www.nimotion.cn

立迈胜-专业的运动控制平台

NGW2000M-D01 可编程网关

使用说明书



目　录

1 安全说明和警告 1
1.1 重要信息 1
1.2 安全注意事项 1
2 关于手册 3
2.1 简介 3
2.2 适用范围 3
3 技术规格 4
3.1 产品型号定义 4
3.2 物理特性与尺寸 4
3.3 技术数据 5
3.4 指示灯信号 9
3.5 连接器定义 9
4 安装调试 12
4.1 安全须知 12
4.2 安装 13
4.3 其他 16
5 网关应用实例 16
5.1 网关连接及程序下载 16
5.2 通信端口和通信协议配置 27
5.3 通信协议之间的变量映射 28
5.4 运动控制功能 31
6 Profinet 配置 32
6.1 Profinet 主站配置 32
6.2 Profinet 从站配置 36
7 Ethercat 配置 41
7.1 Ethercat 设备描述文件添加 41
7.2 EtherCAT 主站配置 42
7.3 自动扫描添加 EtherCAT 设备 43
7.4 手动添加 EtherCAT 设备 44
7.5 CANopen 主站配置 45
7.6 CANopen 从站配置 52
8 MODBUS TCP 配置 54

立迈胜公司版权所有

8.1 Modbus TCP 主站配置 54
8.2 Modbus TCP 从站配置 57
9 MODBUS RTU 配置 60
9.1 Modbus RTU 主站配置 60
9.2 Modbus RTU 从站配置 63
10 OPCUA 服务端配置 66
11 自定义 CAN 配置 66
12 自定义 UDP 配置 68
版本变更记录 73

1 安全说明和警告

1.1 重要信息

在安装和调试 NGW2000M-D01 网关之前，必须仔细阅读本手册；严禁擅自转载、复制本说明书的部分或全部内容；北京立迈胜控制技术有限责任公司有权为了提高产品的性能，进行技术改造，进一步优化改进硬件和软件，恕不另行通知。

1.2 安全注意事项

这里提示的注意事项，目的是为了使您能安全、正确地使用产品，并防患于未然，以免给您和他人造成危害或损伤。请您对其内容充分理解以后再使用本产品。

⚠警告	在操作时违反本警告事项所示的内容要求，可能会导致人员死亡或负重伤。
⚠注意	在操作时违反本警告事项所示的内容要求，可能会导致人员负伤或造成物品损坏。
⚠重要	为了使您能正确地使用产品，在正文的相关使用项目中记载着用户务必遵守的事项。

⚠ 警告

整体

- 请勿在爆炸性气体环境、易燃性气体环境、腐蚀性环境、容易沾水的场所以及可燃物附近使用本产品，否则有可能引起火灾或致伤。
- 设置、连接、运行、操作、检查、故障诊断作业请由具备适当资格的人实施，否则有可能引起火灾、致伤或造成设备损坏。
- 请勿在通电状态下进行移动、设置、连接和检查等操作，请切断电源后再进行，否则有可能引起触电或设备损坏。

搬运

- 搬运时请勿手持网关电缆线，否则有可能造成设备损坏。

安装

- 请将网关安装在指定的、安全的工作环境内，否则有可能造成设备损坏。

连接

- 网关的电源输入电压请务必控制在额定范围内，否则有可能引起设备损坏或火灾。
- 网关的电源，请使用初级和次级强化绝缘的直流电源，否则有可能引起触电。
- 请严格按照连接图进行连接，否则有可能引起设备损坏。
- 请勿强行弯曲、拉扯或夹住连接电缆线。

修理、拆解、改造

- 请勿将网关进行拆解或改造，否则有可能致伤。需要检查或修理时，请与北京立迈胜控制技术有限责任公司联系。

⚠ 注意

整体

- 使用网关时，请勿超过其规格允许的范围，否则有可能致伤或造成设备损坏。

安装

- 网关周围请勿堆放妨碍通风的障碍物，否则有可能造成设备损坏。

运行

- 请按指定的参数要求使用网关，否则有可能造成设备损坏。

保养、检查

- 进行绝缘电阻测量或绝缘耐压试验时，请勿碰触网关，否则有可能引起触电。

报废

- 网关报废时，请尽可能将其拆解，作为工业废弃物实施处理。

⚠ 重要

- 请由具备电气、机械工业等专业知识的人员使用本产品。
- 使用前，请熟读并充分理解「安全说明和警告」，以便正确地使用网关。
- 本产品是为在一般工业设备中使用而设计制造的。请勿将其用于其它用途。无视本忠告而造成的损失，本公司将不承担任何赔偿责任，特此声明，敬请谅解。

2 关于手册

2.1 简介

NGW2000M-D 系列可编程网关是北京立迈胜控制技术有限责任公司最新推出的新一代多协议可编程转换网关，NGW2000M-D01 搭载 CODESYS V3 控制平台；不仅支持常规工业总线通信协议转换，还可以进行基于工业总线通信的伺服电机控制；该转换器支持 IEC61131-3 编程语言、逻辑控制功能及常规运动控制功能。

NGW2000M-D01 支持 EtherCAT、MODBUS RTU/TCP、Profinet、CANopen、OPCUA 等多种现代工业现场总线通信协议，可根据客户需要扩展，性能强大的 ARM Cortex A8 处理器支持复杂的数据处理、实时计算等，适用于现代工业中复杂、多变、以及苛刻的用户需求，为您提供可扩展、简洁、经济而且先进的解决方案。

本手册用于介绍 NGW2000M-D 系列可编程网关的产品功能、产品设置、产品用途及使用方法等，使用本产品前，请仔细阅读这些说明。

2.2 适用范围

NGW2000M-D01 可编程网关

3 技术规格

3.1 产品型号定义

NGW2000M-D01 系列运动网关系列产品每种产品型号的定义如下图所示：

NGW 2000 M - D XX

① ② ③ ④ ⑤

①	NGW：立迈胜可编程网关产品
②	2000：产品系列号
③	M：标准型号
④	D：硬件版本号
⑤	xx：软件类型
	01：Codesys 软件，CoDeSYSControl 授权
	10：NIMCS2000 软件

图 3-1

表 3-1 产品型号定义

产品型号	控制软件	功能描述
NGW2000M-D01	Codesys软件	Codesys runtime for BeagleBone SL License NIM_SML轻量单轴运动控制库 NIM_SMB标准单轴运动控制库
NGW2000M-D10	NIMCS2000 软件	NIM_SML轻量单轴运动控制库 NIM_SMB标准单轴运动控制库

3.2物理特性与尺寸

尺寸单位：mm

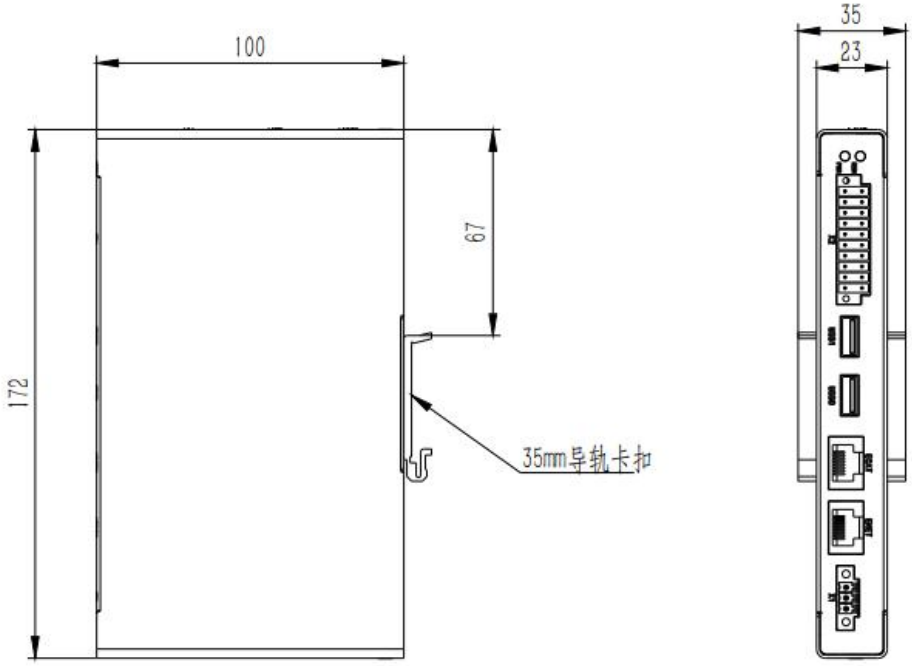


图 3-2

3.3技术数据

3. 3. 1产品电气参数

本产品详细规格参数如下表所示:

表 3-2 规格参数

硬件规格	
外形尺寸	172 × 100 × 23mm
安装方式	导轨安装、螺丝安装
重量	450g
外壳材质	镀锌冷板（CGCC）
输入电源	
电源电压	10VDC~52VDC
输入电压	CE：电源必须符合 IEC 60950 和 IEC 61558-2-4 安全标准
内部功耗	24VDC 额定输入电压最大功耗 5W
保险丝	内置自恢复保险丝
系统	
处理器	ARM Cortex A8 800MHz 256KB L2 cache
内存	512MB

闪存	4GB eMMC
操作系统	
操作系统版本	Debian9.1
系统内核	Linux 4.9.45
串行端口	
RS232	1 路，非隔离
RS485	1 路，隔离
CAN	1 路，隔离
以太网通讯	
端口数量	2
EtherNET	10/100 Base-T (RJ45)　支持以太网和 Modbus-TCP 协议
其他端口	
SD 卡	支持 Micro SD
USB0	仅支持虚拟以太网接口使用(可以用来调试程序)
USB1	USB 2.0　500mA（支持接入 WiFi 模块）
指示灯	
PWR	绿色，上电后常亮
RUN	黄色，运行时闪烁
其他硬件	
蜂鸣器	有，上电后自动蜂鸣一声
RTC	支持
掉电保存	支持
环境	
操作温度	-10℃ ~ +60℃
存储温度	-20℃ ~ +70℃
湿度	5% ~ 85% RH 无冷凝
认证及标准	
常规	CE

3.3.2 NGW2000M-D01产品软件参数

表 3-3 软件参数

软件规格	
实时补丁	RT-PREEMPT
软件运行版本	Codesys 3.5.15 及以上

编程语言	IEC61131-3 标准编程语言：SFC（顺序功能图）、LD（梯形图）、FBD（功能块）、ST（结构化文本）、IL（指令表）；支持单步程序调试	
支持的通信协议		
EtherCAT	支持主站	EtherNET 1/2
Profinet	支持主/从站	
Ethernet/IP	支持主/从站	
Modbus TCP	支持主/从站	
OPCUA	支持服务端、客户端	
自定义协议	自定义 TCP、UDP	
CANopen	支持主/从站	CAN
自定义协议	自定义 CAN2.0A/CAN2.0B	
Modbus RTU	支持主/从站	RS485
自定义协议	自定义串口协议	
通信站点数量		
CANopen 主/从站	1 路	
Modbus RTU 主/从站	1 路	
Ethercat 主站	最多 2 路	
Profinet 主/从站	最多 2 路	
Modbus TCP 主/从站	最多 2 路	
Ethernet/IP 主/从站	最多 2 路	
支持转换的通信协议		
Profinet 从站	Ethercat	主站
	CANopen	主/从站
	Modbus TCP/RTU	主/从站
	Ethernet/IP	主/从站
	OPCUA	服务端
	自定义协议（CAN、TCP、UDP）	

Modbus TCP 从站	Ethercat	主站
	CANopen	主/从站
	Modbus TCP/RTU	主/从站
	Ethernet/IP	主/从站
	Profinet	主/从站
	OPCUA	服务端
	自定义协议（CAN、TCP、UDP）	
Modbus RTU 从站	Ethercat	主站
	CANopen	主/从站
	Modbus TCP	主/从站
	Ethernet/IP	主/从站
	Profinet	主/从站
	OPCUA	主站（服务端）
	自定义协议（CAN、TCP、UDP）	
CANopen 从站	Ethercat	主站
	Modbus TCP/RTU	主/从站
	Ethernet/IP	主/从站
	Profinet	主/从站
	OPCUA	服务端
	自定义协议（TCP、UDP）	
Ethernet/IP 从站	Ethercat	主站
	CANopen	主/从站
	Modbus TCP/RTU	主/从站
	Ethernet/IP	主/从站
	Profinet	主/从站
	OPCUA	服务端
	自定义协议（CAN、TCP、UDP）	
自定义协议（CAN、TCP、UDP）	Ethercat	主站
	Modbus TCP/RTU	主/从站
	Ethernet/IP	主/从站
	Profinet	主/从站
	OPCUA	服务端
	自定义协议（TCP、UDP）	
运动控制功能		

（运动控制库）	
NiM_SML	立迈胜自研轻量单轴运动控制库（支持 CANopen、Ethercat、MODBUS RTU 协议的电机普通运动控制）
NiM_SMB	立迈胜自研标准单轴运动控制库（支持 CANopen、Ethercat 协议的电机循环同步模式的运动控制）
NiM_Robotics	立迈胜自研机器人运动控制库（支持常规工业机器人构型）

3.4指示灯信号

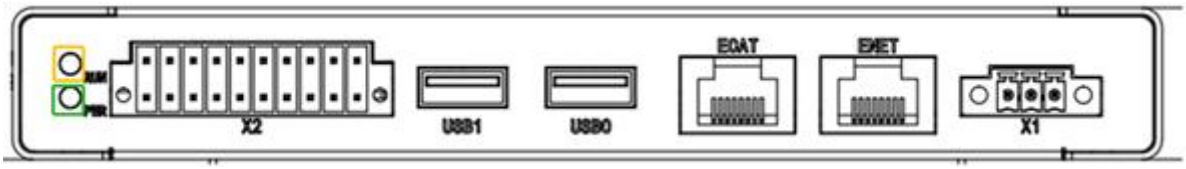


图 3-3

表 3-4

PWR（绿）	RUN（黄）	状态含义
常亮	常亮	电源正常，网关未工作
常亮	慢闪	电源正常，网关工作正常
常亮	快闪	电源正常，PLC 程序运行
灭	灭	未上电或电源故障

3.5 连接器定义

3.5.1概览

运动网关的连接器如下图所示，本图为网关前视图。

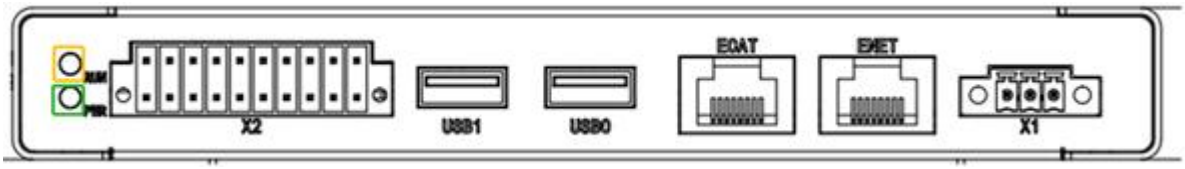
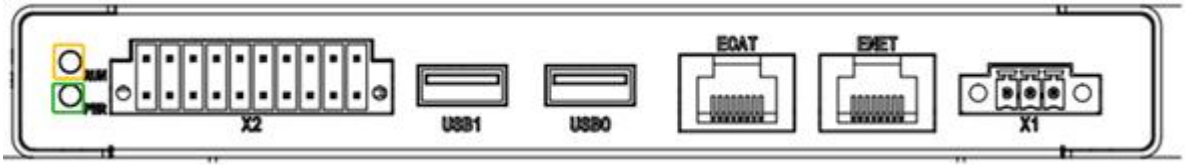


图 3- 4

3.5.2X1（电源）

X1 为系统电源接口，前视图中引脚排序如下图示，左侧引脚为 1。



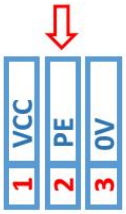


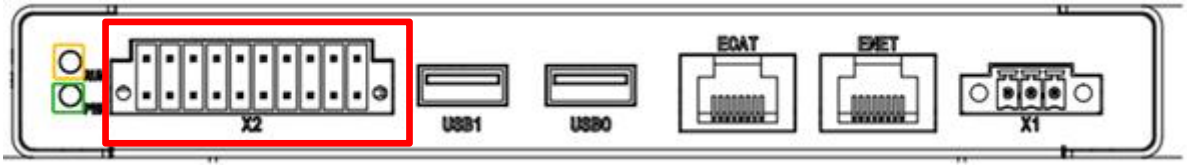
图 3- 5

表 3-5

引脚	功能	说明
1	VCC	电源输入正极
2	PE	保护地
3	0V	电源公共端

3.5.3 X2（通信）

X2 为 485、232 和 CAN 连接器，前视图中引脚排序如下图示，左上角为引脚 1。



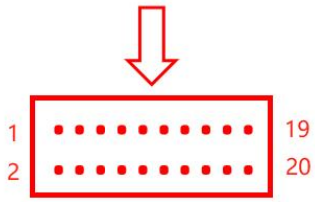


图 3- 6

表 3-6

引脚 Pin	功能 Features	说明 Instructions	备注
1	232T	RS232 TXD 发送	
3	232R	RS232 RXD 接收	
5	232GND	RS232 接地端	非隔离地
7	485A	RS485_A	
9	485B	RS485_B	
11	485G	RS485 接地端	隔离地
13	CANH	CAN H	

15	CANL	CAN L	
17	CANG	CAN 接地端	隔离地
19	—	—	
20	—	—	

注意事项:

1、232 的接地端是非隔离的，串口连接其它设备的时候，系统电源要共地。

2、485 接地端以及 CAN 的接地端都是隔离的，需要单独去接，不能接在一起；也不能接在数字地 DGND 上。

串口号说明: 表 3-7

硬件接口	Codesys 串口号	NIMCS2000 串口号
RS232	COM2	ttyO1
RS485	COM4	ttyO3
CAN	1	1

3.5.4 TF卡（I/O）

TF 卡:

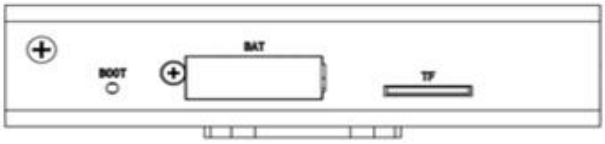


图 3-7

3.5.5 EtherCAT/EtherNET

EtherCAT/EtherNET 接口:

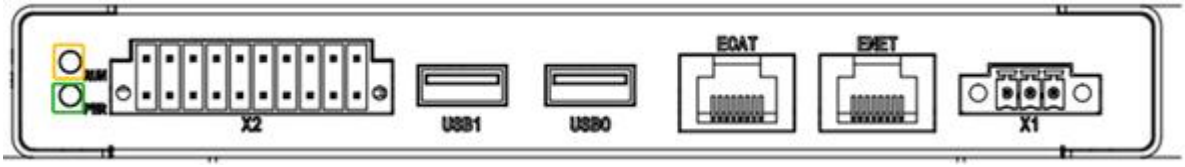


图 3-8

EtherCAT: 默认 IP 地址：192.168.0.200

EtherNET: 默认 IP 地址：192.168.1.200

3.5.6 BOOT按键

BOOT 按键用于选择系统启动位置，默认情况下从网关内部 EMMC 启动，按下 BOOT 键从 SD 卡启动，如下：

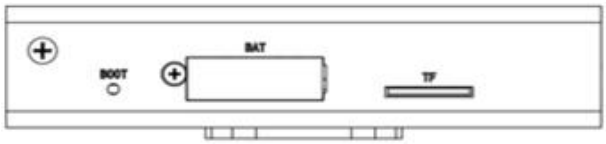


图 3-9

3.5.7 BAT电池

说明：电池为自充电电池，正常情况下无需客户自行更换；当设备 RUN 指示灯以 100ms 间隔持续闪烁时，表示电池电量低，请注意及时更换，更换时请注意将电池正极朝上安装。

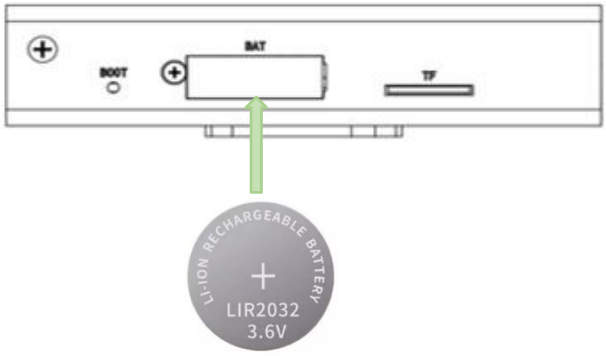


图 3-10

4 安装调试

4.1 安全须知

⚠ 警告

网关使用的环境温湿度应当满足-10℃ ~ +60℃相对湿度 10%RH～85%RH，且无凝结。

网关电源电压额定为 24/48V，最大不能超过 52V，注意电源极性不能接反。

⚠ 重要

线缆连接：

- 网关电源线采用 20～22AWG 红黑双色双绞铜导线。
- CAN 通信线采用特性阻抗为 120Ω，22～24AWG 双绞屏蔽线。
- IO 通道信号电缆长度不超过 1 米，采用 24AWG 红黑双色双绞线铜导线，大于 1 米时

