How to Configure KingSCADA with MGate 5105

Moxa Technical Support Team
support@moxa.com

Contents

 Released on December 24, 2014

About Moxa

Moxa is a leading manufacturer of industrial networking, computing, and automation solutions. With over 25 years of industry experience, Moxa has connected more than 30 million devices worldwide and has a distribution and service network that reaches customers in more than 70 countries. Moxa delivers lasting business value by empowering industry with reliable networks and sincere service for automation systems. Information about Moxa's solutions is available at www.moxa.com. You may also contact Moxa by email at info@moxa.com.

How to Contact Moxa

Tel: +886-2-8919-1230
Fax: +886-2-8919-1231

MOXA®
Reliable Networks ▲ Sincere Service

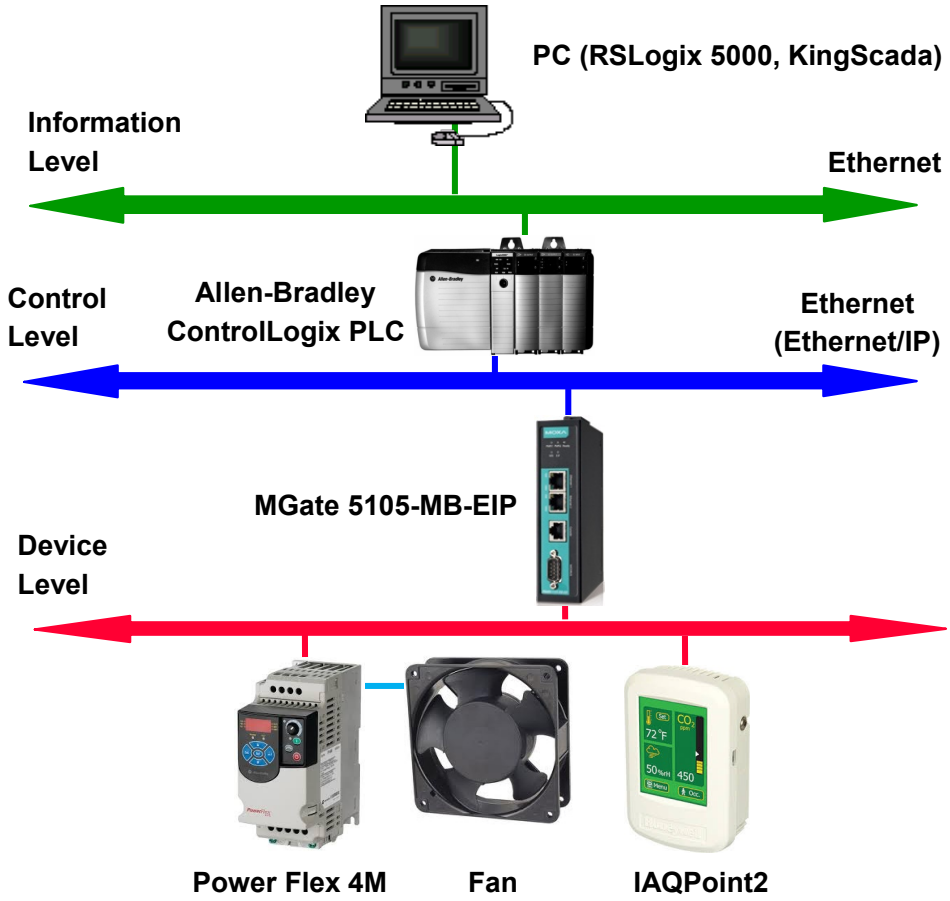
1 Application Description

This application shows how to set up a **KingSCADA** system to control and monitor **Allen-Bradley PLC** tags which are transmitted from **MGate 5105**.

For more information about Allen-Bradley PLC communication with MGate 5105, refer to the documents *Configuring Allen-Brandly ControlLogix PLC with Moxa MGate 5105-MB-EIP* and *How to Configure Pro-face HMI with Allen-Bradley PLC*.

2 System Topology

The following figure shows a system topology where the Modbus end devices, **PowerFlex 4M** and **IAQPoint2**, are connected to the serial port on **MGate 5105-MB-EIP** through RS-485-2W wiring. **MGate 5105-MB-EIP** and **PC** (with **RSLogix 5000** and **KingSCADA** installed) are connected to the Ethernet switch ports on **Allen-Bradley ControlLogix PLC** via Ethernet cables. **A fan** is connected to **PowerFlex 4M** that outputs electric current to power the fan.



3 Hardware and Software Requirements

- **Allen-Bradley ControlLogix PLC:**
  - **Processor:** 1756 L71 ControlLogix5571
  - **Chassis:** 1756-A7
  - **EIP Module:** 1756-EN2TR
- **PowerFlex 4M:**

  PowerFlex 4M is an adjustable frequency AC drive (converter).
- **IAQPoint2:**

  IAQPoint2 is an indoor air quality monitor. It can detect CO2, temperature and humidity levels.
- **KingSCADA:**

  A SCADA system released by WellinTech.

  Rev.: V3.1.
- **RS Logix 5000:**

  Allen-Bradley ControlLogix PLC Edit Program from Rockwell Automation.
  - Rev.: V20
  - **Operating System Support:**

    The RSLogix 5000 V20 software has been tested on the following operating systems:

    a. Microsoft Windows 7 Professional (64-bit) with Service Pack 1
    b. Microsoft Windows 7 Home Premium (64-bit) with Service Pack 1
    c. Microsoft Windows 7 Home Premium (32-bit) with Service Pack 1
    d. Microsoft Windows Vista Business (32-bit) with Service Pack 2
    e. Microsoft Windows XP Professional with Service Pack 3
    f. Microsoft Windows Server 2008 R2 Standard Edition with Service Pack 1
    g. Microsoft Windows Server 2008 Standard Edition with Service Pack 2
    h. Microsoft Windows Server 2003 R2 Standard Edition with Service Pack 2
- **RSLinx:**

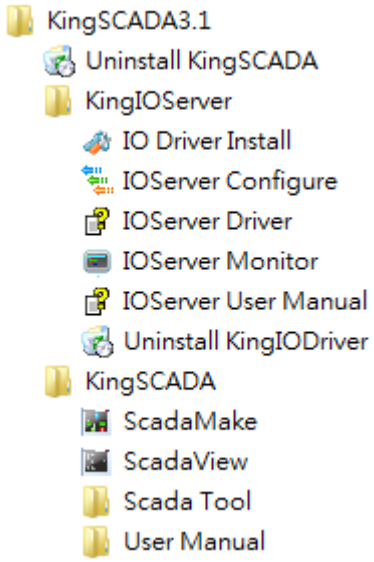
  Communication Utility from Rockwell Automation.

  Rev.: 2.59

4 About KingSCADA 3.1

KingSCADA 3.1 consists of the following components:

- **KingIOServer:** This component acquires data from I/O devices.
- **KingScada:**
  - **ScadaMake:** This is the development environment.
  - **ScadaView:** This is the runtime application.



The following sections describe these components.

4.1 About KingIO Server

A KingSCADA station communicates with I/O devices through KingIOServer. KingIOServer is used to communicate with the on-site devices and acquire real-time data and control on-site data of the modules.

KingIOServer supports popular PLC, intelligent module, intelligent instrument, transducer, and data acquisition boards, etc.

In addition, KingIOServer can communicate with devices through standard communicate interface to transfer data.

With KingIOServer, site engineers are not required to be familiar with the codes and device communication protocols. Instead, they only need to know how to connect with I/O devices and create tags corresponding with the I/O variables.

4.2 About KingSCADA

You can use KingSCADA is to **create a project** with **data and display**.

The following lists the major steps to create a project:

Step1: Create a new project

Create a new directory to store the documents associated with the project.

Step2: Configure the hardware

Configure the hardware settings of the equipment used in the project.

Step3: Define variables (tags)

Define global variables including memory variables and I/O devices.

Step4: Create graphics:

Draw monitoring pictures according to the project requirements.

Step5: Define animation links

Based on the on-site monitoring requirements, define the animation effects for static pictures to simulate process control objects.

Step6: Write an event script

Create scripts in order to complete the control process.

Step7: Configure of other necessary functions

Configure settings such as networks, recipes, SQL access, and web browsing.

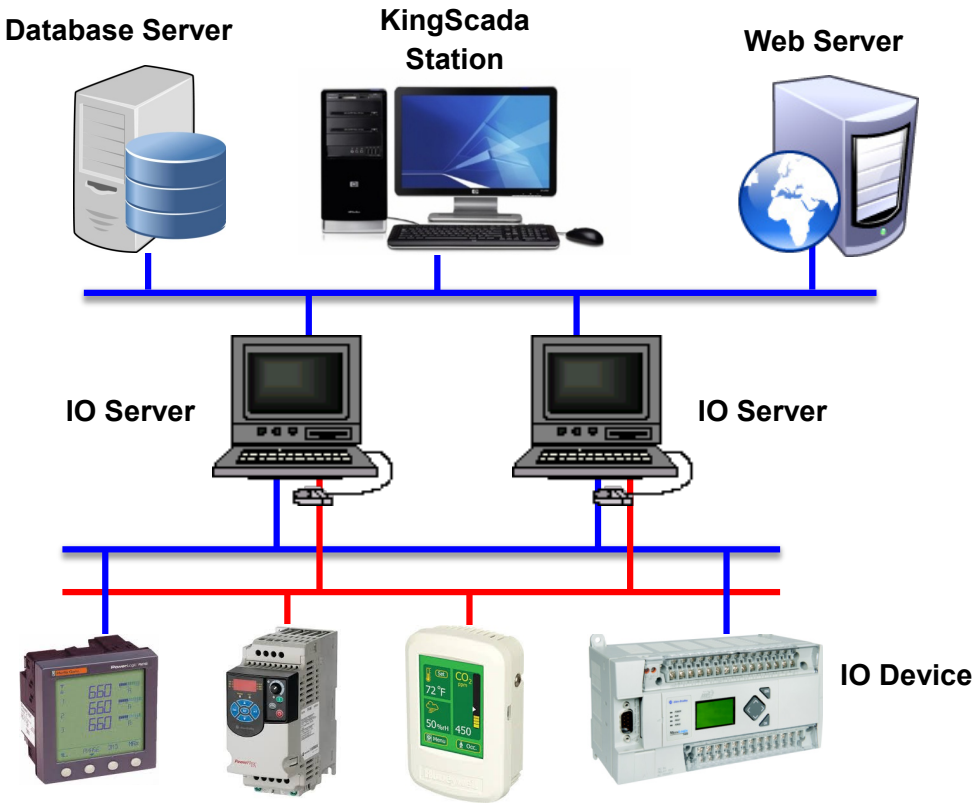
Step8: Operate and debug the project.

After you complete the procedure as described, you can create a simple project. Then, you can run **ScadaView** to start run-time operations.

4.3 KingScada System Architecture

In a large system, the KingScada ystem may deploy some services on multiple servers for load sharing or for security considerations. For example, a run-time project can be executed on KingScada Station for monitoring, on **HMI View** for controlling, and on another **Database Server** for data acquisition and storage. In addition, the web portal can be hosted on another **Web Server** and data can be obtained though several **IO Servers**.

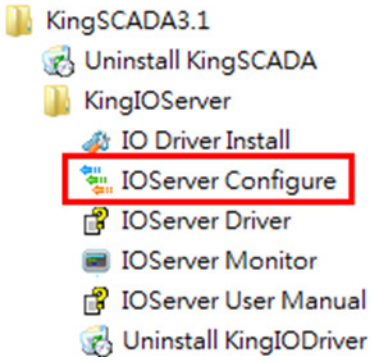
In this document, we show hot to create an IO server and Run-Time View on the same PC.



