



## **CP340/CP341/CP440/CP441 通讯及编程**

CP340/CP341/CP440/CP441 Communication and Programming

Getting Started

Edition (2010 年 05 月)

**摘要** 本文档主要用于讨论了以下几方面问题:

- SIMATIC S7 系列串行通讯模板的功能描述;
- SIMATIC S7 系列串行通讯模板的 ASCII 协议通信;
- SIMATIC S7 系列串行通讯模板的 MODBUS RTU 协议通信

**关键词** 串行通信, 通信模块, ASCII, MODBUS RTU, SIMATIC S7

**Key Words** Serial Communication, CP, ASCII, MODBUS RTU, SIMATIC S7

## 目 录

1 概述 .....	.5
2 相关软件及使用介绍 .....	.6
2.1 STEP7 .....	.6
2.2 PtP 驱动软件包 .....	.6
2.3 MODBUS Master 驱动软件包 .....	.6
2.4 MODBUS Slave 驱动软件包 .....	.6
2.5 ModScan32 / ModSlave .....	.6
2.6 SSCOM32 .....	.6
3 手册及相关应用文档 .....	.7
4 ASCII 协议通信 .....	.8
4.1 CP340 的 ASCII 协议通信 .....	.8
4.1.1 硬件列表 .....	.8
4.1.2 组态和配置 .....	.8
4.1.3 编写通信程序 .....	.12
4.1.4 设备连接 .....	.14
4.1.5 通信测试 .....	.15
4.2 CP341 的 ASCII 协议通信 .....	.16
4.2.1 硬件列表 .....	.16
4.2.2 组态和配置 .....	.16
4.2.3 编写通信程序 .....	.19
4.2.4 设备连接 .....	.22
4.2.5 通信测试 .....	.22
4.3 CP440 的 ASCII 协议通信 .....	.23
4.3.1 硬件列表 .....	.23
4.3.2 组态和配置 .....	.23
4.3.3 编写通信程序 .....	.26
4.3.4 设备连接 .....	.28
4.3.5 通信测试 .....	.28
4.4 CP441 的 ASCII 协议通信 .....	.28
4.4.1 硬件列表 .....	.29
4.4.2 组态和配置 .....	.29
4.4.3 编写通信程序 .....	.34
4.4.4 设备连接 .....	.35
4.4.5 通信测试 .....	.36
5 MODBUS RTU 协议通信 .....	.37
5.1 CP341 的 MODBUS Slave 协议通信 .....	.37
5.1.1 硬件列表 .....	.37
5.1.2 组态和配置 .....	.38
5.1.3 编写通信程序 .....	.42
5.1.4 设备连接 .....	.44
5.1.5 通信测试 .....	.44
5.2 CP341 的 MODBUS Master 协议通信 .....	.47
5.2.1 硬件列表 .....	.47
5.2.2 组态和配置 .....	.47

---

5.2.3 编写通信程序 .....	50
5.2.4 设备连接 .....	53
5.2.5 通信测试 .....	54
5.3 CP441-2 的 MODBUS Master 协议通信 .....	56
5.3.1 硬件列表 .....	56
5.3.2 组态和配置 .....	56
5.3.3 编写通信程序 .....	59
5.3.4 设备连接 .....	61
5.3.5 通信测试 .....	61
5.4 CP441-2 的 MODBUS Slave 协议通信 .....	62
5.4.1 硬件列表 .....	62
5.4.2 组态和配置 .....	63
5.4.3 编写通信程序 .....	67
5.4.4 设备连接 .....	68
5.3.5 通信测试 .....	68
附录一 推荐网址 .....	71

## 1 概述

西门子 SIMATIC S7 系列串行通讯模板主要包括 CP340、CP341、CP440-1、CP441-1/2、以及 ET200S 的 1SI 3964/ASCII、1SI MODBUS/USS 等模块，它们支持的通讯协议主要有 ASCII、RK512、3964(R)、MODBUS Master、MODBUS Slave 以及其他可加载的协议等，SIMATIC S7 系列串行通信模板及各自所支持的通信协议的对应关系如下表：

	ASCII	3964(R)	RK512	MODBUS Master	MODBUS Slave
CP340	X	X	-	-	-
CP341	X	X	X	X	X
CP440	X	X	-	-	-
CP441-1	X	X	-	-	-
CP441-2	X	X	X	X	X

X 支持 - 不支持

表 1 模板和协议的对应关系表

在实际应用中，可以通过以上串行通信模块完成与如下设备的串行通信：

- 第三方 DCS、PLC 等系统；
- 第三方智能设备、仪表等；
- 接收来自扫描仪、条形码阅读器等设备发出的数据；
- 发送数据到串口打印机等接收设备；
- 其他所有支持 ASCII、MODBUS、用户自定义协议设备；

本文主要讲述了 CP340、CP341、CP440、CP441-1 和 CP441-2 五种模板和它们各自所支持的 ASCII、MODBUS RTU 协议的应用，并附带了每种模板使用的例程，供读者在应用时参考。

**郑重声明：**本文的虚拟工程与真实工程实例有重大差别，示例中并未遵循规范的工程设计流程进行编程，请读者切勿将其与工程实例相混淆；由于此例子是免费的，任何用户可以免费复制或传播此程序例子。程序的作者对此程序不承担任何功能性或兼容性的责任，使用者风险自负；西门子不提供此程序例子的错误更改或者热线支持；为了更好的使用这些模板，建议用户仔细阅读相关模板的使用手册。

## 2 相关软件及使用介绍

### 2.1 STEP7

STEP7 编程软件用于编写 PLC 程序，此软件需要从西门子购买，本文档中所有的程序代码均使用 STEP7 V5.4 SP4 编写。

### 2.2 PtP 驱动软件包

PtP Param V5.1 是 CP340/CP341/CP440/CP441 的驱动软件包，安装此驱动软件包后才能给串行通信模板分配参数，并可以在 STEP7 中集成通讯编程所需要使用的功能块。此驱动随购买模块时一起提供，也可以从下面的链接下载，到目前为止最新的软件版本是 PtP Param V5.1 SP11： <http://support.automation.siemens.com/CN/view/zh/27013524>

### 2.3 MODBUS Master 驱动软件包

CP341 或 CP441-2 用于 MODBUS 主站时，需要安装此驱动软件包，但安装此驱动之前必须先安装 PtP Param V5.1 驱动程序软件包，此驱动软件包可以在购买 MODBUS Dongle 时选择购买，也可以从下面的链接下载：

MODBUS Master (RTU) 软件包：

<http://support.automation.siemens.com/CN/view/zh/27774018>

MODBUS Master (ASCII) 软件包：

<http://support.automation.siemens.com/CN/view/zh/25356060>

### 2.4 MODBUS Slave 驱动软件包

CP341 或 CP441-2 用于 MODBUS 从站时，需要安装此驱动软件包，但安装此驱动之前必须先安装 PtP Param V5.1 驱动程序软件包，此驱动可以在购买 MODBUS Dongle 时选择购买，也可以从下面的链接下载

MODBUS Slave (RTU) 软件包

<http://support.automation.siemens.com/CN/view/zh/27774276>

MODBUS Slave (ASCII) 软件包：

<http://support.automation.siemens.com/CN/view/zh/25356060>

### 2.5 ModScan32 / ModSlave

第三方提供的软件，可以从互联网上免费下载。ModScan32 可用来仿真 MODBUS 主站，测试和 MODBUS 从站的通讯。ModSlave 可用来仿真 MODBUS 从站来测试和 MODBUS 主站的通讯。

### 2.6 SSCOM32

第三方提供的串口调试工具，可以从互联网上免费下载，可用于测试串口通讯。

### 3 手册及相关应用文档

安装完所有的驱动程序后，可以在计算机上找到所有 CP 模板的手册，具体位置为“开始-> SIMATIC-> Documentation -> English”或者直接从 STEP 7 安装路径“...Program files-> Siemens -> Step7 -> S7MANUAL -> S7PtP\_Cp”的文件夹中找到。

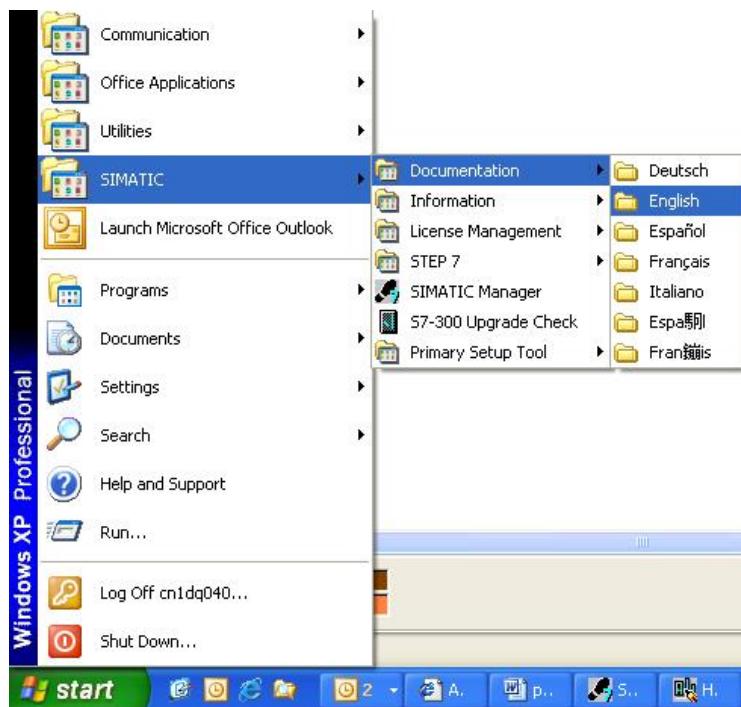


图 1 模块手册路径图

关于西门子串行通信模块的其他应用文档可以登陆西门子自动化与驱动集团网站的下载中心网站 <http://www.ad.siemens.com.cn/download/>，搜索下载如下文档：

文档编号	文档中文标题
A0006	串口通讯模块的信息与使用
A0336	CP341 MODBUS RTU 多站点轮询
A0384	S7-300 CP341 作主 S7-200 作从的 MODBUS 通信
A0123	ET200S MODBUS/USS 模板使用指南
A0413	如何通过 MODBUS 协议实现 S7-1200 与 S7-200 通讯
A0440	CP340 CP341 基于 ASCII 驱动协议的多站点轮询

表 2 下载中心串行通信模块应用文档列表

## 4 ASCII 协议通信

ASCII 协议通讯方式用来实现自定义协议的串行通讯，适用于与支持串行通信的智能仪表、条形码阅读器、扫描仪、打印机等设备进行通讯。

下面以具体示例介绍西门子串行通信模块 CP340、CP341、CP440 和 CP441 的 ASCII 协议通信。

### 4.1 CP340 的 ASCII 协议通信

#### 4.1.1 硬件列表

PS 307	6ES7 307-1EA00-0AA0
CPU 315-2DP	6ES7 315-2AG10-0AB0
MMC	6ES7 953-8LG11-0AA0
CP340	6ES7 340-1AH01-0AE0

#### 4.1.2 组态和配置

1、打开 STEP7，点击 File->New... 创建一新项目，项目名称为 ptp。



图 2 新建项目对话框

2、用鼠标右键点击项目名称，选择 Insert New Object->SIMATIC 300 Station，更改站的名称为 CP340 ASCII。

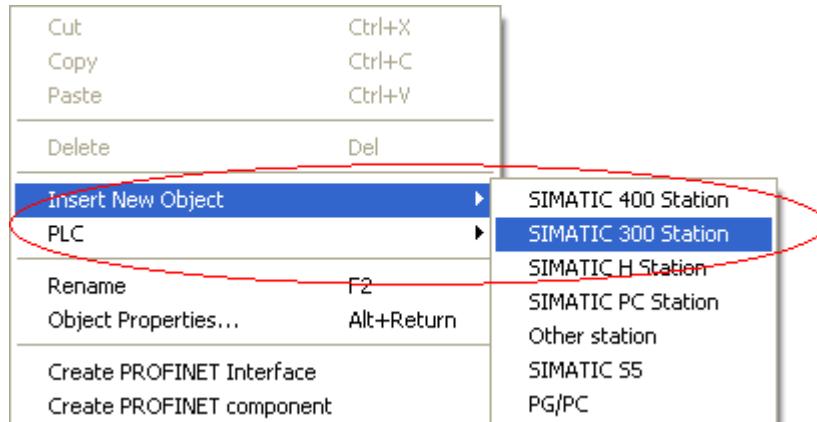


图 3 插入 S7-300 站

3、双击 Hardware 进入硬件组态界面，插入 RACK、CPU315-2DP、CP340。

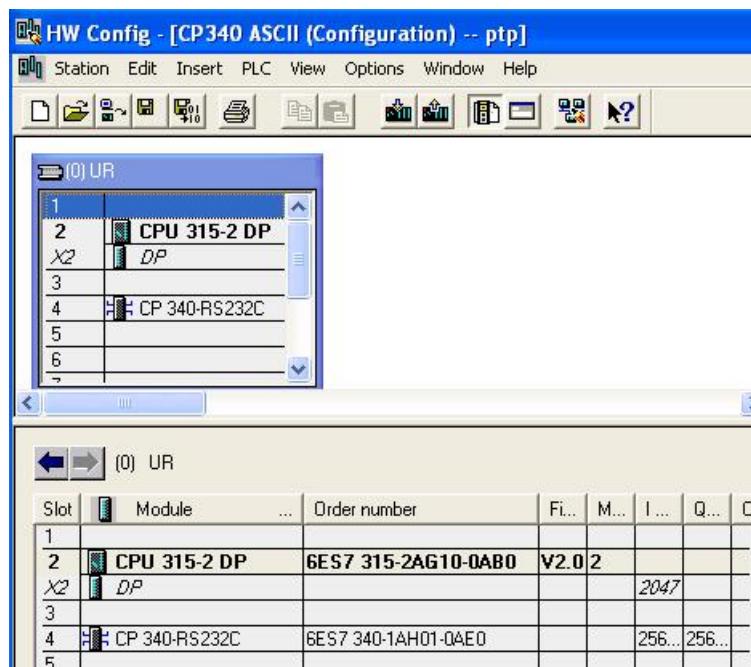


图 4 组态硬件

4、双击 CP340 模板，点击 Parameter...，配置 CP340 参数。

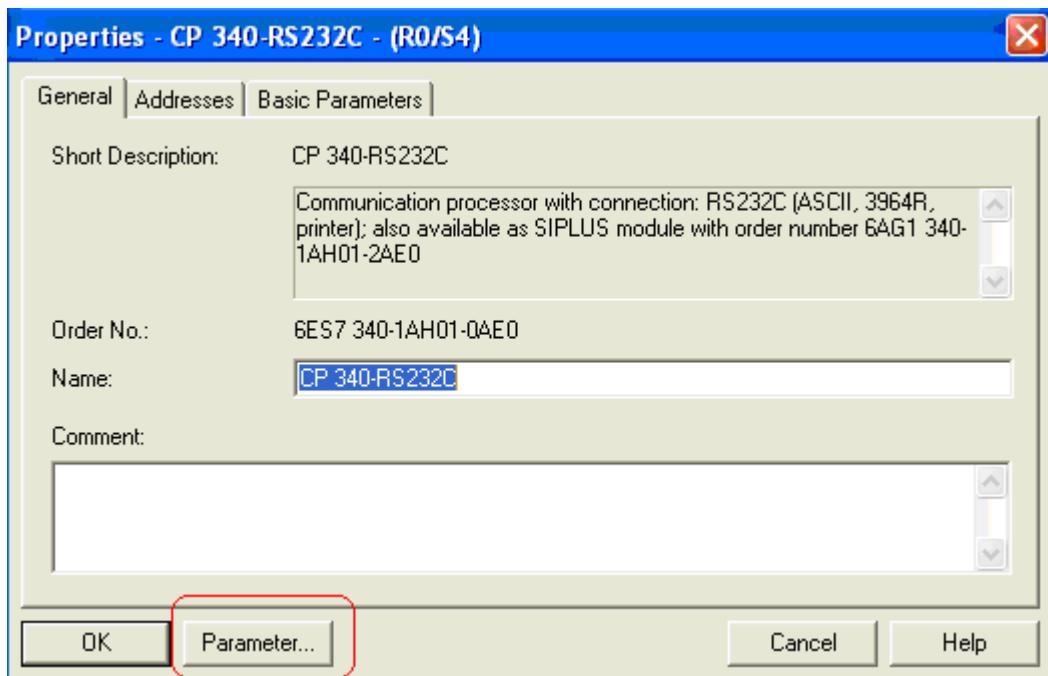


图 5 CP340 属性对话框

5、在 Protocol 选项中选择 ASCII。

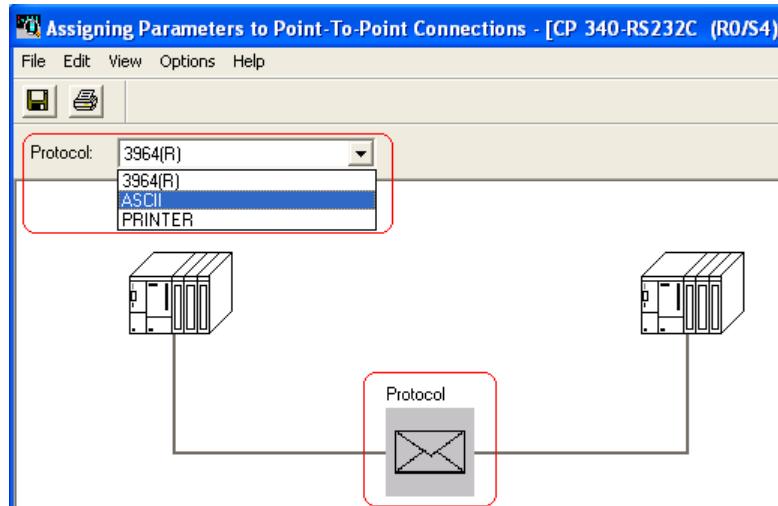


图 6 协议选择

6、双击 Protocol 下的信封图标，配置 ASCII 通信参数，设定的通信参数要和通讯伙伴的相同。通信波特率：9600bps，数据位：8 位，停止位：1 位，奇偶校验：无，字符延迟时间：4ms，其他的参数都采用默认设置。

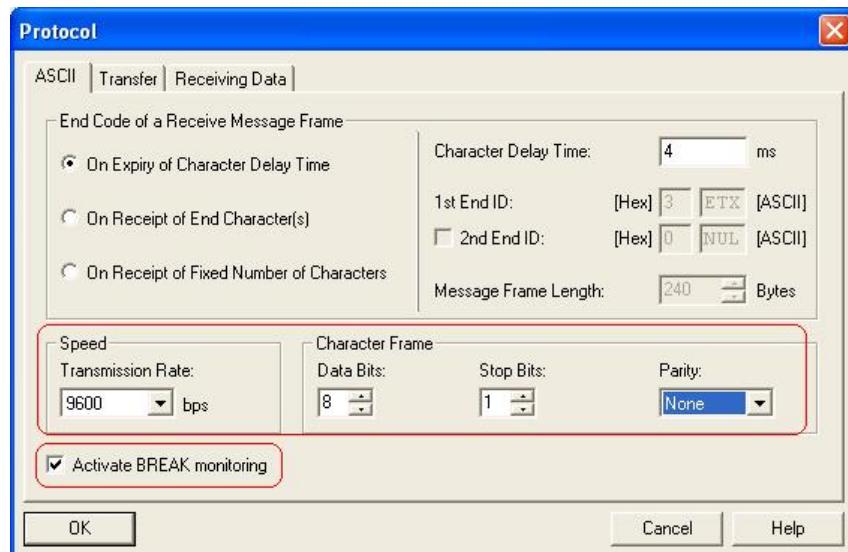


图 7 为 ASCII 协议分配通信参数

如上图所示，如果勾选了“Activate BREAK monitoring”，那么当 RS232C 电缆未连接或者通讯伙伴的串口没有激活时，CP340 的 SF 红色指示灯亮。

## 7、如果选用的是 RS422/485 接口的 CP340 时，还需要设置接口属性，如下所示。

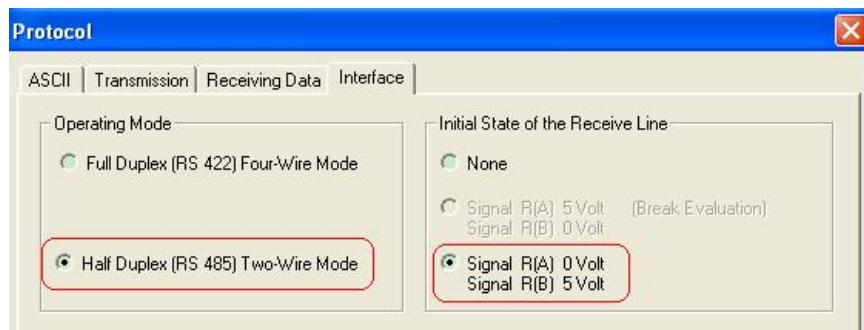


图 8 组态接口参数

当接口类型选择 RS485 (半双工模式) 时，接收端初始状态为 R(A)0V/R(B)5V，不支持断线检测功能；当接口类型选择 RS422 (全双工模式)，并将接收端初始状态设置为 R(A)5V/R(B)0V 时，则支持断线检测功能。这两种状态对应的通信方接收端接口引脚如下：

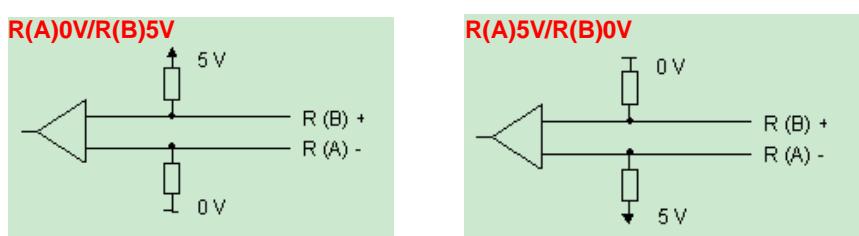


图 9 在 RS485/422 接口处接收器的接线

8、参数配置完成后点击  按钮保存编译硬件组态，并确认没有错误后将硬件组态下载到 CPU 中，至此完成了全部的硬件组态过程。

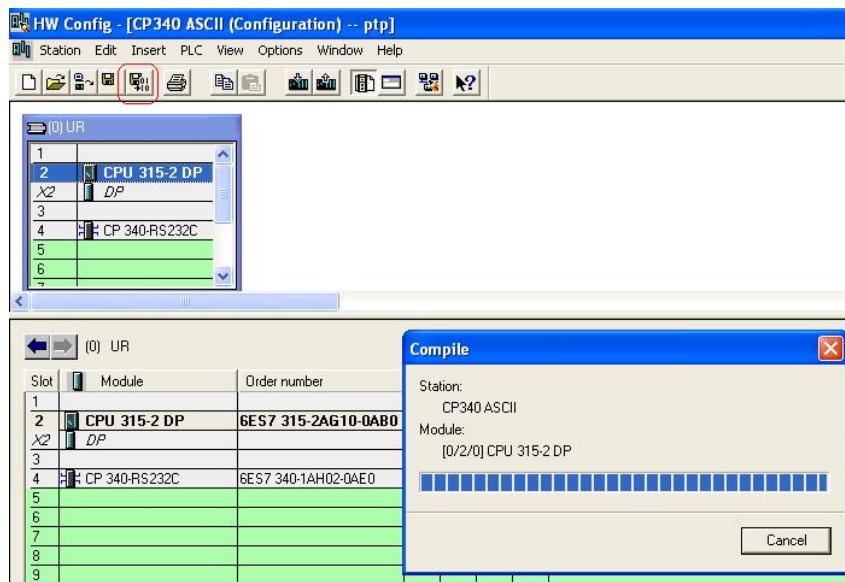


图 10 编译保存硬件组态

#### 4.1.3 编写通信程序

1、双击 OB1，打开 OB1 编程画面，从库 Libraries -> CP PtP -> CP340 中调用发送功能块 FB3 P\_SEND，并为其分配背景数据块 DB3，将块参数 LADDR 设为硬件组态中 CP340 模块的起始逻辑地址 256。

