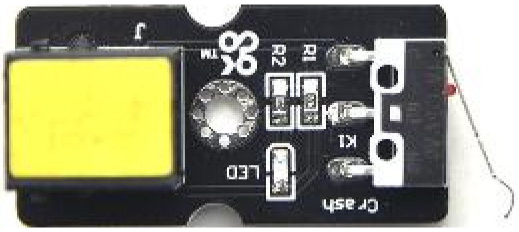
Collision Crash Sensor (000x0000 Article Number) (TS2139)

DESIGN THE WORLD

Product Details



This is the TelePort collision sensor with a metal switch, which can detect the collision. When the switch is pressed, the signal end will output low levels and LED will be on; on the contrary, LED will go off. It features all-round stability and high sensitivity.

Features and Benefits

- Compatible with RJ11 6P6C OKdo TelePort Control boards and expansion shields.
- This snap-action switch is actuated with just a gentle physical force for sensitive physical presence detection.
- The digital on/off output will interface to any digital input of a microcontroller with virtually unlimited use cases.

Technical Specifications

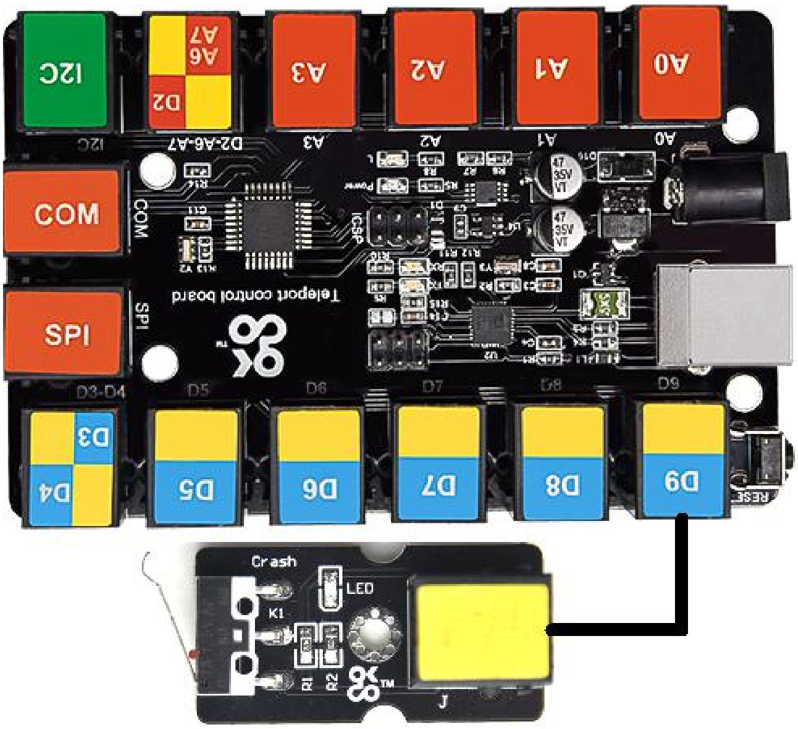
Sensor type	Digital input
Working voltage	3.3V-5V
Dimensions	44mm*20mm*18mm
Weight	5.5g

Applications

- Safety switch and level test of lift
- Gate interlock of microwave
- Automatic vending machine
- 3D photocopier positioning feedback
- Detect blocking problem in the photocopier

