

USB2XXX 系列转换芯片用户手册

V0.1C

USBIO TECH.

目 录

USB2XXX 系列转换芯片用户手册	1
1、 驱动安装.....	3
2、 Demo 软件使用说明	8
3、 上位机 Windows API 使用说明	20
4、 使用过程中的注意事项.....	21
5、 常见问题问答	22

1、 驱动安装

1、1、 下载驱动文件

从 USBIO 公司网站 www.usb-i2c-spi.com 的“在线下载”栏里下载最新的驱动程序。连接地址是：<http://www.usb-i2c-spi.com/cn/down.htm>。

对于 USB2ISP、USB2I2C 或 USB2SPI，选择相应的“开发大礼包”下载。解压缩到本地机器的硬盘里，待用。

USB2I2C 和 USB2SPI 是 USB2ISP 的功能简化版本，可以看作是 USB2ISP 的子集。USB2I2C 实现 USB 转换 I2C 接口；USB2SPI 实现 USB 转换 SPI 接口；USB2ISP 实现 USB 转换 SPI 和 I2C 接口，同时还提供 EPP 或 MEM 并口，也可以当成是 GPIO 口来使用，具体请参考 USB2ISP 的数据手册。USB2I2C (SOP28 封装)、USB2SPI 和 USB2ISP 管脚完全兼容，所以 USB2I2C 和 USB2SPI 的相关设计也可以参考 USB2ISP 的数据手册。本手册以 USB2ISP_DEV 开发板为例，说明 USB2XXX 系列转换芯片的功能。