5 KingSCADA Configuration

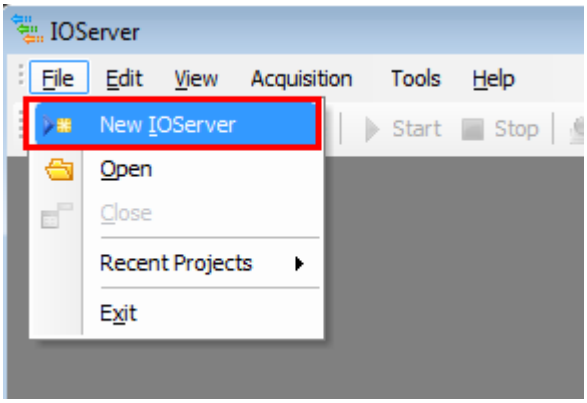
5.1 Creating an IO Server Project

To start the **IOServer Configure** application, click **Start → Program → KingSCADA3.1 → KingIOServer → IOServer Configure** to create an **IO Server** project.

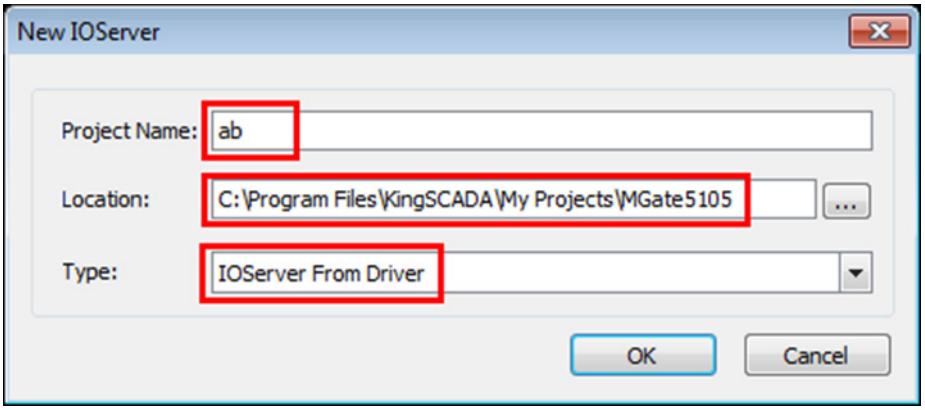


5.1.1 Creating an IO Server

1. In the IOServer screen, click **File → New IOServer** to create a new IO server.

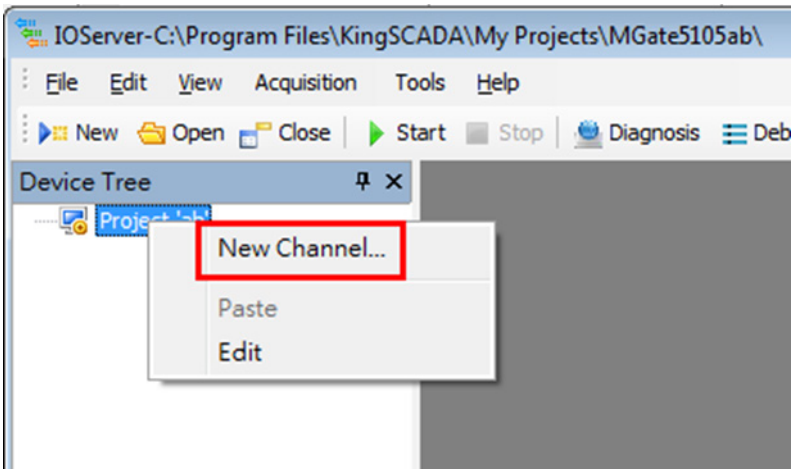


2. In the New IOServer screen, configure the following fields and click **OK**:
   - **Project Name:** Enter a descriptive name.
   - **Location:** Click the … button to choose a location to store the project.
   - **Type:** Select IOServer From Driver from the drop-down list.

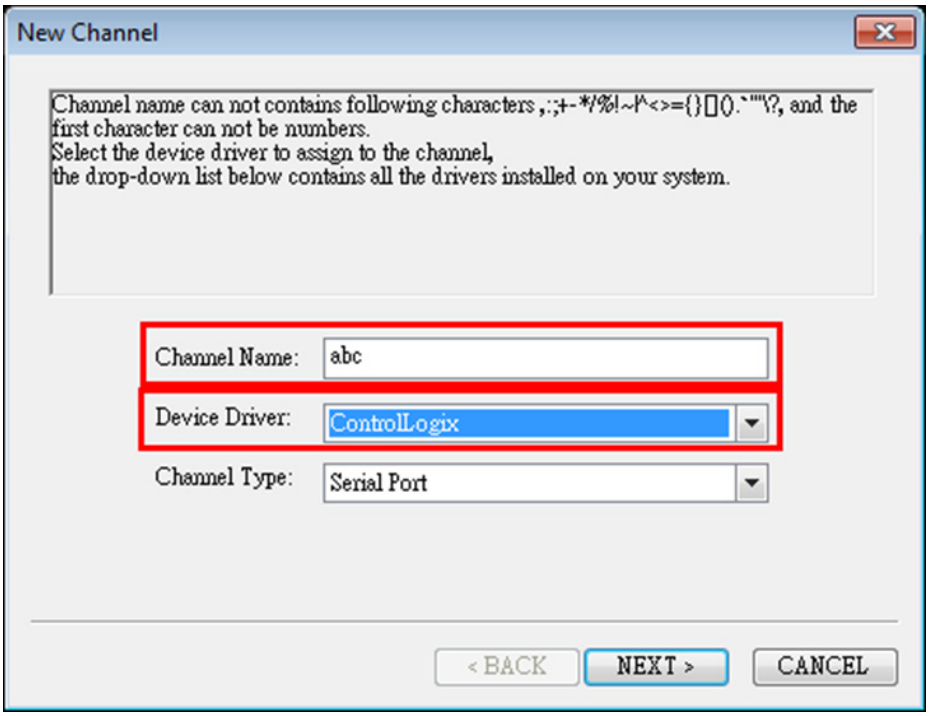


5.1.2 Creating an IO Channel

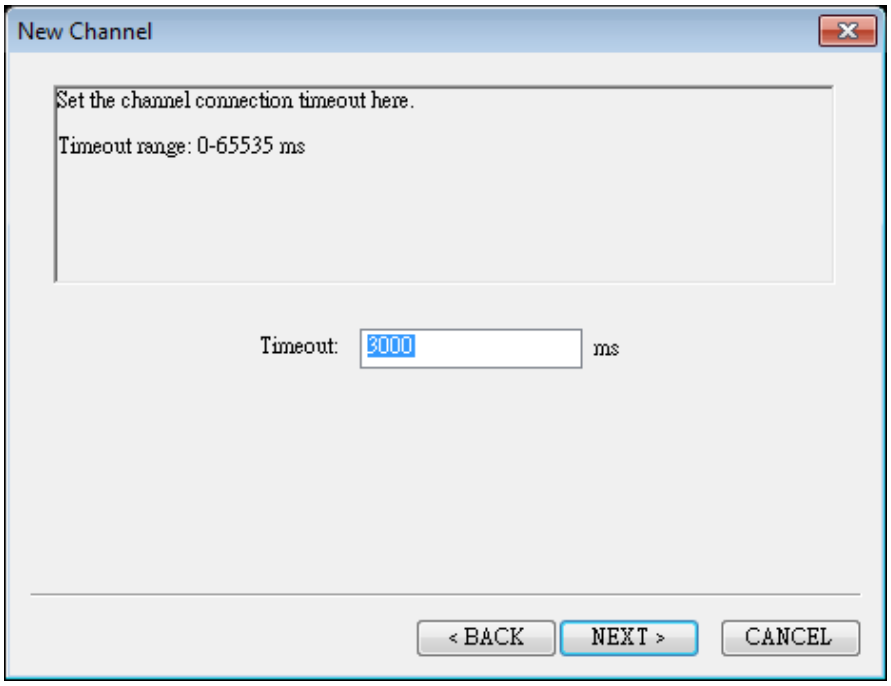
1. Right-click **Device Tree → Project-'ab'** and select **New Channel**. A **New Channel** dialog box appears.



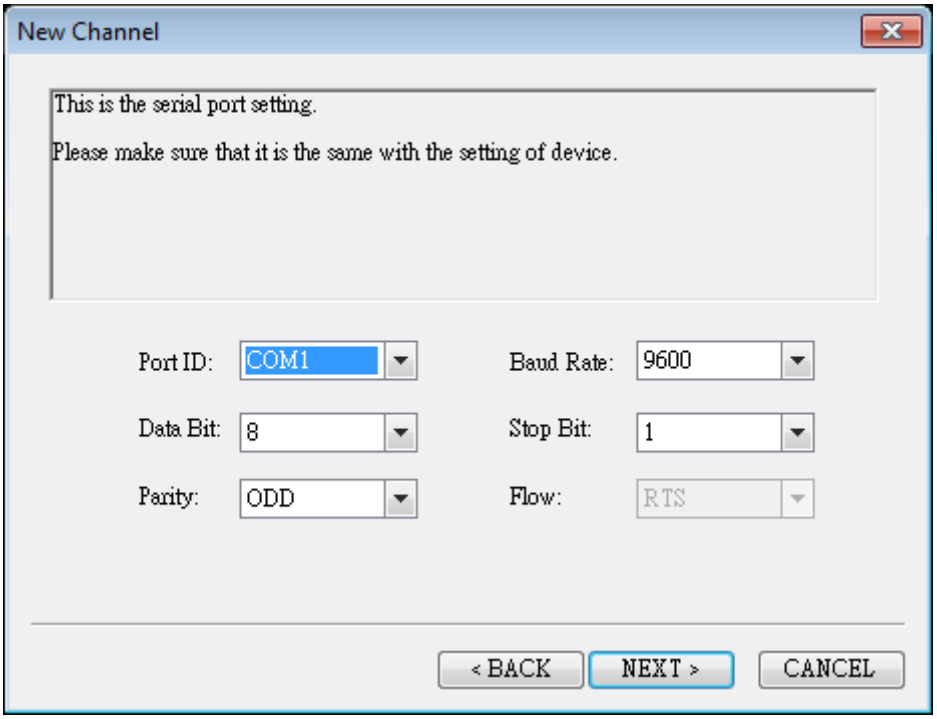
2. In the New Channel dialog box, configure the following fields and click **NEXT**.
   - **Channel Name:** Enter a descriptive name.
   - **Device Driver:** Select ControlLogix from the drop-down list.
   - **Channel Type:** Use the default option.



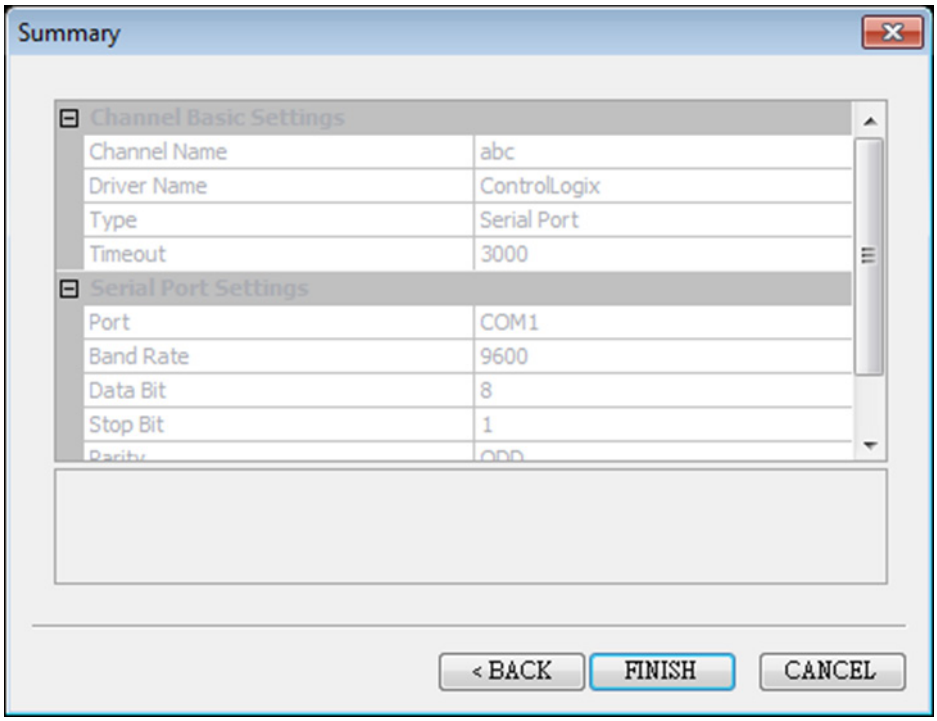
3. Accept the default timeout setting (3000 ms) and click **NEXT**.



