

# 工业物联网网关 IOT-861

## User Manual V 1.1 REV A



**SiboTech Automation Co., Ltd**

**Technical Support: +86-21-5102 8348**

**E-mail: [gt@sibotech.net](mailto:gt@sibotech.net)**

## 目 录

一、引言.....	3
1.1 关于说明书.....	3
1.2 版权信息.....	3
1.3 术语.....	3
二、产品概述.....	4
2.1 产品功能.....	4
2.2 产品特点.....	4
2.3 技术指标.....	4
2.4 电磁兼容性能.....	5
2.4.1 高频干扰试验( GB/T15153.1 classIII ).....	5
2.4.2 快速瞬变脉冲群试验( GB/T17626.4 classIII ).....	5
2.4.3 静电放电干扰( GB/T 17626.2 classIII ).....	6
2.4.4 辐射电磁场( GB/T 17626.3 classIII ).....	6
三、硬件说明.....	7
3.1 产品外观.....	7
3.2 指示灯.....	8
3.3 接口.....	8
3.3.1 电源接口.....	8
3.3.2 以太网接口.....	9
3.3.3 RS-485/RS-422 接口.....	9
3.3.4 RS-232 接口.....	10
3.4 看门狗.....	11
四、快速应用指南.....	12
五、典型应用.....	40
六、安装.....	41
6.1 机械尺寸.....	41
6.2 安装方法.....	42
七、修订记录.....	43

# 一、引言

## 1.1 关于说明书

本说明书描述了适配器 IOT-861 的各项参数，具体使用方法和注意事项，方便工程人员的操作运用。在使用适配器之前，请仔细阅读本说明书。

## 1.2 版权信息

本说明书中提及的数据和案例未经授权不可复制。泗博公司在产品的发展过程中，有可能在不通知用户的情况下对产品进行改版。

**SiboTech<sup>®</sup>** 是上海泗博自动化技术有限公司的注册商标。

该产品有许多应用，使用者必须确认所有的操作步骤和结果符合相应场合的安全性，包括法律方面，规章，编码和标准。

## 1.3 术语

IOT: Internet of Things, 物联网

REST、MQTT: 物联网中用到的通信标准协议。

## 二、产品概述

### 2.1 产品功能

IOT-861 物联网网关是 SiboTech 公司推出的针对工业物联网和互联的网关产品，本产品的主要功能是将工业现场 PLC、DCS 及用户自定义协议的数据转成 IoT 协议的数据，并安全、可靠地上传到 Internet 上。

IOT-861 主要应用于电力通信调度、楼宇智能化、IT 管理及系统架构、自动化流水线、能源管理、水力风电、轨道交通信号等行业。

### 2.2 产品特点

- ◆ 融合 OPC 技术，采集底层设备的数据，如 PLC, DCS, OPC Server 等
- ◆ 赋予底层设备 IoT 接口，如 MQTT、REST、ThingWorx，方便接入 Internet 进行集中管理
- ◆ 丰富的接口设计，如 OPC DA、OPC UA、.NET、ODBC 接口等
- ◆ 工业级设计，适应不同应用场合的需要，一体化设计，使得系统的稳定性好、抗干扰能力强、运算速度快。

### 2.3 技术指标

- [1] 能够连接众多主流厂商的设备，如 Siemens、Rockwell Automation、Schneider、ABB、Omron、GE、Mitsubishi、Honeywell、Toshiba、Yaskawa 和 Beckhoff 等；
- [2] 带有 OPC 接口，支持 DA、XML DA、UA 以及 HDA 接口；
- [3] 带有 IOT 接口，支持 RESTful Client/Server、MQTT、ThingWorx；
- [4] 带有数据库接口，支持向 ODBC 接口的数据库写入设备数据；
- [5] 内部带有看门狗电路，支持快速的故障恢复；
- [6] 新一代物联网网关支持多达 200 种不同种类的协议，也可根据现场需求进行特殊规约的开发；
- [7] 具有以太网、串行口等高速可靠的通信接口用于系统扩展，使得设备的扩展性和配置灵活性大大增强；
- [8] 提供主备冗余、切换功能，可满足用户基于安全设想的各种要求；

[9] 外部接口:

- 5 个 USB2.0 接口, 1 个 USB3.0 接口;
- 2 个 100/1000M 网卡, 支持网络冗余
- 自带 8 个 COM 口, COM1 和 COM2 可选支持 RS485/422 或 RS232, 其余支持 RS232
- 支持 HDMI 高清输出及 VGA 输出

[10] 存储: 支持 Mini-SATA 盘, 满足大容量需求, 稳定可靠;

[11] 电源: 电压输入 12VDC;

[12] 机械尺寸(W\*H\*D): 226mm\*90mm\*260mm;

[13] 安装: 4.5mm 螺母壁挂;

## 2.4 电磁兼容性能

### 2.4.1 高频干扰试验( GB/T15153.1 classIII )

施加场所	电源输入回路-对地 电源输入回路之间 交流输入回路-对地	
施加波形	第 1 波波高值 振荡频率 1/2 衰减时间 重复频度 试验设备输出阻抗	2.5~3kV 1.0~1.5MHz ≥6μs 50 回以上/s 150~200Ω

### 2.4.2 快速瞬变脉冲群试验( GB/T17626.4 classIII )

电压峰值	电源输入和交流加入回路: 2kV 弱电回路: 1kV
重复频率	5 kHz

### 2.4.3 静电放电干扰( GB/T 17626.2 classIII )

施加场所	通常运用时,操作者触及部分
电压、电流	6kV 接触放电, 放电的第一个峰值电流 22.5A
次数	每处 1 秒以上的间隔 10 回以上
极性	正极性

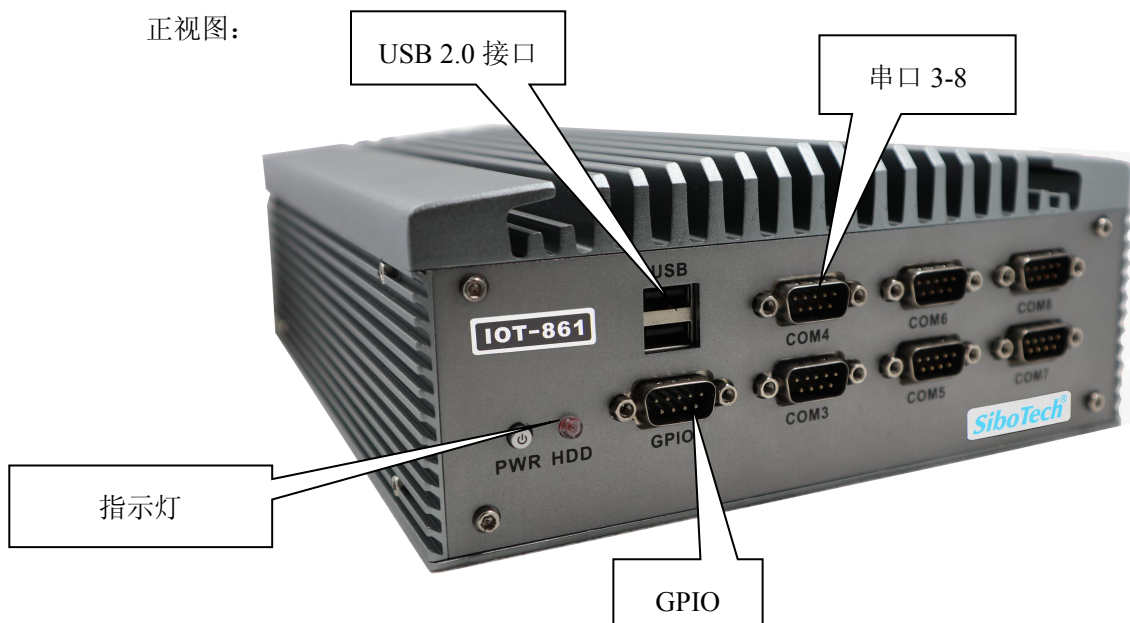
### 2.4.4 辐射电磁场( GB/T 17626.3 classIII )

电波频率	150MHz, 400MHz, 900MHz
试验场强	10 V/m
辐射方法	使得天线前端触碰装置,或接近端子,断续辐射电波

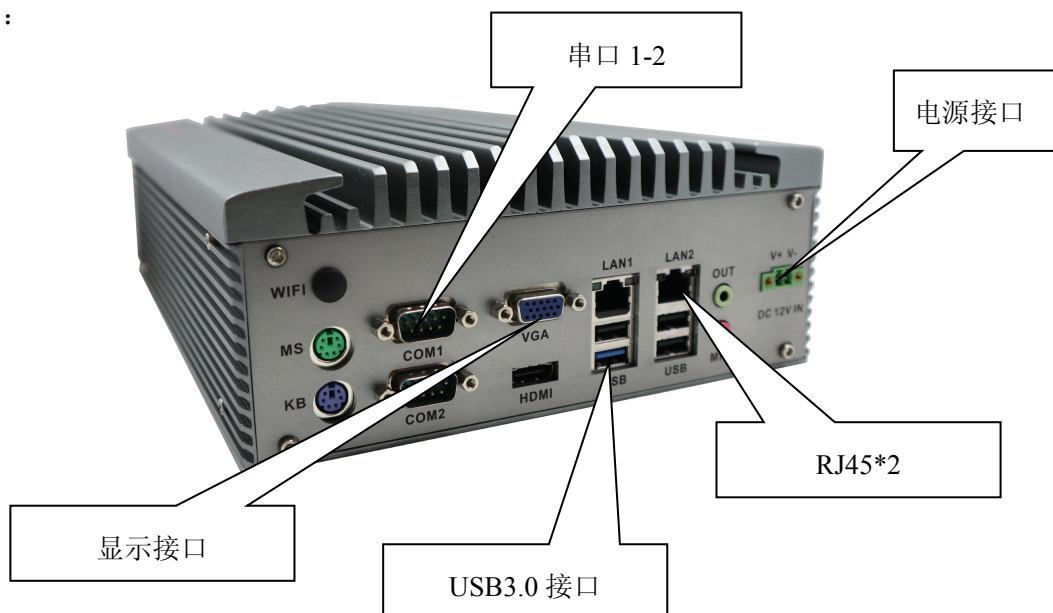
## 三、硬件说明

### 3.1 产品外观

正视图:



背视图:

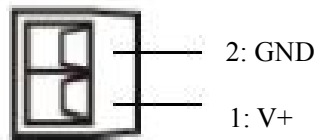


## 3.2 指示灯

Indicator	Status	Description
PWR	蓝色常亮	正在工作
	蓝色灭	已停止工作
HDD	红色闪烁	数据交换中
	红色灭	无数据交换

## 3.3 接口

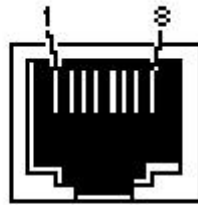
### 3.3.1 电源接口



Pin	Function
1	V+
2	GND



### 3.3.2 以太网接口



RJ-45 port

以太网接口采用 RJ-45 插座，其引脚定义（标准以太网信号）如下：

引脚	信号说明
S1	TXD+, Tranceive Data+, 输出
S2	TXD-, Tranceive Data-, 输出
S3	RXD+, Receive Data+, 输入
S4	Bi-directional Data+
S5	Bi-directional Data-
S6	RXD-, Receive Data-, 输入
S7	Bi-directional Data+
S8	Bi-directional Data-

### 3.3.3 RS-485/RS-422 接口

IOT-861产品的485接口是标准的RS-485接口，以下简述本产品RS-485特性：

➤ RS-485 传输技术基本特征

- ① 网络拓扑：线性总线，两端有有源的总线终端电阻；
- ② 传输速率：1200 bit/s~115.2Kbit/s；
- ③ 介质：屏蔽双绞电缆，也可取消屏蔽，取决于环境条件（EMC）；
- ④ 站点数：每分段 32 个站（不带中继），可多到 127 个站（带中继）；

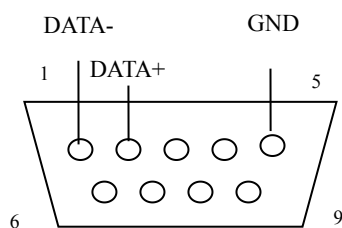
⑤ 插头连接：5 针可插拔端子

➤ RS-485 传输设备安装要点

① 全部设备均与 RS-485 总线连接；

② 总线的最远两端各有一个总线终端电阻，120Ω 1/2W 确保网络可靠运行。

串行接口采用标准的硬串口形式，用户可根据需要进行接线。

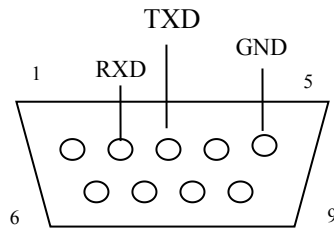


串口侧的接线规则：

引脚	RS-422	RS-485	描述
1	TX-	DATA-	RS-485 数据负/RS-422 发送负
2	TX+	DATA+	RS-485 数据正/RS-422 发送正
3	RX+	NC	RS-422 接收正
4	RX-	NC	RX-422 接收负
5	GND	GND	GND，地
6	NC	NC	NC
7	NC	NC	NC
8	NC	NC	NC
9	NC	NC	NC

### 3.3.4 RS-232 接口

IOT-861 的 RS-232 接口采用标准硬串口形式，用户可根据需要进行接线。



引脚	名字	描述
2	RXD	串口接收，接用户设备的串口发送
3	TXD	串口发送，接用户设备的串口接收
5	GND	GND，地

### 3.4 看门狗

本产品内部支持看门狗复位，如产品遇到故障宕机，系统可以实现自动复位。

## 四、快速应用指南

以下几个步骤可以快速应用您的 IOT-861:

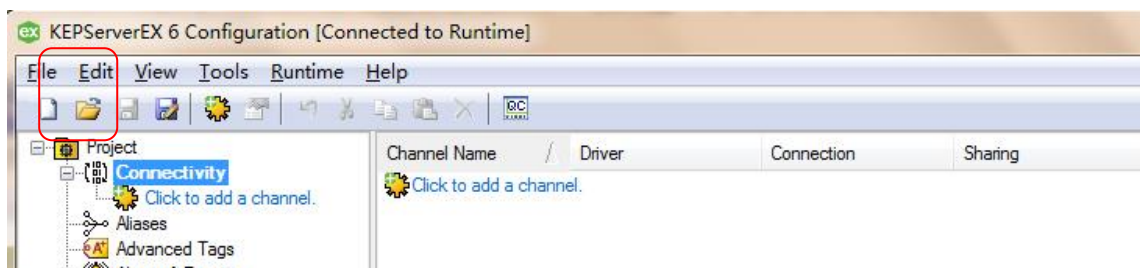
1. 连接电源: 使用适配器将产品连接至 220V 交流电, 点击电源按钮, 即可点亮产品。
2. 简单配置 IOT-861:

### Step1: 获取数据源

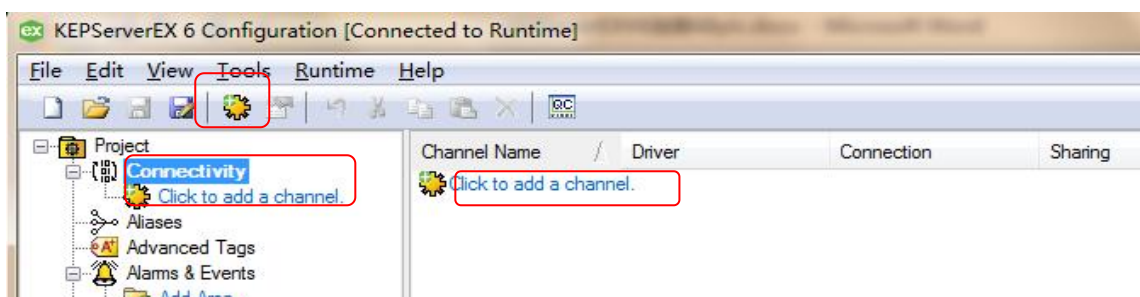
以西门子 S7-300 PLC 为例, 介绍如何连接 PLC 进行 OPC 通信以及如何把数据上传到云。