1、2、 插入 USB2ISP_DEV 开发板

将 USB2ISP_DEV 开发板插入到电脑主板 USB 接口。当 USB2ISP_DEV 开发板向外部供电时，最好插入 PC 机背部的主板 USB 口。

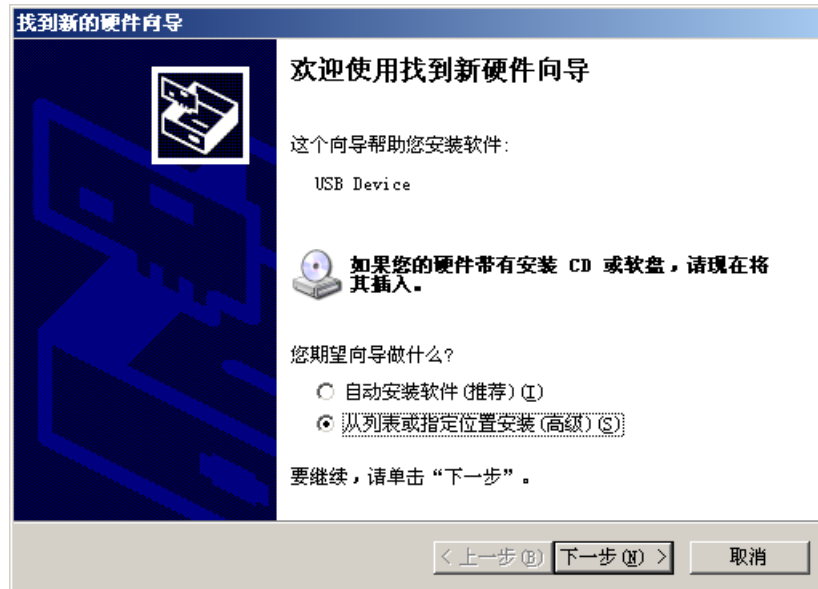
1、3、 Windows 提示发现新硬件



Windows发现了新USB硬件设备

插入 USB2ISP_DEV 开发板后 Windows 提示发现新硬件。

1、4、提示安装驱动

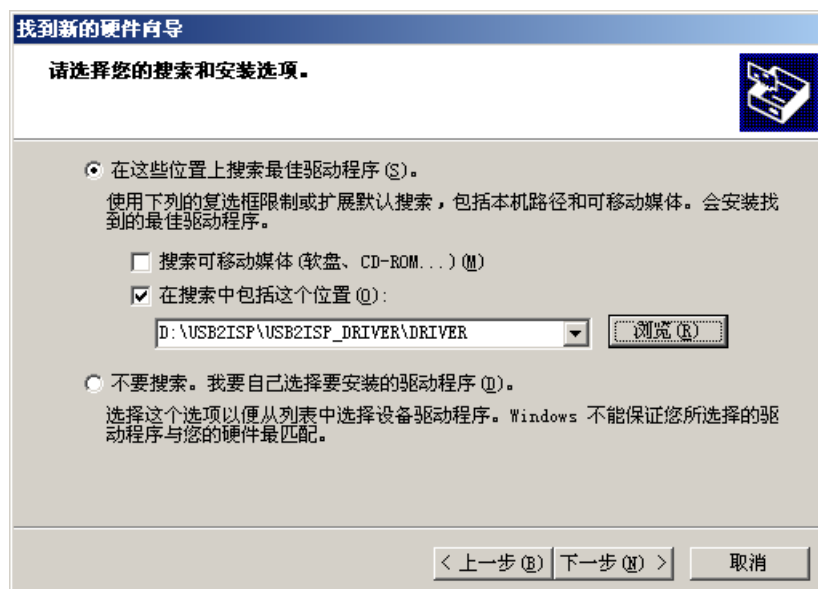


提示安装驱动

选择【从列表或指定位置安装（高级）】选项，然后单击【下一步】按钮。

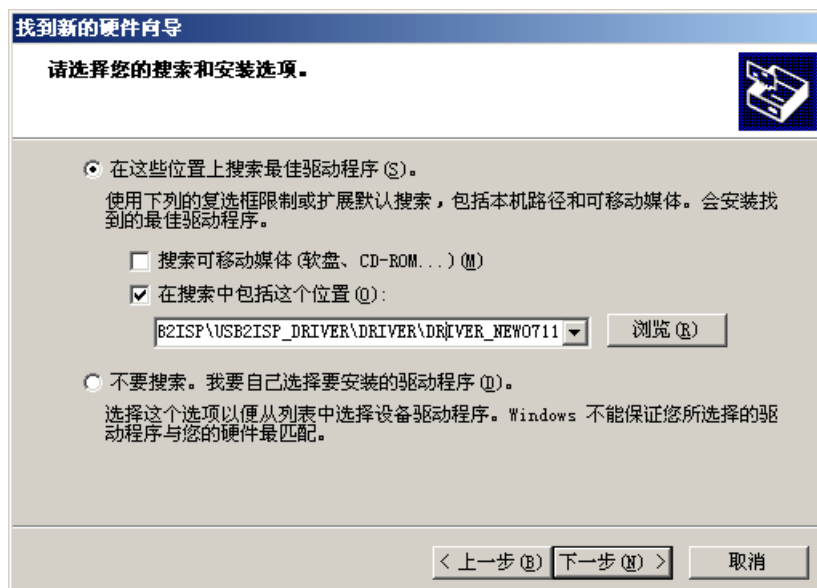
1、5、指定驱动文件的路径

此处需要指定驱动文件的路径。驱动文件就是从网站上下载解压缩后的文件。



注意 :USB2ISP、USB2I2C 或 USB2SPI 芯片 2007 年 11 月之前的产品选择 DRIVER

文件夹内的驱动文件安装。

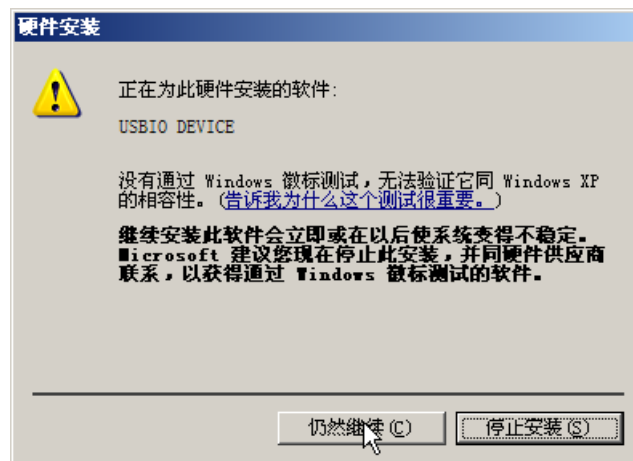


USB2ISP、USB2I2C 或 USB2SPI 芯片 2007 年 11 月以后的产品选择 DRIVER_NEW0711 文件夹内的驱动文件安装。

1、6、复制文件

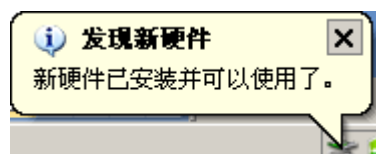


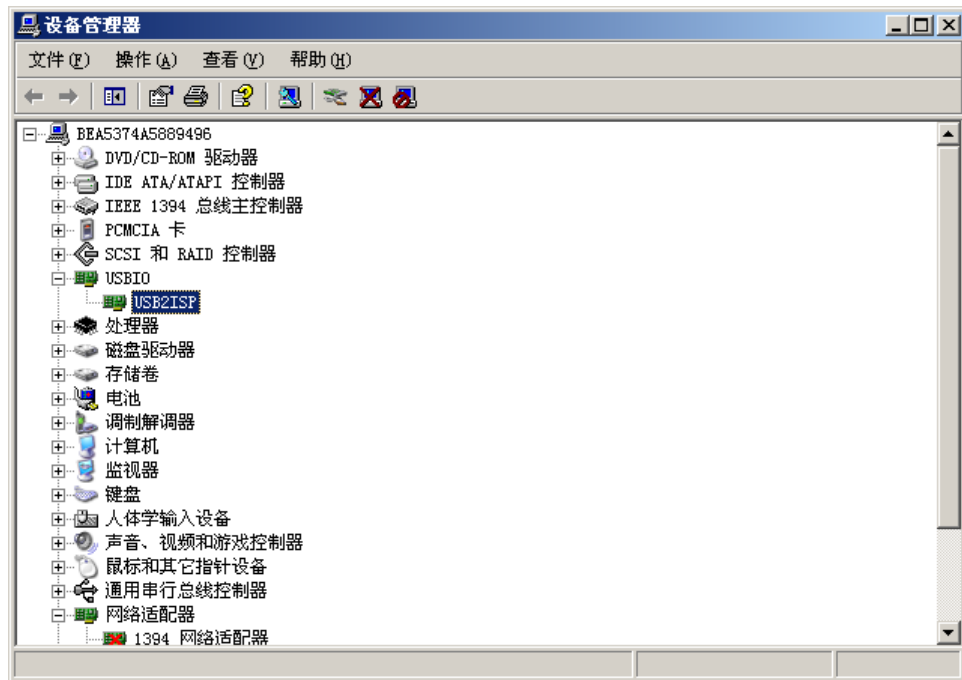
接下来是系统复制驱动的过程。首次安装可能还会提示“没有通过 Windows 徽标测试”，选择【仍然继续】按钮。



