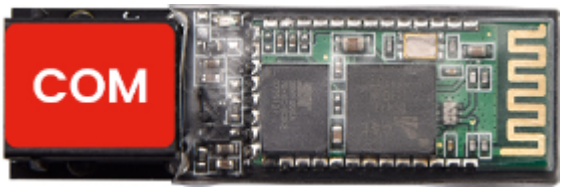
Bluetooth 2.0 Module (000x0000 Article Number) (TS2183)

DESIGN THE WORLD

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℃ to +55℃
Dimensions	52mm*16mm*18mm
Weight	5.3g

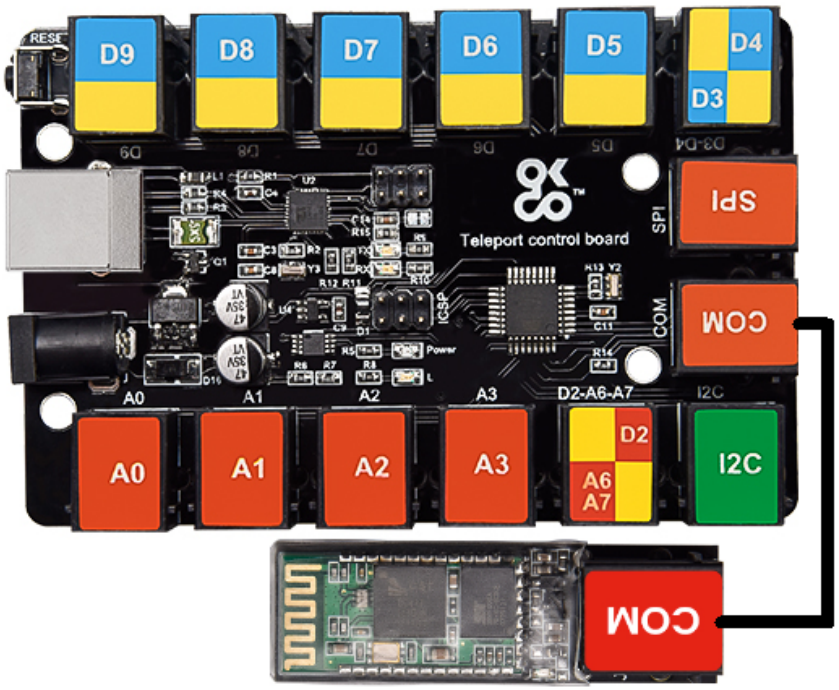
Applications

- 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.