Slot	Module	...	O...	Fi...	M...	I address	Q address	Comment
1								
2	CPU 315-2 DP	6ES7 315-2AG10-0AB0						
X2	DP					2047*		
3								
4	CP 340-RS232C	6ES7 340-1AH02-0AE0						
5								
6								
7								
8								
9								

图 11 CP340 逻辑地址

2、创建发送数据块 DB1。

ddress	Name	Type	Initial value	Comment
*0.0		STRUCT		
+0.0	SEND_Data	ARRAY[1..50]		Temporary place
*1.0		BYTE		
=50.0		END_STRUCT		

图 12 发送数据块 DB1

### 3、调用发送功能块

CP340 的发送功能块 FB3 P\_SEND 的参数设置见下表：

LADDR	硬件组态中的起始逻辑地址，本例中为 256
DB_NO	发送数据块号，本例中为 1 (DB1)
DBB_NO	发送数据的起始地址，本例中为 0 (DB1.DBB0)
LEN	发送数据的长度，本例中为 10
REQ	发送数据触发位，上升沿触发，本例中为 M0.0
R	取消通讯，本例中不用
DONE	发送完成位，发送完成且没有错误时为 TRUE，
ERROR	错误位，为 TRUE 说明有错误
STATUS	状态字，标识错误代码，查看 CP340 手册获得相应的说明

表 3 FB3 P\_SEND 的参数定义

OB1 : “Main Program Sweep (Cycle)”

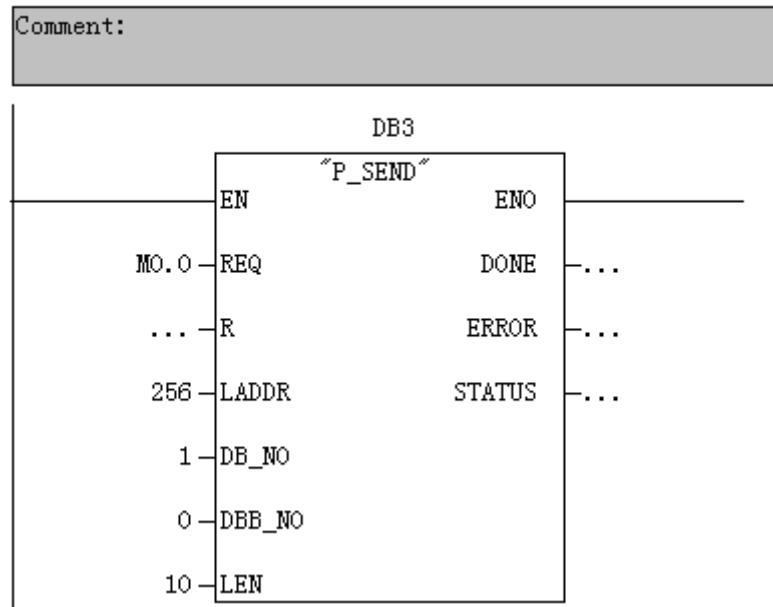


图 13 在 OB1 中调用 FB3 P\_SEND

4、从库 Libraries -> CP PtP -> CP340 中调用接收功能块 FB2 P\_RCV，并为其分配背景数据块 DB2，将 LADDR 设为 CP340 的起始逻辑地址 256。

5、创建接收数据块 DB10。

Address	Name	Type	Initial value	Comment
*0.0		STRUCT		
+0.0	SEND_Data	ARRAY[1..50]		Temporary place
*1.0		BYTE		
=50.0		END_STRUCT		

图 14 接收数据块 DB10

## 6、调用接收功能块

CP340 的接收功能块 FB2 P\_RCV 的参数设置见下表:

LADDR	硬件组态中的起始逻辑地址, 本例中为 256
DB_NO	发送数据块号, 本例中为 10 (DB10)
DBB_NO	发送数据的起始地址, 本例中为 0 (DB10.DBB0)
LEN	接收数据的长度, 本例中为 MW2, 只有在接收到数据的当前周期, 此值不为 0, 可以查看 MW4 的值来确认接收到数据的长度
EN_R	使能接收位, 本例中为 M0.1
R	取消通讯, 本例中不用
NDR	接收完成位, 接收完成并没有错误为 TRUE,
ERROR	错误位, 为 TRUE 说明有错误
STATUS	状态字, 标识错误代码, 查看 CP340 手册获得相应的说明

表 4 FB2 P\_RCV 的参数定义

OB1 : "Main Program Sweep (Cycle)"

Comment:

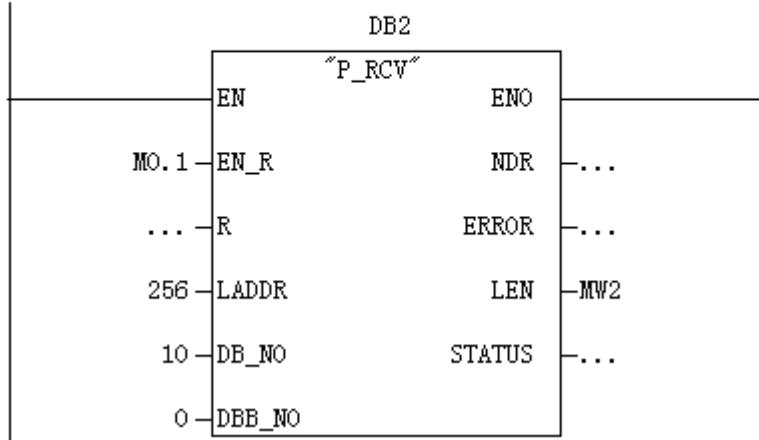


图 15 在 OB1 中调用 FB2 P\_SEND

### 4.1.4 设备连接

使用标准的 RS232C 电缆连接 CP340 和计算机的串口, 具体的电缆连接方法请查看 CP340 手册 B 章节连接电缆部分。

#### 4.1.5 通信测试

##### 1、CP340 从 SSCOM32 接收数据

首先将硬件配置和程序下载到 CPU315-2DP 中，将 M0.1 设为 TRUE，使能接收。在计算机上打开软件 SSCOM32，在字符输入窗口输入 16 进制数据 0102030405060708090A，然后点击发送按钮发送数据。



图 16 通过 SSCOM32 发送数据

在 STEP7 中打开接收数据块 DB10 监控接收到的数据，可以看到 SSCOM32 发送的数据被正确的接收。

Address	Name	Type	Initial value	Actual value
0.0	RECEIVE_Data	BYTE	B#16#0	B#16#01
1.0	RECEIVE_Data	BYTE	B#16#0	B#16#02
2.0	RECEIVE_Data	BYTE	B#16#0	B#16#03
3.0	RECEIVE_Data	BYTE	B#16#0	B#16#04
4.0	RECEIVE_Data	BYTE	B#16#0	B#16#05
5.0	RECEIVE_Data	BYTE	B#16#0	B#16#06
6.0	RECEIVE_Data	BYTE	B#16#0	B#16#07
7.0	RECEIVE_Data	BYTE	B#16#0	B#16#08
8.0	RECEIVE_Data	BYTE	B#16#0	B#16#09
9.0	RECEIVE_Data	BYTE	B#16#0	B#16#0A

图 17 监视接收数据块 DB10

##### 2、CP340 发送数据到 SSCOM32

将 M0.0 从 FALSE 置成 TRUE，CP340 将发送数据，将 DB1 中前 10 个字节设为十六进制的 31 32 33 34 35 36 37 38 39 3A，可以从 SSCOM32 接收窗口中看到收到的数据（十六进制）。



图 18 通过 SSCOM32 接收数据

如果使用的 CP340 是 RS485 接口的模板，使用计算机串口调试程序时要使用 RS232C/RS485 转换器。

## 4.2 CP341 的 ASCII 协议通信

### 4.2.1 硬件列表

PS 307	6ES7 307-1EA00-0AA0
CPU 315-2DP	6ES7 315-2AG10-0AB0
MMC	6ES7 953-8LG11-0AA0
CP341	6ES7 341-1AH01-0AE0

### 4.2.2 组态和配置

1、打开上面创建的项目 ptp，用鼠标右键点击项目名称，选择 Insert New Object->SIMATIC 300 Station，更改站的名称为 CP341 ASCII。

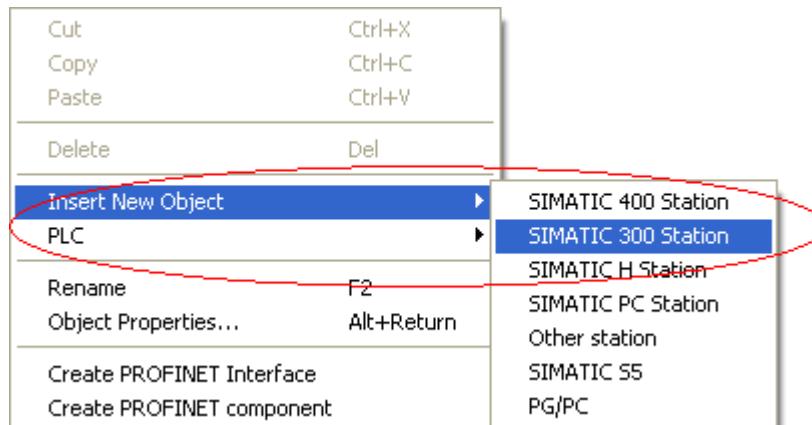


图 19 插入 S7-300 站

2、双击 Hardware 进入硬件组态界面，插入 RACK、CPU315-2DP、CP341。

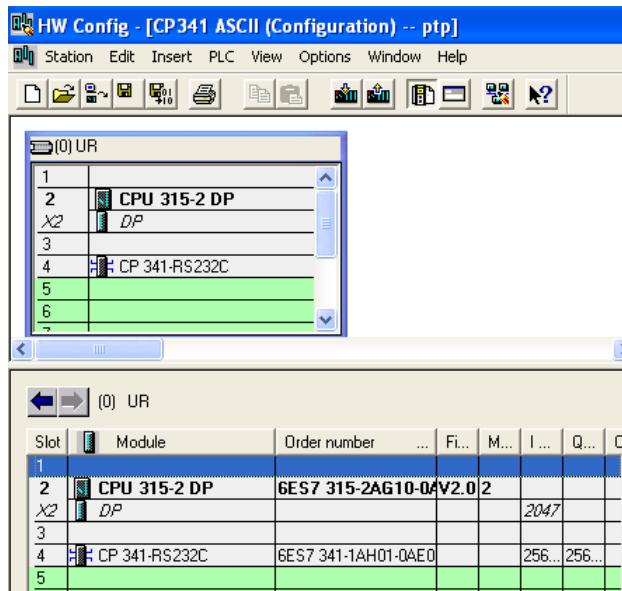


图 20 组态硬件

3、双击 CP341 模板，点击 Parameter...，配置 CP341 参数。

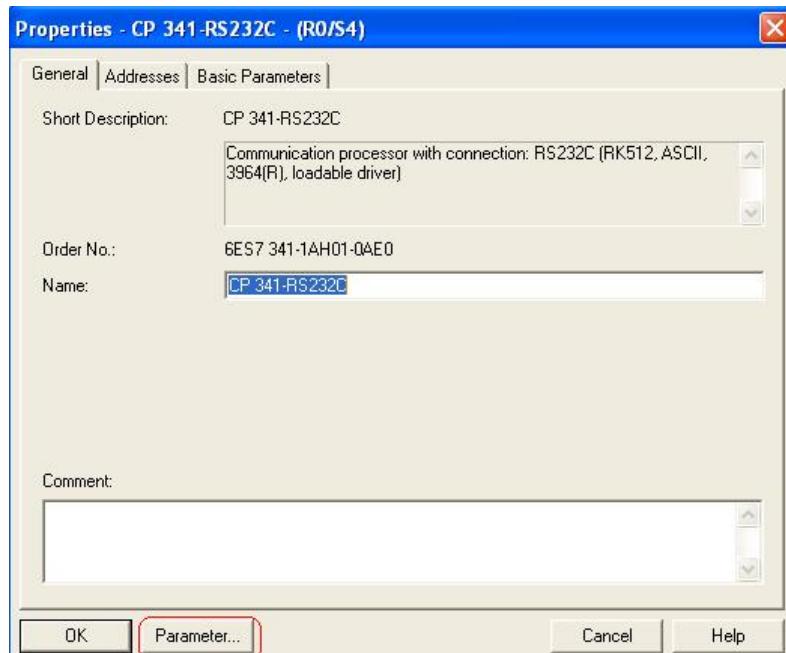


图 21 CP341 属性对话框

3、在 Protocol 选项中选择 ASCII 协议。

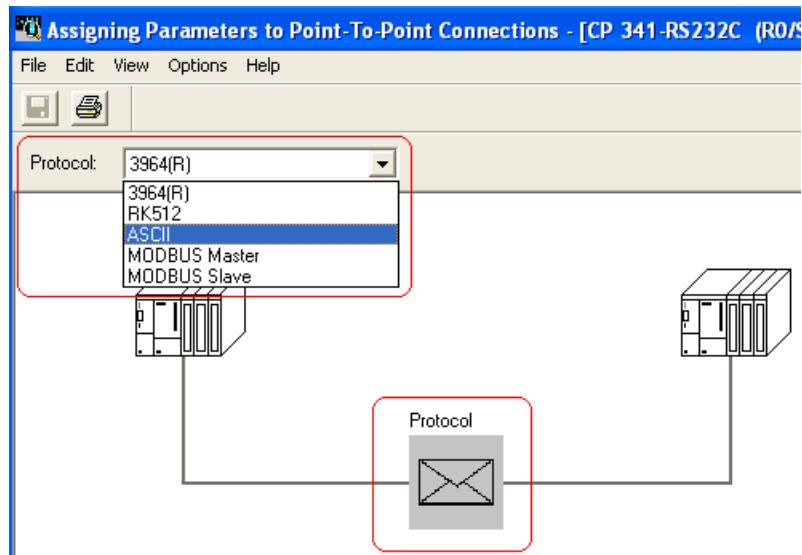


图 22 协议选择

4、双击 Protocol 下的信封图标，配置 ASCII 通信参数，参数的设定要和通讯伙伴的相同。通信波特率：9600bps，数据位：8 位，停止位：1 位，奇偶校验：无，字符延迟时间：4ms，其他参数都采用默认设置。

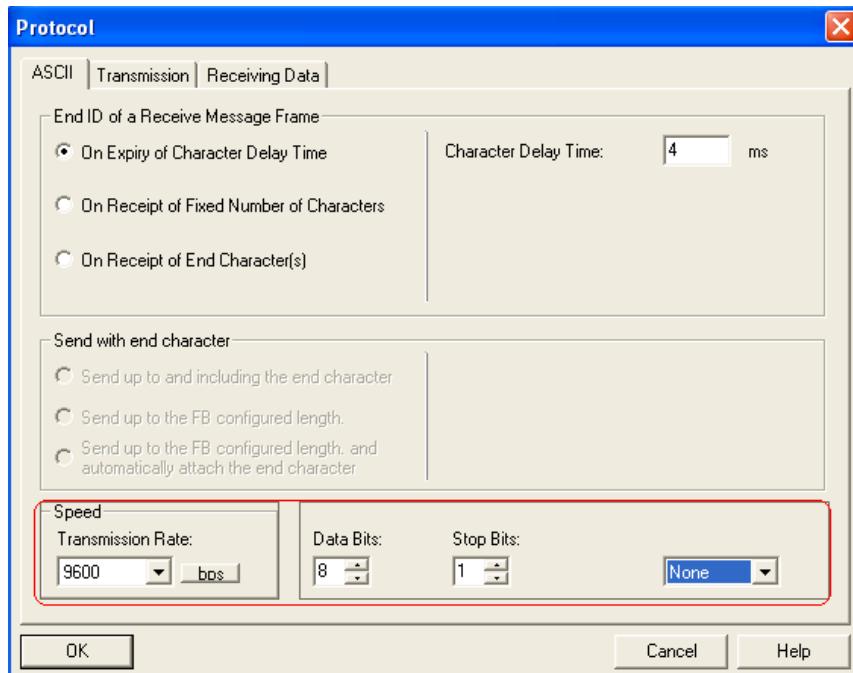


图 23 为 ASCII 协议分配参数

5、如果选用的是 RS422/485 接口的 CP341 时，还需要设置接口属性，如下图所示。  
接收端初始状态可参考上文说明。

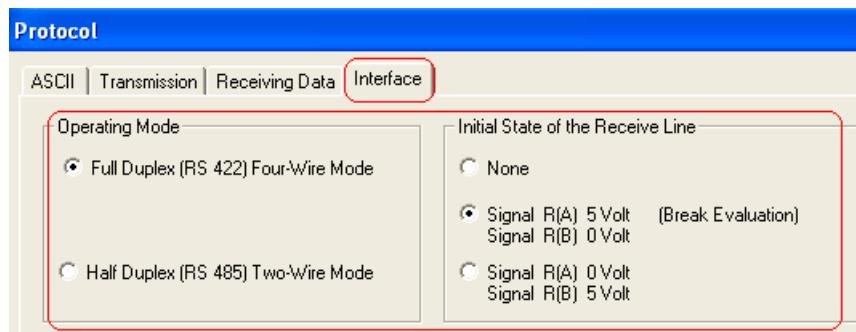


图 24 组态接口参数

6、配置完成后点击 按钮保存编译硬件组态，并确认没有错误后下载到 CPU 中，至此完成了全部的硬件组态过程。

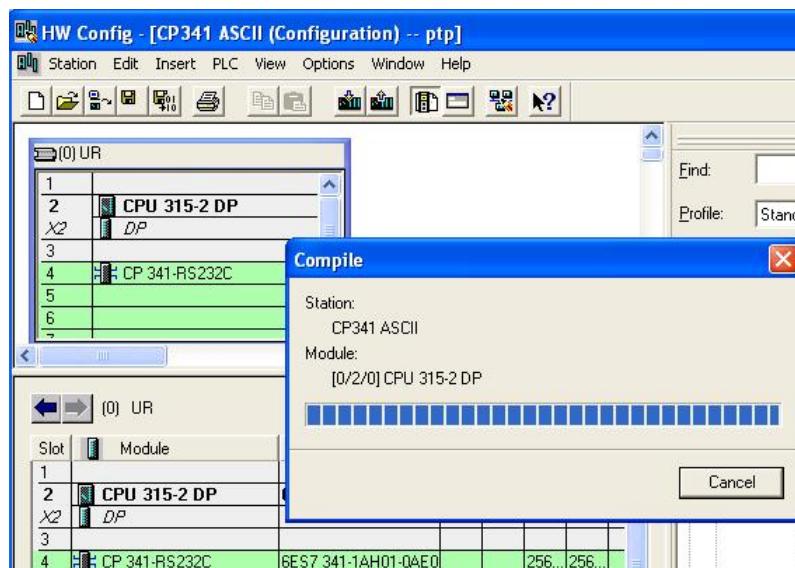


图 25 编译保存硬件组态

#### 4.2.3 编写通信程序

1、双击 OB1，打开 OB1 编程画面，从库 Libraries->CP PtP->CP341 中调用发送功能块 FB8 P\_SND\_RK，为其分配背景数据块 DB8，将参数 LADDR 设为硬件组态中 CP340 的起始逻辑地址 256。

Slot	Module	Order n...	Fi...	M...	I addre...	Q address	Com...
1							
2	CPU 315-2 DP	6E57 315V2.0	2				
X2	DP				2047*		
3							
4	CP 341-RS232C	6E57 341-1			256...271	256...271	
5							

图 26 CP341 逻辑地址

2、创建发送数据块 DB1。

Address	Name	Type	Initial value	Comment
*0.0		STRUCT		
+0.0	SEND_Data	ARRAY[1..50]		Temporary place
*1.0		BYTE		
=50.0		END_STRUCT		

图 27 发送数据块 DB1

3、调用发送功能块

CP341 的发送功能块 FB8 P\_SND\_RK 的参数设置见下表:

LADDR	硬件组态中的起始逻辑地址, 本例中为 256
DB_NO	发送数据块号, 本例中为 1 (DB1)
DBB_NO	发送数据的起始地址, 本例中为 0 (DB1.DBBO)
LEN	发送数据的长度, 本例中为 10
REQ	发送数据触发位, 上升沿触发, 本例中为 M0.0
R	取消通讯,
DONE	发送完成位, 发送完成并没有错误为 TRUE,
ERROR	错误位, 为 TRUE 说明有错误
STATUS	状态字, 标识错误代码, 查看 CP341 手册获得相应的说明
其它参数	与 ASCII 通讯协议无关, 本例中不用

表 5 FB8 P\_SND\_RK 的参数定义

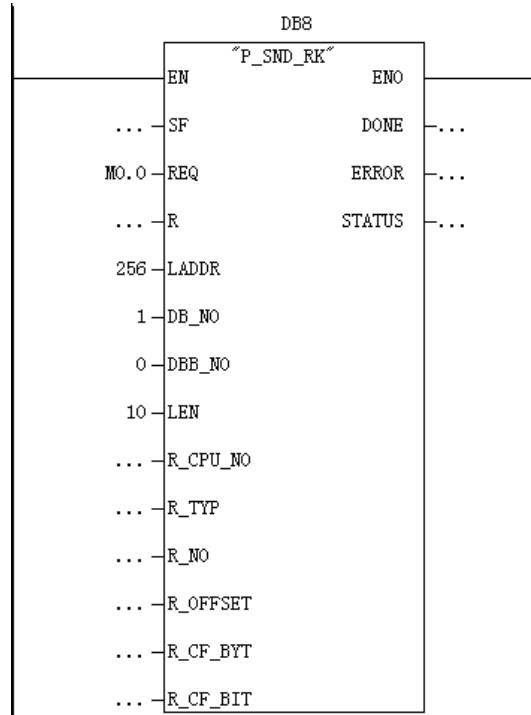


图 28 在 OB1 中调用 FB8 P\_SND\_RK

4、从库 Libraries->CP PtP->CP341 调中用接收程序块 FB7 P\_RCV\_RK，为其分配背景数据块 DB7，将 LADDR 设为硬件组态中的起始逻辑地址 256

### 5、创建接收数据块 DB10

Address	Name	Type	Initial value	Comment
*0..0		STRUCT		
+0..0	SEND_Data	ARRAY[1..50]		Temporary place
*1..0		BYTE		
=50..0		END_STRUCT		

图 29 接收数据块 DB10

### 6、调用接收功能块

CP341 的接收功能块 FB7 P\_RCV\_RK 的参数设置见下表：

LADDR	硬件组态中的起始逻辑地址，本例中为 256
DB_NO	发送数据块号，本例中为 10 (DB10)
DBB_NO	发送数据的起始地址，本例中为 0 (DB10.DBB0)
LEN	接收数据的长度，本例中为 MW2，只有在接收到数据的当前周期，此值不为 0，可以查看 MW4 的值来确认接收到数据的长度
EN_R	使能接收位，本例中为 M0.1
R	取消通讯，本例中不用
NDR	接收完成位，接收完成并没有错误时为 TRUE，
ERROR	错误位，为 TRUE 时说明有错误
STATUS	状态字，标识错误代码，查看 CP341 手册获得相应的说明
其它参数	与 ASCII 通讯协议无关，本例中不用

