

---

# **USB2UIS USER Manual**

Version: 6.0

Release: Jul 15 2016

OnEasyB

---

## Contents

<b>USB2UIS USER MANUAL.....</b>	<b>1</b>
1. OVERVIEW .....	3
General Features .....	3
Electrical characteristics.....	3
2. HARDWARE SYSTEM .....	5
2.1> Outline dimension .....	5
2.2> Appearances.....	5
2.3> <i>location of the interface and port define</i> .....	6
3. RELEASED PACKAGE DESCRIPTION(AFTER EXTRACTED) .....	10
4. DRIVER INSTALLATION .....	11
4.1> <i>Install the driver of USB2ish</i> .....	11
4.2> <i>Install the device of USB2Uart</i> .....	19
5. USAGE IN THE WINDOW SYSTEM .....	20
5.1> <i>Use UART</i> .....	20
5.2> <i>Use I2c</i> .....	21
5.2.1>     Select the device.....	24
5.2.2>     Setting.....	25
5.2.3>     Operation choice .....	27
5.2.4>     Read and write.....	27
5.2.5>     Special function( subsection).....	31
5.3> <i>Use SPI</i> .....	32
5.3.1>     Work as master.....	33
5.3.1.1>     Setting .....	33
5.3.1.2>     Self-checking .....	34
5.3.1.3>     Operation choice.....	36
5.3.1.4>     Read and write operation .....	36
5.3.1.5>     Special function.....	40
5.3.1.6>     Read and write of SPI flash .....	40
5.3.2>     Work as slaver ( Available for extened type) .....	40
5.4> <i>Usage GPIO ( Available for extened type)</i> .....	41
5.4.1>     Definition of digital IO and the interface of the sampling part.....	43
5.4.2>     ADC.....	43
5.4.3>     GPIO .....	44
5.4.4>     PWM.....	45
5.5> <i>File operation</i> .....	46
5.5> <i>Pop-up menu</i> .....	46
5.6> <i>Upgrade Firmware</i> .....	47
6. USAGE IN THE LINUX SYSTEM.....	50
6.1> Use Uart .....	50
6.2> Use I2C, SPI , GPIO,PWM.....	50
7. ISSUES LIST AND SOLUTIONS.....	51
8. MAINTENANCE.....	53

---

## **1. Overview**

USB2UIS is a practical and convenient multi-functional adapter that has two types(basic and extended). Both integrate the USB-UART, USB-I2C, USB-SPI functions, and the extended also integrates USB-ADC, USB-PWM, USB-GPIO functions. It is very convenient to communicate with the device that includes interface of UART, I2C, SPI and so on through the PC software.

### **General Features**

- 1> Optional 3.3V and 5V output voltage.
- 2> Auto-instauration Fuse , to prevent over current.
- 3> USB 2.0 with full speed communication rate.
- 4> Support I2c device address auto-detect and SPI self-test.
- 5> Rich reference codes(VB, VC, labview, QT, c++builder) provided to support the third party to develop.
- 6> Unique device serial number, one PC can open the multiple devices simultaneously.
- 7> Conveniently to acess 24 serials EEPROM and 25 serials FLASH.
- 8> To support the decoding of Hex extension file .
- 9> To support all window system(XP and above) and Linux system.
- 10>Firmware can be updated through the PC USB interface. For special applications, can also be customized.

### **Electrical characteristics**

- 1> All digit IO High level voltage at 2.7 ~3.6V ,low level at -0.3 ~ 0.3V。
- 2> All digit IO Max. current < 10ma
- 3> Baud Rate of UART is up to 230400.
- 4> I2c frequency has 5 settings(100K,200K,300K,400K, 800k) .
- 5> The maximum setting of SPI frequency is 12M, and has 9 settings

---

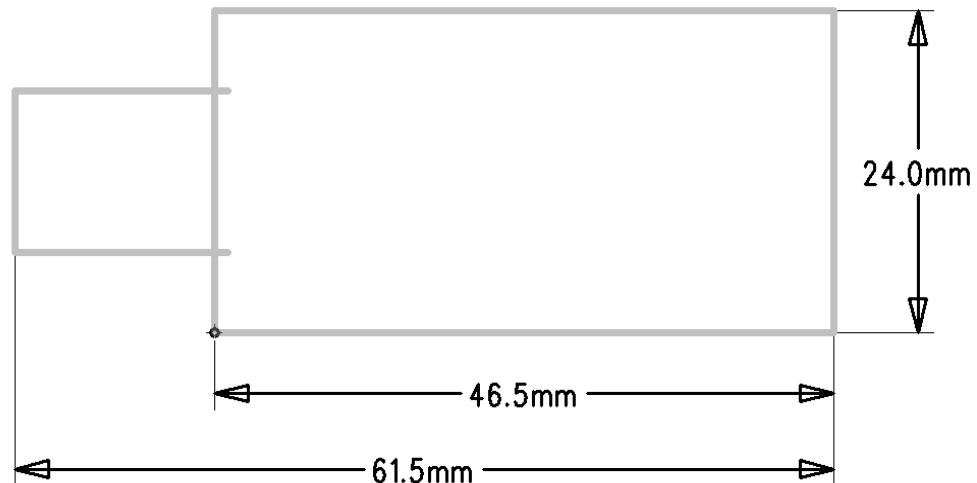
(200K,400K,600K,800K,1M,2M,4M,6M,12M). for extended type, SPI can work as slaver.

- 6> For extended type, it has a 10bit ADC with 4 channels that each can be configured as single-ended and differential inputs.
- 7> For extended type ,It has 8 independent IO that each can be configured as input or output and 2 dedicated IO. One is for output and the other is for interrupted input.
- 8> For extended type ,it has a 4-channels PMW by the same frequency that is up to 100k and the duty cycle is adjustable.

---

## **2. Hardware system**

### **2.1> Outline dimension**



---

### **2.2> Appearances**

Two types ( Basic and Extended).

The left is the basic type and the pin spacing of interface is 2.54mm.

The right is the extended type and the pin spacing of interface is 2.0 mm.



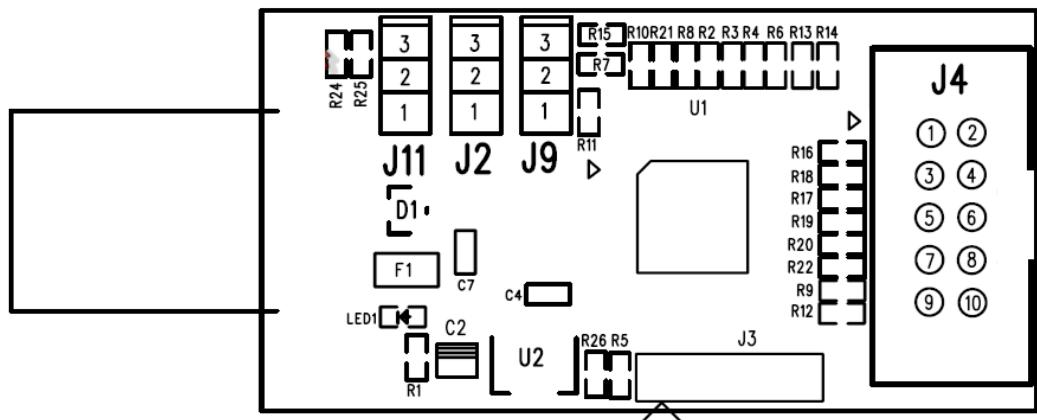
Basic type



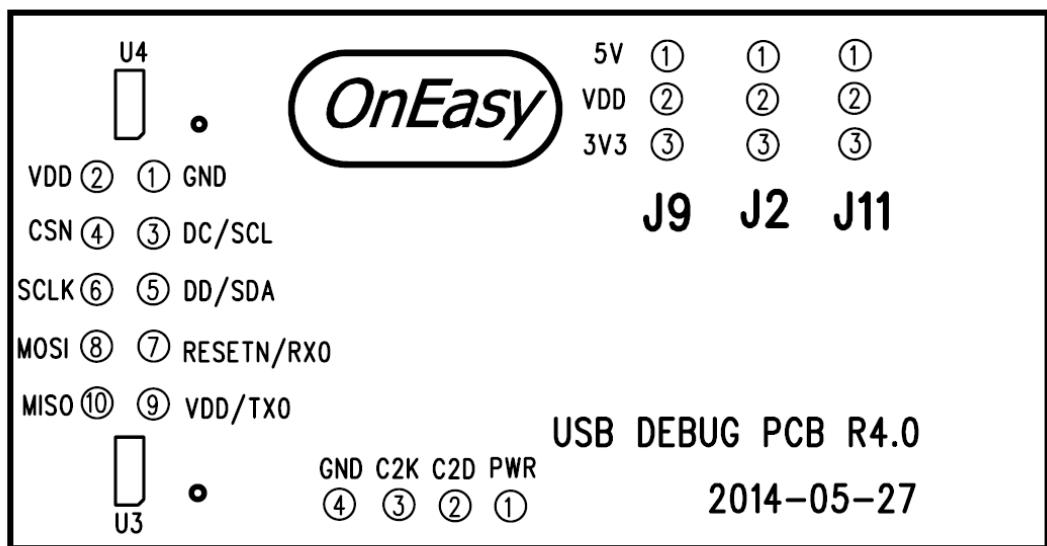
Extended type

## 2.3> location of the interface and port define

### 1> Basic type



Top-view



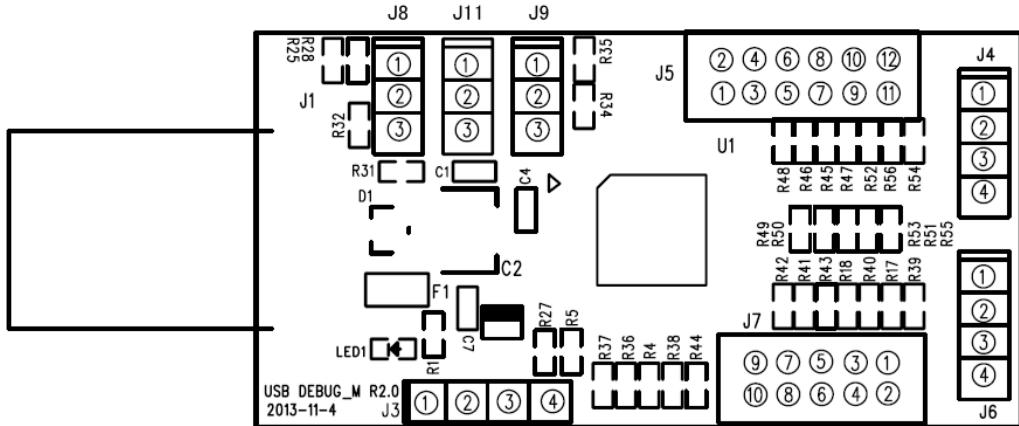
Bottom-view

---

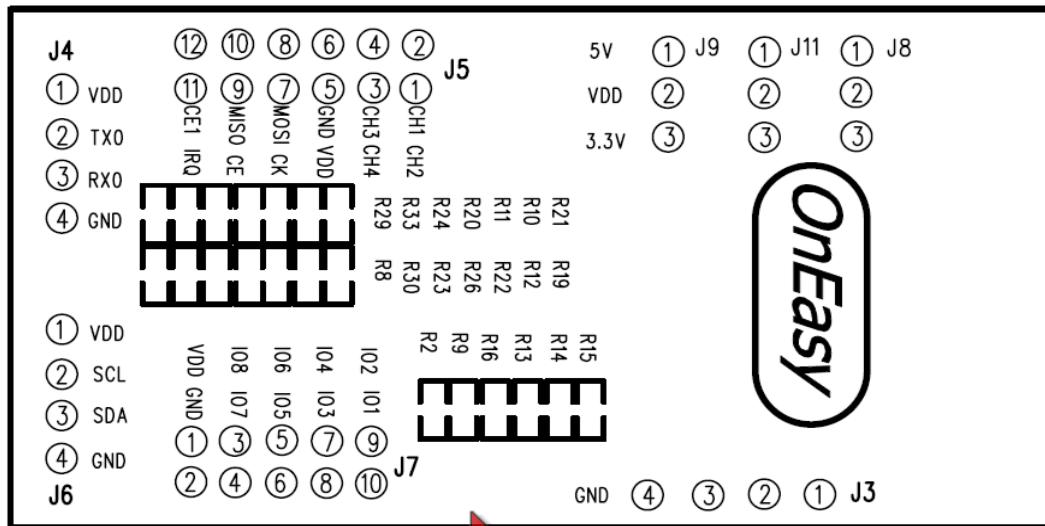
Port description of basic type , as shown below.

Port	function	remark	
J4	POWER	J4-1 GND	J4-2 VDD(out)
	I2C	J4-3 SCL	J4-5 SDA
	SPI	J4-4 CSN J4-8 MOSI	J4-6 CLK J4-10 MISO
	UART	J4-7 RX	J4-9 TX
J2	Function jumpper	1,2 short, UART function enable; 2,3 short, I2C, SPI function enable;	
J9	Voltage jumpper	1,2 short, VDD is 3V; 2,3 short, VDD is 5V;	
J11	Upgrade jumpper	1,2 short, fireware upgrade mode; 2,3 short, normal mode, J2 decide function	

## 2> Extended type



## Top-view



## Bottom-view

Port description of extended type , as shown below:

port	function						
		Pin No.	silk	remark	Pin No.	silk	Remark
J1	USB						
J3	UNUSED						
J4	UART	J4-1	VD D		J4-2	TXD	Uart-tx
		J4-3	RX D	Uart-rx	J4-4	GN D	
J5	PWM	J5-1	CH2	PWM2	J5-2	CH1	PWM1
		J5-3	CH3	PWM3	J5-4	CH4	PWM3
	SPI	J5-5	VD D		J5-6	GN D	
		J5-7	CK	SPI-CLK	J5-8	MO SI	Master out slaver in
		J5-9	CE	SPI-CE	J5-10	MIS O	Master in slaver out
		J5-11	IRQ	Interrupted input pin	J5-12	CE1	Out pin
J6	I2C	J6-1	VD D		J6-2	SCL	I2c-clk
		J6-3	SD A	I2c-data	J6-4	GN D	
J7	GPIO	J7-1	VD D		J7-2	GN D	
		J7-3	IO8	Bit7	J7-4	IO7	Bit6
		J7-5	IO6	Bit5	J7-6	IO5	Bit4
		J7-7	IO4	Bit3	J7-8	IO3	Bit2
		J7-9	IO2	Bit1	J7-10	IO1	Bit0
J8	Function jumper	1,2 short,UART function enable; 2,3 short,I2C,SPI function enable;					
J9	Voltage jumper	1,2 short,VDD is 5V; 2,3 short,VDD is 3.3V					
J11	Upgrade jumper	1,2 short,firmware upgrade mode; 2,3 short,normal mode, J8 decide function					

---

### ***3. Released package description(after extracted)***

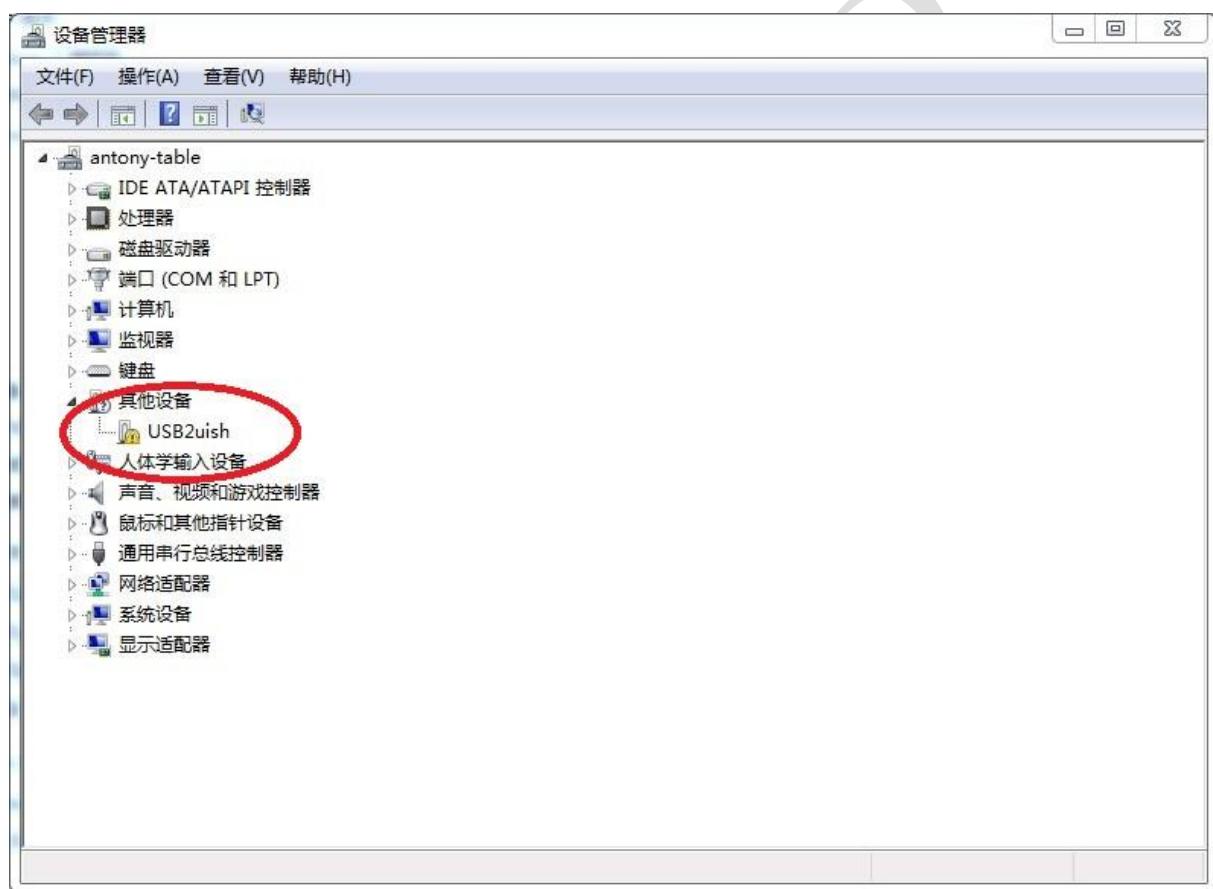
folder	files	remark
APP	Usb2ish_pro.exe	application software( Chinese)
	Usb2ish_pro_en.exe	application software( English)
	HybridEdit.ocx	Active control for usb2ish_pro.exe
	Usb2ish.dll	Dll file for usb2ish_pro.exe
LIB	Linux/libUSB2UIS.so	The lib files in the linux
	Window/usb2uis.dll,usb2uis.lib	The lib files in the window
DOC	USB2UIS 用户手册.pdf	user manual( Chinese )
	USB2UIS User Manual.pdf	user manual( English )
	Lib user Manual.pdf	Lib function description
DRIVER	Ish	Window driver for usb2ish
	Uart	Window driver for Usb2uart
FIRMWARE	USB2ISH_FM_BXX.bin	the firmware file to upgrade Basic type
	USB2ISH_FM_MXX.bin	the firmware file to upgrade Extented type
DEMO	cbc2009	The demo codes in C++builder2009
	Labview2012	The demo codes in labview2012
	Qt-creator301	The demo codes in qt-creator3.01
	VS2010/vc	The demo codes in VS2010 VC
	Vs2010/vb	The demo codes in VS2010 VB
OTHER	uartassist.rar	a tool soft for usb2uart test

## **4. Driver installation**

Below about the driver installation steps of the extended type's board on the window7.