### ➤ 新建通道: New Channel

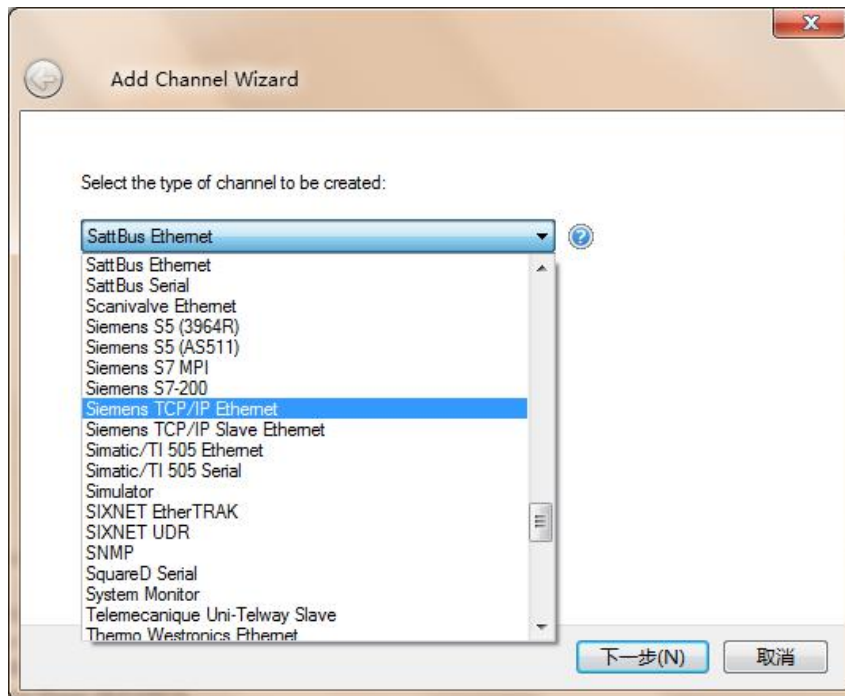
1. 打开“KEPServerEX V6”软件, 点击“File”->“New”, 或者工具栏上的“New Project”, 建立一个新的工程。



2. 单击“Click to add a channel”或者工具栏上的“New channel”, 建立一个新通道。

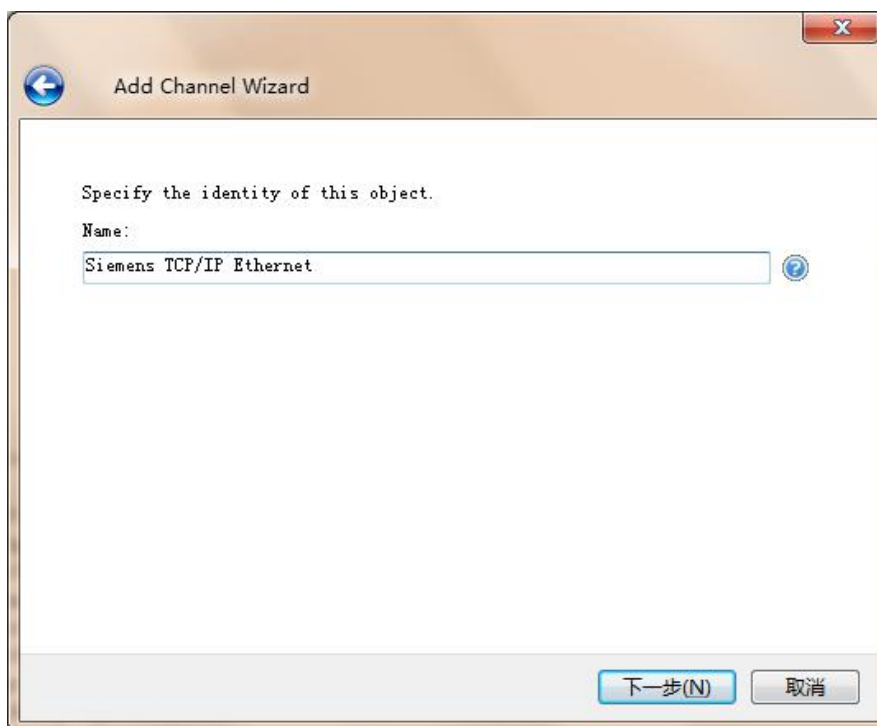


3. 选择需要分配给本通道的设备驱动“Device driver”, 在下拉选项中选择“Siemens TCP/IP Ethernet”, 单击“下一步”。

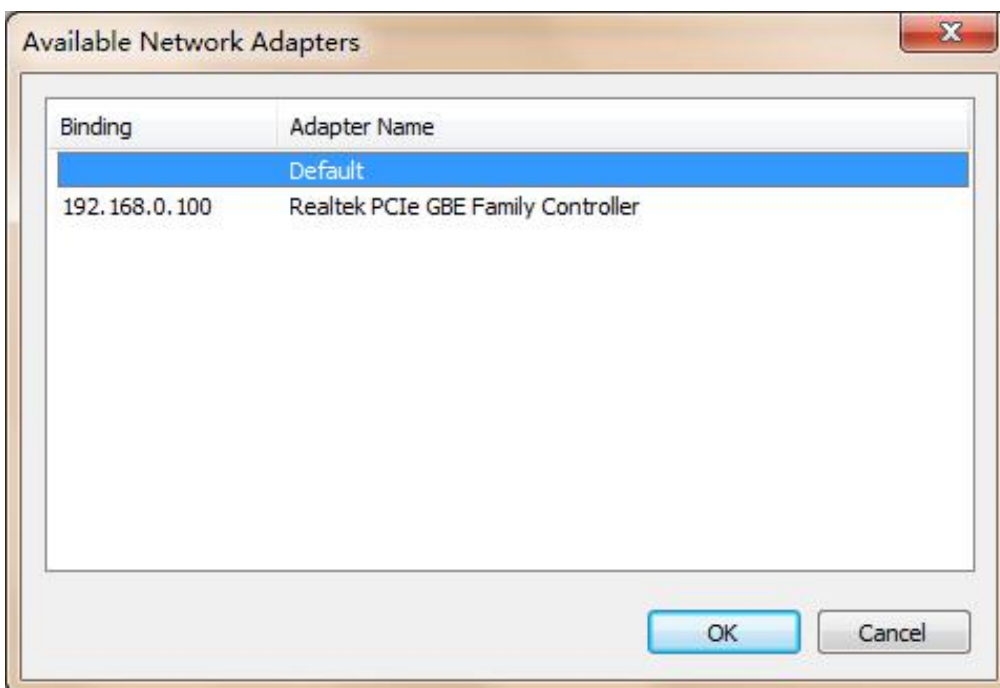
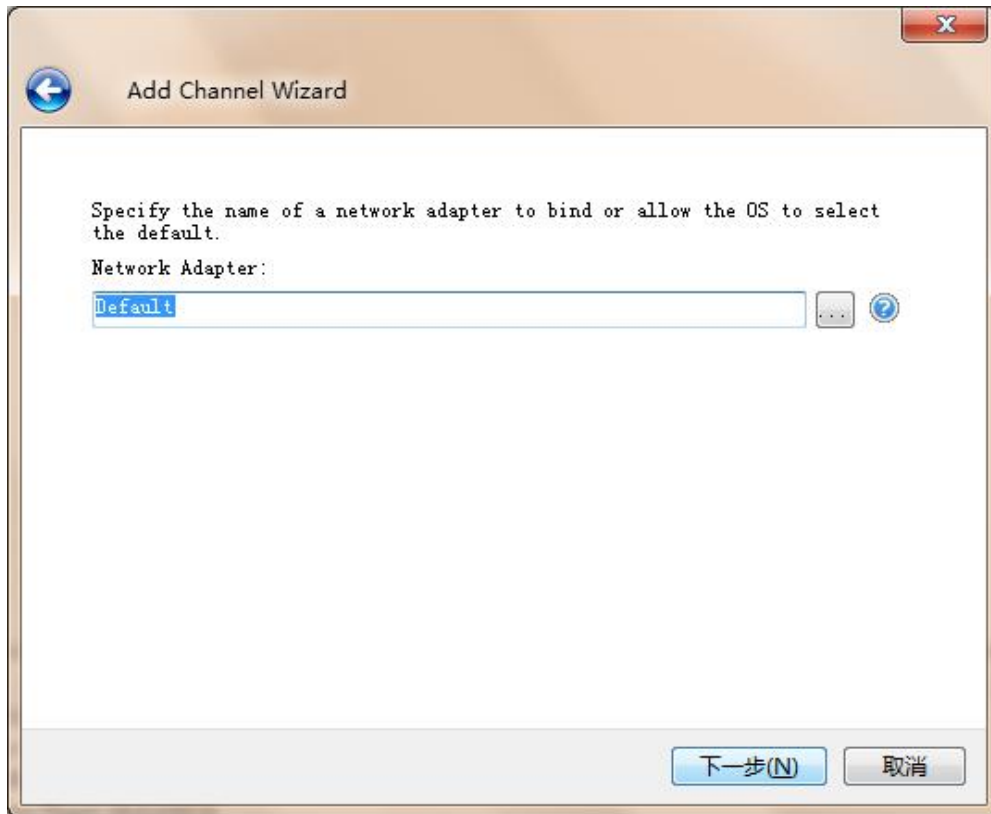


**注意：**如果用户使用过程中在下拉选项中没有找到想要的驱动，可能安装的时候没有完全安装，此时需要对 KEPServerEX V6 进行修改安装，在安装时安装所需要的驱动。

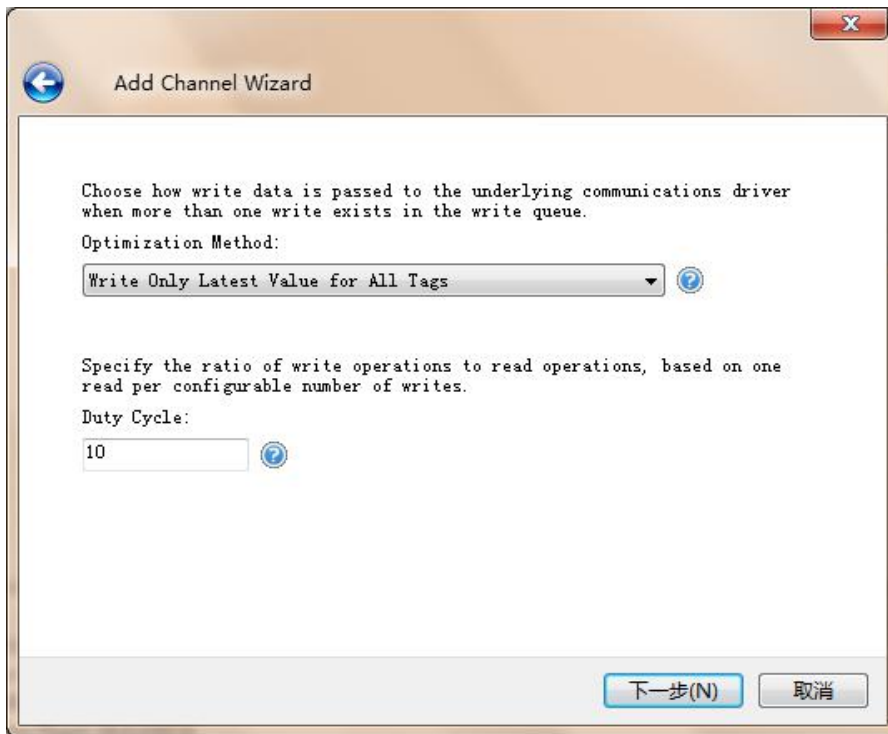
4. 修改通道名称“Channel name”，修改为想要的名称，可不作修改。如下修改为“Siemens TCP/IP Ethernet”，单击“下一步”。



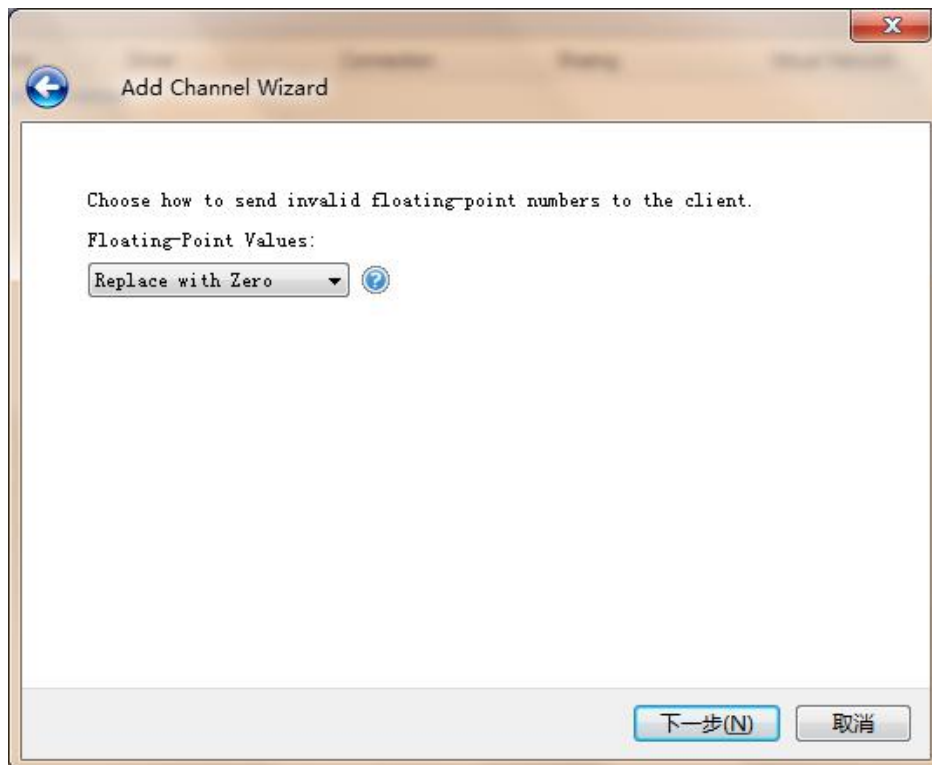
5. 选择设备所用的网络适配器“Network Adapter”。这里选择实际需要使用的网络适配器，点击“...”弹出选择窗口，选择所用网络适配器，点击“OK”之后单击“下一步”。



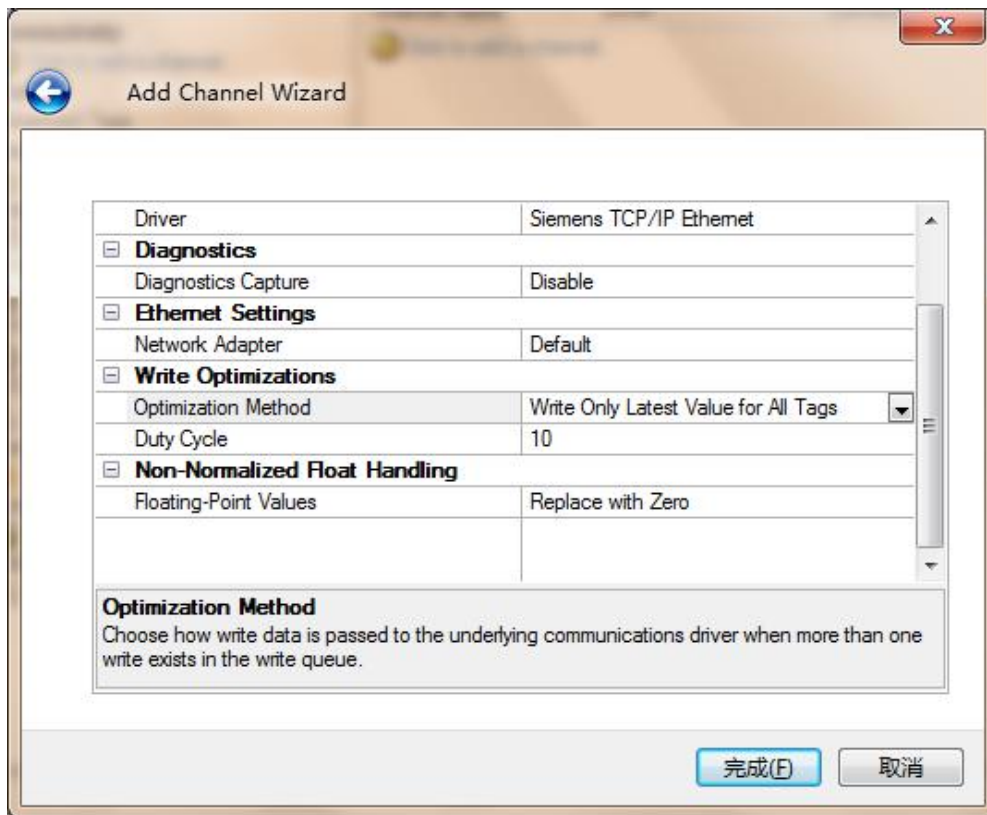
6. 对通道进行优化设置“Optimization Method”，这里保持默认，单击“下一步”。