信号要选用屏蔽电缆。

● 各类电缆布线应分束、分槽布线，不同类的电缆发生交叉，电缆与电缆之间要成直角。

IO 信号线和 CAN 通信线离电源线＞0.2m。

接地要求：

本网关为金属外壳，使用过程中需保证系统接地良好：

● 网关外壳要通过安装孔安装到网关支架上，支架要可靠接保护地，接地电阻≤0.1Ω。

电气隔离要求：

CAN 通信、RS485 通信、IO 端口等为电气隔离，使用过程中保持与系统电源的良好电气隔离。

4.2 安装

网关共有三种安装方式，分别为：

1. 背部导轨安装：

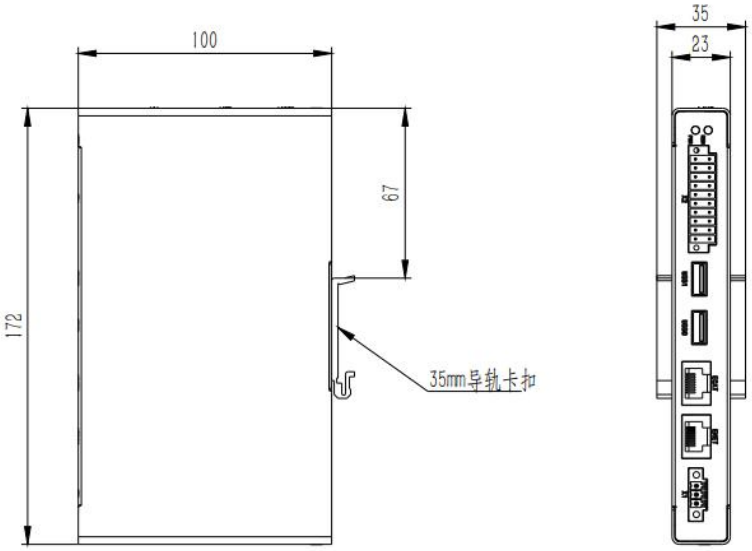


图 4-1

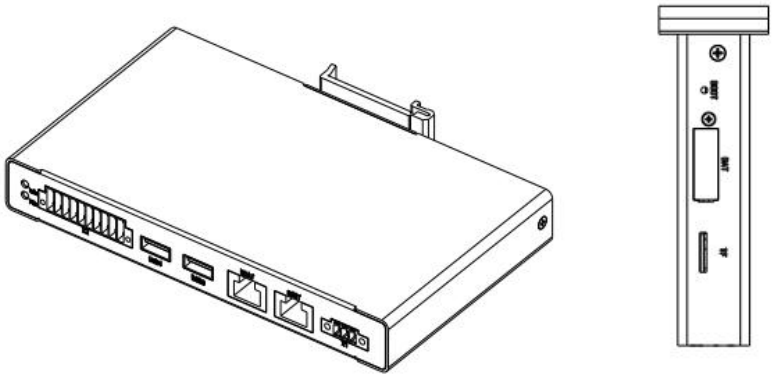


图 4-2

2. 底部导轨安装：

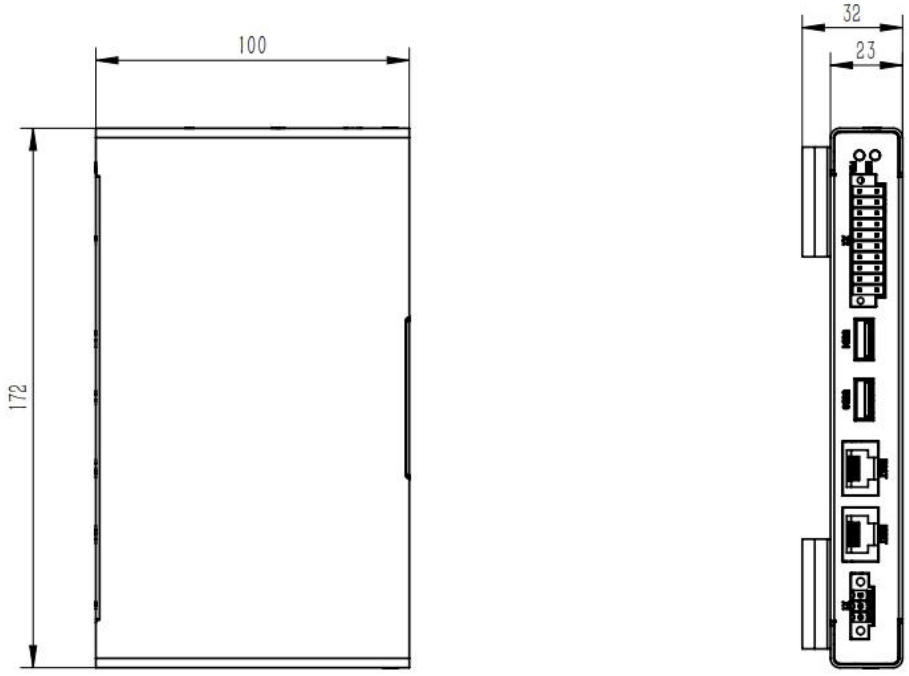


图 4-3

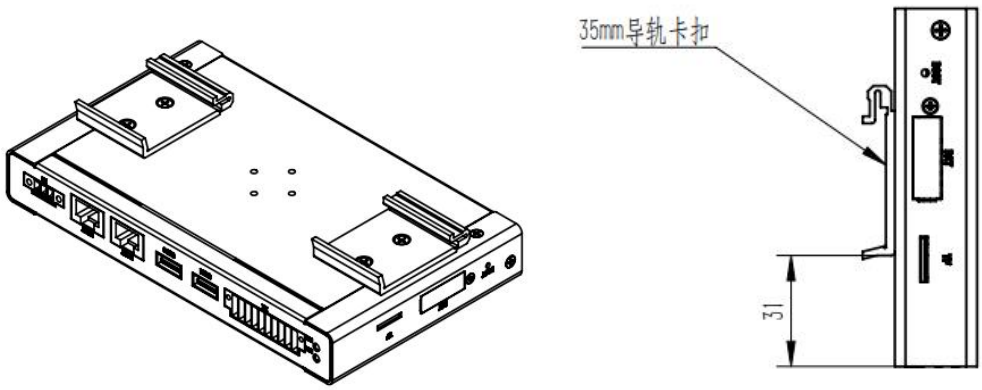


图 4-4

3. 底部螺丝安装：

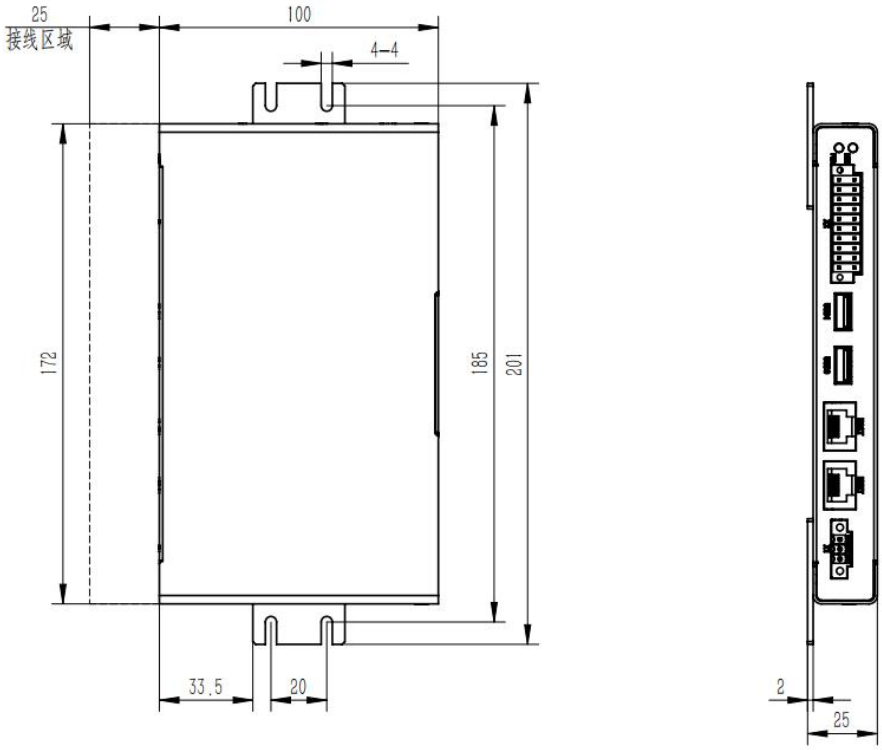


图 4-5

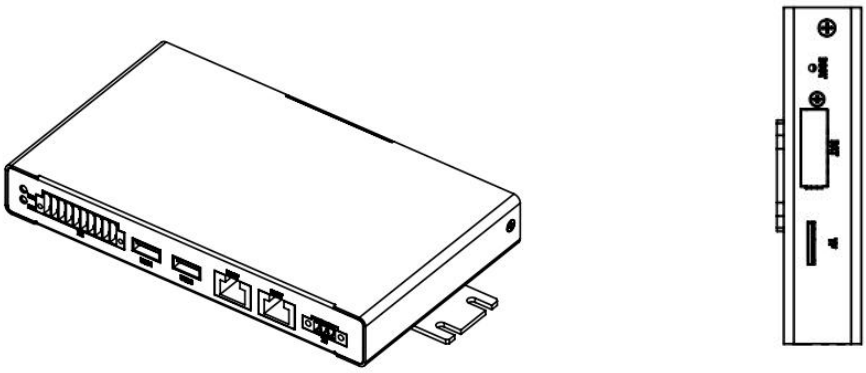


图 4-6

4.3其他

在给网关供电前，确认正确连接通信线缆、电源线缆、需要使用 IO 功能的要连接 IO 线缆。

5 网关应用实例

5.1 网关连接及程序下载

网关连接时所需要的组件:

- 依据手册要求的直流电源
- 可编程网关
- 所需要的通信线缆（标准网线、USB 公转公线）
- 一台安装好 CODESYS 软件的 PC 机

5.1.1 通过网线连接产品

使用网线连接 PC 机与本产品，并将 PC 机本地地址设置为 192.168.0.100（本产品 IP 地址为: 192.168.0.200）。

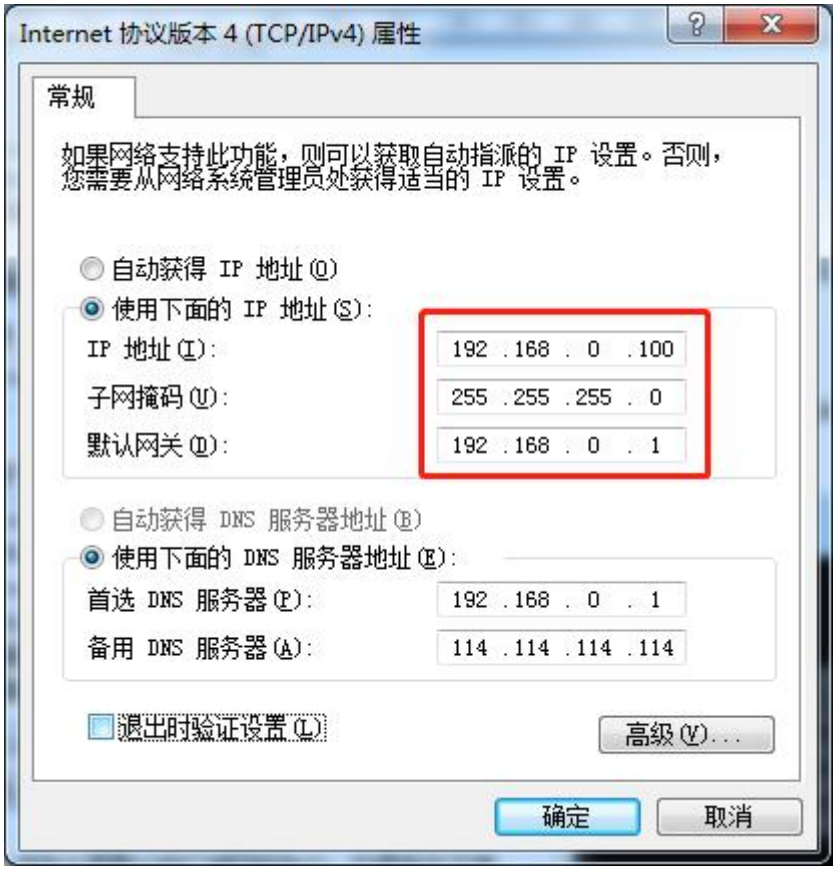


图 5-1

给本产品上电，在 PC 机上 ping 192.168.0.200，能 ping 通说明本产品已启动且通信正常。如下图所示：

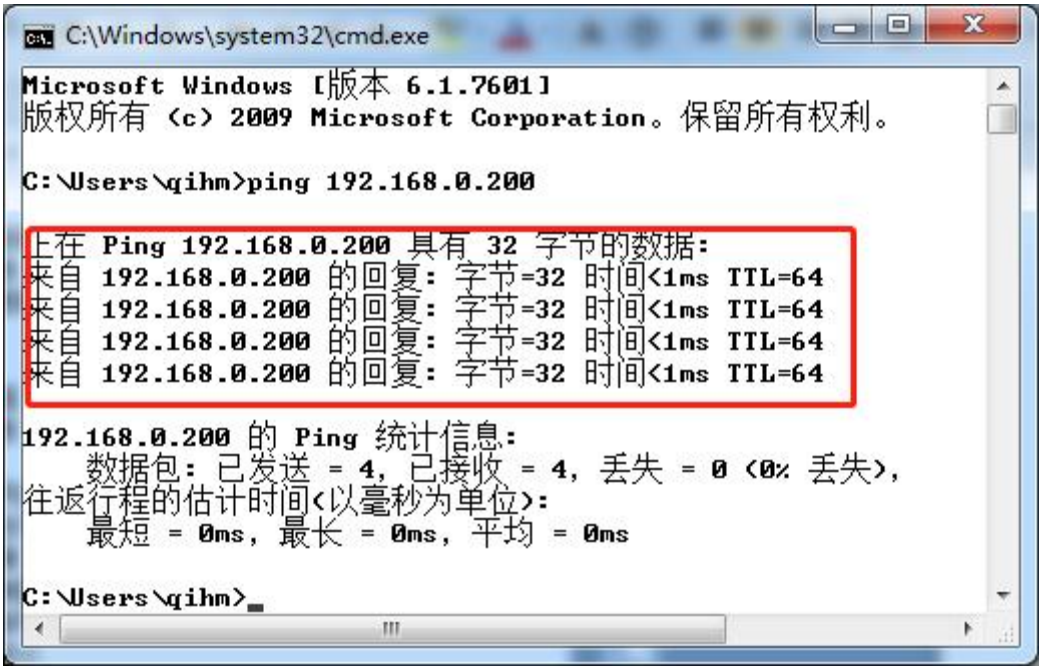


图 5-2

5.1.2 通过USB线连接本产品

使用公对公 USB 线连接电脑和本产品 的 USB0 接口，根据操作系统选择安装合适的驱动程序，下面以 64 位 Windows10 为例介绍驱动安装过程。

1.在 Windows10 安装该驱动需要先关闭强制数字签名，关闭方法如下：

（1） 按下 win+i 组合键打开 Windows 设置，点击“更新和安全”



图 5-3

（2） 找到“恢复”，点击“高级启动”下的“立即重新启动”，重启电脑；通过点击开始菜单，按下 shift 键并点击“电源”->“重启”达到同样效果。



图 5-4

（3） 重启 Windows10 后在高级界面中选择“疑难解答”



图 5-5

（4） 在疑难解答界面中选择“高级选项”

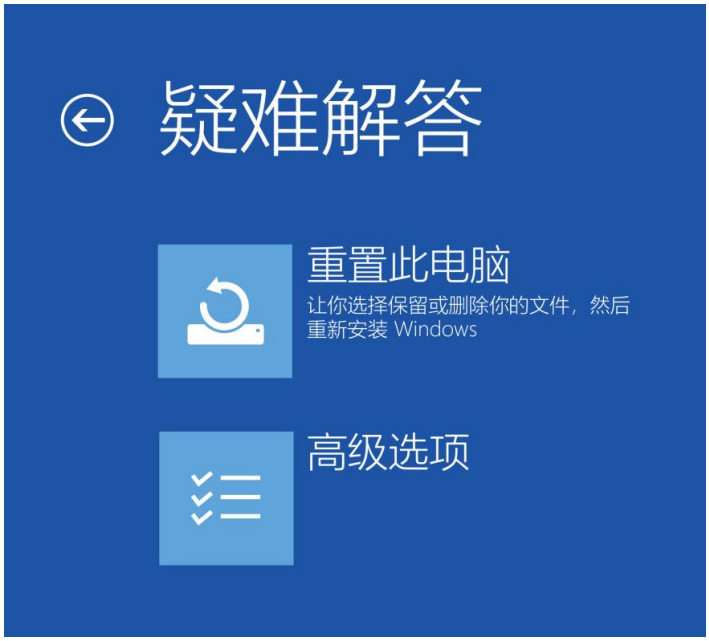


图 5-6

（5） 接着点击“启动设置”



图 5-7

（6） 点击“重启”，会进入“启动设置”页面



图 5-8

（7） 按提示输入“7”禁用驱动程序强制签名，至此数字签名已被禁用，待系统重启完成后即可正常安装驱动程序。

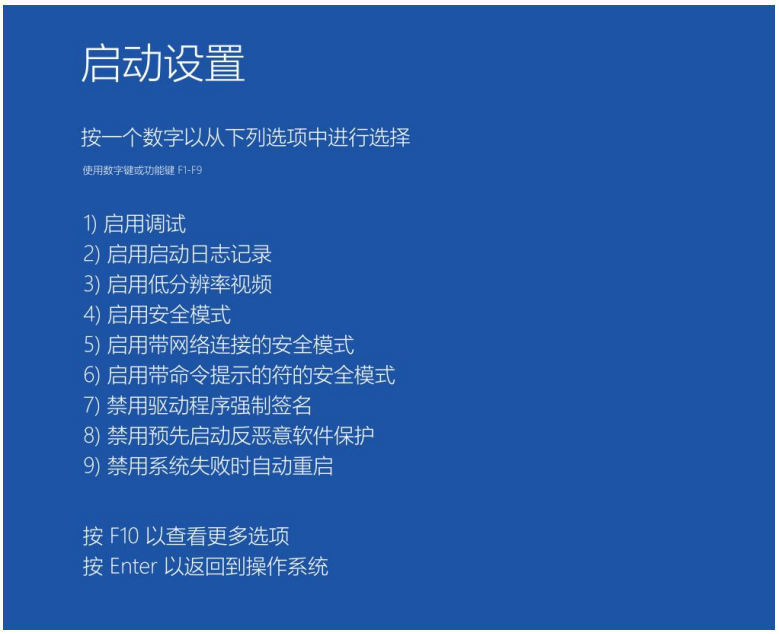


图 5-9

2.使用管理员权限运行“NIMC2000_D64.exe”开始安装驱动

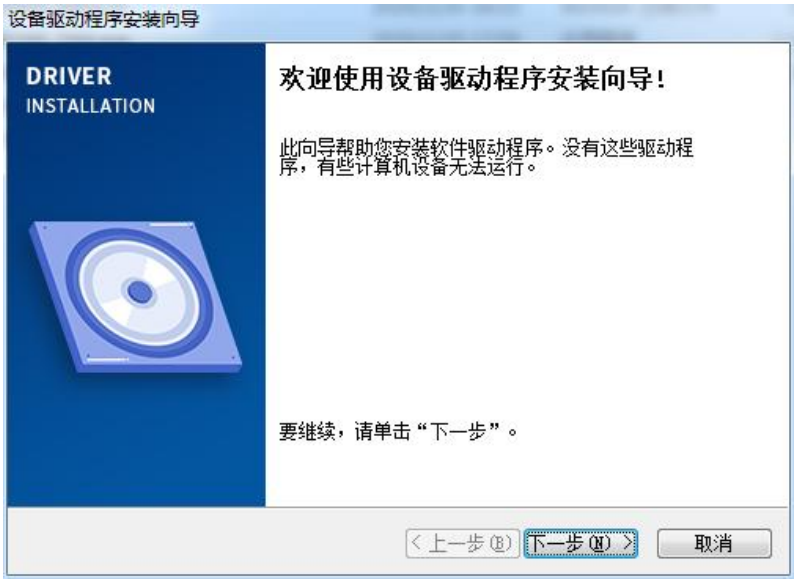


图 5-10

安装过程中会多出弹出如下提示框，选择“始终安装此驱动程序软件(I)”即可。

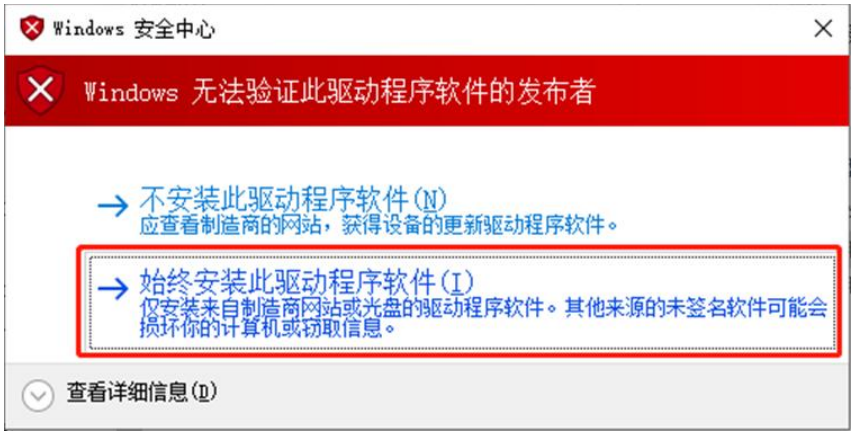


图 5-11

3.在 Windows10 下可能会出现系统不能正确识别本产品的情况，此时需要在设备管理器中设置默认的驱动程序，方法如下：

（1） 打开设备管理器，并找到识别错误的设备，在右键菜单中选择“更新驱动程序”：

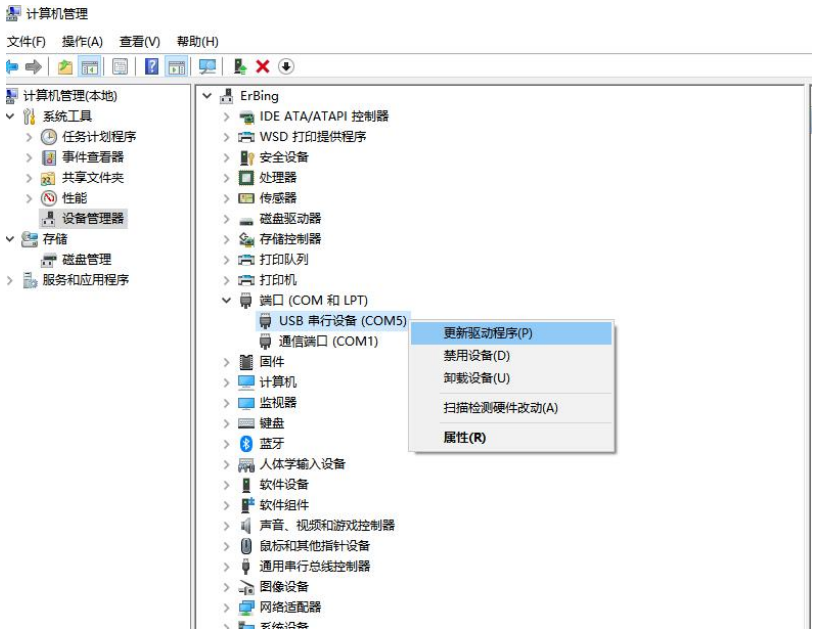


图 5-12

（2） 在弹出的窗口中选择“浏览我的计算机以查找驱动程序软件(R)”



图 5-13

（3） 接下来选择“让我从计算机上的可用程序列表中选取(L)”



图 5-14

（4） 然后选择“USB 串行设备”



图 5-15

（5） 完成后设备管理器中“网络适配器”下出现“Remote NDlS Compatible Device #28”设备说明驱动安装成功

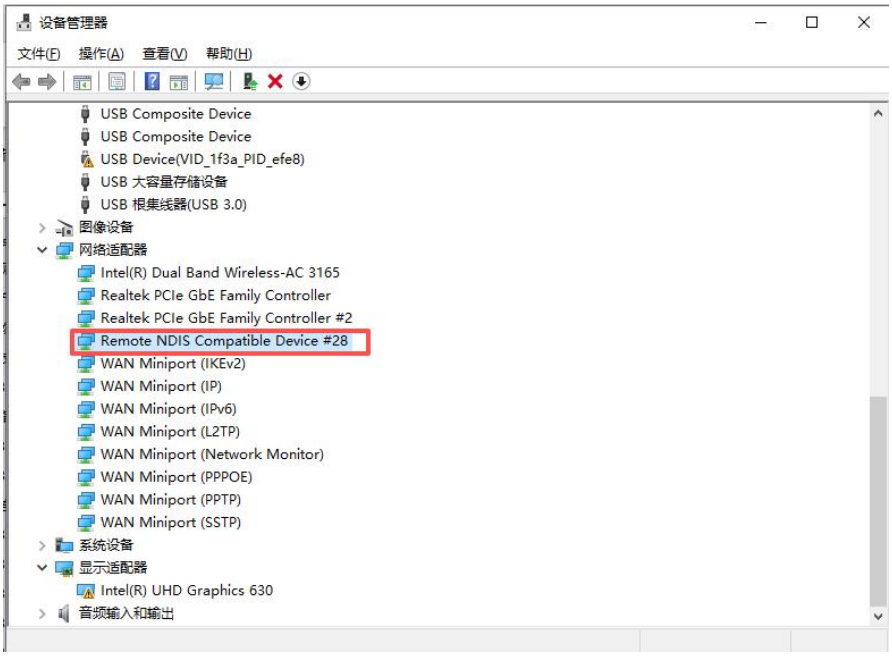


图 5-16

4.此时能 ping 通 192.168.7.2，说明本产品连接成功

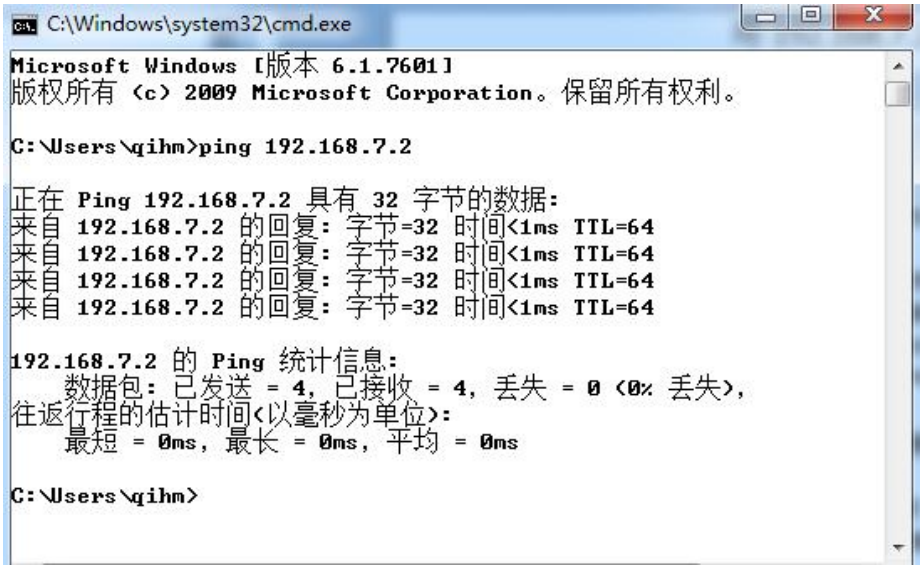


图 5-17

5.1.3 程序下载

1. 安装本产品的设备描述文件。

点击菜单“工具”->“设备存储库”，在弹出的“设备存储库”对话框中点击“安装”按钮，并选择本产品设备描述文件“NIMC2000-D_3.5.14.10.2.devdesc.xml”进行安装，如下图所示：

