of available values are as follows. ● INT8: Signed 8 bit integer (-128 ~ 127) ● INT16: Signed 16 bit integer (-32.768 ~ 32.767) ● INT32: Signed 32 bit integer (-2,147,438,648 ~ 2,147,483,647) ● INT64: Signed 64 bit integer (-9.007.199.254.740.990 ~ 9,007,199,254,740,990) ● UINT8: Unsigned 8 bit integer (0 ~ 255) ● UINT16: Unsigned 16 bit integer (0 ~ 65535) ● UINT32: Unsigned 32 bit integer (0 ~ 4,294,967,295) ● UINT64: Unsigned 64 bit integer (0 ~ 9,007,199,254,740,990) ● FLOAT: 4 byte real number (-3.4028e+38 ~ 3.4028e+38) ● DOUBLE: 8byte real number (-1.797693e+308 ~ 1.797693e+308) ● BCD8: -79 ~ 79 ● BCD16: -7,999 ~ 7,999 ● BCD_32: -79,999,999 ~ 79,999,999 ● BCD_64: -7.999.999.999.999.990 ~ 7.999.999.999.999.990 ● UBCD_8: 0 ~ 99 ● UBCD_16: 0 ~ 9.999 ● UBCD_32: 0 ~ 99.999.999 ● UBCD_64: 0 ~ 9.007.199.254.740.990 ● BIT1: 0 ~ 1 ● BIT2: 0 ~ 3 ● BIT3: 0 ~ 7 ● BIT4: 0 ~ 15
Raw Min. Value	It is the minimum raw value of the tag. If the raw value of the tag is less than this, the value will be treated as ‘Bad’.
Raw Max. Value	It is the maximum raw value of the tag. If the raw value of the tag is greater than this, the value will be treated as ‘Bad’.

(3) Others (Etc.)

They are the other properties related to process and display of tag values.

Property Name	Description
Decimal Point	If the data is [Double] or [Float] type, the decimal places are displayed. When displaying tag values, you can apply this property and display it on the screen.

	Basically, to 4 decimal places are provided and you can set up to 10 decimal places.
Deadband	In case the tag value varies slightly, you can set up the range of deadband to ignore the micro variation of the tag value.
Unit	It indicates the unit of the engineering value.
Initial value	It means the initial value of the tag when starting the project and it is activated only in case of the memory tag. However, when storage options are enabled at the end of the project, please note that the initial value will be ignored and the final value of the saved tag will act as the actual initial value.

(4) Raw Data Conversion

It is the property related to changing the equipment data into the data type that users recognize in the upper system. This property is activated only in case of the I/O tag and it cannot be settable for the memory tag.

Property Name	Description
Data Conversion Rule	It establishes the rules to change the raw value into the engineering value. You can choose one among 4 options; N/A, Ratio/Bias, ratio, ratio/Offset. ● N/A: The raw value is intactly applied as the engineering value without any conversion. ● Ratio/Bias : Conversion method using gradient/intercept. engineering value= (Ratio X raw value) + Bias ● Ratio: Converting the value into ratio using the Min./Max. values of the raw value and engineering value ● Ratio/Offset: Adding Offset to the above ratio value
Ratio	It indicates the value to be multiplied by the raw value. It is activated when the conversion rule is set as [Ratio/Bias].
Bias	It indicates the value to be added to the above value (ration x raw value). It is activated when the conversion rule is set as [Ratio/Bias].
Offset	It indicates the value to be added to the output of ratio conversion. It is activated when the conversion rule is set as [ratio/Offset].

8.3.4 Details – Digital

When the tag is a digital type, it defines the properties related to the digital type.



(1) Data

Property Name	Description
Eng. Data	It is the type of the engineering value of the tag. (BOOL fixed)
Raw Data	It is the type of the raw value of the tag. (BOOL fixed)

(2) String

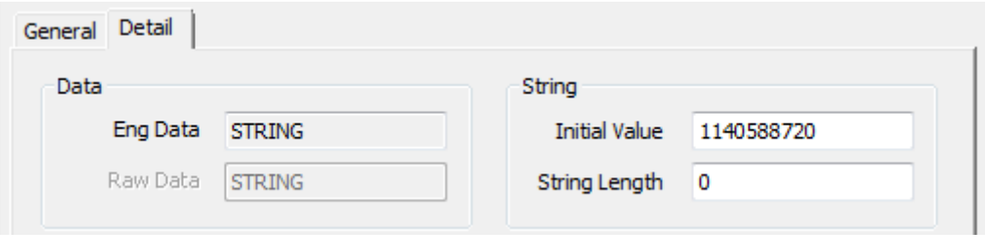
Property Name	Description
ON String	It sets up the sting to be output on the screen when the tag value is 1.
OFF String	It sets up the sting to be output on the screen when the tag value is 0.

(3) Initial Value

Property Name	Description
Initial value	It is activated in the memory tag, not the group tag. To specify the value to use at the beginning stage, you can choose one between OFF(0) and ON(1).

8.3.5 Detail – String

It is the set of properties when the tag is string type.



(1) Data

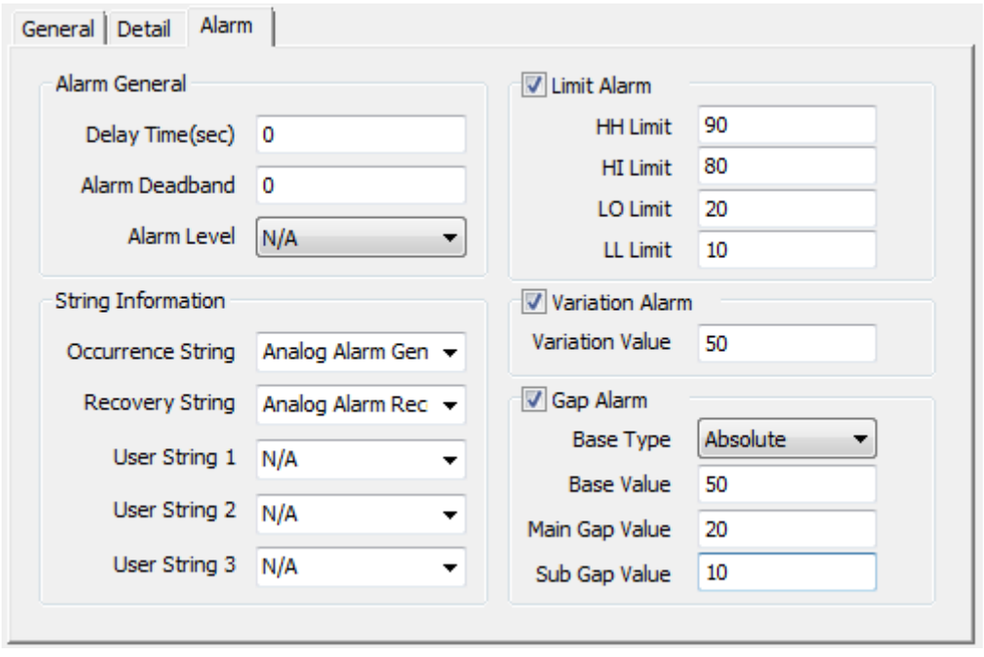
Properties name	Description
Eng. Data	It is the type of the engineering value of the tag. (STRING fixed)
Raw Data	It is the type of the raw value of the tag. (STRING fixed)

(2) String

Properties name	Description
Initial value	It is the type of the engineering value of the tag. (BOOL fixed)
String length	It is the type of the raw value of the tag. (BOOL fixed)

8.3.6 Alarm – Analog

They are the properties to set up alarms when the tag is an analog type.



(1) Alarm-General

The property defines general and common items of alarms.

Properties name	Description
Delay time(sec.)	It is the delay time for alarm occurrence. It means an alarm does not occur immediately; after being putting on hold for the delay time, the alarm is generated by using the value of that point. For more details, refer to the manual on alarm functions.
Alarm Deadband	It means the deadband for an alarm. In case the data value varies slightly in the boundary, you can set up the alarm deadband to prevent the occurrence of unnecessary alarms.
Alarm Level	It can be set from 1 to 10 depending on the severity of alarms.

(2) String Information

It sets up strings related to alarms to display various expressions when an alarm occurs.

Properties name	Description
Occurrence String	It is displayed when an alarm occurs.
Recovery String	It is displayed when an alarm is restored.
User string1	It is defined for a user convenience.
User string2	It is defined for a user convenience.
User string3	It is defined for a user convenience.

(3) Limit Alarm

They are the properties defining the boundary alarm of tags.

Properties name	Description
HH alarm	The HH alarm setting value cannot exceed the max. engineering value and should be greater than the HI value.
HI alarm	The HI alarm setting value should be smaller than the HH setting value and should be greater than the LO setting value.
LO alarm	The LO alarm setting value should be smaller than the HI setting value and should be greater than the LL setting value.
LL alarm	The LL alarm setting value should be smaller than the LO setting value and should be greater than the min. engineering value.

(4) Variation Alarm

It is the property related to variations alarms of the tag value.

Properties name	Description
Variation value	It is the reference value to generate the variation alarm. It compares the current value with the previous one and if the data exceeds the variation value, the variation alarm will occur.

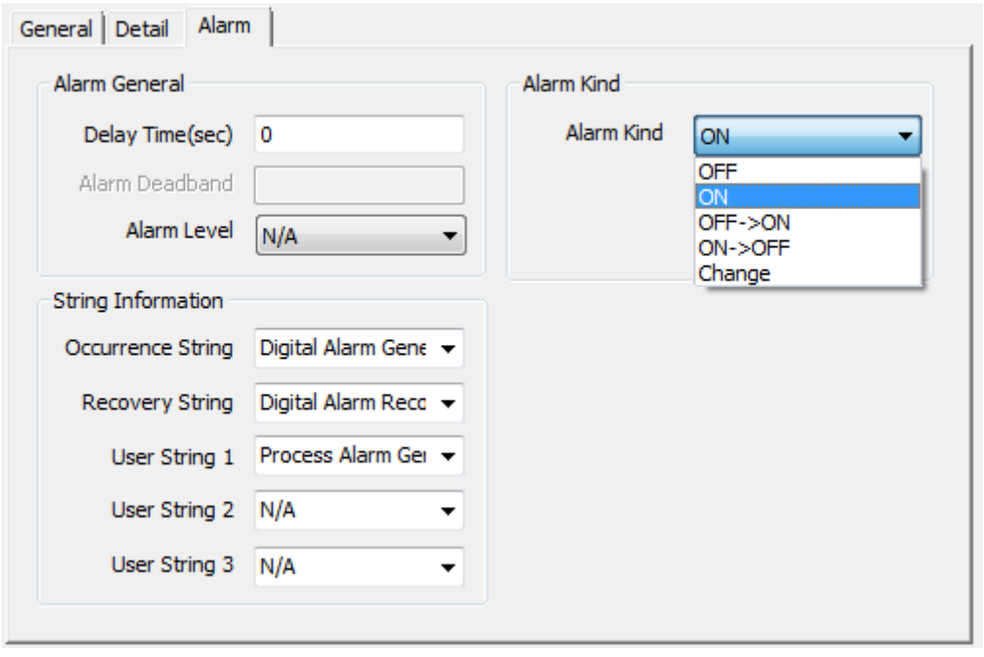
(5) Gap Alarm

They are the properties to set up the separation alarm.

Properties name	Description
Base type	It is the type of the reference value that acts as the baseline to generate the separation alarm. You can choose one between the absolute value and relative value.
Base value	It is the absolute value that acts as the baseline to generate the separation alarm when setting the separation type as the absolute value.
Basic tag name	You can set up the tag name that is the baseline to generate the separation alarm when setting the separation type as the relative value.
Main Gap Value	It means the main separation alarm value.
Sub. Gap Value	It means the secondary separation alarm value.

8.3.7 Alarm – Digital

They are the properties to set up an alarm when the tag is a digital type.

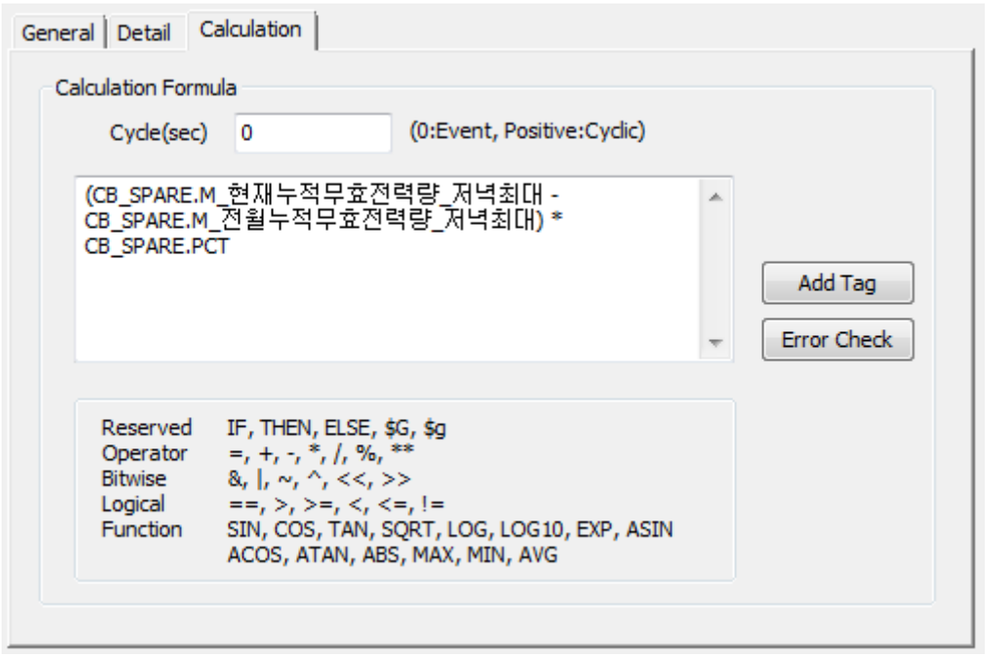


(1) Alarm Kind

Properties name	Description
Alarm Kind	In the case of a digital type, it establishes the rules of alarm occurrence depending on the value. The kinds of alarms are as follows. ● OFF: Occurs when the tag value is OFF. ● ON: Occurs when the tag value is ON. ● OFF→ON: Occurs when the tag value is changed from OFF to ON. ● ON→OFF: Occurs when the tag value is changed from ON to OFF. ● Variation: Occurs when the tag value varies.

8.3.8 Calculations

If you set up the [Memory Tag Operation] flag in the memory tag, the calculation can be settable.



Properties Name	Description
Operating cycle(sec.)	You can input the operating cycle. The default value is 0 and if it is set as 0, the operation expression sets up the context of related tags and works based on the event method. In case it is set as a positive number, each tag independently performs the operation expression in the fixed cycle.
Calculation Formula	You can input the calculation formula.

(1) Fundamental concept

You can calculate the memory tag using the operation expressions such as the four fundamental arithmetic operations, conditional statement, variables, etc. In addition, you can put the formula into the operation expression property of the subject tag that acts as the output of calculation. When starting the project, the data processing engine compiles all operation expressions belonging to the tag and determines each operation set and priorities for calculation.

For example

- The memory tag Tag A is intended to adopt the value of I/O tag Tag B added to 1.
- The memory tag Tag C is intended to adopt the value memory tag Tag A added to I/O tag Tag D.

The above assumption can be expressed as the below formula.

- TagA = TagB + 1
- TagC = TagA + TagD

When the tag DB engineering is applied to the above, you can get the below table. In this case, Tag A and Tag C act as the subject of the operation expression output. You can put the formula into the operation expression property of each tag.

Tag Name	Operating Cycle	Operation Expression	Remarks
TagA	0	TagB + 1	memory tag
TagB	X	X	I/O tag
TagC	0	TagA + TagD	memory tag
TagD	X	X	I/O tag

The above two formulas can be composed of one set with the cause-and-effect relationship. The data processing engine internally establishes the operation sequence of the above set to operate the memory tag. However, in case the tag operation is under cross reference or loop, the operation cannot be performed.

(2) Operation cycle

In the property window, the operation cycle is entered in seconds so you can input the value that is greater than 0. From the viewpoint of the engine performing the operation expression, the cycle can be divided into two types; event method and fixed cycle.

If the cycle is set as “0”, it operates in the event method; if it is set as a positive number, it operates based on the fixed-cycle using a timer.

1) Event method

The operation expression based on the cause-and-effect relationship follows this method, which is derived from the above fundamental concept.

When a user inputs the data manifestly as if he(she) inputs the formula to the mathematical function, the internal engine will determine the order of operation expressions and performs the operations in the manner of artificial intelligence. However, as mentioned above, you need to be careful of the cross reference or loop.

2) Fixed-cycle method (sec.)

Unlike the even method, the operation expression belonging to the relevant cycle is performed. There is no correlation and in the case of operation expressions with the same cycle, the order of execution cannot be determined. It is used when you refer to other tag values in a fixed cycle.

(3) Syntax and usage

A user needs to input the operation expression to the property window of the tag and perform the error checking by pressing the [Error Check] button to confirm whether the operation expression is normal or not.

The available syntax for the operation expression is as below.

Classification	Keywords
Reserved word	IF, THEN, ELSE
Operator	= , +, -, *, /, %, **
Bit	&, |, ~, ^, <<, >>
Function	SIN, COS, TAN, SQRT, LOG, LOG10, EXP, ASIN, ACOS, ATAN, ABS, MAX, MIN, AVG
Separator	(,)

Notice

☞ The reserved word and function name should be in upper case.

☞ IF~THEN~ELSE statement should always be used together. (cannot be omitted)

[Example of using syntax]

Operation Expression	Description
IF (A + B > 10) THEN 4 ELSE 5	If the value of tag A and tag B is greater than 10, 4 is returned; if the value is smaller than 10, 5 is returned.
ABS(A)	The absolute value of A is returned.
IF (A & B) THEN C ELSE D	If A and B are True, the value of C is returned; if they are False, the value of D is returned.

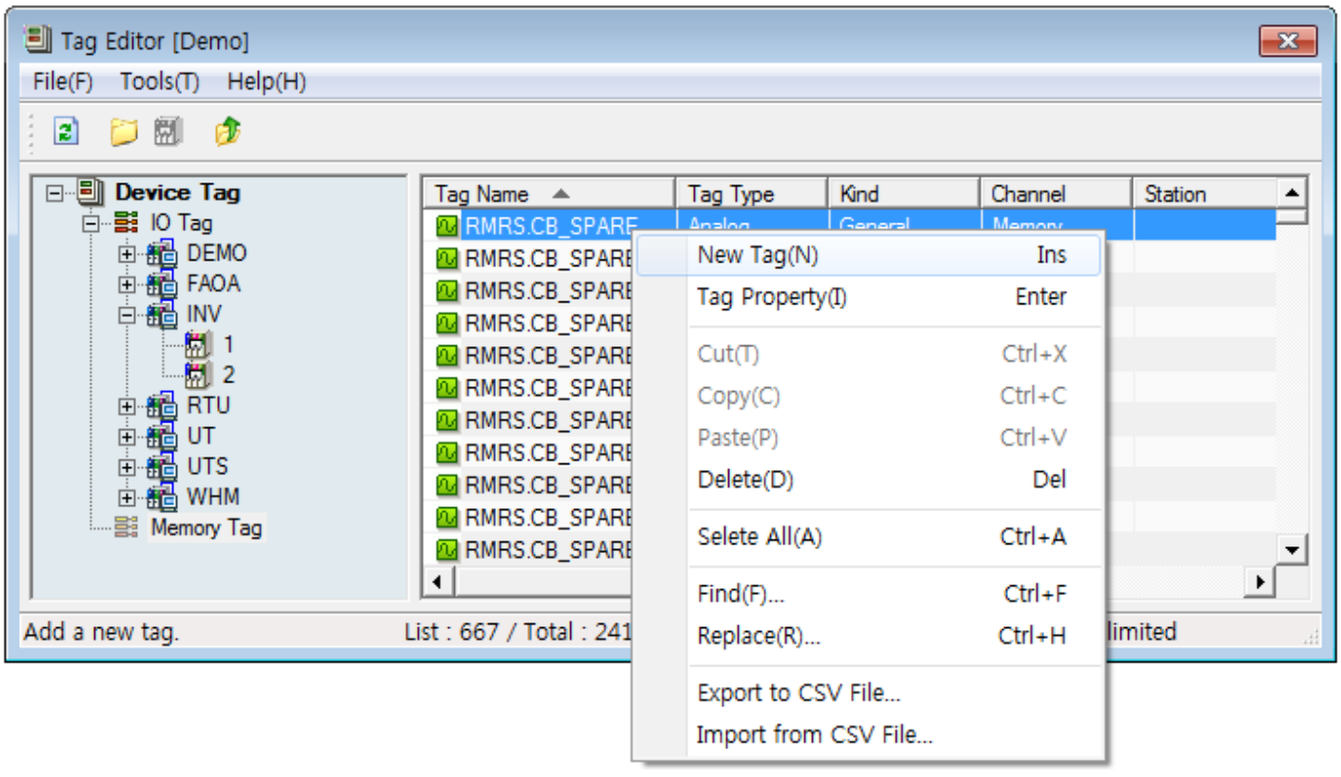
[Actual examples]

Tag	Operation expression
Current active electric power by 15minutes	Electricity No 01.Just 15 minutes ago_Max. current electric power + electricity No 02.Just 15 minutes ago_Max. current electric power + electricity No 01.Just 15 minutes ago_Max. current electric power
Current wattage 2	WDC.current electric power * meter constant.PCT
Current wattage	WDC.current electric power * meter constant.PCT+ meter constant.correction value
Current electric power_ratio	(100 * MEM.current wattage) / 150000
Current electiric power	IF ((WDC.demand time < 10) || (WDC.demand time > 895)) THEN (MEM. average current power) ELSE (WDC.current electric power* meter constant.PCT*900+meter constant.correction value) / (WDC.demand time) Wattage = MEM.current electric power / 4
Pulse integration	WDC.current electric power*5000/4000
Predicted electric power	IF(WDC.demand time < 60) THEN (MEM.current electric power) ELSE (WDC.predicted electric power * meter constant.PCT+ meter constant.correction value)
Load electric power	IF ((WDC.demand time < 61) || (MEM.predicted electric power <= MEM.current wattage)) THEN (MEM.current electric power) ELSE ((MEM.predicted electric power - MEM.current wattage) * 900 / MEM.demand time_remaining time)
Base electric power	WDC.Base electric power * meter constant.PCT
PCT	(Meter constant.PT * meter constant. CT)/(meter constant.PT_2 * meter constant.CT_2)

8.4 Tag Edition

8.4.1 Addition of Tags

If you double-click the empty list on the tag list screen or select [New Tag] in the popup menu, the property window where you can add new tags will show up.



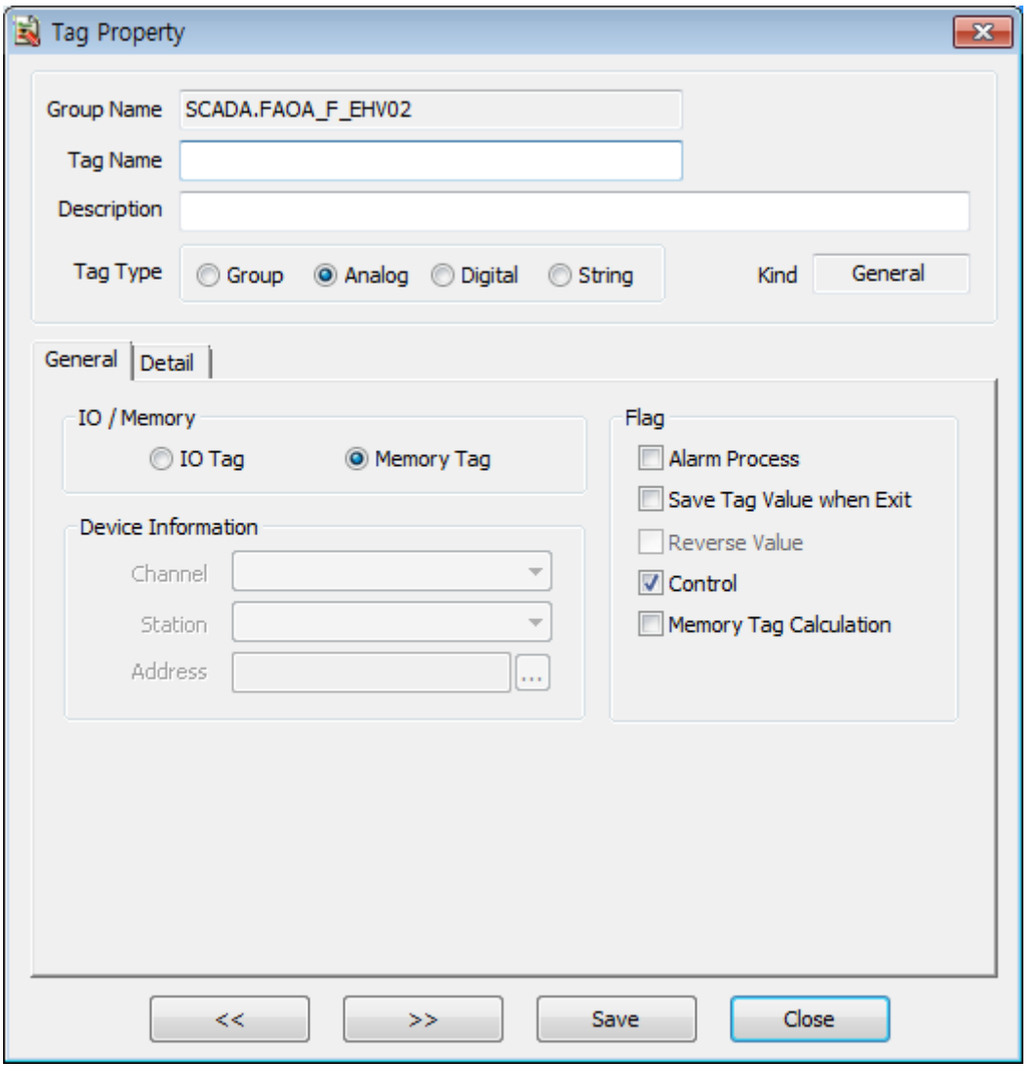
The basic items of the tag property window may be a little different depending on which part of the tag editor calls it up.

(1) Addition of tags in the device view

If you run [New Tag] on condition that the specific station is selected in the device view, the tag property window where the relevant communication channels and stations are basically entered will show up. Of course, although you change the communication channels and stations, tags can be registered.

In the device view, there is no logical tag group so the group name is deactivated and you cannot also select the group tag type. In this case, you need to input the full tag name and if there is a DOT(.) in the middle of the tag name, the parent group of the relevant tag will be automatically created.

Ex.) If the tag name of “HELLO.KOO” is entered and saved, the group name called “HELLO” will be internally searched and when the group named “HELLO” exists, “KOO” will be added as a sub-tag. In case the group named “HELLO does not exist, after adding the “HELLO” group and save it, add the tag called “KOO” as its sub-tag.



(2) Addition of tags in the group view

If you run the [New Tag] command on condition that the specific group is selected in the group view, the tag property window where the group is basically entered will show up. In this case, you cannot change the group name. Namely, in the group view, you can just add the child tag only under the its parent group.

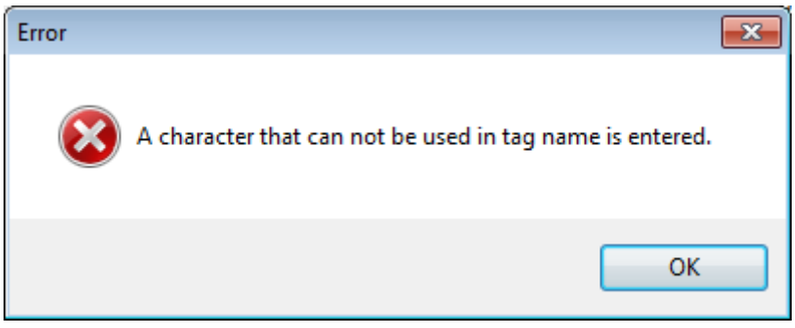
When the tag is added at the top level (route) of the group view, the group name will be deactivated.

8.4.2 Storage of Tags

If you double-click the tag or select [Tag Property] in the popup menu on the right side of the tag, the property window will show up. When you click the [Save] button after editing tag properties, the information will be saved; when you click the [Close] button, the tag property window will be closed without saving the tag.

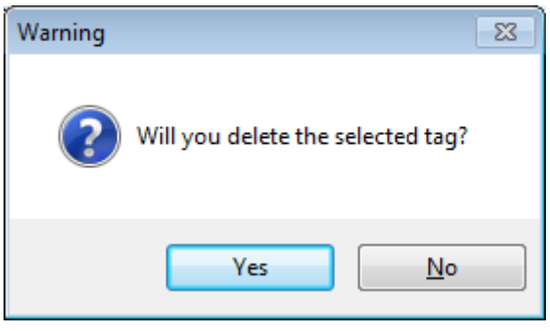
If you modify the tag name and save it in the tag property window, the tag with a new name will be added instead of tag modification.

During saving the tag, perform the integrity check for the tag properties internally and save the data only when the result turns out to be normal. (For the integrity rules of tags, refer to the tag properties)
When the error occurs during the tag test, the below error message will shows up.

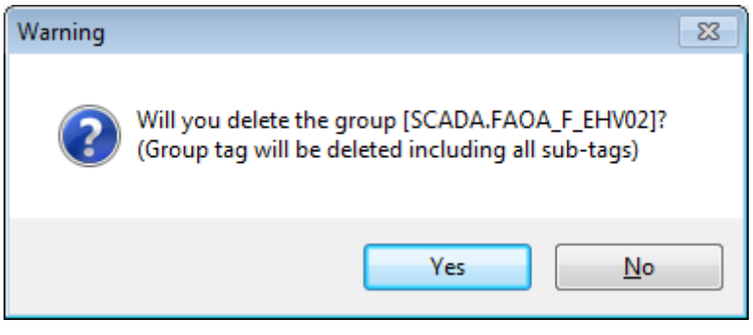


8.4.3 Deletion of Tags

After selecting the tag from the tag list, if you press the [Delete] button, the dialog box asking whether deleting the tag will show up. If you press the [Yes] button, the tag will be deleted



The deletion of tags can be divided into the individual deletion that selects and removes one tag; multiple deletion that selects and removes several tags.
In addition, when you delete tags including the group tags, all sub-tags of the groups will be deleted. When the group tag is included in the tag to delete, the below message will show up.



Notice

Please note that the deleted tags cannot be restored so you must be careful to delete tags.

8.5 CSV Tag I/O

For mass tags works, Not only tag edition using a clipboard but also tag engineering using the CSV file is available. The CSV file means the file whose each property is separated by a comma (,).

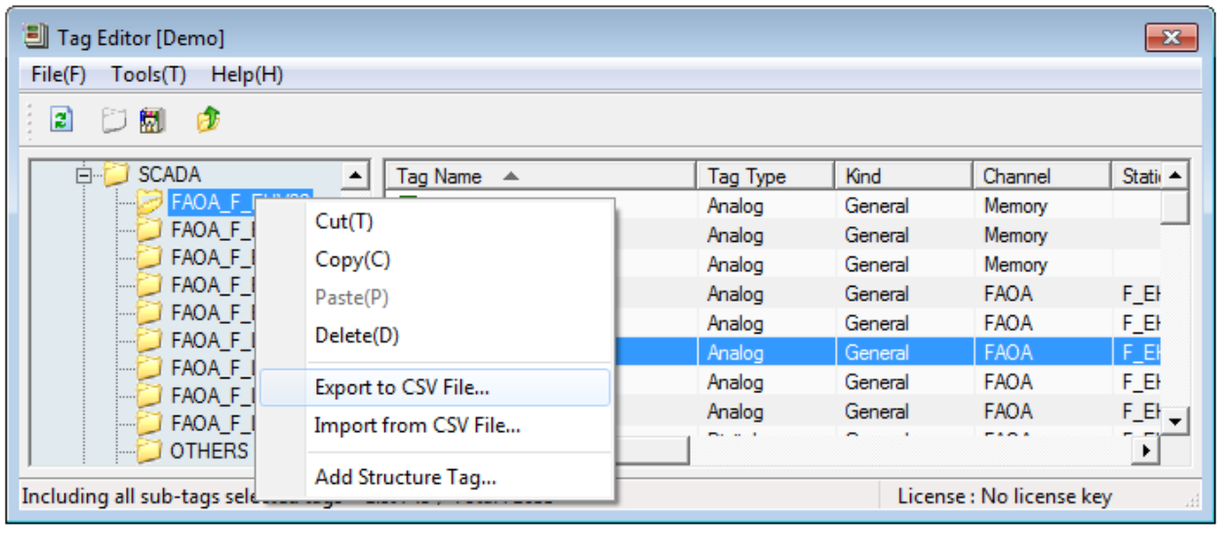
Notice

It cannot be compatible with the CSV file made in the existing InfoU engineering module.
(Cannot be compatible with the earlier versions than 1.9.0)

8.5.1 Export to CSV File

(1) Execution in the tag tree

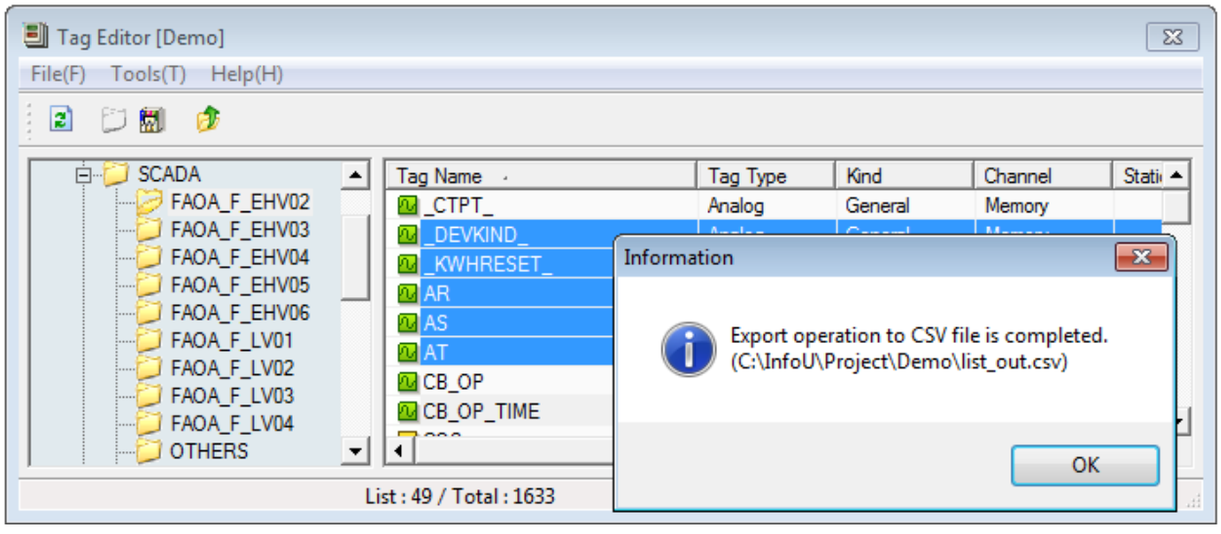
If you select the tag tree item, the tags included in the selected tag tree will be displayed in the tag list. A user can export the tag data shown in the tag list to the CSV file using the [Export to CSV file] function in the popup menu of the tag tree.



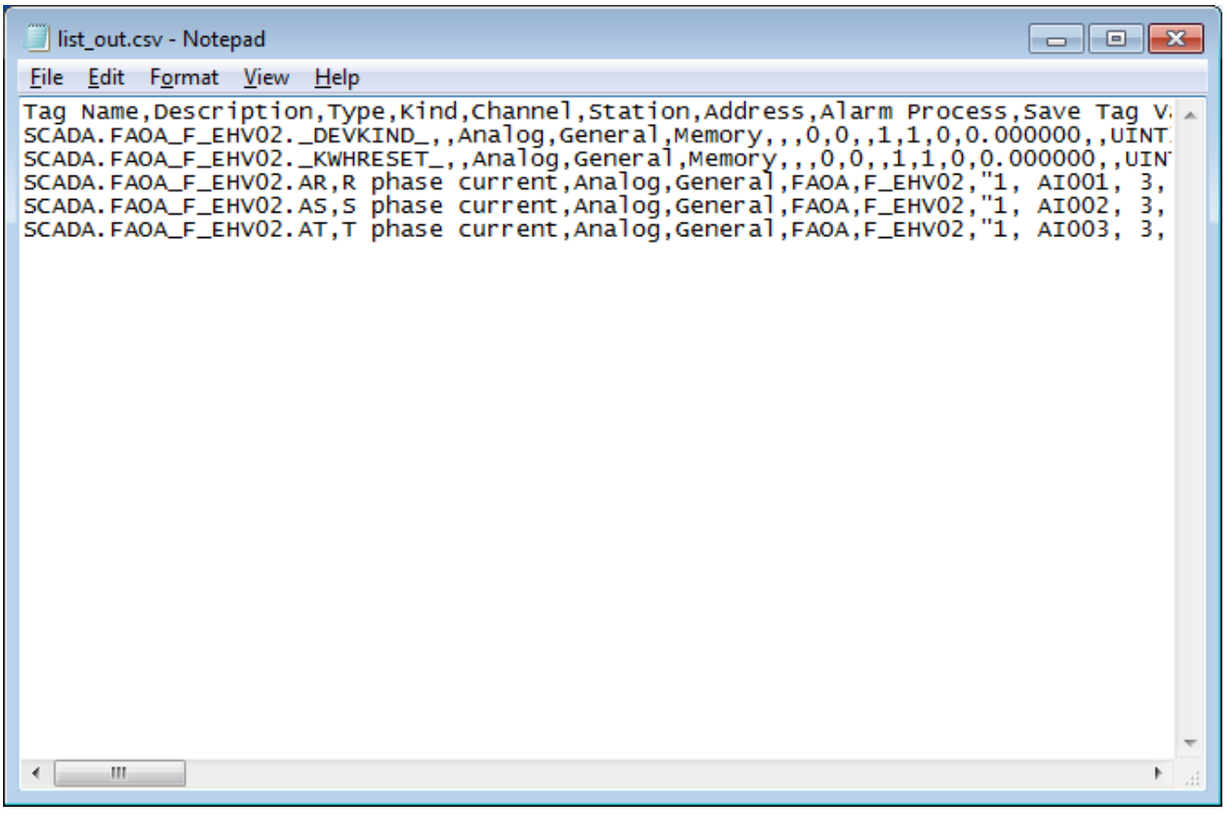
However, the tag data exported from the tree menu is the tags included in the tree item. If you execute [Export to CSV file] in a specific communication channel, the tags only included in the channel will be exported.

(2) Execution in the tag list

After selecting the tags to export in the tag list, if you select [Export to CSV file] in the popup menu, the selected tags only will be exported to the CSV file.



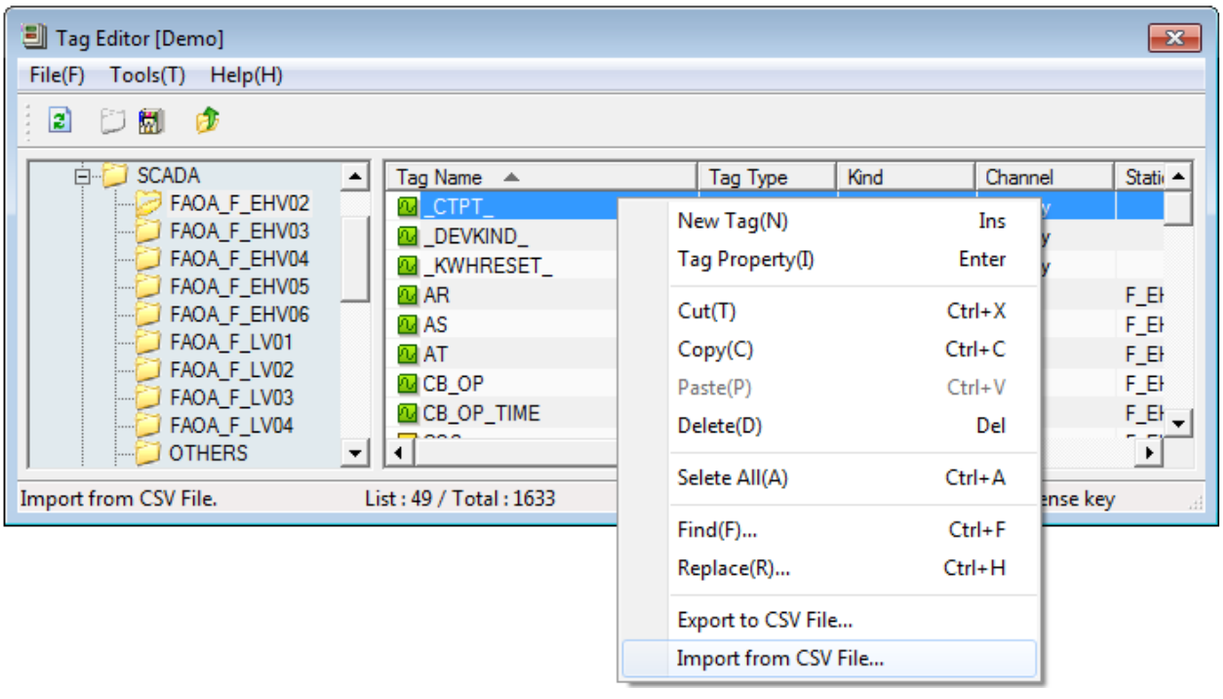
After executing ‘Export’, the below CSV file will be created.



8.5.2 Import from CSV File

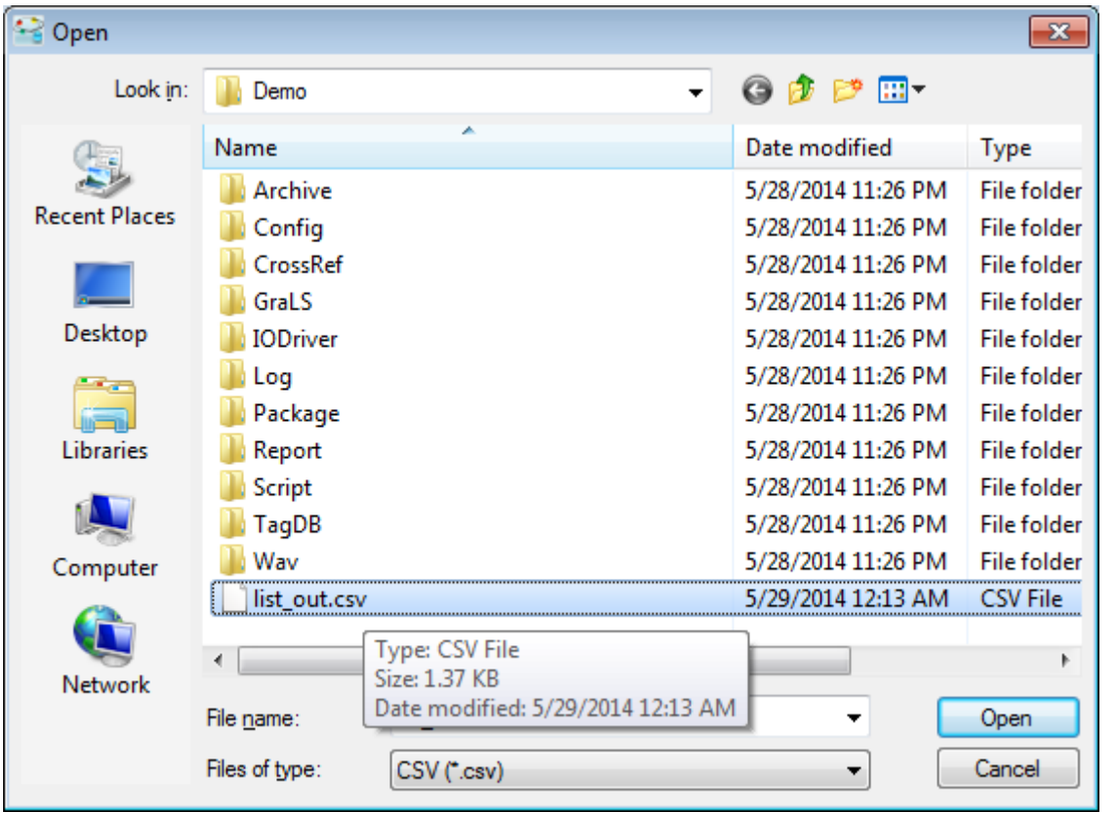
In the previous step, you can import the CSV file to the tag editor again through the [Import from CSV file] menu. However, the CSV file works of tags may internally cause I/O related to saving many tags so it is recommended to split the working unit into a small scale

(1) Running the menu



Unlike 'Export', the [Import from CSV file] menu imports the whole contents of the CSV file regardless of the tree or list, etc.

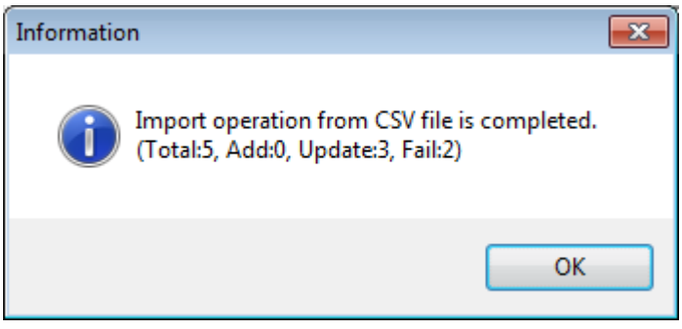
If you run the menu, the screen where you can select the CSV file will show up as below.



After selecting the CSV file to import to the tag editor, when you click the [Open] button, the details of the [Open] button will be imported. If there is the same tag name, the tag property will be changed and in the case of the new tag name, the tag will be added.

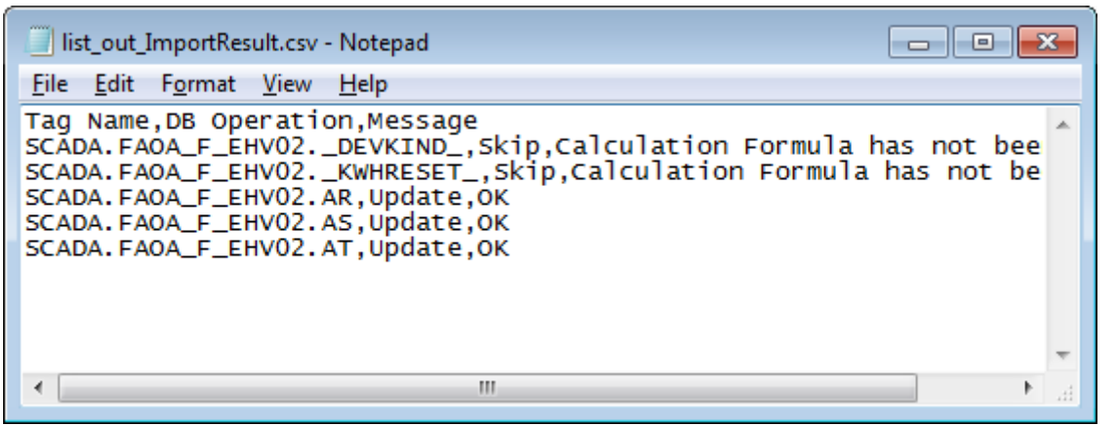
(2) Confirmation of the output

After performing the [Import from CSV file] menu, the message box showing a user the output will be displayed.



The output file for the works named XXX_ImportResult.csv will be created in the folder where the CSV file is located. The output file has three items by tags ; tag name, DB works, message and the meaning of each item is as below.

- Tag name : Full tag name
- DB works : Output of DB works
- Message : Success / Error message



However, in case the value of the item is omitted, the default value will be input so you must be careful of this.

8.5.3 Tag Items of the CSV File

Addition and modification of tags through the tag property window is the way to minimize errors of tag input such as automatic errors caused by the user interface, Enable/Unable by tag classification or input limit through list selection, etc. However, in the case of tag input through the CSV file, the properties of all tags should be written with the data such as text, number, etc. and the error of data integrity may occur so your attention is required.

Each item of tag properties saved as the CSV file is as below. The first line of the CSV file is recognized as the

column name and it is not saved as data. In addition, as shown in the below table, in the case of the CSV file's data, the number of 50 columns should match.

[Tag properties of the CSV file]

Order	Column name	Input details	Remarks
1	Tag name	Tag's full name	
2	Comment	Tag comment	If there is a comma in the text, put quotation marks "".
3	Type	Group / Analog / Digital / String	-
4	Sort	General / system	-
5	Communication channel	communication channel name	-
6	Station	station name	-
7	I/O address	I/O address	-
8	Alarm process	0 / 1	flag items - 0 : Unable, 1 : Enabled
9	Save at the end	0 / 1	flag items - 0 : Unable, 1 : Enabled
10	Reverse of the actual data	0 / 1	flag items - 0 : Unable, 1 : Enabled
11	Write tag value	0 / 1	flag items - 0 : Unable, 1 : Enabled
12	Memory tag operation	0 / 1	flag items - 0 : Unable, 1 : Enabled
13	Decimal places	Decimal places	-
14	Value Deadband	Value Deadband	-
15	Unit	Unit	If there is a comma in the text, put quotation marks "".
16	Data type	BOOL INT8 / INT16 / INT32 / UINT8 / UINT16 / UINT32 / FLOAT / DOUBLE STRING	-
17	Raw value type	BOOL INT8 / INT16 / INT32 / INT64 / UINT8 / UINT16 / UINT32 / UINT64 FLOAT / DOUBLE BCD8 / BCD16 / BCD32 / BCD64 UBCD8 / UBCD16 / UBCD32 / UBCD64 BIT1 / BIT2 / BIT3 / BIT4 STRING	-
18	Min. tag value	Min. tag value	-

19	Max. tag value	Max. tag value	-
20	Min. raw value	Min. raw value	-
21	Max. raw value	Max. raw value	-
22	Initial value	Initial value	If the tag type is string and there is a comma in the text, put quotation marks “”.
23	Conversion rules	0 / 1 / 2 / 3	0 : N/A 1 : Ratio/Bias 2 : ratio 3 : ratio/Offset
24	Ratio	Ratio	-
25	Bias	Bias	-
26	Offset	Offset	-
27	ON string	ON string	If there is a comma in the text, put quotation marks “”.
28	OFF string	OFF string	If there is a comma in the text, put quotation marks “”.
29	String length	String length	-
30	Alarm delay time	Alarm delay time	-
31	Alarm Deadband	Alarm Deadband	-
32	Alarm level	Alarm level	-
33	Occurred string	Occurred string	If there is a comma in the text, put quotation marks “”.
34	Recovery string	Recovery string	If there is a comma in the text, put quotation marks “”.
35	User-defined1	User-defined1	If there is a comma in the text, put quotation marks “”.
36	User-defined2	User-defined2	If there is a comma in the text, put quotation marks “”.
37	User-defined3	User-defined3	If there is a comma in the text, put quotation marks “”.
38	HH boundary value	HH boundary value	-
39	HI boundary value	HI boundary value	-
40	LO boundary value	LO boundary value	-
41	LL boundary value	LL boundary value	-
42	Variation value	Variation value	-
43	Type of separation reference value	0 / 1 / 2	0 : N/A 1 : absolute value 2 : relative value
44	Reference value	Reference value	-

45	Basic tag	Basic tag	-
46	Main separation value	Main separation	-
47	Secondary separation value	Secondary separation	-
48	Digital alarm types	1 / 2 / 3 / 4 / 5	1 : OFF 2 : ON 3 : ON → OFF 4 : OFF → ON 5 : CHANGE
49	Memory operating cycle	memory operating cycle	-
50	Memory operation expression	memory operation expression	In case there is the line-break character in the operation expression string, change it into the symbol ` (Grave key).

8.6 Copy/Paste/Cut of Tags

The InfoU tag editor supports the 'Copy' and 'Paste' functions besides the link with the CSV file that provides the function to edit mass tags. The copied tags are saved to a clipboard so you can paste them to the tag editor or Excel, notepad, etc. In addition, you can also copy the tag data from Excel and import it to the tag editor.

However, the 'Copy' and 'Paste' functions of tags are provided by the group view only and they are not available in the device view. Moreover, these functions may internally cause I/O related to saving many tags so it is recommended to split the working unit into a small scale.

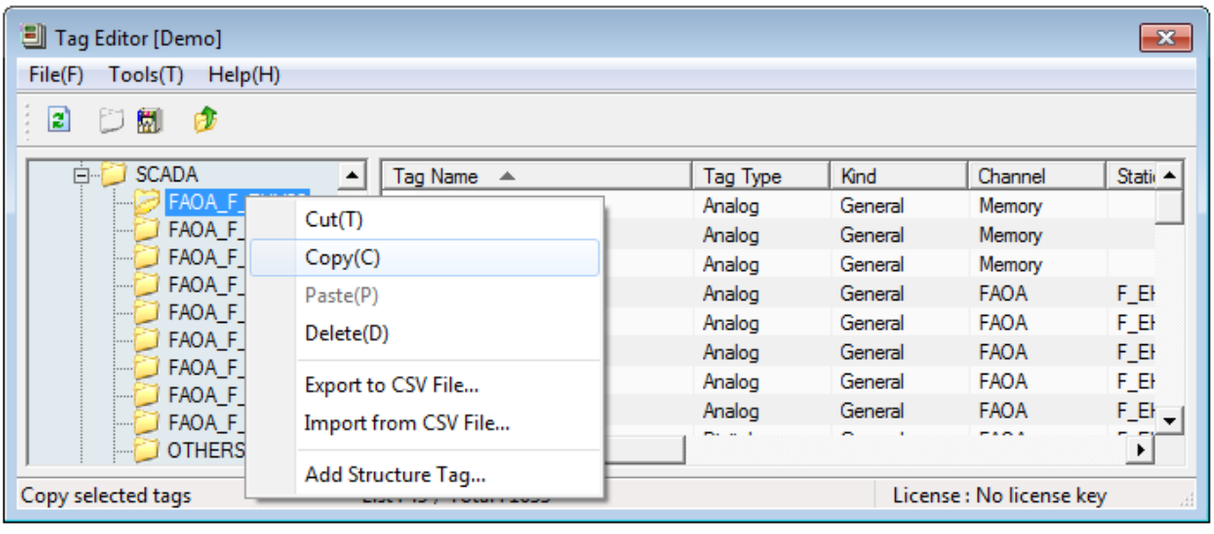
The data copied from the tag editor has basically the same structure and data as the output file exported to the CSV file. However, the data has the following characteristics.

[Restrictions and characteristics of clipboard tag data]

- It does not include the column name.
- The tag name is not the full name but the tag name excluding the parent groups.
- 'Paste' can be applied only when the data is composed of more than 3 columns of the tag name, comment, type.
- Each item is separated by the tab symbol. If the tab symbol is included in a certain property text of the tag, it does not work normally.

8.6.1 Copy

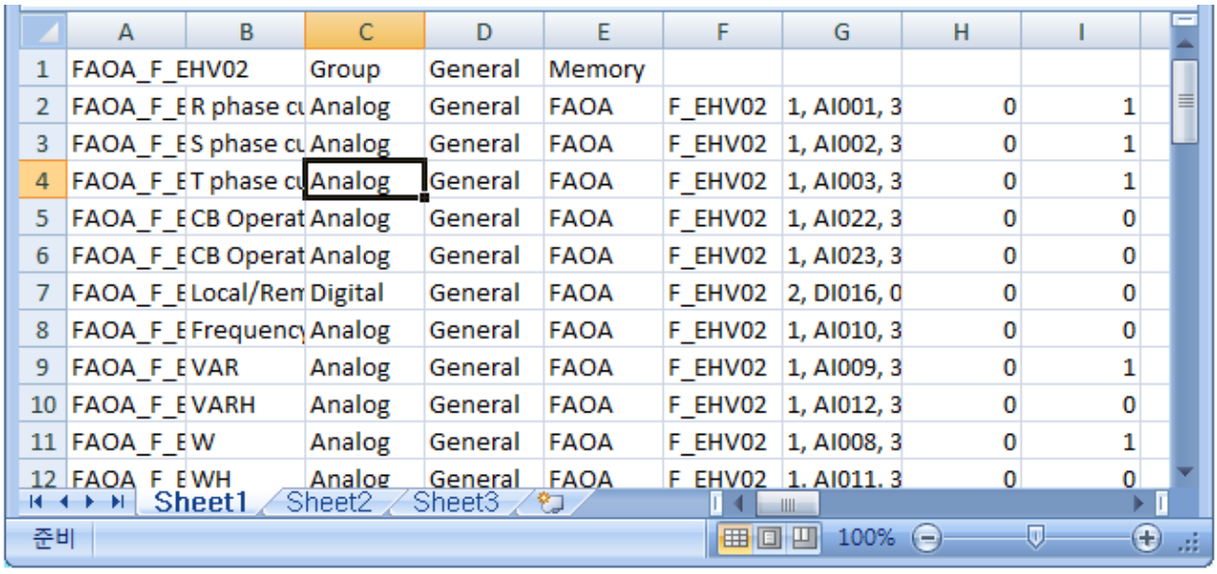
In the group view, you can select [Copy] in the popup menu of the tag tree or tag list.



If the copied tag is a group tag, the group tag and all sub-tags belonging to the group are copied together.

When copying tags, in terms of the tag name, please note that the name will be copied from the name of the selected position is copied, not the full tag name. The copied tag data can be freely used in the programs such as a notepad or Excel, etc.

When pasting the data copied from the tag editor to Excel, you can see the below screen.

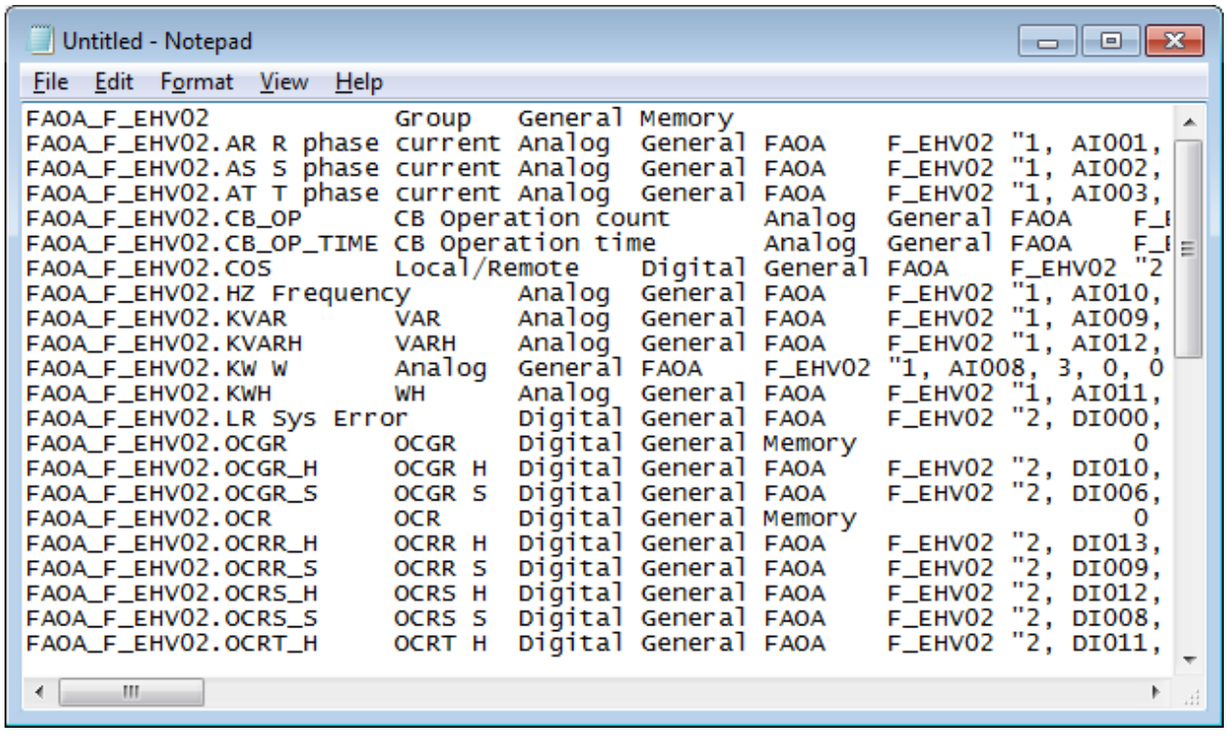


	A	B	C	D	E	F	G	H	I
1	FAOA_F_EHV02		Group	General	Memory				
2	FAOA_F_E	R phase cu	Analog	General	FAOA	F_EHV02	1, AI001, 3	0	1
3	FAOA_F_E	S phase cu	Analog	General	FAOA	F_EHV02	1, AI002, 3	0	1
4	FAOA_F_E	T phase cu	Analog	General	FAOA	F_EHV02	1, AI003, 3	0	1
5	FAOA_F_E	CB Operat	Analog	General	FAOA	F_EHV02	1, AI022, 3	0	0
6	FAOA_F_E	CB Operat	Analog	General	FAOA	F_EHV02	1, AI023, 3	0	0
7	FAOA_F_E	Local/Rem	Digital	General	FAOA	F_EHV02	2, DI016, 0	0	0
8	FAOA_F_E	Frequency	Analog	General	FAOA	F_EHV02	1, AI010, 3	0	0
9	FAOA_F_E	VAR	Analog	General	FAOA	F_EHV02	1, AI009, 3	0	1
10	FAOA_F_E	VARH	Analog	General	FAOA	F_EHV02	1, AI012, 3	0	0
11	FAOA_F_E	W	Analog	General	FAOA	F_EHV02	1, AI008, 3	0	1
12	FAOA_F_E	WH	Analog	General	FAOA	F_EHV02	1, AI011, 3	0	0

Sheet1 Sheet2 Sheet3

준비 100%

When pasting the data copied from the tag editor to a notepad, you can see the below screen.



Untitled - Notepad

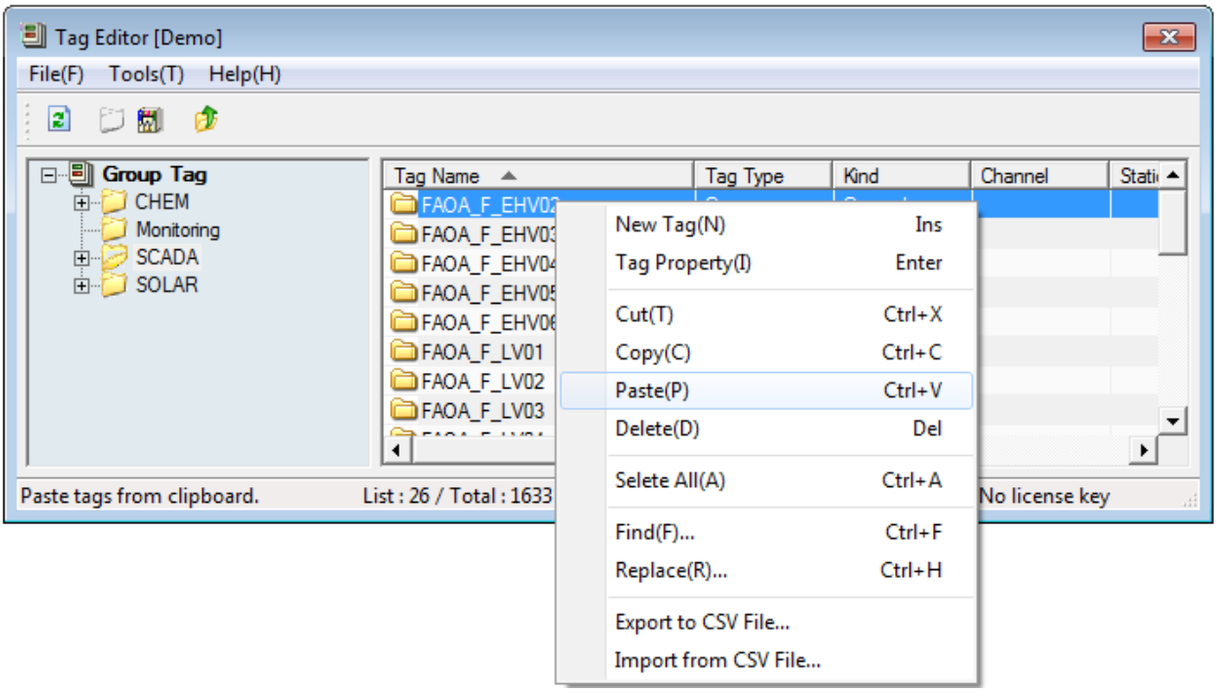
File Edit Format View Help

```
FAOA_F_EHV02              Group    General Memory
FAOA_F_EHV02.AR R phase current Analog  General FAOA    F_EHV02 "1, AI001,
FAOA_F_EHV02.AS S phase current Analog  General FAOA    F_EHV02 "1, AI002,
FAOA_F_EHV02.AT T phase current Analog  General FAOA    F_EHV02 "1, AI003,
FAOA_F_EHV02.CB_OP       CB Operation count       Analog  General FAOA    F_E
FAOA_F_EHV02.CB_OP_TIME CB Operation time        Analog  General FAOA    F_E
FAOA_F_EHV02.COS         Local/Remote    Digital General FAOA    F_EHV02 "2
FAOA_F_EHV02.HZ Frequency        Analog  General FAOA    F_EHV02 "1, AI010,
FAOA_F_EHV02.KVAR        VAR     Analog  General FAOA    F_EHV02 "1, AI009,
FAOA_F_EHV02.KVARH       VARH    Analog  General FAOA    F_EHV02 "1, AI012,
FAOA_F_EHV02.KW W        Analog  General FAOA    F_EHV02 "1, AI008, 3, 0, 0
FAOA_F_EHV02.KWH         WH      Analog  General FAOA    F_EHV02 "1, AI011,
FAOA_F_EHV02.LR Sys Error        Digital General FAOA    F_EHV02 "2, DI000,
FAOA_F_EHV02.OCGR        OCGR    Digital General Memory                  0
FAOA_F_EHV02.OCGR_H      OCGR H  Digital General FAOA    F_EHV02 "2, DI010,
FAOA_F_EHV02.OCGR_S      OCGR S  Digital General FAOA    F_EHV02 "2, DI006,
FAOA_F_EHV02.OCR         OCR     Digital General Memory                  0
FAOA_F_EHV02.OCRR_H      OCRR H  Digital General FAOA    F_EHV02 "2, DI013,
FAOA_F_EHV02.OCRR_S      OCRR S  Digital General FAOA    F_EHV02 "2, DI009,
FAOA_F_EHV02.OCRS_H      OCRS H  Digital General FAOA    F_EHV02 "2, DI012,
FAOA_F_EHV02.OCRS_S      OCRS S  Digital General FAOA    F_EHV02 "2, DI008,
FAOA_F_EHV02.OCRT_H      OCRT H  Digital General FAOA    F_EHV02 "2, DI011,
```

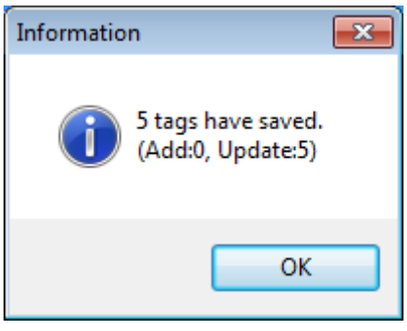
8.6.2 Paste

The [Paste] menu is enabled after copying tags.

If you execute the [Paste] menu in the tree or list items of the tag to paste, the previously copied tag will be added. In case there is the same tag name, the tag property will be changed into new contents.



If you execute ‘Paste’ and save the data, the following message will be displayed.



The CSV file is executed with the full tag name so the same output is obtained even if it is performed anywhere of the tag editor, however, in the case of ‘Paste’, the relative path of the tag is used so the tag is saved depending on the position where the tag is pasted.

8.6.3 Cut

The [Cut] menu is the function to move the tag to other groups.

When executing [Cut], please note that the relevant tag will be removed from the tag database and the deleted data will be kept in a clipboard. After that, a user can paste the tag by using [Paste].

Notice

Please note that the tag will be deleted from the database right after executing [Cut]. At the same time, the information of the deleted tag will be kept in a clipboard.

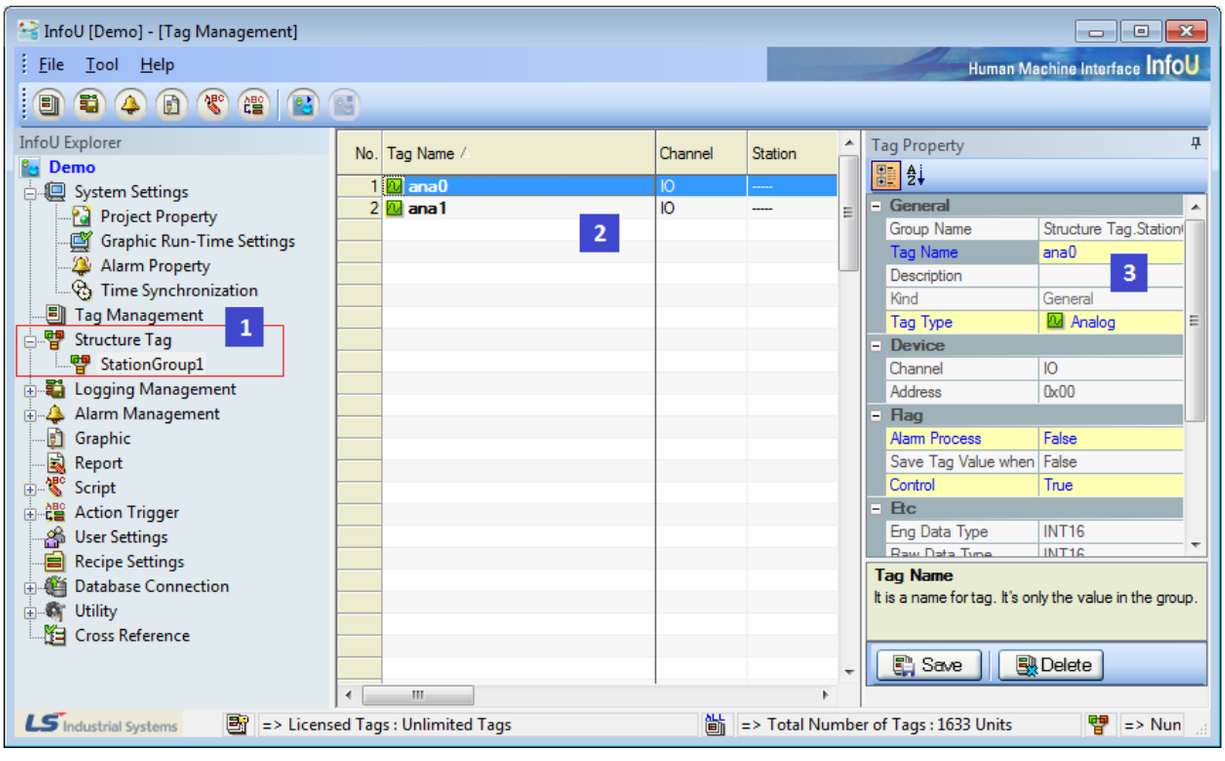
After that, if you recopy the other object or text in Windows without executing [Paste], the information of the previously [Cut] tag will be deleted.

8.7 Sturcture Tag

It is the function to prepare frequently used tag groups and add tags by the group in order to make tags more conveniently. This function can be mainly used to insert the tag sets expressing the physical devices or logical groups to the tag DB repetitively. If you can set the specific tags and register them in the structure tag template, the actual tags will be created by the template and can be added to the tag DB.

The definition of structure tags is performed by the InfoU engineering module and the instantiation process of structure tags is performed by the tag editor.

8.7.1 Main Screen and Explanation of Terms



You can see the registered structure tag templates and the tag's property window through [Structure Tag] in the Tool Search of the InfoU engineering.

No.	Item	Description
1	Structure tag template	Registered structure tag template list
2	Structure tag items	Items of the selected structure tag template
3	Item properties	Properties of the selected item

The description on each item is as shown below.

(1) Structure tag template

When the components of a specific equipment or unit device are the same, the structure tag template means making them into the structure tag template and specifying a framework.

For example, in the case of the scale device A, it is composed of the power supply part, measuring device,

display device, etc. To register 100 scale devices in the HMI system, you need to register 100 groups and tags belonging to each group, etc.
In this case, if you specify the structure tag template named A, you can proceed with tag works easily.

(2) Structure tag item
It means each component belonging to one structure tag template. In the above case of the scale device A, you can specify each item such as power supply part, measuring device, display device, etc.

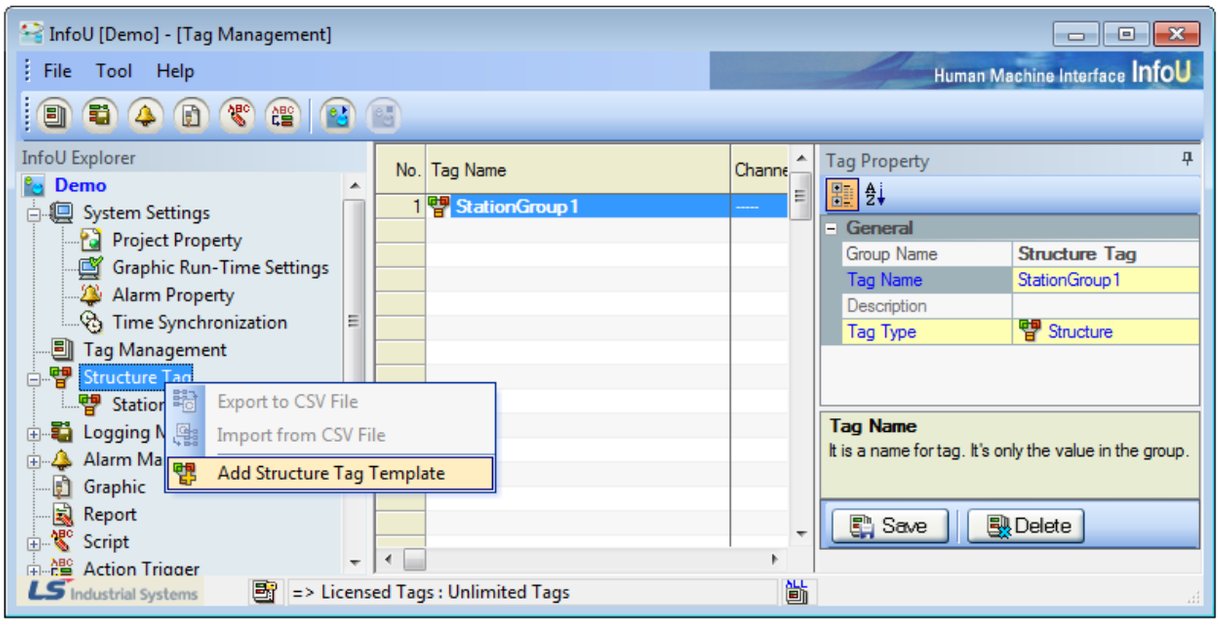
(3) Item property
It means the property of the item. You can specify the unique properties of the item such as name, type, Min./Max. values, etc.(Refer to the Tag Properties Manual)

Notice

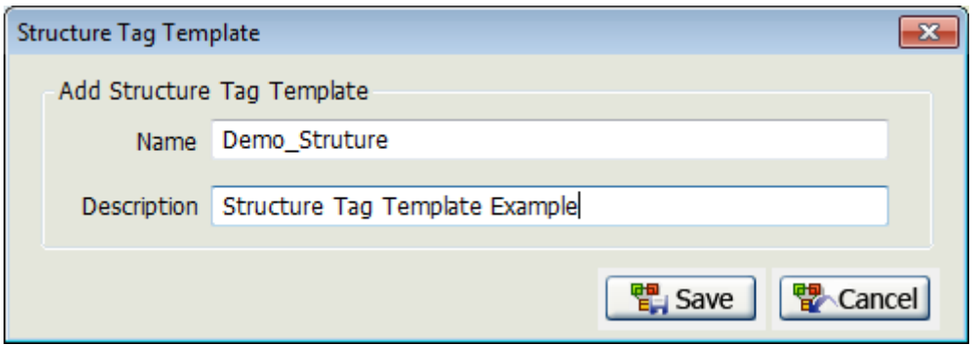
The definition of the structure tag is to define templates and does not mean the actual tag.

8.7.2 Addition of Structure Tag Templates

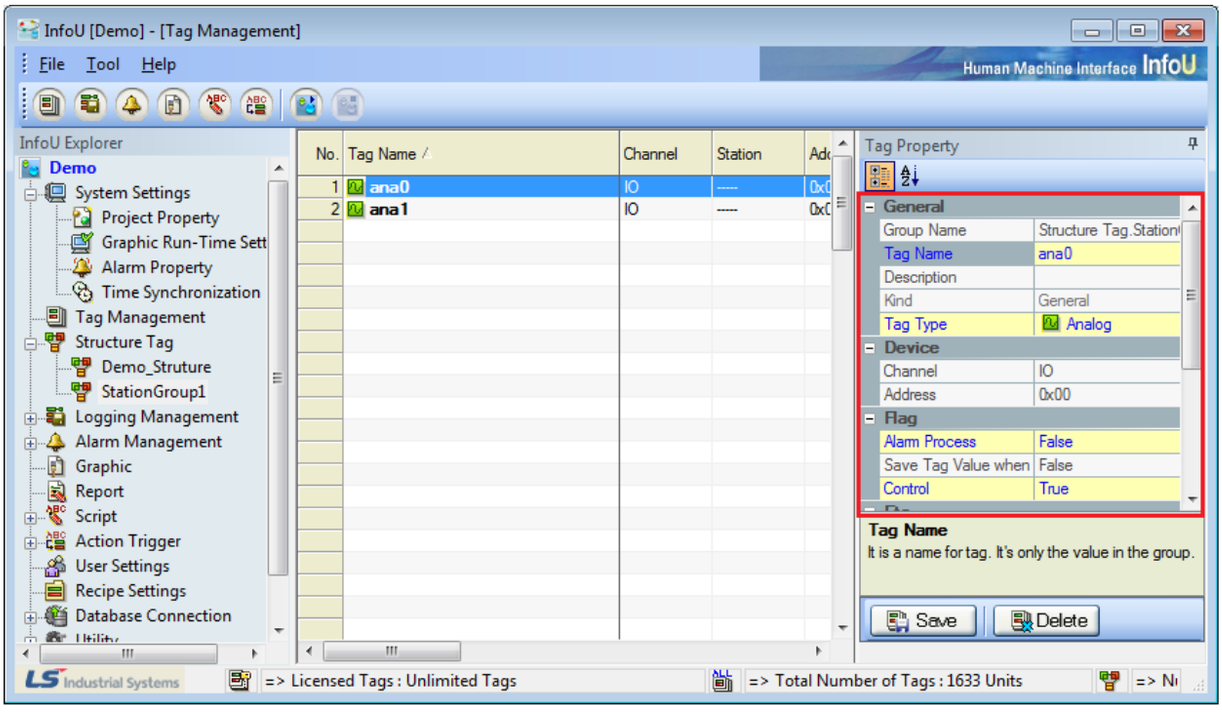
To add the structure tag templates, double-click the structure tag in the tool search or after clicking with the right mouse button, select [Add Structure Tag Template]. In addition, in the template list screen, after clicking with the right mouse button, you can select [Add Structure Tag Template] in the popup menu.



If you select the menu, the window to define the structure tag template will show up as below.



8.7.3 Addition of Structure Tag Items



It is the function to add the items that form the structure tag template.

The structure tag items can be input in the property window on the right side of the InfoU engineering.

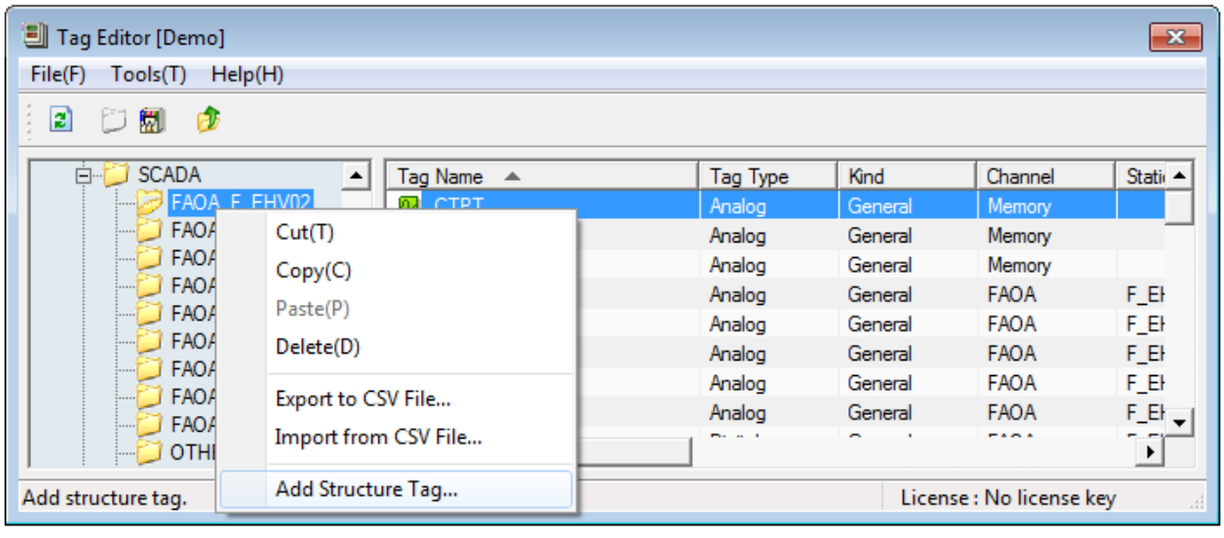
It is similar to the definition of tags but there are some differences. The characteristics of the structure tag items are as below. For more details on each item, refer to the properties of tags.

Item	Description
Tag type	Unlike the general tag, the structure tag does not allow to set up [Group]. Choose one among analog / digital / string.
Communication channel	For the communication channel, you can select [I/O tag] and [Memory tag] only. If you choose [I/O tag], you can input the data at the time when the structure tag is actually input to the tag DB, namely, at the time when the structure tag is added and instantiated. When you select [Memory tag], set the relevant tag as the memory tag.
Station	In common with the communication channel, the stations can be set all at once by adding the relevant template to the structure tag at the time of instantiation.

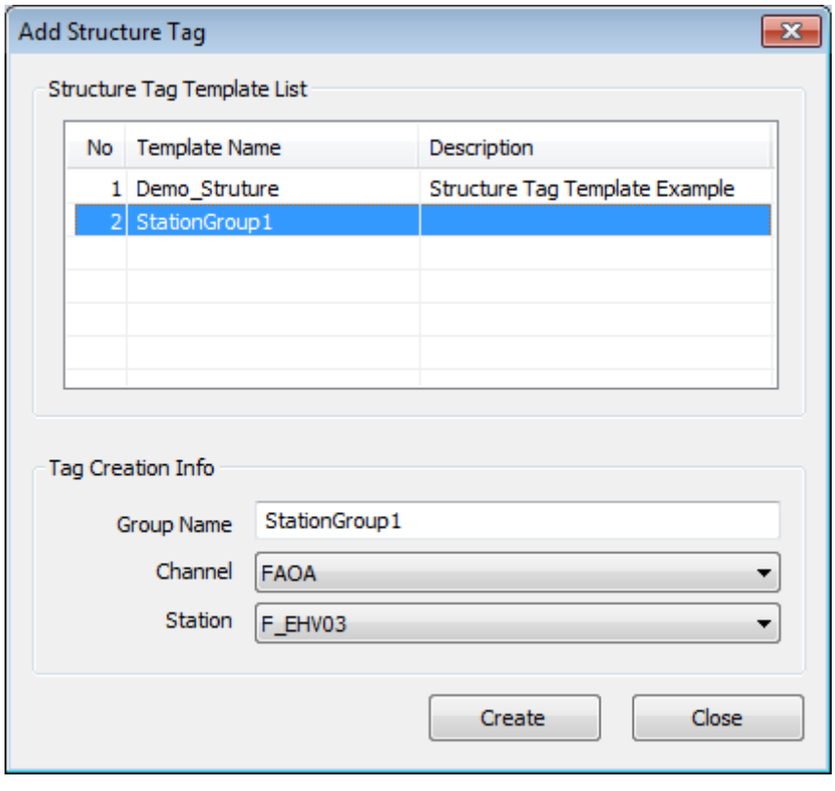
8.7.4 Addition of Structure Tags (Creating the instance of structure tags)

The previously performed works are to define the set structure of the tags that have specific pattern for creating the structure tag templates and structure tag items.

Definition of the structure tag is performed by the InfoU engineering module and the instantiation of the actual tag is performed by the tag editor. The creation of structure tags can be executed in the tree items of the tag editor. Select [Add Structure Tags] in the popup menu of the tree items of the tag editor.



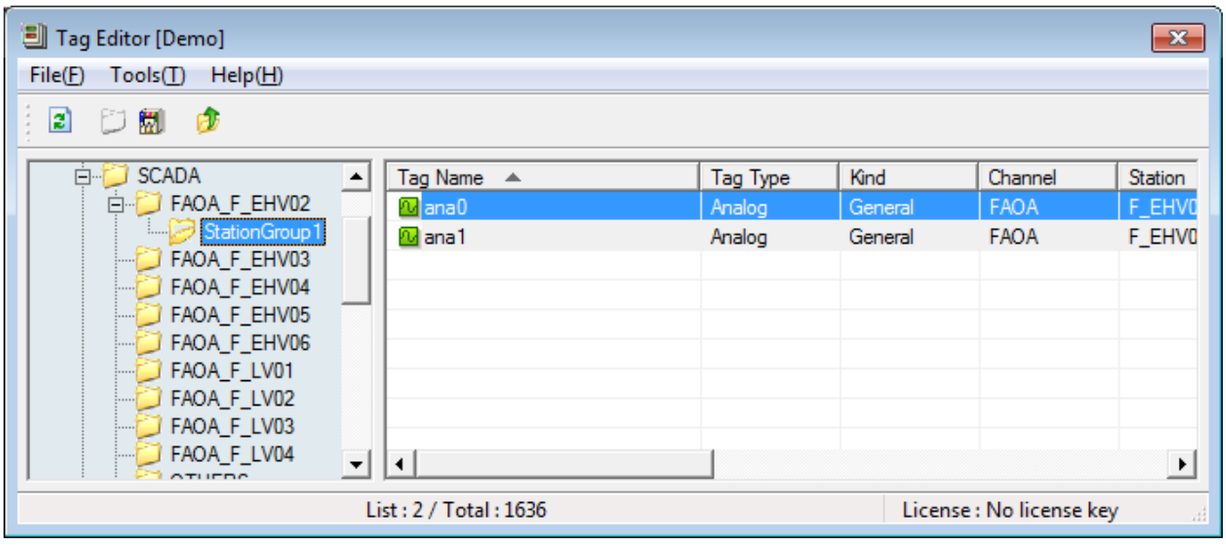
If you select the menu, the window to create the structure tag will show up.



The description on each item is as shown below.

Item	Description
Structure tag template list	The structure tag template list created in the InfoU engineering is shown. A user can select the structure tag template to create.
Group name to create	Input the parent group name at the time when the items of the structure tag are instantiated.
Communication channel	Input the communication channel at the time when the items of the structure tag are instantiated.
Station	Input the station at the time when the items of the structure tag are instantiated.

After inputting the data as shown in the Fig., if you click the [Create] button, the structure tags of the structure tag templates will be made as follows.



The created tags can be used in the same manner as the general tag.

8.8 Others

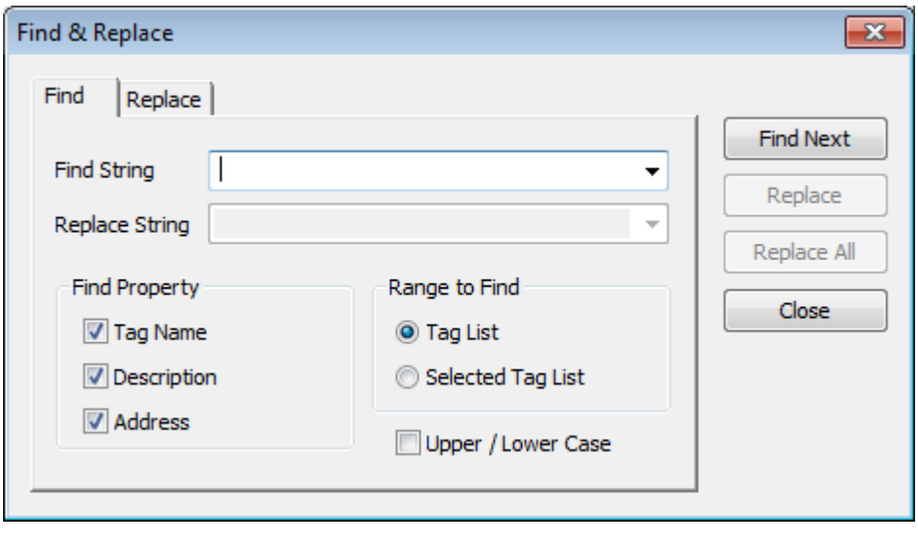
8.8.1 Find / Replace

Through the 'Find / Replace' function, you can easily find the tag that you want and save the changed details.

(1) Find

It is the function to find the items that correspond with the targeted string from the tag list.

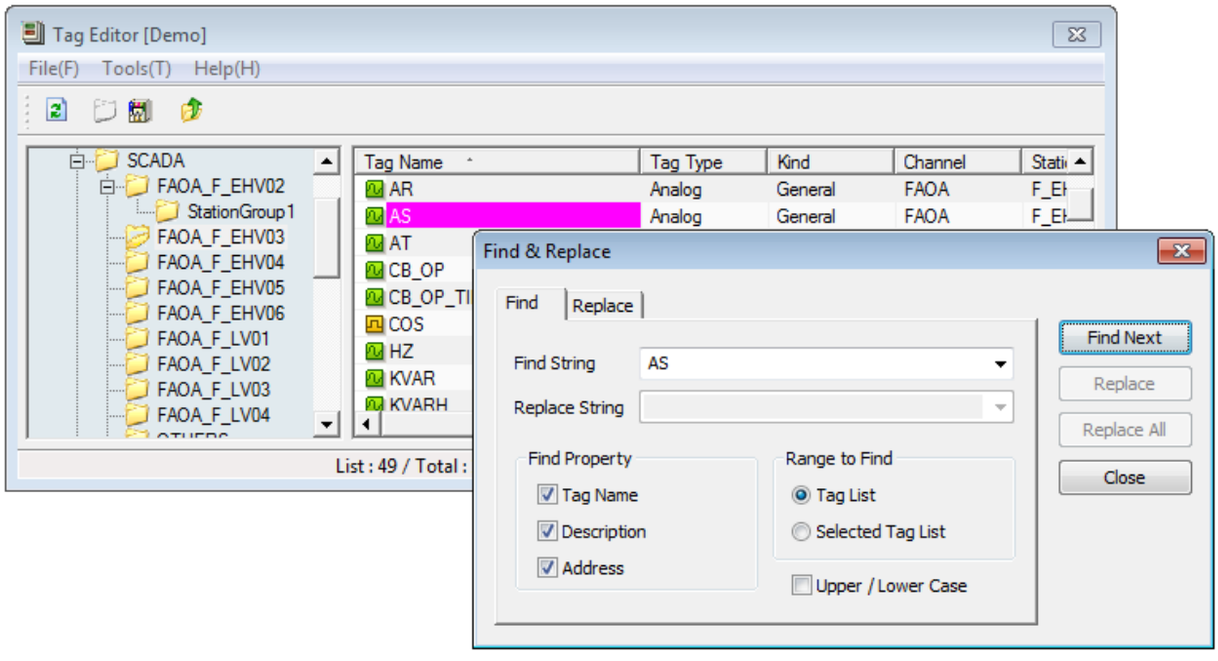
If you execute [Find] in the popup menu of the tag list, the below screen where a user can input data will show up.



The descriptions on the displayed properties are as follows.

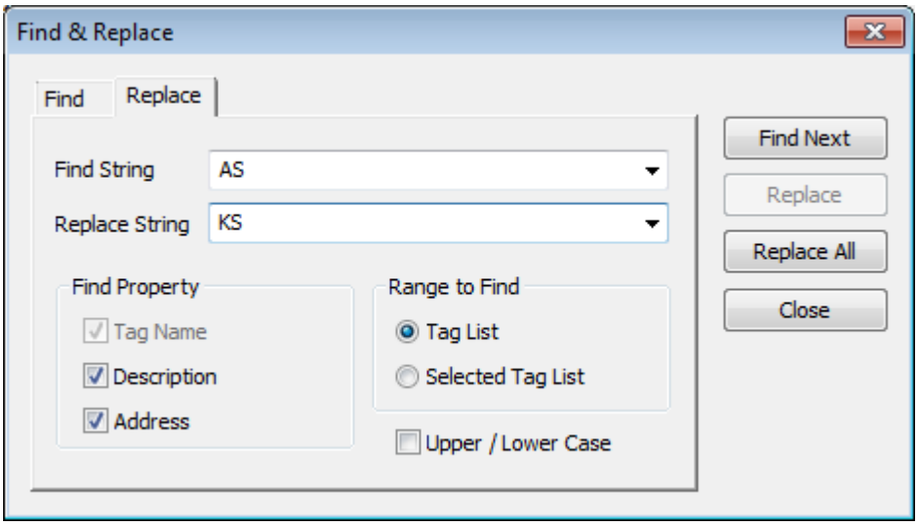
Item	Description
String to find	Input the string that you are looking for.
Property to find	Select the property that you are looking for. The 'Find' function can be performed for the tag name, tag comment, I/O address property.
Range to find	Tag list : Finds the string from the currently displayed tag list. Selected list : Finds the string from the selected are of tag lists
Upper/Lower Case	Select this when you try to find case-sensitive items.
Find Next	Finds the next object.
Close	Closes the window for Find.

After inputting a specific string, if you execute the 'Find' function, the found items are displayed in a different color in the tag editor list. When you click the [Find Next] button, the screen will move to the next item automatically.



(2) Replace

It is the function to find the string to change from the tag list and replace it. If you execute [Replace] in the popup menu of the tag list, the below screen where a user can input data will show up. After selecting 'Find', if you click the [Replace] button, the targeted details will be changed.



The items of the 'Replace' screen are as follows.

Item	Description
String to find	Input the string that you are looking for.
String to replace	Input the string to replace.
Property to Find	Select the property that you are looking for. The changeable properties are the tag command and I/O address.

Range to find	Tag list : Finds the string from the currently displayed tag list. Selected list : Finds the string from the selected are of tag lists
Upper/Lower Case	Select this when you try to find case-sensitive items.
Find Next	Finds the next object.
Replace	Select this when you try to replace the targeted string.
Replace All	Replaces all strings that correspond with the string to find within the range.
Close	Closes the window for Change.

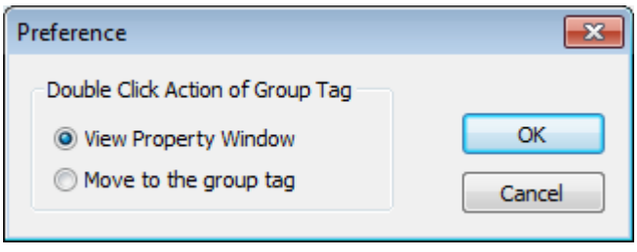
8.8.2 Configuration

It is the item that can be settable for convenience and additional functions of the tag editor.

(1) Setting double-clicking of the group tag

It is the option to determine whether showing the properties for the group tag or entering into the relevant group when you double-click the group tag in the tag list.

When you execute [Tool] → [Option] menu, the following setup window will show up. After selecting the option that you want, if you click [OK] button, the option will be immediately applied.



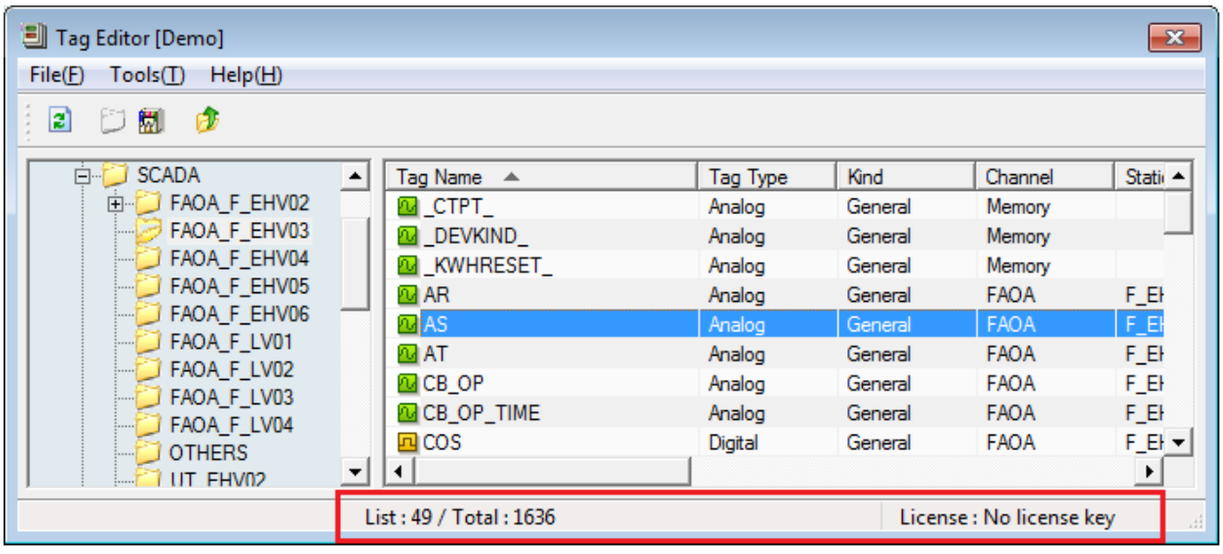
8.8.3 License

The InfoU's license uses the hard lock key of a USB type and the lock key should be applied in accordance with the number of tags. In case the license is not certified, there may be a variety of restrictions on functions. If you do not have the lock key, the system will shut down automatically 1 hour after startup.

The number of license tags is applied to the I/O tag only, not to the system tag. The group tags and memory tags are available regardless of the license.

The license information and the number of tags are displayed in a status bar of the tag editor.

The screen is as shown below.



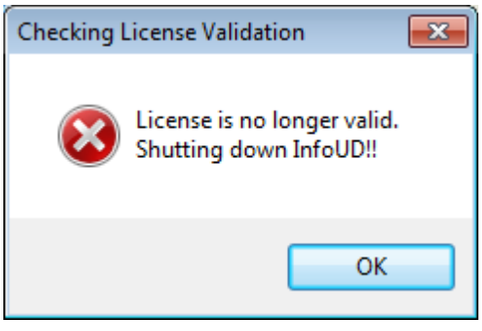
For the license information, there are largely three items shown in the status bar as below table.

Items	Description
No key	License : No license key
Limitation on the number of tags	The number of the current license tags / The number of the total tags that can be registered License : 424 / 500
Unlimited	License : **Unlimited**

Notice

Here are a few things related to the license that you need to be aware of.

☞ If the invalid license is used, the below warning message will show up in an hour and then, the program will be terminated.



☞ If the number of available licenses is exceeded, the tags cannot be saved so be careful of this.

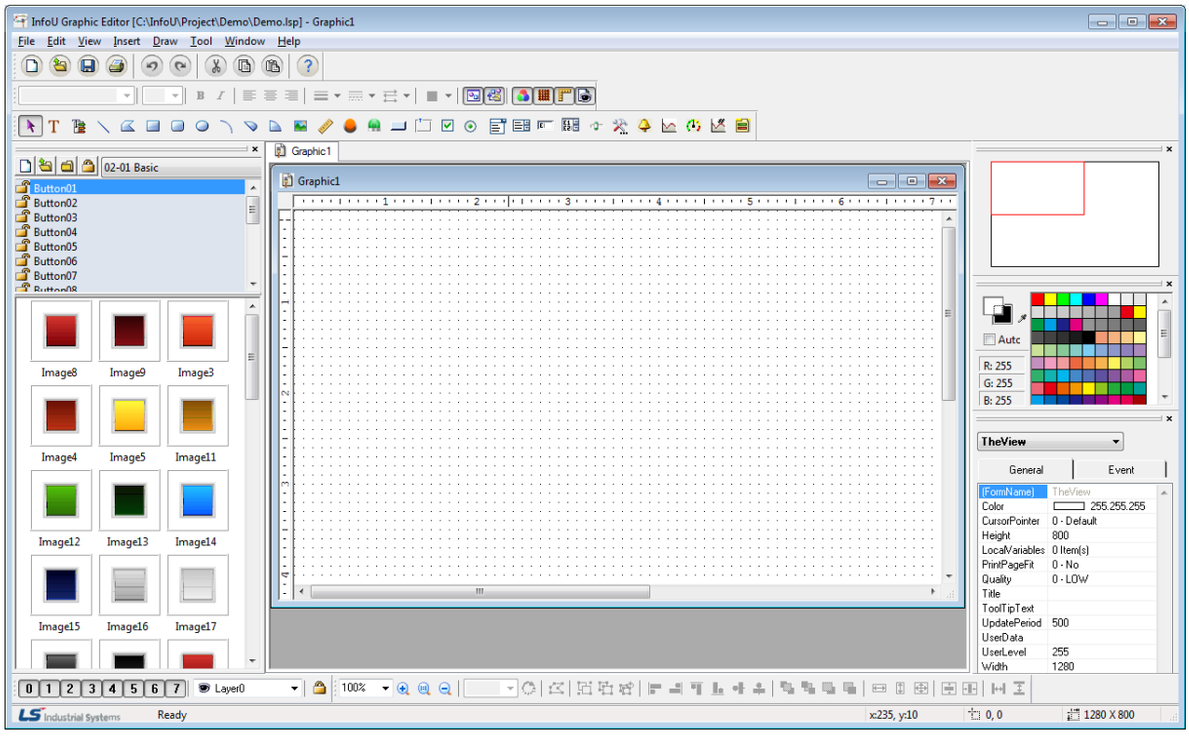
Chapter 9 Graphic Editor

This graphic editor provides the following various functions for editing the monitoring screen as a program used to edit the process screen.

- Graphic drawing tools
- Various graphic libraries
- Support of ActiveX container and its server, ActiveX control
- OLE automation interface for execution and manipulation from other application programs
- Visual basic script for robust applications
- Exclusive ActiveX control and availability of using ordinary ActiveX
- Function to inquire tag information used in the object and screen
- Function to save tag information used in the object and screen in the cross-reference database
- Function to prohibit or allow control according to the user's authorization
- Function to edit diverse and easy-to-use objects
- Support of various dynamic properties and user's action functions to display dynamic information through animation based on the site data
- Support of template screen

9.1 Configuration of Graphic Editor

This graphic editor consists of the following menus and tool collections.



As a system development tool, this InfoU graphic editor defines and sets up system components needed to perform projects such as defining dynamic properties of screen configuration, setting up execution environments and setting up page environments. The figure above shows an example of the InfoU graphic editor screen when a basic page is created and the main tool collection needed for the screen configuration is as follows.

- Standard tools: Command icons the most frequently used in the graphic editor.
- Object tools: Object icons needed when writing a page.
- Format tools: Icons needed to define font or line prosperities.
- Layer tools: Tools that enable to perform layer organization works useful for editing.
- Draw tools: Icons related to Zoom, Group/Regroup and Align.
- Color tools: A tool window that designate the color of page or object conveniently.
- Object Property Window: A window to define each object's properties or define events with scripts.
- Library: A window to register/delete/edit symbols in the library.
- Navigation: Tools to ensure easy moving to a work area the user wants to go.
- Status tools: Tools to show major properties such as object coordinates or Help

9.2 Menu Configuration

Upper-level Menu	Menu	Description
File	New	Create a new page.
	Open	Open one among existing documents
	Close	Close the current document.
	Close All	Close all documents
	Import From WMF File	Import WMF file in the currently activated document.
	Save	Save the activated document with the same name.
	Save As	Save the activated document with the different name.
	Print	Print document.
	Print Preview	Display the contents to be printed in the screen.
	Print Setup	Set up print options
	Recent Files	Open documents that have been worked recently.
	Exit	Exit the graphic editor.
Edit	Undo	Cancel the previously executed command.
	Redo	Re-input the cancelled command.
	Cut	Delete a certain range of data from the document and move it to the clipboard.
	Copy	Copy a certain range of data to the clipboard
	Paste	Paste the copied data to the document from the clipboard.
	Delete	Delete the selected data.
	Select All	Select the entire objects in the page.
	Find/Replace String	Find a certain string used for the object in the page and replace the found string with other string.
	Find/Replace Tag Link	Find a certain tag used for the object in the page and replace the found tag with other tag
	Tag Link MS Excel Export	Send tags mapped in the page to Excel file.
	Smart Symbol Instance Settings	Set the instance for the smart symbol.
	Batch Dynamic Monitoring Properties	It batch-changes the same types of properties mapped in the dynamic property setting window.
	Dynamic Objects Show	The only object whose dynamic properties are mapped is displayed on the screen.
View	Toolbars	Show or hide standard, object, format and draw tools.
	Status Bar	Show or hide status tools.
	Library	Show or hide the symbol library dialog box.

	Project Page	It displays or hides the project page window.
	Page Tab	It displays the project page as the page tab style or hide it.
	Color	Show or hide color tools.
	Navigation	Show or hide navigation tools for moving to other screen.
	Layer	Show or hide layer tools.
	Dynamic Properties	Show or hide dynamic properties of the object and the screen for action settings.
	Object Properties	Show or hide the property screen for editing object types or shapes.
	Ruler	Show or hide the ruler.
	Snap to Grid	Setup or cancel snapping to grid.
	Grid Settings	Designate grid intervals and colors by displaying the grid setting dialog box.
	Zoom	Zoom in/out the page with setting rates.
	Full Screen	Convert the page into a full screen size.
Insert	Figure	Draw various figure objects.
	Window Control	Draw various window control objects.
	Text	Input strings.
	Tag Value	Create an object with tag values.
	Image	Insert image files.
	Meter	Insert meter objects
	Image Button	Insert image button objects
	Image Lamp	Insert image lamp objects
	ActiveX	Insert OCX objects.
	Alarm Viewer	Insert alarm viewer objects.
	Trend Viewer	Insert trend viewer objects.
	Gauge	Insert gauge viewer objects.
	List Trend	Insert list trend objects.
	Recipe	Insert recipe objects.
Draw	Group	Create one group object with several objects.
	Ungroup	Cancel the group object.
	Regroup	Recover the control function over the objects that have been grouped and then cancelled to re-create the group.
	Order	Bring the select object to the very front or sent it to the very end or bring it to the front or send it to the rear.
	Edit Points	Change the shapes of multiple curves or looped curves freely.
	Open/Close	Convert multiple curves to looped curves and looped curves to multiple curves.
	Align	Align, assign and arrange the selected objects.
	Make Same Size	Make the sizes of the selected objects equivalent.
	Rotate	Rotate the selected objects freely to their original locations or 90°rotation to the right

Tool	Run Mode	Preview the operation screen of the runtime.
	Objects	Change the order of the object list and set up the tap sequence for keyboard run.
	Declutter Settings	Set up a range of enlarge rate for each layer to be displayed in each screen during the runtime.
	Graphic Runtime Settings	Set up the execution environment of graphic runtime.
	Option	Set up the environment such as number of cancellation or mode to select multiple options.
Windows	Cascade	Arrange windows in a cascade form.
	Tile	Arrange windows in a tile form.
	Arrange Icons	Arrange windows to the location to designate various tool bar defaults.
Help	User Manual	Show the Help items of the graphic editor.
	About Graphic Editor	Show version information of the graphic editor.

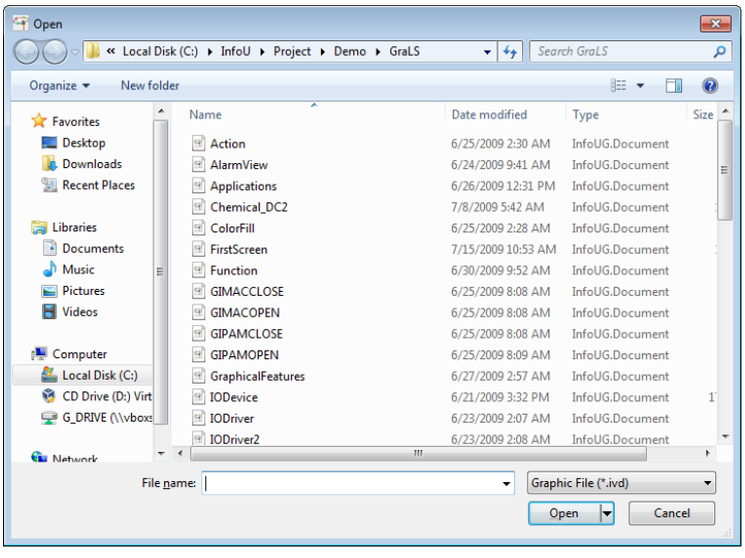
9.2.1 File

(1) New

A new graphic file is created.

(2) Open

This command is used to open one of the existing documents. If this command is made, a dialog box appears to allow the user to designate a file to open from it.



(3) Close

The currently activated document is closed with this command. Before the document is closed, it needs to be saved. If it is closed without saving, the changed contents that have been made before the latest saving will disappear. When it is closed, a dialog box appears to allow the user to designate a name and save it with the name.

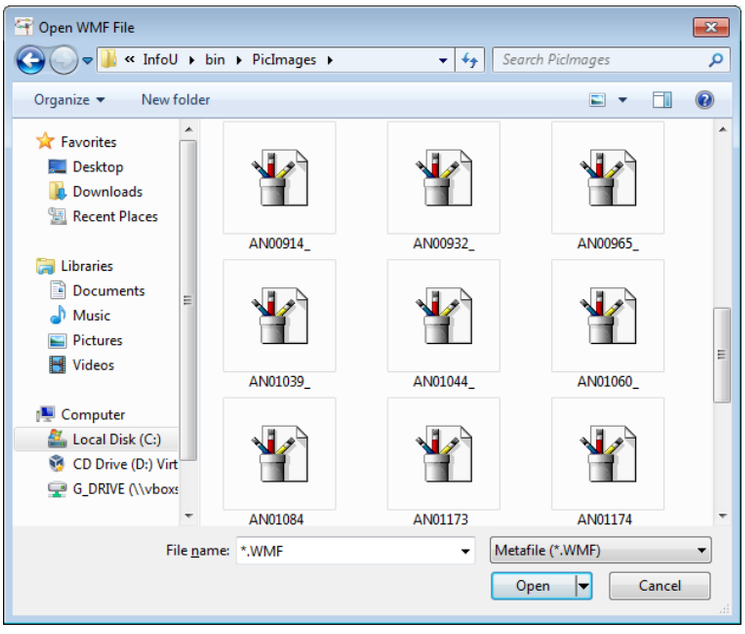
(4) Close All

Currently open documents are all closed so you need to save them before closing.

(5) Import from WMF file

A WMF file is inserted to the current work screen in the graphic editor's own format type. If this command is made, a dialog

box to import WMF files appears to allow the user to select one.

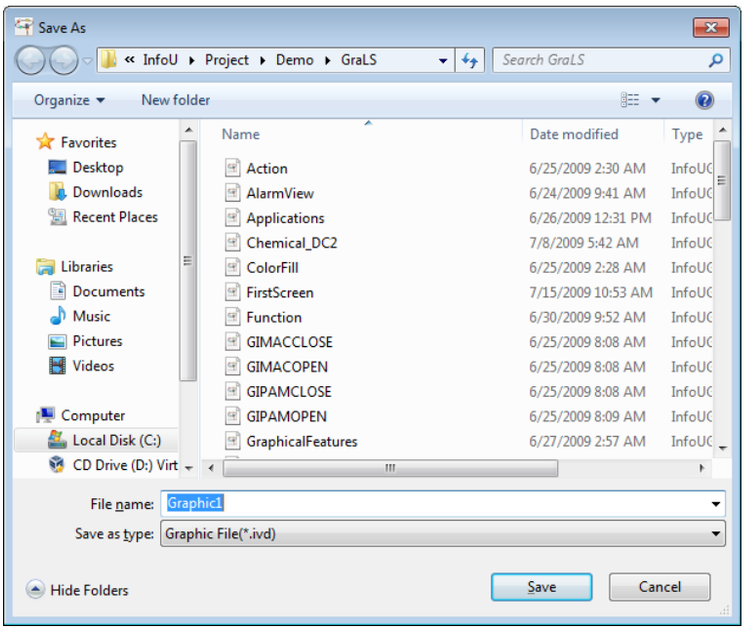


(6) Save

This command is used when saving the activated document with the same name as the current one. If the page document is saved for the first time, the “Save” dialog box appears to allow the user to designate its name. If the user wants to save the document with other name, he/she may choose “Save As” command.

(7) Save As

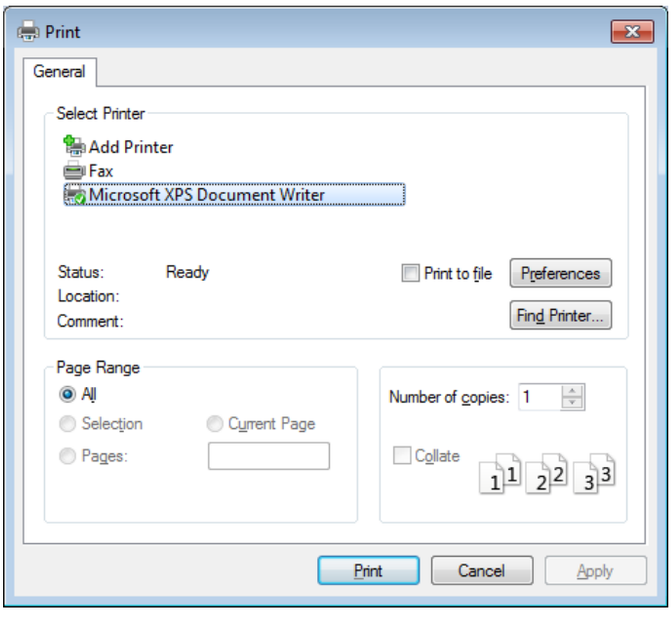
This command is used when saving the activated document with a different name from the current one. the “Save” dialog box appears to allow the user to designate its name. If the user wants to save it with the current name, he/she may use “Save" command.



(8) Print

This command is used when printing document. If this command is executed, the “print” dialog box appears to allow the

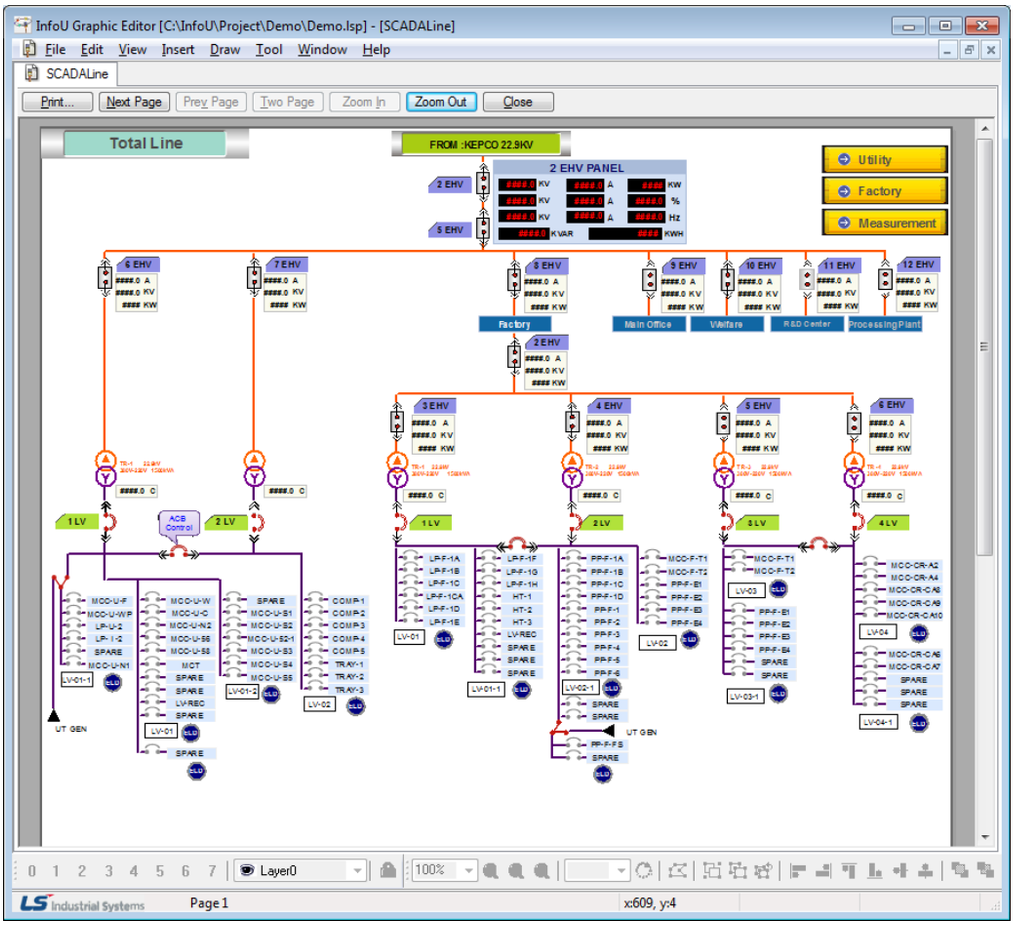
user to select a print range, number of copies, printer and printer option.



(9) Print Preview

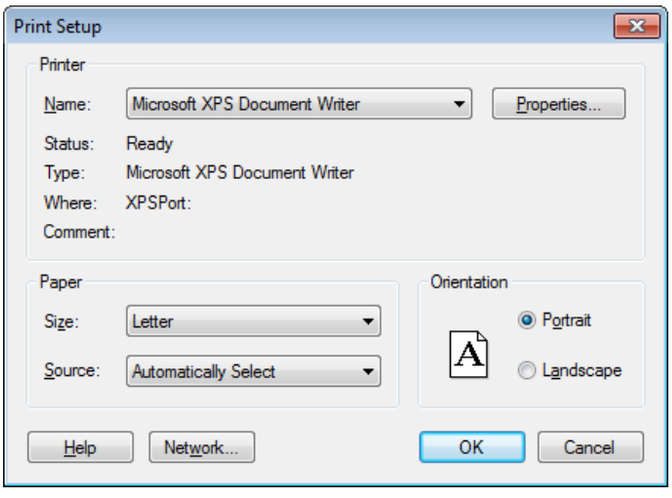
This command allows to user to preview the content to be printed in an actual output format in advance.

If this command is performed, a preview window is displayed. The user may perform various functions such as view one or two pages, view the front and rear page, zoom in/out and print command with the preview tool collection.



(10) Print Setup

Letter size and orientation for printing documents can be selective.

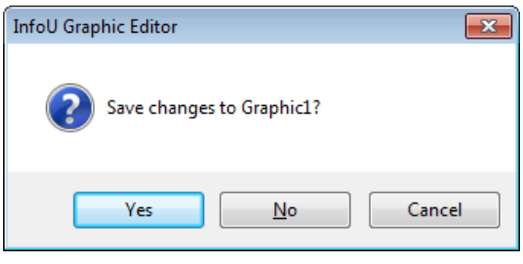


(11) Recent Files

The user can open the documents that have been worked recently in this graphic editor. Up to 10 recent documents can be displayed. The user can select one among the displayed files.

(12) Exit

The graphic editor is ended. If any changed content exists without being saved, a dialog box appears to ask the user whether to save it or not. At this time, press “Yes” if he/she wants to save it otherwise press “No.” If “Cancel” is selected, this ‘Exit’ command is cancelled.



9.2.2 Edit

(1) Undo

This command is used when the user wants to cancel the contents that have been recently inputted and return to the previous status. Up to 20 events of inputting can be cancelled.

(2) Redo

This command is used when re-performing the cancelled contents.

(3) Cut

This command is used to delete the selected data from the document and copy them to the clipboard. If no data is selected currently, this command can not be used. If this “Cut” command is performed, the previous data in the clipboard is replaced.

(4) Copy

This command is used to copy the selected data to the clipboard. If no data is selected currently, this command can not be

used. If this “Copy” command is performed, the previous data in the clipboard is replaced.

(5) Paste

This command is used to copy data in the clipboard and insert them to the document. If the clipboard is empty, this command can not be used.

(6) Delete

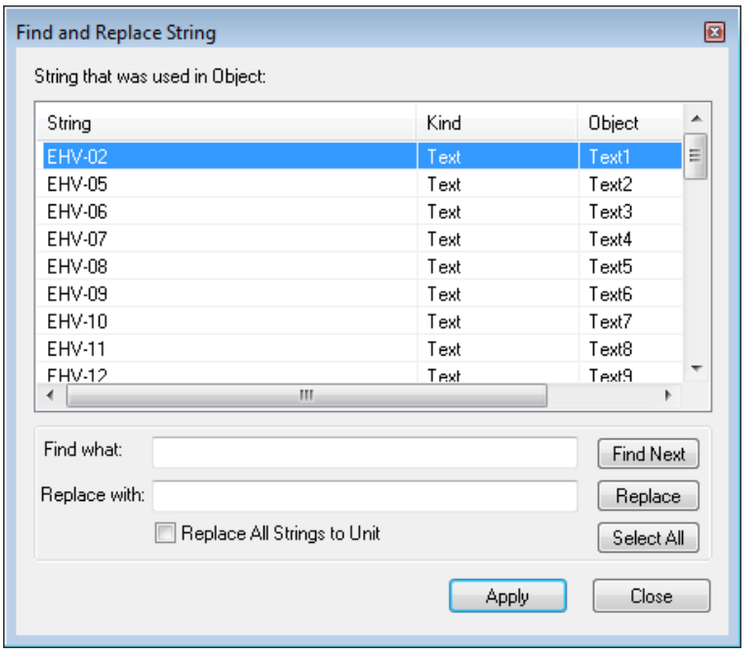
This command is used to delete the selected data. If no data is selected currently, this command can not be used. If important data are about to be deleted, a dialog box appears to ask the user whether he/she really wants to delete them, At this time, he/she may press “Yes” to delete them or he/she may press “No” to cancel the ‘Delete’ command.

(7) Select All

The entire objects in the page are selected.

(8) Find/Replace String

This command is used to find a string that was used throughout the objects in the page and replace the found one with other one. When this command is executed, the “Find and Replace String” dialog box appears.



1) Find what

Input the string to replace with.

2) Replace with

Click “Replace” button to display the planned strings in the list.

3) Apply

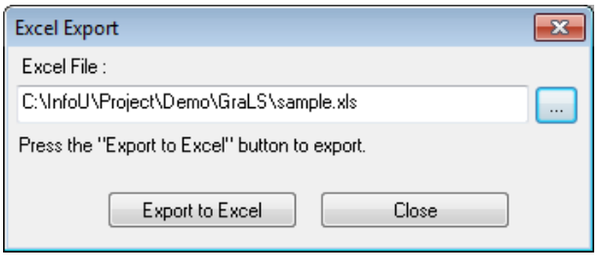
Click “Apply” button to replace the strings with the ones displayed in the list.

(9) Find/Replace Tag Link

This command is used when finding mapped tags among the entire objects or the selected objects in the page or replacing the found tags with others. When this command is executed, the “Find and Replace Tag Link” dialog box appears. For detailed explanation, refer to the “Tag Link” provision.

(10) Excel Export

This command is used when exporting the mapped tags in the page to excel.



(11) Smart Symbol Instance Settings

The dialog box is provided to set the instance for the smart symbol. For more details, refer to the items of 'Smart Symbol'.

(12) Batch Dynamic Monitoring Properties

An across the board change the same types of mapped properties in the dynamic property window. Among the selected objects, they are changed into the properties mapped to the master object in a lump. Accordingly, in order to change the master object, select the relevant object with a mouse, while holding the CTRL key.

(13) Dynamic Objects Show

The only objects whose dynamic properties are mapped are displayed on the screen.

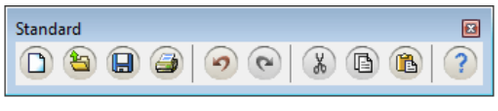
9.2.3 View

(1) Toolbars

There are [Standard], [Object], [Format] and [Draw] tools and they are either shown or hid with various related tools in the tool window upon clicking on each one.

1) Standard

It is used when showing or hiding standard tools. Standard tools are command icons the most frequently used in the graphic editor. When it is shown, the check mark appears in front of the menu.



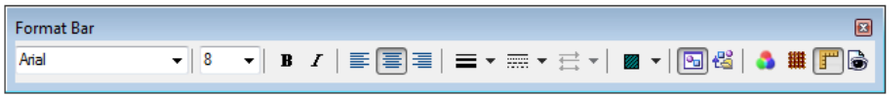
2) Object Bar

It is used when showing or hiding object tools. Object bar has object icons needed to write pages. When it is shown, the check mark appears in front of the menu. For details, refer to the 'Insert' menu.



3) Format Bar

It is used when showing or hiding format tools. This format bar has various icons including font, size, bold, underline, italic, arrange to the left, center and right, line thickness, style and arrow type. When it is shown, the check mark appears in front of the menu.



4) Draw Bar

It is used when showing or hiding draw tools. This draw bar has various icons related to functions to arrange objects in the page and icons for Group/Regroup/Ungroup, Zoon in/out and Rotate. When it is shown, the check mark appears

in front of the menu. For details, refer to the 'Draw' menu.



(2) Status Bar

It is used when showing or hiding status tools. 'Help' messages or coordinates are displayed in the status bar. When it is shown, the check mark appears in front of the menu.

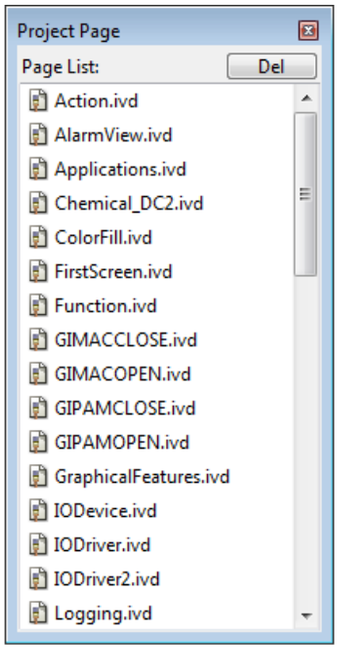


(3) Library

It is used when showing or hiding the library dialog box. In addition to the library provided by the system, the user may register/delete/edit his/her own library. For detailed explanation, refer to the "Library" provision.

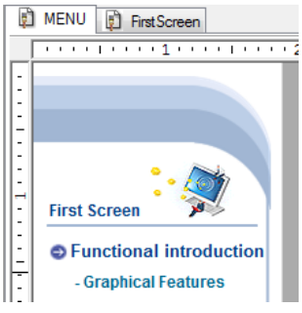
(4) Project Page

It displays or hides the project page window. You can delete the page file with the 'Delete' button in the project page window.



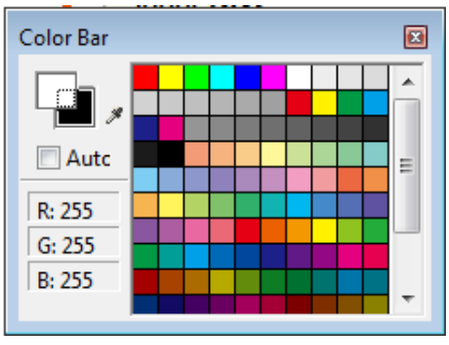
(5) Page Tab

It displays the project page as the page tab style or hides it. If you select the tab by clicking with the right mouse button on the page tab, the page will be closed.



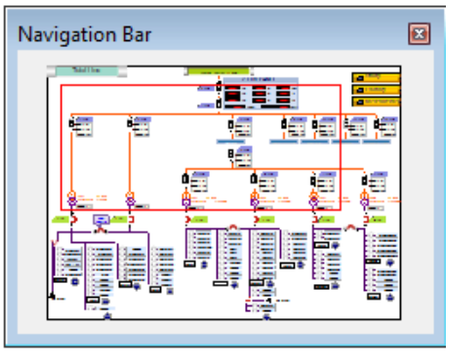
(6) Color

It is used when showing or hiding color tools. This color bar has a part to set up view colors, background colors and transparency setting. When it is shown, the check mark appears in front of the menu. A user can separately specify and maintain colors up to 10.



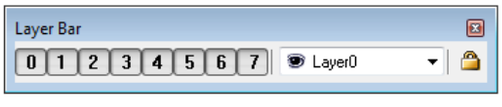
(7) Navigation

It is used when showing or hiding navigation tools. If the current work area is small, this navigation bar allows the user to move pictures to the work area he/she wants. When it is shown, the check mark appears in front of the menu.



(8) Layer

It is used when showing or hiding layer tools. This lay bar can either show or hid the relevant layer and it can also lock it. For detailed explanation, refer to the “Layer” provision.



(9) Dynamic Properties

It is used when setting up dynamic properties and action functions of the relevant object. First, select an object and execute this command to show a dialog box to set up its dynamic properties. The displayed dialog box allows the user to set up dynamic properties and action functions of the relevant object. For detailed explanation, refer to the “Dynamic Property Definition” provision.

1) Horizontal Size

A function to change the object’s size in the horizontal direction according to tag values.

2) Vertical Size

A function to change the object’s size in the vertical direction according to tag values.

3) Horizontal Move

A function to move the object in the horizontal direction according to the designated tag value.

4) Vertical Move

A function to move the object in the vertical direction according to the designated tag value.

5) Horizontal Fill

A function to fill the object in the horizontal direction according to the designated tag value.

6) Vertical Fill

A function to fill the object in the vertical direction according to the designated tag value.

7) Color

A function to change the object in the prescribed color according to the inputted tag value.

8) Blink

A function to blink the object in the specified interval according to the tag status.

9) Visible

A function either to show or hide the object according to the input tag value.

10) Rotate

A function to rotate the object by the designated angle degree according to the input tag value.

11) Display Value

It is used to create an object marked with its tag value.

12) Display String

A function to change the object with the prescribed string according to the inputted tag value.

13) Lamp

The function shows the On image or Off image depending on the input tag value.

14) Click

A control function to execute the defined action when clicking on the object with the mouse or keyboard.

15) Input Value

A function to select an object and input a value in the predefined dialog box with the mouse or keyboard.

16) Horizontal Drag

A function to drag the object in the horizontal direction with the mouse to convert coordinates values to tag values.

17) Vertical Drag

A function to drag the object in the vertical direction with the mouse to convert coordinates values to tag values.

(10) Object Properties

If an object is selected and [View] ☞ [Object Properties] is clicked, object properties of the selected object and the event list window appear. For detailed explanation, refer to the “Object Properties” provision.

(11) Ruler

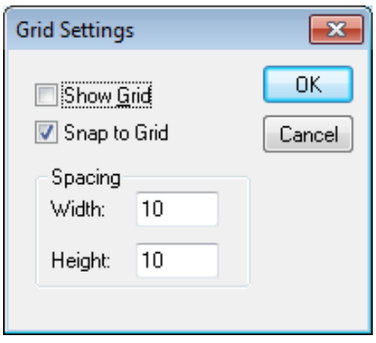
It is used when showing or hiding the rulers on the top and the left side in the current page.

(12) Snap to Grid

It is used to set up or cancel the ‘Snap to Grid’ command. If ‘Snap to Grid.’ is set up, objects move to adjust to the grid coordinates. If cancelled, objects can move freely.

(13) Grid Settings

It is used to designate grid spacing and whether to show grid. If this command is performed, the ‘Grid Settings’ window is displayed. Input grid spacing and check in the box of ‘Show Grid,’ grid appears in the screen.



(14) Zoom

To make a more delicate edition in the screen, use zoom in or out. From 400% to 25% zooming can be designated. It is used to display the full page to the screen size.

9.2.4 Insert

(1) Figure

1) Line

It is used to draw a line. If this command is executed, the edit mode is changed to the line draw mode.

2) Poly Line

It is used when drawing multiple curves or looped curves. If this command is executed, the edit mode is changed to the free draw mode.

3) Rectangular

It is used to draw a rectangular. If this command is executed, the edit mode is changed to the rectangular draw mode.

4) RoundRect

It is used to draw a round square. If this command is executed, the edit mode is changed to the round square draw mode.

5) Ellipse

It is used to draw a circle or an ellipse. If you execute this command, the editing mode will be changed into the drawing mode for an ellipse.

6) Arc

It is used when drawing a circle or an oval. If this command is executed, the edit mode is changed to the arc draw mode.

7) Chord

It is used to draw a fan shape. If this command is executed, the edit mode is changed to the chord draw mode.

8) Pie

It is used to draw a circular arc. If this command is executed, the edit mode is changed to the pie draw mode.

(2) Window Control

1) Button

It is used to draw a button. If this command is executed, the edit mode is changed to the button draw mode.

2) Group Box

It is used to draw a group box. If this command is executed, the edit mode is changed to the group box draw mode.

3) Check Box

It is used to draw a check box. If this command is executed, the edit mode is changed to the check box draw mode.

4) Radio Button

It is used to draw a radio button. If this command is executed, the edit mode is changed to the radio button draw mode.

5) Combo Box

It is used to draw a combo box. If this command is executed, the edit mode is changed to the combo box draw mode.

6) List Box

It is used to draw a list box. If this command is executed, the edit mode is changed to the list box draw mode.

7) Edit Box

It is used to draw an editor box. If this command is executed, the edit mode is changed to the edit box draw mode.

8) Multiple Edit Box

It is used to draw a multiple editor box. If this command is executed, the edit mode is changed to the multiple edit box draw mode.

9) Slider

It is used to draw a slider. If this command is executed, the edit mode is changed to the slider draw mode.

(3) Text

It is used to input text. If this command is executed, the edit mode is changed to the text input mode.

(4) Tag Value

It is used to create an object marked with its tag value. If this command is executed, the edit mode is changed to the object input with tag value mode.

(5) Image

It is used to insert various types of image files. If this command is executed, the edit mode is changed to the image insert mode and the 'Open' dialog box appears. Select an image file to insert in the displayed dialog box.
For detailed explanation, refer to the "Image Insert" provision.

(6) Meter

It is used to create the measuring instrument object. If you perform this command, the editing mode will be converted into the input mode for the measuring instrument object. You can specify the color, large/small gradation size, the number of large/small gradations, label, position, etc.

(7) Image Button

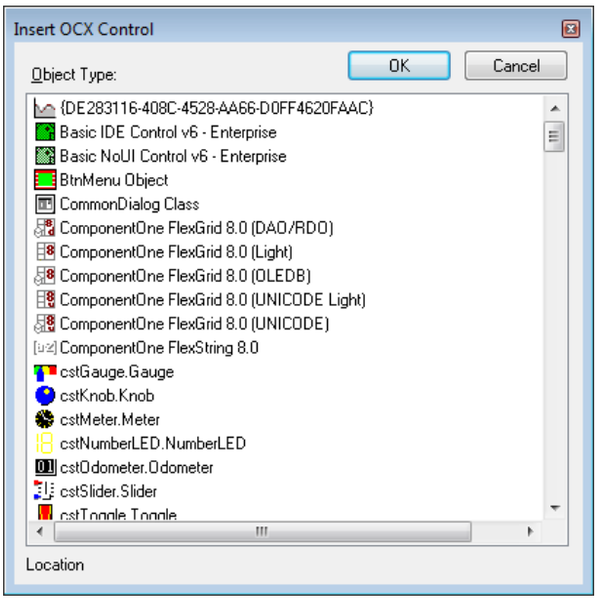
It is used to insert picture files with various formats into the Up/Down image that will be put into the button image. If you perform this command, the editing mode will be changed into the image button mode and the setting dialog box will be displayed. In the displayed dialog box, you can just select the picture file that you want to insert. For more details, refer to 'Objects using images'.

(8) Image Lamp

It is used to insert picture files with various formats into the On/Off image that will be put into the button image. If you perform this command, the editing mode will be changed into the image lamp mode and the setting dialog box will be displayed. In the displayed dialog box, you can just select the picture file that you want to insert. For more details, refer to 'Objects using images'.

(9) ActiveX

It is used to insert Active X object. If this command is executed, the edit mode is changed to the Active X insert mode and the 'Open' dialog box appears. Select an Active X object to insert in the displayed dialog box.



(10) Alarm Viewer

It is used to insert Active Viewer object. If this command is executed, the edit mode is changed to the Active Viewer object insert mode.

(11) Trend Viewer

It is used to insert Trend Viewer object. If this command is executed, the edit mode is changed to the Trend Viewer object insert mode.

(12) Gauge

It is used to insert Gauge object. If this command is executed, the edit mode is changed to the Gauge object insert mode.

(13) List Trend

It is used to insert List Trend object. If this command is executed, the edit mode is changed to the List Trend object insert mode.

(14) Recipe

It is used to insert Recipe object. If this command is executed, the edit mode is changed to the Recipe object insert mode.

9.2.5 Draw

(1) Group

If this command is executed while more than two objects are selected, those selected objects are bound into one. During the period between the time when selecting ‘Group’ and the time when canceling this ‘Group,’ the selected objects move together as a group.

(2) Ungroup

It is used to cancel the selected objects as a group and release them into individual objects. If this command is executed, control functions defined for the group disappear. These control functions for the group can be recovered with the command ‘Regroup’.

(3) Regroup

To edit some of the objects in a group with control functions, cancel the 'Group' command and edit them. After finishing editing, use this command to recover those control functions. Regrouping is not allowed if other objects are newly added or some of them are deleted. This command is available only when original objects are edited.

(4) Order

It is used to decide the order of the overlapped objects. There are [Bring to the Very Front], [Send to the Very Rear], [Bring to Front] and [Send to Back].

(5) Edit Points

It is used to change shapes of free curve, multiple curve and looped curve objects. If this command is executed, a black rectangular mark appears on each vertex. The user may change shapes by selecting the displayed mark and moving it with the mouse.

(6) Open/Close

It is used to convert from multiple curves to looped curves and looped curves to multiple curves. If this command is executed, multiple curves are converted to looped curves or looped curves are converted to multiple curves.

(7) Align

As a function to arrange more than two objects at the same position vertically and horizontally, distribute or sort them, there are [Align to left], [Align to center], [Align to right], [Align to top], [Align to middle], [Align to bottom], [Same horizontal spacing], [Same vertical spacing], [Vertical center] and [Horizontal center].

(8) Make Same Size

It is used to assign the same size to all of the selected objects.

(9) Rotate

It is used to rotate objects in the direction the user wants. [Free Rotate] allows the user to rotate them freely and [Rotate 90°to the right] allows the user to rotate them by 90°. [Original Status] sends the direction of objects to their own status.

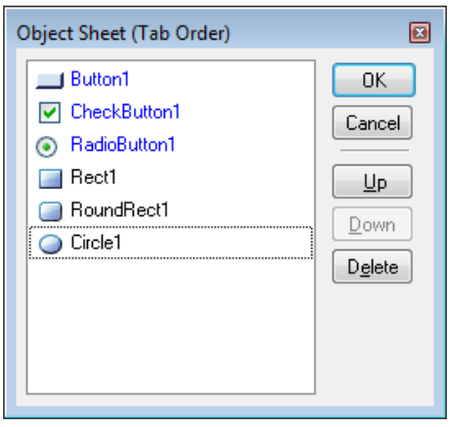
9.2.6 Tool

(1) Run Mode

If this command is executed, a new window is created and runtime begins to operate as in the Runtime screen. The user may test during the editing by loading its actual operation status to check whether it is properly operated.

(2) Objects

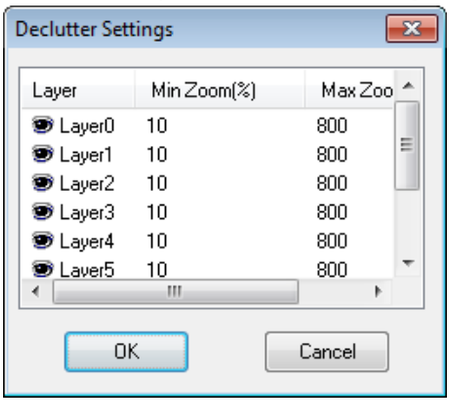
All of the activated objects are displayed in the screen. The user may set up the order among objects or rename or delete objects in the Object Sheet dialog box and he/she may also organize a sequence to move one object to the other one to ensure operation is enabled without mouse during the Runtime.

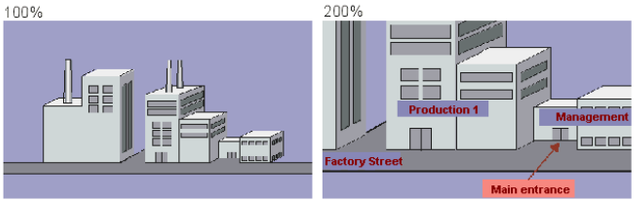


(3) Declutter Settings

During the Runtime, layers (the entire objects in the layers) can be shown or hid depending on the zoom in/ out rate. Therefore, overall objects can be seen in the zoom out screen while closer and more detailed objects can be displayed in the zoom in screen. The user can assign the minimum and maximum zoom rate unique to each layer. Each layer can be seen in the assigned range and the rate of its zoom in/ out are set in percentage.

For detailed explanation on how to assign to layers, refer to the “Layer” provision.





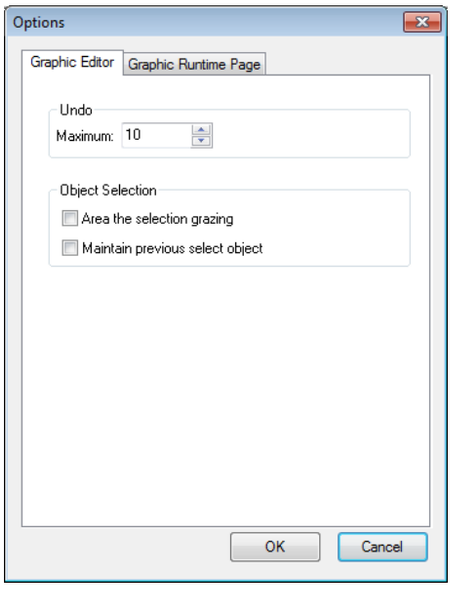
(4) Graphic Runtime Settings

It is used to edit settings during the Graphic Runtime such as window style, execution location, title bar or start page. If this command is executed, a “Graphic Runtime Environment Setting” dialog box appears.

For detailed explanation, refer to the “Execution Settings” provision.

(5) Options

1) Graphic Editor



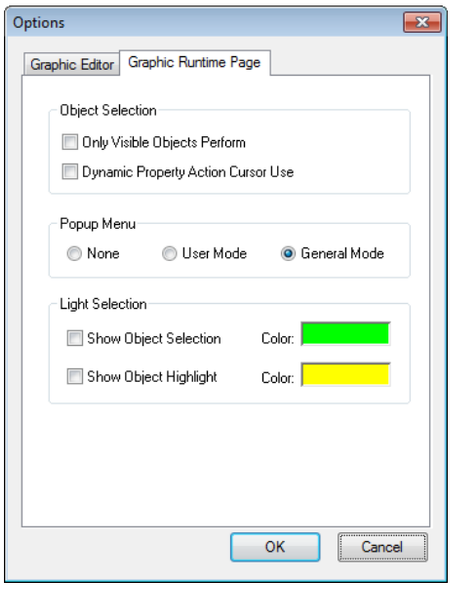
a. Undo

You can specify the maximum number of times of undo.

b. Object Selection

You can choose two options: Selecting the object of grazing area and maintaining the previous selected object.

2) Graphic Runtime Page



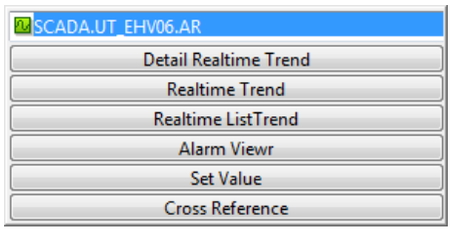
a. Object Selection

For the visible object only, you can specify whether applying dynamic property actions, whether using the dynamic property action cursor.

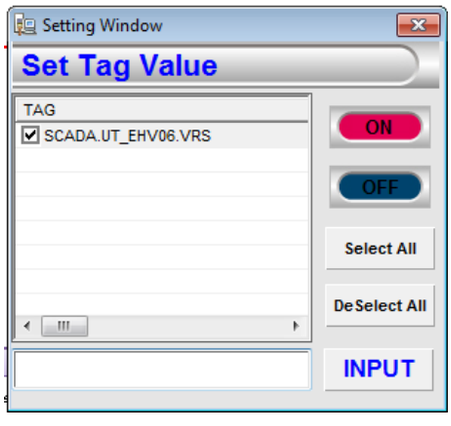
b. Popup Menu

A user can choose one of the three operations; Nonuse of the popup menu, user mode, general mode. In the user mode, a user can edit and control the popup menu page in a lump.

* General mode



* User mode



c. Light Selection

In the graphic runtime, through the lighting function, you can specify whether showing object selection, highlighting the object, in addition, you can also change the color. For more details, refer to ‘Lighting Function’.

9.2.7 Window

(1) Cascade

Windows are arranged in a cascade style.

(2) Tile

Windows are arranged in a tile style.

(3) Arrange Icons

Windows are arranged to the location to designate various tool bar defaults.

9.2.8 Help

(1) User Manual

It is used to show 'Help' provisions. Step-by-step commands and instructions on the displayed provisions are presented. A content button is clicked in the Help screen, the corresponding provision screen appears.

(2) About Graphic Editor

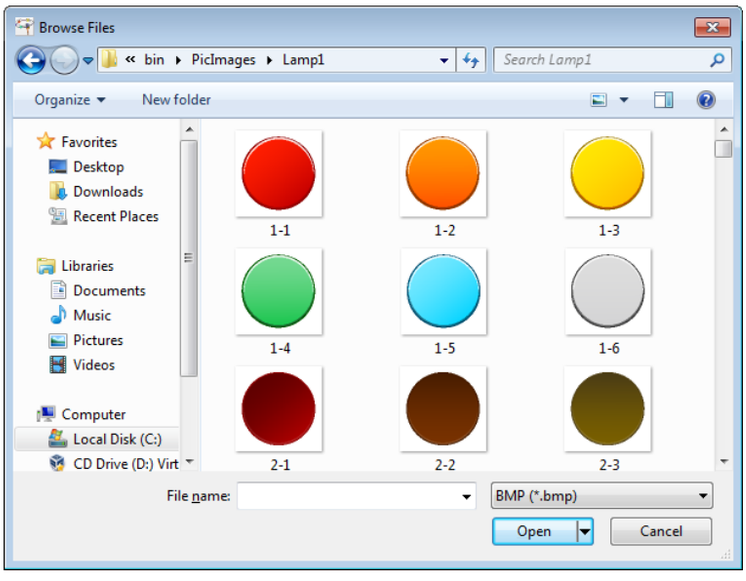
It is used to show the version information of the graphic editor. If this command is executed, version information and copyright message appear.

9.3 Insert Image

A graphic file format refers to the type of a file that contains graphic images. A graphic file can either lower or extend its image capacity and have a different file format depending on how the graphic images are compressed and saved. This InfoU Graphic Editor supports general-purpose graphic file formats.

9.3.1 Insert Image

If the user selects [Insert] ☞ [Image] in the menu or clicks 'Image' button in the Draw Bar to define an image area in the screen page, the 'Open' dialog box appears. The user may select an image file to insert in the displayed dialog box. The image file types that are supported by the Graphic Editor are BMP, JPG, GIF, PNG, TIF, WMF, EMF and etc. The user may also use the clip board to insert images by copying images from external APP and pasting them. Image names and formats can be checked and methods to save images, transparency degree and colors can be designated in 'Image Object Properties.' For GIF animation image objects, a time cycle of each frame can be set up.



9.3.2 Difference between Bitmap and Vector

(1) Bitmap

Bitmap refers to a way to divide one image into several points when saving images. With this way, the user may use various colors and produce vivid colors and flexible pictures like photograph. However, it has some limitations to process delicate lines or shapes and the resolution becomes rough or deteriorated by zooming in/out. Because information is processed in pixel unit, file size is relatively large.

(2) Vector

Vector refers to a way to divide one image into several lines and shapes when saving images. This way is proper when expressing simple shapes and objects and it enables to express delicate lines or forms effectively. Zooming in/out does not have any influence on the resolution. However, available colors are limited and flexible works like photograph can not created in this way.

9.3.3 Characteristics of each Graphic File Format

This command is used to open the existing documents. If you perform this command, the dialog box is displayed and you can specify the file to open.

(1) BMP

As a format to save uncompressed bit map images, this file format is used basically in Windows and OS/2. Logo pictures displayed when the window starts or ends or background pictures in the wallpaper are all BMP-type files. Since BMP saves image data inefficiently, a file with larger capacity than the actually needed one is created. Both RGB mode and Indexed mode are supported and BMP-type files can be compressed in RLE compression method. Layers and alpha channels are not supported.

(2) JPG

It is not the file format that can be used only in particular software or platforms rather, as a data encoding method in which data files can be downloaded only after their size is lowered, this method has significant bigger merit than compression loss and is the one used the most when saving image files among other image saving methods. As one of the graphic file formats used the most on the Internet, its compression rate is the most excellent method. To compress image files, this method deletes overlapped information in the frame to compress color static image data. JPG can be applied to any picture regardless of image contents or resolution and it can also make it possible to use sequential decoding and gradual decoding that gradually decodes the entire picture. Unlike GIF format, JPEG maintains all of color information of RGB image. However, even though it has a high compression rate because it conducts loss compression, the quality becomes significantly deteriorated as compression increases. About 11.2MB is required when saving 24 bit color images, however, the file size lowers to 1.4MB if it is compressed in ratio of 8 to1. This format is considered almost as a standard format on the Internet and file size is very low about 1/10 of that of other formats when saving the same number of colors under similar definition.

(3) GIF

As a graphic format developed by CompuServe, a leading PC computer company in the USA, it can be used in various device types for smooth interchange of graphic information between different device types. GIF files lower image capacity by reducing the number of colors so that this file format has remarkable compression rates to obtain high resolution images by reducing important transfer time. When saving, this method sorts 256 colors mostly used in the overall image to create a color index and the entire images are expressed in those 256 colors. Therefore, the file's capacity remarkably decreases. However, as its biggest shortcoming, the available number of colors may vary depending on the version of this format so that up to 256 colors are supported in version 87a. This way is rarely used for image saving today because a format with 256 colors is the smallest one. Therefore, it is mainly used when producing website icons. Gif 89a Format is mostly used for creating moving animation or files with transparent background.

(4) PNG

As an abbreviation word of Portable Network Graphics, it is introduced to support interface functions and increase the compression rate as PEG does. Like GIF, this image format lowers the number of colors to compress images but, unlike GIF which limits up to 256 colors, it can save 16,000,000 colors and provides more excellent compression rates of 10~30% than those of GIF. Basically, it supports true colors, uses no-loss compression and enables to express original images as they are without image deterioration on the web. PNG provides improved functions in the alpha channel related to image's transparency and supports 256 gray colors and 256 colors as well as true colors. This format can be called an emerging web image format after GIF and JPEG. PNG is currently used in Netscape and Explorer.

(5) TIF

TIFF means an image file format with a tag and it is co-developed by Aldus Corp and Microsoft Corp. This file format is used when files are exchanged between application and computer platform and even though most files are saved without being compressed under this format, it reduces file capacity most significantly by using a "no-loss compression method." Basically, it does not depend on OS so that it can describe resolution and compression methods and as a format to preserve image data from black and white to various colors, it is used to preserve dot-expression images. Most of scanner software or graphic software support this TIFF type and it is mainly used for scanning, saving or sending image files expressed in gray scale. It seems passable to think this format is used for image exchange in Macintosh and Window.

(6) WMF

As an abbreviation word of Windows Meta File, this is the format to support the vector method -drawing programs. As a common format of drawing programs, it is supported by Microsoft Office as clip arts. Not only it can enlarge into any size freely without lowering resolution but also this format is supported by many useful programs. This format also expresses graphic information displayed in the window screen with GDI Function.

(7) EMF

As an abbreviation word of Enhanced Meta File, this can provide enhanced 32bit definition even though it is the same as WMF.

9.4 Tag Link

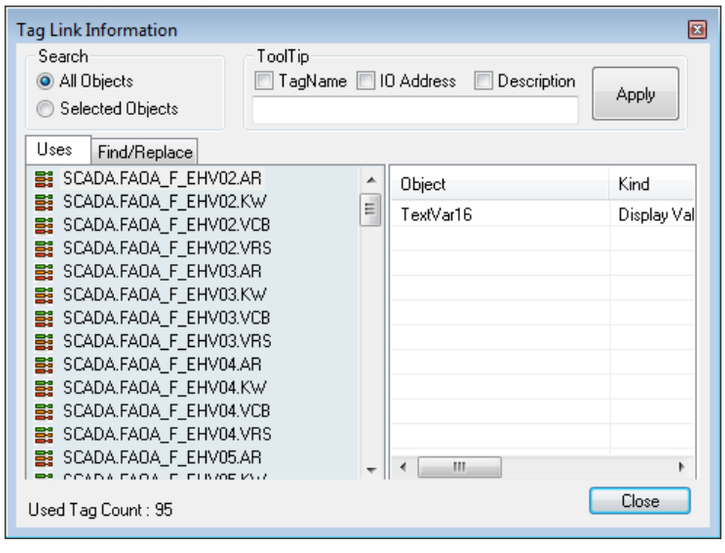
This command is used when finding mapped tags among the entire objects or the selected objects in the page or replacing the found tags with others. The user may define a range to be searched by selecting either 'All Objects' or 'Selected Objects'.

9.4.1 Using Location

The left area of the using location tab indicates the list of tags composed in the object selection. The right area shows the object using each tag and dynamic property list.

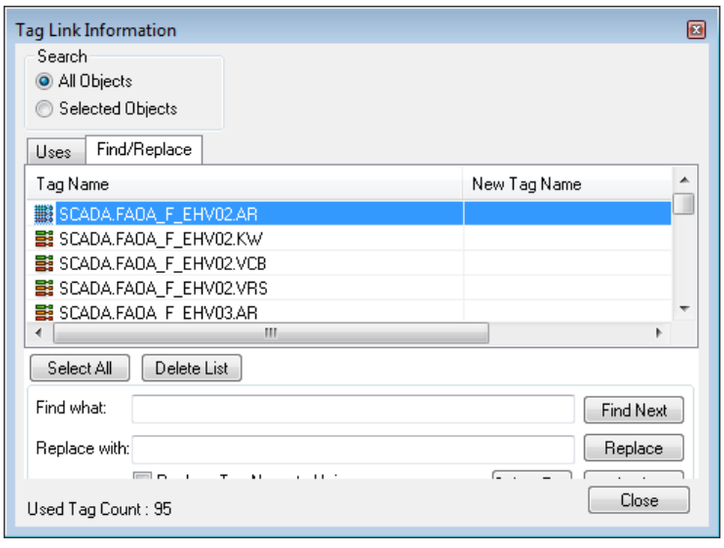
(1) ToolTip

Among objects into which the selected names are mapped, after choosing the desired object, you can batch-apply the tag information and user definition with a tooltip.



9.4.2 Find/Replace

The left area under 'Find/Replace' tab shows the names of the tags organized through selecting objects and it is used when tag link is re-organized for all of the selected objects. Preview is available through single or multiple selection and it is also possible to copy with the clipboard to edit in a Excel file and paste.



(1) Find What

When replacing the selected tag name, directly input a name of the new tag or click 'Select Tag' button and then input it in the 'Select Tag' dialog box.

(2) Select All

All of the tag lists are selected.

(3) Delete List

The selected tag lists are cancelled.

(4) Replace

Click 'Replace' button to check and inspect the planned replace.

(5) Apply

Click 'Apply' button to link the select tag link as seen in the displayed list.

9.5 Library

In addition to the library basically provided by the system, the user may register/delete/edit his/her own library. One of the strong points of this “Library Object” command is that several objects can be selected at the same time to be saved in the library list and at this time, the properties set up in those objects are also saved.

9.5.1 Use Symbol Objects

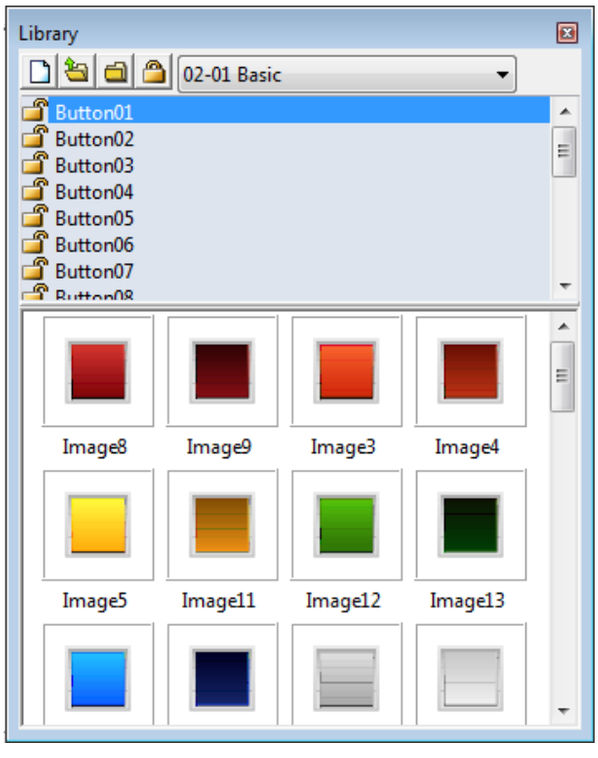
Any object registered in the library can be inserted in the screen. The user may drag any object he/she wants to edit in the library and drop to the page he/she is editing with the left button of the mouse or the user may copy relevant symbols and paste them to the edit page to insert objects. Then, these objects can be reformed to a desired shape through editing.

9.5.2 Register Symbol Objects

In addition to the library basically provided by the system, the user may add his/her own library. To register an object, the user can select an object in the edit page and drag and drop it to the Library dialog box while pressing ‘Alt ‘key or the user can copy the relevant symbol and paste it to a desire location in the Library dialog box.

When creating a new group, the user needs to input a group name. Library data are saved in location ‘InfoU Installation Path\bin\PicLibs’ in file type ‘Group Name.dat.’.

9.5.3 Library Dialog Box



(1) New Name

It is used to create a new library group. If the name of an image to be registered in the library is inputted, the image with the new name is added to the library group.

(2) Open

It is used to open one of existing library groups. The opened library group is added to the list.

(3) Close

The relevant library group is closed. The closed library group is deleted from the list.

(4) Lock

If the relevant library group is locked, any symbol is not added nor deleted from the relevant library group.

(5) Copy/Paste/Delete

It is available to copy, paste or delete symbol objects from the library dialog box.

(6) Preview

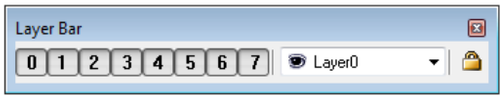
The user may preview symbol objects in icon, name or icon & name type.

9.6 Layer

By using layers such as Show/Hide, Lock and Declutter function in the graphic editor, the user may simplify complicate process screen works into individual object processes. The contents in the process screen can be divided up to eight layers and Layer0 objects are located on the lowest layer while Layer7 objects are located on the top

9.6.1 Layer Settings

In default, all of the layers can be seen and Layer O is activated. Objects can be always added to the activated layer but, they can be quickly moved to other layer. The user may change to assign objects to layers by using the Object Property screen. If the user needs to move multiple objects in different layers to one layer, he/she may use multiple-selection to move them quickly. To change active layers, the user may use the Layer combo box.



9.6.2 Layer Lock/Unlock

The layer lock function makes selecting and editing objects either possible or impossible. By using Lock button, the user may lock the relevant layer and use it as background image.

9.6.3 Layer Show/Hide

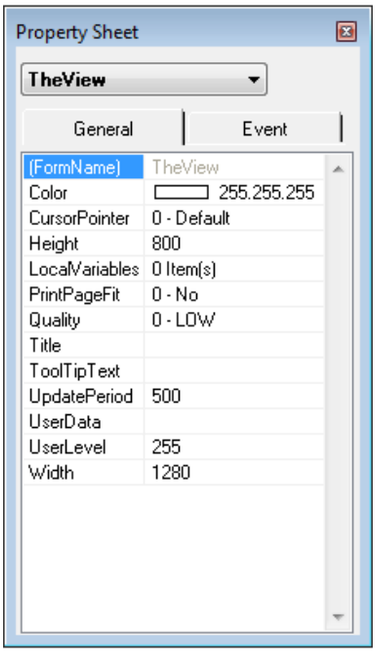
By using the Layer Palette, the user may show or hide the relevant layer individually. That is, it is possible to show/hide the objects assigned to the relevant layer.

9.7 Object Property

9.7.1 General

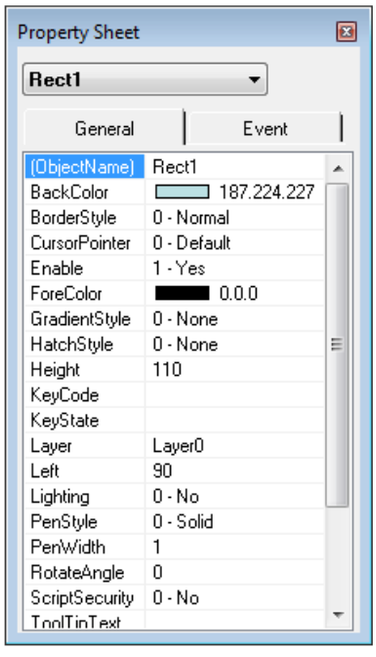
Static characteristics of each object such as page and object size, shape, location and color can be defined. After selecting objects on the screen, you can change the same properties in a lump.

(1) Page Property



Property	Descriptions
(FormName)	Name of page object
Color	Page background color
CursorPointer	Cursor shape when the mouse is on an object
Height	Page height
Width	Page width
PrintPageFit	Designate a rate to align print paper and page
Quality	Designate the quality level of an object to be drawn
UserData	Save data defined by the user
Title	Page name
ToolTipText	Tooltip when the mouse is on an object
UpdatePeriod	Screen update period during Runtime
LocalVariables	Designate variables used in the page
UserLevel	Specify the user class on object actions.

(2) Object Property

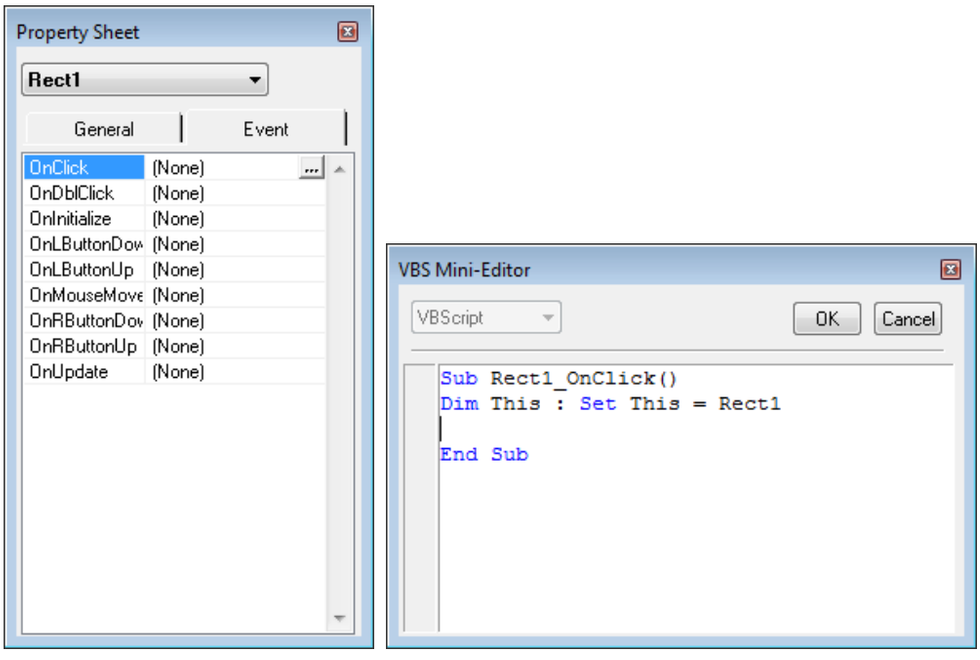


Property	Descriptions
(ObjectName)	Object name
Left	Left location of an object
Top	Top location of an object
Width	Object width
Height	Object height
CursorPointer	Cursor shape when the mouse is on an object
ToolTipText	Tooltip when the mouse is on an object
Enable	Designate to assign event handling of an object
Visible	Designate whether to show an object
UserData	Save data defined by the user
BackColor	Object background color
ForeColor	Object line or letter color
Font	Designate font name, style, size
Layer	Designate layer where an object exists
RotateAngle	Designate a rotation angle of an object
PenWidth	Designate line width
HatchStyle	Designate a hatch pattern to fill
BorderStyle	Normal, 3D, Sunken or Raised.
GradientStyle	Designate a gradient pattern to fill
PenStyle	Designate a line style
Arrowhead	Designate arrow direction
ArrowheadHeight	Designate arrow size

Text	Designate text to be displayed
Alignment	Designate text alignment
Angle1	Designate the start angle
Angle2	Designate the end angle
BorderColor	Object border color
ImageName	Name and format type of the displayed image
ImageInclude	Designate a type to save images
KeyCode	Dynamic property- Specify the shortcut key to click
KeyState	Dynamic property- Specify the secondary shortcut key to click
TransparentColor	Designate the transparent Color
ButtonStyle	Designate the push button style
Layout	How to specify the image size
FrameIndex	Specify the animation Gif frame
ImageIndex	Designate the multi-image index
UserLevel	Specify the user class on object actions
Lighting	Customized lighting function
ScriptSecurity	It is the function to specify the script security for the object. Once you apply the security function, it will never be canceled so you need to keep the original all the time.

9.7.2 Event Property

The user may use scripts when designating complicate dynamic properties defined by the user or executing functions other than the predefined movements. 'Script Engine' provided by Microsoft Corp is used and it is very similar to visual basic programming. If 'Event' tab is selected in the Object Sheet and an event to be edited is selected, VBS Mini-Editor, a scripter editor, appears.



Event	Descriptions
OnInitialize	Event that takes place when the relevant page is open(For setting-up)
OnTerminate	Event that takes place when the relevant page is closed(For clean-up)
OnClick	Event that takes place when the user clicks on an object
OnDblClick	Event that takes place when the user double clicks on an object
OnLButtonDown	Event that takes place when the user clicks the left of the mouse on an object
OnRButtonDown	Event that takes place when the user clicks the right of the mouse on an object
OnMouseMove	Event that takes place when the user moves the mouse
OnLButtonUp	Event that takes place when the user moves up an object with the left mouse
OnRButtonUp	Event that takes place when the user moves up an object with the right mouse
OnUpdate	Event that takes place when screen updates regularly.
OnSetFocus	Event that takes place when enable the page
OnKillFocus	Event that takes place when disable the page

9.8 Define Dynamic Properties

Dynamic Properties are used when setting up dynamic properties and action functions of the relevant object. First, select an object and execute this command to show a dialog box to set up its dynamic properties. The displayed dialog box allows the user to set up dynamic properties and action functions of the relevant object.

9.8.1 Horizontal Size

This is a function to change the size of objects in the horizontal direction according to tag values.

(1) Tag Name

Input a tag name or expression that will be a baseline of horizontal size. A tag name can be designated with 'Search' (...) on the right.

(2) Direction

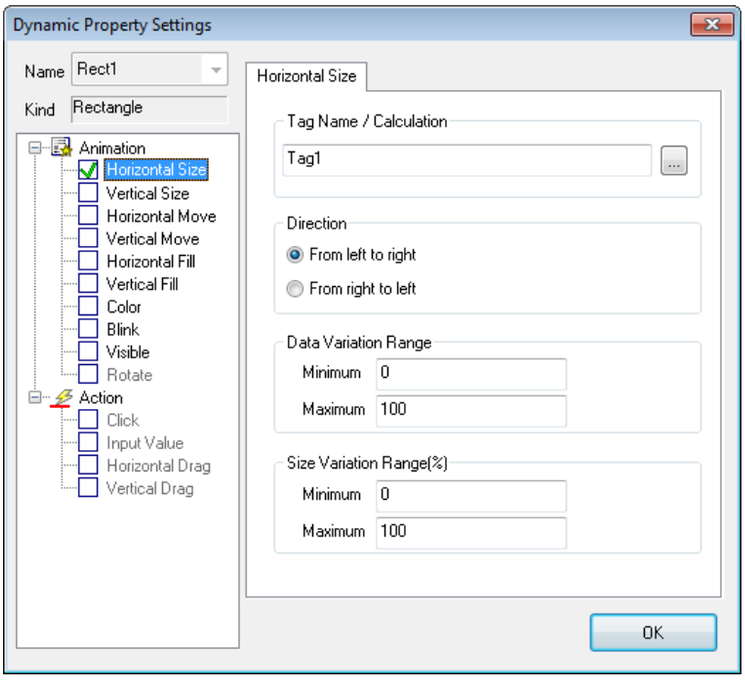
Set up a base location and direction of change in size.

(3) Data Variation Range

Set up a range of data to be referred when changed. The minimum value cannot be bigger than the maximum one.

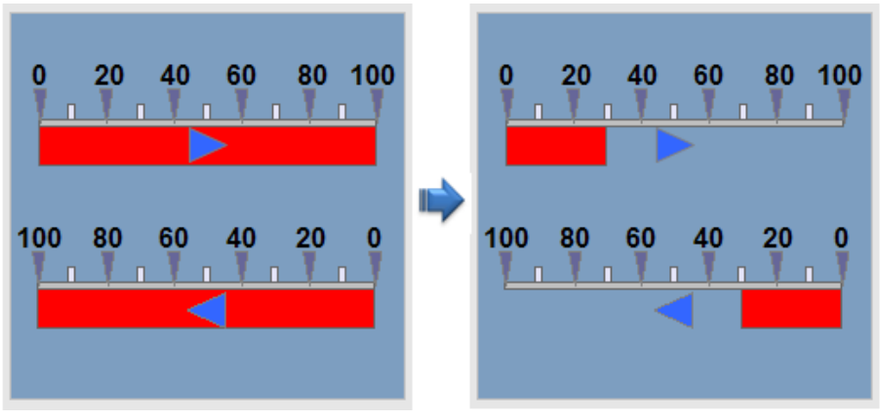
(4) Size Variation Range

Set up a range of size to be changed. The minimum size cannot be bigger than the maximum one.



(5) Sample examples

Action screen when changing the tag value: 100→30



9.8.2 Vertical Size

This is a function to change the size of objects in the vertical direction according to tag values.

(1) Tag Name

Input a tag name or expression that will be a baseline of vertical size. A tag name can be designated with 'Search' (...).

(2) Direction

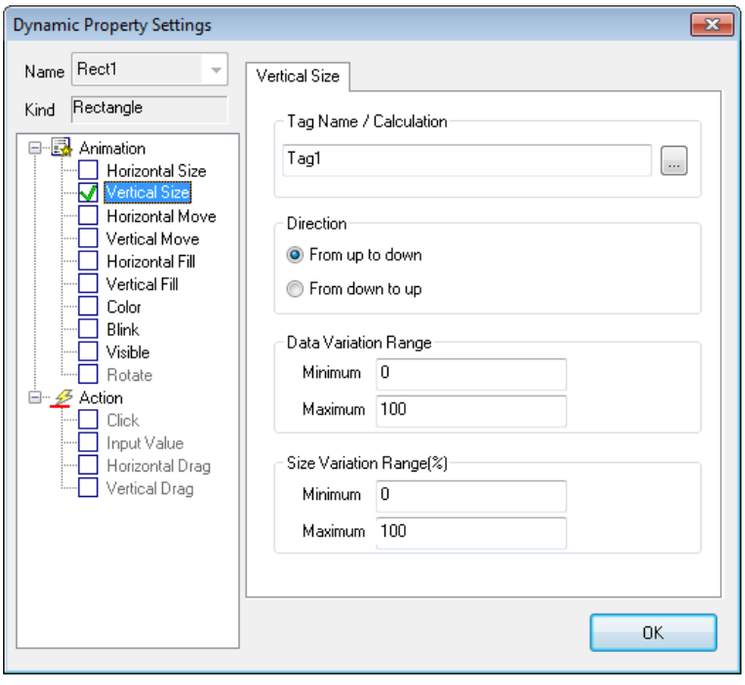
Set up a base location and direction of change in size.

(3) Data Variation Range

Set up a range of data to be referred when changed. The minimum value cannot be bigger than the maximum one.

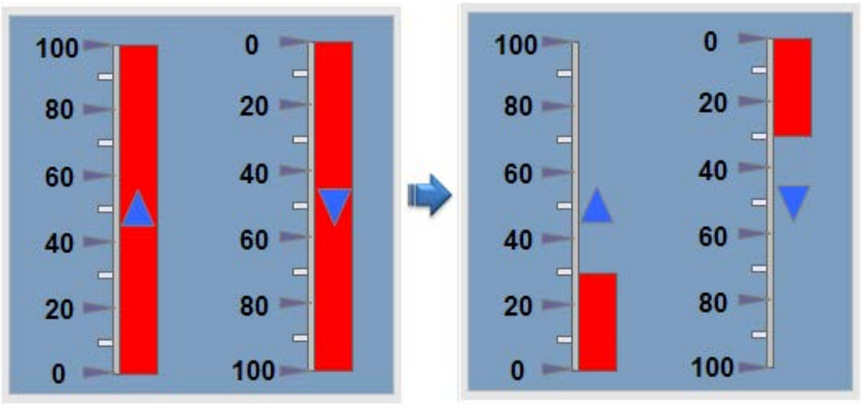
(4) Size Variation Range

Set up a range of size to be changed. The minimum value cannot be bigger than the maximum one.



(5) Sample examples

Action screen when changing the tag value: 100→30



9.8.3 Horizontal Move

This is a function to move the object in the horizontal direction according to the designated tag value.

(1) Tag Name

Input a tag name or expression that will be a baseline of horizontal move. A tag name can be designated with 'Search' (...) on the right.

(2) Direction

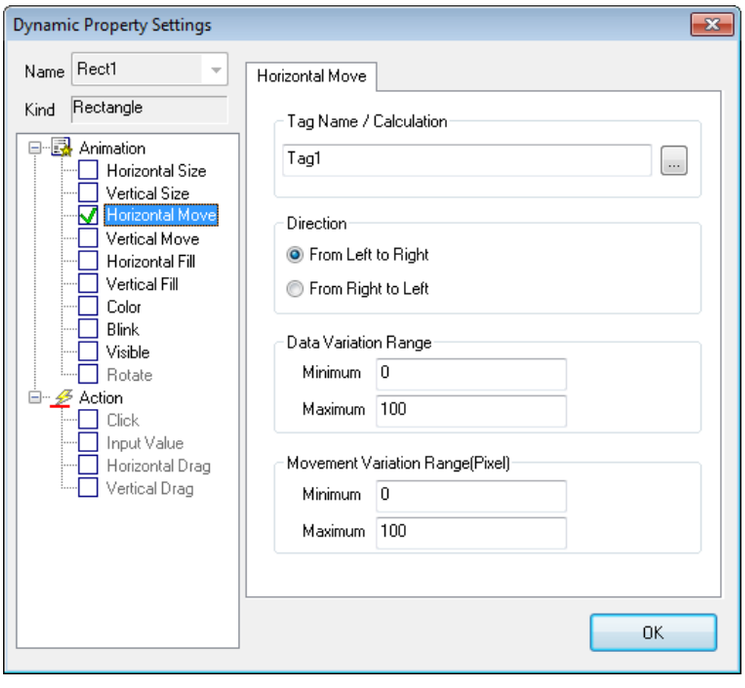
Set up a base location and direction of change in move.

(3) Data Variation Range

Set up a range of data to be referred when changed. The minimum value cannot be bigger than the maximum one.

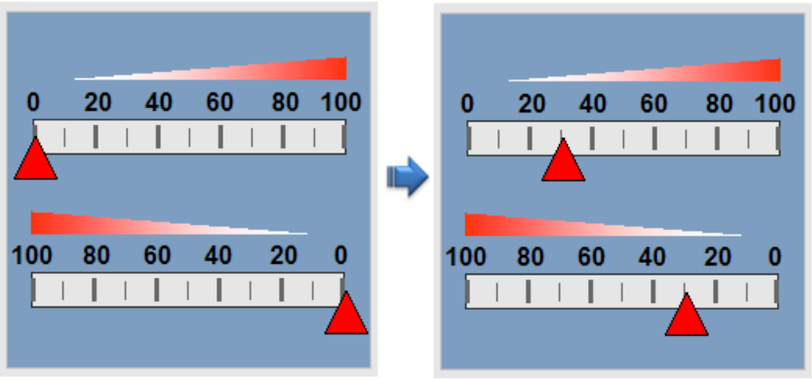
(4) Move Variation Range

Set up a range of move to be changed. The minimum move cannot be bigger than the maximum one.



(5) Sample examples

Action screen when changing the tag value: 0→30



9.8.4 Vertical Move

This is a function to move the object in the vertical direction according to the designated tag value.

(1) Tag Name

Input a tag name or expression that will be a baseline of vertical move. A tag name can be designated with 'Search' (...) on the right.

(2) Direction

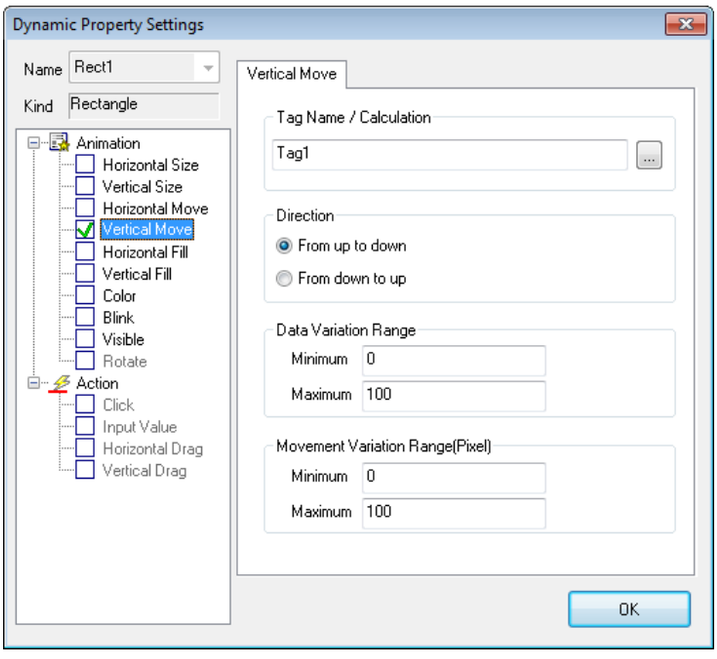
Set up a base location and direction of change in move.

(3) Data Variation Range

Set up a range of data to be referred when changed. The minimum value cannot be bigger than the maximum one.

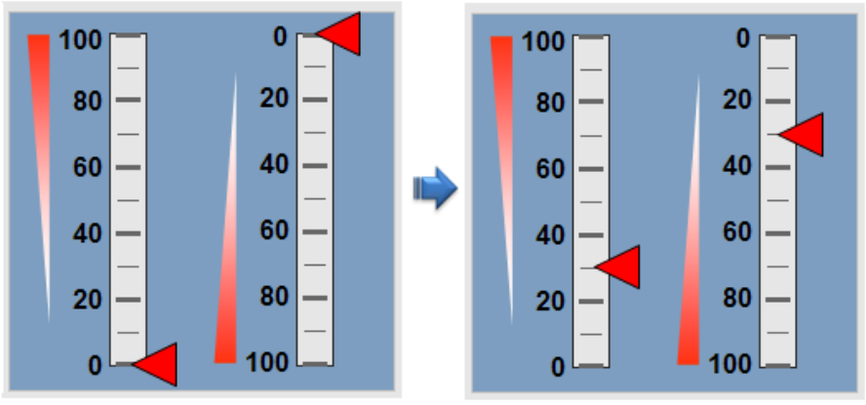
(4) Move Variation Range

Set up a range of move to be changed. The minimum move cannot be bigger than the maximum one.



(5) Sample examples

Action screen when changing the tag value: 0→30



9.8.5 Horizontal Fill

This is a function to fill in the object in the horizontal direction according to the designated tag value.

(1) Tag Name

Input a tag name or expression that will be a baseline of horizontal fill. A tag name can be designated with 'Search' (...) on the right.

(2) Direction

Set up a base location and direction of change in fill.

(3) Data Variation Range

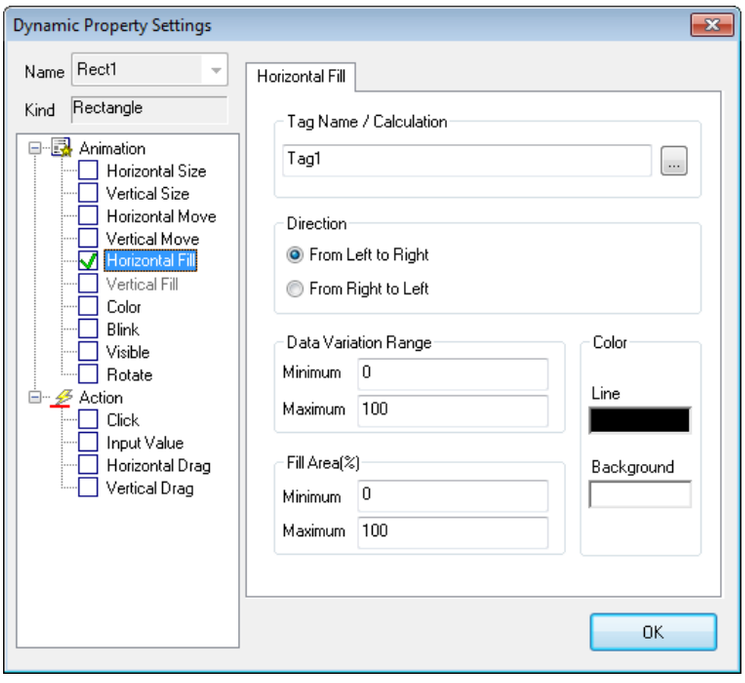
Set up a range of data to be referred when changed. The minimum value cannot be bigger than the maximum one.

(4) Fill Area (%)

Set up an area to fill when changed. The start area cannot be bigger than the maximum one.

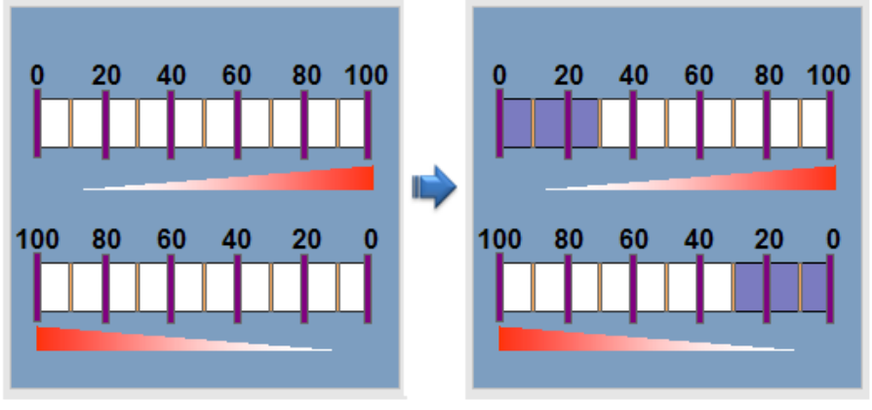
(5) Color

Set up colors of line and background when filling.



(6) Sample examples

Action screen when changing the tag value: 0→30



9.8.6 Vertical Fill

This is a function to fill in the object in the vertical direction according to the designated tag value.

(1) Tag Name

Input a tag name or expression that will be a baseline of vertical fill. A tag name can be designated with 'Search' (...) on the right.

(2) Direction

Set up a base location and direction of change in fill.

(3) Data Variation Range

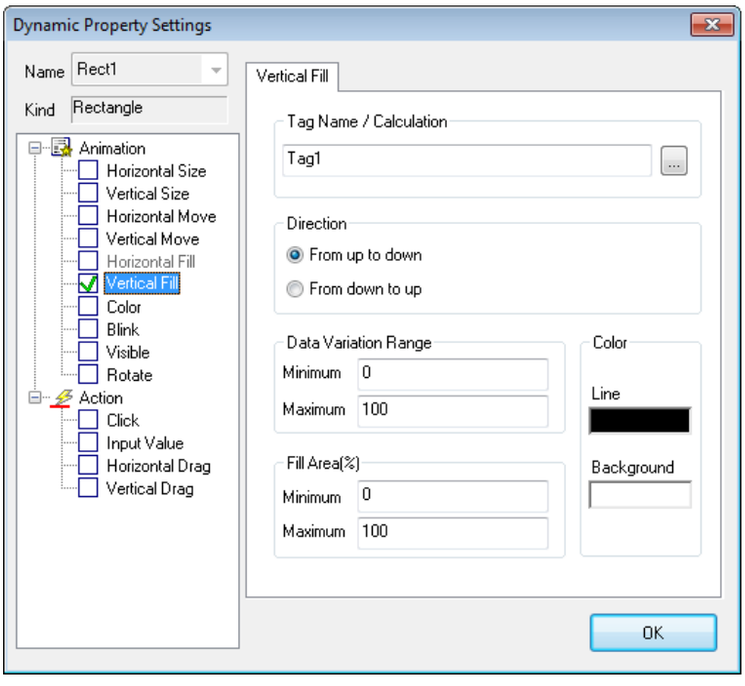
Set up a range of data to be referred when changed. The minimum value cannot be bigger than the maximum one.

(4) Fill Area (%)

Set up an area to fill when changed. The start area cannot be bigger than the maximum one.

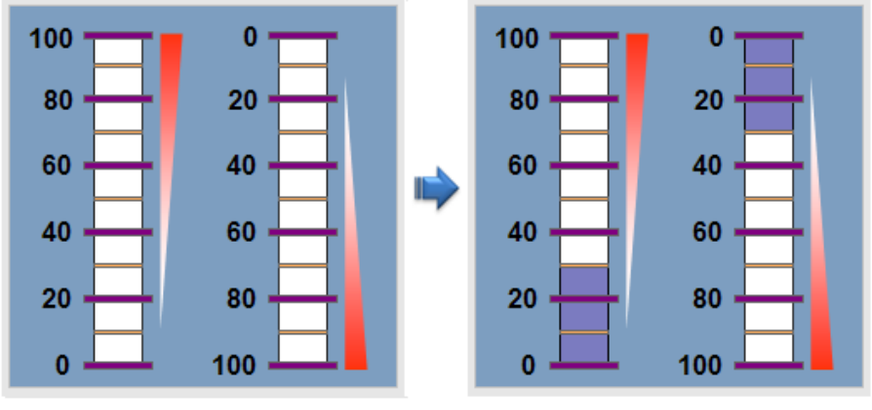
(5) Color

Set up colors of line and background when filling.



(6) Sample examples

Action screen when changing the tag value: 0→30



9.8.7 Color

This is a function to change the object in the prescribed color according to the inputted tag value.

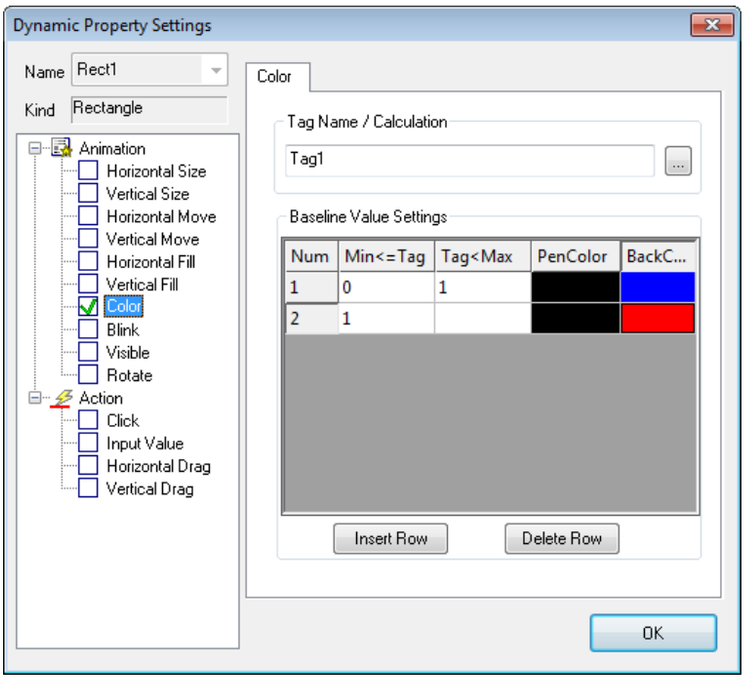
(1) Tag Name

Input a tag name or expression that will be a baseline of color change. A tag name can be designated with 'Search' (...) on the right..

(2) Baseline Value Settings

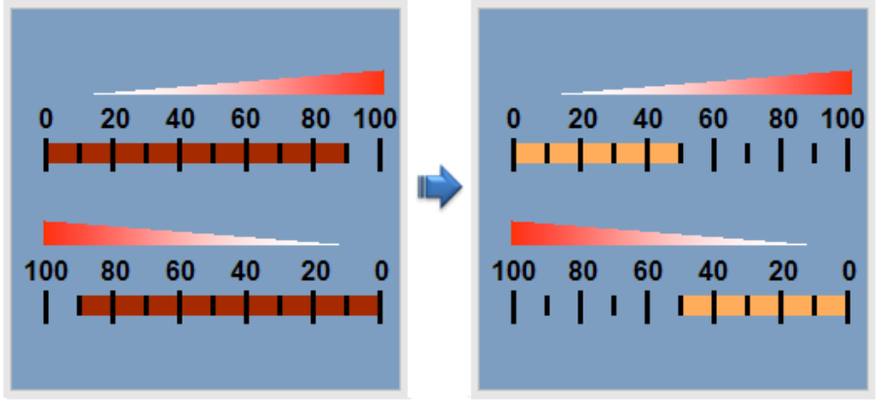
When colors are changed as new rows are inserted to grids, input the minimum and maximum value of the inputted tag or

expression and set up colors of line and background to display under the relevant condition.



(3) Sample examples

Action screen when changing the tag value: 90→50



9.8.8 Blink

This is a function to blink the object in the specified interval according to the tag status.

(1) Tag Name

Input a tag name or expression that will be a baseline of blink. A tag name can be designated with 'Search' (...) on the right.

(2) Condition

1) True

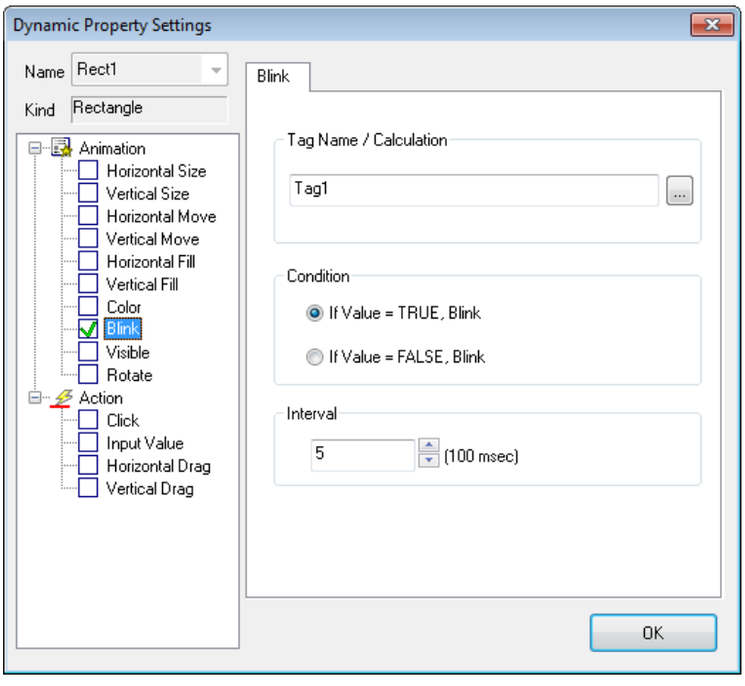
If the tag value designated above is true, blink the object. If false, do not blink it.

2) False

If the tag value designated above is false, blink the object. If true, do not blink it..

(3) Interval

Select a blink interval. Its unit is 100msec. That is, if 1 is inputted, it blinks every 100msec and if 10 is inputted, it blinks every second. Input one value between 1 and 999.



(4) Sample examples

Action screen when the tag value is TRUE



9.8.9 Visible

This is a function either to show or hide the object according to the input tag value.

(1) Tag Name

Input a tag name or calculation that will be a baseline of pop in/out. You can select tag names by clicking the search icon (…) at right side of Tag Name / Calculation.

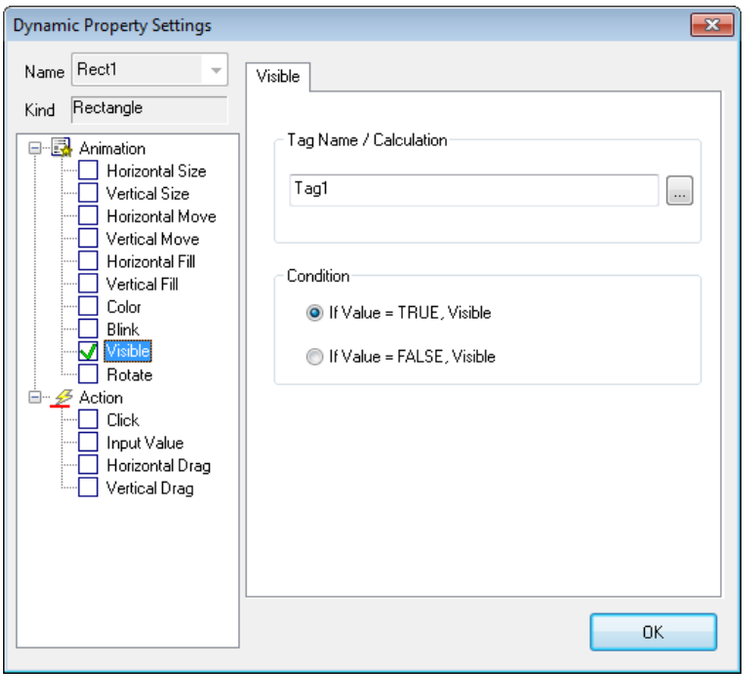
(2) Condition

1) True

If the tag value designated above is true, pop in/out the object. If false, do not pop in/out it.

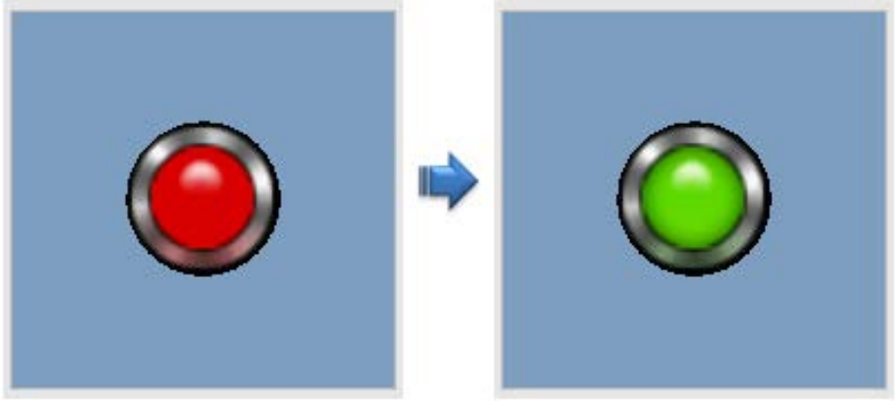
2) False

If the tag value designated above is false, pop in/out the object. If true, do not pop in/out it.



(3) Sample examples

Action screen when changing the tag value: TRUE→FALSE



9.8.10 Rotate

This is a function to rotate the object by the designated angle degree according to the input tag value.

(1) Tag Name

Input a tag name or expression that will be a baseline of rotation. A tag name can be designated with 'Search' (...) on the right.

(2) Direction

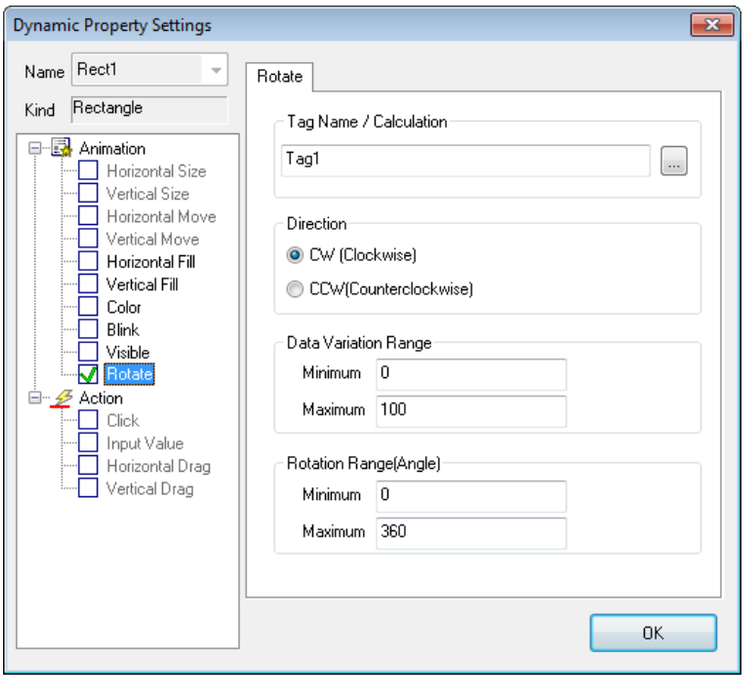
Set up a rotation direction when rotation is changed.

(3) Data Variation Range

Set up a range of data to be referred when changed. The minimum value can not be bigger than the maximum one.

(4) Rotation Range (Angle)

Set up a range of rotation to be changed. The minimum angle can not be bigger than the maximum one.



(5) Sample examples

Action screen when changing the tag value: 0→90



9.8.11 Display Value

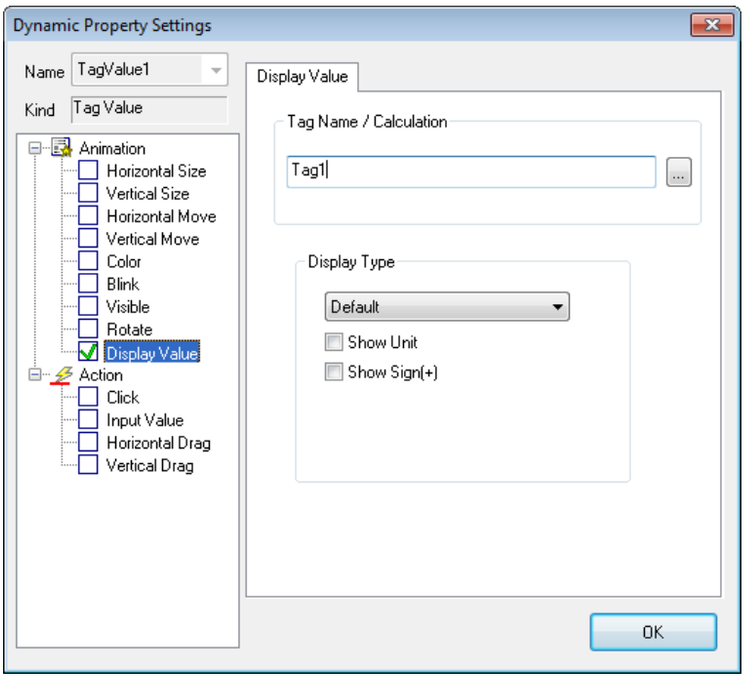
It is used to create an object marked with its tag value.

(1) Tag Name

Input a tag name or expression. A tag name can be designated with 'Search' (...) on the right.

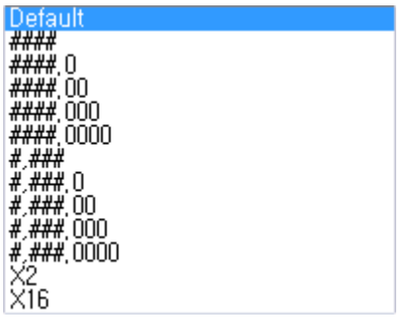
(2) Display Type

You can preview the format displaying the value on the screen. In the case of default setting, the value is applied in the InfoU engineering. In the case of the digital tag, the On/Off string is displayed and for the decimal places of the analog tag, the number of digits set in the InfoU engineering is applied. If you want to display a unit and a sign on the screen, just put a check 'Show Unit' and 'Display Sign'.

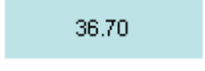


(3) Sample examples

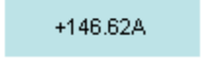
The value is basically set as Default. You can specify the number of digits and decimal places for the tag value as shown below. When the value is set as 'Default', the value sent from the tag is displayed as it is.



1) (When selection ####.00)



2) (When selecting ####.00 and putting a check 'Show Unit' and 'Display Sign')



3) (When selecting 'Default' and putting a check 'Show Unit' and 'Display Sign')



9.8.12 Display String

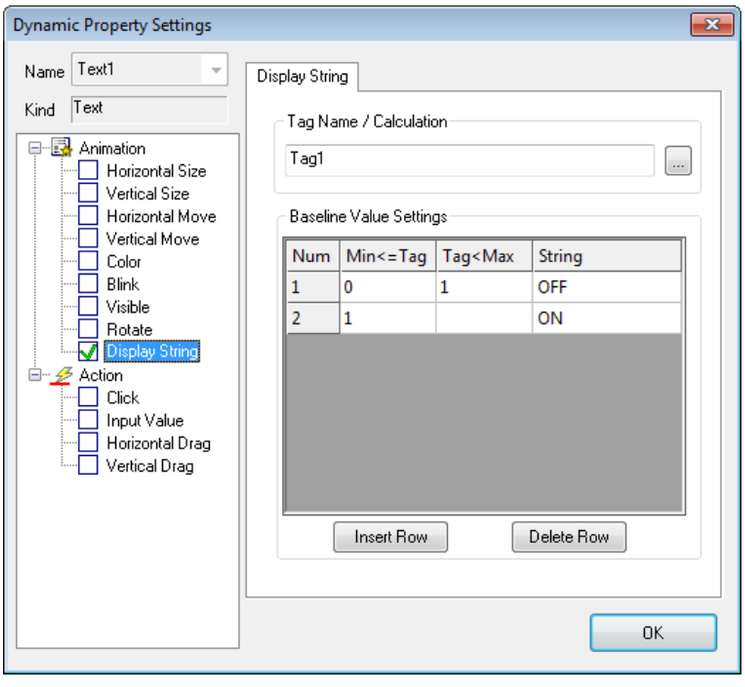
A function to change the object to the prescribed string according to the inputted tag value.

(1) Tag Name

Input a tag name or expression that will be a baseline to change strings. A tag name can be designated with 'Search' (...) on the right.

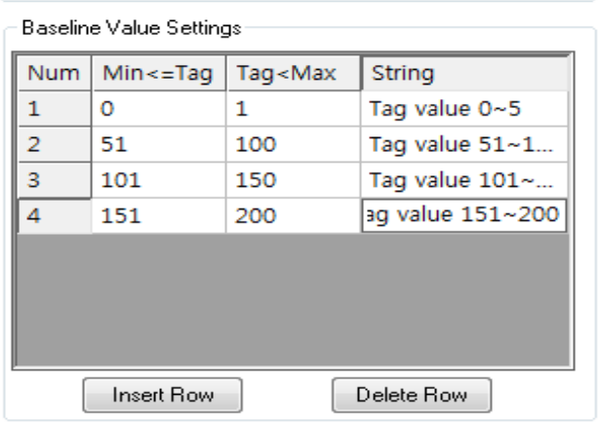
(2) Baseline Value Settings

When strings are changed as new rows are inserted to grids, input the minimum and maximum value of the inputted tag or expression and set up strings to display under the relevant condition.



(3) Sample examples

If you perform the runtime, the displayed string will change depending on the tag value.



Tag Value 0 ~ 5	Tag Value 51 ~ 100
0.5667	61.5667
Tag Value 101 ~ 150	Tag Value 151 ~ 200
111.5667	161.5667

9.8.13 GIF Animation

This is a function to execute GIF animation according to the inputted tag values.

(1) Tag Name

Input a tag name or expression that will be a baseline to execute GIF animation. A tag name can be designated with 'Search' (...) on the right.

(2) Condition

1) True

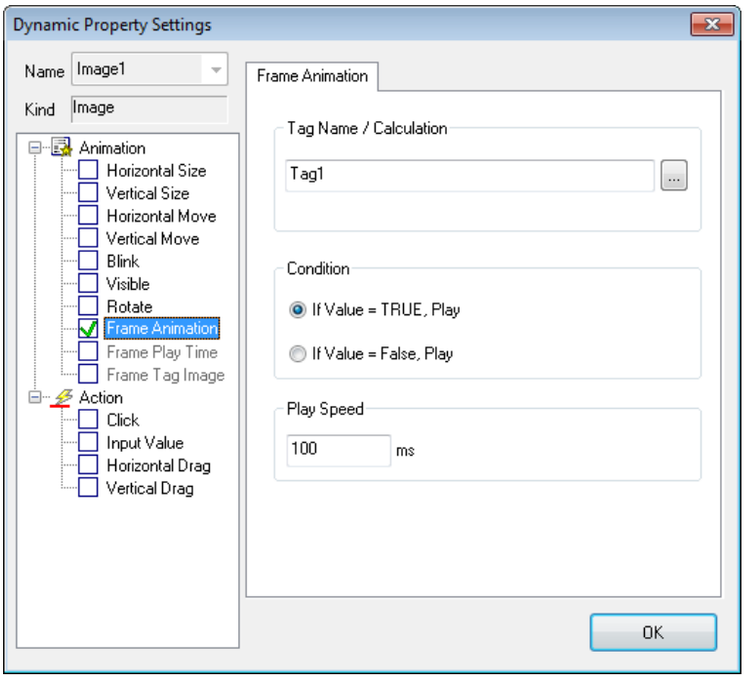
If the tag value designated above is true, execute GIF animation. If false, stop executing GIF animation.

2) False

If the tag value designated above is false, execute GIF animation. If true, stop executing GIF animation.

(3) Play Speed

Designate a play speed between frames.



(4) Sample examples

Action screen when the tag value is TRUE



9.8.14 GIF Play Time

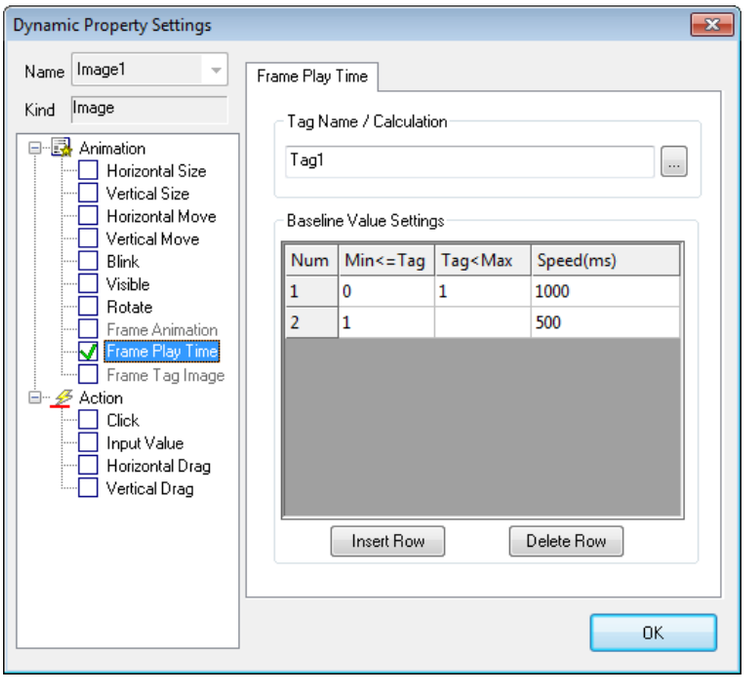
A function to designate a play speed of each GIF animation frame according to the inputted tag value.

(1) Tag Name

Input a tag name or expression that will be a baseline of GIF play time. A tag name can be designated with 'Search' (...) on the right.

(2) Baseline Value Settings

When strings are changed as new rows are inserted to grids, input the minimum and maximum value of the tag or expression that will be a baseline of conversion and set up a play speed under the relevant condition.



9.8.15 GIF Tag Image

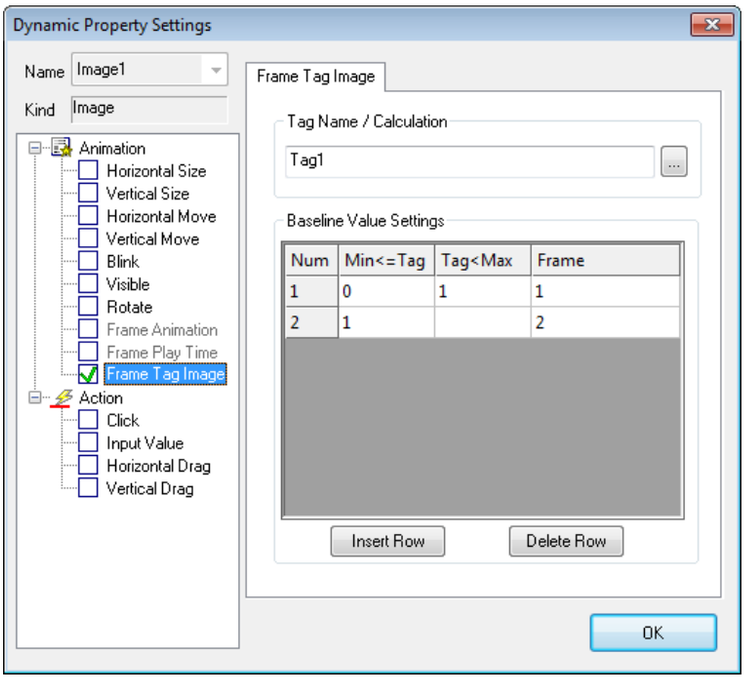
A function to show each GIF animation frame according to the inputted tag value.

(1) Tag Name

Input a tag name or expression that will be a baseline of showing each GIF animation frame. A tag name can be designated with 'Search' (...) on the right.

(2) Baseline Value Settings

When strings are changed as new rows are inserted to grids, input the minimum and maximum value of the tag or expression that will be a baseline of conversion and set up a frame to show under the relevant condition.



9.8.16 Lamp

It shows the On image or the Off image depending on the input tag value.

(1) Tag Name

Input a tag name or calculation that will be a baseline of pop in/out. You can select tag names by clicking the search icon (…) at right side of Tag Name / Calculation.

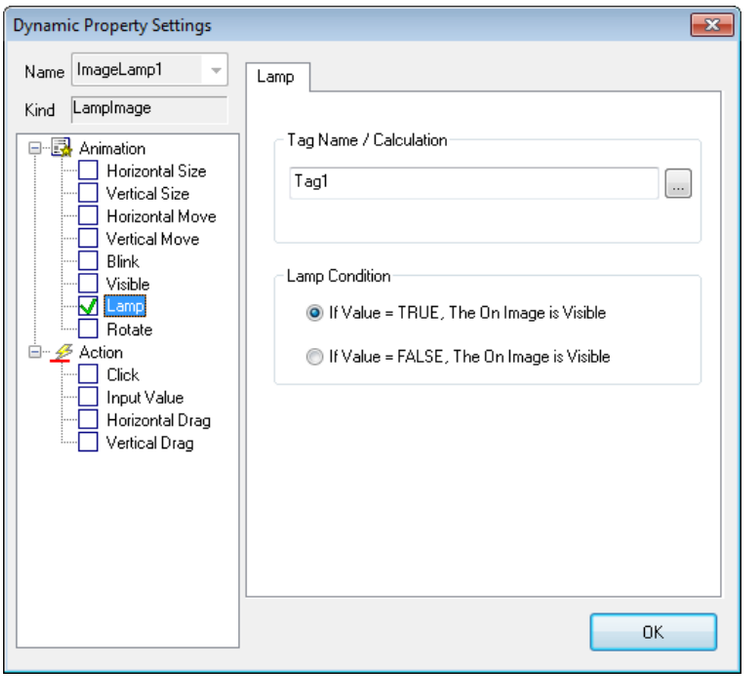
(2) Lamp Condition

1) True

If the above specified tag value is True, the On image will be displayed; If it is False, the Off image will be displayed.

2) False

If the above specified tag value is False, the On image will be displayed; If it is True, the Off image will be displayed.

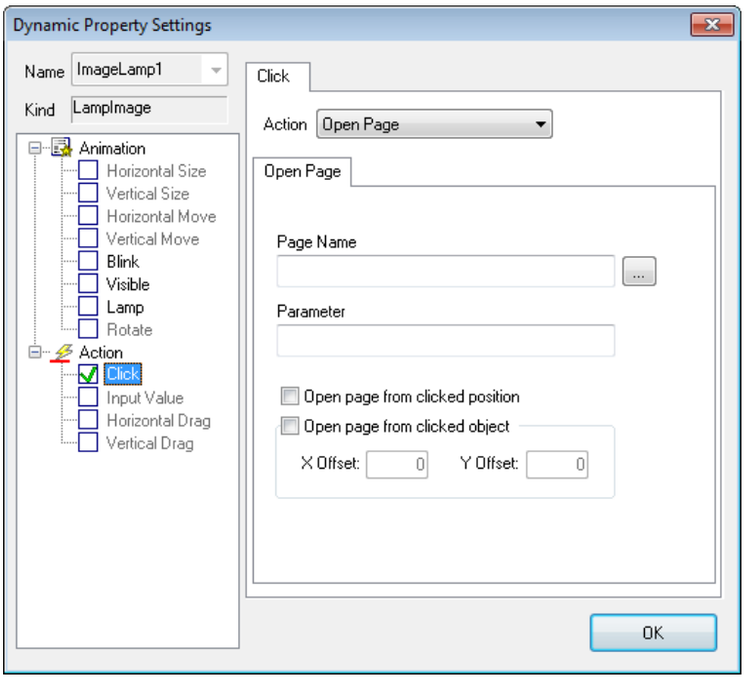


9.8.17 Click

This is a control function to execute the defined actions when clicking the object with the mouse or keyboard.

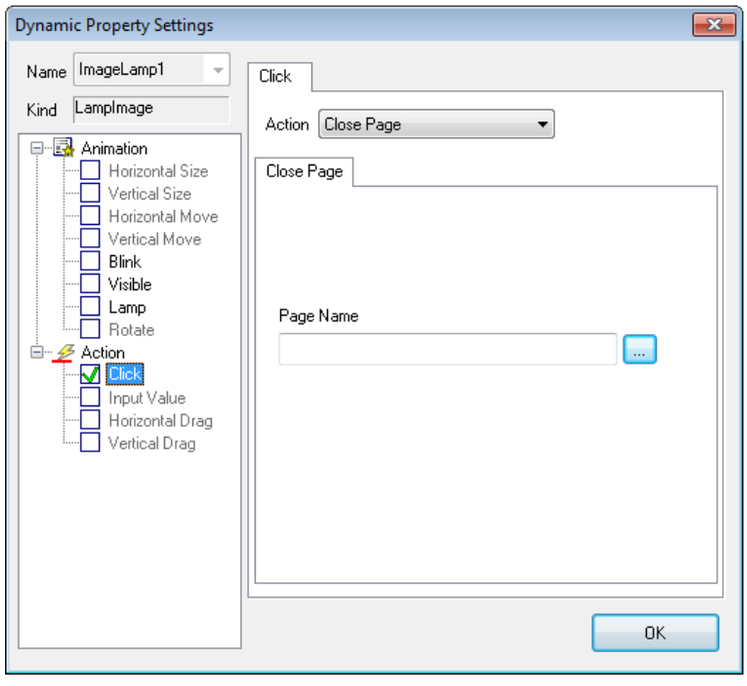
(1) Open Page

The user may directly input a name of the page he/she wants to open or use 'Search' (...) button to move the folder that has the page he/she wants to open and select the relevant page. When opening the page, the user can deliver parameter and open the page from the clicked position.



(2) Close Page

The user may directly input a name of the page he/she wants to close or use 'Search' (...) button to move the folder that has the page he/she wants to close and then, select the relevant page.



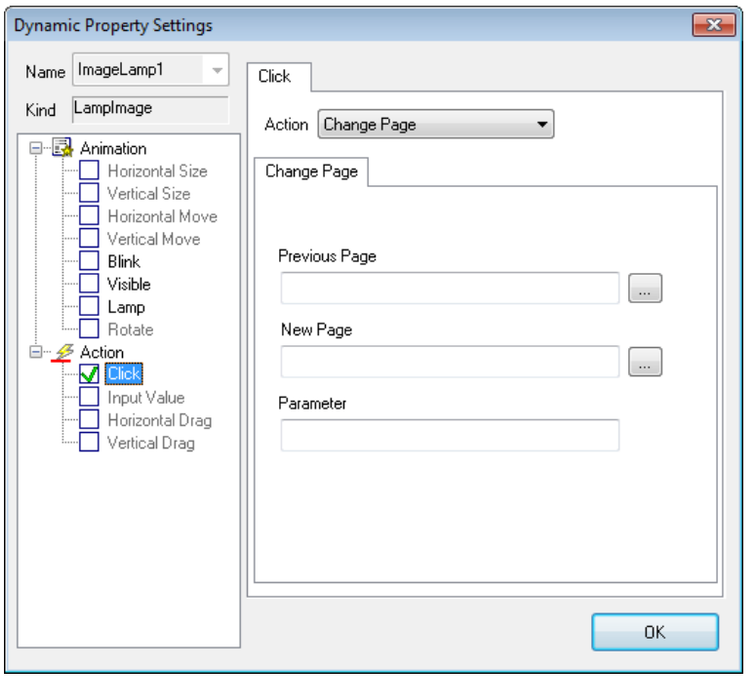
(3) Page Change

1) Previous Page

The user may directly input a name of the page to be closed or use 'Search' (...) button to move the folder that has the page and select the relevant page.

2) New Page

The user may directly input a name of the page to be newly displayed or use 'Search' (...) button to move the folder that has the page and select the relevant page.



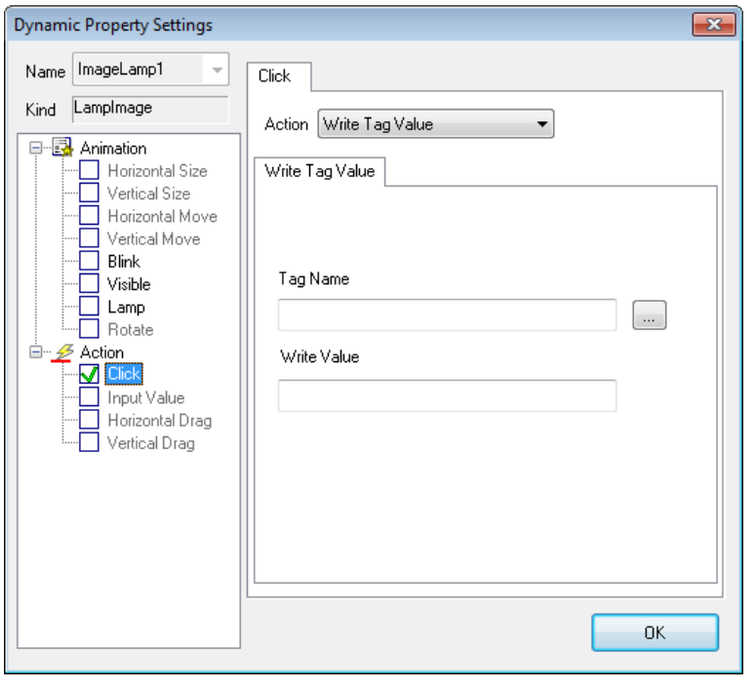
(4) Write Tag Value

1) Tag Name

Directly input a tag name or use 'Search' (...) on the right to designate a tag name.

2) Write Value

Input a value to write.



(5) Write Digital Value

1) Tag Name

Directly input a digital tag name or use 'Search' (...) on the right to designate a tag name.

2) Basic

'Set' designates the value of the relevant digital tag as 1. 'Reset' designates the value as 0. 'Toggle' compares the value of the current digital tag and designates the value as 1 if the result is 0 and the value as 0 if the result is 1.

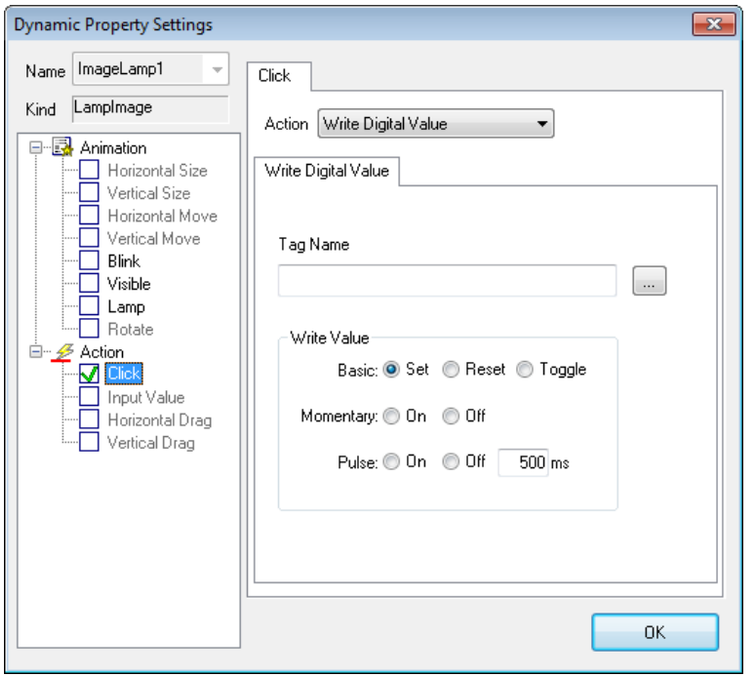
3) Momentory

In case of the On image, the digital tag value is specified as 1 in Mouse Down; 0 in Mouse Up.

In case of the Off image, the digital tag value is specified as 0 in Mouse Down; 1 in Mouse Up.

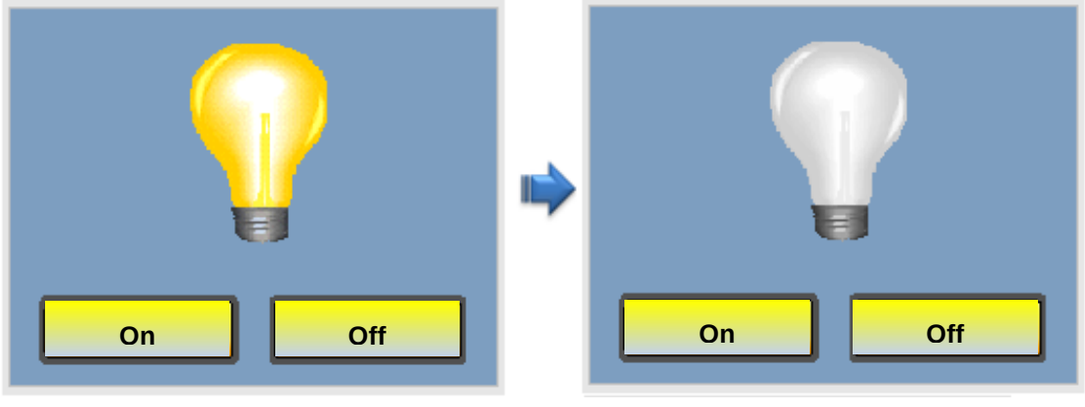
4) Pulse

In case of the On image, the digital tag value is specified as 1 and after the fixed time passes, it is specified as 0. In case of the Off image, the digital tag value is specified as 0 and after the fixed time passes, it is specified as 1.



5) Sample examples

Action screen when changing the tag value: TRUE→FALSE

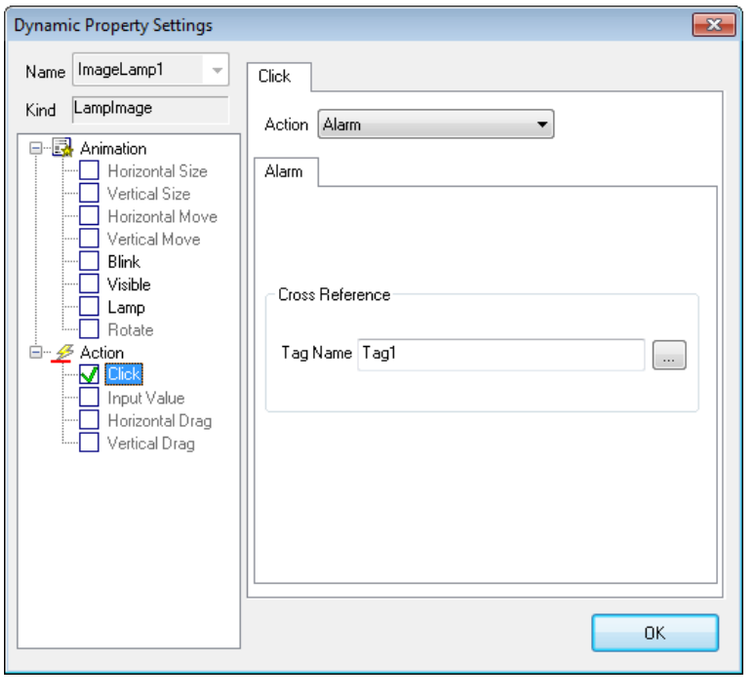


(6) Alarm

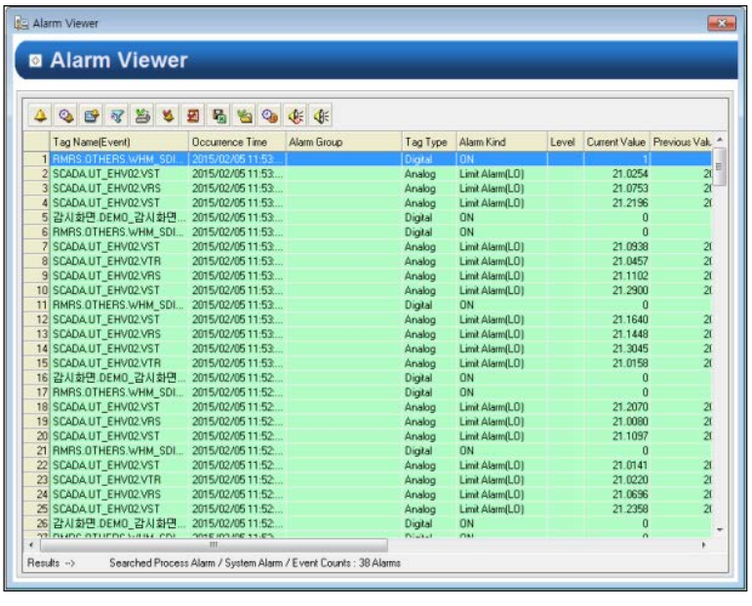
When you click the object, the system alarm page defined in the InfoU will be open. If you enter the tag name, the cross reference will be executed when opening the page.

1) Tag Name

You can enter the tag name directly or specify the tag name with the search (...) button on the right side.



2) System alarm viewer



(7) Report

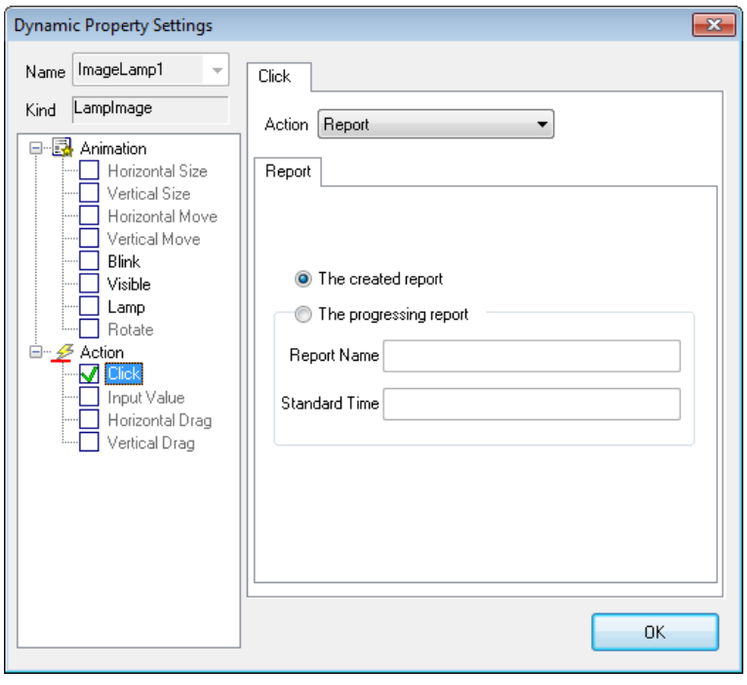
Click the object to run 'Report Settings'.

1) The created report

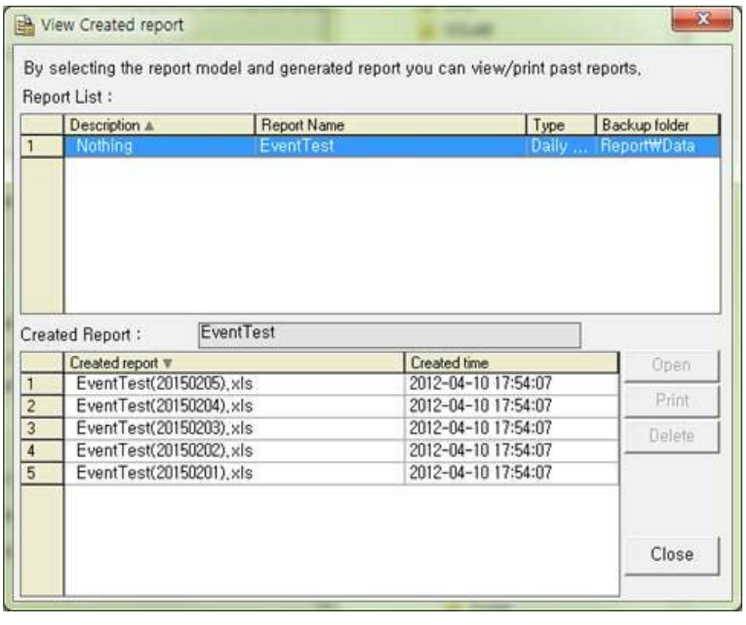
The created 'Report Settings' window is displayed.

2) The progressing report

The ongoing 'Report Settings' window is displayed. You need to set the report name and reference time for retrieval.



3) Report settings



(8) Realtime Trend

If you click the object, the system Realtime Trend page defined in the InfoU will be open.

1) Channel List

Among the channel properties, you can input the tag name, min. and max. value. If you put a check [Channels Reset], the previously saved channel list is reset and the list set in the current setting window only will be applied. If you enter -999 for the min. and max. value, the min, and max. value of the DB tag will be automatically mapped; if you enter 0 or " ", the auto-scale will be set automatically.

2) MultiTrend

It sets to determine whether displaying the trend as the multi-trend style.

3) Display Window

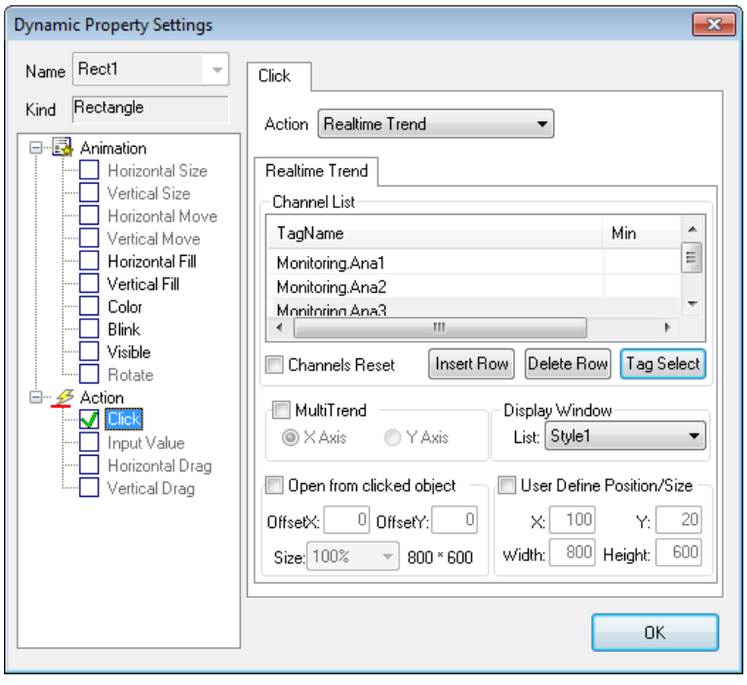
Select the page style to apply.

4) Open from clicked object

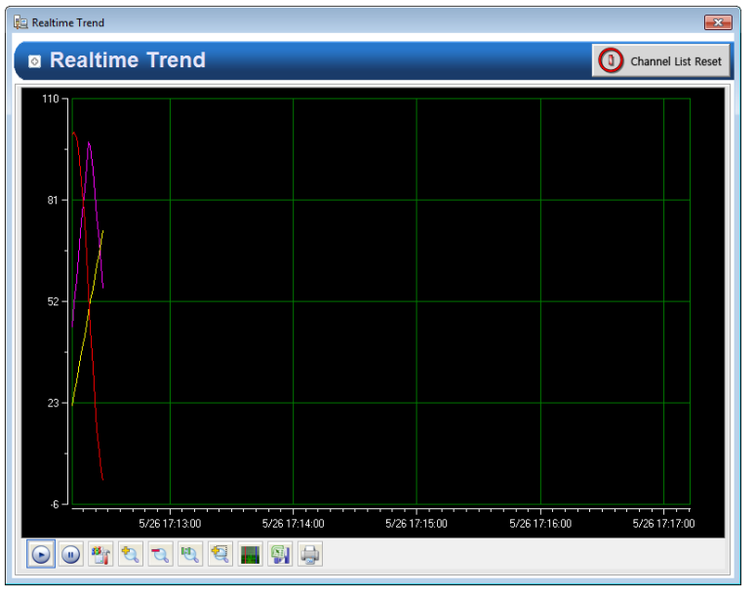
[Open from clicked object] the page will be opened in the position of the relevant object. You can specify X, Y offset and the size.

5) User Define Position/Size

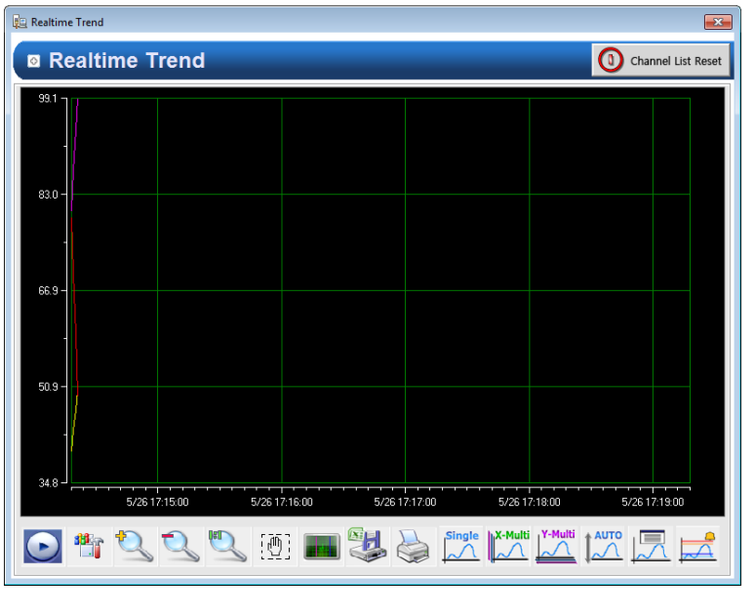
[User Define Position/Size] A user can directly specify the position and size of the page.



6) Display Window- Style1



7) Display Window - Style2



(9) Archive Trend

If you click the object, the system Archive Trend page defined in the InfoU will be open.

1) Channel List

Among the channel properties, you can input the tag name, min. and max. value. If you put a check [Channels Reset], the previously saved channel list is reset and the list set in the current setting window only will be applied. If you enter -999 for the min. and max. value, the min, and max. value of the DB tag will be automatically mapped; if you enter 0 or " ", the auto-scale will be set automatically.

2) MultiTrend

It sets to determine whether displaying the trend as the multi-trend style.

3) Display Window

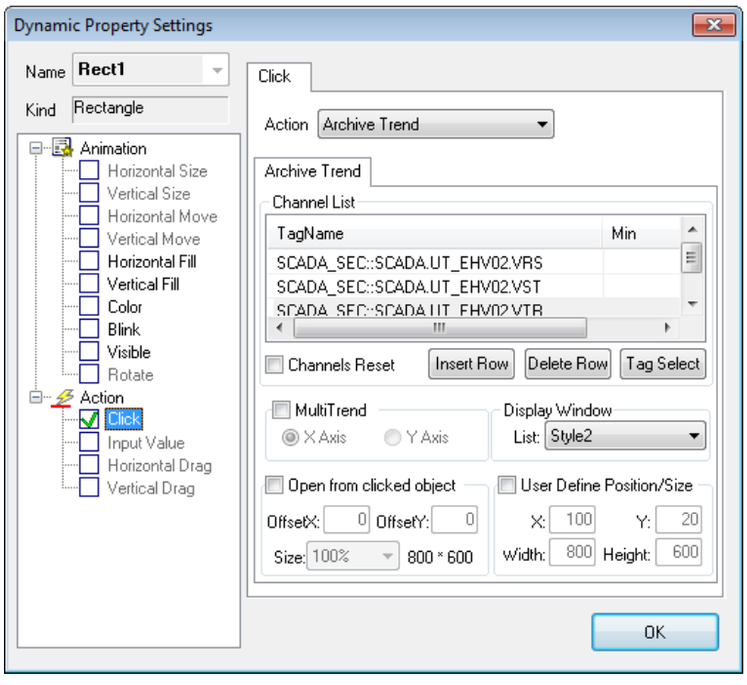
Select the page style to apply.

4) Open from clicked object

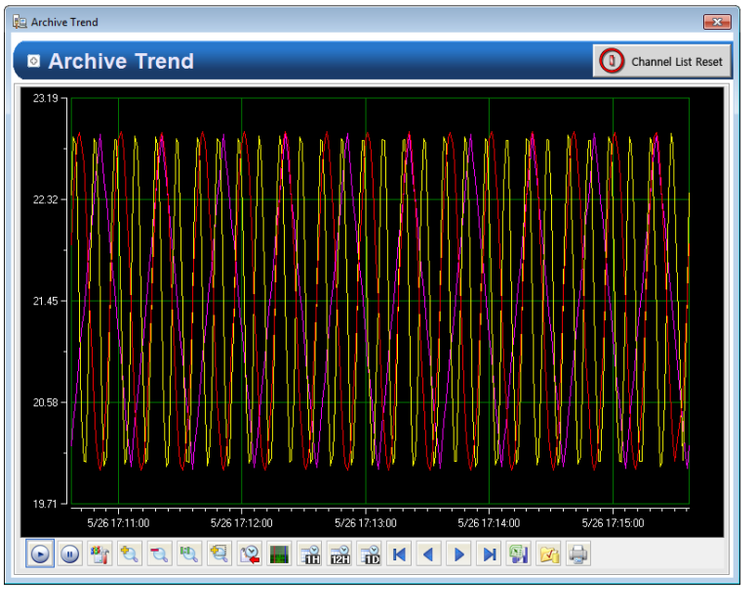
[Open from clicked object] the page will be opened in the position of the relevant object. You can specify X, Y offset and the size.

5) User Define Position/Size

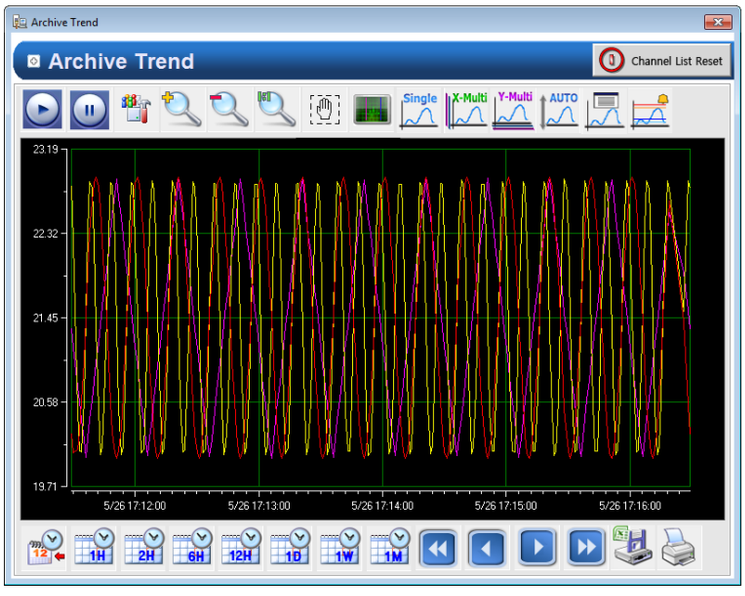
[User Define Position/Size] A user can directly specify the position and size of the page.



6) Display Window- Style1



7) Display Window- Style2



(10) Detail Realtime Trend

If you click the object, the system Detail Realtime Trend page defined in the InfoU will be open.

1) Tag Name

Directly input a digital tag name or use 'Search' (...) on the right to designate a tag name.

2) Trend axis

It can set the minimum and maximum values for y-axis. If you enter -999 for the min. and max. value, the min, and max. value of the DB tag will be automatically mapped; if you enter 0 or " ", the auto-scale will be set automatically.

3) Display Window

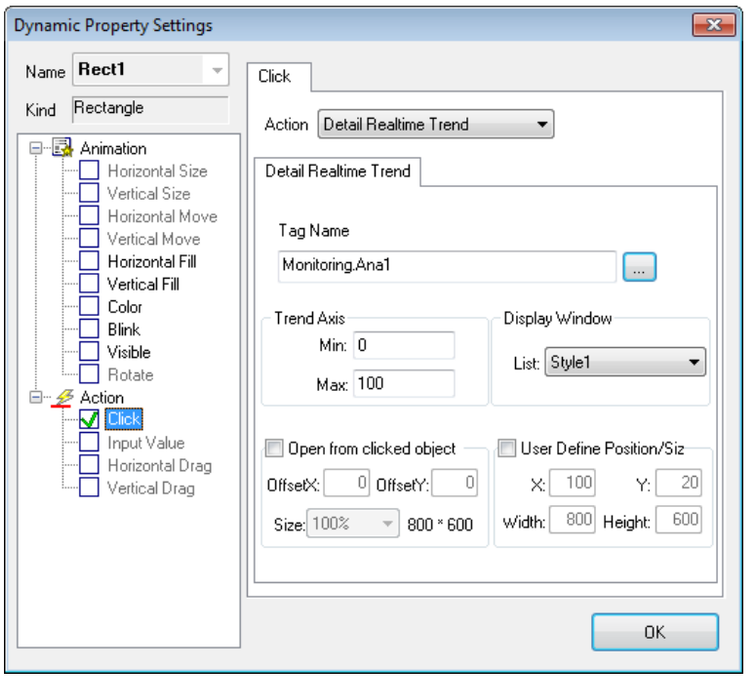
Select the page style to apply.

4) Open from clicked object

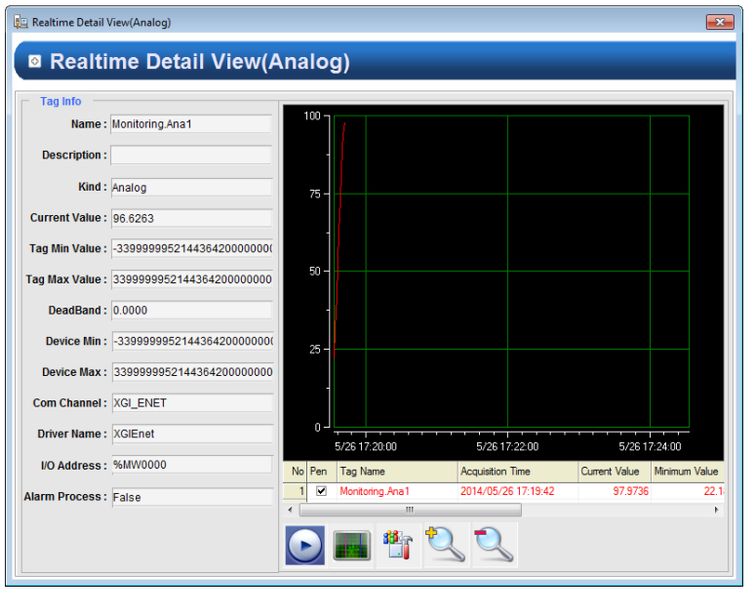
[Open from clicked object] the page will be opened in the position of the relevant object. You can specify X, Y offset and the size.

5) User Define Position/Size

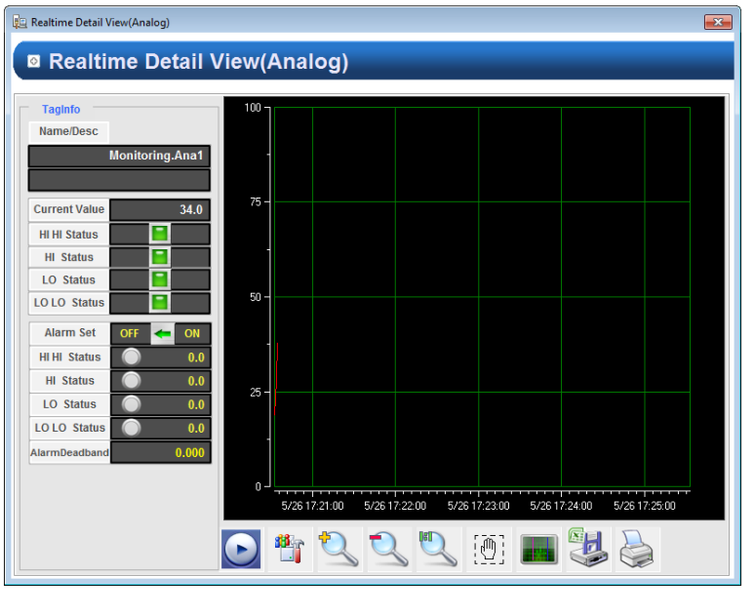
[User Define Position/Size] A user can directly specify the position and size of the page.



6) Display Window- Style1



7) Display Window- Style2



(11) Detail Archive Trend

If you click the object, the system Detail Archive Trend page defined in the InfoU will be open.

1) Tag Name

Directly input a digital tag name or use 'Search' (...) on the right to designate a tag name.

2) Trend axis

It can set the minimum and maximum values for y-axis. If you enter -999 for the min. and max. value, the min, and max. value of the DB tag will be automatically mapped; if you enter 0 or " ", the auto-scale will be set automatically.

3) Display Window

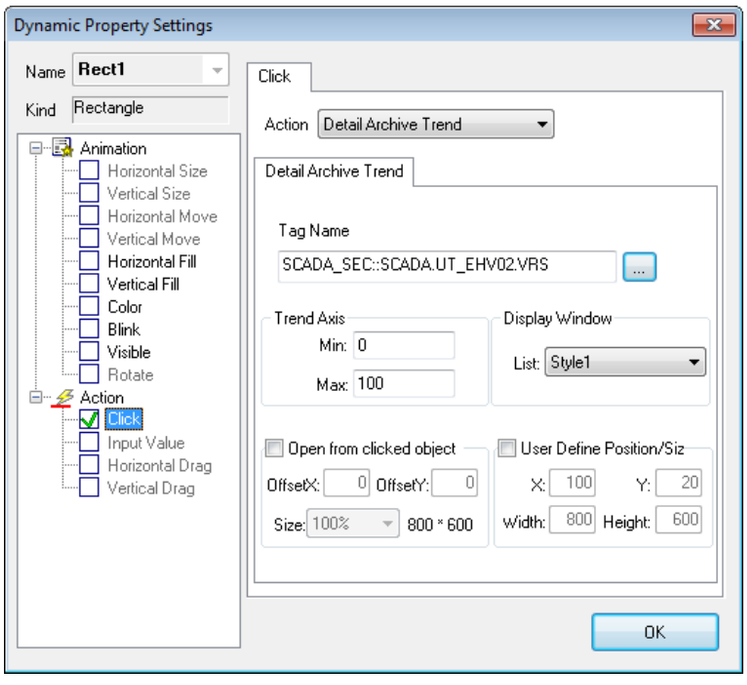
Select the page style to apply.

4) Open from clicked object

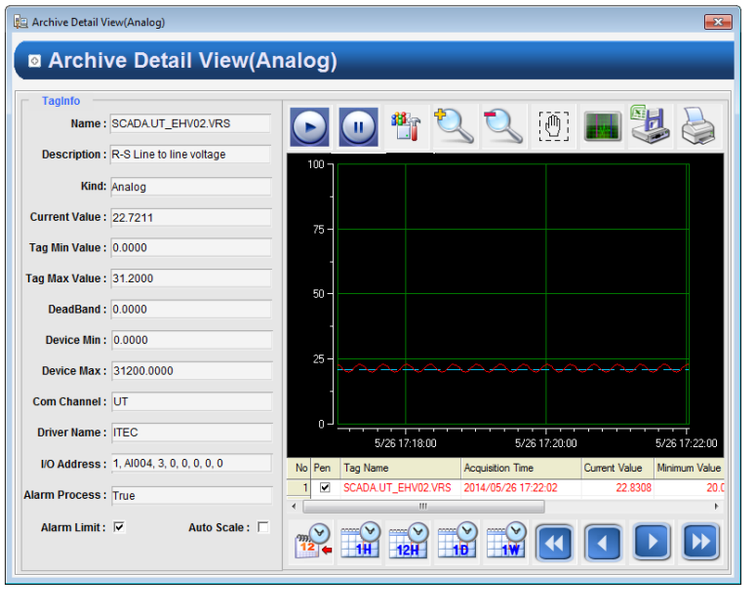
[Open from clicked object] the page will be opened in the position of the relevant object. You can specify X, Y offset and the size.

5) User Define Position/Size

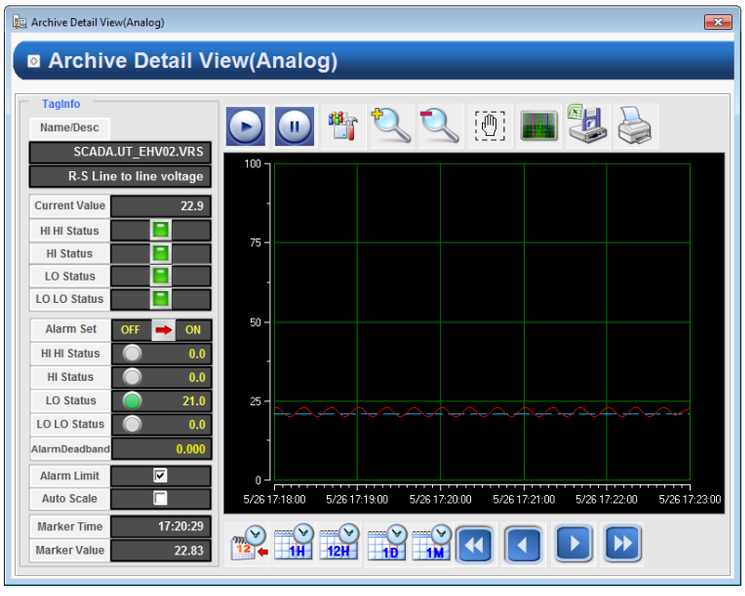
[User Define Position/Size] A user can directly specify the position and size of the page.



6) Display Window- Style1



7) Display Window- Style2



(12) Realtime List Trend

If you click the object, the system Realtime List Trend page defined in the InfoU will be open.

1) Channel List

Among the channel properties, you can input the tag names and descriptions. If you put a check [Channels Reset], the previously saved channel list is reset and the list set in the current setting window only will be applied.

2) Display Window

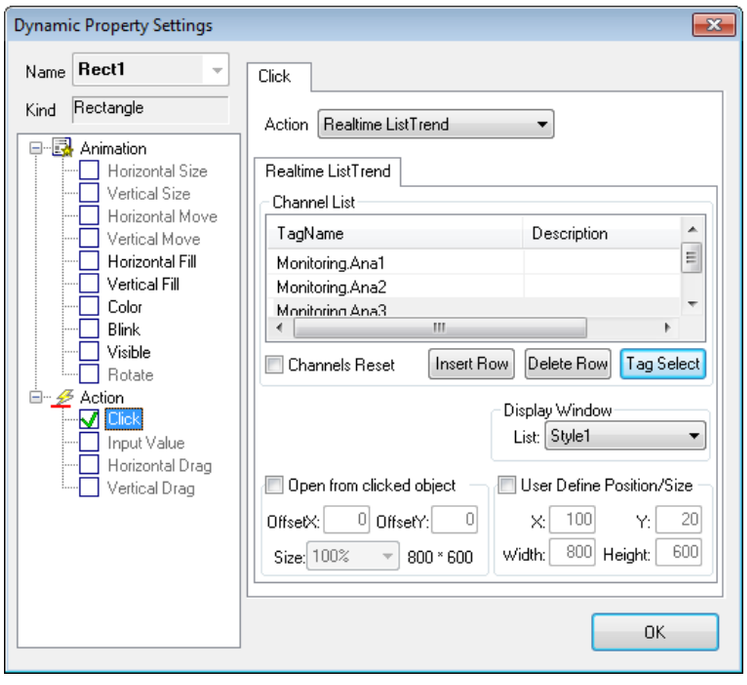
Select the page style to apply.

3) Open from clicked object

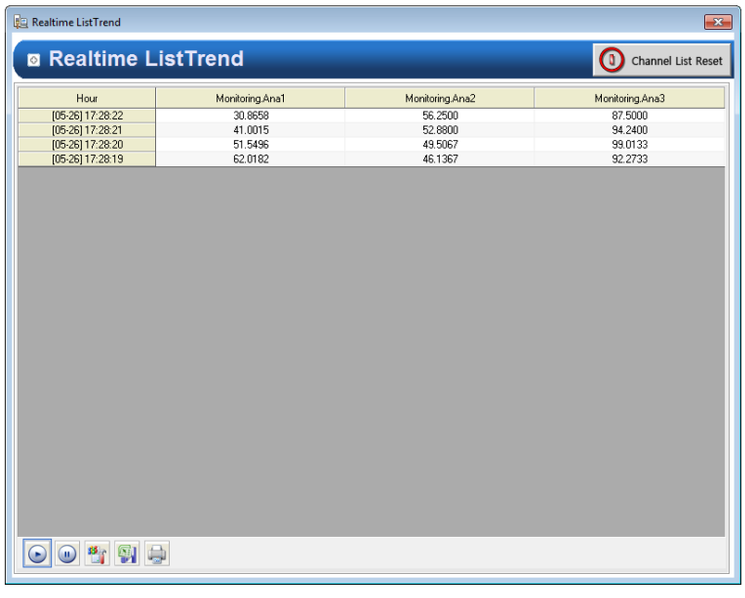
[Open from clicked object] the page will be opened in the position of the relevant object. You can specify X, Y offset and the size.

4) User Define Position/Size

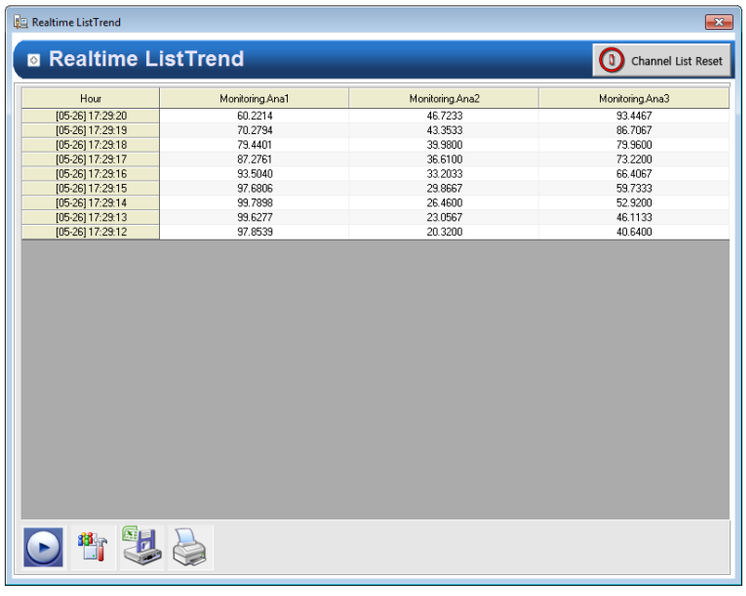
[User Define Position/Size] A user can directly specify the position and size of the page.



5) Display Window- Style1



6) Display Window- Style2



(13) Archive List Trend

If you click the object, the system Archive List Trend page defined in the InfoU will be open.

1) Channel List

Among the channel properties, you can input the tag names and descriptions. If you put a check [Channels Reset], the previously saved channel list is reset and the list set in the current setting window only will be applied.

2) Display Window

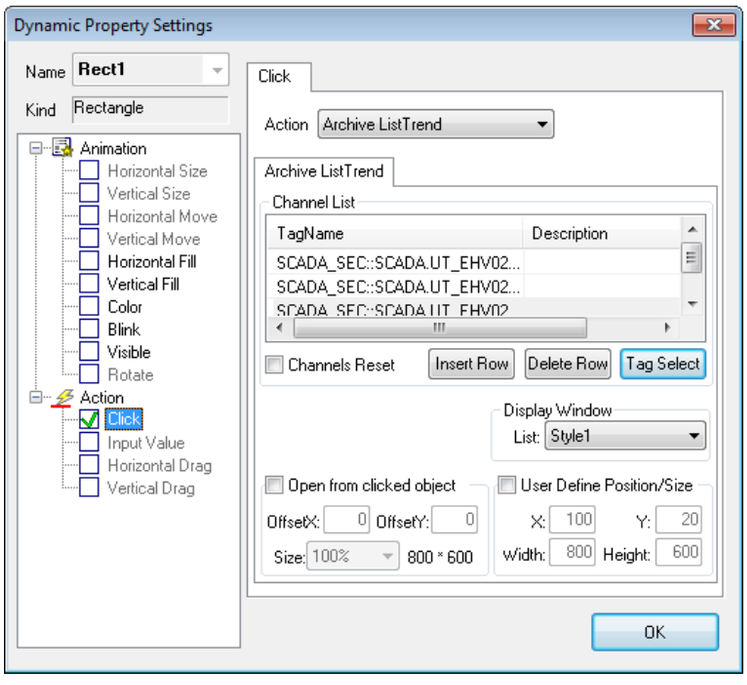
Select the page style to apply.

3) Open from clicked object

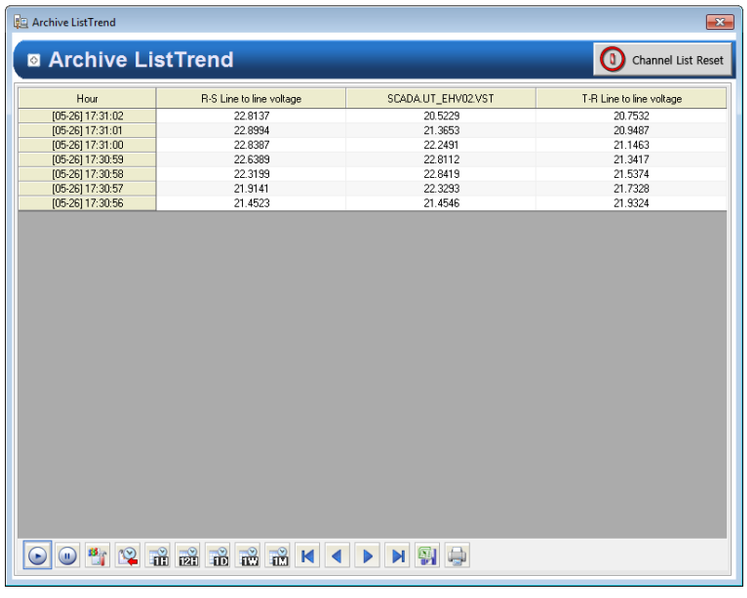
[Open from clicked object] the page will be opened in the position of the relevant object. You can specify X, Y offset and the size.

4) User Define Position/Size

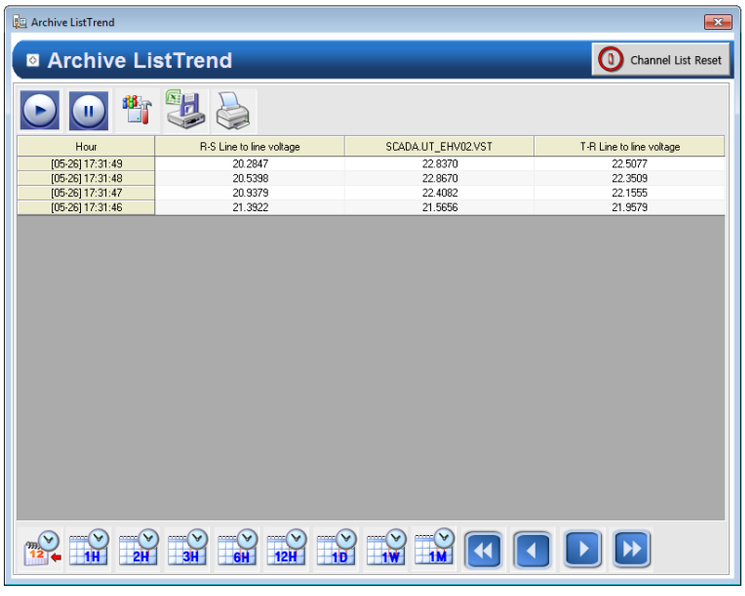
[User Define Position/Size] A user can directly specify the position and size of the page.



5) Display Window- Style1



6) Display Window- Style2



(14) Write BitFlag Value

1) Tag Name

You can directly enter the digital tag name or specify the tag name using the search button (...) on the right side.

2) Basic

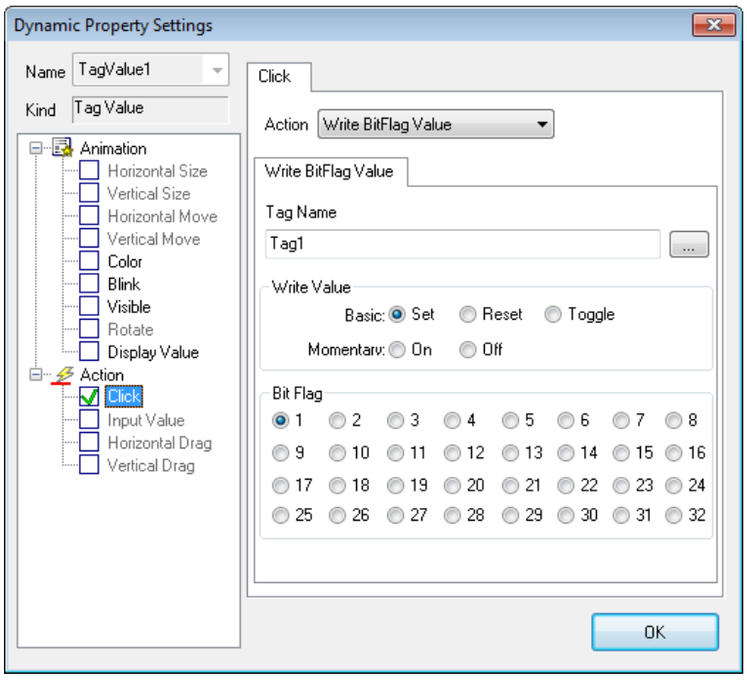
In ‘Set’ mode, specify the bit flat value of the tag as 1. In ‘Reset’ mode, specify the bit flat value of the tag as 0. In the Toggle mode, after comparing the bit flat values of the current tag, if the value is 0, specify it as 1; if the value is 1, specify it as 0.

3) Momentory

In ‘On’ mode, specify the bit flat value of the tag as 1 in Mouse Down; specify it as 0 in Mouse Up. In ‘Off’ mode, specify the bit flat value of the tag as 0 in Mouse Down; specify it as 1in Mouse Up.

4) Bit flag

Specify the bit of the tag that you want to set.



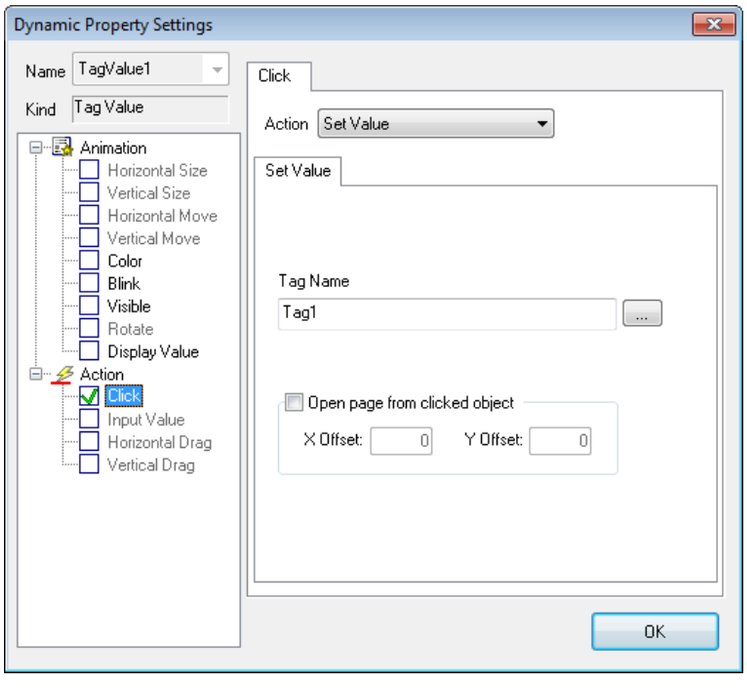
(15) Set Value

1) Tag Name

Directly input a digital tag name or use 'Search' (...) on the right to designate a tag name.

2) Open page from clicked object

[Open from clicked object] the page will be opened in the position of the relevant object. You can specify X, Y offset and the size.



9.8.18 Input Value

A function to select an object and input a value in the predefined dialog box with the mouse or Keyboard.

(1) Tag Name

Input a name of the tag to which the inputted value will apply. A tag name can be designated with 'Search' (...) on the right.

(2) Input Type

The user may choose either 'Number' or 'String' to decide how to input a value.

(3) Input Value Range

Designate the minimum and maximum value to input. The minimum value can not be bigger than the maximum one.

(4) Display Type

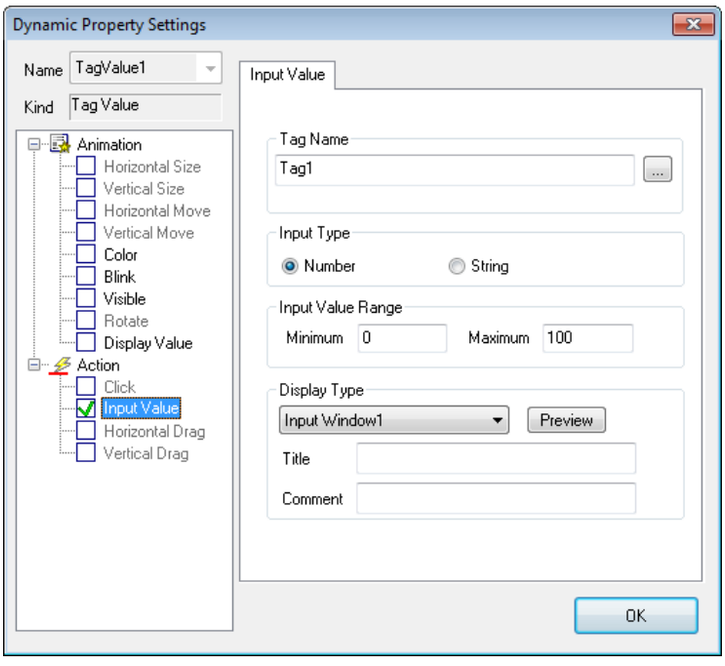
Designate a type of the window screen to be displayed in the screen.

(5) Title

Input a title of the screen..

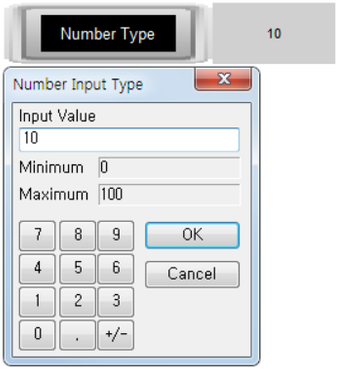
(6) Comment

Input comment on the window.



(7) Sample examples

Action screen when you input the tag value to 10.



9.8.19 Horizontal Drag

This is a function to drag an object in the horizontal direction with the mouse to convert coordinate values to tag values.

(1) Tag Name

Input a tag name to which the value of horizontal sliding will be inserted. A tag name can be designated with 'Search' (...) on the right.

(2) Direction

Set up a base location and direction of horizontal sliding.

(3) Data Variation Range

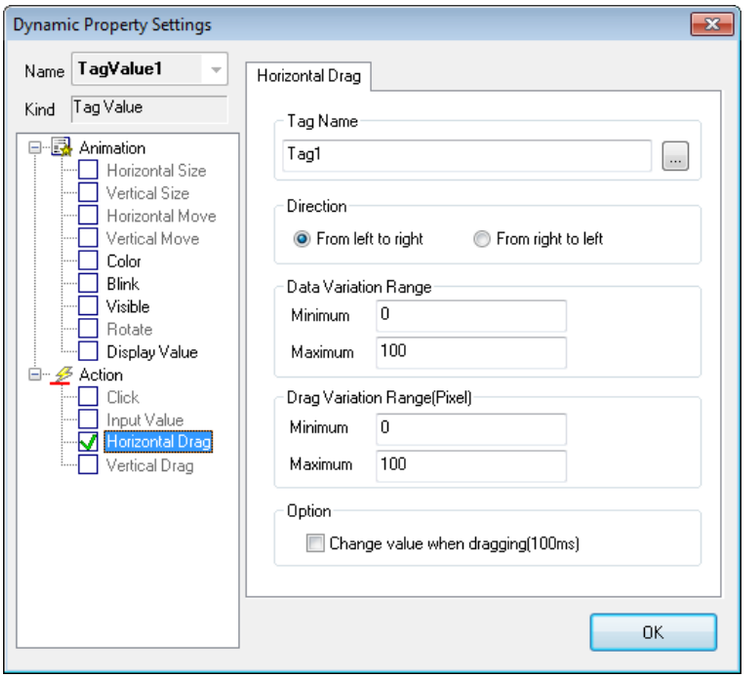
Set up a range of data to be referred during the horizontal sliding. The minimum value can not be bigger than the maximum one.

(4) Drag Variation Range (Pixel)

Set up a range of move during the horizontal sliding. The minimum move can not be bigger than the maximum one.

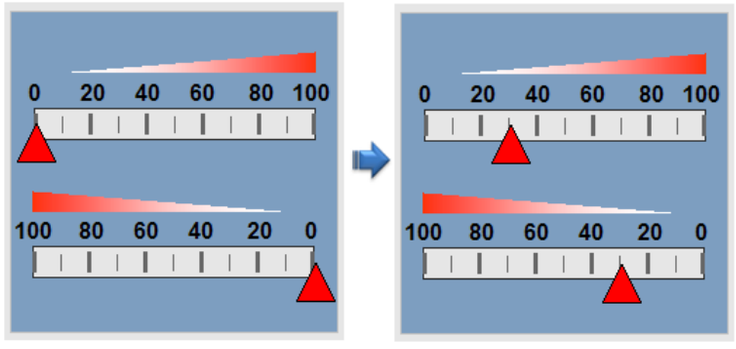
(5) Change value when dragging (100ms)

Check √ in the box if the user wants to operate the dragging value while the mouse is moving.



(6) Sample examples

Action screen when moving the object: 0→30 pixel



9.8.20 Vertical Drag

A function to slide an object in the vertical direction with the mouse to convert coordinate values to tag values.

(1) Tag Name

Input a tag name to which the value of vertical sliding will be inserted. A tag name can be designated with 'Search' (...) on the right.

(2) Direction

Set up a base location and direction of vertical sliding.

(3) Data Variation Range

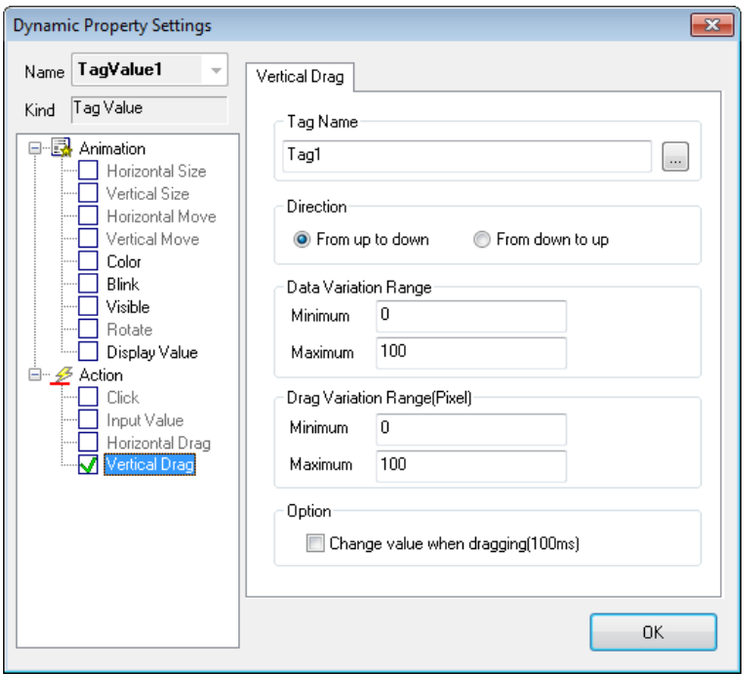
Set up a range of data to be referred during the vertical sliding. The minimum value can not be bigger than the maximum one.

(4) Drag Variation Range (Pixel)

Set up a range of move during the vertical sliding. The minimum move can not be bigger than the maximum one.

(5) Change value when dragging (100ms)

Check √ in the box if the user wants to operate the dragging value while the mouse is moving.

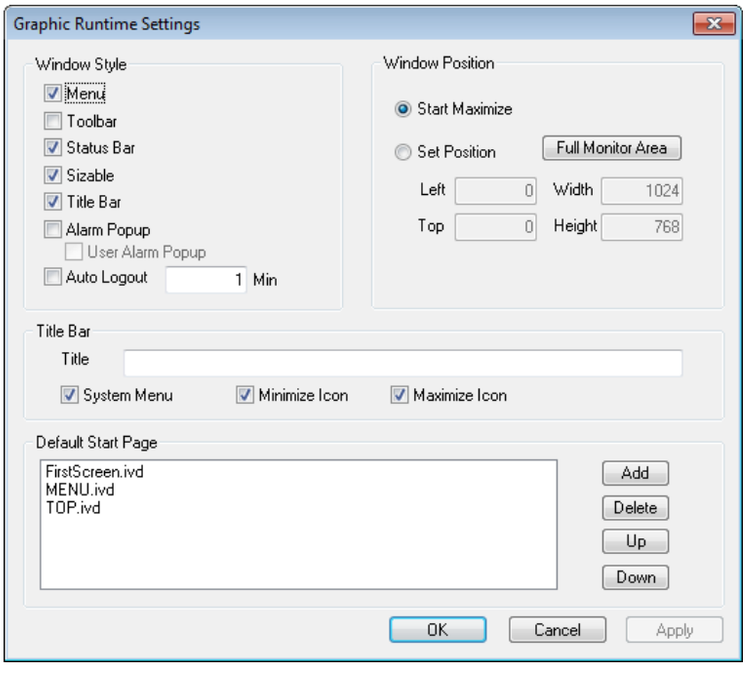


9.9 Runtime Settings

9.9.1 Graphic Runtime Settings

'Runtime settings' is to set up the environment for the graphic runtime (InfoUX.exe).

To set up the environment, the user can set up menu, view/hide toolbar, initial execution position, name to be displayed in the title bar, view/hide window control buttons and other options and also set up a default start page to open simultaneously along with execution.



(1) Window Style

1) Menu

Select whether to view the basic menu.

2) Toolbar

Select whether to view the toolbar.

3) Status Bar

Select whether to view the status bar.

4) Sizable

Select whether to allow the operator to change the screen size.

5) Title Bar

Select whether to display the title window located on the top of the window.

6) Alarm Popup

Select whether executing the alarm popup window when an alarm occurs.

7) Auto Logout

Select whether executing auto-logout when a mouse or a keyboard does not work within the fixed time.

(2) Window Position

1) Start Maximize

Select whether to execute RunTime in the maximum screen.

2) Set Position

The user can designate the location and size of the screen during Run Time.

Click 'Full Monitor Area' button, the area coordinates for the multiple monitors are displayed.

The coordinates reference (0, 0) is for the main monitor.

(3) Title Bar

1) Title

Input the title of the screen.

2) System Menu

Select whether to display system menus in icon types on the left top of the screen.

3) Minimize Icon

Select whether to display minimized buttons on the right top of the screen.

4) Maximize Icon

Select whether to display maximized buttons on the right top of the screen.

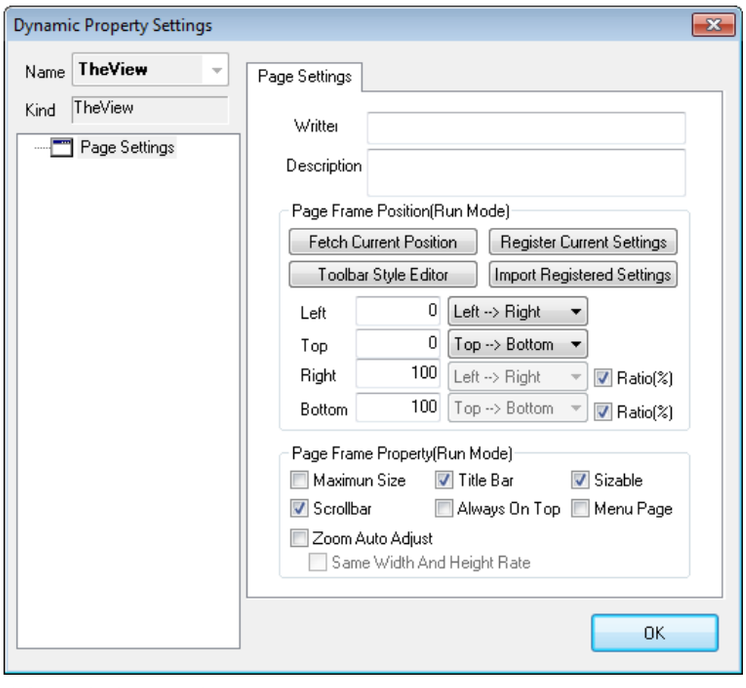
(4) Default Start Page

This is for selecting screens to be displayed at first when executing Run Time. If 'Add' button is pressed, all of the written pages for the current page are listed up and the user may select one to be used as a start page.

9.9.2 Page Settings

The page settings are used to designate a view (size and position, etc) when each graphic screen is displayed. In order to set up the page environment, click [View] ☞ [Object property] of the menu in the screen to display the 'page settings' screen.

The screen to set up the page environment is as follows.



(1) Window Style

1) Writer and Description

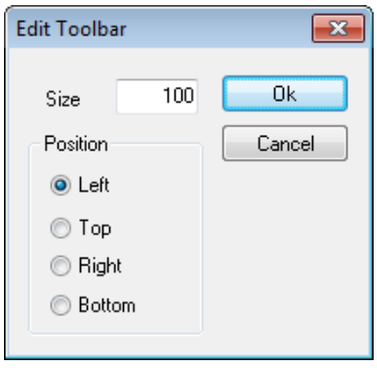
Input the writer of the relevant graphic file and describe on the file. (Optional and informative)

2) Fetch Current Position

Calculate the location and size during the editing in the graphic editor and display them on the location/size input box.

3) Toolbar Style Editor

It provides the editing window to set up the page in a simple toolbar style.



4) Register Current Settings

Register and save the current setting value in the page setting window.

5) Import Registered Settings

Import the registered and saved page setting value to the page setting window.

6) Left (Start X Position)

Show coordinates X in which the relevant page is located when executed. For the base of coordinates X, designate the number of pixels starting from the area displayed in the screen (except the left frame area in the window) out of the main window of the graphic runtime program.

7) Top (Start Y Position)

Show coordinates Y in which the page is located when executing. For the base of coordinates Y, designate the number of pixels starting from the area displayed in the screen (any location other than caption bar, menu bar and tool bar) out of the main window of the graphic runtime program.

8) Right

Designate pixel/ratio and width/right of the page when executing.

9) Bottom

Designate pixel/ratio and height/bottom of the page when executing.

10) Title Bar

Set up whether to show/hide the caption bar (title bar) of the page when executing.

11) Maximum size

Show the page in a full screen of the runtime frame when executing.

12) Sizable

Make the size of the page adjustable. When it is adjustable, the size of the window can be changed by clicking and dragging the edge of the window, and a thick window frame is used. If the size adjustment is not available, the size of the page cannot be changed, and a thin window frame is shown.

13) Scroll bar

Set up whether to show/hide the scroll bar of the page when executing.

14) Always On Top

If 'Always on Top' option is selected, the window is always located on top among various graphic pages and it exists above other windows (among graphic monitoring pages).

15) Menu page

Designate the relevant page as a menu page when executing. The file designated as menu page is not closed and continuously maintained even close (menu, toolbar) command is executed.

(If the user wants to close the page forcibly, he/she may execute 'closing' while pressing shift key.)

16) Zoom Auto Adjust

The zoom ratio (zoom in/out) is automatically adjusted (fit to the screen) by width and length ratio of the screen when opening the page.

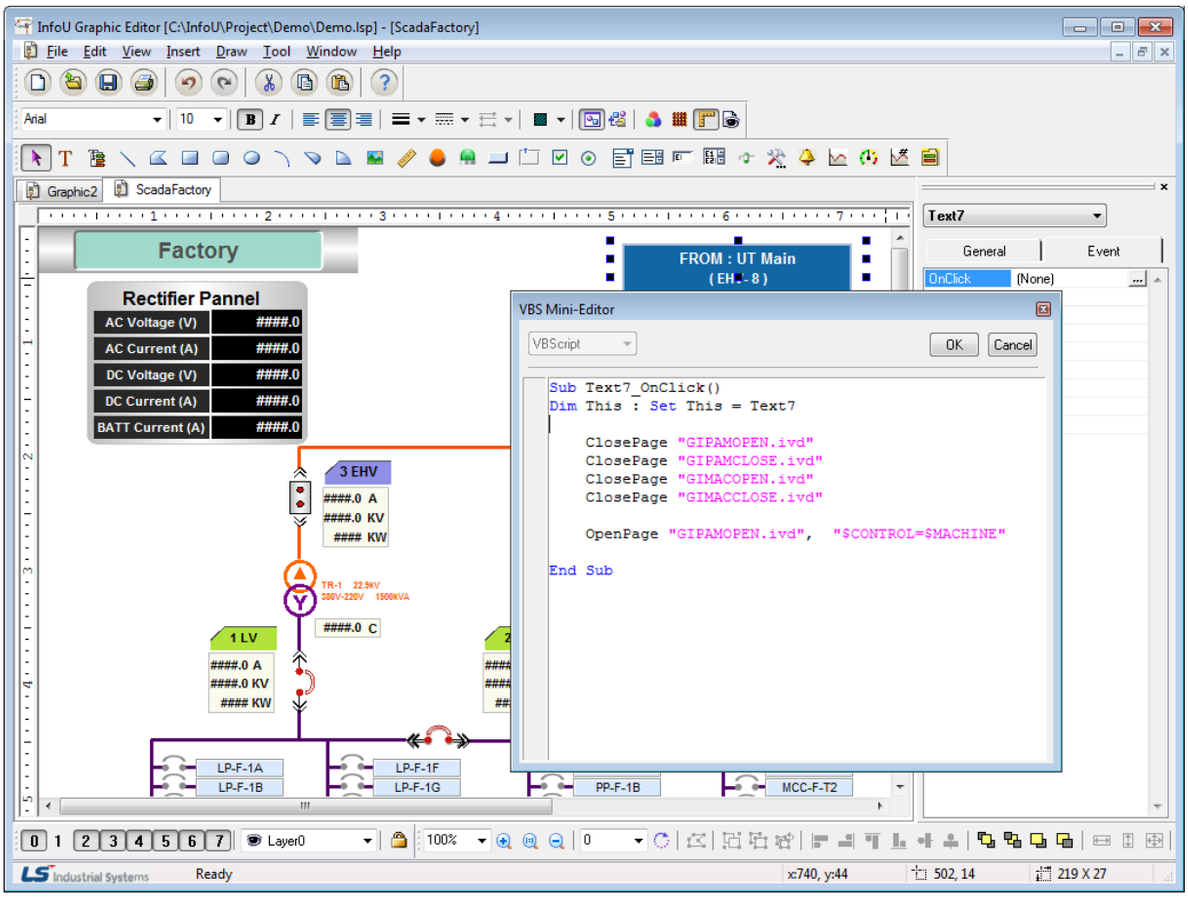
17) Same Width and Height Rate

When opening the page, the zoom ratio (zoom in/out) is automatically adjusted (fit to the screen) by maintaining the ratio of width and height.

9.10 Graphic Script

This Graphic Script is a function to define the activities for the object event (user's mouse action) or periodic execution (OnUpdate Function: Executed at every set-up time in the screen UpdatePeriod) as VB Script.

The available commands at this time are the ones provided by the standard Microsoft VB Script Engine and the graphic functions exclusively available in the InfoU system.



9.10.1 InfoU Graphic Function List

The functions available at the script of this graphic editor are composed of the standard VB script functions and the InfoU graphic functions. The InfoU graphic functions are as seen in the following table.

