

延安西门子PLC总代理商

产品名称	延安西门子PLC总代理商
公司名称	浔之漫智控技术-西门子PLC代理商
价格	.00/件
规格参数	
公司地址	上海市松江区石湖荡镇塔汇路755弄29号1幢一层A区213室
联系电话	15221406036

产品详情

121：如何判断电源出错，如：电池故障？ 如果电源(仅S7 - 400)或缓冲区中的一个错误触发一个事件，则CPU操作系统访问OB81。错误纠正后，重新访问OB81。电池故障情况下，如果电池检测中的BAT T.INDIC开关是激活的，则 S7-400仅访问OB81。如果没有组态OB81，则CPU不会进入操作状态STOP。如果OB81不可用，则当电源出错时，CPU仍保持运行。122：如何诊断模拟量模板？ CPU可以中断用户程序的执行处理诊断报警块OB 82。 在用户程序中你可以调用OB 82 中的SFC 51 或SFC 59 以从模板中获得更为详细的诊断信息。 诊断信息在OB 82 退出之前都是一致的当OB 82 退出时将对模板作出诊断中断响应。123：对于小功率输入的触发器，如何避免线路损坏？ 将触发器连接到SM322-8BF，如果它的1信号功率输入低于10mA(例如1mA)，则可能会报告线路损坏。不建议禁用“ Wire breakage ” 诊断，因为这将禁用所有的诊断，并且当SM322-8BF在一定电流范围(0.5到10mA)内工作时，可能无法安全地开关。补救措施：接一个与触发器平行的电阻，这样至少有10mA(24V ca. 2.4 kOhm)的电流。该电阻应该直接安放在触发器上，从而可以方便检测线路损坏。

124：如何为S7 318-2 CPU进行操作系统更新？ 创建一个操作系统更新卡所必需的条件： 1. S7存储卡的类型：2 MB的闪存，订货号6ES7 953-8LM00-0AA0、6ES7 951-1KL00-0AA0 2. STEP 7 V3.1或更新版本 3. 具有外部PROM编程器的PC或者编程设备(PG)，用于编程存储卡使用STEP7 创建操作系统更新卡的步骤： 1. 下载所需的CPU文件； 2. 双击文件名将文件解压缩； 3. 在Simatic Manager中执行“ File/S7 Memory Card/Delete ”，删除微存储卡内容； 4. 在Simatic Manager中选择“ PLC/Update CPU Operating System ”对操作系统进行编程。选择目标目录并“ 打开 ” CPU_HD.UPD文件，开始编程过程； 5. 当标准鼠标指针重新出现时，即完成对操作系统更新卡的编程。执行操作系统更新： 1. 切断CPU所处机架的电源(PS)； 2. 在CPU中插入准备好的操作系统更新卡； 3. 接通CPU所处机架的电源(PS)； 4. 操作系统从微存储卡传送到内部CPU闪存EPROM。此时，将点亮CPU的所有LED(INTF、EXTFFRCE、< span>CRST、RUN、< span>STOP)； 5. 大约2分钟后完成操作系统更新，此时STOP LED将慢速闪动=>请求系统存储器复位； 6. 切断电源，插入操作所需的微存储卡； 7. 接通电源，CPU执行一次自动存储器复位，之后立即进入准备就绪状态。注意:为执行更新，必须从模块中拆除电池。125：模拟量模板的信号转换时间如何计算？ 转换时间由基本转换时间和模板的以下其它处理时间组成 1) 电阻测试 2) 断线监控

基本转换时间直接取决于模板量输入模板的转换方法积分方法瞬时值转换

对于积分转换方法积分时间将直接影响转换时间积分时间取决于你使用STEP 7

所设置的干扰频率抑制。126：是否可能在两个DP从站之间运行一个DP网络？PLC资料网 不能。

不可能建立从站-从站通信。总是需要一个主站来操作DP网络。智能从站可以访问其它从站的初始值(直接数据通信)，但是总是需要DP主站来控制DP网络。智能从站也可以不使用DP网络而独自工作(单机工作)，但是它们不能作为DP主站来提供服务。仅可以与模块接口共同工作。127：对于有些模拟量输入模板你可以使用STEP 7 设定模拟值的平滑指数，它能起到什么作用？使用平滑指数：模拟值的平滑指数可以保证进一步处理的稳定模拟信号；这对于模拟值与被测值之间的缓慢变化相适应非常重要，例如温度测量时。平滑原理：被测值可以通过数字滤波进行平滑平滑可以通过模板根据转换数字化模拟值的规定数量计算平均值来实现；用户可以在多四个等级赋值平滑参数无低平均高这四个等级决定了用于平均计算的模拟信号数量；所选平滑等级越高所平滑的模拟值将越稳定时间越长直到在一个阶跃响应后适用所平滑的模拟信号。128：模拟信号电缆应该单端接地还是2端接地？

为了减少电子干扰对于模拟信号应使用双绞屏蔽电缆模拟信号电缆的屏蔽层应该两端接地。但是如果电缆两端存在电位差将会在屏蔽层中产生等电线连接电流造成对模拟信号的干PLC扰在这种情况下你应该让电缆的屏蔽层一点接地。129：模拟量信号为：7FFFH，是什么原因？