4. Since this channel is connected through Ethernet, use the default settings for the serial port. Click **NEXT** to continue.

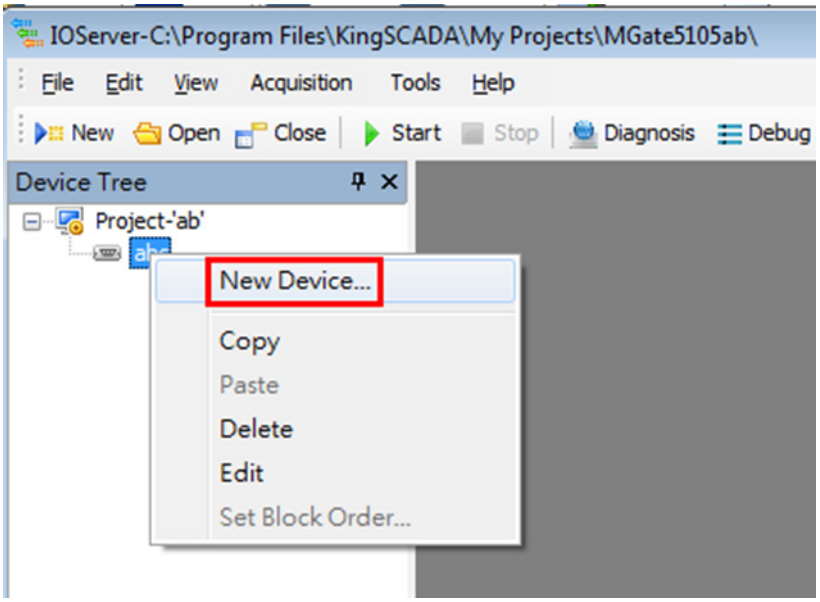


5. Click **Finish** to complete the "Allen-Bradley ControlLogix PLC" channel setting.



5.1.3 Creating an IO Device

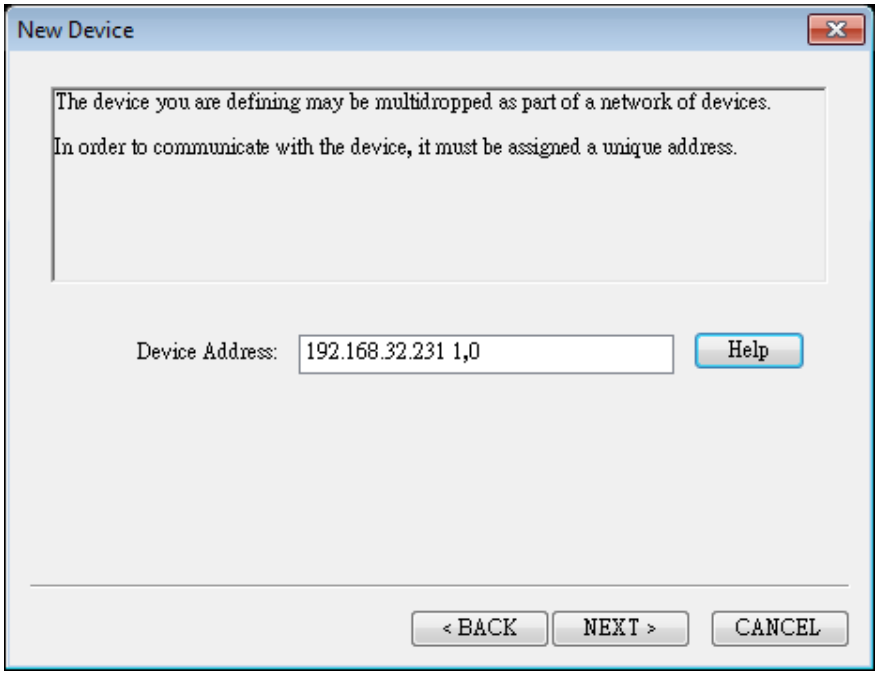
1. Right-click the **abc** channel and select **New Device**.



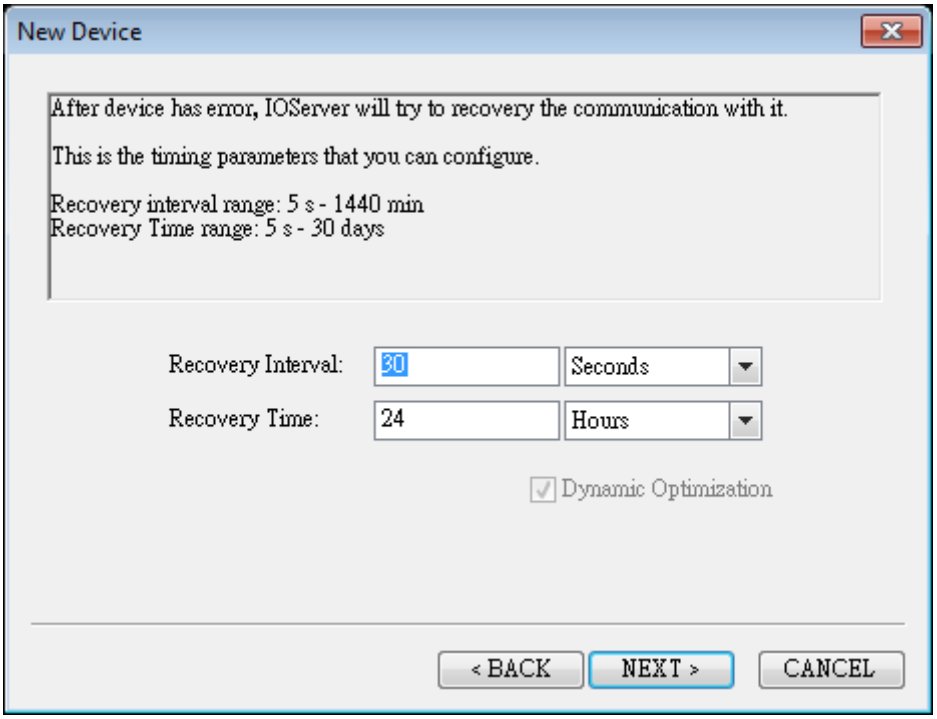
2. In the **New Device** screen that appears, configure the following fields and click **NEXT**:
   - **Device Name:** Enter a descriptive name.

- **Device Series:** Select **AB-ControlLogixTCP** from the drop-down list.

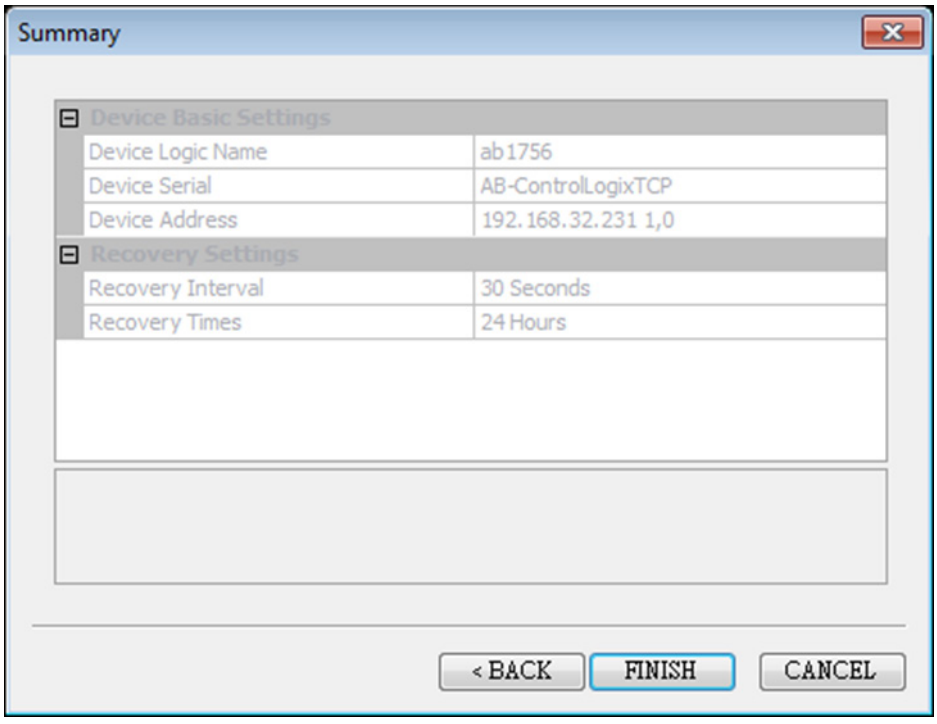
3. In the **Device Address** field, enter the IP address and communication parameter of **Allen-Bradley ControlLogix PLC** in the format [*IP Backplane Slot*]. Then, click **NEXT**.



4. Accept the default settings and click **NEXT** to continue.

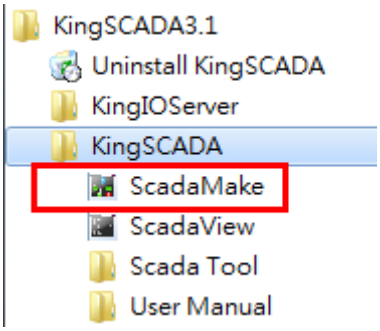


5. Click **Finish** to complete the device setting.

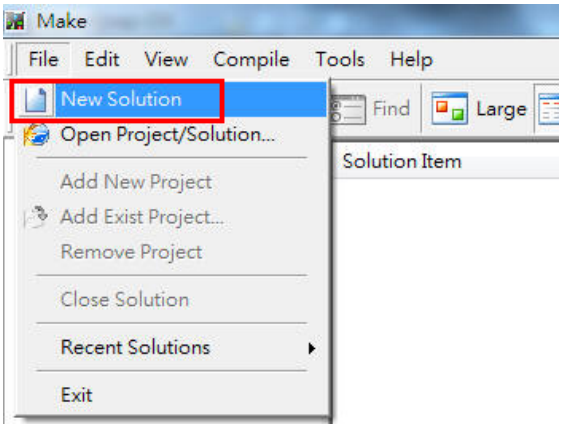


5.2 Creating a SCADA Project

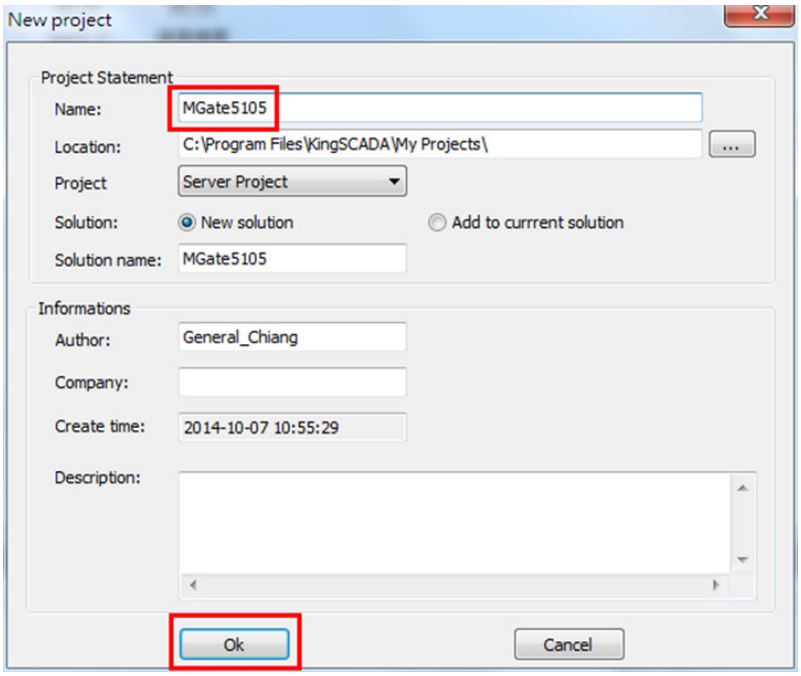
1. To start the **ScadaMake** application, click **Start → Program → KingSCADA3.1 → ScadaMake**.