表 6 FB7 P\_RCV\_RK 的参数定义

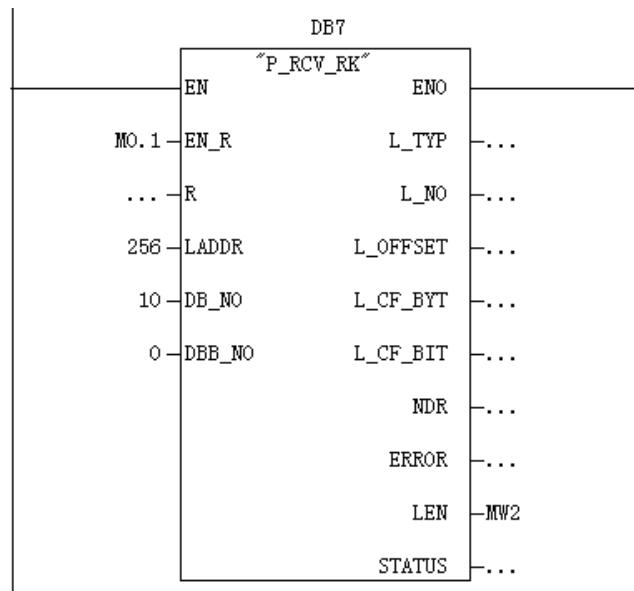


图 30 在 OB1 中调用 FB7 P\_RCV\_RK

#### 4.2.4 设备连接

使用标准的 RS232C 电缆连接 CP341 和计算机的串口，具体电缆的连接方法请查看 CP341 手册 B 章节连接电缆。

#### 4.2.5 通信测试

##### 1、CP341 从 SSCOM32 接收数据

首先将硬件配置和程序下载到 CPU315-2DP 中，将 M0.1 设为 TRUE，使能接收。在计算机上打开软件 SSCOM32，在字符输入窗口输入 16 进制数据 0102030405060708090A，然后点击发送按钮发送数据。



图 31 通过 SSCOM32 发送数据

在 STEP7 中打开接收数据块 DB10 监控接收到的数据，可以看到 SSCOM32 发送的数据被正确的接收。

Address	Name	Type	Initial value	Actual value
0.0	RECEIVE_Data	BYTE	B#16#0	B#16#01
1.0	RECEIVE_Data	BYTE	B#16#0	B#16#02
2.0	RECEIVE_Data	BYTE	B#16#0	B#16#03
3.0	RECEIVE_Data	BYTE	B#16#0	B#16#04
4.0	RECEIVE_Data	BYTE	B#16#0	B#16#05
5.0	RECEIVE_Data	BYTE	B#16#0	B#16#06
6.0	RECEIVE_Data	BYTE	B#16#0	B#16#07
7.0	RECEIVE_Data	BYTE	B#16#0	B#16#08
8.0	RECEIVE_Data	BYTE	B#16#0	B#16#09
9.0	RECEIVE_Data	BYTE	B#16#0	B#16#0A

图 32 监视接收数据块 DB10

## 2、CP341 发送数据到 SSCOM32

将 M0.0 从 FALSE 置成 TRUE，CP341 将发送数据，将 DB1 中前 10 个字节设为十六进制的 31 32 33 34 35 36 37 38 39 3A，可以从 SSCOM32 接收窗口中看到收到的数据（十六进制）。



图 33 通过 SSCOM32 接收数据

如果使用的 CP340 是 RS485 接口的模板，使用计算机串口调试程序时要使用 RS232/RS485 转换器。

## 4.3 CP440 的 ASCII 协议通信

### 4.3.1 硬件列表

RACK-400	6ES7 400-1JA01-0AA0
PS407	6ES7 407-0KA02-0AA0
CPU	6ES7 414-3XJ04-0AB0
CP440	6ES7 440-1CS00-0YE0

### 4.3.2 组态和配置

1、打开创建好的项目 ptp，鼠标右键点击项目名称，选择插入一个新的 SIMATIC 400 Station，更改站的名称为 CP440 ASCII。

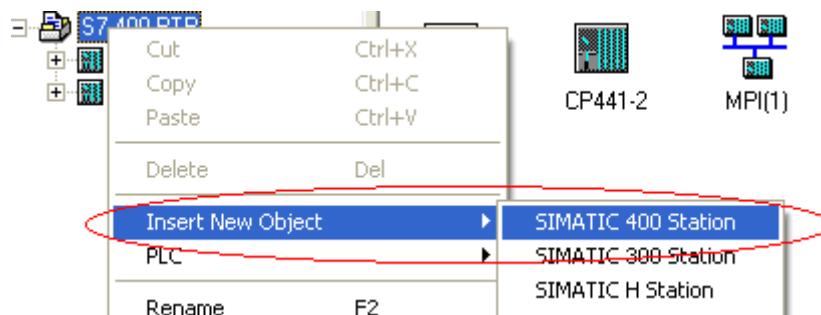


图 34 插入 S7-400 站

2、双击 Hardware 进入硬件配置画面，插入 RACK-400、PS407、CPU414-3DP、CP440。

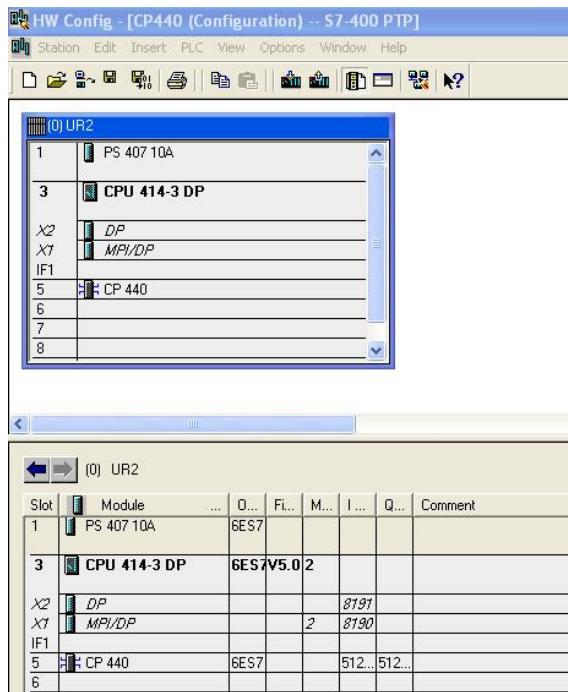


图 35 组态硬件

3、双击 CP440 模板，点击 Parameter 配置 CP440 的参数，在 Protocol 选项中选择 ASCII 协议。

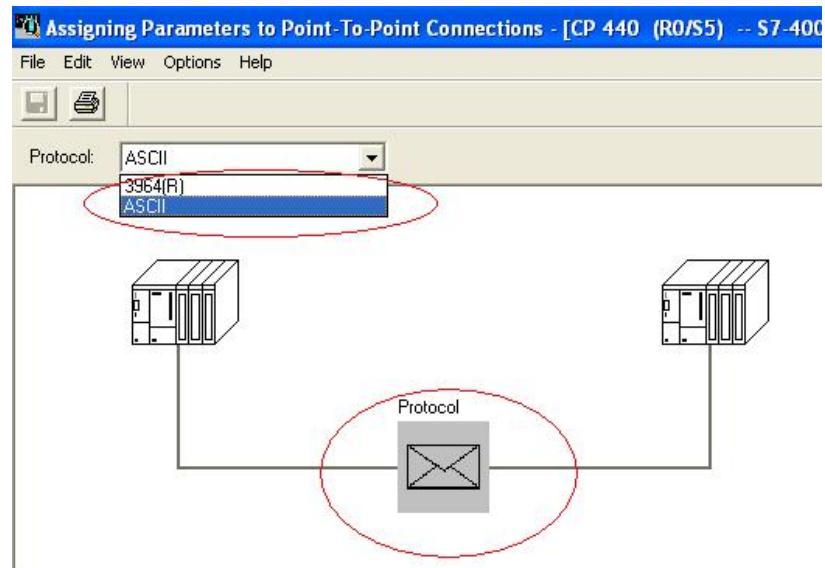


图 36 协议选择

4、双击 Protocol 中的信封图标，配置里面的性能参数，参数的设定一定要和通讯伙伴的相匹配，波特率：9600bps，数据位：8 位，停止位：1 位，奇偶校验：无，字符延迟时间：4ms，其余参数可使用默认数据。

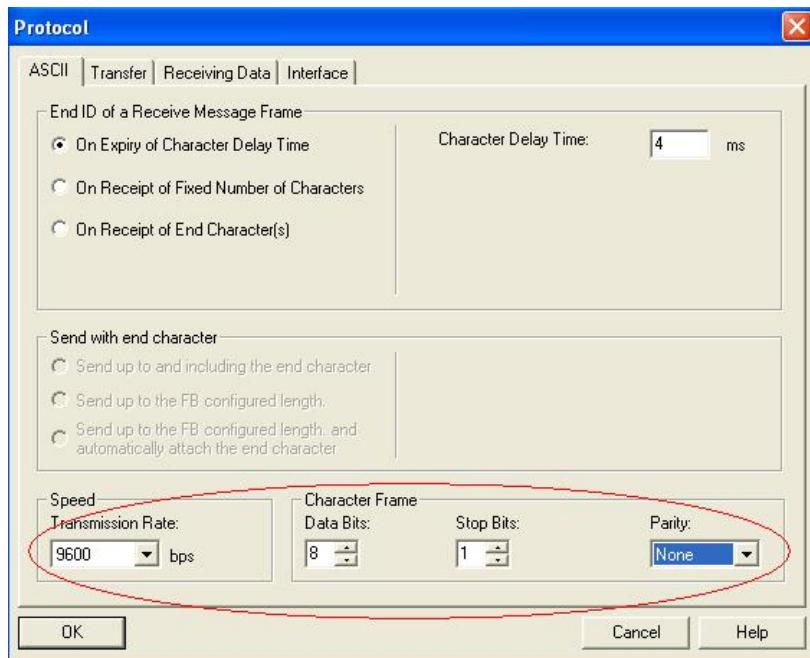


图 37 为 ASCII 协议分配参数

5、因为 CP440 只有一种 RS422/485 接口类型，所以如果与通讯伙伴接口不同时需要外接转换设备，其他参数设置与 CP340/341 作 ASCII 通讯时均相同。

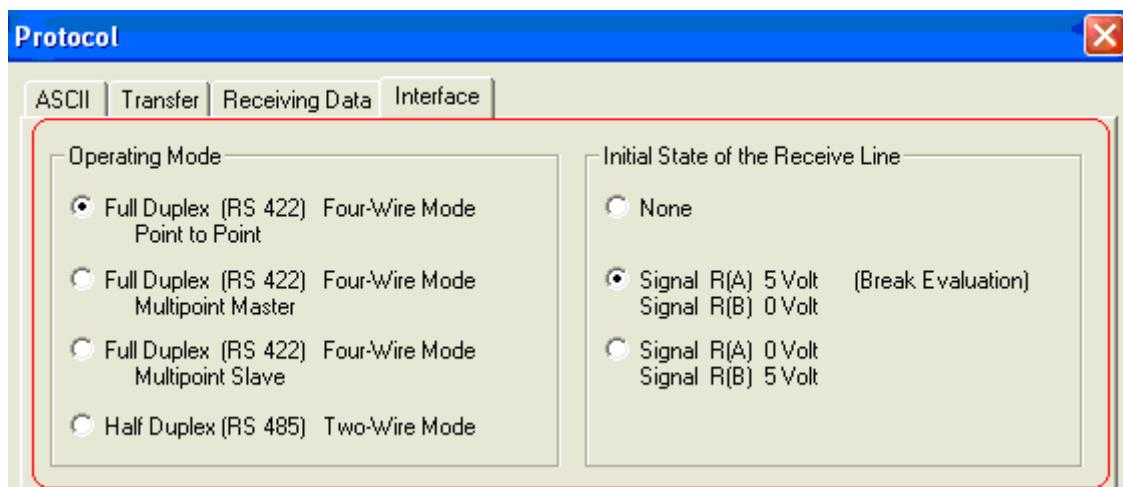


图 38 组态接口参数

6、配置完成后编译并保存硬件组态，确认无错误后下载。

### 4.3.3 编写通信程序

1、双击 OB1，打开编程界面，从库 Libraries→CP PtP→CP440 中调用发送程序块 FB10 SEND\_440，为其分配背景数据块 DB10，将参数 LADDR 设为硬件组态中的起始逻辑地址 512。

Slot	Module	...	O...	Fi...	M...	I address	Q address	Comment
1	PS 407 10A	6ES7						
3	CPU 414-3 DP	6ES7V5.0	2					
X2	DP				8191*			
X1	MPI/DP			2	8190*			
IF1								
5	CP 440	6ES7			512..527	512..527		
6								
7								

图 39 CP440 逻辑地址

2、创建一个发送数据块 DB1。

Address	Name	Type	Initial value	Comment
*0.0		STRUCT		
+0.0	send	ARRAY[1..20]		Temporary
*1.0		BYTE		
=20.0		END STRUCT		

图 40 发送数据块 DB1

3、调用发送数据块

CP440 的发送功能块 FB10 SEND\_440 的参数设置见下表：

名称	数据类型	注释
REQ	BOOL	数据发送触发位，上升沿触发，本例为 M0.0
R	BOOL	取消发送，通讯中断
LADDR	INT	硬件组态中的模块起始逻辑地址，本例为 512
DB_NO	INT	发送数据块号，本例为 1 (DB1)
DBB_NO	INT	发送数据的起始地址，本例中为 0 (DB1.DBBO)
LEN	INT	发送数据的长度，本例中为 15
DONE	BOOL	发送完成位，发送无错误完成时为 TRUE
ERROR	BOOL	错误位，发送有错误时为 TRUE
STATUS	WORD	状态字，表示错误代码，如果 ERROR=1，此状态字中包含错误信息，可在线查看信息代码含义或从 CP440 手册中获得相应说明。

表 7 FB10 SEND\_440 的参数定义

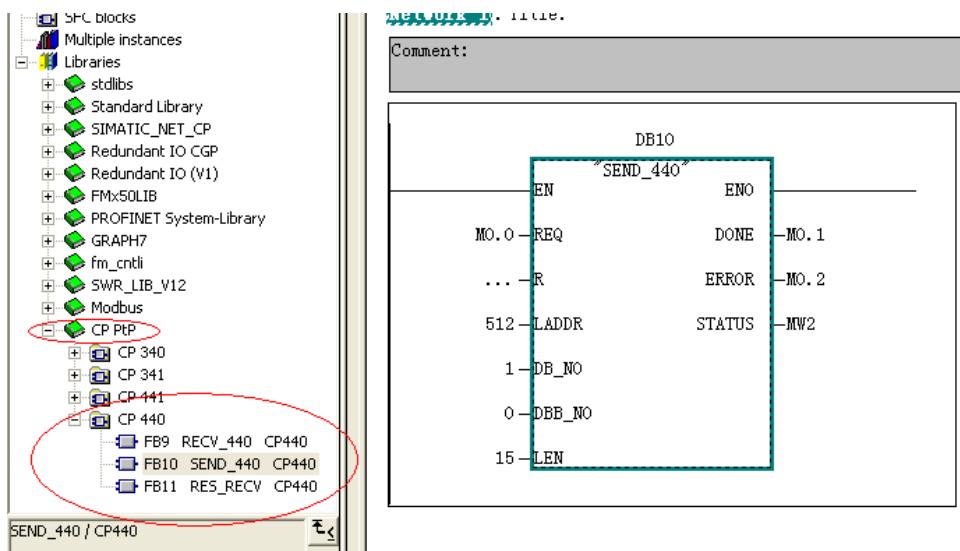


图 41 在 OB1 中调用 FB10 SEND\_440

4、从库中相同文件夹中调用接收功能块 FB9 RECV\_440，为其分配背景数据块 DB9，将参数 LADDR 设为硬件组态中的输入起始逻辑地址 512。

5、创建接收数据块 DB2。

6、调用接收功能块

CP440 的接收功能块 FB9 RECV\_440 的参数设置见下表：

名称	类型	数据格式	注释
EN_R	输入	BOOL	接收使能位，本例中为 M1.0
R	输入	BOOL	取消通讯，放弃接收
LADDR	输入	INT	硬件组态中的模块起始逻辑地址，本例为 512
DB_NO	输入	INT	接收数据块号，本例为 2 (DB2)
DBB_NO	输入	INT	接收数据的起始地址，本例中为 0 (DB2.DBB0)
NDR	输出	BOOL	接收完成位，接收无错误完成后为 TRUE
ERROR	输出	BOOL	错误位，接收有错误时为 TRUE
LEN	输出	INT	接收数据的长度，以字节为单位，只有在接收到数据的当前周期此值不为 0
STATUS	输出	WORD	状态字，表示错误代码，如果 ERROR=1，此状态字中包含错误信息，可在线查看信息代码含义或从 CP440 手册中获得相应说明。

表 8 FB9 RECV\_440 的参数定义

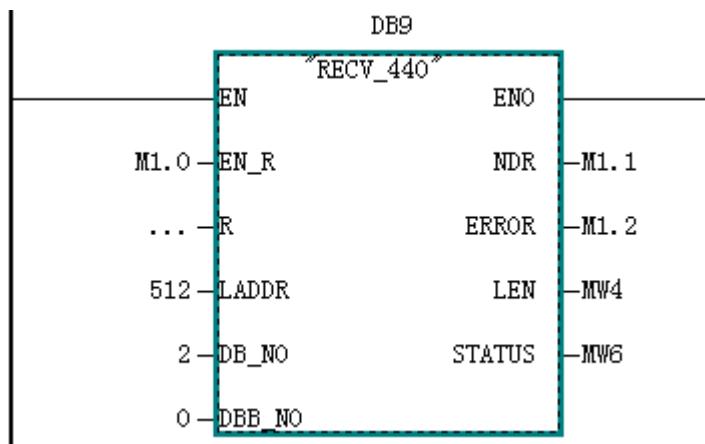


图 42 在 OB1 中调用 FB9 RECV\_440

#### 4.3.4 设备连接

CP440 模块只提供一个 15 针 RS422/485 串行接口，根据接线方式来选择使用 RS422 或者 RS485 接口，只有一个接口有效。其中 RS422 为 4 线制，全双工模式；RS485 为 2 线，半双工模式。具体电缆的连接方法请查看 CP440 手册附录 B Connectiong cables。

#### 4.3.5 通信测试

具体的通信测试过程与上文 CP340/341 ASCII 数据通讯基本一致，在此不再赘述，请参考上文。

#### 4.4 CP441 的 ASCII 协议通信

CP441 通讯模块可根据实际应用选择不同接口类型的子模块，从而实现与一个或多个通讯伙伴的通讯。CP441-1 (1 个可选子模块接口)，CP441-2 (2 个可选子模块接口)，子模块接口类型共分为 3 种：

订货号	接口类型	注释
6ES7963-1AA00-0AA0	RS232C	9 针 D 型针接头
6ES7963-2AA00-0AA0	20mA TTY	9 针 D 型孔接头
6ES7963-3AA00-0AA0	RS422/485	15 针 D 型孔接头

表 9 CP441 接口子模块

下面以 CP441-2 为例说明组态过程。

#### 4.4.1 硬件列表

RACK-400	6ES7 400-1JA01-0AA0
PS407	6ES7 407-0KA02-0AA0
CPU	6ES7 414-3XJ04-0AB0
CP441-2	6ES7 441-2AA03-0AE0 (Interface1: RS232C, 本例使用接口) (Interface2: RS422/485)

#### 4.4.2 组态和配置

1、打开已创建好的项目 ptpt，插入一个新的 SIMATIC 400 Station，更改站的名称为 CP441-2 ASCII。双击 Hardware，进入硬件配置画面，按上文所述步骤插入相应的模块。

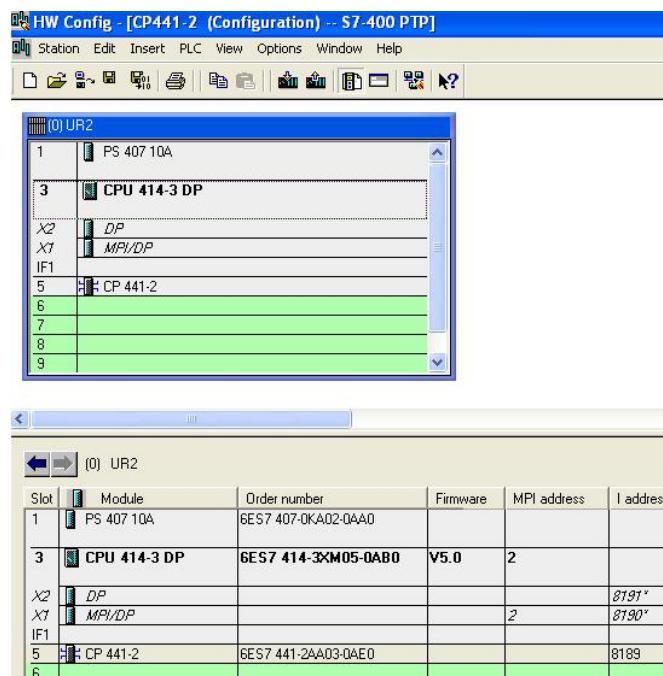


图 43 组态硬件

2、双击 CP441-2 模板，点击进入 Basic Parameters，选择 Interface 地址，并指定子模块的接口类型，本例中使用 RS232C 接口。