每个模拟量输入模板都可提供被测值7FFFH 与发现故障时的参数赋值无关，该被测值可以是上溢故障或通道无使能130：防止静电放电危险一般有哪些措施？保证良好的接地：在处置对静电敏感的设备时应确保人体工作表面和包装有良好的接地这样可以避免充上静电。避免直接接触：只在不可避免的情况下才接触对静电敏感的设备例如在维修时手持模板但不要接触元件的针脚或印刷板的导体用这种方法使放电能量不会影响对静电敏感的设备 如果必须在模板上进行测量在开始测量之前必须先接触

接地的金属部分使人体放电这种方法只适用于接地的测量设备。131：S7-300/400 PLC支持哪些寻址方式？1) 直接寻址 1.直接地址：例如I0.0，Q1.7，PIW256，PQW512，MD20，T15，C16，DB1.DBB10，L10.0等 2.符号寻址：例如qq，ww.aa等 2) 间接寻址 1.存储器间接寻址：16位指针，例如OPN DB[MW2] 32位指针，例如A I[MD0] 2.寄存器间接寻址：32位指针，例如A I[AR1,P#0.0]，A [AR1,P#0.0] PLC

3) S7-300/400寻址方式图解 132：如何使用指针？指针用来指向一个地址。使用这种寻址方式的优点在于可以在程序运行过程中实现变址。指针用于存储器间接寻址 程序中用于存储器间接寻址的语句包含一个指令、一个地址标识符、以及一个偏移量（偏移量必须在方括号内给出）。下面给出一个双字格式的指针的例子：LP#8.7 把指针值装载到累加器1 TI[MD2] 把指针值传送到MD2 AI[MD2] 查询I8.7的信号状态 = Q[MD2] 给输出位Q8.7赋值 存储区域内部寻址及交叉寻址：程序中采用这些寻址方式的语句包含一个指令以及下列内容：地址标识符、地址寄存器标识符、偏移量。地址

寄存器（AR1、AR2）及偏移量必须写在方括号内。存储区域内部寻址例程：指针不包含指示存储区域的信息：LP#8.7 把指针值装载到累加器1 LAR1 把指针从累加器1装载到AR1 AI[AR1,P#0.0] 查询I8.7的信号状态 = Q[AR1,P#1.1] 给输出位Q10.0赋值 偏移量0.0不起作用。输出Q10.0等于8.7 (AR1) 加偏移量1.1。结果是10.0，而不是9.8。存储区域交叉寻址例程：在存储区域交叉寻址中，指针中包含指示存储区域的信息（例子中为I和Q）。LP#I8.7 把指针值及存储区域标识装载到累加器1 LAR1 把存储区域I和地址8.7装载到AR1 LP#Q8.7 把指针值和地址标识符装载到累加器1 LAR2 把存储区域Q和地址8.7装载到AR2 AI[AR1,P#0.0] 查询输入位I8.7的信号状态 = [AR2,P#1.1] 给输出位Q10.0赋值 偏移量0.0不起作用。输出Q10.0等于8.7 (AR2) 加偏移量1.1。结果是10.0，而不是9.8，

133.如何用一个变量作索引实现在一个域中读一个元素或写一个元素？

一个域(数据类型为ARRAY)是几个相同数据元素的连接。在源代码中一个单空间域的声明执行如下：My_Array: ARRAY[4..11] OF INT; 它标识了一个数据类型为“INTEGER”的8(=11-4+1)元素单空间域。为了访问域中的一个元素，输入域名并在方括号中输入希望访问的元素的号码，比如：My_Array[6]。在S7-SCL(结构化控制语言 - 符合IEC 1131-3的结构化文本)可以使用变量做索引：i:INT:=46 My_Array:=0 在FBD/LAD/STL中，索引必须是个常量，因此限制了可使用的域的范围。语句L P##test首先将地址装载到Accu1，然后通过LAR1语句装载到地址寄存器AR1中(可能是LAR1 P##test的简化格式)。通过寄存器间接寻址将ANY指针(10字节长)中的地址信息读出：代码注释 0 L W[AR1,P#0.0] 读出当前Accu1中参数数据类型的代码。2 L W[AR1,P#2.0] 读出Accu1中的重复因数。重复因数表明通过参数类型ANY传递的数据类型的大小。4 L W[AR1,P#4.0] 读数据块的号或者从ANY指针中读出“0”(这个对应于ANY指针的第4到第5字节)。6 L D[AR1,P#6.0]