2. Click **File → New** Solution to create a new solution.

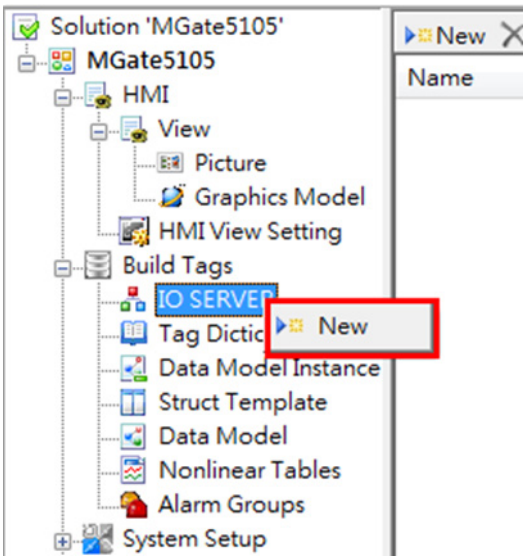


3. Give a project name then click **OK**. System would create a new empty project.

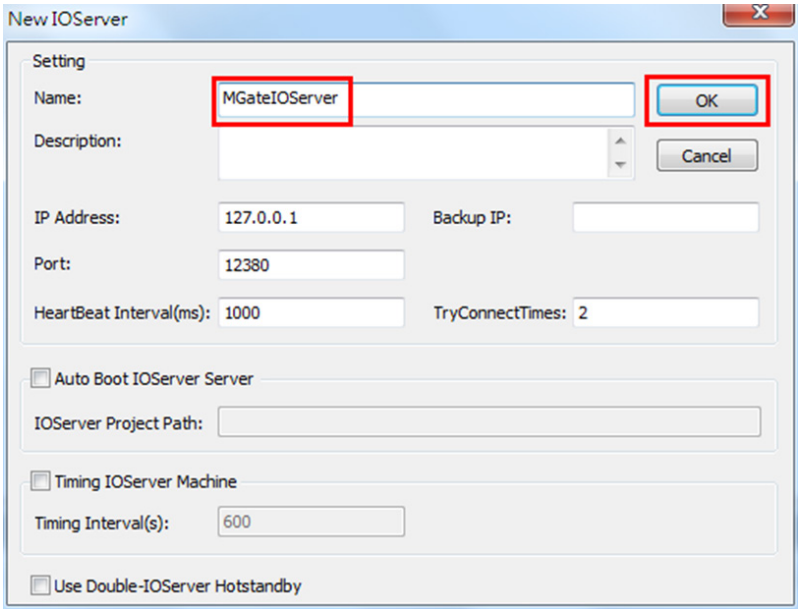


5.3 Building Tags

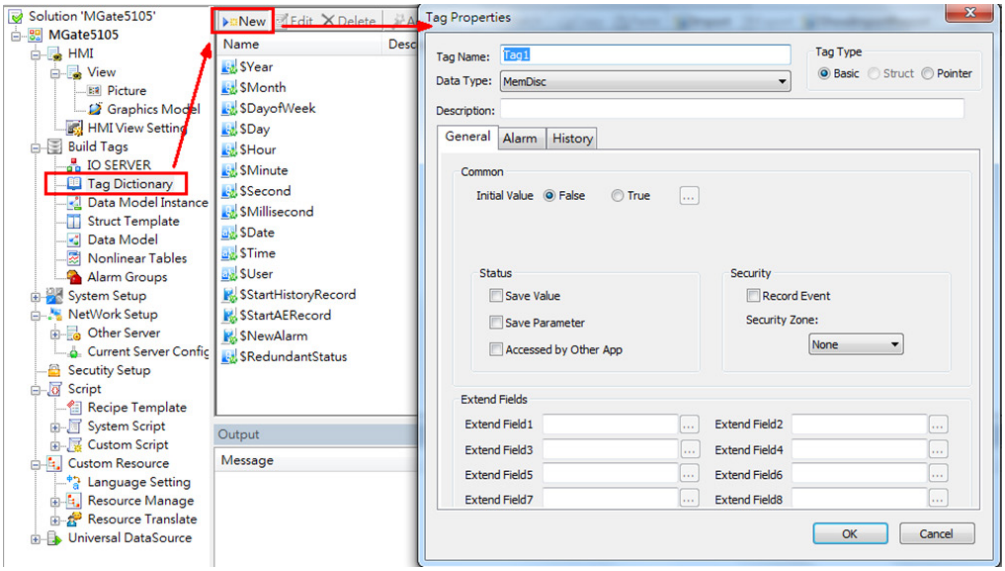
1. Under **Build Tags**, right-click **IO SERVER** and click **New**. The New IOServer screen appears.



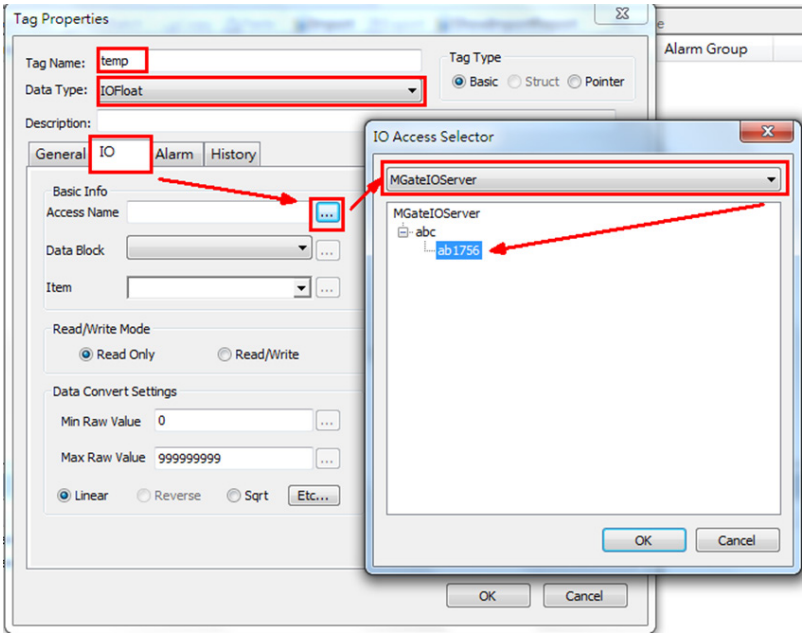
2. In the **New IOServer** screen, enter a server name in the **Name** field and click OK.
Note: Since this IO Server is hosted on the same computer, the IP address is unchanged at 127.0.0.1.



3. Click **Build Tags → Tag Dictionary** and click **New**. The Tag Properties screen appears.

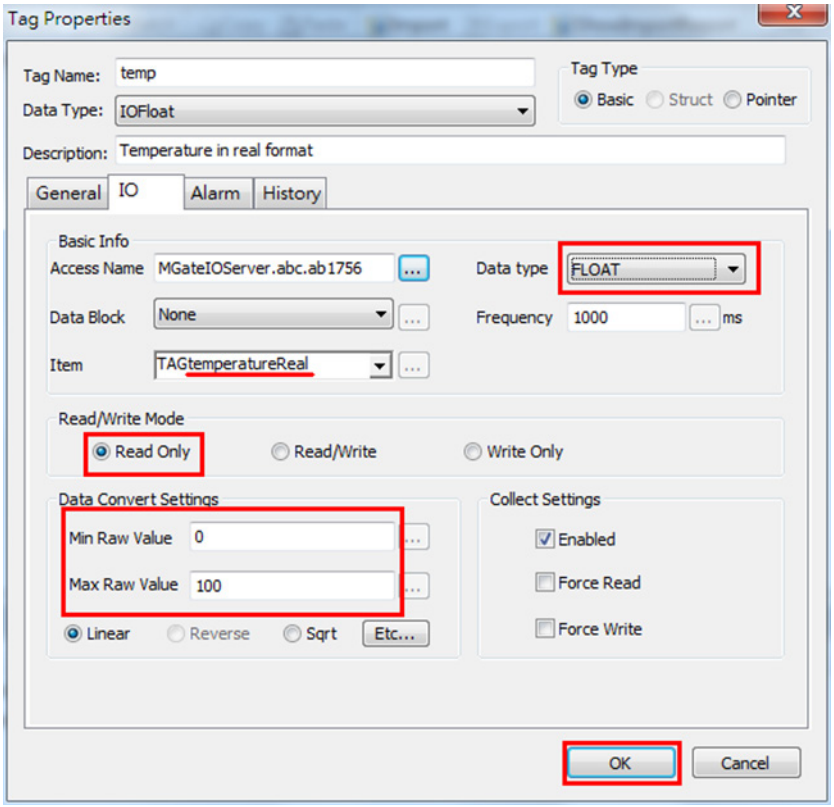


4. In the Tag Properties screen, configure the following fields:
   - **Tag Name:** Enter "tempReal".
   - **Data Type:** Select **IOFloat** from the drop-down list.
5. Select the **IO** tab. For **Access Name**, click the … button to select the **ab1756** IO device (in the IO Access Selector pop-up window, click **MGateIOServer → abc → ab1756**).