### **4.1> Install the driver of USB2ish**

A> Make sure that both the function jumper and the upgrade jumper are 2-3 short, insert the USB2UIS board, skip window 7 driver to search automatically, open the device manager , and you will see



B> Double click on the device USB2uish of yellow with exclamation mark , the properties dialog box of USB2uish device pops up, as shown below.

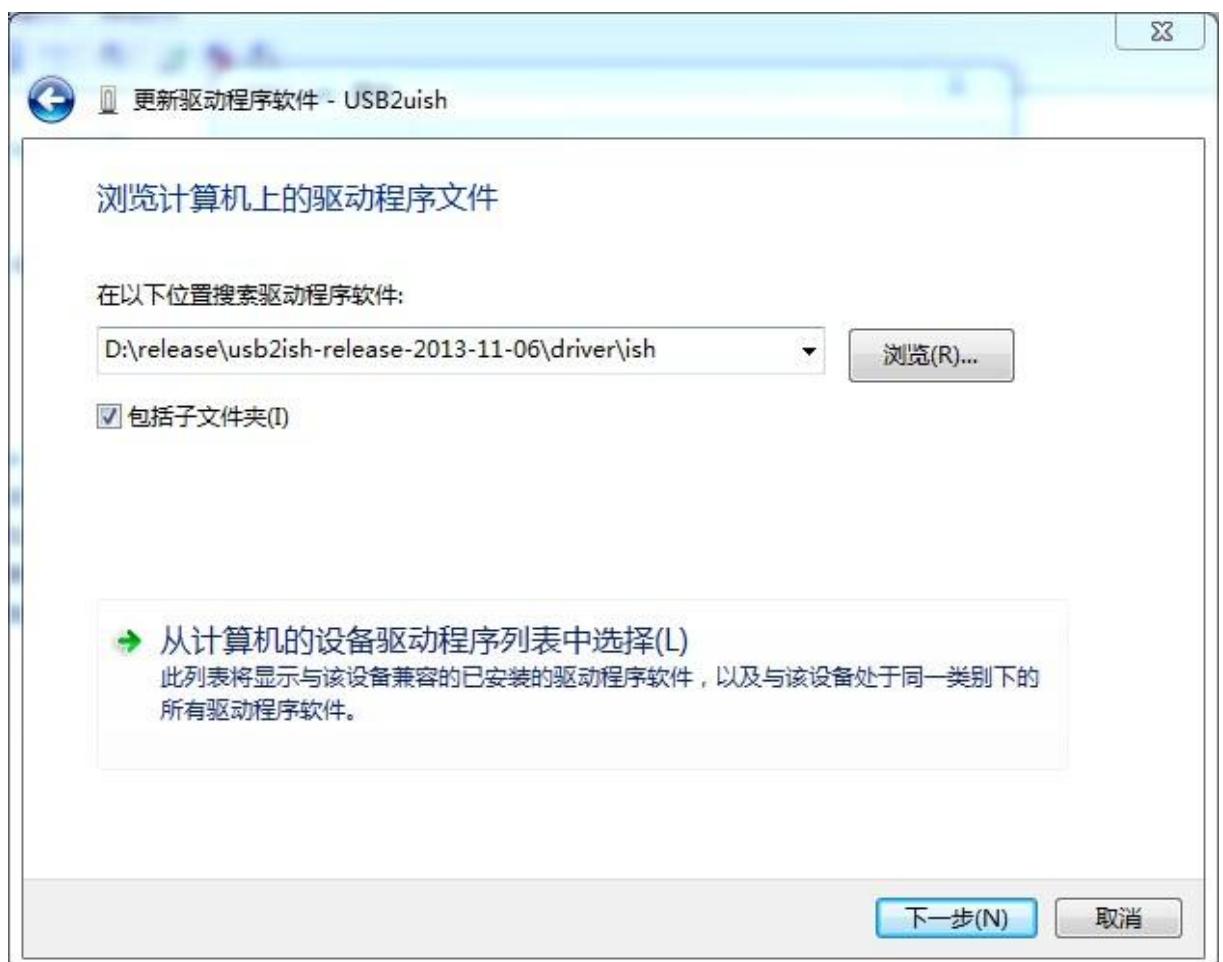


---

C> select “update driver(p)...”, the dialog box of update driver pops up, as shown below.

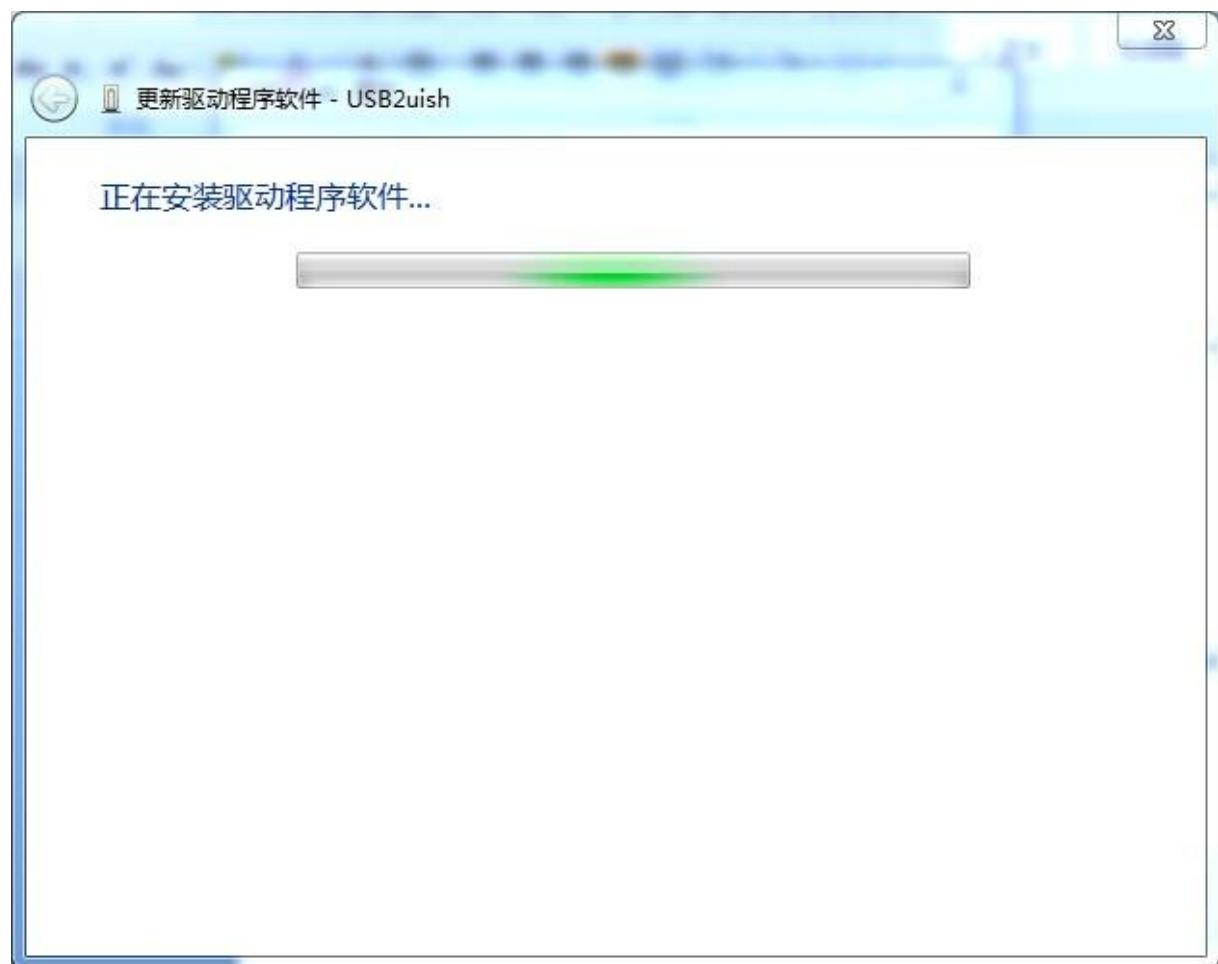


D> Select “ browse the computer to find the driver software(R)”, click next , as shown below.



---

E> Click “ browse(R)”, select the driver file to download the folder where DRIVER\ish\(window platform)\usb2ish.inf is located, click on the next step, start losding the driver, as shown below.





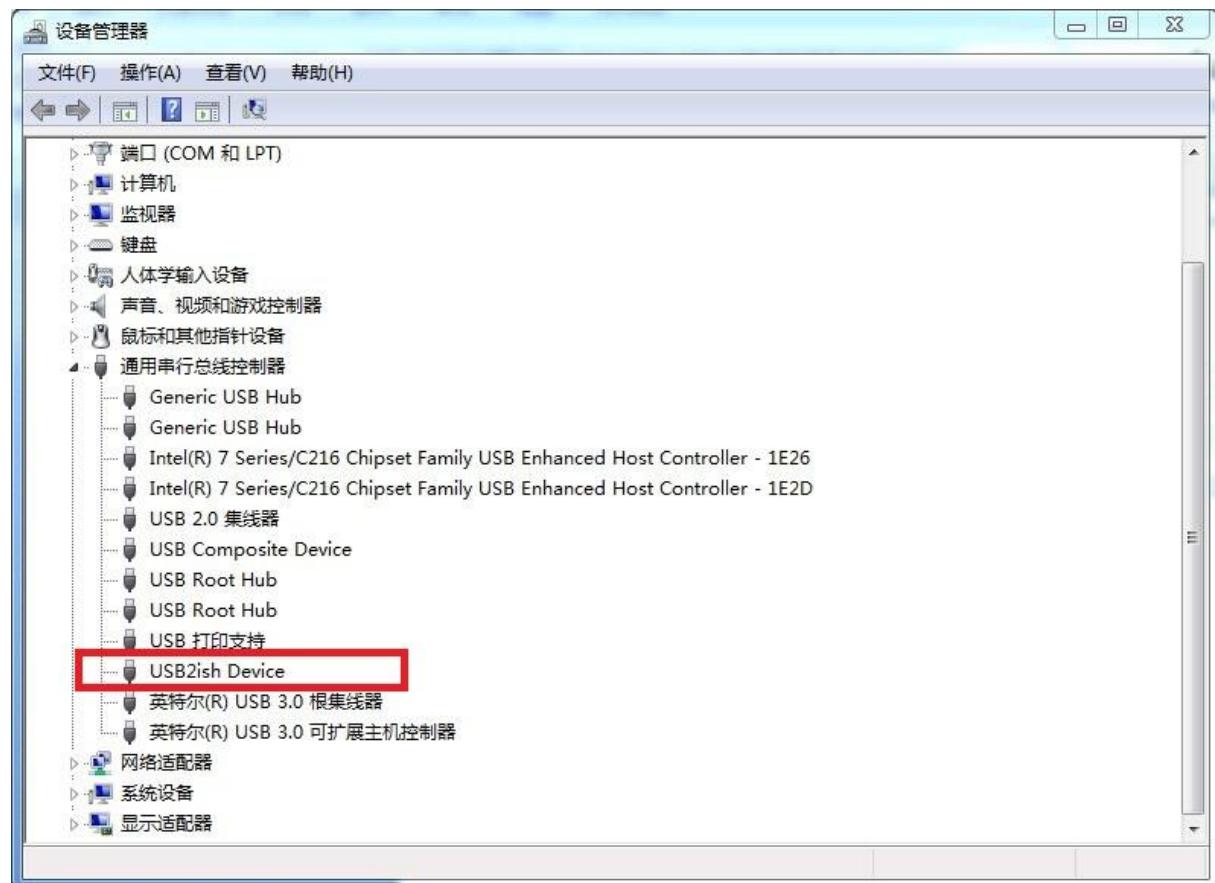
F> After loading, select close , back to the tab of the USB2ish device property driver, and you can see the information on the device driver installation and company digital signature, as shown below.



G> You can see the unique serial identification number of the device on the USB2ish device properties details tab,as shown below.

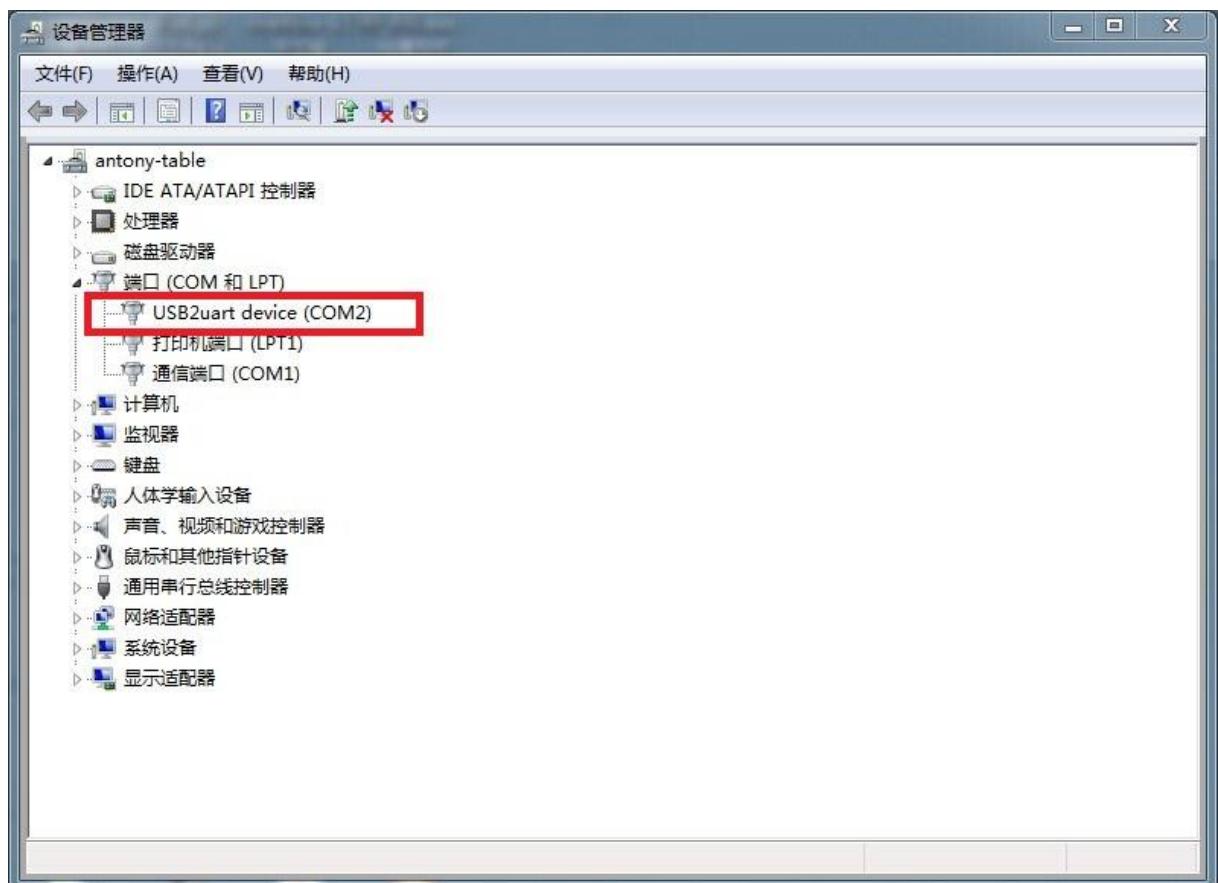


H> Back to the device manager, open the triangle symbol in front of the universal serial bus controller port, a new device can be seen , which proves that the device is installed , as shown below.