133.如何用一个变量作索引实现在一个域中读一个元素或写一个元素？
一个域(数据类型为ARRAY)是几个相同数据元素的连接。在源代码中一个单空间域的声明执行如下：
My_Array: ARRAY[4..11] OF INT; 它标识了一个数据类型为“INTEGER”的8(=11-4+1)元素单空间域。
为了访问域中的一个元素，输入域名并在方括号中输入希望访问的元素的号码，比如：My_Array[6]。
在S7-SCL(结构化控制语言 - 符合IEC 1131-3的结构化文本)可以使用变量做索引：i:INT:=46 My_Array:=0
在FBD/LAD/STL中，索引必须是个常量，因此限制了可使用的域的范围。语句L
P##test首先将地址装载到Accu1，然后通过LAR1语句装载到地址寄存器AR1中(可能是LAR1
P##test的简化格式)。通过寄存器间接寻址将ANY指针(10字节长)中的地址信息读出：代码注释 0 L
W[AR1,P#0.0] 读出当前Accu1中参数数据类型的代码。2 L W[AR1,P#2.0]
读出Accu1中的重复因数。重复因数表明通过参数类型ANY传递的数据类型的大小。4 L W[AR1,P#4.0]
读数据块的号或者从ANY指针中读出“0”(这个对应于ANY指针的第4到第5字节)。6 L D[AR1,P#6.0]

将区域指针读入 Accu1。每次读地址寄存器AR1之后，数据被保存或者缓存(如TLW0)在临时变量“test2”中(ANY指针)。按照Network 1中的语句顺序，传送到功能块FB1的ANY指针被复制到临时变量“test2”中。

137：怎样通过交叉区域寄存器间接寻址访问功能块的本地数据或者功能？

这里必须预先定义本地数据。您可以使用下列语句访问FB或FC的本地数据：

对于存储器间接指针寻址，本地变量必须声明为临时变量(temp)：L P##Lokalvariable LAR1 LW[AR1,P#X.x] 此处不能使用变量类型“bbbb”、“Output”和“In_Out”，将被语法检查视为非法。

138：怎样编程间接访问一个ARRAY类型变量的元素？一个位、字节或者字符域的尺寸是按照字节限制排列的——在所有其它情况下是按照字对齐的。表T6-1中给出了一个域的存储示例。操作系统计算域中单个元素末端位置的位地址。域被分配到从下一个字地址(或字节地址)。下一个数据类型从下一个整字开始(或者整字节)。声明部分：在声明部分，必须定义一个与将被间接寻址的ARRAY有着同一结构的ARRAY。不一定非要声明为IN-OUT变量；也可以声明为TEMP、IN或OUT变量。网络：域宽度(OFFSET)在网络中定义。ARRAY中的单个元素的小常规数据宽度是一个字节；即使在两个变量之间定义一个BOOL。有必要确定相关的域的宽度和确定下一个期望域的起始地址。可使用下面的算法：

地址(指数)： $b = \text{元素长度} * (\text{指数} - 1)$

创建具有不同数据类型的结构时，必须注意，在特定的环境下可能会自动插入填充字节。

保存ARRAY数据类型：示例：ARRAY [1..2,1..3] OF 整数 将生成下列域：

多维域是按照顺序保存的。在本例中整数 [1,1]后面是整数 [1,2]，整数 [1,3]后面是整数 [2,1]。139：STEP

7以哪种格式存储POINTER参数类型？STEP 7以6个字节保存POINTER参数。表4-1显示了用于保存POINTER参数类型的内存区域以及每个字节中保存的数据。i POINTER参数类型保存了下列信息：

DB号(如果DB中没有保存任何数据时为0)。

CPU中的内存区域(表格中列出了不同内存区域的十六进制代码)。数据的地址(按照Byte.Bit格式)。

如果将形式参数声明为POINTER参数类型，则只需要指定内存区域和地址。STEP

7自动将输入项目的格式转换为指针格式。

图片：

西门子S7 - 300

150个问题解答之八141：应用软件冗余当一个长度错误出现导致CPU处于STOP模式应该作些什么？

当在OB100中设置软件冗余功能块FC100时，必须注意下列情况：参数IEC_NO

的背景数据块必须至少有两个字的长度，同样应用于冗余用户程序的参数

DB_NO的数据块也必须时这样。142：在S7程序中，有许多FC、FB块，我怎样对其中的一些块进行保护，

而其它的块可以是开放的呢？1) Step7中，可以先任意打开一块如OB1，在“file”中选择“Generate

Source”或快捷方式“Ctrl+T”，弹出一个画面，填写“bbbbbb

name”如“tt”，然后按OK确认，就会再弹出另一个画面，左边是你的程序中所有的块，如果你需要保护哪些块，

就把这些块移到右边，如FB1、FB2等等，然后退出所有的程序块，再进入SIMATIC Manager中。2) 在S7

Program Sources找到文件tt，双击tt打开，在第四行中加入“Know_How_Protect”，然后编译，

无错后存盘。这样FB1、FB2就被保护住，如想去掉保护，在tt中去掉“Know_How_Protect”编译存盘即可。

注意：千万不要丢失或删除源文件(如tt)，否则程序被保护，用户可以另存到其它目录中，或Export

Source到硬盘中，再删除源文件，这样别人只能看到未保护的块。

143：SFB41,SFB42,SFB43和FB41,FB42,FB43的区别？SFB41(CONT_C),SFB42(CONT_S),SFB43(PULSEGEN)

