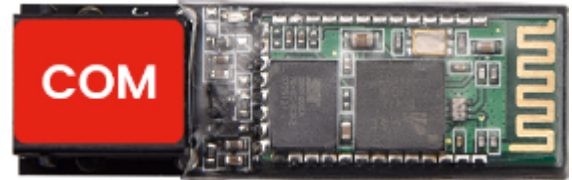


Bluetooth 2.0 Module (000x0000 Article Number) (TS2183)



Product Details

This is a TelePort Bluetooth 2.0 module, which can easily achieve serial wireless data transmission. Its operating frequency is 2.4GHz ISM band. It adopts Bluetooth 2.1+EDR standard. For Bluetooth 2.1, the signal transmission time interval of different devices is 0.5 seconds, which greatly reduces the workload of the Bluetooth chip and saves more sleep time. This sensor uses the RJ11 interface, which can be easily connected to the expansion board and main board of the RJ11 interface series.



Features and Benefits

- Compatible with RJ11 6P6C OKdo TelePort Control boards and expansion shields.
- It can easily achieve serial wireless data transmission.
- The operating frequency is among the most popular 2.4GHz ISM frequency.
- It has built-in onboard antenna which provides high-quality signals.
- Please note: iOS devices (iPhone) are not supported. ✦

Technical Specifications

Working voltage	5V
Bluetooth protocol	Bluetooth 2.1+ EDR standard
Operating frequency	2.4GHz ISM frequency band
USB protocol	USB v1.1/2.0
Modulation mode	GFSK (Gauss Frequency Shift Keying)
Transmit power	≤ 4dBm, class 2
Sensitivity	≤ -84dBm at 0.1% Bit Error Rate
Transfer rate	2.1Mbps(Max)/160kbps ; Synchronous: 1Mbps/1Mbps
Supported configuration	Bluetooth serial port
Operating temperature	-20°C to +55°C
Dimensions	52mm*16mm*18mm
Weight	5.3g

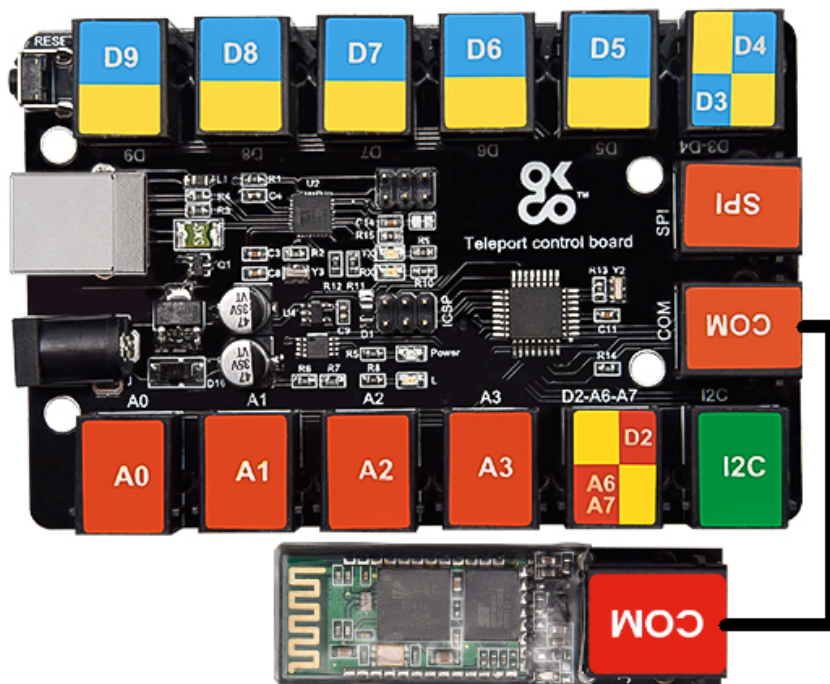
Applications

- Bluetooth remote cars
- Medical equipment

- Bluetooth earphone, mobile phone, PC, laptop, household application, digital cameras and other electronic products

This module is compatible with the TS2180-Raspberry Pi shield, the TS2179-Micro:bit shield and the TS2178-TelePort main board.

Arduino Application



This module is compatible with the TS2178 TelePort control board.

Test Code

Note: first upload the test code to the control board then connect the Bluetooth module; otherwise, you will fail to upload test code.

```
int val;

int ledpin=13;

void setup()
{
  Serial.begin(9600);
  pinMode(ledpin,OUTPUT);
} void loop()
{ val=Serial.read();
```

```
if(val=='a')
{
digitalWrite(ledpin,HIGH);

delay(250);

digitalWrite(ledpin,LOW);

delay(250);

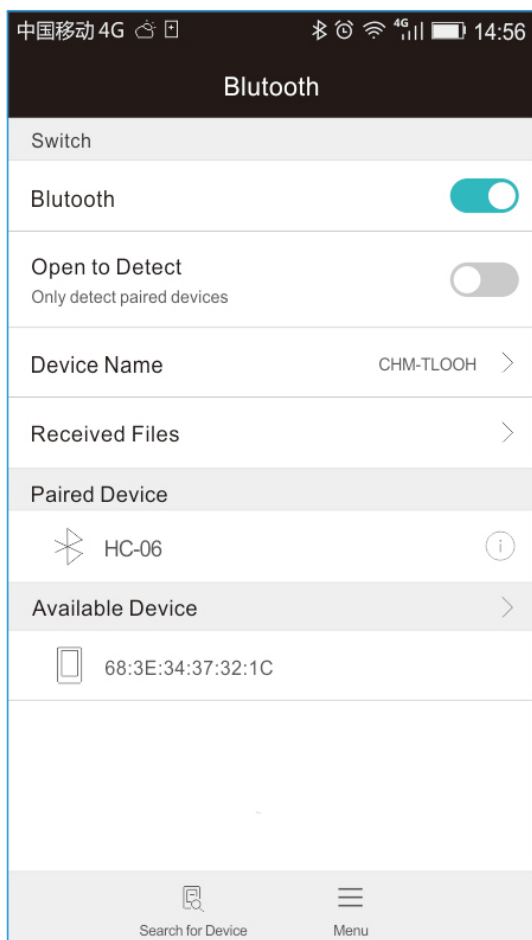
Serial.println("okdo");

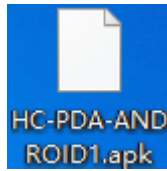
}
}
```

Test Result

Wire up, upload the code to the development board, power it up and insert the Bluetooth module, then its LED will flash.