#### 4.2> Install the device of USB2Uart

Steps are similar to ABCDE of 1>( Function jumper should be changed to 1-2),only but in step D to choose the driver file is the folder where DRIVER\ish\(window platform)\usb2ish.inf is located. After installing, open the system properties->hardware->device manager, open the triangle symbol of the port( COM and LPT), and one more new port can be seen, as shown below.



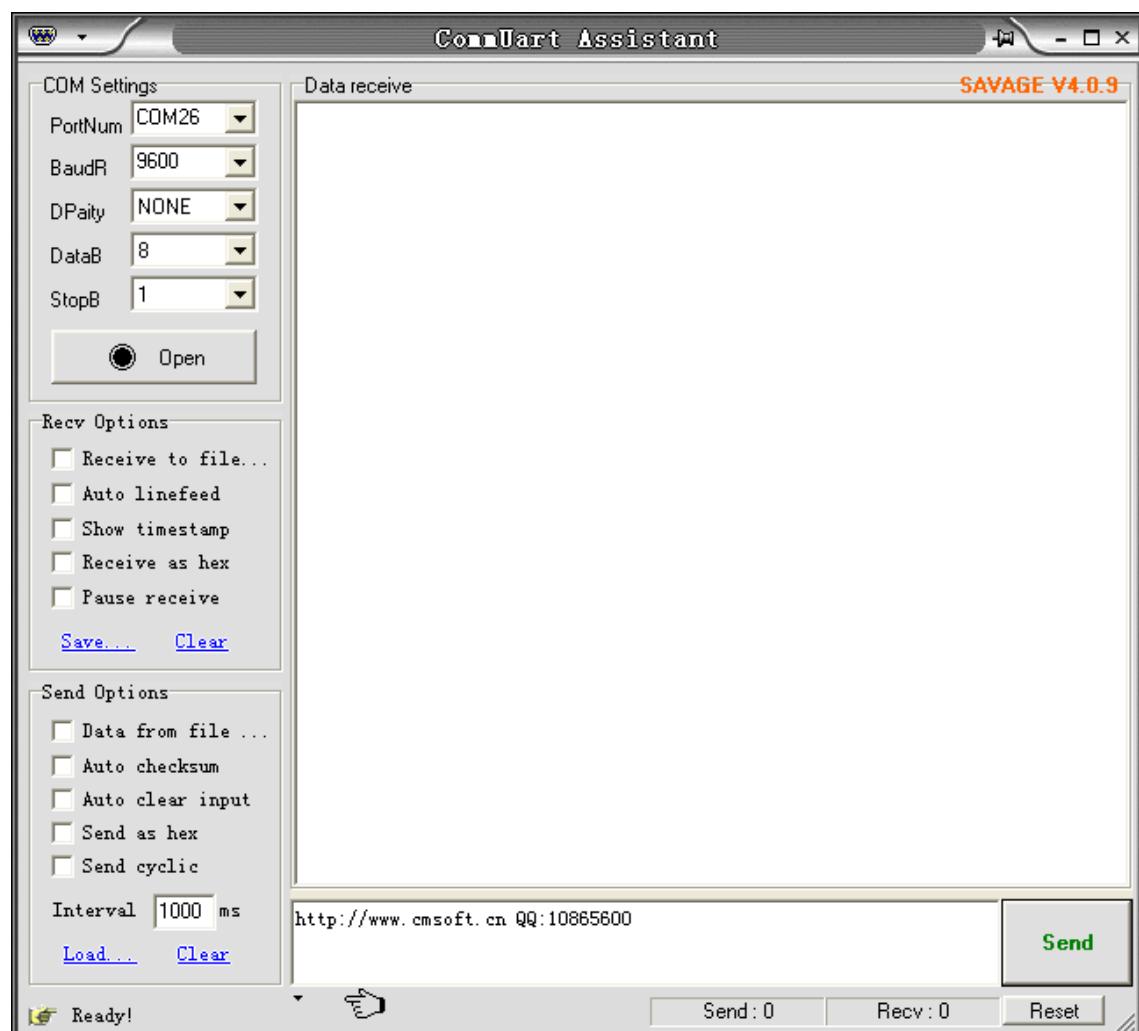
## 5. Usage in the window system

### 5.1> Use UART

Make sure that the function jumper is 1-2 short, and the upgrade jumper is 2-3 short.

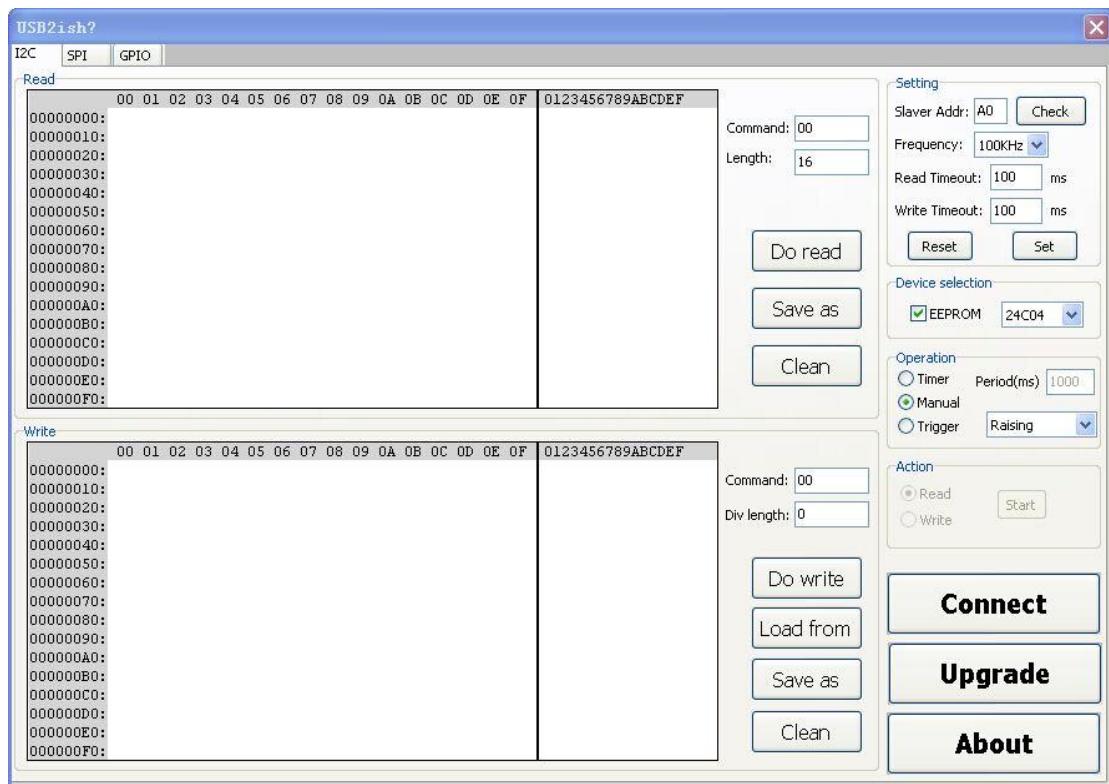
---

Enter the download folder “other” and Open the software “UartAssist.exe”, and select com , you can implement the Com communication. As shown below.

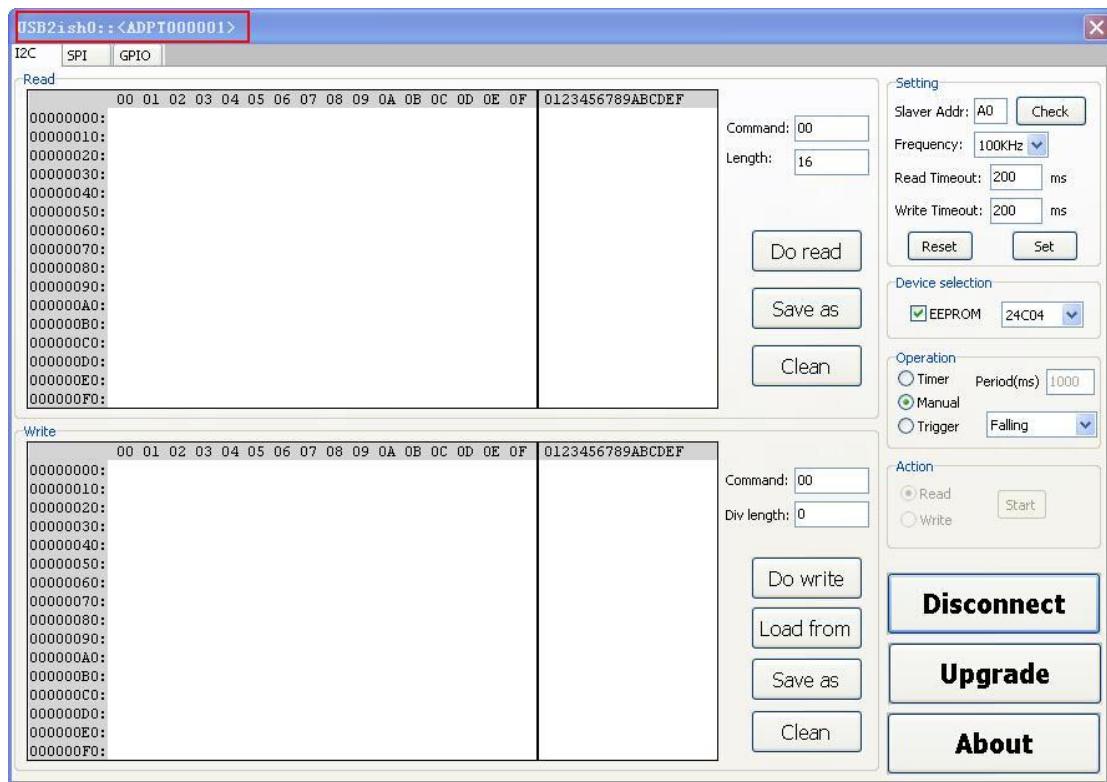


## 5.2> Use I2c

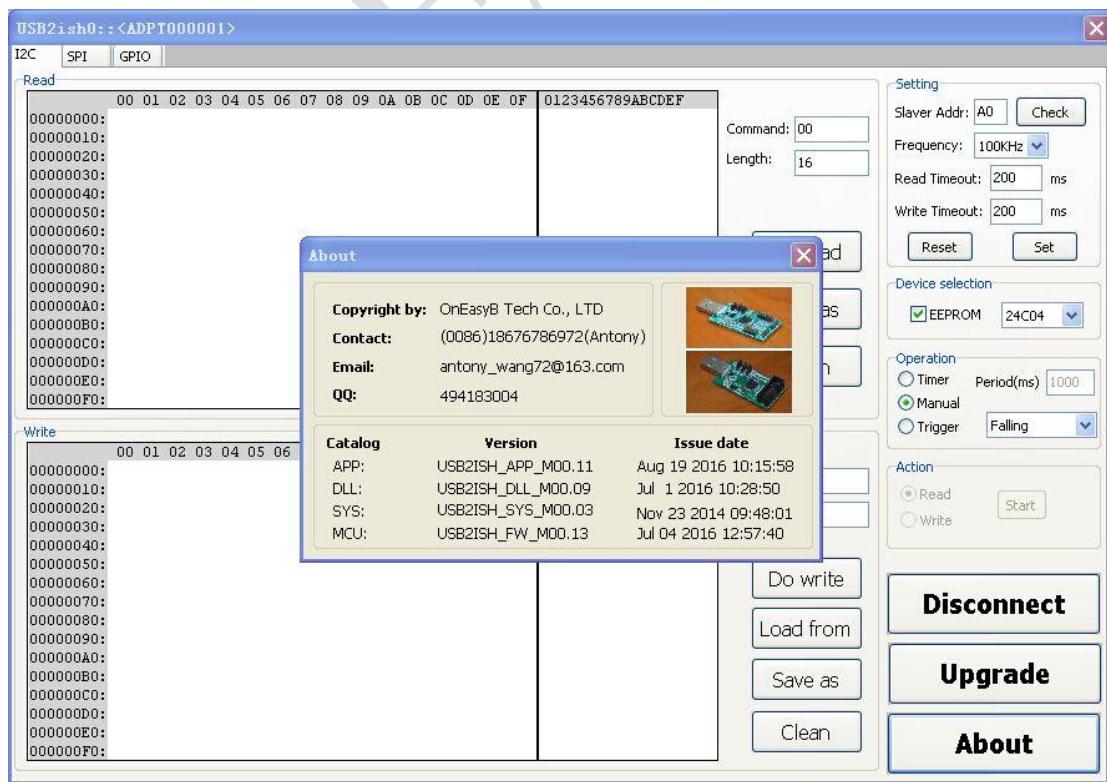
Make sure that both of the function and upgrade jumper are 2-3 short. Enter the download folder “app” and run the software “usb2ish\_Pro\_en.exe” as administrator, as shown below.



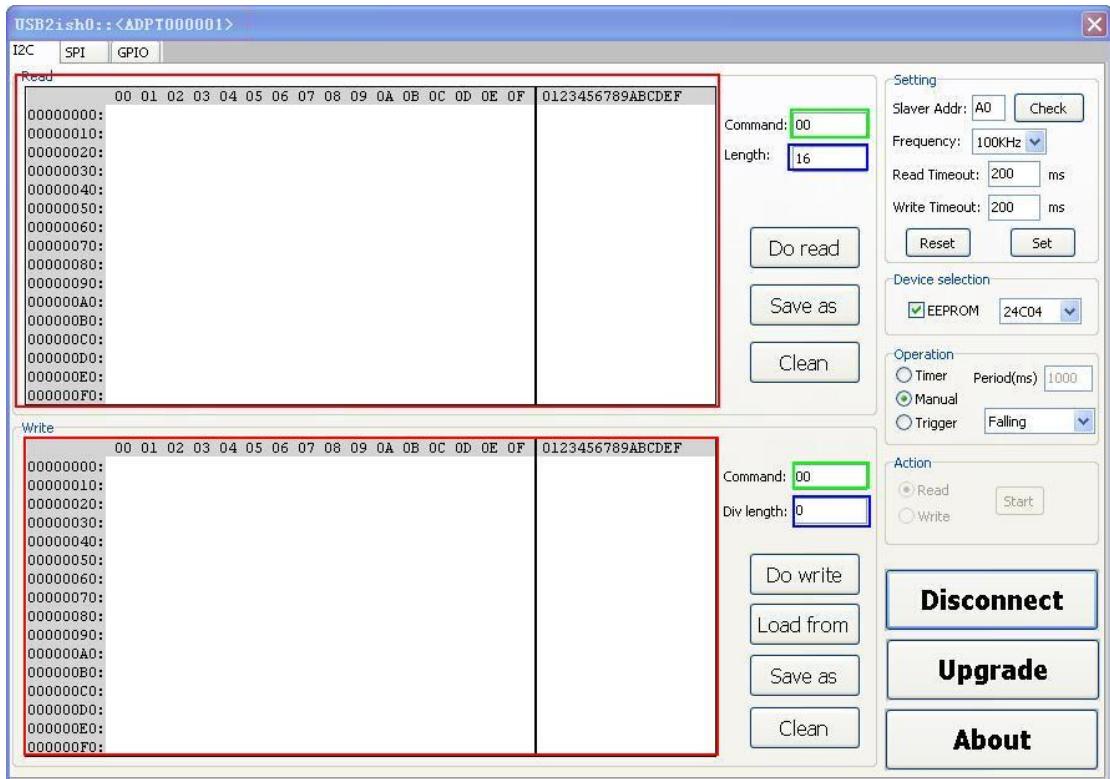
Now device is disconnected, please plug the USB2UIS board and click the “Connect” button, you can see the changes in the window title after successful connection,  
there are device reference number and device serial number more, as shown below.



Continue to click the “About” button, you can see the information of the various software , as shown below.



After successful connection , please switch to I2c page, as shown below.



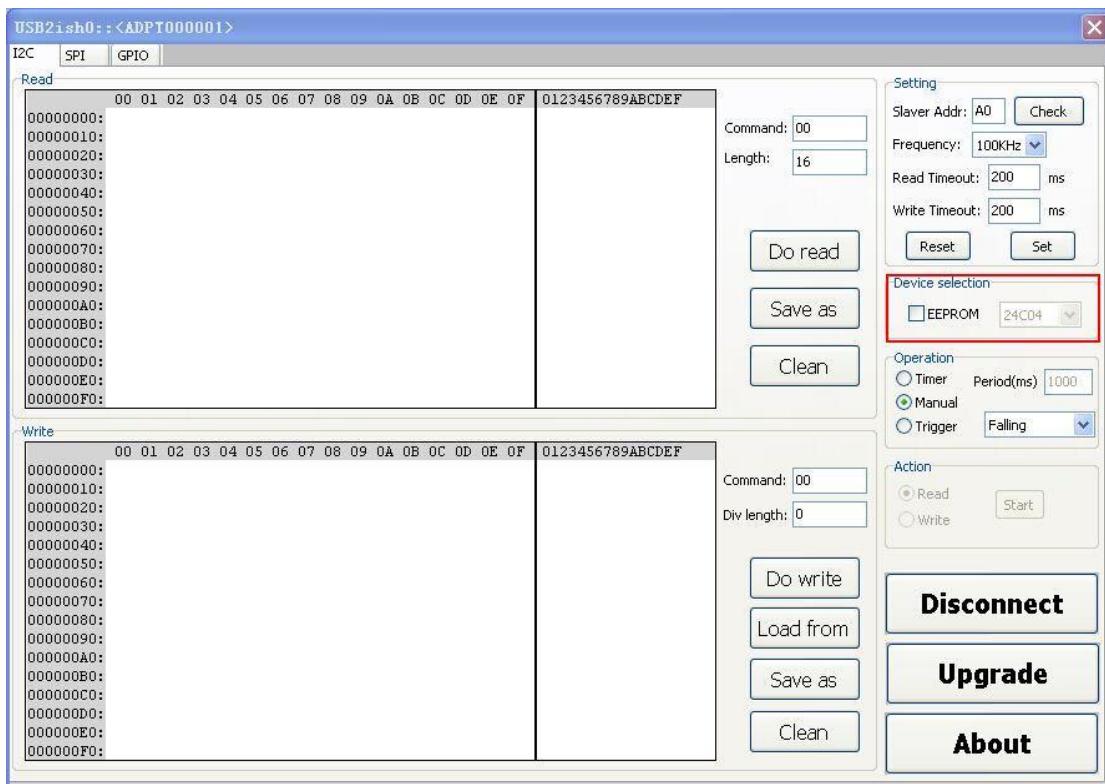
The red area is the data box, used to display the data of read and write, which is divided into two parts. On the left is the hex format mode and right is the text format mode.