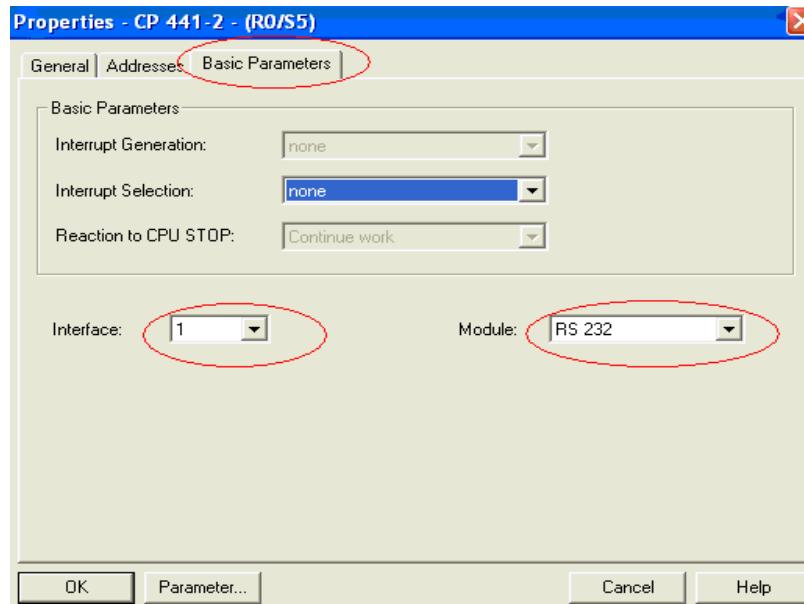


图 44 接口基本参数设置

3、点击 Parameter，配置 CP441-2 参数。

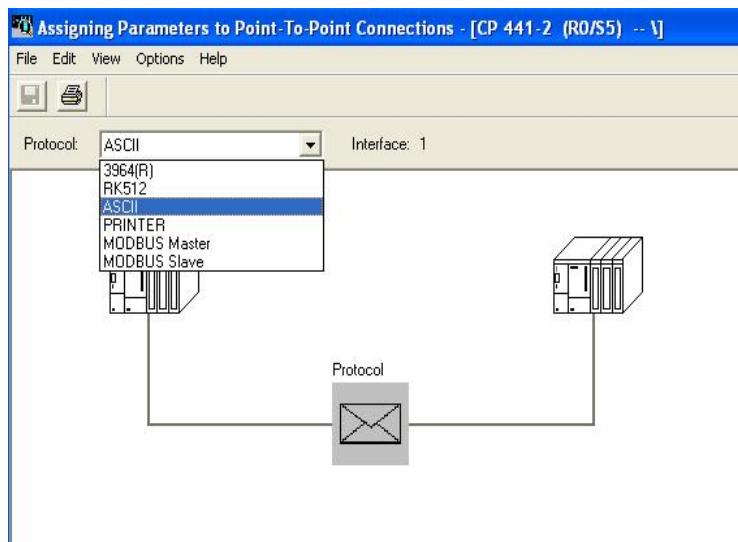


图 45 协议选择

4、双击 Protocol 下的信封图标，配置 ASCII 参数，参数的设定要和通讯伙伴的相同，其他参数都采用默认数据。具体操作可参考上文，此处不再赘述。

5、配置完成后回到选择子模块接口类型的界面，点击 General 进入 Interface→PtP 1，新建一条 PtP 子网，并点击 OK 确认。

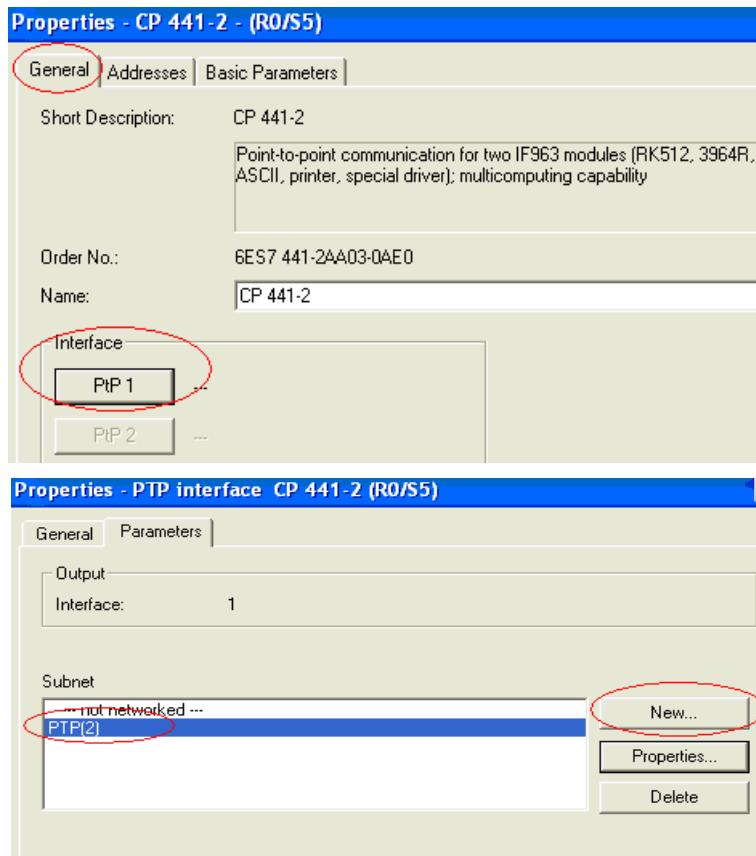


图 46 新建 PtP 网络

此时 PtP 1 的接口状态为连接状态。PtP2 的配置与此相同，由于本例没有使用，故不做说明。



图 47 建立 PtP 的连接后的接口状态

6、然后在硬件组态画面中打开网络组态图标 (configure network)  进入网络配置界面。

点击选中本站中的 CPU，此时可看到页面下方出现了通讯连接表，鼠标右键单击第一行插入新的连接，连接类型为“point-to-point connection”。

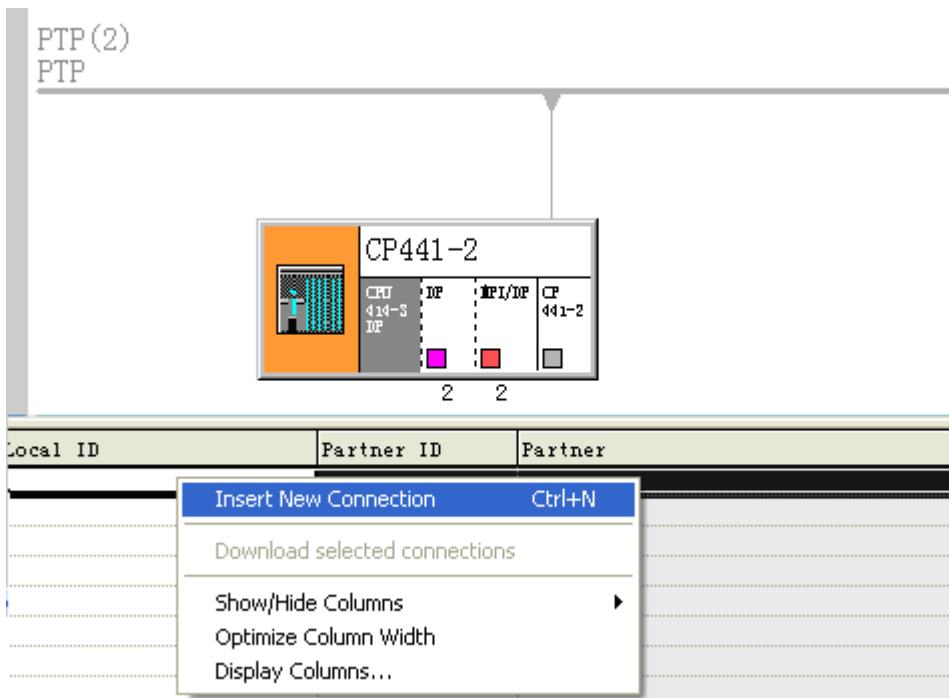


图 48 插入一个新的连接

7、选择连接对象为“Unspecified”，此选项用于和第三方设备以及不带通讯总线的 S7-CP PtP(CP340,CP341)、S5-CP PtP、Printer 等设备的连接；如果通讯伙伴同样是 CP441，要确认其已连接到同一条 PtP 子网上，并直接选择它为连接对象。

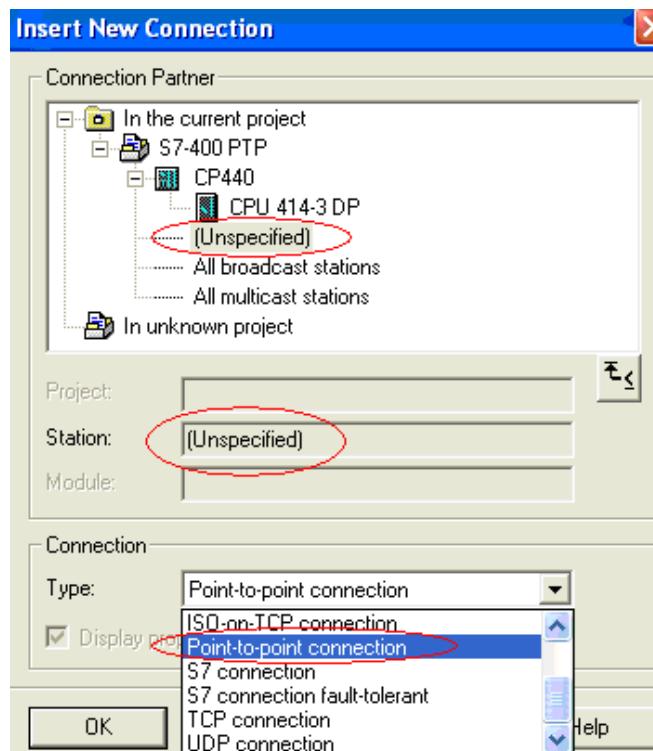


图 49 插入一个新的连接

8、配置完成后点击 OK 确认，此时弹出 PtP 连接属性画面。

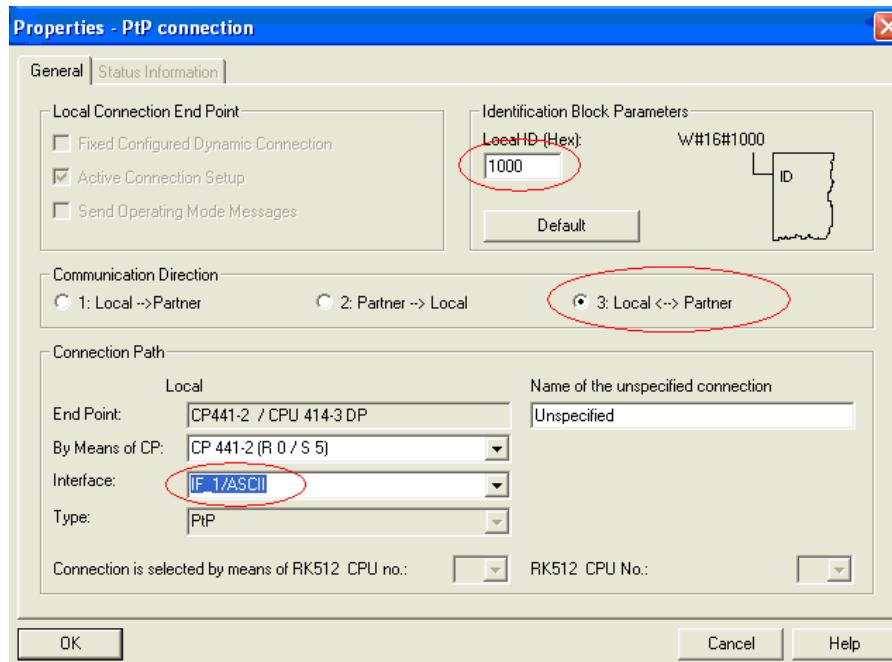


图 50 组态连接

- 在此画面中有一参数 Local ID(Hex) 需要记下，在后面调用系统功能块进行编程时要用到它，您也可以手动修改它，以适应程序中 SFB 组态好的参数，但要注意其范围：W#16#1000---W#16#1400。
- “Communication Direction” 通讯方向的选择可根据实际需要判定，本例使用第三种，双向通讯。点击 OK 确认组态并编译，确认无误后下载。

9、在线确认连接状态，点击 ，如下图，可以看到此时“Connection status”中的状态为“set up”，说明连接已建立。

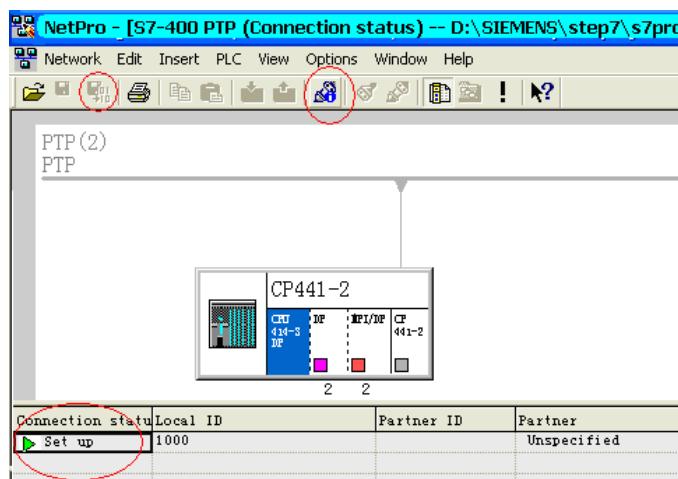


图 51 连接状态监控

#### 4.4.3 编写通信程序

1、双击 OB1，打开编程界面，从库中 Libraries→Standard library→System function blocks 中调用 SFB12 BSEND，为其分配背景数据块 DB12，并新建发送数据块 DB1。

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	SENDDATA	ARRAY[0..10]		Temporary
*1.0		BYTE		
=12.0		END_STRUCT		

图 52 发送数据块 DB1

2、调用发送功能块，SFB12 BSEND 的参数设置参见下表：

名称	数据类型	注释
REQ	BOOL	发送请求位，上升沿触发，本例使用 M0.0
R	BOOL	取消通讯
ID	WORD	Local ID 号，在 PtP 连接属性中获得，十六进制输入
R_ID	DWORD	区分双字，可以是任何数据，但如果在程序中使用了多个 BSEND 块，一定要分配不同的编号。
SD_1	ANY	数据发送区，本例使用 DB 区，发送数据为 5 个字节 也可使用其它地址和数据类型进行访问，具体信息可参考 Step7 SFB12 在线帮助。
LEN	WORD	发送数据长度，以字节为单位计数
DONE	BOOL	发送完成位，任务完成且无错误时置 1
ERROR	BOOL	错误位，有错误时置 1
STATUS	WORD	状态字，标示错误代码，具体信息可参考模块在线诊断内容或 CP441 手册第九章诊断信息

表 10 SFB12 BSEND 的参数定义

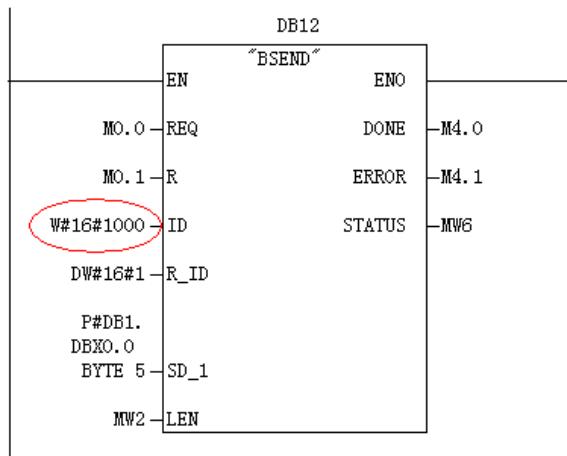


图 53 在 OB1 中调用 SFB12 BSEND

3、同时调用 SFB13 BRCV 进行数据接收，分配背景数据块 DB13，并新建接收数据的 DB 块 DB2。

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	RCV	ARRAY[0..400]		Temporary
*1.0		BYTE		
=402.0		END_STRUCT		

图 54 接收数据块 DB2

名称	数据类型	注释
EN_R	BOOL	接收使能位，为 1 时允许接收，本例使用 M8.0
ID	WORD	Local ID 号，在 PtP 连接属性中获得，十六进制输入
R_ID	DWORD	做 ASCII ,3964(R)通讯时必须使用 W#16#0
RD_1	ANY	数据接收区，本例使用 DB2 进行接收
LEN	WORD	接收数据的长度存储区，以字节单位
NDR	BOOL	接收完成位，接收无错误完成时为 1
ERROR	BOOL	错误位，为 1 时说明有错误
STATUS	WORD	状态字，标示错误代码，具体信息可参考模块在线诊断内容或 CP441 手册第九章诊断信息

表 11 SFB13 BRCV 的参数定义

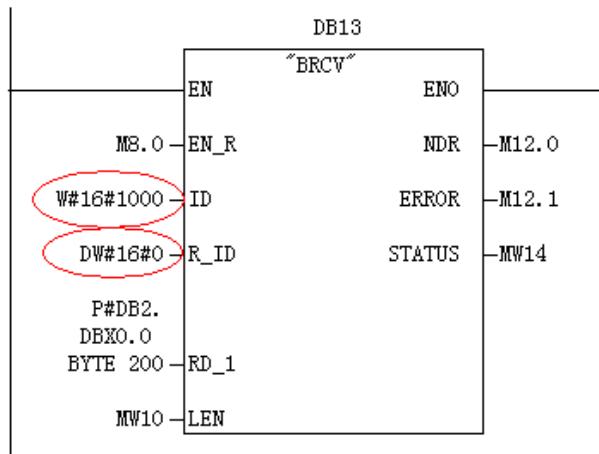


图 55 在 OB1 中调用 SFB13 BRCV

#### 4.4.4 设备连接

使用标准的 RS232C 电缆连接 CP441-2 interface1 和计算机的串口，具体电缆的连接方法请查看 CP441 手册附录 B Connecting cables

#### 4.4.5 通信测试

使用测试软件 SSCOM32 进行数据的通讯测试，将 M8.0 使能为 TRUE，允许 DB2 接收，将 M0.0 从 FALSE 置成 TRUE，触发 CP441-2 发送 DB1 的数据，过程如下图：

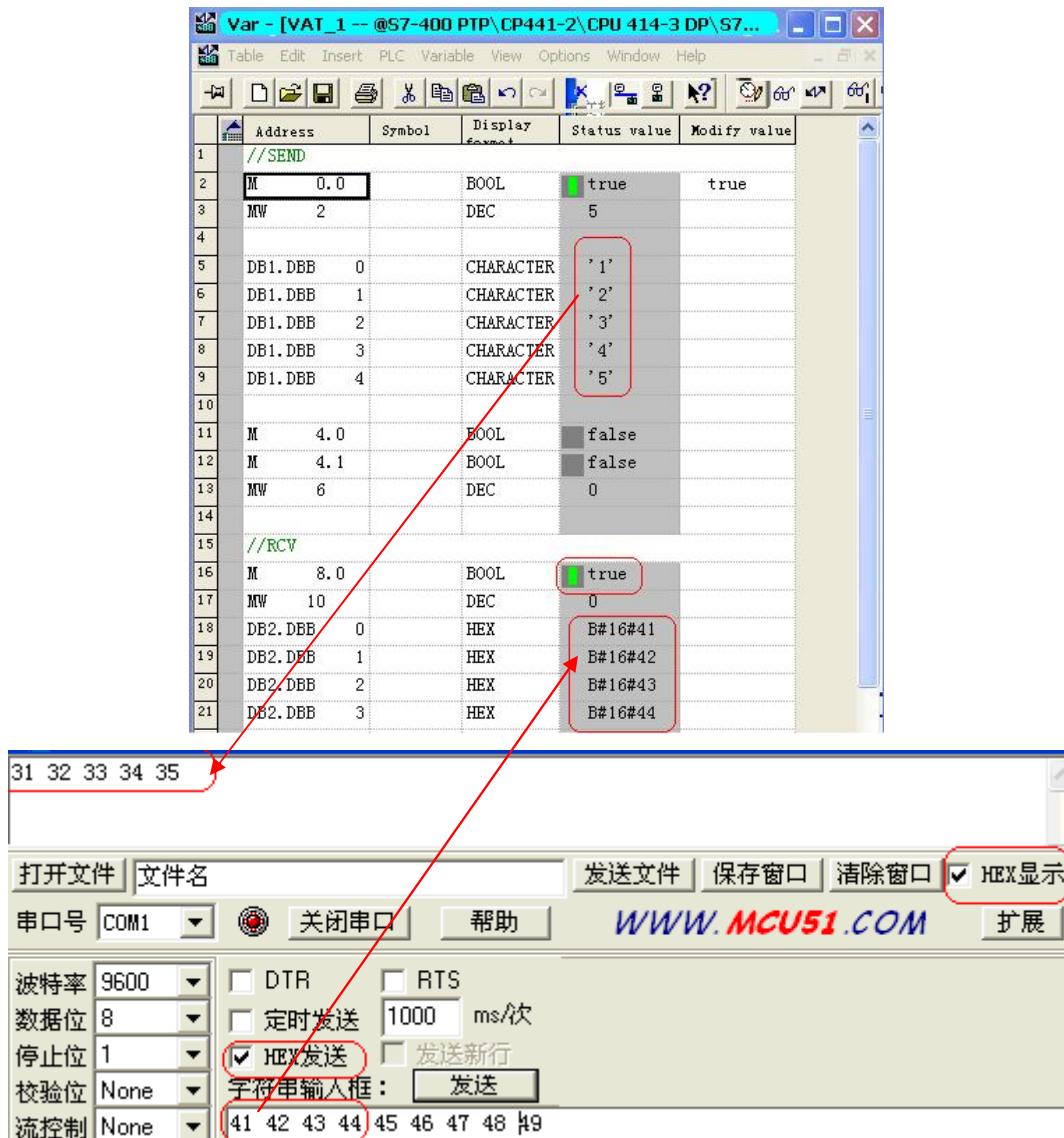


图 56 通信状态监视

## 5 MODBUS RTU 协议通信

MODBUS RTU 通讯协议是主从模式的通信，在传输的过程中主站主动发送请求报文到从站，从站返回应答报文，具体 MODBUS RTU 有关协议的内容请参阅 MODBUS RTU 相关文档或手册。

### 5.1 CP341 的 MODBUS Slave 协议通信

#### 5.1.1 硬件列表

PS 307	6ES7 307-1EA00-0AA0
CPU 315-2DP	6ES7 315-2AG10-0AB0
MMC	6ES7 953-8LG11-0AA0
CP341	6ES7 341-1AH01-0AE0
Dongle	6ES7 870-1AB01-0YA0

应用 CP341 进行 MODBUS 协议通信时需要有 Dongle 的支持。在使用之前先将 Dongle 安装在 CP341 模块的背面的 Dongle 插槽中，Dongle 和插入 Dongle 前后的 CP341 如下图所示：

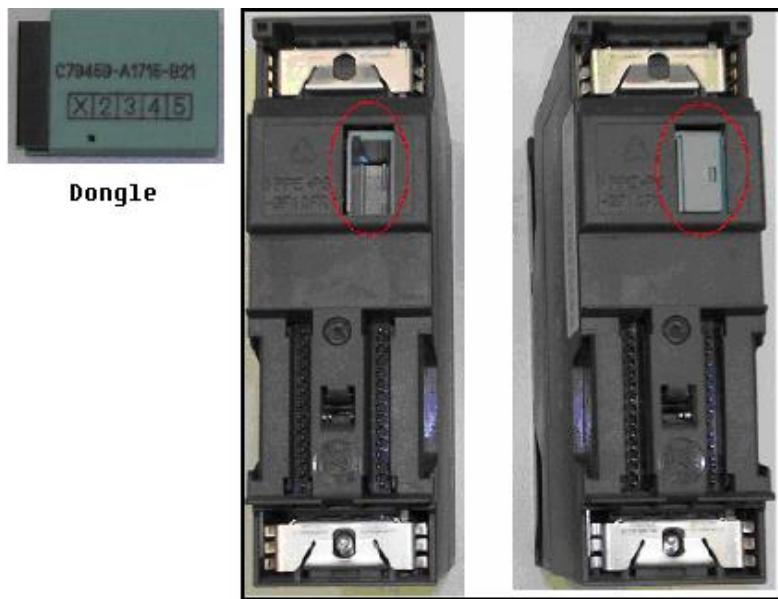


图 57 Dongle 插入前后对比

### 5.1.2 组态和配置

1、打开上面创建的项目 ptp，用鼠标右键点击项目名称，选择 Insert New Object-> SIMATIC 300 Station，更改站的名称为 CP341 Modbus-S。