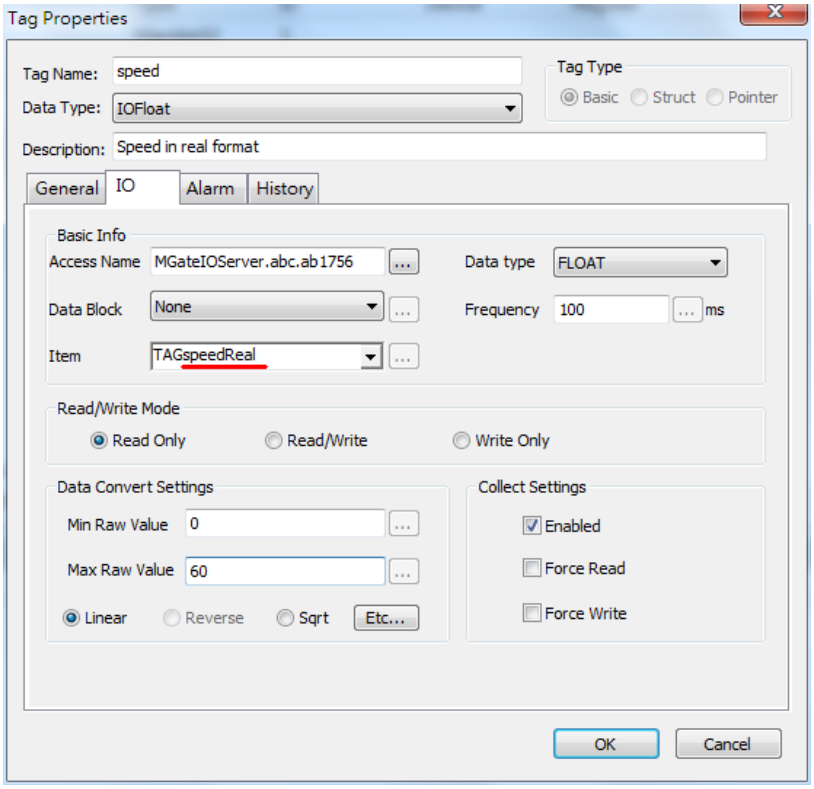


6. In the IO tab, configure the following fields and click **OK** to finish:

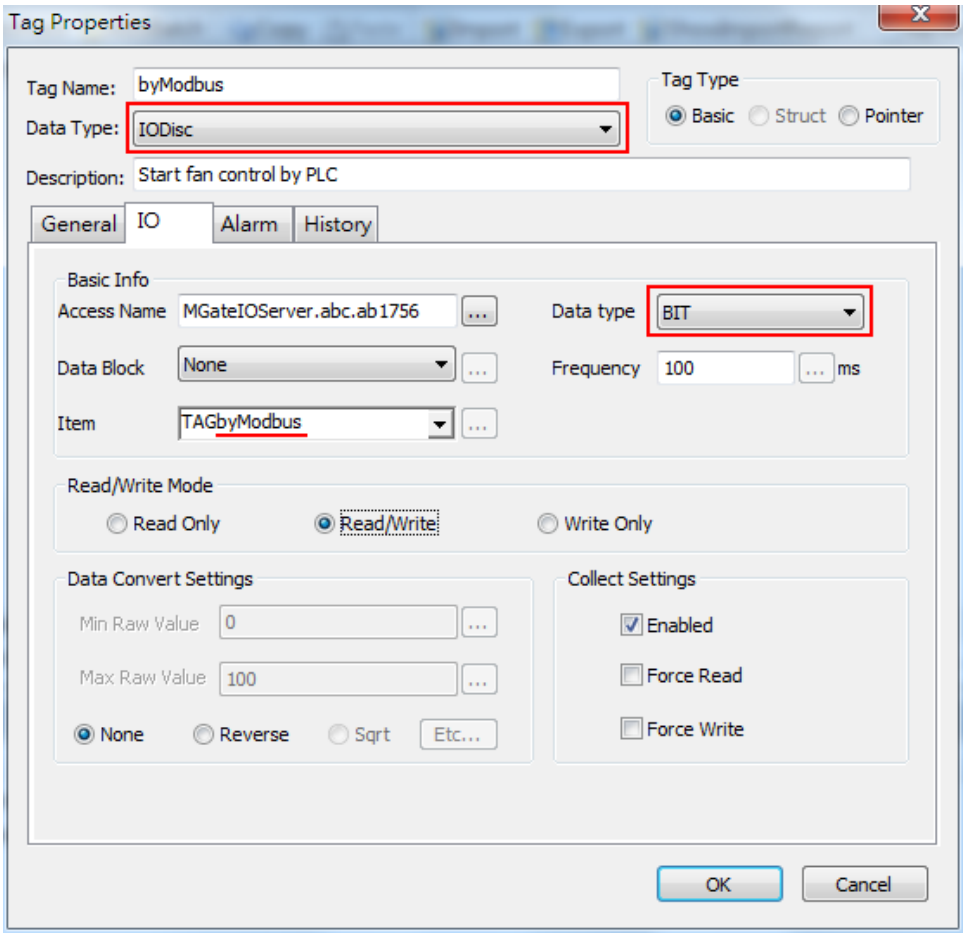
- **Data type:** Select **FLOAT** from the drop-down list.
- **Data Block:** Select **None** from the drop-down list.
- **Item:** Select or enter "TAGtemperatureReal". The format is [TAG*tagname*] as specified in Allen-Bradley ControlLogix PLC.
- **Read/Write Mode:** Select **Read Only**.
- **Data Convert Settings:** Enter "0" in the **Min Raw Value** field and "100" in the **Max Raw Value** field.



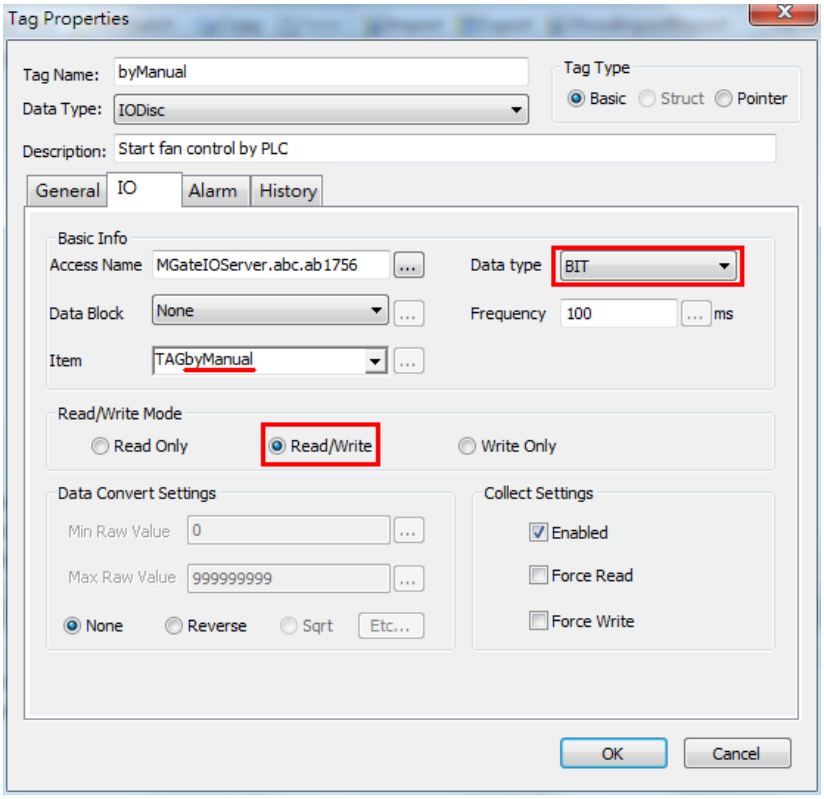
7. Repeat steps 1 to 5 to create another tag to read the **speedReal** tag from Allen-Bradley ControlLogix PLC. Configure the tag properties as shown in the following figure.



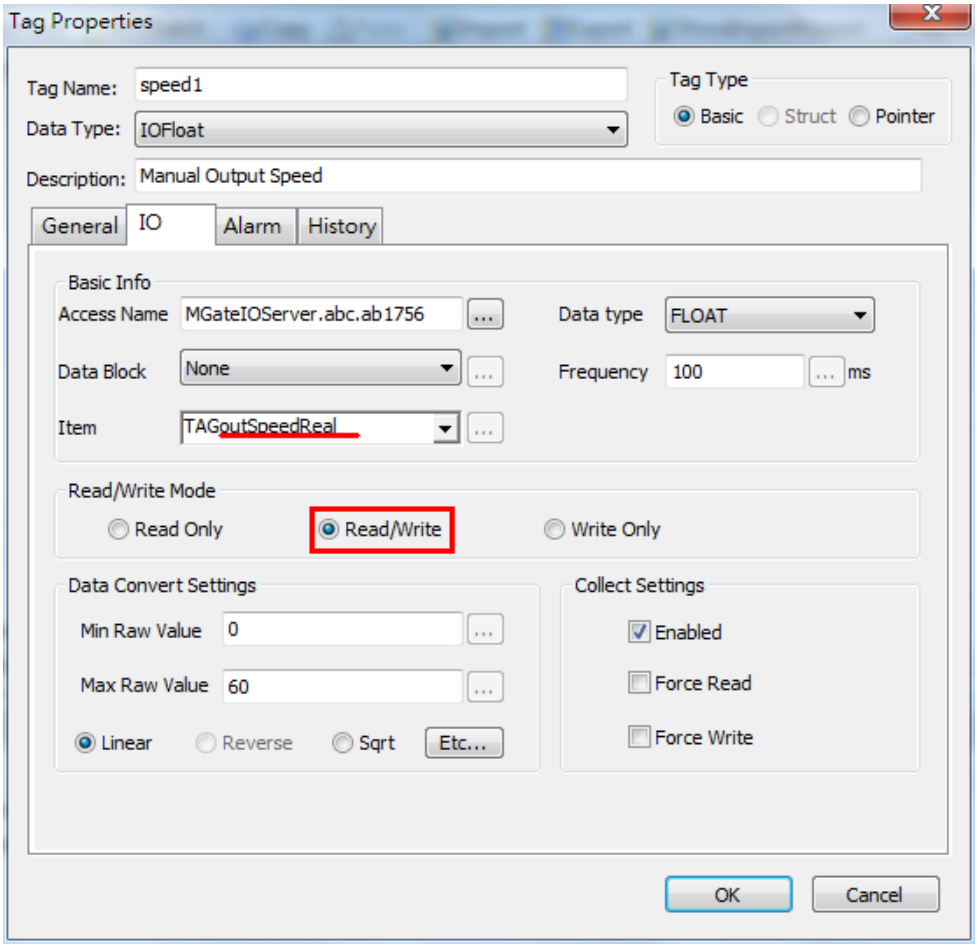
8. Repeat steps 1 to 5 to create a **byModbus** tag to read and write the **byModbus** tag from/to Allen-Bradley ControlLogix PLC. Configure the tag properties as shown in the following figure.



9. Repeat steps 1 to 5 to create a **byManual** tag to read and write the **byManual** tag from/to Allen-Bradley ControlLogix PLC. Configure the tag properties as shown in the following figure.



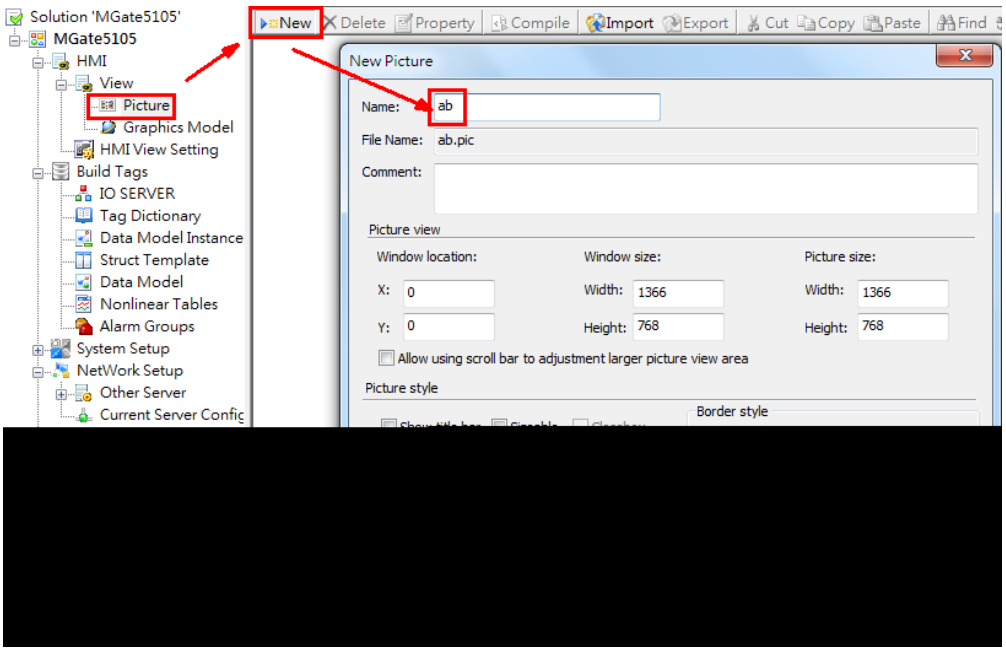
10. Repeat steps 1 to 5 to create a **speed1** tag to read and write the **outSpeedReal** tag from/to Allen-Bradley ControlLogix PLC. Configure the tag properties as shown in the following figure.