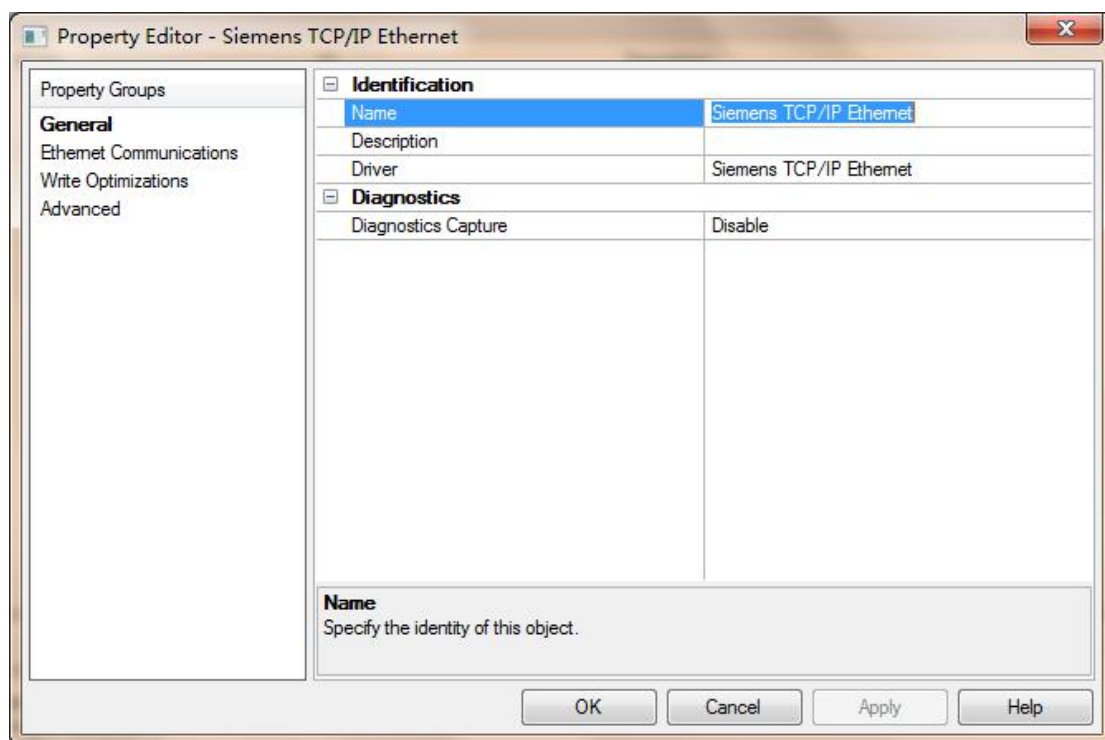
7. 设置浮点型标准“Floating-Point Values”，保持默认值不变，单击“下一步”。



8. 设置总结，单击“完成”，在这里可以对以上所做的设置重新进行修改。



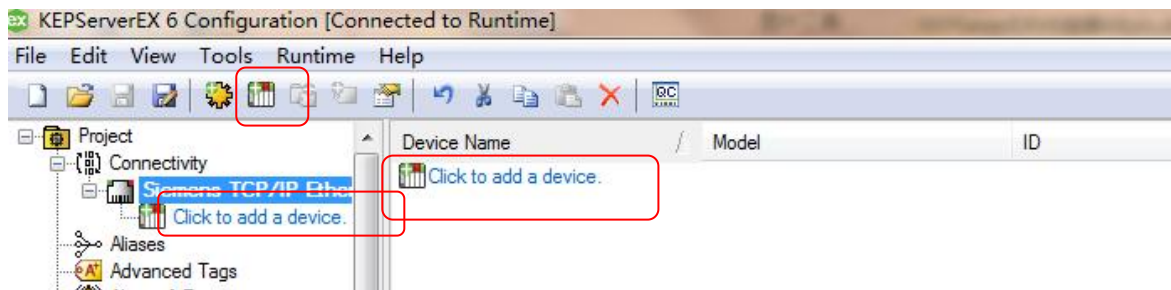
通过右键点击通道名选择“Properties”，或者双击通道名称修改通道参数。



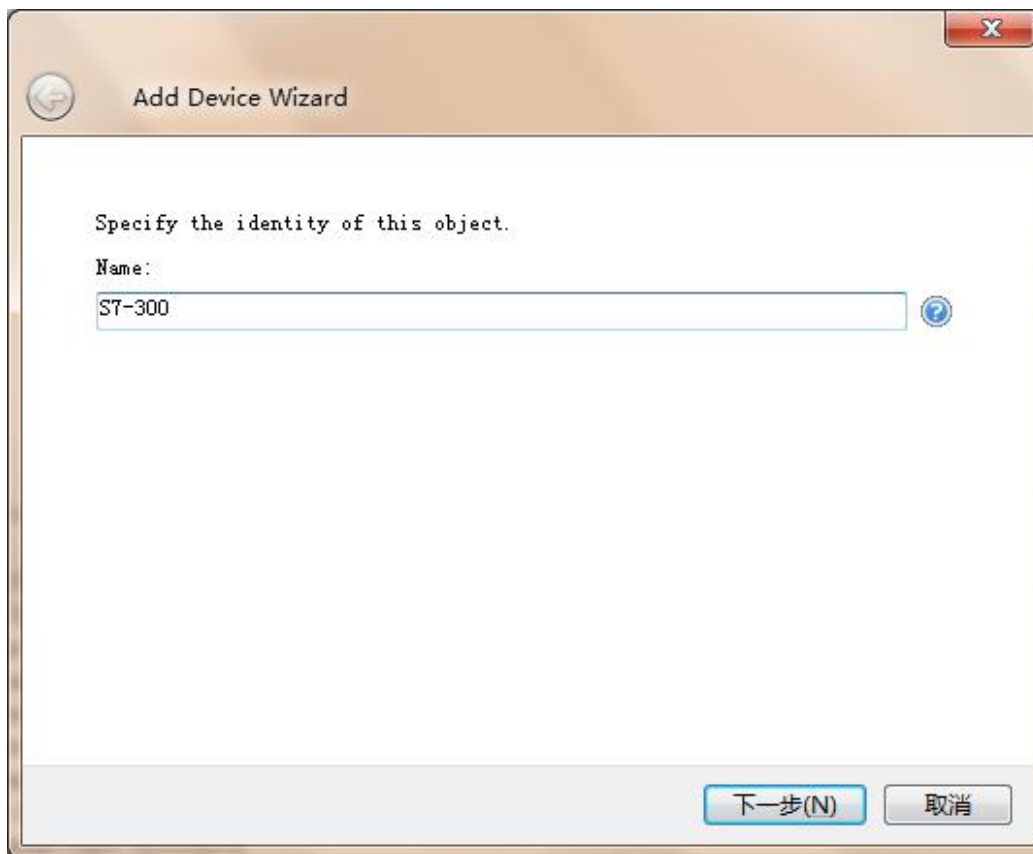


## ➤ 新建设备：New Device

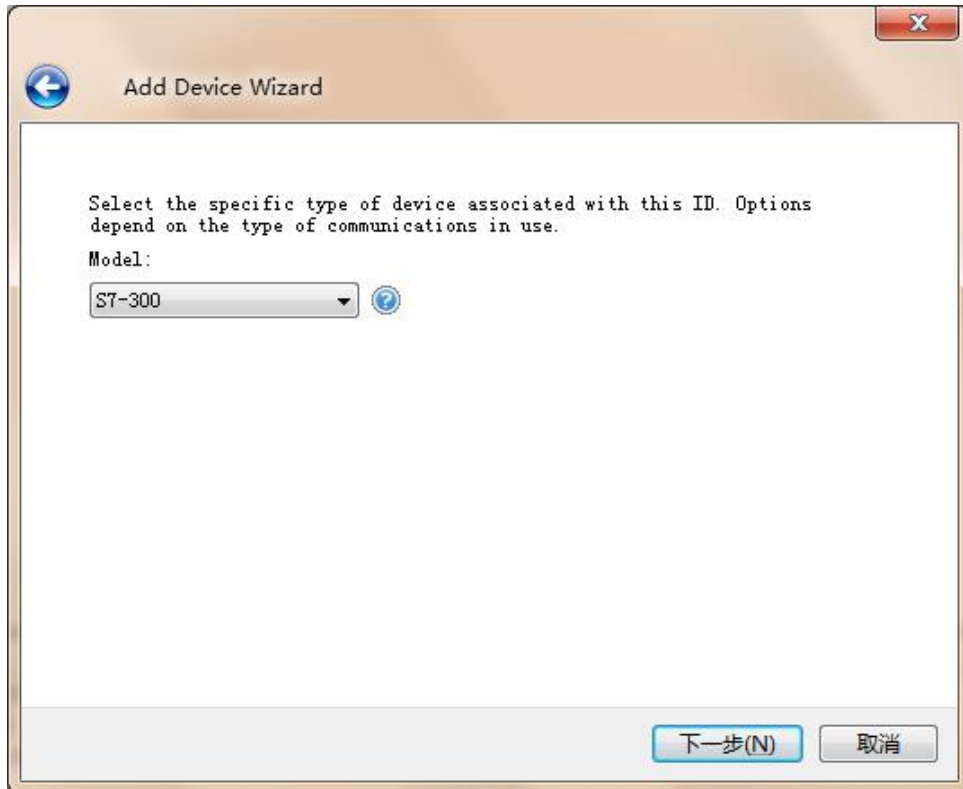
1. 单击软件界面中的“Click to add a device”或者工具栏上的“New Device”，进行设备设置。



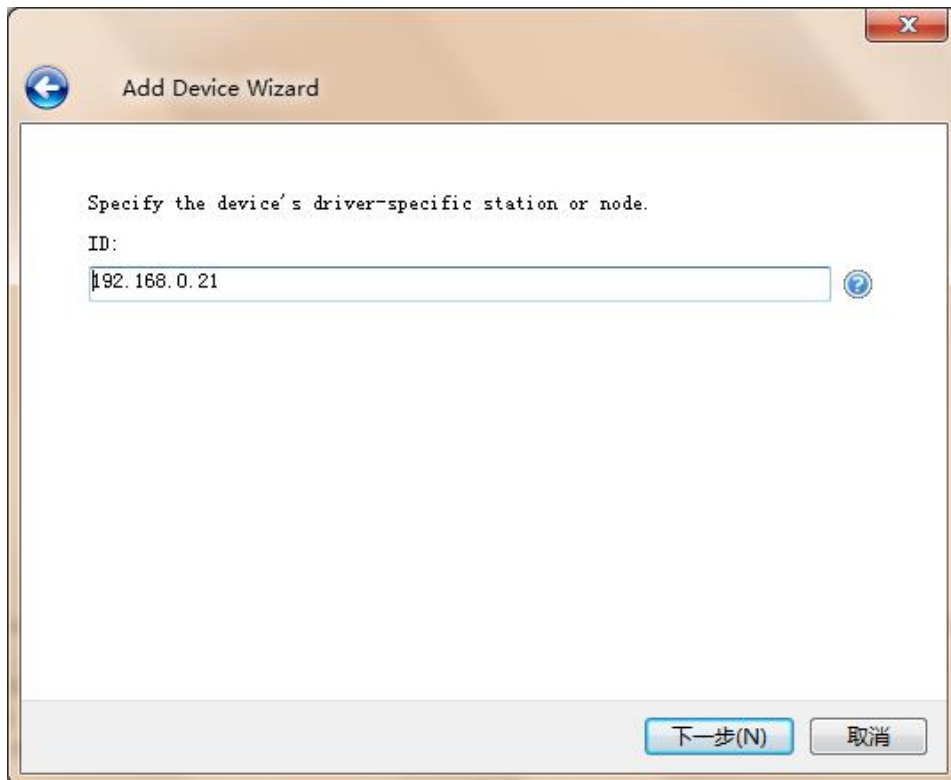
2. 修改设备名称“Name”，修改为需要的名称，这里我们改为“S7-300”，然后单击“下一步”。



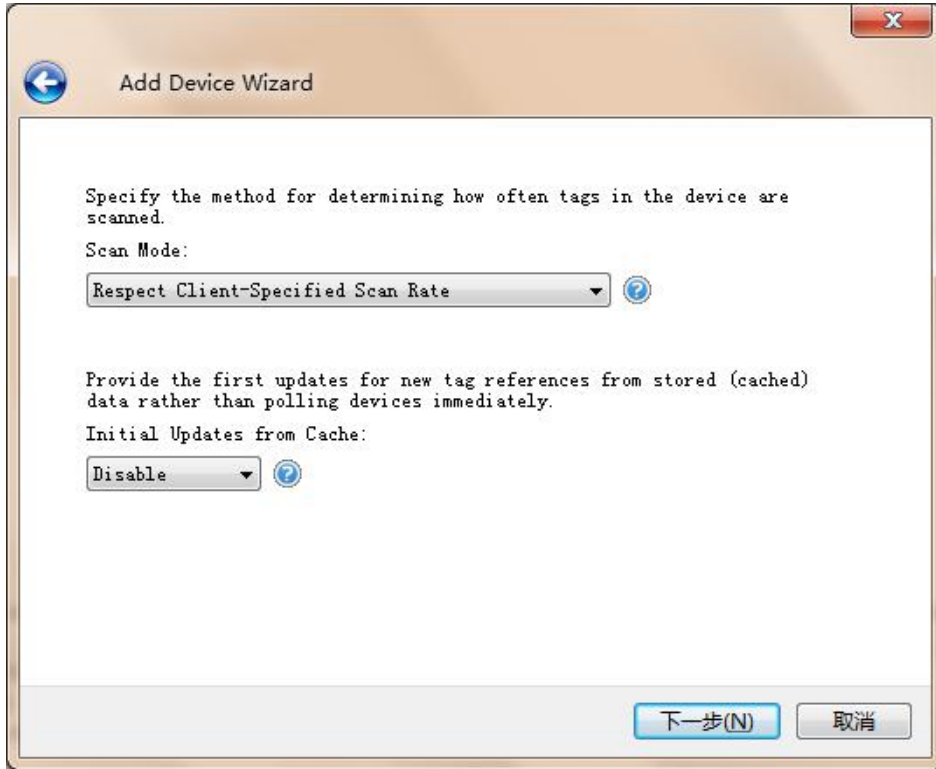
3. 选择设备模型“Model”，这里我们选择“S7-300”，单击“下一步”。



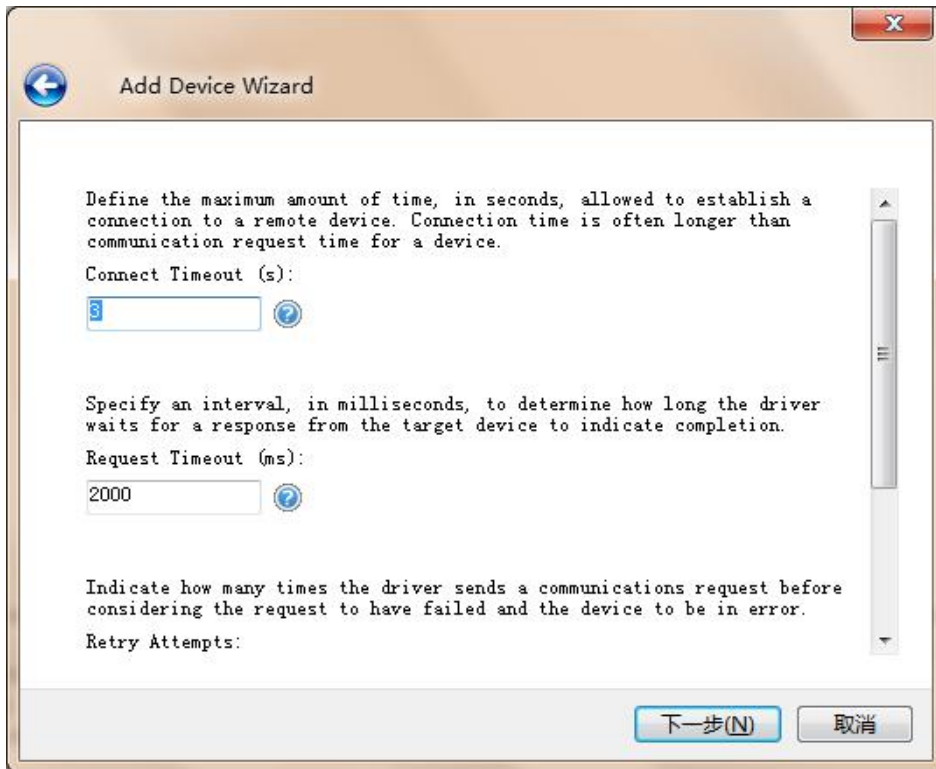
4. 输入设备“ID”，指的是所需要连接的 PLC 的 IP 地址，如下我们填写“192.168.0.21”。