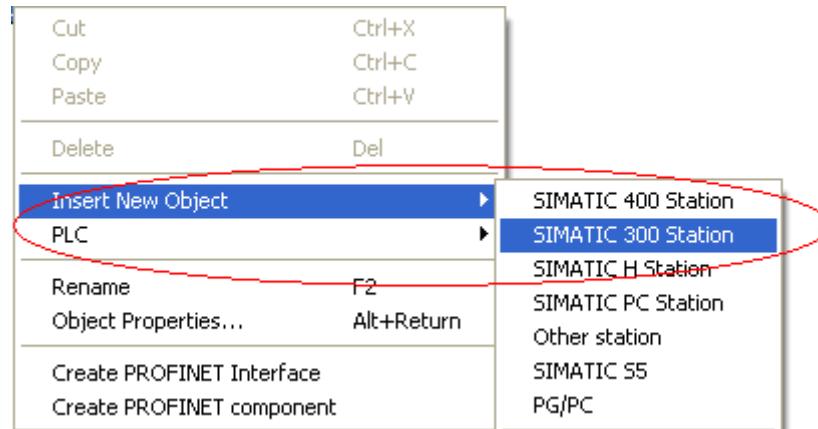


图 58 插入 S7-300 站

2、双击 Hardware 进入硬件配置画面，插入 RACK、CPU315-2DP、CP341。

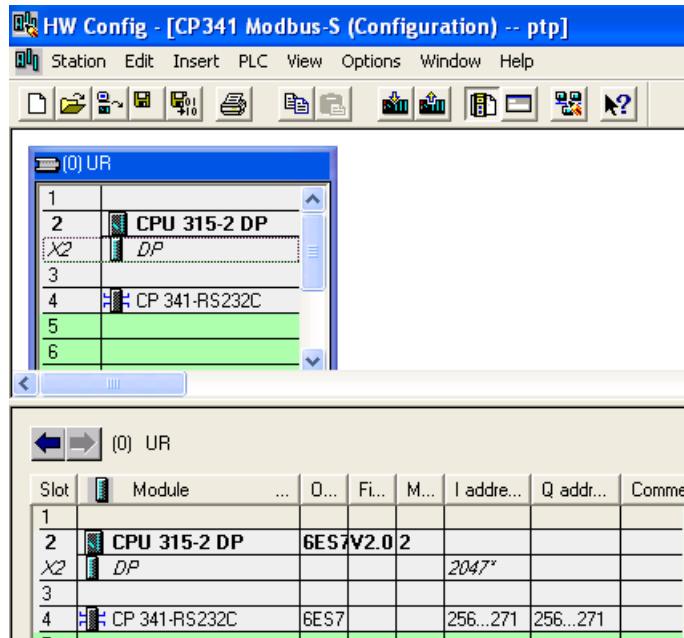


图 59 组态硬件

3、双击 CP341 模板，点击 Parameter...，配置 CP341 参数。在 Protocol 选项中选择 MODBUS Slave。

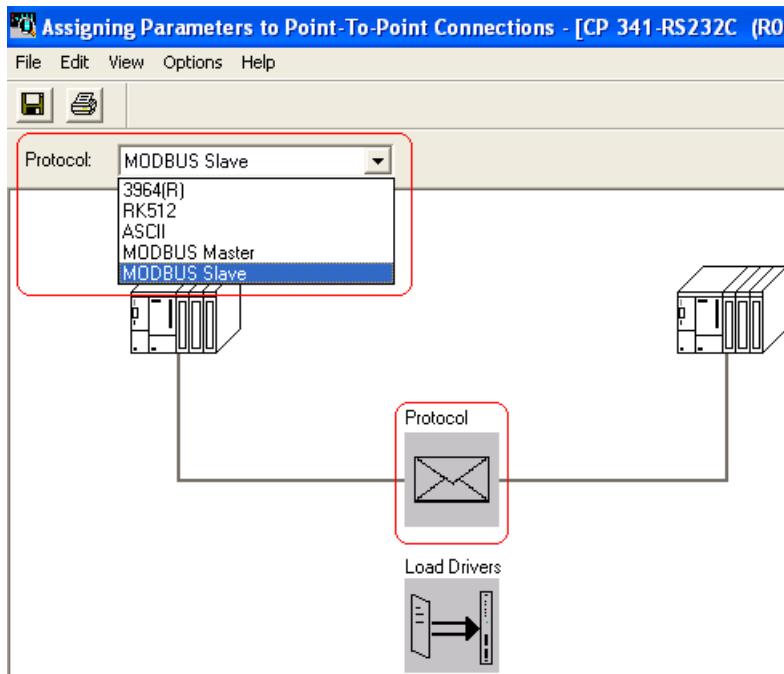


图 60 选择通信协议

4、双击 Protocol 下的信封图标，配置 MODBUS Slave 参数，点击 MODBUS-Slave 按钮。

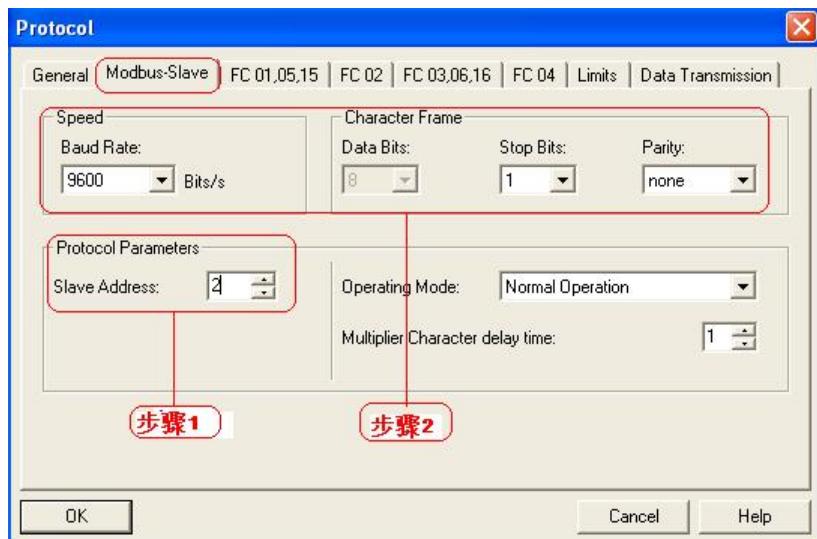


图 61 组态 MODBUS Slave 通信参数

- 步骤 1：设置 MODBUS 从站地址，本例中设为 2；
- 步骤 2：设置 MODBUS 从站波特率，停止位，校验位等参数；
- 接下来配置功能代码所表示的参数，具体参见下面 4 图

5、FC01,05,15：读取、强制输出位的状态；左边的地址为信息传送地址，右边对应西门子的 PLC 地址区，即左边地址从 0~100 对应 MODBUS 地址区为 00001~00101，对应西门子数据区为 M0.0~M12.4；101~200 对应 MODBUS 地址区为 00102~00201，对应西门子数据区为 Q0.0~Q12.3；从地址 201~300, 301~400 对应 Modbus 地址区为 00202~00301, 00302~00401，对应西门子数据区为 Timer, Counter。

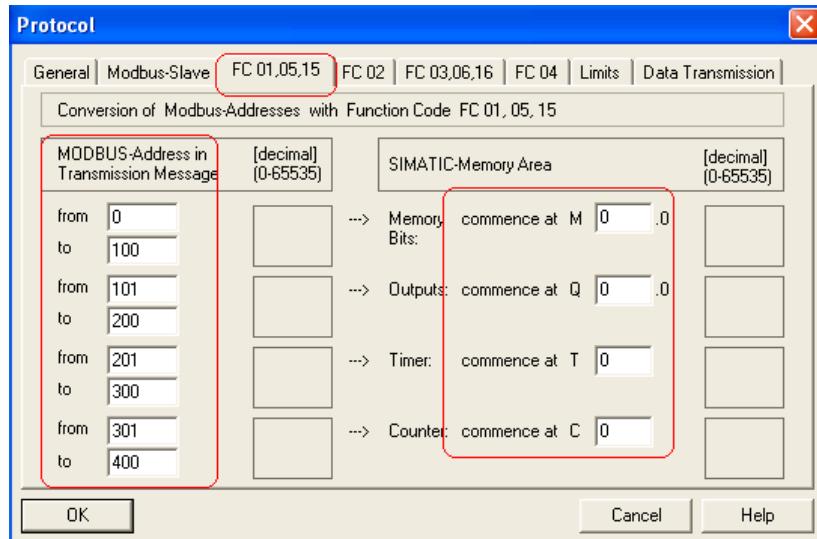


图 62 MODBUS Slave 地址分配

6、FC02 读取输入数据位的状态，地址对应如上文所述。

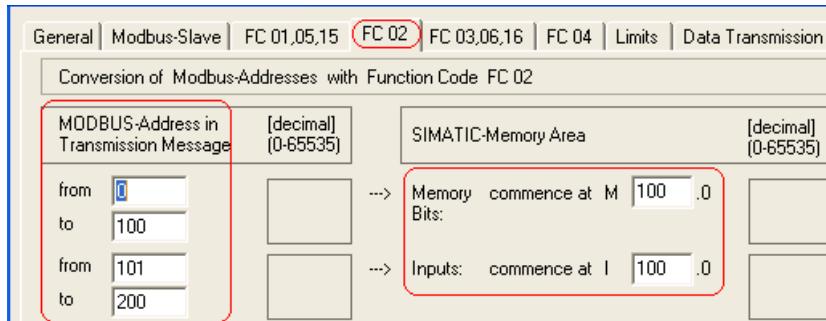


图 63 MODBUS Slave 地址分配

7、FC03,06,16 组态输出寄存器数据区，对应西门子数据区为 DB 块。

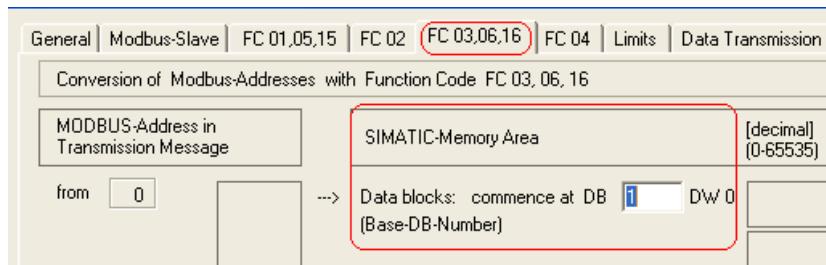


图 64 MODBUS Slave 地址分配

8、FC04 组态输入寄存器数据区，对应西门子数据区同样是 DB 块。

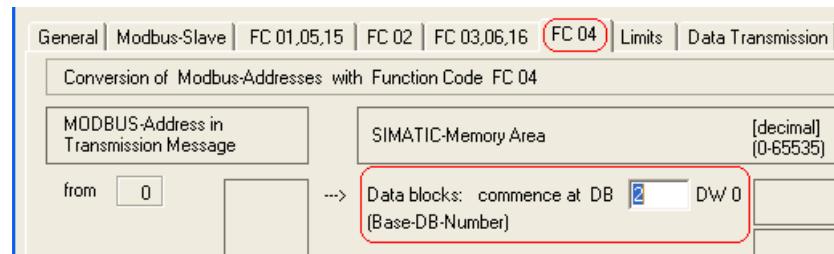


图 65 MODBUS Slave 地址分配

9、设置写参数的限制值。

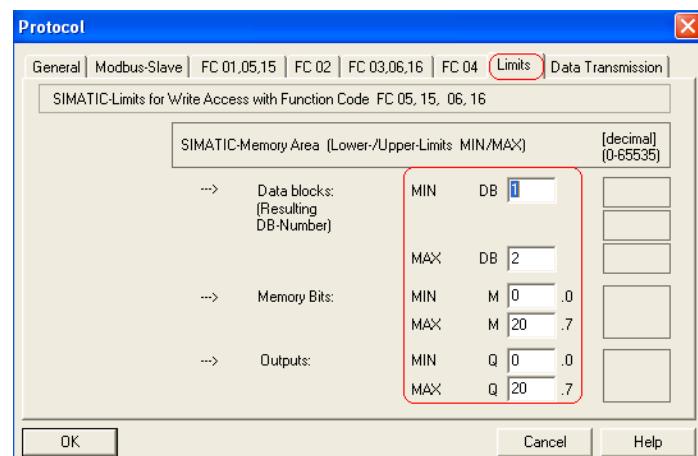


图 66 MODBUS Slave 写参数限制值

10、如果选用的是 RS422/485 接口的 CP341 时，还需要设置接口类型，如下图所示，其它参数均采用默认设置。

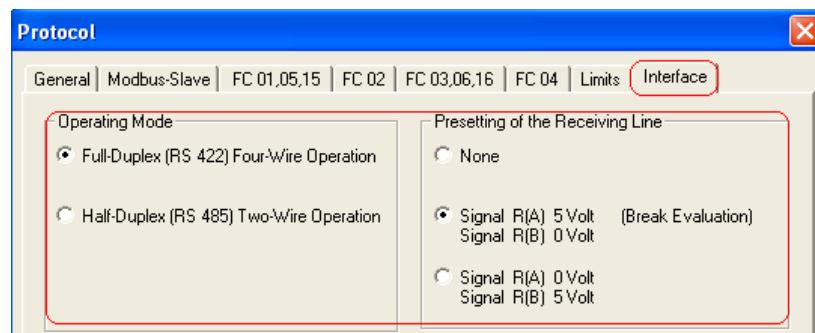


图 67 MODBUS Slave 接口参数分配

11、配置完成后保存时会提示是否装载驱动。

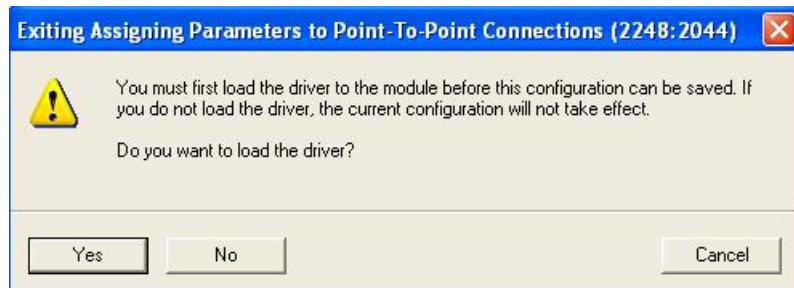


图 68 提示装载驱动对话框

12、此时必须连接到实际的 PLC，点击 Yes 按钮装载驱动，装载时 CPU 必须为 STOP 模式。驱动装载完成后，如果再次装载，STEP7 会提示 Driver already exists，配置完成后点击  按钮保存编译硬件组态，并确认没有错误，

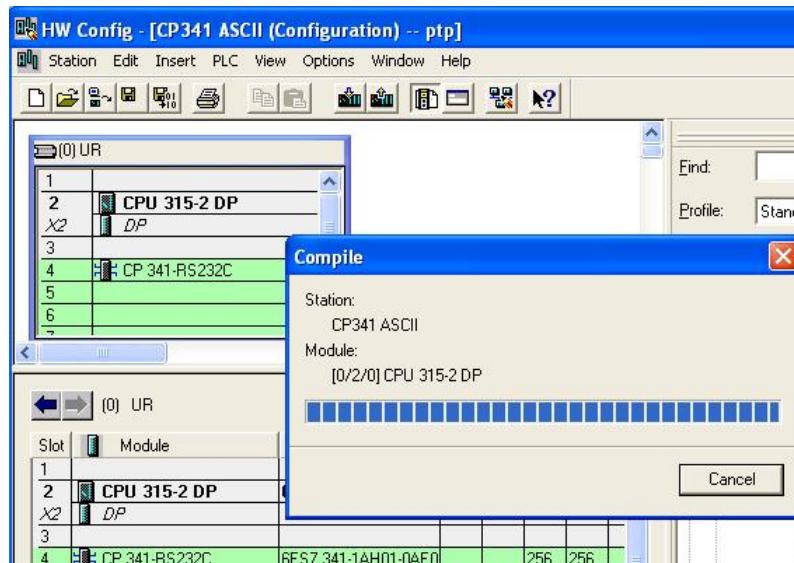


图 69 编译保存硬件组态

### 5.1.3 编写通信程序

1、将 FB7, FB8 添加到程序 Blocks 中，方法为双击 OB1，打开 OB1 编程画面，从库 Libraries->CP PtP->CP341 中调用发送程序块 FB8 和接收程序块 FB7，然后再从 OB1 中删除。因为 MODBUS 从站通讯要用到这两个功能块。

2、调用 MODBUS 从站功能块 FB80，位置在 Libraries->Modbus->Modbus ->FB80。分配背景数据块 DB80，将参数 LADDR 设为硬件组态中的起始逻辑地址 256。

Slot	Module	Order n...	Fi...	M...	I addre...	Q address	Com...
1							
2	CPU 315-2 DP	6ES7 315V2.0	2				
X2	DP				2047*		
3							
4	CP 341-RS232C	6ES7 341-1			256...271	256...271	
5							

图 70 CP341 逻辑地址

## 2、调用 MODBUS 从站功能块

LADDR	硬件组态中的起始逻辑地址, 本例中为 256
START_TIME	超时初始化定时器, 本例中为 T1
START_TIME	超时初始化时间值, 本例中为 1s
OB_MASK	外设访问错误屏蔽位, 本例中为 M100.0
CP_START	FB 初始化始能位, 本例中为 M100.1
CP_START_FM	CP_START 初始化的上升沿位, 本例中为 M100.2
CP_START_NDR	从 CP 写操作位, 本例中为 M100.3
CP_START_OK	初始化成功标志, 本例中为 M100.4
CP_START_ERROR	初始化失败标志, 本例中为 M100.5
ERROR_NR	错误号, 本例中为 MW102
ERROR_INFO	错误信息, 本例中为 MW104, 可以查看 Modbus Slave 手册第 10 章

表 12 FB80 的参数定义

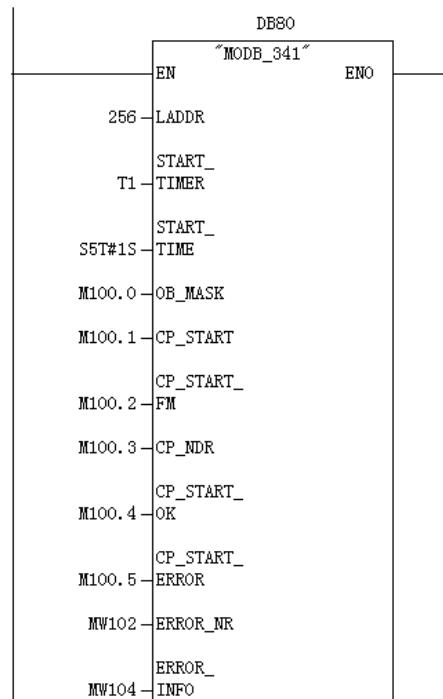


图 71 在 OB1 中调用 FB80

3、创建 FC03,06,16 功能代码通讯数据块 DB1。

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	FC030616_Data	ARRAY[1..50]		Temporary
*1.0		BYTE		
=50.0		END_STRUCT		

图 72 通信数据块 DB1

4、创建 FC04 功能代码通讯数据块 DB2。

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	FC04_Data	ARRAY[1..50]		Temporary
*1.0		BYTE		
=50.0		END_STRUCT		

图 73 通信数据块 DB2

#### 5.1.4 设备连接

使用标准的 RS232C 电缆连接 CP341 和计算机的串口，具体电缆的连接方法请查看 CP341 手册章节 B Connecting Cables，

#### 5.1.5 通信测试

1、首先将硬件配置和程序下载到 CPU315-2DP 中。在计算机上打开 MODBUS 主站仿真软件 Modscan32。

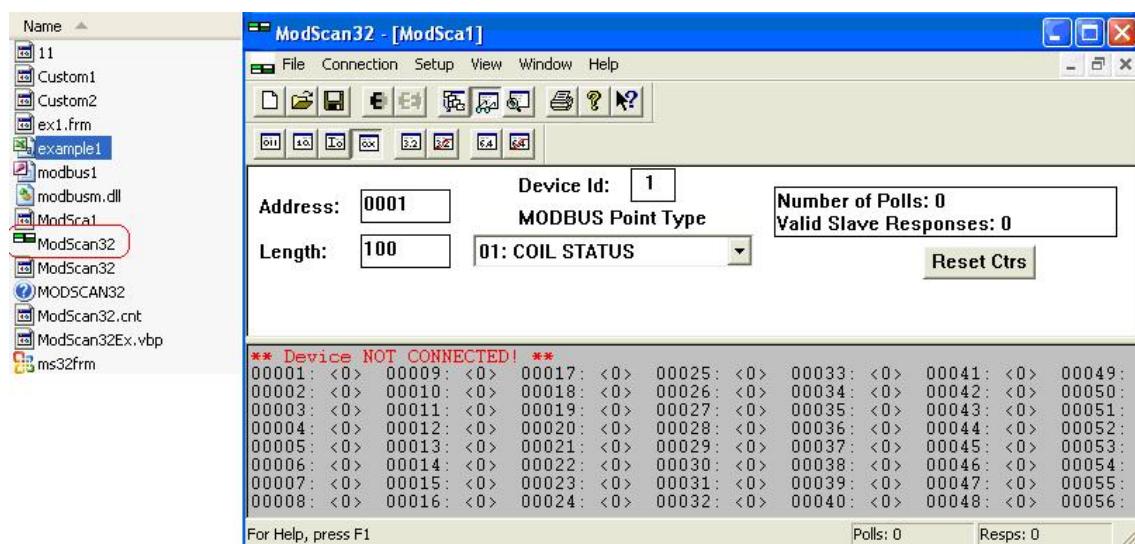


图 74 打开 Modscan32 软件

2、点击 Modscan32 上的 Connection->Connect，显示画面如下，选择计算机和 CP341 相连的 COM 口 COM1，设好波特率，数据位，停止位，校验方式，如下图：

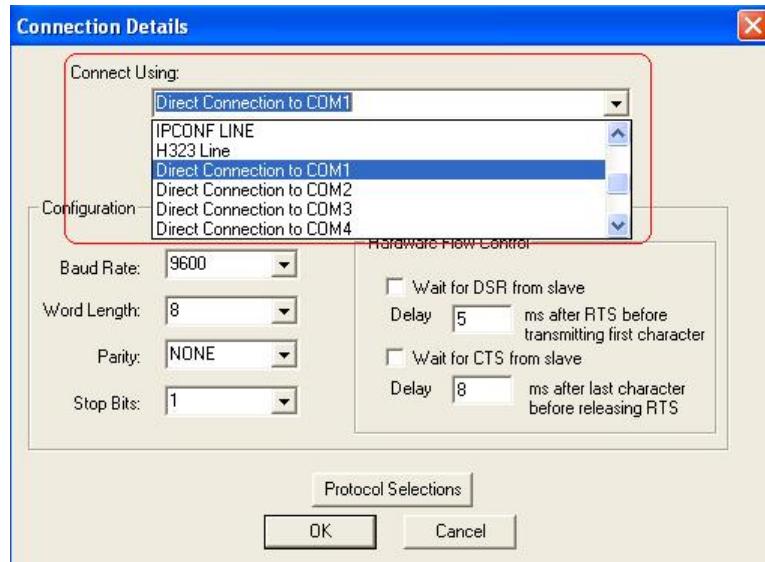


图 75 Modscan32 通信参数设置

3、设置完成，点击 OK，显示画面如下。

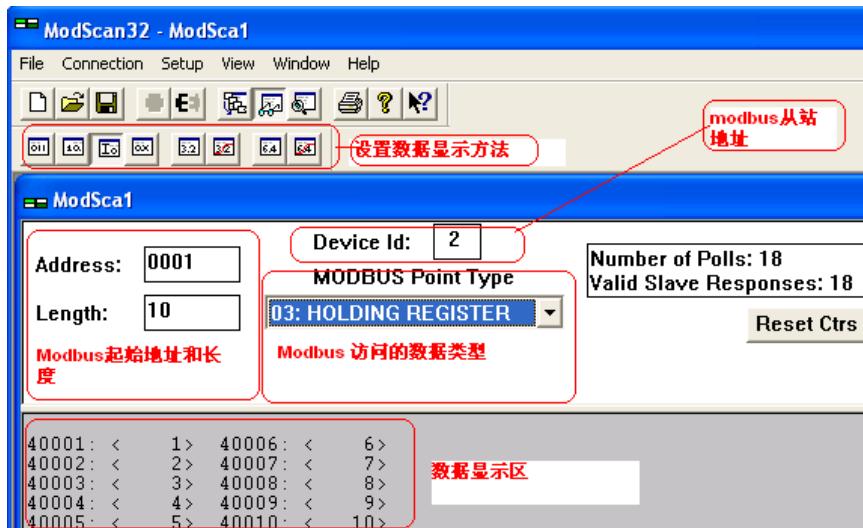


图 76 Modscan32 操作界面

4、数据显示区的 10 个地址的数据对应 CPU 中 DB1 的前 10 个字 (Word) 的数值 (DBW0–DBW10 分别为 1-10 )，如下图。

	Address	Symbol	Display format	Status value
1	DB1.DBW 0		DEC	1
2	DB1.DBW 2		DEC	2
3	DB1.DBW 4		DEC	3
4	DB1.DBW 6		DEC	4
5	DB1.DBW 8		DEC	5
6	DB1.DBW 10		DEC	6
7	DB1.DBW 12		DEC	7
8	DB1.DBW 14		DEC	8
9	DB1.DBW 16		DEC	9
10	DB1.DBW 18		DEC	10

图 77 DB1 数据监控

5、其它数据的读取，只需更改访问的数据类型，此处不再介绍。

6、但要说明一下对于浮点数的显示，PLC 中浮点数的高 16 位和低 16 位与 Modscan32 显示的相反，即 PLC 中浮点数的高 16 位对应 Modscan32 显示浮点数的低 16 位，PLC 中浮点数的低 16 位对应 Modscan32 显示浮点数的高 16 位，可在程序中做一下处理，如下图：