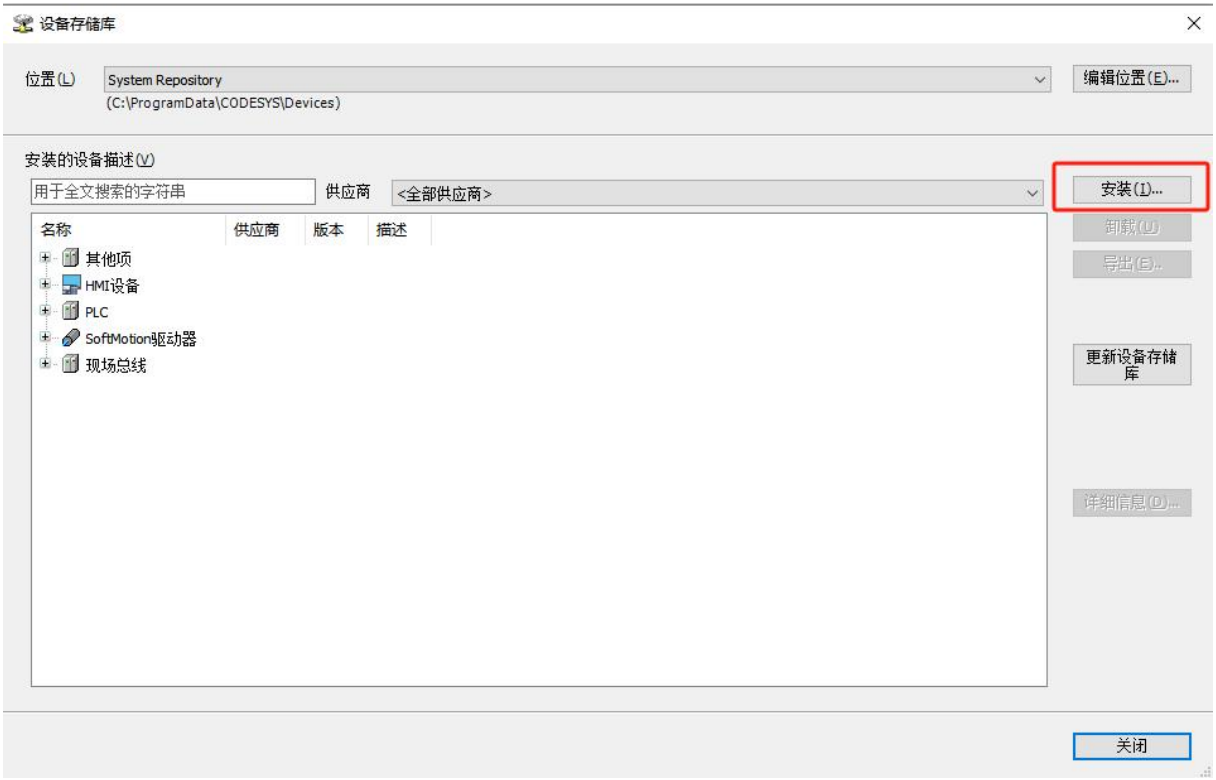


图 5-18

2. 打开或新建工程

3. 扫描并选择本产品设备

在左侧设备树上双击“Device (NIMC2000-D Controller)”并在右侧页面中点击“扫描网络”，然后在弹出的对话框中选择 NIMC2000 设备，点击“确认”按钮。如下图所示：

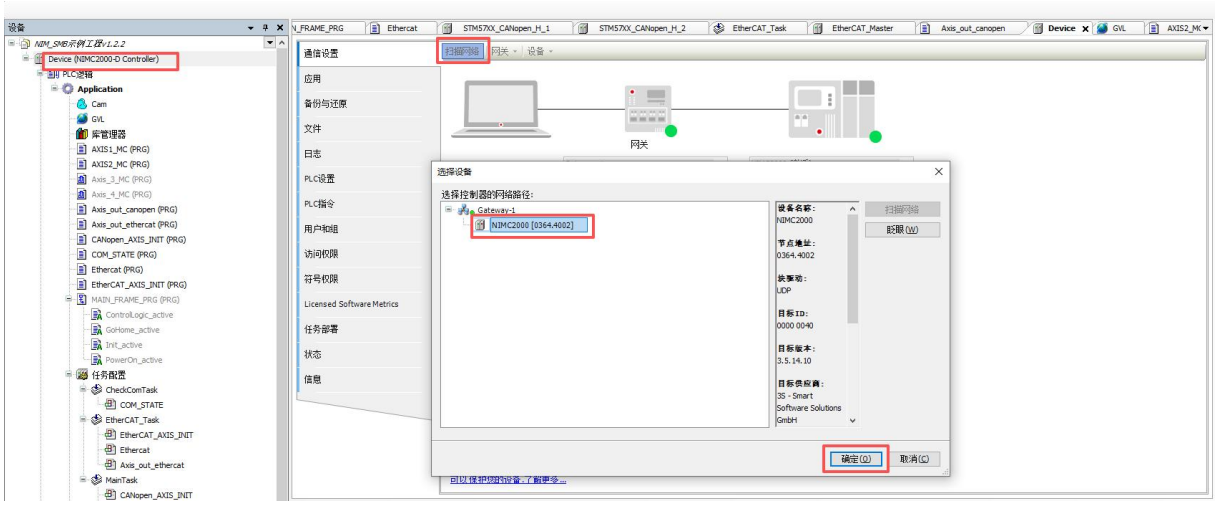


图 5-19

4. 下装并运行

点击菜单“在线”->“登录”登录本产品，这时会弹出如下对话框，点击“OK”开始下装。下装完成后点击菜单栏上的 按钮运行工程。

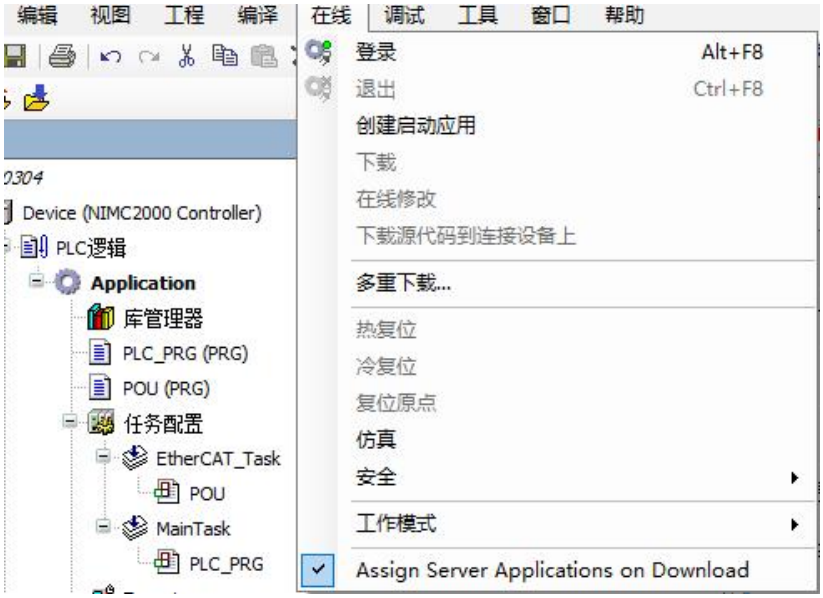


图 5-20

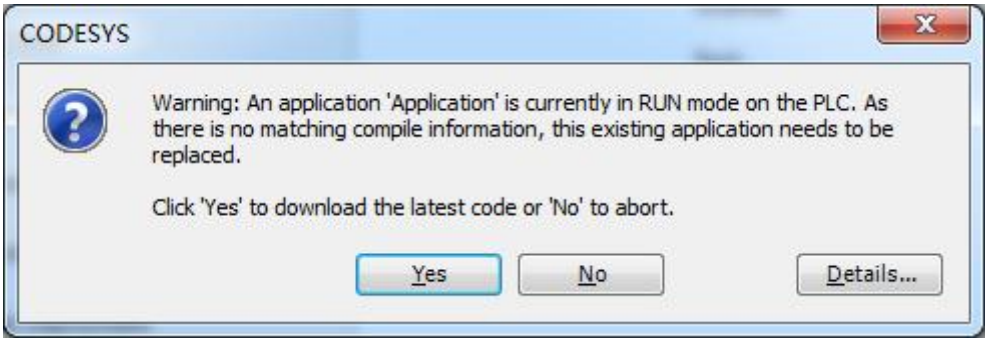


图 5-21

5.2 通信端口和通信协议配置

网关作为 Profinet 从站时，配置步骤参阅第 6.2 节 Profinet 从站配置，可以将数据转为 EtherCAT 主站、CANopen 主站、MODBUS TCP/RTU 主站，Ethernet/IP 主站。

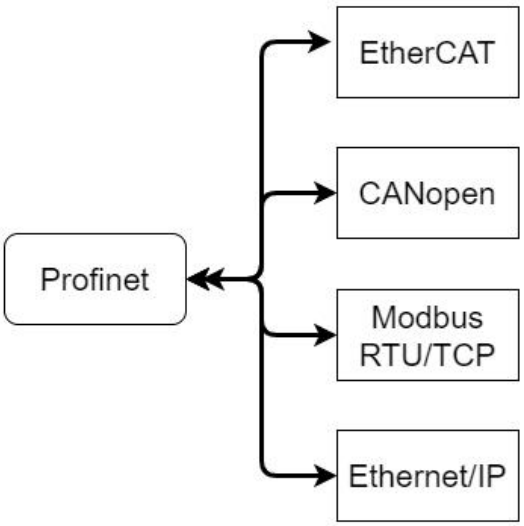


图 5-22

网关作为 MODBUS TCP/RTU 从站时，可以将数据转为 EtherCAT 主站、CANopen 主站、Profinet 主站，Ethernet/IP 主站。

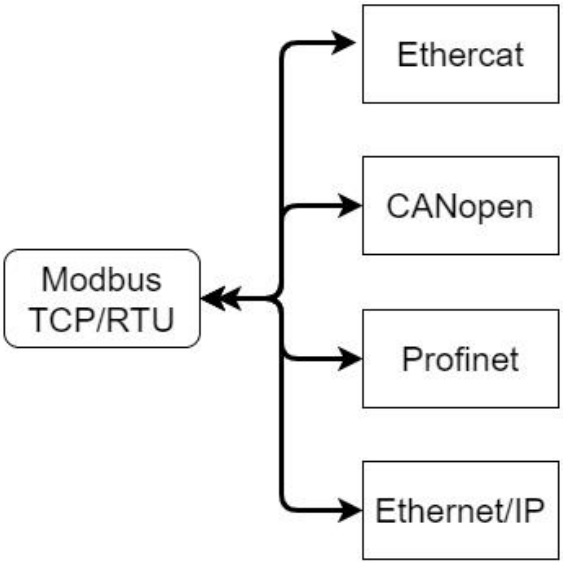


图 5-23

网关作为 OPCUA 服务端时，可以将数据转为 EtherCAT 主站、CANopen 主站、MODBUS TCP/RTU 主站，Profinet 主站。

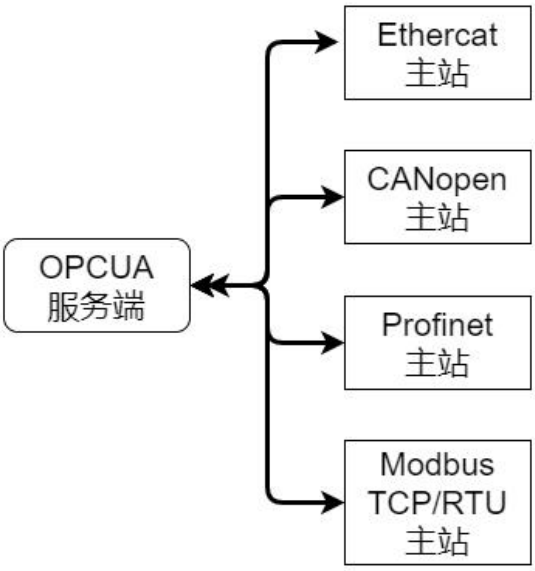


图 5-24

5.3 通信协议之间的变量映射

本产品搭载 CODESYS 编程系统，协议转换时需要手动将接收的变量映射到被转换的协议上。CODESYS 编程系统支持六种编程语言，本文以 ST 语言方式演示变量映射方法。

以 Profinet 转 CANopen 为例。网关此时作为 Profinet 从站和 CANopen 主站，需要将 Profinet 接收到数据映射到 CANopen 上。

网关产品作为 Profinet 从站，配置步骤参阅第 6.2 节 Profinet 从站配置章节描述的具体内容；配置完成后如下图所示。



图 5-25

网关作为 CANopen 主站，配置步骤参阅第 8.1 节 CANopen 主站配置描述内容；配置完成后如下图所示，CANopen 主站下连接一台 CANopen 电机。

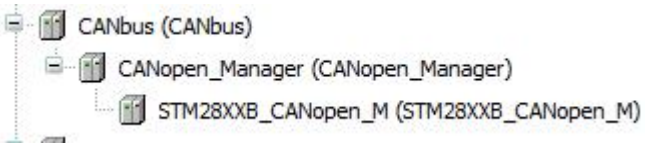


图 5-26

首先获取 Profinet 从站中定义的变量地址，在 CODESYS 对变量的定义中，%Q 的地址代表输出地址，即 Profinet 主站从网关产品中读取的数据（网关向 Profinet 主站发送的数据），%I 的地址代表输入的地址，即 Profinet 主站向网关产品发送的数据（网关从 Profinet 主站接收的数据）。

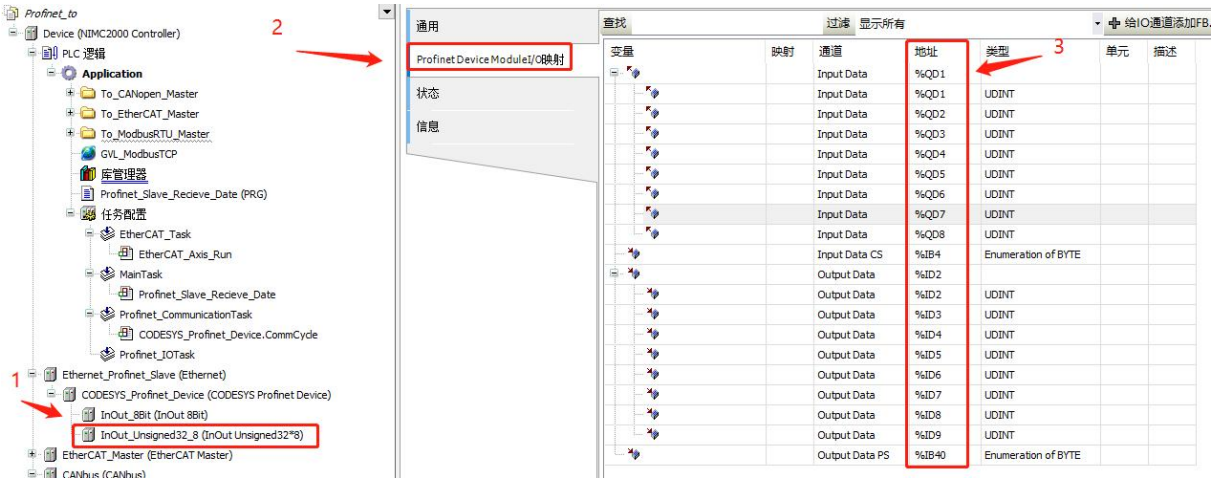


图 5-27

将地址映射到变量中。

```
VAR_GLOBAL
    Data_Input_0:UDINT;
    Data_Input_1:Udint;
    Data_Input_2:Udint;
    Data_Input_3:Udint;
    Data_Input_4:Udint;
    Data_Input_5:Udint;
    Data_Input_6:Udint;
    Data_Input_7:Udint;
    Data_Input_8:Udint;
    Data_Input_9:Udint;

    Data_Output_0:UDINT;
    Data_Output_1:Udint;
    Data_Output_2:Udint;
    Data_Output_3:Udint;
    Data_Output_4:Udint;
    Data_Output_5:Udint;
    Data_Output_6:Udint;
    Data_Output_7:Udint;
    Data_Output_8:Udint;
    Data_Output_9:Udint;
END_VAR
```

```
(* 通过地址获取接收到指令*)
Data_Input_0:=%ID0;
Data_Input_1:=%ID1;
Data_Input_2:=%ID2;
Data_Input_3:=%ID3;
Data_Input_4:=%ID4;
Data_Input_5:=%ID5;
Data_Input_6:=%ID6;
Data_Input_7:=%ID7;
Data_Input_8:=%ID8;
Data_Input_9:=%ID9;

%QD0:=Data_Output_0;
%QD1:=Data_Output_1;
%QD2:=Data_Output_2;
%QD3:=Data_Output_3;
%QD4:=Data_Output_4;
%QD5:=Data_Output_5;
%QD6:=Data_Output_6;
%QD7:=Data_Output_7;
%QD8:=Data_Output_8;
%QD9:=Data_Output_9;
```

图 5-28

获取 CANopen 主站中需要映射的从站寄存器地址。如下图所示。

双击需要映射的 CANopen 设备；选择“CANopen IO 映射”选项卡；选择要映射的寄存器地址。

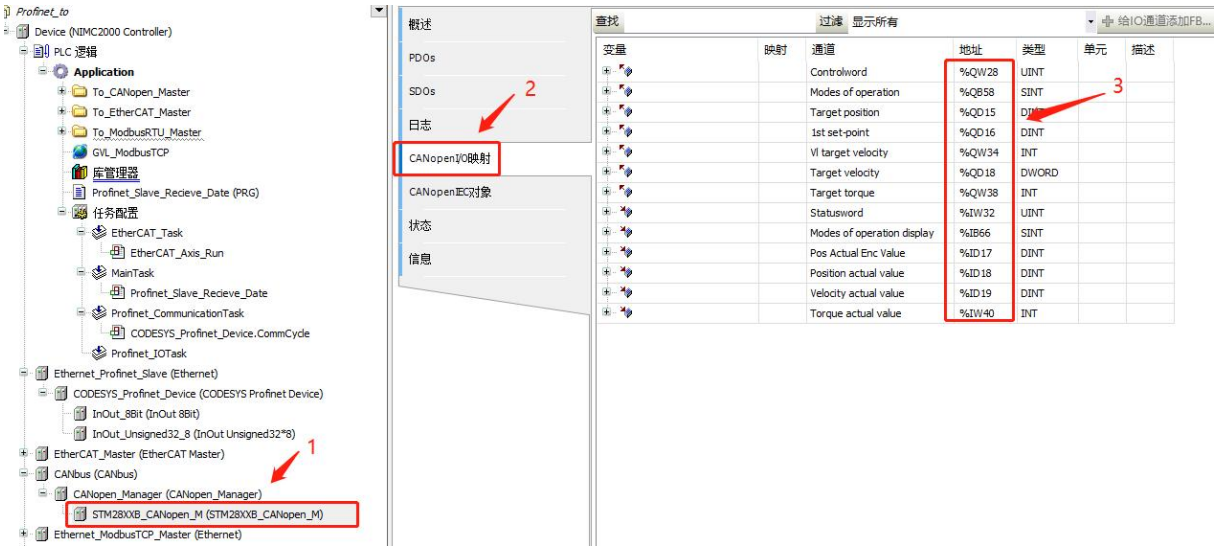


图 5-29

通过 ST 语言的方式将 Profinet 从站接收的数据直接映射到控制的 CANopen 主站上。

```
//直接写电机的控制字
%QW28:=UDINT_TO_UINT(Data_Input_0);

//直接写电机的控制模式
%QB58:=UDINT_TO_BYTE(Data_Input_1);

//直接写电机目标位置
%QD15:=UDINT_TO_DWORD(Data_Input_2);


// 读取电机的状态字
Data_Output_0:=%IW32;

// 读取电机的控制模式
Data_Output_1:=%IB66;

// 读取电机的实际位置
Data_Output_2:=%ID18;
```

图 5-30

5.4 运动控制功能

网关产品对于 CANopen、EtherCAT、MODBUS RTU 等协议的电机支持立迈胜自研轻量轴运动控制库支持，对于 CANopen、EtherCAT 等协议的电机支持立迈胜自研标准轴、机器人运动控制库。