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

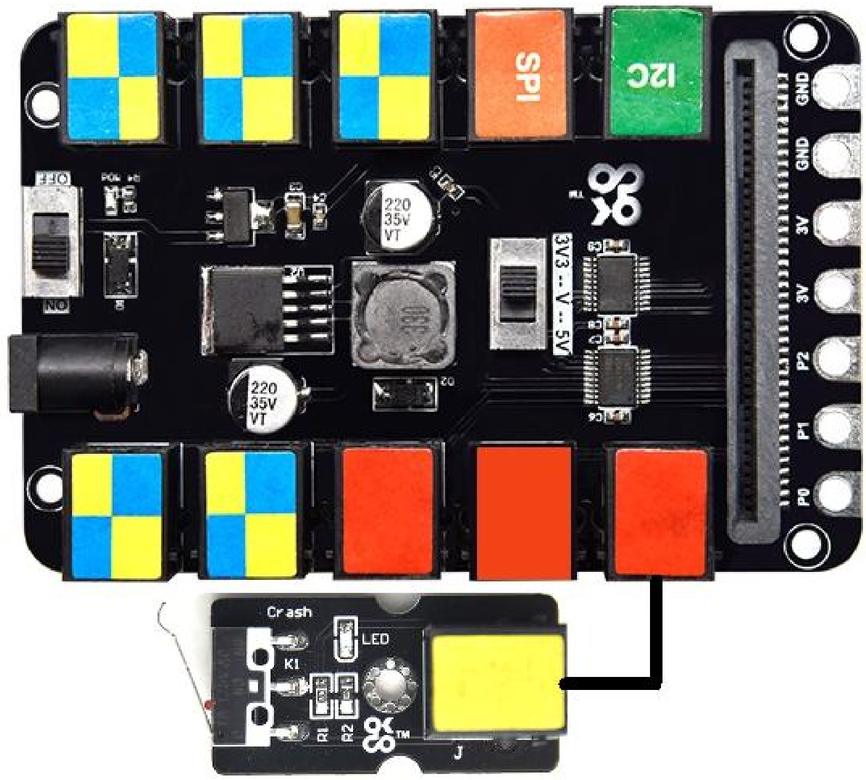
```
int Led=13;// set pin for LED
int Collision=9;// set pin for collision sensor
int val;// set digital variable val
void setup()
{
pinMode(Led,OUTPUT);// set pin LED as output
pinMode(Collision,INPUT);// set collision sensor as input
}
void loop()
{
val=digitalRead(Collision);// read value on pin 9 and assign it to val
if(val==HIGH)// when collision sensor detects a signal, turn LED on.
{
digitalWrite(Led,LOW);
} else
{
digitalWrite(Led,HIGH);
}
}
```

Test Result

Wire up, upload the test code and power it up. When you knock the metal switch with an object, D13 and the indicator of the sensor will light up; if not, indicators will go off.

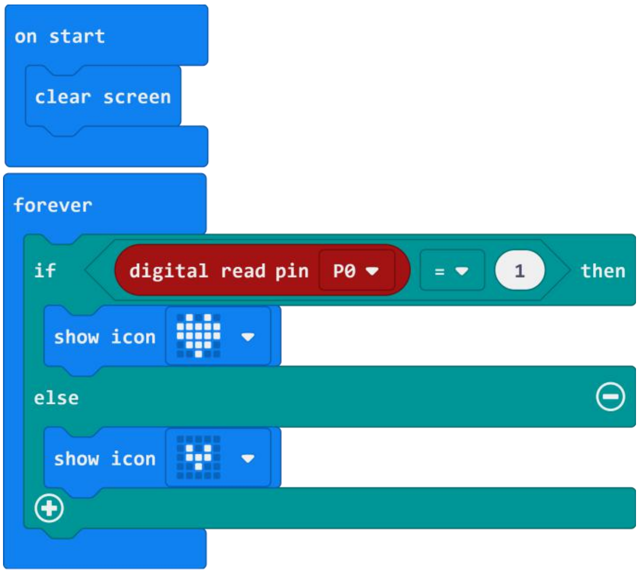
If you want to know more details about Arduino and the TelePort control board, you can refer to TS2178.

➢ Micro:bit Application



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

Test Code



```
on start
    clear screen

forever
    if  digital read pin P0 ▾  = ▾  1  then
        show icon ♥ ▾
    else
        show icon ▦ ▾
```

..............①Run the “on start” block to boot the program

..............②the LED matrix of the Micro:bit goes out

..............③The program is run circularly under the command of “forever” block

..............④When P0=1, the metal switch of the sensor is not collided, execute the next process

..............⑤Micro:bit will display “♥ ”

..............⑥When P0=0, the sensor is collided by an object, execute the code under else.