The green box , which is the input box of the hex format , is used to fill in the device's command parameters or register address. It can be empty and determined by i2c device.

The blue box is used to fill in the length of data and must enter the decimal digit.

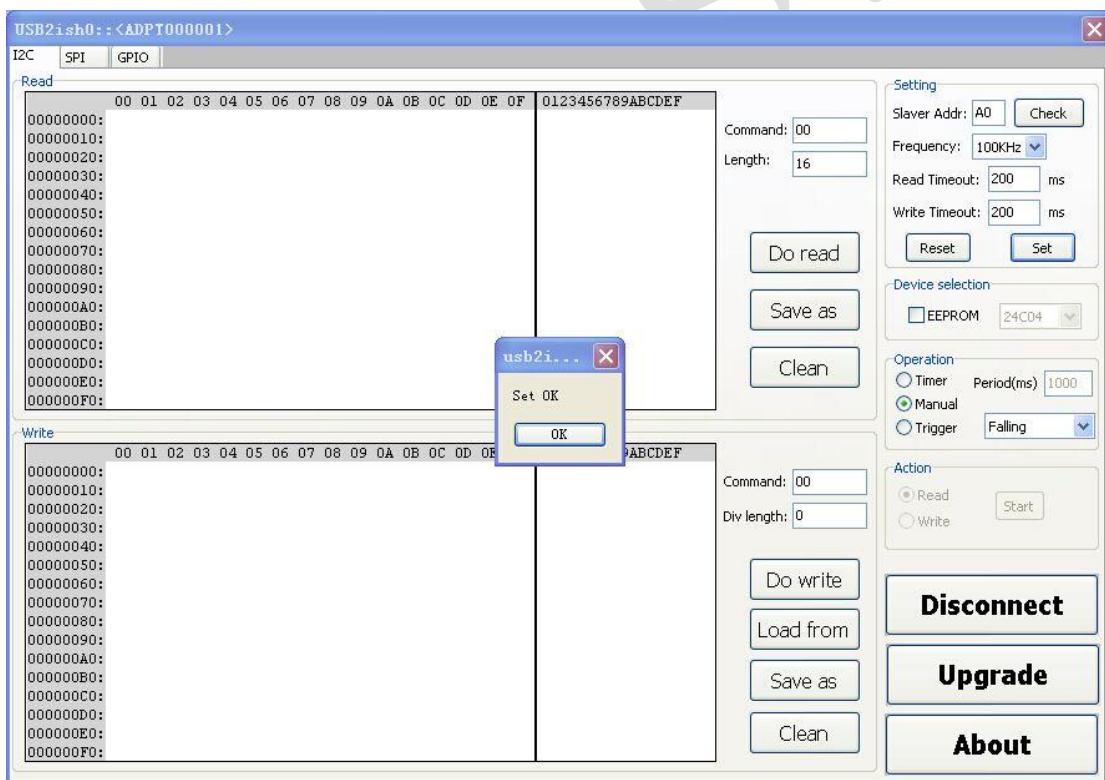
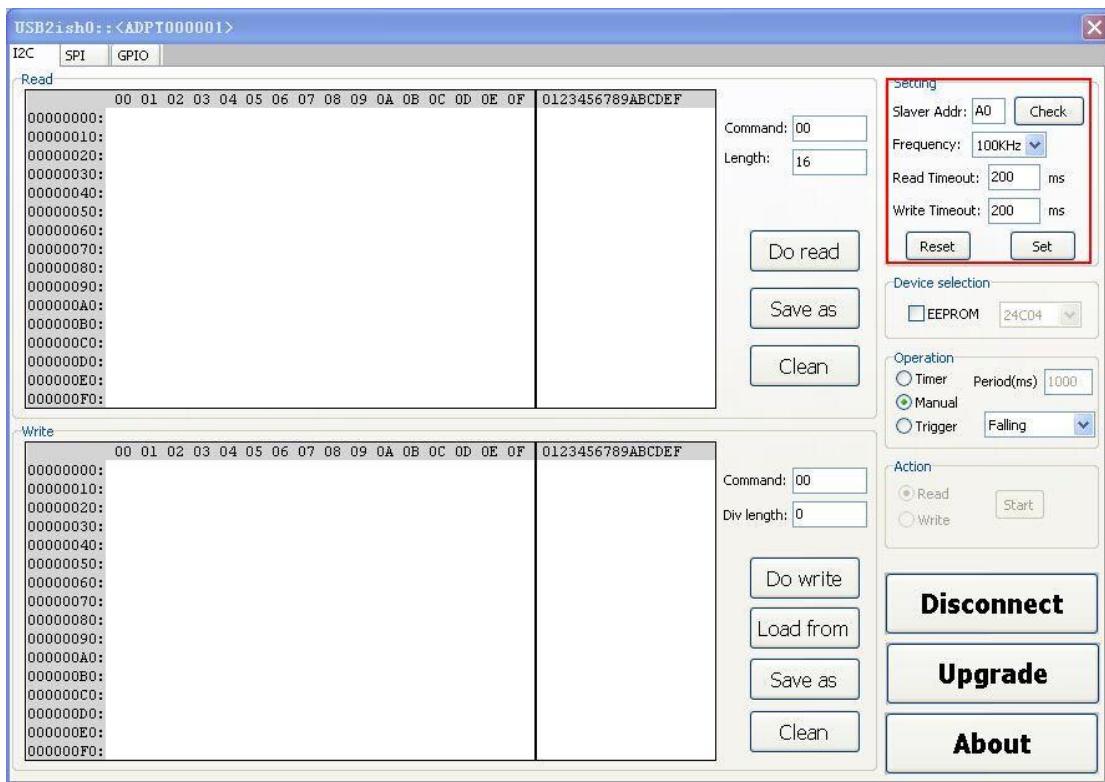
### 5.2.1> Select the device

The default device selection is EEPROM, if the device of read or write is not EEPROM, you must de-check the checkbox of EEPROM. As shown below.



### 5.2.2> Setting

Before starting the read or write operation, please set up the device address of I2c, communication frequency and the read and write timeout parameter. Press the “set” button until the OK dialog box pops up , which proves that device address is valid. Press the “reset” button to initial settings. If you don’t know the device address, you can click the “Check” button, if slave device address found, it will replace the previous one. Then click the “set” button until the OK dialog box pops up to prove device address is valid. As shown below.



### Cautions:

- The edit-box of the device address is 16 hexadecimal characters , up to 2 characters, the default is A0.

- The bit7~bit1 of value of the edit-box is the 7 bit device address and the bit0 is for read/write bit, must fix to 0.
- After the address is set up successfully , the read or write operation is not need to set up repeatedly without powering off.

### 5.2.3> Operation choice

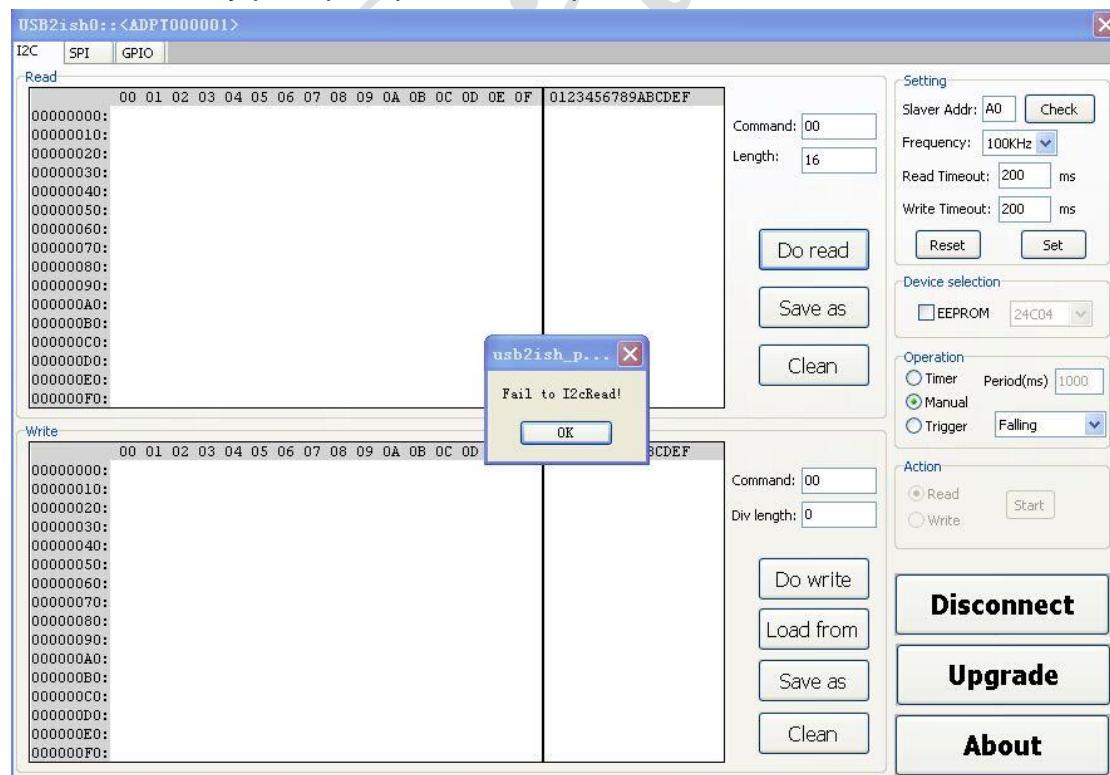
There are three operation models available in operation choice.

- Timing operation: You can do Read and write operation at a specified time interval.
- Manual operation: You must do an operation one time by clicking the “Do read” or “Do write” button.
- Trigger operation: Do the read or write operation when you detect the level changes of IRQ pin, this function only is available for extended type.

Whether it is timing, trigger or manual operation, you only can do read or writer operation.

### 5.2.4> Read and write

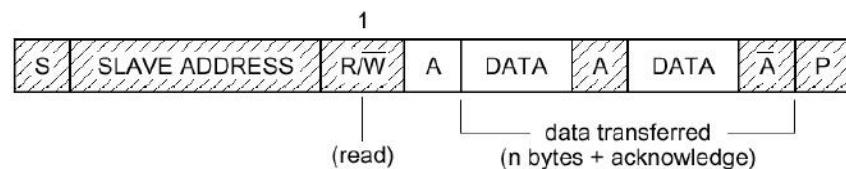
Whether it is read or write, it will pop up the dialog box of read or write failure. If there is no any prompt, it proves the operation successful. As shown below.



The access to I2c device has three kinds of formats. Which mode to use, that is determined by the connection of I2c device.

### 1> Immediate read mode

This pattern is not commonly used, the format is as below.



from master to slave

A = acknowledge (SDA LOW)

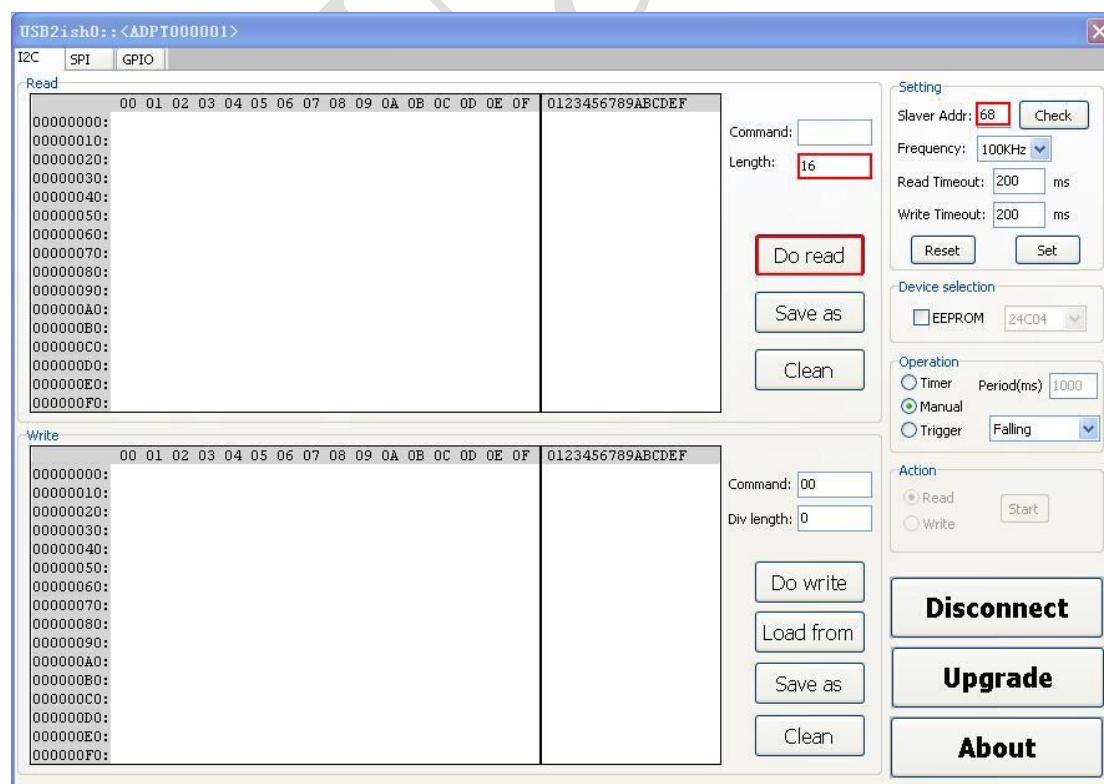
from slave to master

A = not acknowledge (SDA HIGH)

S = START condition

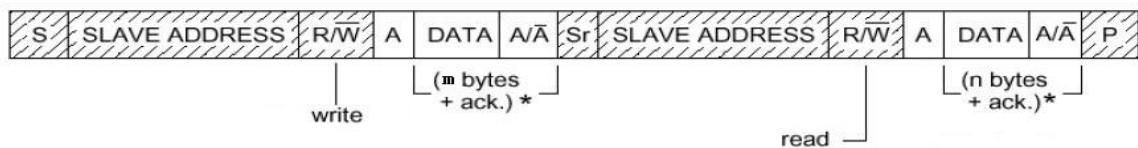
P = STOP condition

For example: the device address is 0x68, n=16, immediate read data is 16 bytes. At first , the device address is set as 68, click the “set” button , next you should fill 16 in the length of read-in , leave the command parameters empty. And then click the “Do read” button. As shown below.



## 2> Compound read mode

This kind is the most read mode, the format is as below.



from master to slave

from slave to master

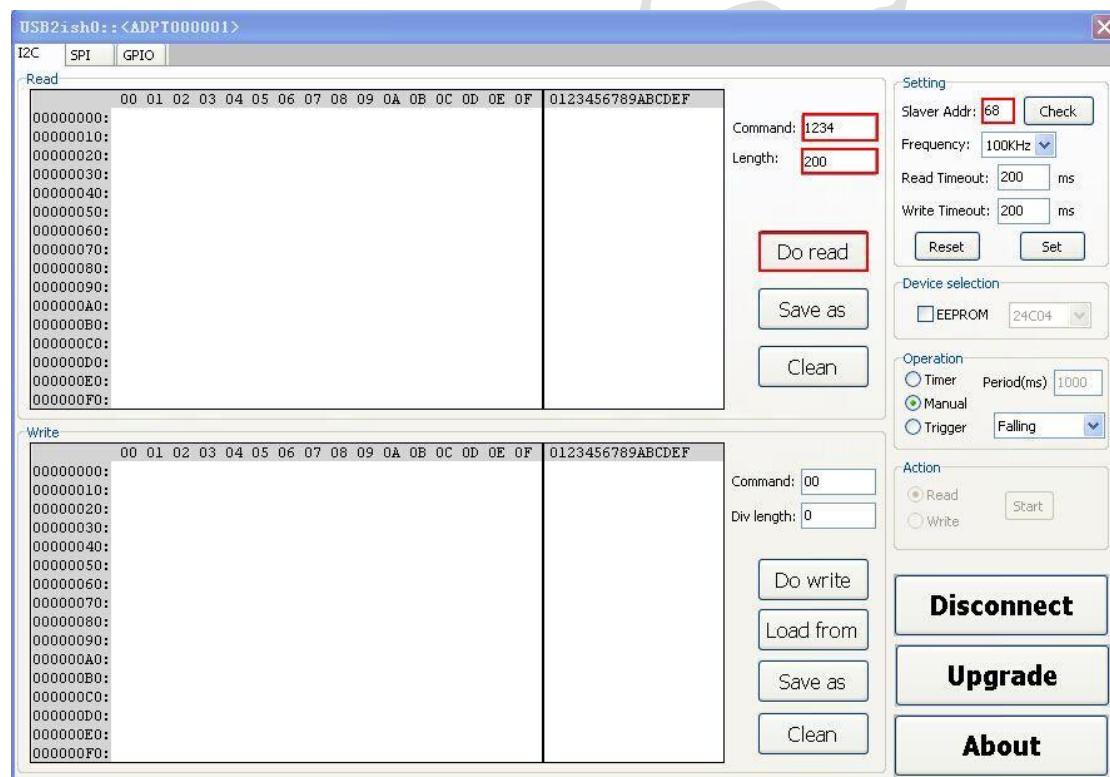
A = acknowledge (SDA LOW)

A = not acknowledge (SDA HIGH)

S = START condition

P = STOP condition

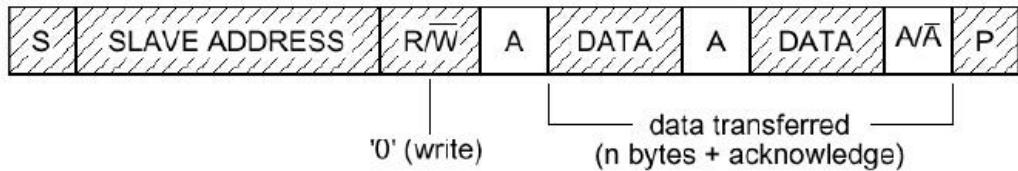
For example: the device address is 0x68, m=2, n=200, the data of area m is 0x1234. Following the figure below.