6 Profinet 配置

6.1 Profinet 主站配置

6.1.1 Profinet设备描述文件添加

① 选择 Codesys 菜单栏->【工具】；

② 选择【设备存储库】选项卡；

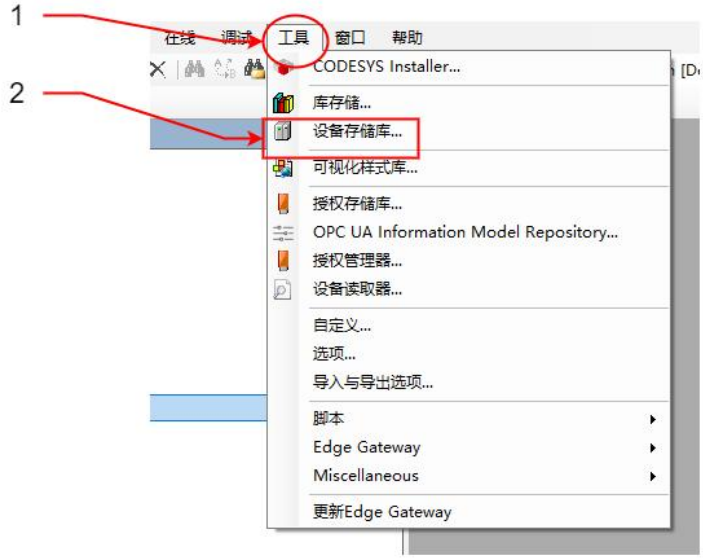


图 6-1

③ 打开设备存储库页面后，选择【安装】；

④ 指定文件格式为【PROFINET IO 配置文件】；

⑤ 选择设备描述文件，然后点击“打开”；

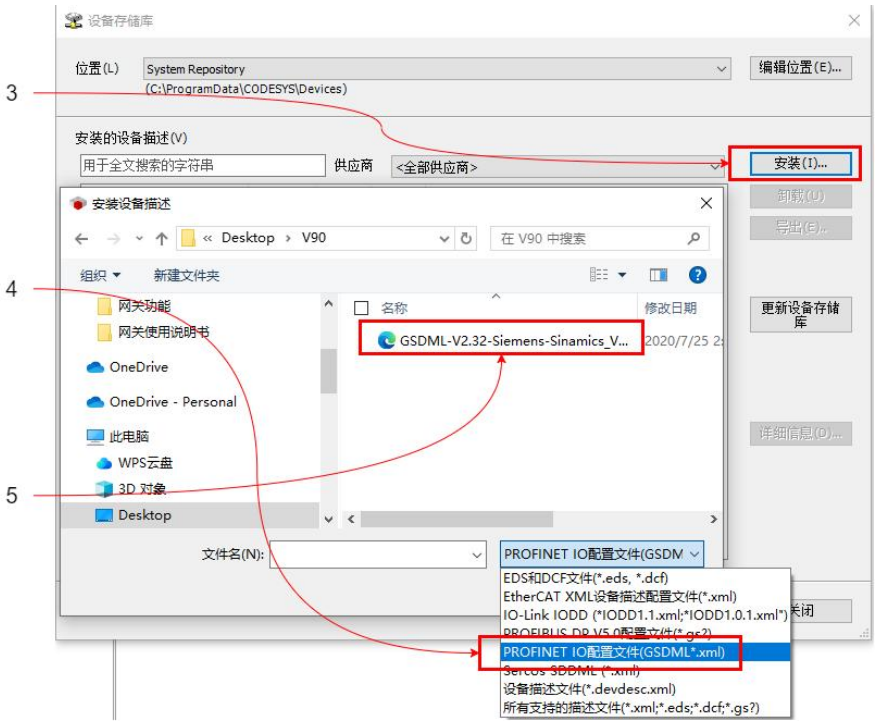


图 6-2

⑥ 添加完毕后提示栏显示【设备已经安装到设备存储库】；

⑦ 点击【关闭】，关闭设备存储库页面。

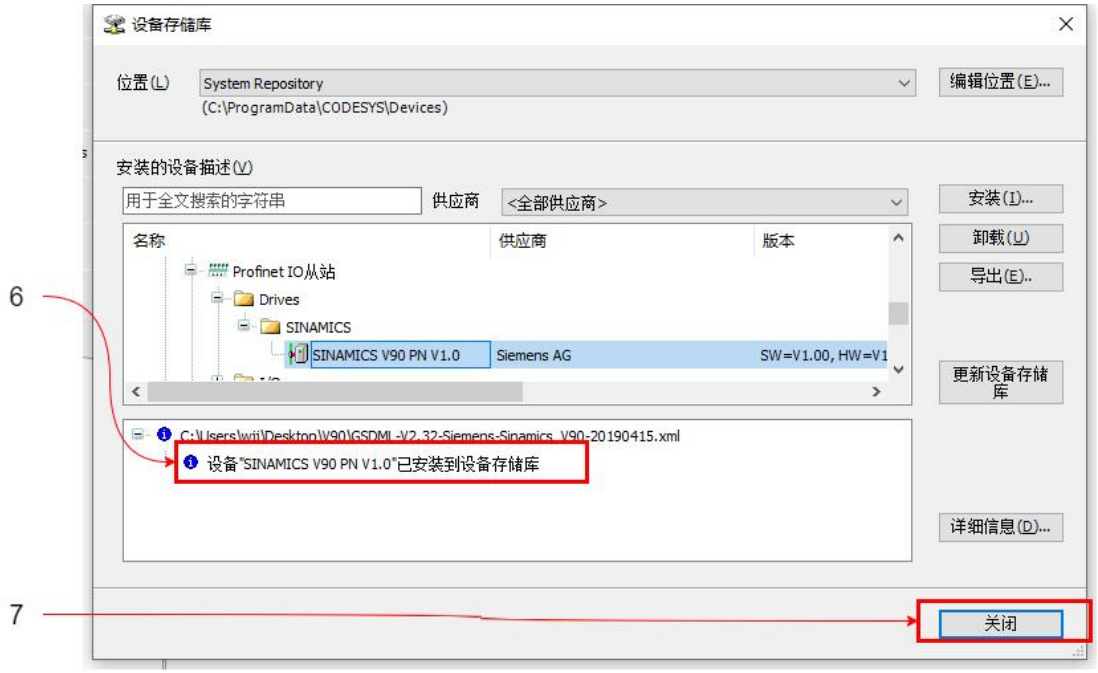


图 6-3

6.1.2 Profinet主站配置

① 鼠标选中 Device 设备，右键单击，弹出属性；

② 选择【添加设备】；

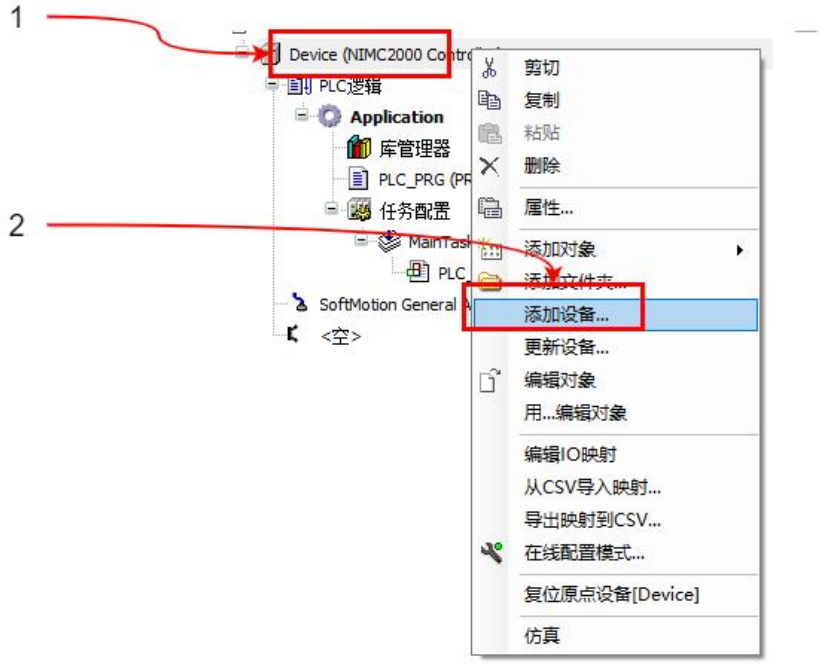


图 6-4

③ 选择现场总线-》Profinet IO-》以太网适配器-》【Ethernet】设备；

④ 点击【添加设备】按钮，添加 Profinet 主站所使用的 Ethernet 网卡；

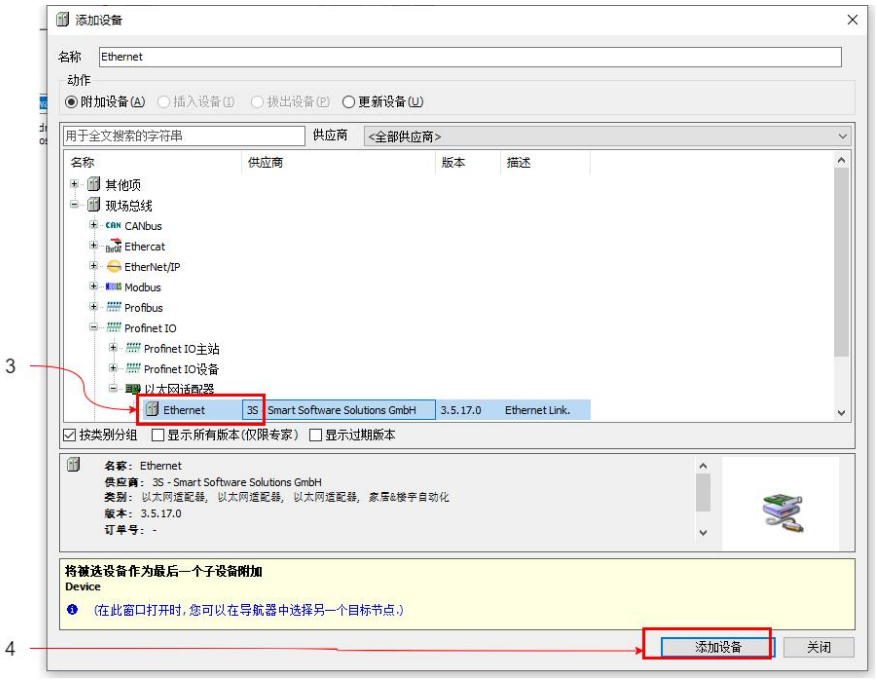


图 6-5

⑤ 鼠标双击【Ethernet】设备，右侧弹出设备设置页面；

⑥ 在通用栏中，选择网络接口后面的【Browse】按钮，弹出网络适配器选择页面；

⑦ 选择使用的网卡；

⑧ 点击【确认】按钮，网卡选择完毕；

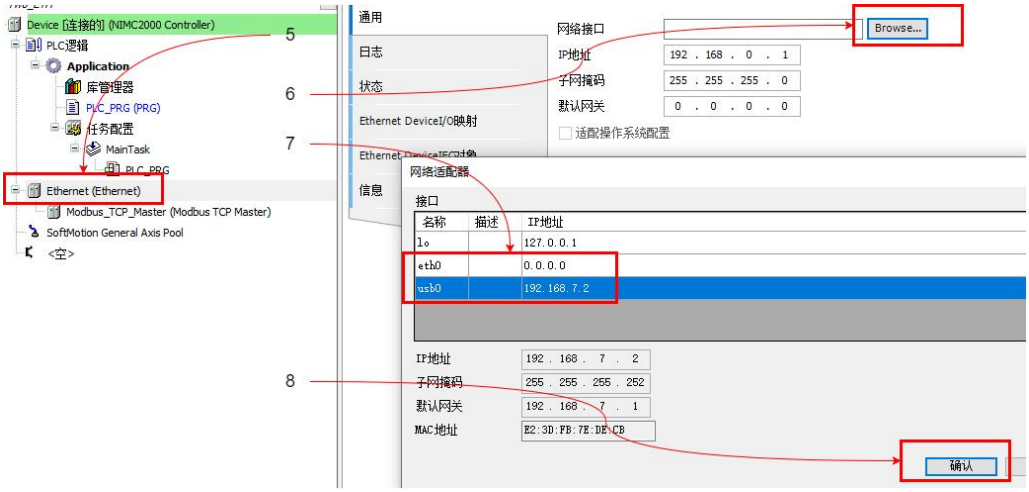


图 6-6

⑨ 鼠标选中 Ethernet 设备，右键单击，弹出属性；

⑩ 选择【添加设备】，在添加设备页面选择 Profinet 主站设备；

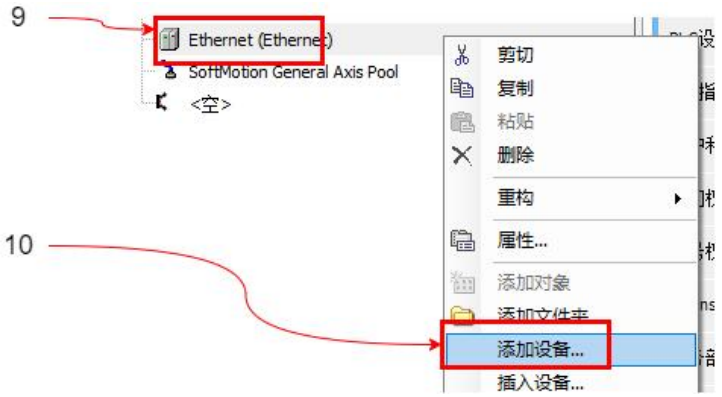


图 6-7

⑪ 选择现场总线-》Profinet IO-》Profinet IO 主站-》【PN-Controller】设备；

⑫ 点击【添加设备】按钮，添加选中的 Profinet 主站设备；

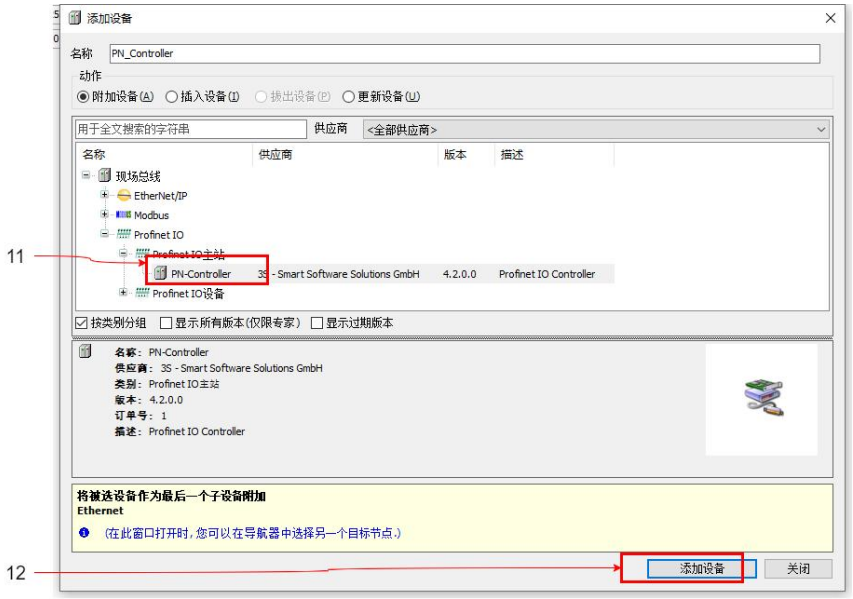


图 6-8

⑬ 鼠标双击【PN-Controller】设备，右侧弹出设备设置页面；

⑭ 设置 Profinet 从站 IP 参数；



图 6-9

⑮ 鼠标选中 Device 设备，右键单击，弹出属性；

⑯ 选择【添加设备】，添加需要的 Profinet 从站；

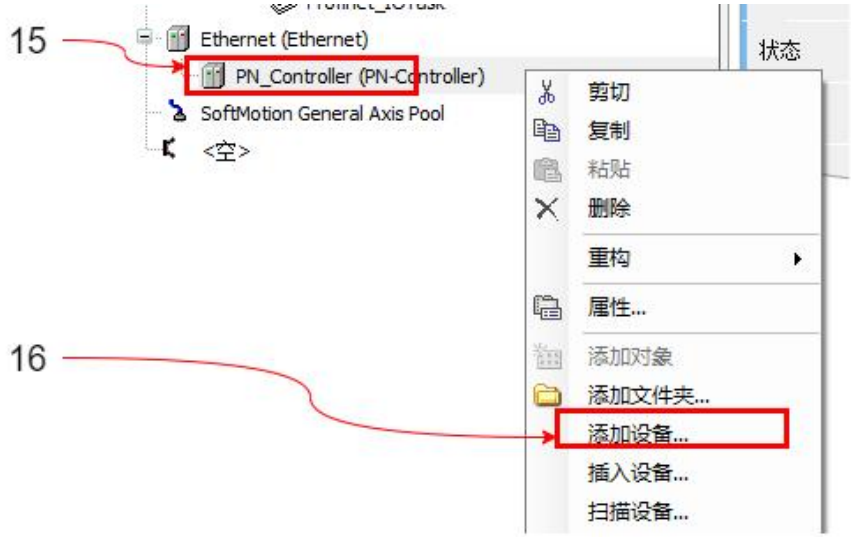


图 6-10

6.2 Profinet 从站配置

6.2.1 Profinet从站添加

① 鼠标选中 Device 设备，右键单击，弹出属性；

② 选择【添加设备】；

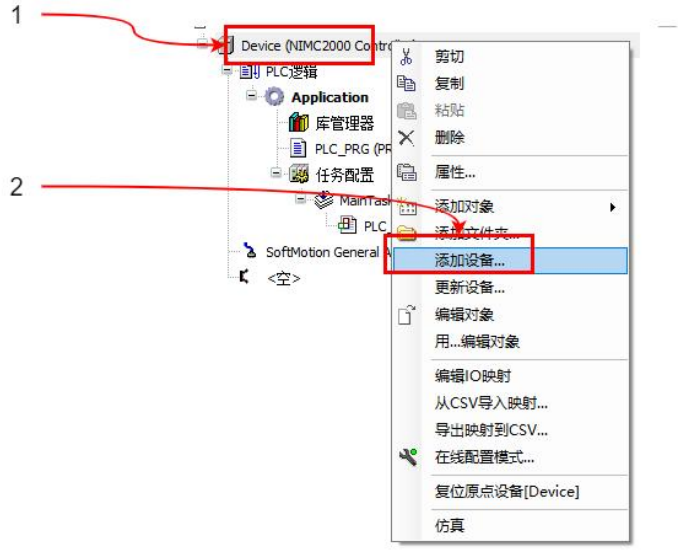


图 6-11

③ 选择现场总线-》Profinet IO-》以太网适配器-》【Ethernet】设备；

④ 点击【添加设备】按钮，添加 Profinet 主站所使用的 Ethernet 网卡；

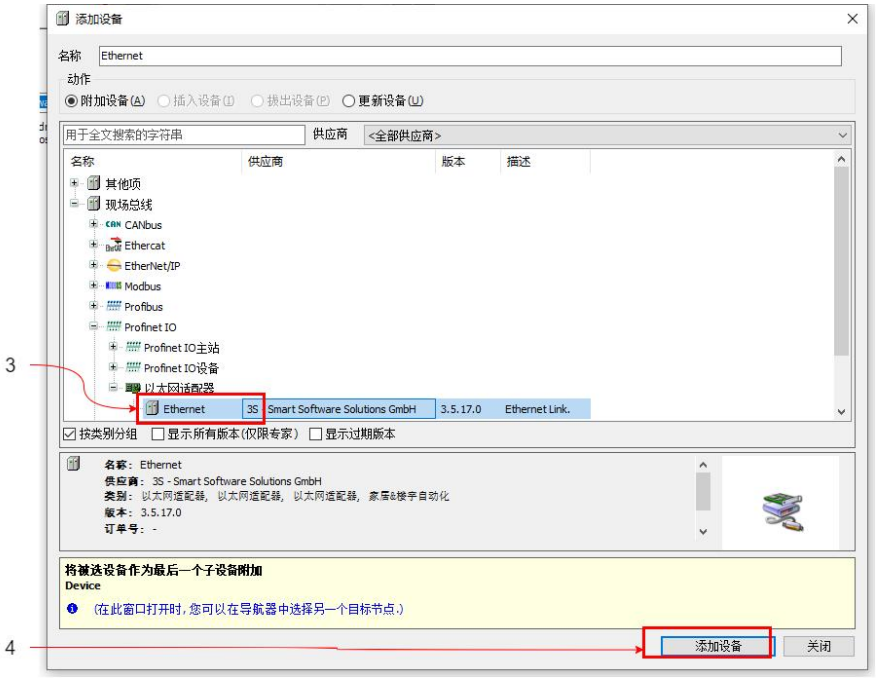


图 6-12

⑤ 鼠标双击【Ethernet】设备，右侧弹出设备设置页面；

⑥ 在通用栏中，选择网络接口后面的【Browse】按钮，弹出网络适配器选择页面；

⑦ 选择使用的网卡；

⑧ 点击【确认】按钮，网卡选择完毕；

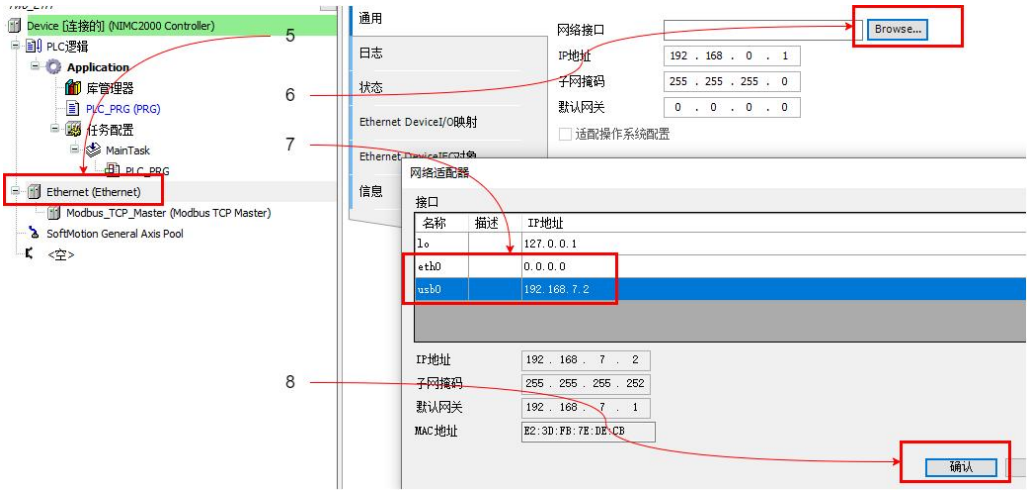


图 6-13

⑨ 鼠标选中 Ethernet 设备，右键单击，弹出属性；

⑩ 选择【添加设备】，在添加设备页面选择 Profinet 主站设备；

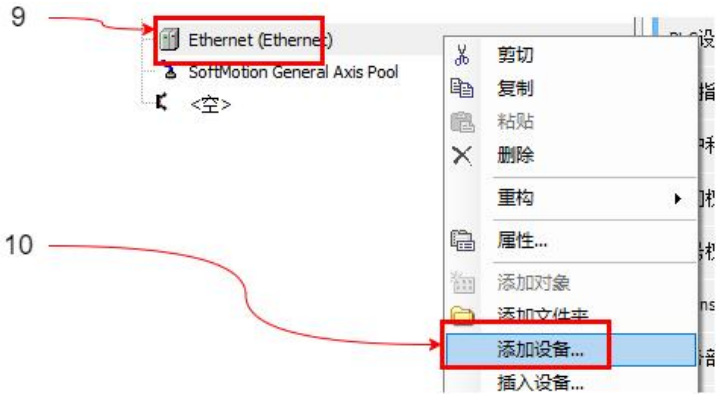


图 6-14

⑪ 选择现场总线-》Profinet IO-》Profinet IO 设备-》【CODESYS Profinet Device】设备；

⑫ 点击【添加设备】按钮，添加选中的 Profinet 从站设备；

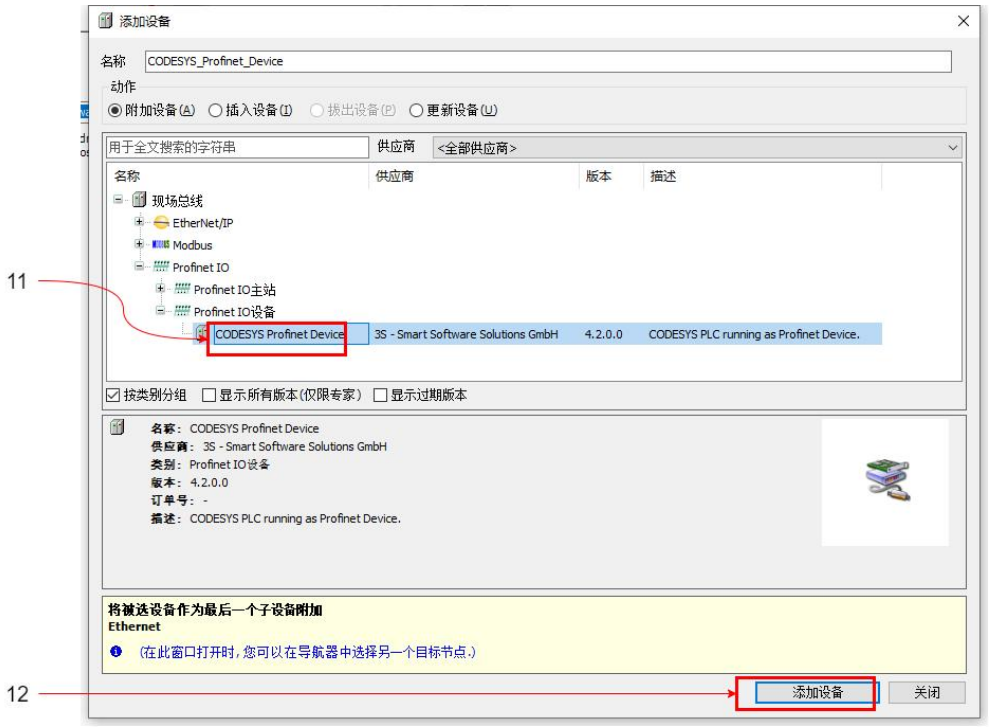


图 6-15

⑬ 鼠标选中 CODESYS Profinet Device 设备，右键单击，弹出属性；

⑭ 选择【添加设备】，添加需要的 Profinet 从站映射寄存器；

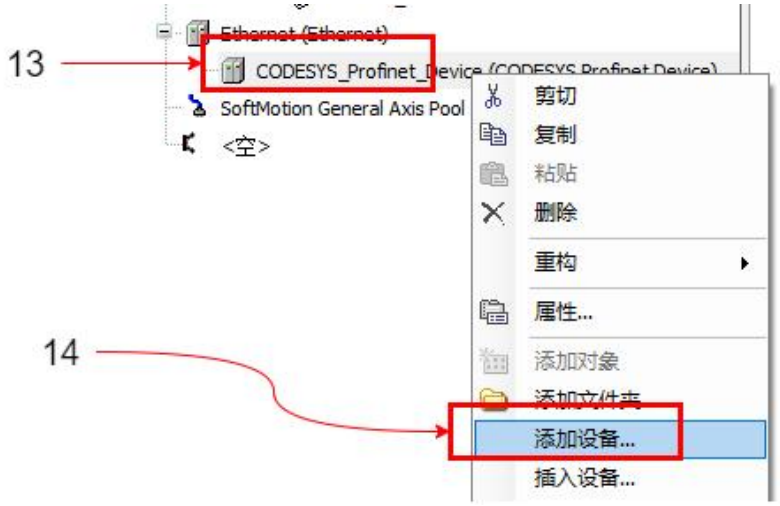


图 6-16

⑮ 选择现场总线-》Profinet IO-》Profinet IO 模块-》【Input/Output】设备；

⑯ 选择【添加设备】，添加需要的 Profinet 从站映射寄存器。

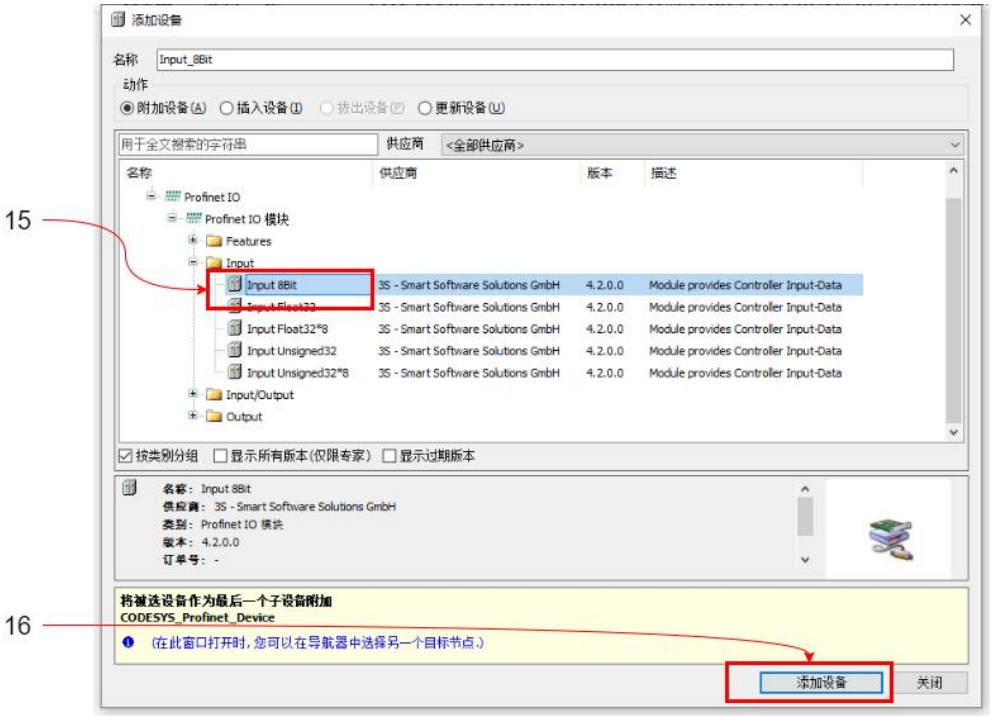


图 6-17

6.2.2 Profinet从站描述文件导出

① 选择 Codesys 菜单栏->【工具】；

② 选择【设备存储库】选项卡；

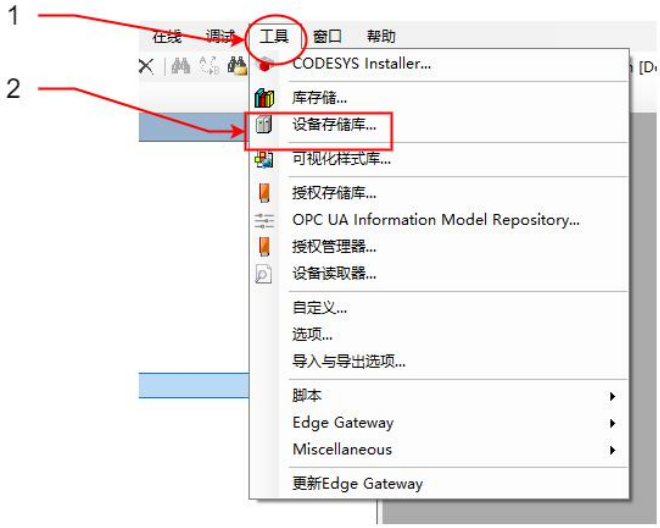


图 6-18

③ 选择现场总线-》Profinet IO-》Profinet IO 从站-》【CODESYS Profinet Device】设备；

④ 点击【导出】按钮，导出选中的 Profinet 从站设备描述文件压缩包，解压导出的压缩包后可以找到 Profinet 从站描述文件；

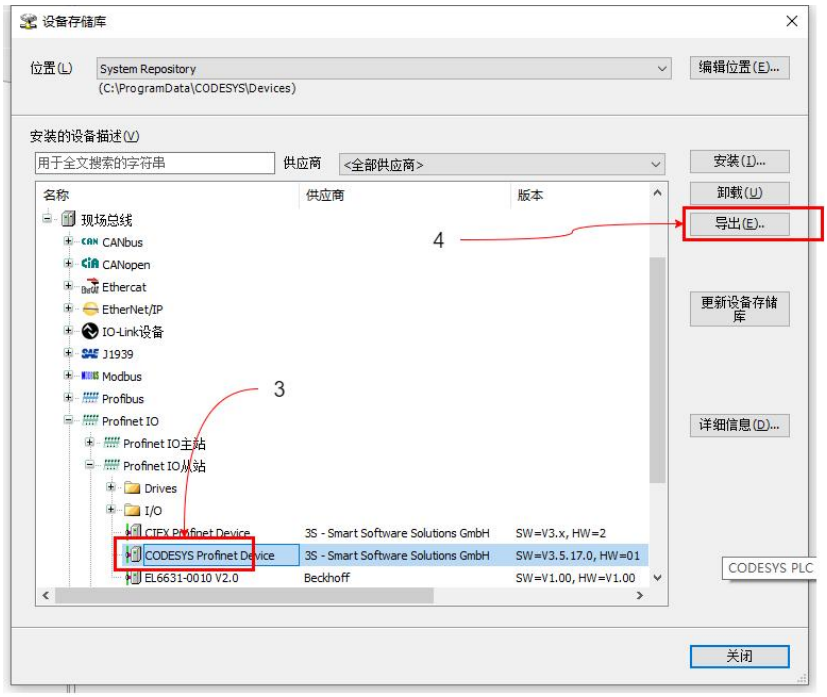


图 6-19

7 Ethercat 配置

7.1 Ethercat 设备描述文件添加

① 选择 Codesys 菜单栏中的“工具”选项卡；

② 选择“设备存储库”选项；

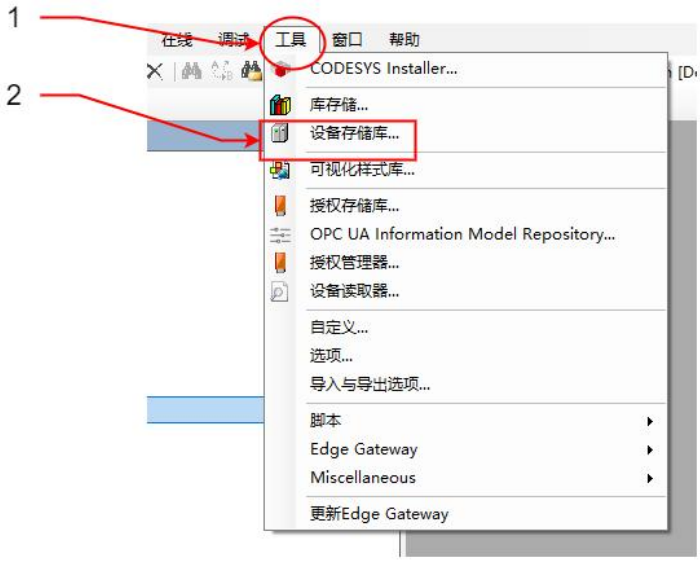


图 7-1

③ 打开设备存储库页面后，选择“安装”；

④ 指定文件格式为“Ethercat XML 设备描述文件”；

⑤ 选择设备描述文件，然后点击“打开”；

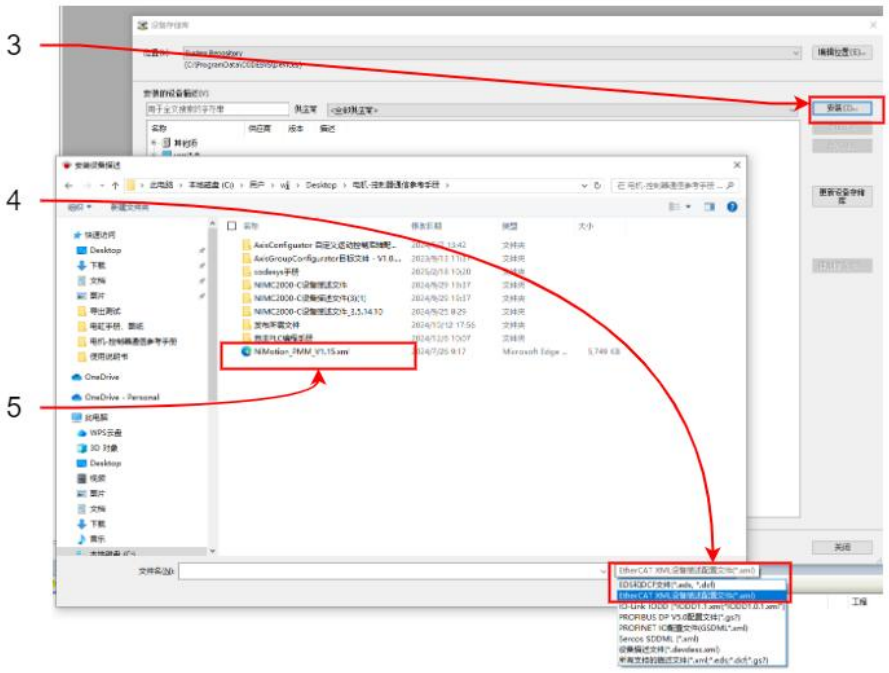


图 7-2

⑥ 安装完成后，设备存储库页面显示设备添加到设备存储库；

⑦ 点击关闭，设备添加完成。

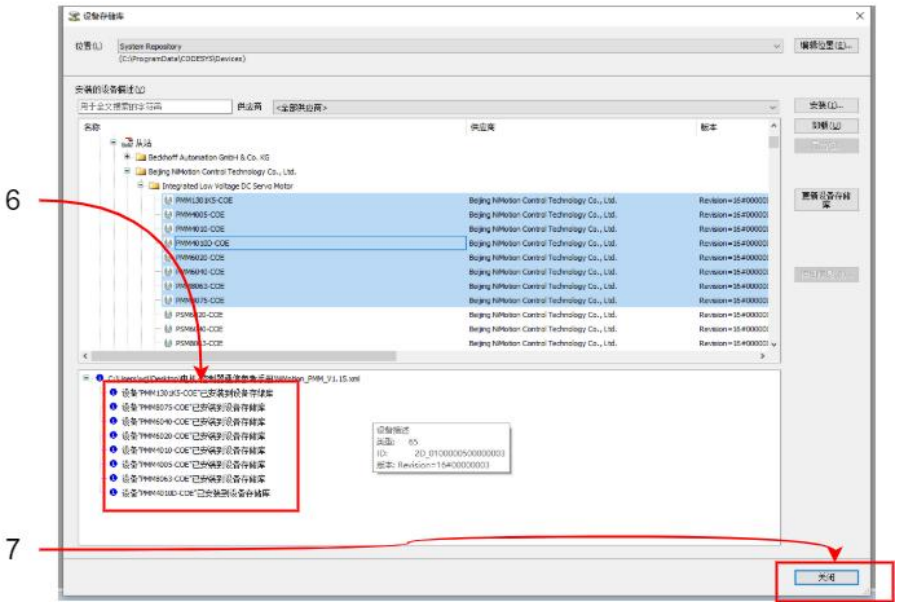


图 7-3

7.2 EtherCAT 主站配置

① 鼠标选中 Device 设备，右键单击，弹出属性；

② 选择【添加设备】；

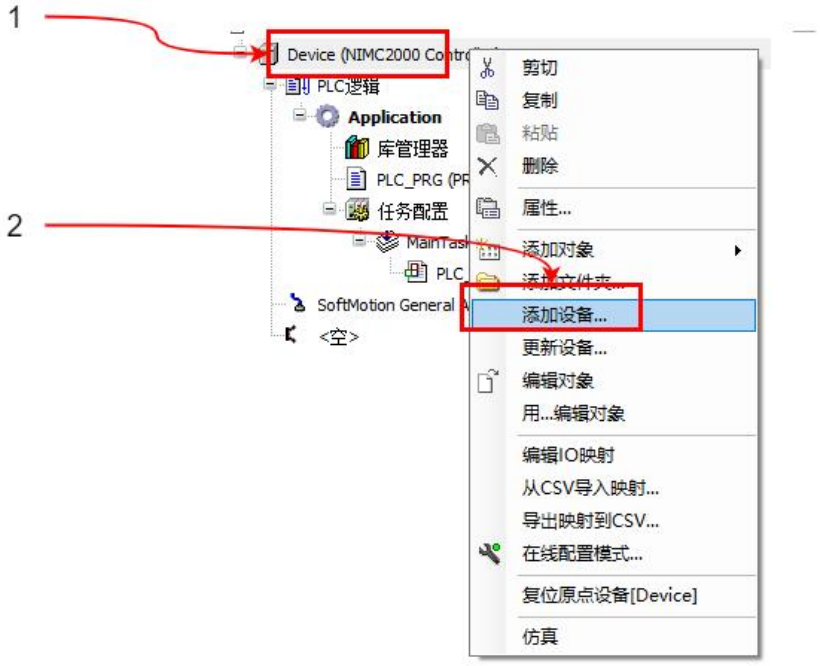


图 7-4

③ 选择现场总线-》Ethercat-》主站-》【EtherCAT Master】设备；

④ 点击【添加设备】按钮，添加 EtherCAT 主站；

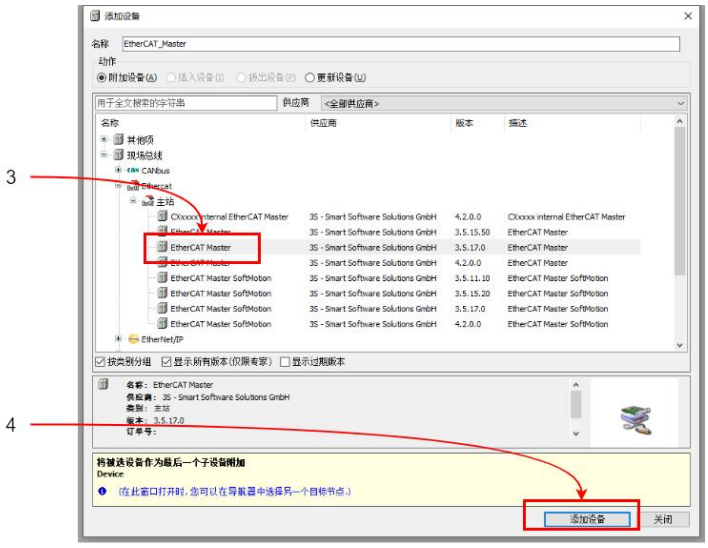


图 7-5

⑤ 鼠标双击【EtherCAT Master】设备，右侧弹出设备设置页面；

⑥ 在通用栏中，选择网络接口后面的【浏览】按钮，弹出网络适配器选择页面；

⑦ 选择使用的网卡；

⑧ 点击【确认】按钮，网卡选择完毕；

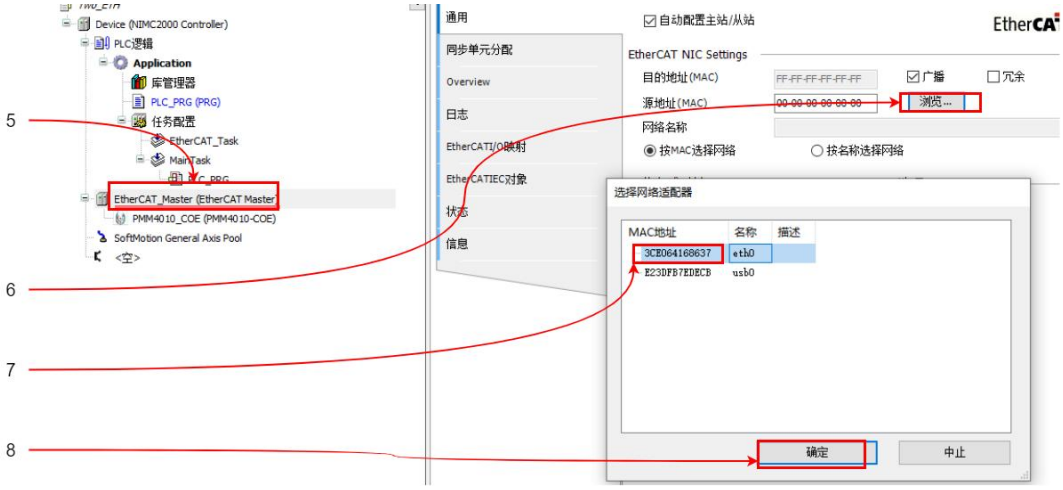


图 7-6