Function Type	Function Declarations
AckAllAlarm	void AckAllAlarm()
ChangePage	long ChangePage(BSTR OldPageName, BSTR NewPageName)
ChangePageEx	long ChangePageEx(BSTR OldPageName, BSTR NesPageName, BSTR Tag Param)
ClosePage	long ClosePage(BSTR PageName)
ExecuteCommand	void ExecuteCommand(BSTR PathName)
GetAllAlarmCount	long GetAllAlarmCount(void)
GetGraLSPath	BSTR GetGraLSPath(void)

GetHistDataCount	long GetHistDataCount(LPCTSTR ModelName, LPCTSTR TagName, LPCTSTR DataKind, double ReferenceTime, long DataCount, long Direction, VARIANT FAR* ResultCount, VARIANT FAR* ArrTimeStamp, VARIANT FAR* ArrDblTagVal, VARIANT FAR* ArrLongTagStatus, VARIANT FAR* ArrLongRets)
GetHistDataCountStr	long GetHistDataCountStr(LPCTSTR ModelName, LPCTSTR TagName, LPCTSTR DataKind, double ReferenceTime, long DataCount, long Direction, VARIANT FAR* ResultCount, VARIANT FAR* ArrTimeStamp, VARIANT FAR* ArrStrTagVal, VARIANT FAR* ArrLongTagStatus, VARIANT FAR* ArrLongRets)
GetHistDataPeriod	long GetHistDataPeriod(LPCTSTR ModelName, LPCTSTR TagName, LPCTSTR DataKind, double FromTime, double ToTime, VARIANT FAR* ResultCount, VARIANT FAR* ArrTimeStamp, VARIANT FAR* ArrDblTagVal, VARIANT FAR* ArrLongTagStatus, VARIANT FAR* ArrLongRets)
GetHistDataPeriodStr	long GetHistDataPeriodStr(LPCTSTR ModelName, LPCTSTR TagName, LPCTSTR DataKind, double FromTime, double ToTime, VARIANT FAR* ResultCount, VARIANT FAR* ArrTimeStamp, VARIANT FAR* ArrStrTagVal, VARIANT FAR* ArrLongTagStatus, VARIANT FAR* ArrLongRets)
GetLoggingModelList	long GetLoggingModelList(long lOutCount,VARIANT arrModelName, VARIANT arrModelDesc, VARIANT arrModelType, VARIANT arrArchivingCycle, VARIANT arrSegmentPeriod)
GetLoggingTag	long GetLoggingTag(LPCTSTR ModelName,long lOutCount, VARIANT arrTagName, VARIANT arrTagType)
GetNoActionsTime	long GetNoActionsTime(void)
GetProjectPath	BSTR GetProjectPath(void)
GetRecentAlarmTagName	BSTR GetRecentAlarmTagName(void)
GetTagOFFSTR	BSTR GetTagOFFSTR(BSTR TagName)
GetTagONSTR	BSTR GetTagONSTR(BSTR TagName)
GetTagVal	VARIANT GetTagVal(BSTR TagVarName)
GetTagValMAX	VARIANT GetTagValMAX(BSTR TagName)
GetTagValMIN	VARIANT GetTagValMIN(BSTR TagName)
GetUnAckAlarmCount	long GetUnAckAlarmCount (void)
GetUnRecoverAlarmCount	long GetUnRecoverAlarmCount (void)
GetUserID	BSTR GetUserID()
GetUserInfoList	long GetUserInfoList(long lOutCount, VARIANT rrUserID, VARIANT arrUserName, VARIANT arrDesc, VARIANT arrPassword,VARIANT arrUserLevel,

	VARIANT arrHPhoneNo, VARIANT arrPhoneNo, VARIANT FAR* arrMailAdd)
GetUserLevel	long GetUserLevel()
HardCopy	long HardCopy()
HolidaySet	long HolidaySet()
FindPage	VARIANT FindPage(LPCTSTR PageName)
IsPageOpen	BOOL IsPageOpen(LPCTSTR PageName)
MsgTrace	void MsgTrace(VARIANT MessageValue)
OpenPage	long OpenPage(BSTR PageName, BSTR TagParam)
OpenPageEx	long OpenPageEx(VARIANT X,VARIANT Y, VARIANT cX, VARIANT cY, BSTR PageName, BSTR TagParam)
OpenPageObject	long OpenPageObject(BSTR PageName, BSTR TagParam, BSTR ObjectName)
PlaySound	void PlaySound(BSTR WaveFile)
PulseOff	void PulseOn(BSTR TagName, long MilliSec)
PulseOn	void PulseOff(BSTR TagName, long MilliSec)
Quit	long Quit()
QuitProject	long QuitProject()
Reload	long ReLoad(BSTR PageName, BSTR TagParam)
ReportDatabase	long ReportDatabase(BSTR ReportName)
ReportDatabaseEx	long ReportDatabaseEx(BSTR ReportName, BSTR RefTime)
ReportDialog	long ReportDialog()
ReportPrint	long ReportPrint(BSTR ReportName)
ReportPrintEx	long ReportPrintEx(BSTR ReportName, BSTR RefTime)
ResetStatistic	long ResetStatistic(BSTR TagName)
RunScript	void RunScript(BSTR ScriptName)
SaveDeskTopToBmp	void SavePageToBmp(LPCTSTR FilePathName)
SavePageToBmp	void SavePageToBmp(LPCTSTR FilePathName)
SendData	BOOL SendData(LPCTSTR PageName, VARIANT FAR& VariantData)
SetLogging	void SetLogging(BOOL TrueFalse)
SetTagVal	long SetTagVal(BSTR TagVarName, VARIANT Value,VARIANT_BOOL bRet)
SetCopyFile	void SetCopyFile(LPCTSTR SourceFile, LPCTSTR NewFile)
ShowAlarmPage	long ShowAlarmPage(BSTR AlarmTag)
ShowCrossReference	void ShowCrossReference (BSTR TagName)
ShowFileDialog	BSTR ShowFileDialog(VARIANT_BOOL bOpen)
ShowHelp	long ShowHelp(BSTR HelpFile, long idx)
ShowLayer	void ShowLayer(LONG Index, VARIANT_BOOL bVisible)
ShowNetworkStatus	long ShowNetworkStatus()
ShowRawDataInfo	long ShowRawDataInfo()

ShowTagView	void ShowTagView(BSTR TagName)
ShowVirtualKeyboard	void ShowVirtualKeyboard(void)
Sleep	void Sleep(long Milliseconds)
StartAlarmSound	long StartAlarmSound(void)
StopAlarmSound	long StopAlarmSound()
TaskBarShow	void TaskBarShow(VARIANT_BOOL bShow)
TaskBarAutoHide	void TaskBarAutoHide(VARIANT_BOOL bAutoHide)
TimeOutMsgBox	void TimeOutMsgBox(BSTR Msg, BSTR Title, long nSec)
UserAdd	void UserAdd(LPCTSTR UserID, LPCTSTR UserPassword, LPCTSTR UserLevel, LPCTSTR UserName, LPCTSTR UserDesc, LPCTSTR UserCompany, LPCTSTR UserPhoneNo, LPCTSTR UserHPhoneNo, LPCTSTR UserMailAdd, VARIANT_BOOL UserSMSCheck, VARIANT_BOOL UserMailCheck)
UserDel	void UserDel(LPCTSTR UserID)
UserCommand1	long UserCommand1(long nCmd, BSTR Str1, BSTR Str2, BSTR Str3, BSTR Str4)
UserLogOff	void UserLogOff(void)
UserLogOn	BOOL UserLogOn(BSTR UserID, BSTR PassWord)
UserPassChange	void UserPassChange(void)

9.10.2 Explanation of InfoU Graphic Function

AckAllAlarm Function	
Description	ACK all alarms.
Syntax	AckAllAlarm()

ChangePage Function	
Description	Change the graphic page.
Syntax	ChangePage(BSTR OldPageName, BSTR NewPageName) OldPageName: Save the previous page. The previous page can be designated "abc.ivd" or "*." The page designated as the previous page will be closed, and the page designated in NewPageName will be opened. NewPageName: Designate a name of the page that will be newly opened. If the user does not designate this dPageName (empty string), the current page will be closed.
Reference	If the user designates "*" to this dPageName, all pages will be closed. However, if the page designated to this dPageName is a menu page, it will not be closed. The menu page can be designated on the 'Page settings' screen in this Graphic Editor.

ChangePageEx Function	
Description	Change the graphic page
Syntax	ChangePage(BSTR OldPageName, BSTR NesPageName, BSTR TagParam) OldPageName: It specifies the previous page. You can set the previous page as “abc.ivd” or “*”. Then, the page that was set as the previous one is closed and the page specified in ‘NewPageName’ will be open. NewPageName: You can specify the file name of the page to open newly. OldPageName: In case you do not specify the page(empty string), the current page will be closed. TagParam: In the template screen, you can specify the name of the tag to replace. If you do not specify the tag to replace, just specify the empty string("").
Reference	If you enter “*” into the ‘OldPageName’, all pages will be closed. However, the page is set as the menu page in the ‘OldPageName’, it will not be closed. You can specify the menu page in ‘Page Configuration’ of the ‘Dynamic Property Settings’ screen of the graphic editor. OldPageName: It specifies the old page. You can set the old page as “abc.ivd” or “*”. Then, the page that was set as the old one is closed and the page specified in ‘NewPageName’ will be open.

ClosePage Function	
Description	Close the graphic page.
Syntax	ClosePage (BSTR PageName) PageName: Designate the page to be closed. It is available to designate with either“abc.ivd” or “*”.
Reference	If the user designates with "*" in the PageName, all pages will be closed. However, if the page designated to this PageName is a menu page, it will not be closed. The menu page can be designated on 'Page settings' screen in this Graphic Editor

ExecuteCommand Function	
Description	Executes an external program.
Syntax	ExecuteCommand(BSTR PathName)
Reference	PathName: Designate the program name to be exeuted.

FindPage Function	
Description	Find the designated page name from the opened pages.
Syntax	FindPage(LPCTSTR PageName)
Reference	The page pointer value is returned in the form of variant type.

GetAllAlarmCount Function	
Description	It notifies you of the number of all occurring alarms.
Syntax	GetAllAlarmCount()

GetGraLSPath Function	
Description	It notifies you the paths where the screen pages are located.
Syntax	BSTR GetGraLSPath(void)

GetHistDataCount Function	
Description	For the specified history model and tag, it informs you of the history data (time stamp, tag value, status value, result value) as much as the number of history data that occurred in the request time.
Syntax	long GetHistDataCount(LPCTSTR ModelName, LPCTSTR TagName, LPCTSTR DataKind, double ReferenceTime, long DataCount, long Direction, VARIANT FAR* ResultCount, VARIANT FAR* ArrTimeStamp, VARIANT FAR* ArrDblTagVal, VARIANT FAR* ArrLongTagStatus, VARIANT FAR* ArrLongRets) ModelName: History of model name TagName: Name of tag DataKind: Data type (default: current value) ReferenceTime: Request time DataCount: Number of history data Direction: Direction or time sequence ResultCount: Result of number of history data ArrTimeStamp: Time stamp array ArrDblTagVal: Tag value array ArrLongTagStatus: Tag status array ArrLongRets: Result of history data
Reference	Dim ModelName Dim TagName Dim DataKind

	Dim RefTime Dim lCount Dim lDir Dim ResultCount Dim TimeStamp Dim NumValue Dim Status Dim Result ModelName = "INV1_SEC" TagName = "INV_1.1_AI_CT" DataKind = "Current value" RefTime = Now() lCount = 5 lDir = 1 GetHistDataCount(ModelName, TagName, DataKind, RefTime, lCount, lDir, ResultCount,TimeStamp, NumValue, Status, Result) For count = 0 To ResultCount -1 MsgBox("NumValue = " + CStr(NumValue(count)) + " Status = " + CStr(Status (count))) Next

GetHistDataCountStr Function	
Description	For the specified string history model and tag, it informs you of the history data (time stamp, tag value, status value, result value) as much as the number of history data that occurred in the request time.
Syntax	long GetHistDataCount(LPCTSTR ModelName, LPCTSTR TagName, LPCTSTR DataKind, double ReferenceTime, long DataCount, long Direction, VARIANT FAR* ResultCount, VARIANT FAR* ArrTimeStamp, VARIANT FAR* ArrDblTagVal, VARIANT FAR* ArrLongTagStatus, VARIANT FAR* ArrLongRets) ModelName: History of model name TagName: Name of tag DataKind: Data type (default: current value) ReferenceTime: Request time DataCount : Number of history data

	Direction: Direction or time sequence ResultCount: Result of number of history data ArrTimeStamp: Time stamp array ArrDblTagVal: Tag value array ArrLongTagStatus: Tag status array ArrLongRets: Result of history data
Reference	Dim ModelName Dim TagName Dim DataKind Dim RefTime Dim lCount Dim lDir Dim ResultCount Dim TimeStamp Dim NumValue Dim Status Dim Result ModelName = "INV1_SEC" TagName = "INV_1.1_AI_CT" DataKind = " Current value " RefTime = Now() lCount = 5 lDir = 1 GetHistDataCount(ModelName, TagName, DataKind, RefTime, lCount, lDir, ResultCount,TimeStamp, NumValue, Status, Result) For count = 0 To ResultCount -1 MsgBox("NumValue = " + CStr(NumValue(count)) + " Status = " + CStr(Status (count))) Next

GetHistDataPeriod Function	
Description	For the specified history model and tag, it informs you of the history data (time stamp, tag value, status value, result value) that occurred within a given period.
Syntax	long GetHistDataPeriod(LPCTSTR ModelName, LPCTSTR TagName, LPCTSTR DataKind, double FromTime, double ToTime, VARIANT FAR* ResultCount, VARIANT FAR* ArrTimeStamp, VARIANT FAR* ArrDblTagVal, VARIANT FAR* ArrLongTagStatus,

	VARIANT FAR* ArrLongRets) ModelName: History of model name TagName: Name of tag DataKind: Data type (default: current value) FromTime: Starting time ToTime: Ending time ResultCount: Result of number of history data ArrTimeStamp: Time stamp array ArrDblTagVal: Tag value array ArrLongTagStatus: Tag status array ArrLongRets: Result of history data
Reference	Dim ModelName Dim TagName Dim DataKind Dim dFrom Dim dTo Dim ResultCount Dim TimeStamp Dim NumValue Dim Status Dim Result ModelName = "INV1_SEC" TagName = "INV_1.1_AI_CT" DataKind = "Current value" dFrom = Now() -1 dTo = Now() GetHistDataPeriod(ModelName, TagName, DataKind, dFrom, dTo, ResultCount, TimeStamp, NumValue, Status, Result) For count = 0 To ResultCount -1 MsgBox("NumValue = " + CStr(NumValue(count)) + " Status = " + CStr(Status (count))) Next

GetHistDataPeriodStr Function	
Description	For the specified string history model and tag, it informs you of the history data (time stamp, tag value, status value, result value) that occurred within a given period.
Syntax	long GetHistDataPeriodStr(LPCTSTR ModelName, LPCTSTR TagName,

	LPCTSTR DataKind, double FromTime, double ToTime, VARIANT FAR* ResultCount, VARIANT FAR* ArrTimeStamp, VARIANT FAR* ArrStrTagVal, VARIANT FAR* ArrLongTagStatus, VARIANT FAR* ArrLongRets) ModelName: History of model name TagName: Name of tag DataKind: Data type (default: current value) FromTime: Starting time ToTime: Ending time ResultCount: Result of number of history data ArrTimeStamp: Time stamp array ArrDblTagVal: Tag value array ArrLongTagStatus: Tag status array ArrLongRets: Result of history data
Reference	Dim ModelName Dim TagName Dim DataKind Dim dFrom Dim dTo Dim ResultCount Dim TimeStamp Dim NumValue Dim Status Dim Result ModelName = "INV1_SEC" TagName = "INV_1.1_AI_CT" DataKind = "Current value" dFrom = Now() -1 dTo = Now() GetHistDataPeriod(ModelName, TagName, DataKind, dFrom, dTo, ResultCount, TimeStamp, NumValue, Status, Result) For count = 0 To ResultCount -1 MsgBox("NumValue = " + CStr(NumValue(count)) + " Status = " + CStr(Status (count))) Next

GetLoggingModelList Function	
Description	It notifies the user on the list of models registered in the history.
Syntax	long GetLoggingModelList(long lOutCount,VARIANT arrModelName, VARIANT arrModelDesc, VARIANT arrModelType, VARIANT arrArchivingCycle, VARIANT arrSegmentPeriod)
Reference	Sub Text2_OnClick() Dim This : Set This = Text2 'Declare variable Dim nCount'Number of registered models Dim ModelName() 'Model Name Dim ModelDesc()'Model Description Dim ModelType()'Model Type Dim ArchivCycle()'Collection cycle (return string: 20 seconds = 20ss, 30 minutes=30mm, 5 hours=05hh, 36 months = 36MM) Dim SavePeriod()'Save period Dim ret Call 'Graphic Script function (ret value is undefined) ret = GetLoggingModelList (nCount, ModelName, ModelDesc, ModelType, ArchivCycle, SavePeriod) Dim strMessage ‘The call result into string … strMessage = "Model Name/Model Description/Model Type/collection cycle/save period" strMessage = strMessage + vbCrlf For idx = 0 To nCount -1 strMessage = strMessage+ ModelName(idx) strMessage = strMessage+ "/" EditBox1.Text = ModelName(idx) ' Use it to search tag by model name. strMessage = strMessage+ ModelDesc(idx) strMessage = strMessage+ "/"

	strMessage = strMessage+ ModelType(idx) strMessage = strMessage+ "/" strMessage = strMessage+ ArchivCycle(idx) strMessage = strMessage+ "/" strMessage = strMessage+ SavePeriod(idx) strMessage = strMessage + vbCrlf Next 'Check the result MsgBox strMessage End Sub

GetLoggingTag Function	
Description	It notifies the user on the list of tags registered in the given history model.
Syntax	long GetLoggingTag(LPCTSTR ModelName, long lOutCount, VARIANT arrTagName, VARIANT arrTagType)
Reference	Sub Text3_OnClick() Dim This : Set This = Text3 Dim nCount'Number of registered models Dim TagName()'Tag name array Dim TagType()'Tag type array Dim ret Dim modelName modelName = EditBox1.Text Call 'Graphic Script function (ret value is undefined) ret = GetLoggingTag (modelName, nCount, TagName, TagType) Dim strMessage

	'The call result into string … strMessage = "Tag name/Tag type" strMessage = strMessage + vbCrlf For idx = 0 To nCount -1 strMessage = strMessage+ TagName(idx) strMessage = strMessage+ "/" strMessage = strMessage+ TagType(idx) strMessage = strMessage + vbCrlf Next 'Check the result MsgBox strMessage End Sub

GetNoActionsTime Function	
Description	After setting auto-logout and then, it gets the time value (ms) that elapsed after Mouse Down in the graphic runtime.
Syntax	LONG GetNoActionsTime(void)

GetProjectPath Function	
Description	It gets the path of the currently running project.
Syntax	BSTR GetProjectPath(void)

Reference	Dim PrjName As String
	
	PrjPath = GetProjectPath()
	If lRet <> 0 Then
	MsgBox "GetProjectPath() False"
	Else
	MsgBox "GetProjectPath() OK " + PrjPath
	End If

GetRecentAlarmTagName Function	
Description	It gets the tag name of the alarm that has recently occurred.
Syntax	BSTR GetRecentAlarmTagName(void)

GetUserInfoList Function	
Description	Get the list of all users and individual information on each user. The user information will be used for SMS call or mail sending.
Syntax	long GetUserInfoList(long lOutCount, VARIANT rrUserID, VARIANT arrUserName, VARIANT arrDesc, VARIANT arrPassword,VARIANT arrUserLevel, VARIANT arrHPhoneNo, VARIANT arrPhoneNo, VARIANT FAR* arrMailAdd)
Reference	Sub Text1_OnClick() Dim This : Set This = Text1 'Declare variable Dim nCount'Number of registered users Dim UserIDs()'User ID Dim UserNames() 'User name Dim Descs()'User description Dim Passwords()'Password Dim UserLevels()'User level Dim HPhoneNos()'Mobile phone number Dim PhoneNos() 'Telephone number Dim MailAdds() 'Mail address Dim ret Call 'Graphic Script function (ret value is undefined) ret = GetUserInfoList (nCount, UserIDs, UserNames, Descs, Passwords, UserLevels, HPhoneNos, PhoneNos, MailAdds) Dim strMessage ‘The call result into string … strMessage = "UserID/UserName/UserDesc/Password/UserLevel/HP No/TelNO/Mail" strMessage = strMessage + vbCrlf For idx = 0 To nCount -1

	strMessage = strMessage+ UserIDs(idx) strMessage = strMessage+ "/" strMessage = strMessage+ UserNames(idx) strMessage = strMessage+ "/" strMessage = strMessage+ Descs(idx) strMessage = strMessage+ "/" strMessage = strMessage+ Passwords(idx) strMessage = strMessage+ "/" strMessage = strMessage+ cstr(UserLevels(idx)) strMessage = strMessage+ "/" strMessage = strMessage+ HPhoneNos(idx) strMessage = strMessage+ "/" strMessage = strMessage+ PhoneNos(idx) strMessage = strMessage+ "/" strMessage = strMessage+ MailAdds(idx) strMessage = strMessage + vbCrlf Next 'Check the result MsgBox strMessage End Sub

GetTagOFFSTR Function	
Description	Return ‘off string’ of the digital tag.
Syntax	GetTagOFFSTR(BSTR TagName) TagName: Designate a name of the tag.
Reference	Return “off string “of the tag whose name is designated in TagName. Off string of the tag is the string defined in the tag engineering, and it is only applied to the digital tag.

GetTagONSTR Function	
Description	Return On String of digital tag.
Syntax	BSTR GetTagONSTR(BSTR TagName) TagName: Designate a name of the tag.
Reference	Return “on string” of the tag whose name is designated in TagName. On string of the tag is the string defined in the tag engineering, and it is only applied to the digital tag

GetTagVal Function	
Description	Return the tag value.
Syntax	Variant GetTagVal(BSTR TagVarName) TagVarName: Designate a name of the tag. The tag name includes the field name of the tag. Ex: “TagVarName:FieldName” The items listed in the tag selection dialogue can be used as the field name. For the details, see 'Tag variable' described in the next clause.
Reference	Return the designated tag value in a variant type. Since the value of the returned tag is in a variant type, the type of the returned value should be used for the inspected number in order to use this appropriately. To inspect the type of the returned value, use IsNull(variant return value), IsEmpty(variant return value) or VarType(variant return value) functions. For more details on how to use these functions, see the MS VB Script manual or relevant appendix. If the type of returned variant value is null, it means it is in a bad status, that is, the tag value is not normal. If it is empty, it means there is no tag, tag property does not exist or it cannot get any tag value because of the system error.

GetTagValMAX Function	
Description	Return the maximum value of the tag.
Syntax	Variant GetTagValMAX(BSTR TagName) TagName: Designate a name of the tag.
Reference	The maximum value of the designated tag is returned in a variant type.

GetTagValMIN Function	
Description	Return the maximum value of tag.

Syntax	Variant GetTagValMAX(BSTR TagName) TagName: Designate a name of the tag.
Reference	The minimum value of the designated tag is returned in a variant type.

GetUnAckAlarmCount Function	
Description	It informs you the number of unacknowledged alarms.
Syntax	GetUnAckAlarmCount()

GetUnRecoverAlarmCount Function	
Description	It informs you the number of unrecovered alarms.
Syntax	GetUnRecoverAlarmCount()

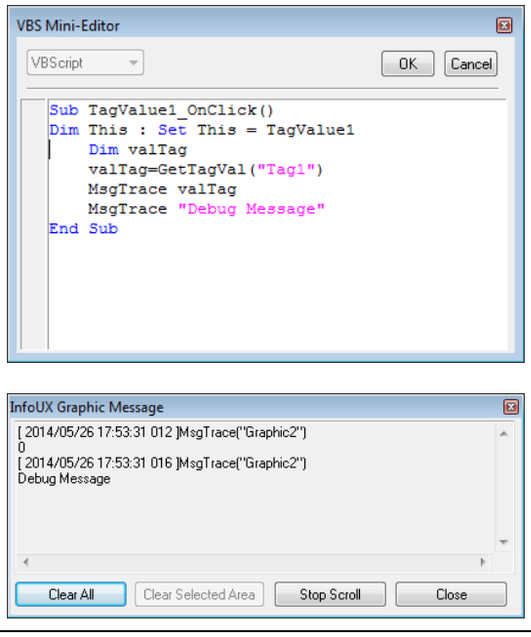
GetUserID Function	
Description	If it is called in the graphic runtime, return the ID of the user who is currently logged on.
Syntax	BSTR GetUserID()
Reference	Return the ID of the user logged on, and if there is no user logged on, the string with the length 0 is returned.

GetUserLevel Function	
Description	If it is called in the graphic runtime, return the priority of the user currently logged on.
Syntax	LONG GetUserLevel()
Reference	The level of the user (As the priority of the user authority, there are level 1 through level 255) currently logged on is returned, and if there is no user logged on, it returns the default level 255 (the lowest one).

HardCopy Function	
Description	Print the entire screen in the main window area of the Graphic runtime program.
Syntax	HardCopy()
Reference	Use the printer set up with default values.

HolidaySet Function	
Description	Show the setting screen window to set up holidays in the graphic runtime program.
Syntax	HolidaySet()

IsPageOpen Function	
Description	Inspect whether the designated page name is opened in the graphic runtime program.
Syntax	IsPageOpen(LPCTSTR PageName)
Reference	Return whether the designated file name is opened.

MsgTrace Function	
Description	It is used when writing a command formula (script) in the graphic script, and checking the message for the purpose of checking error.
Syntax	MsgTrace (VARIANT MessageValue)
Reference	

OpenPage Function	
Description	Open the graphic page.
Syntax	OpenPage(BSTR PageName, BSTR TagParam) PageName : Designate a name of the page to be opened. TagParam : Designate tags to be replaced on the template screen. If there is no tag to be replaced, designate empty string (" ").
Reference	OpenPage "Panel_GIPAM.ivd', "$MACHINE1=WB_EHV04, $MACHINE2=WB_EHV05' While opening "Panel_GIPAM.ivd" file, it converts the $MACHINE1 tag or string into WB_EHV04, and $MACHINE2 tag or string into WB_EHV05.' The template screen can be operated by using this function.

OpenPageEx Function	
Description	Open the graphic page.
Syntax	OpenPageEx(VARIANT X,VARIANT Y, VARIANT cX, VARIANT cY, BSTR PageName, BSTR TagParam)
Reference	The usage method is same as OpenPage Function, and the location to be opened can be designated. This function is not operated at local, and this supports to designate the location and size of a new page. The function which handles events about the object at HTML of web page should be realized.

OpenPageObject Function	
Description	Open the graphic page and mark the designated object.
Syntax	OpenPageObject(BSTR PageName, BSTR TagParam, BSTR ObjectName) PageName : Designate a name of the page to be opened. TagParam : Designate tags to be replaced on the template screen. If there is no tag to be replaced, designate empty string (" "). ObjectName: The name of the object to be marked

PlaySound Function	
Description	Create the sound.
Syntax	PlaySound(BSTR WaveFile) WaveFile: This is the name of a sound file. It should be designated with the path.

PulseOff Function	
Description	As the pulse (msec units) which comes to give the tag which is designated pulse off outputs.
Syntax	PulseOff(BSTR TagName, long MilliSec)

PulseOn Function	
Description	As the pulse (msec units) which comes to give the tag which is designated pulse on outputs.
Syntax	PulseOn(BSTR TagName, LONG MilliSec)

Quit Function	
Description	Quit the graphic runtime program.
Syntax	Quit()
Reference	The graphic runtime program is terminated.

QuitProject Function	
Description	Close the current project, and quit the InfoU monitoring system.
Syntax	QuitProject()
Reference	The current InfoU project is terminated and all of the processes are ended.

SaveDeskTopToBmp Function	
Description	It captures the currently running graphic runtime frame and saves it to the specified path in the format of BMP.
Syntax	void SaveDeskTopToBmp(LPCTSTR FilePathName) FilePathName

SavePageToBmp Function	
Description	It captures the currently selected graphic runtime page and saves it to the specified path in the format of BMP.
Syntax	void SavePageToBmp(LPCTSTR FilePathName) FilePathName

ReLoad Function	
Description	Load the current page again.
Syntax	ReLoad(BSTR PageName, BSTR TagParam)
Reference	The usage method of the function is same as OpenPage()Function. The difference is that the designated page is reopened in the same location with the same screen size instead of the currently loaded page. If the PageName is not designated, the currently opened page is reopened.

ReportDatabase Function	
Description	Create a report..
Syntax	ReportDatabase(BSTR ReportName)

ReportDatabaseEx Function	
Description	Create a report for the designated period.
Syntax	ReportDatabaseEx(BSTR ReportName, BSTR RefTime)

ReportDialog Function	
Description	The report is created at the designated time.
Syntax	ReportDialog()
Reference	A screen is popped up to show the type of the report set up for the current project and the list of the created reports for searching and printing.

ReportPrint Function	
Description	Print a report.
Syntax	ReportPrint(BSTR ReportName)

ReportPrintEx Function	
Description	Print a report.
Syntax	ReportPrintEx(BSTR ReportName, BSTR RefTime)

ResetStatistic Function	
Description	Reset the tag's statistical value.
Syntax	ResetStatistic(BSTR TagName) TagName: Reset a tag whose statistical data is reset.
Reference	The statistical data is reset only for the tag designated as statistical tag.

RunScript Function	
Description	Execute a script that has been written in InfoUD.
Syntax	RunScript(BSTR ScriptName) ScriptName: The script name written in InfoUD.

SendData Function	
Description	Sent the factor value to the page name designated by the graphic runtime program.
Syntax	SendData(LPCTSTR PageName, VARIANT FAR& VariantData) OnReceiveData
Reference	The factor value is delivered in a variant type to the Event Function OnReceiveData of the designated page name.

SetLogging Function	
Description	It should be logged on to the 'Event' for the change (control) in tag values and tag variable values in InfoU

	graphic runtime. However, when any specific job is implemented in the specific system, this function is available for controlling or modifying tag variable values without any need to log onto the 'Event.' After implementing SetLogging(False), execute SetLogging(True) for the safe operation record of the system. If SetLogging(False) is implemented, the event logging for the control is not performed until SetLogging(True) is implemented.
Syntax	SetLogging(BOOL TrueFalse)

SetCopyFile Function	
Description	Copy (controls) a tag value or a tag variable value.
Syntax	void SetCopyFile(LPCTSTR SourceFile, LPCTSTR NewFile) SourceFile: Copy the file path name NewFile: New file path name

SetTagVal Function	
Description	Set up (controls) a tag value or a tag variable value.
Syntax	SetTagVal(BSTR TagVarName, VARIANT Value, VARIANT_BOOL bRet) TagVarName: Tag name or tag variant value. Value: Control value(tag name intended to change) bRet: Return control(True) or No-return control (False)
Reference	Since this is a control function to change tag values, the description on the control is recorded upon the occurrence of any event. For more details on the tag variable, see the 'tag variable' provision of the manual.

ShowAlarmPage Function	
Description	Open an Alarm page. The Bin\GraSys\SysAlarmPage.ivd file of the director where InfoU is installed (ex: :\InfoU\Bin\GraSys\SysAlarmPage.ivd) is opened as a default alarm page. If the user wants to change the default alarm pgae, he/she may draw the "SysAlarmPage.ivd" page with the same name (saved in GraLS of the project) in the relevant project, then "SysAlarmPage.ivd" file of the project will be opened.
Syntax	ShowAlarmPage(BSTR AlarmTag) AlarmTag : The tag name is designated, then it finds the most recent alarm for the tag, and displays it on the alarm screen.

ShowCrossReference Function	
Description	Show the cross reference for the designated tag.
Syntax	ShowCrossReference(BSTR TagName) TagName: Designate a name of the tag.

ShowFileDialog Function	
Description	It shows the open and save dialog box for common window file.
Syntax	BSTR ShowFileDialog(VARIANT_BOOL bOpen) bOpen: It specify the dialog box of open and save file

ShowHelp Function	
Description	Open the help file.
Syntax	ShowHelp(BSTR HelpFile, long idx) HelpFile and idx parameter are not currently used. (Reserved parameters for expanding function)

ShowLayer Function	
Description	Open the layer file.
Syntax	void ShowLayer(LONG Index, VARIANT_BOOL bVisible) Index: Layer index bVisible: Whether to show selected layer

ShowNetworkStatus Function	
Description	Open the screen to show the list and communication status of the device (station)..
Syntax	ShowNetworkStatus()

ShowRawDataInfo Function	
Description	Open the screen to display the information of raw data in communication.
Syntax	ShowRawDataInfo

ShowTagView Function	
Description	Show the tag browser for the designated tag.

Syntax	ShowTagView (BSTR TagName) TagName: Designate a name of the tag.

ShowVirtualKeyboard Function	
Description	Shows the virtual keyboard window.
Syntax	ShowVirtualKeyboard()

Sleep Function	
Description	Delay the implementation of the function for the designated time.
Syntax	Sleep(long dwMilliseconds)

StartAlarmSound Function	
Description	It is the command to generate the alarm sound when an alarm occurs in the InfoU.
Syntax	StartAlarmSound()

StopAlarmSound Function	
Description	If the sound of alarm occurs, this command stops the sound.
Syntax	StopAlarmSound()

TaskBarAutoHide Function	
Description	Select whether applying the auto hide Windows task bar.
Syntax	void TaskBarAutoHide(VARIANT_BOOL bAutoHide)

TaskBarShow Function	
Description	Select whether applying the display Windows task bar.
Syntax	void TaskBarShow(VARIANT_BOOL bShow)

TimeOutMsgBox Function	
Description	Pop up the message box to be displayed for the designated time.
Syntax	TimeOutMsgBox(BSTR Msg, BSTR Title, long nSec)

UserAdd Function	
Description	Command for add users.

Syntax	void UserAdd(LPCTSTR UserID, LPCTSTR UserPassword, LPCTSTR UserLevel, LPCTSTR UserName, LPCTSTR UserDesc, LPCTSTR UserCompany, LPCTSTR UserPhoneNo, LPCTSTR UserHPhoneNo, LPCTSTR UserMailAdd, VARIANT_BOOL UserSMSCheck, VARIANT_BOOL UserMailCheck) UserID UserPassword UserLevel UserName UserDesc UserCompany UserPhoneNo UserHPhoneNo UserMailAdd UserSMSCheck UserMailCheck

UserCommand1 Function	
Description	This function is not operated at local, and it is reserved to be used for specific purpose by realizing the function that sends commands to the graphic page and handles events for the object at the HTML of the web page.
Syntax	UserCommand1(long nCmd, BSTR Str1, BSTR Str2, BSTR Str3, BSTR Str4)

UserDel Function	
Description	It deletes the user ID from the user list
Syntax	void UserDel(LPCTSTR UserID)

UserLogOff Function	
Description	Log off the logged on user.
Syntax	UserLogOff(void)

UserLogOn Function	
Description	Log on the registered user.
Syntax	UserLogOn(BSTR UserID, BSTR PassWord)

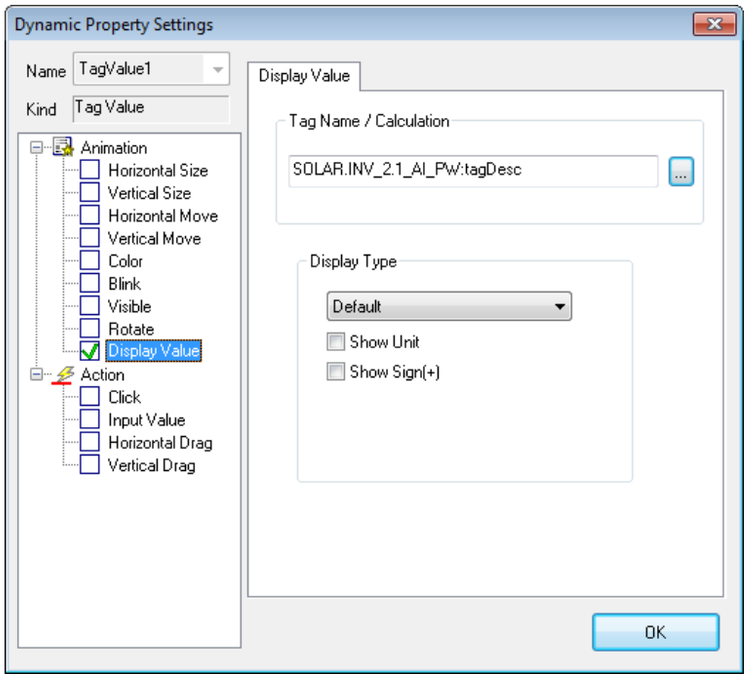
UserPassChange Function	
Description	It shows the window to change the user's password.
Syntax	void UserPassChange(void)

9.11 Tag Variable

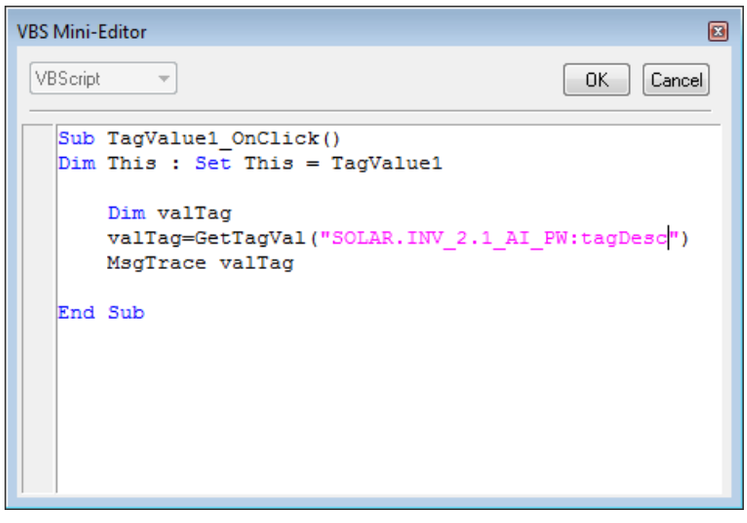
Tag variable is referred to variable or invariable related to tag defined by database. The tag variable can be referred by the same way of "Tag name: Tag variable name.".

9.11.1 Examples Used by the Dynamic Property Settings for Graphic

The functions that are available in the script of the graphic editor are composed of the standard VB script function and graphic function of the InfoU. The types of the graphic functions of the InfoU are as shown in the below table.



9.11.2 Examples of Using in Graphic Script



Tag Variable	Descriptions	Support	Remarks
tagValue	It means the current value of tag. It is the same as the tag's name is used only (If only the tag's name is inputted, the default value appears)	Read/Write	Analog/Digital/ String
tagName	Tag name	Read	
tagId	Tag unique index(long type)	Read	
tagDesc	Tag description (String)	Read/Write	
tagKind	Classification of system tags 0 : Normal tag 1 : System tag	Read	
tagType	Tag type 1:Analog, 2:Digital, 3:String, 4:Structure, 5:Block, 6:Grouop	Read	
structTagTypeId	It is the type index of the structure tag. It indicates the device type and is used to represent the tag information of the device for LSIS only. 0: The general tag does not need this information. 1 or higher: If it is structure tag, this information must be filled.	Read	
displayPrecision	It displays the decimal places. /String length	Read/Write	
tagPriority	Tag's priority value (Control ranking: 1~255)		
engValueType	It shows a type of industrial data value. 1 : bool 2 : char 3 : short 4 : long 5 : longlong 6 : unsigned char 7 : unsigned short 8 : unsigned long 9 : ulonglong 10:float 11:double 101:string type 201:raw type More than 1000:Used for structure tag type.	Read	
devValueType	It shows device value type. 1 : bool 2 : int8 3 : int16 4 : int32 5 : int64 6 : uint8	Read	I/O Analog/Digital/ String

	7 : uint16 8 : uint32 9 : uint64 10: float 11: double 12: bcd_8 13: bcd_16 14: bcd_32 15: bcd_64 16: ubcd_8 17: ubcd_16 18: ubcd_32 19: ubcd_64 20: bit_1 21: bit_2 22: bit_3 23: bit_4 101:string 201: raw		
engUnitId	String index showing the unit of Analog Tag	Read	Analog
onStrId	String index displayed when digital tag is 1	Read	Digital
offStrId	String index displayed when digital tag is 0	Read	Digital
containerId	Delimiter for tag members Parent tag's index when it is a member of structure tag	Read	Analog/Digital/ String
memTagExprFlag	Memory tag calculation flag 0 : Memory Tag is not calculated 1 : Memory Tag is calculated	Read	Memory Tag Analog/Digital
memTagExpr	Expression of memory tag	Read	Memory Tag Analog/Digital
memTagExprCycle	Expression cycle of memory tag	Read	Memory Tag Analog/Digital
initVal	Tag's initial value	Read	Analog/Digital/ String
convertRule	Method to convert raw data to industrial data 0:None, 1:Ratio/Bios, 2:Scale	Read/Write	I/O Analog Tag
rawMin	Minimum value of tag's raw data	Read/Write	
rawMax	Maximum value of tag's raw data	Read/Write	I/O Analog Tag
dataMin	Minimum value of tag's industrial data	Read/Write	Analog Tag
dataMax	Maximum value of tag's industrial data	Read/Write	Analog Tag
dataEngRatio	Ratio value to change tag values	Read/Write	I/O Analog Tag
dataEngBias	Bias value to change tag values	Read/Write	I/O Analog Tag

valueDeadBand	DeadBand value (%)	Read/Write	I/O Analog Tag
alarmDelayTime	Alarm delay time(millisecond)	Read	Analog/Digital
alarmDeadBand	Alarm Deadband (%)	Read/Write	Analog Tag
alarmKind	Alarm process kinds. It is shown in each bit. [Analog] 0000 0000 0000 0001 : Boundary value(HH) 0000 0000 0000 0010 : Boundary value(HI) 0000 0000 0000 0100 : Boundary value(LO) 0000 0000 0000 1000 : Boundary value(LL) 0000 0000 0001 0000 : Changed amount 0000 0000 0010 0000 : Main distance 0000 0000 0100 0000 : Sub distance [Digital] 0000 0001 0000 0000 : OFF 0000 0010 0000 0000 : ON 0000 0100 0000 0000 : ON->OFF 0000 1000 0000 0000 : OFF->ON 0001 0000 0000 0000 : When changed	Read/Write	Analog/Digital
alarmLevel	Value to show alarm level 0 : None(Default Value). 1~10: Actual alarm level	Read	Analog/Digital
alarmGenStrId	String index displayed when alarm is generated	Read	Analog/Digital
alarmRcvStrId	String index displayed when alarm is recovered	Read	Analog/Digital
alarmHH	Analog Tag HIHI Alarm boundary value	Read/Write	Analog
alarmHI	Analog Tag HI Alarm boundary value	Read/Write	Analog
alarmLO	Analog Tag LO Alarm boundary value	Read/Write	Analog
alarmLL	Analog Tag LOLO Alarm boundary value	Read/Write	Analog
alarmChangeValue	Absolute value of the alarm's changed value	Read/Write	Analog
alarmGapRule	Rule to process distance alarm 0: Not used, 1: Absolute value used, 2:Relative value used(designating tag)	Read/Write	Analog
alarmGapBaseValue	Base value of alarm distance(Absolute value)	Read/Write	Analog
alarmGapBaseTagId	Tag index as base value of alarm distance	Read/Write	Analog
alarmGapMain	Main fluctuation value of alarm distance	Read/Write	Analog
alarmGapSub	Sub fluctuation value of alarm distance	Read/Write	Analog
usrDefAlarmStr1Id	String 1 index of alarm defined by the user	Read	Analog/Digital
usrDefAlarmStr2Id	String 2 index of alarm defined by the user	Read	
usrDefAlarmStr3Id	String 3 index of alarm defined by the user	Read	
usrDefAlarmStr4Id	String 4 index of alarm defined by the user	Read	
stationId	Station index connected to tag 1:Memory Tag, More than 2:Station index	Read	I/O Tag

comAddr	Tag I/O address	Read	I/O Tag
stationName	Station name connected to tag	Read	I/O Tag
devId	Device index	Read	I/O Tag
devName	Device name	Read	I/O Tag
driverName	Driver name	Read	I/O Tag
engUnitStr	String showing the unit of tag	Read	Analog
onStr	String index displayed when digital tag is On(1)	Read	Digital
offStr	String index displayed when digital tag is Off(0)	Read	Digital
alarmGenStr	String index displayed when alarm is generated	Read	Analog/Digital
alarmRcvStr	String index displayed when alarm is recovered	Read	Analog/Digital
alarmGapBaseTagName	Base tag name of distance alarm	Read	Analog
pOnOffStr	String index showing On when digital tag is On while showing Off when digital tag is OFF (String)	Read	Digital
pAlarmEn	Whether to process tag alarm	Read/Write	Analog/Digital
pStatEn	Whether to process tag statistics value	Read/Write	Analog/Digital
pAlarmEnHH	Whether to process HH alarm	Read/Write	Analog
pAlarmEnHI	Whether to process HI alarm	Read/Write	Analog
pAlarmEnLO	Whether to process LO alarm	Read/Write	Analog
pAlarmEnLL	Whether to process LL alarm	Read/Write	Analog
pAlarmEnCHG	Whether to process changed alarm	Read/Write	Analog
pAlarmEnMAIN	Whether to process main distance alarm	Read/Write	Analog
pAlarmEnSUB	Whether to process sub distance alarm	Read/Write	Analog
pAlarmEnDIG	Process kind of digital alarm 1:Off, 2:On,3:On->Off,4:Off->On,5:When changed	Read/Write	Digital
tagStatus	Information on tag status (Bit)	Read	Analog/Digital/ String
pTagStatus	Information on tag status (0:Normal, 1:Alarm, 2:bad)	Read	

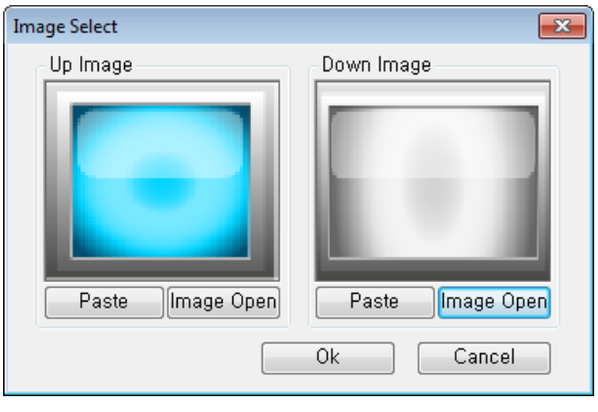
9.12 Objects Using Images

There are pictures, image buttons, and image lamps for the objects using various formats of images. You can specify the layout types and transparent color commonly. In the case of a picture object, you can use a clipboard and can specify storage method. The image button/image lamp provides the storage method based on image pixel data. In the case of the picture/image button object, you can specify the Push button style. In addition, after selecting the objects on the screen of the objects using images, you can change images in the sub-property at once.

9.12.1 Button Image

The image button object acts as the Push button. If you select this object and draw on the screen, the below setting window will be displayed. Then, you can just choose the desired image. The image can be reset by using [Sub-property] of the menu.

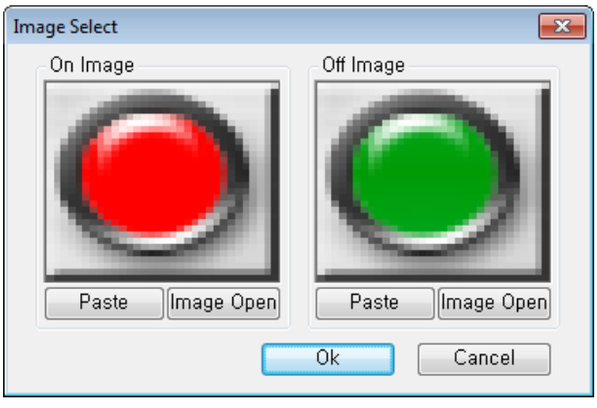
(1) Up Image: Image that you want to display when dragging the mouse up.
(2) Down Image: Image that you want to display when dragging the mouse down.
(3) Library: If you click this button, the image selection window of ‘InfoU installation path\bin\ PicImages’ path will be displayed.
(4) Image Open: If you click this button, the image selection window of GraLS path of the relevant project will be displayed.



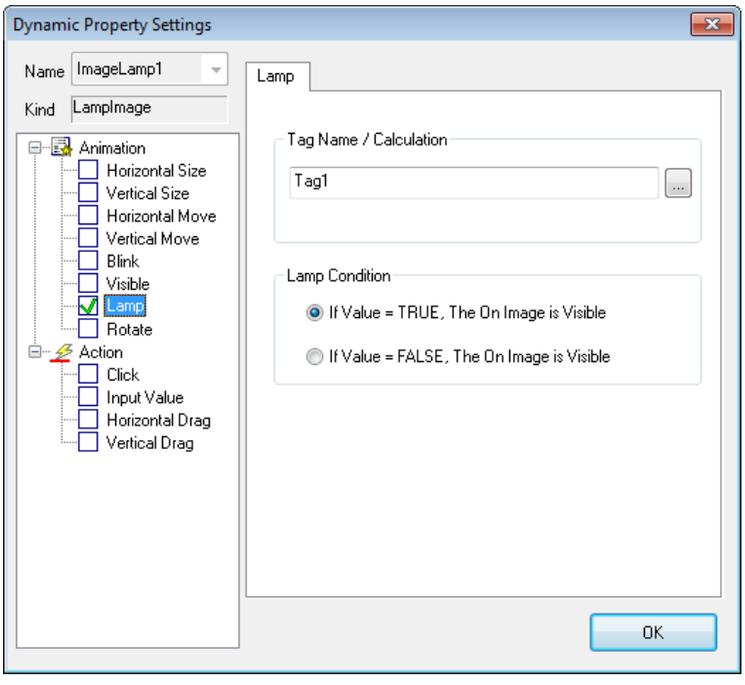
9.12.2 Lamp Image

The image lamp object acts as the lamp according to the tag value. If you select this and draw on the screen, the below setting window will be displayed. You can choose the desired image for each item and the image can be reset by using [Sub-property] of the menu.

(1) On Image: Image that you want to display in case of Mouse-On
(2) Off Image: Image that you want to display in case of Mouse-Off
(3) Library: If you click this button, the image selection window of ‘InfoU installation path\bin\ PicImages’ path will be displayed.
(4) Image Open: If you click this button, the image selection window of GraLS path of the relevant project will be displayed.



The figure shown blows the setting window for the lamp dynamic property. Depending on the input tag value during runtime, On Image or Off Image is displayed.



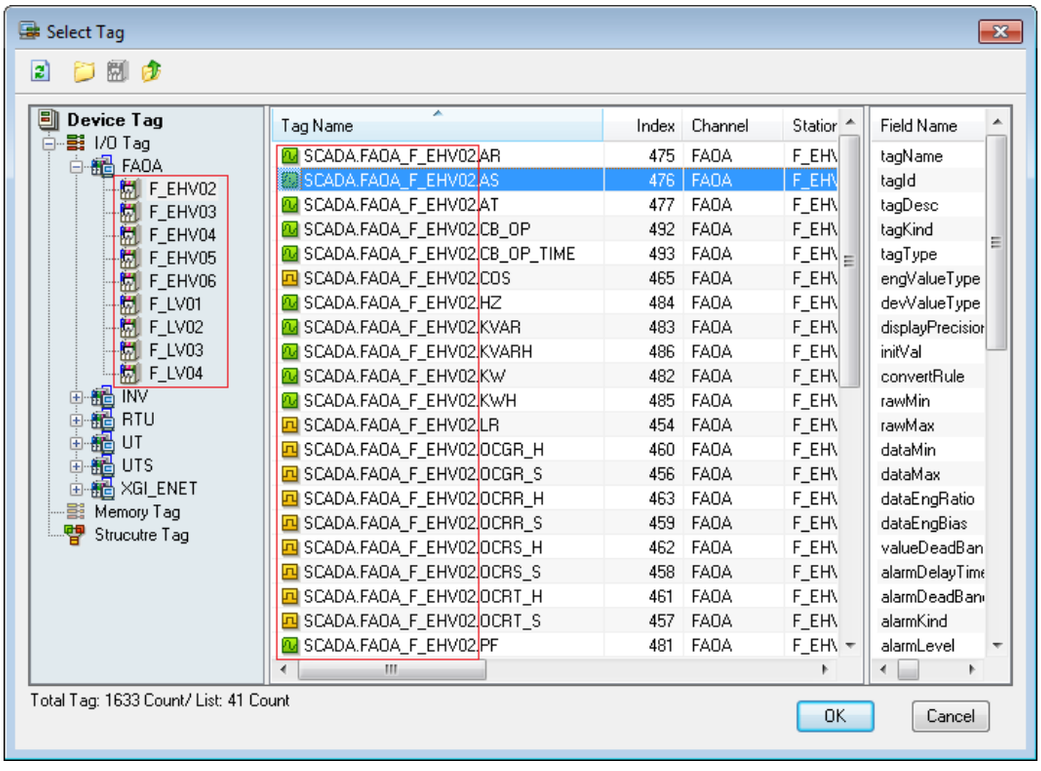
9.13 Smart Symbol

When configuring the dynamic properties of objects individually, the smart symbol aims at reuse of each tag name by arranging the alternative string for structure tags or the specific information excluding the member of a group in the almost same way as the instance method. The merit of the smart symbol is to define the object only once. Namely, you can just specify the alternative string every time you put the registered smart symbol on the page.

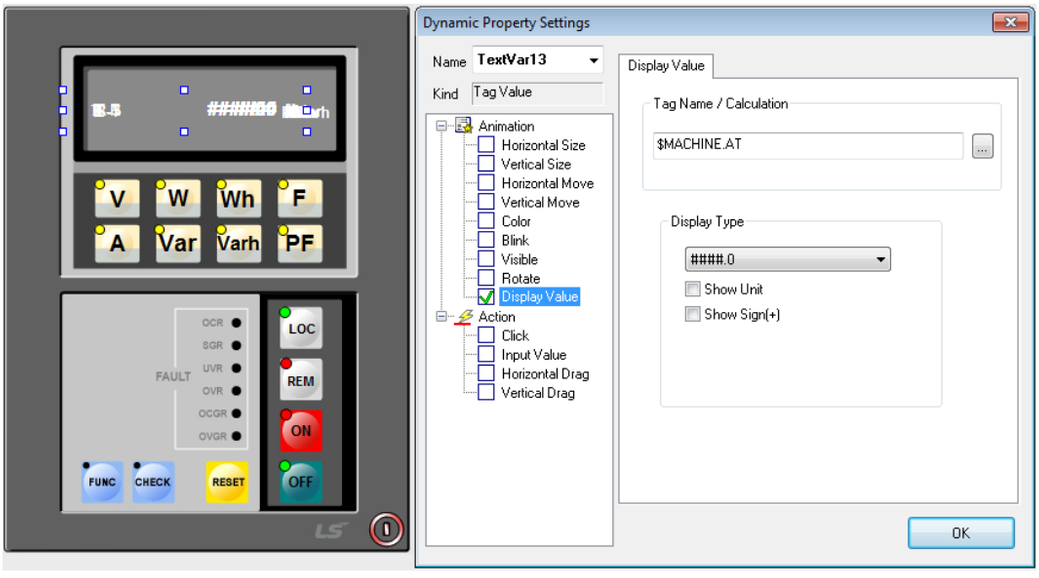
9.13.1 Creating Smart Symbol

A user can draw a desired picture like a regular object. During tag mapping of the dynamic properties of each object, it maps the structure tags or the specific information excluding the member of a group into the alternative string. For smart symbolization, you need to put $ in front of the string to be replaced. If there is no $ separator, you cannot use the window to set the smart symbol instance.

The figure shown blows the tag management of engineering. As shown in a blue mask, the tag name is composed of communication channel + station + member. The member name is repeated according to fixed rules so you can just convert communication channel + station string into the alternative string.

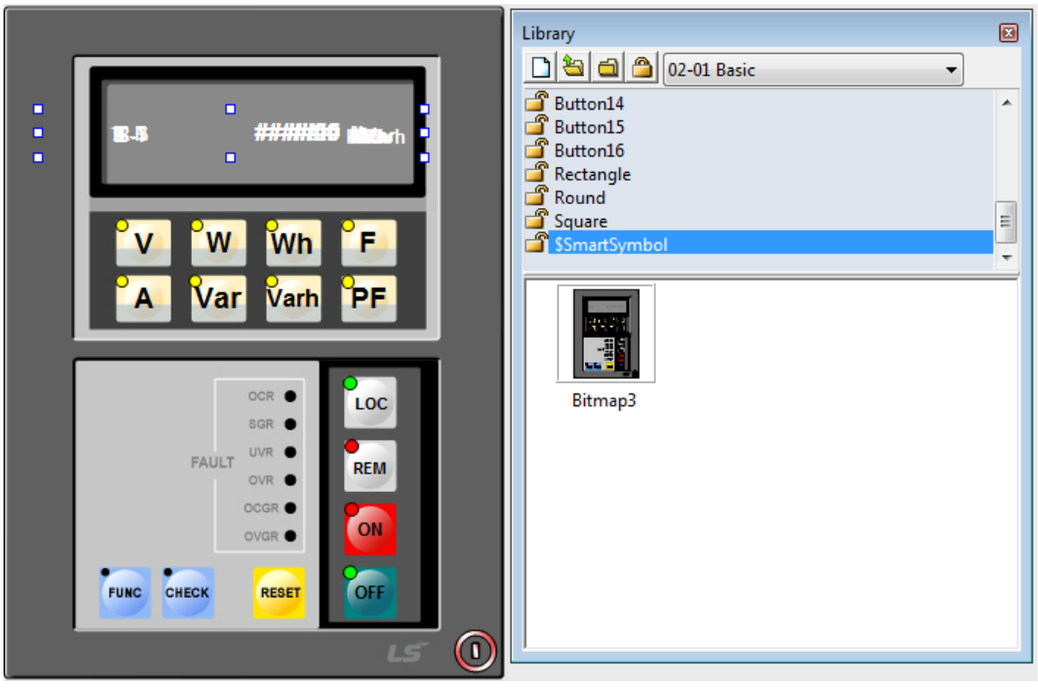


The below figure shows the example of mapping the tag name of the dynamic property into the alternative string (communication channel + station-> $GIMAC) + member.



9.13.2 Registering and Importing the Smart Symbol in the Library

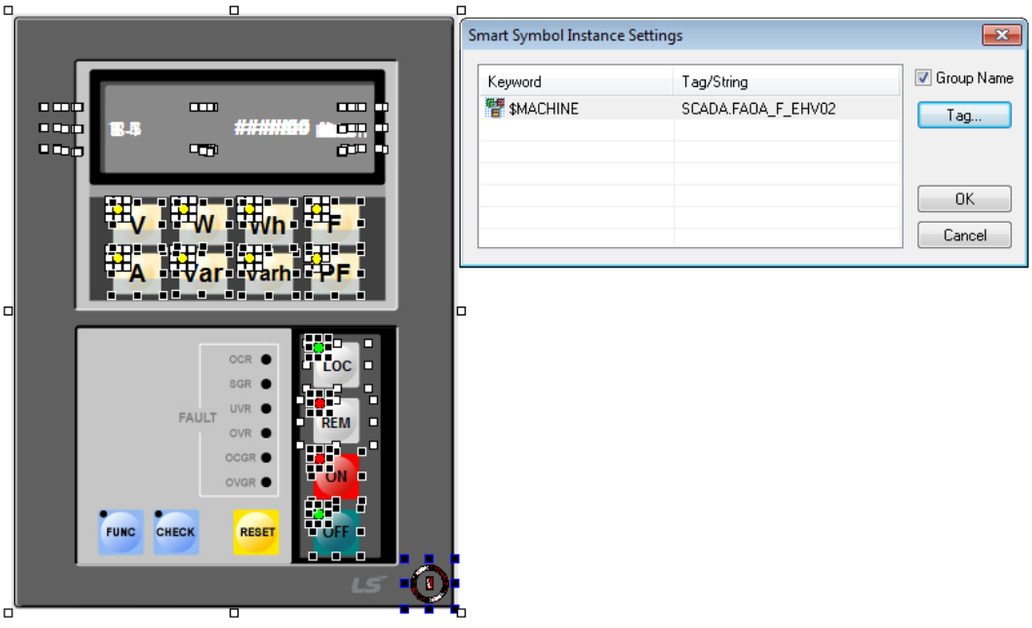
For registration, you can just copy/paste the object made with smart symbol to the library or drag & drop it as if you register general symbols. To use the smart symbol, if you copy/paste the smart symbol to the page or drag & drop it, the smart symbol instance setting window will be displayed. You cannot use the smart symbol setting window for items other than smart symbols.



9.13.3 Smart Symbol Instance Settings

In order to import smart symbols registered in the symbol library to the relevant page and apply them to the runtime, you need to change the keywords of smart symbols into the alternative string by using the smart symbol instance setting window.

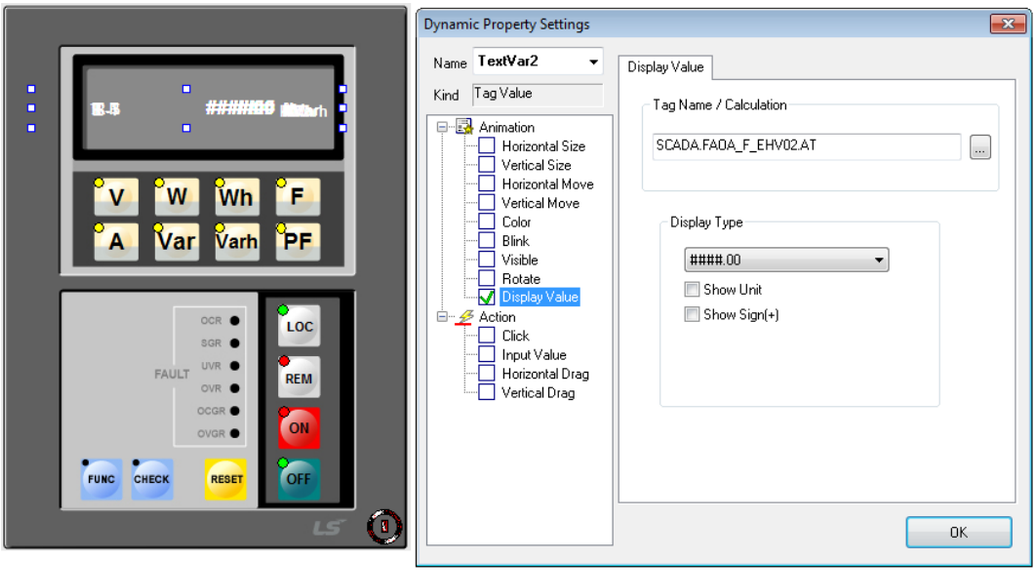
If there is no separator $ in front of the first letter of the keyword, it cannot be displayed in the smart symbol instance setting window. The figure shown blows the process converting the keywords into the alternative string through the smart symbol instance setting window.



(1) Keyword: It shows the list of strings (keywords) to be replaced, which are found in the smart symbol object.

(2) Alternative string: Enter the string to replace the corresponding keyword.

The below figure shows the changes in the tag name of dynamic properties of each object after being replaced with the alternative string through the symbol instance setting window.

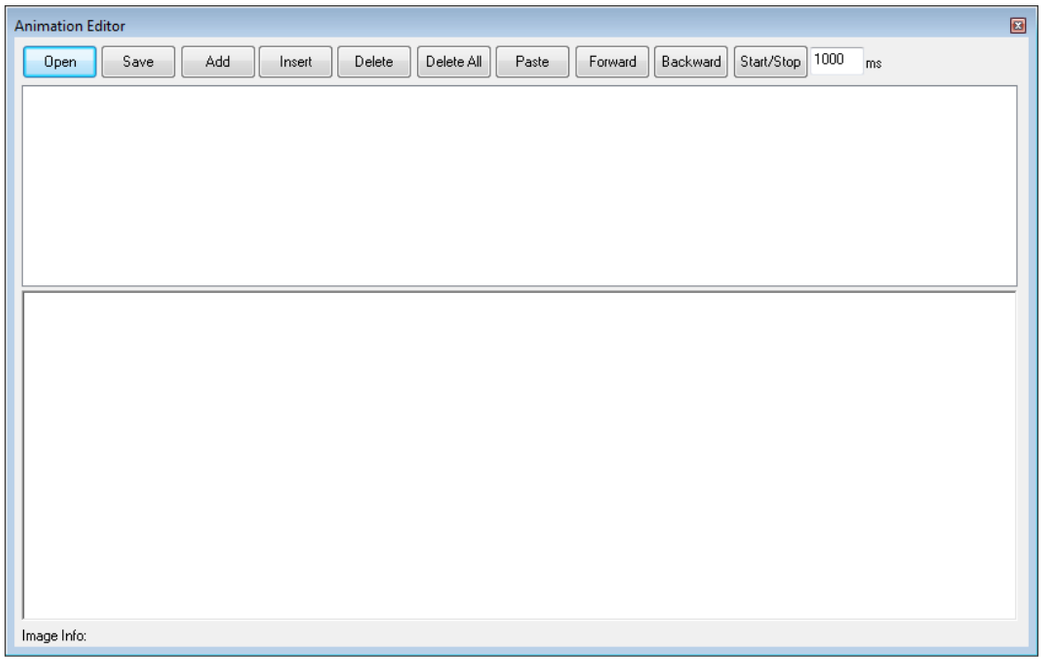


9.14 Animation Editor

The InfoU animation editor is the animation editor program to display the monitoring screen more vividly by configuring different images with the multi frame without vexatious processes such as program download or installation. The InfoU animation editor provides various file formats (BMP, JPG, GIF, PNG, and TIF) to add/insert graphics but for the file format to read/save, TIF supporting the multi-frame is available only. You can easily add, replace and delete a number of image frames. The 'Preview' function of milliseconds is also provided

9.14.1 Execution and Composition of the Animation Editor

If you select [Tool] → [Animation Editor] in the menu, the below animation editor window will be displayed.
The animation editor is the tool to edit and create animation picture objects, which is composed of [Open],[Save],[Add] ,[Insert],[Delete],[Delete All],[Paste],[Forward],[Backward],[Start/Stop].

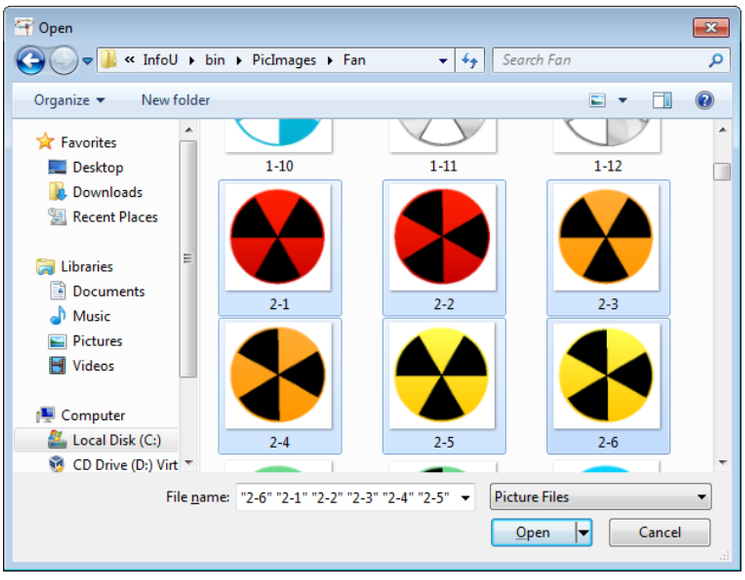


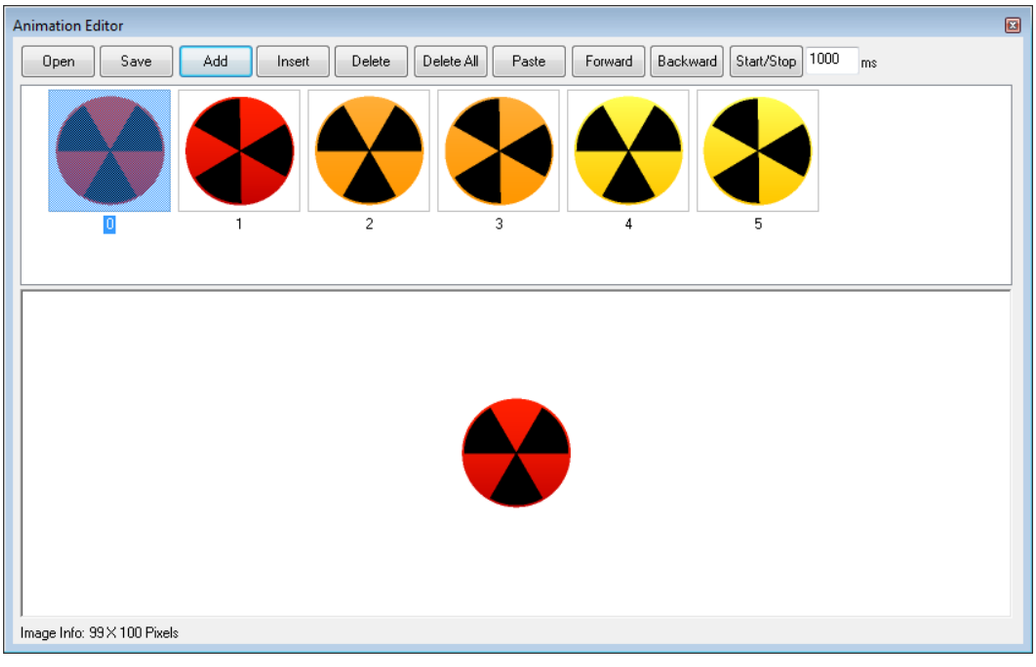
9.14.2 Creation of Animation

First of all, you need to prepare several image files with the same specification (width, height) or object to be used for each frame in the InfoU graphic editor. Generally, if the individual image and object has different sizes, image distortion may occur.

(1) Add Frame

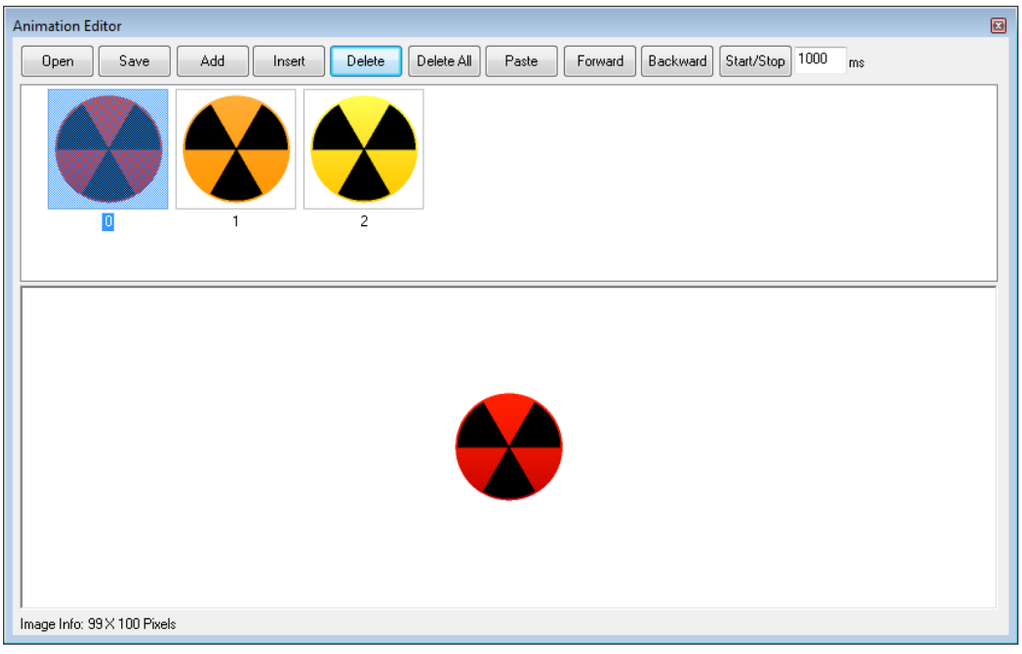
You can select several images with the [Add] button and add them to the frame or insert the image into the frame of the relevant sequence with the [Insert] button. To add the InfoU object to the frame, select the [Paste] button in a clipboard.





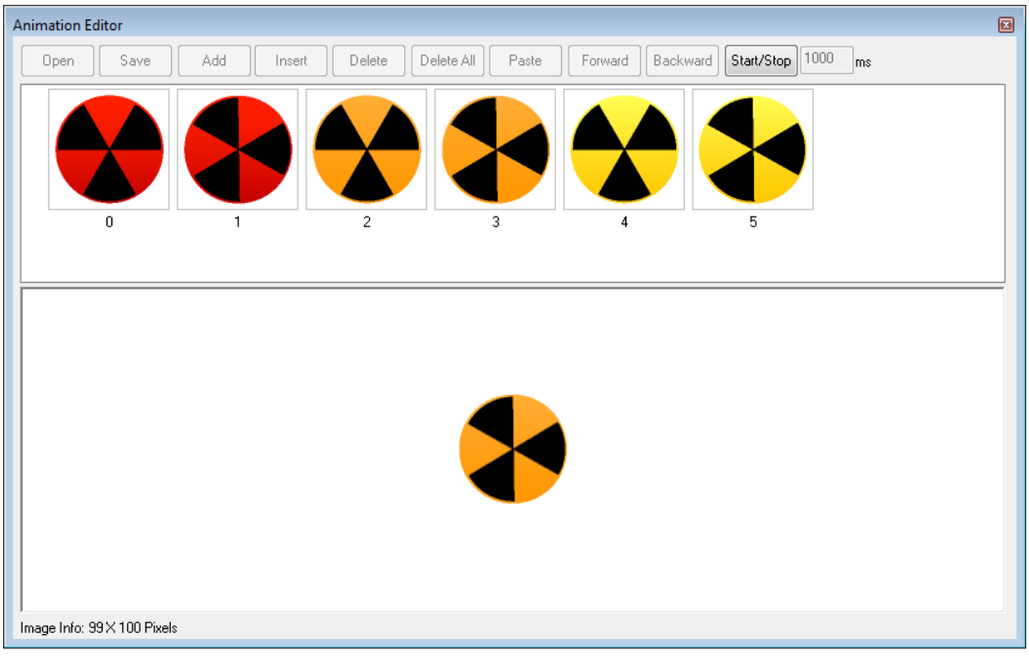
(2) Delete and Move Frame

To delete the whole frames, you can use the [Delete All] button. To delete the frame individually, after selecting the frame, press the [Delete] button. If you want to change the order of animated frames, you can use the [Forward] and [Backward] button.



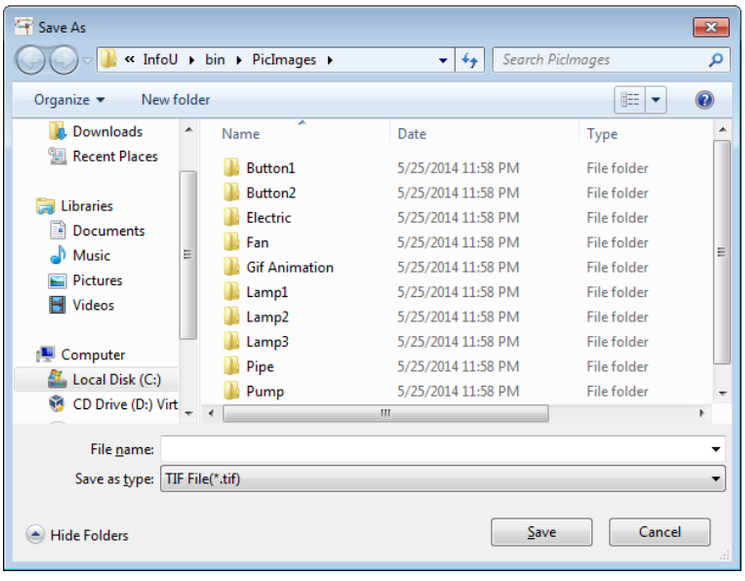
(3) Preview

If you select the thumbnail image of the frame, you can see the information on the width, height of the frame image. After specifying the speed (millisecond), if you select the [Start/Stop] button, you can make preview the same animation conditions as the real one.



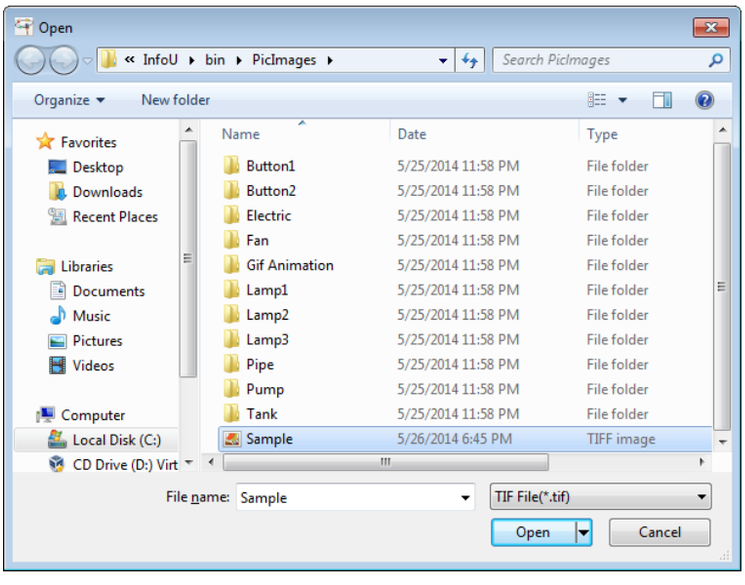
(4) Save (TIF)

If you select the [Save] button, the below dialog box where you can enter the file name to save will be displayed. After entering the file name, if you select the 'Save' button, the current animation file will be saved as the input name. For the format to save the graphic file TIF is available only and each frame image is preserved as it is.



(5) Open (TIF)

If you select the [Open] button, the below dialog box to open the file will be displayed. The available format to import the graphic file is TIF only. After selecting the file, if you press the 'Open' button, the animation frames will be displayed. Each frame image is preserved as it is and it can be added / edited.



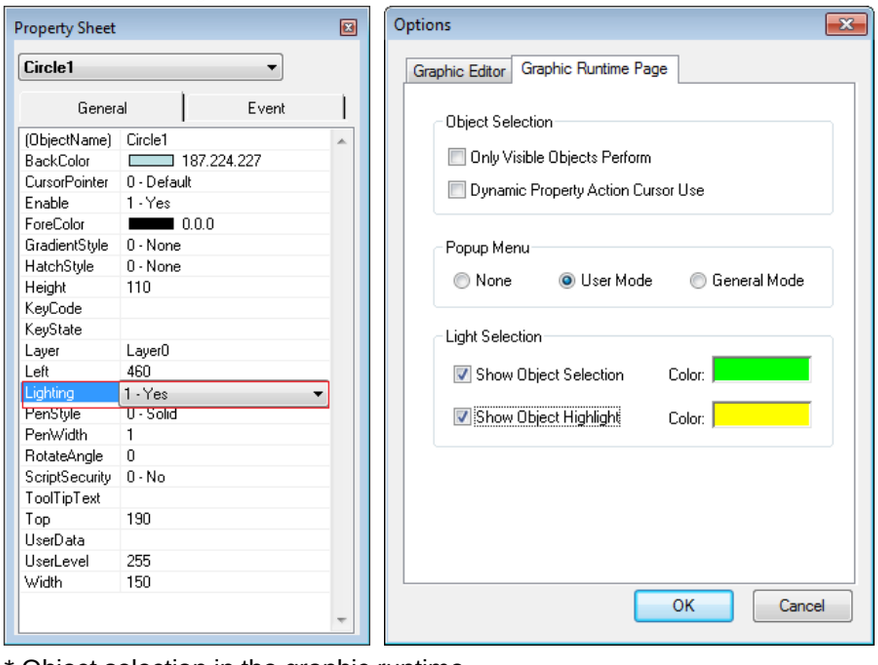
9.15 Lighting Function

It is the function to batch-control the tags mapped for the selected objects or highlight the objects with the same mapping ag by providing multi-selection of objects and highlight function in the graphic runtime.

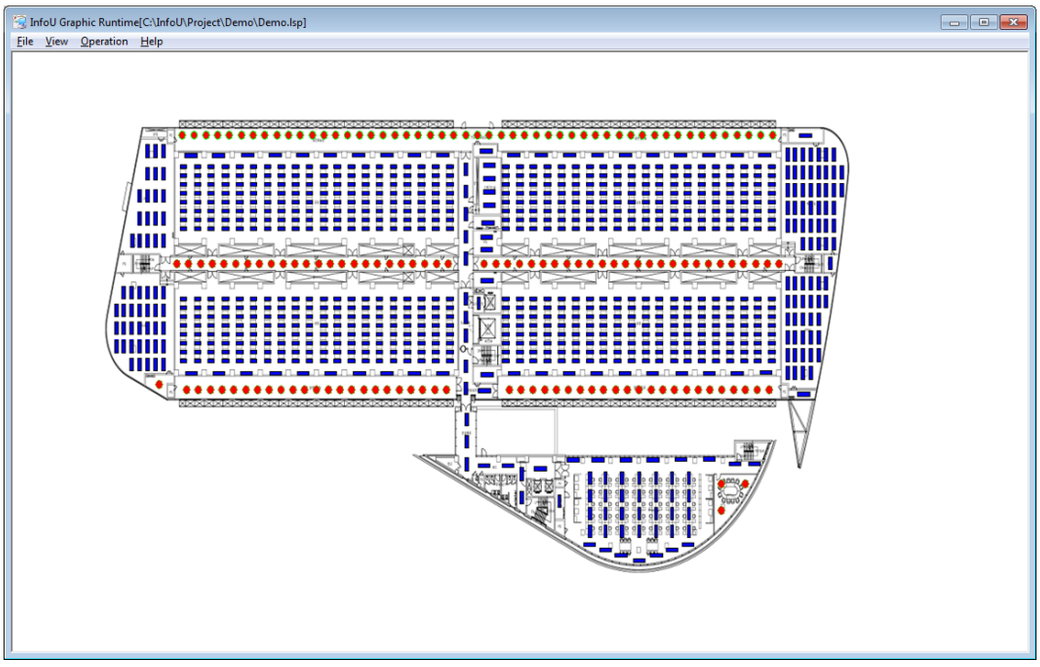
9.15.1 Object Selection Settings in the Graphic Runtime

Set the lighting property as 'Yes' in general property window of the object. Then, in the graphic runtime page tab of [Tools] → [Options], select 'Show Object Selection' and 'Color'. In the graphic runtime, multi-selection for the specified color is possible. If you set the popup menu mode as 'User Mode', you can control tags in a lump.

* Settings in the graphic editor



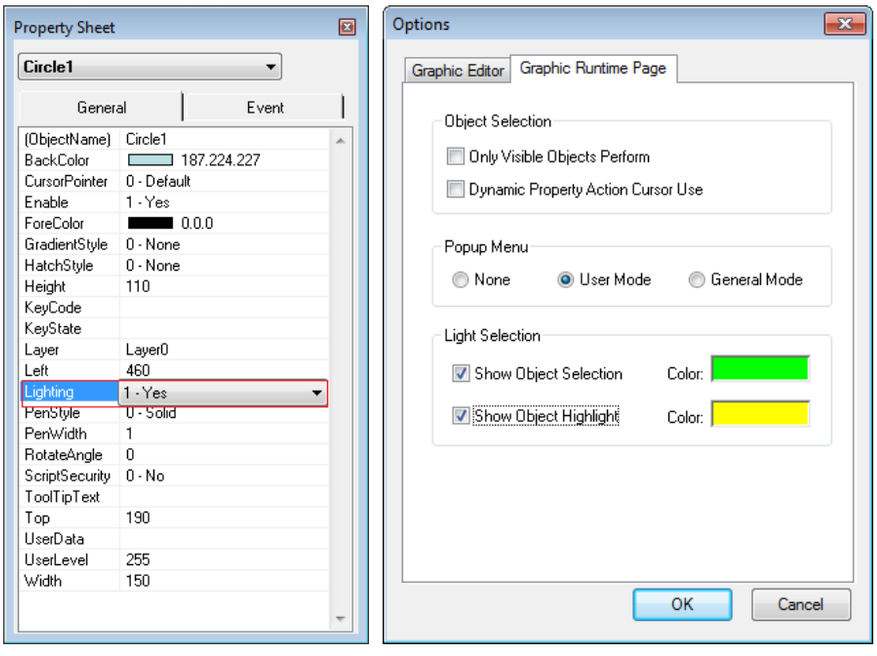
* Object selection in the graphic runtime

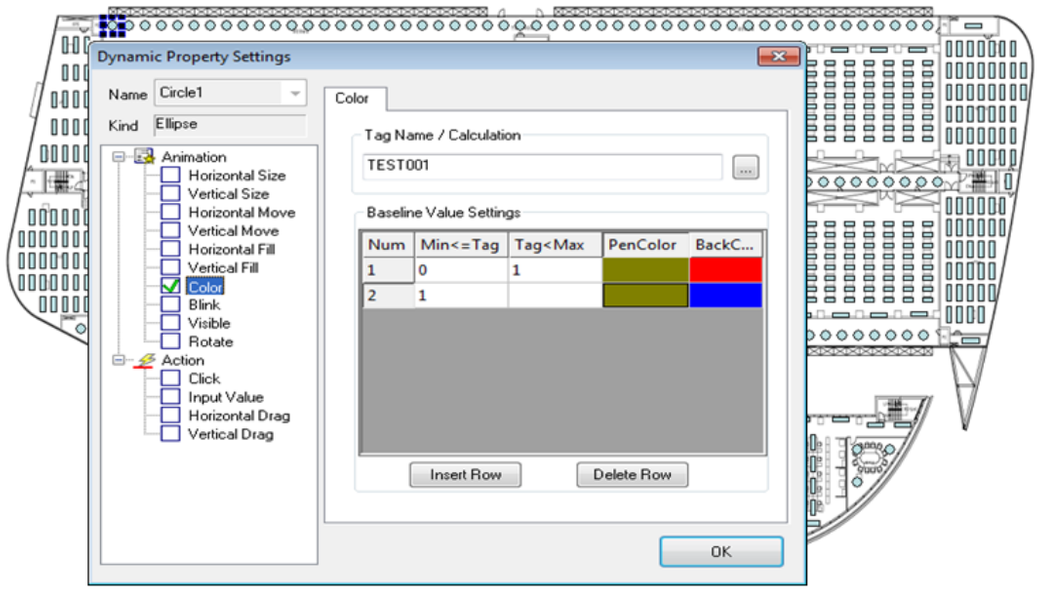


9.15.2 Object Highlight Settings in the Graphic Runtime

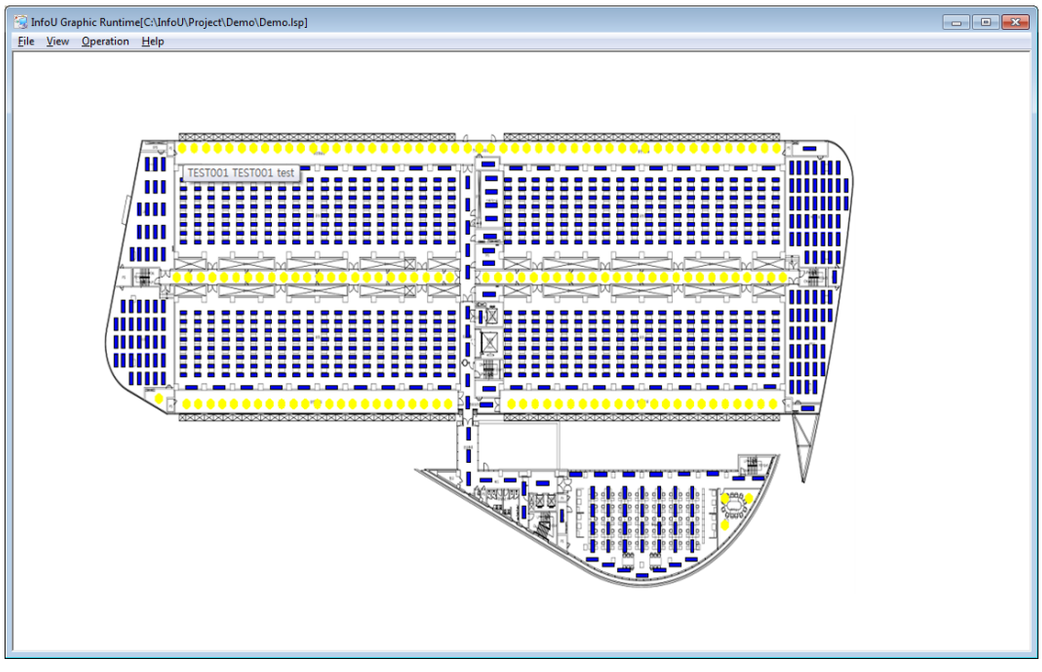
Set the lighting property as 'Yes' in general property window of the object. In the color section of the dynamic property setting window, after setting tag, then, in the graphic runtime page tab of [Tools] → [Options], select 'Show Object Highlight' and 'Color'. It is the function to extract the mapped tag from the color dynamic property of the object selected by moving the mouse in the graphic runtime and then, to highlight the mapped objects with the same tag on the screen.

* Settings in the graphic editor





*Highlight in the graphic runtime

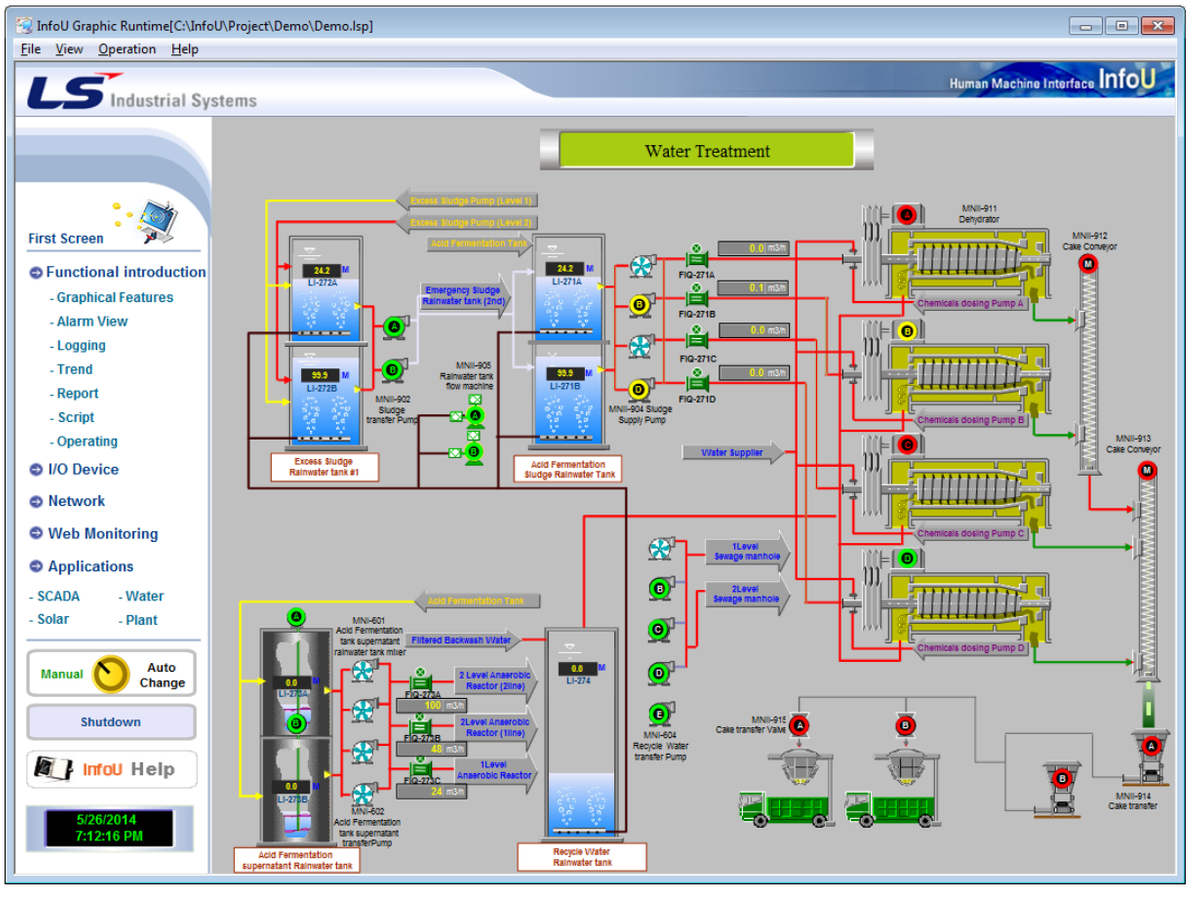


Chapter 10 Graphic Runtime

This Graphic Runtime helps to monitor or control real-time or history information through the graphic screen by using HMI real-time or history information.

10.1 Run

This Graphic Runtime runs if the runtime running starts and the initial screen are created in the Graphic Runtime settings.



10.2 Menu Composition

Upper-level Menu	Menu	Description
File	Default Page	Open the default page set up in the runtime execution environment.
	Open	Open existing document.
	Close	Close the current document.
	Close All	Close all of documents.
	Hard Copy	Print the graphic runtime execution screen in a type of hard copy.
	Print	Print document.
	Print Setup	Set up print options
	Exit	End the graphic runtime.
View	Standard Bar	Show or hide color tools.
	Status Bar	Show or hide status tools.
	Menu	Show or hide main menus.
	Zoom	Zoom in/out the page in set up ratios.
	Auto Adjust	Automatically adjust the screen size to a certain ratio fit to the current view size.
	Navigation	Show or hide navigation tools for screen moving.
	Full Screen	Convert the page into a full screen size.
Operatin	User Login	Log in the registered user.
	User Logout	Log out the logged in user.
	Change Password	Change the log-in password of the registered user.
	Alarm Popup	Show or hide the alarm pop-up screen.
	Tag View	Show all of the tags registered in DB
	Cross Reference View	Show Cross Reference information on the current project.
	I/O Station Status	Show the communication status of the currently connected I/O driver.
	Graphic Editor	It shows the currently open page in the graphic editor.
Help	Graphic Runtime Information	Show the version information of the graphic runtime.

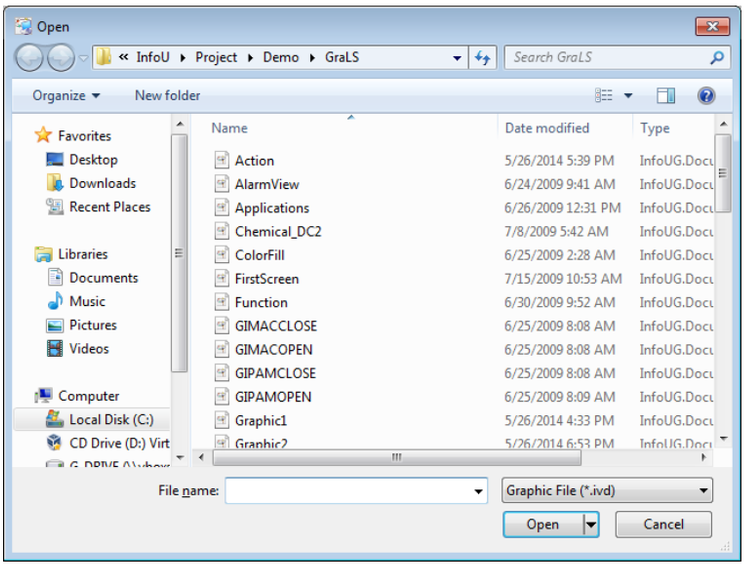
10.2.1 File

(1) Default Page

Open the default page set up in the runtime execution environment.

(2) Open

This command is used to open one of the existing documents. If this command is made, a dialog box appears to allow the user to designate a file to open from it.



(3) Close

The current document is closed.

(4) Close All

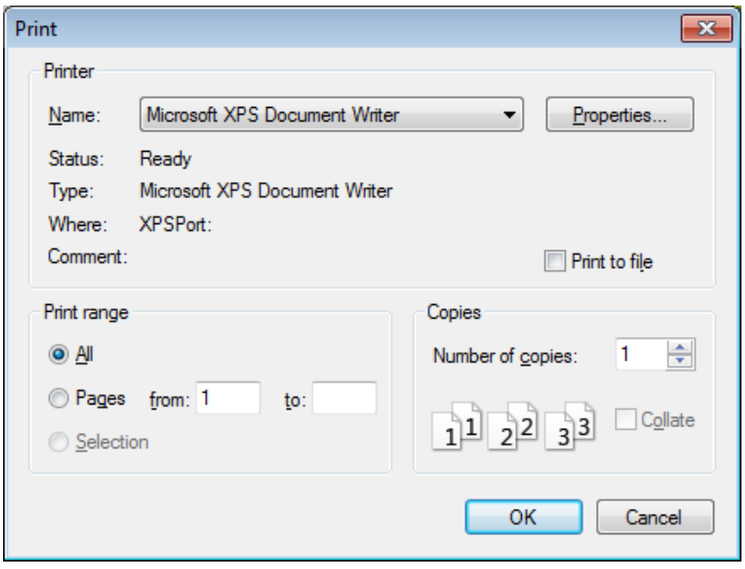
The currently opened documents are all closed.

(5) Hard Copy

Execute the graphic runtime execution screen as hard copy.

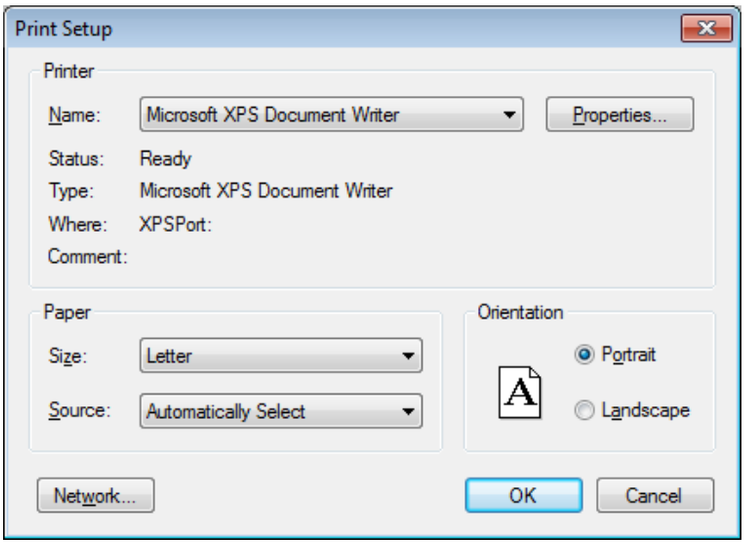
(6) Print

It is used when printing document. If this command is executed, “print” conversation box appears to allow the user to select a print range, number of copies, printer and printer option.



(7) Print Setup

Paper size and orientation for printing documents can be selective.



(8) Exit

The graphic editor is ended.

10.2.2 View

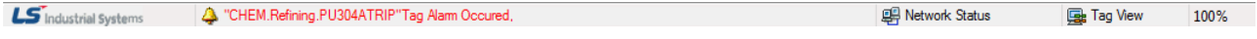
(1) Standard Bar

It is used when showing or hiding standard tools. Standard tools are command icons the most frequently used in the graphic editor. When they are shown, checkmark appears in front of the relevant menu.



(2) Status Bar

It is used when showing or hiding status tools. Status bar provides Help Message. When it is shown, checkmark appears in front of the relevant menu.



(3) Menu

It is used when showing or hiding main menus. When it is shown, checkmark appears in front of the relevant menu.

(4) Zoom

It is used when enlarging or reducing the screen in a predefined ratio during the page runtime.

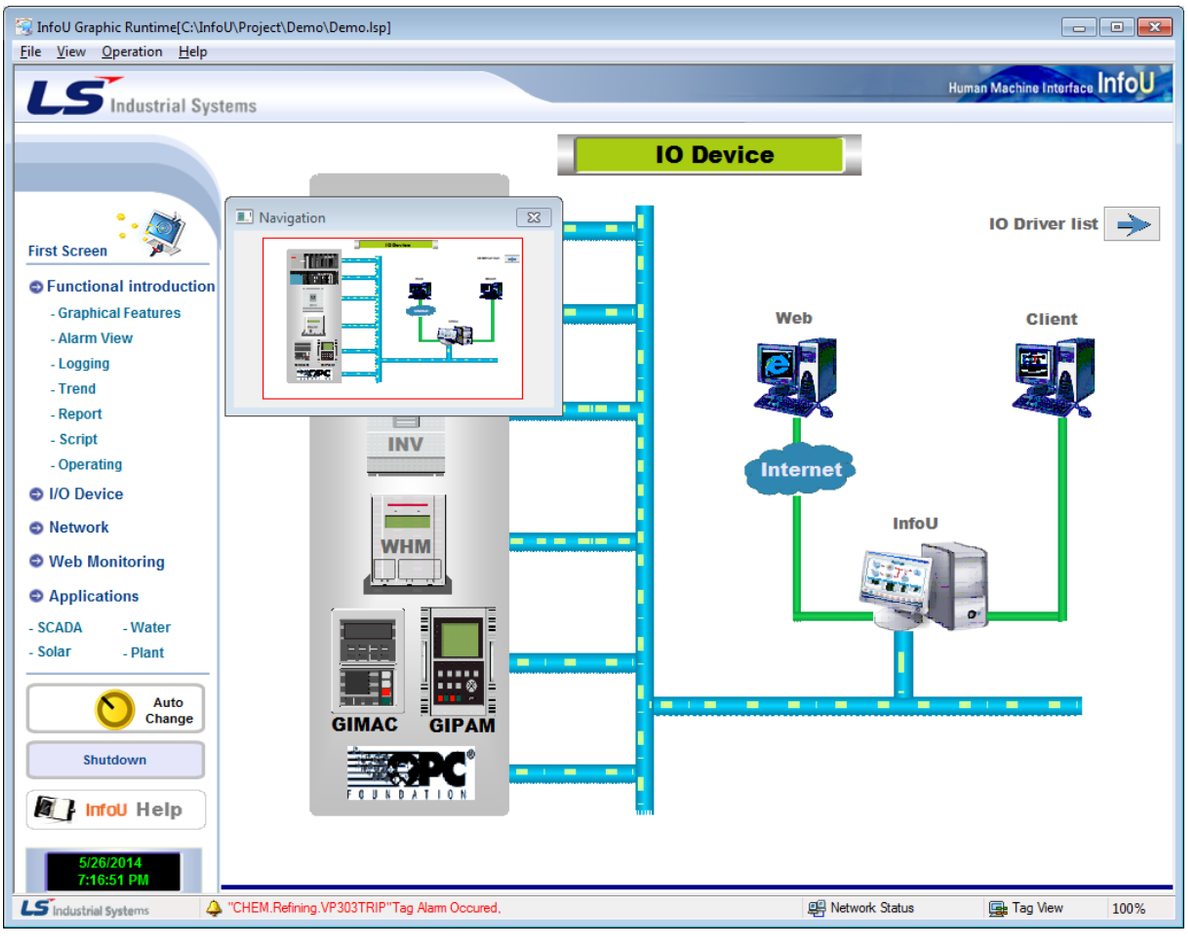
(5) Auto Adjust

It is used when adjusting the view size of the current runtime screen to a certain ratio or setting up to automatically adjust the horizontal and vertical proportions. When it is shown, checkmark appears in front of the relevant menu.

(6) Navigation

It is used when showing or hiding navigation tools.

If the current view screen is narrow, this navigation is used to move it to a desired screen area. When it is shown, checkmark appears in front of the relevant menu.



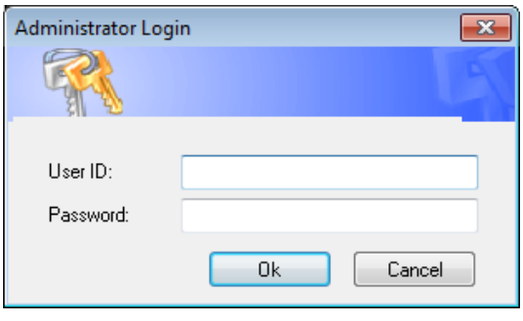
(7) Full Screen

It is used when displaying the current page in a full screen size.

10.2.3 Operation

(1) User Login

It is used when the registered user logs in. If login is done, the user may use the functions assigned to him/her. If this command is performed, the following InfoU Administrator Login window is displayed.



(2) User Logout

It is used when the logged in user logs out.

(3) Change Password

It is used when changing the registered user's login password.

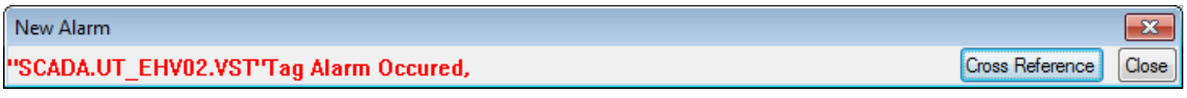
(4) Alarm Popup

It is used when showing or hiding the alarm pop-up screen.

This menu is activated only when the user logs in.

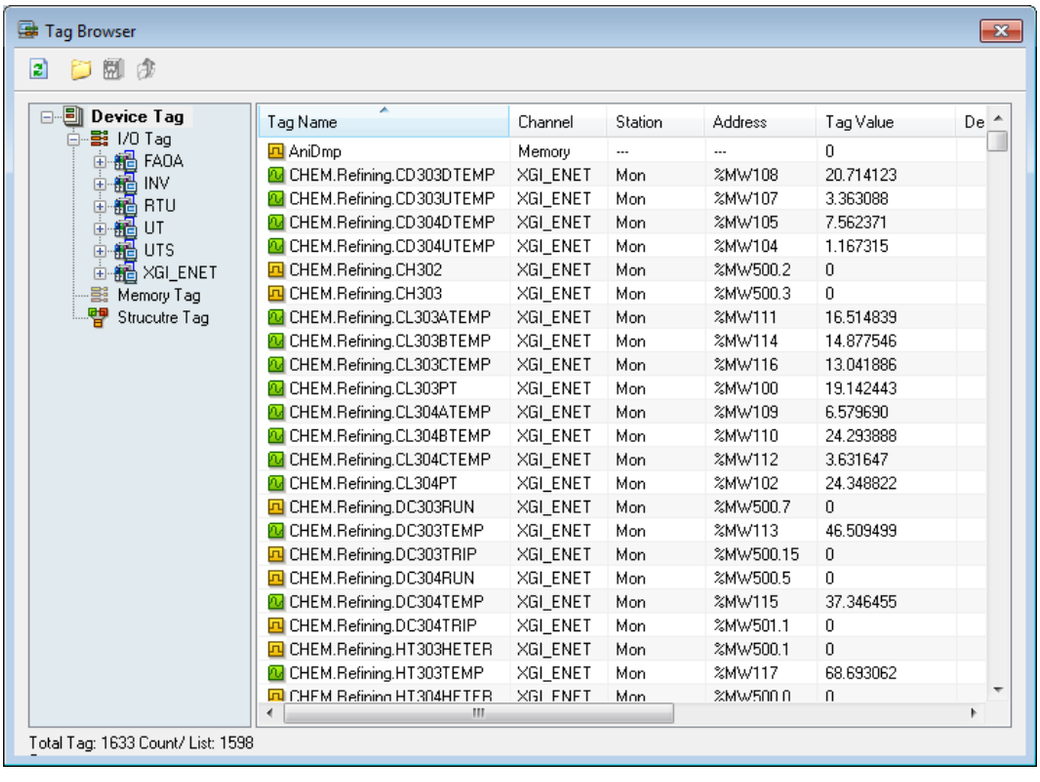
If this item is not checked, the following alarm pop-up window appears to call the operator's attention in case of the occurrence of a new alarm in the system.

If the Cross Reference button on the right is clicked, the Cross Reference Search screen is displayed to move to the Accident Analysis or Plant screen (See Cross Reference Function)



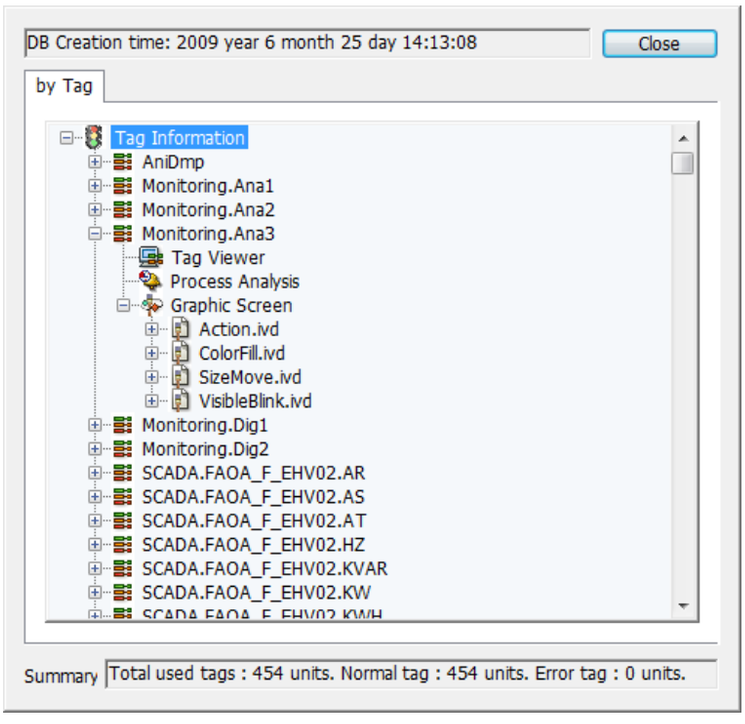
(5) Tag View

Information on the all of the tags registered in DB and status of tag values are displayed. Once this command is executed, the tag browser dialog box appears.



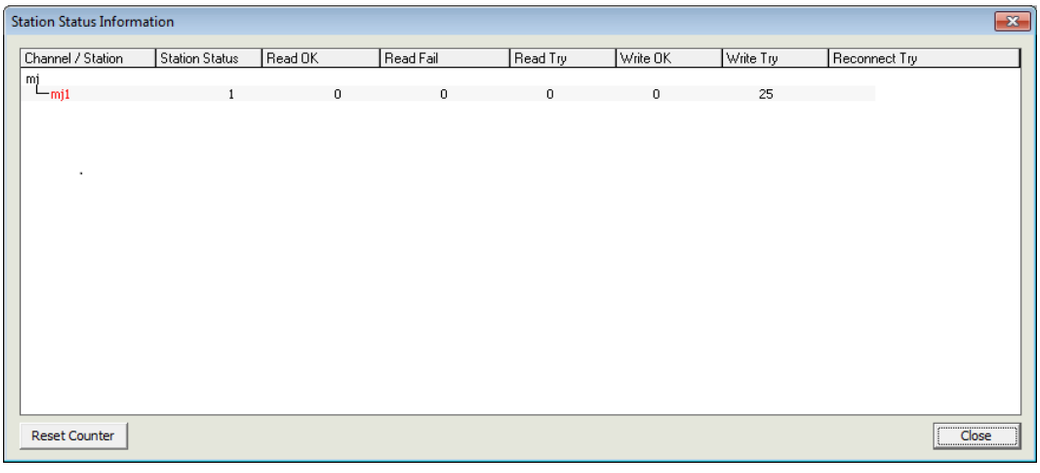
(6) Cross Reference View

The Cross Reference information on the current project is displayed. Once this command is executed, the Cross Reference View dialog box appears. For details, see 'Cross Reference'.



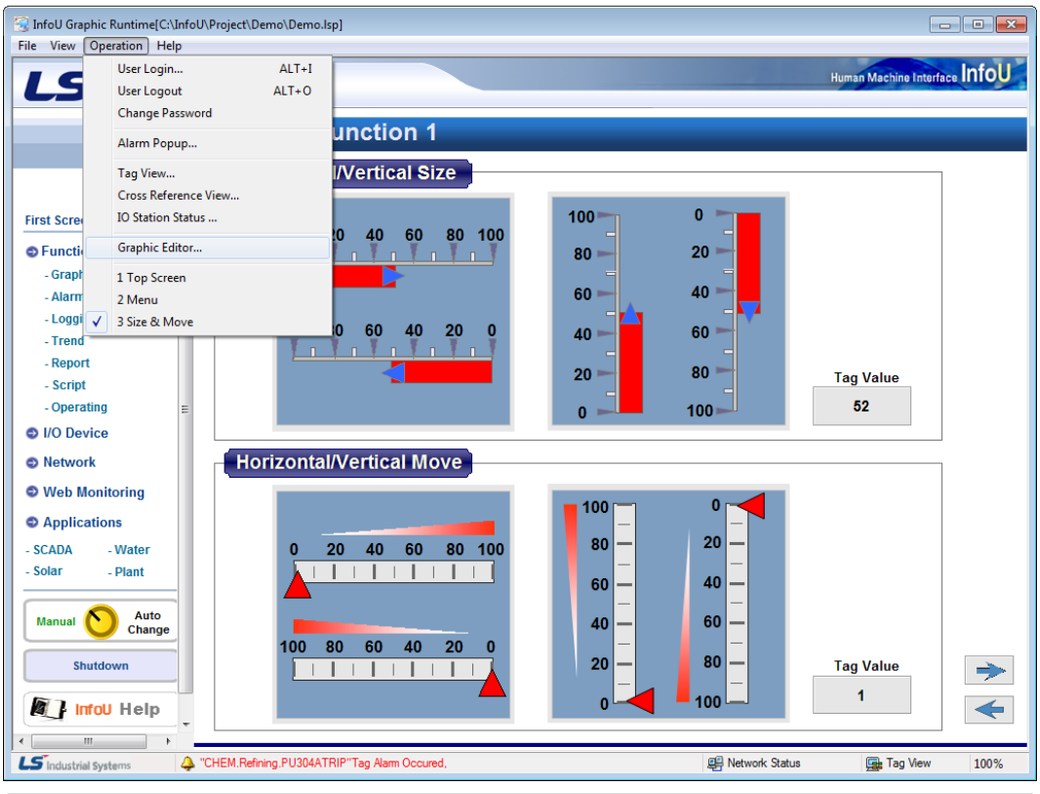
(7) I/0 Station Status

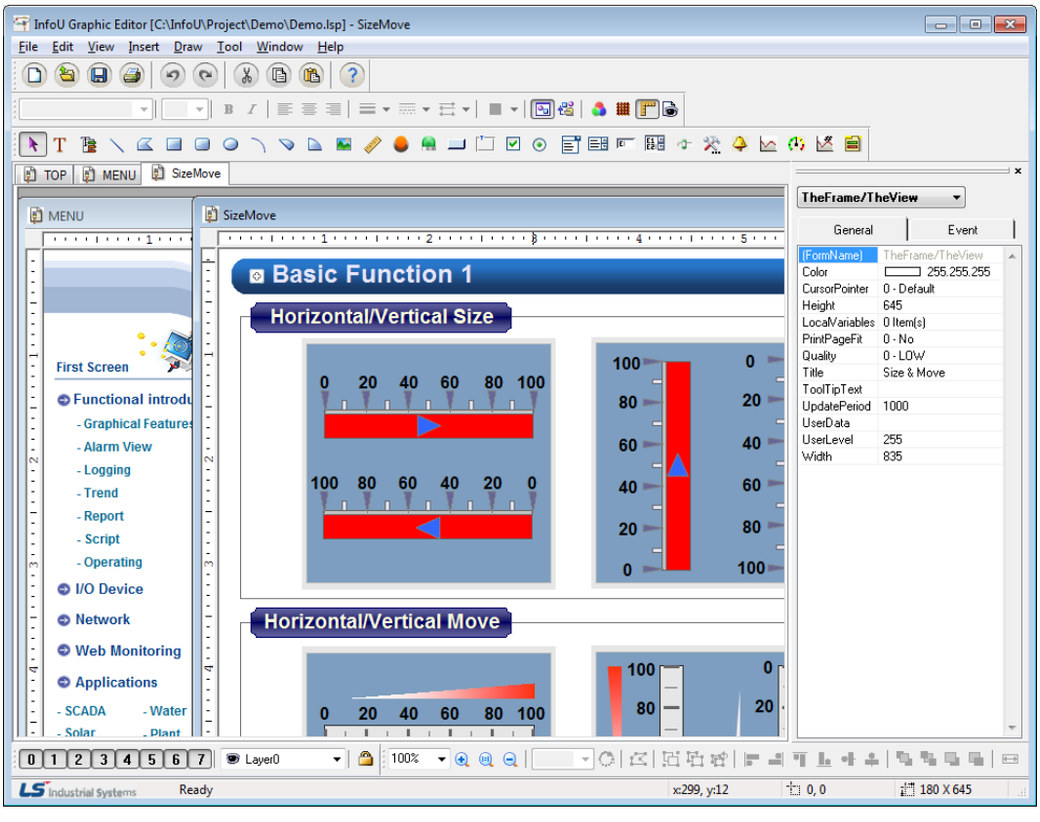
The communication status of the currently connected I/O driver is displayed. Once this command is executed, the I/O Station View dialog box appears.



(8) Graphic Editor

If you execute the [Graphic Editor], the pages that are opened in the current graphic runtime will move to the graphic editor. In case the graphic editor does not run, execute it and open the page. If you execute the [Graphic Editor], you can see the below.

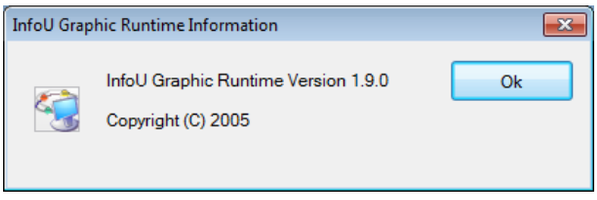

InfoU Graphic Runtime[C:\InfoU\Project\Demo\Demo.lsp]
File View Operation Help
User Login... ALT+I
User Logout ALT+O
Change Password
Alarm Popup...
Tag View...
Cross Reference View...
IO Station Status ...
Graphic Editor...
1 Top Screen
2 Menu
3 Size & Move
Human Machine Interface InfoU
unction 1
/Vertical Size
Tag Value
52
Horizontal/Vertical Move
Tag Value
1
First Scre
Functi
- Graph
- Alarm
- Loggi
- Trend
- Report
- Script
- Operating
I/O Device
Network
Web Monitoring
Applications
- SCADA - Water
- Solar - Plant
Manual Auto Change
Shutdown
InfoU Help
"CHEM.Refining.PU304ATRIP"Tag Alarm Occured.
Network Status
Tag View
100%


InfoU Graphic Editor [C:\InfoU\Project\Demo\Demo.lsp] - SizeMove
File Edit View Insert Draw Tool Window Help
TOP MENU SizeMove
MENU
SizeMove
Basic Function 1
Horizontal/Vertical Size
Horizontal/Vertical Move
First Screen
Functional introdu
- Graphical Features
- Alarm View
- Logging
- Trend
- Report
- Script
- Operating
I/O Device
Network
Web Monitoring
Applications
- SCADA - Water
- Solar - Plant
TheFrame/TheView
General Event
(FormName) TheFrame/TheView
Color 255,255,255
CursorPointer 0 - Default
Height 645
LocalVariables 0 Item(s)
PrintPageFit 0 - No
Quality 0 - LOW
Title Size & Move
ToolTipText
UpdatePeriod 1000
UserData
UserLevel 255
Width 835
Layer0
100%
Ready
x:299, y:12
0, 0
180 X 645

10.2.4 Help

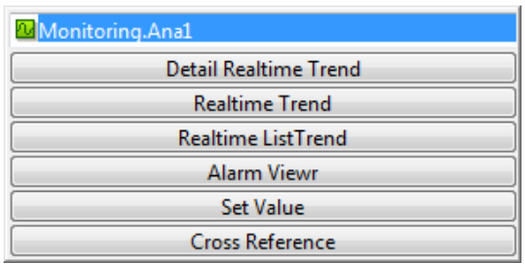
(1) Graphic Runtime Information

It is used to show the version information of the graphic runtime. If this command is executed, version information and copyright message appear.



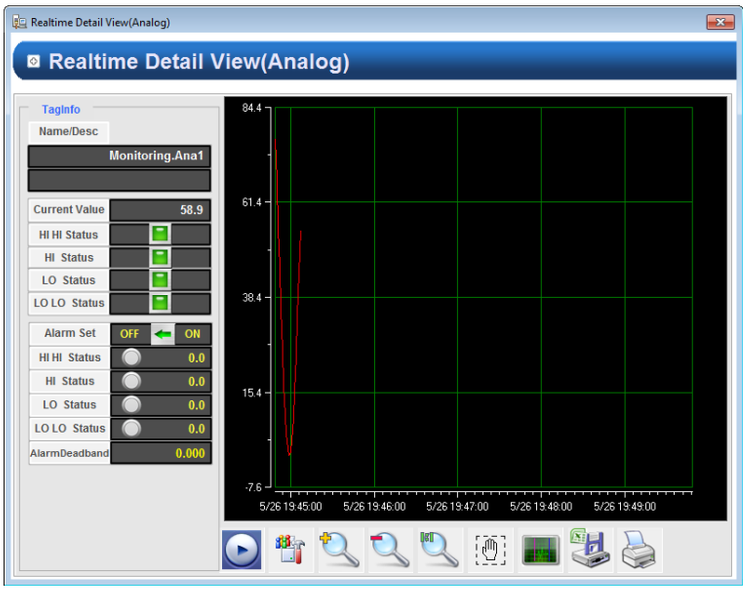
10.3 Popup Menu

In the graphic runtime, the popup menu supports three modes; none mode, user mode, general model.
In a user mode, a user can directly edit the popup menu with the graphic editor and batch-control tags set for the object. A general mode does not allow a user to edit the popup menu but it provides the function to monitor/control tags set for the object in a variety forms.



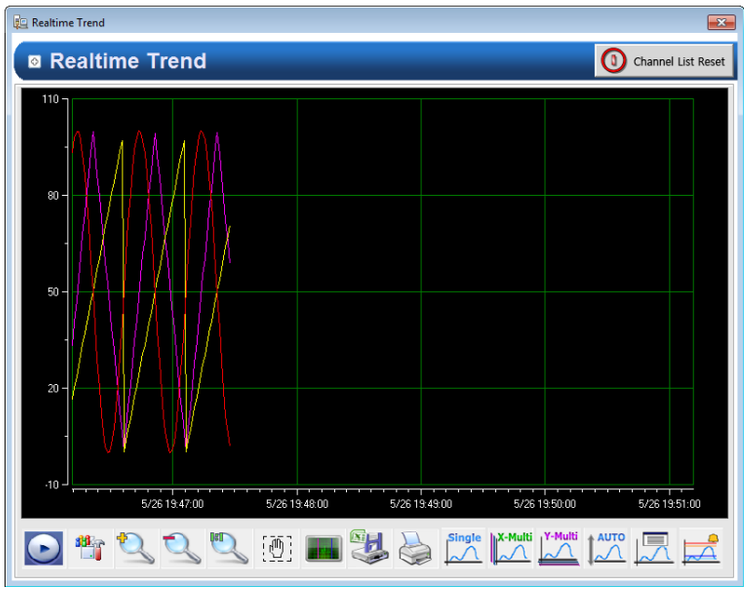
10.3.1 Detailed Realtime Trend

It shows the tags for the object in the form of detailed real time trend. You can edit the screen with the graphic editor.



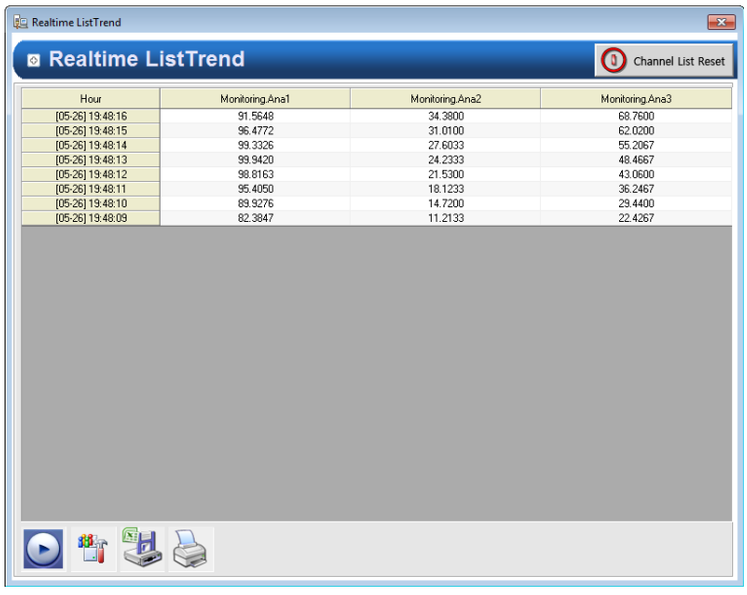
10.3.2 Realtime Trend

It shows the tags for the object in the form of real time trend. You can edit the screen with the graphic editor.



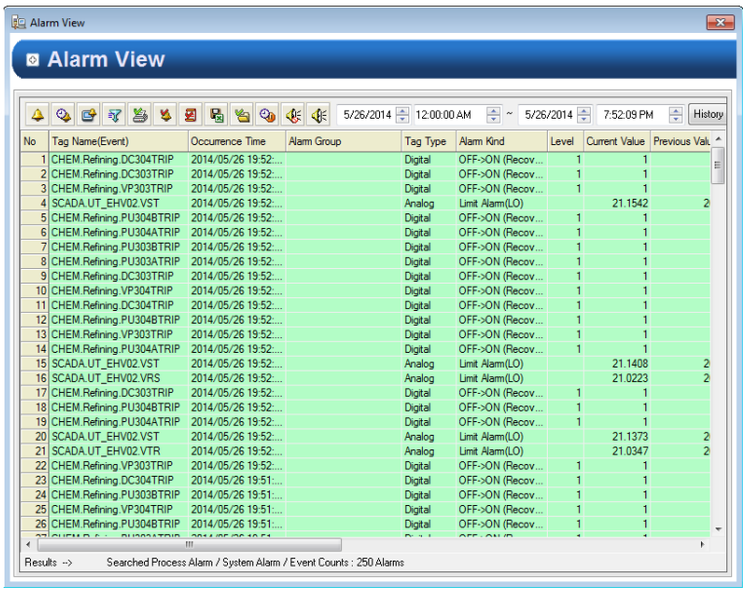
10.3.3 Realtime List Trend

It shows the tags for the object in the form of real time list trend. You can edit the screen with the graphic editor.



10.3.4 Alarm Viewer

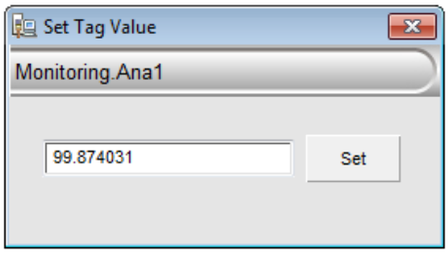
It shows the alarm viewer screen. You can edit the screen with the graphic editor.



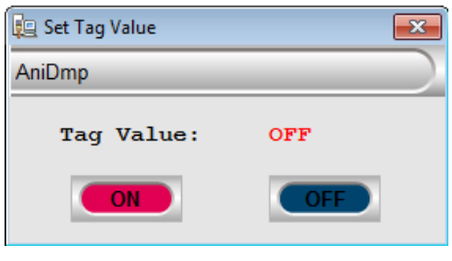
10.3.5 Input Value

It shows the tag control window of tags set for the object. The value input in the analog screen is different from the value input in the digital screen. You can edit the screen with the graphic editor.

* In the analog tag



*In the digital tag



10.3.6 Cross Reference

It shows the cross reference information of the current project. If you perform this command, the dialog box of the cross reference view will show up. For more details, refer to ‘Cross Reference’.

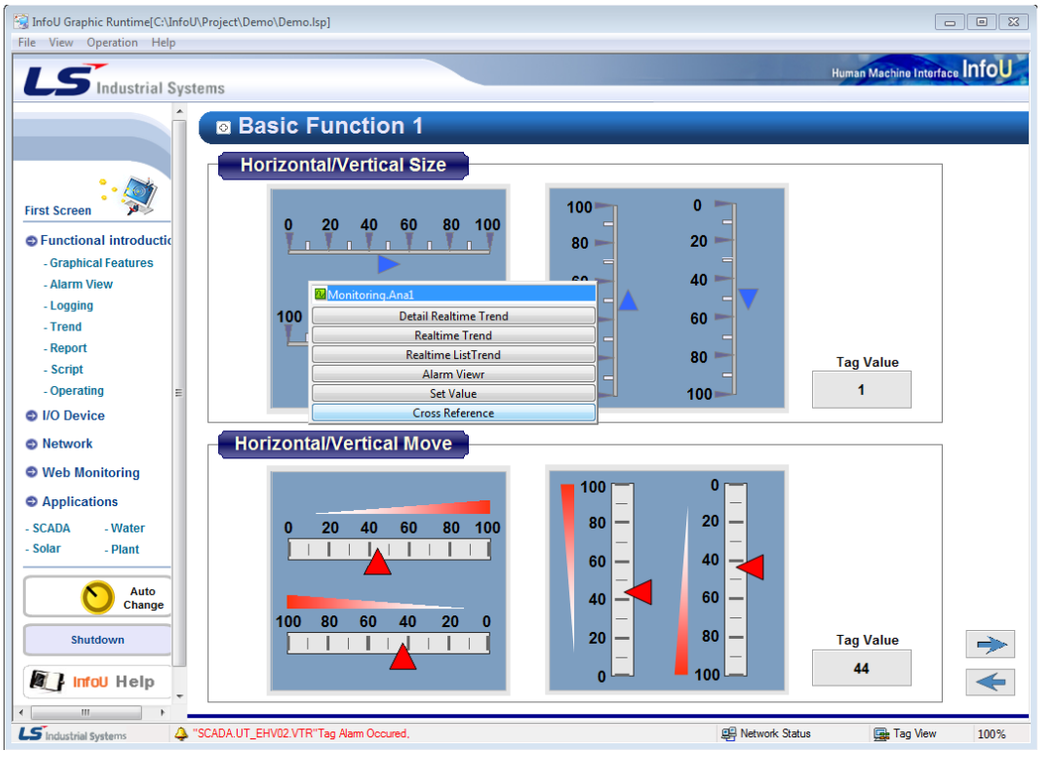
10.4 Cross Reference Function

The Cross Reference is available to use during the runtime.

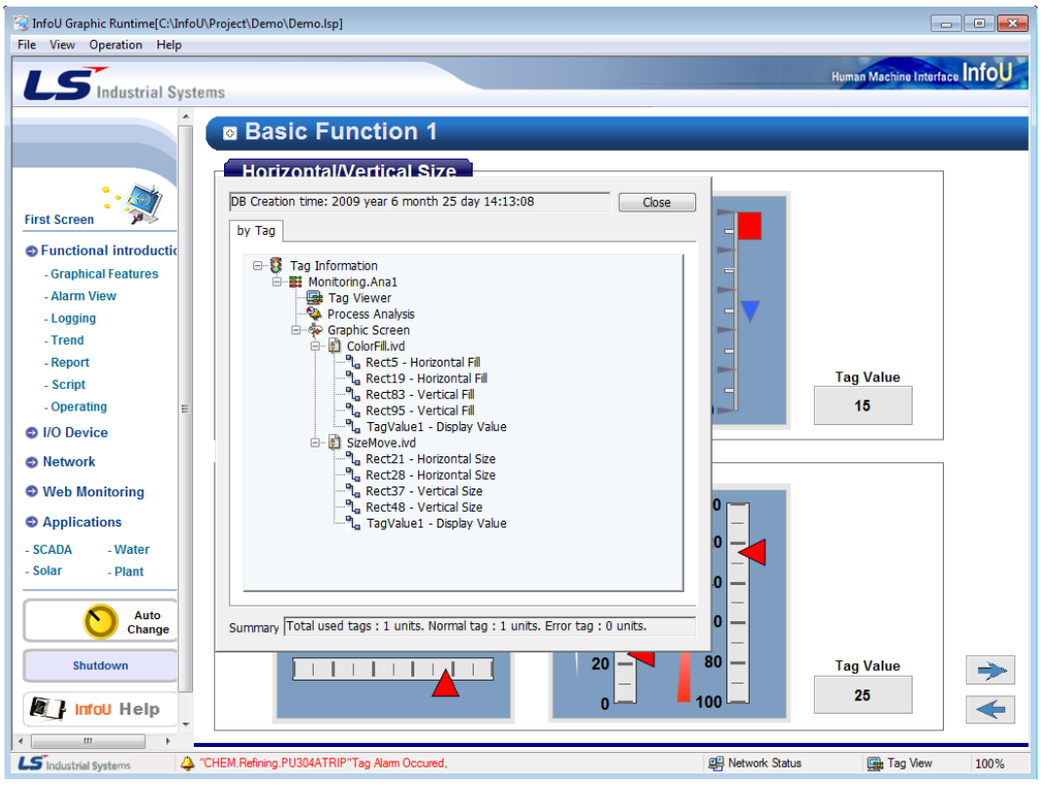
This Cross Reference function is available not only on the plant screen or tag view but also the trend or alarm viewer, and if any alarm occurs during the running, it is possible to move to a relevant screen or trend for analysis by using this Cross Reference function.

10.4.1 Execution on the Plant Screen

If a graphic object with dynamic properties is clicked with the right button of the mouse (MouseUP), the tag list is displayed on the pop-up screen used for the dynamic properties of the graphic object.



If the relevant tag is double clicked with the left button of the mouse on the Cross Reference pop-up screen,

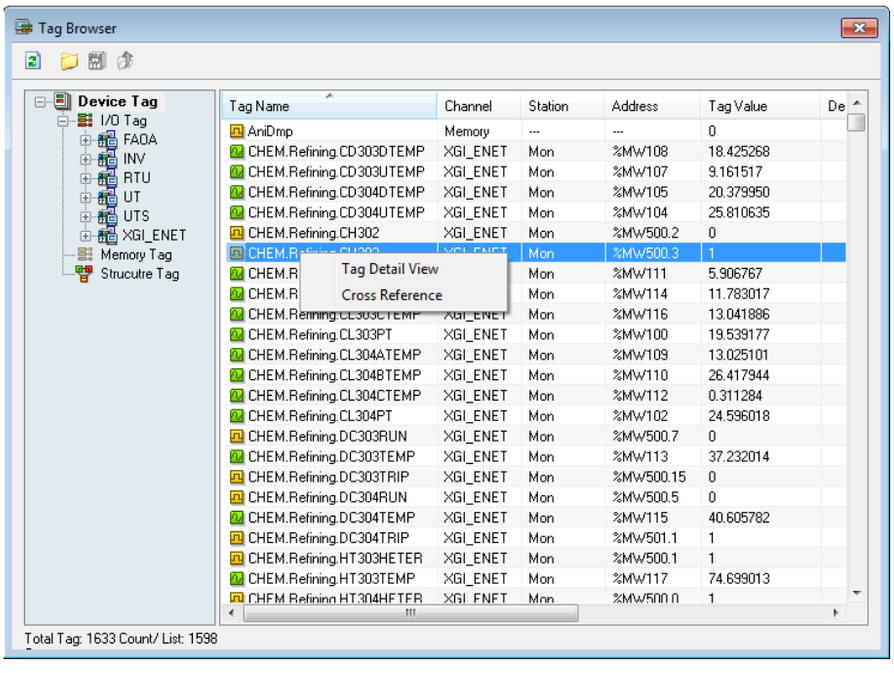


the Cross Reference screen is shown as seen in the figure below and if an item on the screen is double clicked, it moves to a point on the relevant screen and it shows the location of the designated graphic object.

10.4.2 Execution on the Tag View

The Cross Reference of a tag can be inquired on the tag browser.

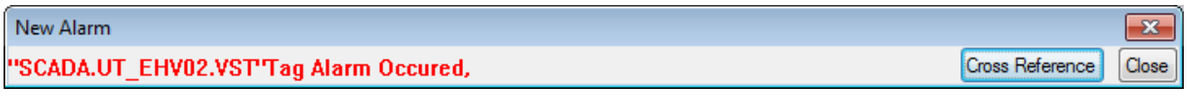
To execute the tag browser, the user may use the menu or tool bar.



If a tag list is clicked with the right button of the mouse on the tag browser, a pop-up menu appears and at this time, if Cross Reference is selected, the Cross Reference screen is displayed.
Also, if the tag detailed view is selected, the detailed screen of the selected tag is displayed as seen in the figure below. At this time, if Cross Reference button is clicked on the tag detailed screen, the Cross Reference screen is inquired. Subsequent actions are the same.

10.4.3 Execution on the Alarm Pop-up Screen

In case that a new alarm occurs, the pop-up window as seen in the figure below appears as the top-level screen and if the 'Cross Reference' button is clicked on the right, the Cross Reference search screen is displayed, and 'Cross Reference' button is clicked with the mouse, the Cross Reference screen can be inquired.

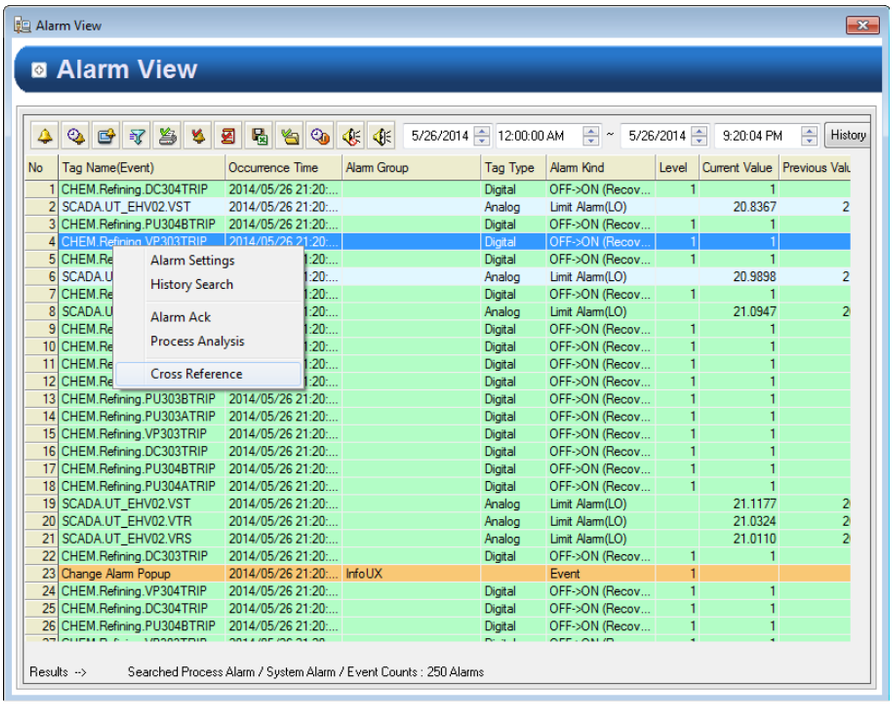


10.4.4 Execution on the Graphic Object

When a new alarm occurs, the alarm popup window will be displayed on the top screen. If you click the 'Cross Reference' button on the right side, the cross reference search screen will be displayed. You can retrieve the cross reference by clicking the 'Cross Reference' button.

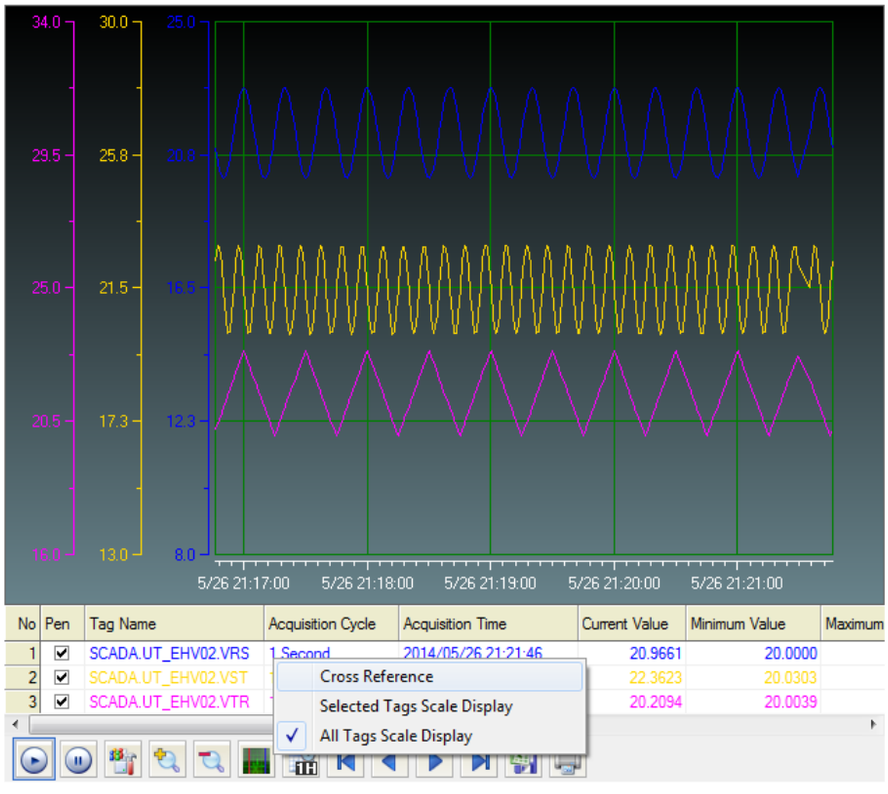
(1) Alarm Viewer

If an alarm item is selected on the alarm viewer graphic object and the right mouse is clicked, the pop-up menu below is displayed and 'Cross Reference' can be selected to display the 'Cross Reference' screen.



(2) Trend

If a data grid item is selected on the trend graphic object and the right mouse is clicked, the pop-up menu below is displayed and ‘Cross Reference’ can be selected to display the ‘Cross Reference’ screen.



Chapter 11 Graphic Object

Graphic Objects can provide various functions that can be implemented just with simple settings such as Alarm Monitoring, Logging Search, Recipe Control and Analog Gauge Display during the real-time running on the graphic screen since they are provided in a form of finished Active X.

In addition, the Graphic Page can be used by inserting external Active X control as a role of ActiveX's container. InfoU Graphic Objects include Alarm Viewer, Trend, List Trend, Recipe Viewer and Gauge Control and it is planned to add new objects continuously.

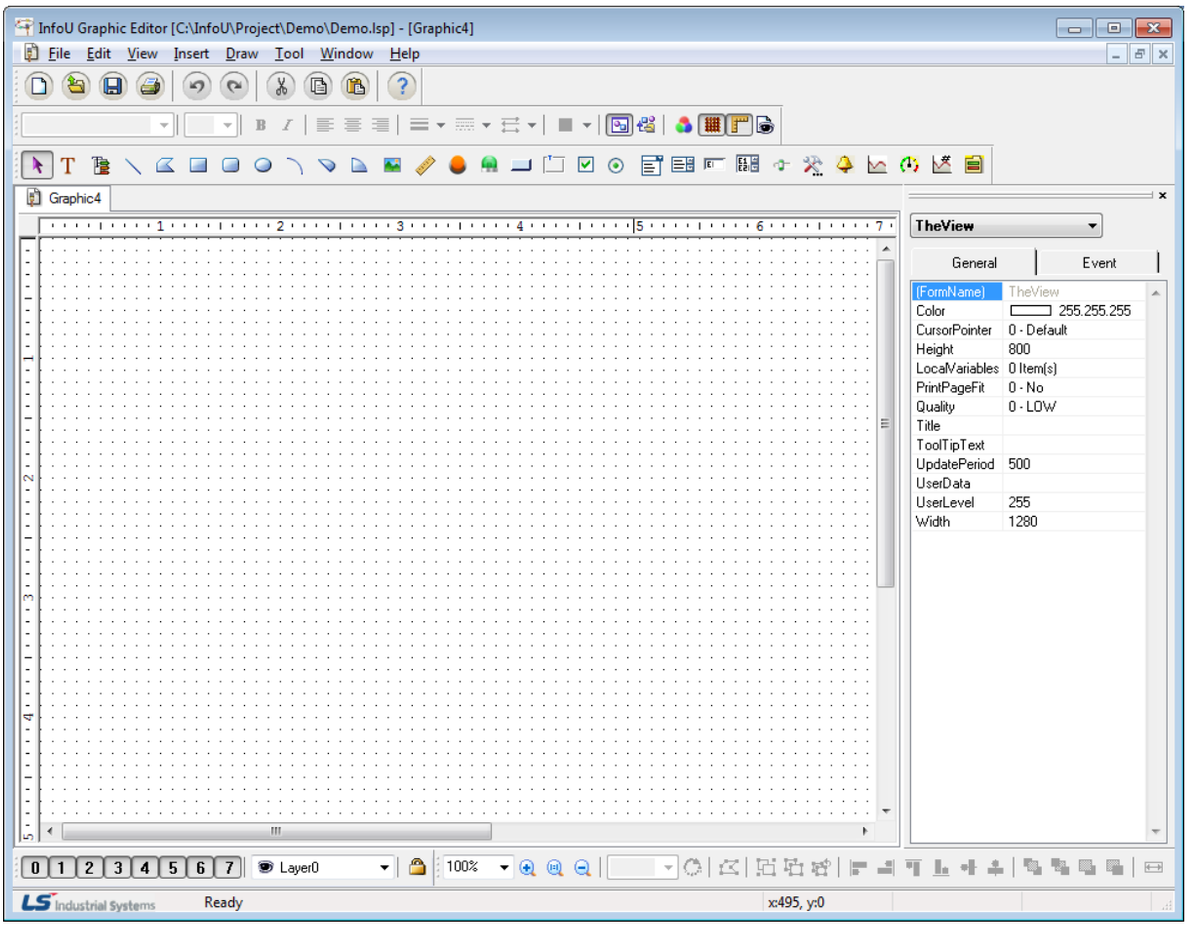
11.1 How to Use Graphic Object

11.1.1 Object Insert

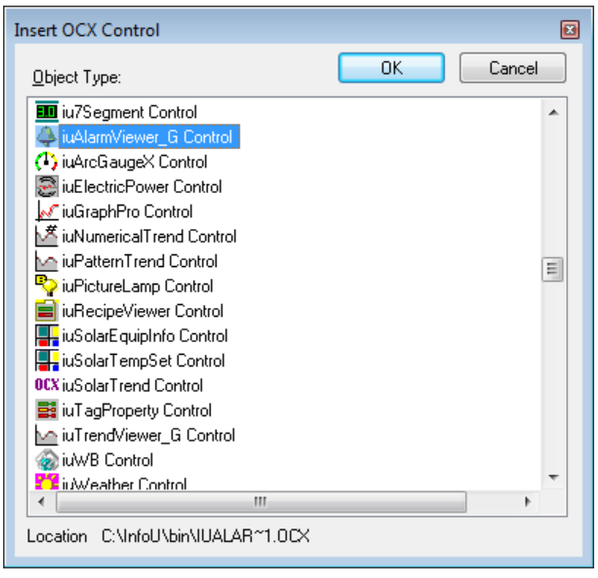
(1) To insert an object immediately, select a relevant shortcut icon on the tool bar.



(2) Select  on the tool menu of the graphic editor.

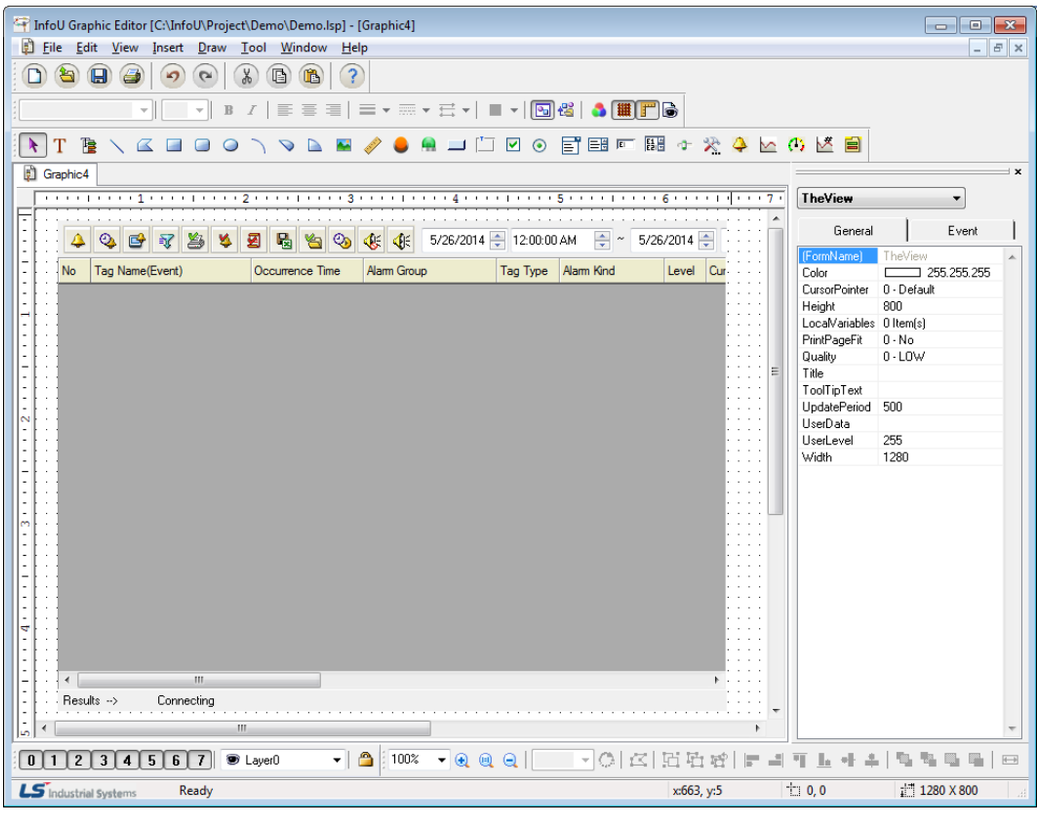


(3) Select an object.



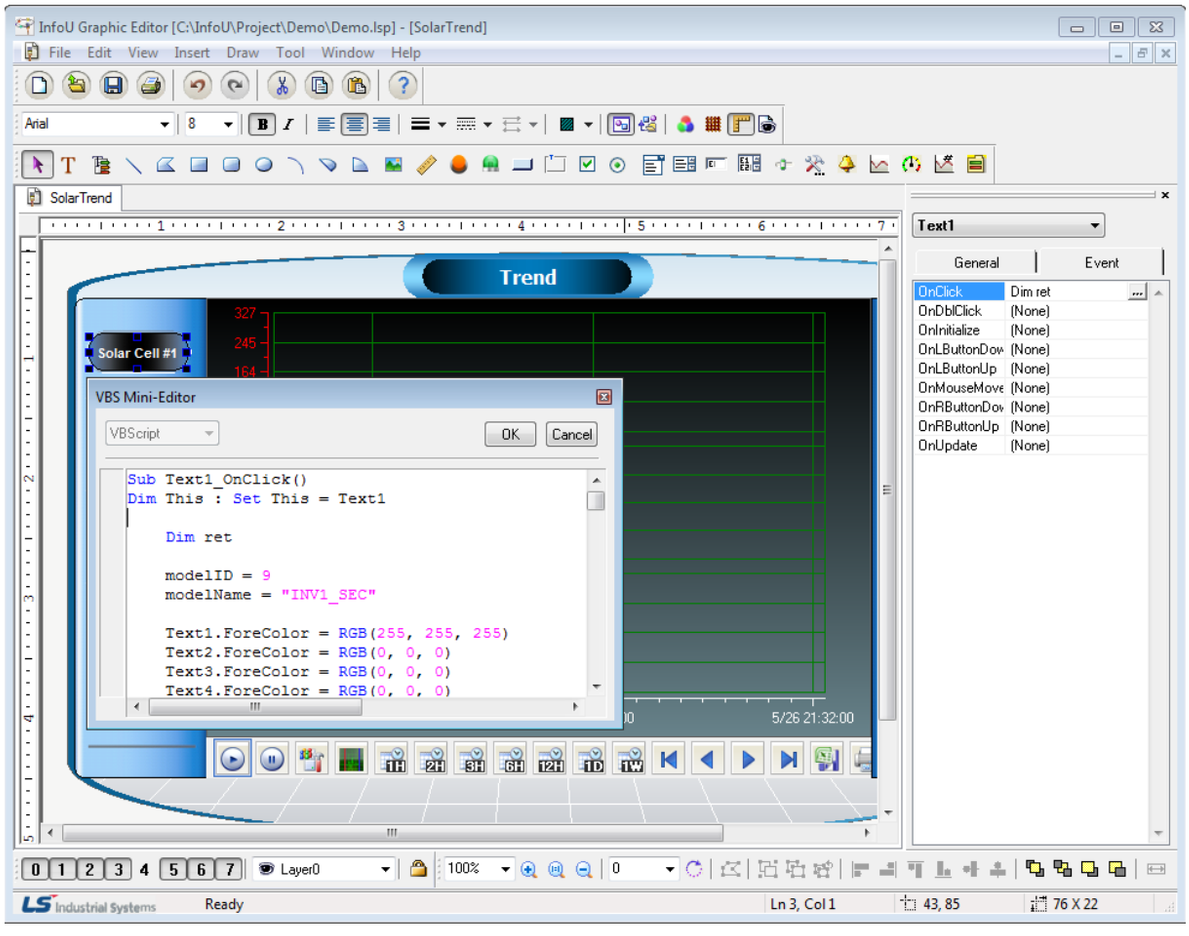
1) iuAlarmViewer: Alarm Viewer
2) iuTrendViewer: Trend Viewer
3) iuNumericalTrend: List Trend Viewer
4) iuRecipeViewer: Recipe Viewer
5) iuArcGaugeX: Gauge Control

(4) The selected object is inserted. The default screen is as follows.



11.1.2 Script Use

It is possible to organize a more diverse and simple screen by controlling the method and properties of the object inserted from the graphic script.



11.2 Alarm Viewer

The Alarm Viewer is an Active X module that searches and displays the occurrence details of alarms and events from the alarm server and performs various functions such as Filtering Search, Backup Alarm File Search and Print.

11.2.1 Prerequisite and Environment

Alarm properties can be set up only after the ActiveX module is inserted to the InfoU graphic editor.

Alarm properties such as update cycle, alarm color, alarm/event item can be selected. Since those alarm properties have been already set up with default values, they act with those default values if the user does not set up additionally.

The property values are saved and then displayed on the alarm/event monitoring screen when the screen opens in the runtime environment.

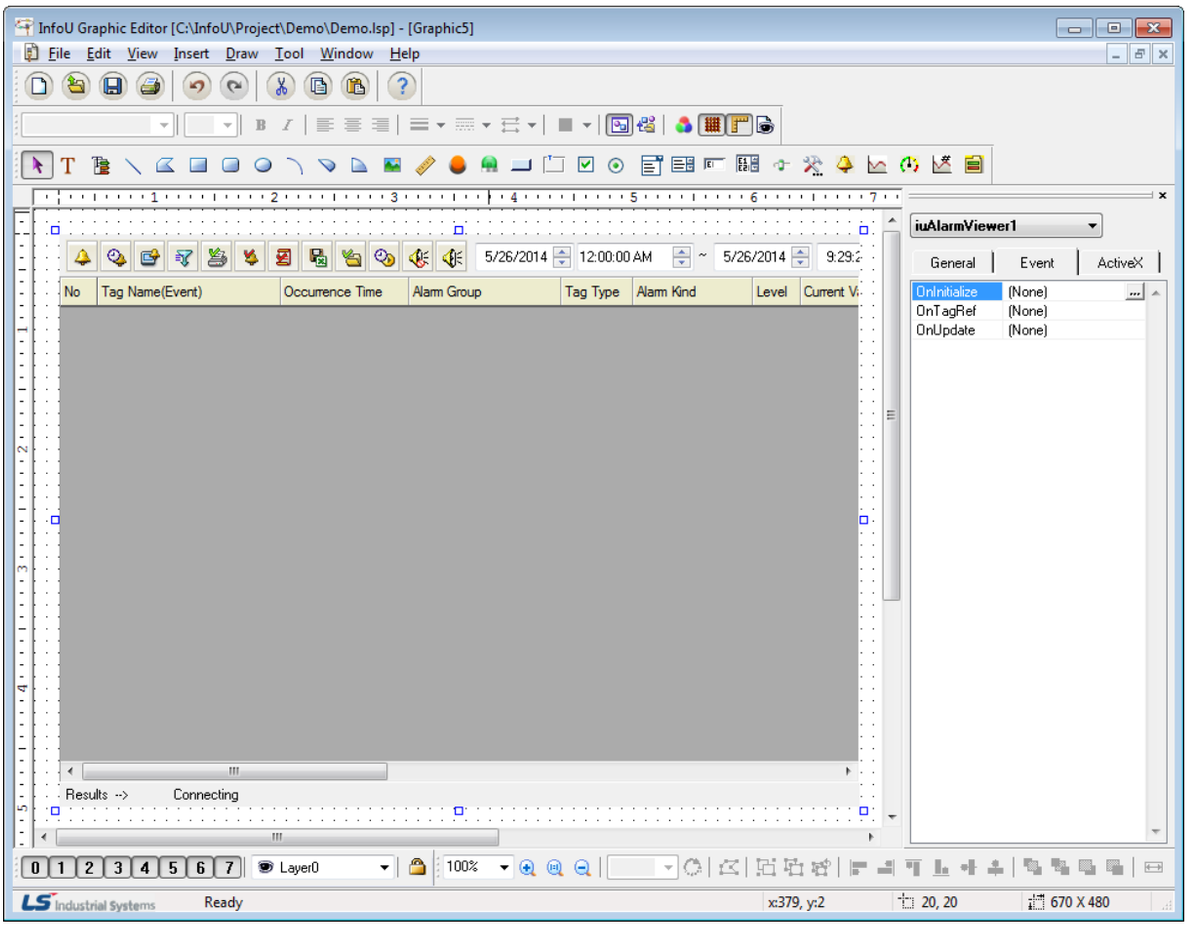
The engineering values are inserted in the following order on the alarm viewer.

If  is selected without selecting  on the InfoU explorer, those values are immediately inserted.

11.2.2 Screen Configuration

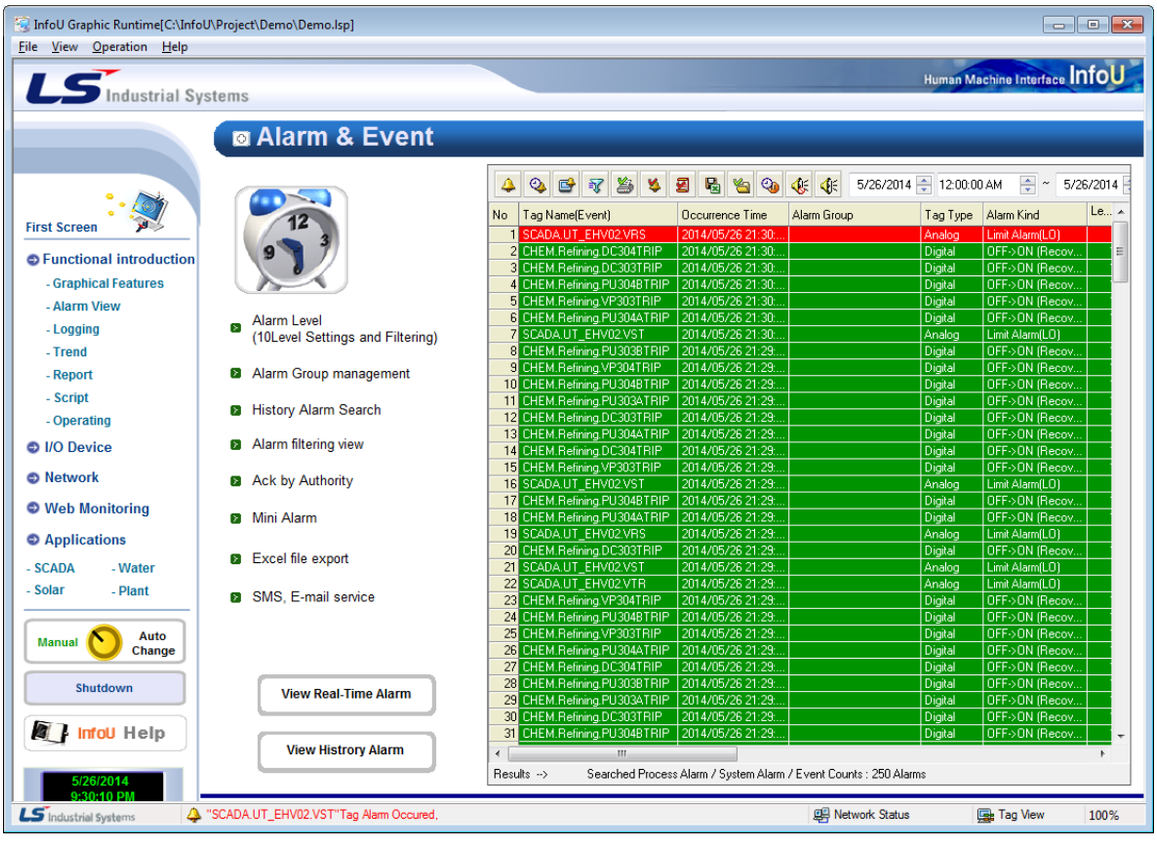
(1) Engineering Screen

The following screen appears after the alarm viewer is inserted.



(2) Runtime Screen

If default trend settings are completed on the engineering screen above and then, the runtime is executed, the screen shows that data values are changed over time.



The splash screen shows real-time alarms and it is for searching alarms, system alarms and events.

1) Tool Bar

Results of tool bar, which consists of necessary functions for operating the viewer, and search are presented in text.

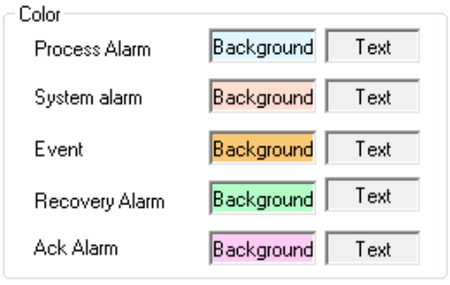
2) Status Display

The current search status is displayed in text.

3) Alarm/Event List

Alarm and event details searched from the Alarm Server are displayed in the Grid.

Various colors are displayed to distinguish progress alarm, system alarm, event, recovery alarm and Ack alarm to let the user easily identify them. Colors are as follows and text and background colors can be changed.



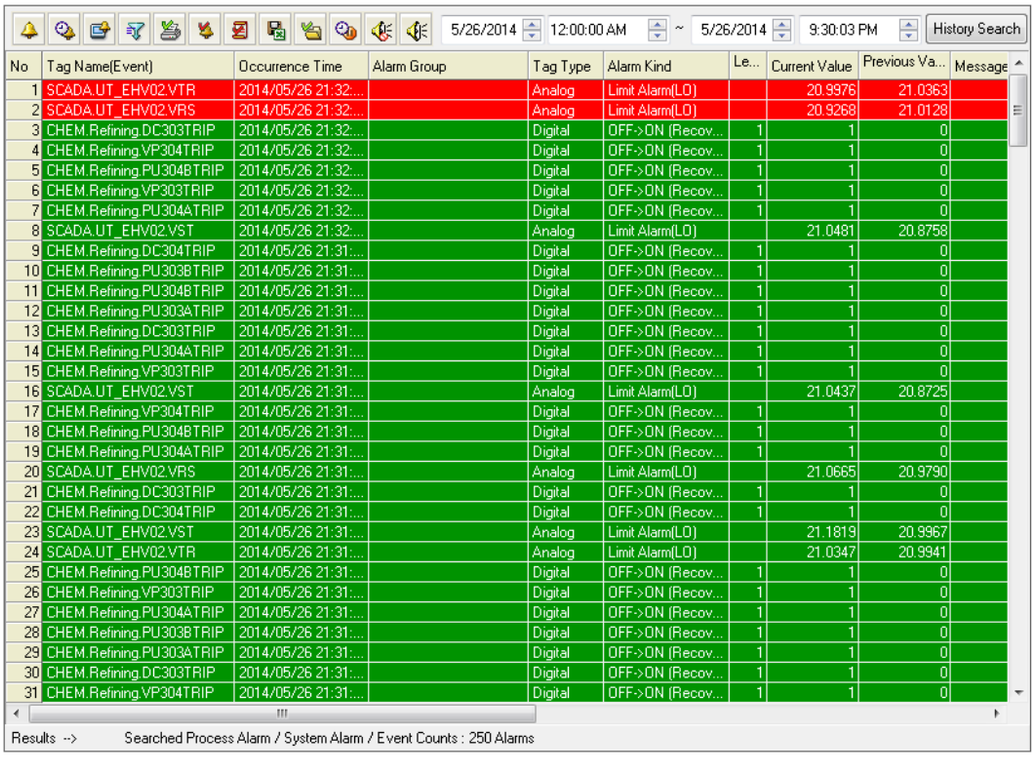
11.2.3 Real-time Function

(1) Real-time Alarm Search ( )

All of the real-time alarms and events that currently take place are searched.

The number that marks real-time alarms is the value decided from alarm server settings and the number of process alarms and system alarms is twice of the number of tags and 500 is the default value of events.

Real-time alarms become invisible in the list through Alarm Recovery and Alarm Ack. That is, the alarm list is updated every second.



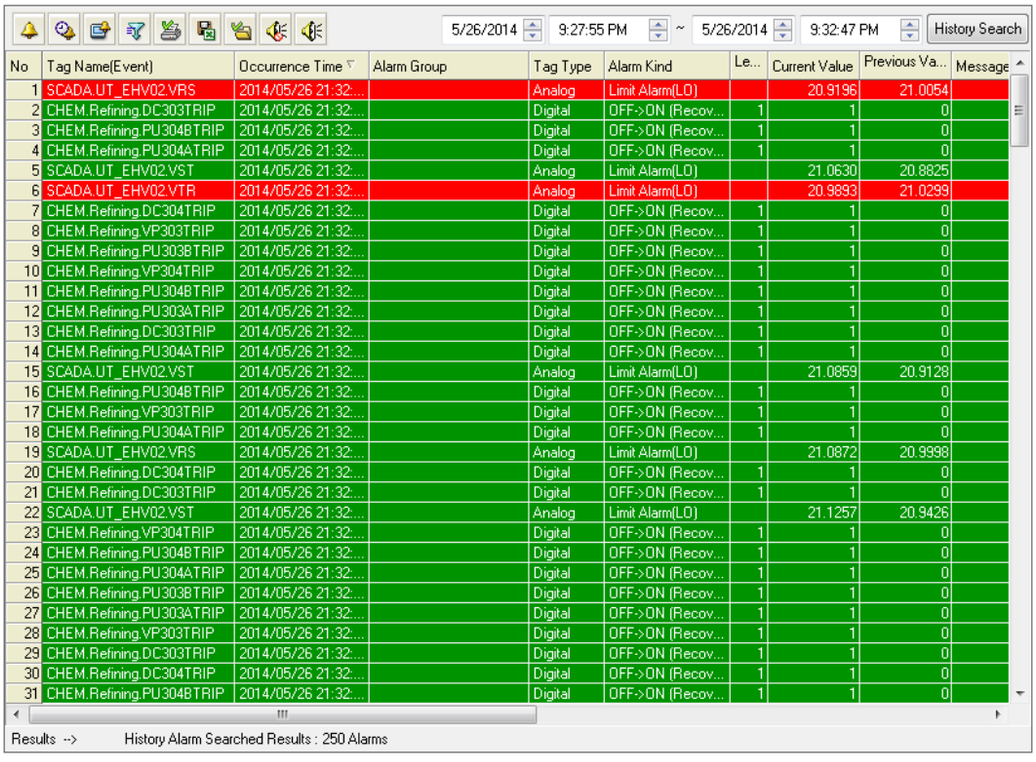
No	Tag Name(Event)	Occurrence Time	Alarm Group	Tag Type	Alarm Kind	Le...	Current Value	Previous Va...	Message
1	SCADA.UT_EHV02.VTR	2014/05/26 21:32:...		Analog	Limit Alarm(LO)		20.9976	21.0363	
2	SCADA.UT_EHV02.VRS	2014/05/26 21:32:...		Analog	Limit Alarm(LO)		20.9268	21.0128	
3	CHEM.Refining.DC303TRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
4	CHEM.Refining.VP304TRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
5	CHEM.Refining.PU304BTRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
6	CHEM.Refining.VP303TRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
7	CHEM.Refining.PU304ATRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
8	SCADA.UT_EHV02.VST	2014/05/26 21:32:...		Analog	Limit Alarm(LO)		21.0481	20.8758	
9	CHEM.Refining.DC304TRIP	2014/05/26 21:31:...		Digital	OFF->ON (Recov...	1	1	0	
10	CHEM.Refining.PU303BTRIP	2014/05/26 21:31:...		Digital	OFF->ON (Recov...	1	1	0	
11	CHEM.Refining.PU304BTRIP	2014/05/26 21:31:...		Digital	OFF->ON (Recov...	1	1	0	
12	CHEM.Refining.PU303ATRIP	2014/05/26 21:31:...		Digital	OFF->ON (Recov...	1	1	0	
13	CHEM.Refining.DC303TRIP	2014/05/26 21:31:...		Digital	OFF->ON (Recov...	1	1	0	
14	CHEM.Refining.PU304ATRIP	2014/05/26 21:31:...		Digital	OFF->ON (Recov...	1	1	0	
15	CHEM.Refining.VP303TRIP	2014/05/26 21:31:...		Digital	OFF->ON (Recov...	1	1	0	
16	SCADA.UT_EHV02.VST	2014/05/26 21:31:...		Analog	Limit Alarm(LO)		21.0437	20.8725	
17	CHEM.Refining.VP304TRIP	2014/05/26 21:31:...		Digital	OFF->ON (Recov...	1	1	0	
18	CHEM.Refining.PU304BTRIP	2014/05/26 21:31:...		Digital	OFF->ON (Recov...	1	1	0	
19	CHEM.Refining.PU304ATRIP	2014/05/26 21:31:...		Digital	OFF->ON (Recov...	1	1	0	
20	SCADA.UT_EHV02.VRS	2014/05/26 21:31:...		Analog	Limit Alarm(LO)		21.0665	20.9790	
21	CHEM.Refining.DC303TRIP	2014/05/26 21:31:...		Digital	OFF->ON (Recov...	1	1	0	
22	CHEM.Refining.DC304TRIP	2014/05/26 21:31:...		Digital	OFF->ON (Recov...	1	1	0	
23	SCADA.UT_EHV02.VST	2014/05/26 21:31:...		Analog	Limit Alarm(LO)		21.1819	20.9967	
24	SCADA.UT_EHV02.VTR	2014/05/26 21:31:...		Analog	Limit Alarm(LO)		21.0347	20.9941	
25	CHEM.Refining.PU304BTRIP	2014/05/26 21:31:...		Digital	OFF->ON (Recov...	1	1	0	
26	CHEM.Refining.VP303TRIP	2014/05/26 21:31:...		Digital	OFF->ON (Recov...	1	1	0	
27	CHEM.Refining.PU304ATRIP	2014/05/26 21:31:...		Digital	OFF->ON (Recov...	1	1	0	
28	CHEM.Refining.PU303BTRIP	2014/05/26 21:31:...		Digital	OFF->ON (Recov...	1	1	0	
29	CHEM.Refining.PU303ATRIP	2014/05/26 21:31:...		Digital	OFF->ON (Recov...	1	1	0	
30	CHEM.Refining.DC303TRIP	2014/05/26 21:31:...		Digital	OFF->ON (Recov...	1	1	0	
31	CHEM.Refining.VP304TRIP	2014/05/26 21:31:...		Digital	OFF->ON (Recov...	1	1	0	

(2) History Alarm Search ( )

All of the real-time alarms and events that have taken place are searched

All of the alarms that have taken place are displayed in the chronicle order such as occurred alarms, recovered alarms, acknowledged alarms and events including invisible alarms in the real-time alarm list.

The storing reference for history alarms is decided from alarm server settings and the default values are set up to save up to 10,000 process alarms for 3 years, up to 5000 system alarms for 3 years and up to 5000 events for 3 years.



No	Tag Name(Event)	Occurrence Time	Alarm Group	Tag Type	Alarm Kind	Le...	Current Value	Previous Va...	Message
1	SCADA.UT_EHV02.VRS	2014/05/26 21:32:...		Analog	Limit Alarm(LO)		20.9196	21.0054	
2	CHEM.Refining.DC303TRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
3	CHEM.Refining.PU304BTRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
4	CHEM.Refining.PU304ATRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
5	SCADA.UT_EHV02.VST	2014/05/26 21:32:...		Analog	Limit Alarm(LO)		21.0630	20.8825	
6	SCADA.UT_EHV02.VTR	2014/05/26 21:32:...		Analog	Limit Alarm(LO)		20.9893	21.0299	
7	CHEM.Refining.DC304TRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
8	CHEM.Refining.VP303TRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
9	CHEM.Refining.PU303BTRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
10	CHEM.Refining.VP304TRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
11	CHEM.Refining.PU304BTRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
12	CHEM.Refining.PU303ATRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
13	CHEM.Refining.DC303TRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
14	CHEM.Refining.PU304ATRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
15	SCADA.UT_EHV02.VST	2014/05/26 21:32:...		Analog	Limit Alarm(LO)		21.0859	20.9128	
16	CHEM.Refining.PU304BTRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
17	CHEM.Refining.VP303TRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
18	CHEM.Refining.PU304ATRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
19	SCADA.UT_EHV02.VRS	2014/05/26 21:32:...		Analog	Limit Alarm(LO)		21.0872	20.9998	
20	CHEM.Refining.DC304TRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
21	CHEM.Refining.DC303TRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
22	SCADA.UT_EHV02.VST	2014/05/26 21:32:...		Analog	Limit Alarm(LO)		21.1257	20.9426	
23	CHEM.Refining.VP304TRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
24	CHEM.Refining.PU304BTRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
25	CHEM.Refining.PU304ATRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
26	CHEM.Refining.PU303BTRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
27	CHEM.Refining.PU303ATRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
28	CHEM.Refining.VP303TRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
29	CHEM.Refining.DC303TRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
30	CHEM.Refining.DC304TRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	
31	CHEM.Refining.PU304BTRIP	2014/05/26 21:32:...		Digital	OFF->ON (Recov...	1	1	0	

(3) Viewer Settings ( )

Setting values of the current alarm viewer can be changed.

The setting window will be further explained later.

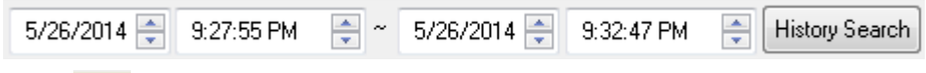
(4) History Inquiry Setting Screen ( )

It is used when the user wants to search history alarms after filtering by conditions.

The history searching setting window will be further explained later.

(5) Alarm retrieval using the history search bar

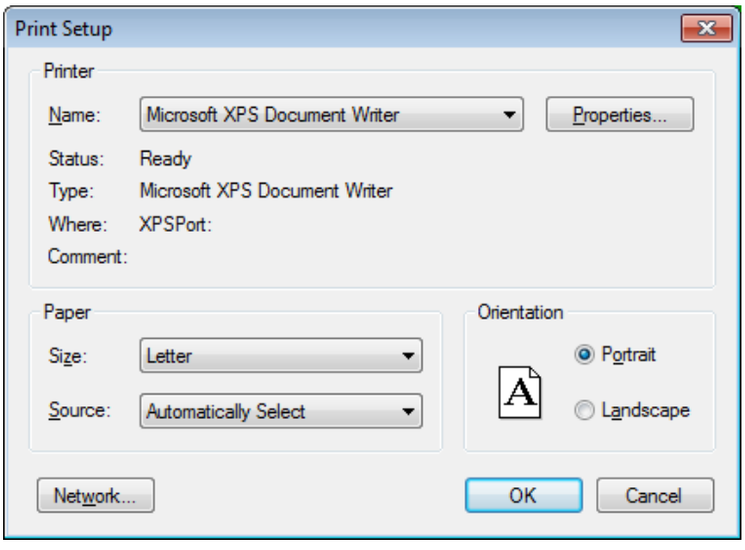
Through this function, a user can conveniently retrieve history alarms by filtering the start and end time with the search bar.



(6) Print ( )

The screen is printed.

Select a printer on the Print window and click on Properties button to set up whether to print vertically or horizontally. Information is outputted from the selected printer.



(7) Individual Alarm Ack ( )

The currently selected alarm is acknowledged. For individual alarm ack, double click after selecting an alarm to acknowledge it.

For event and history alarms, Ack is not available.

(8) Page Ack ( )

All of the alarms in the current page are acknowledged.

For event and history alarms, Ack is not available.

(9) Save as Excel File ( )

Occurred alarms are saved in Excel files. Press Save button and insert a file name for each alarm to be saved.

(10) Backup File Import ( )

The alarm backup files saved on the server or other storage devices can be imported and viewed in a list.

Detailed functions on Load Backup File are described later.

(11) Sound Stop ( )

Basically, sound is generated upon the occurrence of alarm, and this function is to stop the sound. If another new alarm occurs, sound is regenerated. To stop sound for the entire alarms, alarm settings shall be modified on the engineering program.

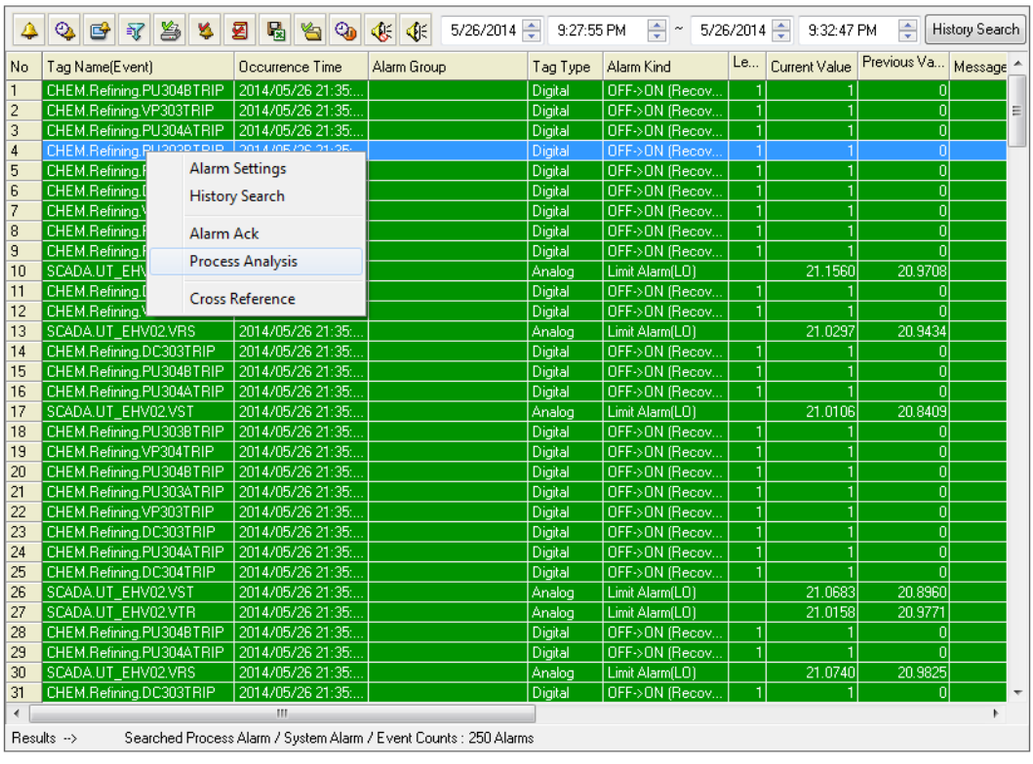
(12) Start ( ) and Stop ( )

Buttons to start and stop updating the alarm list in real-time.

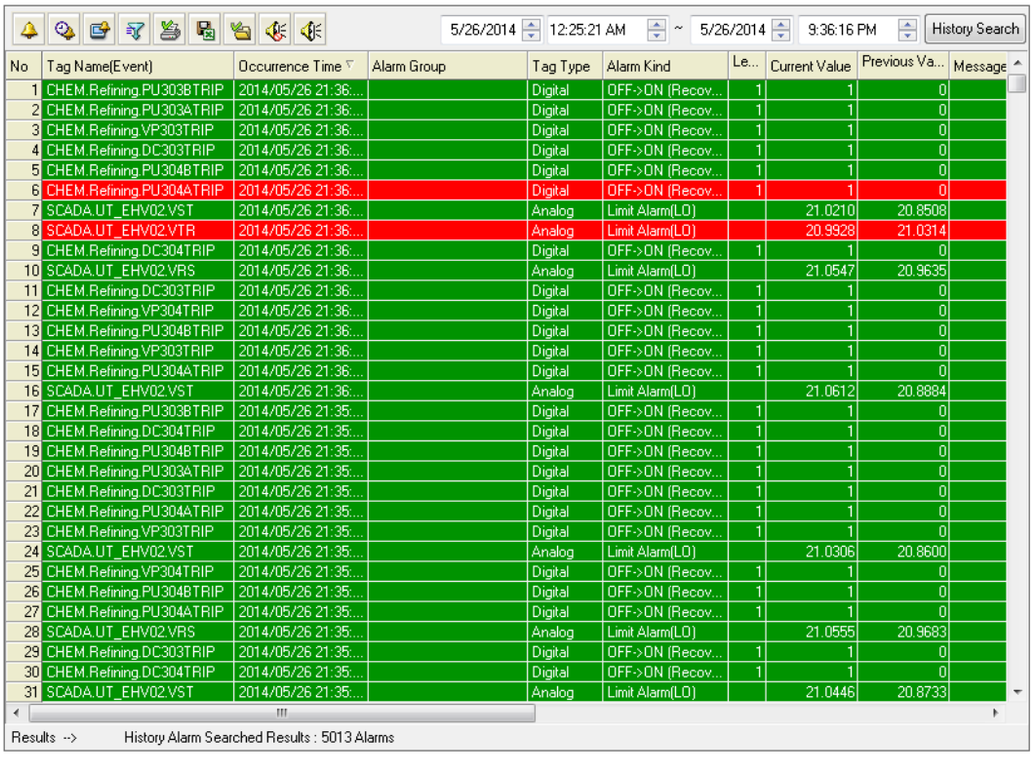
(13) Process Analysis

This function is to move the alarm that takes place on the real-time alarm list to the history alarm list for analyzing. Select an alarm from the real-time alarm list and press the right side of the mouse to display the following menu. If Process Analysis is selected here, it moves the relevant alarm, which will be highlighted in

red, to the history alarm list for analyzing.

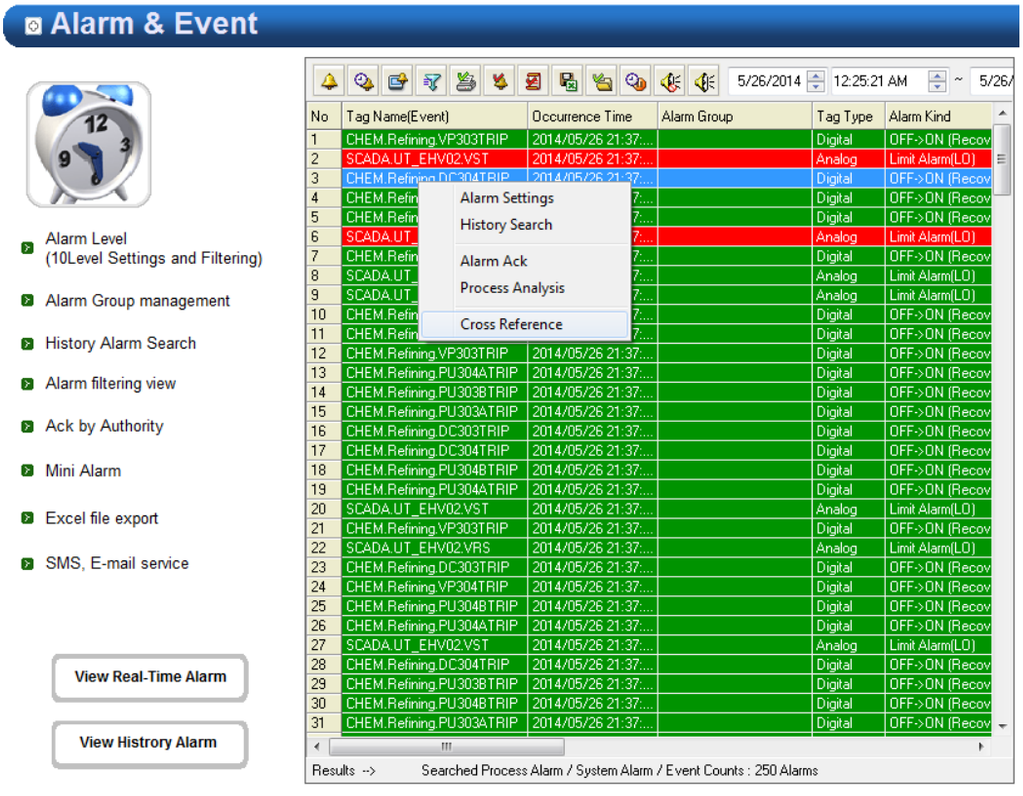


The following screen shows the result of moving to the history alarm list.

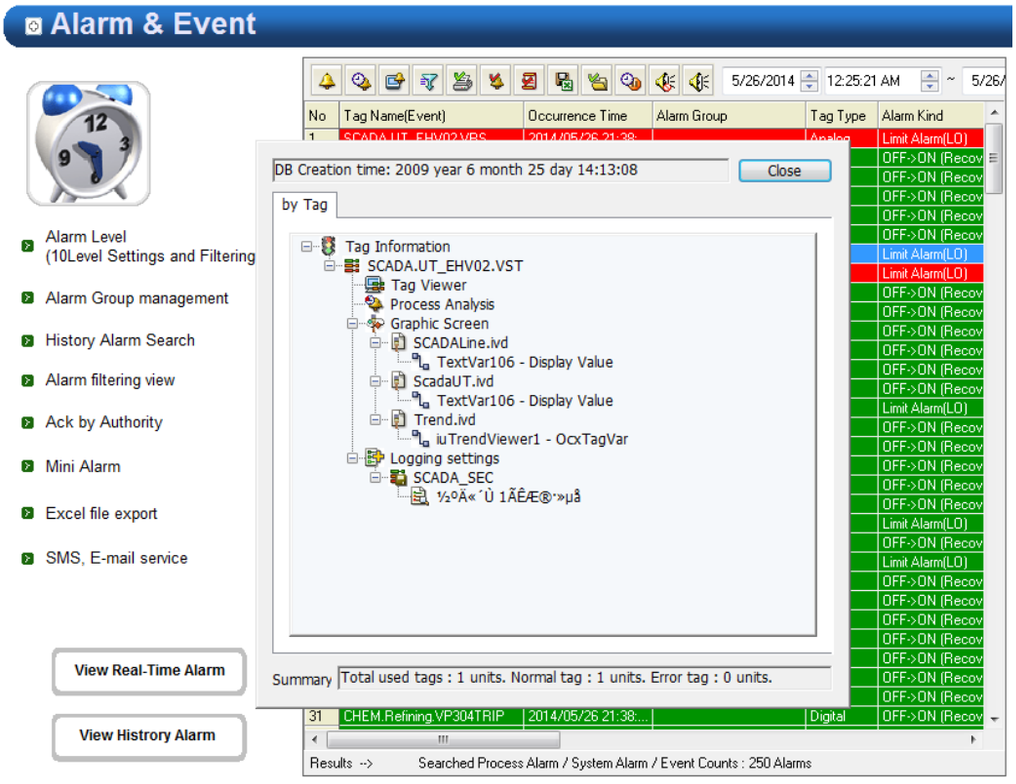


(14) Cross Reference

Cross Reference makes the screen move to the graphic screen to which the selected alarm belongs.



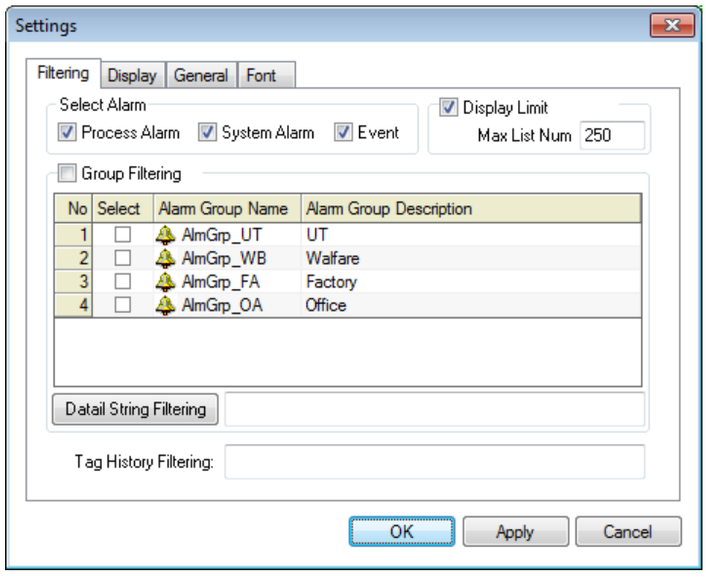
If an alarm is selected in the real-time alarm list and the right side of the mouse is clicked to select Cross Reference, the following Cross Reference screen is displayed. It is possible to move to the place the user wants to go.



11.2.4 Settings

If the right side of the mouse is clicked on Alarm Viewer ocx and the 'Properties' menu is selected, the setting window appears. The same setting screen is used for the engineering screen and the run-time screen. In addition, once property values are saved on the engineering screen, those values will be continuously valid however, if those property values are changed on the runtime screen, the changed values are applied only to the runtime screen. The changed values are cancelled if the runtime stops and the property values set up on the engineering screen are applied.

(1) Filtering Setting Items



Application range of Alarm viewer can be configured by filtering setting items such as Select Alarm and Group Filtering.

1) Select Alarm

You can set the alarm type to be displayed in the alarm viewer.

2) Display Limit

You can set the maximum number of list alarms to be displayed in the alarm viewer.

3) Group Filtering

You can determine whether applying the group filtering and which group you will view. When selecting the group filtering, the only relevant group will be applied to the alarm viewer and displayed. If you do not select the group filtering, it will be applied to the whole alarm tags and displayed.

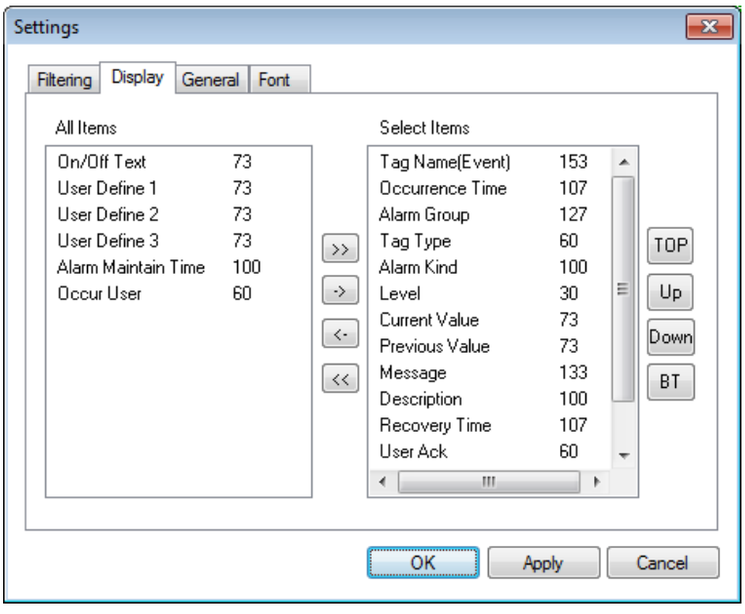
4) Detail String Flitering

In the alarm viewer, it is the function to filter detailed characters for the alarm group name. ";" is the separator in word unit and "+" is the separator in string unit. For example, if you type "UT;FA+Grp" on the above screen and press the detailed character filtering button, AlmGrp_UT and the group name of AlmGrp_FA will be automatically set in the alarm group name.

5) Tag History Filtering

Tag history filtering is applied to tag names generated in the history retrieval setting window of the graphic runtime and the only corresponding alarm is displayed. ";" is the separator in word unit and "+" is the separator in string unit.

(2) Alarm Display Item



It is the function to choose alarm items that you want to see in the alarm viewer. There are 18 alarm items; tag name(event), occurrence time, alarm group(occurrence area), tag type, alarm(event) type, grade, current value(status), previous value(status), message, description, recovery time, recognizing user, recognizing time, unit, user definition 1, user definition 2, user definition 3, user definition 4. The number next to the item is the column size of the item in the alarm viewer and it can be changeable. Double-click the item and just input the desired size.

1) >> Button

   Move the marked item from the entire list to the selected list.

2) << Button

   Move the unmarked item from the entire list to the selected list.

3) -> Button

   Move from the entire list to the selected list.

4) <- Button

   Move from the selected list to the entire list.

5) TOP Button

   Move the marked location on the selected column from the selected list to the top left.

6) ▲ Button

   Move the marked location on the selected column from the selected list to the next cell on the left side.

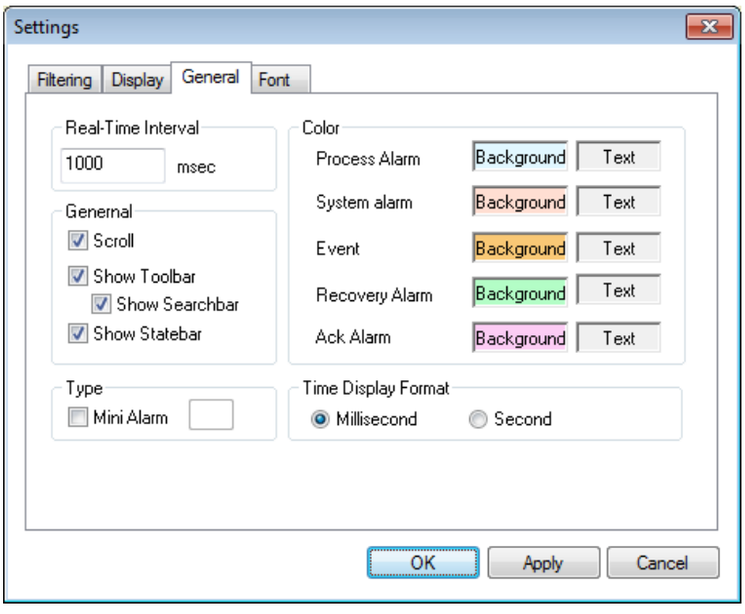
7) ▼ Button

   Move the marked location on the selected column from the selected list to the next cell on the right side.

8) BT Button

   Move the marked location on the selected column from the selected list to the end on the right side.

(3) General



1) Real-Time Interval

Enter a time to update the viewer. Alarms are searched on the relevant severs at the entered time and a list is displayed.

2) Scroll

Select whether to scroll or not on the viewer.

3) Show Toolbar

Set 'Show/Hide toolbar' of the viewer.

4) Show Searchbar

Set 'Show/Hide history search bar' of the viewer.

5) Show Statebar

Set 'Show/Hide status bar' of the viewer.

6) Mini Alarm

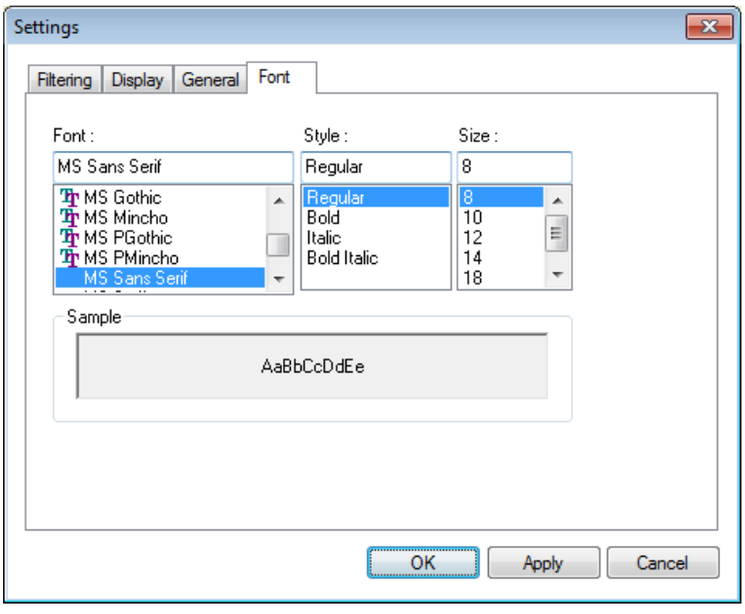
Select whether to use mini alarms.

If mini alarms are used, tool bars or scroll do not exist.

7) Color

Select each background color and text color of Process Alarm, System Alarm, Event, Recovery Alarm and Ack Alarm.

(4) Font



Decide Font, Style and Size to be displayed on the alarm viewer list.

11.2.5 Filtering

As a function used when the user wants filter history alarms with several conditions and searches only the history alarms that meet those conditions, it proceeds filtering as follows.

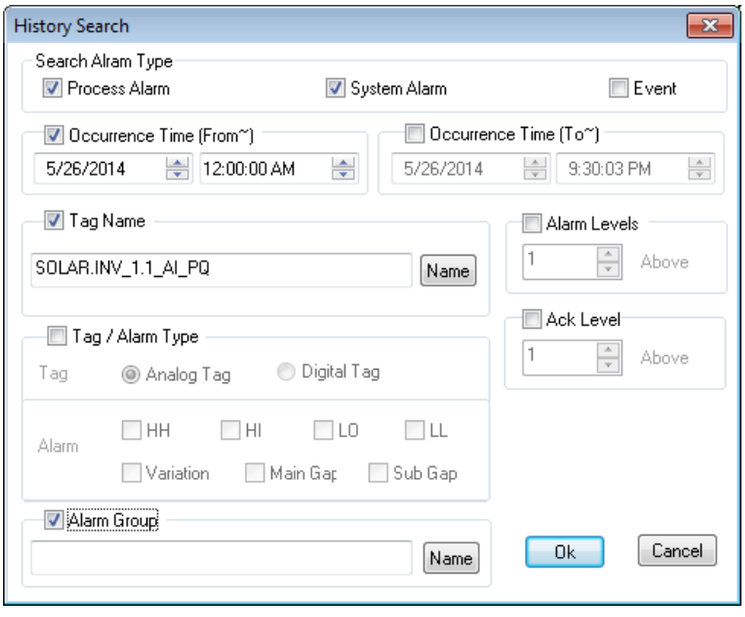
(1) Select items to filter and search with the input condition 'And' condition.

(2) The alarms that meet the search conditions are displayed on the alarm list.

(3) The timer stops automatically.

(4) To return to the real-time search, click on Real-Time Alarm Search button.

The Filtering Search Screen is as following



1) Search Alarm Type/Event

Select one among Process Alarm, System Alarm and Event to search.

2) Occurrence Time (From)

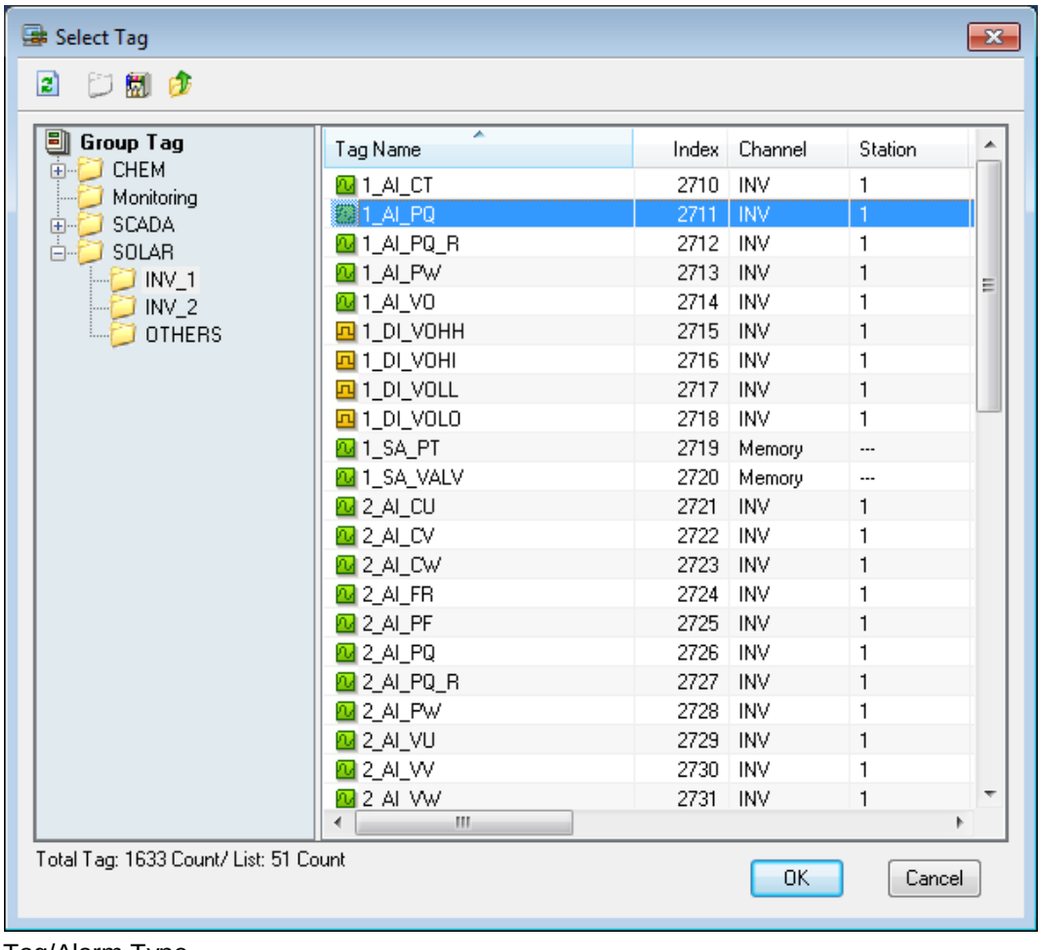
Only the alarms that have occurred since the occurrence time can be searched.

3) Occurrence Time (To)

Only the alarms that had occurred before the occurrence time can be searched.

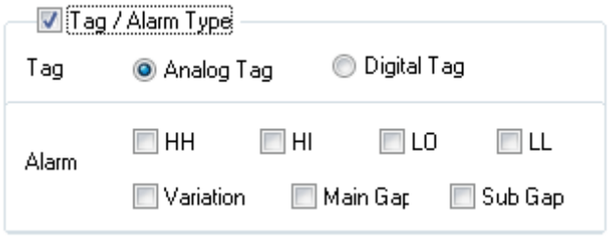
4) Tag Name

Designate the tag name to search. Click on Name button to display the tag browser screen to select a tag from it.



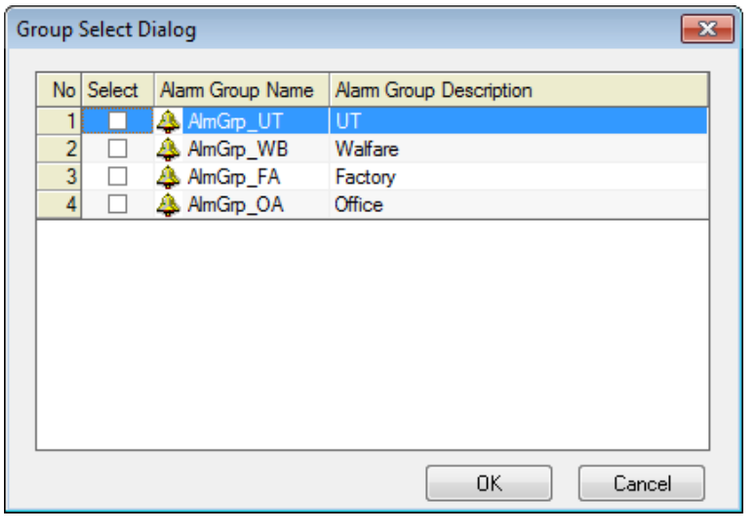
5) Tag/Alarm Type

Select either Analog Tag or Digital Tag and then, select an alarm kind that fits to it to search. If Digital Tag is selected, the following screen appears.



6) Select an alarm group

Select an alarm group to search.



7) Alarm Level

Only the alarms higher than the entered alarm level are searched. Levels 1 through 99 are available.

8) Ack Level

Only the alarms higher than the entered tag priority are searched. Levels 1 through 99 are available.

11.2.6 Other Functions

(1) Alarm Ack

As a function to acknowledge (Ack) the occurred alarm, it is available to acknowledge not only by alarm but also by page unit.

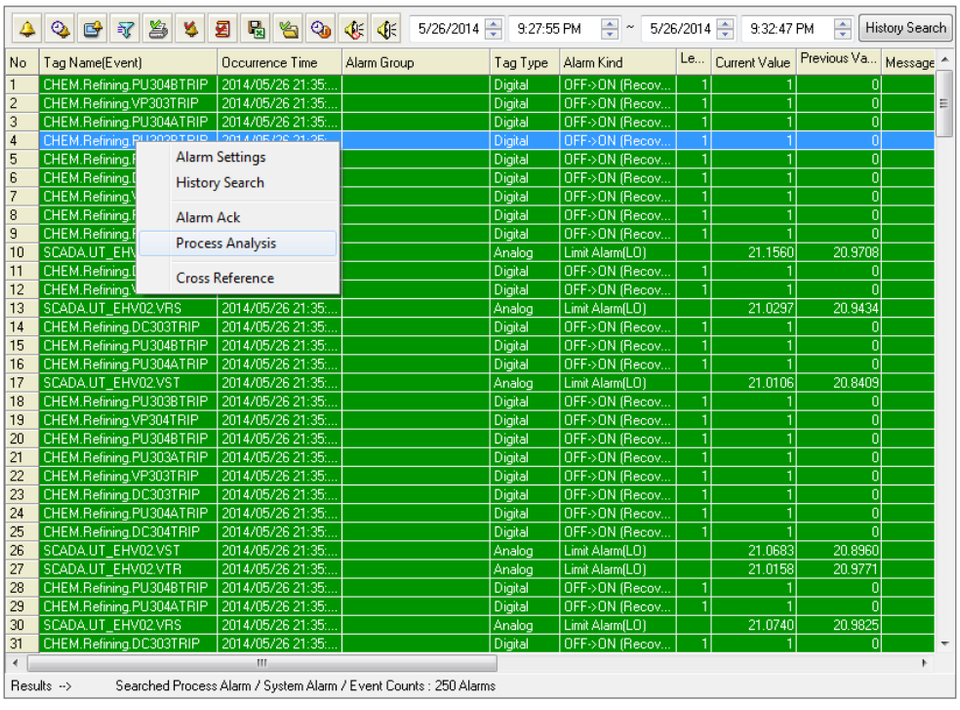
For individual alarm Ack, double click after selecting an alarm or press [Ack] button on the tool bar. It is also available to press the right button of the mouse. For Event and History Alarm, Ack is not available.

For Page Ack, press [Page Ack] button on the tool bar to acknowledge all of the alarms seen in the current page.

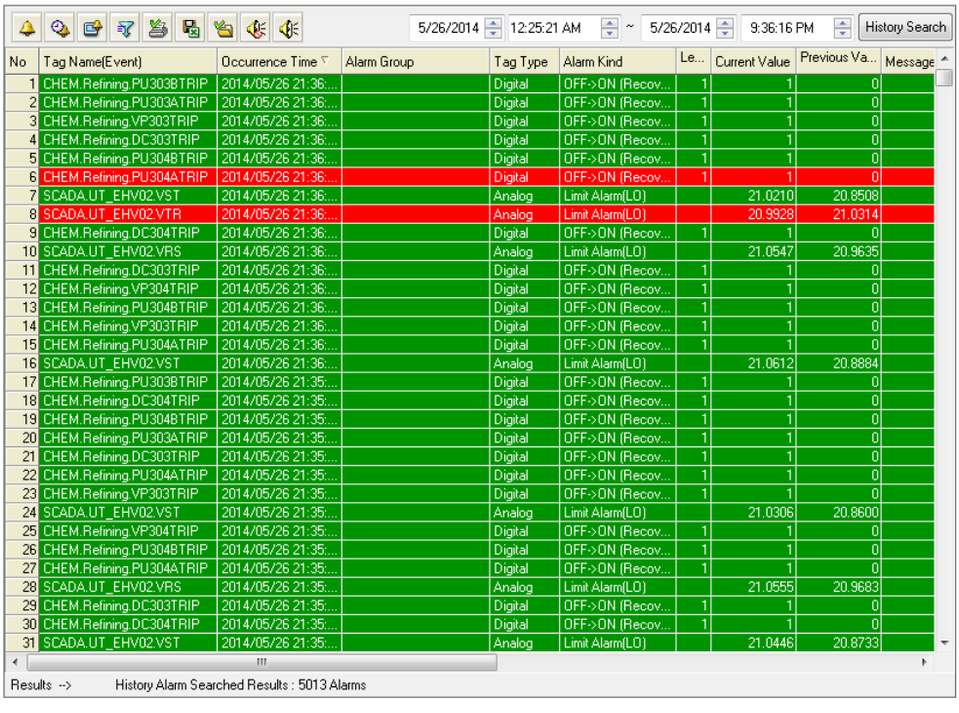
(2) Process Analysis

This function is to move the alarm that takes place on the real-time alarm list to the history alarm list for analyzing. Process Analysis is useful when viewing the alarms before and after the occurred alarm in the real-time.

Select an alarm on the real-time alarm list and press the right side of the mouse to display the following menu. Select Process Analysis from the menu to move the relevant alarm to the history alarm list for analyzing. Once moved to the history alarm list, the selected alarm is highlighted in red for convenient identification.



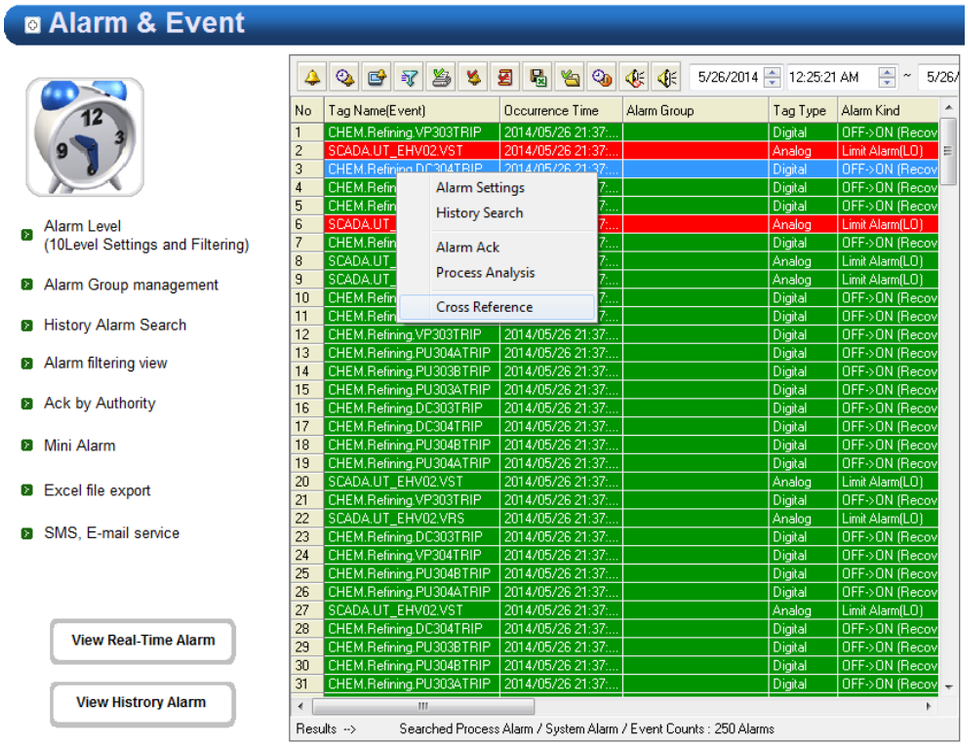
The following screen shows the result of moving to the history alarm list.



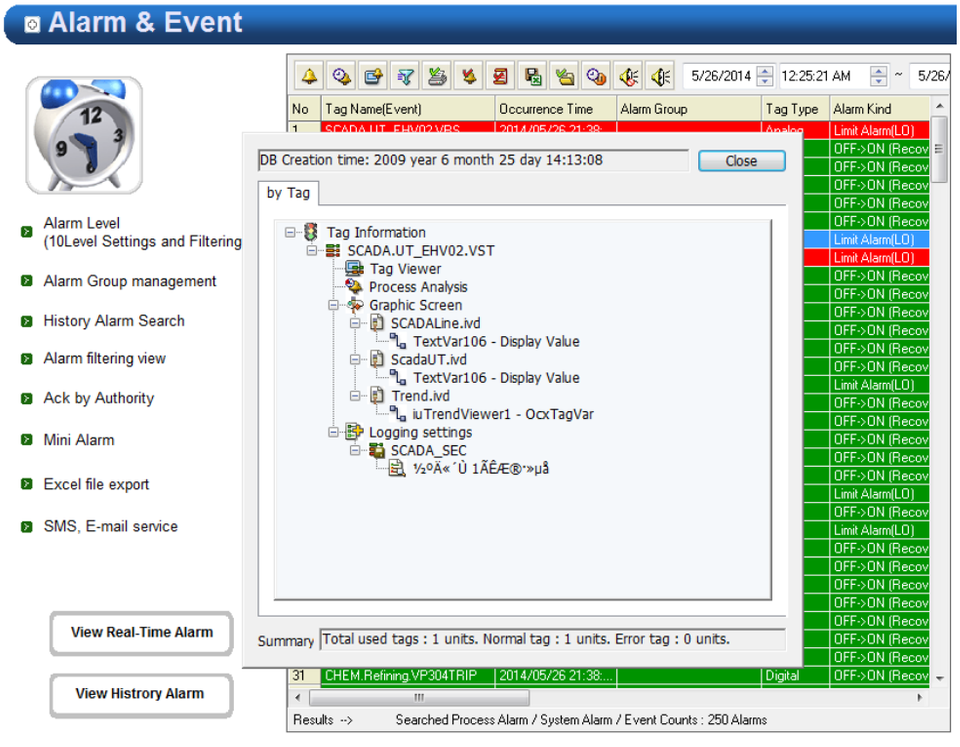
1) Cross Reference

This function is to make the screen move to the graphic screen to which the selected alarm belongs.

This function helps to immediately identify at which object the current alarm takes place and analyze the current situation by moving to the screen.

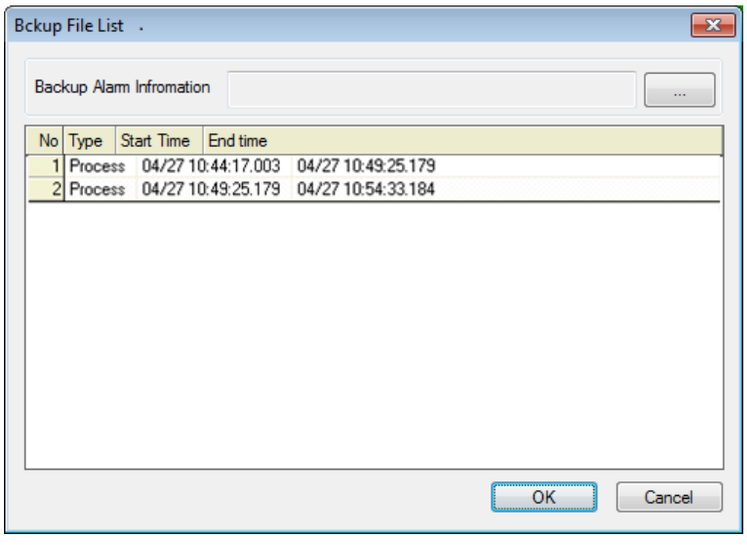


Select an alarm on the real-time alarm list and press the right side of the mouse to display the following menu. Select Cross Reference from the menu to display the following Cross Reference screen. If the user selects one among the screens to which the occurred alarm belongs, it moves to the relevant screen. Select a screen and double click the mouse to move to the screen.



2) Backup Alarm Import

If Backup Alarm Import button is selected, the following screen appears. The list shows backup files on the saving path. If a list is selected and Ok button is pressed, a list of the alarms in the backup files is displayed.



11.2.7 How to Use Script

(1) Function List

Function Name	Function Declaration
AckPageAlarm	long AckPageAlarm(void)
AckRecentAlarm	long AckRecentAlarm(void)
AckSelectAlarm	long AckSelectAlarm(void)
DisplayHistoryAlarm	long DisplayHistoryAlarm(void)
DisplayRealAlarm	long DisplayRealAlarm(void)
ExportToExcel	long ExportToExcel(void)
LoadPeriodSearch	void LoadPeriodSearch(LPCTSTR strFrom, LPCTSTR strTo)
SetAlarmGroupList	void SetAlarmGroupList(LPCTSTR strGroupList)
SetAlarmList	long SetAlarmList(long alarmKind, LPCTSTR str)
SetDetailStringFiltering	void SetDetailStringFiltering(LPCTSTR strGroupList)
SetHistoryList	long SetHistoryList(long alarmKind, LPCTSTR str)
SetFilteringOption	long SetFilteringOption(ushort almType, LPCTSTR startTime, LPCTSTR endTime, LPCTSTR tagName, short tagType, short almKind, short almOccurType, short almGrade, short ackGrade, LPCTSTR almGroup)
SetTagHistoryFiltering	void SetTagHistoryFiltering(LPCTSTR strTagFilter)
ShowConfigDlg	void ShowConfigDlg(void)
ShowFilteringDlg	long ShowFilteringDlg(void)

StartSound	long StartSound(void)
StopAlarm	long StopAlarm(long bFlag)
StopSound	long StopSound(void)
PrintList	long PrintList(void)

(2) Function Description

AckPageAlarm Function	
Description	It is used when the user wants to acknowledge the entire alarms seen on the current page of the alarm viewer.
Syntax	long AckPageAlarm(void)
Reference	To acknowledge the entire alarms on the page AckPageAlarm ()

AckRecentAlarm Function	
Description	It is used to recognize the currently visible latest alarm in the alarm viewer.
Syntax	long AckRecentAlarm(void)

AckSelectAlarm Function	
Description	It is used to recognize the currently selected alarm in the alarm viewer.
Reference	long AckSelectAlarm(void)

DisplayHistoryAlarm Function	
Description	It is used to display history alarms in the alarm viewer.
Syntax	long DisplayHistoryAlarm(void)

DisplayRealAlarm Function	
Description	It is used to display real time alarms in the alarm viewer.
Syntax	long DisplayRealAlarm(void)

ExportToExcel Function	
Description	It is used when the user wants to export the alarms taken place on the alarm viewer to Excel.
Syntax	long ExportToExcel(void)
Reference	To export the occurred alarms to Excel ExportToExcel ()

LoadPeriodSearch Function	
Description	It is used to retrieve the section with the start time and end time in the alarm viewer.
Syntax	void LoadPeriodSearch(LPCTSTR strFrom, LPCTSTR strTo) strFrom: Search from start time strTo: Search to end time

SetAlarmGroupList Function	
Description	This function is to allow the user to search only the alarm groups he/she wants and make them appear on the relevant list. With this function, the user can see the list by alarm groups in the early stages.
Syntax	void SetAlarmGroupList(LPCTSTR strGroupList) strGroupList: Insert the alarm groups in a form of string values. Be careful about distinguishing one alarm group from the other with “;” and not forgetting to insert “;” at the end.
Reference	To see AlarmGroup_1 and AlarmGroup_2 SetAlarmGroupList (“AlarmGroup_1;AlarmGroup_2;”)

SetAlarmList Function	
Description	This function is to allow the user to select an alarm kind to make it appear on the relevant list. With this function, the user can see the real-time alarm list by kinds in the early stages.
Syntax	long SetAlarmList(long alarmKind, LPCTSTR str) alarmKind: Enter an alarm kind. Insert the following values according to alarm kinds. 0 - All 1 - Process Alarm 2 - System Alarm 3 - Process Alarm + System Alarm 4 - Event 5 - Process Alarm + Event 6 - System Alarm + Event str: enter empty value (“”).
Reference	To see a process alarm list SetAlarmList(1, “”)

SetDetailStringFiltering Function	
Description	It is the function to apply the string filtering value for the desired alarm groups only of the alarm viewer and retrieve and display them on the list.
Syntax	void SetDetailStringFiltering(LPCTSTR strGroupList) strGroupList: Input the string filtering value of the alarm group name to retrieve. “;” is the separator in word unit and “+” is the separator in string unit.
Reference	When you want to view the alarm groups called Group_1 and Group_2 with the string. SetAlarmGroupList (“Group_1;Group_2;”)

SetHistoryList Function	
Description	This function is to allow the user to select an alarm kind to make it appear on the relevant list. With this function, the user can see the history alarm list by kinds in the early stages.
Syntax	long SetHistoryList(long kind, LPCTSTR str) kind: Enter an alarm kind. Insert the following values according to alarm kinds. 0 - All 1 - Process Alarm 2 - System Alarm 3 - Process Alarm + System Alarm 4 - Event 5 - Process Alarm + Event 6 - System Alarm + Event Str: enter empty value (“”).
Reference	To see a process alarm list out of history alarm lists SetHistoryList(1, “”)

SetFilteringOption Function	
Description	This function is to filter and see alarms by kind or group with several filtering options. This function applies only to history alarms. It does not apply to real-time alarms.
Syntax	long SetFilteringOption(ushort almType, LPCTSTR startTime, LPCTSTR endTime, LPCTSTR tagName, short tagType, short almKind, short almOccurType, short almGrade, short ackGrade, LPCTSTR almGroup) almType: Enter an alarm kind. Insert the following values according to alarm kinds. 0 - All 1 - Process Alarm 2 - System Alarm

	3 - Process Alarm + System Alarm 4 - Event 5 - Process Alarm + Event 6 - System Alarm + Event startTime: Write a start time to search. Write a string value in a form of “2007/08/17.” Enter empty value (“”) if the user wants to search regardless of the start time. endTime: Write an end time to search. Write a string value in a form of “2007/08/17.” Enter empty value (“”) if the user wants to search until now. tagName: Write a tag name to search. Enter empty value (“”) to search the entire tags. tagType: Enter a tag type. Enter 1 for analog tags and 2 for digital tags. almKind: Enter 0. almOccurType: A parameter to select one from occurrence, recovery or Ack of alarm and search Enter the following value. 1 - Occurrence / Recovery / Ack 2 - Occurrence / Recovery 3 - Occurrence / Ack 4 - Occurrence 5 - Recovery / Ack 6 - Recovery 7 - Ack almGrade: Enter an alarm grade to search. ackGrade: Enter an ack grade to search. almGroup: Enter an alarm group to search. If the user wants to search multiple groups, he/she needs to distinguish one group from the other with “;”. If empty value or 0 is entered for each parameter, searching is conducted regardless of the conditions of the relevant parameter.
Reference	To search the recovered alarms and ack alarms among the alarms that took place in Alarm_Group1 from August 15 through 17, 2007 SetFilteringOption (0, “2007/08/15”, “2007/08/17”, “”, 0, 0, 5, 0, 0, “Alarm_Group1”)

SetTagHistoryFiltering Function	
Description	It is the function to apply the string filtering value for the desired history tags of the alarm viewer and retrieve and display them on the list.
Syntax	void SetTagHistoryFiltering(LPCTSTR strTagFilter); strTagFilter: Input the string filtering value of the tag history to retrieve. “;” is the separator in word unit and “+” is the separator in string unit.
Reference	When you want to view the history alarm tag called Tag_1 and Tag_2 with the string. SetAlarmGroupList (“Tag_1;Tag_2;”)

ShowConfigDlg Function	
Description	The configuration setting window of the alarm viewer is shown.
Syntax	void ShowConfigDlg(void)
Reference	To show the configuration setting window ShowConfigDlg()

ShowFilteringDlg Function	
Description	The filtering search window of the alarm viewer is shown.
Syntax	long ShowFilteringDlg(void)
Reference	To show the filtering search window ShowFilteringDlg ()

StartSound Function	
Description	It is used to generate the sound when an alarm occurs.
Syntax	long StartSound(void)
Reference	When you want to turn on the alarm sound StartSound ()

StopAlarm Function	
Description	It is used when stopping for a while or restarting the alarm viewer.
Syntax	long StopAlarm(long bFlag) bFlag: - To stop the alarm: True - To start the alarm: False
Reference	To stop the alarm StopAlarm(True) To restart the alarm StopAlarm(False)

StopSound Function	
Description	It is used when stopping alarm sound.
Syntax	long StopSound(void)
Reference	To stop alarm sound StopSound ()

PrintList Function	
Description	It is used when printing the alarms that take place on the alarm viewer.
Syntax	long PrintList(void)
Reference	To print the occurred alarms PrintList ()

11.3 Trend Viewer

This Trend Viewer is an Active X module that displays charts monitoring online and logging data on the tags registered in the Trend.

As a screen to monitor changes in tag values over time, this Trend Viewer is an Active X module that displays online tag data graphs as well as logging data graphs stored in the logging server.

In addition, this Trend can be used in analyzing changes in logging data.

11.3.1 Prerequisite and Environment

Trend properties can be set up only after the ActiveX module is inserted to the InfoU graphic editor.

Channel assignment, tag assignment, X axis/Y axis value input and other tag properties to be monitored are designated in advance as Trend properties

Channels or tags are not basically assigned to the trend status inserted in the graphic editor. It is essential to display the Property Window before setting up the property values needed to run the Trend.

After the property values are saved, the Trend Monitoring Screen appears upon the opening of the screen under the Run time environment.

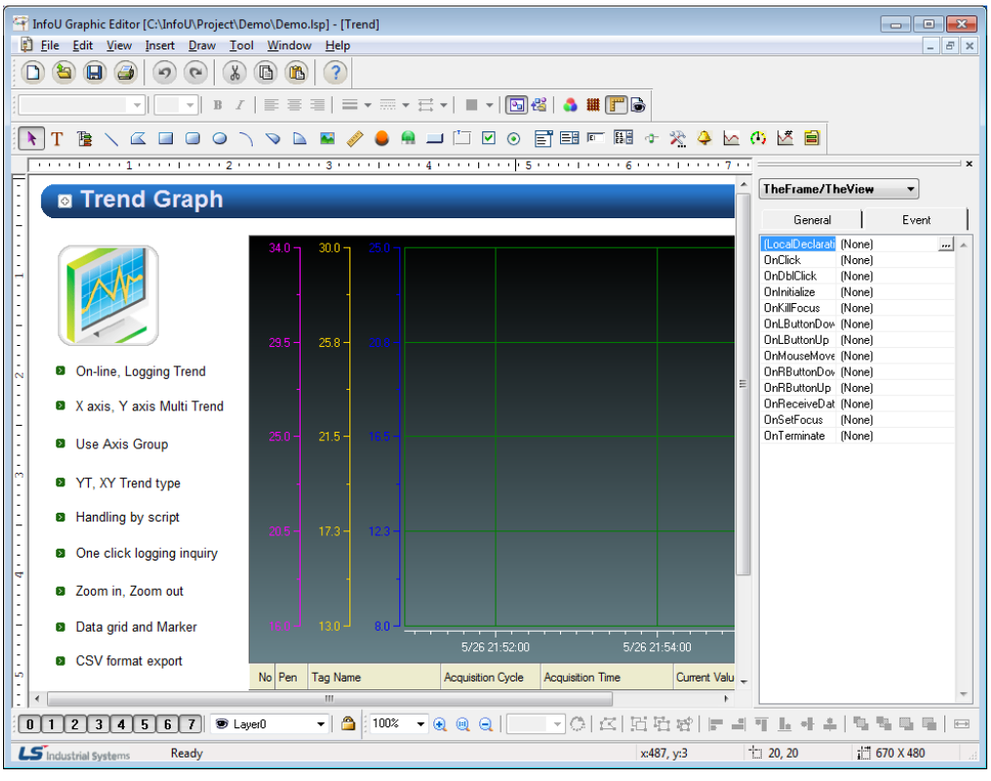
The engineering values are entered in the following order on the Trend viewer.

If  is selected from the tools, immediate inserting can be performed.

11.3.2 Screen Configuration

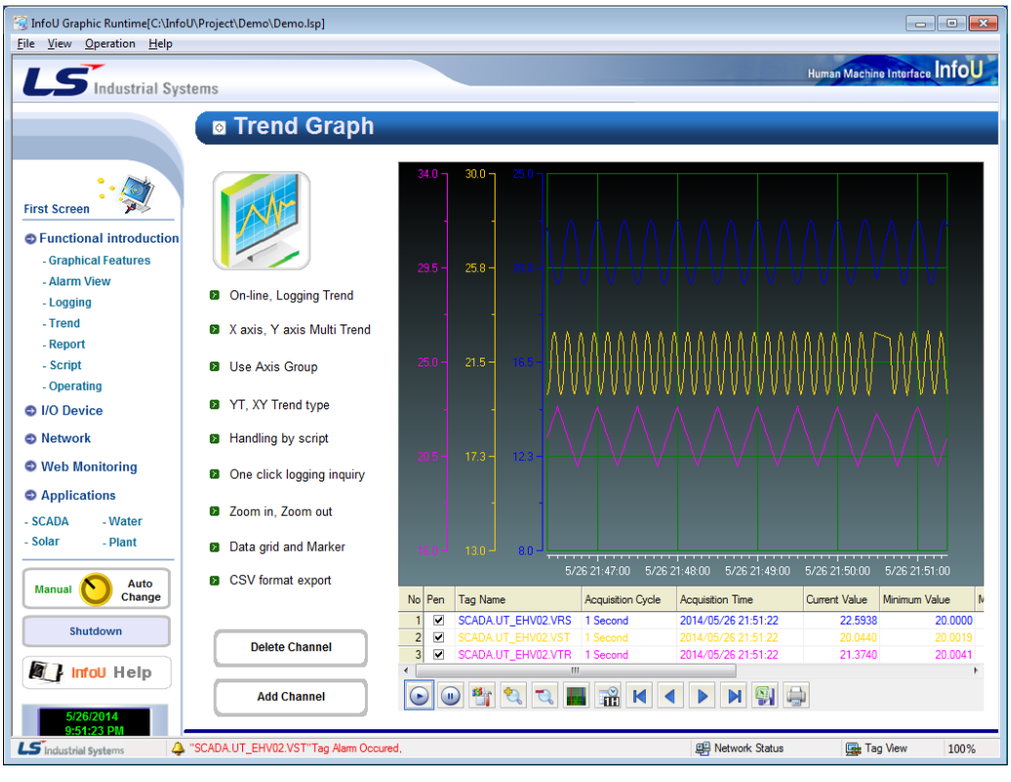
(1) Engineering Screen

If Trend is inserted and a tag to be monitored is selected and then settings are all completed, the following screen appears. The screen in this figure shows that three tags are selected and gradation effect is applied.



(2) Runtime Screen

If default trend settings are completed on the engineering screen above and then, the runtime is executed, the screen shows that data values are changed over time.

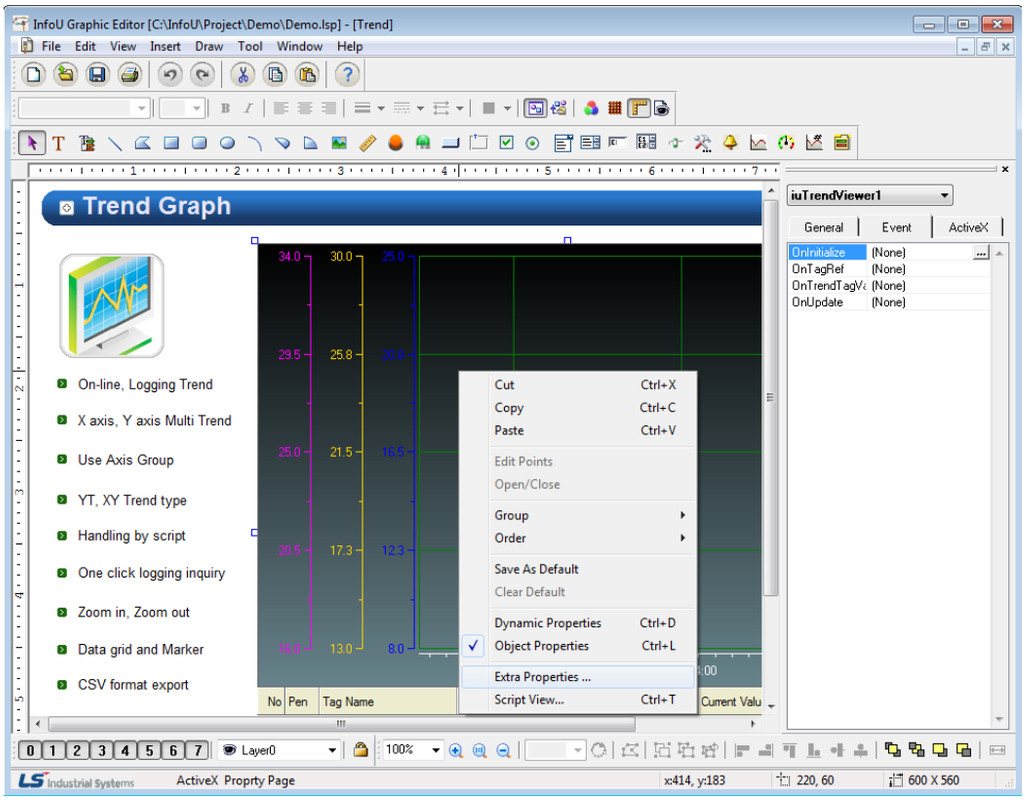


11.3.3 Settings

If the right side of the mouse is clicked on the Trend and the 'Active X Properties' menu is selected, the setting window appears.

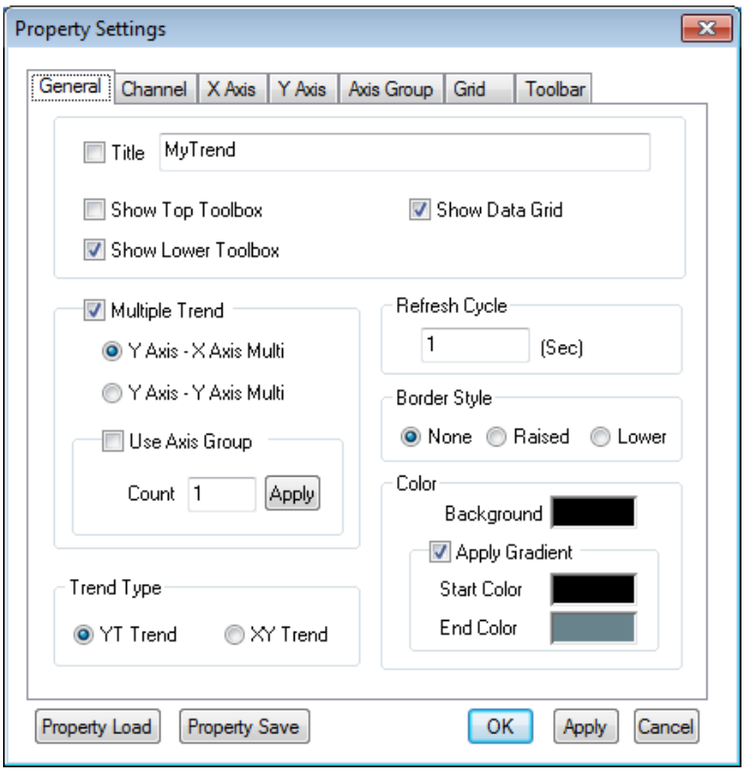
The same setting screen is used for the engineering screen and the runtime screen. But, the runtime screen does not have Grid Settings and Toolbar Selection functions.

In addition, once property values are saved on the engineering screen, those values will be continuously valid however, if those property values are changed on the runtime screen, the changed values are applied only to the runtime screen. The changed values are cancelled if the runtime stops and the property values set up on the engineering screen are applied.



(1) General

The Engineering Screen and the Runtime Screen have the same general functions.



1) Title

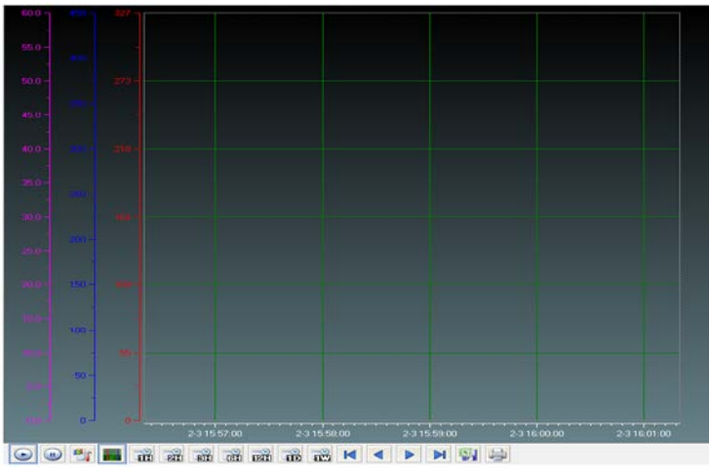
Designate a title of the trend.

2) Show Top Toolbox/Show Lower Toolbox

Set up whether to display Trend Toolbox on the top or the bottom during the running. Both can not be selected at the same time and it is allowed neither of them is selected. ‘Show Lower Toolbox’ has been selected as the default value.

3) Show Data Grid

Select whether to show or hide data grid during the running. The following figure shows the screen when ‘Show Data Grid’ is not selected. ‘Show Data Grid’ has not been selected as the default value.

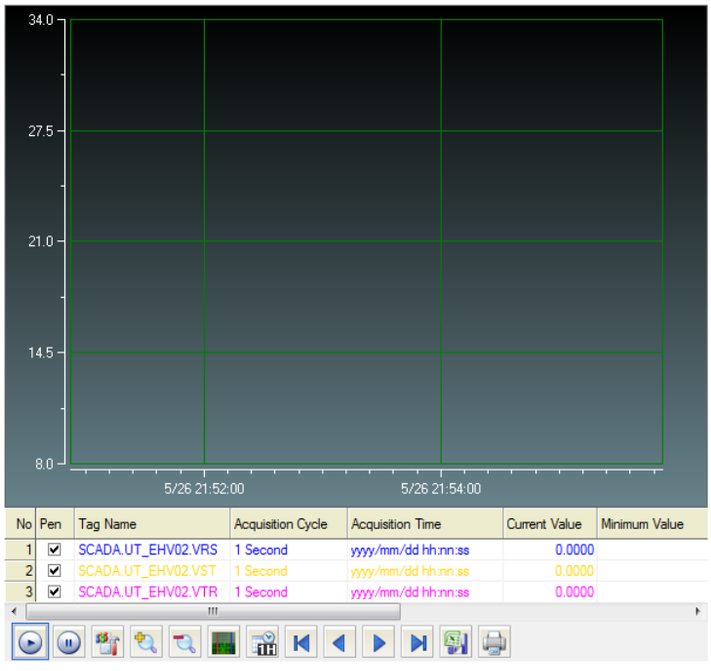


a. Multiple Trend

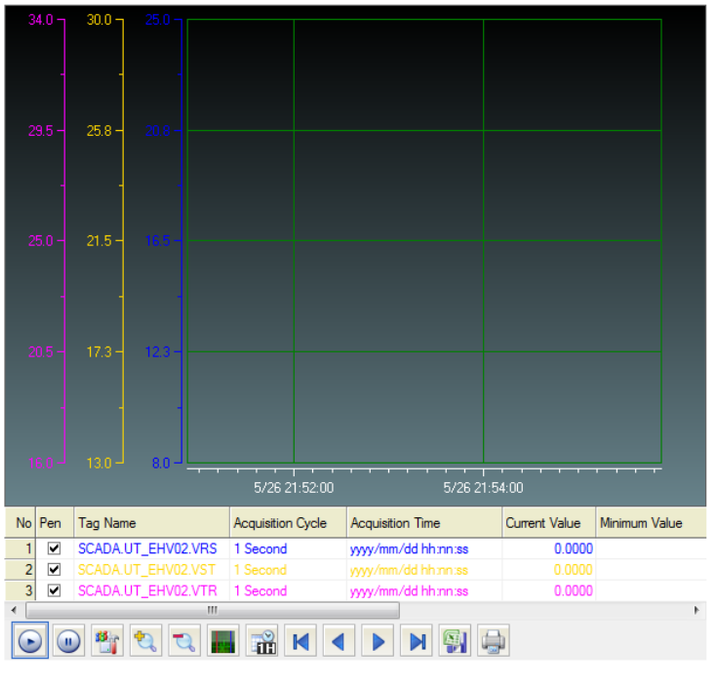
Select ‘Multiple Trend.’

In general trends, the axis that takes the lowest value among the minimum values as the min value and the highest value among the maximum values as the max value is commonly used. However, if 'Multiple Trend' is selected, any axis can be used individually.

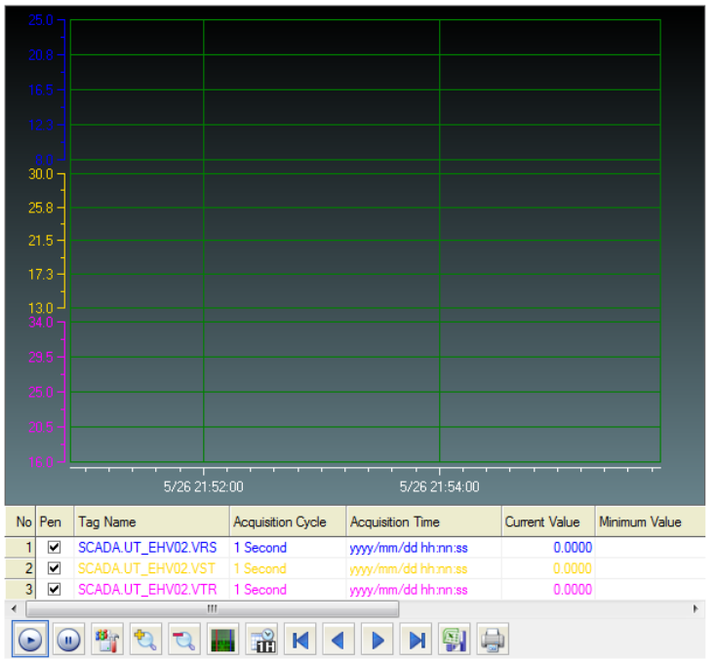
A general trend screen is as following.



A Multiple Trend screen that selects x-axis is as following.



A Multiple Trend screen that selects y-axis is as following.



Multiple trends can use Axis Group.

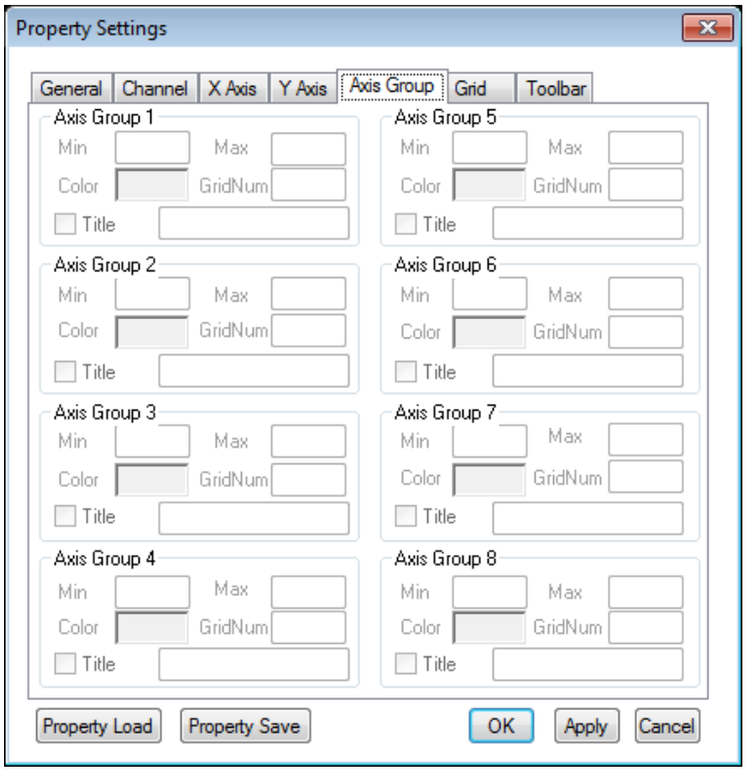
Axis Group is a function to re-divide several channels into desired groups to view them.

It is used as following.

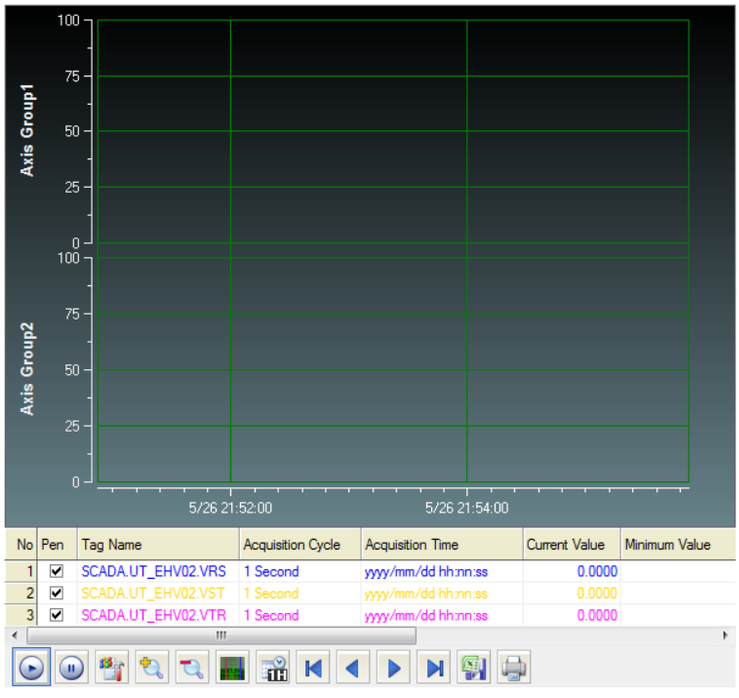
- Check ‘Use Axis Group.

- Input the number and press ‘Apply’ button.

- Once it moves to ‘Axis Group’ tap upon the ‘Apply’ button, Axis Groups as much as the inputted number are activated. Input the min value and max value here.



If it is set up that Channel 1 and 2 belong to Axis Group 1 and Channel 3 belongs to Axis 2, the following screen appears.



4) Refresh Cycle

Designate a refresh cycle. One second has been set up as the default value.

5) Color

Select a background color and set up ‘Apply Gradient.’ If ‘Apply Gradient’ is selected, the background color is not applied. In addition, it needs to set up ‘Start Color’ and 'End Color’.

6) Border Style

Select a Border Style for the relevant trend.

7) Trend Type

Select either YT Trend or XY Trend.

If YT Trend is selected, x axis indicates time and Y axis shows tag values while if XY Trend is selected, both X and Y axis consist of tag values.

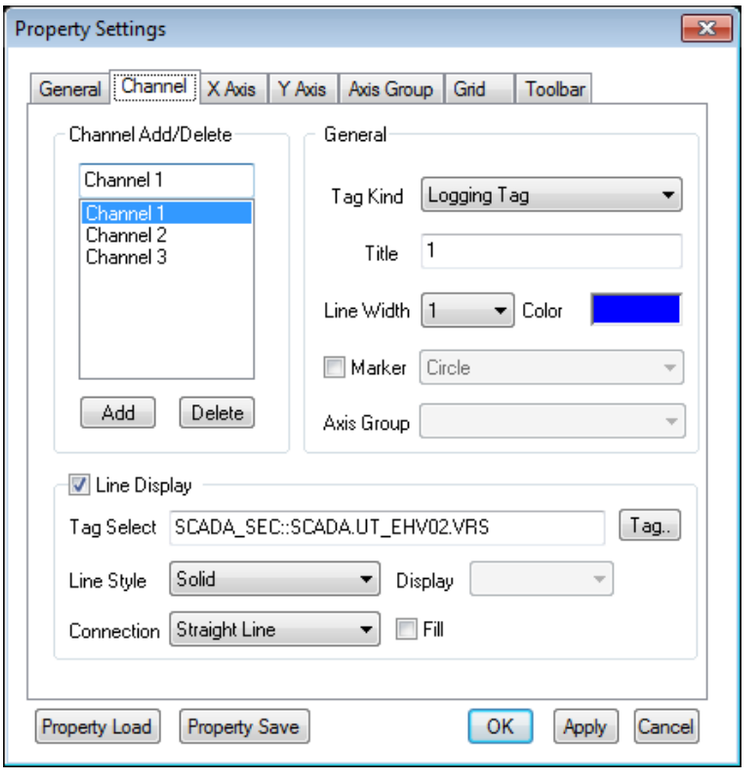
(2) Channel

The Engineering Screen and the Runtime Screen have the same Channel functions.

Up to 20 channels are provided in the Trend. There is at least one channel and it is available to designate Tag Kind, Title, Color, Marker, Tag, Line Style and Connection for each channel.

If one channel is an online tag, the rest channels have to be online tags. Likely, if one channel is a logging tag, the rest channels have to be logging tags.

Also, if any logging data needs to be saved or searched, the logging tag should be selected and used. Online tags do not provide the logging function.

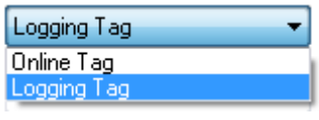


1) Channel Add/Delete

A channel is added or deleted. At least one or more than one but not more than 20 channels can be selected.

2) Tag Kind

Select either Logging Tag or Online Tag.



3) Title

Designate a title of the channel.

4) Line Width

Select a Line Width of the channel. Numbers O through 4 are available.

5) Color

Decide color of the channel.

6) Marker

As a point to indicate the tag value of the relevant channel, the following shapes are available.



7) Tag Select

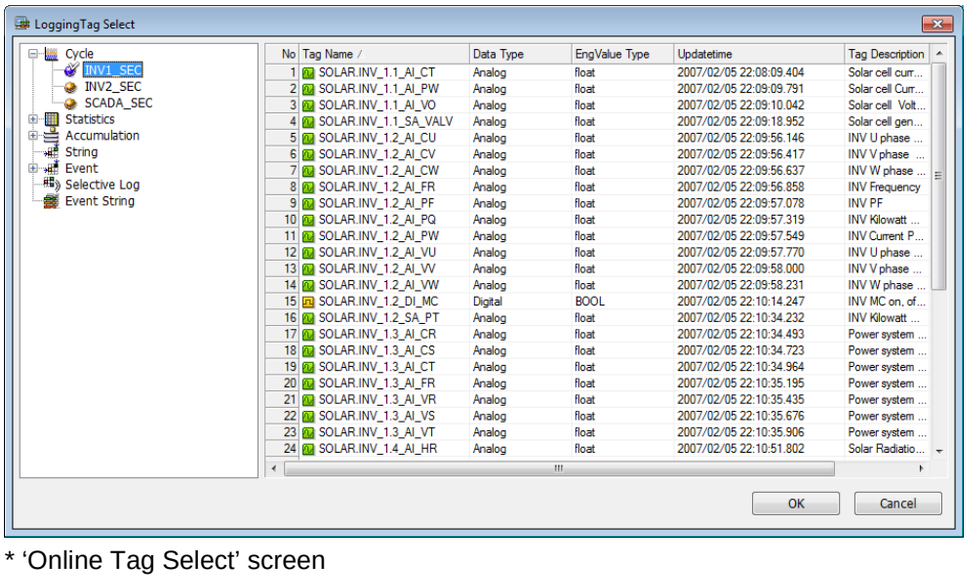
Assign a tag to the channel. One tag can be assigned to one channel.

If a logging tag is selected, the logging tag browser where the associated logging group is set up is called while the online tag browser is called for an online tag.

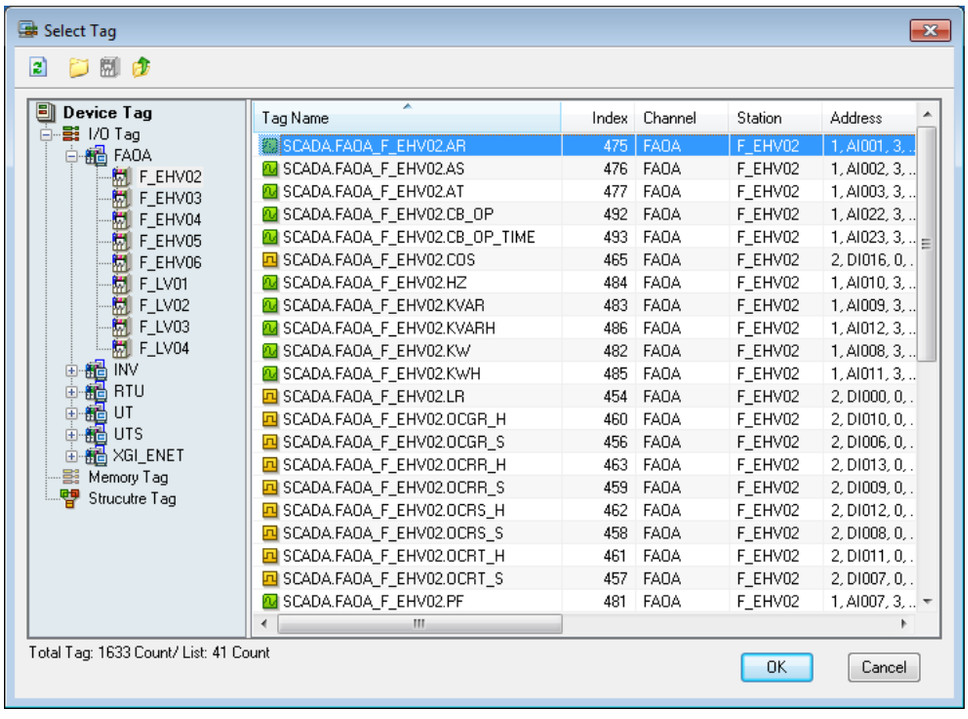
Be cautious that logging data can be searched only when a logging tag is selected.

If a tag is selected, the tag's minimum value and maximum values are displayed. These min/max values apply to Y-axis values for Y axis settings.

* 'Logging Tag Select' screen

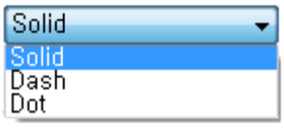


* 'Online Tag Select' screen



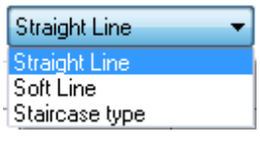
8) Line Style

Select one among Solid, Dash and Dot.



9) Connection

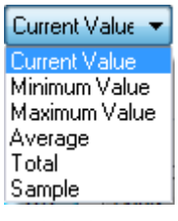
Select one among Straight Line, Soft Line and Staircase Type.



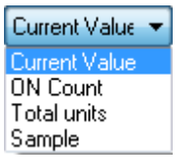
10) Display

This function becomes active only when a statistics tag is selected. For analog tags, select one value among current value, min value, max value, average value, sum and sample and for digital tags, select one value among current value, On count, On Time, Total count and sample. If the tag is not a statistics tag, it becomes inactive.

* Analog Tag



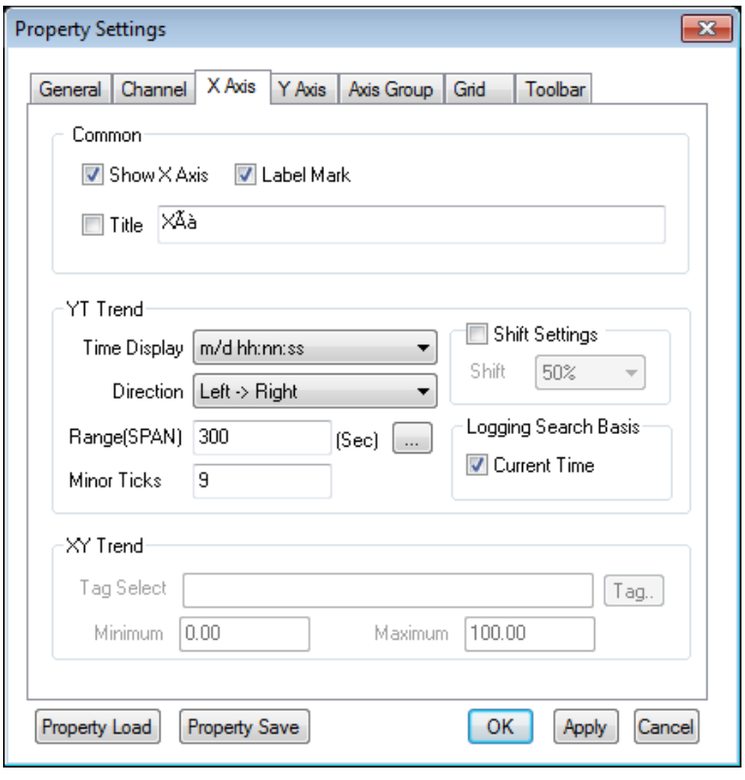
* Digital Tag



(3) X Axis

The Engineering Screen and the Runtime Screen have the same X axis functions.

If XY Trend is selected on the General tap, setting items for YT Trend become inactive while those for XY Trend become inactive if YT Trend is selected.



1) Common

a. Show X Axis

Decide whether to show or hide X Axis.

b. Label Mark

Select whether to mark labels on X axis or decide a title for identification.

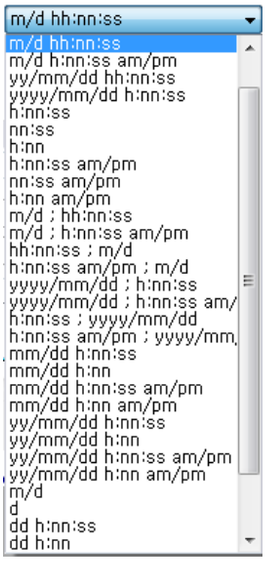
c. Title

Decide whether to mark a title on X axis.

2) YT Trend

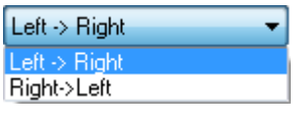
a. Time Display

Decide one among various Time Display types.



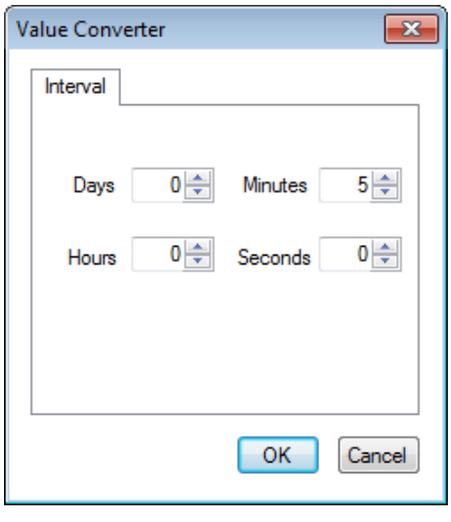
b. Direction

Select either ‘Left->Right’ or ‘Right->Left.



c. Range (Span)

Save a range of X-axis values. Input a value directly or select one from the range window. Input time and the unit is second.



3) Shift Settings

The Shift function refers to moving the current value to the left as much as the shift setting value of x axis if the graph goes to the end. If the user wants this function, he/she needs to set up Shift and press ‘Apply’ button. Available values are from 0 through 100 %. Shift has not been set up as the default value and it starts after drawing a range of X axis.

4) History retrieval

In the case of Trend, basically, history retrieval is done at a distance of regular time. When selecting the current time, history retrieval will be performed based on the current time.

5) XY Trend

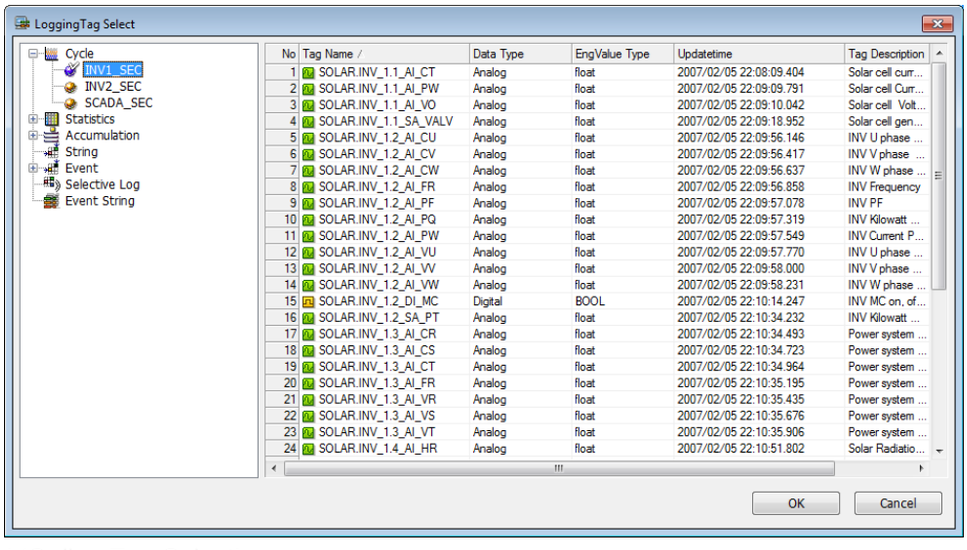
a. Minimum/Maximum

Specify the min/max value of the relevant tag to be marked on X-axis. Only one tag can be on X axis.

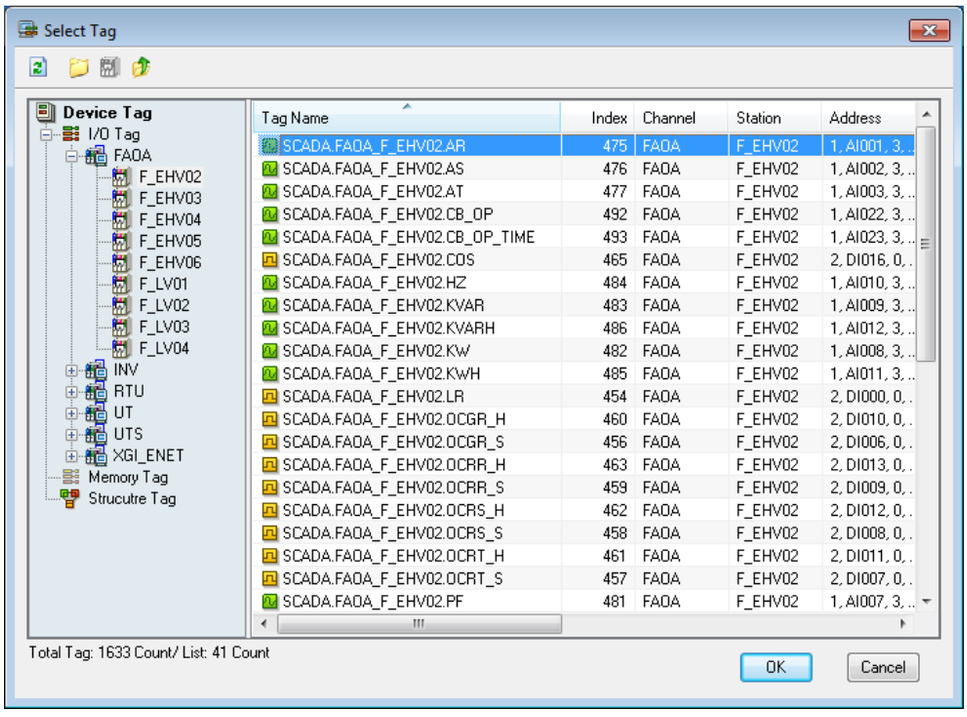
b. Tag Select

Assign a tag to the channel. If a logging tag is selected, the logging tag browser where the associated logging group is set up is called while the online tag browser is called for an online tag. Logging data can be searched only when a logging tag is selected.

* 'Logging Tag Select' screen

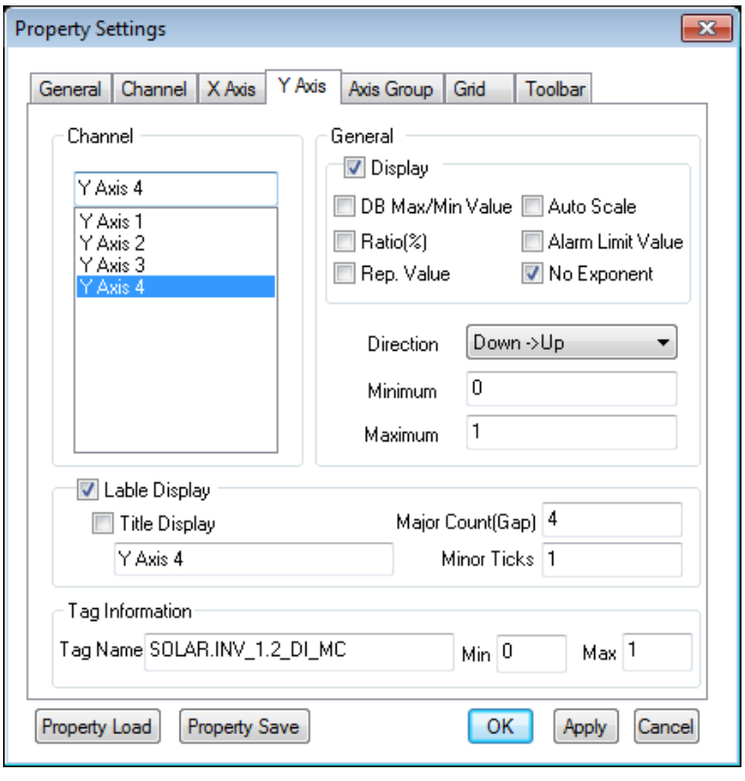


* 'Online Tag Select' screen



(4) Y Axis

The Engineering Screen and the Runtime Screen have the same Y axis functions.



1) Channel

Show the channels that are assigned to Y axis.

2) Display

Select either show/hide Y axis.

3) DB Max/Min Value

DB Max/Min Value is used instead of user's max/min value.

4) Auto scale

For Y-axis's min./max. value of each channel, use the min./max. value of the currently occurring channel.

5) Ratio (%)

Display Y axis value in Ratio not in tag value. If Display in Ratio is selected for even one Y axis, Y axis is displayed in percentage unless Multi Trend has selected. Also, if this item is selected, moving with the mouse becomes impossible along Y axis.

6) Alarm Limit Value

If this item is selected, HH value and LL value of tag alarm are drawn in lines and Alarm Limit Value is displayed.

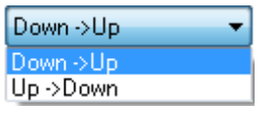
HH value and LL value are not visible.

7) Use of typical value

It sets the min./max. value of a specific channel as the typical axis and displays them. It is not applied to the multi trend.

8) Direction

Select either 'Down->Up' or 'Up->Down'.



9) Minimum/Maximum

Set up max/min value needed to show Y axis. Tag's max/min value has been set up as the default value.

10) Use Tag Information

Show tag information of the channel, which has been set up on Channel tap. Tag name and tag's max/min value are displayed.

11) Label Display

Select whether to display label and how to mark the title.

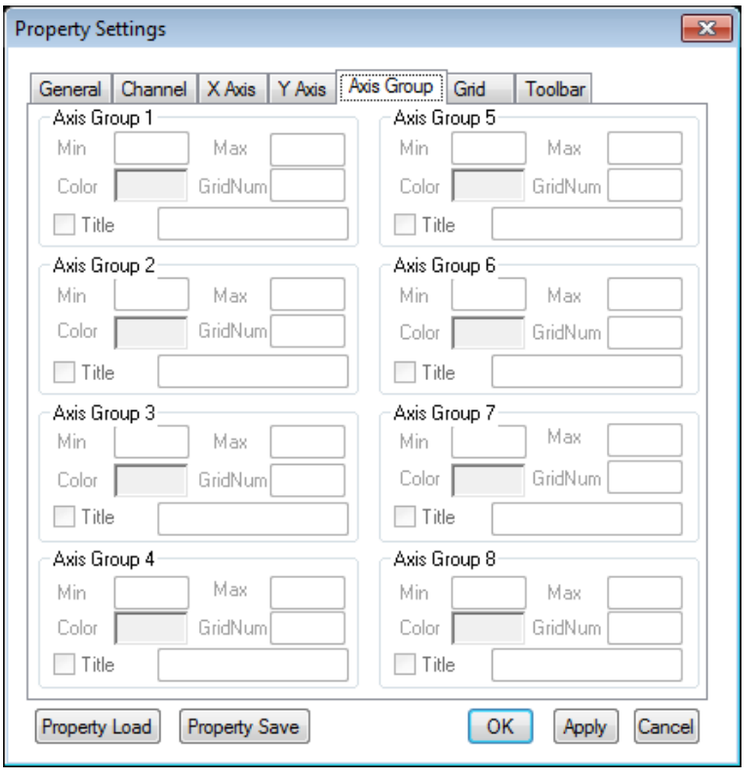
12) Grid Count

Grid Count is the number of lines for Y axis values visible on the screen. This value is adjusted to display max/min value. But, this value may not be adjusted in case that Y axis is too narrow to display channels.

(5) Axis Group

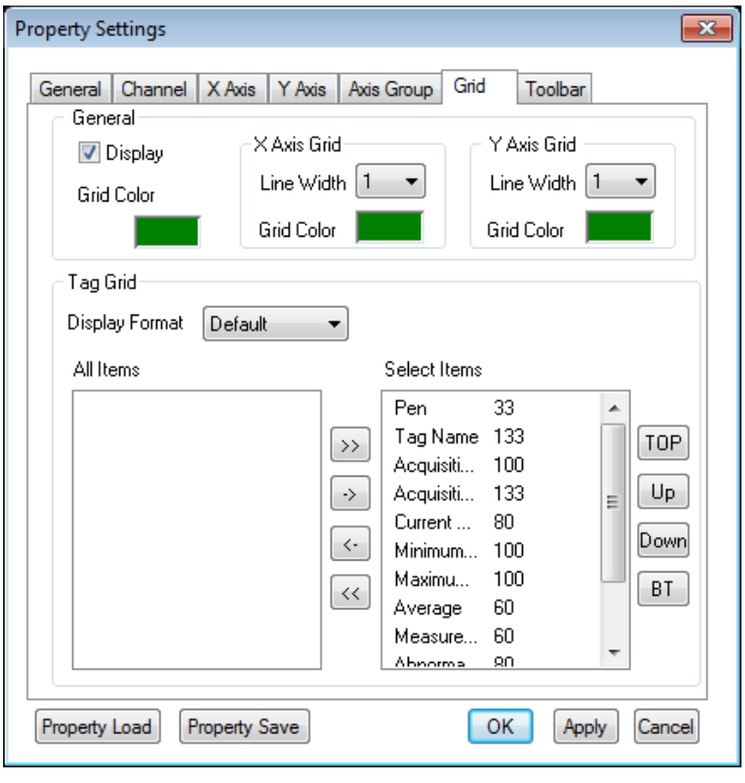
This tap becomes active when 'Multi Trend' is selected on the General tap and Axis Group is applied.

Input the min/max value and title if necessary for each Axis Group.



(6) Grid

This Grid setting function is only available on the Engineering Screen. It is not displayed on the Runtime Screen. It consists of General for color settings and Tag Grid for tag grid item and order settings.



1) General

a. Grid Color

Select a color of grid border.

b. X Axis Grid

Select Line Width and Grid Color for X Axis Grid.

c. Y Axis Grid

Select Line Width and Grid Color for Y Axis Grid.

2) Tag Grid

Select alarm items to display on the viewer.

'Select Items' include Pen, Tag Name, Acquisition Cycle, Acquisition Time, Current Value, Minimum, Maximum, Zero, Span, Tag Kind, Status and Tag Description.

Numbers next to items indicate the column size of each item on the Trend Viewer and they are changeable. Double click on the item to change and insert a new size.

a. >> Button

Move the marked item from the entire list to the selected list.

b. << Button

Move the unmarked item from the entire list to the selected list.

c. -> Button

Move from the entire list to the selected list.

d. <- Button

Move from the selected list to the entire list.

e. TOP Button

Move the marked location on the selected column from the selected list to the top left.

f. ▲ Button

Move the marked location on the selected column from the selected list to the next cell on the left side.

g. ▼ Button

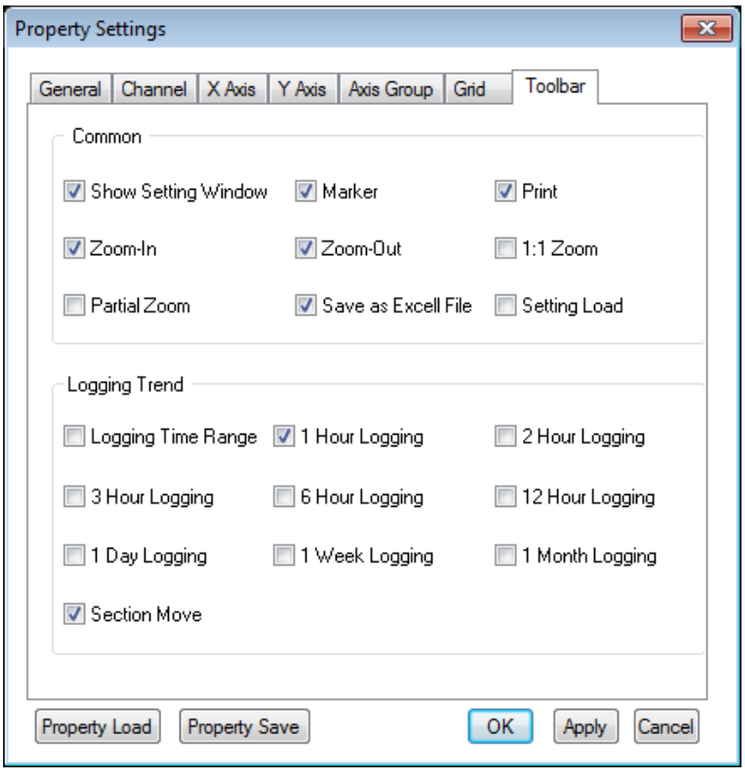
Move the marked location on the selected column from the selected list to the next cell on the right side.

h. BT Button

Move the marked location on the selected column from the selected list to the end on the right side.

(7) Toolbar

Select Toolbar (menus).



1) Start

This is a common function that applies to both online trends and logging trends,

Once trend monitoring starts, data is renewed every second and 5 minutes is set as the time for X axis.

To view the current screen again during the logging search, press this button and the current time screen is displayed.

This Start button can not be selected as the default value.

2) Stop

Trend Monitoring stops. This function is available only for logging trends. That is, this function exists only when logging tags are selected and online trends do not have this function.

This Stop button can not be selected as the default value like the Start button.

3) Show Setting Window

Setting values of the current trend can be changed. This is a common function that applies to both online trends and logging trends.

This Setting Window has been described before.

4) Zoom-in

The current trend screen is enlarged twice to search data. Zoom-in makes both of X axis and Y axis enlarged twice. This is a common function that applies to both online trends and logging trends.

5) Zoom-out

The current trend screen is reduced twice to search data. Zoom-in makes both of X axis and Y axis reduced twice. This is a common function that applies to both online trends and logging trends.

6) 1:1 Zoom

This function is used to return the enlarged or reduced trend screen into its original size -100% screen.

This is a common function that applies to both online trends and logging trends.

7) Partial Zoom

The trend screen is partly zoomed in/out to search data. A part of the trend screen is dragged with the mouse and it becomes enlarged as soon as the mouse is released. At this time, both X axis and Y axis are enlarged.

This is a common function that applies to both online trends and logging trends.

8) Save as Excel File

Convert and save the current trend being monitored into a csv file.

The csv file can be opened in Excel and it can be viewed as normal text.

9) Print

Print the current trend being monitored. In printing, the background screen is changed in white color.

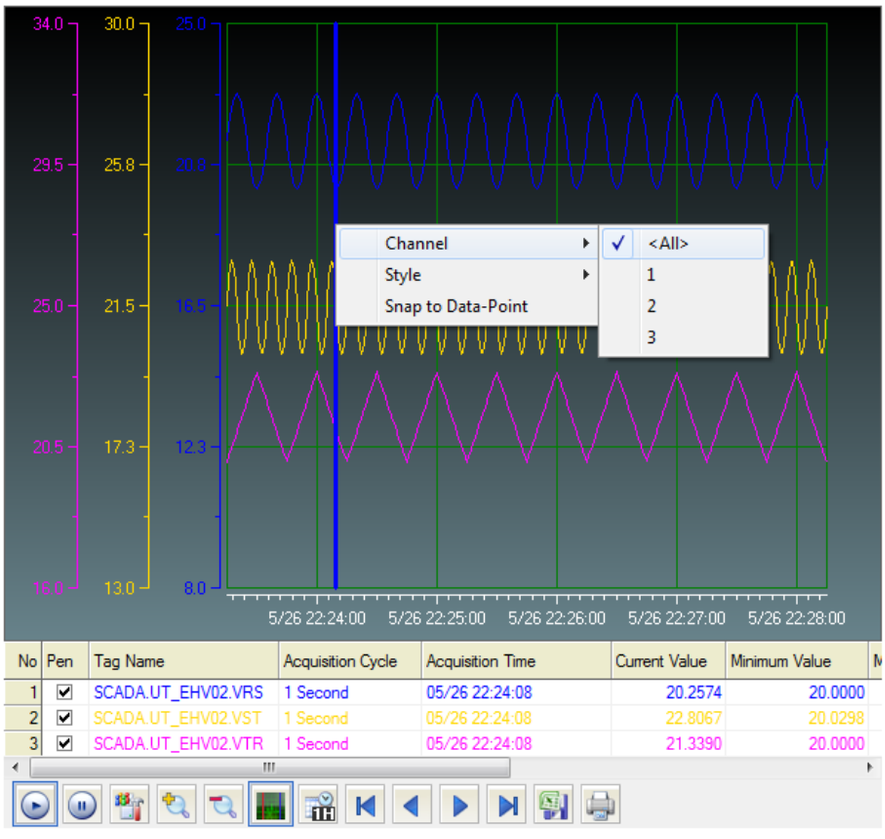
10) Load properties

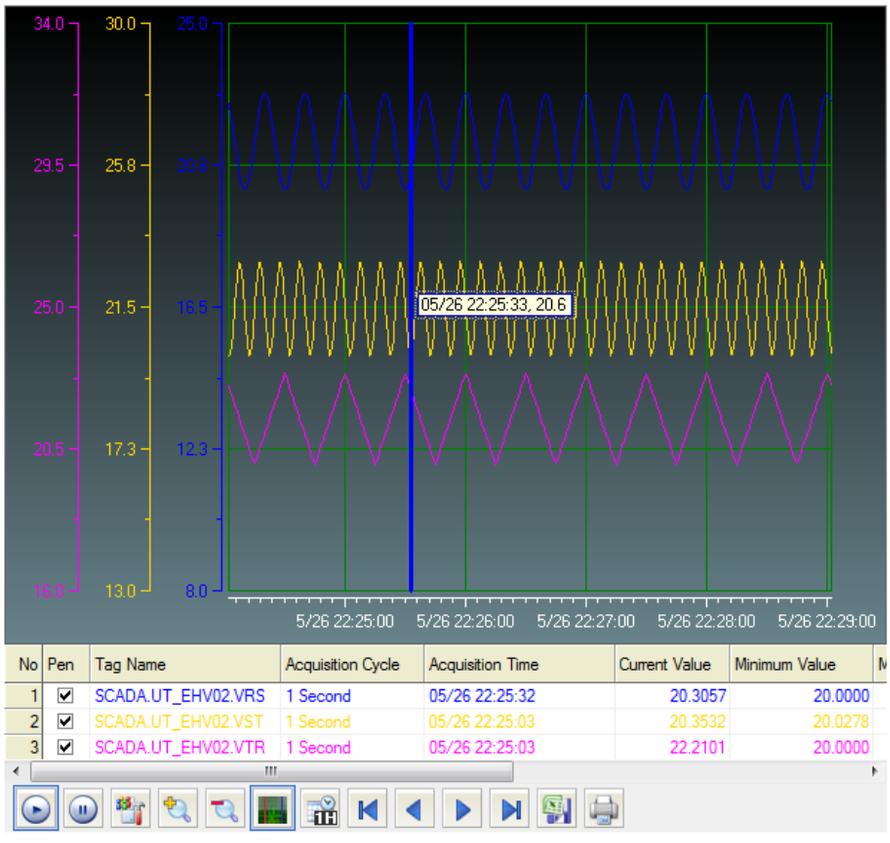
It shows the saved trend property setting file in the form of a list. If you select the desired property settings, they will be applied to the current trend. For more details, refer to ‘Load/Save Properties’.

11) Marker

If Marker button is pressed, a yellow marker line is displayed and values of channels are also listed on the right screen. If the right button of the mouse is pressed on the marker, the channel value can be seen in a tool tip type. To see another channel value, select it on the right screen. At this time, the marker color is the same with the channel color.

To prevent the bar appeared on the screen from re-appearing, click on Marker button one more time. This is a common function that applies to both online trends and logging trends.





12) Logging Trend (Logging Time Range)

Input logging conditions (period and sampling conditions) to search past data. The logging search will be further explained later. This function is available only for logging trends.

13) 1 Hour Logging

View 1 hour logging data including the current time. For example, if it is 10:20, the logging data between 10 and 11o'clock are viewed. This function is available only for logging trends.

14) 2 Hour Logging

View logging data in the time range where the current time belongs to by dividing 24 hours into 0~2 o'clock, 2~4 o'clock, 4~6 o'clock, 20~22 o'clock and 22~midnight. For example, if it is 10:20, the logging data between 10 o'clock and noon are viewed. This function is available only for logging trends.

15) 3 Hour Logging

View logging data in the time range where the current time belongs to by dividing 24 hours into 0~3 o'clock, 3~6o'clock, 6~9o'clock,18~21 o'clock and 21~midnight. For example, if it is 10:20, the logging data between 9 o'clock and noon are viewed. This function is available only for logging trends.

16) 6 Hour Logging

View logging data in the time range where the current time belongs to by dividing 24 hours into 0~6 o'clock, 6~12o'clock, 12~18o'clock and 18~midnight For example, if it is 10:20, the logging data between 6 o'clock and noon are viewed. This function is available only for logging trends.

17) 12 Hour Logging

View logging data in the time range where the current time belongs to by dividing 24 hours into 0~noon and noon~ midnight. For example, if it is 10:20, the logging data between 0 o'clock and noon are viewed.

This function is available only for logging trends.

18) 1 Day Logging

View today's logging data. This function is available only for logging trends.

19) 1 Week Logging

View this week's logging data. The period is from Monday through the midnight of Sunday. This function is available only for logging trends.

20) 1 Month Logging

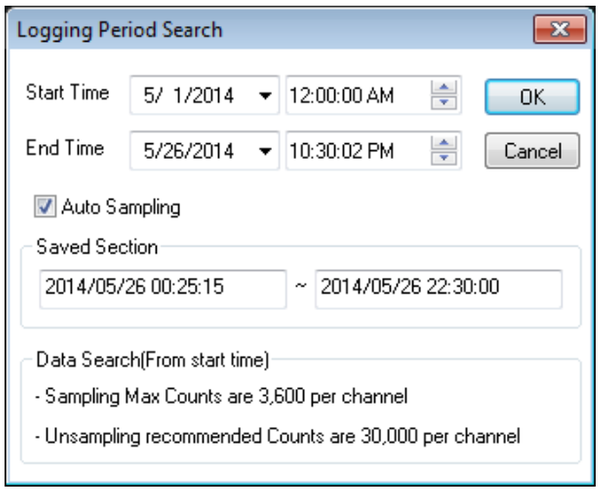
View this month's logging data This function is available only for logging trends.

21) Load Backup File

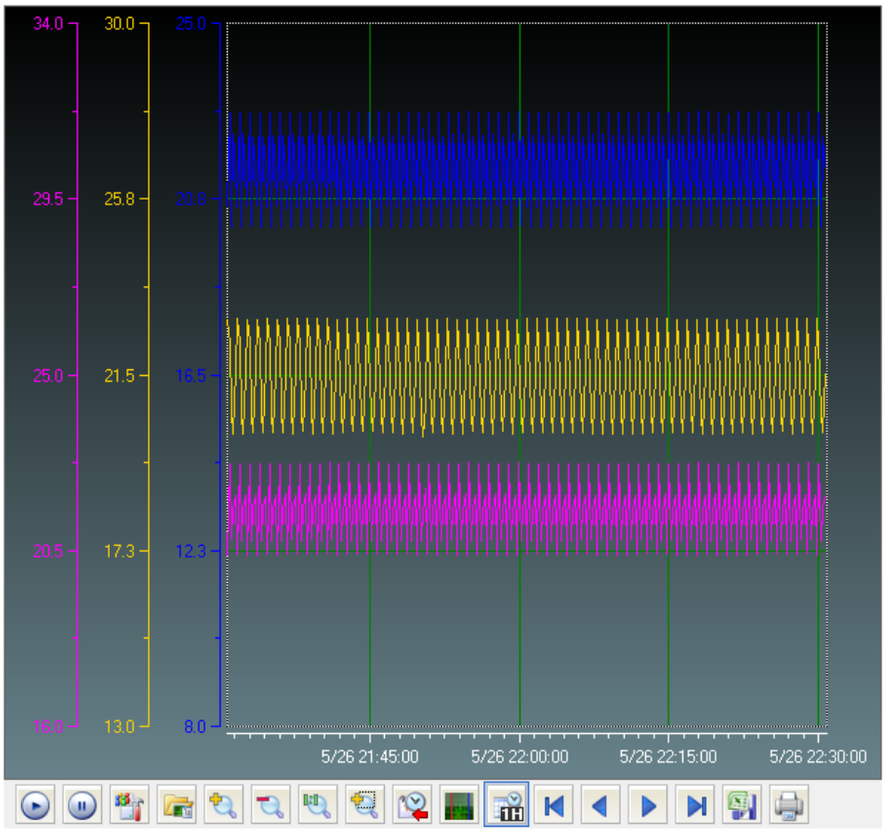
The trend backup files saved on the server or other storage devices can be imported and viewed in a list. Detailed functions on Load Backup File are described later.

11.3.4 Logging Search

This is a function to select logging search conditions. The following Logging Period Search window is displayed upon the click on Logging Search button.



The saved section indicates the start and end time of saving the whole data. You can determine the start and end time referring to this. Trend can express the data values up to 3600 and if it exceeds the limit, the data will be displayed through the automatic sampling.



※ Difference between logging search and online search

For online search, the mouse can be moved along the axis. Y axis can be selected with the mouse and moving is available with mouse wheel, mouse drag or keyboard arrow. For logging search, any axis can not be moved and only data in the search range are displayed.

11.3.5 Others

(1) Save as Excel File

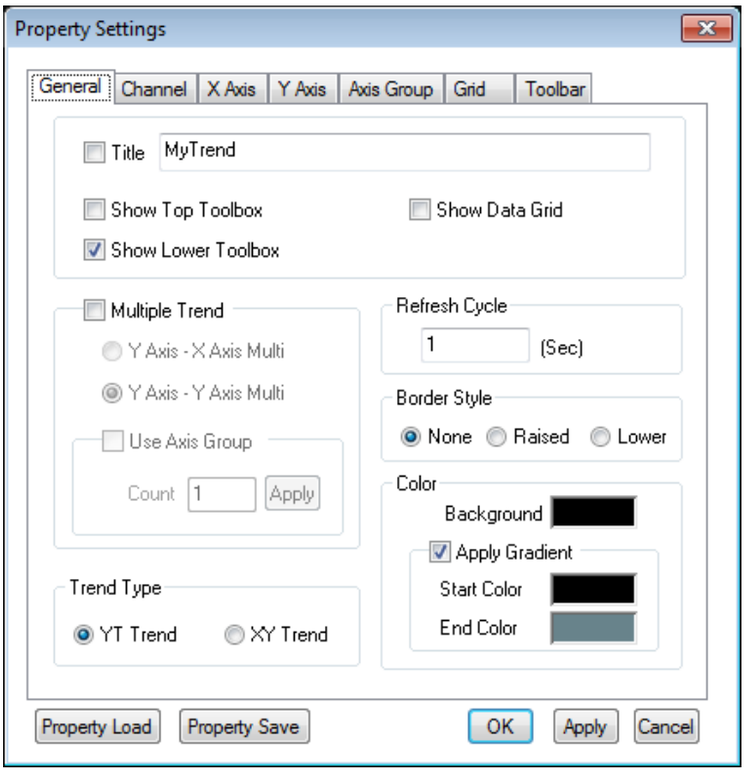
'Save as Excel File' is a function to save data on the current screen in a csv files.

(2) Print

The current trend screen is printed. Basically, the screen is changed from black to white when printing.

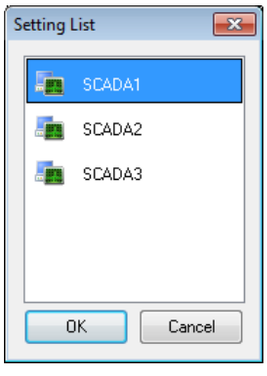
(3) Properties Load/Save

You can import and save the properties applied to the trend in the format of a special file. They can be applied to the graphic editor/runtime. You can also individually import the saved file using the script.



1) Property Load

In the Property Settings window, if you click the toolbar button of 'Property Load' or 'Import Settings', the saved trend property file will be displayed in the format of a list as below. If a user selects the desired property, it will be applied to the trend.



