

文档名称	USB 转接板(多功能型)用户手册
文档编号	USB2ISH_AD00R3
文档版本	3.00

USB 转接板用户手册

(多功能版)

拟制: Tommy . zhong 日期: 2013 年 11 月 16 日

评审: Antony 日期: 2013 年 11 月 26 日

批准: _____ 日期: _____ 年 _____ 月 _____ 日

签发: _____ 日期: _____ 年 _____ 月 _____ 日



OnEasyB Technologies Co . , Ltd.

版权所有，侵权必究

USB 转接板(多功能版)用户手册

修改记录

日期	版本	章节	修改描述	作者
2013-11-26	R2. 00		创建完成	<u>Tommy . zhong</u> Antony
2013-12-10	R2. 01	第三章	端口表 J5 标识错	Antony
2014-02-25	R2. 02	第三章	MISO 引脚标识错	Antony
2014-07-15	R3. 00		功能优化	Antony

USB 转接板用户手册	1
一、 简介	4
二、 外观	5
三、 接口布局位置及端口定义说明	7
四、 附带文件说明	9
五、 驱动程序安装	10
5.1 安装 USB 转 SPI 和 I2C 驱动	10
5.2 安装 USB 转 UART 驱动程序	19
六、 使用简介	20
6.1> 作为 UART 口使用	20
6.1.1> window 下使用:	20
6.2.2> linux 下使用:	20
6.2 作为 I2c 使用	21
6.2.1> 选择设备	23
6.2.2> 设置	23
6.2.3> 操作选择	24
6.2.4> 读写操作	24
A. 立即读模式	24
B. 复合读模式	25
C. 写模式	26
6.2.5> 特殊功能(批量写)	28
6.3 作为 SPI 使用	29
6.3.1> 作为主设备使用	30
A. 设置	30
B. 操作选择	30
C. 读写操作	31
D. 特殊功能	34
6.3.2> 作为从设备使用	34
6.4> 作为 HDQ 使用	35
6.5> 作为 GPIO 使用	36
6.5.1> 数字 IO 和 ADC 采样端接口定义	38
6.5.2> ADC 使用说明	38
6.5.3> 数字 IO 的使用	39
6.5.4> PWM 使用	39
6.6> 补充说明	40
6.7> 固件更新	42
七、 常见问题及解决方法	44
八、 维护	45

一、 简介

USB2UIS 转接板是一款实用方便的多功能转换板，它集成了 USB 转 UART，USB 转 I2C，USB 转 SPI 接口，USB 转 ADC，USB 转 PWM，USB 转 GPIO 接口。通过上位机软件很方便地同带有 UART，I2C，SPI 等接口的设备通讯，附带可控制的 ADC 采样，PWM 和 GPIO 输出。

系统特征:

- 1> 可选的 3.3V 和 5V 输出电压。
- 2> 自恢复熔丝，防止过流。
- 3> USB2.0 全速通讯速率。
- 4> 附带演示软件供参考使用,还增加了 nordic 的 RF2401 专用测试应用程序。
- 5> 三种读写模式(手动，定时和中断)。
- 6> 提供 DLL 接口供第三方开发。
- 7> 唯一的设备序列号，可同时打开多个设备。
- 8> 支持 win XP，vista，winodw7 和 window8(32bit,64bit)。
- 9> 支持 linux 下的 USB 转 UART 口应用
- 10> 可通过电脑 USB 线更新固件，软件升级，解决维护烦恼,对于特殊应用，还可定制。

接口特征:

- 1> UART 波特率最大可达 230400
- 2> I2C 波特率最大可设置 800k，支持带中断的读写
- 3> SPI 波特率最大可设置 12M，支持带中断的读写及扩展 CE 的控制,支持从模式
- 4> 10bit ADC 采样精度及 200K 的采样频率，四个通道可以配置为单端和差分输入
- 5> 8 个数字 IO 可任意配置的输入和输出，一个扩展 IO 输出和一个中断 IO 输入
- 6> 四路 PWM 同频输出，频率最大可达 100K 占空比任意可调。

用途:

设备调试，生产测试。无线模块控制，LED 控制等。

二、外观

两种外观

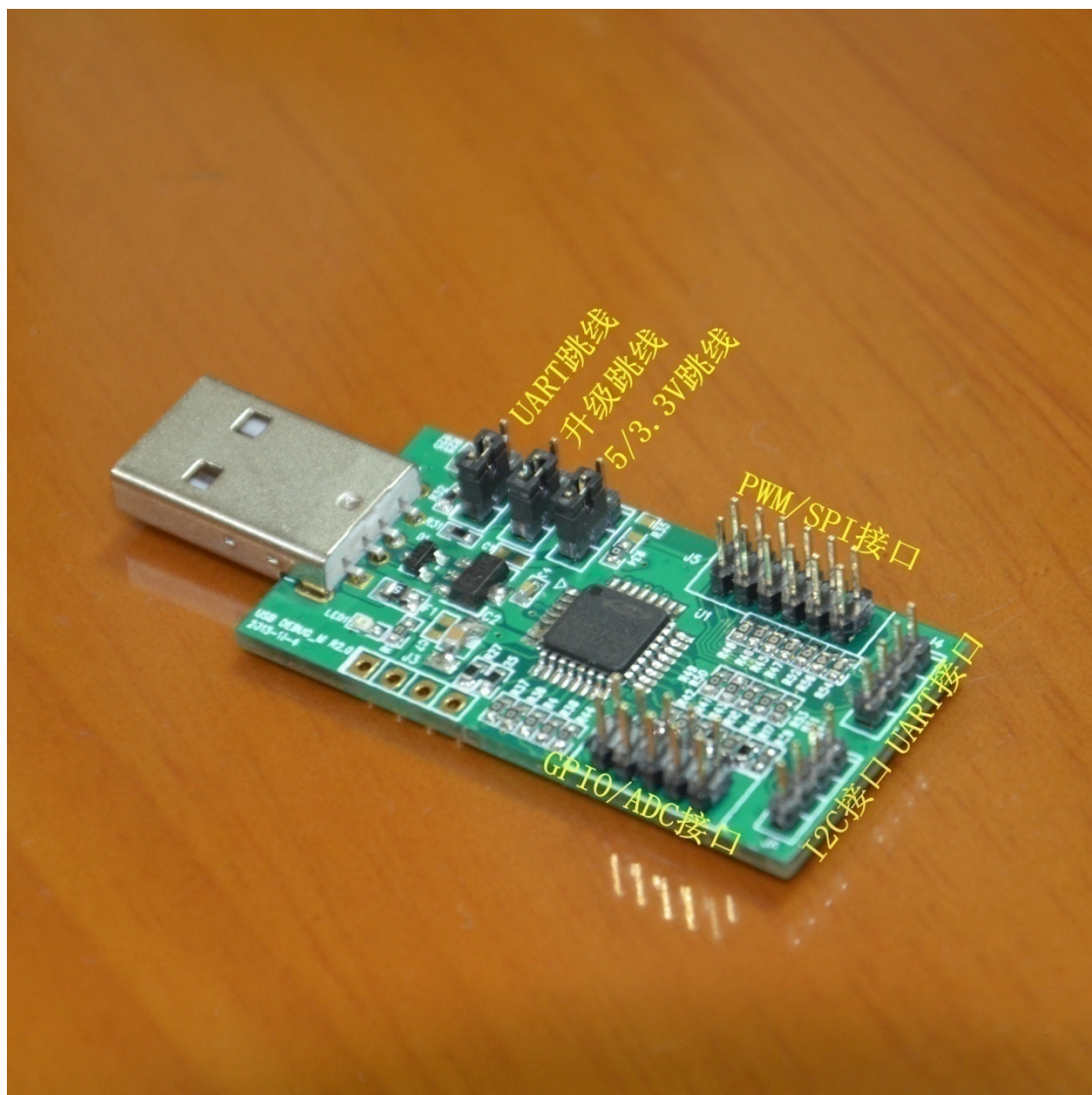
图左白色的是两个带卡套的连接座.

