以太网络扩展多外设接口芯片 CH9130

Windows 应用程序开发手册

版本：V1.0

http://wch.cn

一. 简介

CH9130是一款用于以太网络扩展多外设接口的芯片，可同时扩展：2路UART、16路GPIO、2路SPI、3路PWM、1路3通道ADC、8位被动并口或8/16位总线接口。PC端提供了各个外设接口的API函数库，所有外设的初始化、配置及读写操作均由用户直接调用接口库中的API函数完成。

二. 开发流程

在 windows 系统下，针对每个设备的操作有以下大体流程：

1. PC 端应用程序使用 CH9130Connect 去连接 CH9130 设备，函数调用成功后返回设备句柄，设备句柄代表当前打开的连接资源，一般后续的操作都以此设备句柄为第 1 个参数；
2. CH9130GetDevStatus 获取设备状态，检查设备是否空闲或是被其他 PC 占用，注意：CH9130GetDevStatus 是可选的操作，因为如果设备被一台 PC 占用，另一台 PC 再去打开的话，打开失败；
3. 调用 CH9130OpenXXXX 函数去打开某个具体的外设。打开设备成功后，可以使用 CH9130SetCallBack 设置在通信中出错时的回调函数，推荐使用这种回调机制；
4. 调用 CH9130InitXXXX 初始化某个外设参数。

三. 函数说明

下面函数是 CH9130 内部各个外设接口可公用的函数：

CH9130Init 函数

初始化 CH9130DLL.DLL 库，次函数在应用程序里调用一次即可。

函数原型：BOOL WINAPI CH9130Init();

参数 ：无

返回值：成功返回 TRUE，失败返回 FALSE.

CH9130StrToAddr：

将包含的 IP 字符串(例如"192.168.1.1")转换成 4 字节 ULONG，大端模式。等价于 htonl(inet_addr(cp))

函数原型：ULONG WINAPI CH9130StrToAddr(const char* cp);

参数 ： cp，以 NULL 结尾的 IP 地址字符串(例如"192.168.1.1")

返回值： 4 字节网络 IP 地址，大端模式。返回值直接给 CH9130Connect 的第 1 个参数使用。

CH9130Connect：

连接 CH9130 服务器，返回设备句柄。

函数原型：PVOID WINAPI CH9130Connect(ULONG ulP,USHORT uPort);

参数 ：ulP， 设备的网络 IP 地址。

uPort，设备的端口。

返回值：返回设备句柄。

CH9130GetDevStatus:

获取设备状态，检查设备是否被其他客户端打开。

函数原型：BOOL WINAPI CH9130GetDevStatus(PVOID pDevInf,PUSHORT pDevStatus) ;

参数 ：pDevInf 设备句柄。

pDevStatus 设备状态，见下文 BIT_PERI_GPIO 等定义。

返回值：成功返回 TRUE，失败返回 FALSE。

```
#define BIT_PERI_GPIO   ( 1<<0 )        /* GPIO 状态，0：关闭，1：占用 */
#define BIT_PERI_UART0  ( 1<<1 )        /* UART0 状态，0：关闭，1：占用 */
#define BIT_PERI_UART1  ( 1<<2 )        /* UART1 状态，0：关闭，1：占用 */
#define BIT_PERI_SPI0       ( 1<<3 )        /* SPI0 状态，0：关闭，1：占用 */
#define BIT_PERI_SPI1       ( 1<<4 )        /* SPI1 状态，0：关闭，1：占用 */
#define BIT_PERI_PWM0   ( 1<<5 )        /* PWM0 状态，0：关闭，1：占用 */
#define BIT_PERI_PWM1   ( 1<<6 )        /* PWM1 状态，0：关闭，1：占用 */
#define BIT_PERI_PWM2   ( 1<<7 )        /* PWM2 状态，0：关闭，1：占用 */
#define BIT_PERI_PARA   ( 1<<8 )        /* 并口状态，0：关闭，1：占用 */
#define BIT_PERI_ADC        ( 1<<9 )        /* ADC 状态，0：关闭，1：占用 */
```

CH9130Disconnect：

断开与 CH9130 服务器的连接，并释放 CH9130Connect 所分配的资源。

函数原型：BOOL WINAPI CH9130Disconnect(PVOID pDevInf,ULONG TimeOut);