11.3.6 How to Use Script

(1) Function List

Function Name	Function Declaration
AddChannel_Archive	long AddChannel_Archive(long channelNo, LPCTSTR modelName, LPCTSTR tagName, double minValY, double maxValY)
AddChannel_ArchiveEx	long AddChannel_ArchiveEx(long channelNo, BSTR modelName, BSTR tagName, double minValY, double maxValY, long LineWidth,

	long LineColor, long LineStyle, long LineConnection, BSTR Labels)
AddChannel_RealTime	long AddChannel_RealTime(long channelNo, LPCTSTR tagName, double minValY, double maxValY)
AddChannel_RealTimeEx	long AddChannel_RealTimeEx(long channelNo, BSTR tagName, double minValY, double maxValY, long LineWidth, long LineColor, long LineStyle, long LineConnection, BSTR Labels)
AddTrendChannel	long AddTrendChannel(LPCTSTR dataSource, long channelNo, long modelID, LPCTSTR modelName, long tagID, LPCTSTR tagName, long minValueY, long maxValueY)
ApplyTrend	void ApplyTrend(void)
ApplyProperty	long ApplyProperty(VARIANT_BOOL bFlag)
DeleteTrendChannel	long DeleteTrendChannel(LPCTSTR channelStr, long channelNo)
GoFirstTrend	long GoFirstTrend(void)
GoLastTrend	long GoLastTrend(void)
GoNextTrend	long GoNextTrend(void)
GoPreviousTrend	long GoPreviousTrend(void)
Loadhistory12h	void Loadhistory12h (void)
Loadhistory1day	void Loadhistory1day (void)
Loadhistory1h	void Loadhistory1h (void)
Loadhistory1month	void Loadhistory1month (void)
Loadhistory1week	void Loadhistory1week (void)
Loadhistory2h	void Loadhistory2h (void)
Loadhistory3h	void Loadhistory3h (void)
Loadhistory6h	void Loadhistory6h (void)
LoadHistoryTrend	void LoadHistoryTrend(long year, long month, long week, long day, long hour, BOOL bFlag)
LoadPeriodDataStartTrend	void LoadPeriodDataStartTrend(BSTR strFrom, BSTR strTo)
LoadPeriodTrend	long LoadPeriodTrend(long days, BOOL bFlag)
LoadProperty	void LoadProperty(BSTR FileName)
Load_ArchivePeriod_Trend	long Load_ArchivePeriod_Trend(long type, long days, VARIANT_BOOL bFlag)
LoadPeriodDataTrend	long LoadPeriodDataTrend(LPCTSTR strFrom, LPCTSTR strTo, long bSample, LPCTSTR bFlag)
PauseTrend	void PauseTrend(void)
PrintTrend	void PrintTrend(void)
SaveProperty	void SaveProperty(BSTR FileName)
SetAlarmLimit	long SetAlarmLimit(long chanNo, VARIANT_BOOL bFlag, BSTR reserved)
SetAutoScale	long SetAutoScale(long chanNo, VARIANT_BOOL bFlag, BSTR reserved)
SetAxisXStartSpan	void SetAxisXStartSpan(VARIANT_BOOL bStartFromNowTime, long XSpan)

SetChannelMinMax	long SetChannelMinMax(long channelNo, double minValue,double maxValue)
SetTrendGridVisible	long SetTrendGridVisible(long isVisible)
SettSaveasexcel	void SettSaveasexcel(void)
SetZoom100	void SetZoom100 (void)
SetZoomIn	void SetZoomIn(void)
SetZoomOut	void SetZoomOut (void)
SetZoomRect	void SetZoomRect (void)
ShowMarkerValue	long ShowMarkerValue(void)
ShowSettingLoad	void ShowSettingLoad(void);
ShowTrendProperty	void ShowTrendProperty(void)
StartTrend	void StartTrend(void)

(2) Function Description

AddChannel_Archive Function	
Description	This function is to add a channel in a logging group name or a tag name to the trend if the tag is a logging trend.
Syntax	long AddChannel_Archive(long channelNo, LPCTSTR modelName, LPCTSTR tagName, double minValY, double maxValY) channelNo: Enter LONG-type numbers as the number of the channel to be added. modelName: Enter the logging group name of the logging tag to be added. This is a string value. tagName: Enter the name of the logging tag to be added. This is a string value. minValY: Enter DOUBLE-type numbers as the tag's minimum value. maxValY: Enter DOUBLE-type numbers as the tag's maximum value. If minValY and maxValY is -999, the minimum value and maximum value of DB will be automatically set; if minValY and maxValY is 0, it will be set as an automatic scale.
Reference	To add three channels to the Trend AddChannel_Archive(1, "SCADA_SEC", "SCADA.UT_EHV02.VRS", 0, 400) AddChannel_Archive(2, "SCADA_SEC", "SCADA.UT_EHV02.VST", 0, 400) AddChannel_Archive(3, "SCADA_SEC", "SCADA.UT_EHV02.VTR", 0, 400)

AddChannel_ArchiveEx Function	
Description	The history trend is the function to add channels to the trend with the model name and tag name.
Syntax	long AddChannel_ArchiveEx(long channelNo, LPCTSTR modelName, LPCTSTR tagName, double minValY, double maxValY, long LineWidth, long LineColor, long LineStyle, long LineConnection, BSTR Labels) channelNo: Enter the number of LONG type for the channel No. to add.

	modelName: Enter the model name of the archive tag to add. It is the string value. tagName: Enter the tag name of the archive tag to add. It is the string value. minValY: Enter the number of DOUBLE type for the tag's minimum value. maxValY: Enter the number of DOUBLE type for the tag's maximum value. LineWidth: Enter the number of LONG type for the line thickness of the channel to add. LineColor: Enter the number of LONG type for the line color of the channel to add. LineStyle: Enter the number of LONG type for the line style of the channel to add. LineConnection: Enter the number of LONG type for the line connection style of the channel to add. If minValY and maxValY is -999, the minimum value and maximum value of DB will be automatically set; if minValY and maxValY is 0, it will be set as an automatic scale.
Reference	To add three channels to the trend AddChannel_ArchiveEx(1, "SCADA_SEC", "SCADA.UT_EHV02.VRS", 0, 400, 2, RGB(255,0,0), 0, 1) AddChannel_ArchiveEx(2, "SCADA_SEC", "SCADA.UT_EHV02.VST", 0, 400, 400 ,2, RGB(255,255,0), 0, 1) AddChannel_ArchiveEx(3, "SCADA_SEC", "SCADA.UT_EHV02.VTR", 0, 400, 400, 2, RGB(0,0,0), 0, 1)

AddChannel_RealTime Function	
Description	This function is to add a channel in a tag name to the trend if the trend is an online tag. But, in most cases, online monitoring is possible with logging tags, therefore, AddChannel_Archive is more widely used than this function.
Syntax	long AddChannel_RealTime(long channelNo, BSTR tagName, double minValY, double maxValY) channelNo: Enter LONG-type numbers as the number of the channel to be added. tagName: Enter the name of the online tag to be added. This is a string value. minValY: Enter DOUBLE-type numbers as the tag's minimum value. maxValY: Enter DOUBLE-type numbers as the tag's maximum value. If minValY and maxValY is -999, the minimum value and maximum value of DB will be automatically set; if minValY and maxValY is 0, it will be set as an automatic scale.
Reference	To add three channels to the trend AddChannel_RealTime (1, "SCADA.UT_EHV02.VRS", 0, 400) AddChannel_RealTime (2, "SCADA.UT_EHV02.VST", 0, 400) AddChannel_RealTime (3, "SCADA.UT_EHV02.VTR", 0, 400)

AddChannel_RealTimeEx Function	
Description	It is the function to add the channel to the trend with the tag name in the real-time trend. However, in most cases, real-time monitoring can be performed through the archive tag so the AddChannel_Archive function is used more widely than this function.
Syntax	long AddChannel_RealTimeEx(long channelNo, BSTR tagName, double minValY, double maxValY, long LineWidth, long LineColor, long LineStyle, long LineConnection, BSTR Labels) channelNo: Enter the number of LONG type for the channel No. to add. tagName: Enter the tag name of the real-time tag to add. It is the string value. minValY: Enter the number of DOUBLE type for the tag's minimum value. maxValY: Enter the number of DOUBLE type for the tag's maximum value. LineWidth: Enter the number of LONG type for the line thickness of the channel to add. LineColor: Enter the number of LONG type for the line color of the channel to add. LineStyle: Enter the number of LONG type for the line style of the channel to add. LineConnection: Enter the number of LONG type for the line connection style of the channel to add. If minValY and maxValY is -999, the minimum value and maximum value of DB will be automatically set; if minValY and maxValY is 0, it will be set as an automatic scale.
Reference	To add three channels to the trend AddChannel_RealTimeEx (1, "SCADA.UT_EHV02.VRS", 0, 400, 2, RGB(255,0,0), 0, 1) AddChannel_RealTimeEx (2, "SCADA.UT_EHV02.VST", 0, 400, 2, RGB(255,255,0), 0, 1) AddChannel_RealTimeEx (3, "SCADA.UT_EHV02.VTR", 0, 400, 2, RGB(0,0,0), 0, 1)

AddTrendChannel Function	
Description	This function is to add a channel to the trend. But, this function is rarely used because the user needs to input ID information on the logging group and tag as parameters. It is recommended to use AddChannel_Archive if the logging group ID or tag ID is unknown.
Syntax	long AddTrendChannel(LPCTSTR dataSource, long channelNo, long modelID, LPCTSTR modelName, long tagID, LPCTSTR tagName, long minValueY, long maxValueY) dataSource: Enter the following string values to identify whether the tag to be added is a logging tag or an online tag. - Logging Tag: "Archive" - Online Tag: "OnLine" channelNo: Enter LONG-type numbers as the number of the channel to be added. modelID: Enter LONG- type numbers as the logging group ID of the logging tag to be added. modelName: Enter the logging group name of the logging tag to be added. This is a string value. tagID: Enter LONG- type numbers as the tag ID of the logging tag to be added.

	tagName: Enter the tag name of the online tag to be added. minValY: Enter DOUBLE-type numbers as the tag's minimum value. maxValY: Enter DOUBLE-type numbers as the tag's maximu value. If minValY and maxValY is -999, the minimum value and maximum value of DB will be automatically set; if minValY and maxValY is 0, it will be set as an automatic scale.
Reference	To add three channels to the Trend AddTrendChannel("Archive", 1, 9, "INV1_SEC", 2710, "SOLAR.INV_1.1_AI_CT", 0, 327) AddTrendChannel("Archive", 2, 9, "INV1_SEC", 2714, "SOLAR.INV_1.1_AI_VO", 0, 450) AddTrendChannel("Archive", 3, 9, "INV1_SEC", 2713, "SOLAR.INV_1.1_AI_PW", 0, 60)

ApplyTrend Function	
Description	This function is to make the monitoring start after adding a channel to the trend.
Syntax	void ApplyTrend(void)
Reference	ApplyTrend ()

ApplyProperty Function	
Description	This function is to apply the trend properties to the trend after changing them.
Syntax	long ApplyProperty(VARIANT_BOOL bFlag) bFlag: Input TRUE
Reference	ApplyProperty(TRUE)

DeleteTrendChannel Function	
Description	This function is to delete a certain channel or the entire channels from the trend.
Syntax	long DeleteTrendChannel(LPCTSTR channelStr, long channelNo) channelStr: Enter the channel name to be deleted. channelNo: Enter the channel number to be deleted. But, when deleting the entire channels, enter empty character ("") in channelStr and '0' in channelNo.
Reference	To delete 'Channel 3' from the trend (in case that the channel name is known) DeleteTrendChannel ("Channel3", 3) To delete 'Channel 3' from the trend (in case that the channel name is unknown) DeleteTrendChannel ("", 3) The above two methods are used because there are same tags in channels. To delete the entire channels from the trend DeleteTrendChannel ("", 0)

GoFirstTrend Function	
Description	It is the function to retrieve sectional history for the next setting time based on the start of the first logging time.
Syntax	long GoFirstTrend(void)

GoLastTrend Function	
Description	It is the function to retrieve sectional history for the previous time based on the latest logging time.
Syntax	long GoLastTrend(void)

GoNextTrend Function	
Description	It is the function to retrieve sectional history for the next time based on the current logging time.
Syntax	long GoNextTrend(void)

GoPreviousTrend Function	
Description	It is the function to retrieve sectional history for the previous time based on the current logging time.
Syntax	long GoPreviousTrend(void)

Loadhistory12h Function	
Description	It is the function to retrieve the history for the section of previous 12 hours based on the current logging time.
Syntax	void Loadhistory12h (void)

Loadhistory1day Function	
Description	It is the function to retrieve the history for the section of previous day based on the current logging time.
Syntax	void Loadhistory1day (void)

Loadhistory1h Function	
Description	It is the function to retrieve the history for the section of previous an hour based on the current logging time.
Syntax	void Loadhistory1h (void)

Loadhistory1month Function	
Description	It is the function to retrieve the history for the section of previous a month based on the current logging time.
Syntax	void Loadhistory1month (void)

Loadhistory1week Function	
Description	It is the function to retrieve the history for the section of previous a week based on the current logging time.
Syntax	void Loadhistory1week (void)

Loadhistory2h Function	
Description	It is the function to retrieve the history for the section of previous two hours based on the current logging time.
Syntax	void Loadhistory2h (void)

Loadhistory3h Function	
Description	It is the function to retrieve the history for the section of previous three hours based on the current logging time.
Syntax	void Loadhistory3h (void)

Loadhistory6h Function	
Description	It is the function to retrieve the history for the section of previous six hours based on the current logging time.
Syntax	void Loadhistory6h (void)

LoadHistoryTrend Function	
Description	This function is to search logging data in a designated time zone. Enter year, month, week, day, hour to set the start and end time of a certain time zone and search logging data in the time zone. Enter a number in year, moth, week and day and enter 0 in the rest. Make bFlag TRUE to search data on a specific day/month/year or make it FALSE to search them as of the current time.
Syntax	void LoadHistoryTrend(long year, long month, long week, long day, long hour, BOOL bFlag) year: Enter the year to search in LONG-type numbers. month: Enter the month to search in LONG-type numbers. week: To search data in a certain week, enter 1. If not, enter 0.

	day: Enter the day to search in LONG-type numbers. hour: Enter the hour to search in LONG-type numbers. bFlag: To search data on a specific day/month/year, enter TRUE. If not, enter FALSE.
Reference	To search logging data in the time zone in which a certain date belongs to Year Search - LoadHistoryTrend(2007, 0, 0, 0, 0, TRUE) Month Search - LoadHistoryTrend(2007, 5, 0, 0, 0, TRUE) Week Search - LoadHistoryTrend(2007, 5, 1, 0, 0, TRUE) Day Search - LoadHistoryTrend(2007, 5, 0, 13, 0, TRUE) Hour Search- LoadHistoryTrend(2007, 5, 0, 13, 10, TRUE) For One Hour Search as of the current time, LoadHistoryTrend(0, 0, 0, 0, 1, FALSE) For 2-Hour Search as of the current time, LoadHistoryTrend(0, 0, 0, 0, 2, FALSE) For 6-Hour Search as of the current time, LoadHistoryTrend(0, 0, 0, 0, 6, FALSE)

LoadPeriodTrend Function	
Description	As a function to search data in a certain period, it presents One Hour Search, One Day search, One Week Search and One Month Search as of the current time.
Syntax	long LoadPeriodTrend(long days, BOOL bFlag) days – Enter the following as parameters for searching. One Hour Search: 1 One Day search: 2 One Week Search: 3 One Month Search: 4 bFlag – Input TRUE..
Reference	For One Hour Search, LoadPeriodTrend (1, TRUE) For One Day Search, LoadPeriodTrend (2, TRUE) For One Week Search, LoadPeriodTrend (3, TRUE) For One Month Search, LoadPeriodTrend (4, TRUE)

Load_ArchivePeriod_Trend Function	
Description	As a function to search data in a certain period, it presents Hour Search, One Day search, One Week Search and One Month Search as of the current time. For Hour Search, 1, 2, 3, 6 and 12 hour search are available.
Syntax	long Load_ArchivePeriod_Trend (long type, long days, VARIANT_BOOL bFlag) type – Enter the following as period parameters for searching. One Hour Search: 1 One Day Search: 2 One Week Search: 3 One Month Search: 4 days – This parameter is applied only when One Hour Search is entered for type. Enter 1, 2, 3, 6 or 12 to search 1, 2, 3, 6 or 12 hour search. bFlag – Input TRUE.
Reference	For One Hour Search after selecting Hour Search, Load_ArchivePeriod_Trend(1, 1, TRUE) For One Day Search, Load_ArchivePeriod_Trend(2, 0, TRUE) For One Week Search, Load_ArchivePeriod_Trend(3, 0, TRUE) For One Month Search, Load_ArchivePeriod_Trend(4, 0, TRUE)

LoadPeriodDataStartTrend Function	
Description	It is the function to look up the section. You just need to enter the start and end time to look up.
Syntax	void LoadPeriodDataStartTrend ()
Reference	To search data from 13 o'clock on February 11, 2008 to 14 o'clock on February 11 LoadPeriodDataStartTrend ("2008/02/11 13:00:00", "2008/02/11 14:00:00")

LoadPeriodDataTrend Function	
Description	As a function to search data in a certain period, if the user inputs the Start time and End time, this function searches logging data during the period.
Syntax	long LoadPeriodDataTrend(LPCTSTR strFrom, LPCTSTR strTo, long bSample, LPCTSTR bFlag)

	strFrom – Enter the following type as the Start time. Number 0 through 24 are used for Time Display. "2008/02/14 11:00:00" strTo – Enter the following type as the End time. Number 0 through 24 are used for Time Display. "2008/02/14 11:10:00" bSample – Decide whether to display data after applying sampling. Enter 0 as the default value. bFlag – This is a reserve parameter variable. Enter"".
Reference	To search data from 13 o'clock on February 11, 2008 to 14 o'clock on February 11, LoadPeriodDataTrend("2008/02/11 13:00:00", "2008/02/11 14:00:00", 0, "") Remember that there is empty space between day and time.

LoadProperty Function	
Description	It is the function to import the saved property file.
Syntax	void LoadProperty (BSTR FileName)
Reference	LoadProperty ("file name")

PauseTrend Function	
Description	It is the function to stop the real-time lookup.
Syntax	void PauseTrend(void)

PrintTrend Function	
Description	It is the function to print out the current trend screen
Syntax	void PrintTrend(void)

SaveProperty Function	
Description	It is the function to save the set properties as a file.
Syntax	void SaveProperty (BSTR FileName)
Reference	SaveProperty ("file name")

SetAlarmLimit Function	
Description	It is the function to determine whether applying the alarm limit value to the channel to be specified.
Syntax	long SetAlarmLimit(long chanNo, VARIANT_BOOL bFlag, BSTR reserved)
Reference	SetAlarmLimit (1,True, "")

SetAutoScale Function	
Description	It is the function to determine whether applying the automatic scale to the channel to be specified.
Syntax	long SetAutoScale(long chanNo, VARIANT_BOOL bFlag, BSTR reserved)
Reference	SetAutoScale (1,True, “”)

SetAxisXStarSpan Function	
Description	It is the function to determine whether setting the min. value of the X-axis as the current time with the interval.
Syntax	void SetAxisXStartSpan(VARIANT_BOOL bStartFromNowTime, long XSpan)
Reference	SetAxisXStartSpan (True,300)

SetChannelMinMax Function	
Description	It is the function to set the minimum and maximum value of the channel.
Syntax	long SetChannelMinMax(long channelNo, double minValue, double maxValue)

SettSaveasexcel Function	
Description	It is the function to show the dialog box where you can save the data displayed in the trend as the Excel file.
Syntax	void SettSaveasexcel(void)

SetZoom100 Function	
Description	It is the function to show the trend screen at the ratio of 100%.
Syntax	void SetZoom100 (void)

SetZoomIn Function	
Description	It is the function to zoom in the trend screen.
Syntax	void SetZoomIn(void)

SetZoomOut Function	
Description	It is the function to zoom out the trend screen.
Syntax	void SetZoomOut (void)

SetZoomRect Function	
Description	It is the function to zoom in the trend screen partially.
Syntax	void SetZoomRect (void)

ShowMarkerValue Function	
Description	This function is to show marker visible properties of the trend.
Syntax	void ShowMarkerValue()
Reference	ShowMarkerValue ()

ShowSettingLoad Function	
Description	It is the function to load the property.
Syntax	void ShowSettingLoad ()
Reference	ShowSettingLoad ()

ShowTrendProperty Function	
Description	This function is to show the property window of the trend.
Syntax	void ShowTrendProperty()
Reference	ShowTrendProperty(TRUE)

StartTrend Function	
Description	It is the function for real-time lookup based on the established screen cycle.
Syntax	void StartTrend(void)

11.4 List Trend Viewer

As a screen to monitor changes in tag values over time, this List Trend Viewer is an Active X module that displays online data as well as logging data stored in the logging server.

11.4.1 Prerequisite and Environment

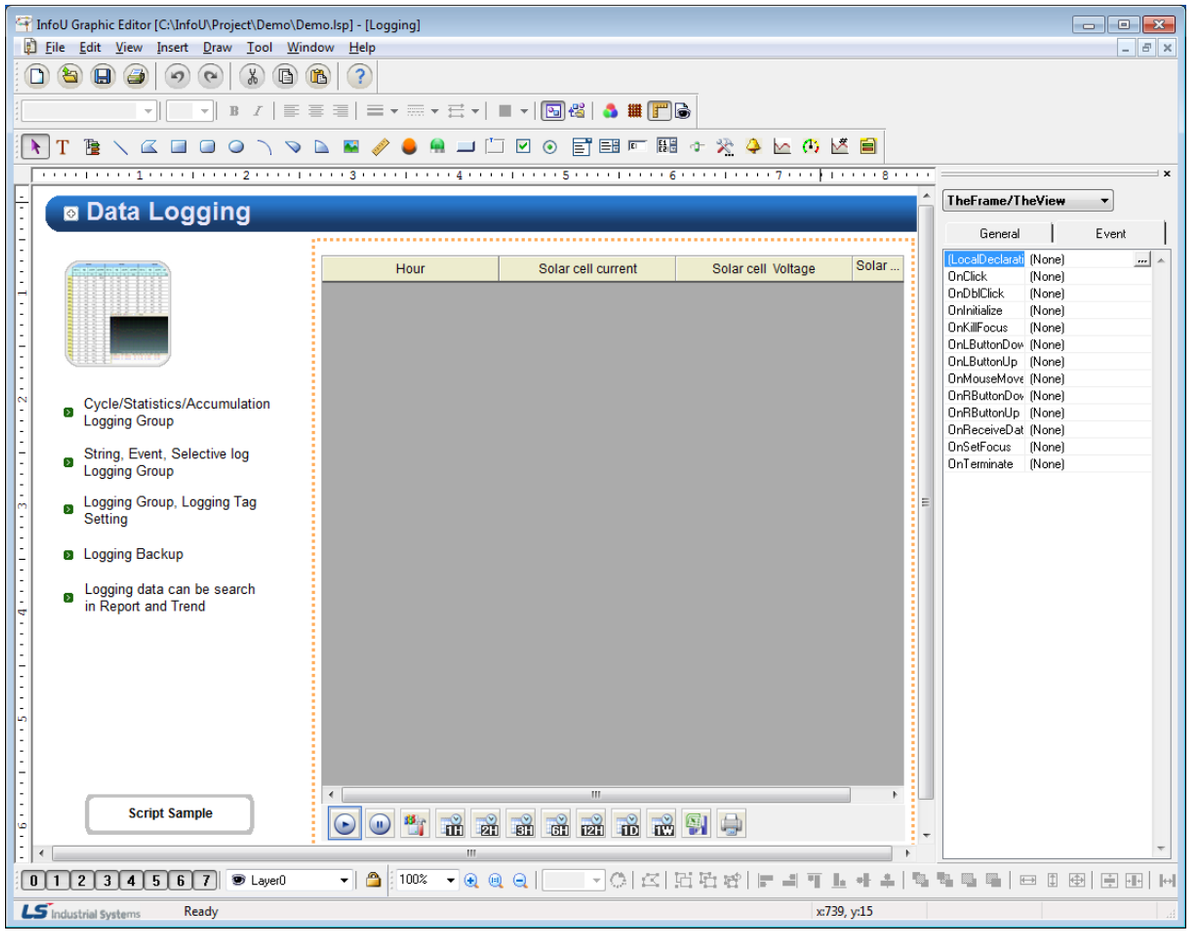
List trend properties can be set up only after the ActiveX module is inserted to the InfoU graphic editor. Channel assignment, tag assignment and other tag properties to be monitored are designated in advance as List Trend properties. Channels or tags are not basically assigned to the list trend status inserted in the graphic editor. It is essential to display the Property Window before setting up the property values needed to run the List Trend. After the property values are saved, the List Trend Monitoring Screen appears upon the opening of the screen under the Run time environment.

The engineering values are entered in the following order on the List Trend viewer. If  is selected from the tools, immediate inserting can be performed.

11.4.2 Screen Configuration

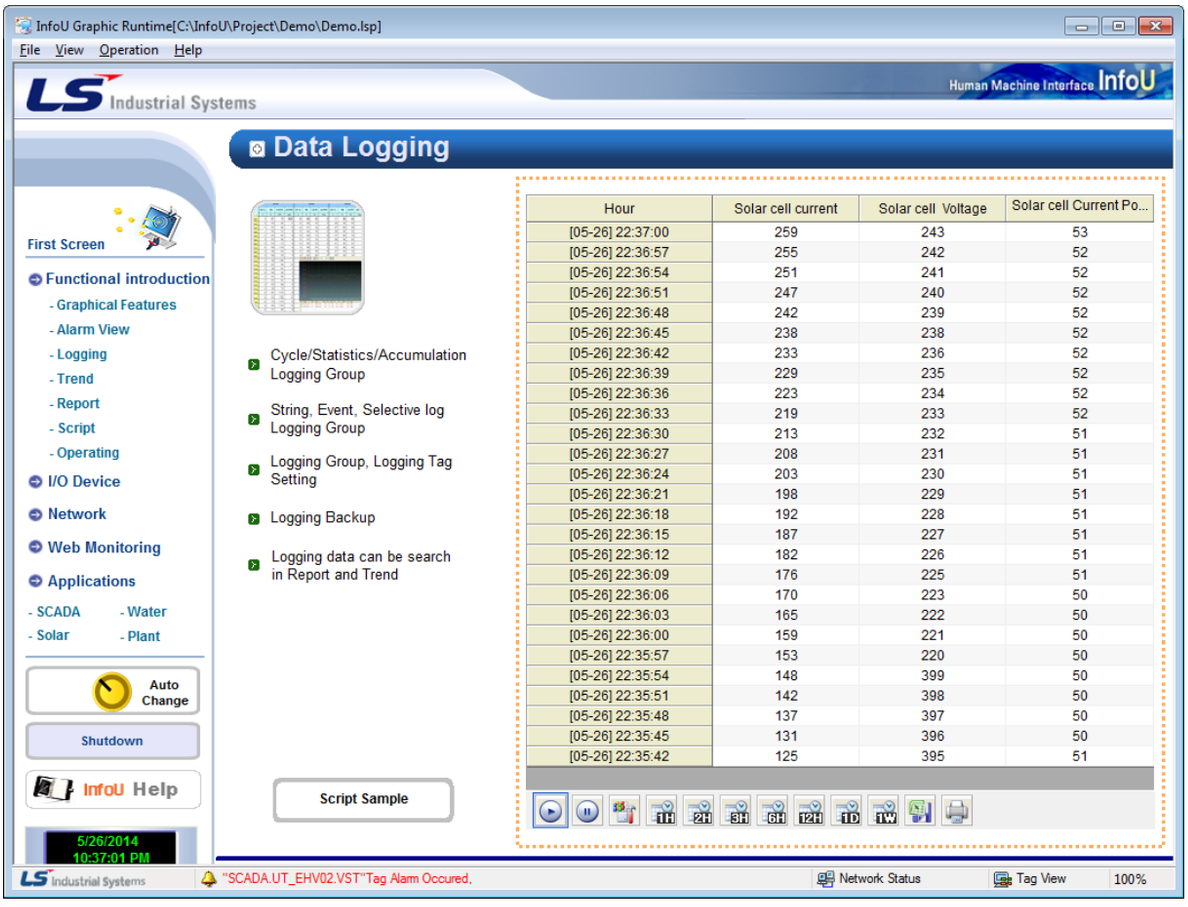
(1) Engineering Screen

If List Trend is inserted and a tag to be monitored is selected and then settings are all completed, the following screen appears. The screen in this figure shows that three tags are selected.



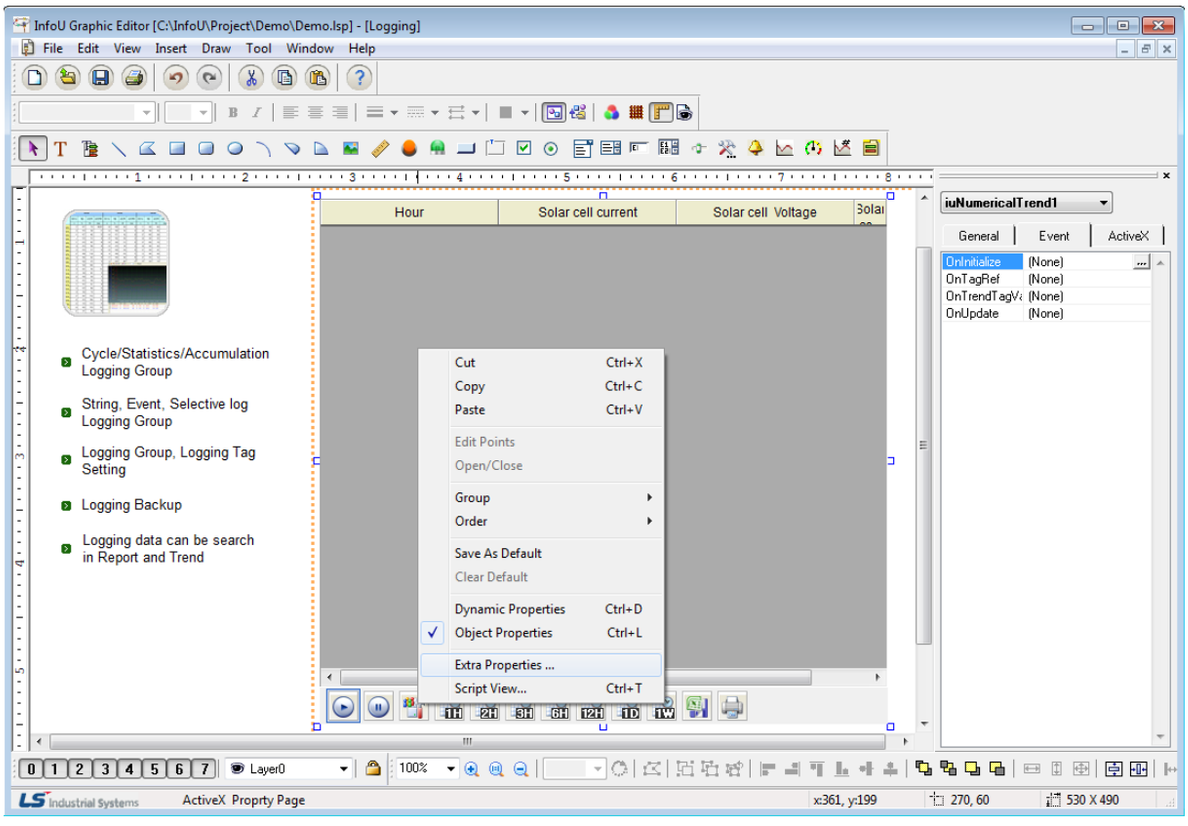
(2) Runtime Screen

If default list trend settings are completed on the engineering screen above and then, the runtime is executed, the screen shows that data values are changed over time.



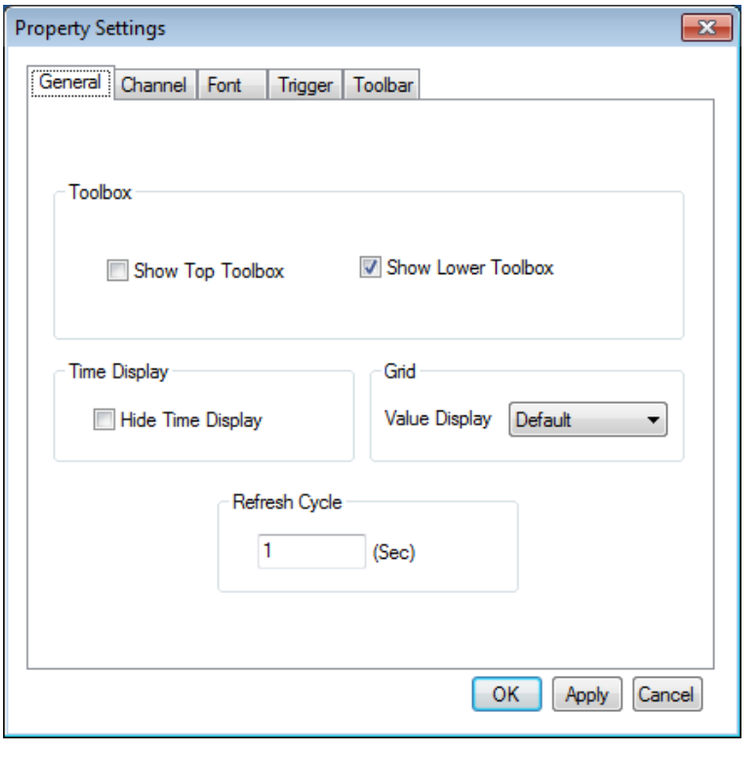
11.4.3 Settings

If the right side of the mouse is clicked on the List Trend and the 'Active X Properties' menu is selected, the setting window appears. The same setting screen is used for the engineering screen and the run-time screen. But, the runtime screen has Channel Setting function only. In addition, once property values are saved on the engineering screen, those values will be continuously valid however, if those property values are changed on the runtime screen, the changed values are applied only to the runtime screen. The changed values are cancelled if the runtime stops and the property values set up on the engineering screen are applied.



(1) General

The Engineering Screen and the Runtime Screen have the same general function.



1) Show Top Toolbox/Show Lower Toolbox

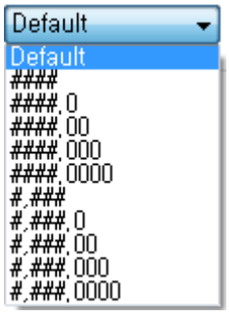
Set up whether to display Trend Toolbox on the top or the bottom during the running. Both can not be selected at the same time and it is allowed neither of them is selected. 'Show Lower Toolbox' has been selected as the default value.

2) Time Display

Select whether to show/hide Time Display column on the screen. 'Hide Time Display'' has not been selected as the default value.

3) Grid (Value Display)

The user may select a method to display grid value.



Default : Display the default vale.

: Do not display decimal points regardless of tag DB

####.0 : Indicate down to one decimal point regardless of tag DB.

####.00 : Indicate down to two decimal points regardless of tag DB.

####.000 : Indicate down to three decimal points regardless of tag DB.

####.0000: Indicate down to four decimal points regardless of tag DB.

#,### : Do not display decimal points regardless of tag DB and indicate digit with comma.

#,###.0 : Indicate down to one decimal point regardless of tag DB and indicate digit with comma

#,###.00 : Indicate down to two decimal points regardless of tag DB and indicate digit with comma.

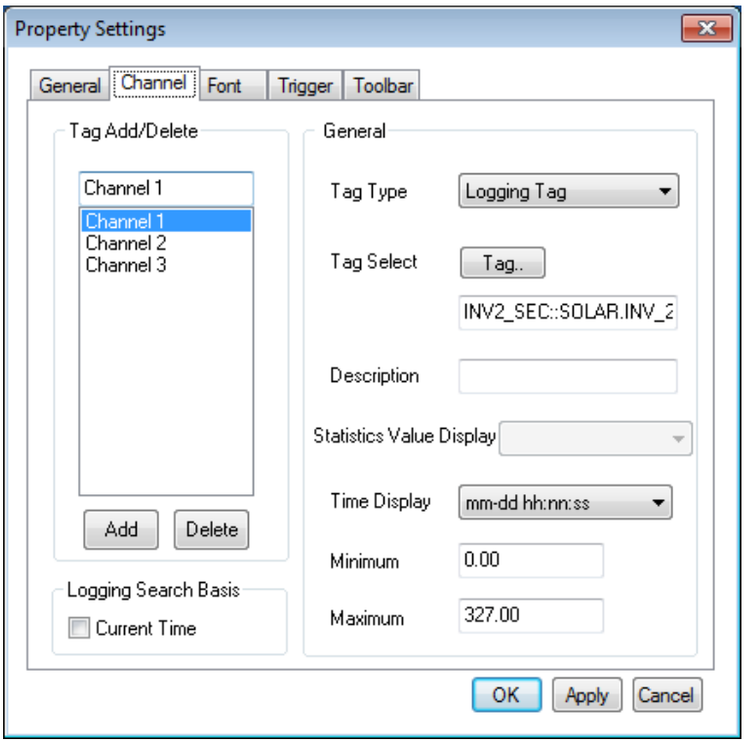
#,###.000 : Indicate down to three decimal points regardless of tag DB and indicate digit with comma.

#,###.0000: Indicate down to four decimal points regardless of tag DB and indicate digit with comma..

4) Refresh Cycle

Designated a refresh cycle. One second has been set up as the default value.

(2) Channel



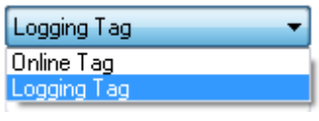
The Engineering Screen and the Runtime Screen have the same Channel functions. Up to 20 channels are provided in the List Trend. There is at least one channel and it is available to designate tags for each channel. If one channel is an online tag, the rest channels have to be online tags. Likely, if one channel is a logging tag, the rest channels have to be logging tags. Also, if any logging data needs to be saved or searched, the logging tag should be selected and used. Online tags do not provide the logging function.

1) Channel Add/Delete

A channel is added/deleted. At least one or more than one but not more than 20 channels can be selected.

2) Tag type

Select either Logging Tag or Online Tag.



3) Tag Select

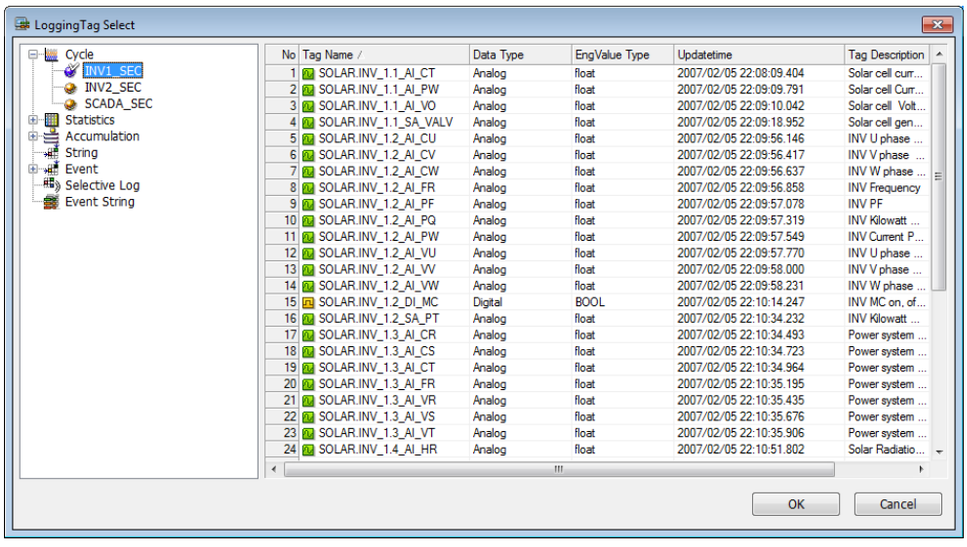
Assign a tag to the channel. One tag can be assigned to one channel.

If a logging tag is selected, the logging tag browser where the associated logging group is set up is called while the online tag browser is called for an online tag.

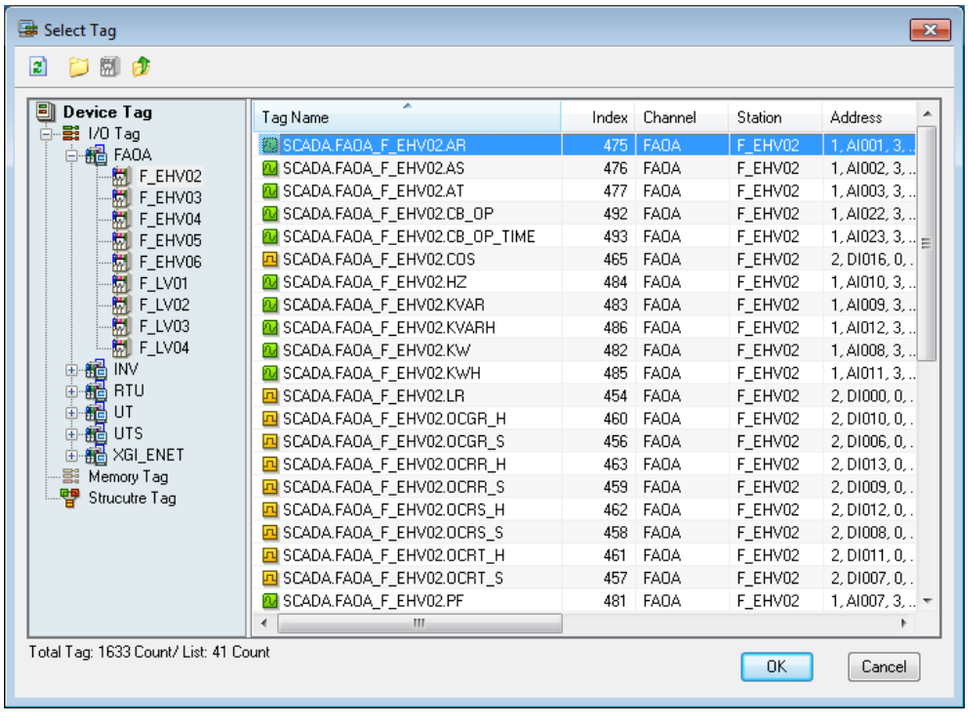
Be cautious that logging data can be searched only when a logging tag is selected.

If a tag is selected, the tag's minimum value and maximum values are displayed. These min/max values apply to Y-axis values for Y axis settings.

* 'Logging Tag Select' screen



* 'Online Tag Select' screen



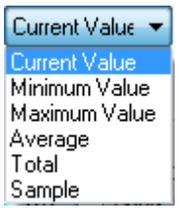
4) Description

Describe the selected tag. If any description has already been written, this description is displayed instead of the tag name on the grid. If not, the tag name is displayed. If any tag description has been recorded in the tag DB, that description is displayed even the user skips this item.

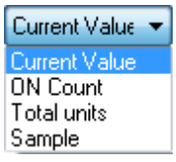
5) Statistics Value Display

This function becomes active only when a statistics tag is selected. For analog tags, select one value among current value, min value, max value, average value, sum and sample and for digital tags, select one value among current value, On count, On Time, Total count and sample. If the tag is not a statistics tag, it becomes inactive.

* Analog Tag



* Digital Tag



6) Time Display

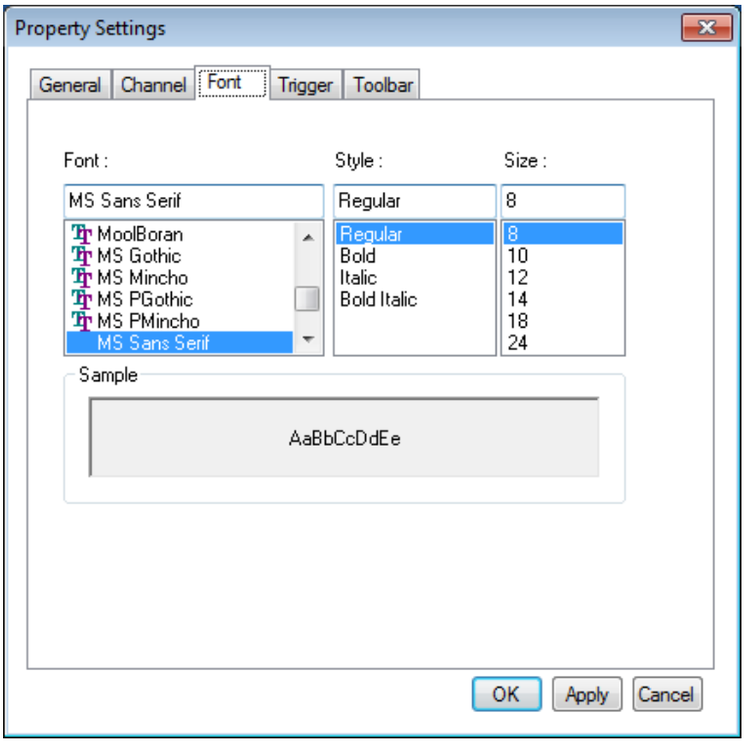
Select a time type to display on the grid.



7) History retrieval

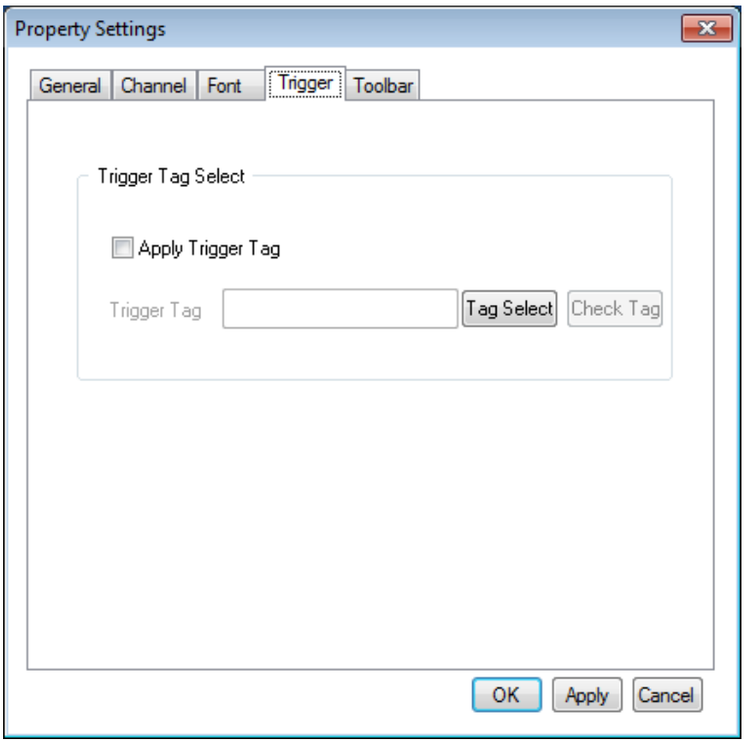
In the case of Trend, basically, history retrieval is done at a distance of regular time. When selecting the current time, history retrieval will be performed based on the current time.

(3) Font



Decide Font, Style and Size to be displayed on the list trend viewer list.

(4) Trigger



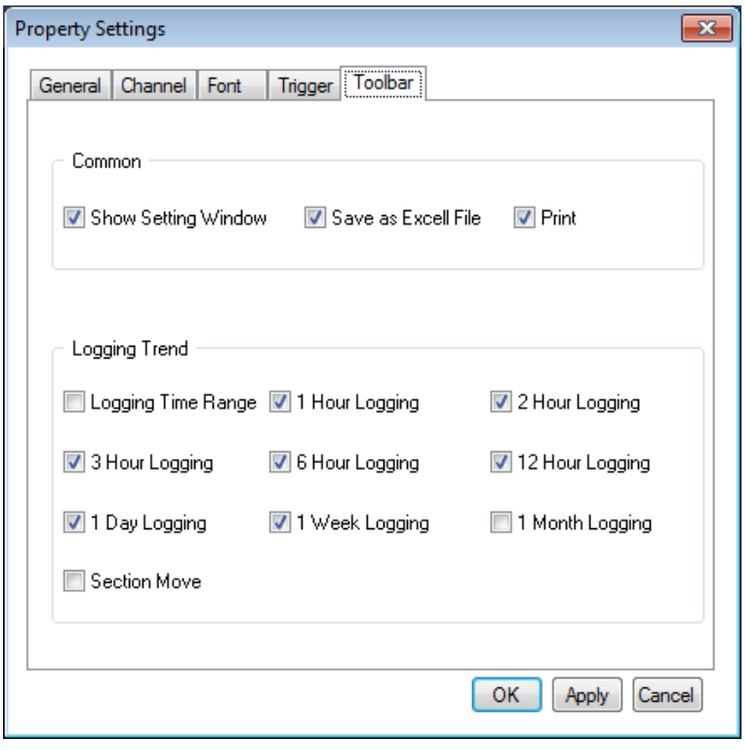
This Trigger is a function used to run the list trend only when the base tag for trigger is True.

If 'Apply Trigger Tag' is clicked, the trigger tag item becomes active. Press 'Tag Select' to select a tag. At this time, be cautious that the tag should be corresponding to the tag kind designated during the channel settings. Usually, they are corresponding to each other and the Tag Selection Window is displayed but, if Tag Kind is changed latter, trigger setting items shall be checked.

If the value of a trigger tag is O, that is, if it is False, the List Trend does not show any tag value. It shows tag values only the value is not O.

(5) Toolbar

Select toolbar (menus).



1) Start ( )

This is a common function that applies to both online trends and logging trends,

Once trend monitoring starts, data is renewed every second and 5 minutes is set as the time for X axis.

To view the current screen again during the logging search, press this button and the current time screen is displayed.

This Start button can not be selected as the default value.

2) Stop ( )

Trend Monitoring stops. This function is available only for logging trends. That is, this function exists only when logging tags are selected and online trends do not have this function.

This Stop button can not be selected as the default value like the Start button.

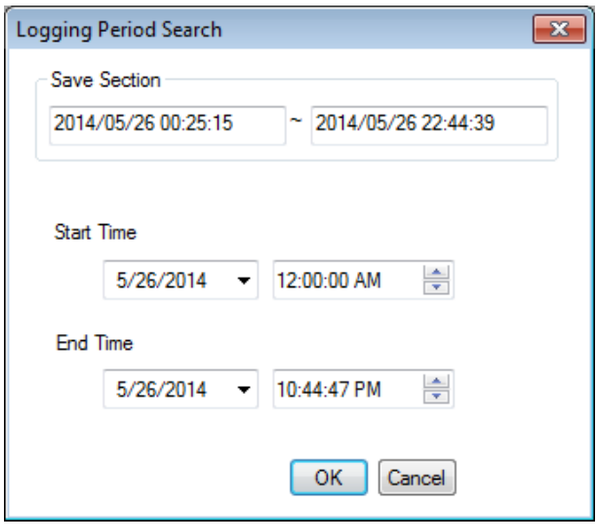
3) Show Setting Window ( )

Setting values of the current trend can be changed. This is a common function that applies to both online trends and logging trends,

This Setting Window has been described before.

4) Logging Trend (Logging Time Range) ( )

Input logging conditions (period and sampling conditions) to search past data. Click on the relevant icon to display the following window.



'Saved Section' shows the starting time and the end time to save the entire data and this is referred when deciding the start time and the end time. The following figure shows that 'One Hour Search' is selected and executed.

Hour	Solar cell current	Solar cell Voltage	Solar cell Current Power
[05-26] 22:44:45	205	230	51
[05-26] 22:44:42	200	229	51
[05-26] 22:44:39	194	228	51
[05-26] 22:44:36	189	227	51
[05-26] 22:44:33	183	226	51
[05-26] 22:44:30	178	225	51
[05-26] 22:44:27	172	224	50
[05-26] 22:44:24	167	223	50
[05-26] 22:44:21	161	222	50
[05-26] 22:44:18	155	221	50
[05-26] 22:44:15	150	220	50
[05-26] 22:44:12	144	398	50
[05-26] 22:44:09	139	397	50
[05-26] 22:44:06	133	396	50
[05-26] 22:44:03	127	395	50
[05-26] 22:44:00	121	394	51
[05-26] 22:43:57	116	393	51
[05-26] 22:43:54	111	392	51
[05-26] 22:43:51	105	391	51
[05-26] 22:43:48	100	390	51
[05-26] 22:43:45	94	389	51
[05-26] 22:43:42	89	388	51
[05-26] 22:43:39	84	387	51
[05-26] 22:43:36	79	386	52
[05-26] 22:43:33	74	384	52
[05-26] 22:43:30	69	383	52
[05-26] 22:43:27	64	382	52
[05-26] 22:43:24	60	381	52
[05-26] 22:43:21	55	380	52
[05-26] 22:43:18	51	379	52
[05-26] 22:43:15	47	378	52
[05-26] 22:43:12	43	377	53

This function is available only for logging trends.

5) 1 Hour Logging ( )

View 1 hour logging data including the current time. For example, if it is 10:20, the logging data between

10 and 11o'clock are viewed. This function is available only for logging trends.

6) 2 Hour Logging ()
View logging data in the time range where the current time belongs to by dividing 24 hours into 0~2 o'clock, 2~4 o'clock, 4~6 o'clock, 20~22 o'clock and 22~midnight. For example, if it is 10:20, the logging data between 10 o'clock and noon are viewed. This function is available only for logging trends.

7) 3 Hour Logging ()
View logging data in the time range where the current time belongs to by dividing 24 hours into 0~3 o'clock, 3~6o'clock, 6~9o'clock, 18~21 o'clock and 21~midnight. For example, if it is 10:20, the logging data between 9 o'clock and noon are viewed. This function is available only for logging trends.

8) 6 Hour Logging ()
View logging data in the time range where the current time belongs to by dividing 24 hours into 0~6 o'clock, 6~12o'clock, 12~18o'clock and 18~midnight For example, if it is 10:20, the logging data between 6 o'clock and noon are viewed. This function is available only for logging trends.

9) 12 Hour Logging ()
View logging data in the time range where the current time belongs to by dividing 24 hours into 0~noon and noon~ midnight. For example, if it is 10:20, the logging data between 0 o'clock and noon are viewed. This function is available only for logging trends.

10) 1 Day Logging ()
View today's logging data. This function is available only for logging trends.

11) 1 Week Logging ()
View this week's logging data. The period is from Monday through the midnight of Sunday. This function is available only for logging trends.

12) 1 Month Logging ()
View this month's logging data This function is available only for logging trends.

13) Section Search ()
When you applying the history retrieval, you can look up the section using the section retrieval function.
a. If you press the very left button, the screen will be moved to the start time stored in the current DB server.
b. If you press the second button, the screen will be moved to the previous section as far as the lookup distance from the current section. It can be movable to the very beginning.
c. If you press the third button, the screen will be moved to the next section as far as the lookup distance from the current section. It can be movable to the very end.
d. If you press the very right button, the screen will be moved to the current time. The section retrieval function is available only when applying the history retrieval. The function is not available in the real-time monitoring status, available only in the history trend.

11.4.4 How to Use Script

(1) Function List

Function Name	Function Declaration
AddChannel_Archive	long AddChannel_Archive(long channelNo, LPCTSTR modelName, LPCTSTR tagName, LPCTSTR tagDesc)
AddChannel_RealTime	long AddChannel_RealTime(long channelNo, LPCTSTR tagName, LPCTSTR tagDesc)
AddTrendChannel	long AddTrendChannel(LPCTSTR dataSource, long channelNo, long modelID, LPCTSTR modelName, long tagID, LPCTSTR tagName, LPCTSTR tagDesc)
ApplyTrigger	void ApplyTrigger(bApply, modelName, tagName)
GoFirstTrend	long GoFirstTrend(void)
GoLastTrend	long GoLastTrend(void)
GoNextTrend	long GoNextTrend(void)
GoPreviousTrend	long GoPreviousTrend(void)
Loadhistory12h	void Loadhistory12h (void)
Loadhistory1day	void Loadhistory1day (void)
Loadhistory1h	void Loadhistory1h (void)
Loadhistory1month	void Loadhistory1month (void)
Loadhistory1week	void Loadhistory1week (void)
Loadhistory2h	void Loadhistory2h (void)
Loadhistory3h	void Loadhistory3h (void)
Loadhistory6h	void Loadhistory6h (void)
LoadHistoryTrend	void LoadHistoryTrend(long year, long month, long week, long day, long hour, BOOL bFlag)
LoadPeriodTrend	long LoadPeriodTrend(long type, BOOL bFlag)
Load_ArchivePeriod_Trend	long Load_ArchivePeriod_Trend(long type, long hours, ARIANT_BOOL bFlag)
LoadPeriodDataTrend	long LoadPeriodDataTrend(LPCTSTR strFrom, LPCTSTR strTo, long bSample, LPCTSTR bFlag)
PauseTrend	void PauseTrend(void)
PrintTrend	void PrintTrend(void)
SaveCSV	void SaveCSV(LPCTSTR Filename)
SettSaveasexcel	void SettSaveasexcel(void)
SetVisibleChanGrid	long SetVisibleChanGrid(LPCTSTR strTagName, long chanNo, long bFlag)
ShowTrendProperty	void ShowTrendProperty(void)
StartTrend	void StartTrend(void)

(2) Function Description

AddChannel_Archive Function	
Description	This function is to add a channel in a logging group name or a tag name to the list trend if the tag is a logging list trend.
Syntax	long AddChannel_Archive(long channelNo, LPCTSTR modelName, LPCTSTR tagName, LPCTSTR tagDesc) channelNo: Enter LONG-type numbers as the number of the channel to be added. modelName: Enter the logging group name of the logging tag to be added. This is a string value. tagName: Enter the name of the logging tag to be added. This is a string value. tagDesc: Enter the description of the logging tag to be added. If there is no value, use the description value of the tag DB and if there is a value, use it. This is a string value.
Reference	To add three channels to the List Trend AddChannel_Archive(1, “SCADA_SEC”, “SCADA.UT_EHV02.VRS”, “VRS”) AddChannel_Archive(2, “SCADA_SEC”, “SCADA.UT_EHV02.VST”, “VST”) AddChannel_Archive(3, “SCADA_SEC”, “SCADA.UT_EHV02.VTR”, “VTR”)

AddChannel_RealTime Function	
Description	This function is to add a channel in a tag name to the trend if the trend is an online tag. But, in most cases, online monitoring is possible with logging tags, therefore, AddChannel_Archive is more widely used than this function.
Syntax	long AddChannel_RealTime(long channelNo, BSTR tagName, LPCTSTR tagDesc) channelNo: Enter LONG-type numbers as the number of the channel to be added. tagName: Enter the name of the online tag to be added. This is a string value. tagDesc: Enter the description of the logging tag to be added. If there is no value, use the description value of the tag DB and if there is a value, use it. This is a string value.
Reference	To add three channels to the List Trend AddChannel_RealTime (1, “SCADA.UT_EHV02.VRS”, “VRS”) AddChannel_RealTime (2, “SCADA.UT_EHV02.VST”, “VST”) AddChannel_RealTime (3, “SCADA.UT_EHV02.VTR”, “VTR”)

AddTrendChannel Function	
Description	This function is to add a channel to the list trend. But, this function is rarely used because the user needs to input ID information on the logging group and tag as parameters. It is recommended to use AddChannel_Archive if the logging group ID or tag ID is unknown.
Syntax	long AddTrendChannel(LPCTSTR dataSource, long channelNo, long modelID, LPCTSTR modelName, long tagID, LPCTSTR tagName, LPCTSTR tagDesc)

	dataSource: Enter the following string values to identify whether the tag to be added is a logging tag or an online tag. - Logging Tag: “Archive” - Online Tag: “OnLine” channelNo: Enter LONG-type numbers as the number of the channel to be added. modelID: Enter LONG- type numbers as the logging group ID of the logging tag to be added. modelName: Enter the logging group name of the logging tag to be added. This is a string value. tagID: Enter LONG- type numbers as the tag ID of the logging tag to be added. tagName: Enter the name of the online tag to be added. This is a string value. tagDesc: Enter the description of the logging tag to be added. If there is no value, use the description value of the tag DB and if there is a value, use it. This is a string value.
Reference	To add three channels to the List Trend AddTrendChannel("Archive", 1, 9, "INV1_SEC", 2710, "SOLAR.INV_1.1_AI_CT", “CT”) AddTrendChannel("Archive", 2, 9, "INV1_SEC", 2714, "SOLAR.INV_1.1_AI_VO", “VO”) AddTrendChannel("Archive", 3, 9, "INV1_SEC", 2713, "SOLAR.INV_1.1_AI_PW", “PW”)

ApplyTrigger Function	
Description	This function is to apply the trigger tag properties to the list trend. A value appears unless the trigger value is O (True). If the value is O, it is not displayed.
Syntax	void ApplyTrigger(bApply, modelName, tagName) bApply : - Trigger is applied: True - Trigger is not applied: False modelName: Make sure to enter the logging group name if the list trend is logging. If the user fails to enter the logging group name, noting is recorded because the trigger tag can not be searched. tagName: Enter the tag name.
Reference	ApplyTrigger(TRUE, “SCADA_SEC”, “Update”): In case that a tag, whose logging group is “SCADA_SEC”and name is“Update,”is designated as a trigger tag.

GoFirstTrend Function	
Description	It is the function to retrieve sectional history for the next setting time based on the start of the first logging time.
Syntax	long GoFirstTrend(void)

GoLastTrend Function	
Description	It is the function to retrieve sectional history for the previous setting time based on the latest logging time.
Syntax	long GoLastTrend(void)

GoNextTrend Function	
Description	It is the function to retrieve sectional history for the next setting time based on the current logging time.
Syntax	long GoNextTrend(void)

GoPreviousTrend Function	
Description	It is the function to retrieve sectional history for the previous setting time based on the current logging time.
Syntax	long GoPreviousTrend(void)

Loadhistory12h Function	
Description	It is the function to retrieve the history for the section of previous12 hours based on the current logging time.
Syntax	void Loadhistory12h (void)

Loadhistory1day Function	
Description	It is the function to retrieve the history for the section of previous day based on the current logging time.
Syntax	void Loadhistory1day (void)

Loadhistory1h Function	
Description	It is the function to retrieve the history for the section of previous an hour based on the current logging time.
Syntax	void Loadhistory1h (void)

Loadhistory1month Function	
Description	It is the function to retrieve the history for the section of previous a month based on the current logging time.
Syntax	void Loadhistory1month (void)

Loadhistory1week Function	
Description	It is the function to retrieve the history for the section of previous a week based on the current logging time.
Syntax	void Loadhistory1week (void)

Loadhistory2h Function	
Description	It is the function to retrieve the history for the section of previous two hours based on the current logging time.
Syntax	void Loadhistory2h (void)

Loadhistory3h Function	
Description	It is the function to retrieve the history for the section of previous three hours based on the current logging time.
Syntax	void Loadhistory3h (void)

Loadhistory6h Function	
Description	It is the function to retrieve the history for the section of previous six hours based on the current logging time.
Syntax	void Loadhistory6h (void)

LoadHistoryTrend Function	
Description	This function is to search logging data in a designated time zone. Enter year, month, week, day, hour to set the start and end time of a certain time zone and search logging data in the time zone. Enter a number in year, moth, week and day and enter 0 in the rest. Make bFlag TRUE to search data on a specific day/month/year or make it FALSE to search them as of the current time.
Syntax	void LoadHistoryTrend(long year, long month, long week, long day, long hour, BOOL bFlag) year: Enter the year to search in LONG-type numbers. month: Enter the month to search in LONG-type numbers. week: To search data in a certain week, enter 1. If not, enter 0. day: Enter the day to search in LONG-type numbers. hour: Enter the hour to search in LONG-type numbers. bFlag: To search data on a specific day/month/year, enter TRUE. If not, enter FALSE.
Reference	To search logging data in the time zone in which a certain date belongs to, Year Search - LoadHistoryTrend(2007, 0, 0, 0, 0, TRUE) Month Search - LoadHistoryTrend(2007, 5, 0, 0, 0, TRUE)

	Week Search - LoadHistoryTrend(2007, 5, 1, 0, 0, TRUE) Day Search - LoadHistoryTrend(2007, 5, 0, 13, 0, TRUE) Hour Search- LoadHistoryTrend(2007, 5, 0, 13, 10, TRUE) For One Hour Search as of the current time, LoadHistoryTrend(0, 0, 0, 0, 1, FALSE) For 2-Hour Search as of the current time, LoadHistoryTrend(0, 0, 0, 0, 2, FALSE) For 6-Hour Search as of the current time, LoadHistoryTrend(0, 0, 0, 0, 6, FALSE)

LoadPeriodTrend Function	
Description	As a function to search data in a certain period, this presents One Hour Search, One Day search, One Week Search and One Month Search as of the current time.
Syntax	long LoadPeriodTrend(long days, BOOL bFlag) type: Enter the following as parameters for searching. - One Hour Search: 1 - One Day search: 2 - One Week Search: 3 - One Month Search: 4 bFlag – Input TRUE.
Reference	For One Hour Search, LoadPeriodTrend (1, TRUE) For One Day Search, LoadPeriodTrend (2, TRUE) For One Week Search, LoadPeriodTrend (3, TRUE) For One Month Search, LoadPeriodTrend (4, TRUE)

Load_ArchivePeriod_Trend Function	
Description	As a function to search data in a certain period, Hour Search, this presents One Day search, One

	Week Search and One Month Search as of the current time. For Hour Search, 1, 2, 3, 6 and 12 hour search are available.
Syntax	long Load_ArchivePeriod_Trend (long type, long hours, VARIANT_BOOL bFlag) type: Enter the following as period parameters for searching. - One Hour Search: 1 - One Day search: 2 - One Week Search: 3 - One Month Search: 4 hours: This parameter is applied only when One Hour Search is entered for type. Enter 1, 2, 3, 6 or 12 to search 1, 2, 3, 6 or 12 hour search. bFlag – Input TRUE.
Reference	For One Hour Search after selecting Hour Search, Load_ArchivePeriod_Trend(1, 1, TRUE) For One Day Search, Load_ArchivePeriod_Trend(2, 0, TRUE) For One Week Search, Load_ArchivePeriod_Trend(3, 0, TRUE) For One Month Search, Load_ArchivePeriod_Trend(4, 0, TRUE)

LoadPeriodDataTrend Function	
Description	As a function to search data in a certain period, if the user inputs the Start time and End time, this function searches logging data during the period.
Syntax	long LoadPeriodDataTrend(LPCTSTR strFrom, LPCTSTR strTo, long bSample, LPCTSTR bFlag) strFrom: Enter the following type as the Start time. Number 0 through 24 are used for Time Display. “2008/02/14 11:00:00” strTo: Enter the following type as the End time. Number 0 through 24 are used for Time Display. “2008/02/14 11:10:00” bSample: Decide whether to display data after applying sampling. Enter 0 as the default value. bFlag: This is a reserve parameter variable. Enter“”.
Reference	To search data from 13 o’clock on February 11, 2008 to 14 o’clock on February 11, LoadPeriodDataTrend(“2008/02/11 13:00:00”, “2008/02/11 14:00:00”, 0, “”) Remember that there is empty space between day to time.

PauseTrend Function	
Description	It is the function to pause the real-time lookup.
Syntax	void PauseTrend(void)

PrintTrend Function	
Description	It is the function to print out the current list trend screen.
Syntax	void PrintTrend(void)

SaveCSV Function	
Description	It is the function to show the dialog box where you can save the data displayed in the list trend as the Excel file.
Syntax	void SaveCSV(LPCTSTR Filename) Filename: File name for save as

SettSaveasexcel Function	
Description	It is the function to show the dialog box where you can save the data displayed in the list trend as the Excel file.
Syntax	void SettSaveasexcel(void)

SetVisibleChanGrid Function	
Description	This function is to show or hide channels.
Syntax	long SetVisibleChanGrid(LPCTSTR strTagName, long chanNo, long bFlag) strTagName: Enter the tag name of the channel to be shown or hidden. chanNo: Enter the number of the channel to be shown or hidden. bFlag: Input True if the channel is to be shown or input False if it is to be hidden.
Reference	ShowTrendProperty(TRUE)

ShowTrendProperty Function	
Description	This function is to show the property window of the trend.
Syntax	void ShowTrendProperty()
Reference	ShowTrendProperty(TRUE)

StartTrend Function	
Description	It is the function for real-time lookup based on the established screen cycle
Syntax	void StartTrend(void)

11.5 Recipe Viewer

This Recipe Viewer is an Active X module that displays the recipe contents set up during the engineering and has Recipe Control, Dynamic Data Loading and Dynamic Data Saving function.

11.5.1 Prerequisite and Environment

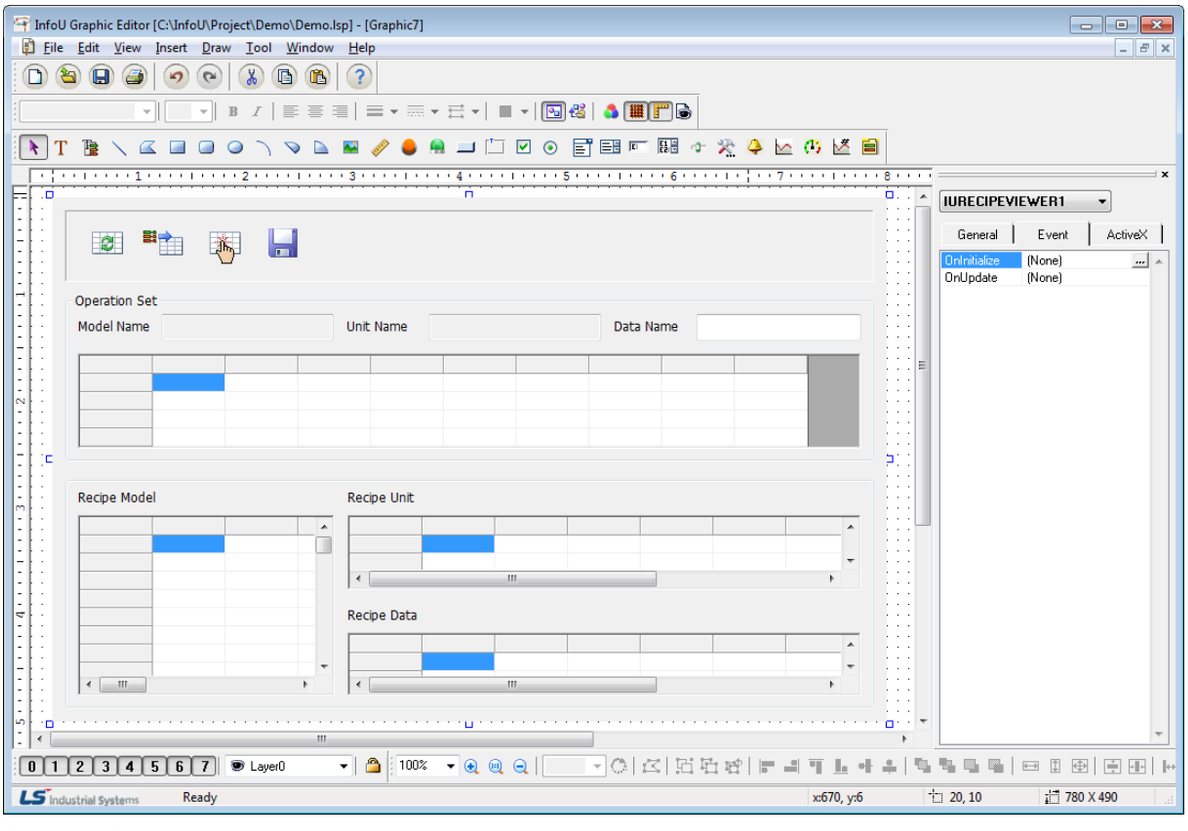
The ActiveX module is inserted to the InfoU graphic editor. The recipe viewer displays the data that have been set up during the engineering and does not have its own property.

The recipe viewer is executed in the following order. If  is selected from the tools, immediate inserting can be performed.

11.5.2 Screen Configuration

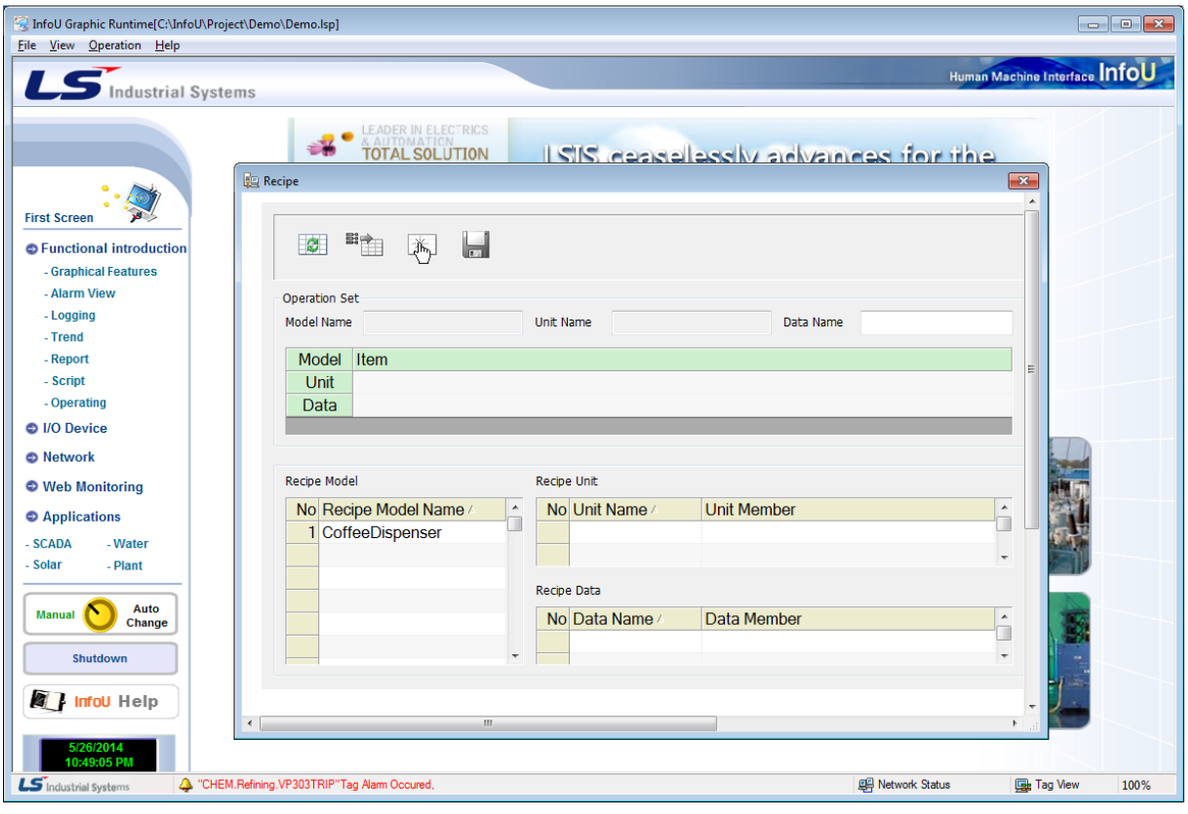
(1) Engineering Screen

The following screen appears after the recipe viewer is inserted.



(2) Runtime Screen

If the above Engineering Screen is executed on the runtime, the screen is as follows.

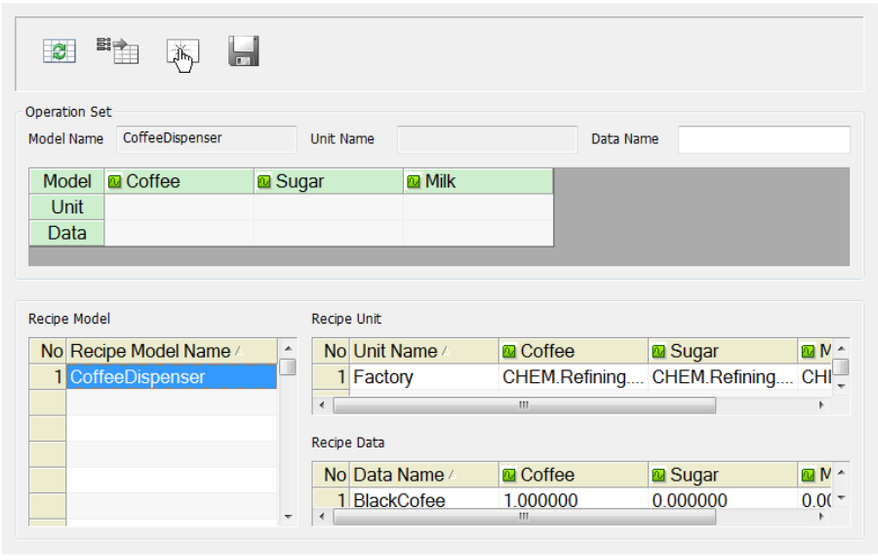


11.5.3 Real-Time Function

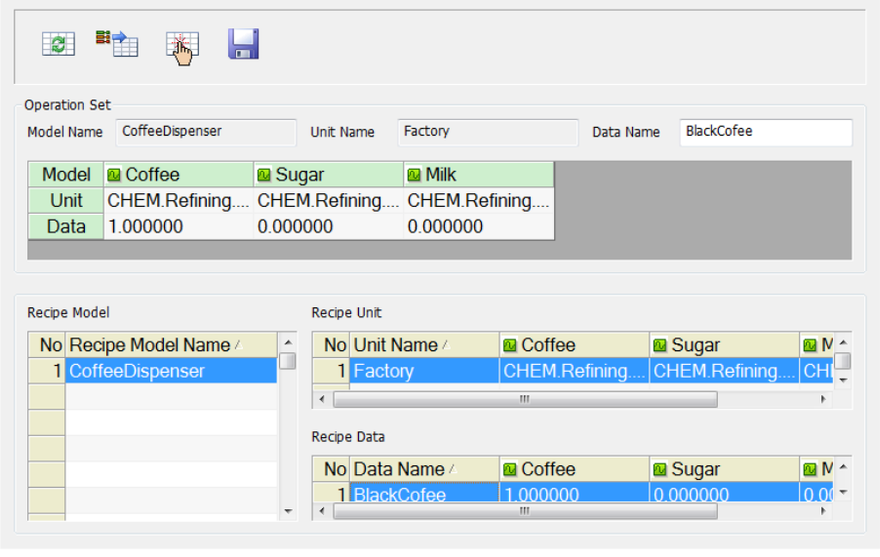
(1) Setting Data Diplay

The recipe viewer is largely divided into the upper Operation Set part and the lower setting data display and select part. The user may conduct various operations by selecting a recipe model and each unit and data item and combining them properly and the selected items are displayed on the upper Operation Set part. During the runtime, a recipe model list is displayed on the left and if a certain model is selected, the associated recipe units and data to the selected model appear.

* In case that only a recipe model is selected

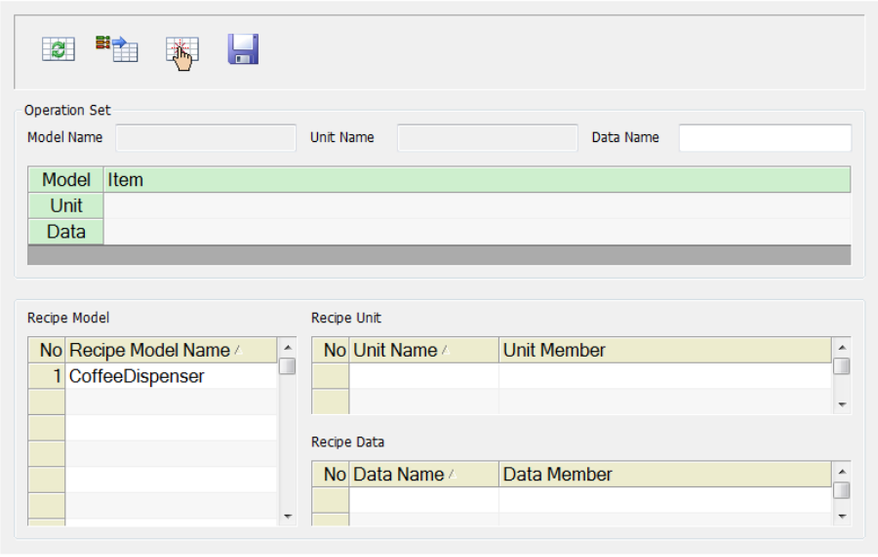


If recipe units and data are selected, the selected items are displayed on 'Operation Set' as seen in the figure.



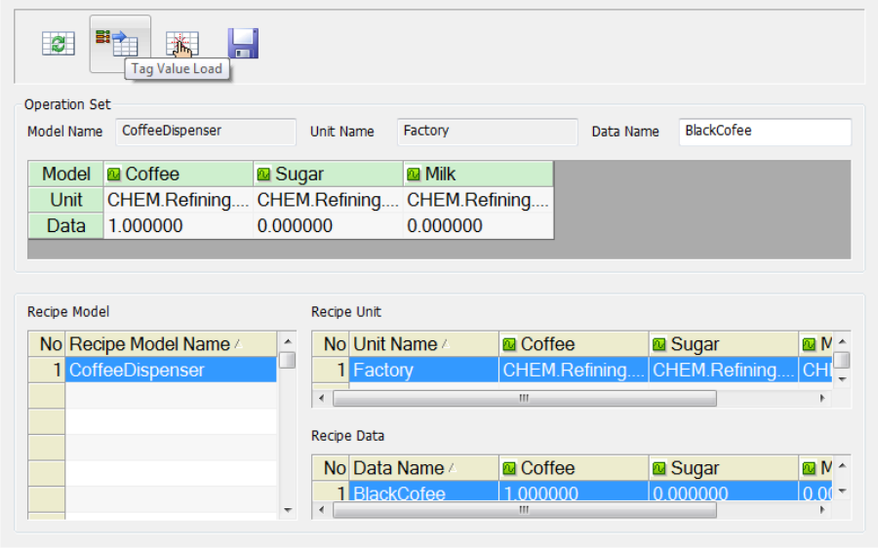
(2) Refresh ( )

If Refresh button is selected, the existing operations are cleared and it is converted to the initial screen.
(Only the model list is displayed)



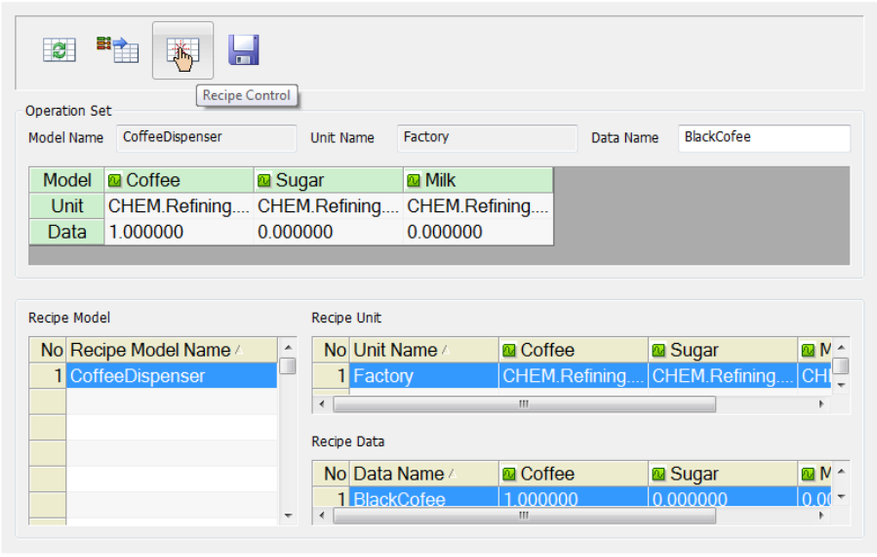
(3) Tag Value Load ( )

If models and units are selected, 'Tag Value Load' becomes active. If this button is selected, the tag values displayed on the Operation Set are loaded from the real-time data service module and they are displayed on the Operation Set. In this status, the user may control the recipe since its requirements (model + unit + data) are met and also save this data in the DB. Of course, it is also available to edit it directly on the data cell of 'Operation Set' to control and save it in the DB.

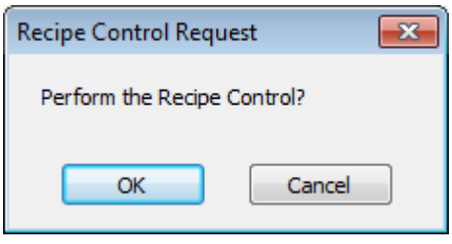


(4) Recipe Control ( )

For Recipe Control, each Model, Unit and Data should be selected. Of course, if some data are secured through Tag Value Load instead of selecting data, Recipe Control is also possible even in such a case. Recipe Control is possible only when the current user's authority is higher than the required authority according to the recipe model.



If Recipe Control is performed, the following dialog box appears to confirm the function.

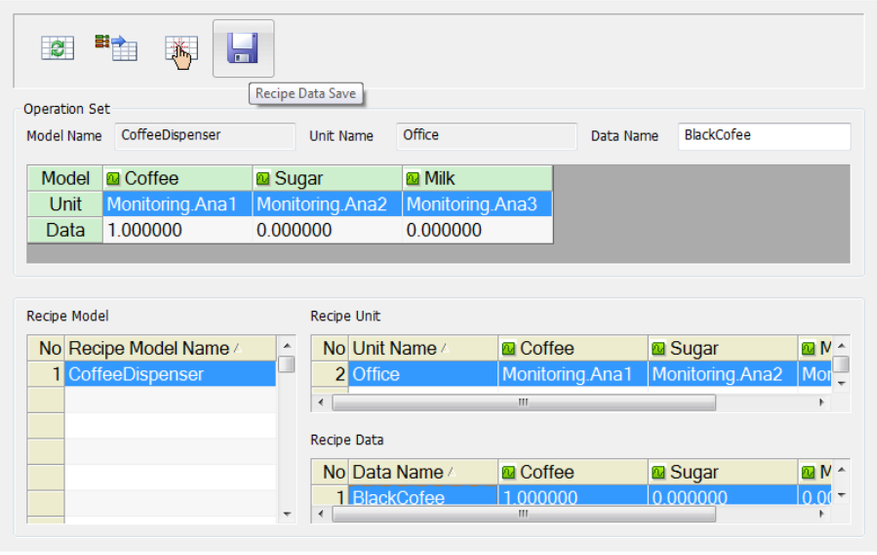


For 'Control Results,' two options-Returns or No Return-are presented. Select one and press OK to perform the Recipe Control. If the user has an insufficiently high authority to control the recipe model, he/she may not control it.

(5) Recipe Data Save ( )

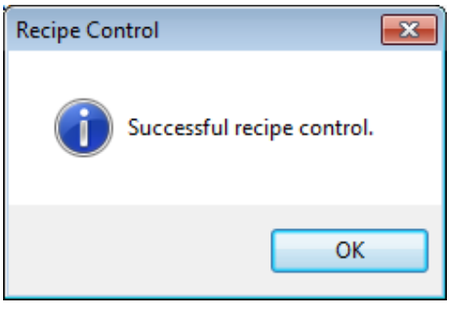
The recipe viewer presents a function to save recipe data during the runtime. But, only up to 64 recipe data can be saved in a model. The user may load the corresponding values to the unit displayed on the Operation Set from the real-time data service and save them through the recipe viewer or input data directly to cells to save them.

When saving data, use the same data name as set up on the Operation Set and if other recipe data has the same name, it is updated. If the data name item is empty, the default string value of "year + month+ day + hour+ minute+ second + msecond" is given as a data name.



(6) Recipe Control Result

The control result can be received if “Return” has been set during the Recipe Control (But, this is applied only when it is not under the web environment). The following screen is displayed automatically if the Recipe Control Result is received from I/O Manager.



11.5.4 How to Use Script

(1) Function List

Function Name	Function Declaration
SelectRecord	short SelectRecord(LPCTSTR ModelName, LPCTSTR UnitName, LPCTSTR DataName)
ControlWorkingSet	short ControlWorkingSet(short nMode)
ControlSet	short ControlSet(short nMode, LPCTSTR ModelName, LPCTSTR UnitName, LPCTSTR DataName)
LoadDynamic	short LoadDynamic(void)
ChangeWSDataName	short ChangeWSDataName(LPCTSTR DataName)
SaveData	short SaveData(void)
SelectRecord	short SelectRecord(LPCTSTR ModelName, LPCTSTR UnitName, LPCTSTR DataName)
ControlWorkingSet	short ControlWorkingSet(short nMode)
ControlSet	short ControlSet(short nMode, LPCTSTR ModelName, LPCTSTR UnitName, LPCTSTR DataName)
LoadDynamic	short LoadDynamic(void)
ChangeWSDataName	short ChangeWSDataName(LPCTSTR DataName)
SaveData	short SaveData(void)
SelectRecord	short SelectRecord(LPCTSTR ModelName, LPCTSTR UnitName, LPCTSTR DataName)
ControlWorkingSet	short ControlWorkingSet(short nMode)
ControlSet	short ControlSet(short nMode, LPCTSTR ModelName, LPCTSTR UnitName, LPCTSTR DataName)
LoadDynamic	short LoadDynamic(void)
ChangeWSDataName	short ChangeWSDataName(LPCTSTR DataName)
SaveData	short SaveData(void)
SelectRecord	short SelectRecord(LPCTSTR ModelName, LPCTSTR UnitName, LPCTSTR DataName)
ControlWorkingSet	short ControlWorkingSet(SHORT nMode)

(2) Function Description

SelectRecord Function	
Description	This function is to make WorkingSet organized by selecting model name, unit name and data name. Model name is an essential item while unit name and data name are optional. If this function is executed, the relevant items are selected on the Recipe Viewer Control to form WorkingSet.

Syntax	short SelectRecord(LPCTSTR ModelName, LPCTSTR UnitName, LPCTSTR DataName) ModelName: As the name of the model to be selected and it is a string value. UnitName: As the name of the unit to be selected and it is a string value. DataName: As the name of the data to be selected and it is a string value.
Reference	Recipe Model Name: Coffee vending machine, Unit Name: Vending machine on the1 floor, Data Name: Milk coffee is selected, SelectRecord(“Coffee vending machine”, “Vending machine on the1 floor”, “milk coffee”) To select only model and unit except data, SelectRecord(“Coffee vending machine”, “Vending machine on the1 floor”, “”)

ControlWorkingSet Function	
Description	A function used when controlling recipe with the WorkingSet data that have been already selected. Controlling is performed only model, unit and data are all equipped with. For local system (not web), it is available to select whether to receive the control result or not.
Syntax	short ControlWorkingSet(short nMode) nMode: A flag whether to receive the control result or not. (1: Won't receive , 2: Will receive)
Reference	To perform controlling WorkingSet (Will receive), ControlWorkingSet(2) To perform controlling WorkingSet (Won't receive), ControlWorkingSet(1)

ControlSet Function	
Description	A function used to perform controlling through a certain method without selecting directly on the recipe viewer control. Internally, this function derives the same result as SelectRecord and ControlWorkingSet are executed simultaneously.
Syntax	short ControlSet(short nMode, LPCTSTR ModelName, LPCTSTR UnitName, LPCTSTR DataName) nMode: A flag whether to receive the control result or not. (1: Won't receive , 2: Will receive) ModelName: As the name of the model to be selected and it is a string value.

	UnitName: As the name of the unit to be selected and it is a string value. DataName: As the name of the data to be selected and it is a string value.
Reference	Recipe Model Name: Coffee vending machine, Unit Name: Vending machine on the1 floor, Data Name: Milk coffee is selected, (Not received the result), Controlset(1, “Coffee vending machine”, “Vending machine on the1 floor”, “milk coffee”)

LoadDynamic Function	
Description	This function is to load tag values that belong to the unit member from the InfoU real-time server controlling without using the values that have been set up during the engineering. This function can be executed only when model and unit are selected.
Syntax	short LoadDynamic(void)
Reference	With this function, it is possible to load real-time values of a tag and save them as recipe data or re-control them.

ChangeWSDataName Function	
Description	This function is to rename the recipe data of WorkingSet.
Syntax	short ChangeWSDataName(LPCTSTR DataName) DataName: As the name of the data to be selected and it is a string value.
Reference	With this function, the name of the data shown on the current WorkingSet can be modified. This function only helps to modify the name seen on the screen and does not mean to save it in the DB. After this function is used, the data can be saved in the DB with the following SaveData function.

SaveData Function	
Description	This function is to save the recipe data displayed on the WorkingSet.
Syntax	short SaveData(void)
Reference	This function is to save the data seen on the current WorkingSet in the DB. At this time, use the data name on the WorkingSet of the recipe viewer control for the name of the saved data. This name is changeable by using ChangeWSDataName function. If the same data name already exists in the DB, a new data value is renewed in the DB, If the same data name does not exists in the DB, a new recipe data is added to the DB. Also, if any recipe data name is not given, a new name is created and saved by combining time such as year, month, day, hour, minute, second and msecond.

11.6 Gauge Control

This Gauge Control is an ActiveX module to display and monitor the registered tag information in a type of gauge that fits well to the user's purpose by selecting or setting up a proper gauge. A gauge can be formed in various types by designating basic range or degree.

11.6.1 Prerequisite and Environment

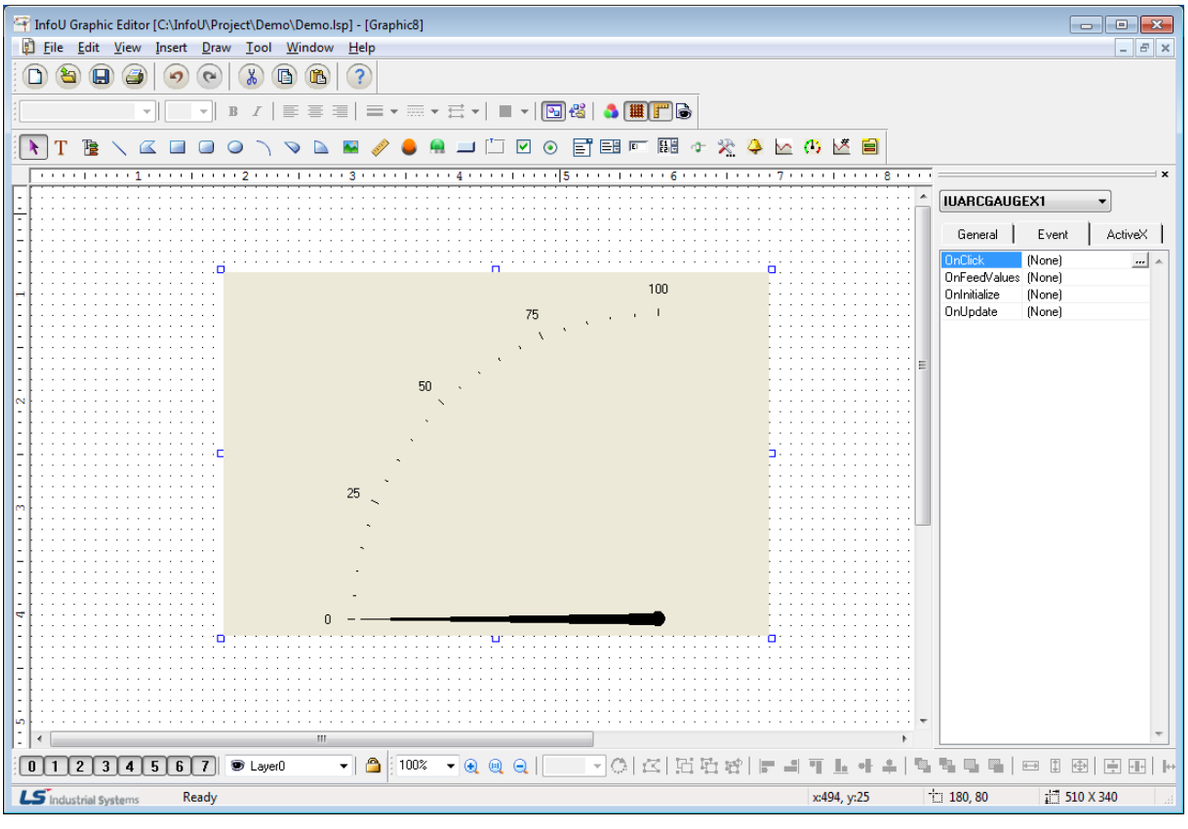
Gauge properties can be set up only after the ActiveX module is inserted to the InfoU graphic editor. Gauge properties broadly consist of General and Ticks. Since those gauge properties have been already set up with default values, they act with those default values if the user does not set up additionally.

The Gauge Control is executed in the following order. If  is selected from the tools, immediate inserting can be performed.

11.6.2 Screen Configuration

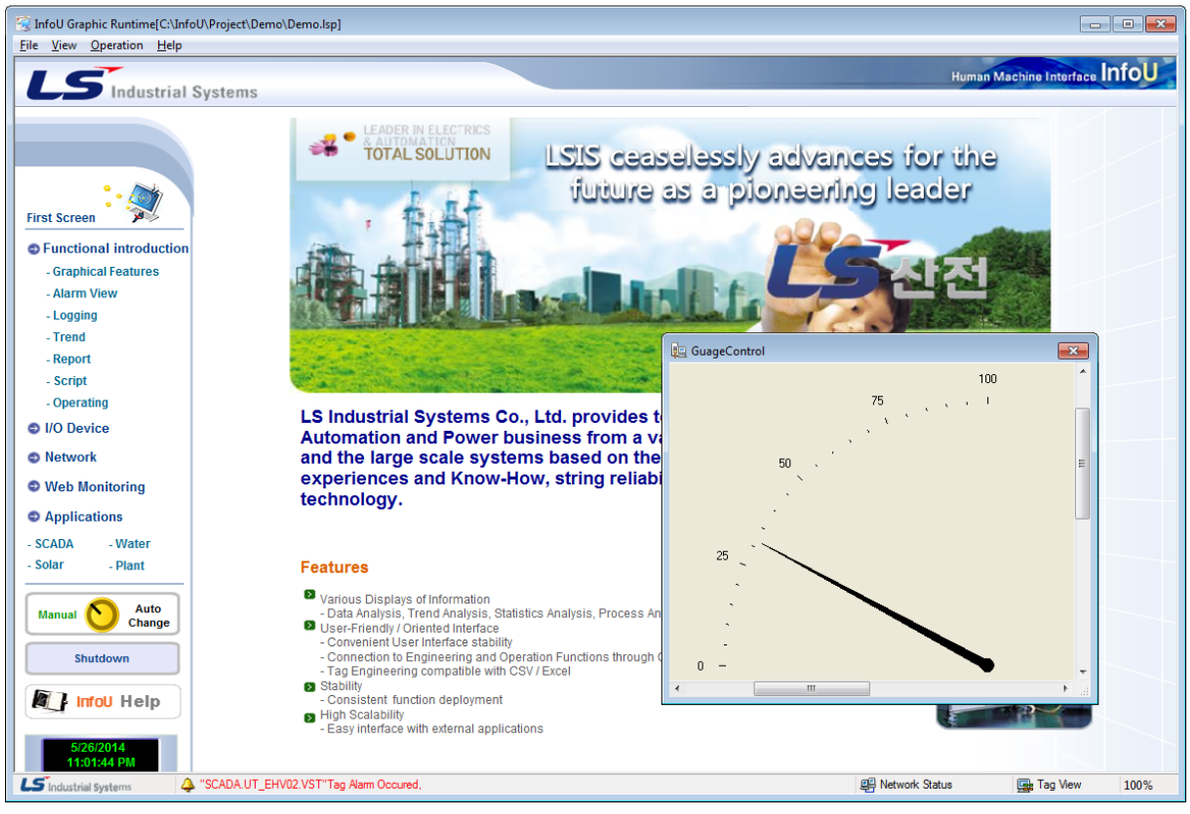
(1) Engineering Screen

The following screen appears after the Gauge Control is inserted.



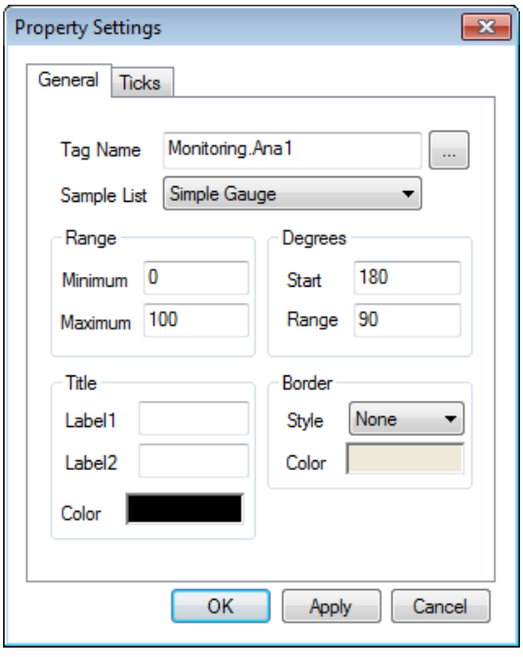
(2) Runtime Screen

If the above Engineering Screen is executed on the runtime, the screen is as follows.



11.6.3 Property Settings

(1) General



1) Tag Name

Enter a tag name to be gauged. A tag name can be designated with 'Search' (...) button on the right. Its

update cycle is the same with the screen cycle of the InfoU graphic editor.

2) Sample List

Various gauge samples are presented to fit to the user's purpose.

3) Range

Designate a gauge range, that is, enter the min/max value.

4) Degrees

Designate a gauge degree, that is, enter the start degree and the range degree.

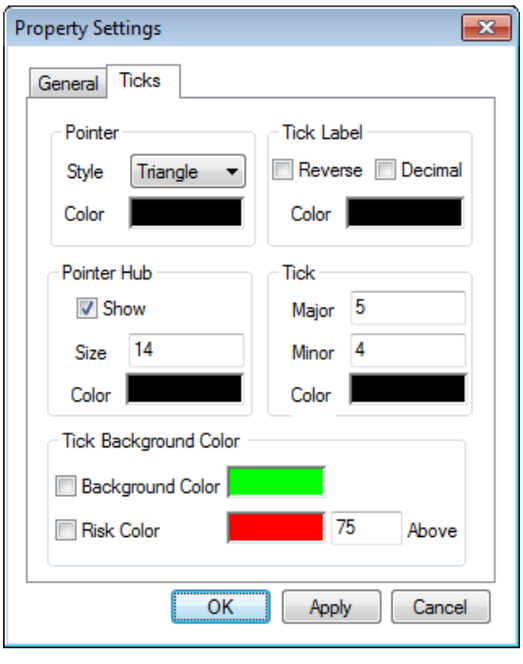
5) Title

Designate a gauge title and color.

6) Border

Designate background color and border style.

(2) Ticks



1) Pointer

Designate a gauge pointer style and color.

2) Tick Label

Designate a tick label color and style (Reverse, Decimal).

3) Pointer Hub

Designate whether to show the pointer hub or not and decide its size and color.

4) Tick

Designate the size of the Major Tick and Minor Tick and color.

5) Tick Background Color

Designate the tick background color and the user may set up an additional risk reference value to designate a risk color.

If not checked, the background color is set up to display.

11.7 How to Use External ActiveX

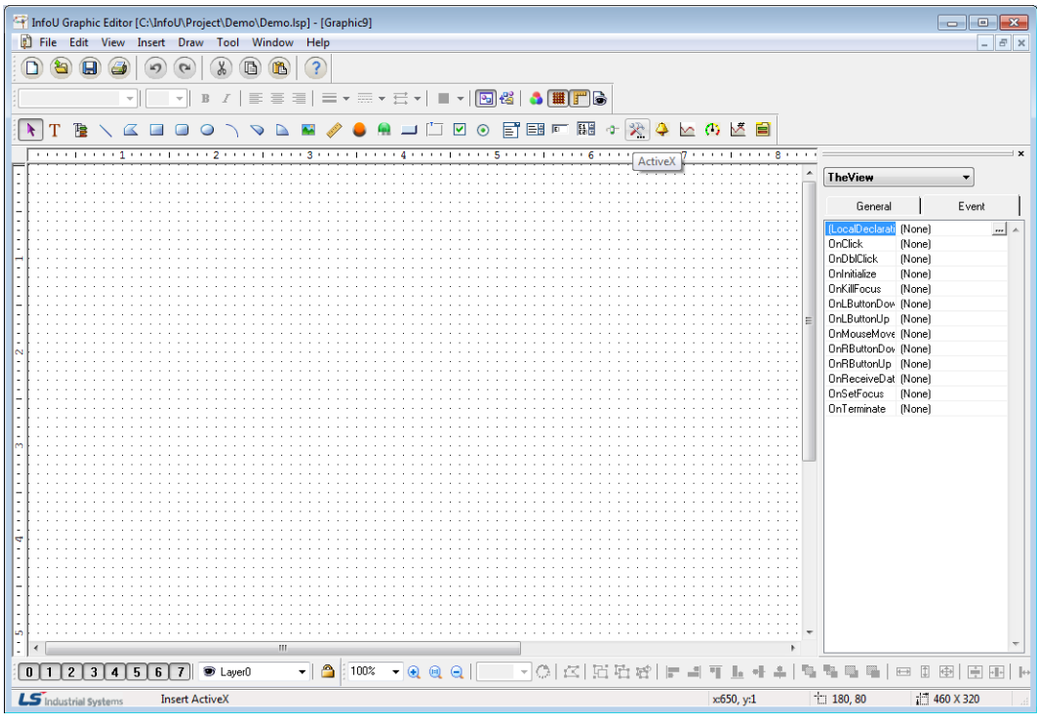
External ActiveX can be inserted and used on the graphic editor.

Here, Window Media Player OCX is used as an example of external The same way can be applied to other ActiveX.

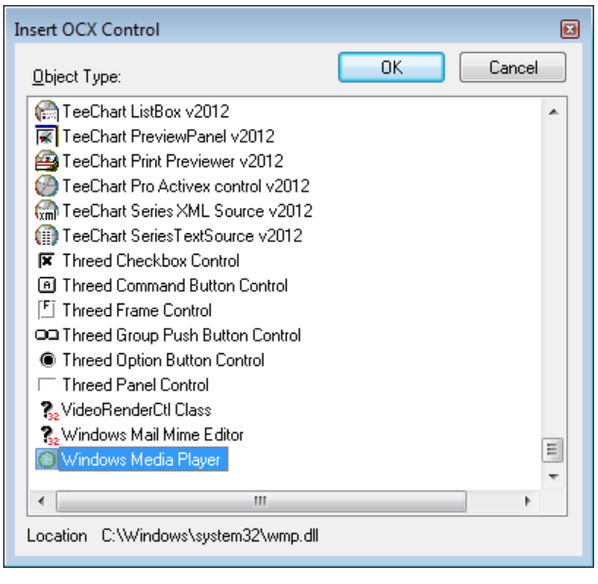
11.7.1 How to Use

Insert the ActiveX module into the InfoU graphic editor. You can run the Window Media Player Control based on the below procedures.

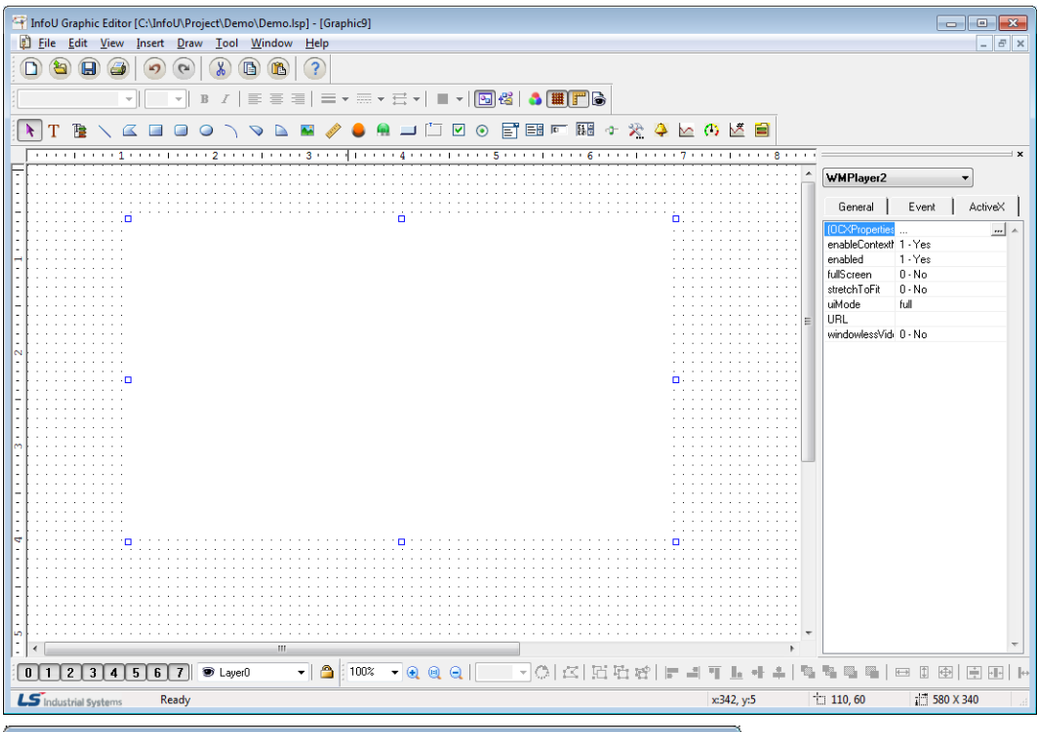
(1) Select  in the tool menu of the graphic editor.

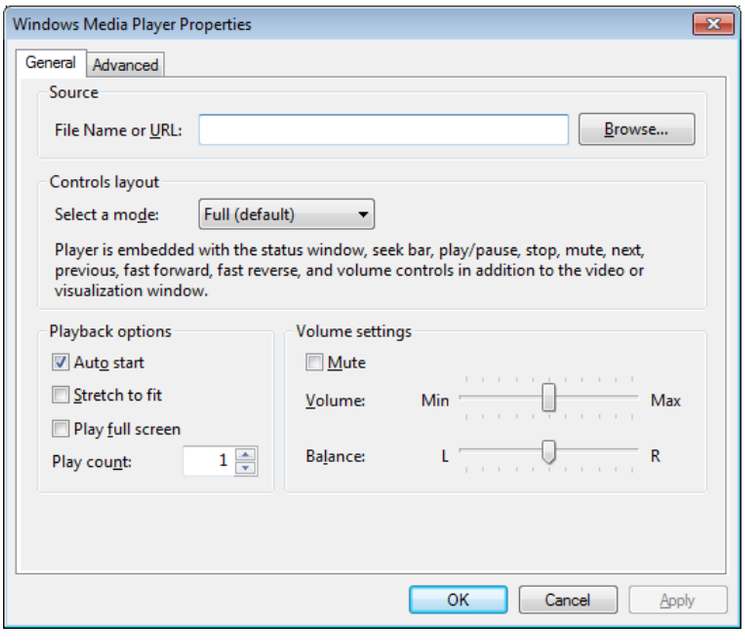


(2) Select a 'Window Media Player'



(3) If you select the 'Window Media Player', it will be inserted.





Chapter 12 Alarm Management

Alram function in HMI system, including the InfoU, is important.
This section describes about the process dealing with alarm, configuration of alarm group and alarm service.

12.1 Alarm Kind and Rules

Alarm settings are available only for analog and digital tags in the InfoU and alarms are processed only when the relevant tag value is normal.

Each alarm is prompted or cancelled according to the following rules.

Tag Type	Alarm Type	Alarm Kind	Occurrence rule	Cancel rule
Analog	Limit Alarm	HH Alarm	The alarm occurs when the tag value is more than HH	The alarm is cancelled if HH alarm has previously occurred and the tag value is less than HH.
		HI Alarm	It occurs when the tag value is more than HI but less than HH.	It is cancelled if HI alarm has previously occurred and the tag value is less than HI or more than HH.
		LO Alarm	It occurs when the tag value is more than LL but less than LO.	It is cancelled if LO alarm has previously occurred and the tag value is less than LL or more than LO
		LL Alarm	It occurs when the tag value is less than LL	It is cancelled if LL alarm has previously occurred and the tag value exceeds LL
	Change Value Alarm	Change Value Alarm	It occurs when the tag value is significantly changed from the past value by more than the baseline value.	It is cancelled if the Change Alarm has previously occurred and the tag value is changed from the past value by less than the baseline value.
	Gap Alarm	Main Gap Alarm	It occurs when the tag value is different from the base value by more than the Main Gap Value.	It is cancelled if the Main gap Alarm has previously occurred and the tag value is different from the base value by less than Main Gap Value
		Sub Gap Alarm	It occurs when the tag value is different from the base value by more than the Sub Gap Value but less the than Main Gap Value.	It is cancelled if the Sub gap Alarm has previously occurred and the tag value is different from the base value by more than the Main Gap Value or less than the Sub Gap Value.
Digital	Digital Alarm	ON Alarm	It occurs when the tag value is “On.”	It is cancelled if the On Alarm has previously occurred and the tag value is Off.

		OFF Alarm	It occurs when the tag value is “Off.”	It is cancelled if the Off Alarm has previously occurred and the tag value is On
		ON → OFF Alarm	It occurs when the tag value is changed from On to Off.	Occurrence and cancellation takes place at the same time
		OFF → ON Alarm	It occurs when the tag value is changed from Off to On	Occurrence and cancellation takes place at the same time
		Alarm When Changed	It occurs when the tag value is changed	Occurrence and cancellation takes place at the same time

12.1.1 Analog Alarm

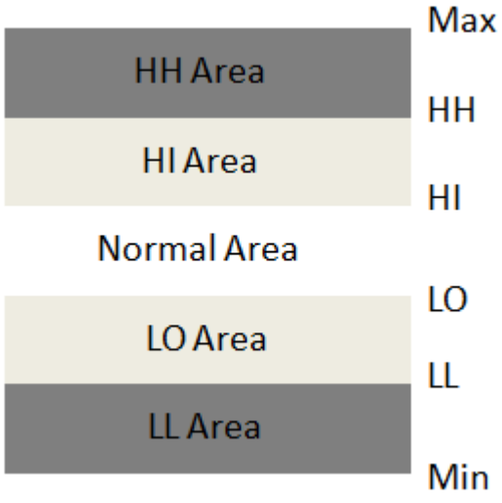
There are three types of analog tags; Limit Alarm, Variation Alarm, Gap Alarm.

There are four types of Limit Alarms; HH, HI, LO, LL alarm. The Gap Alarm can be divided into the Main Gap Alarm and the Sub. Gap Alarm.

(1) Limit Alarm

The border alarm is used most among other alarms for analog tags.

The user may set up four border values such as HH, HI, LO and LL for the border alarm when tag engineering. If the user divides the region corresponding to the tag values horizontally and locates the baseline values to relevant regions, the relevant alarm occurs when the tag value is in the corresponding region and it is cancelled when the tag value is out of the region.

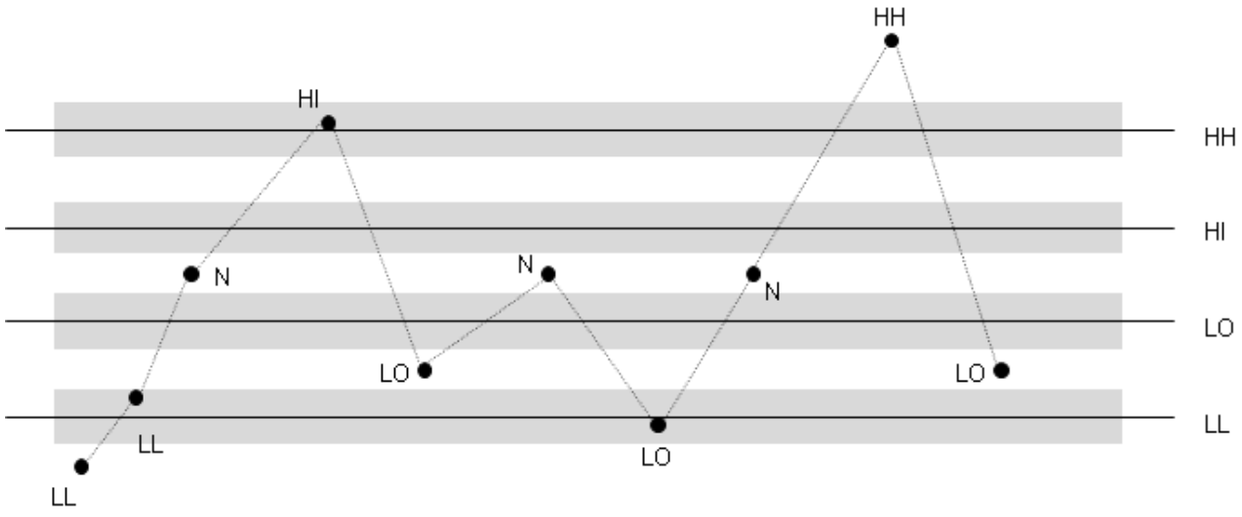


Each border alarm occurs in each corresponding region under the following condition:

* HH Alarm: The tag value is in HH region.
* HI Alarm: The tag value is in HI region
* LO Alarm: The tag value is n LO region
* LL Alarm: The tag value is in LL region

In addition, if any alarm Deadband value is set up, the Deadband rule applies according to the change trend of the tag value.

The following figure shows a simple example of this.



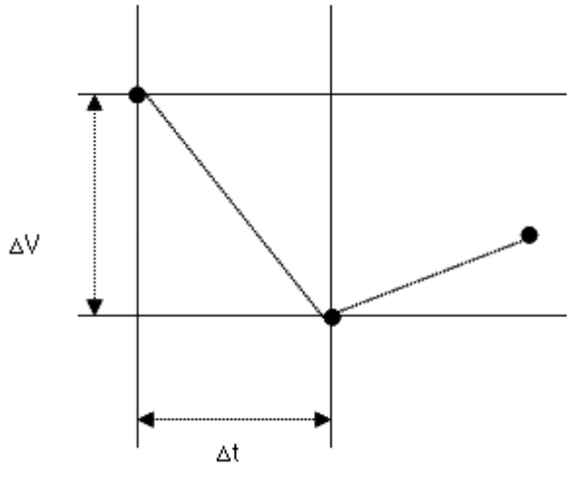
(2) Change Value Alarm

The change alarm occurs when the current tag value is significantly changed from the past tag value by more than the allowed change value designated to each tag property item.

Similarly, if the change alarm occurs and then, the tag value is changed afterward but it does not reach the designated value, the alarm is cancelled. This alarm is set to ensure the tag value is not change significantly.

The alarm Deadband rule does not apply to the change alarm.

As shown in the below figure, the horizontal △t means the collecting cycle (not absolute time) and △V means the difference in values. If the variations between the previous value and the current value exceed the variation value, the corresponding alarm will occur.



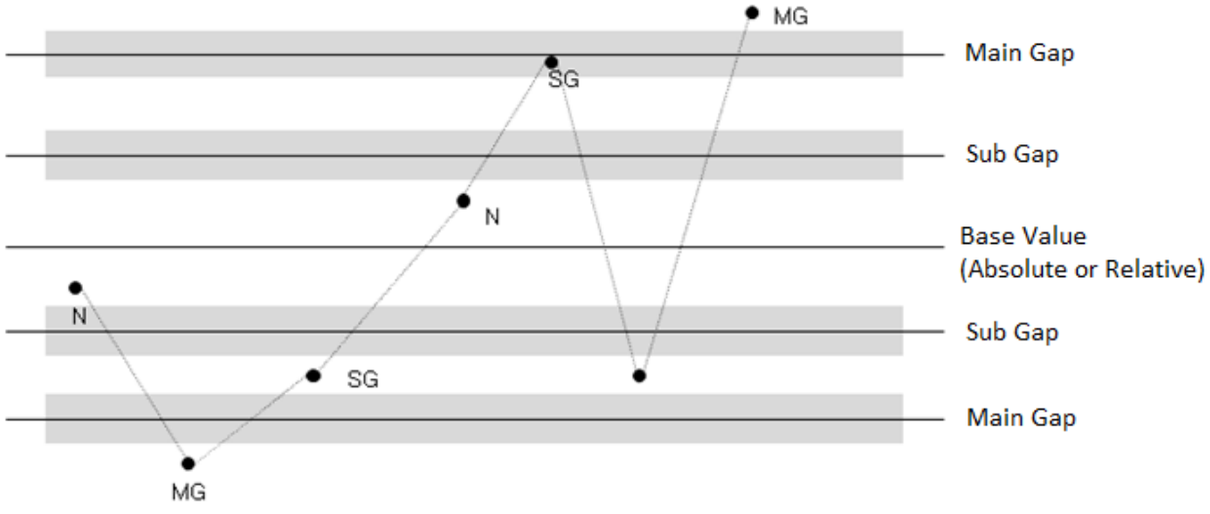
(3) Gap Alarm

The gap alarm occurs when the delta rule of the Main and Sub Gap applies based on the baseline value. Similarly to the border alarm, the alarm Deadband rule applies to the gap alarm.

The gap alarm is similar to the border alarm but several differences are as follows:

-There is no upper or lower border. It is only divided into the Main and Sub Gap.

-The base value can be set with an absolute value or a relative value (tag).



12.1.2 Digital Alarm

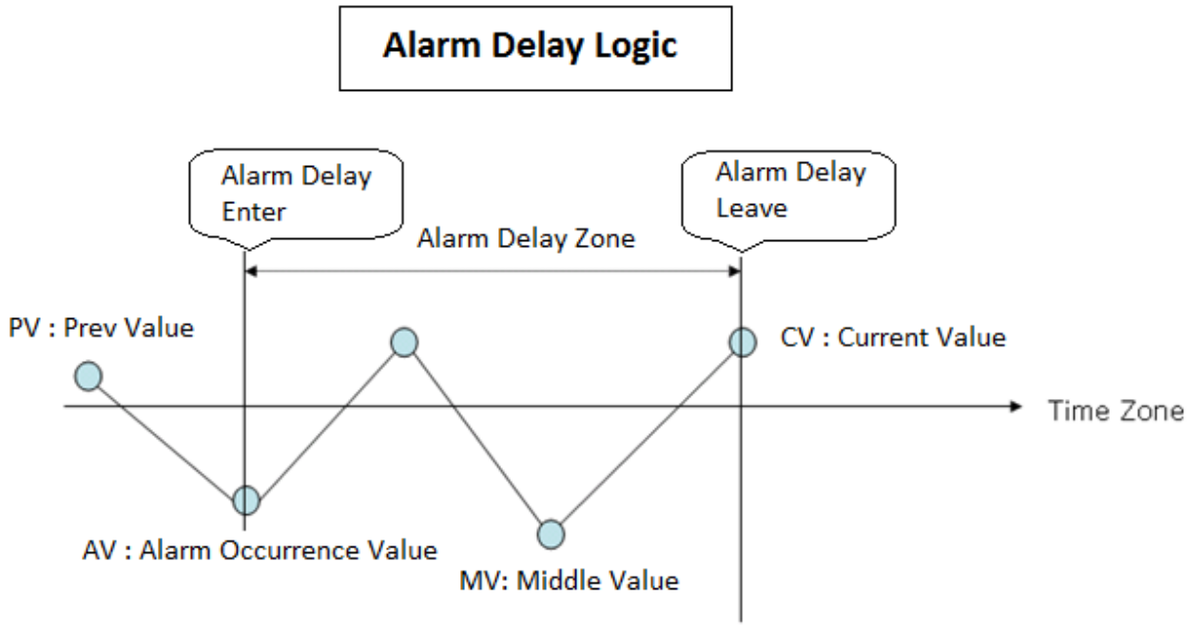
Digital alarms occur only for digital tags and they occur according to the previous value and the current value, consisting of five types such as On->On Alarm, Off Alarm, On->On->Off Alarm, Off->On->On Alarm, Alarm When Changed.

Alarm Kind	Description
ON Alarm	The alarm occurs when the tag value is On. The alarm is cancelled when the tag value is Off.
OFF Alarm	The alarm occurs when tag value is Off The alarm is cancelled when the tag value is On
ON → OFF Alarm	The alarm is occurred and cancelled at the same time when the tag value is changed from On to Off.
OFF → ON Alarm	The alarm is occurred and cancelled at the same time when the tag value is changed from Off to On.
Alarm When Changed	The alarm is occurred and cancelled at the same time when the tag value is changed.

12.1.3 Alarm Delay

(1) Alarm Delay Description

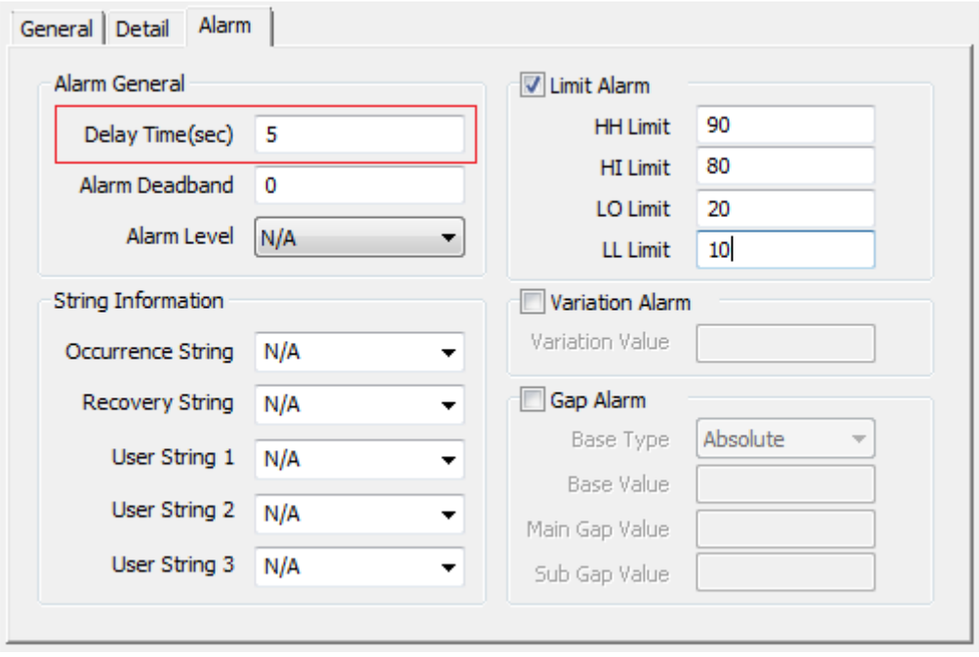
Alarm delay is a function to delay the occurrence of an alarm when some abnormal value is produced in the site by assigning a certain delay time to each tag and comparing the value after the specified time and the one before the initial occurrence of the alarm to prompt an alarm in stead of prompting an alarm immediately when some abnormal value is produced.The alarm delay diagram is as follows



PV : The last value before entering the alarm delay zone
AV : The value when entering the alarm delay zone
MV: The value in the alarm delay zone
CV : The current value at the end of alarm delay zone

(2) Alarm Delay settings

You need to set the following two items to use the Alarm Delay function. First of all, enter the delay time value in the alarm setting property window when editing tags. This value should be set for each tag individually.



The alarm delay time is applied all alarms of the relevant tag. Then, for the item to be set next, in the bin directory of the InfoU installation directory, open the iuDataProcess.ini file with a notepad or other editing tools.

After setting the value of the following item as TRUE in the ini configuration file, save it.

```
DP_ALARMDELAY_FLAG=TRUE
```

If you start up the InfoU runtime, you can check the alarm delay time is applied.

Notice

Setting the alarm delay time may cause considerable load when applying it to the InfoU runtime.
Alarm delay time settings should be used only if absolutely necessary.

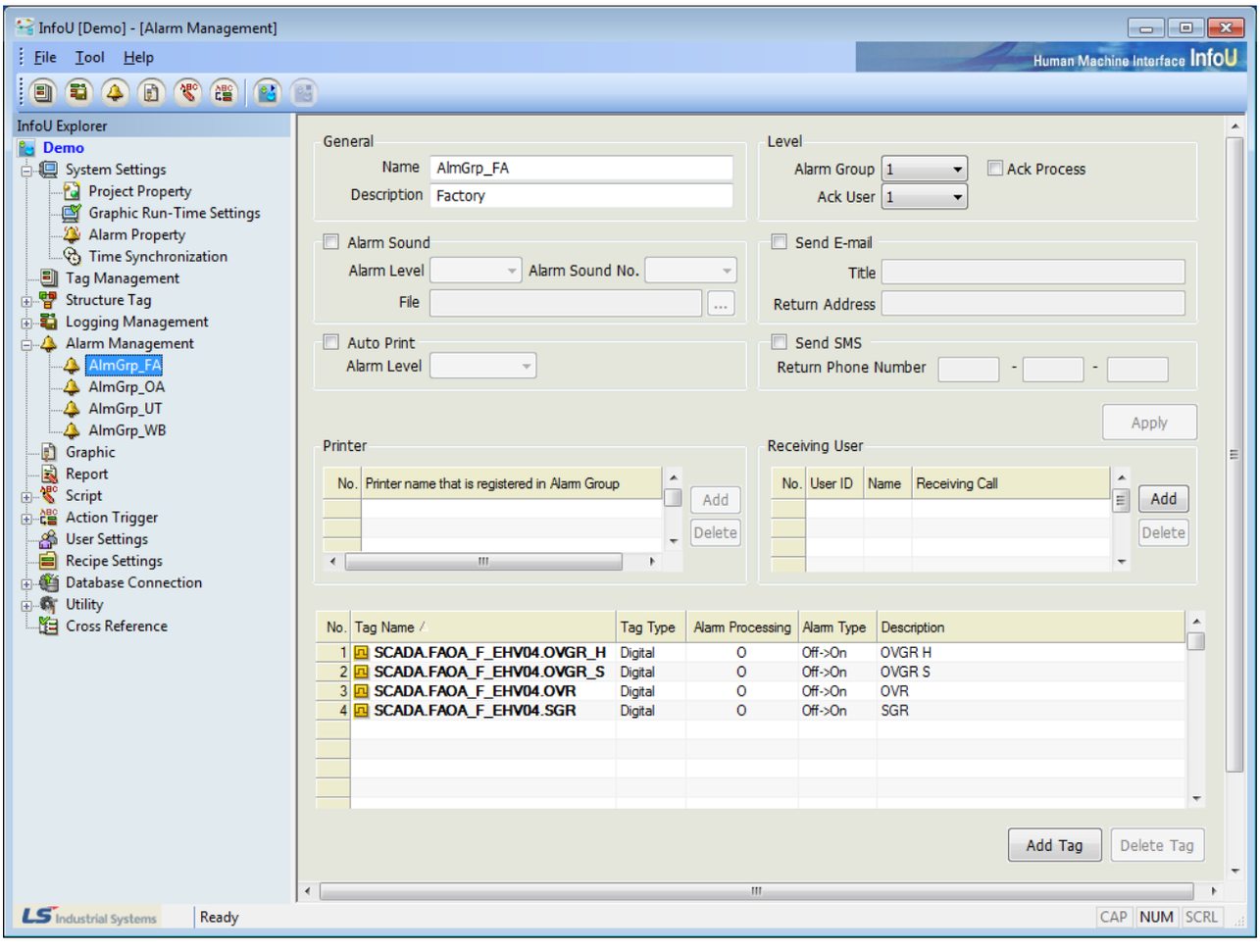
12.2 Alarm Group Settings

If any alarm occurs, alarm sounding, printing, sending SMS and sending e-mail are carried out as actions against the relevant alarm. Those actions are set up for each alarm group. Therefore, several tags are registered in one alarm group and if any alarm occurs in one of the registerd tags, the designated actions to the group are performed.

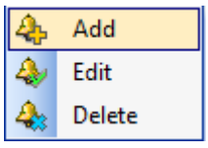
12.2.1 Alarm Group Management

(1) Basic Settings

Select 'Alarm Management' from InfoU Exploerer. If there are alarm groups that have been already registered, the alarm group list appears under the 'Alarm Management.' If one alarm group is selected, information on the alarm group is displayed.



1) Add

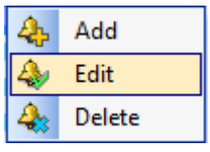


Click the right key of the mouse and check ‘Add’ menu.

An empty box in the Alarm Group Settings’ is displayed.

If a new name is designated to the existing alarm group without any modification, a new alarm group is registered..

2) Edit



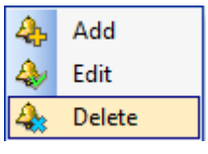
Click the right key of the mouse after selecting an alarm group and check ‘Edit.’

Selecting an alarm group itself plays the same role as selecting ‘Edit’ does.

Existing settings are displayed in the ‘Alarm Group Settings’’

After modifying settings, finish editing with saving.

3) Delete



Select an alarm group to delete. Click the right key of the mouse and check ‘Delete’ menu.

The selected alarm group is deleted.

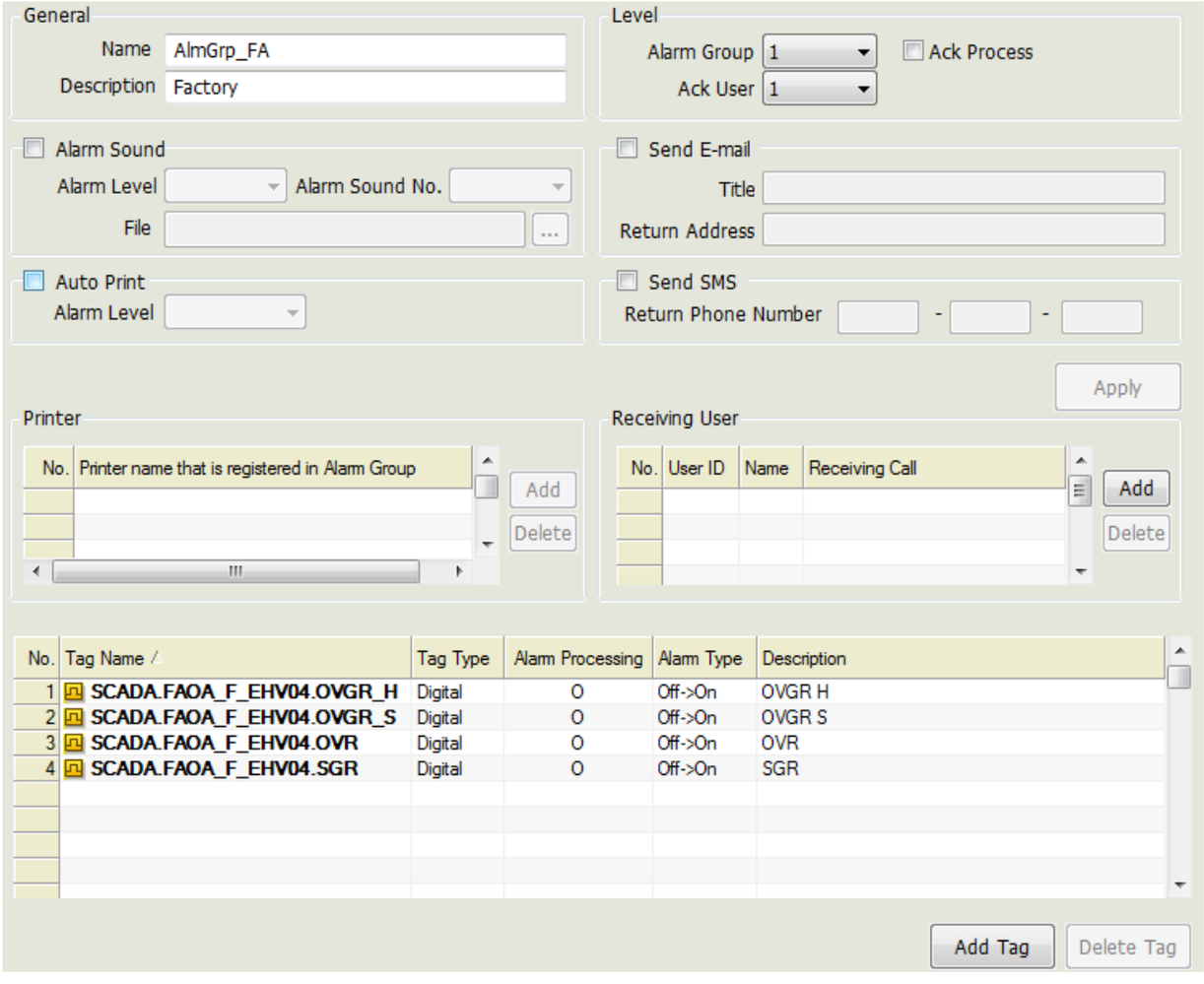
(2) Alarm group property window

If you select the alarm group name in the alarm group list or select the [Edit] menu, the settings of the alarm group will be displayed in the alarm group property widow on the right side. For the description on each item, refer to [Alarm Group Settings].

12.2.2 Alarm Group Settings

The alarm group setting screen is as below.

The single setting items at the top of the screen are saved only when you press the [Apply] button in the middle. After the single items setting is saved, multiple setting times such as printer and receiving user, tag list, etc. are activated. The multiple setting items are automatically saved.



(1) General

Items	Description
Name	Input name of the alarm group. If changing the existing name, the group is automatically acknowledged additionally
Description	Input description of the alarm group

(2) Level and Ack Process

Items	Description
Alarm Group	Select the level of alarm group. (1 ~ 10)
Ack User	Select a level of the acknowledged users(1 ~ 255)
Ack Process	If 'Ack Process' is selected, any alarm can not be deleted from the alarm list without the user's acknowledgement. Even though it has been set up in the 'Alarm Settings' that any alarm can be deleted without acknowledgement, 'Ack Process' check should have priority over those settings.

(3) Alarm Sound

Items	Description
Alarm Sound	In case of the occurrence of any alarm, output of a wave file is set up for a particular group.
Alarm Level	Alarm sound is generated only when the alarm level is higher than the designated alarm level.
Alarm Sound No.	Select the number of alarm sound. (Unlimited / No Sound)
File	Select an alarm sound to generate. Select a wave file.

(4) Send E-mail

Items	Description
Send E-mail	Select whether to use Send E-Mail in case of the occurrence of any alarm in the current group..
Title	Input a title of the mail when sending e-mail on alarm information
Return Address	Input e-mail address of the sender when sending e-mail on alarm information.

(5) Auto Print

Items	Description
Auto Print	Select whether to use Auto Print in case of the occurrence of any alarm in the current group.
Alarm Level	Auto Print is processed only when the alarm level is higher than the designated alarm level.

(6) Send SMS

Items	Description
Send SMS	Select whether to use SMS in case of the occurrence of any alarm in the current group.
Return Phone Number	Input the phone number of the sender who will be displayed when sending SMS.

(7) Printer

Indicate a printer to be used for alarm output.

Be cautious that only line printers are supported in this system.

(8) Receiving User

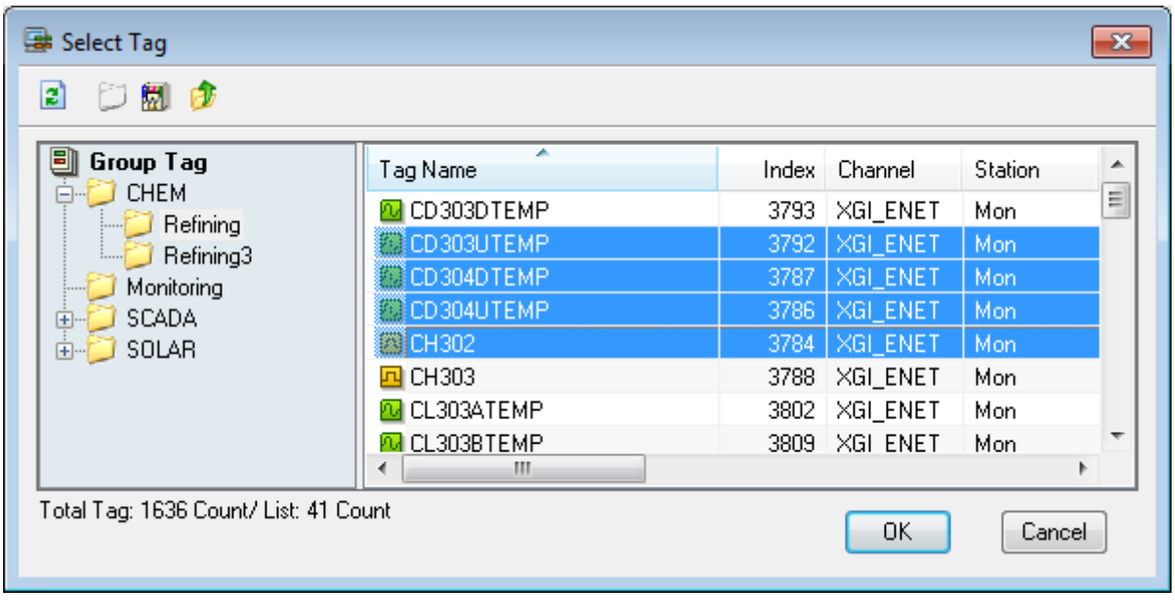
Send e-mail or SMS to the registered users. The User must be registered previously when to use.

(9) Link Tag Settings

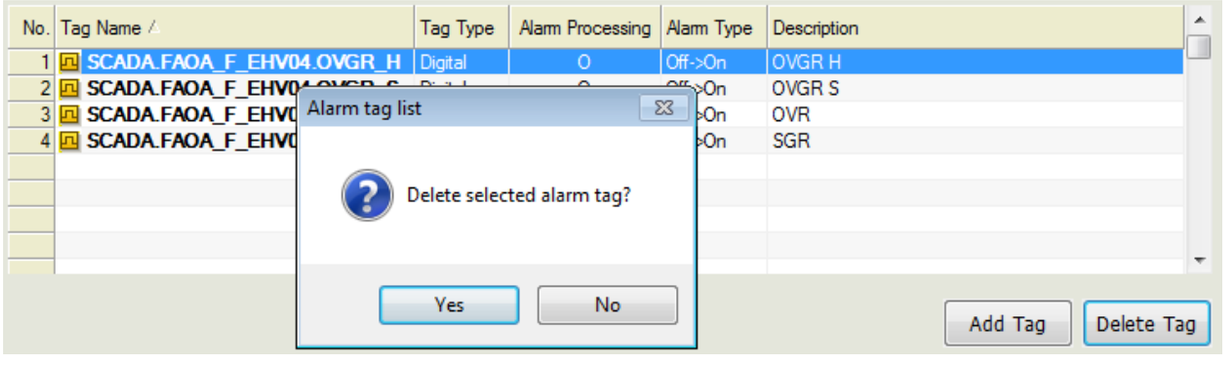
Link Tag Settings are to register a tag to an alarm group. For this, select a tag from the additional tag browser and register it. If any alarm occurs in the selected tag here, the actions that have been designated to the corresponding alarm group are processed.

If the tag browser is displayed, select a tag to add to the list. It is also available to select multiple tags.

Once selecting is completed, the selected tag is added to the Link Tag list.



If you want to delete tags from the alarm group, after selecting the tag on the tag list screen of the alarm setting window, press the [Delete] button to remove the registered tag. Deleting the alarm tag in the alarm group setting window is just the function to cancel tag registration, it does not mean the registered tag is not actually deleted.



12.3 Alarm Service

When the events such as occurrence and cancellation of alarms are generated in the InfoU runtime, the alarm service items describe how a user recognizes it by various means and manage them.

12.3.1 Alarm Sound

Among alarm service items, the alarm sound is related to sounds. It shows you how to generate the preset sound and recognize and manage it when an alarm occurs.

(1) Occurrence of alarm sound

When the sound is set for the alarm group including the tag where the alarm occurred, the relevant WAVE file will be output. In the real-time alarm list, the sound is generated in the order of age with the highest priority (low level value). When a user recognizes the alarm tag, the alarm sound will be cleared.

(2) Alarm sound stop

You can stop the alarm sound by using the 'Sound Stop' function of the alarm viewer or the script. However, this is temporary suspension, namely, the sound conditions of the tag targeted by the alarm sound are not cleared. After stopping the sound, if a user commands restart of sound, the sound will resume for the previous alarm tag.

When the alarm is sounded, the sound will stop in the following cases.

[Cases to stop the alarm sound]

Sound stop cases	Description
When the alarm sound is recognized	If a user recognizes the alarm, the sound will stop. Then, the next alarm of the real-time alarm list will be sounded.
When full recognition is done	If a user recognizes alarms fully, all currently occurring alarms will be recognized. It means sound occurrence conditions on the real-time alarm will be removed. After that, when a new alarm subject to sound occurs, it will be sounded.
When the alarm is deleted from the real-time alarm list	The alarm sound is intended for the real-time alarm list. When the alarm is deleted from the real-time list, the alarm will stop. Then, the next target will be selected and sounded.
Sound stop button / stop script	When a user stops the sound through the 'Sound Stop' button of the alarm viewer or the script, the sound will be suspended. This is just the temporary pause function of the sound so it does not mean the alarm sound will be removed completely. If you execute the 'Sound Start' command, the alarm will be sounded again.

(3) Alarm sound settings

The alarm sound engine gets the desired alarm group data from the real-time list on the specified cycle and generates sound. If you want to generate sound targeting the cycle to check the tags for alarm sound or the desired alarm group only, you can specify actions of the sound engine by editing the setting file.

Open [InfoU project folder]- [Config folder] –[iuAlarm.ini file] using a notepad, etc.

The item related to the alarm sound is [iuAlarmAgent] section and you can specify the following items.

Items	Description
SoundAlarm groupFilter	By using the vertical line(vertical bar), set the alarm group to generate the sound as the condition. The default is ALL(whole groups). For example, if you want to set up “AlmGroup1” and “AlmGroup2”, set as follows. SoundAlarm groupFilter =AlmGroup1|AlmGroup2 If you want to target the alarm group “A”, “B”, “C”, set as follows. SoundAlarm groupFilter=A|B|C If you want to target all alarm groups rather than specifying a certain alarm group, set as ALL. SoundAlarm groupFilter=ALL
SoundAlarmScanTime	Set the cycle to communicate with the alarm server. The unit is a millisecond. If the set value is too small, it may lower the performance of the InfoU system. SoundAlarmScanTime=2000

12.3.2 Common Items for Alarm Service

The items required for the line printer output, sending out mails and SMS are performed through common and individual items of [InfoU project folder]- [Config folder] –[iuAlarm.ini file].

(1) AS_CONFIG section

Through AS_CONFIG section, you can specify the common and individual items to perform printer, mail and SMS functions.

Items	Description
PrinterColSize	It specifies the line printer’s column length (printer).
ValuePrecision	It specifies the decimal points when using the value (common).

(2) Column specified items (common)

They are the basic items to specify columns for services.

The rule of “column name=length” is applied.

Column definition	Description	Remarks
TAG_NAME	Tag name	
ALM_TIME	Time	When the columns are specified for mail and SMS functions, enter the time of alarm occurrence. For the printer, enter the occurrence/recovery time according to the occurrence/recovery. [Format] YYYY/MM/DD HH24:MI:SS.MMM 2013/05/31 11:19:46.971
ALM_TYPE	Alarm type	PROCESS / SYSTEM / EVENT
ALM_AREA	Alarm group	-
TAG_TYPE	Tag type	ANALOG / DIGITAL
ALM_KIND	Alarm kind	HH / HI / LO / LL Change Val, Main Gap / Sub Gap OFF / ON / ON→OFF / OFF→ON / CHANGE
ALM_LEVEL	Alarm level	-
ALM_CURVAL	Current value	-
ALM_PREVVAL	Previous value	-
ALM_MSG	Message	Occurring string / Recovery string
ALM_INFO	Tag information	-
UNIT_ONOFF	Unit ON/OFF STRING	In the case of the analog tat, unit is output. In the case of the digital tag, ON/OFF string is output.
USER_DEF1	User definition 1	-
USER_DEF2	User definition 2	-
USER_DEF3	User definition 3	-

12.3.3 Automatic Printout of the Line Printer

It is the function to print out the items automatically with the registered line printer when an alarm occurs and is cleared. A user can perform the desired functions by specifying the item and length to print out and by editing [InfoU project folder]- [Config folder] –[iuAlarm.ini file]. Then, you can specify the column and length using [AS_COLLIST_PRINTER] section of the ini file. For using column, refer to the column specified items (common).

Example of Use) [AS_COLLIST_PRINTER] section of the iuAlarm.ini file

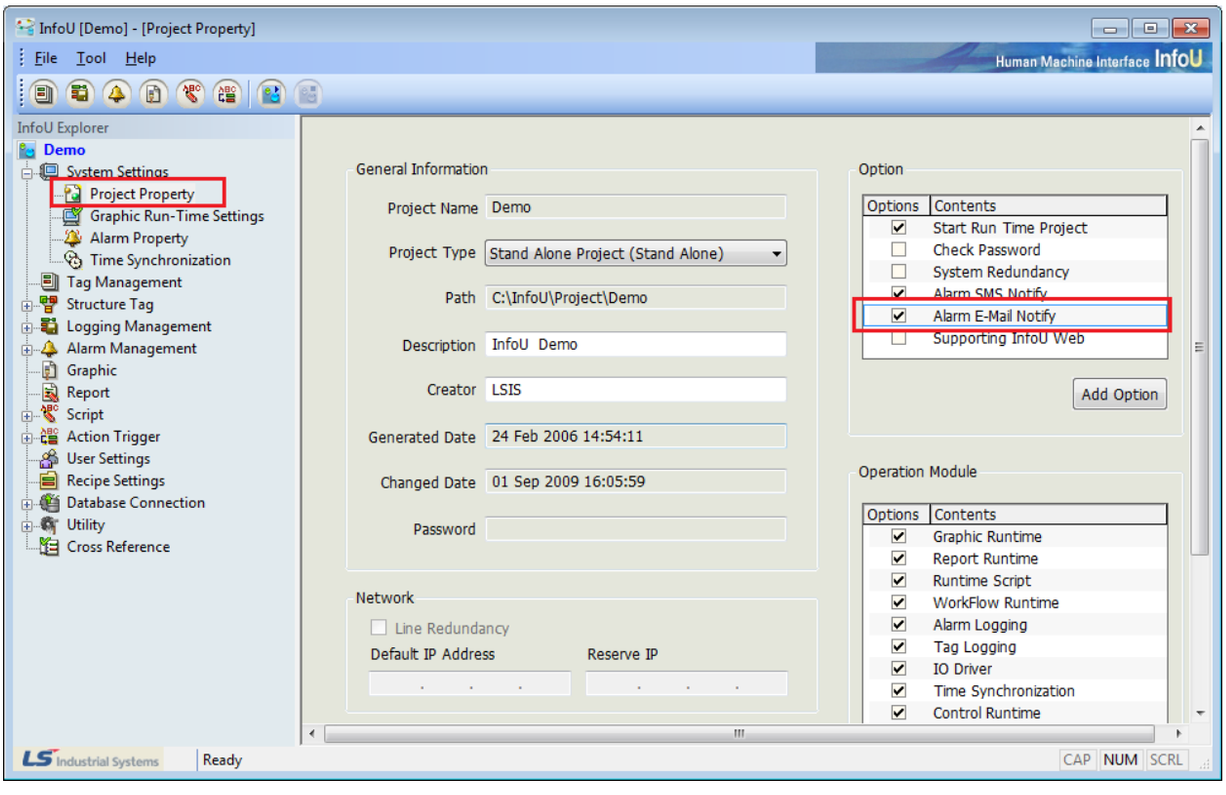
```
[AS_COLLIST_PRINTER]
ALM_TIME=25
ALM_TYPE=10
TAG_NAME=30
ALM_KIND=10
TAG_TYPE=10
ALM_CURVAL=20
ALM_MSG=20
UNIT_ONIFF=10
```

12.3.4 Mail Sending

It is the function to send out the alarm details to a user's e-mail account, when the alarm occurs in the nfoU. The InfoU sends out an e-mail to users through the external mail server (SMTP server). This section describes how to set up the items to use the mail sending function stage by stage.

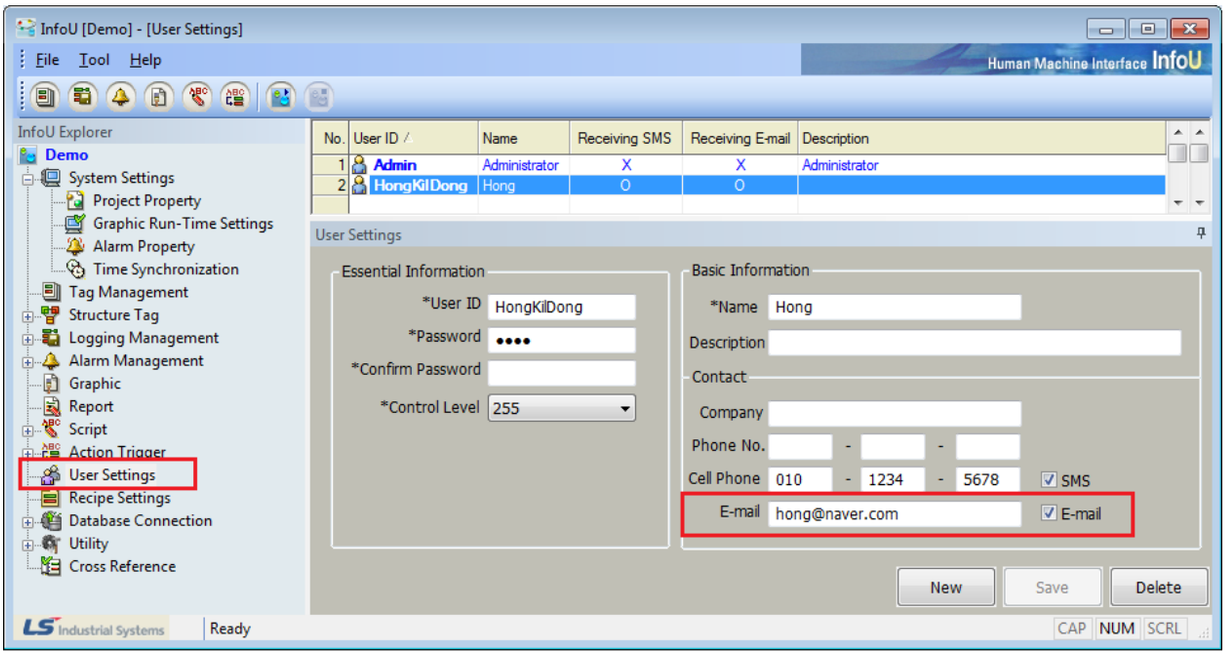
(1) Project option settings

Select the [Alarm E-Mail Notify] option in [System Settings]- [Project Property] of the InfoU engineering.



(2) User Settings

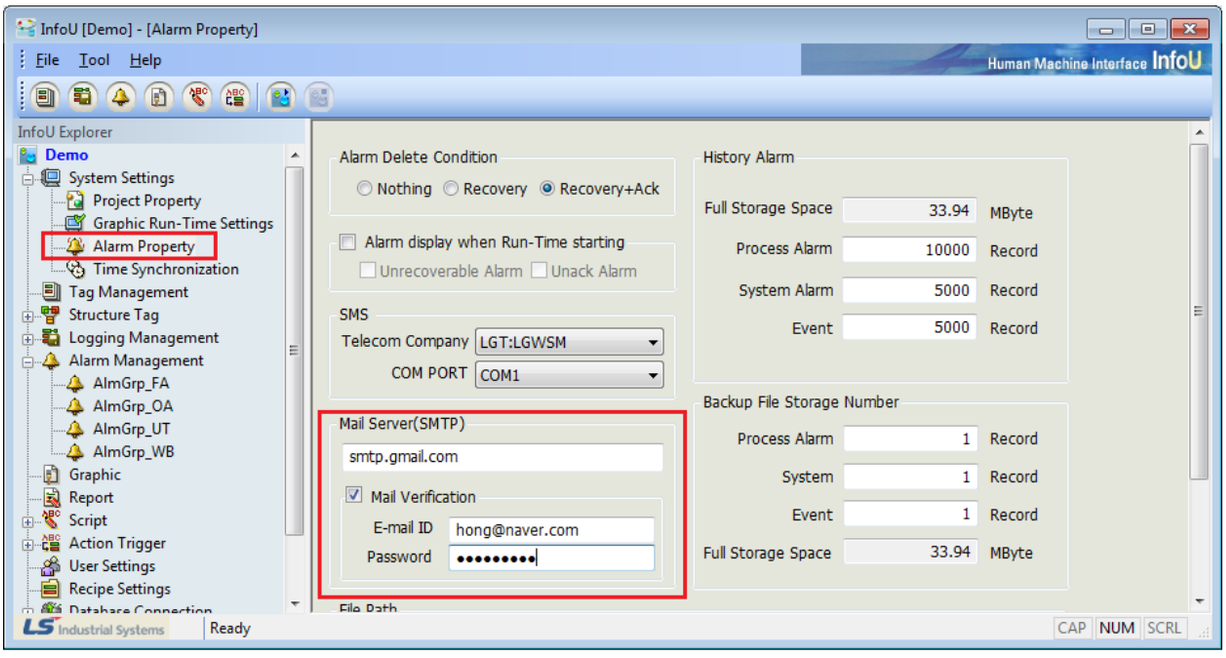
After registering the user in [User Settings] of the InfoU engineering, put a check the e-mail and reception columns.



(3) Alarm property settings

You can set up the following items in [System Settings] → [Alarm Property] of the InfoU engineering.

Item	Description
Mail server	Enter the mail server address. Ex.) Google : smtp.gmail.com
Mail Verification	If authentication is required, put a check Mail Verification.
Mail ID	Enter the ID of a mail account.
Password	Enter the password of a mail account.



Notice

The InfoU adopts the external mail server to use mail function. The setting items of mail server means the foresaid mail server. (Not the final destination of the mail)

☞ As shown in the fig., if you set the SMTP server as smtp.gmail.com, the flow of the mail data is as below.

Occurrence of InfoU alarm	➔	Via the mail server (smtp.gmail.com) (Using the account hong@gmail.com)	➔	Arriving at the destination (user E-mail) (hong@naver.com)

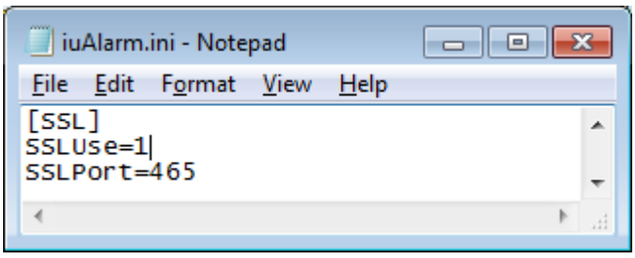
(4) SSL Settings

The majority of the previous SMTP servers were confined to simple SMTP mail servers without the security function. However, recently, the importance of security has been strengthened so the existing simple SMTP mail servers gradually disappear. In the InfoU, you can set the mail server using the SSL(Secured Socket Layer). To set up the SSL mail server, you can perform the function by editing [InfoU project folder]- [Config folder] –[iuAlarm.ini file] (SSL section).

[SSL Section]

Item	Description
SSLUse	To use the SSL function, the value should be set as 1.
SSLPort	Specify the port used for the SSL mail server. (Ex.) For Google mail server: 465

[Example of Use]

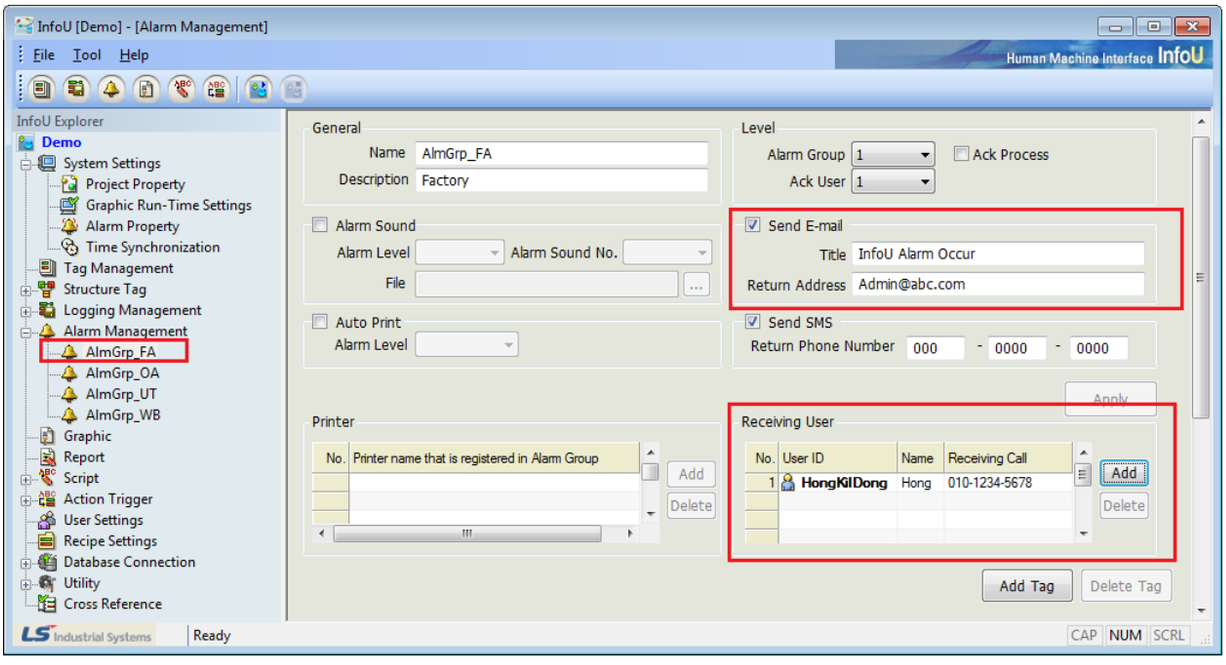


(5) Alarm group settings

Alarm group settings related to mail transmission are as follows.

For the overall settings of other alarm groups, refer to the items of [Alarm Group Settings].

Item	Description
Send E-mail (check)	When an alarm occurs, put a check this to use the mail transmission function.
Title	Set up the mail title.
Return Address	Set up the address of the mail to be sent out.
Receiving user	When the alarm occurs in the tag registered in the alarm group, you need to register the user who will receive the mail. It should be preregistered in User Settings.

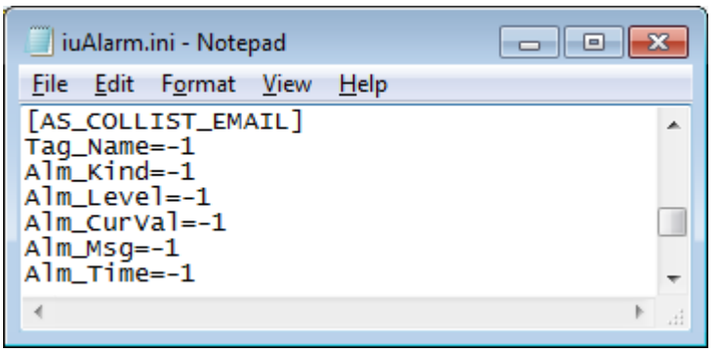


(6) Mail tag item settings

When an alarm occurs, the tag items received by a mail are similar to those of a printer.

You can choose the desired item by editing [InfoU project folder]- [Config folder] –[iuAlarm.ini file]. Then, specify the column and length through the [AS_COLLIST_EMAIL] section of the Ini file. For using column, refer to the column specified items (common).

```
[AS_COLLIST_EMAIL]
TAG_NAME =-1
ALM_LEVEL=-1
ALM_LEVEL=-1
ALM_CURVAL=-1
```



Notice

☞ When setting the column length as -1, the whole length of the tag column data is used instead of being cut in the specified length.

☞ Unlike a printer or SMS with the limit on the whole length, an e-mail has no limit so you can input -1.

12.3.5 SMS Transmission

It is the function to send a text message to a user's cell phone through the external CDMA modem when an alarm occurs.

The InfoU adopts BSM-856R model that is the external CDMA modem made by Telit wireless solutions.

After purchasing the modem, you need to enroll in the telecom companies to start up the service. Then, you can perform SMS functions only when the modem is connected to a computer through RS-232C



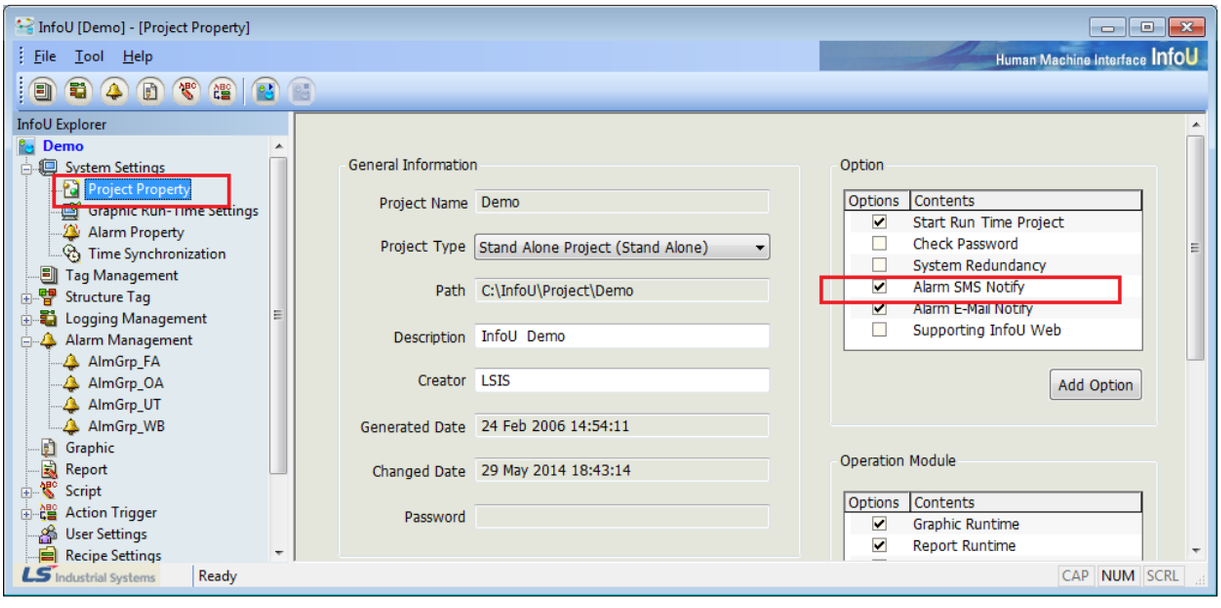
[BSM-856R]

LSIS does not sell this device directly so for more details on purchasing the modem and starting services, refer to the manufacturer's website.

This section describes how to set up SMS functions stage by stage on the guess that your external CDMA modem is normally opened and connected to a computer.

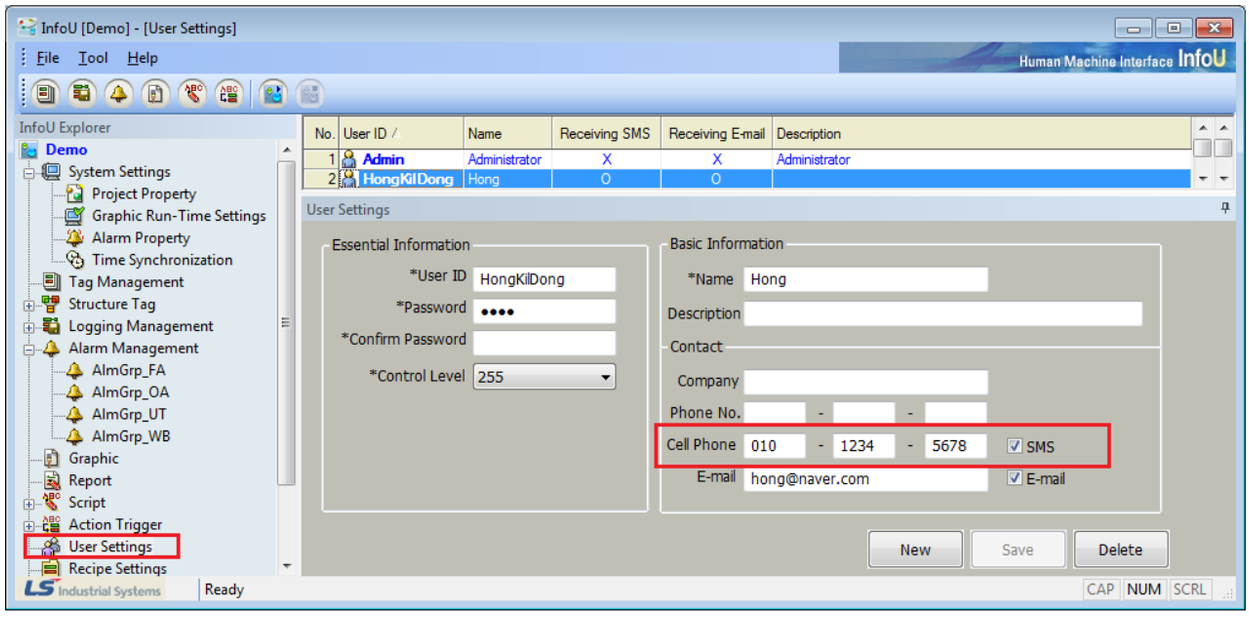
(1) Project Option Settings

Select the [Alarm SMS Notify] option in [System Settings]- [Project property] of the InfoU engineering.



(2) User Settings

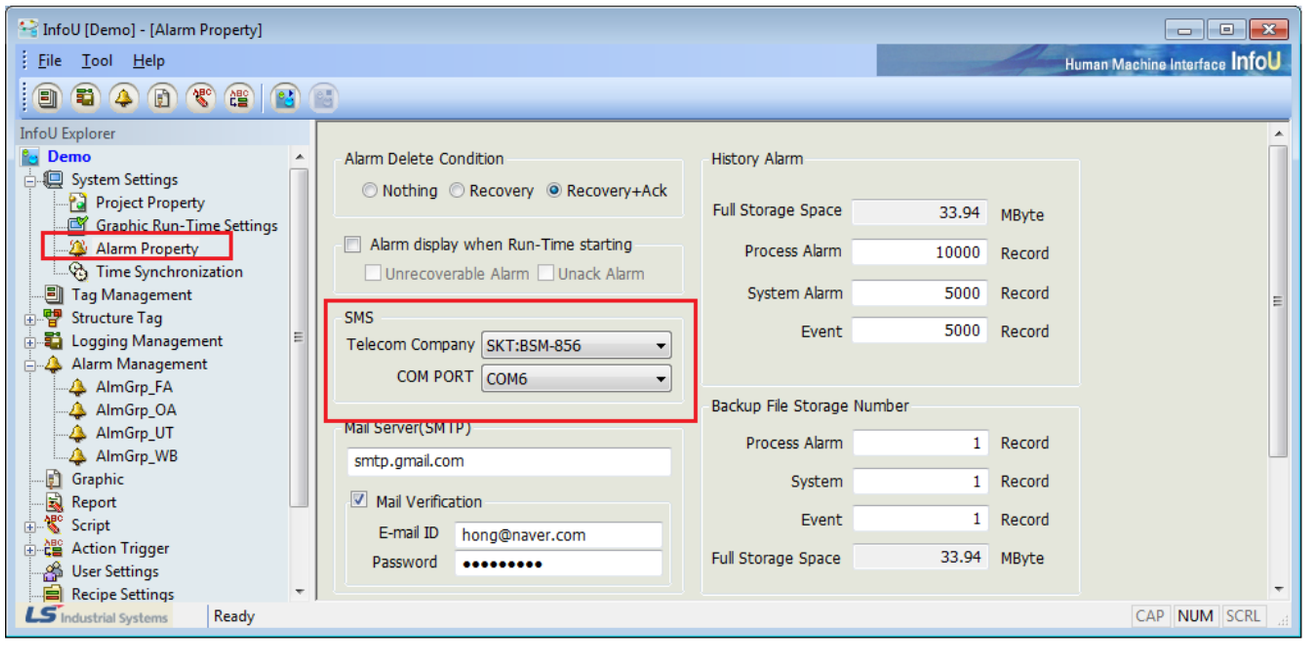
After registering the user in [User Settings] of the InfoU engineering, put a check the cell phone and SMS.



(3) Alarm Property of System Settings

You can set up the following items in [System Settings] → [Alarm Property] of the InfoU engineering.

Item	Description
Telecom Company	SKT:BSM-856
COM PORT	Select the communication port connected to the external CDMA modem.



Notice

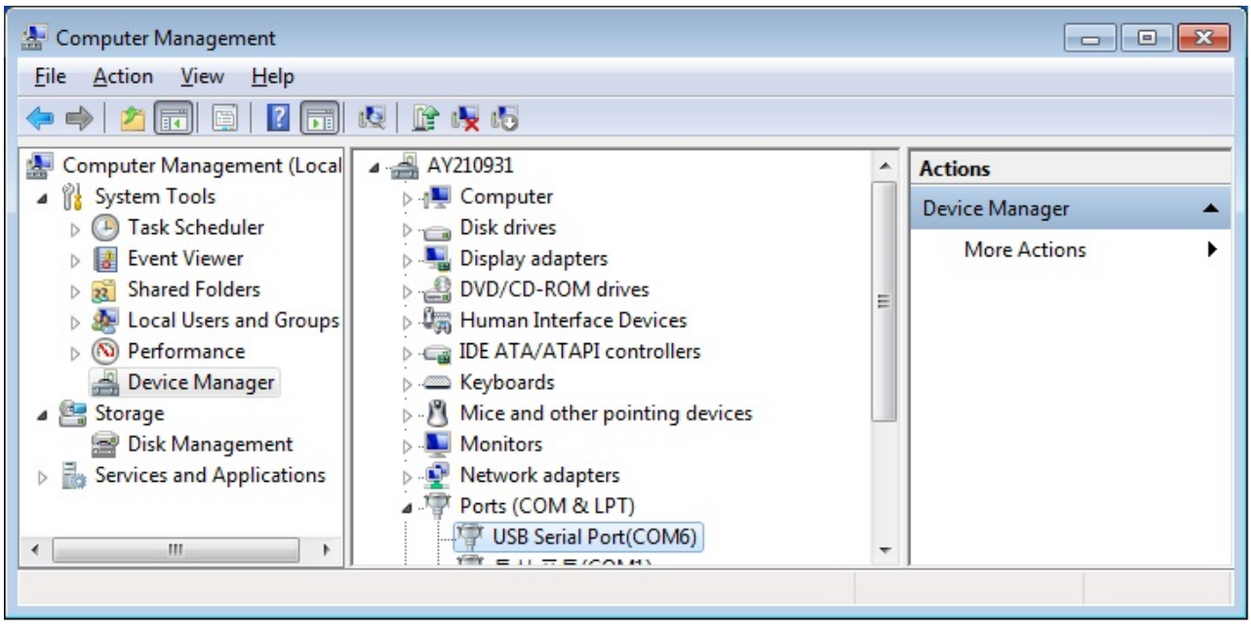
Recently, most of the computers do not have a serial port.

In this case, you can set up the serial port function using the USB port.

After purchasing USB and connecting it to a serial cable, if you install the driver, the system will work normally.

Then, the COM PORT No. will be registered and displayed in the device manager.

In the below fig., select [COM6] for InfoU settings.

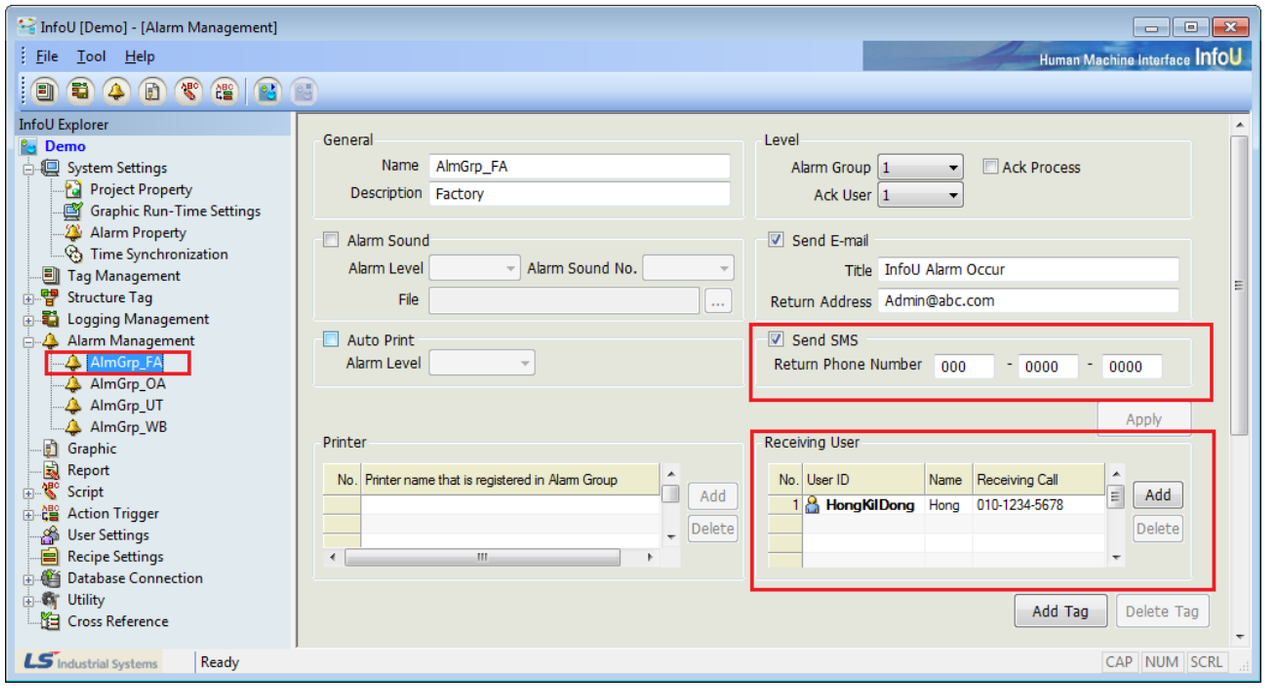


(4) Alarm Group Settings

Alarm group settings related to SMS transmission are as follows.

For the overall settings of other alarm groups, refer to the items of [Alarm Group Settings].

Item	Description
Send SMS(check)	In order to use the SMS transmission function when an alarm occurs, put a check this.
Return Phone Number	Set the return phone number.
Receiving user	When the alarm occurs in the tag registered in the alarm group, you need to register the user who will receive the SMS. It should be preregistered in User Settings.



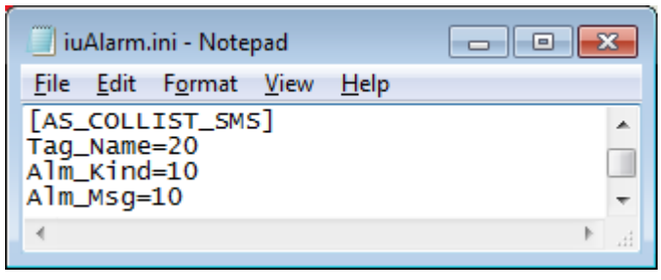
(5) SMS Tag Item Settings

When an alarm occurs, the tag items forming SMS are similar to those of a printer.

You can choose the desired item by editing [InfoU project folder]- [Config folder] –[iuAlarm.ini file]. Then, specify the column and length through the [AS_COLLIST_SMS] section of the Ini file. However, in the case of SMS (Short Message Service), the length of contents is limited so you need to specify the proper size of each item. For using column, refer to the column specified items (common).

Example of Use) [AS_COLLIST_SMS] section of the iuAlarm.ini file

```
[AS_COLLIST_SMS]
TAG_NAME=20
ALM_KIND=10
ALM_MSG=10
```



Chapter 13 Logging Management

Important data needed for logging management can be managed under the data logging group created in accordance with the user's purpose.
It is possible to assign one tag data to multiple logging groups for diverse data analyses.

● Inquiry of Logging Data
Data can be inquired from Trend, Report or List Trend, and API, which can be inquired from script, is provided.
● Automatically calculated statistics/integrated data
Statistics/integrated data are created without the use of additional scripts or applications.
● Backup
Backup is available for all of the logging groups and backup files can be inquired from Trend and List Trend.

13.1 Logging Type

The logging group editor sets up a logging group that the history server will collect during the Runtime and various logging data are acquired and saved based on the logging group.

13.1.1 Logging Type

Logging groups can be divided into Cycle, Statistics, Accumulation, String, Event, Selective Log and Event String groups depending on their type and they are acquired and saved in either periodic or event way.

Logging Group Type	Description
Cycle	Tag values are acquired in a certain cycle.
Statictics	Statistics data on the designated tag are created periodically.
Accumulation	Statistics data on the accumulation values of the tag in which values have been accumulated are created periodically.
String	String tag values are acquired in a certain cycle.
Event	Tag values are acquired in an event way according to event conditions.
Selective Log	Tag values are acquired periodically according to event conditions.
Event String	String tag values are acquired in an event way according to event conditions.

The following data can be inquired from Report or Trend according to the type of the logging group.

Logging Group Type	Description
Cycle	Analog (Current Value), Digital (Current Status)
Statictics	Analog (Current Value, Min Value, Min Time, Max Value, Max Time, Average Value), Digital(On Count, Operation Time, Change Count)
Accumulation	Analog (Change Amount, Current Value)
String	String(Current Value)

Event	Analog (Current Value), Digital(Current Status)
Selective Log	Analog (Current Value), Digital (Current Status)
Event String	String(Current Value)

13.1.2 Calculation Method for Logging Data

[HDD capacity calculation method for data collection]

Up to 40 tags can be registered in one logging group and 40 tag values can be saved at the same time with one saving.

Each logging group can save the following amount of data for one time:

- Cycle Logging: 1,048 byte
- Statistics and Accumulation: 3,768 byte
- Stirng Logging: 5,768 byte

Needed disc capacity: Data to be saved per time x logging count

- If it is planned to save data every second for one day under cycle logging,
  1,048 X (60 X 60 X 24) = 90,547,200 byte (about 86.35 MB)

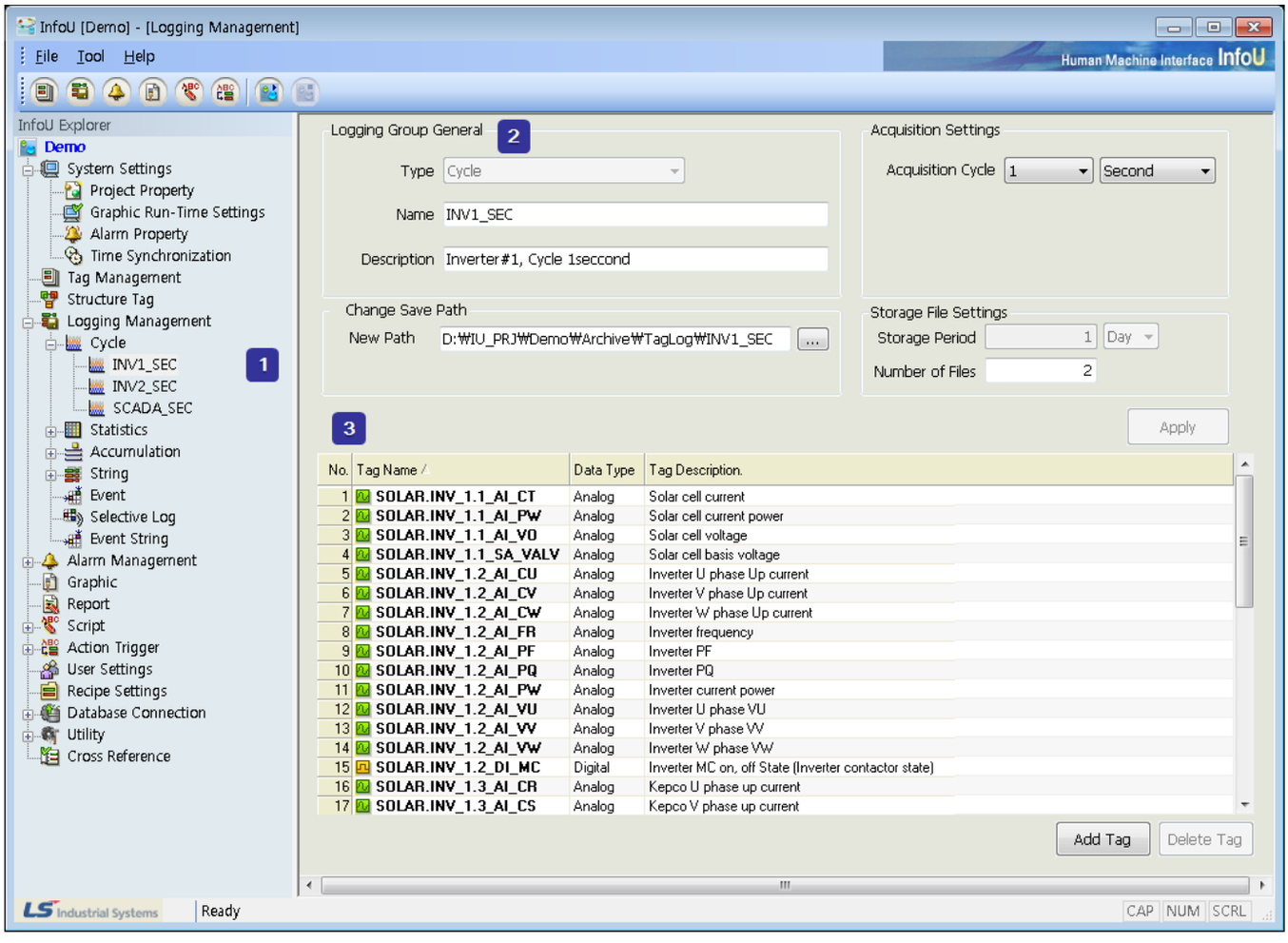
- If it is planned to save statistics data every hour for one year,
  3,768 X (24 X 365) = 33,007,680 byte (about 31.47 MB)

- If it is planned to save string data every minute for one month,
  5,768 X (60 X 24 X 31) = 257,483,520 byte (about 245.55 MB)

13.2 Logging Settings

13.2.1 Configuration Interface

If [Tool] – [Logging Management] is selected from the InfoUD menu, a screen as seen in [Figure] appears.
Or, this screen can be displayed by selecting [Logging Management] from the InfoU explorer.



This screen consists of the following components:

(1) Logging Group Tree View

Logging group items are displayed in the InfoU explorer. The kinds of logging groups and the logging groups set up by the user are displayed in the tree. Logging groups can be newly added, edited or deleted by clicking on the right button of the mouse.

(2) Logging Group Property Settings

Properties of the set up logging group are displayed. Adding and editing logging groups are also available. Acquisition method items are displayed differently depending on the kinds of the logging groups and they can be saved by clicking on “Apply” button.

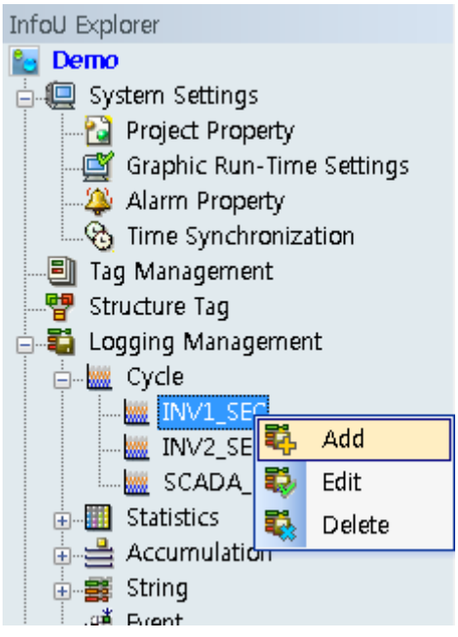
(3) Logging Tag List

Tags registered in the logging group are displayed in a list form.

The acquired tags can be registered or deleted with “Add Tag” or “Delete Tag,”

13.2.2 Add, Edit and Delete

By clicking the right button of the mouse on the “Logging Management” of the InfoU explorer, logging groups can be added, edited or deleted.



(1) Add

A new logging group is added. If “Add” is selected and the initialized right screen is filled in with new information, properties of the newly added logging group can be set up.

The logging group can be saved if “Apply” button is clicked after setting up is completed.

Detailed settings for each logging group type are explained below.

(2) Edit

Properties of the existing logging group are edited. The changed contents are saved if “Apply” button is clicked after property editing is completed.

Also, properties can be edited on the right view screen if the relevant item is selected with the mouse.

※ A new logging group can be added by changing only the name of the existing logging group on the editing screen. At this time, the same property information as that of the previously displayed data will be saved.

(3) Delete

The selected logging group is deleted.

13.2.3 General

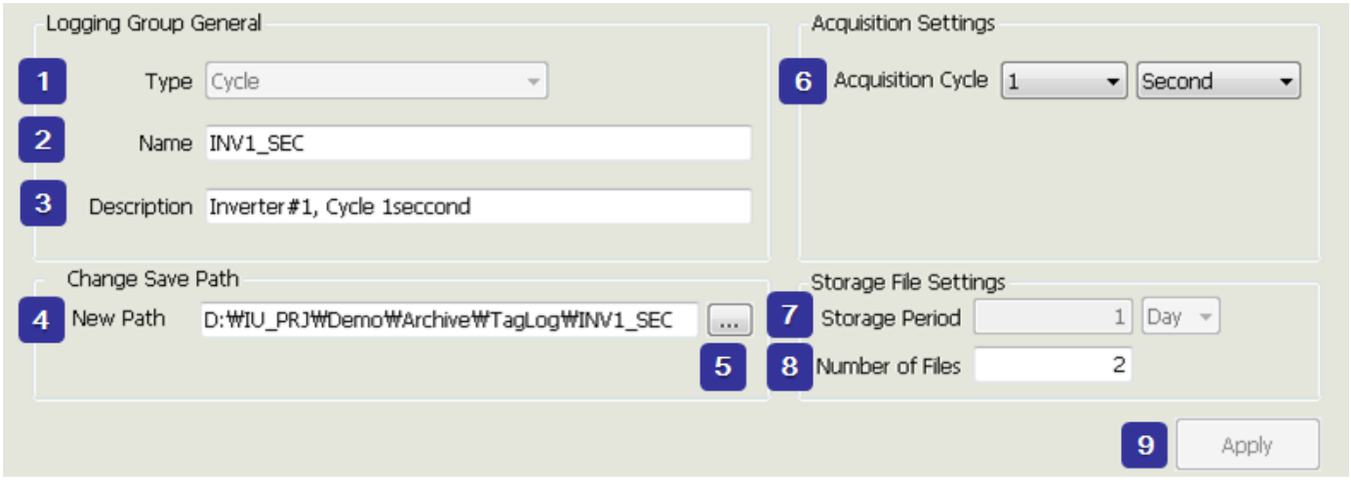
Logging group settings are carried out through the following two steps:

(1) Logging Group Settings

An acquisition method is set up along with logging group type, logging group name and logging group description, and storage and backup of logging groups are also set up. Up to 256 logging groups can be set up in a project.

Especially, a sufficient space shall be secured in the hard disc when setting backup.

Acquisition methods are displayed on different setting screens depending on the type of logging groups.



1) Logging Group Type: Select one among Cycle, Accumulation and String.
2) Logging Group Name: Input a name to identify the relevant logging group. Up to 31 letters are available to input.
3) Logging Group Description: Input description on the logging group. Up to 63 letters are available to input.
4) Saving path: It is used to change the saving path. When this is empty, the file is automatically saved to the default path ("Project/TagLog/LoggingGroupName") so it is recommended not to set this item.
5) Path Selection Button: Click the button to display the path selection dial log.
6) Acquisition Cycle: Select a cycle to acquire data. Acquisition cycle may set up differently depending on the type of the logging group.
7) Storage Period: Set up a storage period of the history file. For storage period, select one between number 1 and 100 and for unit, select one among time, day and month.
8) Number of files: The logging files are created as you enter and if the number exceeds the limit, the file will be deleted in order of age.
9) Apply: The inputted information on the logging group is saved.

The storage period of the file depending on the archiving cycle is as below.

Archiving cycle	Storage period	Number of files	Max. of storage period
1 ~ 30sec	1day	1 ~ 90	90day
1 ~ 30min	1month	1 ~ 36	3year
1 ~ 12hour	1year	1 ~ 3	3year
1day	1year	1 ~ 3	3year

Notice

☞ Storage period settings

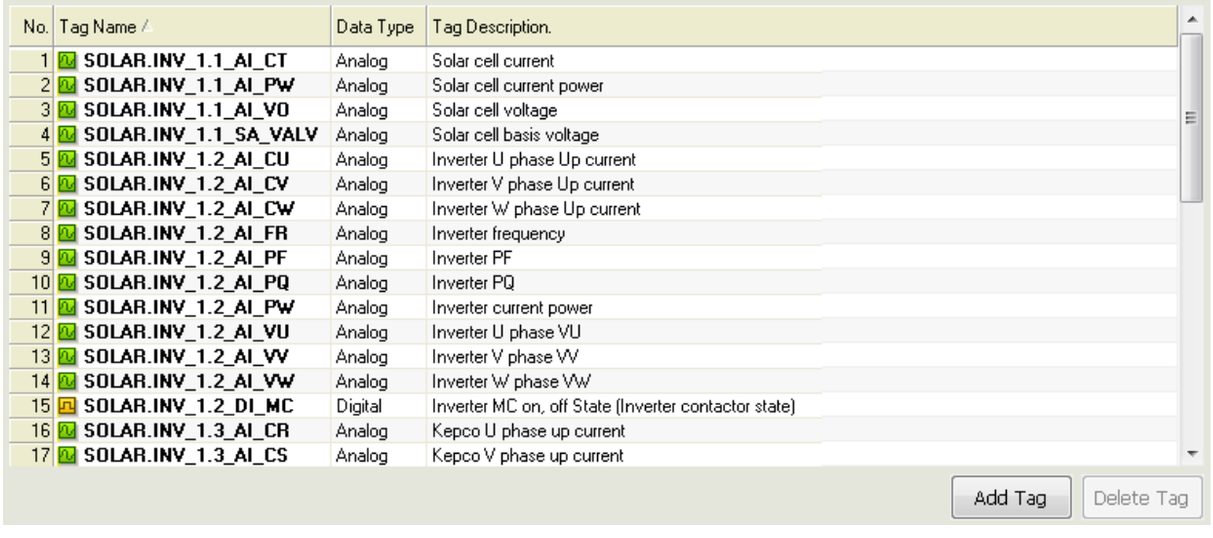
The “storage period” is fixed in accordance with the "archiving cycle" of the logging group.
Instead, you can set to create multiple logging files and the files are automatically connected to query.

To extend the storage period, you just set the “number of files” as long as you want.
Accordingly, the total logging period is “storage period” X “number of files”.
The logging file is created one by one as many as the number of files set during logging and if the number exceeds the limit, the oldest logging file will be deleted to maintain the number of files.

The default storage path of the file is Project/TagLog/Logging group name (when the “saving path” is a blank).
If you input the “saving path”, the file will be saved to the specified location.

(2) Tag Settings

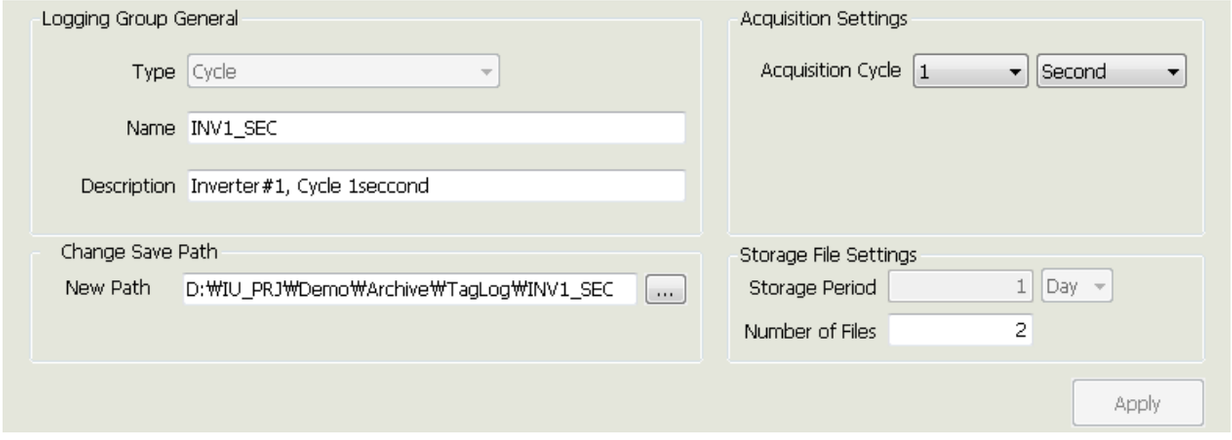
Once adding logging groups is completed, select a tag that will acquire logging data.
One logging group can save up to 40 tags.



1) Add Tag: Select a tag to save from the tag selector.
2) Delete Tag: Delete the selected tag.

13.2.4 Cycle

The minimum acquisition cycle is one second so that this cycle is suitable for fast real-time data logging and it is mainly used for inquiring sequential values from Trend or Report.
The current value (analog) and the current status (digital) can be inquired.



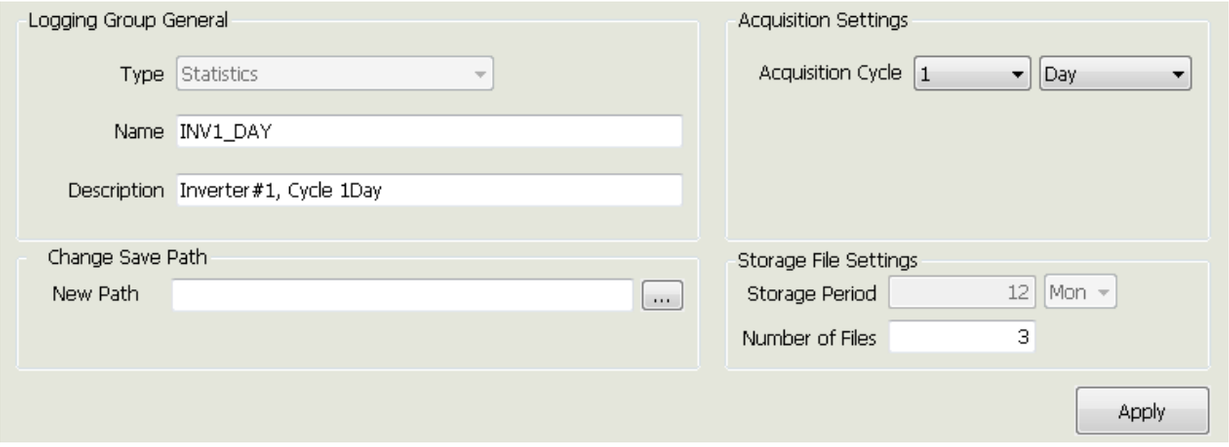
During the runtime, cycle logging groups are acquired and saved on time.
Ex) If the acquisition cycle is set 2 seconds, logging data are saved every 0, 2, 4, , …, seconds.

Acquisition Settings: Select a cycle and unit. The following cycles can be selected for each unit:

Unit	Available cycles
sec	1, 2, 3, 4, 5, 6, 10, 12, 15, 20, 30
min	1, 2, 3, 4, 5, 6, 10, 12, 15, 20, 30
hour	1, 2, 3, 4, 6, 8, 12
day	1

13.2.5 Statistics

The minimum acquisition cycle is one second so that this logging group type can automatically calculate and save various statistics data.
It is used to identify data status for a certain period and mainly used for inquiring statistics status from Report or Trend.
The following data can be inquired.
Analog: current value, min value, min time, max value, max time, average value
Digital: ON count, ON time, change count



During the runtime, statistics logging groups are acquired and saved on time.

Ex) If the acquisition cycle is set 2 minutes, logging data are saved every 0, 2, 4, …, minutes.

Acquisition Settings: Select a cycle and unit. The following cycles can be selected for each unit:

Unit	Available cycles
min	1, 2, 3, 4, 5, 6, 10, 12, 15, 20, 30
hour	1, 2, 3, 4, 6, 8, 12
day	1 ~ 31
month	1 ~ 12

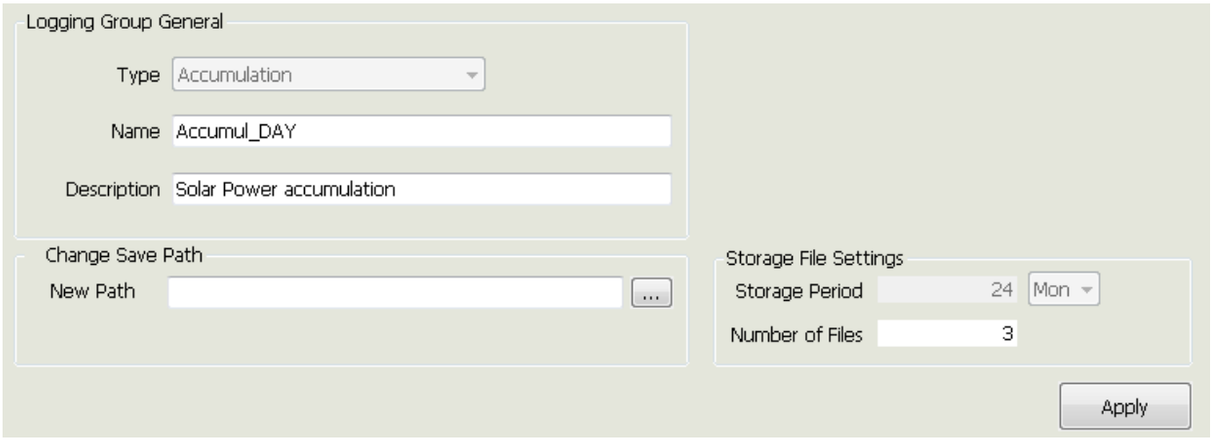
13.2.6 Accumulation

Tags that have accumulation property such as power gauge and fuel indicator are registered and the change amount for a certain period is calculated and saved.

Change amount for every hour, day or month, that is, consumption amount can be inquired without using any additional scrip or memory tag.

The following data can be inquired.

Tag Type	Data
Analog	change amount, current value

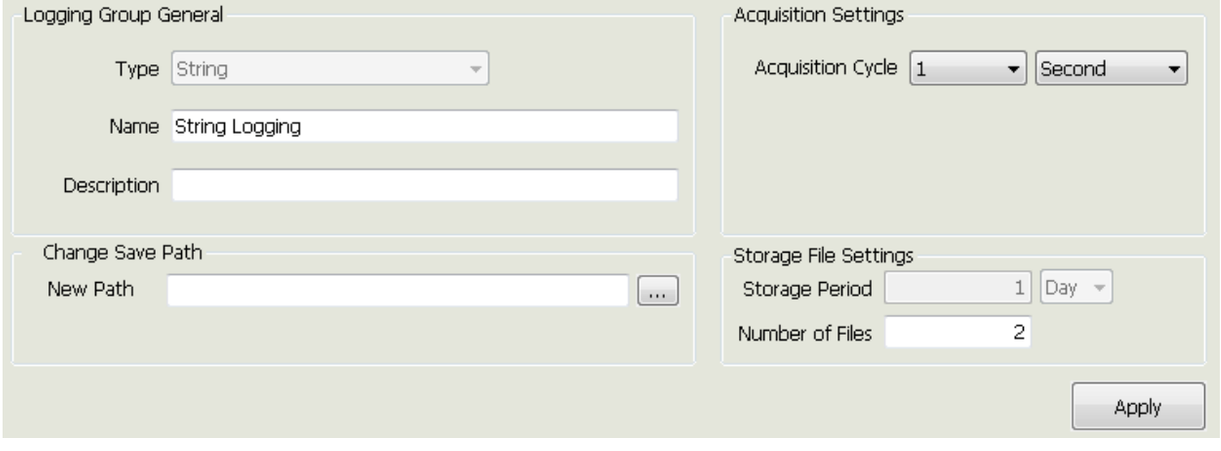


Accumulation statistics data for the accumulation logging group are automatically created every hour, day and month.
The created accumulation logging groups are classified into _HOUR, _DAY and _MONTH.

During the runtime, accumulation logging groups are acquired and saved on time.
Ex) Data are acquired and saved on the hour for hourly logging groups, at 00:00:00 every day for daily logging groups and at 00:00:00 on the first day of every month for monthly logging groups.

※ Only analog tags can be set up as accumulation logging groups.

13.2.7 String



During the runtime, string logging groups are acquired and saved on time.
Ex) If the acquisition cycle is set 2 seconds, logging data are saved every 0, 2, 4, , …, seconds.

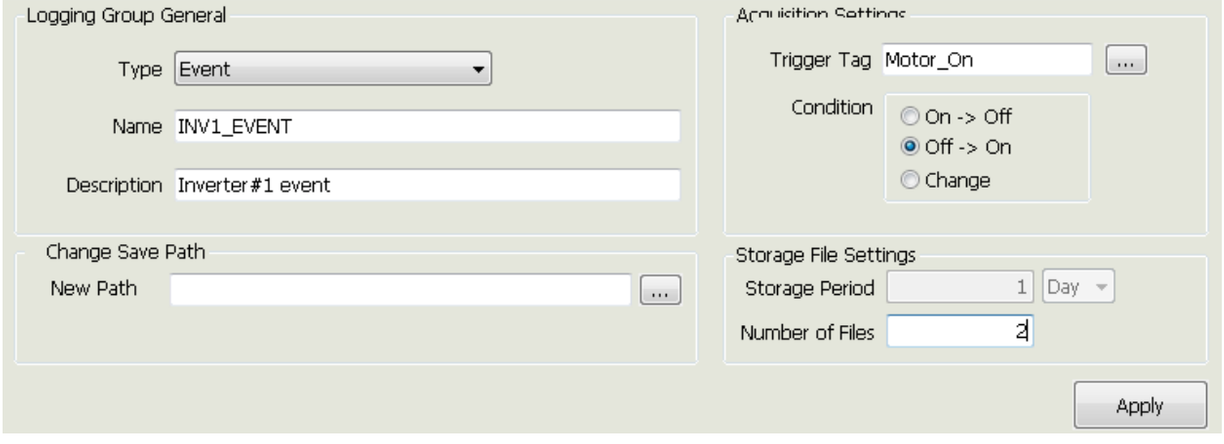
※ Only string tags can be set up as string logging groups.

Acquisition Settings: Select a cycle and unit. The following cycles can be selected for each unit:

Unit	Available cycles
sec	1, 2, 3, 4, 5, 6, 10, 12, 15, 20, 30
min	1, 2, 3, 4, 5, 6, 10, 12, 15, 20, 30
hour	1, 2, 3, 4, 6, 8, 12
day	1

13.2.8 Event

This logging group type is used for saving tag values according to event conditions. Here, changes in digital tag values play a role in constructing event conditions and these digital tags are called trigger tags. Event logging groups save the values of 40 tags that belong to the logging groups according to changes in trigger tags.



Event logging groups are set up as follows:

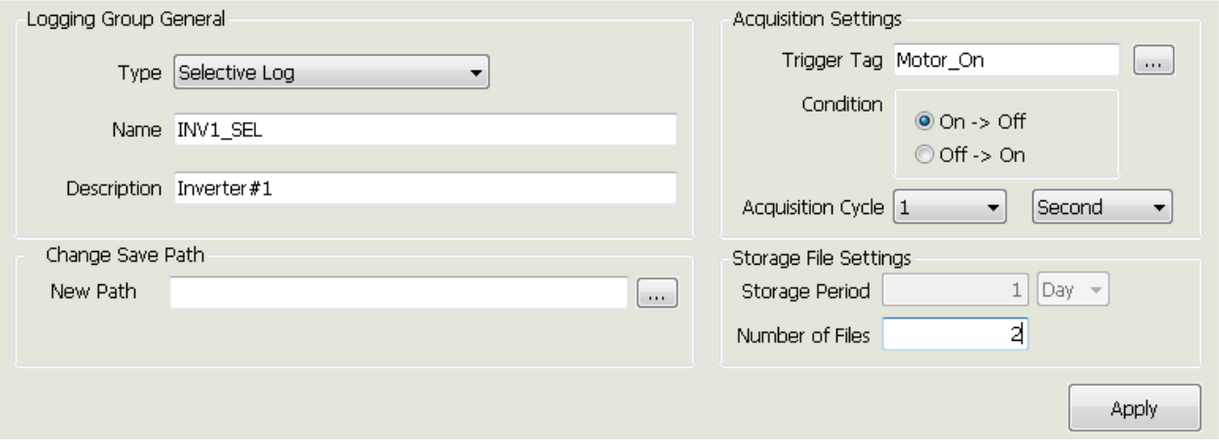
- Trigger Tag: Digital Tag
- Logging: Analog, Digital Tag
- Storage Method: Saved only one time when any event takes place

Event conditions available to be set up in trigger tags are as follows:

- When it is triggered off -> on
- When it is triggered on -> off
- When it is changed on <--> off

13.2.9 Selective Log

It is used to save the tag value periodically depending on the event condition. The event condition means the variation of a digital value that is called the action tag. While the variations of the selective storing logging group and the condition persist, 40 tag values belonging to the logging group are saved periodically.



Event logging groups are set up as follows:

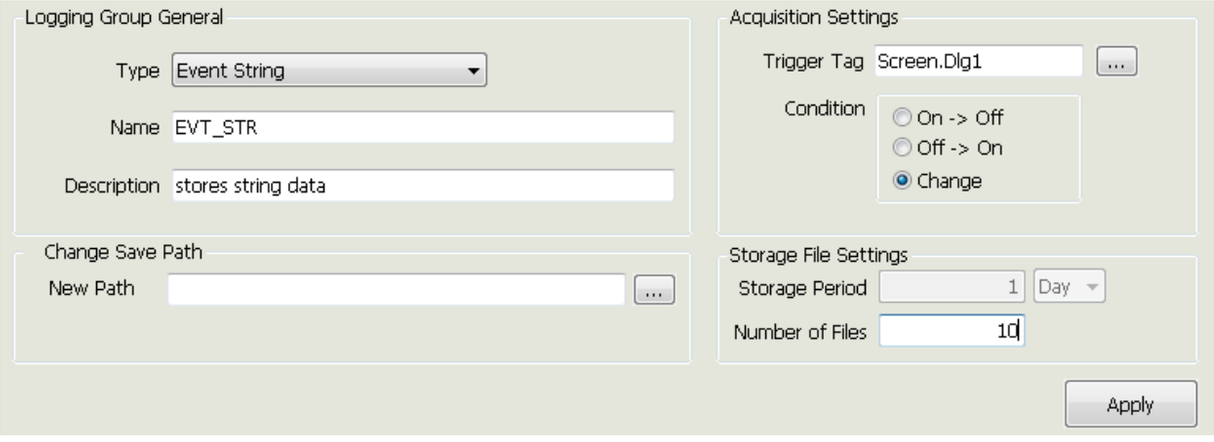
- Trigger Tag: Digital Tag
- Logging: Analog, Digital Tag
- Storage Method: Saved periodically while the conditions of trigger tags remain when any event takes place.
- Storage Cycle: One second ~ one hour

Event conditions available to be set up in trigger tags are as follows:

- Saved periodically when it is triggered off -> on
- Saved periodically when it is triggered on -> off

13.2.10 Event String

It is used to save the string tag value depending on the event condition. The event condition means the variation of a digital value that is called the action tag. In the event logging group, 40 tag values are saved together depending on variations of the action tag.



You can set the event string logging group as below.

- Trigger Tag: Digital tag
- Logging: String tag
- Storage method: Saves only 1 time when event occurs

The configurable event conditions in the action tag are as follows.

- When trigger to Off → On
- When trigger to On → Off
- When changes to On ←→ Off

Chapter 14 Script

The user can implement various functions he/she needs by writing programs freely.

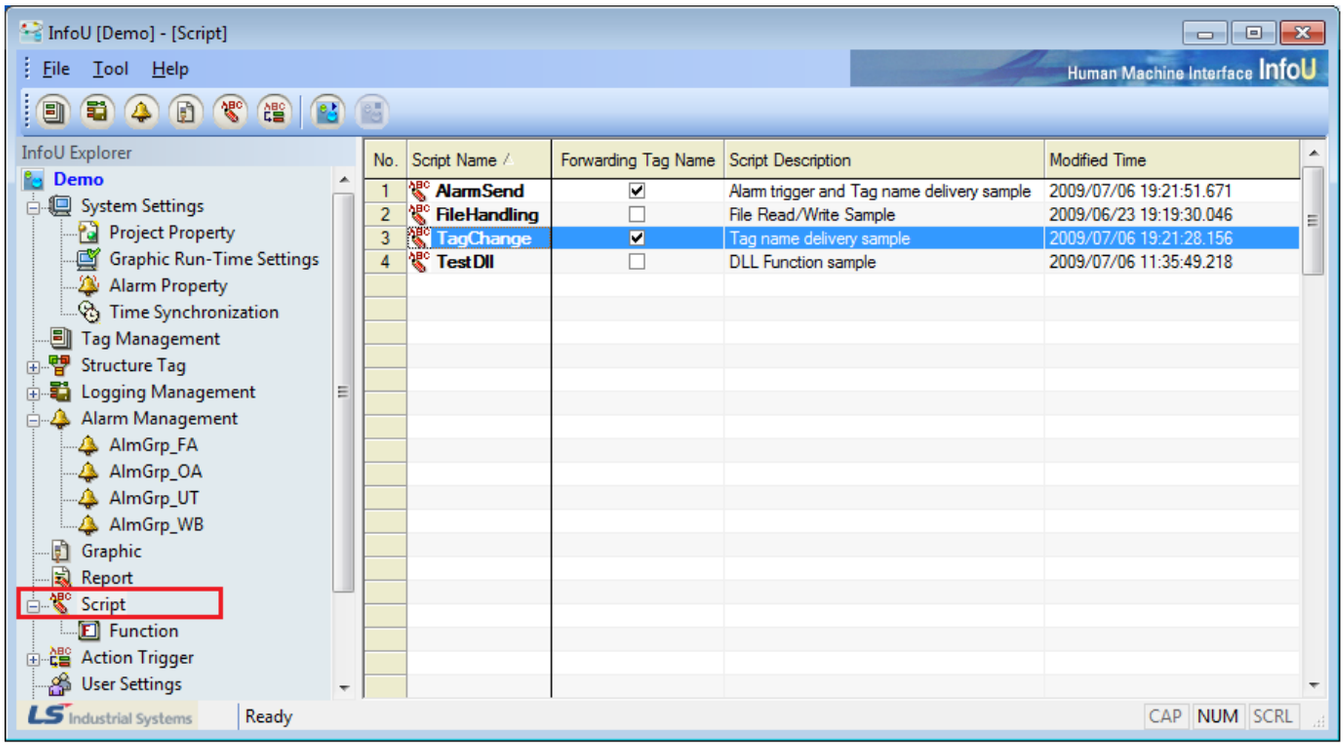
Since this program is of the Visual Basic Script Type, the user can implement various functions by directly using the internal functions provided in a type of OLE in addition to general functions provided by the Visual Basic Script:

- VB Script support
- COM, OLE Object Import
- VB Style Text Color and Dot Operation support
- Script grammar error check

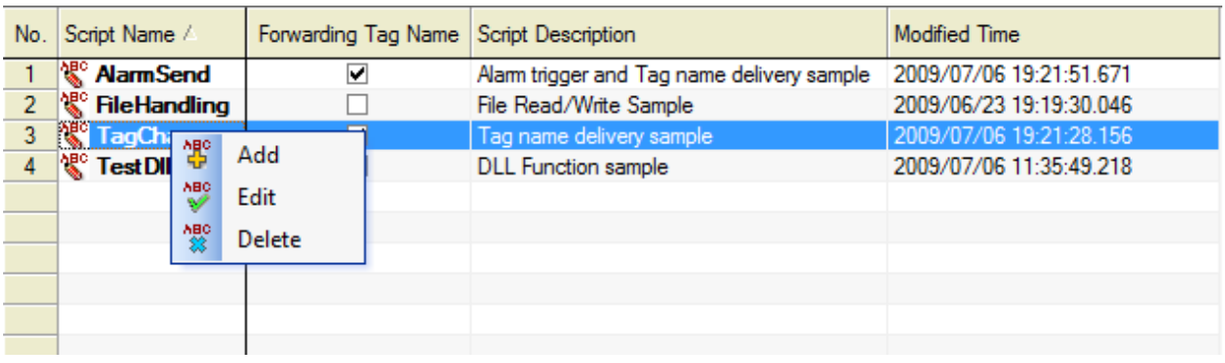
The registered script is implemented during the runtime according to the Trigger rule defined in the 'Action Trigger.'

14.1 Start

If “Edit Script” from menu or “Script” from the InfoU explorer is selected, the following script list is displayed.



Click the right button of the mouse to show Add, Edit and Delete menu.



- Add : A new script is added. Double click on the empty item, the new input screen is displayed.
- Edit : The selected script is edited. If the item is double clicked, the edit screen is displayed
- Delete : The selected script is deleted.

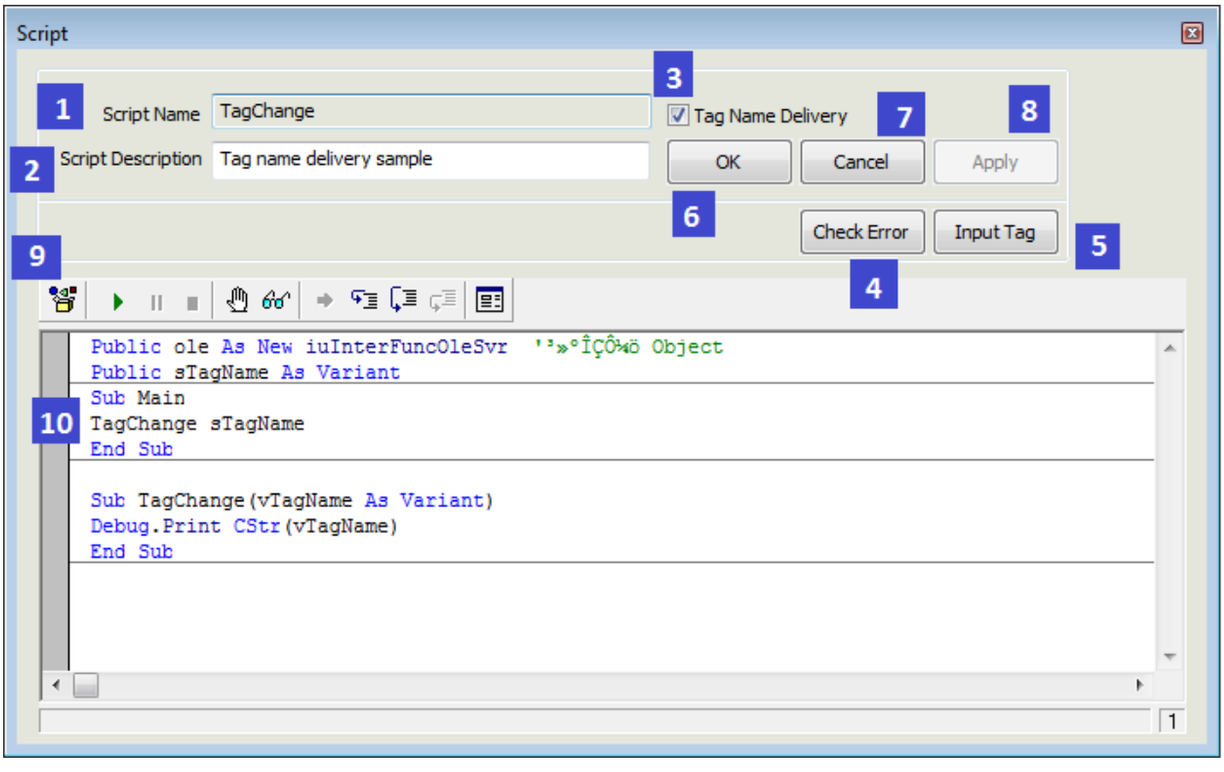
14.2 Editor Screen

The script editor screen is of Modaless dialog type and it is constructed as follow:

If other item is selected from the above script list while the screen is displayed, information on the selected script is displayed.

Except the following cases, script modifications along with other changes during the runtime are all reflected at the same time.

- When a script is added as a new one
- When the option 'Tag Name Deliver' is changed
- When the 'Action Trigger' is redefined



(1) Script Name

- The script name is displayed.
- If the script is renamed and saved, a new script is created and added automatically.
- Any two scripts shall not have the same name.
- Any script name can not be changed when the function option ‘Tag Name Deliver’ is selected.

(2) Script Description

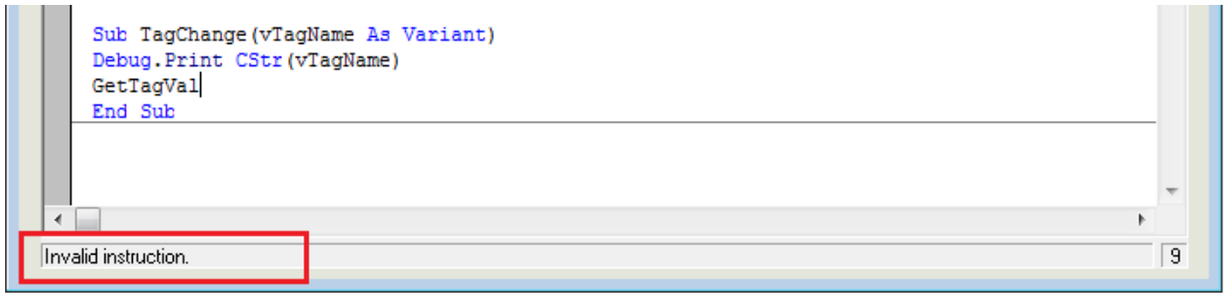
- Input description on the script.

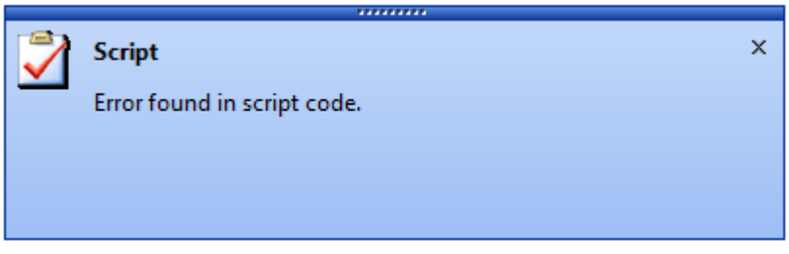
(3) Tag Name Delivery

- Tag names can be delivered in script functions during the runtime.
- Be cautious that the contents that are still written is initialized and the mode is changed if this Tag Name Delivery is selected.
- The tag name shall be the same with the one used when any “action upon the alarm status” or “action upon the change in tag values” takes place.
- For details, refer to ‘Hot to use script.

(4) Check Error

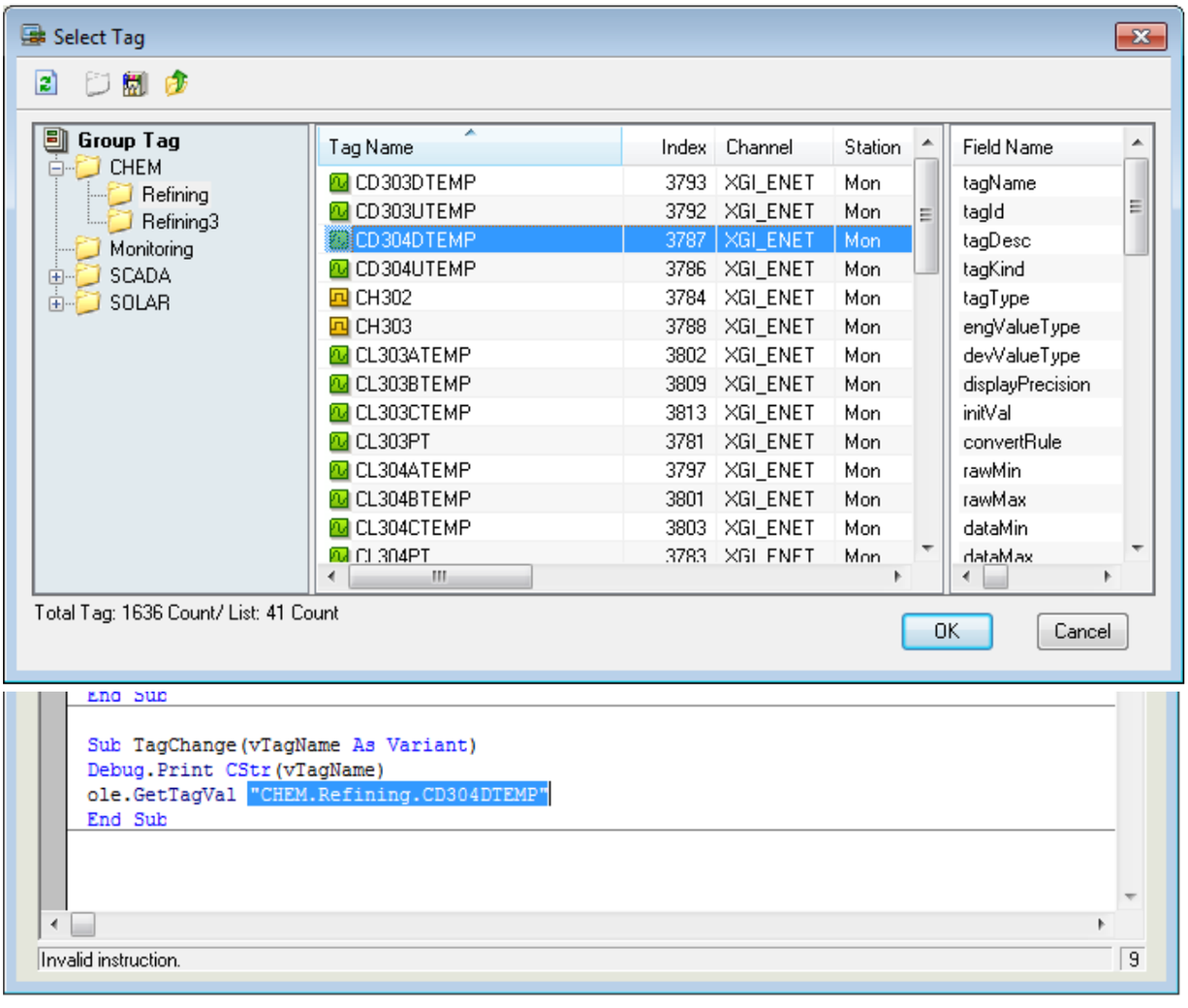
- The script that is still written is checked for grammar errors.
- Only grammar errors are detected and neither logical errors nor error data during the runtime can detected.
- If any error is detected, it is displayed on the ‘Script Error Message window’ and the error type and the line on which the error is detected are displayed in the status bar on the bottom of the editor.





(5) Input Tag

● The tag list of the current project is displayed and if a tag is selected, the relevant tag name is inputted on the editor window.



(6) OK

●The script is checked for grammar errors and if any error is not found, the script is saved and the screen is ended.

(7) Cancel

● Any modification is cancelled and the screen is ended.

(8) Apply

● The script is checked for grammar errors and if any error is not found, what has been edited is saved.

(9) Editor Toolbar

● The script is implemented on the editor screen and debugging work is also available. For details, refer to 'How to use.'

14.3 How to Use

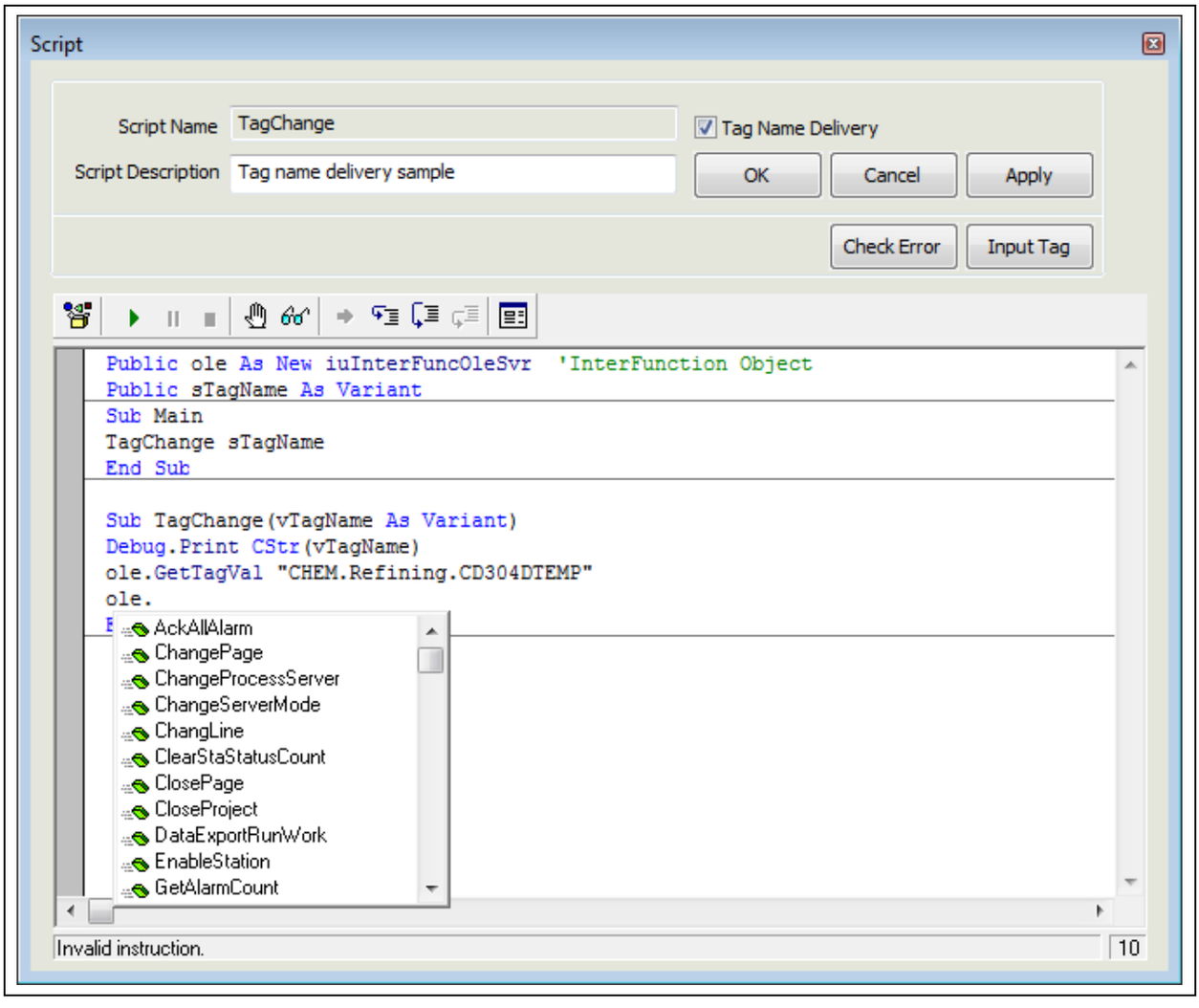
This explains how to register and use the script.

14.3.1 Use of Internal Fuctions

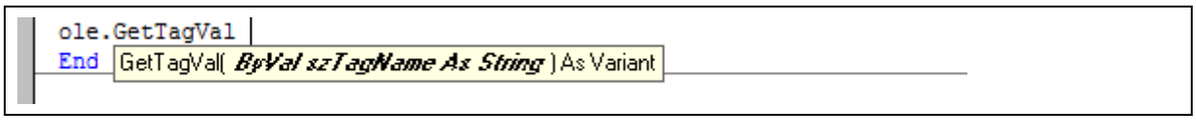
● Declare iuInterFuncOleSvr objects as follows and assign them by using Set command.

```
Public ole As New iuInterFuncOleSvr  'InterFunction Object
```

● If "."(Dot) is inputted in a declared object variant, available functions are displayed upon the Dot Operation.

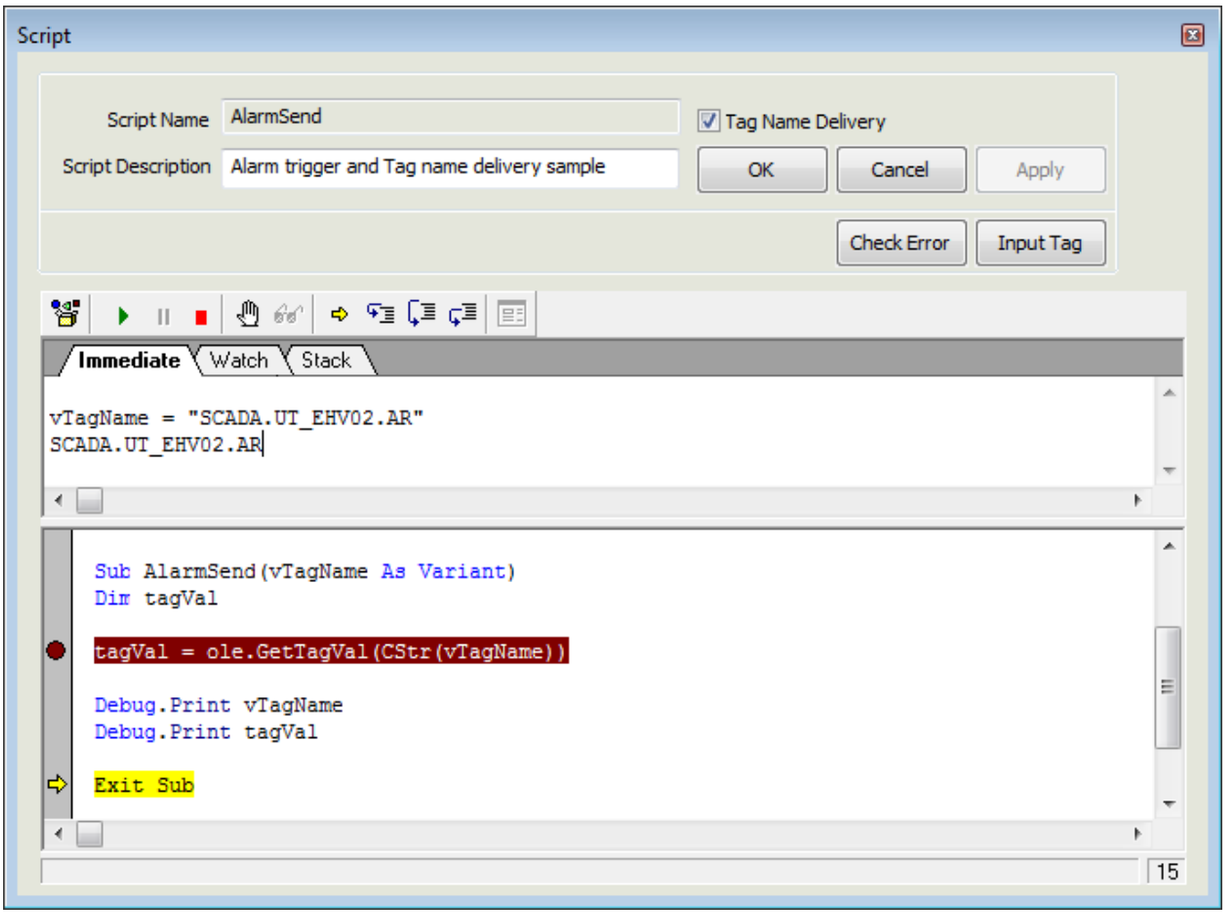


● Since even the function parameter as seen in the figure below is displayed, it ensures convenient inputting.



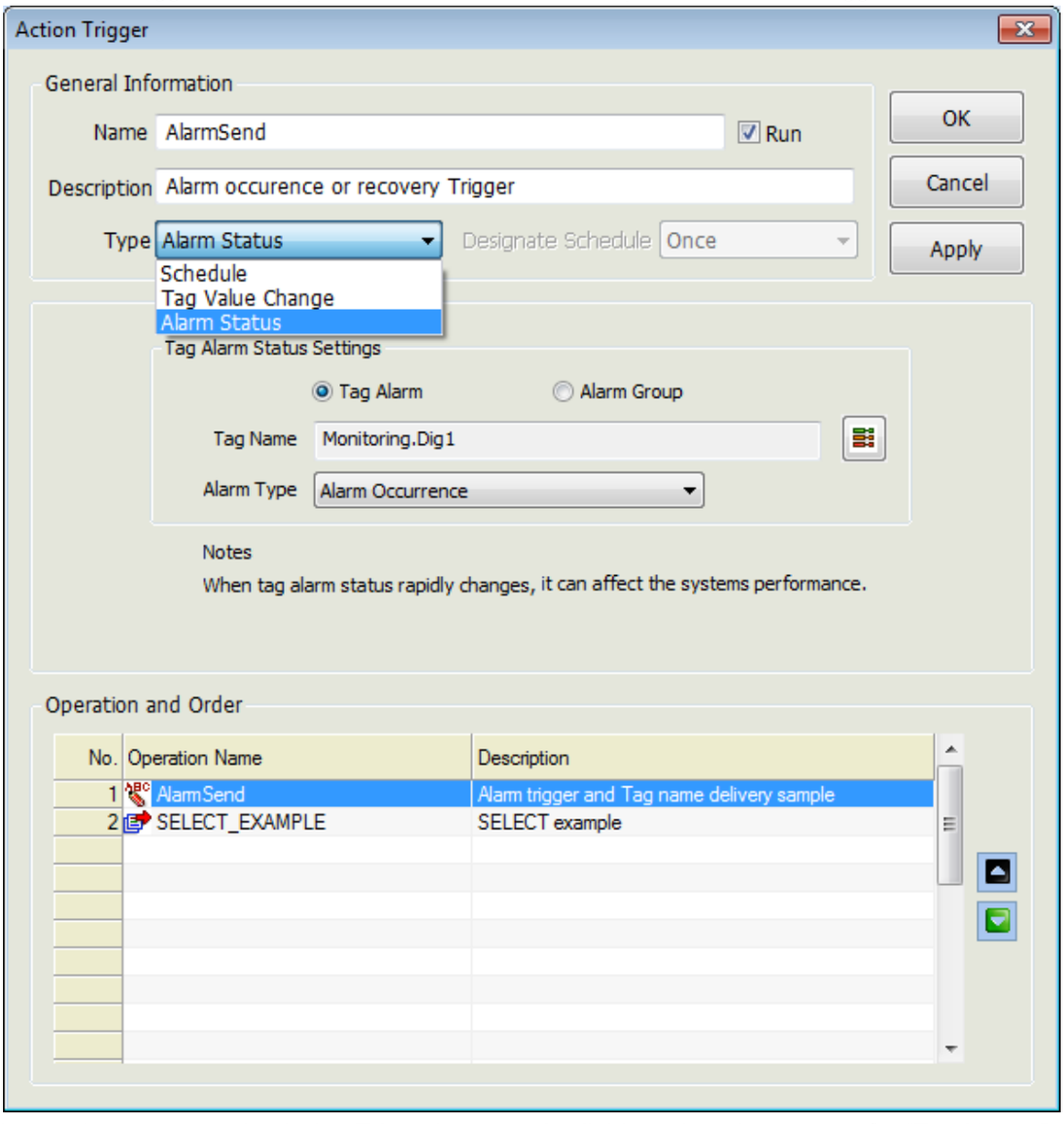
14.3.2 Tag Name Delivery

A script function is registered as a 'Tag Name Delivery' script function and processes to trigger actions are explained on the 'Action Trigger' screen.



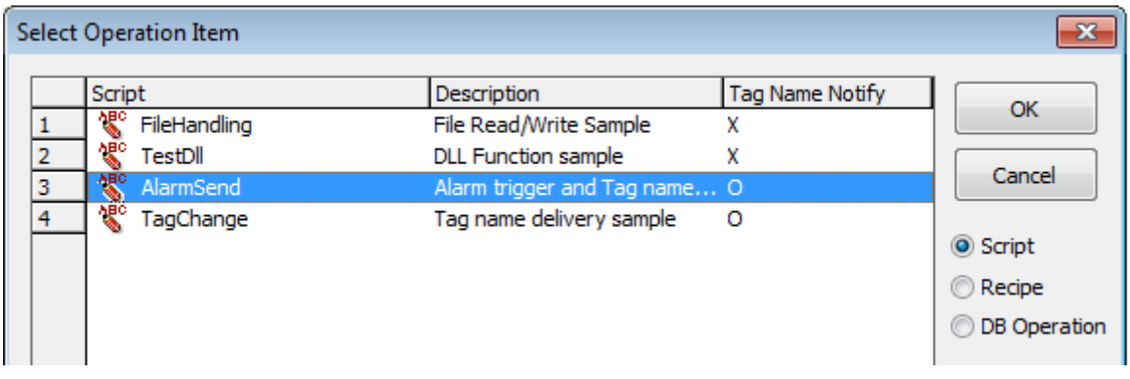
- Select 'Tag Name Delivery' option on the 'Script' screen.
- A sub-routine function (AlarmSend) is created with the same name with that of the script inputted among '"Main" function scripts. At this time, Korean letters can not be used for script names.
- sTagName is declared as a Public variant and the Main function is structured to deliver variants upon the call of subroutine function.
- vTagName converted to the parameter of AlarmSend(vTagName As Variant) function in the subroutine function view can be used as a tag name.
- Since the process to deliver the actual tag name can not be carried out on the editor screen while debugging is performed, debugging can be done in a method of inputting variant values manually with a formula by directly substituting each tag name on the execution screen.

● Then, 'Action Trigger' is set up to execute the selected script on the 'Action Trigger' screen.



● Select a trigger type on the Action Trigger' screen. Available action trigger types for "Tag Name Deliver" are "Alarm Status" and "Tag Value Change."

● Select and assign a script available for "Tag Name Deliver" as following when selecting a work subject.

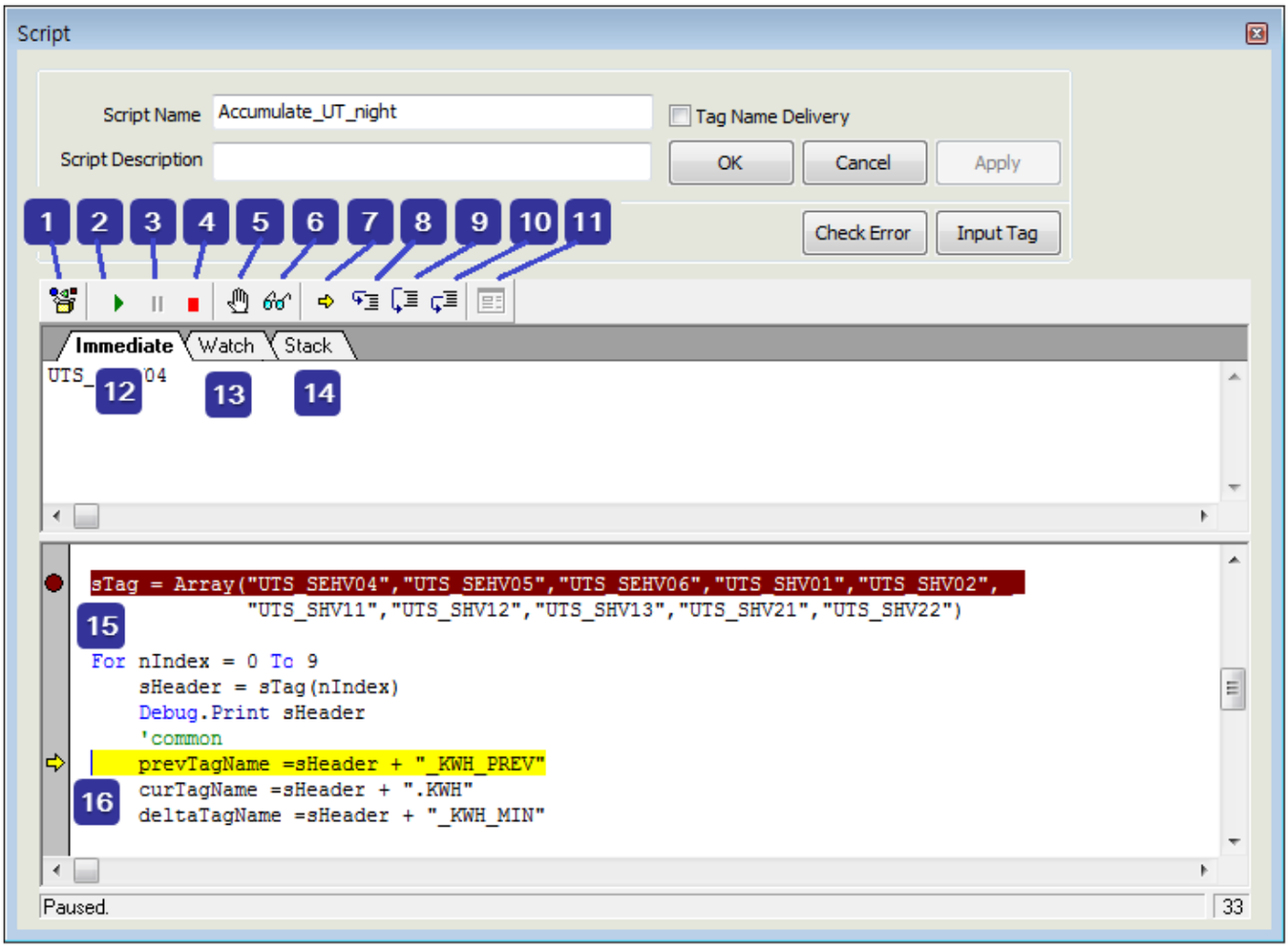


14.3.3 Editor Debugging

Debugging methods are explained on the 'Script' screen.

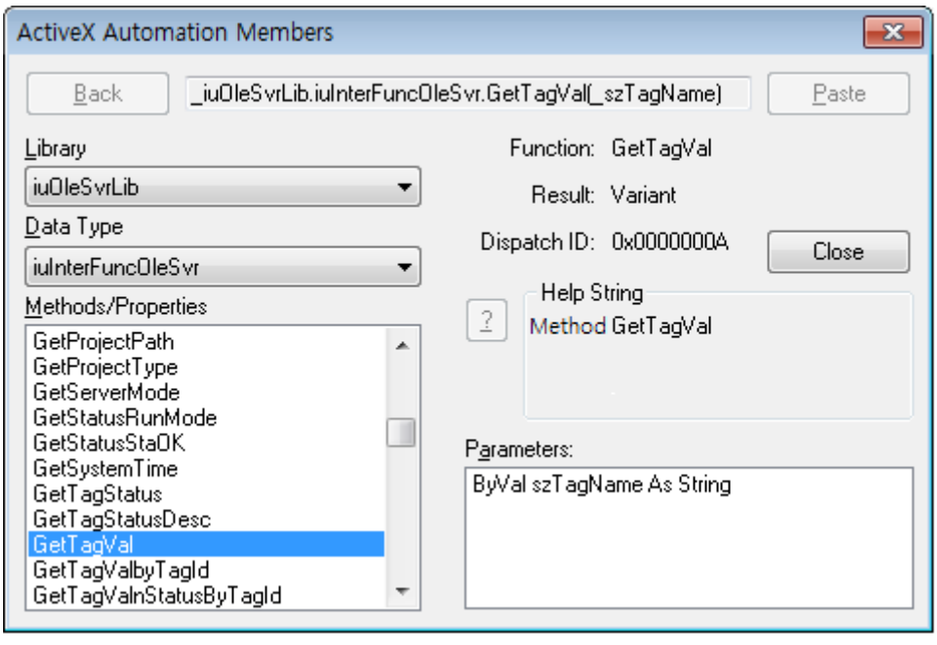
The inquiry of correct results is ensured only when debugging is worked while the Runtime is executed to have internal functions act correctly.

The following screen represents one of the script debug screen. Click on [2], [8], [9] and [10] on the 'Script' to show the followings.



(1) Browse Object

- Methods and Parameters of the referred ActiveX object are displayed.
- As default settings, information on the InfoU Function is displayed.



(2) Start/Resume:

- The user may execute a script on the editor (Since the execution of a correct function is ensured only when it is executed during the runtime, a code shall start for debugging).

(3) Pause

- The script that is being played is paused.

(4) End

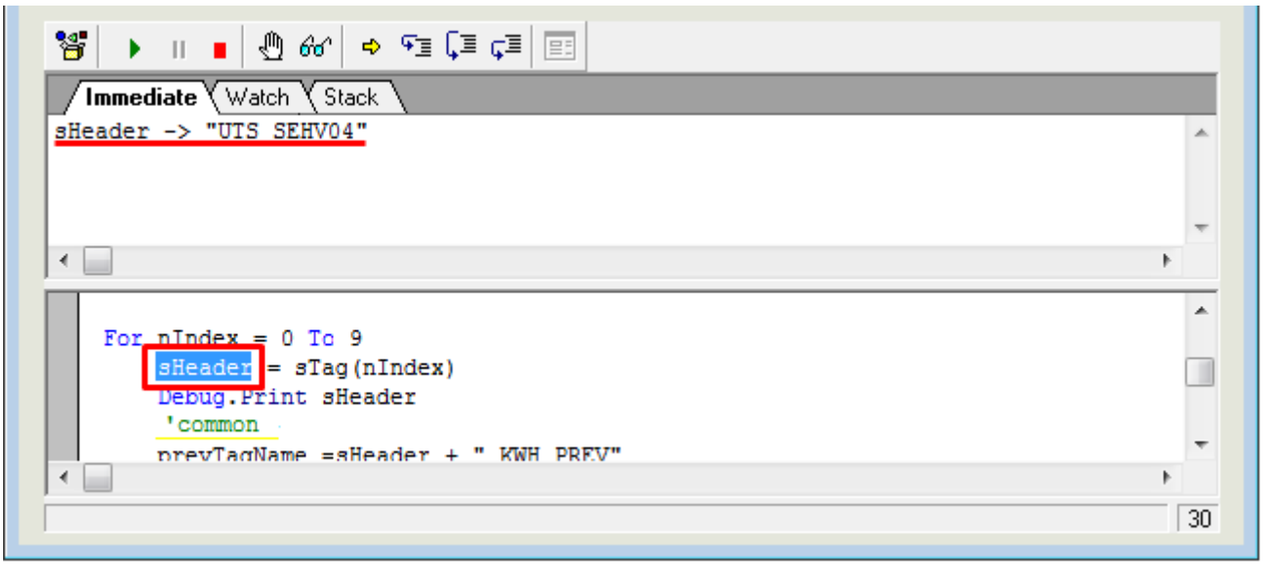
- The script that is being played is ended and the debug mode is terminated.

(5) Toggle Break

- A break point shall be either selected or cancelled.
- When setting a break point, it should be ensured that it stops at the break point to perform debugging.

(6) Evaluate Expression

- If a variant selected and clicked on the editor window, values are displayed on [12] immediate window.



(7) Show Current Statement

- The current statement is shown when executing.

(8) Step Into

- Each line of the scrip is executed one by one. If any ‘Sub Routine’ exists in the relevant line, it steps into the sub routine for debugging.

(9) Step Over

- The script statement is executed step by step. If any ‘Sub Routine’ exists, it moves to the next step after the sub routine is executed and its result is returned.
- If any break point exists in the sub routine, it stops at the break point.

(10) Step Out

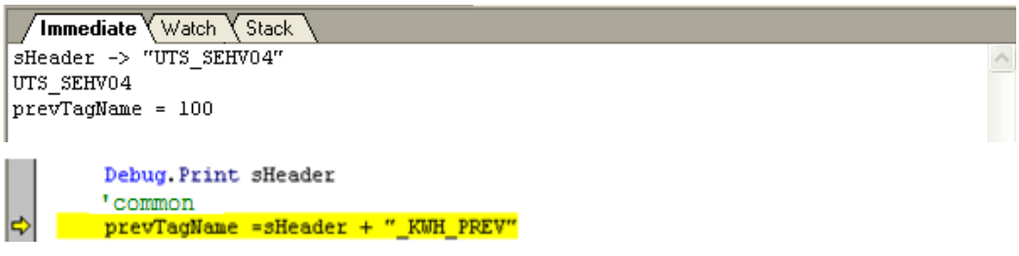
- It gets out of the current step and goes to upper procedure after completely executing all of the current functions and loop statements.
- If any break point exists in the sub routine, it stops at the break point.

(11) Edit User Dialog

- This dialog box editor can not be selective during the debugging and it is activated only in the ‘Edit’ mode.
- InfoU does not support this dialog box in the runtime environment.

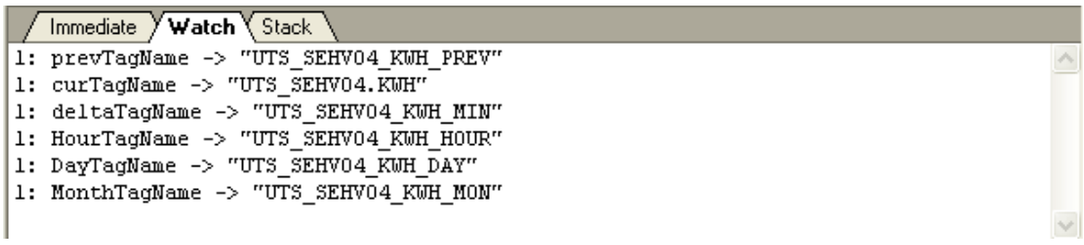
(12) Immediate

- This ‘Immediate’ window is used as follows.
  - Result value when clicking on [6] Evaluate Expression
  - Show the return valued of debug.Print
  - Change variant values when any ‘Immediate’ formula is inputted.



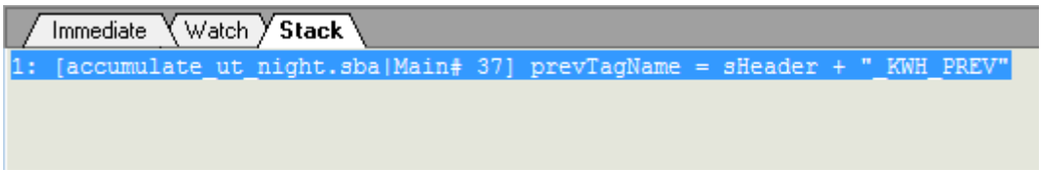
(13) Watch

- Variants are assigned and function or expression values can be inquired.
- For assigning, select a variant and click the right side of the mouse to select Debug-->Add Watch.



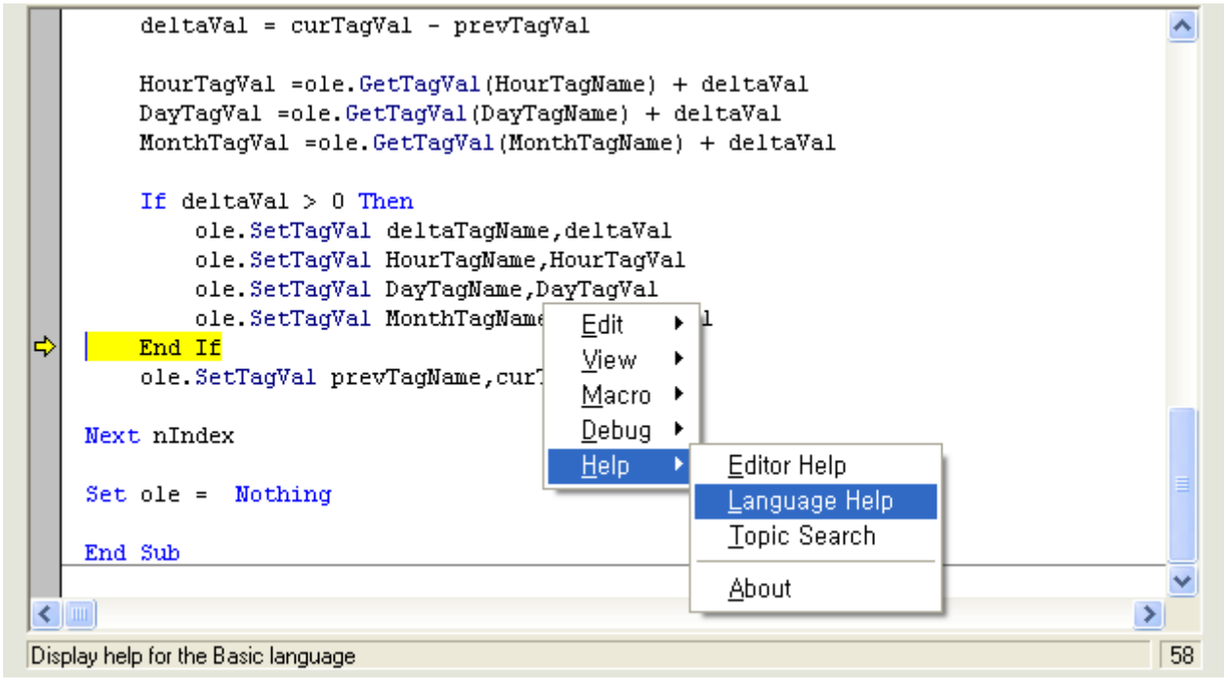
(14) Stack

● The call list of the current execution line is shown.



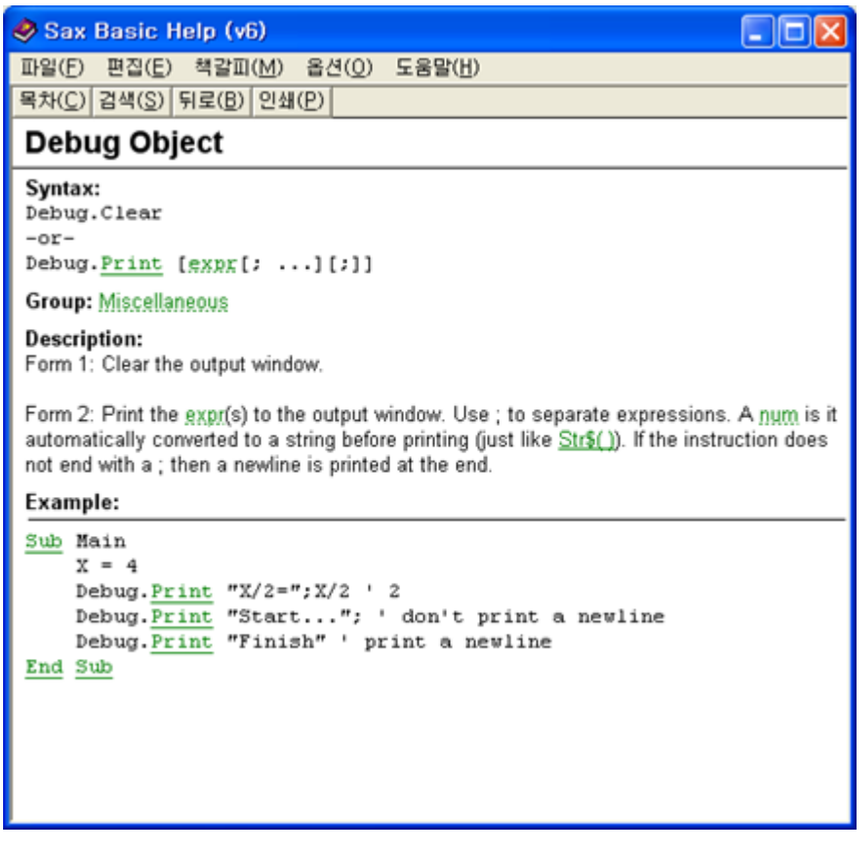
(15) Menu Function

● If the right side of the mouse is clicked, a menu is shown and it has functions available in the debug mode as well as functions available in the Edit mode.



(16) Help

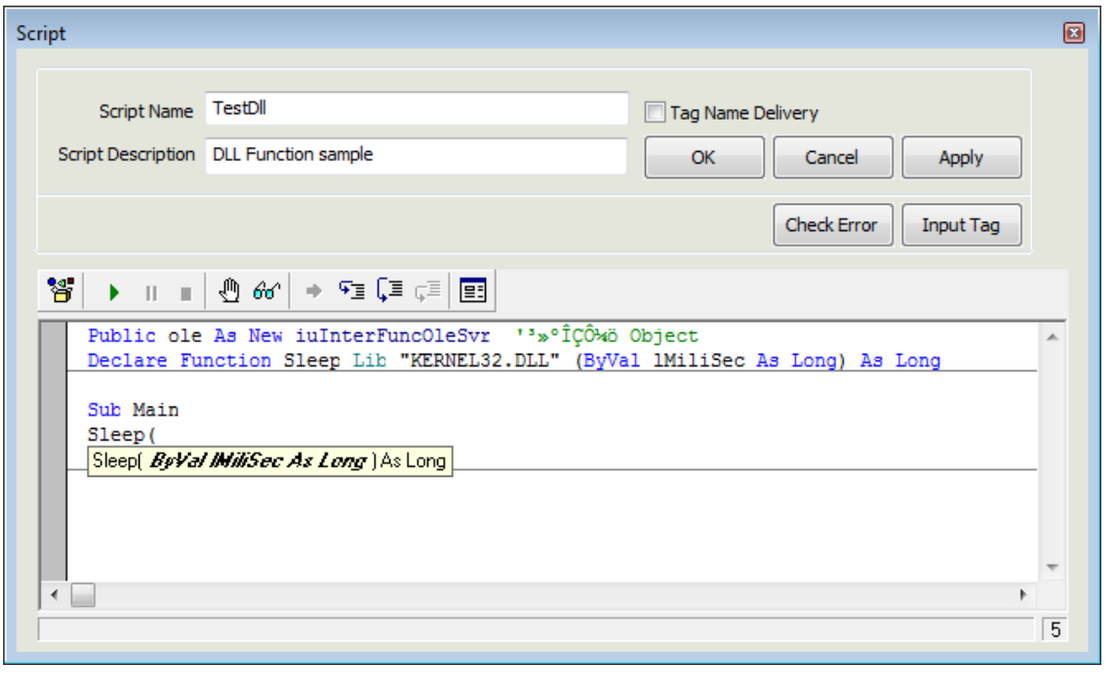
● Help' is available for overall information on various functions and script languages provided by the editor.


Sax Basic Help (v6)
파일(F) 편집(E) 책갈피(M) 옵션(O) 도움말(H)
목차(C) 검색(S) 뒤로(B) 인쇄(P)
Debug Object
Syntax:
Debug.Clear
-or-
Debug.Print [expr[; ...][;]]
Group: Miscellaneous
Description:
Form 1: Clear the output window.
Form 2: Print the expr(s) to the output window. Use ; to separate expressions. A num is it automatically converted to a string before printing (just like Str$( )). If the instruction does not end with a ; then a newline is printed at the end.
Example:
Sub Main
X = 4
Debug.Print "X/2=";X/2 ' 2
Debug.Print "Start..."; ' don't print a newline
Debug.Print "Finish" ' print a newline
End Sub

14.3.4 How to use DLL

How to use external DLL functions in the script is explained.

DLL API use: It is an example of declaration of ‘Sleep’ function included in System KERNEL32.DLL.



14.3.5 How to use OLE object

It is an example of file ‘Open and Write’. (Refer to examples in Demo projects.)

```
Dim fso As Object       'file object
Dim cFile As Object     'file object

… … …
Set fso = CreateObject("Scripting.FileSystemObject")
'OpenTextFile(filename,mode,create) 'mode 1:Read,2:Write,8:Append

'write example
On Error GoTo ERR_FILE_WRITE_OPEN   'error handling
Writefile = filePath + "\Taglist.txt"
    If Not fso.FileExists(Writefile) Then               'find file
        Set cFile = fso.CreateTextFile(Writefile,True)  'nothing--> create
    Else
        Set cFile = fso.OpenTextFile(Writefile,2,True)  'exist--> write option = 2
    End If

    DataLine = ""
    For n = 0 To tagCount-1
        DataLine = CStr(sTag(n))
        DataLine = DataLine  + vbrf
        cFile.WriteLine(DataLine)
    Next
    cFile.Close
```

14.3.6 Error Handling

It is an example of how to handle errors (Refer to examples in Demo projects.)

Write On Error statement on the top of the error to be handled. If any error occurs below the On Error statement, it moves to the Error Handle Statement written next to Goto Statement.

Since On Error statement is effective until it meets with the next On Error statement, 'Error Handle' is possible according to work steps by writing the On Error statement at several locations.

By using the 'On Error Resume Next Statement', the next codes can be performed seamlessly even any minor error occurs.

```
'append example
On Error GoTo ERR_FILE_APPEND_OPEN  'error handler
Appendfile = filePath + "\" + Format(Now(),"YYMMDD") + "_timelist.txt"
    If Not fso.FileExists(Writefile) Then               'find file?
        Set cFile = fso.CreateTextFile(Appendfile,True) 'No:create
    Else
        Set cFile = fso.OpenTextFile(Appendfile,8,True) 'yes:write option = 8
    End If

    DataLine = Format(Now(),"YYYY/MM/DD hh:mm:ss") + vbTab
    For n = 0 To tagCount-1
        tagVal = ole.GetTagVal(CStr(tagName(n)))
        AppendLine = AppendLine + CStr(tagVal) + ","
    Next
    DataLine = DataLine + AppendLine + vbrf
    cFile.WriteLine(DataLine)
    cFile.Close
    Debug.Print "File Append OK!"
Exit Sub

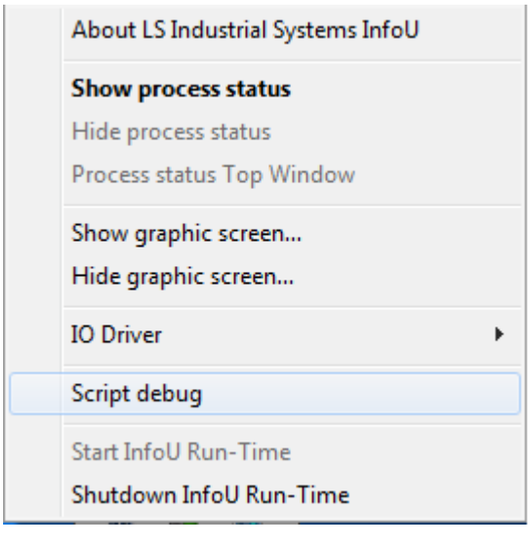
ERR_FILE_READ_OPEN:
    errFlag = -1
    Debug.Print "ERR_FILE_READ_OPEN"
    Exit Sub

ERR_FILE_WRITE_OPEN:
    errFlag = -1
    Debug.Print "ERR_FILE_WRITE_OPEN"
    Exit Sub
```

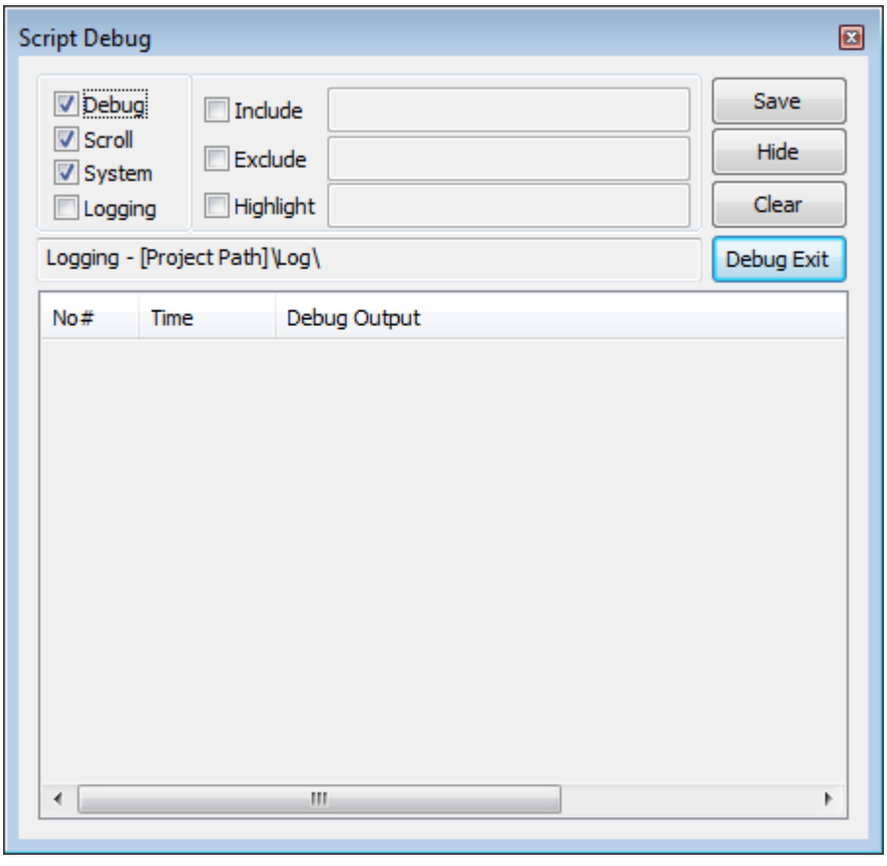
14.4 Runtime Debug

By providing the debug screen during the Runtime, this ensures the user inspects the current execution status of the script and check for any operational and logical error.

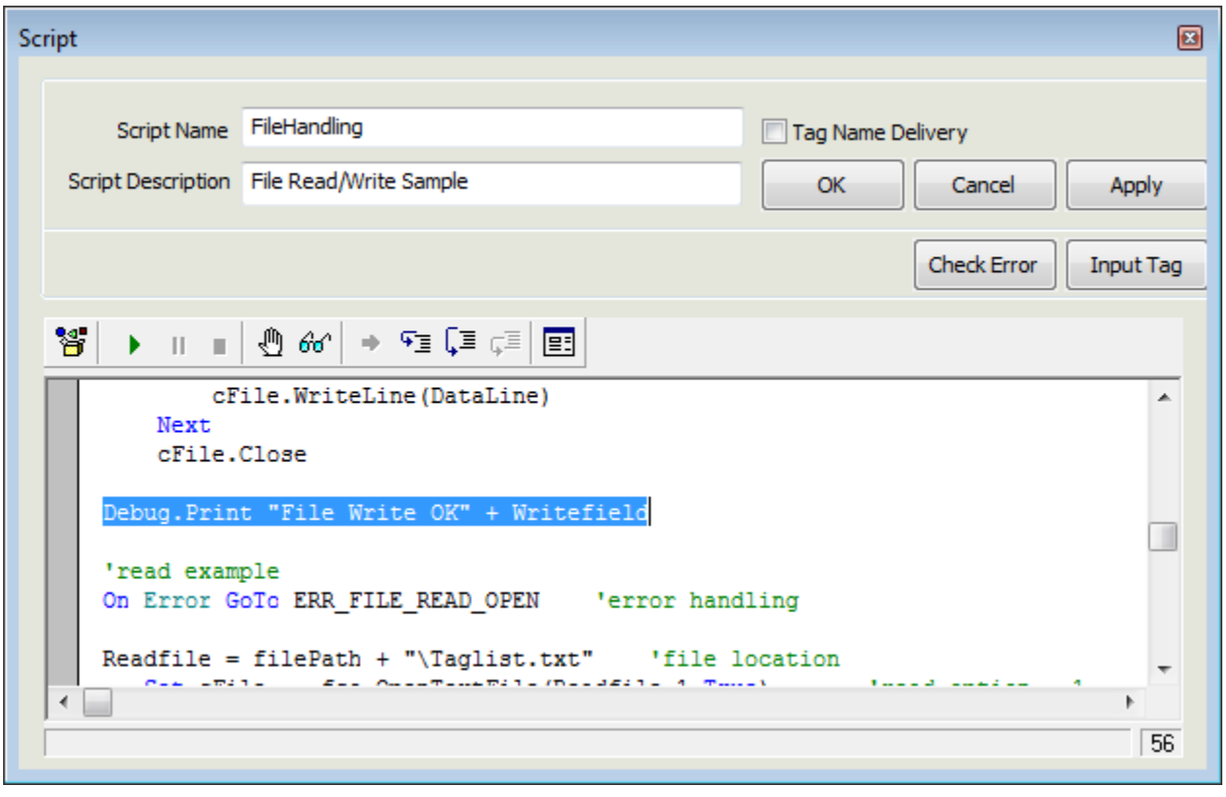
● Execution



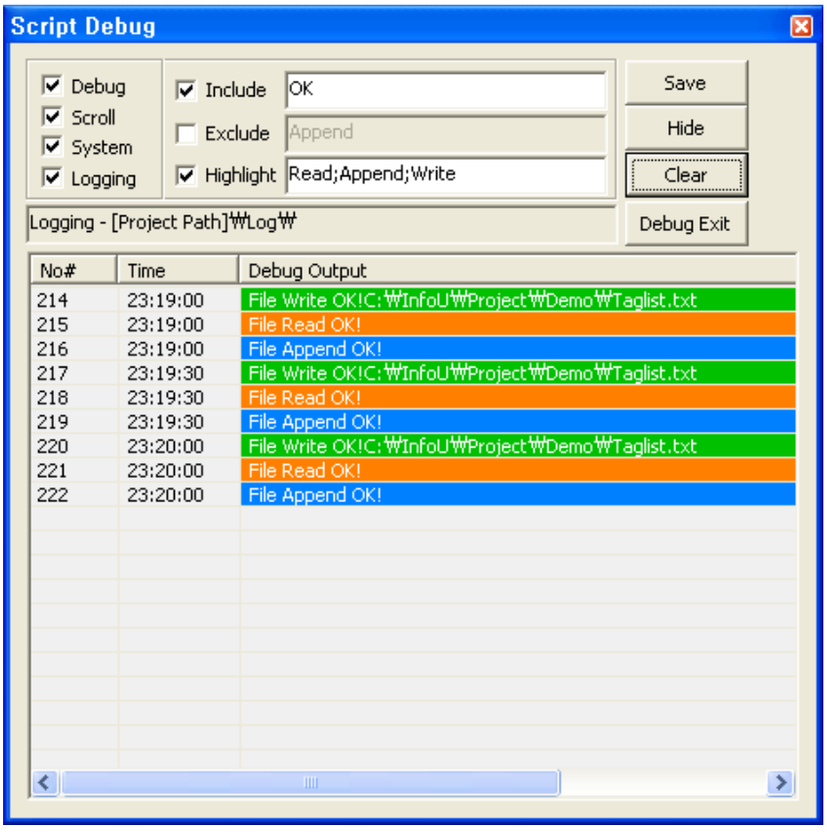
Click the right side of the mouse on the InfoU tray icon during the Runtime to select ‘Scrip Debug’ from the pop-up menu and show the screen below.



The output written in Debug.Print [expr] on the 'Script' screen is displayed as a result during the runtime



● Runtime Debugging



This screen is displayed as the highest window.

-Debug: Select whether to conduct debug works,

-Scroll: Scroll the screen to show the latest list.

-System: Select whether to execute script except Debug.Print construction.

-Logging: Save debugging contents in the \Log folder on the current project path.

-Include: The letters to be displayed are filtered and displayed.

-Exclude: The letters to be excluded are filtered and displayed.

-Highlight: The contents included as inputted letters are highlighted with colors.

-Include, Exclude and Highlight can be overlapped by using [;].

-Save: The current settings and the debug screen size are saved and the current conditions will be displayed on the screen in the next execution.

-Hide: The debug screen is hid.

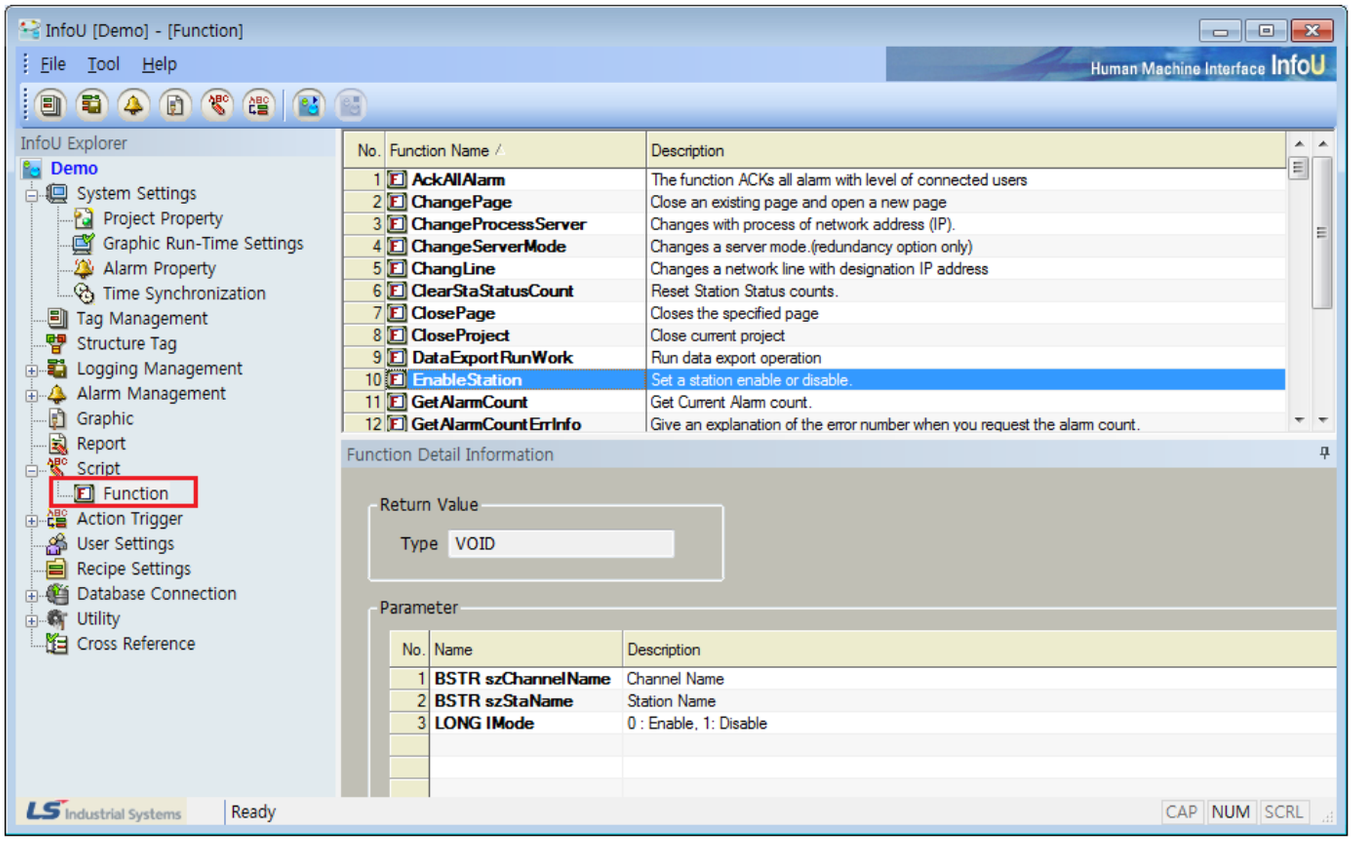
-Clear: All of the lists currently displayed are deleted.

-Debug Exit: The runtime debug is ended and the screen is closed.

Chapter 15 Function

Various default functions are presented to perform various actions in the InfoU system. The following list shows functions that can be used for Script and OLE Automation.

15.1 Funtion Inquire



After logging in withInfoUD.exe, select ‘Function’ under ‘Script’ on the InfoU Explorer’ To see detailed information on a certain function, select and click on the relevant function with mouse. Then, input ‘Return Value’ of the relevant function on the 'Function Detail Information’ screen to display a list that shows the properties and description of each parameter requested in the relevant function in regular sequence

Notice

☞ For the example of the internal function and detailed usage, refer to ‘Internal Functions’ of the Appendix.

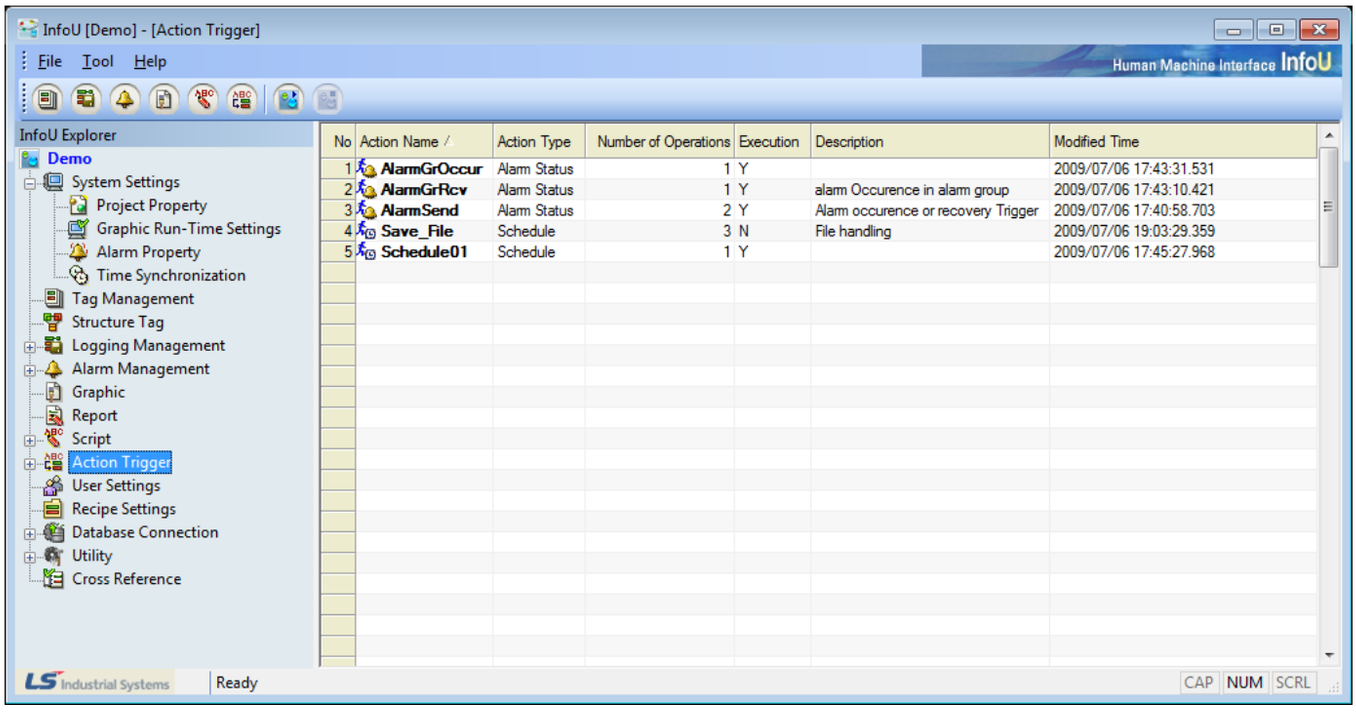
Chapter 16 Action Trigger

Action Trigger is set up during the runtime to execute input and output works of the script, recipe and database. The relevant application can be executed by setting up Action Trigger according to 'Schedule', 'Tag Value Chang' or 'Alarm Status'.

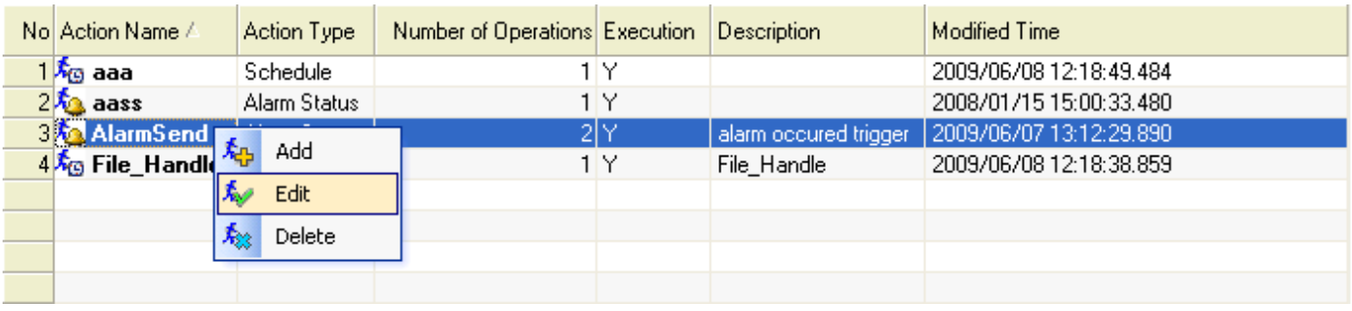
One-stop settings for various application actions at one place ensure an effective and systematic control.

16.1 Start

If 'Action Trigger' in the menu or 'Edit Script' in the InfoU explorer is selected, the list of action trigger is displayed as follows:



Click the right button of the mouse on the list of Action Trigger to show Add, Edit and Delete menu.



- Add: A new action trigger is added. Double click on the empty item to display the new input screen.
- Edit: The selected action trigger is edited. Double click an item to display the edition screen.
- Delete: Delete the selected action trigger.

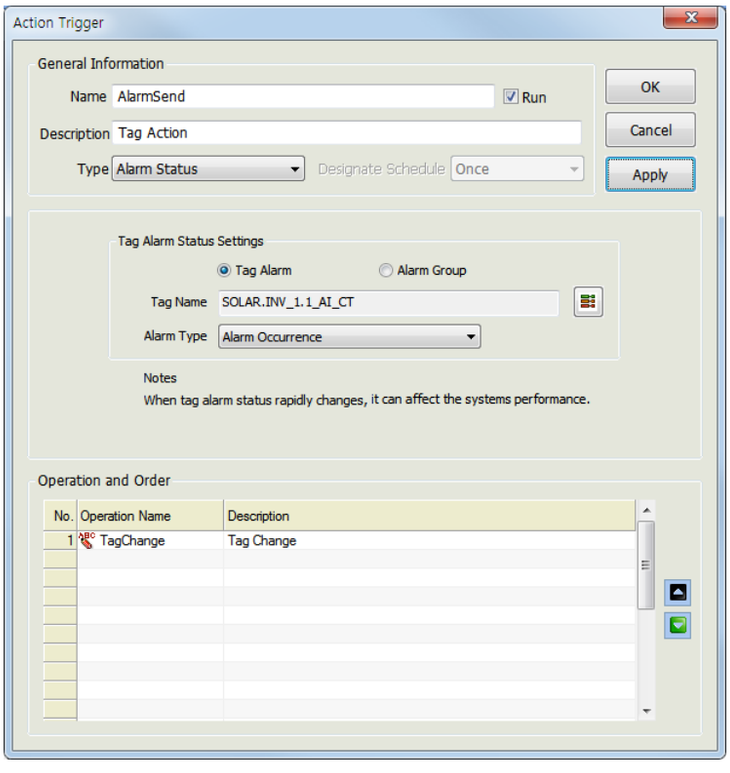
16.2 Configuration

16.2.1 General Information

The edition screen of action trigger is of the dialog type and it is organized as follows:

If other item is selected from the above action trigger list while the screen is displayed, information on the selected action trigger is displayed.

Any modification, which may be created during the edition of action trigger, is not reflected during the runtime and it is applied after the runtime is restarted.



(1) Name

- The action trigger name is displayed.
- If the action trigger is renamed and saved, a new action trigger is created and added automatically.
- Any two action trigger s shall not have the same name.

(2) Description

Input description on the action trigger.

(3) Run

It displays whether to execute the script actions during the runtime.

(4) Type

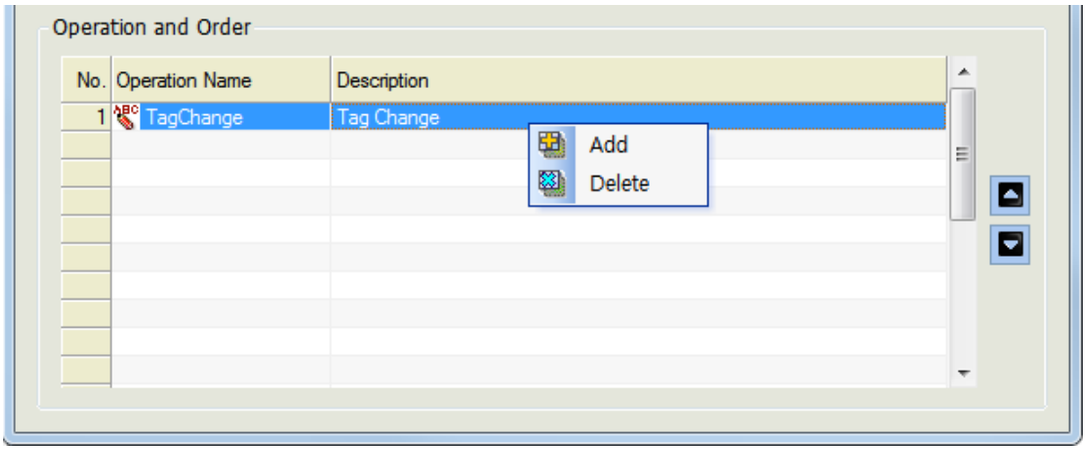
The script actions are triggered by one of the following condition according to the designated settings:

- Schedule
- Tag Vale Change
- Alarm Status

(5) Operation and Order

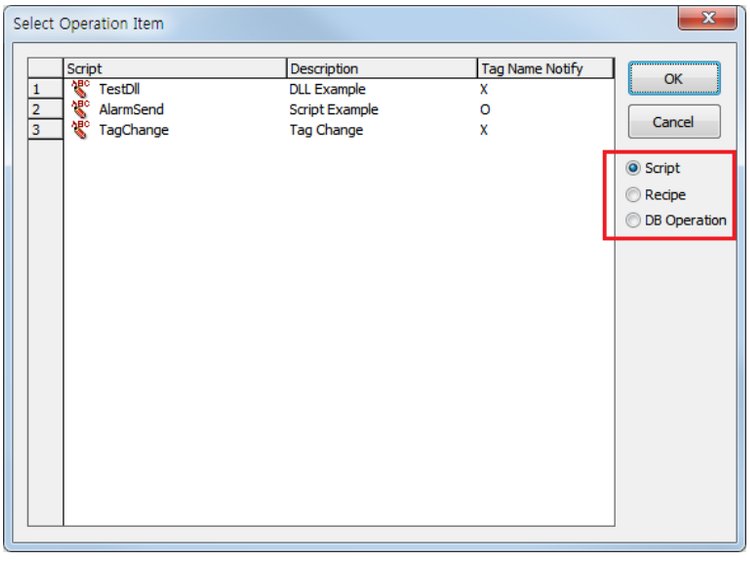
Select an operation to execute among the defined actions and designate the order.