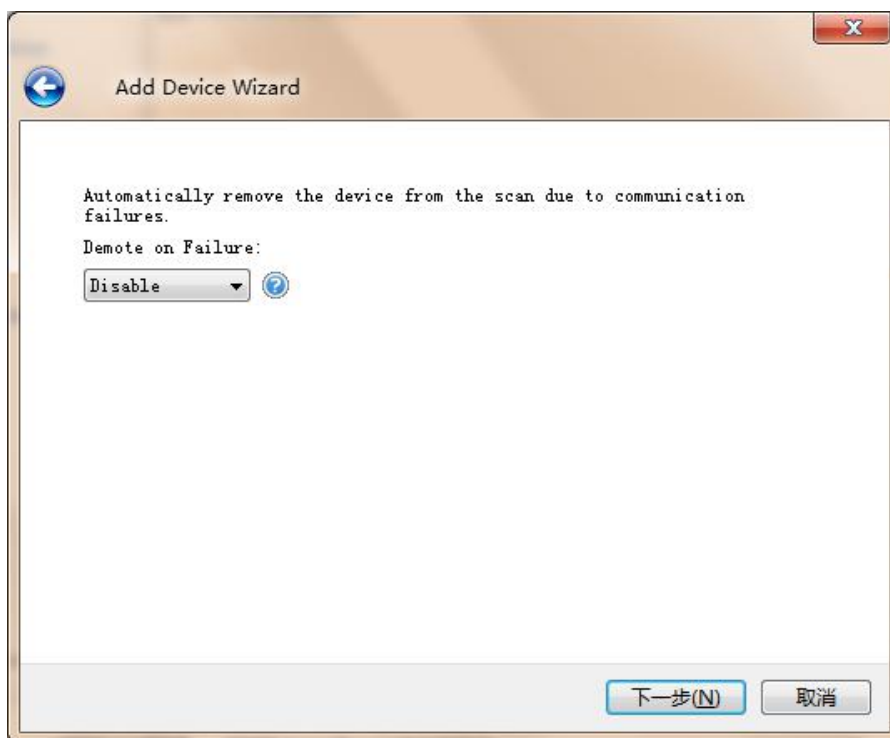
5. 选择扫描方式“Scan Mode”，保持默认，单击“下一步”。



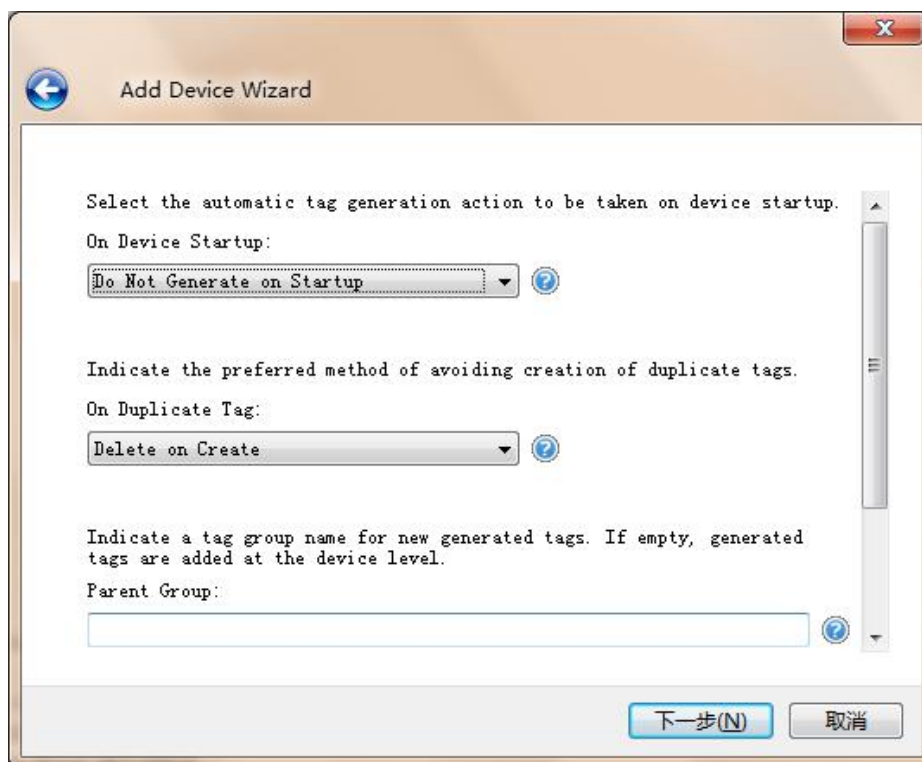
6. 设置通信的时间参数“Timing”，保持默认值不变，单击“下一步”。



7. 设置失败降级 “Demote on Failure”，保持默认设置不变，单击“下一步”。

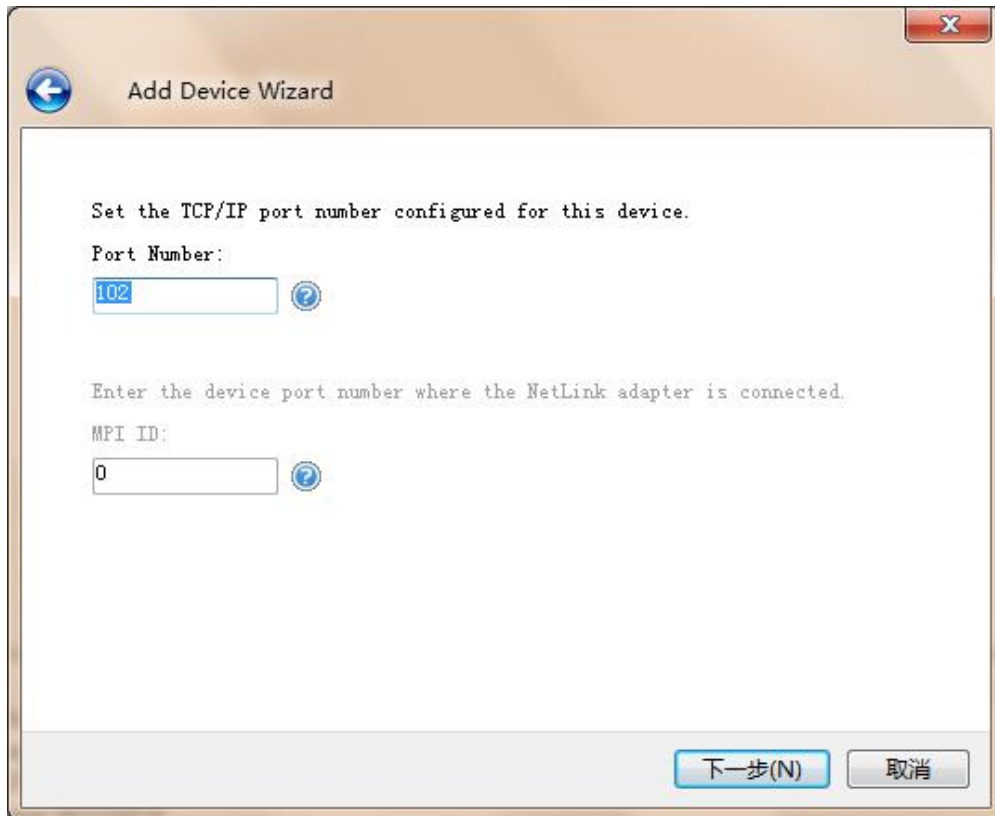


8. 自动生成标签设置，保持默认设置，单击“下一步”。



注意：若从 step7 工程文件中自动生成 tag，需选择“always generate on start up”。目前支持 TIA Portal 和 step7 工程文件的上传。

9. 通信参数设置，我们保持默认值不变，单击“下一步”。

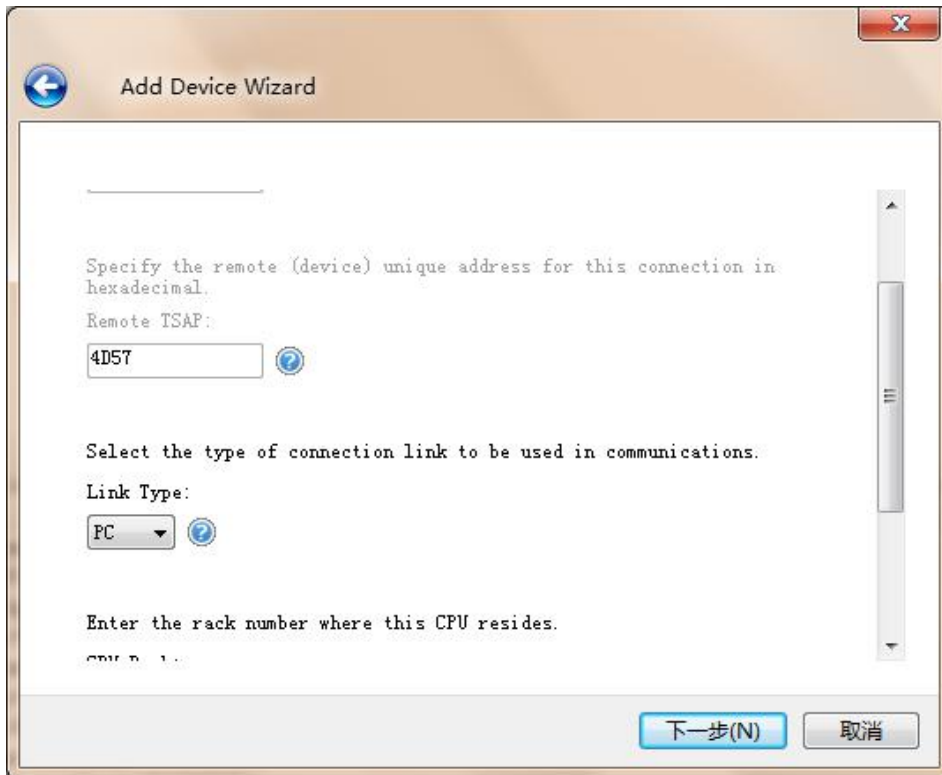


10. 设置 S7 通讯参数，这里我们保持默认值不变，单击“下一步”。

其中：“Link Type”：连接类型，一般默认选择“PC”。

“CPU Rack”：CPU 所在机架的位置，要与 STEP7 中设置的位置一致。

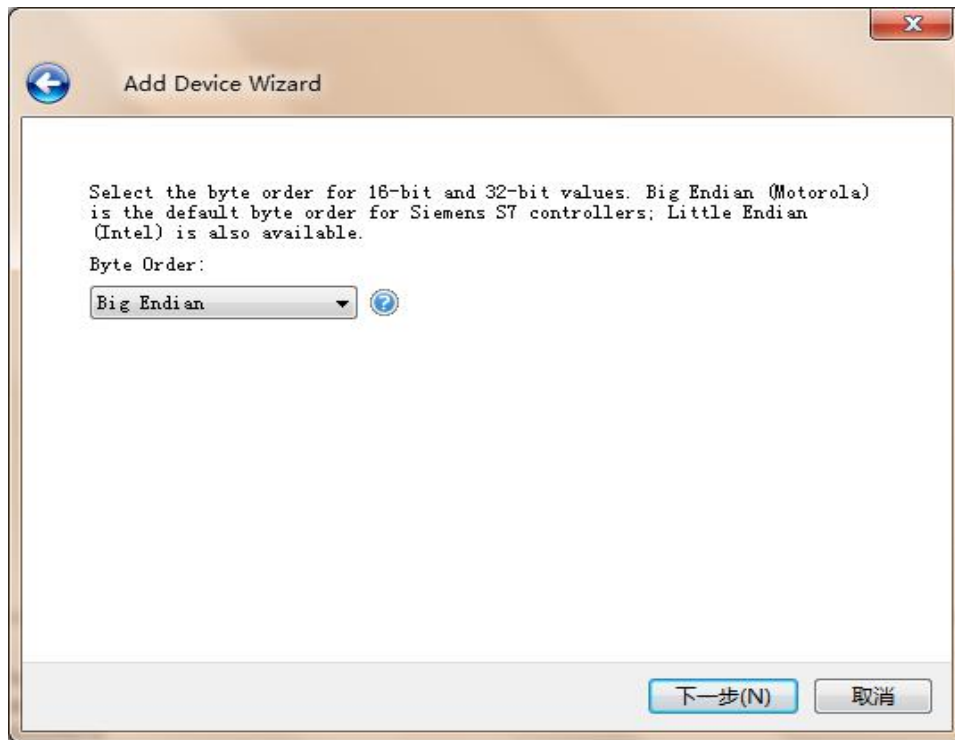
“CPU Slot”：实际连接的 PLC 的 CPU 所在的槽位，要与 STEP7 中设置的位置一致。



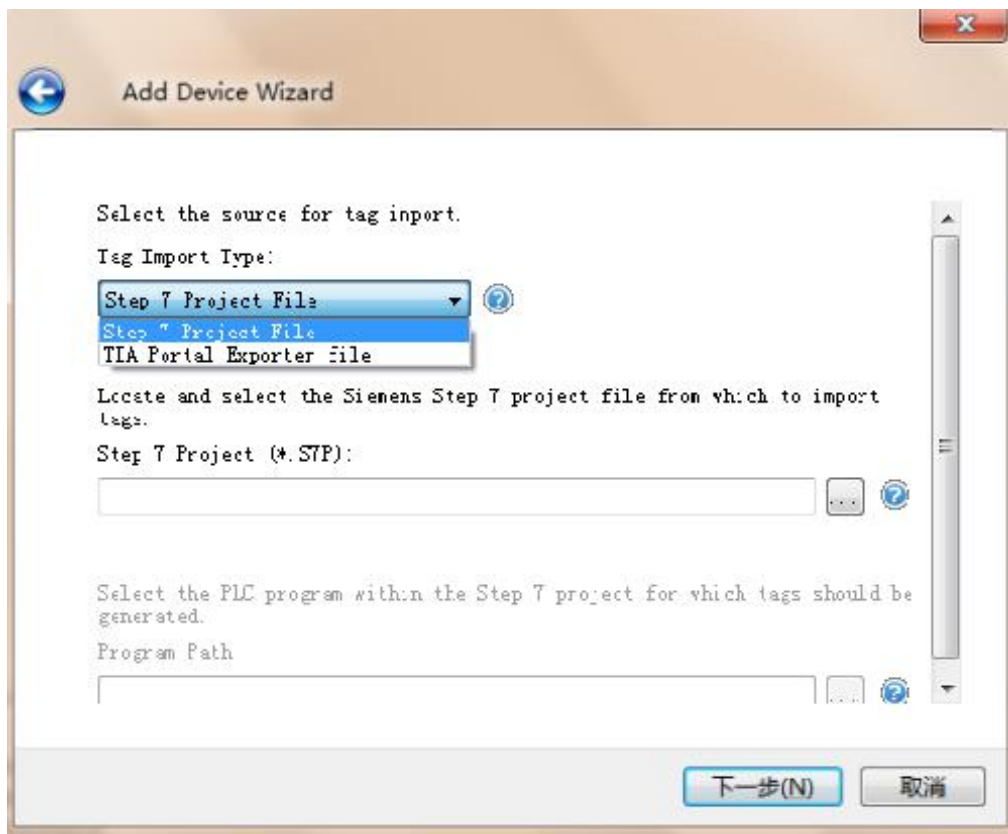
11. 设置字节顺序，我们保持默认不变，单击“下一步”。

其中，Big Endian: 大端模式

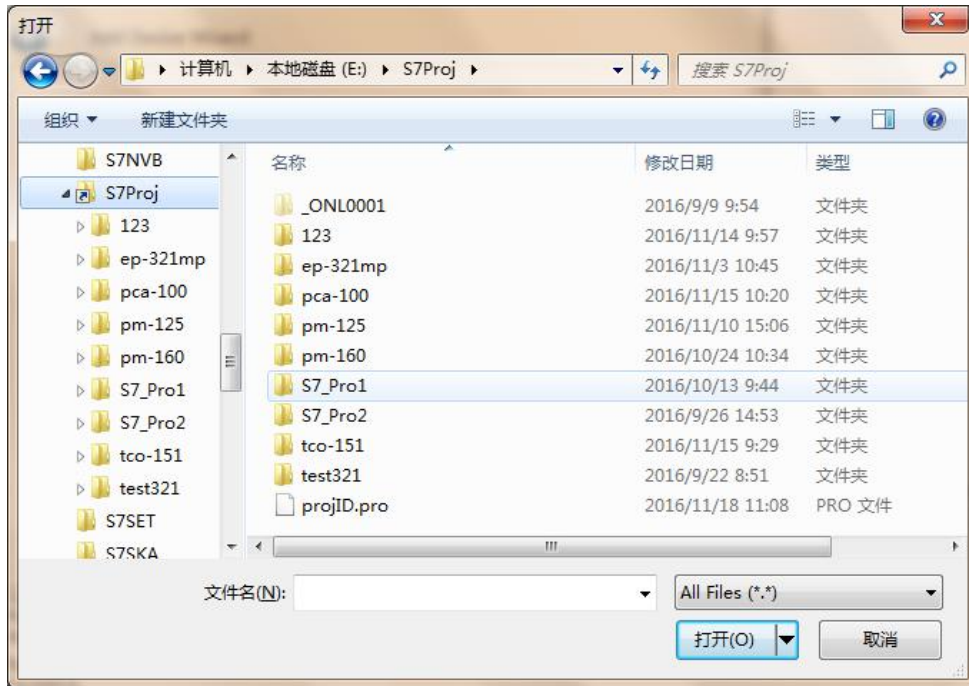
Little Endian: 小端模式



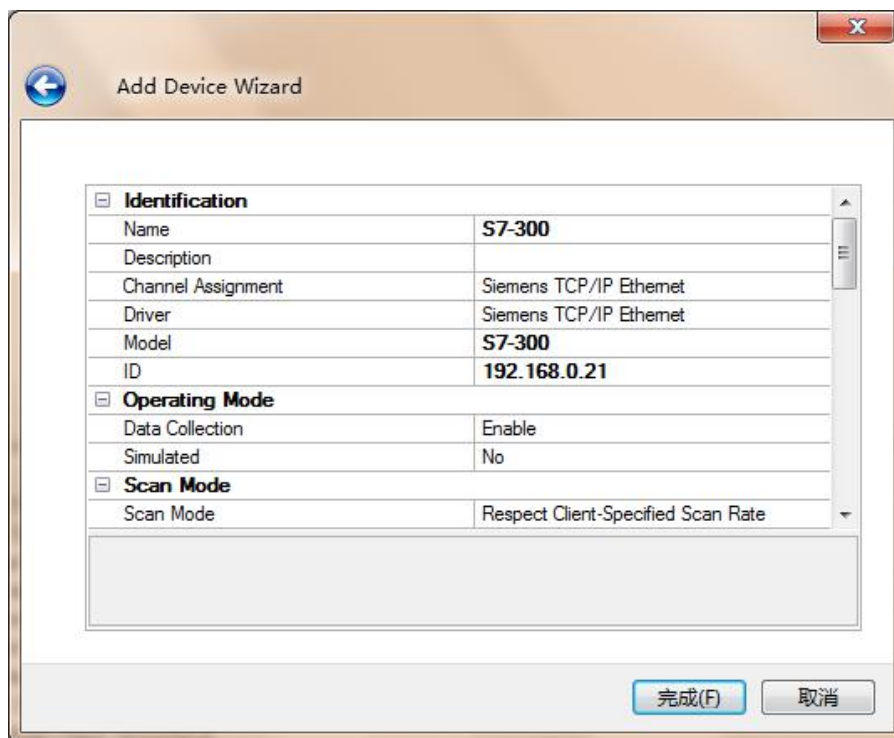
12. 选择添加工程文件，可以不做选择，这里我们直接单击“下一步”。



注意：KEPServerEX 支持从 step7 工程文件和 TIAportal 生成的“.TPE”文件中自动生成 tag，在 Tag Import Type 下拉菜单中选择相应的导入标签类型，然后选择相应的文件路径，并在 tag generation 中选择“always generate on start up”。