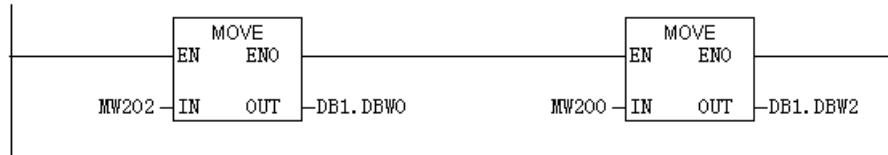


图 78 浮点数数据处理

此时 Modscan32 显示的浮点数 40001 对应 MD200 的浮点数值，但 MODBUS 地址对应 DB1.DBW0。

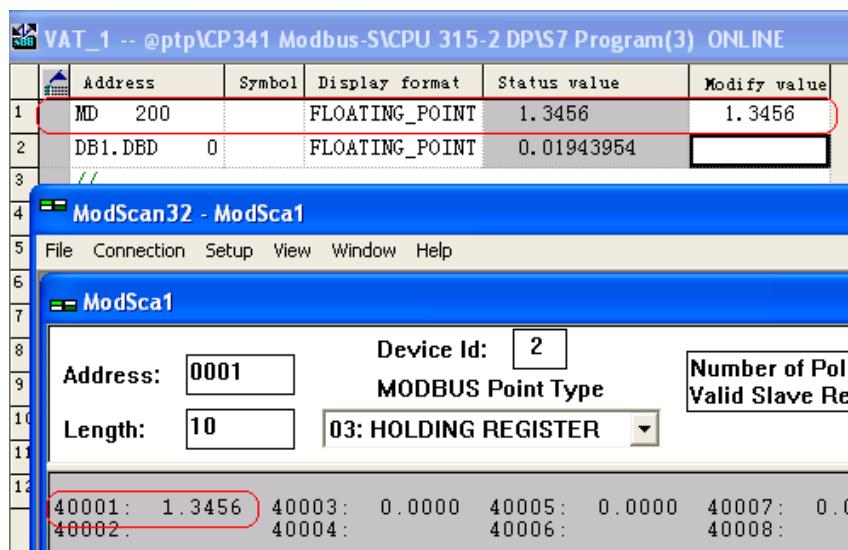


图 79 浮点数数据交换

如果使用的 CP341 是 RS422/485 接口的模板，使用计算机串口调试程序时要使用 RS232 转 RS422/485 转换器。

## 5.2 CP341 的 MODBUS Master 协议通信

### 5.2.1 硬件列表

PS 307	6ES7 307-1EA00-0AA0
CPU 315-2DP	6ES7 315-2AG10-0AB0
MMC	6ES7 953-8LG11-0AA0
CP341	6ES7 341-1AH01-0AE0
Dongle	6ES7 870-1AA01-0YA0

将 Dongle 插入 CP341 中，Dongle 和插入 Dongle 前后的 CP341 如下图所示。

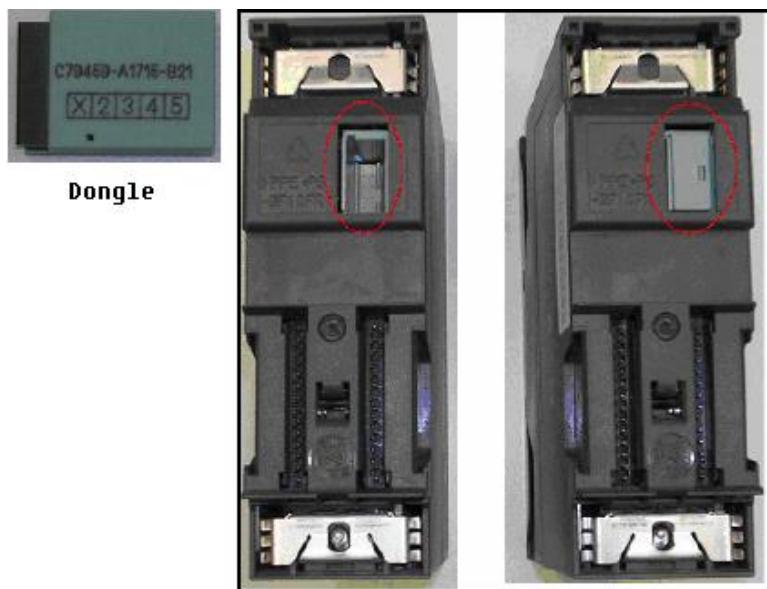


图 80 Dongle 插入前后对比

### 5.2.2 组态和配置

1、打开上面创建的项目 ptp，用鼠标右键点击项目名称，选择 Insert New Object->SIMATIC 300 Station，更改站的名称为 CP341 Modbus-M。

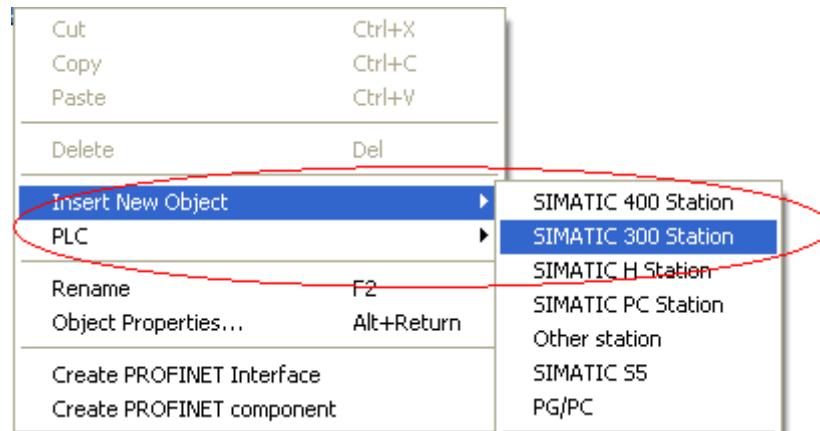


图 81 插入 S7-300 站

2、双击 Hardware 进入硬件配置画面，插入 RACK、CPU315-2DP、CP341。

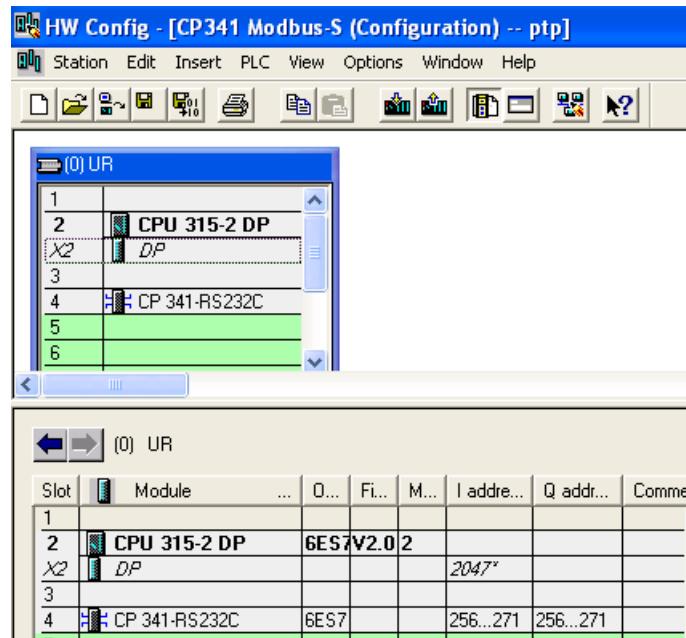


图 82 组态硬件

3、双击 CP341 模板，点击 Parameter...，配置 CP341 参数。在 Protocol 选项中选择 MODBUS Master。

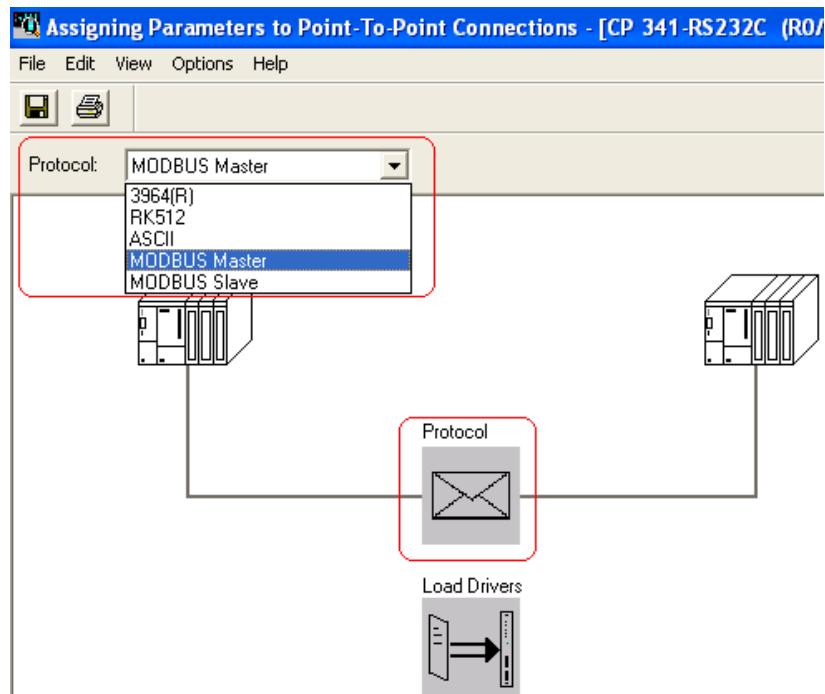


图 83 选择通信协议

4、双击 Protocol 下的信封图标，配置 Modbus Master 参数，点击 Modbus-Master 按钮，设置通讯波特率等参数。

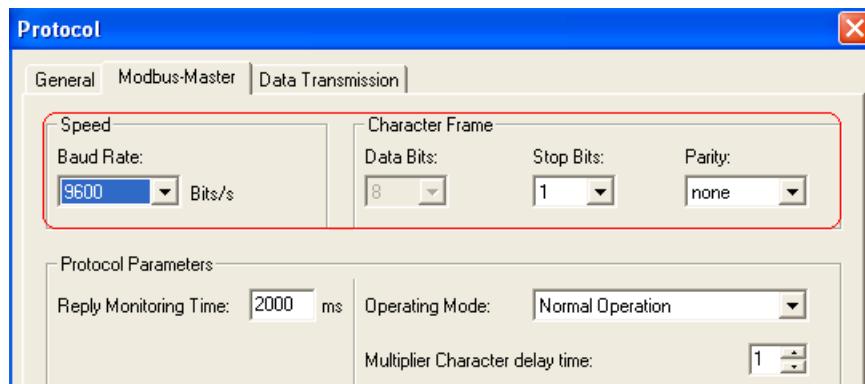


图 84 组态 MODBUS Master 通信参数

5、配置完成后保存时会提示是否装载驱动。

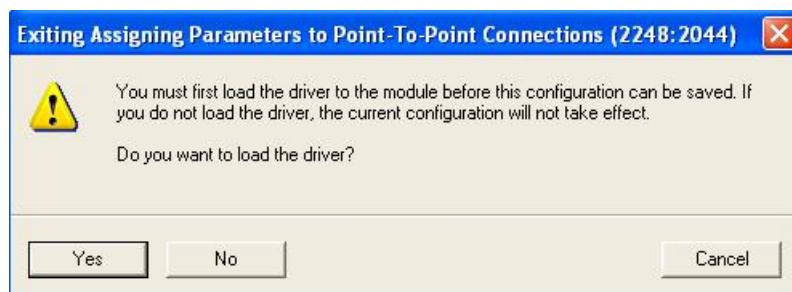


图 85 提示装载驱动对话框

6、此时必须连接到实际的 PLC，点击 Yes 按钮装载驱动，或点击 Load Drivers 装载，装载时 CPU 必须为 STOP 模式。

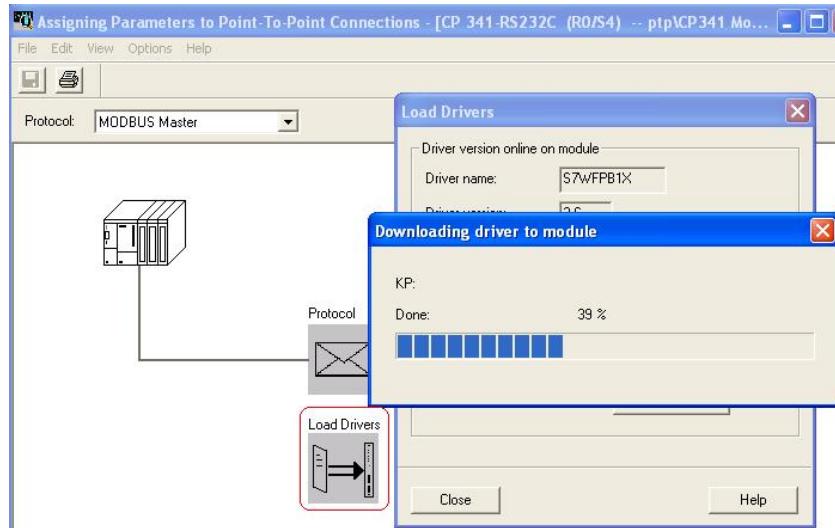


图 86 装载驱动程序

7、装载完成后，如果再次装载，Step7 会提示 Driver already exists。配置完成后点击

按钮保存编译硬件组态，并确认没有错误。

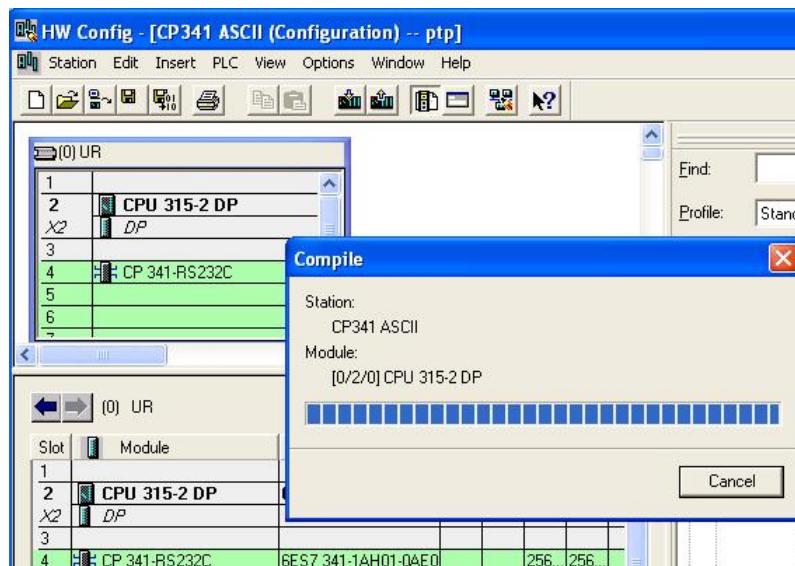


图 87 编译保存硬件组态

### 5.2.3 编写通信程序

1、双击 OB1，打开 OB1 编程画面，从库 Libraries->CP PtP->CP341 中调用发送程序块 FB8 P\_SND\_RK，分配背景数据块 DB8，将参数 LADDR 设为硬件组态中的输入起始逻辑地址 256。

Slot	Module	Order n...	Fi...	M...	I addre...	Q address	Com...
1							
2	CPU 315-2 DP	6ES7 315V2.0	2				
X2	DP				2047*		
3							
4	CP 341-RS232C	6ES7 341-1		256...271	256...271		
5							

图 88 CP341 逻辑地址

2、创建发送数据块 DB1，从站地址为 2，功能码为 3，起始地址为 0，读 4 个寄存器数据，具体功能码 3 的使用参见 Modbus Master 手册 5.3 Function Code 03 - Read Output Registers 部分的说明。

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	slave_address	BYTE	B#16#2	
+1.0	function_code	BYTE	B#16#3	
+2.0	reg_startadr	WORD	W#16#0	
+4.0	reg_num	WORD	W#16#4	
=6.0	END_STRUCT			

图 89 发送数据块 DB1

### 3、调用发送功能块

FB8 P\_SND\_RK 的参数设置见下表：

SF	'S' 为发送，此处必须为大写的'S'
LADDR	硬件组态中的起始逻辑地址，本例中为 256
REQ	发送数据触发位，上升沿触发，本例中为 M0.0
DB_NO	发送数据块号，本例中为 1 (DB1)
DBB_NO	发送数据的起始地址，本例中为 0 (DB1.DBB0)
LEN	发送数据的长度，本例中为 6
R_TYP	'X' 为扩展的数据块，此处必须为大写的'X'
R	取消通讯，本例中不用
DONE	发送完成位，发送完成后为 TRUE，同时没有错误
ERROR	错误位，为 TRUE 说明有错误
STATUS	状态字，标识错误代码，查看 Modbus Master 手册第 7 章获得相应的说明
其它参数	本例中不用

表 13 FB8 P\_SND\_RK 的参数定义

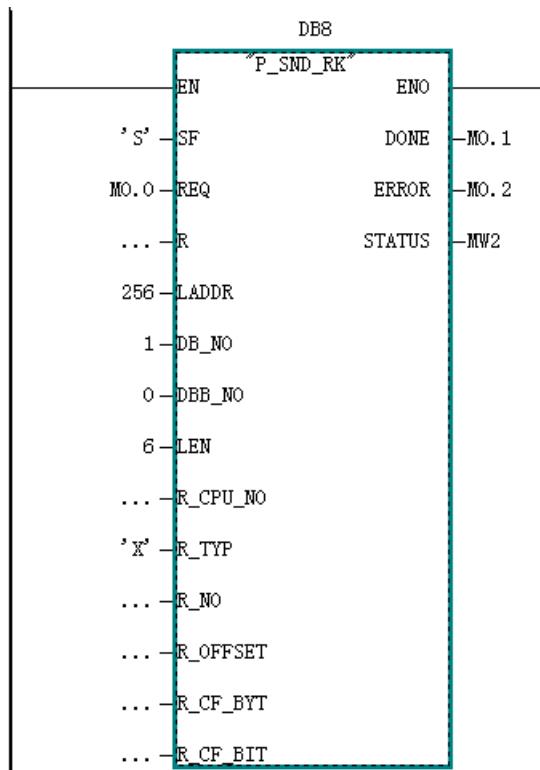


图 90 在 OB1 中调用 FB8 P\_SND\_RK

4、从库中 Libraries->CP PtP->CP341 调用接收程序块 FB7 P\_RCV\_RK，分配背景数据块 DB7，将 LADDR 设为硬件组态中的输入起始逻辑地址 256。

Slot	Module	Order n...	Fi...	M...	I addre...	Q address	Com
1							
2	CPU 315-2 DP	6ES7 315V2.0	2				
X2	DP				2047*		
3							
4	CP 341-RS232C	6ES7 341-1		256...271	256...271		

图 91 CP341 逻辑地址

5、创建接收数据块 DB2。

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	Receive_Data	ARRAY[1..50]		Temporary
*1.0		BYTE		
=50.0		END_STRUCT		

图 92 接收数据块 DB2

6、调用接收功能块

FB7 P\_RCV\_RK 的参数设置见下表：

LADDR	硬件组态中的起始逻辑地址, 本例中为 256
DB_NO	发送数据块号, 本例中为 2 (DB2)
DBB_NO	发送数据的起始地址, 本例中为 0 (DB2.DBB0)
LEN	接收数据的长度, 本例中为 MW4, 只有在接收到数据的当前周期, 此值不为 0, 可以查看 MW8 的值来确认接收到数据的长度
EN_R	使能接收位, 本例中为 M0.3
R	取消通讯, 本例中不用
NDR	接收完成位, 接收完成后为 TRUE, 同时没有错误
ERROR	错误位, 为 TRUE 说明有错误
STATUS	状态字, 标识错误代码, 查看 Modbus Master 手册第 7 章获得相应的说明
其它参数	本例中不用

表 14 FB7 P\_RCV\_RK 的参数定义

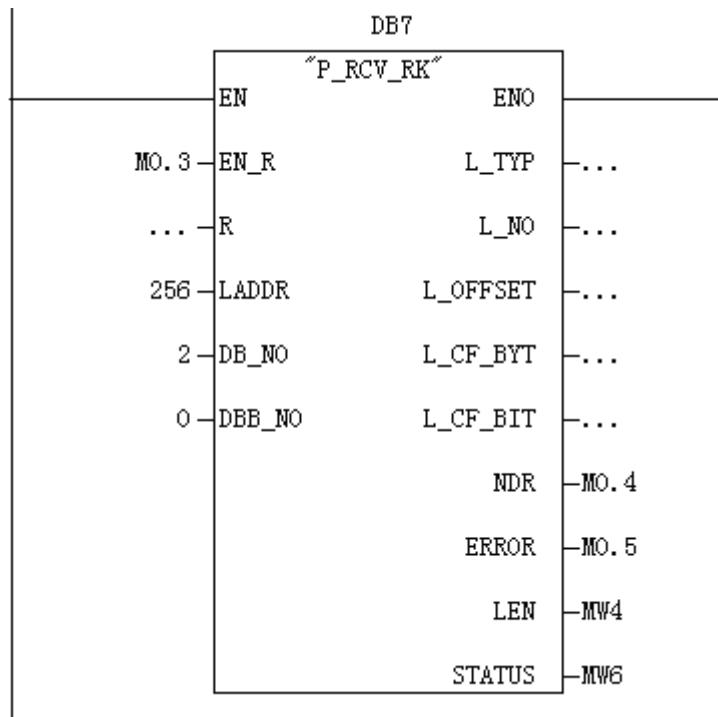


图 93 在 OB1 中调用 FB7 P\_RCV\_RK

#### 5.2.4 设备连接

使用标准的 RS232C 电缆连接 CP341 和计算机的串口, 具体电缆的连接方法请查看 CP341 手册章节 B Connecting Cables,

### 5.2.5 通信测试

1、安装 Modbus Slave 测试软件，安装完成后，在桌面上生成一图标，双击此图标，打开 modbus slave 测试软件，如图，

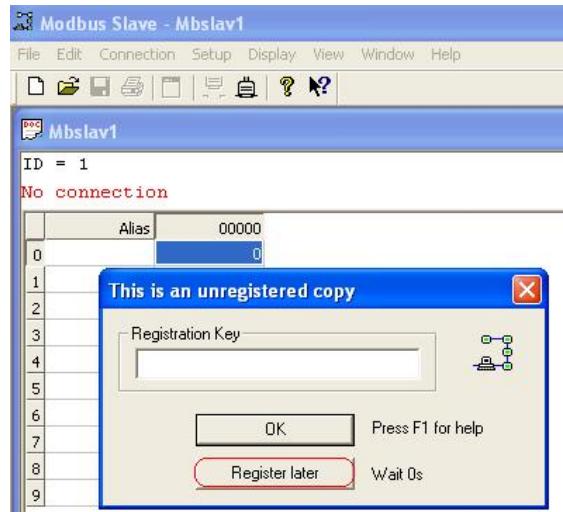


图 94 打开 modbus slave 测试软件

2、点击 Register later 按钮，不注册此软件，可以使用 30 天，但每次只能使用 15 分钟，网上下载的软件，没有注册码。然后点击菜单 Connection->connect 配置串口参数，要和 CP341 的参数相同。

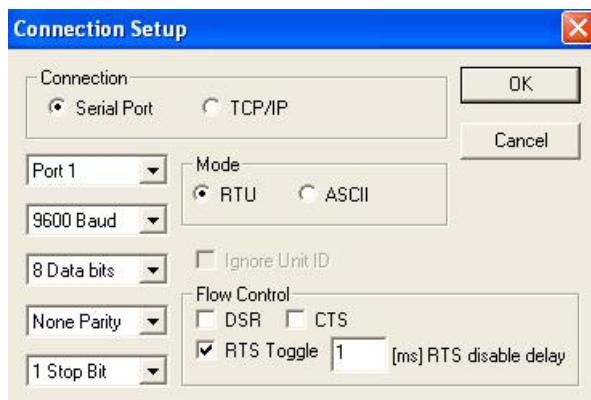


图 95 设置通信参数

3、点击 OK 按钮，然后选择菜单 Setup->Slave Definition... 设定通讯数据，因为 CP341 使用的是功能码 3，所以选择 03 Holding Register，CP341 访问的从站地址为 2，Slave ID 设为 2，起始地址从 1 开始，画面如下：

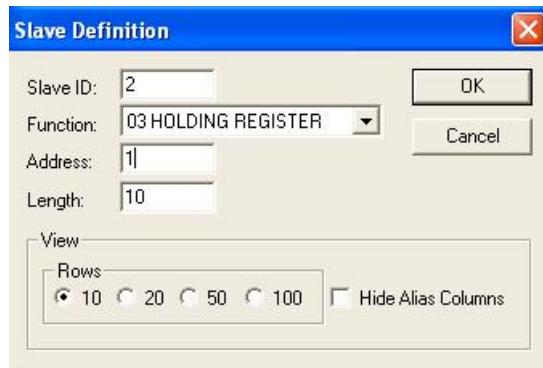


图 96 设置通信参数

4、然后设置地址 1 到 4 的数据分别为 111 到 444，然后在 Step7 变量表中设置 M0.3 为 TRUE，使能接收，然后将 M0.0 置成 TRUE（上升沿触发，想再发送需要将 M0.0 从 TRUE 置成 FALSE，然后再置成 TRUE），可以监控到 DB1.DBW0 到 DB1.DBW4 的值为 111 到 444，如下图。