块和FB41(CONT_C),FB42(CONT_S),FB43(PULSEGEN)的区别：SFB41,42,43与FB41,42,43的参数设置是一

样的。SFB41,42,43只能用于CPU314IFM;FB41,42,43可用于CPU313以上的CPU(除CPU314IFM),因为SFB41,42,

43集成于CPU314IFM中,而在普通CPU中没有这些块。144：如何在多例兼容功能块中找到变量地址？在

多例兼容功能块中，为块参数和静态本地数据加载与地址寄存器AR2有关的地址。如果要在情景数据块

中找到变量的地址，则必须把域内指针(只是AR2的地址)加载到变量地址。样例: TAR2

```
//将地址寄存器加载到累加器AC1(偏移量) UD DW#16#00FF_FFFF // 关闭区域ID L P##variable
```

```
//加载变量地址 +D // 增加偏移量和变量地址 LAR1 // 保存地址寄存器AR1中的结果
```

这样可在AR1中获得地址以进行进一步处理。当功能块作为本地实例调用时，总要使用该方法。比如，

在一个功能块中创建可做为类型“FB”的静态变量的功能块。这样所调用的情景数据块的数据偏移量就存储到情景数据块中。单个实例的偏移量在AR2中。如果不调用作为本地实例的功能块，则可以结束计算。值‘0’就在地址寄存器中。注意事项:如果改变程序中的地址寄存器AR2,那么也改变了变量的偏移量地址。因此就不能保证再对变量进行正确访问。在这种情况下,必须预先保存AR2,并在对程序动作完之后将其复位为初始状态。样例: TAR2 // 在累加器中加载偏移量地址 T #save // 比如, 静态本地数据域 0 中的变量 L DID 0 // 地址是由DI加载的beds L AR2 // 核对 145: 怎样才能访问上一函数的本地数据? 如要访问以前的本地数据 (“V - e > L” 数据), 可以传送一自创建ANY指针或地址到被调用的FC。如果传送的是自创建ANY指针, 区域指针会指向其本地数据, 而这是无意义的。>

因此“V”区域码必须置于ANY指针中。此码必须准确传送。当指定地址时可由编译器完成此工作。

常规程序结构: 以ANY指针或(地址)方式传送: LAR1 P##target //本地定义的ANY变量初始地址 L W#16#87 //为“V”区域载入码 T LB[AR1,P#6.0] //传送到ANY变量 CALL FCxy source1: = #target //传送自创建的ANY指针 source2: = P#L 0.0 BYTE 8 //传送完全L数据区 寻址与通过ANY指针寻址的区别: 下图显示了寻址与通过自创建ANY指针寻址间的区别。在这种情况下“

87”码不传送, 因此ANY指针指向其本地区域。 146: 怎样在DB内进行间接寻址? 请参照:

在DB内进行间接寻址例子 147: 下面是一个在STEP7软件中实现简单指针寻址的例子程序, 您可以参考它的结构实现您的指针寻址功能? siemens指针寻址例子_cpu315

148: 地址和符号寻址的定义和区别是什么? 在STEP7程序中要用到I/O信号、位寄存器、计数器、定时器、数据块及功能块。在程序中可以采用直接地址, 或者更便于读程序的符号寻址, 例如Motor_A_On, 或采用你的公司或行业常用的代码。这样在你的用户程序中就可以通过符号来寻址。

地址: 地址由地址标识符和存储器的位置组成, 例如Q1.0, I1.1, M2.0, FB21等。符号地址

: 如果为地址指定一个符号名, 程序会更便于阅读和查错。STEP7可以自动将符号名翻译成所需的地址。如果要使用符号名存取ARRAY、STRUCT、数据块、局部数据、逻辑块、以及用户自定义数据类型, 必须首先为地址指定一个符号名。例如, 可以为Q0.0指定一个符号名MOTOR_ON, 然后在程序中将MOTOR_ON作为一个地址使用。使用符号地址更容易将您的过程控制项目中的元件与程序中的元件相对应。注意: 在符号名中不允许使用两个连续的下划线, 例如MOTOR__ON。编程支持

: LAD、FBD、STL中地址、参数、块名可用地址或符号表示。用菜单命令View > Display > Symbolic Representation, 可以切换地址和符号地址

为了使用符号地址编程更加容易, 可以同时显示地址和符号名。使用菜单命令 View > Display > Symbol In bbbbbation来激活。这就意味着STL语句的注释包含了更多的信息。不能在该界面下进行修改, 只能在符号表 (symbol table) 或变量声明表 (variable declaration table) 中进行修改。

149: S7-300/400系统存储区域共有多少种? S7

CPU的系统存储区域分为下表中列出的地址区域。在程序中可以根据相应的地址直接读取数据。

150: 如何把一个DATE_AND_TIME变量转换为bbbbbb变量?