7.3 自动扫描添加 EtherCAT 设备

① 鼠标选中 EtherCAT Master 设备，右键单击，弹出属性；

② 选择【扫描设备】；

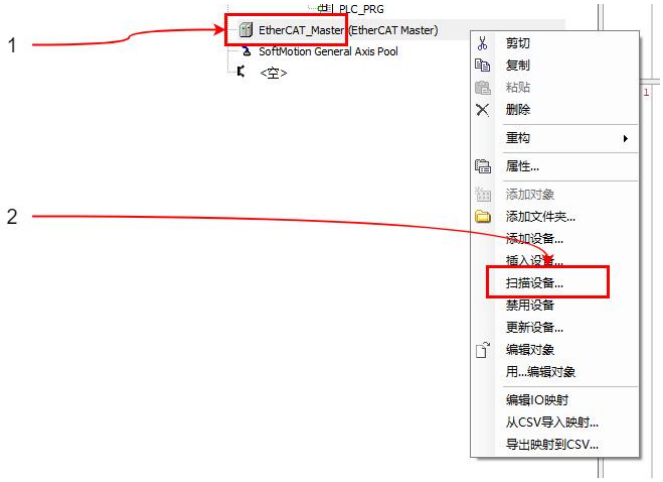


图 7-7

③ 在扫描设备页面，点击【扫描设备】按钮；

④ 点击【复制所有设备到工程】按钮；

⑤ 点击【关闭】按钮；



图 7-8

7.4 手动添加 EtherCAT 设备

① 鼠标选中 EtherCAT Master 设备，右键单击，弹出属性；

② 选择【添加设备】；

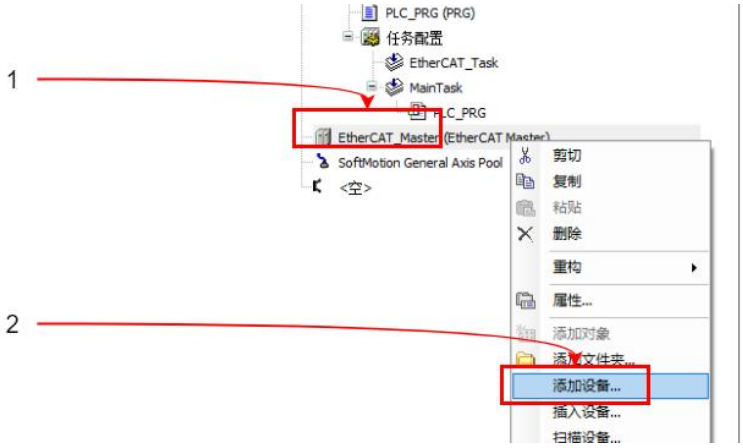


图 7-9

③ 选择现场总线-》Ethercat-》从站，选择【要添加的从站设备】设备；

④ 点击【添加设备】按钮，添加选择的从站；

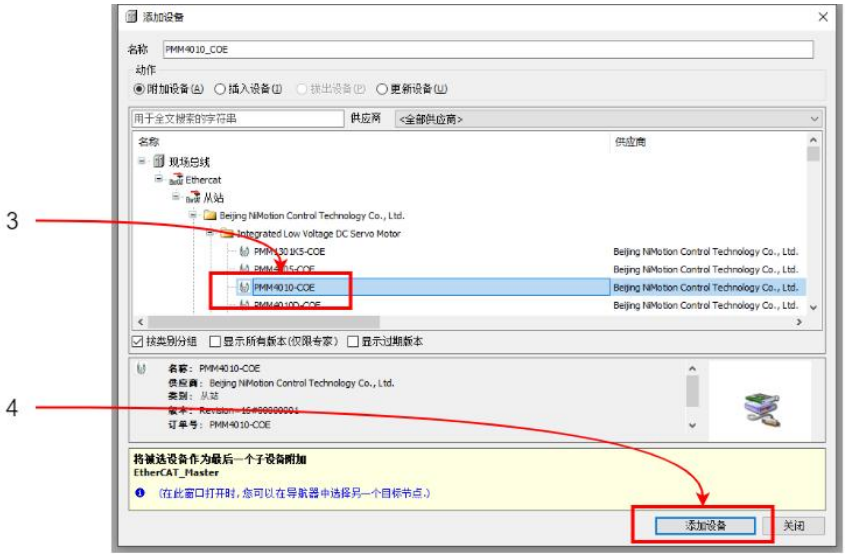


图 7-10

7.5 CANopen 主站配置

7.5.1 添加CANopen设备描述文件

① 选择 Codesys 菜单栏->【工具】；

② 选择【设备存储库】选项卡；

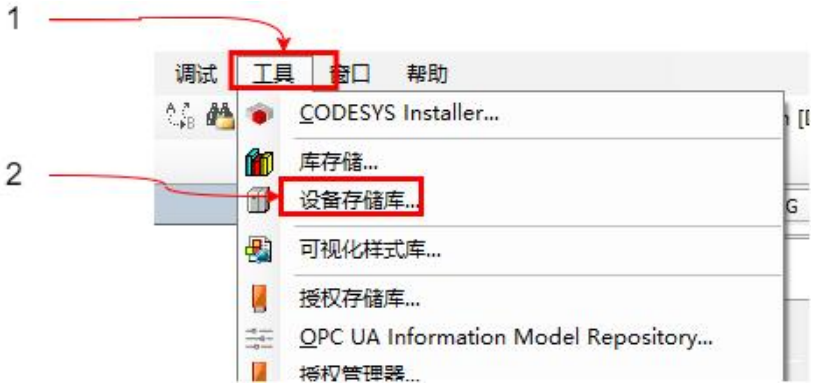


图 7-11

③ 打开设备存储库页面后，选择“安装”；

④ 指定文件格式为“EDS 设备描述文件”；

⑤ 选择设备描述文件，然后点击“打开”；

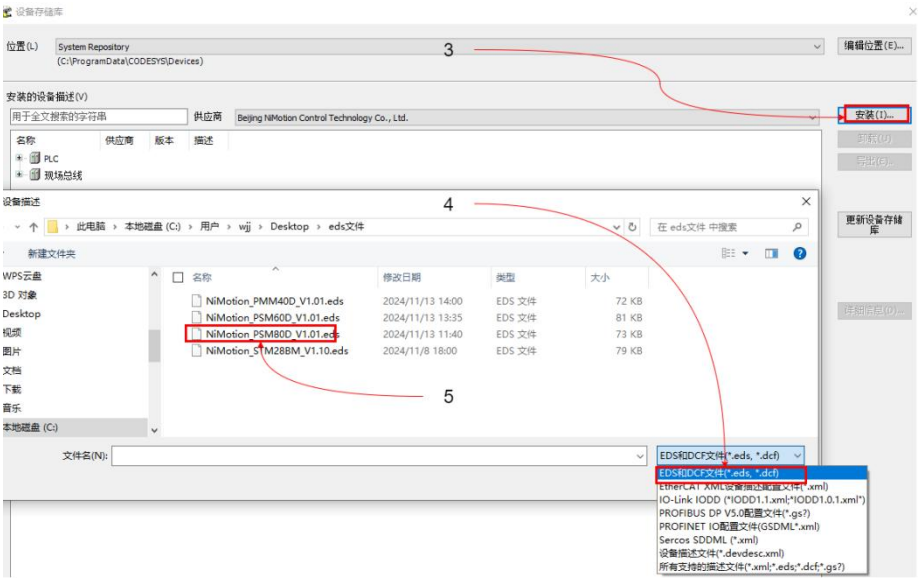


图 7-12

⑥ 安装完成后，设备存储库页面显示设备添加到设备存储库；

⑦ 点击关闭，设备添加完成。

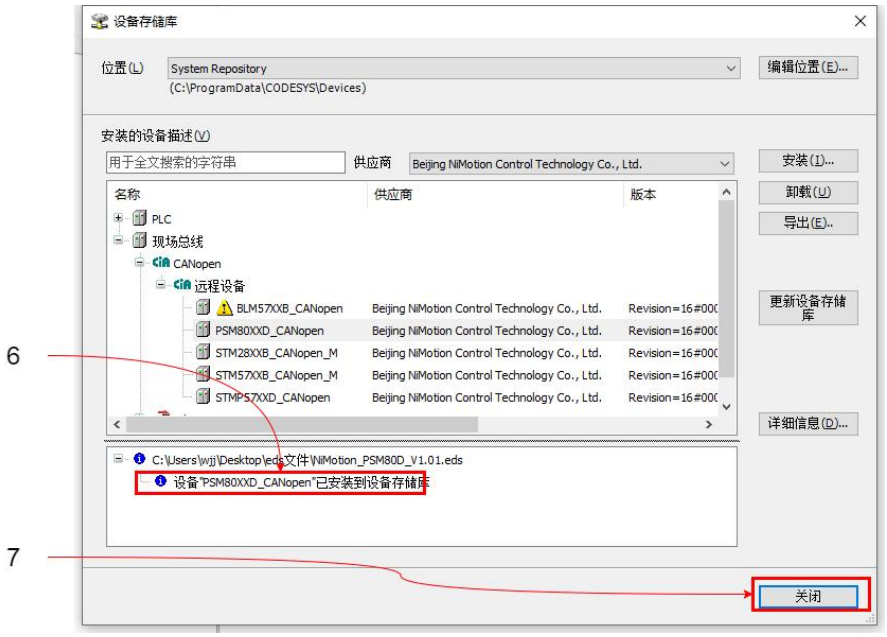


图 7-13

7.5.2 添加CANopen主站

① 鼠标选中 Device 设备，右键单击，弹出属性；

② 选择【添加设备】；



图 7-14

③ 选择添加设备页面的“附加设备”；

④ 选择现场总线-》CANbus-；

⑤ 选择【CANbus】设备；

⑥ 点击【添加设备】按钮，添加 CANopen 主站；

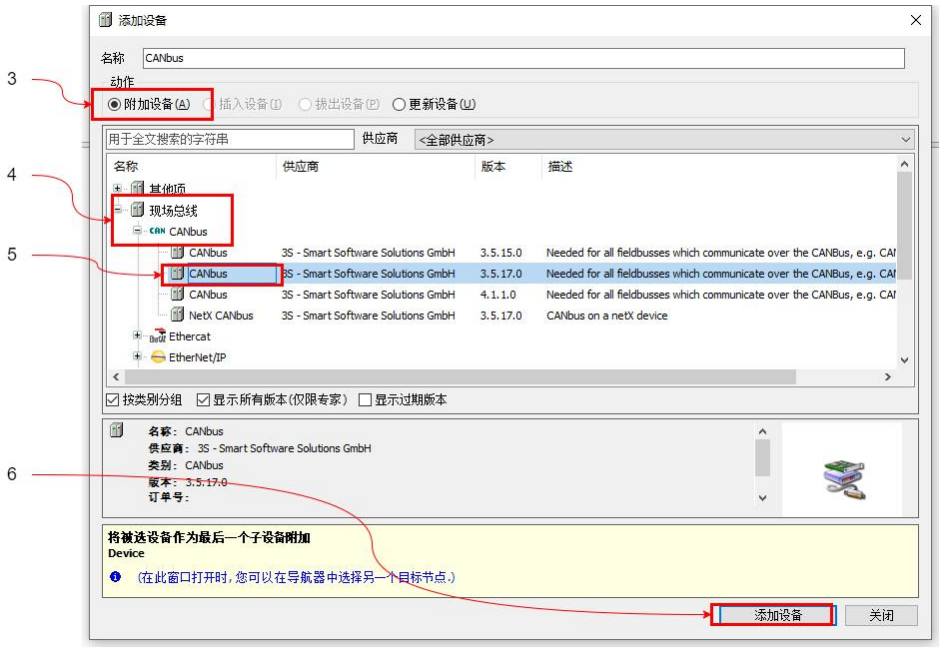


图 7-15

⑦ 鼠标双击【CANbus】设备，右侧弹出设备设置页面；

⑧ 在通用栏中，设置使用的 CAN 网络号和 CAN 波特率；

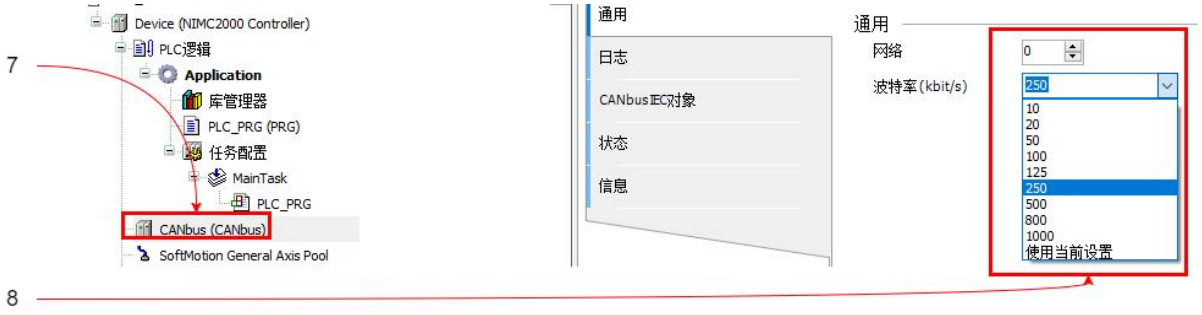


图 7-16

⑨ 鼠标右击【CANbus】设备，右侧弹出属性选择框；

⑩ 点击【添加设备】按钮，添加 CANopen Manger 管理器；



图 7-17

⑪ 选择现场总线-》CANopen-》CANopen 管理器；

⑫ 选择【CANopen Manger】设备；

⑬ 点击【添加设备】按钮，添加 CANopen Manger 管理器。

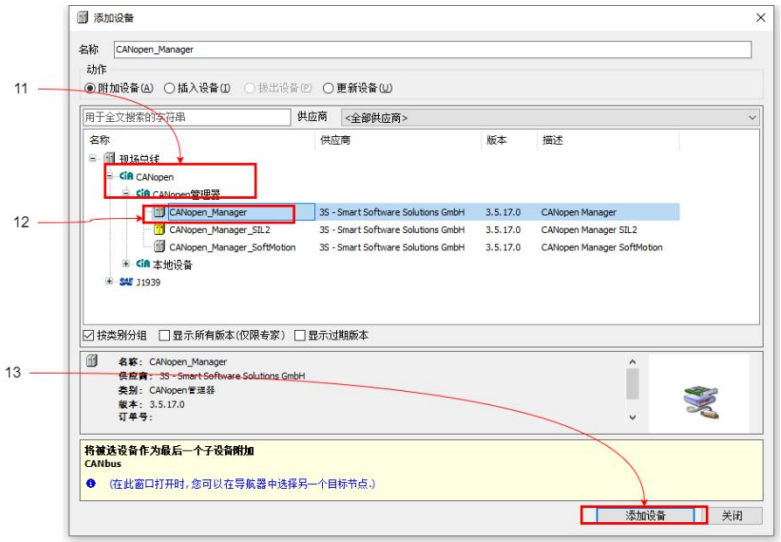


图 7-18

7.5.3 自动添加CANopen设备

⑥ 鼠标选中 CANopen Manger 设备，右键单击，弹出属性；

⑦ 选择【扫描设备】；

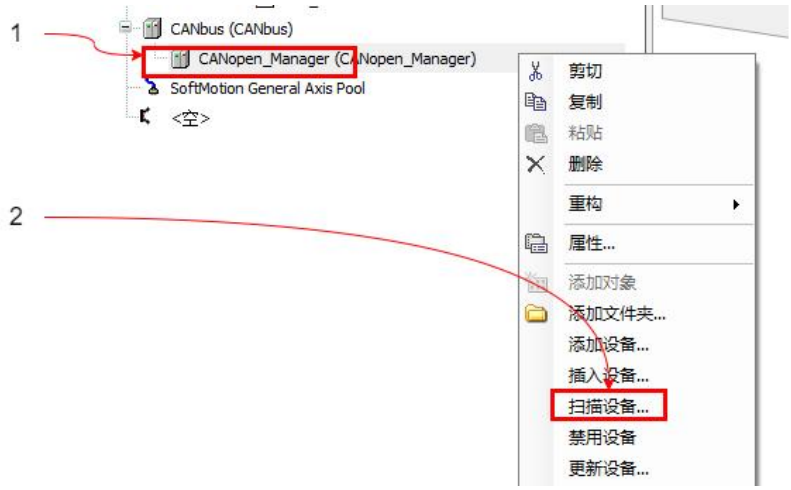


图 7-19

⑧ 在扫描设备页面，点击【扫描设备】按钮；

⑨ 选择扫描到的设备；

⑩ 点击【复制所有设备到工程】按钮；



图 7-20

⑪ 设备添加完成；

⑫ 登录后设备在线表示添加成功。

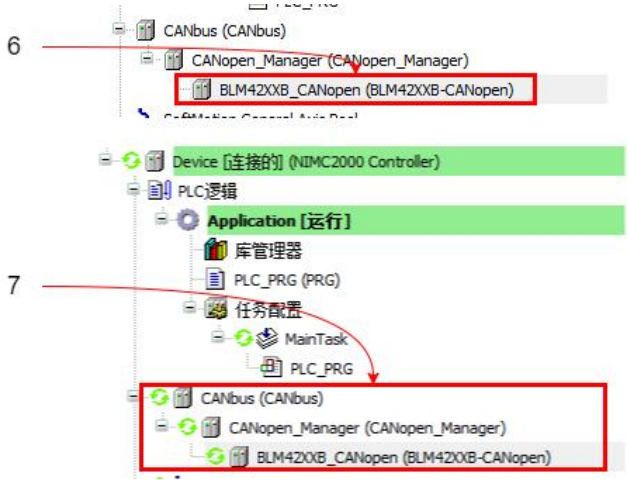


图 7-21

7.5.4 手动添加CANopen设备

⑤ 鼠标选中 CANopen Manger 设备，右键单击，弹出属性；

⑥ 选择【添加设备】；

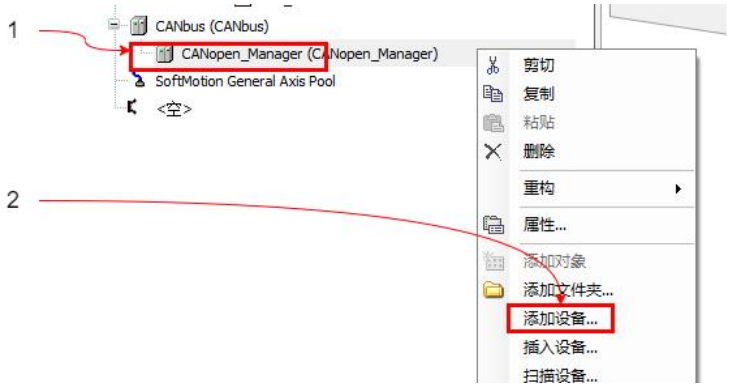


图 7-22

⑦ 选择现场总线-》CANopen-》远程设备；

⑧ 选择【要添加的从站设备】设备；

⑨ 点击【添加设备】按钮，添加选择的从站；

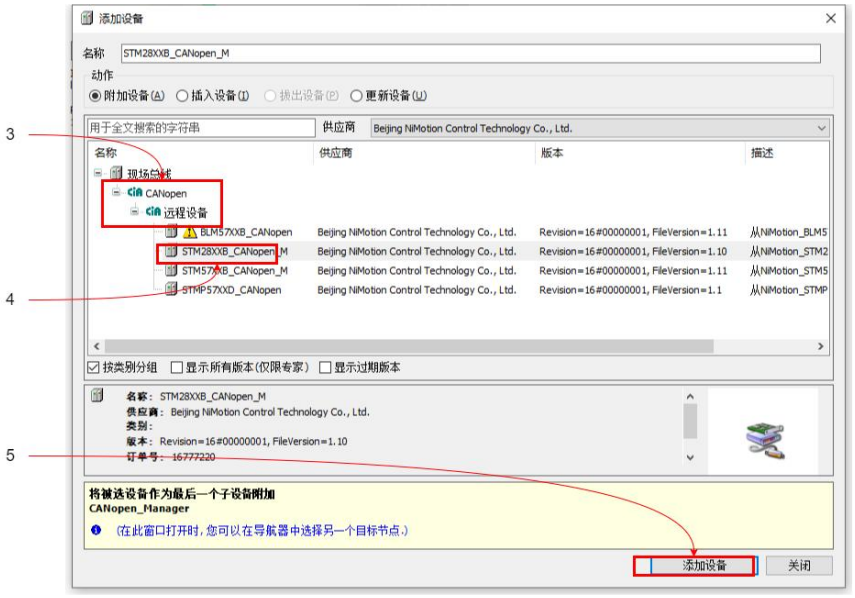


图 7-23

⑩ 鼠标双击添加的 CANopen 从站设备，右侧弹出设备设置页面；

⑪ 在通用栏中，设置使用的 CANopen 从站节点 ID；



图 7-24

⑫ 设备添加完成；

⑬ 登录后设备在线表示添加成功。

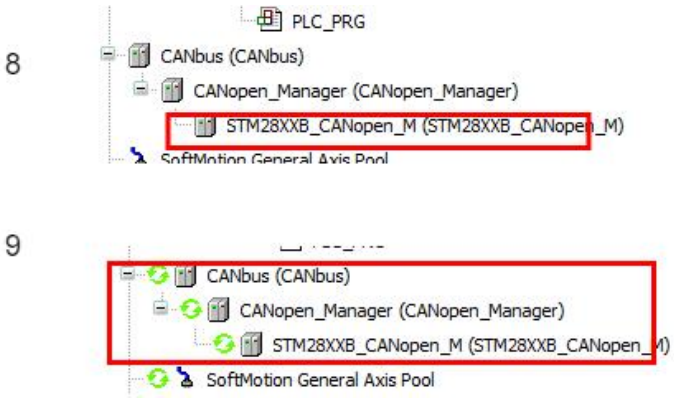


图 7-25

7.6 CANopen 从站配置

① 鼠标选中 Device 设备，右键单击，弹出属性；

② 选择【添加设备】；



图 7-26

③ 选择添加设备页面的“附加设备”；

④ 选择现场总线-》CANbus-；

⑤ 选择【CANbus】设备；

⑥ 点击【添加设备】按钮，添加 CANopen 主站；

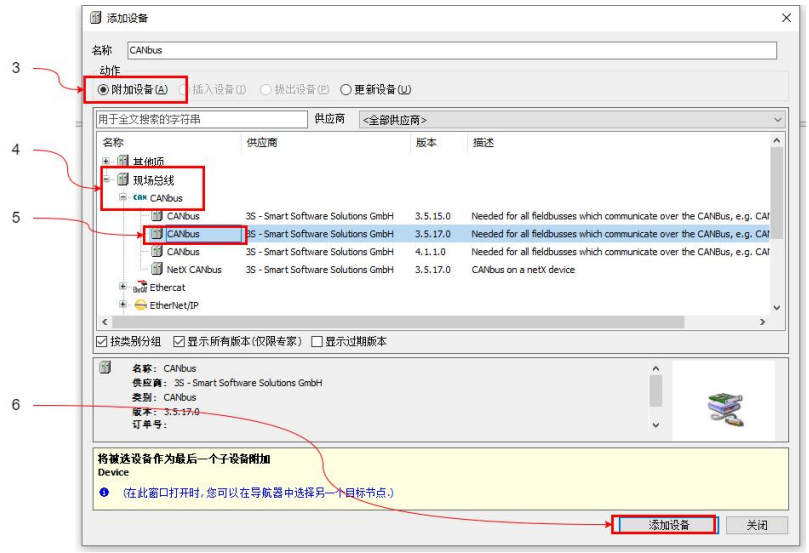


图 7-27

⑦ 鼠标双击【CANbus】设备，右侧弹出设备设置页面；

⑧ 在通用栏中，设置使用的 CAN 网络号和 CAN 波特率；



图 7-28

⑨ 鼠标右击【CANbus】设备，右侧弹出属性选择框；

⑩ 点击【添加设备】按钮，添加 CANopen Manger 管理器；



图 7-29

⑪ 选择现场总线-》CANopen-》本地设备；

⑫ 选择【CANopen Device】设备；

⑬ 点击【添加设备】按钮，添加 Device。

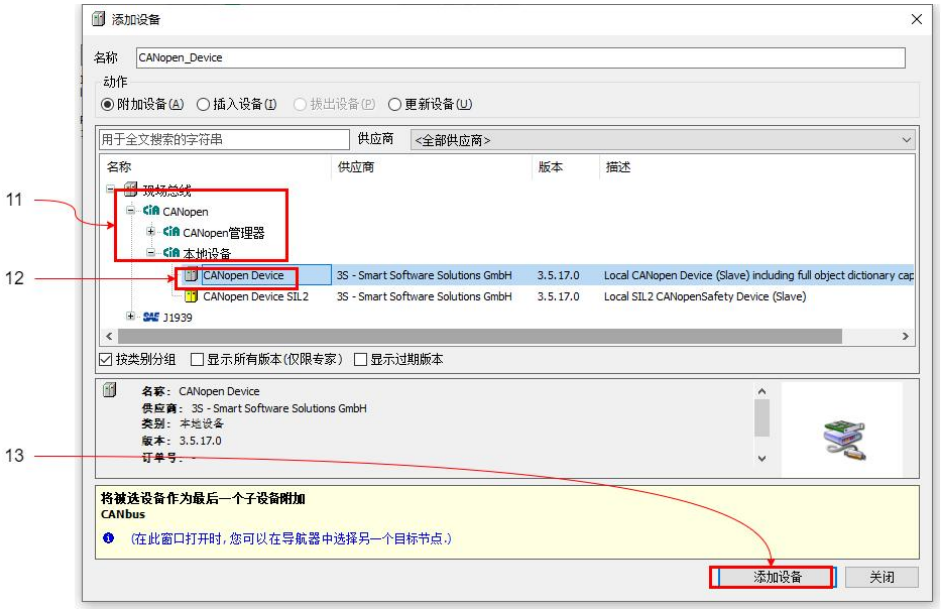


图 7-30

8 MODBUS TCP 配置

8.1 Modbus TCP 主站配置

① 鼠标选中 Device 设备，右键单击，弹出属性；

② 选择【添加设备】；

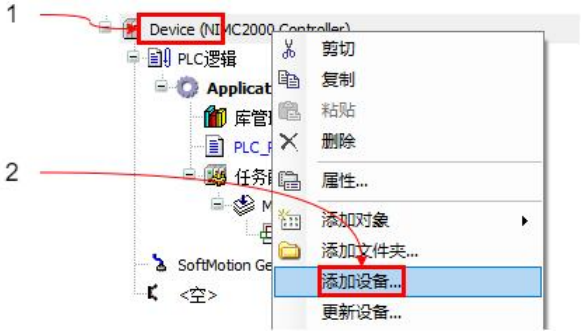


图 8-1

③ 选择现场总线-》以太网适配器-》【Ethernet】设备；

④ 点击【添加设备】按钮，添加 Ethernet 网卡；

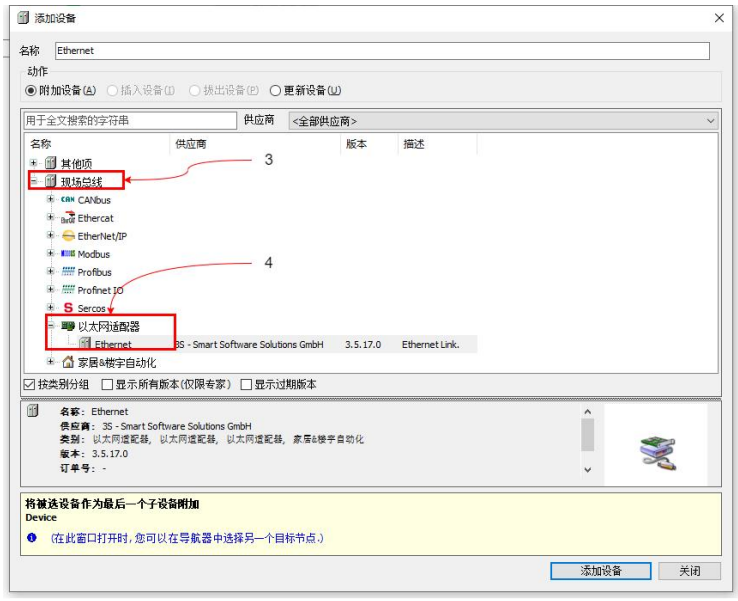


图 8-2

⑤ 鼠标双击【Ethernet】设备，右侧弹出设备设置页面；

⑥ 在通用栏中，选择网络接口后面的【Browse】按钮，弹出网络适配器选择页面；

⑦ 选择使用的网卡；

⑧ 点击【确认】按钮，网卡选择完毕；

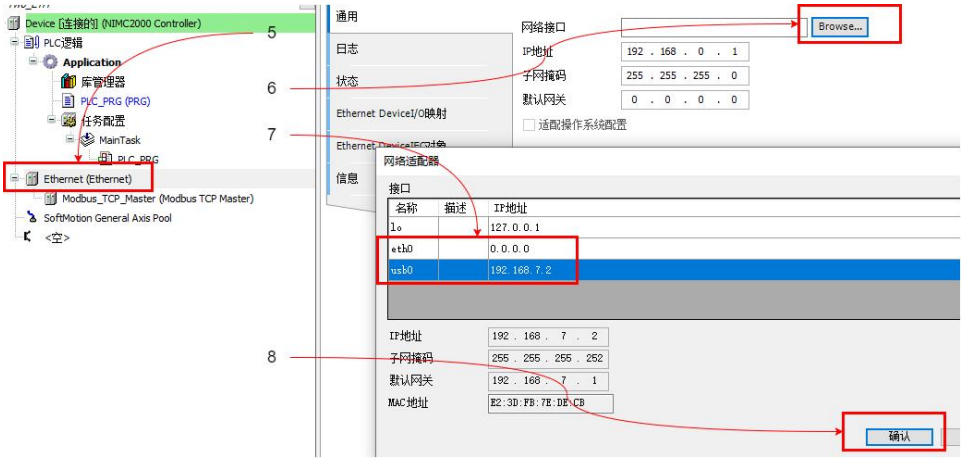


图 8-3

⑨ 鼠标选中 Ethernet 设备，右键单击，弹出属性；

⑩ 选择【添加设备】，在添加设备页面选择 MODBUS TCP 主站设备；

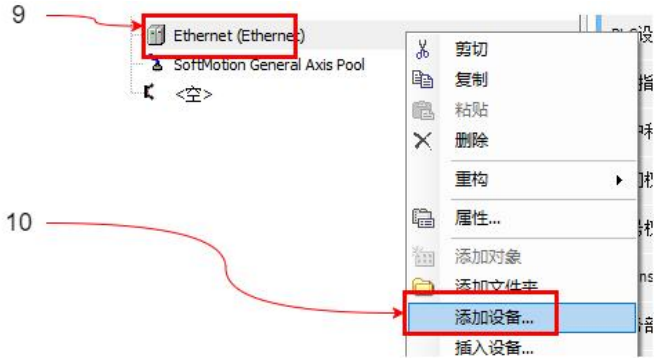


图 8-4

⑪ 选择现场总线-》Modbus-》MODBUS TCP 主站-》【MODBUS TCP Master】；

⑫ 点击【添加设备】按钮，添加 MODBUS TCP 主站设备；

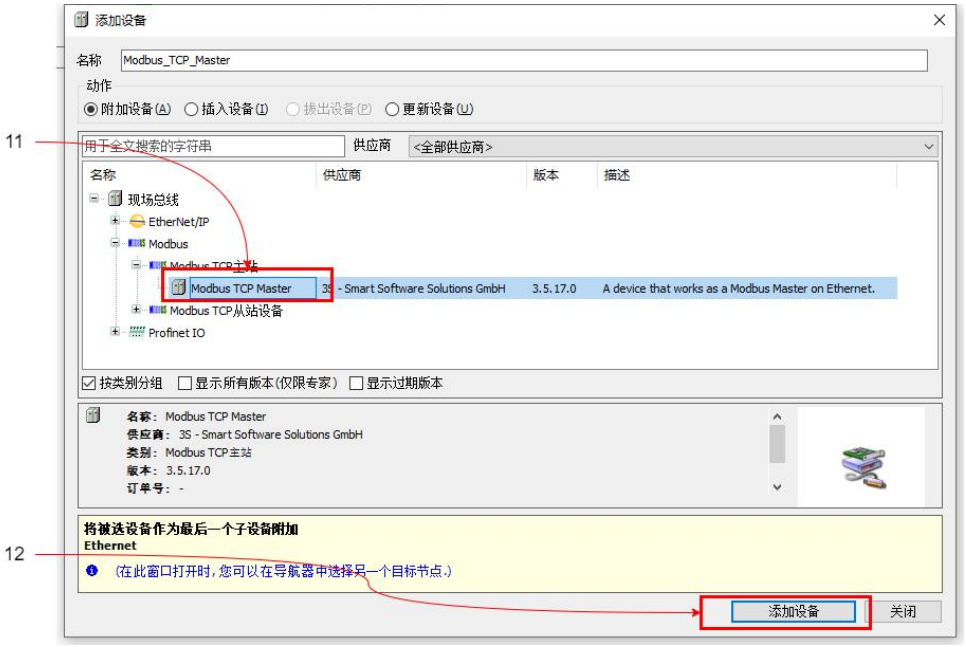


图 8-5

⑬ 鼠标选中 MODBUS TCP Master 设备，右键单击，弹出属性；

⑭ 选择【添加设备】，在添加设备页面选择 MODBUS TCP 从站设备；

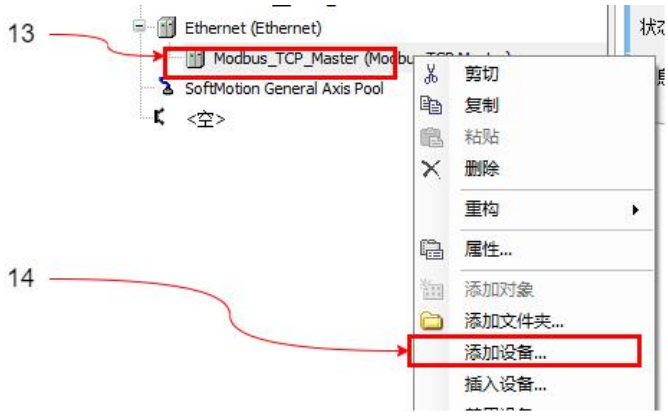


图 8-6

