

常问问题 • 07/2014

WinCC V7.2 Modbus TCP 通讯

Simatic WinCC V7.2

目录

1 概述	3
2 WinCC 和 Quantum PLC 通讯	3
2.1 PLC 组态	3
2.2 WinCC 组态.....	5
3 WinCC 和 Premium PLC 通讯	14
3.1 PLC 组态	14
3.2 WinCC 组态.....	16
4 WinCC 和第三方 Modbus TCP/IP 设备通讯	18

1 概述

从版本 WinCC V7.0 开始，WinCC 支持 Modbus TCP 通讯，WinCC 中的 Modbus TCP 驱动主要是针对施耐德 PLC 开发的，支持的 PLC 类型如下：

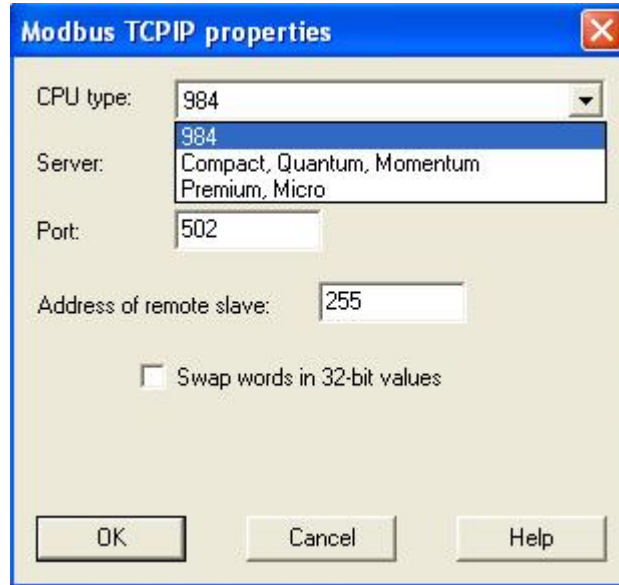


图 1

本文档以 Quantum CPU651 和 Premium P57 为例，介绍 WinCC V7.2 的 Modbus TCP 通讯的组态步骤。并在最后列出 WinCC 和第三方 Modbus TCP 设备通讯的注意事项。

2 WinCC 和 Quantum PLC 通讯

2.1 PLC 组态

2.1.1 硬件配置

在 PLC 的编程软件 Unity Pro 中按如下配置硬件：

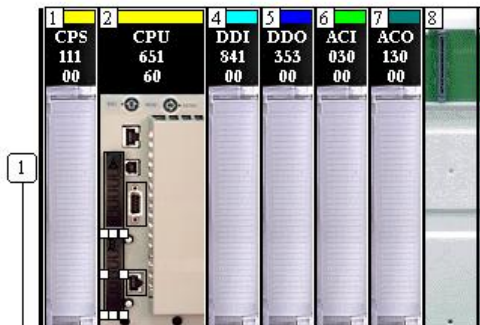


图 2

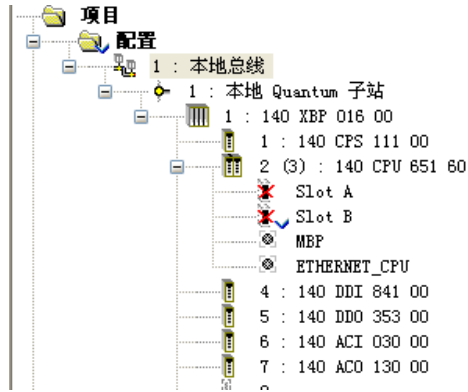


图 3

2.1.2 网络配置

在通讯→网络下创建新网络:



图 4

双击新建的网络，配置 IP 地址及网络类型:

图 5

在硬件配置中，为 CPU 上的以太网口分配网络:

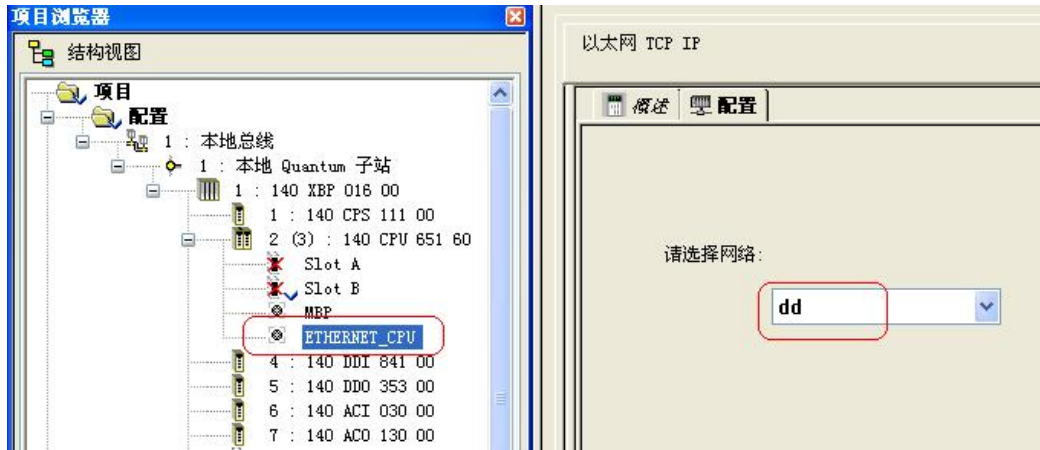


图 6

2.1.3 创建数据表

Quantum PLC 寄存器编址从 1 开始，与 Modbus 地址的对应关系为：%m 对应 0X, %i 对应 1X, %iw 对应 3x, %mw 对应 4x。