图右全是 2.0mm 间距的排针



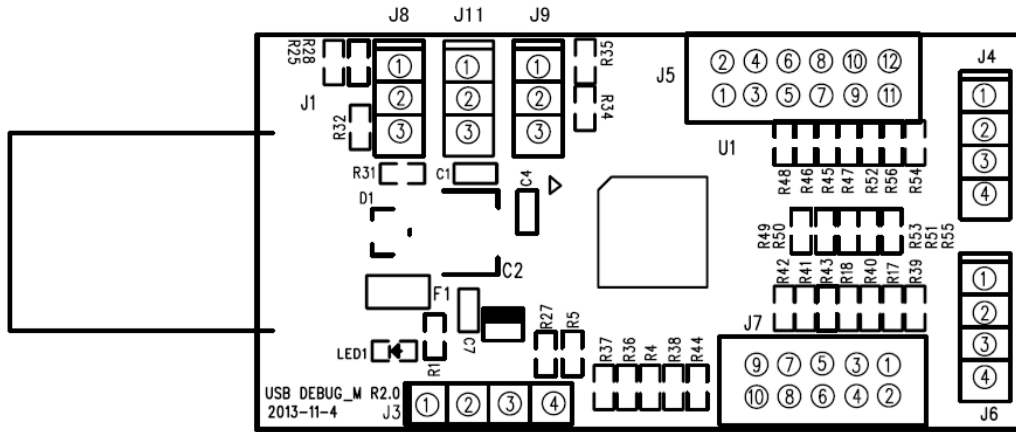
下面是每个接口的说明



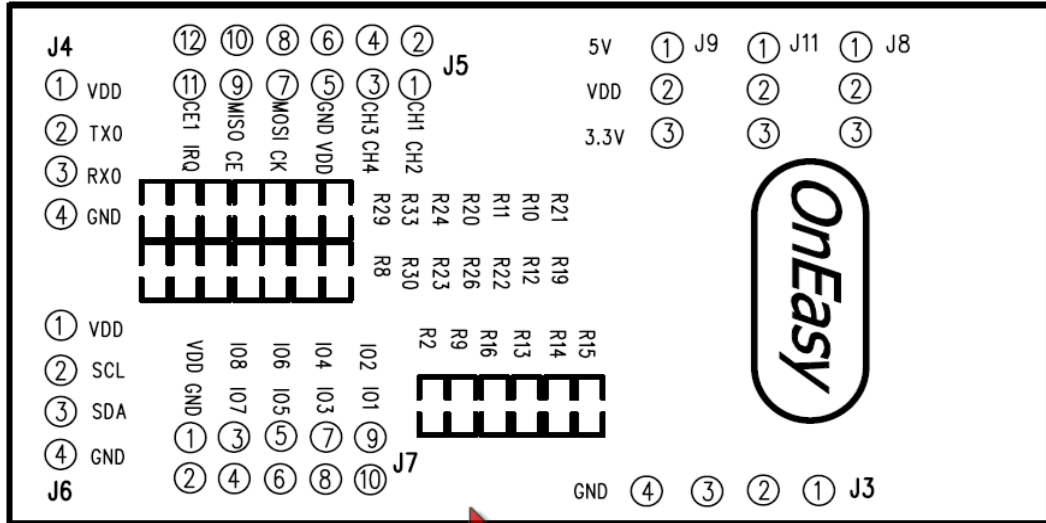


三、 接口布局位置及端口定义说明

接口布局如下图



正面布局图



背面丝印图

端口说明见下表

接口 编号	功能	引脚定义					
		引脚定义	丝印	软件功能	引脚定义	丝印	软件功能
J1	USB						
J3	未用						
J4	UART	J4-1	VDD	电源正	J4-2	TXD	发送端
		J4-3	RXD	接收端	J4-4	GND	电源地
J5	PWM	J5-1	CH2	二号通道	J5-2	CH1	一号通道
		J5-3	CH3	四号通道	J5-4	CH4	三号通道
	SPI	J5-5	VDD	电源正	J5-6	GND	电源地
		J5-7	CK	时钟端	J5-8	MOSI	主出从入端
		J5-9	CE	片选端	J5-10	MISO	主入从出端
		J5-11	IRQ	IO 中断端	J5-12	CE1	扩展片选
J6	I2C	J6-1	VDD	电源正	J6-2	SCL	时钟端
		J6-3	SDA	数据端	J6-4	GND	电源地
J7	GPIO	J7-1	VDD	电源正	J7-2	GND	电源地
		J7-3	IO8	Bit0	J7-4	IO7	Bit1
		J7-5	IO6	Bit2	J7-6	IO5	Bit3
		J7-7	IO4	Bit4	J7-8	IO3	Bit5
		J7-9	IO2	Bit6	J7-10	IO1	Bit7
J8	输出功 能选择	1,2 短接,UART 功能允许; 2,3 短接,IIC,SPI,HDQ,GPIO 功能允许					
J9	输出电 压选择	1,2 短接,输出 VDD 为 5V; 2,3 短接,输出 VDD 为 3.3V					
J11	固件更 新选择	1,2 短接,作为固件更新; 2,3 短接,由 J8 决定功能					

四、 附带文件说明

附带文件说明见下表

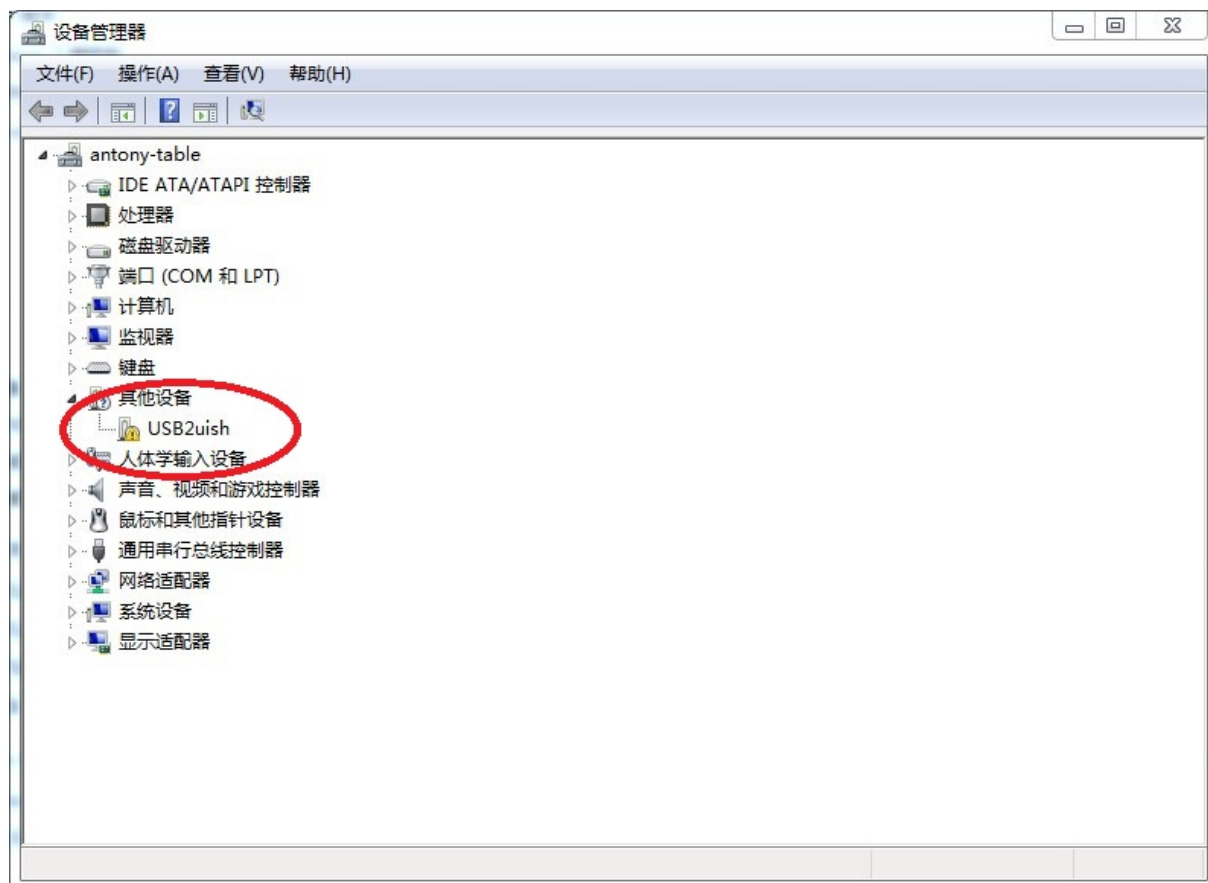
文件夹	内容	运行系统	用 途
APP	Usb2ish_pro.exe, RF_RW.exe	Window 系 统	Usb2ish_pro.exe 是 USB 转 I2C, SPI, HDQ, ADC, PWM 和 GPIO 的读写程序; RF_RW 是 nRF2401 模块读写程序。
DLL	usb2uis.dll, usb2uis.lib,usbio.h	Window 系 统	库文件和头文件,供第三方开发调用
Doc	USB2uis 转接板用户手册(多功能版)R30.pdf	不限	操作说明书
	DLL 函数说明.pdf		DLL 函数说明
	RF_RW_R3 使用说明		连接 nRF2401 模块使用说明
Driver	Uart 和 ish	Window 系 统	Ish 是 Usb 转 I2C,HDQ,SPI 和 GPIO 驱动程序文件夹,UART 是 Usb 转 UART 驱动程序文件夹
Firmware	USB2ISH_FM_M00_XX.bin	Window 系 统	固件升级目标文件
VC_DEMO	SPI_RW 项目文件	Window 系 统	通过 SPI 读写的 VC 参考例程
	I2C_RW 项目文件		通过 I2C 读写的 VC 参考例程
PCcomAPI	pcommtest.exe, comm.dll	Window 系 统	USB 转 UART 口测试程序