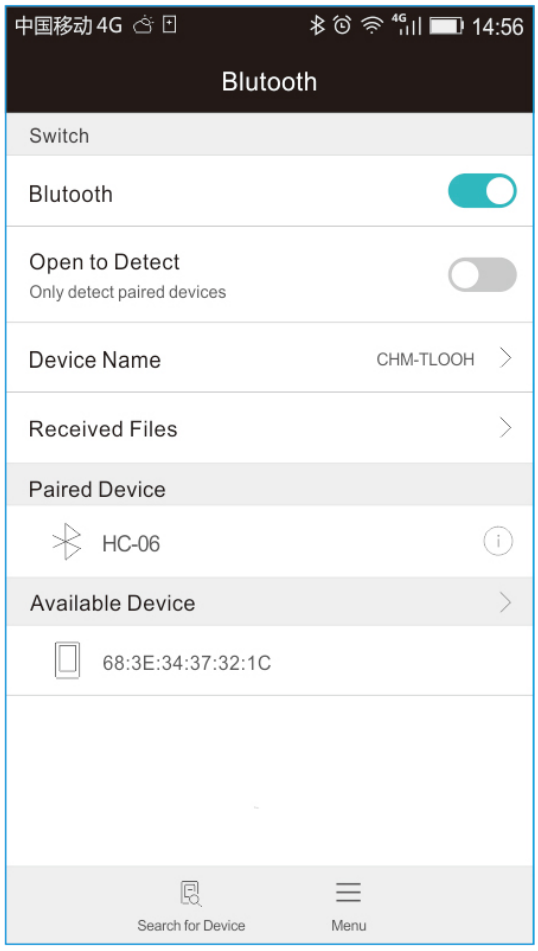
```
int val;

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.

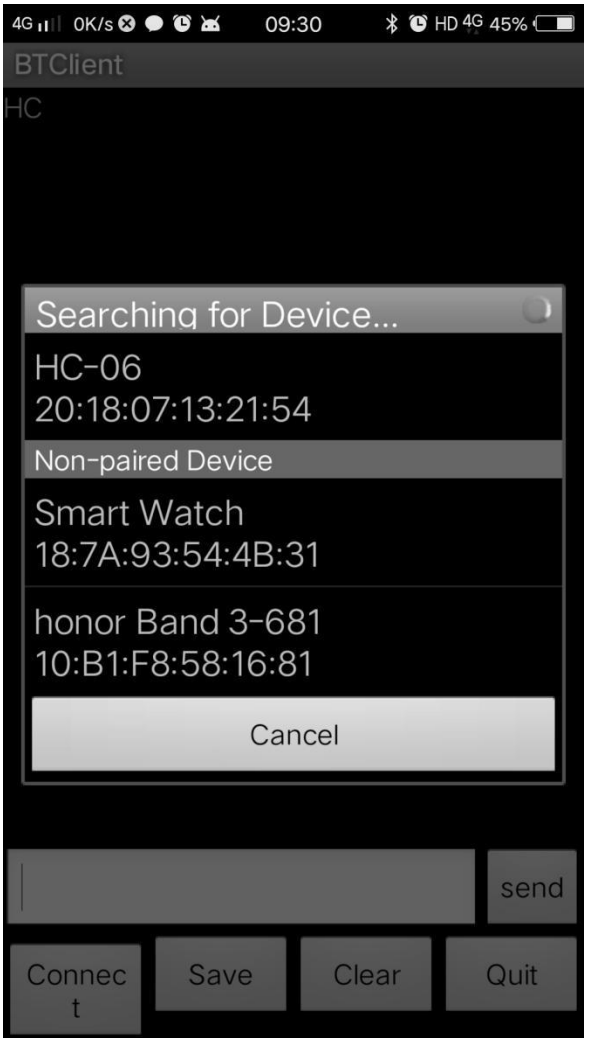
Turn on Bluetooth on your phone, pair the Bluetooth device, enter 1234, and you will see the paired device.



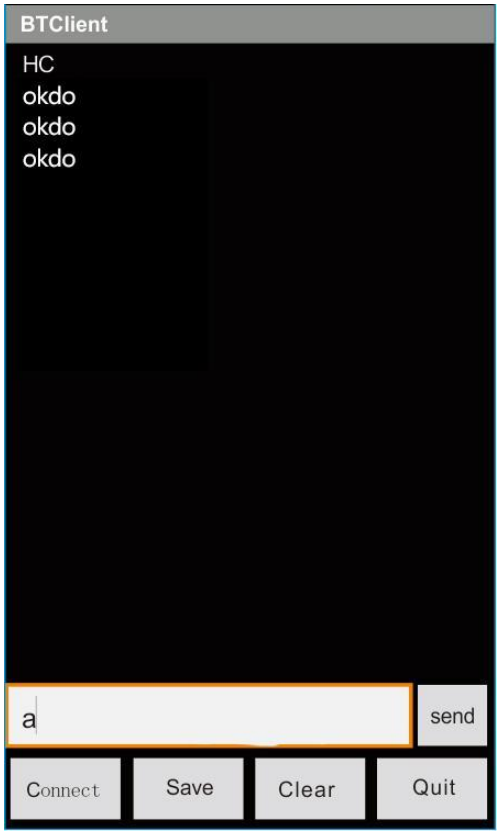
2. Transfer the file 

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).

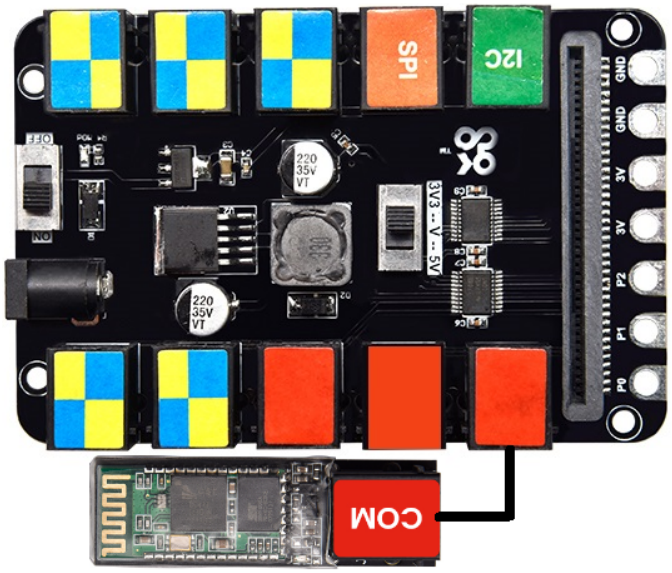


4. After connecting BTClient, when you send a, BTClient will show okdo and LED on the control board will flash.



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.

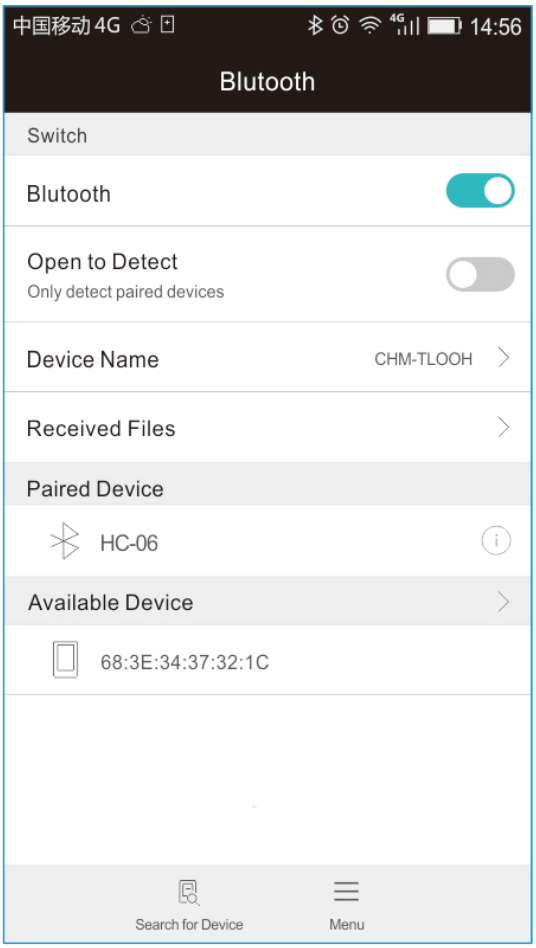
Test Code


```
.............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

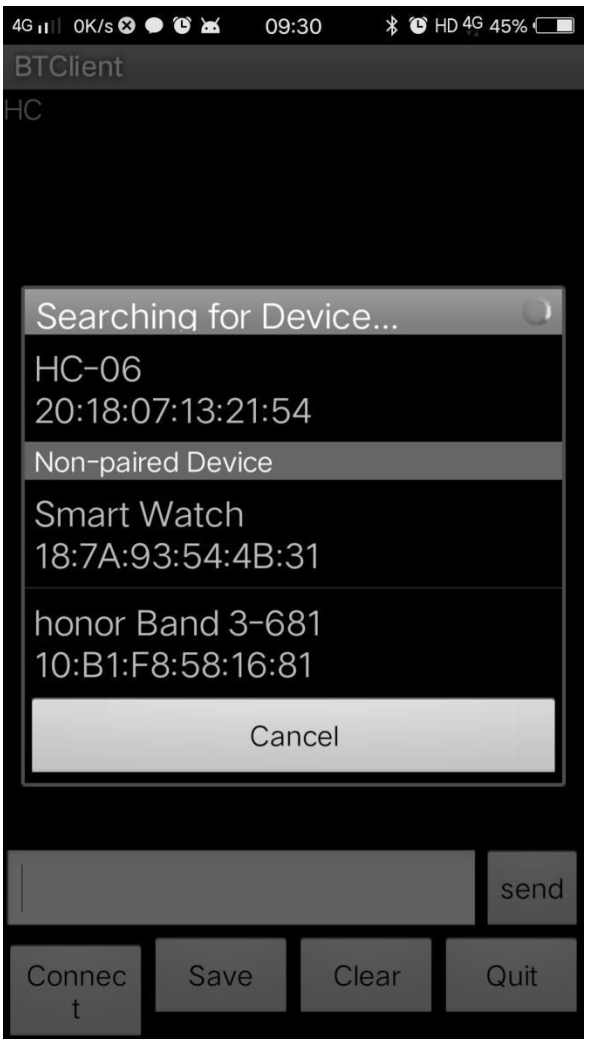
Turn on Bluetooth on your phone, pair the Bluetooth device, enter 1234, and you will see the paired device.