例如：%i1 对应 100001；%m 对应 000001；%iw1 对应 300001；%mw1 对应 400001。



图 7

2.2 WinCC 组态

2.2.1 组态连接

右键“变量管理”选择“打开”：

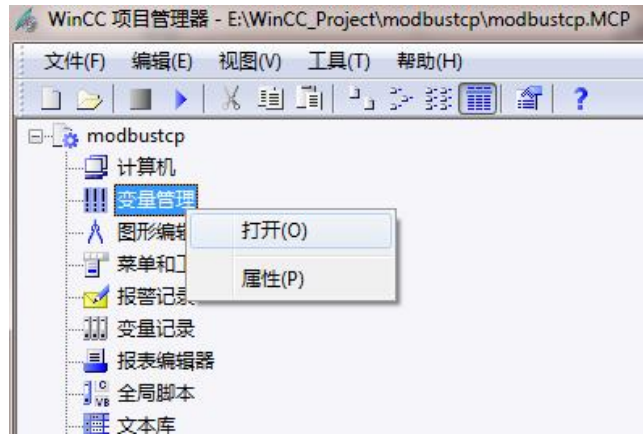


图 8



图 9

右键“ Modbus TCP/IP Unit #1” 选择“新建连接”：

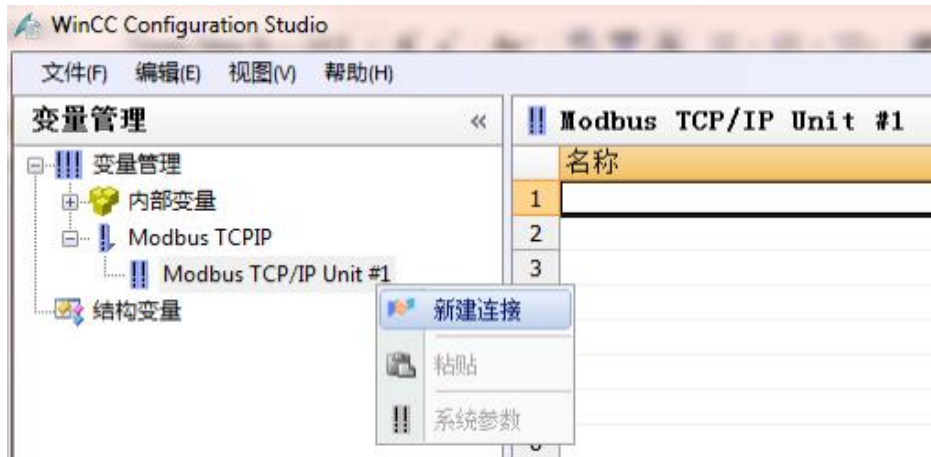


图 10

为新建的连接命名，并右键“连接参数”，设置连接属性：



图 11

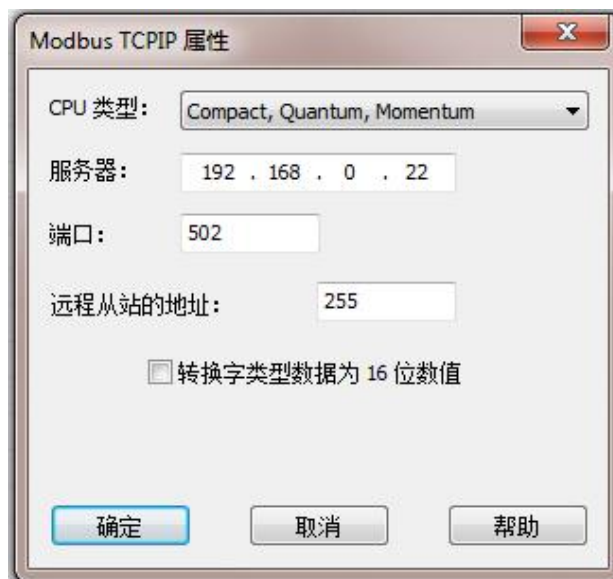


图 12

CPU 类型：不同 CPU 的字和位的关系、寄存器起始地址存在不同，所以选择不同的 CPU 类型，WinCC 变量地址设定及数据处理会有些不同。

这里选择“ Compact, Quantum, Momentum”。

服务器：PLC 作为 Modbus TCP 通讯的服务器，WinCC 作为客户机。这里输入 PLC 的以太网 IP 地址。

端口：Modbus TCP 通讯默认端口为 502

远程从站的地址：使用桥接器（例如 MB+到 Modbus TCPIP）时，此处输入远程控制器的从站地址。如果未使用桥接器，则必须输入默认值 255 或 0 作为地址。

转换数据类型为 16 位数值：此处翻译有点问题，应该为“交换 32 位值中的字” (Swap words in 32-bit values)。

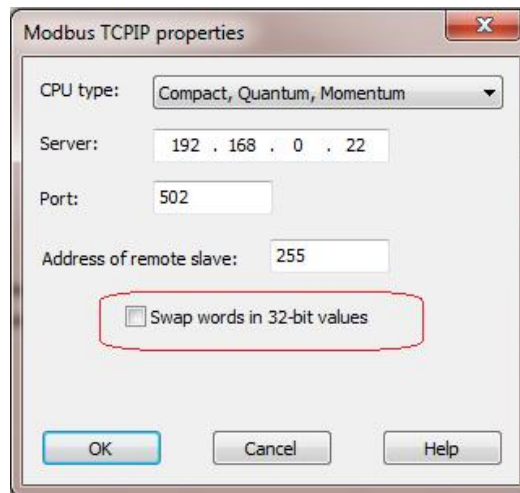


图 13

此选择只影响“有符号 32 位数”“无符号 32 位数”和“浮点数 32 位 IEEE 754”三种数据类型。

连接施耐德 PLC 时，此处不要选择。

2.2.2 创建变量

“ Modbus TCPIP”通道支持以下数据类型：二进制变量、有符号 16 位数、无符号 16 位数、有符号 32 位数、无符号 32 位数、浮点数 32 位 IEEE 754、文本变量 8 位字符集、文本变量 16 位字符集。

在 Modbus 连接下创建以下变量：

Quantum						
	名称	数据类型	长度	连接	组	地址
1	tag_i01	二进制变量	1	Quantum		1x100001.1
2	tag_m1	二进制变量	1	Quantum		0x1.1
3	tag_iw1	无符号的16位值	2	Quantum		2x300001
4	tag_mw1	有符号 16 位数	2	Quantum		3x400001

图 14

其中“ tag_i01”地址%i1，对应 modbus 地址 100001:



图 15

“ tag_m1”地址%m1，对应 modbus 地址 000001:

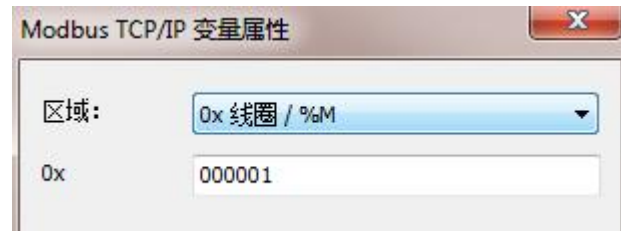


图 16

“ tag_iw1”地址%iw1，对应 modbus 地址 300001:



图 17

“ tag_mw1”地址%mw1，对应 modbus 地址 400001:



图 18

2.2.3 运行结果

下载项目到 PLC，打开 unity pro 项目中的数据表，在线监视 PLC 变量的值：



名称	值	类型
%i1	1	EBOOL
%m1	0	EBOOL
%iw1	8	INT
%mw1	0	INT

图 19

运行 WinCC:



图 20

在 WinCC 中修改变量的值:



图 21

PLC 的数值发生相应变化:

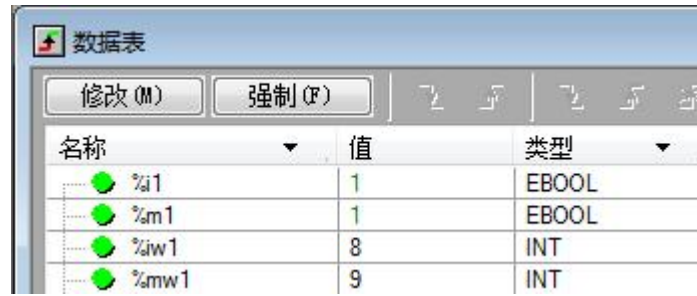


图 22

2.2.4 按位访问字寄存器

WinCC 二进制变量地址可以设定为 4x(或者 3x)的某一位，例如：



图 23

关于这种访问方式，有两点需要注意：

1、位与字的关系

Quantum PLC、WinCC Modbus 中的位与字的对应关系如下表：

位(Modbus)	400001.01	400001.02	400001.08	400001.09	400001.10	400001.16
字	400001 (%mw1)							
位(Quantum)	%mw1.15	%mw1.14	%mw1.8	%mw1.7	%mw1.6	%mw1.0

表 1

2、对于写操作

在更改指定位后，整个字将写回 PLC。但期间并不检查字中的其它位是否已改变。

2.2.5 实数、32 位整数变量

PLC 中的 DINT 和 REAL 变量：

名称	值	设置值	类型
%i1	0		EBOOL
%iw1	8		INT
%mw1	0		INT
%m1	0		EBOOL
%md100	66	66	DINT
%mf102	8.8	8.8	REAL

图 24

WinCC 创建相应变量:

名称	数据类型	长度	格式调整	连接	组	地址
1 tag_i01	二进制变量	1		Quantum		1x100001.1
2 tag_iw1	无符号的16位值	2	WordToUnsignedWord	Quantum		2x300001
3 tag_m1	二进制变量	1		Quantum		0x1.1
4 tag_md100	有符号 32 位数	4	LongToSignedDword	Quantum		3x400100
5 tag_mf102	浮点数 32 位 IEEE 754	4	FloatToFloat	Quantum		3x400102
6 tag_mw1	有符号 16 位数	2	ShortToSignedWord	Quantum		3x400001

图 25

运行 WinCC:

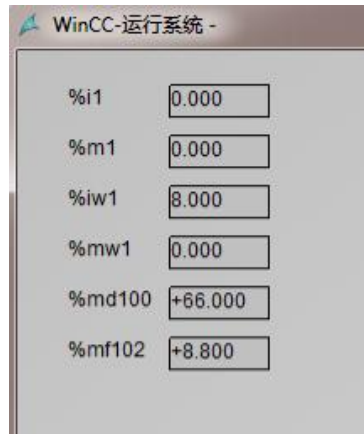


图 26

2.2.6 Modbus TCP/IP 字符串变量

创建如下的字符串变量:

名称	数据类型	长度	格式调整	连接	组	地址
1 tag_i01	二进制变量	1		Quantum		1x100001.1
2 tag_iw1	无符号的16位值	2	WordToUnsignedWord	Quantum		2x300001
3 tag_m1	二进制变量	1		Quantum		0x1.1
4 tag_md100	有符号 32 位数	4	LongToSignedDword	Quantum		3x400100
5 tag_mf102	浮点数 32 位 IEEE 754	4	FloatToFloat	Quantum		3x400102
6 tag_mw1	有符号 16 位数	2	ShortToSignedWord	Quantum		3x400001
7 tag_string	文本变量 8 位字符集	8		Quantum		3x400104

图 27

运行:

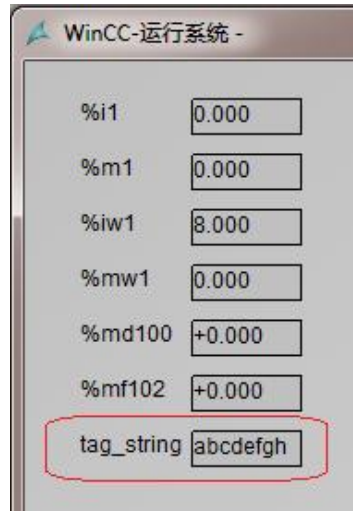


图 28

在 unity pro 数据表中监视（把显示格式改为 ASCII）：

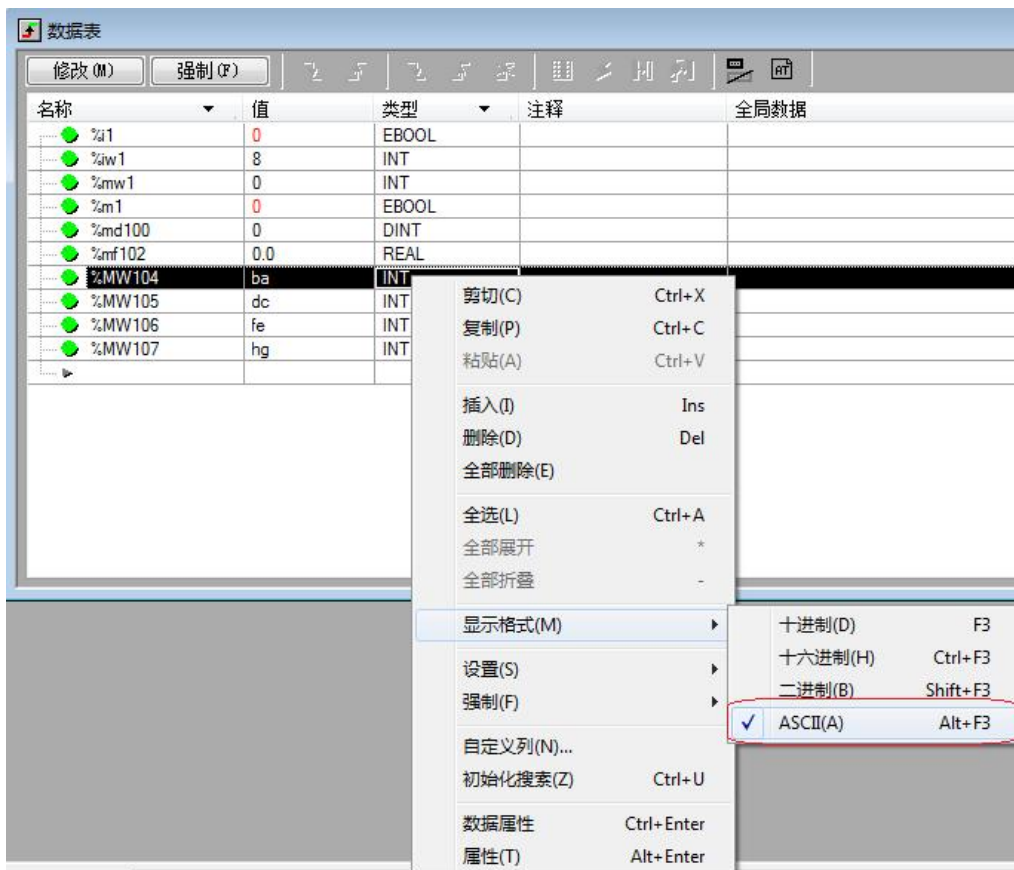


图 29

3 WinCC 和 Premium PLC 通讯

3.1 PLC 组态

3.1.1 硬件配置

在 PLC 的编程软件 Unity Pro 中按如下配置硬件：

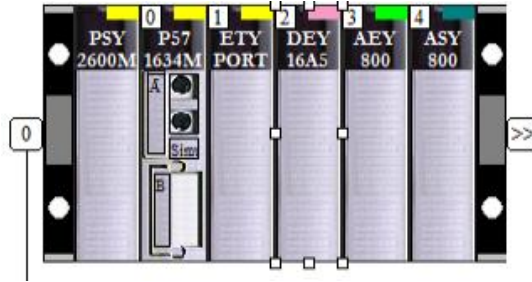


图 30



图 31

3.1.2 网络配置