五、 驱动程序安装

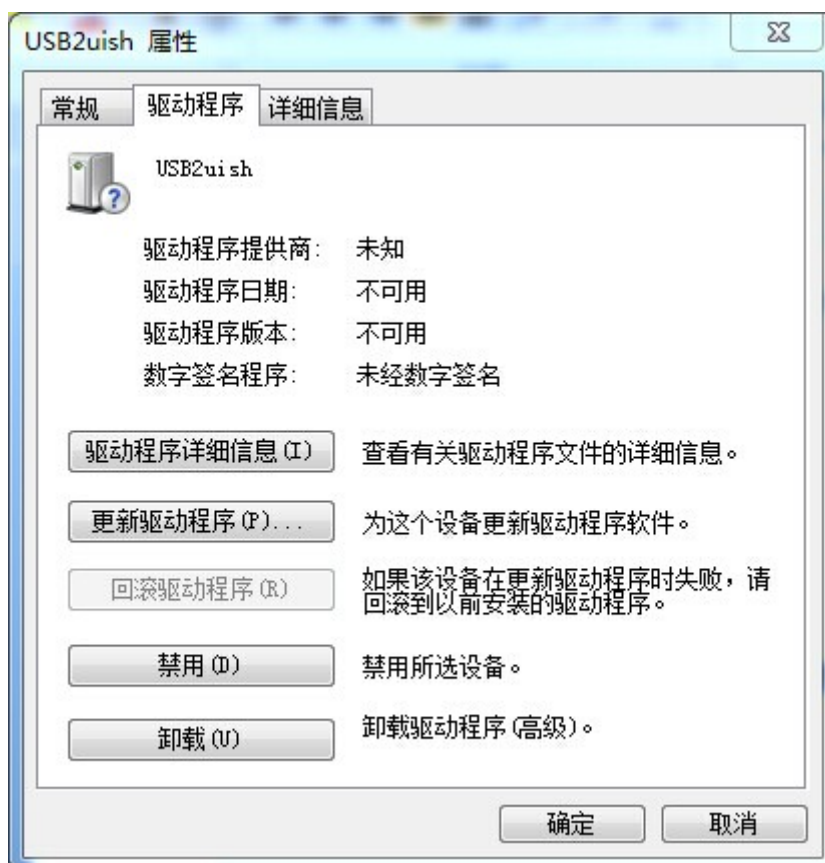
下面以 window7 为例说明驱动程序的安装

5.1 安装 USB 转 SPI 和 I2C 驱动

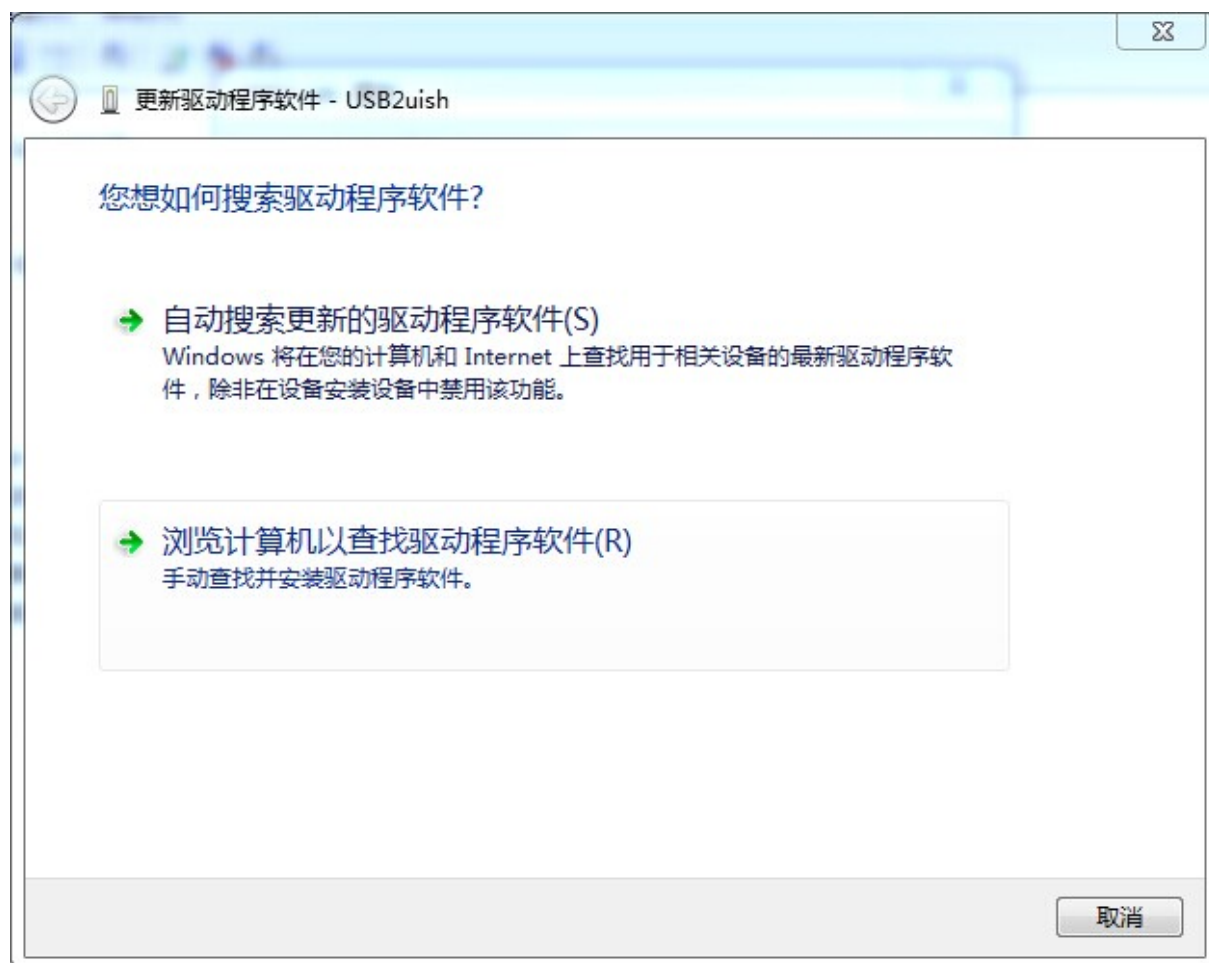
A>确保跳线为 J8 2-3, J11 2-3 短接后, 插入 USB2uish 转接板, 跳过 window7 的驱动程序自动搜索, 打开设备管理器, 将会看到



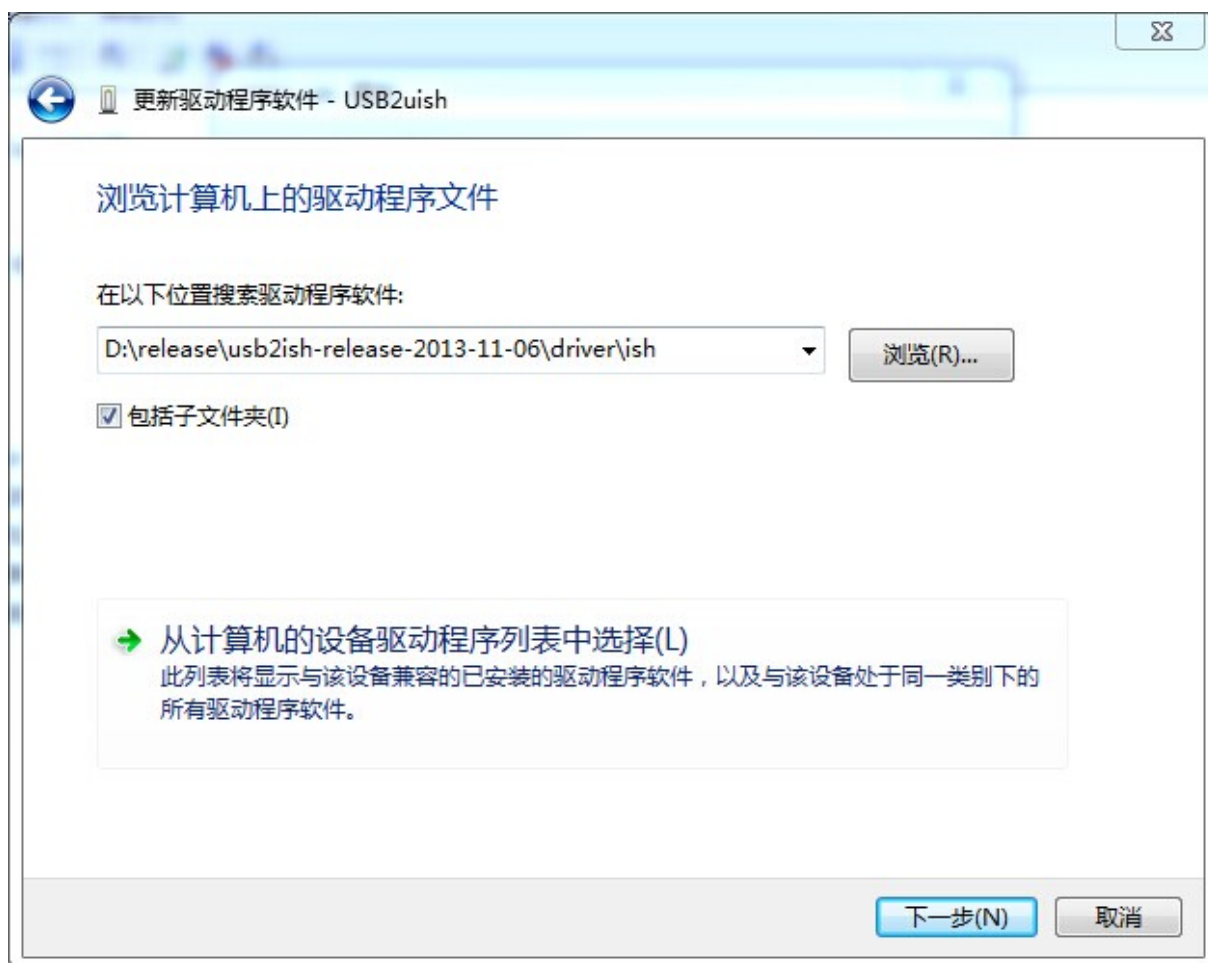
B. 双击黄色带感叹号的设备 USB2uish, 弹出 USB2uish 设备属性对话框, 如下图