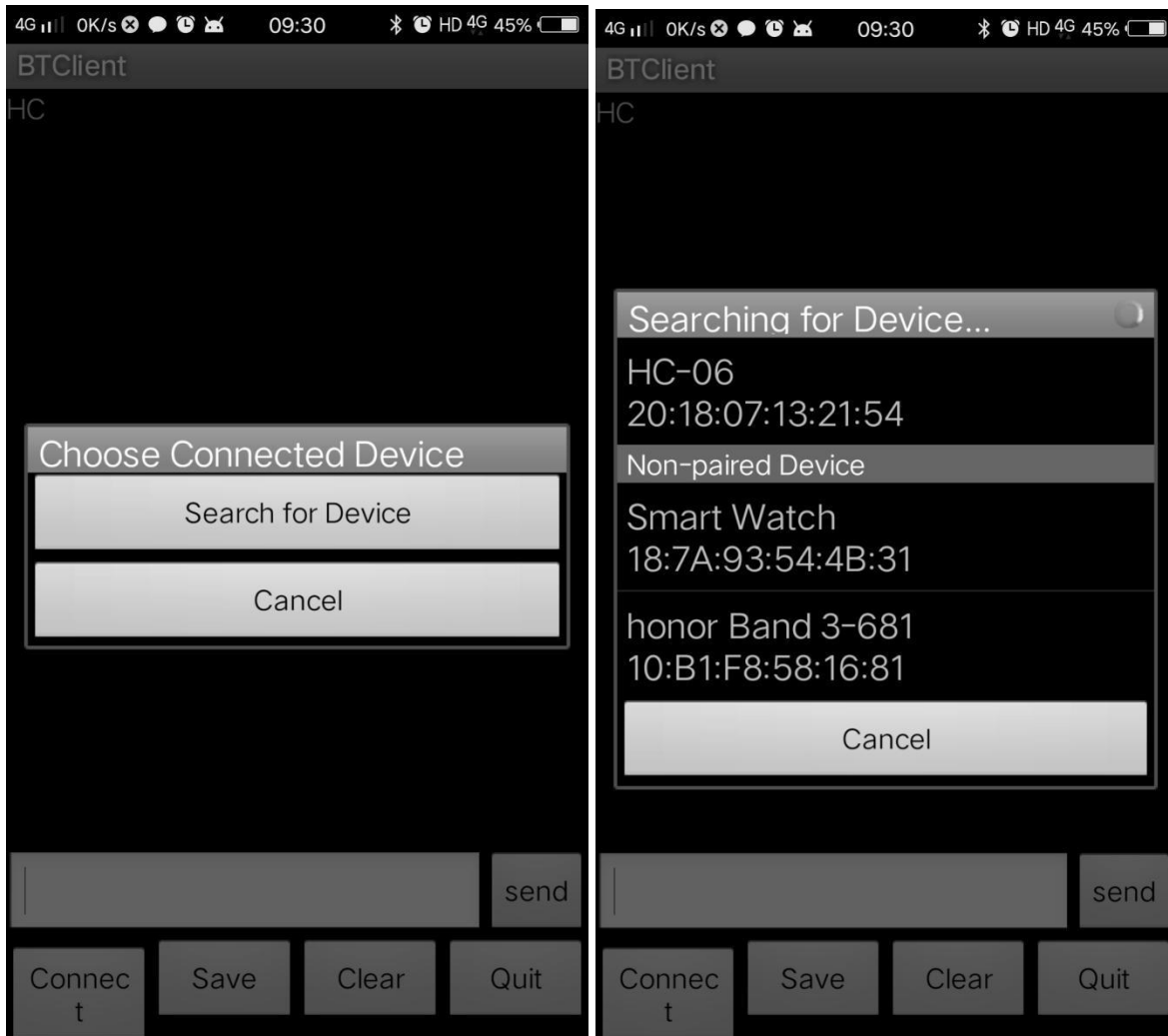
Turn on Bluetooth on your phone, pair the Bluetooth device, enter 1234, and you will see the paired device.



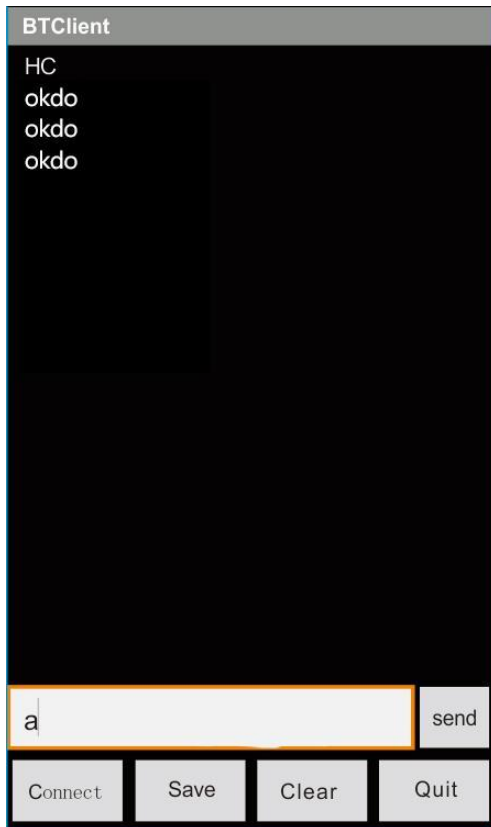


2. Transfer the file **HC-PDA-ANDROID1.apk** we provide to the ipad or Android cellphone, then download it and open the Bluetooth assistant- BTClient
3. Click"Connect"---->"Search for Device"and select your Bluetooth device(HC-06).

www.OKdo.com | #LetsOKdo | Design The World



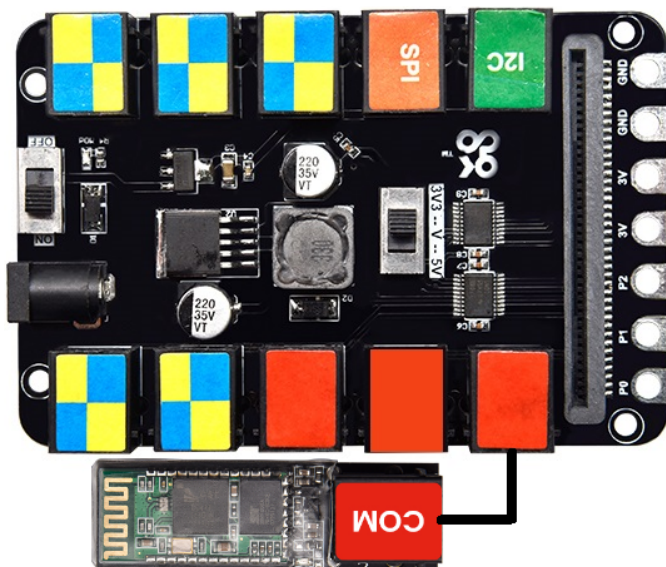
4. After connecting BTClient, when you send **a**, BTClient will show **okdo** and LED on the control board will flash.



www.OKdo.com | [#LetsOKdo](https://twitter.com/LetsOKdo) | Design The World

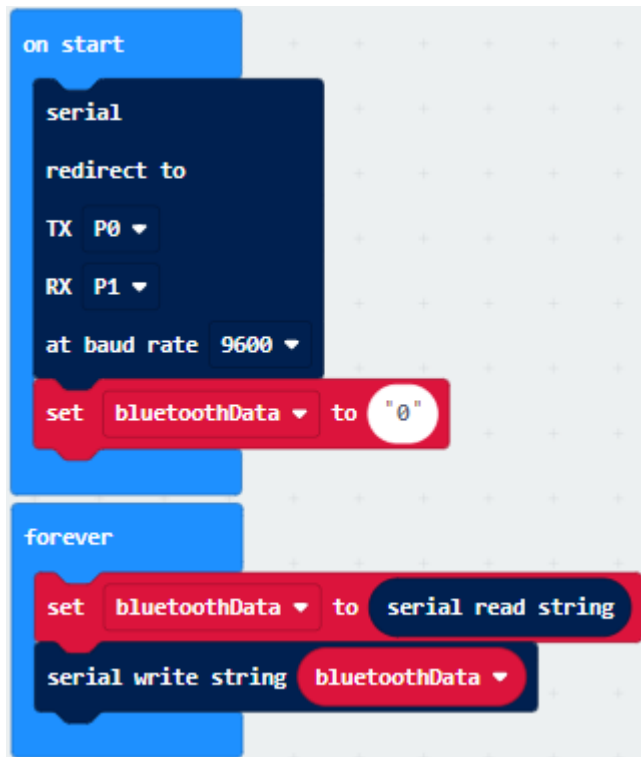
If you want to know more details about Arduino and the TelePort control board, you can refer to TS2178.