在通讯→网络下创建新网络，并配置网络 IP 地址及网络类型：

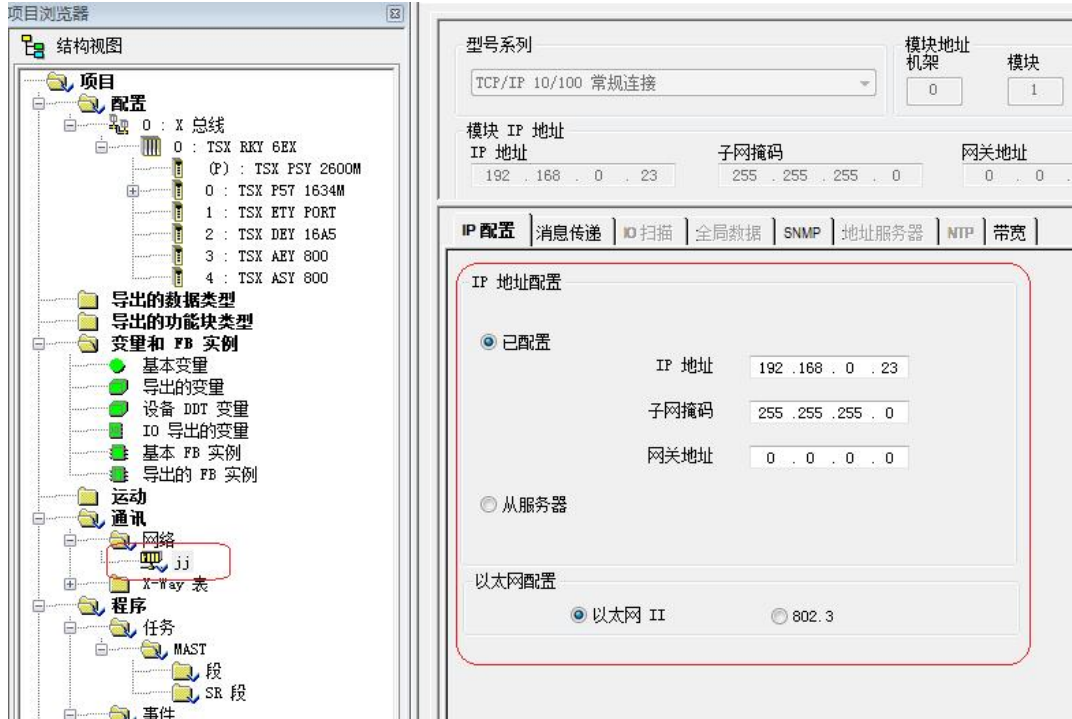


图 32

在硬件配置中，为以太网口分配网络：

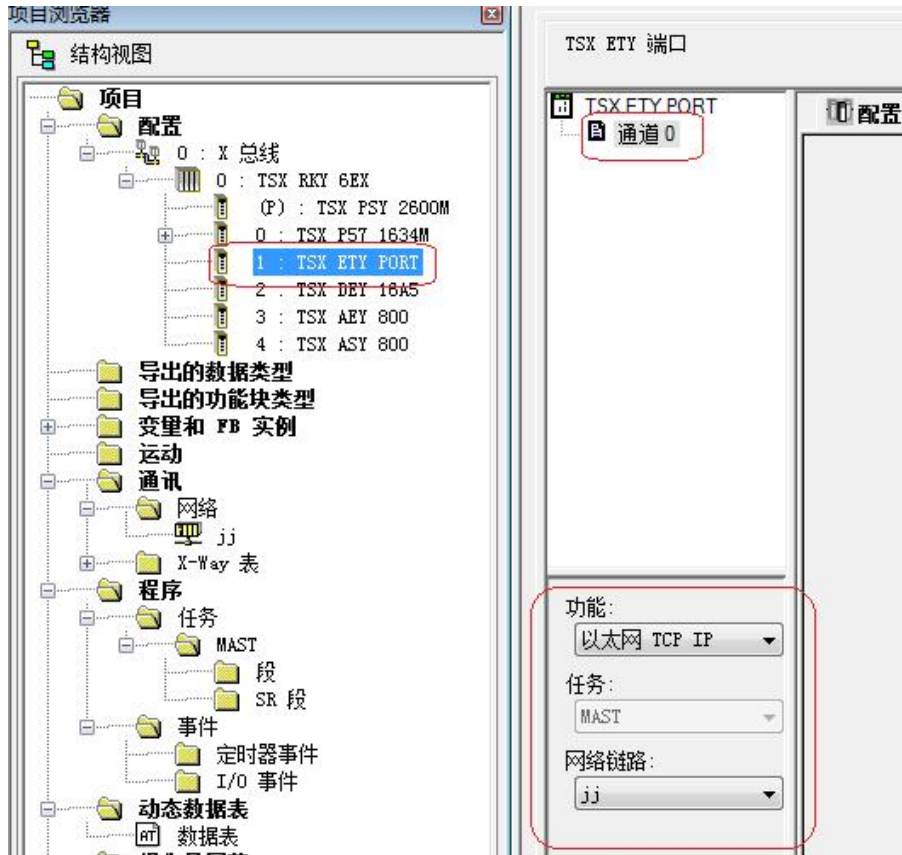


图 33

Premium PLC 只支持 0x 和 4x，也就是 Modbus TCP/IP 客户端只能读取 Premium PLC 的 %m 和 %mw 区。

另外，Premium PLC 的各寄存器编址从 0 开始。

名称	值	类型
%m0	1	EBOOL
%m1	0	EBOOL
%mw0	6	INT
%mw1	8	INT
%md2	100	DINT
%mf4	9.8	REAL