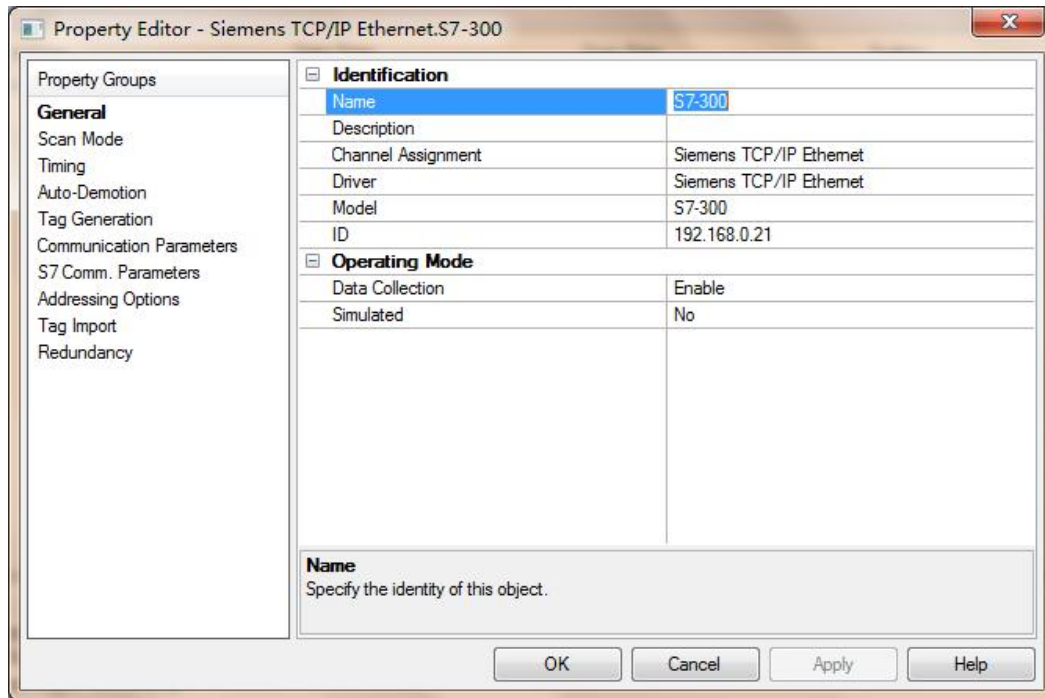


13. 设置总结，单击“完成”。在这里可以对之前的设置进行修改。



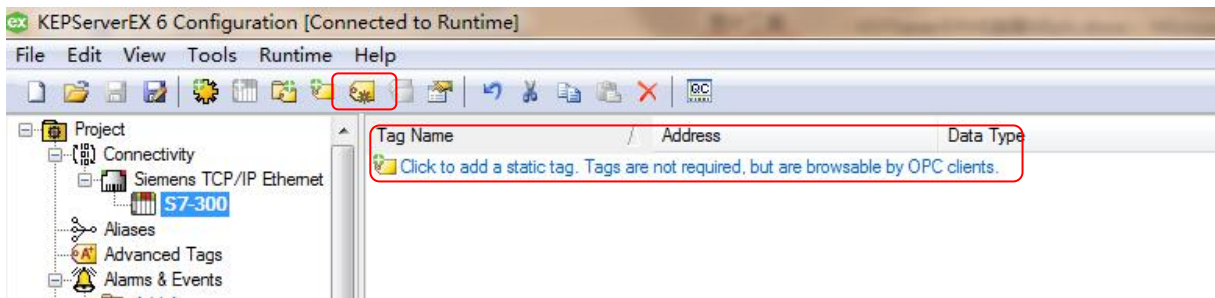


用户可以右键单击设备名称选择“Properties”或者双击设备名称来修改设备参数。

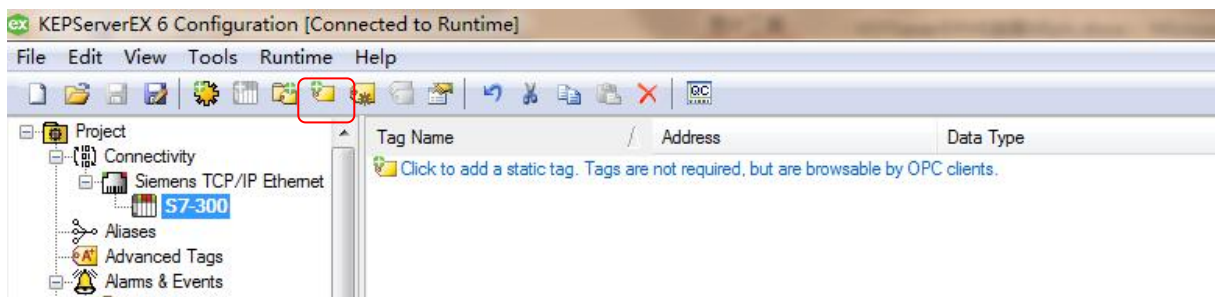


## ➤ 新建标签：New Tag

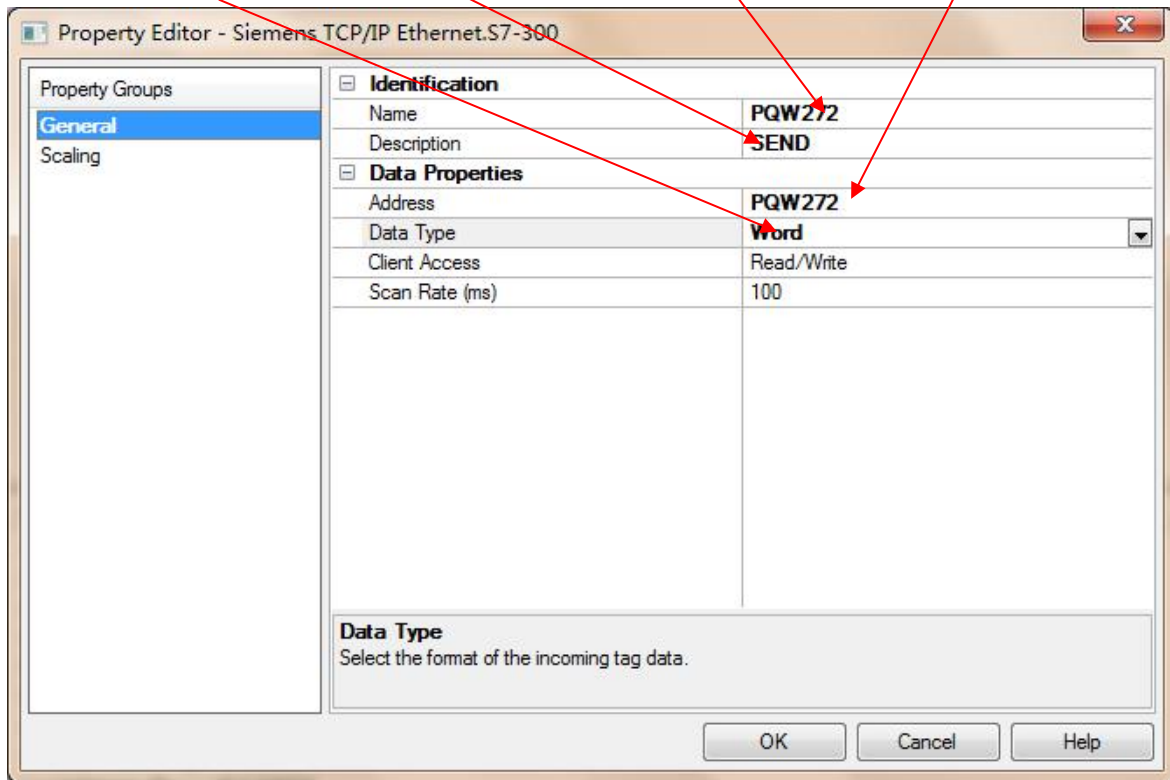
1. 单击软件界面中的“Click to add a static tag”，或者工具栏中的“New Tag”新增标签。



如果有需要建立标签组，也可以在新建标签之前先建立 New Tag Group，单击工具栏“New Tag Group”。

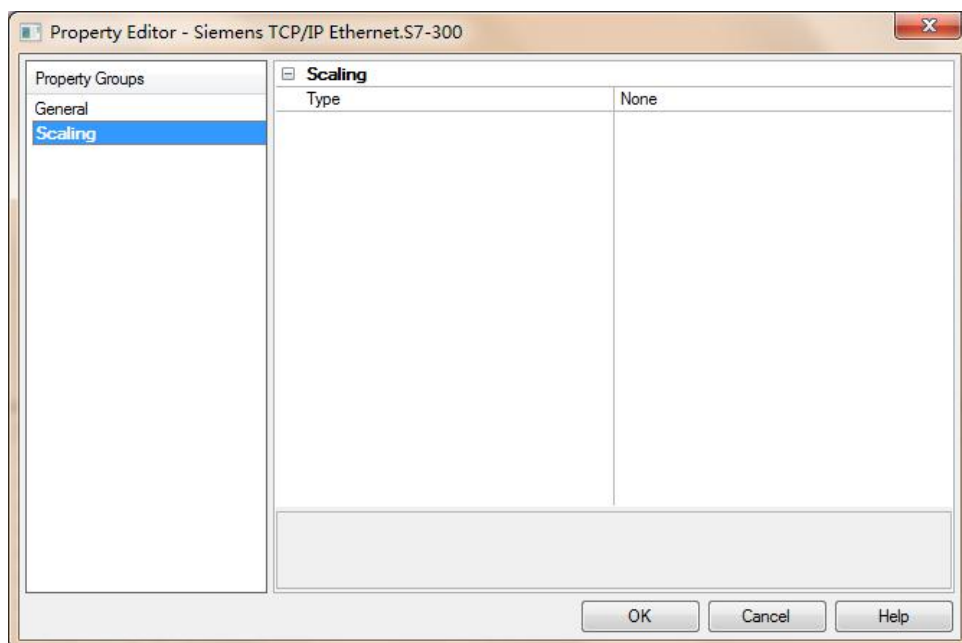


2. 设置 Tag 属性，这里我们设置 Name（名称）为“PQW272”，Address（地址）“PQW272”，Data Type（数据类型）：“Word”，Description（描述）：“SEND”，其他保持默认不变，单击“OK”。

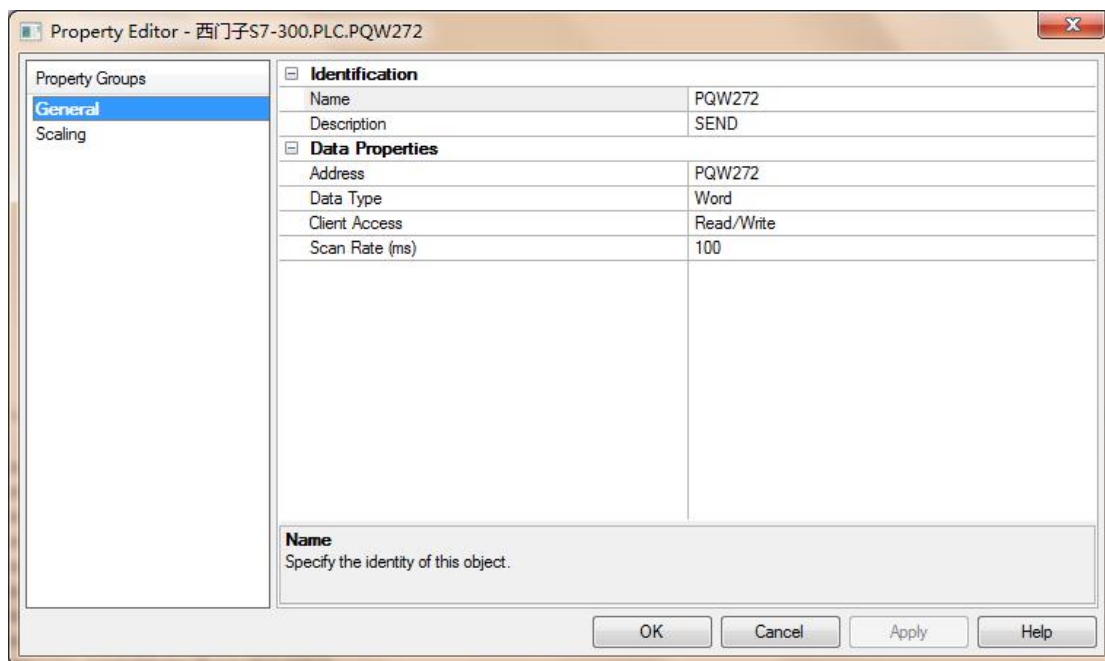


注意：填写 Address（地址）时，必须确保填写的内容是硬件设备内部允许访问的地址。

Scaling 设置：这里我们保持默认值。



- 用户可以通过选中标签右键单击选择“Properties”，或者双击 Tag 名称查看修改标签参数。至此 KEPServerEX 服务器端设置完成。



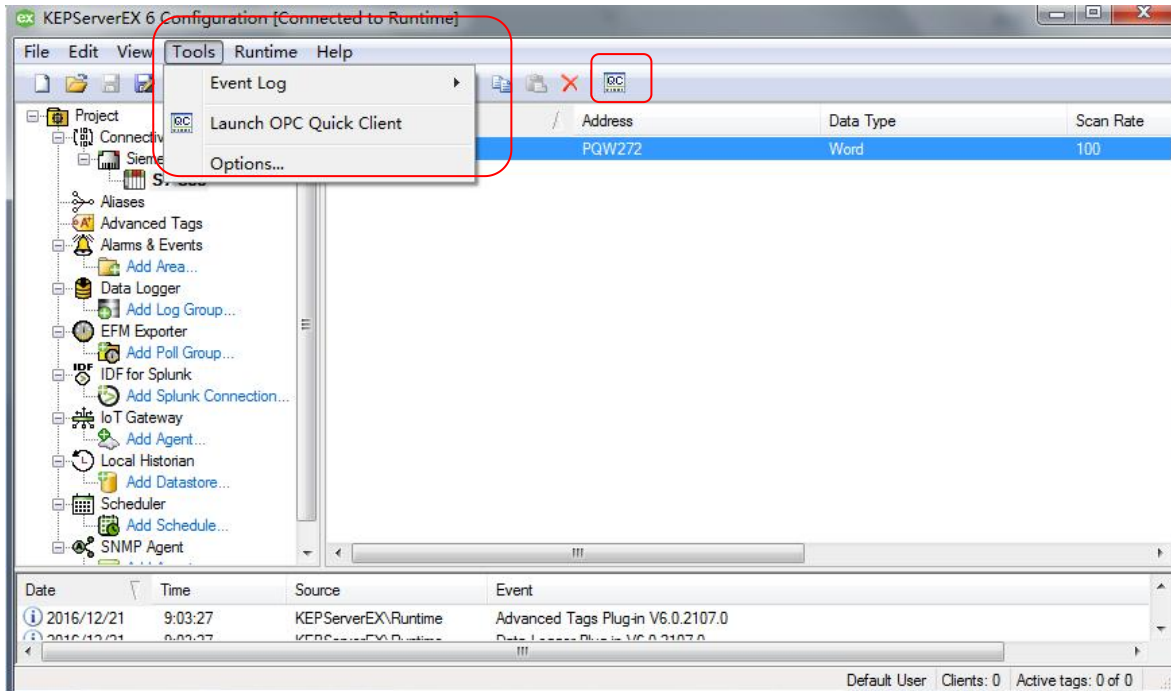
对设置完成的 KEPServerEX 服务器进行测试

- 首先打开西门子 S7-300 PLC，用 STEP7 对 PLC 进行编程，下载程序，必须确保程序中有 Pqw272，这个变量就是 PLC 的内部寄存器，它和上图中的 Address 所赋的值必须一样，否则 OPC client 访问不到。具体请参考“在 STEP7 中设置 S7-300”。
- 点击已设置完成的 KEPServerEX 工具栏中的“QC”，或者点击界面中的“Tools→Launch OPC Quick Client”进行测试。