Micro:bit Application



It is compatible with the Micro:bit board and the TS2179 Micro:bit expansion board.

Test Code



.....①Run the "on start" block to boot the program

.....②TX is connected to P0, RX is connected to P1 and set baud rate to 9600

.....③set the initial value of the variable bluetoothData to 0

.....④The program is run circularly under the command of "forever" block

.....⑤set the variable bluetoothData to character string at serial port

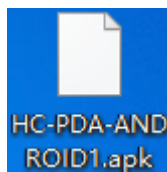
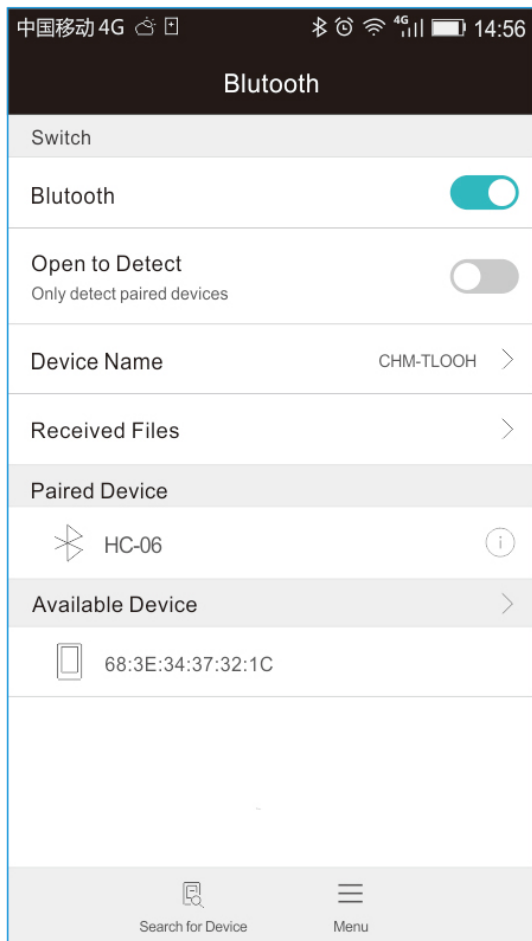
.....⑥serial port prints the variable bluetoothData

Test Result

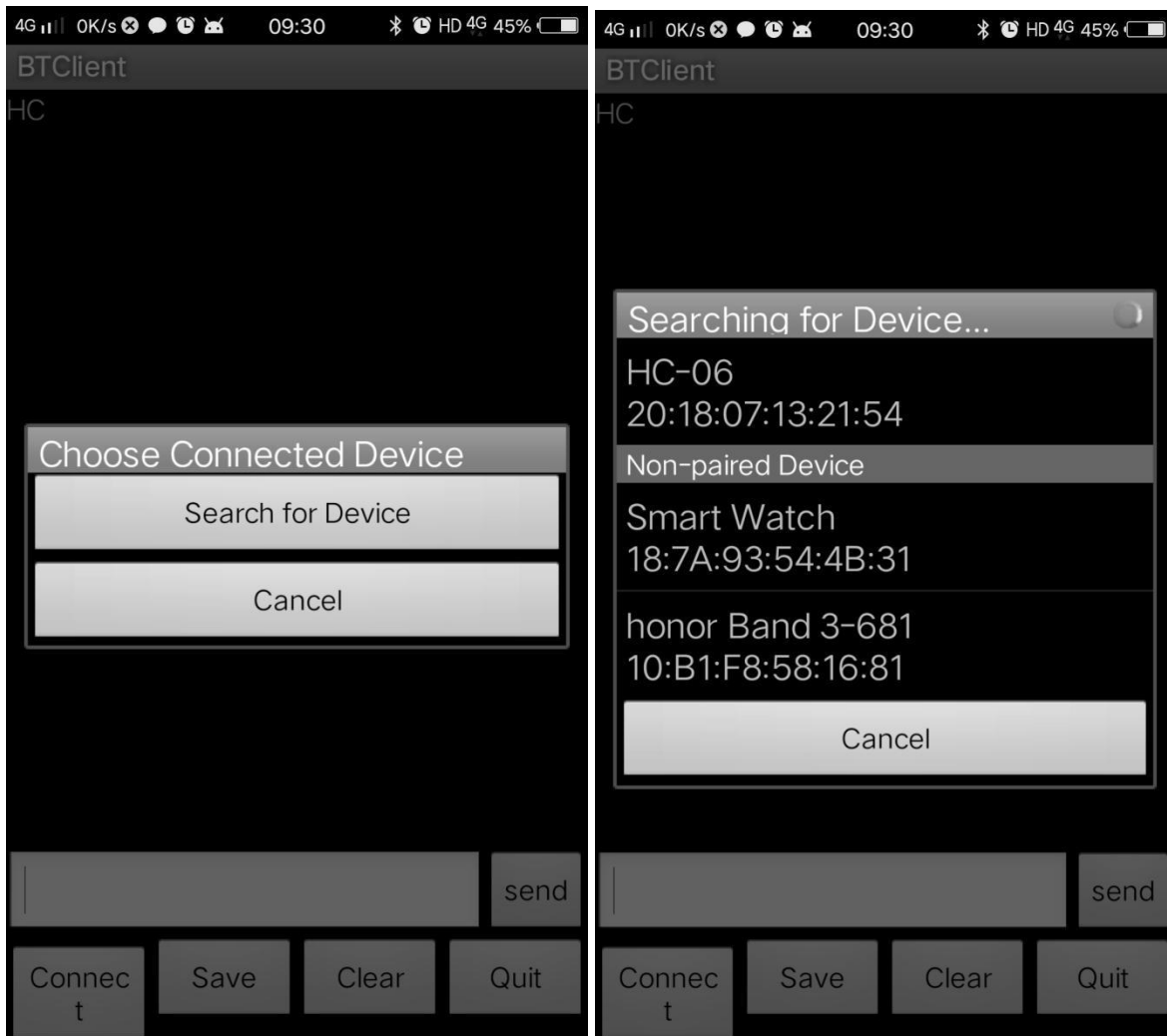
Wire up, insert the Micro:bit V2.0 board, turn the DIP switch to 3V3 end.

Upload test code, power it up and insert the Bluetooth module; then LED on the module will flash

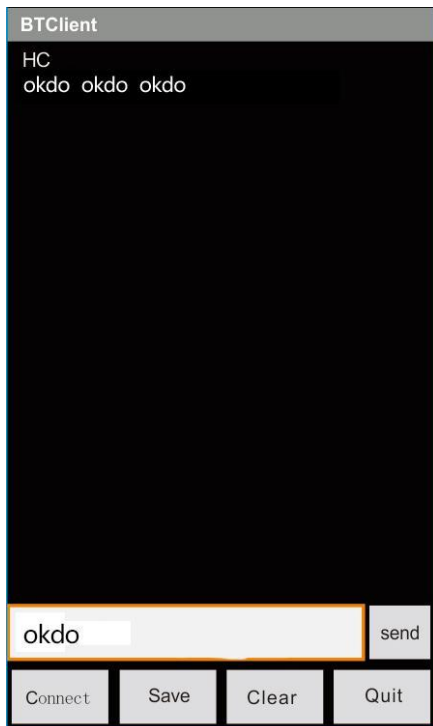
Turn on Bluetooth on your phone, pair the Bluetooth device, enter 1234, and you will see the paired device.