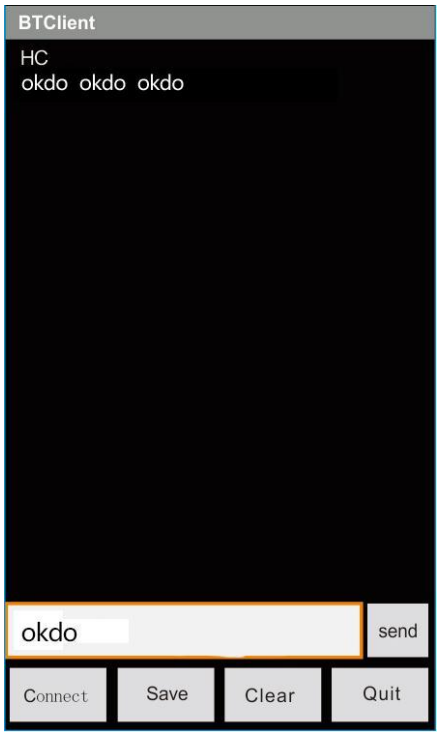
5. Transfer the file 

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).

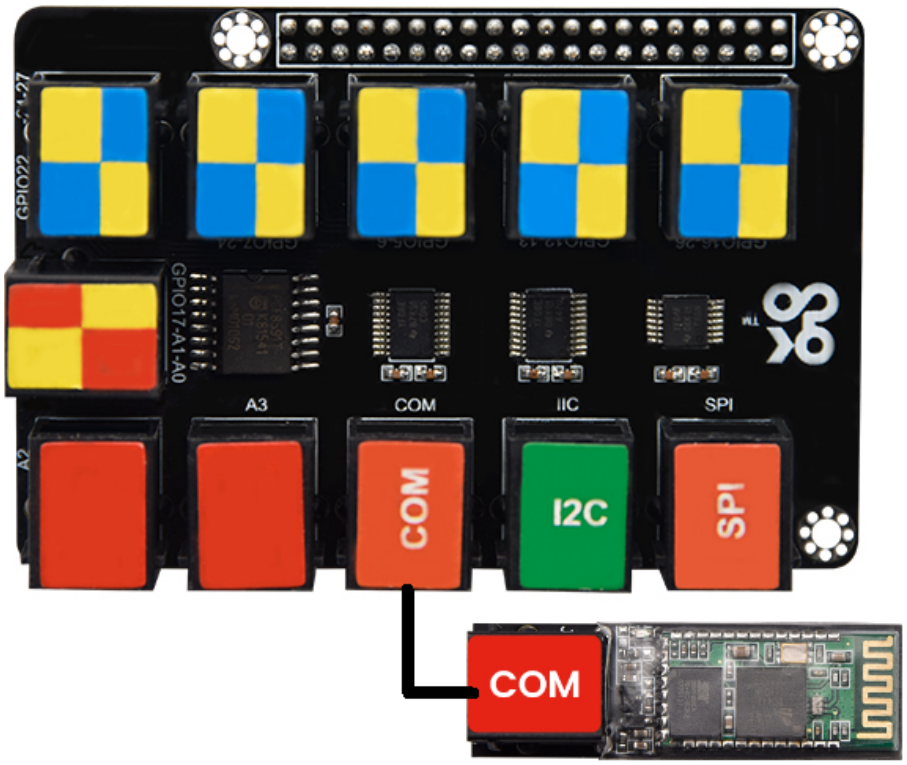


After connecting BTClient, when you send okdo, BTClient will show okdo .



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

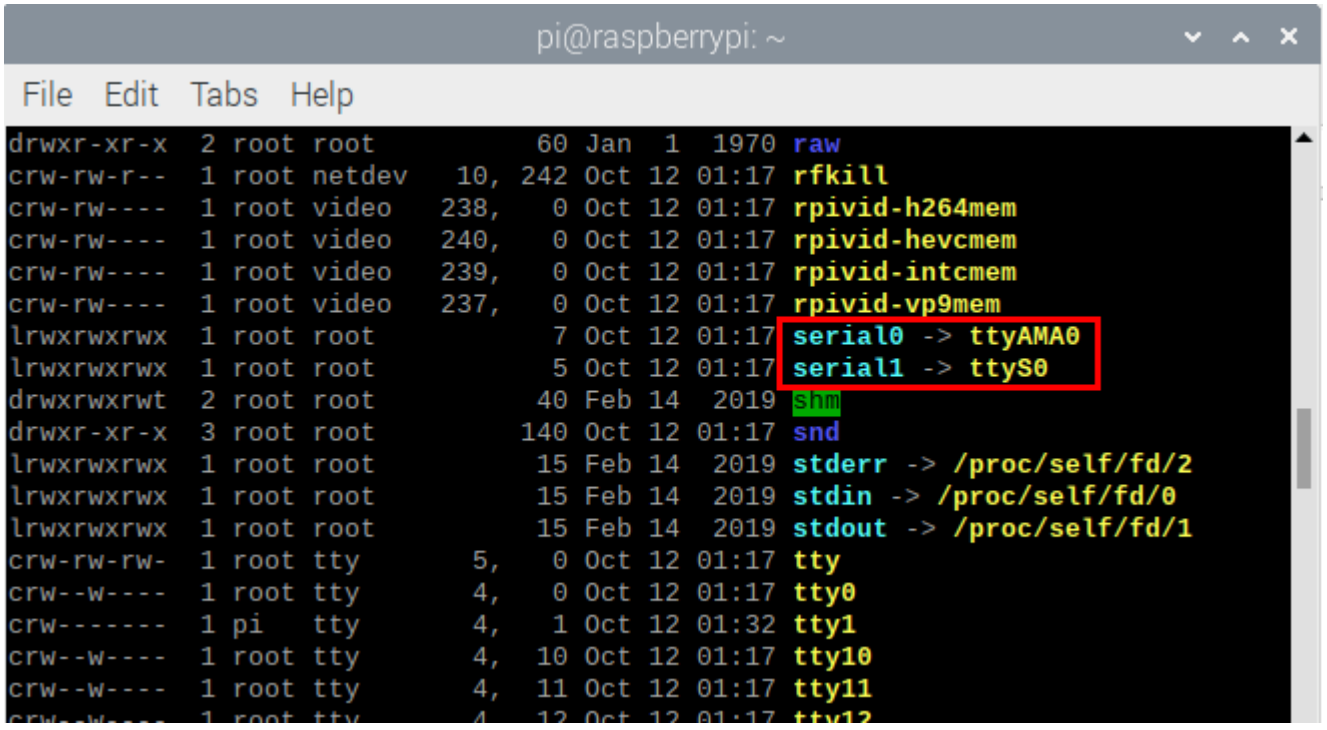
Raspberry Pi Application



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;



Serial ports serial0 - serial1 - are used for the Bluetooth module.