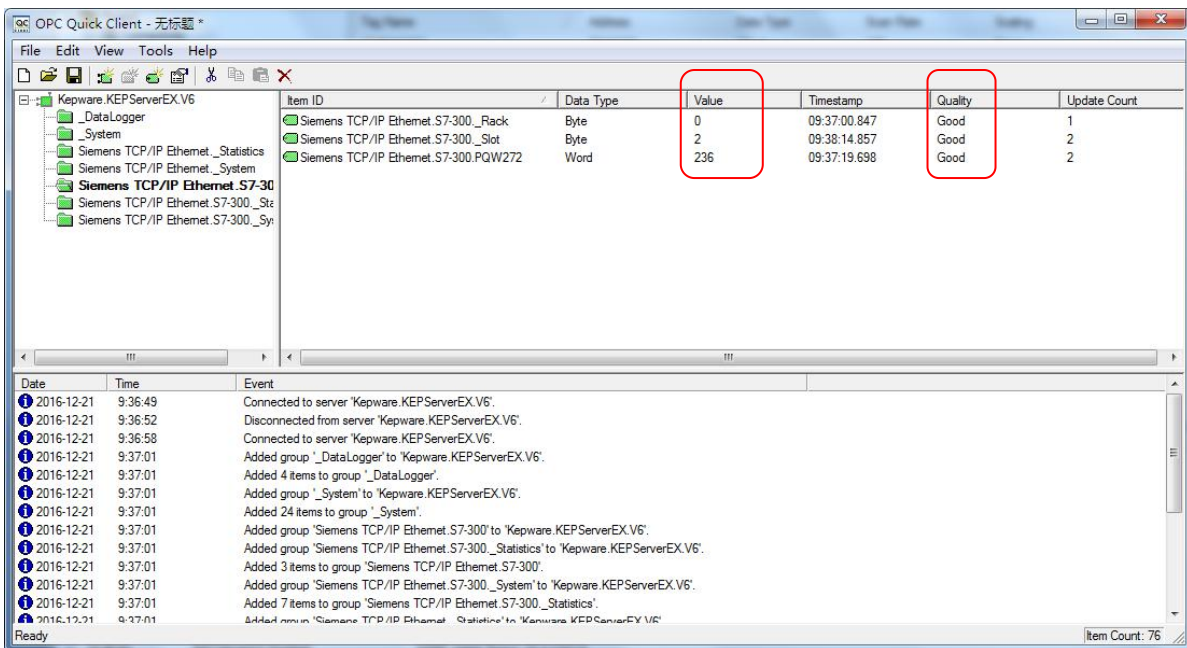
# IOT-861

## 工业物联网网关

### User Manual



3. 点击“QC”后出现以下界面：

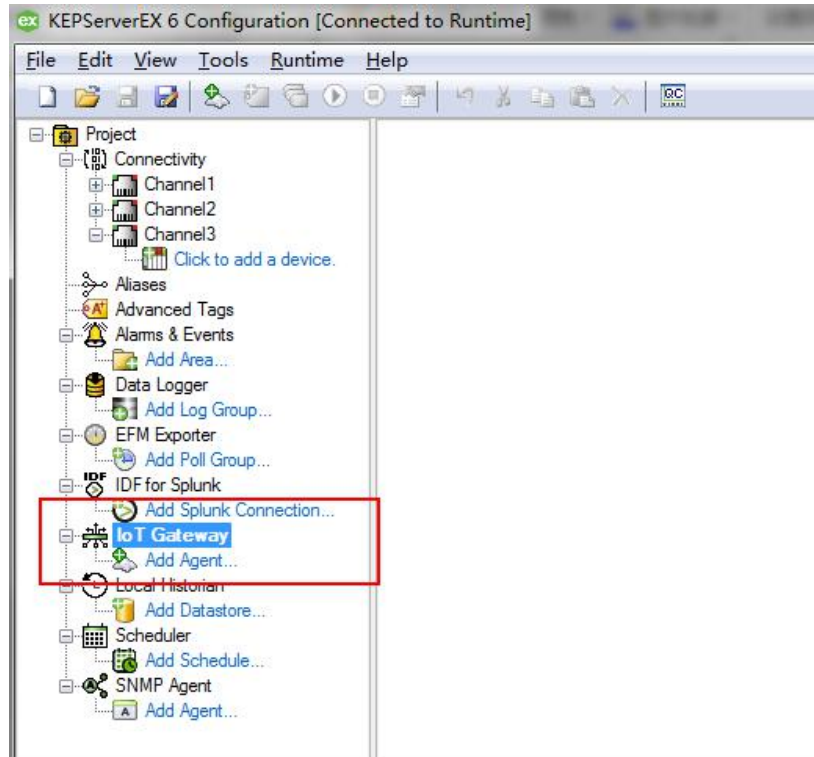


当图中的“Quality”变为“Good”时说明 OPC Server 已与西门子 S7-300 PLC 正确连接，读取到数据在“Value”中显示。

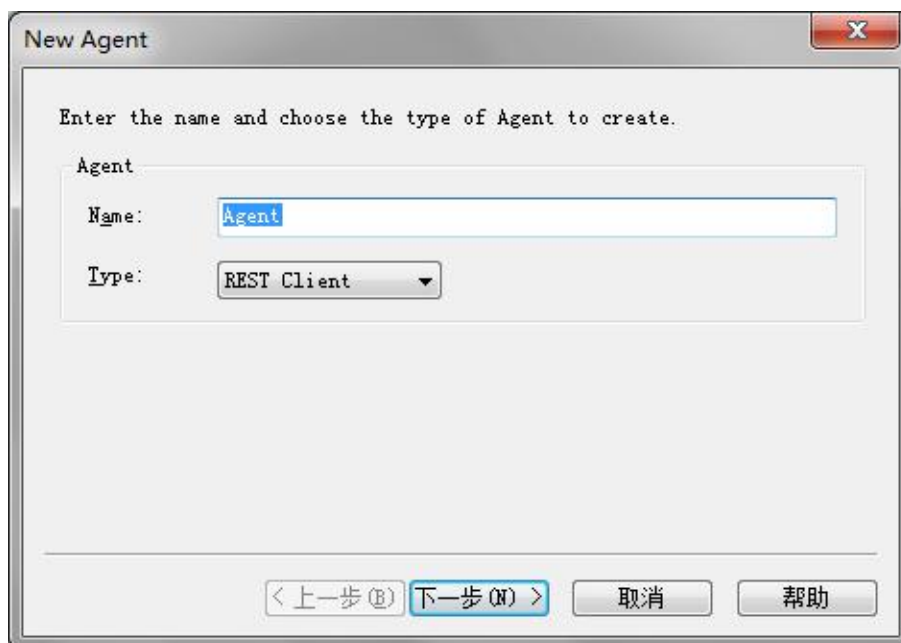
**Step2: 把数据推送到云**

本文使用 Node-RED 应用来验证云数据通信;

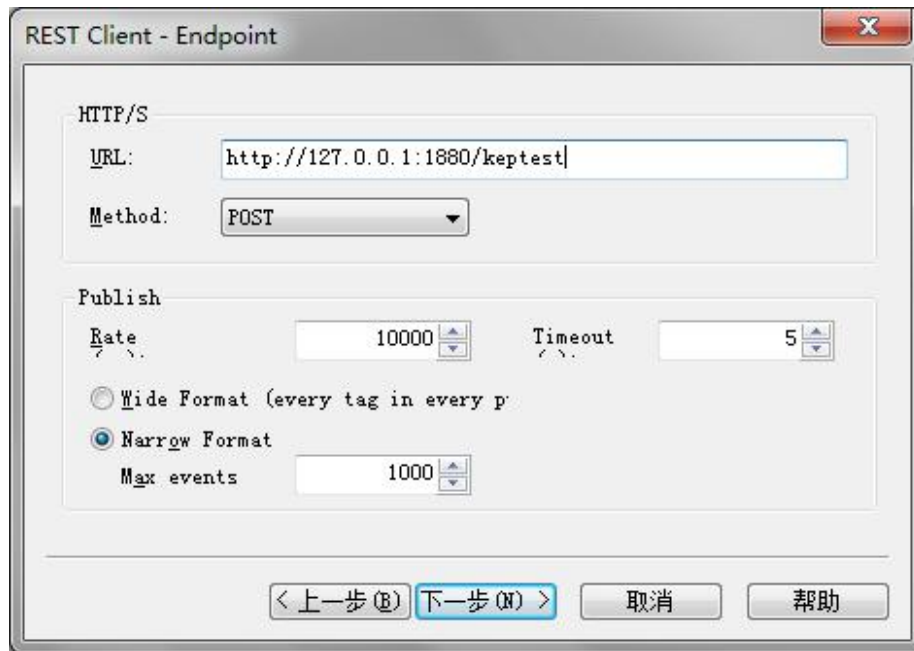
选中 IOT-Gateway 插件, 点击 “Add Agent...”



点击 “Click to add an agent”, 选择 REST Client

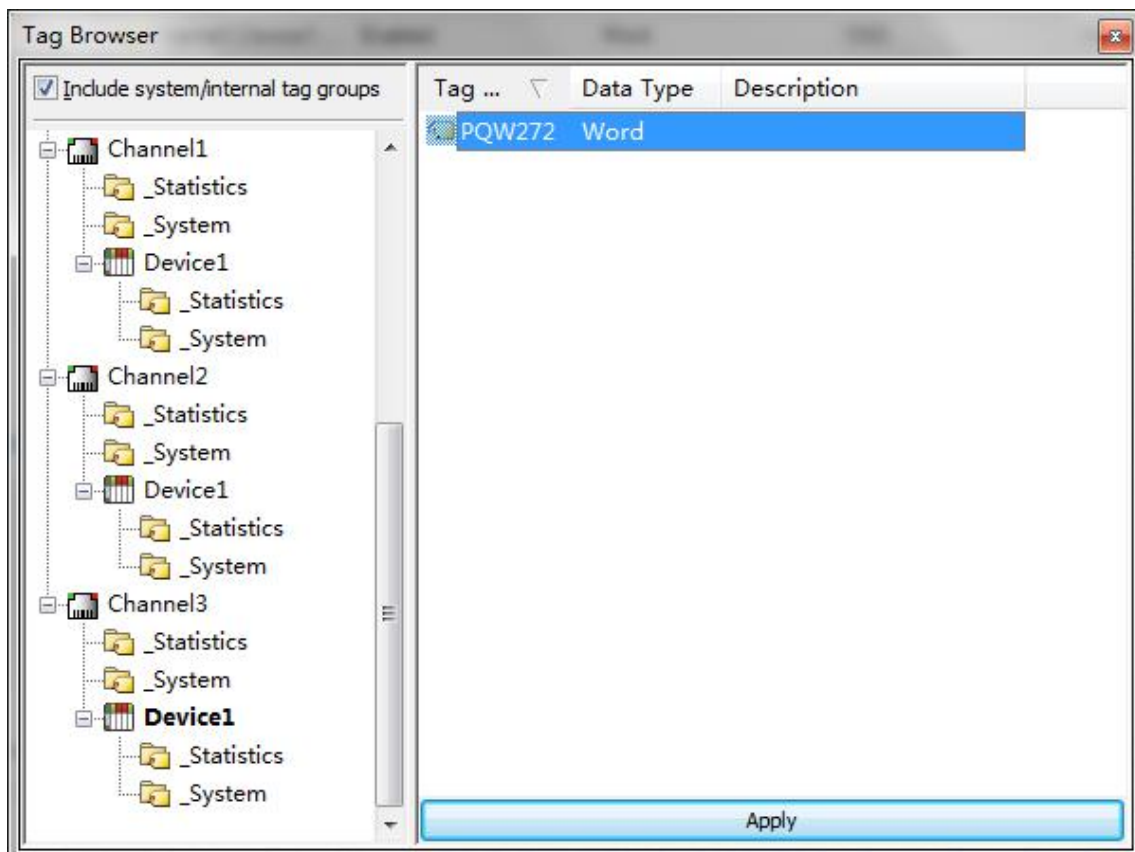


下一步，端口配置，改变默认端口号 3000 到 1880

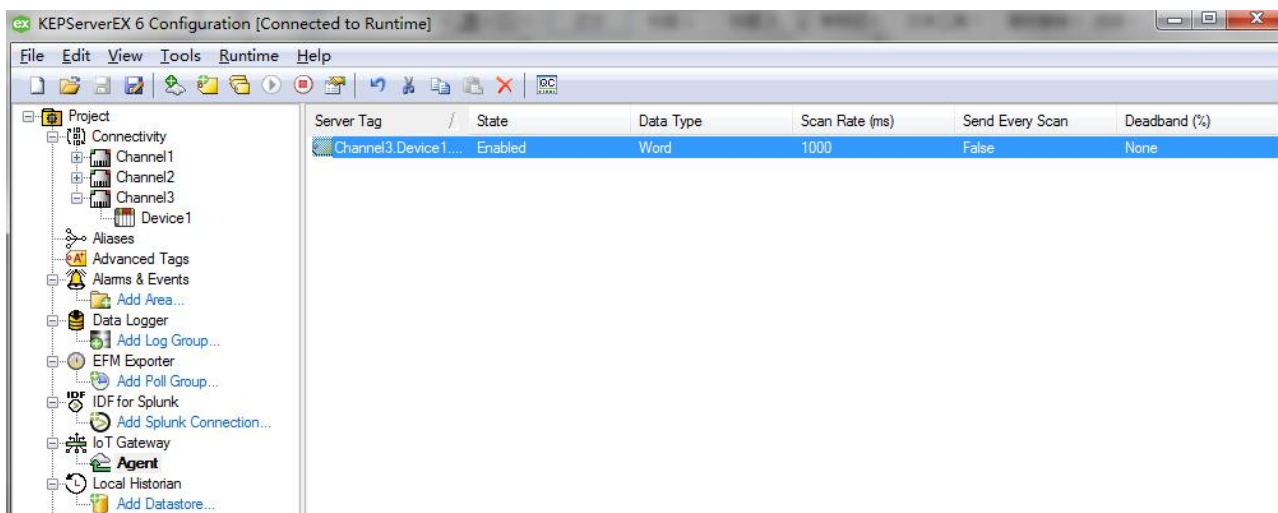
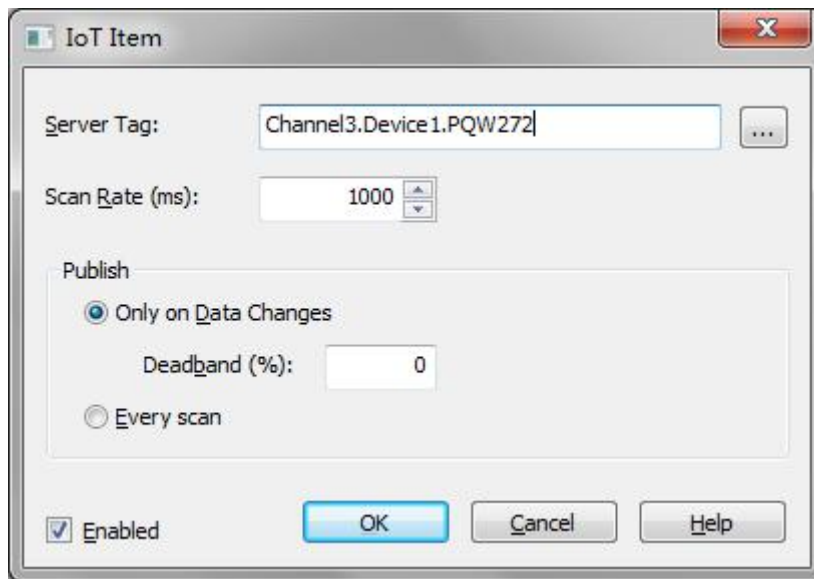