微软徽标认证

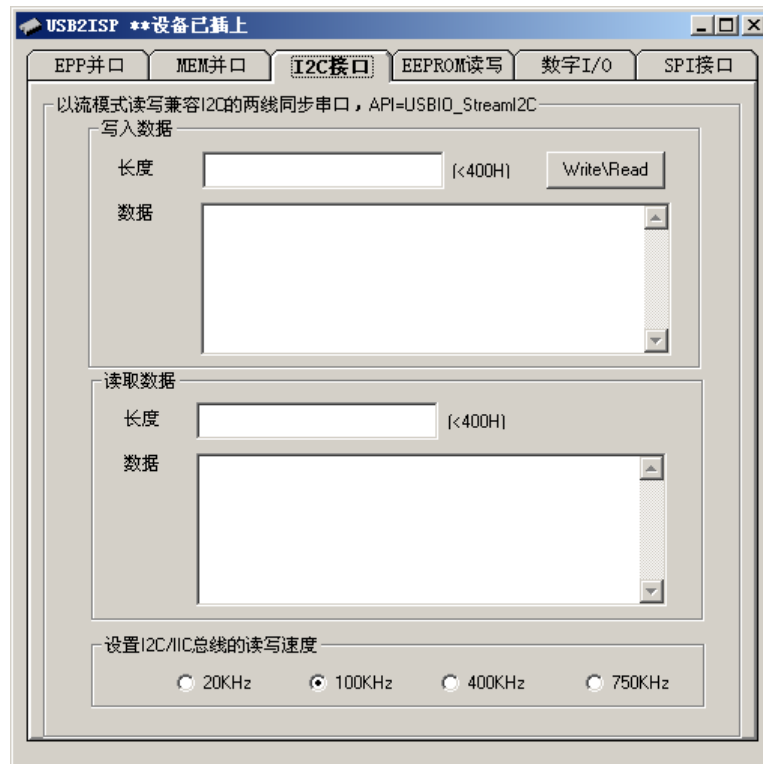
1、7、安装成功





可以通过我的【电脑】→【属性】→【硬件设备管理器】来查看新安装的设备。

也可以打开 USB2ISP_DEMO_VBCN.exe，此时状态来显示



设备已插上

至此，驱动安装完成，可以进一步的测试工作了。

2、 DEMO 软件使用说明

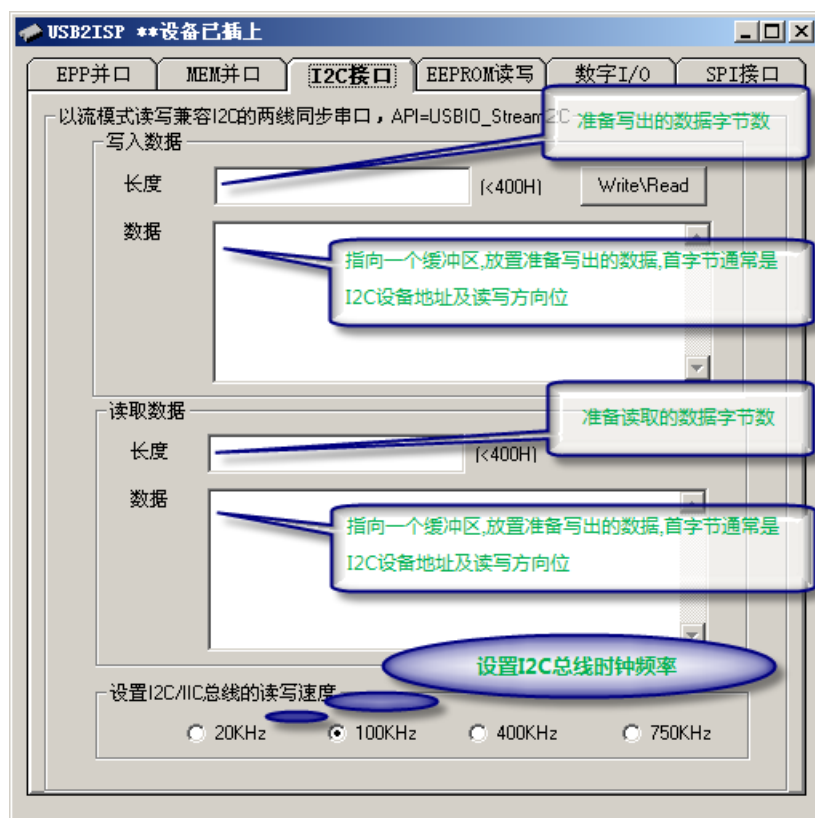
USB2I2C 是 USB 转换 I2C 总线 (兼容 TWI 和 SMBus 总线) 专用芯片 ; USB2SPI 是 USB 转换 SPI 总线专用芯片 ; USB2ISP 芯片是 USB2I2C 和 USB2SPI 的合集 , 除了实现 I2C 和 SPI 总线外 , 还实现了 EPP 并口和 MEM 并口。下面以 USB2ISP 说明 USB2ISP_DEMO 软件的使用 , 涉及到 I2C 和 SPI 方面的内容完全适用于 USB2I2C 和 USB2SPI。

2、 1、 I2C 总线功能说明

I2C 接口选项卡 , 调用的是 USB2ISP 驱动 USBIOX.DLL 文件中的 USBIO_StreamI2C API , 关于 USBIO_StreamI2C 的详细说明见 USBIOX.H 文件。为方便说明摘录在下面 :

```
USBIO_StreamI2C( // 处理 I2C 数据流
    ULONG iIndex, // 指定 USB2ISP 设备序号
    ULONG iWriteLength, // 准备写出的数据字节数
    PVOID iWriteBuffer, // 指向一个缓冲区,放置准备写出的数据,
                        //第 1 个字节通常是 I2C 设备地址及读写方向位
    ULONG iReadLength, // 准备读取的数据字节数
    PVOID oReadBuffer ); // 指向一个缓冲区,返回后是读入的数据

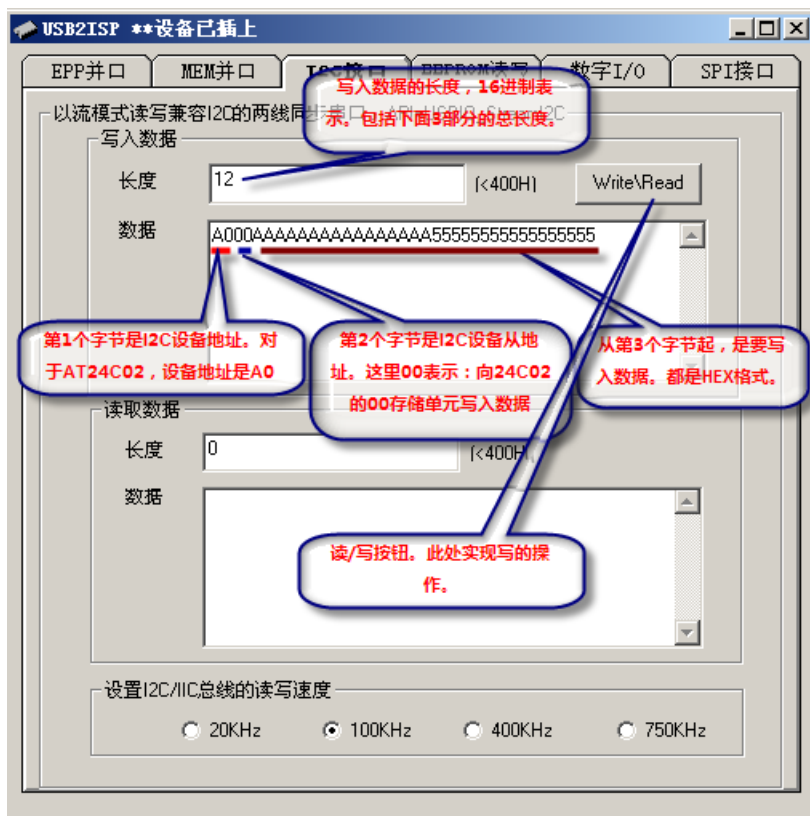
USBIO_SetStream( // 设置串口流模式
    ULONG iIndex, // 指定 USB2ISP 设备序号
    ULONG iMode ); // 指定模式,见下的说明
iMode 说明: 位 1-位 0: I2C 接口速度/SCL 频率, 00=低速/20KHz,01=标准/100KHz(默认值),10=快速/400KHz,11=高速/750KHz。位 2: SPI 的 I/O 引脚数, 0=单入单出(SCK 时钟/MOSI 出/MISO 入)(默认值),1=双入双出(SCK 时钟/MOSI 出 MSOSI2 出/MISO 入 MISO2 入), 位 7: SPI 字节中的位顺序, 0=低位在前, 1=高位在前。 其它保留,必须为 0。
```

