

西安西门子PLC代理商|供货商

| | |
|------|-------------------------------|
| 产品名称 | 西安西门子PLC代理商 供货商 |
| 公司名称 | 浔之漫智控技术（上海）有限公司 |
| 价格 | .00/件 |
| 规格参数 | 品牌:西门子 型号:PLC 西门子:代理商 |
| 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 |
| 联系电话 | 18717946324 18717946324 |

产品详情

西安西门子PLC代理商|供货商

telnet方式

通过telnet登录上板子，执行命令AHC port [baud]

参数port：串口号，值为0-6，0表示DEBUG串口，1表示USBOTG转虚拟串口，2-6分别表示板子的COM2-COM6。

参数baud：波特率，可选参数，如果不填表示保持原波特率，支持1200，2400，4800，9600，19200，38400，57600，115200。当port为0时，baud固定为115200，当port为1时，baud值不生效。

命令执行后，DEBUG口可以看到打印提示信息。

打印格式说明

打印结果为数行，其中每行的格式均为：类型 ID号 占用情况 名称

以下图一次打印的部分截图为例：

类型

PID表示为process进程。TID表示为上面进程下的thread线程。

ID号

即进程ID值或线程ID值。

占用情况

显示格式为 K n% U m% total%

n值为该进程或线程在Kernel系统层的占用

m值为该进程或线程在User用户层的占用

total值为总占用，它应当等于n+m的和

进程下各个线程total占用和应当等于进程的total占用

名称

进程名即EXE的名称，线程默认没有名称，下一节会介绍如何给线程命名，从而能在App Helper中显示出来。

进程及线程监视说明

AppHelper会打印系统下所有的进程的CPU占用信息。

只有在NandFlash目录下的exe生成的进程会额外打印出它下面所有线程的CPU占用信息。

默认情况下，生成的线程只有ID号，没有名称，如果线程较多会不便于查看。我们可以通过简单代码给线程命名。

以光盘里的串口例程SPT_HEX为例：

添加一个结构体的定义

```
typedef struct _THREAD_INDEX
```

```
{
```

```
    DWORD dwSize;
```

```
DWORD dwThreadID;
```

```
TCHAR szThreadName[32];
```

```
_THREAD_INDEX* pNext;
```

```
}THREAD_INDEX;
```

在创建线程后给线程命名

这里把串口接收线程命名为"CommRecvTread"

```
hRecvThread = CreateThread(0, 0, CommRecvTread, this, 0, &m_dwTID);
```

```
HANDLE hHLP;
```

```
DWORD dwLen;
```

```
hHLP = CreateFile(L"HLP1:", GENERIC_READ | GENERIC_WRITE, 0, 0, OPEN_EXISTING, 0, 0);
```

```
THREAD_INDEX threadIndex;
```

```
wsprintf(threadIndex.szThreadName, L"CommRecvTread");
```

```
threadIndex.dwThreadID = m_dwTID;
```

```
threadIndex.dwSize = sizeof(THREAD_INDEX);
```

```
WriteFile(hHLP, &threadIndex, sizeof(THREAD_INDEX), &dwLen, NULL);
```

```
CloseHandle(hHLP);
```

在结束线程后取消命名

线程结束后应当手动将命名取消掉，避免不必要的显示错误，设置线程名为空，即可取消原命名。

```
wsprintf(threadIndex.szThreadName, L"");
```

命名线程后再使用AppHelper查看，启动接收线程后，就可以看到CommRecvTread这个线程，另外个没有命名的线程为SerialPort程序的主线程。

计算原理及误差说明

CPU占用时间是通过计算一段时间内（AppHelper设置为2000毫秒）CPU空闲tick值与这段时间里CPU运算周期tick值得出。

CPU空闲tick值 = CPU空闲tick计数t2 - CPU空闲tick计数t1

CPU总周期tick值 = CPU总周期tick计数t2 - CPU总周期tick计数t1

CPU占用 = 1 - (CPU空闲tick值/CPU总周期tick值) ×

进程或线程的CPU占用，是通过计算一段时间CPU运算周期tick值，和这段周期里Kernel或User运行线程或进程的tick值，通过相除得到