

# 南宁市西门子中国（授权）一级代理商-西门子选型-西门子技术支持-西门子维修服务

|      |   |
|------|---|
| 产品名称 | 南宁市西门子中国（授权）一级代理商-西门子选型-西门子技术支持-西门子维修服务       |
| 公司名称 | 广东湘恒智能科技有限公司                                  |
| 价格   | .00/件   |
| 规格参数 | 变频器:西门子代理商<br>触摸屏:西门子一级代理<br>伺服电机:西门子一级总代理    |
| 公司地址 | 惠州大亚湾澳头石化大道中480号太东天地花园2栋二单元9层01号房（仅限办公）（注册地址） |
| 联系电话 | 18126392341 15267534595                       |

## 产品详情

在西门子S7-300和S7-400的编程中经常需要调用一些系统功能或功能块，在输入参数时经常碰到有指针类型的参数，那么你对指针类型了解吗？我第一次接触指针一词是在学习C语言的时候，指针和链表是C语言中的一个重点难点。在C语言中，指针即存储器地址，在西门子PLC中的指针也是指地址。下面看看西门子POINTER类型的结构：

参数类型POINTER存储下列信息：

· DB编号(或0，如果数据没有存储在DB中) · CPU中的存储区域(下表给出了参数类型POINTER存储器区的十六进制代码) ·

| 十六进制代码  | 存储区 | 描述         |
|---------|-----|------------|
| b#16#81 | I   | 输入区域       |
| b#16#82 | Q   | 输出区域       |
| b#16#83 | M   | 位存储区域      |
| b#16#84 | DB  | 数据块        |
| b#16#85 | DI  | 背景数据块      |
| b#16#86 | L   | 本地的数据(L堆栈) |
| b#16#87 | V   | 先前的本地数据    |

数据的地址(格式为字节.位) STEP 7提供指针格式：p#memory\_area byte.bit\_address.