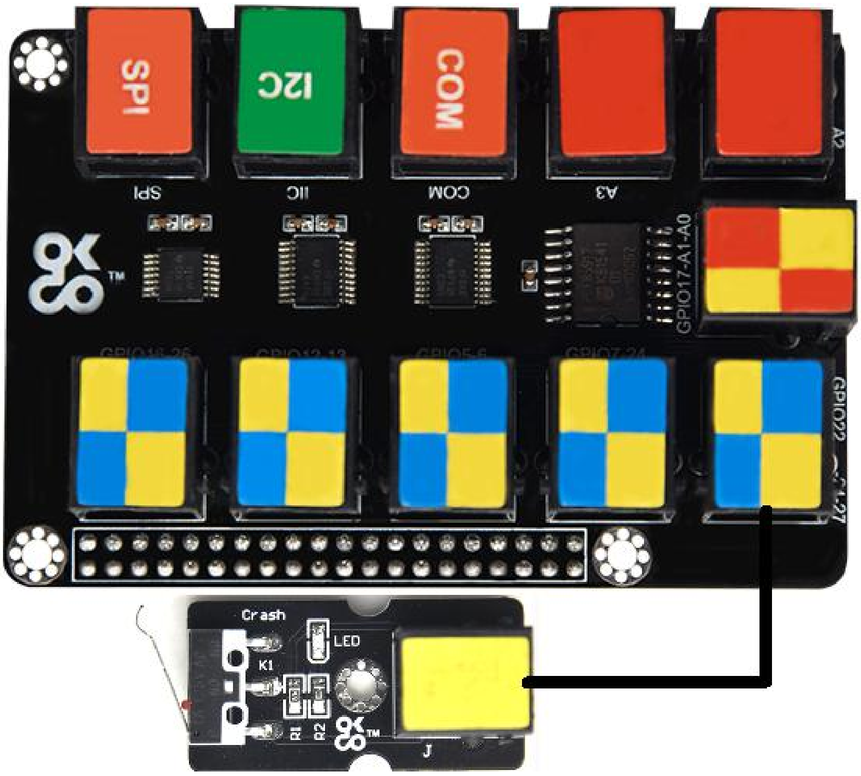
..............⑦Micro:bit will show ▦”

Test Result

Wire up, insert the Micro:bit V2.0 into the shield, turn DIP switch to 3V3, upload test code and power it up. When the metal switch is not touched by the object, the Micro:bit will show“♥ ”; otherwise, the Micro:bit will display the image“▦”.