I2C接口选项卡

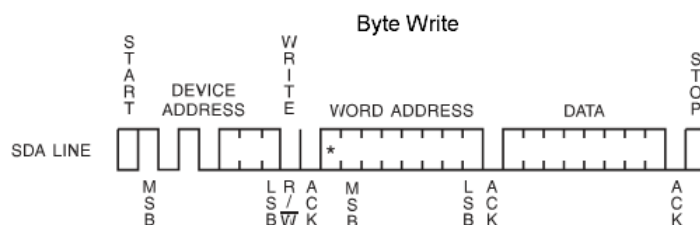
下面以调用 USBIO_StreamI2C 读写 AT24C02 EEPROM 为例，说明如何使用 USBIO_StreamI2C 实现各类灵活的 I2C 操作。

2.1.1、用 USBIO_StreamI2C 实现向 AT24C02 的 00 存储单元写数据。如下图所示。

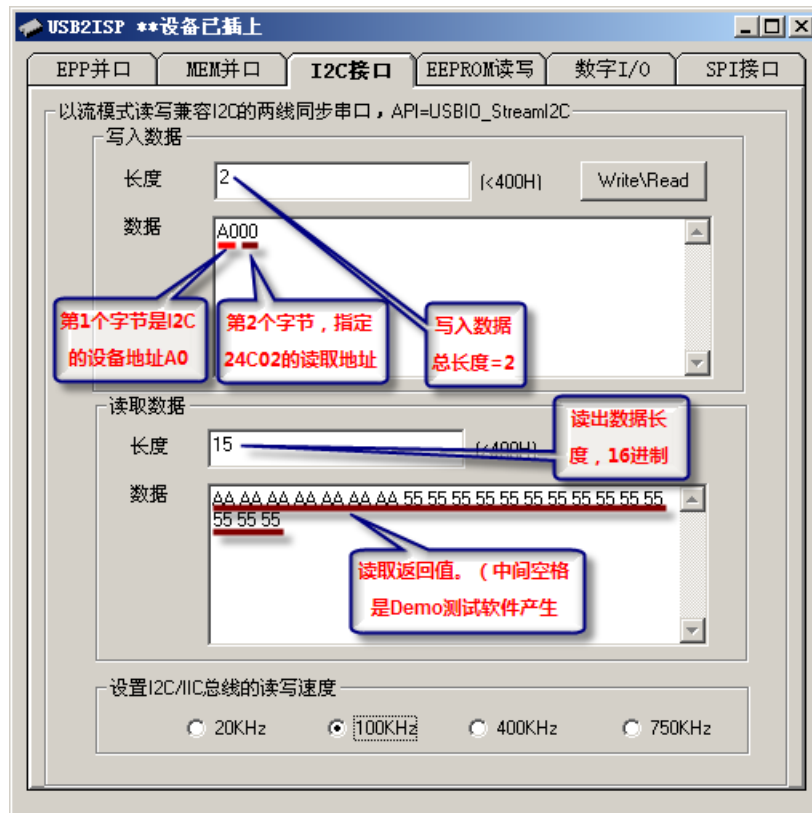


向AT24C02的00存储单元写数据

时序参考：

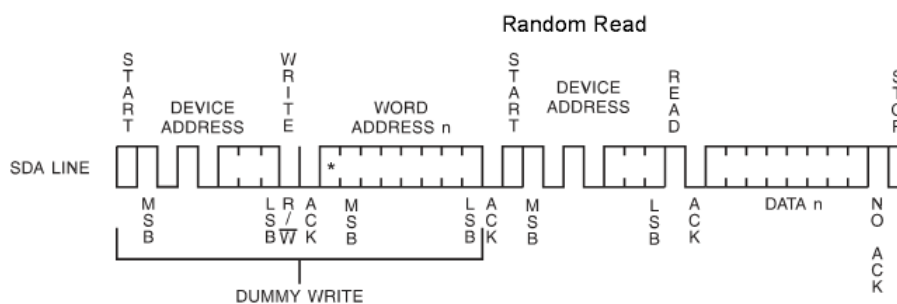


2.1.2、用 USBIO_StreamI2C 实现从 AT24C02 的 00 存储单元读取数据。如下图所示。



从AT24C02的00位置读取数据

时序参考：



2、2、EEPROM 读写说明

读写 I2C 接口的 EEPROM 还可以通过调用专用 API 来实现：

```

USBIO_ReadEEPROM( // 从 EEPROM 中读取数据块
    ULONG          iIndex, // 指定 USB2ISP 设备序号
    EEPROM_TYPE     iEepromID, // 指定 EEPROM 型号
    ULONG          iAddr, // 指定数据单元的地址

```

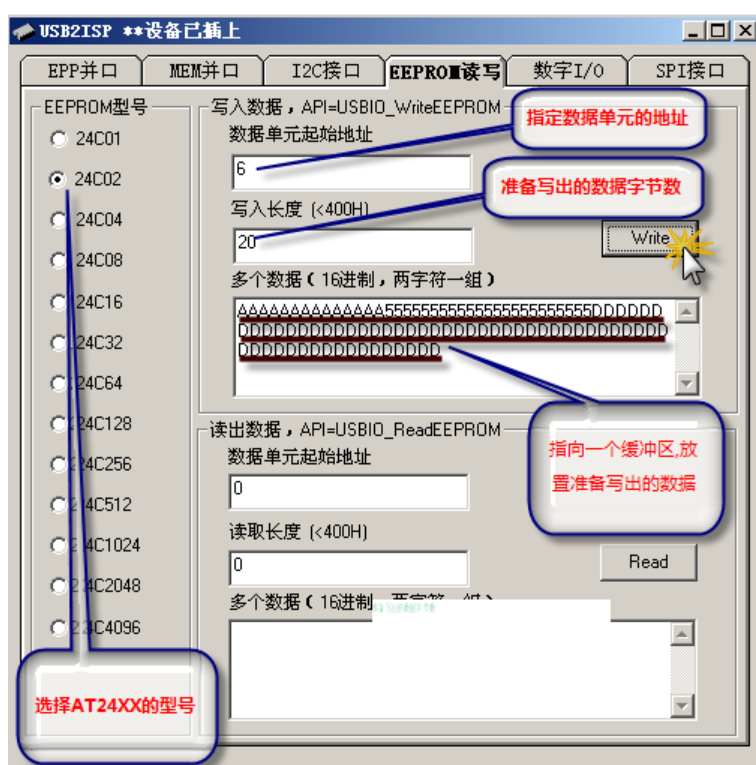
```

ULONG          iLength, // 准备读取的数据字节数
PUCHAR         oBuffer ); // 指向一个缓冲区,返回后是读入的数据
USBIO_WriteEEPROM( // 向 EEPROM 中写入数据块
ULONG          iIndex, // 指定 USB2ISP 设备序号
EEPROM_TYPE     iEepromID, // 指定 EEPROM 型号
ULONG          iAddr, // 指定数据单元的地址
ULONG          iLength, // 准备写出的数据字节数
PUCHAR         iBuffer ); // 指向一个缓冲区,放置准备写出的数据