Click the right side of the mouse on the list to show the menu below.



- Add: Click an additional menu or double click to show a screen for selecting a new operation.
- Delete: Delete the selected operation from the list.

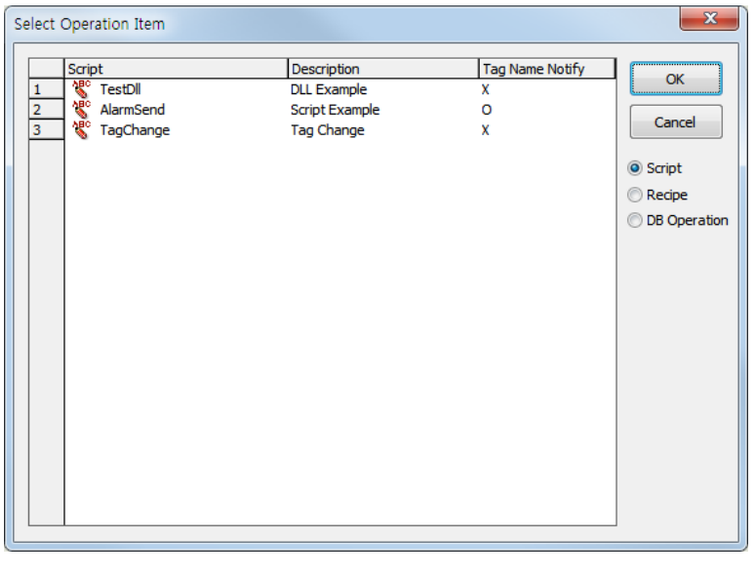
You can select the task item in the below screen.



There are 3 types of task items; “script”, “recipe”, “DB works”.

If you select the type of task items on the right side, the type will be displayed on the list.

1) Script list

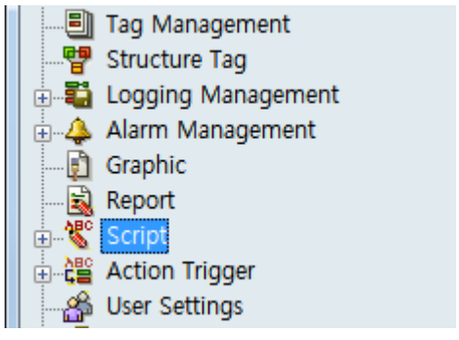


In case of the script, whether forwarding the tag name is displayed.

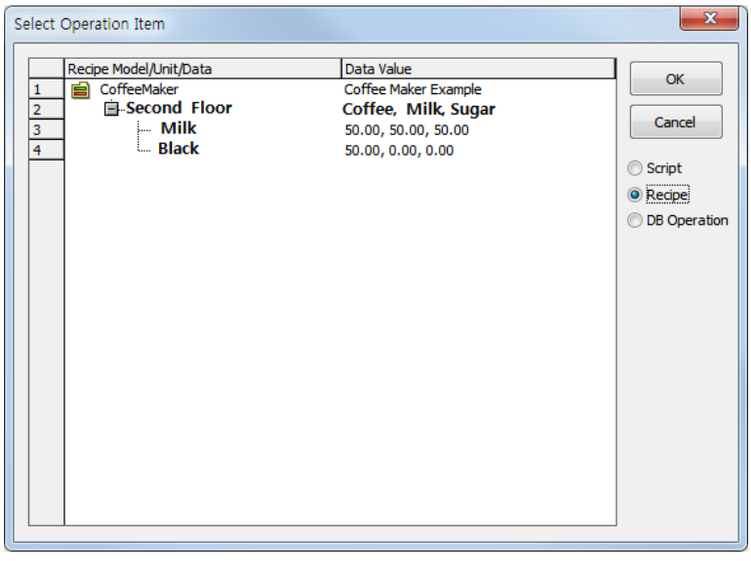
Notice

☞ In the task item, you can just select the script.

☞ You can add, delete and edit the script in the below script menu.



2) Recipe list

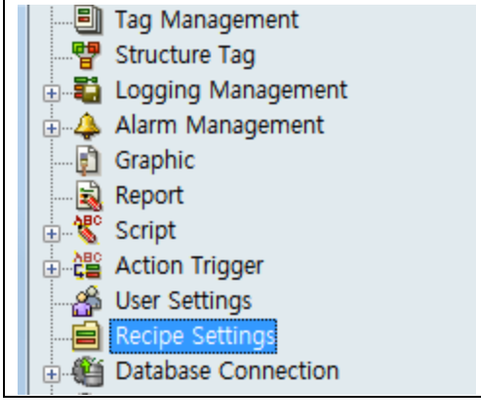


The recipe is based on the tree type of the model, unit and data; you can just select the item of the lowest level only.

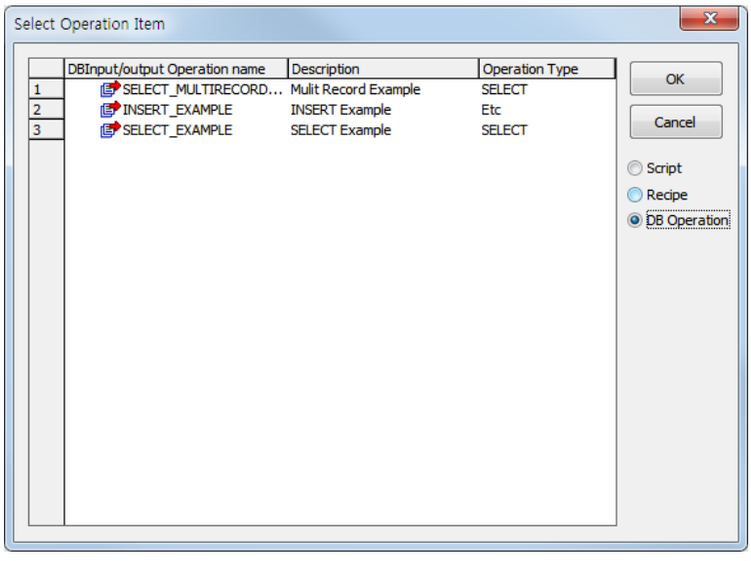
Notice

☞ In the task item, you can just select the recipe.

☞ You can add/delete/edit the recipe in the below recipe setting menu.



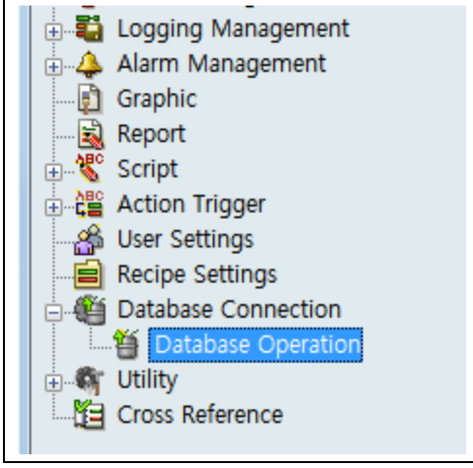
3) DB operation item



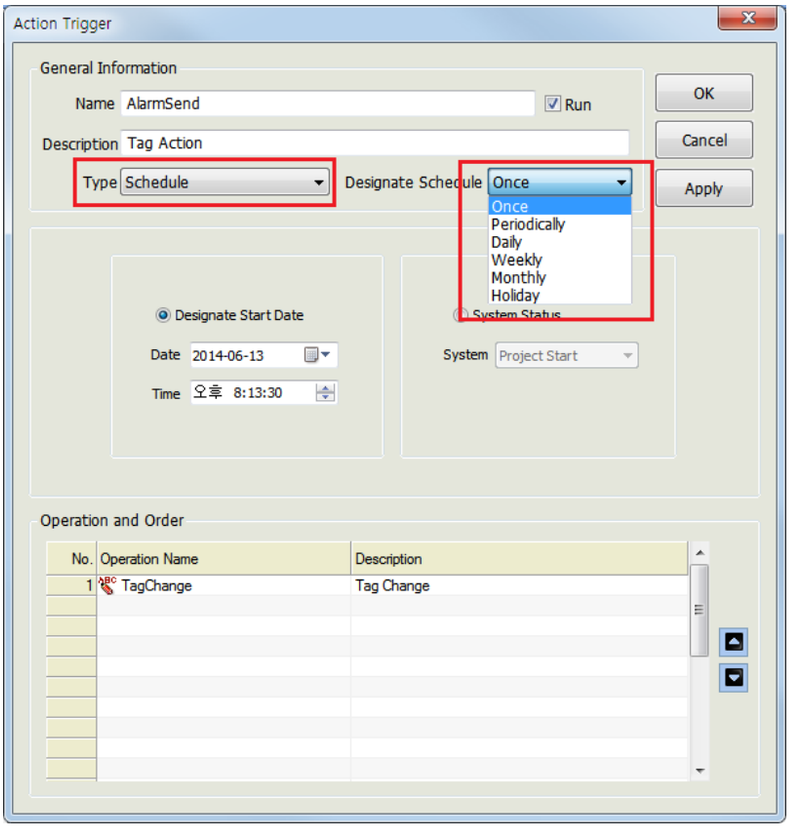
Notice

☞ In the task item, you can just select the DB operation items.

☞ You can add/delete/edit the DB operation in the below DB operation menu.



16.2.2 Schedule

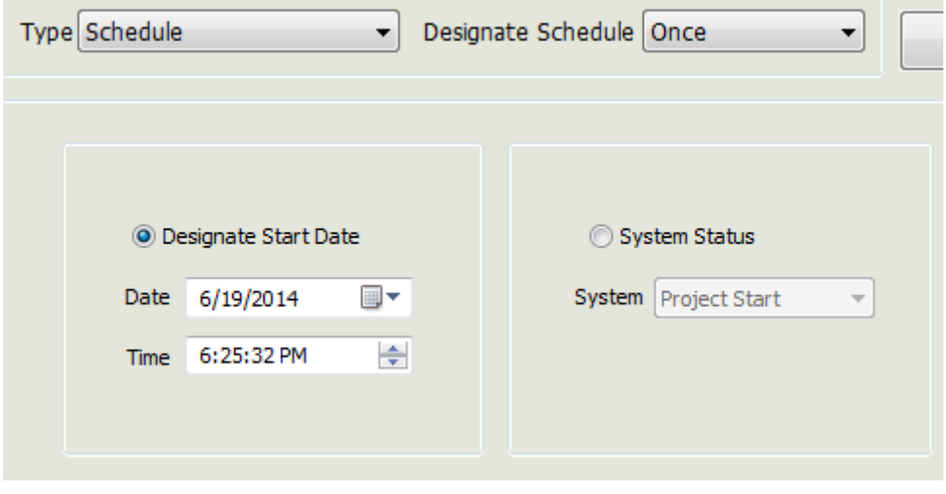


(1) General Information

If 'Schedule' is selected for 'Type',

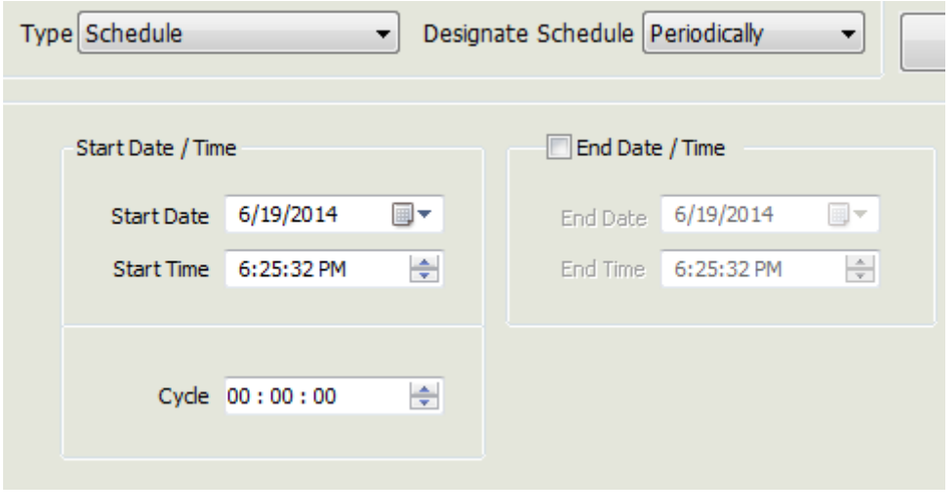
Designate Schedule' becomes activated and here, one of [Once], [Periodically], [Daily],[Weekly], [Monthly] and [Holiday] can be selected.

(2) Once



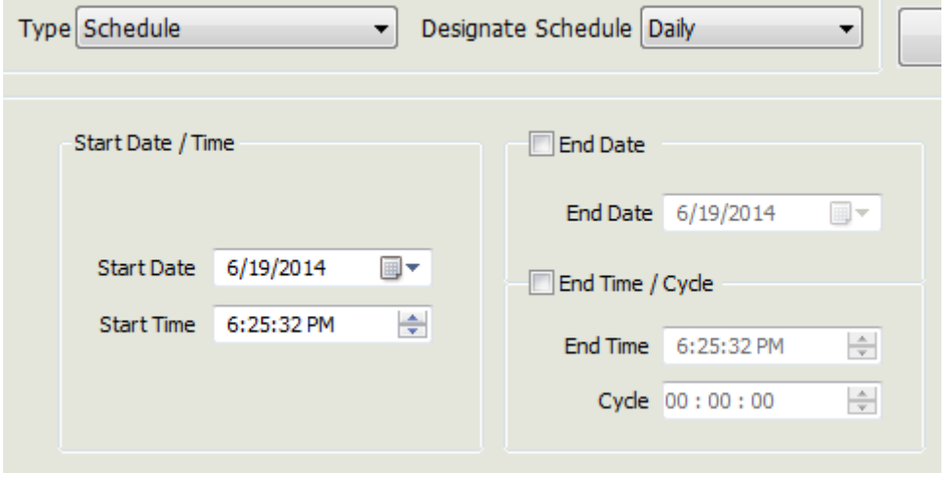
- Designate Start Date: If 'Designate Start Date' is selected, it is allowed to input date and time to start the selected operation and it is executed only once at the designated time.
- System Status: The operation is executed according to the system's management status over the project.
  - Project Start
  - Project Close
  - Redundancy Switch

(3) Periodically



- Start Date/Time: Time point at which the periodic cycle actions start
- Cycle: Input a cycle of actions.
- End Date/Time: Designate date and time to stop actions.

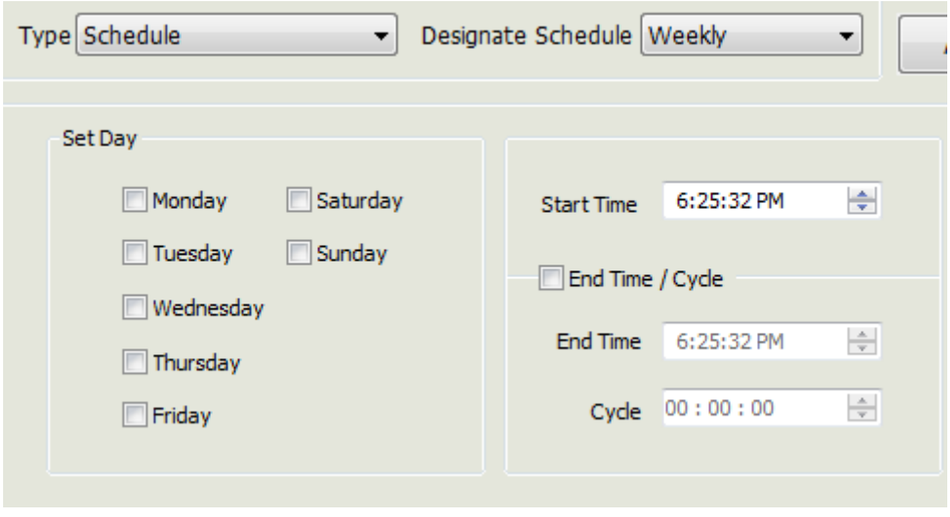
(4) Daily



- Start Date/Time: Time to start daily actions. If any time interval is not set up, actions are executed only once at the relevant time.
- End Date: Designate date to stop actions

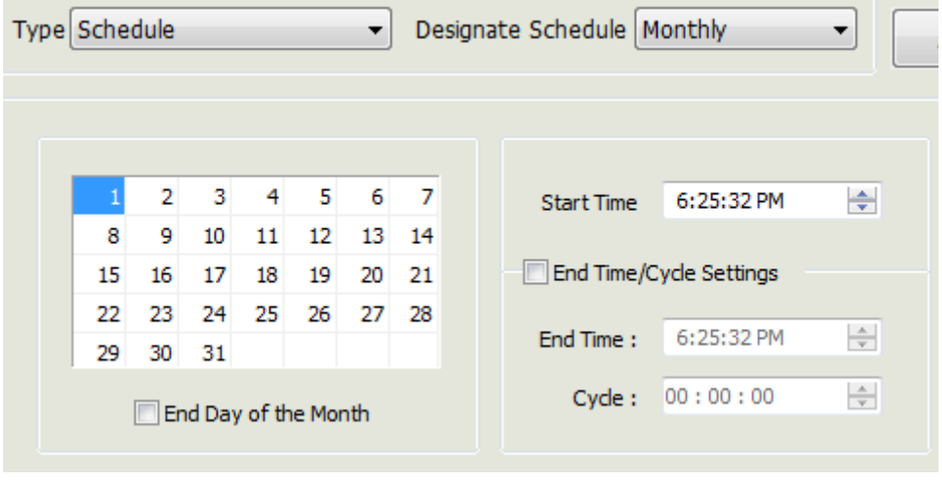
● End Time/Cycle: Input ‘End time’ and a cycle of actions executed from the time to execute to the time to end.

(5) Weekly



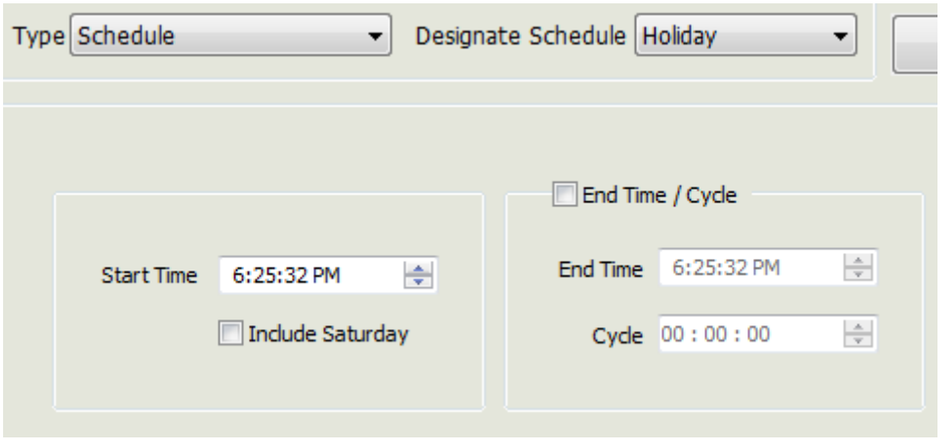
● Set Day: Select a day to execute actions. Multiple choices are allowed.
● Start Time: Time to start actions on the relevant day. If any time interval is not set up, actions are executed only once at the relevant time.
● Endtime/Cycle: Input ‘End time’ and a cycle of actions executed from the time to execute to the time to end.

(6) Monthly



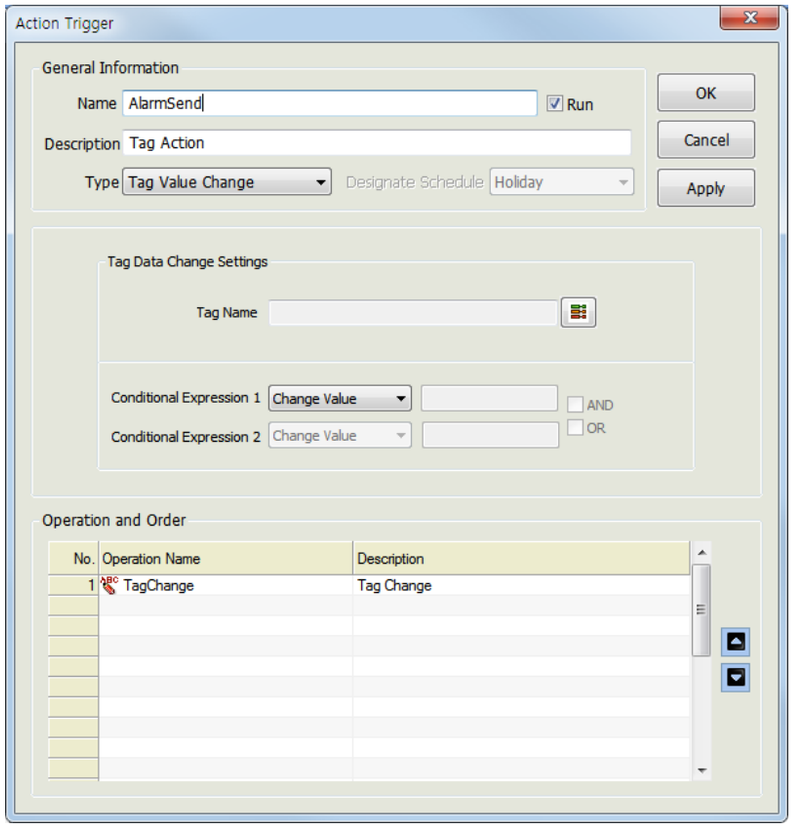
● Select Day: Select a day to execute actions in the relevant month. Click the mouse on a day to select and the selected day is highlighted in yellow.
● Start Time: Time to start actions on the relevant day and If any time interval is not set up, actions are executed only once at the relevant time.
● End Time/Cycle Setting: Input ‘End time’ and a cycle of actions executed from the time to execute to the time to end.

(7) Hoilday



- Start Time: Time to start actions on the relevant date and If any time interval is not set up, actions are executed only once at the relevant time. It is possible to include Saturday.
- End Time/Cycle: Input ‘End time’ and a cycle of actions executed from the time to execute to the time to end.

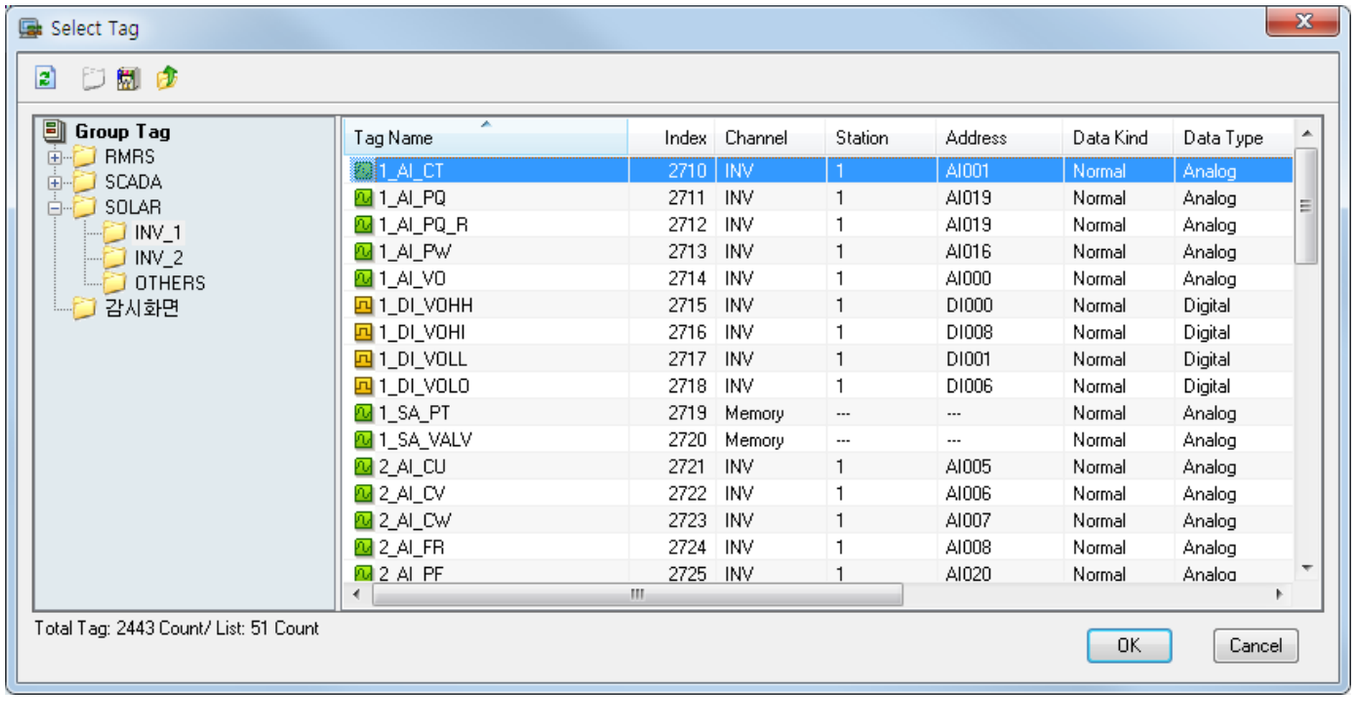
16.2.3 Tag Value Change



- Type: Select [Tag Value Change].

● Tag Name: If [3] is clicked, the tag selection screen is displayed and the tag name is inserted in box for tag name if one of tags is selected.

● Tag Selection Button: Click on this button to display the screen below and select a tag on the screen.

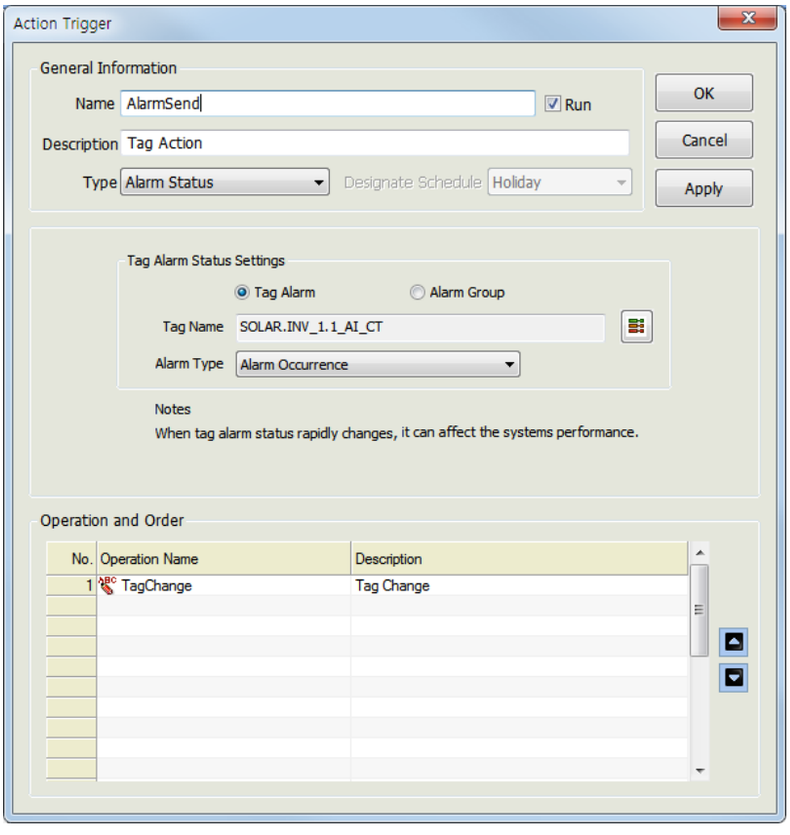


● Conditional Expression: Conditional Expression of the inputted tag value.

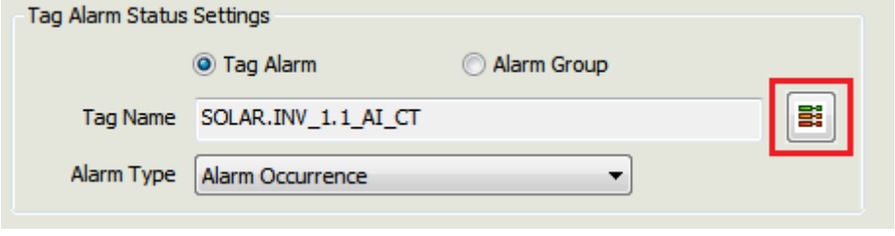
- [Change Value]: The operation is conducted if the tag value is changed. Not available to set up Conditional Expression. Not available to input values
- [=]: The action is performed if the tag value is equal to the set-up value.
- [<>]: The condition is implemented when the tag value is different form the set-up value
- [>]: The condition is implemented when the tag value is higher than the set-up value
- [>=]: The condition is implemented when the tag value is higher or equal to the set-up value
- [<]: The condition is implemented when the tag value is lower than the set-up value
- [<=]: The condition is implemented when the tag value is lower or equal to the set-up value

● AND / OR: Implemented when Conditional Expression 1 and Conditional Expression 2 are AND(OR).

16.2.4 Alarm Status

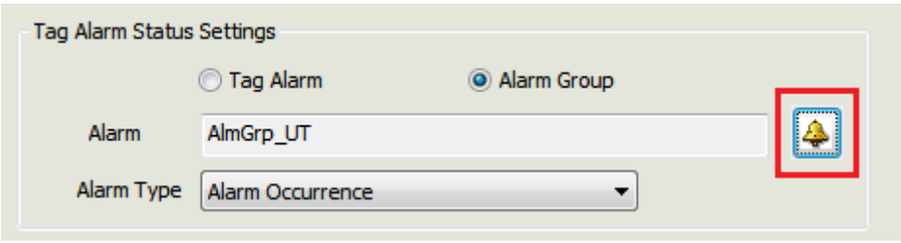


- Type: Select [Alarm Status]
- Tag Alarm: Set up actions for each tag upon the occurrence of alarm.
- Alarm Group: Set up actions for each tag that belongs to an alarm group upon the occurrence of alarm.
- The following items are displayed according to the mode of Tag Alarm and Alarm Group:
  - Tag Name: Click on [5] to display the tag selection screen and input the selected tag name on the screen.



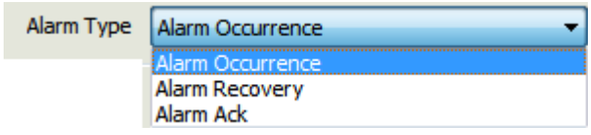
  - Alarm Group: Click on [5] to display the alarm group selection screen and input the selected alarm group name on the screen.

- Tag/Alarm Group Selection: Click to display the 'Tag/Alarm Group Selection' screen and select a tag (alarm group) on the screen.

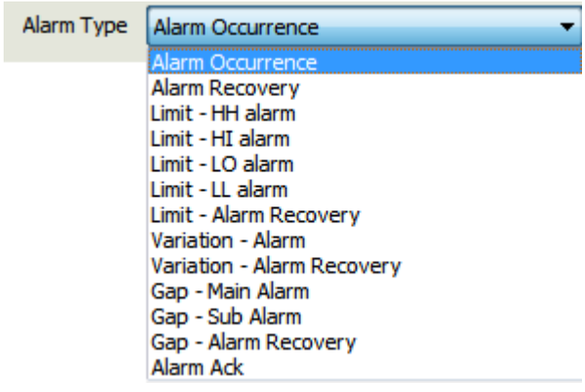


● Alarm Type: Available alarm types are displayed differently between digital and analog tags.

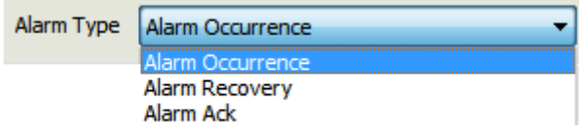
- Digital



- Analog



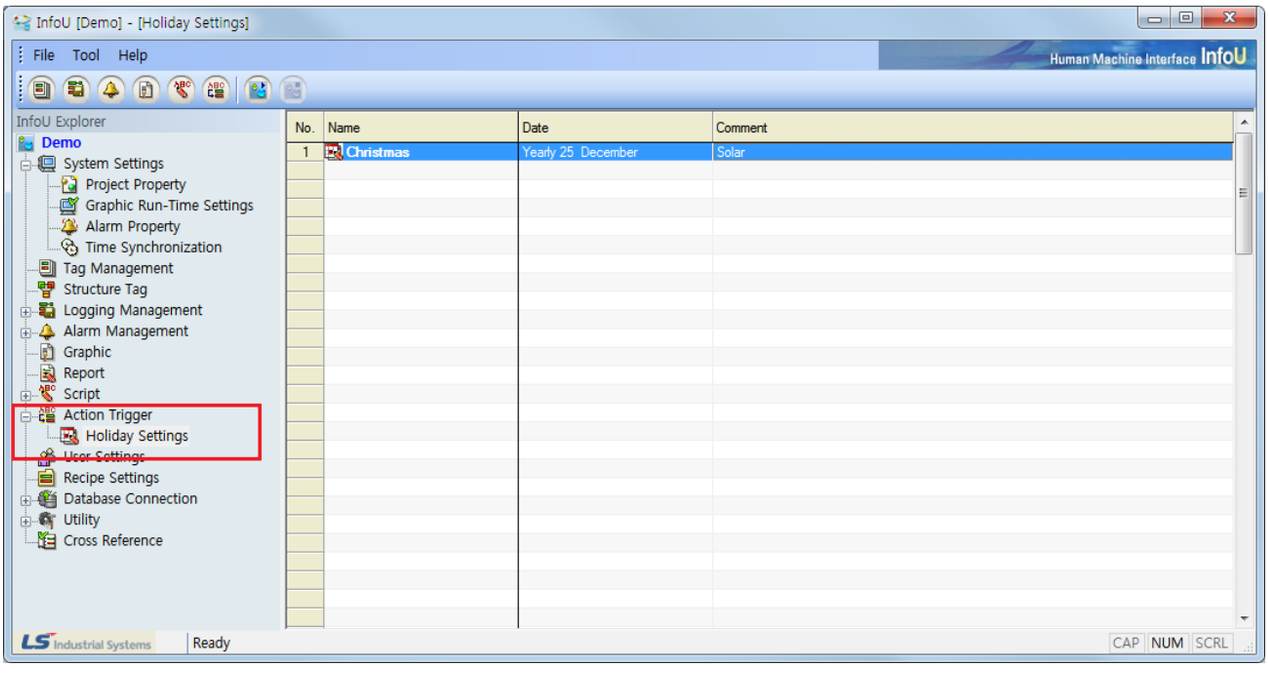
- Alarm Group



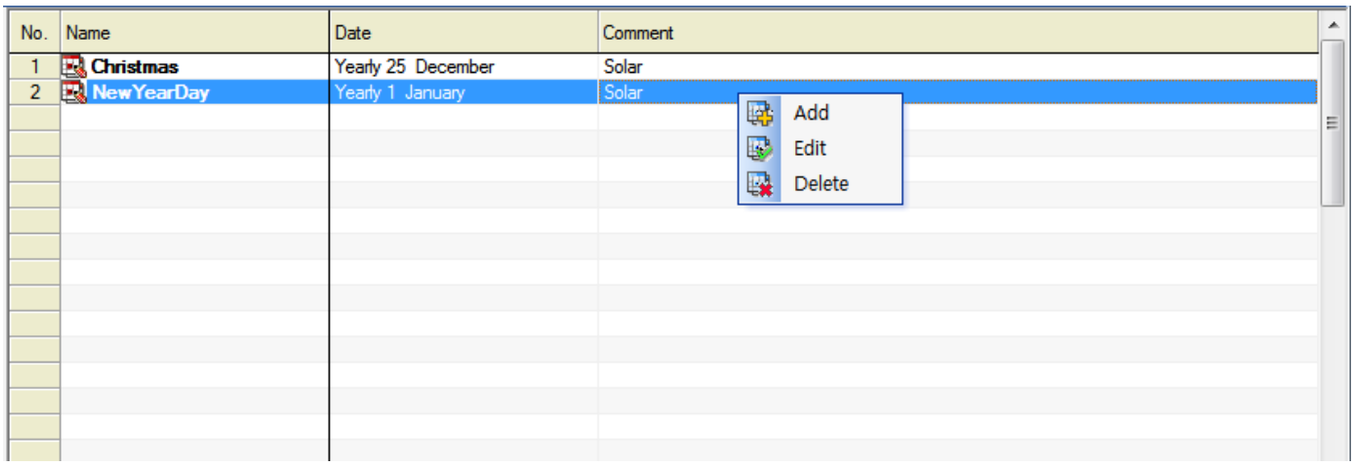
16.3 Holiday Settings

(1) Start

If 'Holiday Settings' in the menu or the InfoU explorer is selected, the list of action trigger is displayed as follows:



Click the right button of the mouse on the list of Action Trigger to show Add, Edit and Delete menu.

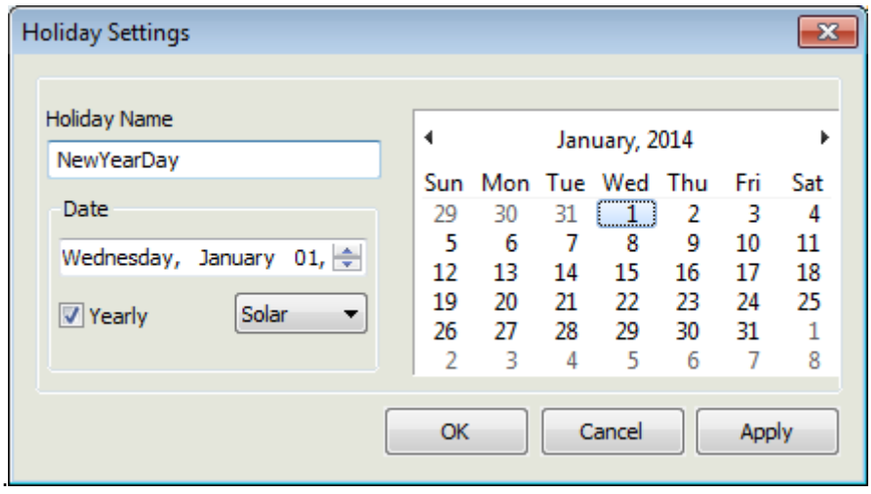


- Add: A new action trigger is added. Double click on the empty item to display the new input screen.
- Edit: The selected action trigger is edited. Double click on the item is double clicked to display the edition screen.
- Delete: Delete the selected action trigger.

(2) Edition Screen

The holiday setting edition screen is of modaless dialog type and it is constructed as follows:

If other item is selected from the above action trigger list while the screen is displayed, information on the selected action trigger is displayed.



- Holiday Name: The selected holiday name is displayed or other holiday name can be newly inputted.
- Date: Input the holiday date.
- Yearly: Set up whether it is repeatedly executed every year or only one time.
- Solar/Lunar: Select either Solar or Lunar
- Calendar: If a date is selected, the date is inputted in the box for holiday date.
- OK: The contents newly designated are saved and the screen is closed.
- Cancel: The contents newly designated are not saved and the screen is closed.
- Apply: The contents newly designated are saved and the work is continued.

Notice

☞ Any modification, which may be created during the edition of action trigger, is not reflected during the runtime and it is applied after the runtime is restarted.

Chapter 17 User Settings

Users are registered for each project and only the registered users are allowed to use InfoU. That is, the registered users are given authority to access to various InfoU functions but their access to some InfoU functions is limited according to their level of authority.

● Alarm Notification Function

The users registered in Alarm Group are notified through e-mail or SMS in case of the occurrence of any alarm.

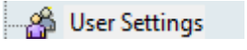
● Limited Functions by Level

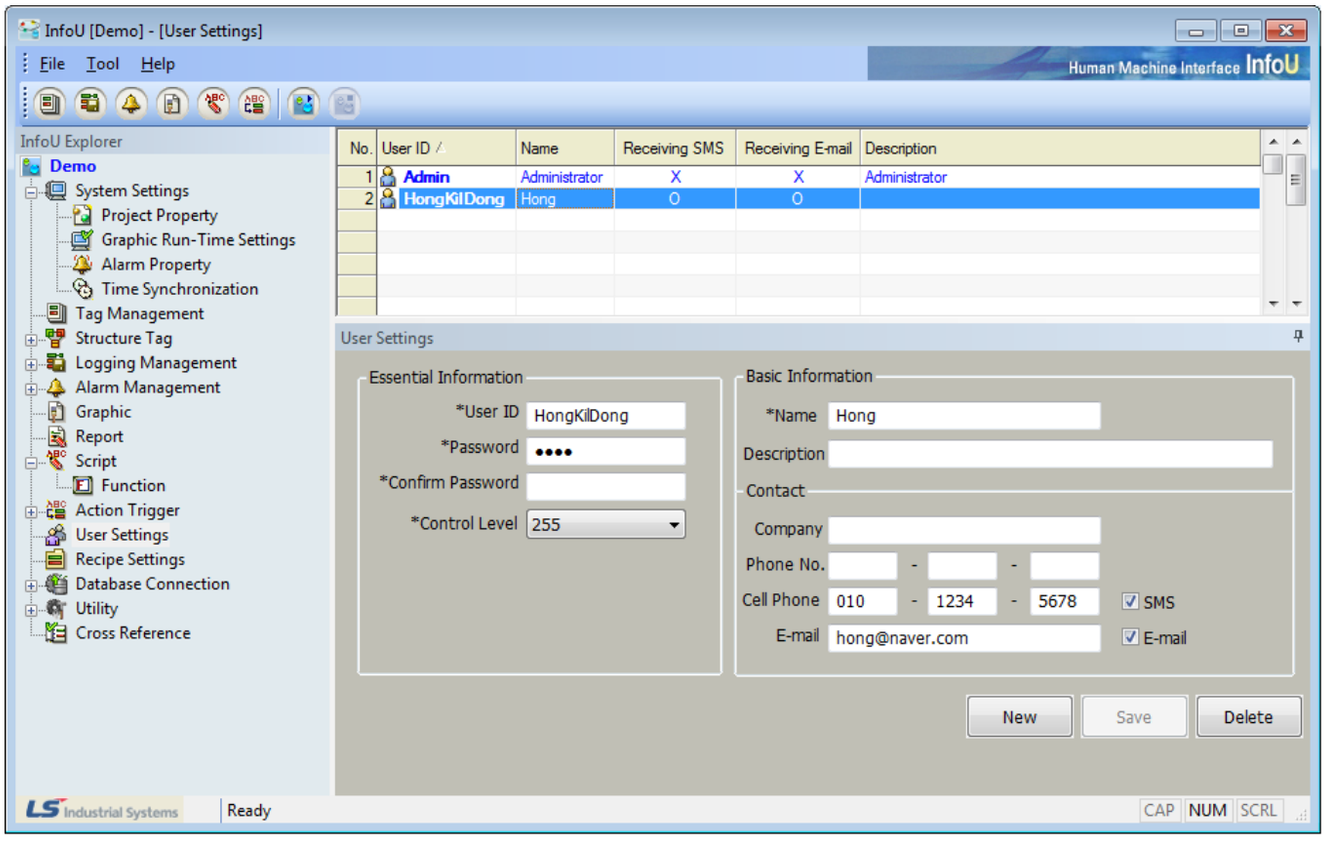
The same as tag levels (1~255), the users are assigned level 1 through 255 and according to this level, their authority to execute and use various InfoU functions such as tag control, inquiry and alarm acknowledgement is limited.

17.1 Setting Screen

(1) Start

Select User Settings in the InfoU explorer or the menu.





(2) User List

A list of the registered users is displayed to allow the user to delete or modify.

Notice

☞ Admin: System administrator's account as the user registered basically. This account can not be changed except password.

(3) User Settings

When newly registering a user or modifying, input information on the user..

1) Essential Information

a. User ID: Input User ID.

b. Password: Input password.

c. Confirm Password: Re-input password for confirmation.

2) Level (1~255)

Select a level. Level 1 through level 255 is available.

3) Basic Information

a. Name: Input the user name.

b. Description: Give additional information except the user name.

4) Contract

a. Company: The usesr's company

b. Phone No.: Contactable wire telephone number

c. Cell Phone: User's mobile phone number. This information is essential if Option SMS receiving has been selected.

d. E-mail: User's e-mail information. This information is essential if Option e-mail receiving has been selected.

e. SMS: Check √ in the box if the user wants to receive alarm information through SMS in case of the occurrence of alarm in the alarm group the user is registered in.

f. E-mail: Check √ in the box if the user wants to receive alarm information through e-mail in case of the occurrence of alarm in the alarm group the user is registered in

5) New

The user information is initialized to allow the user to input new information.

6) Save

The registered new information or modified information is saved.

In case of new registration, the user is added to the list.

7) Delete

The selected user's information in the list is deleted from DB.

Chapter 18 Recipe

Recipe is a function used when producing various products on the same production/process line according to the prescribed values and it defines a model and controls the process with the combination of component tags (units) and defined values (data). As a method used for large varieties but in small volumes, this function is applicable to various fields.

18.1 Component

To perform the recipe functions, the user needs to understand model, unit and data concept.

18.1.1 Model

It defines a recipe template on the same production/process line and it also defines a model and its components (items). After a model is defined, settings on the model items are not changeable if units and data have been already inputted

18.1.2 Unit

It defines a tag collection subordinated to the defined model and it also defines the tags that have the same number with that of the defined items for the model. That is, it has information on the set for the actually controlled points.

18.1.3 Data

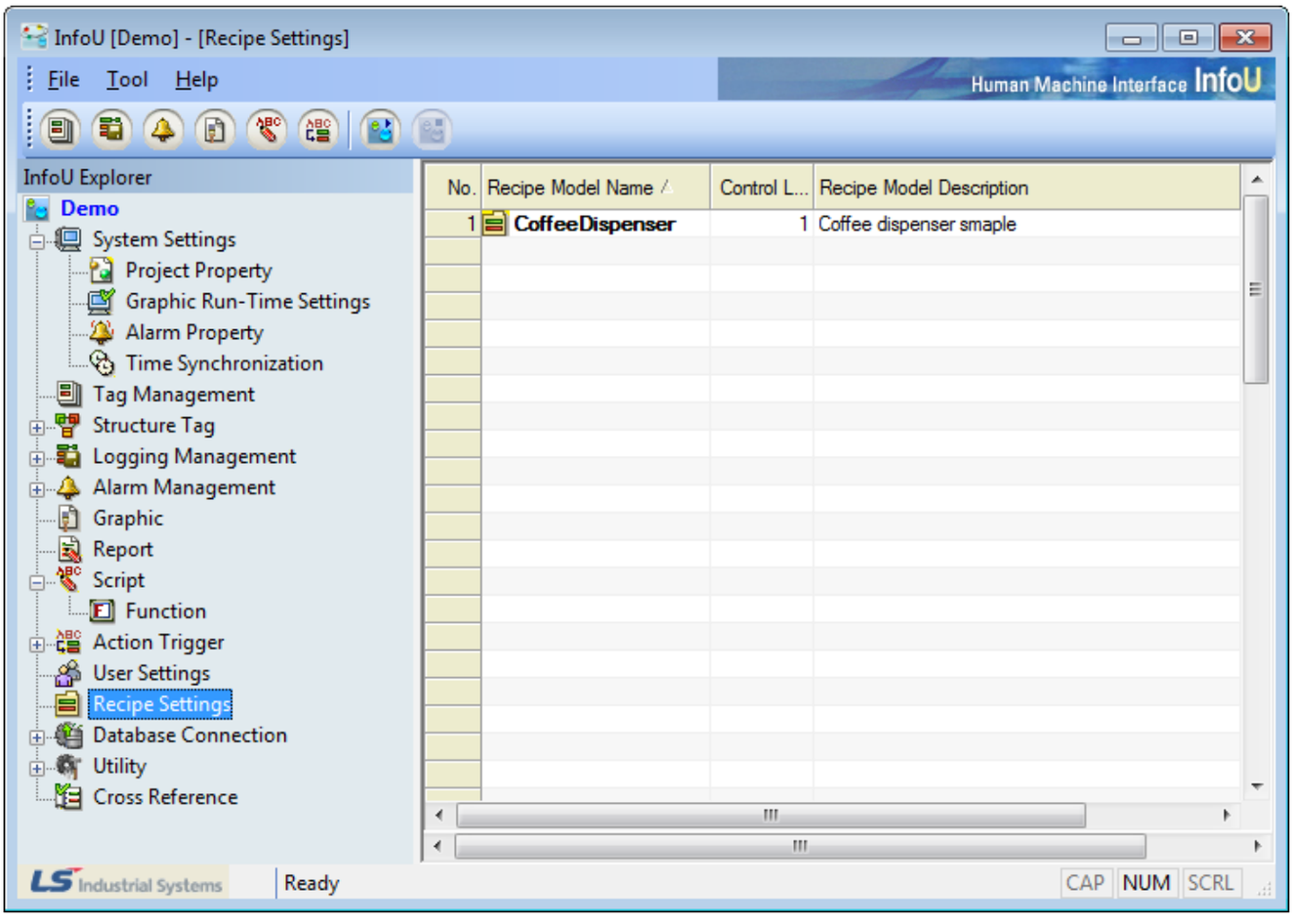
It defines a data set subordinated to the defined model and it also defines the actually controlled values.

● Example of Recipe Definition – Coffee Vending Machine

When a person is about to drink a cup of coffee from the vending machine, he/she needs to define its model.
The coffee vending machine (model) consists of three components- coffee, cream and sugar (item).
If the vending machine is in the building, its physical location (unit) has to be defined as a vending machine on the first floor or a vending machine on the second floor.
Then, the person needs to decide whether to drink black coffee, cream coffee or cream and sugar coffee and define data on the amount of coffee, milk or sugar that will be actually inserted.
If the person wants to drink a cup of milk coffee from the second floor vending machine, he/she can drink the coffee he/she wants at the place he/she wants with three combinations of model (coffee vending machine), unit (second floor) and data (milk coffee).

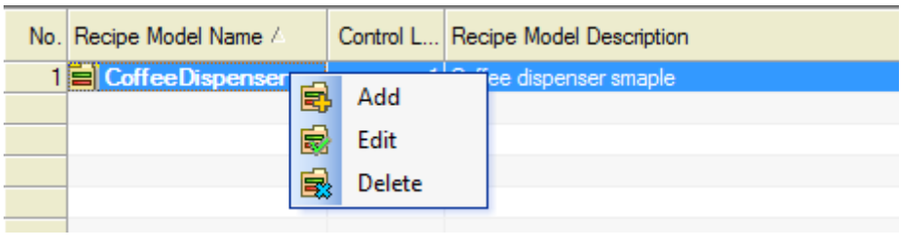
18.2 Settings

Select 'Tool'-'Recipe Settings' in the InfoU menu to display the screen as seen in [Figure].



A list of the registered recipe models appears on the screen and detailed information on the relevant model can be viewed or edited through double clicking on each item or pop-up menu.

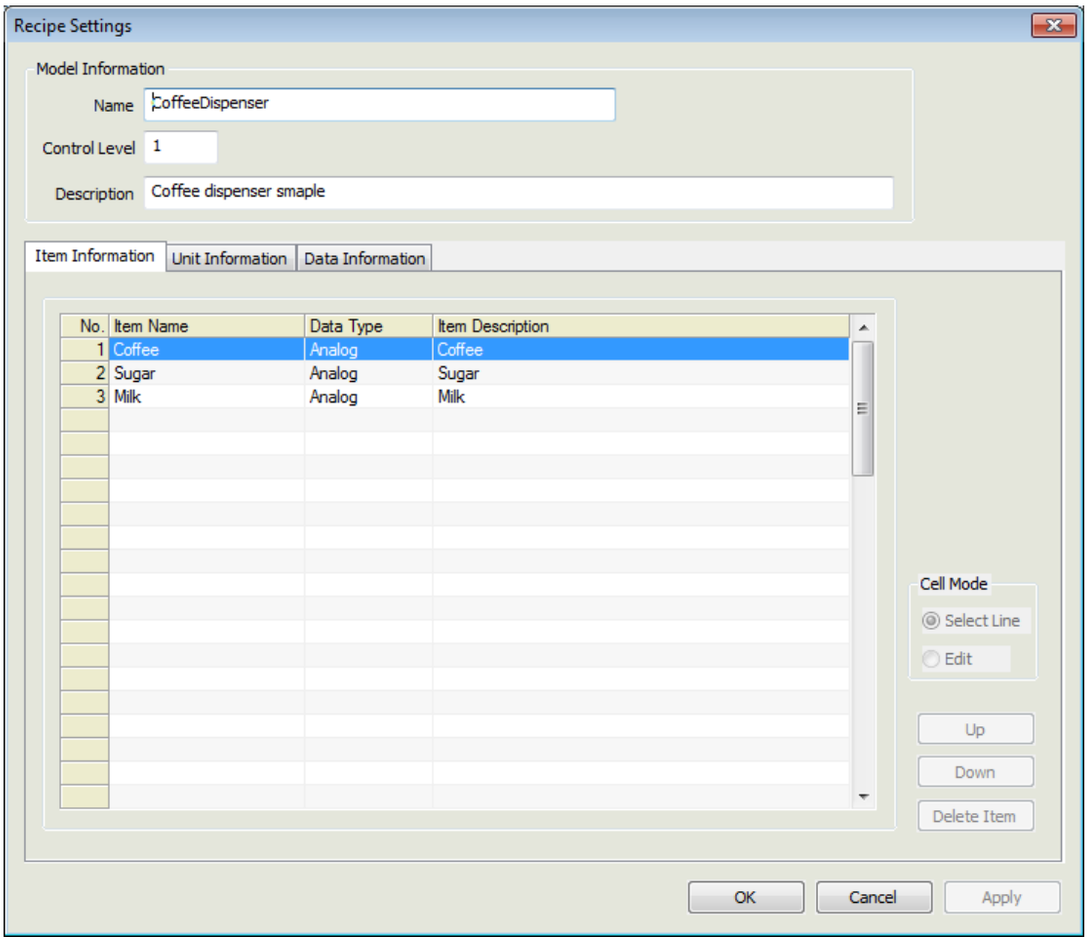
The following pop-up menu appears upon the click on the list screen.



- Add : This is a menu used when a new recipe model is registered. A blank screen appears if this menu is selected.Up to 64 models can be registered.
- Edit: This is a menu used when the information on the selected model is viewed or modified.
- Delete: If selected, a window appears to ask the user to confirm deleting. If "Yes" is selected, the relevant model is deleted and the units and data that belong to the model are also deleted.

18.2.1 Model Information

A screen to set up a recipe model. Item on the model and three items such as item, unit and data are divided with taps.



(1) Property

- Name: Register a name of the model. Duplication is not allowed.
- Control Level : Input a control level 1 through 255. The level of the InfoU user shall be higher than the control level of the recipe model to control the recipe (Level 1 is the highest one and Level 255 is the lowest one).
- Description: Register description on the recipe model.
- Item Information: Register items (components) that consist of the model. Up to 64 items are available to register for one model.
  - Item Name: A name of the item
  - Data Type: Nature of the item data. Select one among Analog, Digital and String.
  - Item Description: Description on the item

(2) Function Button

- Cell Mode
  - Select Line: An option used when selecting an item as a line mode.

- Edit: An option used when editing grids. The user may directly input information on the grid by double clicking on the grid or selecting ‘Edit’ button.

● Up: It is activated upon the selection of an item and used for moving the selected item to higher order..

● Down: It is activated upon the selection of an item and used for moving the selected item to lower order.

● Delete Item: It is activated upon the selection of an item and used for deleting the selected item..

Notice

☞ If the linked unit or data is added after a model is defined, the model can not be modified.

18.2.2 Unit Information

A screen to register a recipe unit

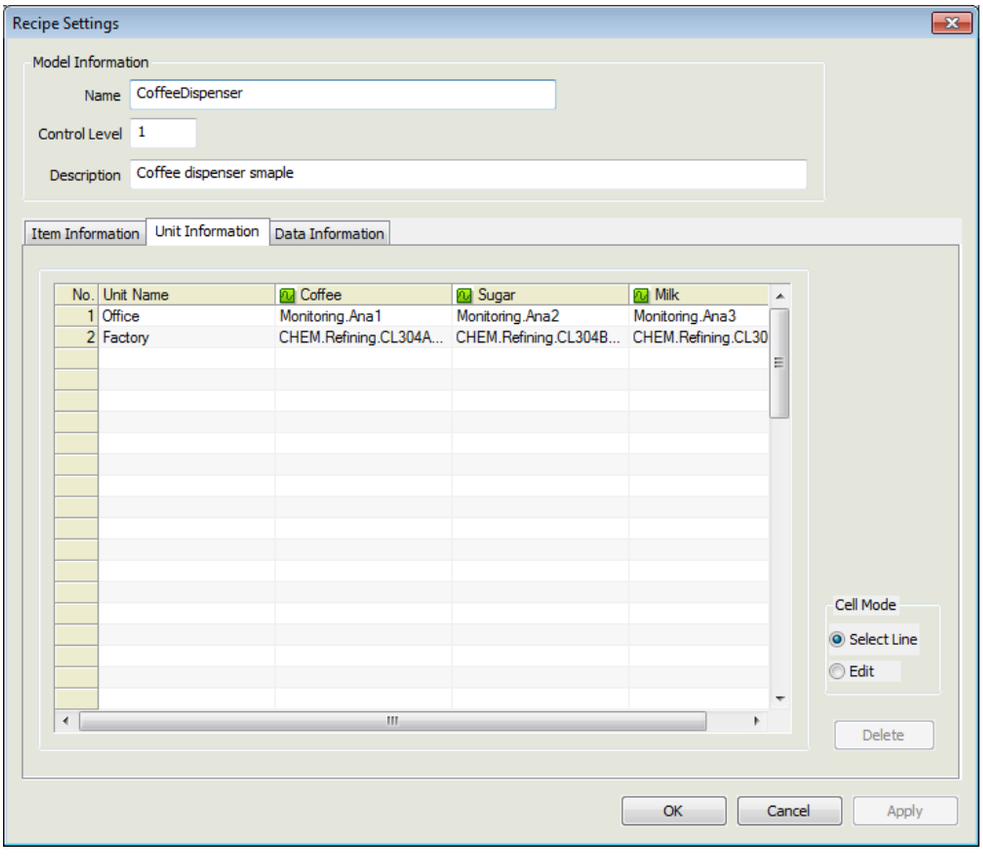
Here, a unit means a tag collection and 10 tags and memory tags can be used.

First, items that have been defined for a certain model appear on the unit screen and unit tags suitable for each item type are registered for the relevant unit.

An editing method to directly input in each cell is used.

Up to 64 unit sets are available for registering (64 unit sets per model).

As seen above, since the item number available to register for one model is 64, total 64 X 64 tags can be registered on the unit definition screen.



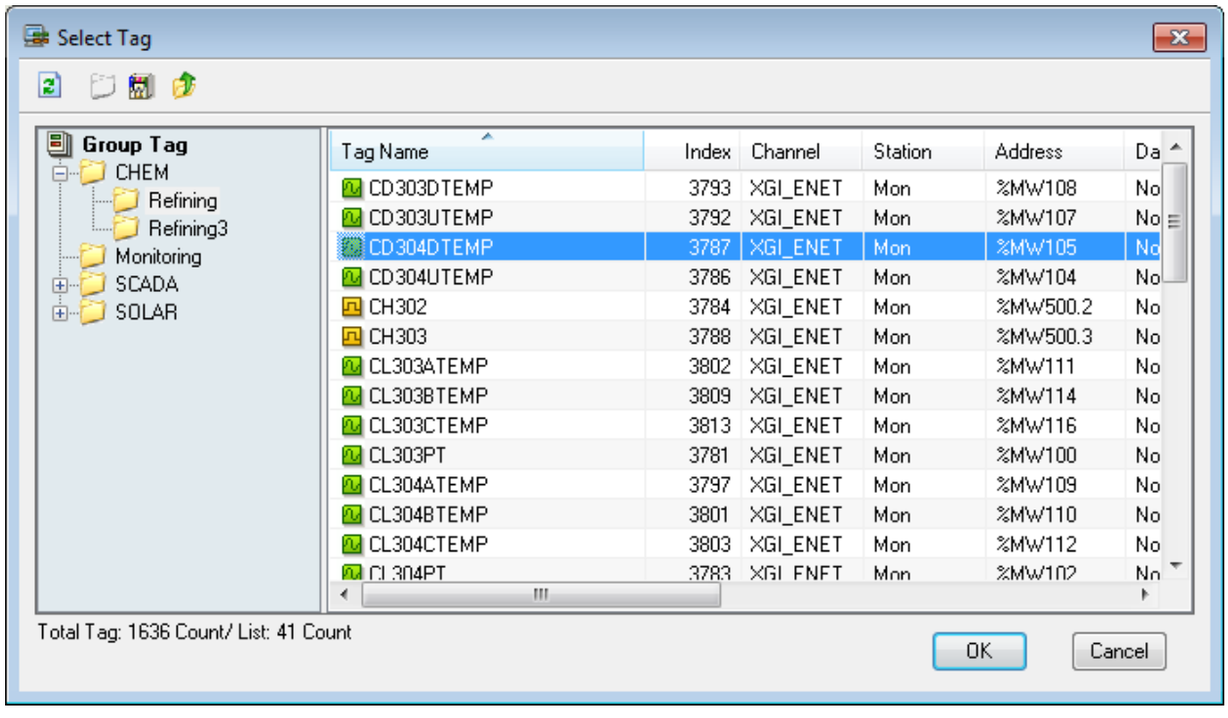
(1) Property

● Unit Name: Register a name of the unit.

● Unit: A screen to register units (tags) suitable for the item type defined for the model.

Directly input the tag name in the cell or click the cell […] button to select a tag to register.

If […] button is clicked, the tag selection screen appears and if a tag is double clicked on the tag selection screen, the screen disappears and the tag is marked on the grid.



(2) Function Button

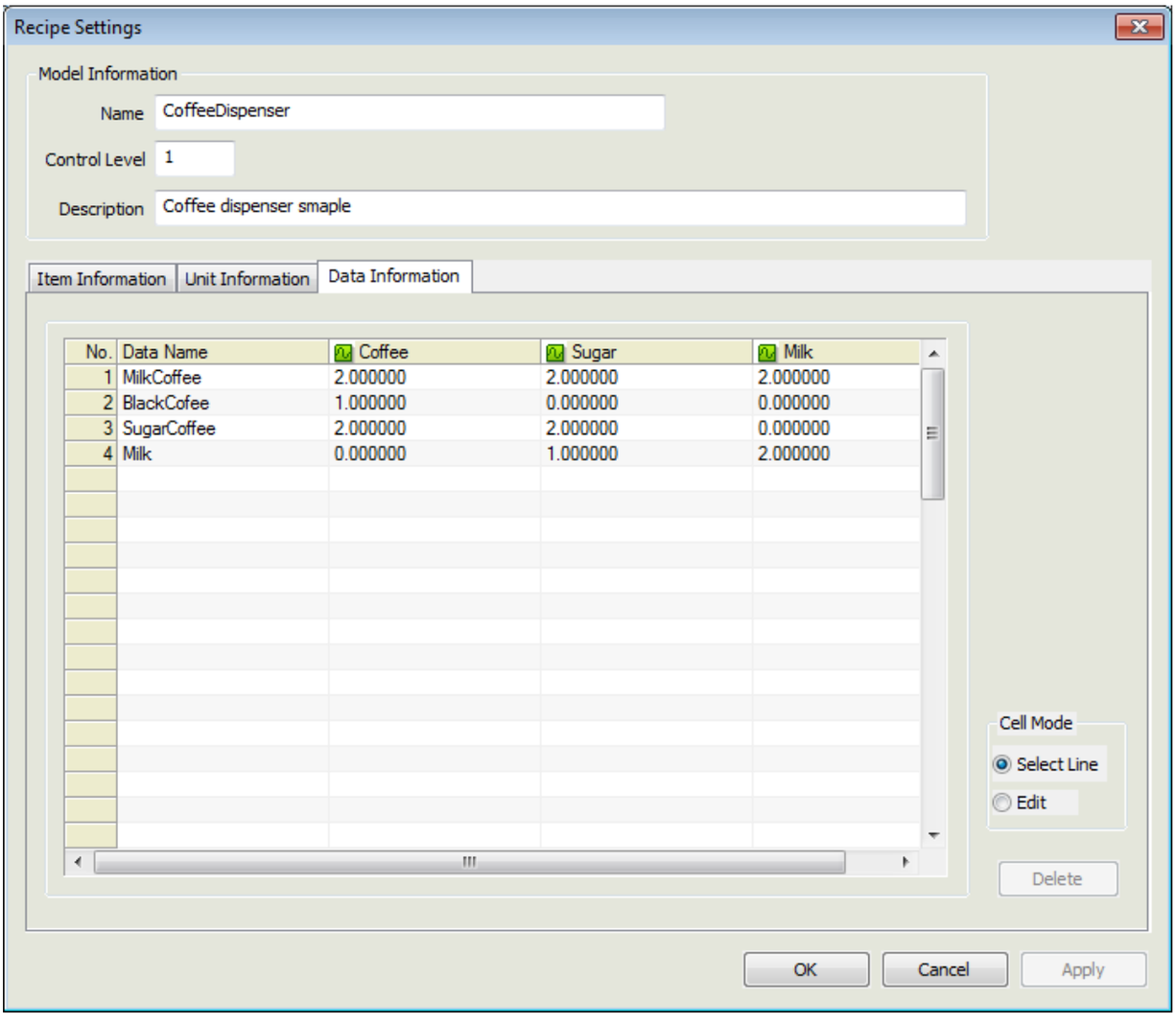
● Delete: The currently selected unit name is deleted.

18.2.3 Data Information

A screen to register a recipe data

Here, data mean the values that are actually controlled. Register a value suitable for the item type defined for the model. An editing method to directly input in each cell is used as done for unit information.

Up to 64 data per model are available for registering.



(1) Property

- Data Name: Register a name of the data..
- Data: Define the control value suitable for the item type defined for the model..

(2) Function Button

- Delete: The currently selected data name is deleted.

18.3 Recipe Runtime

To use recipe functions online, InfoU provides two methods- one is to use OCX components and the other is to use the internal functions. In addition, they can be used in the Script and Action Trigger.
Recipe functions that can be performed in the Runtime are broadly divided into three functions; Working Set, Real-time Data Inquiry and Saving, and Recipe Control.

18.3.1 Function

(1) Working Set Composition (Working Set)

- The recipe uses a virtual collection concept called working set.
- The recipe's essential components are model, unit and data. Those three components are combined to perform recipe control.
- For OCX, relevant components are clicked in the list to form a working set and the internal functions corresponding to each function have been already defined.

(2) Real-time Data Inquiry and Saving

- This is a function to import the defined tags for the unit from the real-time server during the runtime to form a working set.
- The working set is formed with models and units and this function does not only import the predefined data from DB but also inquire the current tag values.
- Surely, the real-time data can be saved in DB. But, only up to 64 data are available per model.

OCX does not provide the 'Delete' function.

(3) Recipe Control

- The purpose of forming a working set is to perform recipe control.
- Recipe control is ready for performing once the combination of the prescribed model, unit and data (real-time data) is equipped with.
- Recipe viewer OCX can perform simple control in InfoU as well as direct control is also available by using data saved the control DB using the working set.

18.3.2 Recipe Viewer OCX

Recipe Viewer OCX provides specialized functions at control. For detailed functions, refer to 'Recipe Viewer of Graphic Object".

18.3.3 Internal Function

Recipe functions can be implemented by internal functions.
For internal functions that can perform recipe functions during the runtime, refer to 'Function Help.

Chapter 19 Database

19.1 Database Connection

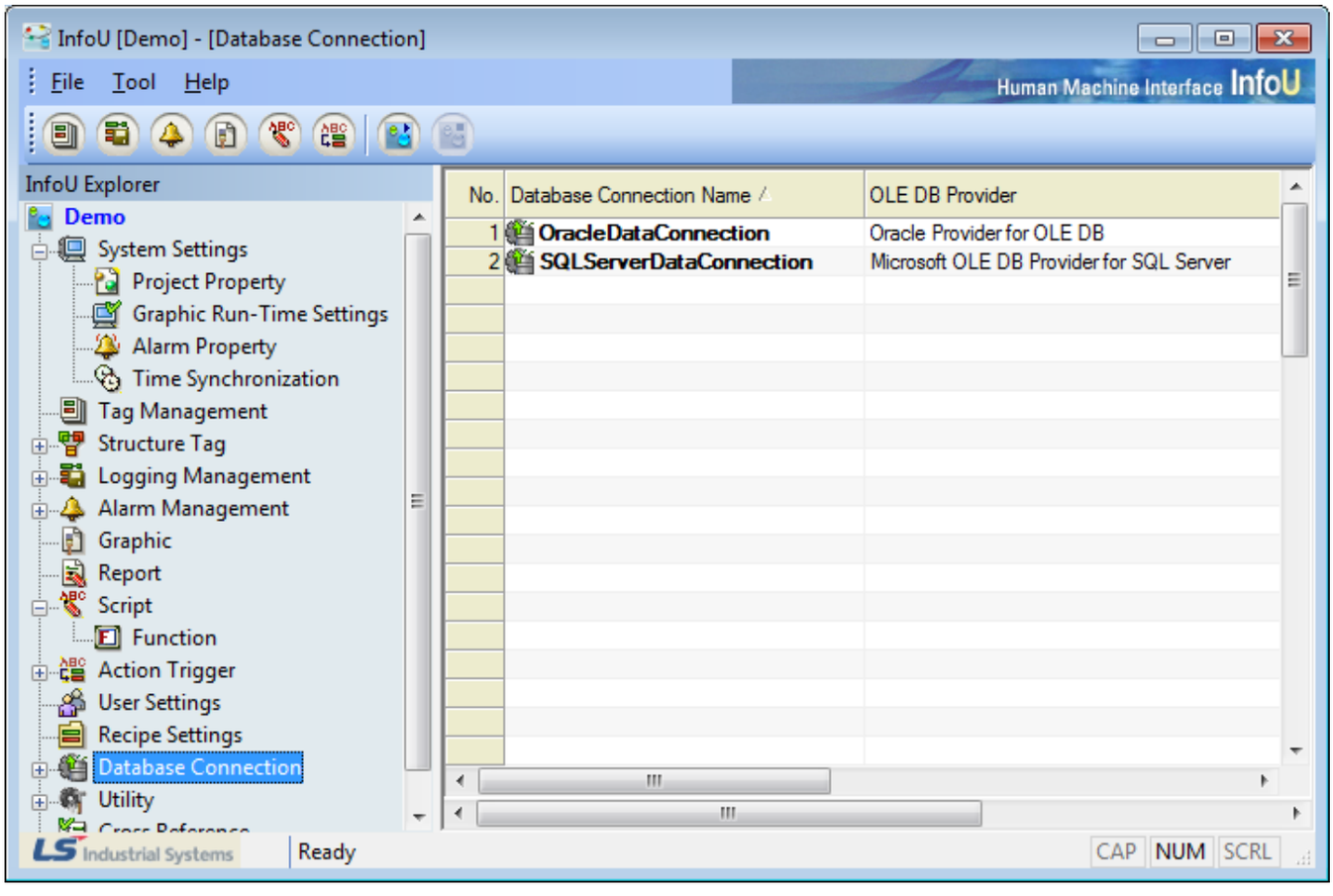
This Database Connection is a function to connect this InfoU system to universal database systems such as SQLServer and Oracle by using Microsoft OLE DB interface.

InfoU can define up to 64 connections.

For Database Operation, it is necessary to define connection with external databases in advance and the connection is used as follows.

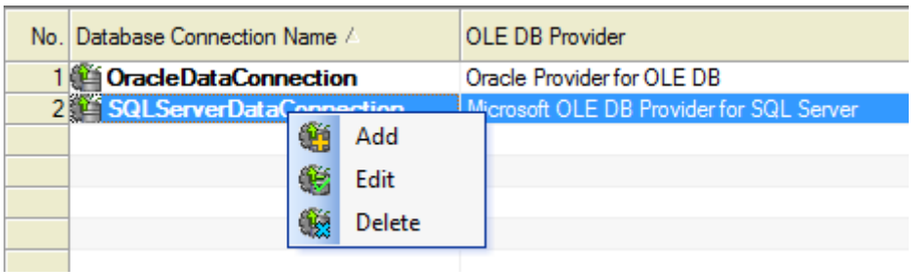
19.1.1 Demo

Select 'Tool'- 'Database Connection' in the InfoUD menu, the screen as seen in the [Figure] below is displayed. This 'Database Connection' menu is displayed on the lower-level of the Database Operation.



A list of the registered database for 'Database Connection' appears on the screen and detailed information on the relevant database can be viewed or edited through double clicking on each item or pop-up menu.

The following pop-up menus appear upon the click on the list screen.



(1) Add

- This is a menu used when a new database is registered for ‘Database Connection’. If this menu is selected, an empty screen is displayed.
- Up to 64 databases can be registered for ‘Database Connection.’.

(2) Edit

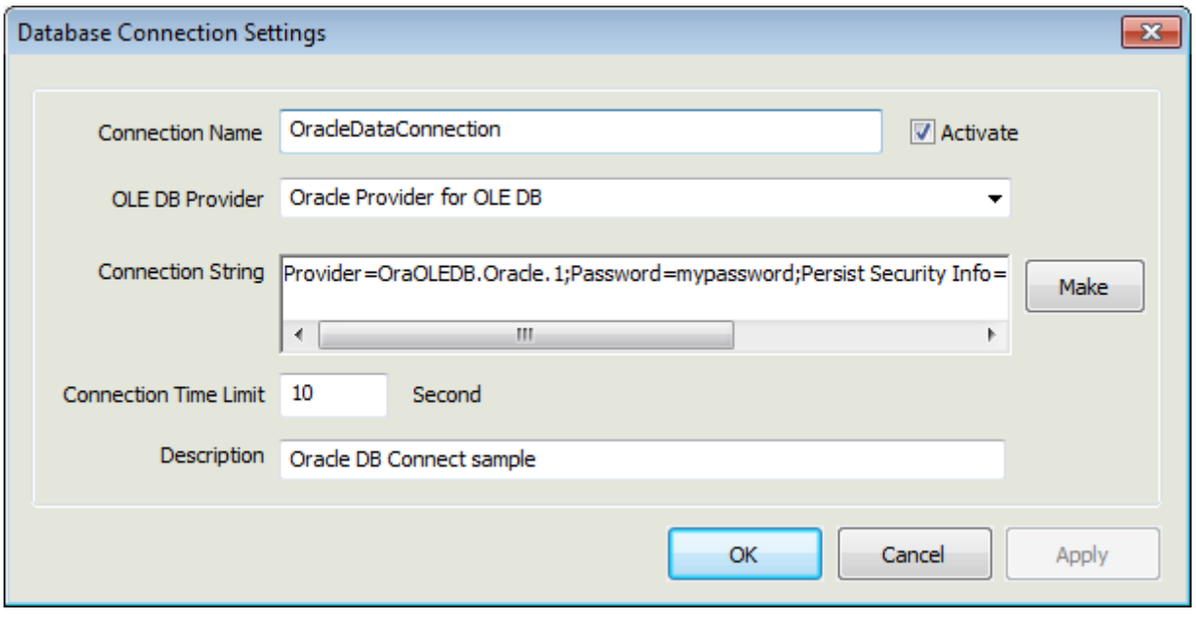
- This is a menu used when the information on the selected database is viewed or modified.
- If one of existing databases is renamed and saved, the database is automatically added and saved as a new database.

(3) Delete

- If selected, a window appears to ask the user to confirm deleting. If “Yes” is selected, the relevant database is deleted from the ‘Database Connection’ list and the ‘Database Operation’ that belongs to the database is also deleted..

19.1.2 Database Connection Settings

This screen is the one to define ‘Data Connection.’ Double click on the list or select the pop-up menu to display the following screen. Each item and button is explained as follows



(1) Connection Name

- Register its own name for identification and any duplicated name is not allowed.

(2) Active

- This ‘Database Connection’ is attempted only for activated data when the database engine is running. The user may register several Data Connections and designate this ‘Activate’ option only to the necessary connections to prevent the external database support from being wasted unnecessarily.

(3) OLE DB Provider

- Register OLE DB providers. The combo box shows the list of the registered OLEDB providers in the local computer and the user may either select one from the list or input the name of the OLEDB provider directly on the combo box..

(4) Connection String

- Register a Connection String used for the relevant OLEDB provider. (If the user does not know a Connection String, he/she may create a file with extension ‘.udl' and double click it to display a screen to define ‘OLEDB Data Connection.’ If the user opens the .udl file on the notepad after setting up each item as shown on the screen, he/she may check whether the Connection String has been saved. The user may copy and register this syntax. But, he/she needs to select [Allow saving password] on the checkbox of [Connect] tap to ensure the password is also saved when the .udl file is saved).

(5) Connection Time Limit (Second)

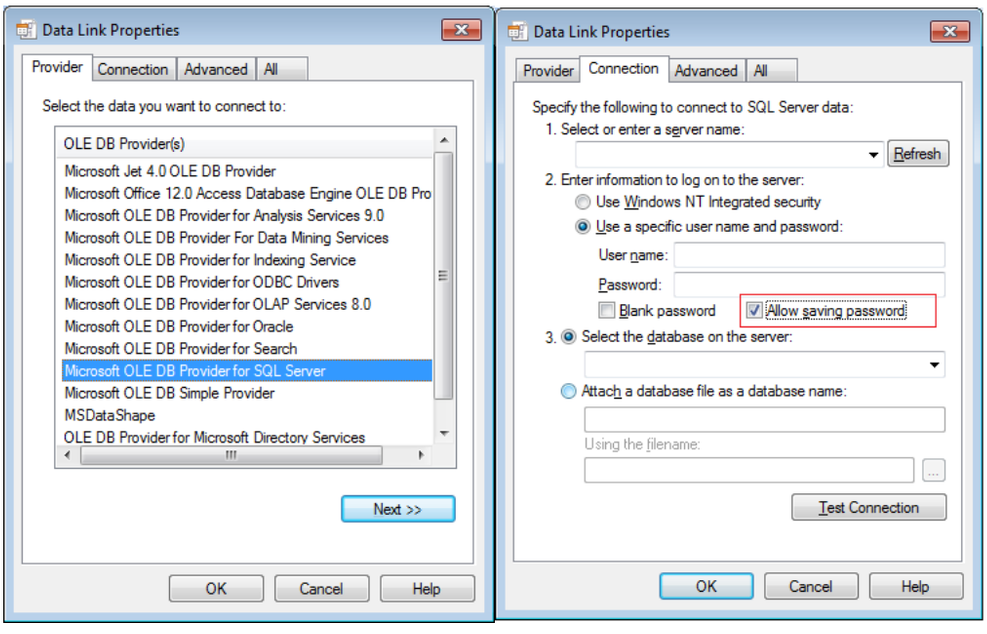
- Register a limit time when attempting connection with the designated database. The user needs to input a value higher than 0. If 0 is inputted, connecting is attempted endlessly until it is finally succeeded..

(6) Description

- Input description on the ‘Data Connection.’.

(7) Make

- Click ‘Make’ button to display a screen to define ‘OLEDB Data Connection.’ If [OK] button is clicked after setting Connection Strings in the order, those are automatically registered on the 4) Connection String text box. (If Check box [Allow saving password] is selected on the [Connection] tap, the password can be included in those Connection Strings.)



[Figure :Screen of Data Link Properties]

19.2 Database Operation

Database Operation is a function to ensure the InfoU system links to universal database systems such as SQLServer and Oracle. That is, the user can record and save InfoU tag values in an external database or import records from the external database to map to InfoU tag values.

For Database Operation, the user may use the predefined Database Connection and register up to 256 operations.

19.2.1 Operation Type

Database Operation can be divided into 'SELECT' Operation and 'ETC' Operation according to operation properties.

(1) SELECT

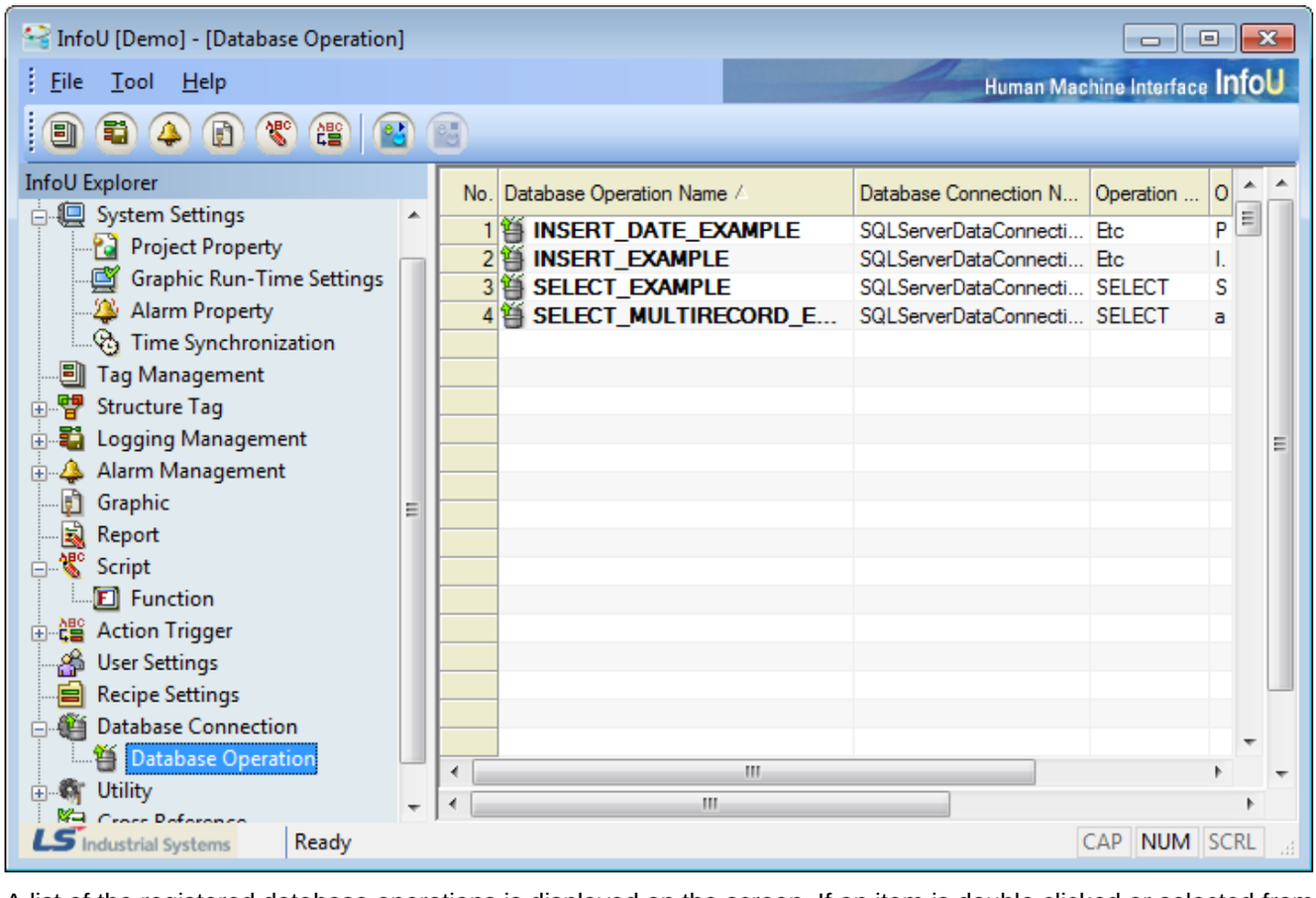
- This operation is to import records from an external database and save them in InfoU tag values.

(2) ETC

- This operation is to record and save InfoU tag values in an external database.

19.2.2 Demo

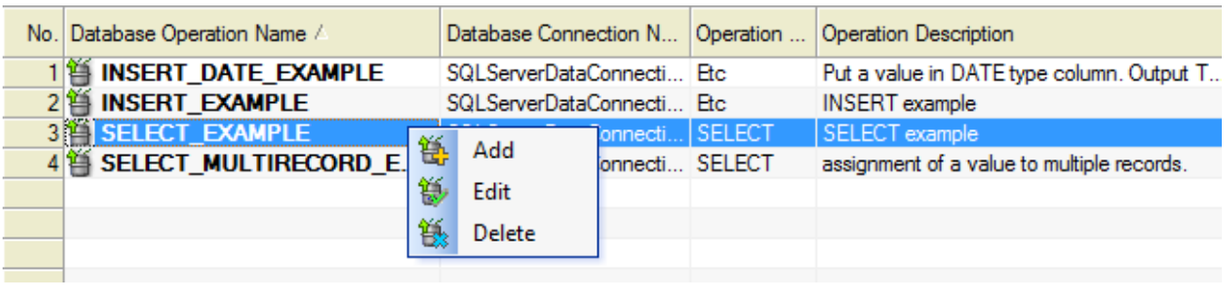
Select 'Tool'- 'Database Operation' in the InfoUD menu to display the following screen..



A list of the registered database operations is displayed on the screen. If an item is double clicked or selected from

the pop-up menu, the detailed information on the relevant operation can be viewed or modified.

The following pop-up menu appears upon the click on the list screen.



(1) Add

- A menu to be used when registering a new database operation. An empty screen appears if this menu is selected. Up to 256 database operations can be registered

(2) Edit

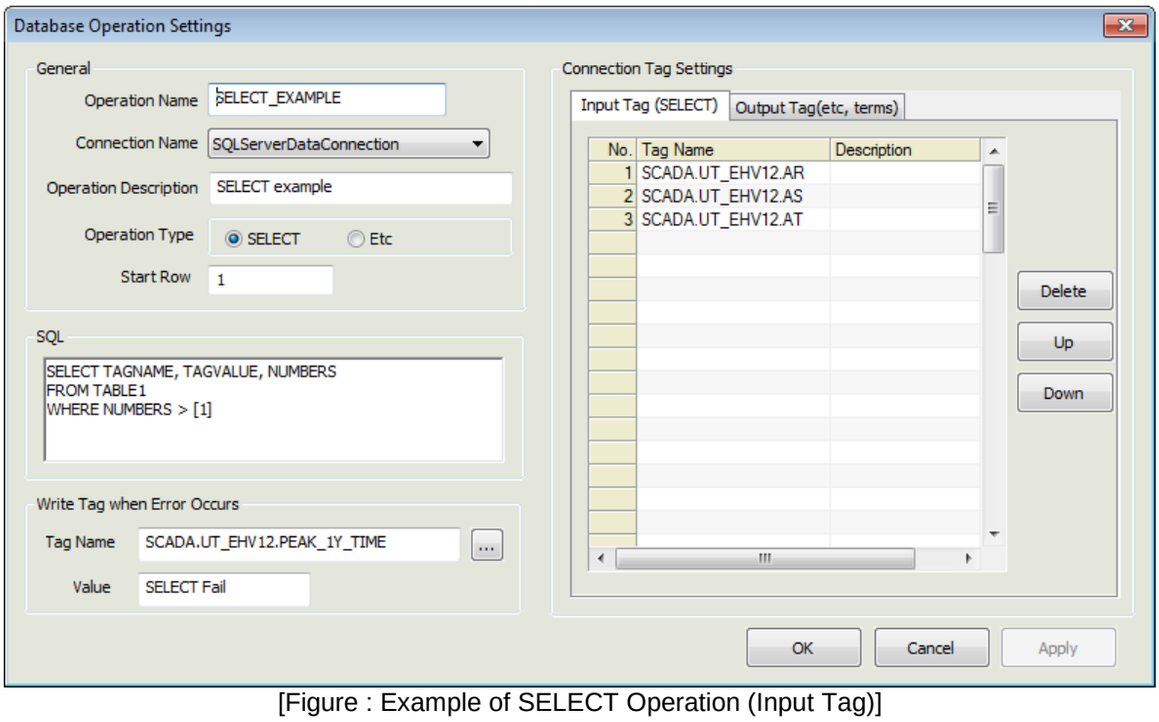
- This is a menu used when the information on the selected database operation is viewed or modified.

(3) Delete

- If selected, a window appears to ask the user to confirm deleting. If “Yes” is selected, the relevant database operation is deleted.

19.2.3 Database Operation Settings

A screen to define ‘Database Operation’ Database Operation can be divided into ‘SELECT’ operation used to import data from an external database and set up them as tag values and ‘ETC’ operation used to conduct operation for an external database with the InfoU tag data.



[Figure : Example of SELECT Operation (Input Tag)]

(1) Operation Name
- This is the name of a database operation and it has a unique value.

(2) Connection Name
- Select a name of the Database Connection to be used for the relevant operation.
- The Connections set up in the Database Connection Settings are displayed on the combo box. If no Connection is set up, the database operation can not be saved. (Set up 'Database Connection' first to define 'Database Operation')

(3) Operation Description
- Describe the operation.

(4) Operation Type
- Specify a type of the Operation.
- For 'SELECT', input and output tag taps are activated on the ''Connection Tag Setting' window.
- For 'ETC', only output tag tap is activated on the window (The input tag is not activated)

(5) Start Row
- When a record set is imported from a database, the user can define a record start number that will be a baseline when extracting data of the record set (The value shall not less than 1)
- For example, if there are five record sets and the start row is 1, tag mapping starts from the first record while if the start row is designated 2, it starts from the second record, skipping the first record.

(6) SQL Text
- Input a proper SQL syntax (Refer to the following detailed information on SELECT and ETC)

(7) Tag Name
- The tag name that will record errors if the operation fails.

(8) Value
- The tag value that will record errors if the operation fails.

(9) Input Tag
- This menu is activated only when 'SELECT' is selected.
- It is used when importing external data and controlling them with tag values.
- It is used when importing external data and controlling them with tag values.
- Date of the record set is applied in the order of input.

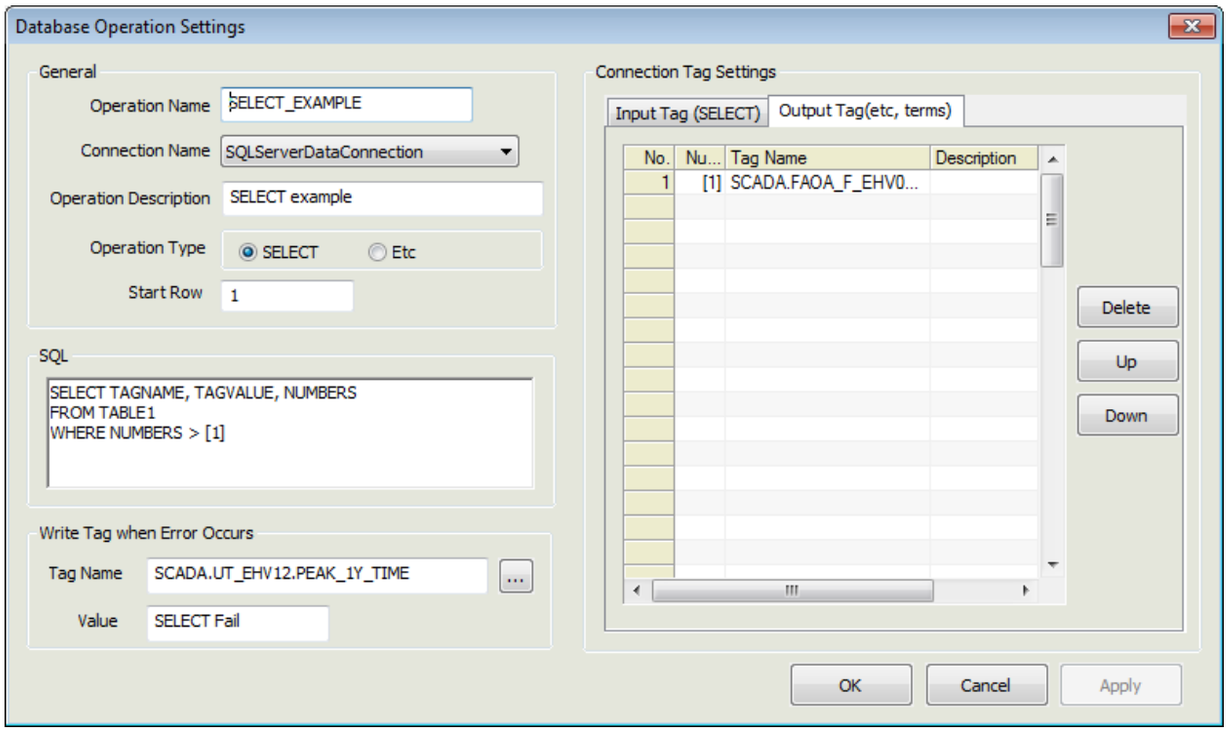
(10) Output Tag
- This menu is activated only when 'SELECT' and other is selected.
- Up to 64 tags per operation can be registered.
- It is used when carrying out operations external data with InfoU tag data.
- Input order is not important. (Tag reservation numbers of SQL syntax and registered output tag numbers are mapped.)
- If the same tag reservation numbers are repeated in the SQL syntax, the corresponding tags to those numbers are also repeated.

(11) Delete
- Delete the selected tag.

(12) Up
- Move the selected tag one step up.

(13) Down

- Move the selected tag one step down.



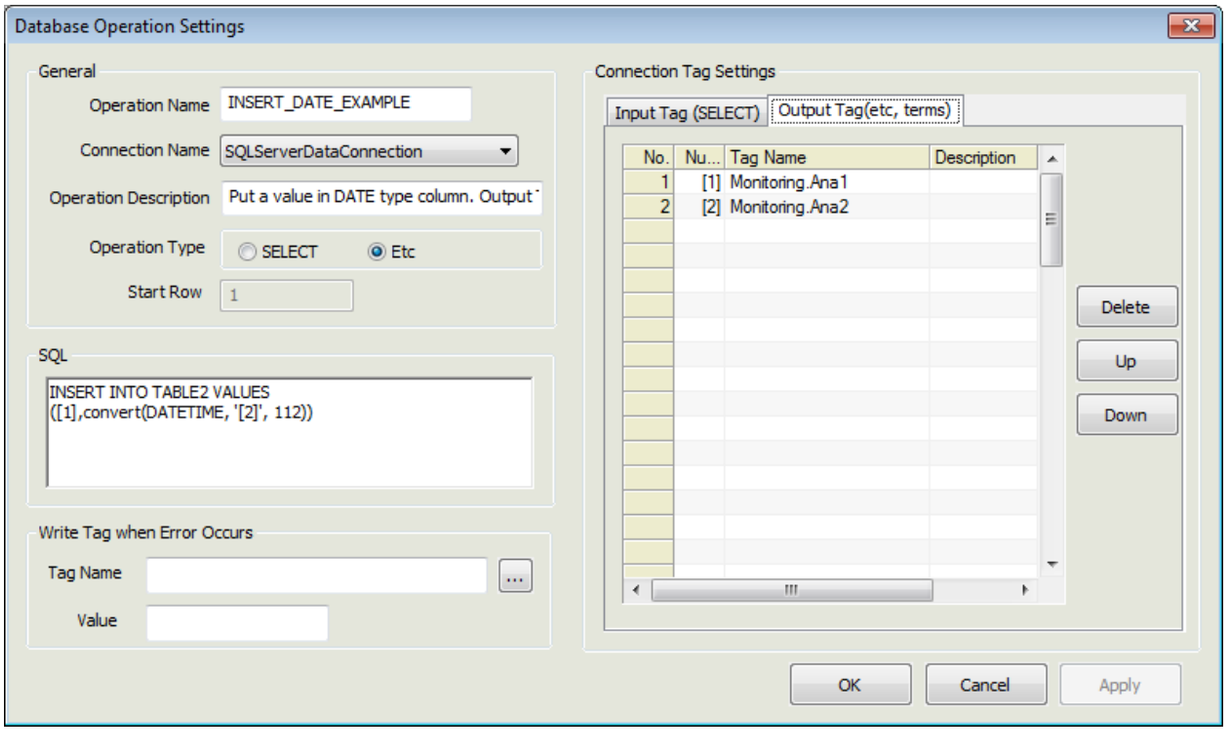
[Figure : EXAMPLE OF SELECT OPERATION(Output Tag)]

(14) SELECT

- It is used only when the operation type is 'SELECT'.
- The column name next to 'SELECT' is mapped with the tag defined on the phase of Input Tag one by one.
- Tag reservation numbers ([digit]) derived from FROM clause or WHERE clause are mapped with the output tags.
- Tag reservation numbers (numbers higher than 1) are recorded in positive numbers.
- Tag reservation numbers or the registered tag values (digit, character..) are expressed properly such as using quotation marks in the SQL syntax according to properties of each DB.
  (ex.1) In case that tag values to be used are digits: SELECT COL1 FROM TAB1 WHERE ID = [1]
  (ex.2) In case that tag values to be used are characters: SELECT COL1 FROM TAB1 WHERE ID ='[1]'
- The record set to be inquired starts from the location where the Start Row is registered and controlling tag values starts from that location.
- Controlling tag values is applied in the order of being registered for input tags (It is not necessarily matched between the data number and the input tag number)

- Example of application

Case	SQL	Input Tag	Query Result	Tag Value Control Result
If the input tag number is less than the inquiry data number (Start Row 1)	SELECT ID, NAME FROM TABLE1	Tag01 Tag02 Tag03	ID NAME ---------- 1 Hong Gil Dong 2 KooSangYeop 3 SCOTT	Tag01 = 1 Tag02 = Hong Gil Dong Tag03 = 2
If the input tag number is less than the inquiry data number (Start Row 2)	SELECT ID, NAME FROM TABLE1	Tag01 Tag02 Tag03	ID NAME ---------- 1 Hong Gil Dong 2 KooSangYeop 3 SCOTT	Tag01 = 2 Tag02 = Hong Gil Dong Tag03 = 3
If the input tag number is more than the inquiry data number	SELECT ID, NAME FROM TABLE2	Tag01 Tag02 Tag03 Tag04 Tag05 Tag06	ID NAME ---------- 1 Hong Gil Dong 2 KooSangYeop	Tag01 = 1 Tag02 = Hong Gil Dong Tag03 = 2 Tag04 = KooSangYeop



[Figure: Example of ETC Operation]

(15) ETC (INSERT, UPDATE, DELETE)

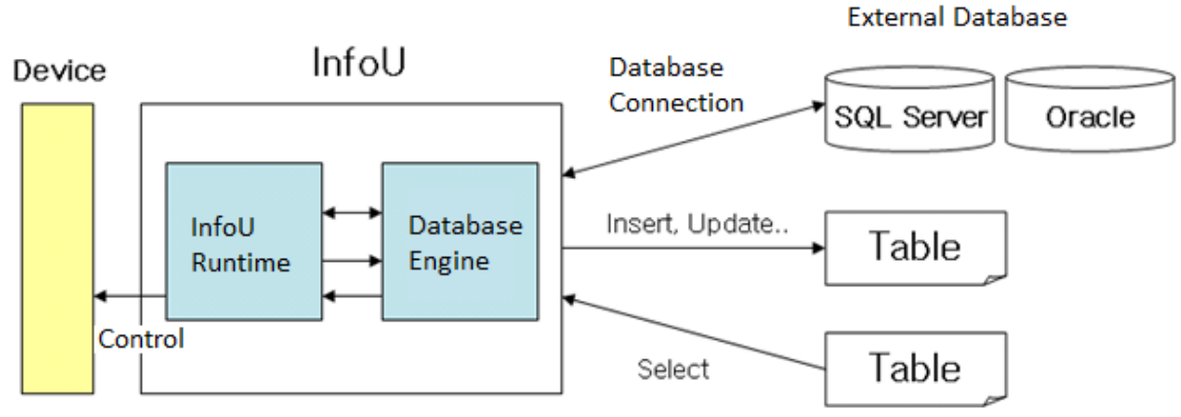
- It is used only when the operation type is ‘ETC.’
- It can not used for input tags.
- Tag reservation numbers ([Number]) used in this syntax are mapped with output tags
- Tag reservation numbers are recorded in positive numbers equal to or higher than 1.
- Tag reservation numbers or the registered tag values (digit, character.) are expressed properly such as using quotation marks in the SQL syntax according to properties of each DB.
  (ex.1) In case that tag values to be used are digits: INSERT INTO TAB1 VALUES([1], [2])
  (ex.2) In case that tag values to be used are characters: INSERT INTO TAB1 VALUES(‘[1]’, ‘[2]’)
- Example of application

Case	SQL	Output tag and value	Query Result
If tag reservation numbers are matched with output tags	INSERT INTO TABLE1 VALUES ([1], [2], ‘[3]’)	[1] Tag01 20 [2] Tag02 30 [3] Tag03 40	INSERT INTO TABLE1 VALUES (20, 30, ‘40’)
If tag reservation numbers are not matched with output tags	UPDATE HYTAG SET VALUE = [1] WHERE NAME = ‘[2]’ AND TIME = [3]	[2] Tag01 50 [1] Tag02 Dong	UPDATE HYTAG SET VALUE = 50 WHERE NAME = ‘Dong’ AND TIME = [3]
If tag reservation numbers are repeated	INSERT INTO TABLE1 VALUES ([1], [2], [1])	[1] Tag01 20 [2] Tag02 30	INSERT INTO TABLE1 VALUES (20, 30, 20)

19.3 Database Runtime

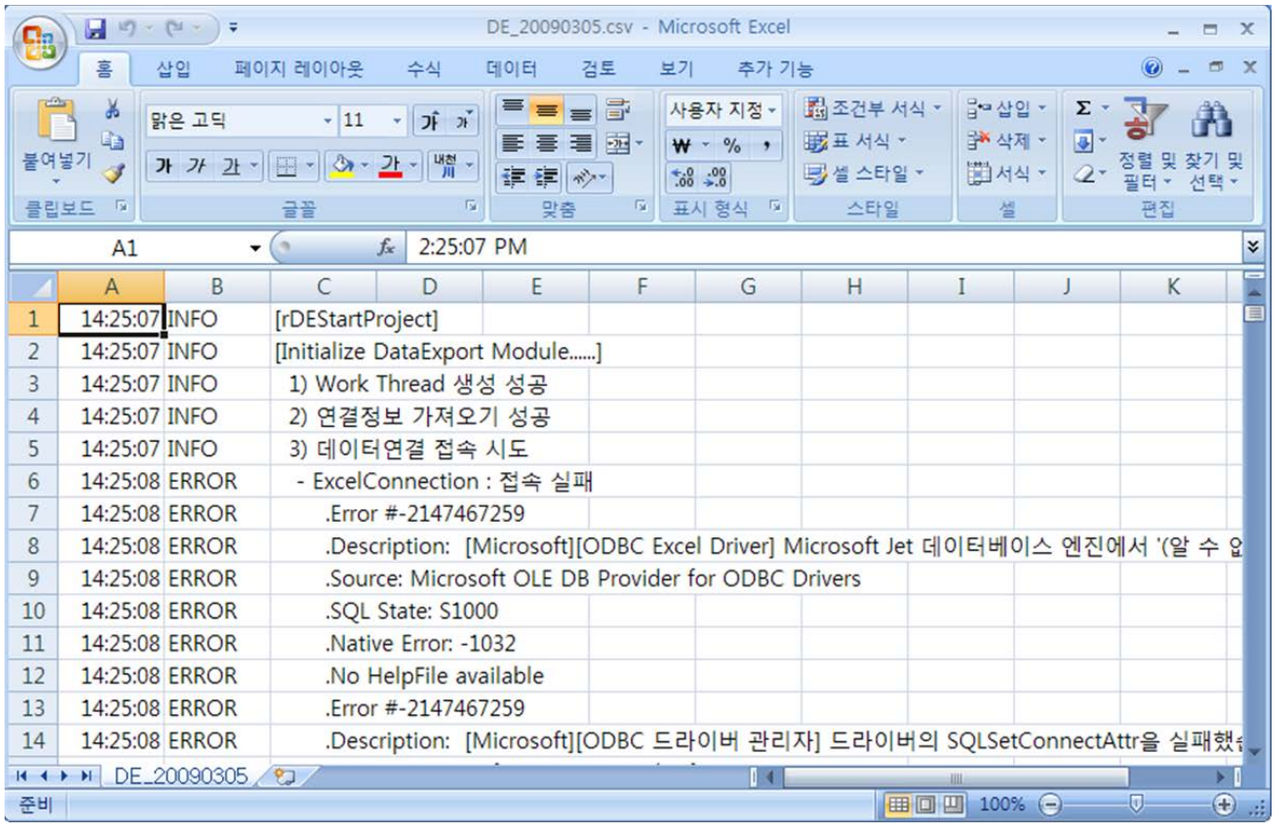
Database Runtime plays a role in performing operations linked to external databases during the Runtime by using the Connect Data and Operation Data set up during the engineering.

The user can perform Database Runtime Function in the stages of ‘Script’ or 'Action Trigger,’.



The log of the database work is daily created as the .csv file in the log folder of the project path.

This log includes the information on success/failure of the external database, success of each database work with the error information. If database task does not work normally, a user can open this file to determine the exact cause.



19.3.1 Function

(1) SELECT

- This function is to inquire data from the external database and control them as InfoU's internal tag data.
- Even though the inquired data number is not equal to that of the registered input tags, it can control them by extracting common data.

(2) ETC

- This function is to record and save InfoU tag values in an external database. SQL syntaxes supported by each database can be used in various ways.

19.3.2 Internal Function

The internal functions that perform database operations are supported. For details and how to use, see 'Help' of 'Internal Function.'.

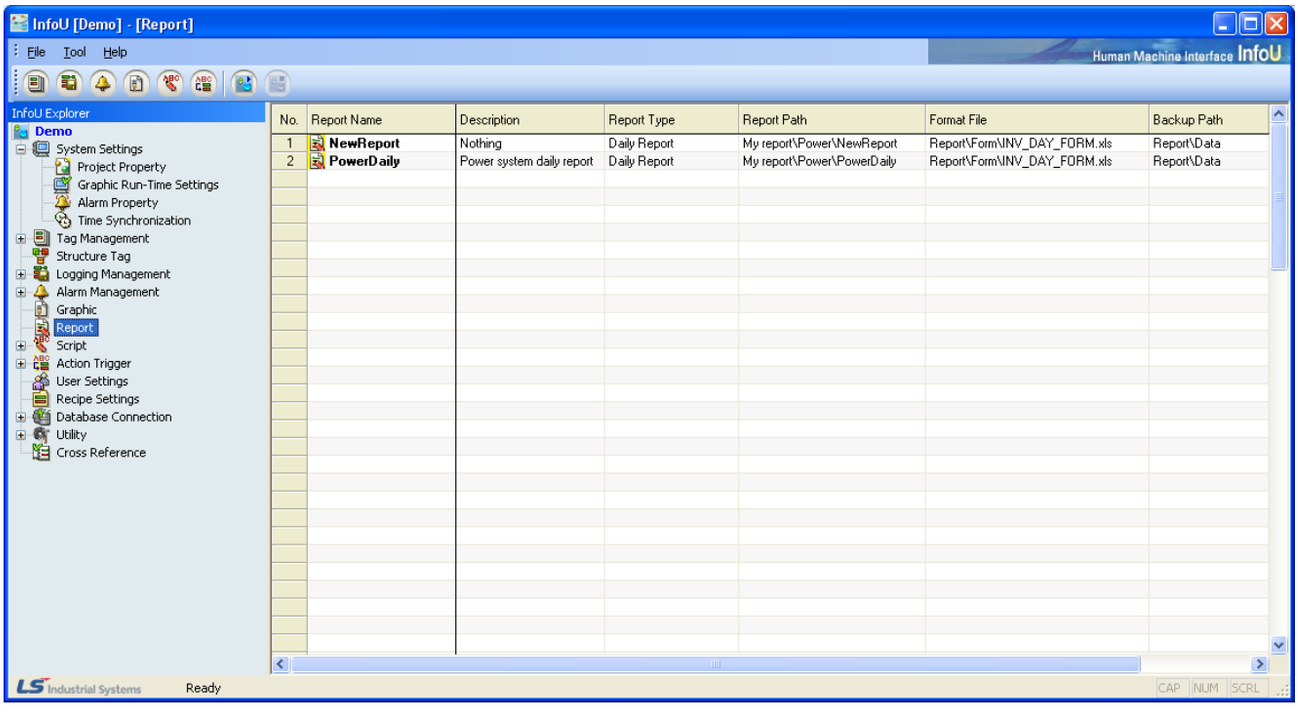
Chapter 20 Report

InfoU Report is to inquire the logging data defined in the logging group and create or output Excel files at a designated time. It can be outputted at any desired interval such as hourly, daily, weekly, monthly, yearly or freely.

- Conveniently create similar reports with the report copy function.
- Report layout formation is easy to use with dragging and dropping function.
- Editing is also available with copying/pasting to Excel.
- Since Excel is used, graph, diagram, formula, statistics and other functions supported by Excel program can be utilized.
- Since the tags used for report can be cross-referred, it is possible to check which tags are used when deleting or modifying tags.
- This manual is based on Microsoft Excel 2007.

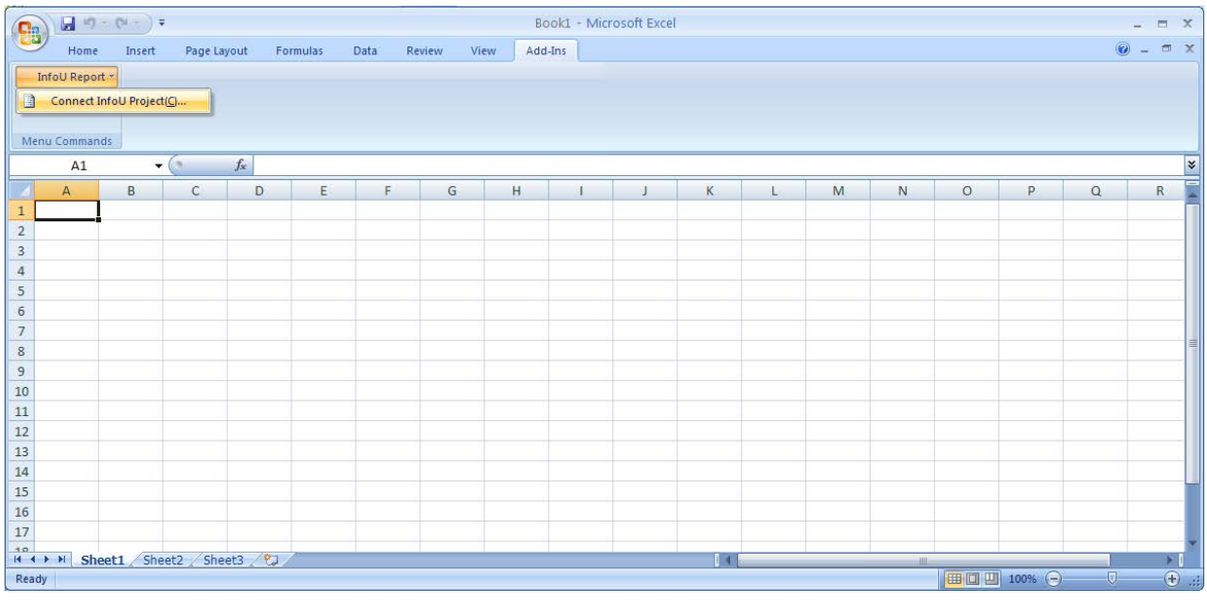
20.1 Start

Select 'Report' from the menu or the InfoU explorer to display the report list.
Double click the list to execute Excel to make a report.



20.1.1 Report Menu

InfoU Report is produced with ActiveX COM and executed on the MS Excel menu and if it is normally installed, the following [Report] menu items are displayed to allow the user to select “Connect InfoU Project”menu.

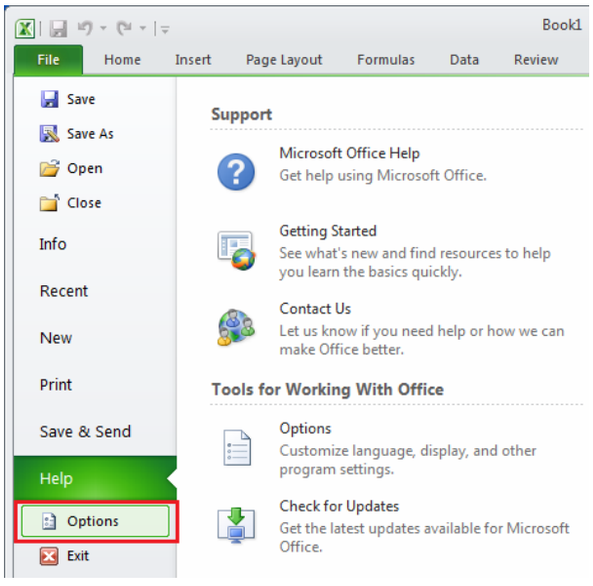


Notice

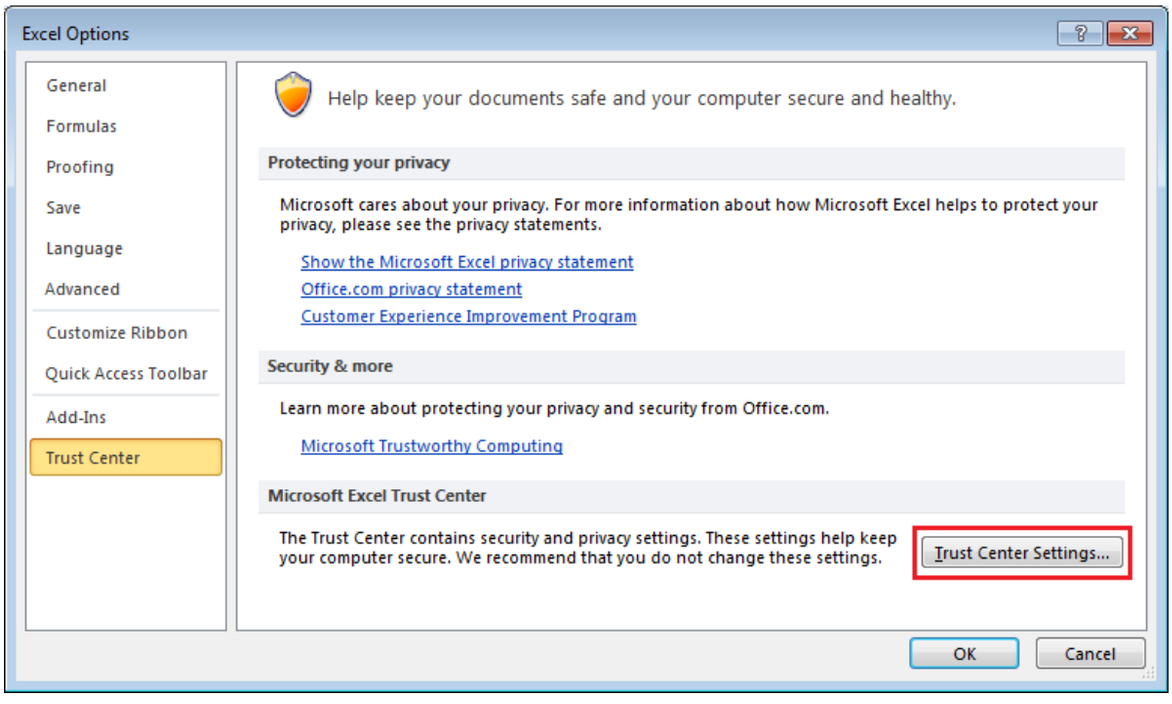
☞ In case [Additional Functions]→[InfoU report] menu is not displayed, take measures base d on the below procedures.

(1) Options of Excel

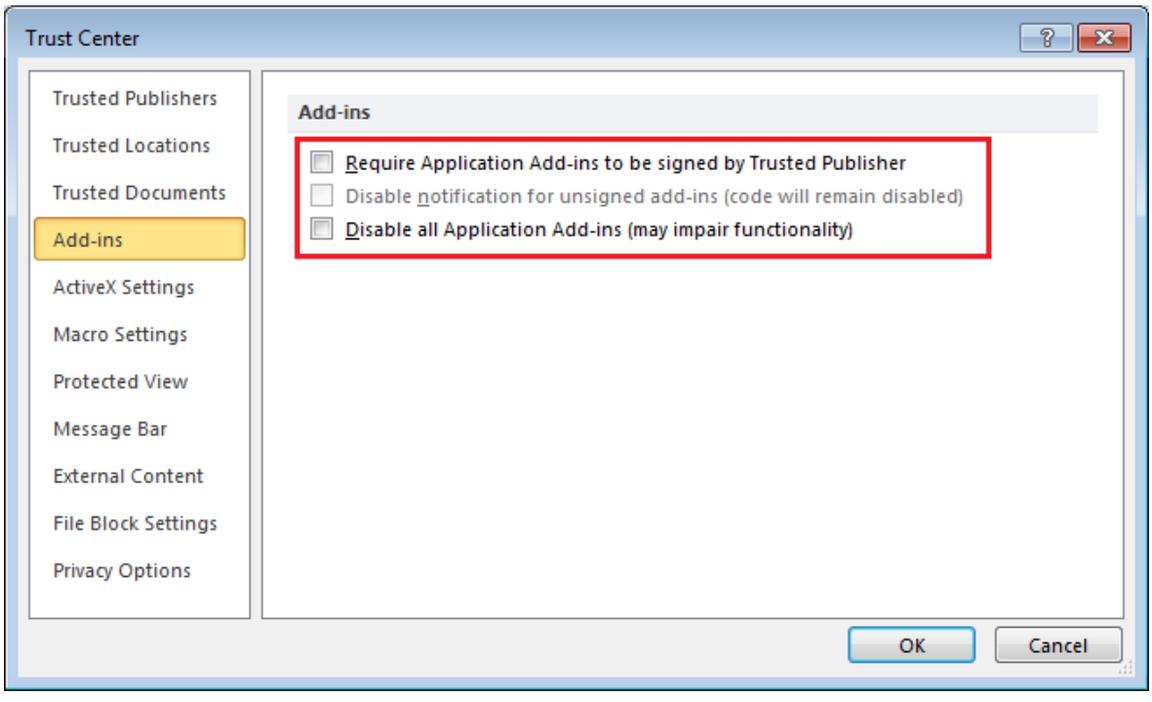
1) Click [File] → [Options]



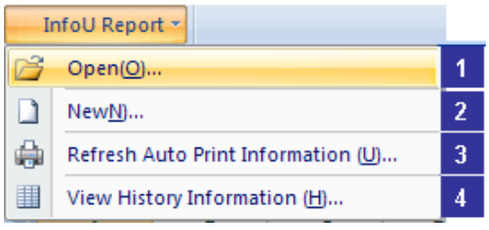
2) Click [Trust Center] → [Trust Center Settings]



3) Click [Add-ins], uncheck the all options

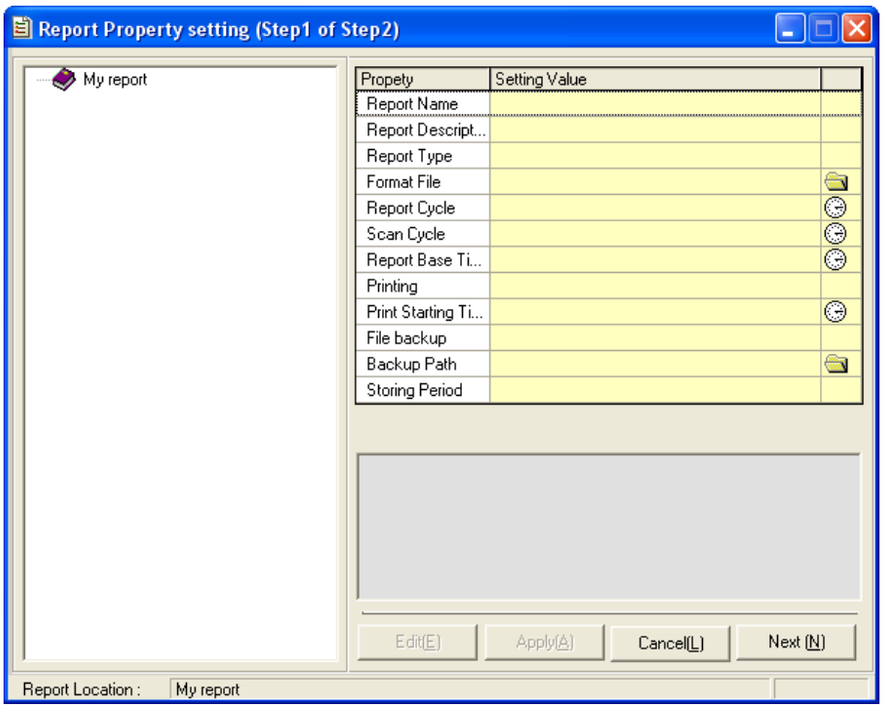


4) The following menu appears upon the selection of “Connect InfoU Project”if InfoU has been installed and it is available to download basic projects.



20.1.2 New

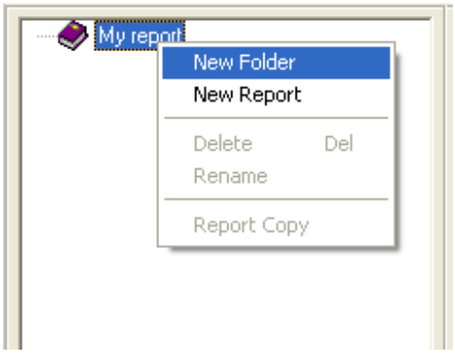
If [New] is selected from the menu, a screen for creating a new report for basic projects or the current project appears.



(1) Menu

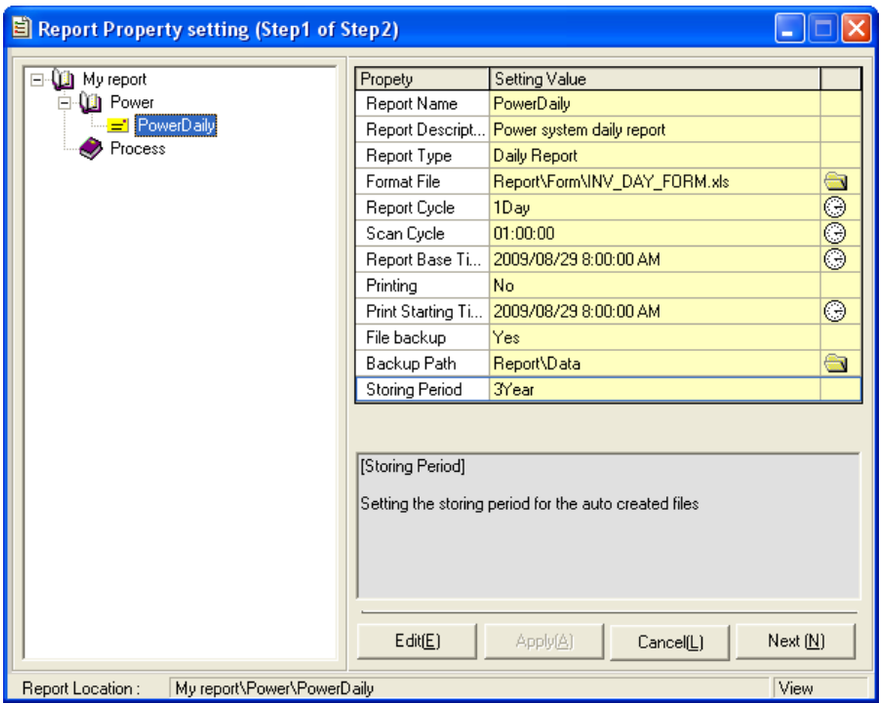
If [New] is selected, the following dialog box appears.

If the highest-level folder is selected and the right side of the mouse is clicked, the following pop-up menu appears.

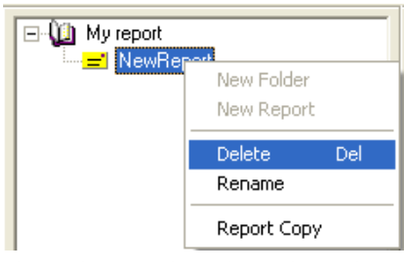


1) New Folder: A new folder is created below the selected item.

2) New Report: The following screen appears to create a new report.

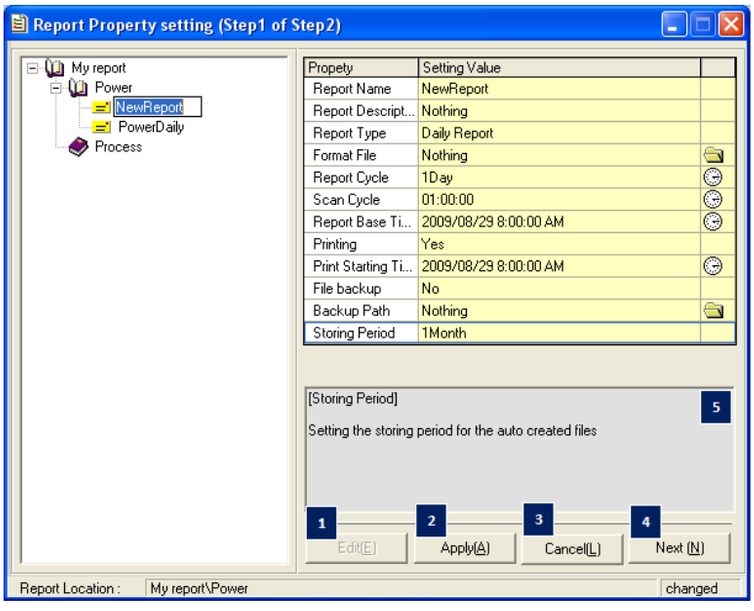


Select the report icon and click the right side of the mouse to display the following pop-up menu.



3) Delete: Delete the selected report.
4) Rename: Change the report name.
5) Report Copy: A report can be copied or moved to other folder.

20.1.3 Report Property Settings

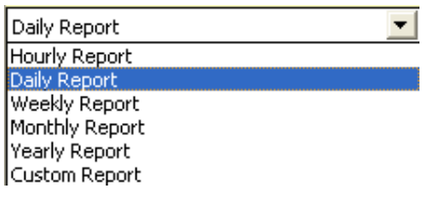


(1) Edit

1) Edit: Click Edit button to change the properties of the report.
2) Apply: The changed properties are saved.
3) Cancel: The edition is cancelled.
4) Next: The properties are saved and the next step [Report Layout] is displayed.
5) Help Window: If the relevant item is selected, instruction and information on its properties are displayed.

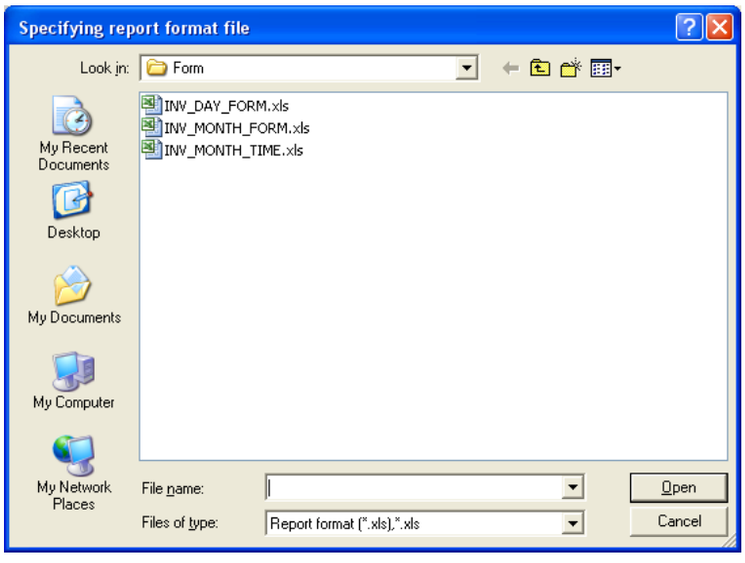
(2) Property

1) Report Name: (16 letters) A name to identify the report. Any two reports do no have the same name in the same folder. It is used to distinguish one from the other.
2) Description: (49 letters) In case that it is obscure to distinguish one from the other only by their names, the user may record details on the report.
3) Type: For regular reports (hourly, daily, weekly, monthly, annually reports), the [Report Interval] property on the bottom is automatically set up while for free reports, the user needs to select an interval. If a value area is selected, a list box showing report types is displayed to allow the user to select one among them.



4) Format File: Reports are made in Excel file format with functions edited by the user. Therefore, Excel file (.xls) or Excel format file (.xlt) shall be designated.
Click one of the right folder icons to display a selection screen. Then, select a file and click Open button to make the relevant file name automatically inserted. If these files are selected in the project folder,

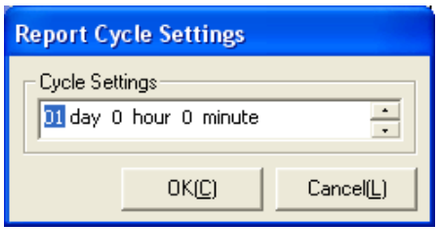
previous project paths are omitted and saved and if other paths except them are selected, the computer name is also automatically added and displayed.



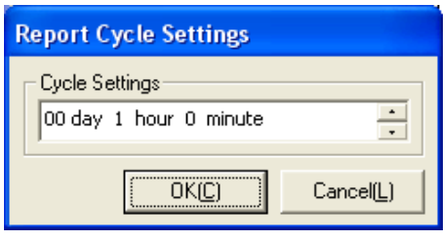
5) Report Cycle: Only Free Report can be set up a cycle to create a report. Data are collected according to the following cycle and once data are collected, a report is created.

Ex.1) Cycle of daily report = 1 day, Scan cycle = If it is set up one hour, one daily report is created once 24 set of one hour data are collected.

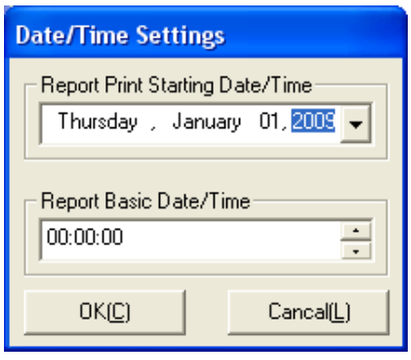
Ex.2) Cycle of daily report = 1 day, Scan cycle = If it is set up 15 minutes, one daily report is created once 96 set of 15-minute data are collected.



6) Scan Cycle: An interval displayed in the report as a time distance to collect data. If the clock mouse on the right side of the relevant item is clicked, the following screen appears to allow the user to select a cycle.

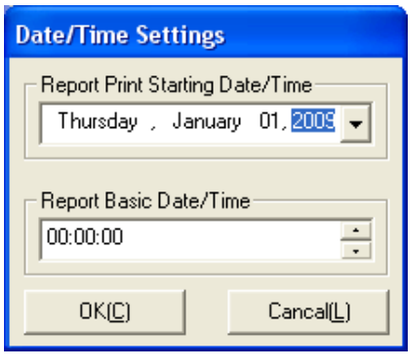


7) Report Base time: A report is written from the set up time on the set up date. Scan Cycle is the same with Report Cycle. That is, a report is created on 'the set up time every report cycle' based on the time to write a report.



8) Printing: It displays whether to execute 'Auto Print.' If 'No' is selected, 'Auto Print' is not executed and the following output time is ignored.

9) Printing Start Time: A report is outputted from the set up time on the set up date. For example, if it is set up as seen in the following figure, output or backup of the relevant report is executed at noon.



10) File Backup: It displays whether to execute 'Report Backup.' Backup is executed at the same time at the same interval with the 'Printer Output' above. If 'No' is selected, backup is not executed while if 'Yes' is selected, the user needs to designated a backup path.

11) Backup Path: Select a backup path on the screen and press ‘Ok’ button. If these files are selected in the project folder, previous project paths are omitted and saved and if other paths except them are selected, the computer name is also automatically added and displayed.

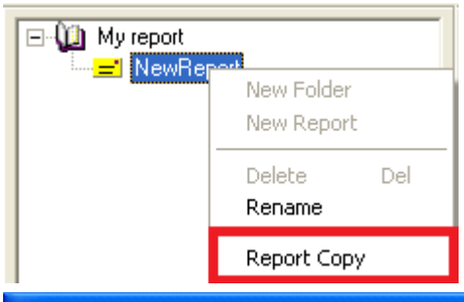


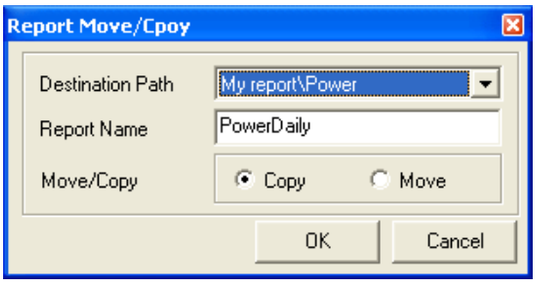
12) Storing Period: The user may set up a period to store backup data. If this period passes, the backup data is automatically deleted.

13) Writer: Record a person who writes the report.

(3) Copy / Move

Select ‘Report Copy’ from the menu below to display the following screen.



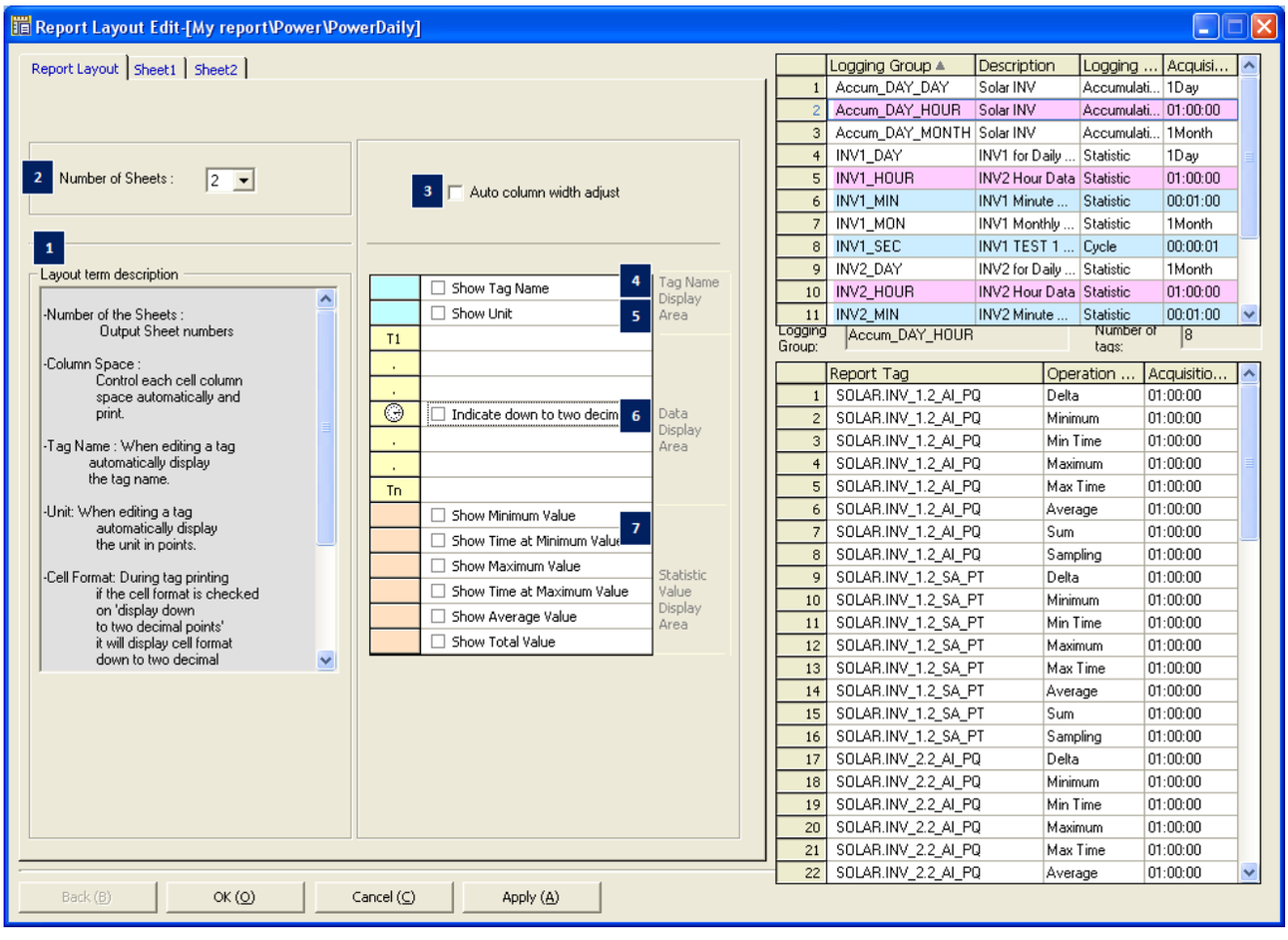


1) Destination Path: Designate a path to which a report is copied or moved.
2) Report Name: Designate a report name to be created after being copied. If the same name already exists, a new report is created in a form of [report name] (1).
3) After modifying the report properties, the user may select [Auto Print Information Renew] from the main menu to apply the modified information online.
4) OK: It performs the settings
5) Cancel: It performs exits the screen without changes.

20.2 Layout

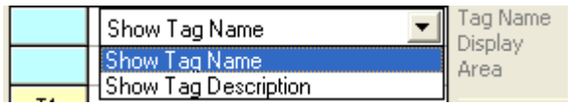
20.2.1 Report Layout

Once 'which tag is outputted on which date/time is defined, the next step is to define which type the outputted report has. Open the Excel file defined as a format file and display the layout setting screen.



(1) Layout term description: Each item on the Report Layout screen is described.
(2) Number of sheet: Set up the Excel sheet number of the relevant report. (Up to 5 sheets are available)
(3) Auto column width adjust: The row distance of each cell is automatically adjusted and displayed.
(4) Show Tag Name/Show Tag Description: During the tag edition, the selected tag name or description is

automatically displayed. To select a tag name or description, press the cell next to the check box.



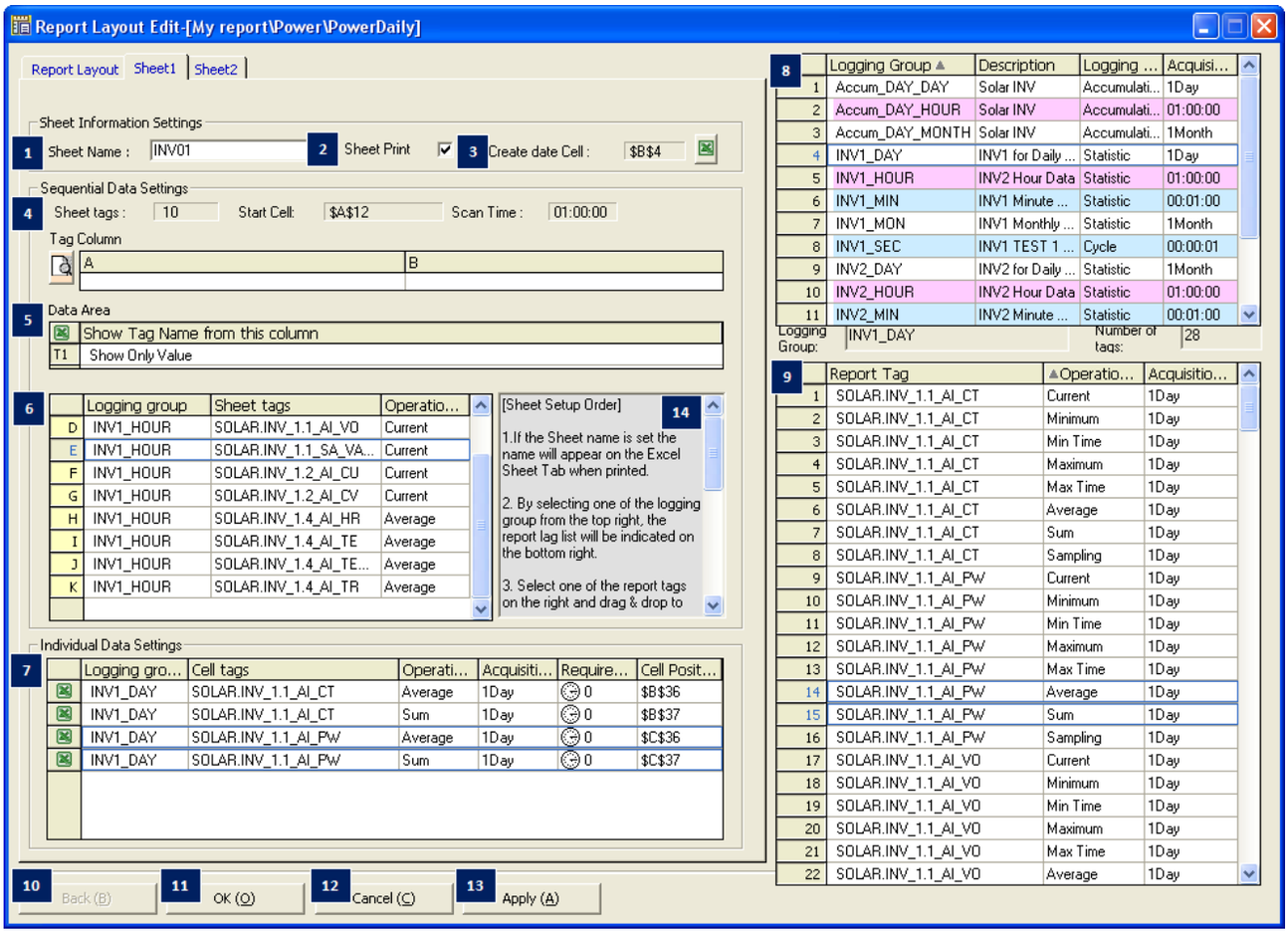
(5) Unit: During the tag edition, properties of the selected tag unit are automatically displayed.

(6) Statistics value: A desired statistics value is automatically computed and displayed. Statistics values are displayed on the report in the same order with the one currently displayed and if any specific statistics value is not selected, they are moved up one row and displayed.

(7) The statistics values computed here will be those of cells and they are the same with the ones for which expressions are used. To obtain more precise data statistics values than the actual ones, it is desirable to designate data statistics values to cell tags and display them.

20.2.2 Sheet Property

Set up a layout for each sheet.

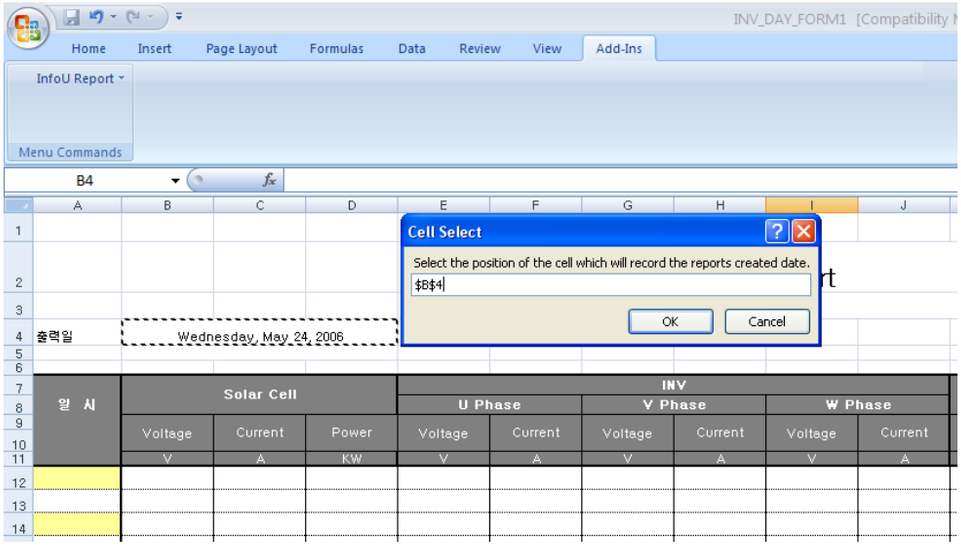


(1) Sheet Name: Designate and save a name of the Excel sheet when saving a report backup.

(2) Sheet Print: Set up whether to output the sheet.

(3) Created Date Cell: Designate a cell to display the date on which the report is created.

Click on  to display the following screen and input cell address by clicking the relevant cell of Excel.



(4) Sequential Data Settings

It is the report having the form that multiple tags are repeated at a regular time interval such as daily report, monthly report, annually report, etc. It has the same property so you can register report tags in the sequential settings.

1) Sheet tag: It displays the number of the currently set sequential data.

2) Start cell location: It displays the location where the sequential data will start. You can specify the location of the start cell by selecting  of the data area.

3) Archiving cycle: It displays the archiving cycle of the logging data that will be output as the sequential data.

4) Tag column preview: If you click , you can preview the currently set layout and check the settings.

(5) Data area

It shows the shape of the start cell where the value is displayed depending on the layout property settings.

1) In case nothing is selected



The value is output and displayed in the start cell.

2) In case 'Show Tag Name'(description) is selected



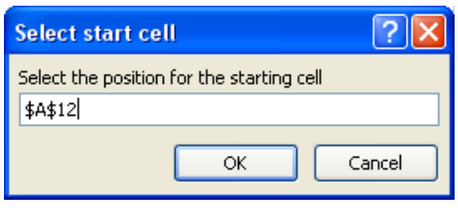
After the tag name is output in the start cell, the value is output and displayed from the next cell (Raw).

3) In case 'Show Tag Name', 'Show Unit' are selected



After the tag name and unit are output in the start cell, the value will be output and displayed from the next cell (Raw).

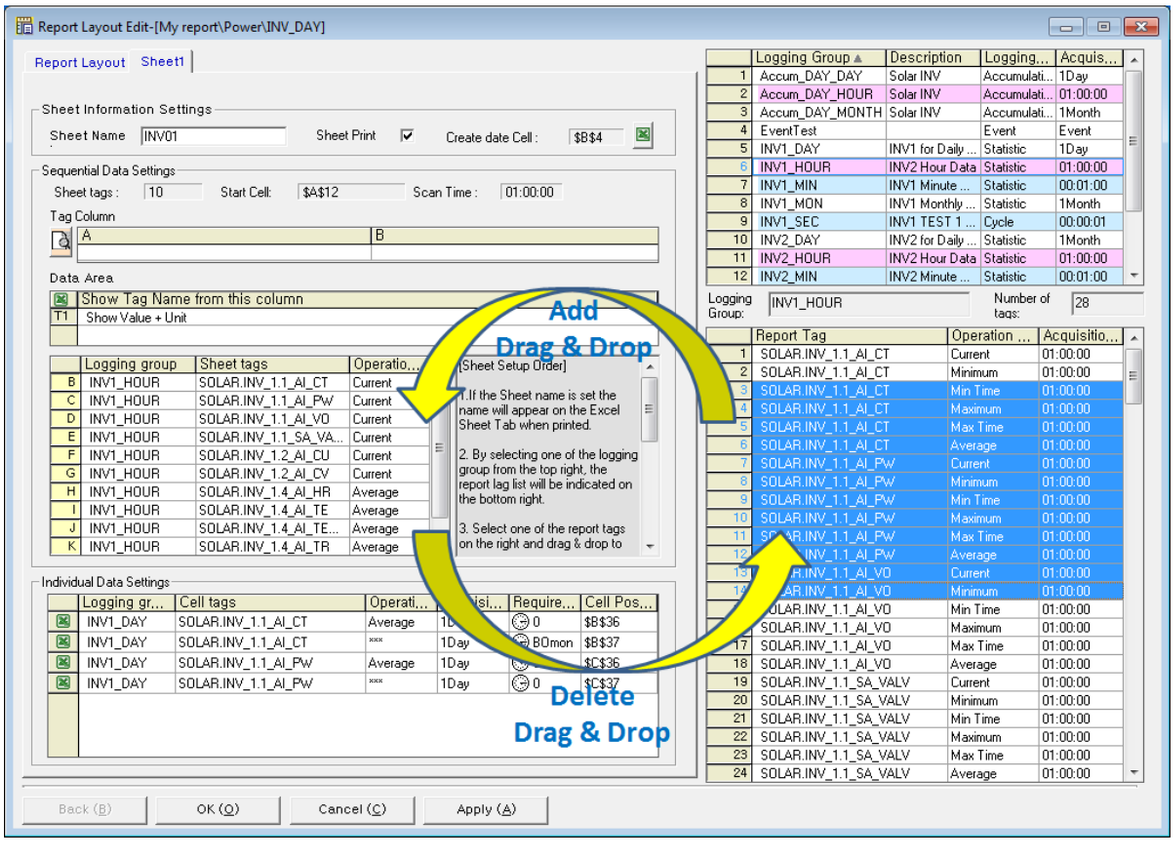
4) If you click , the below message box will show up. When you click the relevant cell of Excel, the cell address will be input and the locaiton where the sequential data starts will be specified.



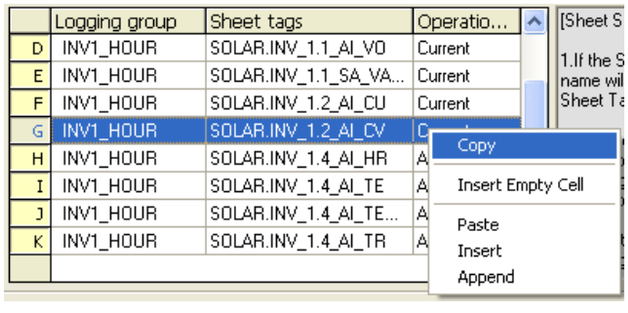
(6) Sequential data tag input:

You can set the tag list to be displayed as the sequential data.

1) If you select the logging collection model in the upper right corner, the tag list and operation types belonging to the relevant collection model will be displayed.
2) After selecting the targeted tag, add it to the list through Drag & Drop. (For deleting the tag, Drag & Drop in the opposite direction)
3) If you click the header of each list, the relevant column will be sorted.



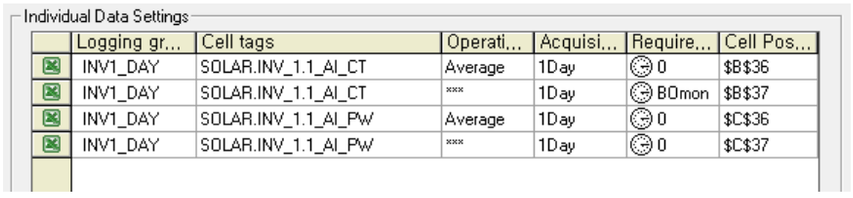
4) Copy/Paste: If you click with the right mouse button, the below menu will show up. After copying the list, you can Edit, Paste, Insert and Append it in excel and clipboard.



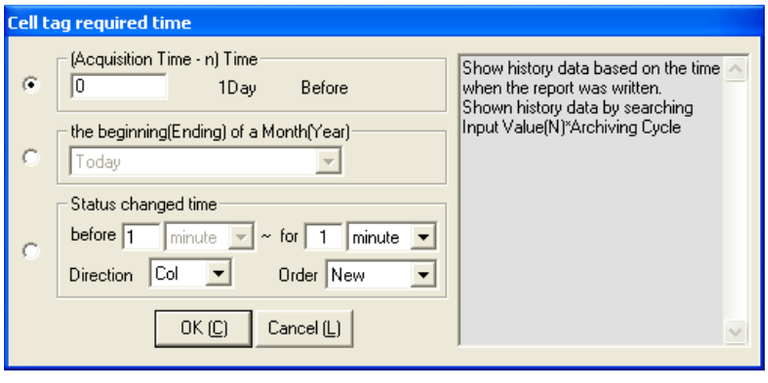
a. Insert Empty Cell: If you want to add the column where the data is not displayed, select 'Insert Empty Cell'. Then, the column is not populated by the data so you can use the arithmetic function of Excel.

(7) Individual Data Settings

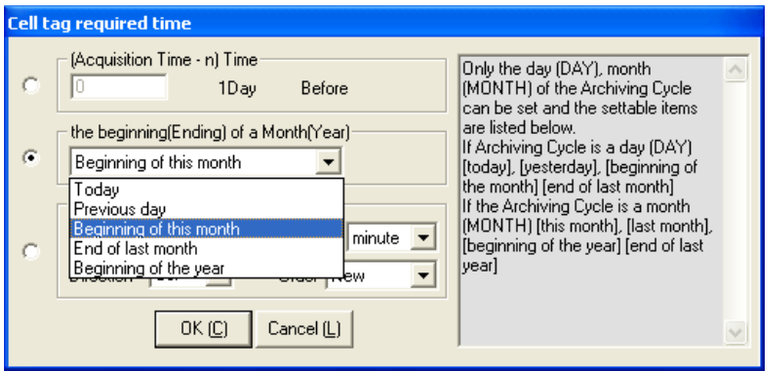
Unlike the Sequential Data Settings, Individual Data Settings can be used when you want to display the report tag in the specific cell of the sheet. Accordingly, after inputting the tag of the logging group, you should set the required time to output the data in the desired time.



1) After selecting the targeted tag among the report tags on the right side, add it to the individual setting list through Drag & Drop (For deleting the tag, Drag & Drop in the opposite direction)
2) If you click the clock icon displayed in the required time column of the tag to which you want to apply individual settings, the below screen will show up.
   For multiple selections, you can apply the same settings to the chosen cell tag.

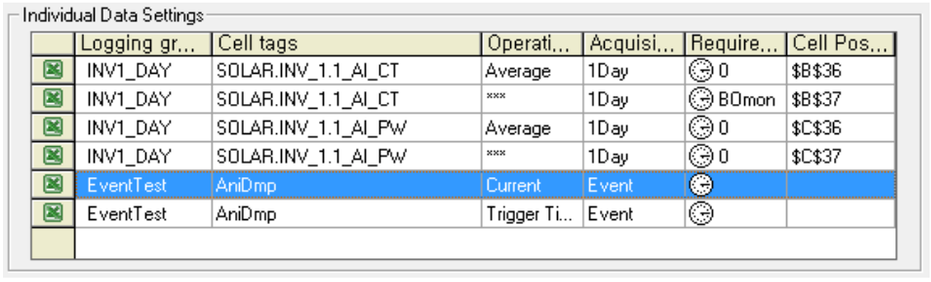


3) The above screen describes the method how to bring the data from the point in the past based on the acquisition time of the report of the individual settings. It reads the previous data for the time of N* archiving cycle. If you enter “0” into the acquisition time, the data will be queried and output. Namely, if “00h 00m 00s” is the output time of the report, the data of the cell will be 00h 00m 00s. In the case of “12”, the data of 12 hours earlier than “00h 00m 00s” will be queried and output.
4) Setting the Beginning (end) of the month/Beginning (end) of the year
   The below screen is available only when the archiving cycle is a DAY or a MONTH. The configurable items are as below.



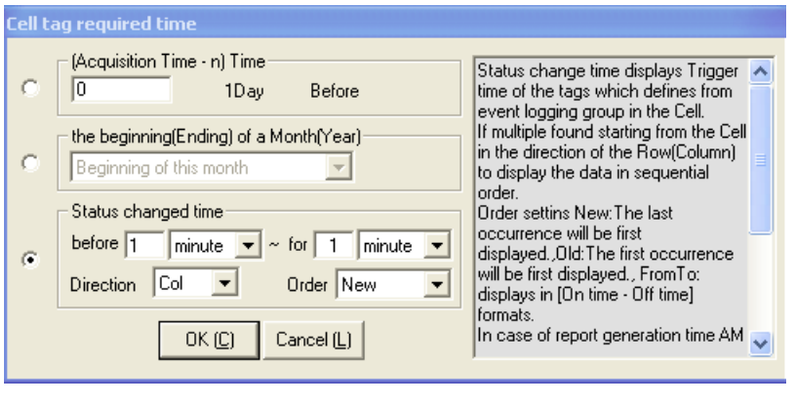
a. If the archiving cycle is a DAY: "today", "yesterday", "beginning of the month", "end of the last month"
b. If the archiving cycle is a MONTH: "this month", "last month", "beginning of the year", "end of last year"
   When setting as Today (Month), the same result as the set value “0” of the above acquisition time will be output.

5) Event logging output settings

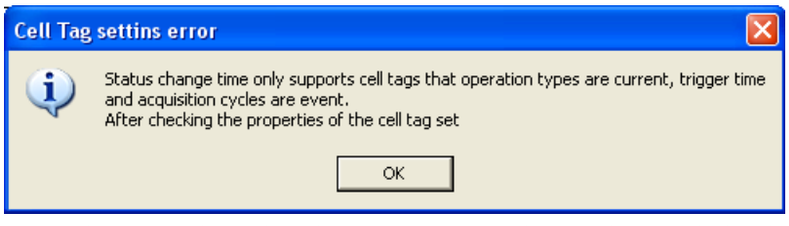


a. In the case of the event logging, the number of data output is not fixed. The trigger time and current value can be output.

If you click the clock icon displayed in the required time column of the tag to which you want to apply individual settings, the below screen will show up.



b. If you select the tag type other than the event logging, the below message will be displayed.



c. For the event time, the trigger occurrence time of the tag defined from the event logging group is displayed in the cell.

d. When multiple tags are searched, starting from the specified cell position, in the direction of Col (horizontal) or Row (vertical), the data is displayed in the sequential order.

e. Order settings New: The latest occurrence is first displayed, Old: The first occurrence will be first displayed, FromTo: It is displayed in [On time – Off time] format.

f. In case the acquisition time of the report is 00, when you select [1 day before], [for 1 day], the event occurrence data between 00:00 of the last day and 00:00 of today will be displayed. This condition is proper for the daily report.

g. In case the acquisition time of the report is 00, when you select [3 hours before], [for 1 hour], the event occurrence data between 20:00 of the last day and 21:00 of today will be displayed.

h. The below figure shows the example of outputting the event logging report.

Energy	Integrating watt-hour	Motor running time	Status
KWH	KWH		
1,128	46	23:20:00	1
858	2	23:20:00	1
43	56	23:20:00	1
1,047	43	23:20:00	1
42	75	23:20:00	1
1,000	17	23:20:00	1
16	32	23:20:00	1
61	47	23:20:00	1
4	94	23:20:00	1
980	94	23:20:00	1
1,020	25	23:20:00	1
44	2	23:20:00	1
1,009	75		
12	45		
1,140	23		

6) Copy/Paste: If you click with the right mouse button, the below menu will show up. After copying the list, you can Edit, Paste, Insert, Append it in Excel.

7) When inputting the cell position, there is inconvenience of clicking  one by one for setting so it can be used when you input the cell tag having certain rules. For example, when the position value of the selected cell tag is B30", after selecting 5 ones, if you select [Sequential input of row(vertical)], B31" ~ "B34" will be automatically input in regular order; if you select [Sequential input of column(horizontal)], B31" ~ "B34" will be automatically input in regular order.

(8) History Data Collection Model

A list of the data collection models registered in the history data collection models.

1) Red Color: Collection models corresponding to the report data creation cycle. That is, they are data collection models that have one -hour collection cycle in case of daily reports.
2) Blue Color: Collection models that can create reports even though they are not corresponding. That is, they are data collection models that have collection cycle less than one hour and multiples of integers (one minute, 10 minutes, 30 minutes, etc).
3) White Color: Collection models that can not be applied to continual tag data areas.
4) If a certain collection model is selected, a list of tags that belong to the model is displayed at the bottom and the data kinds available to select vary according to the properties of the collection model.

Logging Group	Analog	Digital	Remarks
Cycle	Current value	Current status	Minimum collection 1second Mainly used for the history trend at fast time intervals(less than 1minute)
Statistics	Current value, Min value,	On count, Running time,	Minimum collection 1 minute Min value,

	Max value, Min time, Max time, Average value, Total value, Sampling count	Change count	Max value, Min time, Max time, Operation cycle 1 minute Mainly used for report of trend more than time unit
Accumulation	Change value, Sampling count	Not supported	Collection cycle: 1hour, 1day, 1month When tag data has incremental properties such as accumulation electric power meter or flow meter, data fluctuation is saved for each time unit.
Event	Trigger Time, Current Value	Trigger Time, Current Value	Display the trigger time and the current value of that time.

When an item is selected, the tag list to which the relevant collection model belongs and its data kind is displayed in the bottom list. The continual tags are limited to the available tags to be set up according to the report preparation rules but, individual data cells can be entered with any tag regardless of cycle or model kind.

(9) Tag List

As a tag list of the collection models selected above, it can be set up by dragging and dropped to the continual data or individual data table.

(10) Back

The screen moves back to the property edition screen after edition is stopped.

(11) Ok

The settings are saved and the screen is closed.

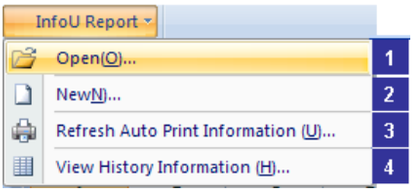
(12) Cancel

The edition work is stopped and the screen is closed.

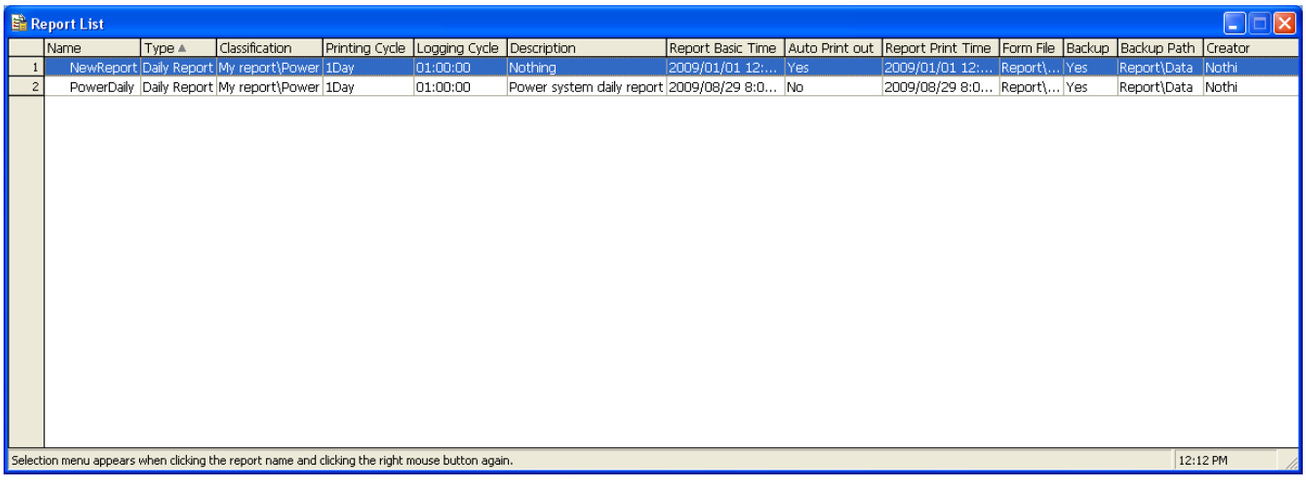
(13) Apply

Settings are saved.

20.3 Report Open



If [Open] is selected from the report main menu. The report list screen is displayed as seen in the figure below. All of the report information defined is displayed on this screen.



The following actions can be applied to the list.

● Click on the header of each column to rearrange the column contents in the ascending or descending order. (The arrangement order is displayed while the direction of the arrow displayed on the column header is changed)

● The column is rearranged if a certain column is clicked and held while moving the clicked column before other column.

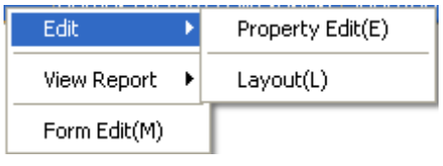
● Double click on the border of the column on the right side to make the width of the column automatically adjusted.

● If some contents are not visible on a cell, bring the mouse on the cell to view all of the hidden part.

● Select an item and click the right side of the mouse to pop up available menus.

20.3.1 Edit

Two steps of [New] are editable individually. That is, operations such as property edition and layout are available individually.

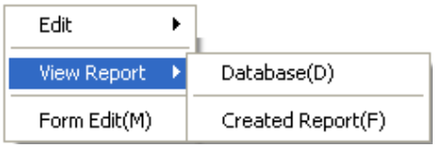


(1) Property Edit: The report property edition screen is displayed to allow the user to set up properties.

(2) Layout: The report layout edition screen is displayed to allow the user to set up properties.

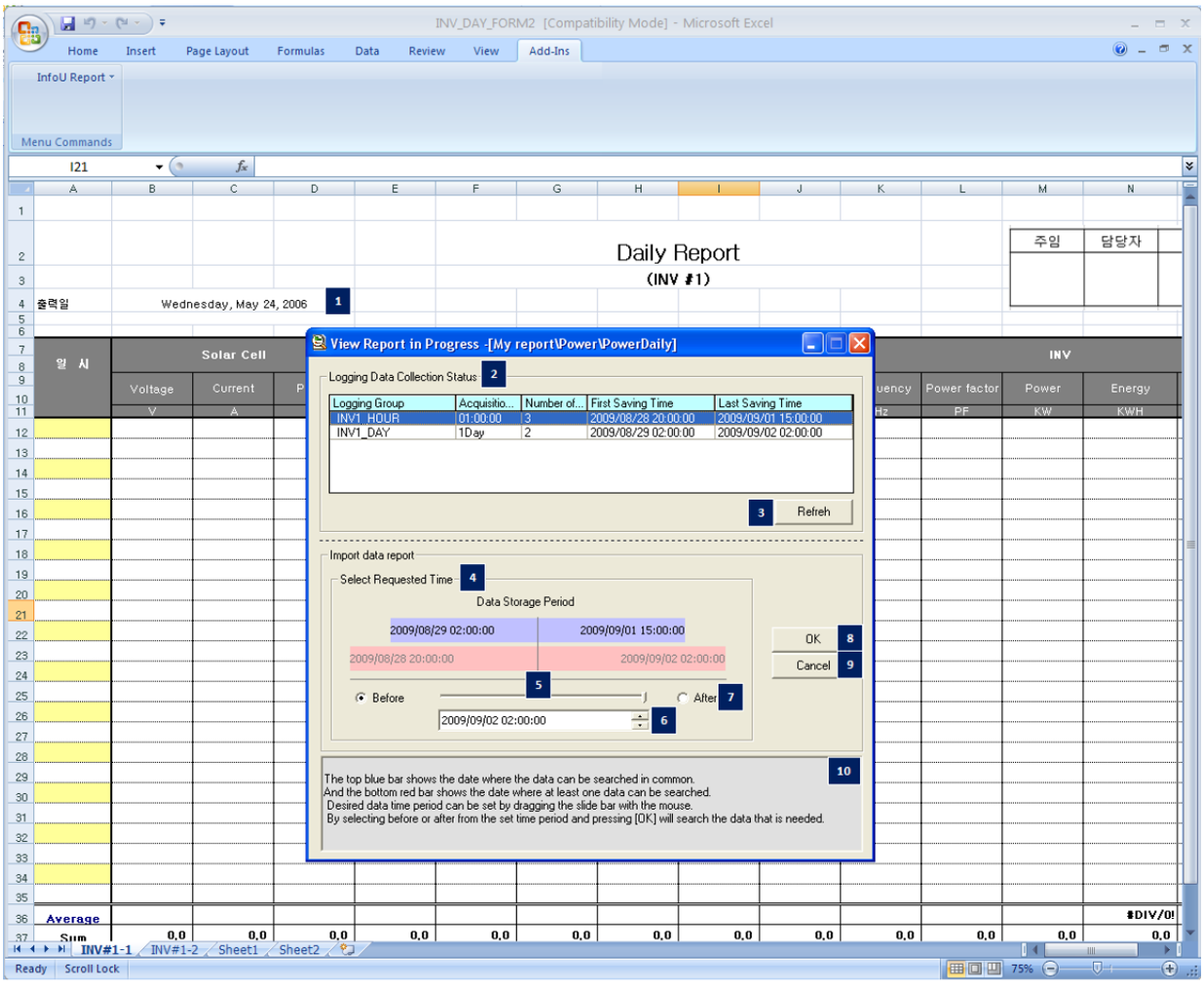
20.3.2 Report View

This function ensures not only the newly saved or created reports are properly searched but also the data retained by the history data collection model are searched during a certain period the user wants.



(1) Database

Reports are automatically printed or saved as backup files periodically. This section explains a function to view reports in the time zone the user wants on the screen unlike printer outputs. This function can run only in the runtime mode.



1) Format File

A setup Excel format file is opened and the [Progressing Report View] screen is displayed.

2) Report Data Collection status

   The data saving status of the history data collection models set up in the current report is displayed in a form of list and the list can be referred by the user when he/she searches data and each column is described as follows:

   a. Logging Group: The name of the data Logging Group.

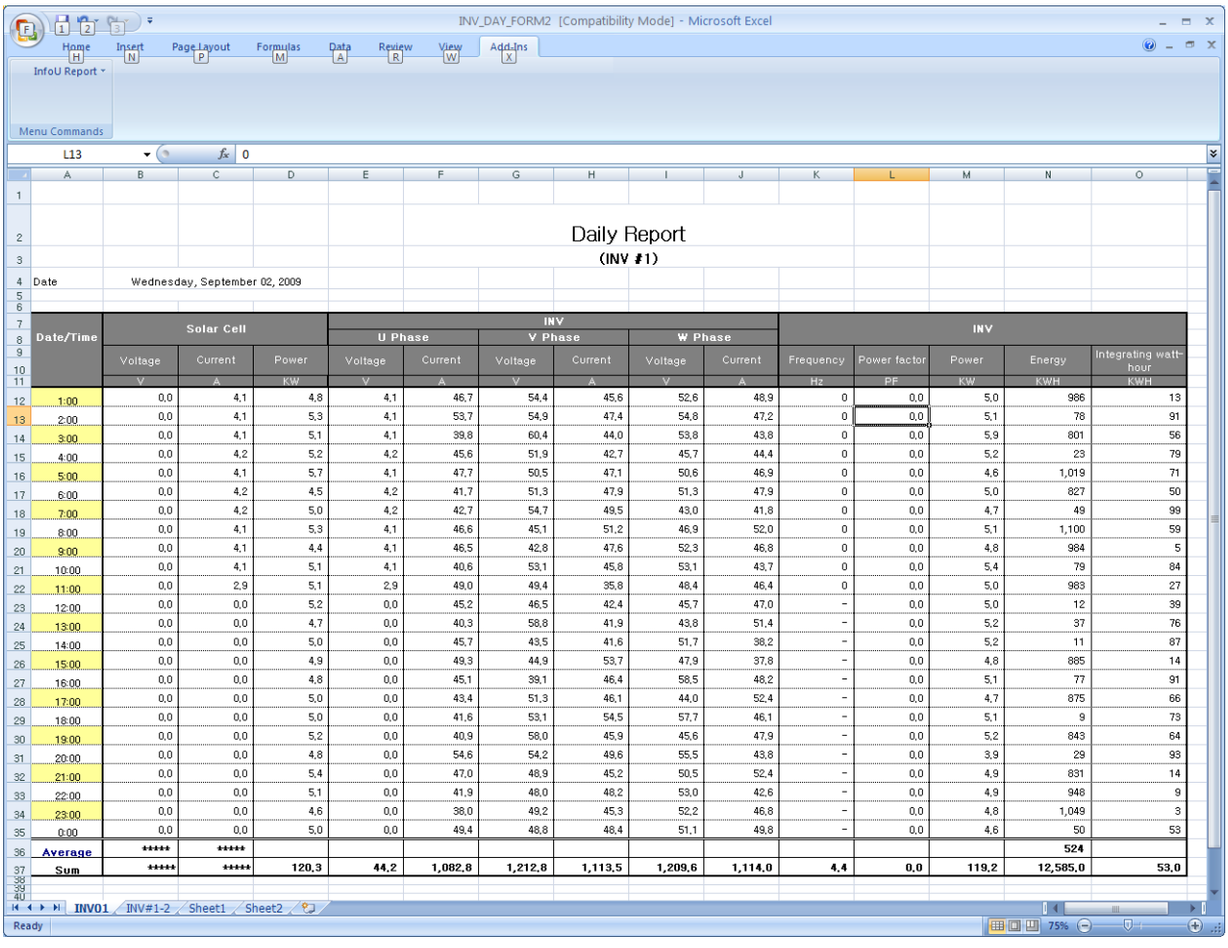
   b. Acquisition Cycle: Data collection cycle of the relevant collection model.

   c. Number of records: The number of the stored data until now.

   d. First Saving time: The oldest data time

   e. Last Saving Time: The latest data time

3) Refresh: The list is renewed with the latest information.
4) Select Requested Time: Select a time period to inquire.
5) Data Storage Period: The blue upper bar shows the period during which all of the data can be displayed while the red bottom bar shows the period during which at least one data can be displayed.
6) Reference Date Designate: Set up data in the desired time zone by pressing and holding the slider knob of [Demand Time Select] and moving the mouse.
7) Reference Time: Enter and set up a reference time when the user wants to inquire.
8) Before/After: Previous or next data to the reference time can be inquired if either Previous or Next is selected.
9) OK: Data in the selected time zone are outputted (report cycle/collection cycle).
10) Cancel: The inquiry is stopped and the screen is closed.
11) Help: The user can be received on the information on how to inquire database reports.
12) The data inquiry results are as follows:



INV_DAY_FORM2 [Compatibility Mode] - Microsoft Excel

Daily Report

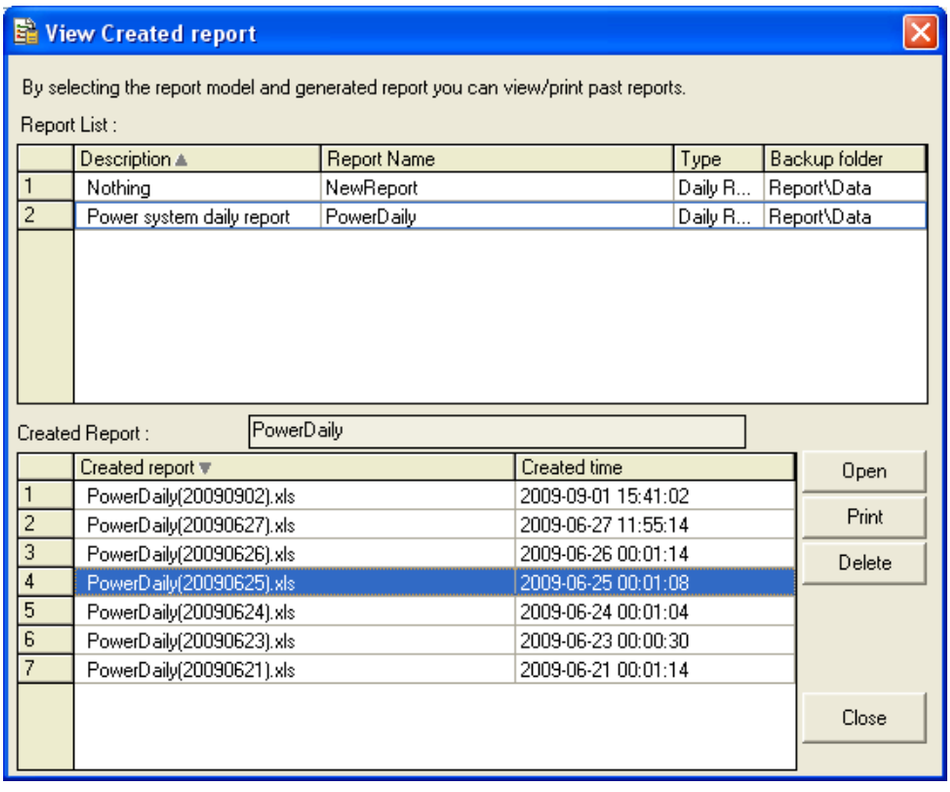
(INV #1)

Date Wednesday, September 02, 2009

Date/Time	Solar Cell			INV U Phase		INV V Phase		INV W Phase		INV				
	Voltage	Current	Power	Voltage	Current	Voltage	Current	Voltage	Current	Frequency	Power factor	Power	Energy	Integrating watt-hour
	V	A	KW	V	A	V	A	V	A	Hz	PF	KW	KWH	KWH
1:00	0,0	4,1	4,8	4,1	46,7	54,4	45,6	52,6	48,9	0	0,0	5,0	986	13
2:00	0,0	4,1	5,3	4,1	53,7	54,9	47,4	54,8	47,2	0	0,0	5,1	78	91
3:00	0,0	4,1	5,1	4,1	39,8	60,4	44,0	53,8	43,8	0	0,0	5,9	801	56
4:00	0,0	4,2	5,2	4,2	45,6	51,9	42,7	45,7	44,4	0	0,0	5,2	23	79
5:00	0,0	4,1	5,7	4,1	47,7	50,5	47,1	50,6	46,9	0	0,0	4,6	1,019	71
6:00	0,0	4,2	4,5	4,2	41,7	51,3	47,9	51,3	47,9	0	0,0	5,0	827	50
7:00	0,0	4,2	5,0	4,2	42,7	54,7	49,5	43,0	41,8	0	0,0	4,7	49	99
8:00	0,0	4,1	5,3	4,1	46,6	45,1	51,2	46,9	52,0	0	0,0	5,1	1,100	59
9:00	0,0	4,1	4,4	4,1	46,5	42,8	47,6	52,3	46,8	0	0,0	4,8	984	5
10:00	0,0	4,1	5,1	4,1	40,6	53,1	45,8	53,1	43,7	0	0,0	5,4	79	84
11:00	0,0	2,9	5,1	2,9	49,0	49,4	35,8	48,4	46,4	0	0,0	5,0	983	27
12:00	0,0	0,0	5,2	0,0	45,2	46,5	42,4	45,7	47,0	-	0,0	5,0	12	39
13:00	0,0	0,0	4,7	0,0	40,3	58,8	41,9	43,8	51,4	-	0,0	5,2	37	76
14:00	0,0	0,0	5,0	0,0	45,7	43,5	41,6	51,7	38,2	-	0,0	5,2	11	87
15:00	0,0	0,0	4,9	0,0	49,3	44,9	53,7	47,9	37,8	-	0,0	4,8	885	14
16:00	0,0	0,0	4,8	0,0	45,1	39,1	46,4	58,5	48,2	-	0,0	5,1	77	91
17:00	0,0	0,0	5,0	0,0	43,4	51,3	46,1	44,0	52,4	-	0,0	4,7	875	66
18:00	0,0	0,0	5,0	0,0	41,6	53,1	54,5	57,7	46,1	-	0,0	5,1	9	73
19:00	0,0	0,0	5,2	0,0	40,9	58,0	45,9	45,6	47,9	-	0,0	5,2	843	64
20:00	0,0	0,0	4,8	0,0	54,6	54,2	49,6	55,5	43,8	-	0,0	3,9	29	93
21:00	0,0	0,0	5,4	0,0	47,0	48,9	45,2	50,5	52,4	-	0,0	4,9	831	14
22:00	0,0	0,0	5,1	0,0	41,9	48,0	48,2	53,0	42,6	-	0,0	4,9	948	9
23:00	0,0	0,0	4,6	0,0	38,0	49,2	45,3	52,2	46,8	-	0,0	4,8	1,049	3
0:00	0,0	0,0	5,0	0,0	49,4	48,8	48,4	51,1	49,8	-	0,0	4,6	50	53
Average	*****	*****											524	
Sum	*****	*****	120,3	44,2	1,082,8	1,212,8	1,113,5	1,209,6	1,114,0	4,4	0,0	119,2	12,585,0	53,0

(2) Created Report

Periodic backup report files are inquired. Click on the menu to display the following screen.



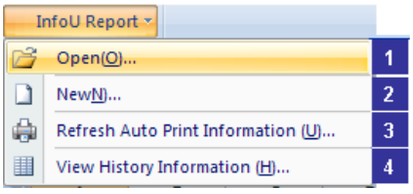
At first, the created report list of the selected report models from the previous report list is displayed at the bottom [2],

1) Report List: The report list defined in the current project is displayed.
   a. Click on the list header to arrange the created reports in the order and each column is described as follows:
   b. Type: Types of the report (Daily report, monthly report, free report, etc….)
   c. Report Name: The report name including its folder is displayed.
   d. Backup Folder: As the path in which report backup files are saved, the project path displays the corresponding path and any path other than the project path displays the entire paths including computer name.

2) Created Report: If the above report model item is selected, the report file list in which the relevant backup path is created appears.

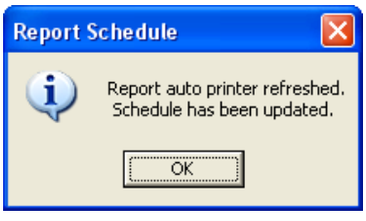
3) Open: The report file selected from the bottom list is opened. The same actions are triggered by double click.

20.3.3 Refresh Auto Output Information

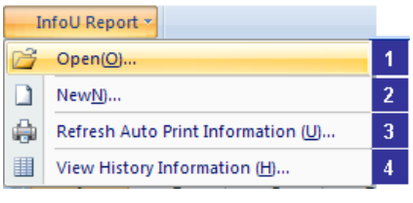


If [Refresh Auto Output Information] is clicked from the Report Main Menu after the report properties are modified, the modified contents are applied online.

This function is executed only in the runtime mode and it renews the report operation list by sending the Renewal Command to the report scheduler responsible for auto output and backup of the report after the report completes its modification process.

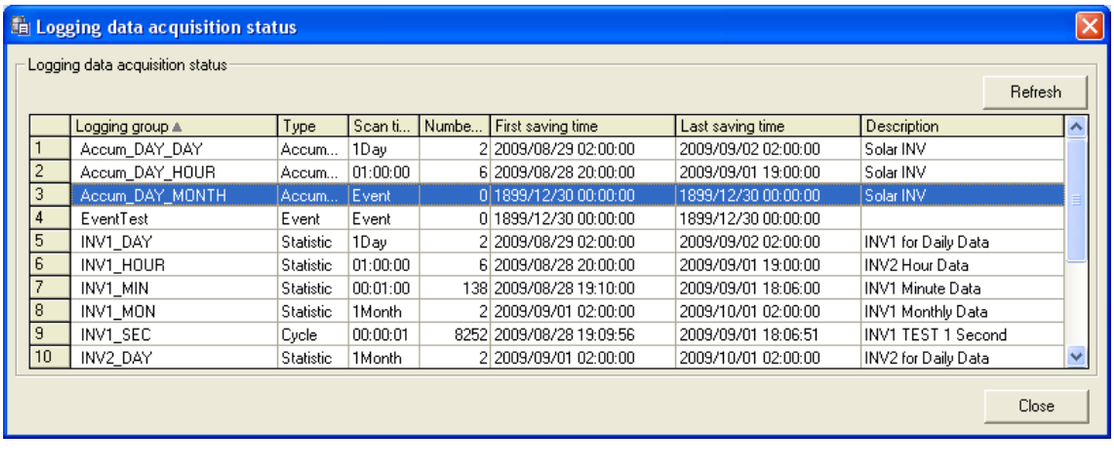


20.3.4 View History Information



If ④ [History Information View] is selected from the report main menu after Report Properties are edited, the following screen appears.

If it is executed only in the runtime mode and the data save status of the history data collection models can be inquired and data can be renewed if [Refresh] button is clicked.



	Logging group	Type	Scan ti...	Numbe...	First saving time	Last saving time	Description
1	Accum_DAY_DAY	Accum...	1Day	2	2009/08/29 02:00:00	2009/09/02 02:00:00	Solar INV
2	Accum_DAY_HOUR	Accum...	01:00:00	6	2009/08/28 20:00:00	2009/09/01 19:00:00	Solar INV
3	Accum_DAY_MONTH	Accum...	Event	0	1899/12/30 00:00:00	1899/12/30 00:00:00	Solar INV
4	EventTest	Event	Event	0	1899/12/30 00:00:00	1899/12/30 00:00:00	
5	INV1_DAY	Statistic	1Day	2	2009/08/29 02:00:00	2009/09/02 02:00:00	INV1 for Daily Data
6	INV1_HOUR	Statistic	01:00:00	6	2009/08/28 20:00:00	2009/09/01 19:00:00	INV2 Hour Data
7	INV1_MIN	Statistic	00:01:00	138	2009/08/28 19:10:00	2009/09/01 18:06:00	INV1 Minute Data
8	INV1_MON	Statistic	1Month	2	2009/09/01 02:00:00	2009/10/01 02:00:00	INV1 Monthly Data
9	INV1_SEC	Cycle	00:00:01	8252	2009/08/28 19:09:56	2009/09/01 18:06:51	INV1 TEST 1 Second
10	INV2_DAY	Statistic	1Month	2	2009/09/01 02:00:00	2009/10/01 02:00:00	INV2 for Daily Data

20.3.5 How to Use during the Runtime

During the runtime, a report can be defined on the plant screen and [Created File] report view is called. Refer to ‘Help’ of ‘Graphic Script.’

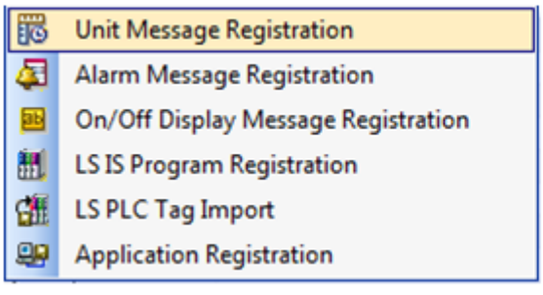
Chapter 21 Utility

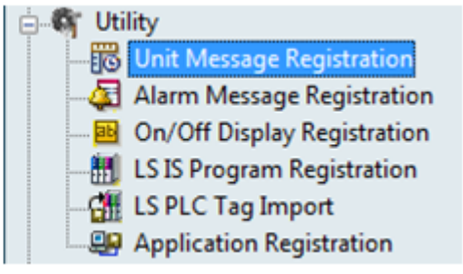
21.1 Unit

A function to set up unit for tag analog values such as length, width, current or voltage

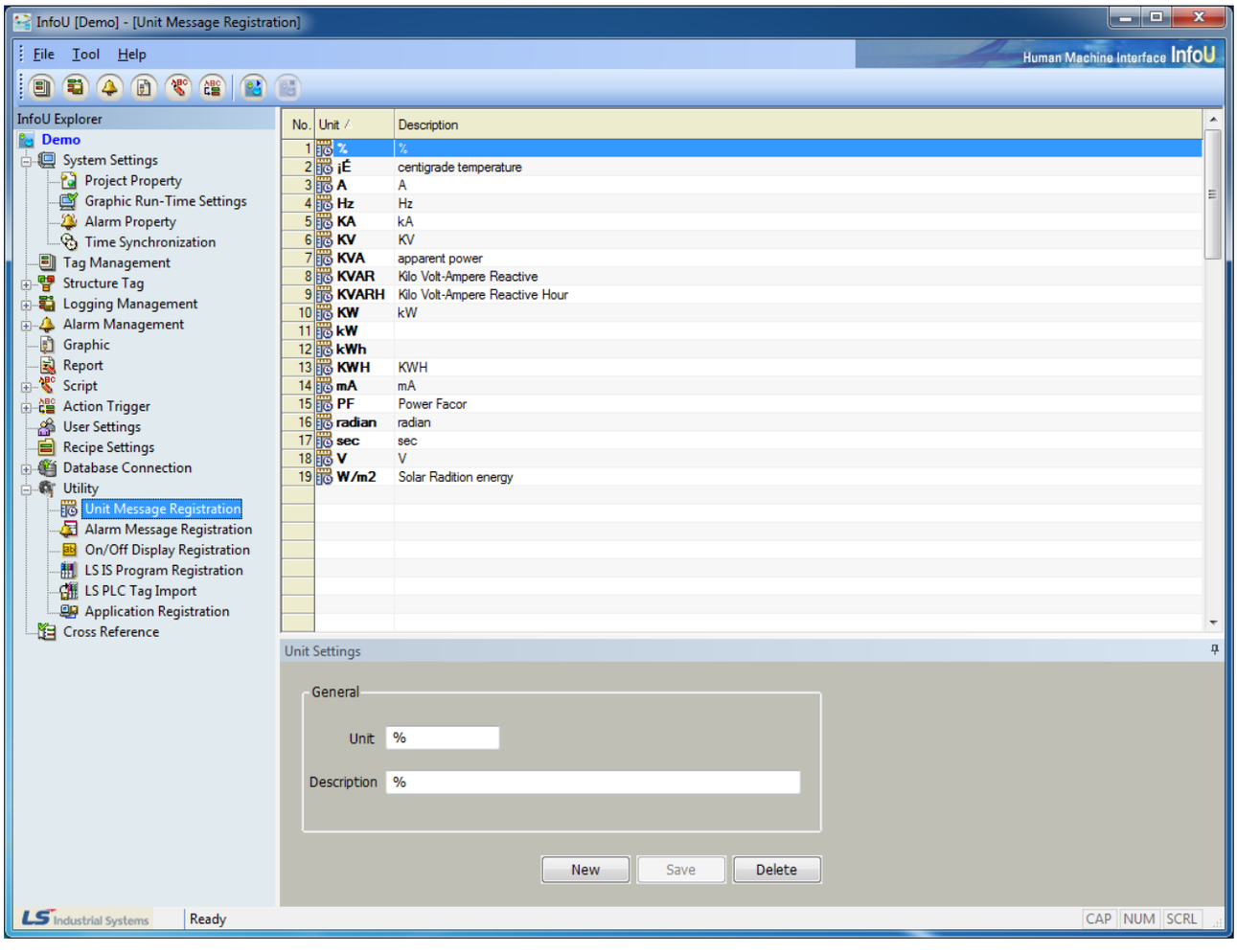
(1) Start

Select‘Unit Message Registration’in the menu or the InfoU explorer.





A list of the registered units is displayed as seen in the figure below and the setting screen is shown on the bottom.



(2) Settings

1) Unit

●Input a unit. If any existing unit is selected and changed, the changed unit is automatically added as a new unit.

●Up to 63 Korean letters and 126 Alphabet letters can be inputted and this item is one of essential information to be inputted.

2) Description

●Up to 127 Korean letters and 254 Alphabet letters can be inputted for this item and this item is allowed to be omitted.

●Briefly explain on the inputted unit.

3) New

●If this button is selected, [Unit] and [Descirption] are initialized to allow the user to input a new unit.

4) Save

●If [Save] button is selected, the inputted unit is saved and it is shown on the notice window.

5) Delete

●Select [Delete] button to delete the selected unit display it on the notice window.

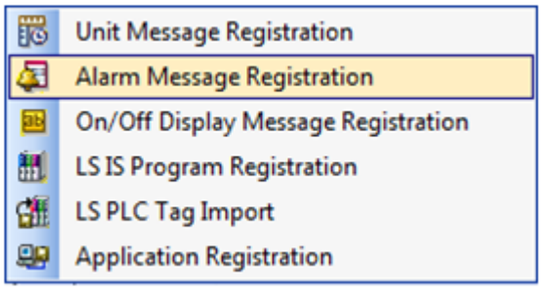
●It is also available to select multiple items in the list and delete them at once.

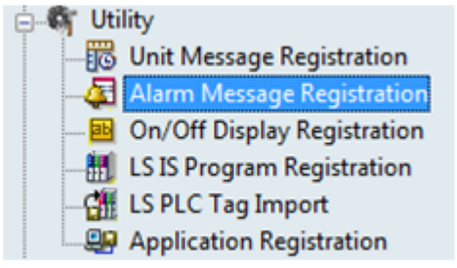
21.2 Alarm Message Registration

A function to register a string to be displayed in case of the occurrence or recovery of alarm

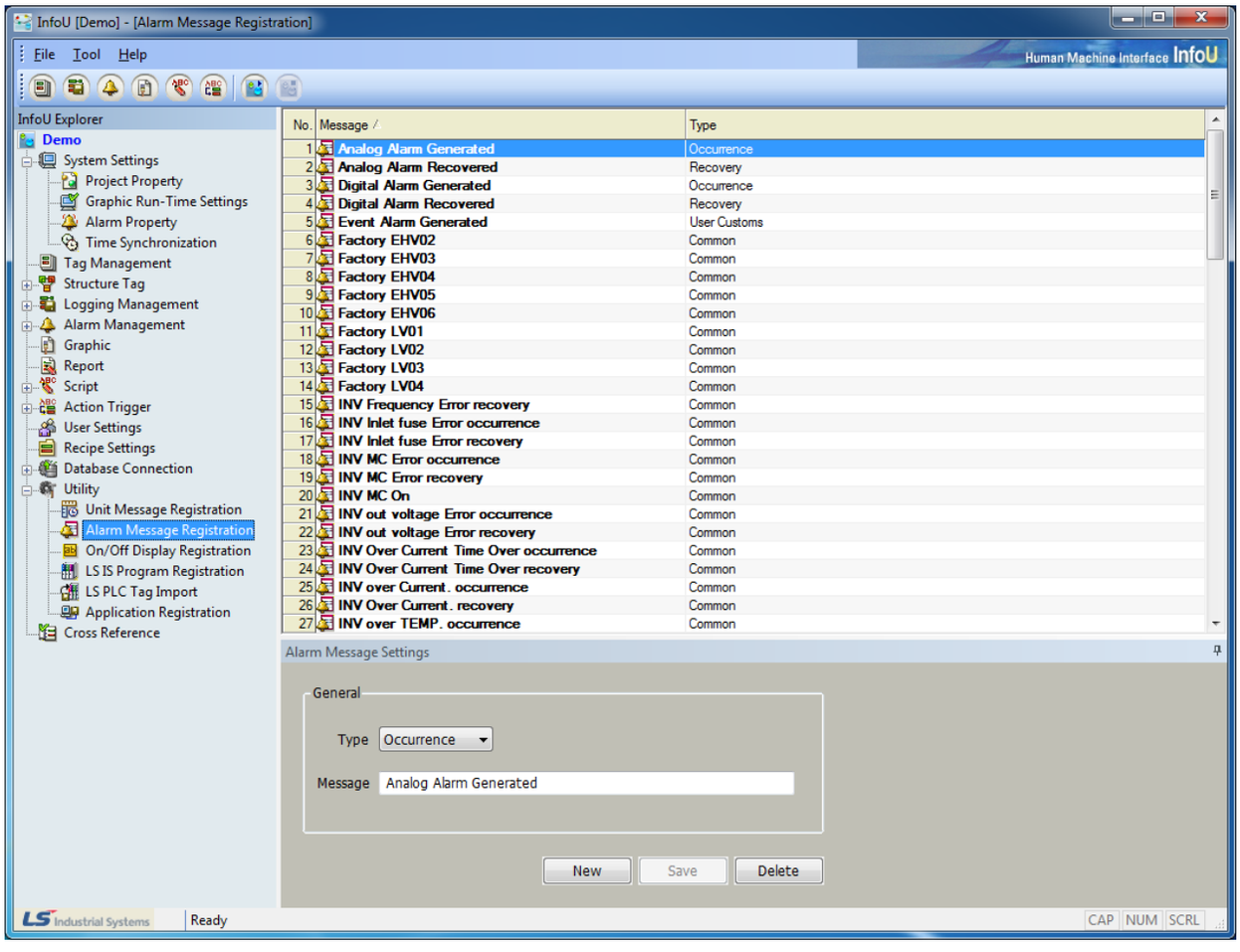
(1) Start

Select ‘Alarm Message Registration’ in the menu or the InfoU explorer.





A list of the registered alarm messages is displayed as seen in the figure below and the setting screen is shown on the bottom.



(2) Settings

1) Type

●Select a type of the occurred alarm.

●Four types such as [Common], [Occurrence], [Cancellation] and [User Customs] are available and this information is mandatory.

2) Message

●Input a string to be displayed in case of the occurrence of alarm.

●Up to 63 Korean letters and 126 Alphabet letters can be inputted and it is one of essential information to be inputted.

3) New

●The existing [Message] is initialized to allow the user to input a new message.

4) Save

●If [Save] button is selected, the inputted message is saved and it is shown on the notice window.

5) Delete

●Select [Delete] button to delete the selected item display it on the notice window.

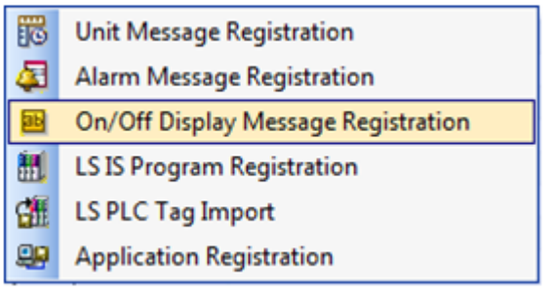
●It is also available to select multiple items in the list and delete them at once.

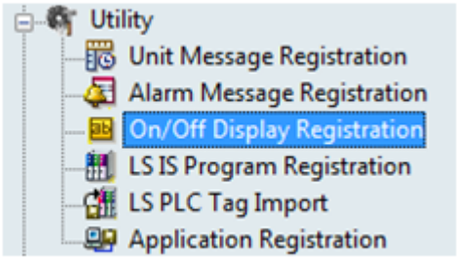
21.3 On/Off Display Message Registration

A function to set up a string to be displayed regarding to On/Off of the tag digital value.

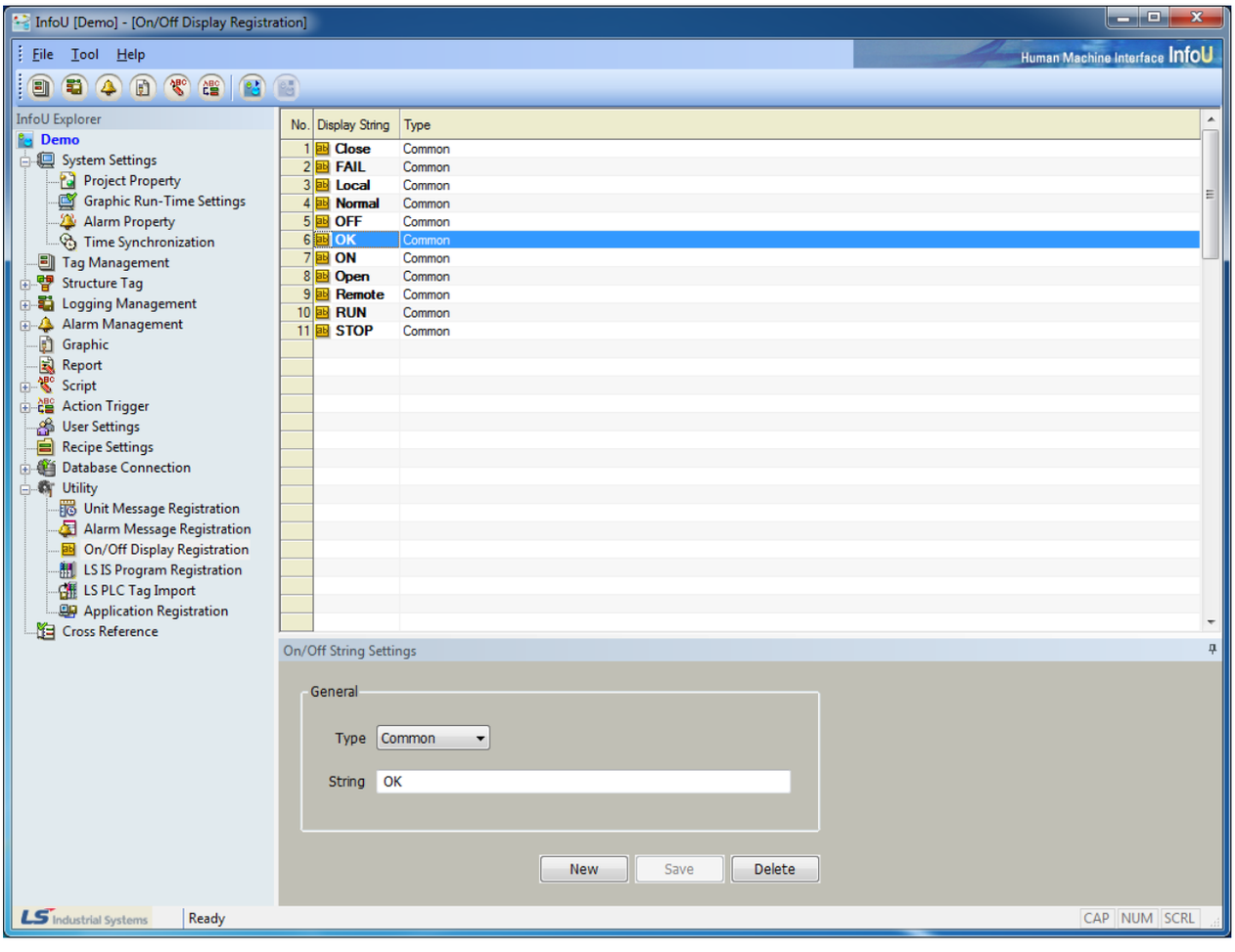
(1) Start

Select 'On/Off Display Message Registration' in the menu or the InfoU explorer.





A list of the registered On/Off messages is displayed as seen in the figure below and the setting screen is shown on the bottom.



(2) Settings

1) Type

●Three types such as [Common], [On (1)] and [Off (0)] are available and this information is mandatory

●Up to 63 Korean letters and 126 Alphabet letters can be inputted and it is one of essential information to be inputted.

2) String

●Input a string to be displayed.

●Up to 63 Korean letters and 126 Alphabet letters can be inputted and it is one of essential information to be inputted.

3) New

●The existing [String] is initialized to allow the user to input a new one.

4) Save

●Select [Save] button to save the inputted On/Off string and display it on the notice window.

5) Delete

●Select [Delete] button to delete the selected item display it on the notice window.

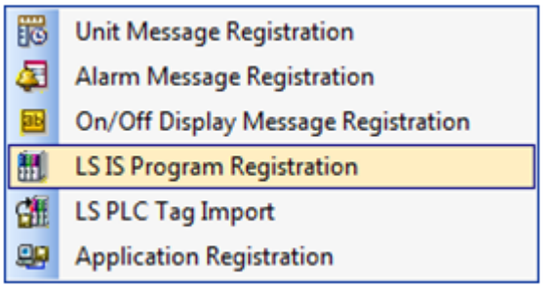
●It is also available to select multiple items in the list and delete them at once.

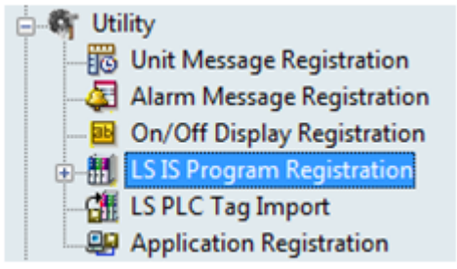
21.4 LSIS Program Registration

A function to register GLOFA-GM programs such as GMWIN.exe, PLC products of LS Industrial System, in the InfoUD to execute GMWIN programs and also to import information on the point engineered by GMWIN.exe in form of tag information

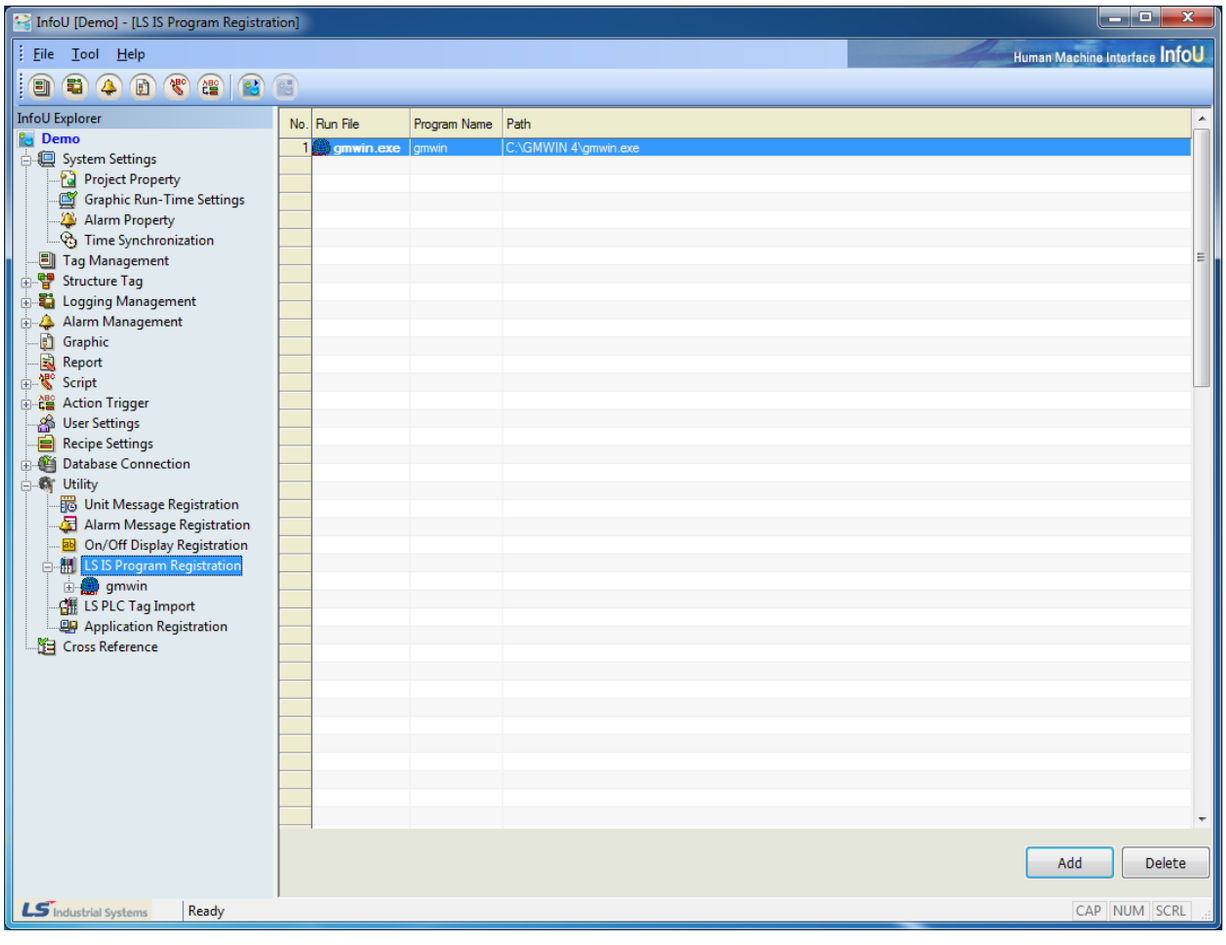
(1) Start

Select 'LS IS Program Registration' in the menu or the InfoU explorer.



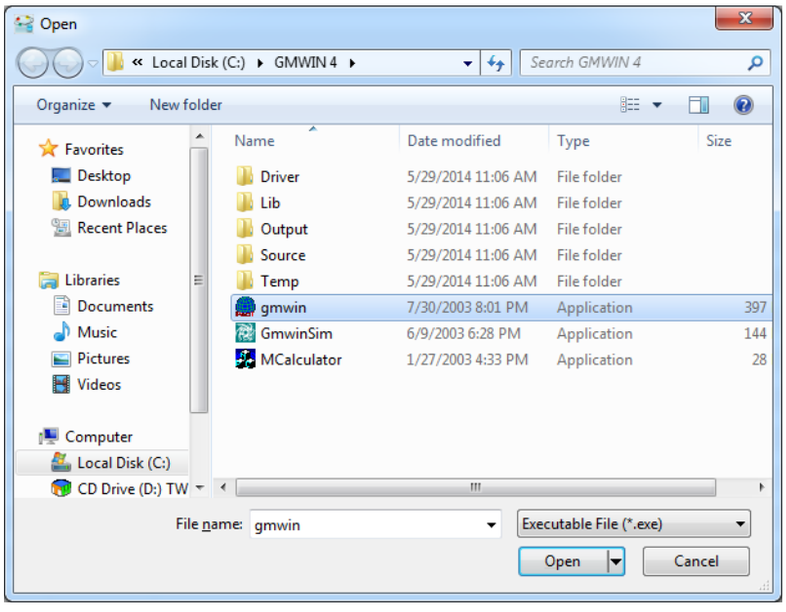


A screen as seen in the figure below is displayed and [Add] and [Delete] button appear on the bottom.

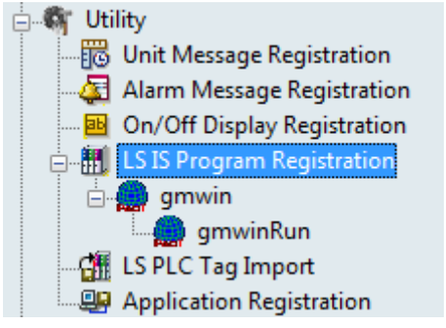


(2) Add

1) Select [Add] button to display the [Open] dialog box. Select GMWIN.exe-LS PLS program-on the [Open] dialog box.



2) Once the selection is completed, the following menu is created on the InfoU Explorer.



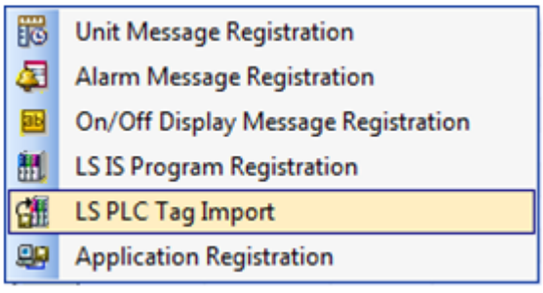
3) Run

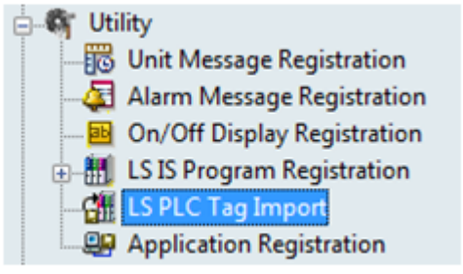
If [GMWIN Run], one of registered programs, is selected on the [InfoU Explorer], GMWIN.exe Program is run.

21.5 LS PLC Tag Import

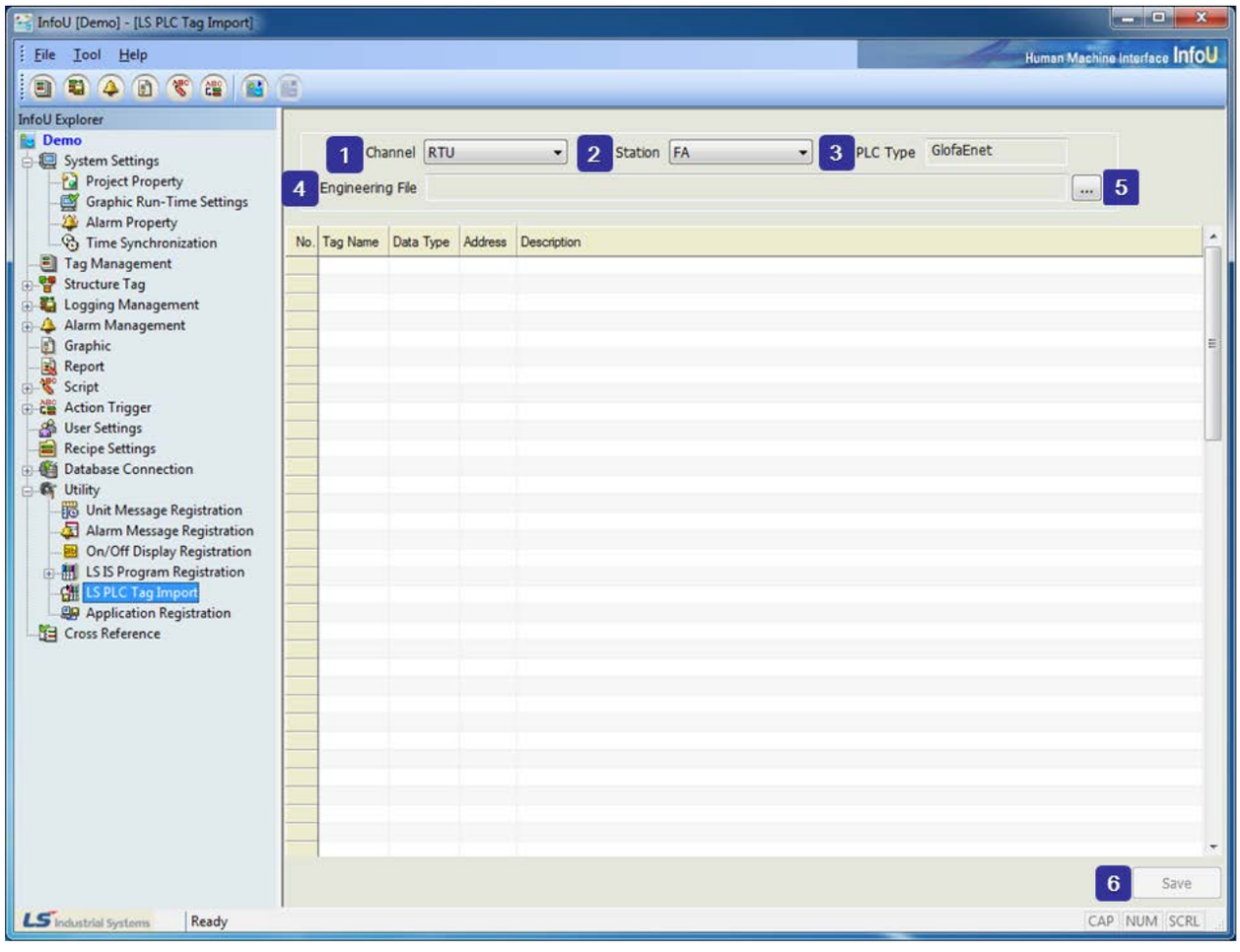
(1) Start

Select 'LSPLC Tag Import' in the menu and tool explorer and then, run it.



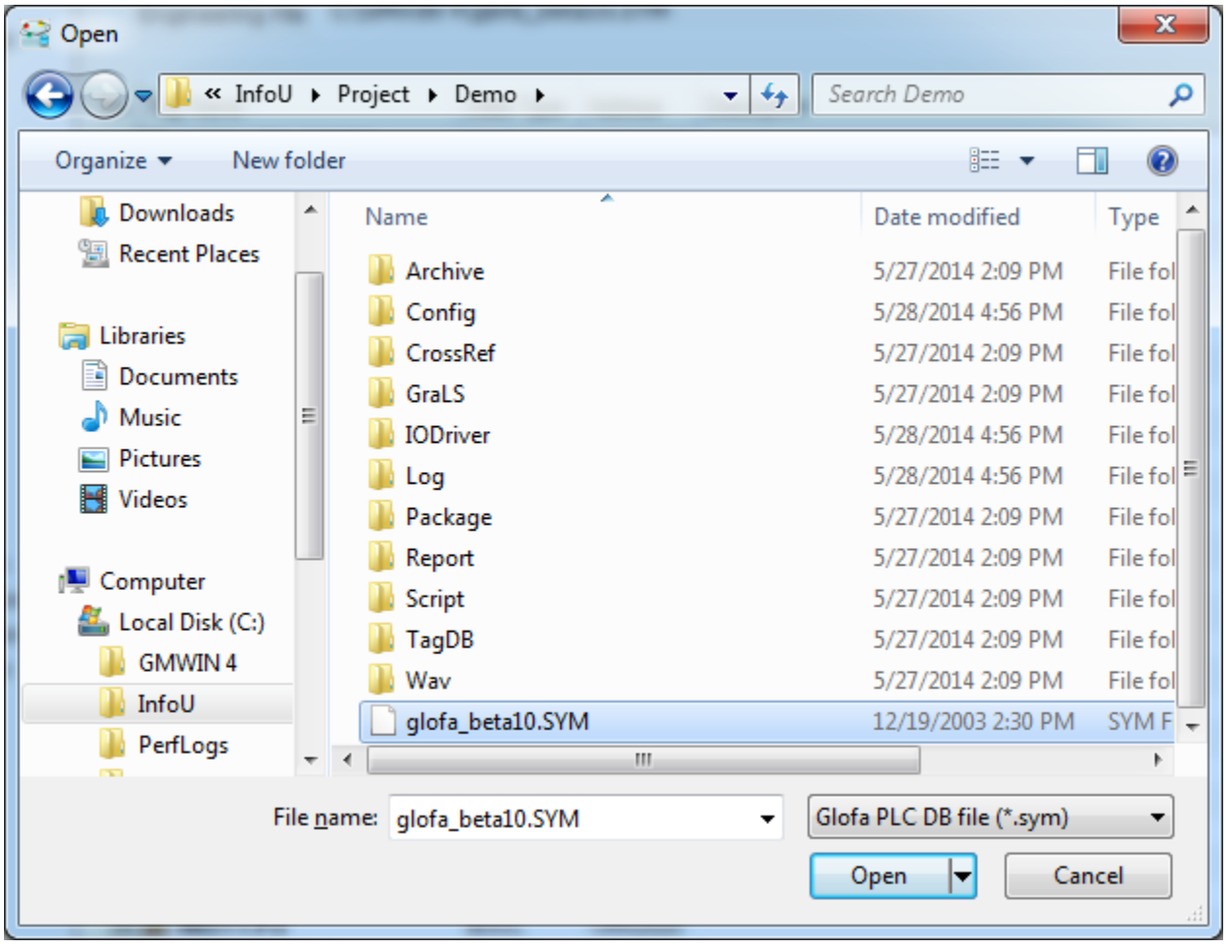


(2) Import a LS PLC Tag

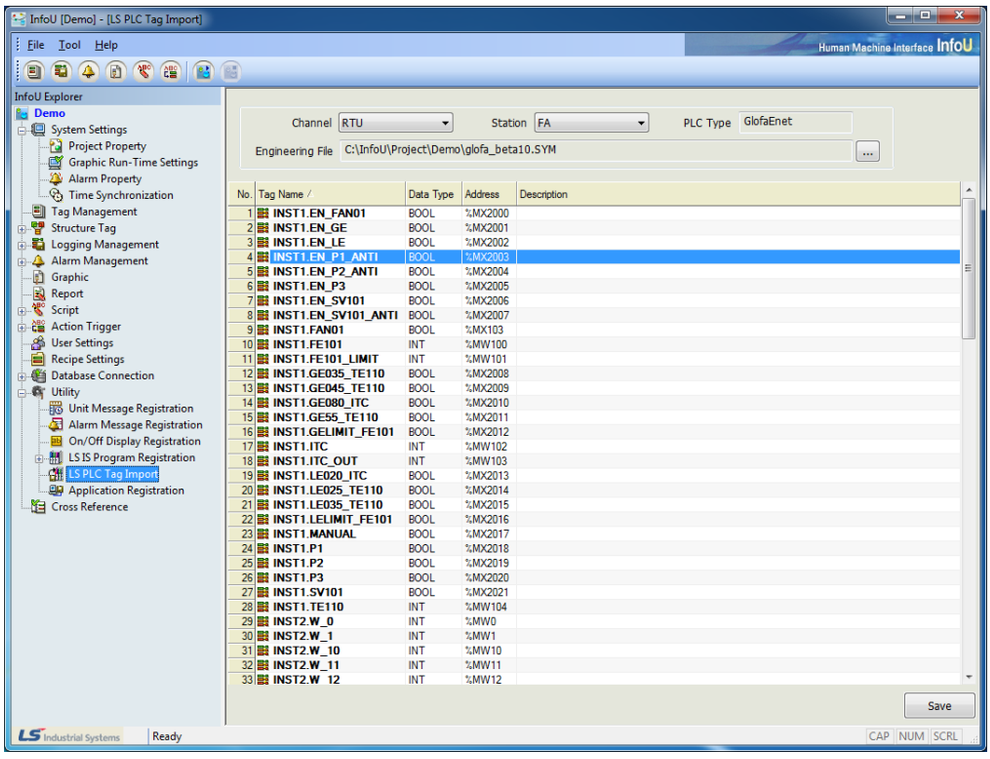


If [LS PLC Tag Import] is selected on the InfoU Explorer, menu bar or tool bar, the screen above is displayed.

1) GLOFA, XGK or XGI communication channels registered in the current project are displayed (GLOFA-Enet/Cnet/Loader, XGK-Enet/Cnet, XGI-ENet/CNet).
2) If a communication channel is selected, a list of the stations registered in the relevant channel is displayed.
3) The PLC type of the currently selected communication channel is displayed.
4) The path of the PLC engineering file to be imported is is displayed.
5) A PLC engineering file can be slected.
6) For GMWIN files, a *.sym file, for XG5000 engineering files, a *.csv file can be slected according to their types.



a. Select an engineering file of the relevant PLC project.
b. Click on [Open] to display the tag list below on the screen.

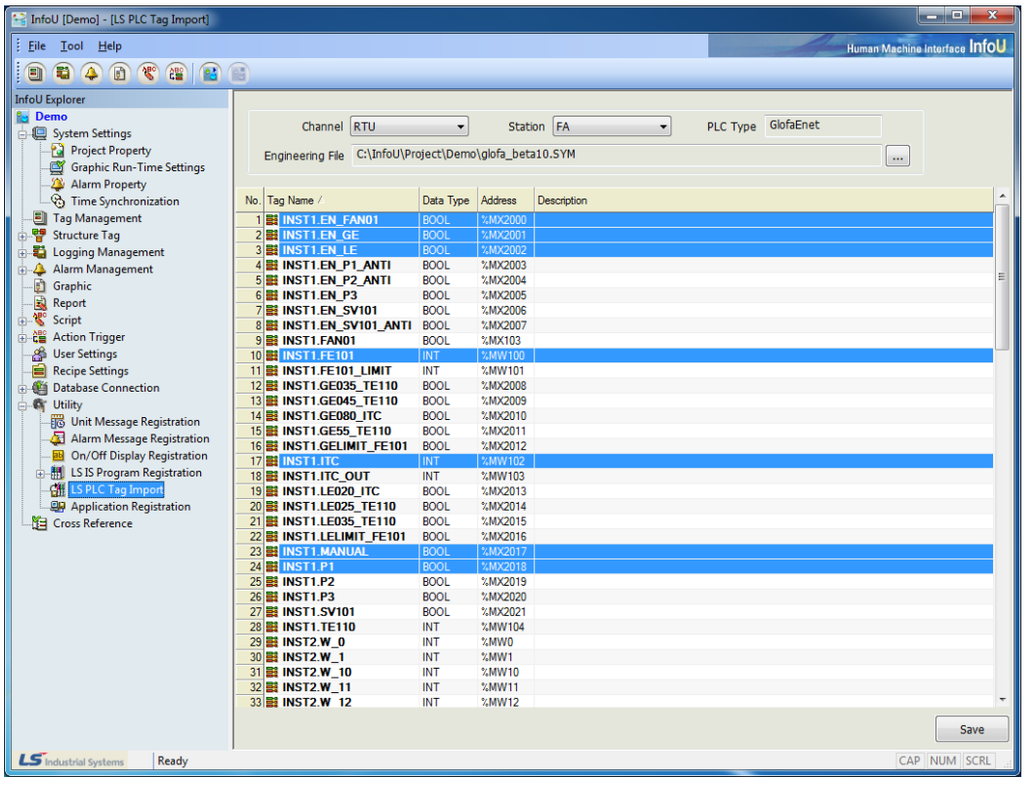


7) Save a PLC Tag

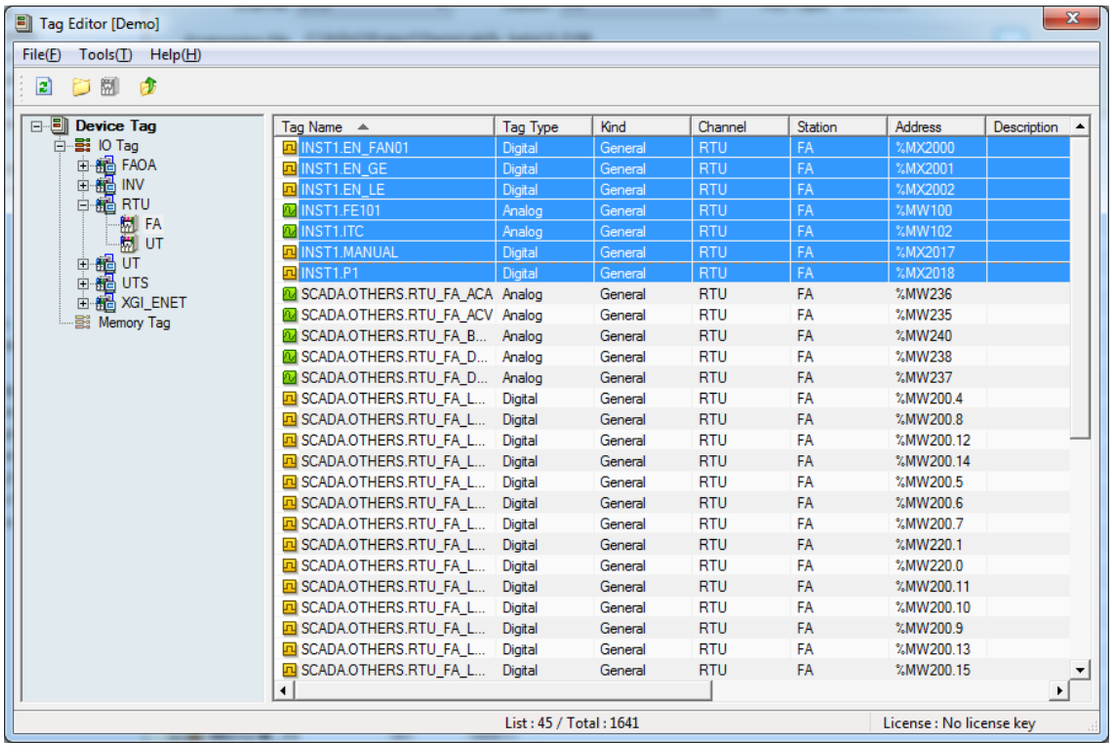
a. Select [IO Device] and [Device Name] to save.

b. Select the LS PLC Application through mouse drag or Ctrl + LClick, Shift + LClick, Grid RowColumn Header (Select All).

c. Click on [Save] button.



- If you select the menu, [Tag Management] you can check the tag list belonging to the IO Device/equipment selected during storage



- The program name created during the PLC engineering operation is inputted as a tag group and variables are inputted as individual tags.

Data types are saved according to PLC data types as follows:

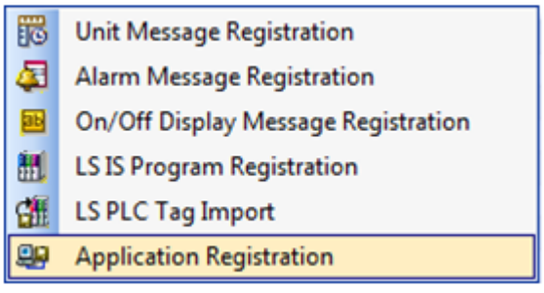
PLC Type	Tag Type	Device Type
tBOOL	BOOL	BOOL
tBYTE	UINT8	UINT8
tWORD	UINT16	UINT16
tDWORD	UINT32	UINT32
tLWORD	DOUBLE	UINT64
SINT	INT8	INT8
tINT	INT16	INT16
DINT	INT32	INT32
LINT	DOUBLE	INT64
USINT	UINT8	UINT8
tUINT	UINT16	UINT16
UDINT	UINT32	UINT32
ULINT	DOUBLE	UINT64
REAL	FLOAT	FLOAT
LREAL	DOUBLE	DOUBLE
TIME	UINT32	UINT32
tDATE	UINT16	UINT16
TIME_OF_DAY	UINT32	UINT32
DATE_AND_TIME	DOUBLE	UINT64
PLC_STRING	STRING	STRING
BIT	BOOL	BOOL
NIBBLE	UINT8	BIT4
DIRECT_TYPE	Not supported	
ENUM_TYPE		
ENUM_MEMBER		
ARRAY_TYPE		
ARRAY_MEMBER		
STRUCTURE_TYPE		
STRUCTURE_MEMBER		
FB_NAME		
LABEL_TYPE		
VAR_ARRAY_MEM		
FB_OUTPUT		
NOTYPE		
VAR_TYPE_NUM		

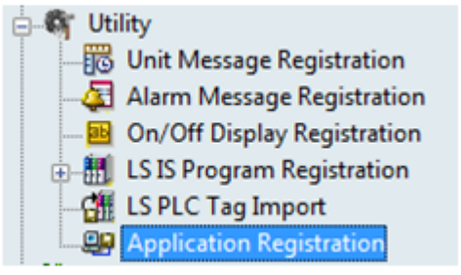
21.6 Application Registration

A function to register other programs of the computer in which InfoUD is installed in InfoUD to run external programs.

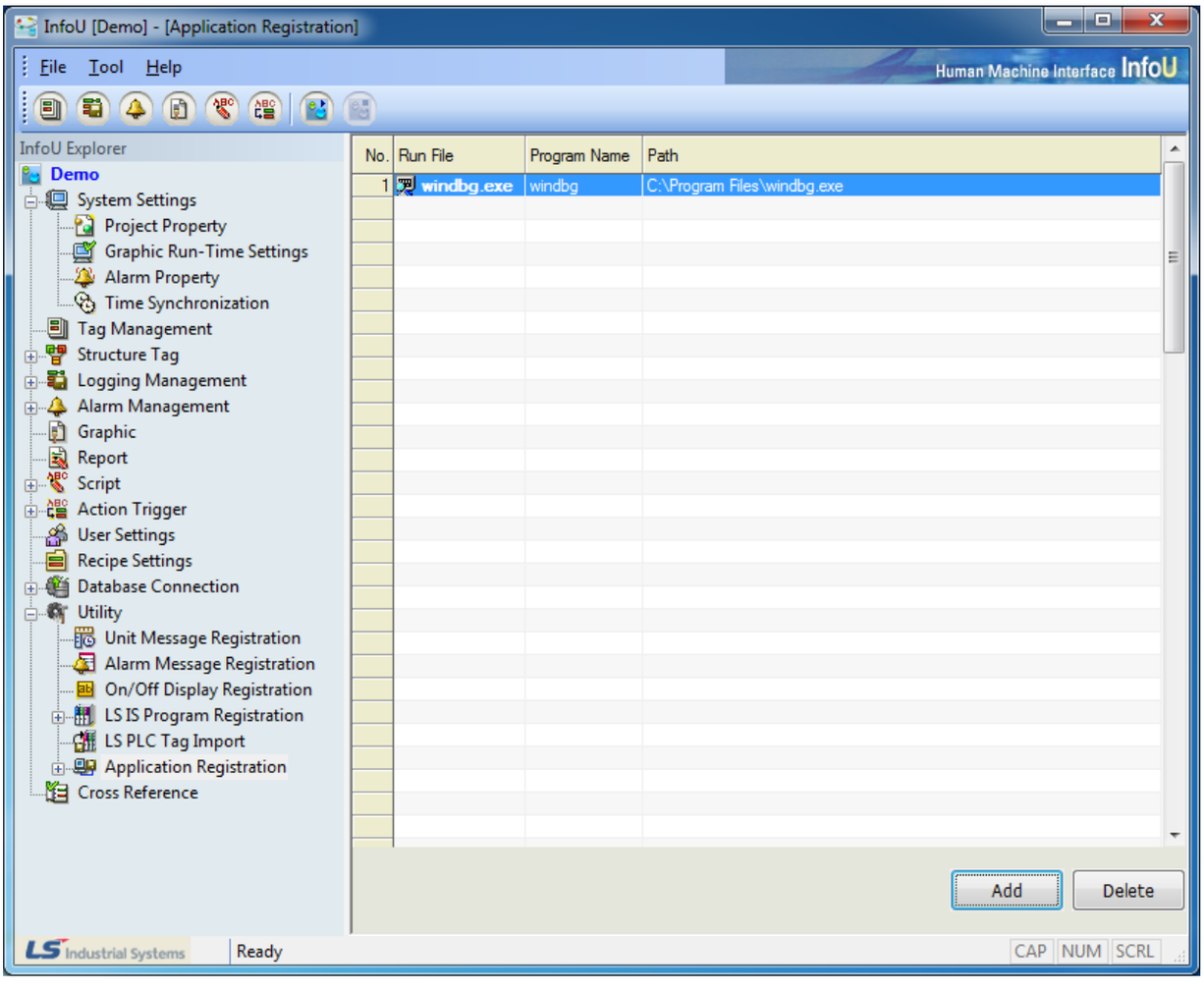
(1) Start

Select and run 'LS IS Program Registration' in the menu or the InfoU explorer.



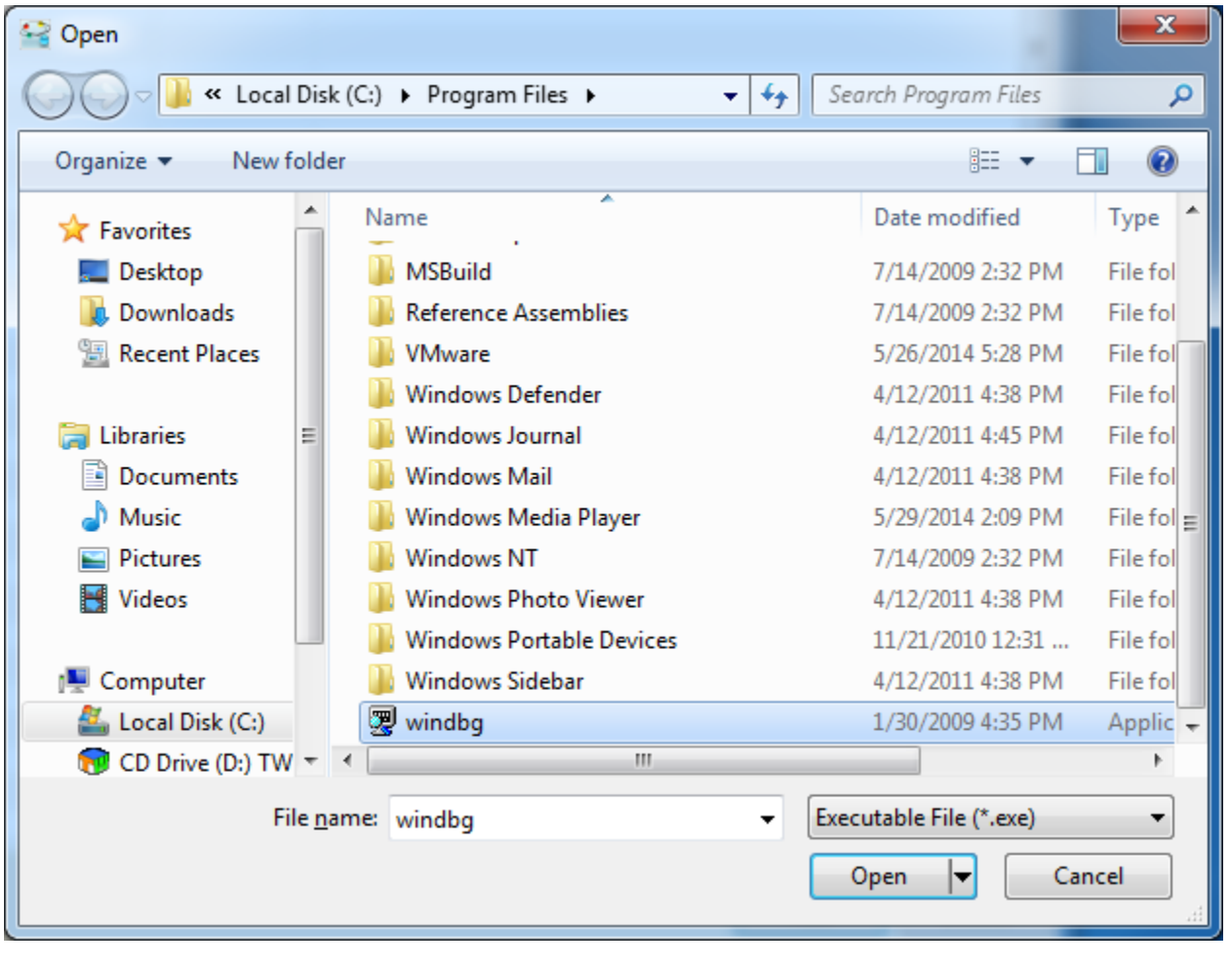


A screen as seen in the figure below is displayed and [Add] and [Delete] button appear on the bottom.

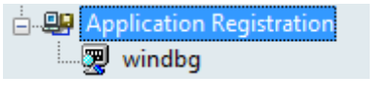


(2) Add

1) Select [Add] button to display the [Open] dialog box and select *.exe of the program to be registered on the [Open] dialog box.



2) The list above is displayed and the relevant program is registered as an icon in the Application Registration menu of the InfoU Explorer.



3) Run: Click on the icon to run the relevant program.

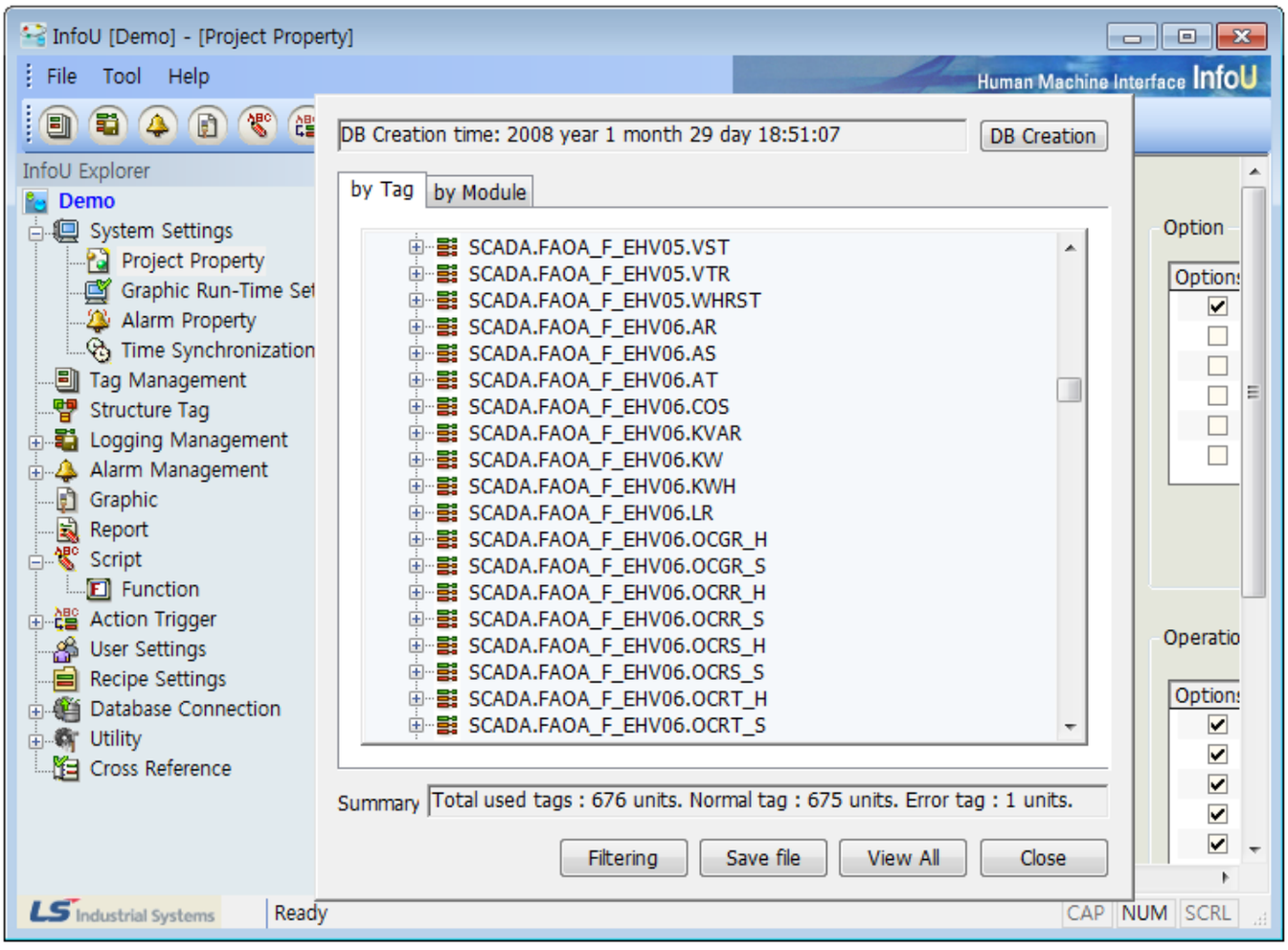
(3) Delete

1) Select a program to delete and select [Delete] button to display the dialog box

2) Once deleting is completed, the ‘Delete Notice’ window appears and the sub item that used to be displayed on the [InfoU Explorer] is also deleted.

Chapter 22 Cross Reference

The correlation of the registered tags such as graphic file, history settings (collection model), alarm group and report can be inquired on one screen. Especially, it has a function to detect any wrong use of deleted tags so that it is possible to conduct more effective engineering operations and in addition.

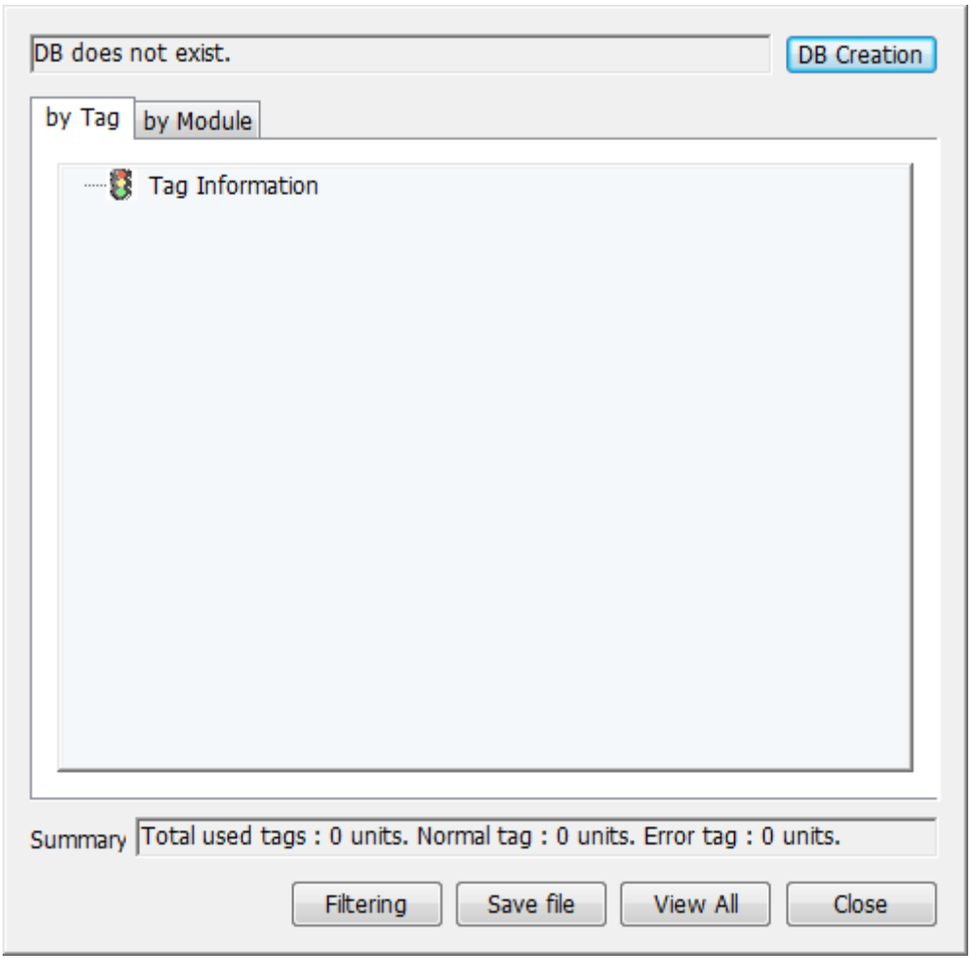


22.1 Function and Property

Functions	Descriptions
Tag Use Trace	It is possible to inquire information on the modules used for each tag at one place and the use details of each tag can be traced conveniently.
Filtering	Through filtering by tag, the correlation can be inquired only among the desired tags.
File Save	The inquired contents are saved in CSV files to make it possible to edit and output in various types of documents.
Inquiry Function by tag or by module	Two types of correlation inquiry screens are presented, one is to compile the used modules by tag and the other is to compile the used tags by module.
Jump	This jump function, which enables to skip to the graphic editor.

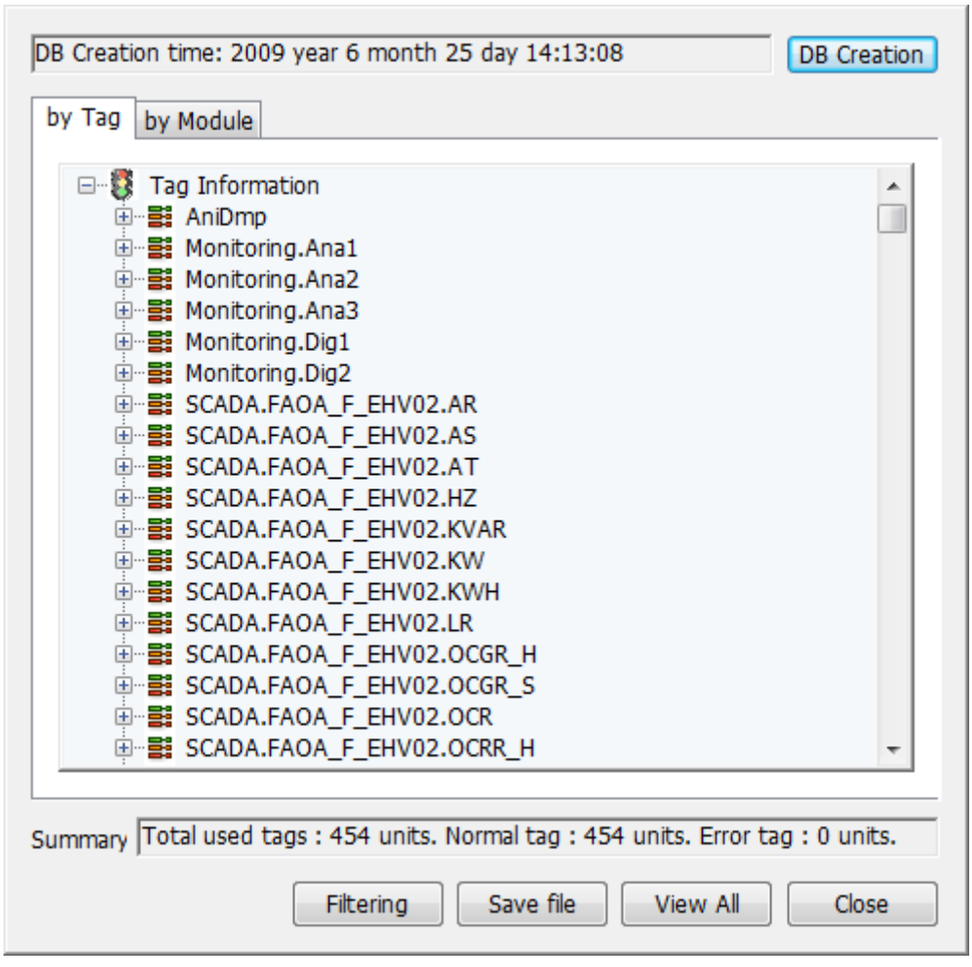
22.2 Cross Reference DB Creation

In order to use the cross reference, you need to create the DB manually after completing all engineering works. After engineering works, if you run the cross reference, the blank screen will show up as below.



To create the cross reference DB, click the [DB Creation] button.

The cross reference module retrieves the engineering data and creates the cross reference DB. The cross reference DB is the self-DB collecting correlations between modules and tags. In addition, afterward, if you perform InfoU engineering works such as addition and edition of tags and graphic pages, etc., the details will not be automatically applied to the cross reference DB so you should create the database manually.



The screen of the cross reference is composed of the following items.

Items	Descriptions
DB creation time	It indicates the time that the cross reference DB is created. You can determine whether the cross reference DB is up to date or whether it needs to be created by comparing the creation time after engineering works.
DB creation	It is the button to create the cross reference DB.
By tag	It is the tab to retrieve the created data by tag.
By module	It is the tab to retrieve the created data by module.
Summary	It shows the summarized information of the created data. You can correct the wrong engineering tasks by checking the normal and error tags.
Filtering	It retrieves the cross reference DB by tag.
Save File	It saves the retrieved details to the CSV file.
View All	It is used to move to the ‘View All’ screen after filtering.
Close	It closes the cross reference screen.

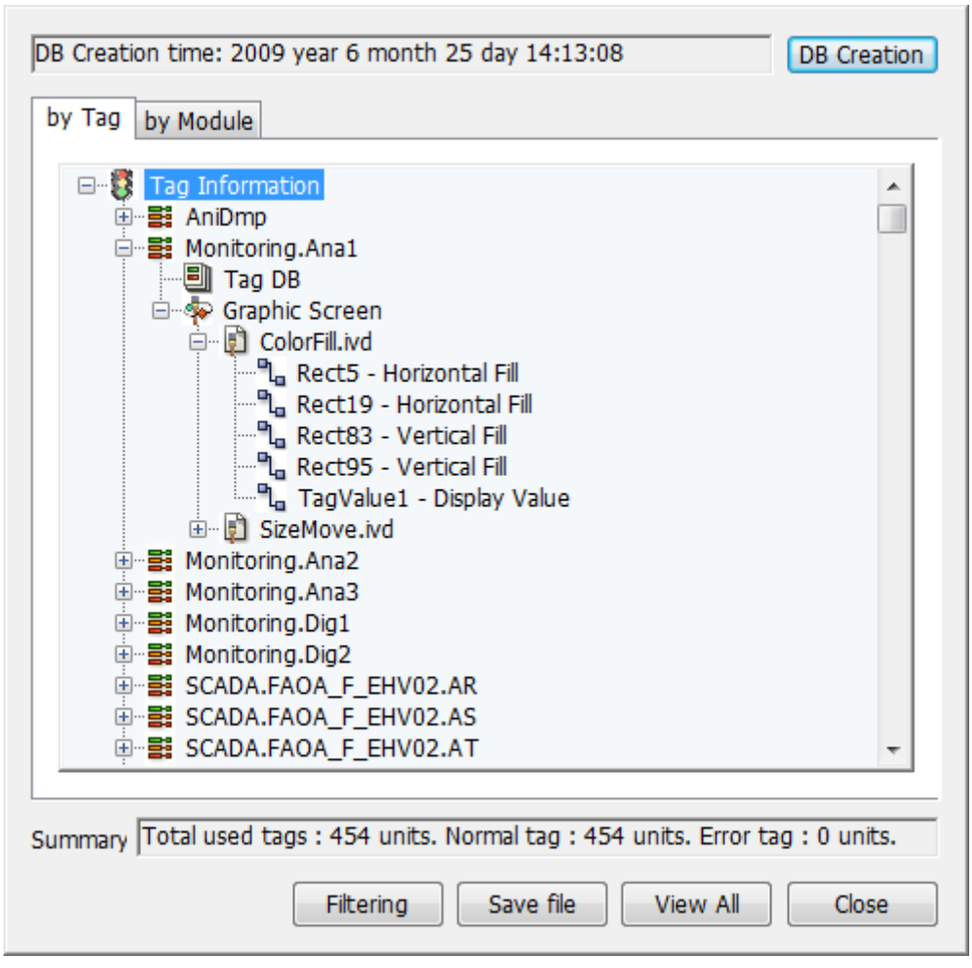
22.3 Engineering Function

You can perform the below functions using the created cross reference DB.

22.3.1 Search by Tag

The ‘Search by tag’ is the basic screen of the cross reference, which displays the tags used during engineering based on the tag name.

If you expand each tag, you can see which module is using the tag.



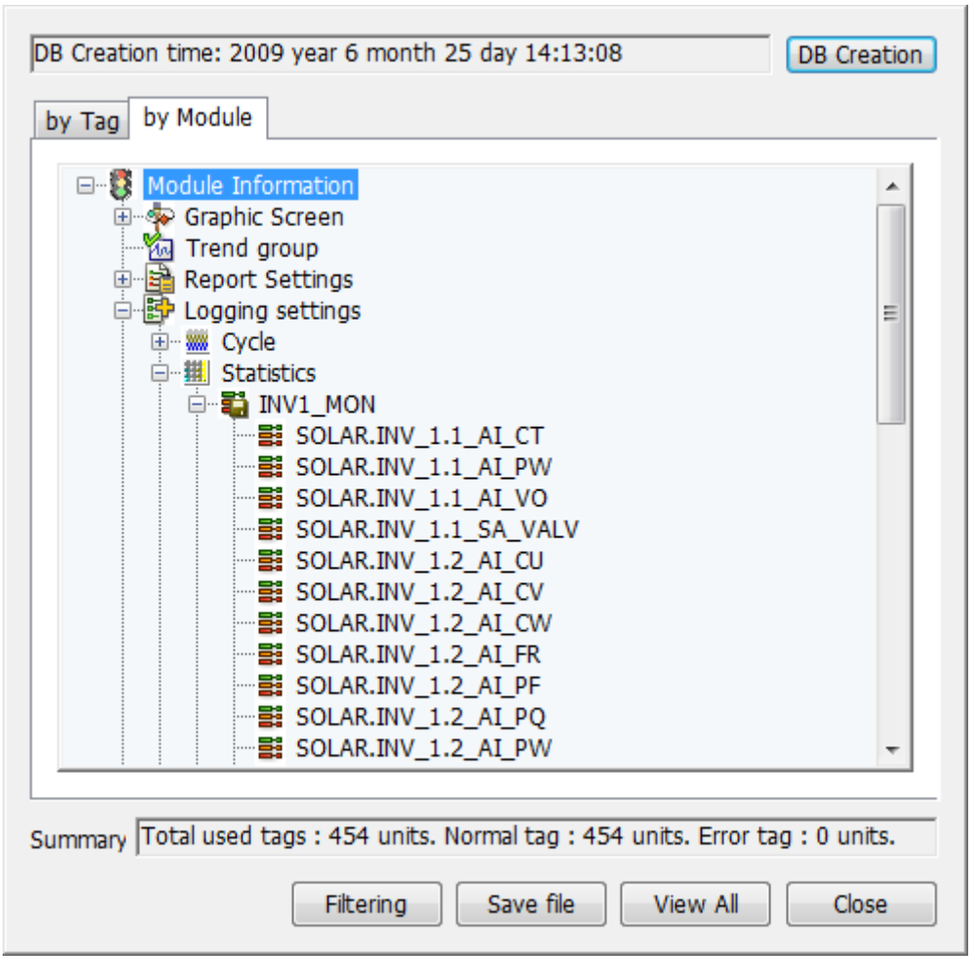
22.3.2 Search by Module

The function develops graphic screen, trend group, report settings, logging settings, tag used for the module of the alarm group based on the module.

If you expand each module, you will see which module is using the tag.(‘Trend Group’ is not currently used, it is the spare item specified for further expanding function).

The development order from each module to tags is as below.

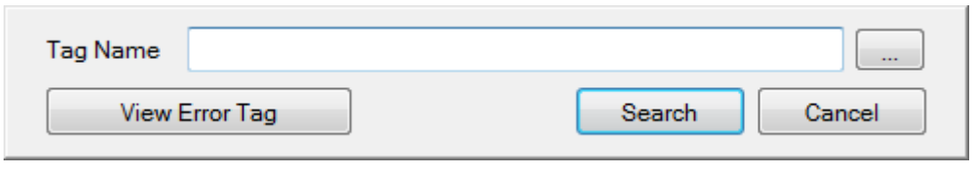
Module	Development
Graphic screen	Graphic page file(*.ivd) → object → tag and dynamic properties
Report Settings	Report name → Sheet name → vale and tag types
Logging Settings	Logging group types → logging group name → tag
Alarm Group	Alarm group name → tag



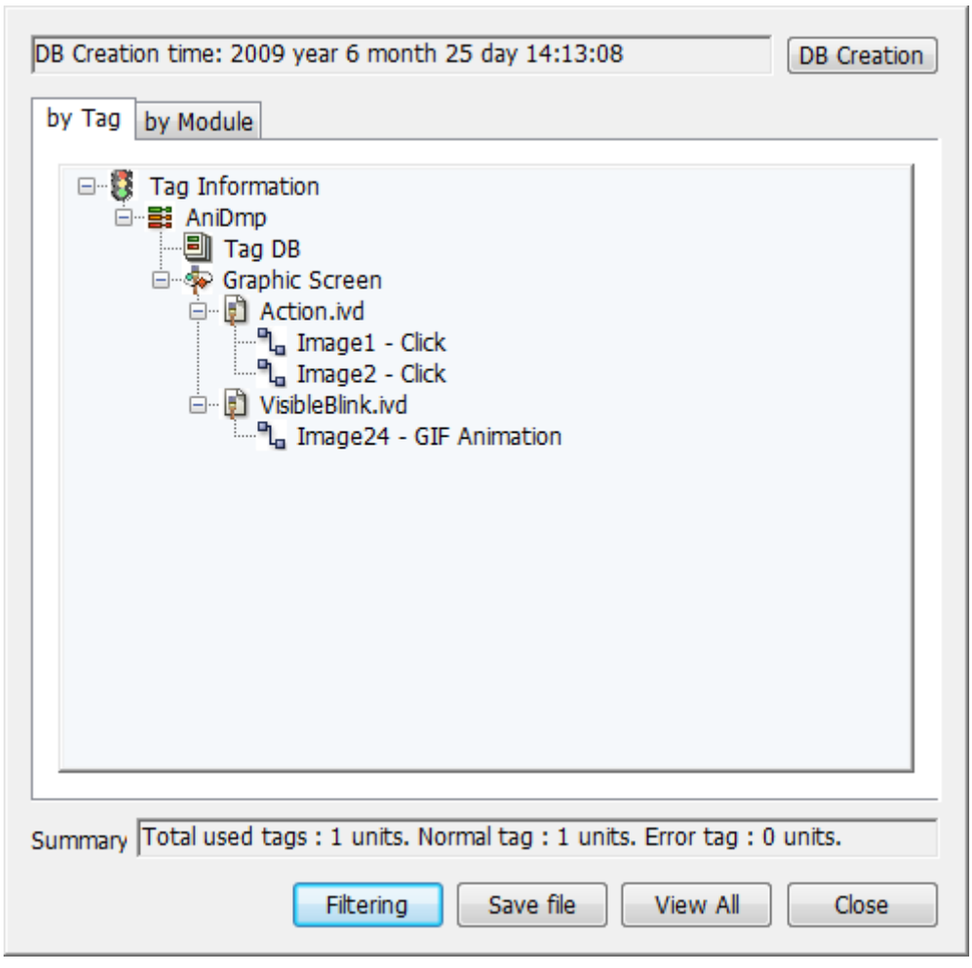
22.3.3 Filtering

It is the function to search the cross reference DB by tag name.

If you click the [Filtering] button, the below screen where you can enter the tag name will show up.



A user can enter the tag name directly or select the tag by clicking the tag browser button[…]. Then, if you execute the [Search] function, the cross reference DB will be retrieved by the tag name and the result will be displayed.



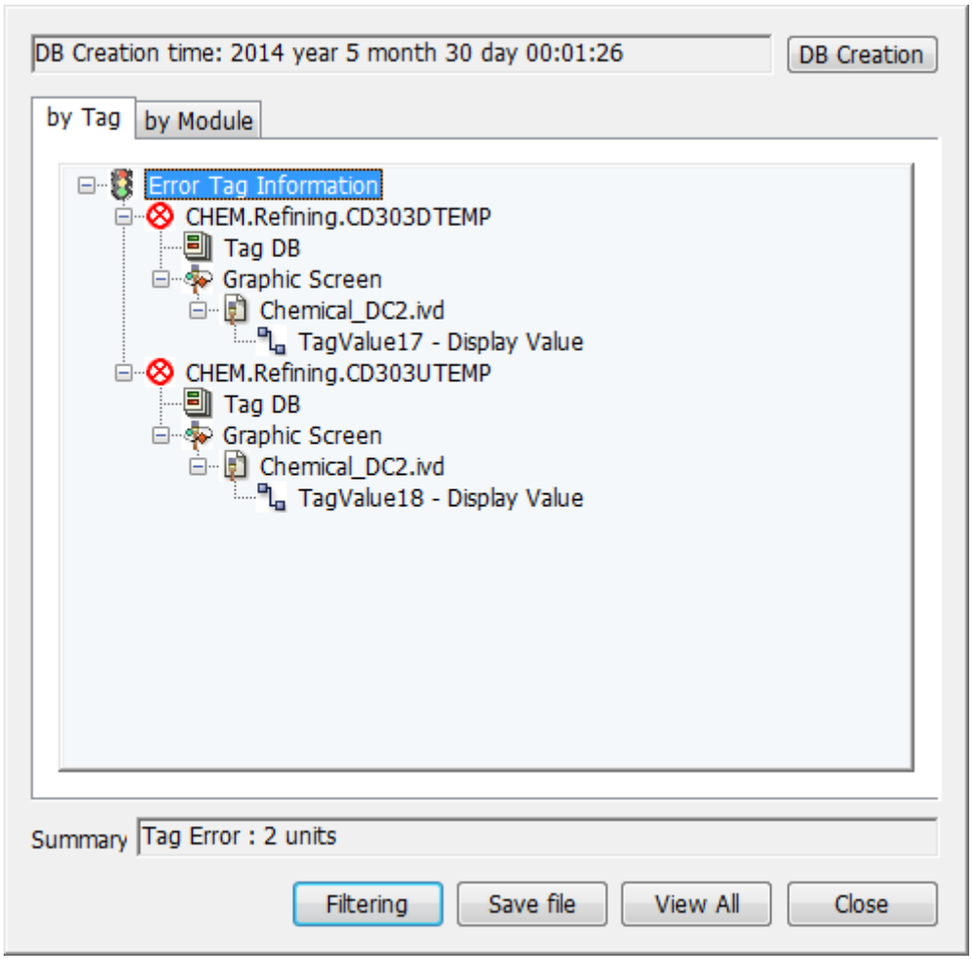
In order to return to the previous full screen, click the [View All] button.

22.3.4 Error Tag Search

The cross reference provides the function to detect wrongly linked tags.
For error tag search, click the [Filtering] → [Error Tag Information] button.
Error tags may be generated due to the following causes.

- In case non-existent tag is registered during engineering.
- In case the used tags are deleted after engineering works such as graphics, report, etc.

The below figure shows there is an error with the CHEM.Refining.CD303DTEMP tag. This tag is misused in the “Chemical_DC2.ivd” graphic page file and the figure shows non-exist tag is applied in the TagValue17 object of the file of the relevant page.



22.3.5 Jump Function

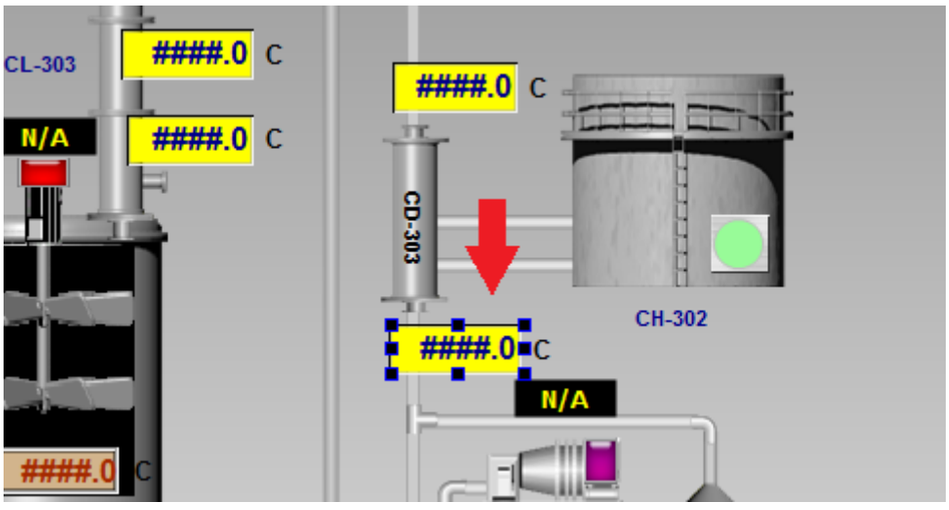
If you double-click the item on the cross reference window, you can move to the relevant module through the Jump Function. The Jump Function is provided for the graphic screen only. If you click the object in the tree of the cross reference window, the page will be opened and the focus will move to the relevant object.

Drawing the graphic page is composed of drawing each object, mapping dynamic properties into each object. After graphic page works are done, each tag mapping is linked within the graphic object and it is not visible to a user directly.

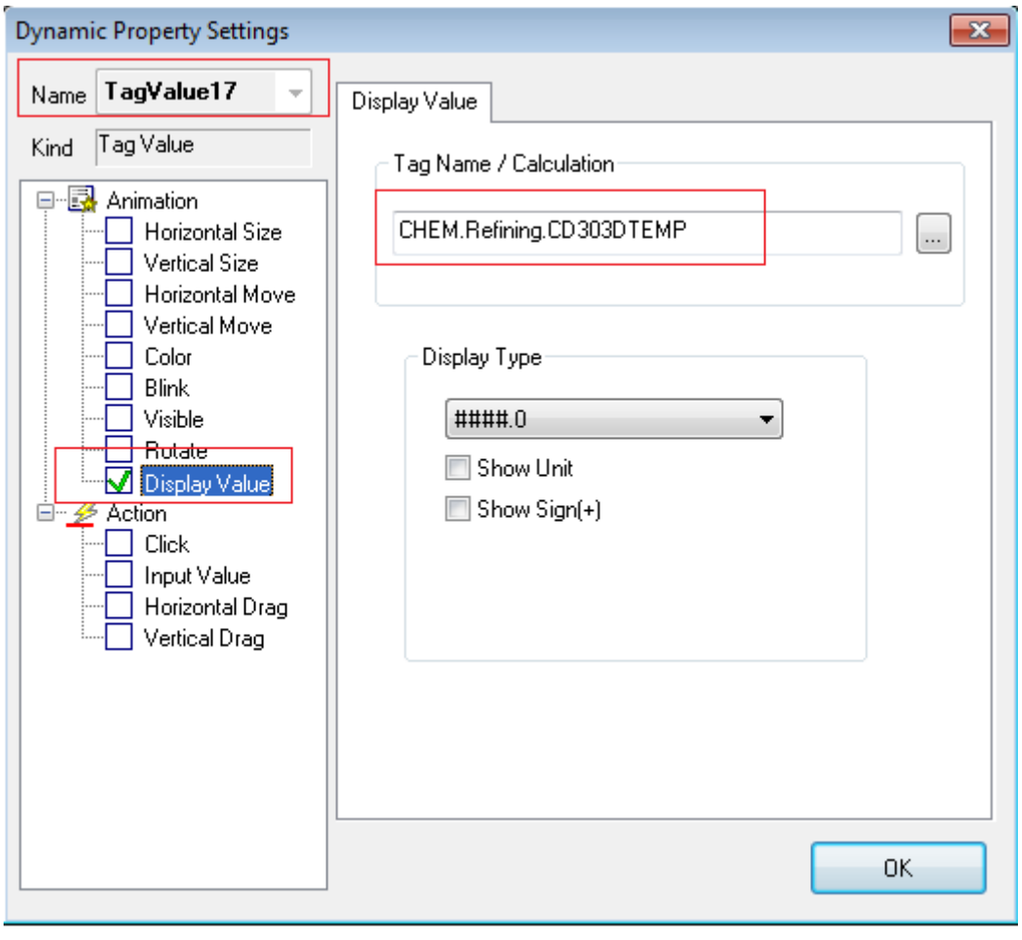
In many case, there are from tens of tags to thousands of tags mapped in one graphic page. If there is no cross reference function, you need to open each object one by one and check the tag manually. Through the Jump Function of the cross reference, you can perform such tasks more effectively.

This section describes the process from error tag search to finding the graphic object.

If you double-click “Line32 – Color” in the previous error tag information screen, the graphic page screen will be open with the situation that the Line32 object is selected.
The area marked by a red arrow indicates the object is selected.



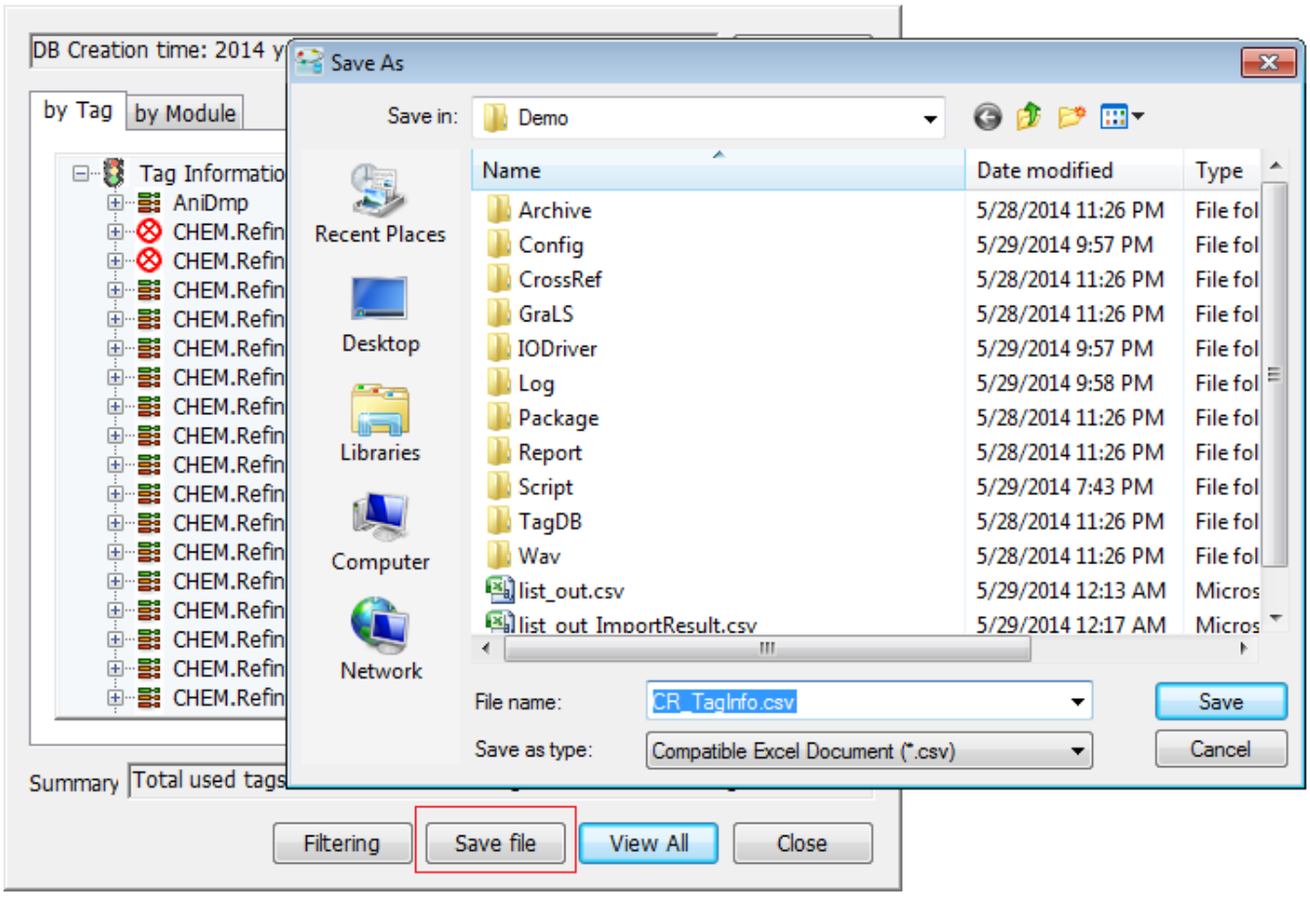
After clicking with the right mouse button on the object selected by a focus, if you run the [View Dynamic Properties] menu, you will see the tag called [CHEM.Refining.CD303DTEMP] is set in [Display Value] of the [Tag Value 17] object.



22.3.6 Save File

It is the function to save the results of the cross reference search to the CSV file.

If you press the [Save File] button in the search window, the dialog box asking the storage location will show up.



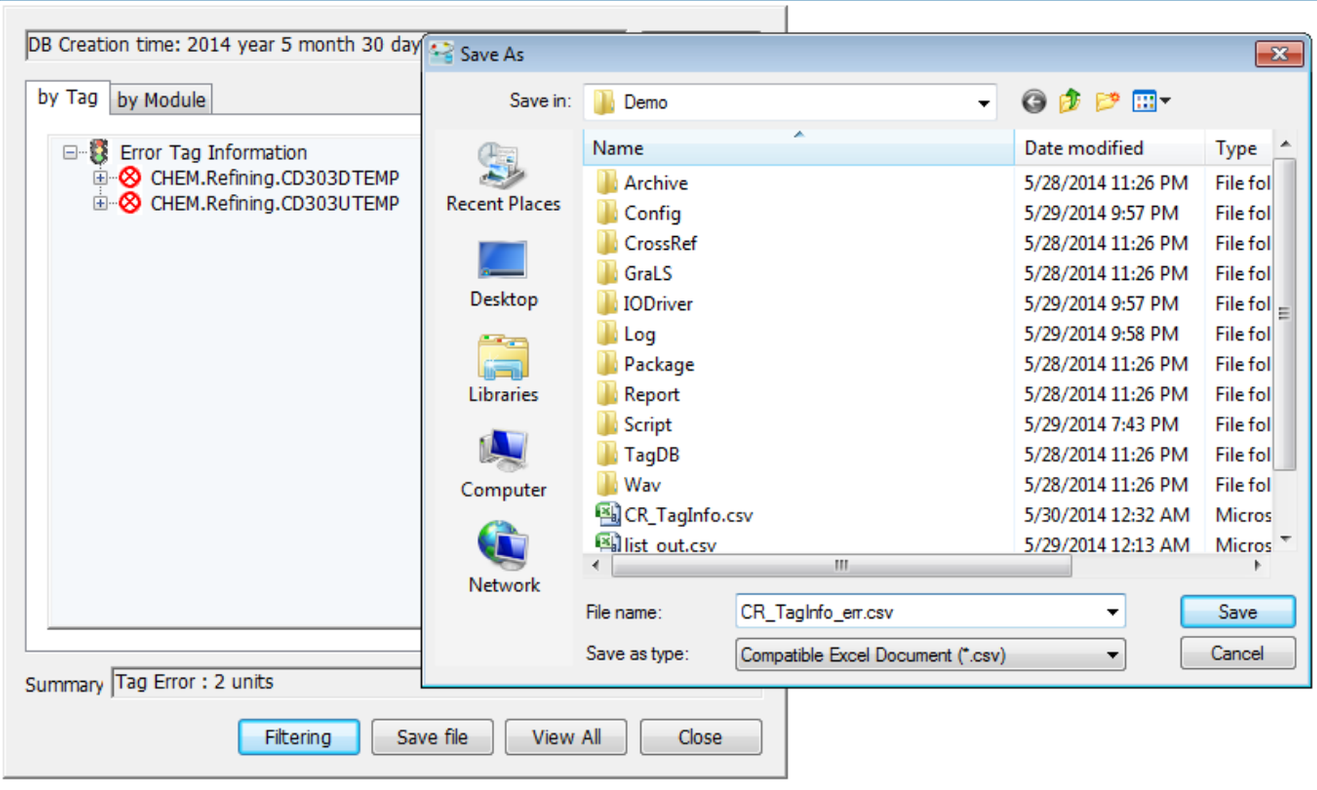
If you enter the file name and save it, the result of the cross reference retrieval will be saved to the CSV file.

This file can be used for various purposes such as engineering summary report, etc.

	A	B	C	D	E	F
1	No	Tag Status	Tag Name	Module Name	File/Model Name	Object/Description
2	1	Tag in Use	AniDmp	Graphic Screen	Action.ivd	TagValue1 - Display Value
3						Image6 - Visible
4						Image12 - Visible
5						Image1 - Click
6						Image2 - Click
7					VisibleBlink.ivd	Image24 - Frame Animation
8				Logging settings	EventTest	
9	2	Tag in Use(Unregistered)	CHEM.Refining.CD303DTEMP	Graphic Screen	Chemical_DC2.ivd	TagValue17 - Display Value
10	3	Tag in Use(Unregistered)	CHEM.Refining.CD303UTEMP	Graphic Screen	Chemical_DC2.ivd	TagValue18 - Display Value
11	4	Tag in Use	CHEM.Refining.CD304DTEMP	Graphic Screen	Chemical_DC2.ivd	TagValue19 - Display Value
12	5	Tag in Use	CHEM.Refining.CD304UTEMP	Graphic Screen	Chemical_DC2.ivd	TagValue26 - Display Value
13	6	Tag in Use	CHEM.Refining.CH302	Graphic Screen	Chemical_DC2.ivd	Circle9 - Color
14	7	Tag in Use	CHEM.Refining.CH303	Graphic Screen	Chemical_DC2.ivd	Circle1 - Color
15	8	Tag in Use	CHEM.Refining.CL303ATEMP	Graphic Screen	Chemical_DC2.ivd	TagValue1 - Display Value
16	9	Tag in Use	CHEM.Refining.CL303BTEMP	Graphic Screen	Chemical_DC2.ivd	TagValue3 - Display Value

In addition, you can save the specific tag only using the filtering function or run [Save File] after searching error tags. In this case, the tag list displayed on the screen will be saved.

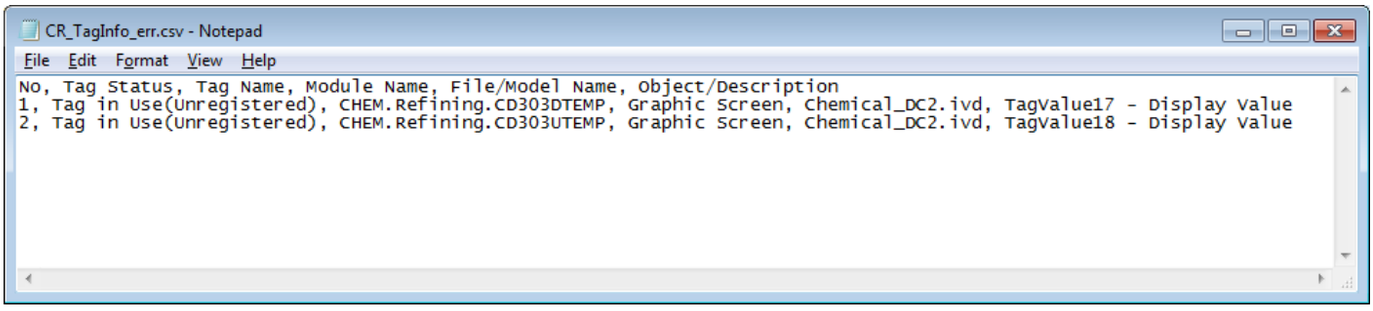
After running [Error Tag Information] in the [Filtering] menu, click the [Save File] button.



If you open the saved results in excel, you can see the below figure.

	A	B	C	D	E	F
1	No	Tag Status	Tag Name	Module Name	File/Model Name	Object/Description
2	1	Tag in Use(Unregistered)	CHEM.Refining.CD303DTEMP	Graphic Screen	Chemical_DC2.ivd	TagValue17 - Display Value
3	2	Tag in Use(Unregistered)	CHEM.Refining.CD303UTEMP	Graphic Screen	Chemical_DC2.ivd	TagValue18 - Display Value
4						
5						
6						

If you open the saved results in a notepad, you can see the below figure.



22.4 Runtime Function

After engineering works are done, the created cross reference DB can be used for the plant screen of the InfoU runtime, tag view, alarm popup menu, graphic object, etc.

For the description on the cross reference function in the runtime, refer to [Cross Reference Functions] of the [Graphic Runtime] manual.

Chapter 23 Web

23.1 InfoU Web Configuration

23.1.1 Prerequisite

(1) The web functions can be realized in the Microsoft Windows-based system and support Internet Explorer 6.0 or higher.

(2) In the Windows XP or earlier version, among the components, the Internet Information Service (IIS) should be pre-installed and be running. For the method how to install the Internet Information Service (IIS) in Windows XP, refer to [InfoU Web Q&A Collection] of the Appendix. (In the Windows 7 or higher version, if the IIS is not pre-installed, it will be installed automatically.)

(3) InfoU HMI Server has been installed.

(4) The table shown below, whether supports the web functions by OS.

Operation System		Availability	Concurrent connected user	Pre-installation of IIS
Windows ME, 2000 Server or less		X	-	N/A
Windows XP	Home Edition	X	-	Windows XP
	Professional	O	5 Users	
Windows 2003 Server		O	Default 5 Users For additional users, follow the OS License	X
Windows Vista	Home Edition	X	-	Windows Vista
	Professional	O	5 Users	
	Business	O		
	Ultimate	O		
Windows 7	Home Edition	X	-	Windows 7
	Professional	O	5 Users	
	Business	O		
	Ultimate	O		
Windows 2008 Server		O	Default 5 Users For additional users, follow the OS License	X
Windows 2008 Server R2		O	For Windows 2008 Server R2, only 64bit is produced	X

(5) You need the license key that can support the web function.

23.1.2 Configuration Procedures

(1) To use the web functions of the InfoU, the installation should be done based on the below procedures.

1) Put a check the items supporting web functions in [InfoU Engineering]➔[Project Property]➔ [Option].
2) Run the InfoU Web Navigation Management to set the Web Navigator Site.
3) Register the user and set the default screen by logging in with the InfoU (Admin account) in the InfoU Web screen.
4) When you log in with the registered user in the InfoU Web screen, the web monitoring screen will be displayed.

Notice

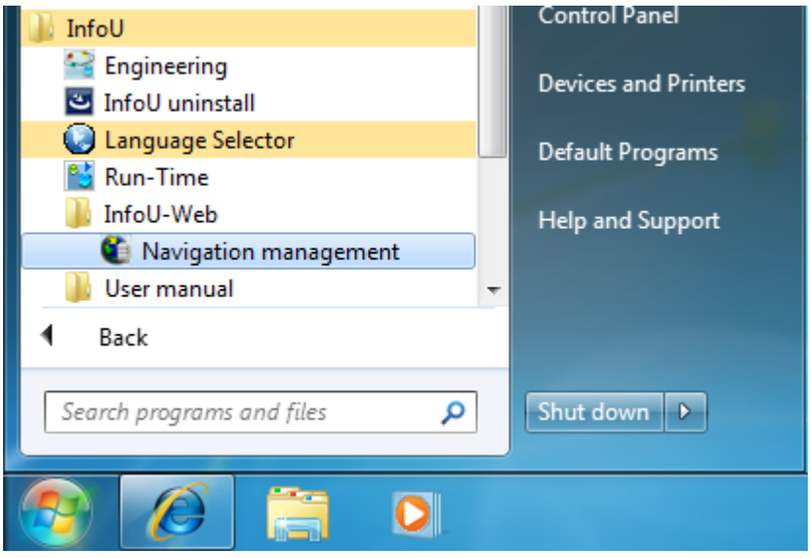
For your information, the manual was written in Windows 7 Enterprise K 32 Bit and Internet Explorer 9.
In case the Internet Explorer version is different from the Internet Explorer 9, the screen may differ slightly.
For errors occurred during installation or unexpected problems during operation, refer to [InfoU Web Q&A Collection] of the Appendix.

23.2 Creatiing InfoU Web Site

This Site Automatic Setup ensures even an inexperience user who does not have any expertise in web servers or website formation can configure a website easily and assess to the InfoU HMI system through the web browser from anywhere. The user can form a site he/she wants and create a new website. In addition, he/she may add a virtual folder to the existing website.

23.2.1 Execution of the Navigation Management

(1) Run the InfoU Web Navigator by selecting [Start]➔[All Programs]➔[InfoU]➔ [InfoU-Web] ➔ [Navigation management].



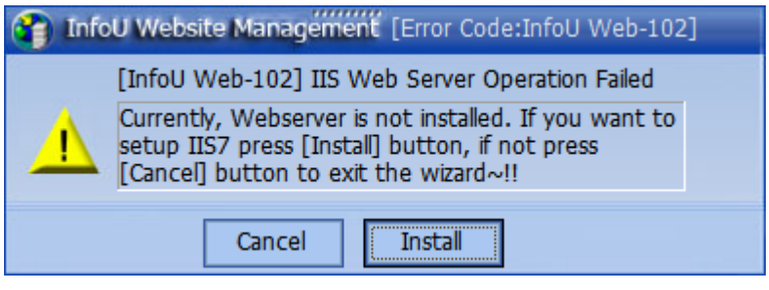
(2) Brief information on the program is displayed on the splash screen.



1) If the IIS7 is not installed, in Windows 7, it is installed automatically as below.

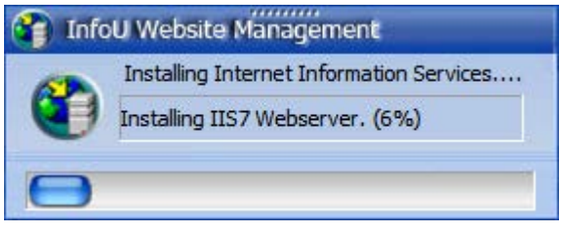
a. Message confirming installation of the IIS7

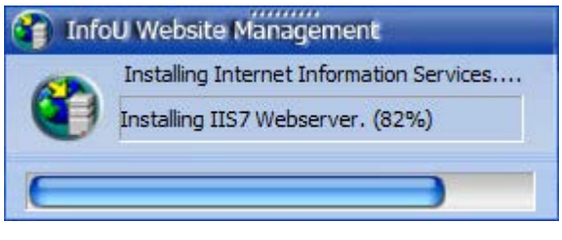
To the install the IIS7, click the [Install] button.



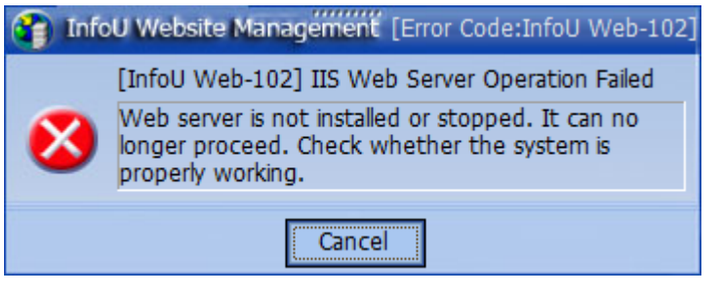
b. Message on the progress of IIS7 installation

When the installation proceeds is done 100%, it will be switched into the next screen automatically.



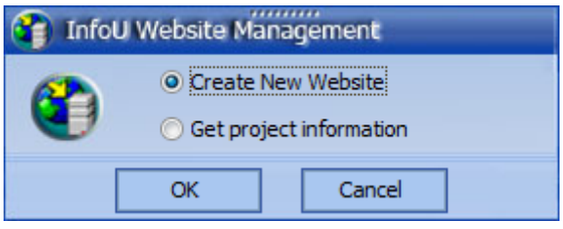


2) If the prerequisite is not satisfactory or automatic installation of the IIS7 is not performed normally, the below error message will be displayed and the program will not proceed any more. For more details on the error message and solutions, refer to [InfoU Web Q&A Collection] of the Appendix.