	Address	Symbol	Display format	Status value
1	M 0.0		BOOL	true
2	M 0.3		BOOL	true
3	MW 8		DEC	8
4	//			
5	DB2.DBW 0		DEC	111
6	DB2.DBW 2		DEC	222
7	DB2.DBW 4		DEC	333
8	DB2.DBW 6		DEC	444

图 97 接收数据监视

5、如果想读取浮点数，必须设置数据格式，点击菜单 Display->Float Inverse。

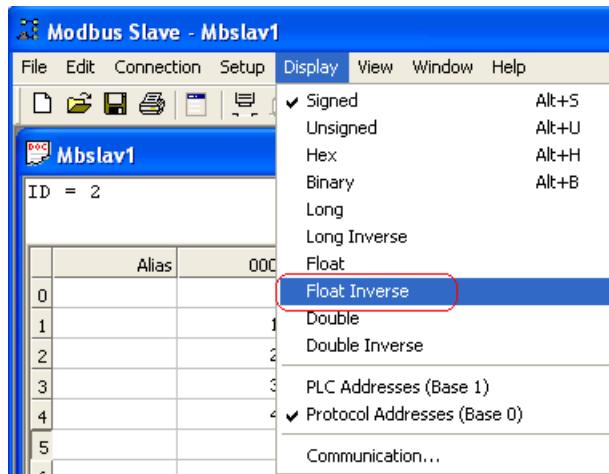


图 98 浮点数数据格式设置

6、设置地址 1 和 3 的值分别为 1.234 和 5.678，触发 M0.0 发送，可以看到接收到的数据为 1.234 和 5.678，如下图：

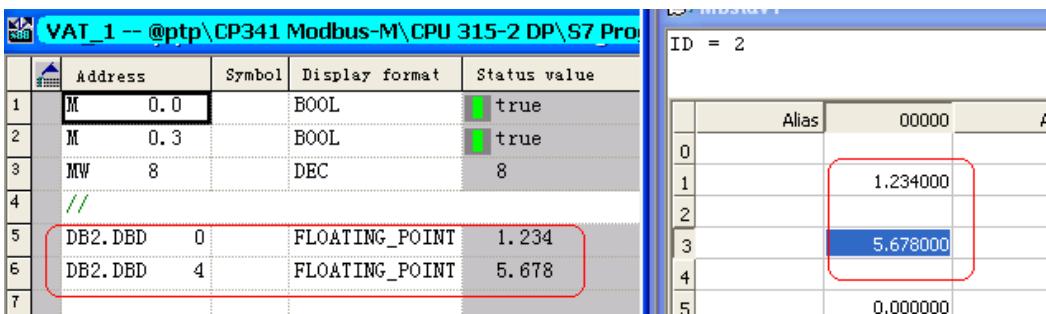


图 99 接收数据监视

### 5.3 CP441-2 的 MODBUS Master 协议通信

#### 5.3.1 硬件列表

RACK-400	6ES7400-1JA01-0AA0
PS407	6ES7407-0KA02-0AA0
CPU414-3	6ES7414-3XJ04-0AB0
CP441-2	6ES7441-2AA03-0AE0
Dongle	6ES7870-1AA01-0YA0

Dongle 的安装位置和 S7-300 CP341 一样，都在模块的后面，可参考前文的图片。

#### 5.3.2 组态和配置

1、打开创建好的项目 ptp，插入一个新的 SIMATIC 400 Station，更改站名为 CP441-2 modbus Master，双击 Hardware 进入硬件组态画面，插入 RACK、PS、CPU、CP441-2，具体步骤可参考上文信息。

2、CP441-2 模板，点击 Parameter 配置 CP441-2 参数，在 Protocol 中选择 MODBUS Master。

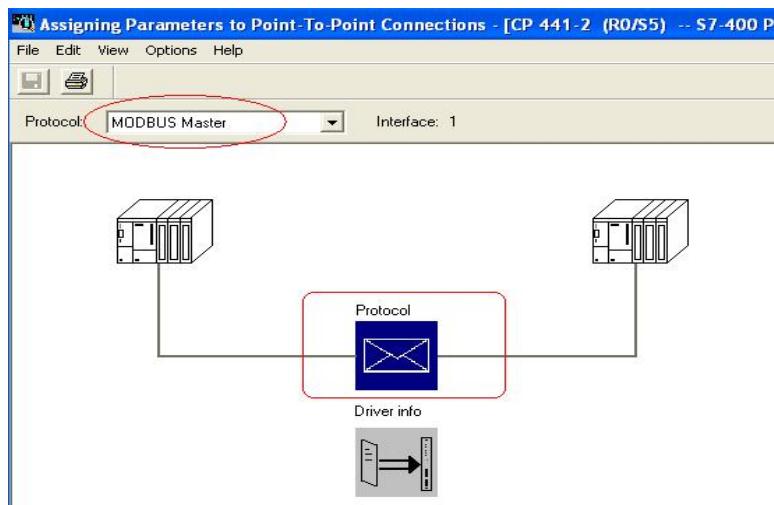


图 100 选择通信协议

3、Protocol 下的信封图标，配置 Modbus master 参数，点击 Modbus-master 按钮，设置通讯速率、报文信息等基本参数，其它可选择默认参数配置。

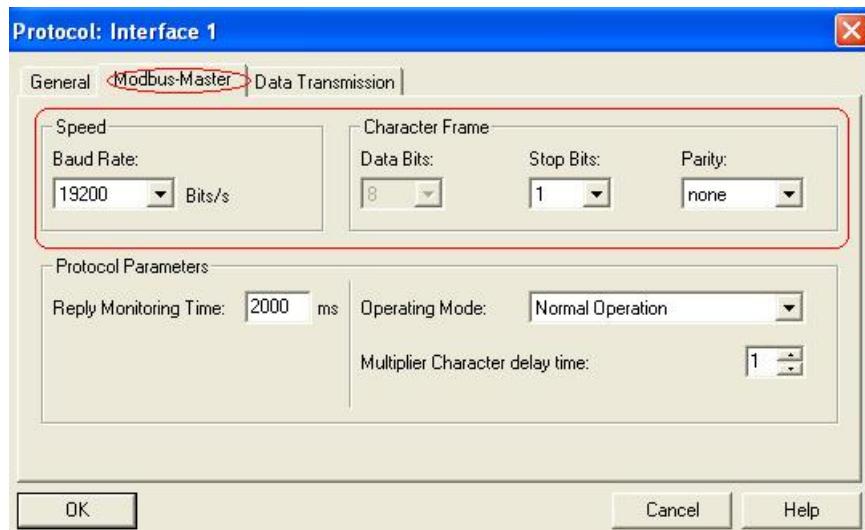


图 101 组态 MODBUS Master 通信参数

4、配置完成后点击 OK 并保存，关闭对话框，和用 CP441-2 做 ASCII，通讯时一样，要为所使用的接口新建 PtP，连接并记录 Local ID(Hex) 连接号。

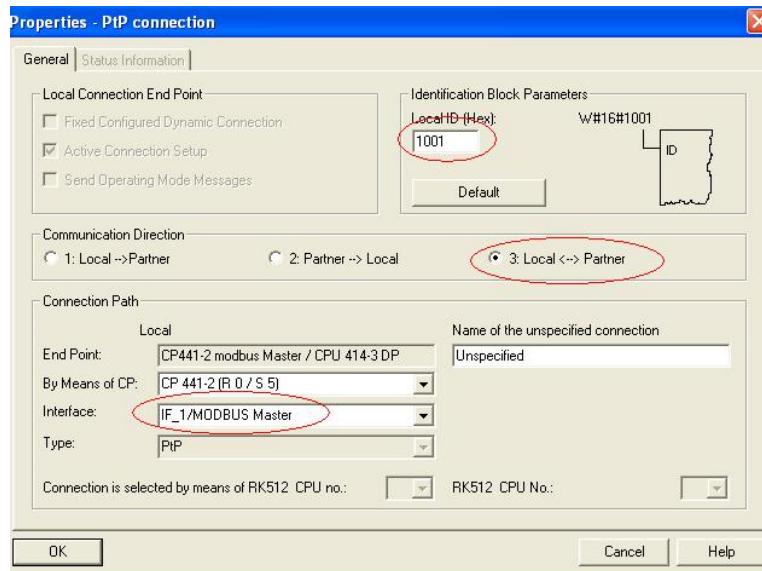


图 102 组态连接

5、 保存编译组态信息，如果没有错误，选中 CPU 进行组态信息的整体下载，并使

用 在线查看连接状态，直到“connection status”的显示图标为：，说明连接正确建立。

6、CP441 做 MODBUS 通讯时，还要更改 CPU 属性中的“Startup”参数：

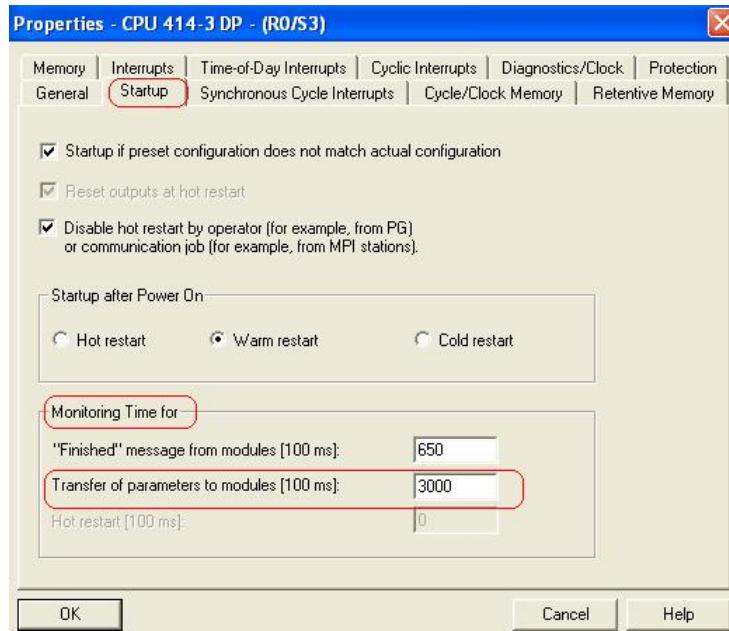


图 103 更改 CPU 属性中的“Startup”参数

“Monitoring Time for”→“Transfer of parameters to modules(100ms)” 设置为最小为 3000 (=300s) 的值，这样做是为了保证有足够长的时间将使用的加载协议驱动下载到 CP 卡中。

### 5.3.3 编写通信程序

1、打开 OB1，与做 ASCII 通讯时一样从库中调用 SFB12、SFB13，并分配背景数据块，SFB12 背景数据块为 DB12，发送数据的数据块为 DB1。

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	Slaveaddress	BYTE	B#16#5	Temporary p
+1.0	functionnumber	BYTE	B#16#1	
+2.0	bitstart	WORD	W#16#0	
+4.0	bitquan	WORD	W#16#10	
=6.0		END_STRUCT		

图 104 发送数据块 DB1

2、调用发送功能块。使用的从站地址为 5，功能码为 FC01，起始地址为 0，读 16 个输出的状态。当使用不同的功能码进行通讯时请求报文的定义请参考 MODBUS MASTER 手册第五章节 Function code 的相关说明。

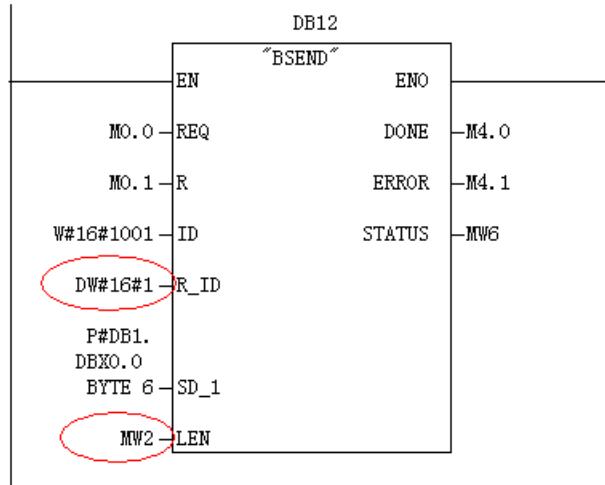


图 105 在 OB1 中调用 SFB12

3、Modbus RTU 通讯中使用 SFB12 的补充说明如下，其余参数与 CP441 ASCII 通讯时含义相同，可参考上文设置。

名称	注释
R_ID	用于在同一连接中不同块的区分，使用 Modbus RTU 通讯时此值的取值范围是 0-255，本例使用的是 DW#16#1。
LEN	取值范围取决于所使用的功能码，可参考下表。

功能码	长度（以字节为单位）
01	6
02	6
03	6
04	6
05	6
06	6
07	2
08	6
11	2
12	2
15	》 6
16	》 6

4、FB13 背景数据块为 DB13，并创建接收数据的 DB 块 DB2。

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	rcv	ARRAY[0..50]		Temporary
*1.0		BYTE		
=52.0		END_STRUCT		

图 106 接收数据块 DB2

5、调用接收功能块

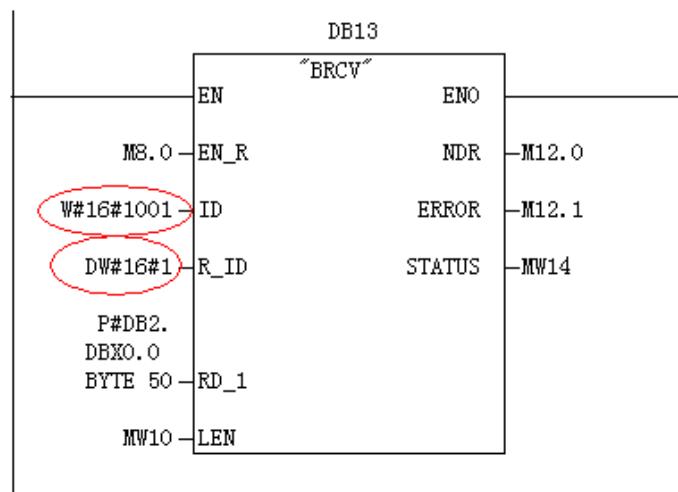


图 107 在 OB1 中调用 SFB13

SFB13 中“R\_ID”的设置必须与成对调用的 SFB12 中的“R\_ID”一致，用来激活接收信息的程序，所以此处必须设置为 DW#16#1。可在 MW20 中查看接收数据的长度信息。

### 5.3.4 设备连接

使用标准的 RS232C 电缆连接 CP441-2 interface1 和计算机的串口，具体电缆的连接方法请查看 CP441 手册附录 B Connecting cables

### 5.3.5 通信测试

1、同样使用 Modbus slave 测试软件，连接设置请参考上文 CP341 测试的操作步骤，从站的设置如下图所示。

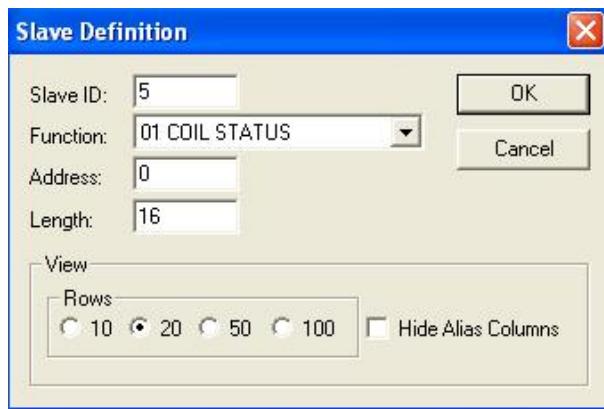


图 108 Modbus slave 测试软件设置

2、在 Step7 变量表中设置 M8.0 为 TRUE，使能接收，手动触发 M0.0 以产生上升沿，使能发送，可看到数据通讯结果如下图所示。

	Address	Symbol	Disp	Status value	Modify value
1	//send				
2	M 0.0	BOOL	true		
3	MW 2	DEC	6	6	
4	M 4.0	BOOL	false	true	
5	M 4.1	BOOL	false		
6	MW 6	DEC	0		
7					
8	//receive				
9	M 8.0	BOOL	true	true	
10					
11	DB2.DBB 0	BIN	2#1000_0000		
12	DB2.DBB 1	BIN	2#1111_1111		
13					
14	MW 20	DEC	2		
15	M 12.0	BOOL	false		
16	M 12.1	BOOL	false		
17	MW 14	DEC	0		

00000

Alias	00000
0	1
1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	0
9	0
10	0
11	0
12	0
13	0
14	0
15	1

对应DB2.DBB1

对应DB2.DBB0

图 109 在线监视通信状态

3、测试不同的功能码时相应的修改主站发送的数据请求报文即可，

Address	Name	Type	Initial value	Actual value
0.0	Slaveaddress	BYTE	B#16#5	B#16#05
1.0	functionnumber	BYTE	B#16#3	B#16#03
2.0	bitstart	WORD	W#16#0	W#16#0000
4.0	bitquan	WORD	W#16#3	W#16#0003

图 110 创建发送数据帧

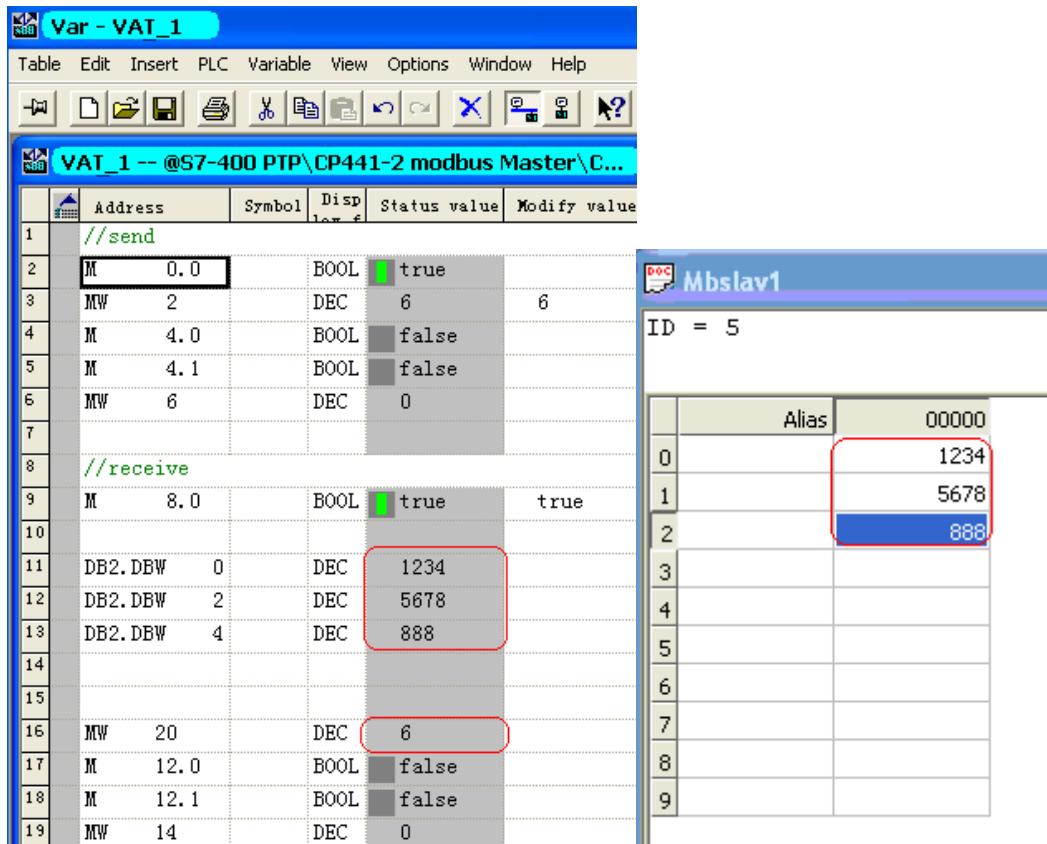


图 111 在 OB1 中调用 SFB13

## 5.4 CP441-2 的 MODBUS Slave 协议通信

### 5.4.1 硬件列表

RACK-400	6ES7400-1JA01-0AA0
PS407	6ES7407-0KA02-0AA0
CPU414-3	6ES7414-3XJ04-0AB0
CP441-2	6ES7441-2AA03-0AE0
Dongle	6ES7870-1AB01-0YA0

#### 5.4.2 组态和配置

- 1、打开创建好的项目 ptp，插入一个新的 SIMATIC 400 Station，更改站名为 CP441-2 modbus Slave
- 2、双击 Hardware 进入硬件组态画面，插入 RACK、PS、CPU、CP441-2，具体步骤可参考上文信息。
- 3、CP441-2 模板，点击 Parameter 配置 CP441-2 参数，在 Protocol 中选择 MODBUS Slave。

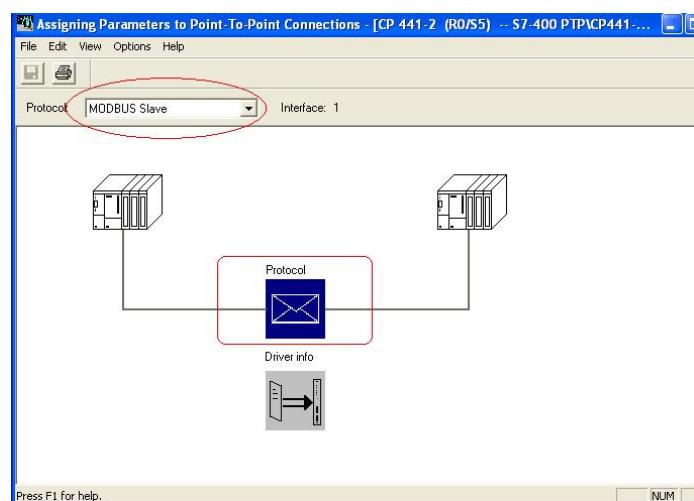


图 112 选择通信协议

- 4、击 Protocol 下的信封图标，配置 Modbus slave 参数：

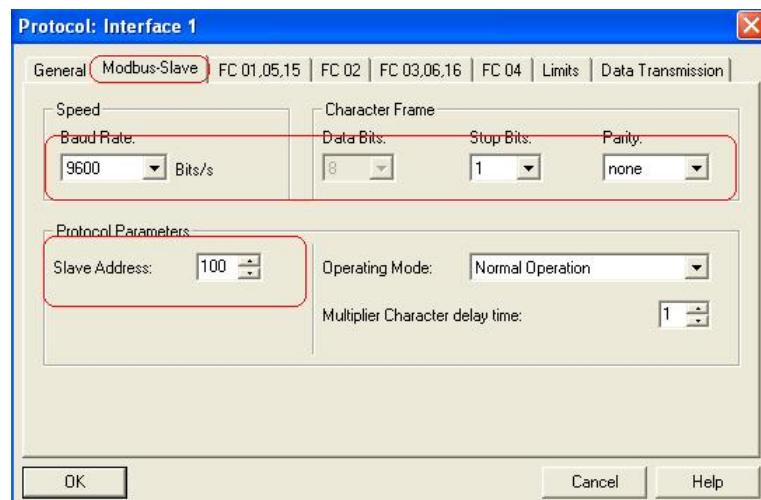


图 113 组态 MODBUS Slave 通信参数

5、设置通讯速率、报文信息等基本参数，本例中从站地址设为 100（默认为 222，取值范围是 1~255）。然后进入从站功能码配置界面，具体信息如下。

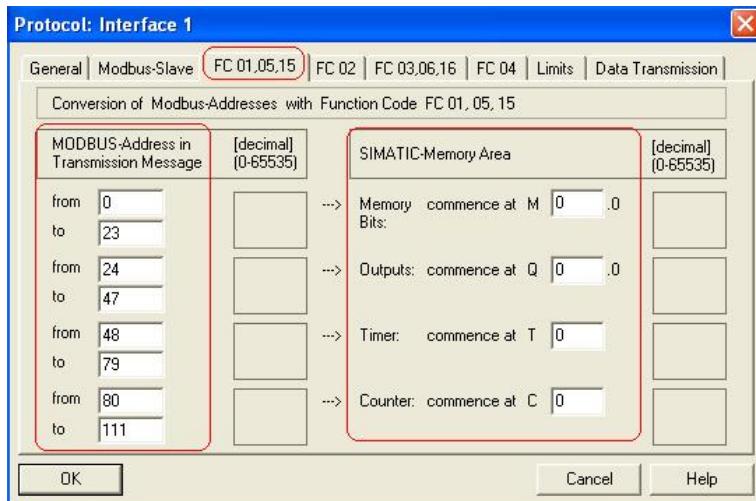


图 114 MODBUS Slave 地址分配

6、FC01, 05, 15：读取、强制输出位的状态；左边的地址为信息传送地址，右边对应西门子的 PLC 地址区，即左边地址从 0~23 对应 Modbus 地址区为 00001~00024，对应西门子数据区为 M0.0~M2.7；24~47 对应 Modbus 地址区为 00025~00048，对应西门子数据区为 Q0.0~Q2.7；从地址 48~79, 80~111 对应 Modbus 地址区为 00049~00080, 00081~00112，对应西门子数据区为 Timer, Counter。FC02 读取输入数据位的状态，地址对应如上文所述。