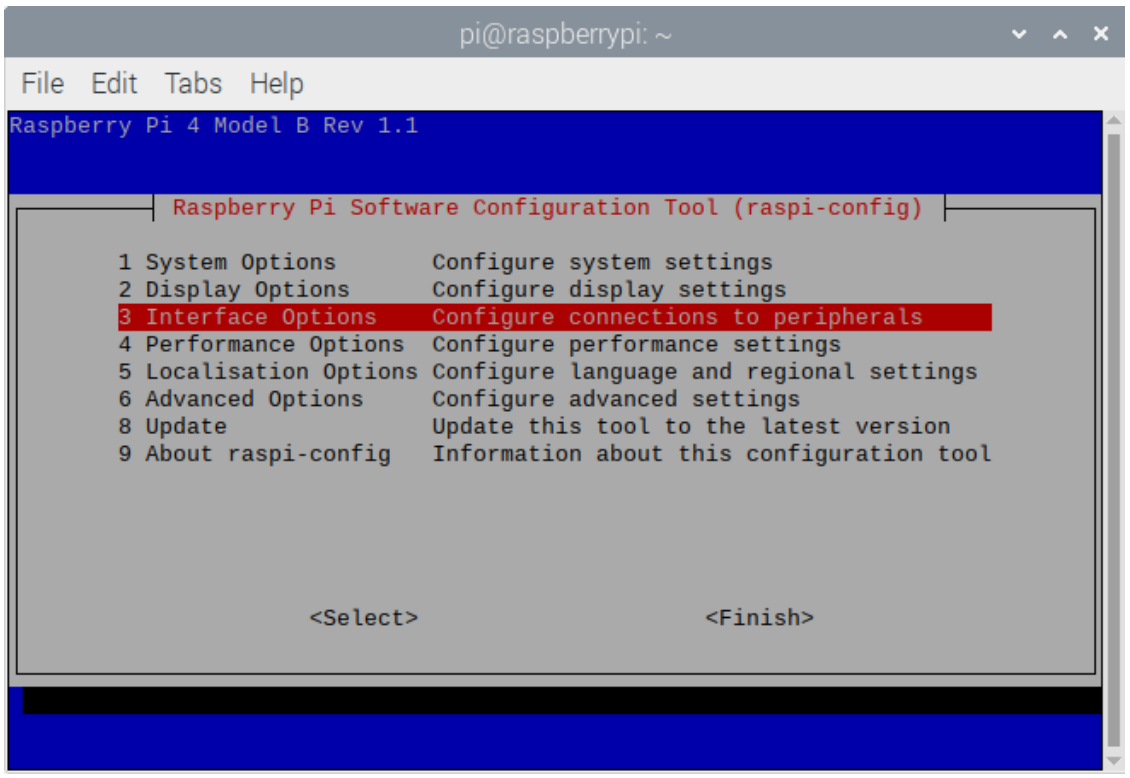
Set the hardware port to GPIO port.

First log in the command terminal

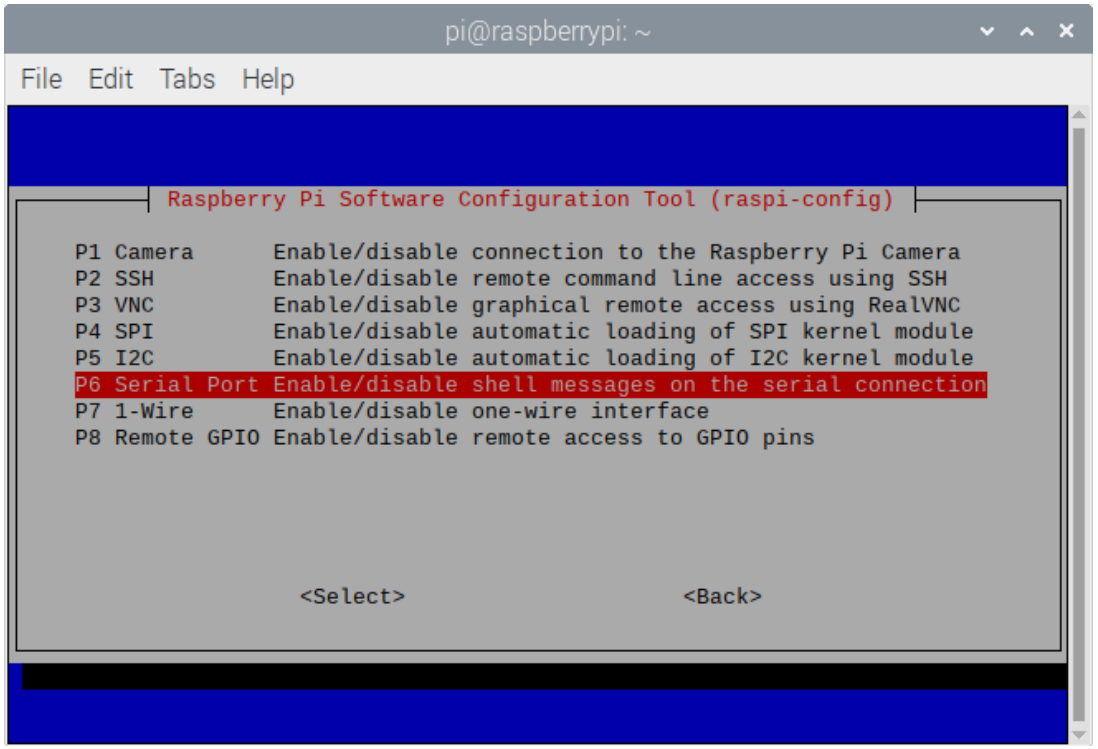
Input sudo raspi-config in the terminal to enter the Raspberry Pi configuration interface.