23.2.2 Creating New Website

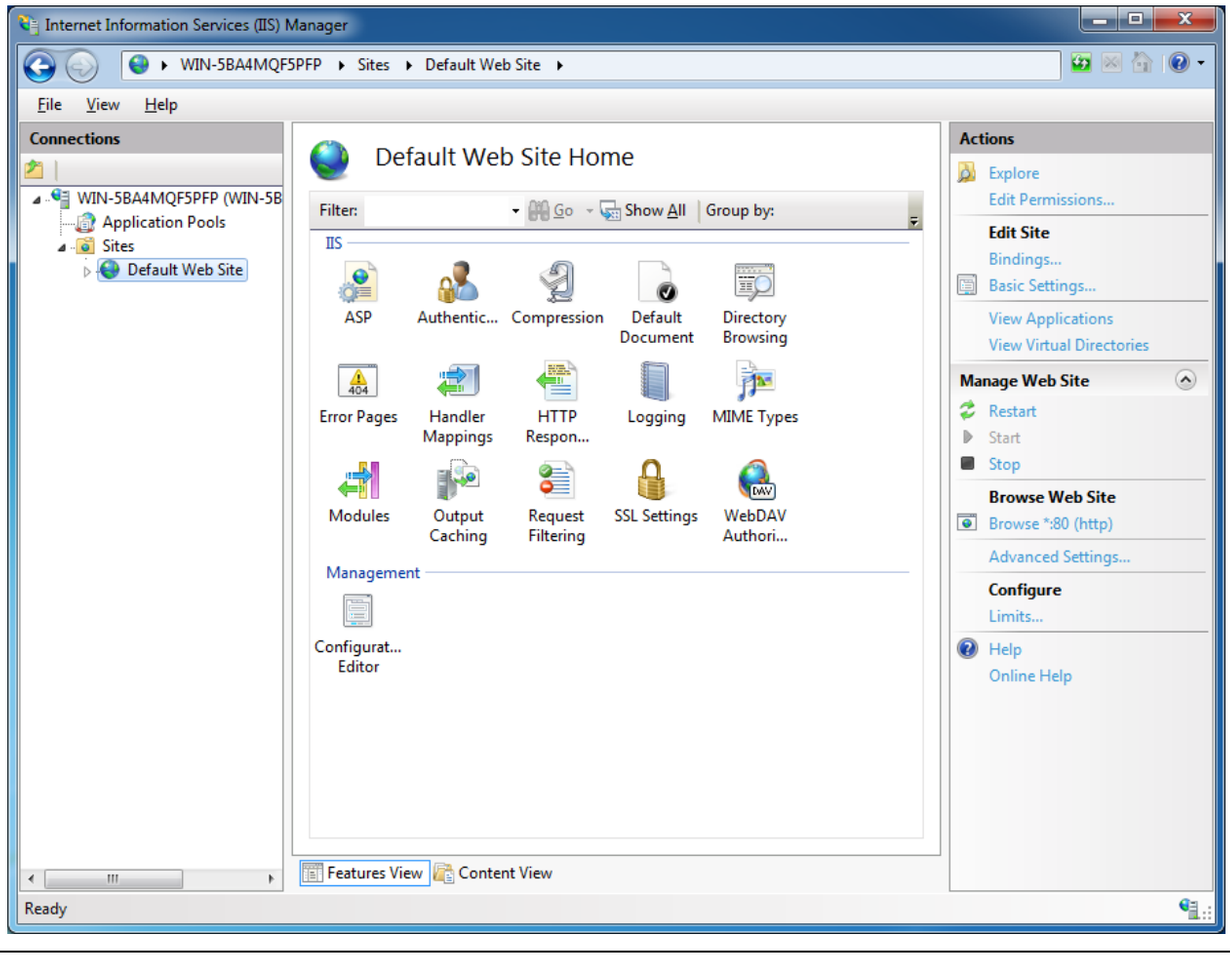
In the Internet Information server (IIS), you can create the website that you can directly manage. If there is no history of creating the InfoU Web site in the past, [Create New Website] is automatically selected as shown below.



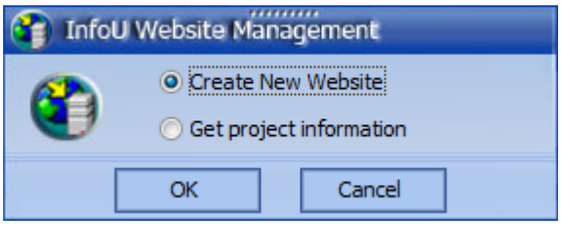
You can create websites as many as you want and manage them by site.

Notice

☞Before creating a new web site, if you check the information of [IIS(Internet Information Service) Administrator] in [Start] ➔ [Control Panel] ➔ [System and Security] ➔ [Administrative Tools], currently, only the basic website exists.



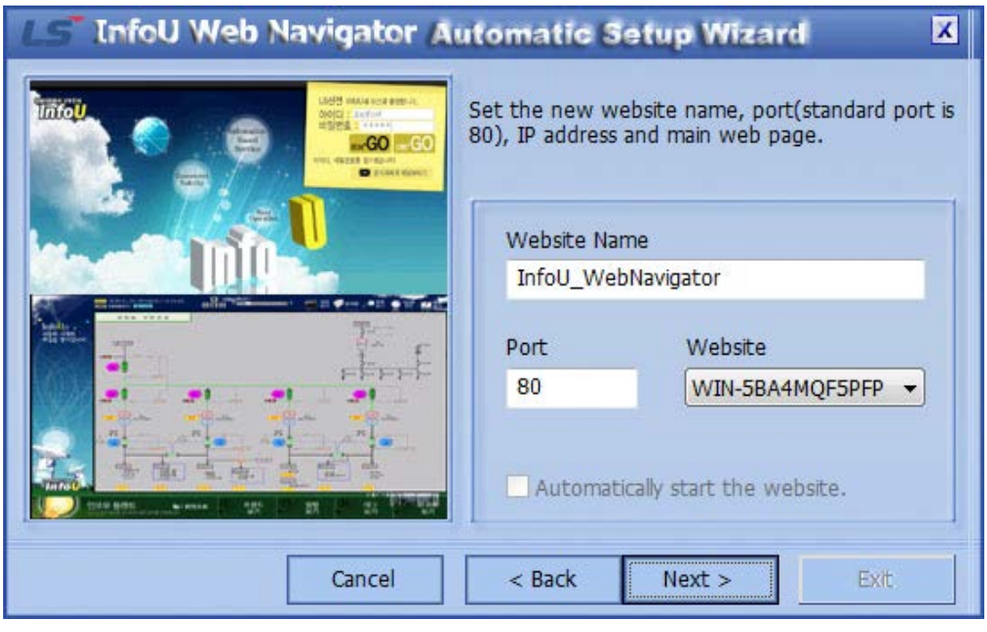
(1) After selecting [Create New Website], click the [OK] button.



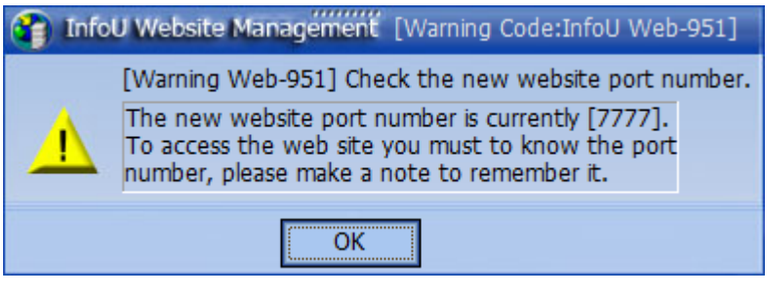
(2) After selecting [Create New Website], click the [Next] button.



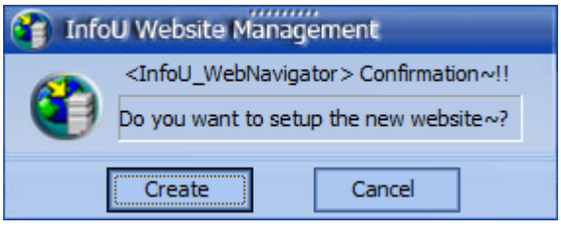
(3) Enter the site name and port No. that the InfoU Web user wants. After entering the desired name of the new website, click the [Next] button. The default access port is set as 80. If another web server is operated by 80 access port, a user can set other port No. depending on the environment.



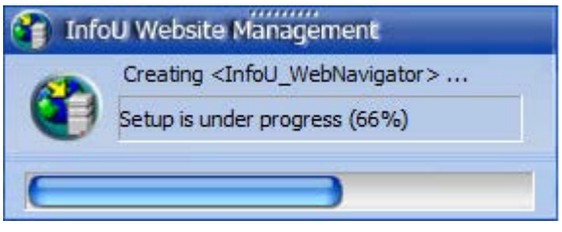
In the 64bit OS of Windows 7 or more, the default port is set as [7777] since the phenomenon that the default website is initialized into the basic site every time you reboot the system. You can freely change and use the port.



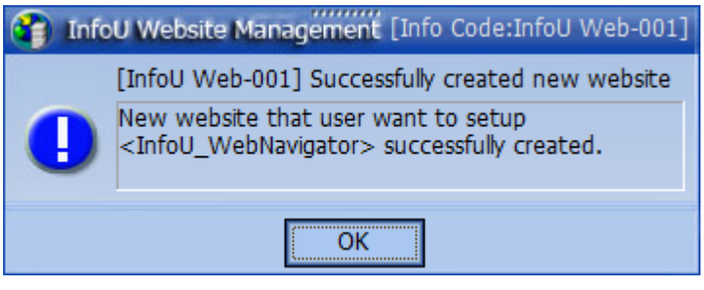
(4) To create the new website, click on [Create] button.



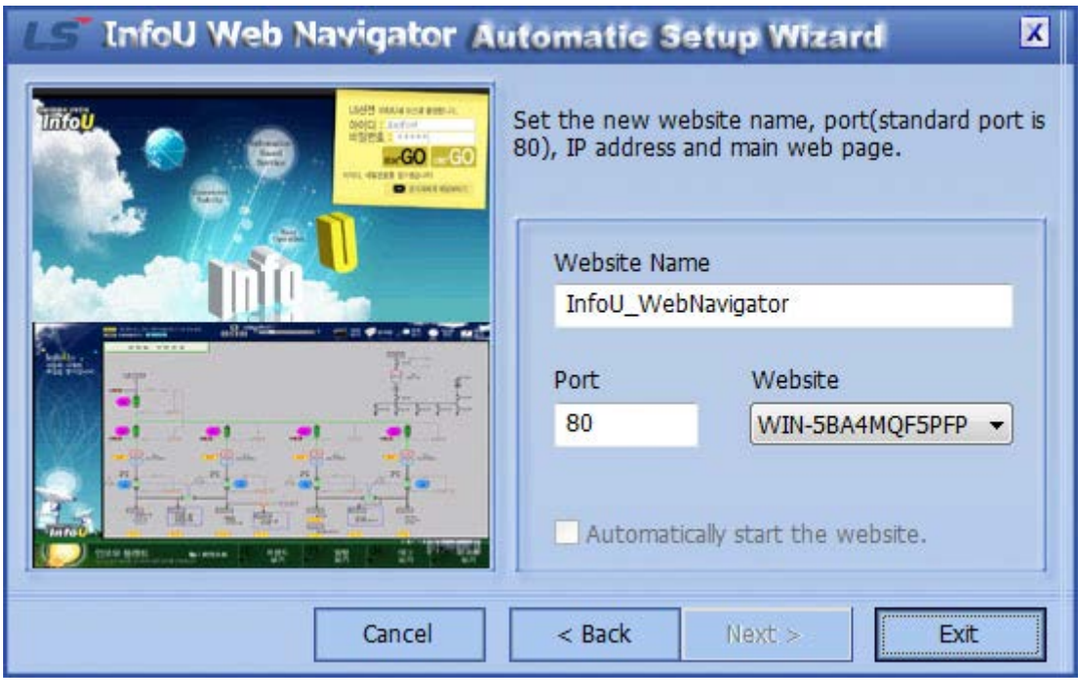
(5) The progress of creating the new website is shown on the program bar.



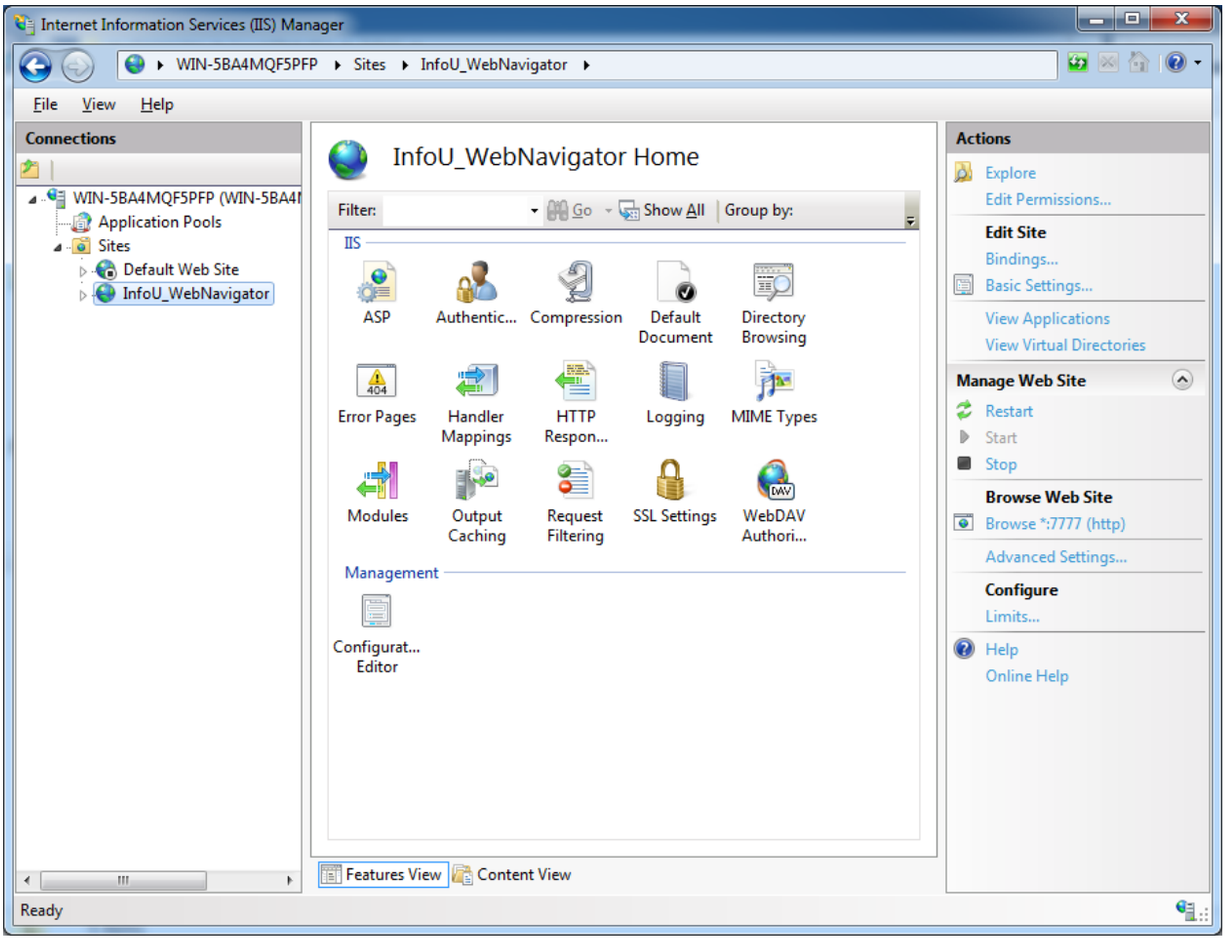
(6) The message of [Successfully created new website] is displayed.



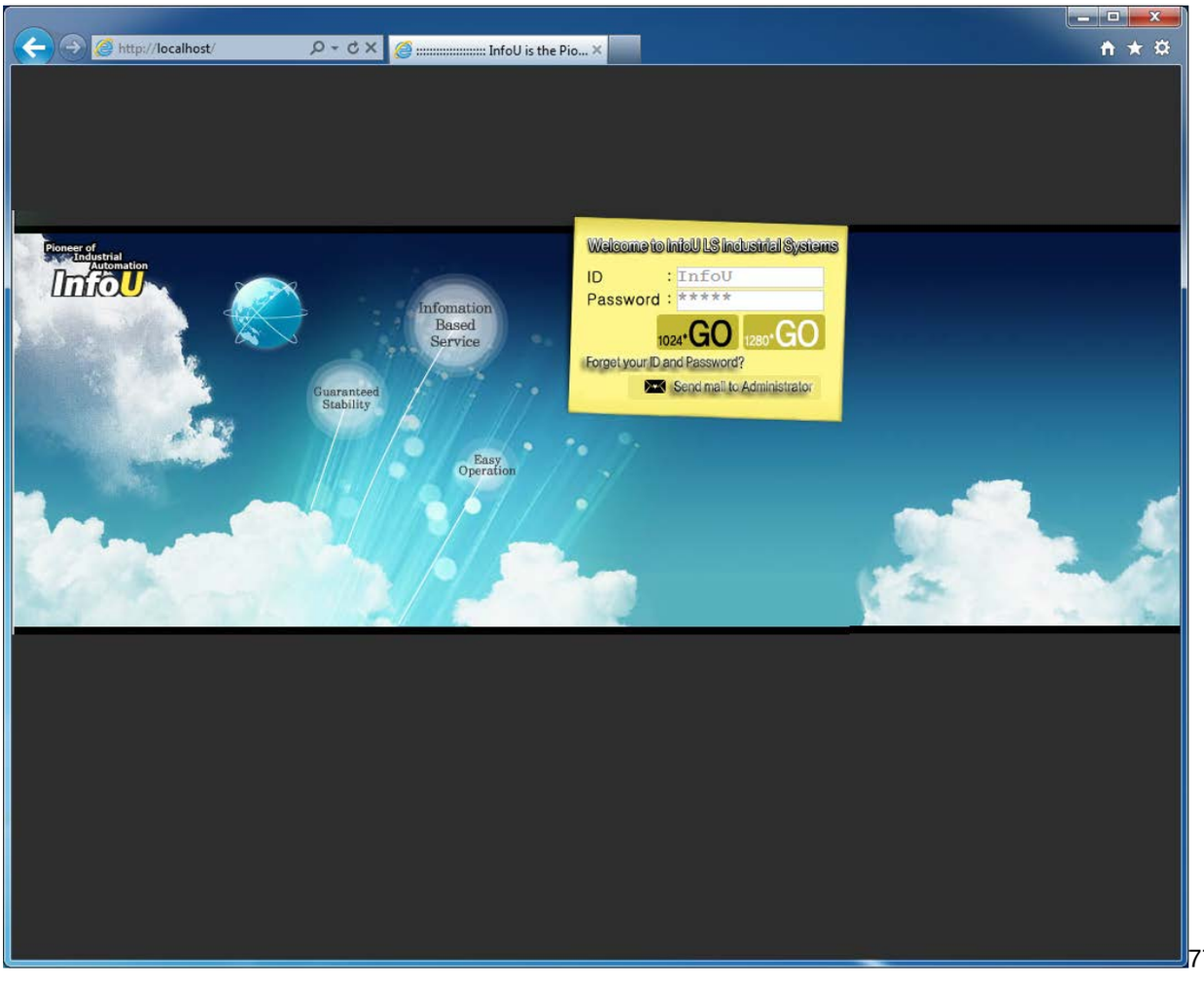
(7) After creating the new web site, click the [Exit] button.



(8) When you check the information of [IIS(Internet Information Service) Administrator] in [Start] ➔ [Control Panel] ➔ [System and Security] ➔ [Administrative Tools], you will see the new web site InfoU_WebNavigator is created. Currently, the default website is suspended and the InfoU_WebNavigator is running.



(9) After creating a new web site, you need to access to ‘http://localhost’ or ‘http://user ip’. If you use a different port other than the default one, 80 port, you need to enter the using port into the address.
Ex.) http://user ip:8080

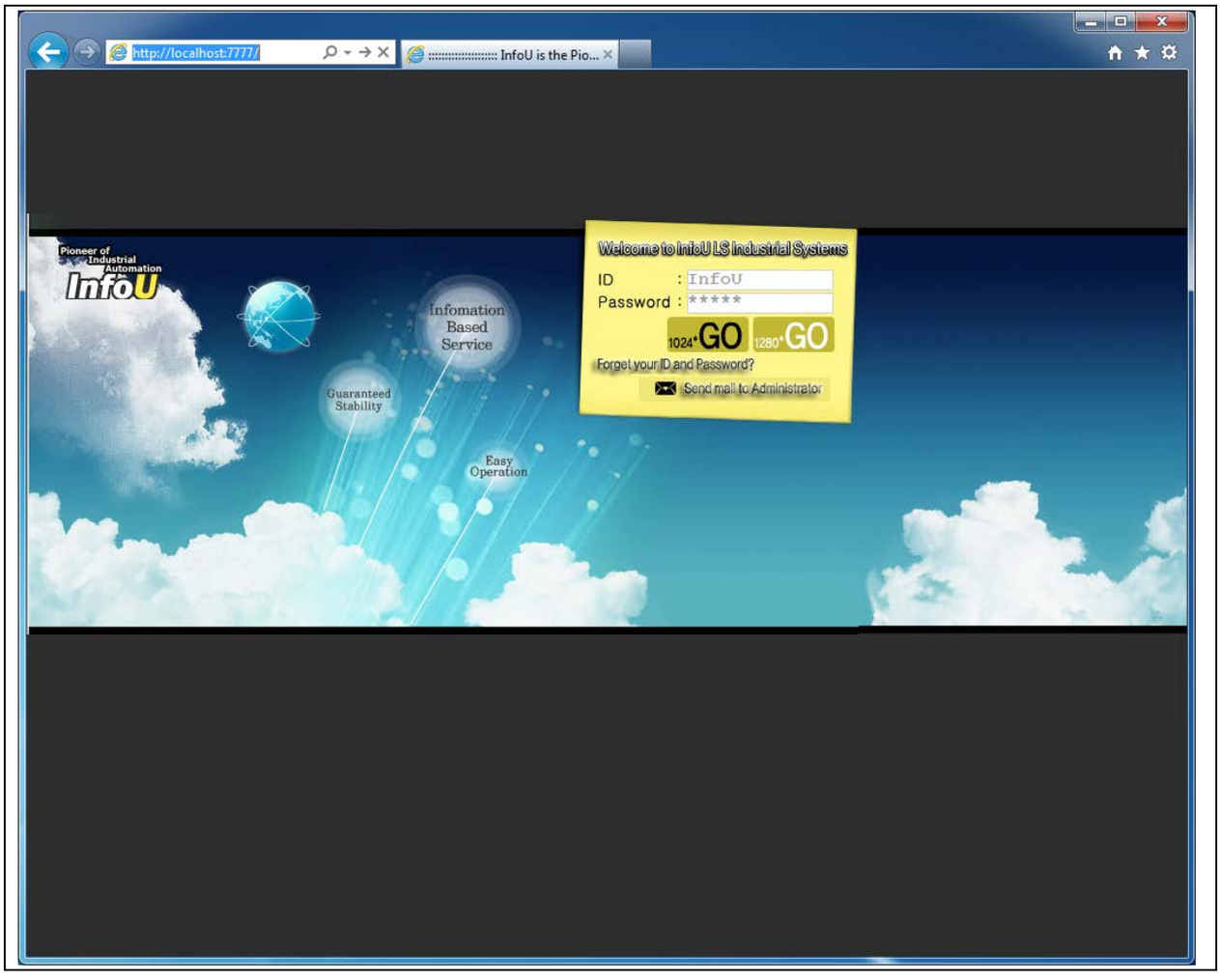


777

Notice

☞When creating a new web site, if you enter a different port other than the default port(80), you need to access to the web address including the entered port.
Ex.) http://localhost:7777

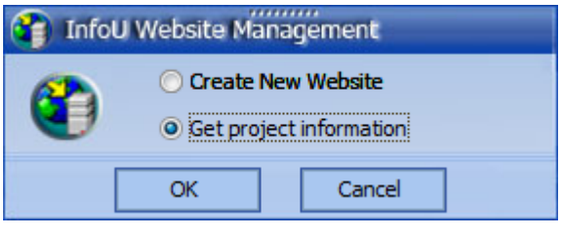
☞In Windows 7 64Bit OS, the default port is set as 7777 so if you want to apply another port, you need to enter the desired port when creating a new website.


http://localhost:7777/
InfoU is the Pio...
Pioneer of Industrial Automation
InfoU
Infomation Based Service
Guaranteed Stability
Easy Operation
Welcome to InfoU LS Industrial Systems
ID : InfoU
Password : *****
1024*GO
1280*GO
Forget your ID and Password?
Send mail to Administrator

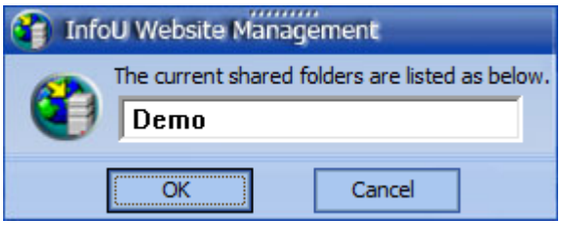
23.2.3 Get Project Information

You can get the project information such as graphics and report file, etc. by selecting the project of the InfoU HMI server. To get the project information, the InfoU HMI server should be preinstalled. Then, you can get the graphic.ivd file and Report.xsl, information related cross reference.
If you create a new website, the project information is automatically obtained so you do not need separate works. However, when the graphics or report file is added or changed, you should execute [Get Project Information] to apply the variations to the nfoU Web newly.

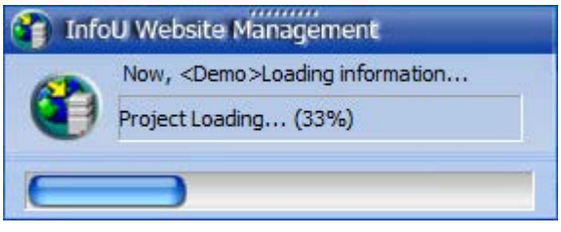
(1) When [Get Project Information] is selected
If the InfoU Web site has already been created, [Get Project Information] is automatically selected.



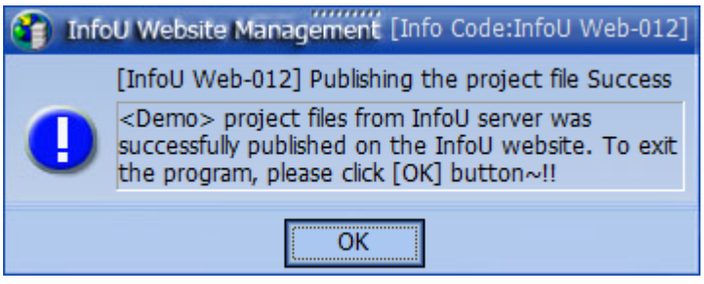
(2) The projet name of the current InfoU server appears and the screen turns to the next step screen upon the click on [OK].



(3) You can see the message showing 'the project information loading is in progress'.



(4) Once the current InfoU server project is successfully distributed to the InfoU Website, the following success message window appears

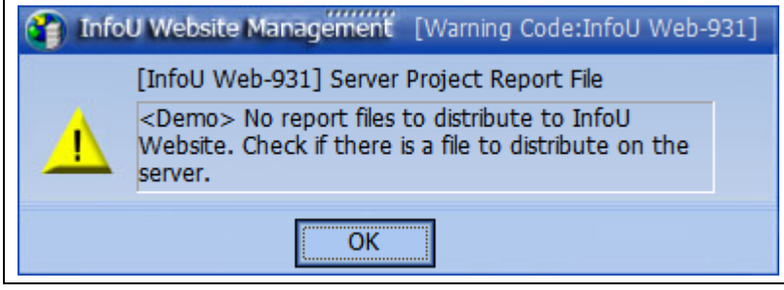


Notice

☞In case the server does not have the report file, you will get the warning message; 'No report files to distribute to InfoU website'. This is just a warning message so distribution for the remaining files except the report file will continue.

☞To prevent the below warning message, you just need to put the prepared report file into the server. The report file is located under the currently running project with the report name.

Ex.) C:\InfoU\Project\Demo\Report



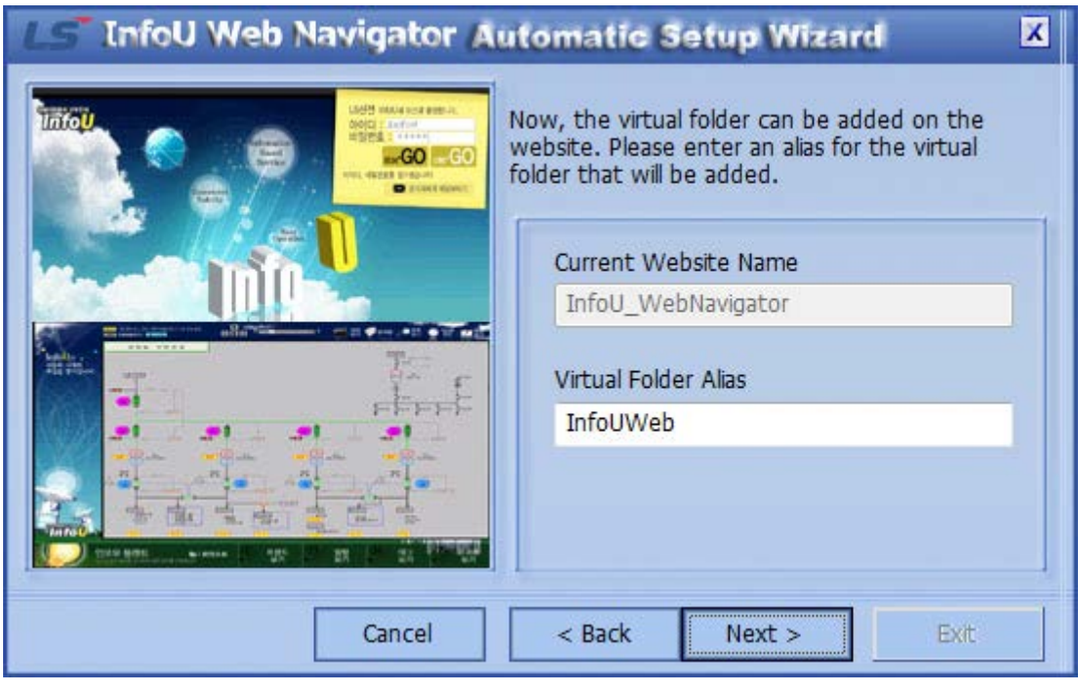
23.2.4 Adding the Virtual Folder on the Existing Website

If it is difficult to add and operate a new website in the current system since the existing site has been already running, you can add the virtual folder under the existing one to manage the InfoU Web site.

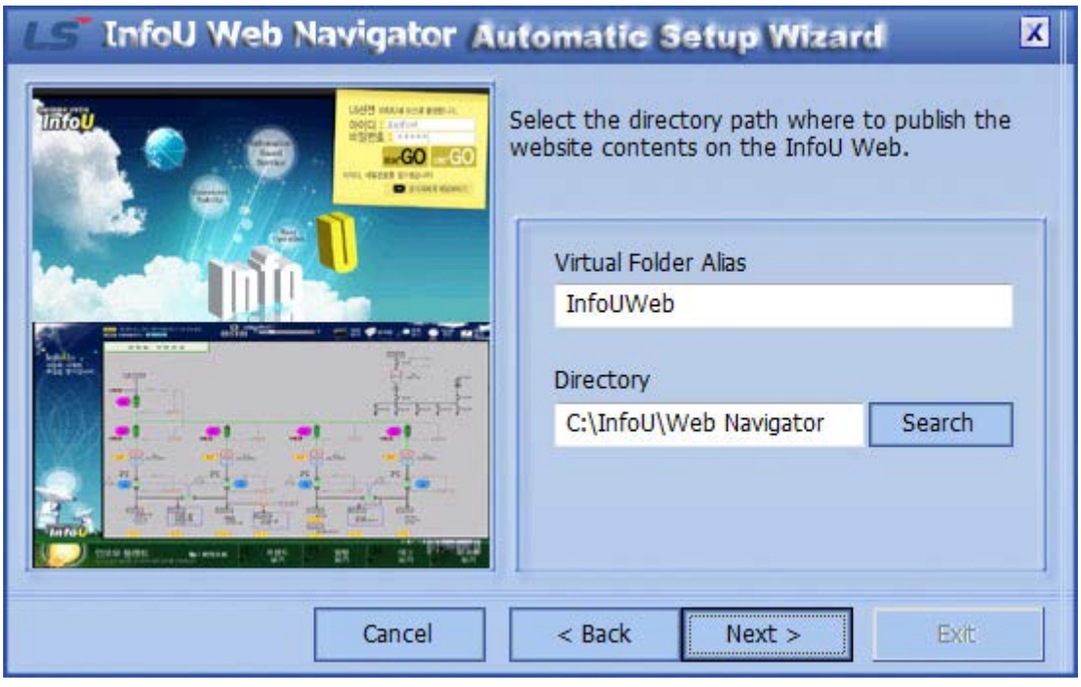
(1) Select 'Adding virtual folder on existing website' in the below screen and press the [Next] button.



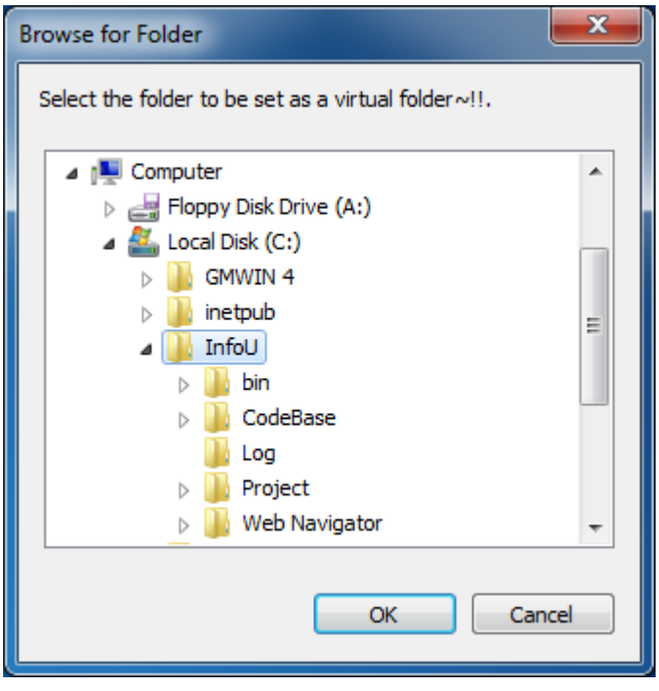
(2) You can enter the desired alias for the virtual folder that will be added. Then, the virtual folder is created on the site that is currently running. In the InfoU Web, basically, the site is created with InfoU_WebNavigator name and the virtual folder is created on the site. If you want to change the default website, refer to [InfoU Web Q&A Collection] of the Appendix.



(3) Set the directory that will be added to the virtual folder. The default directory is set as Web Navigator under the directory where the InfoU system is installed. Ex.) C:\InfoU\Web Navigator

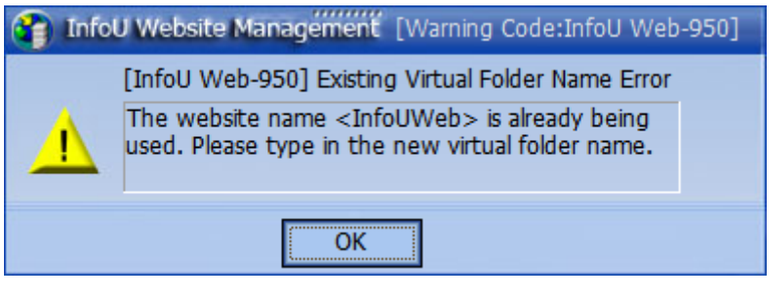


Click the [Search] button to select the directory that you want to specify as the virtual folder. The default setting is the InfoU Web folder so you can proceed with this without any changes. After setting the virtual folder, click the [Next] button.

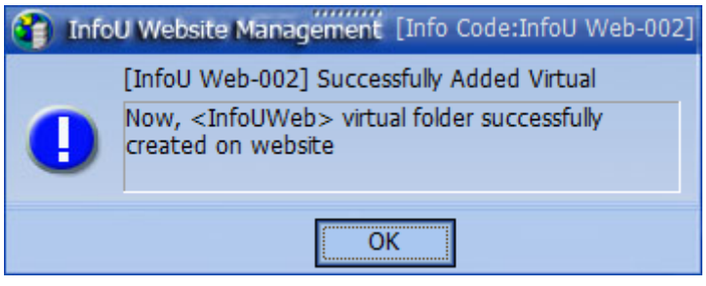


(4) Then, you will get the message; the virtual folder is successfully added to the web server.

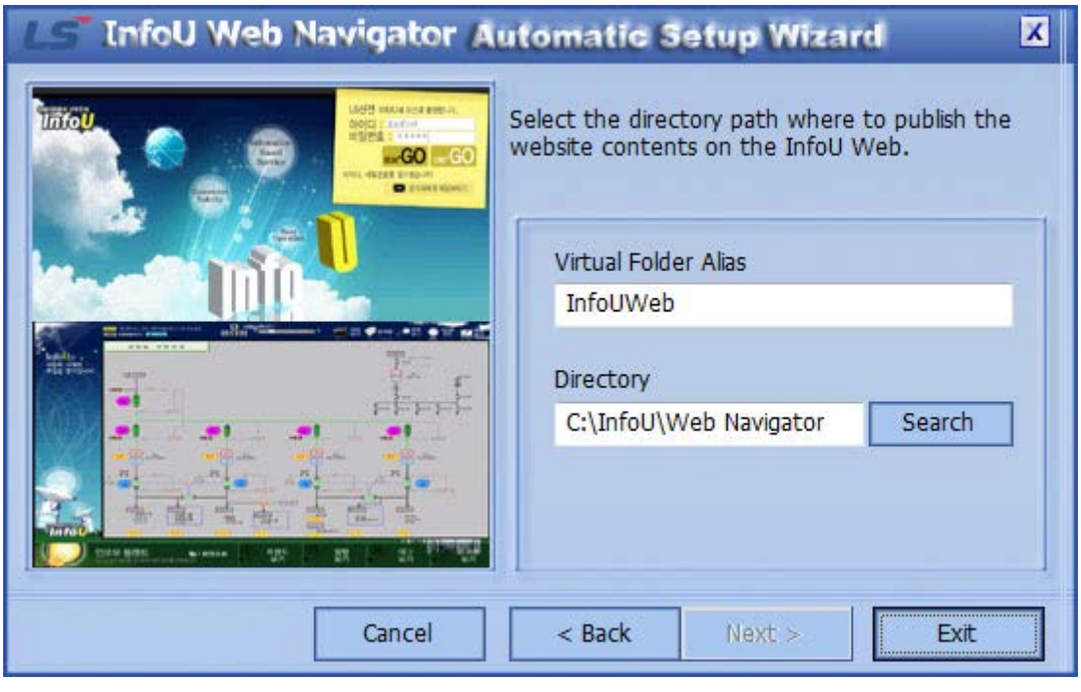
1) If the virtual directory with the same name is already being used.



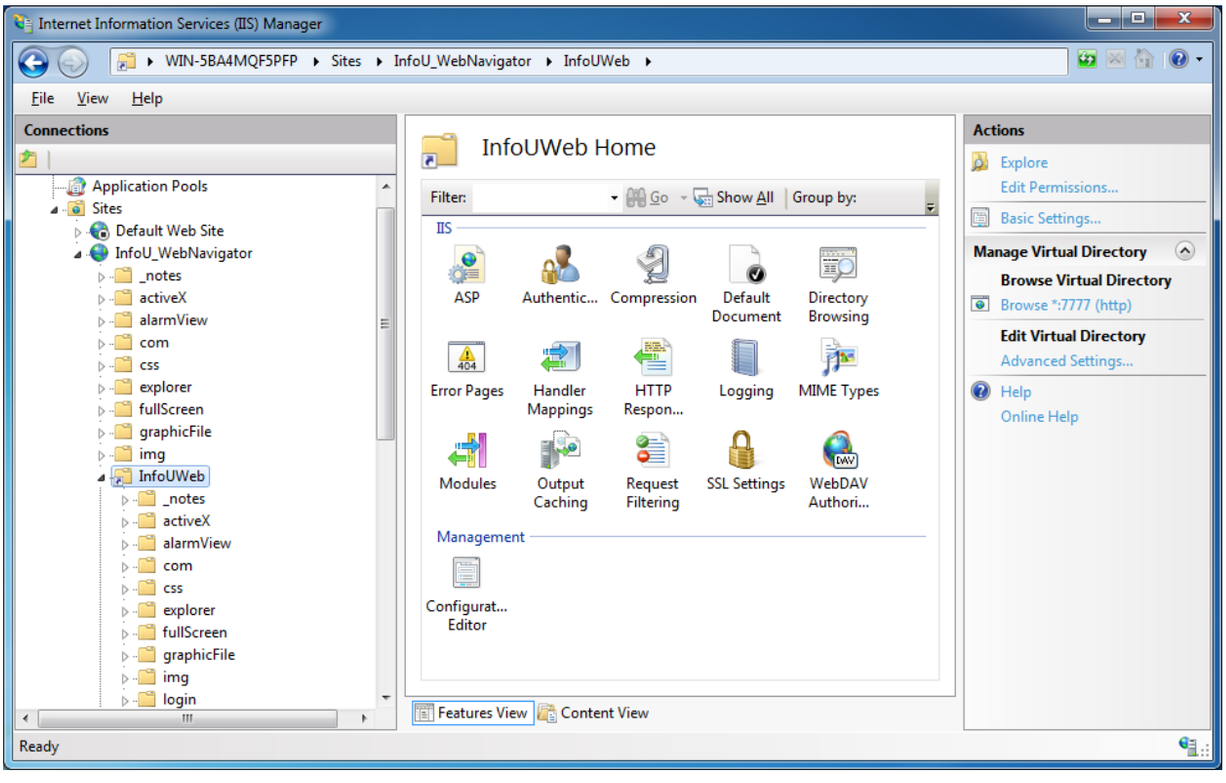
2) If the virtual directory is successfully created on the website.



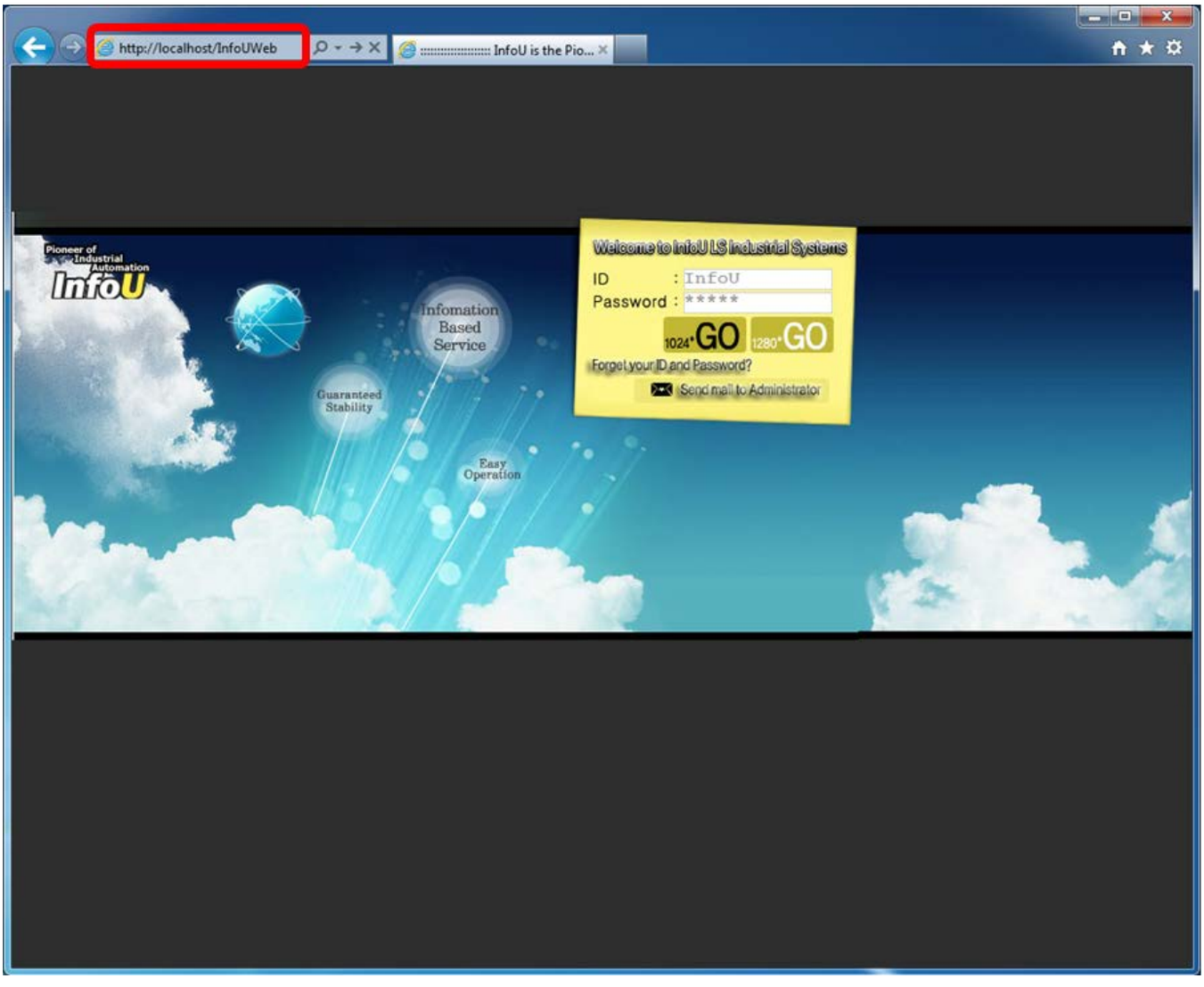
(5) Click the [Exit] button to end the program.



(6) If you check the information in [Start] ➔ [Control Panel] ➔ [System and Security] ➔ [Administrative Tools] ➔ [IIS(Internet Information Service) administrator], you will see the InfoUWeb that is a virtual folder is newly created on the existing web site.

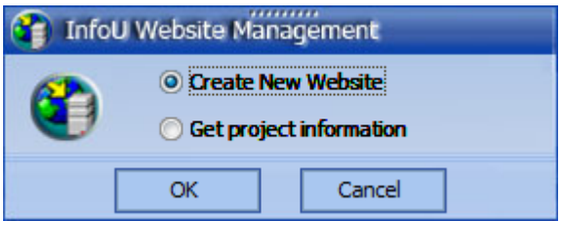


(7) To access to the site, you need to connect to ‘http://user ip: port/virtual folder’ name. If you use a different port other than the default one, port 80, you need to enter the address including the using port.
Ex.) http://localhost/InfoUWeb

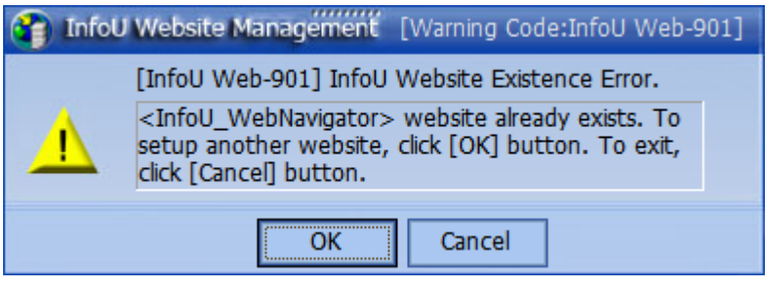


23.2.5 Creation of a Website in Addition to the Existing One

(1) After selecting [Create New Website], click the [OK] button.



(2) If you installed the InfoU Web site previously and the InfoU default web site already exists, the below message will show up.

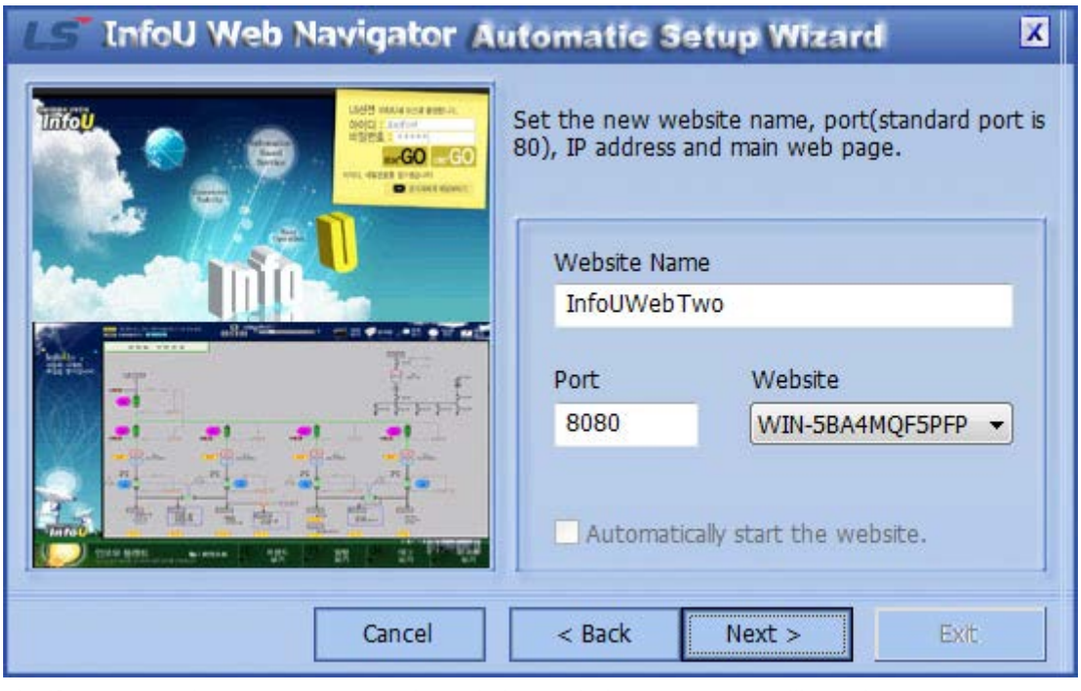


To create a new website additionally besides the existing website, click the [OK] button. If you want to exit the program without creating a new one, just click the [Cancel] button.

(3) After selecting [Create New Website], click the [Next] button.



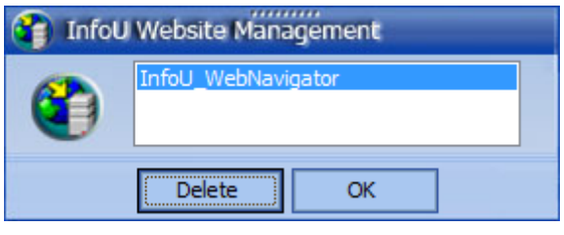
(4) Enter the site name and port No. that the InfoU Web user wants. After entering the desired name of the new website, click [Next] button. The default access port is set as port 80.
If another web server is operated by 80 access port, a user can set other port No. depending on the environment.



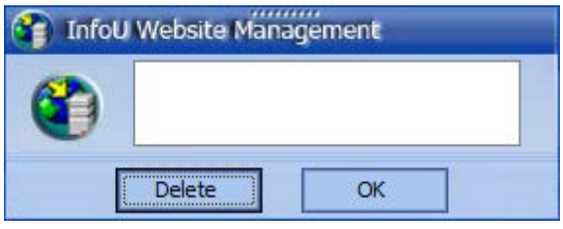
1) If you enter the same name as the existing website name, you will get the warning message as shown below.



2) Select the existing website and click the [Delete] button. Then, press the [OK] button.



3) After deleting the existing website, click the [OK] button.

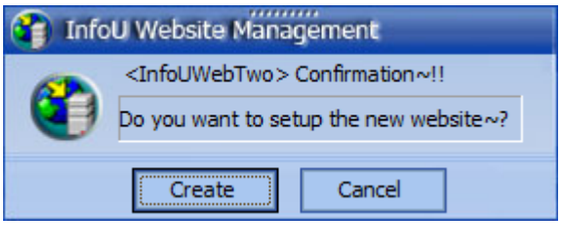


4) If you want to use the entered default information, just click the [Next] button. To change the website name and port newly, after entering the desired name and port , click the [Next] button.

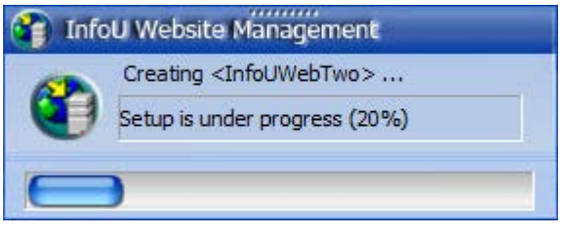




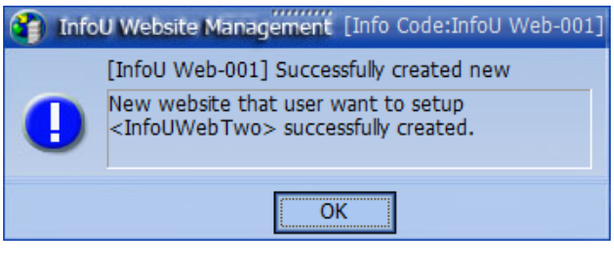
(5) Click the [Create New Website] button.



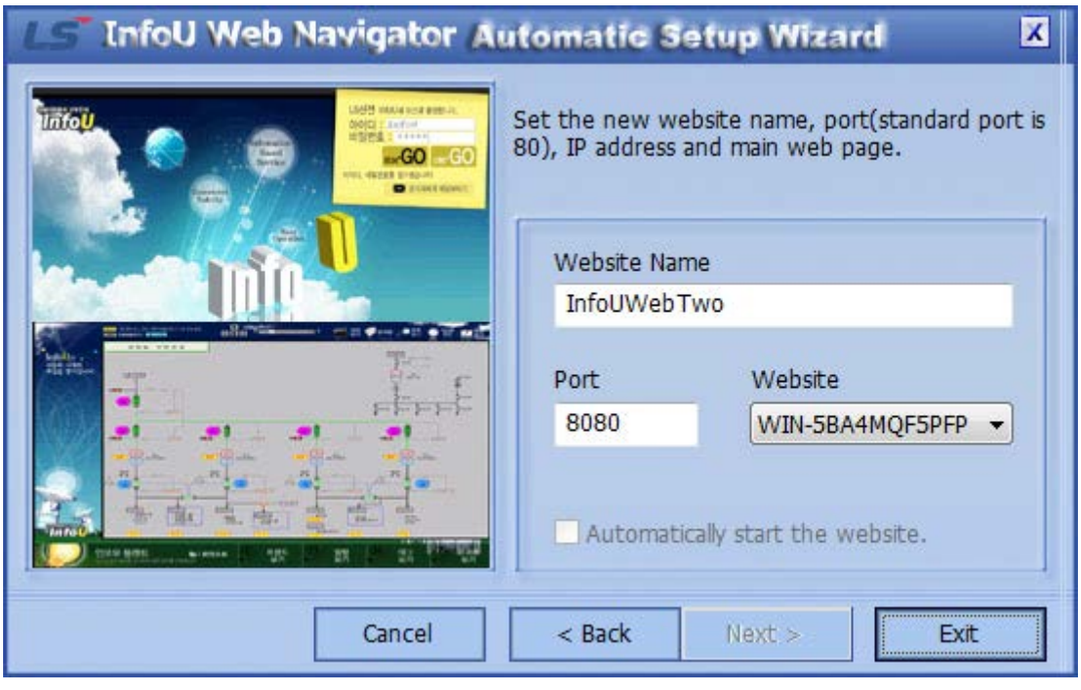
(6) You can get the message; creating a new website is under progress.



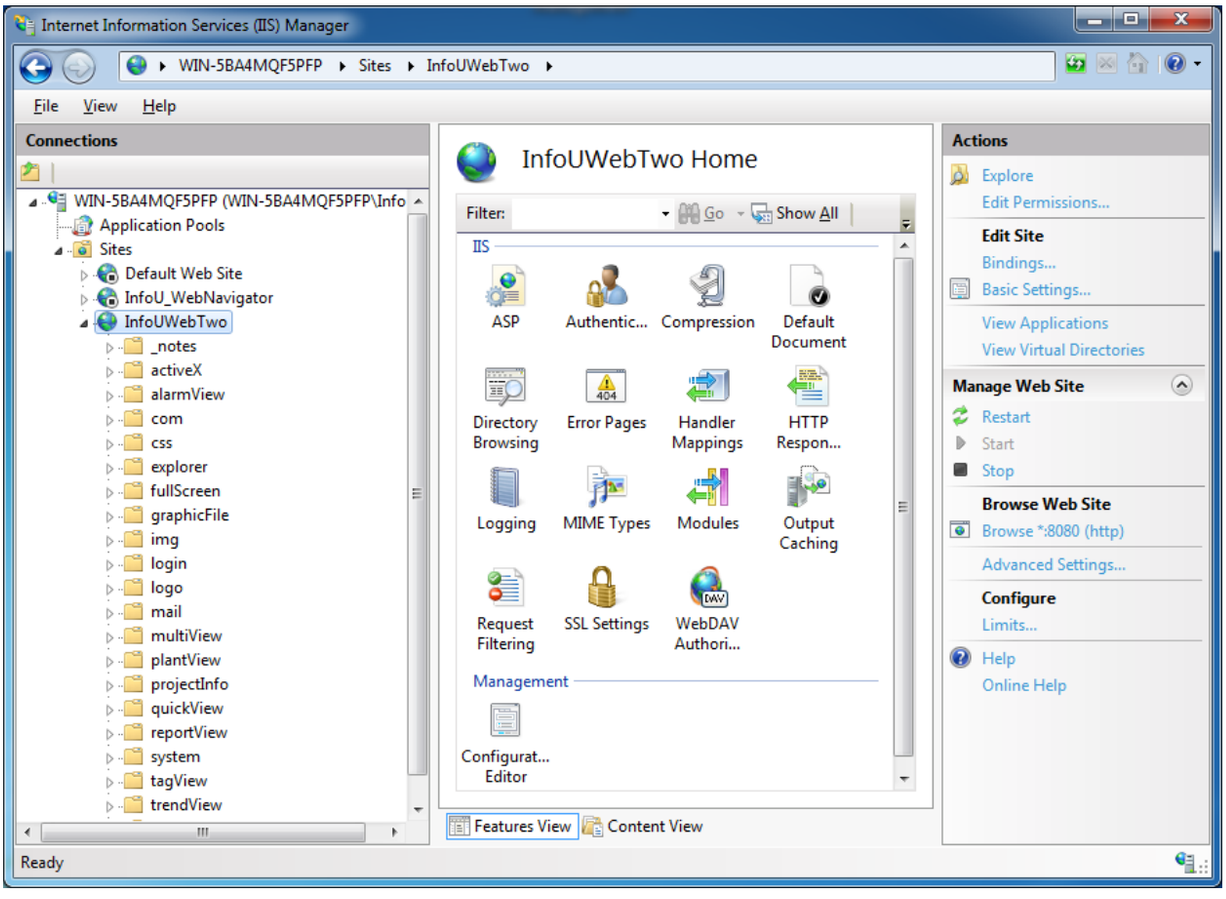
(7) Then, the below message will show up; [New website is successfully created]



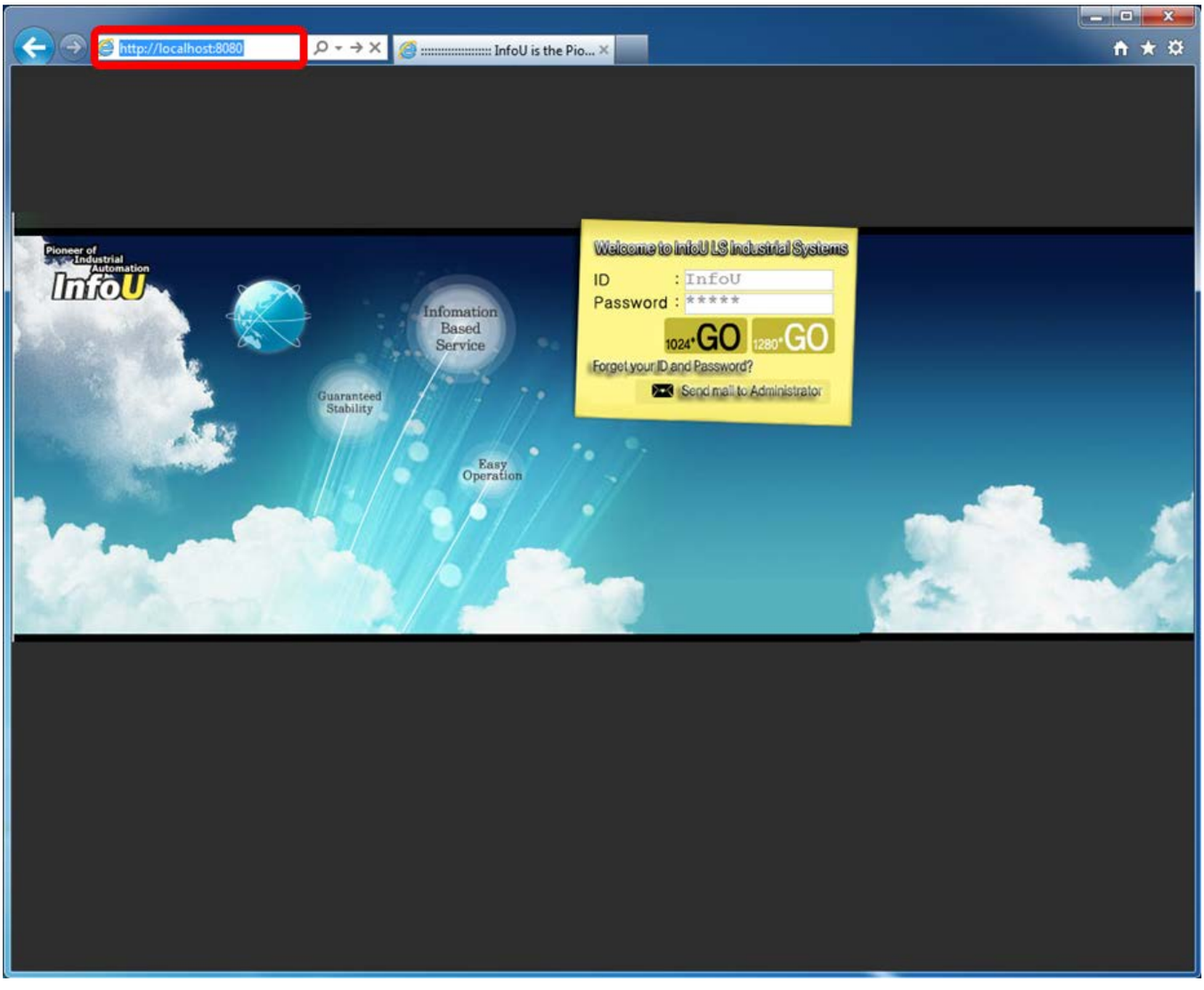
(8) After creating a new website, click the [Exit] button.



(9) If you check the information in [Start] ➔ [Control Panel] ➔ [System and Security] ➔ [Administrative Tools] ➔ [IIS (Internet Information Service) administrator], you will see the new website InfoUWebTwo is additionally created. If you add the website newly, the previous website will stop and instead, the new one will be operated. You can also operate several sites with different ports.



(10) After creating a new web site, you need to access to ‘http://localhost’ or ‘http://user ip’. If you use a different port other than the default one, 80 port, you need to enter the using port into the address.
Ex.) http://user ip:8080



23.3 InfoU Website Configuration

23.3.1 Overview

To use the web-based HMI function of the InfoU, you need to register the user account and the default screen.

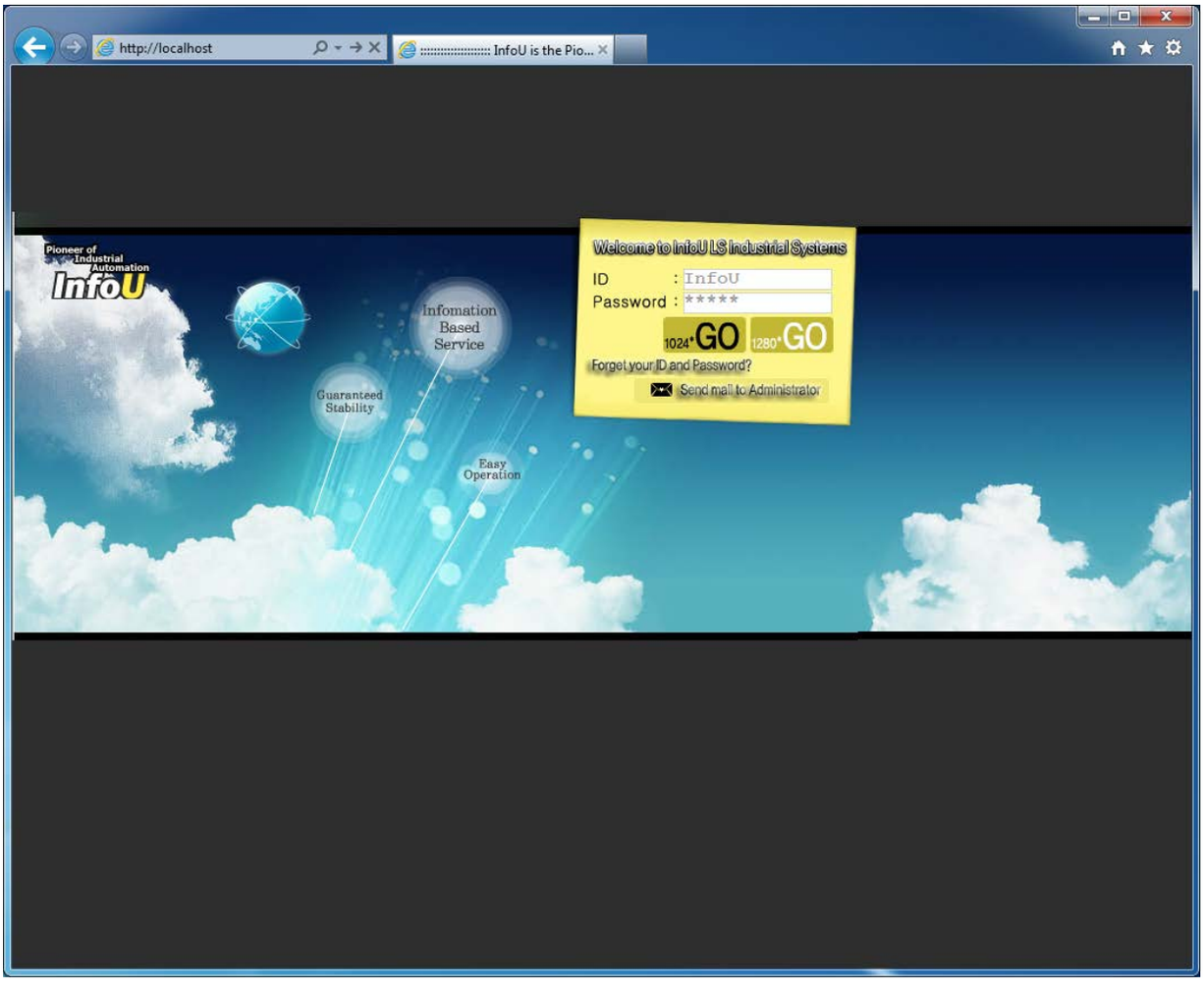
23.3.2 Prerequisite

(1) The InfoU web site should be created through [Navigation Management] and the web server should be running.
(2) InfoU real-time operation should be executed.
(3) The InfoU WebHMI server is running.

23.3.3 Login Screen

(1) Enter the user ip into the web browser address bar. If you did not use the default port “80” when creating the InfoU web site, you need to enter the used port No. together.
Ex.) http://localhost or http://userip:7777

(2) Log in to the InfoU for initial settings. The default password of the InfoU is just “InfoU” and the default administrator account is InfoU/InfoU. They are case-sensitive so must be entered correctly. It is recommended that you use a new administrator account or change the password for further security.

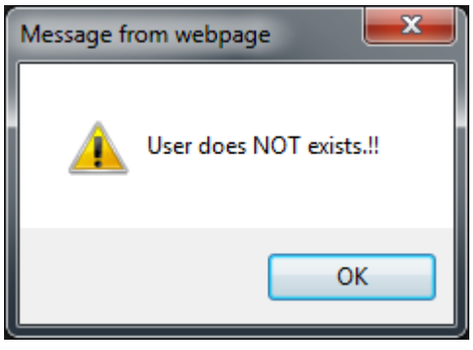


(3) Description of Login Screen Menu

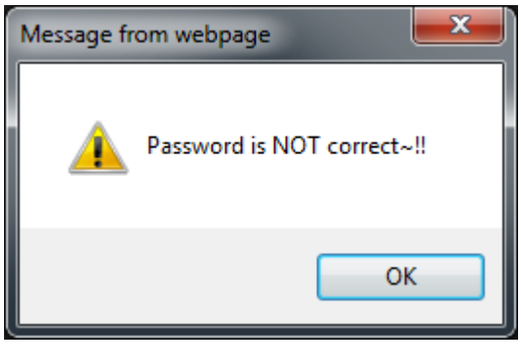
Enter the user ID and password after logging in the InfoU Website on the web browser. If the user is not authorized or the password is not correct, the error message is displayed.

1) Error Message when login fails

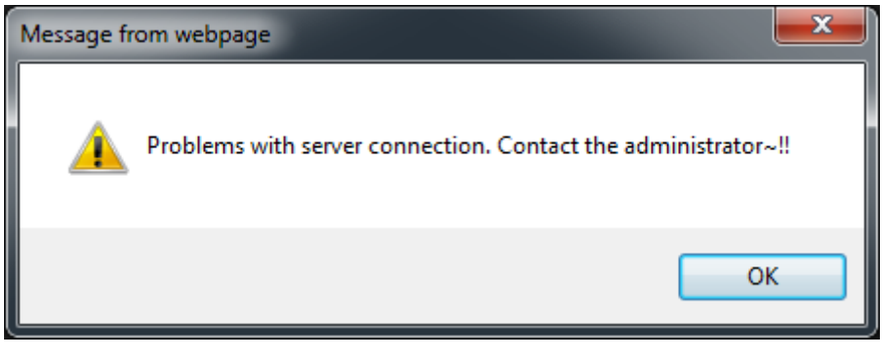
a. If the user ID is not correct



b. If the password is not correct



c. If there are communication problems or other issues for the solutions, refer to appendix 6, InfoU Web Q&A Collection.



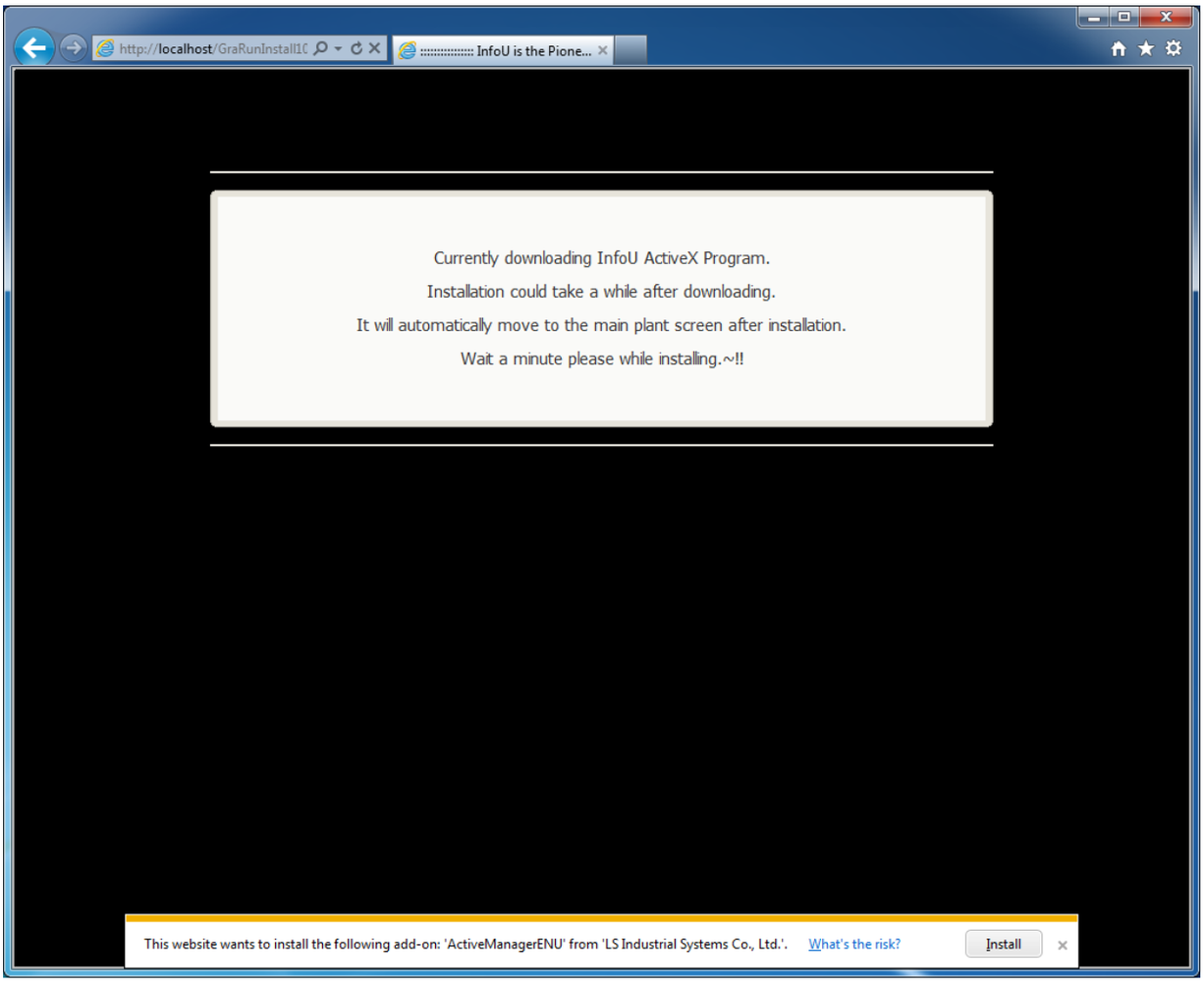
(4) Screen Mode Select

The user may select either 1024*768 or 1280*1024 mode as the optimized mode.

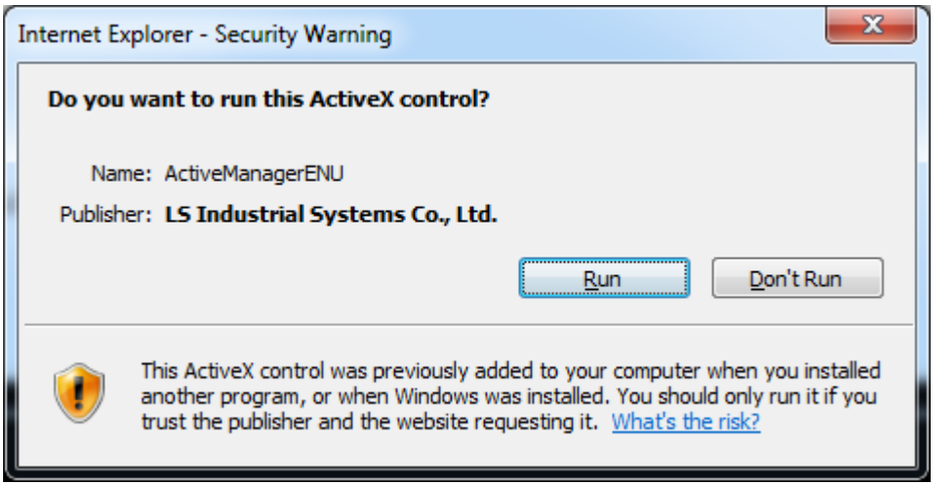
(5) Send Mail to Administrator

If the user ID or password is forgotten, send e-mail to the administrator

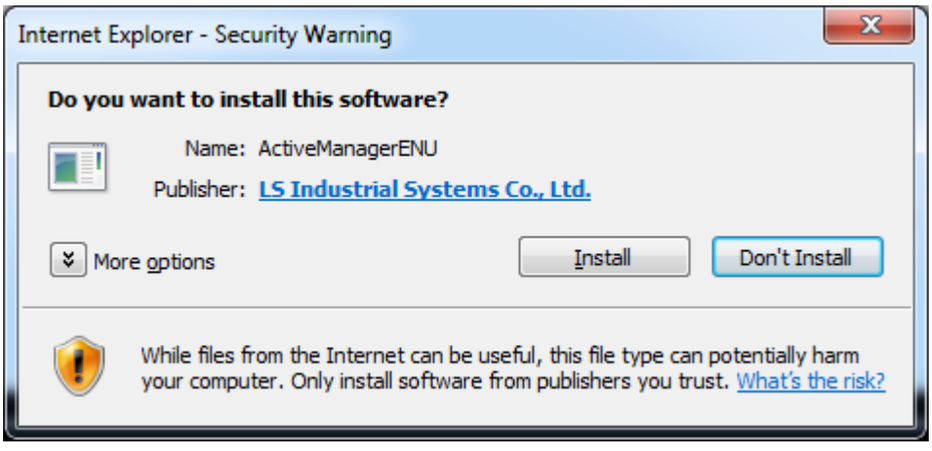
(6) If the message on installing additional functions of ActiveX is the displayed at the bottom of the page after login, press the [Install] button.



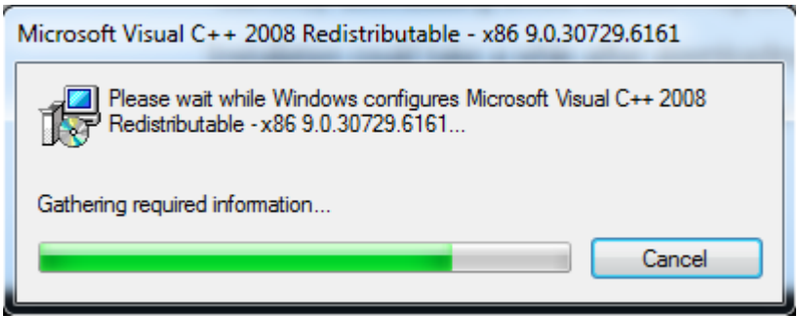
(7) If the security warning message is displayed as below, press the [Run] button.

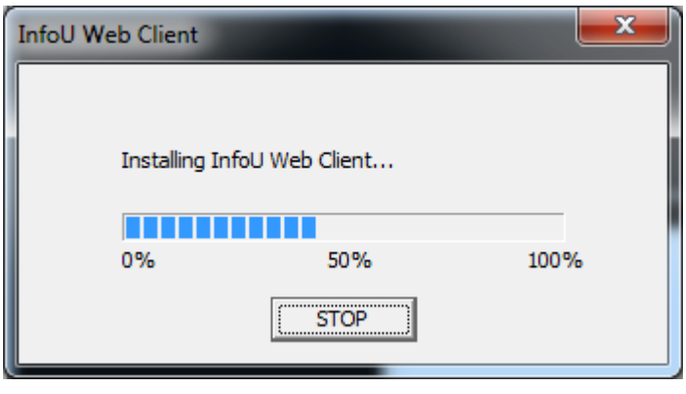


(8) If the security warning message on ActiveX installation is displayed, press the [Install] button.



(9) The components to execute the program are installed.

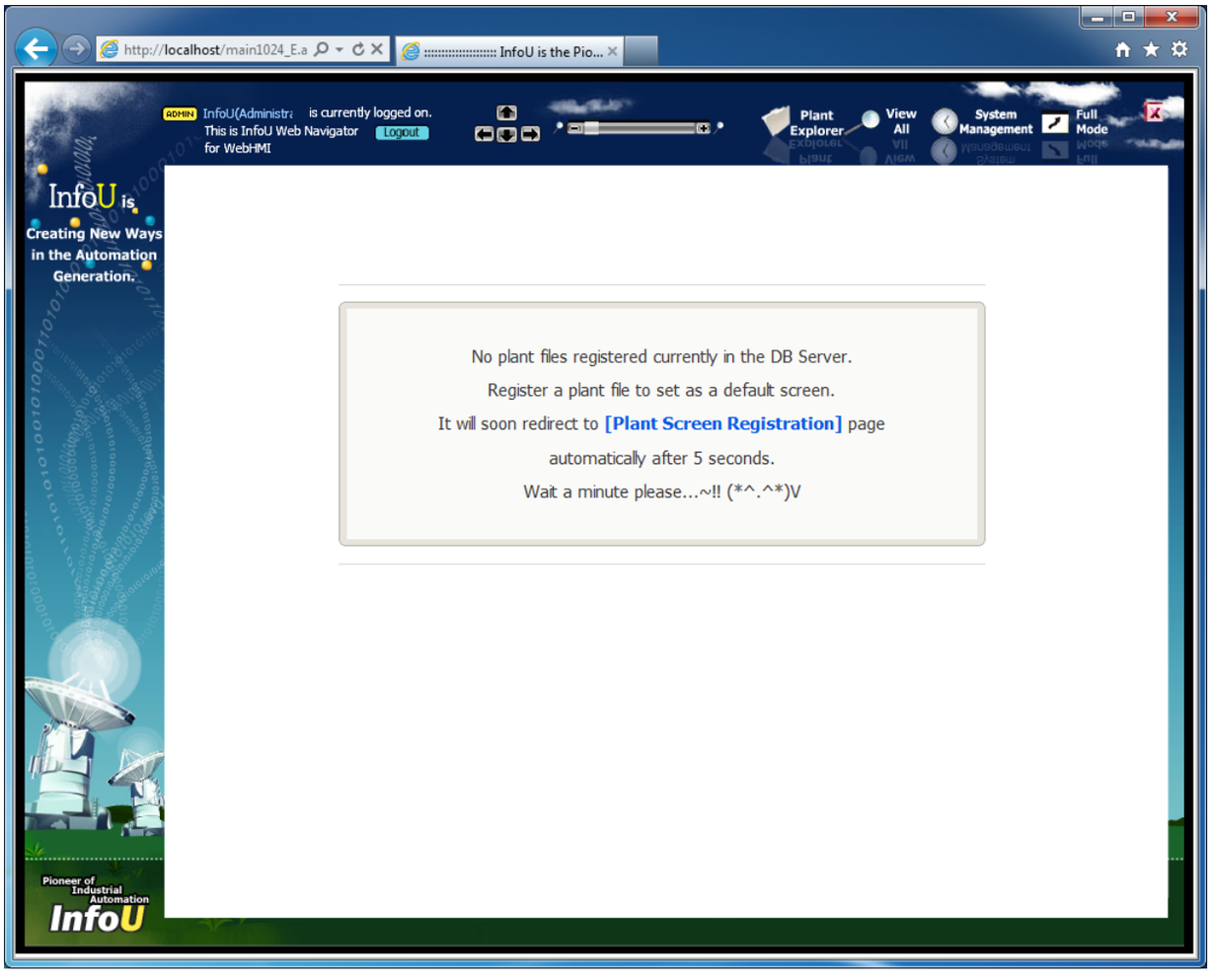


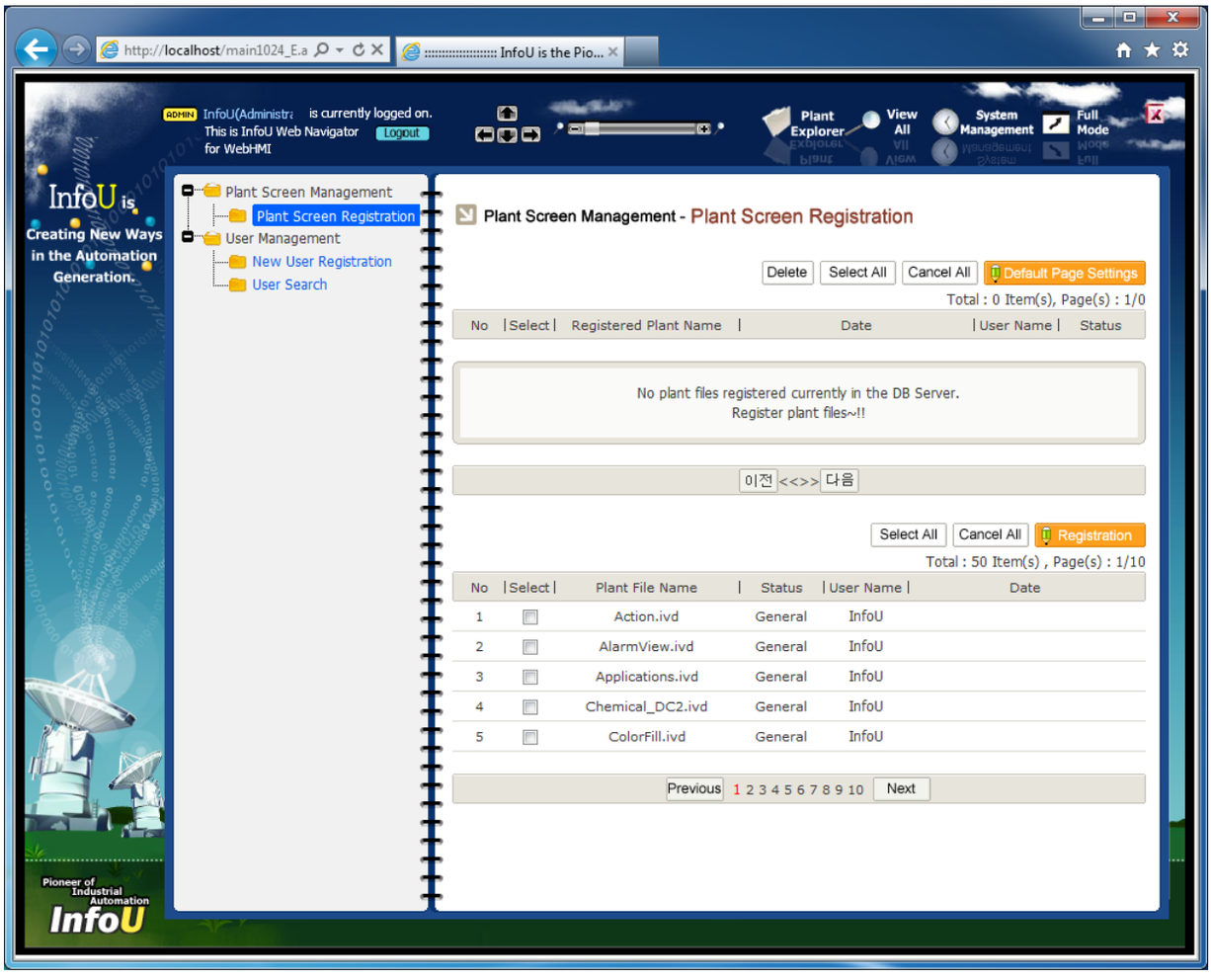


(10) When logging in to the website for the first time after installation.

1) When logging in to the website with a administrator account:

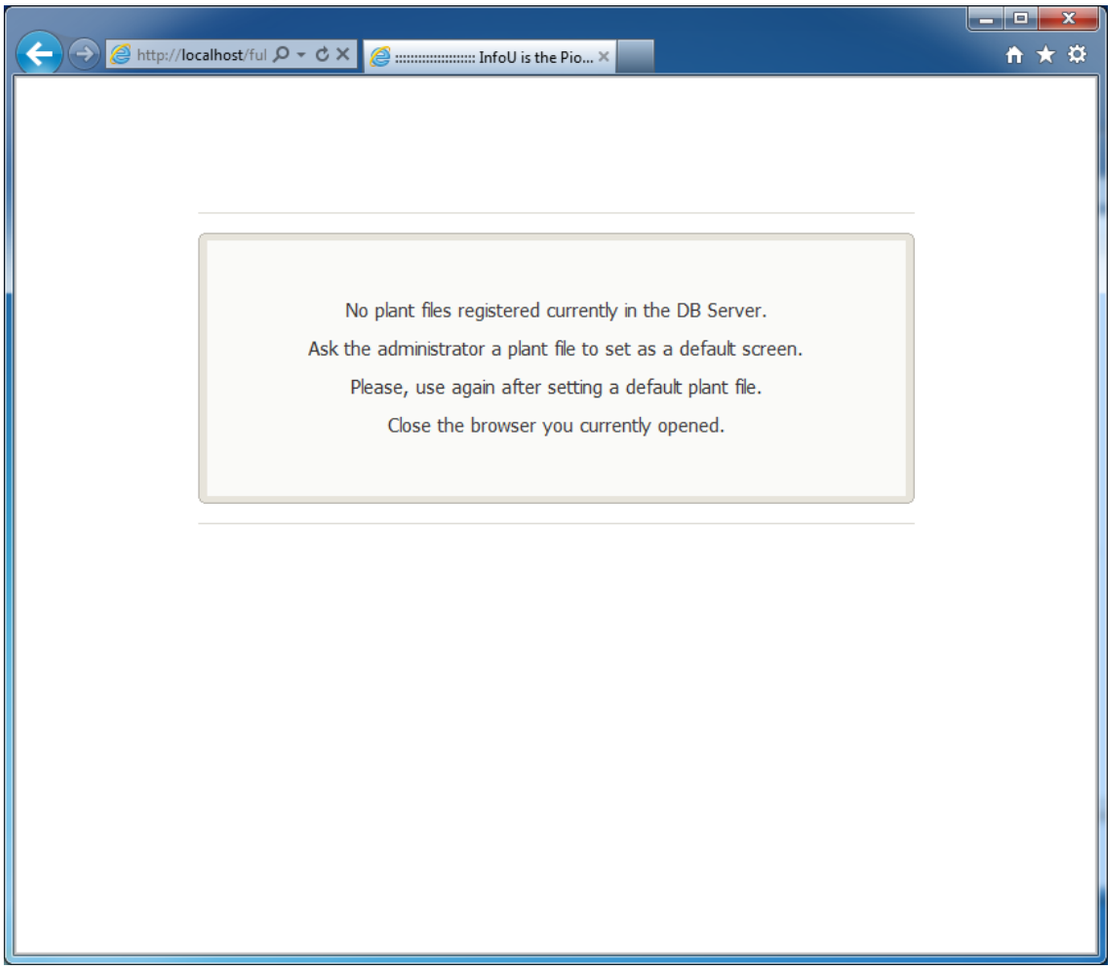
When you log in to the site for the first time, there is no plant screen registered in the server. If there is no plant screen registered in the server, you will get the message; “No plant files registered currently in the DB Server. Register a plant file to set as a default screen. It will soon redirect to [Plant Screen Registration] page automatically after 5 minutes. Wait a minute please.”



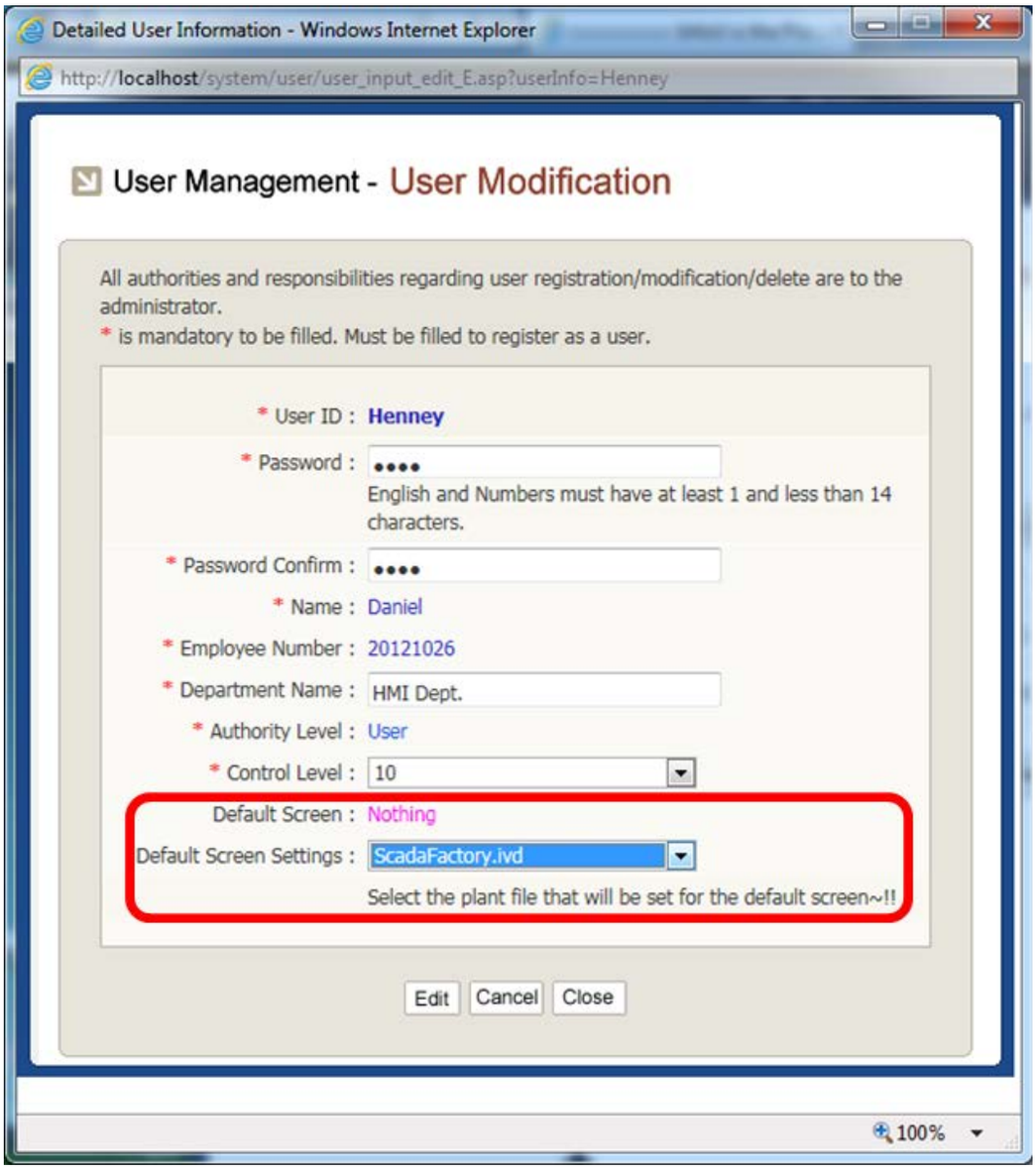

http://localhost/main1024_E.a
InfoU is the Pio...
InfoU(Administra is currently logged on.
This is InfoU Web Navigator for WebHMI
Logout
Plant Explorer
View All
System Management
Full Mode
InfoU is
Creating New Ways in the Automation Generation.
Plant Screen Management
Plant Screen Registration
User Management
New User Registration
User Search
Plant Screen Management - Plant Screen Registration
Delete
Select All
Cancel All
Default Page Settings
Total : 0 Item(s), Page(s) : 1/0
No | Select | Registered Plant Name | Date | User Name | Status
No plant files registered currently in the DB Server.
Register plant files~!!
이전 <<>> 다음
Select All
Cancel All
Registration
Total : 50 Item(s) , Page(s) : 1/10
No | Select | Plant File Name | Status | User Name | Date
1 Action.ivd General InfoU
2 AlarmView.ivd General InfoU
3 Applications.ivd General InfoU
4 Chemical_DC2.ivd General InfoU
5 ColorFill.ivd General InfoU
Previous 1 2 3 4 5 6 7 8 9 10 Next
Pioneer of Industrial Automation
InfoU

2) When logging in to the website with a general user account

a. If you log in to the site with the general user account for the first time, you will get the message; “No plant files registered currently in the DB Server. Ask the administrator a plat file to set as a default screen. Please, use again after setting a default plant file. Close the browser you currently opened.” As shown in the message, you need to request the administrator to register the plant file to set as a default screen, then, log in to the site.

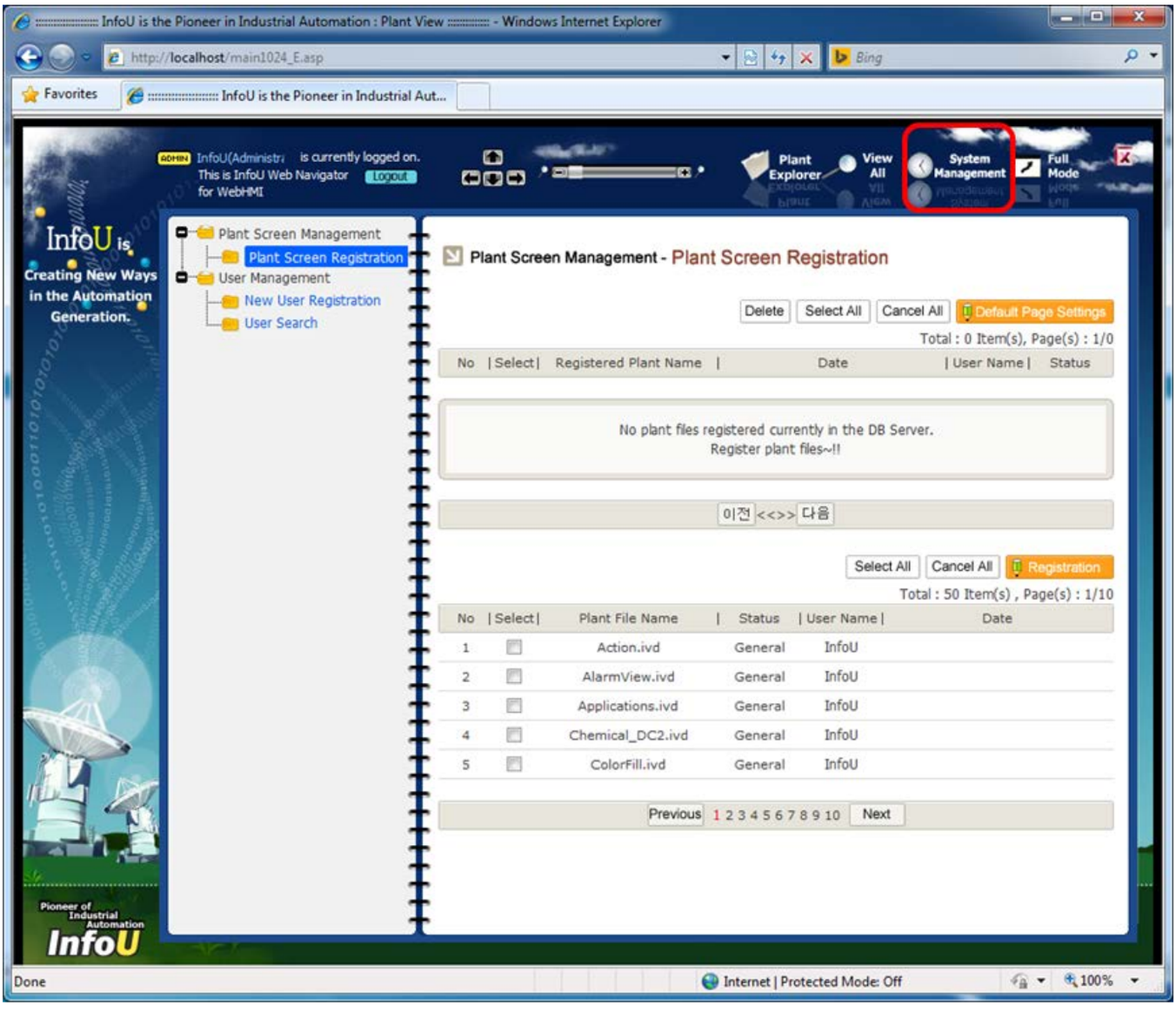


b. After logging in to the site with an administrator account, select the user who will register the default plant screen in [System Management]➔[User Information]. If you click the ID, the user information screen will show up. In the dialog box of [User Management–User Modification], set the default screen. Under the situation of [Default screen: Nothing], click the list box in [Default Screen Settings] and select the plant screen to be set as the default.

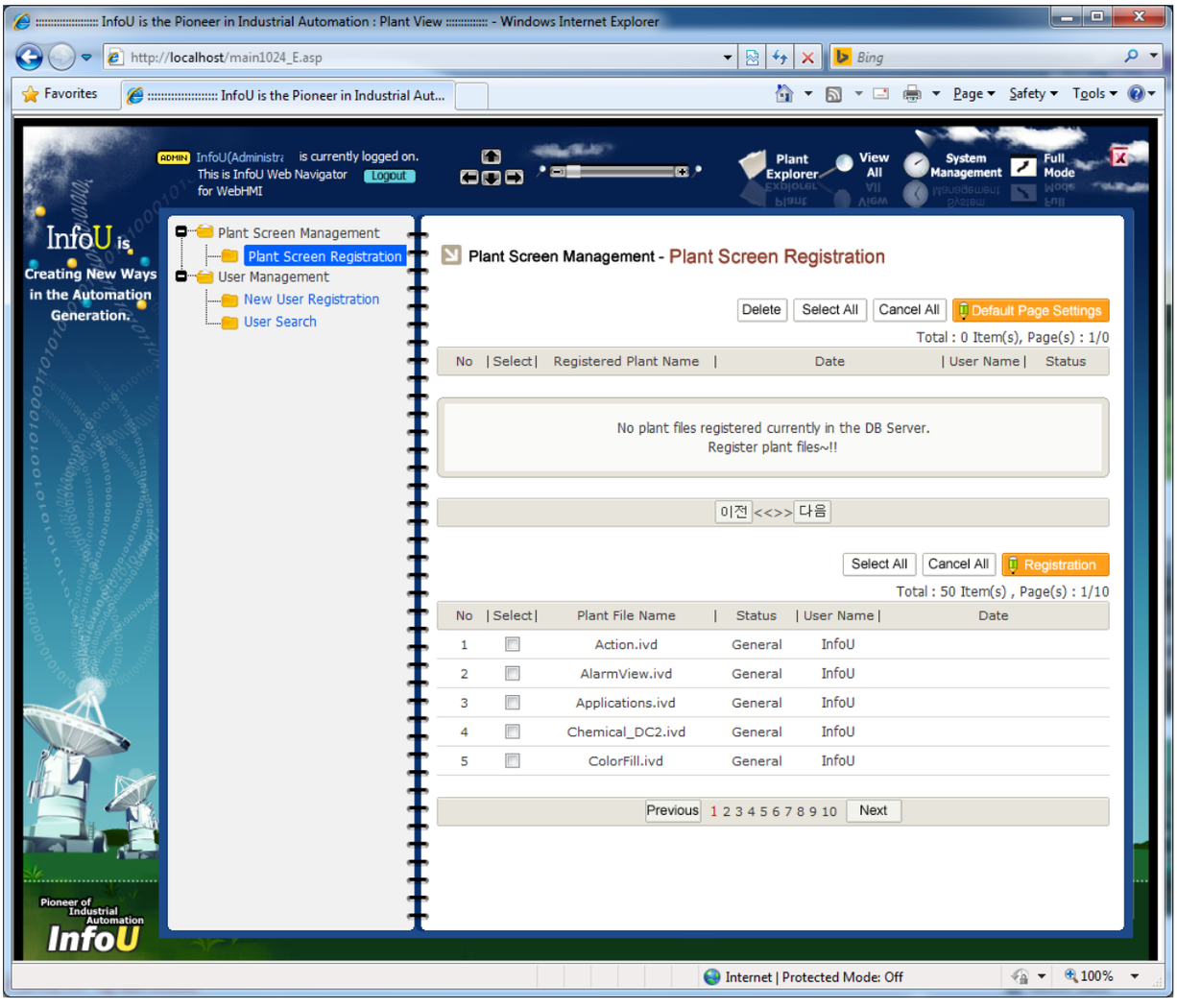


23.3.4 System Management

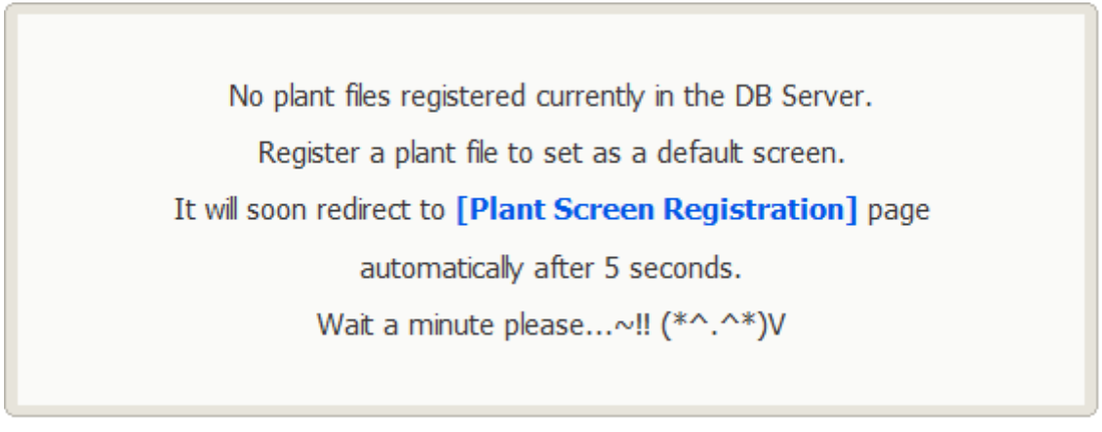
(1) Register the monitoring screen by clicking the [System Management] icon.



(2) If you log in to the sire for the first time, when the ActiveX and components are completely installed, the web setting screen will be displayed as below.

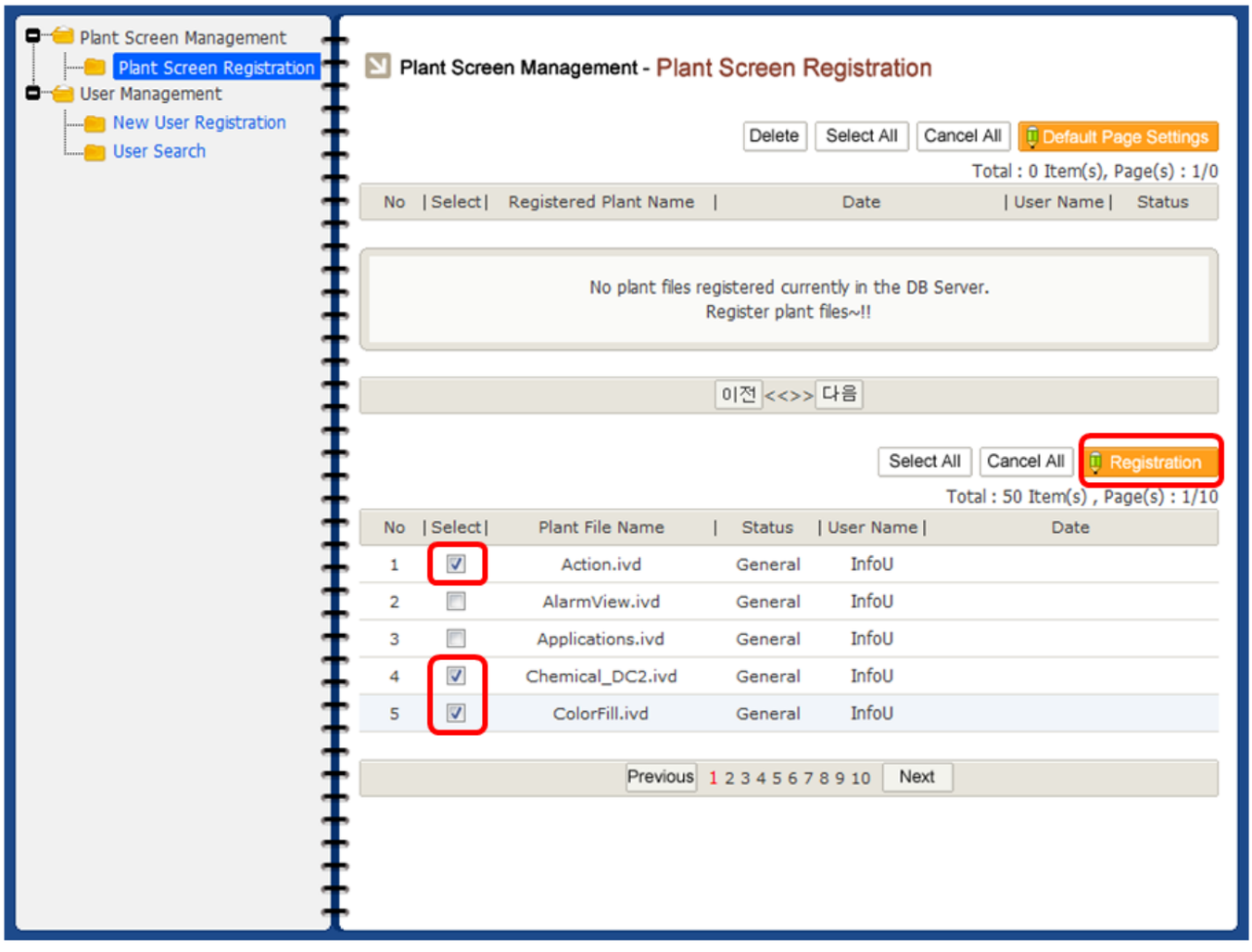


1) If the default screen is not set, the below message will be displayed and it will move to the system setting screen automatically. Otherwise, if you click the [System Management] icon, the screen will show up.

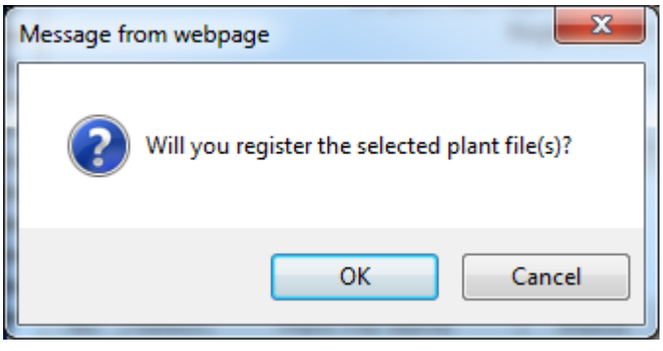


(3) Plant Screen Registration

Select the plant file to be monitored and click the 'Registration' button. You can select and register multiple plant files for monitoring.



If you click the [Registration] button, the message asking whether registering the selected plant file(s) will be displayed.

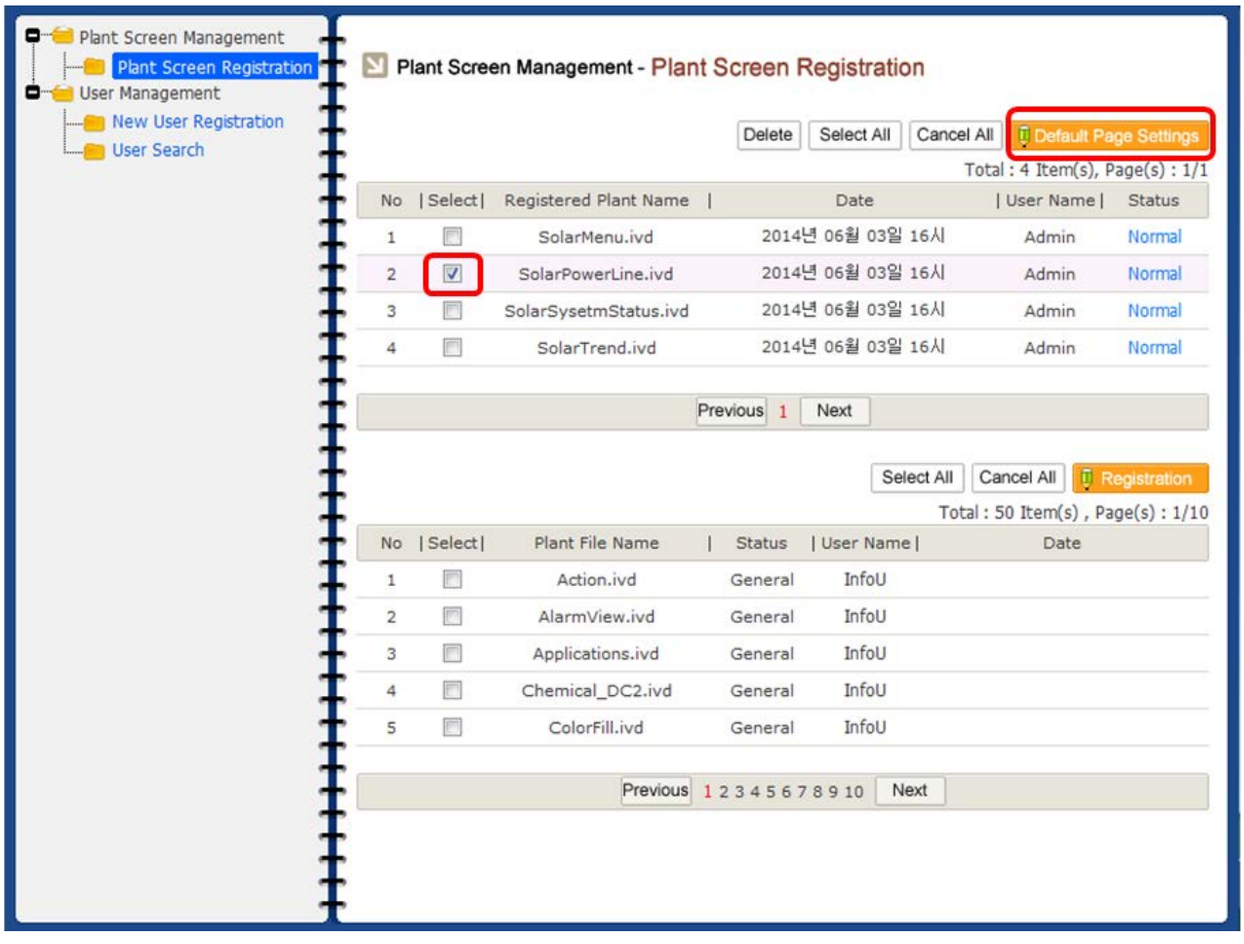


(4) Default Page Settings

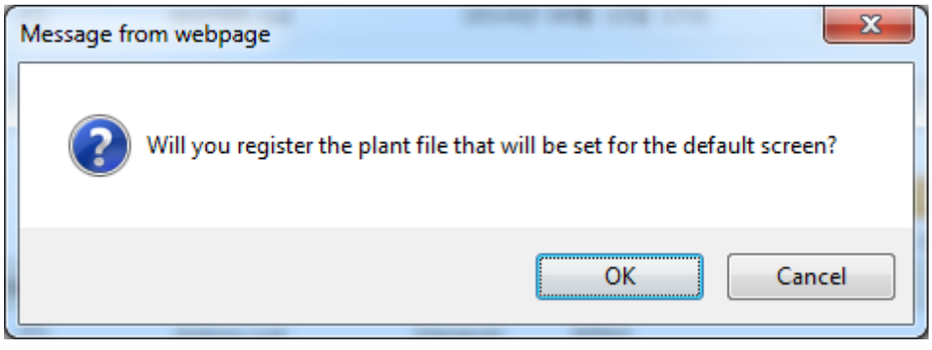
1) You can specify the default screen that you want.

2) After selecting the screen in the upper list, click the [Default Screen Settings] button.

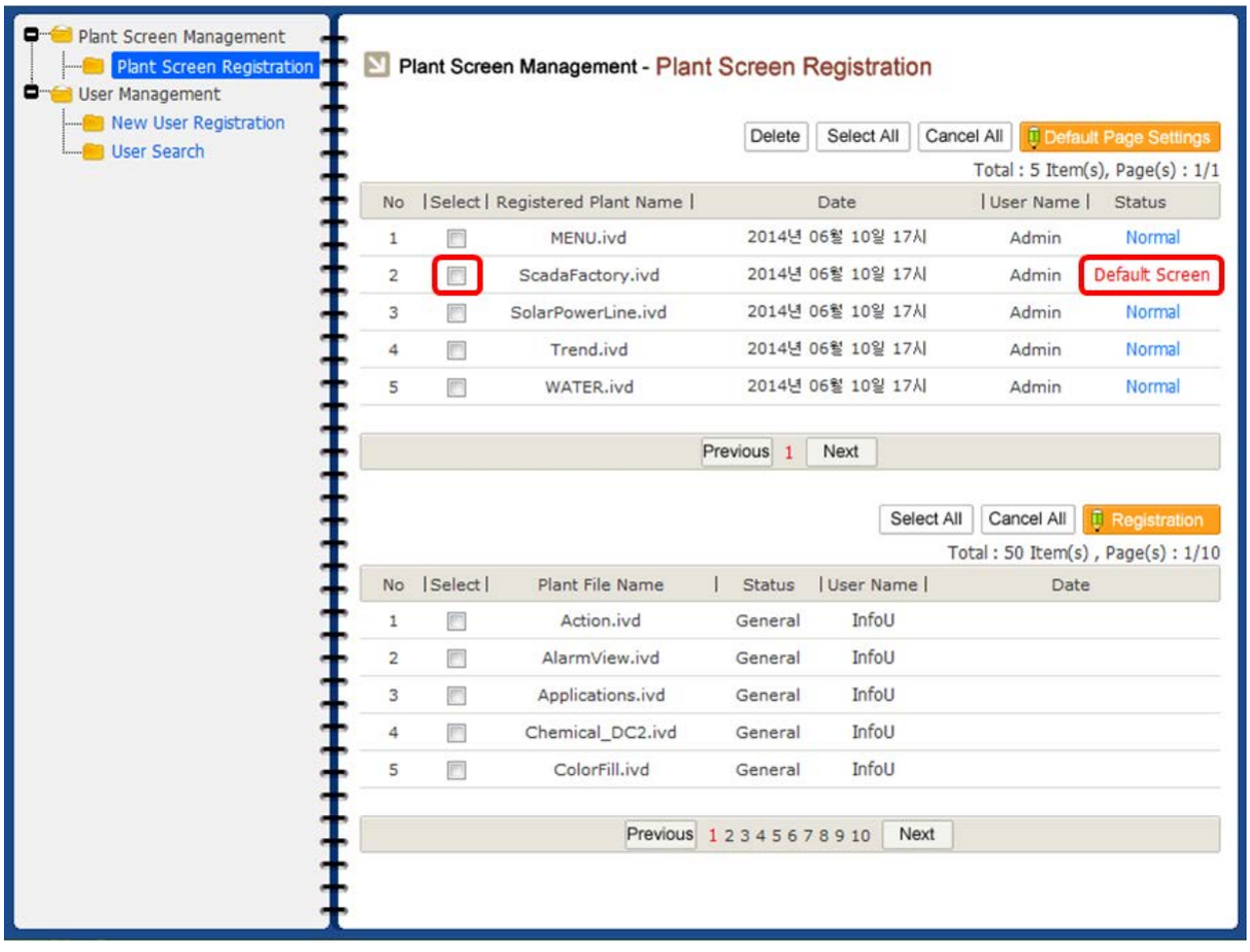
3) Single default screen can be specified for each user and it is impossible to set multiple screens.



The message confirming whether you will set the plant file as the default screen will be displayed.

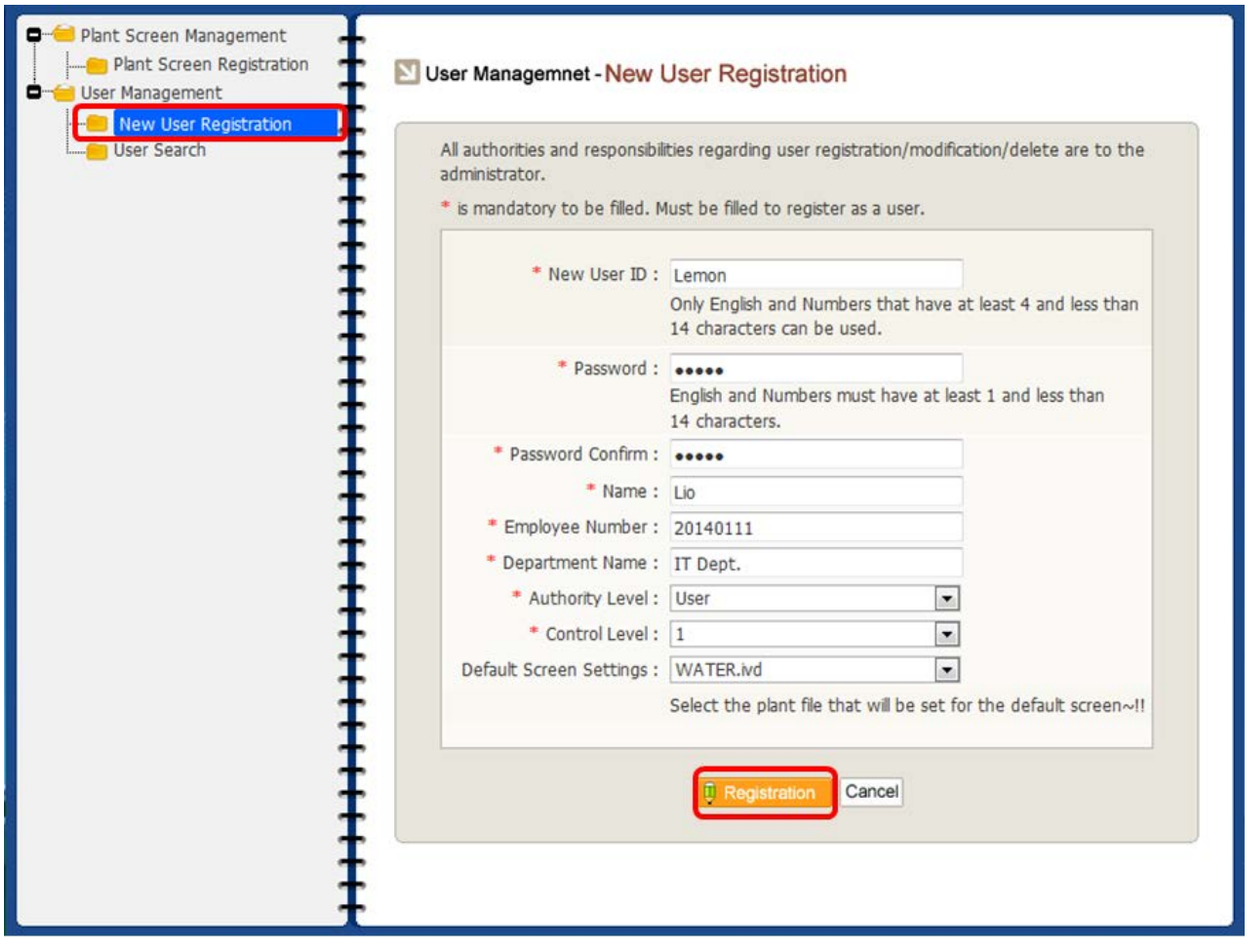


4) In the registration status, you can see the plant file that is set as the default screen is displayed as [Default Screen].



(5) New User Registraion

1) Register a general user.
2) The “*” mark means the required input items.
3) If a general user does not enter the default screen settings, the default screen set by the administrator will be displayed.
4) If the default screen is selected, the default screen specified by the user who logs in to the site is displayed.
5) The default screen can be separately set depending on the user.



The message notifying whether a user is normally registered will show up.

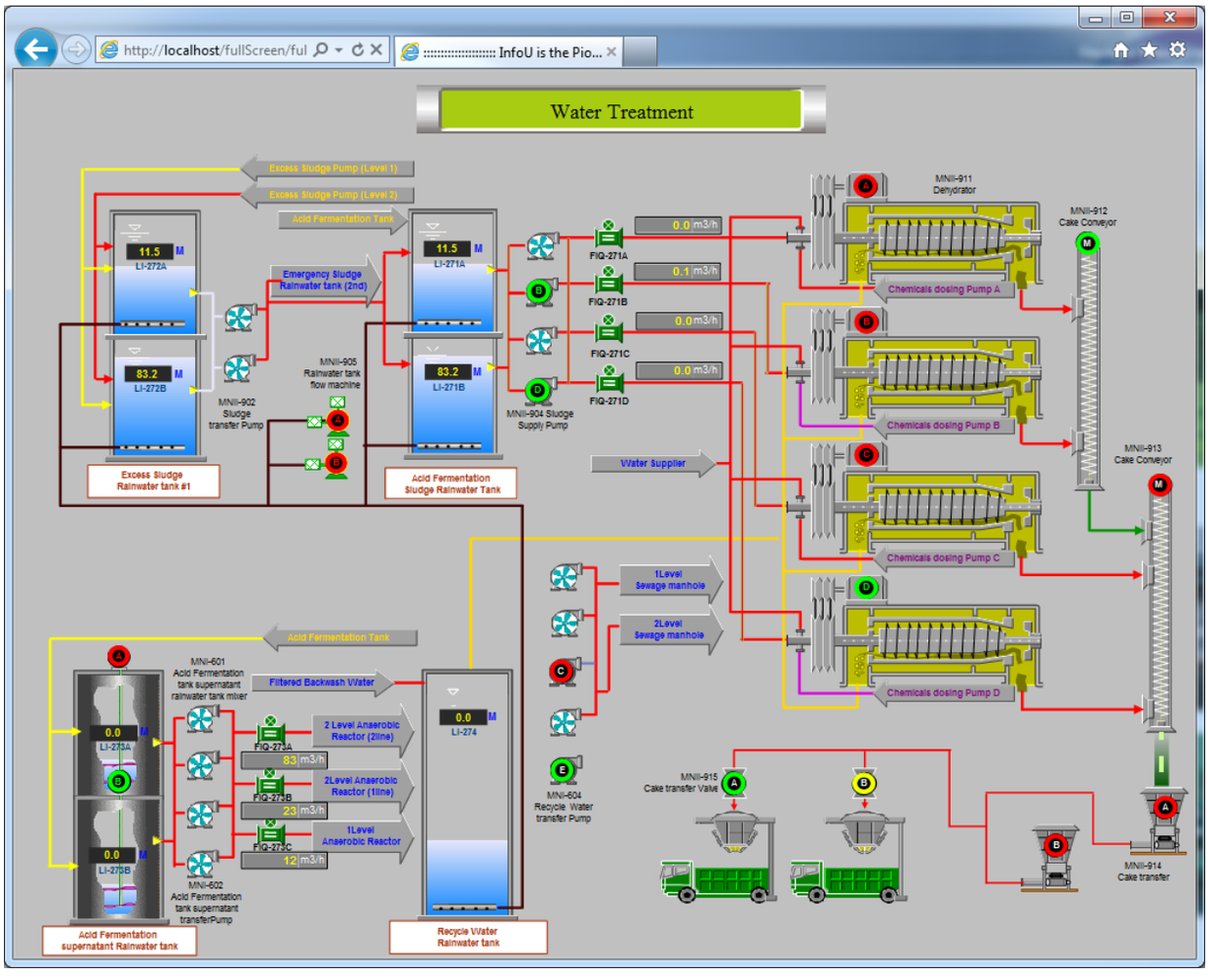
User Managemnet - New User Registration

Registration Complete

Result Code : [000000]
Success : 1 Item
Failure : 0 Item
Result Message : User is registered.

OK

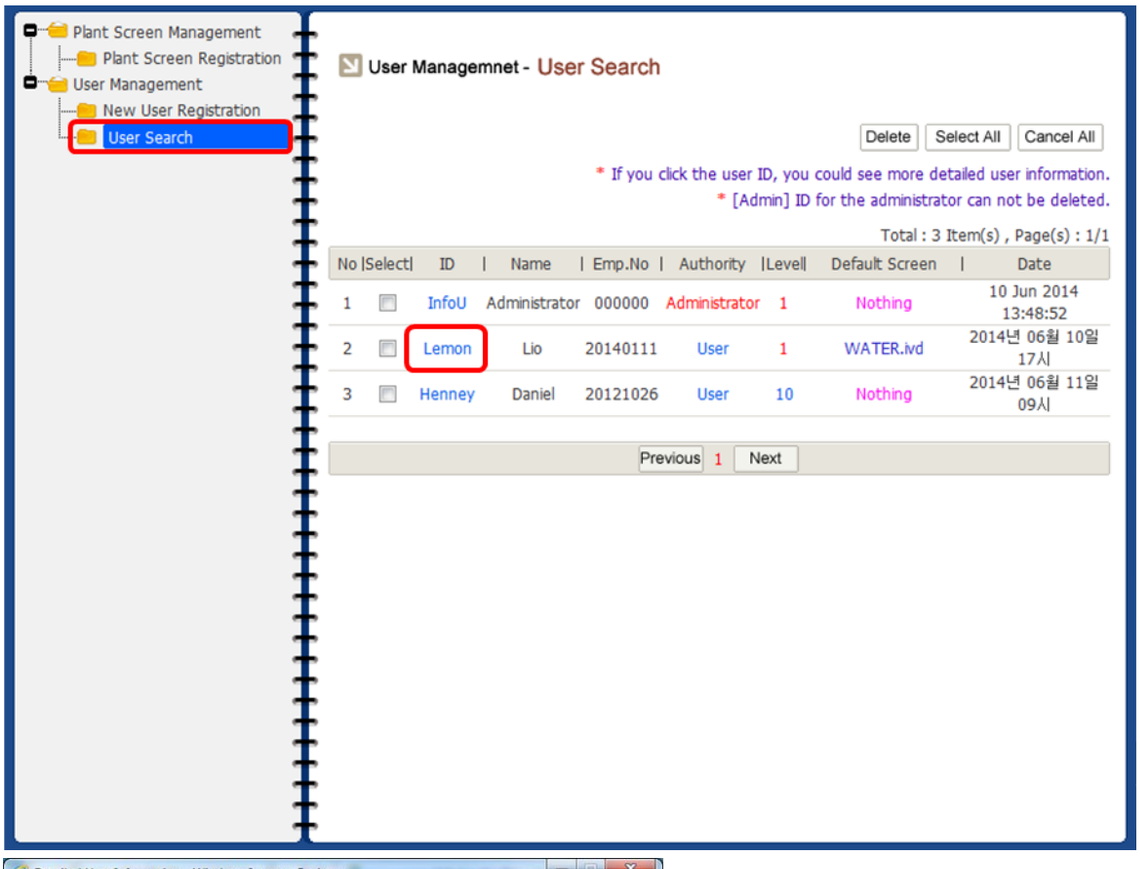
When you log in to the page with the newly registered user ID and password, the plant screen specified by the user will show up as the default screen.

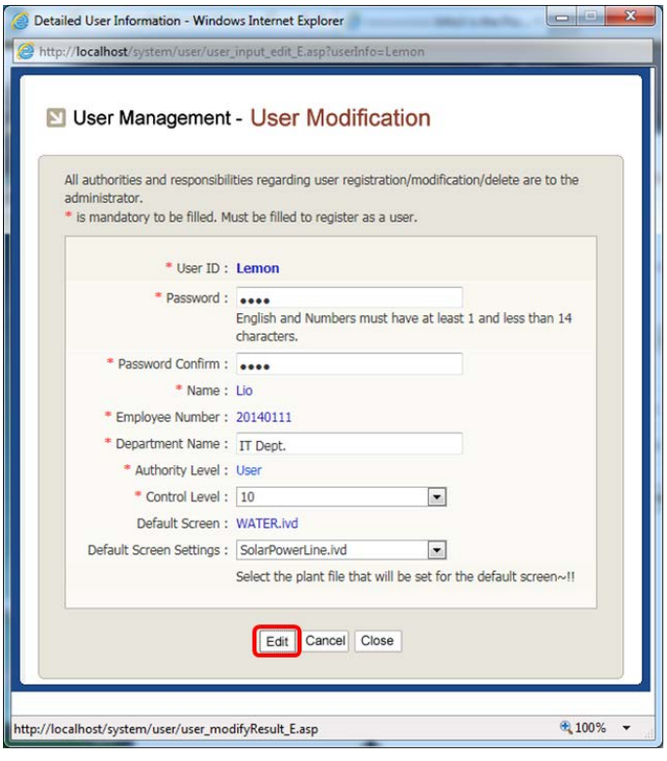


(6) User Search

1) You can search users to monitor and control the plant screen.

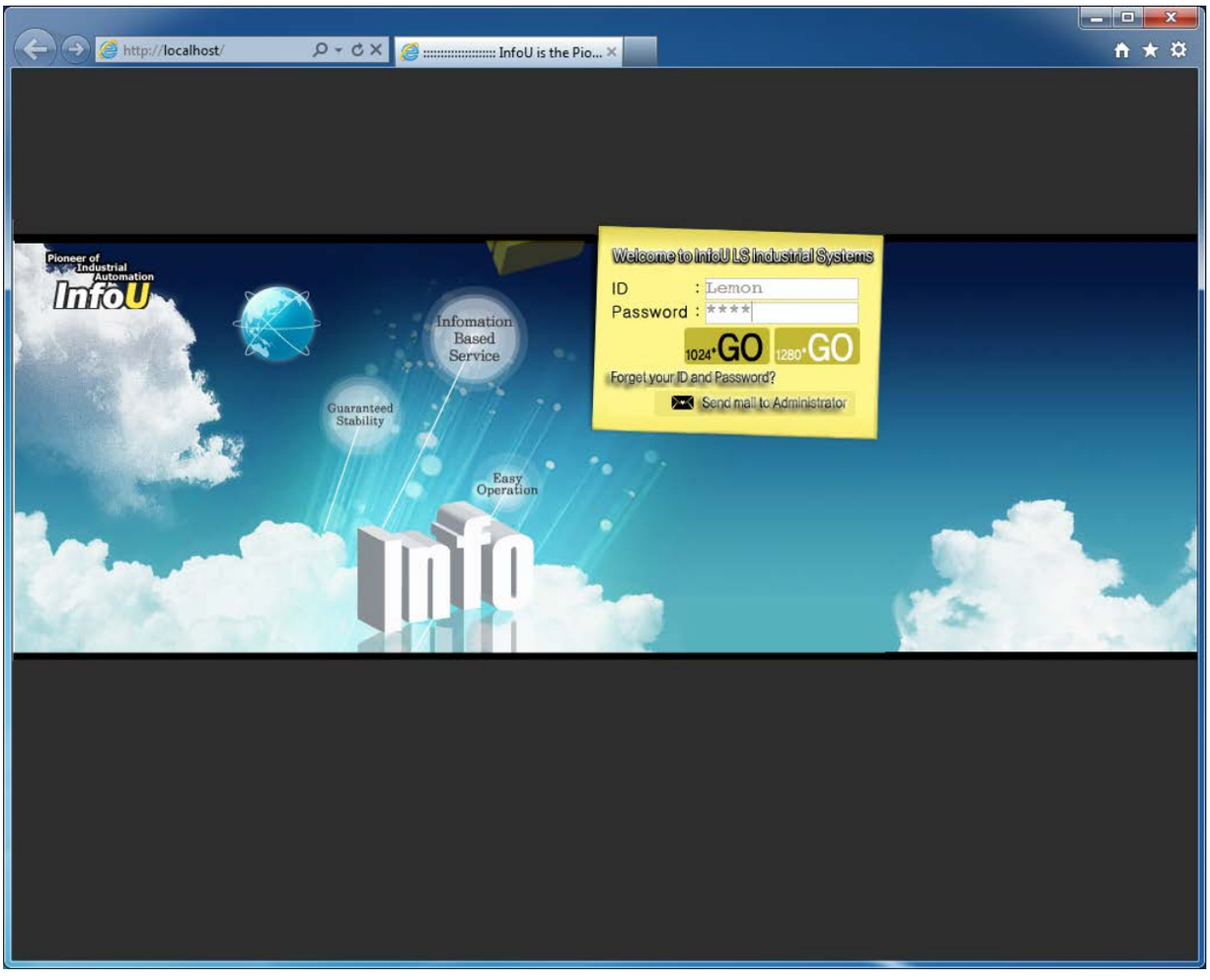
2) If you click the user ID, you can check and edit the detailed information of the user.



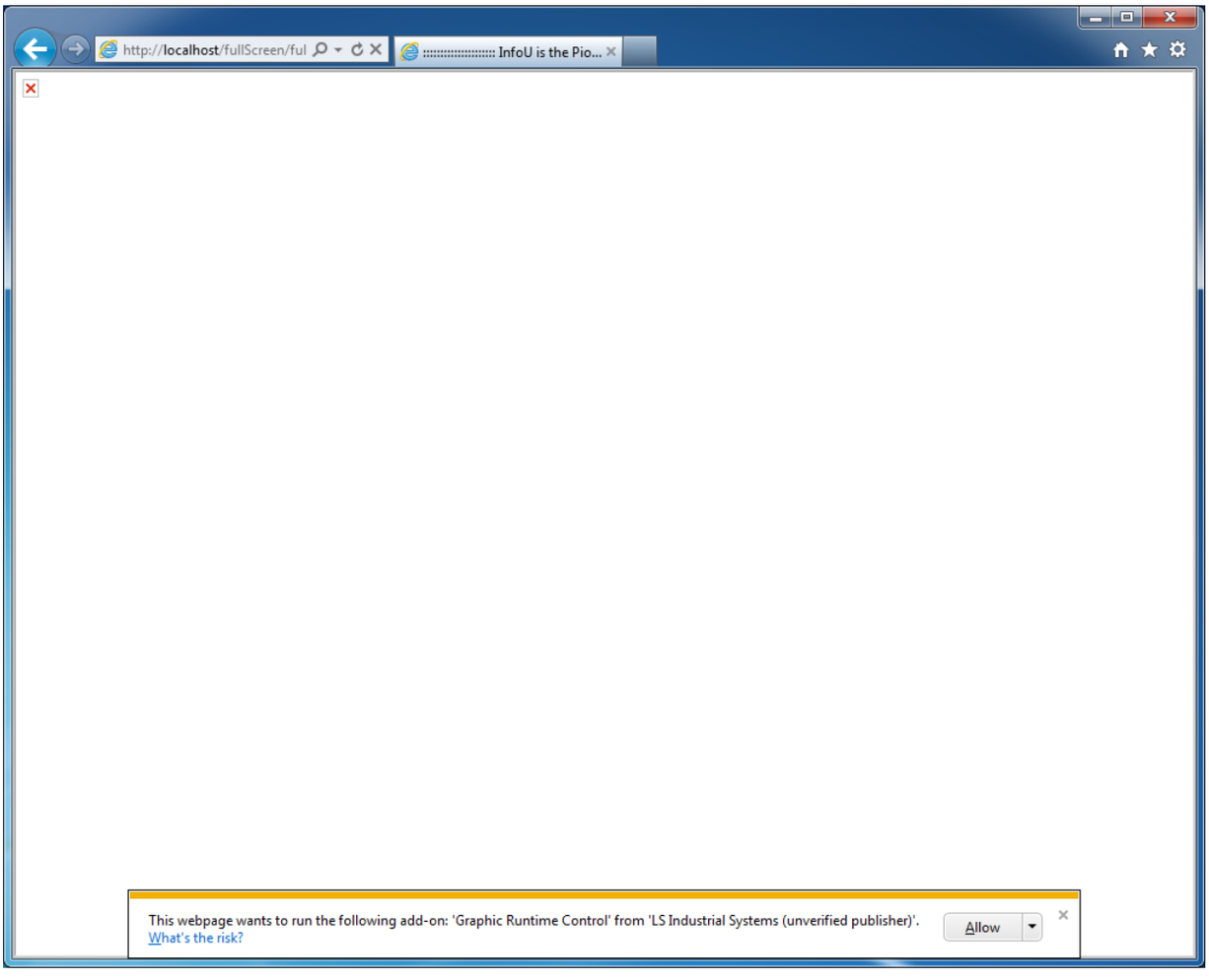


23.3.5 Monitoring Web User

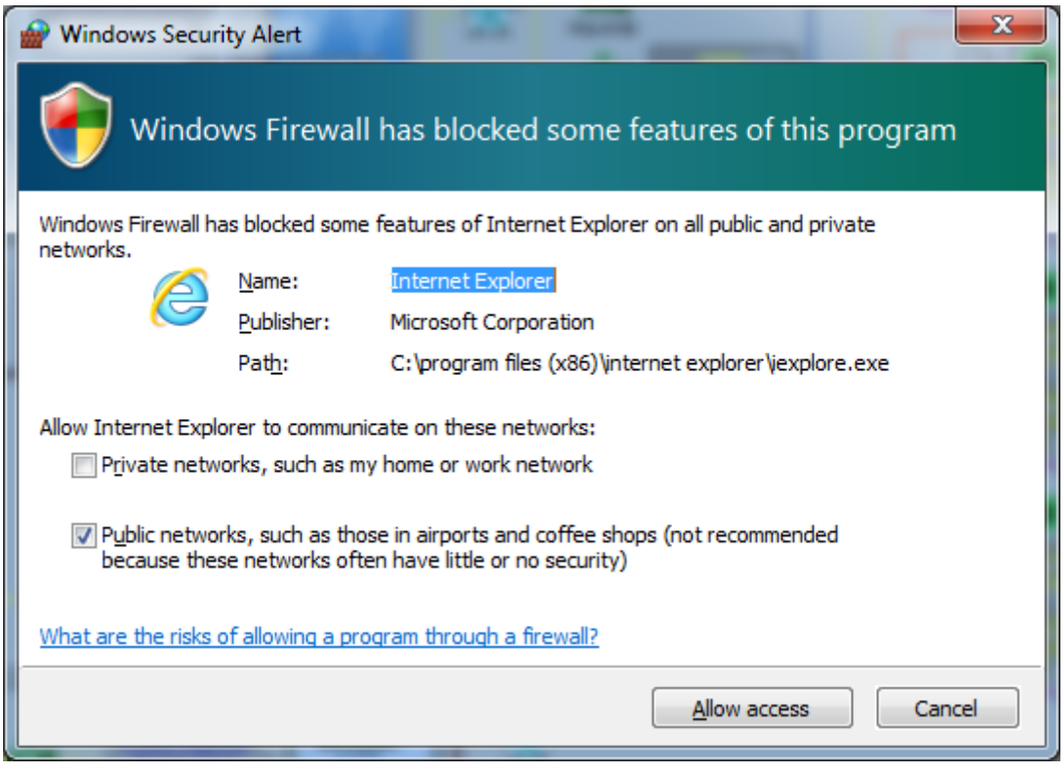
(1) Enter the ID and password of the registered user and then, click the [Go] button.



(2) If the security message asking whether executing ActiveX is displayed at the bottom of the page, click the [Allow] button.

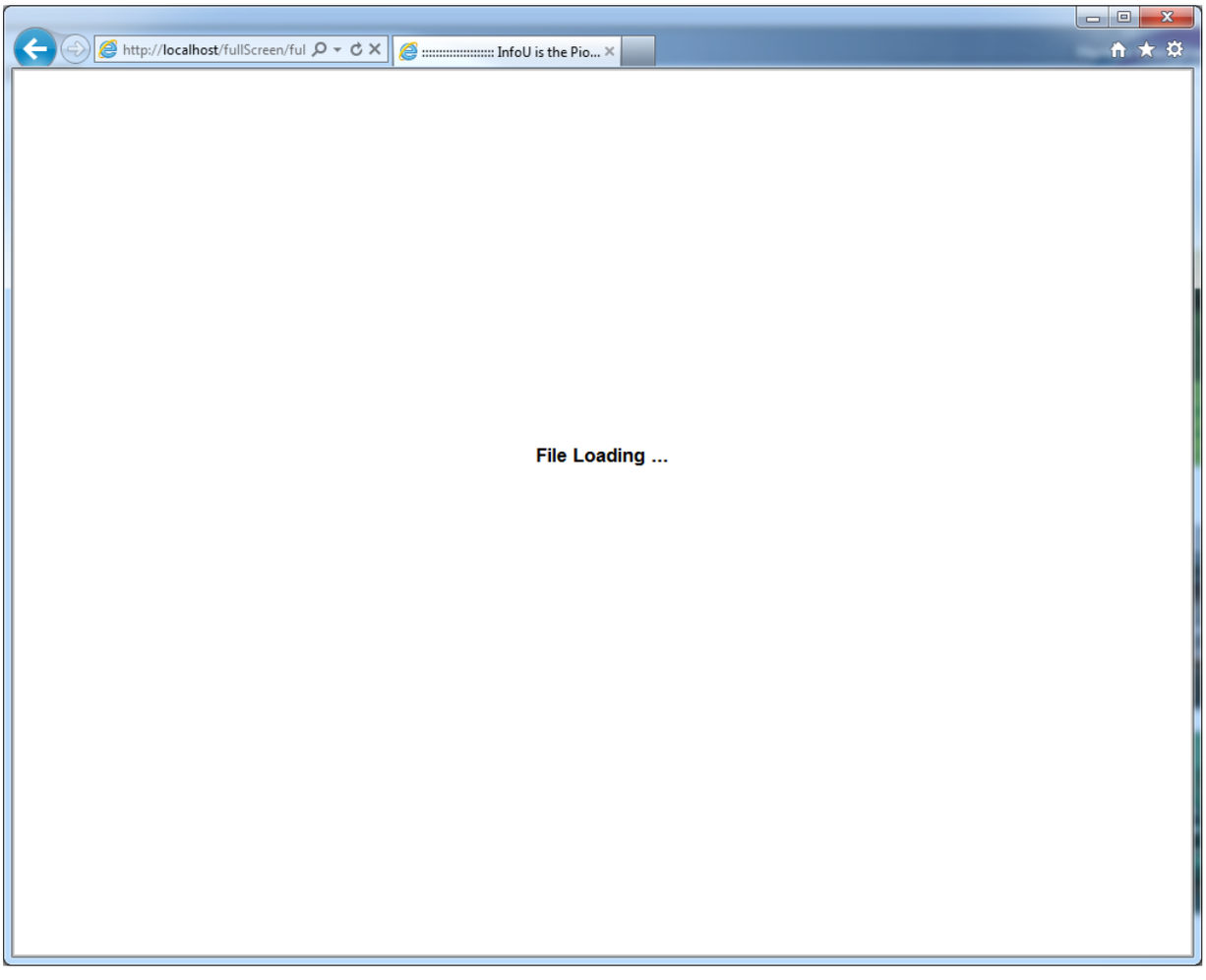


(3) If the message box related to firewall security is displayed, click the [Allow access] button.

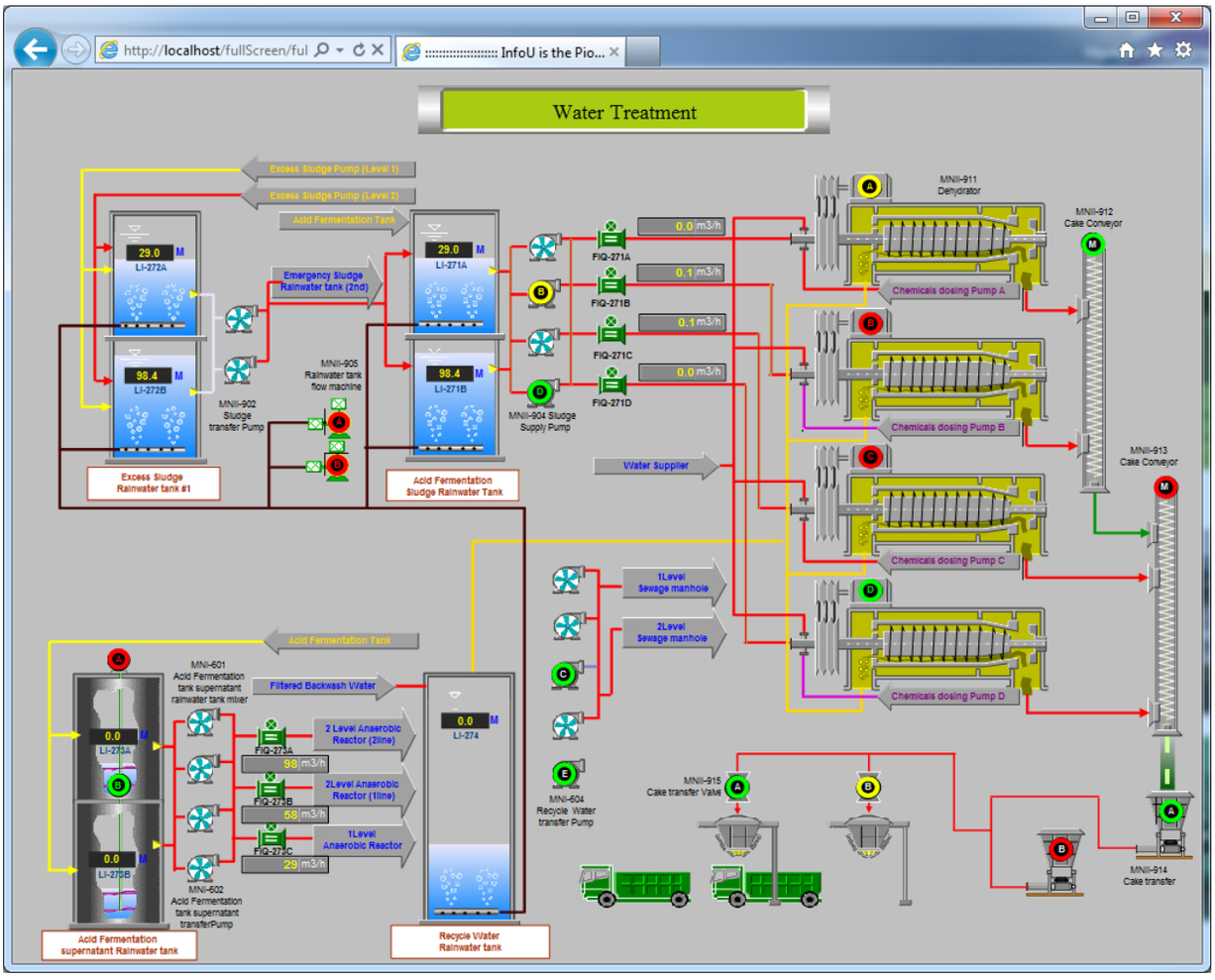


(4) The default screen specified by the user who logs in to the site is displayed in the Internet Explorer web browser

1) The message on loading the plant screen shows up.

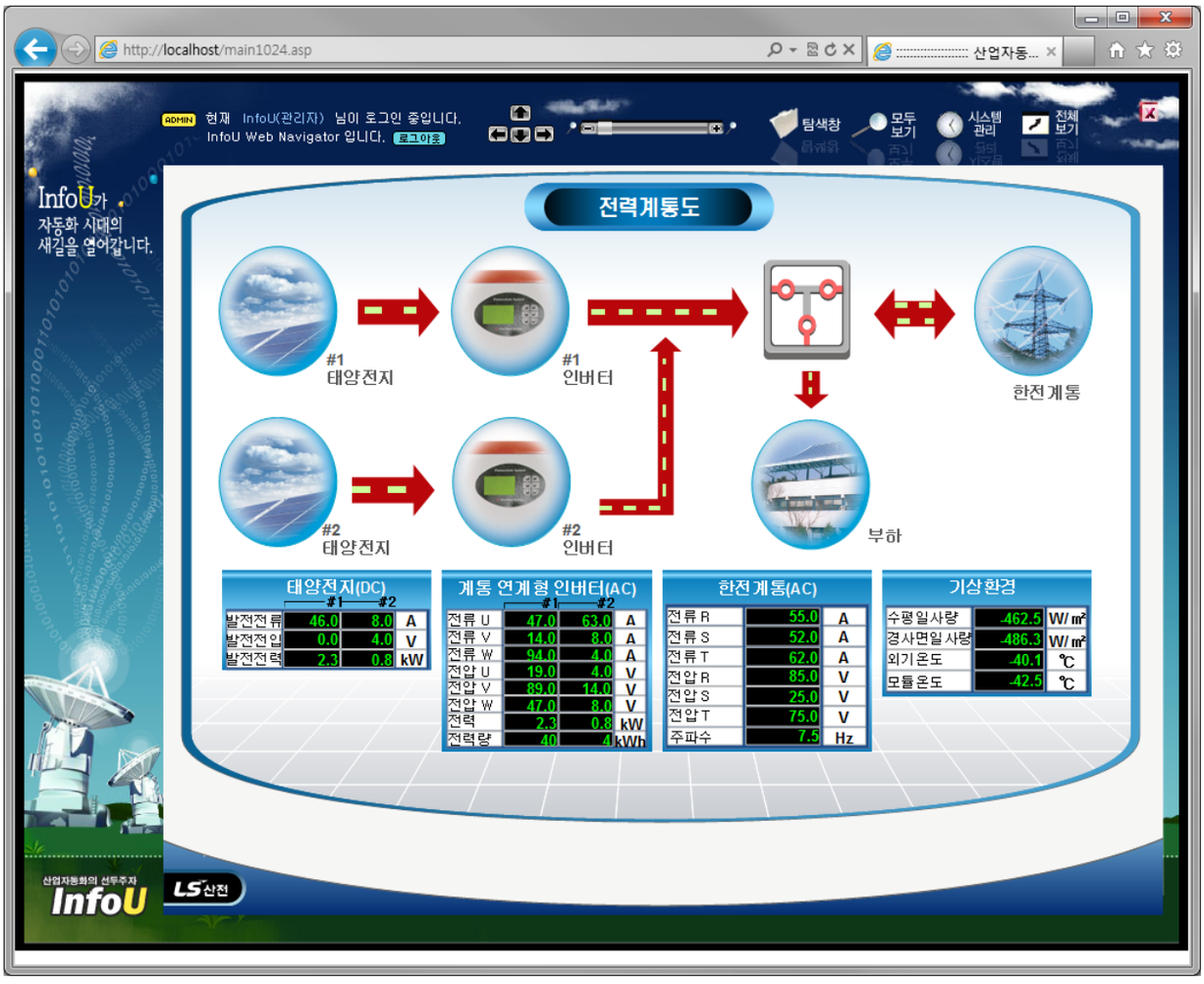


2) The main plant screen specified by a user is displayed.



23.3.6 Main Screen

This InfoU Plant main screen shows the relevant user's ID and the current system time after the user is successfully logged in. Click on a menu to show the functions the menu can perform on the InfoU Plant main screen. At first right after log in, the initial Plant screen which has been set up when the user registered the Plant Screen is loaded.



(1) 

Logout

If you press the logout button, you will log out from the InfoU (administrator) account and the login screen will be displayed.

(2)  Up/Down/Left/Right Arrow

It pretypifies the Up/Down/Left/Right screen depending on the hierarchy diagram of the currently registered plant screen. (The function is not available in the current system.)

1) Currently, the available function is the middle arrow only. If you click the middle arrow, the plant screen that is set as the default screen will be displayed in the main screen.

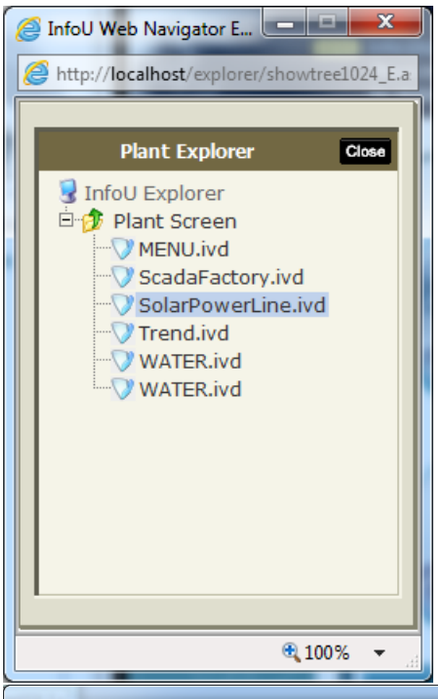


(3)  Zoom In/Zoom Out

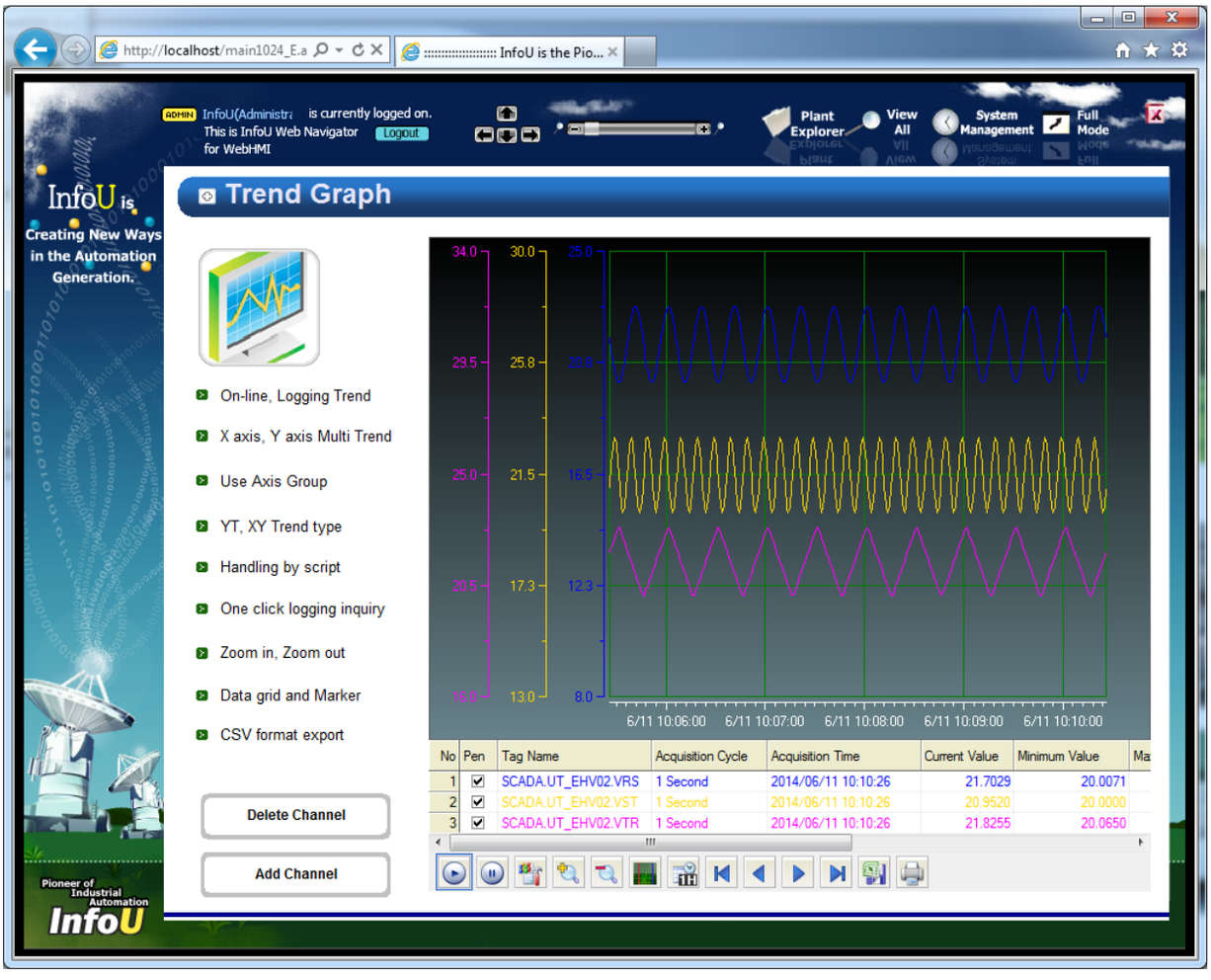
It zooms in and out the currently displayed plant screen. (The function is not available in the current system.)

(4) Plant Explorer

It shows the plant screens registered by a user in the format of tree structure. If you click the plant screen file, the plant screen will be displayed in the main screen. This function is useful to preview the whole plant screens.







(5)  View All

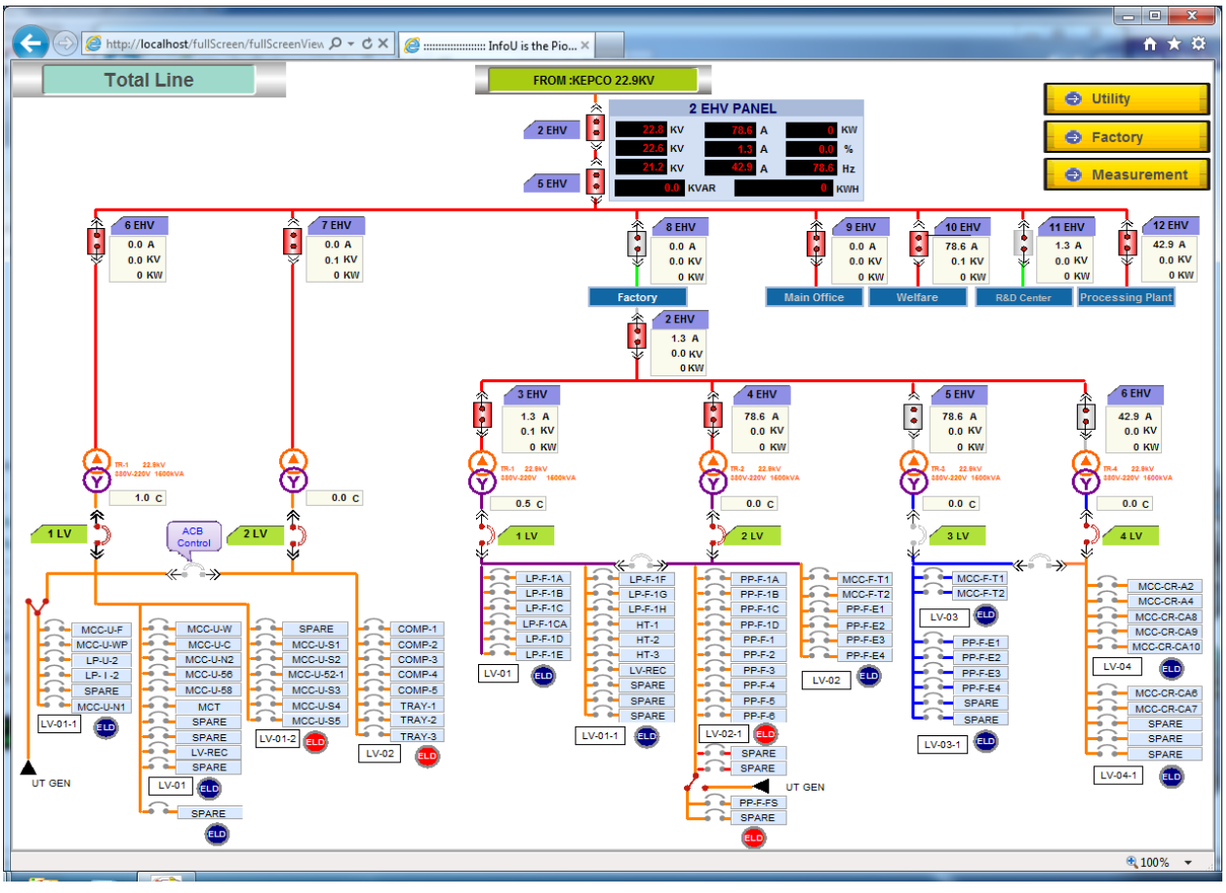
As a menu that makes the plant, trend, alarm, report screen better manageable, this menu show the entire menus on a single screen. (The function is not available in the current system.)

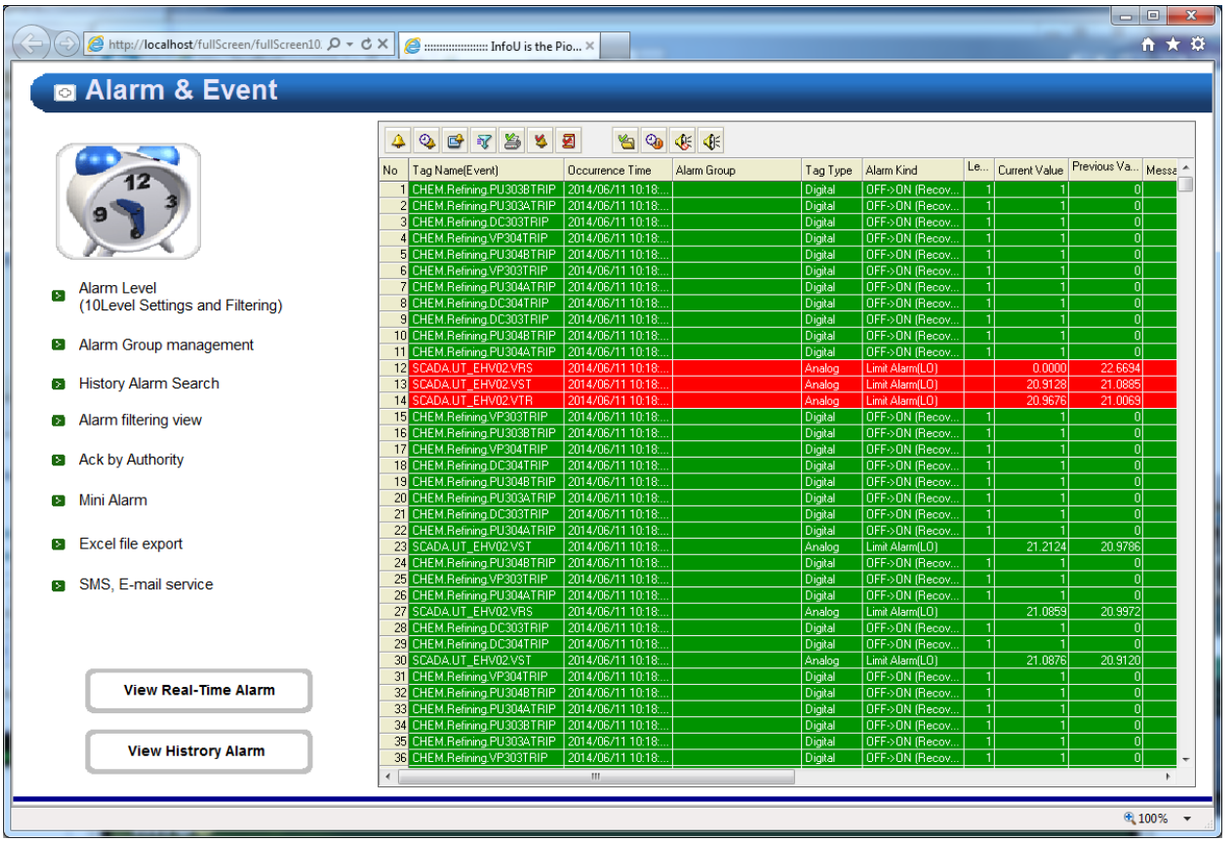
(6)  System Management

Through this menu, a new user is created and the management and plant screen is registered.

(7)  Full Mode

The plant screen is shown in a full screen without administrator's system menu. The plant screen is displayed in a new browser.

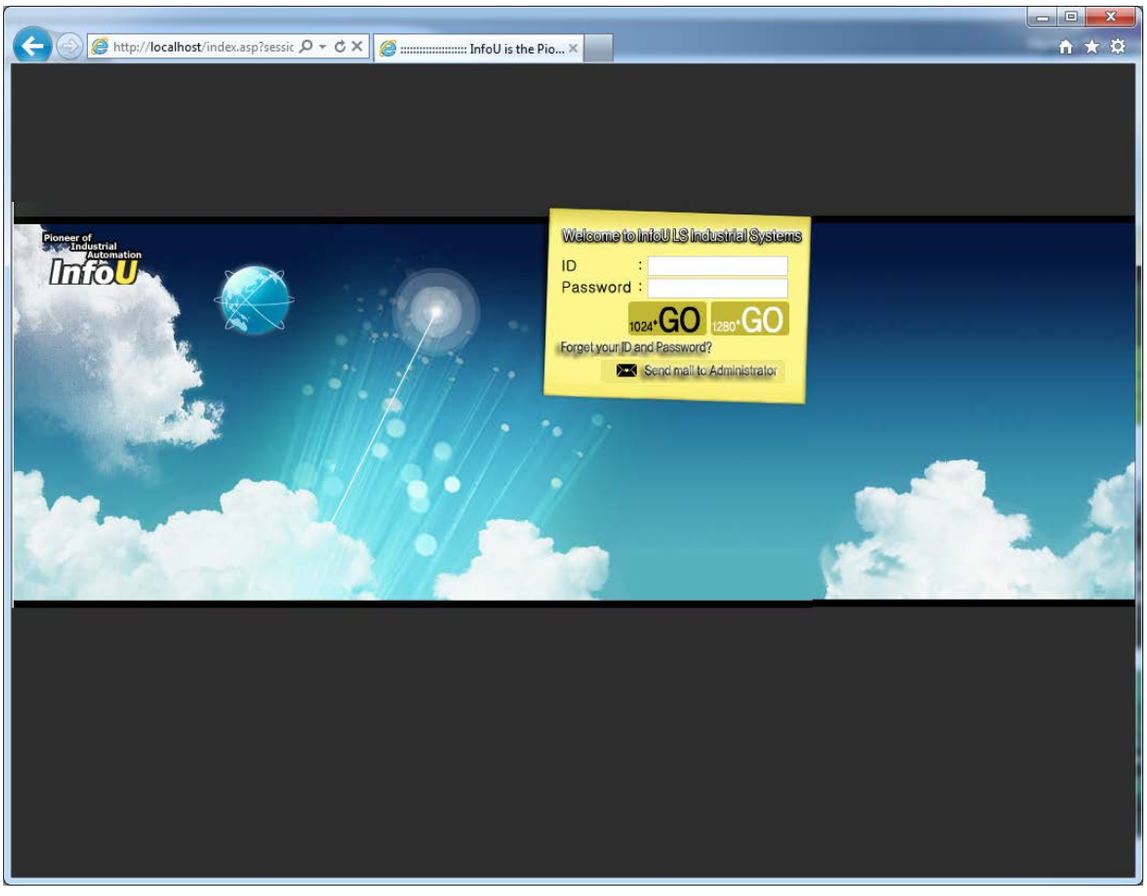




No	Tag Name(Event)	Occurrence Time	Alarm Group	Tag Type	Alarm Kind	Le...	Current Value	Previous Va...	Messa
1	CHEM.Refining.PU303BTRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
2	CHEM.Refining.PU303ATRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
3	CHEM.Refining.DC303TRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
4	CHEM.Refining.VP304TRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
5	CHEM.Refining.PU304BTRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
6	CHEM.Refining.VP303TRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
7	CHEM.Refining.PU304ATRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
8	CHEM.Refining.DC304TRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
9	CHEM.Refining.DC303TRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
10	CHEM.Refining.PU304BTRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
11	CHEM.Refining.PU304ATRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
12	SCADA.UT_EHV02.VRS	2014/06/11 10:18:...		Analog	Limit Alarm(LO)		0.0000	22.6694	
13	SCADA.UT_EHV02.VST	2014/06/11 10:18:...		Analog	Limit Alarm(LO)		20.9128	21.0885	
14	SCADA.UT_EHV02.VTR	2014/06/11 10:18:...		Analog	Limit Alarm(LO)		20.9676	21.0069	
15	CHEM.Refining.VP303TRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
16	CHEM.Refining.PU303BTRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
17	CHEM.Refining.VP304TRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
18	CHEM.Refining.DC304TRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
19	CHEM.Refining.PU304BTRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
20	CHEM.Refining.PU303ATRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
21	CHEM.Refining.DC303TRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
22	CHEM.Refining.PU304ATRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
23	SCADA.UT_EHV02.VST	2014/06/11 10:18:...		Analog	Limit Alarm(LO)		21.2124	20.9786	
24	CHEM.Refining.PU304BTRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
25	CHEM.Refining.VP303TRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
26	CHEM.Refining.PU304ATRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
27	SCADA.UT_EHV02.VRS	2014/06/11 10:18:...		Analog	Limit Alarm(LO)		21.0859	20.9972	
28	CHEM.Refining.DC303TRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
29	CHEM.Refining.DC304TRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
30	SCADA.UT_EHV02.VST	2014/06/11 10:18:...		Analog	Limit Alarm(LO)		21.0876	20.9120	
31	CHEM.Refining.VP304TRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
32	CHEM.Refining.PU304BTRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
33	CHEM.Refining.PU304ATRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
34	CHEM.Refining.PU303BTRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
35	CHEM.Refining.PU303ATRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	
36	CHEM.Refining.VP303TRIP	2014/06/11 10:18:...		Digital	OFF->ON (Recov...	1	1	0	

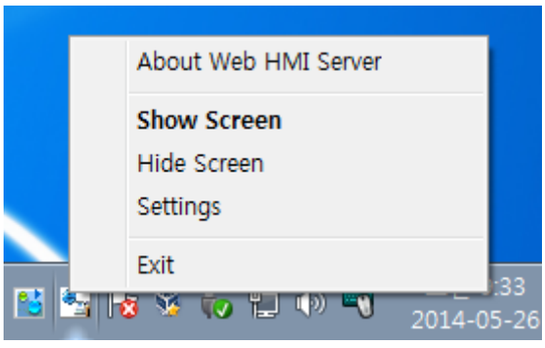
(8)  Close Window

It closes the current window to log out.



23.4 Web HMI Server

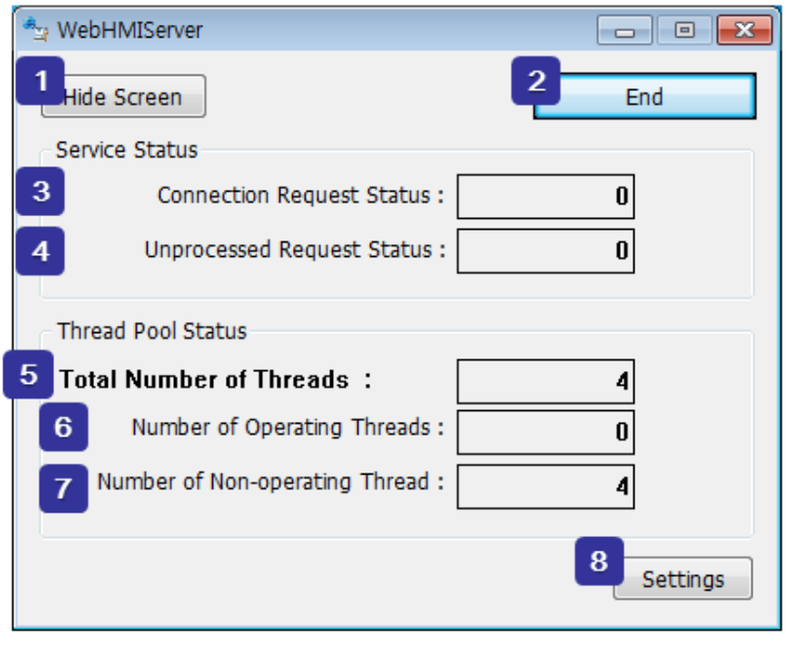
The Web HMI Server is to play a role in extending to access to the InfoU HMI through web. To present HMI service through the web, the Web HMI Server has to run on the server side. To make the Web HMI Server run normally, the InfoU project should run. Also, to access to the Web HMI Server, the Internet Information Server has to be installed and relevant files needed for the HMI service should be copied and installed.



If the system runs, the tray icons are created and Show Screen, Hide Screen, Settings and Exit function are performed through the pop-up menu.

23.4.1 Operation Settings

The Web HMI Server (iuWebHMIServer.exe)runs automatically if the web is set up on the project option. Select 'View Screen' menu from the Tray Icons to display the following screen.

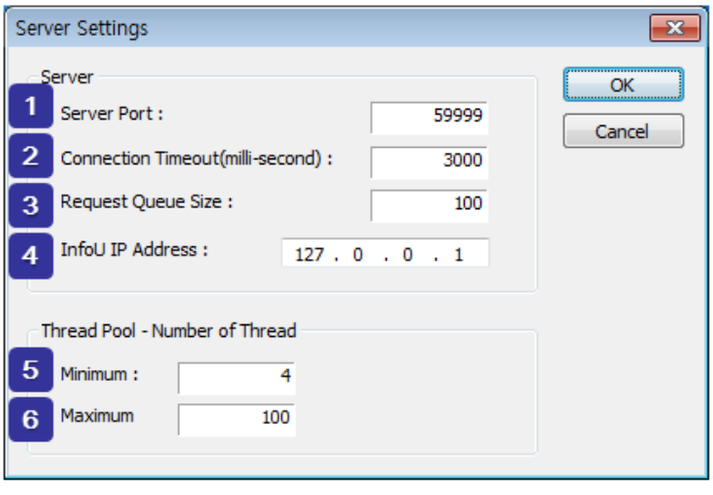


1	Hide this operation screen.
2	The Web HMI Server stops.

3	The total number of request for connection from the client is displayed.
4	The number of remaining requests from the client, which has not been processed, is displayed.
5	The total number of the created threads in the thread pool is displayed to process connection.
6	The number of operating threads after being assigned the current work is displayed.
7	The number of standby threads in the thread pool is displayed.
8	This button is used when opening the server setting screen to change setting values.

23.4.2 Server Settings

If you execute **"Settings"** menu in the tray icon or click the "**Settings**" button in the operation screen, you can see the Server Settings window as below.



1	Set up a server port.
2	Set up a timeout time (milliseconds) used for connecting to TCP/IP Socket. When setting up, make sure to take into account of some delay due to the network status.
3	Set up a size of the request queue.
4	As the address of the PC where the InfoU HMI is located, it has been set up with "127.0.0.1"as the default address.
5	Set up the number of the created threads during the first running.
6	Set up the maximum number of the available threads to be created in the thread pool.

Chapter 24 Redundancy and Client Server

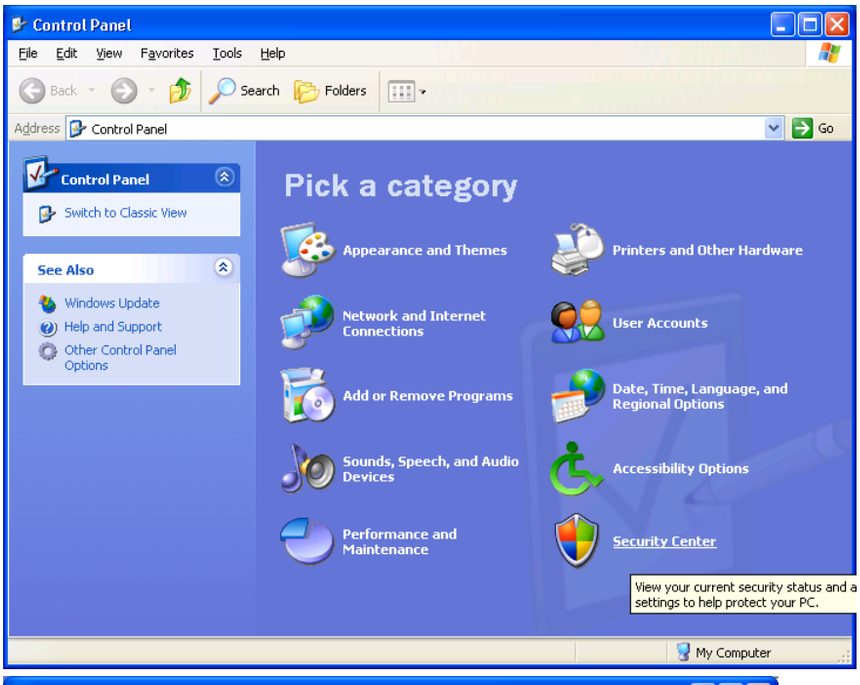
24.1 System Environment Settings

24.1.1 Cancel Network Security

Since data shall be exchanged on the network to configure Redundancy and Client Server systems, all of the related computers have to cancel their security as follows:

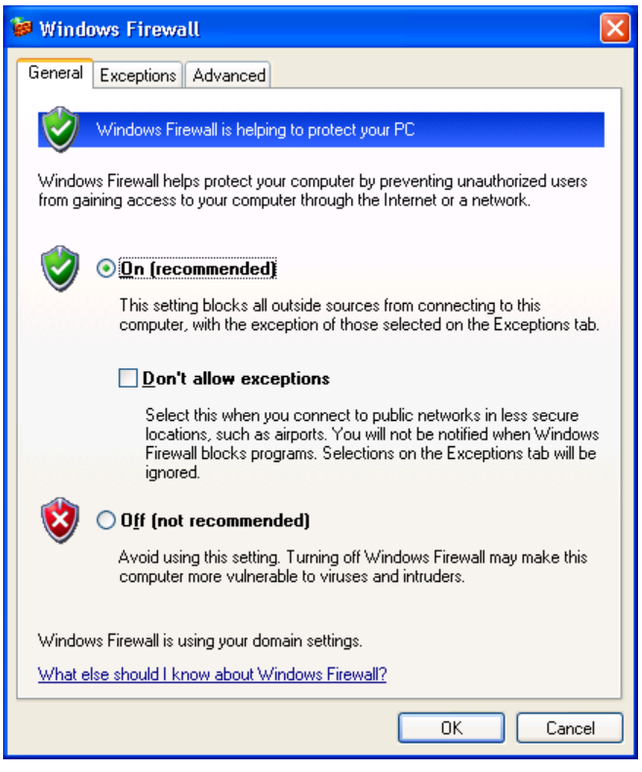
(1) Windows XP

① Run the security center on the control board.

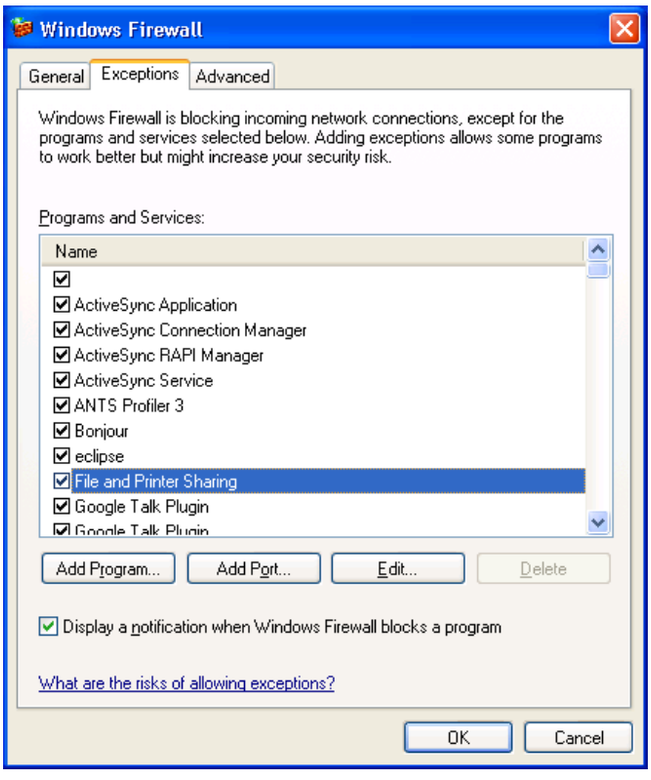




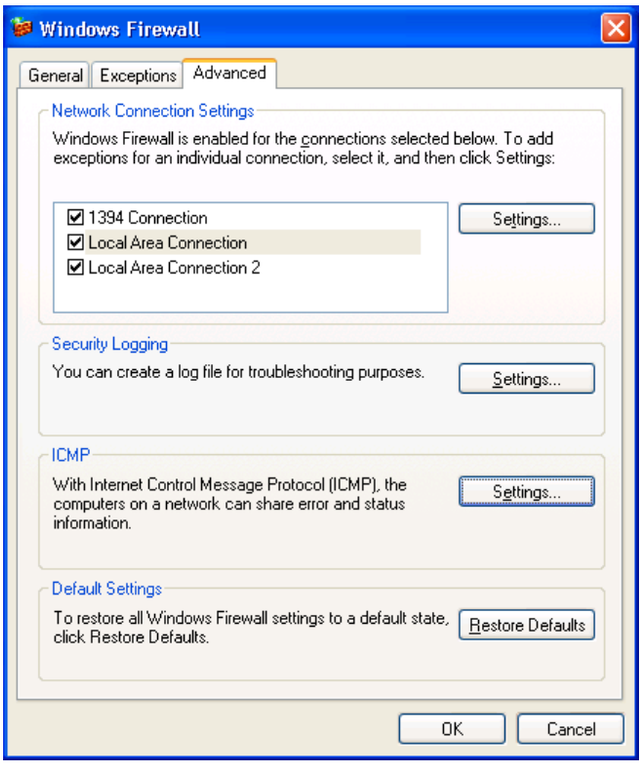
② Run Windows Firewall on the security center settings



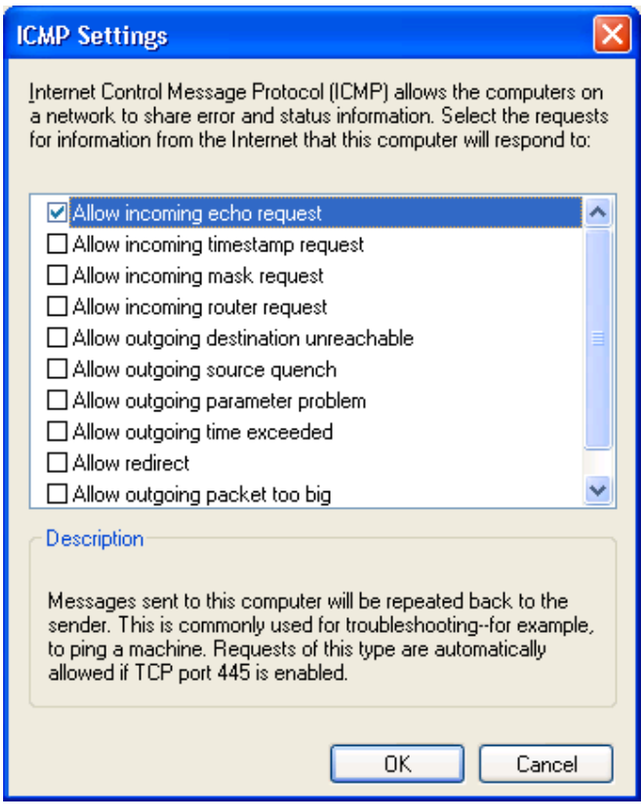
③ Allow exceptions on the Windows Firewall setting screen as follows: Select “Fire and Printer Sharing” on the Exceptions tap.



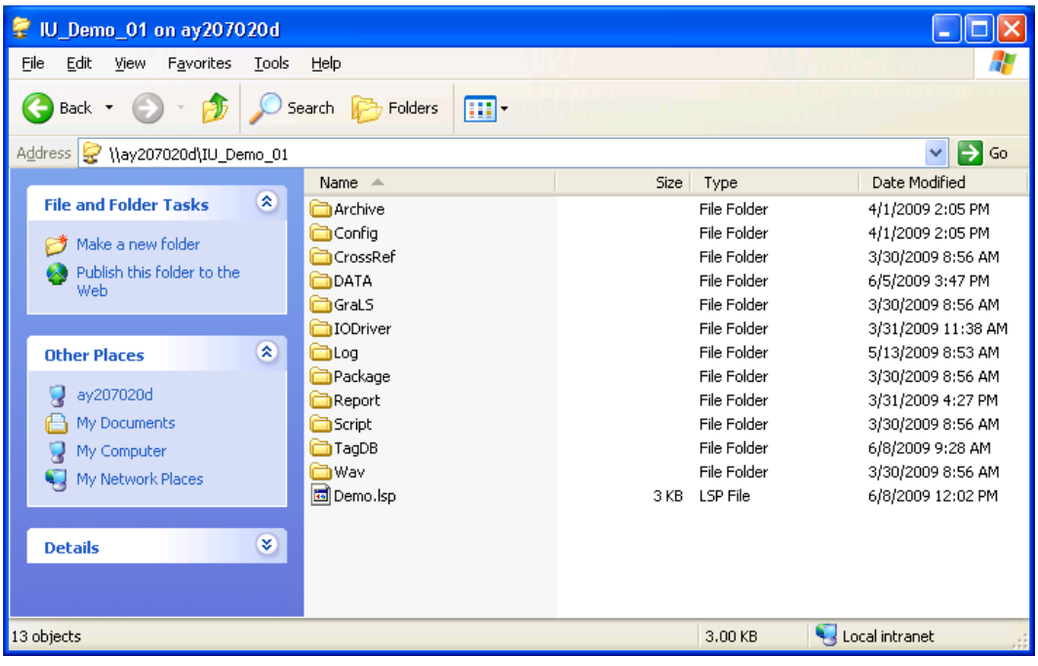
④ Select ‘Advanced’ tap on the Window Firewall setting screen and click on “Settings” button of ICMP.



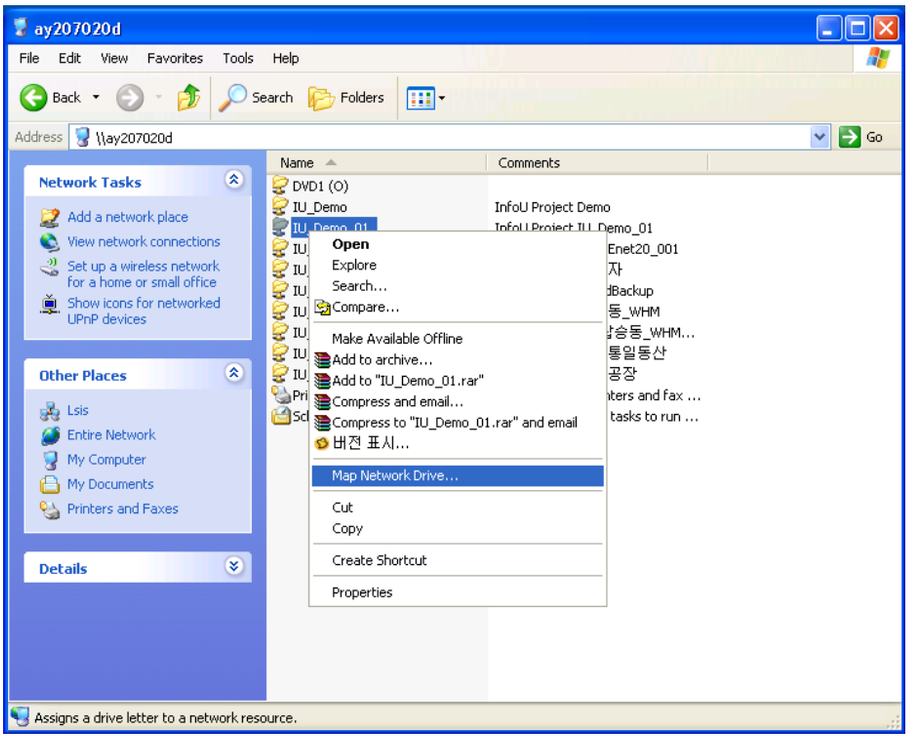
⑤ Select “Allow incoming echo request.



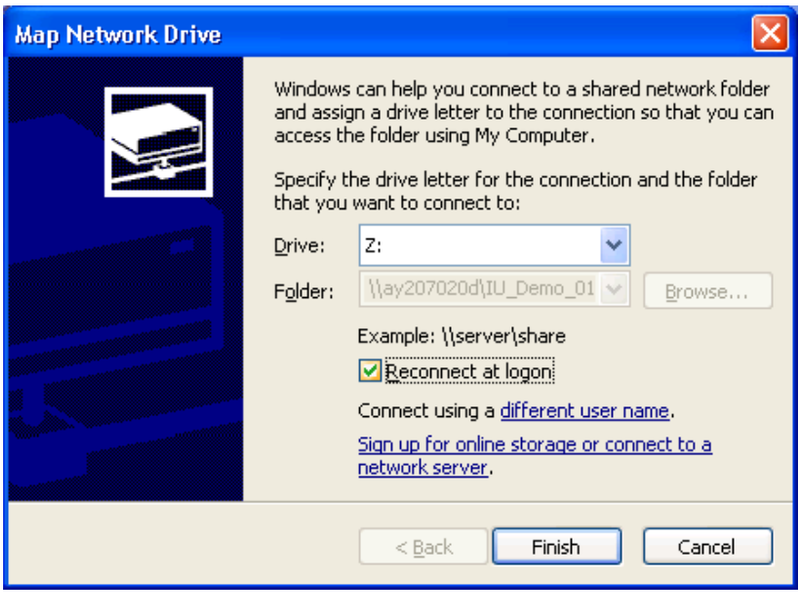
⑥ It should be possible to connect to the joint folder of the partner computer.



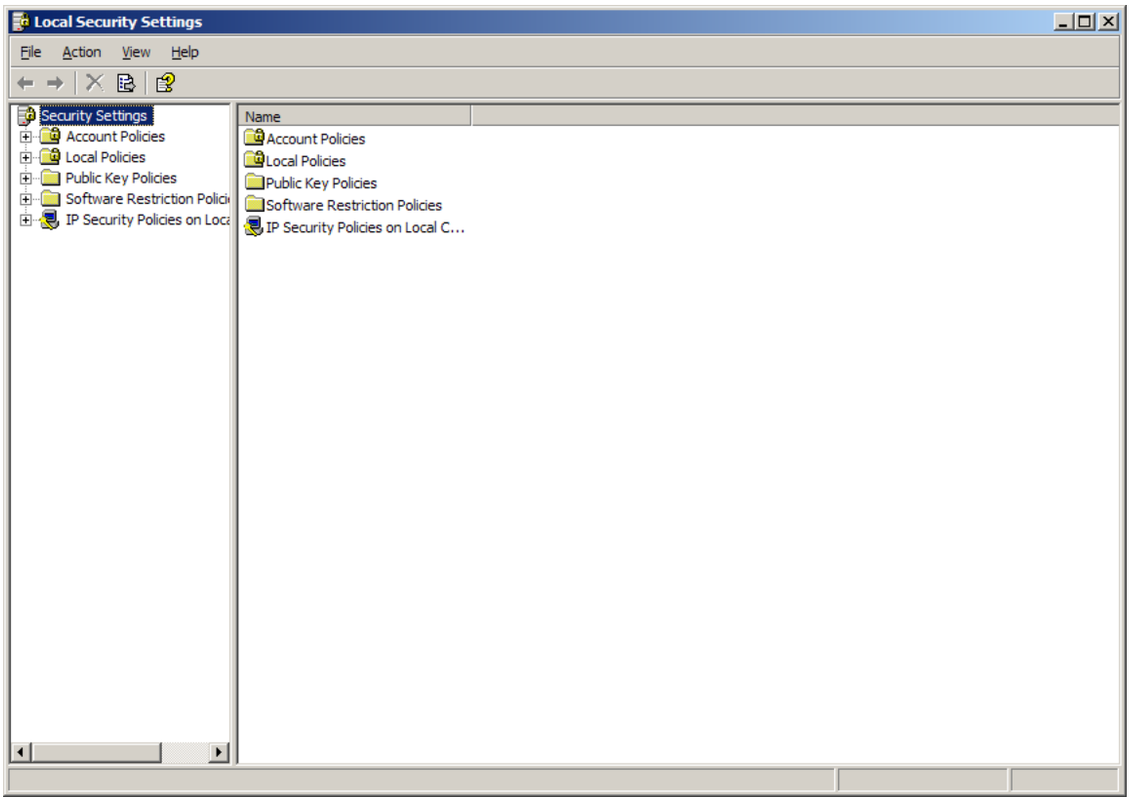
⑦ To make it connectable even when rebooting, set up ‘Reconnect at logon’ and make it reconnect to the relevant folder automatically when booting.



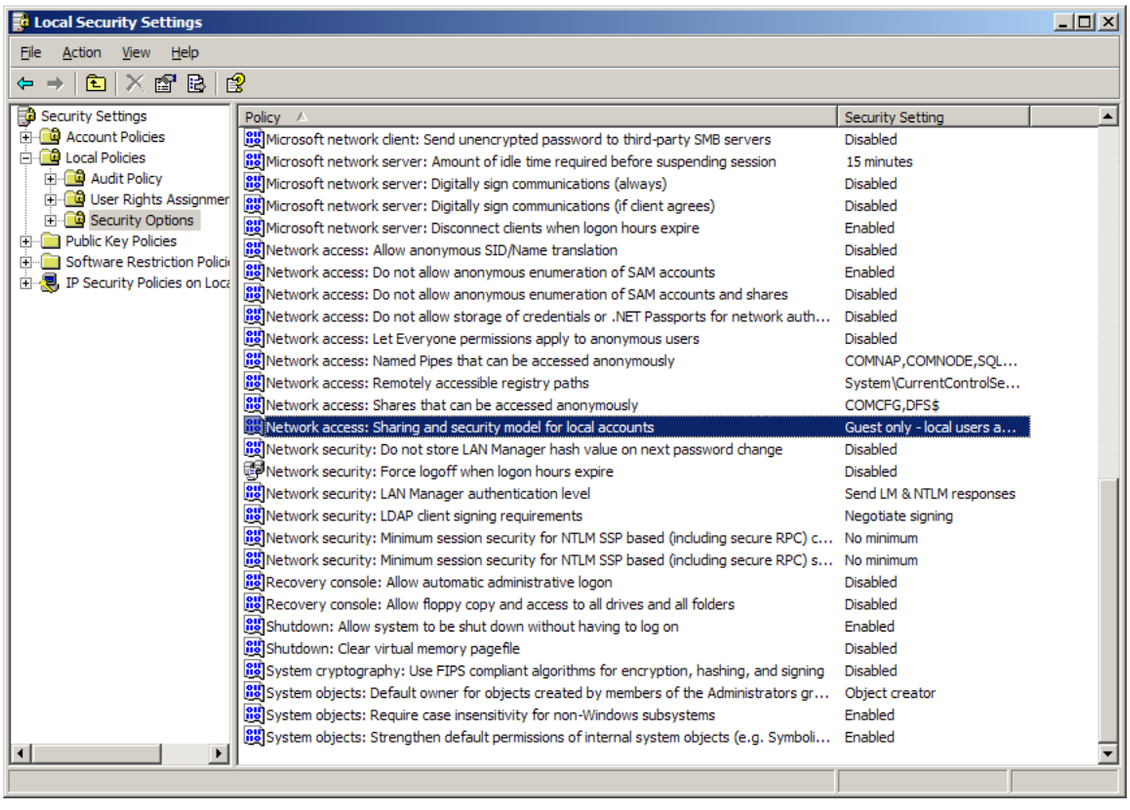
⑧ Select ‘Reconnect at logon’ on the Map Network Driver.



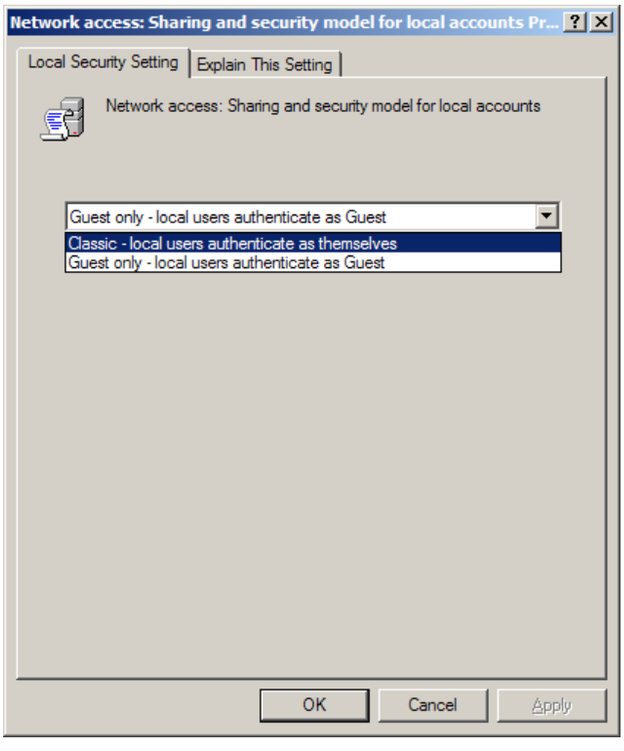
⑨ Execute Control Board → Management Tool → Local Policies.



⑩ Local Policies → Security Options → “Network Access: Execute “Sharing and security model for local accounts”



⑪ Select“Classic – Local users authenticate as themselves”.

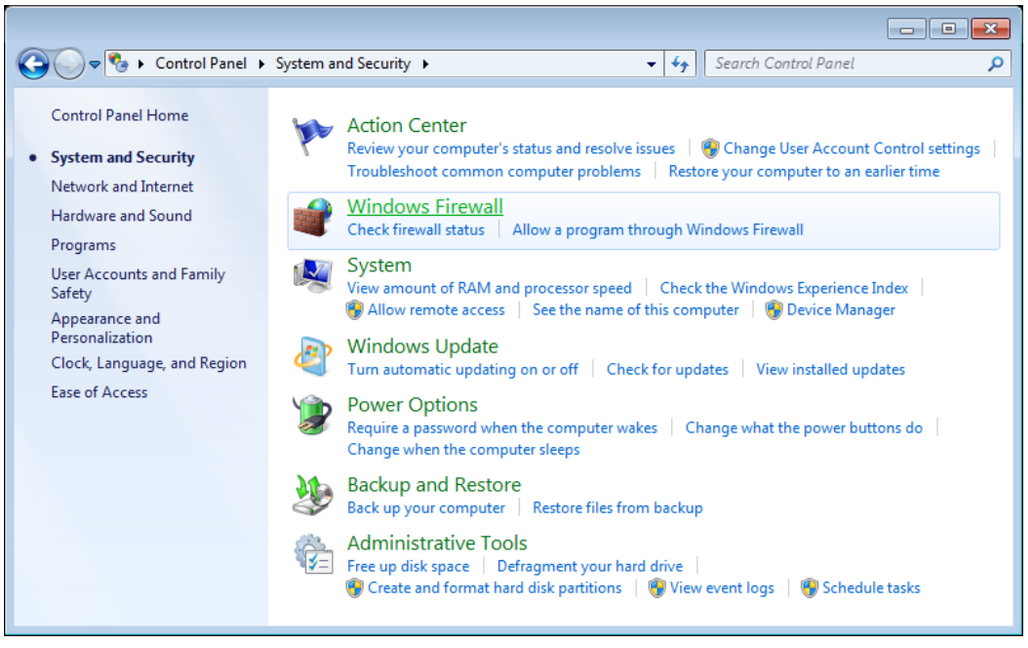


(2) Windows7

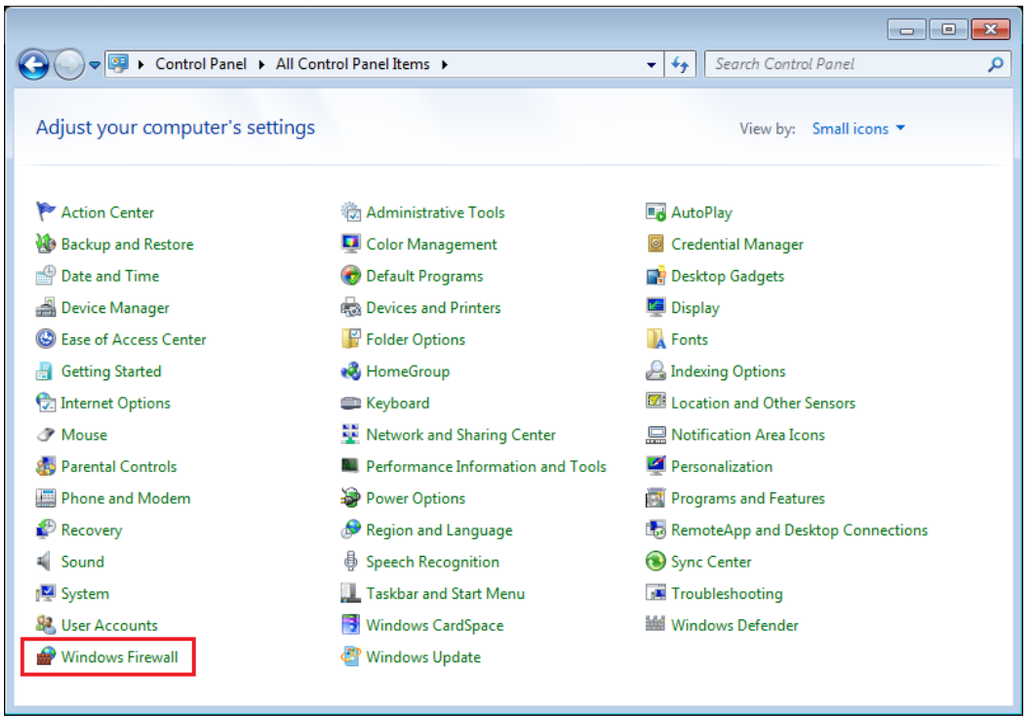
① Enables the Windows Firewall in control panel

a. Click [System and Security] → [Windows Firewall].



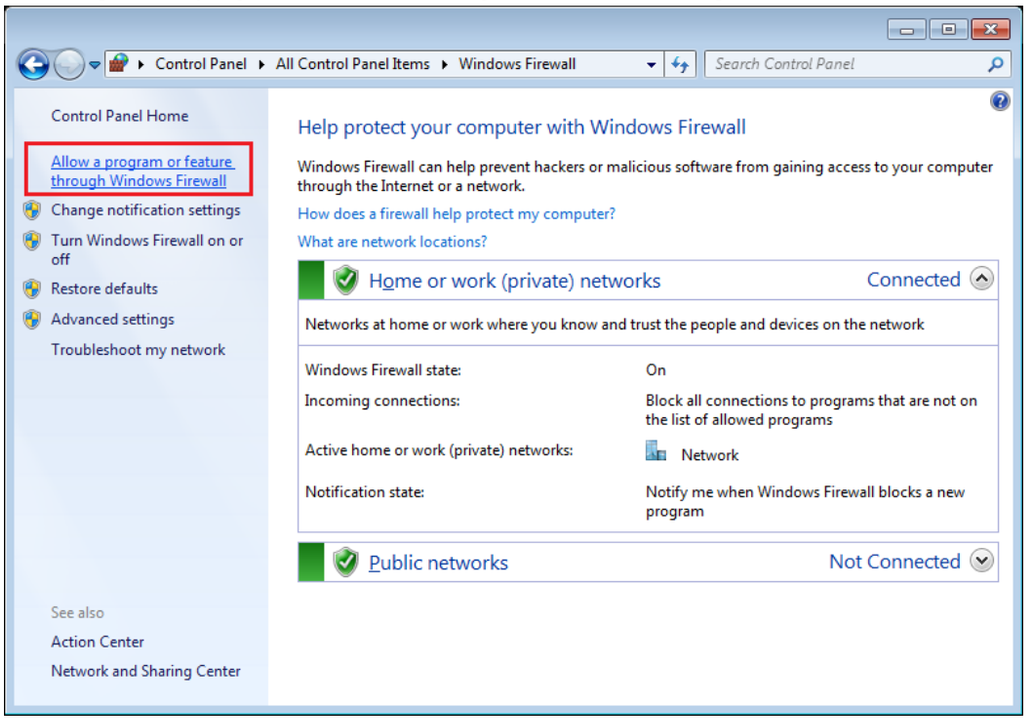


b. Click [Windows Firewall]

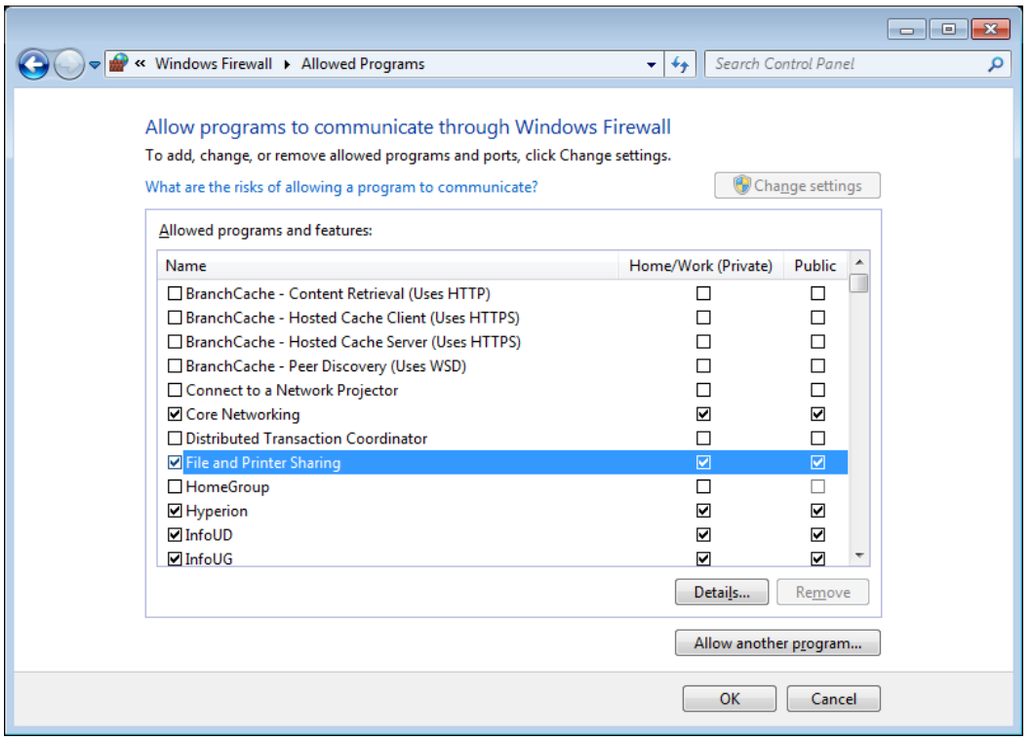


② Configuration of File and Printer Sharing

a. Click [Allow a program or feature through Windows Firewall]

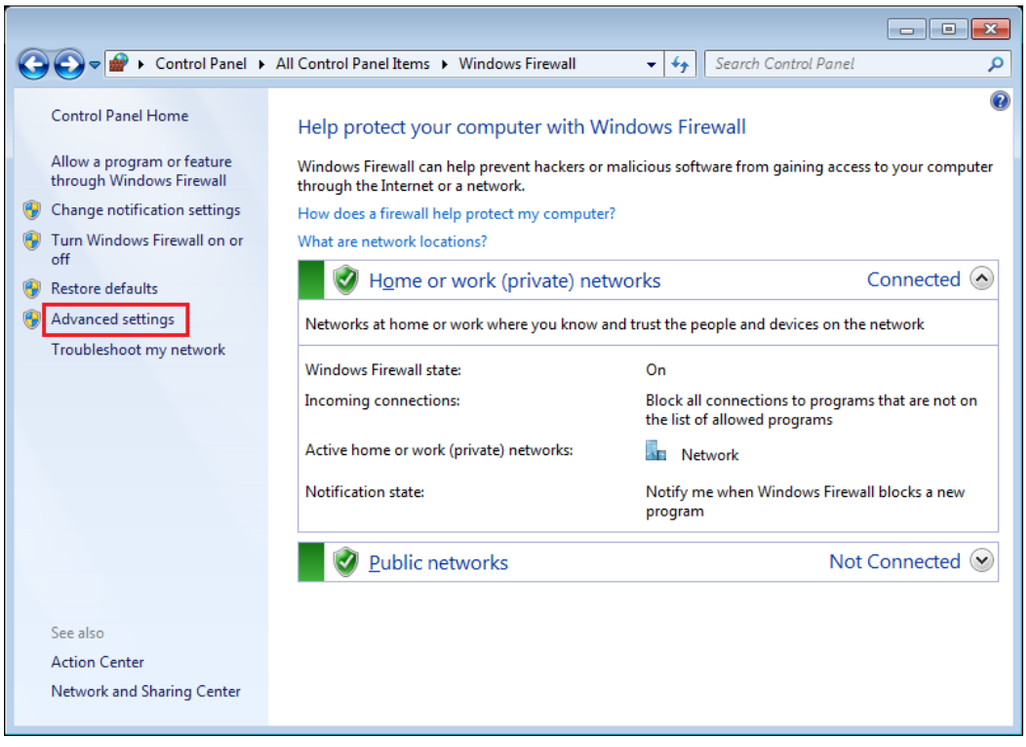


b. Select the [File and Printer Sharing] for setting as shown in the picture.

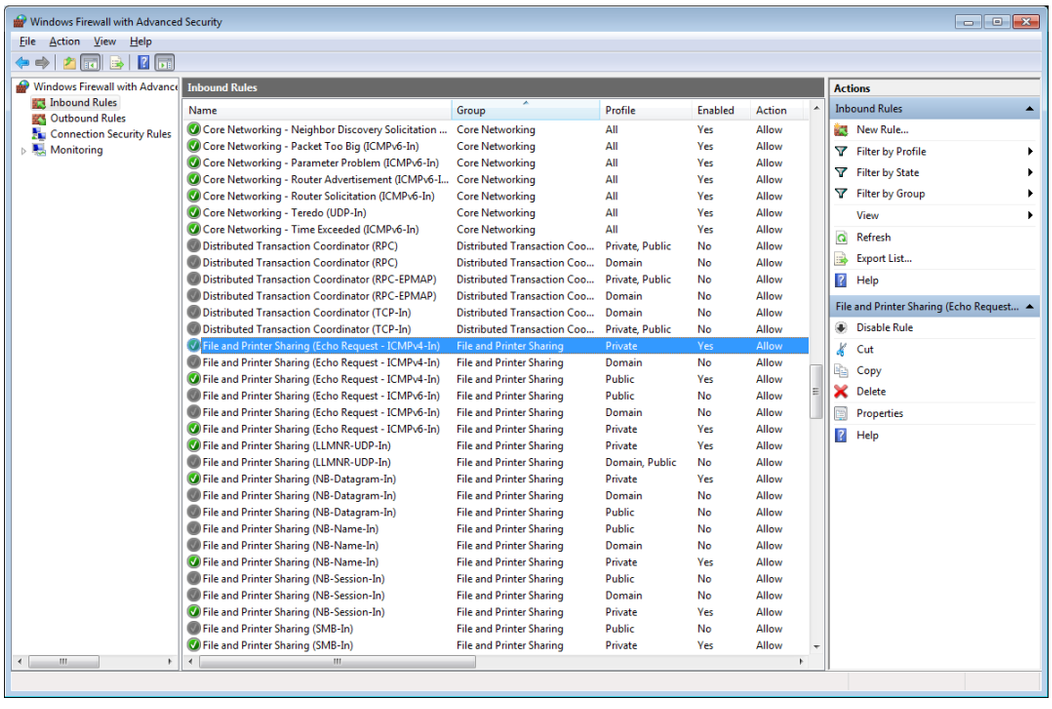


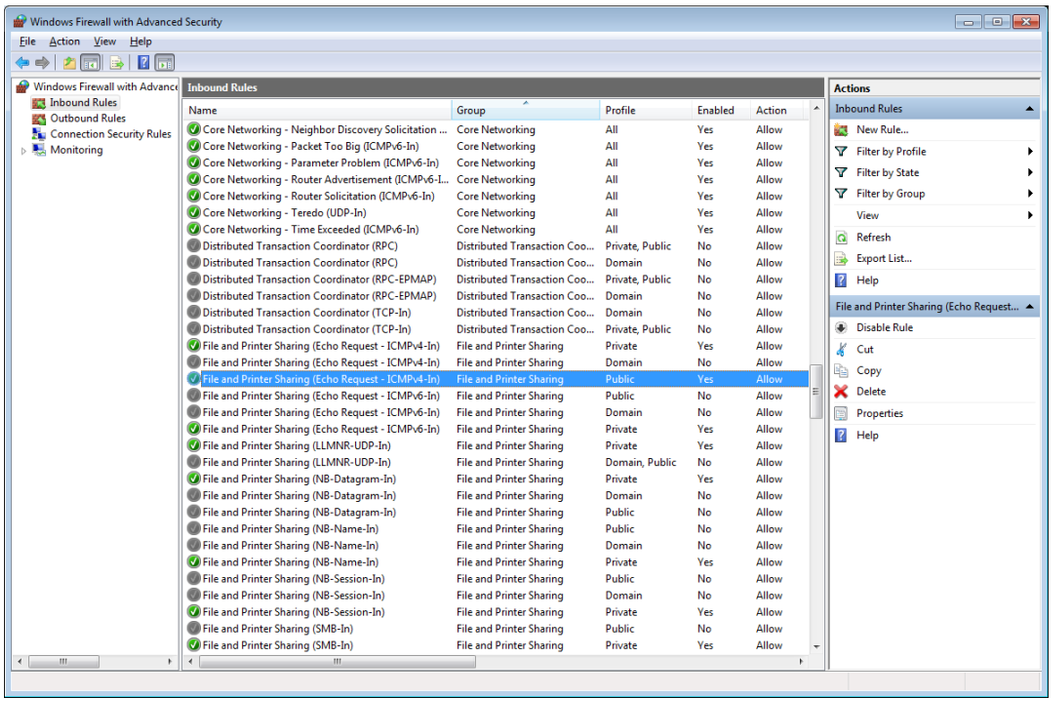
③ “ICMP” settings

a. Click [Advanced Settings] in the Windows Firewall setting screen.

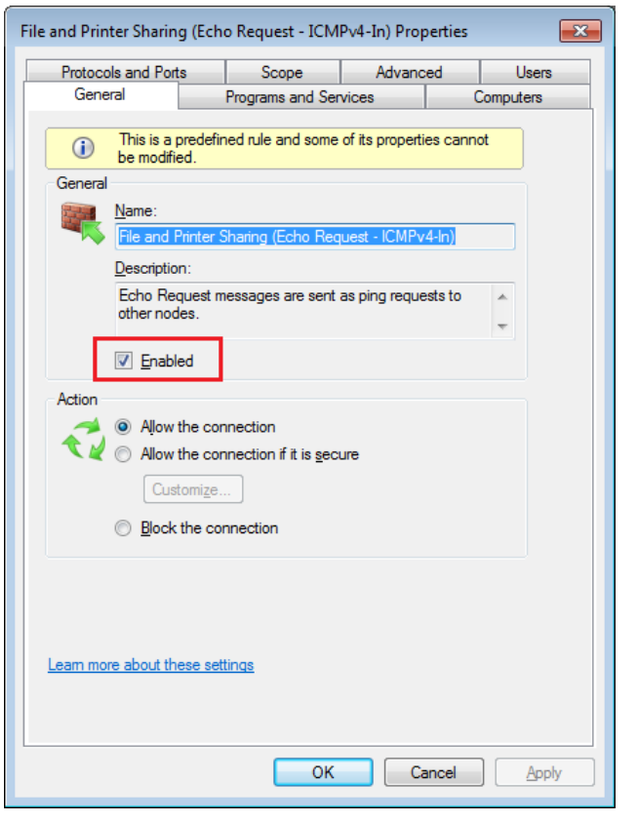


b. Double-click [Inbound Rules] → [File and Printer Sharing (Echo Request - ICMPv4-In)], Private, Public]



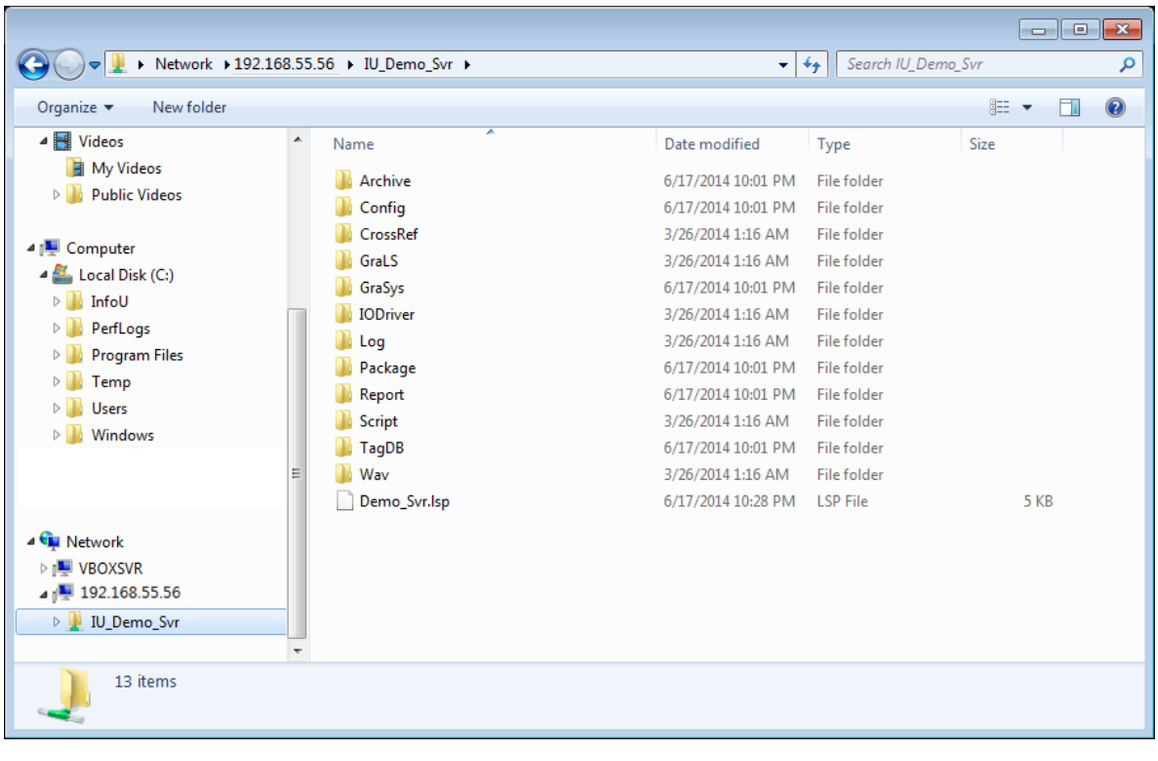


c. In the property window of [File and Printer Sharing (Echo Request - ICMPv4-In)], put a check
[General tab] → [General] → [Enable] options.

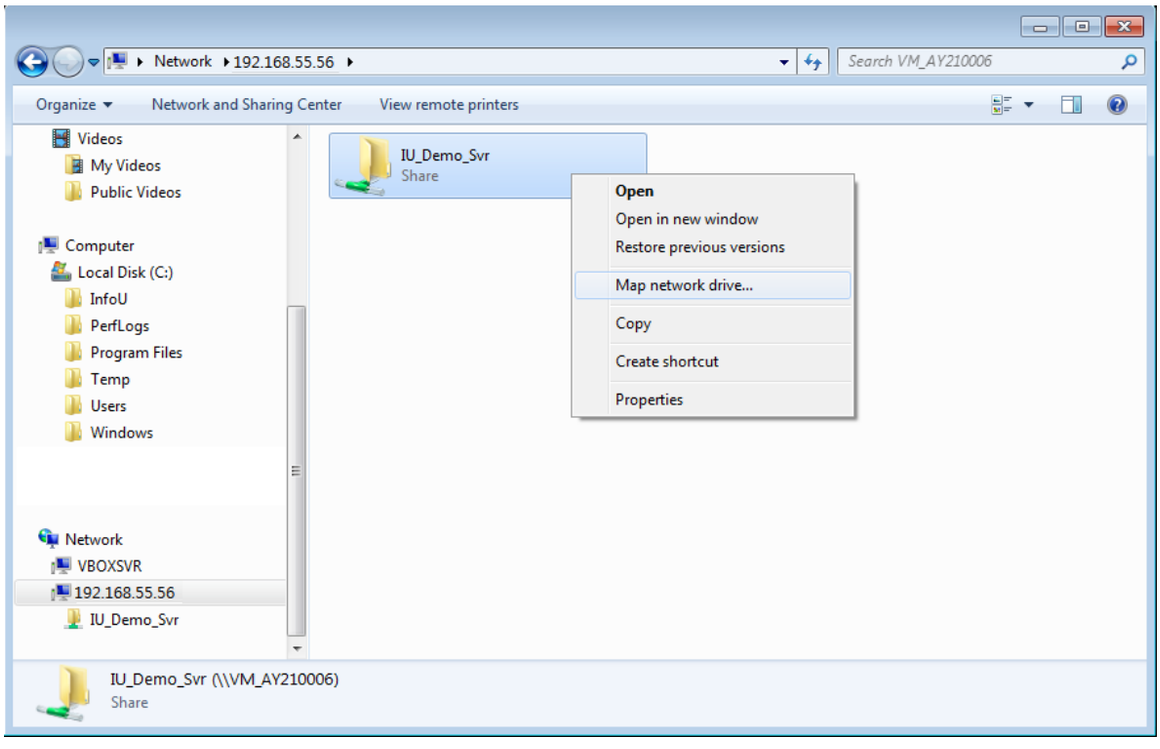


④ Setting the access to the shared folder of the computer acting as a partner

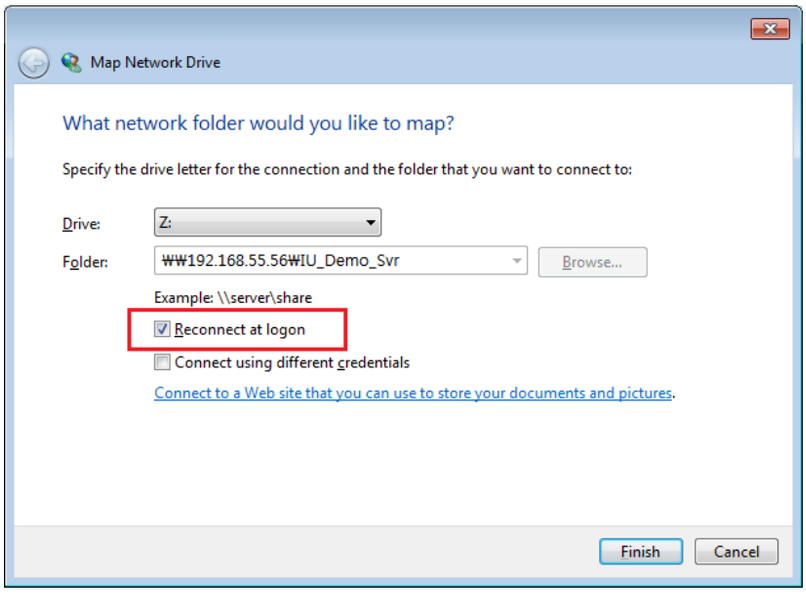
a. Check the shared folder of the computer through the explorer.



b. In order to access to the folder even during rebooting the computer, set the connection with the network driver. Then, you can access to the folder automatically when booting the computer.



c. In network driver connection settings, put a check [Reconnect at logon].

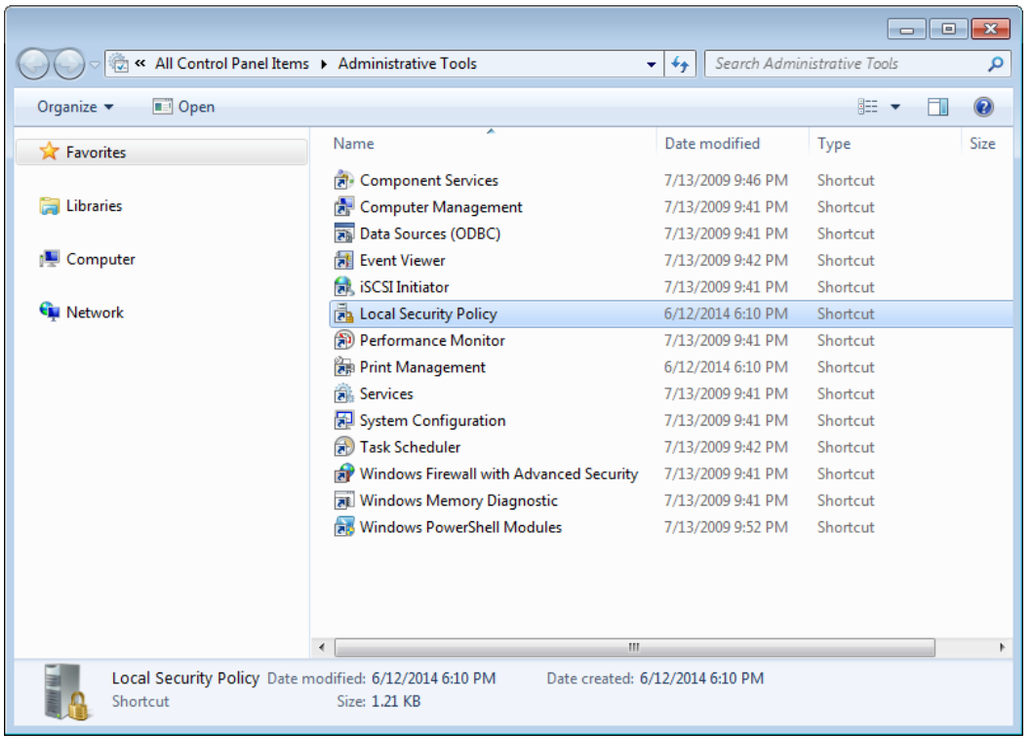


⑤ Configuration of Local Security Policy

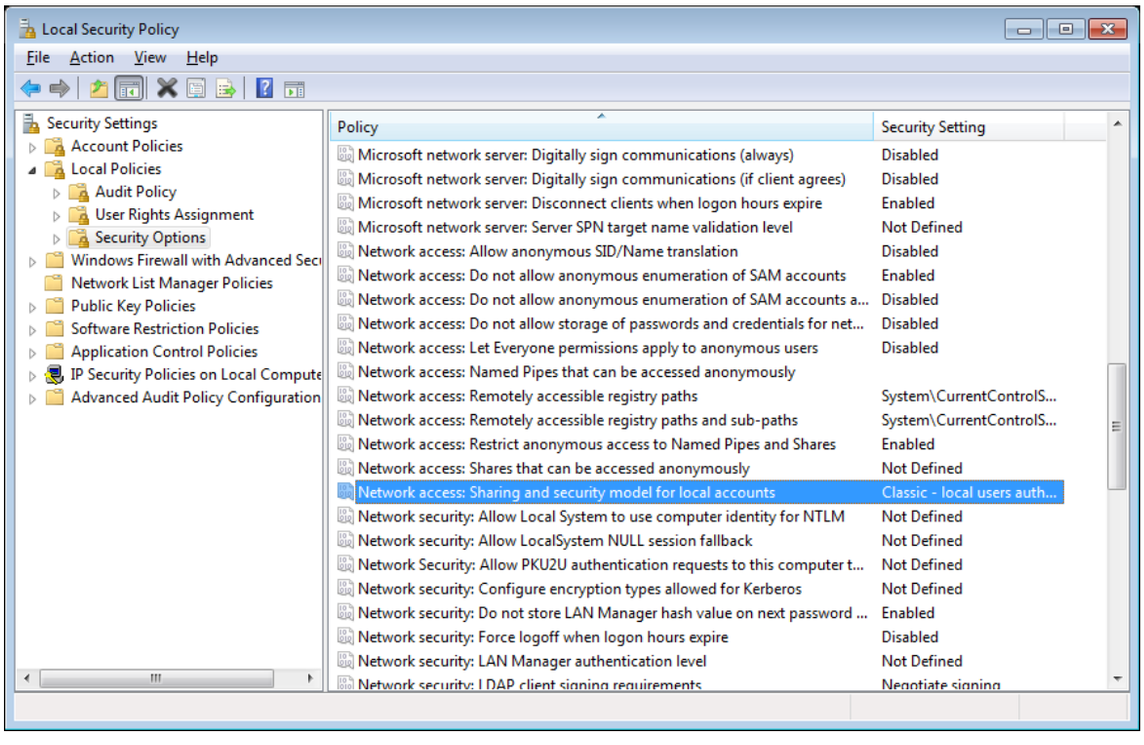
a. Click [Control Panel] → [System and Security] → [Administrative Tools]

b. Click [Control Panel] →[Administrative Tools]

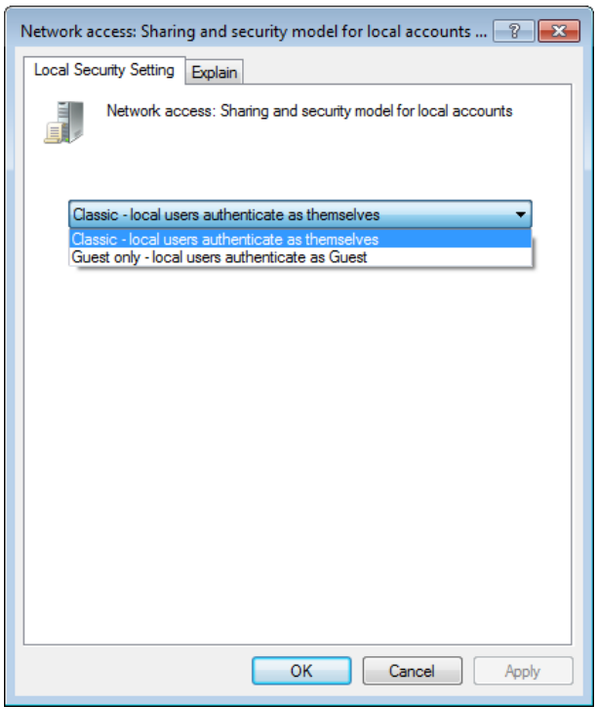
c. Double-click the [Local Security Policy]



d. Double-click [Network access: Sharing and security model for local accounts]



e. Select [General - Local users authenticate as themselves] in the [Local Security Setting] tab.

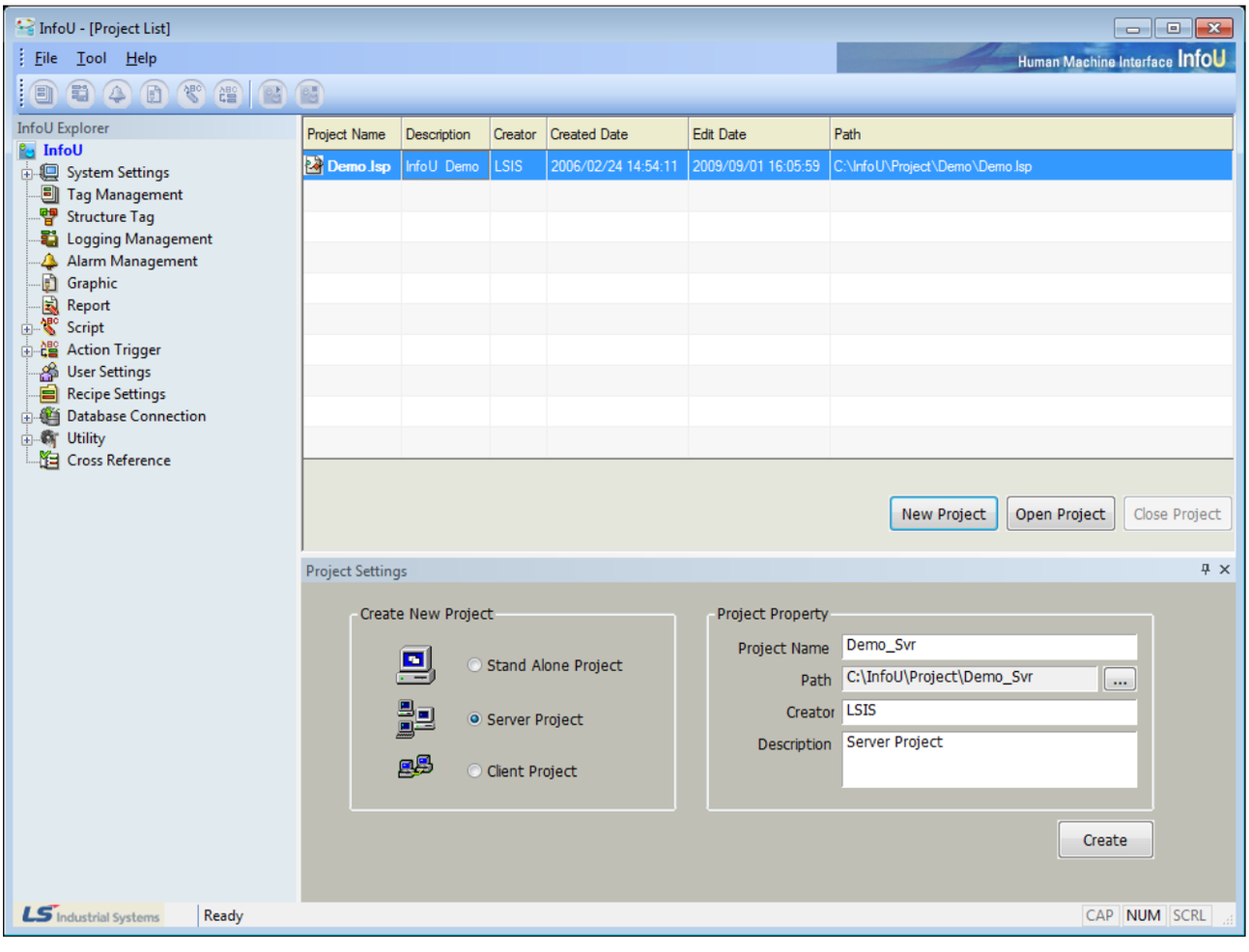


24.2 Redundancy Settings

24.2.1 Procedure

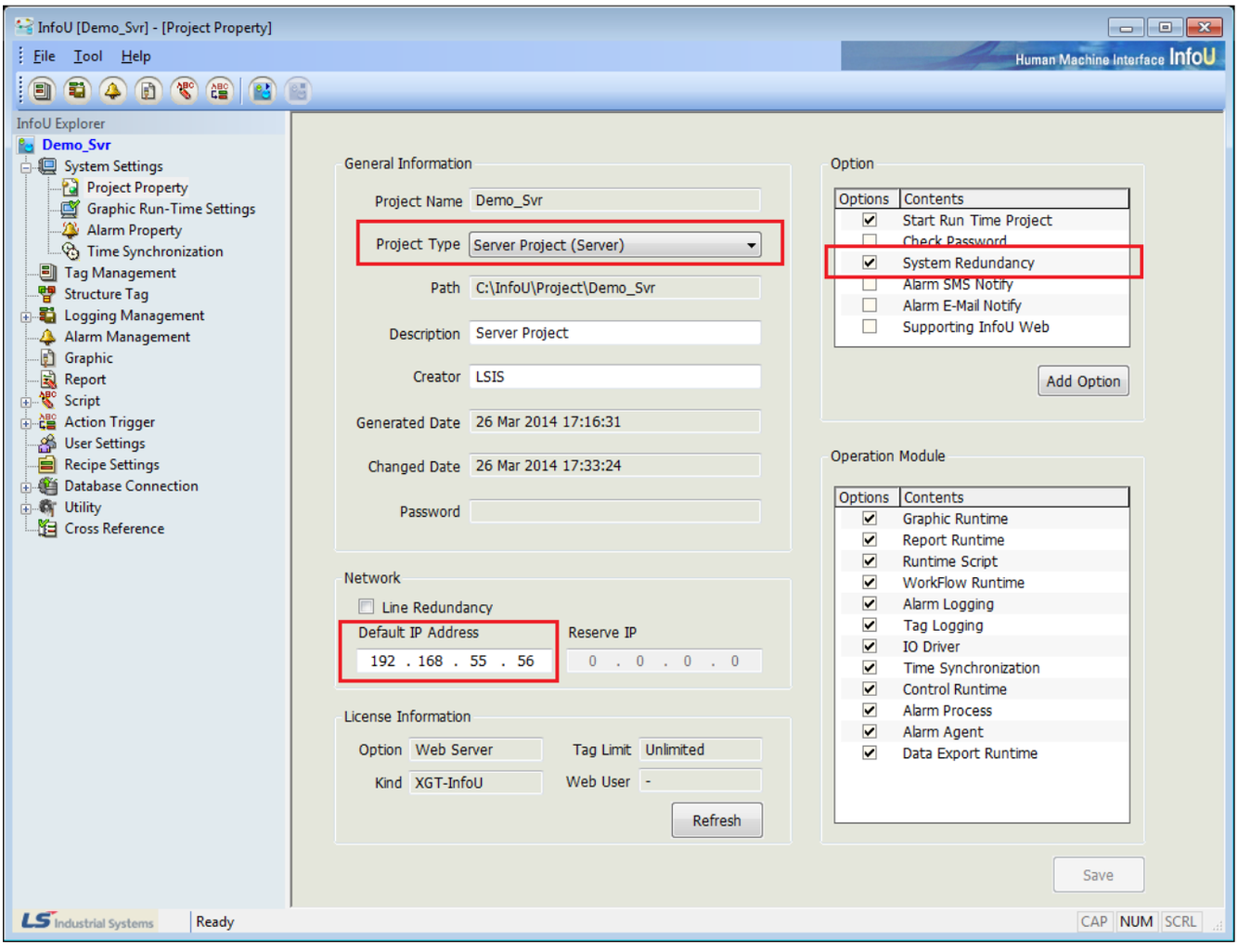
a. Set up the system network environment allowing data communication.
b. Create Project: Create a server project.
c. Select Redundancy Option: Click on 'Redundancy' option on the project Property settings.
d. Set up Redundancy: Save redundancy settings.
e. Duplicate Redundancy Project: Duplicate the redundancy project on the current server in the partner server. At this time, the settings of the partner server are automatically changed.
f. Check the project on the partner server: Check whether the redundancy settings are correctly duplicated and if any item (ex> serial port, etc) is differently set up from the master server, re-set up and save it.
g. Execute real-time running.

24.2.2 Create Server Project



a. When creating a new project, select 'Server Project' for its type.
b. Also, it is possible to change and use any existing Stand Alone project to a server project on the Project Property setting screen.

24.2.3 Project Property Settings



a. Enter the current computer's IP Address on Network.

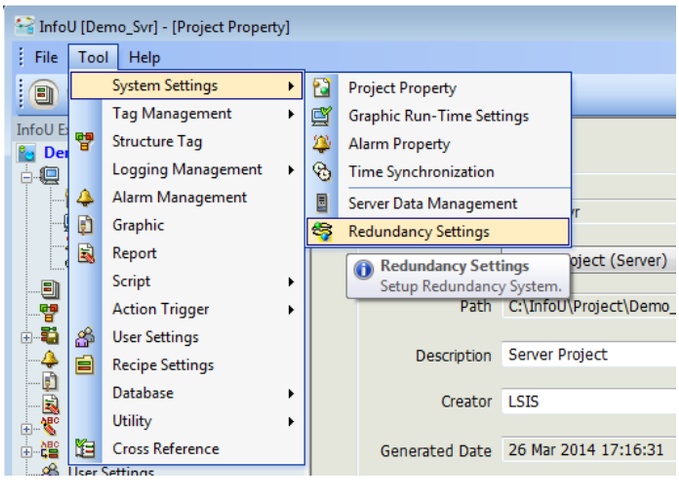
b. When using Line Redundancy, that is, when using two LAN cards, select 'Line Redundancy' and enter the reserve IP Address.

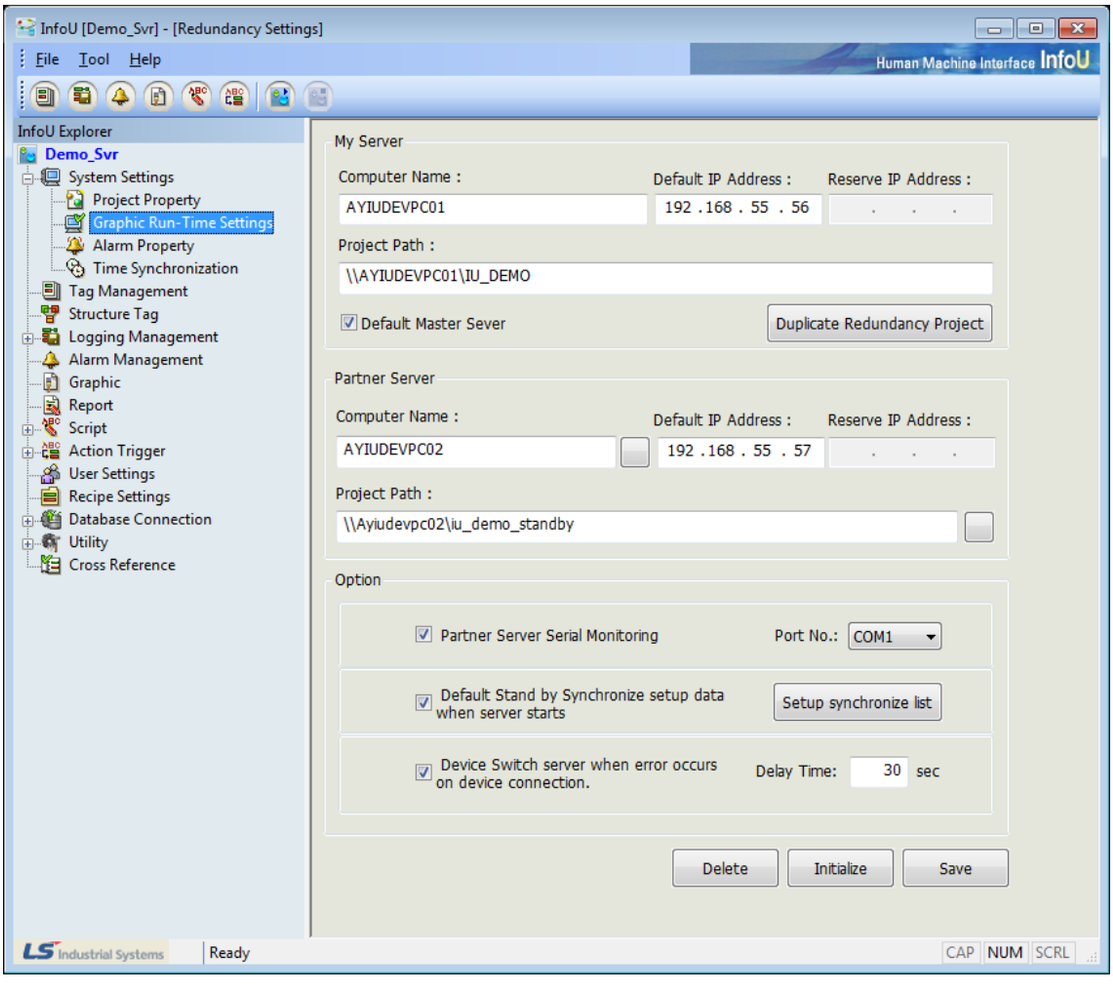
c. Select and save 'System Redundancy' on Option.

24.2.4 Redundancy Settings

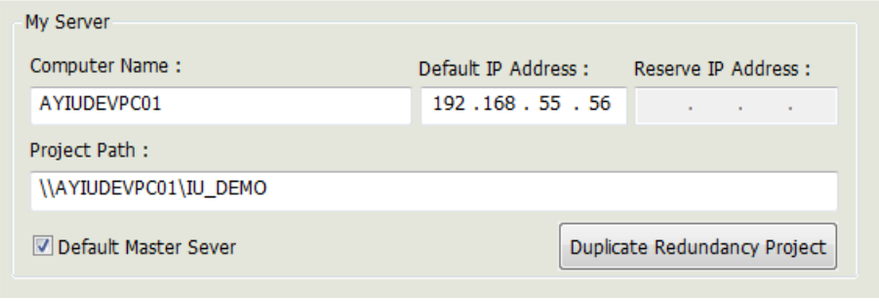
(1) Select Menu

Select 'Tool-System Settings-Redundancy Settings.'





(2) My Server



a. Computer Name: Enter a computer name for identification on the computer network environment.

b. Default IP Address: Enter the default IP Address.

c. Reserve IP Address: When line Redundancy option is used, enter the reserve IP Address.

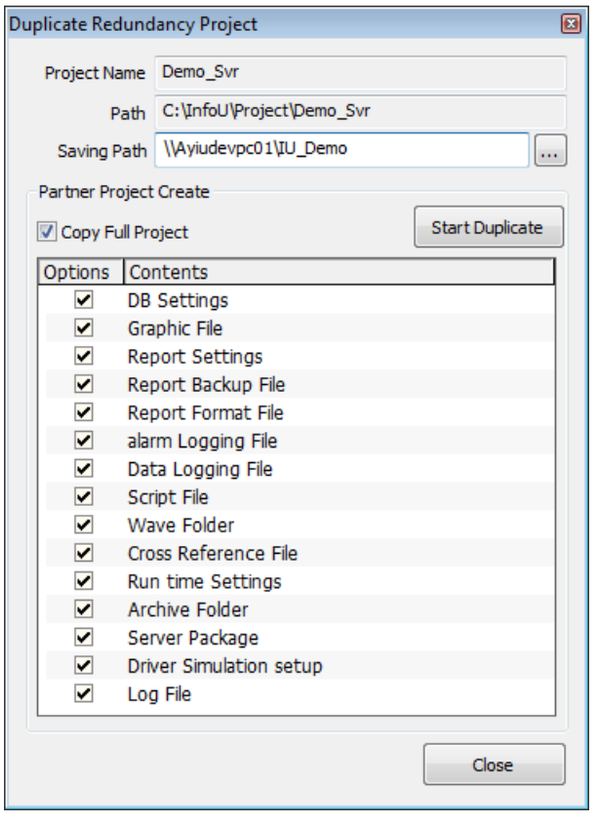
d. Project Path: Enter an accessible path of the current project on the network.

e. Default Master Server: Assign a role of the master server to 'My Server.'

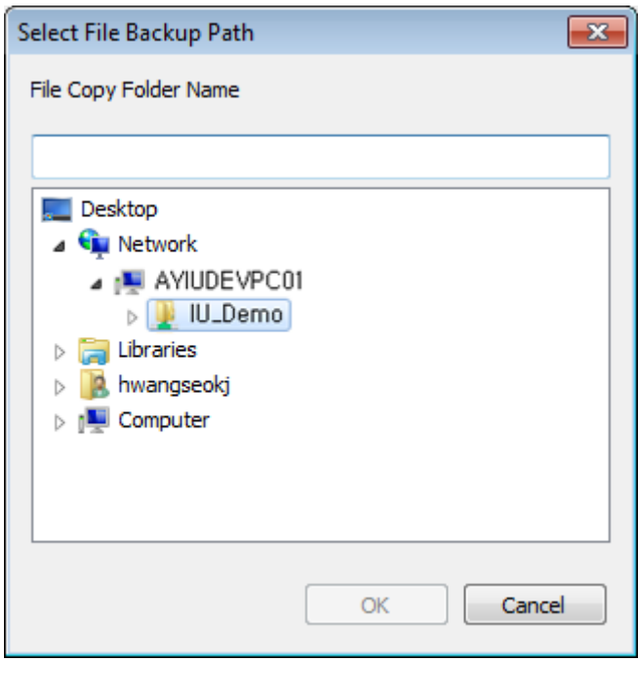
Notice

- ☞ When two servers run simultaneously, 'My Server' always runs as the master server.
- ☞ If this option is not set up, 'My Server' always runs as the Standby server when two servers run simultaneously but it runs as the Master server if 'Partner Server' does not run.
- ☞ This option shall be set up for only one of two servers.
- ☞ If this is set up for both of them or it is set up neither of them, errors may take place.

f. Duplicate Redundancy Project: A screen on which the current project can be duplicated in the partner server is displayed.



- Project Name: The current project name is displayed.
- Path: The local Project Path of the current project is displayed.
- Saving Path: Input a project path on which the redundancy project is saved in the partner server Click on the button to display the 'Select File Backup Path' screen and select a saving location.



- Copy Full Project: All of the settings various engineering files and logging data files on the current project path are copied.

Notice

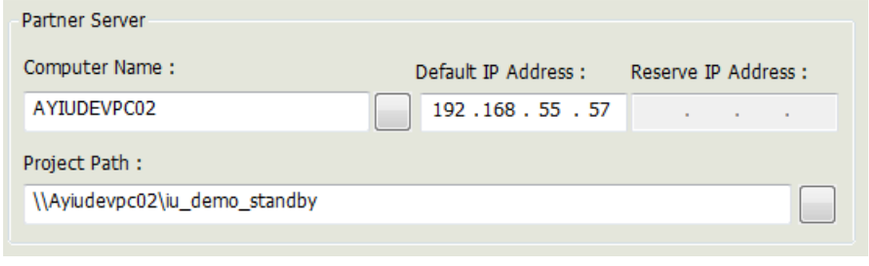
- ☞ At this time, any data other than on the project path can not be duplicated.
- ☞ After engineering works, it is essential to duplicate and synchronize settings between projects before starting another running.
- ☞ After this option (Copy Full Project) is cancelled, individual items can be selected and copied.

- Start Duplicate: Duplication is executed. It may take a lot of time depending on the file capacity.

Notice

- ☞ In case that duplication has been executed before, only the current files are updated.

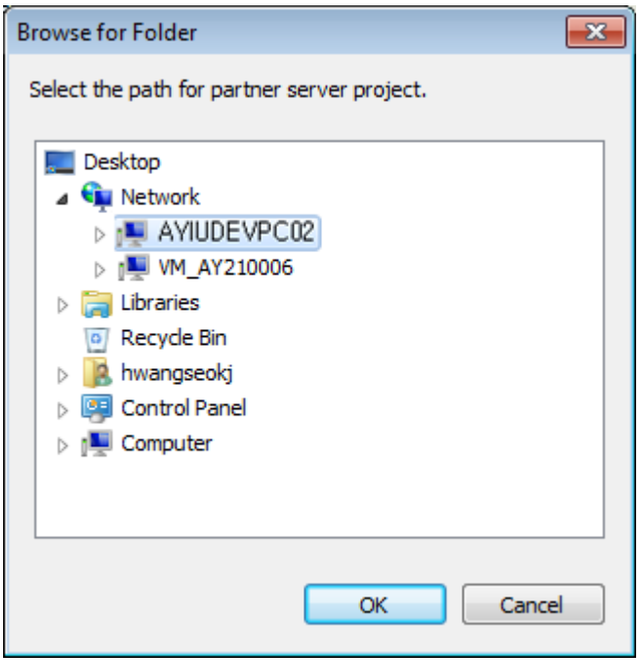
(3) Partner Server



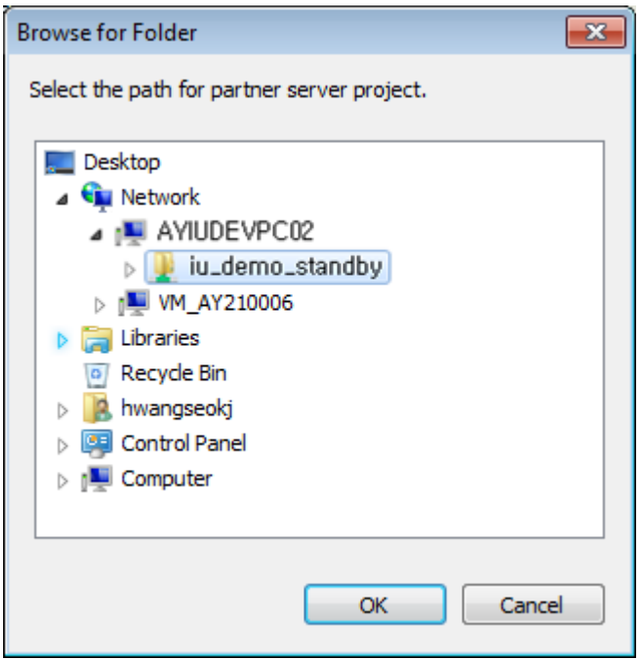
a. Computer Name: Enter a computer name for identification on the computer network environment.

b. Default IP Address: Enter the default IP Address.

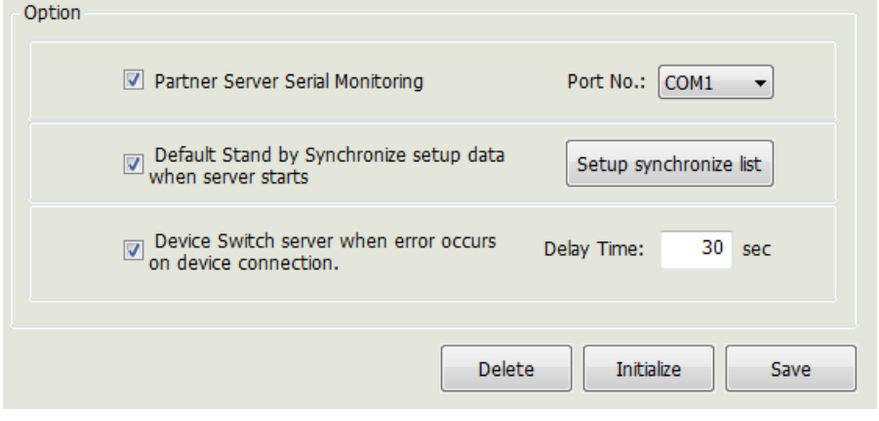
c. Reserve IP Address: When line Redundancy option is used, enter the reserve IP Address. Click on the button to select a computer on the network.



d. Project Path: Enter an accessible path of the current project on the network. Click on the button to display the Select File Backup Path screen and select a saving location.



(4) Option

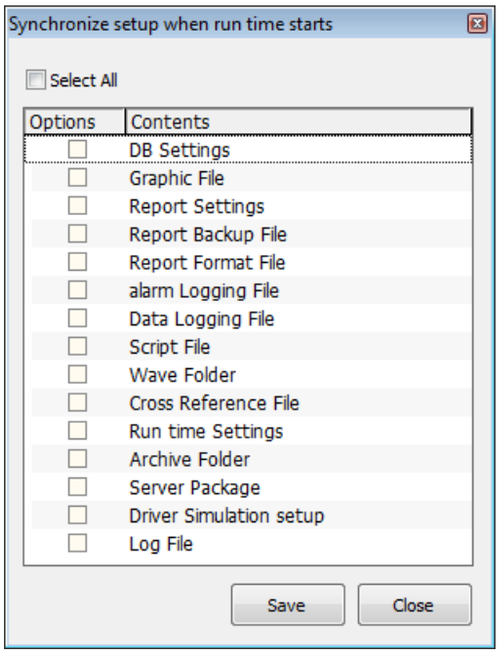


a. Partner Server Serial Monitoring: Connect a serial cable to monitor the Partner Server.

b. Port No: Show the serial port number subject to connection.

c. Default Stand by Sychronize setup data when server starts: The Standby server starts after the settings of the Master server is duplicated.

d. Setup synchronize list: When starting, a screen to select synchronization items is displayed.



- Select All: All of the sub items are selected. After this option (Select All) is cancelled, individual items can be selected and copied.
- Save: Save the settings.
- Close: Close the screen.

Notice

☞ It may take a lot of time to start the Standby server depending on the file capacity.

☞ Since most of the logging files have a large capacity, it is better to select basic settings only.

e. Delay Time: Device Switch server when error occurs on device connection: When any communication error occurs, switching action is performed.

f. Delete: Delete redundancy settings.

g. Initialize: Return the settings to the previous ones.

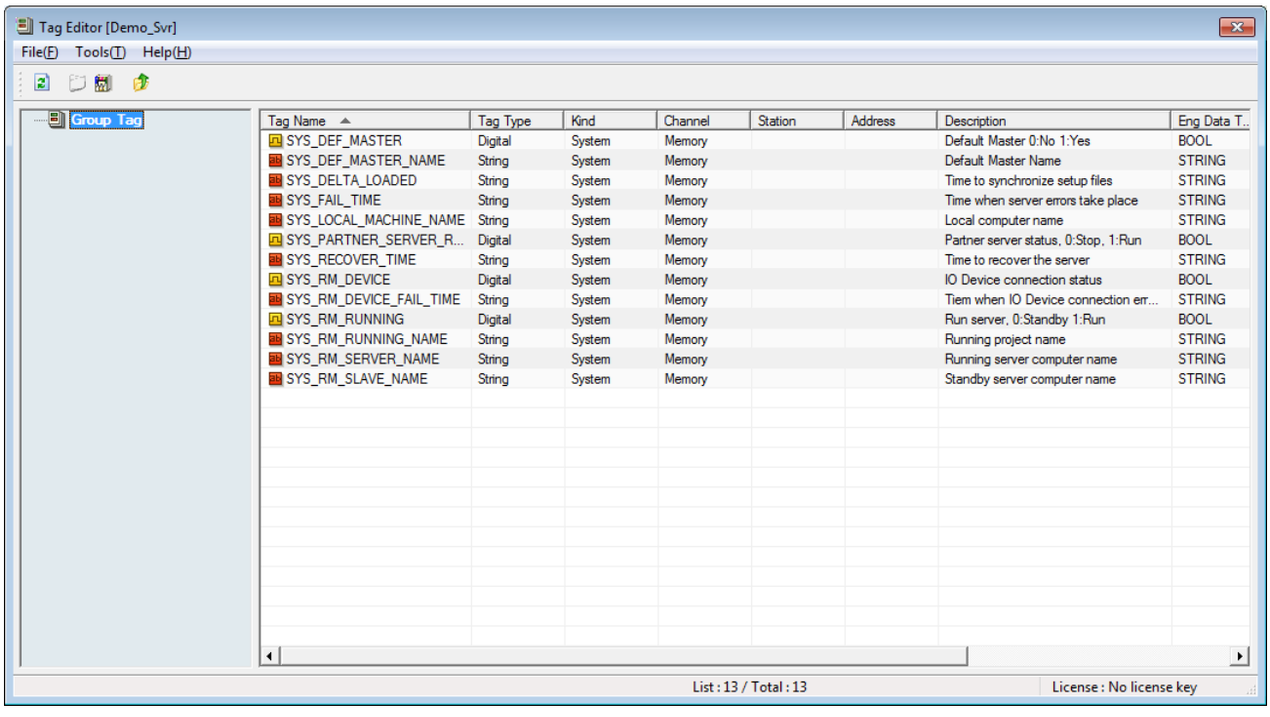
h. Save: Save the settings.

24.2.5 RunTime Operation

(1) System Tag

When running in the redundancy mode, a tag that shows the system status during the real-time running is created to allow the user to monitor the current redundancy running status.

1) System tags and status codes related to redundancy

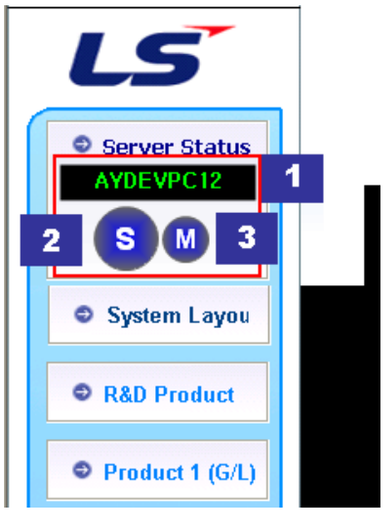


The above tag browser screen during the real-time running shows the status of each tag value as follows:

Tag Name	Description	Type
SYS_DEF_MASTER	Default Master server, 0:No 1:Yes	Digital
SYS_DEF_MASTER_NAME	Default Master server name	String
SYS_DELTA_LOADED	Time to synchronize setup files	String
SYS_FAIL_TIME	Time when server errors take place	String
SYS_LOCAL_MACHINE_NAME	Local computer name	String
SYS_PARTNER_SERVER_RUN	Partner server status, 0:Stop 1:Run	Digital
SYS_RECOVER_TIME	Time to recover the server	String
SYS_RM_DEVICE	IO Device connection status (All) 0: normal 1: fail	Digital
SYS_RM_DEVICE_FAIL_TIME	Time when IO Device connection errors take place	String
SYS_RM_RUNNING	Running server, 0:Standby 1:Run	Digital
SYS_RM_RUNNING_NAME	Running project name	String
SYS_RM_SERVER_NAME	Running server comput name	String
SYS_RM_SLAVE_NAME	Standby server comput name	String

2) Status Dispaly

By using the above system tags, each tag's status can be displayed on the graphic screen.



- [1]: Running Server computer name
- [2]: Current server's status and run mode
- [3]: Partner server's status and run mode

24.3 Client-Server Settings

24.3.1 Procedure

a. Set up the system network environment allowing data communication.

b. Create Server Project: Create a server project.

c. Create Server Data File: Create a server data file that saves information on IP Address, Redundancy Information, Joint folder name to connect to the client server project.

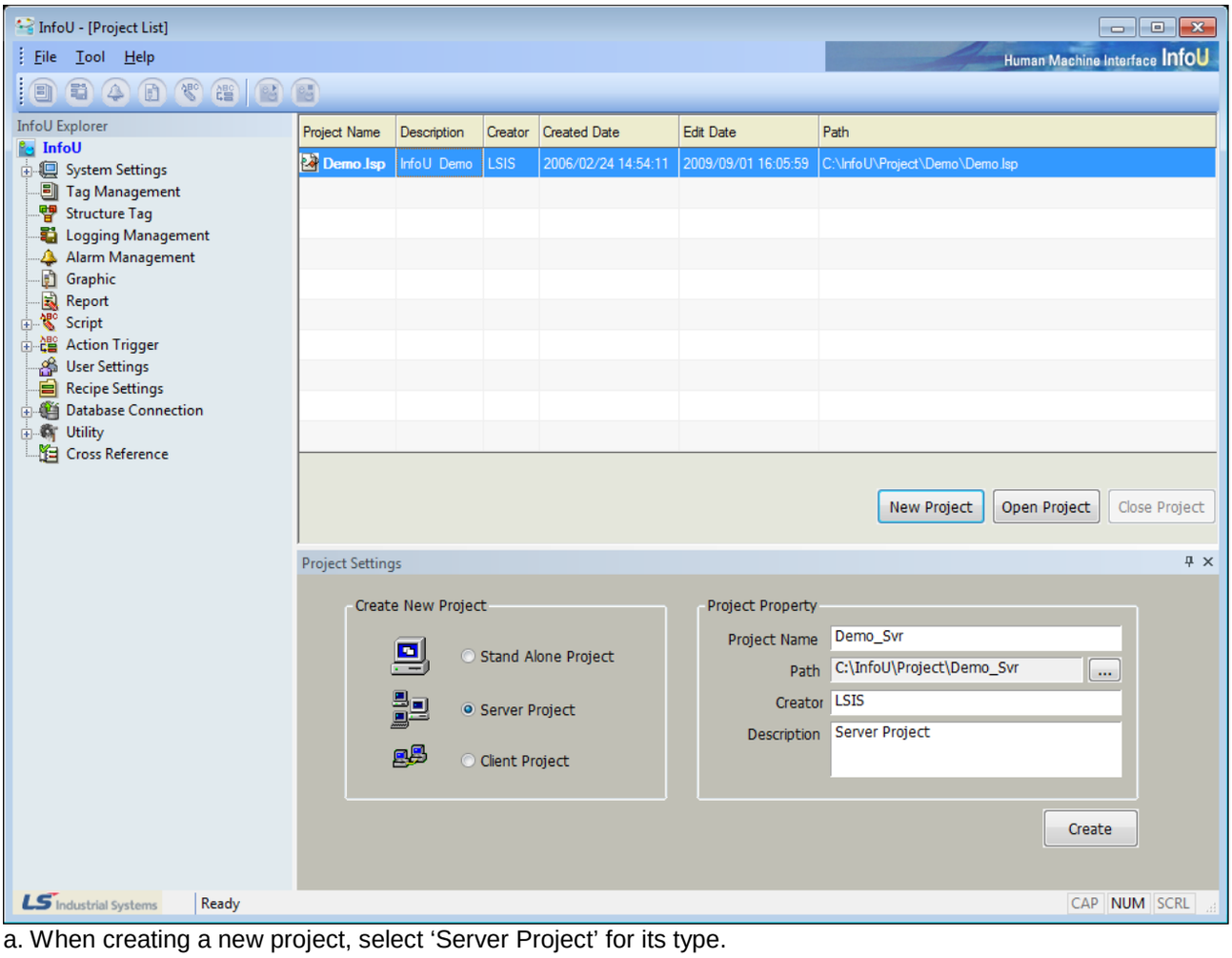
e. Create Client Project: Create a client project.

f. Load Server Data File: Load the server data file to connect from the client project.

g. Execute server real-time running.

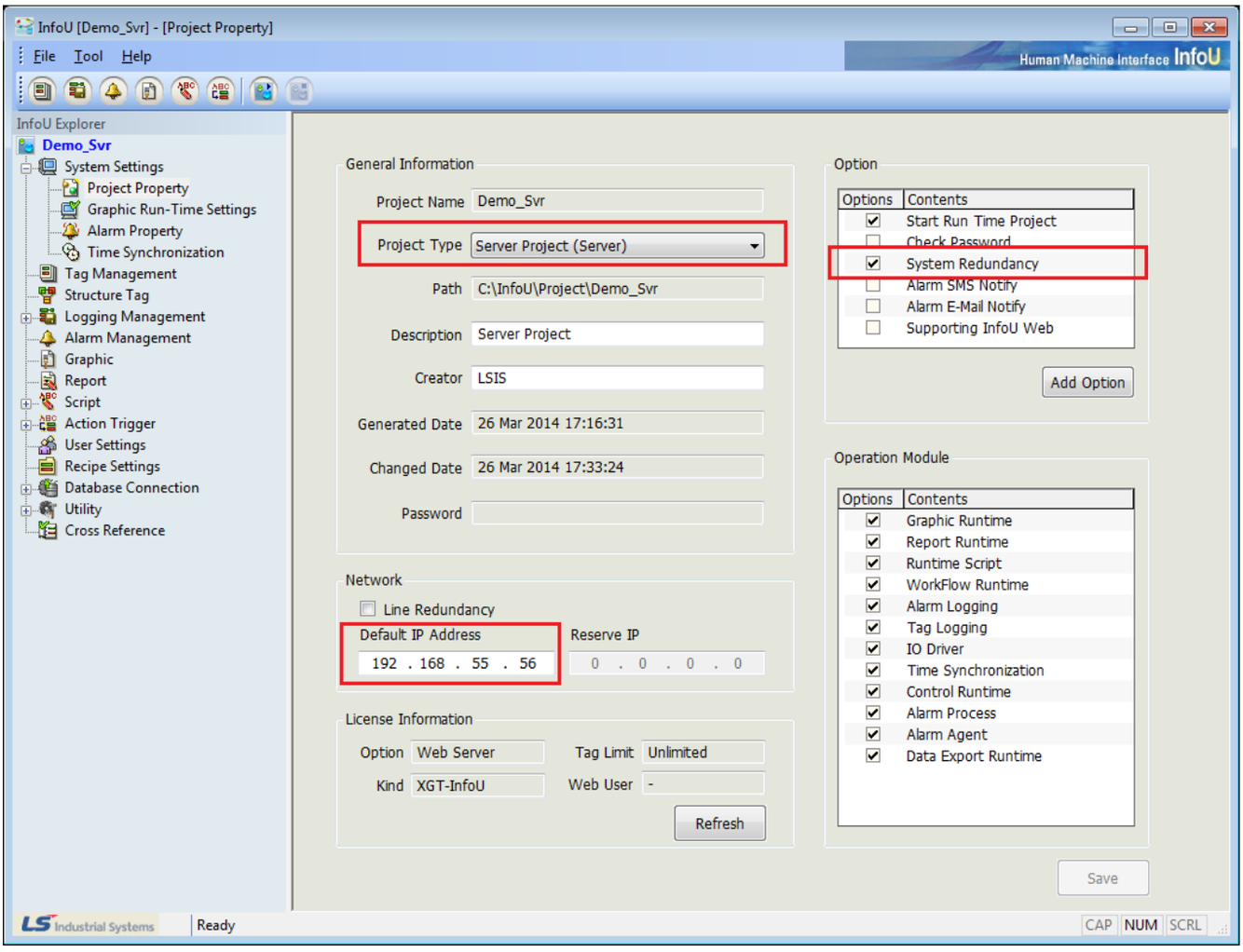
h. Execute client real-time running.

24.3.2 Server Project Settings



a. When creating a new project, select 'Server Project' for its type.

b. Also, it is possible to change and use any existing Stand Alone project to a server project on the Project Property setting screen.

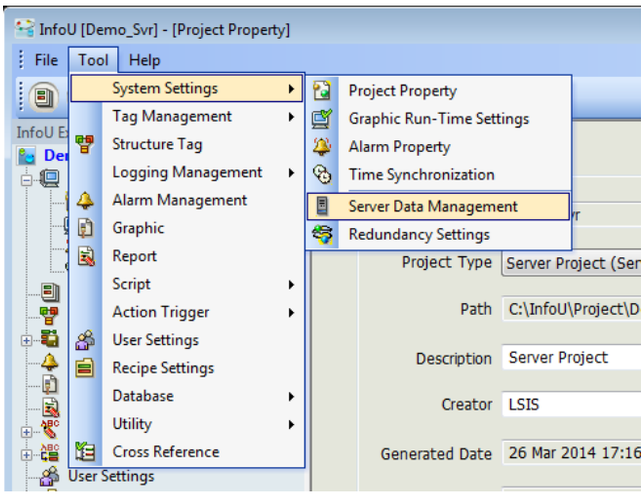


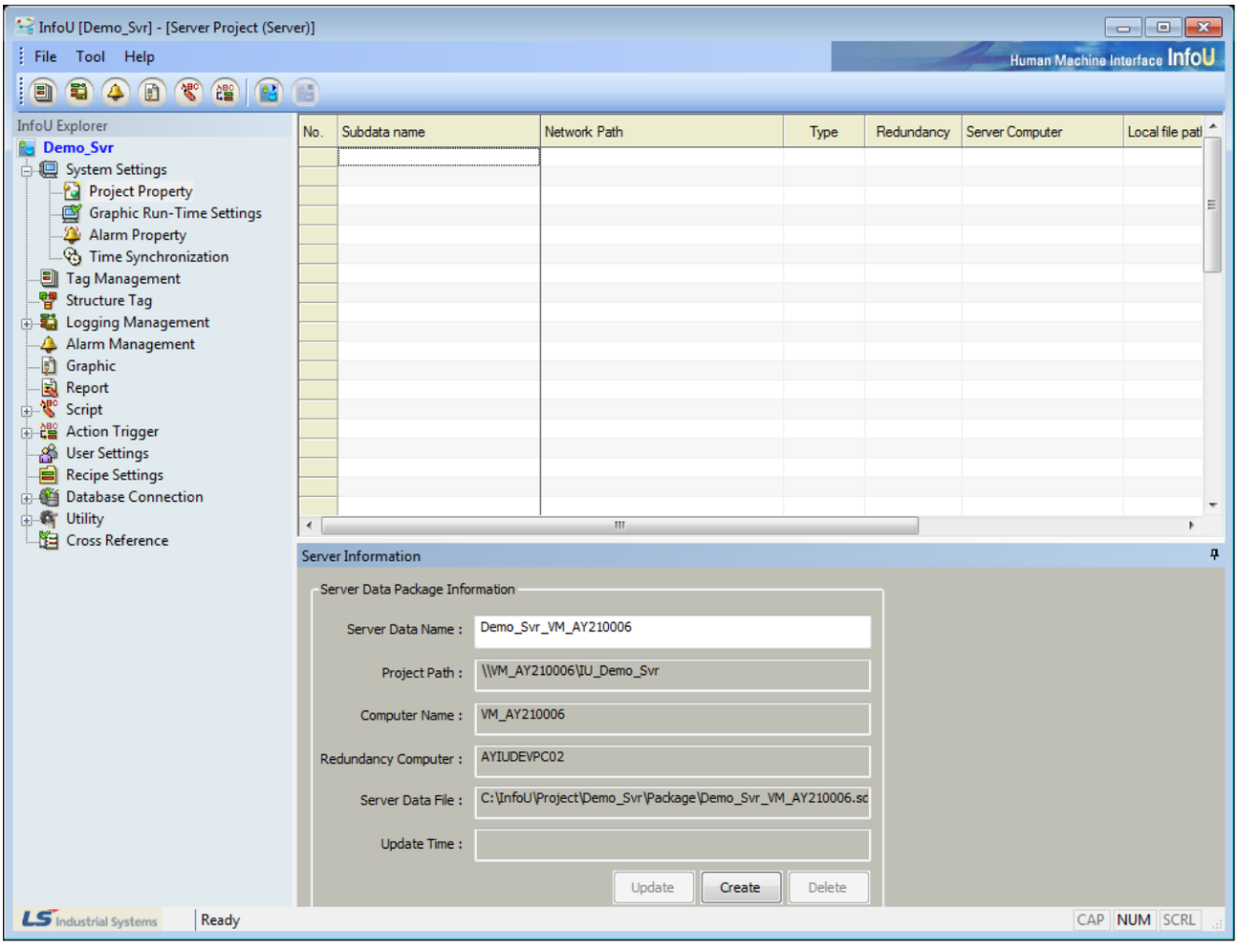
c. Enter the current computer's IP Address on Network.

d. When using Line Redundancy, that is, when using two LAN cards, select 'Line Redundancy' and enter the reserve IP Address.

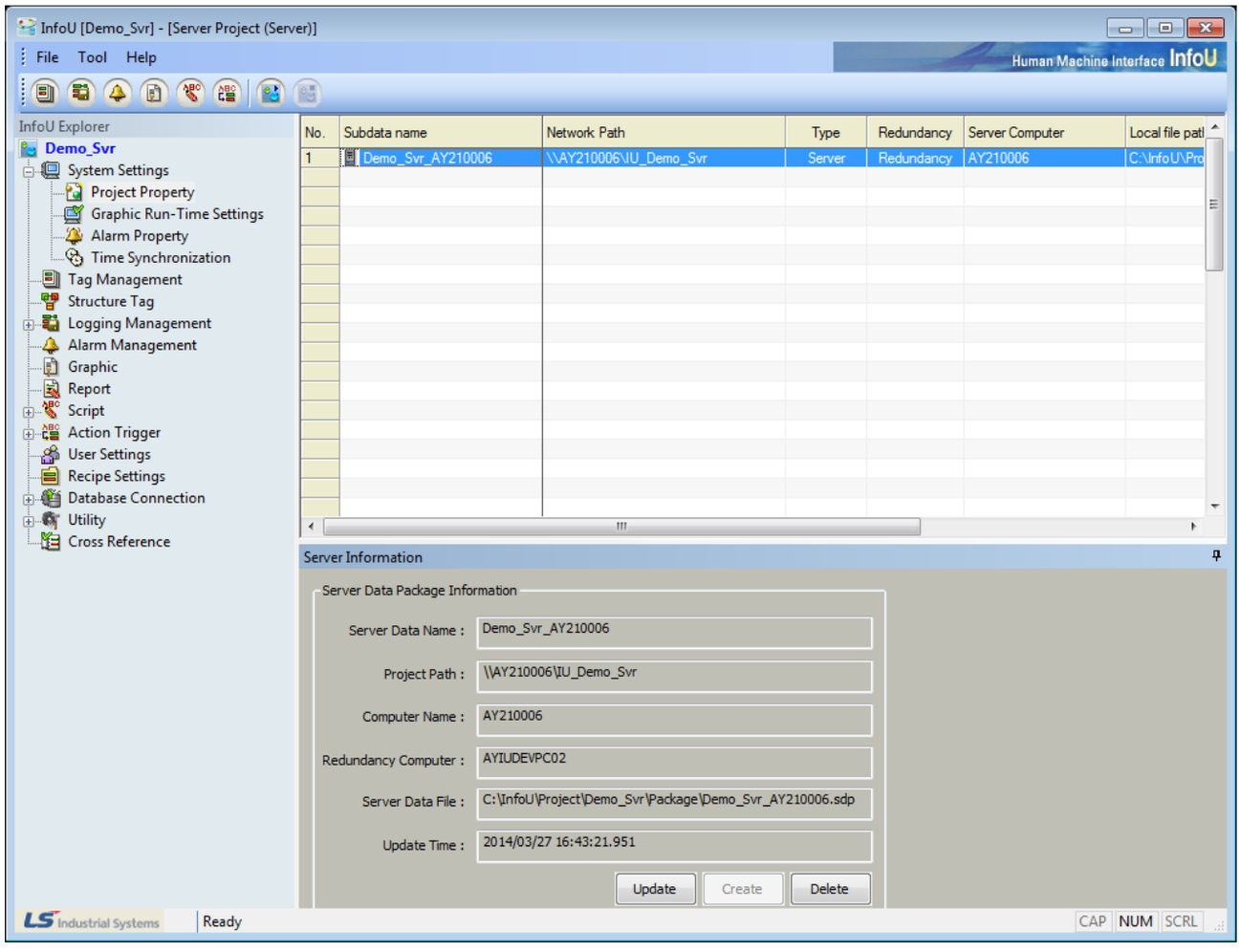
e. If you want to configure the duplex server system, put a check [Option] → [System redundancy]. When you configure the standalone system, do not put a check [Option] → [System redundancy].

f. Select 'Tool-System Settings-Server Data Management.'





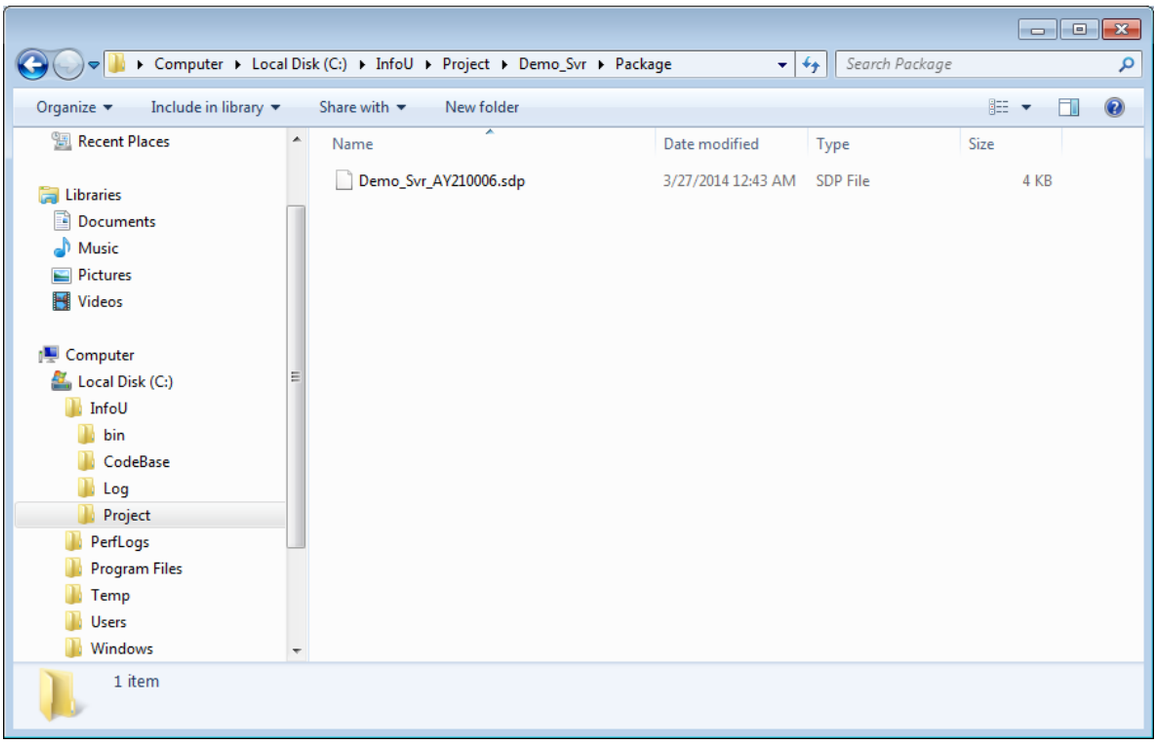
g. In case that any server data has not been saved, a server data name can be entered and the rest items are displayed automatically and the ‘Create’ button becomes active.



h. If “Create’ button is clicked, the server data file is created and the associated information is displayed on the screen.

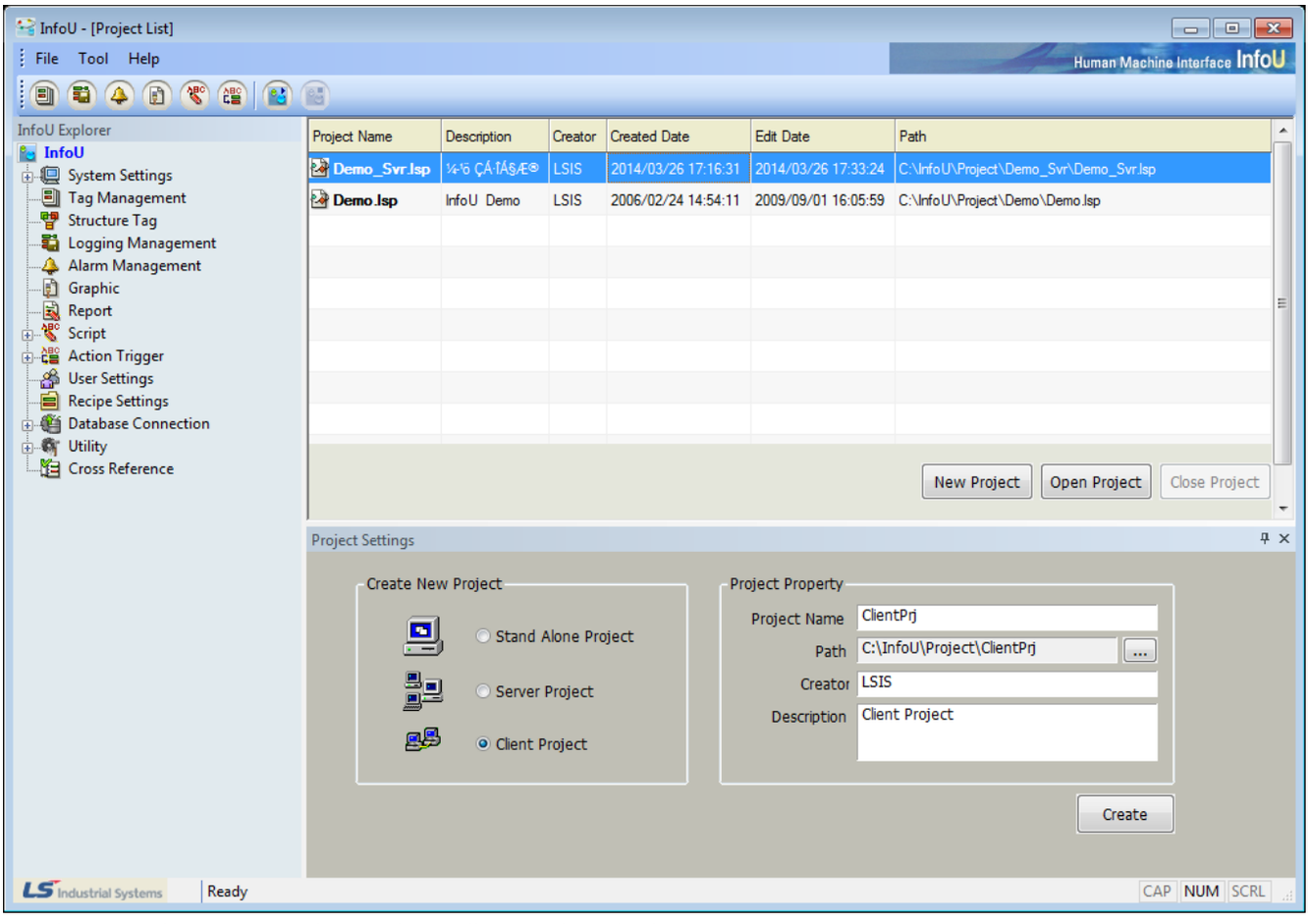
i. If ‘Update’ button is clicked, the changed engineering settings are updated and saved.

j. If ‘Delete’ button is clicked, the server data file is deleted

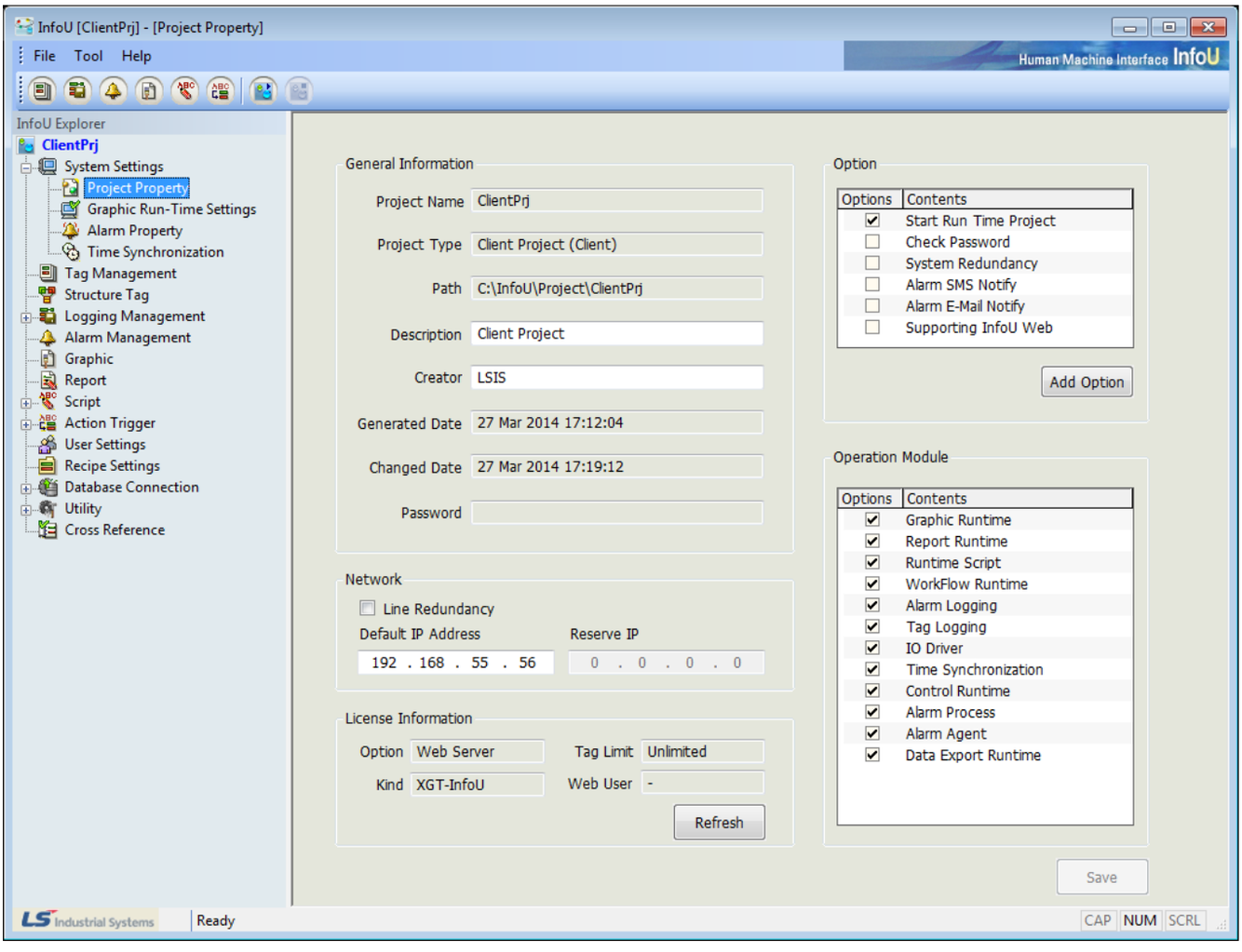


k. The file is created in “Package” among the project folders. At this time, extension “sdp” is used.