点击下一步，直至完成

增加一个 tag 到 REST Client Agent 里面。



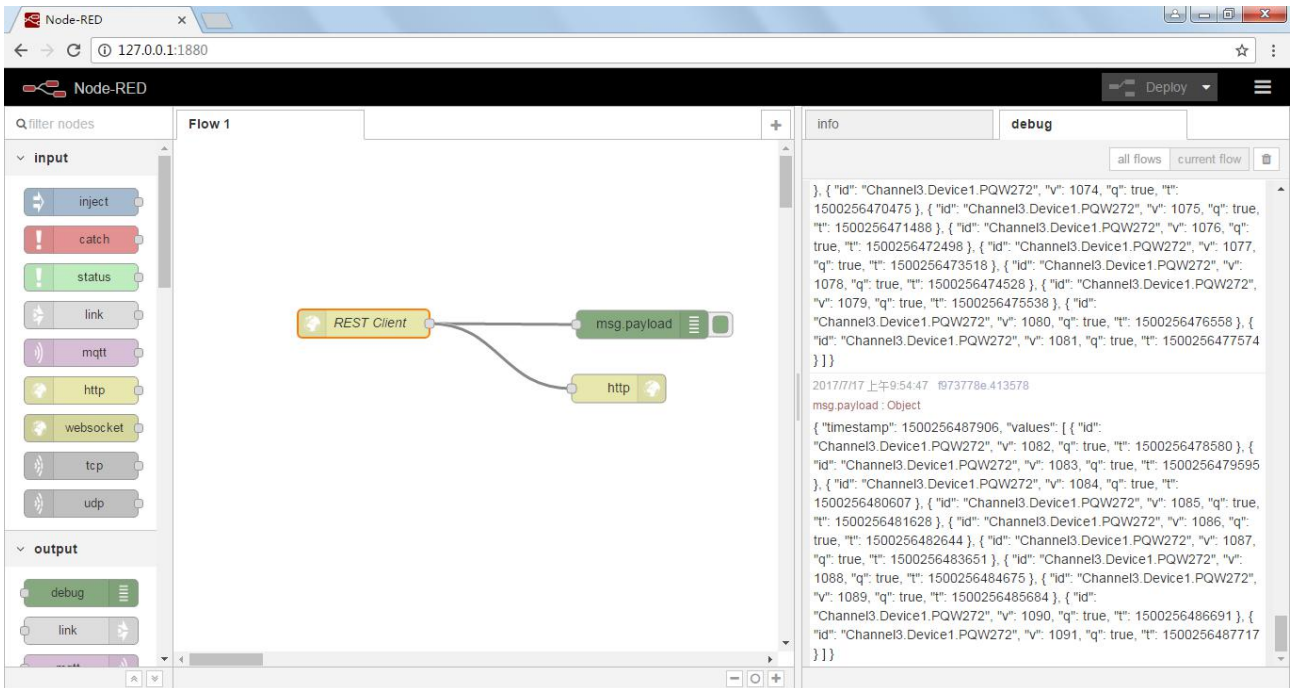




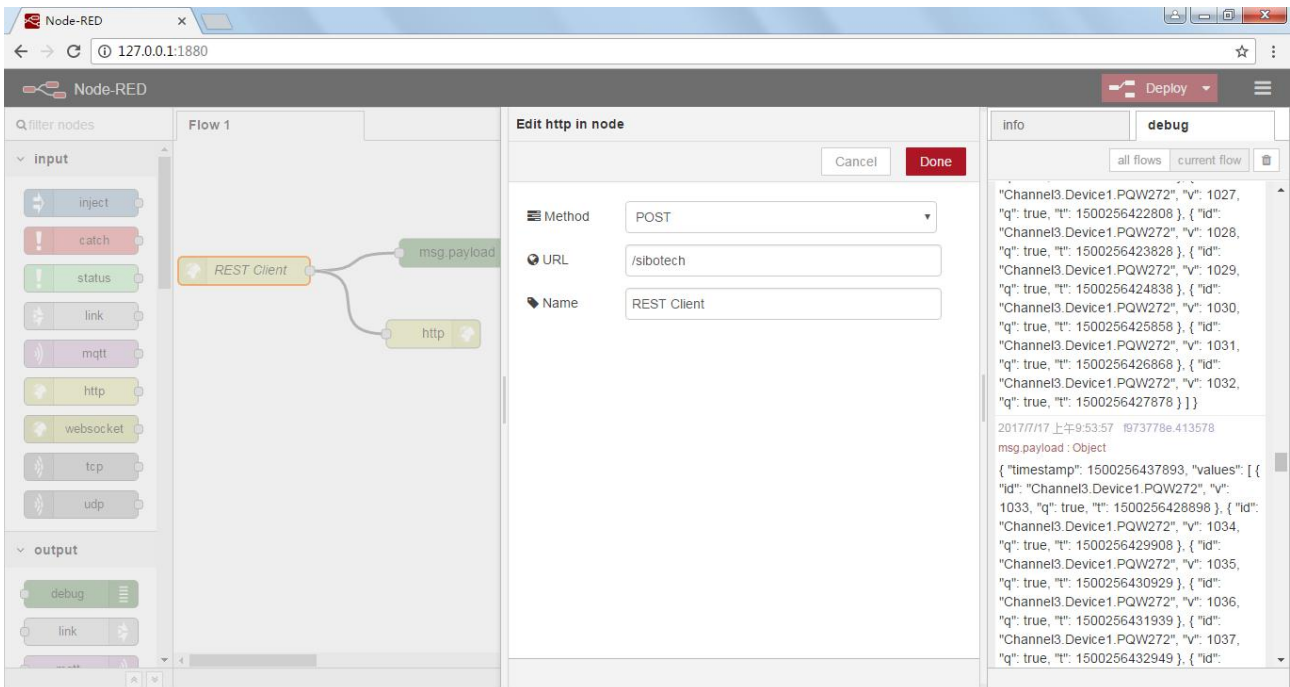
在 Node-Red 中，新建一个流向图如下：

<Method:POST;URL:./keptest>

部署好配置，JSON 格式的数据就会显示在“debug”界面



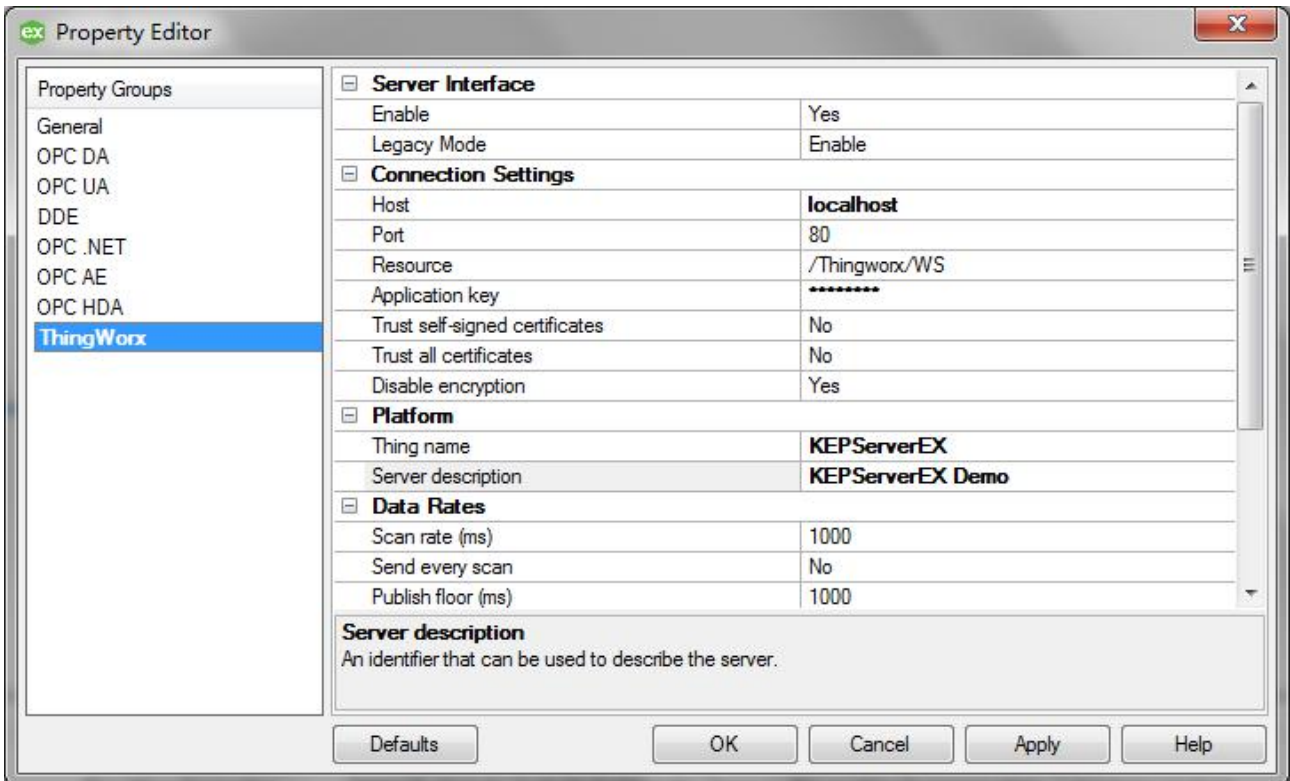
至此，使用 IOT-861 把数据传输到云平台演示完毕。





下面将演示如何连接到自己的 ThingWorx 平台：

1. 启用 ThingWorx Native Interface
  - a. 打开KEPServerEX, 选择Edit | Properties 选择 ThingWorx
  - b. Server Interface | Enable 选择 Yes
  - c. Legacy Mode选择Enable
2. 设置 Connection Settings 属性
  - a. 依次输入 ThingWorx 平台所使用的 Host 地址、Port 端口、Application Key  
(本示例中使用 Localhost、默认的 80 端口)
  - b. Disable encryption 选项因为没有使用任何安全加密策略，故置为 Yes，  
(如需更换 ThingWorx 使用的端口，需要在 Tomcat 配置文件中修改)
3. 设置 Thing 名称
  - a. Platform | Thing name 输入的名称需要和将来 ThingWorx 平台中的 Thing 名称一致  
(可以自定义描述信息以供参考)



4. 其余设置
  - a. Data Rates中可设置 KEPServerEX 的 Scan rate

(类似于 OPC 客户端程序对设备的扫描概念)

以及 Publish Floor (数据从 KEPServerEX 到 ThingWorx 的频率)

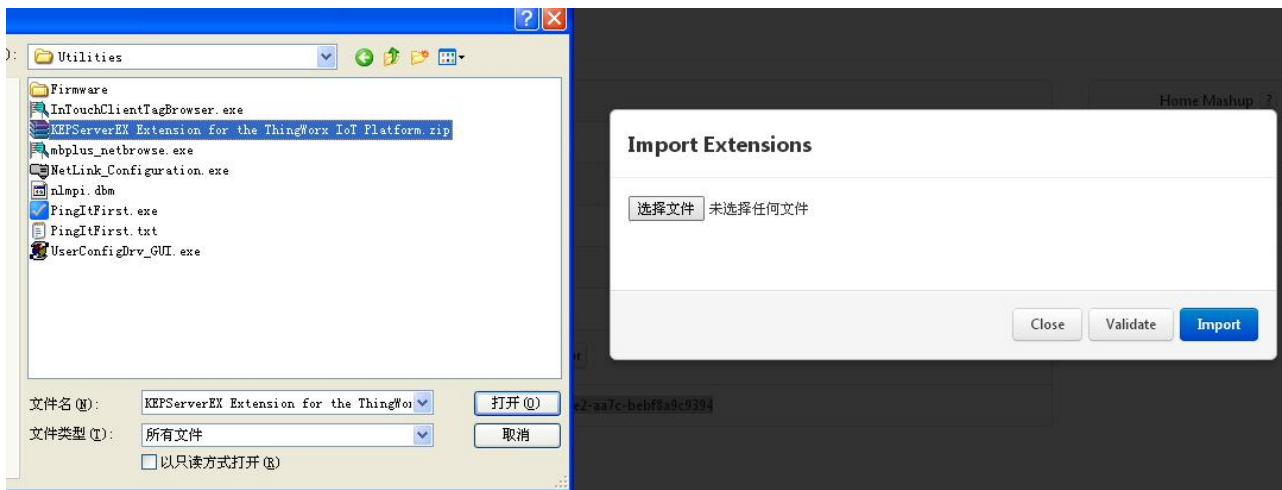
b. Logging 中选择 Yes, 可以在 EventLog 中看到关于 ThingWorx Native Interface 的相关事件记录, 方便故障诊断

5. 完成上述设置后, 运行 ThingWorx Composer

a. 导入扩展组件, 此 zip 组件随 V5.21 版本 KEPServerEX 安装文件名称为: KEPServerEX Extension for the ThingWorx IoT Platform

文件路径为 C:\Program Files (x86)\Kepware\KEPServerEX 5\Utilities

b. 导入成功会提示重启 ThingWorx Composer

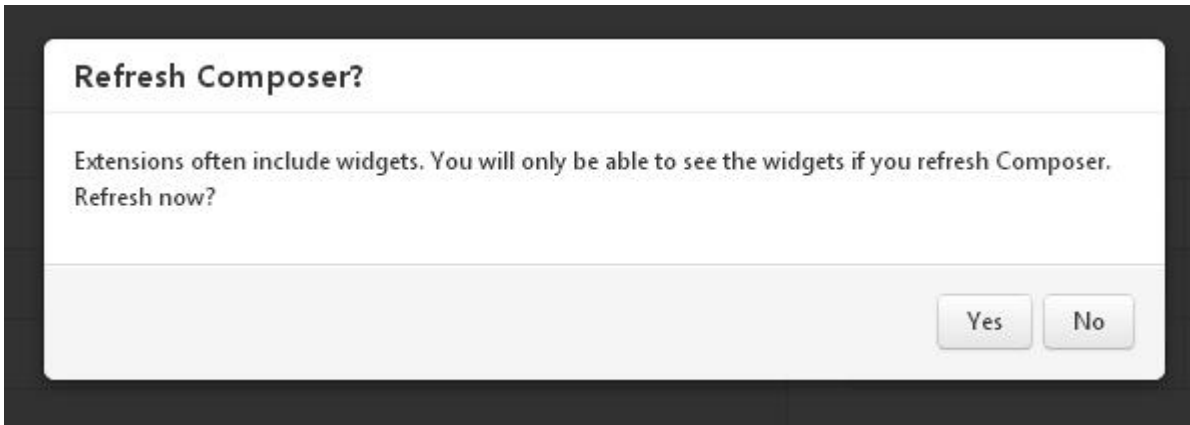


Import

提示导入结果, 已成功

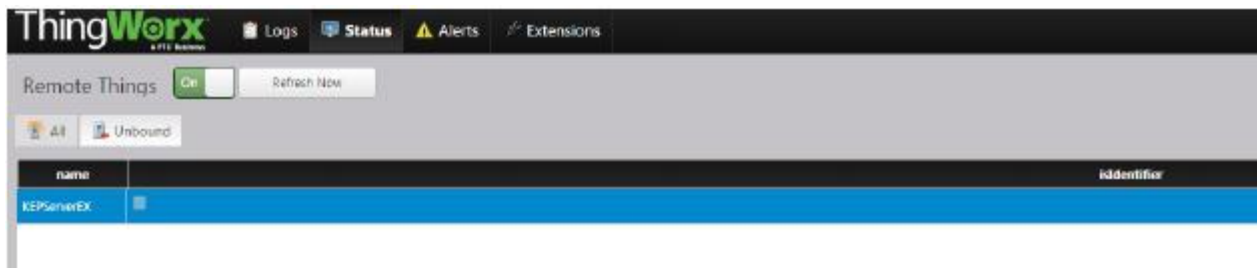
## Import Extensions

选择文件	KEPServerEX Extension for the ThingWorx IoT Platform.zip
<b>Validation Results</b>	
✓	KEPServerEX:1.6.0
<b>Installation Results</b>	
✓	KEPServerEX:1.6.0



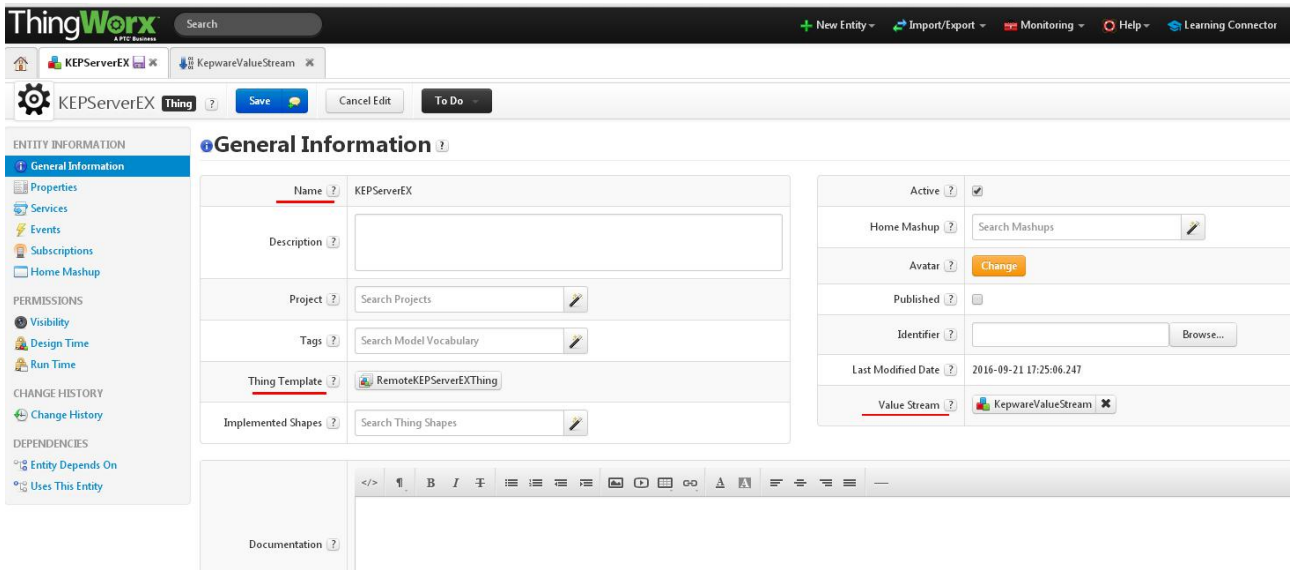
导入成功后，提示重启Composer，选择Yes

- c. ThingWorx Composer | Monitoring | Remote Things 中可以查看到新增的 Unbound Remote Things ‘KEPServerEX’(Thing 名称与步骤 3 中一致)