⑮ 选择现场总线-》Modbus-》MODBUS TCP 从站-》【MODBUS TCP Slave】；

⑯ 点击【添加设备】按钮，添加 MODBUS TCP 从站设备；

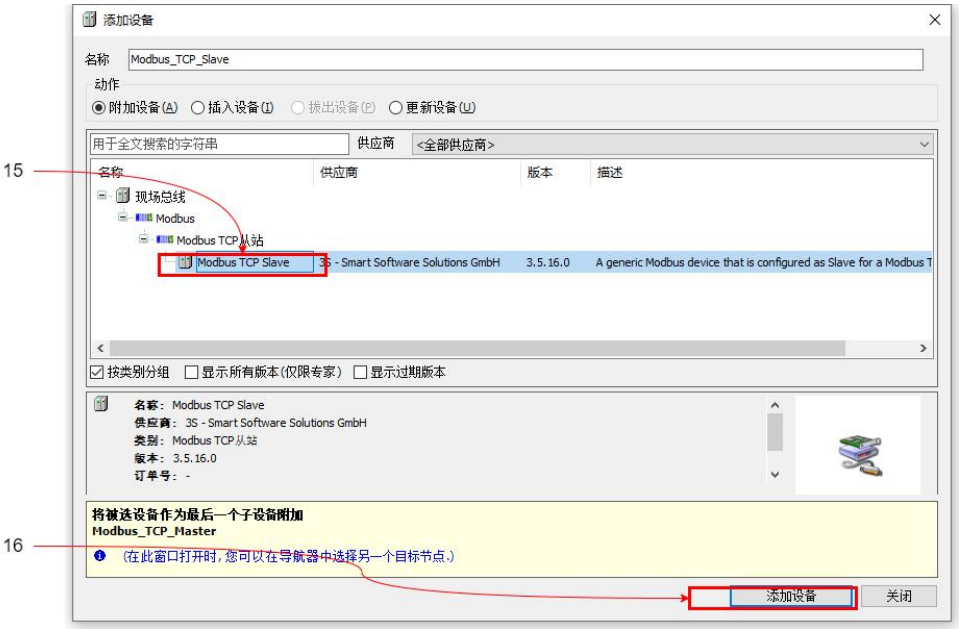


图 8-7

⑰ 鼠标双击【MODBUS TCP Slave】设备，右侧弹出设备设置页面；

⑱ 在通用栏中，对要连接的从站配置 IP、端口、响应超时时间；



图 8-8

8.2 Modbus TCP 从站配置

① 鼠标选中 Device 设备，右键单击，弹出属性；

② 选择【添加设备】；

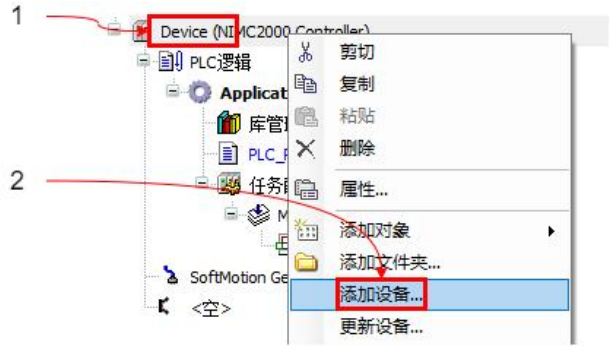


图 8-9

③ 选择现场总线-》以太网适配器-》【Ethernet】设备；

④ 点击【添加设备】按钮，添加 Ethernet 网卡；

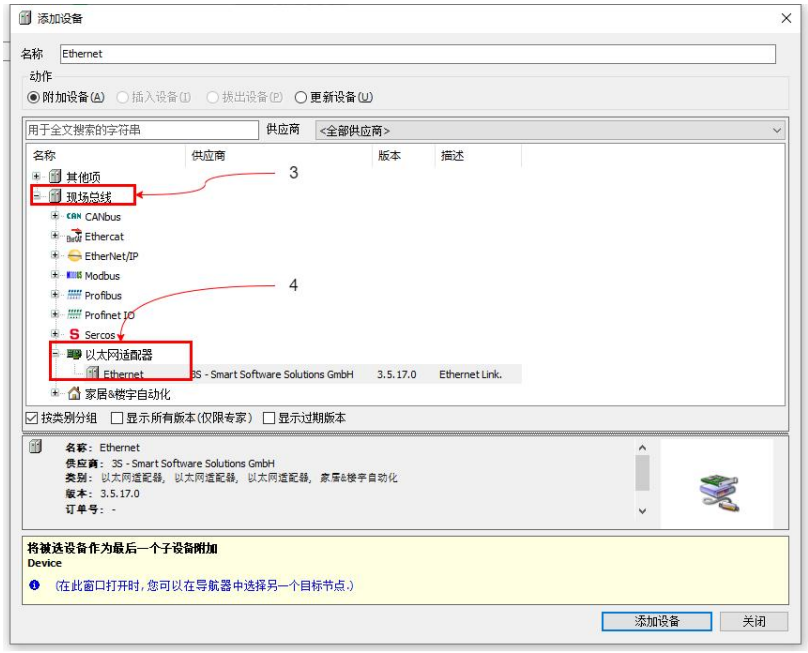


图 8-10

⑤ 鼠标双击【Ethernet】设备，右侧弹出设备设置页面；

⑥ 在通用栏中，选择网络接口后面的【Browse】按钮，弹出网络适配器选择页面；

⑦ 选择使用的网卡；

⑧ 点击【确认】按钮，网卡选择完毕；

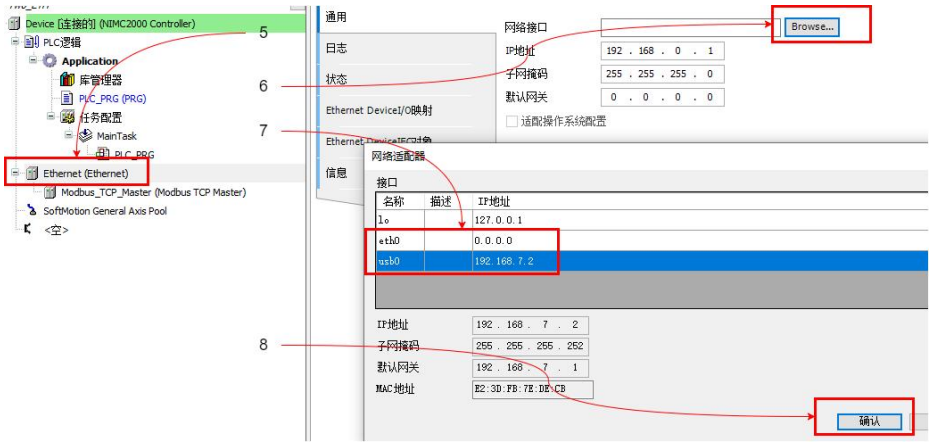


图 8-11

⑨ 鼠标选中 Ethernet 设备，右键单击，弹出属性；

⑩ 选择【添加设备】，在添加设备页面选择 MODBUS TCP 主站设备；

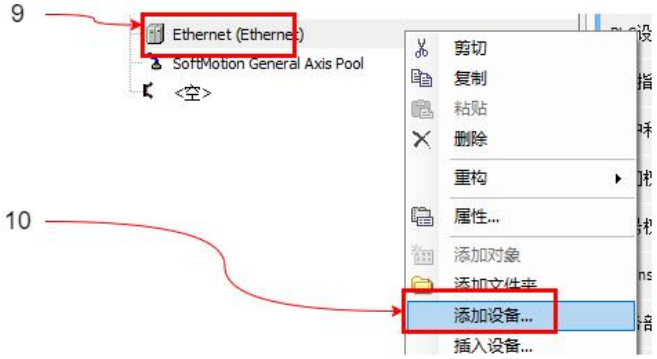


图 8-12

⑪ 选择现场总线-》Modbus-》MODBUS TCP 从站设备-》【MODBUS TCP Slave Device】；

⑫ 点击【添加设备】按钮，添加 MODBUS TCP 从站设备；

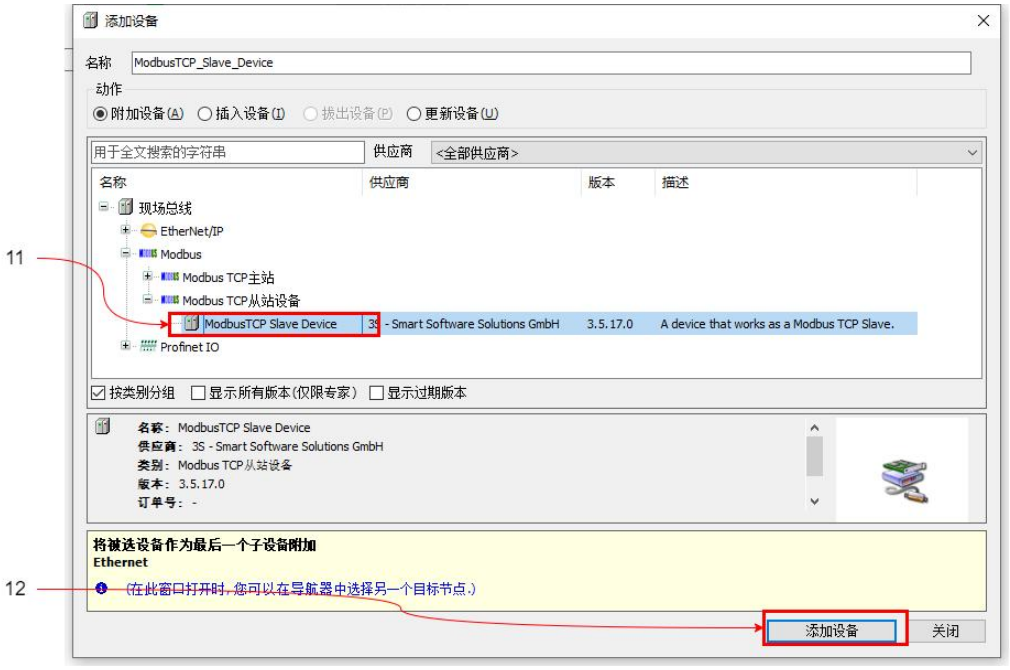


图 8-13

⑬ 鼠标双击【MODBUS TCP Slave Device】设备，右侧弹出设备设置页面；

⑭ 在通用栏中，对要配置的从站设置寄存器数量；

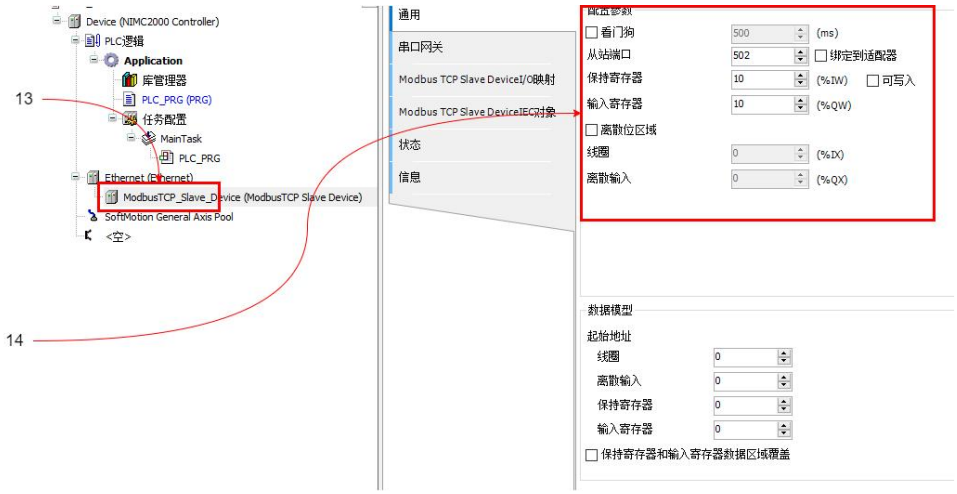


图 8-14

9 MODBUS RTU 配置

9.1 Modbus RTU 主站配置

① 鼠标选中 Device 设备，右键单击，弹出属性；

② 选择【添加设备】；

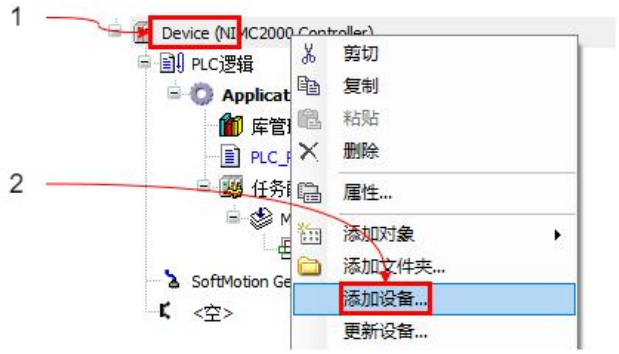


图 9-1

③ 选择现场总线-》MODBUS-》Modbus 串行端口-》【Modbus COM】设备；

④ 点击【添加设备】按钮，Modbus 串行端口；

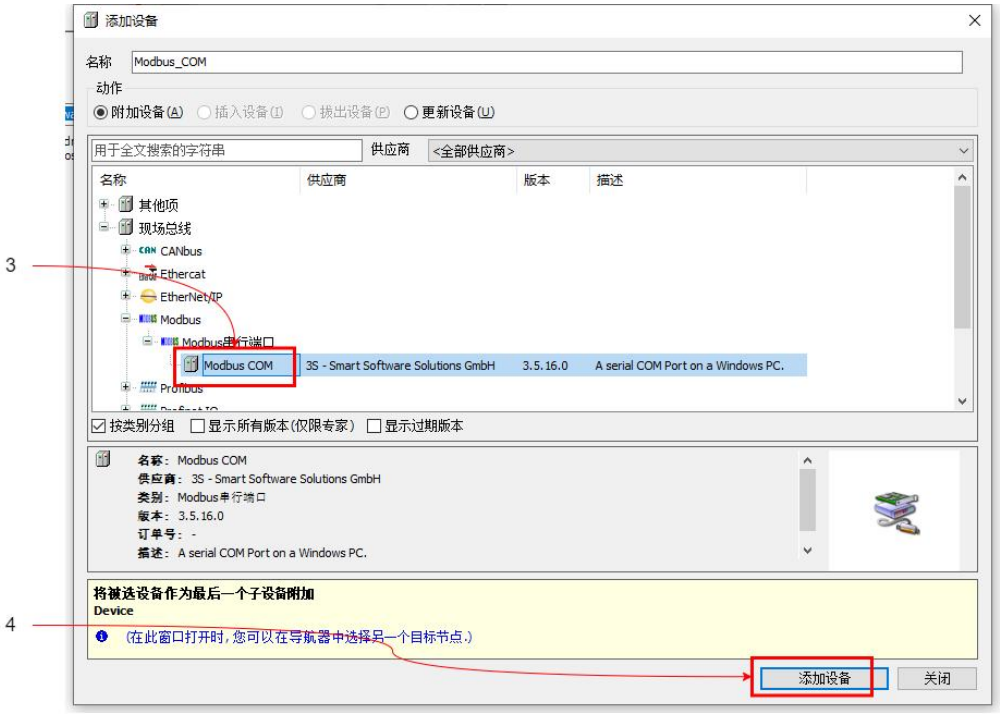


图 9-2

⑤ 鼠标双击【Modbus COM】设备，右侧弹出设备设置页面；

⑥ 在通用栏中，设置端口号、波特率、校验位、数据位、停止位等；

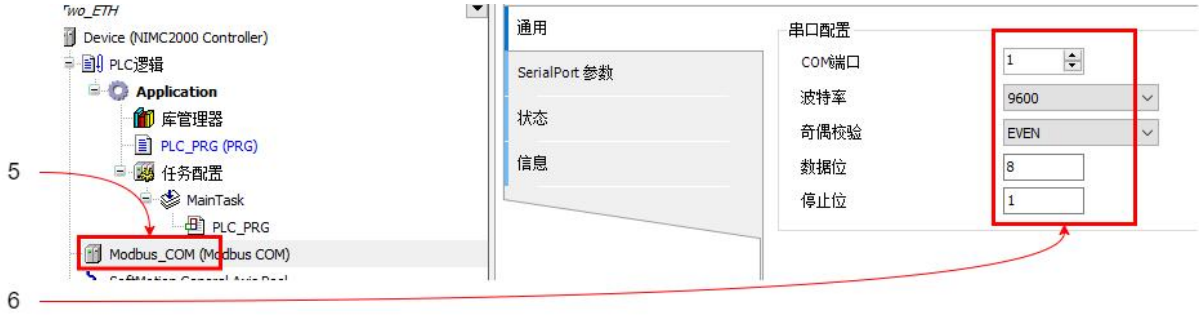


图 9-3

⑦ 鼠标选中 Modbus COM 设备，右键单击，弹出属性；

⑧ 选择【添加设备】，在添加设备页面选择 MODBUS Master，COM Port 主站设备；

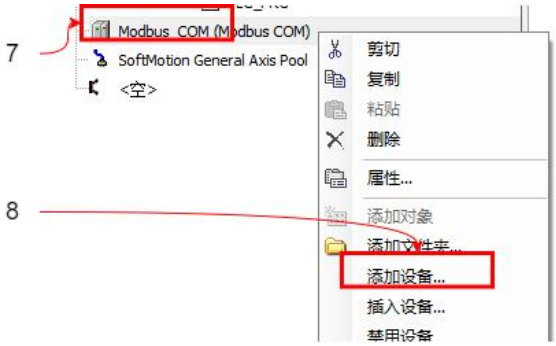


图 9-4

⑨ 选择现场总线-》MODBUS-》Modbus 串行主站-》【MODBUS Master，COM Port】设备；

⑩ 点击【添加设备】按钮；

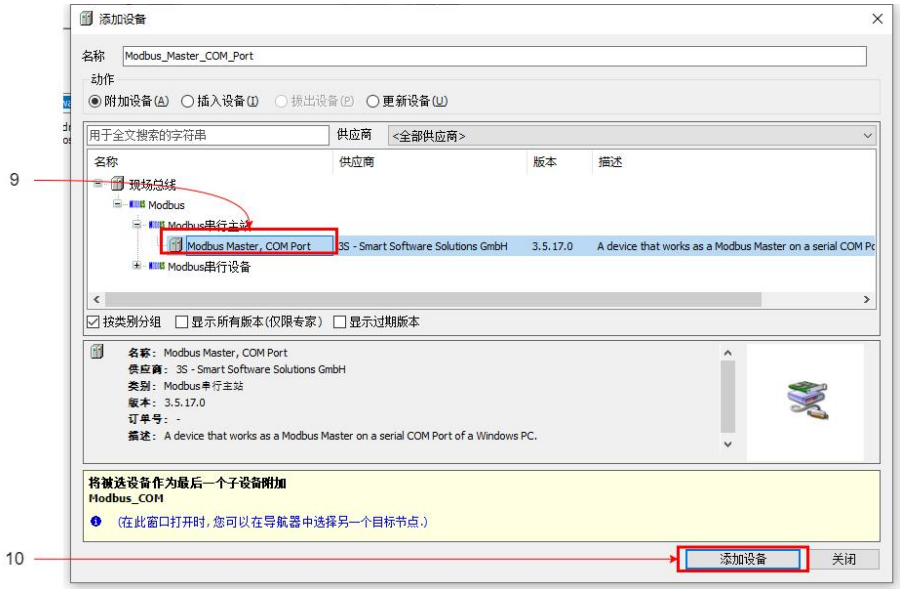


图 9-5

⑪ 鼠标双击【MODBUS Master，COM Port】设备，右侧弹出设备设置页面；

⑫ 在通用栏中，设置传输模式、响应超时时间等。

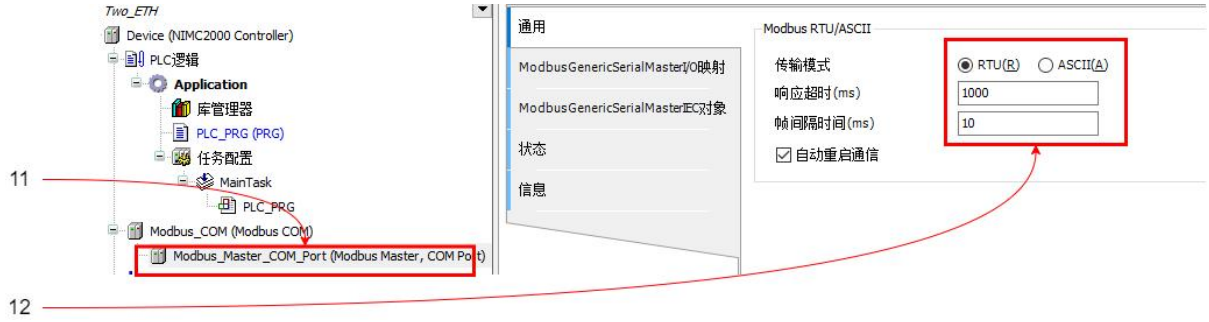


图 9-6

⑬ 鼠标选中 MODBUS Master，COM Port 设备，右键单击，弹出属性；

⑭ 选择【添加设备】，在添加设备页面选择 MODBUS Slave，COM Port 主站设备；

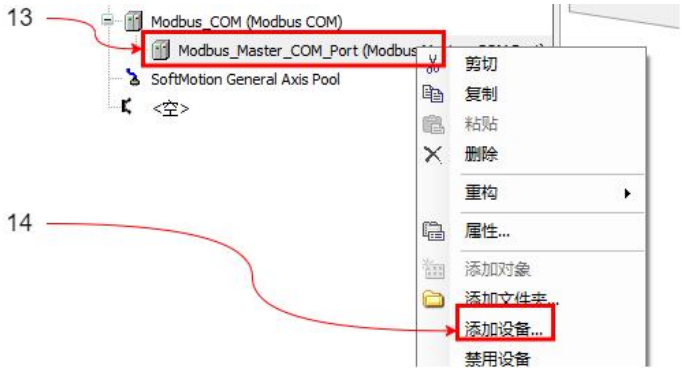


图 9-7

⑮ 选择现场总线-》MODBUS 串行从站-》Modbus 串行端口-》【MODBUS Slave，COM Port】设备；

⑯ 点击【添加设备】按钮；

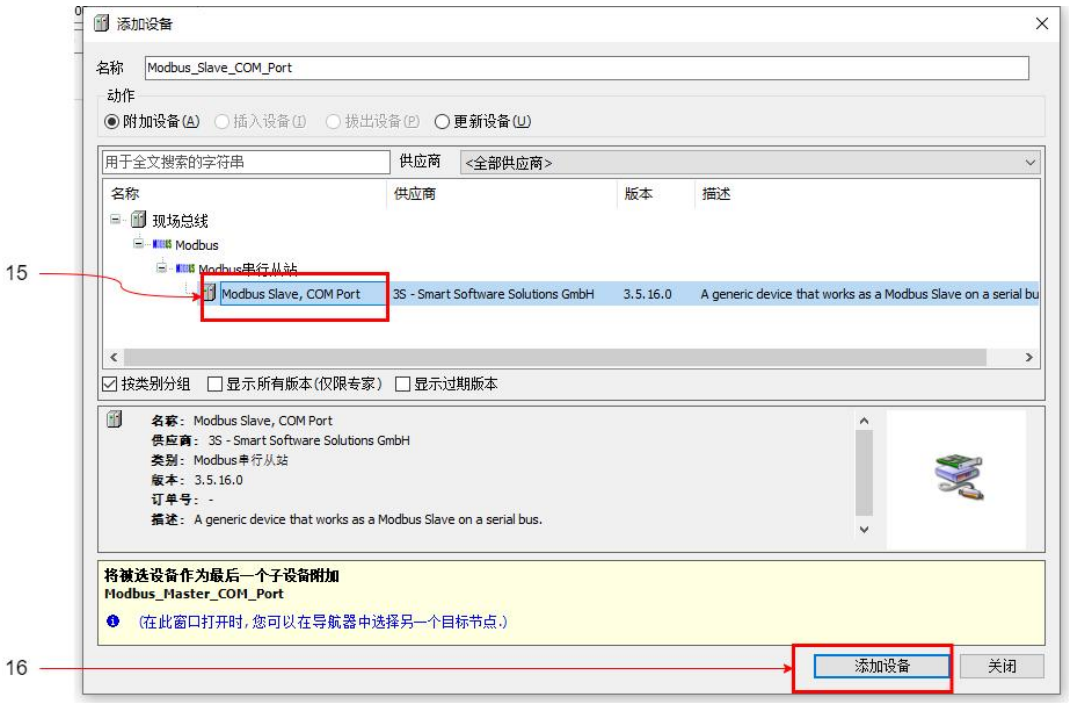


图 9-8

⑰ 鼠标双击【MODBUS Slave，COM Port】设备，右侧弹出设备设置页面；

⑱ 在通用栏中，设置从站地址、响应超时时间等。

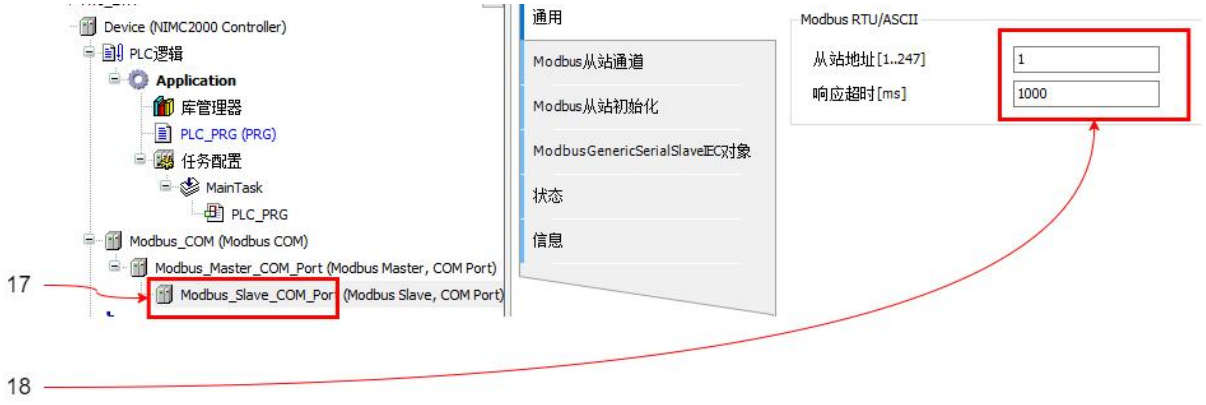


图 9-9

9.2 Modbus RTU 从站配置

① 鼠标选中 Device 设备，右键单击，弹出属性；

② 选择【添加设备】；

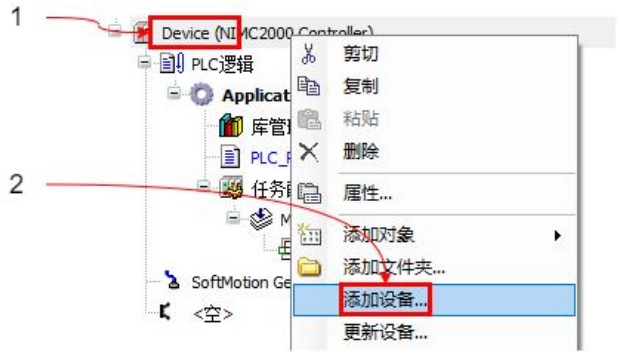


图 9-10

③ 选择现场总线-》MODBUS-》Modbus 串行端口-》【Modbus COM】设备；

④ 点击【添加设备】按钮，Modbus 串行端口；

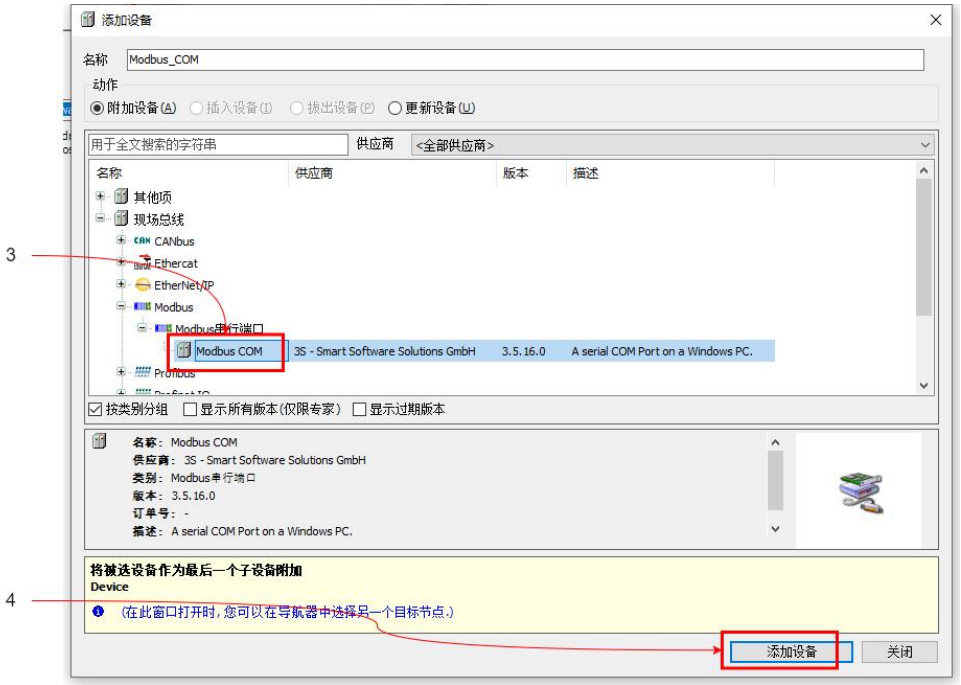


图 9-11

⑤ 鼠标双击【Modbus COM】设备，右侧弹出设备设置页面；

⑥ 在通用栏中，设置端口号、波特率、校验位、数据位、停止位等；

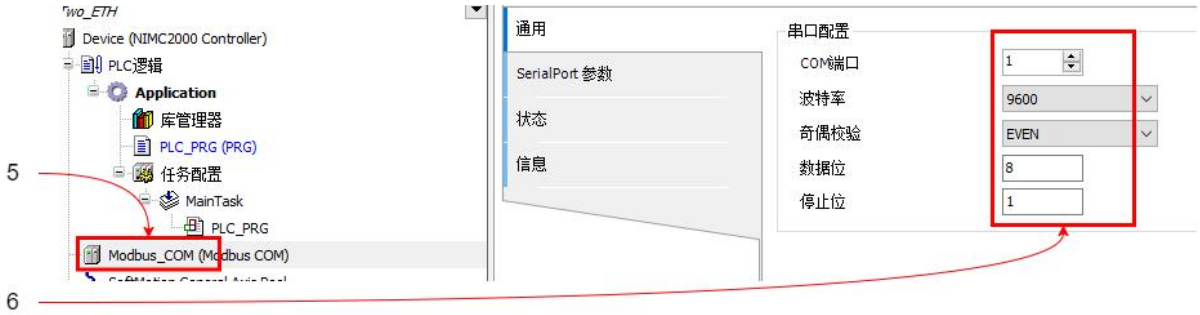


图 9-12

⑦ 鼠标选中 Modbus COM 设备，右键单击，弹出属性；

⑧ 选择【添加设备】，在添加设备页面选择 MODBUS Serial Device 从站设备；

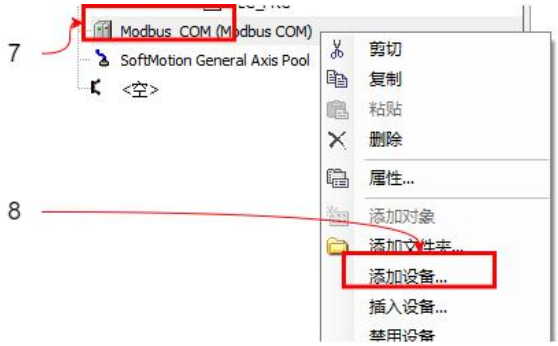


图 9-13

⑨ 选择现场总线-》MODBUS-》Modbus 串行设备-》【MODBUS Serial Device】设备；

⑩ 点击【添加设备】按钮；

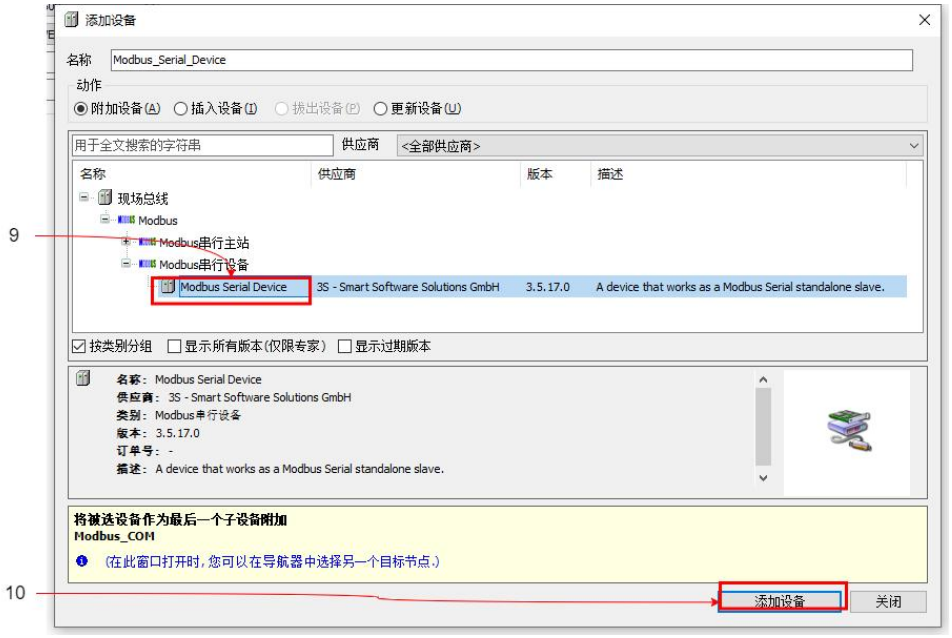


图 9-14

⑪ 鼠标双击【MODBUS Serial Device】设备，右侧弹出设备设置页面；

⑮ 在通用栏中，对要配置的从站设置寄存器数量；



图 9-15

10 OPCUA 服务端配置

① 鼠标选中 Application，右键单击，弹出属性；

② 选择【添加对象】；

③ 选择【符号配置】；

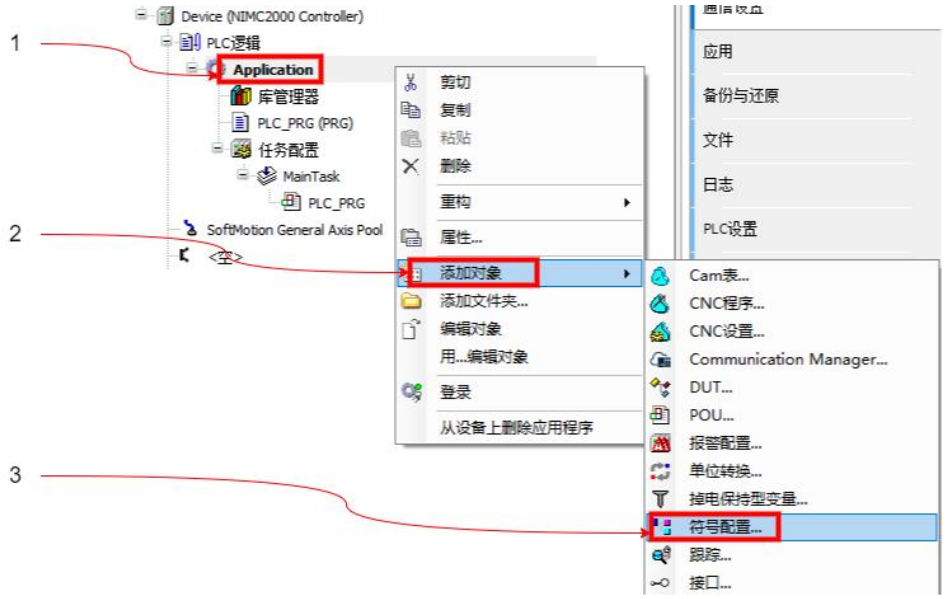


图 10-1

④ 双击【符号配置】选项，右侧弹出设置页面；

⑤ 选择要添加到 opcua 客户端的变量；

⑥ 点击【编译】按钮后，将程序下载到本产品中，通过 OPCUA 客户端访问 PLC 的 IP，既可获取到映射的变量。

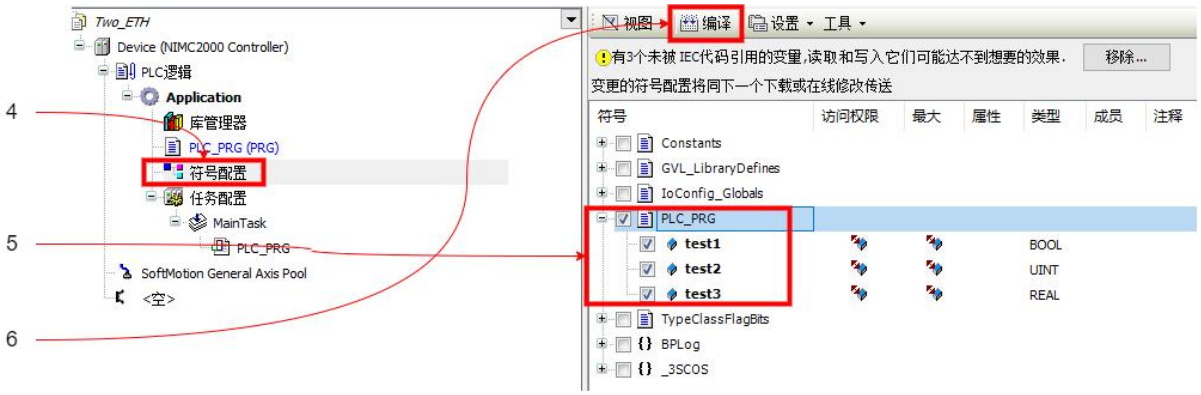


图 10-2

11 自定义 CAN 配置

Codesys 自定义 CAN 协议的实现，可以通过两个库分别实现：

CANBUS 库：这是 Codesys 的 CAN 标准库，CANopen 也是基于该库，但这个库不开源；

CANAPI 库：这是 Codesys 的 CAN 接口开源库，支持自定义 CAN2.0A 和 CAN2.0B 协议，这个功能简单，可以方便使用。