为了以日期/时间字符串 (bbbbbb) 形式显示DATE_AND_TIME变量, 必须DATE_AND_TIME变量的专用字节转换为相应的CHAR字符。读取DATE_AND_TIME变量的专用字节并从这两位数字所在ASCII码中创建专用位。将确定了了的CHAR字符存储在字符串的期望位置。示例: L 字节年 SLD 12 SRW 4 SRD 8 L W#16#3030 + I T 数字字符串年

也可以用相反顺序将一个bbbbbb格式的日期/时间字符串转换为一个DATE_AND_TIME变量

延安西门子PLC总代理商

101: 我如何做到对自己的程序块进行加密保护?

您能够通过STEP7软件的KNOW_HOW_PROTECT功能实现对您程序代码的加密保护。

如果您双击鼠标打开经过加密的程序块时, 您只能看到该程序块的接口数据 (即IN, OUT 和 IN/OUT 等类型的参数) 和注释信息, 而程序块中的代码及代码的注释, 临时/静态变量是不能被看到的。同时您也无法对加密保护的程序块做出任何改动。 如何实现程序块保护:

1. 打开程序编辑窗口LAD/FBD/STL;
2. 将要进行加密保护的程序块生成转换为源代码文件 (通过选择菜单 File—>Generate source 生成);
3. 在LAD/FBD/STL 窗口中关闭您的程序块, 并在SIMATIC Manager项目管理窗口的source文件夹中打开上一步所生成的source文件;

4. 在程序块的声明部分, TITLE行下面的一行中输入 " KNOW_HOW_PROTECT " ;

5. 存盘并编译该source文件 (选择菜单File à Save , File à Compile) ; PLC资料网

6. 现在就完成了您程序块的加密保护 ; 102 : 我如何做到对自己的程序块进行加密 ?

取消对程序块的加密保护 1. 打开程序块的Source源文件 ; 2.

删除文件中的KNOW_HOW_PROTECT ; 3. 存盘并编译该source文件 ;

现在程序块的加密保护已经取消。 注意: 如果没有 STL source 源文件, 您是无法对已

经加密的程序块进行编辑的 ; 103 : 在冗余电源配置中, 电源模块掉电, 调用那个OB可以防止CPU停机

? 通过在程序中添加OB83可以防止CPU停机而添加OB81不能防止CPU停机。 通常我们很容易

以为OB81就是处理所有电源故障的OB块, 但对于冗余电源配置中, 某个电源模块掉电故障, 实际上CPU

当作模块插拔故障来处理, 因此需调用OB83。 PLC资料网104 : WinCC作为Modbus主站, 进行浮点数

读取时数据不正确是什么原因 ?

WinCC作为Modbus主站, 进行浮点数读取时, Tag的类型应当选为浮点数32位, 注意地址偏移为

32的整数倍 + 1 (即33、65、97) , 如果采用选用bbbb Bits/Output Bits方式读写 (Function Code

01 , 02) , 在PLC当中应当将一个字的高低8位进行对调。 如果选用bbbb Words/Output

Words方式读写 (Function Code 03 , 04) , 在PLC当中将一个双字的高低16位进行对调, S7300

Modbus程序块的浮点数处理存在误差, 大致在0.5%左右。 105 : CP341

modibus通讯时, modibus功能码如何设定? Function Code 01 , 05 , 15

对应M , Q , T , C等数据区, 可读可写, 具体的字节范围由您在modbus从站组态时设定。 Function

Code 02对应M , I数据区, 只读, 具体的字节范围由您在modbus从站组态时设定。 Function Code

03 , 06 , 16 对应DB区, 可读可写, 在modbus从站组态时设定对应的DB块。 Function Code 04

对应DB区, 只读, 在modbus从站组态时设定对应的DB块。 106 :

CP34x的通讯连接电缆中, 自己制作电缆应该注意哪些? 如果你使用自己制作的电缆, 那

么必须使用带屏蔽外壳的D型接头, 屏蔽线应当和接头的外壳连接, 禁止将电缆的屏蔽层和GND连接,

否则会造成通讯接口的损坏, 请注意RS232不支持热插拔, 所以一定要断电后在插拔通讯电缆 ; 107 :

在用CP340,CP341与第三方产品通讯时(如PC机,用VB,VC读数据)怎样识别联线是否是好的?

在用CP340,CP341与PC机通讯时,常常读不出数据,这样有两个方面原因。

其一,在PLC侧程序是否正确。用上升沿触发P_Send,可以看见TXD灯闪,这样可以判断PLC侧程序没问题;

其二,PC侧 VB,VC程序的问题及电缆线的联接好坏,无从知道,如果联线没问题,就可以集中精力在PC侧找原因,

判定联线是否接好,显得非常重要,有一个小方法可以测出。在PLC侧修改CP340用ASCII方式传送,在发

送DB块中写几个字符形式的数据(如'1','2','A'等)并间隔触发P_Send功能块。 在PC侧修改串口参

数与PLC一致,如波特率,数据长度,停止位,奇偶校验,无握手信号等。在bbbbbs下附件中打开"Hyper

Terminal"建立一个直接到串口的连接,这样就可以读到从PLC中发送的数据如 ' 12A ' 等。同样用"Send

Text File"的功能,把一些字符送到PLC。 这样如果联接电缆是好的,那么可以简单地判断问题是出在哪里。 1

08 : 如何使用CP341模板实现Modbus主站或从站通讯, 我应当订购那些产品? 1) CP341 模块: 6ES7

341-1xH01-0AE0 (x:= A: RS232; x:= B: TTY; x:= C: RS422/485) 2) *(MODBUS master): 6ES7