Click the "Do read" button , that is the execute Compound read operation.

### 3> Write mode

The format is below.



from master to slave

from slave to master

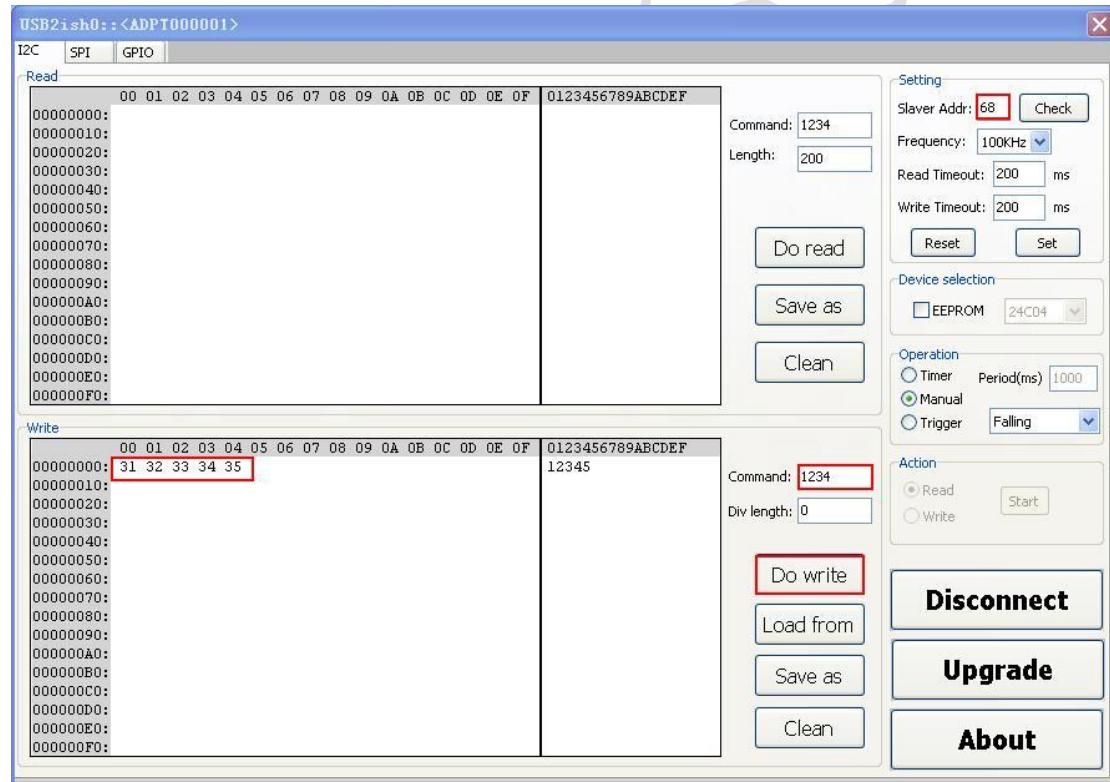
A = acknowledge (SDA LOW)

A = not acknowledge (SDA HIGH)

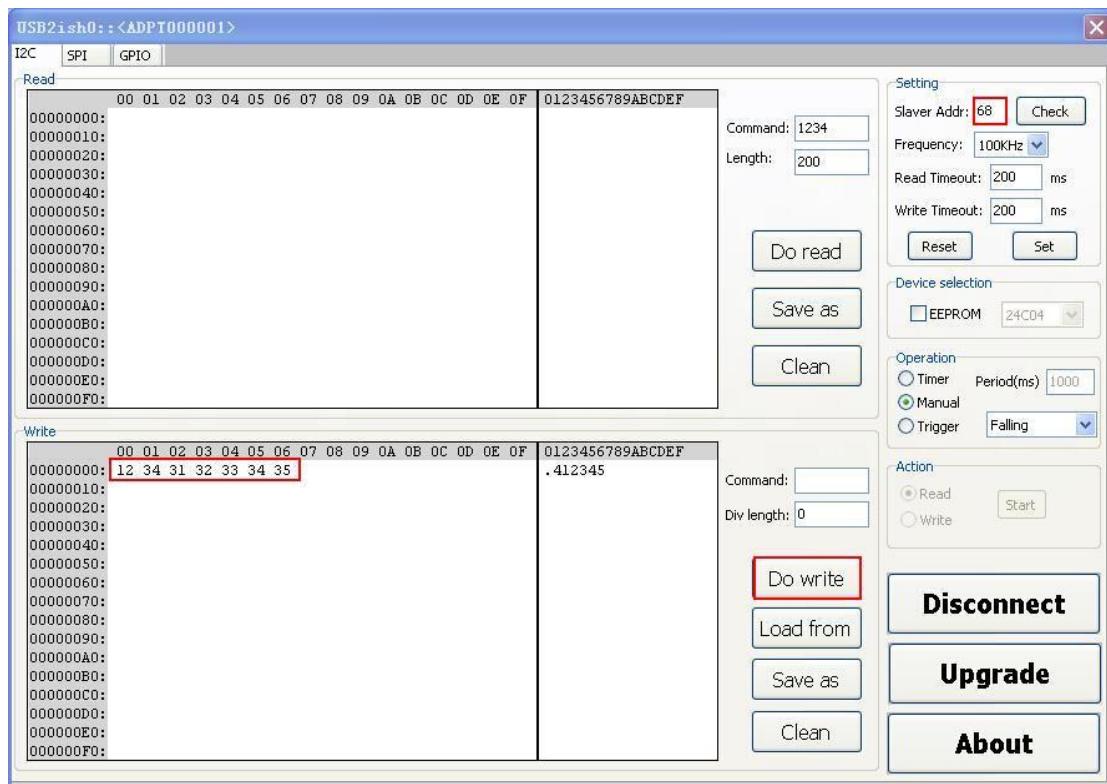
S = START condition

P = STOP condition

For example : Writer 0x31, 0x32, 0x33, 0x34, 0x35 to 0x1234, following the figure below.



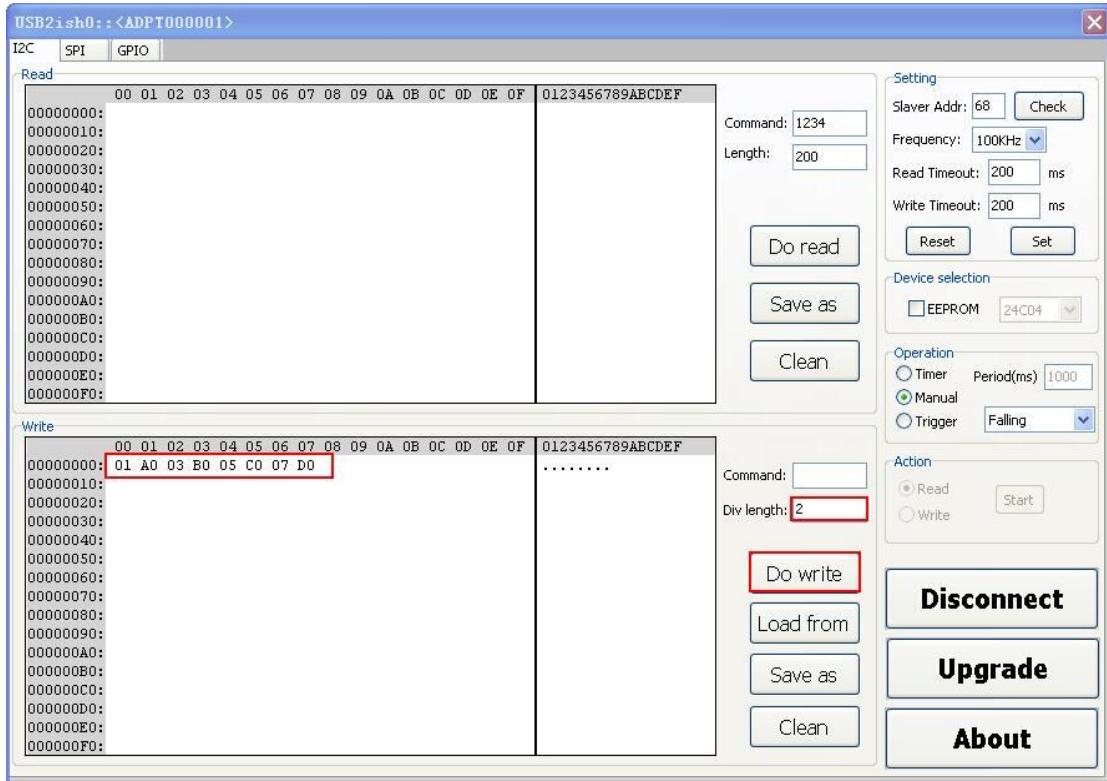
Or



It is the same effect after clicking the “Do write” button.

### 5.2.5> Special function( subsection)

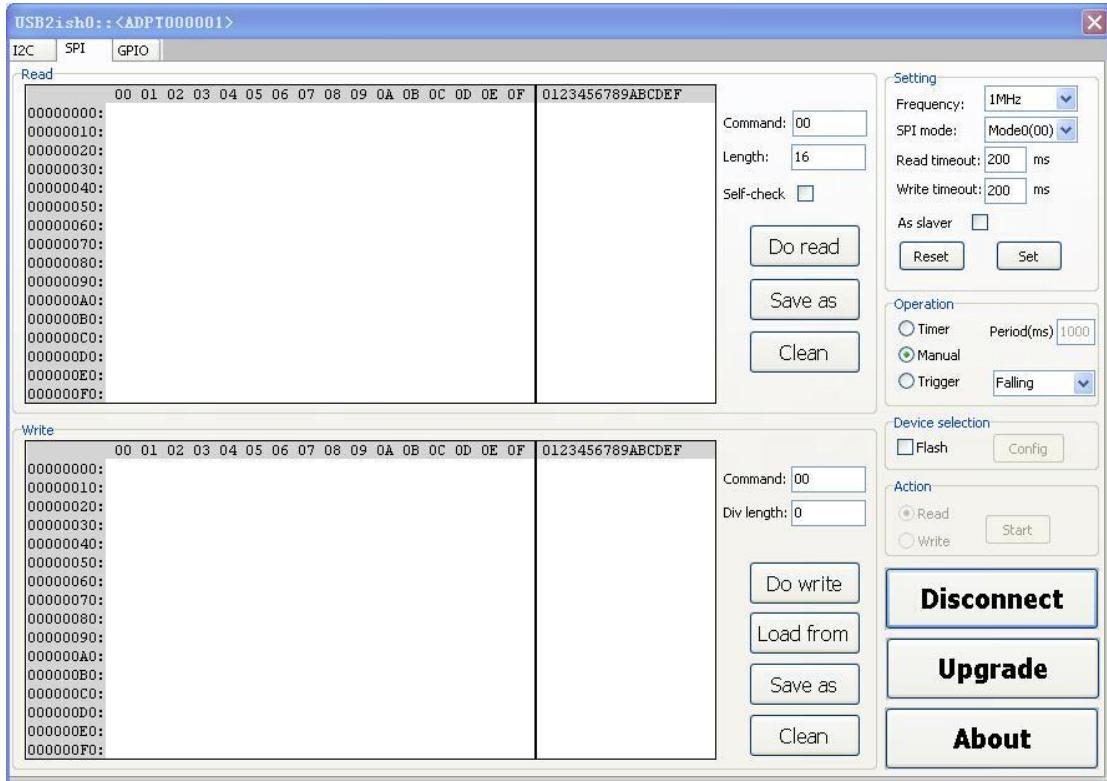
For example, when you write data 0xA0, 0xB0, 0xC0, 0xD0 to register respectively 0x01,0x03,0x05,0x07, you can use the function of subsection . Firstly set Div length as 2, write configuration as shown below.



Finally click the “Do write” button to finish the operation of write by subsection.

### 5.3> Use SPI

Make sure that both the function jumper and the upgrade jumper are 2-3 short. Enter the download folder “APP” and run the software of ‘usb2ish\_Pro\_en.exe.’ as administrator Please click the tag if it is not the SPI page, as shown below.

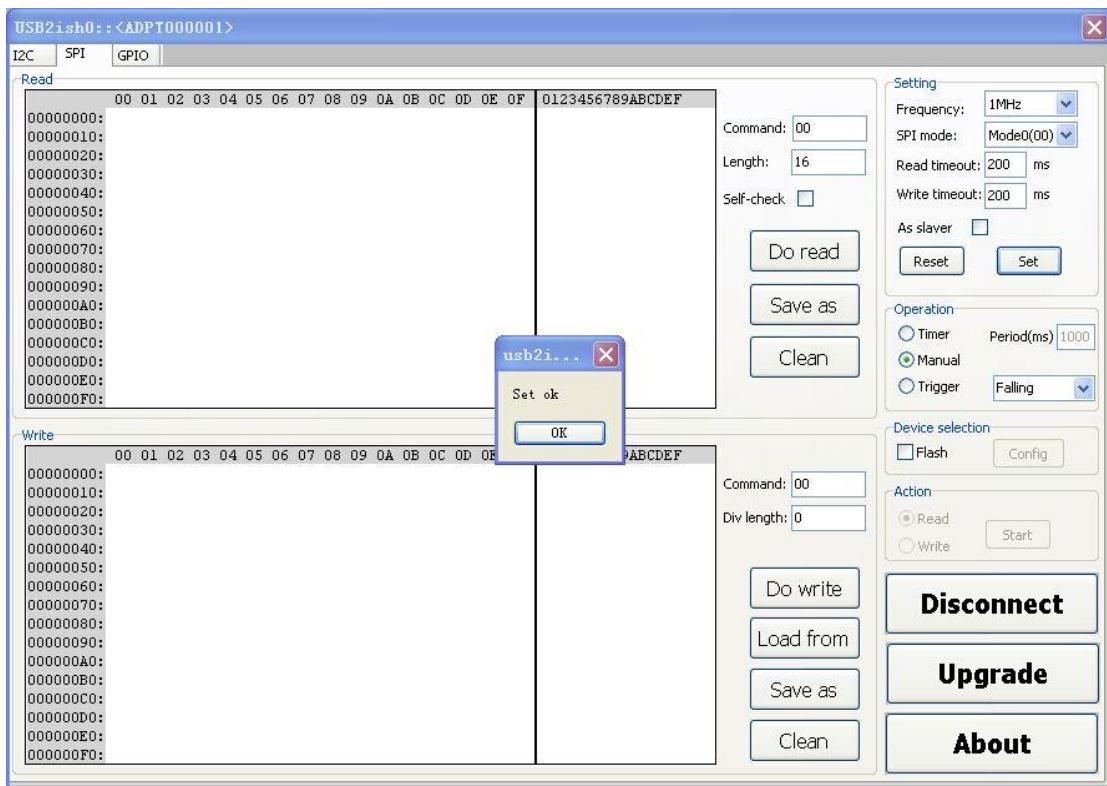


About the input format of each box , please refer to I2C description.  
 SPI can work as master or slaver. The default is master.

### 5.3.1> Work as master

#### 5.3.1.1> Setting

Before the operation of read or write , please set the communication frequency ,mode and the timeout parameters of read and write. It proves successful if the ok dialog pops up by clicking the “set” button. Clicking the “reset” button to initial settings. As shown below.