5.4 Creating HMI View

5.4.1 Creating a View Picture

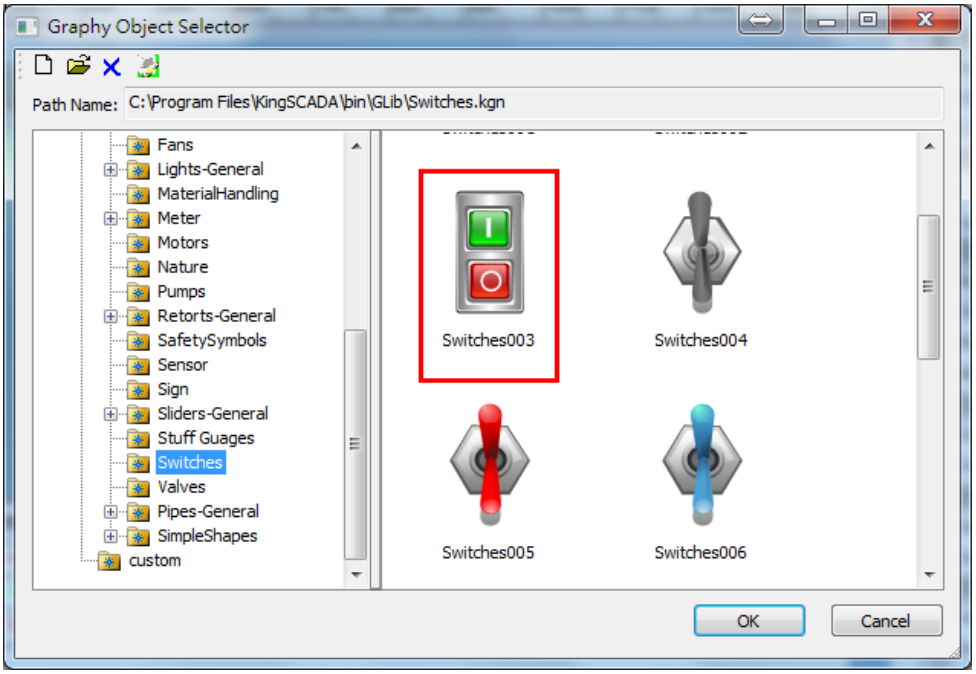
1. Click **HMI → View → Picture** and click **New** to create a view picture.



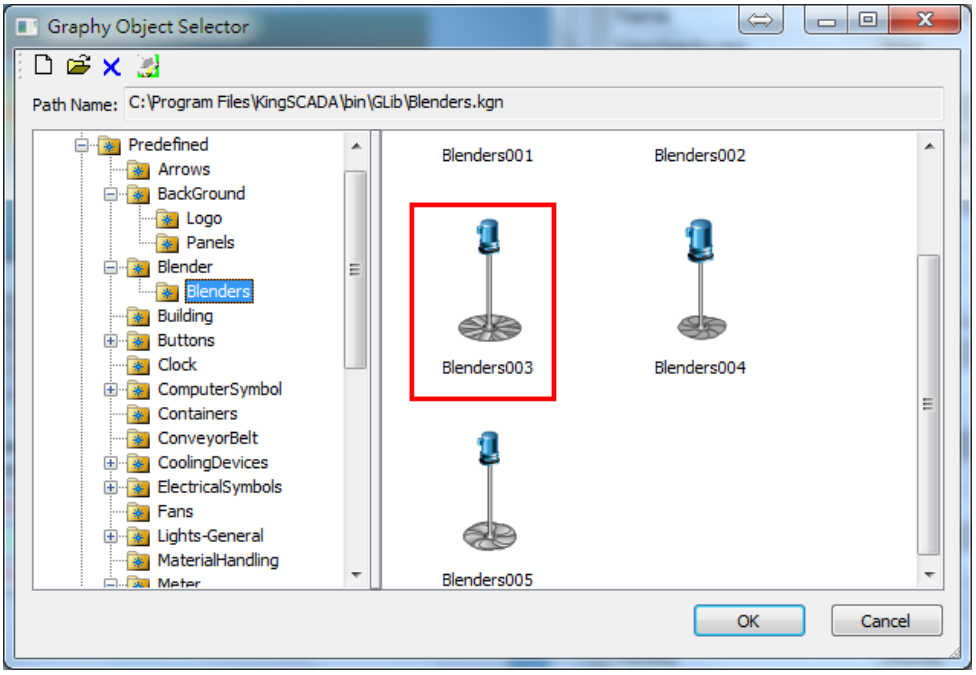
2. After adding object graphics, the following figure shows the complete view picture.

From **File → Open → Genius**, you can get the Switches, Blenders, and Sign object pictures as shown in the following figures.

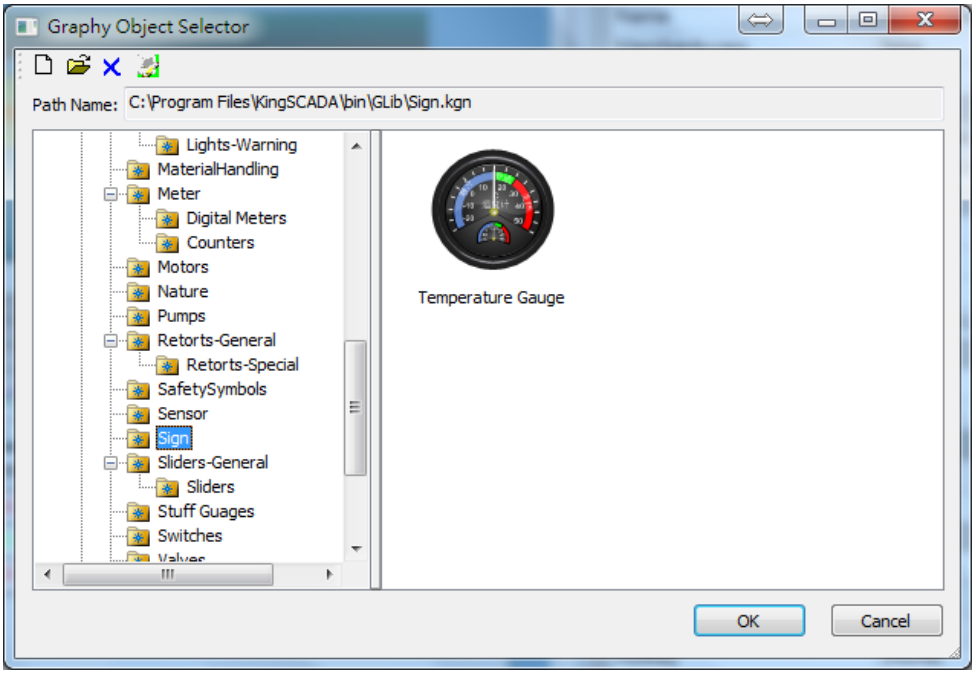
- **Switches**



- **Blenders**



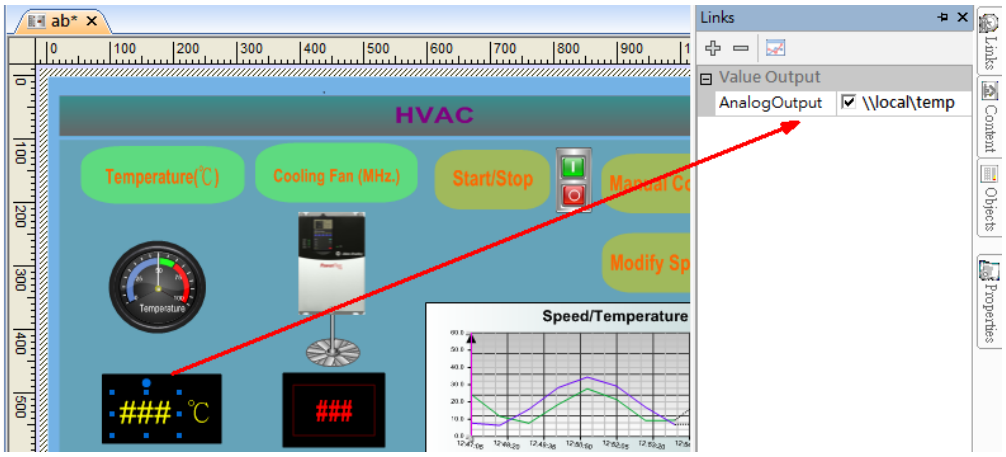
- **Sign**



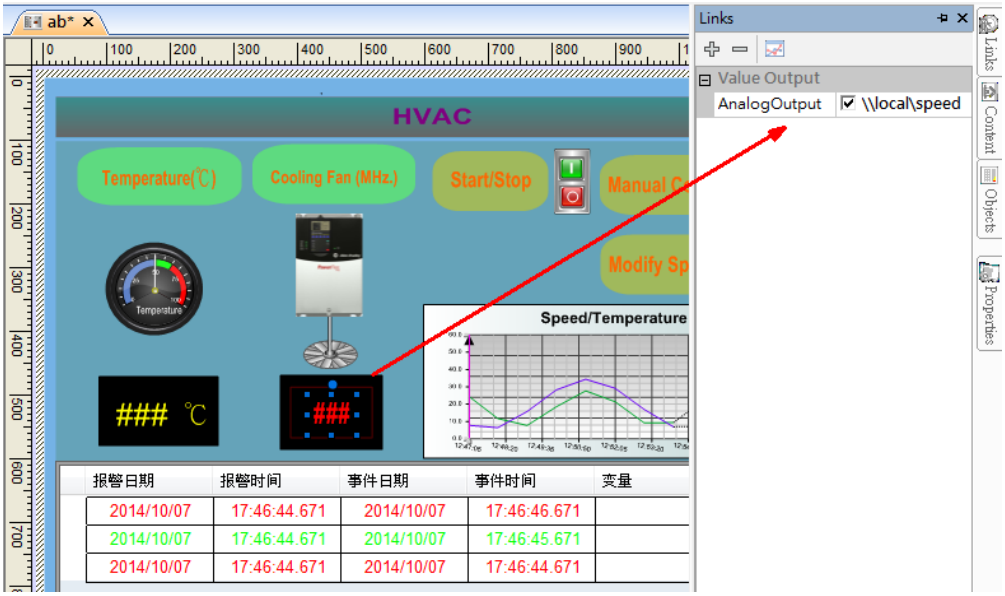
5.4.2 Adding Links

After adding the object pictures, add links to enable the system to dynamically update object graphic animation or input and output element value.

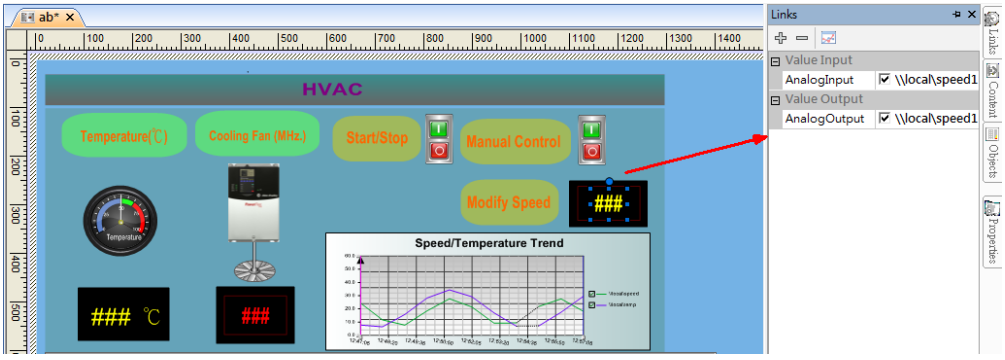
1. Select the **Temperature** input box; then, in the Links window, click the + icon to add the **AnalogOutput** source as the **\\local\temp** tag.