本文档通过 CANAPI 库来讲解自定义 CAN 协议的实现

变量定义部分：

```
VAR
    (* CAN接口配置，使用CAN1，波特率1M，数据帧缓存：10条 *)
    g_busConfig     : CAN.DRIVER_CONFIG := (usiNetwork:= 1, uiBaudrate:= 1000, ctMessages:= 10);
    (* 接收报文超时时间 *)
    g_tTimeLimit        : TIME := T#5MS;

    (* CAN 驱动 *)
    driver1lbit     : CAN.CANBus_1lbit := (DriverConfig:= g_busConfig);

    (* 接收句柄 *)
    hMaskReceiver   : CAN.CAA.HANDLE;

    (* 接收消息处理进程，类似于中断，接收的报文存储在这个进程里面 *)
    msgProcessor    : MsgProcessor_EchoST;

    (* 定义接收掩码 *)
    maskConfig      : CAN.RECEIVER_MASK := (
                                    dwIdValue:= 0,
                                    dwIdMask:= 0,
                                    xRTRValue:= FALSE,
                                    xRTRMask:= FALSE,
                                    x29BitIdValue:= FALSE,
                                    x29BitIdMask:= FALSE,
                                    xTransmitValue:= FALSE,
                                    xTransmitMask:= TRUE,
                                    xAlwaysNewest:= FALSE);

    (* 发送消息的结构体 *)
    send_mess: CAN.MESSAGE;

    (*开始发送标志*)
    send_start:BOOL;

    (*定时发送—定时器*)
    send_ton:TON;

    (*发送的数据*)
    send_data : ARRAY[0..7] OF BYTE := [1,2,3,4,5,6,7,8];
```

图 11-1

程序部分：CAN 配置

```
/// 配置CAN 驱动
driver1lbit(DriverConfig:= g_busConfig, eError=> );

/// 接收进程
msgProcessor(CANdriver:= driver1lbit);

/// Create a Mask Receiver to receive all messages
/// Each call of GetMaskReceiver generates a new Receiver
IF hMaskReceiver = CAN.CAA.gc_hINVALID THEN
    hMaskReceiver:= driver1lbit.GetMaskReceiver(Mask:= maskConfig, eError=> );
END_IF


/// Receive messages and hand them over to the MsgProcessor_Echo
driver1lbit.ReceiveMessage(hReceiver:= hMaskReceiver, itfMsgProcessor:= msgProcessor, tTimeLimit:= g_tTimeLimit, eError=> );
```

图 11-2

程序部分：自定义发送

```
send_ton(IN:= NOT send_ton.Q, PT:= T#10S, Q=> , ET=> );

(*要发送的帧ID*)
send_mess.udiCanID:=16#301;
(*要发送的帧数据*)
send_mess.abyData:=send_data;
(*要发送的数据长度*)
send_mess.usiDataLength:=8;
(*是否是扩展帧*)
send_mess.xIs29BitMessage:=FALSE;
(*是否是远程帧*)
send_mess.xRTR:=FALSE;

(*定时发送*)
IF send_ton.Q AND send_start THEN
    (*发送函数*)
    driver11bit.SendMessage(Message:= send_mess, eError=> );
END_IF
```

图 11-3

程序部分：CAN 接收处理

```
/// 接收到数据
IF _pCANdriver <> 0 THEN
    (*获取接收到的数据*)
    CAN_Recieve_data:=Message.abyData;

    (*获取接收到的帧ID*)
    CAN_Recieve_ID:=Message.udiCanID;

    (*接收数据的处理*)
    Message.udiCanId:= Message.udiCanID + 1;
    _pCANdriver^.SendMessage(Message:= Message, eError=> );

    (*接收报文计数*)
    g_countMsg_ST:= g_countMsg_ST +1;
END_IF
```

图 11-4

12 自定义 UDP 配置

Codesys 库文件安装

UDP 通信需要用到的 codesys 库：Network,3.5.17.0 与 UDP,3.5.17.0。

在工程中打开库管理器->高级->输入库名称搜索，如下图：

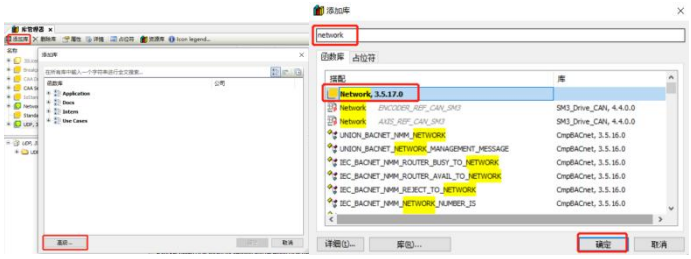


图 12-1

安装完成标志如下图:

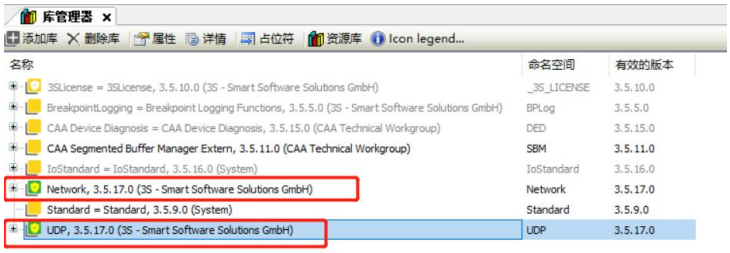


图 12-2

设置相应的 IP 地址和端口号

PLC 的 IP 地址和目标主机的 IP 必须在相同网段下。

```
PLC_IpAddr   : NBS.IP_ADDR:= (sAddr := '192.168.7.2');  // PLC 本机地址:
                                                        // USB 的IP地址: 192.168.7.2
                                                        // Eth0的IP地址: 192.168.1.200
                                                        // ETH1的IP地址: 192.168.0.200

Target_IpAddr : NBS.IP_ADDR:= (sAddr := '192.168.7.1'); // 目标UDP主机的IP地址