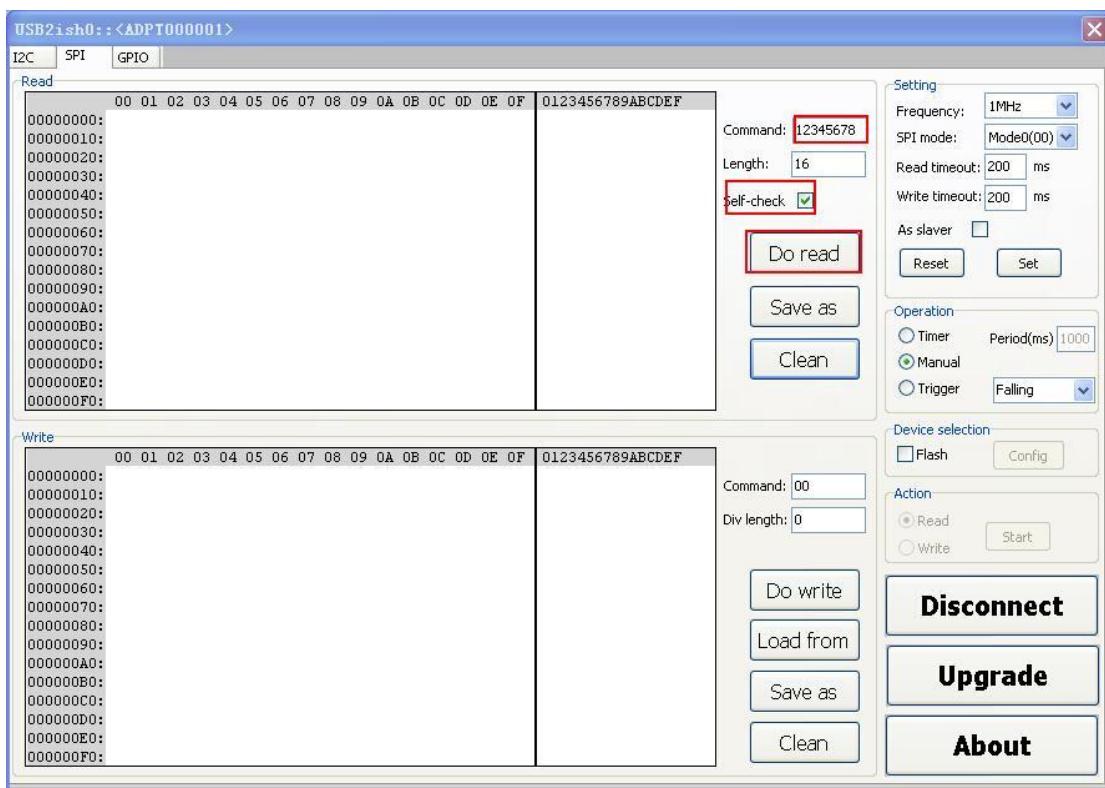
### Description of SPI mode

- Mode 00: SCK is low level in the idle state, the first edge sampled data in the SCK cycle.
- Mode 01: SCK is high level in the idle state, the first edge sampled data in the SCK cycle.
- Mode 10: SCK is low level in the idle state, the second edge sampled data in the SCK cycle.
- Mode 11: SCK is high level in the idle state, the second edge sampled data in the SCK cycle.

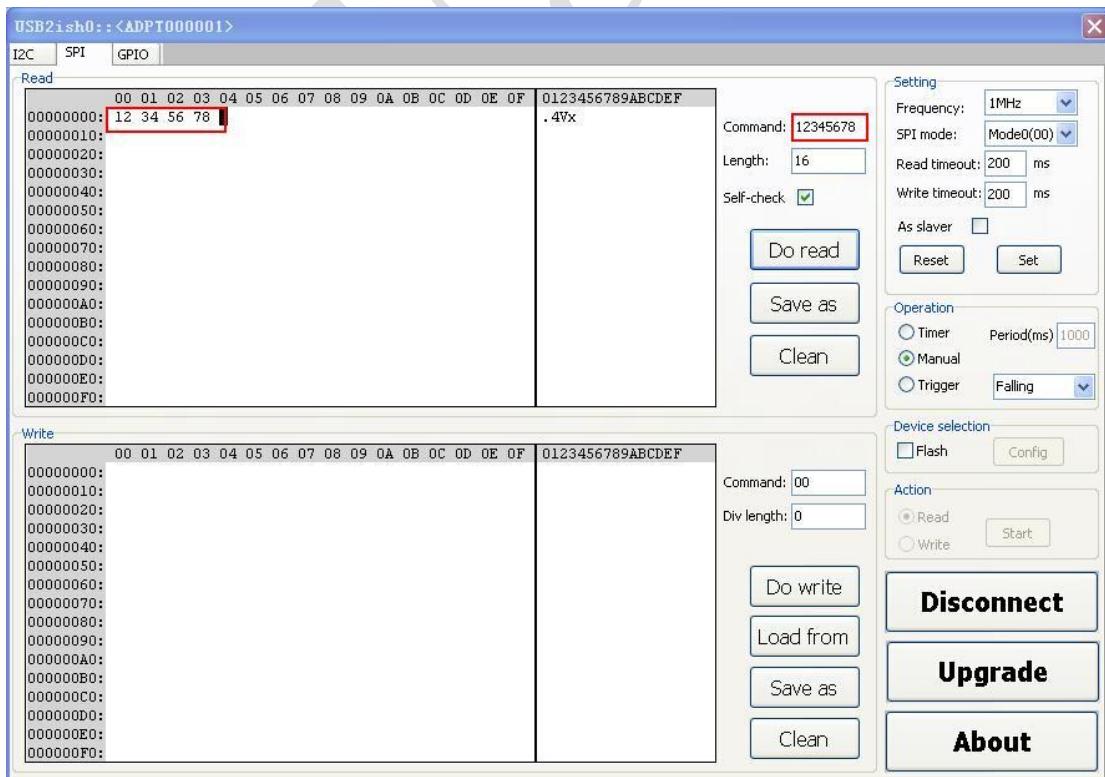
Send order of SPI  
MSB Priority

#### **5.3.1.2> Self-checking**

The pin of MISO and MOSI of SPI are shorted together, tick the “self-check” and fill the command parameters of “12345678” as shown below.



Then click the “Do read” button, the data will displayed in the content box which is the same as in the command box, this proves that the function of SPI is normal .As shown below.



---

### **5.3.1.3> Operation choice**

There are three operation models available in operation choice.

- Timing operation: You can do Read and write operation at a specified time interval.
- Manual operation: You must do an operation one time by clicking the “Do read” or “Do write” button.
- Trigger operation: Do the read or write operation when you detect the level changes of IRQ pin, this function only is available for extended type.

Whether it is timing, trigger or manual operation, you only can do read or writer operation.

### **5.3.1.4> Read and write operation**

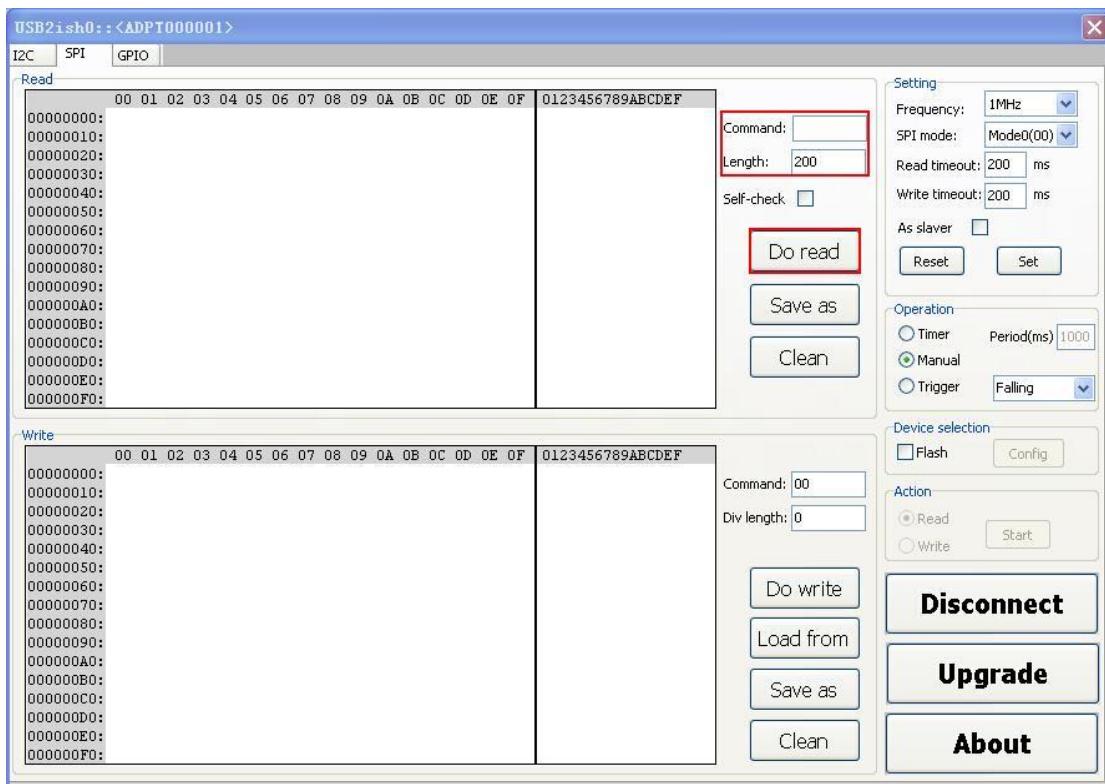
The read and write of SPI has three patterns.

#### **1. Read mode without command parameter**

As following table:

BYTE	First	Second	Third	.....	Last
MOSI	0x00	0x00	0x00	.....	0x00
MISO	Data1	Data2	Data3	.....	DataN
CSN	LOW				

Such as N=200, Read 200 byte through SPI, read configuration, as shown below.



Then click the “Do read” button .

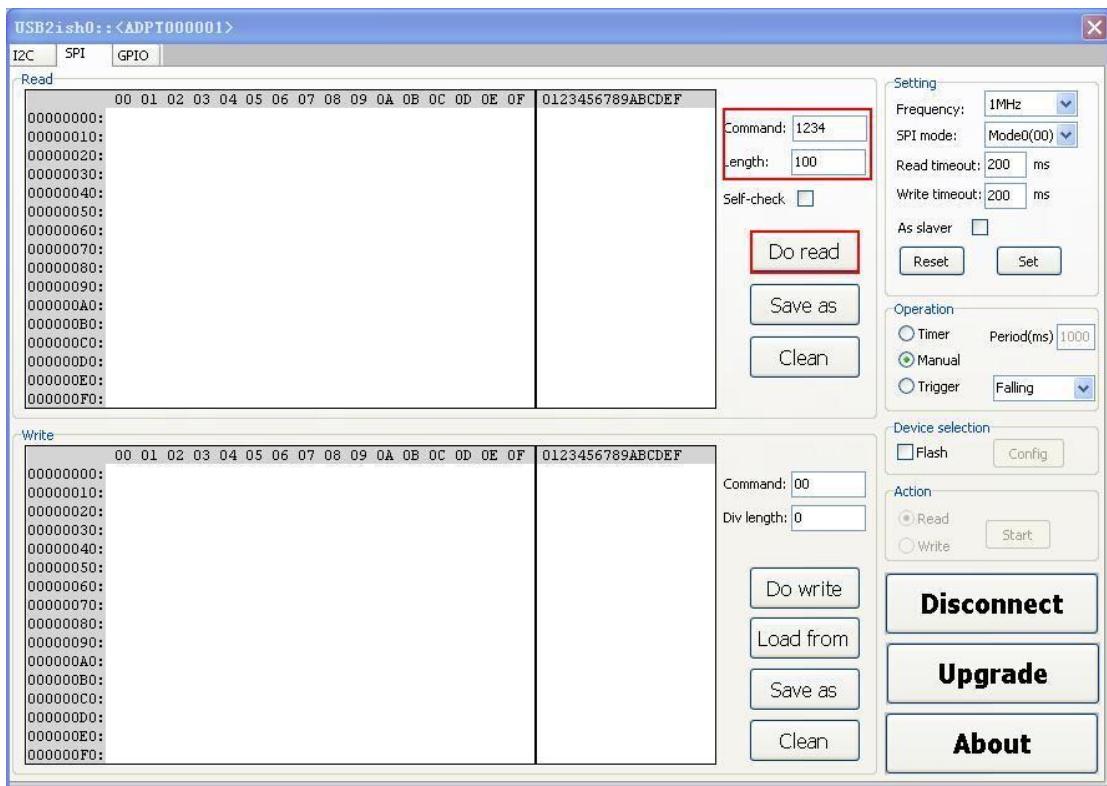
## 2. Read mode with command parameter

As shown below .

BYTE	First	Second	Third	.....	Last
MOSI	COMMAND1	COMMAND2	0x00	.....	0x00
MISO	Don't care	Don't care	Data1	.....	DataN
CSN	LOW				

Here the longest of COMMNAAD can be 4 byte.

If command is 0x1234, read 100 byte through SPI , read configuration, as shown below.



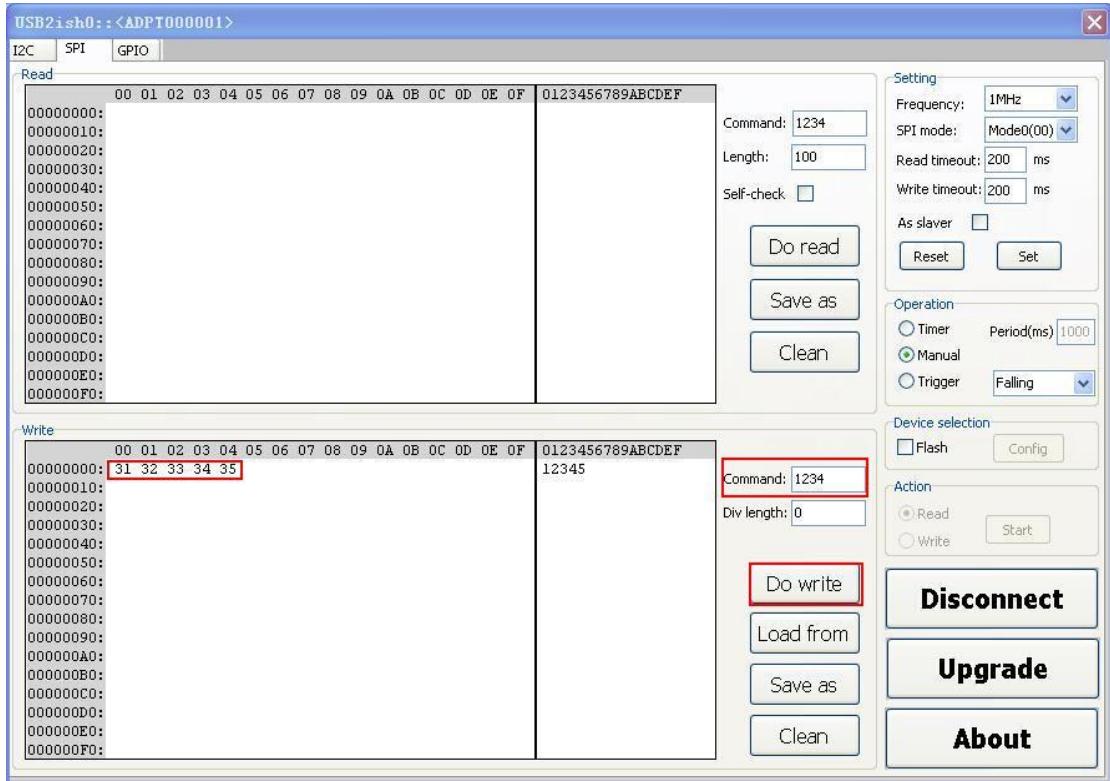
Then click the “Do read” button.

### 3. Write mode

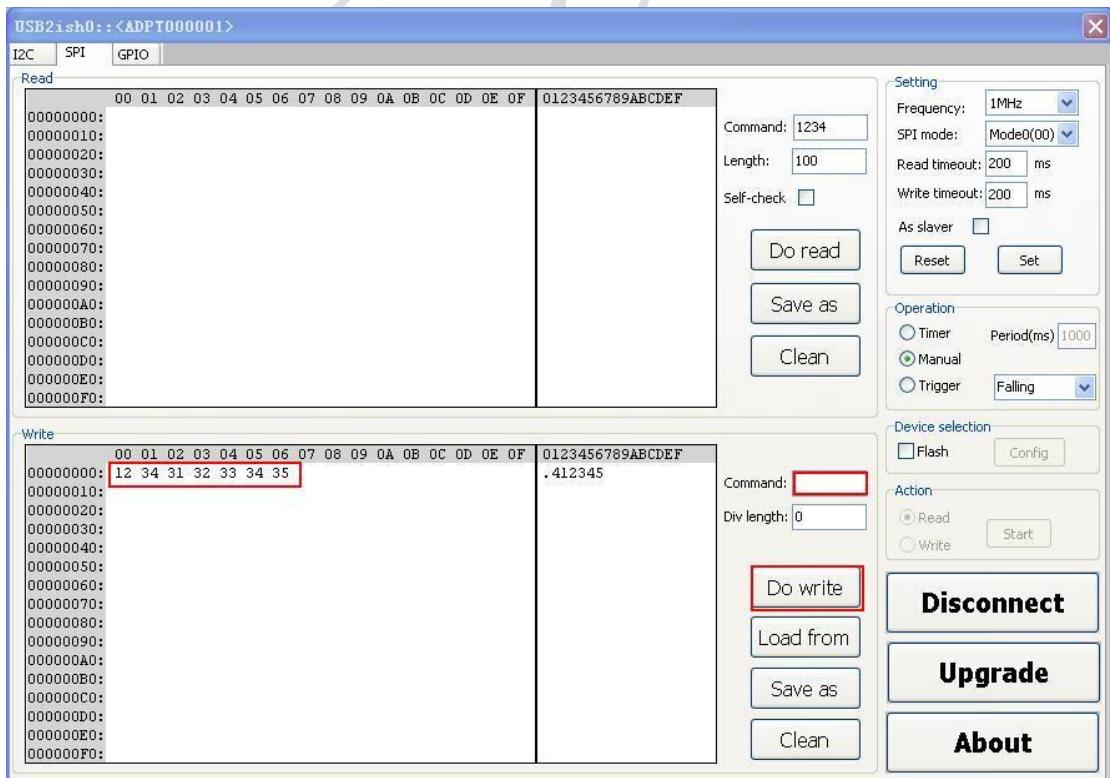
BYTE	First	Second	Third	.....	Last
MOSI	COMMAND1	COMMAND2	DATA1	.....	DATAN
MISO	Don't care	Don't care	Don't care	.....	Don't care
CE	LOW				

Here the longest of COMMNAD can be 4 byte.