参数 ：pDevInf，设备句柄

TimeOut，超时时间，单位：毫秒。超时主要是用于等待服务器返回的 FIN

返回值：返回设备句柄。

CH9130SetCallBack：

设置外设异常时的回调函数，CH9130DLL 库在检查到外设没有应答，网络异常时会调用 pFuc 函数。

函数原型：BOOL WINAPI CH9130SetCallBack(PVOID pDevInf ,pCheckDevRmv pFuc,PVOID pArg);

参数 ：pDevInf，设备句柄。

pFuc， 回调函数地址。

pArg， 调用 pFuc 时，传递给 pFuc 的参数。

返回值：成功返回 TRUE，失败返回 FALSE。

CH9130GetLastError：

返回最后的错误码。

函数原型：DWORD WINAPI CH9130GetLastError(PVOID pDevInf);

参数 ：pDevInf，设备句柄。

返回值：返回错误码，错误码如下。

```
#define CH9130_STATUS_OK                    ( 0 )   //无错误
#define CH9130_STATUS_WIN32_ERROR           ( -1 )      //Windows API 错误
#define CH9130_STATUS_NETWORK_ERROR     ( -2 )  //NETWORK API 错误
#define CH9130_STATUS_ACK_TIMEOUT           ( -3 )  //设备应答超时
#define CH9130_STATUS_INVALID_PARAMETER     ( -4 )  //无效的参数
#define CH9130_STATUS_DEV_USED              ( -5 )  //设备已被其他 PC 打开
#define CH9130_STATUS_DEV_REMOVE            ( -6 )  //设备已移除
```

CH9130GetNetWorkErrorID：
返回网络错误码，在 CH9130GetLastError 返回 CH9130_STATUS_NETWORK_ERROR 时，可以调用 CH9130GetNetWorkErrorID 返回具体的网络错误码。
函数原型：DWORD WINAPI CH9130GetNetWorkErrorID(PVOID pDevInf);
参数 ：pDevInf，设备句柄
返回值：返回错误码。
内部缓冲区：在 UART0，UART1，SPI0 设备模式，ADC 流模式，PARA 被动模式时，DLL 库会先把数据缓冲下来。
CH9130ClearBufData：
清空内部缓冲区。
函数原型：BOOL WINAPI CH9130ClearBufData(PVOID pDevInf);
参数 ：pDevInf，设备句柄。
返回值：成功返回 TRUE，失败返回 FALSE。
CH9130QueryBufLen：
查询内部缓冲区字节数。
函数原型：BOOL WINAPI CH9130QueryBufLen(PVOID pDevInf , PULONG pRecvLen);
参数 ：pDevInf，设备句柄。
pRecvLen，函数返回时返回内部缓冲区字节数。
返回值：成功返回 TRUE，失败返回 FALSE。

GPIO 函数

CH9130OpenGPIO：
打开 GPIO 设备
函数原型：BOOL WINAPI CH9130OpenGPIO(PVOID pDevInf);
参数 ：pDevInf 设备句柄
返回值：打开成功返回 TRUE，打开失败返回 FALSE。
CH9130CloseGPIO：
关闭 GPIO 设备
函数原型：BOOL WINAPI CH9130CloseGPIO(PVOID pDevInf);
参数 ：pDevInf 设备句柄
返回值：打开成功返回 TRUE，打开失败返回 FALSE。
CH9130SetGPIODir0:
设置 GPIO 的 D0~D7 方向。
函数原型：BOOL WINAPI CH9130SetGPIODir0(PVOID pDevInf,UCHAR Data);
参数 ： pDevInf 设备句柄
Data BIT0 对应 D0，BIT7 对应 D7,相应位为 0 表示输入，为 1 表示输出。
返回值：成功返回 TRUE，打开失败返回 FALSE。
CH9130SetGPIODir1:
设置 GPIO 的 D8~D15 方向。
函数原型：BOOL WINAPI CH9130SetGPIODir1(PVOID pDevInf,UCHAR Data);
参数 ： pDevInf 设备句柄
Data BIT0 对应 D8，BIT15 对应 D7,相应位为 0 表示输入，为 1 表示输出。
返回值：成功返回 TRUE，打开失败返回 FALSE。
CH9130ReadGPIOData0:

读取 GPIO 的 D0~D7 的值。
函数原型：BOOL WINAPI CH9130SetGPIODir0(PVOID pDevInf,UCHAR Data);
参数 ： pDevInf 设备句柄
Data BIT0 对应 D0，BIT7 对应 D7，相应位为 0 表示低电平，为 1 表示高电平。
返回值：成功返回 TRUE，打开失败返回 FALSE。
CH9130ReadGPIOData1:
读取 GPIO 的 D8~D15 的值。
函数原型：BOOL WINAPI CH9130SetGPIODir1(PVOID pDevInf,UCHAR Data);
参数 ： pDevInf 设备句柄
Data BIT0 对应 D8，BIT15 对应 D7，相应位为 0 表示低电平，为 1 表示高电平。
返回值：成功返回 TRUE，打开失败返回 FALSE。
CH9130WriteGPIOData0:
设置 GPIO 的 D0~D7 的值。
函数原型：BOOL WINAPI CH9130SetGPIODir0(PVOID pDevInf,UCHAR Data);
参数 ： pDevInf 设备句柄
Data BIT0 对应 D0，BIT7 对应 D7，相应位为 0 表示低电平，为 1 表示高电平。
返回值：成功返回 TRUE，打开失败返回 FALSE。
CH9130WriteGPIOData1:
设置 GPIO 的 D8~D15 的值。
函数原型：BOOL WINAPI CH9130SetGPIODir1(PVOID pDevInf,UCHAR Data);
参数 ： pDevInf 设备句柄
Data BIT0 对应 D8，BIT15 对应 D7，相应位为 0 表示低电平，为 1 表示高电平。
返回值：成功返回 TRUE，打开失败返回 FALSE。

UART0 函数

CH9130OpenUART0:
打开 UART0 设备
函数原型：BOOL WINAPI CH9130OpenUART0(PVOID pDevInf);
参数 ：pDevInf 设备句柄。
返回值：打开成功返回 TRUE，打开失败返回 FALSE。
CH9130CloseUART0：
关闭 UART0 设备。
函数原型：BOOL WINAPI CH9130CloseUART0(PVOID pDevInf);
参数 ：pDevInf 设备句柄
返回值：成功返回 TRUE，失败返回 FALSE。

```
BAUD rate setting
#define B300        300
#define B600        600
#define B1200       1200
```

```
#define B2400       2400
#define B4800       4800
#define B9600       9600
#define B14400  14400
#define B19200  19200
#define B28800  28800
#define B38400  38400
#define B57600  57600
#define B76800  76800
#define B115200     115200
#define B230400     230400
#define B460800     460800
#define B921600 921600

Parity define
#define P_ODD   0x00        //奇校验
#define P_EVEN  0x01        //偶校验
#define P_MARK  0x02        //标志位
#define P_SPC       0x03        //空白位
#define P_NONE  0x04        //无校验

Stop bits define
#define STOP_1  0x00
#define STOP_2  0x01

Data bits define
#define BIT_5       0x00
#define BIT_6       0x01
#define BIT_7       0x02
#define BIT_8       0x03
```

CH9130InitUART0：

初始化 UART0 设备参数

函数原型：BOOL WINAPI CH9130InitUART0(PVOID pDevInf,ULONG ulRate,UCHAR ucCheck,UCHAR ucStop,UCHAR ucData,UCHAR ucFIFO,UCHAR ucFlowCtrl,BOOL bHalf);