870-1AA01-0YA0或者*(MODBUS slave): 6ES7 870-1AB01-0YA0109 : 每当断电重启后, CP341模板和调制解

调器 (如SAbbb的modem) 之间的通讯出错是什么原因? 这是因为 DTR、

RTS信号默认为0造成的, 可以在OB1中调用FC6 (V24_SET). 参数RTS和DTR设置为"TRUE". 110 : 配置"CP

340 RS232C"打印工作应注意什么? 调用功能块FB4"P_PRINT"打印字符信息。功能块"P_PRINT"传送

信息给通讯处理器CP340,CP340发送信息给打印机把信息打印出来。为了打印这些信息必须知道参数"P_P

RINT","Pointer DB","Variables DB"和"bbbbat bbbbbb"的相对关系。 111 : CP 341 / CP 441-2

需要多少许可证 (License) ? 在任何情况下, 每一个MODBUS站都需要一个单独的许可证。 6ES7 87

0-1AA01-0YA0包含软件CD, 文档(PDF文件)和硬件狗。如果在同一个站使用几个CP341或CP441通讯处理

器, 则每个CP都需要一个MODBUS许可证副本(copy license), 即不带手册和软件的MODBUS

Dongle, 只有一个硬件狗, 订货号为6ES7 870-1AA01-0YA1。 PLC资料网注意事项 :

对于有两个接口的 CP 441-2, 仅需要一个许可证, 并且每个接口都可以使用任何协议。 112 : 关于CP34x

串行通讯中的起始位、数据位、校验位、停止位的说明? CP34x通讯模板与其他设备进行通讯时,

串口传输的是一个10位或11位的数据帧, 在模块的bbbbeters属性窗口中您可以设置数据帧的格式, 对这个10或11位组成的数据帧的组成遵循以下规则: 帧的位必须是起始位, 始终为1, 不能修改;

起始位之后是7 - 8位数据位, 由用户收发数据组成;

数据位之后是校验位, odd奇校验, even偶校验, any不校验但校验位为0, none无校验位;

校验位之后是停止位，可以设为1 - 2个位，始终为1，不能修改；113:如何通过单向通信从S7-300 / 400访问S7-200上的数据？通过MPI，从带有X_PUT / X_GET的S7-300 / 400可以访问几个S7-200 CPU。在这种情况下，g S7-300 / 400是主站，S7-200是从站。对于CPU22x系列的S7-200 CPU，可以在19,2 kBd和187,5 kBd下工作。从某个固件状态起，CPU21x系列的S7-200 CPU只能控制波特率19,2 kBd。114:使用通讯处理器(CP340, CP341, CP441-2, CP441-1)时对于协议3964(R)和RK512可以使用2线和4线模式吗？在点对点通讯中，在使用上述的通讯处理器时，只能在4线模式中执行协议3964(R)和RK512。115:当传输字符串或ASCII字符应注意什么？当传送字符串必须确认高为254字符。一个bbbbbb(character bbbbbbb)的数据格式大256字符,包括254个字符,大的字符长度在字符串的字节中定义,实际长度在第二个字节中定义,接下来的是字符ASCII格式。当传送字符串数据时,指针必须指向用户数据的位而不是字符串的个位。116:通过PROFIBUS-DP使用SFC 58/59或者SFB 52/53进行通信时,存在通信限制吗？当使用写操作标签和读操作标签(SFC 58/59或者SFB 52/53)时,存在根据CPU而定的对活动工作的限制：CPU 31x-2DP: 每个DP链上允许同时进行4个写操作和4个读操作。CPU 317, CPU 318-2: 每个DP链上允许同时进行8个写操作和8个读操作。CPU 41x-2DP: 每个DP链上允许同时进行8个写操作和8个读操作。在外部链上,每个SFC/SFB同时进行的写操作和读操作的数目总共不能超过32。因此,每个CPU和SFC/SFB上可以同时进行多40个(内部链上8个+外部链上32个)作业。PLC如果在PROFIBUS Net上运行几个通信伙伴,请确保不要激活比指示值更多的作业。一个SFC/SFB可能持续几个CPU循环周期。117: MPI/DP接口的分配如何定义？118: 可以用模拟输入模块SM 331-7NF00进行参数设置的大干扰频率是多少？通过硬件配置,可以设置参数为400Hz的大干扰频率。通过特殊的滤波器可以得到这个数值。小的有效综合时间是10ms——这样就可抑制100Hz的干扰频率。模块的小基本执行时间(释放所有通路)是140ms。119: 关于机架故障,在编程OB86时需要注意哪些事项？只要检测出机架故障,操作系统就会调用OB86,例如:报告机架故障(例如:缺少IM或IM故障导致连接电缆中断),机架的分布式电源故障,在总线系统PROFIBUS-DP的主站系统中,DP从站发生故障。在S7-300和S7-400中处理故障的方法不同:在S7-300中,当中央机架出现故障时,不调用OB 86;只有分布式机架发生故障才调用它。在S7-400中,非分布式或分布式机架出现故障的情况下,均调用OB 86。120:关于300的硬件安装配置应该注意什么？通常一套S7-300 PLC系统有一个主机架,安装有CPU的机架称为主机架,当主机架上的I/O模块(多8块)上的控制点数不够时,可以再增加1-3个扩展机架,每个扩展机架多可安装8个I/O模块,装在4到11槽,3个扩展机架多安装24个I/O模块。在使用扩展机架时,需要机架(Rack),电源模块(PS),接口模块(IM),连接电缆368,S7-300的模块(信号模块、通讯模块、功能模块等)。S7-300的安装机架是一种导轨。你可以使用该导轨,安装S7-300系统的所有模板。S7-300既可以水平安装,也可以垂直安装。要注意其允许的环境温度为:垂直安装:0至40°C、水平安装:0至60°C、CPU和电源必须安装在左侧或底部