5. Transfer the file **HC-PDA-ANDROID1.apk** we provide to the ipad or Android cellphone, then download it and open the Bluetooth assistant- BTClient
6. Click“Connect”---->“Search for Device”and select your Bluetooth device(HC-06).

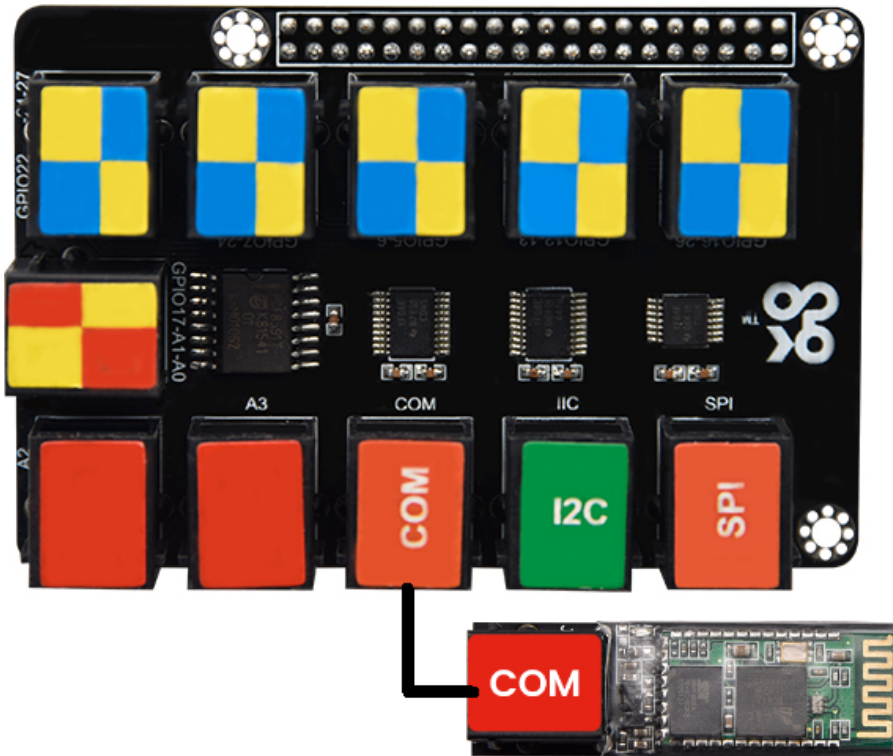


After connecting BTClient, when you send **okdo**, BTClient will show okdo .



If you want to know more details about Micro:bit and the Micro:bit shield, you can refer to TS2179.

Raspberry Pi Application



This module is compatible with the Raspberry Pi board and the TS2180 Raspberry Pi shield.

Serial Configuration

Run the command `ls /dev -al` in the terminal to check default port distribution method, as shown below;

```
pi@raspberrypi: ~  
File Edit Tabs Help  
drwxr-xr-x 2 root root 60 Jan 1 1970 raw  
crw-rw-r-- 1 root netdev 10, 242 Oct 12 01:17 rfkill  
crw-rw---- 1 root video 238, 0 Oct 12 01:17 rpivid-h264mem  
crw-rw---- 1 root video 240, 0 Oct 12 01:17 rpivid-hevcmem  
crw-rw---- 1 root video 239, 0 Oct 12 01:17 rpivid-intcmem  
crw-rw---- 1 root video 237, 0 Oct 12 01:17 rpivid-vp9mem  
lrwxrwxrwx 1 root root 7 Oct 12 01:17 serial0 -> ttyAMA0  
lrwxrwxrwx 1 root root 5 Oct 12 01:17 serial1 -> ttyS0  
drwxrwxrwt 2 root root 40 Feb 14 2019 shm  
drwxr-xr-x 3 root root 140 Oct 12 01:17 snd  
lrwxrwxrwx 1 root root 15 Feb 14 2019 stderr -> /proc/self/fd/2  
lrwxrwxrwx 1 root root 15 Feb 14 2019 stdin -> /proc/self/fd/0  
lrwxrwxrwx 1 root root 15 Feb 14 2019 stdout -> /proc/self/fd/1  
crw-rw-rw- 1 root tty 5, 0 Oct 12 01:17 tty  
crw--w---- 1 root tty 4, 0 Oct 12 01:17 tty0  
crw----- 1 pi tty 4, 1 Oct 12 01:32 tty1  
crw--w---- 1 root tty 4, 10 Oct 12 01:17 tty10  
crw--w---- 1 root tty 4, 11 Oct 12 01:17 tty11  
crw--w---- 1 root tty 4, 12 Oct 12 01:17 tty12
```

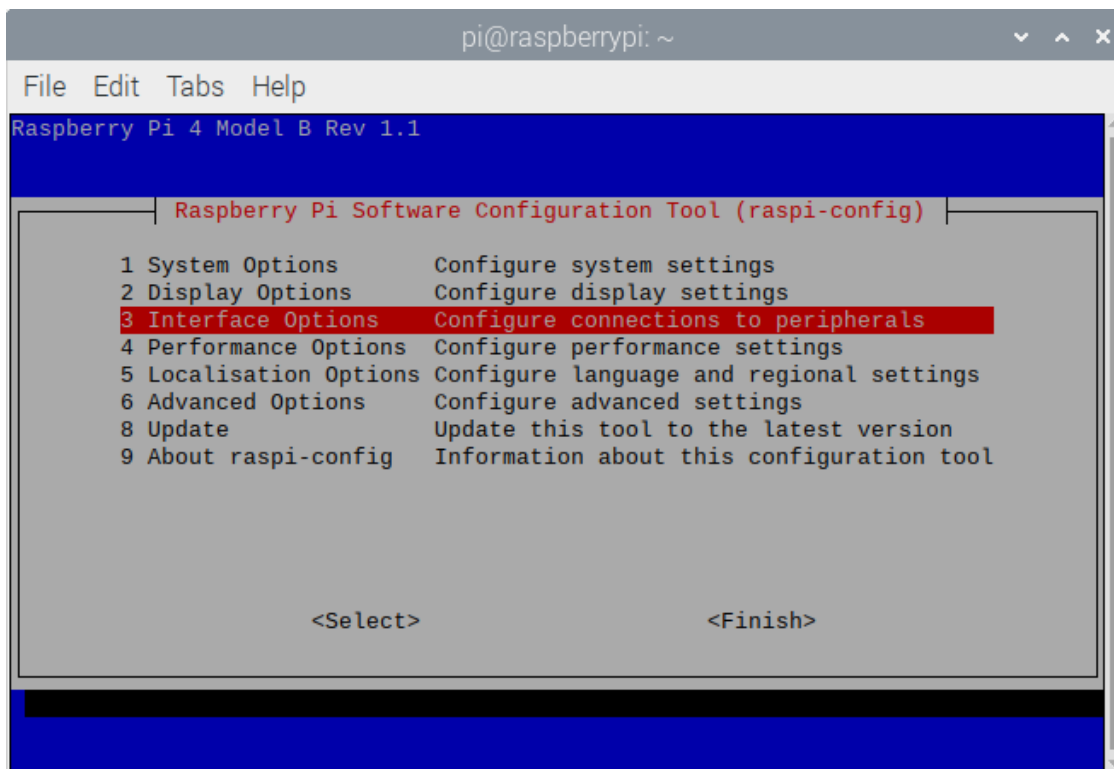
`serial0 -`
`serial1 -`

Serial ports `serial0` and `serial1` are used for the Bluetooth module.
Set the hardware port to GPIO port.
First log in the command terminal