参数 ：pDevInf 设备句柄

ulRate 串口波特率(例如 9600 波特率，ulRate 直接传值 9600)，见上文 BAUD rate setting

ucCheck 检验位，见上文 Parity define

ucStop 停止位，见上文 Stop bits define

ucData 数据位，见上文 Data bits define

ucFlowCtrl 流控使能，0：禁止，1：使能

ucFIFO FIFO 大小，0=1byte, 1=4bytes(推荐值), 2=8bytes, 3=14bytes

bHalf 半双工模式控制，0：禁止，1：使能

返回值：成功返回 TRUE，失败返回 FALSE。

CH9130WriteUART0Data：
向 UART0 设备发送数据，注意超时时间要考虑串口的波特率问题，例如在 600 波特率发送 512 字节的话，所需时间差不多是 9 秒，超时时间要设置成大于 9 秒。
函数原型：BOOL WINAPI CH9130WriteUART0Data(PVOID pDevInf,PVOID pData,ULONG Len,ULONG TimeOut)；
参数 ：pDevInf 设备句柄
pData 缓冲区数据
Len 缓冲区长度，长度不要超过 512 字节
TimeOut 超时时间，单位：毫秒。在网络异常或者设备异常时，会出现没有应答的情况，这时函数超时返回
返回值：成功返回 TRUE，失败返回 FALSE。
CH9130ReadUART0Data：
读 UART0 设备数据，注意超时时间是内部缓冲区没有数据时的等待时间。
函数原型：BOOL WINAPI CH9130ReadUART0Data(PVOID pDevInf,PVOID pData,PULONG Len,ULONG TimeOut)；
参数 ：pDevInf 设备句柄
pData 缓冲区数据
Len 函数调用前是缓冲区长度，长度不要超过 512 字节。函数返回后是实际读到的数据长度。
TimeOut 超时时间，单位：毫秒。在网络异常或者设备异常时，会出现没有应答的情况，这时函数超时返回。
返回值：成功返回 TRUE，失败返回 FALSE。

UART1 函数

CH9130OpenUART1 ， CH9130CloseUART1 ， CH9130InitUART1 ， CH9130WriteUART1Data CH9130ReadUART1Data 参数与 UART0 一样，见上文。

SPI0 函数

CH9130OpenSPI0:
打开 SPI0 设备。
函数原型：BOOL WINAPI CH9130OpenSPI0(PVOID pDevInf)；
参数 ：pDevInf 设备句柄。
返回值：打开成功返回 TRUE，打开失败返回 FALSE。
CH9130InitSPI0：
初始化 SPI0 设备参数。
函数原型：BOOL WINAPI CH9130InitSPI0(PVOID pDevInf,BOOL bDevMod,BOOL ucSckMod3,UCHAR ucClkDiv,BOOL bHostFlowEnable)；
参数 ：pDevInf 设备句柄。
bDevMod SPI 主从模式选择位，0：主机模式，1：设备模式。
ucSckMod3 主机模式下时钟模式配置位，0：模式 0（SCK 空闲为低电平），1：模式 3（SCK 空闲为高电平）。
ucClkDiv 分频系数，>=2
bHostFlowEnable 从机模式，流控使能，0：禁止，1：使能。
返回值：打开成功返回 TRUE，打开失败返回 FALSE。
CH9130CloseSPI0：
关闭 SPI0 设备。

函数原型：BOOL WINAPI CH9130CloseSPIO(PVOID pDevInf)；
参数 ：pDevInf 设备句柄。
返回值：成功返回 TRUE，失败返回 FALSE。

CH9130WriteDataSPIOHost：
SPIO 主机模式下写数据。
函数原型：BOOL WINAPI CH9130WriteDataSPIOHost(PVOID pDevInf,PVOID pData,ULONG Len,ULONG TimeOut)；
参数 ：pDevInf 设备句柄
pData 缓冲区数据
Len 缓冲区长度，长度不要超过 512 字节
TimeOut 超时时间，单位：毫秒。在网络异常或者设备异常时，会出现没有应答的情况，这时函数超时返回。
返回值：成功返回 TRUE，失败返回 FALSE。
CH9130ReadDataSPIOHost：
SPIO 主机模式下读数据。
函数原型：BOOL WINAPI CH9130ReadDataSPIOHost(PVOID pDevInf,PVOID pData,ULONG Len,ULONG TimeOut)；
参数 ：pDevInf 设备句柄
pData 缓冲区数据
Len 函数调用前是缓冲区长度，长度不要超过 512 字节。函数返回后是实际读到的数据长度。
TimeOut 超时时间，单位：毫秒。在网络异常或者设备异常时，会出现没有应答的情况，这时函数超时返回。
返回值：成功返回 TRUE，失败返回 FALSE。
CH9130WriteDataSPIODevice：
SPIO 设备模式下写数据。
函数原型:BOOL WINAPI CH9130WriteDataSPIODevice(PVOID pDevInf,PVOID pData,ULONG Len,ULONG TimeOut)；
参数 ：pDevInf 设备句柄
pData 缓冲区数据
Len 缓冲区长度，长度不要超过 512 字节
TimeOut 超时时间，单位：毫秒。在网络异常或者设备异常时，会出现没有应答的情况，这时函数超时返回。
返回值：成功返回 TRUE，失败返回 FALSE。
CH9130ReadDataSPIODevice：
SPIO 设备模式下读数据。
函数原型:BOOL WINAPI CH9130ReadDataSPIODevice(PVOID pDevInf,PVOID pData,PULONG Len,ULONG TimeOut)；
参数 ：pDevInf 设备句柄
pData 缓冲区数据
Len 函数调用前是缓冲区长度，长度不要超过 512 字节。函数返回后是实际读到的数据长度。
TimeOut 超时时间，单位：毫秒。在网络异常或者设备异常时，会出现没有应