121: 如何判断电源出错,如:电池故障? 如果电源(仅S7-400)或缓冲区中的一个错误触发一个事件,则CPU操作系统访问OB81。错误纠正后,重新访问OB81。电池故障情况下,如果电池检测中的BAT T.INDIC开关是激活的,则S7-400仅访问OB81。如果没有组态OB81,则CPU不会进入操作状态STOP。如果OB81不可用,则当电源出错时,CPU仍保持运行。122: 如何诊断模拟量模板? CPU可以中断用户程序的执行处理诊断报警块OB 82。在用户程序中你可以调用OB 82中的SFC 51或SFC 59以从模板中获得更为详细的诊断信息。诊断信息在OB 82退出之前都是一致的当OB 82退出时将对模板作出诊断中断响应。123: 对于小功率输入的触发器,如何避免线路损坏? 将触发器连接到SM322-8BF,如果它的1信号功率输入低于10mA(例如1mA),则可能会报告线路损坏。不建议禁用“Wire breakage”诊断,因为这将禁用所有的诊断,并且当SM322-8BF在一定电流范围(0.5到10mA)内工作时,可能无法安全地开关。补救措施:接一个与触发器平行的电阻,这样至少有10mA(24V ca. 2.4 kOhm)的电流。该电阻应该直接安放在触发器上,从而可以方便检测线路损坏。

124: 如何为S7 318-2 CPU进行操作系统更新? 创建一个操作系统更新卡所必需的条件: 1. S7存储卡的类型: 2 MB的闪存,订货号6ES7 953-8LM00-0AA0、6ES7 951-1KL00-0AA0 2. STEP 7 V3.1或更新版本 3. 具有外部PROM编程器的PC或者编程设备(PG),用于编程存储卡使用STEP7 创建操作系统更新卡的步骤: 1. 下载所需的CPU文件; 2.

双击文件名将文件解压缩； 3. 在Simatic Manager中执行“ File/S7 Memory Card/Delete”，删除微存储卡内容； 4. 在Simatic Manager中选择“ PLC/Update CPU Operating System”对操作系统进行编程。选择目标目录并“打开”CPU_HD.UPD文件，开始编程过程； 5. 当标准鼠标指针重新出现时，即完成对操作系统更新卡的编程。执行操作系统更新： 1. 切断CPU所处机架的电源(PS)； 2. 在CPU中插入准备好的操作系统更新卡； 3. 接通CPU所处机架的电源(PS)； 4. 操作系统从微存储卡传送到内部CPU闪存EPROM。此时，将点亮CPU的所有LED(INTF、EXTFFRCE、CRST、RUN、STOP)； 5. 大约2分钟后完成操作系统更新，此时STOP LED将慢速闪动=>请求系统存储器复位； 6. 切断电源，插入操作所需的微存储卡； 7.

接通电源，CPU执行一次自动存储器复位，之后立即进入准备就绪状态。注意:为执行更新，必须从模块中拆除电池。125：模拟量模板的信号转换时间如何计算？

转换时间由基本转换时间和模板的以下其它处理时间组成 1) 电阻测试 2) 断线监控

基本转换时间直接取决于模板量输入模板的转换方法积分方法瞬时值转换

对于积分转换方法积分时间将直接影响转换时间积分时间取决于你使用STEP 7

所设置的干扰频率抑制。126：是否可能在两个DP从站之间运行一个DP网络？PLC资料网 不能。

不可能建立从站-从站通信。总是需要一个主站来操作DP网络。智能从站可以访问其它从站的初始值(直接数据通信)，但是总是需要DP主站来控制DP网络。智能从站也可以不使用DP网络而独自工作(单机工作)，但是它们不能作为DP主站来提供服务。仅可以与模块接口共同工作。127：对于有些模拟量输入模板你可以使用STEP 7 设定模拟值的平滑指数，它能起到什么作用？