```

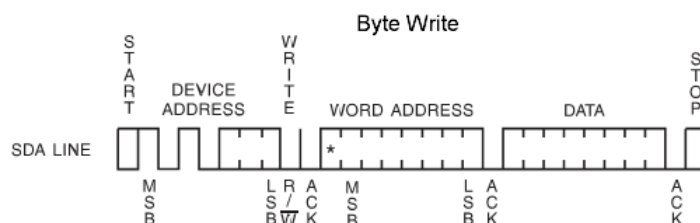
下面以读写 AT24C02 为例说明：

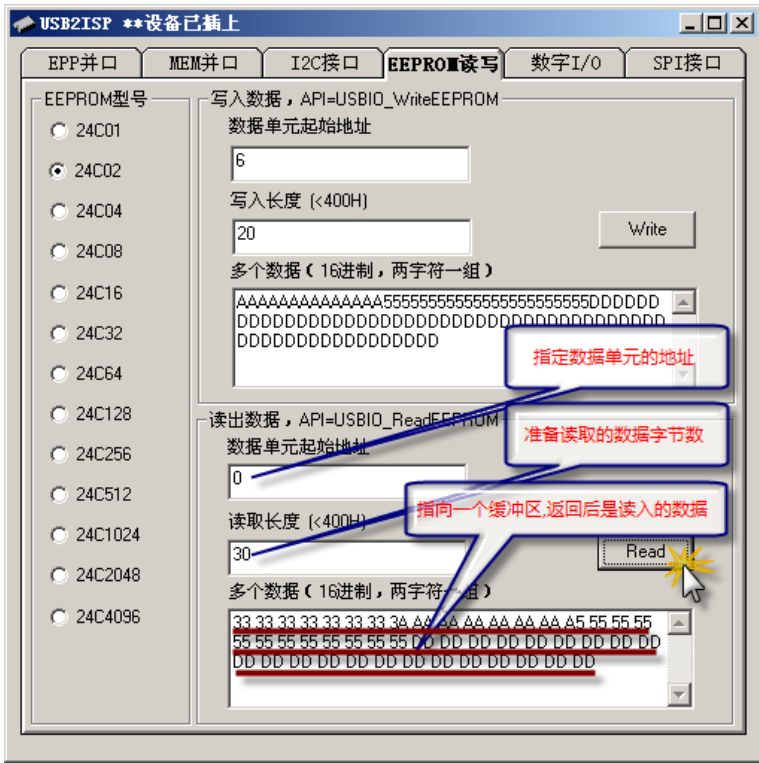
2.2.1、向 AT24C02 的 06 存储单元写入 2 个 0 数据。



向AT24C02的06存储单元写入20个数据

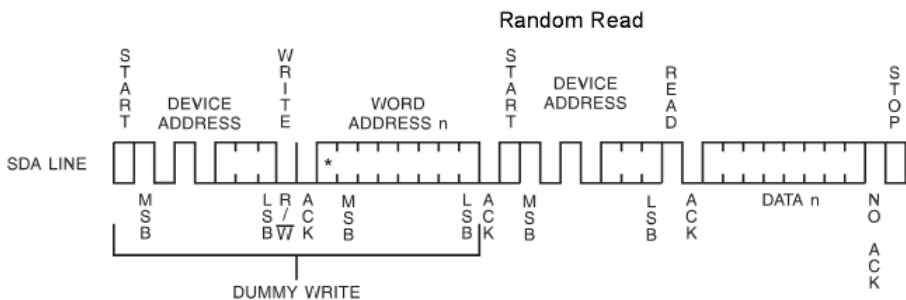
时序参考：



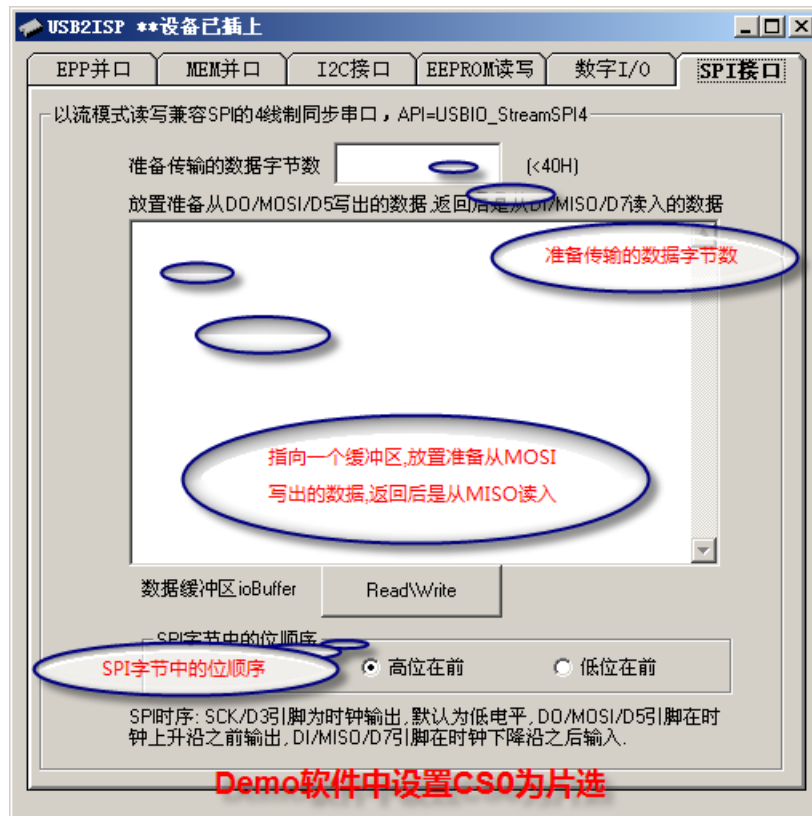


从AT24C02中地址00处读取30个数据

时序参考：



2、3、SPI 总线功能说明



SPI接口选项卡

```

USBIO_StreamSPI4 ( // 处理 SPI 数据流, 支持 SPI Mode0/1
    ULONG iIndex, // 指定 USB2ISP 设备序号
    ULONG iChipSelect, // 片选控制, 位 7 为 0 则忽略片选控制, 位 7 为 1 则参数有效:
                        // 位 1 位 0 为 00/01/10 分别选择 D0/D1/D2 引脚作为低电平