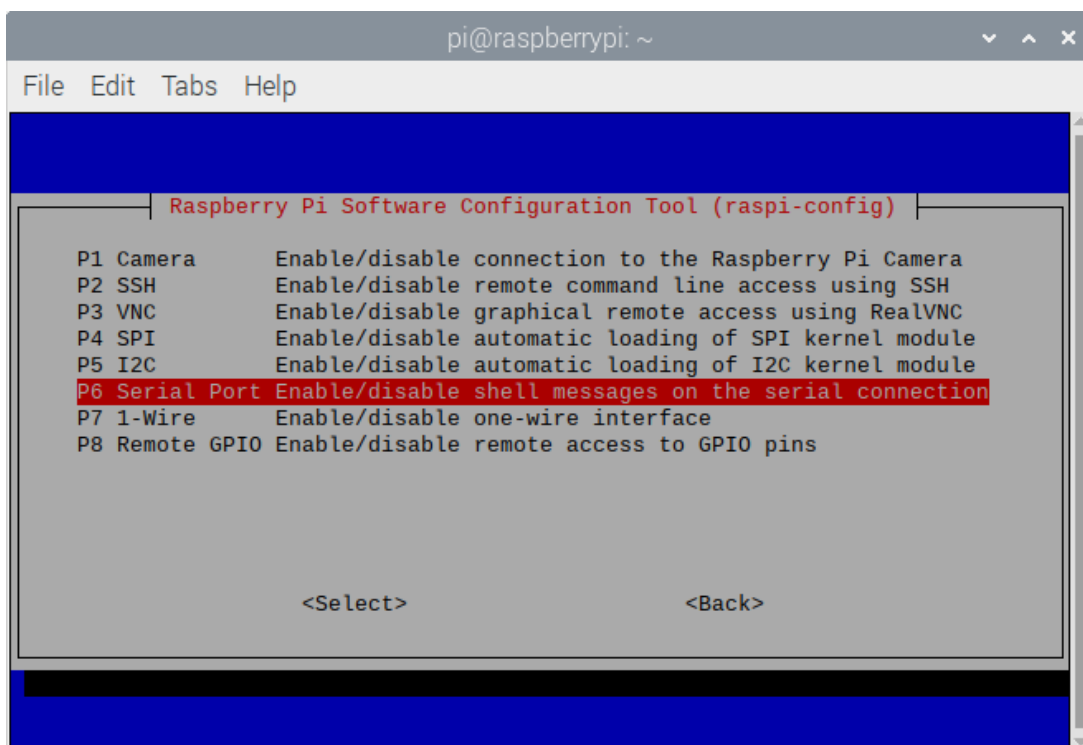
Input `sudo raspi-config` in the terminal to enter the Raspberry Pi configuration interface.

```
pi@raspberrypi:~/python $ sudo raspi-config
```

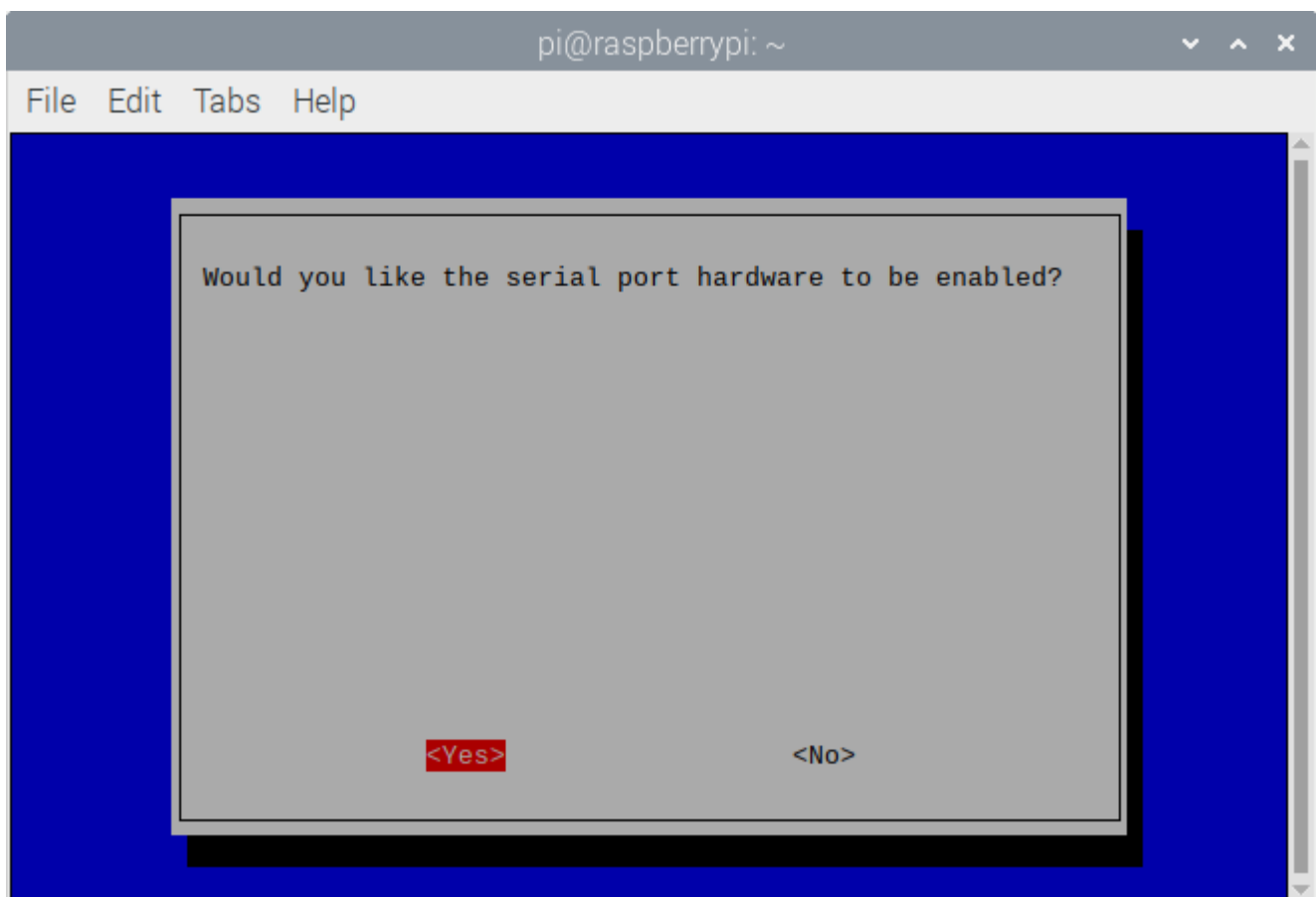
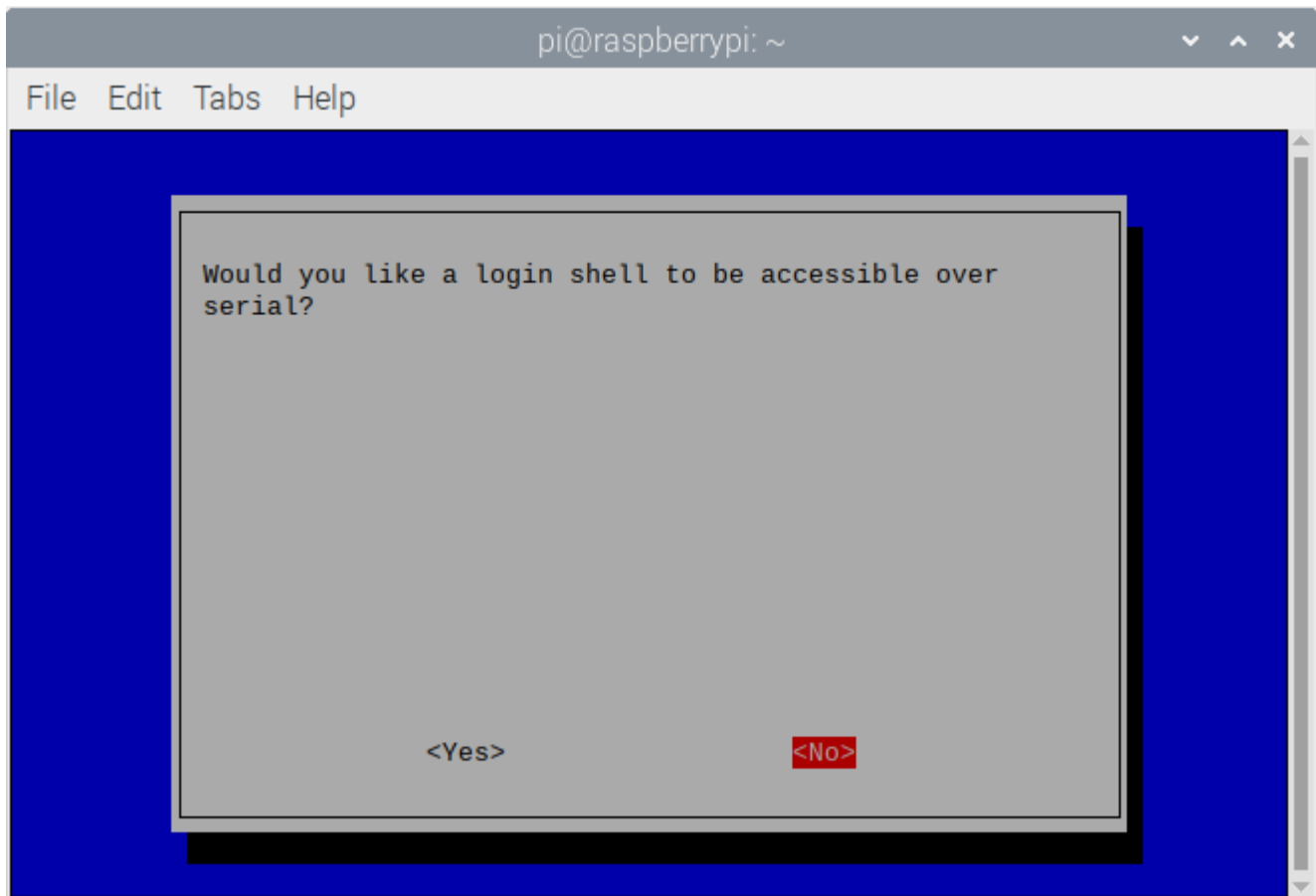
Select the third interface options: (press ←,↑,↓,→ then“Enter”)