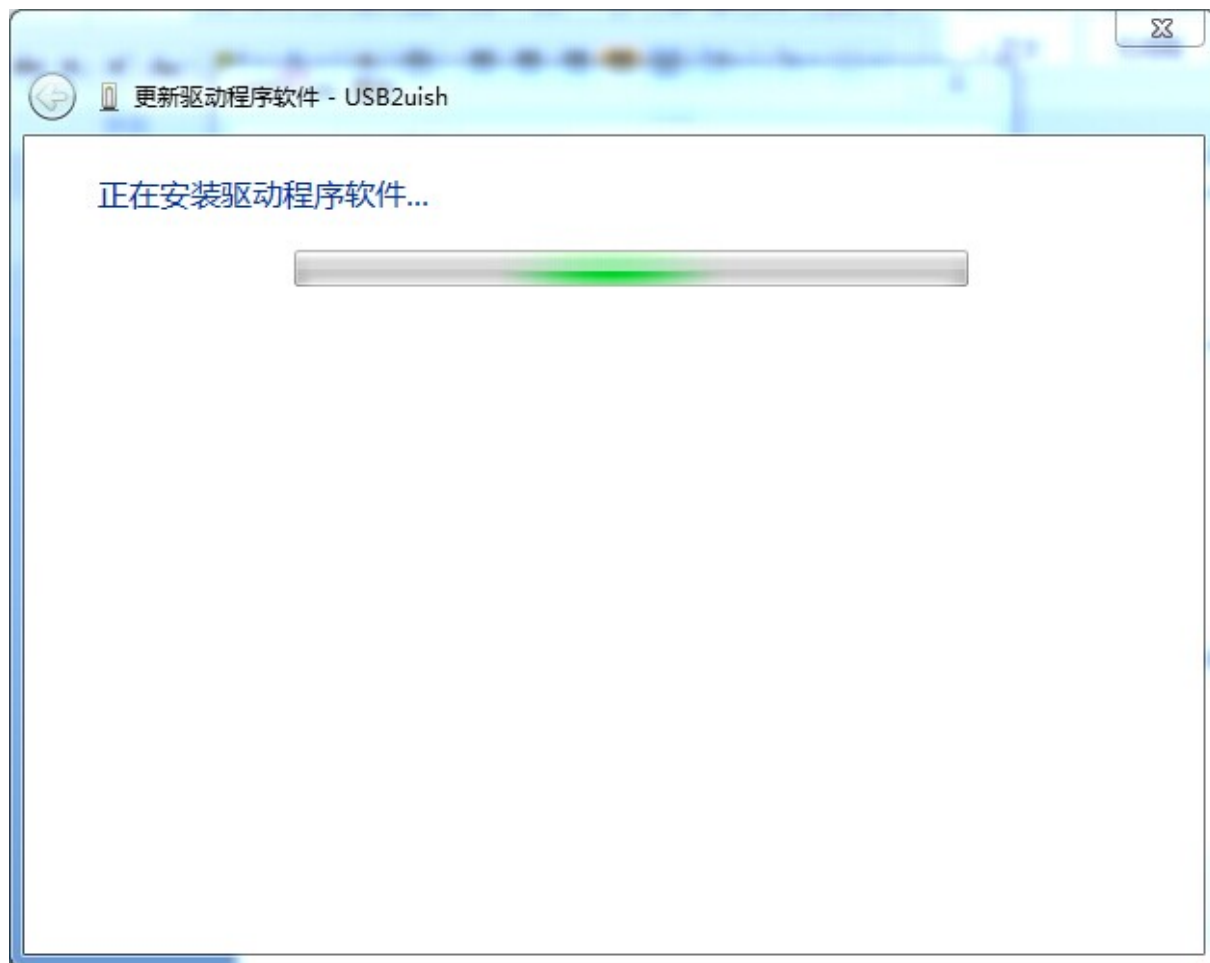
C>选择“更新驱动程序(P)...”,弹出更新驱动程序对话框, 如下图:

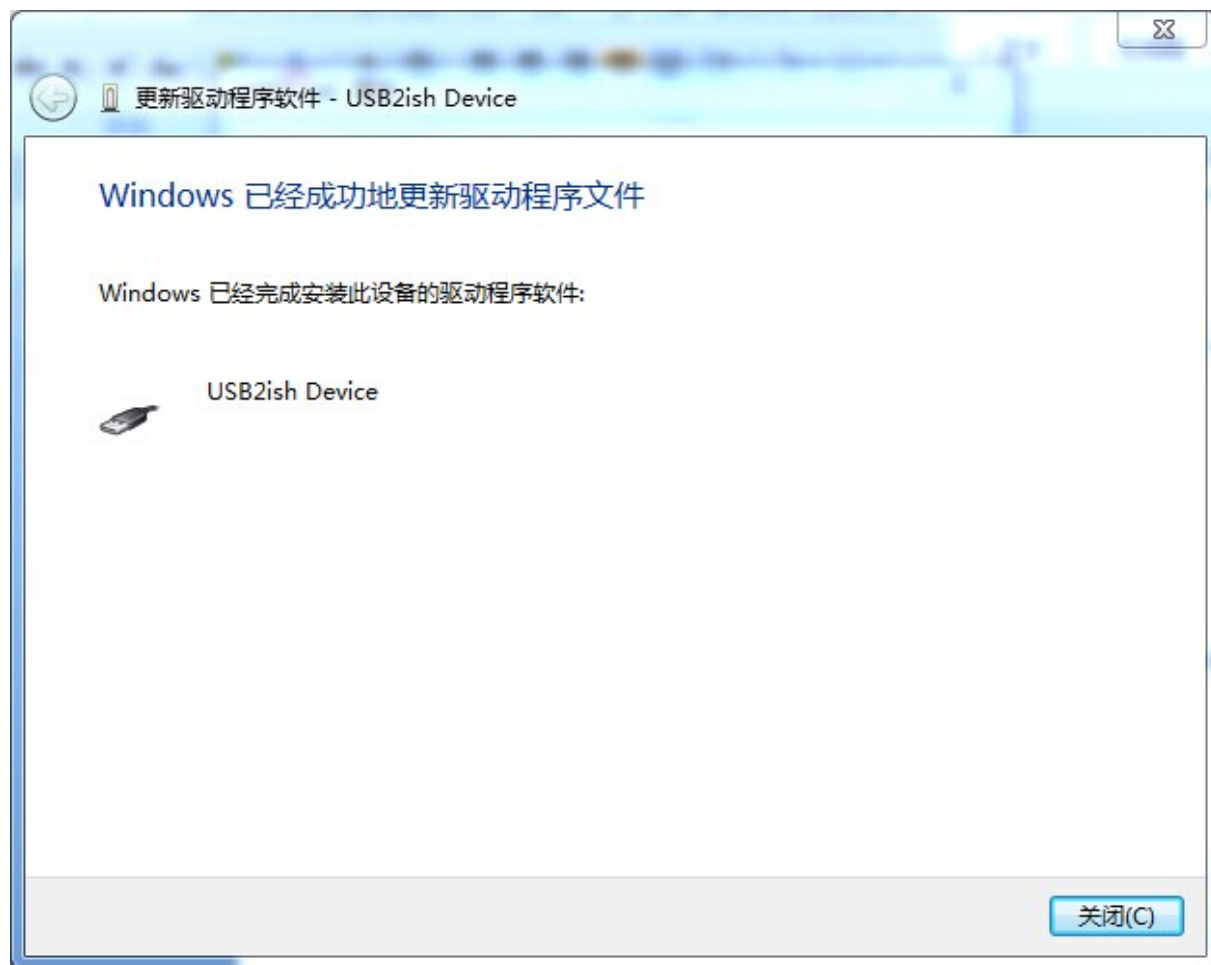


d>选择“浏览计算机以查找驱动程序软件(R)”后，点击下一步，进入下图：



E>点击“浏览 (R)”选择驱动程序文件 usb2ish.inf 所在的文件夹，点击下一步，开始加载驱动程序，如下图：

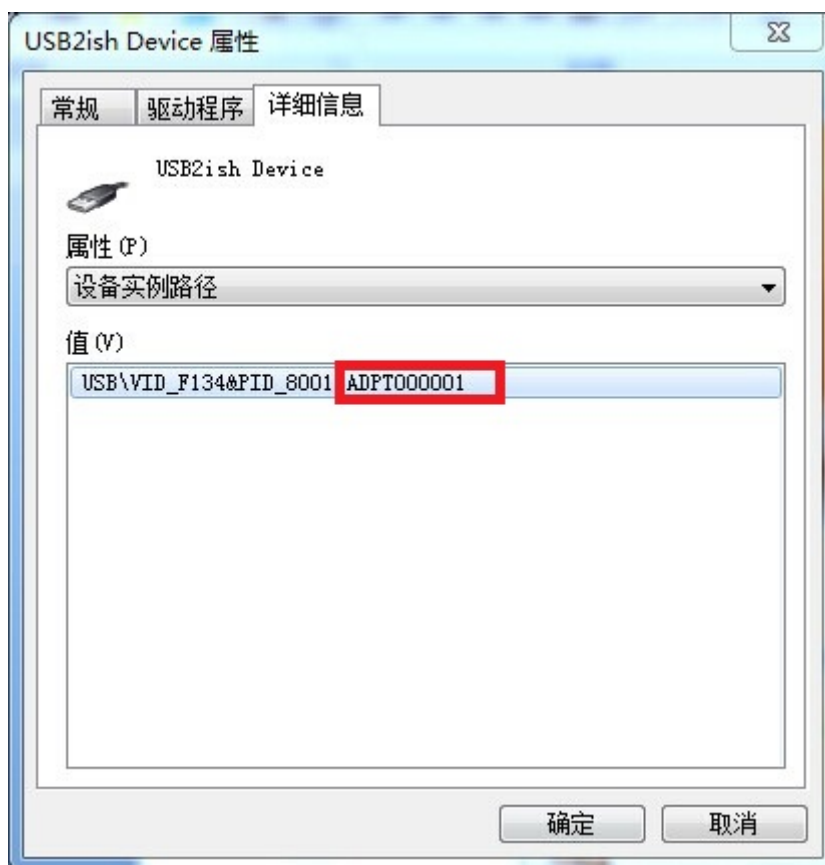




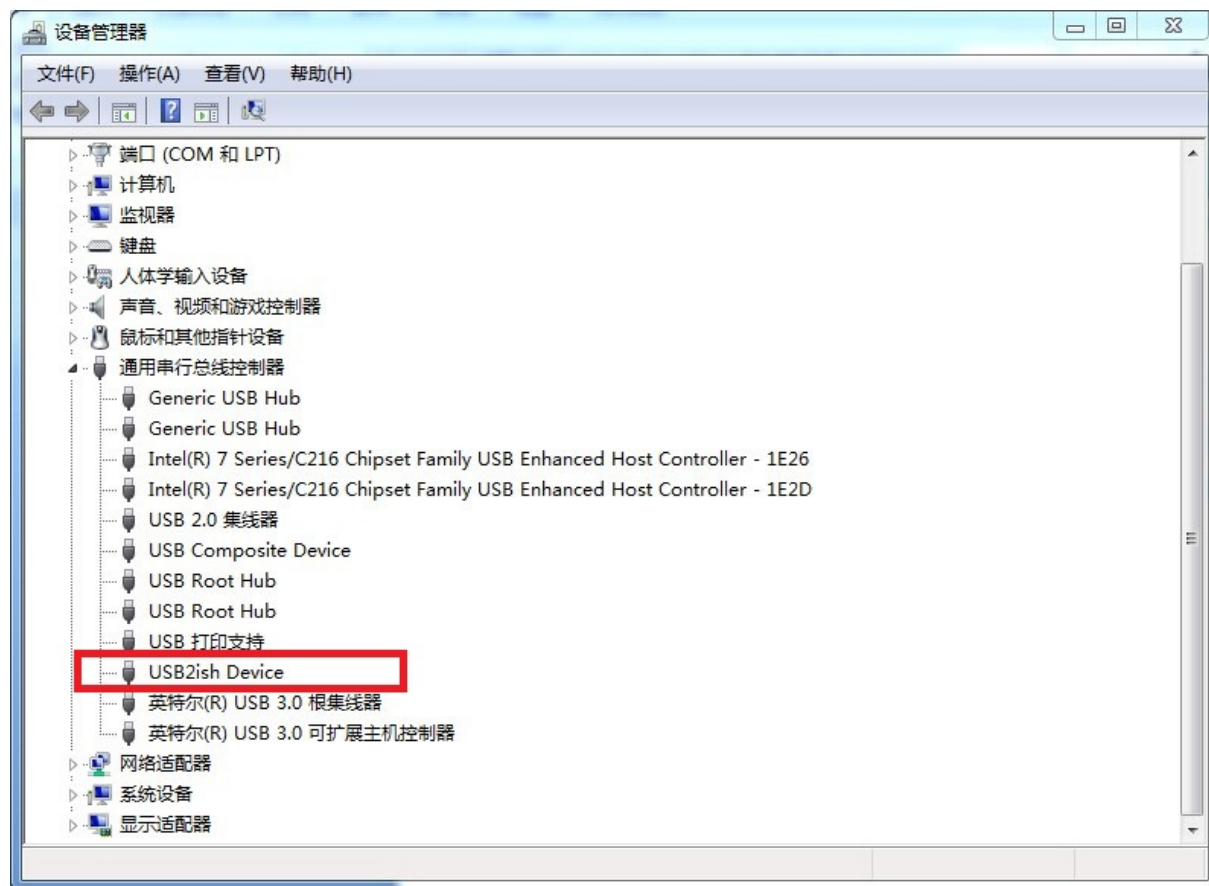
F>加载完成后，选择关闭，回到 USB2ish 设备属性驱动程序标签页，可以看到设备驱动安装信息和公司数字签名的信息，如下图：



G>在 USB2ish 设备属性详细信息标签页，可以看到设备唯一的序列标识号，如下图：

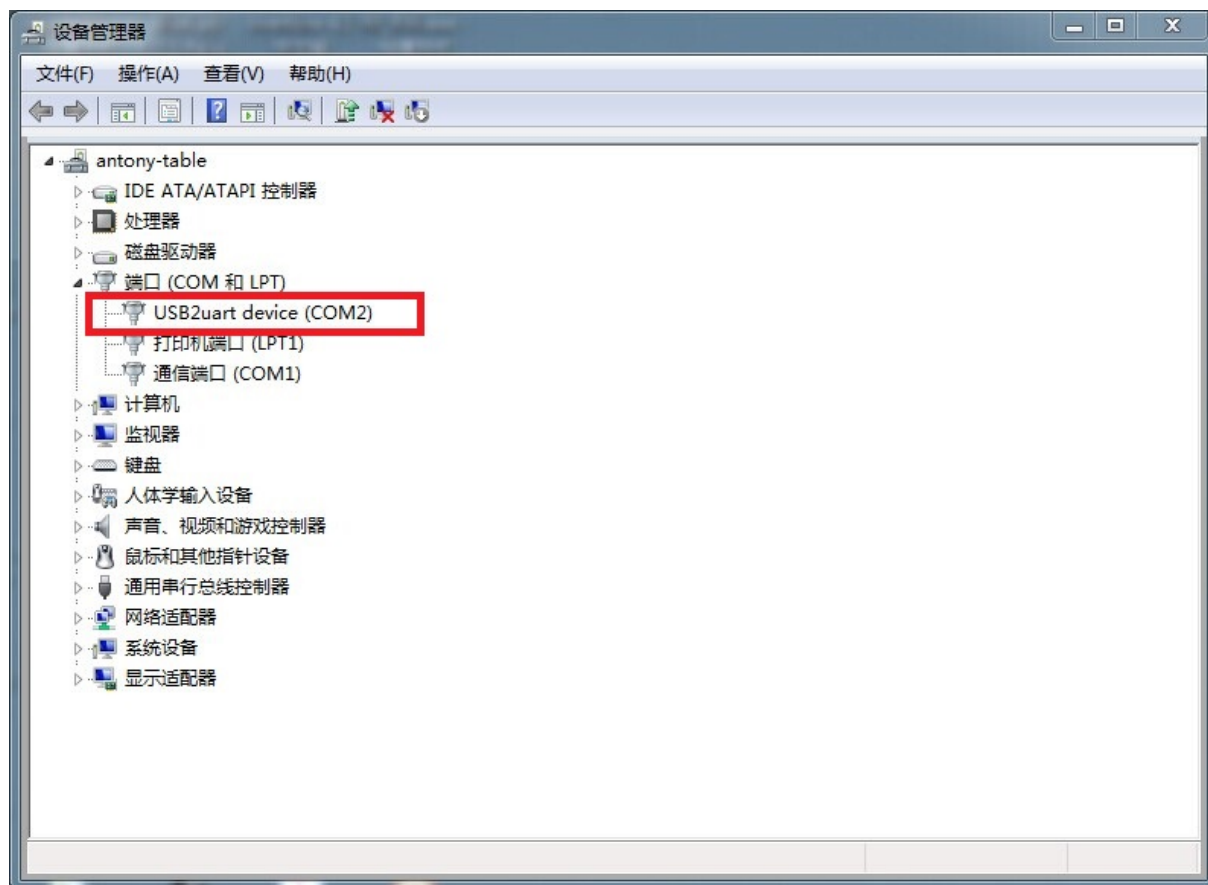


H>回到设备管理器,展开通用串行总线控制器端口前面的三角符号, 可以看到在下面多了一个新的设备, 这样证明驱动程序安装完毕。如下图:



5.2 安装 USB 转 UART 驱动程序

步骤同 1>中的步骤 ABCDE(跳线要改为 J8 1-2, J11 2-3 短接), 只不过在步骤 D 中要选择的驱动程序文件是 usb2uart.inf 所在的文件夹。安装完成后, 打开系统属性->硬件->设备管理器, 展开端口(COM 和 LPT)的三角符号, 可以看到在下面多了一个新的端口, 如下图:



六、 使用简介

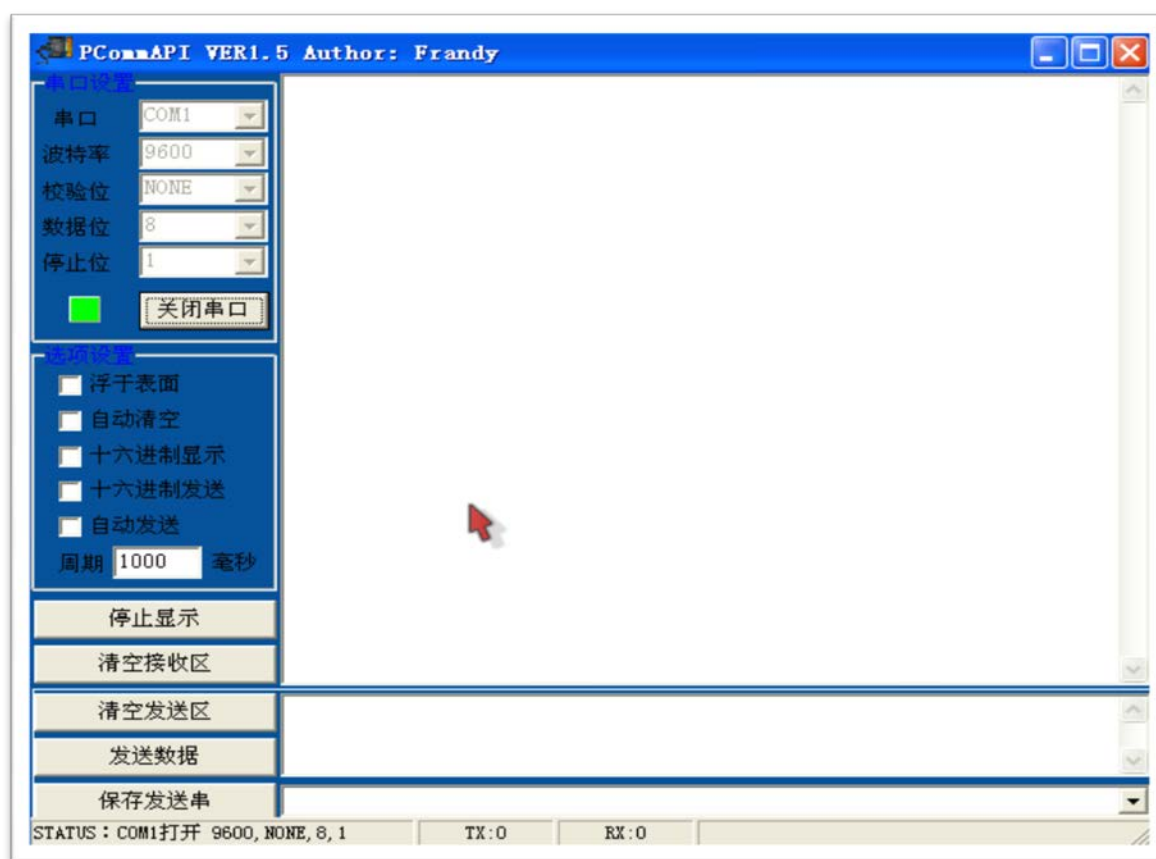
6.1> 作为 UART 口使用

确保跳线为 J8 1-2, J11 2-3 短路。

6.1.1> window 下使用:

打开软件，打开 pcommtest.exe，选择 com，就可以进行 com 通讯了。

如下图



6.2.2> linux 下使用:

插上设备，在控制台输入 `ls /dev/ttyACM*`，如果发现有新设备 `ttyACM0~9` 出现，则证明可以在 linux 下进行 com 通讯了。

6.2 作为 I2c 使用

确保跳线为 J2 2-3, J11 2-3 短路。打开软件 Usb2ish_pro.exe, 如下图:



此时设备还没有连接, 插上 USB 转接板, 点击下边的连接设备按钮, 可以看到连接成功后窗口标题的变化, 多了设备索引号和设备序列号, 如下图:



继续点击版本信息按钮, 可以看到各个软件的信息, 如下图:



连接成功后，切换到 I2c 页面,如下图



6.2.1> 选择设备

USB 转 I2C 默认的设备选择是 EEPROM，如果读写的设备不是 EEPROM 时，则必须在设备选择项里去掉 EEPROM 前面的钩选。如下图：



6.2.2> 设置

进行读写操作前先设置好 I2C 设备地址，通讯频率和读写超时参数，按设置按钮，直到弹出设置 OK 对话框证明设置成功。复位按钮恢复初始设置。如下图：



注意：

- 设备地址编辑框是 16 进制字符，最多 2 个字符，默认是 0xA0。
- 设备地址的放在 bit7~bit1，bit0 是读写 bit，固定是 0。
- 设置地址设置成功后，只要不断电，读写时不用反复设置。

6.2.3> 操作选择

操作选择里头有三种操作模式可供选择：

定时操作：每隔一定的时间进行一次读或写操作。

手动操作：必须点击执行读出或执行写入按钮进行一次操作。

触发操作：检测 IRQ 脚上的电平变化进行一次读或写操作。

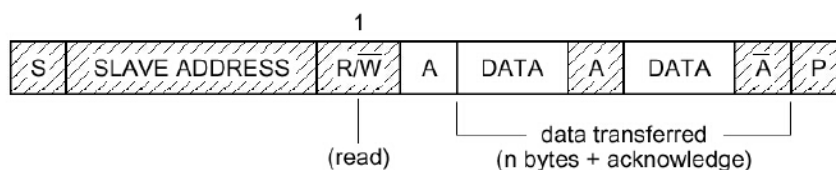
无论是定时操作，触发还是手动操作，都只能是读或写操作。


6.2.4> 读写操作

I2c 设备读写有三者种格式，分别如下：

A. 立即读模式

此种模式不常用，格式如下：



 from master to slave

 from slave to master

A = acknowledge (SDA LOW)

\bar{A} = not acknowledge (SDA HIGH)

S = START condition

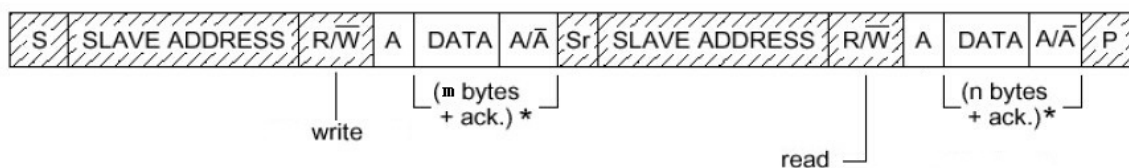
P = STOP condition

例如：设备地址是 0x68，n=16，立即读 16 个 bytes，应先设置设备地址 68，点击设置按钮，接着在读入的数据长度填 16，命令或地址参数不填，然后点击执行读出，如下图：



B. 复合读模式

此种模式读用的最多，格式如下：



 from master to slave

 from slave to master

A = acknowledge (SDA LOW)

\bar{A} = not acknowledge (SDA HIGH)

S = START condition

P = STOP condition

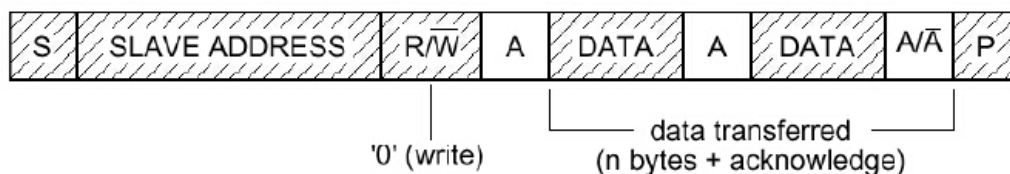
例如:设备地址是 0x68, m =2,, n =200, m 区的 data 是 0x1234, 按下图设置:




点击执行读出, 即执行立即读操作。

c. 写模式

格式如下:



 from master to slave

 from slave to master

A = acknowledge (SDA LOW)

\bar{A} = not acknowledge (SDA HIGH)

S = START condition

P = STOP condition

例如：向 0x1234 处写入 0x31, 0x32, 0x33, 0x34, 0x35，按下图设置：



或者



或者



点击执行写入后都是一样的效果。

6.2.5> 特殊功能(批量写)