                        // 有效片选
    ULONG iLength, // 准备传输的数据字节数
    PVOID ioBuffer ); // 指向一个缓冲区, 放置准备从 MOSI 写出的数据,
                        // 返回后是从 MISO 读入的数据

```

```

USBIO_SetStream( // 设置串口流模式
    ULONG iIndex, // 指定 USB2ISP 设备序号
    ULONG iMode ); // 指定模式, 见下的说明

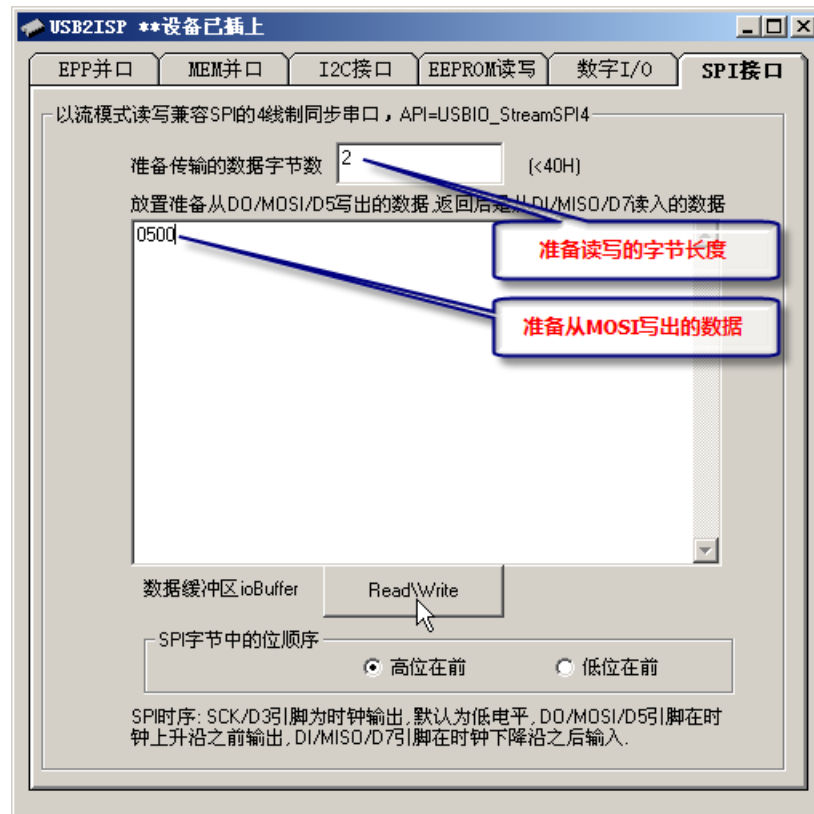
```

iMode 说明: 位 1-位 0: I2C 接口速度/SCL 频率, 00=低速/20KHz, 01=标准/100KHz(默认值), 10=快速/400KHz, 11=高速/750KHz。位 2: SPI 的 I/O 引脚数, 0=单入单出(SCK 时钟/MOSI 出/MISO 入)(默认值), 1=双入双出(SCK 时钟/MOSI 出 MSOSI2 出/MISO 入 MISO2 入), 位 7: SPI 字节中的位顺序, 0=低位在前, 1=高位在前。其它保留, 必须为 0。