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

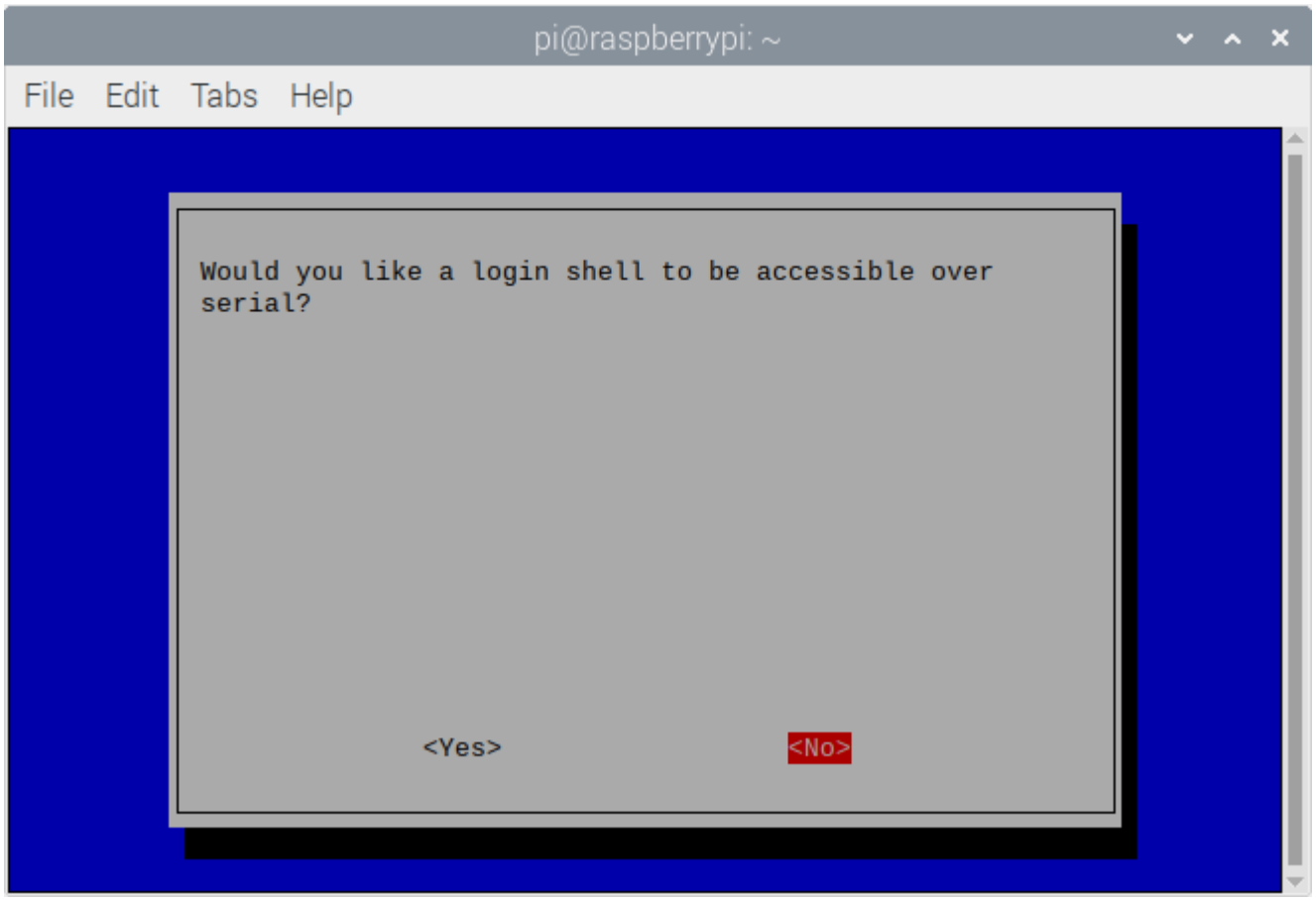
Select the third interface options: (press ←,↑,↓,→ then“Enter”)