PLC_IPPort    :    UINT := 8000;  // PLC      本地的端口号
Target_IPPort :    UINT := 6000;  // 目标UDP 主机的端口号
```

图 12-3

配置 UDP 句柄、发送功能块、接收功能块

```
fbPeerServer    :    NBS.UDP_Peer;          //UDP 句柄

fbSend          :    NBS.UDP_Send;          //UDP 发送功能块
xSend           :    BOOL;                  //UDP 发送标志位
sSendMsg        :    STRING;                //UDP 要发送的内容
SendNum         :    UDINT;                 //发送计数

fbReceive       :    NBS.UDP_Receive;       //UDP 接收功能块
xRecieve        :    BOOL;                  //UDP 接收标志位
sRcvMsg         :    STRING;                //UDP 接收的内容
RecieveNum      :    UDINT;                 //接收计数
```

图 12-4

```
// UDP句柄初始化
fbPeerServer(
    xEnable:= xenable,      // UDP 句柄使能
    xDone=> ,
    xBusy=> ,
    xError=> ,
    ipAddr:= PLC_IpAddr,    //配置本机IP地址
    uiPort:= PLC_IPPort,    //配置本机端口号
    ipMultiCast:= ,
    eError=> ,
    xActive=> ,
    hPeer=> );
```

图 12-5

```
// UDP 发送
IF xSend AND fbPeerServer.hPeer <> CAA.gc_hINVALID THEN

    fbSend.xExecute:=TRUE;                  //发送功能块使能

    IF fbSend.xDone OR fbSend.xError THEN   //上一次发送完毕后准备下次发送
        fbSend.xExecute:=FALSE;
        SendNum:=SendNum+1;                 //发送计数
        sSendMsg:=concat('SendNum:', UDINT_TO_STRING(SendNum)); // 发送内容
    END_IF
ELSE
    fbSend.xExecute:=FALSE;
END_IF


fbSend(
    xExecute:= ,
    udiTimeOut:= ,
    xDone=> ,
    xBusy=> ,
    xError=> ,
    hPeer:= fbPeerServer.hPeer,  //UDP 句柄使能后的 指针
    ipAddr:= Target_IpAddr,      //UDP 目标主机 IP地址
    uiPort:= Target_IPPort ,     //UDP 目标主机 端口号
    szSize:= len(sSendMsg),      //UDP 要发送内容的长度
    pData:= ADR(sSendMsg),       //UDP 要发送内容的地址
    eError=> );
```

图 12-6

```
// UDP 接收
IF xRecieve AND fbPeerServer.hPeer <> CAA.gc_hINVALID THEN
    fbReceive.xEnable:=TRUE;          //UDP 接收使能

    IF fbReceive.xReady THEN          //UDP 接收完成计数
        RecieveNum:=RecieveNum+1;
    END_IF
ELSE
    fbReceive.xEnable:=FALSE;
END_IF

fbReceive(
    xEnable:= ,
    xDone=> ,
    xBusy=> ,
    xError=> ,
    hPeer:= fbPeerServer.hPeer ,   //UDP 句柄使能后的 指针
    szSize:= SIZEOF(sRcvMsg),      //UDP 接收字节数
    pData:= ADR(sRcvMsg),          //UDP 接收内容
    eError=> ,
    xReady=> ,
    ipFrom=> ,                     //UDP 接收内容的IP地址
    uiPortFrom=> ,                 //UDP 接收内容的端口号
    szCount=> );
```

图 12-7

UDP 发送接收测试

UDP 发送、接收测试软件：正点原子网络调试助手 或 SSCOM（丁丁串口软件）；

设置协议为 UDP，设定电脑的 IP 和接收数据端口号，打开连接；



图 12-8

（3）输入目标主机（PLC）的 IP 地址和端口号

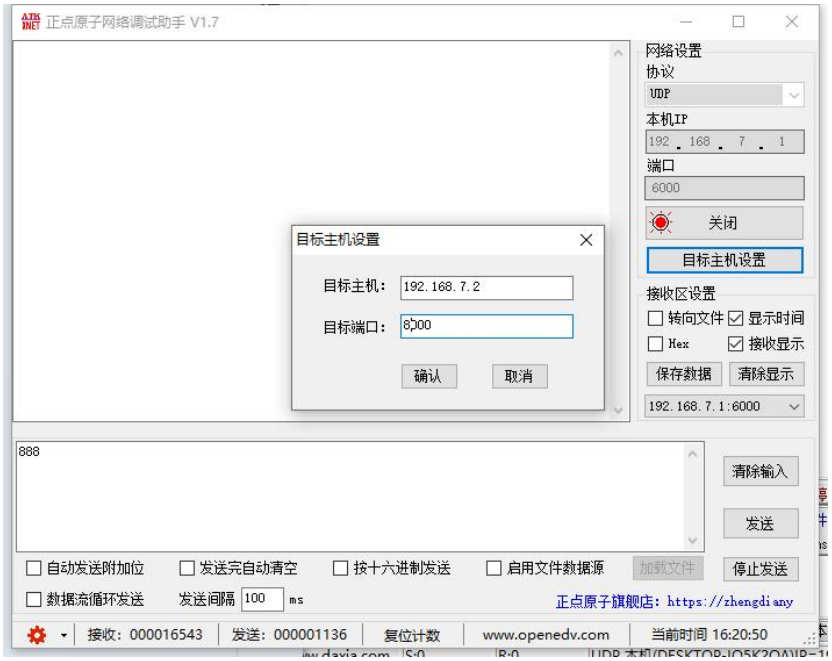


图 12-9

输入发送的内容，点击发送

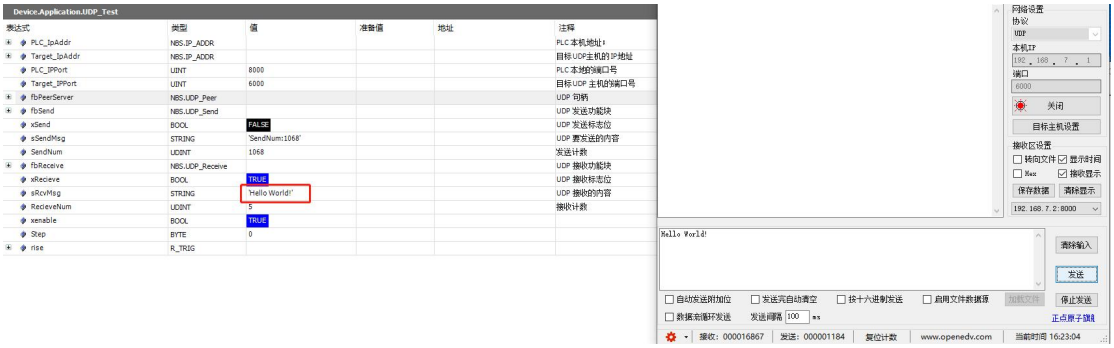


图 12-10

（4）PLC 中打开发送标志位，上位机软件变会持续接收 UDP 发送的内容。（UDP 测试软件打开连接后，必须先向 PLC 的 IP 地址发送一段内容，建立连接后才能正常接收 PLC 中 UDP 返回的数据）

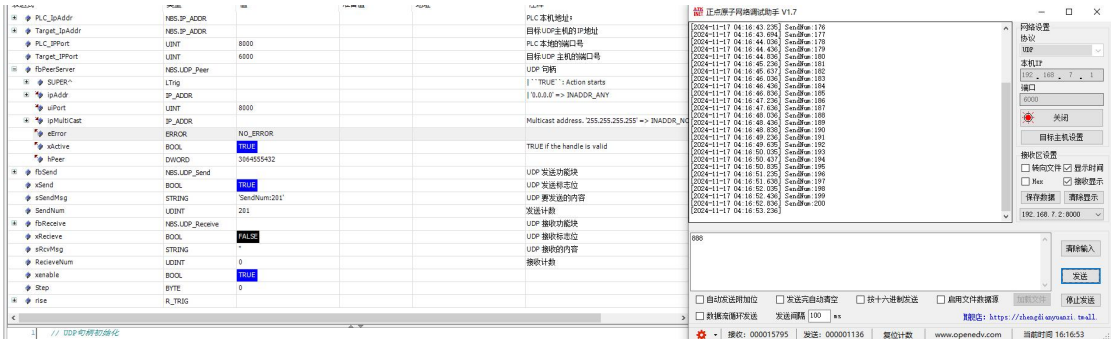


图 12-11

版本变更记录

版本号	修订内容简述	修订日期
A	创建	2026-04-15

更多最新的立迈胜资讯，请扫码关注

公众号　　视频号

北京立迈胜控制技术有限责任公司
Beijing NiMotion Control Technology Co., Ltd.

地 址：北京市大兴区金星路12号院3号楼
电 话：010-60213882
传 真：010-60213882
邮 箱：nimotion@nimotion.com
网 址：www.nimotion.cn

南京立迈胜机器人有限公司
Nanjing NiMotion Robotics Co., Ltd.

地 址：南京市六合区虎跃东路8号B6栋
电 话：025-57569916
传 真：025-57569916
邮 箱：salesNJ@nimotion.com
网 址：www.nimotion.net

立迈胜深圳办事处
NiMotion Group Shenzhen Office

地 址：深圳市南山区云谷创新产业园二期4栋
手 机：19926697610
邮 箱：salesSZ@nimotion.com

※本产品目录中所刊载的产品性能和规格，如因产品改进等原因发生变更时，恕不另行通知，敬请谅解。

（版权所有，翻版必究）