答的情况，这时函数超时返回。
返回值：成功返回 TRUE，失败返回 FALSE。

SPI1 函数

CH9130OpenSPI1 ， CH9130InitSPI1 ， CH9130CloseSPI1 ， CH9130WriteDataSPI1Host ，CH9130ReadDataSPI1Host 参数与 SPI0 一样，见上文。

PWM0 函数

CH9130OpenPWM0:
打开 PWM0 设备.
函数原型：BOOL WINAPI CH9130OpenPWM0(PVOID pDevInf);
参数 ：pDevInf 设备句柄。
返回值：打开成功返回 TRUE，打开失败返回 FALSE。
CH9130InitPWM0：
初始化 PWM0 设备参数。
函数原型：BOOL WINAPI CH9130InitPWM0(PVOID pDevInf,UCHAR ucRpt,BOOL bDMALoop,BOOL bPOLAR,ULONG PwmCyc,UCHAR PwmDuty);
参数 ：pDevInf 设备句柄。
ucRpt 重复次数，0:重复 1 次,1:重复 4 次,2:重复 8 次,3:重复 16 次。
bDMALoop DMA 地址循环模式，0：禁止，1：使能。
bPOLAR 默认输出极性设置位，0：默认低电平，高电平有效，1：默认高电平，低电平有效。
PwmCyc 周期，纳秒：10～2680000000，最小值由主频决定。
PwmDuty 占空比，0%～100%。
返回值：打开成功返回 TRUE，打开失败返回 FALSE。
CH9130ClosePWM0：
关闭 PWM0 设备。
函数原型：BOOL WINAPI CH9130ClosePWM0(PVOID pDevInf);
参数 ：pDevInf 设备句柄。
返回值：成功返回 TRUE，失败返回 FALSE。
CH9130SetPWM0Duty:
设置 PWM0 设备占空比.
函数原型：BOOL WINAPI CH9130SetPWM0Duty(PVOID pDevInf,ULONG Data);
参数 ：pDevInf 设备句柄。
Data 百分比值，0~100。
返回值：成功返回 TRUE，失败返回 FALSE。
CH9130SetPWM0Out：
使能 PWM0 设备输出。
函数原型：BOOL WINAPI CH9130SetPWM0Out(PVOID pDevInf,BOOL bEnable);
参数 ：pDevInf 设备句柄。
Data TRUE:开始输出 PWM 波形，FALSE:停止输出。
返回值：成功返回 TRUE，失败返回 FALSE。

PWM1 函数

CH9130OpenPWM1 ， CH9130InitPWM1 ， CH9130ClosePWM1 ， CH9130SetPWM1Duty ，CH9130SetPWM1Out 参数与 PWM0 一样，见上文。

PWM2 函数

CH9130OpenPWM2 ， CH9130InitPWM2 ， CH9130ClosePWM2 ， CH9130SetPWM1Duty ，CH9130SetPWM1Out 参数与 PWM0 一样，见上文。

ADC 函数

CH9130OpenADC：

打开 ADC 设备。

函数原型：BOOL WINAPI CH9130OpenADC(PVOID pDevInf)；

参数 ：pDevInf 设备句柄。

返回值：打开成功返回 TRUE，打开失败返回 FALSE。

CH9130InitADC：

初始化 ADC 设备参数。

函数原型：BOOL WINAPI CH9130InitADC(PVOID pDevInf,UCHAR ucCycClk,BOOL bAutoWid,UCHAR ucClkDiv,UCHAR ucAdcChan,UCHAR ucAutoCnt,USHORT usRefValue)；