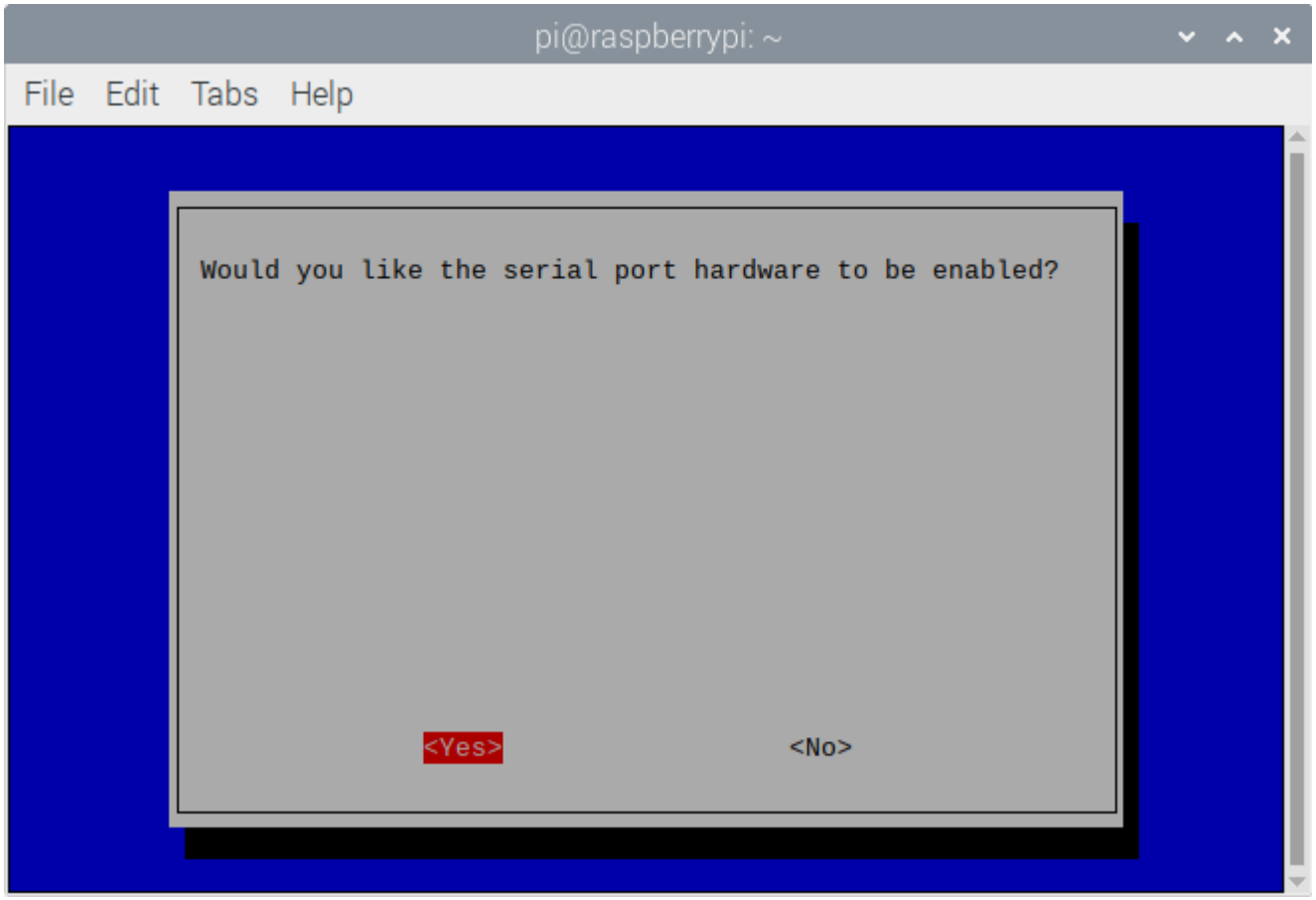


Enter P6 Serial Port

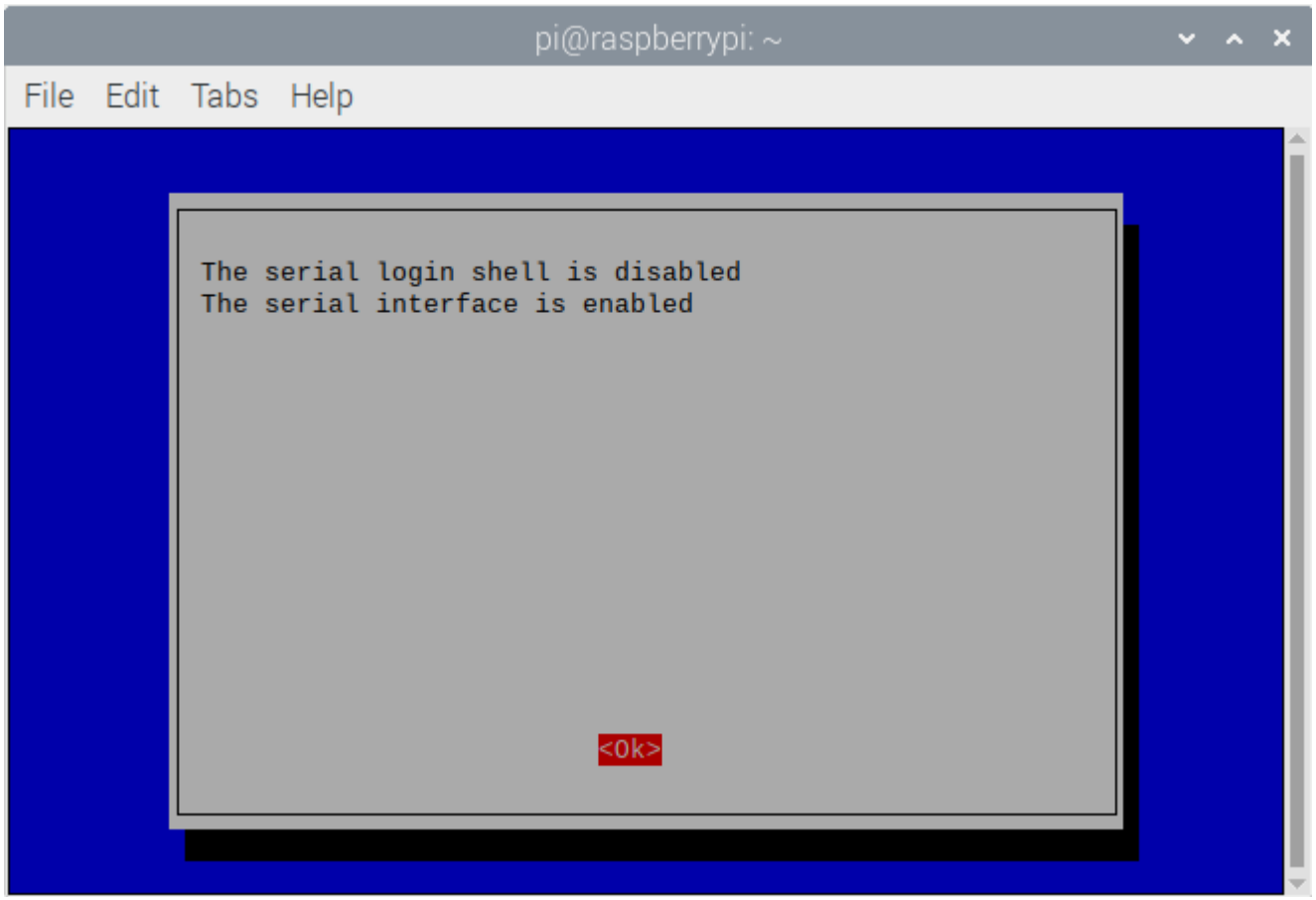


Select to turn off serial login function, and open hardware serial debug function:



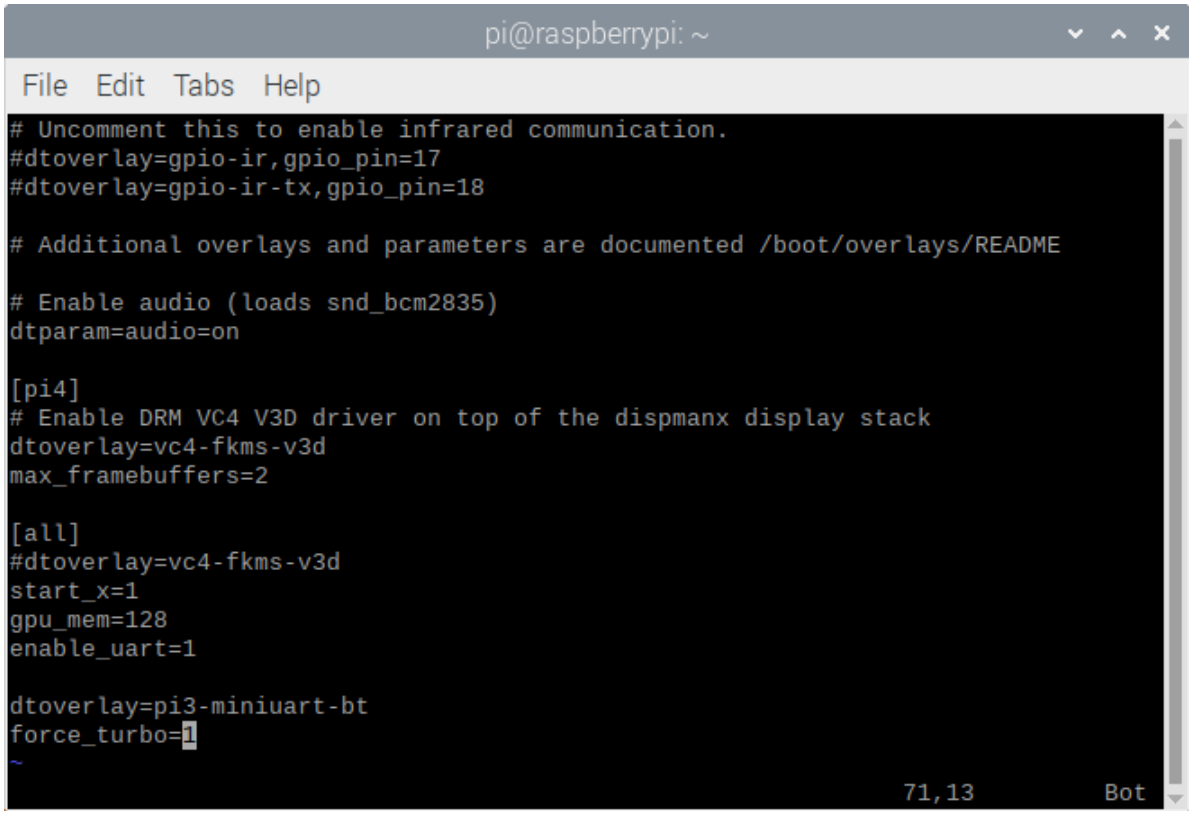


Then click OK

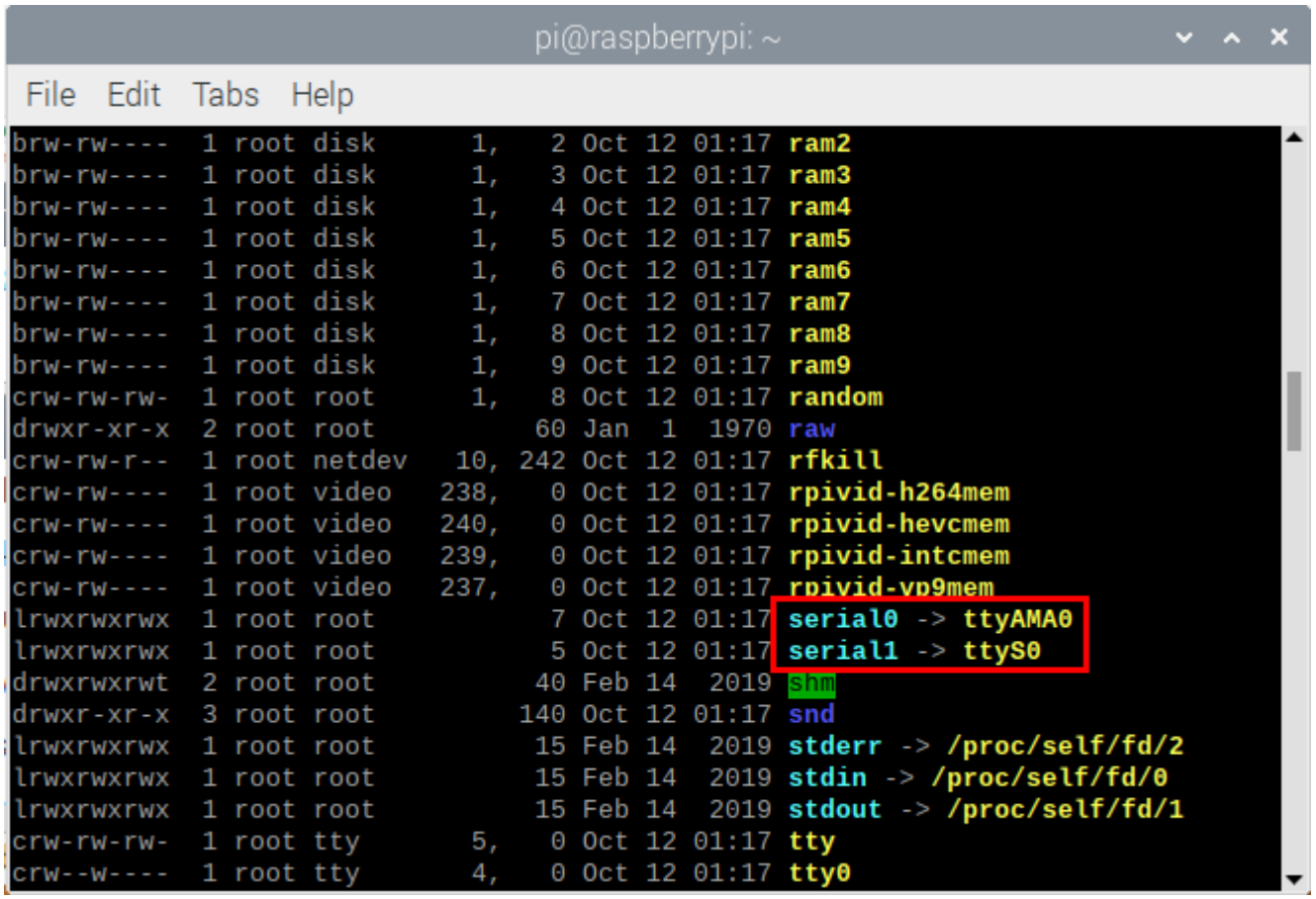


Set hardware serial port to GPIO port

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;