下面举例说明 USBIO_StreamSPI4 的使用方法。

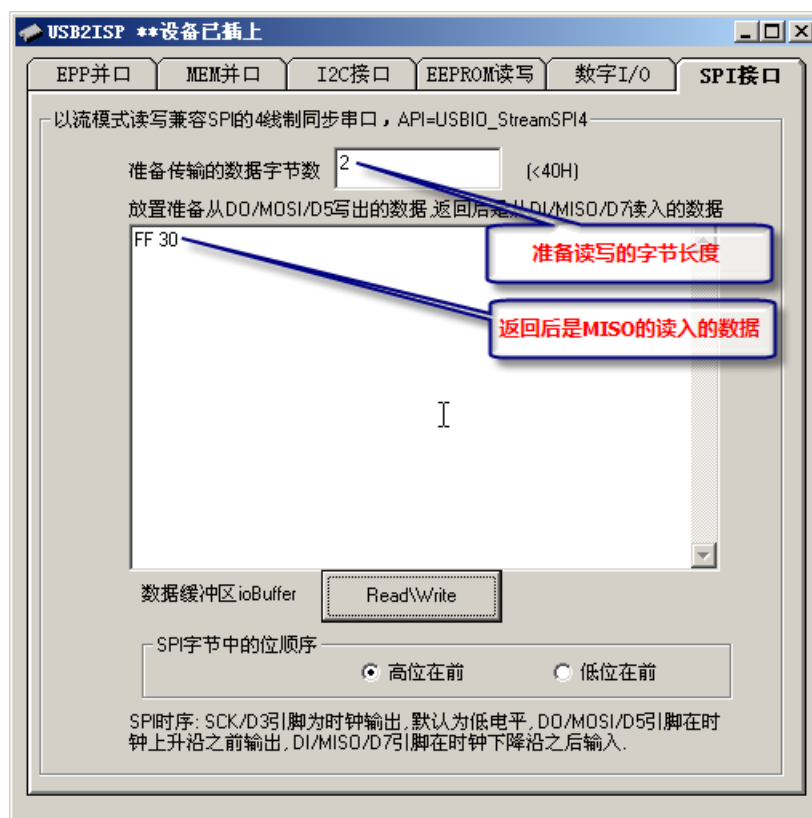
A、用 USBIO_StreamSPI4 读写 X5045

读 X5045 的状态寄存器，命令码为：05（Hex），00（Hex，实际上这个字节可以任意填充，只是为了产生必要的 SCK 时钟）。



读X5045的状态寄存器

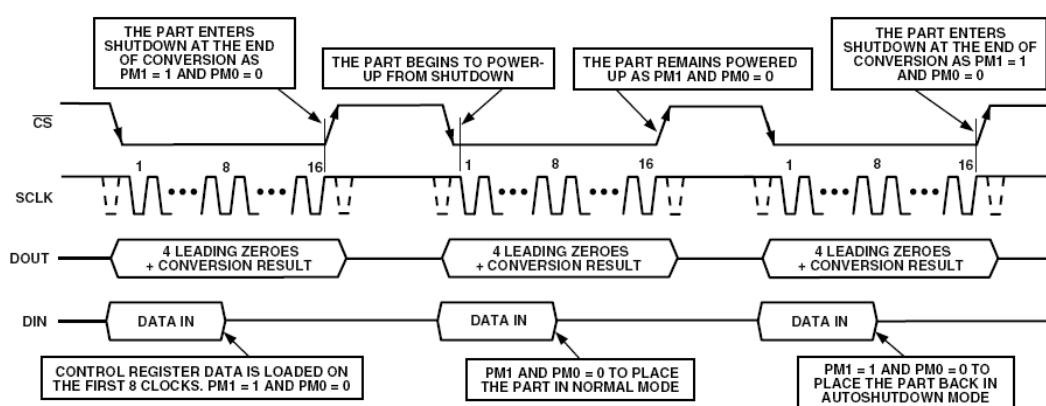
读 X5045 的状态寄存器的返回数据。



读X5045的状态寄存器

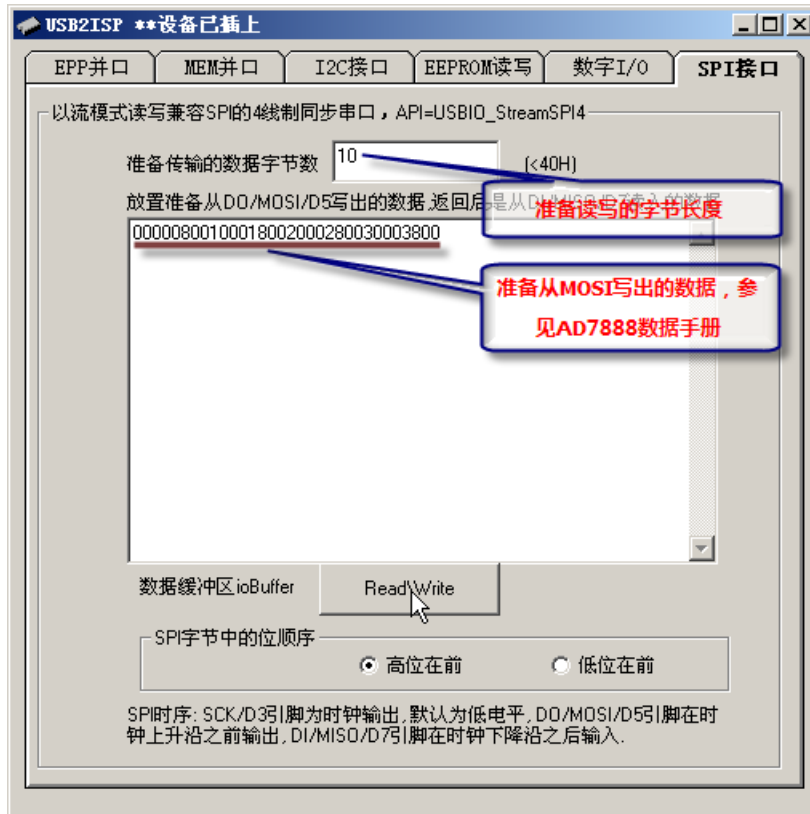
B、用 USBIO_StreamSPI4 读写 AD7888

下图是 AD7888 的读写时序，具体操作请参考 AD7888 的数据手册。

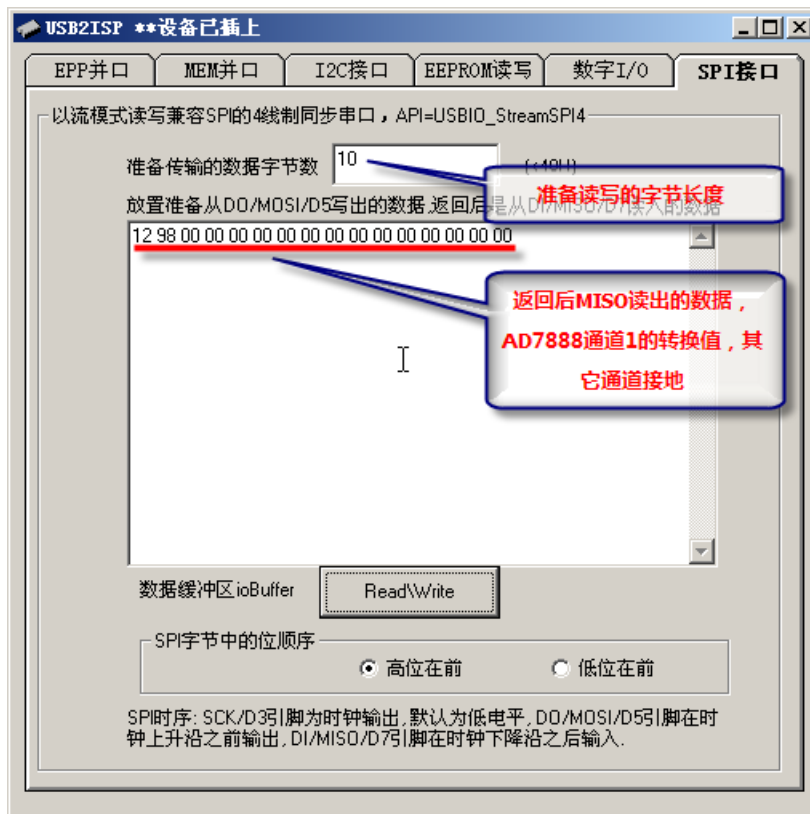


AD7888读写时序

USB2ISP 调用 USBIO_StreamSPI4 读 AD7888 的数据。



读写AD7888



读写AD7888返回值

2、4、EPP 并口操作说明

```
USBIO_EppReadData( // EPP 方式读数据: WR#=1, DS#=0, AS#=1, D0-D7=input
    ULONG iIndex, // 指定 USB2ISP 设备序号
    PVOID oBuffer, // 指向一个足够大的缓冲区,用于保存读取的数据
    PULONG ioLength ); // 指向长度单元,输入时为准备读取的长度,
                        // 返回后为实际读取的长度

USBIO_EppReadAddr( // EPP 方式读地址: WR#=1, DS#=1, AS#=0, D0-D7=input
    ULONG iIndex, // 指定 USB2ISP 设备序号
    PVOID oBuffer, // 指向一个足够大的缓冲区,用于保存读取的地址数据
    PULONG ioLength ); // 指向长度单元,输入时为准备读取的长度,
                        // 返回后为实际读取的长度

USBIO_EppWriteData( // EPP 方式写数据: WR#=0, DS#=0, AS#=1, D0-D7=output
    ULONG iIndex, // 指定 USB2ISP 设备序号
    PVOID iBuffer, // 指向一个缓冲区,放置准备写出的数据
    PULONG ioLength ); // 指向长度单元,输入时为准备写出的长度,
                        // 返回后为实际写出的长度

USBIO_EppWriteAddr( // EPP 方式写地址: WR#=0, DS#=1, AS#=0, D0-D7=output
    ULONG iIndex, // 指定 USB2ISP 设备序号
    PVOID iBuffer, // 指向一个缓冲区,放置准备写出的地址数据
    PULONG ioLength ); // 指向长度单元,输入时为准备写出的长度,
                        // 返回后为实际写出的长度

USBIO_EppSetAddr( // EPP 方式设置地址: WR#=0, DS#=1, AS#=0, D0-D7=output
    ULONG iIndex, // 指定 USB2ISP 设备序号
    UCHAR iAddr ); // 指定 EPP 地址
```

2、5、MEM 并口操作说明

```
USBIO_MemReadAddr0( // MEM 方式读地址 0: WR#=1, DS#/RD#=0, AS#/ADDR=0,
    ULONG iIndex, // 指定 USB2ISP 设备序号
    PVOID oBuffer, // 指向一个足够大的缓冲区,用于保存从地址 0 读取的数据
    PULONG ioLength ); // 指向长度单元,输入时为准备读取的长度,
                        // 返回后为实际读取的长度

USBIO_MemReadAddr1( // MEM 方式读地址 1: WR#=1, DS#/RD#=0, AS#/ADDR=1,
    ULONG iIndex, // 指定 USB2ISP 设备序号
    PVOID oBuffer, // 指向一个足够大的缓冲区,用于保存从地址 1 读取的数据