参数 ：pDevInf 设备句柄

ucCycClk ADC 时钟采样模式，0：手动采样，其它：自动采样周期的时钟数

bAutoWid 自动采样脉宽设置位，0：1 时钟周期，1：2 时钟周期

ucClkDiv 时钟分频系数

ucAdcChan 采样通道

ucAutoCnt 自动采样计数

usRefValue 未实现

返回值：打开成功返回 TRUE，打开失败返回 FALSE。

CH9130CloseADC：

关闭 ADC 设备。

函数原型：BOOL WINAPI CH9130CloseADC(PVOID pDevInf)；

参数 ：pDevInf 设备句柄。

返回值：成功返回 TRUE，失败返回 FALSE。

CH9130SetADCAutoData:

使能 ADC 自动采样。

函数原型：BOOL WINAPI CH9130SetADCAutoData(PVOID pDevInf,BOOL bEnable)；

参数 ：pDevInf 设备句柄。

bEnable TRUE:开启 ADC 自动采样，FALSE:关闭 ADC 自动采样。

返回值：成功返回 TRUE，失败返回 FALSE。

CH9130ReadADCData：

主动采样一次 ADC 数据。

函数原型：BOOL WINAPI CH9130ReadADCData(PVOID pDevInf,PUSHORT pData)；

参数 ：pDevInf 设备句柄。

pData 函数返回 ADC 值，注意：16 位为 1 个采样数据。

返回值：成功返回 TRUE，失败返回 FALSE。

CH9130ChangeADCChan：

改变 ADC 采样通道。

函数原型：BOOL WINAPI CH9130ChangeADCChan(PVOID pDevInf,UCHAR Data)；

参数 ：pDevInf 设备句柄。

Data 通道值，0：通道 0，1：通道 1，2：通道 2。

返回值：成功返回 TRUE，失败返回 FALSE。

CH9130ReadADCDataStream：

自动采样的 ADC 数据。

函数原型：BOOL WINAPI CH9130ReadADCDataStream(PVOID pDevInf,PUSHORT pData,PULONG pLen,ULONG TimeOut)；

参数 ：pDevInf 设备句柄。

pData 缓冲区地址，注意：16 位为 1 个采样数据。

pLen 缓冲区长度。

TimeOut 超时时间，单位：毫秒。

返回值：成功返回 TRUE，失败返回 FALSE。

PARA 函数

CH9130OpenPARA：

打开 PARA 设备。

函数原型：BOOL WINAPI CH9130OpenPARA(PVOID pDevInf)；

参数 ：pDevInf 设备句柄。

返回值：打开成功返回 TRUE，打开失败返回 FALSE。

CH9130InitPARA：

初始化 PARA 设备参数。

函数原型：BOOL WINAPI CH9130InitPARA(PVOID pDevInf,BOOL bSlvMode,BOOL bTwoClk,BOOL bWordWidth,BOOL bSlvEn,UCHAR ucRwCyc,UCHAR ucHldCyc)；

参数 ：pDevInf 设备句柄。

bSlvMode 模式选择,0：外部总线使能，1：被动并口使能。

bTwoClk 外部总线读写周期，0：1 个时钟周期，1：2 个时钟周期。

bWordWidth 外部总线数据位控制域，0：8 位数据总线，1：16 位或 32 位数据总线。

bSlvEn 被动并口流控：0：禁止，1：使能。

ucRwCyc 外部总线的读写总周期。

ucHldCyc 外部总线的读写信号结束点时钟数。

返回值：成功返回 TRUE，失败返回 FALSE。

CH9130ClosePARA：

关闭 PARA 设备。

函数原型：BOOL WINAPI CH9130ClosePARA(PVOID pDevInf)；

参数 ：pDevInf 设备句柄。

返回值：成功返回 TRUE，失败返回 FALSE。

CH9130ReadSlavePARAData：

读被动 PARA 设备数据。

函数原型：BOOL WINAPI CH9130ReadSlavePARAData(PVOID pDevInf,PVOID pData,PULONG pLen,ULONG TimeOut)；