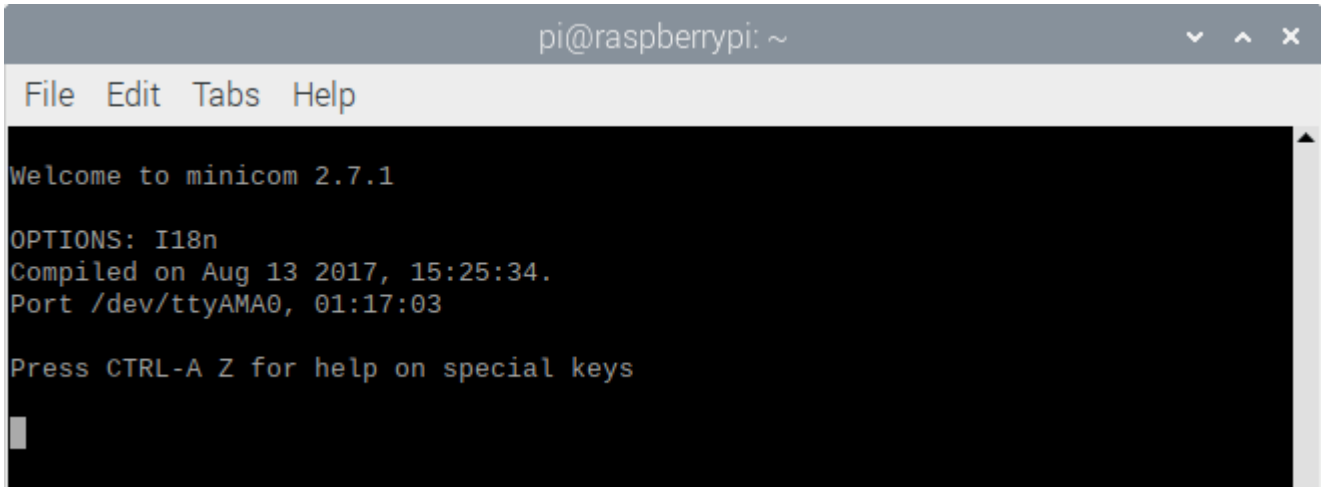


Input the rebooting command sudo reboot in the terminal. After rebooting Raspberry Pi, input ls /dev -al again. Then you will see two serial ports



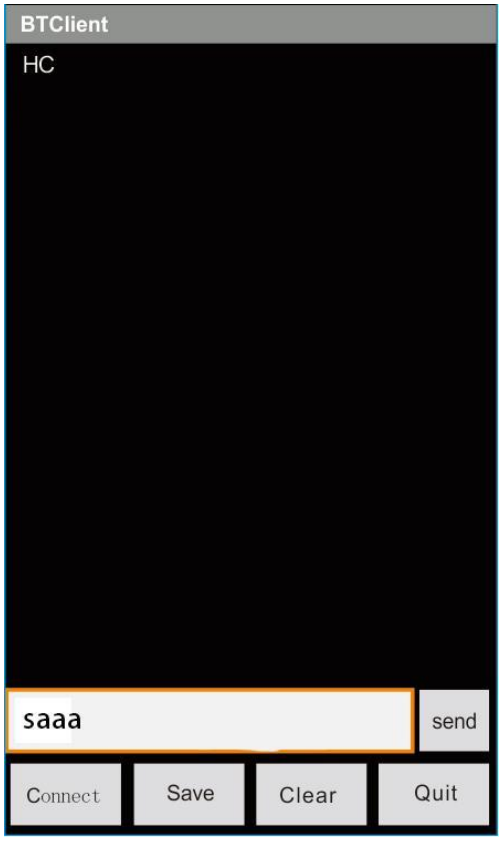
Minicom Serial Assistant Test

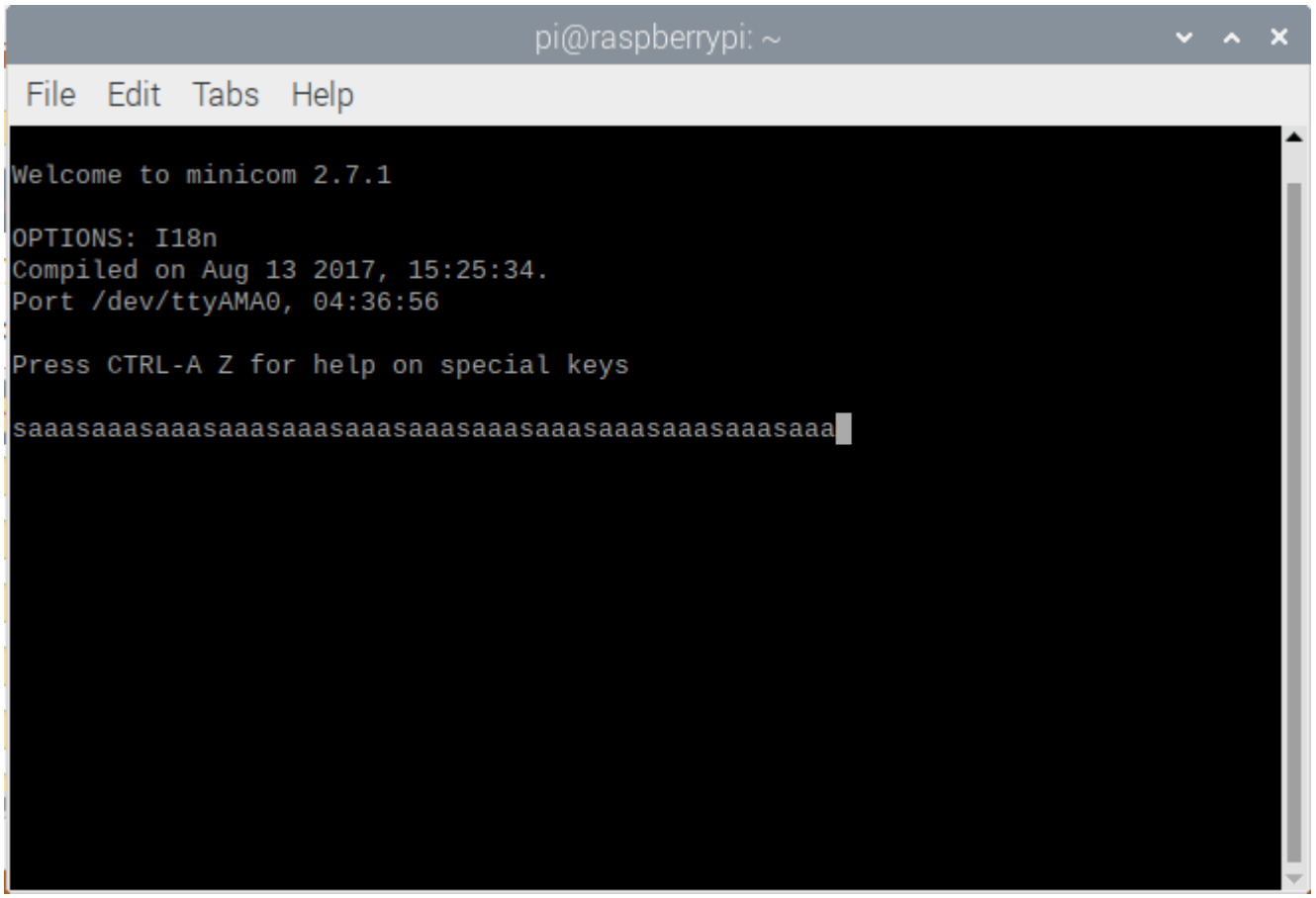
First input sudo apt-get install minicom in the terminal to install minicom. After installing, then input minicom -D /dev/ttyAMA0 -b 9600 to boot minicom, as shown below;



-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