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

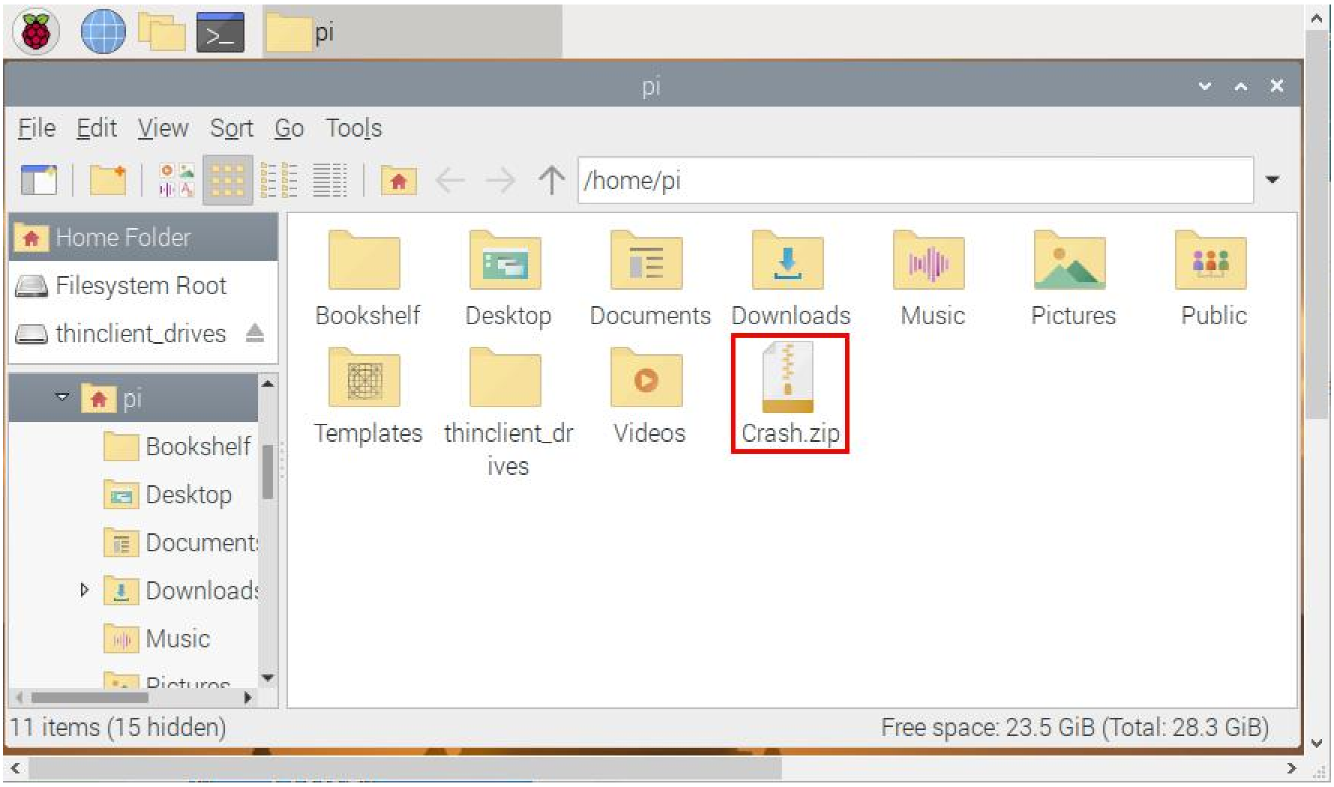
➢ Raspberry Pi Application

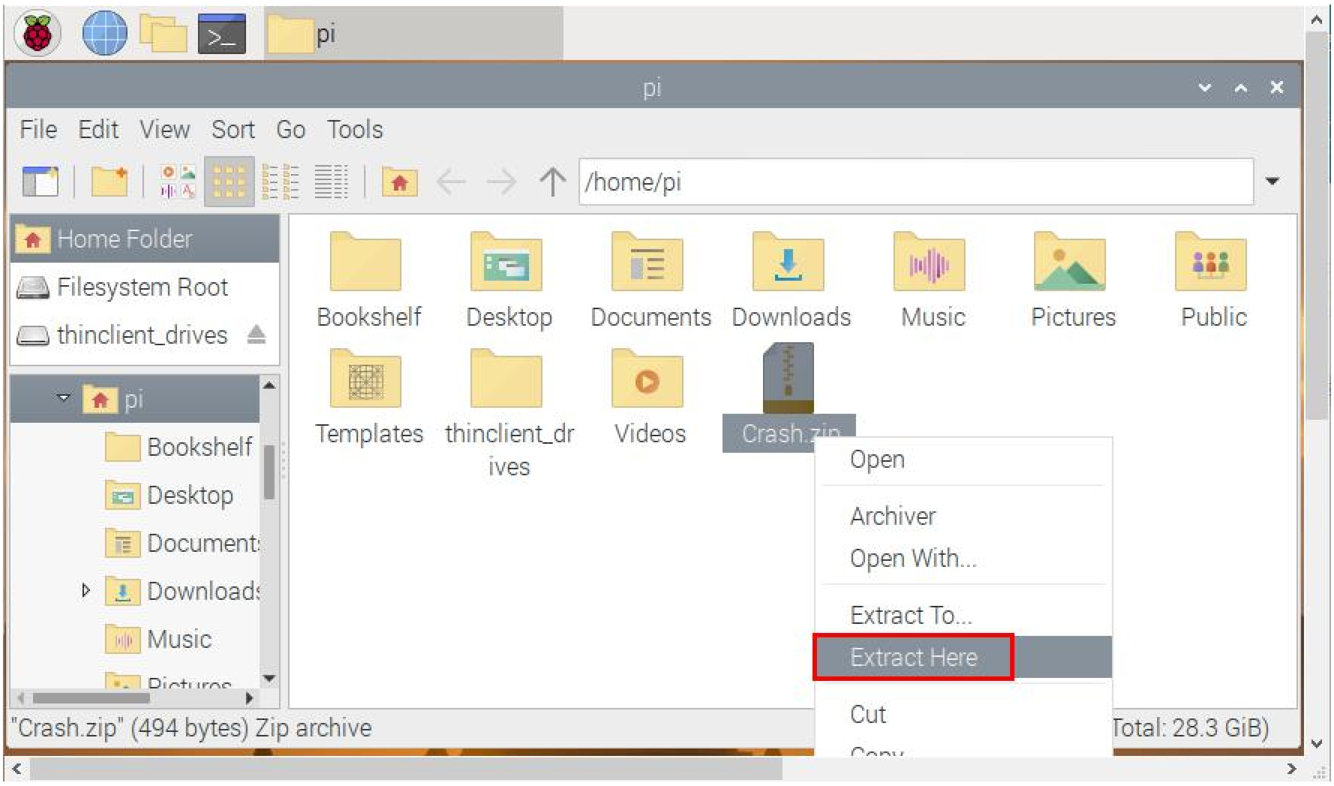


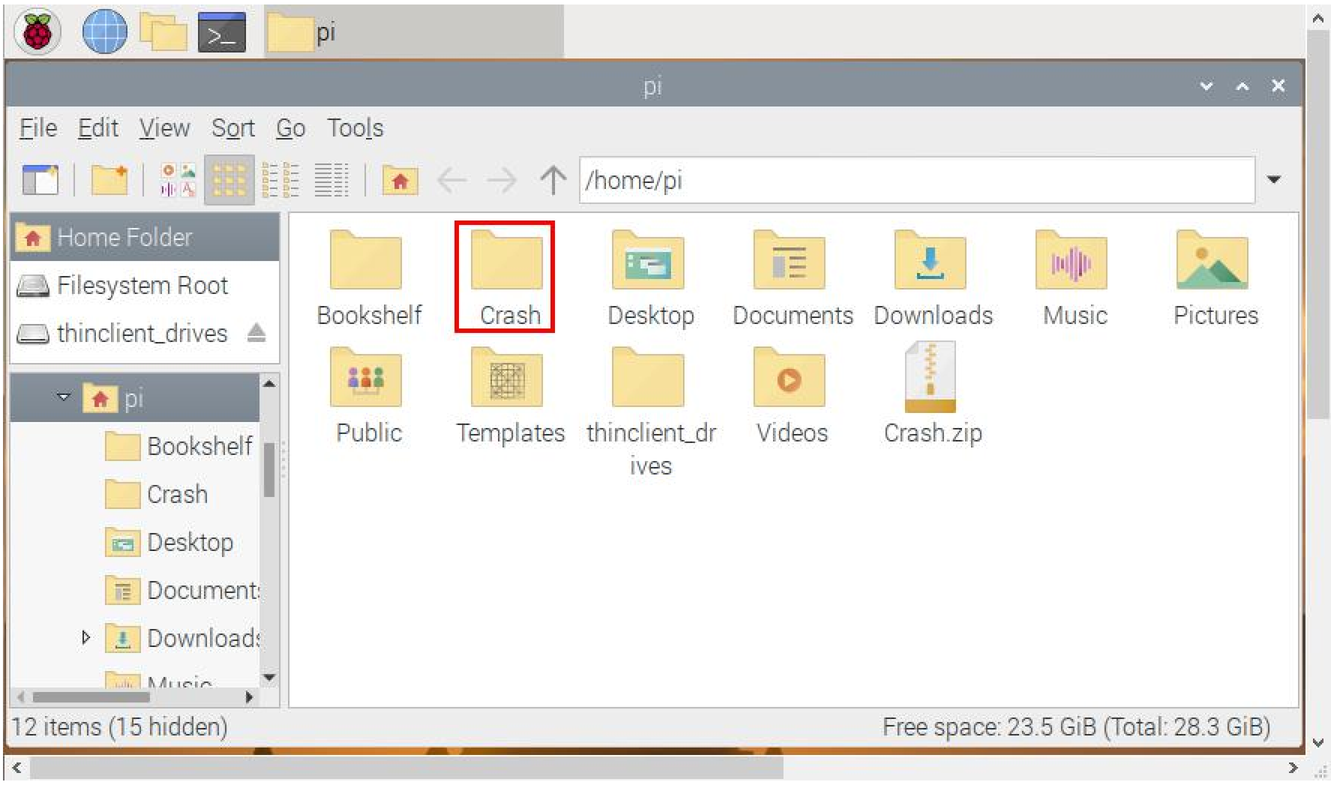
This module is compatible with the Raspberry Pi board and the TS2180 Raspberry Pi shield.

Copy the test code to Raspberry Pi system to run it

(1) Save the test code in the **pi** folder of Raspberry Pi system. Then place the Crash.zip file we provide in the **pi** folder, right-click and click **Extract Here.** As shown below:







(2) Compile and run test code：

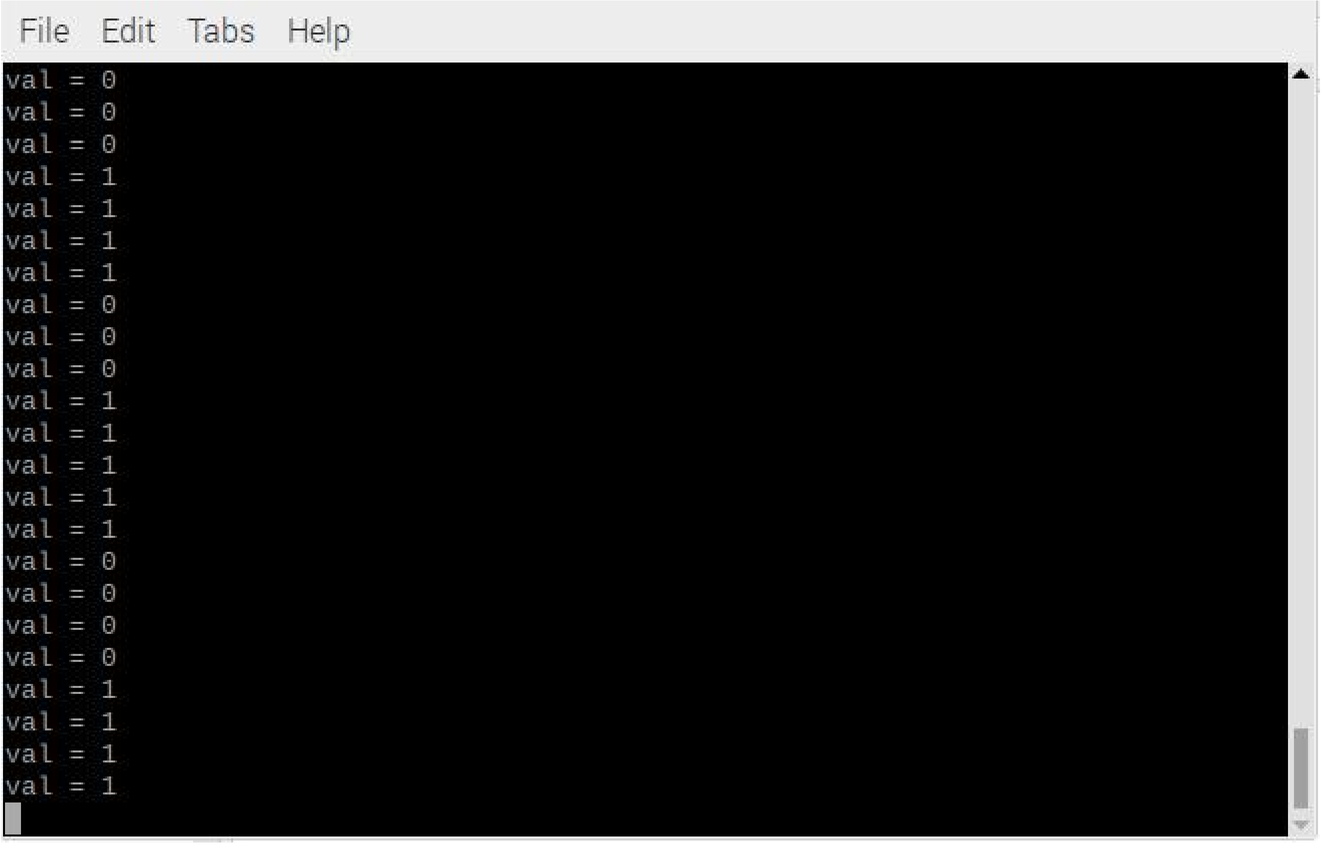
Input the following code and press“Enter”

```
cd /home/pi/Crash
gcc Crash.c -o Crash -lwiringPi
sudo ./Crash
```

(3) Test Result：

Insert the shield into the Raspberry Pi board. After programming finishes, when the shrapnel of switch is pressed, the terminal will print 0; on the contrary, the terminal will show 1.

Note: press Ctrl + C to exit code running



```
val = 0
val = 0
val = 0
val = 1
val = 1
val = 1
val = 1
val = 0
val = 0
val = 0
val = 1
val = 1
val = 1
val = 1
val = 1
val = 0
val = 0
val = 0
val = 0
val = 1
val = 1
val = 1
val = 1
```

Test Code

File name: Crash.c

```
#include <wiringPi.h>
#include <stdio.h>
#define crash 3   //crash pin BCM GPIO 22
int main()
{
  wiringPiSetup();
  char val;
  {
    pinMode(crash,INPUT);  //set the crash pin INPUT mode
  }

  while(1)
  {
    val=digitalRead(crash);  // digital read
    printf("val = %d\n", val);
    delay(50);
  }
}
```

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