参数 ：pDevInf 设备句柄。

pData 缓冲区地址，注意：16 位为 1 个采样数据。

pLen 函数调用前是缓冲区长度，长度不要超过 512 字节。函数返回后是实际读到的数据长度。

TimeOut 超时时间，单位：毫秒。

返回值：成功返回 TRUE，失败返回 FALSE。

CH9130WriteSlavePARAData：
写被动 PARA 设备数据。
函数原型：BOOL WINAPI CH9130WriteSlavePARAData(PVOID pDevInf,PVOID pData,ULONG uLen,ULONG TimeOut);
参数 ：pDevInf 设备句柄
pData 缓冲区地址
pLen 缓冲区长度
TimeOut 超时时间，单位：毫秒。
返回值：成功返回 TRUE，失败返回 FALSE。
CH9130ReadMasterPARAByte：
按字节方式，读主动 PARA 设备数据。
函数原型：BOOL WINAPI CH9130ReadMasterPARAByte(PVOID pDevInf,UCHAR OffsetAddr,BOOL bAddInc,PVOID pData,ULONG uLen,ULONG TimeOut);
参数 ：pDevInf 设备句柄。
OffsetAddr 外部总线地址。
bAddInc 外部总线地址是否递增。
pData 缓冲区地址，注意：长度不要超过 512 字节。
uLen 缓冲区长度。
TimeOut 超时时间，单位：毫秒。
返回值： 成功返回 TRUE，失败返回 FALSE。
CH9130ReadMasterPARAWord：
按字方式，读主动 PARA 设备数据。
函数原型：BOOL WINAPI CH9130ReadMasterPARAWord(PVOID pDevInf,UCHAR OffsetAddr,BOOL bAddInc,PVOID pData,ULONG uLen,ULONG TimeOut);
参数 ：pDevInf 设备句柄
OffsetAddr 外部总线地址
bAddInc 外部总线地址是否递增
pData 缓冲区地址，注意：长度不要超过 512 字节
uLen 缓冲区长度
TimeOut 超时时间，单位：毫秒。
返回值：成功返回 TRUE，失败返回 FALSE。
CH9130ReadMasterPARADword：
按双字方式，读主动 PARA 设备数据。
函数原型：BOOL WINAPI CH9130ReadMasterPARADword(PVOID pDevInf,UCHAR OffsetAddr,BOOL bAddInc,PVOID pData,ULONG uLen,ULONG TimeOut);
参数 ：pDevInf 设备句柄
OffsetAddr 外部总线地址
bAddInc 外部总线地址是否递增
pData 缓冲区地址，注意：长度不要超过 512 字节
uLen 缓冲区长度
TimeOut 超时时间，单位：毫秒。
返回值：成功返回 TRUE，失败返回 FALSE。
CH9130WriteMasterPARAByte：
按字节方式，写主动 PARA 设备数据。

函数原型：BOOL WINAPI CH9130WriteMasterPARAByte(PVOID pDevInf,UCHAR OffsetAddr,BOOL bAddInc,PVOID pData,ULONG uLen,ULONG TimeOut);

参数 ：pDevInf 设备句柄
OffsetAddr 外部总线地址
bAddInc 外部总线地址是否递增
pData 缓冲区地址，注意：长度不要超过 512 字节
uLen 缓冲区长度
TimeOut 超时时间，单位：毫秒。

返回值：成功返回 TRUE，失败返回 FALSE。

CH9130WriteMasterPARAWord：

按字方式，写主动 PARA 设备数据。

函数原型：BOOL WINAPI CH9130WriteMasterPARAWord(PVOID pDevInf,UCHAR OffsetAddr,BOOL bAddInc,PVOID pData,ULONG uLen,ULONG TimeOut);

参数 ：pDevInf 设备句柄
OffsetAddr 外部总线地址
bAddInc 外部总线地址是否递增
pData 缓冲区地址，注意：长度不要超过 512 字节
uLen 缓冲区长度
TimeOut 超时时间，单位：毫秒。

返回值：成功返回 TRUE，失败返回 FALSE。

CH9130WriteMasterPARADword：

按双字方式，写主动 PARA 设备数据。

函数原型：BOOL WINAPI CH9130WriteMasterPARADword(PVOID pDevInf,UCHAR OffsetAddr,BOOL bAddInc,PVOID pData,ULONG uLen,ULONG TimeOut);

参数 ：pDevInf 设备句柄。
OffsetAddr 外部总线地址。
bAddInc 外部总线地址是否递增。
pData 缓冲区地址，注意：长度不要超过 512 字节。
uLen 缓冲区长度。
TimeOut 超时时间，单位：毫秒。

返回值：成功返回 TRUE，失败返回 FALSE。