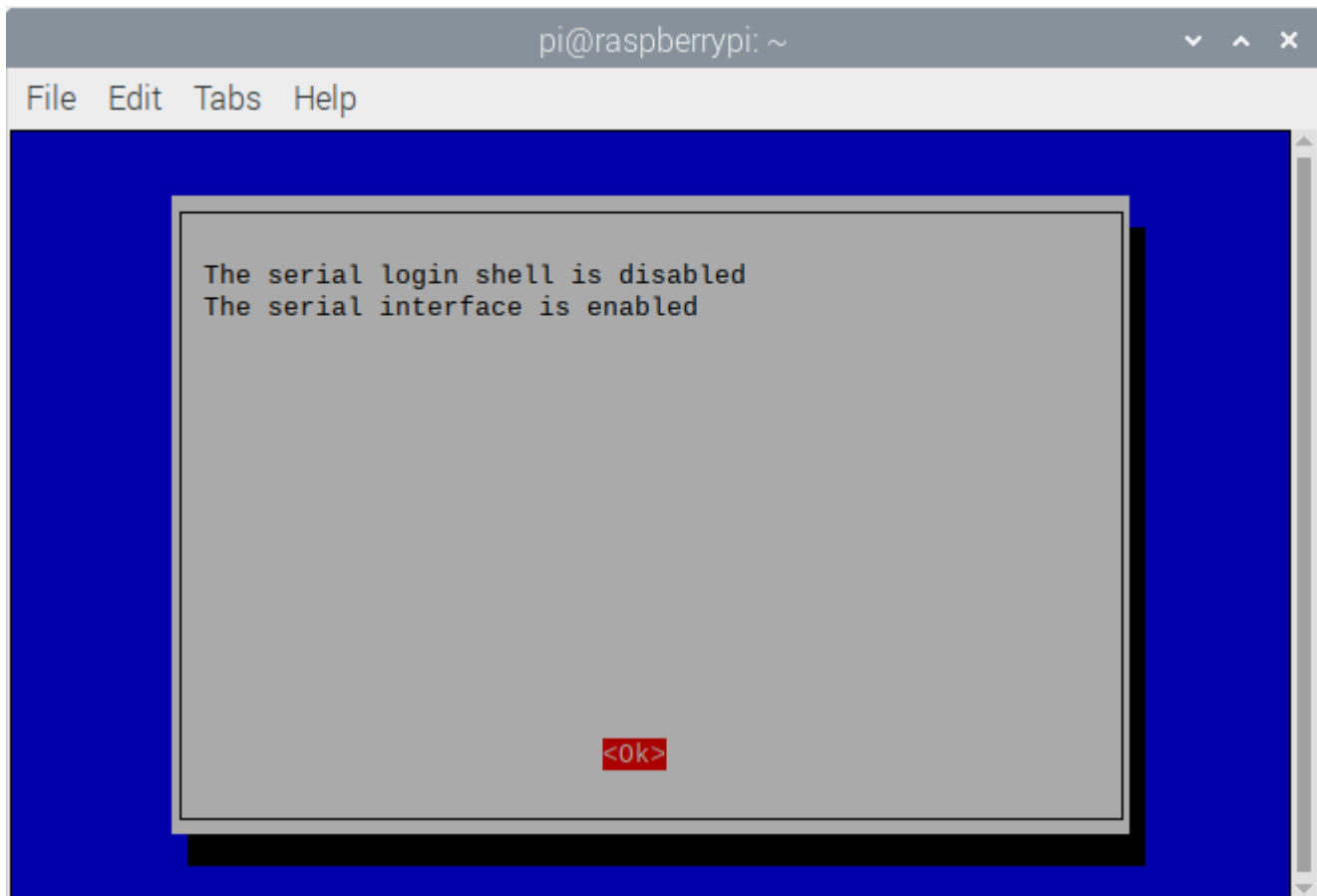
Enter P6 Serial Port



Select to turn off serial login function, and open hardware serial debug function:



Then click [OK](#)



Set hardware serial port to GPIO port

Configure the serial port as our GPIO serial port, input `sudo vim /boot/config.txt` command in the terminal and add `dtoverlay=pi3-miniuart-bt force_turbo=1`, then save and exit, as shown below;