24.3.3 Client Project Settings



a. When creating a new project, select ‘Client Project’ for its type

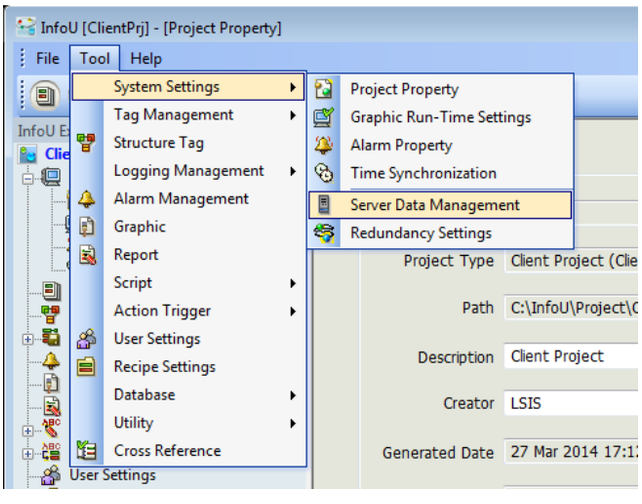


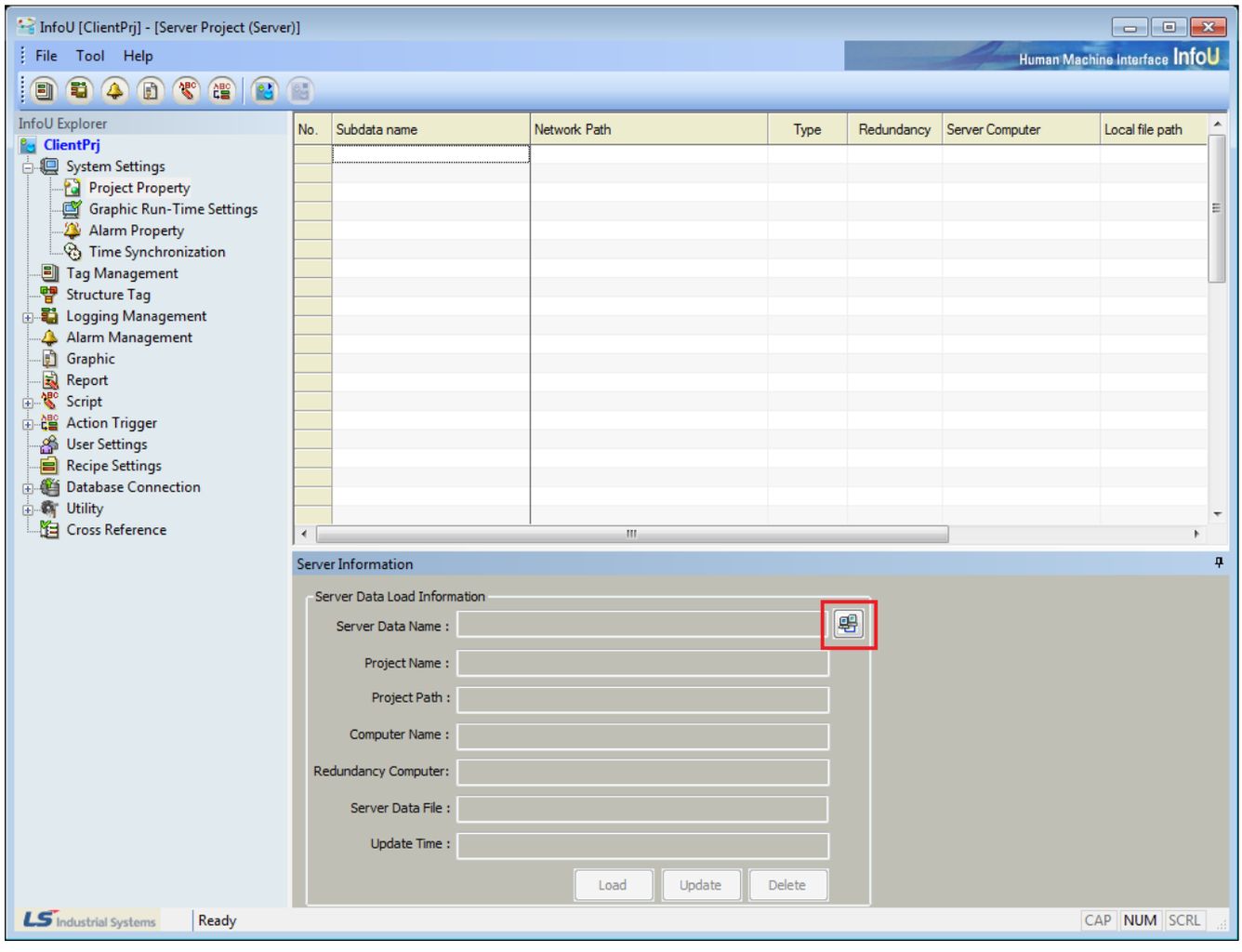
b. The client project type can not be changed.

c. Enter the current computer's IP Address on Network.

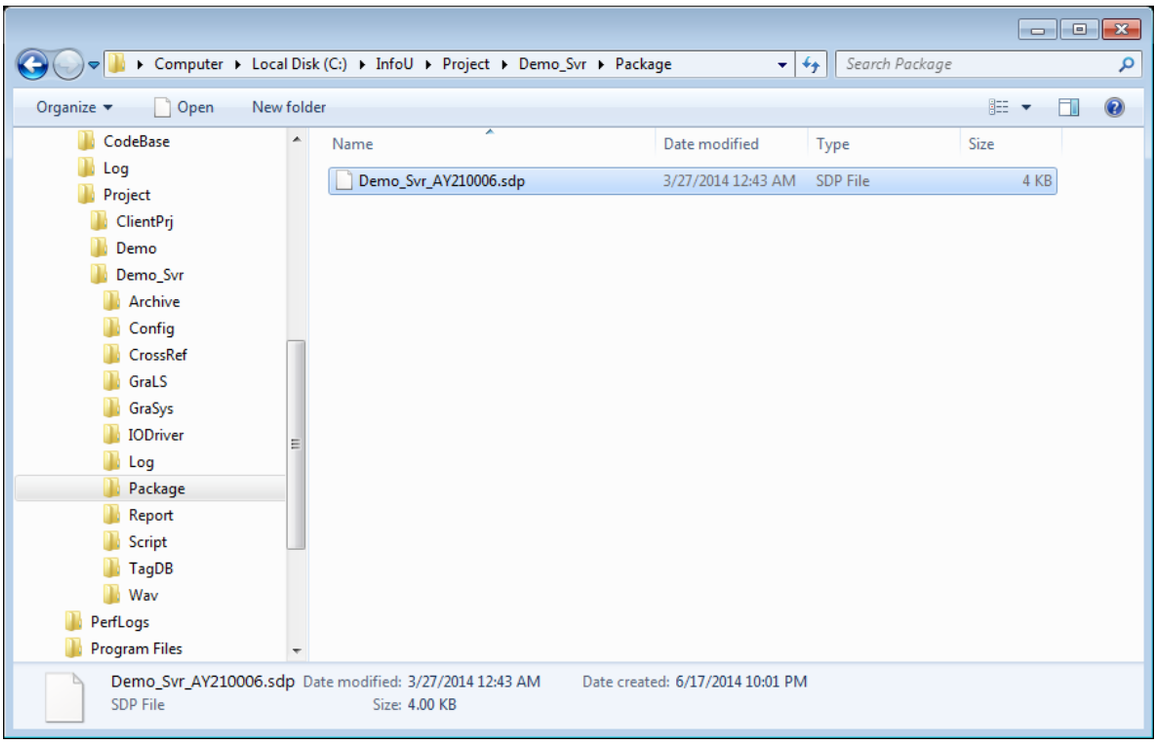
d. When using Line Redundancy, that is, when using two LAN cards, select 'Line Redundancy' and enter the reserve IP Address.

e. Select 'Tool-System Settings-Server Data Management.'

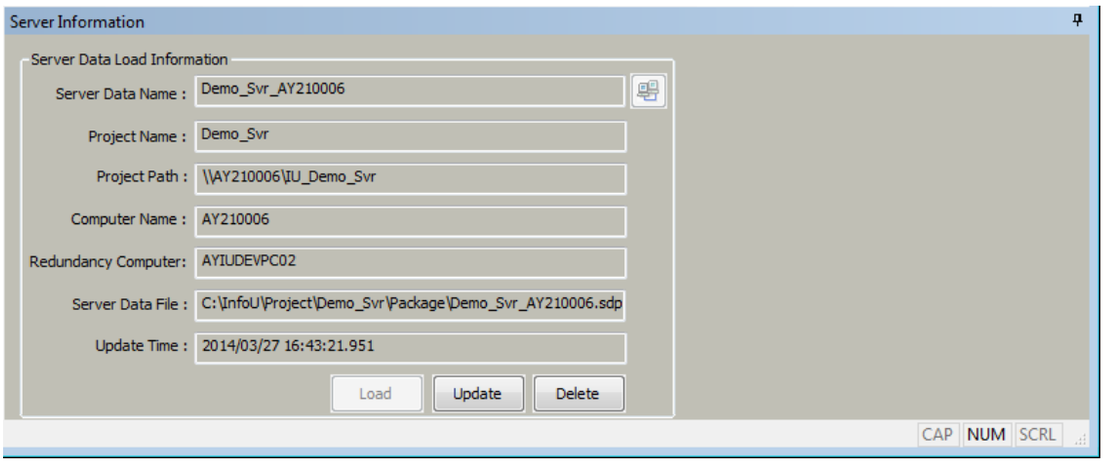




f. In case that any server data has not been saved, any information is not displayed and a button is activated to explore server data files.

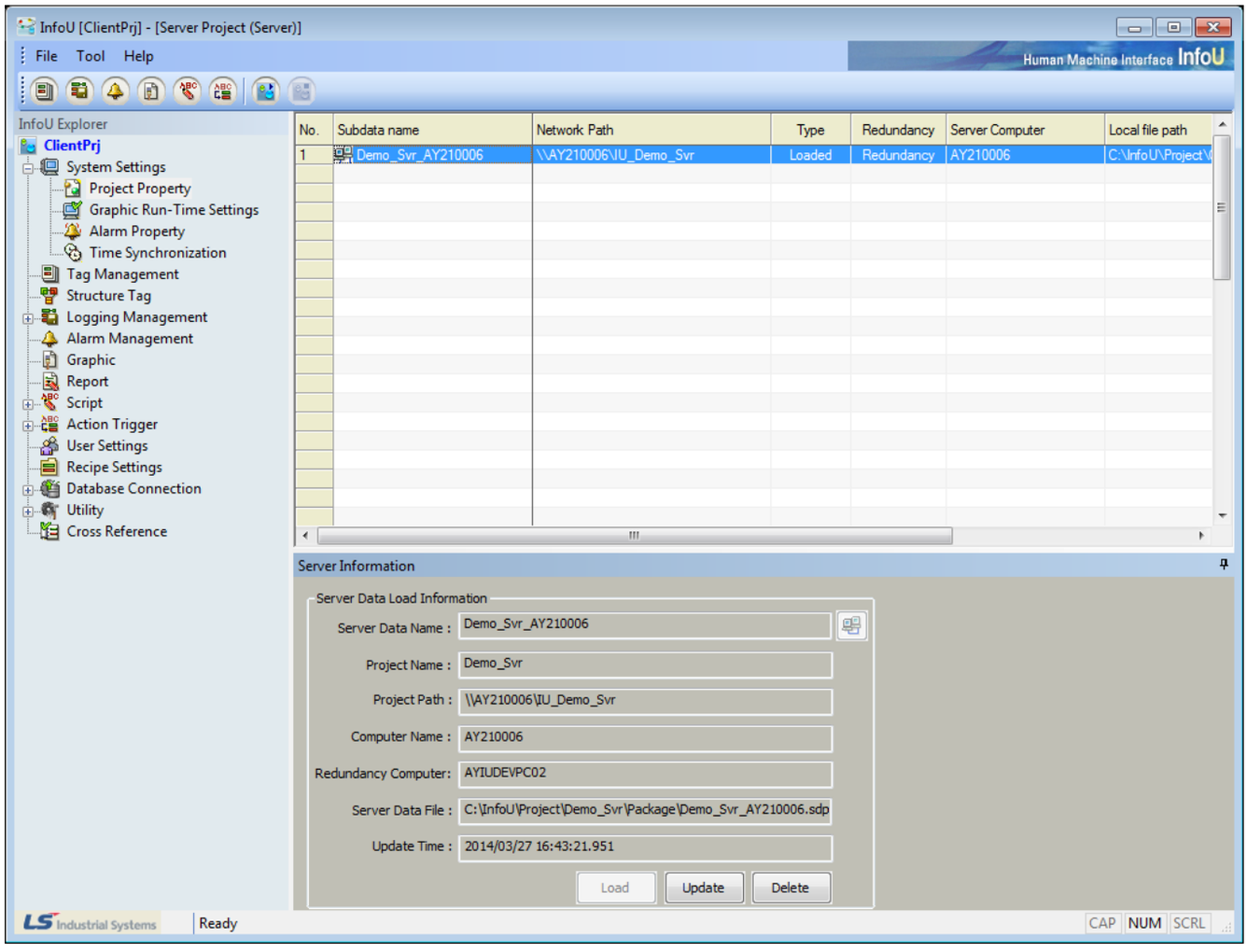


g. Select a server data file to connect to.

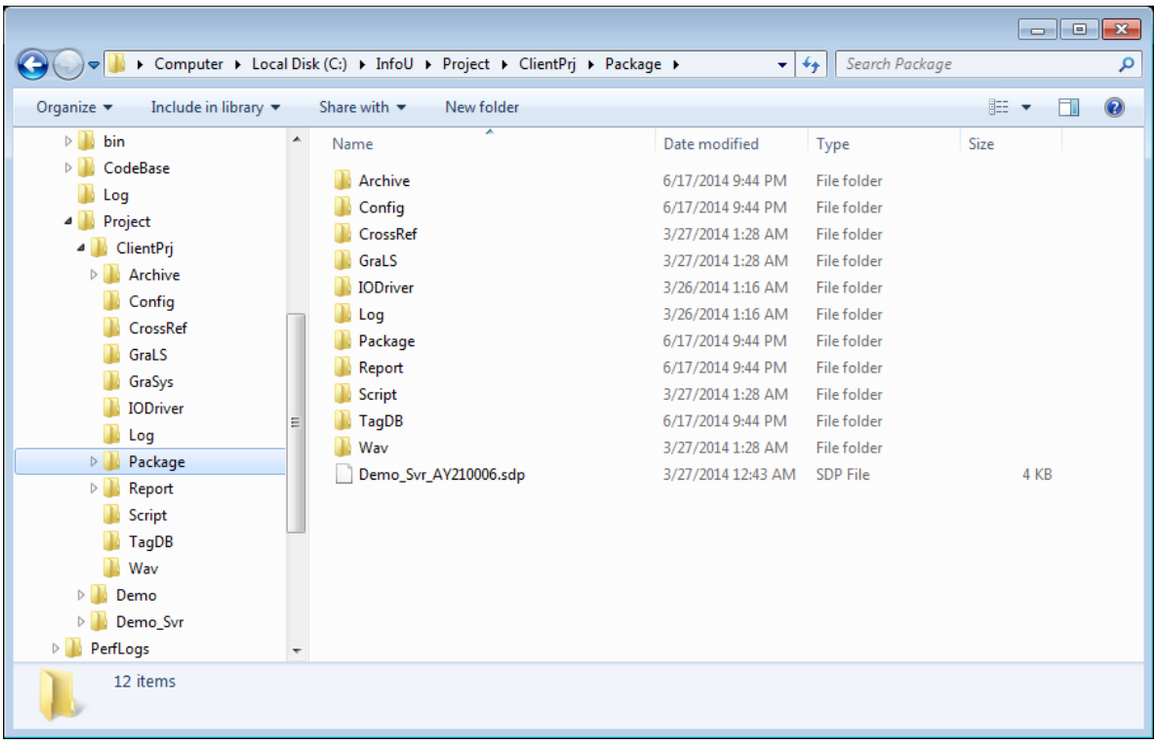


h. The information loaded from the selected server data file is displayed and 'Load' button become active.

i. Click on ‘Load’ to copy the needed server data file to the local computer. At this time, it may take a lot of time depending on the file capacity.



j. Once loading and duplicating the server data file are completed, information is displayed on the list and 'Delete' button becomes active.



k. The file is created in “Package” among the project folders.

l. If ‘Delete’ button is clicked, the server data file is deleted.

Notice

☞ At this time, the copied filed in the “Package” folder are not deleted.

Chapter 25 Mobile

The InfoU mobile is based on the Client/Server structure. The mobile designer and mobile server are installed at the server side; mobile Client (app) is installed in mobile devices (smart phone, tablet, etc.) at the client side.
The plant screen drawn by the mobile designer is serviced through the mobile server. The mobile client can remotely monitor and control the field conditions by accessing to the mobile server.

25.1 Mobile Designer

The InfoU Mobile Graphic Editor (hereinafter referred to as mobile designer) is the graphic editor that provides various functions to make monitoring screens.

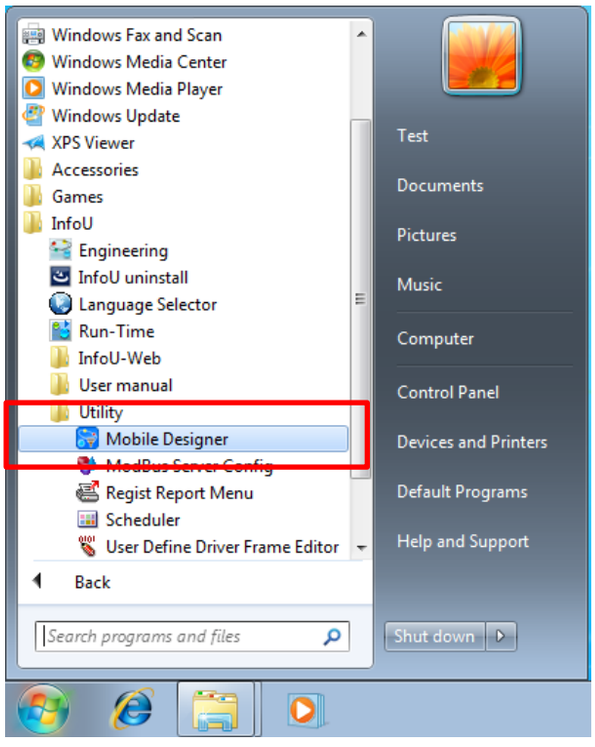
25.1.1 Startup

(1) Functions

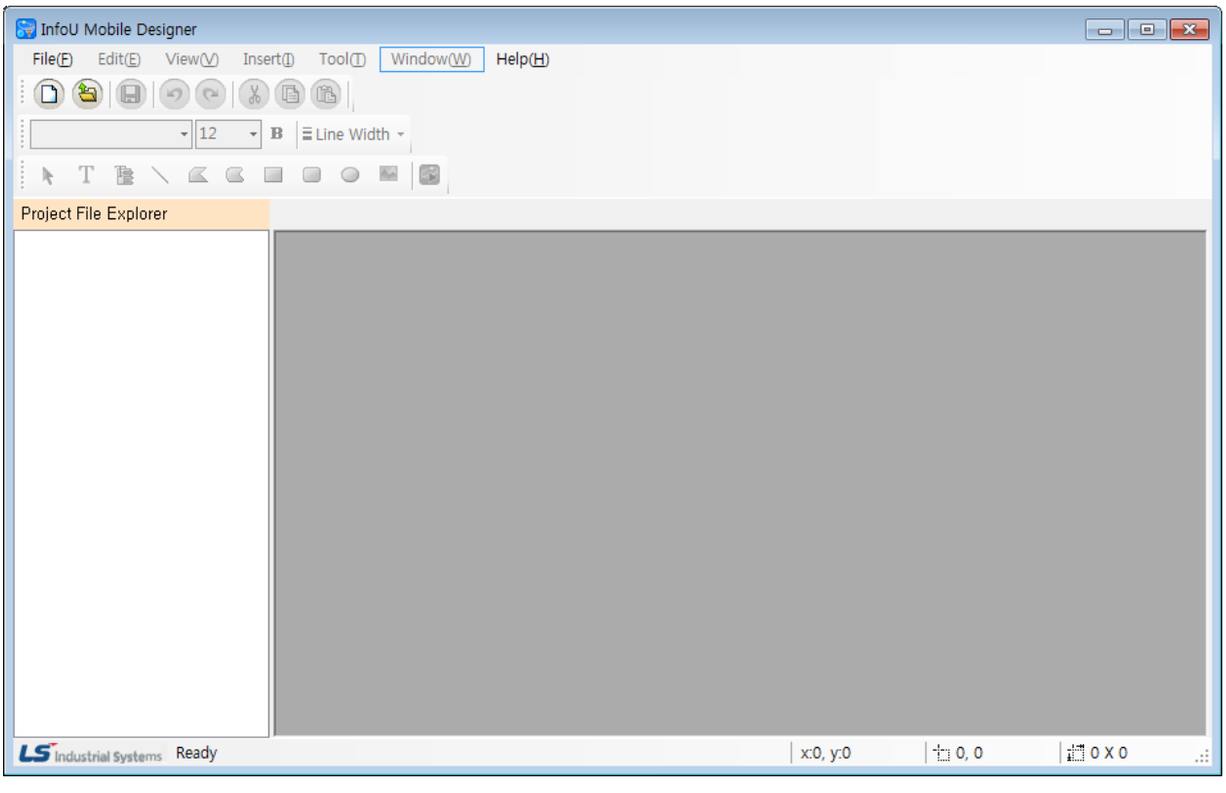
1) Various graphic object tools are provided. (line, polyline, polycurve, rectangle, Rounded Rectangle, ellipse, image, text, tag value)
2) Diverse animations and user action functions are provided by setting dynamic properties. (horizontal/vertical size, horizontal/vertical movement, color, blink, visible, rotation, writing tag values, etc.)
3) Various resolution can be applied depending on the mobile device's screen and it also supports the multi-resolution.
4) It provides the Control Unable/Enable function by user authority.
5) You can register the tags to be used in the mobile client by setting service tags. (tag group, alarm, trend group)

(2) Startup

You can select [**InfoU**] → **[Utility]** → [**Mobile Designer**] in the program list to start up the mobile designer.

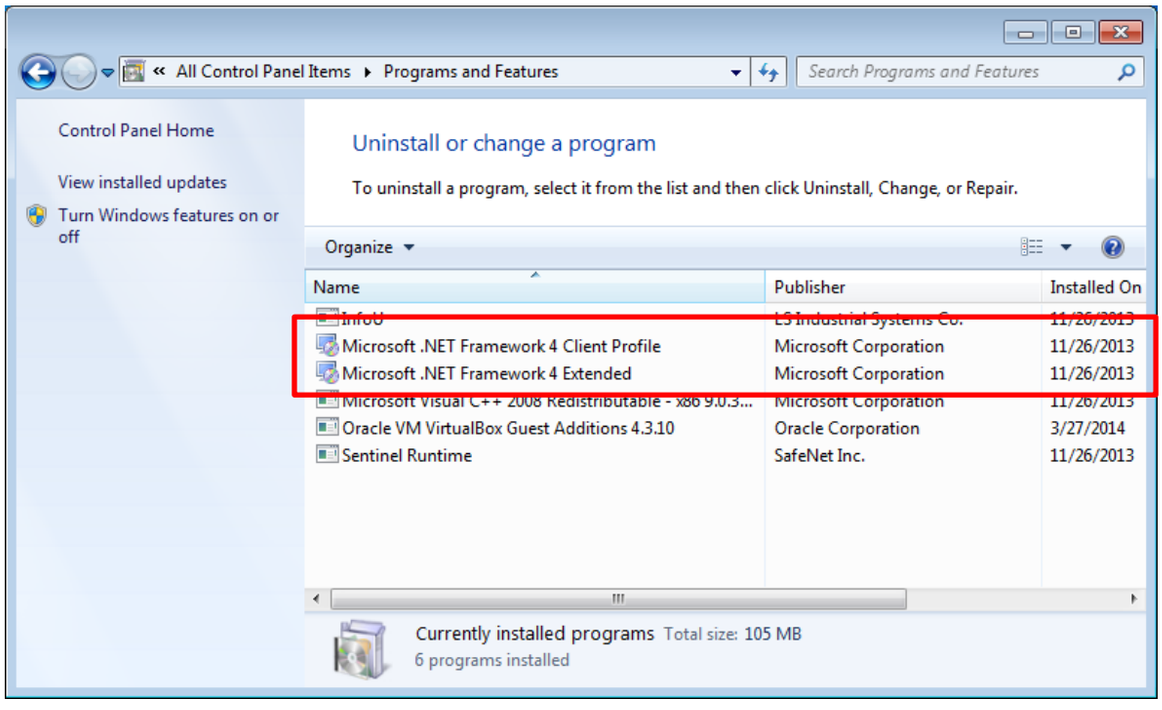


Then, the mobile designer will run as below.



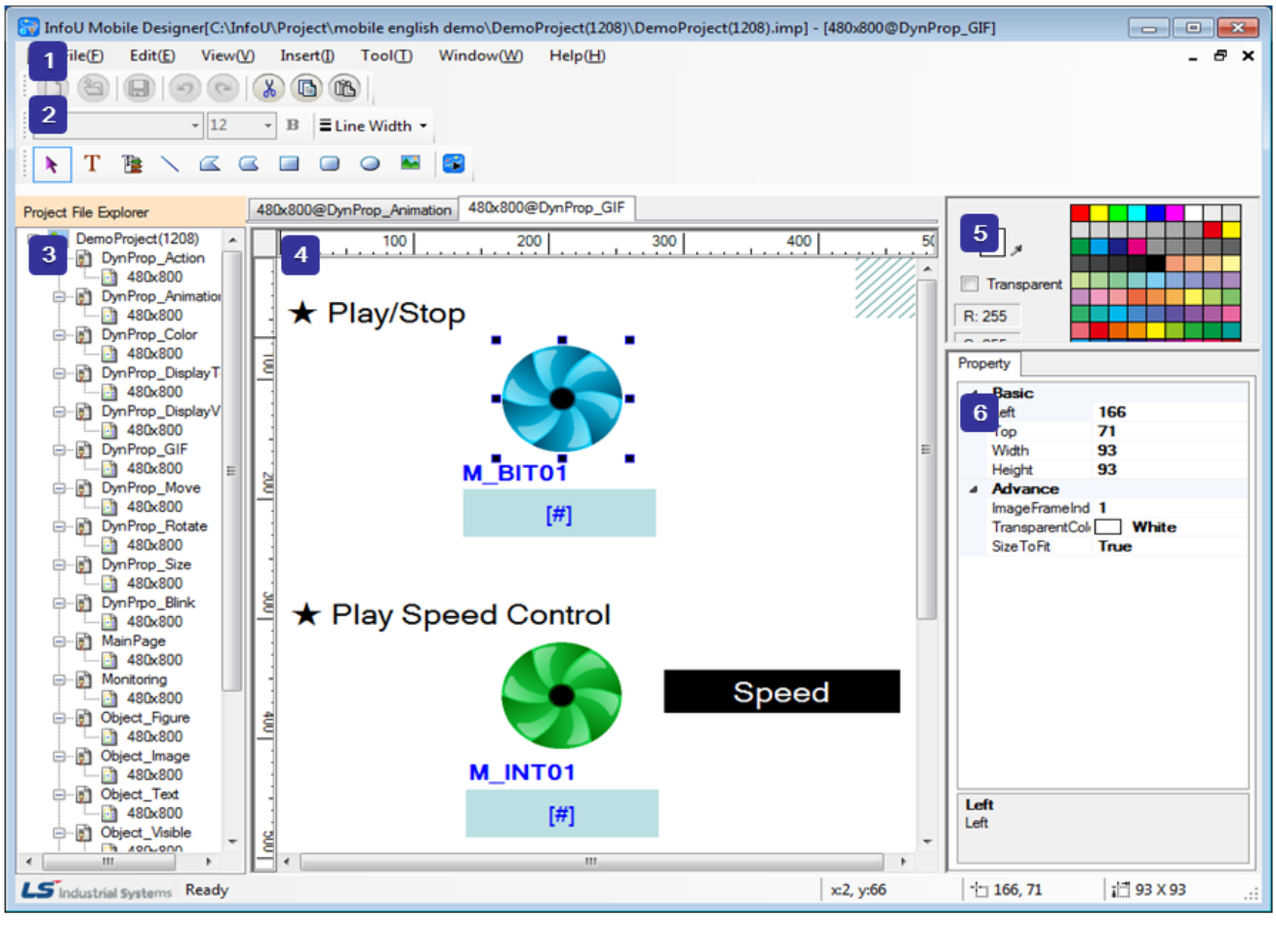
Notice

To run the mobile designer, .NET 4.0 full version should be installed.



(3) Configuration

The mobile designer is the system development tool, which defines and sets the system components required by the InfoU mobile Client such as screen composition, definition of dynamic properties, setup of project services, etc.



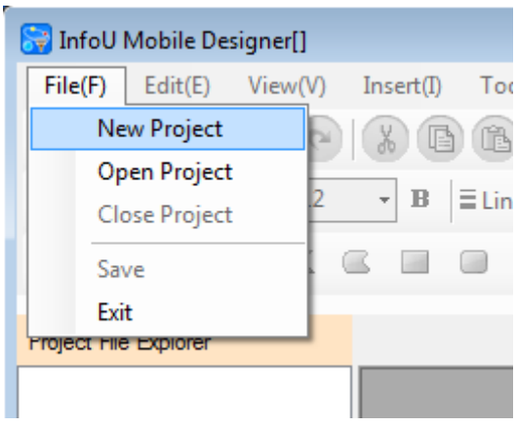
The descriptions on the screen components of the mobile designer are as follows.

1	**Main menu**: You can use the functions provided by the InfoU mobile graphic editor.
2	**Standard tool and object, form tools**: They are the icons of standard tools frequently used in the graphic editor and object tools required to make a page, tools required to define the properties of texts and lines.
3	**Project File Explorer**: It expresses the currently working project as a tree, which is composed of several resolutions for each page.
4	**Page editing screen**: It is the screen to edit the page. You can draw images using diverse objects and define user actions such as definition of dynamic properties, etc.
5	**Color tool**: You can conveniently specify the color of the page and object through this tool.
6	**Object property window**: You can change the properties of each object.

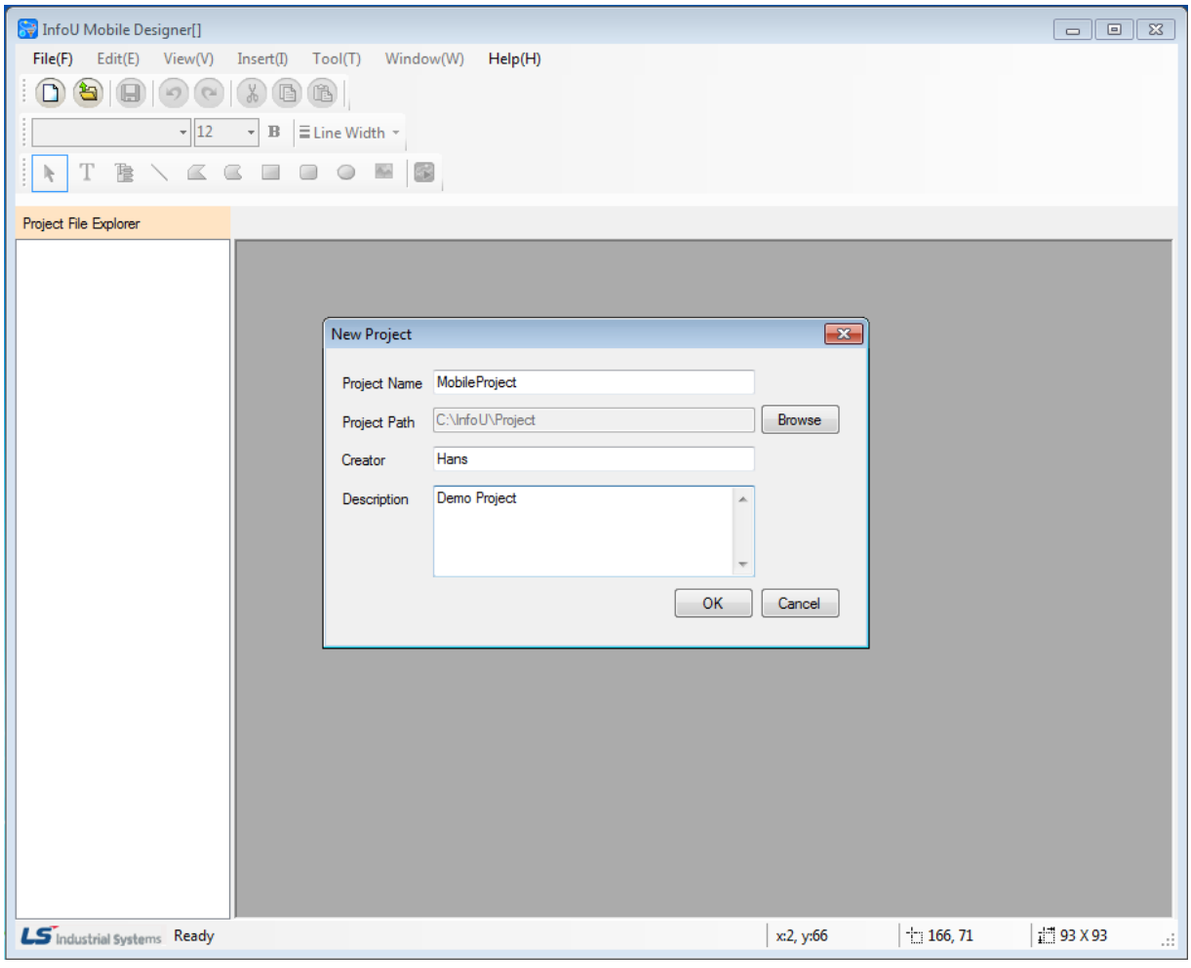
25.1.2 Mobile Server Project

1) Creation of the project

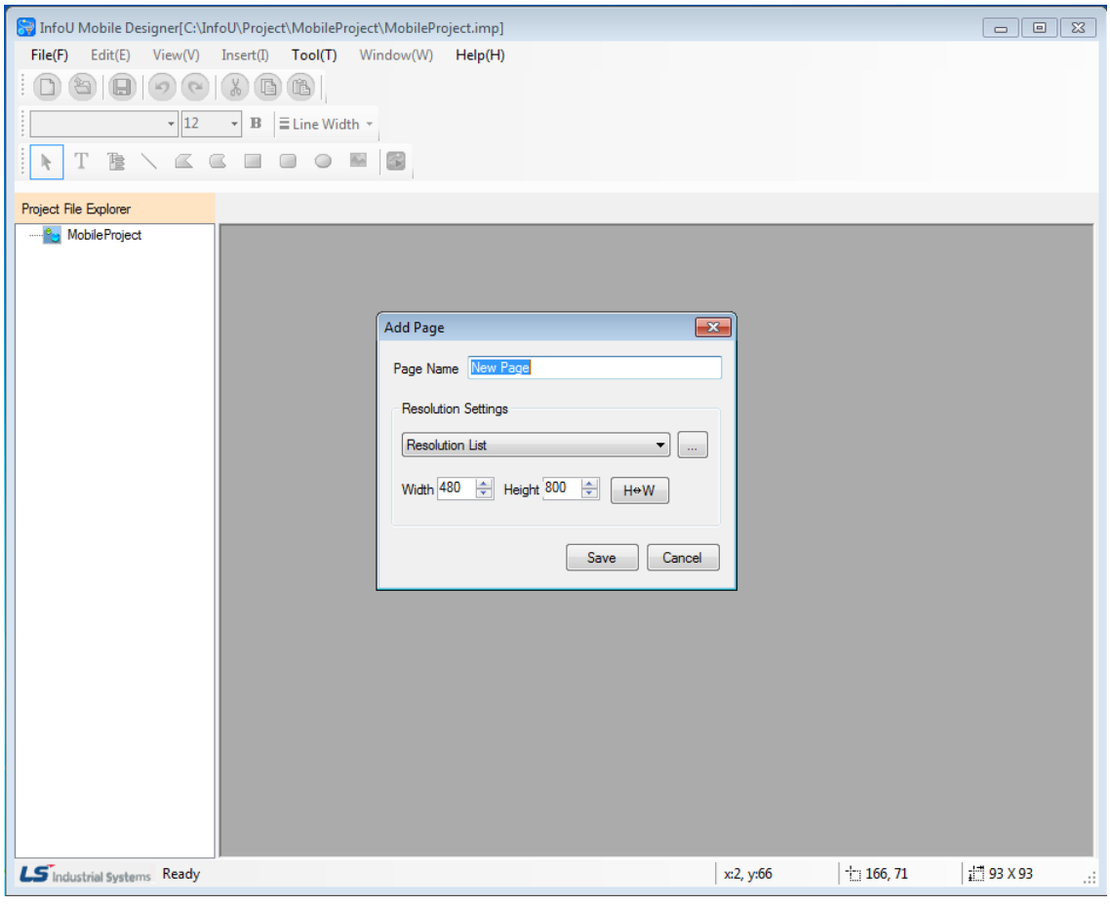
Create the mobile server project.



Run **[File]** → **[New Project]** in the main menu.

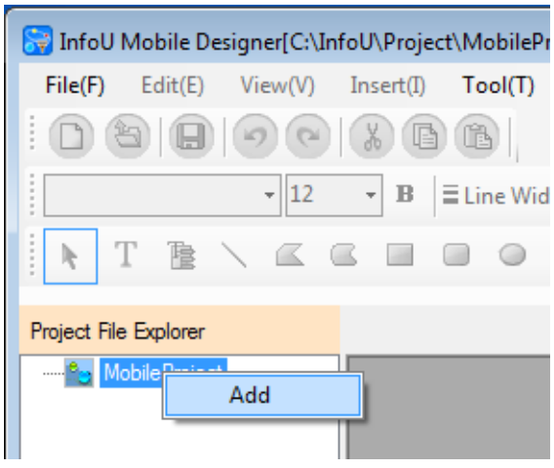


Input "**Project name**", "**Project path**", "**Creator**", "**Project description**" and click the "**Create**" button to make a new project. In this case, "Creator" and "Project description" can be omitted.



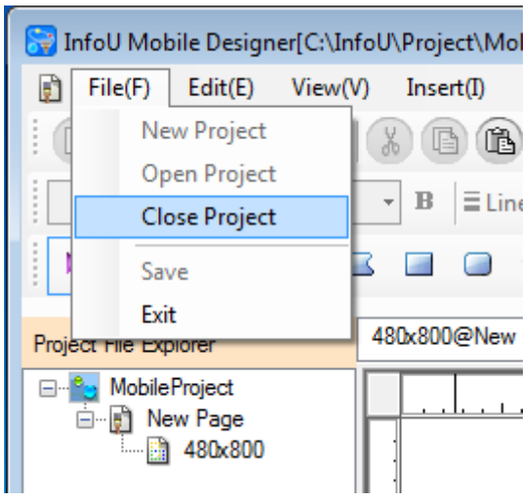
After the project is created, the window to add the page and resolution will be displayed as above. Input the "**Page Name**" and set the **width** and **height** of the page. Then, when you click the "**Save**" button, the page will be created.

You can also create the page and resolution by running the **"Add"** menu after executing the popup menu in the "**Project File Explorer**" window.



2) Close Project

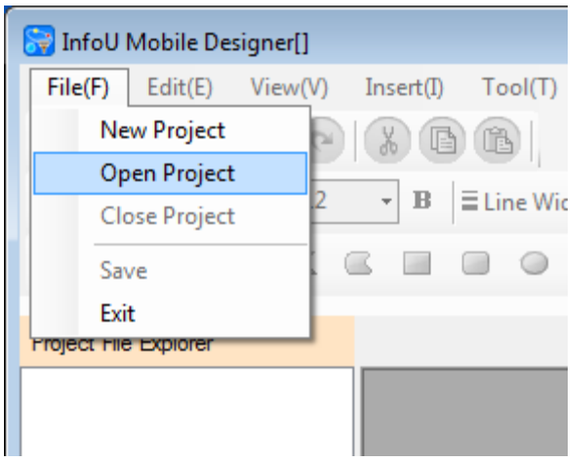
It closes the mobile server project.



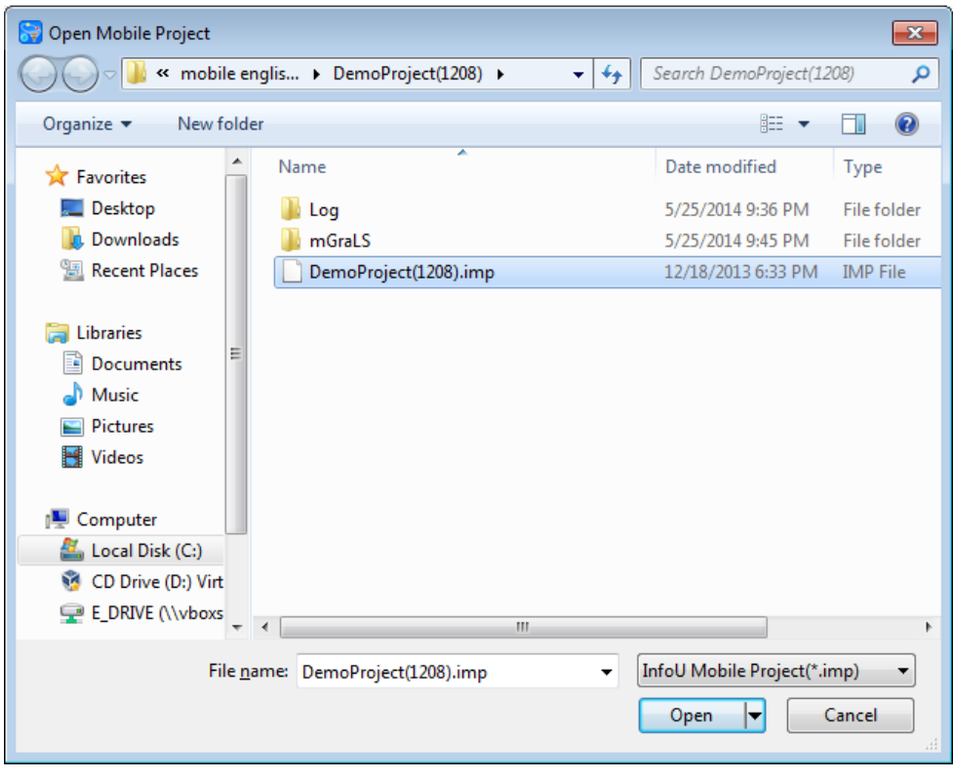
If you run **[File]** → [**Close Project**] in the main menu, the project will be closed and it will be switched into the initial screen.

3) Open Project

It can open the mobile server project to perform tasks.



If you run **[File]** → [**Open Project**] in the main menu, the dialog box to select the file will show up.



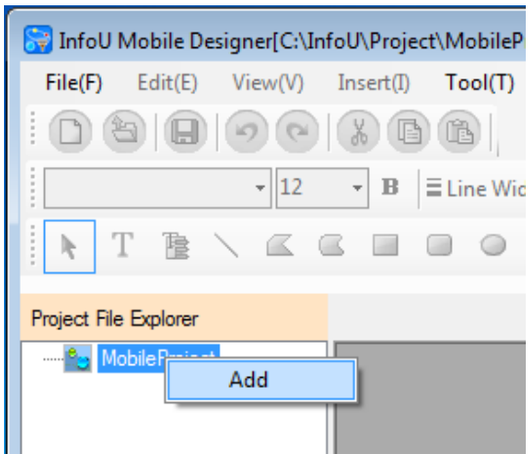
After selecting the mobile project file, if you click the **"Open"** button, the project will be open. The extension of the InfoU mobile project is "***.imp**".

25.1.3 Mobile APP Page

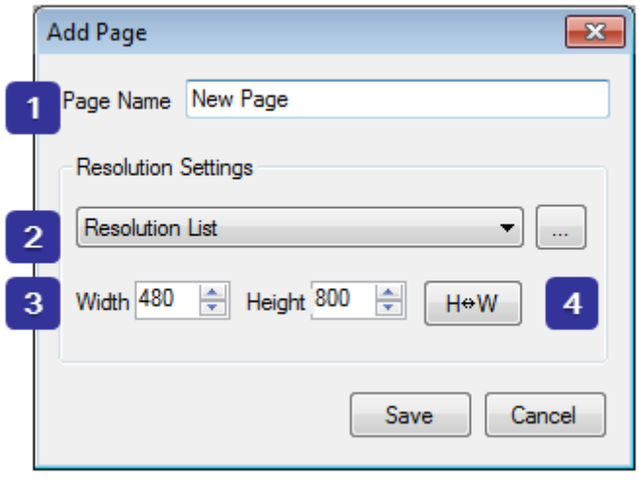
The mobile app page is the monitoring screen used for the mobile client. There are so various types of mobile clients with their own diverse resolutions. Even in the same mobile client, different resolution documents are required to provide the best monitoring screen depending on each viewpoint. Accordingly, the mobile app page should be expressed as several resolution documents even in the same monitoring screen. The InfoU mobile app page is composed of various resolution documents.

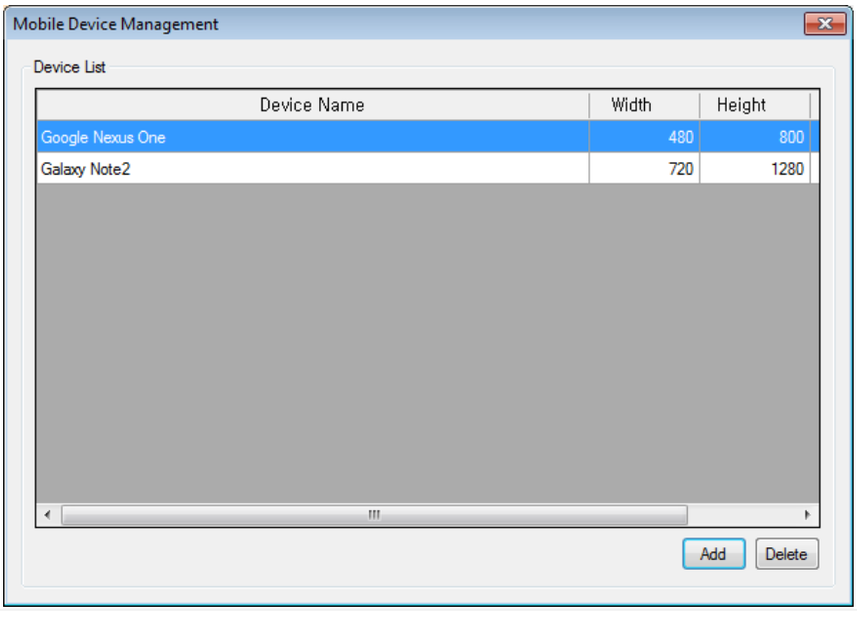
1) Add Page

It creates the mobile app page.



Run the **"Add"** menu by executing the popup menu in the "Project File Explorer" window.

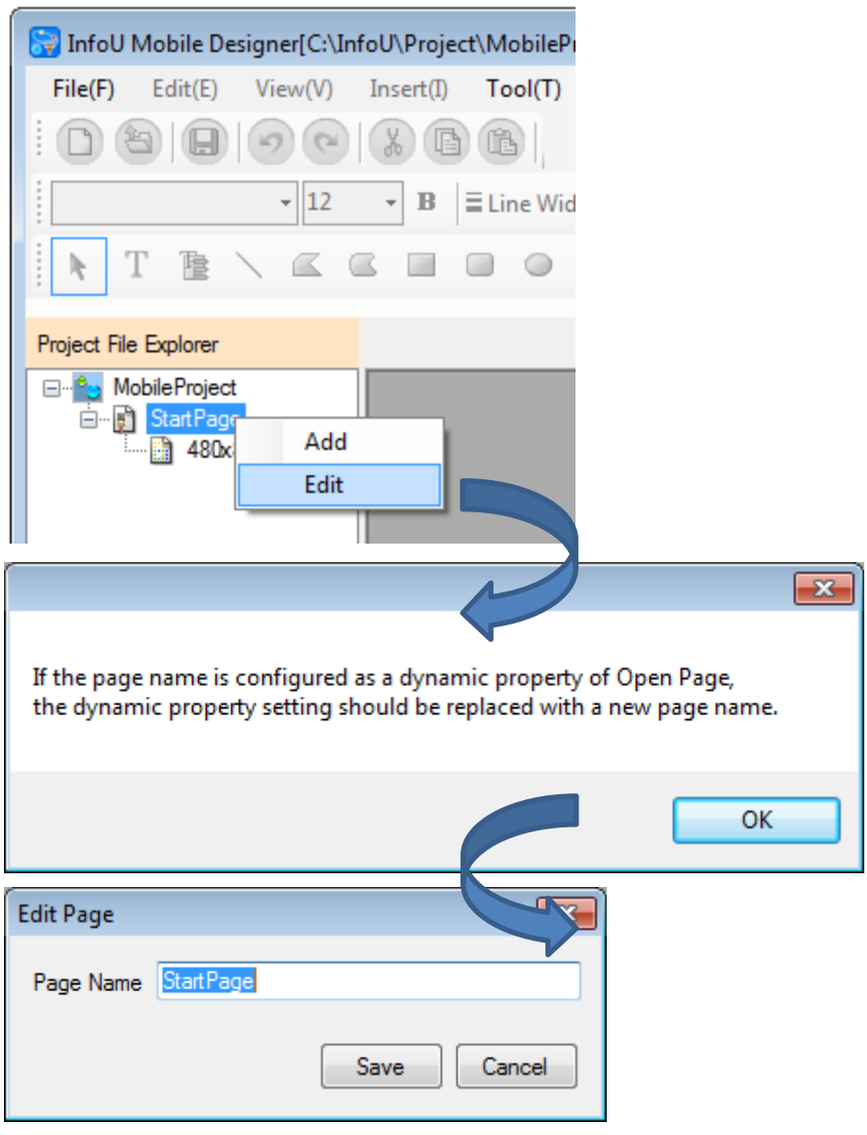


1	Enter the page name.
2	You can select the pre-programmed resolution. If you click the "…" button, the window for resolution management will be displayed. Then, you can add/delete the desired list. 
3	You can set up any resolution discretionally.
4	You can change the width and height of the resolution.

If you press the **"Save"** button, 1 mobile app page and 1 resolution document will be created.

2) Edit Page

You can edit the name of the mobile app page.



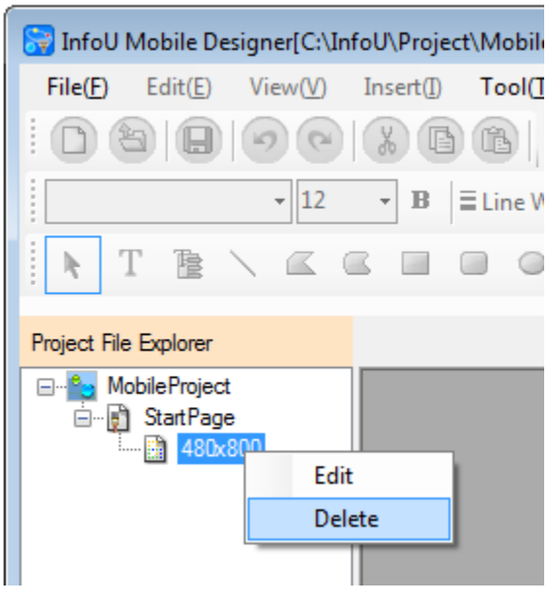
If you select "**Edit**" in the popup menu, the information window will be displayed. When you press the "**OK**" button in the information window, the 'Edit Page' window will be displayed. If you enter a new page name in the edition window and click the **"Save"** button, the page name will be modified.

Notice

In case the previous page name was applied to the dynamic property of transition, if you edit the page name, the relevant dynamic property will not run. Namely, if you edit the page name, you need to change the settings of the dynamic property of the page transition.

3) Delete Page

It deletes the mobile app page. The mobile app page is composed of resolution documents. If the resolution documents are all deleted, the relevant mobile app page will also be deleted.



You can delete the resolution document by running **"Delete"** in the popup menu.

Notice

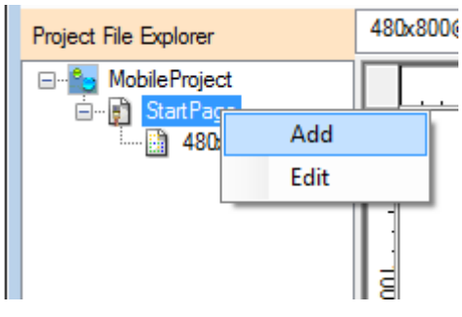
If there are several resolution documents, when you run "**Delete**", the selected resolution document only will be deleted.

25.1.4 Resolution Document

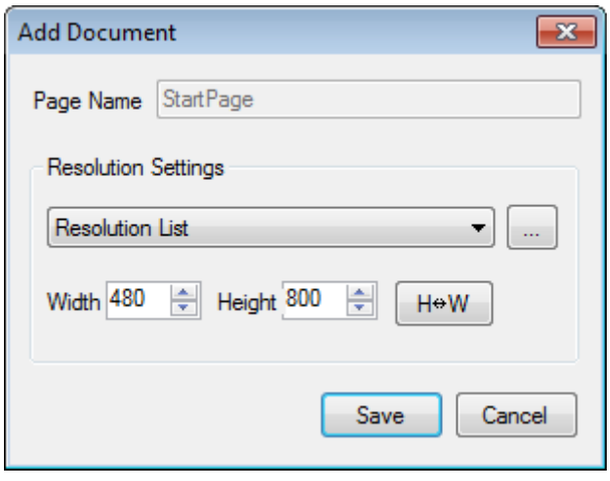
It is the document that expresses the mobile app page with a specific resolution. The mobile app page can have multiple resolution documents to realize the screen optimized for the devices with various resolutions.

1) Add Resolution Document

It adds the resolution document to the mobile app page.



Run **"Add"** in the popup menu of the mobile app page.



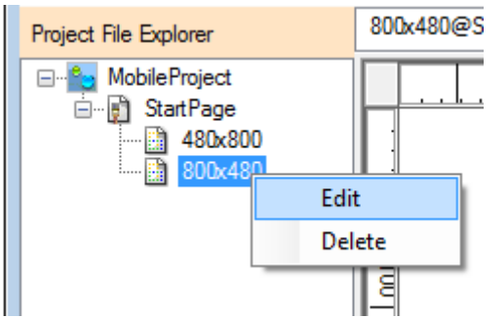
After setting the resolution of the resolution document, if you click the **"Save"** button, the document will be added.

Notice

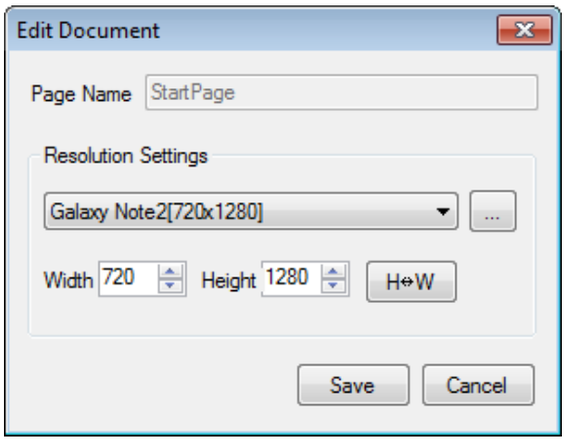
You cannot repetitively add the resolution document with the same resolution.

2) Edit resolution document

It edits the resolution of the resolution document.



You can run "**Edit**" in the popup menu of the resolution document.



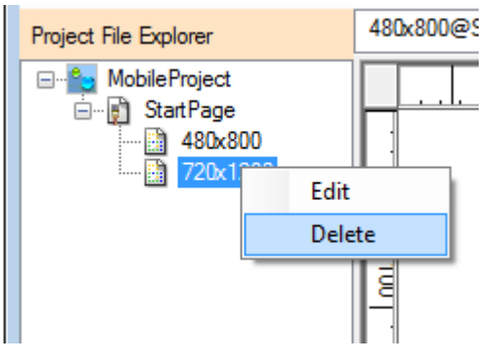
If you edit the resolution (width, height) and click the "**Save**" button", the modified resolution will be saved.

Notice

If the resolution document to edit is open, edition works can be performed only after closing the relevant document.

3) Delete Resolution Document

It deletes the resolution document from the mobile app page.



You can urn "**Delete"** in the popup menu of the resolution document.

Notice

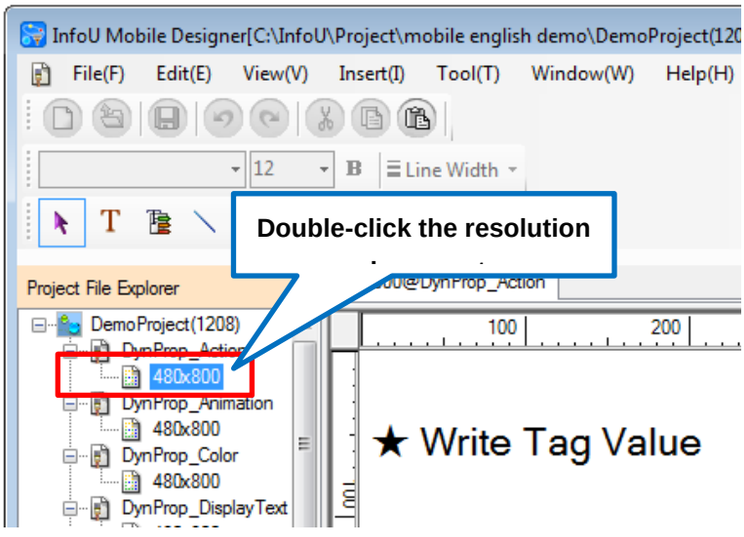
Refer to the following instructions to delete the resolution document.

☞ In case there is one resolution document in the mobile app page, when you delete the resolution document, the mobile app page will also be deleted.

☞ In case the resolution document to be deleted is open, deletion works can be performed only after closing the document.

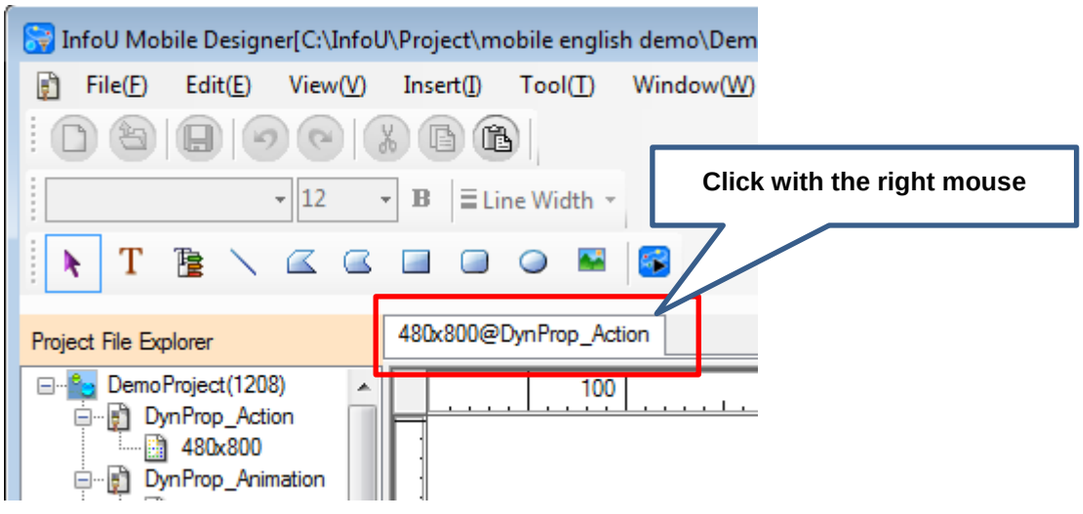
4) Open Resolution Document

You can open the resolution document and perform drawing works by double-clicking the resolution document in the project file explorer.



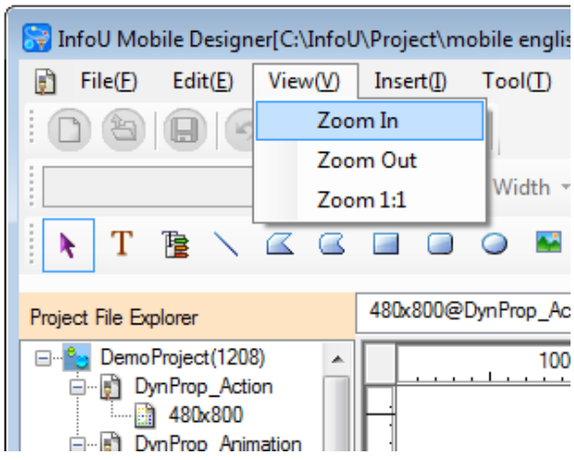
5) Close Resolution Document

To close the resolution document, click with the right mouse button on the tab of the open resolution document.



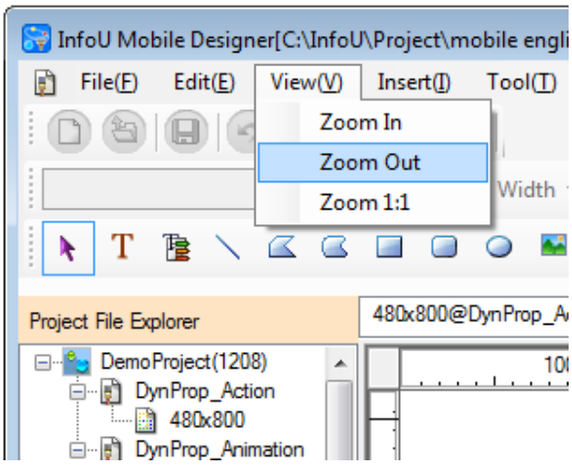
6) Zoom In Resolution Document

To zoom in the resolution document, select [**View**] → [**Zoom In**] in the main menu.



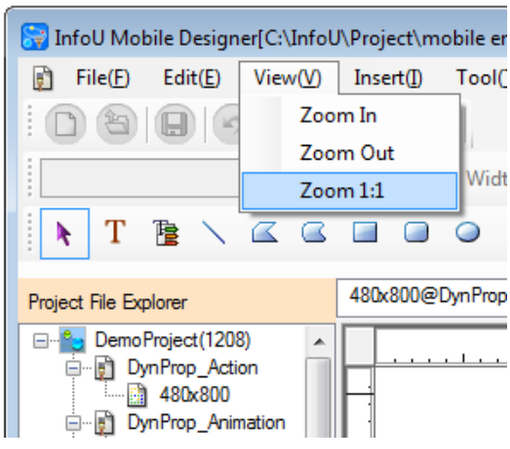
7) Zoom Out Resolution Document

To zoom out the resolution document, select [**View**] → [**Zoom Out**] in the main menu.



8) View the original of the resolution document

To view the original size of the resolution document select [**View**] → [**1:1 ratio**] in the main menu.



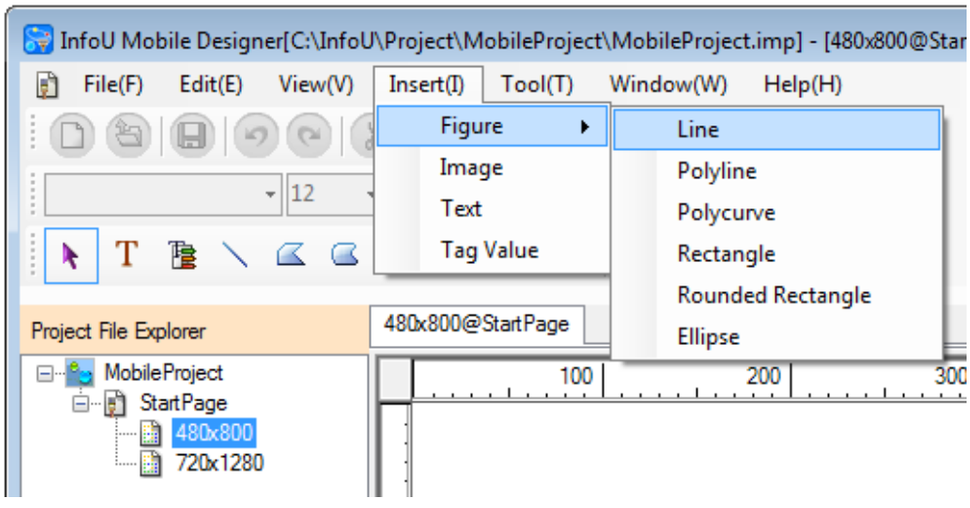
9) Scrolling of resolution document

Under the status of Zoom In, you can move the screen while you click with the right mouse button on the resolution document. When scrolling starts, the mouse cursor will be changed into the shape of () standing for movement.

25.1.5 Object

1) Line

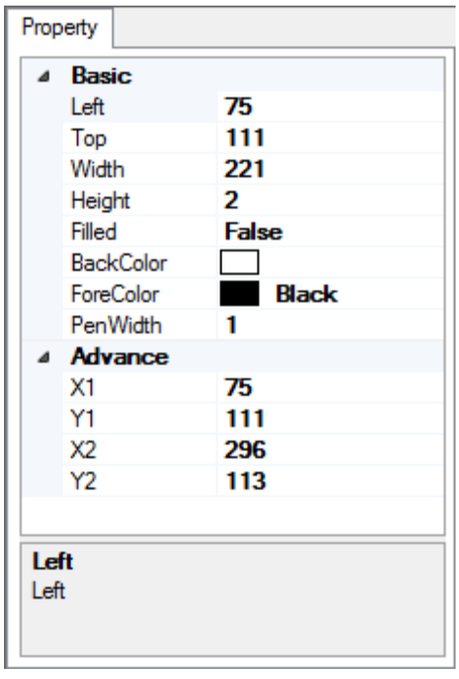
It displays the linear object.





For drawing a line, you can select **[Insert]** → **[Figure]** → [**Line**] in the main menu or select the icon in the shape of “Line” from the toolbar.

The properties of the linear object are as follows.



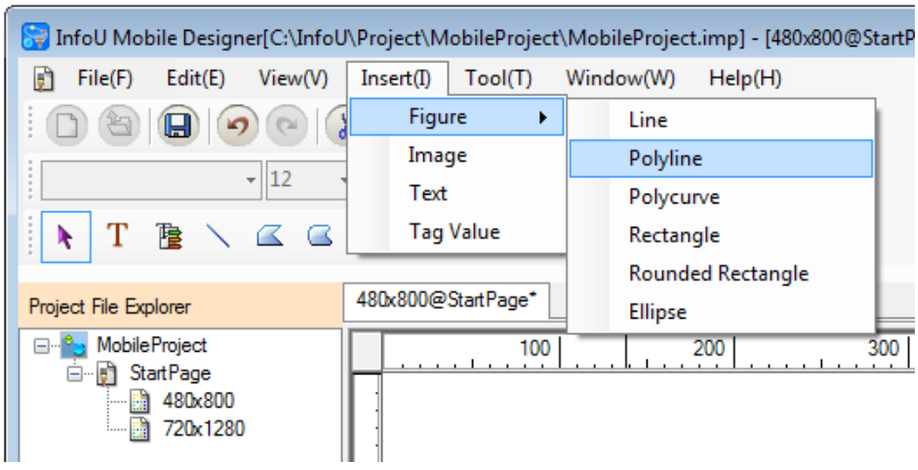
- **LEFT coordinate, TOP coordinate**: Coordinate values of the upper left corner of the object
- **Width, Height**: Width and height of the object
- **Filled**: You can determine whether filling the inside of the object with the background color. It is not available in the line object.
- **BackColor**: You can select the color to fill the inside of the object. It is not available in the line object.
- **ForeColor**: You can select the color of the object.
- **Line thickness**: You can select the thickness of the line forming the object.
- **Start point X coordinate, start point Y coordinate**: Coordinate value of the upper left corner of the object
- **End point X coordinate, end point Y coordinate**: Coordinate value of the bottom right corner of the object

Notice

There is no limit on the range of value input, however, please note that you need to consider the screen size. If you input too large value, it will be displayed outside of the screen so the object may not be displayed on the screen or cut off.

2) Polyline

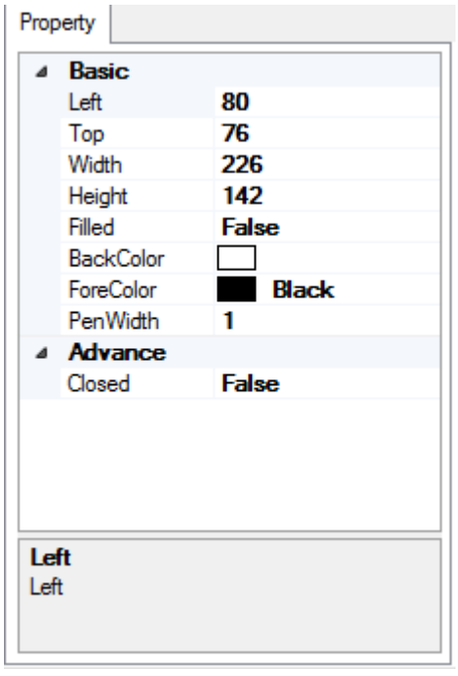
It displays the polyline object.





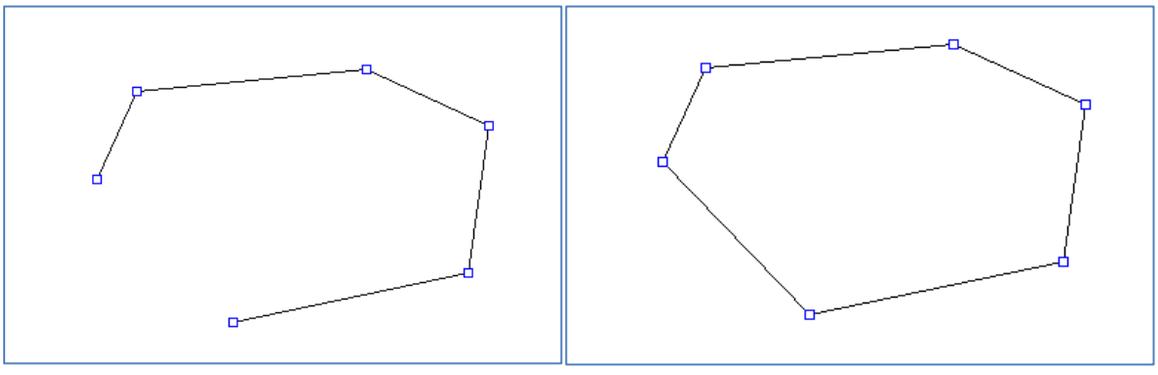
For drawing a polyline, you can select **[Insert]** → **[Figure]** → [**Polyline**] in the main menu or select the icon in the shape of “Polyline” from the toolbar. You can draw a figure in the manner of clicking the mouse and adding a dot. To exit Drawing Polyline, click with the right mouse button.

The properties of the polyline object are as follows.



- **LEFT coordinate, TOP coordinate**: Coordinate values of the upper left corner of the object
- **Width, Height**: Width and height of the object
- **Filled**: You can determine whether filling the inside of the object with the background color.

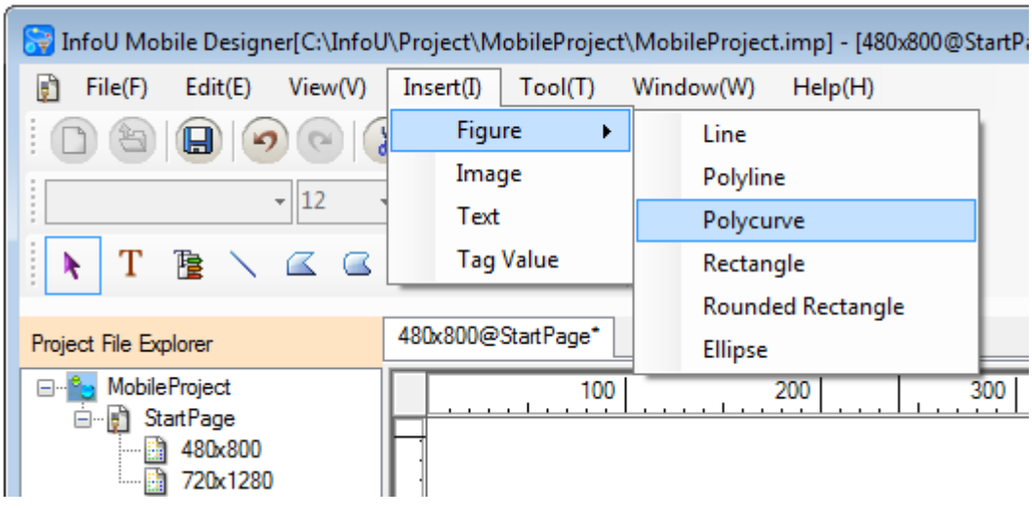
- **BackColor**: You can select the color to fill the inside of the object.
- **ForeColor**: You can select the color of the object.
- **Line thickness**: You can select the thickness of the line forming the object.
  - **Path Close**: If it is TRUE, the open object will become the closed one; If it is FALSE, the closed object become the open one.



< Path Close: False> < Path Close: True>

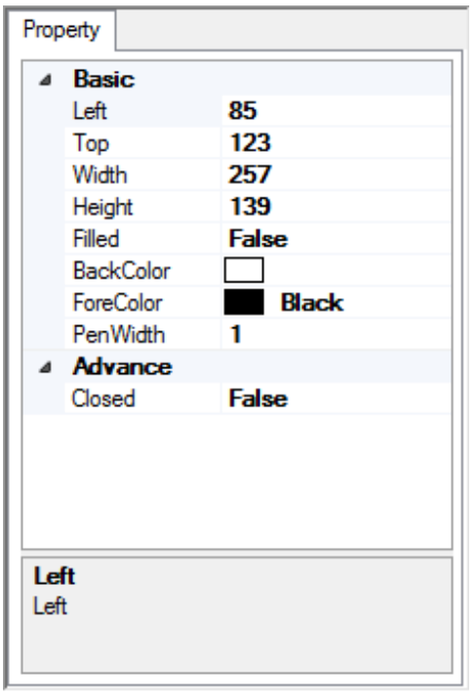
3) Polycurve

It displays the polycurve object.





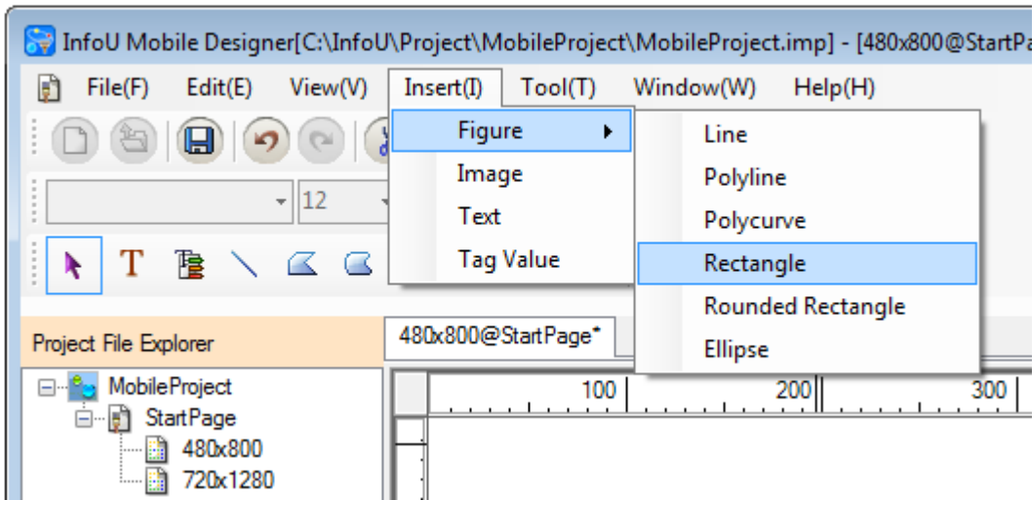
For drawing a polycurve, you can select **[Insert]** → **[Figure]** → [**Polycurve**] in the main menu or select the icon in the shape of “Polycurve” from the toolbar. The drawing method is the same as the polyline. The properties of the polycurve object are as follows.



- **LEFT coordinate, TOP coordinate**: Coordinate values of the upper left corner of the object
- **Width, Height**: Width and height of the object
- **Filled**: You can determine whether filling the inside of the object with the background color.
- **BackColor**: You can select the color to fill the inside of the object.
- **ForeColor**: You can select the color of the object.
- **Line thickness**: You can select the thickness of the line forming the object.
- **Path Close**: If it is TRUE, the open object will become the closed one. If it is FALSE, the closed object become the open one. The way of Path Close is the same as the polyline's.

4) Rectangle

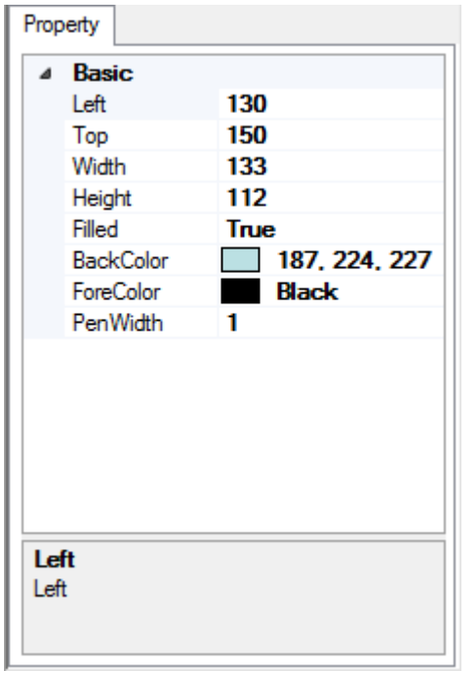
It displays the rectangle object.





For drawing a rectangle, you can select **[Insert]** → **[Figure]** → [**Rectangle**] in the main menu or select the icon in the shape of “Rectangle” from the toolbar.

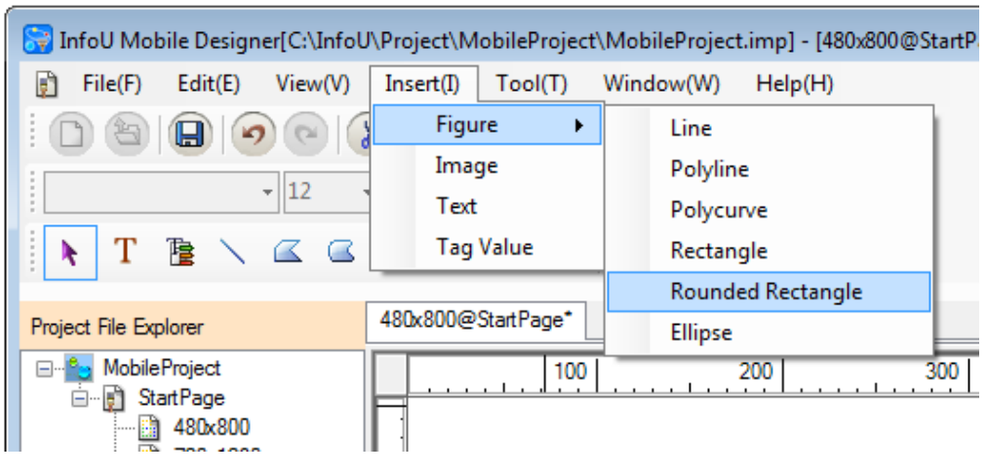
The properties of the rectangle object are as follows.



- **LEFT coordinate, TOP coordinate**: Coordinate values of the upper left corner of the object
- **Width, Height**: Width and height of the object
- **Filled**: You can determine whether filling the inside of the object with the background color.
- **BackColor**: You can select the color to fill the inside of the object.
- **ForeColor**: You can select the color of the object.
- **Line thickness**: You can select the thickness of the line forming the object.

5) Rounded Rectangle

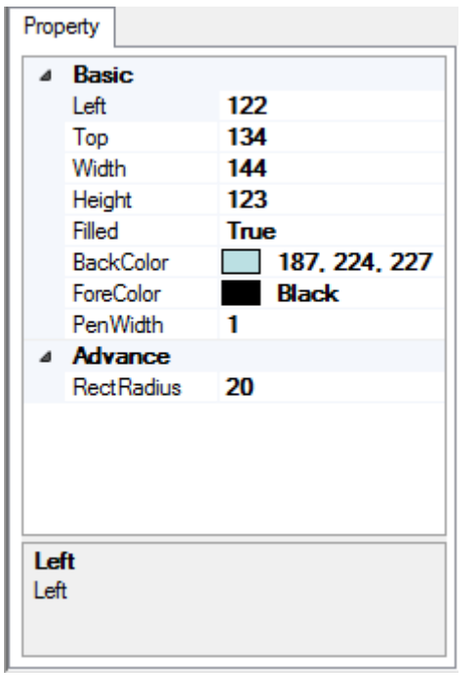
It displays the Rounded Rectangle object.



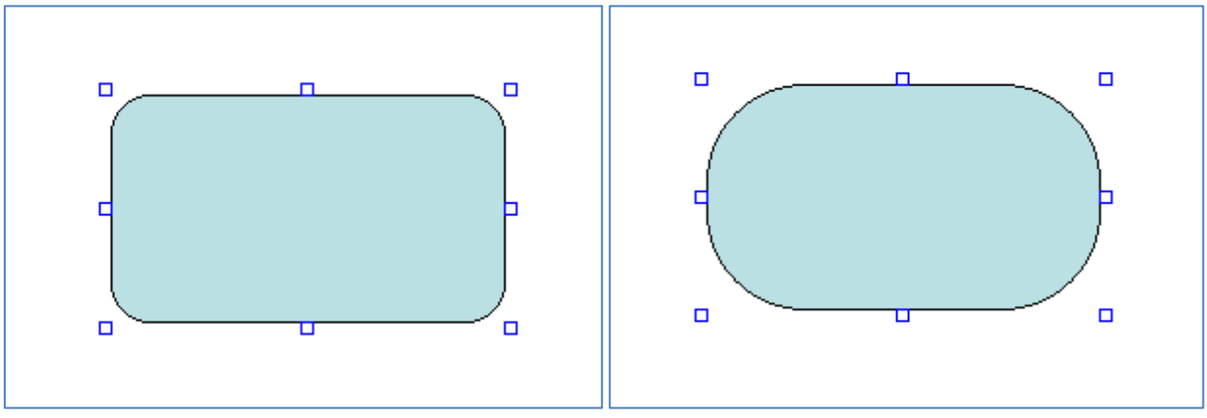


For drawing a Rounded Rectangle, you can select **[Insert]** → **[Figure]** → [**Round rectangle**] in the main menu or select the icon in the shape of “Rounded Rectangle” from the toolbar.

The properties of the Rounded Rectangle object are as follows.



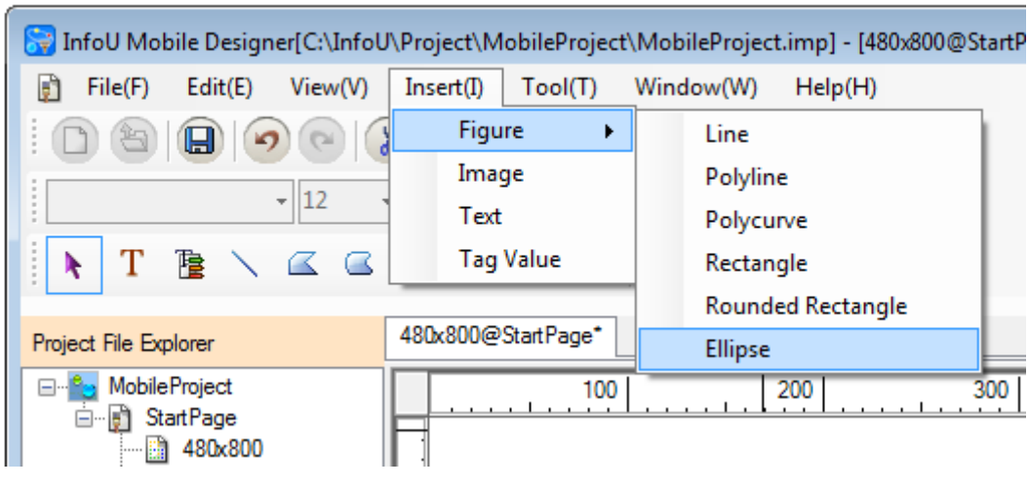
- **LEFT coordinate, TOP coordinate**: Coordinate values of the upper left corner of the object
- **Width, Height**: Width and height of the object
- **Filled**: You can determine whether filling the inside of the object with the background color.
- **BackColor**: You can select the color to fill the inside of the object.
- **ForeColor**: You can select the color of the object.
- **Line thickness**: You can select the thickness of the line forming the object.
- **Rounding**: You can set the round value of a vertex. The greater the value is the bigger the rounding area is.



<Rounding: 20> <Rounding: 50>

6) Ellipse

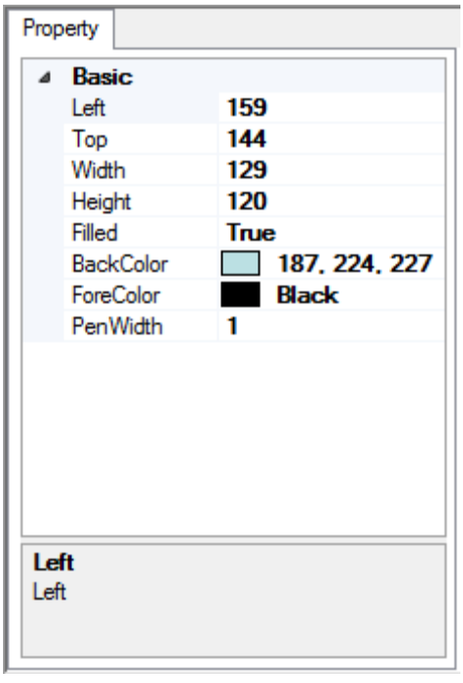
It displays the ellipse object.





For drawing an ellipse, you can select **[Insert]** → **[Figure]** → [**Ellipse**] in the main menu or select the icon in the shape of “Ellipse” from the toolbar.

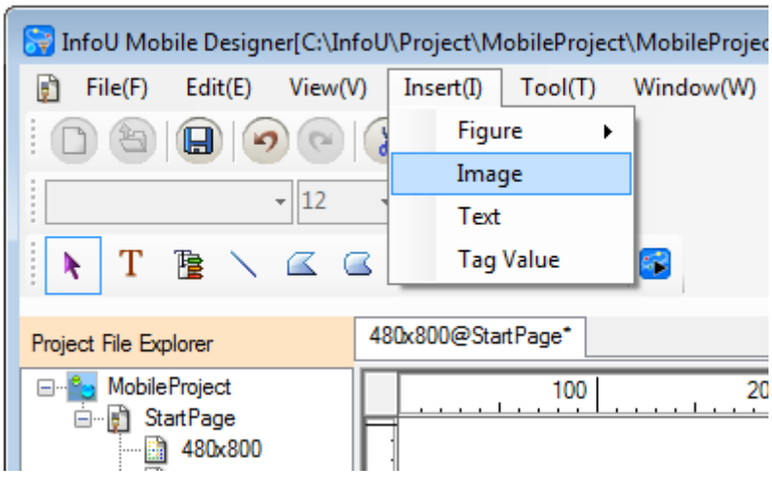
The properties of the ellipse object are as follows.



- **LEFT coordinate, TOP coordinate**: Coordinate values of the upper left corner of the object.
- **Width, Height**: Width and height of the object
- **Filled**: You can determine whether filling the inside of the object with the background color.
- **BackColor**: You can select the color to fill the inside of the object.
- **ForeColor**: You can select the color of the object.
- **Line thickness**: You can select the thickness of the line forming the object.

7) Image

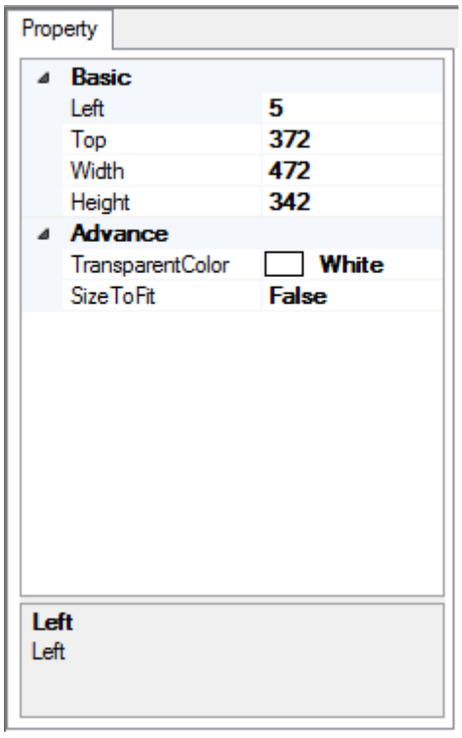
It displays the image object.





To insert an image into the screen, you can select **[Insert]** → **[Image]** in the main menu or select the icon in the shape of “Image” from the toolbar. After clicking with the mouse on the position to insert the image, if you select the targeted image file in the dialog box for file selection, the image will be inserted into the page. The available image formats are “bmp”, “gif”, “jpg”, “png”, tiff”, etc.

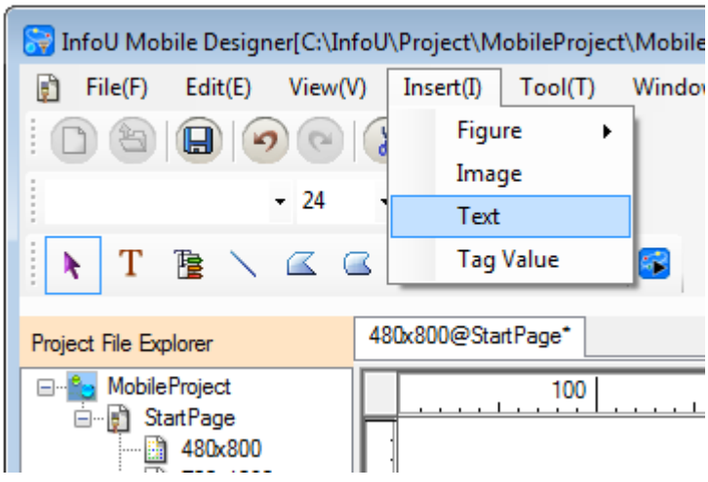
The properties of the image object are as follows.



- **LEFT coordinate, TOP coordinate**: Coordinate values of the upper left corner of the object
- **Width, Height**: Width and height of the object
- **Transparent color**: You can set the color to transparentize.
- **Size to Fit**: If it is TRUE, the image object size will be fit to the image size. In case users change the image object size randomly, it will be displayed as FALSE.

8) Text

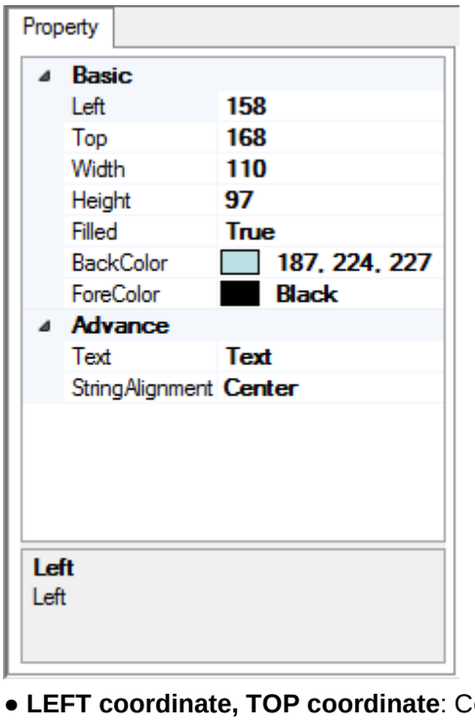
It displays the text object.





To insert a text, you can select **[Insert]** → **[Text]** in the main menu or select the icon in the shape of "Text" from the toolbar.

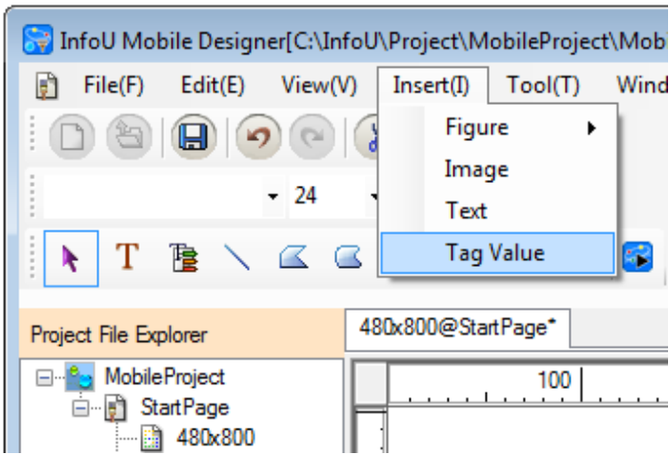
The properties of the text object are as follows.



- **LEFT coordinate, TOP coordinate**: Coordinate values of the upper left corner of the object
- **Width, Height**: Width and height of the object
- **Filled**: You can determine whether filling the inside of the object with the background color.
- **BackColor**: You can select the color to fill the inside of the object.
- **ForeColor**: You can select the color of the object.
- **Text**: You can specify the text of the object.
- **String Alignment**: You can select the text alignment of the object.

9) Tag value

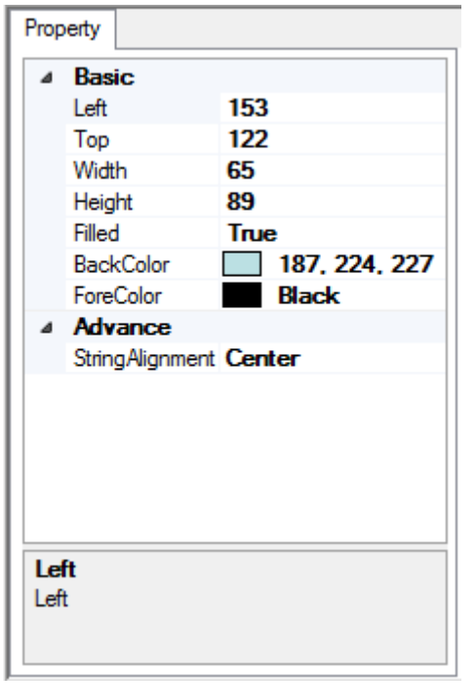
It displays the tag value object.





To insert the tag value object, you can select **[Insert]** → **[Tag Value]** in the main menu or select the icon in the shape of "Tag Value" from the toolbar. If you select the tag through the dynamic property setting (value display), the tag value will be displayed.

The properties of the tag value object are as follows.

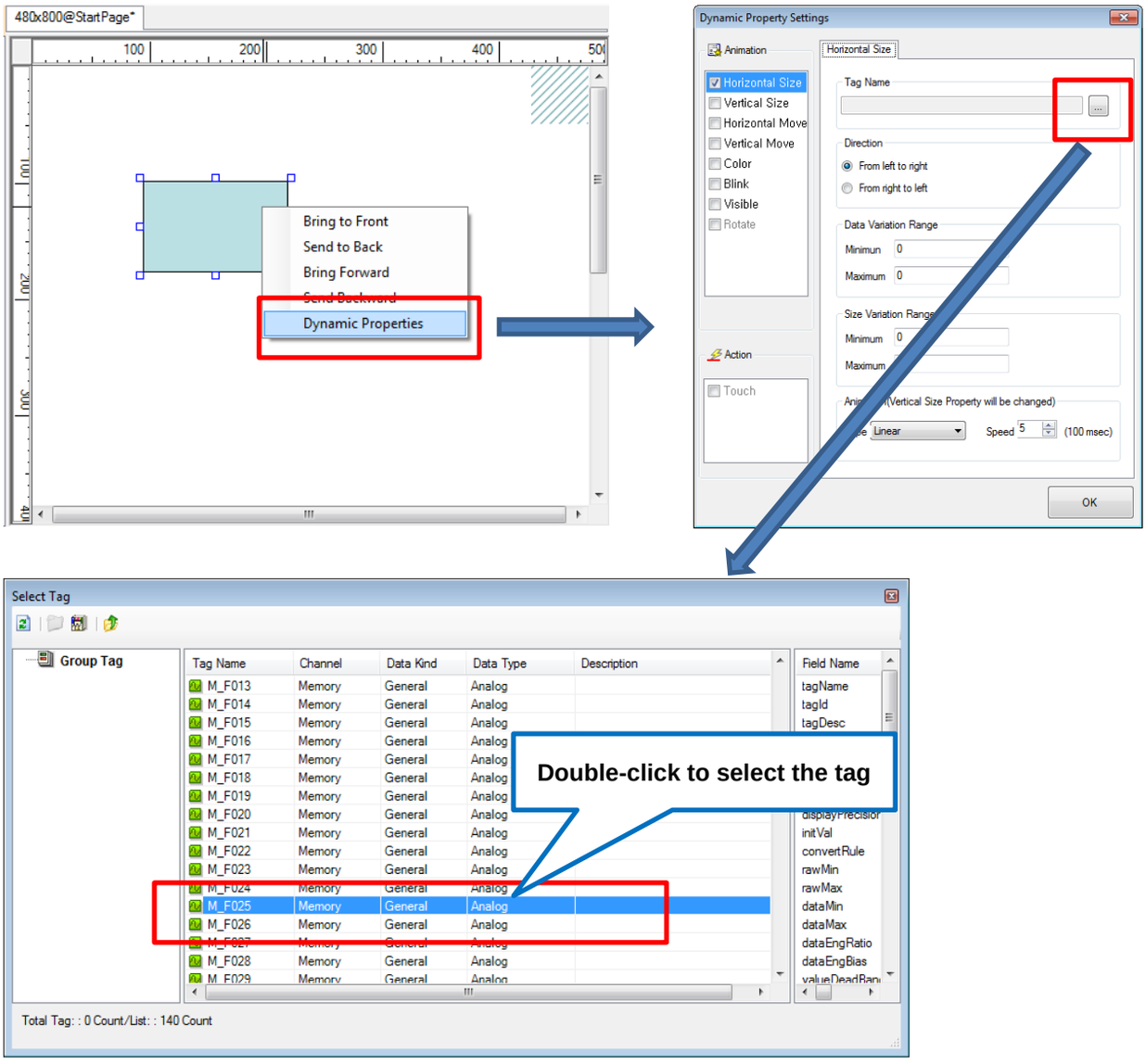


- **LEFT coordinate, TOP coordinate**: Coordinate values of the upper left corner of the object
- **Width, Height**: Width and height of the object
- **Filled**: You can determine whether filling the inside of the object with the background color.

- **BackColor**: You can select the color to fill the inside of the object.
- **ForeColor**: You can select the color of the object.
- **String Alignment**: You can select the text alignment of the object

25.1.6 Dynamic Properties

You can set the dynamic properties of the object.



Click with the right mouse button on the object to run the popup menu. If you run **"Define Dynamic Properties**" in the popup menu, the window for setting dynamic properties will be displayed. The dynamic properties accompanying the animation are basically linked with tags. If you select the "**…**" button next to the tag name, the tag selection window will show up and then, double-click the targeted tag.

Notice

The instructions related to dynamic properties are as follows.

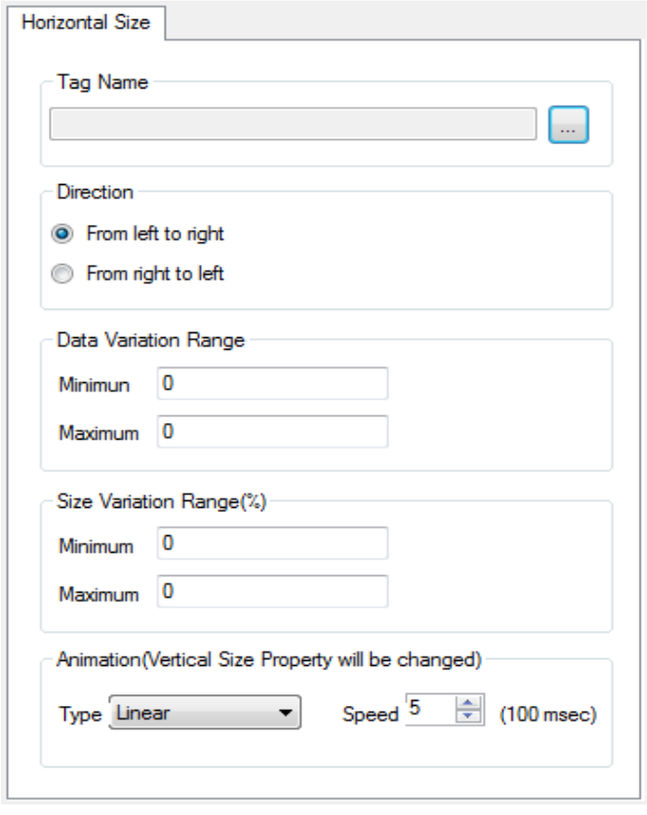
☞ The dynamic properties should be set for each object.

☞ The dynamic property can be visible only when the relevant property is checked.

☞ You can set up the multiple dynamic properties for one object.

1) Horizontal size

It sets the dynamic property whose size varies in the horizontal direction.



- **Tag Name**: You can select the tag that receives the input value for displaying the dynamic properties.
- **Direction**: You can select the direction to move.
- **Data Variation Range**: You can enter the range of tag input value used for dynamic properties. For example, if you set the minimum value (50), the maximum value (100), when the tag value is between 50 and 100, the dynamic properties will be displayed. Although there is no limit on the rage of input value, you need to consider the range of the input tag value.
- **Size Variation Range (%)**: You can enter the size variation range by percent (%). The input value should be between 0 and 100.
- **Animation (vertical size property will be changed)**: You can select animation effects among the below tree types.
  -Linear: It is displayed according to the value without any special animation effects.
  -Bounce: It shows the bounce effect at the last part.
  -Elastic: It shows the elastic effect on the move.

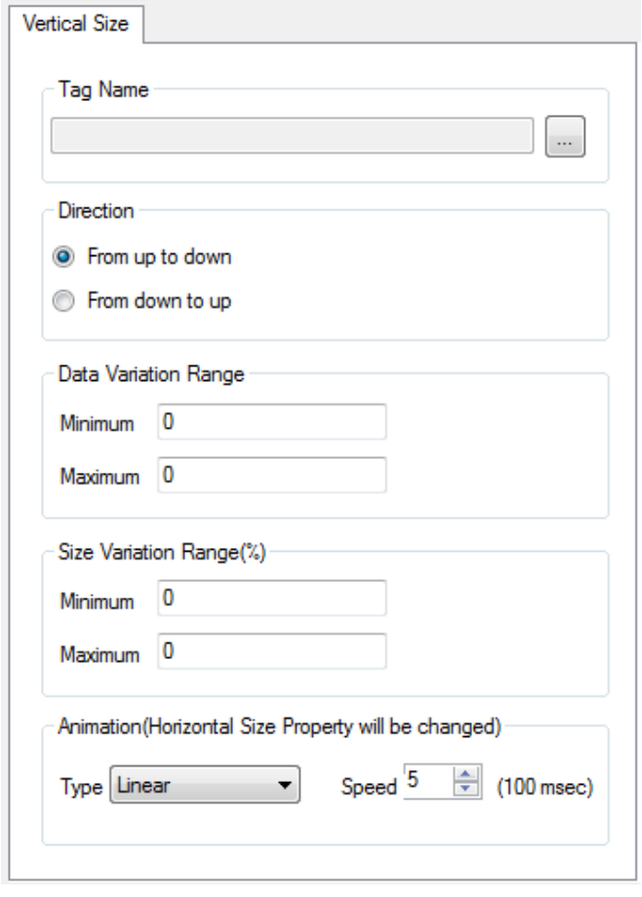
The speed indicates the time required to display the animation. The smaller the value is, the faster the dynamic properties are displayed. The input value should be between 0 and 100.

Notice

The animation property of the horizontal size dynamic property is also used for the vertical size dynamic property. Therefore, if you change the animation property of the former, the animation property of the latter will also be changed.

2) Vertical size

It sets the dynamic property whose size varies in the vertical direction.



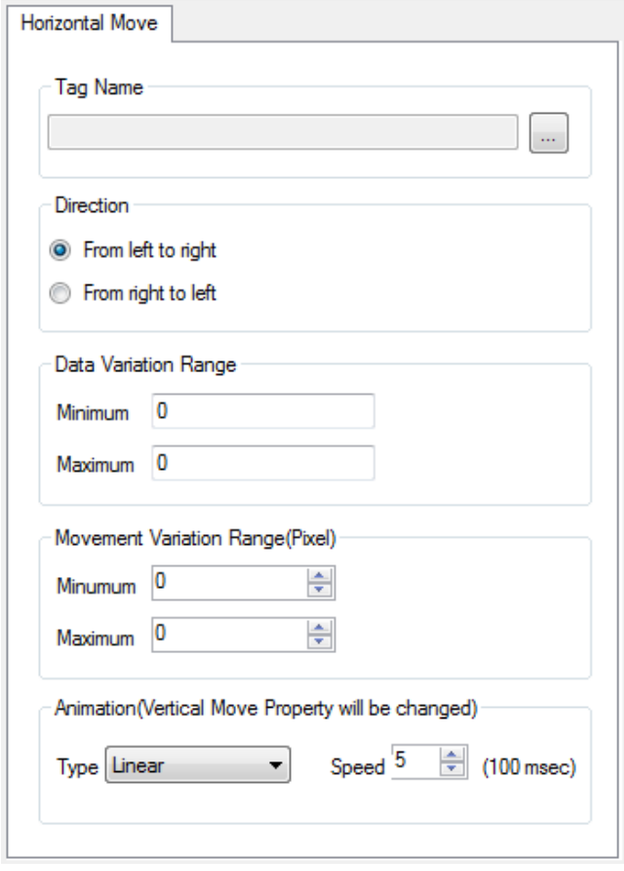
- **Tag Name**: You can select the tag that receives the input value for displaying the dynamic properties.
- **Direction**: You can select the direction to move.
- **Data Variation Range**: You can enter the range of tag input value used for dynamic properties. For example, if you set the minimum value (50), the maximum value (100), when the tag value is between 50 and 100, the dynamic properties will be displayed. Although there is no limit on the rage of input value, you need to consider the range of the input tag value.
- **Size Variation Range (%)**: You can enter the size variation range by percent(%). The input value should be between 0 and 100.
- **Animation (horizontal size property will be changed)**: You can select animation effects. Refer to the horizontal size dynamic property.

Notice

The animation property of vertical size dynamic property is also used for the horizontal size dynamic property. Therefore, if you change the animation property of the former, the animation property of the latter will also be changed.

3) Horizontal movement

It sets the dynamic property moving in the horizontal direction.



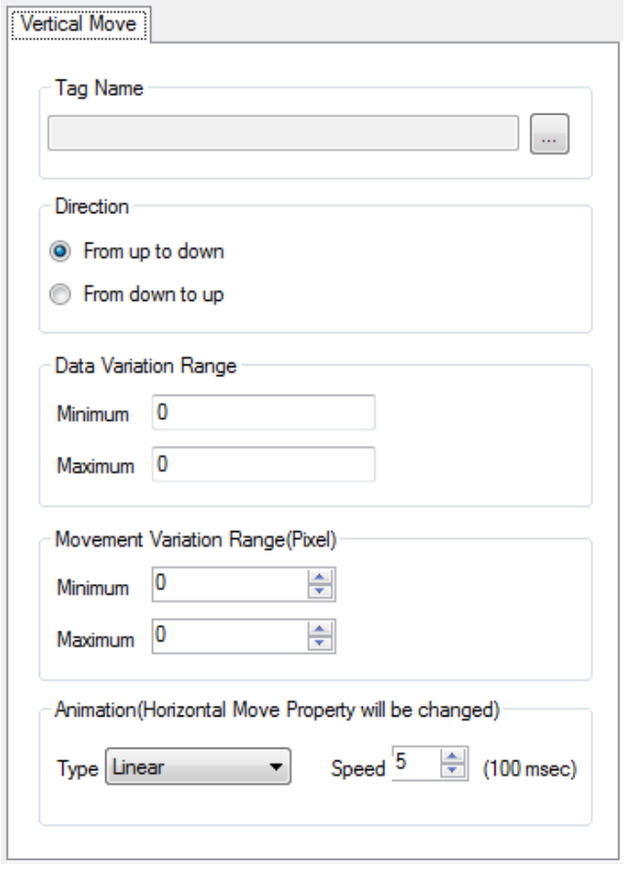
- **Tag Name**: You can select the tag that receives the input value for displaying the dynamic properties.
- **Direction**: You can select the direction to move.
- **Data Variation Range**: You can enter the range of tag input value used for dynamic properties. For example, if you set the minimum value (50), the maximum value (100), when the tag value is between 50 and 100, the dynamic properties will be displayed. Although there is no limit on the rage of input value, you need to consider the range of the input tag value.
- **Movement Variation Range (pixel)**: You can input the size variation range by pixel. There is no limit on the range of value input, however, you need to consider the screen size. If you input too large value, it will be displayed outside of the screen so the object may not be displayed on the screen or cut off.
- **Animation (Vertical move property will be changed)**: You can select animation effects.
  Refer to the horizontal size dynamic property.

Notice

The animation property of the horizontal movement dynamic property is also used for the vertical movement dynamic property. Therefore, if you change the animation property of the former, the animation property of the latter will also be changed.

4) Vertical Movement

It sets the dynamic property moving in the vertical direction.



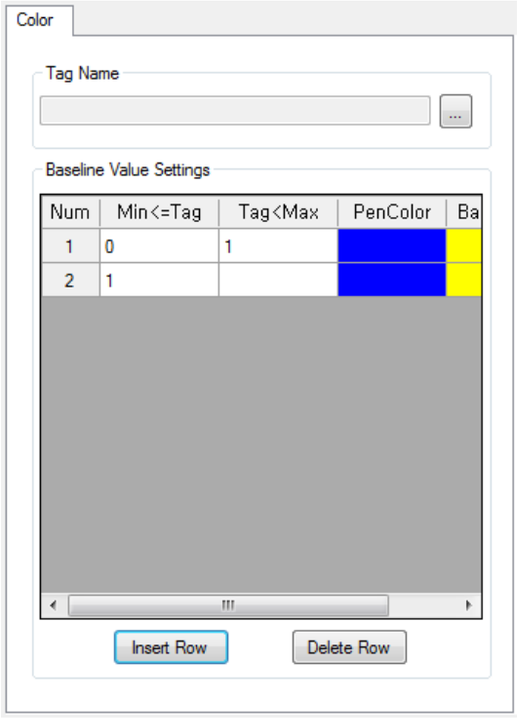
- **Tag Name**: You can select the tag that receives the input value for displaying the dynamic properties.
- **Direction**: You can select the direction to move.
- **Data Variation Range**: You can enter the range of tag input value used for dynamic properties. For example, if you set the minimum value (50), the maximum value (100), when the tag value is between 50 and 100, the dynamic properties will be displayed. Although there is no limit on the rage of input value, you need to consider the range of the input tag value.
- **Movement Variation Range (Pixel)**: You can input the size variation range by pixel. There is no limit on the range of value input, however, you need to consider the screen size. If you input too large value, it will be displayed outside of the screen so the object may not be displayed on the screen or cut off.
- **Animation (Horizontal move property will be changed)**: You can select animation effects. Refer to the horizontal size dynamic property.

Notice

The animation property of vertical movement dynamic property is also used for the horizontal movement dynamic property. Therefore, if you change the animation property of the former, the animation property of the latter will also be changed.

5) Color

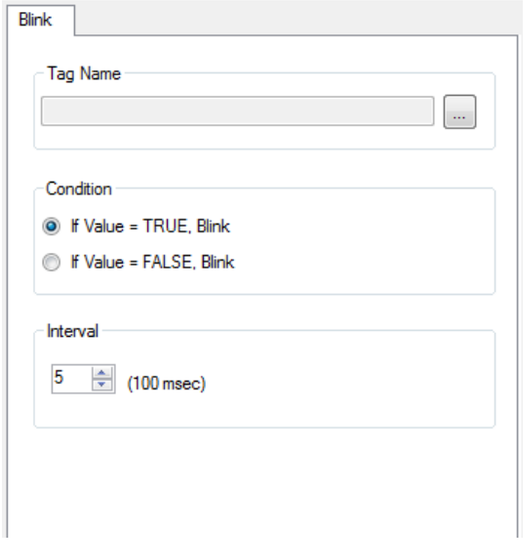
It sets the dynamic properties changing the color.



- **Tag Name**: You can select the tag that receives the input value for displaying the dynamic properties.
- **Baseline Value Settings**: Select the color to be displayed based on the value.
  If you press the "**Insert Row**" button, you can add the value range and color.
  If you press the "**Delete Row**" button, the last line will be deleted.

6) Blink

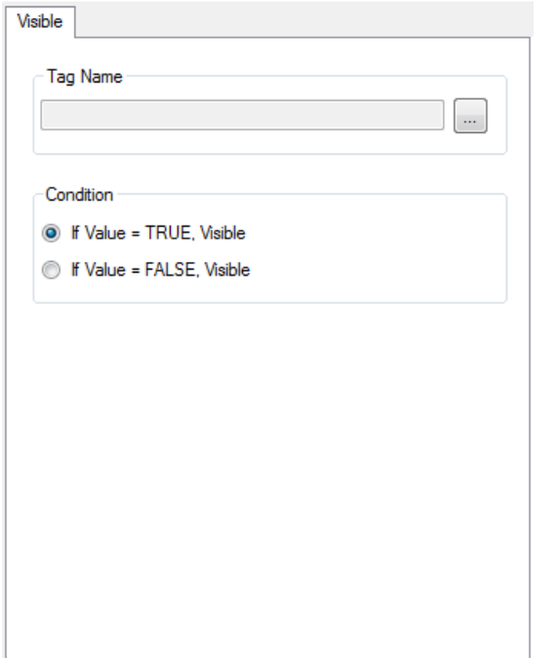
It sets the blinking dynamic properties.



- **Tag Name**: You can select the tag that receives the input value for displaying the dynamic properties.
- **Blink Condition**: You can select the condition to display the blinking dynamic properties.
- **Blink Interval**: You can select the time interval to display blink. The smaller the value is, the faster the property blinks. The input value should be between 0 and 100.

7) Visible

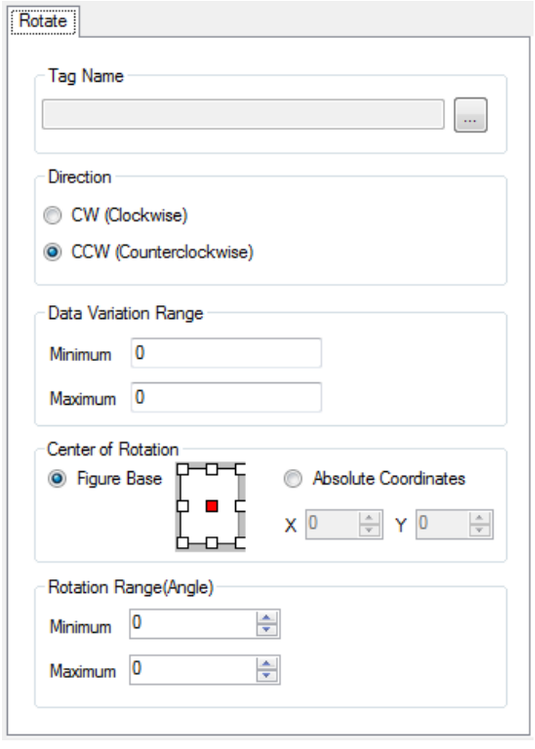
It sets the visible dynamic properties.



- **Tag Name**: You can select the tag that receives the input value for displaying the dynamic properties.
- **Visible Condition**: You can select the condition to display the visible dynamic property.

8) Rotation

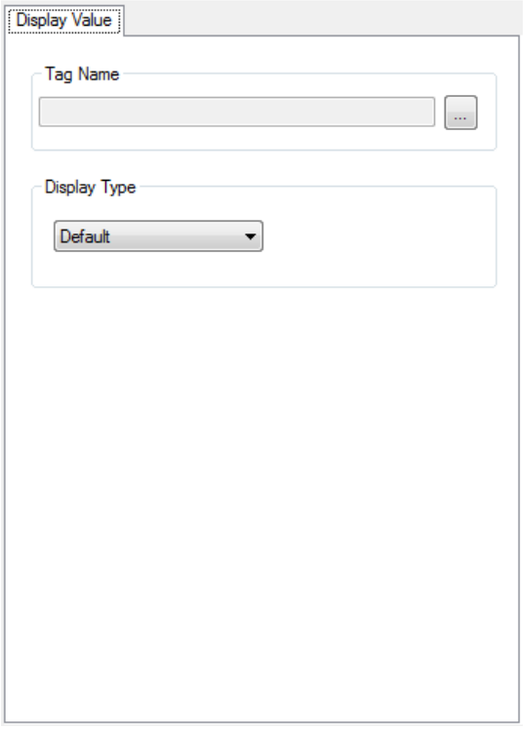
It sets the rotating dynamic properties.



- **Tag Name**: You can select the tag that receives the input value for displaying the dynamic properties.
- **Rotation Direction**: Select the rotation direction.
- **Data Variation Range**: You can enter the range of tag input value used for dynamic properties. For example, if you set the minimum value (50), the maximum value (100), when the tag value is between 50 and 100, the dynamic properties will be displayed. Although there is no limit on the rage of input value, you need to consider the range of the input tag value.
- **Center of Rotation**: Select the center of rotation. For the absolute coordinate, you need to input the value between 0 and 9999. In the case of rotation based on the absolute coordinate, depending on the position of the center point, the object rotates outside of the screen and may not be displayed on the screen
- **Rotation Range (Angle)**: You can set the rotation range according to the input value. The input value should be between 0 and 9999.

9) Display Value

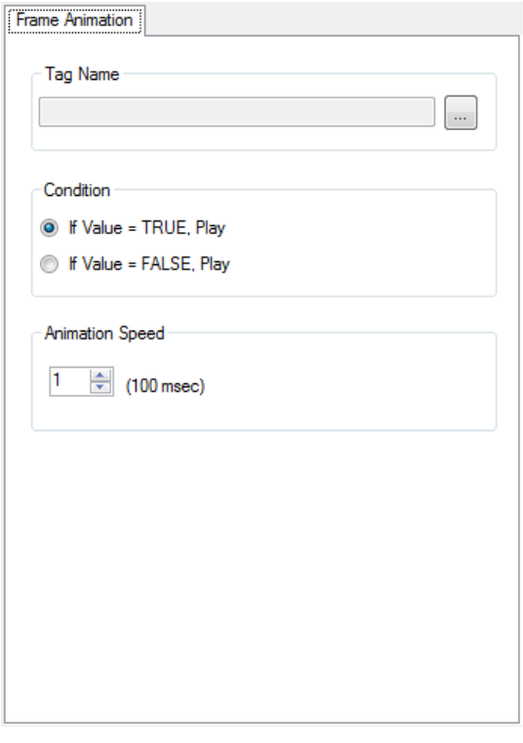
It sets the dynamic properties to display the tag value.



- **Tag Name**: You can select the tag that receives the input value for displaying the dynamic properties.
- **Value Display Type**: You can select the type to display the value.

10) Frame animation

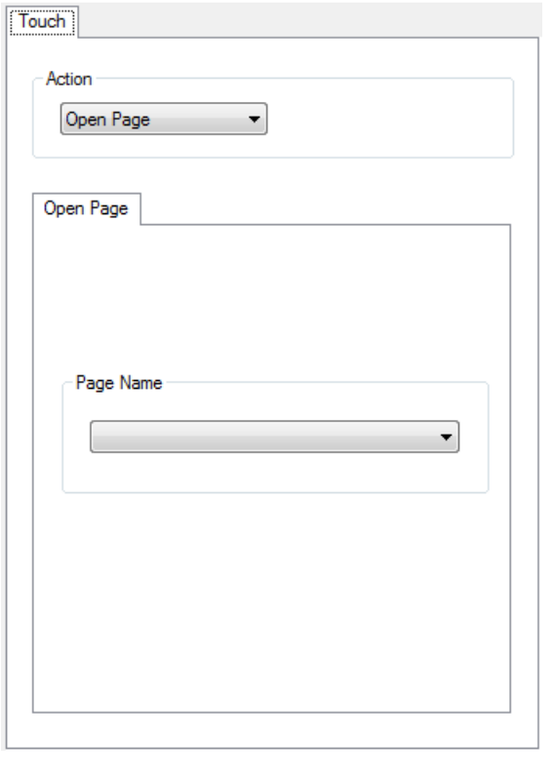
It sets the playing conditions and speed of the GIF animation.



- **Tag Name**: You can select the tag that receives the input value for displaying the dynamic properties.
- **Playing Condition**: You can determine whether paying the animation when the tag value is TRUE; when the tag value is FALSE.
- **Animation Speed**: Set the playing speed by 100 msec.

11) Touch: Open Page

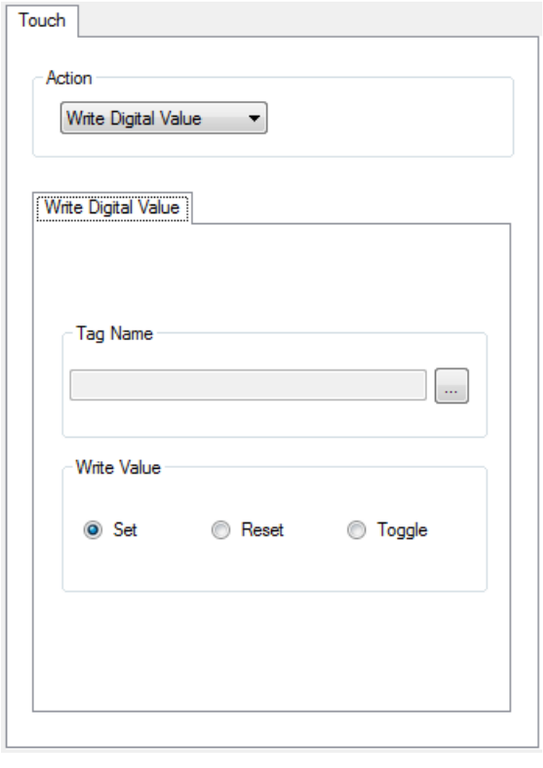
When touching the object, the screen will be turned to the set page.



- **Action**: Select "Open Page".
- **Open Page**: Among pages registered in the project, select the page to turn.

12) Touch: Write Digital Value

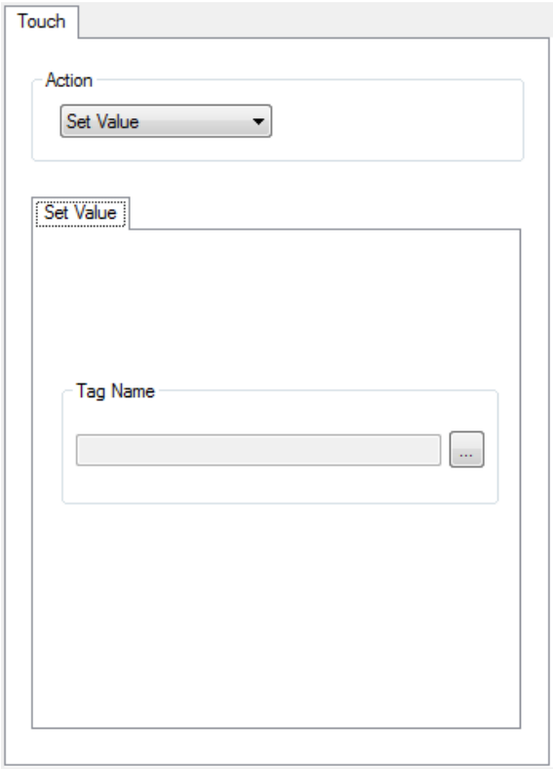
When touching the object, the tag value will be changed into the selected digital value.



- **Action**: Select "Write Digital Value".
- **Tag Name**: Select the tag to be used for writing value.
- **Write Value**: Select the type of 'Write Value'.
  Set: Writing TRUE vale
  Reset: Writing FALSE value
  Toggle: Writing the reverse of the current value

13) Touch: Set Value

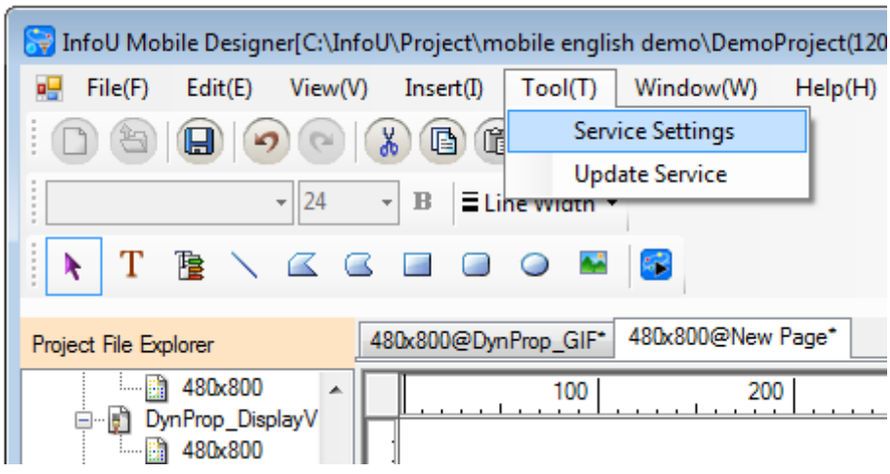
When touching the object, the tag value will be changed into the input analog value.



- **Action**: Select “Set Value”.
- **Tag Name**: Select the tag to be used for setting value.

25.1.7 Service Settings

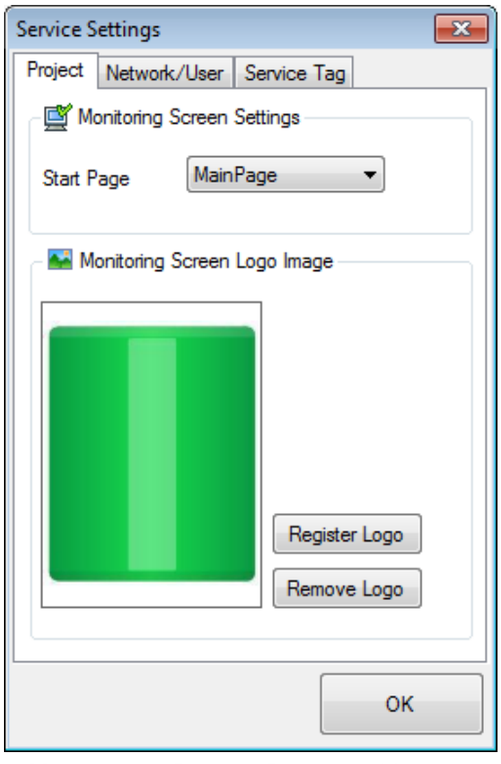
You can set the items required for the mobile project service.



Run [Tool] → [Service Settings] in the main menu.

1) Project

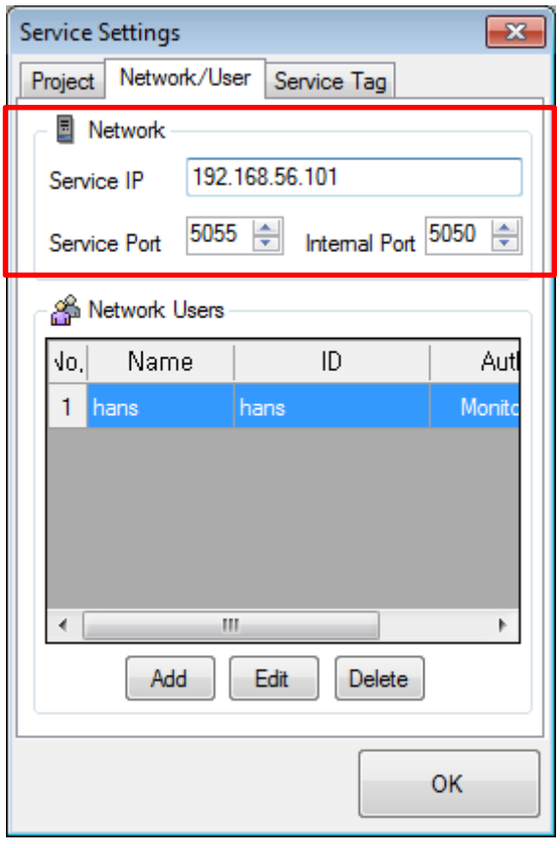
You can set the basic items for the mobile project service.



- **Monitoring Screen Settings**: You can select the initial page that will be displayed first when starting the monitoring screen among the registered projects.
- **Monitoring Screen Logo Image**: You can register the logo image in the format of bmp, gif, jpg, png, tiff. The logo image can be omitted.

2) Network

You can set the IP address and port No. of the mobile server.



- **Service IP**: You can input the service IP address or host name URL. It is used as the access address in the mobile app.
- **Service Port and Internal Port**: You can set the port No. between 1 and 65535.

Notice

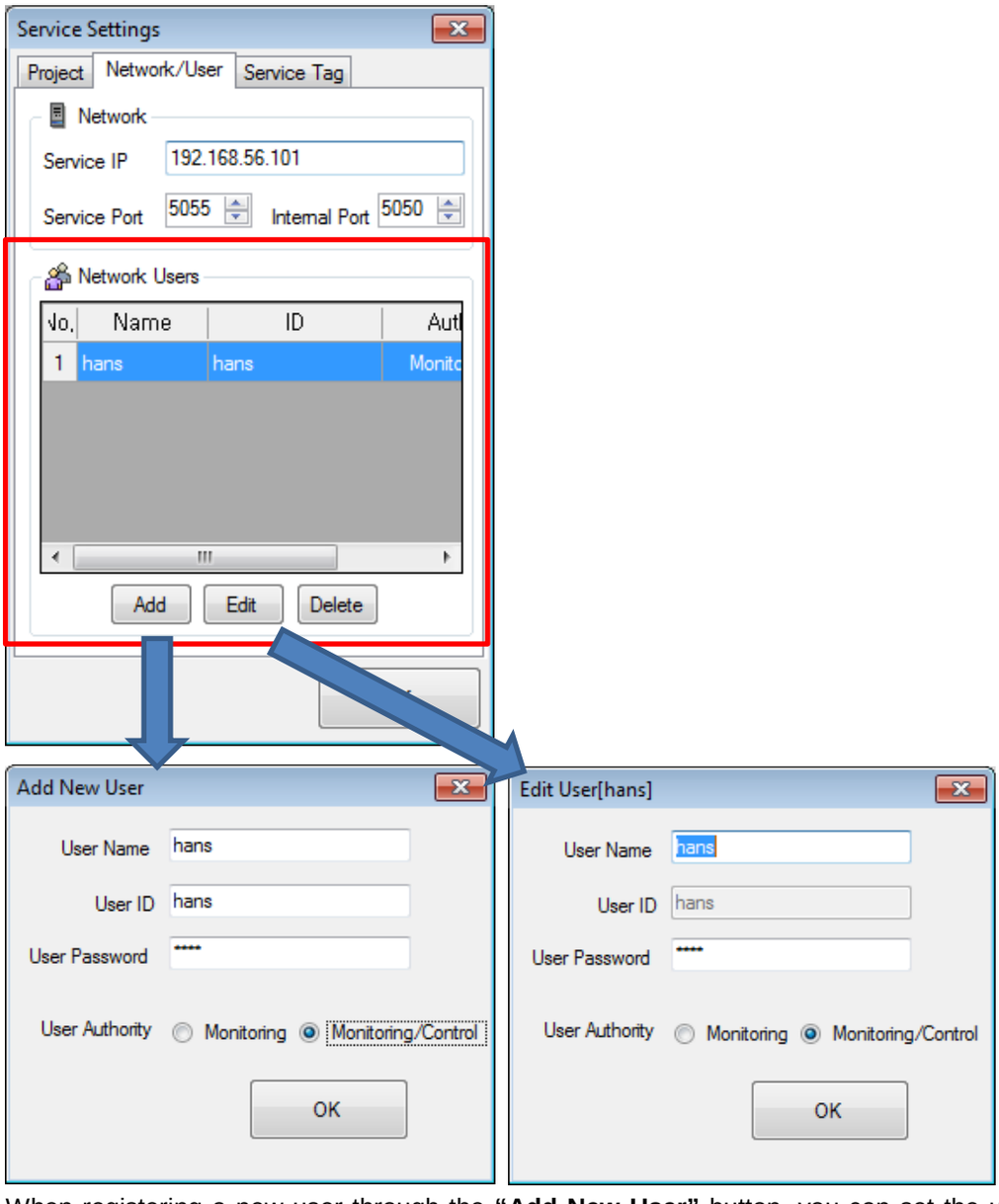
The instructions for network settings are as follows.

☞ In case the entered port No. is used in other processes, the server program does not work properly so make sure to avoid confliction of port No. during setting.

☞ Make sure to enter the service IP address.

3) User

You can set the ID and password to be used when accessing to the mobile server in the mobile Client.



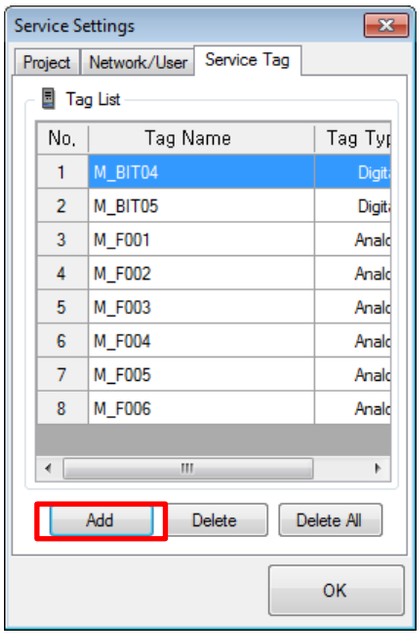
When registering a new user through the **"Add New User"** button, you can set the user name, ID and password, user authority. For the User Authority, you can choose one between Monitoring and Monitoring/Control. You can selectively "Edit" or "Delete" the registered users.

Notice

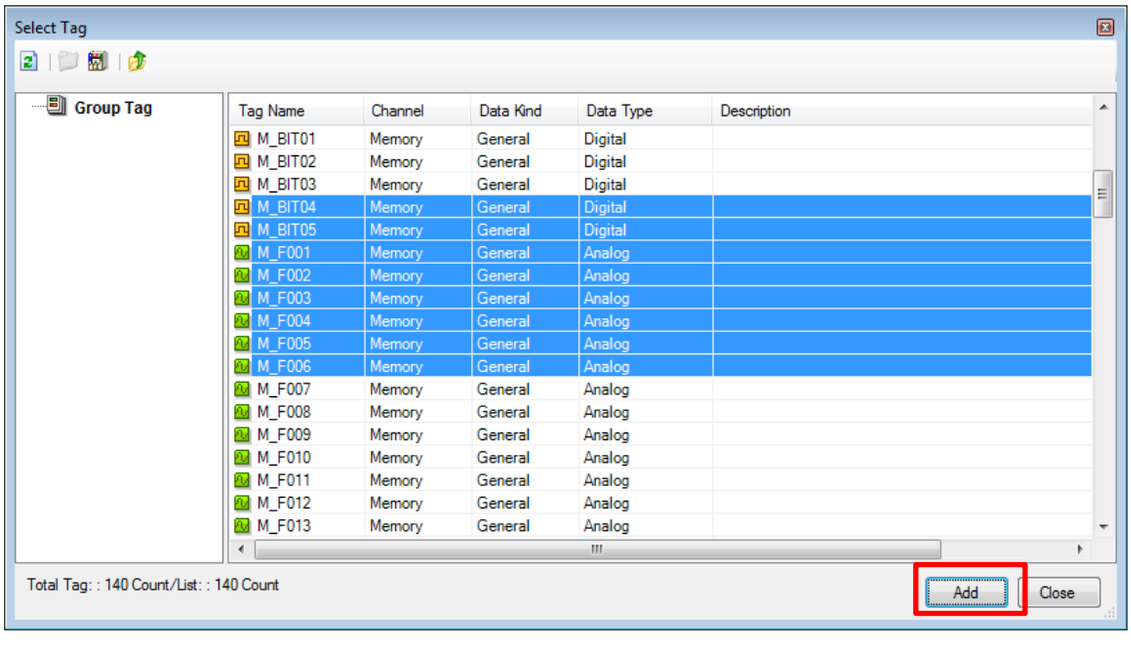
The 'User Name' is used for management only. When accessing to the mobile server, you need to "ID" and "password". "ID" can be deleted but cannot be modified.

4) Service tag

You can register the tag used for the mobile client.



If you click the **"Add"** button, the tag selection window will be displayed.



After choosing the targeted tag in the tag selection window, if you click "**Add**" button, the tag will be registered. Selecting several tags should be done with pressing the "Shift" key. The registered tags are used for monitoring tag values and creating trend groups in the mobile app.

Notice

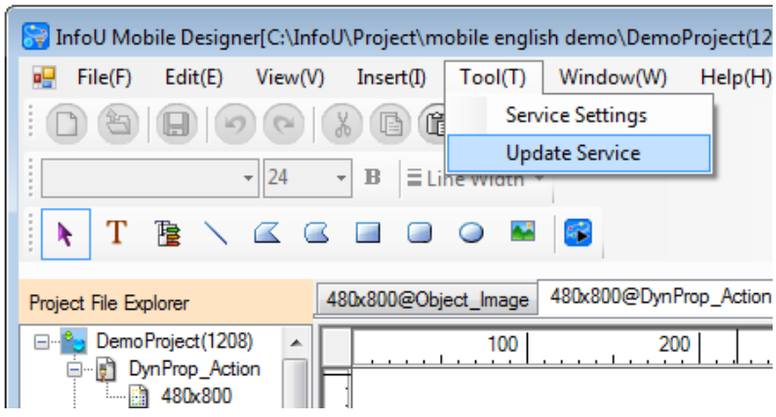
To select tags, the InfoU project containing the relevant tag should be executed in runtime or be loaded from the InfoU engineering.

25.2 Mobile Server

The mobile server provides the services to transmit the data of the mobile project and InfoU to the mobile client.

(1) Execution of the mobile server

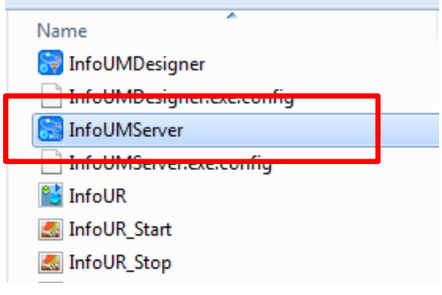
You can run the mobile server in the mobile designer.



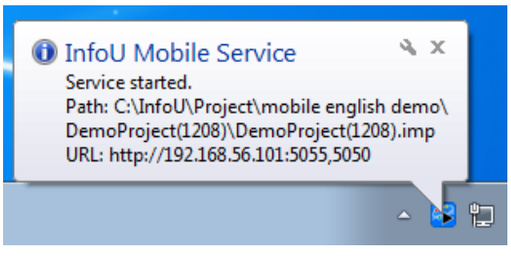


To run the mobile server select **[Tool]** → [**Service Update**] in the main menu of the mobile designer or select the "Service Update" icon in the toolbar. In this case, the project loaded to the mobile designer is serviced by the mobile server.

Otherwise, you can directly run "InfoUMServer.exe" in the InfoU\bin folder. In this case, the serviced project is the mobile project that was serviced last. If there is no service history, the server starts up only and you can select the project in the popup menu of the server.

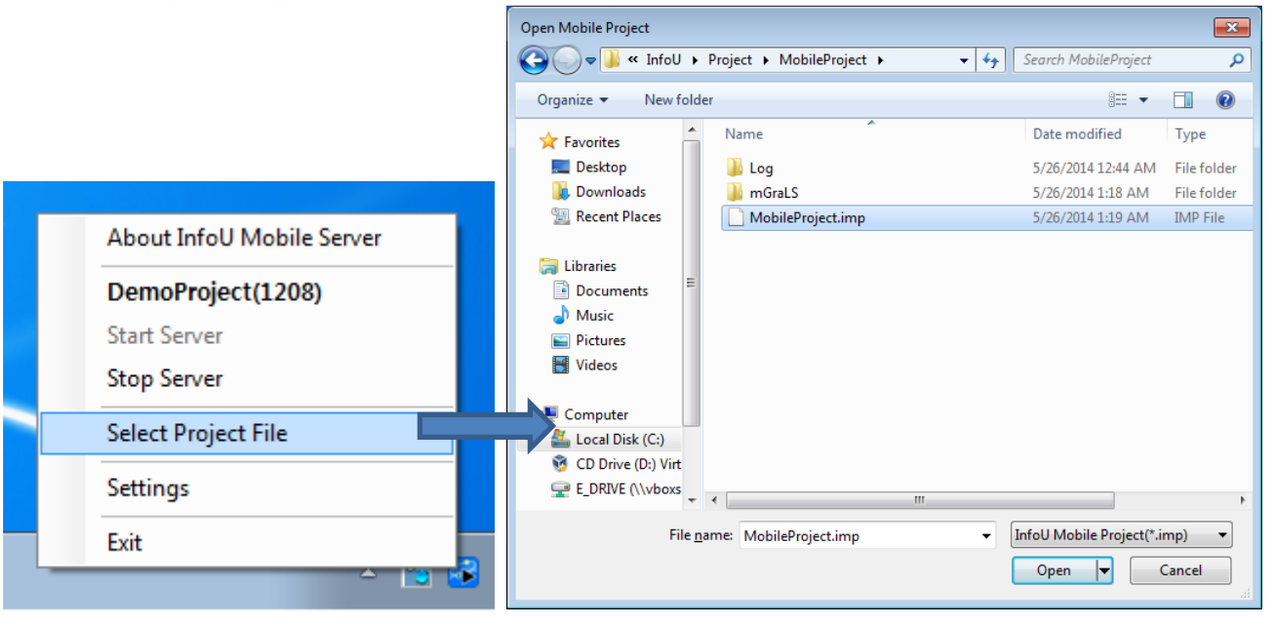


When the mobile server starts, the icon will be created in the taskbar and the starting message will be displayed.



(2) Selection of the mobile project

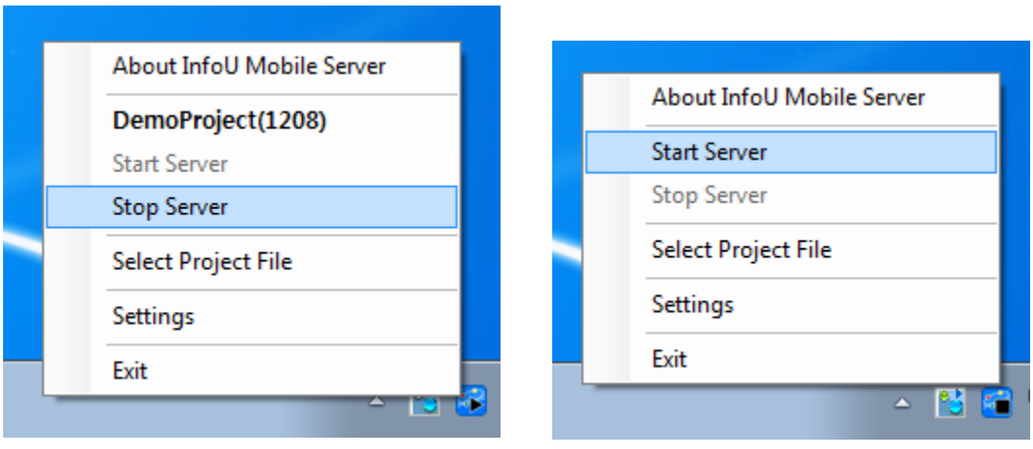
You can change the targeted project for service using 'Select Project File' menu of the mobile server.



To run the popup menu, clcik with the right mouse button on the icon of the taskbar of the mobile server. Then, run "**Select Project File**" in the popup menu. After choosing the mobile project file in the dialog box to open the mobile project, if you click the "**Open**" button, the selected project will be applied.

(3) Stop and Start of Mobile server

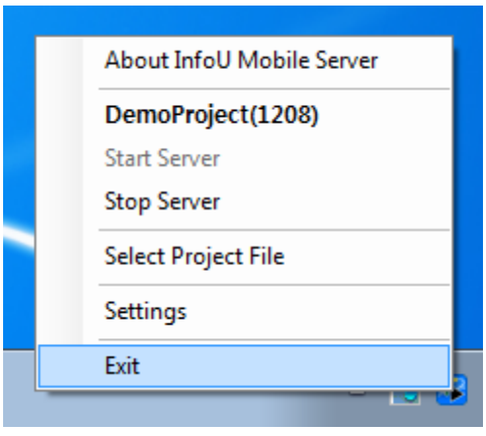
You can stop or start the mobile service.



To run the popup menu, click with the right button on the icon of taskbar of the mobile server. Then, run "**Stop Server**" in the popup menu to stop the service. For the suspended server, you can resume the service by running "**Start Server**" in the popup menu.

(4) Exit of the mobile server

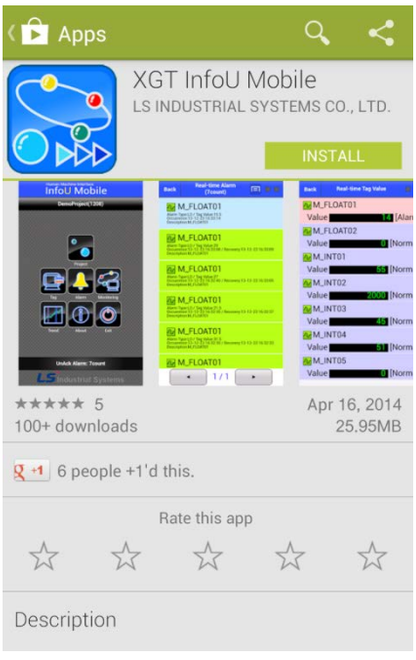
You can exit the mobile server.



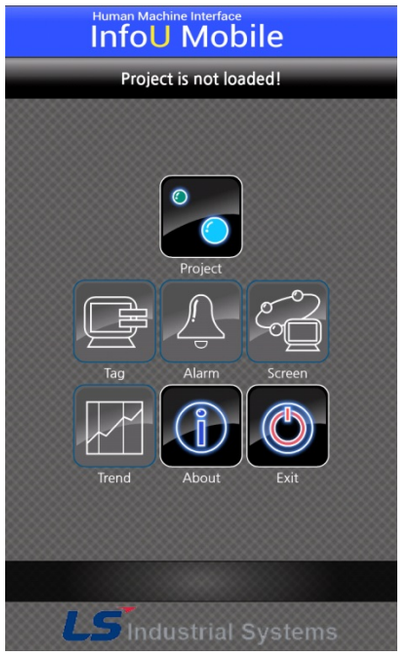
To run the popup menu, click with the right mouse button on the icon of the taskbar of the mobile server. If you run “Exit” in the popup menu, the mobile server will end.

25.3 Mobile Client

The InfoU mobile app that is installed and executed in the smartphone or tablet acts as the client for the mobile server. The mobile Client (app) receives the project information serviced by the mobile server and performs monitoring.







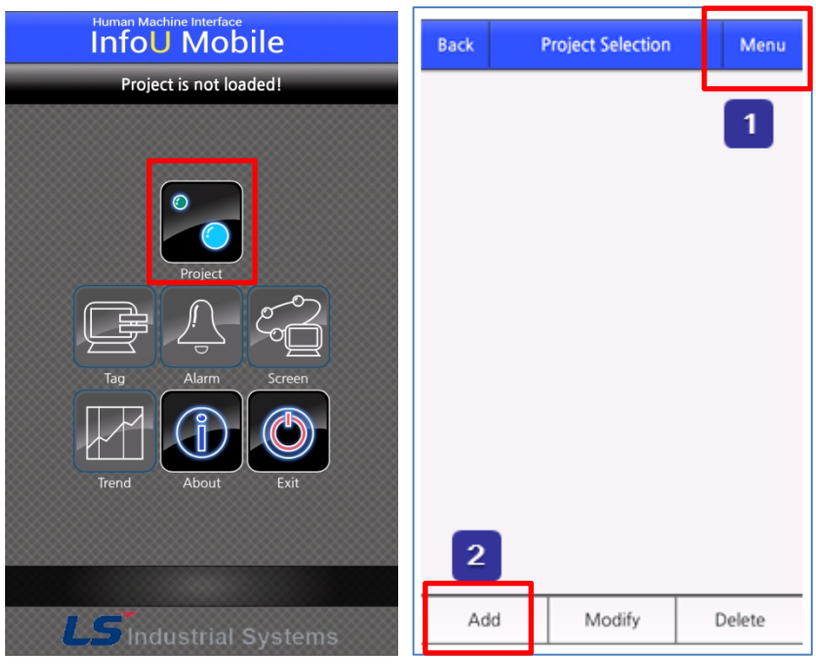
In Google Play Store, search for the "**XGT InfoU**" and find "**XGT InfoU mobile**" app, then, install it. If you run the installed "**InfoU mobile**" app by touching, the initial screen will show up.

Notice

For normal operation of the mobile client, the InfoU runtime and mobile server should be running.

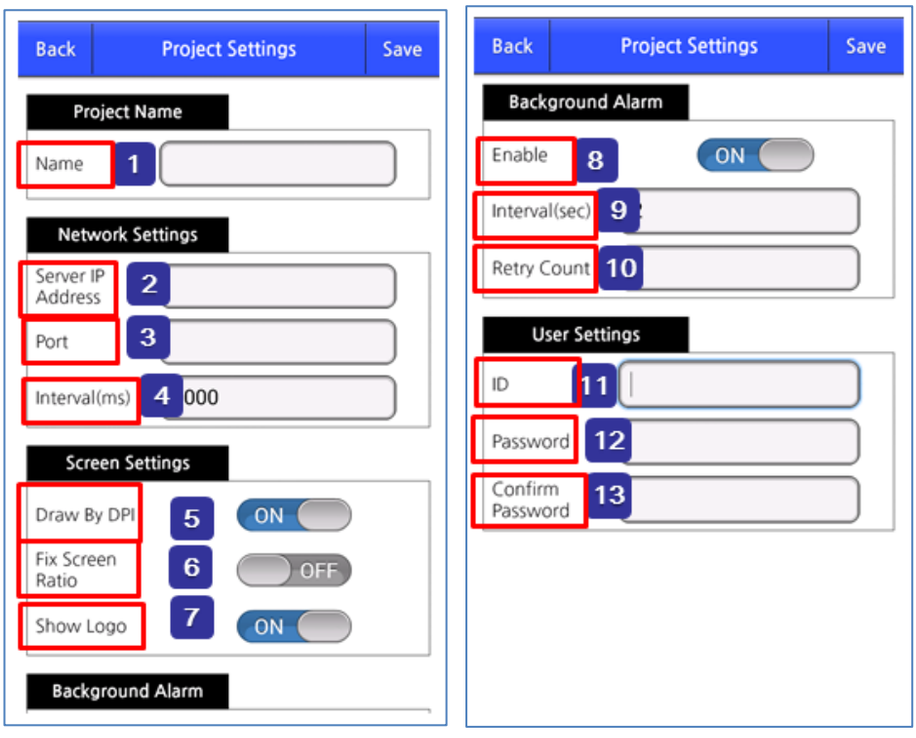
(1) Addition of the project

You can add the mobile project.



Run the "**project**" icon in the initial screen.

1	If you touch the "Menu" item in the right upper corner of the project selection screen, the menu bar will be displayed at the bottom of the app screen.
2	Run "Add" in the menu bar at the bottom of the screen.



1	**Name**: Enter the project name.
2	**Server IP address**: Enter the IP address or host name URL of the PC where the mobile server runs.
3	**Port**: Enter the set port No. into the mobile server project.
4	**Interval (communication cycle)**: Input the communication cycle of the mobile server. The input value should be between 500 ~ 10,000(0.5~10 sec.).
5	**Draw By DPI**: It displays the screen drawn in the mobile designer based DPI of the client device.
6	**Fix Screen Ratio**: It displays the screen created in the mobile designer with the proper ratio.
7	**Show Logo**: Select whether displaying the logo or not.
8	**Background Alarm**: Select whether displaying the background alarm when the mobile client program is hidden.
9	**Background Alarm Interval**: Set the cycle of checking the background alarm.
10	**Retry Count**: Set the number of times to retry when communication fails during checking the background alarm.
11	**ID**: Enter the ID for accessing to the mobile server. You can access to the mobile server only when the ID is registered in the mobile server project.
12	**Password**: Enter the password for accessing to the mobile server.
13	**Confirm Password**: To confirm the password, reenter the password.

Notice

The instructions for setting the project are as follows.

☞ The smaller the communication cycle value is, the more frequent communication with the mobile server is and the faster the updating speed is. However, please note that it may also increase the battery consumption.

☞If 'Draw By DPI' is ON, the most suitable resolution document will be displayed based on the screen size of the mobile device. If 'Draw By DPI' is OFF, the most suitable resolution document will be displayed based on the resolution of the mobile device.

(2) Deletion of the project

You can delete the mobile project.



1	Select the "**Menu**" button in the upper right corner of the project selection screen.
2	If you select "**Delete**" in the menu bar at the bottom, the set name displayed in the project list will be changed into red.
3	Select the project to delete.
4	If you press the "**Yes**" button in the information window, the project will be deleted.

(3) Addition of tag groups

You can check and control tags in real time by setting the tag group.



1	Run the "**Tag**" icon on the initial screen. The "**Tag**" icon is activated only when you access to the mobile server.
2	Choose the "**Menu**" button in the window to select the tag group.
3	Select "**Add**" in the menu bar at the bottom.
4	In the tag selection screen, check the tag that you want to check and enter the group name.
5	Save the tag group by selecting the "**Save**" button in the upper right corner of the tag selection screen.

(4) Deletion of tag groups

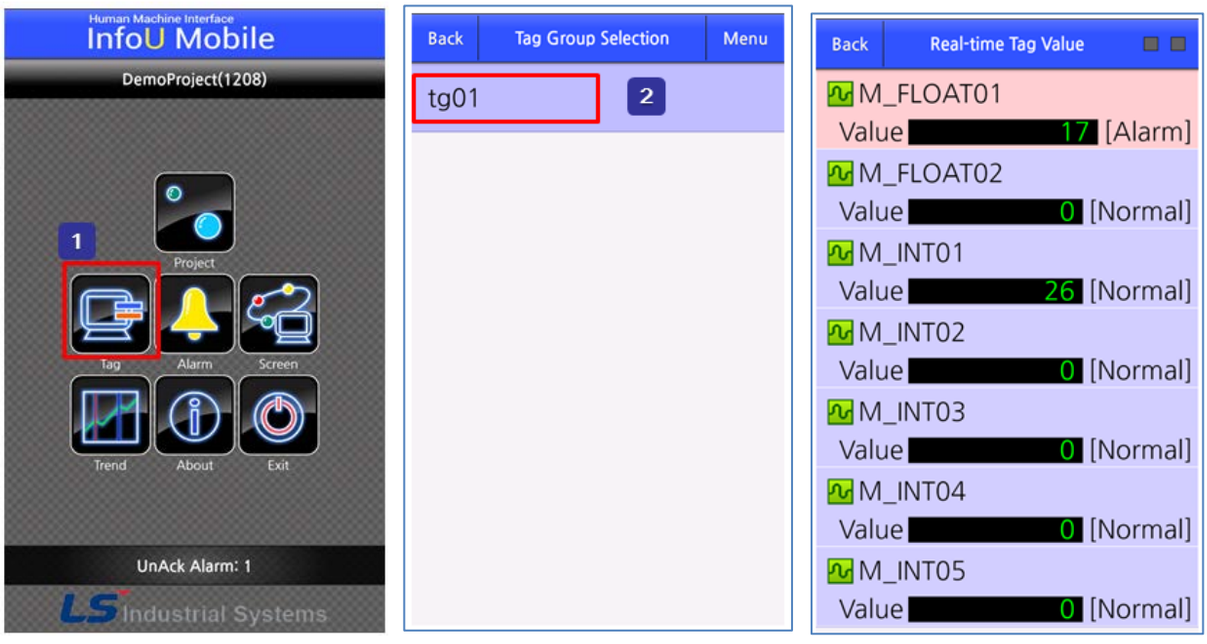
You can delete the tag group.



1	Choose the "**Menu**" button in the tag group selection screen.
2	If you select "**Delete**" in the menu bar at the bottom, the tag group name displayed in the tag group list will be changed into red.
3	Select the group to delete in the tag group list.
4	Press the "**Yes**" button in the information window to delete the tag group.

(5) Monitoring tag values

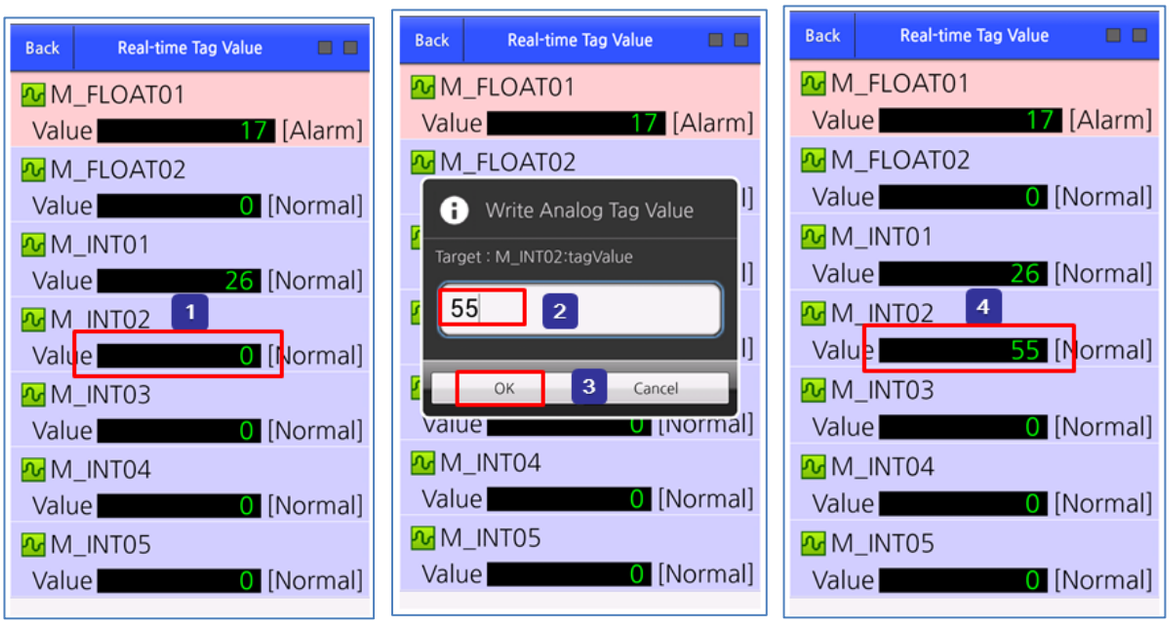
You can select the tag group and monitor the changes of tag values.



1	Run the “Tag” icon in the initial screen. The “**Tag**” menu is activated only when you access to the mobile server.
2	Choose the tag group to monitor in the tag selection screen.

(6) Control of tag values

You can control the tag values.



1	In the real-time tag value, double-touch the area where the targeted tag value is displayed	
2	Enter the desired value in the window to write the tag value.	
3	Choose the “OK” button.	
4	Then, you can see the control value is reflected.	

(7) Real-time alarm check

You can check the alarm in real time.

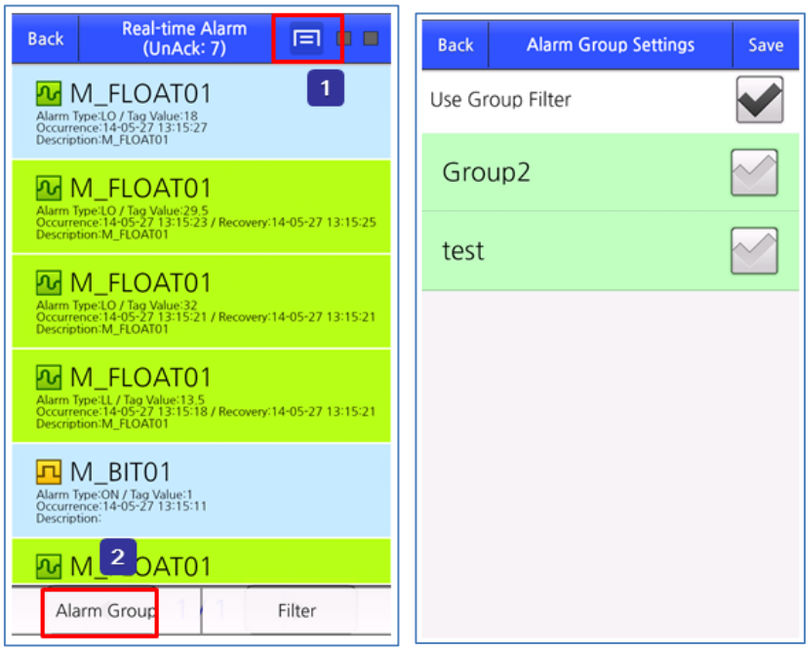




Run the “**Alarm**” icon in the initial screen. The “**Alarm**” icon is activated only when you access to the mobile server. In the Real-time Alarm window, you can check the currently occurring alarm. If you have a large number of occurring alarms, the alarms will be dividedly displayed in several pages. You can move to the next page by pressing the paging button.

1) Alarm Groups Settings

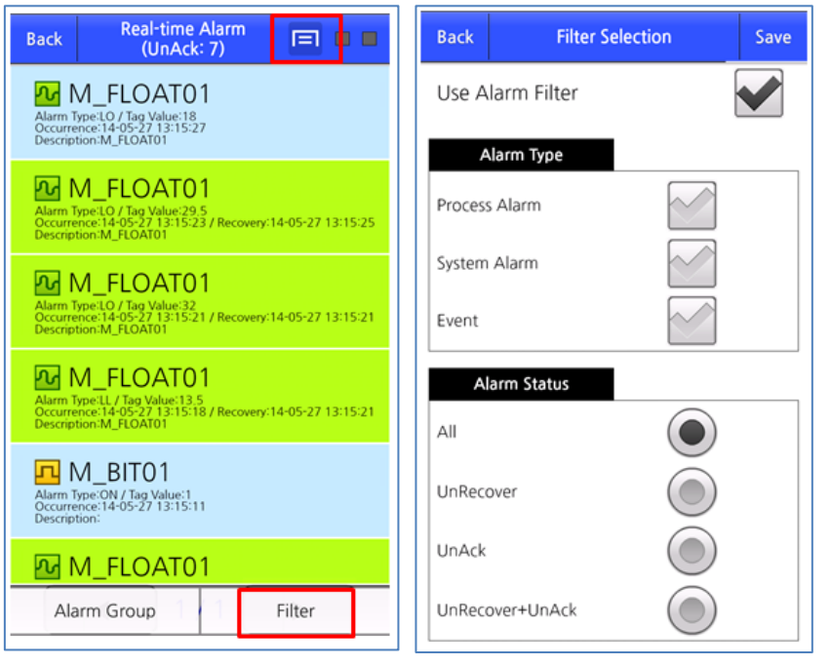
You can filter alarms by group and check them through alarm group settings.



1	If you touch the menu button in the upper right corner of the real-time alarm page, the menu bar will be displayed at the bottom.
2	When you select "**Alarm Groups Settings**" in the menu bar at the bottom, the alarm group settings window will be displayed. In the alarm group settings window, check whether using the group filter. After checking the targeted group, if you press the "**Save**" button, the details will be applied.

2) Filter Selection

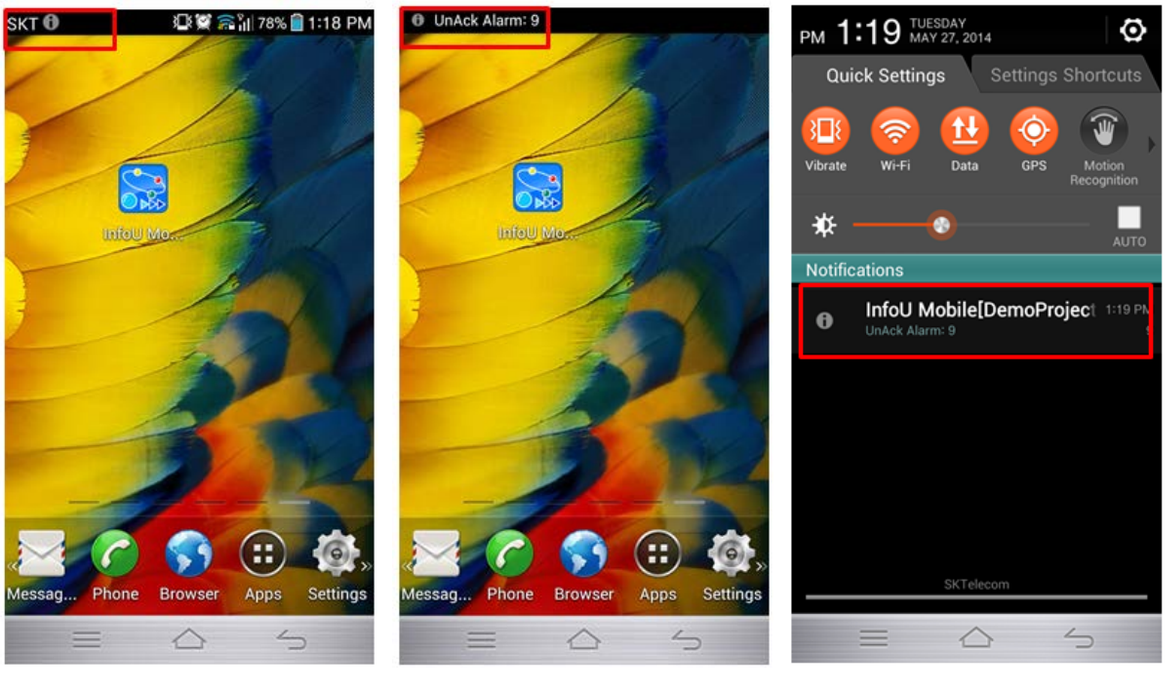
Through Filter Selection you can filter alarms by type and check them.



After setting the type and status of the alarm, if you press the "**Save**" button, the filter will be applied.

(8) Background Alarm Notice

In case the mobile client is hidden, you can get the background alarm notice. However, in order to use this function, you need to check “Use Background Alarm” in the mobile client project settings.



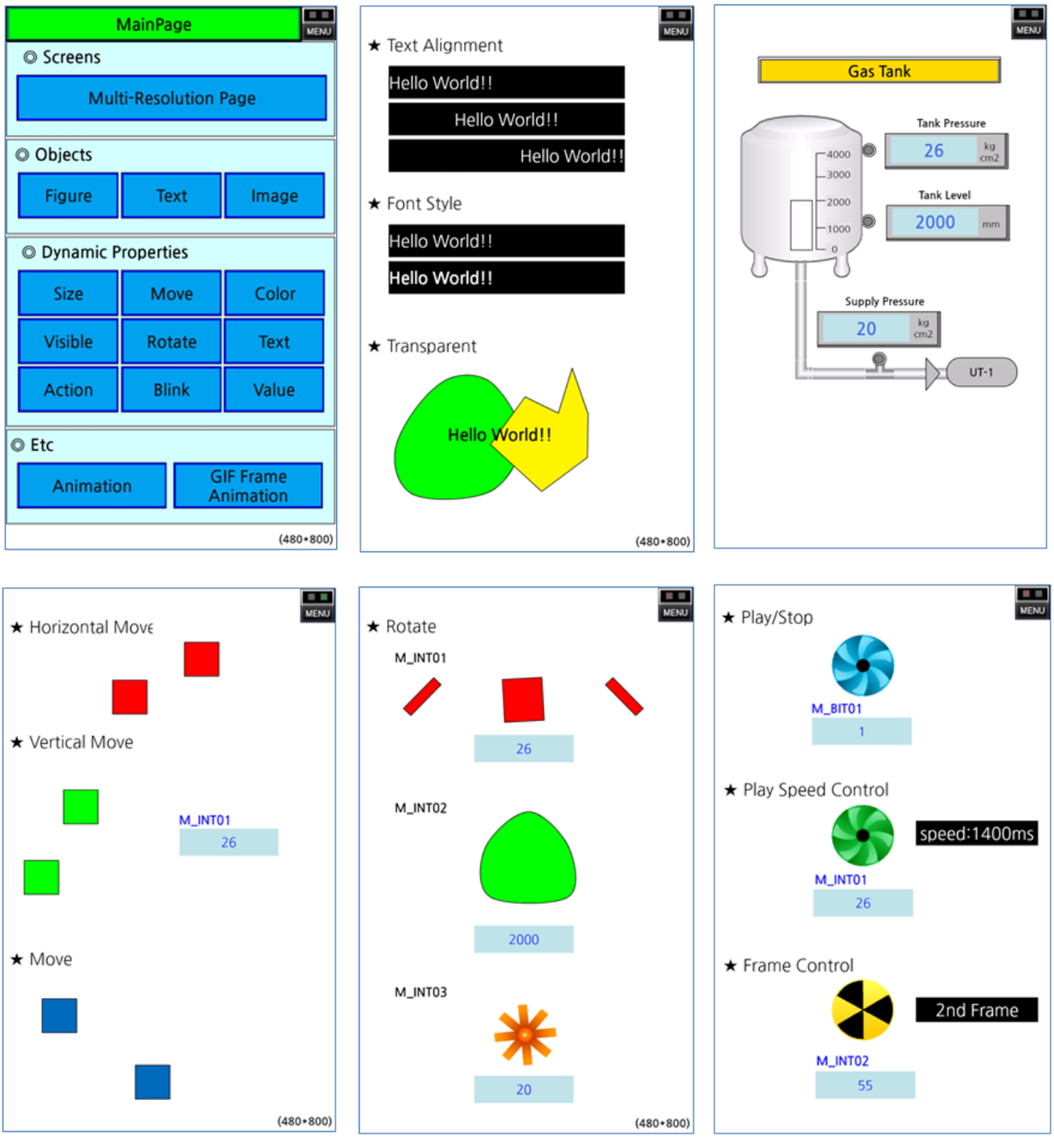
When an alarm occurs, the background alarm will be displayed as above.

(9) **Monitoring screen**

You can monitor/control the field conditions by downloading the screen drawn in the mobile designer.



Run the “**Monitoring Screen**” icon in the initial screen. The “**Monitoring Screen**” icon is activated only when you access to the mobile server. It can display various screens as below.



(10) Trend

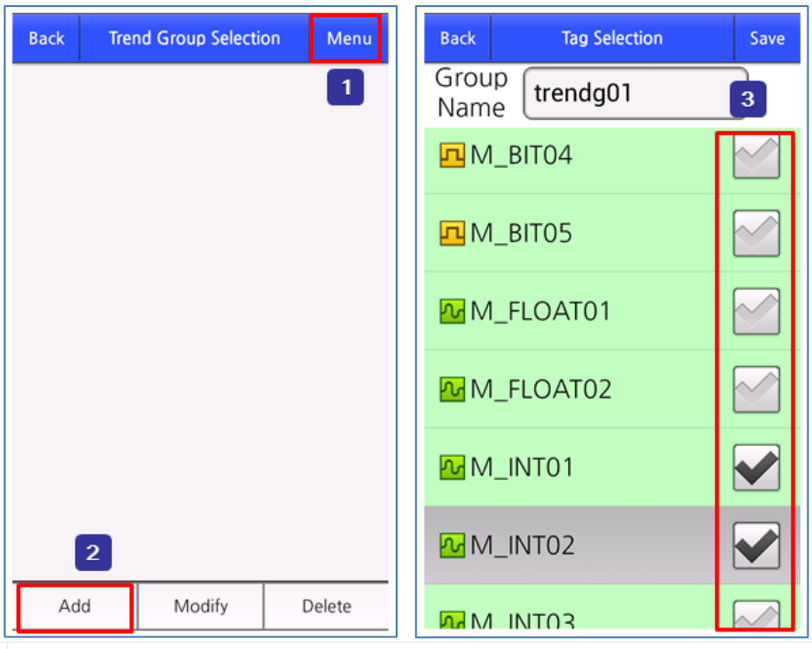
You can monitor the variations of the tag data in real time by setting the trend group.



Run the "**Trend**" icon in the initial screen. The "**Trend**" icon is activated only when you access to the mobile server. If you select the desired trend group, the trend monitoring screen will be displayed.

1) Addition of trend groups

You can add the trend groups.



1	If you choose the '**Menu**' in the trend group selection page, the menu bar will be displayed at the bottom.
2	If you select "**Add**" in the menu bar at the bottom, the tag selection page will show up.
3	Enter the group name and choose the tag to monitor. Then, if you press the "**Save**" button, the group will be created. The number of tags that you can choose is up to 2EA.

2) Deletion of trend groups

You can delete the trend group.



After selecting the "**Menu**" on the top, press the "**Delete**" button in the menu bar at the bottom. Select the trend group to delete. If you press the "**Yes**" button in the information window, the selected trend group will be deleted.

(11) Information

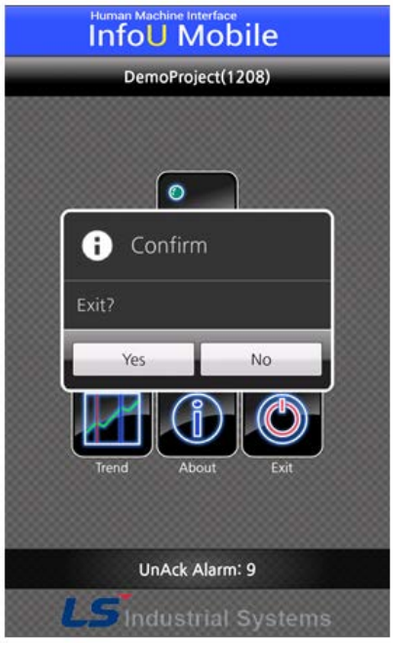
If you run the "Information" icon in the initial screen, you can check the version information of the mobile client.



(12) Exit

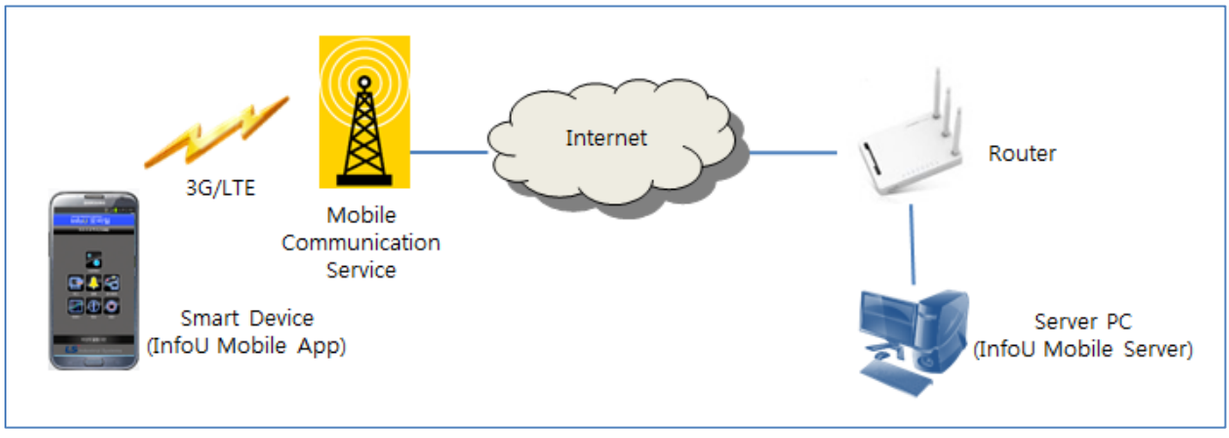
Run the "**Exit**" icon in the initial screen. If you press the "**Yes**" button in the information window, you can terminate the mobile client.





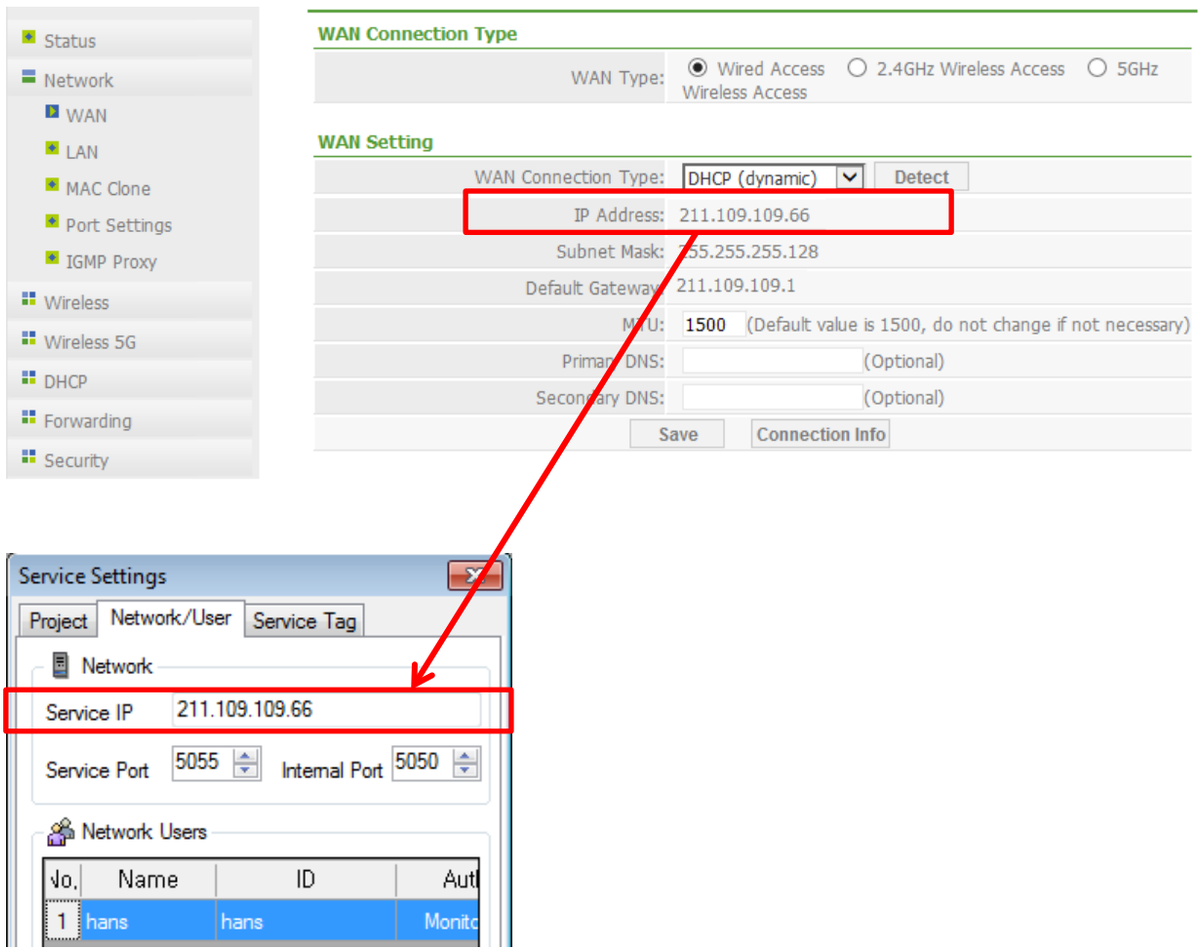
25.4 Mobile Network Settings

When operating the service with a router, you can set up the service as follows.



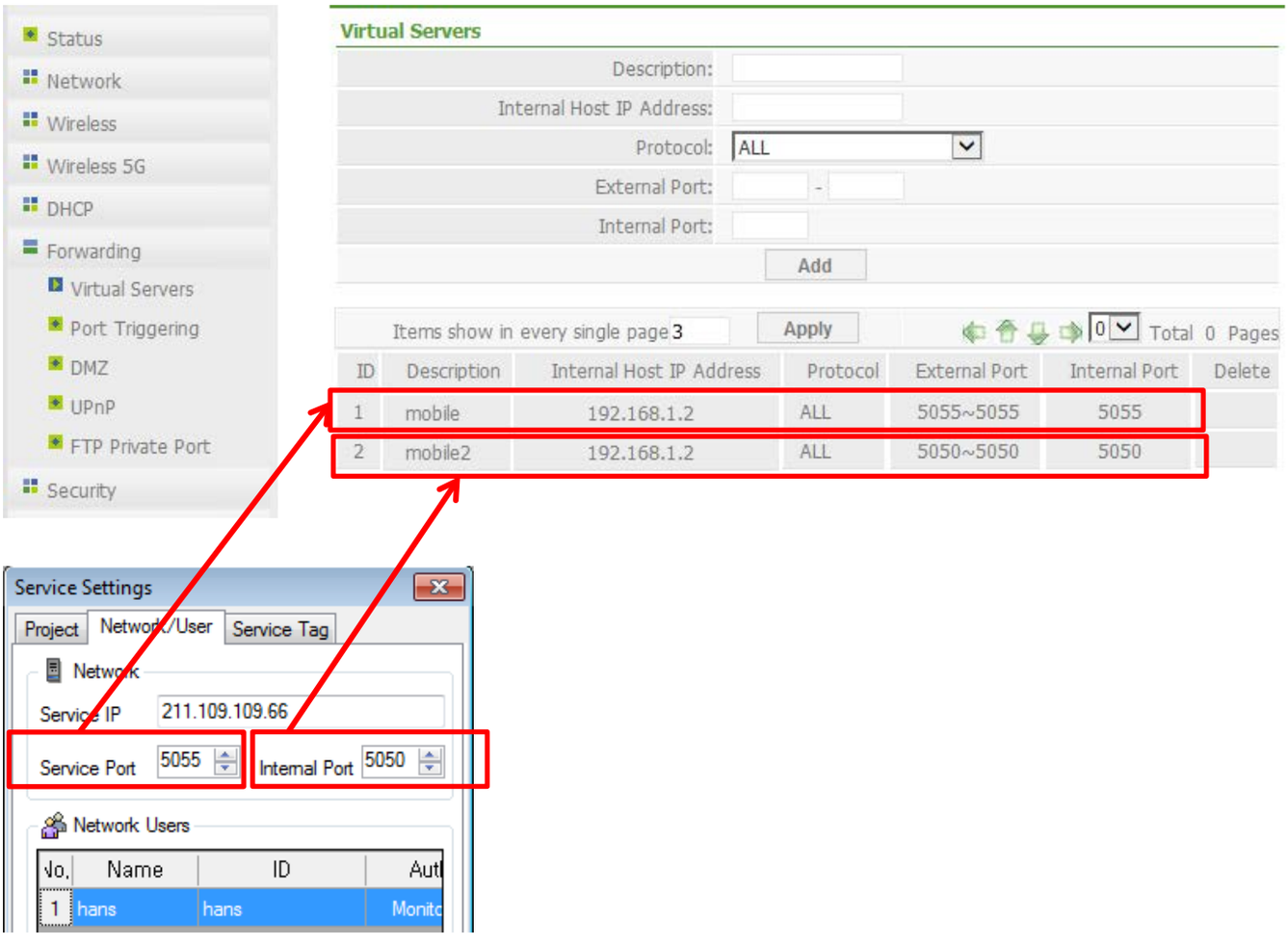
(1) IP setting for the mobile project service

For the IP address to set the mobile project service, enter the external IP address of the router. When using the DDNS, enter the host name URL. (Example of entering the host name URL: mserver.iptime.org)



(2) Setting port-forwarding of the router

For example, when accessing to the 5055 service port from the outside, set up the forwarding to 5055 port of the internal IP 192.168.0.70. The internal IP is the private IP address of the server PC where the InfoU mobile server is installed, which is given by the router. It can be checked by running the ipconfig in the command window. The internal IP should be handled so as not to be changed through manual settings in the router. Otherwise, in the network address property, you can enter the internal IP manually to fix it.

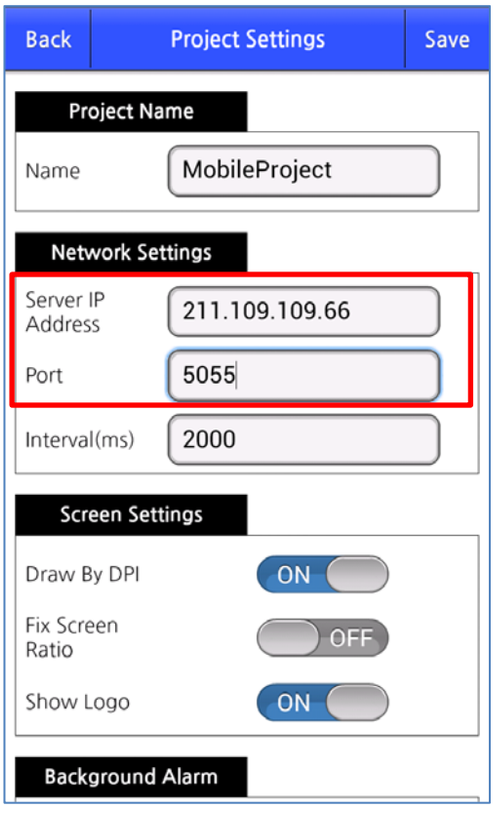


The port-forwarding should be respectively set up for the service port and internal port.

The internal port of service settings is used to update the service of the mobile server in the mobile designer.

(3) Mobile app address settings

For project settings of the mobile app, enter the IP address of the router for the server IP address and enter the service port where port-forwarding is done for the port value. If the service address is host name URL, enter the relevant URL into the server IP address. Ex) mserver.iptime.org



Appendix 1 Modbus Server

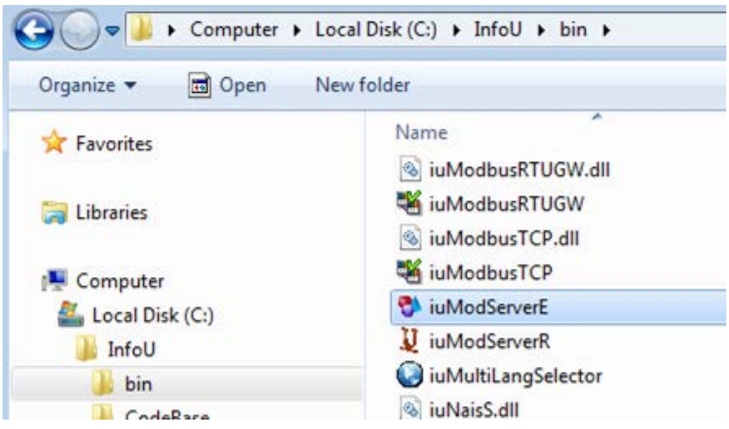
The Modbus sever program sends the tag information of the InfoU using the Modbus protocol. The editor and runtime (run-time module) are separated.

It targets the current project of the InfoU so if there is no using project, you cannot execute the Modbus sever.

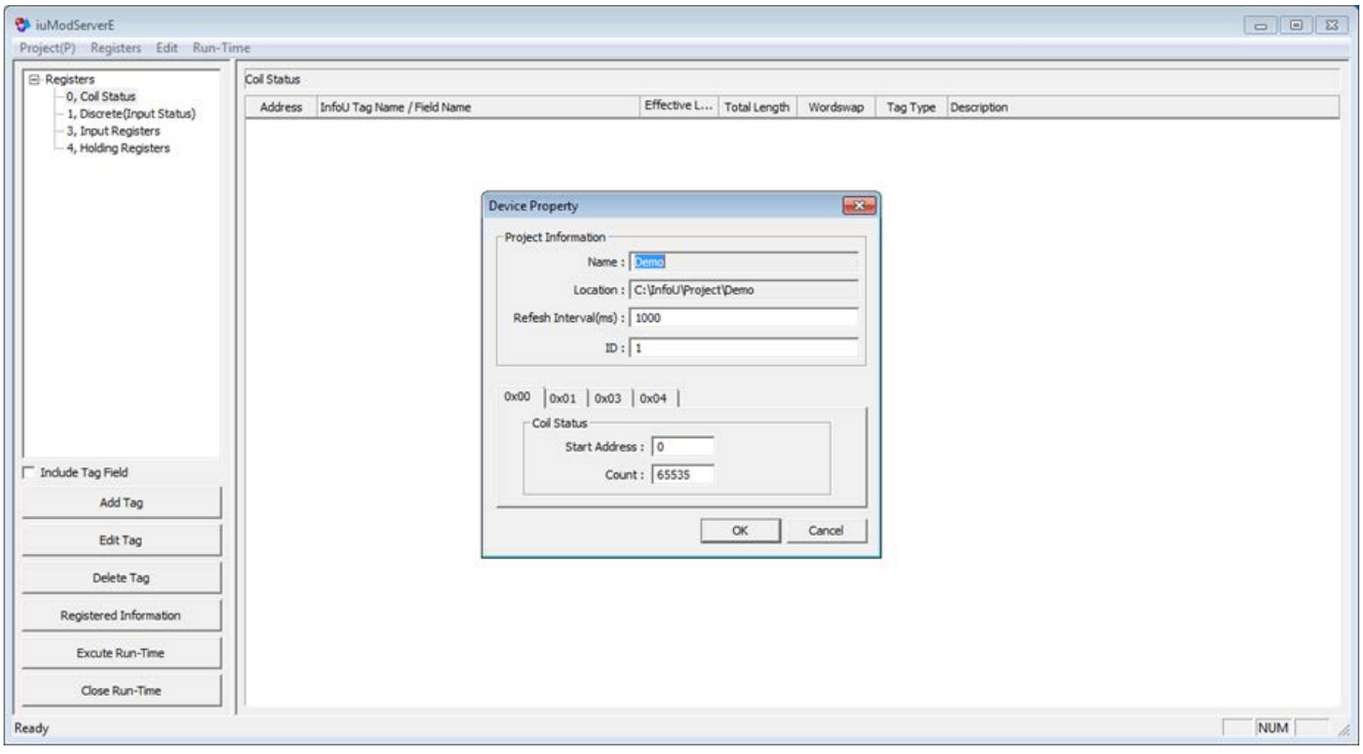
1.1 Modbus Server Editor

1.1.1 Execute Editor

Run the iuModServerE.exe program in the InfoU\bin folder.

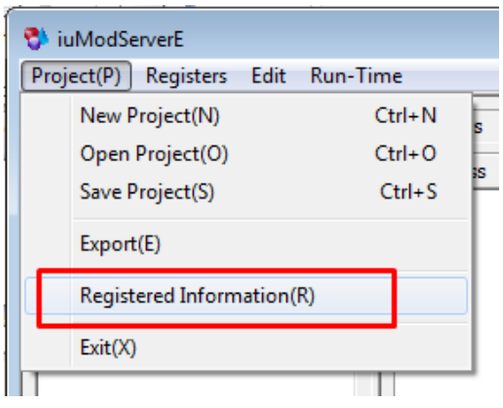


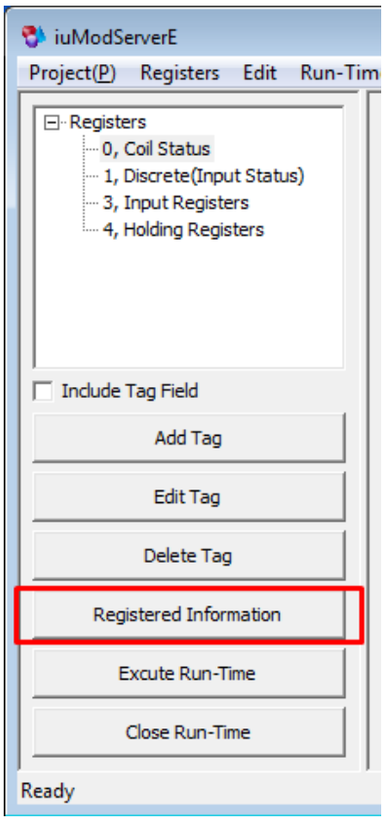
If you run the editor, the below screen will show up. When executing the editor for the first time, there is no registered information (Device Property) so the window where you can input the registered information will also be displayed.



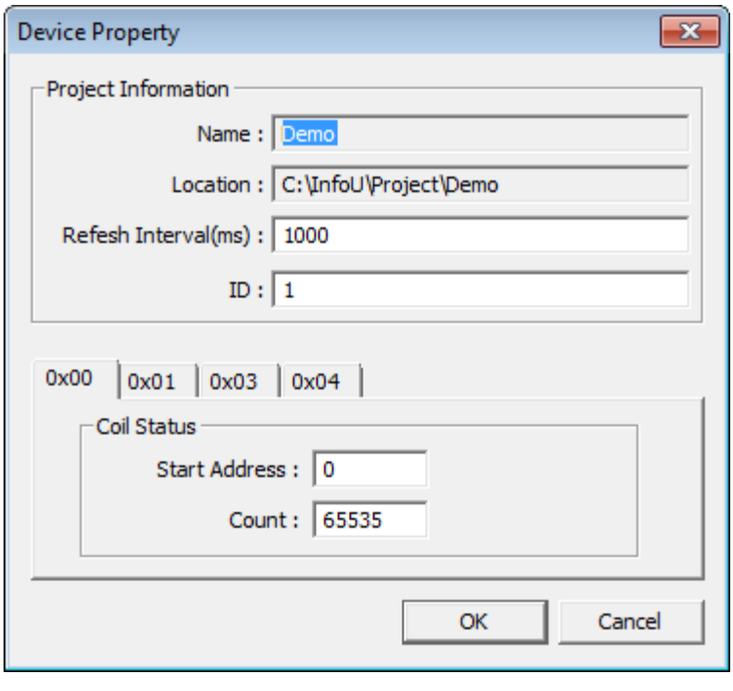
1.1.2 Registered Information

If you select [Project] → [Registered Information] in the upper menu or click the “registered information” button in the lower left side, you can run the ‘Registered Information’ window. You can also edit/enter the registered information by executing the ‘Registered Information’ window in the same way.





The composition of the registered information window is as below.



● Project settings (Project Information)

It displays the information of the currently used InfoU project.

- Name: Name of the currently used InfoU project
- Location: Location where the currently used InfoU project is stored
- Refresh Interval (ms): Data update cycle. It is entered in 1/1000 sec. (msec).
- ID: InfoU Modbus Server's ID

Notice

☞ You cannot change the InfoU project with this editor. To change the InfoU project, after changing the project in InfoUD, you need to rerun this editor.

● Register settings (Coil Status)

The Modbus Register used in InfoU are as below.

- 0x00: Coil Status (digital)
- 0x01: Discrete Inputs (digital)
- 0x03: Input Registers (analog)
- 0x04: Holding Registers (analog)

You can respectively specify the Start Address and Count for the above registers.

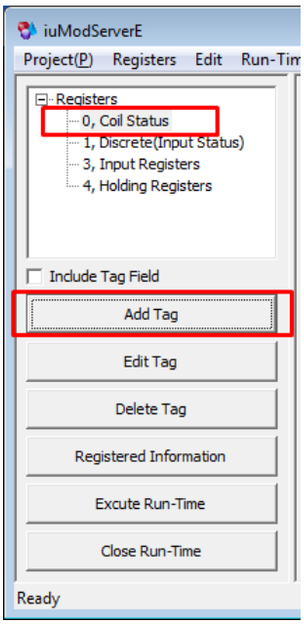
- Start Address: Start No. of the Register
- Count: Number of Registers

Notice

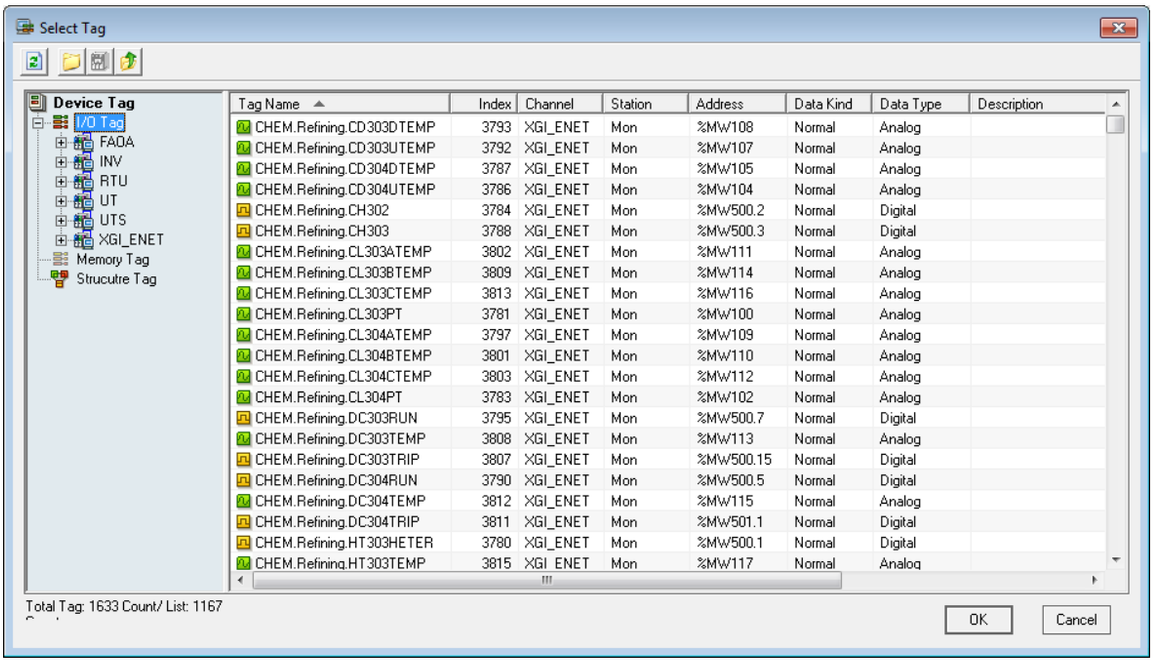
☞ If you change the number of registers, the DB of the register will be reset so you must be careful. You are recommended to preset the sufficient number of registers before entering them.

1.1.3 Tag Setting

(1) Add Tag



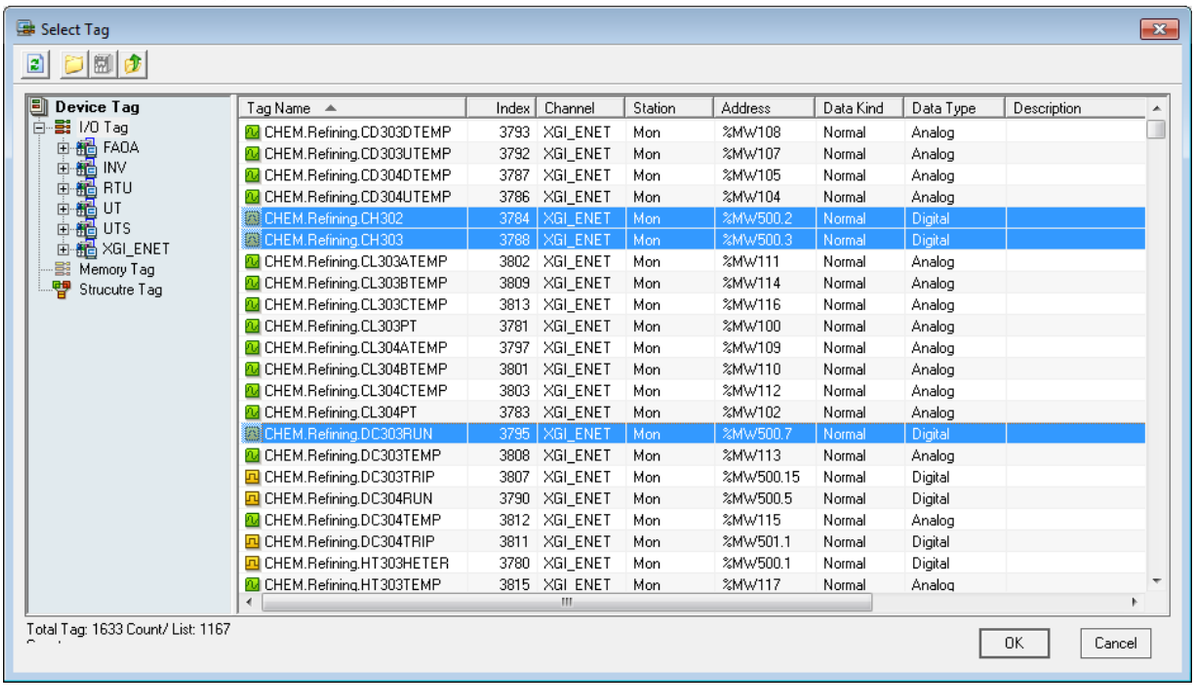
If you select the register to edit and click the 'Add Tag' button, the tag selection window will be executed as below.



Notice

- For the Coil Status(0x00) and Discrete Inputs(0x01) Register, you can input the digital data only.
- For the Input Registers(0x03) and Hold Registers(0x04), you can input the analog data only.

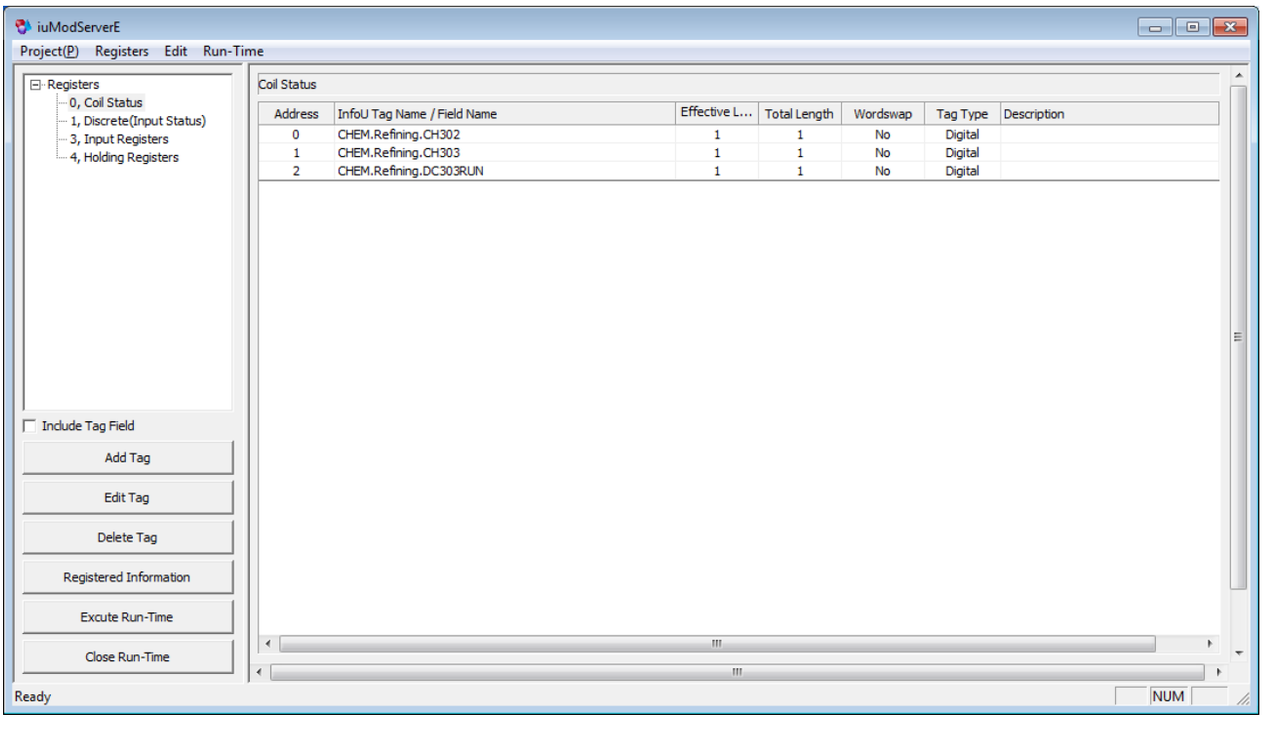
Select the tag that you want to register and click the OK button.



Notice

☞ If “Include Tag Field” is checked in the left menu, you can select just one tag. For multi-selection of tags, “Include Tag Field” should be unchecked.

You can see three tags are input to the Coil Status Register. At this time, the start address is automatically input considering the tag size every time the tag is added.

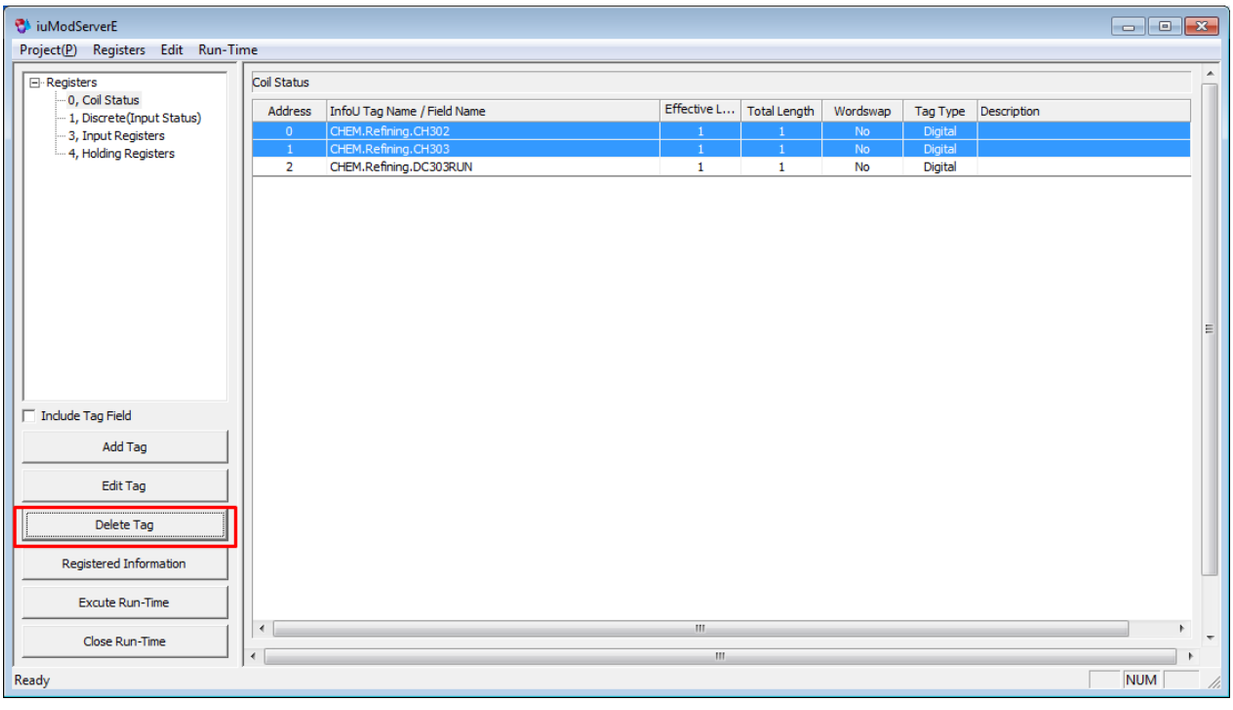


In the above figure, the start address (Register No.) of each tag is as below.

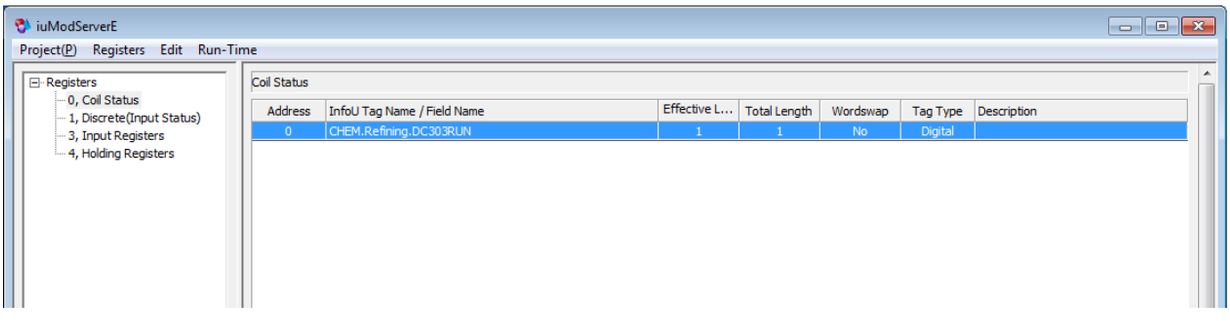
Register No.	Tag
0	Monitoring screen.DEMO_monitoring screen_StationOK
1	Monitoring screen.Dig1
2	Monitoring screen.Dig2

(2) Delete Tag

After selecting the tag that you want to delete from registered tags, if you click the 'Delete Tag' on the right side, the tag will be deleted.

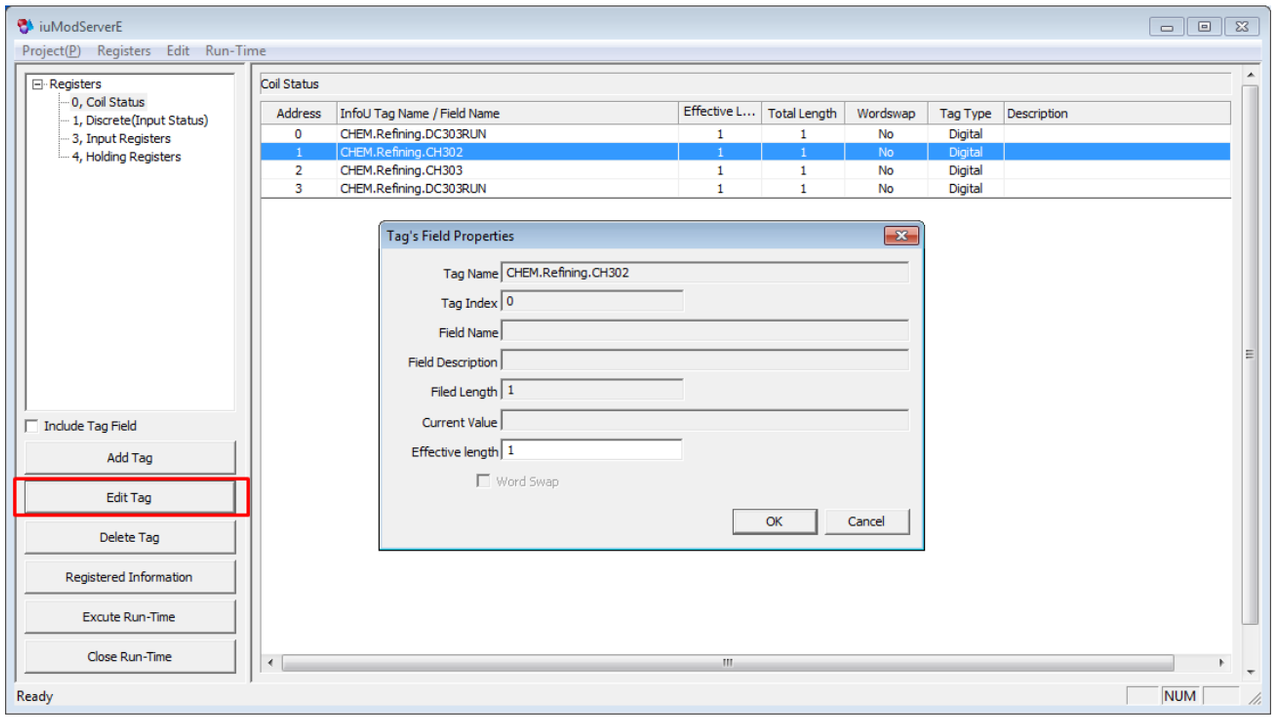


After deletion, the start address of the remaining tags are automatically modified considering the tag size.



(3) Edit Tag

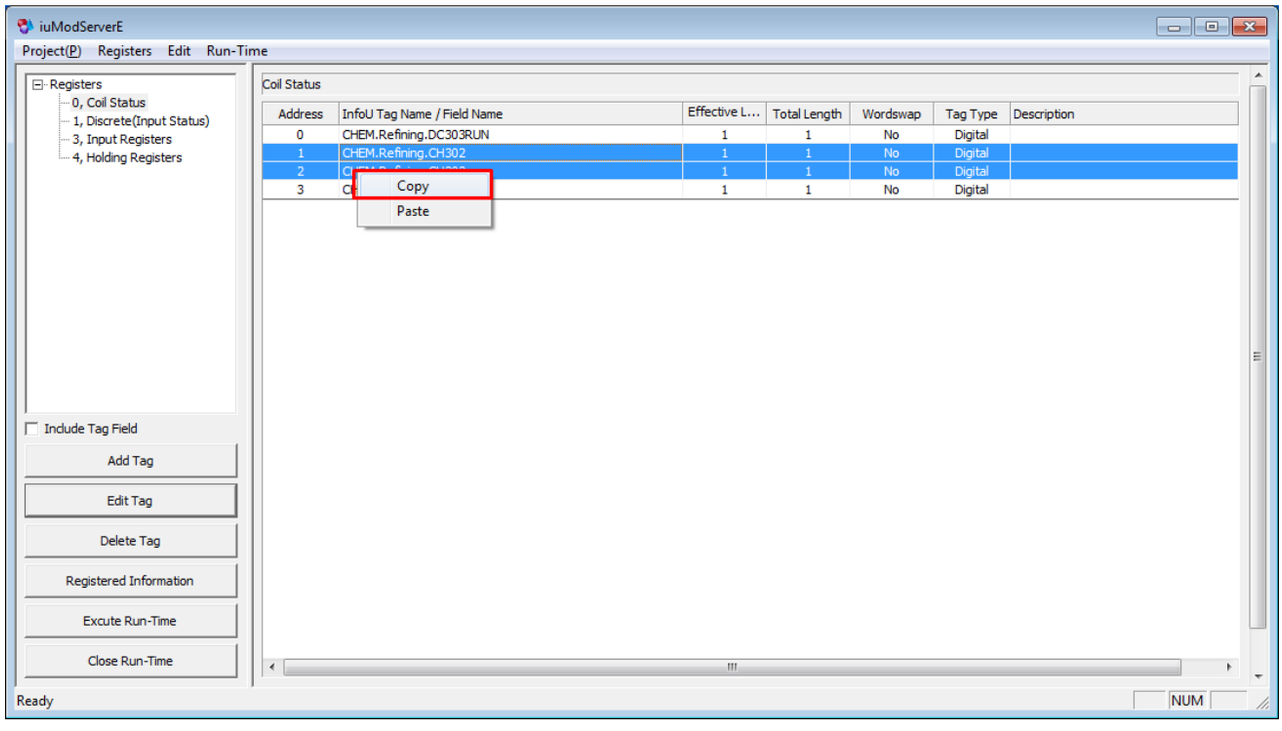
After selecting the tag that you want to edit from the registered tags, if you click the “Edit Tag” button on the left side, the menu where you can edit the information of the selected tag will be displayed on the screen. You can edit items of the effective length/Wordswap depending on the tag type.



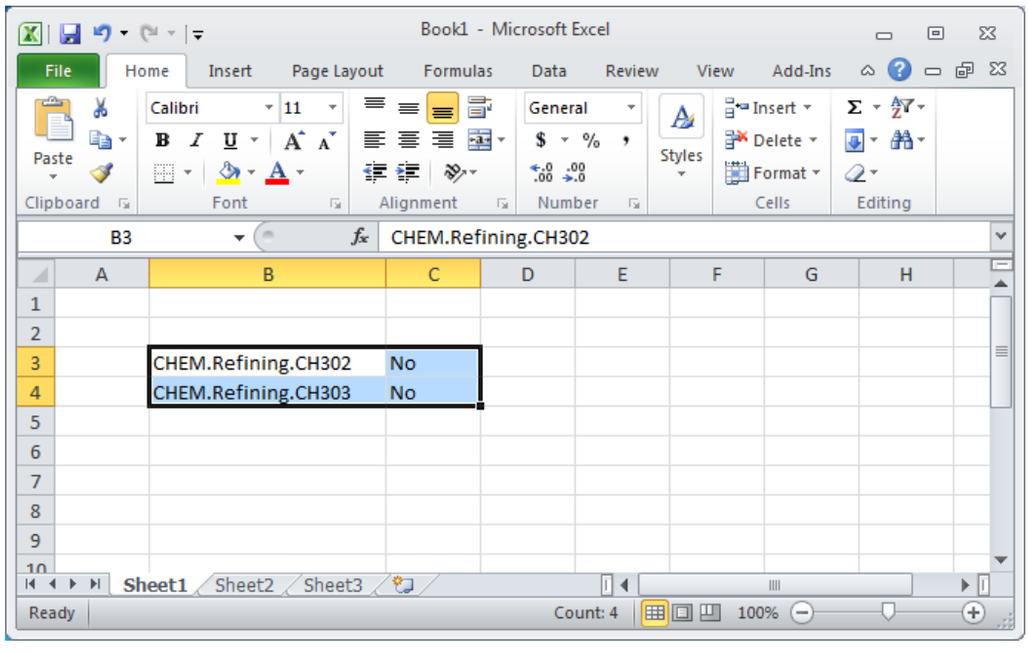
1.1.4 Copy / Paste Clipboard

(1) Copy to Clipboard

You can copy the tag registered in the editor to a clipboard. If you select the desired tag from the registered tags by dragging it and click with the right mouse, the below popup menu will show up. If you click [Copy] in the popup menu, the information of the selected tag will be copied to the clipboard.



Through the editor, you can copy the details saved in the clipboard and paste it to Excel.

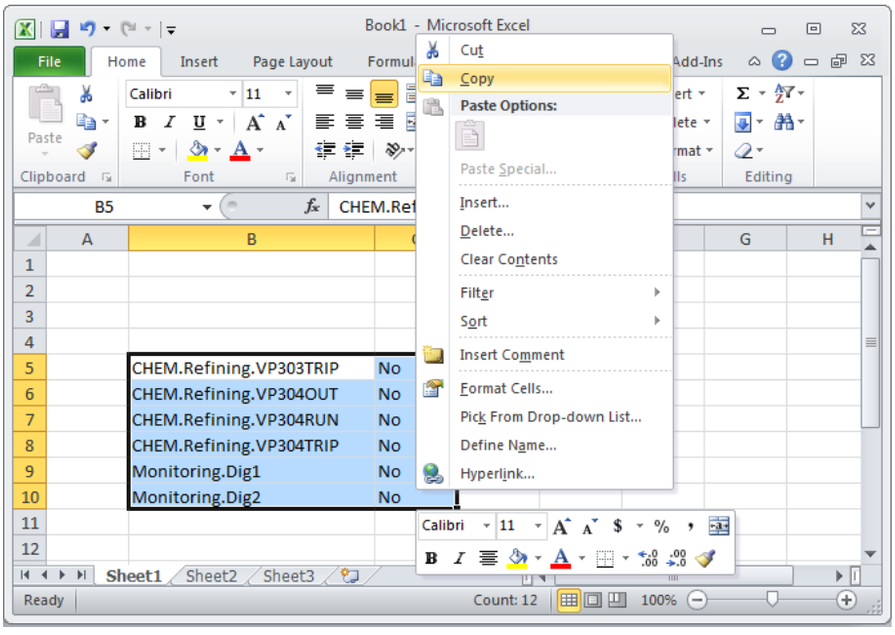


Notice

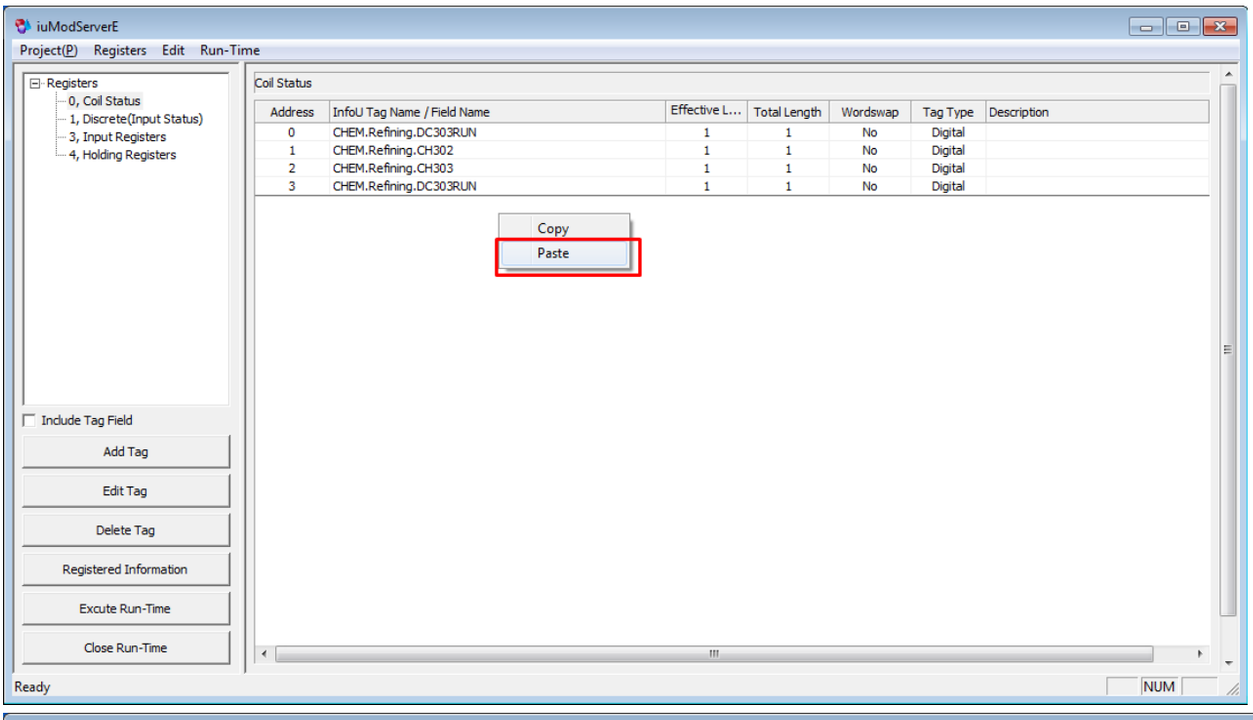
☞ The tag information copied from a clipboard is the tag name and Wordswap information. When pasting them to Excel, the left column indicates the tag name and the right column indicates the Wordswap information.

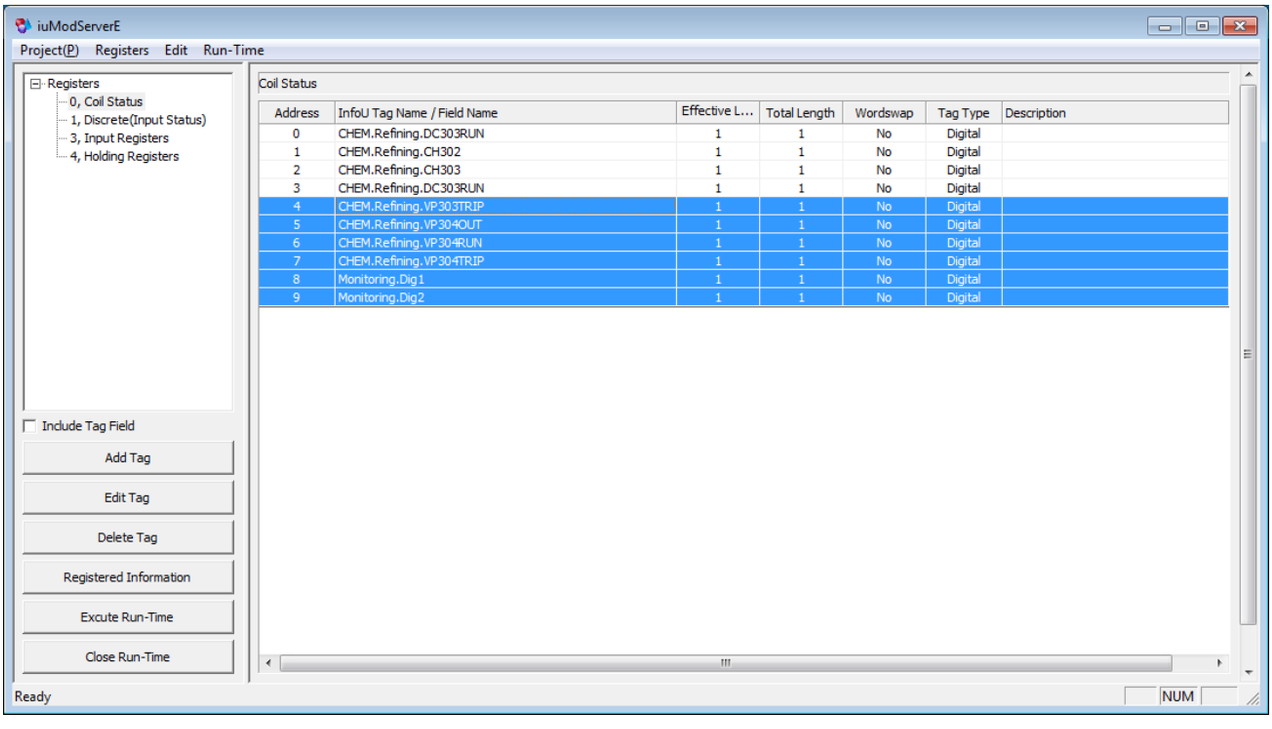
(2) Paste from Clipboard

You can edit the information of the tag to be registered in Excel and copy it to the clipboard.



After selecting the register for the tags treated in Excel, if you click with the right mouse button on the tag registration, the below popup menu will show up. If you click [Paste] in the popup menu, the tags processed in Excel will be added to the editor.





Notice

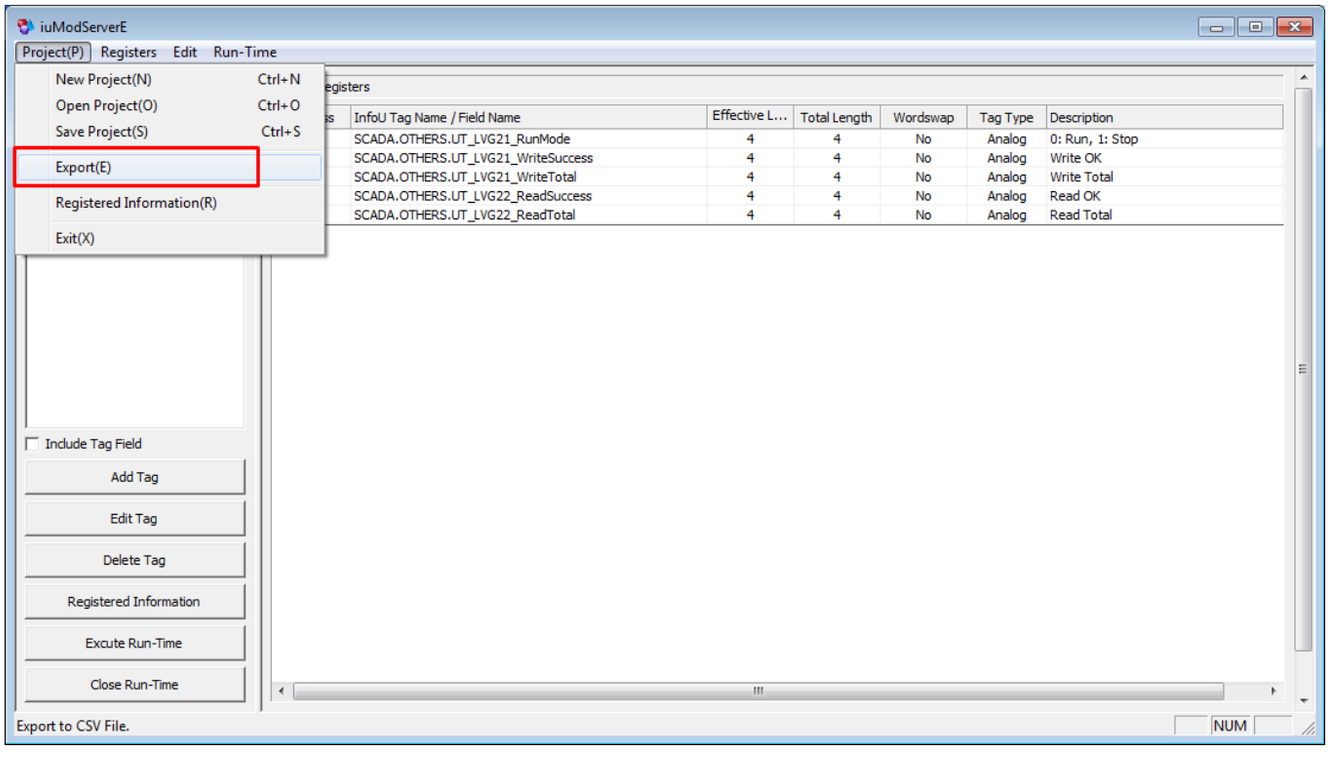
☞ When processing the tag information in Excel, the tag should be registered in the InfoU Project with the right spelling and style.

☞ Depending on the property of the register, you need to separate digital/analog data for Paste.

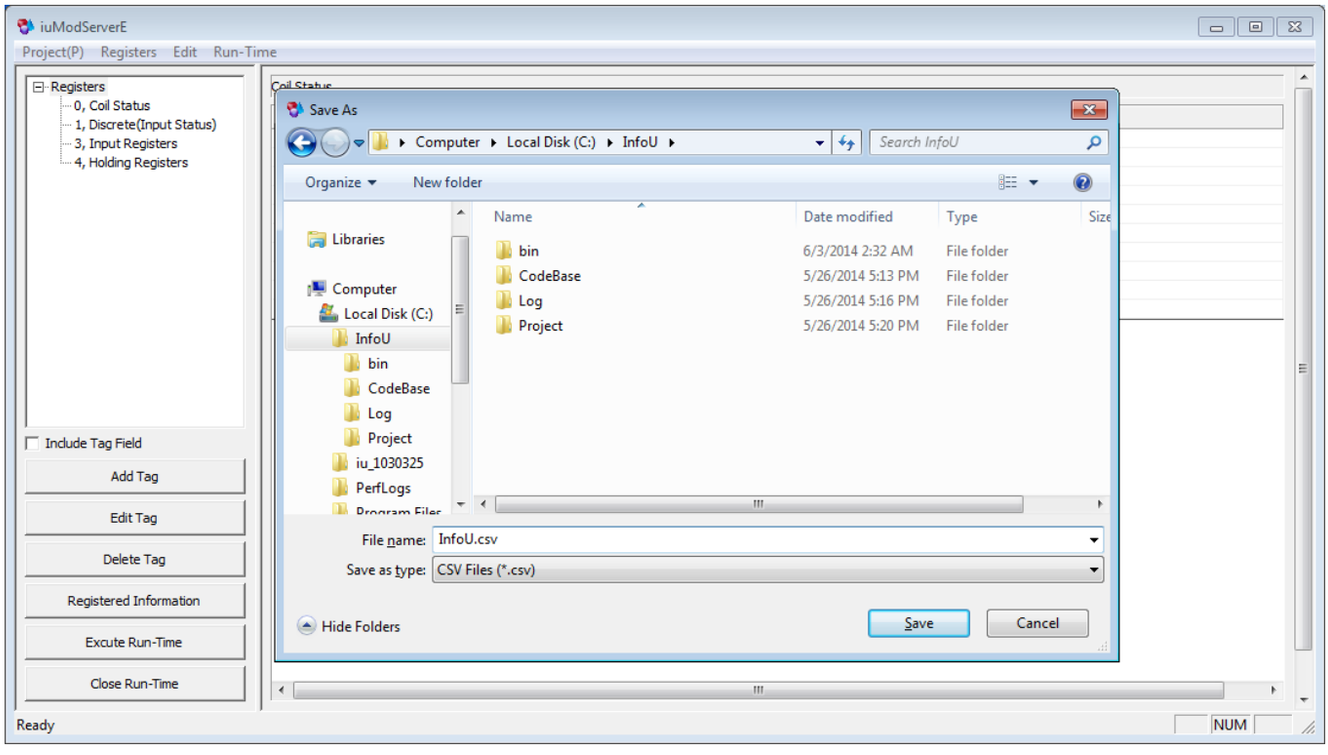
- For the Coil Status (0x00) and Discrete Inputs (0x01) Register, you can input the digital data only.
- For the Input Registers (0x03) and Hold Registers (0x04), you can input the analog data only.

1.1.5 Export to CSV file

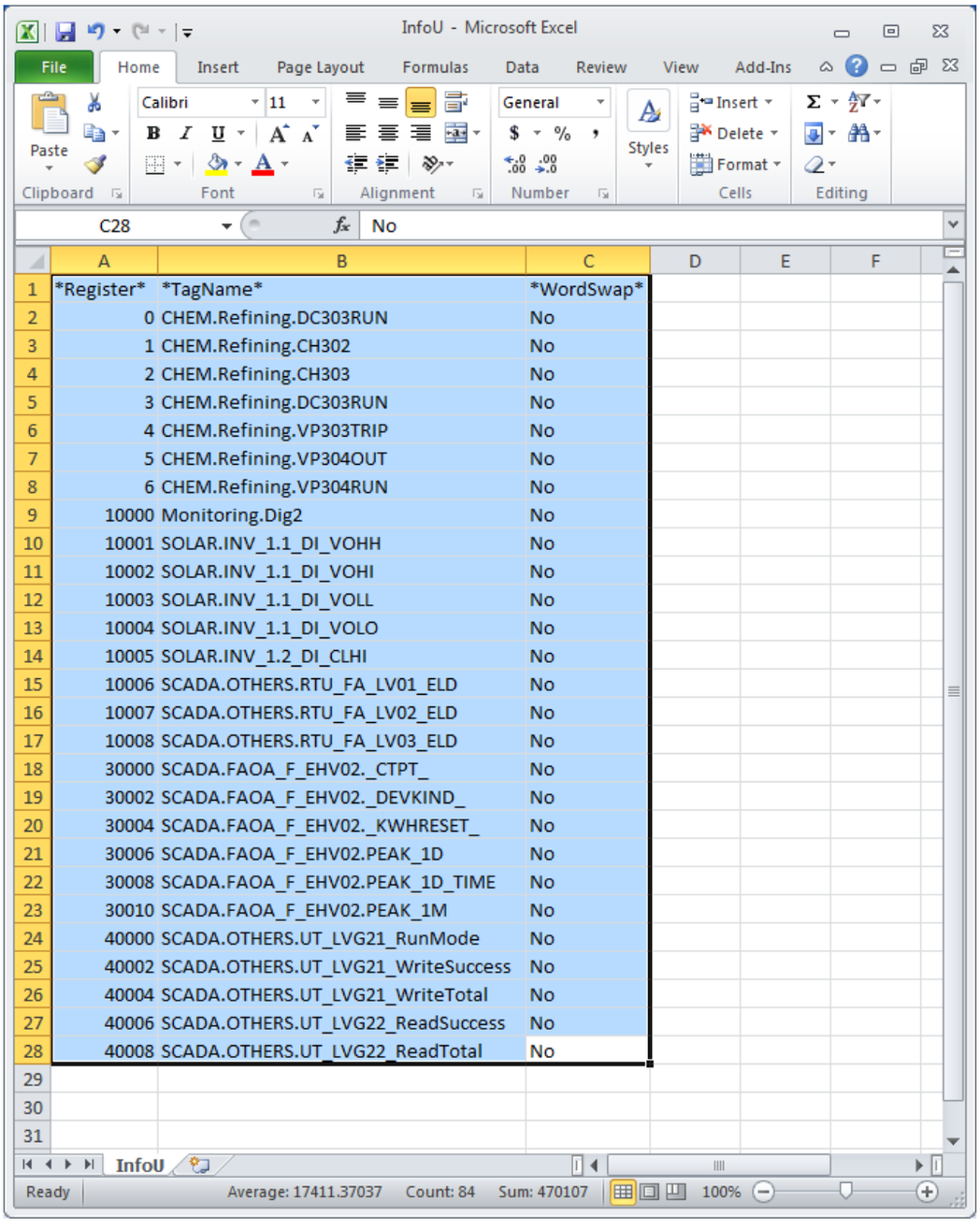
You can export the whole tags registered in the editor to the CSV file at once.



After selecting [Project] → [Export] in the upper menu, you can save the CSV file with the desired path and file name as below.



When you open the saved CSV file, you can see the information of the Register No., Tag name and Wordswap, is saved as shown below.



InfoU - Microsoft Excel

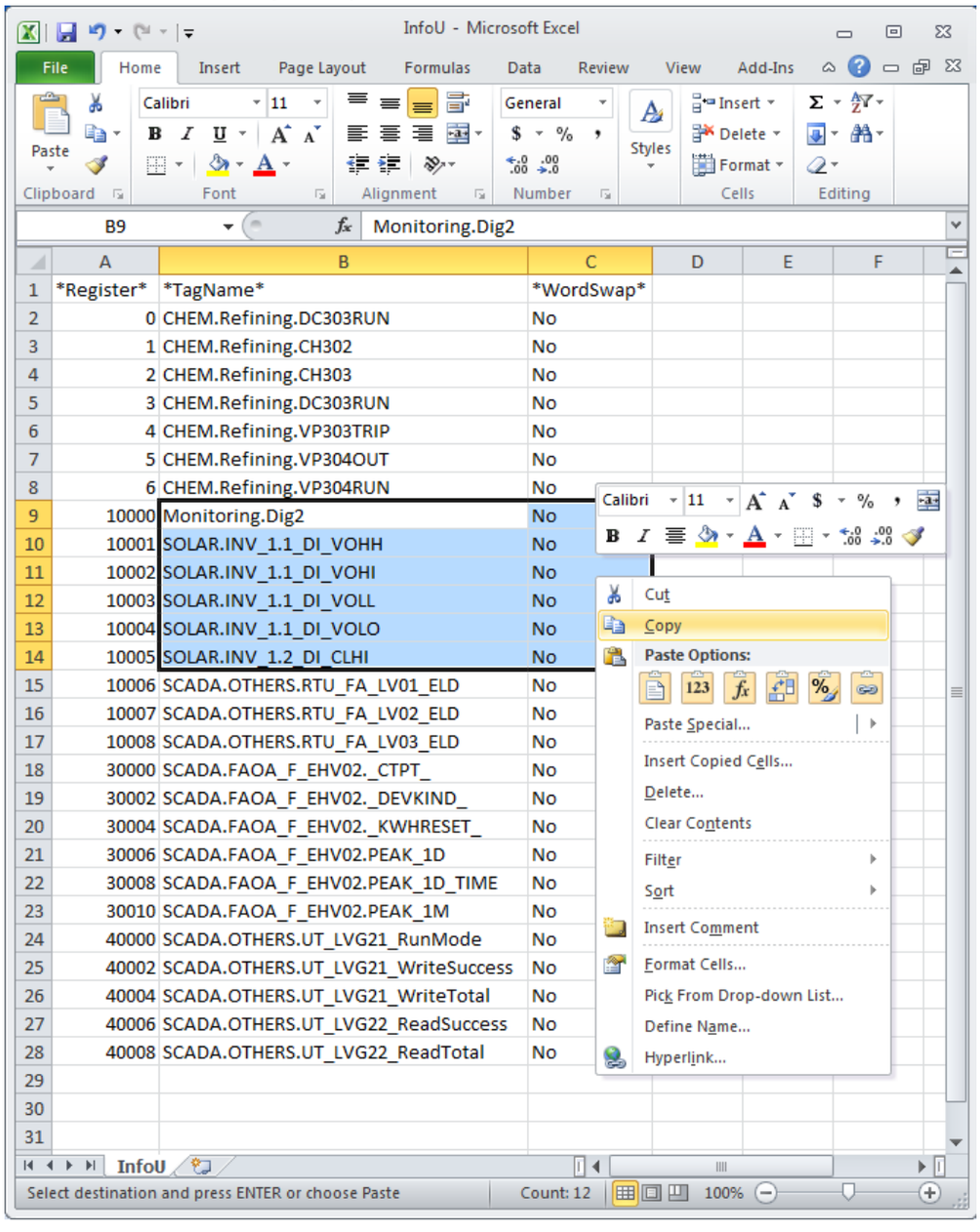
	A	B	C	D	E	F
1	*Register*	*TagName*	*WordSwap*			
2	0	CHEM.Refining.DC303RUN	No			
3	1	CHEM.Refining.CH302	No			
4	2	CHEM.Refining.CH303	No			
5	3	CHEM.Refining.DC303RUN	No			
6	4	CHEM.Refining.VP303TRIP	No			
7	5	CHEM.Refining.VP304OUT	No			
8	6	CHEM.Refining.VP304RUN	No			
9	10000	Monitoring.Dig2	No			
10	10001	SOLAR.INV_1.1_DI_VOHH	No			
11	10002	SOLAR.INV_1.1_DI_VOHI	No			
12	10003	SOLAR.INV_1.1_DI_VOLL	No			
13	10004	SOLAR.INV_1.1_DI_VOLO	No			
14	10005	SOLAR.INV_1.2_DI_CLHI	No			
15	10006	SCADA.OTHERS.RTU_FA_LV01_ELD	No			
16	10007	SCADA.OTHERS.RTU_FA_LV02_ELD	No			
17	10008	SCADA.OTHERS.RTU_FA_LV03_ELD	No			
18	30000	SCADA.FAOA_F_EHV02._CTPT_	No			
19	30002	SCADA.FAOA_F_EHV02._DEVKIND_	No			
20	30004	SCADA.FAOA_F_EHV02._KWHRESET_	No			
21	30006	SCADA.FAOA_F_EHV02.PEAK_1D	No			
22	30008	SCADA.FAOA_F_EHV02.PEAK_1D_TIME	No			
23	30010	SCADA.FAOA_F_EHV02.PEAK_1M	No			
24	40000	SCADA.OTHERS.UT_LVG21_RunMode	No			
25	40002	SCADA.OTHERS.UT_LVG21_WriteSuccess	No			
26	40004	SCADA.OTHERS.UT_LVG21_WriteTotal	No			
27	40006	SCADA.OTHERS.UT_LVG22_ReadSuccess	No			
28	40008	SCADA.OTHERS.UT_LVG22_ReadTotal	No			
29						
30						
31						

Notice

The start address (Register No.) registered by Register is as blow.

- ☞ Coil Status(0x00)→ 0, 1, 2, …
- ☞ Discrete Inputs(0x01)→ 10000, 10001, 10002, …
- ☞ Input Registers(0x03)→ 30000, 30002, 30004, …
- ☞ Hold Registers(0x04)→ 40000, 40002, 40004, …

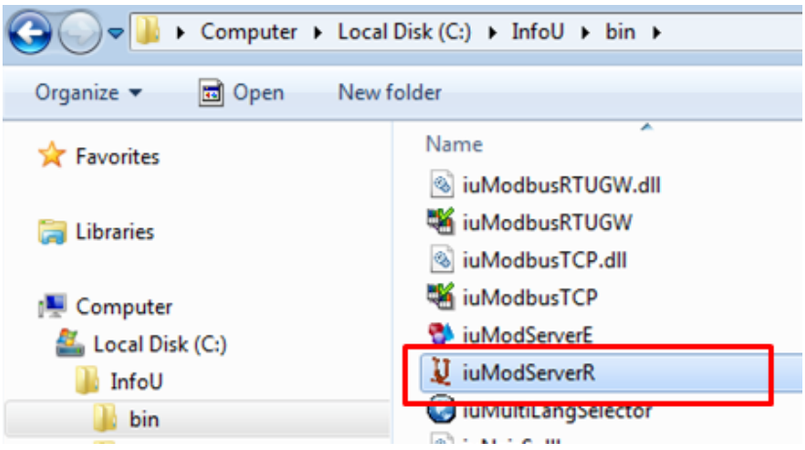
You can copy the second and thirtd row of the desired tags from the CSV file generated by ‘Export’ and paste them to the editor through a clipboard.



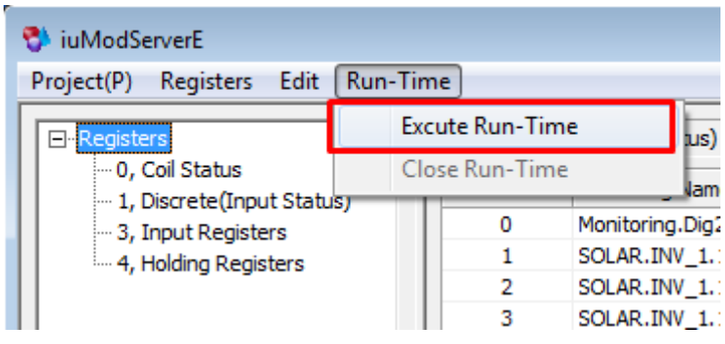
1.2 Modbus Server Runtime

1.2.1 Execute Runtime

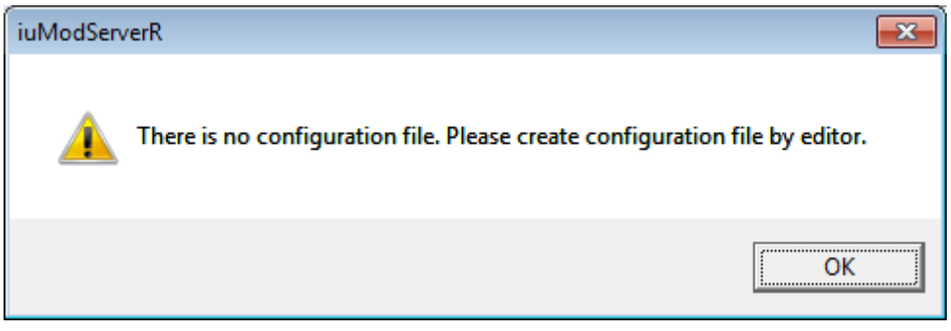
You can run the iuModServerR.exe program in the InfoU\bin folder.



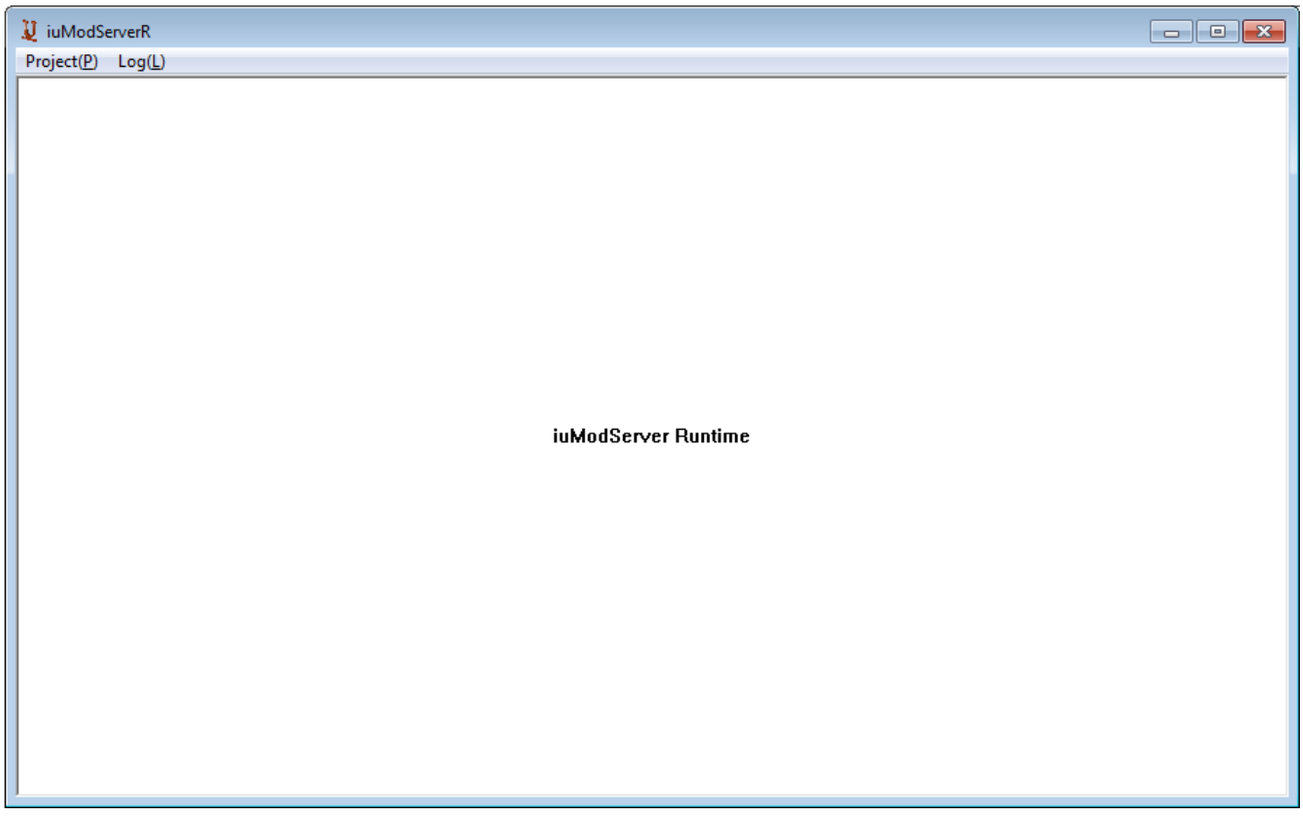
Or you can click [Run-time] → [Execute Run-Time] in the upper menu of the editor.



If there is no information edited by the editor, the runtime will not work with the below warning message.



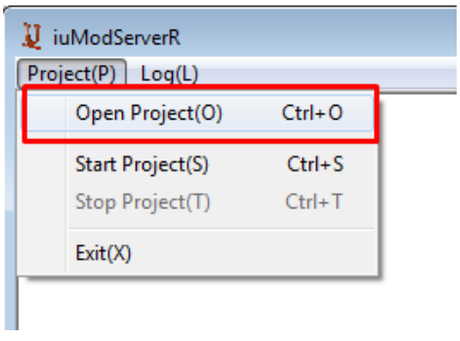
If the normal file exists, the file is read and the below runtime screen will be executed. Then, it will be switched over the Modbus standby mode.



1.2.2 Runtime Menu

(1) Open Project

If you click [Project] → [Open Project] in the upper menu of the runtime window, you can reread the configuration file treated by the editor

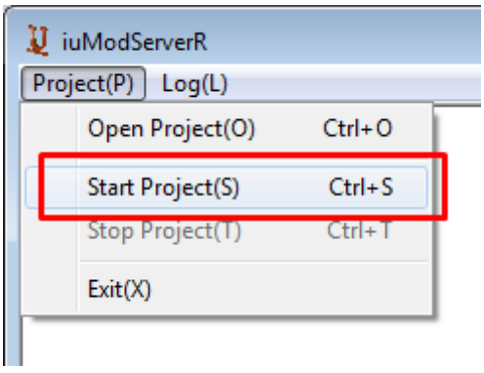


Notice

☞ If the file is edited by the editor while the runtime is executed, it is not applied immediately so you need to reread it.

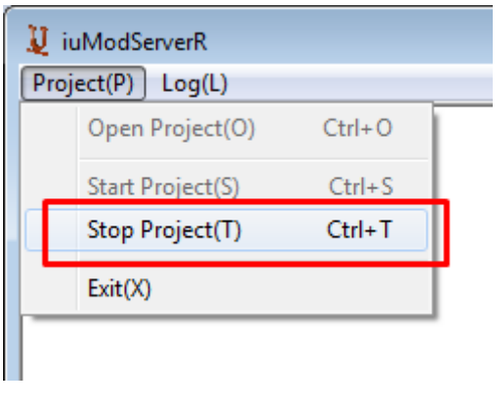
(2) Start Project

Click [Project] → [Start Project] in the upper menu of the runtime window and scan the InfoU data periodically based on the cycle (Refresh Interval) set by the editor. Then, the information is transmitted at the request of the Modbus Client.



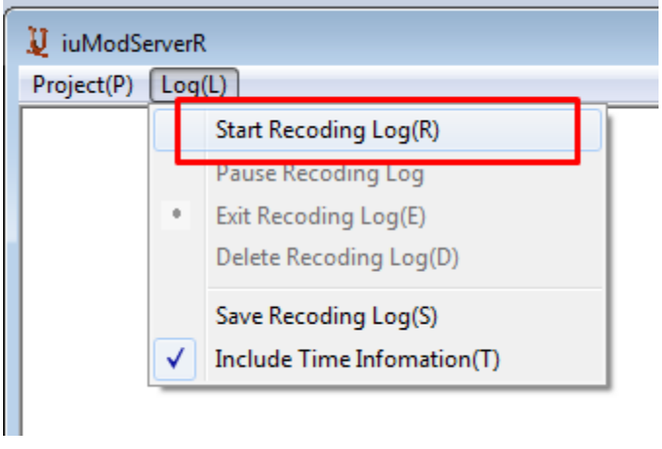
(3) Stop Project

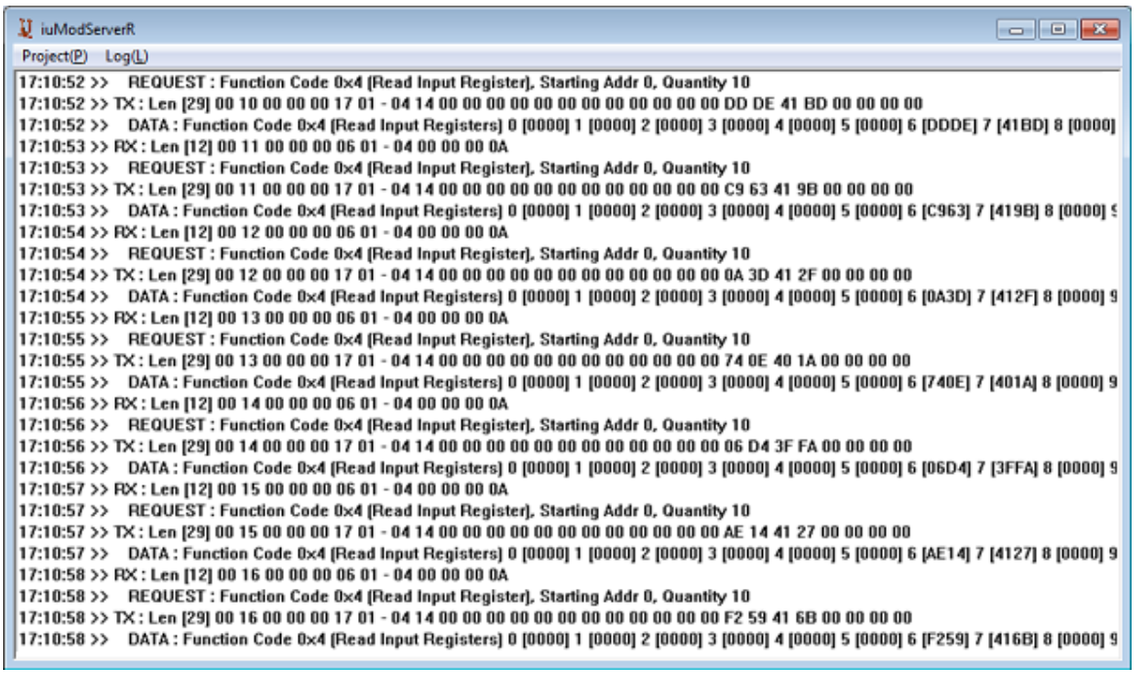
Click [Project] → [Stop Project] in the upper menu of the runtime window and exit the provision of data for the InfoU.



(4) Recoding Log

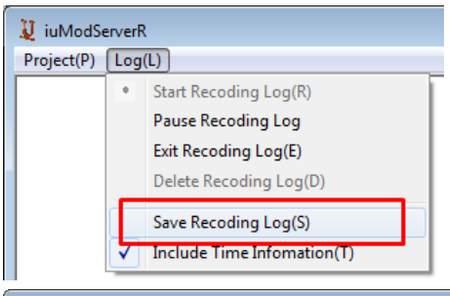
You can check the communication packet by clicking [Log] → [Start Recoding Log] in the upper menu of the runtime window.

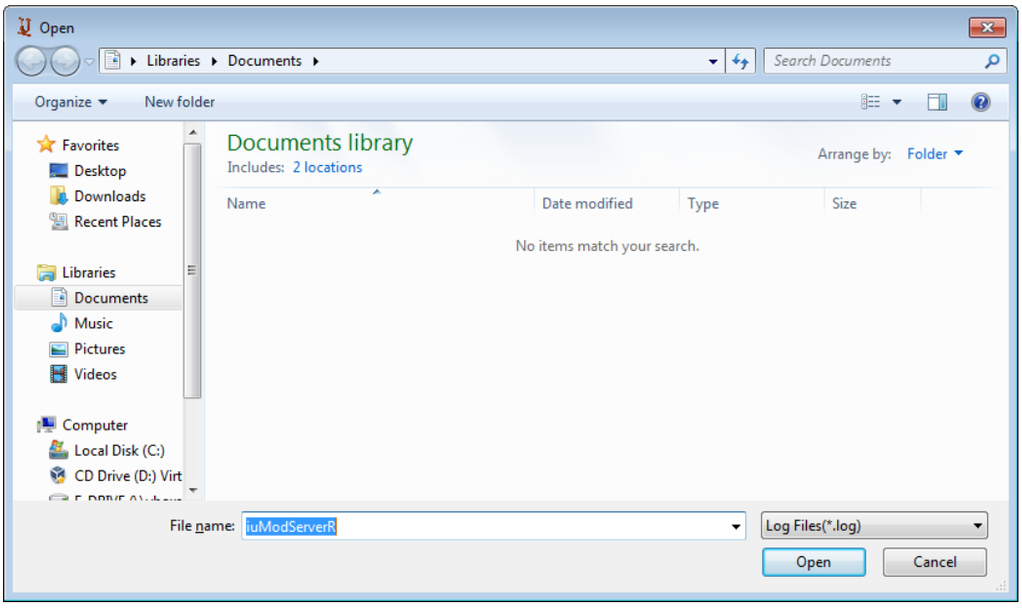




(5) Save Recoding Log

If you click [Log] → [Save Recoding Log] in the upper menu of the runtime window, you can save the log as a file with the desired path and name.





Appendix 2 LSIS InfoU OPC Server

2.1 Introduction

OPC is intended to standardize communication with various industrial devices under the Microsoft window operation system. This operates based on COM/DCOM functions of the window operation system, consisting of two kinds of software- OPC Client software such as InfoU and OPC Server software. This session explains how to use the Company's OPC Server that can exchange data through OPC (OLE for Process Control)- method communication. LS InfoU OPC Server presents a Configuration Program to select tags acquired from LSIS OPC DA Server, which will provide data during the runtime, and OPC DA Server. The Configuration Program contains monitoring function for local and remote diagnosis with its own InfoU OPC Server.

2.2 Install Program

2.2.1 Installation

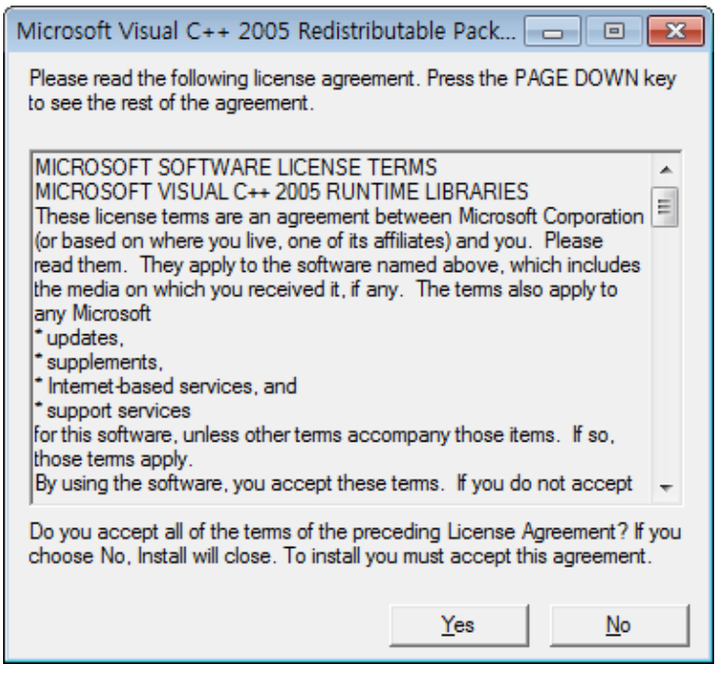
The program is installed according to the following sequence.

(1) Install .NET Framework 2.0

Run "dotnetfx.exe".

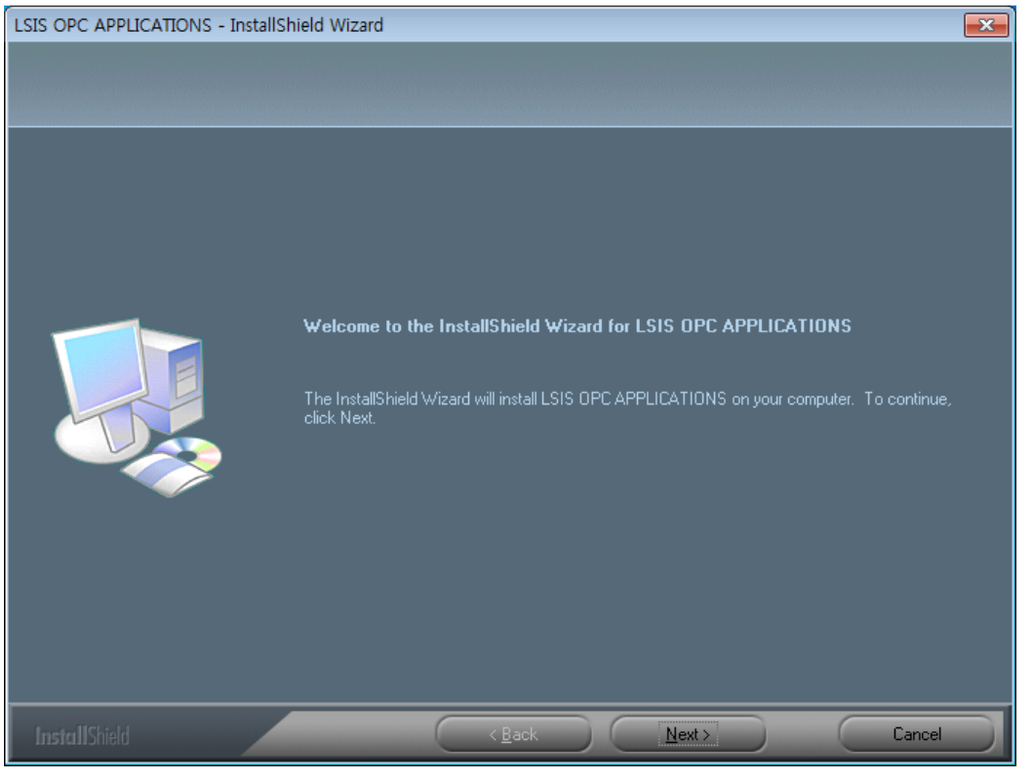
(2) Install Microsoft Visual C++ 2005 Run Time library"

Run "vcredist_x86.exe"



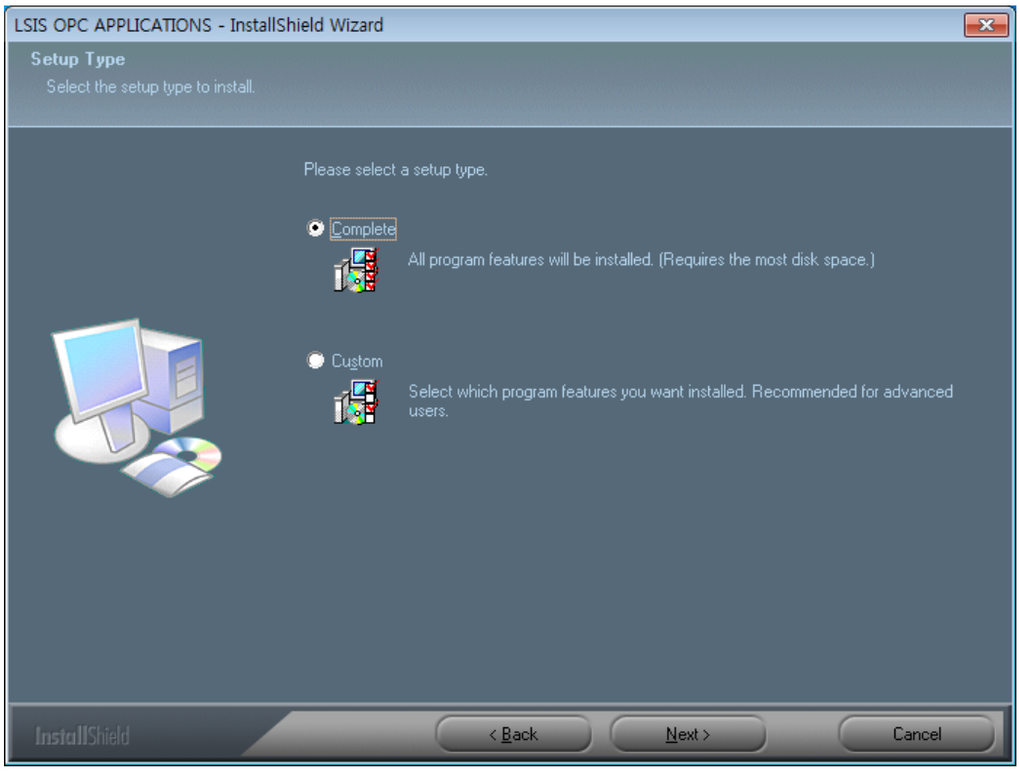
(3) Install LS InfoU OPC DA Server"

1) Run "LSISOPC Server For InfoU_V_1_4.exe"
2) Select "Yes" to "Accept" all of the terms of the License Agreement for using "Microsoft Visual C++ 2005 Run Time library".

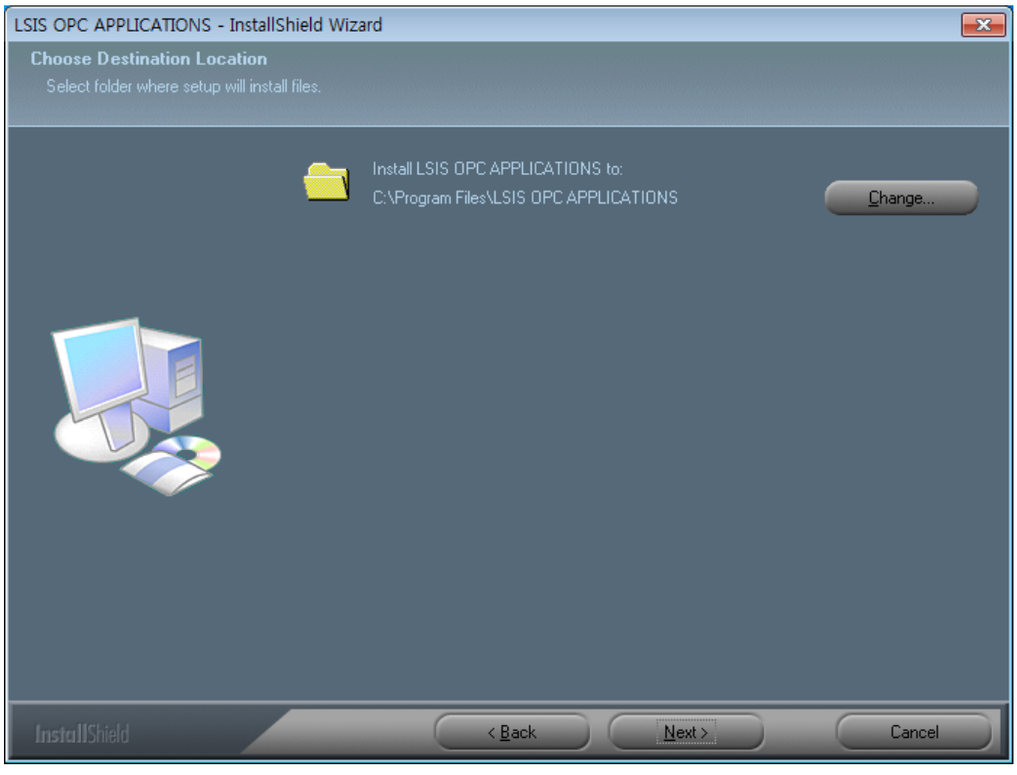


3) Select the "Setup" type

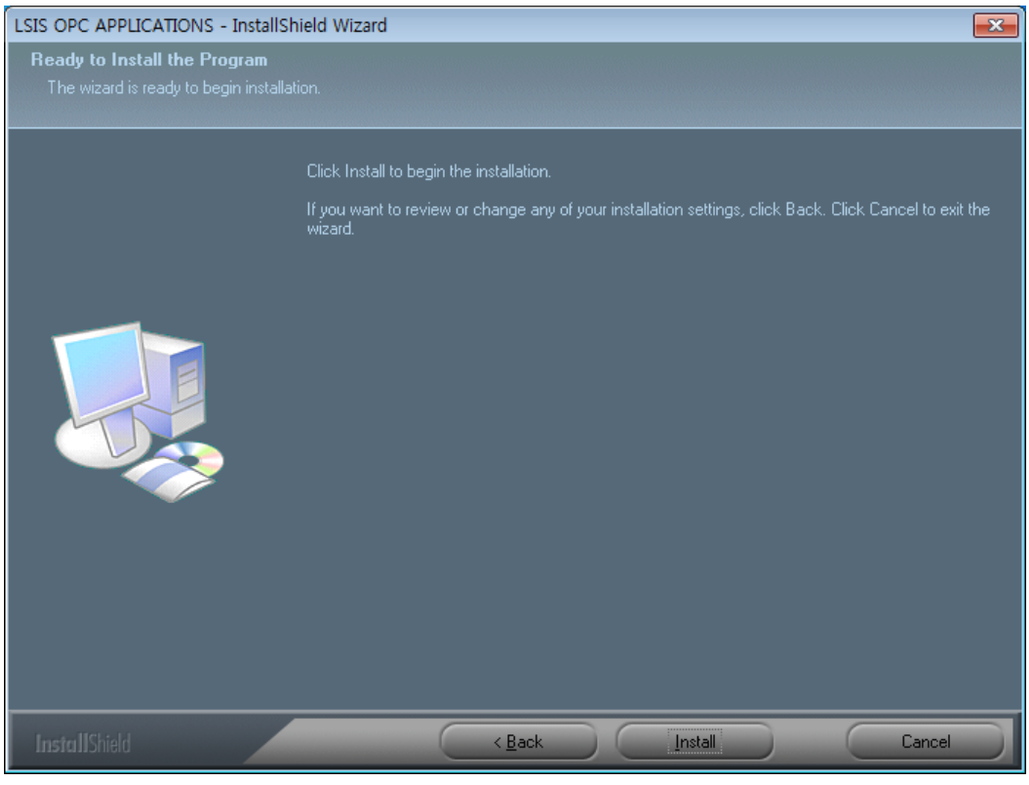
- Complete: All program features will be installed.
- Custom: Only the selected features by the user will be installed.

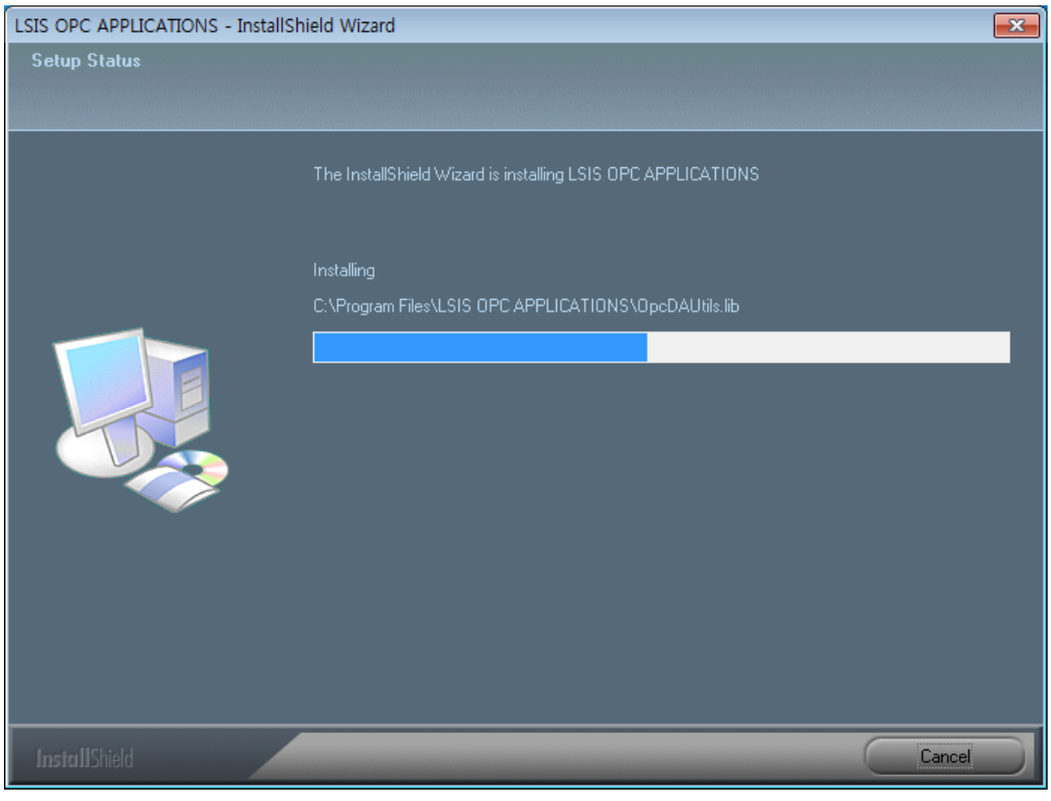


4) Choose a destination location. When changing the default folder, press “Change” button to select a folder where setup files will be installed.

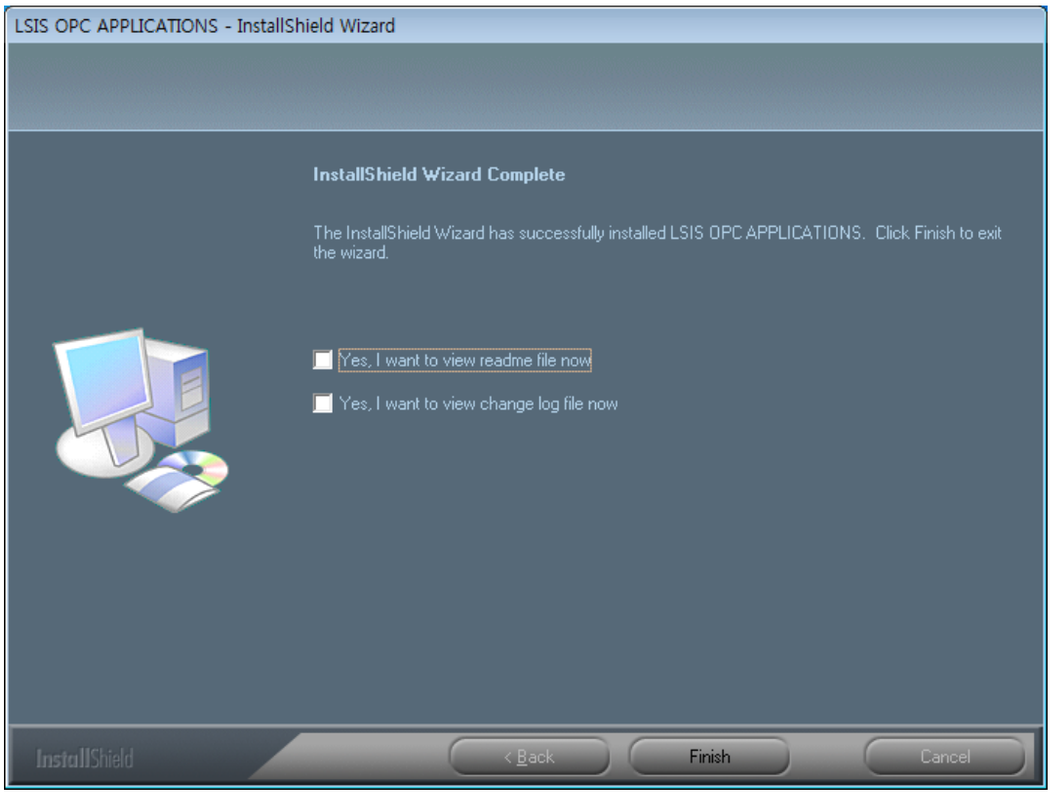


5) Press “Install” button to start installing.





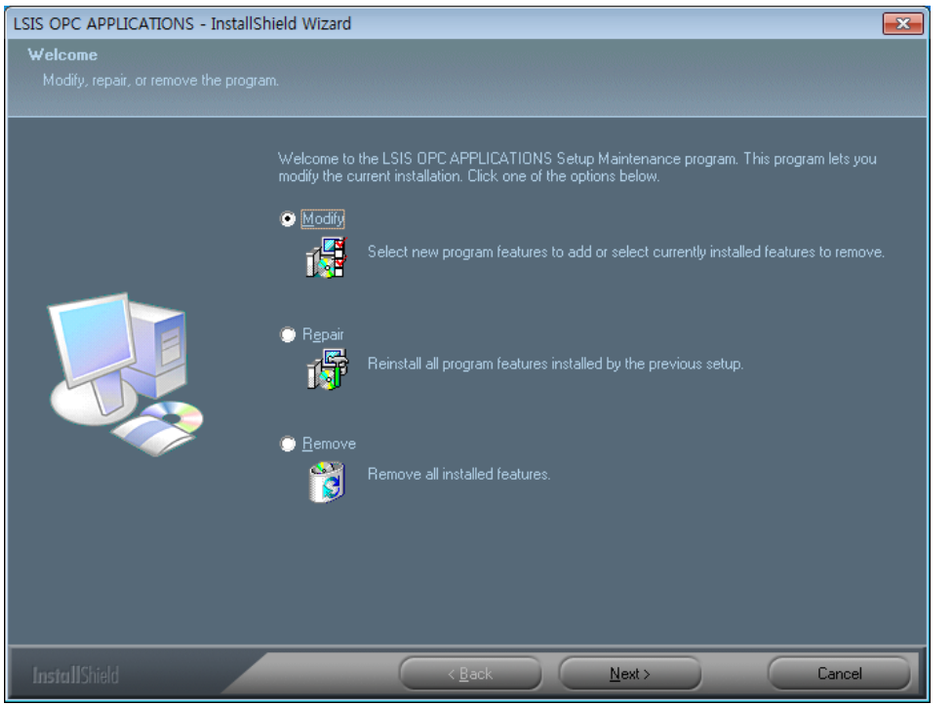
6) Press“Finish” button and exit installation.



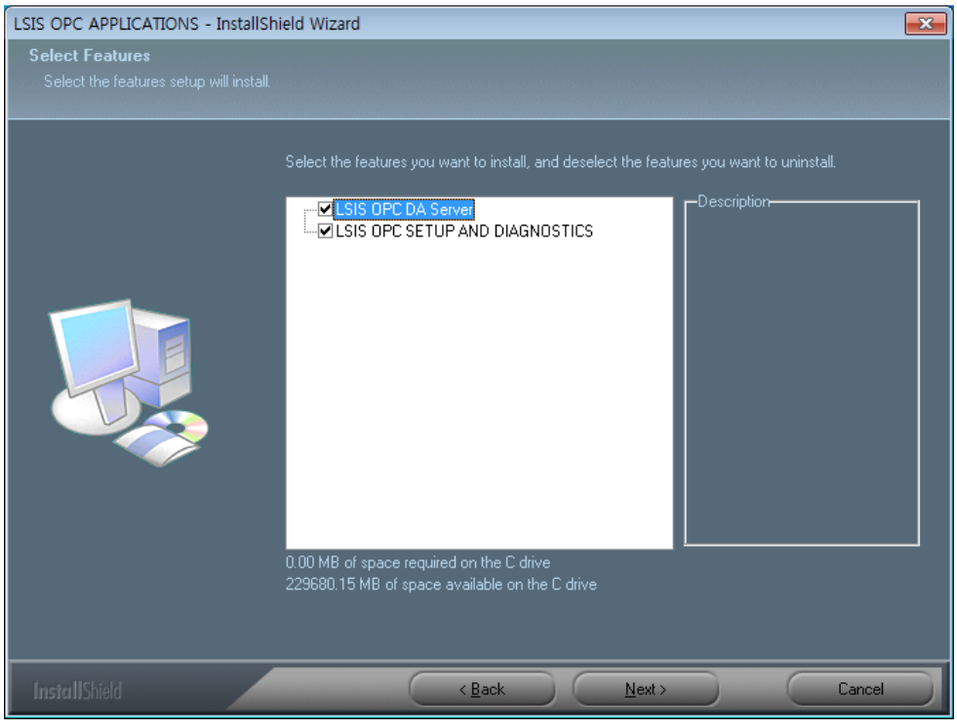
2.2.2 Modify/Repair/Remove Program

(1) Modify

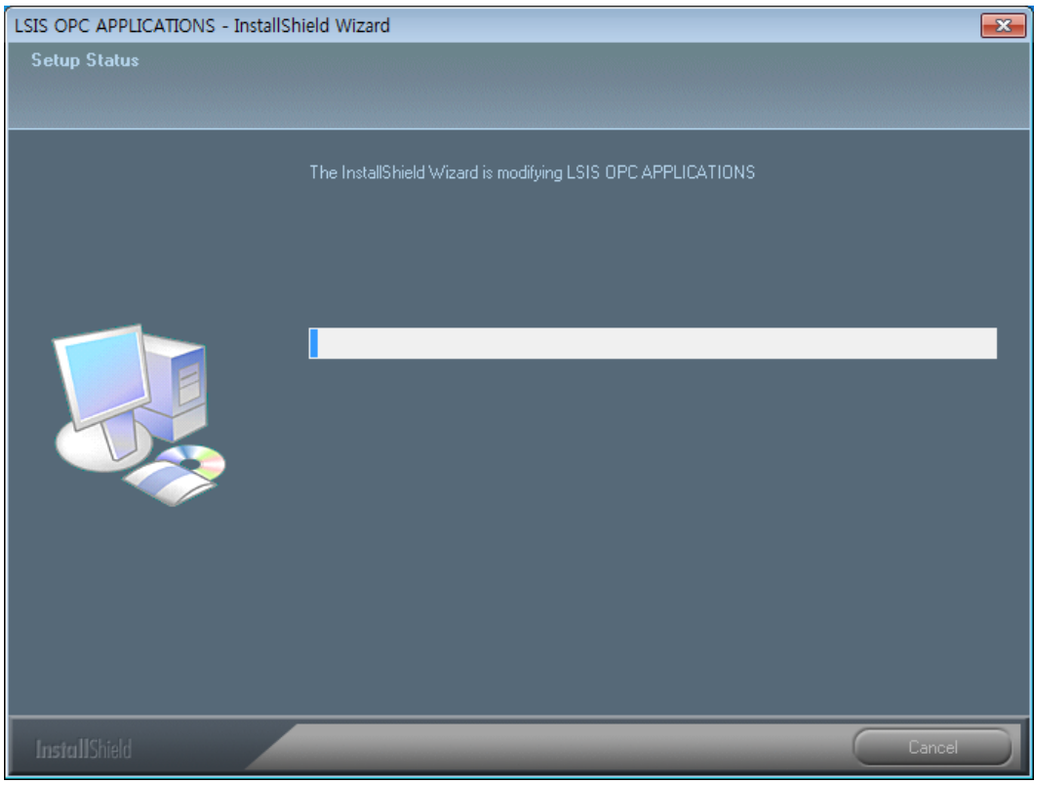
1) Run “LSISOPC Server For InfoU_V_1_4.exe”



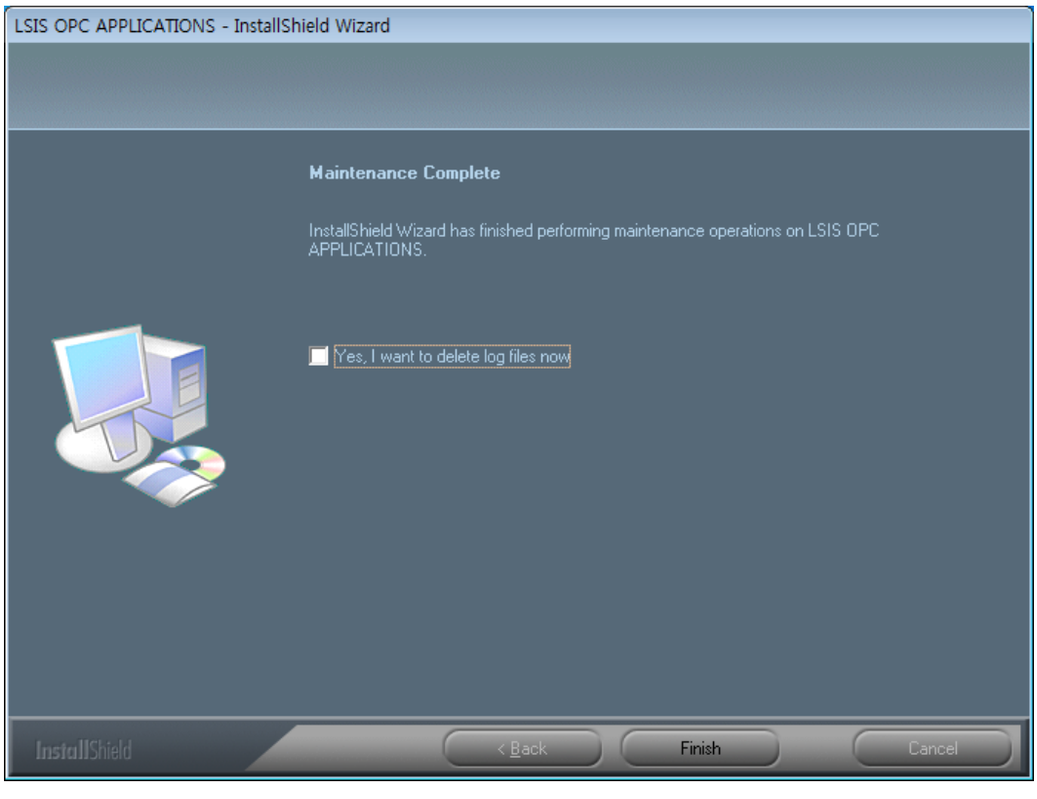
2) Decide whether to install or remove each program. If the check box is selected, the relevant program is installed or it not, it is removed.