    PULONG ioLength ); // 指向长度单元,输入时为准备读取的长度,
                        // 返回后为实际读取的长度
```

```
USBIO_MemWriteAddr0( // MEM 方式写地址 0: WR#=0, DS#/RD#=1, AS#/ADDR=0,
    ULONG iIndex, // 指定 USB2ISP 设备序号
    PVOID iBuffer, // 指向一个缓冲区,放置准备向地址 0 写出的数据
    PULONG ioLength); // 指向长度单元,输入时为准备写出的长度,
                        // 返回后为实际写出的长度
```

```
USBIO_MemWriteAddr1( // MEM 方式写地址 1: WR#=0, DS#/RD#=1, AS#/ADDR=1,
    ULONG iIndex, // 指定 USB2ISP 设备序号
    PVOID iBuffer, // 指向一个缓冲区,放置准备向地址 1 写出的数据
    PULONG ioLength); // 指向长度单元,输入时为准备写出的长度,
                        // 返回后为实际写出的长度
```

2、6、GPIO 引脚说明

USB2ISP 的 D5-D0 引脚可以当作是 GPIO 来使用。

```
USBIO_Set_D5_D0( // 设置 USB2ISP 的 D5-D0 引脚的 I/O 方向
    ULONG iIndex, // 指定 USB2ISP 设备序号
    ULONG iSetDirOut, // 设置 D5-D0 各引脚的 I/O 方向,
                        //某位清 0 则对应引脚为输入,某位置 1 则对应引脚为输出,
                        //并口方式下默认值为 0x00 全部输入
    ULONG iSetDataOut); // 设置 D5-D0 各引脚的输出数据,
                        //如果 I/O 方向为输出,那么某位清 0 时对应引脚输出低电平,
                        //某位置 1 时对应引脚输出高电平
// 以上数据的位 5-位 0 分别对应 USB2ISP 的 D5-D0 引脚
/* ***** 谨慎使用该 API, 防止修改 I/O 方向使输入引脚变为输出引脚导致与其它输出引脚之间短路而损坏芯片 ***** */
```

3、上位机 WINDOWS API 使用说明

上位机 API 使用说明详见 USBIOX.H 文件里的说明。

4、使用过程中的注意事项

暂无说明

5、常见问题问答

5.1、Q：USB2I2C，USB2SPI 和 USB2ISP 有什么不同？

A：USB2I2C 是 USB 转换 I2C 总线（兼容 TWI 和 SMBus 总线）专用芯片；USB2SPI 是 USB 转换 SPI 总线专用芯片；USB2ISP 芯片是 USB2I2C 和 USB2SPI 的合集，除了实现 I2C 和 SPI 总线外，还实现了 EPP 并口和 MEM 并口。

5.2、Q：USB2I2C，USB2SPI 和 USB2ISP 需要自己写固件吗？

A：USB2I2C，USB2SPI 和 USB2ISP 是专用芯片，不需要编写任何固件。上尉机软件调用 USBIOX.DLL 动态连接库里的专用 API 即可。

5.3、Q：USB2I2C 和 USB2ISP 转换的 I2C 接口都可以读写那些 I2C 器件？

A：USB2I2C 和 USB2ISP 转换的 I2C 总线，是标准的 I2C 总线，任何符合 I2C 协议标准的器件都可以读写。

5.4、Q：USB2I2C 和 USB2ISP 转换的 I2C 接口支持高速传输吗？

A：USB2I2C 和 USB2ISP 转换的 I2C 接口的时钟频率支持：50KHz，100KHz，400KHz 和 750KHz。750KHz 属于高速模式。

5.5、Q：USB2I2C 和 USB2ISP 支持特殊协议的 I2C 接口器件吗？

A：请参考 TEST 目录下的 EXAM_API.C 文件。

5.6、Q：USB2I2C 和 USB2ISP 支持 10bit 地址的 I2C 接口器件吗？

A：目前还不支持。

5.7、Q : USB2I2C 和 USB2ISP 可以作为 I2C 从设备吗？

A : 不支持。USB2I2C 和 USB2ISP 只能作为 I2C 主设备。