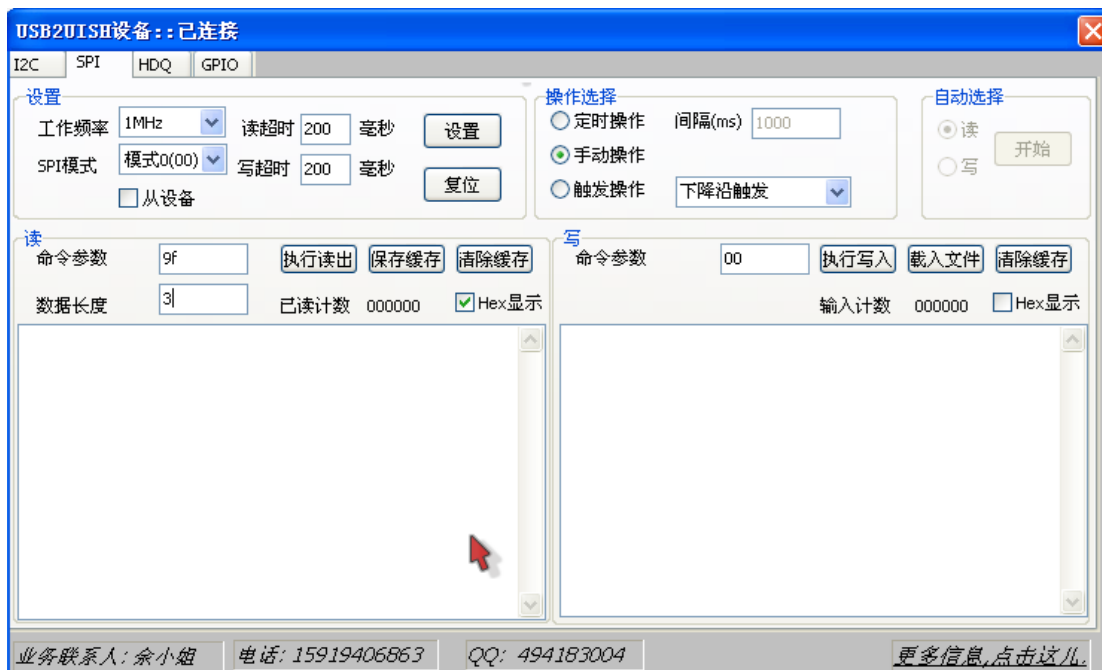
例如要向寄存器 0x01, 0x03, 0x05, 0x09 分别写入 0xA0, 0xA1, 0xB0, 0xB1, 则可以用分段写功能, 先勾选分段写, 然后设置分段写长度为 2, 写区间配置如下图:



最后点击执行写入,则会按设置的长度分段写入,直到写完为止。

6.3 作为 SPI 使用

确保跳线为 J8 2-3, J11 2-3 短路。打开软件 usb2uis.exe.如果不是 SPI 页面,点击 SPI 标签,如下图



USB 转 SPI 既可以作为主设备使用，又可以作为从设备使用。默认是主设备。

6.3.1> 作为主设备使用

A. 设置

进行读写操作前先设置好 SPI 通讯频率和读写超时参数，按设置按钮，直到弹出设置 OK 对话框证明设置成功。复位按钮恢复初始设置。如下图：



SPI 模式说明：

- 模式 00：SCK 在空闲状态时处于低电平，在 SCK 周期的第一个边沿采样数据
- 模式 01：SCK 在空闲状态时处于低电平，在 SCK 周期的第二个边沿采样数据
- 模式 10：SCK 在空闲状态时处于高电平，在 SCK 周期的第一个边沿采样数据
- 模式 11：SCK 在空闲状态时处于高电平，在 SCK 周期的第二个边沿采样数据

B. 操作选择

操作选择里头有三种操作模式可供选择：

- 定时操作：每隔一定的时间进行一次读或写操作。
 - 手动操作：必须点击执行读出或执行写入按钮进行一次操作。
 - 触发操作：检测 IRQ 脚上的电平变化进行一次读或写操作。
- 无论是定时操作，触发还是手动操作，都只能是读或写操作。

C. 读写操作

SPI 读写也有三种格式：

1. 无命令参数的读模式

如下表：

BYTE	First	Second	Third	Last
MOSI	DUMMY	DUMMY	DUMMY	DUMMY	DUMMY
MISO	Data1	Data2	Data3	DataN

DUMMY 意思随便什么数据，一般填 00

如 N = 100，通过 SPI 读 100 个 byte，读区间配置如下图：



.然后点击执行读出。

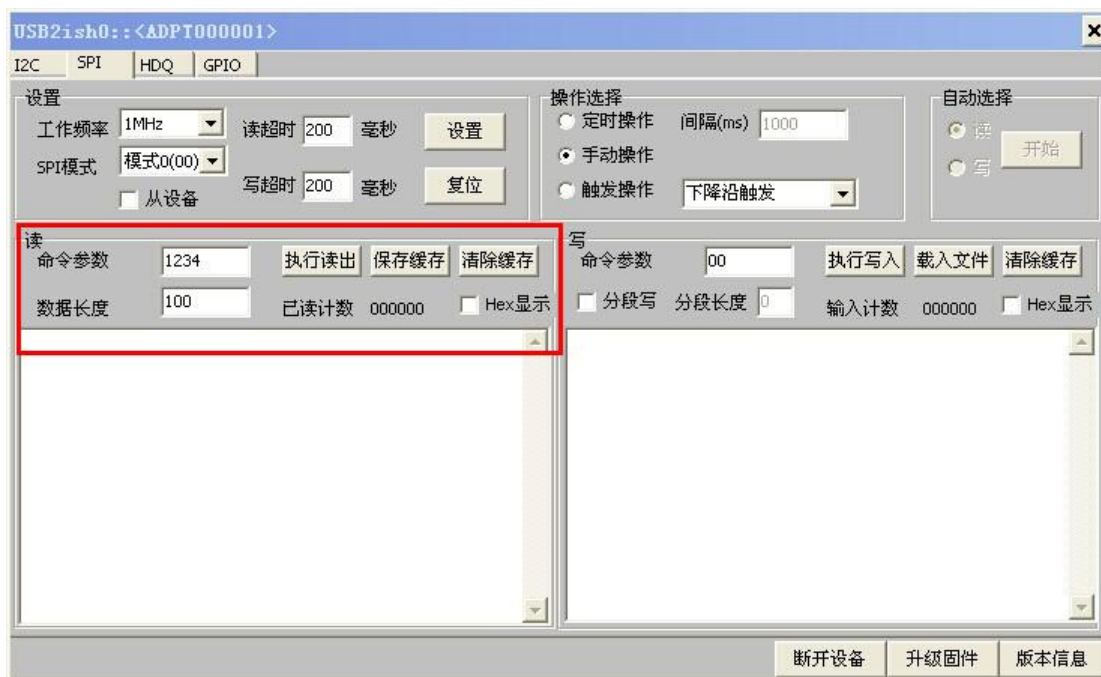
2. 有命令参数的读模式

如下表：

BYTE \	First	Second	Third	Last
MOSI	COMMAND1	COMMAND2	DUMMY	DUMMY	DUMMY
MISO	DUMMY	DUMMY	Data1	DataN

这儿 COMMNAD 最长可以是 4 个 byte。

如 command 是 0x1234，N = 100，通过 SPI 读 100 个 byte，读区间配置如下图：



然后点击执行读出。

3. 写模式

BYTE \	First	Second	Third	Last
MOSI	COMMAND1	COMMAND2	DATA1		DATAN
MISO	DUMMY	DUMMY	DUMMY	DUMMY

这个 COMMAND 最长可以是 4 个 byte。

如 command 是 0x1234，N = 5，写入的数据时 0x31, 0x32, 0x33, 0x34,0x35，写区间配置如下图：



或者



或者



然后点击执行写入后都是一样的效果。

D. 特殊功能

如 I2c 写一样，SPI 写也有分段写功能，操作方式跟 I2c 一样。

6.3.2> 作为从设备使用

要 SPI 工作为从设备，需要勾选从设备，并点击设置按钮，直到弹出设置 ok 对话框指示设置成功。如下图：



作为从设备工作时,功能是如下限制:

- 1> 从设备只能接收，不能发送。
- 2> 主设备最大的时钟频率不能超过 4M。
- 3> 主设备单次发送的数据长度不能大于 128 BYTES。
- 4> 作为从设备时，接收的数据显示在读数据区域。

6.4> 作为 HDQ 使用

HDQ 通信协议是美国德州仪器的单线通信协议。智能电池的检测模块就是用这个协议。上网可搜到有关 HDQ 的通讯规范。

如 I2C 使用那样，连接成功后，切换到 HDQ 页面，如下图：



如果读写数据量比较大时,需要把读写超时的设置加大些。时钟请根据设备的最佳时钟设置。进行读写操作前先设置好 HDQ 通讯时序和读写超时参数,按设置按钮,直到弹出设置 OK 对话框证明设置成功。复位按钮恢复初始设置

操作选择里头有两种操作模式可供选择

定时操作:每隔一定的时间进行一次读或写操作

手动操作:必须点击执行读出或执行写入按钮进行一次操作

6.5> 作为 GPIO 使用

如 I2C 使用那样,连接成功后,切换到 GPIO 页面,如下图:



此页面功能分为三个部分：ADC，数字 IO 和 PWM，如下图：



6.5.1> 数字 IO 和 ADC 采样端接口定义

标号	丝印	软件功能配置	功能说明
J7-1	VDD		电源正
J7-2	GND		电源地
J7-3	IO8	J7-03	ADC 采用通道或数字 IO
J7-4	IO7	J7-04	ADC 采用通道或数字 IO
J7-5	IO6	J7-05	ADC 采用通道或数字 IO
J7-6	IO5	J7-06	ADC 采用通道或数字 IO
J7-7	IO4	J7-07	ADC 采用通道或数字 IO
J7-8	IO3	J7-08	ADC 采用通道或数字 IO
J7-9	IO2	J7-09	ADC 采用通道或数字 IO
J7-10	IO1	J7-10	ADC 采用通道或数字 IO

6.5.2> ADC 使用说明

10位的分辨率，最大200 ksps采样频率，支持单端/差分输入。带有内部参考电压和一个温度传感器。J7-3和J7-10可以配置为任何通道的正极或负极输入。最多可以四个通道轮流采样。当GND被选择为负极输入时，ADC工作在单端方式；否则，ADC工作在差分方式