3) Click on“Next”to run and complete modifying.



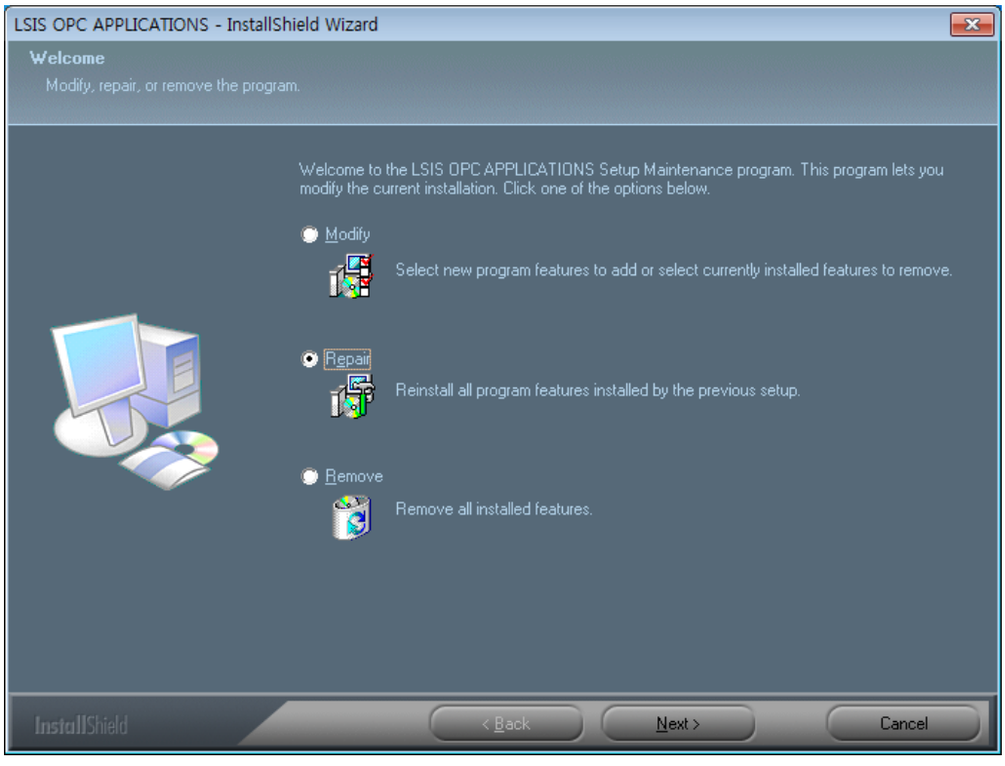
4) Press “Finish” button and complete the installation.



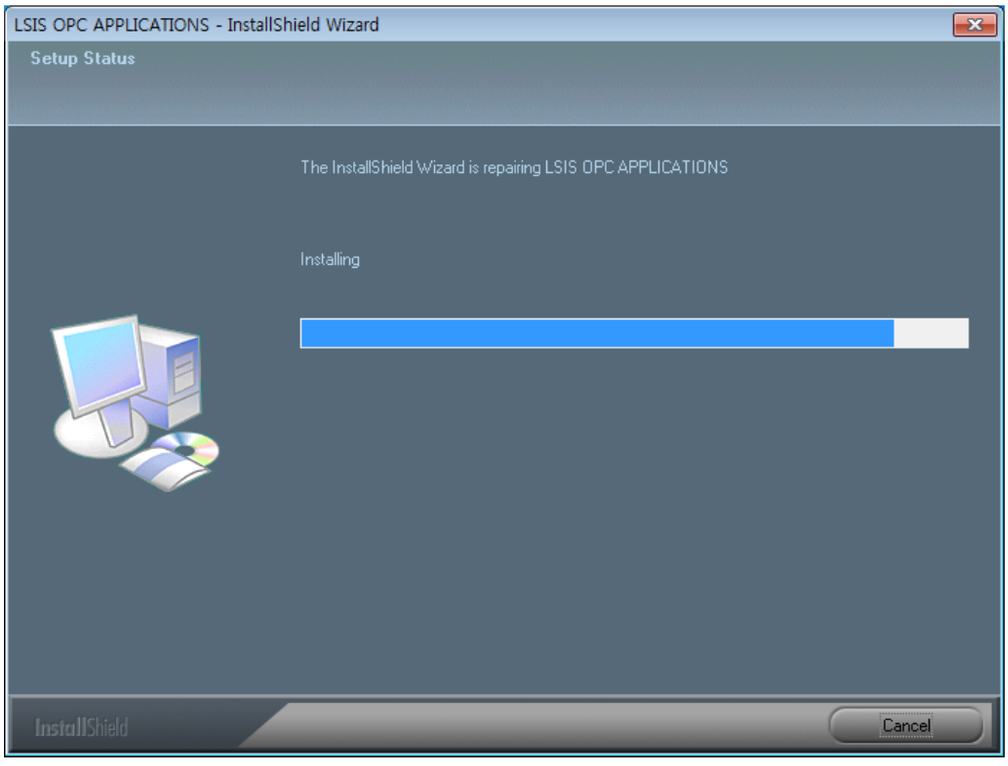
(2) Repair

1) Run “LSISOPC Server For InfoU_V_1_4.exe” file.

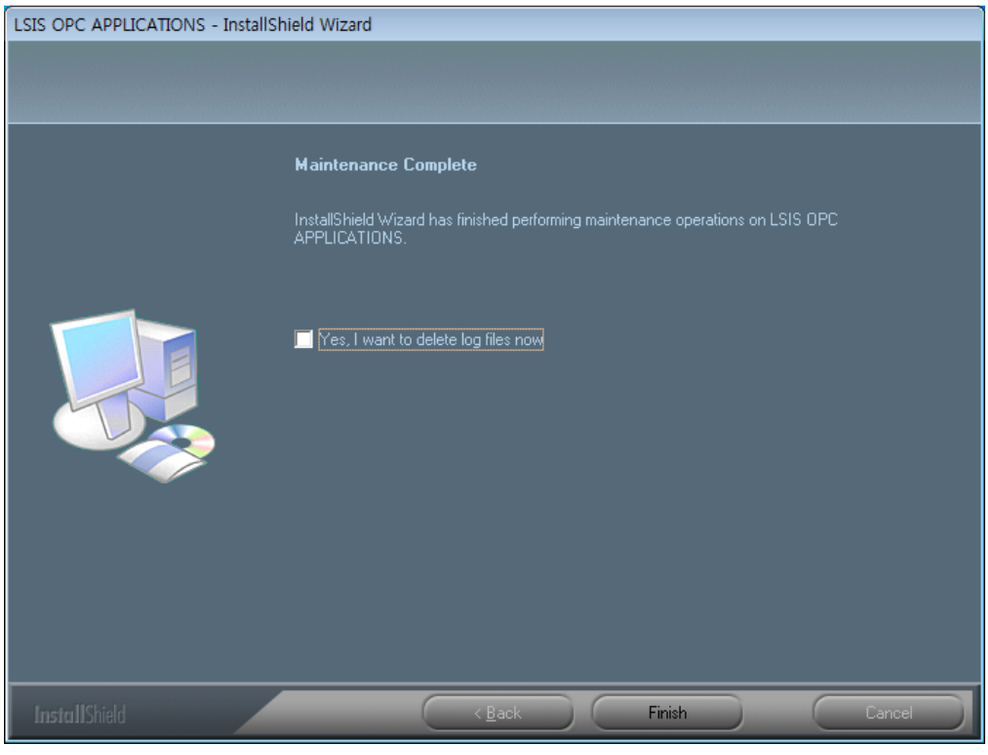
2) To reinstall all program features installed by the previous setup, select “Repair.”



3) Click on“Next”to run and complete repairing.



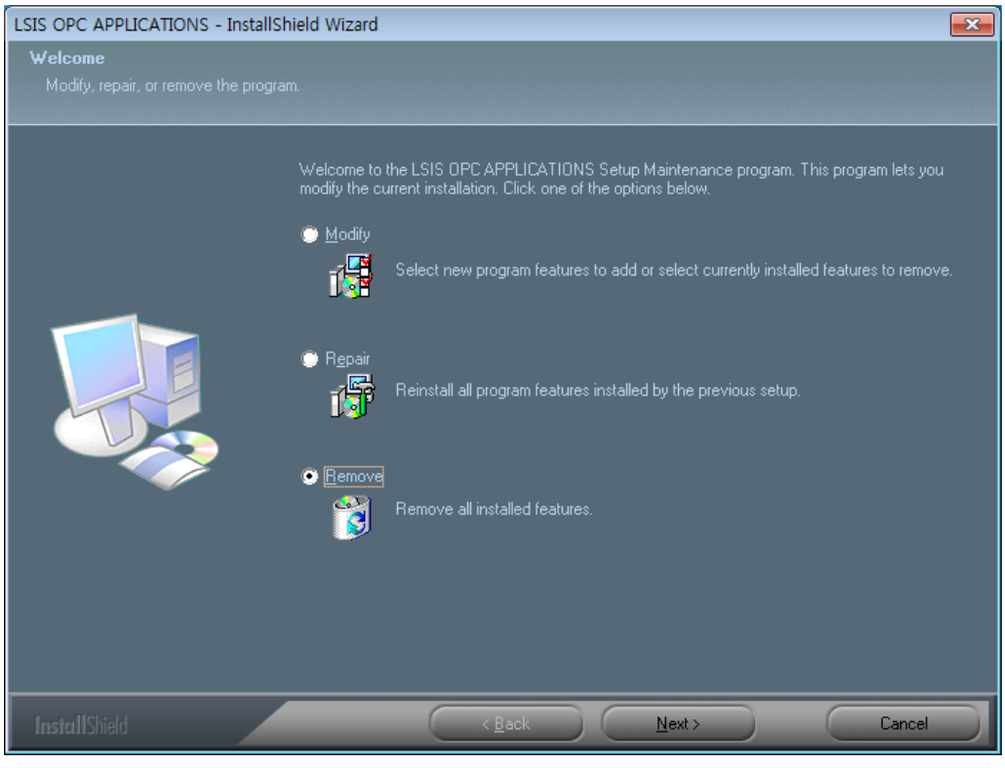
4) Press “Finish” button and complete the installation.



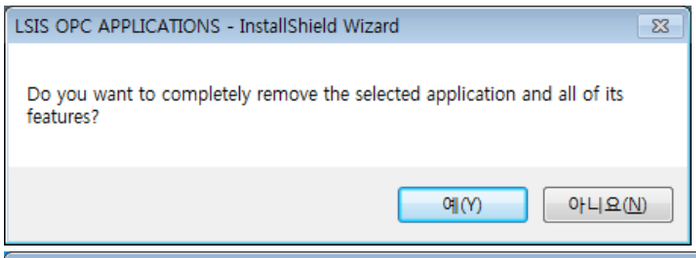
(3) Remove

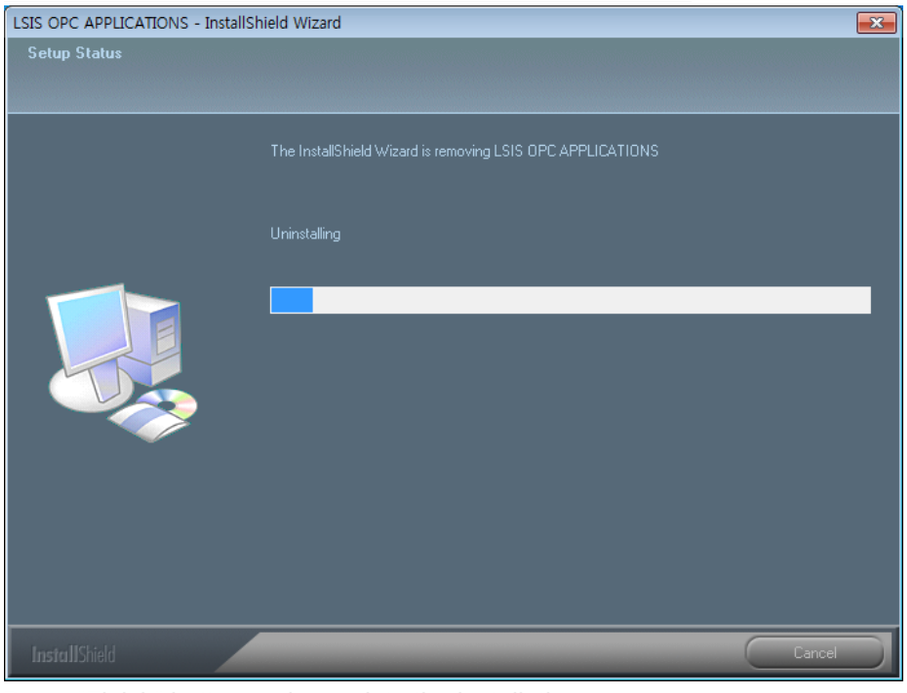
1) Run “LSISOPC Server For InfoU_V_1_4.exe”file.

2) Select “Remove”to remove all OPC programs related to installation.

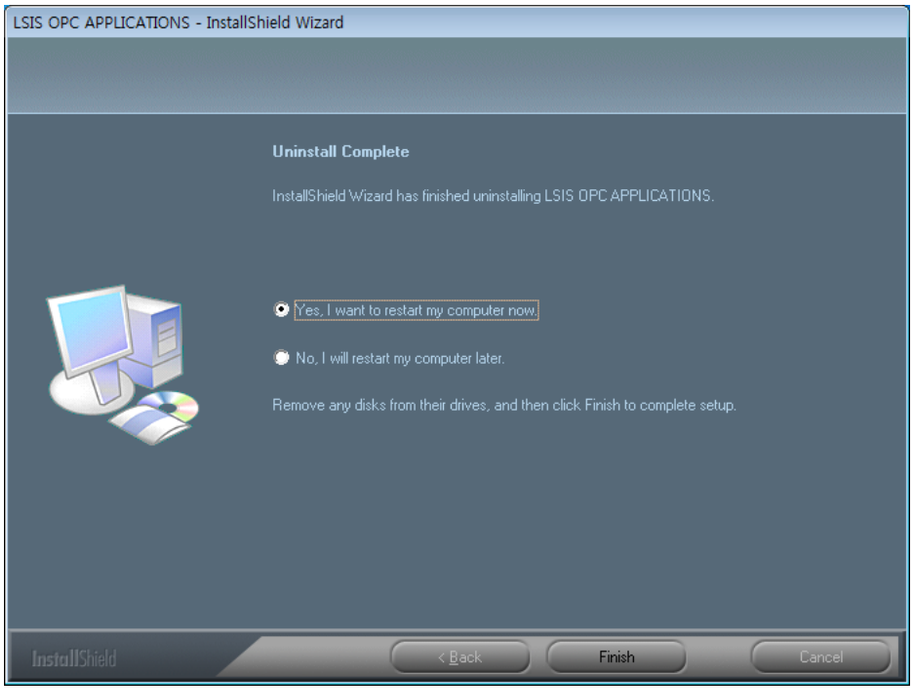


3) To remove, click on “Yes” button.



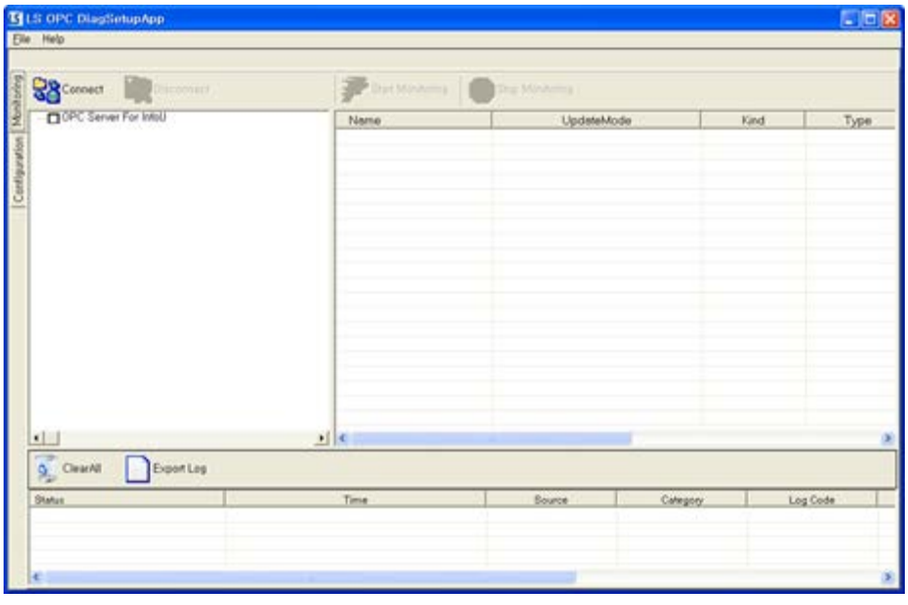


4) Press “Finish” button and complete the installation.



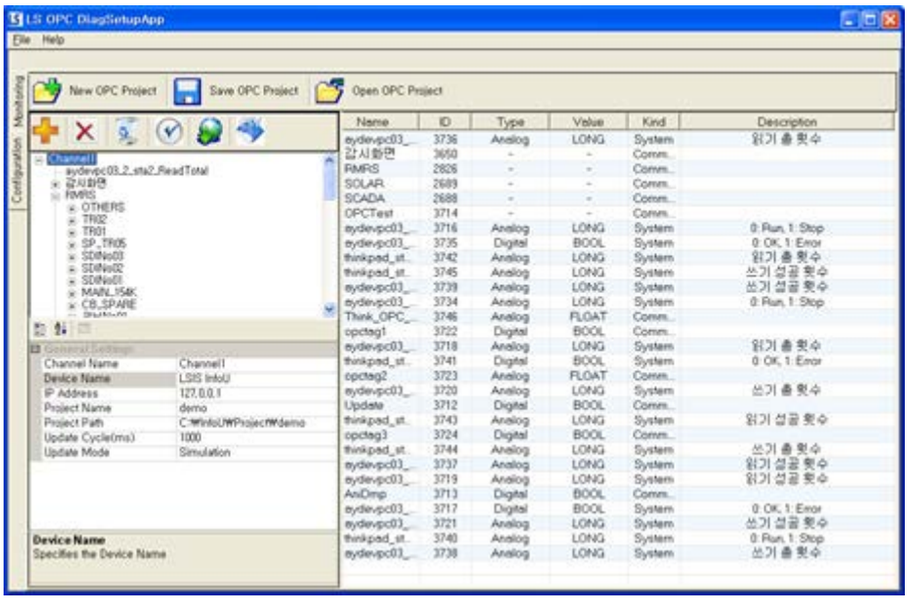
2.3 Configuration of Default Screen

Select [Start] → [All Programs] → [LSIS OPC Server APPLICATION] → [LSIS OPC SETUP AND DIAGNOSTICS] on the operation display line.

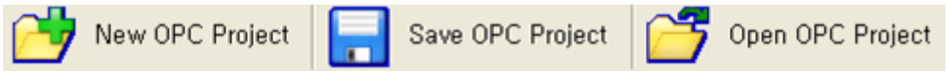


The program above has OPC DA Server Configuration and OPC DA Server Monitoring features. Among them, this shows a monitoring screen during the execution. If “Configuration” tap is selected from the program, it becomes available to select a tag list to be provided by the server as a configuration screen for OPC DA Server.

2.3.1 Configuration



(1) Toolbar (OPC Project)



1) New OPC Project: Contain information on new tags to be acquired in the runtime and create a new OPC project.
2) Save OPC Project: Save a new or modified OPC project.
3) Open OPC Project: Load an existing OPC project to be acquired by OPC DA server in the runtime.

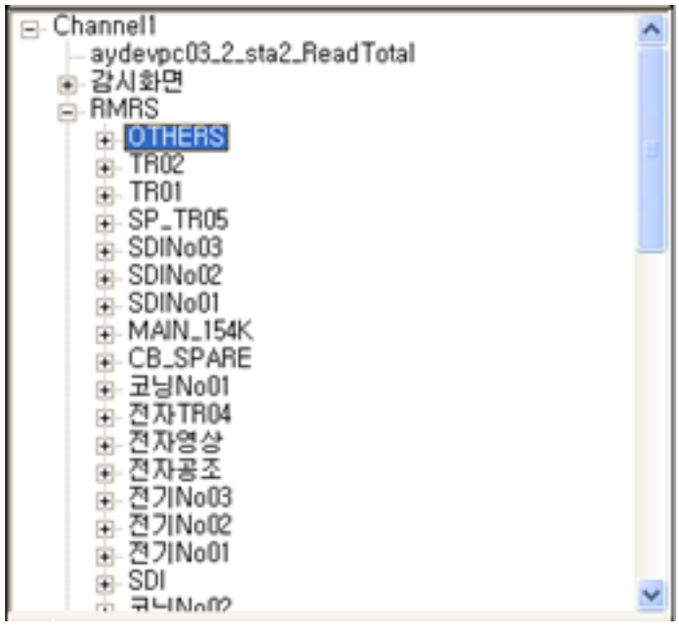
(2) Toolbar (Channel)



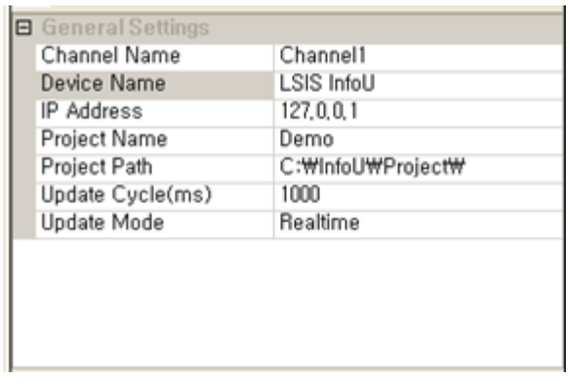
1) Add a channel, that is, set up one infoU project to be executed on other computer for each channel.
2) Delete the selected channel.
3) Delete all of the channels registered with OPC project.
4) Check the selected channel for its communication status.
5) Load tag information on the selected channel from InfoU project.
6) Load the created CSV from InfoUD and set up tags.

(3) Channel Window

View the tag DB loaded from the selected channel's InfoU project in a tree view.

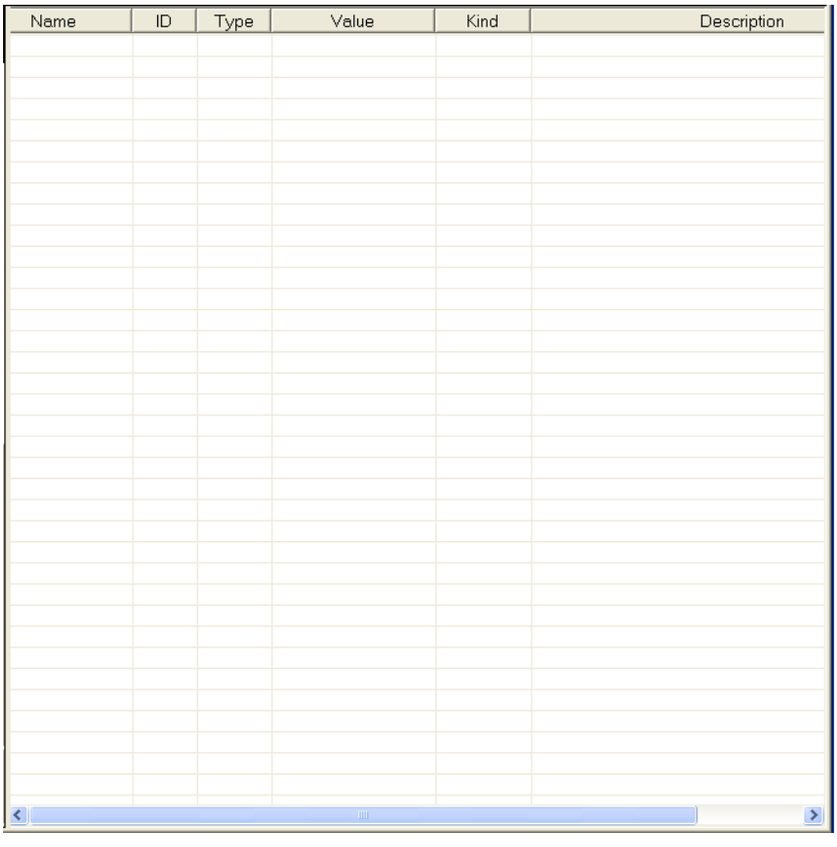


(4) Channel Property Window



1) Channel Name: The channel name should be a unique one. It helps the user to identify InfoU project.
2) Device Name: It lets the user know it is LSIS InfoU.k
3) IP Address: Set up the computer's IP address where InfoU project runs.
4) Project Name: Set up the name of a project which is currently running on InfoU.
5) Project Path: Set up a location where InfoU project exists.
6) Update Cycle(ms): Set up a cycle to import tag values from InfoU.
7) Update Mode: Decide whether to provide real tag values on InfoU or simulation in OPC DA Server.

(5) Tag List Window

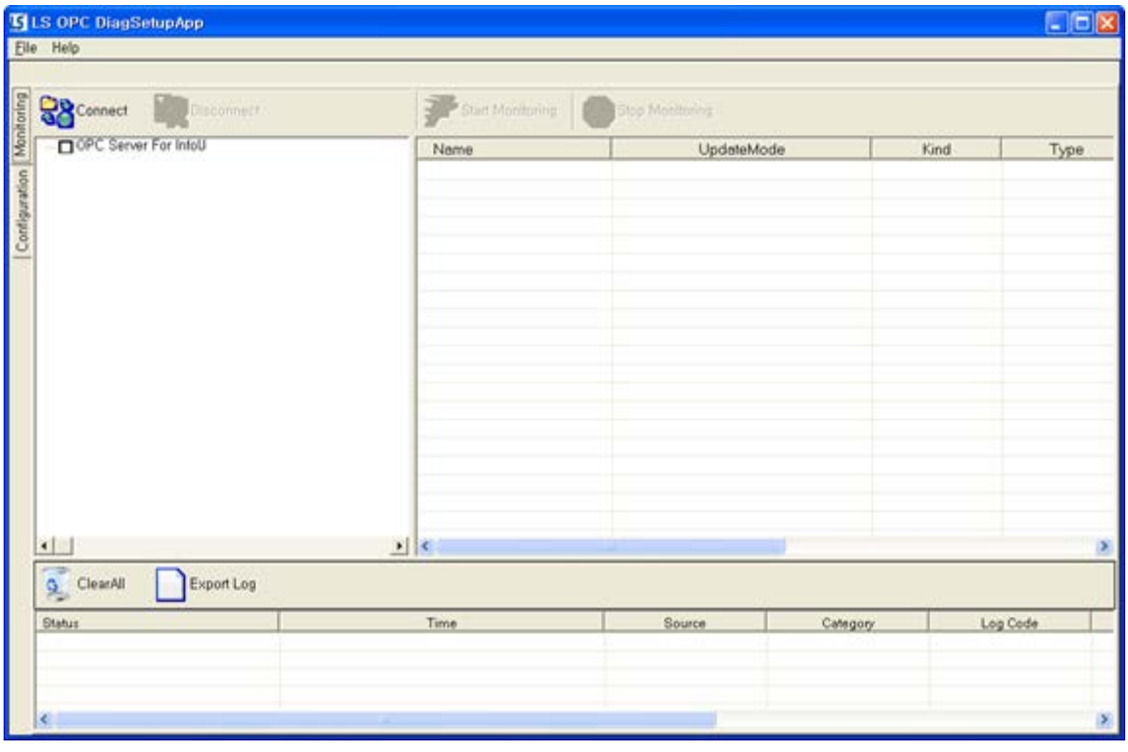


(6) Description Window

Description on each field of the Property Window is displayed.

Channel Name
Specifies the Channel Name

2.3.2 Monitoring



(1) Toolbar

1) 

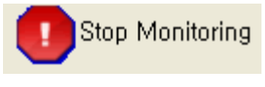
: Connect to LSIS InfoU OPC Server.

2) 

: Disconnect to LSIS InfoU OPC Server.

3) 

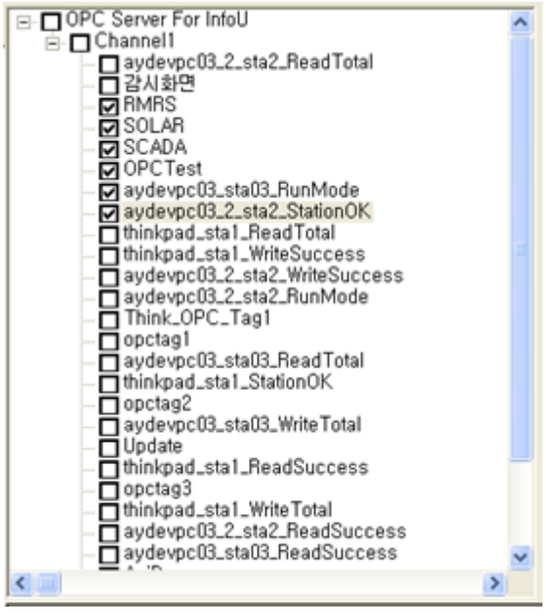
: Start monitoring.

4) 

: Stop monitoring.

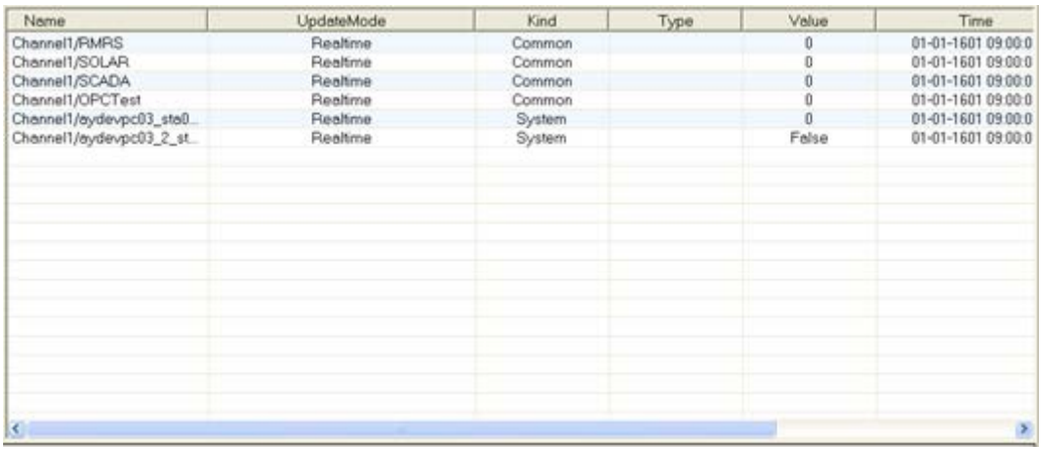
(2) Tag Window

This window shows all of the tag lists registered with the connected LSIS InfoU OPC Server and allows the user to select only the tags to be monitored.



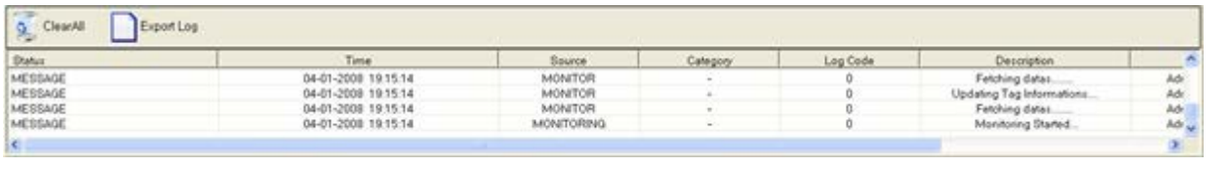
(3) Monitoring Window

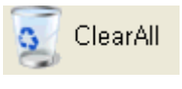
This window shows the selected tags' values.



(4) Log Window

1) This window shows the log information generated during the monitoring.



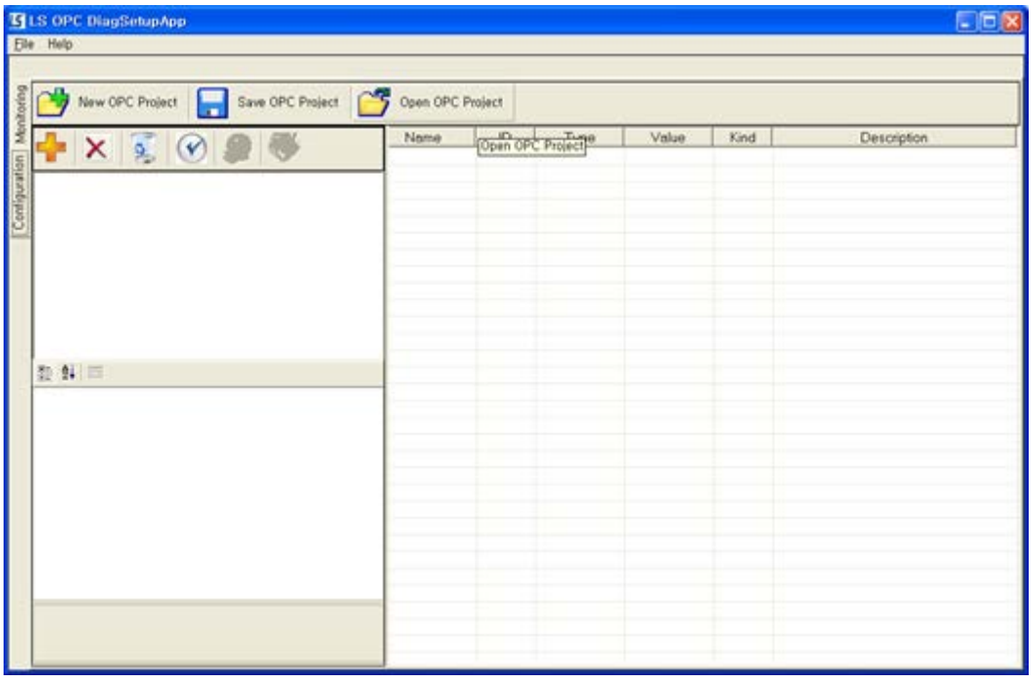
2) 

: Delete the contents from the log list.

3) 

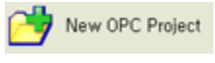
: Export log information to a file.

2.4 OPC Project Configuration



2.4.1 Create New OPC Project

(1) Create project

To create a new OPC project, click on 

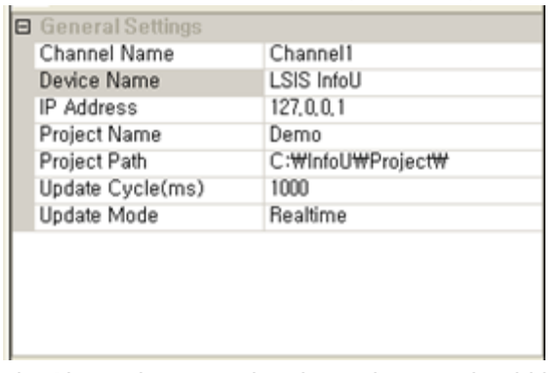
.

(2) Add channel

To add a channel to the OPC project, click on .

(3) Channel property

On the created channel property window, select basic information on InfoU project to be acquired from LS InfoU OPC DA Server and a cycle to acquire data.

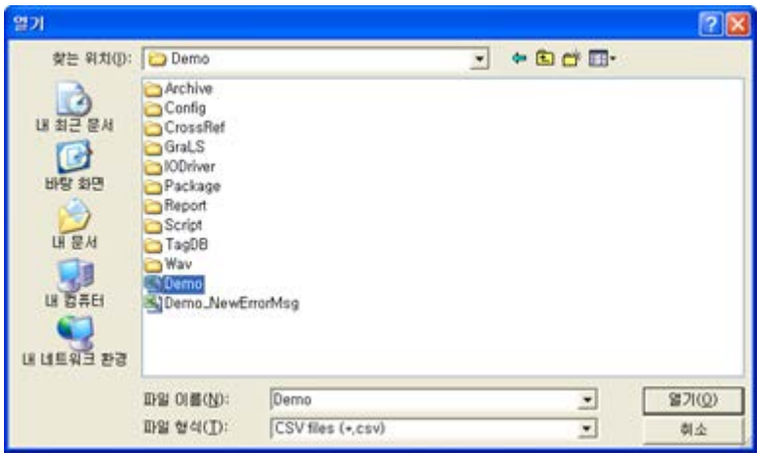


1) Channel Name: The channel name should be a unique one. It helps the user to identify InfoU project.
2) Device Name : It lets the user know it is LSIS InfoU.
3) IP Address : Set up the computer's IP address where InfoU project runs.
4) Project Name : Set up the name of a project which is currently running on InfoU.

5) Project Path : Set up a location where InfoU project exists.
6) Update Cycle(ms) : Set up a cycle to import tag values from InfoU.
7) Update Mode : Decide whether to provide real tag values on InfoU or simulation in OPC DA Server.
8) To check a channel for its communication status, click on .

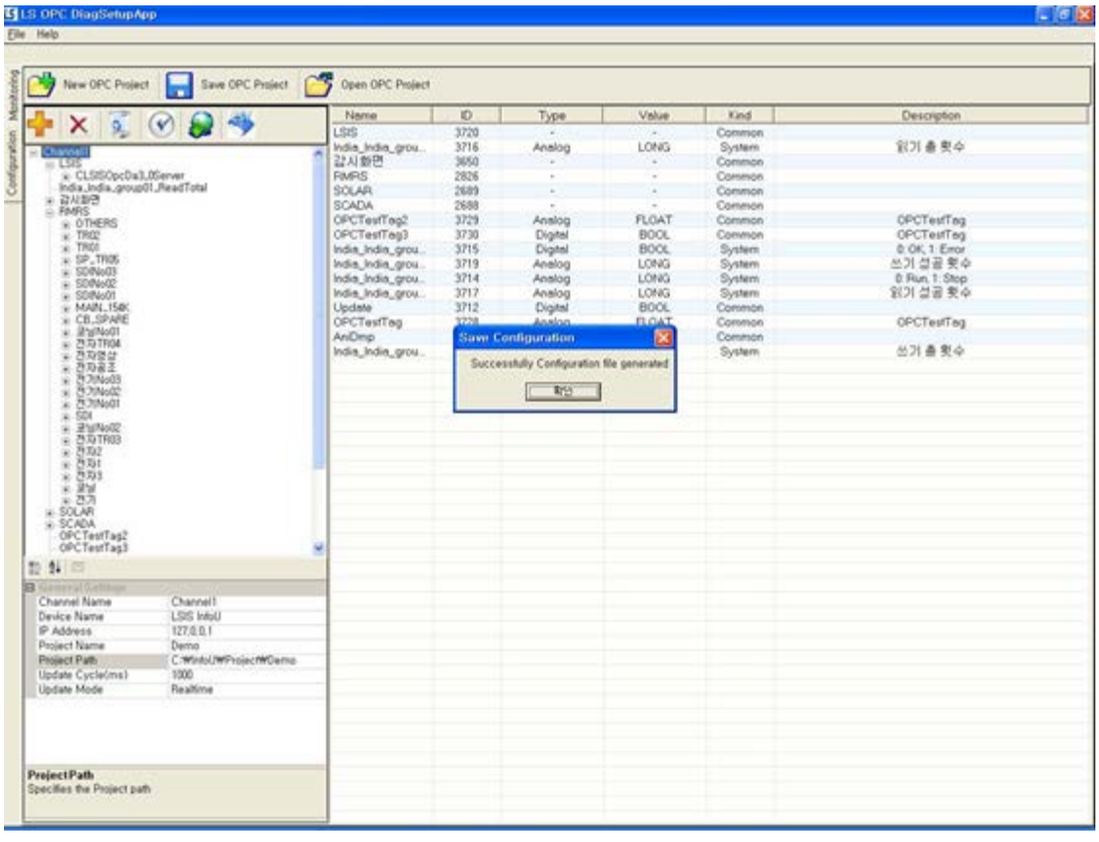


9) To load a tag list from the channel to register, click on . Or, click on  to select a relevant CSV file created on InofUD and import it.



10) Click on 

to create Configuration File(LSISOpcDaServer.config.xml), a OPC project. Configuration File is created at the position where the program is installed and this file is loaded to import tag data when LSIS InfoU OPC DA Server starts running.



2.4.2 Open or Modify OPC Project

(1) Open OPC Project

Select 

to import Configuration File (LSISOpcDaServer.config.xml), one of existing OPC projects.

(2) Channel property

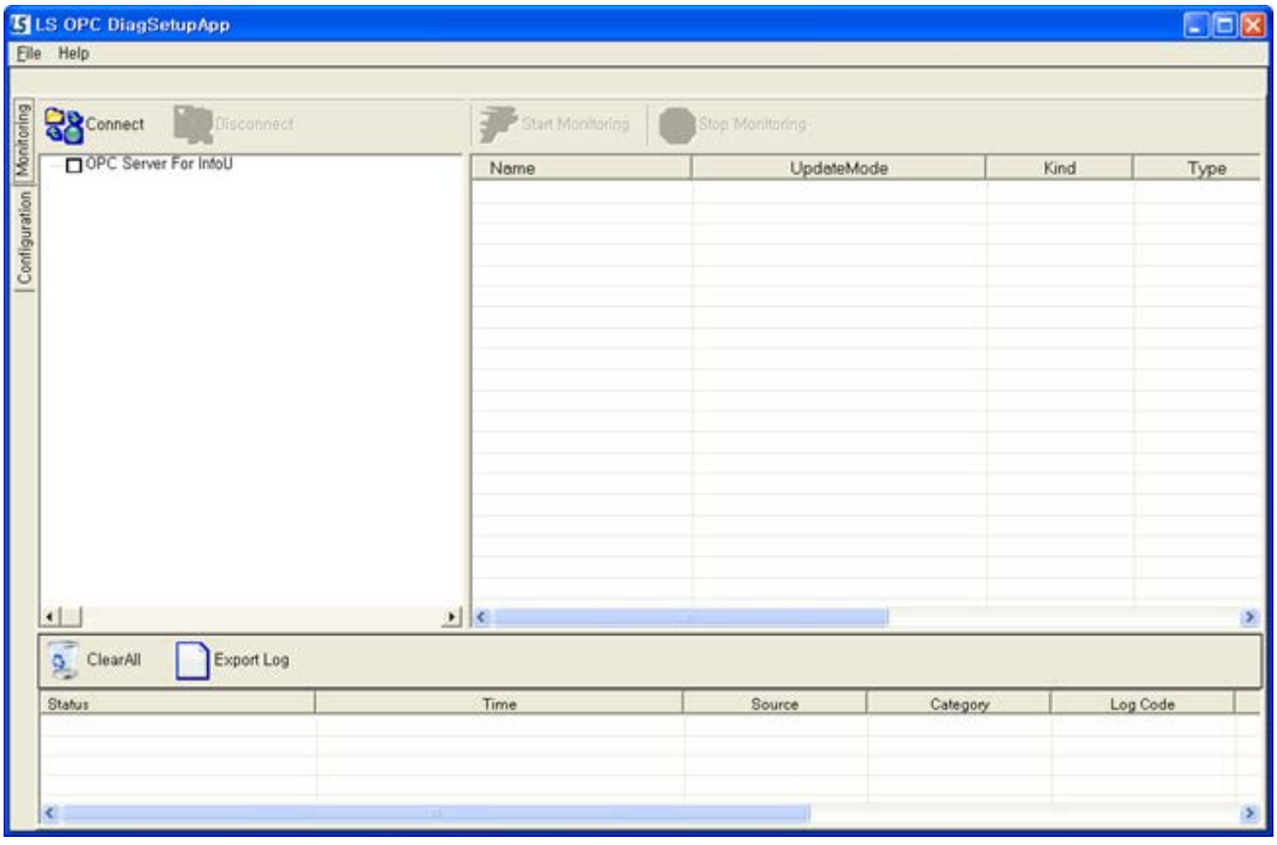
Update the Project Name, Project path or IP Address on the channel property window.

(3) Save OPC Project

Click on 

to save the modified information on OPC project in Configuration file (LSISOpcDaServer.config.xml).

2.5 Run Monitoring

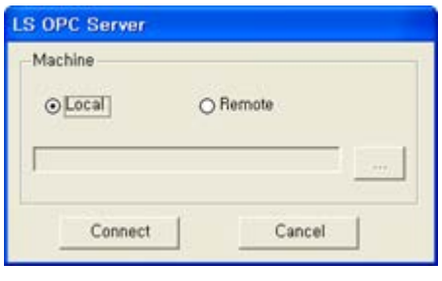


2.5.1 Connect LS InfoU OPC Server / Start Monitoring

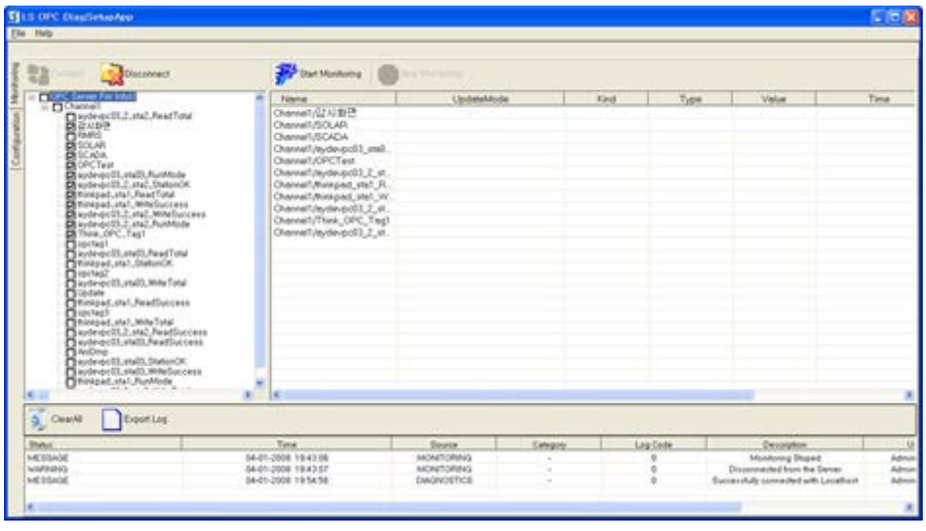
(1) Connect OPC Server

1) Click on 

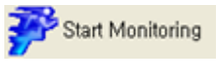
tool bar. The following window appears to allow the user to select an OPC server to be connected to.

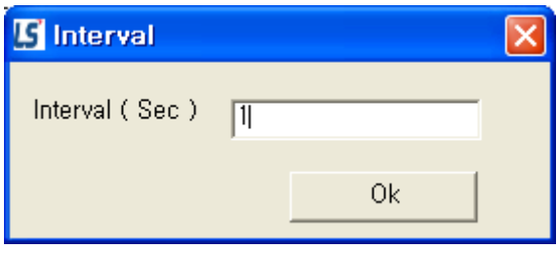


2) To select Local, click on “Connect” button. To select Remote, press “Remote” radio button and directly input the name of a computer on which LS InfoU OPC Server runs or input the name through the browser and then, click on “Connect” button.

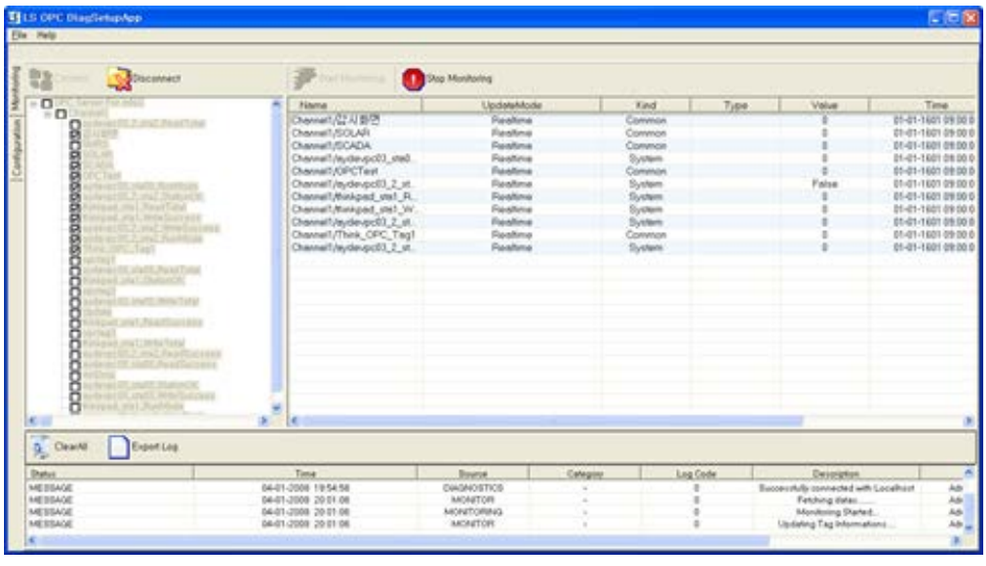


3) Select a tag to be monitored through the tag Window. Register the selected tag with the monitoring Window.

4) Click on  toolbar. The following window appears to allow the user to select an interval to update tag data from LS InfoU OPC Server. Input seconds for interval and select OK button.



5) Start monitoring after making the tag window inactive because it is impossible to select additional tags to be monitored any more. The monitoring window shows the results of monitoring on tag data.



2.5.2 Disconnect LS InfoU OPC Server / Stop Monitoring

(1) To stop monitoring, click on 

.

(2) To stop connecting to the LSIS InfoU OPC Server, click on 

.

2.6 DCOM Configuration

Please refer to the "Appendix3 OPC DCOM Configuration" of InfoU manual.

Appendix 3 OPC DCOM Configuration

To connect to the OPC server in the remote computer, you need to set up DCOM. There are computer settings and settings for the LS InfoU OPC DA server for DCOM Settings. This section describes DCOM settings in Windows 7 and the similar procedures are required for another OS.

3.1 Windows 7 Configuration

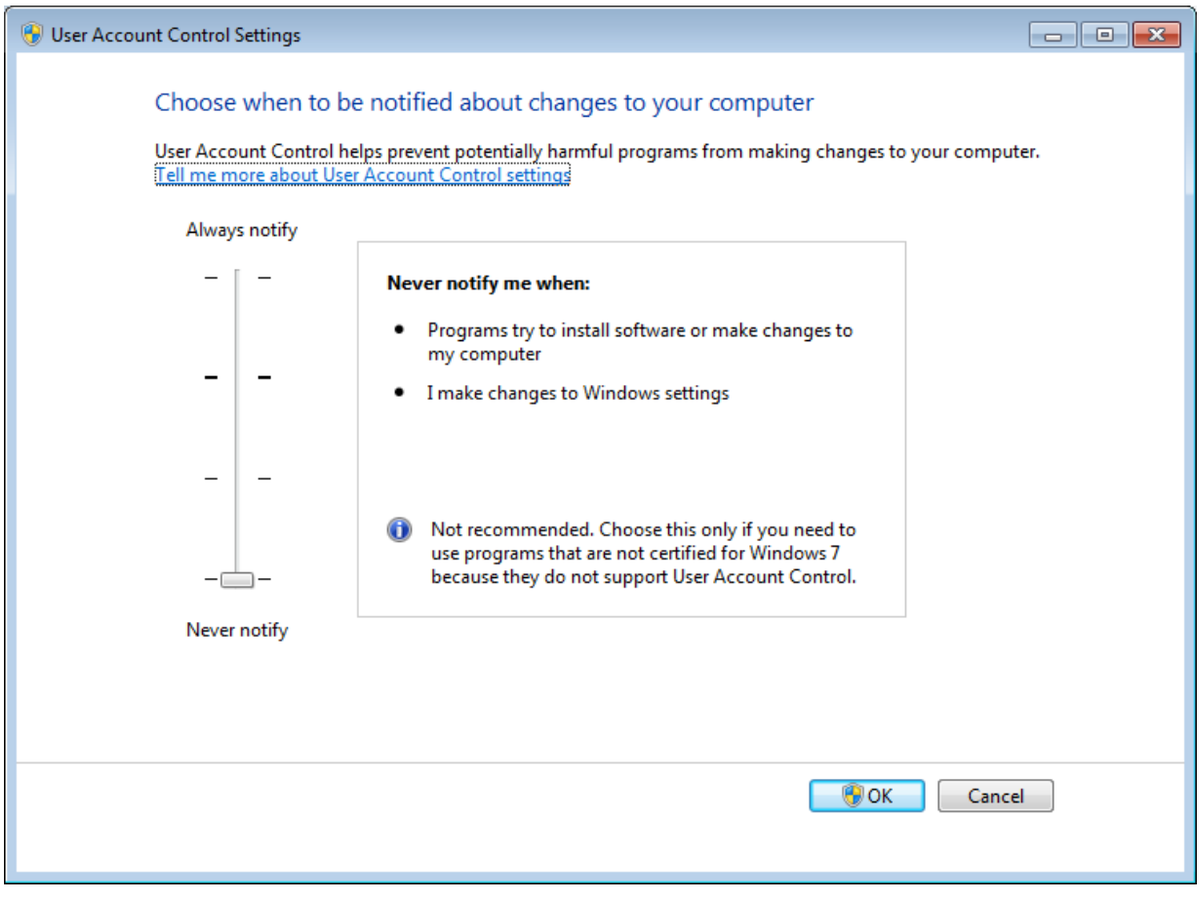
3.1.1 Common (Client&Server) Settings

Common settings should be done for both the Client and the Server. You need to set up them in the Client PC and the Server PC respectively based on the below procedures.

(1) User Account

1) User Account Control Settings

Select [Control Panel] → [User Account] → [User Account Control Settings]



Change the control value into “Never Notify” in the [User Account Control Settings] window. Then, restart Windows.

2) User Name and Password Setting

To operate the OPC function normally, the user should be registered in the Client PC and Server PC under the same name with the same password. In addition, you need to log in to the Client PC and Server PC with the same account.

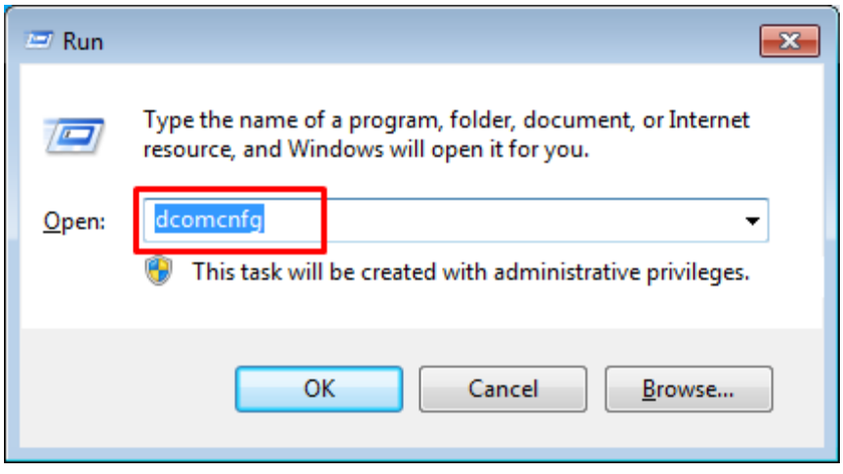
(2) Install OPC Core Components

Download the “OPC Core Components Redistributable” and install it to the PC. You are recommended to download the file of the latest version according to the specifications of the PC.

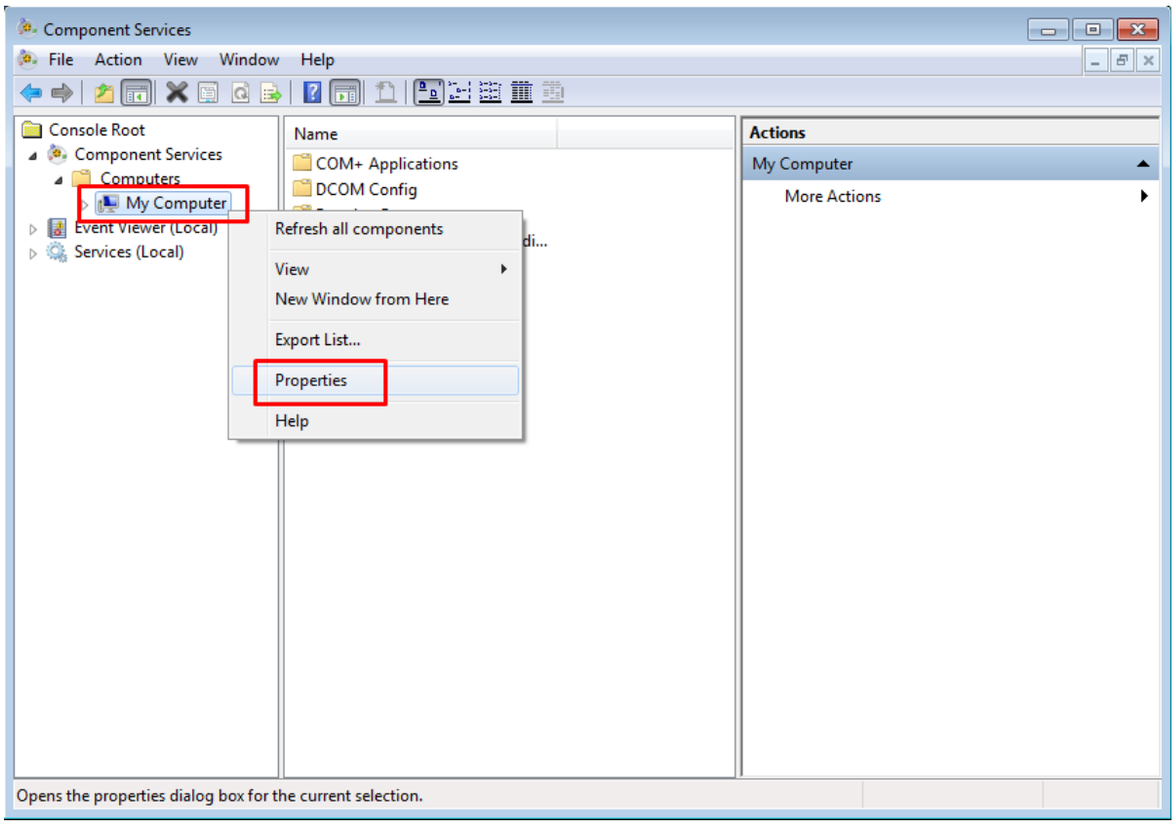
(3) Component Services Setting

1) Computer Setting

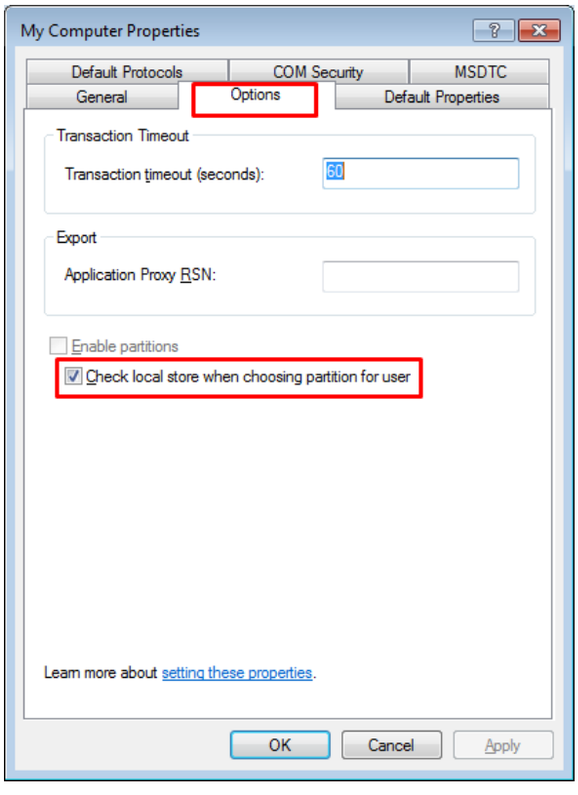
After entering “dcomcnfg” in [Start] → [Run], click the [OK] button.



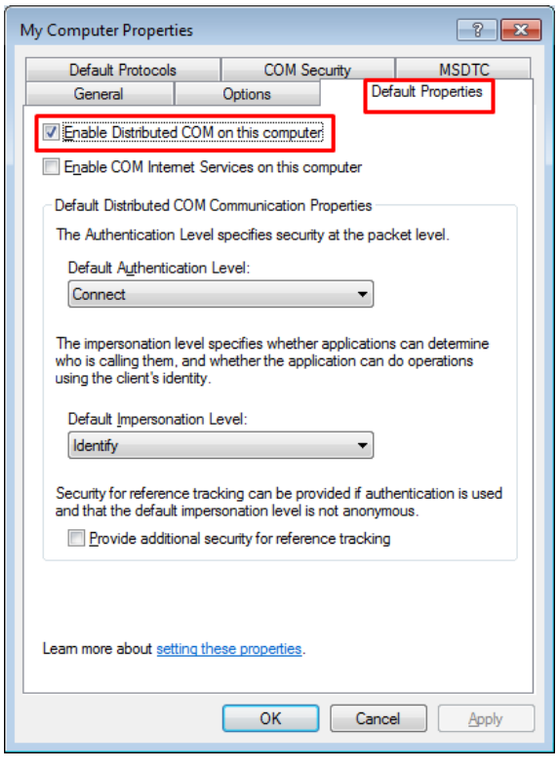
Then, “Component Services” window will be displayed as below. Expand the “Component Services” by clicking the Console Root, click the [Computer]. Click with the right mouse on [Computer]-[My Computer] and run the [Properties] menu.



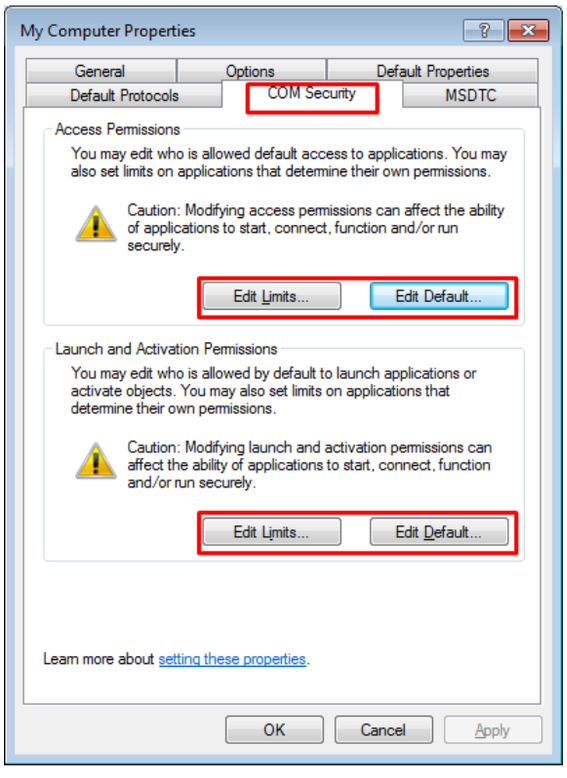
Put a check “Check local store when choosing partition for user” in the [Options] tab.



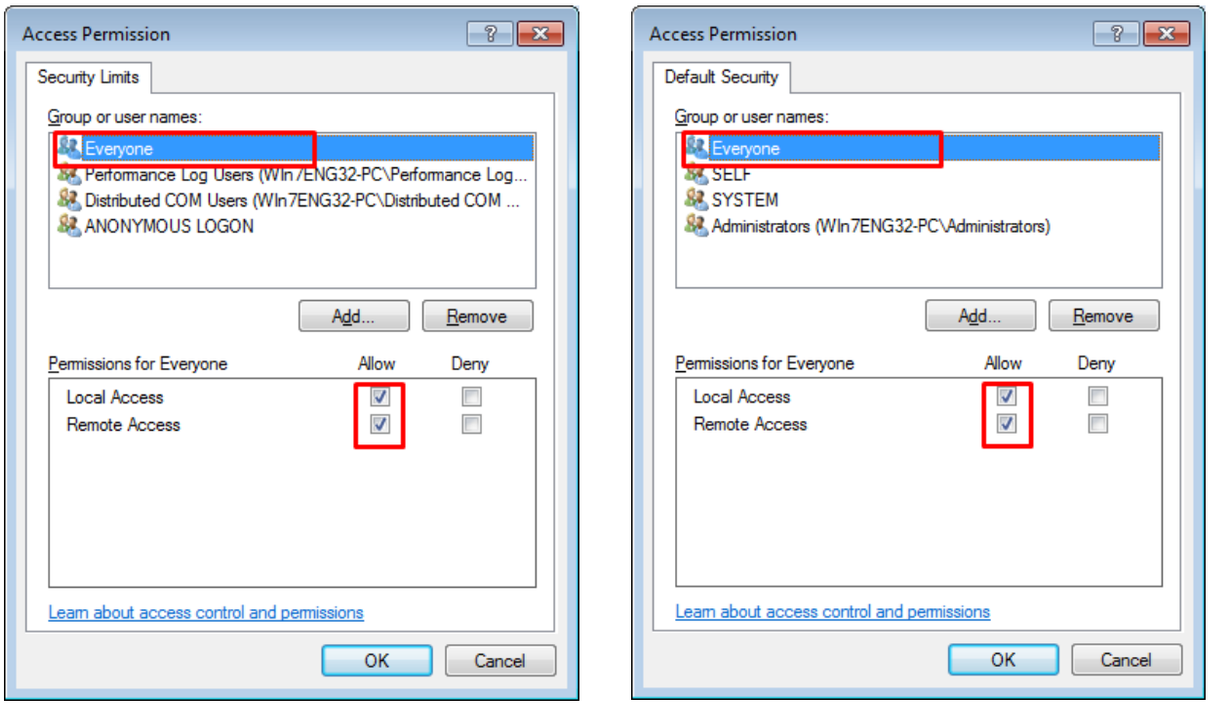
Put a check “Enable Distributed COM on this computer” in the [Default Properties] tab.



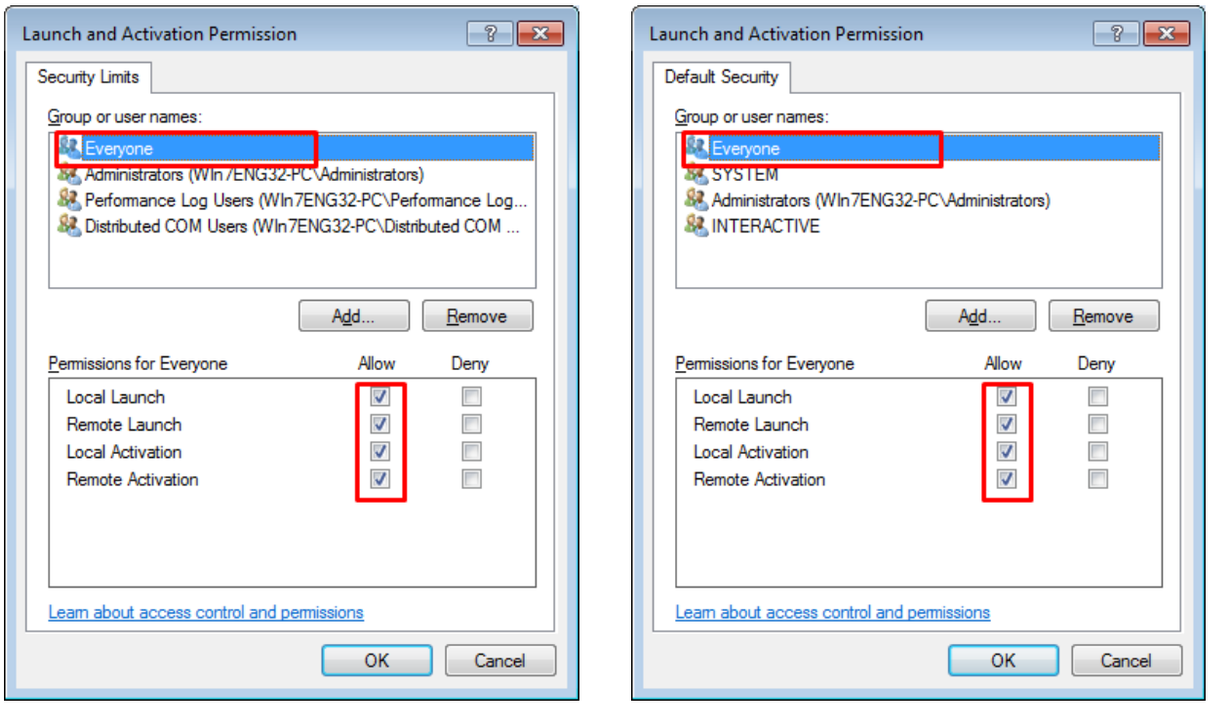
Add the “Everyone” account to “Access Permissions” of the [COM Security] tab and [Edit Limits] of “Launch and Activation Permissions” and [Edit Default] . Then, set the access permission as “Permissions for Everyone”.

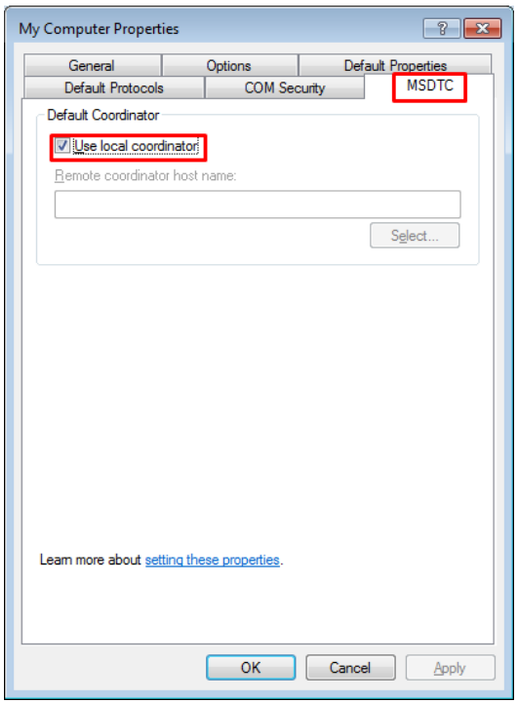


Add the “Everyone” account to [Edit Limits] of “Access Permission” and [Edit Default]. Then, set the access permission as “Permissions for Everyone”.



Add the “Everyone” account to the [Edit Limits] of “Launch and Activation Permission” and [Edit Default]. Then, set the access permission as “Permissions for Everyone”.





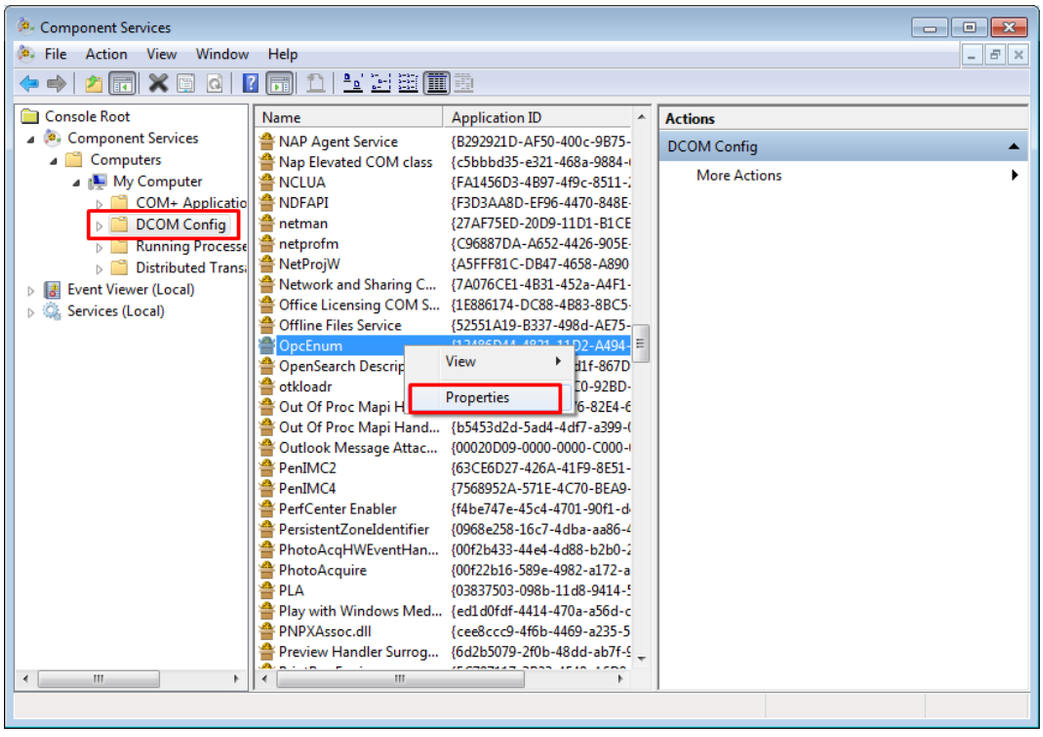
Put a check “Use local coordinator” in the [MSDTC] tab.

After the settings are completed, press the [Apply] → [OK] button and exit the task.

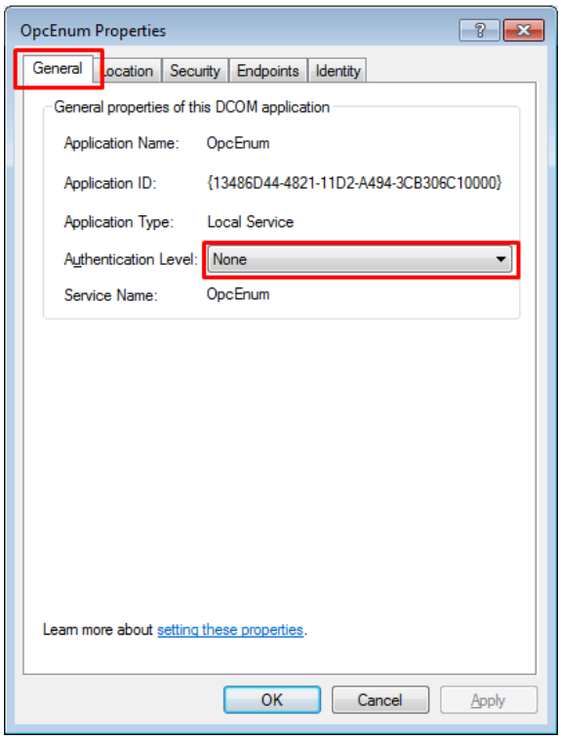
2) OpcEnum Setting

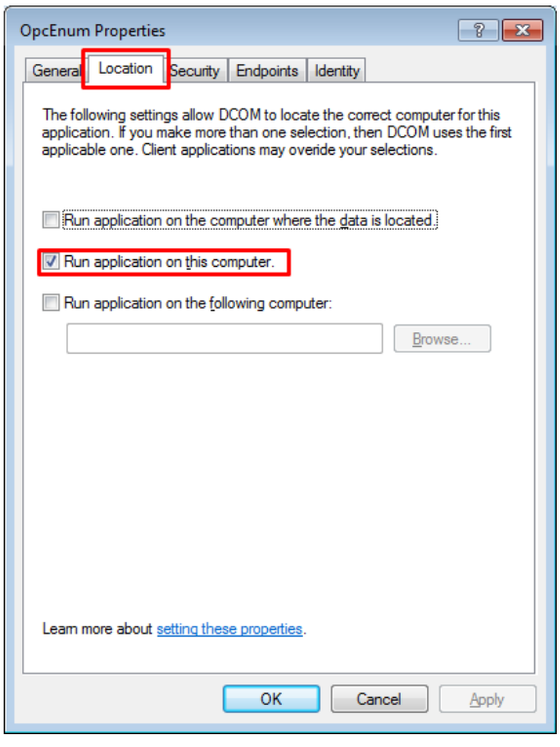
Select [Components Service] → [Computer] → [My Computer] → [DCOM Configuration].

Select “OpcEnum” and click with the right mouse to pick [Properties].

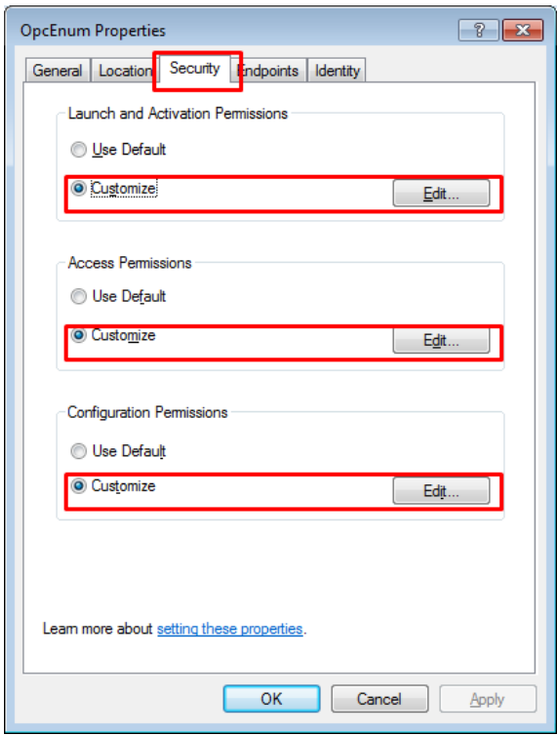


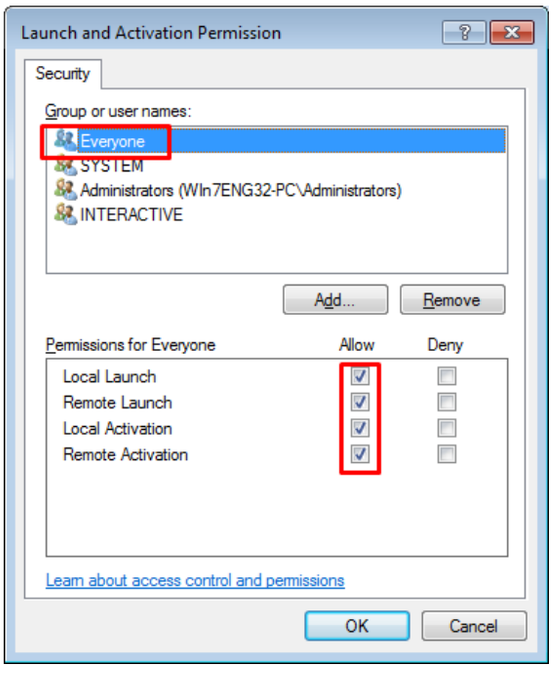
In the [OpcEnum Properties] window, set the details of [General], [Location] tabs as shown below.



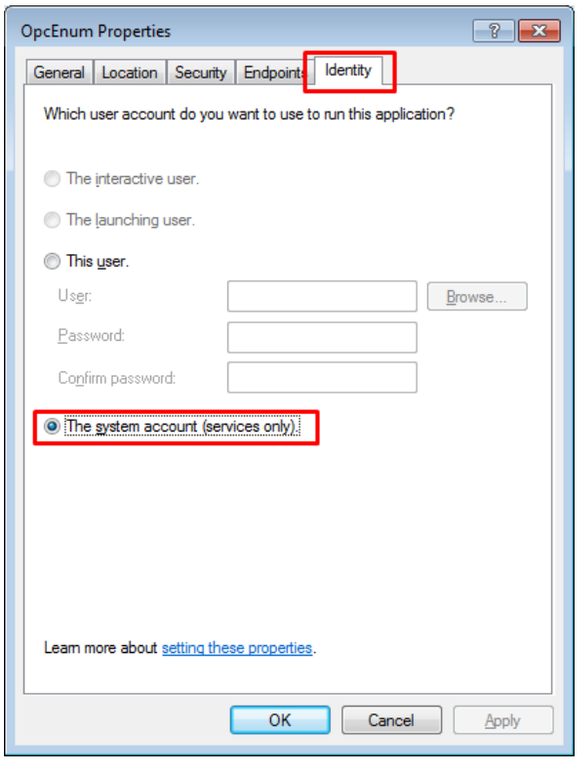


After setting all items of the [Security] tab as “Customize”, press the [Edit] button to add the “Everyone” account to all permissions. Then, set the access permission as “Permissions for Everyone”.





Select “system account (services only)(S)” in the [ID] tab.



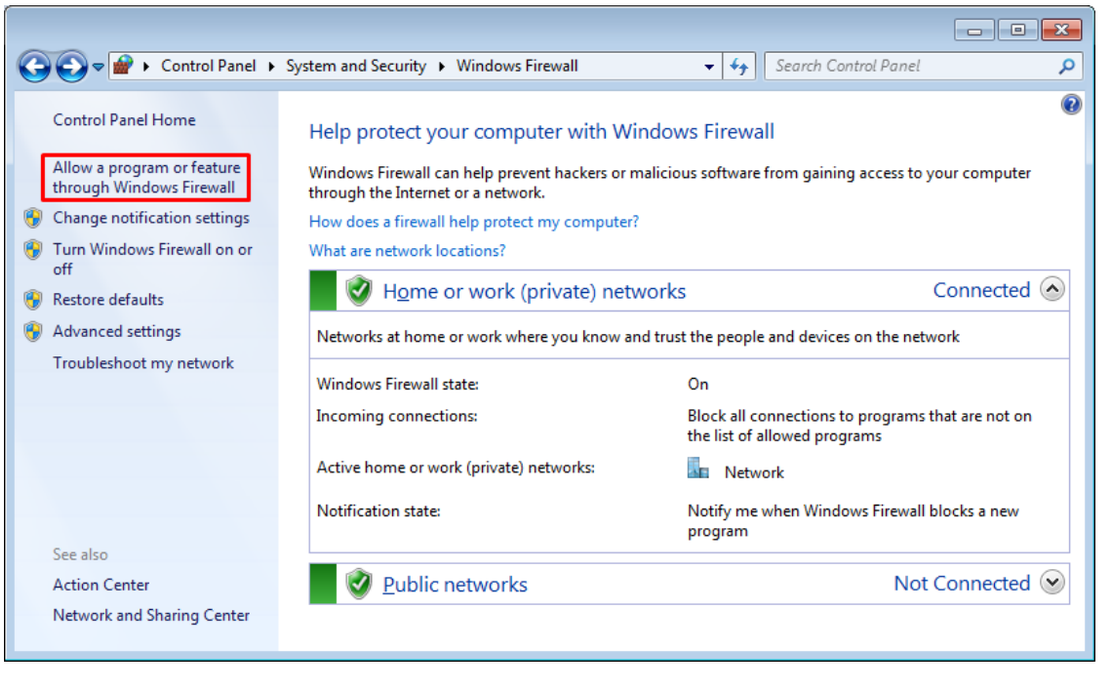
After settings are completed, press the [Apply] → [OK] button and exit the task.

(4) Firewall Setting

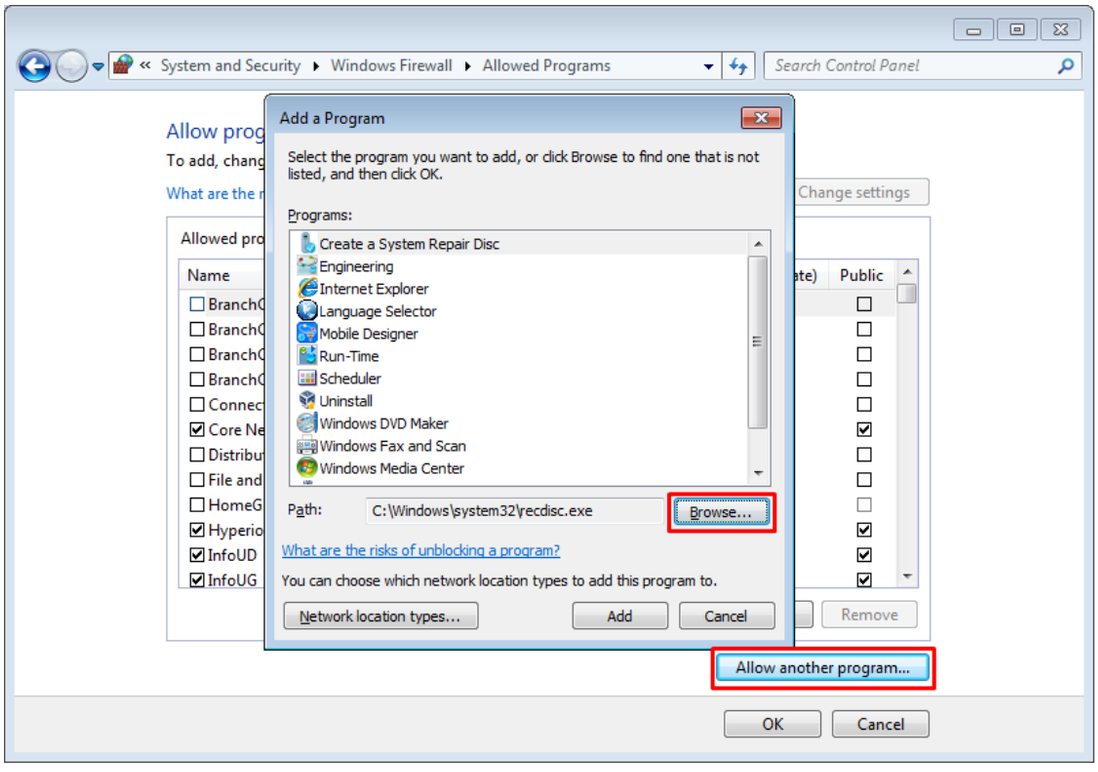
1) Add Program (OpcEnum)

Run [Control Panel] → [Windows Firewall].

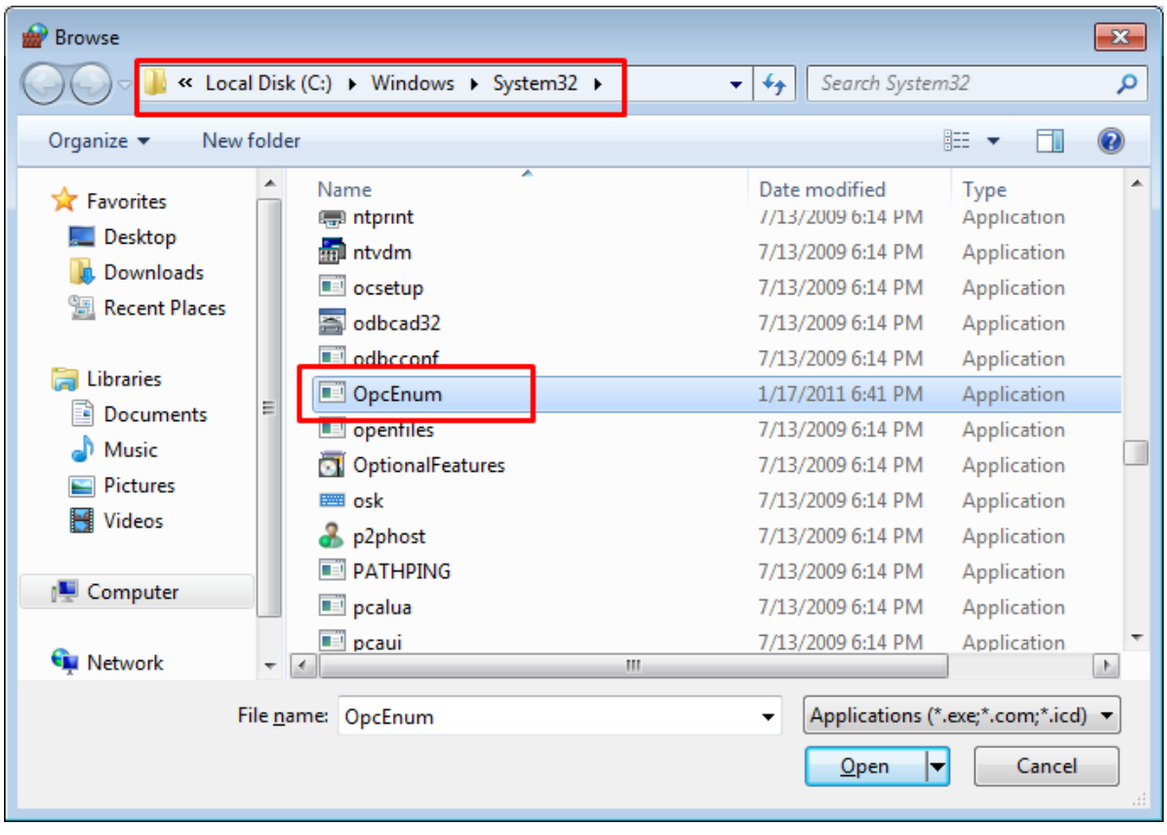
Click “Allow a program or feature through Windows Firewall” on the left side.



Press the [Allow another program] button and then, press [Browse] button in the ‘Add a program’ window.



Add the file C:\Windows\System32\OpcEnum.exe



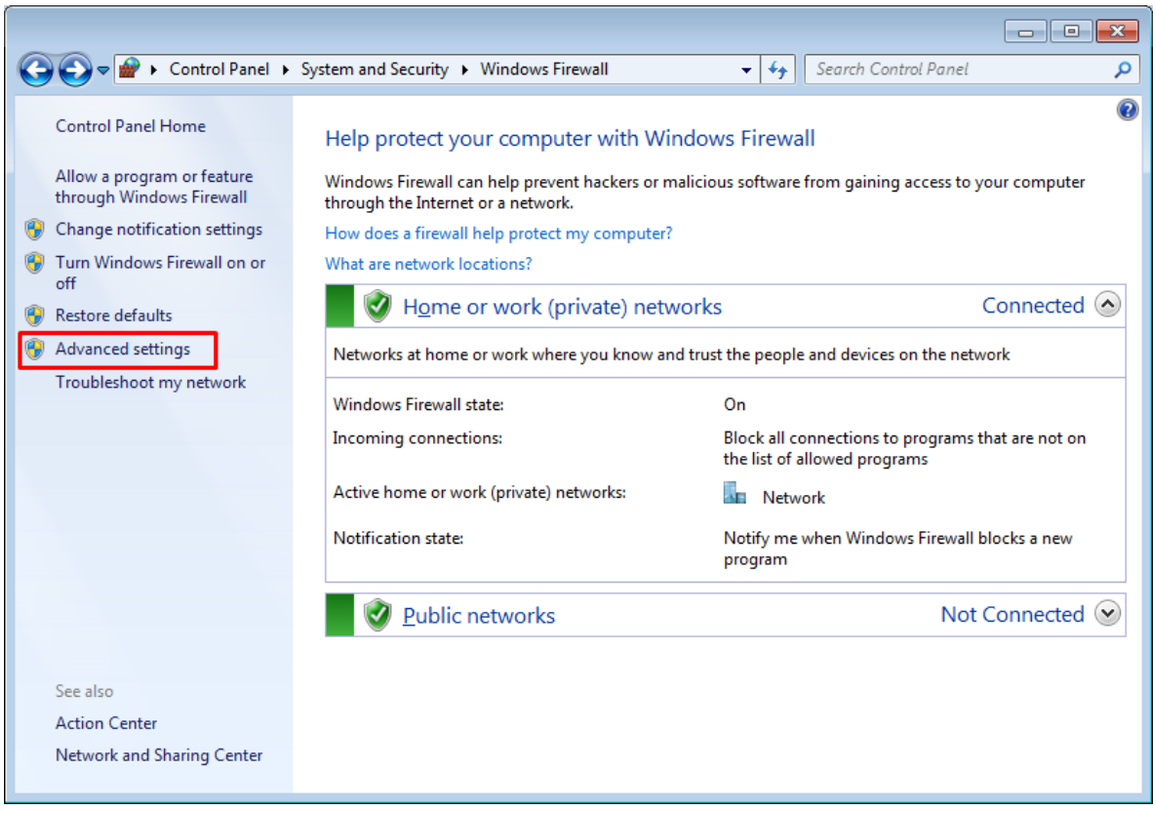
3.1.2 Client Settings

After setting the PC in accordance with the instructions of “Common (Client & Server) Setting”, perform the additional settings based on the below procedures.

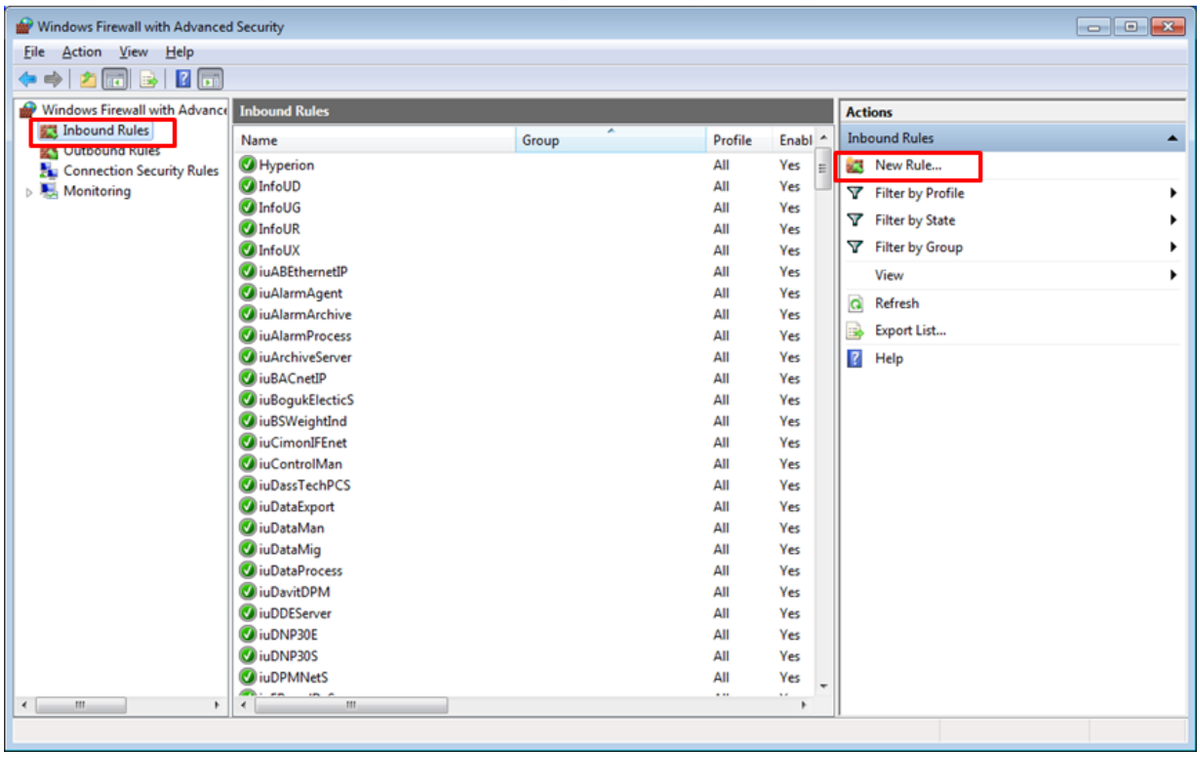
(1) Firewall Setting

1) Inbound Rule Setting (Add Port: 135)

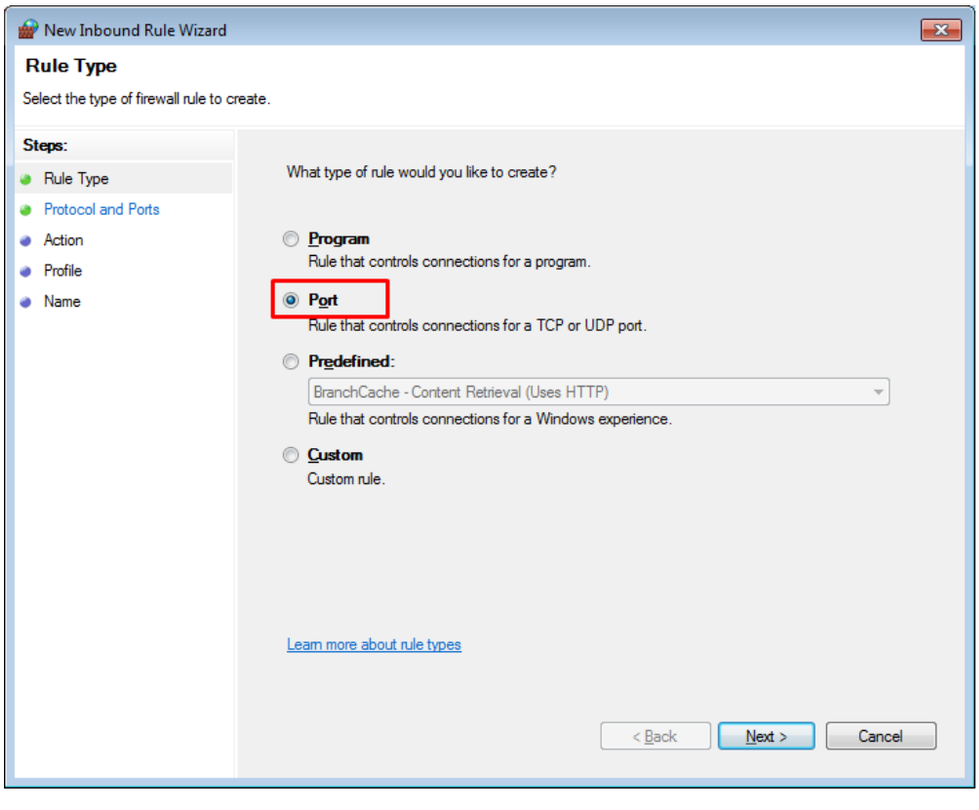
Run [Control Panel] → [Windows Firewall] and select [Advanced Settings].



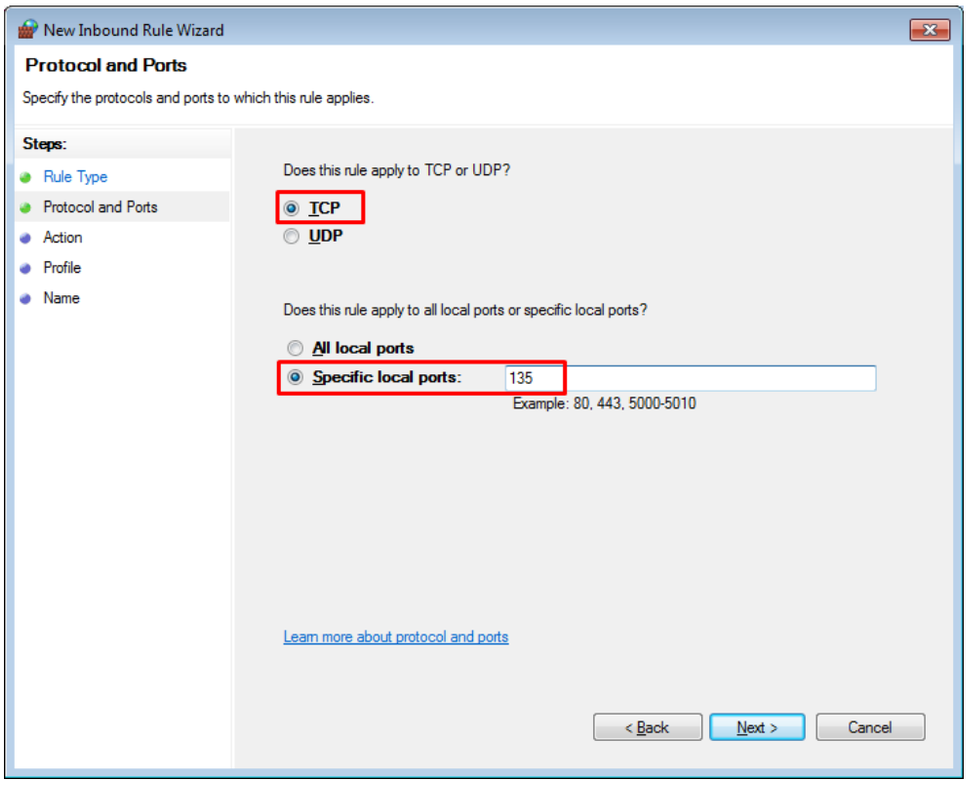
Select [Inbound Rules] on the left side and click [New Rules] on the right side.



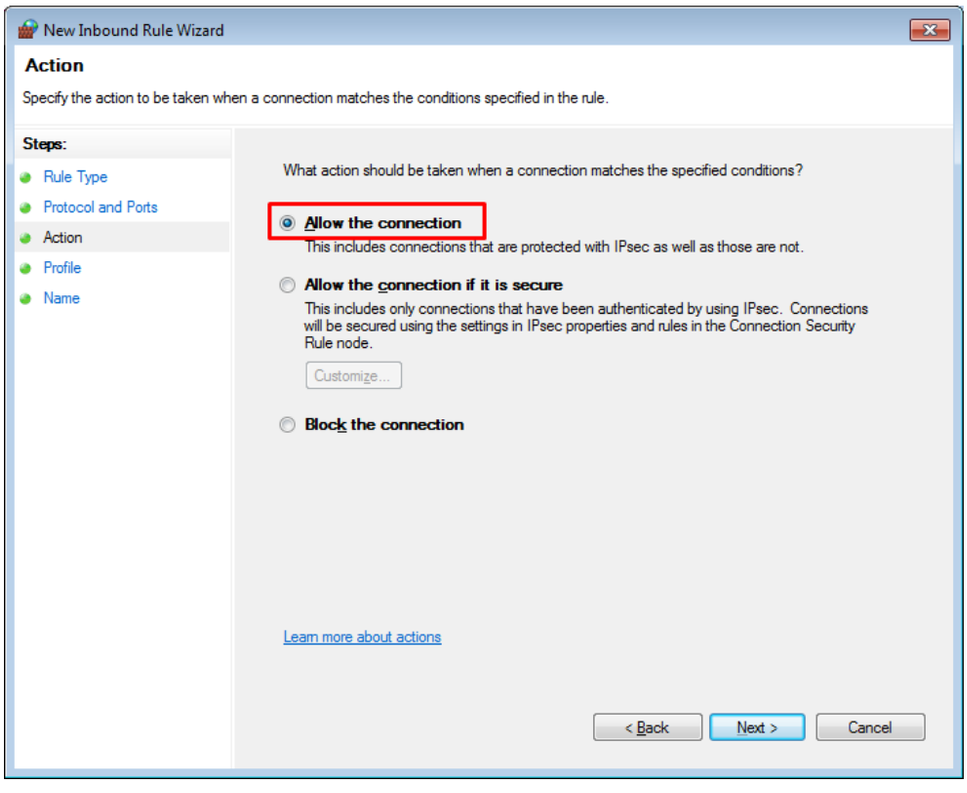
Select the “port” for the rule type that you want to create and click the [Next] button.



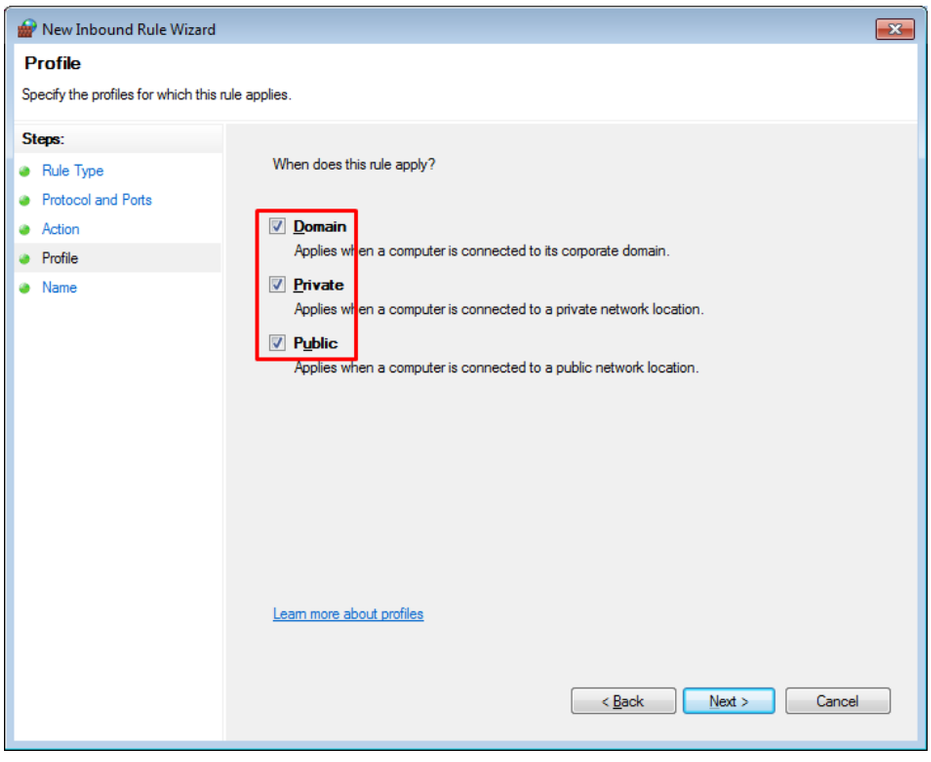
Select the “TCP” to be applied to the new rule and set the port as “Specific local ports”. Then, enter “135” and click the [Next] button.



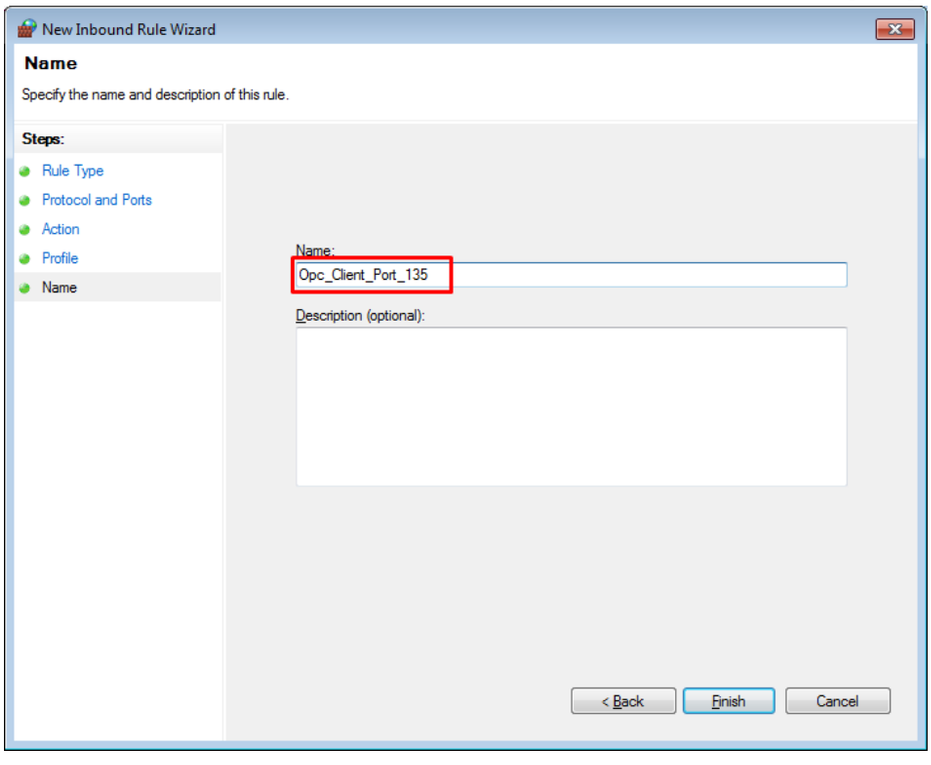
Select “Allow the connection” for the specified task item, click the [Next] button.



Put a check “Domain”, “Private”, “Public” in the profile items and click the [Next] button.



Put a check “Domain”, “Private”, “Public” in the profile items and click the [Next] button.



3.1.3 Server Settings

After setting the PC based on “Common (Client & Server) Setting” and perform the additional settings based on the below procedures.

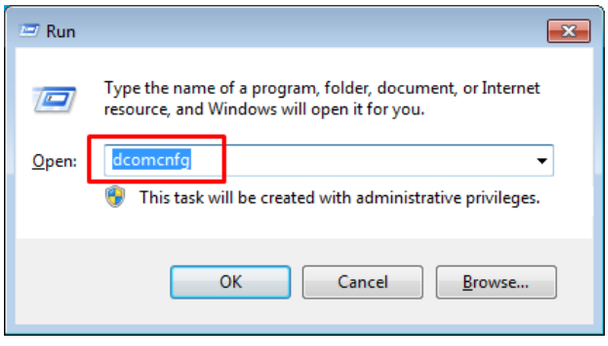
(1) Install LSIS OPC Server

Install the “LSIS OPC Server For InfoU” program.

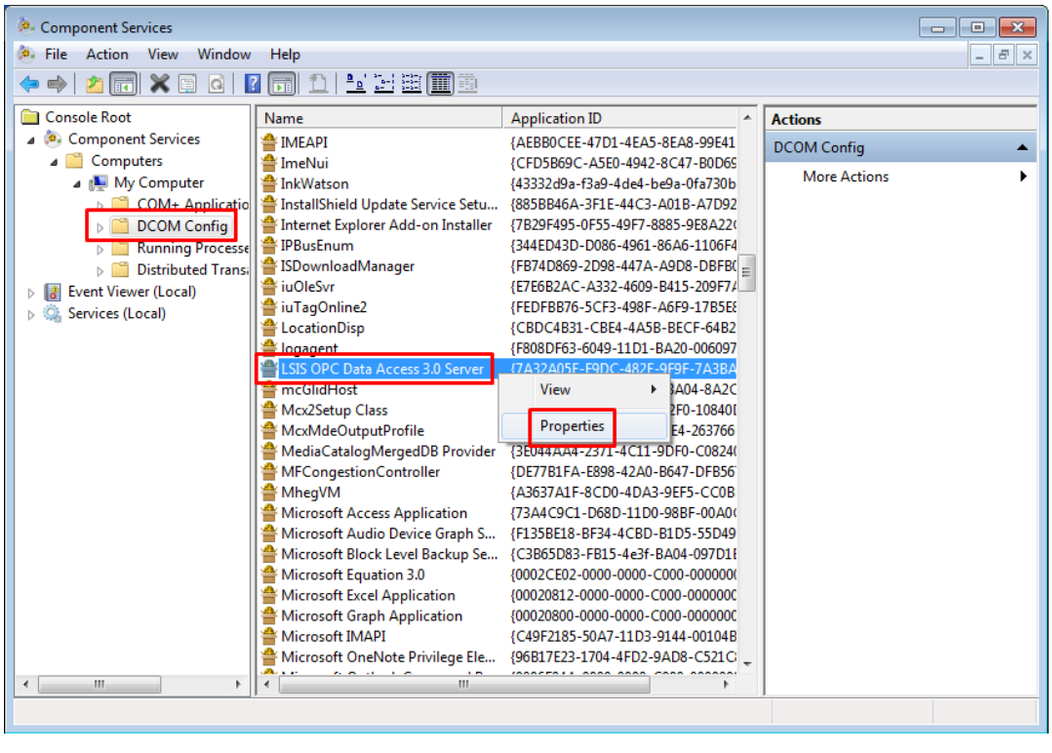
(2) Component Service Setting

1) LSIS OPC Server Setting

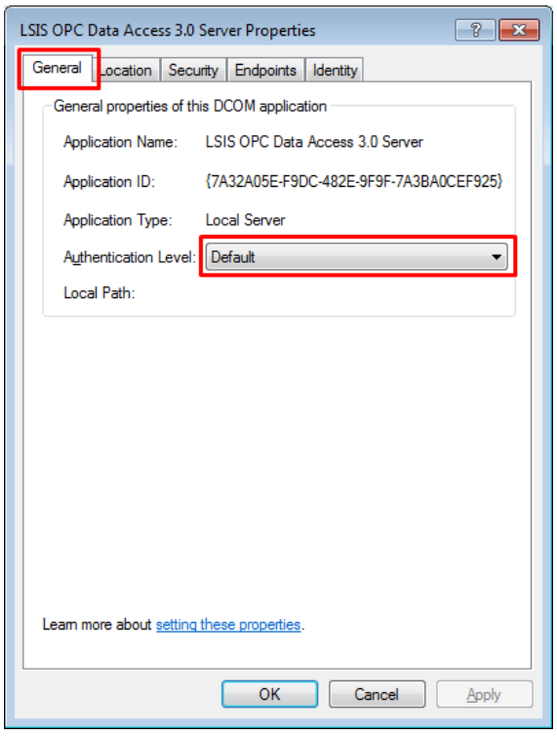
After entering “dcomcnfg” in [Start] → [Run], press the [OK] button.

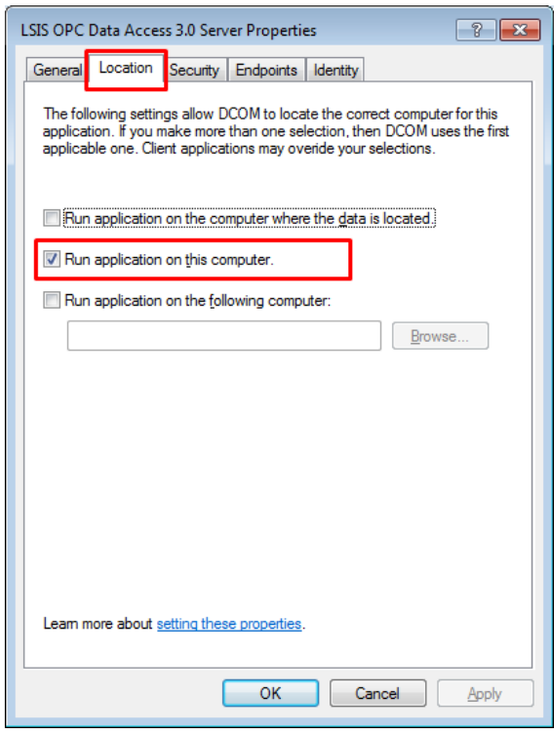


After selecting [Computer] → [My Computer] → [DCOM Configuration], click with the right mouse button on the “LSIS OPC Data Access Server” of the right list and run the [Properties] menu.

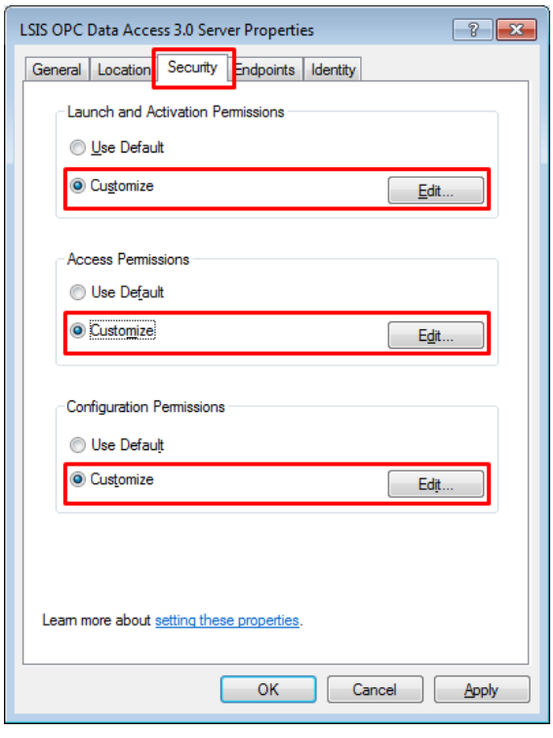


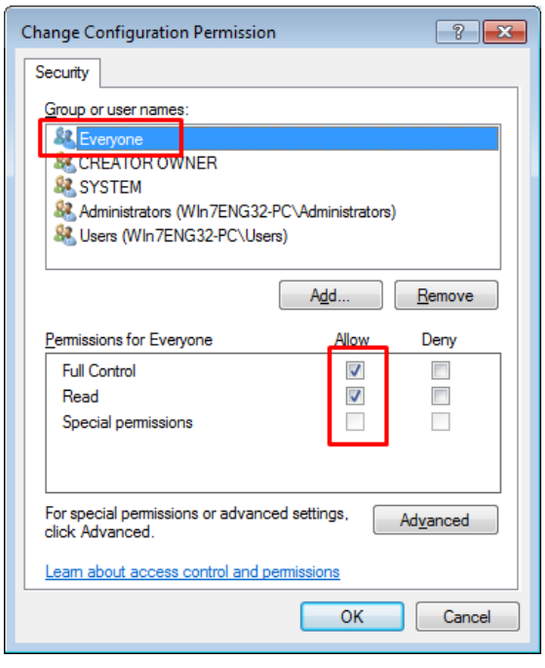
Set the [General] tab and [Location] tab as below.



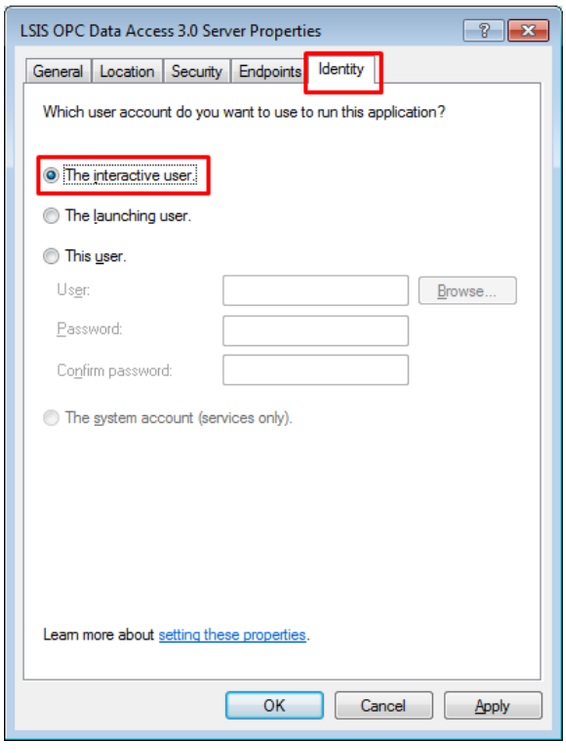


After setting all items of the [Security] tab as “Customize”, press the [Edit] button to add the “Everyone” account to all permissions. Then, set the access permission as “Permissions for Everyone”.



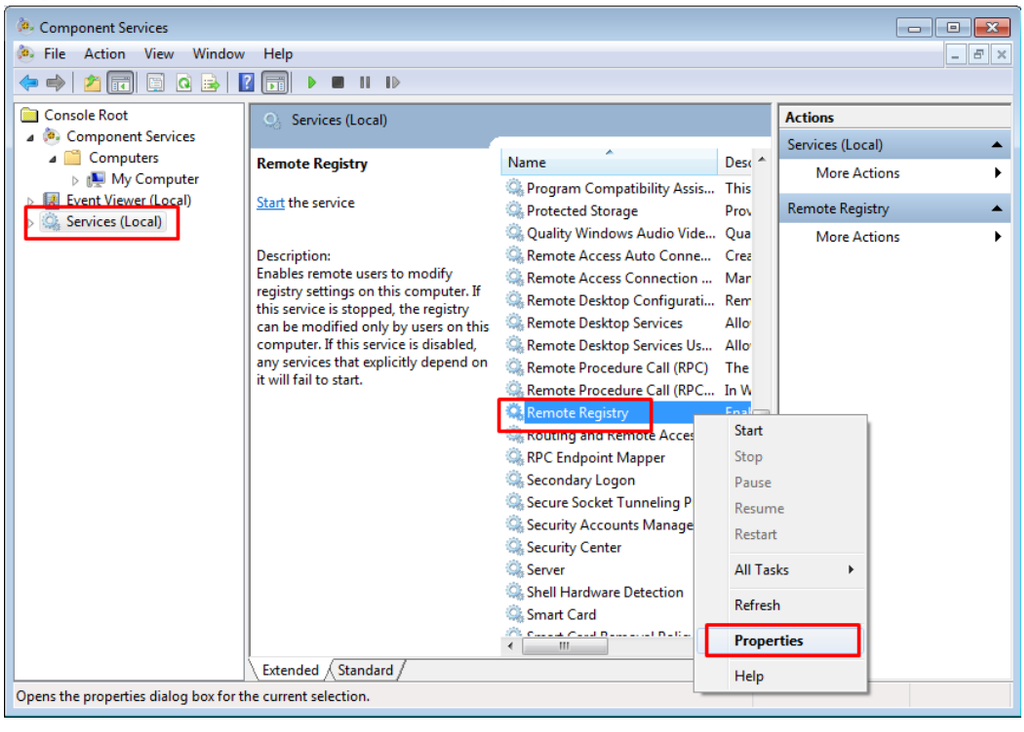


Select “The interactive user” in the [ID] tab.

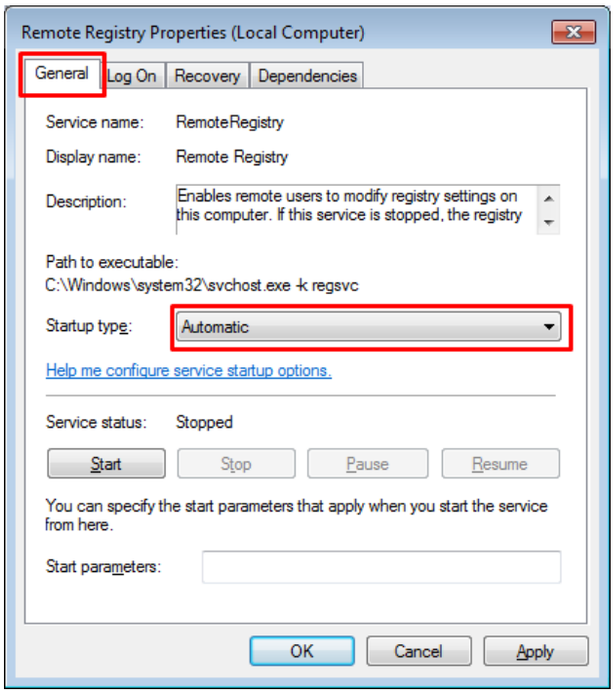


2) RemoteRegistry Setting

After selecting [Services(local)] in the Dcomcnfg setting window, click with the right button on the “Remote Registry” of the right list and select the ‘Properties’.



Set “Start type” as “Automatic” in the [General] tab.

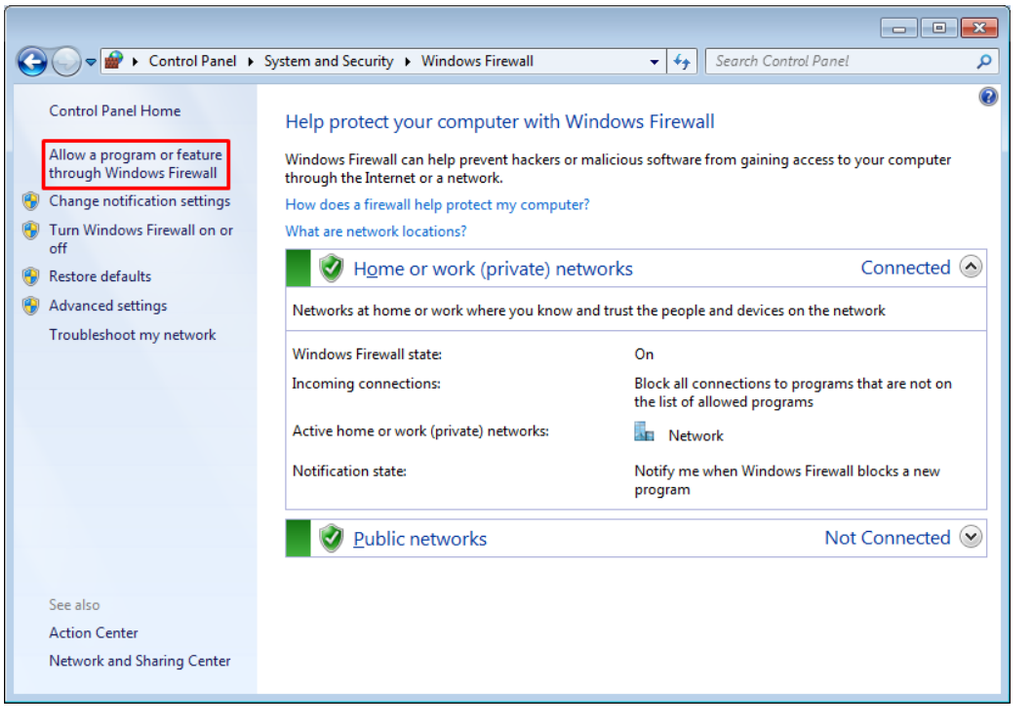


(3) Firewall Setting

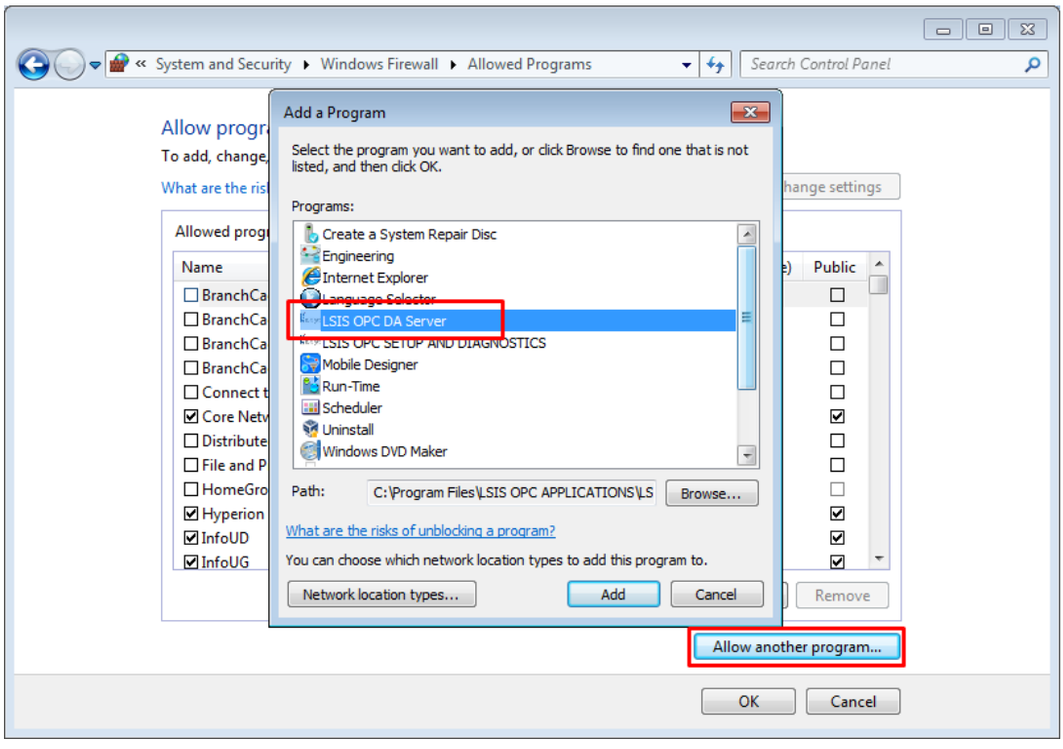
1) Add Program (LSIS DA Server)

Run [Control Panel] → [Windows Firewall].

Click “Allow a program or feature through Windows Firewall ” on the left side.

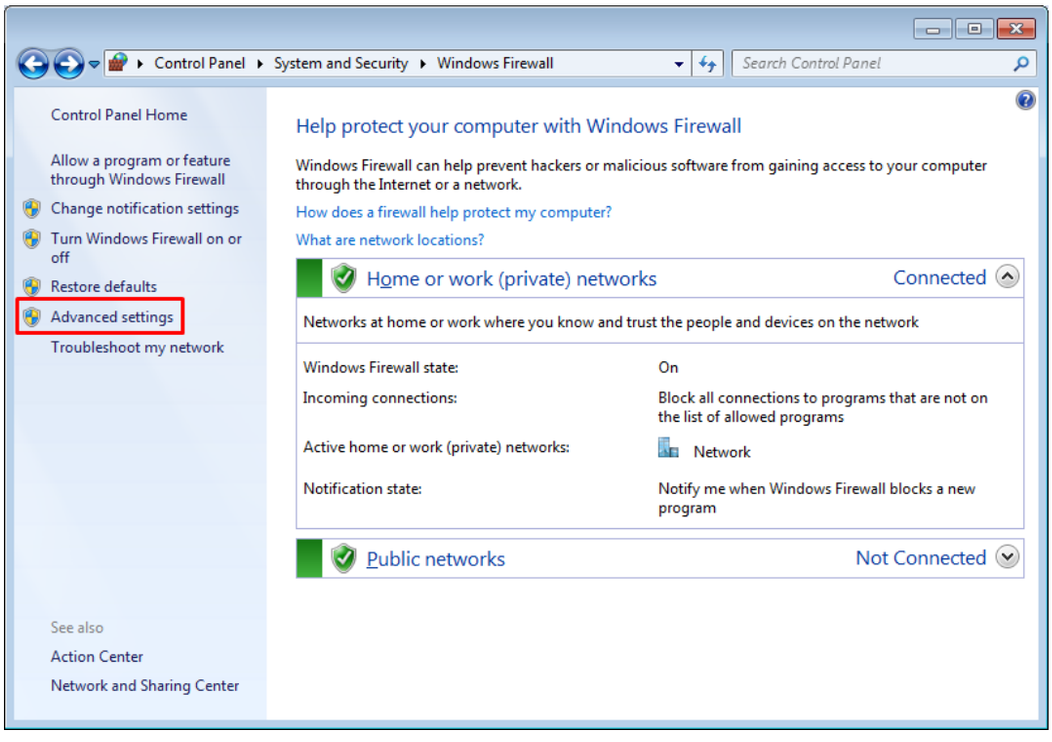


Press the [Allow another program] button. Then, select “LSIS OPC DA Server” in the list of ‘Add a program’ window and press the [Add] button.

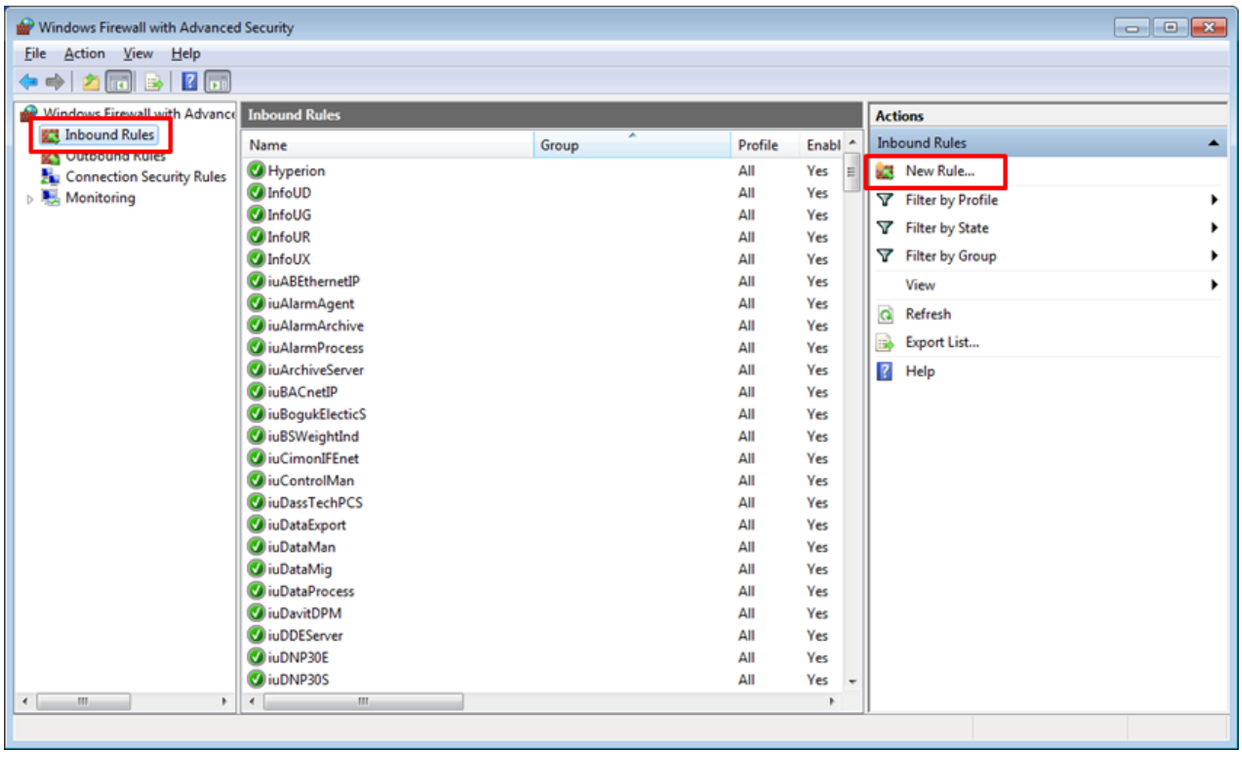


2) Inbound Rule Setting (Add Port: 135-139)

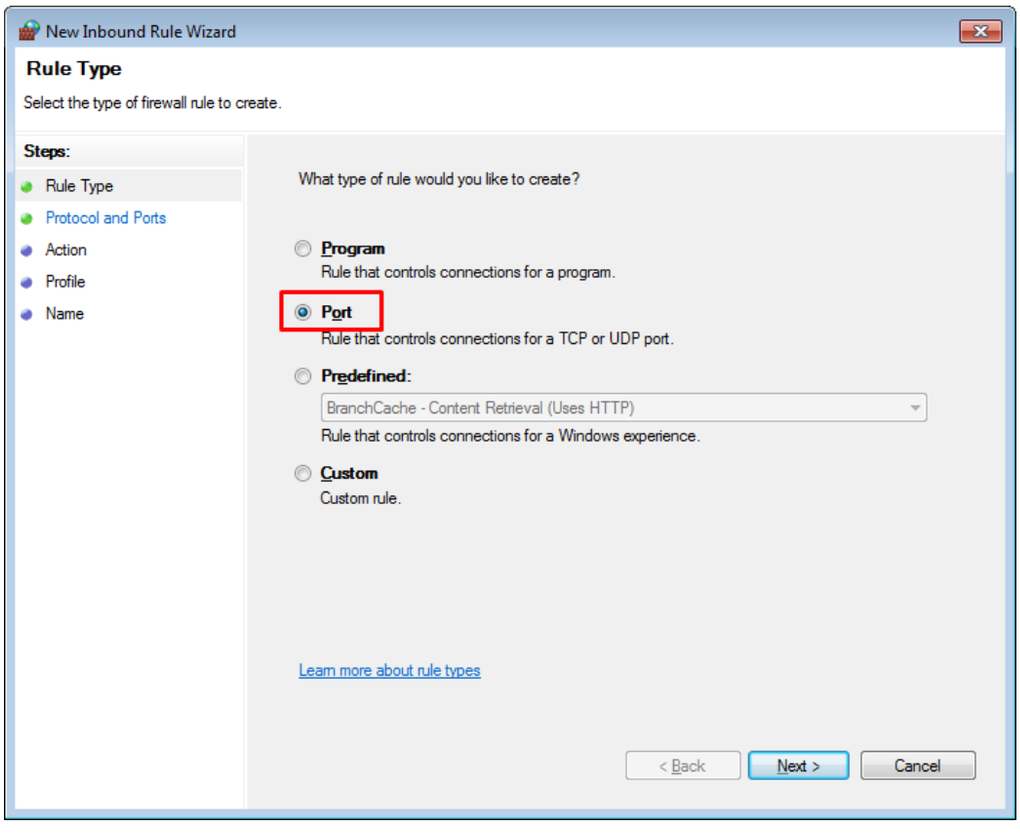
Run [Control Panel] → [Windows Firewall] and click “Advanced Settings” on the left side.



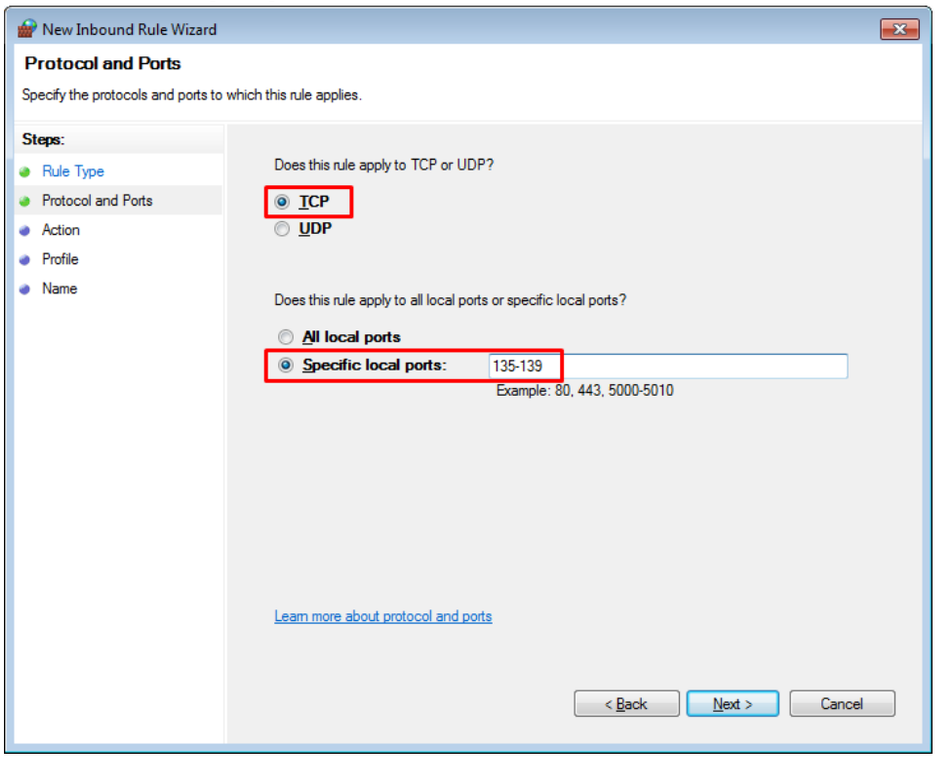
Select [Inbound Rules] on the left side and click [New Rules] on the right side.



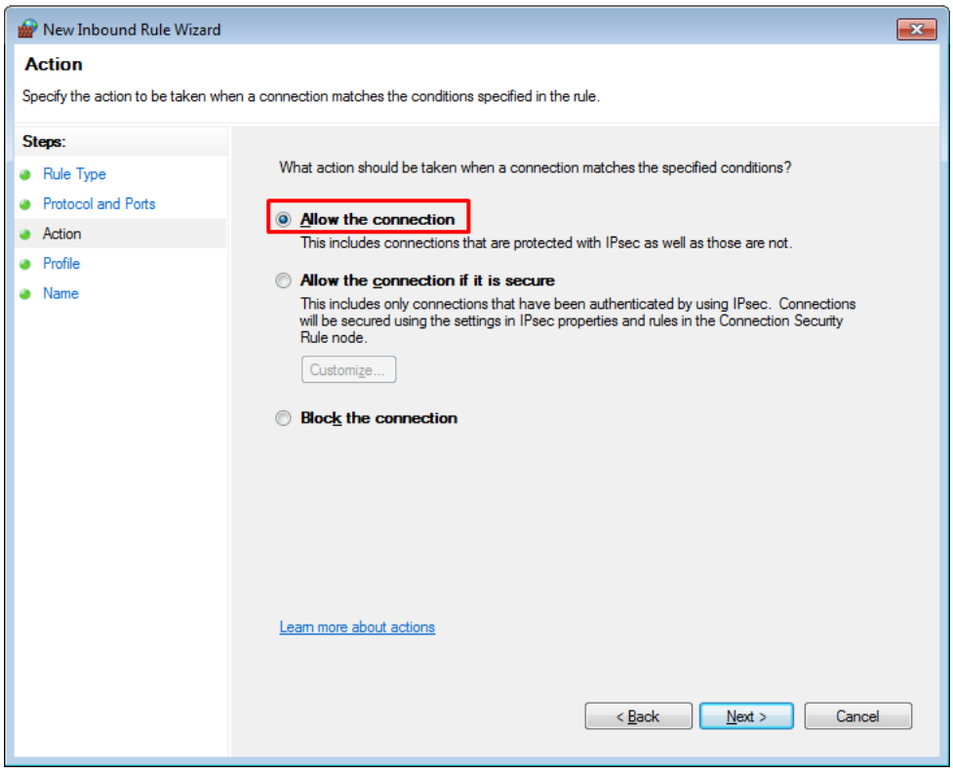
Select the “Port” for the rule type that you want to create and click the [Next] button.



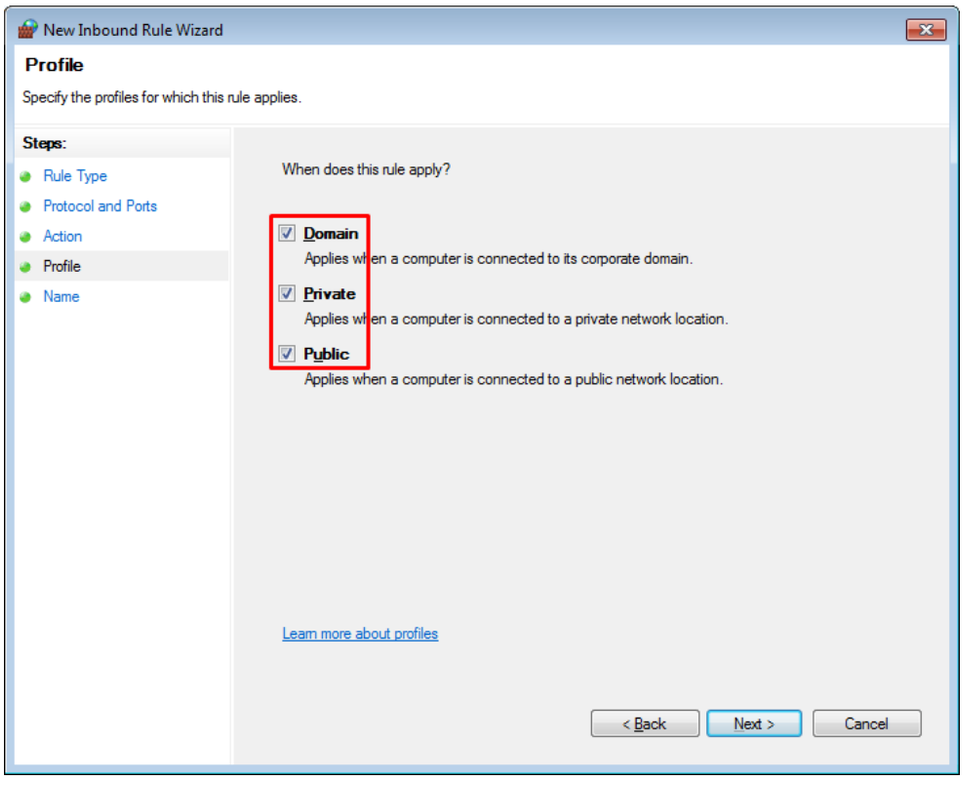
Select the “TCP” to be applied to the new rule and set the port as “Specific local ports”. Then, enter “135-139”and click the [Next] button.



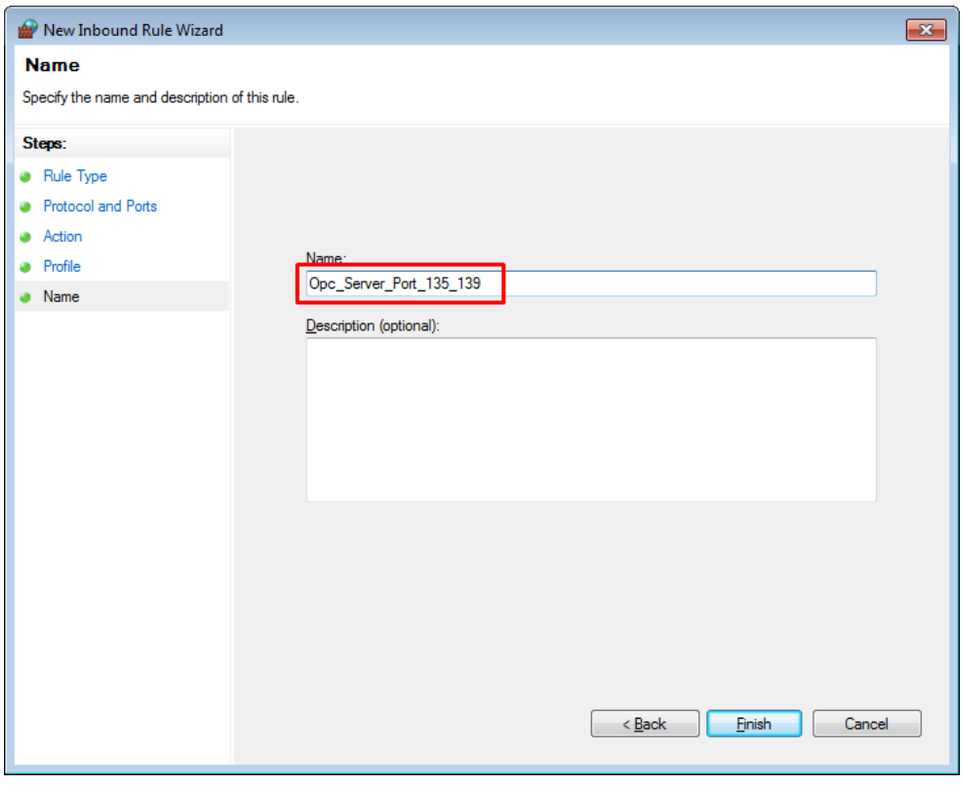
Select “Allow the connection” for the specified action items and click the [Next] button.



Put a check “Domain”, “Private”, “Public” in the profile items and click the [Next] button.



Enter the name and description of the newly defined rule.



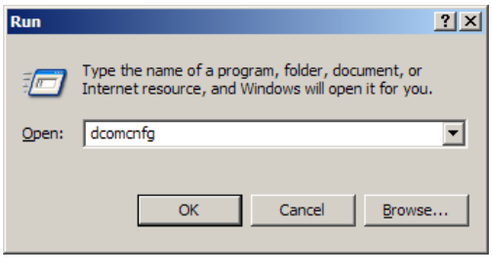
3.2 Windows XP Configuration

3.2.1 Client Settings

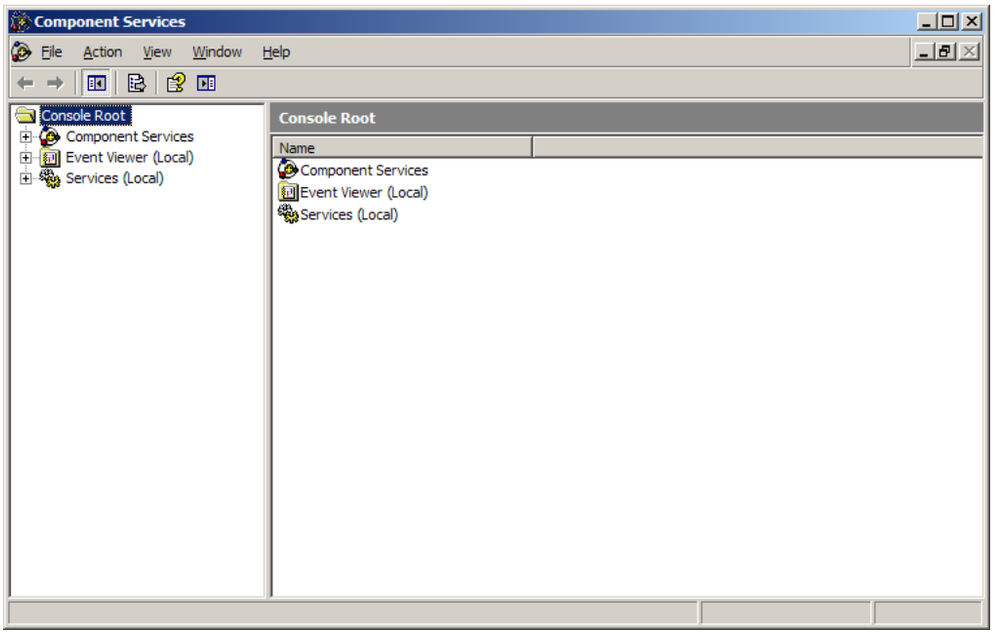
(1) Default

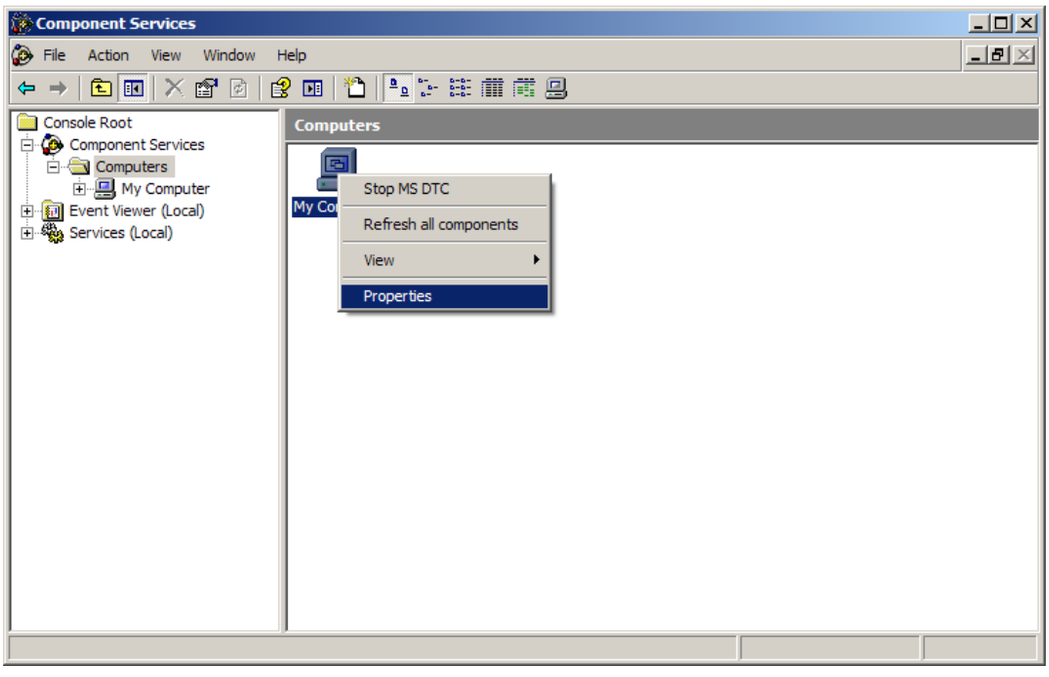
- The user needs to set up DOM to use the OPC servers installed on the remote PC.
- DCOM is set up as the following procedure and the local PC and the remote PC are also applied the same settings as DCOM.
- The following description is about DCOM settings in the window XP and for the DCOM settings in other OS, any similar type is used.
- The same account shall be registered for both of the local PC and remote PC and the same password shall be used for both of them.
- The local PC and remote PC shall be accessed with the same account.
- The firewall in the remote PC shall be cancelled to log in to the system.
- To use the OPC servers installed in the remote PC, basic OPC files shall be installed in the local PC. (Install the OPC Core Components Redistributable program first. Visit www.opcfoundation.org)

(2) Component Service Settings



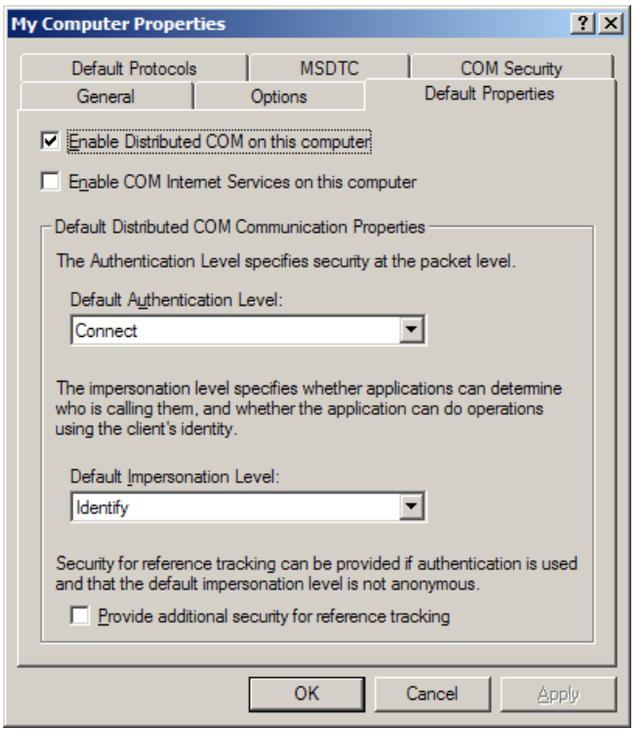
- Input “dcomcnfg” in the window and press OK.
- The following “Component Services” window will appear.





● Find “My Computer” and execute “Properties” menu.

● The following “My Computer Properties” window will appear.

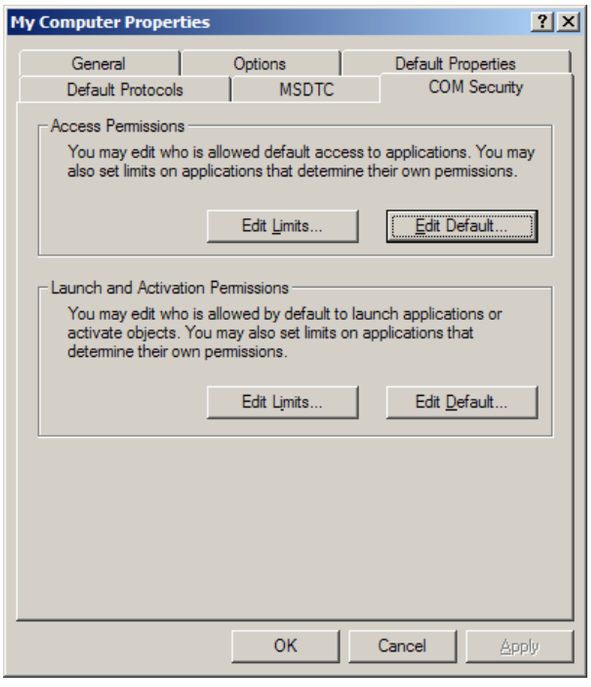


● Select “Default Properties”tap.

● Check √ in the checkbox of “Enable Distributed COM on this computer”.

● Select “Connect” for “Default Authentication Level” item.

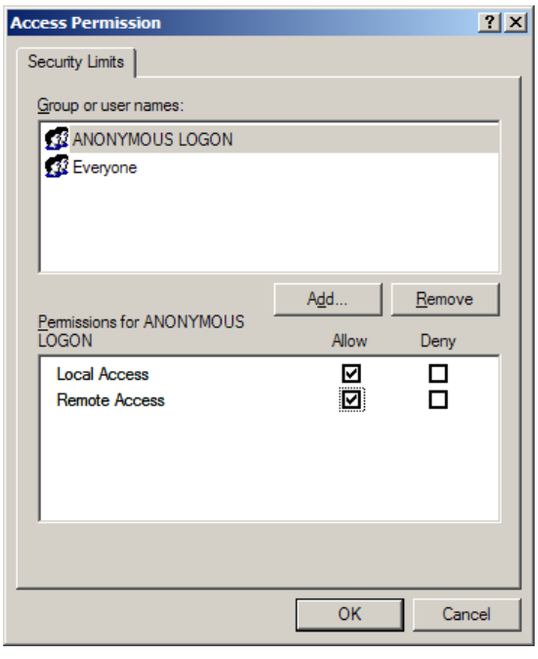
● Select “ID” for “Default Impersonation Level” item.



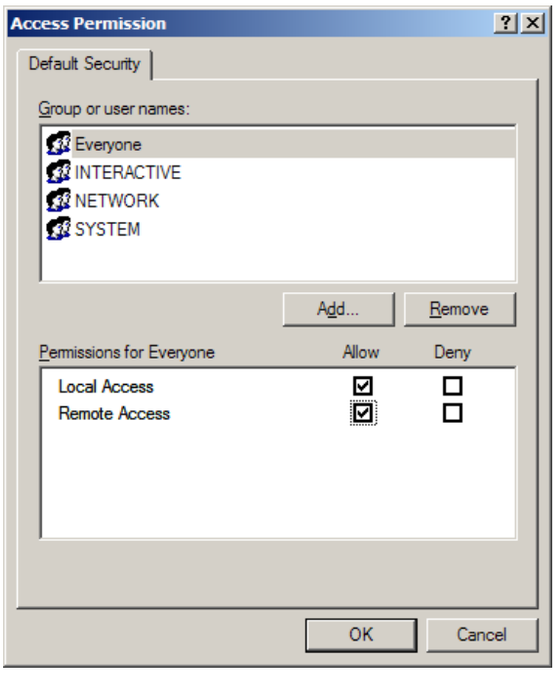
● Select"COM Security" tap.

● To set up "Access Permissions", press "Edit Limits" and "Edit Default" to set up as follow:

- Edit Limits: Check √ in both of the boxes under "Allow" for "Local Access" and "Remote Access" to permit ANONYMOUS LOGON and Authenticated Users.

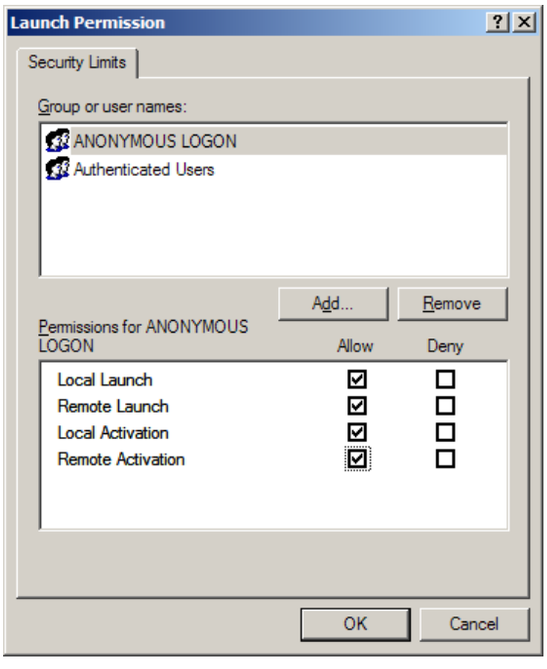


● Edit Default: - Edit Limits; Check ¡Ìin both of the boxes under “Allow” for “Local Access” and “Remote Access” to permit INTERACTIVE, NETWORK and SYSTEM users to log in.

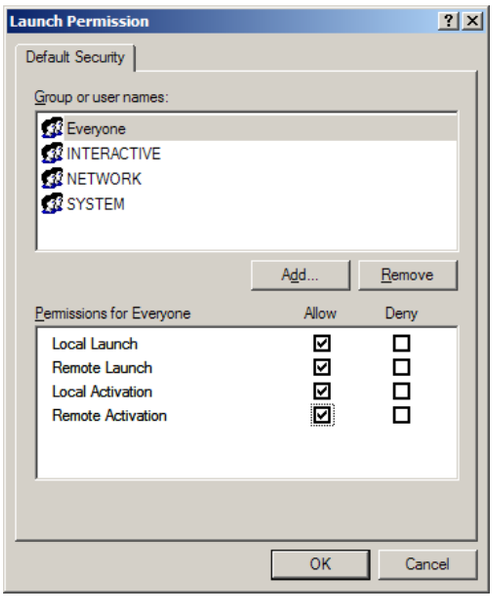


● To set up “authorization to launch and activate”, press “Edit Default” and set up as follow:

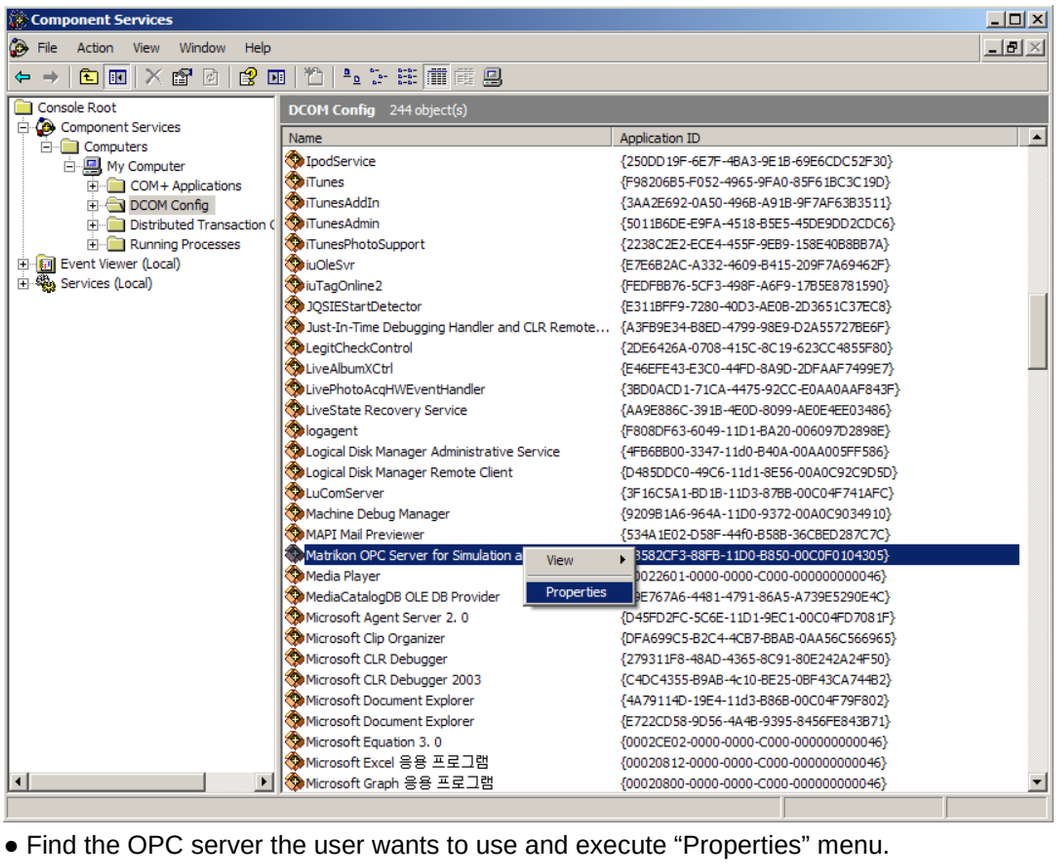
- Edit Limits: Check √ in all of the boxes under “Allow” for “Local Launch”, “Remote Launch”, “Local Activation” and “Remote Local Activation” to permit Authenticated Users to launch and activate.



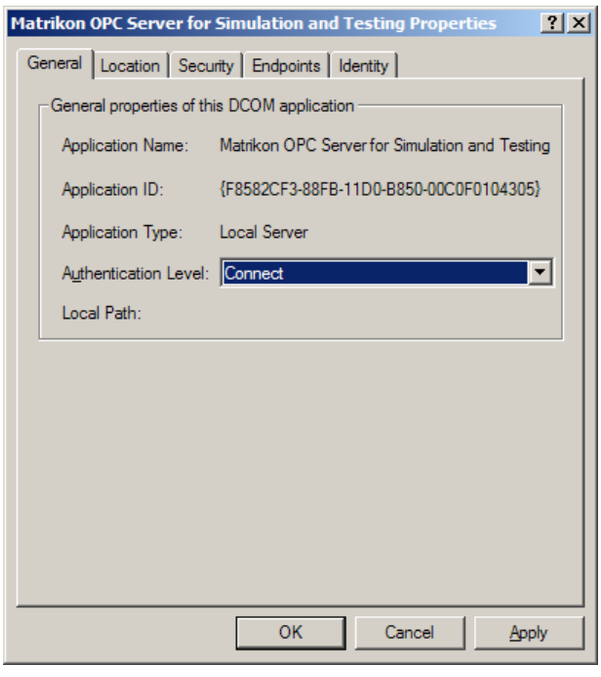
- Edit Default: Check √ in all of the boxes under" Allow" for "Local Launch", "Remote Launch","Local Activation" and "Remote Local Activation" to permit INTERACTIVE, NETWORK and SYSTEM users to launch and activate.



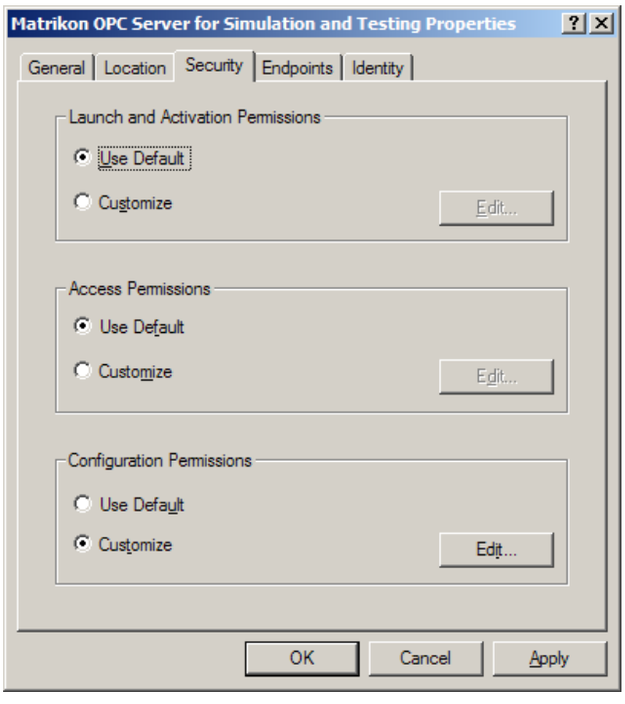
- In the next step, DOM settings are carried out on the OPC server to be used.
- The user needs to set up the OPC server he/she wants to use according to the following procedure:



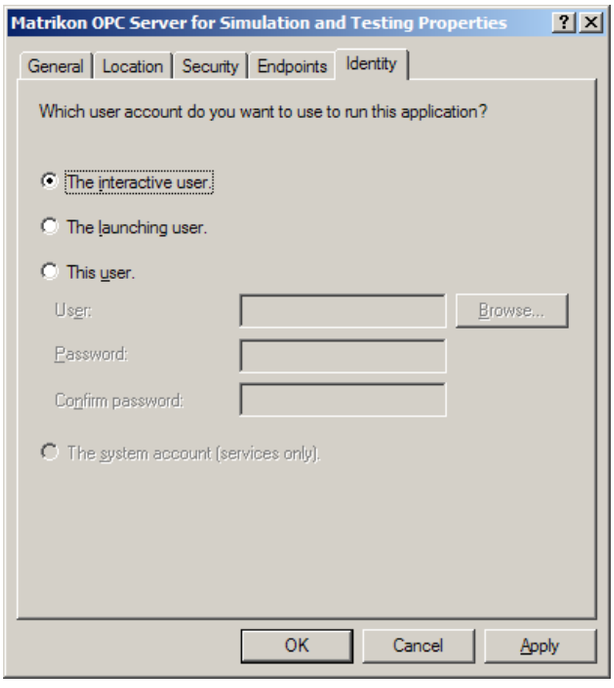
- Find the OPC server the user wants to use and execute "Properties" menu.



- Select “Connect” for “Authentication Level” from “General” tap.

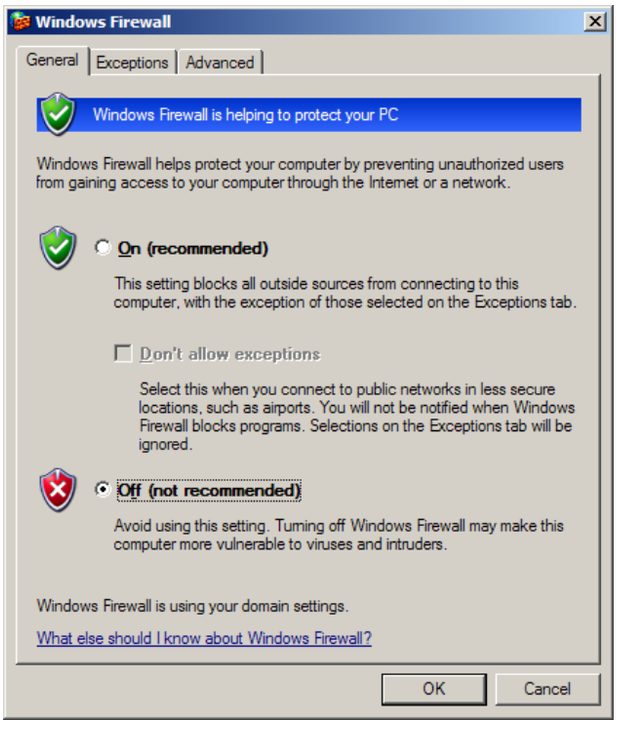


- Select “Security” tap.
- Select “Use Default” for “Launch and Activation Permissions.”
- Select “Use Default” for “Access Permissions.”



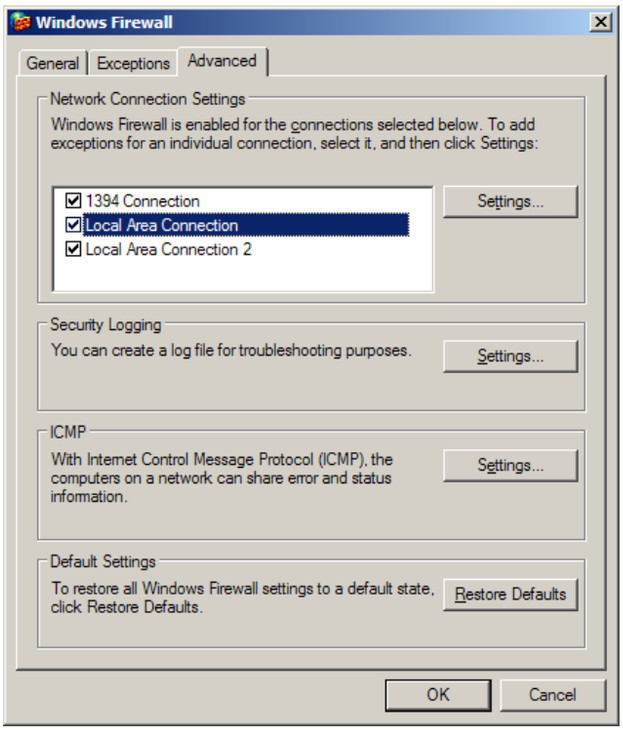
- Select “Identity” tap.
- Select “The Interactive User.”

(3) Firewall Settings

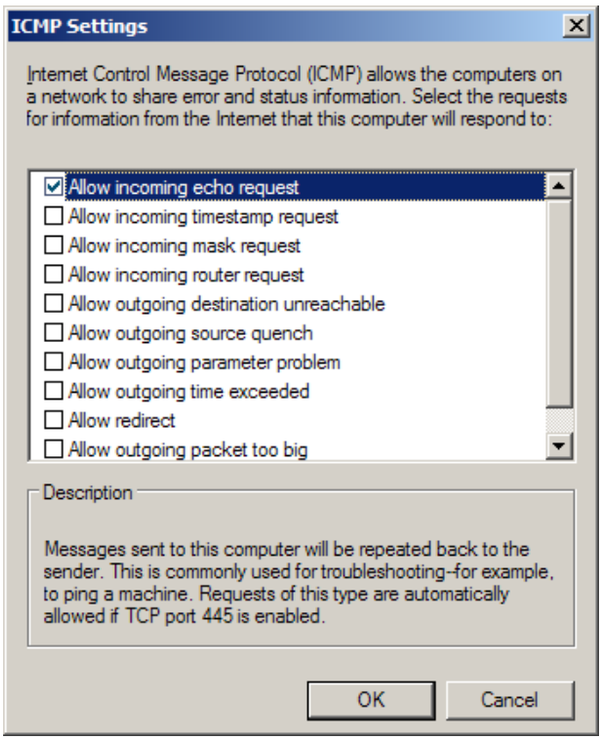


- The above Windows Firewall window is displayed.
- Select “Off (not recommended)”.

● Select “Advanced” tap.



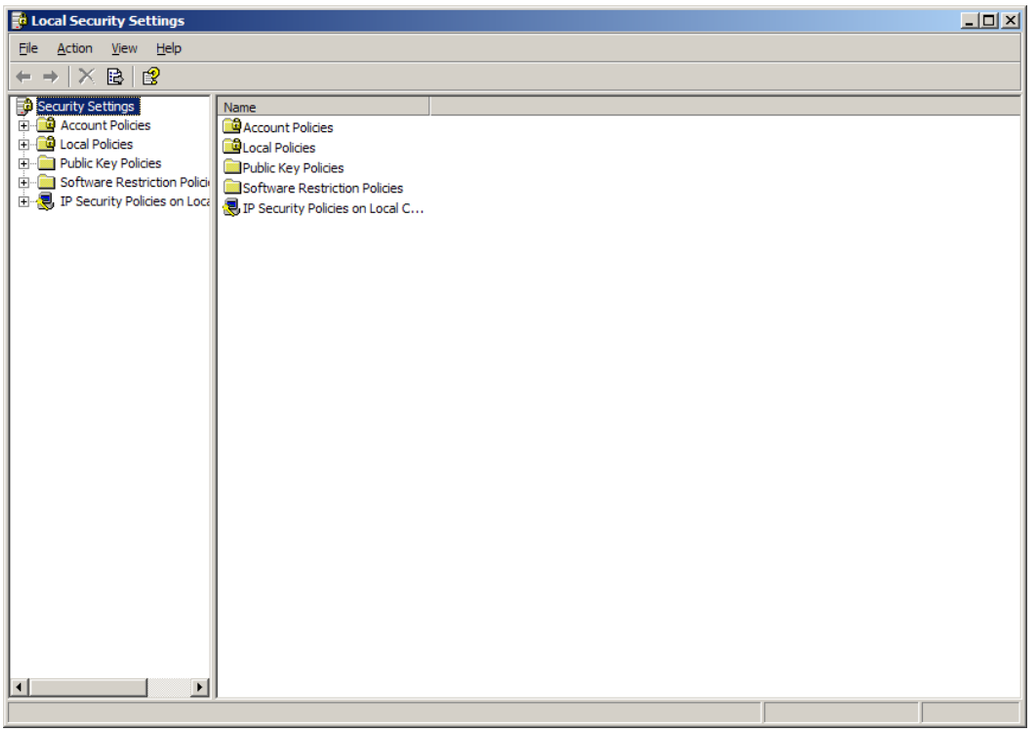
● Click “Settings” button for ICMP.



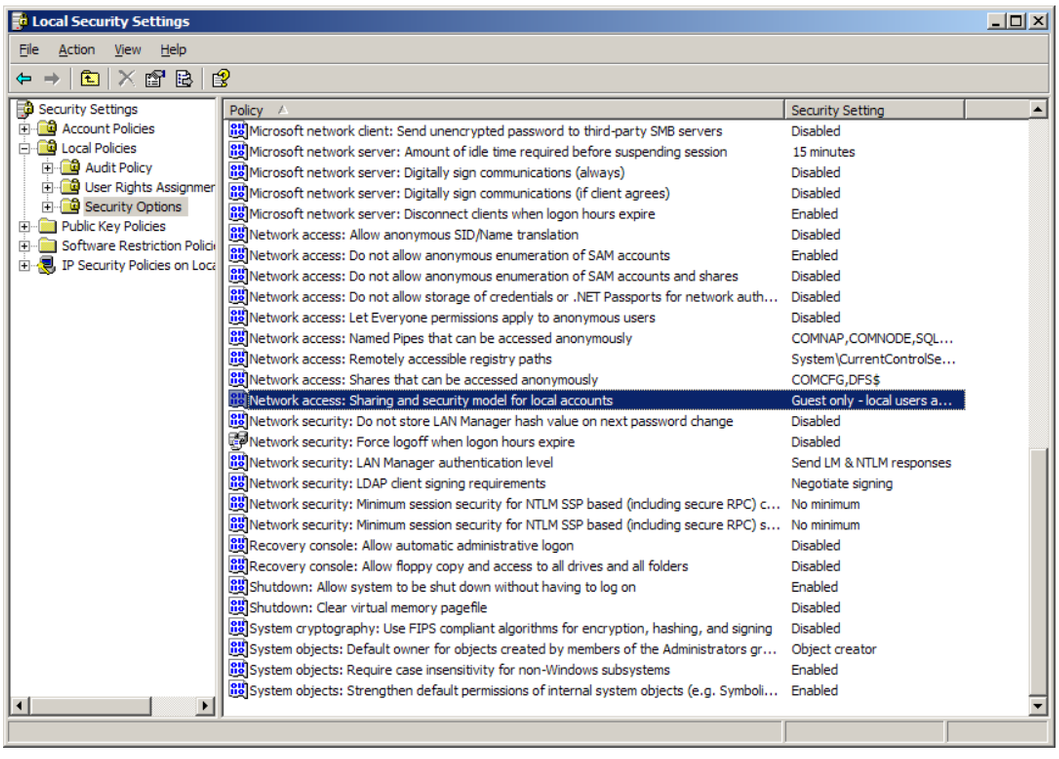
● Check “Allow incoming echo request”.

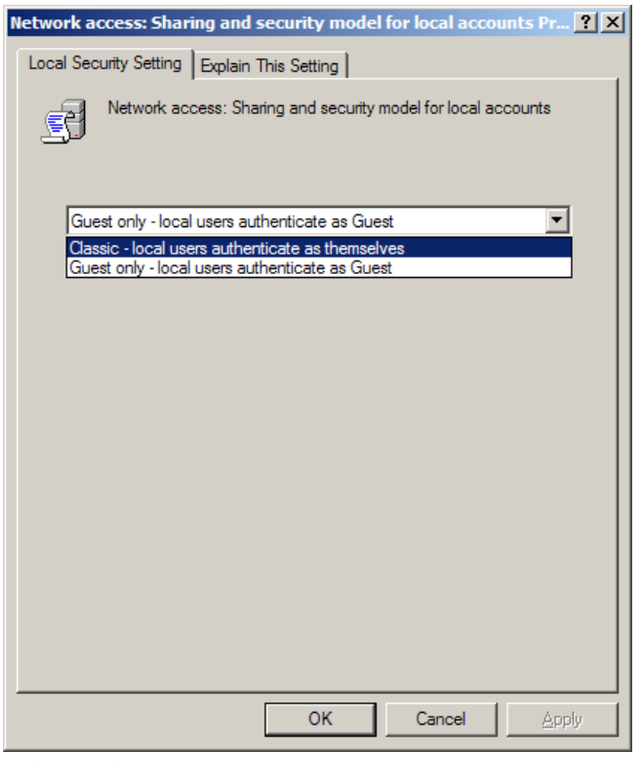
(4) Security Settings

● Execute [Control Board] → [Management Tool] → [Local Policies].



● [Local Policies] → [Security Options] → "Network Access: Execute "Sharing and security model for local accounts".





● Select“Classic – Local users authenticate as themselves”.

3.2.2 Server Settings

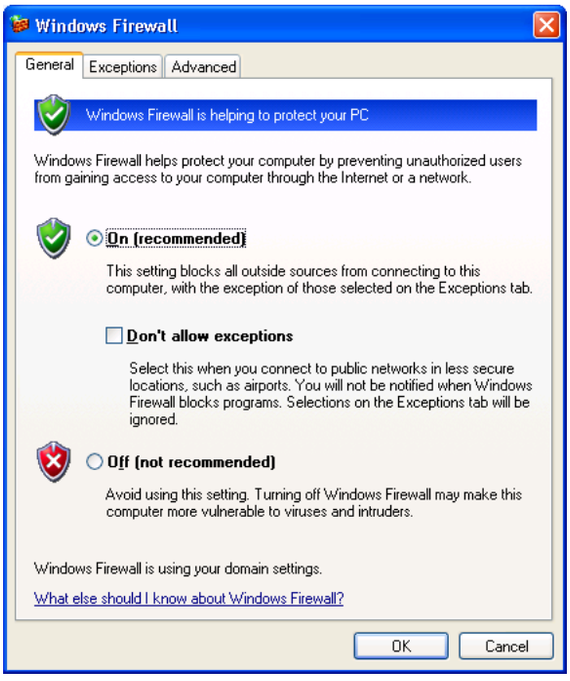
DCOM configuration is necessary to connect to OPC server on the remote computer. For DCOM configuration, there are computer configuration and LS InfoU OPC DA Server configuration. In this appendix, configuration is explained with an example of DCOM configuration on the Window XP service Pack 2 and similar procedure applies to other OSs, too.

Basic Procedure)

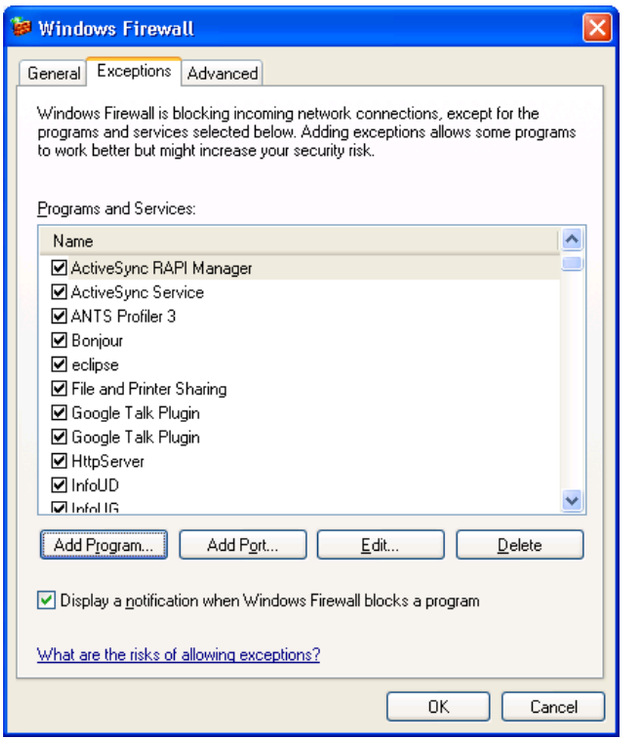
- The local computer and the remote computer should be registered with the same account, sharing the same password.
- The local computer and the remote computer should be logged in with the same account
- To use OPC DA Server on the remote computer, basic OPC files have been installed on the local computer.

(1) Firewall Setting

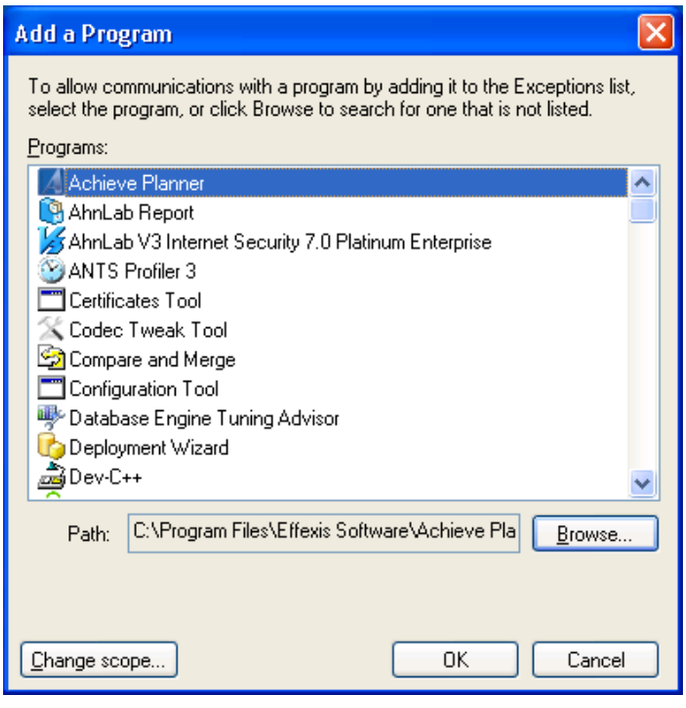
- Select“Start → Control Board → Security Center → Window Firewall”. “On (recommended)”has been set up as the default configuration. This configuration is recommended to protect the computer from Microsoft and OPC. However, select“Not Use” for OPC communication or follow the instruction below:

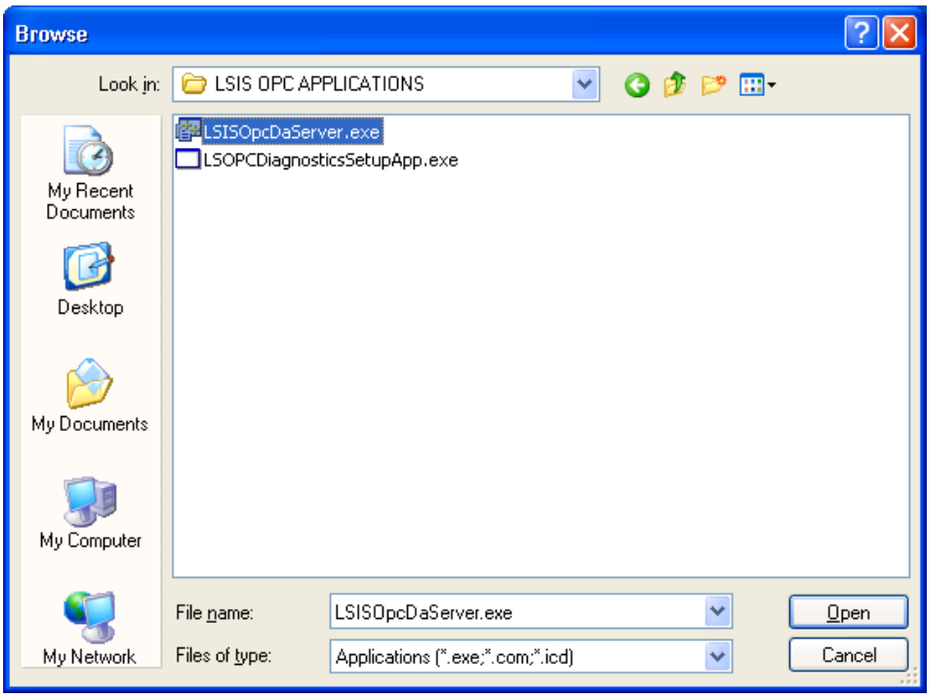


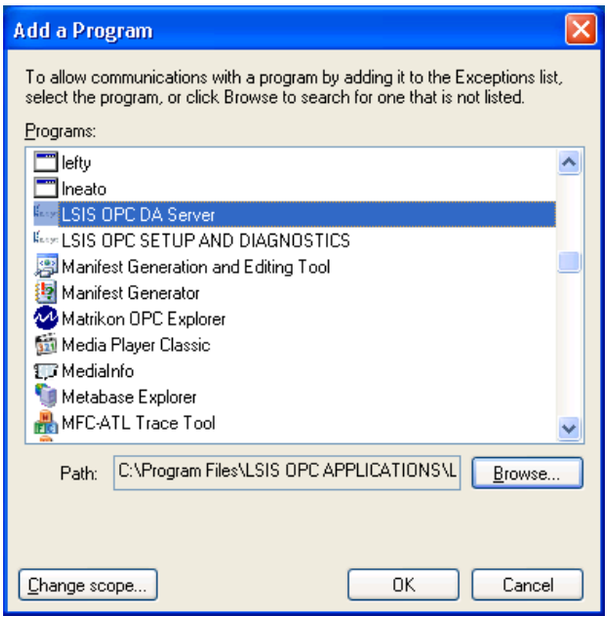
● Select “Exceptions” tap and press “Add Program” as below.



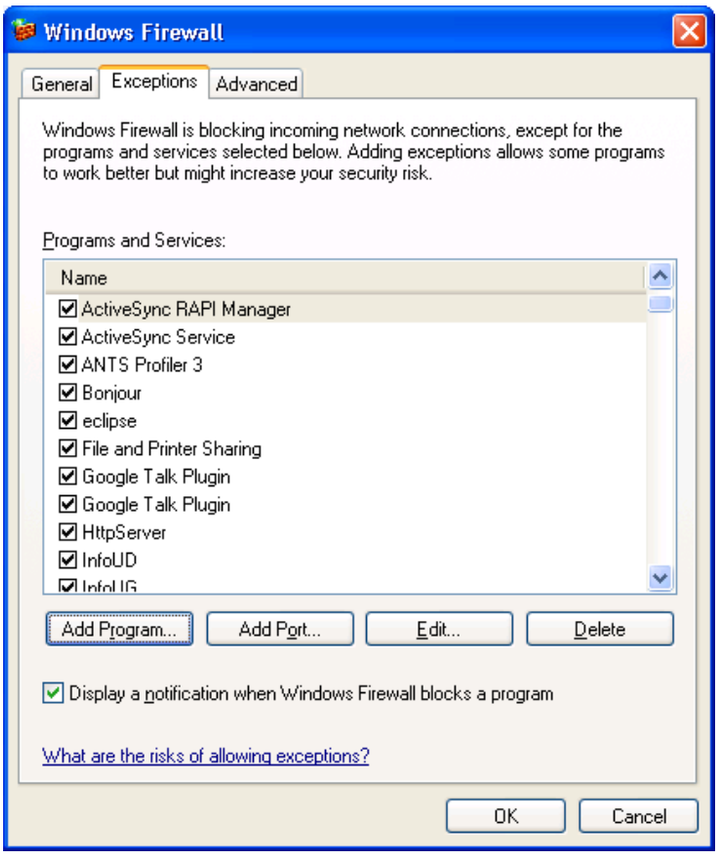
● Select “Browse” button to register LSISOpcDaServer.exe and LSOPCDiagnosticsSetuipApp.exe the runtime and the configuration program of LSIS InfoU OPC DA Server respectively.




Browse
Look in:
LSIS OPC APPLICATIONS
LSISOpcDaServer.exe
LSOPCDiagnosticsSetupApp.exe
My Recent Documents
Desktop
My Documents
My Computer
My Network
File name:
LSISOpcDaServer.exe
Files of type:
Applications (*.exe;*.com;*.icd)
Open
Cancel

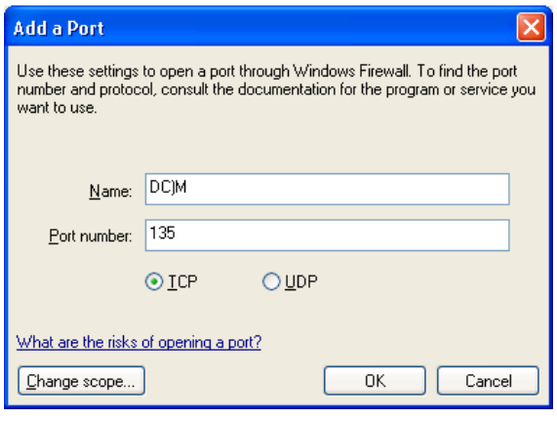

Add a Program
To allow communications with a program by adding it to the Exceptions list, select the program, or click Browse to search for one that is not listed.
Programs:
lefty
lneato
LSIS OPC DA Server
LSIS OPC SETUP AND DIAGNOSTICS
Manifest Generation and Editing Tool
Manifest Generator
Matrikon OPC Explorer
Media Player Classic
MediaInfo
Metabase Explorer
MFC-ATL Trace Tool
Path:
C:\Program Files\LSIS OPC APPLICATIONS\L
Browse...
Change scope...
OK
Cancel

● Add TCP Port 135 to initialize DOCM communication. Click on “Add Port” button on “Exceptions” tap of Windows Firewall.



● Input the following items on “Add a Port” window.

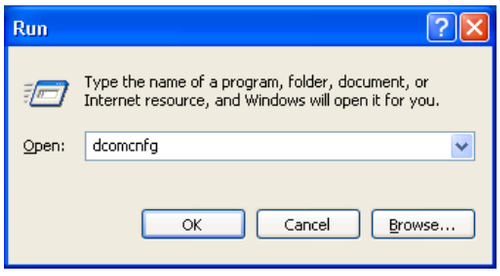
- Name: DCOM
- Port Number: 135
- “TCP” Select radio button



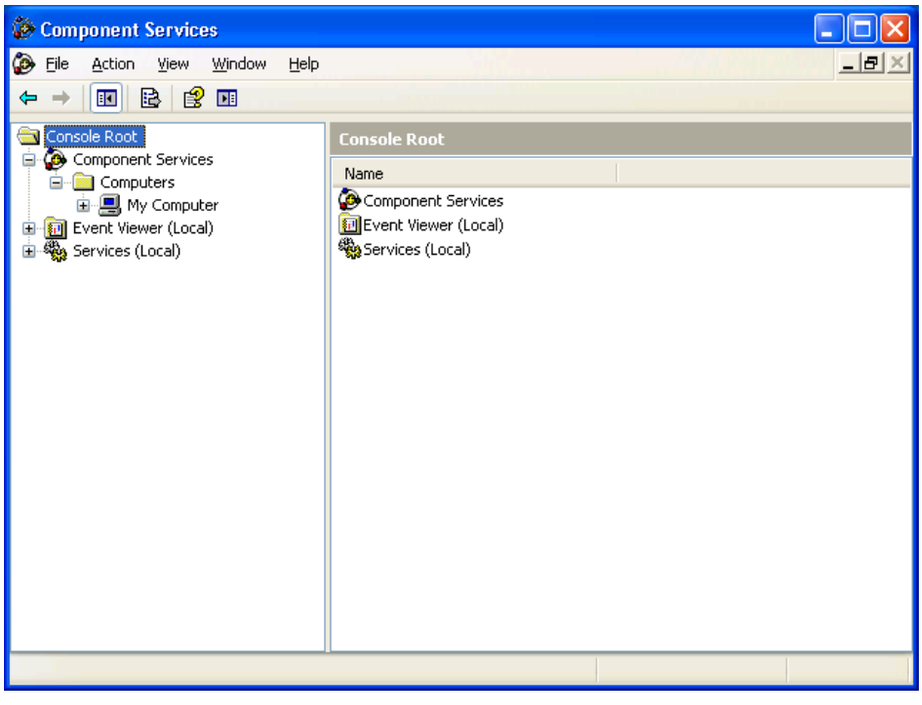
(2) DCOM Setting

1) Computer Setting

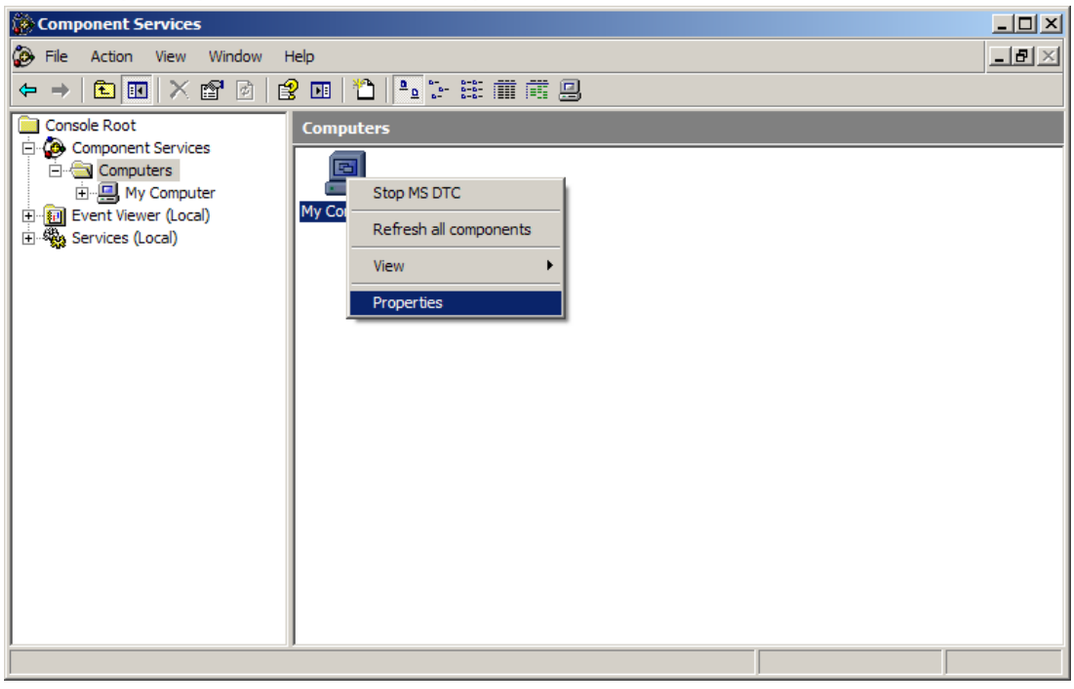
● Input [Start] → [Run] → “DCOMCnfg” and then, press “OK” button.



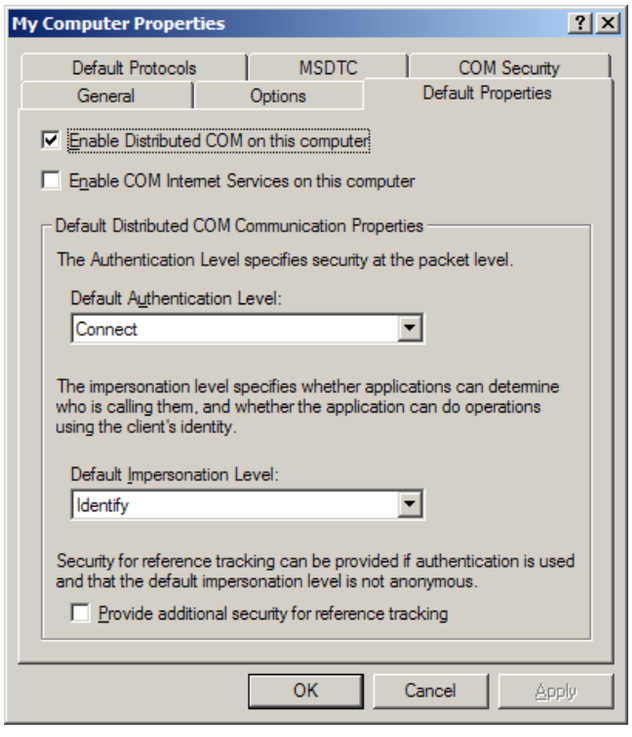
● “Component Services” window is displayed as below. Click on Console Root to enlarge the Component Services and then, click on My Computer.



- Find ”My Computer” item and run ”Properties” menu.



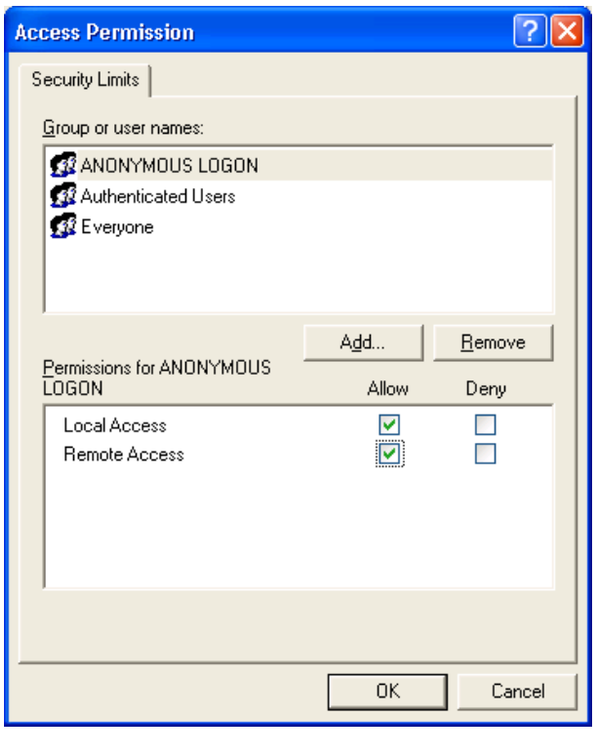
- Set up designated values for the following items on “Default Properties” tap as below.
  - "Enable Distributed COM on this computer”: Check “Use.”
  - Default Authentication Level: Connect.
  - Default Impersonation Level: ID

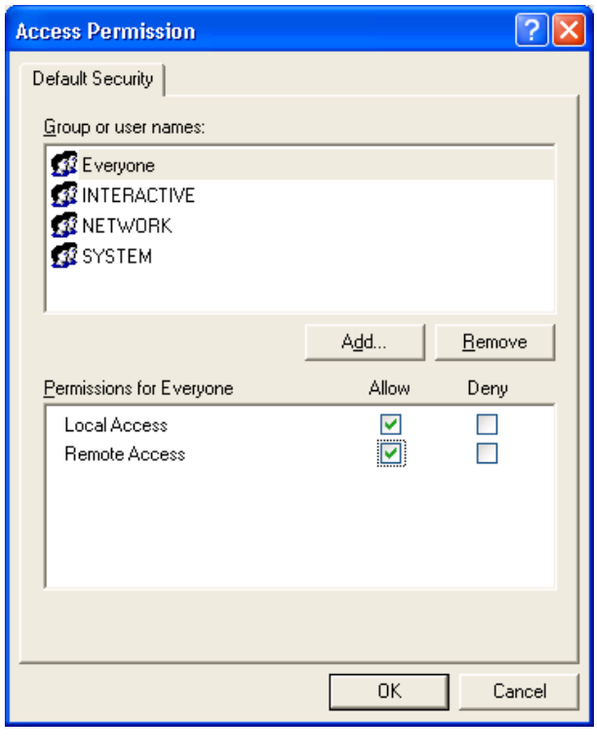


- After selecting “COM Security” tap, set up “Access Permission” and “Launch Permission” as below.

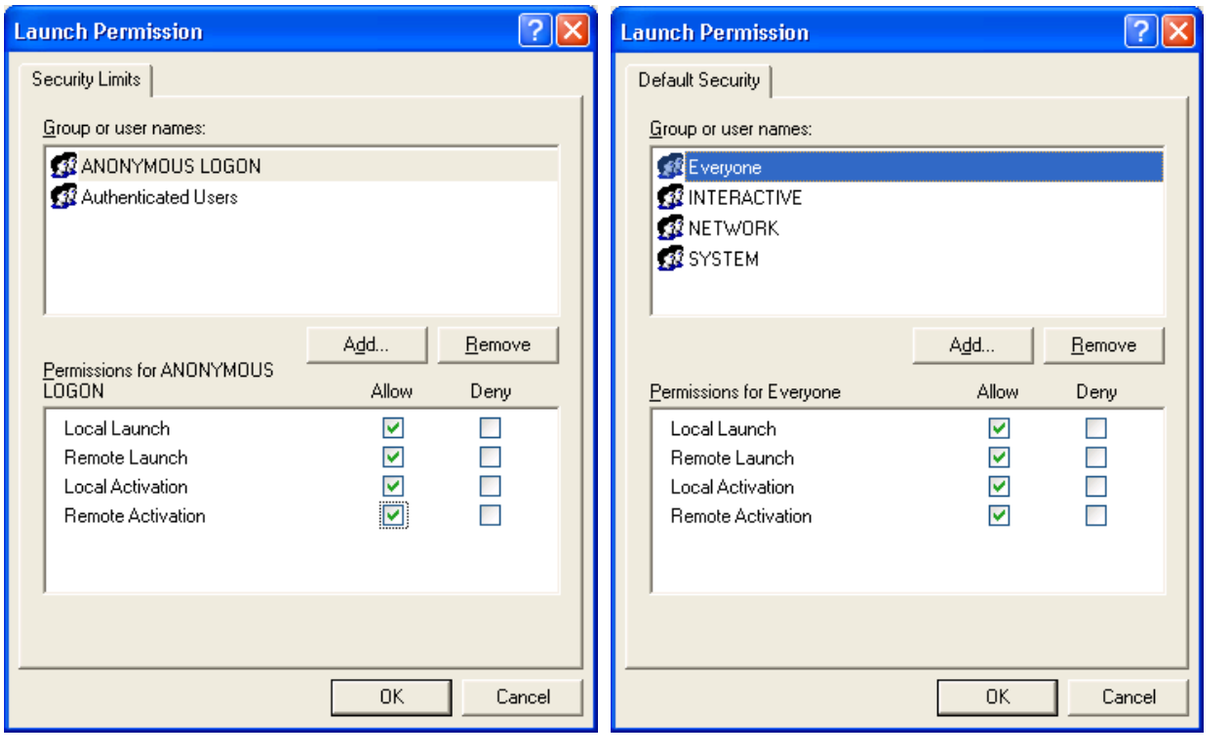
● For “Access Permission,” set up “Security Limits” and “Default Security.”

- Security Limits: After adding ANONYMOUS LOGIN, Everyone and Authenticated Users to ‘Group or User Names’ item, Check √ in all of the boxes under “Allow” for “Local Assess” and “Remote Access”.
- Default Security: Check √ in all of the boxes under “Allow” for “Local Access” and “Remote Access” for Everyone, INTERACTIVE, NETWORK and SYSTEM user.



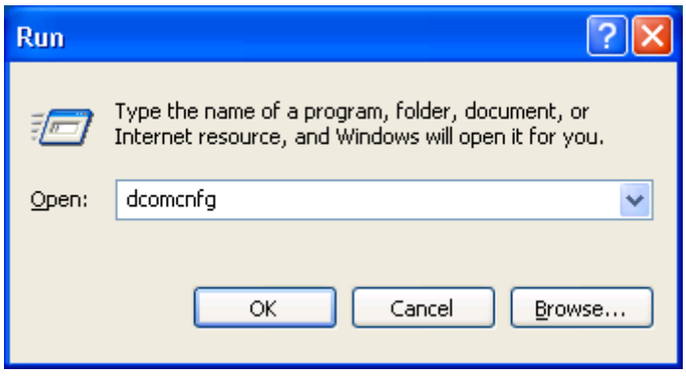


- For “Launch Permission,” set up “Security Limits” and “Default Security” as below.
  - Security Limits: Check √ in all of the boxes under “Allow” for “Local Launch”, “Remote Launch”, “Local Launch” and “Remote Launch” for ANONYMOUS LOGON and Authenticated Users.
  - Default Security: Check √ in all of the boxes under “Allow” for “Local Launch”, “Remote Launch”,“Local Launch” and “Remote Launch” for Everyone, INTERACTIVE, NETWORK and SYSTEM user.



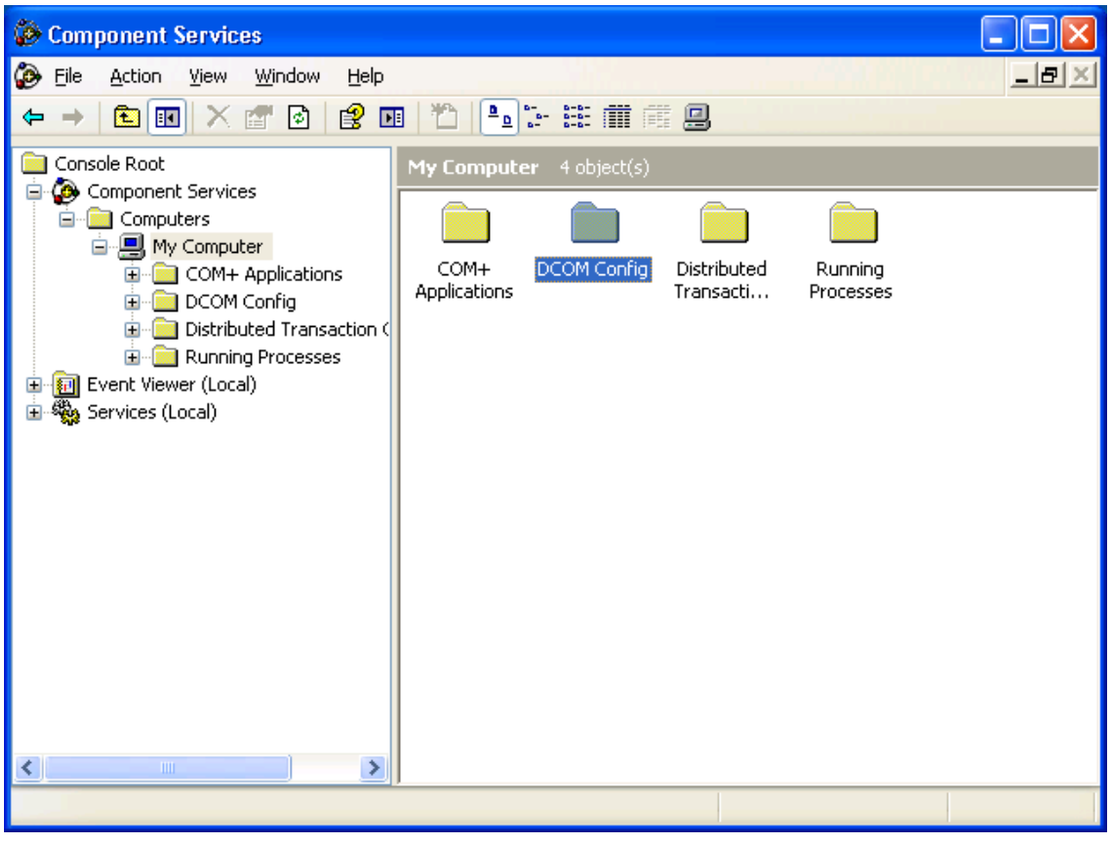
2) LSIS InfoU OPC Server Setting

● After entering [Start] → [Run] → “DCOMCnfg”, press “OK” button.

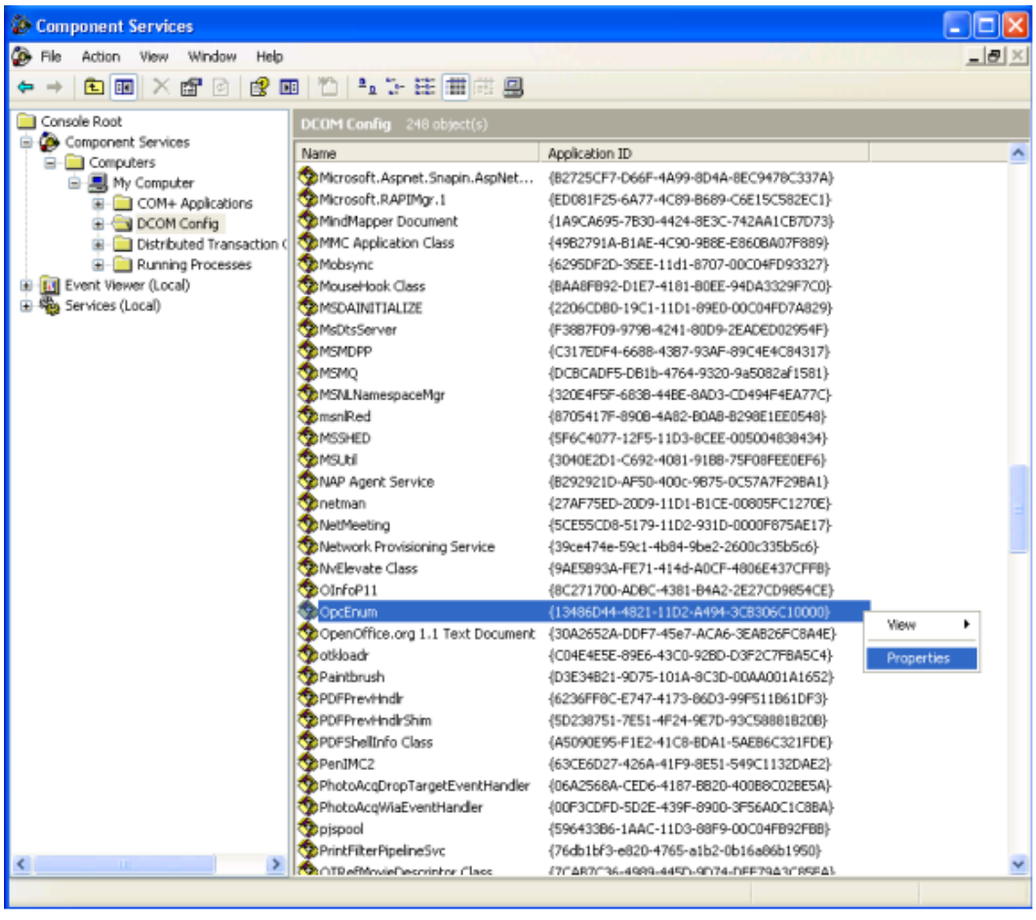


● “Component Services” window is displayed as below. Click on Console Root to enlarge the Component Services and then, click on Computers. Find My Computer and set up properties.

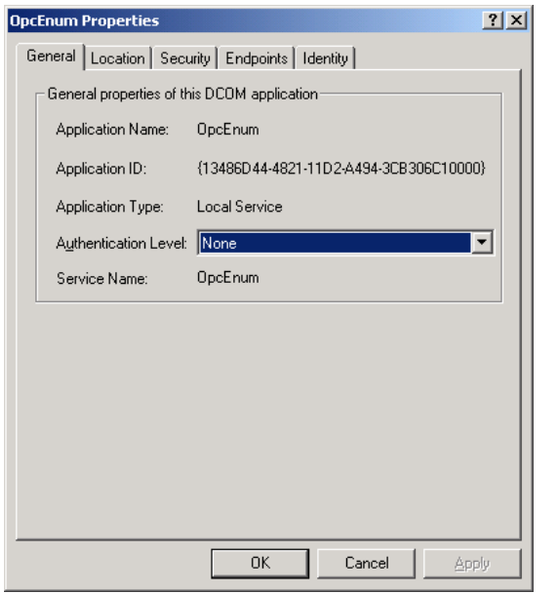
● Double click on “DCOM Config.”



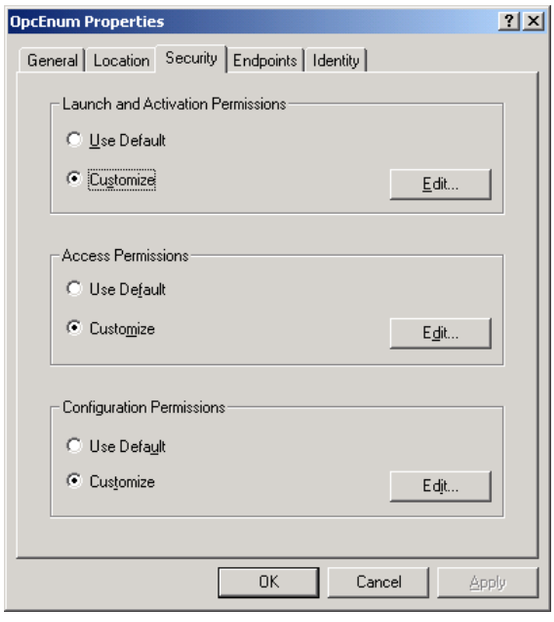
- Set up properties on “OpcEnum” to permit ANONYMOUS LOGON. (If “OpcEnum”is not used, Remote Access by any anonymous user is not permitted.)
- Find “OpcEnum” from the list and run “Properties” menu.



- Set up “None” for ”Authentication Level” on ”General” tap of “OpcEnum Property” Window.

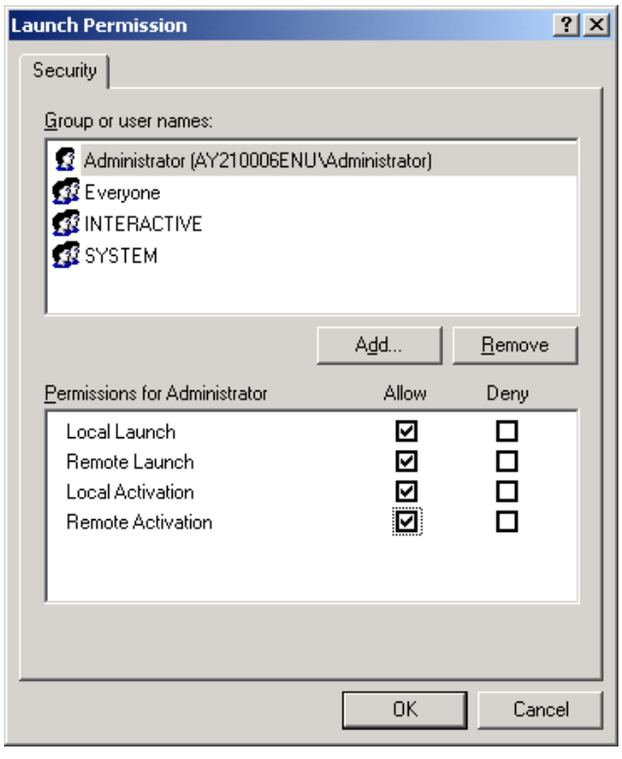


● Select “Security” tap.

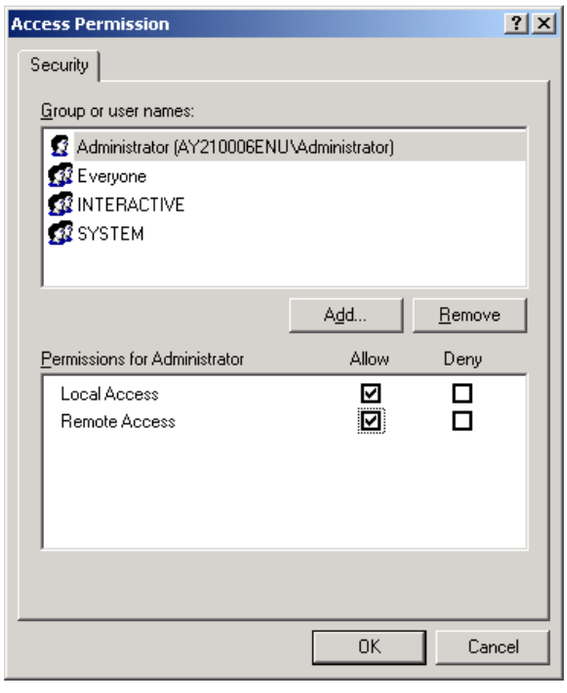


● After selecting “User Specified” for ”Launch Permission” and pressing “Edit” button, set up as below.

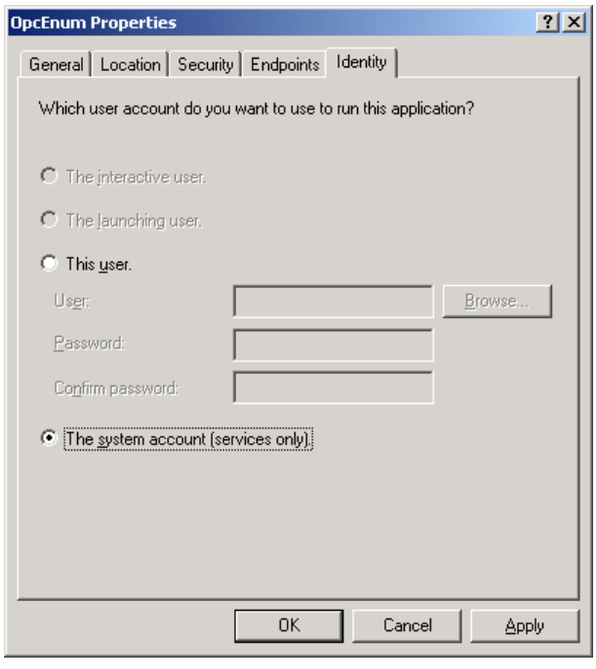
- “Group or User Names” : Check √ in all of the boxes under “Allow” for “Local Launch”, “Remote Launch”, “Local Launch” and “Remote Launch” for each user.



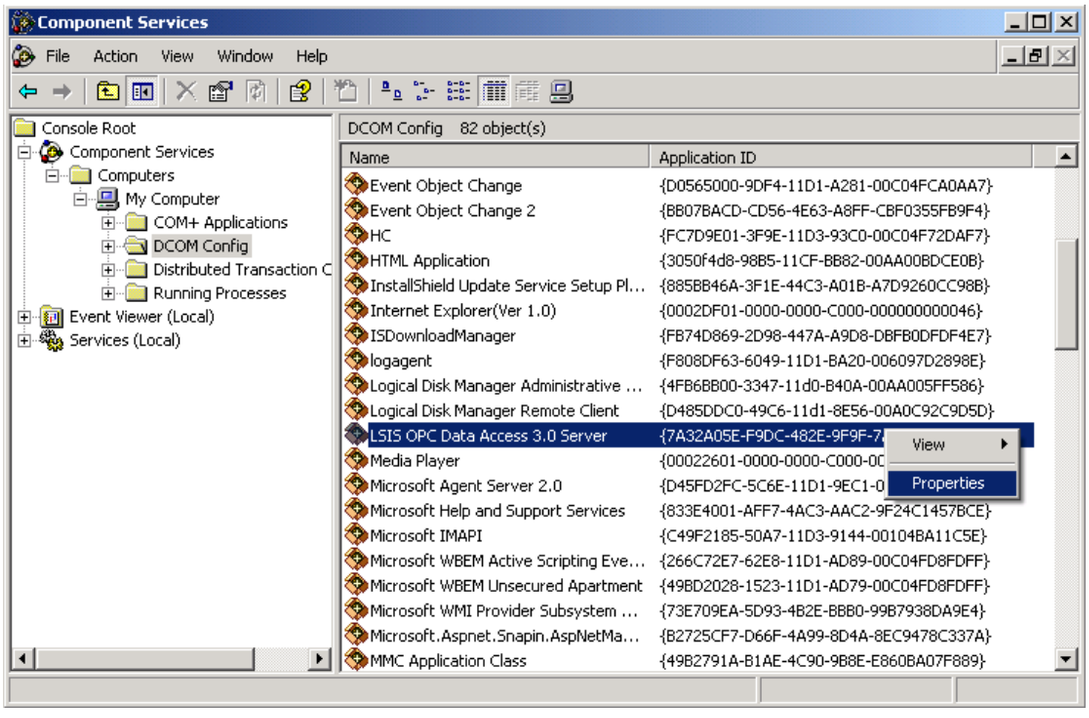
- After setting up “Access Permission” and pressing “Edit” button, set up as below.
  - “Group or User Names”: Check √ in all of the boxes under “Allow” for “Local Access” and “Remote Access” for each user.



- After clicking on “ID” tap, select “The system account (services only)”.

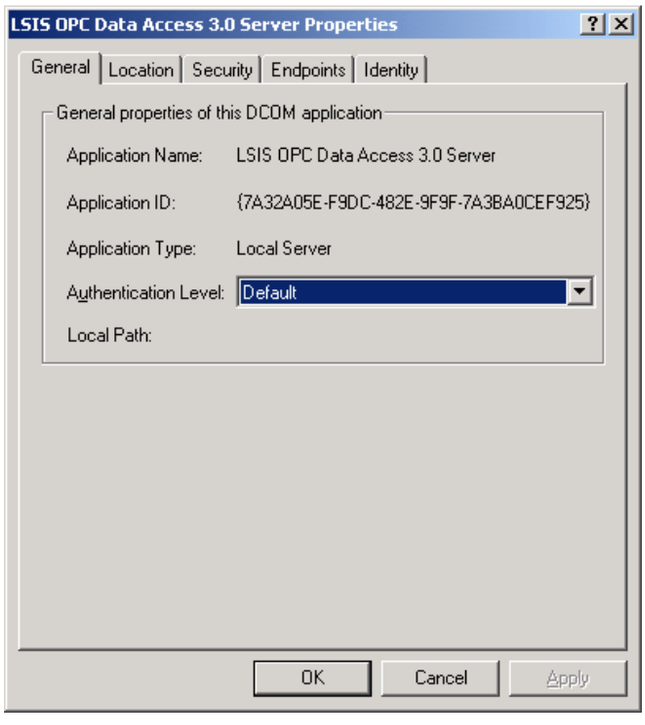


● Find" LSIS OPC Data Access 3.0 Server" from the list and run "Properties" menu.

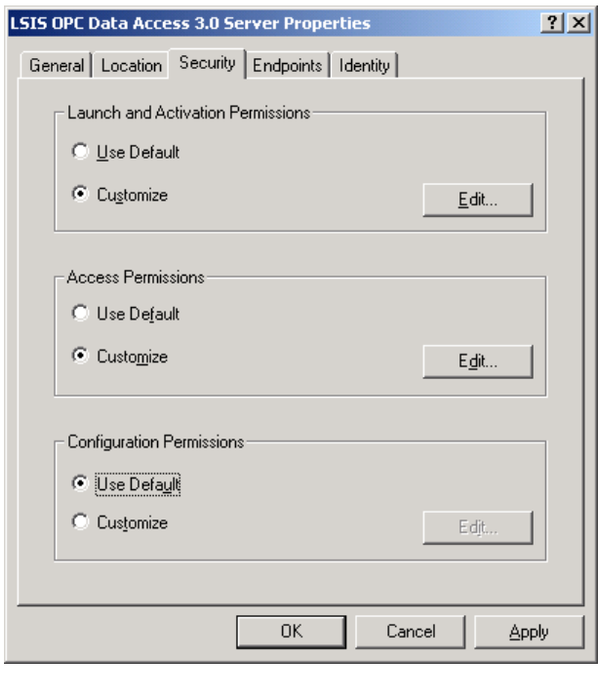


● "LSIS OPC Data Access 3.0 Server Property" Window is displayed.

● Maintain "Default" for "Authentication Level" item on "General" tap.

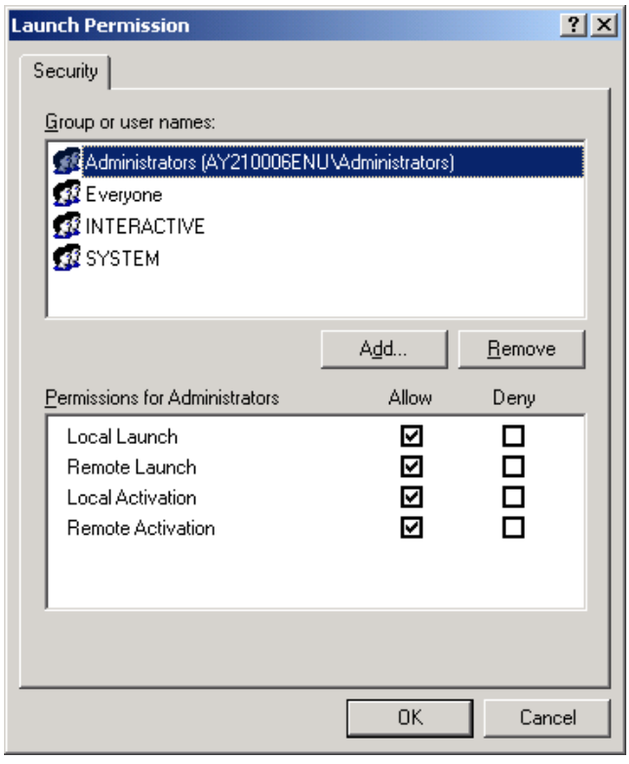


● Select “Security” tap.



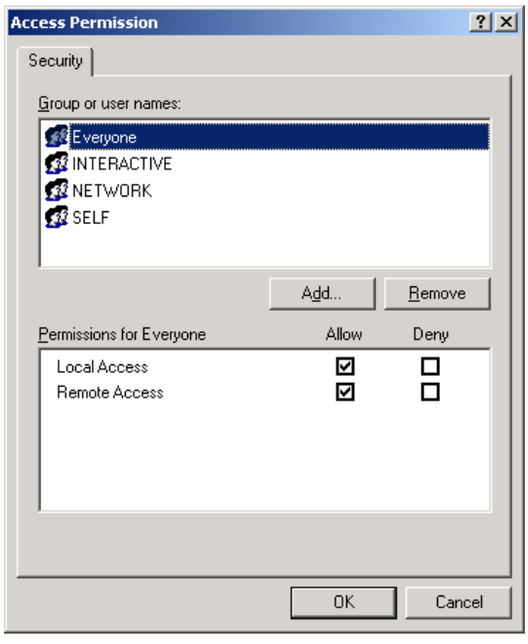
● After selecting “User Specified” for” Launch Permission” and pressing “Edit” button, set up as below.

- “Group or User Names”: Check √ in all of the boxes under “Allow” for “Local Launch”, “Remote Launch”, “Local Activation” and “Remote Activation” for each user.



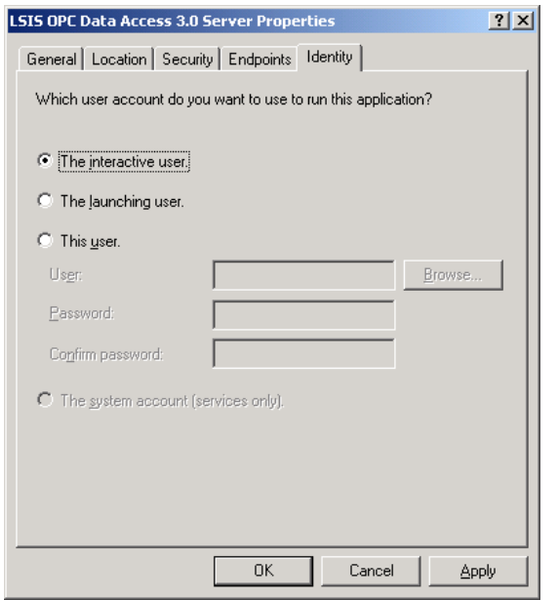
● After setting up “Access Permission” and pressing “Edit” button, set up as below.

- “Group or User Names”: Check √ in all of the boxes under “Allow” for “Local Access” and “Remote Access” for each user.



● Set up “Use Default” for “Configuration Authority”.

● After selecting “Identity” tap, select “The interactive user”.



● After clicking on “Apply” button, press “OK” button.

Appendix 4 Visual Basic Script Help

4.1 Script Function of Microsoft Script Engine

Function Name	Form
Abs	Function Abs(Number)
Array	Function Array(ArgList)
Asc	Function Asc(String As String)
Atn	Function Atn(Number As Double) As Double
CBool	Function CBool(Expression) As Double
CByte	Function CByte(Expression) As Byte
CCur	Function CCur(Expression) As Currency
CDate	Function CDate(Expression) As Date
CDbl	Function CDbl(Expression) As Double
Chr	Function Chr(CharCode As Long
CInt	Function CInt(Expression) As Integer
CLng	Function CLng(Expression) As Long
Cos	Function Cos(Number As Double) As Double
CreateObject	Function CreateObject(Class As String)
CSng	Function CSng(Expression) As Single
CStr	Function CStr(Expression) As String
Date	Function Date() As Date
DateAdd	Function DateAdd(Interval As String, Number As Double, Date)
DateDiff	Function DateDiff(Interval As String, Date1, Date2, [FirstDayOfWeek As VbDayOfWeek = vbSunday], [FirstWeekOfYear As VbFirstWeekOfYear = vbFirstJan1])
DatePart	Function DatePart(Interval As String, Date, [FirstDayOfWeek As VbDayOfWeek = vbSunday], [FirstWeekOfYear As VbFirstWeekOfYear = vbFirstJan1])
DateSerial	Function DateSerial(Year As Integer, Month As Integer, Day As Integer)
DateValue	Function DateValue(Date As String)
Day	Function Day(Date)
Eval	Function Eval(Expression As String)
Exp	Function Exp(Number As Double) As Double
Filter	Function Filter(InputStrings As Array, Value As String [, Include As Boolean [, Compare As VbCompareMethod = vbBinaryCompare]]])
Fix	Function Int(Number)
FormatCurrency	Function FormatCurrency(Expression[,NumDigitsAfterDecimal [, [,UseParensForNegativeNumbers [,GroupDigits]]]])
FormatDateTime	Function FormatDateTime(Date[, NamedFormat])

FormatNumber	Function FormatNumber(Expression [,NumDigitsAfterDecimal [,IncludeLeadingDigit [,UseParensForNegativeNumbers [,GroupDigits]]]])
FormatPercent	Function FormatPercent(Expression[,NumDigitsAfterDecimal [,IncludeLeadingDigit [,UseParensForNegativeNumbers [,GroupDigits]]]])
GetObject	Function GetObject([PathName], [Class])
Hex	Function Hex(Number)
Hour	Function Hour(Time)
InputBox	Function InputBox(Prompt, [Title], [Default], [XPos], [YPos], [HelpFile], [Context]) As String
InStr	Function InStr([Start], [String1], [String2], [Compare As VbCompareMethod = vbBinaryCompare])
InStrRev	Function InStrRev([Start], [String1], [String2], [Compare As VbCompareMethod = vbBinaryCompare])
Int	Function Int(Number)
IsArray	Function IsArray(VarName) As Boolean
IsDate	Function IsDate(Expression) As Boolean
IsEmpty	Function IsEmpty(Expression) As Boolean
IsNull	Function IsNull(Expression) As Boolean
IsNumeric	Function IsNumeric(Expression) As Boolean
IsObject	Function IsObject(Expression) As Boolean
Join	Function Join(List As Array[, Delimiter As String])
LBound	Function LBound(Arrayname[, Dimension As Number])
LCase	Function LCase(String)
Left	Function Left(String, Length As Long)
Len	Function Len(Expression)
LoadPicture	Function LoadPicture([filename], [widthDesired As Long], [heightDesired As Long], [flags As LoadPictureConstants = Default]) As IpictureDisp
Log	Function Log(Number As Double) As Double
LTrim	Function LTrim(String)
Mid	Function Mid(String, Start As Long, [Length])
Minute	Function Minute(Time)
Month	Function Month(Date)
MonthName	Function MonthName(Month As Number[, Abbreviate As Boolean])
MsgBox	Function MsgBox(Prompt, [Buttons As VbMsgBoxStyle = vbOKOnly], [Title], [HelpFile], [Context]) As VbMsgBoxResult
Now	Function Now()
Oct	Function Oct(Number)
Replace	Function Replace(Expression As String, Find As String, ReplaceWith As String [, Start As Number[, Count As Number[, Compare As VbCompareMethod = vbBinaryCompare]]])

Right	Function Right(String, Length As Long)
Rnd	Function Rnd([Number]) As Single
RGB	Function RGB(red, green, blue)
Round	Function Round(Expression[, NumDecimalPlaces])
RTrim	Function RTrim(String)
ScriptEngine	Function ScriptEngine() As String
ScriptEngineBuildVersion	Function ScriptEngineBuildVersion As Number
ScriptEngineMajorVersion	Function ScriptEngineMajorVersion Function As Number
ScriptEngineMinorVersion	Function ScriptEngineMinorVersion Function As Number
Second	Function Second(Time)
Sgn	Function Sgn(Number)
Sin	Function Sin(Number As Double) As Double
Space	Function Space(Number As Long)
Split	Split(Expression[, Delimiter As String [, Count As Number[, Compare As Number]]])
Sqr	Function Sqr(Number As Double) As Double
StrComp	Function StrComp(String1, String2, [Compare As VbCompareMethod = vbBinaryCompare])
String	Function String(Number As Long, Character)
StrReverse	Function StrReverse(String)
Tan	Function Tan(Number As Double) As Double
Time	Function Time() As Date
TimeSerial	Function TimeSerial(Hour As Integer, Minute As Integer, Second As Integer)
TimeValue	Function TimeValue(Time As String)
Trim	Function Trim(String)
TypeName	Function TypeName(VarName) As String
UBound	Function UBound(Arrayname[, Dimension As Number])
UCase	Function UCase(String)
VarType	Function VarType(VarName) As VbVarType
Weekday	Function WeekDay(Date, [FirstDayOfWeek As VbDayOfWeek = vbSunday])
WeekdayName	Function WeekdayName(Weekday As Number, Abbreviate As Boolean, FirstDayOfWeek As Number)
Year	Function Year(Date)

Abs Function	
Description	The absolute value of a number is returned.
Syntax	Abs(number) The factor of a number should be a valid expression. If the number has the value zero, null is returned. And if the number is a non-initialized variable, 0 is returned

Reference	The absolute value of a number shows only its size without code. For example, in both Abs(-1) and Abs(1), 1 is returned. In the following example, Abs Function calculates the absolute value of a number. *Dim MyNumber* *MyNumber = Abs(50.3) '50.3 is returned* *MyNumber = Abs(-50.3) '50.3 is returned*

Array Function	
Description	A Variant including Array is returned
Syntax	Array(arglist) The arglist factor consists of values distinguished with pauses and it designates the Array factor included in the Variant to this value. If any factor is not designated, Array whose length is 0 is created.
Reference	This notation is used when referring to variable elements consisting of variable names next to a bracket that has index numbers indicating essential elements. For example, the first statement creates variable A equal to Variant. The second statement designates Array to variable A. The last statement designates a value that includes the variable's second Array element. Dim A As Variant A = Array(10,20,30) B = A(2) The minimum limit of the Array created with this Array Function is decided with the minimum limit that Option Base statement designates. The lower limit of the Array created with this Array Function is decided with the lower limit that Option Base statement designates or the name of the Array Function is decided with the name of the Form Library (ex: VBA.Array). Once the name is designated with the Form Library, this Array Function is not influenced by Option Base statement anymore. Even Variants not declared by Memo Array can include Array. Variant variables can include all forms of Array except fixed-length string forms and customized forms. Even though Variant Form including Array is different from Array, whose element is of variant form in terms of concept, the elements of the relevant Array can be equally accessed.

Asc Function	
Description	ANSI character code corresponding to the first character of a string is returned.
Syntax	Asc(string) The string factor should be a valid character expression. If there is no character in the string,

	Runtime errors take place.
Reference	Asc returns the ANSI character code of the first character of each string as seen in the following example. Dim MyNumber MyNumber = Asc("A") '65 is returned. MyNumber = Asc("a") '97 is returned. MyNumber = Asc("Apple") '65 is returned. Memo AscB Function uses Byte data that the string has. AscB does not return the character code of the first character but the first Byte of the character. AscW Function is provided for 32bit platforms using Unicode characters. Since this Function returns the character codes of Unicode, it is not necessary to convert Unicode to ANSI code..

Atn Function	
Description	The arctangent of a number is returned.
Syntax	Atn(number) The number factor should be a valid expression.
Reference	This Atn Function uses the ratio between lengths of two sides of a right triangle and returns alternate angles in radian unit. This ratio is obtained by dividing the length of the side facing the relevant angle by the length of the side adjacent to this angle. The results are ranged between pi/2 and pi/2 radian. To convert angle to radian, multiply angle by pi/180 and to convert radian to angle, multiply radian by 180/pi. In the following example, Atn calculates pi value. Dim pi pi = 4 * Atn(1) ' Calculate pi value.. Memo : Atn Function is a reverse triangle function of Tan that returns the ratio between two sides of a right triangle by using angle as a factor. Do not confuse Atn with cotangent, a reverse tangent (1/tangent)..

CBool Function	
Description	The expression converted to a Boolean subtype of Variant is returned.
Syntax	CBool(expression) The expression factor should be a valid expression.
Reference	If the expression is 0, False is returned or if not, True is returned. This expression can not be

	interpreted into number values and Runtime errors take place. In the following example, CBool Function converts the expression to Boolean. If the expression is not 0, CBool returns True or if it is 0, It returns False Dim A, B, Check A = 5: B = 5 ' Initialize variables. Check = CBool(A = B) ' Check gets True. A = 0 ' Define a variable. Check = CBool(A) ' Check gets False

CByte Function	
Description	The expression converted to a Byte subtype of Variant is returned.
Syntax	CByte(expression) The expression factor should be a valid expression.
Reference	Typically, if a code is created with a subtype of conversion function, the operation result can be seen in a particular data type rather than in a basic data type. For example, If Cbyte is used for call, single precision, double precision or integer arithmetic operation, Byte arithmetic operation is carried out. If CByte Function is used, other types of data can be converted to a Byte subtype widely used throughout the world. For example, symbols identified with different digit numbers such as symbols identified with a number in the thousands are properly recognized according to the locale settings of the computer. If an expression exceeds the tolerance range of Byte subtype, errors take place. In the following example, Cbyte Function converts the expression to Byte. Dim MyDouble, MyByte MyDouble = 125.5678 ' MyDouble is equal to Double. MyByte = CByte(MyDouble) ' MyByte gets value 126

CCur Function	
Description	The expression converted to a Currency subtype of Variant is returned.
Syntax	CCur(expression) The expression factor should be a valid expression.
Reference	Typically, if a code is created with a subtype of conversion function, the operation result can be seen in a particular data type rather than in a basic data type. For example, if Ccur is used for an integer arithmetic operation, a call arithmetic operation is executed. If CCur Function is used, other types of data can be converted to a widely-used Currency subtype throughout the world. For example, symbols identified with different digit numbers such

	as symbols identified with a number in the thousands are properly recognized according to the locale settings of the computer. In the following example, CCur Function converts the expression to Currency. Dim MyDouble, MyCurr MyDouble = 543.214588 ' MyDouble is equal to Double. MyCurr = CCur(MyDouble * 2) ' The value multiplied MyDouble by 2 (1086.429176) is converted to Currency (1086.4292).

CDate Function	
Description	The expression converted to a Date subtype of Variant is returned
Syntax	CDate(date) The date factor should be a valid expression.
Reference	If IsDate Function is used, it is decided whether to convert a date factor into date or time. CDate Function recognizes numbers within the tolerance range of date as well as date literal and time literal. If number is converted to date, the integer part is converted to date. The fractional part of number is converted to time of the day that starts from the midnight. CDate Function recognizes date type according to the computer's locale settings. If any unrecognized date type is provided, it is impossible to decide a correct order of year, month and day. It is also impossible to recognize the marked date type in detail even though there is a corresponding string to the day of the week. In the following example, CDate Function converts a string to date. Typically, it is not good to use the date and hard coded time with strings as seen in this example. Instead, it is better to use date and time literal such as #10/19/1962# or #4:45:23 PM#, MyDate = "October 19,1962" ' Decide date. MyShortDate = CDate(MyDate) ' It is converted to Date data type. MyTime = "4:35:47 PM" ' Decide time MyShortTime = CDate(MyTime) ' It is converted to Time data type

CDbl Function	
Description	The expression converted to a Double subtype of Variant is returned.
Syntax	CDbl(expression) The expression factor should be a valid expression.
Reference	Typically, if a code is created with a subtype of conversion function, the operation result can be seen in a particular data type rather than in a basic data type. For example, if CDbl or CSng is used for a call or integer arithmetic operation, single or double precision arithmetic operation is carried out.

	If CDbl Function is used, other types of data can be converted to a widely-used Double subtype throughout the world. For example, symbols identified with different digit numbers such as symbols identified with a number in the throusands are properly recognized according to the locale settings of the computer. In the following example, CDbl Function converts the expression to Double. Dim MyCurr, MyDouble MyCurr = CCur(234.456784) ' MyCurr is equal to Currency. MyDouble = CDbl(MyCurr * 8.2 * 0.01) ' The result is converted to Double

Chr Function	
Description	A character related to the designated ANSI character code is returned.
Syntax	Chr(charcode) The charcode factor is a number to identify characters
Reference	Numbers 0 through 31 are equal to the ASCII standard codes that can not be printed. For example, Chr(10) returns line feed characters. In the following example, Chr Function returns the character related to the designated ANSI character. Dim MyChar MyChar = Chr(65) ' A is returned. MyChar = Chr(97) ' a is returned. MyChar = Chr(62) ' > is returned. MyChar = Chr(37) ' %i s returned. Memo ChrB Function uses Byte data that the string has. 1Byte is always returned rather than 1Byte or 2Byte characters. ChrW Function is provided for 32bit platforms using Unicode characters. Since the factor of this Function is a Unicode character code, it is not necessary to convert ANSI to Unicode

CInt Function	
Description	The expression converted to an Integer subtype of Variant is returned
Syntax	CInt(expression) The expression factor should be a valid expression.
Reference	Typically, if a code is created with a subtype of conversion function, the operation result can be seen in a particular data type rather than in a basic data type. For example, if CInt or CLng Function is used for call, single precision or double precision arithmetic operation, integer arithmetic operation is carried out. If CInt Function is used, other types of data can be converted to a widely-used Integer subtype

	throughout the world. For example, symbols identified with different digit numbers such as symbols identified with a number in the throusands are properly recognized according to the locale settings of the computer. If an expression exceeds the tolerance range of Integer subtype, error takes place. In the following example, CInt Function converts the value to integer. Dim MyDouble, MyInt MyDouble = 2345.5678 ' MyDouble is equal to Double. MyInt = CInt(MyDouble) ' MyInt gets value 2346. Memo Cint is different from Fix or Int Function that does not round off but cut away decimal points. If the decimal point is exactly 0.5, CInt Function always rounds off the nearest even number. For example, 0.5 is rounded off to 0 and 1.5 is rounded off to 2.

CLng Function	
Description	The expression converted to a Long subtype of Variant is returned.
Syntax	CLng(expression) The expression factor should be a valid expression.
Reference	Typically, if a code is created with a subtype of conversion function, the operation result can be seen in a particular data type rather than in a basic data type. For example, if CInt or CLng Function is used for call, single precision or double precision arithmetic operation, integer arithmetic operation is carried out. If CLng Function is used, other types of data can be converted to a widely-used Long subtype throughout the world. For example, symbols identified with different digit numbers such as symbols identified with a number in the throusands are properly recognized according to the locale settings of the computer. If an expression exceeds the tolerance range of Integer subtype, errors take place. In the following example, CLng Function converts the value to Long. Dim MyVal1, MyVal2, MyLong1, MyLong2 MyVal1 = 25427.45: MyVal2 = 25427.55 ' MyVal1 and MyVal2 are Doubles. MyLong1 = CLng(MyVal1) ' MyLong1 gets value 25427. MyLong2 = CLng(MyVal2) ' MyLong2 gets value 25428. Memo CLng Function is different from Fix or Int Function that does not round off but cut away decimal points. If the decimal point is exactly 0.5, CLng Function always rounds off the nearest even number. For example, 0.5 is rounded off to 0 and 1.5 is rounded off to 2.

Cos Function	
Description	The cosine of an angle is returned
Syntax	Cos(number) The number factor should be a valid expression showing the angle in a radian.
Reference	Cos Function uses angles and returns the ratio between two sides of a right triangle that is, the ratio between the length of the adjoining side and the length of the oblique side of each angle. The results are between -1 through 1. To convert angle to radian, multiply angle by pi/180 and to convert radian to angle, multiply radian by 180/pi. In the following example, Cos Function returns the cosine value of the angle. Dim MyAngle, MySecant MyAngle = 1.3 ' Define an angle in a radian value. MySecant = 1 / Cos(MyAngle) ' Caculate the Secant value.

CreateObject Function	
Description	A reference for an automatic object is created and returned
Syntax	CreateObject(class) The class factor uses Syntax servername.typename and gets the following factors Description on factor servername: the name of an application program providing objects. typename: The type or class of an object to be produced.
Reference	An automatic server provides at least more than one object type. For example, the word processing program provides application program objects, document object and toolbar objects. To create automatic objects, designate the returned object from CreateObject Function as an object variable. Dim ExcelSheet Set ExcelSheet = CreateObject("Excel.Sheet") This code starts an application program while creating the relevant object, that is, Microsoft Excel spread sheet. Once the object is created, the object can be referred by using customized variables. In the following example, properties and methods of the new object are accessed by using ExcelSheet object variables and Excel objects including Application objects and ActiveSheet.Cells collections. The application examples are as follows: ' Make Excel program appear through Application objects. ExcelSheet.Application.Visible = True ' Put the relevant text on the first cell of the sheet. ExcelSheet.ActiveSheet.Cells(1,1). Value = "Column A , Row 1."

	' Save the sheet. ExcelSheet.SaveAs "C:\DOCS\TEST.XLS" ' Close Excel program by using 'Quit' method of the Application object. ExcelSheet.Application.Quit ' Release the object variable. Set ExcelSheet = Nothing

CSng Function	
Description	The expression converted to a Single subtype of Variant is returned
Syntax	CSng(expression) The expression factor should be a valid expression.
Reference	Typically, if a code is created with a subtype of conversion function, the operation result can be seen in a particular data type rather than in a basic data type. For example, if CDbl or CSng is used for a call or integer arithmetic operation, single or double precision arithmetic operation is carried out. If CSng Function is used, other types of data can be converted to a widely-used Single subtype throughout the world. For example, symbols identified with different digit numbers such as symbols identified with a number in the throusands are properly recognized according to the locale settings of the computer. If an expression exceeds the tolerance range of Byte subtype, errors take place. In the following example, CSng Function converts the value to Single. Dim MyDouble1, MyDouble2, MySingle1, MySingle2 ' MyDouble1 and MyDouble2 are Doubles. MyDouble1 = 75.3421115: MyDouble2 = 75.3421555 MySingle1 = CSng(MyDouble1) ' MySingle1 gets value 75.34211. MySingle2 = CSng(MyDouble2) ' MySingle2 gets value 75.34216.

CStr Function	
Description	The expression converted to a String subtype of Variant is returned
Syntax	CStr(expression) The expression factor should be a valid expression.
Reference	Typically, if a code is created with a subtype of conversion function, the operation result can be seen in a particular data type rather than in a basic data type. For example, if CStr is used, the result is expressed in a String. If CStr Function is used instead of Str Function, other types of data can be converted to a widely-used String subtype throughout the world. For example, symbols identified with different digit numbers are properly recognized according to the computer's locale settings. The data used in an expression decide the returned results as seen in the table.

Expression Data	CStr Returned Result
Boolean	String containing True or False
Date	String containing simplified date by computer settings.
Null	Runtime error
Empty	String whose length is 0 ("")
Error	String showing error number next to word 'Error'
Other	String containing numbers

In the following example, CStr Function converts the value to String.

```
Dim MyDouble, MyString
MyDouble = 437.324              ' MyDouble is equal to Double.
MyString = CStr(MyDouble)    ' MyString gets value "437.324.".
```

Date Function

Description	The Date of the current system is returned.
Syntax	Date
Reference	In the following example, Date Function returns the Date of the current system. Dim MyDate MyDate = Date ' MyDate gets the date of the current system.

DateAdd Function

Description	The Date that adds the designated time interval is returned.
Syntax	DateAdd(interval, number, date) DateAdd Function Syntax has following factors. Description on Factor

interval	A mandatory factor. A literal expression indicating the date that adds interval. For the value, refer to the following settings.
number	A mandatory factor. A numerical expression of the interval number to be added. In the numerical expression, a positive number refers to a future date while a negative number refers to a past date.
date	A mandatory factor. Variant or literal indicating the date that adds interval.

The value of the interval factor is as follows:

Setting	Description
yyyy	Year

	<table><tr><td>q</td><td>Quarter</td></tr><tr><td>m</td><td>Month</td></tr><tr><td>y</td><td>Year(As of one year)</td></tr><tr><td>d</td><td>Day</td></tr><tr><td>w</td><td>Weekday</td></tr><tr><td>ww</td><td>Week(As of one year)</td></tr><tr><td>h</td><td>Hour</td></tr><tr><td>n</td><td>Minute</td></tr><tr><td>s</td><td>Second</td></tr></table>
Reference	If DateAdd Function is used, it is possible to add the designated time interval to the date or subtract it from the date. For example, if DateAdd Function is used, it is possible to calculate the date that will be in 30 days from today or the time that will be in 45 minutes from the current time. To add a certain day to the date, use year("y"), day("d") or weekday("w"). In DateAdd Function, any wrong date is not returned. In the following example, one month is added to January 31. NewDate = DateAdd("m", 1, "31-Jan-95") In this case, DateAdd Function returns 28-Feb-95 rather than 31-Feb-95. If the date is 31-Jan-96, 29-Feb-96 is returned because year 1996 is a leap year. If the date to be calculated is before AD 100, errors take place. If it is not a number of Long value, the nearest integer shall be rounded off before calculation.

DateDiff Function	
Description	The difference between two dates is returned
Syntax	DateDiff(interval, date1, date2 [,firstdayofweek[, firstweekofyear]]) DateDiff Function Syntax has following factors. <table><tr><th>Factor</th><th>Description</th></tr><tr><td>interval</td><td>A mandatory factor. A literal expression of interval used to calculate the difference between Date 1 and Date 2. For the value, refer to the following settings.</td></tr><tr><td>date1, date2</td><td>A mandatory factor. Two dates used to calculate in the date expression.</td></tr><tr><td>firstdayofweek</td><td>An optional factor. Sunday is deemed as the day of the week if any Constant is not designated to indicate which day of the week. For the value, refer to the following settings.</td></tr></table>

firstweekofyear	An optional factor. A constant indicating the first week of the year. If not indicated, the week that contains January 1 is deemed as the first week in the year. For the value, refer to the following settings.

Description on Factor

interval: A mandatory factor. A literal expression of interval used to calculate the difference between Date 1 and Date 2. For the value, refer to the following settings.

date1, date2: A mandatory factor. Two dates used to calculate in the date expression.

firstdayofweek: An optional factor. Sunday is deemed as the day of the week if any Constant is not designated to indicate which day of the week. For the value, refer to the following settings.

firstweekofyear: An optional factor. A constant indicating the first week of the year. If not indicated, the week that contains January 1 is deemed as the first week in the year. For the value, refer to the following settings

Settings

The value of the interval factor is as follows:

Setting	Description
yyyy	Year
q	Quarter
m	Month
y	Year(As of one year)
d	Day
w	Weekday
ww	Week(As of one year)
h	Hour
n	Minute
s	Second

The value of the firstweekofyearl factor is as follows:

Constant	Value	Description
vbUseSystem	0	NLS(National Language Support). Use API settings
vbSunday	1	Sunday(Default Value)
vbMonday	2	Monday
vbTuesday	3	Tuesday
vbWednesday	4	Wednesday
vbThursday	5	Thursday
vbFriday	6	Friday
vbSaturday	7	Saturday

<table>
<tr><td></td><td>Description on Constant Value
<table>
<tr><th>Constant</th><th>Value</th><th>Description</th></tr>
<tr><td>vbUseSystem</td><td>0</td><td>NLS(National Language Support). Use API settings</td></tr>
<tr><td>vbFirstJan1</td><td>1</td><td>Start from the week that contains January 1 (Default Value)</td></tr>
<tr><td>vbFirstFourDays</td><td>2</td><td>Start from the week that contains the first four days in the new year.</td></tr>
<tr><td>vbFirstFullWeek</td><td>3</td><td>Start from the week that contains the first full week (7 days) in the new year.</td></tr>
</table></td></tr>
<tr><td>Reference</td><td>If DateDiff Function is used, it is possible to know the time difference between two dates. For example, if DateDiff Function is used, it is possible to calculate the day number between two dates or the week number between today and the end of the year.
To calculate the day number between Date 1 and Date 2, use year ("y") or day ("d") as of the year. If interval is expressed with weekday ("w"), DateDiff Function returns the week number between two dates. If date1 is on Monday, DateDiff Function calculates how many Mondays are there by date2. That is, date2 is calculated instead of date 1. But, if interval is expressed with week ("ww"), DateDiff Function returns the calendar week numbers between two dates. That is, the Sunday number is calculated between date1 and date2. In DateDiff Function, if date2 is on Sunday, it calculates date2 but, if date1 is also on Sunday, it does not calculate date1.

If date1 is latter than date2, DateDiff Function returns a negative number.

The firstdayofweek factor has influence on the calculation using interval symbols such as "w" and "ww."

If date1 or date2 is a date literal, the designated year becomes the permanent element of the relevant date. But, if a pair of quotation marks (" ") is place around date1 or date2 and the relevant year is omitted, the current year shall be inserted to the user code whenever date1 or date2 expression is calculated. The code that can be used for the next year can be written in this way.

If December 31 is compared to the next January 1, DateDiff returns 1 to the year ("yyyy") factor even only one day is elapsed.
In the following example, DateDiff Function indicates the elapsed days until now from the given date.

Function DiffADate(theDate)
DiffADate = "Elapsed Day: " & DateDiff("d", Now, theDate)
End Function</td></tr>
</table>

DatePart Function

Description	The designated part of a given date is returned.

Syntax

DatePart(interval, date[, firstdayofweek[, firstweekofyear]])

DatePart Function Syntax has following factors.

factor	Description
interval	A mandatory factor. A literal expression of the time interval to be returned. For the value, refer to the following settings.
date	A mandatory factor. A date expression to be calculated.
firstdayof week	An optional factor. Sunday is deemed as the day of the week if any Constant is not designated to indicate which day of the week. For the value, refer to the following settings.
firstweekofyear	An optional factor. A constant indicating the first week of the year. If not indicated, the week that contains January 1 is deemed as the first week in the year. For the value, refer to the following settings.

Settings

The value of the interval factor is as follows:

Settings	Description
yyyy	Year
q	Quarter
m	Month
y	Year (As of one year)
d	Day
w	Weekday
ww	Week(As of one year)
h	Hour
n	Minute
s	Second

The value of the firstweekofyear factor is as follows:

Constant	Value	Description
vbUseSystem	0	NLS(National Language Support). Use API settings
vbSunday	1	Sunday(Default Value)
vbMonday	2	Monday
vbTuesday	3	Tuesday
vbWednesday	4	Wednesday
vbThursday	5	Thursday
vbFriday	6	Friday

	vbSaturday | 7 | Saturday Description on Constant Value **Constant** | **Value** | **Description** vbUseSystem | 0 | NLS(National Language Support). Use API settings vbFirstJan1 | 1 | Start from the week that contains January 1 (Default Value) vbFirstFourDays | 2 | Start from the week that contains the first four days in the new year. vbFirstFullWeek | 3 | Start from the week that contains the first full week (7 days) in the new year.
Reference	If DatePart Function is used, the date is calculated and a certain time interval is returned. For example, it is possible to calculate a weekday or the current time by using DatePart. The firstdayofweek factor has influence on the calculation using interval symbols such as "w" and "ww." If the date is a date literal, the designated year becomes the permanent element of the relevant date. But, if a pair of quotation marks (" ") is place around the date and the relevant year is omitted, the current year shall be inserted to the user code whenever the date expression is calculated. The code that can be used for the next year can be written in this way. The following example shows Quarter of the relevant year by using DatePart Function and date. Function GetQuarter(TheDate) GetQuarter = DatePart("q", TheDate) End Funtion

DateSerial Function	
Description	The Date subtype of Variant of the designated year, month and day is returned.
Syntax	DateSerial(year, month, day) DateSerial Function Syntax has following factors **Factor** | **Description** *year* | Numbers 100 through 9999 or a numerical expression *month* | All expressions *day* | All expressions
Reference	For example, to designate a certain date such as December 31, 1991, the number of each DateSerial factor should be within the tolerance range of the relevant. That is, for day, it should be one from the first day through the thirty first day and it should be one from January through December for month. But, the counter date of each factor can be designated by using a

	numerical expression showing the year, month or day before and after a particular date. The following example uses a numerical expression instead of an absolute date number. Here, DateSerial Function returns the date subtracted one day from the first day(1-1) on the month two months before August (8-2) in the year 10 years before 1990 (1990-10). DateSerial(1990 - 10, 8 - 2, 1 - 1) If the value of the year factor is one from 0 through 99, it is interpreted from 1900 through 1999. In all other year factors except that, a four-digit number such as 1800 is used. If the tolerance range of the factor is exceeded, it goes to the next bigger unit For example, if the thirty fifth day is designated, it is calculated to which day of one month according to the applicated year. But, if a certain factor exceeds range -32,768~32 and 767 or a date designated directly or through an expression by using three factors. Errors take place.

DateValue Function	
Description	The Date subtype of Variant is returned.
Syntax	DateValue(date) Typically, the date factor is a literal expression indicating from January 1, 100 to December 31, 9999 and the expression shows the date, time or date/time in this range
Reference	If the date factor contains time information, DateValue Function does not return the information. However, some wrong information such as "89:98" is in the date factor, errors take place. If the date factor is a string that has only the numbers divided by valid date separating symbols, DateValue Function recognizes year, month and day according to the simplified date marking type of the computer settings. Also, it is possible to recognize an accurate date that contains the formal or informal date name. For example, DateValue Function recognizes December 30, 1991 or Dec 30, 1991 as well as 2/30/1991 or 12/30/91. If the year part is omitted from the date factor, DateValue Function uses the current year as set up on the computer. In the following example, DateValue Function converts a string to date. It is possible to directly assign the date to Variant variable by using the date literal (ex. MyDate = #9/11/63). Dim MyDate MyDate = DateValue("September 11, 1963") ' Date is returned.

Day Function	
Description	The integer indicating the date from 1 through 31 is returned.
Syntax	Day(date)

	The date factor is an expression showing the date. If the date factor has the value zero, Null is returned
Reference	Dim MyDay MyDay = Day("Oct 19, 1962") 'MyDay gets value 19.

Eval Function	
Description	A sentence as a factor is calculated and Result Value or the stack value of variable is returned.
Syntax	Eval(Expr[, Depth]) Expr : As an expression to be operated. VB script syntax should be inputted as a string value. Depth: Numbers should be inputted. Input the value location to the local variable's stack. If Depth = 0, the current value at the operation stage is returned and if omitted, Depth is 0

Reference	The following example shows how Eval Function is used.
	
	Ex1)
	Dim Guess, RndNum
	RndNum = Int((100) * Rnd(1) + 1)
	Guess = CInt(InputBox("Enter your guess:",,0))
	Do
	If Eval("Guess = RndNum") Then
	MsgBox "Congratulations! You guessed it!"
	Exit Sub
	Else
	Guess = CInt(InputBox("Sorry! Try again.",,0))
	End If
	Loop Until Guess = 0
	
	Ex 2)
	Sub Main
	Dim X As String
	X = "Hello"
	Debug.Print Eval("X") 'Return Hello
	A
	End Sub
	Sub A
	Dim X As String
	X = "Bye"
	Debug.Print Eval("X") 'Return Bye
	Debug.Print Eval("X",1) 'Retusn the previous value of Hello
	End Sub

Exp Function	
Description	‘e’, a root of natural logarithms raised by the multiplier is returned
Syntax	Exp(number) The number factor should be a valid expression
Reference	If the number value exceeds 709.782712893, errors take place. Constant is about 2.718282. Memo Exp Function makes up for Log Function and it is sometimes called an antilogarithm. In the following example, Exp Function returns e for angle. MyAngle = 1.3 ' Define an angle in a radian value. MyAngle = 1.3 ' Calculate the sine value of a hyperbola. MyHSin = (Exp(MyAngle) - Exp(-1 * MyAngle)) / 2

Filter Function	
Description	The Array based on 0, which has a subset of String Array according to the designated filter base, is returned
Syntax	Filter(InputStrings, value[, Include[, compare]]) Filter Function Syntax has following factors

Factor	Description
InputStrings	A mandatory factor. 0ne dimension Array of the String to be searched.
value	A mandatory factor. The String to be searched.
Include	An optional factor. A boolean value showing whether to return the sub string including or excluding *value.* If it is True, Filter Function returns a subset of the Array corresponding to *value* as a sub string. If it is False, Filter Function returns a subset of the Array not corresponding to *value* as a sub string.
compare	An optional factor. A number value showing the comparison type to be used. For the value, refer to the following settings.

Settings

Comparison factors have the following values

Constant	Value	Description
vbBinaryCompare	0	Run Binary Comparison.
vbTextCompare	1	Run Text Comparison.

Reference	If there is no value agreed with InputStrings, Filter Function returns empty Array. If InputStrings has the value zero or it is not one dimension Array, errors take place. The returned Array in Filter Function has only the elements enough to contain the agreed item number. In the following example, Filter Function returns the Array that has "Mon" search reference. Dim MyIndex Dim MyArray (3) MyArray(0) = “Sunday” MyArray(1) = “Monday” MyArray(2) = “Tuesday” MyIndex = Filter(MyArray, "Mon") 'MyIndex(0) contains “Monday”

Fix Function	
Description	The integer part of a number is returned
Syntax	Int(number) Fix(number) The number factor should be a valid expression. The number has the value zero, Null is returned.
Reference	Both of Int Function and Fix Function return the integer value that shows the results of removing the part after the decimal point. If the difference between Int Function and Fix Function is a negative number, Int Function returns the first negative integer that is lower or equal to the number but, Fix Function returns the negative integer that is higher or equal to the number. For example, Int Function converts -8.4 to -9 and Fix Function converts -8.4 to -8. Fix (number) Function is equal to the following expression. Sgn(number) * Int(Abs(number)) The following example shows how Int and Fix Function convert the integer part of a number. MyNumber = Int(99.8) ' 99 is returned. MyNumber = Fix(99.2) ' 99 is returned. MyNumber = Int(-99.8) ' -100 is returned. MyNumber = Fix(-99.8) ' -99 is returned. MyNumber = Int(-99.2) ' -100 is returned. MyNumber = Fix(-99.2) ' -99 is returned.

FormatCurrency Function

Description

A formalized expression with the call values that use the defined call symbols on the computer's [Control Board] is returned.

Syntax

FormatCurrency(expression[,NumDigitsAfterDecimal [,IncludeLeadingDigit [,UseParensForNegativeNumbers [,GroupDigits]]]])

FormatCurrency Function Syntax has following factors.

Factor	Description
expression	A mandatory factor. An expression whose format is specified.
NumDigitsAfterDecimal	An optional factor. A number value indicating the number of digits after the decimal point. The Default Value is -1, showing the use of the computer's national settings.
IncludeLeadingDigit	An optional factor. A Tristate Constant showing whether to mark 0 before a valid number after the decimal point. For the value, refer to the following settings.
UseParensForNegative Numbers	An optional factor. A Tristate Constant showing whether to place a negative value in a paranthesis. For the value, refer to the following settings.
GroupDigits	An optional factor. A Tristate Constant showing whether to identify a number with the designated group distinguishing symbols according to the computer's national settings.

Settings

IncludeLeadingDigit, UseParensForNegativeNumbers and GroupDigits factor have the following values

Constant	Value	Description
TristateTrue	-1	True
TristateFalse	0	False
TristateUseDefault	-2	Use of the computer's national settings.

Reference

If more than one factor is omitted, the factor value that omits the computer's national setting value is used. The location of the call symbols related to call values is decided by the computer's national setting value.

Memo

The entire setting information except previously acted 0 (zero) on [Number] tap is on [Call] tap of [National settings].

FormatDateTime Function

Description	A formalized expression with date or time is returned
Syntax	FormatDateTime(date[,NamedFormat]) FormatDateTime Function Syntax has following factors

Factor	Description
date	A mandatory factor. Specify a date expression.
NamedFormat	An optional factor. A number value showing the used date/time format. If this factor is omitted, vbGeneralDate is used.

Settings

NamedFormat factor has the following settings

Constant	Value	Description
vbGeneralDate	0	Mark date or time. If there is date part, mark the date in a simple date type, If there is time part, mark the time in a detailed time type. If date and time exist at the same time, mark both of them.
vbLongDate	1	Show date in the detailed date type designated by the computer's national settings.
vbShortDate	2	Show date in the simple date type designated by the computer's national settings.
vbLongTime	3	Show time in the time type designated by the computer's national settings.
vbShortTime	4	Show time in 24-hr type (hh:mm).

Reference

In the following example, FormatDateTime Function designates the expression in a long date type and exports it to MyDateTime.

```
Function GetCurrentDate
   'FormatDateTime specifies Date in a long date type.
   GetCurrentDate = FormatDateTime(Date, 1)
End Function
```

FormatNumber Function

Description	A formalized expression with number is returned
Syntax	FormatNumber(expression[,NumDigitsAfterDecimal [,IncludeLeadingDigit [,UseParensForNegativeNumbers [,GroupDigits]]]]) FormatNumber Function Syntax has following factors

Factor	Description
expression	A mandatory factor. An expression whose format is specified.

NumDigitsAfterDecimal	An optional factor. A number value indicating the number of digits after the decimal point. The Default Value is -1, showing the use of the computer's national settings.
IncludeLeadingDigit	An optional factor. A Tristate Constant showing whether to mark 0 before a valid number after the decimal point. For the value, refer to the following settings.
UseParensForNegativeNumbers	An optional factor. A Tristate Constant showing whether to place a negative value in a parenthesis. For the value, refer to the following settings.
GroupDigits	An optional factor. A Tristate Constant showing whether to identify a number with the designated group distinguishing symbols set up on [Control Board]. For the value, refer to the following settings.

Settings

IncludeLeadingDigit, UseParensForNegativeNumbers and GroupDigits factor have the following settings

Constant	Value	Description
TristateTrue	-1	True
TristateFalse	0	False
TristateUseDefault	-2	The computer's national settings are used.

Reference

If more than one optional factor is omitted, the factor value that omits the computer's national setting value is used.

Memo

The entire setting information is on [Number] tap of [National Settings].

In the following example, FormatNumber Function specifies a number in a four-digit number of a decimal system.

```
Function FormatNumberDemo
   Dim MyAngle, MySecant, MyNumber
   MyAngle = 1.3                    ' Define an angle in a radian value.
   MySecant = 1 / Cos(MyAngle)    ' Caculate the Secant value. .
   FormatNumberDemo = FormatNumber(MySecant,4) ' Specify MySecant in a four-digit number of a decimal system.
End Function
```

FormatPercent Function

Description

An expression that has character % at the end and whose format is designated in percentage multiplied by 100 is returned.

Syntax

FormatPercent(expression[,NumDigitsAfterDecimal [,IncludeLeadingDigit [,UseParensForNegativeNumbers [,GroupDigits]]]])

FormatPercent Function Syntax has following factors.

factor	Description
expression	A mandatory factor. A formalized expression is returned.
NumDigitsAfterDecimal	An optional factor. A number value indicating the number of digits after the decimal point. The Default Value is -1, showing the use of the computer's national settings.
IncludeLeadingDigit	An optional factor. A Tristate Constant showing whether to mark 0 before a valid number after the decimal point. For the value, refer to the following settings.
UseParensForNegativeNumbers	An optional factor. A Tristate Constant showing whether to place a negative value in a parenthesis. For the value, refer to the following settings.
GroupDigits	An optional factor. A Tristate Constant showing whether to identify a number with the designated group distinguishing symbols set up on [Control Board]. For the value, refer to the following settings.

Settings

IncludeLeadingDigit, UseParensForNegativeNumbers, GroupDigits factor는 have the following settings

Constant	Value	Description
TristateTrue	-1	True
TristateFalse	0	False
TristateUseDefault	-2	The computer's national settings are used.

Reference

If more than one optional factor is omitted, the factor value that omits the computer's national setting value is used.

Memo

The entire setting information is on [Number] tap of [National Settings].

In the following example, FormatPercent Function specifies an expression in a type of percentage.

```
Dim MyPercent
MyPercent = FormatPercent(2/32) 'MyPercent gets 6.25%.
```

GetObject Function

Description	A reference on automatic objects in the file is returned
Syntax	Syntax GetObject([pathname] [, class]) GetObject Function Syntax has following factors

Factor	Description
pathname	An optional factor. A string showing the name and entire paths of a file that has the object to be searched. If *pathname* is omitted, *class* should exist.
class	An optional factor. A string showing the Class of an object.

The class factor uses appname.objectype syntax, and its configuration element is described as follows

Configuration Element	Description
appname	A mandatory factor. A string showing the name of the application program supplying the relevant object.
objectype	A mandatory factor. A string showing the class of format of the object to be created.

Hex Function

Description	A string showing the hexadecimal value of a number is returned
Syntax	Hex(number) The number factor should be a valid expression
Reference	Numbers that are not integer are rounded off to the nearest integer before calculating.

Number Value	Result of Hex Function
Null	Null
Empty	0
Numbers other than two value above	Hexadecimal characters less than 8

Hexadecimal can be expressed by attaching &H before the numbers in a valid range. For example, &H10 expresses hexadecimal mark of decimal number 16.

In the following example, Hex Function returns the hexadecimal value of a number.

```
Dim MyHex
MyHex = Hex(5)      ' 5 is returned.
MyHex = Hex(10)     ' A Is returned.
MyHex = Hex(459)    ' 1CB is returned.
```

Hour Function	
Description	An integer indicating the hour of a day from 0 through 23 is returned.
Syntax	Hour(time) The time factor is an expression showing time. If time has the value zero, Null is returned..
Reference	In the following example, Hour Function exports hour from the current time. Dim MyTime, MyHour MyTime = Now MyHour = Hour(MyTime) ' MyHour gets a number showing the current hour

InputBox Function	
Description	It displays a prompt in the dialog box or it waits until any text is inputted or a button is pressed or it returns the inputted information
Syntax	InputBox(prompt[, title][, default][, xpos][, ypos][, helpfile, context]) InputBox Function Syntax has following factors.
Reference	If both of helpfile and context are provided, [Help] button is automatically added in the dialog

factor	Description
prompt	A literal expression showing message in the dialog box. *prompt* is about 1,024 characters long at the most even though it varies depending on the width of the characters. If *prompt* is long enough to take more than two lines, the lines can be divided by carriage return character (Chr(13)), line feed character(Chr(10)) or a combination of carriage return and line feed characters (Chr(13) & Chr(10)).
title	A literal expression showing on the title line in the dialog box. If *title*< is omitted, the application program title appears on the tile line.
default	A literal expression basically displayed on the text box if noting is inputted. If *default* is omitted, an empty text box appears.
xpos	A numerical expression specifying the horizontal length from the screen left end to the dialog left end in a twip unit. If *xpos* is omitted, the dialog box is located in the middle on the screen.
ypos	A numerical expression specifying the vertical length from the screen upper end to the dialog upper end in a twip unit. If *ypos* is omitted, the dialog box is located on the point 1/3 above from the bottom of the screen.
helpfile	A literal expression identifying the ‘Help’ file to provide assistance according to the situation If *helpfile* is provided, it is necessary also to provide *context*.
context	A numerical expression identifying the Help context number as the relevant Help items designated by the author. If *context* is provided, it is necessary also to provide *helpfile*.

	box. In InputBox Function, the contents in the text box are returned if [Ok] or Key is pressed. If [Cancel] is pressed, string ("") whose length is 0 is returned. In the following example, InputBox Function displays a text box and assigns string to variable. Dim Input Input = InputBox("Input a name") MsgBox ("The following name is inputted: " & Input)

InStr Function

Description	The position on which a string firstly appears in other string is returned.
Syntax	InStr([start,]string1, string2[, compare]) InStr Function Syntax has following factors

factor	Description
start	An optional factor. A numerical expression specifying a position at which each search starts. If omitted, search starts from the first character. The *start* factor is Null, errors takes place. To specify *Compare,* the *start* factor is needed.
string1	A mandatory factor. A literal expression to be searched.
string2	A mandatory factor. A literal expression subject to searching.
compare	An optional factor. A number value displayed in a comparison type to be used for assessing sub strings. For the value, refer to the following settings. If omitted, Run Binary Comparison.

Settings

Comparison factors have the following values.

Constant	Value	Description
vbBinaryCompare	0	Run Binary Comparison.
vbTextCompare	1	Run Text Comparison.

Result Value

InStr Function returns the following values.

Condition	Result of InStr Function
If the length of string1 is 0	0
If string1 is Null	Null
If the length of string2 is 0	*start*
If string2 is Null	Null

	If there is no string2 | 0
	If string1 has string2 | *The relevant location in string1*
	start > Len(*string2*) | 0
Reference	In the following example, InStr searches a string. Dim SearchString, SearchChar, MyPos SearchString ="XXpXXpXXPXXP" ' String to be searched SearchChar = "P" ' Search “P” ' A text comparison starts at position 4. 6 is returned. MyPos = Instr(4, SearchString, SearchChar, 1) ' A binary comparison starts at position 1. 9 is returned. MyPos = Instr(1, SearchString, SearchChar, 0) ' Comparison is binary by default (last argument is omitted). MyPos = Instr(SearchString, SearchChar) ' Returns 9. ' A binary comparison starting at position 1. Returns 0 ("W" is not found). MyPos = Instr(1, SearchString, "W") Memo InStrB Function uses Byte data that the string has. InStrB Function returns Byte position rather than the position where a string appears in the other string for the first time.

InStrRev Function

Description	The position on which a string firstly appears in other string is returned at the end of the string.
Syntax	InStrRev(string1, string2[, start[, compare]]) InStrRev Function Syntax has following factors.

factor	Description
string1	A mandatory factor. A literal expression to be searched.
string2	A mandatory factor. A literal expression subject to searching.
start	An optional factor. A numerical expression specifying a position at which each search starts. If omitted, -1 is used and it means search starts at the last character If *start* is Null, errors takes place.
compare	An optional factor. A number value displayed in a comparison type to be used for assessing sub strings. If omitted, Run Binary Comparison. For the value, refer to the following settings.

Settings

Comparison factors have the following values

Constant	Value	Description
vbBinaryCompare	0	Run Binary Comparison.
vbTextCompare	1	Run Text Comparison.

Result Value

InStrRev Function returns the following values

Condition	InStrRev FunctionResult of
If the length of string1 is 0	0
If string1 is Null	Null
If the length of string2 is 0	*start*
If string2 is Null	Null
If there is no string2	0
If string1 has string2	*The relevant location in string1*
start > Len(*string2*)	0

Reference

In the following example, InStrRev Function searches a string.

```
Dim SearchString, SearchChar, MyPos
SearchString ="XXpXXpXXPXXP"          ' String to be searched
SearchChar = "P"                      ' Search “P”

' A binary comparison starts at position 10. 9 is returned.
MyPos = InstrRev(SearchString, SearchChar, 10, 0)

' A text comparison starts at the last position. 12 is returned.
MyPos = InstrRev(SearchString, SearchChar, -1, 1)

' For comparison, the Default Value is binary (The last factor is omitted). 0 is returned.
MyPos = InstrRev(SearchString, SearchChar, 8)
```

Memo InStrRev Function Syntax is different from InStr Function Syntax.

Int Function

Description	The integer part of a number is returned.
Syntax	Int(number) Fix(number)

	The number factor should be a valid expression. number에 Has the value zeroNull is returned.
Reference	Both of Int Function and Fix Function return the integer value that shows the results of removing the part after the decimal point. If the difference between Int Function and Fix Function is a negative number, Int Function returns the first negative integer that is lower or equal to the number but, Fix Function returns the negative integer that is higher or equal to the number. For example, Int Function converts -8.4 to -9 and Fix Function converts -8.4 to -8. Fix(number) Function is equal to the following expression. Sgn(number) * Int(Abs(number)) The following example shows how Int and Fix Function convert the integer part of a number. MyNumber = Int(99.8) ' 99 is returned. MyNumber = Fix(99.2) ' 99 is returned. MyNumber = Int(-99.8) ' -100 is returned. MyNumber = Fix(-99.8) ' -99 is returned. MyNumber = Int(-99.2) ' -100 is returned. MyNumber = Fix(-99.2) ' -99 is returned.

IsArray Function	
Description	A boolean value showing whether variable is Array or not is returned
Syntax	IsArray(varname) The varname factor is variable.
Reference	IsArray Function returns True if variable is Array and it returns False if variable is not Array. IsArray Function is especially useful if Variant that contains Array is used. In the following example, IsArray Function tests whether MyVariable is Array or not. Dim MyVariable Dim MyArray(3) MyArray(0) = “Sunday” MyArray(1) = “Monday” MyArray(2) = “Tuesday” MyVariable = IsArray(MyArray) ' MyVariable gets "True."

IsDate Function	
Description	A boolean value showing whether an expression is converted to date is returned
Syntax	IsDate(expression) The expression factor is a literal expression that can be recognized as a date expression or date

	or time
Reference	IsDate Function returns True if the relevant expression is date or it can be converted to a valid date while if not, False is returned. Even though a date range from January 1, AD 100 to January 1, AD 9999 is valid on the Microsoft Windows, its range varies depending on the operation system. In the following example, IsDate Function decides whether an expression can be converted to date. Dim MyDate, YourDate, NoDate, MyCheck MyDate = "October 19, 1962": YourDate = #10/19/62#: NoDate = "Hello" MyCheck = IsDate(MyDate) ' True is returned. MyCheck = IsDate(YourDate) ' True is returned. MyCheck = IsDate(NoDate) ' False is returned

IsEmpty Function	
Description	A boolean value showing whether Variable is initialized or not is returned
Syntax	IsEmpty(expression) The expression factor can be any expression. However, a typical expression factor is the name of a single variable because IsEmpty Function is used to ensure whether an individual variable is initialized or not
Reference	IsEmpty Function returns True if the variable is not initialized or the variable is expressly designated as Empty or False is returned. If the expression has more than two variables, False is always returned. In the following example, IsEmpty Function judges whether the variable is initialized. Dim MyVar, MyCheck MyCheck = IsEmpty(MyVar) ' True is returned. MyVar = Null ' Assign Null. MyCheck = IsEmpty(MyVar) ' False is returned. MyVar = Empty ' Assign Empty. MyCheck = IsEmpty(MyVar) ' True is returned.

IsNull Function	
Description	A boolean value showing whether a Null expression has any valid date that is, it is Null or not is returned
Syntax	IsNull(expression) The expression factor can be any expression

Reference	IsNull Function retunrs True if the expression is Null that is, it have no valid date, otherwise, False is returned. If the expression consists of more than two variables, Null that belongs to the entire configuration variables returns True for the full expressions. Null value means the relevant variable has no valid data. Null is different from Empty, which shows the relevant variable is not yet initialized and it is also different from String (""), whose length is 0 and it is sometimes called Null String. Memo It is decided whether an expression has Null value by using IsNull Function. In some cases, an expression is always False even though True is expected as seen in expressions such as If Var = Null and If Var <> Null. It becomes False because the expression itself contains Null. In the following example, IsNull Function decides whether the variable has Null value. Dim MyVar, MyCheck MyCheck = IsNull(MyVar) ' False is returned. MyVar = Null ' Designate Null. MyCheck = IsNull(MyVar) ' True is returned. MyVar = Empty ' Designate Empty. MyCheck = IsNull(MyVar) ' Designate False.

IsNumeric Function	
Description	A boolean value showing whether an expression can be evaluated with numbers
Syntax	IsNumeric(expression) The expression factor can be any expression
Reference	If the expression is a date expression, IsNumeric Function returns False. In the following example, IsNumeric Function decides whether the variable can be evaluated with numbers. Dim MyVar, MyCheck MyVar = 53 ' Designate a value. MyCheck = IsNumeric(MyVar) ' True is returned. MyVar = "459.95" ' Designate a value. MyCheck = IsNumeric(MyVar) ' True is returned. MyVar = "45 Help" ' Designate a value. MyCheck = IsNumeric(MyVar) ' False is returned.

IsObject Function

Description	A boolean value showing whether an expression refers valid automatic objects is returned
Syntax	IsObject(expression) The expression factor can be any expression
Reference	IsObject Function returns True if the relevant expression is a subtype variable of Object or a customized object, otherwise, False is returned. In the following example, IsObject Function decides whether the identifier shows individual variables. Dim MyInt, MyCheck, MyObject Set MyObject = Me MyCheck = IsObject(MyObject) ' True is returned. MyCheck = IsObject(MyInt) ' False is returned

Join Function

Description	A string made by combining various sub strings included in Array is returned
Syntax	Join(list[, delimiter]) Join Function Syntax has following factors

factor	Description
list	A mandatory factor. A one-dimension Array including sub strings to be combined.
delimiter	An optional factor. A character used to separate sub strings of the returned string. If omitted, empty character (" ") is used. If *delimiter* is a string whose length is 0, the entire items in the list are combined without any symbol to divide them.

LBound Function

Description	The smallest subscript available in the designated Array level is returned
Syntax	LBound(arrayname[, dimension]) LBound Function Syntax has following factors

Factor	Description
arrayname	Comply with the notation rule of general variable with Array variable name
dimension	An integer showing the lower limit of the returned level is returned. Use 1 for one dimension, 2 for two dimension, etc. If *dimension* is omitted, 1 is deemed.

Reference	LBound Function is used along with UBound Function to decide the size of Array. If UBound Function is used, the upper limit can be known in the Array level. The lower limit of the entire dimensions is always 0

LCase Function

Description	A string converted to decimal point is returned
Syntax	LCase(string) The string factor should be a valid character expression. If there is Null in string, Null is returned
Reference	This function converts only capital letters to small letters and makes other characters except them remain as they are without converting. In the following example, CDate Function converts capital letters to small letters. Dim MyString Dim LCaseString MyString = "VBSCript" LCaseString = LCase(MyString) ' LcaseString gets "vbscript."

Left Function

Description	Characters as many as designated from the left are returned
Syntax	Left(string, length) Left Function Syntax has following factors

Factor	Description
string	A literal expression to be returned from the character at the left end. If there is Null in *string*, Null is returned.
length	A numerical expression showing the character count to be returned. If it is 0, a string whose length is 0("") is returned. Their entire strings are returned if it is equal or more than the character count that is in *string*.

Reference	To decide the character count of string, use Len Function. In the following example, Left Function returns the first three characters of MyString. Dim MyString, LeftString MyString = "VBSCript" LeftString = Left(MyString, 3) 'LeftString gets "VBS." Memo LeftB Function uses Byte data that the string has. It designates Byte count rather than character count to be returned to length factor.

Len Function

Description	The Byte count needed for saving character count or variable that is in the string is returned
Syntax	Len(string | varname) Len Function Syntax has following factors

Factor	Description
string	A literal expression. If there is Null in *string*, Null is returned.
varname	It should be a valid variable name. *Varname* has the value zero, Null is returned.

Reference	In the following example, Len Function returns the character count of a string. Dim MyString MyString = Len("VBSCRIPT") 'MyString gets value 8. Memo LenB Function uses Byte data that the string has. Instead of returning character count to the string, LenB returns Byte count used for expressing this string.

LoadPicture Function

Description	If a picture object is returned, it can be used only on the 32-bit platform
Syntax	LoadPicture(picturename) The picturename factor is a literal expression showing the name of a picture file to be loaded
Reference	LoadPicture Function recognizes the following graphic types; bitemap files(.bmp), icon files(.ico), execution length symbolization files(.rle), meta files(.wmf), improved meta files(.emf), GIF files(.gif) and JPEG files(.jpg).

Log Function

Description	A natural logarithm of a number is returned
Syntax	Log(number) The number factor should be a valid expression more than 0
Reference	A natural logarithm is a log whose base is e. Constant e is about 2.718282. Natural logarithm logn(x) whose base is n and whose value is x can be calculated by dividing natural logarithm logn(x) by natural logarithm logn(n). Logn(x) = Log(x) / Log(n) The following example shows a customized Function that calculates Log Function whose base is 10. Function Log10(X) Log10 = Log(X) / Log(10) End Function

LTrim Function

Description	LTrim Function returns a string copy which has no left empty space or no right empty space or neither left nor right empty space
Syntax	LTrim(string) RTrim(string) Trim(string) The string factor should be a valid character expression. If there is Null in string, Null is returned.
Reference	In the following example, LTrim Function trims the left empty space, RtrimFunction trims the right empty space and Trim Function trims both of them. Dim MyVar MyVar = LTrim(" vbscript ") 'MyVar gets "vbscript ." MyVar = RTrim(" vbscript ") 'MyVar gets " vbscript." MyVar = Trim(" vbscript ") 'MyVar gets "vbscript"."

Mid Function

Description	Charaters as many as designated by string are returned.
Syntax	Mid(string, start[, length]) Mid Function Syntax has following factors

factor	Description
string	A literal expression containing the characters to be returned. If there is Null in *string*, Null is returned.
start	*A character position at which returning starts in the string. If start factor is more than character count of string,* Mid *Function returns a string* ("") *whose length is 0 is returned*.
length	Character count to be returned. If omitted or the actual character count is less than the character count of *length* factor including *start* factor, the entire characters from the start position to the end character of the relevant string shall be returned.

Reference	The character count of a string can be decided by Len Function. In the following example, Mid Function returns six characters starting from the fourth character of a string. Dim MyVar MyVar = Mid("VB Script is fun!", 4, 6) 'MyVar returns "Script."

	Memo MidB Function uses Byte data that the string has. In this Function, Byte count is designated rather than character count.

Minute Function	
Description	An integer indicating the minute from 0 through 59 is returned
Syntax	Minute(time) The time factor is an expression showing time. The time has the value zero, Null is returned
Reference	In the following example, Minute Function returns minute. Dim MyVar MyVar = Minute(Now)

Month Function	
Description	The integer indicating month from 1 through 12 is returned
Syntax	Month(date) The date factor is an expression showing the date. If date has the value zero, Null is returned
Reference	In the following example, Month Function returns the current month. Dim MyVar MyVar = Month (Now) ' MyVar gets a number corresponding to the current month

MonthName Function	
Description	A string showing the designated month is returned
Syntax	MonthName(month[, abbreviate]) MonthName Function Syntax has following factors.
Reference	In the following example, MonthName Function returns the month name to be inputted to the date expression in an abbreviated word type. Dim MyVar MyVar = MonthName(10, True) ' MyVar gets "Oct."

Factor	Description
month	A mandatory factor. Number showing month such as 1 for January and 2 for February, etc.
abbreviate	An optional factor. A boolean value showing whether to display a month name in an abbreviated word type. If omitted, it become False, the Default Value, and shows its full name.

MsgBox Function

Description	A value that shows whether any message is displayed in the dialog box or whether it waits until any text is inputted, or indicates the button pressed by the user is returned

Syntax

MsgBox(prompt[, button][, title][, helpfile, context])

MsgBox Function Syntax has following factors.