使用平滑指数：模拟值的平滑指数可以保证进一步处理的稳定模拟信号；这对于模拟值与被测值之间的缓慢变化相适应非常重要，例如温度测量时。平滑原理：被测值可以通过数字滤波进行平滑平滑可以通过模板根据转换数字化模拟值的规定数量计算平均值来实现；用户可以在多四个等级赋值平滑参数无低平均高这四个等级决定了用于平均计算的模拟信号数量；所选平滑等级越高所平滑的模拟值将越稳定时间越长直到在一个阶跃响应后适用所平滑的模拟信号。128：模拟信号电缆应该单端接地还是2端接地？

为了减少电子干扰对于模拟信号应使用双绞屏蔽电缆模拟信号电缆的屏蔽层应该两端接地。但是如果电缆两端存在电位差将会在屏蔽层中产生等电线连接电流造成对模拟信号的干 PLC扰在这种情况下你应该让电缆的屏蔽层一点接地。129：模拟量信号为：7FFFH，是什么原因？

每个模拟量输入模板都可提供被测值7FFFH 与发现故障时的参数赋值无关，该被测值可以是上溢故障或通道无使能130：防止静电放电危险一般有哪些措施？

保证良好的接地：在处置对静电敏感的设备时应确保人体工作表面和包装有良好的接地这样可以避免充上静电。避免直接接触：只在不可避免的情况下才接触对静电敏感的设备例如在维修时手持模板但不要接触元件的针脚或印刷板的导体用这种方法使放电能量不会影响对静电敏感的设备 如果必须在模板上进行测量在开始测量之前必须先接触接地的金属部分使人体放电这种方法只适用于接地的测量设备。131：S7-300/400 PLC支持哪些寻址方式？

1) 直接寻址 1.直接地址：例如I0.0，Q1.7，PIW256，PQW512，MD20，T15，C16，DB1.DBB10，L10.0等

2.符号寻址：例如qq，ww.aa等 2) 间接寻址 1.存储器间接寻址：16位指针，例如OPN DB[MW2]

32位指针，例如A I[MD0] 2.寄存器间接寻址：32位指针，例如A I[AR1,P#0.0]，A [AR1,P#0.0] PLC

3) S7-300/400寻址方式图解

语句L P##test首先将地址装载到Accu1，然后通过LAR1语句装载到地址寄存器AR1中(可能是LAR1 P##test的简化格式)。通过寄存器间接寻址将ANY指针(10 字节长)中的地址信息读出：代码 注释 0 L

W[AR1,P#0.0] 读出当前Accu1中参数数据类型的代码。 2 L W[AR1,P#2.0]

读出Accu1中的重复因数。重复因数表明通过参数类型ANY传递的数据类型的大小。 4 L W[AR1,P#4.0]

读数据块的号或者从ANY指针中读出“0”(这个对应于ANY指针的第4到第5字节)。 6 L D[AR1,P#6.0]

将区域指针读入 Accu1。每次读地址寄存器AR1之后，数据被保存或者缓存(如T LW

0)在临时变量“test2”中(ANY指针)。按照Network 1中的语句顺序，传送到功能块FB1

的ANY指针被复制到临时变量“test2”中。

137：怎样通过交叉区域寄存器间接寻址访问功能块的本地数据或者功能？

这里必须预先定义本地数据。您可以使用下列语句访问FB或FC的本地数据：

对于存储器间接指针寻址，本地变量必须声明为临时变量(temp)：L P##Lokalvariable LAR1 L

W[AR1,P#X.x] 此处不能使用变量类型“bbbb”、“Output”和“In_Out”，将被语法检查视为非法。

138：怎样编程间接访问一个ARRAY类型变量的元素？一个位、字节或者字符域的尺寸是按照字节限制排列的——在所有其它情况下是按照字对齐的。表T6-1中给出了一个域的存储示例。操作系统计算域中单个元素末端位置的位地址。域被分配到从下一个字地址(或字节地址)。下一个数据类型从下一个整字开始(或者整字节)。声明部分：在声明部分，必须定义一个与将被间接寻址的ARRAY有着同一结构的ARRAY。不一定非要将ARRAY声明为IN-OUT变量；也可以声明为TEMP、IN或OUT变量。网络：域宽度(OFFSET)在网络中定义。ARRAY中的单个元素的小常规数据宽度是一个字节；即使在两个变量之间定义一个BOOL。有必要确定相关的域的宽度和确定下一个期望域的起始地址。可使用下面的算法：

地址(指数)： $b = \text{元素长度} * (\text{指数} - 1)$

创建具有不同数据类型的结构时，必须注意，在特定的环境下可能会自动插入填充字节。

保存ARRAY数据类型：示例：ARRAY [1..2,1..3] OF 整数 将生成下列域：

多维域是按照顺序保存的。在本例中整数 [1,1]后面是整数 [1,2]，整数 [1,3]后面是整数 [2,1]。139：STEP 7以哪种格式存储POINTER参数类型？STEP 7以6个字节保存POINTER参数。表4-1显示了用于保存POINTER参数类型的内存区域以及每个字节中保存的数据。iPOINTER参数类型保存了下列信息：

图片：