A、使用采样功能

勾选所用的ADC通道，选择每个通道的正负极输入选择，按设置按钮，知道弹出设置OK对话框，确认后ADC采样设置成功。

选择采样的操作模式(分为手动，定时和触发三种模式)后，按开始采样按钮，进行采样操作。触发模式和定时采样模式启动后，要停下的话需要再按开始采样按钮(此时的按钮标题已变成停止采样)。最新的采样值总是显示在当前值得编辑框里头，先前的采样值保存在历史模拟数据的编辑框里头。历史数据可以以文本文件的形式保存在指定的地方。

B、采样电压计算

定义Value 为采样读取的值, V_{REF} 为参考电压，为2.44V，

则采样的电压值 V_S 计算如下：

单端方式 $V_S = \text{Value} * V_{REF} / 1023$; (V_S 范围在0~ V_{REF} 之间)

差分方式 $V_S = \text{Value} * V_{REF} / 511$; (Value的高位Bit9~15全为0)

$V_S = -(\text{Value} \& 0xFE) * V_{REF} / 511$; (Value的高位Bit9~15全为1)

C、内部温度传感器计算

$\text{Temp} = (V_S - 0.776) * 1000 / 2.86$ 单位 $^{\circ}\text{C}$

6.5.3> 数字 IO 的使用

当J7-03~J7-10不被用作ADC通道使用时，可以配置为数字IO使用。

- 1> 在数字IO配置栏目中勾选相应的IO口，则配置为输入，不勾选的配置为输出，选择好后要按设置按钮使设置生效，弹出设置OK对话框后证明设置成功，相应的数字IO读写栏目里头的电平会重新刷新一次。
- 2> 数字IO读写栏目里头的读出和写入按钮用于刷新或变更数字IO的端口电平。配置为ADC采样选择的IO口不能配置为数字IO，相应数字IO配置栏里头的勾选变为灰色。配置为ADC采样选择或配置为数字IO输入的IO口在数字IO读写栏里头变为灰色，只能读出，不能写入。且配置为ADC采样选择的IO在读入时其电平总是0。

6.5.4> PWM 使用

本设备最多可以输出四路，100K的PWM，每个PWM的输出频率都是一样的且同时开启和关闭。

占空比则可以单独调整。通道数为0是没有PWM输出，为1时一号通道输出，为2时，一二号通道输出，以此类推。

选择好通道数，频率和占空比后，按设置按钮，弹出设置OK对话框后证明设置成功。

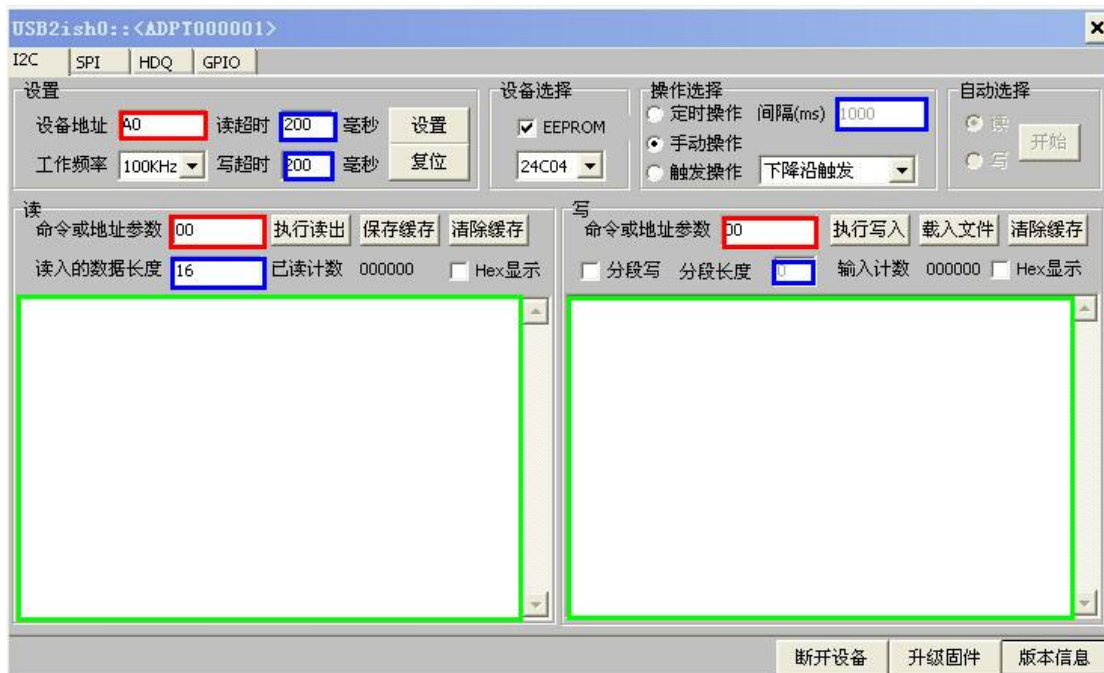
由于软件运行误差，最大占空比有一点点限制，跟PWM频率有一定的关系。

实测最大占空比见下表:

频率(KHz)	最小占空比	最大占空比	占空比为0时输出	超过最大占空比时输出
1	0.1%	99.9%	高电平	低电平
2	0.1%	99.8%	高电平	低电平
4	0.1%	99.7%	高电平	低电平
6	0.1%	99.6%	高电平	低电平
8	0.1%	99.5%	高电平	低电平
10	0.1%	99.3%	高电平	低电平
20	0.1%	98.7%	高电平	低电平
40	0.1%	97.5%	高电平	低电平
60	0.1%	96.2%	高电平	低电平
80	0.1%	95.0%	高电平	低电平
100	0.1%	93.7%	高电平	低电平

6.6> 补充说明

数据显示格式，如下图：





- 1> 红色的编辑框只能是十六进制的字符。
- 2> 蓝色的编辑框只能是十进制的字符。
- 3> 绿色的编辑框既可显示为十六进制字符也可显示为文本字符。
- 4> 执行读出或执行写入成功后，没有任何成功提示。如果失败则会弹出一个失败对话框。如下图：



6.7> 固件更新

确保跳线为 J11 1-2 短路，插入 USB2ish 设备，成功连接设备后，点击固件升级按钮， 如下图：



点击选择文件后，再按执行，成功后如下图：



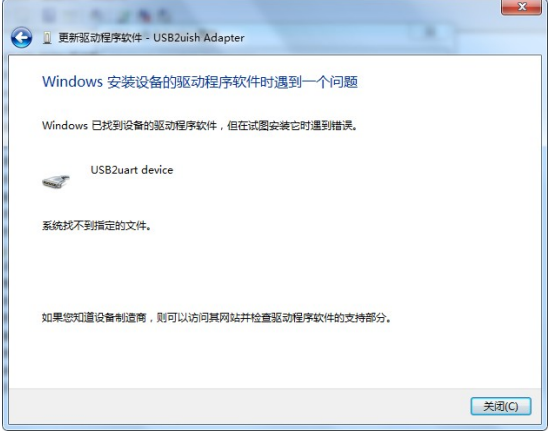
不同的软件有不同的版本和发行时间，更新成功后，重新跳线，回到 IIC 或 SPI 或 HDQ 使用界面，点击版本信息，可以查看固件版本和发行时间，检测是否更新成功。如下图：



版本如有变化，以实际使用为准。

七、 常见问题及解决方法

	故障现象	可能原因	解决方法
1	拔插 USB，设备管理器没有反应	PC USB 端口坏	更换至其他 USB 端口
		驱动程序僵死	重新启动电脑
		USB 转接板坏	更换转接板
2	读不到软件版本信息，如下图：很多问号 	跳线设置了升级模式	拔下，更新跳线再插上
		驱动程序僵死	重新启动电脑
3	读写失败，甚至复位	跳线设置了升级模式	拔下，更新跳线再插上
		驱动程序僵死	重新启动电脑
		USB 转接板坏	更换转接板
4	Demo 软件不能运行，如下图： 	应用程序没有执行权限	右键选择以管理员权限运行
5	驱动程序安装失败	检测跳线，驱动文件夹选择是否正确	
		360 杀毒软件	关闭 360 后再安装驱动
6	DLL 调用函数 USBIO_OpenDevice 返回 0xFF	原因同第 2 项	解决方法如第 2 项
		动态调用是有此现象，DLL 装载后需要时间初始化	装载 DLL 后，延时 100ms 才可调用 DLL 函数

7	找不到 DLL 函数说明		解压附带软件包后，用文本编辑器打开 DLL 文件下的 USBio.h 头文件，即可看到函数说明
8	安装 UART 驱动失败问题,如下图: 	1> window 系统 inf 文件夹下，没有 mdmcpq.inf 文件，或者 system32\dirver\ 没有 usbser.sys 文件 2> mdmcpq.inf 文件段中的语法错	1> 网上下载相应系统版本的 mdmcpq.inf 和 usbser.sys 到指定的文件夹 2> 将 :\Windows\inf\mdmcpq.inf 中 [FakeModemCopyFileSection] usbser.sys,,,0x20 改为 [FakeModemCopyFileSection] ;usbser.sys,, ,0x20

八、 维护

- 1> 三个月的非人为硬件损坏，可以免费更换。
- 2> 软件二年免费更新和升级。
- 3> 特殊要求的可定制。