A screenshot of a terminal window on a Raspberry Pi. The window title is 'pi@raspberrypi: ~'. The menu bar shows 'File', 'Edit', 'Tabs', and 'Help'. The background is black. The terminal displays the contents of the /boot/config.txt file. The text is as follows:

```
# Uncomment this to enable infrared communication.
#dtoverlay=gpio-ir,gpio_pin=17
#dtoverlay=gpio-ir-tx,gpio_pin=18

# Additional overlays and parameters are documented /boot/overlays/README

# Enable audio (loads snd_bcm2835)
dtparam=audio=on

[pi4]
# Enable DRM VC4 V3D driver on top of the dispmanx display stack
dtoverlay=vc4-fkms-v3d
max_framebuffers=2

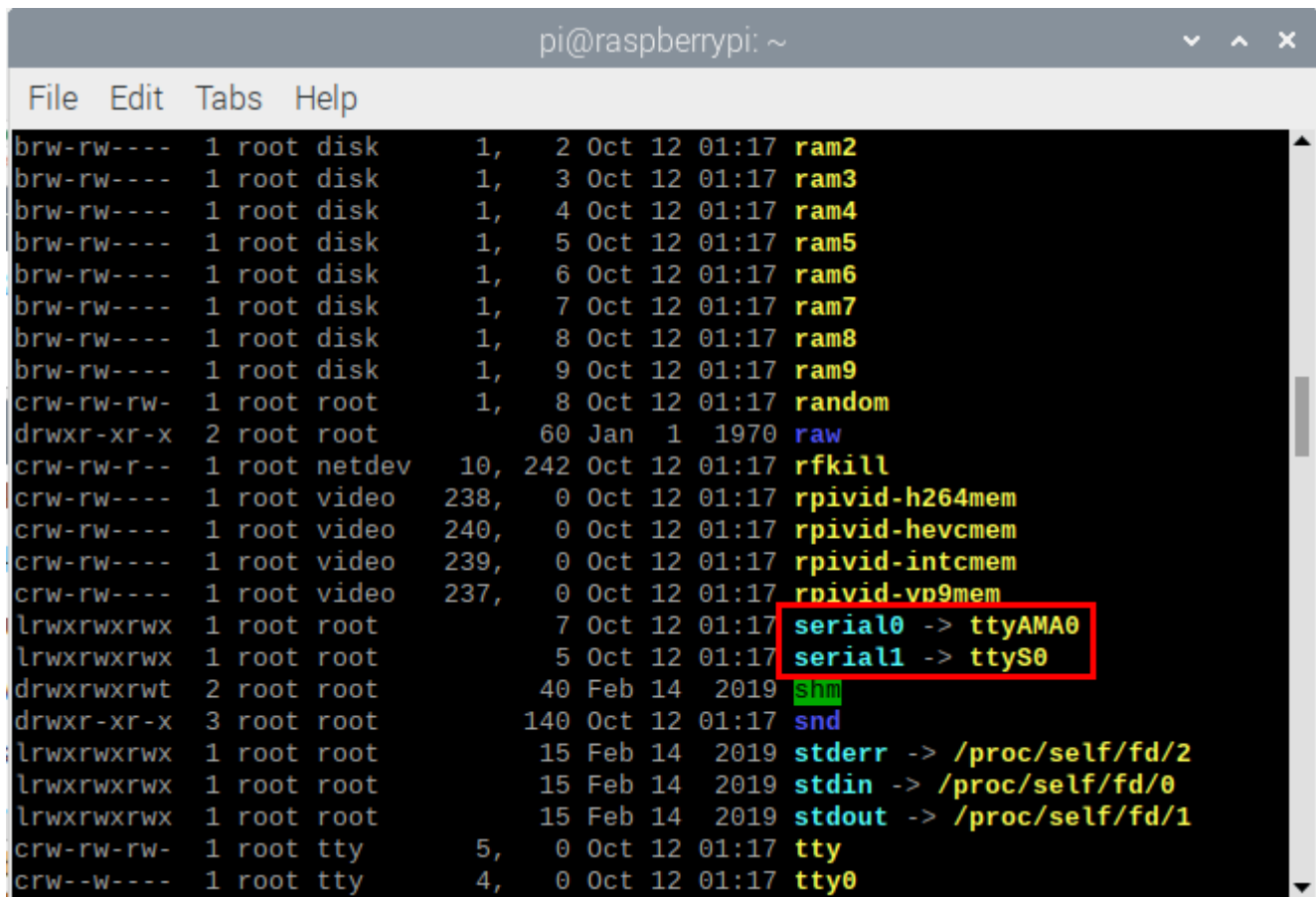
[all]
#dtoverlay=vc4-fkms-v3d
start_x=1
gpu_mem=128
enable_uart=1

dtoverlay=pi3-miniuart-bt
force_turbo=1

~
```

At the bottom right of the terminal window, the text '71,13' and 'Bot' are visible.

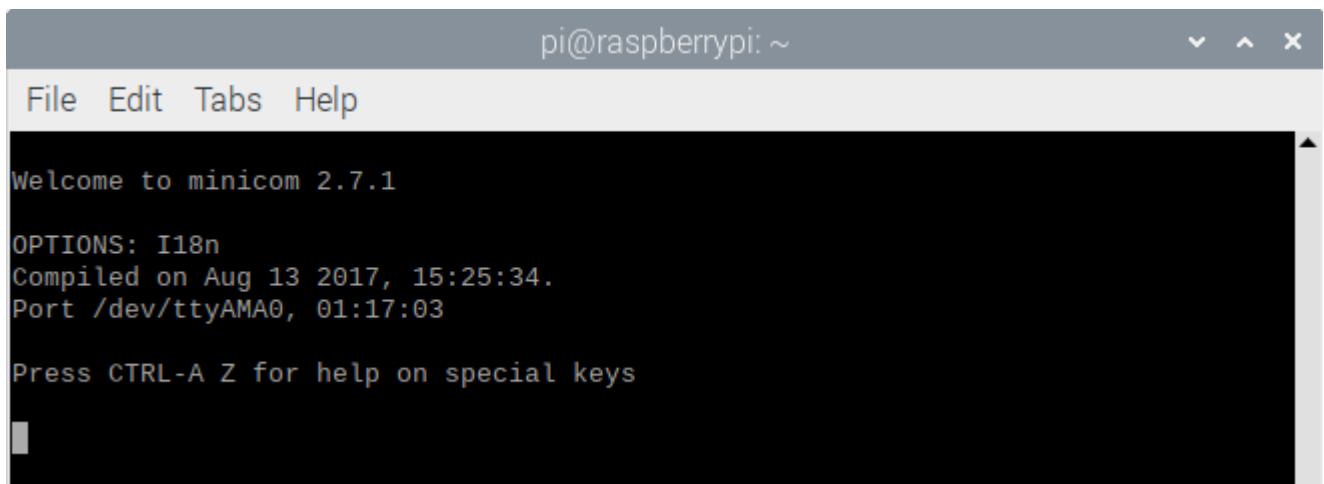
Input the rebooting command `sudo reboot` in the terminal. After rebooting Raspberry Pi, input `ls /dev -al` again. Then you will see two serial ports



```
pi@raspberrypi: ~  
File Edit Tabs Help  
brw-rw---- 1 root disk 1, 2 Oct 12 01:17 ram2  
brw-rw---- 1 root disk 1, 3 Oct 12 01:17 ram3  
brw-rw---- 1 root disk 1, 4 Oct 12 01:17 ram4  
brw-rw---- 1 root disk 1, 5 Oct 12 01:17 ram5  
brw-rw---- 1 root disk 1, 6 Oct 12 01:17 ram6  
brw-rw---- 1 root disk 1, 7 Oct 12 01:17 ram7  
brw-rw---- 1 root disk 1, 8 Oct 12 01:17 ram8  
brw-rw---- 1 root disk 1, 9 Oct 12 01:17 ram9  
crw-rw-rw- 1 root root 1, 8 Oct 12 01:17 random  
drwxr-xr-x 2 root root 60 Jan 1 1970 raw  
crw-rw-r-- 1 root netdev 10, 242 Oct 12 01:17 rfkill  
crw-rw---- 1 root video 238, 0 Oct 12 01:17 rpivid-h264mem  
crw-rw---- 1 root video 240, 0 Oct 12 01:17 rpivid-hevcmem  
crw-rw---- 1 root video 239, 0 Oct 12 01:17 rpivid-intcmem  
crw-rw---- 1 root video 237, 0 Oct 12 01:17 rpivid-vp9mem  
lrwxrwxrwx 1 root root 7 Oct 12 01:17 serial0 -> ttyAMA0  
lrwxrwxrwx 1 root root 5 Oct 12 01:17 serial1 -> ttyS0  
drwxrwxrwt 2 root root 40 Feb 14 2019 shm  
drwxr-xr-x 3 root root 140 Oct 12 01:17 snd  
lrwxrwxrwx 1 root root 15 Feb 14 2019 stderr -> /proc/self/fd/2  
lrwxrwxrwx 1 root root 15 Feb 14 2019 stdin -> /proc/self/fd/0  
lrwxrwxrwx 1 root root 15 Feb 14 2019 stdout -> /proc/self/fd/1  
crw-rw-rw- 1 root tty 5, 0 Oct 12 01:17 tty  
crw--w---- 1 root tty 4, 0 Oct 12 01:17 tty0
```