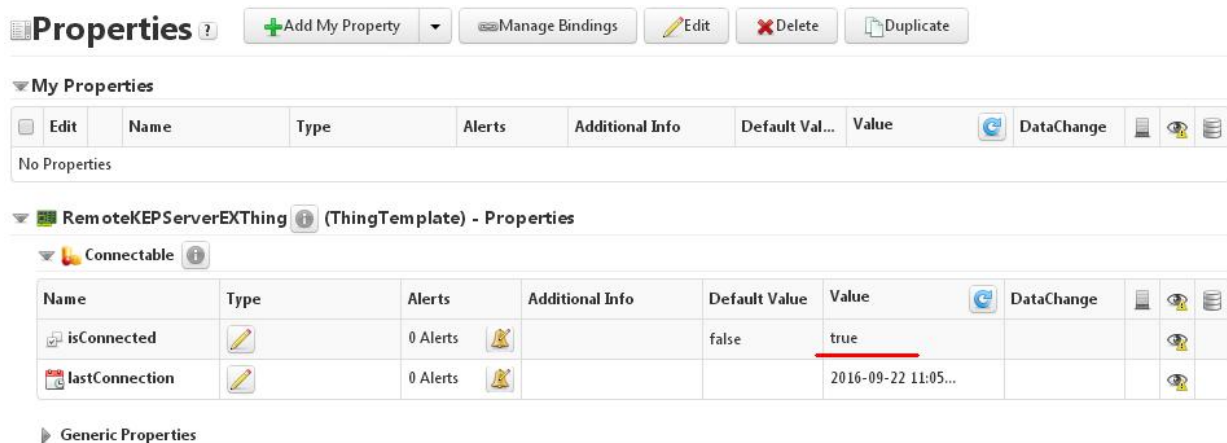


6. ThingWorx 中添加 New Thing

- a. Name 为 KEPServerEX
- b. Thing Template 为 RemoteKEPServerEXThing
- c. Value Stream 为 KepwareValueStream  
(安装步骤五中的扩展组件后可见)



d. Thing 'KEPServerEX' | Properties 中可以查看连接状态已变为 True



## 7. 添加 Properties

a. 点击 Services | AddItems | Test

Action	Service Name	Test	Service Type	Inputs	Output
	AddItems	Test	Remote AddItems	items	result
	BrowseGroups	Test	Remote BrowseGr...	- path - filter	result
	BrowseItems	Test	Remote BrowseIte...	- path - filter	result
	GetConfiguration	Test	Remote GetConfig...		result
	GetEventSubscriptions	Test	Local (Java Code)		result
	GetPropertySubscriptions	Test	Local (Java Code)		result
	GetQueuedTasks	Test	Local (Java Code)		result
	ProcessRemoteEvents	Test	Local (Java Code)	values	-
	RemoveItems	Test	Remote RemoveIt...	items forceRemove	result
	SendQueuedTasks	Test	Local (Java Code)		-

b. 弹出框中点击 Edit

Please be careful. Only execute services and queries where you understand the impacts.  
Note that this is testing the currently saved script - any changes you have made since you have saved this entity will not be reflected in this test!

Inputs:  
items

c. 点击 Add, 并输入相应信息

Source 为 KEPServerEX 中的标签地址, 如 Channel1.Device1.Tag1

Name 为 ThingWorx 中显示的 Property 名称, 如 Tag1

继续点击 Add, 可以添加多个 Tags/Properties

(可对数据类型、保存等属性进行编辑)

**Editing Infotable**

		+ Add		✖ Delete					
	ReadOnly	ScanRateMS	Description	BaseType	SourceType	Persistent	Logged	Source	Name
<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	ChannelLDevice.	Tag1
<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="checkbox"/>	ChannelLDevice.	Tag2

Cancel Save

d. 点击 Save 之后，点击 Execute Service成功返回

e. Properties 中添加对应的 Property

点击 Add My Property或者 Manage Bindings 进行 Tag-Property 的绑定

(Add My Property 中无需操作，打开关闭即可看到 Properties

Manage Bindings 中需要重新拖动添加)!

数据类型可能需要适当修改!

**Properties** ? + Add My Property Manage Bindings Edit Delete Duplicate

▼ My Properties

<input type="checkbox"/>	Edit	Name	Type	Alerts	Additional Info	Default Value	Value	<input type="checkbox"/>	DataChange	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>		Tag1	KEPServerEX: Tag1	0 Alerts				<input type="checkbox"/>	Set Value	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>		Tag2	KEPServerEX: Tag2	0 Alerts				<input type="checkbox"/>	Set Value	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

▼ RemoteKEPServerEXThing (ThingTemplate) - Properties

▼ Connectable

Name	Type	Alerts	Additional Info	Default Value	Value	<input type="checkbox"/>	DataChange	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
isConnected		0 Alerts		false	true	<input type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
lastConnection		0 Alerts			2016-09-22 11:05:0...	<input type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

▶ Generic Properties

8. 刷新即可看到数据变化，由此完成数据从 KEPServerEX 到 ThingWorx 平台的传输

The screenshot shows the KEPServerEX Properties window for a 'Thing' entity. The interface includes a top toolbar with 'Save', 'Cancel Edit', and 'To Do' buttons. A left sidebar lists various entity information categories like General Information, Properties, Services, Events, Subscriptions, Home Mashup, Permissions, Visibility, Design Time, Run Time, Change History, and Dependencies.

The main content area is divided into two sections:

#### My Properties

Edit	Name	Type	Alerts	Additional Info	Default Value	Value	DataChange	Value	Set
	Tag1	Channel1@@Devi...	0 Alerts	Push: ALWAYS		538	Value		<input type="button" value="Set"/>
	Tag2	Channel1@@Devi...	0 Alerts	Push: ALWAYS		0	Value		<input type="button" value="Set"/>

#### RemoteKEPServerEXThing (ThingTemplate) - Properties

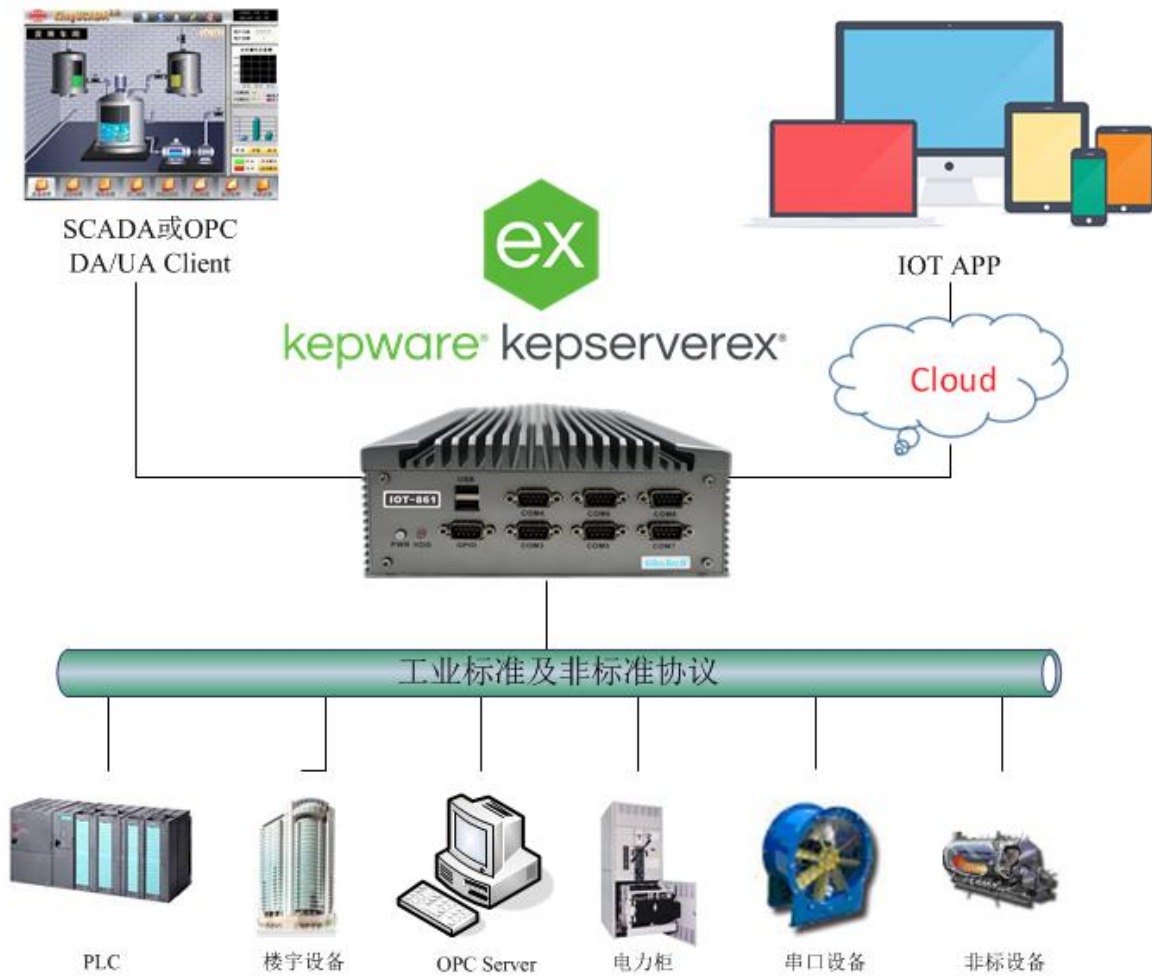
Connectable

Name	Type	Alerts	Additional Info	Default Value	Value	DataChange
isConnected		0 Alerts		false	true	
lastConnection		0 Alerts			2016-09-22 11:17:4...	

Generic Properties

## 五、典型应用

IOT-861 可以将工业标准及非标准的协议设备采集过来，如 PLC、楼宇设备、OPC Server 等，统一的转换成 OPC 协议，可以提供标准的 DA/UA Server 接口供客户端访问，同时，它也能将数据传送到云端，供 IOT APP、浏览器等应用安全地访问。

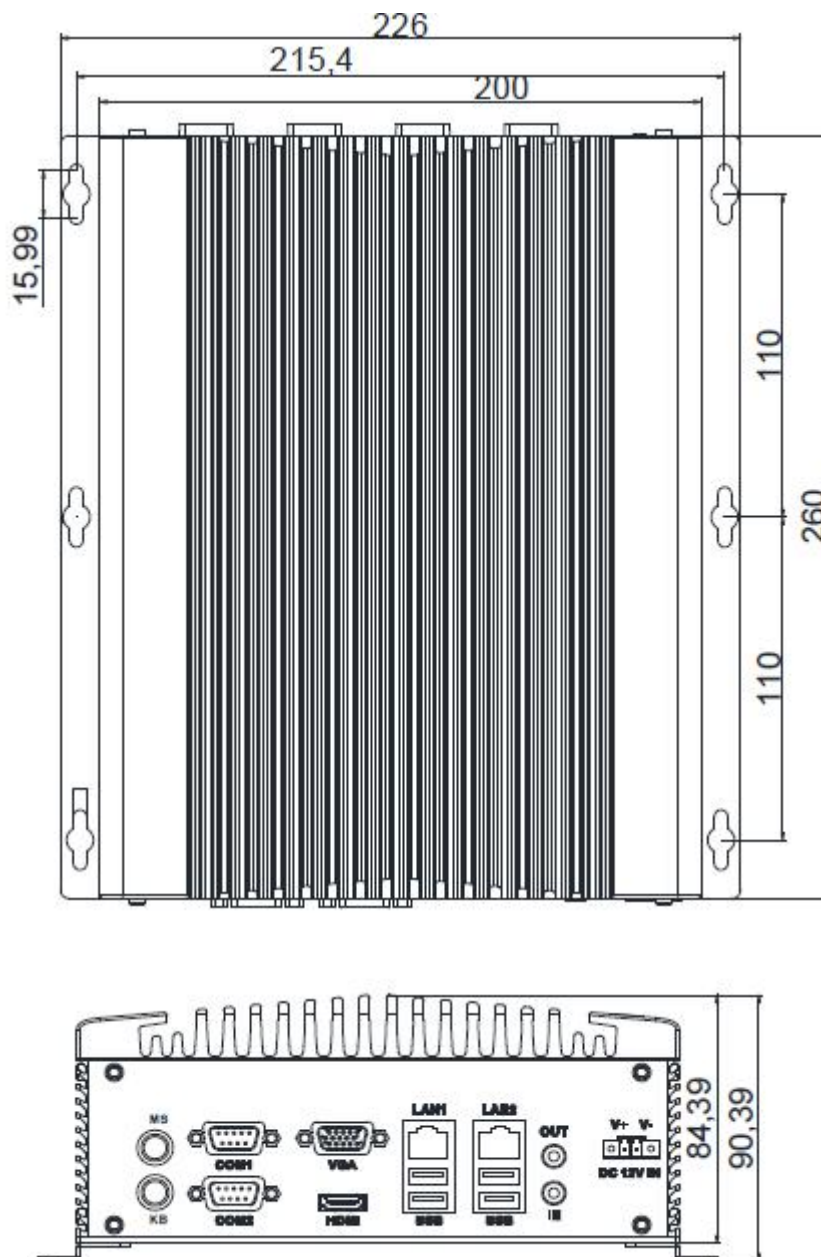




## 六、安装

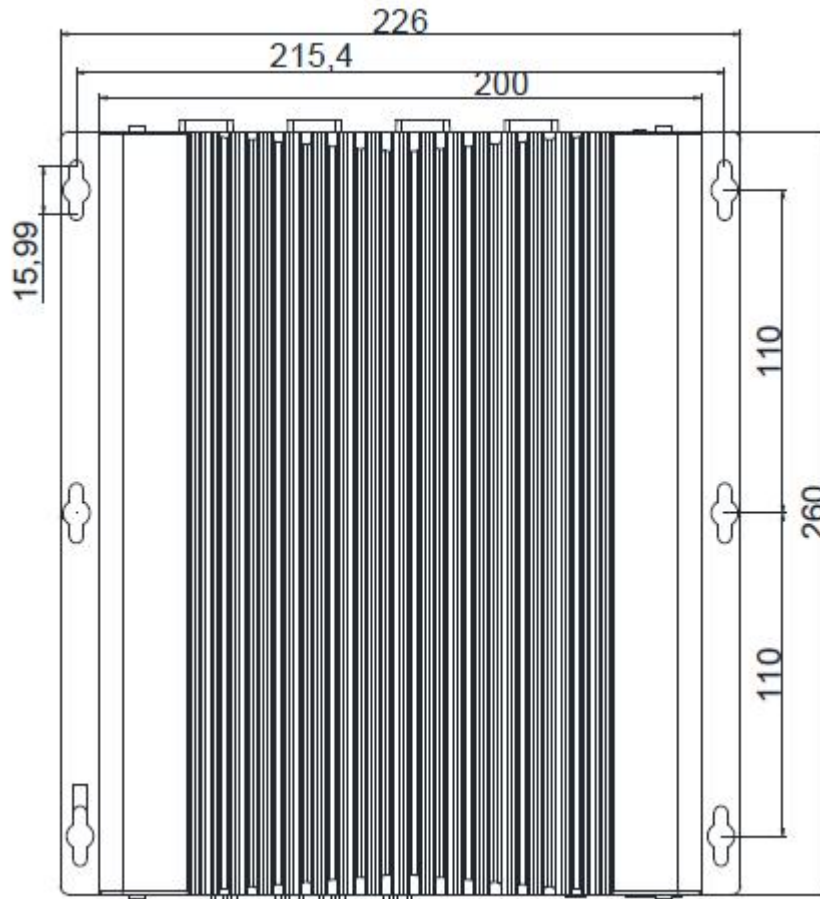
### 6.1 机械尺寸

尺寸 (W\*H\*D): 22.6cm\*9cm\*26cm



## 6.2 安装方法

壁挂式安装。



## 七、修订记录

时间	修订版本	修改内容
2019-1-15	A	修改产品图片，修改了电源、串口数量、USB 口数量以及产品尺寸。