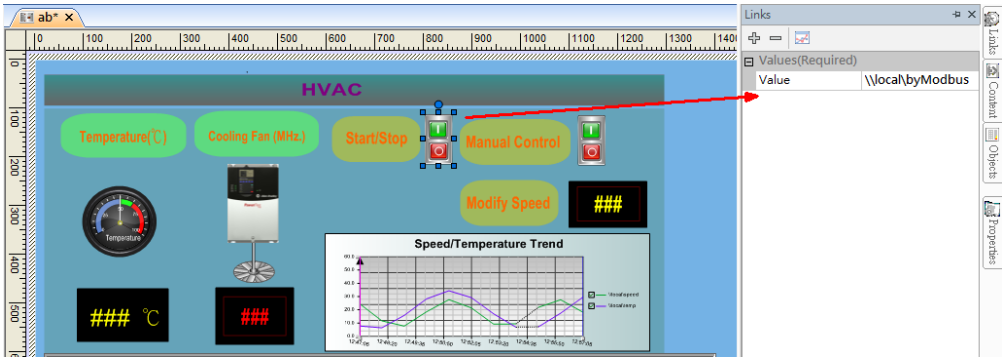
2. Select the **Speed** input box; then, in the Links window, click the + icon to add the **AnalogOutput** source as the**\\local\speed** tag.



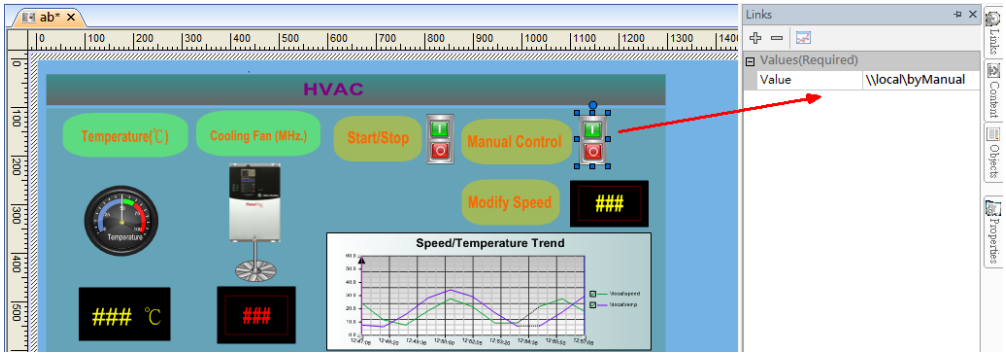
3. Select the **Modify Speed** input box; then, in the Links window, click the + icon to add the **AnalogOutput** source as the **\\local\outSpeed1** tag and the **AnalogInput** source as the **\\local\outSpeed1** tag.



4. Select the **Start/Stop** Switch; then, in the Input window, click the + icon o add the **Value** source as the **\\local\byModbus** tag.

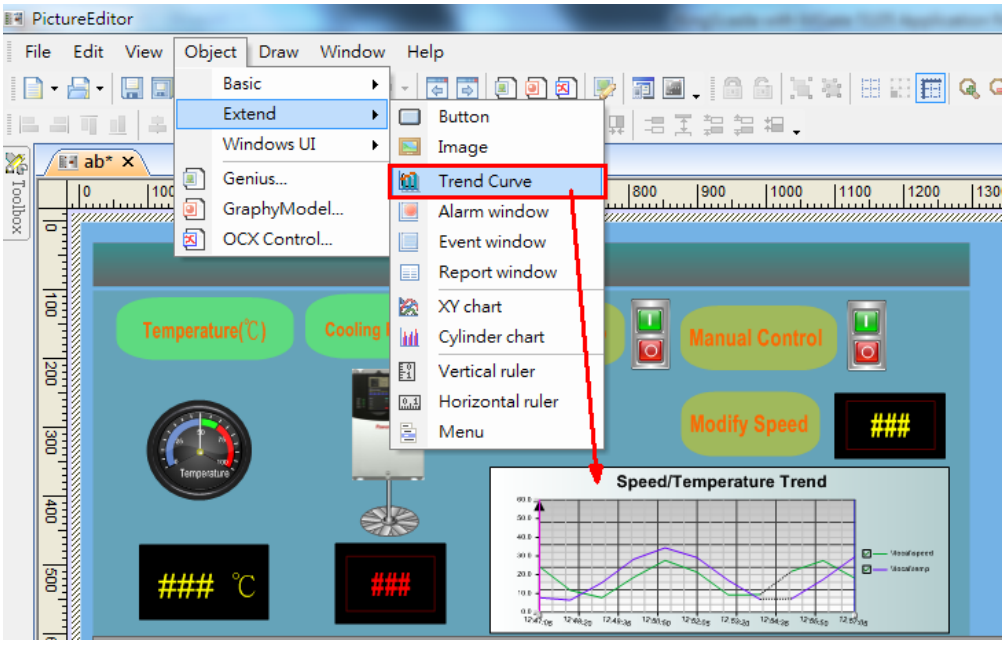


5. Select the **Manual Control** switch; then, in the Input window, click the + icon to add the **Value** source as the **\\local\byManual** tag.

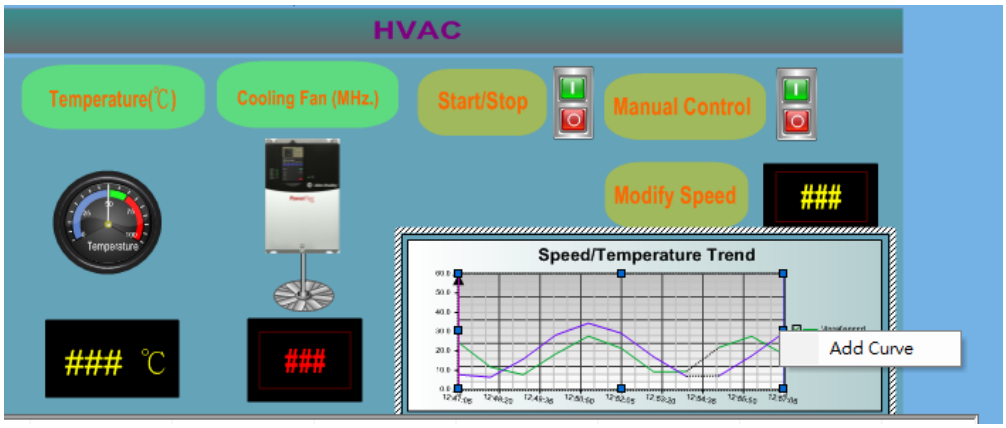


5.4.3 Adding a Trend Curve

1. In the PictureEditor screen, click **Object → Extend → Trend Curve** to create a trend curve chart.

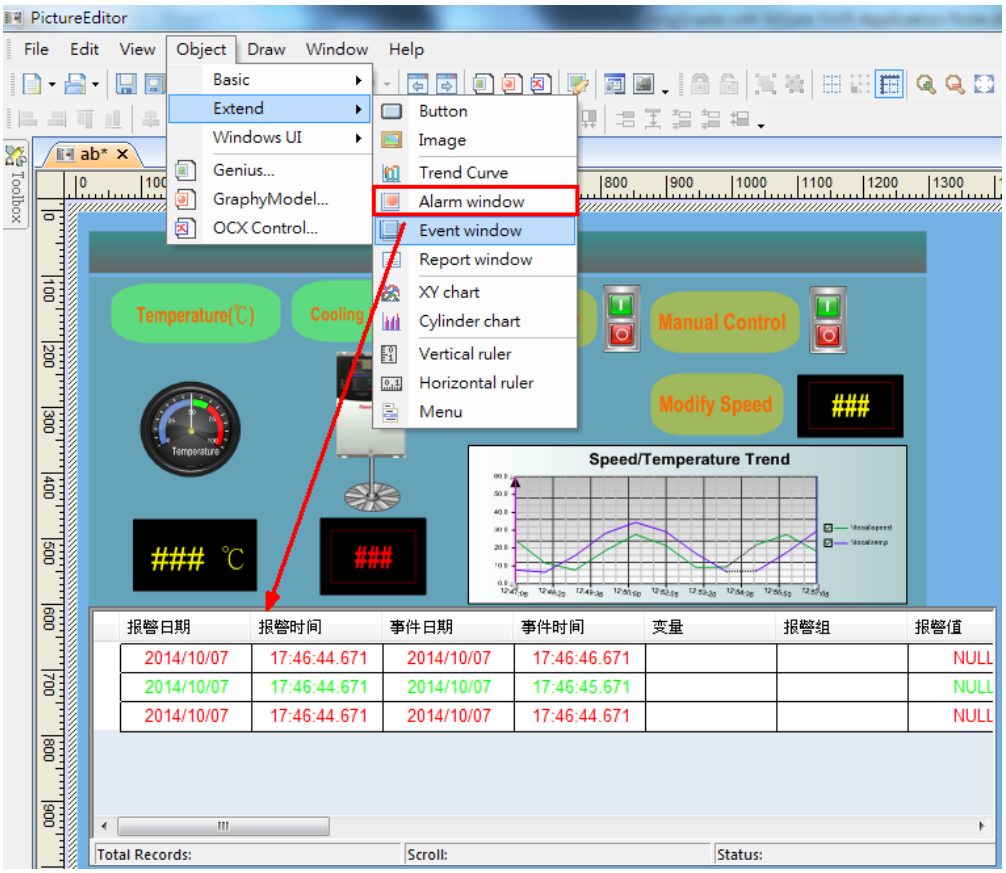


2. Right-click on the Chart object and click **Add Curve** to add the **Speed** and **Temperature** curves that are linked to the \\local\speed and \\local\temp tags respectively.