Minicom Serial Assistant Test

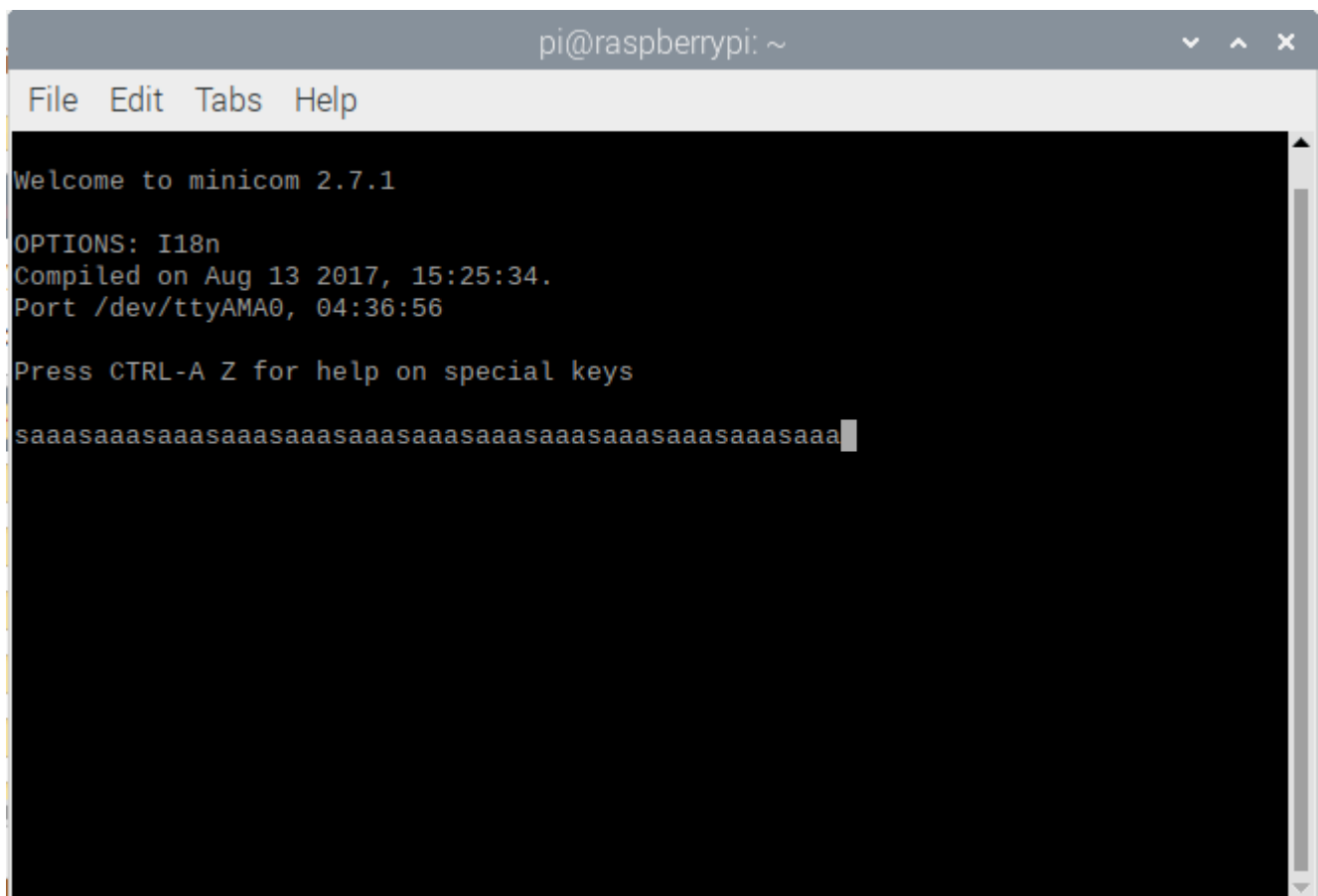
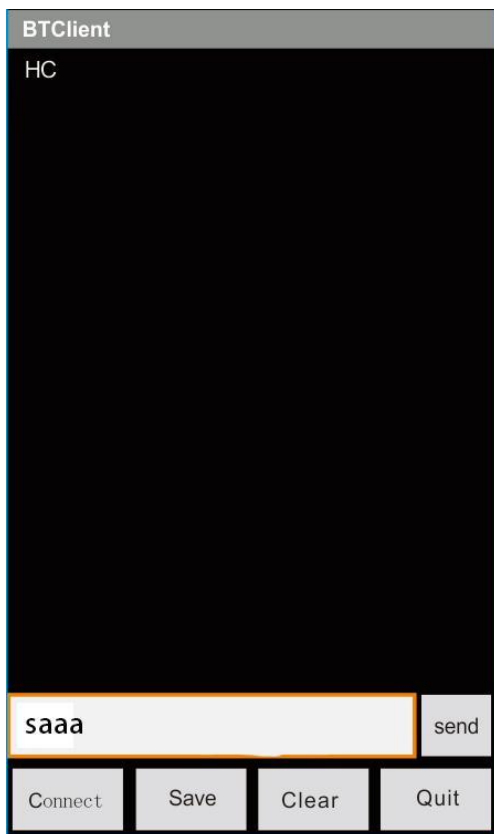
First input `sudo apt-get install minicom` in the terminal to install minicom. After installing, then input `minicom -D /dev/ttyAMA0 -b 9600` to boot minicom, as shown below;



```
pi@raspberrypi: ~  
File Edit Tabs Help  
Welcome to minicom 2.7.1  
OPTIONS: I18n  
Compiled on Aug 13 2017, 15:25:34.  
Port /dev/ttyAMA0, 01:17:03  
Press CTRL-A Z for help on special keys  
█
```

-D indicates serial port `/dev/ttyAMA0`, -b means setting baud rate to 9600. You don't need to set this parameter, since it can be defaulted as 11520. Then press Ctrl+A and Z to exit the minicom.

Connect Raspberry Pi to TTL, then you can transfer data to Raspberry Pi through PC serial assistant. Send `saaa` in the PC serial assistant, then the terminal will receive `saaa`. But when the data is sending, the terminal won't show anything, as shown below;



If you want to know how to utilize Raspberry Pi and the Raspberry Pi shield, you can refer to TS2180.

END