If command is 0x1234, the data of read-in is 0x31,0x32,0x33,0x34,0x35, read configuration, as shown below.



or



---

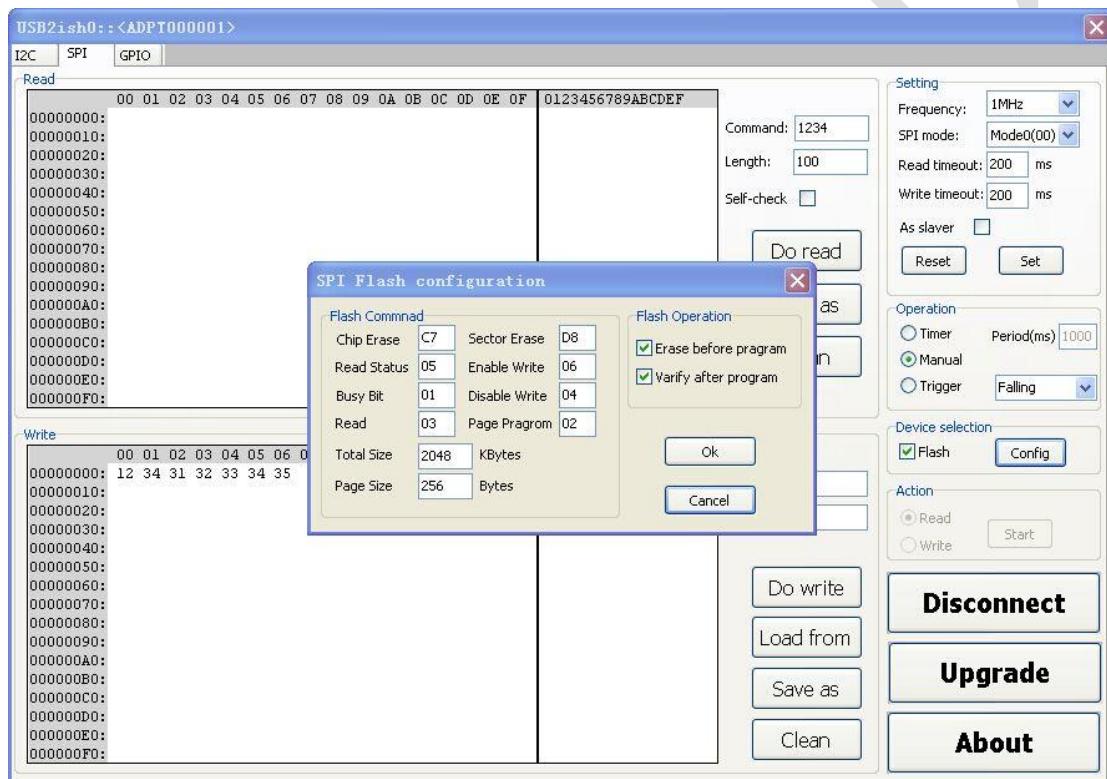
It is the same effect after clicking the “Do write” button.

### 5.3.1.5> Special function

SPI also has the function of write Special by subsection same as I2c. For operation step , please refer to the subsection of 12 c .

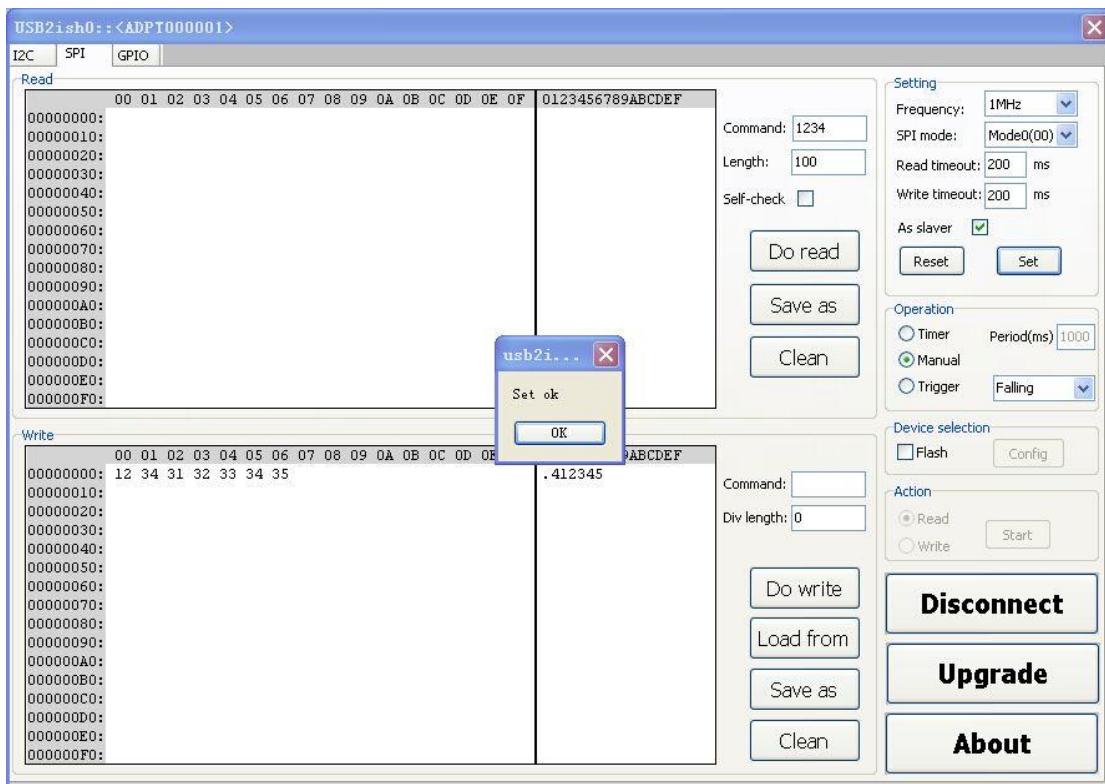
### 5.3.1.6> Read and write of SPI flash

Before the read or write of SPI flash device, you should tick flash,then click the “Config” button to finish the configuration of flash. the command parameter of read or write area is the offset address of SPI flash. as shown below



### 5.3.2> Work as slaver ( Available for extened type)

To make SPI work as the slaver, you need to tick the “As slaver” checkbox, then click the “set” button until the OK dialog pops up. As shown below.



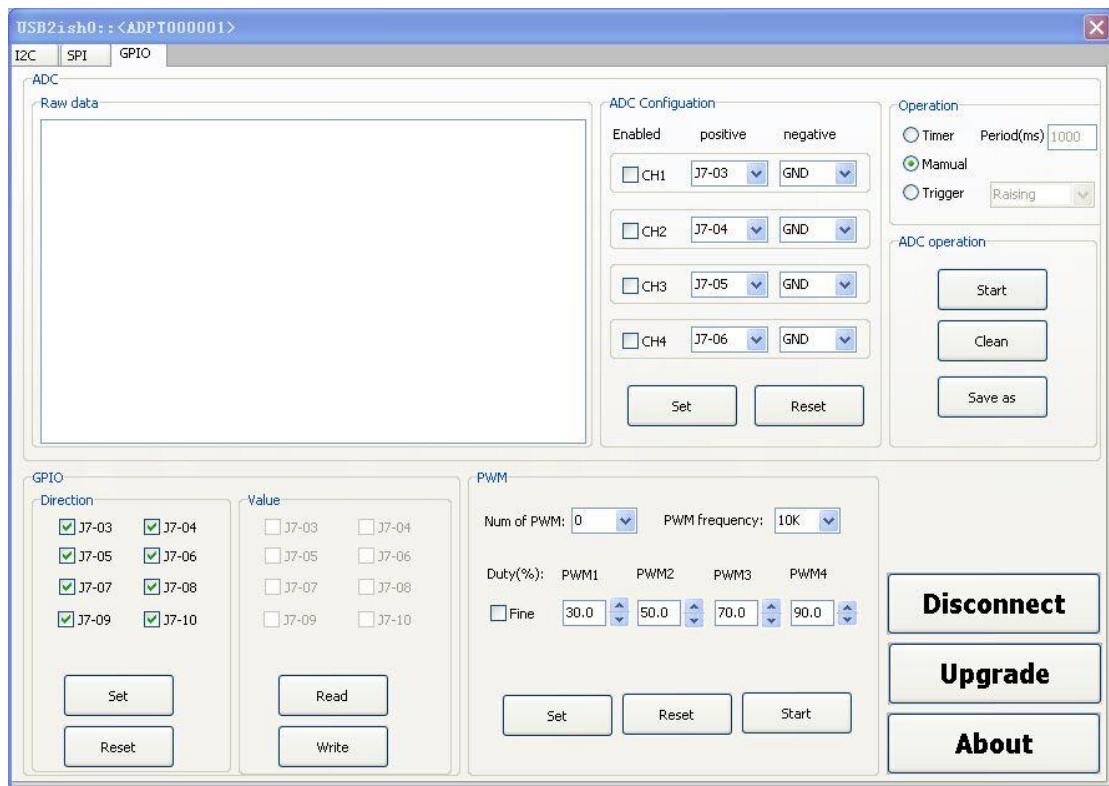
Work as the slaver, the limits of functions is as follows,

- The slaver can only receive the data from MOSI while data of MSIO is not determined.
- The maximum clock rate of the master is not more than 4M.
- The length of single packets sent from the master is not more than 128 bytes.
- The interval of packets sent from the master should be longer than 5ms.

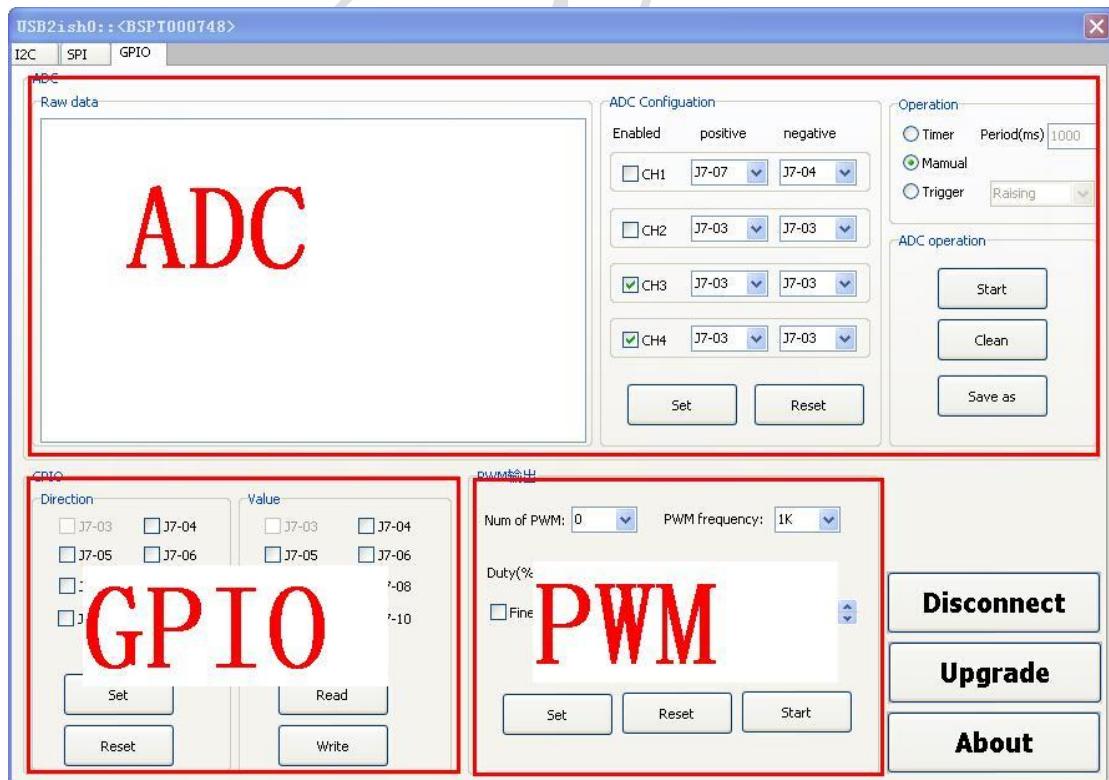
Work As the salver, the received data is displayed in the read data area .

#### 5.4> Usage GPIO ( Available for extened type)

Such as the use of I2c, the connection is successful, switch to the GPIO page, as shown below.



The page function is divided into three parts: ADC, GPIO and PWM, as shown below.



---

#### **5.4.1> Definition of digital IO and the interface of the sampling part.**

Label	silk	功能说明 Function description
J7-3	IO8	ADC sampling channel or digital IO
J7-4	IO7	ADC sampling channel or digital IO
J7-5	IO6	ADC sampling channel or digital IO
J7-6	IO5	ADC sampling channel or digital IO
J7-7	IO4	ADC sampling channel or digital IO
J7-8	IO3	ADC sampling channel or digital IO
J7-9	IO2	ADC sampling channel or digital IO
J7-10	IO1	ADC sampling channel or digital IO

#### **5.4.2> ADC**

10-bit resolution, the maximum sampling rate is 200 ksps, to support the single-ended and differential input. It has the internal reference voltage and a temperature sensor. J7-3 ~ J7-10 can be configured as the positive or negative input for any channel. Up to 4 channels to take sample by turns. When GND is selected as the negative input, the ADC works in single-ended mode , otherwise, it works in differential mode.

##### **1> Usage of sampling**

Tick one or more channels of ADC, select the positive and negative of every channel to input and click on the “set” button, until the OK dialog pops up , after that ,selecting an operation mode of sampling( divide into three modes of manual, timing and trigger), then click the “start” button to start ADC. If ADC is running at the mode of the trigger or timing sampling, the “Start” button title is changed as “Stop”. You can click the “Stop” button to stop ADC. ( The button title has become “Start” at this time). The latest sample value is always displayed in the data box 。It can be stored in the specified place in the form of text file.

---

## 2> Calculate of sample voltage

Define Value as the sample value ,  $V_{REF}$  as the reference voltage, always as 2.44V,

Calculate of Sample voltage  $V_s$  is as below:

$$\text{Single-ended } V_s = \text{Value} * V_{REF} / 1023 \quad (V_s \text{ } 0 \sim V_{REF})$$

$$\text{Differential } V_s = \text{Value} * V_{REF} / 511 \text{ (if bit9-15 of Value is 0)}$$

$$V_s = -(\text{Value} \& 0xFE) * V_{REF} / 511 \text{ (if bit9-15 of Value is 1)}$$

## 3> Calculate of internal temperature sensor

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

### 5.4.3> GPIO

When J7-03~J7-10 is not used as the channel of ADC, it can be configured AS GPIO.

1> Tick the corresponding IO in GPIO direction section, that is configured for input . The IO that is not ticked is configured for output.The set is effective by clicking the “Set” button after choosing. The set is successful until the OK dialog pops up . The status of checkbox in GPIO value section will be refreshed again when the corresponding GPIO direction configurations take effect.

2> The “Read” button and the “write” button In GPIO value section is used to refresh or change the level of GPIO. GPIO that used as the input of ADC sampling can not be configured as in/out . GPIO that configured as the input of

---

GPIO can not be write ,but can be read.

#### 5.4.4> PWM

The PWM has four channels with max frequency of 100K. The frequency of each channel is the same and it opens and closes at the same time.

Duty of each channel can be individually adjusted. No PWM output when the channel number is 0, and when it is 1 , channel#1 outputs , and 2, channel#1,2 output, and so on .

Please select the channel number , the frequency and the duty of ehch chanel, click the “set” button, and the set is successful until the OK dialog pops up.