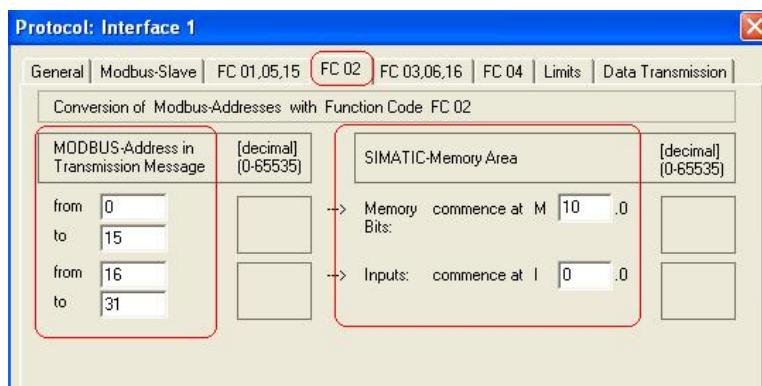


图 115 MODBUS Slave 地址分配

7、FC03,06,16 组态输出寄存器数据区，对应西门子数据区为 DB 块。

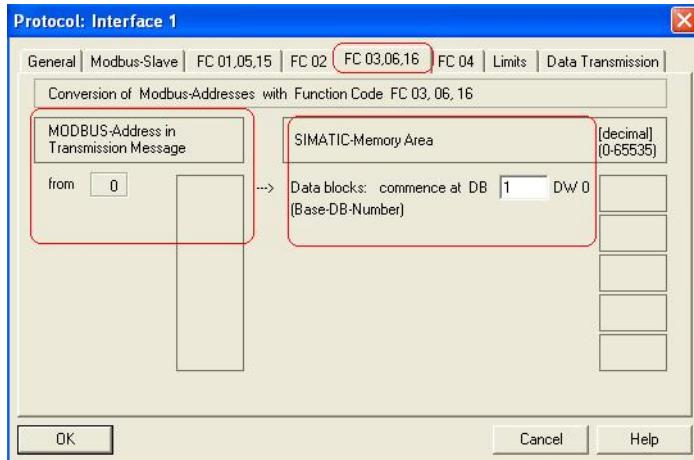


图 116 MODBUS Slave 地址分配

8、FC04 组态输入寄存器数据区，对应西门子数据区同样是 DB 块

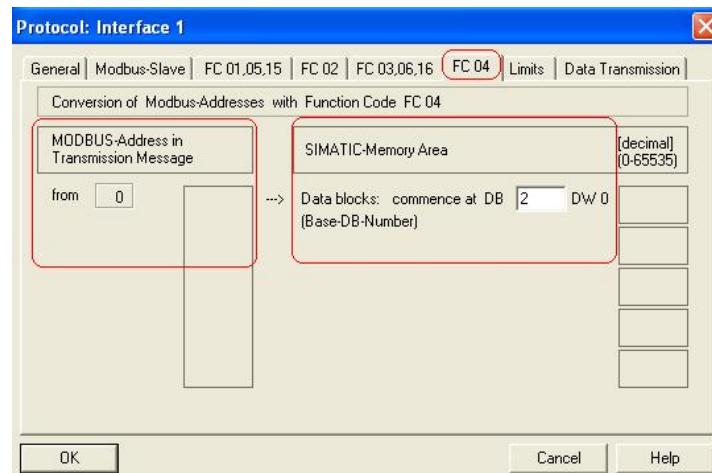


图 117 MODBUS Slave 地址分配

9、设置 FC05, 06, 15, 16 写参数的限制值:

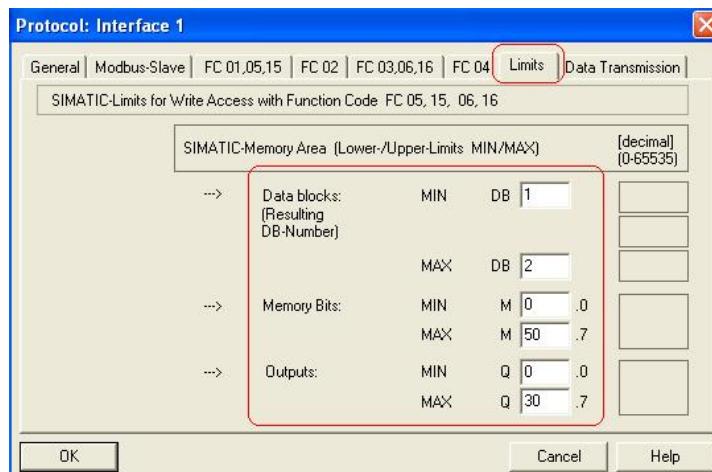


图 118 MODBUS Slave 地址分配

10、配置完成后点击 OK 并保存，关闭对话框，与上文使用 CP441-2 时，一样要为所使用的接口新建 PtP 连接并记录 Local ID(Hex) 连接号。

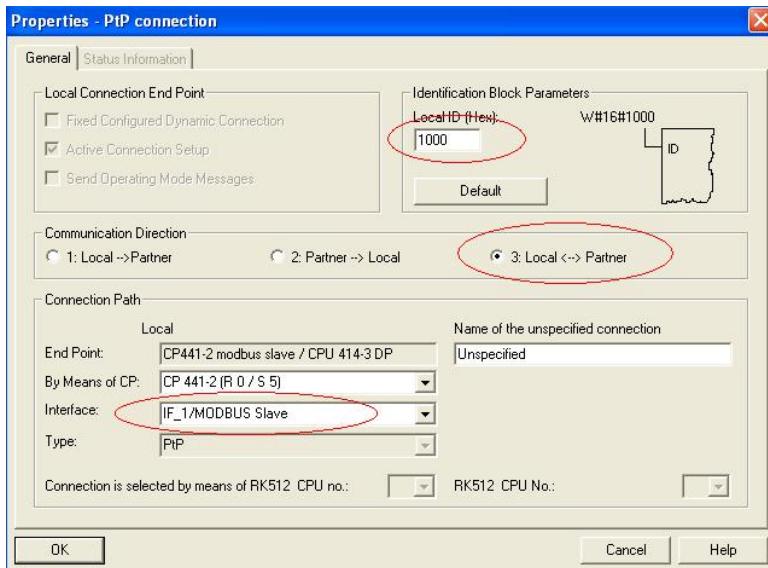


图 119 组态连接

11、点击 编译组态信息，若没有错误，选中 CPU 进行组态信息的整体下载，并使用 在线查看连接状态，直到“connection status”的显示图标为：，说明连接正确建立。与使用 CP441 做 MODBUS Master 通讯时一样，还要更改 CPU 属性中的“Startup”参数：

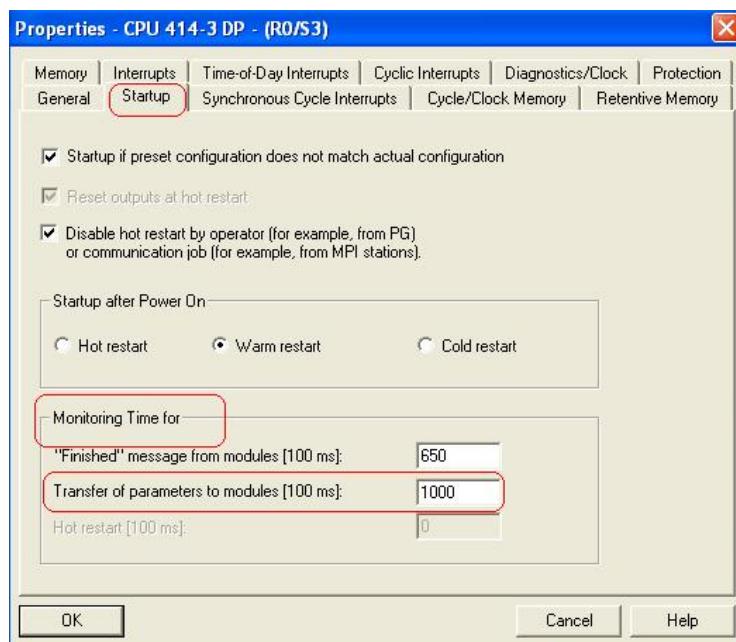


图 120 更改 CPU 属性中的“Startup”参数

将“Monitoring Time for”→“Transfer of parameters to modules(100ms)” 设置为最小为 1000 (=100s) 的值，这样做是为了保证有足够长的时间将使用的加载协议驱动在参数分配过程中下载到 CP 卡。

#### 5.4.3 编写通信程序

1、双击 OB1，打开编程画面，在 Libraries -> Modbus 中调用 FB180，并分配背景数据块 DB180，新建数据块 FC03 06 16，FC04 功能码通讯数据的 DB1，DB2，本例都为 50 个 WORD 的数组结构。

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	DB_VAR	ARRAY[0..50]		Temporary placeholder variable
*2.0		WORD		
=102.0		END_STRUCT		

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	DB_VAR	ARRAY[0..50]		Temporary placeholder variable
*2.0		WORD		
=102.0		END_STRUCT		

图 121 通信数据块 DB1、DB2

2、FB180 的参数设置参见下表：

名称	数据类型	注释
ID	INPUT/INT	Local ID 号，在 PtP 连接属性中获得，十六进制输入
START_TIMER	INPUT/TIMER	初始化超时定时器，本例中为 T5
START_TIME	INPUT/S5TIME	初始化超时时间值，本例为 5S
STATUS_TIMER	INPUT/TIMER	通讯状态读取定时器，本例为 T6
STATUS_TIME	INPUT/S5TIME	通讯状态读取时间值，本例为 2S
OB_MASK	INPUT/BOOL	IO 访问错误屏蔽位，本例使用 M20.0 0：不屏蔽 IO 访问错误 1：屏蔽 IO 访问错误，并使报警延时
CP_START	INPUT/BOOL	FB 初始化使能位，本例使用 M20.1
CP_START_FM	INPUT/BOOL	CP_START 初始化的上升沿位，本例 M20.2
CP_NDR	OUTPUT/BOOL	CP 写操作位，本例为 M20.3
CP_START_OK	OUTPUT/BOOL	初始化成功完成位，本例 M20.4
CP_START_ERR_OR	OUTPUT/BOOL	初始化错误标志，本例 M20.5
ERROR_NR	OUTPUT/WORD	错误号，本例为 MW22
ERROR_INFO	OUTPUT/WORD	错误信息，本例为 MW24，诊断信息可参考手册

表 15 FB180 参数定义

### 3、调用 FB180 功能块。

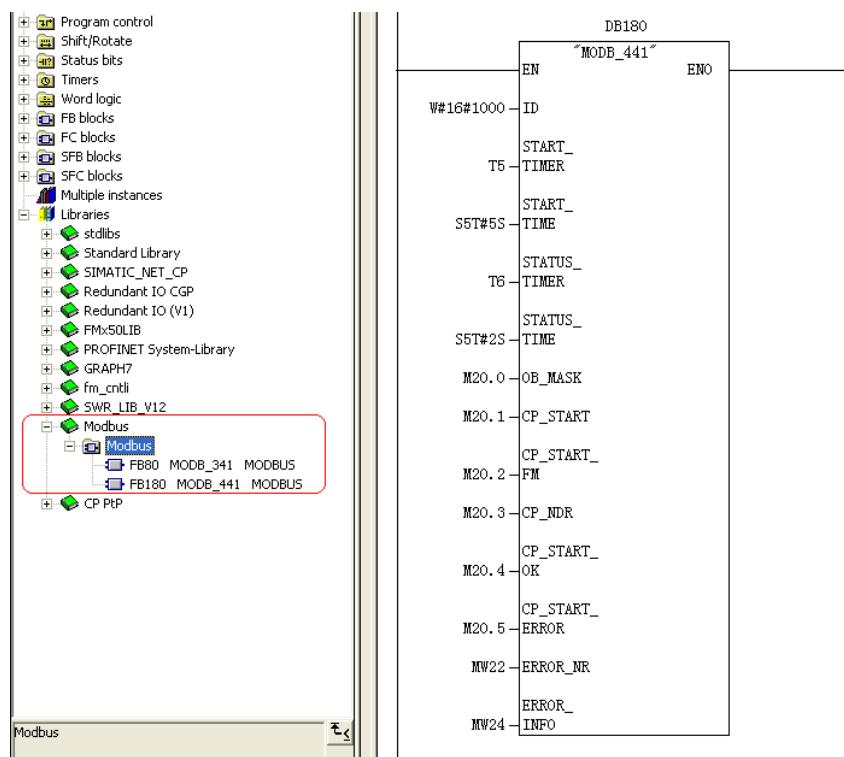


图 122 调用 FB180 功能块

#### 5.4.4 设备连接

使用标准的 RS232C 电缆连接 CP341 和计算机的串口，具体电缆的连接方法请查看 CP341 手册章节 B Connecting Cables，

#### 5.3.5 通信测试

硬件组态和程序下载到 CPU 中，在计算机上打开软件 Modscan32，点击 Modscan32 上的 Connection—> Connect，显示如下画面。选择所应用的串口，设置好波特率，数据位，停止位，校验方式。并可以根据所应用的连接来选择使用哪种通讯协议。设置完成后，点击 OK，显示监控画面。

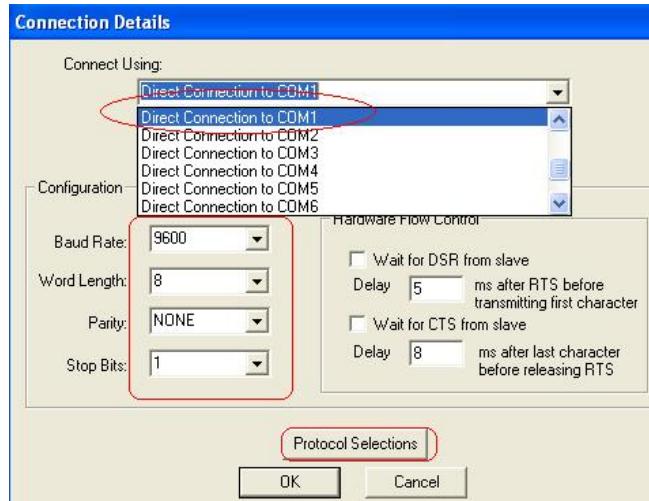


图 123 分配通信参数

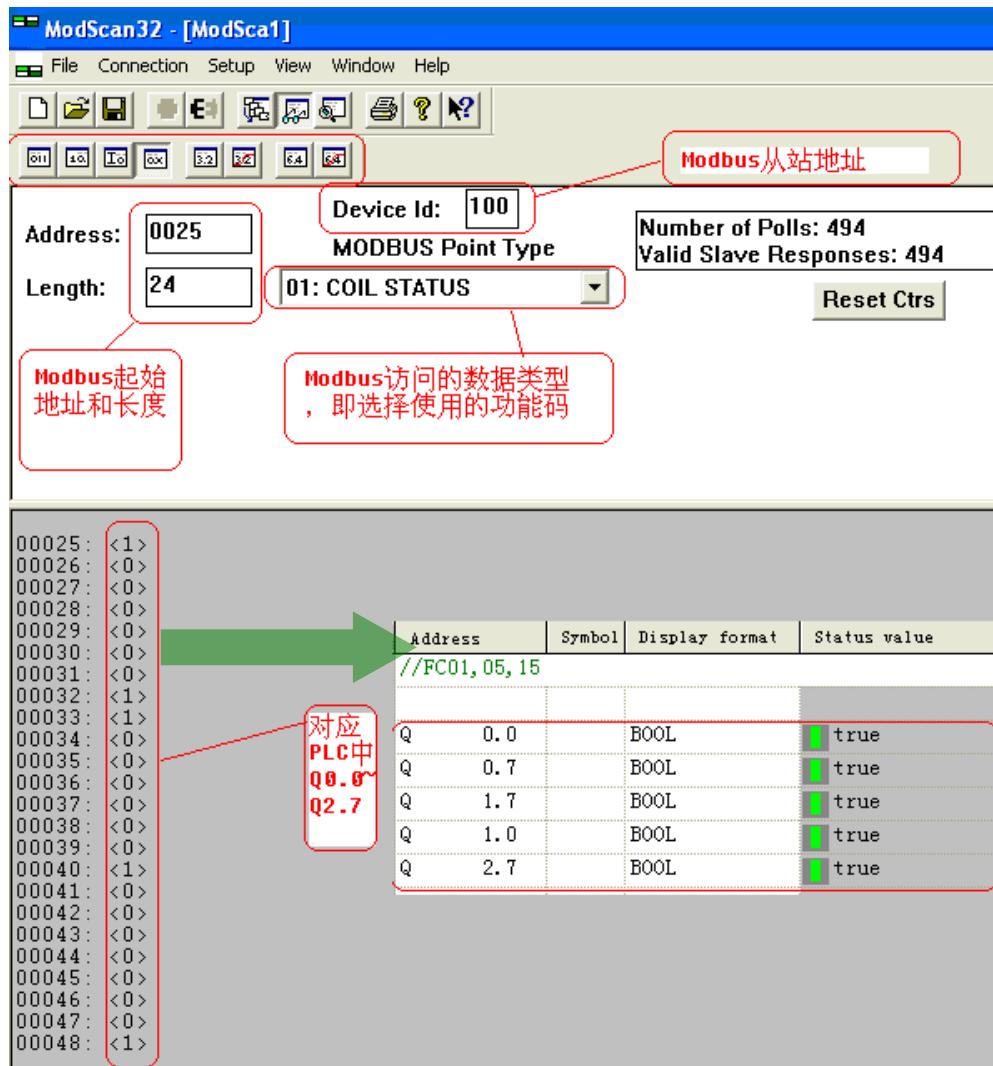


图 124 通信监控界面

下图所示为应用功能码 FC04 时对应的通讯监控画面。其他数据的读取，只需更改相应的访问数据的类型，对于浮点数的显示，也可参考上文 CP341 做 Modbus Slave 通讯时的读取方式。

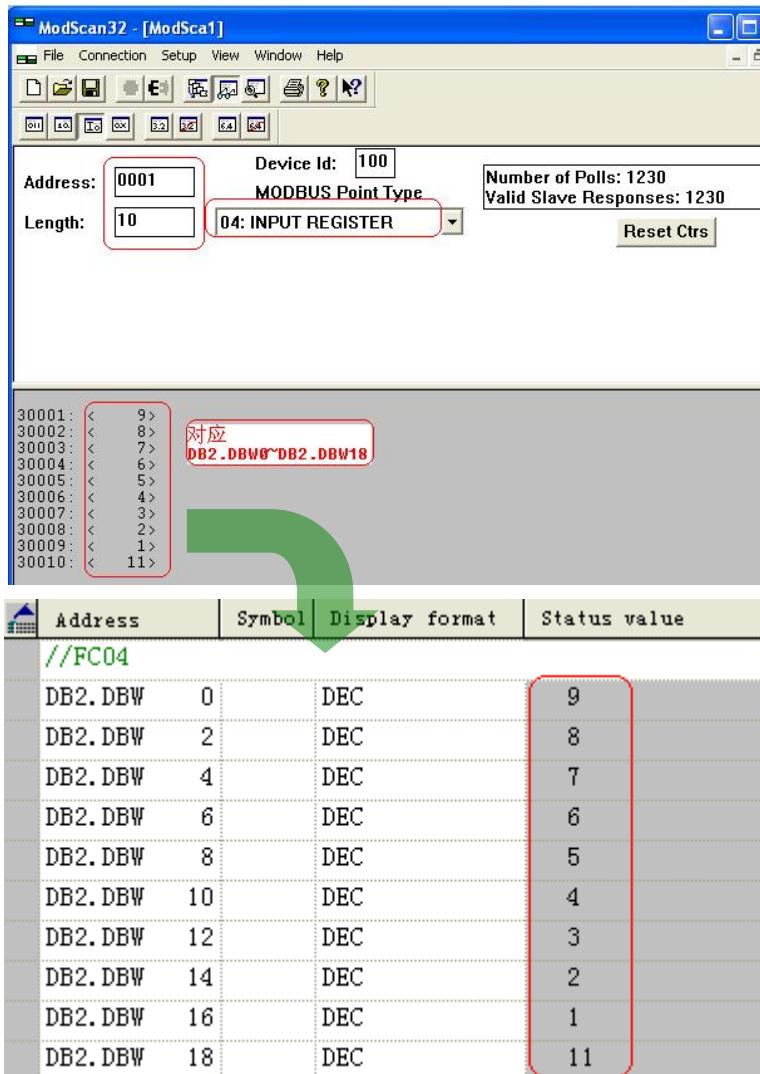


图 125 通信监控界面

如果您对该文档有任何建议，请将您的宝贵建议提交至[下载中心留言板](#)。

该文档的文档编号：**A0081**

## 附录一 推荐网址

### 自动化系统

西门子（中国）有限公司

工业自动化与驱动技术集团 客户服务与支持中心

网站首页：[www.4008104288.com.cn](http://www.4008104288.com.cn)

自动化系统 下载中心：

<http://www.ad.siemens.com.cn/download/DocList.aspx?Typeld=0&CatFirst=1>

自动化系统 全球技术资源：

<http://support.automation.siemens.com/CN/view/zh/10805045/130000>

“找答案”自动化系统版区：

<http://www.ad.siemens.com.cn/service/answer/category.asp?cid=1027>

### 驱动技术

西门子（中国）有限公司

工业自动化与驱动技术集团 客户服务与支持中心

网站首页：[www.4008104288.com.cn](http://www.4008104288.com.cn)

驱动技术 下载中心：

<http://www.ad.siemens.com.cn/download/DocList.aspx?Typeld=0&CatFirst=85>

驱动技术 全球技术资源：

<http://support.automation.siemens.com/CN/view/zh/10803928/130000>

“找答案”驱动技术版区：

<http://www.ad.siemens.com.cn/service/answer/category.asp?cid=1038>

**注意事项**

应用示例与所示电路、设备及任何可能结果没有必然联系，并不完全相关。应用示例不表示客户的具体解决方案。它们仅对典型应用提供支持。用户负责确保所述产品的正确使用。这些应用示例不能免除用户在确保安全、专业使用、安装、操作和维护设备方面的责任。当使用这些应用示例时，应意识到西门子不对在所述责任条款范围之外的任何损坏/索赔承担责任。我们保留随时修改这些应用示例的权利，恕不另行通知。如果这些应用示例与其它西门子出版物(例如，目录)给出的建议不同，则以其它文档的内容为准。

**声明**

我们已核对过本手册的内容与所描述的硬件和软件相符。由于差错难以完全避免，我们不能保证完全一致。我们会经常对手册中的数据进行检查，并在后续的版本中进行必要的更正。欢迎您提出宝贵意见。

版权© 西门子（中国）有限公司 2001-2010 版权保留

复制、传播或者使用该文件或文件内容必须经过权利人书面明确同意。侵权者将承担权利人的全部损失。权利人保留一切权利，包括复制、发行，以及改编、汇编的权利。

西门子（中国）有限公司