(如果形式参数被声明为参数类型POINTER，只需要指出存储区域和地址。STEP 7将自动地重定输入指针

的格式。)下面的实例说明如何为以M50.0开始的数据输入参数类型POINTER：P#M50.0 存储器间接寻址：使用存储器间接寻址的程序语句包含一条指令，后面跟有[地址]标识符，最后是一个(地址必须括在方括号内)。根据所用的地址标识符，该指令会将存储于指定地址的数据解释为字或双字指针。完整的数据地址由地址标识符和指针构成，如下例所示。间接寻址的优点是能在程序执行期间动态修改指令的数据地址。存储器间接寻址使用以下两部分地址：1. 地址标识符对于由位逻辑运算寻址的位，可分配地址标识符I、Q、M、L、DIX或DBX。对于由装载指令寻址的字节、字和双字，可使用存储区I、Q、M、L、D和PI，分配IB、IW、ID、DBB、DBW、DBD、DIB、DIW、DID、PIB、PIW、PID等形式的地址标识符。对于由传送指令寻址的字节、字和双字，可使用存储区I、Q、M、L、DB、DI和PQ，分配IB、IW、ID、DBB、DBW、DBD、DIB、DIW、DID、PQB、PQW、PQD等形式的地址标识符。要寻址定时器、计数器或块，可使用T、C、DB、DI、FB、FC形式的区域标识符。2. 括在方括号"[ ]"内的字或双字指针的地址字指针 - 含有定时器(T)、计数器(C)、数据块(DB、DI)或逻辑块(FC、FB)的标识号的字。字指针是十进制整数。双字指针 - 指含有位、字节、字或双字的确切位置的双字。双字指针的格式为：P#字节.位。指针必须存储在下列区域之一，才能进行存储器间接寻址：M - 位存储器L - 本地数据D - 数据块(DB或DI)STAT 静态数据(不是用于具有多重实例能力的块的静态数据)注意如果要寻址使用存储器间接寻址的字节、字或双字，请确保指针的位号为双字格式0。字指针实例：L 5 //将指针值载入ACCU 1。T MW2 //将指针传送到MW2中。L T[MW2] //将5号定时器的当前时间值载入ACCU 1。L C[MW2] //将5号计数器的当前计数值载入ACCU 1。OPN DB[MW2] //将数据块DB5作为共享数据块打开。OPN DI[MW2] //将数据块DB5作为背景数据块打开。双字指针实例：L P#8.7 //将指针值载入ACCU 1。T MD2 //将指针传送到MD2中。A I [MD2] //扫描输入位8.7的状态，并将其= Q [MD2] //信号状态分配给输出位Q 8.7。区域内寄存器间接寻址：使用区域内寄存器间接寻址的程序语句包含一条指令和以下组成部分：地址标识符[地址寄存器标识符,地址]。区域内寄存器间接寻址使用以下两部分地址：1. 地址标识符对于由位逻辑运算寻址的位，可以分配地址标识符I、Q、M、L、DIX或DBX。对于由装载指令寻址的字节、字和双字，可使用存储区I、Q、M、L、D和PI，分配IB、IW、ID、DBB、DBW、DBD、DIB、DIW、DID、PIB、PIW、PID等形式的地址标识符。对于由传送指令寻址的字节、字和双字，可使用存储区I、Q、M、L、DB、DI和PQ，分配IB、IW、ID、DBB、DBW、DBD、DIB、DIW、DID、PQB、PQW、PQD等形式的地址标识符。2. 方括号"[ ]"中的内容包括地址寄存器引用(AR1或AR2)、逗号分隔符","以及双字指针。双字指针 - 指包含位、字节、字或双字的部分地址的双字。双字指针的格式为：P#字节.位。注意请记住您现在使用的是两个格式为"P#字节.位"的指针。一个指针已被jingque表示出来。另一个指针通过对地址寄存器AR1或AR2的引用来确定。如果要寻址字节、字或双字，请确保指针的位号为0。使用寄存器间接寻址的语句不更改地址寄存器的内容。指针实例：L P#8.7 //将指针值载入ACCU 1。LAR1 //用ACCU 1中的指针装载AR1。A I [AR1, P#0.0] //检查输入位I 8.7并将有符号的状态分配给Q 10.0。= Q [AR1, P#1.1] //确切地址8.7在AR1中。偏移量不对其产生影响。确切位置10.0由8.7 (AR1)加上1.1 (偏移量)得出，结果是10.0而不是9.8。区域内寄存器间接寻址实例：A I [AR1,P#4.3] 对其位置是由AR1中的内容加上4个字节，再加上3个位计算得出的输入位，执行逻辑与运算。= DIX [AR2, P#0.0] 将RLO位状态分配给位于AR2中的实例数据位。L IB [AR1, P#10.0] 将输入字节载入ACCU 1中。地址由AR1的内容加上十个字节计算得出。T LD [AR2,P#53.0] 将ACCU 1中的内容传送到本地双字中(该双字的位置由AR2的内容加上53个字节计算得出)。区域内寄存器间接寻址的特点是：地址标示符在方括号前确定，方括号中的指针均为不含存储区信息(如[AR1, P#4.3]中AR1=P#8.7，两个指针都不含有存储区信息)，否则将会与方括号前的地址标示符表示的存储区冲突。跨区域寄存器间接寻址：使用跨区域寄存器间接寻址的程序语句包含一条指令和以下组成部分：地址标识符[地址寄存器标识符,地址]。跨区域寄存器间接寻址使用以下两部分地址：1. 对已寻址的数据对象(地址标识符)大小的规定数据对象大小规定 位(无规定表示是一个位)B 字节W 字D 双字2. 方括号"[ ]"中的内容包括地址寄存器引用(AR1或AR2)、逗号分隔符","以及双字指针。双字指针 - 指包含位、字节、字或双字的部分地址的双字。指针具有以下区域内格式：P#字节.位。注意事先必须已将跨区域双字指针载入到由寄存器间接地址引用的地址寄存器中。跨区域双字指针 - 包含位的部分地址(对于位逻辑指令)或字节、字或双字的部分地址(对于装载和传送指令)的双字。地址前面的区域标识符位于用来指定字节和位。跨区域双字指针格式为：P#区域标识符字节.位。对于由位逻辑指令寻址的位，可以分配跨区域指针区域标识符I、Q、M、DIX或DBX。对于由装载或传送指令寻址的字节、字和双字，可以分配跨区域

指针区域标识符I、Q、M、DIX、DBX或P。注意要在指针中指定外设输入或PI区域，请以P#Px.y形式输入指针。区域被指定为P。在跨区域指针中不能使用外设输出PQ区域。请记住您正使用两个指针：作为偏移量，直接在地址中表示的区域内双字指针，例如P#4.0。存储在地址寄存器(AR1或AR2)中的跨区域双字指针，例如P#Q4.0。如果要访问通过直接寻址方式进行寻址的字节、字或双字，请确保这两个指针的位号均为0。使用寄存器间接寻址的语句不更改地址寄存器的内容。