图 34

3.2 WinCC 组态

3.2.1 创建连接

在 Modbus TCP/IP 下新建连接，选择 CPU 类型为“ Premium, Micro”：



图 35



图 36

3.2.2 创建变量

在 WinCC 中创建变量，变量地址中直接输入对应的地址，而不是 Modbus 标准地址。

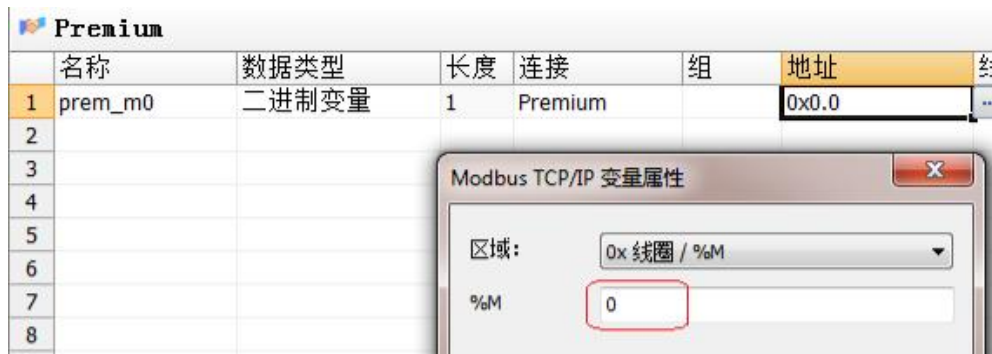


图 37

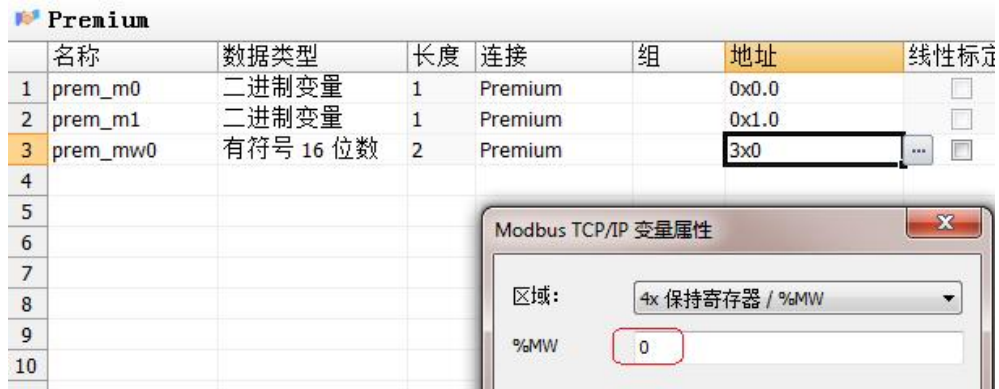


图 38

这里需要说明两点：

虽然 Premium PLC 只支持 0x 和 4x，但 CPU 类型选择“ Premium, Micro” 时，WinCC Modbus TCPIP 驱动也可以读取 1x 和 3x 类型的数据（连接第三方 Modbus TCPIP 设备时）。

Premium PLC 各存储器地址从 0 开始编址，并且 WinCC 变量地址对应的是 PLC 的直接地址，但其和标准 Modbus 也是对应的。例如 %m0 对应 000001， %mw0 对应 400001。

WinCC 所有变量创建如下：

Premium						
	名称	数据类型	长度	连接	组	地址
1	prem_m0	二进制变量	1	Premium		0x0.0
2	prem_m1	二进制变量	1	Premium		0x1.0
3	prem_mw0	有符号 16 位数	2	Premium		3x0
4	prem_mw1	有符号 16 位数	2	Premium		3x1
5	prem_md2	有符号 32 位数	4	Premium		3x2
6	prem_mf4	浮点数 32 位 IEEE	4	Premium		3x4

图 39

3.2.3 运行 WinCC:



图 40

4 WinCC 和第三方 Modbus TCP/IP 设备通讯

当使用 WinCC 和第三方 Modbus TCP/IP 设备通讯时，需要注意以下两方面：

4.1、WinCC 使用的 Modbus 功能码如下表所示：

寄存器区	数据类型	功能码	
		读	写
0x	二进制	01	05
1x	二进制	02	-
3x	二进制 有符号 16 位数 无符号 16 位数 有符号 32 位数 无符号 32 位数 浮点数 32 位 IEEE 754 文本变量 8 位字符集 文本变量 16 位字符集	04	-
4x	二进制 有符号 16 位数 无符号 16 位数	03	06
	有符号 32 位数 无符号 32 位数 浮点数 32 位 IEEE 754 文本变量 8 位字符集 文本变量 16 位字符集		16

表 2

4.2、WinCC Modbus TCP/IP 中各 CPU 类型的区别

双字、字、位的关系

当在连接参数中没有选择“ Swap words in 32-bit values”时，不同 CPU 类型的双字、字、位之间的关系如下：

"984" 和 "Compact, Quantum, Momentum":

双字	400100					
字	400100			400101		
位	400100.1	400100.16	400101.1	400101.16

表 3

"Premium, Micro":

双字	400100 (%mw99)					
字	400101			400100		
位	400101.15	400101.0	400100.15	400100.0

表 4

WinCC Modbus TCP/IP 驱动将按照以上关系来处理读到的字和位。

当在连接参数中选择“ Swap words in 32-bit values”时，WinCC 会相应交换高字和低字的顺序之后再赋值给对应的双字变量（包括实数）。