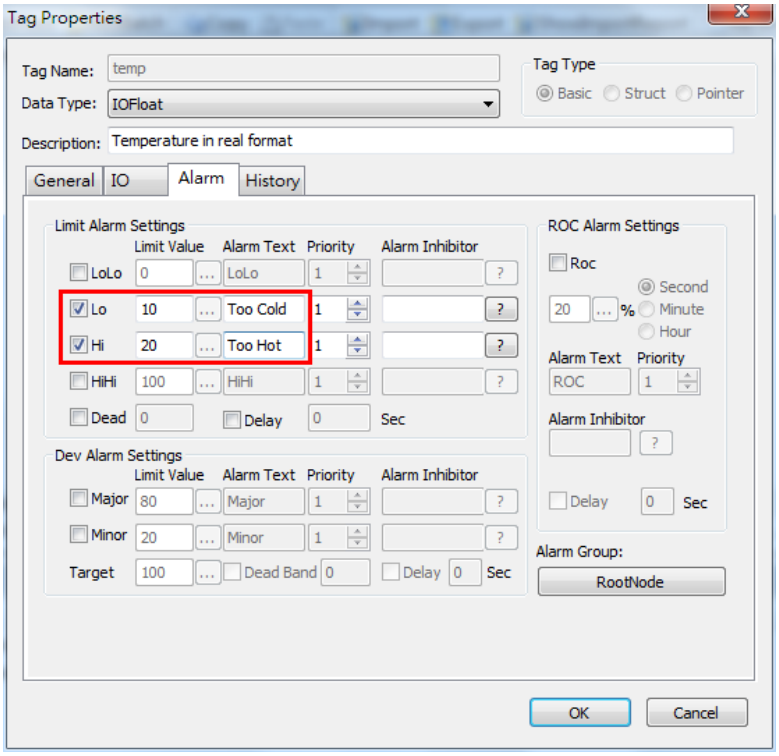


5.4.4 Adding an Alarm Window

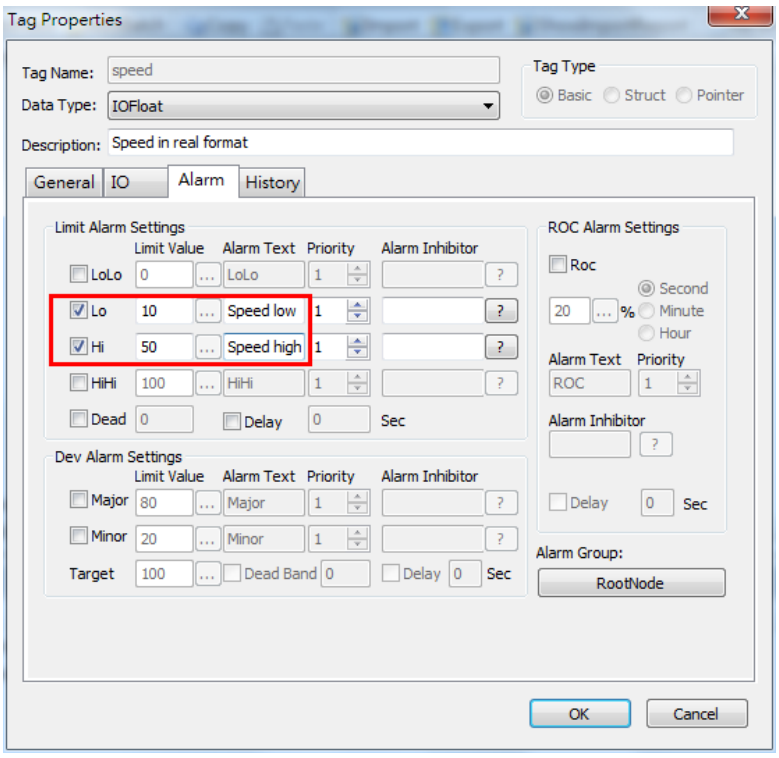
1. In the PictureEditor screen, click **Object → Extend → Alarm** window to create an alarm window.



2. Configure the alarm settings for the **Temp** tag as shown in the following figure.

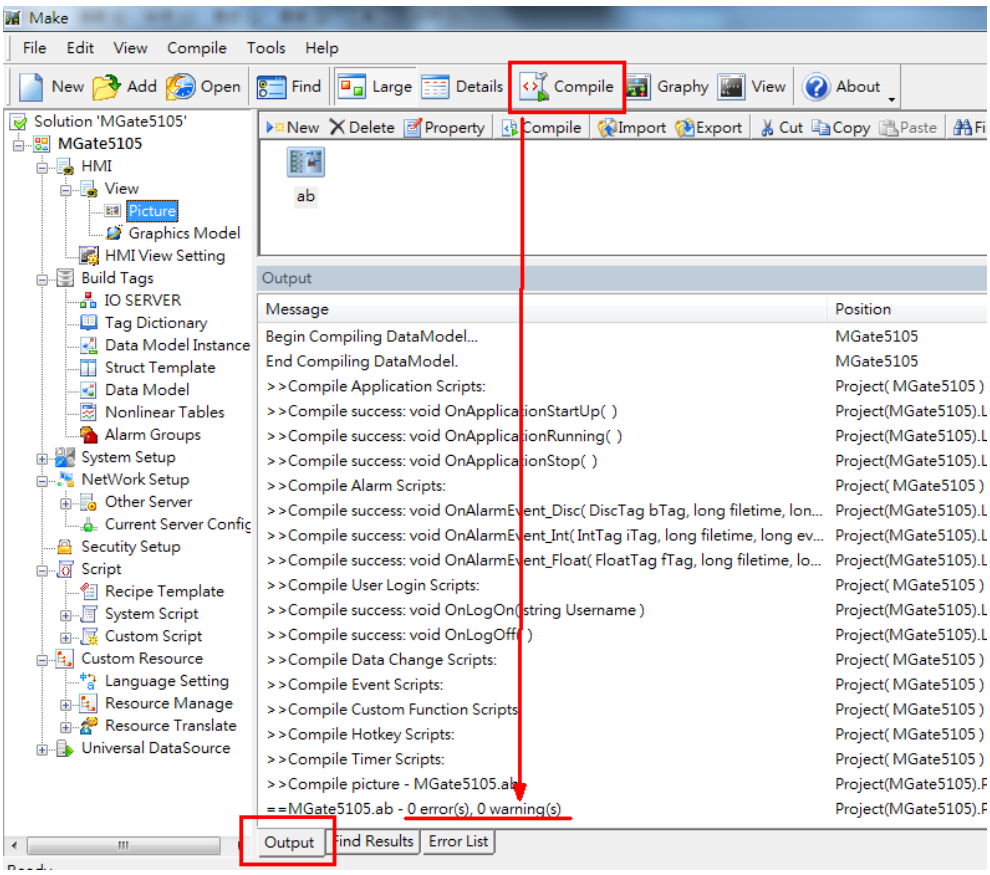


3. Configure the alarm settings for the **Speed** tag as shown in the following figure.



5.5 Compiling a Project

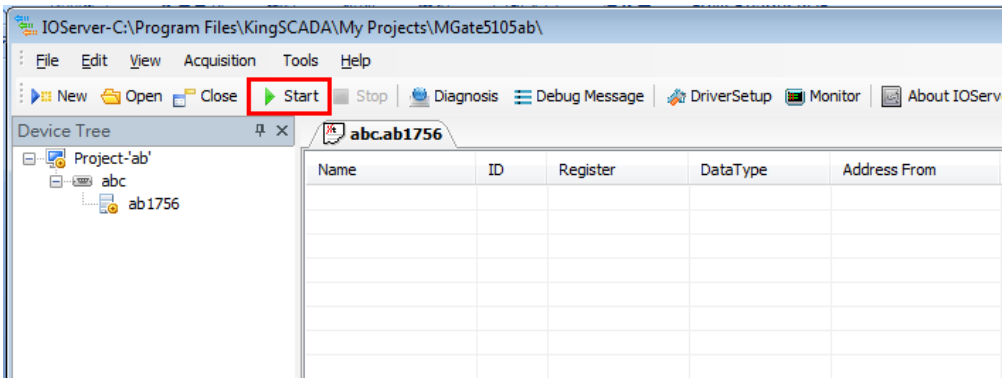
Click **Compile** to compile project. Make sure that there is no error or warning in the **Output** tab window.



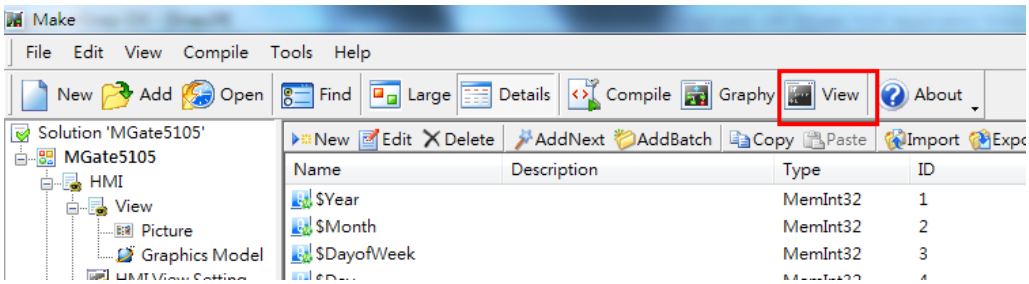
6 Runtime Test

Before you execute **ScadaView**, you must start the IO server first.

1. In the **IOServer Configure** program, click **Start** to start the IO server.



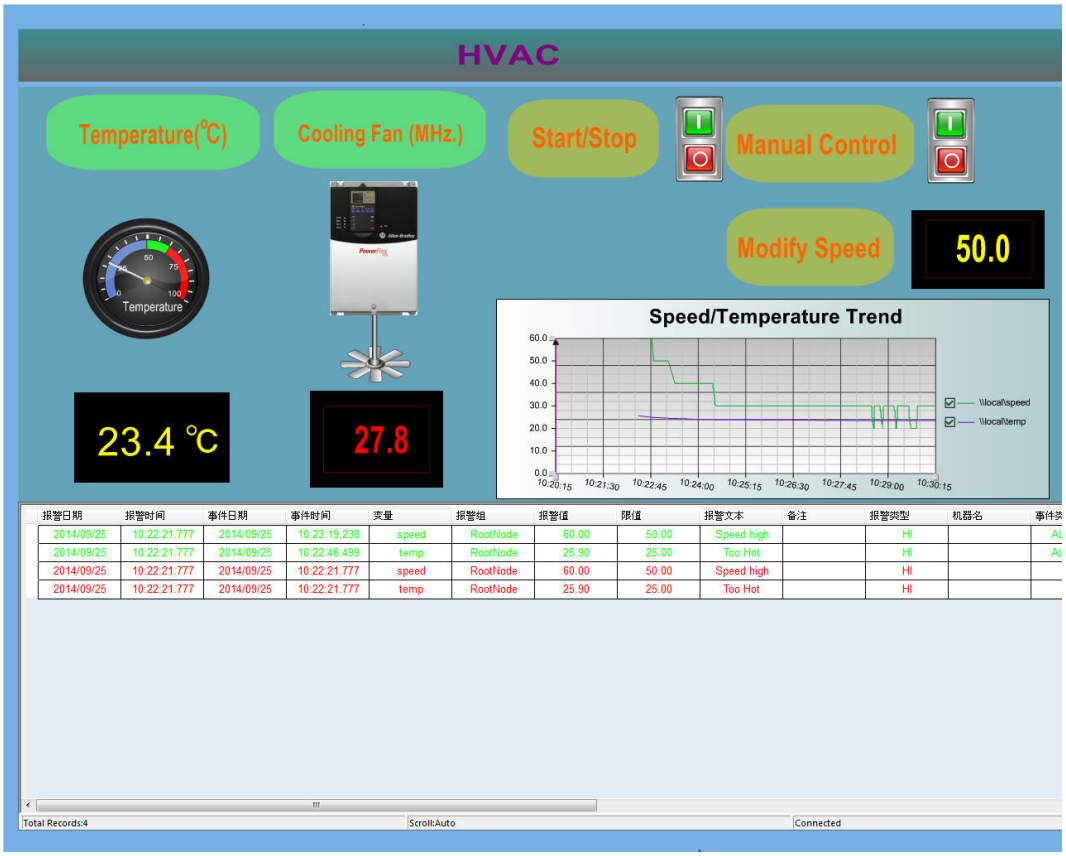
2. In the **ScadaMake** program, click **View** to run the SCADA View system. Alternatively, you can execute the **ScadaView** program to run this project.



The screen displays the **HMI View**. SCADA polls the tags of **Allen-Bradley ControlLogix PLC** and updates values on the HMI View (for example, displaying the current temperature and speed values).

You can click the **Start/Stop** switch to start or stop the PowerFlex 4M converter.

If you user a hairdryer to increase the meter temperature, the converter will increase its output speed. The **Speed/Temperature Trend Chart** area will also show the trend of the temperature and speed change. If the temperature or speed value exceeds the alarm threshold, the system logs an event in the **Alarm Window** area.



If you click the **Manual Control** switch to enable the manual speed control function, an Input dialog box appears that allows you to specify the speed value in the **Input analog value** field.