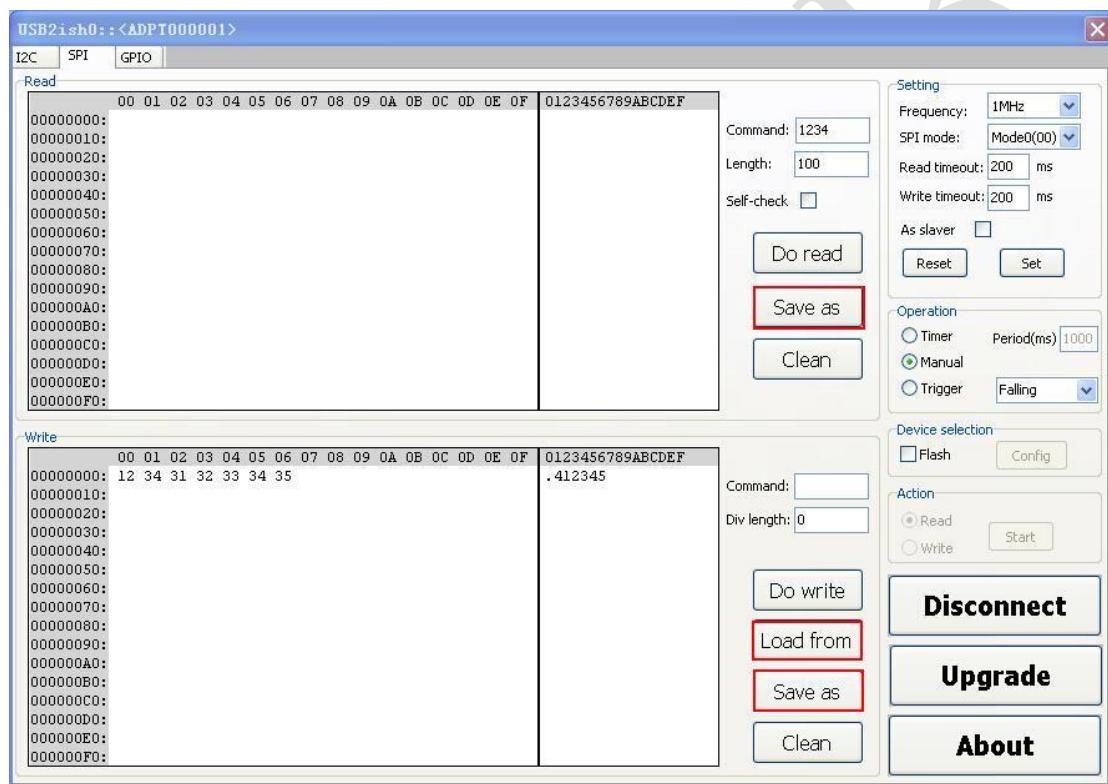
Due to the running error of software , there is a little limit to the maximum duty, which is related to the frequency , the maximum duty of actual measurement is listed below.

frequen cy (KHz)	Minimum Duty	Maximum Duty	Output when duty is 0	Output over the maximum duty
1	0.1%	99.9%	High	Low
2	0.1%	99.8%	High	Low
4	0.1%	99.7%	High	Low
6	0.1%	99.6%	High	Low
8	0.1%	99.5%	High	Low
10	0.1%	99.3%	High	Low
20	0.1%	98.7%	High	Low
40	0.1%	97.5%	High	Low

<b>60</b>	<b>0.1%</b>	<b>96.2%</b>	<b>High</b>	<b>Low</b>
<b>80</b>	<b>0.1%</b>	<b>95.0%</b>	<b>High</b>	<b>Low</b>
<b>100</b>	<b>0.1%</b>	<b>93.7%</b>	<b>High</b>	<b>Low</b>

## 5.5> File operation

The operations of file is accomplished by clicking the “Save as” button or the “Load from” button on the tab of I2c and SPI.

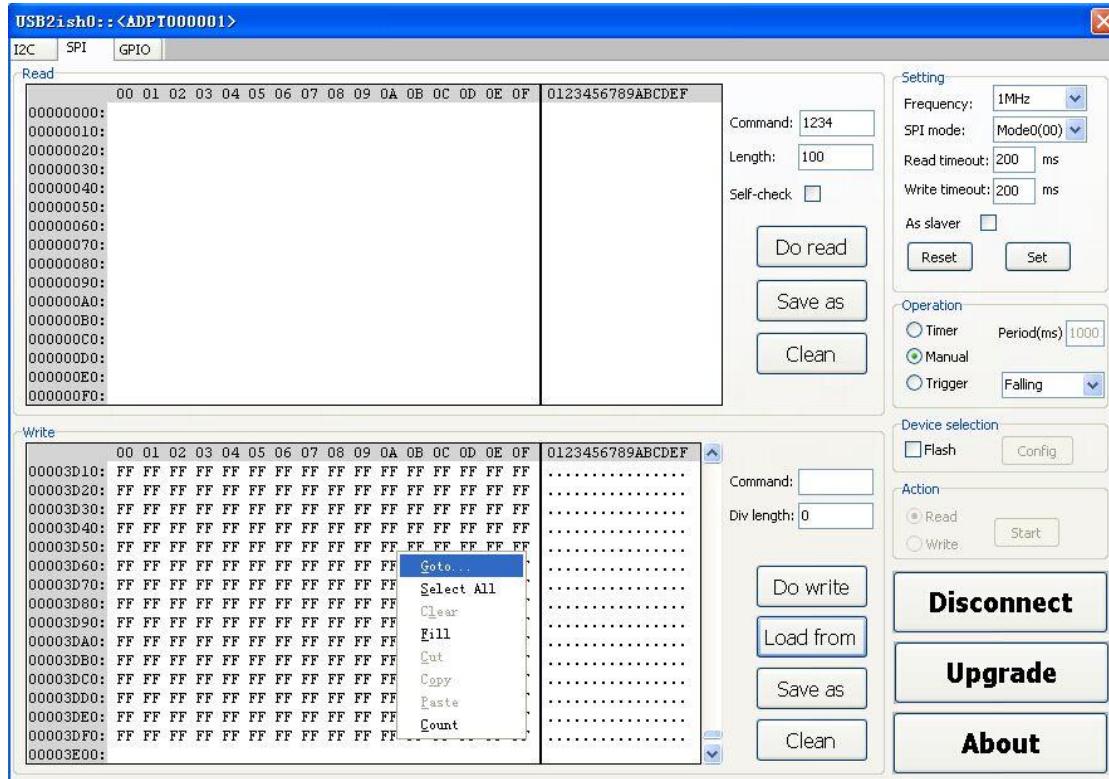


**Load from:** Any format files can be loaded from , only the file of extension called hex can decode, others are loaded the data box with the format of the text file , file size is consistent with the data length. Non ASCII characters are displayed as ‘.’ in text area . Display its hex value in hex area .

**Save as:** Save the data in the data box on the computer in the format of the text, file size is consistent with the data length of data box.

## 5.5> Pop-up menu

If you move the mouse to the area of the data box, click the right mouse button , a menu will pops up , as shown below.



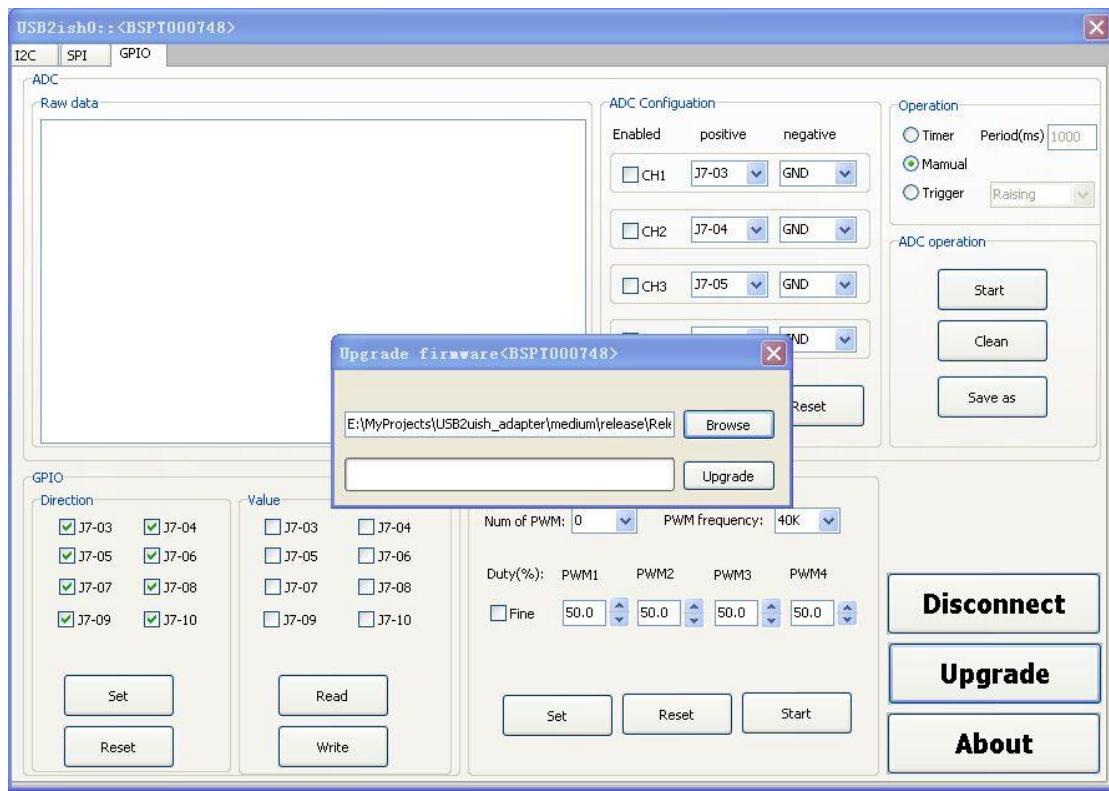
### Menu description

- **Goto:** Position the edit cursor to the offset address.
- **Select All:** Select all the contents of the cache.
- **Clean:** Delete the selected content.
- **Fill:** Change the content of a block to the same byte.
- **Cut:** Delete the selected content and copy to the paste board.
- **Copy:** Copy the selected content to the paste board.
- **Paste:** Insert the content of the paste board into the edit cursor location
- **Count:** Count the cache length of content box ,unit is byte.

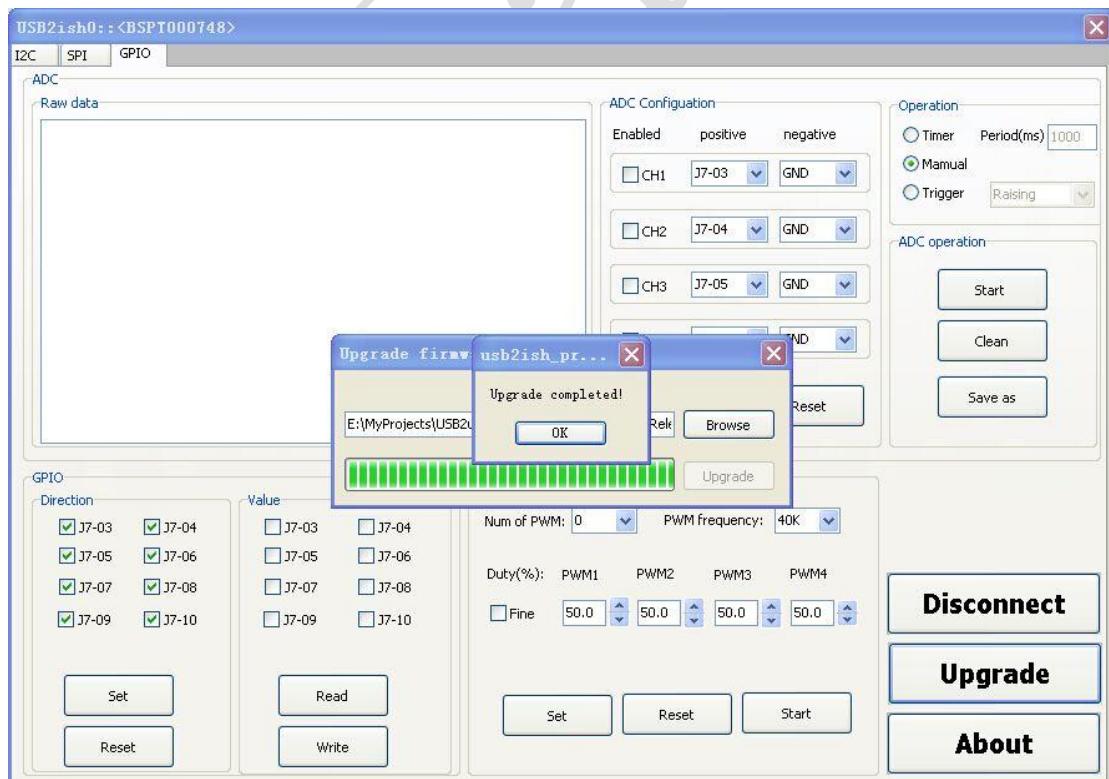
### 5.6> Upgrade Firmware

Make sure that the upgrade jumper is 1-2 short. Insert the USB2UIS board.

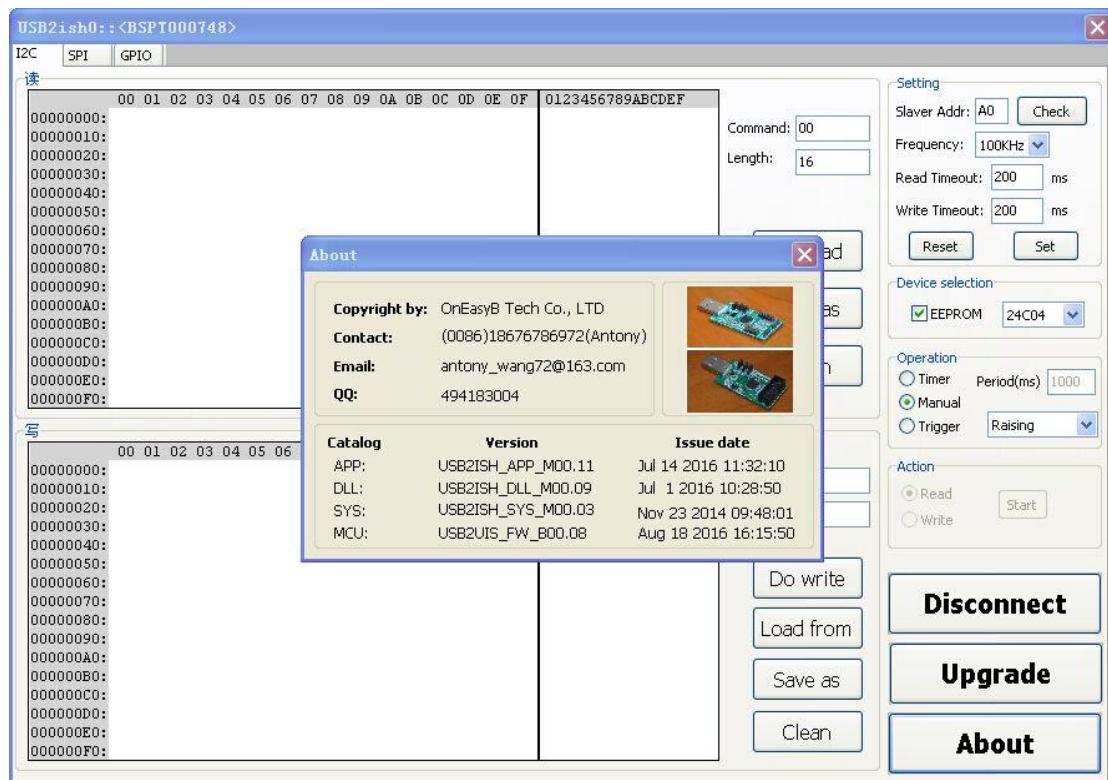
After successful connection to the device, then click the “Upgrade” button, as shown below.



Click the “Browse” button to select file, and then press the “Upgrade” button to start. After success , as shown below.



Different software has different version and release time . After the successful upgrade, change the upgrade jumper back to 2-3 short. Insert the board, click the “About” button after the successful connection to the device , you can view the firmware version and release time and detect whether the upgrade is successful or not . As shown below.



If the version has something turn , subject to the actual use .

---

## **6. Usage in the linux system**

### **6.1> Use Uart**

Make sure that the function jumper is 1-2 short, and the upgrade jumper is 2-3 short.

Plug the board, and enter ls /dev/ttyACM\* on the console . A new device ttyACM0~9 appears , which proves that you can implement the Com communication in Linux. After Installing a com software such as minicom, you can use usb2uart function.

### **6.2> Use I2C, SPI , GPIO,PWM**

#### **6.2.1 > Ready**

- Make sure that both of the function and upgrade jumper are 2-3 short.
- Make sure that the software package of libusb has be installed.
- Make sure that the software pageage of Qt-creator (3.0.1)has be installed.

#### **6.2.2> Steps**

- 1> Open the console and Enter the fold of linux/lib ,then enter sudo sh install.sh to install libusb2ish.so.
- 2> Open qt-creator and open the demo project “I2C\_RW.pro” or “SPI\_RW.pro”.
- 3> Compile and build the project , then run .

## 7. Issues list and solutions

	issue	Possible causes	Solutions
1	Plug USB, device manager isn't responding.	PC USB port Damaged	move to other USB ports
		Driver out of work	Restart the computer
		US2UIS damaged	Replace USB2UIS board
2	Can not read the software version information , as shown below, a lot of question mark.	App out of work	Close the app,Unplug and plug the USB2UIS board, re-open the app.
		Driver out of work	Restart the computer
3	To run Usb2ish_pro_en.exe, pop-up the following prompt window.	Application has no execute permissions.	Right click the app and select run as administrator
4	Failure of Driver installation	Jumper error or select a wrong driver	Re-check Jumper ,and make sure selected Driver folder is

		files	right .
	360 anti-virus software	Close 360, and re-install driver	
6	Failure problem of installin UART driver , as shown below.	<p>1&gt; Window system under the inf folder, there is no mdmcpq.inf file, or system32\dirver\ has no usbser.sys file.</p> <p>2&gt; Syntax error in paragraph of mdmcpq.inf file</p>	<p>1&gt; Download the corresponding system version of mdmcpq.inf and usbser.sys to the specified folder online.</p> <p>2&gt; change  c:\Windows\inf\mdmcpq.inf  [FakeModemCopyFileSection]  usbser.sys,,0x20  改为  [FakeModemCopyFileSection]  ;usbser.sys,,0x20</p>

---

## **8. Maintenance**

- 8.1> The products, which was used properly, will provide 12 months free maintenance service .
- 8.2> If the warranty period is expired, the repair can be provided free but user need to pay all the other fee (such as express fee , materials fee and so on).

OnEasyB