Factor	Description
prompt	A literal expression showing message in the dialog box. *prompt* is about 1,024 characters long at the most even though it varies depending on the width of the characters. If *prompt* is long enough to take more than two lines, the lines can be divided by carriage return character (Chr(13)), line feed character(Chr(10)) or a combination of carriage return and line feed characters (Chr(13) & Chr(10)).
button	A numerical expression showing the kind and number of the button to be displayed, icon types to be used, the sum of values designating targets of basic button confirmation and message box. For the value, refer to the following settings. If omitted, 0 appears, the Default Value of the button.
title	A literal expression showing on the line for the title in the dialog box. If *title is* omitted, the application program title appears on the tile line.
helpfile	A literal expression identifying the 'Help' file to provide assistance according to the situation If *helpfile* is provided, it is necessary also to provide *context*. Not available in the 16-bit platform.
context	A numerical expression identifying the Help context number as the relevant Help items designated by the author. If *helpfile* is provided, it is necessary also to provide *context*. Not available in the 16-bit platform.

Settings

The button factor has the following settings.

Constant	Value	Description
vbOKOnly	0	Display OK button only.
vbOKCancel	1	Display OK and Cancel button.
vbAbortRetryIgnore	2	Display Abort, Retry and Ingnore button.
vbYesNoCancel	3	Display Yes, No, and Cancel button.
vbYesNo	4	Display Yes and No button.
vbRetryCancel	5	Display Retry and Cancel button.
vbCritical	16	Display Critical Error Message Icon.
vbQuestion	32	Display Warning Query Icon.
vbExclamation	48	Display Warning Message Icon.

	vbInformation	64	Display Information Message Icon.
	vbDefaultButton1	0	The first button is the default one.
	vbDefaultButton2	256	The second button is the default one.
	vbDefaultButton3	512	The third button is the default one.
	vbDefaultButton4	768	The fourth button is the default one.
	vbApplicationModal	0	An application modal. The user has to answer to the questions in the message box before continuing to work on the current application program.
	vbSystemModal	4096	A system modal. All application programs aborts until the user answer to the questions in the message box.

The first group values (0-5) show the button types and numbers displayed in the dialog box. The second group (16, 32, 48, 64) shows icon types. The third group (0, 256, 512, 768) decides basic buttons and the fourth group (0, 4096) decides the modal of the message box. When the final value of the button factor is obtained by adding values, only one number is used from each group.

Result Value

MsgBox Function returns the following values

Constant	Value	Button
vbOK	1	OK
vbCancel	2	Cancel
vbAbort	3	Abort
vbRetry	4	Retry
vbIgnore	5	Ignore
vbYes	6	Yes
vbNo	7	No

Reference

If both helpfile and context are provide, press F1 key to see Help items for each situation.
If the dialog box has [Cancel] button, pressing Esc key leads to the same results as pressing [Cancel] button. If the dialog box has [Help] button, Help items are provided for each situation. However, one of other buttons is pressed, any value is not returned.

In the following example, MsgBox Function returns the value that displays the message box and explains the selected button.

```
Dim MyVar
MyVar = MsgBox ("Hello?", 65, "MsgBox Example") ' MyVar gets value 1 or 2 according to the selected button
```

Now Function

Description	The current date and time are returned according to the computer's system date and time
Syntax	Now
Reference	In the following example, Now Function returns the current date and time. Dim MyVar MyVar = Now ' MyVar gets the current date and time

Oct Function

Description	An octal value of a number is returned.
Syntax	Oct(number) The number factor should be a valid expression
Reference	Numbers that are not integer are rounded off to the nearest integer before calculating

Value Kind of a number	Result of Oct Function
Null	Null
Empty	0
Numbers other than two value above	Octal characters less than 11

Octal can be expressed by attaching &O before the numbers in a valid range. For example, &O10 shows the octal notation of decimal number 8.

In the following example, Oct Function returns the octal value.

```
Dim MyOct
MyOct = Oct(4)      ' 4 is returned.
MyOct = Oct(8)      ' 10 is returned.
MyOct = Oct(459)    ' 713 is returned
```

Replace Function

Description	A string that replaces the designated sub string with other sub strings is returned
Syntax	Replace(expression, find, replacewith[, start[, count[, compare]]]) Replace Function Syntax has following factors

factor	Description
expression	A mandatory factor. A literal expression containing the sub strings to be replaced.
find	A mandatory factor. A sub string to be found.
replacewith	A mandatory factor. A sub string to be replaced.
start	An optional factor. A position at which searching a sub string starts in the

	expression. If omitted, 1 is deemed. It shall be used along with *count*.
number	An optional factor. Number to replace the sub string. If omitted, Default Value becomes -1 and replaces all of the relevant strings. It shall be used along with *start.*
compare	An optional factor. A number value showing the comparison type of the sub string to be used. For the value, refer to the following settings. If omitted, Default Value becomes 0 and performs binary comparison.

Settings

Comparison factors have the following values

Constant	Value	Description
vbBinaryCompare	0	Run Binary Comparison.
vbTextCompare	1	Run Text Comparison.

Result Value

Replace Function returns the following values

Condition	Result of Replace Function
If it is *expression* whose length is 0	A string whose length is 0("")
If *expression* is Null	Error
If it is *find* whose length is 0	*Expression copy*
If it is *replacewith* whose length is 0	*Omission of the entire sesarches for find and expression copy*
start > Len(*expression*)	A string whose length is 0
If *number* is 0	*expression copy*

Reference

Replace Function starts at the start position and replaces at the end of the expression string, and then, returns the string. That is, the returned string as a result is not the copy of the original strings.

In the following example, Replace Function returns the string.

```
Dim MyString

'If a Binary Comparison starts at the beginning of a string, "XXYXXPXXY" is returned.
MyString = Replace("XXpXXPXXp", "p", "Y")

'If a Text comparison starts at position 3, "YXXYXXY" is returned.
MyString = Replace("XXpXXPXXp", "p", "Y", 3, -1, 1)
```

Right Function

Description	Characters as many as designated from the right are returned
Syntax	Right (string, length) Right Function Syntax has following factors. **Factor** / **Description** *string* — A literal expression to be returned from the right end character. If there is Null in *string*, Null is returned. *length* — A numerical expression showing the character count to be returned. If it is 0, a string whose length is 0("") is returned. Their entire strings are returned if it is equal or more than the character count that is in *string*.
Reference	To decide the character count of string, use Len Function. In the following example, Right Function returns characters as many as designated from the right side of the string. Dim AnyString, MyStr AnyString = "Hello?" 'Designate a string. MyStr = Right(AnyString, 1) ' "?" is returned. MyStr = Right(AnyString, 6) ' "llo?" is returned. MyStr = Right(AnyString, 20) ' "Hello?" is returned. Memo RightB Function uses Byte data that the string has. It designates Byte count rather than character count to be returned to length factor.

Rnd Function

Description	A random number is returned
Syntax	Rnd[(number)] The number factor should be a valid expression
Reference	Rnd Function returns a value that is bigger than 0 and smaller than 1. The number value decides how Rnd Function generates random numbers **Number Value** / **Result of Rnd Function** < 0 — Every time the same number by seed *number* > 0 — Next random number created sequentially. = 0 — The latest random number Omitted — Next random number created sequentially. In case of any given initial seed, numbers in the same sequence are generated because

	previous numbers are used as seeds to have next sequential numbers in continual calls for Rnd Function at each time. Before calling Rnd, use Randomize statement and initialize the random number generator with seed based on the system timer. To make random integer in a given range, use the following formula: Int((upperbound - lowerbound + 1) * Rnd + lowerbound) upperbound refers to the biggest number and lowerbound refers to the smallest number in the range. Memo To repeat the sequence of generating random numbers, call Rnd Function with negative factors right before using Randomize with number factors. To use Randomize with the same value, the previous sequence is not repeated.

RGB Function

Description	The integers showing RGB color values are returned.
Syntax	RGB(red, green, blue) RGB Function has following factors

factor	Description
red	A mandatory factor. Numbers 0 through 255 indicating red color elements.
green	A mandatory factor. Numbers 0 through 255 indicating green color elements.
blue	A mandatory factor. Numbers 0 through 255 indicating blue color elements.

Reference

Numbers showing RGB color values are necessary for methods and properties of the application program receiving color designation. RGB color values designate relative concentration of red, green and blue and shows particular colors.
Lower Byte contains red value, middle byte contains green value and upper byte contains blue value.

Even though the following Function shows the reverse byte order, it provides the same information to the application programs that need that sequence.

```
Function RevRGB(red, green, blue)
    RevRGB= CLng(blue + (green * 256) + (red * 65536))
End Function
```

For RGB Function's factor values that exceed 255, they are deemed as 255

Round Function

Description	The number rounded off at the designated decimal place is returned
Syntax	Round(expression[, numdecimalplaces]) Round Function Syntax has following factors

Factor	Description
expression	A mandatory factor. A numerical expression to be rounded off.
numdecimalplaces	An optional factor. A number showing decimal place. If omitted, Round Function returns integers.

Reference	In the following example, Round Function rounds off a number two-digit decimal. Dim MyVar, pi pi = 3.14159 MyVar = Round(pi, 2) 'MyVar gets 3.14

RTrim Function

Description	LTrim Function returns a string copy which has no left empty space or no right empty space or neither left nor right empty space.
Syntax	LTrim(string) RTrim(string) Trim(string) The string factor should be a valid character expression. If there is Null in string, Null is returned
Reference	In the following example, LTrim Function trims the left empty space, Rtrim Function trims the right empty space and Trim Function trims both of them. Dim MyVar MyVar = LTrim(" vbscript ") 'MyVar gets "vbscript ". MyVar = RTrim(" vbscript ") 'MyVar gets " vbscript". MyVar = Trim(" vbscript ") 'MyVar gets "vbscript".

ScriptEngine Function

Description	A string showing the script language in use is returned
Syntax	Result Value ScriptEngine Function returns the following strings

String	Description
VBScript	It shows that Microsoft(R) Visual Basic(R) Scripting Edition is the current script engine.

	JScript — It shows that Microsoft JScript(R) is the current script engine *VBA* — It shows that Microsoft Visual Basic for Applications is the current script engine
Reference	In the following example, ScriptEngine Function returns the script language in use. Function GetScriptEngineInfo Dim s s = "" ' Script a string with necessary information. s = ScriptEngine & “Version” s = s & ScriptEngineMajorVersion & "." s = s & ScriptEngineMinorVersion & "." s = s & ScriptEngineBuildVersion GetScriptEngineInfo = s 'Result is returned. End Function

ScriptEngineBuildVersion Function	
Description	The configuration version number of the script language in use is returned
Syntax	ScriptEngineBuildVersion
Reference	The returned Result Value is equal to the version information contained in the DLL of the script language in use. In the following example, ScriptEngineBuildVersion Function returns the configuration version number of the script engine. Function GetScriptEngineInfo Dim s s = "" 'Script a string with essential information. s = ScriptEngine & “Version “ s = s & ScriptEngineMajorVersion & "." s = s & ScriptEngineMinorVersion & "." s = s & ScriptEngineBuildVersion GetScriptEngineInfo = s ‘Result is returned. End Function

ScriptEngineMajorVersion Function	
Description	The major version number of the script language in use is returned
Syntax	ScriptEngineMajorVersion
Reference	The returned Result Value is equal to the version information contained in the DLL of the script language in use. In the following example, ScriptEngineMajorVersion Function returns the version number of the

	script engine. Function GetScriptEngineInfo Dim s s = "" 'Script a string with essential information. s = ScriptEngine & " Version " s = s & ScriptEngineMajorVersion & "." s = s & ScriptEngineMinorVersion & "." s = s & ScriptEngineBuildVersion GetScriptEngineInfo = s 'Result is returned. End Function

ScriptEngineMinorVersion Function	
Description	The minor version number of the script language in use is returned
Syntax	ScriptEngineMinorVersion
Reference	The returned Result Value is equal to the version information contained in the DLL of the script language in use. In the following example, SScriptEngineMinorVersion Function returns the minor version number of the script engine. Function GetScriptEngineInfo Dim s s = "" ' Script a string with necessary information. s = ScriptEngine & " Version " s = s & ScriptEngineMajorVersion & "." s = s & ScriptEngineMinorVersion & "." s = s & ScriptEngineBuildVersion GetScriptEngineInfo = s 'Result is returned. End Function

Second Function	
Description	An integer indicating the second from 0 through 59 is returned
Syntax	Second(time) The time factor is an expression showing time. If the time factor has the value zero, Null is returned.
Reference	In the following example, Second Function returns the current second. Dim MySec MySec = Second(Now) 'MySec has information on the current second.

Sgn Function	
Description	An integer showing number sign is returned
Syntax	Sgn(number) The number factor should be a valid expression. Result Value Sgn Function returns the following values. Number Value Result of Sgn Function > 0 , 1 = 0, 0 < 0 -1
Reference	The sign of a number factor decides the result value of Sgn Function. In the following example, Sgn Function decides number signs. Dim MyVar1, MyVar2, MyVar3, MySign MyVar1 = 12: MyVar2 = -2.4: MyVar3 = 0 MySign = Sgn(MyVar1) ' 1 is returned. MySign = Sgn(MyVar2) ' -1 is returned. MySign = Sgn(MyVar3) ' 0 is returned.

Sin Function	
Description	The sine value of an angle is returned
Syntax	Sin(number) The number factor should be a valid expression showing the angle in a radian
Reference	Sin Function returns the ratio between lengths of two sides by using alternate angles of a right triangle. This ratio is the one obtained by dividing the length of the count side to the relevant angle by the length of the hypotenuse of a right triangle. The returned results are ranged between -1 and 1. To convert angle to radian, multiply angle by pi/180 and to convert radian to angle, multiply radian by 180/pi. In the following example, Sin Function returns the tangent value of the angle. Dim MyAngle, MyCosecant MyAngle = 1.3 ' Define an angle in a radian value. MyCosecant = 1 / Sin(MyAngle) ' Calculate cosine value.

Space Function

Description	The string that contains empty space as many as designated is returned
Syntax	Space(number) The number factor is a number of spaces to be filled in the string with.
Reference	In the following example, Space Function returns a string that contains spaces as many as designated. Dim MyString MyString = Space(10) ' A string that has 10 spaces is returned. MyString = "He" & Space(10) & "low?" ' 10 spaces are inserted between strings

Split Function

Description	One dimension Array based on 0, which contains sub strings as many as designated is returned.
Syntax	Split (expression[, delimiter[, number[, compare]]]) Split Function Syntax has following factors.

Factor	Description
expression	A mandatory factor. A literal expression containing sub strings and distinguishing symbols. If the length of *expression* is 0, Split returns empty Array, that is, Array that has no configuration element and data.
delimiter	An optional factor. A string character used to identify the limit of a sub string. If omitted, empty character (" ") is deemed as a distinguishing symbol. *Delimiter* is 0 long, a single element containing *expression* strings is returned.
number	An optional factor. A number of a sub string to be returned. -1 shows that all of sub strings are to be returned.
compare	An optional factor. A number value showing the comparison type of the sub string to be used. For the value, refer to the following settings.

Settings

Comparison factors have the following values

Constant	Value	Description
vbBinaryCompare	0	Run Binary Comparison.
vbTextCompare	1	Run Text Comparison.

Reference	In the following example, Split Function returns the string. This Function executes text comparison of distinguishing characters and returns all of sub strings.

	Dim MyString, MyArray MyString = Split("VBScriptXisXfun!", "x", -1, 1) ' MyString(0) gets "VBScript". ' MyString(1) gets "is" ' MyString(2) gets value"fun!"

Sqr Function

Description	The square root of a number is returned
Syntax	Sqr(number) The number factor is a valid expression that is equal or higher than 0.
Reference	In the following example, Sqr Function calculates the square root of a number. Dim MySqr MySqr = Sqr(4) ' 2 is returned. MySqr = Sqr(23) ' 4.79583152331272 is returned. MySqr = Sqr(0) ' 0 is returned. MySqr = Sqr(-4) ' It causes runtime errors

StrComp Function

Description	The result value of comparing strings is returned
Syntax	StrComp(string1, string2[, compare]) StrComp Function Syntax has following factors.

factor	Description
string1	A mandatory factor. A valid character expression.
string2	A mandatory factor. A valid character expression.
compare	An optional factor. A number value displayed in a comparison type to be used for assessing strings. If omitted, Run Binary Comparison. For the value, refer to the following settings.

Settings

Comparison factors have the following values.

Constant	Value	Description
vbBinaryCompare	0	Run Binary Comparison.
vbTextCompare	1	Run Text Comparison.

Result Value

StrComp Function returns the following values

	<table><tr><th>Condition</th><th>Result of StrComp Function</th></tr><tr><td>*If string1 is less than string2*</td><td>-1</td></tr><tr><td>*If string1 is equal to string2*</td><td>0</td></tr><tr><td>*If string1 is more than string2*</td><td>1</td></tr><tr><td>*If string1 or string2 is* Null</td><td>Null</td></tr></table>
Reference	In the following example, StrComp Function returns the result of comparing strings. If the third factor is 1, a text comparison is executed. If the third factor is 0 or omitted, a binary Comparison is performed. Dim MyStr1, MyStr2, MyComp MyStr1 = "ABCD": MyStr2 = "abcd" ' Define a variable. MyComp = StrComp(MyStr1, MyStr2, 1) ' 0 is returned. MyComp = StrComp(MyStr1, MyStr2, 0) ' -1 is returned. MyComp = StrComp(MyStr2, MyStr1) ' 1 is returned.

String Function	
Description	The expression converted to a String subtype of Variant is returned
Syntax	CStr(expression) The expression factor should be a valid expression
Reference	Typically, if a code is created with a subtype of conversion function, the operation result can be seen in a particular data type rather than in a basic data type. For example, if CStr is used, the result is expressed in a String. If CStr Function is used instead of Str Function, other types of data can be converted to a widely-used String subtype throughout the world. For example, symbols identified with different digit numbers are properly recognized according to the computer's locale settings. The data used in an expression decide the returned results as seen in the table. <table><tr><th>*Expression Data*</th><th>CStr Returned Result</th></tr><tr><td>Boolean</td><td>String containing True or False</td></tr><tr><td>Date</td><td>String containing simplified date by computer settings</td></tr><tr><td>Null</td><td>Runtime error</td></tr><tr><td>Empty</td><td>String whose length is 0 ("")</td></tr><tr><td>Error</td><td>String showing error number next to word 'Error'</td></tr><tr><td>Other Numbers</td><td>String containing numbers</td></tr></table> In the following example, CStr Function converts the value to String.

	Dim MyDouble, MyString MyDouble = 437.324 ' MyDouble is equal to Double. MyString = CStr(MyDouble) ' MyString gets value"437.324"

StrReverse Function	
Description	The reverse string of a designated string is returned
Syntax	StrReverse(string1) The string1 factor is a string of characters in the reverse order. If string1 is a string whose length is 0(""), it returns a string whose length is 0 while if it is Null, errors takes place
Reference	In the following example, Replace Function returns a string in the reverse order. Dim MyStr MyStr = StrReverse("VBScript") 'MyStr gets "tpircSBV."

Tan Function	
Description	The tangent value of an angle is returned
Syntax	Tan(number) The number factor should be a valid expression showing the angle in a radian.
Reference	Tan Function returns the ratio between lengths of two sides by using angles of a right triangle. This ratio is obtained by dividing the length of the side facing the relevant angle by the length of the side adjacent to this angle. To convert angle to radian, multiply angle by pi/180 and to convert radian to angle, multiply radian by 180/pi. In the following example, Tan Function returns the tangent value of the angle. Dim MyAngle, MyCotangent MyAngle = 1.3 ' Define an angle in a radian value. MyCotangent = 1 / Tan(MyAngle) ' Calculate cotangent Value

Time Function	
Description	A Date subtype Variant showing the current system time is returned.
Syntax	Time
Reference	In the following example, Time Function returns the current system time. Dim MyTime MyTime = Time ' The current system time is returned.

TimeSerial Function

Description	The Date subtype of Variant of the designated year, month, day is returned
Syntax	TimeSerial(hour, minute, second) TimeSerial Function Syntax has following factors

Factor	Description
time	Numbers (Including 0 and 23) between 0(AM 12:00) and 23(PM 11:00) or a numerical expression
minute	All expressions
second	All expressions

Reference

To designate time such as 11:59: 59, each TimeSerial factor number should be within the tolerance range of the relevant factor. That is, it should be between 0 and 23 for time and between 0 and 59 for minute or second. However, it is also possible to designate relative time of each factor by using a numerical expression showing hour, minute and second before and after a particular time.

The following example is using an expression rather than an absolute time. Here, TimeSerial Function returns the time 15 minutes before 6o'clock, that is AM 5:45.

```
TimeSerial(12 - 6, -15, 0)
```

If the factor exceeds the tolerance range, it goes to the next bigger unit. For example, if 75 minutes are designated, it is calculated into 1:15. But, if a certain factor exceeds range -32,768~32 and 767 or a date designated directly or through an expression by using three factors, errors take place

TimeValue Function

Description	The Date subtype of Variant that contains time is returned
Syntax	TimeValue(time) Typically, it is the time factor as a literal expression showing time from 0:00:00 (AM 12:00:00) to 23:59:59(PM 11:59:59), therefore, an expression showing time within this range can be a time factor. The time has the value zero, Null is returned.
Reference	A valid time unit can be inputted to the 12-h or 24-h clock. That is, "2:24PM" and "14:24" are all valid time factors. If the time factor contains time information, TimeValue Value Function does not return the information. However, if the time factor has wrong date information, errors takes place. In the following example, TimeValue Function converts a string to time. If the date literal is used, it is possible to directly designate time to variant (Ex, MyTime = #4:35:17 PM#). Dim MyTime MyTime = TimeValue("4:35:17 PM") ' MyTime gets value"4:35:17 PM"

Trim Function	
Description	LTrim Function returns a string copy which has no left empty space or no right empty space or neither left nor right empty space
Syntax	LTrim(string) RTrim(string) Trim(string) The string factor should be a valid character expression. If there is Null in string, Null is returned
Reference	In the following example, LTrim Function trims the left empty space, RtrimFunction trims the right empty space and Trim Function trims both of them. Dim MyVar MyVar = LTrim(" vbscript ") 'MyVar gets "vbscript." MyVar = RTrim(" vbscript ") 'MyVar gets " vbscript." MyVar = Trim(" vbscript ") 'MyVar gets "vbscript."

TypeName Function	
Description	A string that provides Variant subtype information on variable is returned
Syntax	TypeName(varname) A mandatory factor. The varname factor may be any kind of variable Result Value TypeName Function returns the following values

Value	Description
Byte	Byte Value
Integer	Integer Value
Long	Long integer Value
Single	Single precision floating point number value
Double	Double precision floating point number value
Currency	Call Value
Decimal	Decimal Value
Date	Date or Time Value
String	String Value
Boolean	Boolean Value True or False
Empty	Not initialized
Null	No valid data
<object type>	Real type name of object
Object	General object

	<table><tr><td>Unknown</td><td>Unknown object type</td></tr><tr><td>Nothing</td><td>Object variable that does not refer to object instance</td></tr><tr><td>Error</td><td>Error</td></tr></table>
Reference	In the following example, TypeName Function returns variable reference information. Dim ArrayVar(4), MyType NullVar = Null ' Null Designate a value. MyType = TypeName("VBScript") ' "String" is returned. MyType = TypeName(4) ' "Integer" is returned. MyType = TypeName(37.50) ' "Double" is returned. MyType = TypeName(NullVar) ' "Null" is returned. MyType = TypeName(ArrayVar) ' "Variant()" is returned

UBound Function

Description	The biggest subscript available in the designated Array level is returned.
Syntax	UBound(arrayname[, dimension]) UBound Function Syntax has following factors. <table><tr><th>Factor</th><th>Description</th></tr><tr><td>*arrayname*</td><td>A mandatory factor. Comply with the notation rule of general variable with Array variable name</td></tr><tr><td>*dimension*</td><td>An optional factor. An integer showing the upper limit of the returned level is returned. Use integers 1 for one dimension, 2 for two dimension, etc. If *dimension* is omitted, 1 is deemed.</td></tr></table>
Reference	LBound Function is used along with UBound Function to decide the size of Array. If Lbound Function is used, the lower limit can be known in the Array level. The lower limit of the entire dimensions is always 0. Therefore, UBound Function returns Array values as following according to each dimension. Dim A(100,3,4) <table><tr><th>Statement</th><th>Result Value</th></tr><tr><td>UBound(A, 1)</td><td>100</td></tr><tr><td>UBound(A, 2)</td><td>3</td></tr><tr><td>UBound(A, 3)</td><td>4</td></tr></table>

UCase Function

Description	A string converted to decimal point is returned
Syntax	UCase(string) The string factor should be a valid character expression. If there is Null in string, Null is returned
Reference	This function converts only capital letters to small letters and remains other characters except those as they are without converting. In the following example, UCase Function returns the capital version of a string. Dim MyWord MyWord = UCase("Hello World") ' "HELLO WORLD" is returned

VarType Function

Description	The value showing variable subtype is returned
Syntax	VarType(varname) The varname factor may be any kind of variable Result Value VarType Function returns the following values

Constant	Value	Description
vbEmpty	0	Empty(Not initialized)
vbNull	1	Null(No valid data)
vbInteger	2	Integer
vbLong	3	Long integer
vbSingle	4	Single precision floating point number
vbDouble	5	Double precision floating point number
vbCurrency	6	Call
vbDate	7	Date
vbString	8	String
vbObject	9	Automatic Object
vbError	10	Error
vbBoolean	11	Boolean Value
vbVariant	12	Variant(used only in variant Array)
vbDataObject	13	Data Access Object
vbByte	17	Byte
vbArray	8192	Array

Memo

It is possible to use those names in any place instead of actual values since they are the

	Constant designated in the VBScript
Reference	VarType Function itself never returns the relevant Array value and shows a particular type of Array by being combined with other values at all times. The value corresponding to Variant is returned only when it shows that VarType Function is Array by being combined with a particular value. For example, the integer value returned to Array is calculated to 2+8192 or 8194. If an object that has basic properties, VarType (object) Function returns the basic properties of the relevant object. In the following example, VarType Function decides variable's subtype. Dim MyCheck MyCheck = VarType(300) ' 2 is returned. MyCheck = VarType(#10/19/62#) ' 7 is returned. MyCheck = VarType("VBScript") ' 8 is returned.

Weekday Function

Description	An integer showing weekday is returned
Syntax	Weekday(date, [firstdayofweek]) Weekday Function Syntax has following factors.

Factor	Description
date	An expression showing Date. If there is date Null in *date*, Null is returned.
firstdayofweek	Constant that designates the first weekday. If omitted, vbSunday is deemed.

Settings

The firstdayofweek factor has the following settings

Constant	Value	Description
vbUseSystem	0	NLS(National Language Support). Use API settings
vbSunday	1	Sunday
vbMonday	2	Monday
vbTuesday	3	Tuesday
vbWednesday	4	Wednesday
vbThursday	5	Thursday
vbFriday	6	Friday
vbSaturday	7	Saturday

Result Value

Typically, Weekday Function returns the following values

Constant	Value	Description
vbSunday	1	Sunday
vbMonday	2	Monday
vbTuesday	3	Tuesday
vbWednesday	4	Wednesday
vbThursday	5	Thursday
vbFriday	6	Friday
vbSaturday	7	Saturday

Reference

In the following example, Weekday Function brings a weekday from the designated date

```
Dim MyDate, MyWeekDay
MyDate = #October 19, 1962#       ' Designate date.
MyWeekDay = Weekday(MyDate)       ' MyWeekDay gets value 6 because MyDate is on Friday
```

WeekdayName Function

Description: A string showing the designated weekday is returned

Syntax

WeekDayName(weekday, abbreviate, firstdayofweek)

WeekDayName Function Syntax has following factors

Factor	Description
weekday	A mandatory factor. A number showing weekday. Each weekday has a different value according to *firstdayofweek* settings.
abbreviate	An optional factor. A boolean value showing whether weekday names are expressed with abbreviated words. If omitted, it becomes False, the Default Value, full names are displayed.
firstdayofweek	An optional factor. A number value showing the first weekday. For the value, refer to the following settings.

Settings

The value of the firstdayofweek factor is as follows

Constant	Value	Description
vbUseSystem	0	NLS(National Language Support). Use API settings
vbSunday	1	Sunday(Default Value)
vbMonday	2	Monday
vbTuesday	3	Tuesday
vbWednesday	4	Wednesday
vbThursday	5	Thursday
vbFriday	6	Friday
vbSaturday	7	Saturday

Reference	In the following example, WeekDayName Function returns the designated day. Dim MyDate MyDate = WeekDayName(6, True) 'MyDate gets Fri

Year Function	
Description	An integer showing year is returned
Syntax	Year(date) The date factor is an expression showing the date. If the date factor has the value zero, Null is returned

Reference	In the following example, YearFunction brings the date of the year from the designated date
	
	Dim MyDate, MyYear
	MyDate = #October 19, 1962# ' Designate date.
	MyYear = Year(MyDate) ' MyYear gets 1962

Appendix 5 Function

5.1 Function Classification

5.1.1 List by Item

Item	Relevant Function
Project	LONG **CloseProject** ()
	BSTR **GetProjectName** ()
	BSTR **GetProjectPath** ()
	BSTR **GetProjectType** ()
	LONG **OpenProject** (BSTR szProjectName, BSTR szProjectPath)
	LONG **QuitProject** ()
Screen	LONG **ChangePage** (BSTR szOldPage, BSTR szNewPage)
	LONG **ClosePage** (BSTR szPageName)
	VOID **GetHWND** (ULONG* hwnd)
	VOID **HardCopy** ()
	BOOL **IsRunning** ()
	VOID **OpenPage** (BSTR szPageName, BSTR szPageParam)
	LONG **OpenPageObject** (BSTR szPageName, BSTR szObjectName)
	VOID **Quit** ()
Tag	ULONG **GetTagStatus** (BSTR szTagName)
	BSTR **GetTagStatusDesc** (ULONG lTagStatus)
	VARIANT **GetTagVal** (BSTR szTagName)
	VARIANT **GetTagValbyTagId** (LONG lTagId)
	BSTR **GetTagValnStatusByTagId** (LONG lTagId, VARIANT* vtTagVal, LONG *lScanMode, LONG *lMainQ, LONG *lSubQCnt, LONG *lSubQ1, LONG *lSubQ2, LONG *lSubQ3, LONG *lSubQ4, LONG *lSubQ5)
	LONG **GetTagValueStatus** (BSTR szTagName, VARIANT* vtTagVal)
	LONG **ResetStaticValue** (BSTR szTagName)
	VOID **ShowCrossReference** (BSTR szTagName)
	VOID **ShowRawDataInfo** ()
	VOID **ShowSIMCfgInfo** ()
	VOID **SetTagVal** (BSTR szTagName, VARIANT vtTagVal)
	LONG **SetTagValEx** (SHORT nMode, BSTR szTagName, VARIANT vtTagVal, BSTR szClientName, ULONG hClientHandle)
Script	VOID **RunScript** (BSTR szScriptName)
	VOID **StopScript** (BSTR szScriptName)
	VOID **TimeOutMsgBox** (BSTR Msg, BSTR Title, LONG lSec)

Item	Relevant Function
Event	LONG **NotifyAlarmEvent** (DOUBLE llOccur, USHORT nAlmLevel, BSTR szMessage, BSTR szArea, BSTR szOUser, BSTR zInformation, BSTR szUserDefine1, BSTR szUserDefine2, BSTR szUserDefine3, BSTR szUserDefine4, BSTR szPrev, BSTR szCur, BSTR szObject)
	VOID **SendEmail** (BSTR szMailAddTo, BSTR szMailAddFrom, BSTR szMailTitle , BSTR szMailMsg)
	VOID **SendSMS** (BSTR szRecvPhoneNo, BSTR szCallBackPhoneNo, BSTR szSMSMsg)
Log	VOID **NotifyError** (USHORT nLogCategory, BSTR szMsg, BSTR szDesc)
	VOID **NotifyHostError** (BSTR szHostName, USHORT nLogCategory, BSTR szMsg, BSTR szDesc)
	VOID **NotifyHostInfo** (BSTR szHostName, USHORT nLogCategory, BSTR szMsg, BSTR szDesc)
	VOID **NotifyHostLog** (USHORT nLogType, USHORT nLogCategory, BSTR szHostName, BSTR szMsg, BSTR szDesc)
	VOID **NotifyHostWarn** (BSTR szHostName, USHORT nLogCategory, BSTR szMsg, BSTR szDesc)
	VOID **NotifyInfo** USHORT nLogCategory (USHORT nLogCategory, BSTR szMsg, BSTR szDesc)
	VOID **NotifyLog** (USHORT nLogType, USHORT nLogCategory, BSTR szMsg, BSTR szDesc)
	VOID **NotifyWarn** (USHORT nLogCategory, BSTR szMsg, BSTR szDesc)
Network	BSTR **ChangLine** (BSTR szLineMode, BSTR szTargetLineIP)
	LONG **ChangeProcessServer** (BSTR szProcessName, BSTR szProcessIP)
	LONG **ChangeServerMode** (BSTR szServerMode)
	LONG **GetCurLine** (int* nCurLine)
	INT **GetLineStatus** (int nLineMode)
	BSTR **GetNodeCurIP** (BSTR szNodeName)
	BSTR **GetServerMode** ()
	BOOL **IsDualLine** ()
System	VARIANT **GetSystemTime** ()
	VOID **ProcessSleep** (LONG dwMiliseconds)
	VOID **RunApp** (BSTR szAppPath)
	VOID **ShowAlarmPage** (BSTR szAlarmTag)
	VOID **Sleep** (LONG dwMiliseconds)
	BSTR **TimeConvert** (DOUBLE dTimeValue, INT nTimeType)
Database	VOID **DataExportRunWork** (BSTR szWorkName)

Item	Relevant Function
Recipe	LONG **RecipeChangeData** (BSTR ModelName, BSTR RecipeName)
	LONG **RecipeChangeUnit** (BSTR ModelName, BSTR UnitName)
	LONG **RecipeControlSet** (BSTR szModelName, BSTR szUnitName, BSTR szDataName)
	LONG **RecipeControlSetEx** (SHORT nMode, BSTR szClientProgName, ULONG hClientHandle, BSTR szModelName, BSTR szUnitName, BSTR szDataName, LONG* plTransNo, SHORT* pnResult)
	LONG **RecipeControlWorkingSet** ();
	LONG **RecipeControlWorkingSetEx** (SHORT nMode, BSTR szClientProgName, ULONG hClientHandle, LONG* plTransNo, SHORT* pnResult);
	LONG **RecipeDeleteData** (BSTR ModelName, BSTR RecipeName)
	LONG **RecipeDeleteModel** (BSTR ModelName)
	LONG **RecipeDeleteUnit** (BSTR ModelName, BSTR UnitName)
	LONG **RecipeLoadDynamic** (BSTR ModelName, BSTR UnitName)
	LONG **RecipeLoadSet** (BSTR ModelName, BSTR UnitName, BSTR RecipeName)
	LONG **RecipeSaveAsArchive** (BSTR ModelName, BSTR RecipeName)
	LONG **RecipeSetData** (BSTR ModelName, BSTR ItemName, VARIANT setValue)
Externel Device	VOID **PlaySound**(BSTR szWaveFile)
	VOID **PlaySoundFile** (BSTR szSoundFile)
User	BSTR **GetUserID** ()
	LONG **GetUserLevel** ()
	LONG **UserLogOff** ()
	LONG **UserLogOn** (BSTR szUserID, BSTR szPassWord)
Report	VOID **ReportDialog** ()
	VOID **ReportListRefresh** ()
	VOID **ReportPrint** (BSTR szReportName)
	VOID **ReportPrintEx** (BSTR szReportName, BSTR szReqTime)
	VOID **ReportSchedulerHide** ()
	VOID **ReportSchedulerShow** ()

Item	Relevant Function
Alarm	BOOL **AckAllAlarm** ()
	BSTR **GetAlarmCountErrInfo** (LONG lCount)
	LONG **GetAlarmLogging** (DOUBLE dtFrom, DOUBLE dtTo, BSTR szTag, BSTR szAlarmGr, LONG lAlarmType, LONG *lOutCount, SAFEARRAY **pszRstrTag, SAFEARRAY **pszRstrGr, SAFEARRAY **pdbRdtOccAlarm, SAFEARRAY **pdbRdtRevAlarm, SAFEARRAY **pdbRdtAckAlarm, SAFEARRAY **plRnAlarmType, SAFEARRAY **pszRstrMsg, SAFEARRAY **pszRstrAlarmKind)
	LONG **GetAlarmLoggingEx** (DOUBLE dtFrom,DOUBLE dtTo,BSTR szTag, LONG nAlarmGroupCount, SAFEARRAY **pszAlarmGr,LONG lAlarmType, LONG lAlarmGrade, BSTR szAlarmKind, LONG *lOutCount, SAFEARRAY**pRstrTag,SAFEARRAY**pRstrGr, SAFEARRAY **pdbRdtOccAlarm, SAFEARRAY **pdbRdtRevAlarm, SAFEARRAY **pdbRdtAckAlarm, SAFEARRAY **plRnAlarmType, SAFEARRAY **pszRstrMsg, SAFEARRAY **pszRstrAlarmKind, SAFEARRAY **pdbRvOccDValue, SAFEARRAY**pszRvOccSValue, SAFEARRAY **plAlarmGrade)
	LONG **GetAlmCount** (LONG lkind, LONG lStatus)
	LONG **GetGroupUserInfoByTag** (BSTR szTagName, LONG* , SAFEARRAY(BSTR) *pszUserID, SAFEARRAY(BSTR) *pszUserName, SAFEARRAY(LONG) *plMailCheck, SAFEARRAY(LONG) *plSMSMCheck, SAFEARRAY(BSTR) *pszMailAdd, SAFEARRAY(BSTR) *pszHPhoneNo)
	LONG **GetHistAlmCount** (LONG lkind, LONG lStatus)
	LONG **GetUserInfoList** (LONG* lOutCount, SAFEARRAY(BSTR) *pszUserID, SAFEARRAY(BSTR) *pszUserName, SAFEARRAY(BSTR) *pszDesc, SAFEARRAY(BSTR) *pszPassword, SAFEARRAY(LONG) *plUserLevel, SAFEARRAY(BSTR) *pszHPhoneNo, SAFEARRAY(BSTR) *pszPhoneNo, SAFEARRAY(BSTR) *pszMailAdd)
	LONG **SetUserAlarm** (BSTR szAlarmMessage, BSTR szAlarmGroup)
	LONG **SetUserAlarmEx** (DOUBLE dOccur, LONG lAlarmLevel, BSTR szAlarmMessage ,BSTR szAlarmGroup, DOUBLE dCur, DOUBLE dPrev)
	LONG **SetUserEvent** (BSTR szEventMessage)
	VOID **StopAlarmSound** ()

Item	Relevant Function
History Data	VOID **GetHistDataCount** (BSTR szModelName, BSTR , BSTR szDataKind, DOUBLE dRefTime, LONG lCount, LONG lDir, LONG* lResultCount, SAFEARRAY(double) *psadblTimeStamp, SAFEARRAY(double) *psadblNumValue, SAFEARRAY(LONG) *psalStatus, SAFEARRAY(LONG) *psalResult)
	VOID **GetHistDataPeriod** (BSTR szModelName, BSTR szTagName, BSTR szDataKind, DOUBLE dFrom, DOUBLE dTo, LONG* lResultCount, SAFEARRAY(double) *psadblTimeStamp, SAFEARRAY(double) *psadblNumValue, SAFEARRAY(LONG) *psalStatus, SAFEARRAY(LONG) *psalResul)
	LONG **GetLoggingModelList** (LONG *lOutCount ,SAFEARRAY(BSTR) *psaszModelName, SAFEARRAY(BSTR) *psaszModelDesc, SAFEARRAY(BSTR) *psaszModelType, SAFEARRAY(BSTR) *psaszArchvingCycle, SAFEARRAY(BSTR) *psaszSegmentPeriod)
	LONG **GetLoggingTag** (BSTR szModelName, LONG *lOutCount ,SAFEARRAY(BSTR) *psaszTagName, SAFEARRAY(BSTR) *psaszTagType)
	LONG **SetHistData** (BSTR szModelName, BSTR szTagName, BSTR szDataKind, LONG lDataCount, SAFEARRAY(double) *psadblSaveTime, SAFEARRAY(double) *psadblNumValue, SAFEARRAY(LONG) *psalTagStatus, SAFEARRAY(LONG) *psalResult)
I/O Driver	VOID **ClearStaStatusCount** (BSTR szChannelName, BSTR szStaName)
	VOID **EnableStation** (BSTR szChannelName, BSTR szStaName, LONG lMode)
	LONG **GetCountReadSuccess** (BSTR szChannelName, BSTR szStaName)
	LONG **GetCountReadTotal** (BSTR szChannelName, BSTR szStaName)
	LONG **GetCountWriteSuccess** (BSTR szChannelName, BSTR szStaName)
	LONG **GetCountWriteTotal** (BSTR szChannelName, BSTR szStaName)
	LONG **GetStatusRunMode** (BSTR szChannelName, BSTR szStaName)
	LONG **GetStatusStaOK** (BSTR szChannelName, BSTR szStaName)
	VOID **ShowNetworkStatus** ()
	VOID **ShowStationInfo** ()

5.1.2 List by Alphabetic Order

Alphabet	Function Name	Item
A	BOOL **AckAllAlarm** ()	Alarm
C	BSTR **ChangLine** (BSTR szLineMode, BSTR szTargetLineIP)	Network
	LONG **ChangeProcessServer** (BSTR szProcessName, BSTR szProcessIP)	Network
	LONG **ChangeServerMode** (BSTR szServerMode)	Network
	LONG **ChangePage** (BSTR szOldPage, BSTR szNewPage)	Screen
	VOID **ClearStaStatusCount** (BSTR szChannelName, BSTR szStaName)	I/O Driver
	LONG **ClosePage** (BSTR szPageName)	Screen
	LONG **CloseProject** ()	Project
D	VOID **DataExportRunWork** (BSTR szWorkName)	Database
E	VOID **EnableStation** (BSTR szChannelName, BSTR szStaName, LONG lMode)	I/O Driver
G	LONG **GetAlmCount** (LONG lkind, LONG lStatus)	Alarm
	BSTR **GetAlarmCountErrInfo** (LONG lCount)	Alarm
	LONG **GetAlarmLogging** (DOUBLE dtFrom, DOUBLE dtTo, BSTR szTag, BSTR szAlarmGr, LONG lAlarmType, LONG *lOutCount, SAFEARRAY **pszRstrTag, SAFEARRAY **pszRstrGr, SAFEARRAY **pdbRdtOccAlarm, SAFEARRAY **pdbRdtRevAlarm, SAFEARRAY **pdbRdtAckAlarm, SAFEARRAY **plRnAlarmType, SAFEARRAY **pszRstrMsg, SAFEARRAY **pszRstrAlarmKind)	Alarm
	LONG **GetAlarmLoggingEx** (DOUBLE dtFrom,DOUBLE dtTo,BSTR szTag, LONG nAlarmGroupCount, SAFEARRAY **pszAlarmGr,LONG lAlarmType, LONG lAlarmGrade, BSTR szAlarmKind, LONG *lOutCount, SAFEARRAY**pRstrTag,SAFEARRAY**pRstrGr, SAFEARRAY **pdbRdtOccAlarm, SAFEARRAY **pdbRdtRevAlarm, SAFEARRAY **pdbRdtAckAlarm, SAFEARRAY **plRnAlarmType, SAFEARRAY **pszRstrMsg, SAFEARRAY **pszRstrAlarmKind, SAFEARRAY **pdbRvOccDValue, SAFEARRAY**pszRvOccSValue, SAFEARRAY **plAlarmGrade)	Alarm
	LONG **GetCountReadSuccess** (BSTR szChannelName, BSTR szStaName)	I/O Driver
	LONG **GetCountReadTotal** (BSTR szChannelName, BSTR szStaName)	I/O Driver
	LONG **GetCountWriteSuccess** (BSTR szChannelName, BSTR szStaName)	I/O Driver
	LONG **GetCountWriteTotal** (BSTR szChannelName, BSTR szStaName)	I/O Driver
	LONG **GetCurLine** (int* nCurLine)	Network

Alphabet	Function Name	Item
	LONG **GetGroupUserInfoByTag** (BSTR szTagName, LONG* , SAFEARRAY(BSTR) *pszUserID, SAFEARRAY(BSTR) *pszUserName, SAFEARRAY(LONG) *plMailCheck, SAFEARRAY(LONG) *plSMSMCheck, SAFEARRAY(BSTR) *pszMailAdd, SAFEARRAY(BSTR) *pszHPhoneNo)	Alarm
G	LONG **GetHistAlmCount** (LONG lkind, LONG lStatus)	Alarm
	VOID **GetHistDataCount** (BSTR szModelName, BSTR , BSTR szDataKind, DOUBLE dRefTime, LONG lCount, LONG lDir, LONG* lResultCount, SAFEARRAY(double) *psadblTimeStamp, SAFEARRAY(double) *psadblNumValue, SAFEARRAY(LONG) *psalStatus, SAFEARRAY(LONG) *psalResult)	History Data
	VOID **GetHistDataPeriod** (BSTR szModelName, BSTR szTagName, BSTR szDataKind, DOUBLE dFrom, DOUBLE dTo, LONG* lResultCount, SAFEARRAY(double) *psadblTimeStamp, SAFEARRAY(double) *psadblNumValue, SAFEARRAY(LONG) *psalStatus, SAFEARRAY(LONG) *psalResul)	History Data
	VOID **GetHWND** (ULONG* hwnd)	Screen
	INT **GetLineStatus** (int nLineMode)	Network
	LONG **GetLoggingModelList** (LONG *lOutCount ,SAFEARRAY(BSTR) *psaszModelName, SAFEARRAY(BSTR) *psaszModelDesc, SAFEARRAY(BSTR) *psaszModelType, SAFEARRAY(BSTR) *psaszArchvingCycle, SAFEARRAY(BSTR) *psaszSegmentPeriod)	History Data
	LONG **GetLoggingTag** (BSTR szModelName, LONG *lOutCount ,SAFEARRAY(BSTR) *psaszTagName, SAFEARRAY(BSTR) *psaszTagType)	History Data
	BSTR **GetNodeCurIP** (BSTR szNodeName)	Network
	BSTR **GetProjectName** ()	Project
	BSTR **GetProjectPath** ()	Project
	BSTR **GetProjectType** ()	Project
	BSTR **GetServerMode** ()	Network
	LONG **GetStatusRunMode** (BSTR szChannelName, BSTR szStaName)	I/O Driver
	LONG **GetStatusStaOK** (BSTR szChannelName, BSTR szStaName)	I/O Driver
	VARIANT **GetSystemTime** ()	System
	ULONG **GetTagStatus** (BSTR szTagName)	Tag
	BSTR **GetTagStatusDesc**(ULONG lTagStatus)	Tag
	VARIANT **GetTagVal** (BSTR szTagName)	Tag
	VARIANT **GetTagValbyTagId** (LONG lTagId)	Tag

Alphabet	Function Name	Item
G	BSTR **GetTagValnStatusByTagId** (LONG lTagId, VARIANT* vtTagVal, LONG *lScanMode, LONG *lMainQ, LONG *lSubQCnt, LONG *lSubQ1, LONG *lSubQ2, LONG *lSubQ3, LONG *lSubQ4, LONG *lSubQ5)	Tag
	LONG **GetTagValueStatus** (BSTR szTagName, VARIANT* vtTagVal)	Tag
	BSTR **GetUserID** ()	User
	LONG **GettUserInfoList** (LONG* lOutCount, SAFEARRAY(BSTR) *pszUserID, SAFEARRAY(BSTR) *pszUserName, SAFEARRAY(BSTR) *pszDesc, SAFEARRAY(BSTR) *pszPassword, SAFEARRAY(LONG) *plUserLevel, SAFEARRAY(BSTR) *pszHPhoneNo, SAFEARRAY(BSTR) *pszPhoneNo, SAFEARRAY(BSTR) *pszMailAdd)	Alarm
	LONG **GetUserLevel**()	User
H	VOID **HardCopy** ()	Screen
I	BOOL **IsDualLine** ()	Network
	BOOL **IsRunning** ()	Screen
	LONG **IsHoliday** (BSTR szDateTime)	System
N	LONG **NotifyAlarmEvent** (DOUBLE llOccur, USHORT nAlmLevel, BSTR szMessage, BSTR szArea, BSTR szOUser, BSTR zInformation, BSTR szUserDefine1, BSTR szUserDefine2, BSTR szUserDefine3, BSTR szUserDefine4, BSTR szPrev, BSTR szCur, BSTR szObject)	Event
	VOID **NotifyError** (USHORT nLogCategory, BSTR szMsg, BSTR szDesc)	Log
	VOID **NotifyHostError** (BSTR szHostName, USHORT nLogCategory, BSTR szMsg, BSTR szDesc)	Log
	VOID **NotifyHostInfo** (BSTR szHostName, USHORT nLogCategory, BSTR szMsg, BSTR szDesc)	Log
	VOID **NotifyHostLog** (USHORT nLogType, USHORT nLogCategory, BSTR szHostName, BSTR szMsg, BSTR szDesc)	Log
	VOID **NotifyHostWarn** (BSTR szHostName, USHORT nLogCategory, BSTR szMsg, BSTR szDesc)	Log
	VOID **NotifyInfo** (USHORT nLogCategory, USHORT nLogCategory, BSTR szMsg, BSTR szDesc)	Log
	VOID **NotifyLog** (USHORT nLogType, USHORT nLogCategory, BSTR szMsg, BSTR szDesc)	Log
	VOID **NotifyWarn** (USHORT nLogCategory, BSTR szMsg, BSTR szDesc)	Log
O	VOID **OpenPage** (BSTR szPageName, BSTR szPageParam)	Screen
	LONG **OpenPageObject** (BSTR szPageName, BSTR szObjectName)	Screen
	LONG **OpenProject** (BSTR szProjectName, BSTR szProjectPath)	Project

Alphabet	Function Name	Item
P	VOID **PlaySound**(BSTR szWaveFile))	External Device
	VOID **PlaySoundFile** (BSTR szSoundFile)	External Device
	VOID **ProcessSleep** (LONG dwMiliseconds)	System
Q	VOID **Quit** ()	Screen
	LONB **QuitProject** ()	Project
R	LONG **RecipeChangeData** (BSTR ModelName, BSTR RecipeName)	Recipe
	LONG **RecipeChangeUnit** (BSTR ModelName, BSTR UnitName)	Recipe
	LONG **RecipeControlSet** (BSTR szModelName, BSTR szUnitName, BSTR szDataName)	Recipe
	LONG **RecipeControlSetEx** (SHORT nMode, BSTR szClientProgName, ULONG hClientHandle, BSTR szModelName, BSTR szUnitName, BSTR szDataName, LONG* plTransNo, SHORT* pnResult)	Recipe
	LONG **RecipeControlWorkingSet** ();	Recipe
	LONG **RecipeControlWorkingSetEx** (SHORT nMode, BSTR szClientProgName, ULONG hClientHandle, LONG* plTransNo, SHORT* pnResult);	Recipe
	LONG **RecipeDeleteData** (BSTR ModelName, BSTR RecipeName)	Recipe
	LONG **RecipeDeleteModel** (BSTR ModelName)	Recipe
	LONG **RecipeDeleteUnit** (BSTR ModelName, BSTR UnitName)	Recipe
	LONG **RecipeLoadDynamic** (BSTR ModelName, BSTR UnitName)	Recipe
	LONG **RecipeLoadSet** (BSTR ModelName, BSTR UnitName, BSTR RecipeName)	Recipe
	LONG **RecipeSaveAsArchive** (BSTR ModelName, BSTR RecipeName)	Recipe
	LONG **RecipeSetData** (BSTR ModelName, BSTR ItemName, VARIANT setValue)	Recipe
	VOID **ReportDialog** ()	Report
	VOID **ReportListRefresh** ()	Report
	VOID **ReportPrint** (BSTR szReportName)	Report
	VOID **ReportPrintEx** (BSTR szReportName, BSTR szReqTime)	Report
	VOID **ReportSchedulerHide** ()	Report
	VOID **ReportSchedulerShow** ()	Report
	LONG **ResetStaticValue** (BSTR szTagName)	Tag
	VOID **RunApp** (BSTR szAppPath)	System
	VOID **RunScript** (BSTR szScriptName)	Script

Alphabet	Function Name	Item
S	VOID **SendEmail** (BSTR szMailAddTo, BSTR szMailAddFrom, BSTR szMailTitle , BSTR szMailMsg)	Event
	VOID **SendSMS** (BSTR szRecvPhoneNo, BSTR szCallBackPhoneNo, BSTR szSMSMsg)	Event
	LONG **SetHistData** (BSTR szModelName, BSTR szTagName, BSTR szDataKind, LONG lDataCount, SAFEARRAY(double) *psadblSaveTime, SAFEARRAY(double) *psadblNumValue, SAFEARRAY(LONG) *psalTagStatus, SAFEARRAY(LONG) *psalResult)	History Data
	VOID **SetTagVal** (BSTR szTagName, VARIANT vtTagVal)	Tag
	LONG **SetTagValEx** (SHORT nMode, BSTR szTagName, VARIANT vtTagVal, BSTR szClientName, ULONG hClientHandle)	Tag
	LONG **SetUserAlarm** (BSTR szAlarmMessage, BSTR szAlarmGroup)	Alarm
	LONG **SetUserAlarmEx** (DOUBLE dOccur, LONG lAlarmLevel, BSTR szAlarmMessage ,BSTR szAlarmGroup, DOUBLE dCur, DOUBLE dPrev)	Alarm
	LONG **SetUserEvent** (BSTR szEventMessage)	Alarm
	VOID **ShowAlarmPage** (BSTR szAlarmTag)	System
	VOID **ShowCrossReference** (BSTR szTagName)	Tag
	VOID **ShowNetworkStatus** ()	I/O Driver
	VOID **ShowRawDataInfo** ()	Tag
	VOID **ShowSIMCfgInfo** ()	Tag
	VOID **ShowStationInfo** ()	I/O Driver
	VOID **Sleep** (LONG dwMiliseconds)	System
	VOID **StopAlarmSound** ()	Alarm
	VOID **StopScript** (BSTR szScriptName)	Script
T	BSTR **TimeConvert** (DOUBLE dTimeValue, INT nTimeType)	System
	VOID **TimeOutMsgBox** (BSTR Msg, BSTR Title, LONG lSec)	Script
U	LONG **UserLogOff** ()	User
	LONG **UserLogOn** (BSTR szUserID, BSTR szPassWord)	User

5.2 Function Descriptions and Examples

5.2.1 Project Function

Item	Description
Type	LONG **CloseProject** ()
Kind	Project Function
Description	Close current project

Example	Sub Main
	Dim lRet As Long
	lRet = ole.CloseProject()
	If lRet <> 0 Then
	MsgBox "CloseProject() False"
	Else
	MsgBox "CloseProject() OK"
	End If
	End Sub

Item	Description
Type	BSTR **GetProjectName** ()
Kind	Project Function
Description	Get current project name.

Example	Sub Main
	Dim PrjName As String
	PrjName = ole.GetProjectName()
	If lRet <> 0 Then
	MsgBox "GetProjectName()False"
	Else
	MsgBox "GetProjectName() OK " + PrjName
	End If
	End Sub

Item	Description
Type	BSTR **GetProjectPath** ()
Kind	Project Function
Description	Get current project path.

Example	Sub Main
	Dim PrjName As String
	PrjPath = ole.GetProjectPath()
	If lRet <> 0 Then
	MsgBox "GetProjectPath() False"
	Else
	MsgBox "GetProjectPath() OK " + PrjPath
	End If
	End Sub

Item	Description
Type	BSTR **GetProjectType** ()
Kind	Project Function
Description	Get current project type
Example	Sub Main Dim PrjType As String PrjType = ole.GetProjectType MsgBox "GetProjectType : " + PrjType End Sub

Item	Description
Type	LONG **OpenProject**(BSTR szProjectName, BSTR szProjectPath)
Kind	Project Function
Description	Open Project. *szProjectName* : Project Name *szProjectPath* : Project Path

Example	Sub Main
	Dim lRet As Long
	lRet = ole.OpenProject("ProjectName", "ProjectPath")
	If lRet <> 0 Then
	MsgBox "OpenProject() False"
	Else
	MsgBox "OpenProject() OK"
	End If
	End Sub

Item	Description
Type	LONG **QuitProject** ()
Kind	Project Function
Description	Close current project

Example	Sub Main
	Dim lRet As Long
	lRet = ole.QuitProject()
	If lRet <> 0 Then
	MsgBox "QuitProject() False"
	Else
	MsgBox "QuitProject() OK"
	End If
	End Sub

5.2.2 Screen Function

Item	Description
Type	LONG **ChangePage** (BSTR szOldPage, BSTR szNewPage)
Kind	Screen Function
Description	Close an existing page and open a new page *szOldPage*: Old page *szNewPage*: New page

Example	Sub Main
	Dim lRet As Long
	lRet = ole.ChangePage("Page1.ivd", "Page2.ivd")
	If lRet <> 0 Then
	MsgBox "ChangePage() False"
	Else
	MsgBox "ChangePage() OK"
	End If
	End Sub

Item	Description
Type	LONG **ClosePage** (BSTR szPageName)
Kind	Screen Function
Description	Closes the specified page *szPageName* : Page name

Example	Sub Main
	Dim lRet As Long
	lRet = ole.ClosePage("page.ivd")
	If lRet <> 0 Then
	MsgBox "ClosePage() False"
	Else
	MsgBox "ClosePage() OK"
	End If
	End Sub

Item	Description
Type	VOID **GetHWND**(ULONG* hwnd)
Kind	Screen Function
Description	Get handle of runtime program. *hwnd* : Get handle of runtime program

Example	Sub Main
	Dim lRet As Long
	lRet = ole.GetHWND
	If lRet <> 0 Then
	MsgBox "GetHWND()OK" + CStr(lRet)
	Else
	MsgBox "GetHWND() False => lRet : " + CStr(lRet)
	End If
	End Sub

Item	Description
Type	VOID **HardCopy**()
Kind	Screen Function
Description	Prints the full screen associated with the graphic runtime window.
Example	Sub Main ole.HardCopy End Sub

Item	Description
Type	BOOL **IsRunning** ()
Kind	Screen Function
Description	Get current run-time status.

Example	Sub Main
	Dim bRet As Boolean
	bRet = ole.IsRunning
	If bRet = True Then
	MsgBox "IsRunning OK"
	Else
	MsgBox "IsRunning FALSE"
	End If
	End Sub

Item	Description
Type	VOID **OpenPage** (BSTR szPageName, BSTR szPageParam)
Kind	Screen Function
Description	Opens the file whose page name is specified by szPageName *szPageName*: Graphic page name *szPageParam*: Replace name for template screen.

Example	Sub Main
	Dim lRet As Long
	lRet = ole. **OpenPage** ("PAGE1.ivd", “$GR=PLC1”)
	If lRet <> 0 Then
	MsgBox "OpenPage() False"
	Else
	MsgBox "OpenPage () OK"
	End If
	End Sub

Item	Description
Type	LONG **OpenPageObject** (BSTR szPageName, BSTR szObjectName)
Kind	Screen Function
Description	Opens the file whose pagename is specified by szPageName. *szPageName*: Graphic page name *szObjectName*: Replace name for template screen

Example	Sub Main
	Dim lRet As Long
	lRet = ole.OpenPageObject("Page.ivd", "ObjectName")
	If lRet <> 0 Then
	MsgBox "OpenPageObject() False"
	Else
	MsgBox "OpenPageObject() OK"
	End If
	End Sub

Item	Description
Type	VOID **Quit**()
Kind	Screen Function
Description	Terminates the execution of InfoU rumtime.
Example	Sub Main ole.Quit End Sub

5.2.3 Tag Function

Item	Description
Type	ULONG **GetTagStatus** (BSTR szTagName)
Kind	Tag Function
Description	Get the current tag status. *szTagName* : Tag Name

Example	Sub Main
	Dim lRet As Long
	lRet = ole.GetTagStatus("SCAnaTag")
	If lRet <> 0 Then
	MsgBox "GetTagStatus() OK"
	Else
	MsgBox "GetTagStatus() False"
	End If
	End Sub

Item	Description
Type	BSTR **GetTagStatusDesc** (ULONG lTagStatus)
Kind	Tag Function
Description	Let you know the explanation of the tag status by string. *lTagStatus*: Tag Status Value
Example	Sub Main Dim lRet As Long Dim szDesc As String lRet = ole.GetTagValueStatus("tagName") szDesc = ole.GetTagStatusDesc(lRet) MsgBox szDesc End Sub

Item	Description
Type	VARIANT **GetTagVal** (BSTR szTagName)
Kind	Tag Function
Description	Get tag value. *szTagName*: Tag Name
Example	Sub Main Dim Value As Variant Value = 12.0 Value = ole.GetTagVal("FAOA_F_EHV02.AR") MsgBox "FAOA_F_EHV02.AR Value = " + CStr(Value) End Sub

Item	Description
Type	VARIANT **GetTagValbyTagId** (LONG lTagId)
Kind	Tag Function
Description	Get tag values by tag id *lTagId*: Tag ID (Index)
Example	Sub Main Dim Value As Variant Dim Tagid As Long Value = ole.GetTagValbyTagId(Tagid) MsgBox "SCDigTag Value = " + Value End Sub

Item	Description
Type	BSTR **GetTagValnStatusByTagId** (LONG lTagId, VARIANT* vtTagVal, LONG *lScanMode, LONG *lMainQ, LONG *lSubQCnt, LONG *lSubQ1, LONG *lSubQ2, LONG *lSubQ3, LONG *lSubQ4, LONG *lSubQ5)
Kind	Tag Function
Description	Get tag status by tag id. The return values is the description of tag status *lTagId*: Tag ID (Index) *vtTagVal*: Tag Value *lScanMode*: The main mode of tag status. (1:SCANSTOP, 2:SCANSTART) *lMainQ*: Main Quality Code *lSubQCnt*: Detail Quality Code Count *lSubQ1*: Detail Quality Code1 *lSubQ2*: Detail Quality Code2 *lSubQ3*: Detail Quality Code3 *lSubQ4*: Detail Quality Code4 *lSubQ5*: Detail Quality Code5

Item	Description
Type	LONG **GetTagValueStatus** (BSTR szTagName, VARIANT *vtTagVal)
Kind	Tag Function
Description	Inform the tag value and status. In this time, the return value let you know the status value *szTagName* : Tag Name *vtTagVal* : Tag Value

Example	Sub Main
	Dim lRet As Long
	Dim TagValue As Variant
	lRet = ole.GetTagValueStatus("Tag Name" , TagValue)
	If lRet <> 0 Then
	MsgBox " GetTagValueStatus () : SCAN START , Value"
	+ CStr(TagValue)
	Else
	MsgBox " GetTagValueStatus () : SCAN STOP False"
	End If
	End Sub

Item	Description
Type	LONG **ResetStaticValue** (BSTR szTagName)
Kind	Tag Function
Description	Reset statistic value of specified tag *szTagName*: Tag Name

Example	Sub Main
	Dim lRet As Long
	lRet = ole.ResetStaticValue("SCAnaTag")
	If (lRet = 0) Then
	MsgBox "ResetStaticValue() OK "
	Else
	MsgBox "ResetStaticValue() FALSE "
	End If
	End Sub

Item	Description
Type	LONG **ShowCrossReference** (BSTR szTagName)
Kind	Tag Function
Description	Inform you where specific tag is being used currently. *szTagName* : Tag Name
Example	Sub Main ole.ShowCrossReference("SCADA.UT_EHV07._KWHRESET_") End Sub

Item	Description
Type	VOID **ShowRawDataInfo**()
Kind	Tag Function
Description	Display Raw Data Information dialog.
Example	Sub Main ole. ShowRawDataInfo End Sub

Item	Description
Type	VOID **ShowSIMCfgInfo**()
Kind	Tag Function
Description	Display Simulation Setting dialog.
Example	Sub Main ole. ShowSIMCfgInfo End Sub

Item	Description
Type	VOID **SetTagVal** (BSTR szTagName, VARIANT vtTagVal)
Kind	Tag Function
Description	Control tag *szTagName*: Tag Name *vtTagVal*: Tag value which was controled
Example	Sub Main Dim Value As Variant MsgBox " Before : SetTagVal(SCDigTag) Value = " + ole.GetTagVal("tagName") Value = 1 ole.SetTagVal("tagName", Value) MsgBox " After : SetTagVal(Tag Name) Value = " + ole.GetTagVal("tagName") End Sub

Item	Description
Type	LONG **SetTagValEx** (SHORT nMode, BSTR szTagName, VARIANT vtTagVal, BSTR szClientName, ULONG hClientHandle)
Kind	Tag Function
Description	Control specified tag value *nMode*: Control Mode (Only No Return mode supported : 1) *szTagName*: Tag Name *vtTagVal*: Tag Value to Control *szClientName* : Client Program Name that perform control *hClientHandle*: Client Window Handle that perform control
Example	Sub Main Dim Value As Variant Dim TransNo As Long MsgBox " Before : SetTagValEx(SCDigTag) Value = " + ole.GetTagVal("tagName") Value = 1 TransNo = ole.SetTagVaLEx (1, "tagName", Value , “Script”, NULL) MsgBox " After : SetTagValEx(tagName) Value = " + ole.GetTagVal("tagName") End Sub

5.2.4 Script Function

Item	Description
Type	VOID **RunScript** (BSTR szScriptName)
Kind	Script Function
Description	Run Script *szScriptName*: Script Name
Example	Sub Main Dim scTime As Variant ole.RunScript("Script Name") End Sub

Item	Description
Type	VOID **StopScript** (BSTR szScriptName)
Kind	Script Function
Description	Stop Script *szScriptName*: Script Name
Example	Sub Main Dim scTime As Variant ole.StopScript("Script Name") End Sub

Item	Description
Type	VOID **TimeOutMsgBox** (BSTR Msg, BSTR Title, Int nSec)
Kind	Script Function
Description	Display message box while time interval In case of use normal message box in script, be careful of closing script function *Msg*: message contents Title: title nSec: time interval (second)
Example	Sub Main ole.TimeOutMsgBox "Today is Holiday","Title",5 End Sub

5.2.5 Event Function

Item	Descriptio
Type	LONG **NotifyAlarmEvent** (DOUBLE llOccur, USHORT nAlmLevel, BSTR szMessage, BSTR szArea, BSTR szOUser, BSTR szInformation, BSTR szUserDefine1, BSTR szUserDefine2, BSTR szUserDefine3, BSTR szUserDefine4, BSTR szPrev, BSTR szCur, BSTR szObject)
Kind	Event Function
Description	Occur Alarm & Event. *llOccur*: Occurrence Time. NULL = current time *nAlmLevel*: The level of alarm *szMessage*: Message *szArea*: Occurrence Region *szOUser*: Occurrence User *szInformation*: Information *szUserDefine1*: User Definition1 *szUserDefine2*: User Definition2 *szUserDefine3*: User Definition3 *szUserDefine4*: User Definition4 *szPrev*: Pervious Value of Message *szCur*: New Value of Message *szObject*: Message Object

Example	Sub Main
	Dim lRet As Long
	Dim dDate As Double
	Dim iLevel As Integer
	ole.ChangeProcessServer("ALM_PROC","150.150.32.60")
	If lRet <> 0 Then
	MsgBox "ChangeProcessServer() False : ret = " + CStr(lRet)
	Else
	MsgBox "ChangeProcessServer() OK : ret = " + CStr(lRet)
	End If
	dDate = NUL
	iLevel = 1
	lRet = ole.NotifyAlarmEvent(dDate, iLevel,"NotifyAlarmEvent Test", _
	"Script", "Script", "Interfun API Success", "", "", "", _
	"", “before bun", "after Run","")
	If lRet <> 0 Then
	MsgBox "NotifyAlarmEvent() False : ret = " + CStr(lRet)
	Else
	MsgBox "NotifyAlarmEvent() OK : ret = " + CStr(lRet)

	End If End Sub

Item	Description
Type	VOID **SendEmail** (BSTR szMailAddTo, BSTR szMailAddFrom, BSTR szMailTitle , BSTR szMailMsg)
Kind	Event Function
Description	Send E-Mail *szMailAddTo* : E-Mail Sender *szMailAddFrom* : E-Mail Receiver *szMailTitle*: E-Mail Title *szMailMsg* : E-Mail Body
Example	Sub Main ole.SendEmail("receiver@lsis.biz","sender@lsis.biz","title","message: Script Test Good~~!!") End Sub

Item	Description
Type	VOID **SendSMS** (BSTR szRecvPhoneNo, BSTR szCallBackPhoneNo, BSTR szSMSMsg)
Kind	Event Function
Description	Send SMS. *szRecvPhoneNo* : The phone number of SMS receiver *szCallBackPhoneNo* : The phone number of SMS sender *szSMSMsg* : SMS Message (English:80, Korean:40)
Example	Sub Main ole.SendSMS("0194851234","0194850000","Script Test") End Sub

5.2.6 Log Function

Item	Description
Type	VOID **NotifyError** (USHORT nLogCategory, BSTR szMsg, BSTR szDesc)
Kind	Log Function
Description	Error Logs are recorded. *nLogCategory*: Log Category (Default : LOG_CATE_SCRIPT= 1, 0) *szMsg*: Log Message *szDesc*: Log Description
Example	Sub Main ole.NotifyError(1, "ERROR", "Notify_ERROR") End Sub

Item	Description
Type	VOID **NotifyHostError** (BSTR szHostName, USHORT nLogCategory, BSTR szMsg, BSTR szDesc)
Kind	Log Function
Description	Error Logs are recorded to specified computer. *szHostName*: Target computer Name *nLogCategory*: Log Category (Default : LOG_CATE_SCRIPT= 1, 0) *szMsg*: Log Message *szDesc*: Log Description
Example	Sub Main ole.NotifyHostError("127.0.0.1",1, "HOST_ERROR", "Notify_HOST_ERROR") End Sub

Item	Description
Type	VOID **NotifyHostInfo** (BSTR szHostName, USHORT nLogCategory, BSTR szMsg, BSTR szDesc)
Kind	Log Function
Description	Information Logs are recorded to specified computer. *szHostName*: Target computer Name *nLogCategory*: Log Category (Default : LOG_CATE_SCRIPT= 1, 0) *szMsg*: Log Message *szDesc*: Log Description
Example	Sub Main ole.NotifyHostInfo ("127.0.0.1",1, "HOST_INFO", "Notify_HOST_INFO") End Sub

Item	Description
Type	VOID **NotifyHostLog** (USHORT nLogType, USHORT nLogCategory, BSTR szHostName, BSTR szMsg, BSTR szDesc)
Kind	Log Function
Description	Logs are recorded to specified computer. *nLogType*: Log type(1 = INFO_LOG_TYPE(information), 2 = WARN_LOG_TYPE (warning), 3 = ERRO_LOG_TYPE(error)) *nLogCategory*: Log Category (Default : LOG_CATE_SCRIPT= 1, 0) *szHostName*: Target computer Name *szMsg*: Log Message *szDesc*: Log Description
Example	Sub Main ole.NotifyHostLog(1, 1, "127.0.0.1", "HOST_INFO", "NotifyHostLog__ INFO") ole.NotifyHostLog(2, 1, "127.0.0.1", "HOST_WARN", "NotifyHostLog _ WARN") ole.NotifyHostLog(3, 1, "127.0.0.1","HOST_ERROR", "NotifyHostLog _ ERROR") End Sub

Item	Description
Type	VOID **NotifyHostWarn** (BSTR szHostName, USHORT nLogCategory, BSTR szMsg, BSTR szDesc)
Kind	Log Function
Description	Warning Logs are recorded. *szHostName*: Target computer Name *nLogCategory*: Log Category (Default : LOG_CATE_SCRIPT= 1, 0) *szMsg*: Log Message *szDesc*: Log Description
Example	Sub Main ole.NotifyHostWarn ("127.0.0.1",1, "HOST_WARN", "Notify_HOST_WARN") End Sub

Item	Description
Type	VOID **NotifyInfo** (USHORT nLogCategory, BSTR szMsg, BSTR szDesc)
Kind	Log Function
Description	Information Logs are recorded. *nLogCategory*: Log Category (Default : LOG_CATE_SCRIPT= 1, 0) *szMsg*: Log Message *szDesc*: Log Description
Example	Sub Main ole.NotifyInfo(1, "INFO", "Notify_INFO") End Sub

Item	Description
Type	VOID **NotifyLog** (USHORT nLogType, USHORT nLogCategory, BSTR szMsg, BSTR szDesc)
Kind	Log Function
Description	Logs are recorded *nLogType*: Log type (1 = INFO_LOG_TYPE(Information), 2 = WARN_LOG_TYPE (Warning), 3 = ERRO_LOG_TYPE(Error)) *nLogCategory*: Log Category (Default : LOG_CATE_SCRIPT= 1, 0) *szMsg*: Log Message *szDesc*: Log Description
Example	Sub Main ole.NotifyLog(1, "INFO", 1,"NotifyLog _ INFO") ole.NotifyLog(2, "WARN", 1, "NotifyLog _ WARN") ole.NotifyLog(3, "ERROR",1,"NotifyLog _ ERROR") End Sub

Item	Description
Type	VOID **NotifyWarn** (USHORT nLogCategory, BSTR szMsg, BSTR szDesc)
Kind	Log Function
Description	Warning Logs are recorded *nLogCategory*: Log Category (Default : LOG_CATE_SCRIPT= 1, 0) *szMsg*: Log Message *szDesc*: Log Description
Example	Sub Main ole.NotifyWarn(1, "WARN", "Notify_WARN") End Sub

5.2.7 Network Function

Item	Description
Type	BSTR **ChangLine** (BSTR szLineMode, BSTR szTargetLineIP)
Kind	Network Function
Description	Changes a network line with designation IP address *szLineMode*: line mode (DEFAULT, SUB) *szTargetLineIP*: line IP address

Example	Sub Main
	Dim bRet As Boolean
	bRet = ole.ChangLine("DEFAULT", "xxx.xxx.xxx.xxx")
	If bRet <> 0Then
	MsgBox "ChangLine FALSE"
	Else
	MsgBox "ChangLine OK"
	End If
	End Sub

Item	Description
Type	LONG **ChangeProcessServer** (BSTR szProcessName, BSTR szProcessIP)
Kind	Network Function
Description	Changes with process of network address (IP). *szProcessName*: process name *szProcessIP*: target computer's IP address

Example	Sub Main
	Dim lRet As Long
	lRet = ole.ChangeProcessServer("ALM_PROC","xxx.xxx.xxx.xxx")
	If lRet <> 0 Then
	MsgBox "ChangeProcessServer() False"
	Else
	MsgBox "ChangeProcessServer() OK "
	End If
	End Sub

Item	Description
Type	LONG **ChangeServerMode** (BSTR szServerMode)
Kind	Network Function
Description	Changes a server mode.(redundancy option only) *szServerMode*: target mode (NORMAL, RUN, STANDBY)
Example	Sub Main

	Dim lRet As Long
	lRet = ole.ChangeServerMode("NORMAL")
	If lRet <> 0 Then
	MsgBox "ChangeServerMode()False"
	Else
	MsgBox "ChangeServerMode() OK"
	End If
	End Sub

Item	Description
Type	LONG **GetCurLine** (int* nCurLine)
Kind	Network Function
Description	Get current used line.(line redundancy option only) *nCurLine*: network line(0: Default IP, 1: Sub IP , 2: stop)

Example	Sub Main
	Dim lRet As Long
	Dim lLineSatus As Long
	lRet = ole.GetCurLine(lLineSatus)
	If lRet <> 0 Then
	MsgBox "GetCurLine()False => Status " + CStr(lLineSatus)
	Else
	MsgBox "GetCurLine() OK => Status " + CStr(lLineSatus)
	End If
	End Sub

Item	Description
Type	INT **GetLineStatus** (INT nLineMode)
Kind	Network Function
Description	Get current line status *nLineMode*: line status

Example	Sub Main
	Dim lRet As Long
	Dim lLineStatus As Long
	
	lLineStatus = 0
	lRet = ole.GetLineStatus(lLineStatus)
	
	If (lRet = 1) Then
	MsgBox "GetLineStatus() OK: " + CStr(lLineStatus)
	Else

	MsgBox "GetLineStatus() FALSE: " + CStr(lLineStatus) End If End Sub

Item	Description
Type	BSTR **GetNodeCurIP** (BSTR szNodeName)
Kind	Network Function
Description	Get current used node IP address *szNodeName*: node name(computer name)
Example	Sub Main Dim IP As String IP = ole.GetNodeCurIP("SERVER") MsgBox "GetNodeCurIP() => SERVER " + IP End Sub

Item	Description
Type	BSTR **GetServerMode** ()
Kind	Network Function
Description	Get server mode(mode: run, standby)
Example	Sub Main Dim SvrMode As String SvrMode = ole.GetServerMode MsgBox "GetServerMode : " + SvrMode End Sub

Item	Description
Type	BOOL **IsDualLine** ()
Kind	Network Function
Description	Get network line redundancy settings

Example	Sub Main
	Dim bRet As Boolean
	bRet = ole.IsDualLine
	If bRet = True Then
	MsgBox "IsDualLine OK"
	Else
	MsgBox "IsDualLine FALSE"
	End If
	End Sub

5.2.8 System Function

Item	Description
Type	VARIANT **GetSystemTime** ()
Kind	System Function
Description	Get current system time

Example	Sub Main
	  Dim scTime As Variant
	  scTime = ole.GetSystemTime
	  If lRet <> 0 Then
	  MsgBox "GetSystemTime() False"
	  Else
	  MsgBox "GetSystemTime() OK" + scTime
	  End If
	End Sub

Item	Description
Type	LONG **IsHoliday** (szDateTime)
Kind	System Function
Description	It informs you of whether the date delivered as a factor is a holiday or not. szDateTime: YYYY/MM/DD HH:MM:SS Return Value: 0=Weekday, 1=Saturday, 2=Sunday, 3=Holiday

Example	Sub Main
	Dim szDateTime
	Dim lRetn
	  szDateTime = Format(Now(),"YYYY/MM/DD HH:MM:SS")
	  lRetn = ole.IsHoliday(szDateTime)
	
	  If lRetn = 0 Then
	  Debug.Print "Today is a Weekday"
	  ElseIf lRetn = 1 Then
	  Debug.Print "Today is Saturday"
	  ElseIf lRetn = 2 Then
	  Debug.Print "Today is Sunday"
	  ElseIf lRetn = 3 Then
	  Debug.Print "Today is Holiday"
	  End If
	End Sub

Item	Description
Type	VOID **ProcessSleep**(LONG dwMiliseconds)
Kind	System Function
Description	Suspends the current process for a specified time dwMiliseconds: millisecond
Example	Sub Main Dim scTime As Variant ole.ProcessSleep(1000) End Sub

Item	Description
Type	VOID **RunApp** (BSTR szAppPath)
Kind	System Function
Description	Excute application. (include path) *szAppPath*: application path
Example	Sub Main Dim scTime As Variant ole.RunApp("C:\WINDOWS\System32\mspaint.exe") End Sub

Item	Description
Type	VOID **ShowAlarmPage**(BSTR szAlarmTag)
Kind	System Function
Description	This function displays the alarm page with tag which is occured alram *szAlarmTag*: Tag name
Example	Sub Main Dim scTime As Variant ole. ShowAlarmPage ("Tag name") End Sub

Item	Description
Type	VOID **Sleep**(LONG dwMiliseconds)
Kind	System Function
Description	Suspends the current script for a specified time dwMiliseconds: millisecond
Example	Sub Main Dim scTime As Variant ole. Sleep(1000) End Sub

Item	Description
Type	BSTR **TimeConvert**(DOUBLE dTimeValue, INT nTimeType)
Kind	System Function
Description	According to format outputs a time in string *dTimeValue* current time. (ex <=0 : curent time) *nTimeType* : 0 : HH:MM 1 : HH:MM AM/PM 2 : HH:MM:SS 3 : HH:MM:SS AM/PM 7 : mm/dd 8 : mm-dd 10 : yy/mm/dd 11 : yy-mm-dd 13 : yyyy/mm/dd 14 : yyyy-mm-dd 15 : mm/dd HH:MM 16 : mm/dd HH:MM AM/PM 17 : mm/dd HH:MM:SS 18 : mm/dd HH:MM:SS AM/PM 19 : mm-dd HH:MM 20 : mm-dd HH:MM AM/PM 21 : mm-dd HH:MM:SS 22 : mm-dd HH:MM:SS AM/PM 23 : yy/mm/dd HH:MM 24 : yy/mm/dd HH:MM AM/PM 25 : yy/mm/dd HH:MM:SS 26 : yy/mm/dd HH:MM:SS AM/PM 27 : yy-mm-dd HH:MM 28 : yy-mm-dd HH:MM AM/PM 29 : yy-mm-dd HH:MM:SS 30 : yy-mm-dd HH:MM:SS AM/PM 31 : yyyy/mm/dd HH:MM 32 : yyyy/mm/dd HH:MM AM/PM 33 : yyyy/mm/dd HH:MM:SS 34 : yyyy/mm/dd HH:MM:SS AM/PM 35 : yyyy-mm-dd HH:MM 36 : yyyy-mm-dd HH:MM AM/PM 37 : yyyy-mm-dd HH:MM:SS 38 : yyyy-mm-dd HH:MM:SS AM/PM

Item	Description
Type	BSTR **TimeConvert**(DOUBLE dTimeValue, INT nTimeType)
Kind	System Function
	45 : yy/mm 47 : yyyy/mm 49 : yy 50 : yyyy 51 : mm 52 : dd 53 : HH 54 : MM 55 : SS 65 : “SUN” ~ “SAT”
Example	Sub Main Dim szTime As String szTime = ole.TimeConvert (1, 33) MsgBox szTime End Sub

5.2.9 Recipe Function

Item	Description
Type	LONG **RecipeChangeData** (BSTR szModelName, BSTR szRecipeName)
Kind	Recipe Function
Description	Change Data Set of the workingset (Model must be the workingset's model) *szModelName*: Recipe Model Name *szRecipeName*: The data name belonging to the recipe model

Example	Sub Main
	Dim Result As Long
	Result = ole.RecipeLoadSet("CoffeDispenser2",NUL,"CreamCoffee")
	
	If (Result <> 0) Then
	MsgBox("RecipeLoadSet : Error~~!!")
	Else
	Result = ole.RecipeChangeData("CoffeDispenser2","CoffeeA")
	
	If (Result <> 0) Then
	MsgBox("RecipeChangeData : Error~~!!")
	Else
	MsgBox("RecipeChangeData : OK~~!!")
	End If
	End If
	End Sub

Item	Description
Type	LONG **RecipeChangeUnit** (BSTR szModelName, BSTR szUnitName)
Kind	Recipe Function
Description	Change Unit Set of the workingset (Model must be the workingset's model) *szModelName*: Recipe Model Name *szUnitName*: The unit name belonging to the recipe model

Example	Sub Main
	Dim Result As Long
	Result = ole.RecipeLoadSet("CoffeDispenser2",NUL,NUL)
	If (Result <> 0) Then
	MsgBox("RecipeLoadSet : Error~~!!")
	Else
	Result = ole.RecipeChangeUnit("CoffeDispenser2", _
	"OfficeDispenser")

	If (Result <> 0) Then
	MsgBox("RecipeChangeUnit : Error~~!!")
	Else
	MsgBox("RecipeChangeUnit : OK~~!!")
	End If
	End If
	
	End Sub

Item	Description
Type	LONG **RecipeControlSet** (BSTR szModelName, BSTR szUnitName, BSTR szDataNamet) LONG **RecipeControlSetEx** (SHORT nMode, BSTR szClientProgName, ULONG hClientHandle, BSTR szModelName, BSTR szUnitName, BSTR szDataName, LONG* plTransNo, SHORT* pnResult)
Kind	Recipe Function
Description	Perform recipe control with the model, unit and data information. (This function has no connection with workingset) *nMode*: Control Mode (nMode : 1: no return, 2: Return) *szClientProgName*: The program name that want to recipe control. *hClientHandle*: The client handle that want to recipe control. *szModelName* : Recipe Model Name *szUnitName* : The unit name belonging to the recipe model *szDataName* : The data name belonging to the recipe model *plTransNo*: In the case of recipe control return mode, The transaction number when control was succefully completed *pnResult* : Result of function

Example	Sub Main
	Dim TransNo As Long
	Dim Result As Long
	Dim Result2 As Integer
	Result = ole.RecipeLoadSet("CoffeDispenser2",NUL,"CoffeeA")
	
	If (Result <> 0) Then
	MsgBox("RecipeLoadSet : Error~~!!")
	Else
	Result2 = ole.RecipeControlSetEx(2,"Script",this, _
	"CoffeDispenser2","OfficeDispenser","CoffeeA",TransNo)
	Result2 = ole.RecipeControlSet("CoffeDispenser2", _

	"OfficeDispenser","CoffeeA")
	
	If (Result2 <> 0) Then
	MsgBox("RecipeControlSet : Error~~!!")
	Else
	MsgBox("RecipeControlSet : OK~~!! , TransNO = " _
	+ CStr(TransNo))
	End If
	End If
	End Sub

Item	Description
Type	LONG **RecipeControlWorkingSet** (); LONG **RecipeControlWorkingSetEx** (SHORT nMode, BSTR szClientProgName, ULONG hClientHandle, LONG* plTransNo, SHORT* pnResult);
Kind	Recipe Function
Description	Control workingset data with unit name. (There is no control return) *nMode*: Control Mode (nMode: 1:No Return, 2:Return) *szClientProgName*: The program name that want to control *hClientHandle*: The client handle that want to control *plTransNo*: In the case of recipe control return mode, The transaction number when control was succefully performed *pnResult* : Result of function

Example	Sub Main
	Dim TransNo As Long
	Dim Result As Long
	Dim Result2 As Integer
	Result = ole.RecipeLoadSet("CoffeDispenser2",NUL,"CoffeeA")
	
	If (Result <> 0) Then
	MsgBox("RecipeLoadSet : Error~~!!")
	Else
	Result2 = ole.RecipeControlWorkingSetEx(2,"Script",this, TransNo)
	'Result2 = ole.RecipeControlWorkingSet()
	
	If (Result2 <> 0) Then
	MsgBox("RecipeControlWorkingSet : Error~~!!")
	Else
	MsgBox("RecipeControlWorkingSet : OK~~!! , TransNO = " + CStr(TransNo))

	End If End If End Sub

Item	Description
Type	LONG **RecipeDeleteData** (BSTR szModelName, BSTR szRecipeName)
Kind	Recipe Function
Description	Delete Recipe Data *szModelName* : Recipe Model Name *szRecipeName* : The data name belonging to the recipe model

Example	Sub Main
	Dim Result As Long
	Result = ole.RecipeDeleteData("CoffeDispenser2","Amerian")
	If (Result <> 0) Then
	MsgBox("RecipeDeleteData : Error~~!!")
	Else
	MsgBox("RecipeDeleteData : OK~~!!")
	End If
	End Sub

Item	Description
Type	LONG **RecipeDeleteModel** (BSTR szModelName)
Kind	Recipe Function
Description	Delete Recipe Model (The recipe unit and data will be deleted belonging to recipe model) szModelName: Recipe Model Name

Example	Sub Main
	Dim Result As Long
	Result = ole.RecipeDeleteModel("CoffeDispenser3")
	If (Result <> 0) Then
	MsgBox("RecipeDeleteModel : Error~~!!")
	Else
	MsgBox("RecipeDeleteModel : OK~~!!")
	End If
	End Sub

.

Item	Description
Type	LONG **RecipeDeleteUnit** (BSTR szModelName, BSTR szUnitName)
Kind	Recipe Function
Description	Delete Recipe Unit *szModelName*: Recipe Model Name *szUnitName* : The unit name belonging to the recipe model

Example	Sub Main
	Dim Result As Long
	Result = ole.RecipeDeleteUnit("CoffeDispenser2","SubOfficeDispenser")
	If (Result <> 0) Then
	MsgBox("RecipeDeleteUnit : Error~~!!")
	Else
	MsgBox("RecipeDeleteUnit : OK~~!!")
	End If
	End Sub

Item	Description
Type	LONG **RecipeLoadDynamic** (BSTR szModelName, BSTR szUnitName)
Kind	Recipe Function
Description	Get real-time data of the unit and put recipe model and data on workingset *szModelName*: Recipe Model Name *szUnitName*: The unit name belonging to the recipe model

Example	Sub Main
	Dim Result As Long
	Result = ole.RecipeLoadDynamic("CoffeDispenser2","MainOfficeDispenser ")
	If (Result <> 0) Then
	MsgBox("RecipeLoadDynamic : Error~~!!")
	Else
	MsgBox("RecipeLoadDynamic : OK~~!!")
	End If
	End Sub

Item	Description
Type	LONG **RecipeLoadSet** (BSTR szModelName, BSTR szUnitName, BSTR szRecipeName)
Kind	Recipe Function
Description	Put recipe model, unit and data on workingset *szModelName*: Recipe Model Name *szUnitName*: The unit name belonging to the recipe model *szRecipeName* : The data name belonging to the recipe model

Example	Sub Main
	Dim Result As Long
	Result = ole.RecipeLoadSet("CoffeDispenser2",NUL,"CreamCoffee")
	If (Result <> 0) Then
	MsgBox("RecipeLoadSet : Error~~!!")
	Else
	MsgBox("RecipeLoadSet : OK~~!!")
	End If
	End Sub

Item	Description
Type	LONG **RecipeSaveAsArchive** (BSTR szModelName, BSTR szRecipeName)
Kind	Recipe Function
Description	Save workingset data as recipe name of the model name. (Model must be the workingset's model) *szModelName*: Recipe Model Name *szRecipeName*: The data name belonging to the recipe model

Example	Sub Main
	Dim Result As Long
	Result = ole.RecipeLoadSet("CoffeDispenser2",NUL,"BlackCoffee")
	If (Result <> 0) Then
	MsgBox("RecipeLoadSet : Error~~!!")
	Else
	Result = ole.RecipeSaveAsArchive("CoffeDispenser2", _
	"BlackCoffee")
	If (Result <> 0) Then
	MsgBox("RecipeSaveAsArchive : Error~~!!")
	Else
	MsgBox("RecipeSaveAsArchive : OK~~!!")
	End If
	End If
	End Sub

Item	Description
Type	LONG **RecipeSetData** (BSTR szModelName, BSTR szItemName, VARIANT szSetValue)
Kind	Recipe Function
Description	Change position value of the item in the workingset (Model must be the workingset's model) *szModelName* : Recipe Model Name *szItemName* : Item in the workingset *szSetValue* : The change value of the item position in the workingset

Example	Sub Main
	Dim vtSetValue As Variant
	Dim Result As Long
	
	Result = ole.RecipeLoadSet("CoffeDispenser2",NUL,NUL)
	vtSetValue = 55
	If (Result <> 0) Then
	MsgBox("RecipeLoadSet : Error~~!!")
	Else
	Result = ole.RecipeSetData("CoffeDispenser2", _
	"Coffee",vtSetValue)
	
	If (Result <> 0) Then
	MsgBox("RecipeSetData : Error~~!!")
	Else
	MsgBox("RecipeSetData : OK~~!!")
	End If
	End If
	End Sub

5.2.10 Database Function

Item	Description
Type	VOID **DataExportRunWork** (BSTR szWorkName)
Kind	Database Function
Description	Run data export operation *szWorkName* : Operation Name
Example	Sub Main ole. DataExporRuntWork (“OperationName”) End Sub

5.2.11 Alarm Function

Item	Description
Type	BOOL **AckAllAlarm**()
Kind	Alarm Function
Description	The function ACKs all alarm with level of connected users.
Example	Sub Main ole. AckAllAlarm End Sub

Item	Description
Type	BSTR **GetAlarmCountErrInfo** (LONG lCount)
Kind	Alarm Function
Description	Give an explanation of the error number when you request the alarm count. *lCount*: Function GetAlmCount(),The Return Value of GetHistAlmCount() < 0

Example	Sub Main
	Dim szErrDesc As String
	Dim AlmCount As Long
	
	AlmCount = ole. GetHistAlmCount (1, 0)
	If AlmCount < 0 Then
	szErrDesc = ole.GetAlarmCountErrInfo (AlmCount)
	MsgBox szErrDesc
	End If
	End Sub

Item	Description
Type	LONG **GetAlarmLogging** (DOUBLE dtFrom, DOUBLE dtTo, BSTR szTag, BSTR szAlarmGr, LONG lAlarmType, LONG *lOutCount, SAFEARRAY **pszRstrTag, SAFEARRAY **pszRstrGr, SAFEARRAY **pdbRdtOccAlarm, SAFEARRAY **pdbRdtRevAlarm, SAFEARRAY **pdbRdtAckAlarm, SAFEARRAY **plRnAlarmType, SAFEARRAY **pszRstrMsg, SAFEARRAY **pszRstrAlarmKind)
Kind	Alarm Function
Description	This function retrieves history alarm list according to the condition *dtFrom*: Beginning Time (from the first data, if 0) *dtTo*: End Time (to the last data, if 0) *szTag*: Tag Name (NULL: ALL) *szAlarmGr*: Alarm Group (NULL: ALL) *lAlarmType*: Alarm Type (0: ALL, 1: Alarm(Occurrence), 2: Event, 3: Alarm(Recovery), 4: Alarm (Occurrence+Recovery)) *lOutCount*: Alarm List Count *pszRstrTag*: Tag Name List *pszRstrGr*: Alarm Group List *pdbRdtOccAlarm*: List of Alarm Occurrence Times *pdbRdtRevAlarm*: List of Alarm Occurrence Times *pdbRdtAckAlarm*: List of Alarm Ack Times *plRnAlarmType*: List of Alarm Types *pszRstrMsg*: List of Alarm Messages *pszRstrAlarmKind*: Alarm Type (HH,HI,LO,LL,Achange,MGap,SGap,On,Off,OnOff,OffOn,DChagne)

Item	Description (Continued)
Example	Sub Main Dim dtFrom As Date Dim dtTo As Date Dim szTag As String Dim szAlarmGr As String Dim lAlarmType As Long Dim lOutCount As Long Dim aRstrTag() As String Dim aRstrGr() As String Dim aRdtOccAlarm() As Double Dim aRdtRevAlarm() As Double Dim aRdtAckAlarm() As Double Dim aRnAlarmType() As Long Dim aRstrMsg() As String Dim aRstrAlarmKind() As String dtFrom = Now() - 27 dtTo =Now() szTag ="" szAlarmGr = "AlmGrp_UT" lAlarmType = 0 lRet = ole.GetAlarmLogging(dtFrom ,dtTo , szTag , szAlarmGr,lAlarmType,lOutCount, aRstrTag, aRstrGr, aRdtOccAlarm, aRdtRevAlarm,aRdtAckAlarm, aRnAlarmType, aRstrMsg,aRstrAlarmKind) For i = 0 To lOutCount - 1 MsgBox (aRstrTag(i) + " " + aRstrGr(i) + " "+ CStr(CDate(aRdtOccAlarm(i))) + " "+ CStr(CDate(aRdtRevAlarm(i))) + " "+ CStr(CDate(aRdtAckAlarm(i))) + " " + CStr(aRnAlarmType(i)) + " " + aRstrMsg(i) + " " + aRstrAlarmKind(i)) Next i End Sub

Item	Description
Type	LONG **GetAlarmLoggingEx** (DOUBLE dtFrom,DOUBLE dtTo,BSTR szTag, LONG nAlarmGroupCount, SAFEARRAY **pszAlarmGr,LONG lAlarmType, LONG lAlarmGrade, BSTR szAlarmKind, LONG *lOutCount, SAFEARRAY**pRstrTag,SAFEARRAY**pRstrGr, SAFEARRAY **pdbRdtOccAlarm, SAFEARRAY **pdbRdtRevAlarm, SAFEARRAY **pdbRdtAckAlarm, SAFEARRAY **plRnAlarmType, SAFEARRAY **pszRstrMsg, SAFEARRAY **pszRstrAlarmKind, SAFEARRAY **pdbRvOccDValue, SAFEARRAY**pszRvOccSValue, SAFEARRAY **plAlarmGrade)
Kind	Alarm Function
Description	This function retrieves history alarm list according to the condition, extension function of GetAlarmLogging() *dtFrom*: Beginning Time (from the first data, if 0) *dtTo*: End Time (to the last data, if 0) *szTag*: Tag Name (NULL: ALL) *nAlarmGroupCount*: Alarm Group Count *pszAlarmGr*: Alarm Group List (NULL: ALL) *lAlarmType*: Alarm Type (Event has only Occurrence.) (0: ALL, 1: Alarm(Occurrence), 2: Event, 3: Alarm(Recovery), 4: Alarm (Occurrence+Recovery)) *lAlarmGrade*: Alarm Level. You can get the alarm lists that have higher level than this level. Level1 is higher than Level10. ex) If input is level5, returns level1 ~ level5. *szAlarmKind*:: Alarm Type (NULL: ALL, HH,HI,LO,LL,Achange,MGap,SGap, On,Off,OnOff,OffOn,DChagne) *lOutCount*: Alarm List Count *pszRstrTag*: List of Tag Names *pszRstrGr*: List of Alarm Groups *pdbRdtOccAlarm*: List of Alarm Occurrence Times *pdbRdtRevAlarm*: List of Alarm Recovery Times *pdbRdtAckAlarm*: List of Alarm Ack Times *plRnAlarmType*: List of Alarm Types *pszRstrMsg*: List of Alarm Messages *pszRstrAlarmKind*: Alarm Type

	(HH,HI,LO,LL,Achange,MGap,SGap,On,Off,OnOff,OffOn,DChagne) *pdbRvOccDValue* List of Current Values (Only if Alarm) *pszRvOccSValue*: List of Current Values (only if Event) *plAlarmGrade*: List of Alarm Levels
Example	Sub Main Dim dtFrom As Double Dim dtTo As Double Dim szTag As String Dim nAlarmCount As Long Dim aAlarmGr(0) As String Dim lAlarmType As Long Dim lAlarmGrade As Long Dim szAlarmKind As String Dim lOutCount As Long Dim aRstrTag() As String Dim aRstrGr() As String Dim aRdtOccAlarm() As Double Dim aRdtRevAlarm() As Double Dim aRdtAckAlarm() As Double Dim aRnAlarmType() As Long Dim aRstrMsg() As String Dim aRstrAlarmKind() As String Dim aOccDvalue() As Double Dim aOccSValue() As String Dim aAlarmLevel() As Long dtFrom = Now() - 27 dtTo = Now szTag ="" nAlarmCount = 1 aAlarmGr(0) = "AlmGrp_UT" lAlarmType = 4 lAlarmGrade = 10 szAlarmKind = "LO" lRet = ole.GetAlarmLoggingEx (dtFrom ,dtTo , szTag , nAlarmCount, aAlarmGr,lAlarmType,lAlarmGrade, szAlarmKind, lOutCount, aRstrTag, aRstrGr, aRdtOccAlarm, aRdtRevAlarm,aRdtAckAlarm, aRnAlarmType,

```
                                aRstrMsg,aRstrAlarmKind, aOccDvalue,
                                aOccSValue, aAlarmLevel)



  For i = 0 To lOutCount
    MsgBox (aRstrTag(i) + " " + aRstrGr(i)    + " "+
                        CStr(CDate(aRdtOccAlarm(i)))+ " "+
                        CStr(CDate(aRdtRevAlarm(i)))+ " "+
                        CStr(CDate(aRdtAckAlarm(i))) + " " +
                        CStr(aRnAlarmType(i)) + " " +
                         aRstrMsg(i) + " " + aRstrAlarmKind(i))
  Next i
End Sub
```

Item	Description
Type	LONG **GetAlmCount** (LONG lKind, LONG lStatus)
Kind	Alarm Function
Description	Inform the real-time alarm count. *lKind*: Alarm Kind 0 : All 1 : Process Alarm 2 : System Alarm 3 : Process + System 4 : Event 5 : Process + Event 6 : System + Event *lStatus*: Alarm Status 0 : All 1 : Unrecovered 2 : Unrecognized 3 : Unrecovered + Unrecognized

Example	Sub Main
	Dim AlmCount As Long
	AlmCount = ole. GetAlmCount(1, 0)
	If AlmCount > 0 Then
	MsgBox " Real-time Alarm Count = " + CStr(AlmCount)
	End If
	End Sub

Item	Description
Type	LONG **GetGroupUserInfoByTag** (BSTR szTagName, LONG* lOutCount, SAFEARRAY(BSTR) *pszUserID, SAFEARRAY(BSTR) *pszUserName, SAFEARRAY(LONG) *plMailCheck, SAFEARRAY(LONG) *plSMSMCheck, SAFEARRAY(BSTR) *pszMailAdd, SAFEARRAY(BSTR) *pszHPhoneNo)
Kind	Alarm Function
Description	Get registered user information of the alarm group that contains specified tag by tag name. It is used to call the relevant user with tag information (alarm information) *szTagName*: Tag Name *lOutCount*: Count of the returned user information *pszUserID* : User ID *pszUserName*: User Name *plMailCheck*:: Check whether to receive a mail notice in case of the occurrence of alarm *plSMSMCheck*: Check whether to receive a SMS notice in case of the occurrence of alarm *pszMailAdd*: Mail address registered by the user *pszHPhoneNo*: Mobile phone number registered by the user
Example	Sub Main Dim lOutCount As Long Dim aUserID() As String Dim aUserName() As String Dim aMailCheck() As Long Dim aSMSCheck() As Long Dim aMailAdd() As String Dim aHPhoneNo() As String ole.GetGroupUserInfoByTag("AlmGrpTestTag", lOutCount, aUserID, aUserName, aMailCheck, aSMSCheck, aMailAdd, aHPhoneNo) For i = 0 To lOutCount MsgBox (aUserID(i) + " " + aUserName(i) + " " + CStr(aMailCheck(i)) + " " + CStr(aSMSCheck(i)) + " " + aMailAdd(i) + " " + aHPhoneNo(i)) Next i End Sub

Item	Description
Type	LONG **GetHistAlmCount** (LONG lKind, LONG lStatus)
Kind	Alarm Function
Description	Inform the history alarm count *lKind*: Alarm Kind 0 : All, 1 : Process Alarm, 2 : System Alarm, 3 : Process + System, 4 : Event, 5 : Process + Event, 6 : System + Event *lStatus*: Alarm Status 0 : All, 1 : UnRecover, 2 : UnAck

Example	Sub Main
	Dim AlmCount As Long
	AlmCount = ole. GetHistAlmCount (1, 0)
	
	If AlmCount > 0 Then
	MsgBox “History Alarm Count = “ + CStr(AlmCount)
	End If
	
	End Sub

Item	Description
Type	LONG **GetUserInfoList**(LONG* lOutCount, SAFEARRAY(BSTR) *pszUserID, SAFEARRAY(BSTR) *pszUserName, SAFEARRAY(BSTR) *pszDesc, SAFEARRAY(BSTR) *pszPassword, SAFEARRAY(LONG) *plUserLevel, SAFEARRAY(BSTR) *pszHPhoneNo, SAFEARRAY(BSTR) *pszPhoneNo, SAFEARRAY(BSTR) *pszMailAdd)
Kind	Alarm Function
Description	The function returns all user's list and informations. *lOutCount*: Returned list count *pszUserID* : User ID *pszUserName*: User name *pszDesc* : User's description *pszPassword* : User's password *plUserLevel* : User's level

	pszHPhoneNo: User's cell phone number *pszPhoneNo*: User's phone number *pszMailAdd*: User's e-mail address
Example	Sub Main Dim lRet As Long Dim lOutCount As Long Dim aUserID() As String Dim aUserName() As String Dim aDesc() As String Dim aPassword() As String Dim aUserLevel() As Long Dim aMailAdd() As String Dim aHPhoneNo() As String Dim aPhoneNo() As String lRet = ole.GetUserInfoList(lOutCount, aUserID, aUserName, aDesc, aPassword, aUserLevel, aHPhoneNo, aPhoneNo, aMailAdd) For i = 0 To lOutCount - 1 MsgBox (aUserID(i) + " " + aUserName(i) + " "+ CStr(aUserLevel(i))+ " " + aHPhoneNo(i) + " " + aPhoneNo(i)+ " " + aMailAdd(i)) Next i End Sub

Item	Description
Type	LONG **SetUserAlarm** (BSTR szAlarmMessage, BSTR szAlarmGroup)
Kind	Alarm Function
Description	This is the alarm function that user occurs. The alarm occurrence and recovery are generated at the same time *szAlarmMessage*: Alarm Message *szAlarmGroup* : Alarm Group Automatic entry *user* : Current logon user *Occured Time*, *Recovery Time* : current system time *Alarm type* : Process Alarm *AlarmLevel*: Input 10 level. Input "$Script" in tag name

Example	Sub Main ole.SetUserAlarm "UserAlarm ", "AlarmGroup" End Sub

Item	Description
Type	LONG **SetUserAlarmEx** (DOUBLE dOccur, LONG lAlarmLevel, BSTR szAlarmMessage ,BSTR szAlarmGroup, DOUBLE dCur, DOUBLE dPrev)
Kind	Alarm Function
Description	This is the extended alarm function that user occurs. The alarm occurrence and recovery are generated at the same time *dOccur*: Time Input - If you set this value by 0, current time value will be applied *lAlarmLevel*: Alarm Level - If you set this value from 0 to 10, then setting value will be applied. Else 10 will be applied automatically *szAlarmMessage* : Alarm Message *szAlarmGroup* : Alarm Group *dCur* : Current Value *dPrev* : Previous Automatic entry *user* : Current logon user *Alarm type* : Process Alarm Input "$Script" in tag name
Example	Sub Main ole.SetUserAlarmEx Now(), 2, " UserAlarm ", "AlarmGroup", _ 22.34, 44.56 End Sub

Item	Description
Type	LONG **SetUserEvent** (BSTR szEventMessage)
Kind	Alarm Function
Description	This is the event function that user occurs. *szEventMessage* : Event Message Auto input Item E*vent time* : Current system tme *Event occur* : Current logon user *Occurred Computer*: This Computer Name

	Alarm type : Event Alarm
Example	Sub Main ole.SetUserEvent "UserEvent" End Sub

Item	Description
Type	VOID **StopAlarmSound** ()
Kind	Alarm Function
Description	The function stops playing sound of the alarm.
Example	Sub Main ole. StopAlarmSound End Sub

5.2.12 External Device Function

Item	Description
Type	VOID **PlaySound**(BSTR szWaveFile)
Kind	External Device Function
Description	Play sound (Async) *szSoundFile* : wave file path
Example	Sub Main ole.PlaySound ("wave file .wav") End Sub

Item	Description
Type	VOID **PlaySoundFile** (BSTR szSoundFile)
Kind	External Device Function
Description	Play sound (Sync) *szSoundFile* : wave file path
Example	Sub Main ole.PlaySoundFile("SoundFile.wav") End Sub

5.2.13 User Function

Item	Description
Type	BSTR **GetUserID** ()
Kind	User Function
Description	The function returns ID of the user

Example	Sub Main
	  Dim szUserID As String
	  szUserID = ole. GetUserID
	  MsgBox " User ID = “szUserID
	End Sub

Item	Description
Type	LONG **GetUserLevel** ()
Kind	User Function
Description	The function returns Level of the user

Example	Sub Main
	  Dim lRet As Long
	  lRet = ole. GetUserLevel
	  MsgBox " User Priority = “+ CScr(lRet)
	End Sub

Item	Description
Type	LONG **UserLogOff** ()
Kind	User Function
Description	Notifies InfoU Runtime that a user has logged off

Example	Sub Main
	  Dim lRet As Long
	  lRet = ole.UserLogOff
	  If lRet <> 0 Then
	  MsgBox "UserLogOff() False"
	  Else
	  MsgBox "UserLogOff() OK"
	  End If
	End Sub

Item	Description
Type	LONG **UserLogOn** (BSTR szUserID, BSTR szPassWord)
Kind	User Function
Description	Notifies InfoU Runtime that a user has logged on *szUserID*: User ID *szPassWord*: User Password

Example	Sub Main
	  Dim lRet As Long
	
	  lRet = ole.UserLogOn("UserID", " UserPassword ")
	  If lRet <> 0 Then
	  MsgBox "UserLogOn() False"
	  Else
	  MsgBox "UserLogOn() OK"
	  End If
	End Sub

5.2.14 Report Function

Item	Description
Type	VOID **ReportDialog** ()
Kind	Report Function
Description	Shows the report windows
Example	Sub Main ole. ReportDialog End Sub

Item	Description
Type	VOID **ReportListRefresh**()
Kind	Report Function
Description	Refresh a report list
Example	Sub Main ole. ReportListRefresh End Sub

Item	Description
Type	VOID **ReportPrint** (BSTR szReportName)
Kind	Report Function
Description	Outputs the report *szReportName* : report name
Example	Sub Main ole. ReportPrint (“ReportName”) End Sub

Item	Description
Type	VOID **ReportPrintEx** (BSTR szReportName, BSTR sz ReqTime)
Kind	Report Function
Description	Outputs the report *szReportName* : report name *szReqTime* : request report start time
Example	Sub Main ole. ReportPrintEx (“ReportName”, ”2007/02/01”) End Sub

Item	Description
Type	VOID **ReportSchedulerHide** ()
Kind	Report Function
Description	Hide the report schedule windows
Example	Sub Main ole.ReportSchedulerHide End Sub

Item	Description
Type	VOID **ReportSchedulerShow** ()
Kind	Report Function
Description	Shows report schedule list
Example	Sub Main ole. ReportSchedulerShow End Sub

5.2.15 History Data Function

Item	Description
Type	VOID **GetHistDataCount** (BSTR szModelName, BSTR , BSTR szDataKind, DOUBLE dRefTime, LONG lCount, LONG lDir, LONG* lResultCount, SAFEARRAY(double) *psadblTimeStamp, SAFEARRAY(double) *psadblNumValue, SAFEARRAY(LONG) *psalStatus, SAFEARRAY(LONG) *psalResult)
Kind	History Data Function
Description	This function returns logging data from the logging group you specified according to the base time, direction and logging data count. (TimeStamp, Tag Value, Tag Status, Result) *szModelName* : Logging Group Name *szTagName* : Logging Tag Name *szDataKind*: Value Type (Default : Current Value) *dRefTime*: Base Time *lCount*, : Logging Data Count *lDir*: Direction (forward or backward from base time) *lResultCount*: Logging Data Count *psadblTimeStamp*: Array of TimeStamps *psadblNumValue*: Array of Tag Values *psalStatus*: Array of Tag Status *psalResult: Result*
Example	Sub Main Dim ModelName As String Dim TagName As String Dim DataKind As String Dim RefTime As Date Dim lCount As Long Dim lDir As Long Dim ResultCount As Long Dim TimeStamp() As Double Dim NumValue() As Double Dim Status() As Long Dim Result() As Long ModelName = "INV1_SEC" TagName = "INV_1.1_AI_CT" DataKind = "Current" RefTime = Now() lCount = 5

	lDir = 1 ole.GetHistDataCount(ModelName, TagName, DataKind, RefTime, lCount, lDir, ResultCount, TimeStamp, NumValue, Status, Result) For count = 0 To ResultCount -1 MsgBox("NumValue = " + CStr(NumValue(count)) + " Status = " + CStr(Status (count))) Next End Sub

Item	Description
Type	VOID **GetHistDataPer**iod (BSTR szModelName, BSTR szTagName, BSTR szDataKind, DOUBLE dFrom, DOUBLE dTo, LONG* lResultCount, SAFEARRAY(double) *psadblTimeStamp, SAFEARRAY(double) *psadblNumValue, SAFEARRAY(LONG) *psalStatus, SAFEARRAY(LONG) *psalResul)
Kind	History Data Function
Description	This function returns logging data from logging group you specified within a given period of time. (TimeStamp, Tag Value, Tag Status, Result) *szModelName* : Logging Group Name *szTagName* : Logging Tag Name *szDataKind*: Value Type (Default : Current Value) *dFrom*: Beginning Time *dTo*: End Time *lResultCount*: Logging Data Count *psadblTimeStamp*: Array of TimeStamps *psadblNumValue*: Array of Tag Values *psalStatus*: Array of Tag Status *psalResult:* Result
Example	Sub Main Dim ModelName As String Dim TagName As String Dim DataKind As String Dim dFrom As Date Dim dTo As Date Dim ResultCount As Long Dim TimeStamp() As Double

```
  Dim NumValue() As Double
  Dim Status() As Long
  Dim Result() As Long

  ModelName = "INV1_SEC"
  TagName = "INV_1.1_AI_CT"
  DataKind = "Current Value"
  dFrom = Now() -1
  'To = Now()
  ole.GetHistDataPeriod( ModelName, TagName, DataKind, _
                                    dFrom, dTo, ResultCount, TimeStamp, _
                                    NumValue, Status, Result)
  For count = 0 To ResultCount -1
    MsgBox("NumValue = " + CStr(NumValue(count))
                    + "    Status = " + CStr(Status (count)))
  Next
End Sub
```

Item	Description
Type	LONG **GetLoggingModelList** (LONG *lOutCount ,SAFEARRAY(BSTR) *psaszModelName, SAFEARRAY(BSTR) *psaszModelDesc, SAFEARRAY(BSTR) *psaszModelType, SAFEARRAY(BSTR) *psaszArchvingCycle, SAFEARRAY(BSTR) *psaszSegmentPeriod)
Kind	History Data Function
Description	This function returns Logging Group Lists *lOutCount*: Logging Group count *psaszModelName*: List of Logging Group Names *psaszModelDesc*: List of Logging Group Descriptions *psaszModelType*: List of Logging Group Types *psaszArchvingCycle*: List of Acquisition Cycles *psaszSegmentPeriod:* List of Storage Periods
Example	Sub Main Dim lRet As Long Dim lOutCount As Long Dim aModelName() As String Dim aModelDesc() As String Dim aModelType() As String Dim aArchivingCycle() As String Dim aSegmentPeriod() As String

```
lRet = ole.GetLoggingModelList(lOutCount, aModelName, _
            aModelDesc, aModelType,aArchivingCycle, aSegmentPeriod )

For i = 0 To lOutCount -1
  MsgBox (aModelName(i) + " " + aModelDesc(i)  + " "+ _
                aModelType(i)+ " " + aArchivingCycle(i) +  _
                        " " + aSegmentPeriod(i))
Next i
End Sub
```

Item	Description
Type	LONG **GetLoggingTag**(BSTR szModelName, LONG *lOutCount ,SAFEARRAY(BSTR) *psaszTagName, SAFEARRAY(BSTR) *psaszTagType)
Kind	History Data Function
Description	This function returns Logging Tags according to the Logging Group. *szModelName*: Logging Group Name *lOutCount*: Logging Tag Count in the Logging Group *psaszTagName*: List of Tag Names *psaszTagType*: List of Tag Types
Example	Sub Main Dim lRet As Long Dim lOutCount As Long Dim aTagName() As String Dim aTagType() As String lRet = ole.GetLoggingTag ("INV1_MON", lOutCount, aTagName, aTagType) For i = 0 To lOutCount -1 MsgBox ("INV1_MON " + aTagName(i) + " " + aTagType(i)) Next i End Sub

Item	Description
Type	VOID **SetHistData** (BSTR szModelName, BSTR szTagName, BSTR szDataKind, LONG lDataCount, SAFEARRAY(double) *psadblSaveTime, SAFEARRAY(double) *psadblNumValue, SAFEARRAY(LONG) *psalTagStatus, SAFEARRAY(LONG) *psalResult)
Kind	History Data Function
Description	This function saves tag values as logging data in the logging group user specified. *szModelName* : Logging Group Name *szTagName* : Logging Tag Name *szDataKind*: Value Type (Default : Current Value *lDataCount* : Data Count *psadblSaveTime* : TimeStamp *psadblNumValue* : Tag Value psalTagStatus : Tag Status *psalResult:* Result

Example	Sub Main
	Dim szModelName As String
	Dim szTagName As String
	Dim szDataKind As String
	Dim lCount As Long
	Dim dtSaveTime(3) As Double
	Dim dblTagNumValue(3) As Double
	Dim ulTagStatus(3) As Long
	Dim lResult() As Long
	Dim lRet As Long
	szModelName = "INV1_SEC"
	szTagName = "INV_1.1_AI_CT"
	szDataKind = "Current Value"
	lCount = 3
	For count = 0 To lCount -1
	dtSaveTime(count) = Now() + count*0.1
	dblTagNumValue(count) = 100 + count + 1
	ulTagStatus(count) = 0
	Next
	lRet = ole.SetHistData(szModelName ,szTagName , szDataKind,
	lCount , dtSaveTime, dblTagNumValue, ulTagStatus, lResult)
	If lRet <> 0 Then
	MsgBox("SetHistData Error!!!1")
	Else

	For count = 0 To lCount -1 MsgBox("Result = " + CStr(lResult (count))) Next End If End Sub

5.2.16 I/O Driver Function

Item	Description
Type	VOID **ClearStaStatusCount** (BSTR szChannelName, BSTR szStaName)
Kind	I/O DRIVE FUNCTION
Description	Reset Station Status counts. *szChannelName* : Channel Name *szStaName* : Station Name
Example	Sub Main ole. ClearStaStatusCount (“ChannelName”, “StationName”) End Sub

Item	Description
Type	VOID **EnableStat**ion (BSTR szChannelName, BSTR szStaName, LONG lMode)
Kind	I/O DRIVE FUNCTION
Description	Set a station enable or disable. *szChannelName* : Channel Name *szStaName* : Station Name *lMode* : 0 : Enable, 1: Disable
Example	Sub Main Dim SuccessCount As Long SuccessCount = ole. EnableStation (“ChannelName”, StationName”,0) End Sub

Item	Description
Type	LONG **GetCountReadSuccess** (BSTR szChannelName, BSTR szStaName)
Kind	I/O DRIVE FUNCTION
Description	Return Read OK counts *szChannelName* : Channel Name *szStaName* : Station Name
Example	Sub Main Dim SuccessCount As Long SuccessCount = ole. GetCountReadSuccess (“ChannelName”,“StationName”) MsgBox “Read Success Count = “ + CStr(SuccessCount) End Sub

Item	Description
Type	LONG **GetCountReadTotal** (BSTR szChannelName, BSTR szStaName)
Kind	I/O DRIVE FUNCTION
Description	Return Read Total counts *szChannelName:* Channel Name *szStaName:* Station Name
Example	Sub Main Dim ReadCount As Long ReadCount = ole. GetCountReadTotal (“ChannelName”,“StationName”) MsgBox “Read Count = “ + CStr(ReadCount) End Sub

Item	Description
Type	LONG **GetCountWriteSuccess** (BSTR szChannelName, BSTR szStaName)
Kind	I/O DRIVE FUNCTION
Description	Return Write OK counts. *szChannelName:* Channel Name *szStaName:* Station Name
Example	Sub Main Dim SuccessCount As Long SuccessCount = ole. GetCountWriteSuccess (“ChannelName”,“StationName”) MsgBox “Write Success Count = “ + CStr(SuccessCount) End Sub

Item	Description
Type	LONG **GetCountWriteTotal** (BSTR szChannelName, BSTR szStaName)
Kind	I/O DRIVE FUNCTION
Description	Return Write Total counts. *szChannelName:* Channel Name *szStaName:* Station Name
Example	Sub Main Dim WriteCount As Long WriteCount = ole. GetCountWriteTotal (“ChannelName”,“StationName”) MsgBox “Write Count = “ + CStr(WriteCount) End Sub

Item	Description
Type	LONG **GetStatusRunMode** (BSTR szChannelName, BSTR szStaName)
Kind	I/O DRIVE FUNCTION
Description	Return Station Runmode. *szChannelName:* Channel Name *szStaName:* Station Name
Example	Sub Main Dim RunModeCount As Long RunModeCount = ole. GetStatusRunMode (“ChannelName”,“StationName”) MsgBox “Station Run Mode Count = “ + CStr(RunModeCount) End Sub

Item	Description
Type	LONG **GetStatusStaOK** (BSTR szChannelName, BSTR szStaName)
Kind	I/O DRIVE FUNCTION
Description	Return Station OK status. *szChannelName:* Channel Name *szStaName:* Station Name
Example	Sub Main Dim OKCount As Long OKCount = ole. GetStatusStaOK (“ChannelName”,“StationName”) MsgBox “OK Count = “ + CStr(OKCount) End Sub

Item	Description
Type	VOID **ShowNetworkStatus** ()
Kind	I/O DRIVE FUNCTION
Description	Display Station Status dialog.
Example	Sub Main ole.ShowNetworkStatus End Sub

Item	Description
Type	VOID **ShowStationInfo** ()
Kind	I/O DRIVE FUNCTION
Description	Display Station Status dialog.
Example	Sub Main ole.ShowStationInfo End Sub

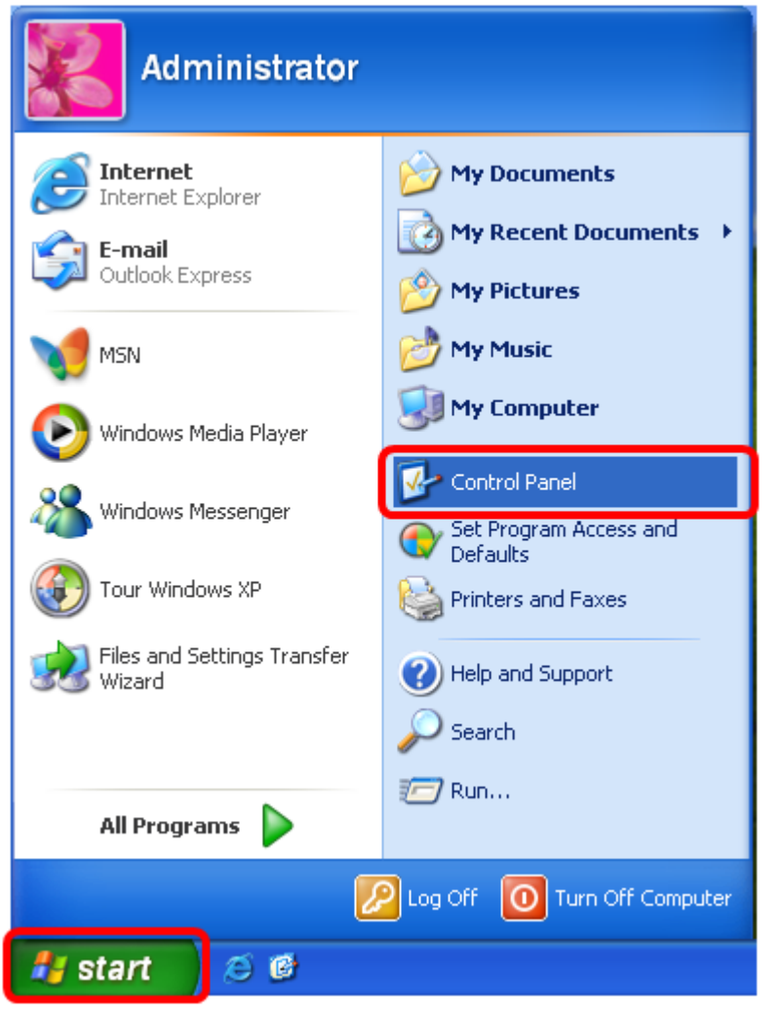
Appendix 6 InfoU Web Q&A Collection

6.1 Q&A for Windows XP

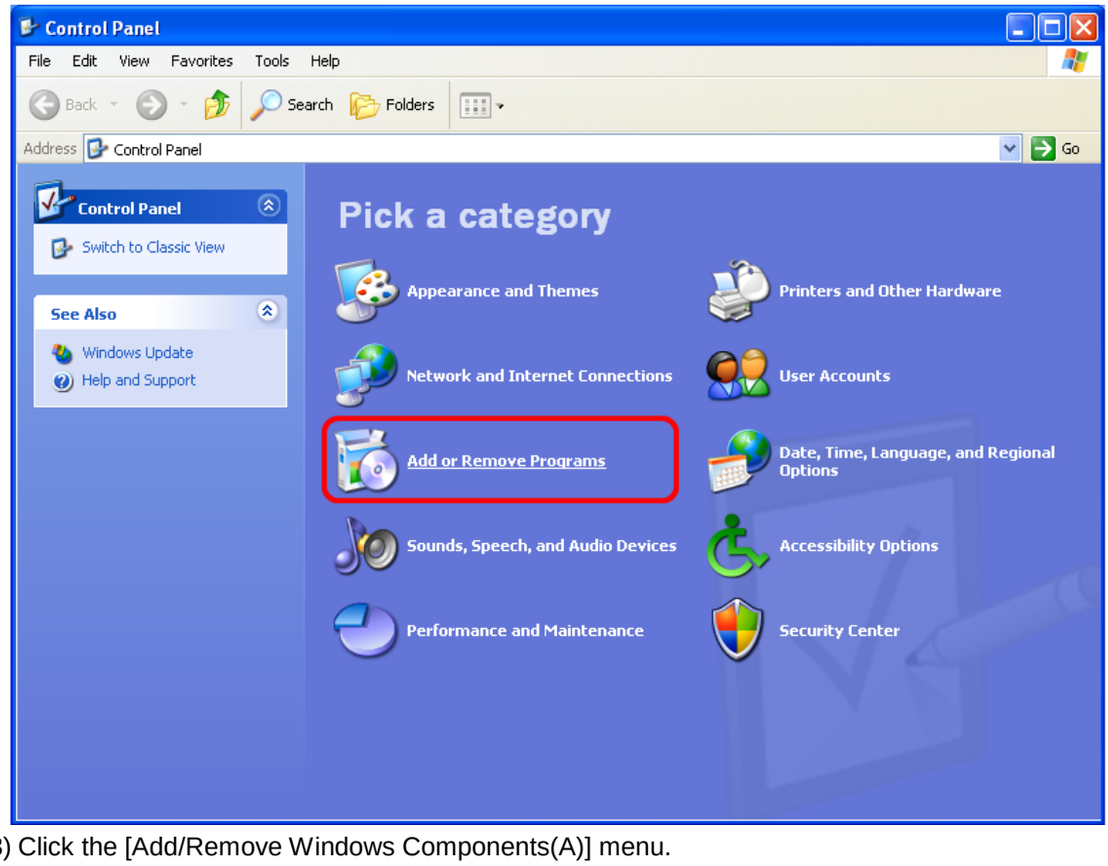
6.1.1 Q: How can I install the Internet Information Service (IIS) in Windows XP?

A: In order to install the internet information service (hereinafter referred to as 'IIS') in Windows XP, you need the CD to install Windows XP OS in your computer. On condition that you have the Windows XP OS installation CD, you can refer to the below installation procedures. Namely, if you do not have the OS installation CD, you cannot install IIS.

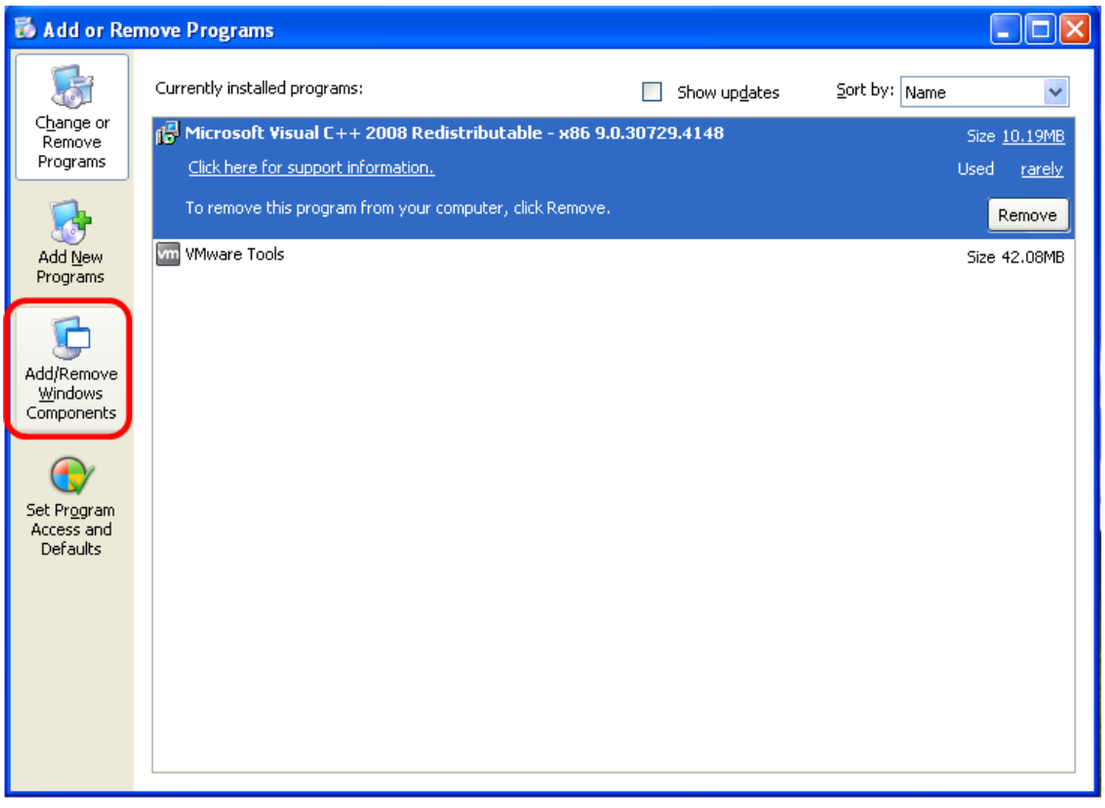
(1) Click the [Start]➔[Control Panel] menu.



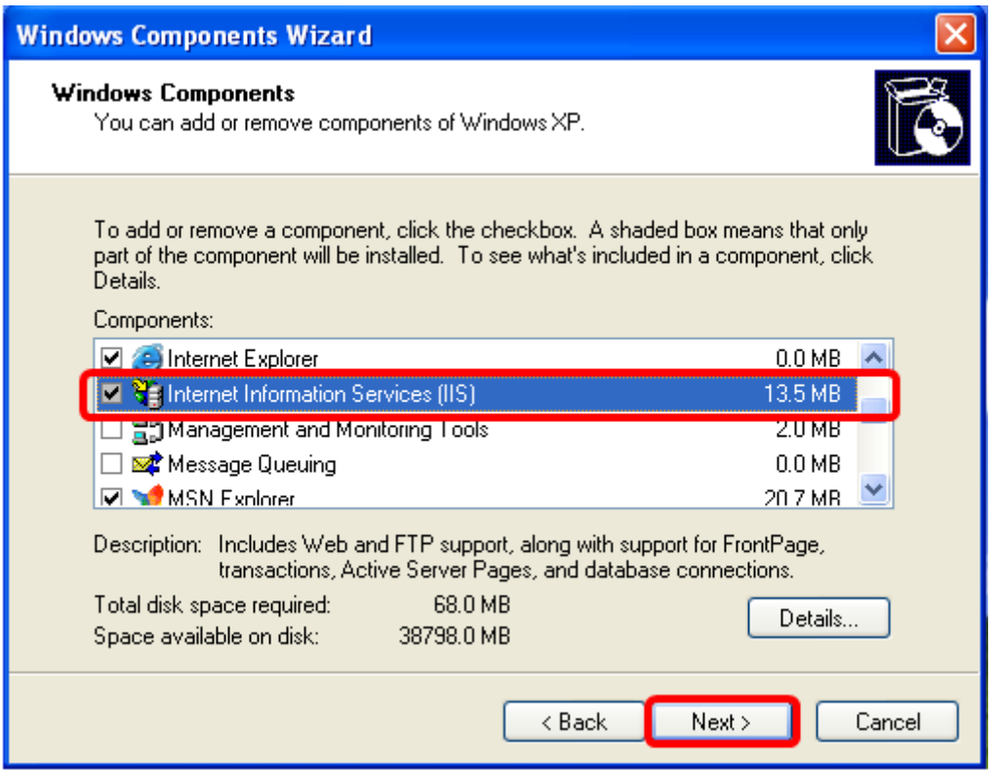
(2) Click the [Add/Remove Programs] in the [Control Panel].



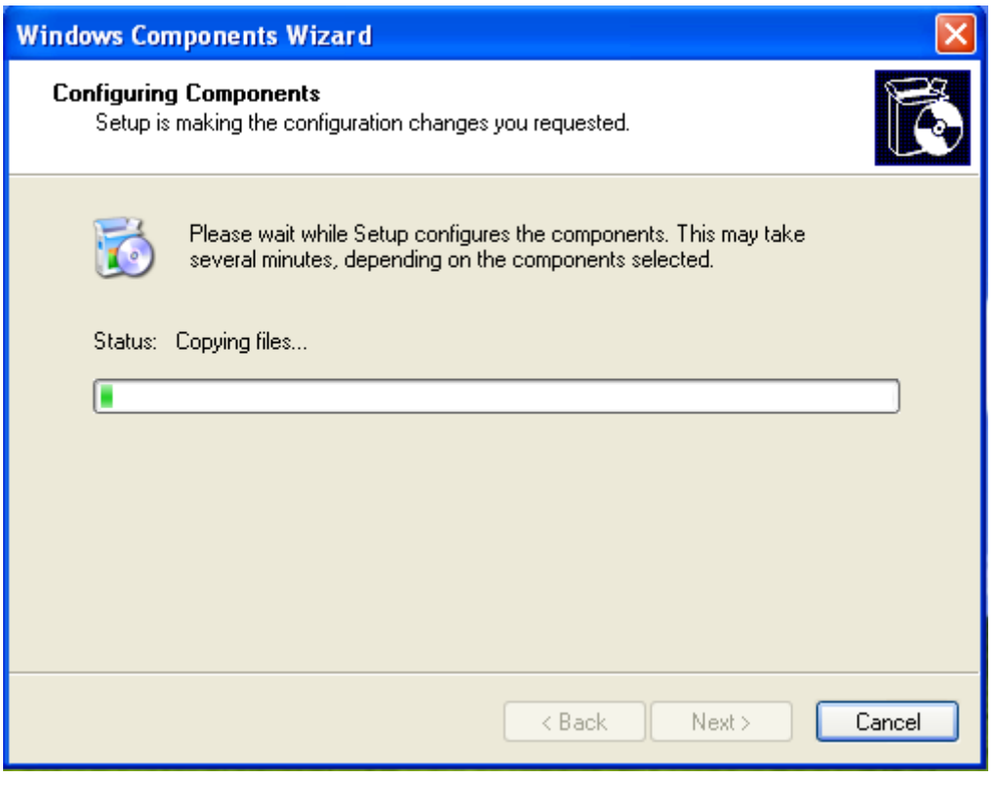
(3) Click the [Add/Remove Windows Components(A)] menu.

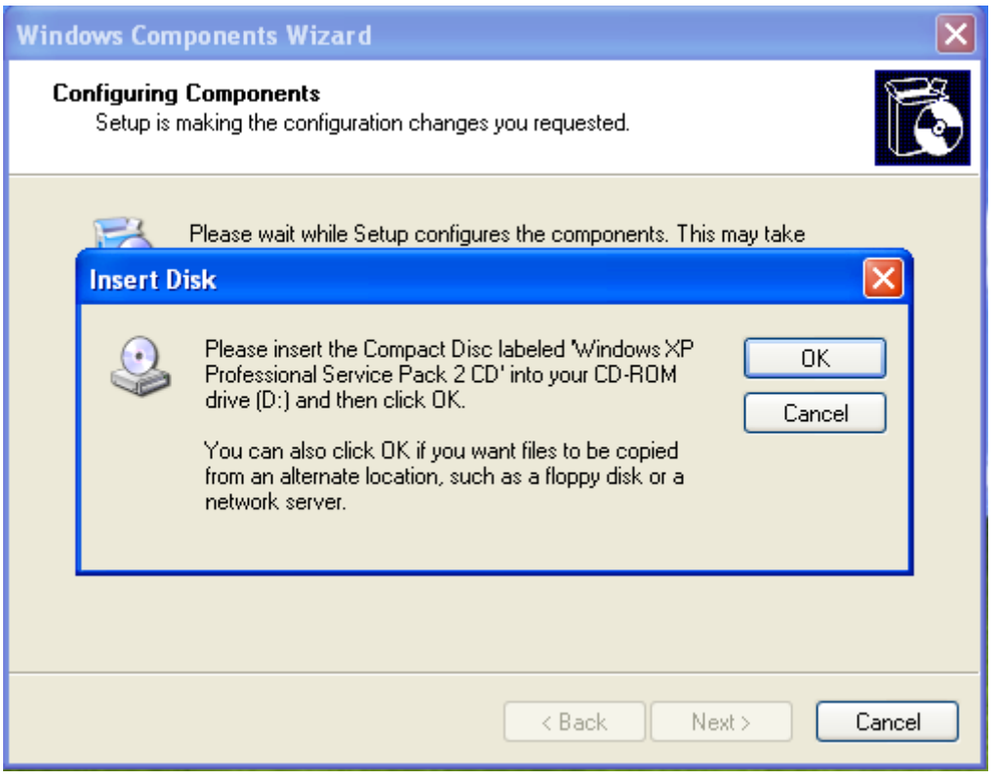


(4) After selecting internet information service (IIS) in the dialog box of [Windows Components Wizard], click the [Next] button.

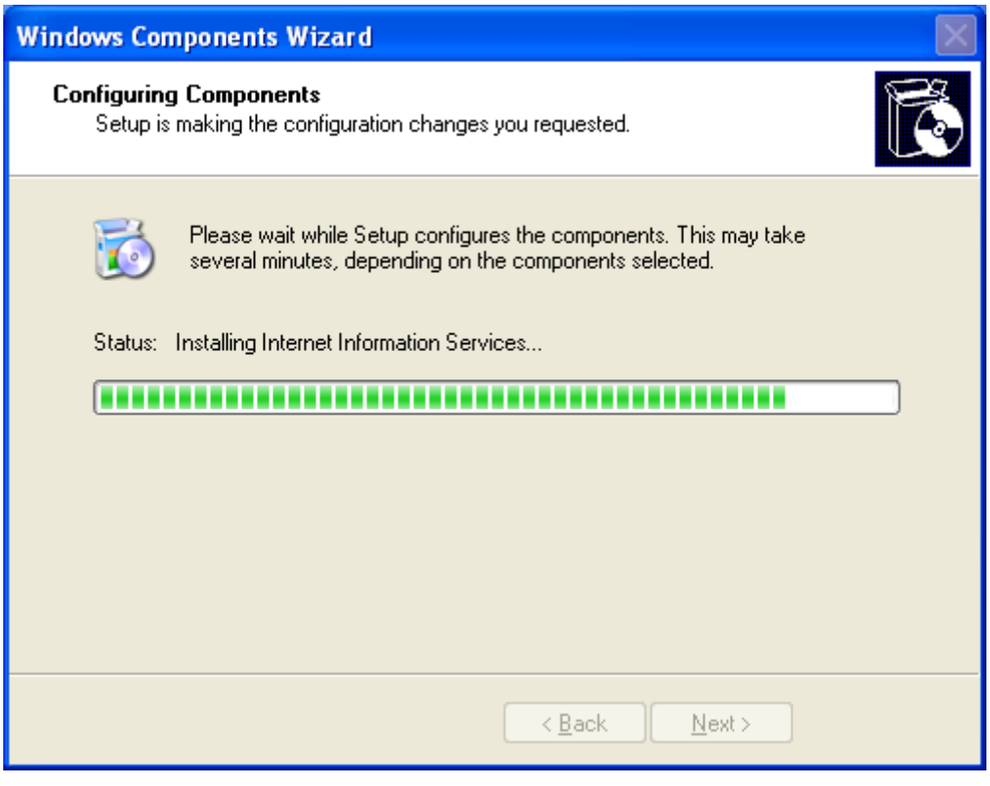


If the message is displayed; 'There is no Windows installation CD', insert the CD', insert the CD and wait until file copying is completed.





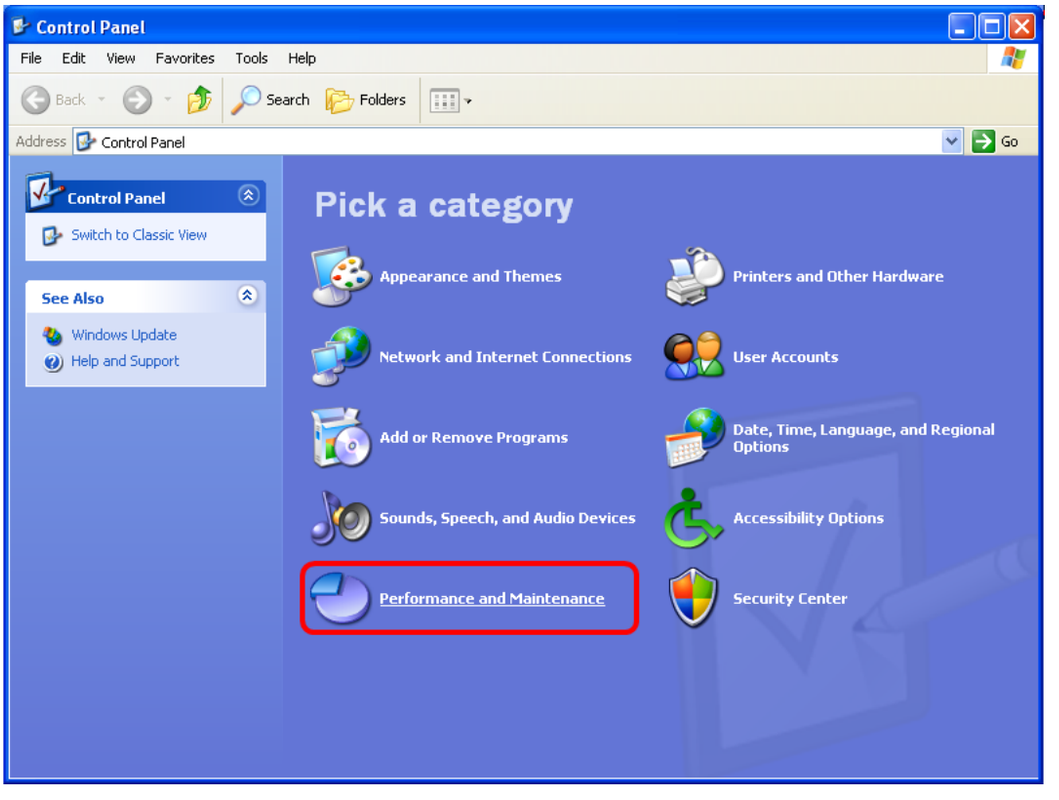
It shows the process copying file from the Windows original CD.



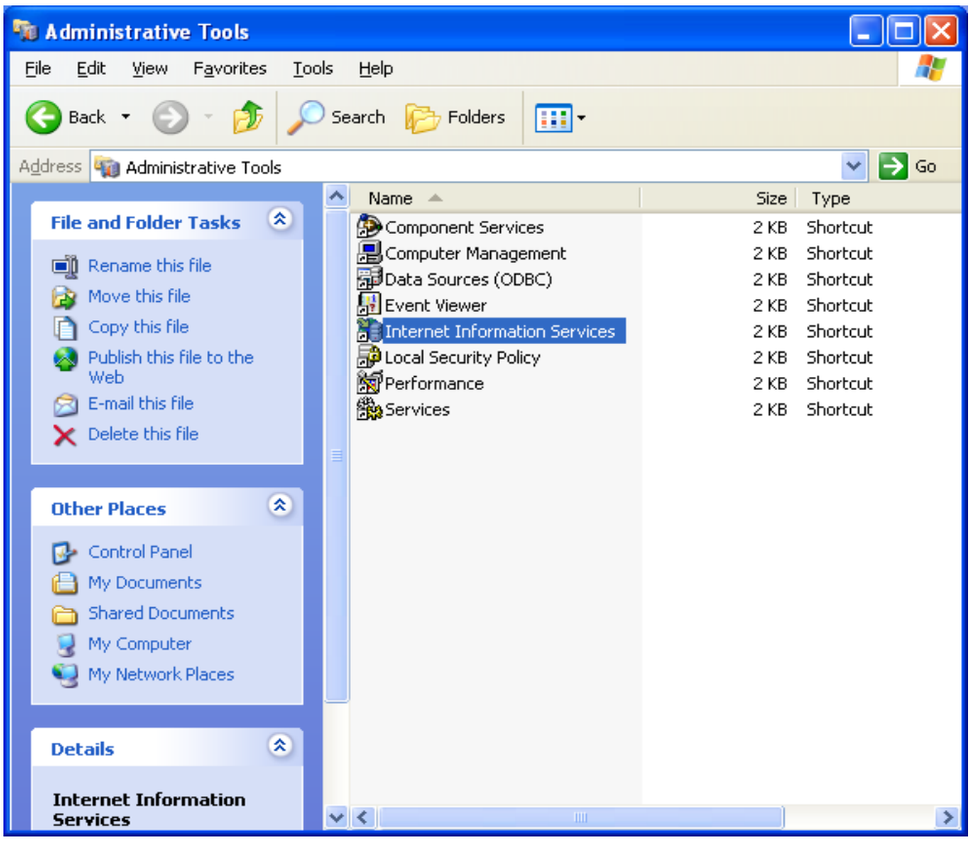
(5) If the message is displayed; Windows Components Wizard is completed, click the [Finish] button.



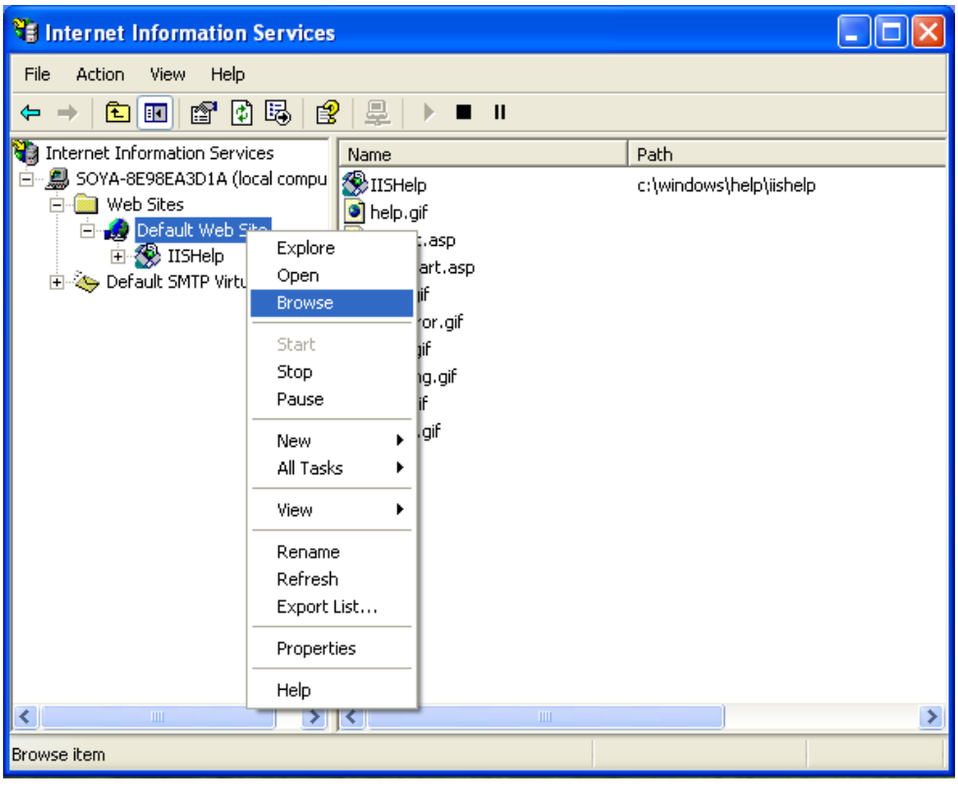
(6) Select [Start]➔[Control Panel]➔[Performance and Maintenance].



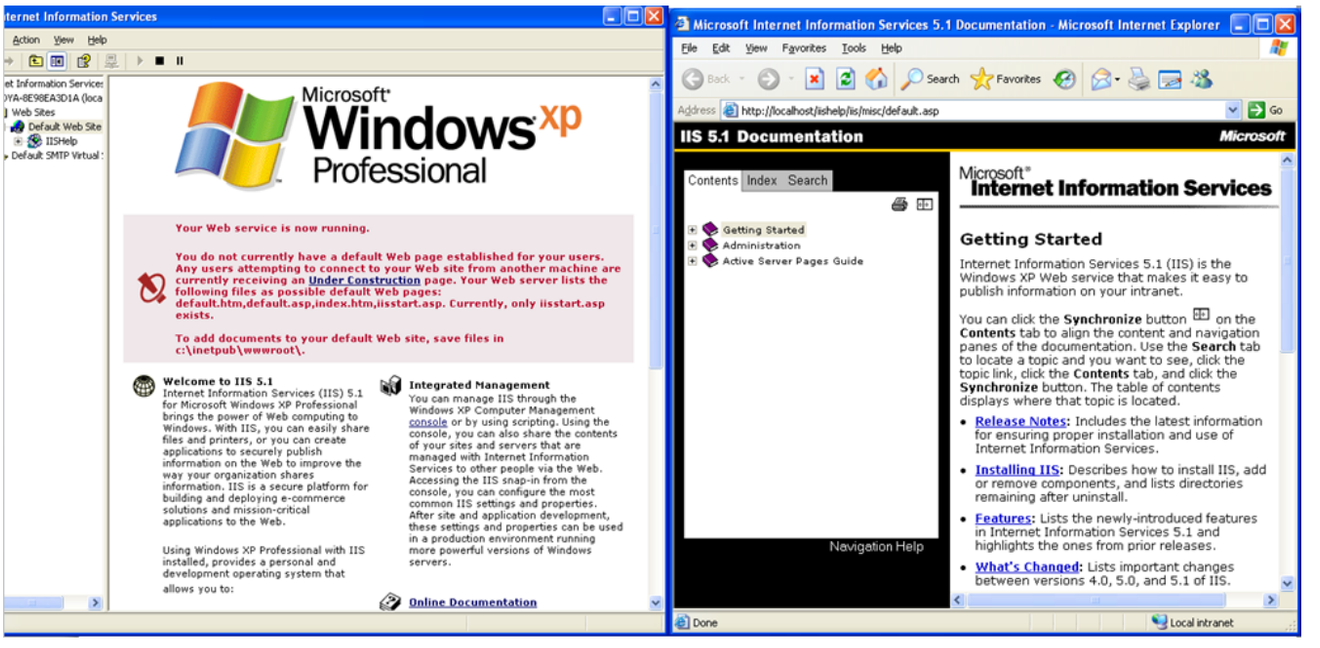
(7) Select [Administrative Tools]➔[Internet Information Service].



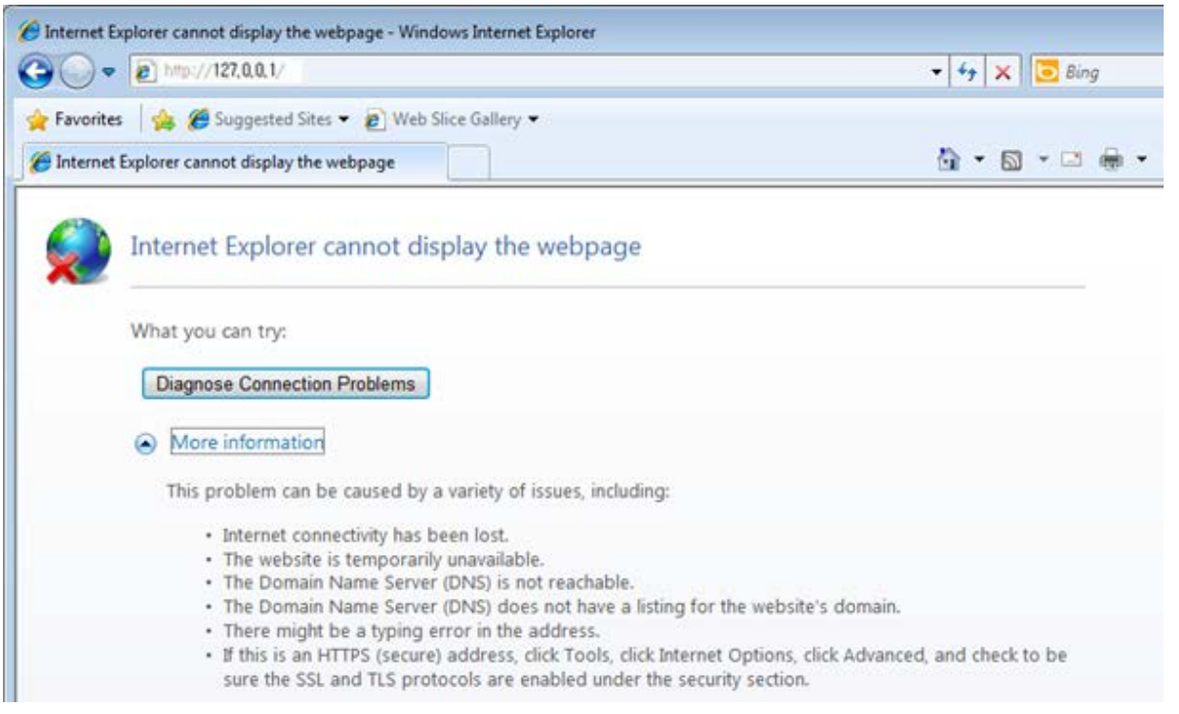
After selecting [Default Web Site], click with the right mouse button. Then, choose the [Browse] menu.



(8) When IIS is normally installed, the default page will show up as below. Alternatively, if you open the web browser and input the local host, the default page will also be displayed.

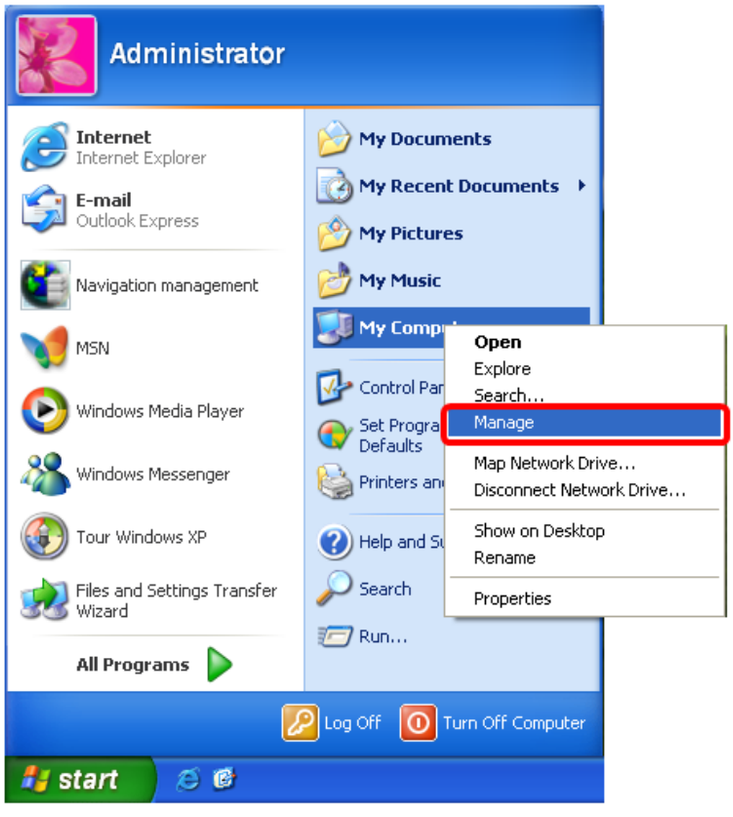


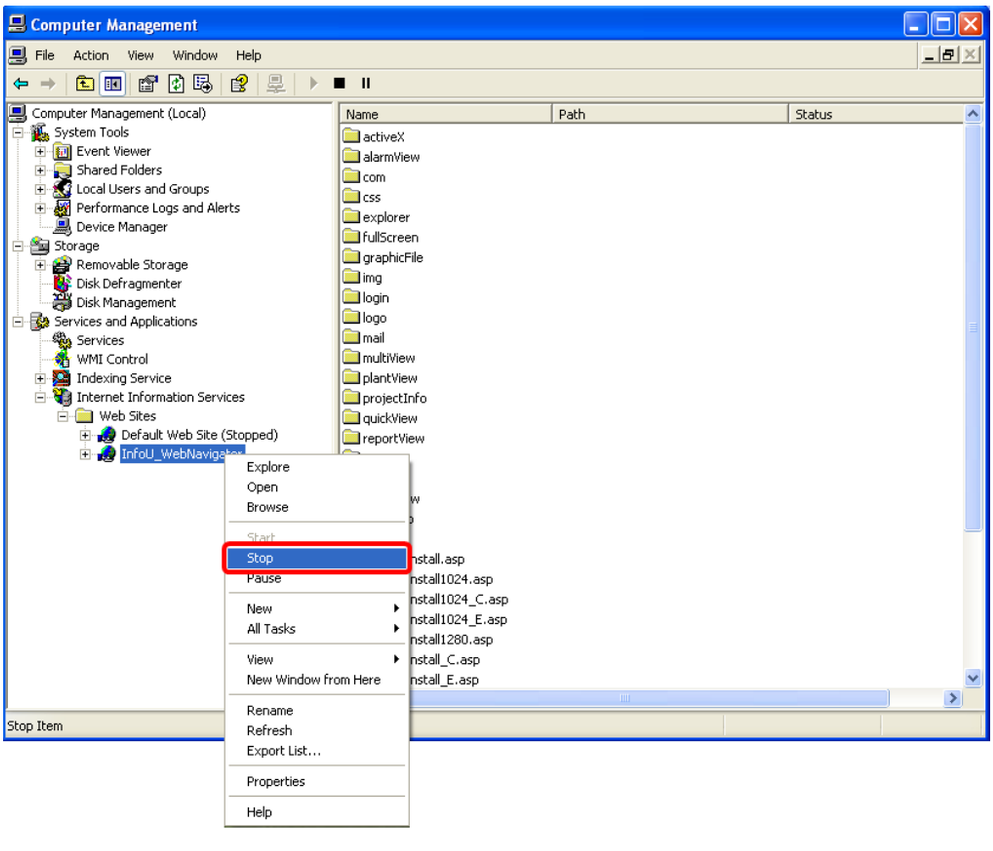
6.1.2 Q: After installing IIS in Windows XP and operating the web server, if I access to http://127.0.0.1, the below message is displayed; Internet Explorer cannot display the webpage.

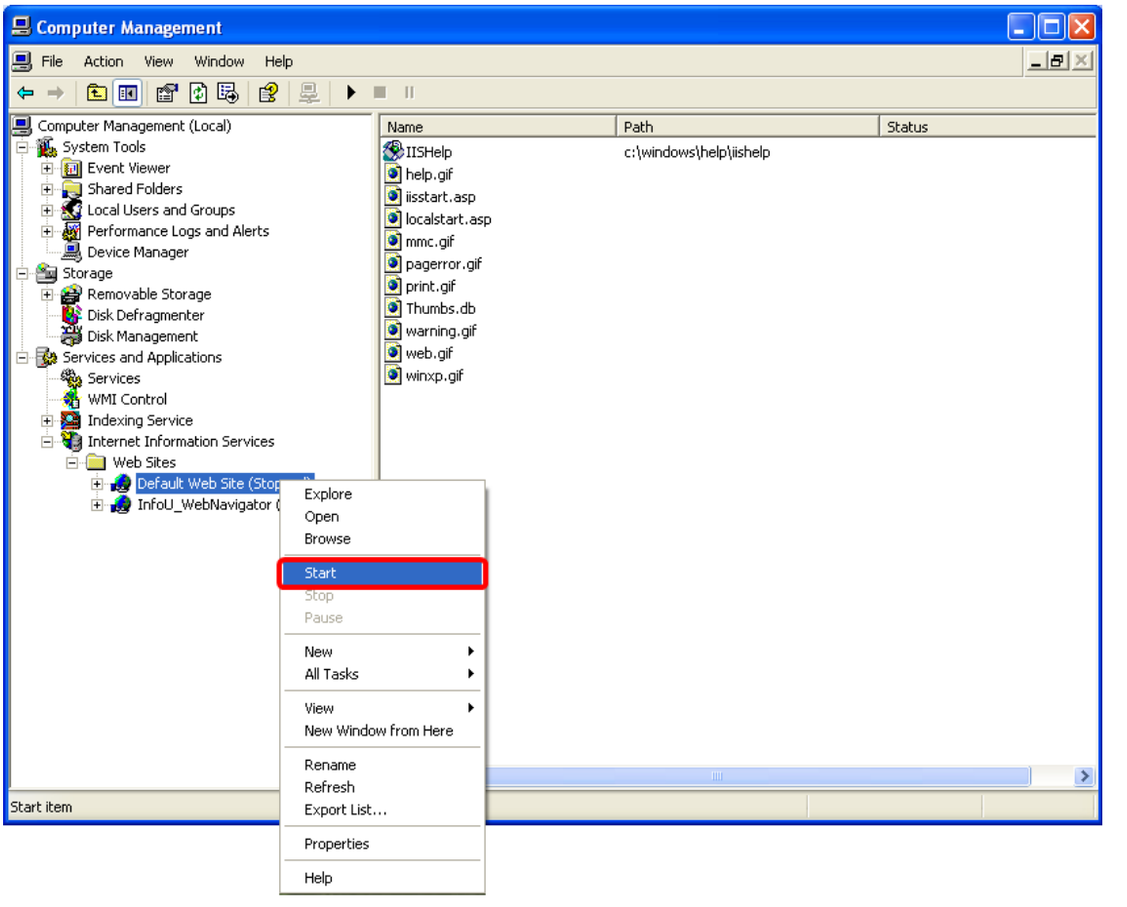


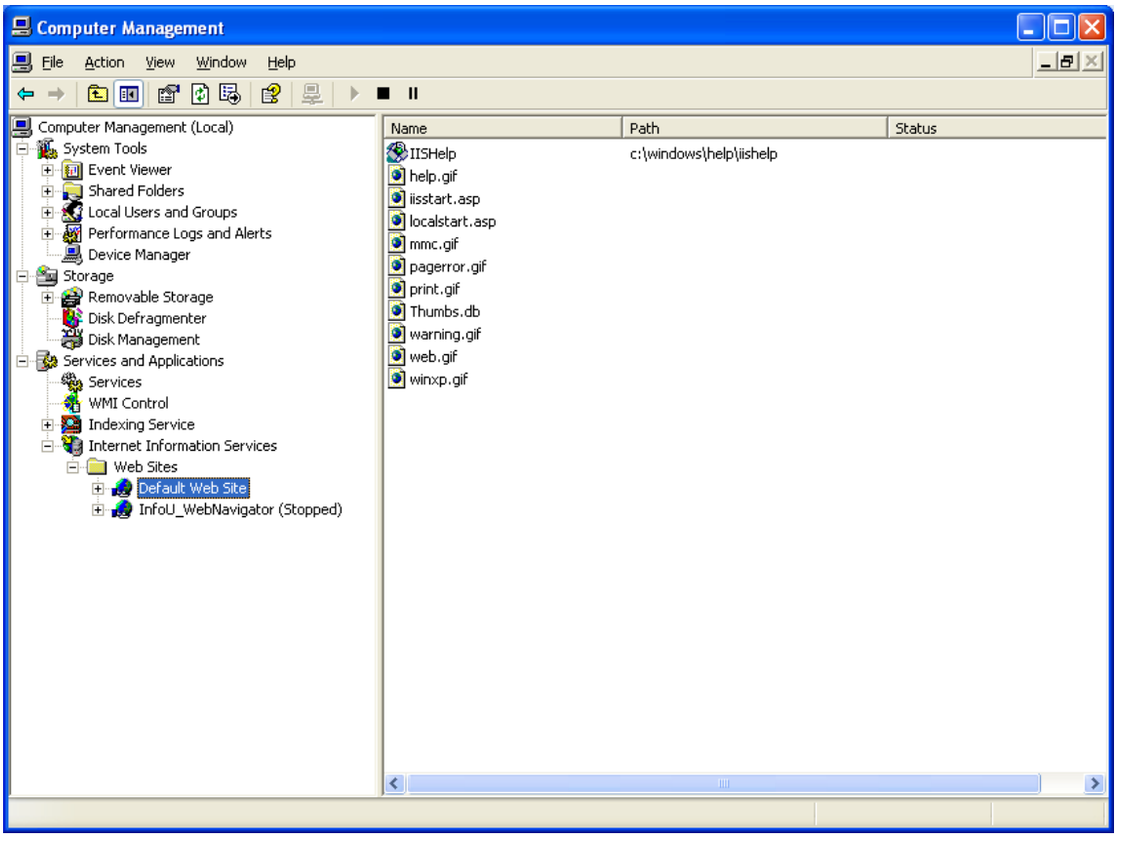
A: In case the above massage; [Internet Explorer cannot display the webpage] shows up instead of IIS error code page, it means IIS is not normally installed.

(1) Select [Start]➔[My computer]➔[Right Mouse]➔[Manage]. In [Service and Applications]➔ [Internet Information Service]➔[Web Sites], stop all other websites and change [Default Web Site] into 'Start'.

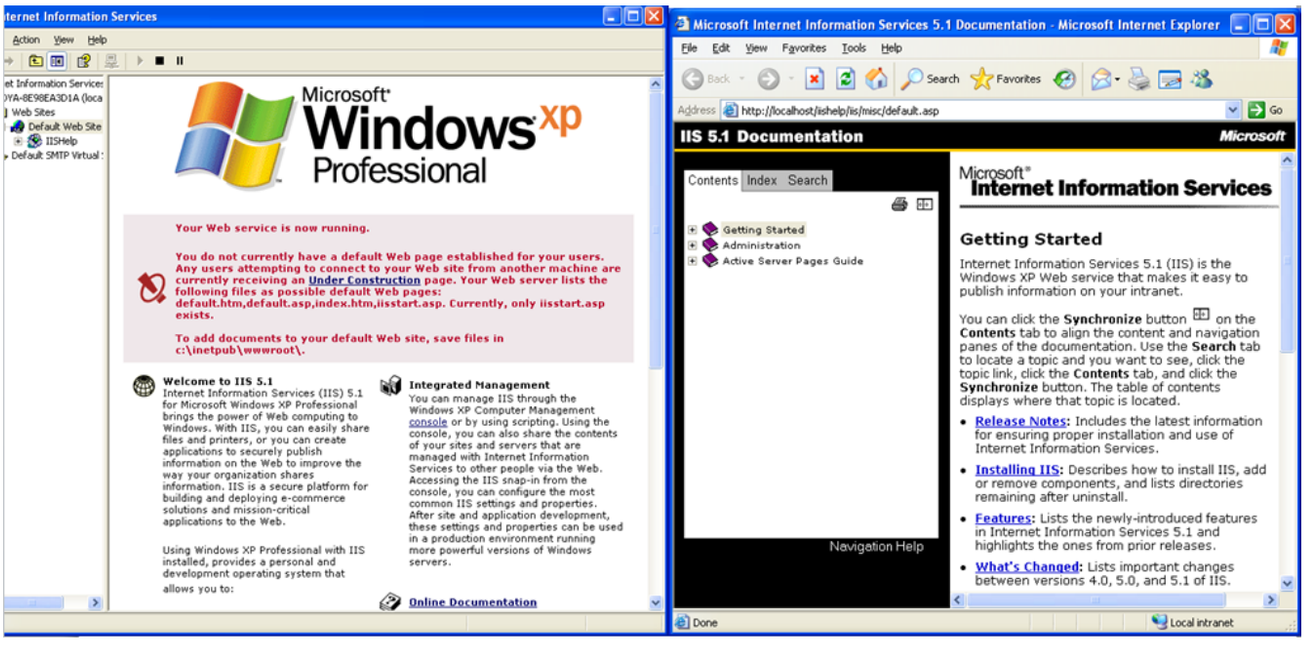





Computer Management
File Action View Window Help
Computer Management (Local)
System Tools
Event Viewer
Shared Folders
Local Users and Groups
Performance Logs and Alerts
Device Manager
Storage
Removable Storage
Disk Defragmenter
Disk Management
Services and Applications
Services
WMI Control
Indexing Service
Internet Information Services
Web Sites
Default Web Site (Stop
InfoU_WebNavigator (
Name
Path
Status
IISHelp
c:\windows\help\iishelp
help.gif
iisstart.asp
localstart.asp
mmc.gif
pagerror.gif
print.gif
Thumbs.db
warning.gif
web.gif
winxp.gif
Explore
Open
Browse
Start
Stop
Pause
New
All Tasks
View
New Window from Here
Rename
Refresh
Export List...
Properties
Help
Start item

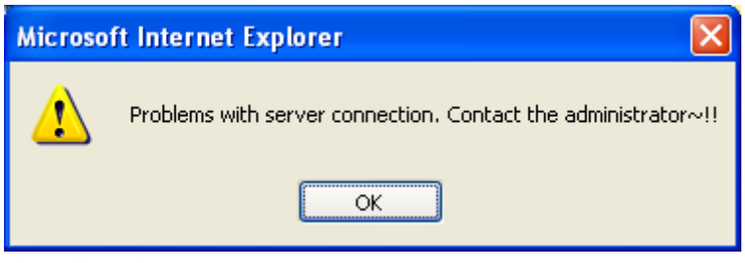

Computer Management
File Action View Window Help
Computer Management (Local)
System Tools
Event Viewer
Shared Folders
Local Users and Groups
Performance Logs and Alerts
Device Manager
Storage
Removable Storage
Disk Defragmenter
Disk Management
Services and Applications
Services
WMI Control
Indexing Service
Internet Information Services
Web Sites
Default Web Site
InfoU_WebNavigator (Stopped)
Name
Path
Status
IISHelp
c:\windows\help\iishelp
help.gif
iisstart.asp
localstart.asp
mmc.gif
pagerror.gif
print.gif
Thumbs.db
warning.gif
web.gif
winxp.gif

(2) Open the web browser and input http://localhost or http://127.0.0.1. Then, the default page will be normally opened as shown below. If the default page is not opened, it means IIS installation is wrong or there are some problems with the components of IIS. In this case, after removing IIS, reinstall it.

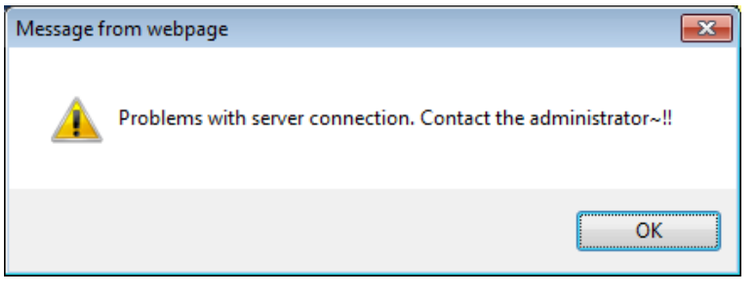


6.1.3 Q: When entering the ID and password in the login screen, the below message shows up.

(1) Windows XP version



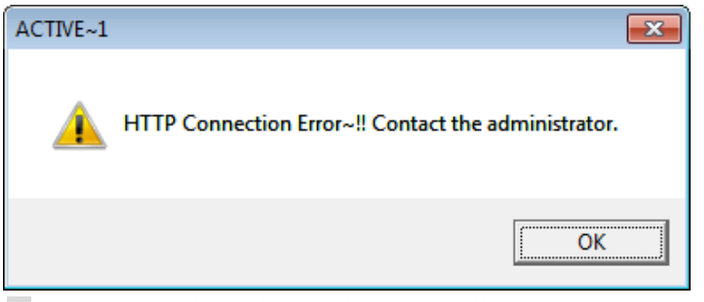
(2) Windows 7 version



A: It means the project information was not obtained properly. After running the [Navigation Management], select [Get project information] to read the project information.

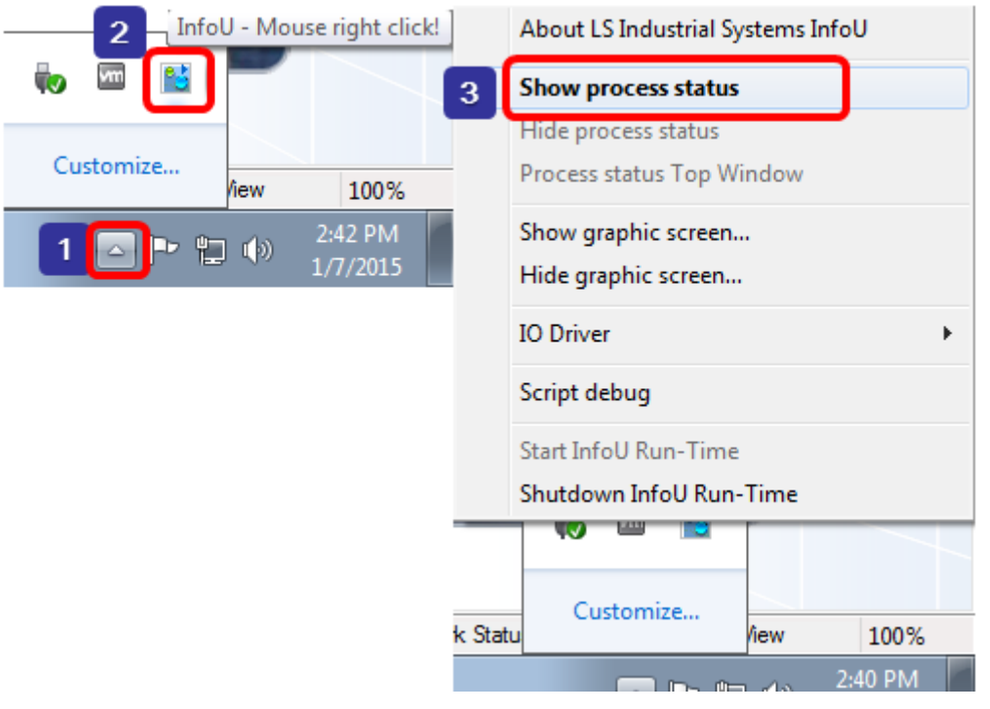


6.1.4 Q: When logging in to the site, the page is executed and the below message is displayed.

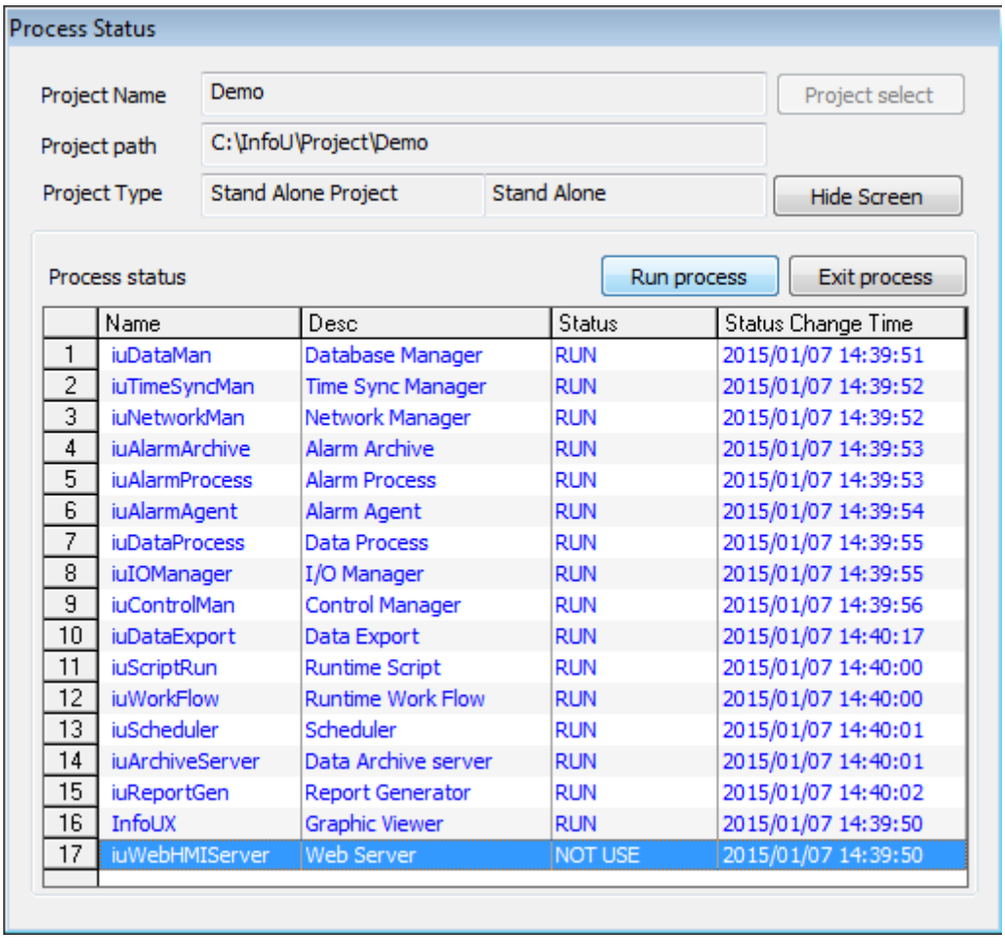


A: To access to the InfoU Web site, real-time operation should run and the Web HMI Server should be running. To run the Web HMI Server, follow the procedures shown below.

(1) After selecting the tray icon of real-time operation, if you click with the right mouse button, the below screen will show up.

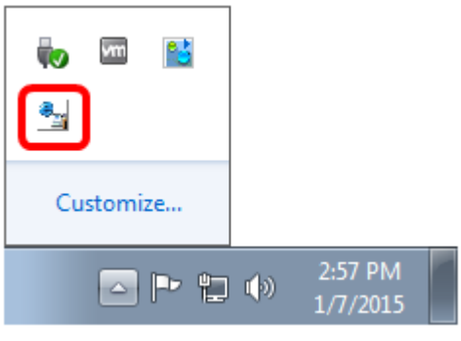


(2) Select [Show Process Status]. After selecting 17th item, iuWebHMIServe, click [Run process].

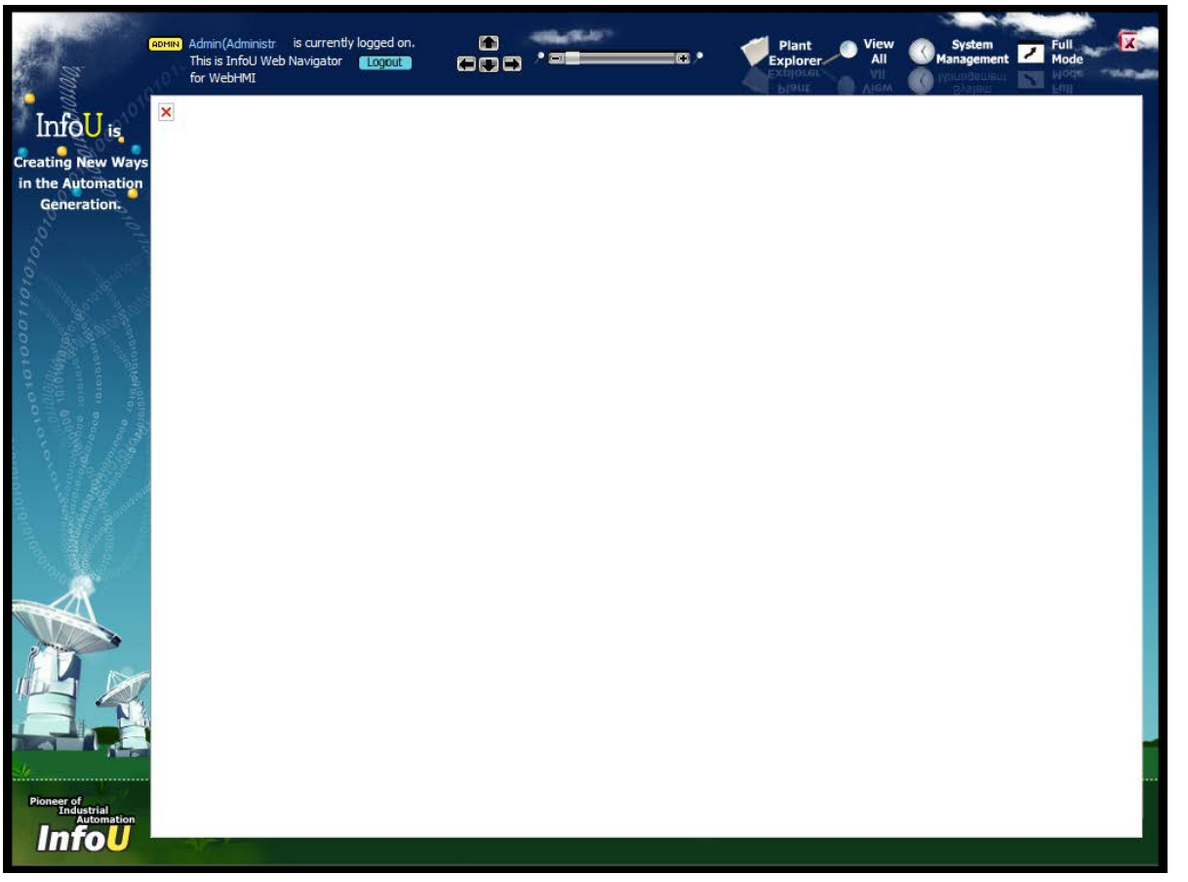


	Name	Desc	Status	Status Change Time
1	iuDataMan	Database Manager	RUN	2015/01/07 14:39:51
2	iuTimeSyncMan	Time Sync Manager	RUN	2015/01/07 14:39:52
3	iuNetworkMan	Network Manager	RUN	2015/01/07 14:39:52
4	iuAlarmArchive	Alarm Archive	RUN	2015/01/07 14:39:53
5	iuAlarmProcess	Alarm Process	RUN	2015/01/07 14:39:53
6	iuAlarmAgent	Alarm Agent	RUN	2015/01/07 14:39:54
7	iuDataProcess	Data Process	RUN	2015/01/07 14:39:55
8	iuIOManager	I/O Manager	RUN	2015/01/07 14:39:55
9	iuControlMan	Control Manager	RUN	2015/01/07 14:39:56
10	iuDataExport	Data Export	RUN	2015/01/07 14:40:17
11	iuScriptRun	Runtime Script	RUN	2015/01/07 14:40:00
12	iuWorkFlow	Runtime Work Flow	RUN	2015/01/07 14:40:00
13	iuScheduler	Scheduler	RUN	2015/01/07 14:40:01
14	iuArchiveServer	Data Archive server	RUN	2015/01/07 14:40:01
15	iuReportGen	Report Generator	RUN	2015/01/07 14:40:02
16	InfoUX	Graphic Viewer	RUN	2015/01/07 14:39:50
17	iuWebHMIServer	Web Server	NOT USE	2015/01/07 14:39:50

(3) Check the tray icon in the lower right corner of Windows to confirm whether the Web HMI Server works properly.



6.1.5 Q: Although I log in to the monitoring screen, there is nothing on the screen. It looks like a blank one.



A: If the server version does not match up with the web system version when you update the file randomly, such a problem may occur. Namely, it means the version of GraRun.dll involved in a web plant screen does not match with the local one so the screen is not loaded. In this case, you need to match dll of C:\InfoU\bin with C:\InfoU\Web Navigator\activeX\InfoUWebClient dll version.

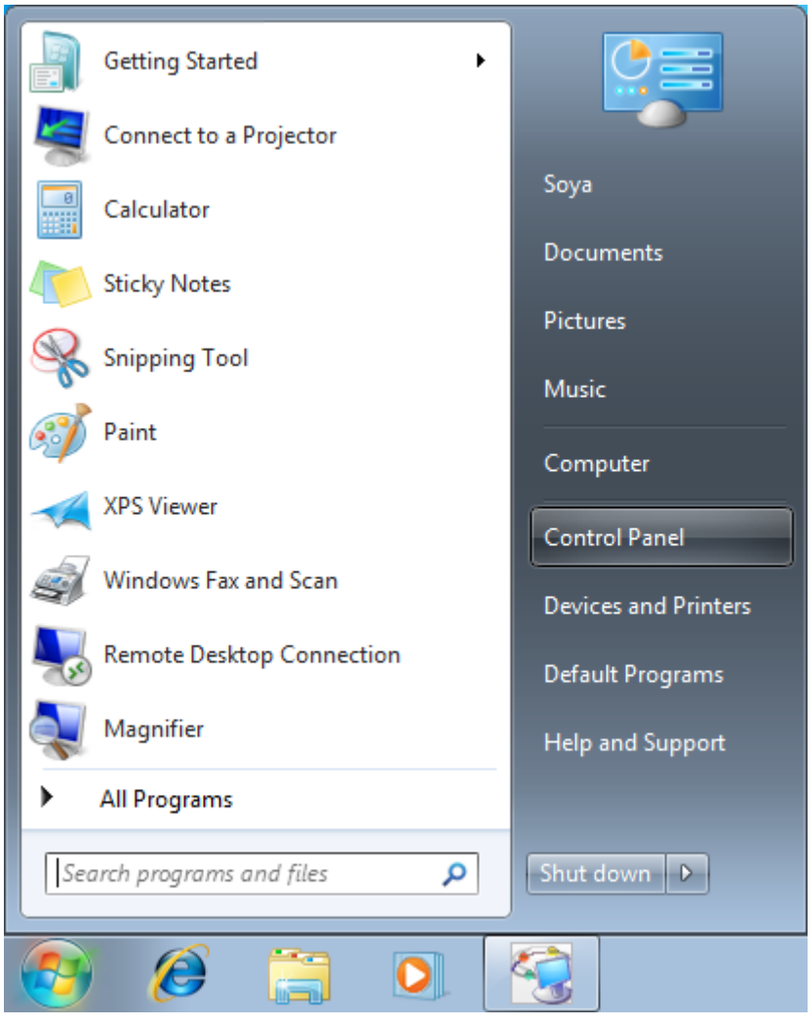
6.2 Q&A for Windows 7

6.2.1 Q: If IIS7 is not automatically installed in Windows 7, how can I install it manually?

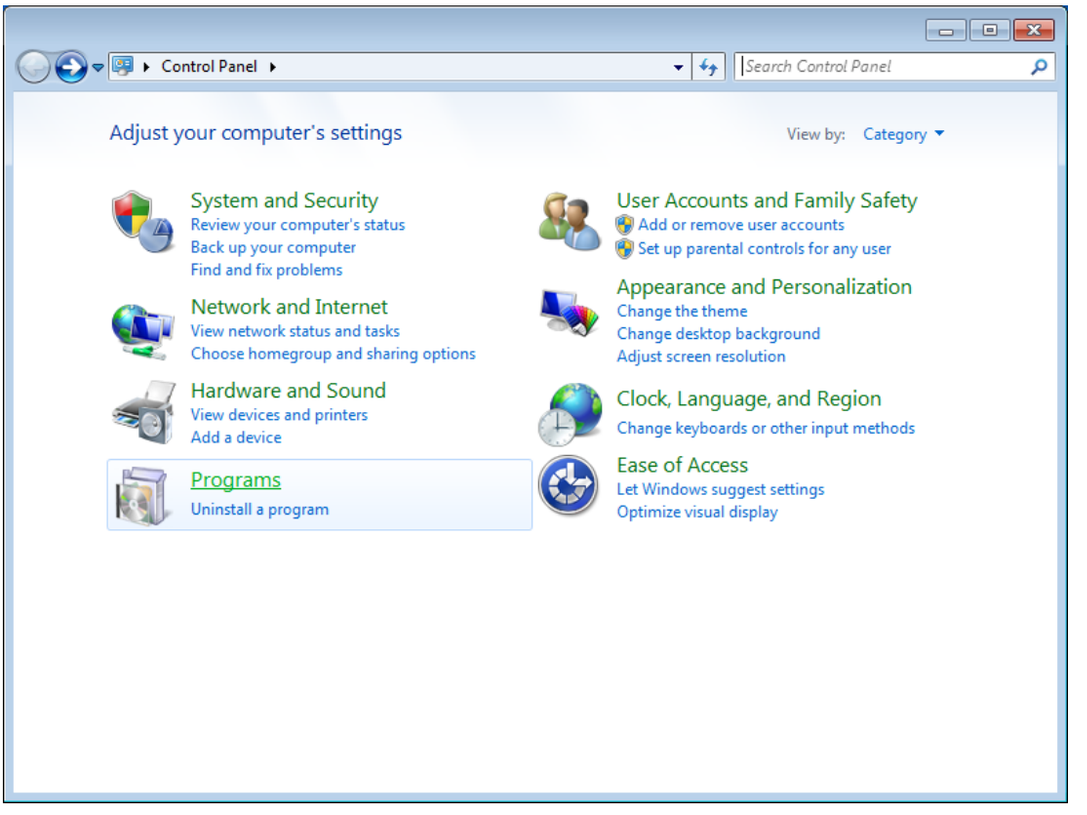
A: In order to use InfoU Web, you can install Windows7 IIS7 based on the following procedures.

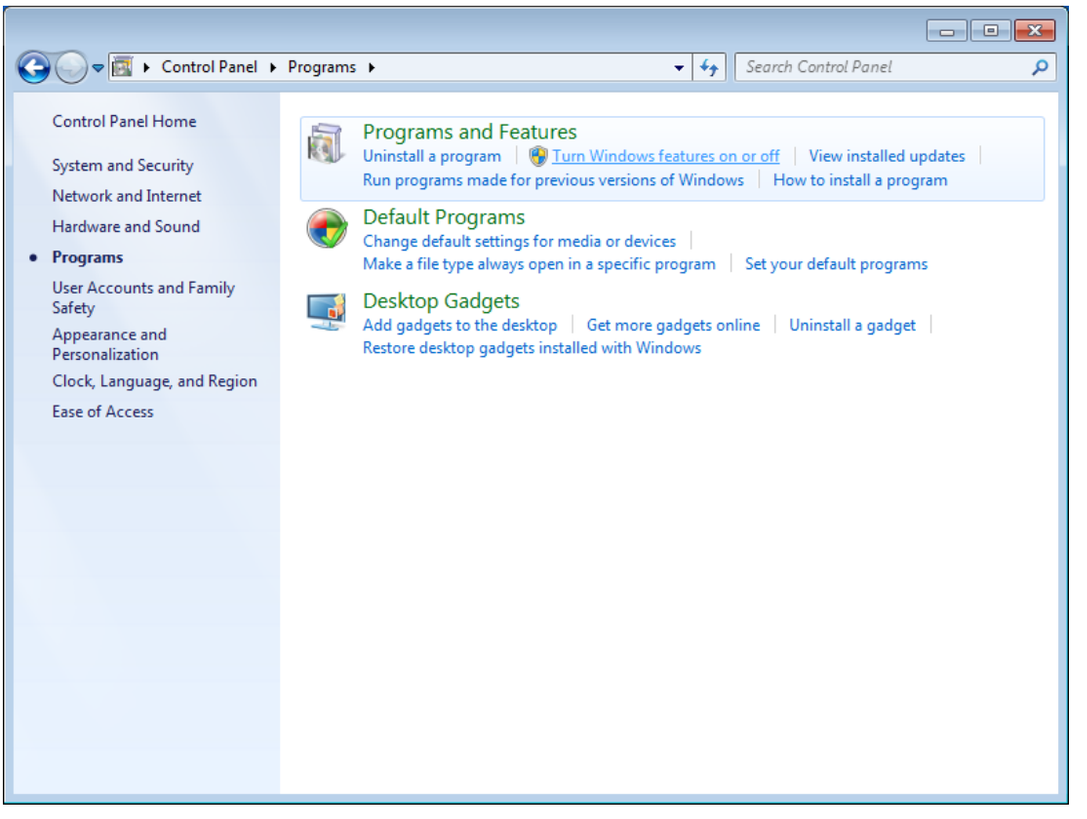
< How to install IIS 7 in Windows 7 >

(1) Click [Start] ➔ [Control Panel]

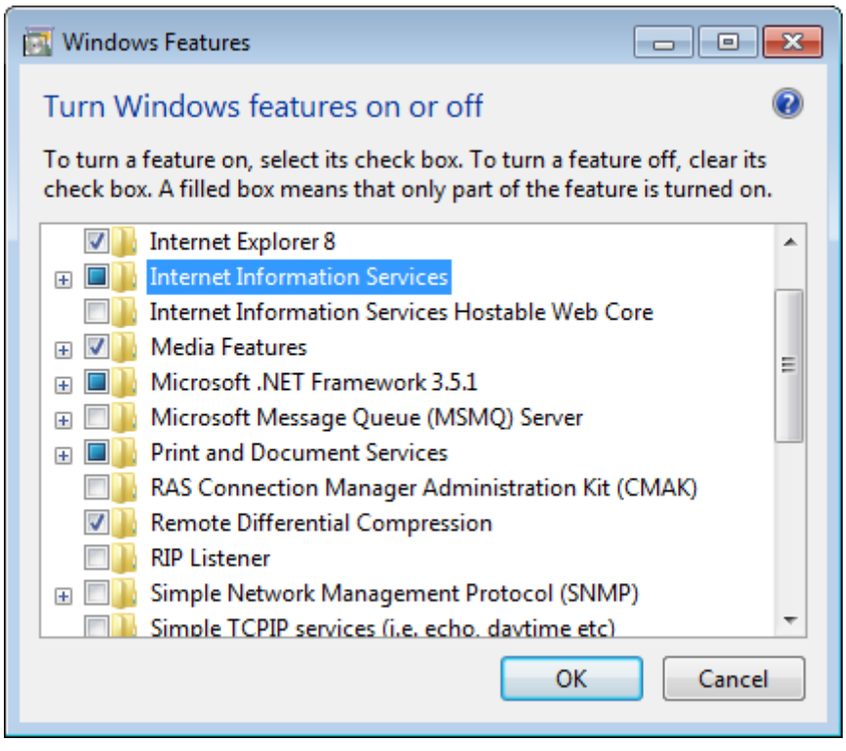


(2) After selecting the [Programs] menu in the Control Panel, click the [Turn Windows features on or off] menu.

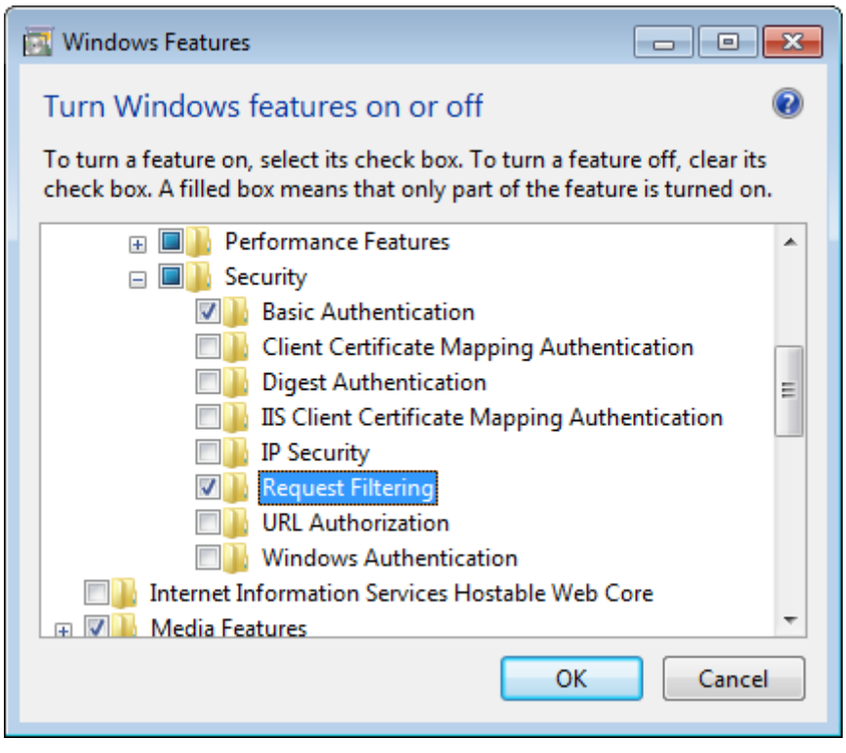




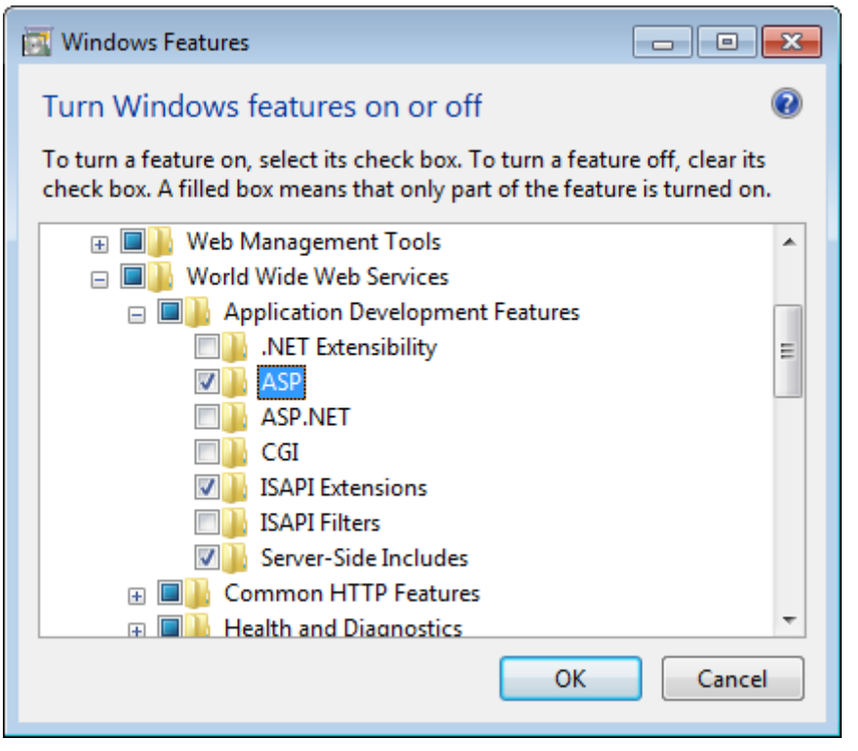
(3) Click the [Internet Information Service] in the dialog box of the Windows Features.



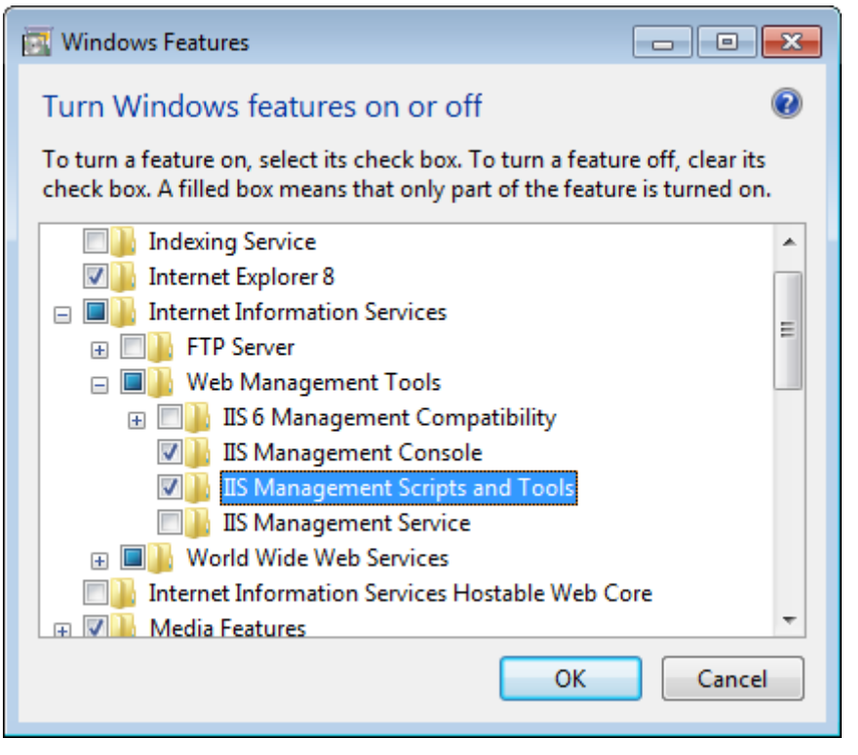
(4) After expanding [Internet Information Service]-[World Wide Web service], put a check the [Basic Authentication] and [Request Filtering] menu in [Security].



(5) In the [Application Development Features], put a check [ASP], [ISAPI Extensions], [Server-Side Includes] menus.



(6) In the [Web Management Tools], put a check [IIS Management Script and Tools].

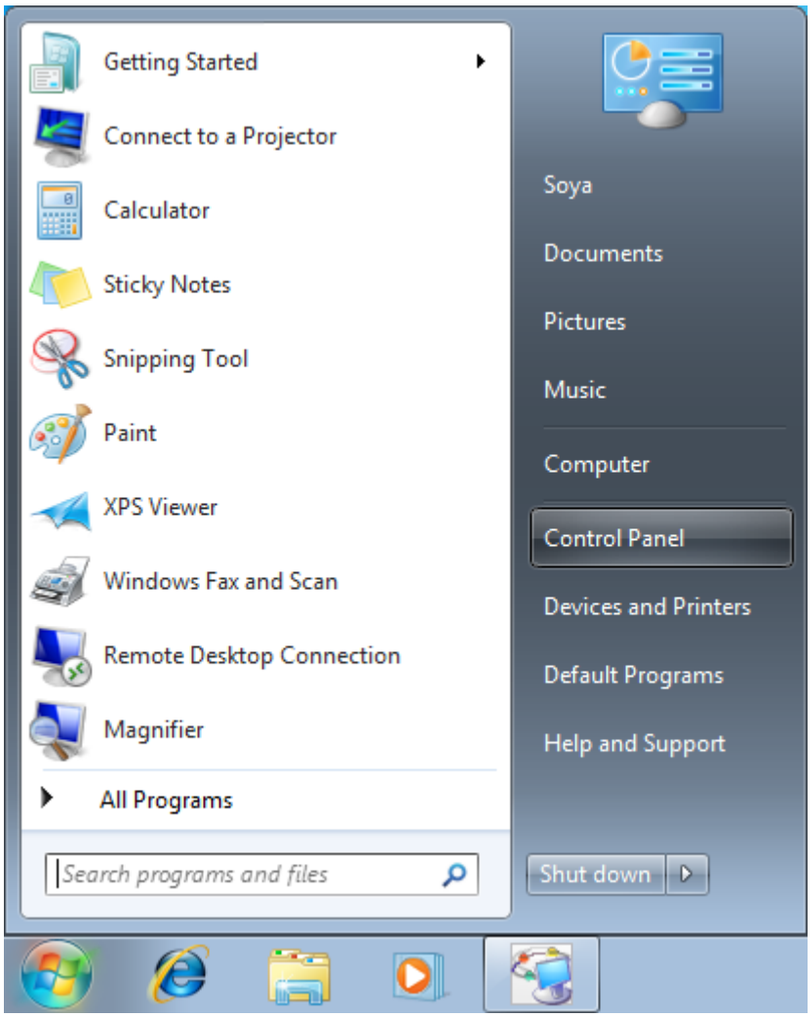


6.2.2 Q: The site seems not to work properly due to wrong settings of the InfoU Web IIS7. How can I configure IIS7 in Windows 7?

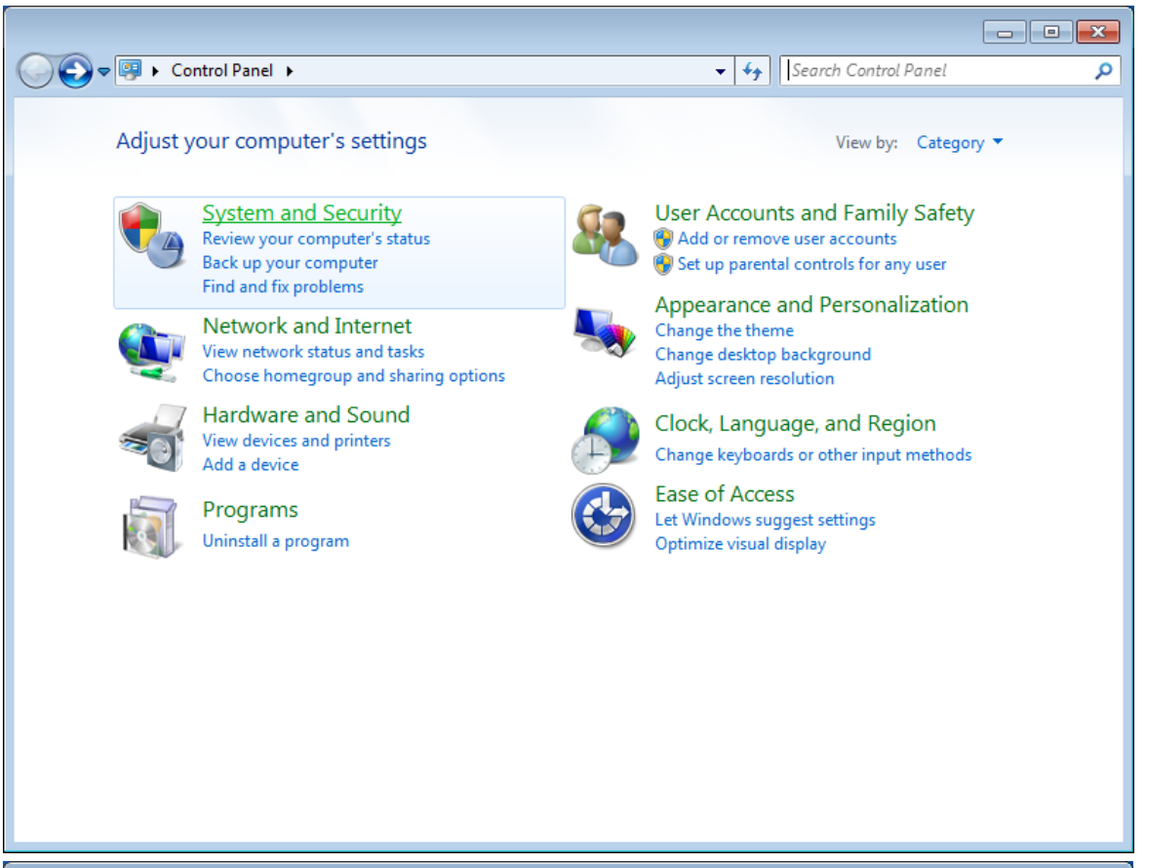
A: In order to use InfoU Web, you can install Windows7 IIS7 based on the following procedures.

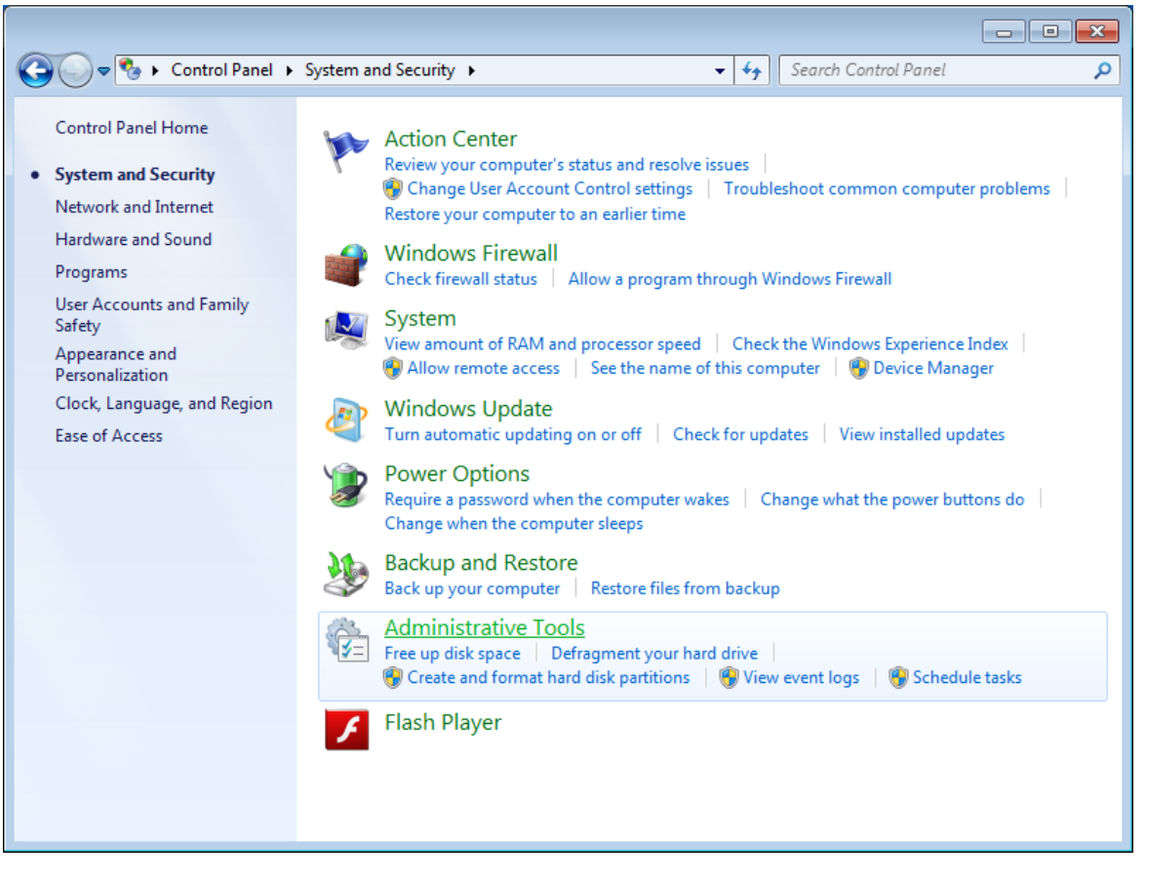
< How to set IIS7 (internet information service) administrative tools in Windows 7>

(1) Click [Start] ➔ [Control Panel].

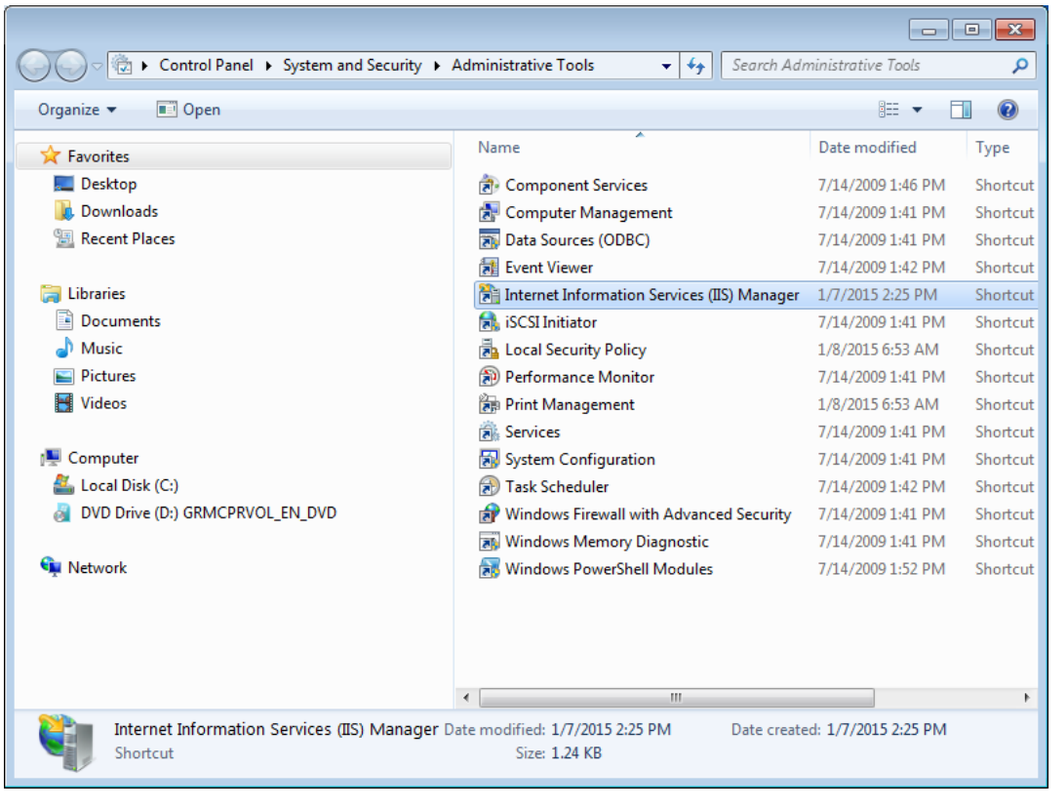


(2) After selecting the [System and Security] menu, choose [Administrative Tools].

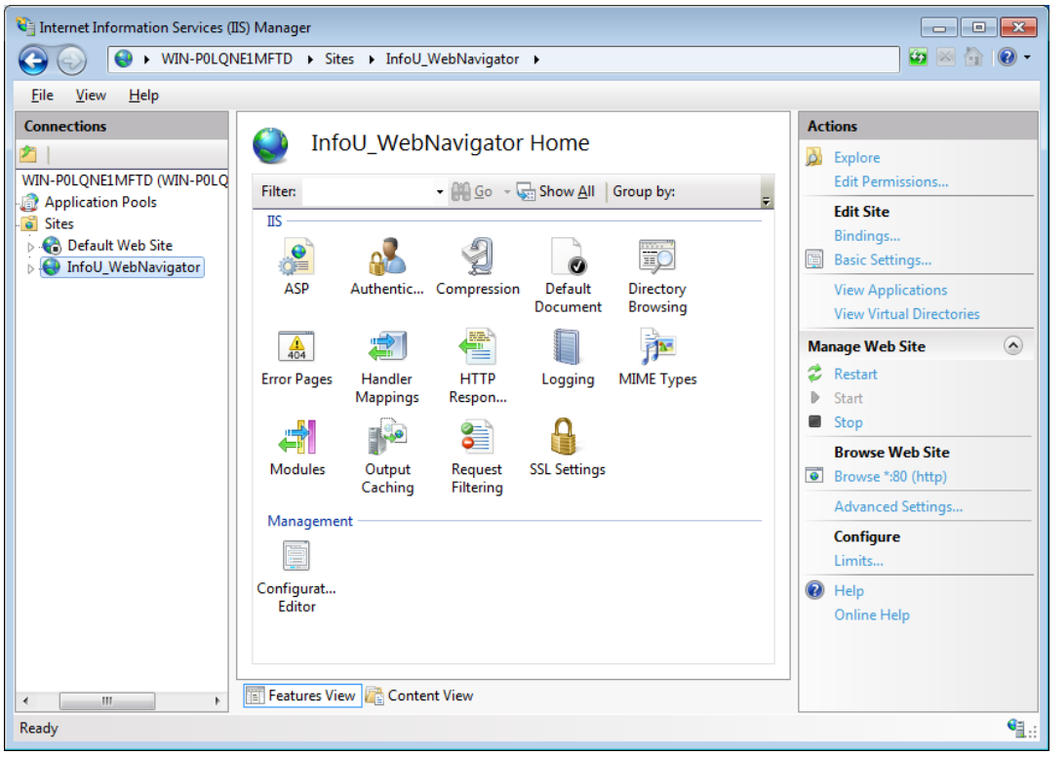




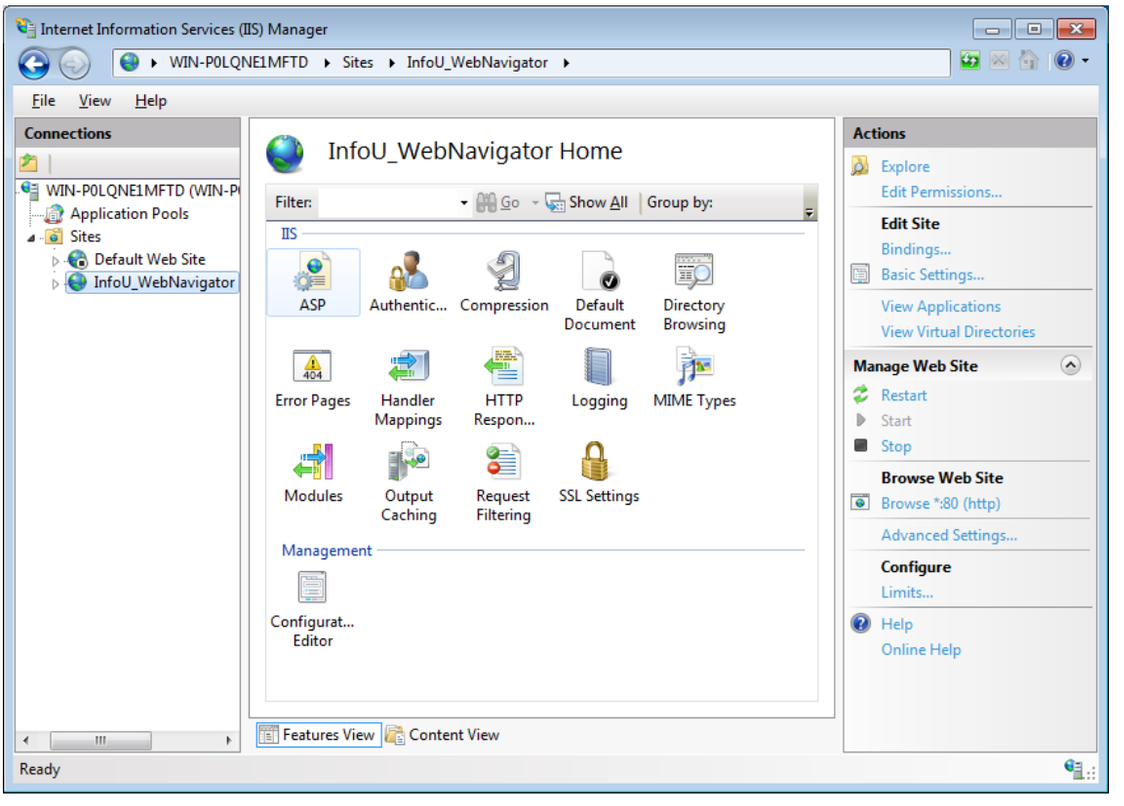
(3) After clicking the [Administrative Tools] menu, click the [Internet Information Service (IIS) Manager].



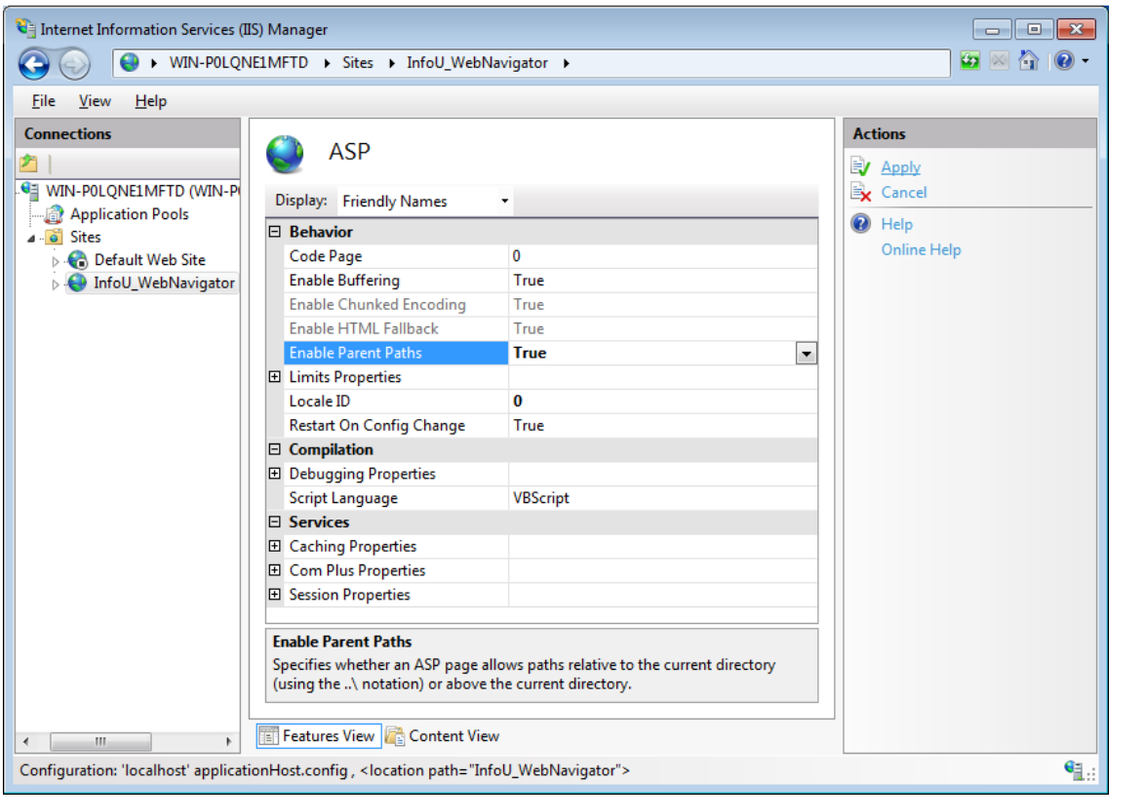
(4) Expand the site in the tree structure on the left side of the screen and click the [InfoU_WebNavigator] site.



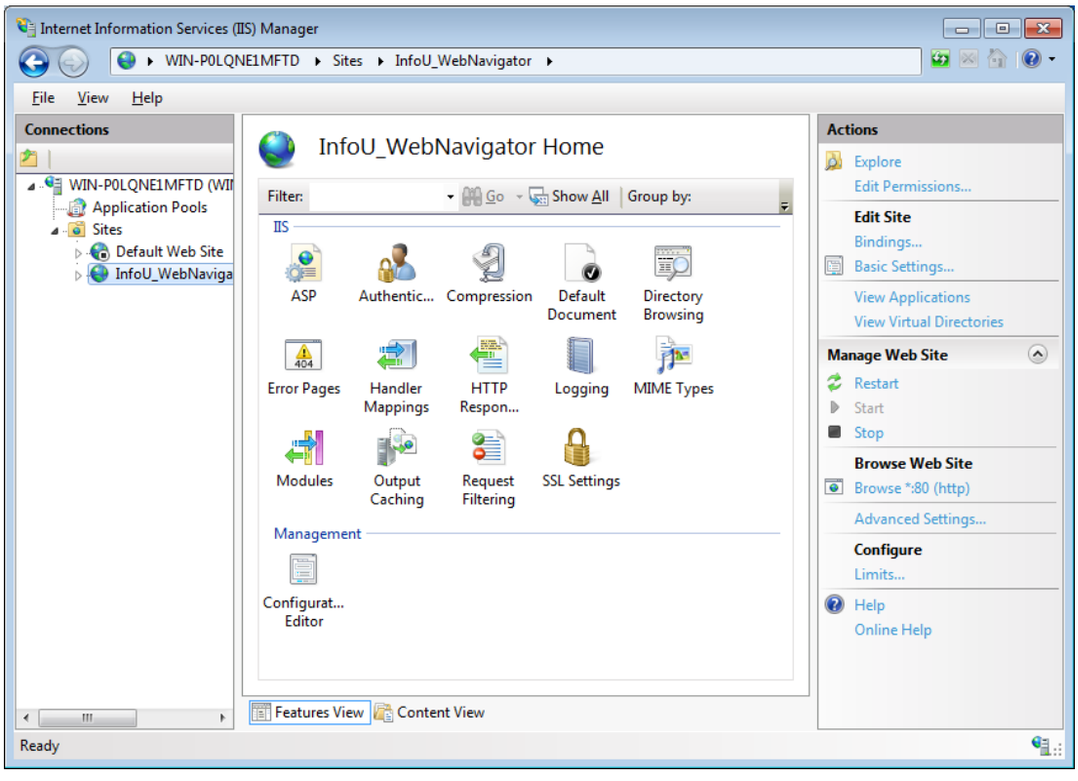
(5) After selecting [IIS] ➔ [ASP] in the [Features View] window, double-click it.



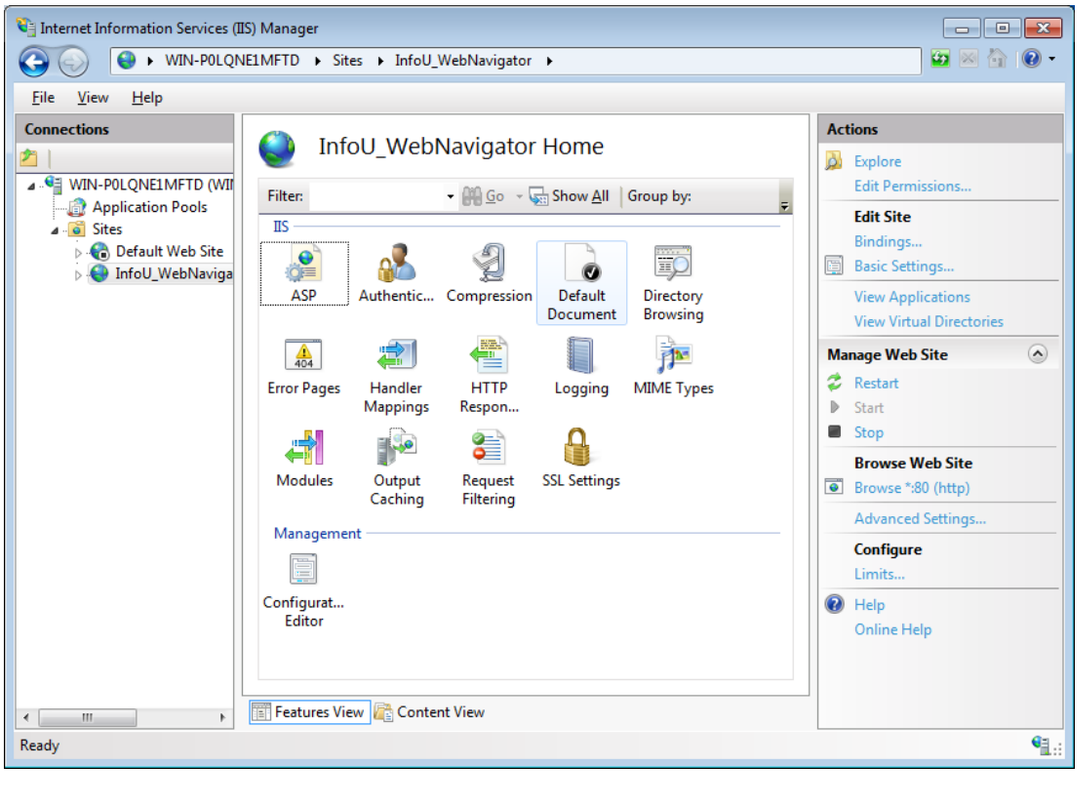
(6) Set [Enable Parent Paths] as True in the ASP screen and click Apply in the Actions.

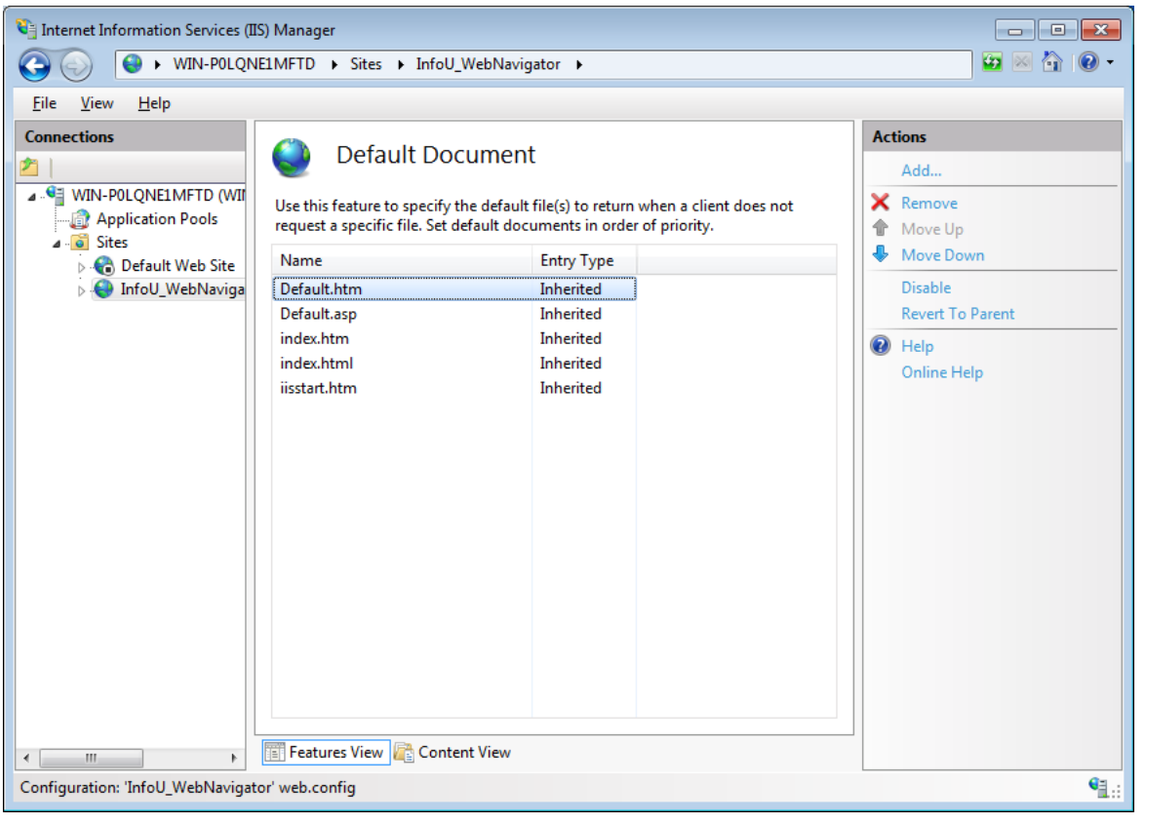


(7) In the tree structure on the left side of the screen, click the [InfoU_WebNavigator] site.

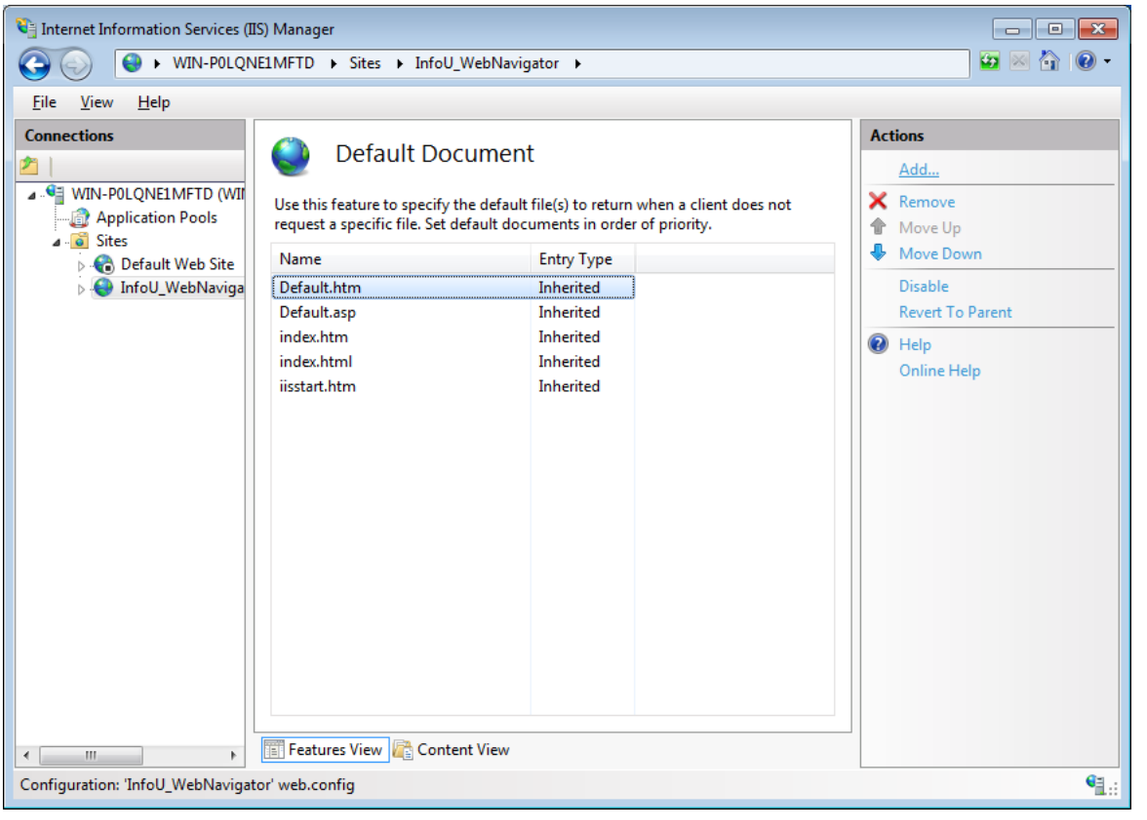


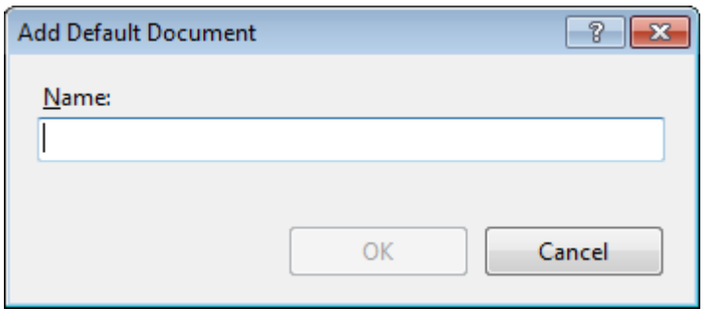
(8) After selecting [IIS] ➔ [Default Document] in the [Features View] screen, double click it.



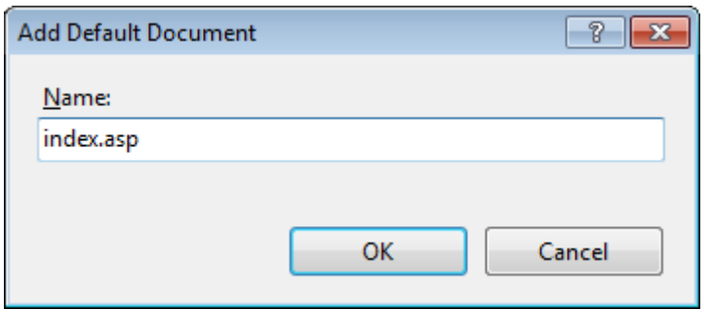


(9) In the [Actions] on the right side of the Default Document page, select the [Add] menu.

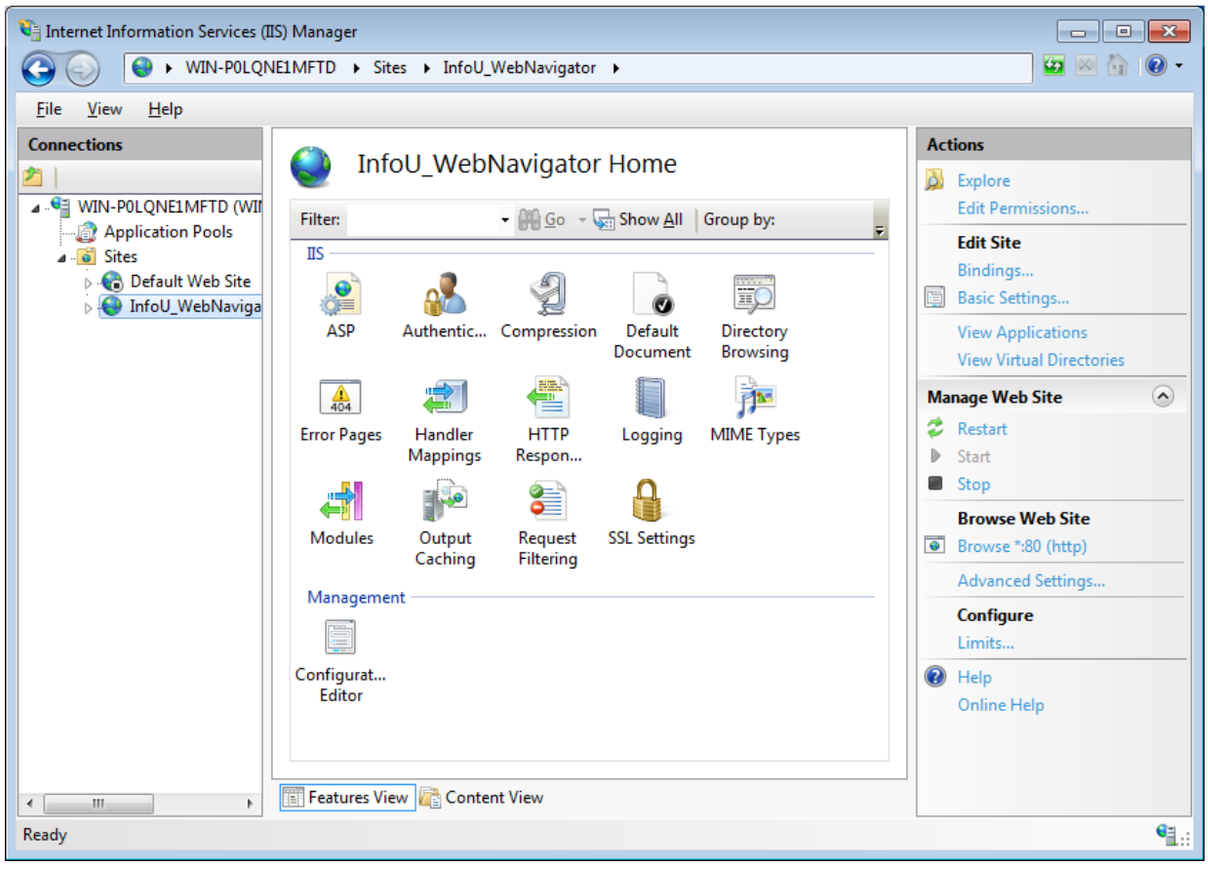




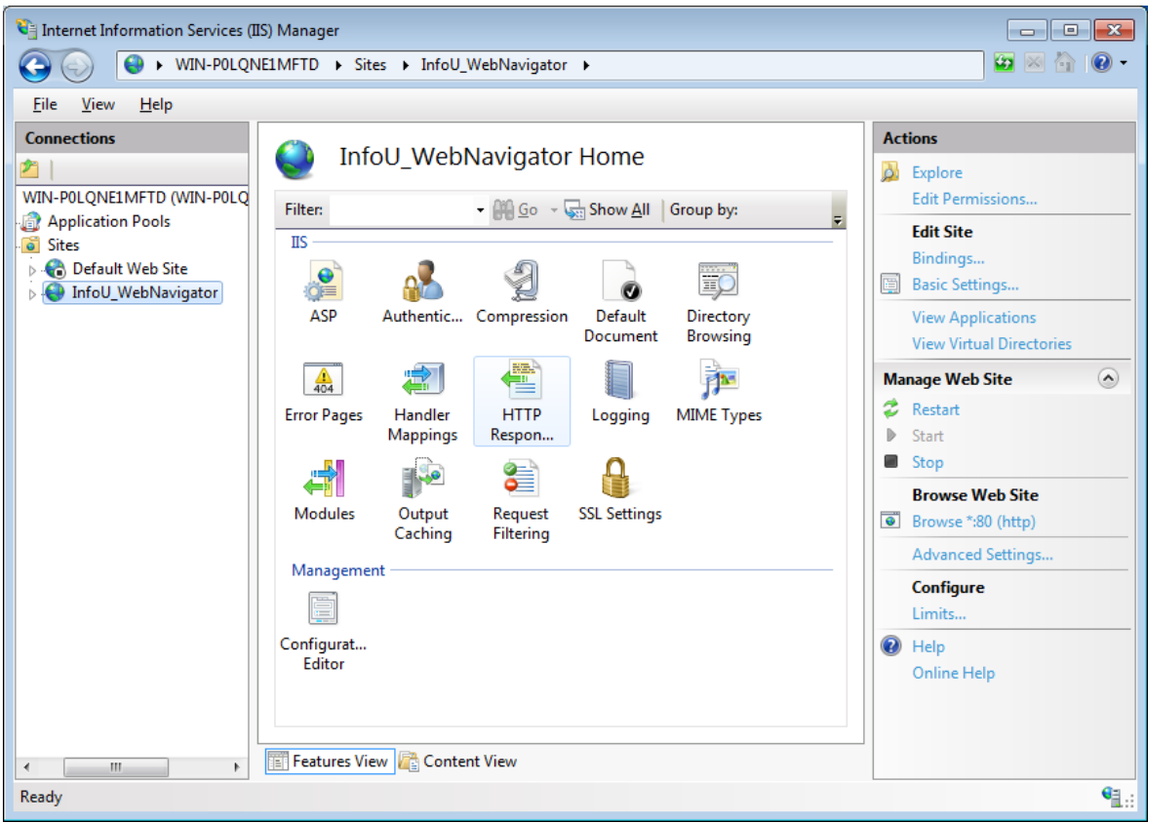
(10) In the dialog screen of Add Default Document, after entering index.asp into the Name column, click the OK button.



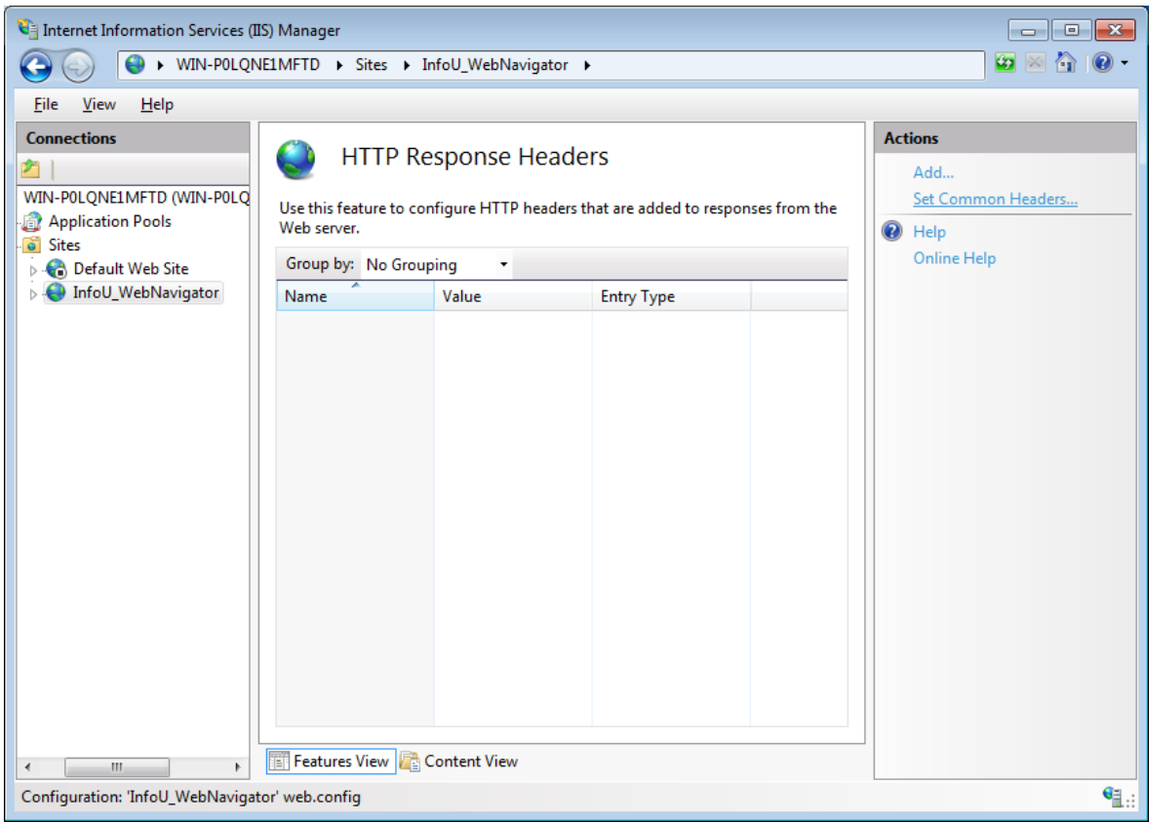
(11) In the tree structure on the left side of the screen, click the [InfoU_WebNavigator] site.



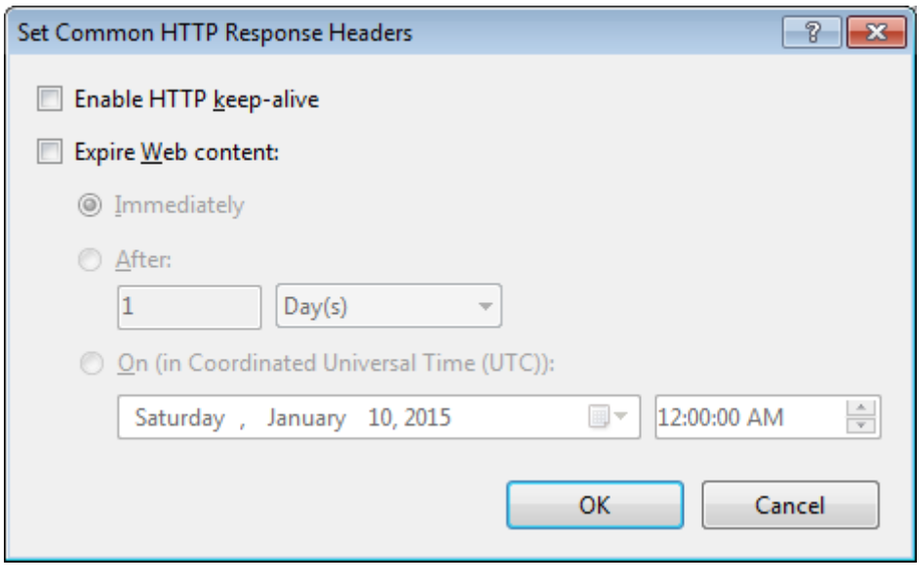
(12) After selecting [IIS] ➔ [HTTP Response Headers] in the [Features View] screen in the center of the screen, double-click it.



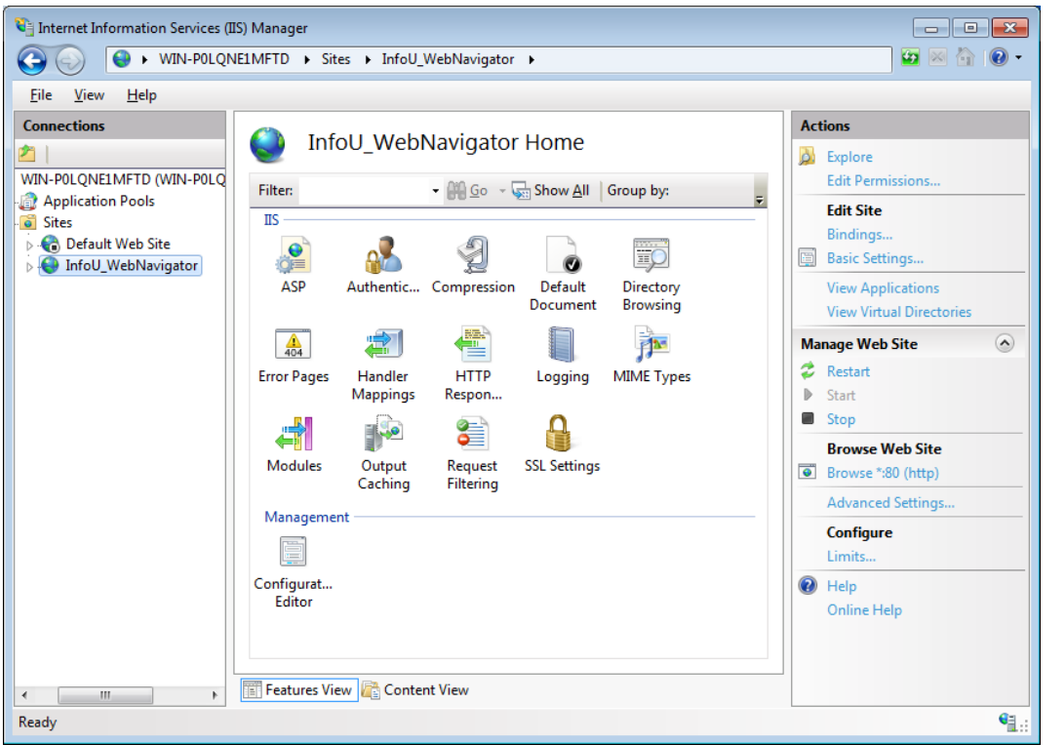
(13) Select the [Set Common Headers] menu in Actions the screen on the right side of the HTTP Response Headers page.



(14) In the dialog box of the Set Common HTTP Response Headers, after unchecking the [Enable HTTP keep-alive] item, click the OK button.

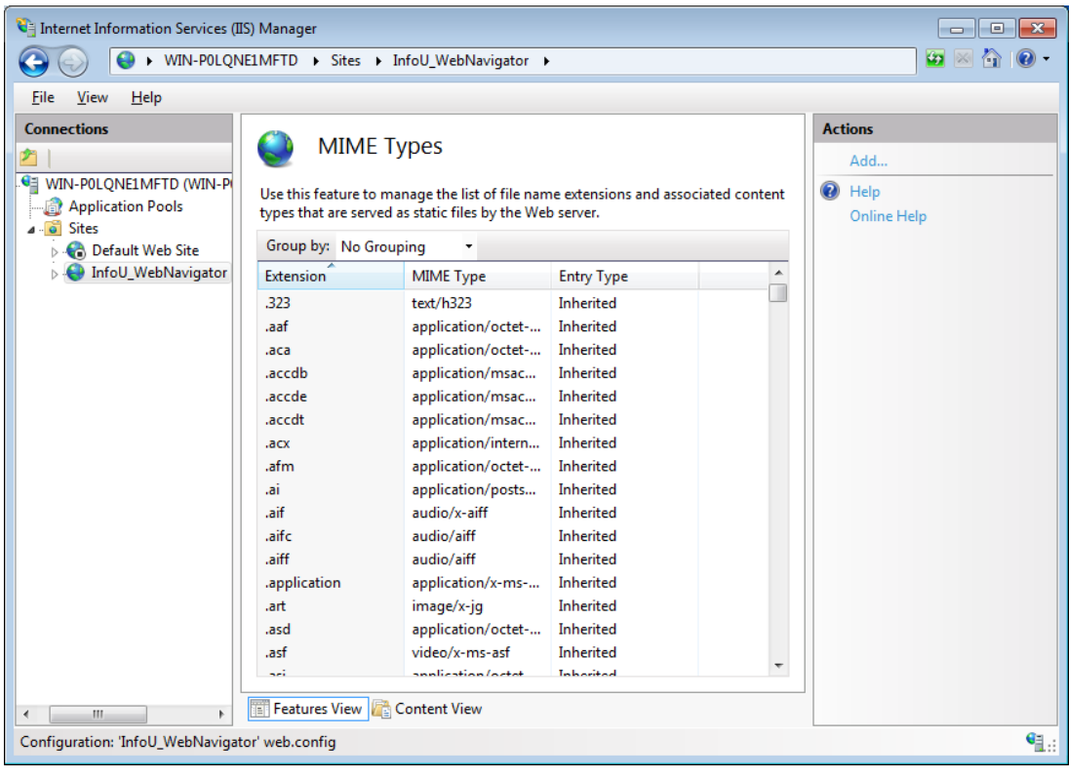


(15) In the tree structure on the left side of the screen, click the [InfoU_WebNavigator] site.

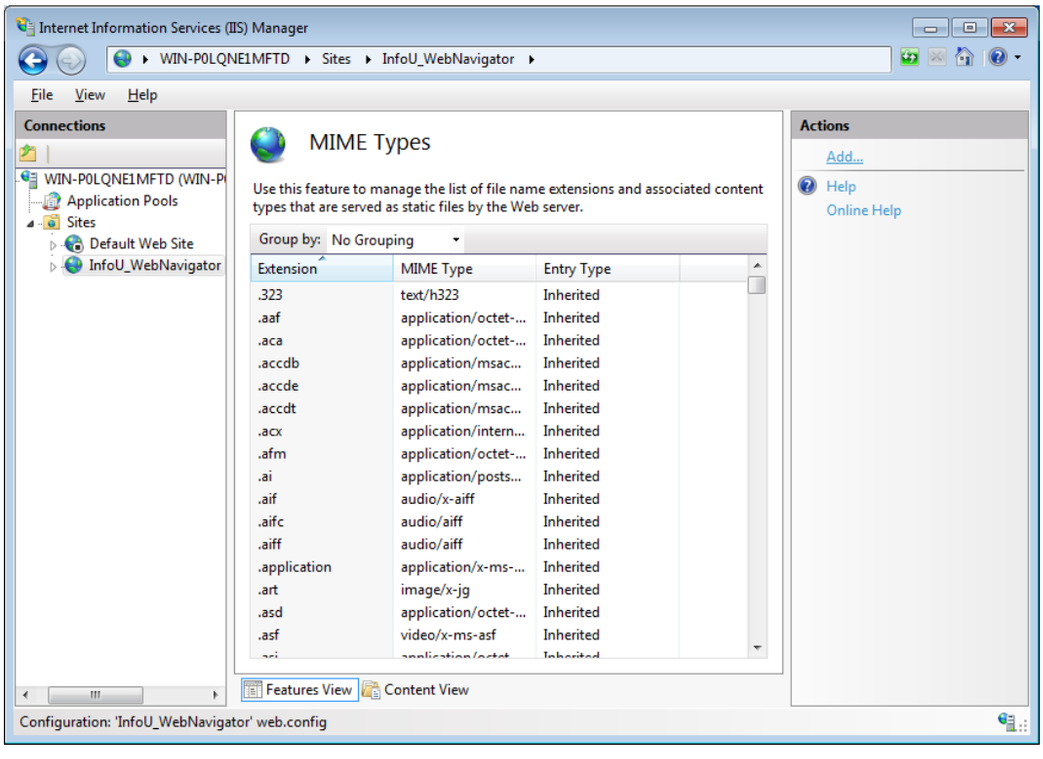


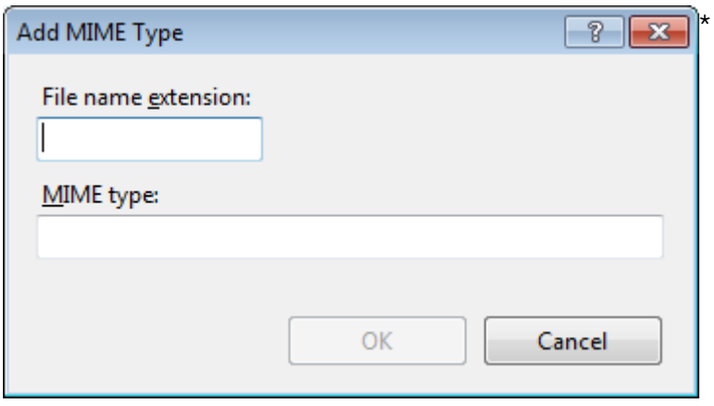
(16) After selecting [IIS] ➔ [MIME Types] in the [Features View] screen in the center of the screen, double-click it.



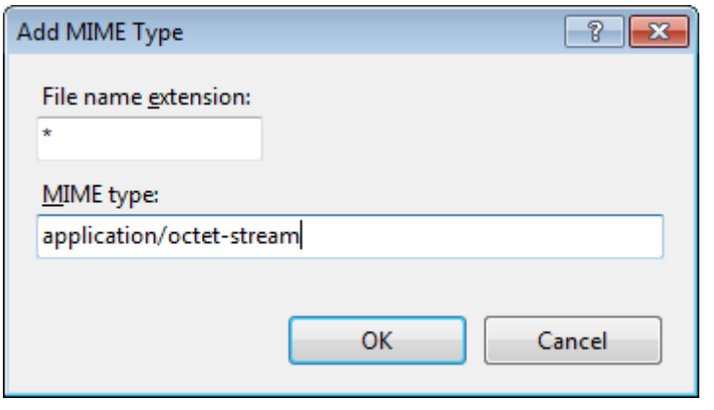


(17) In the [Actions] on the right side of the MIME Types page, select the [Add] menu.

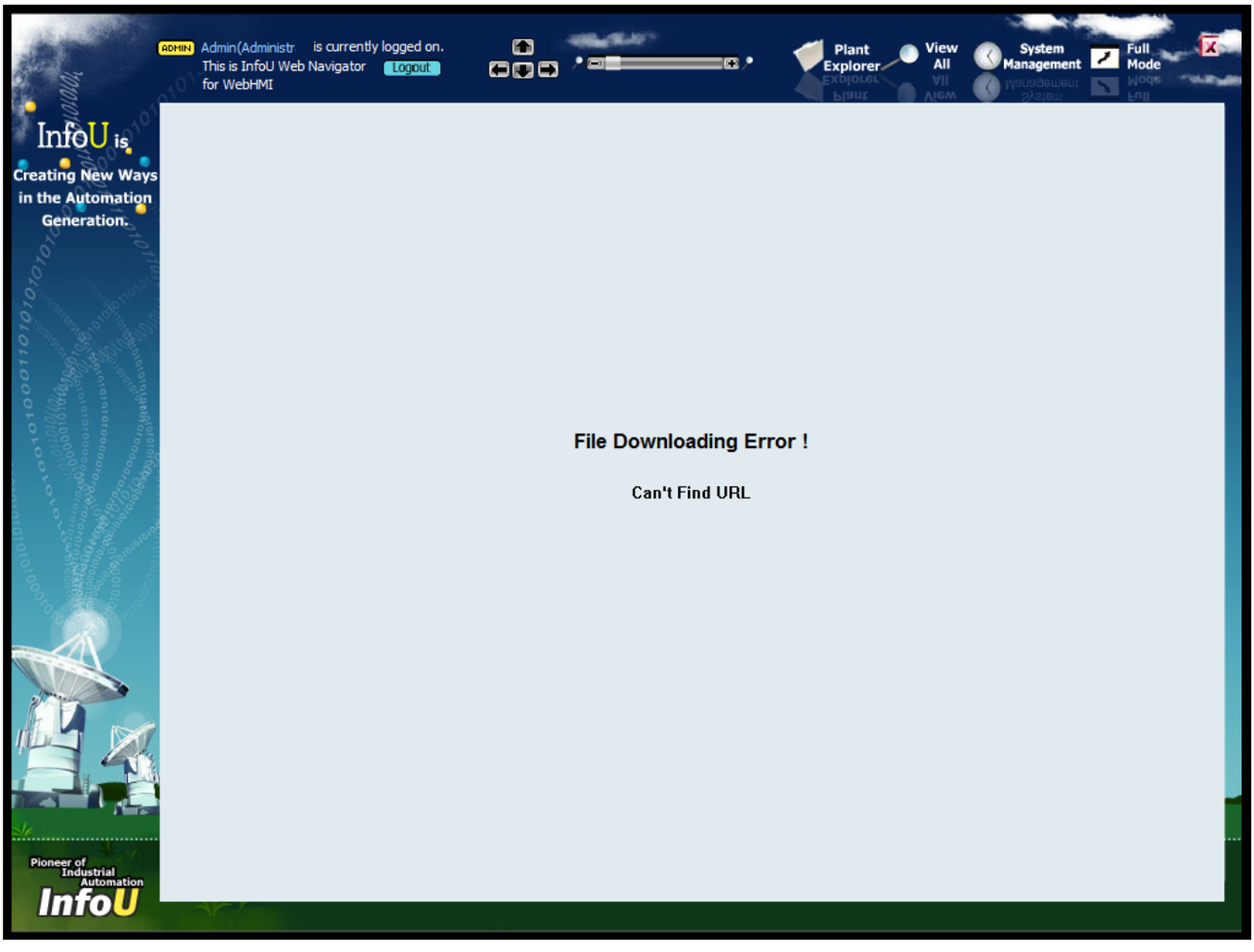




(18) In the dialog box of [Add MIME Type], enter * into the [File name extension] column and enter application/octet-stream into the [MIME type] column. Then, click the OK button.

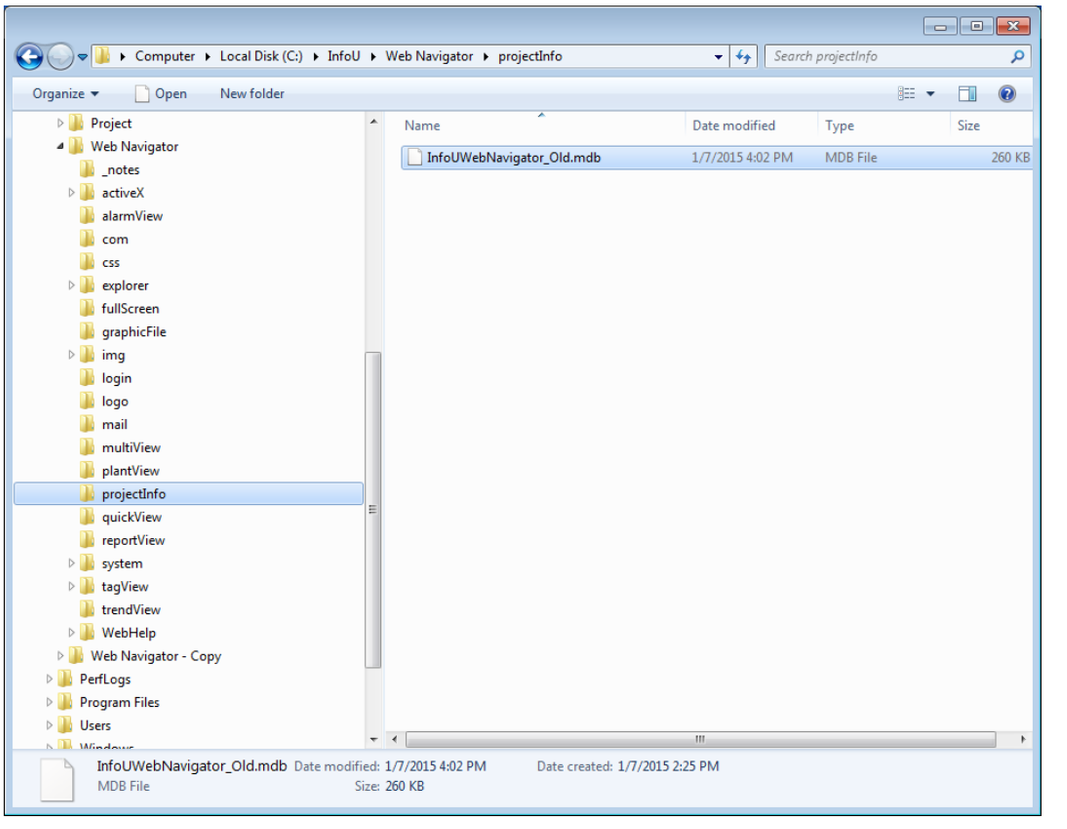


6.2.3 Q: After deleting the file by mistake during version update, the web site does not work and in addition, the monitoring screen is not displayed as below. How can I check on normal operation?

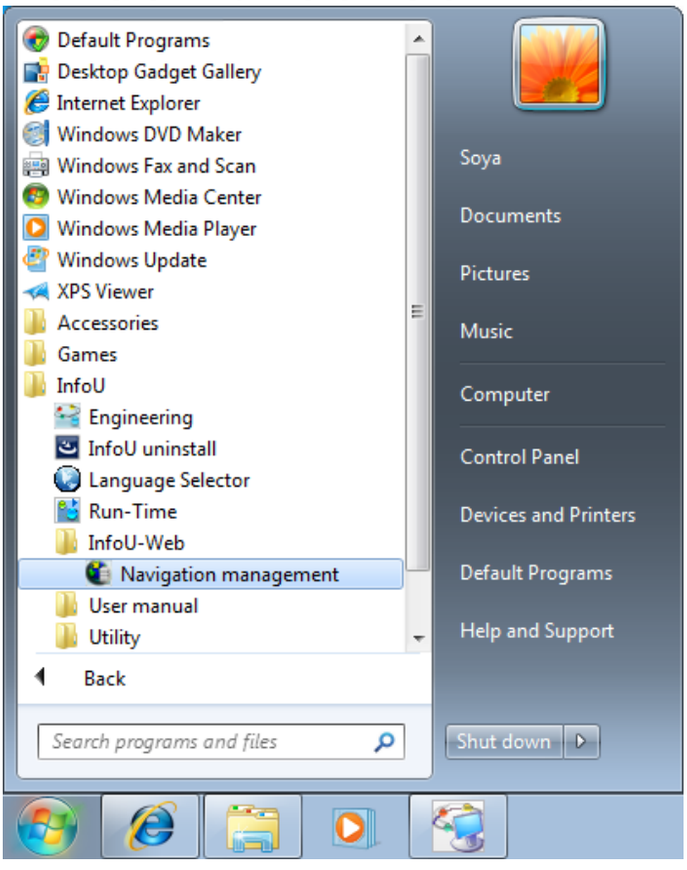


A: It seems the project file is damaged during update. After creating the project file newly and check whether the web site works normally.

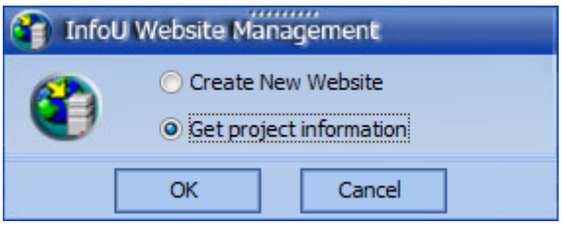
(1) C:\InfoU\Web Navigator\projectInfo\InfoUWebNavigator.mdb ← Change the name of the project information file into InfoUWebNavigator_Old.mdb.



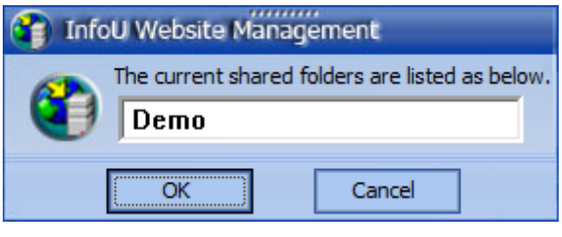
(2) Run [Start]➔[All Programs]➔[InfoU]➔[InfoU-Web]➔[Navigation Management].



(3) After selecting [Get project information], click the [OK] button.

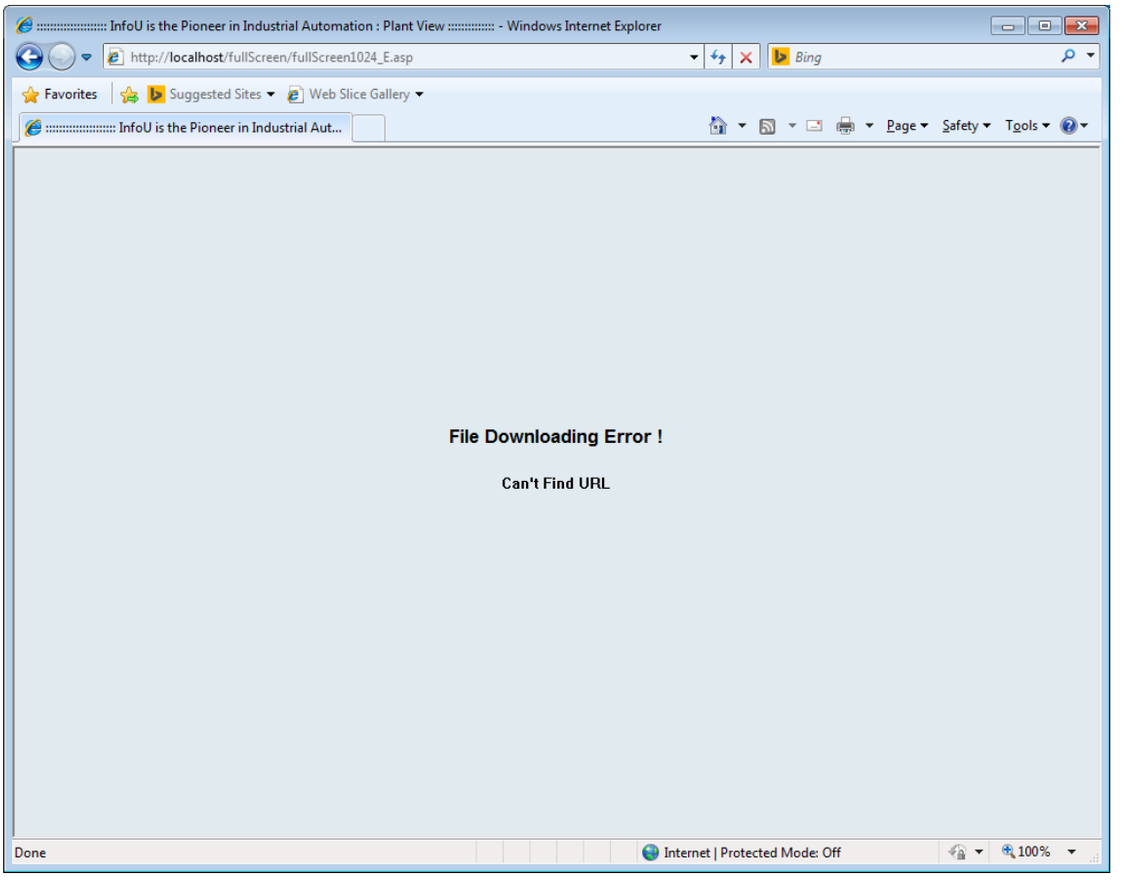


(4) To get the project information, click the [OK] button.



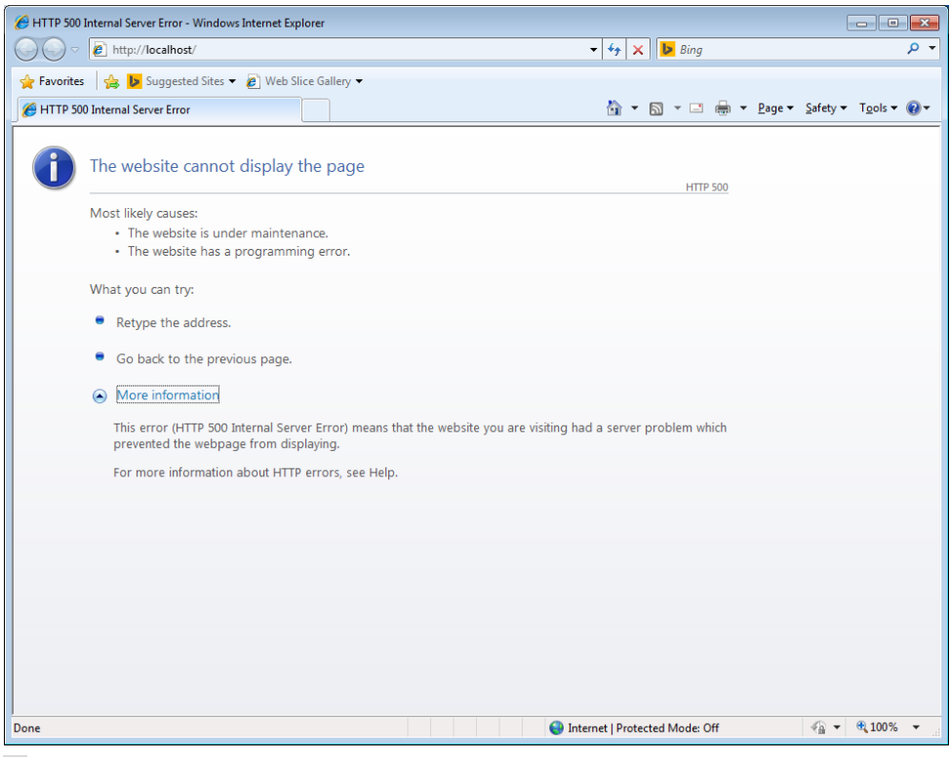
After getting the project information again as above, you can try to access to the plant screen.

6.2.4 Q: After logging in to the site, the monitoring screen is not displayed with the message 'Can't Find URL'



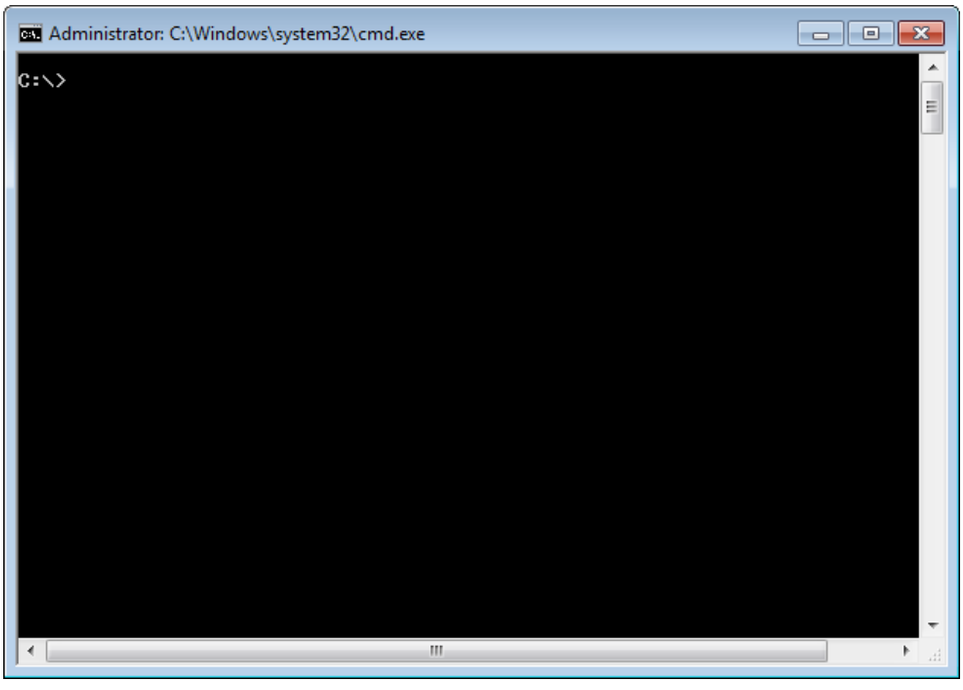
A: As you see from the error message, please check whether the file exists in the web site path. This problem may occur when a user deletes the file or change the name so check whether the file properly exists. In terms of the web site folder, you need to check if the plant file exists in the Web Navigator\graphicFile folder located in the drive where the InfoU is installed. If the InfoU system is installed in C:\ drive, the C:\InfoU\Web Navigator\graphicFile\system status.ivd file must exist.

6.2.5 Q: If I access to the InfoU Web site, the error message is displayed instead of the login screen as below.

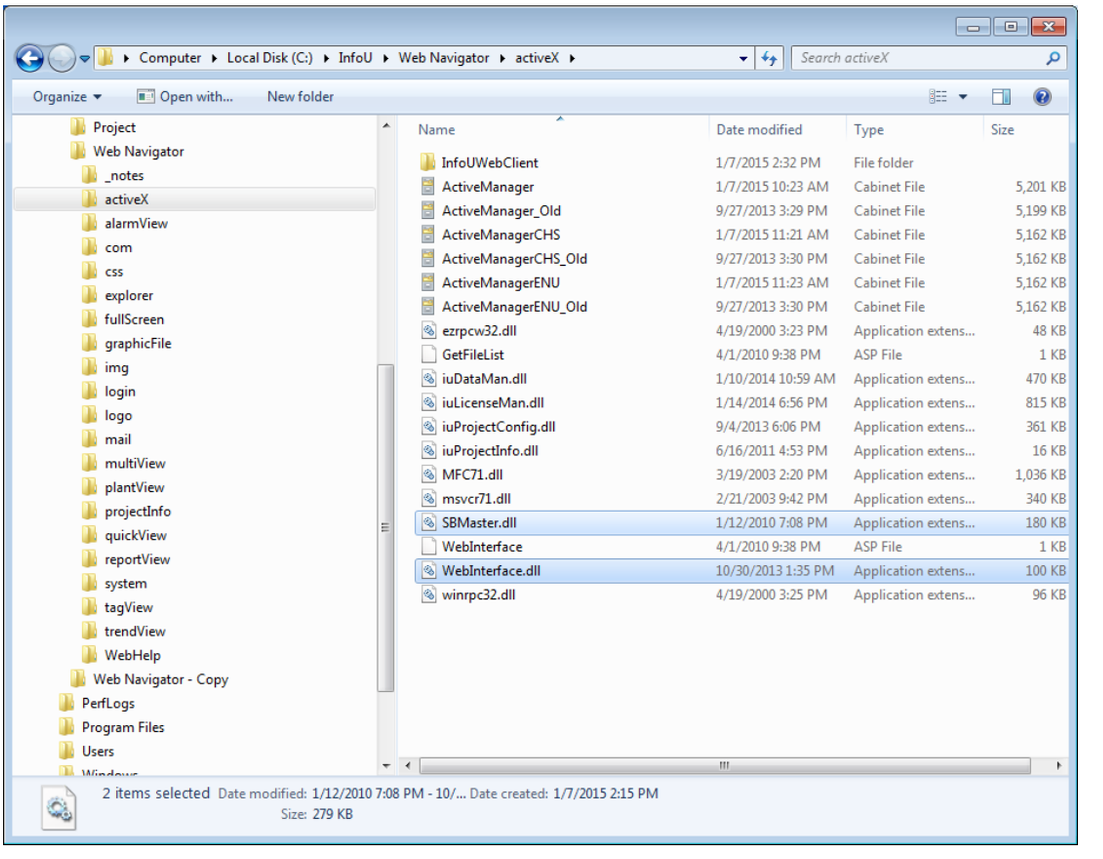


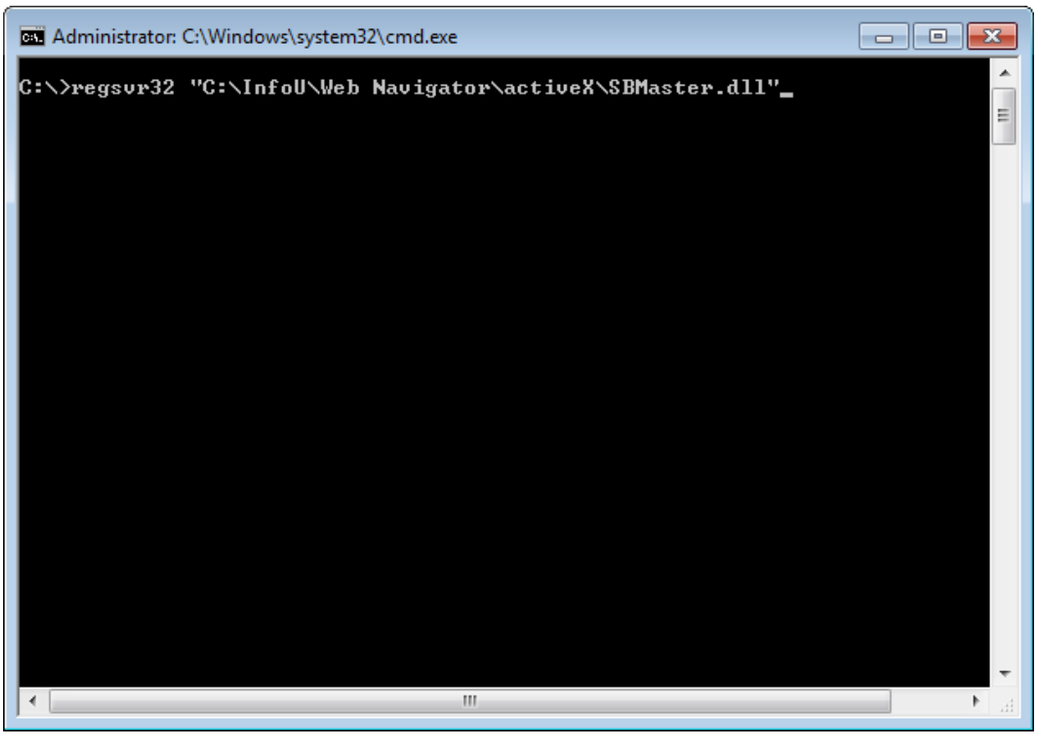
A: Such a screen may be displayed when InfoU Web is not installed normally. In case DLL registration is not properly done, the screen is not displayed. In this case, register DLL manually as shown below.

(1) Run [Start]➔[Run]➔ and enter the cmd(command).

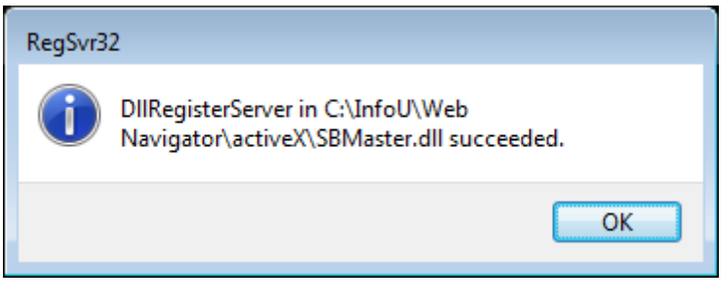


(2) Enter regsvr32 SBMaster.dll in the command widow or drag it from the below folder. After getting the message showing registration is done, reenter regsvr32 WebInterface.dll in the command window or drag it from the below folder.

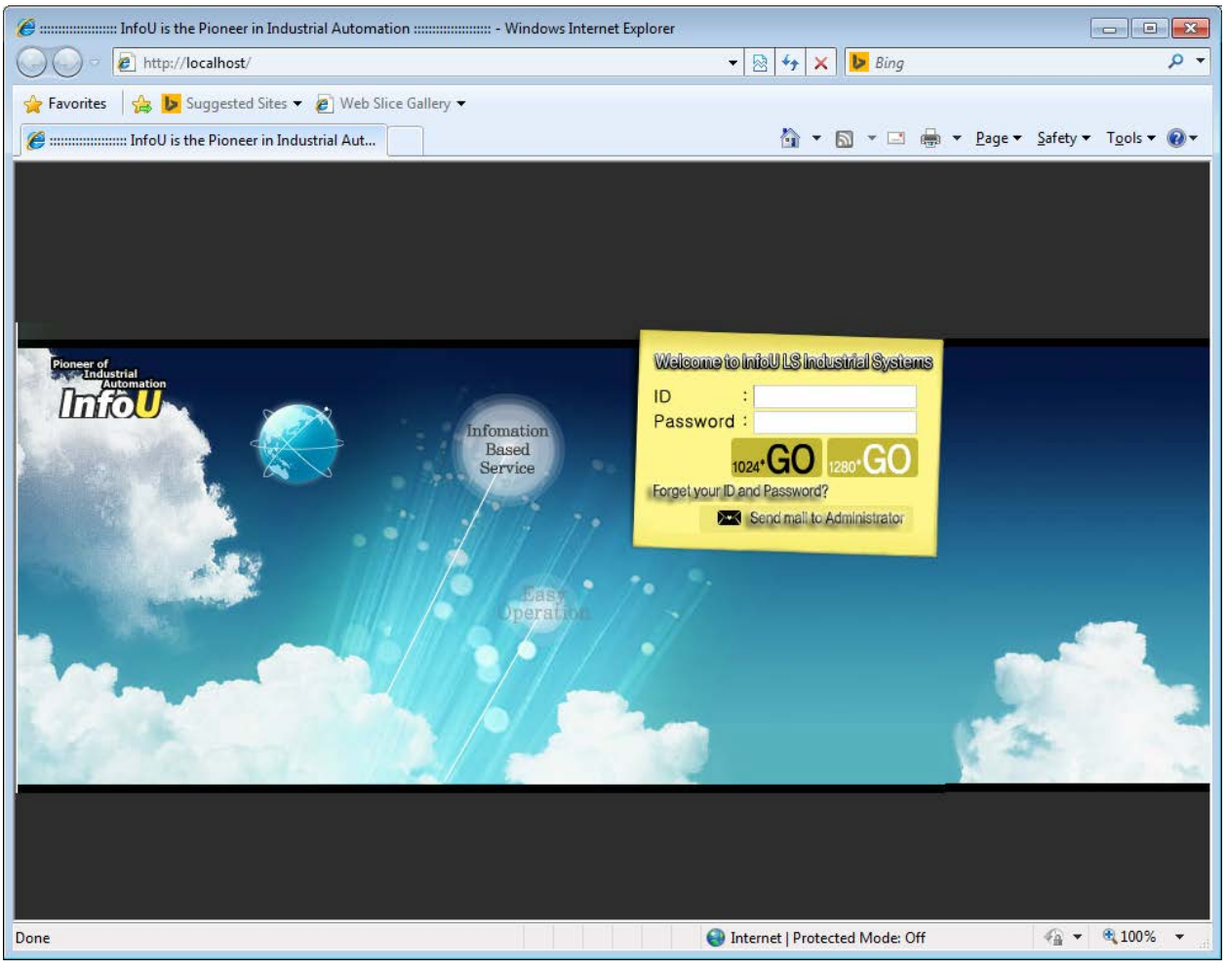




(3) If you get the message showing registration is done, it means DLL registration is normally completed.



(4) Confirm it by accessing to the InfoU Web site.





LS values every single customer.

Quality and service come first at LSIS.

Always at your service, standing for our customers.

www.lsis.com

10310001053

■ **HEAD OFFICE**
LS Tower, 127, LS-ro, Dongan-gu, Anyang-si, Gyeonggi-Do, 431-848, Korea
http://www.lsis.com
■Southeast Asia +82-2-2034-4888 cshwang@lsis.com (Charles Hwang)
■Europe +82-2-2034-4676 sukyong@lsis.com (Brian Choi)
■Turkey/Israel/CIS +82-2-2034-4879 dkimc@lsis.com (Daniel Kim)
■Oceania +82-2-2034-4394 kacho@lsis.com (Kendra Cho)
■North/Latin America +82-2-2034-4286 hkchung@lsis.com (Hank Raul Chung)
■Southwest Asia/Africa +82-2-2034-4467 myleed@lsis.com (Henry Lee)
■Middle East +971-4-886-5360 khchoi1@lsis.com (Lambert Choi)

■ **LSIS(Dalian) Co., Ltd. /CHINA**
No. 15, Liaohexi 3-Road, Economic and Technical Development zone,
Dalian, P.R. China
Tel : 86-411-8731-7542 Fax : 86-411-8730-7560 E-Mail : dskim@lsis.com

■ **LSIS(Wuxi) Co., Ltd./CHINA**
102-A, National High & New Tech Industrial Development Area,
Wuxi, Jiangsu, P.R. China
Tel : 86-510-8534-6666 Fax : 86-510-8534-4078 E-Mail : sojin@lsis.com

■ **LS Hukai Electric(Hubei) Co., Ltd./CHINA**
No. 100, Tanjiahe Road, Dianjun District, Yichang City, Hubei Province, P.R. China
Tel : 86-717-667-7536 Fax : 86-717-667-7222 E-Mail : jaewoongh@lsis.com

■ **LS-VINA Industrial Systems Co., Ltd./VIETNAM**
Room 1311, 13th, M3-M4 Building 91 Nguyen Chi Thanh street, Hanoi, Vietnam
Tel : 84-4-6275-8055 Fax : 86-21-5237-7189

■ **LSIS(ME) FZE/U.A.E.**
LOB 19-205, JAFZA View Tower, Jebel Ali Free Zone, Dubai, United Arab Emirates
Tel : 971-4-886-5360 Fax : 971-4-886-5361 E-Mail : shunlee@lsis.com

■ **LSIS Europe B.V./NETHERLANDS**
1st. Floor, Tupolevlaan 48, 1119NZ,Schiphol-Rijk, The Netherlands
Tel : 31-20-654-1420 Fax : 31-20-654-1429 E-Mail : htha@lsis.com

■ **LSIS Japan Co., Ltd./JAPAN**
16th,Higashi-Kan, Akasaka Twin Tower, 2-17-22, Akasaka, Minato-ku, Tokyo, Japan
Tel : 847-941-8240 Fax : 847-941-8259 E-Mail : ybleeb@lsis.com

■ **LSIS USA Inc./U.S.A.**
2000 Millbrook Drive, Lincolnshire, Chicago, IL 60069, United States of America
Tel : 86-510-8534-6666/Fax : 86-510-8534-4078 e-mail : caidx@lsis.com.cn

■ **LSIS Gurgaon Office/INDIA**
109 First Floor, Park Central, Sector-30, Gurgaon- 122 002, Haryana, India
Tel : +0091-124-493-0070 Fax : 91-1244-930-066 E-Mail : hwyim@lsis.com

※ LSIS constantly endeavors to improve its product so that
Information in this manual is subject to change without notice.

InfoU / 2015. 2

ⓒ LSIS Co., Ltd 2015 All Rights Reserved.