

Capacitive Touch Sensor (000x0000 Article Number) (TS2141)



Product Details

The TelePort capacitive touch sensor adopts the most advanced capacitive sensing technology. Its touch area is based the touch detection chip and can be used a switch.



This sensing area can "feel" the touch of people and metal and send high/low voltage levels. Although it is separated by objects,

it still can feel the touch. The sensitivity will decrease as the thickness of the object rise.

It features the low power consumption and wide working voltage. In addition, after booting module, it will take about 0.5 seconds to stabilize.

Features and Benefits

- Compatible with RJ11 6P6C OKdo TelePort Control boards and expansion shields.
- Digital capacitive touch sensor can be connected to any digital input on a microcontroller.
- This sensor can "feel" people or metal objects to produce a high/low voltage level.

Technical Specifications

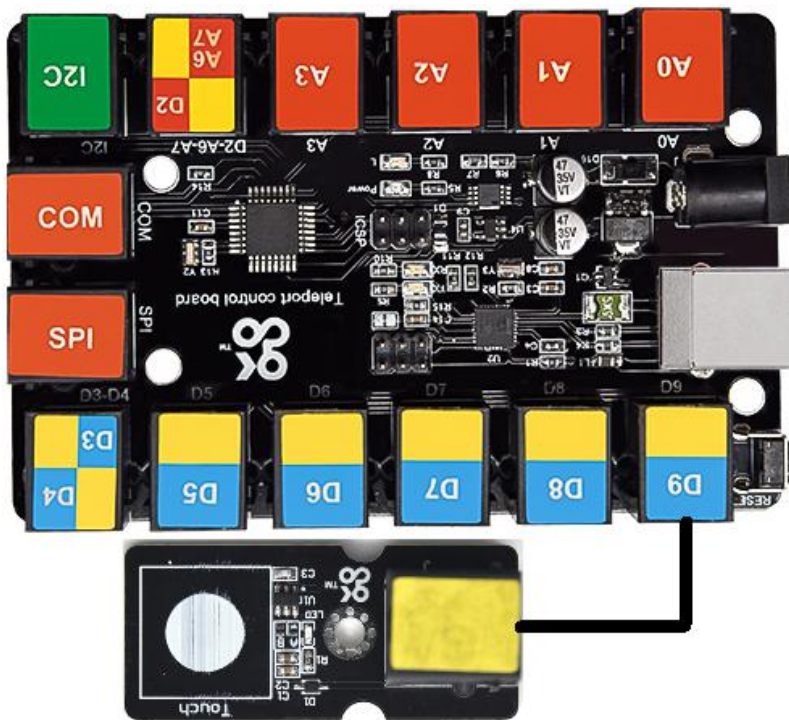
Sensor type	Digital input
Working voltage	3.3V-5V
Dimensions	44mm*20mm*18mm
Weight	5.5g

Applications

- PC Peripherals
- MP3 Players
- Remote Controls
- Mobile Phones
- Lighting Controls

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

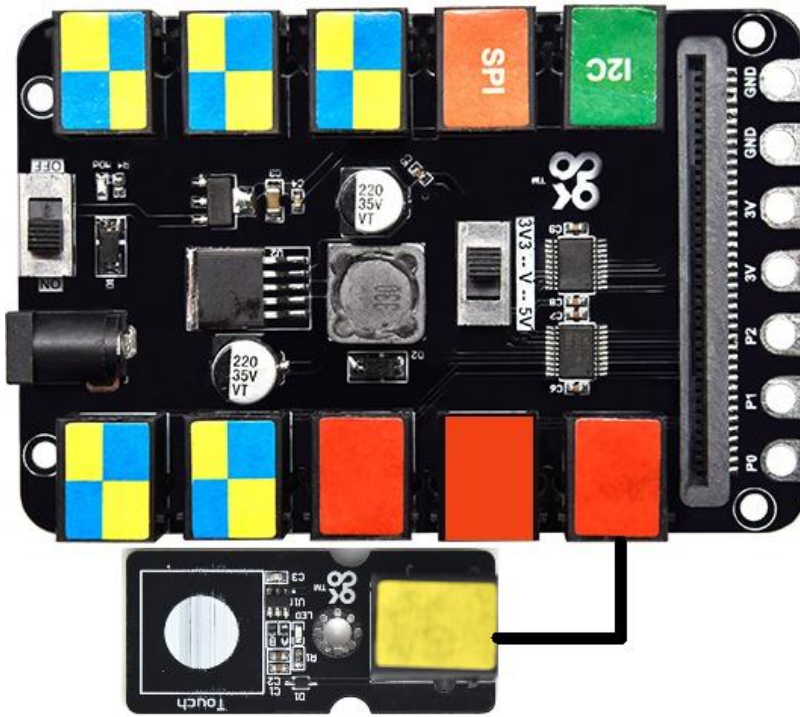
```
int ledPin = 13; // Connect LED on pin 13, or use a built-in led.
int KEY = 9; // Connect Touch sensor on Digital Pin 9
void setup(){
  pinMode(ledPin, OUTPUT); // Set ledPin to output mode
  pinMode(KEY, INPUT); // Set touch sensor pin to input mode
}
void loop(){
  if(digitalRead(KEY)==HIGH) { // Read Touch sensor signal
    digitalWrite(ledPin, HIGH); // if Touch sensor is HIGH, then turn led on
  }
  else{
    digitalWrite(ledPin, LOW); // if Touch sensor is LOW, then turn led off
  }
}
```

Test Result

Wire up, upload test code and power it up. When you touch the sensing area, D13 on the main board will be on; on the contrary, the indicator will go off.

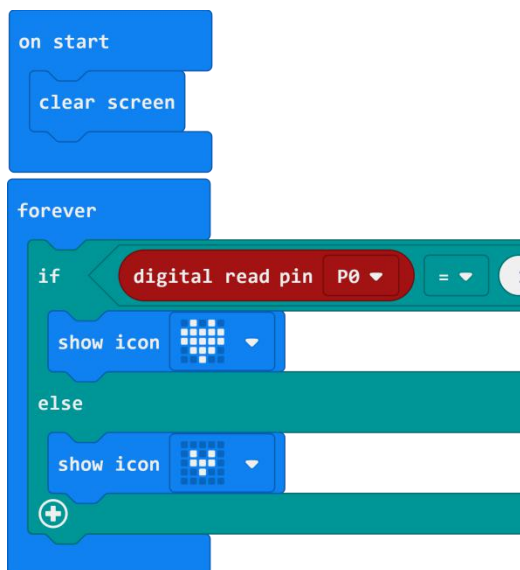
If you want to know more details about Arduino and the TelePort control board, you can refer to TS2178.

➤ Micro:bit Application



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

Test Code



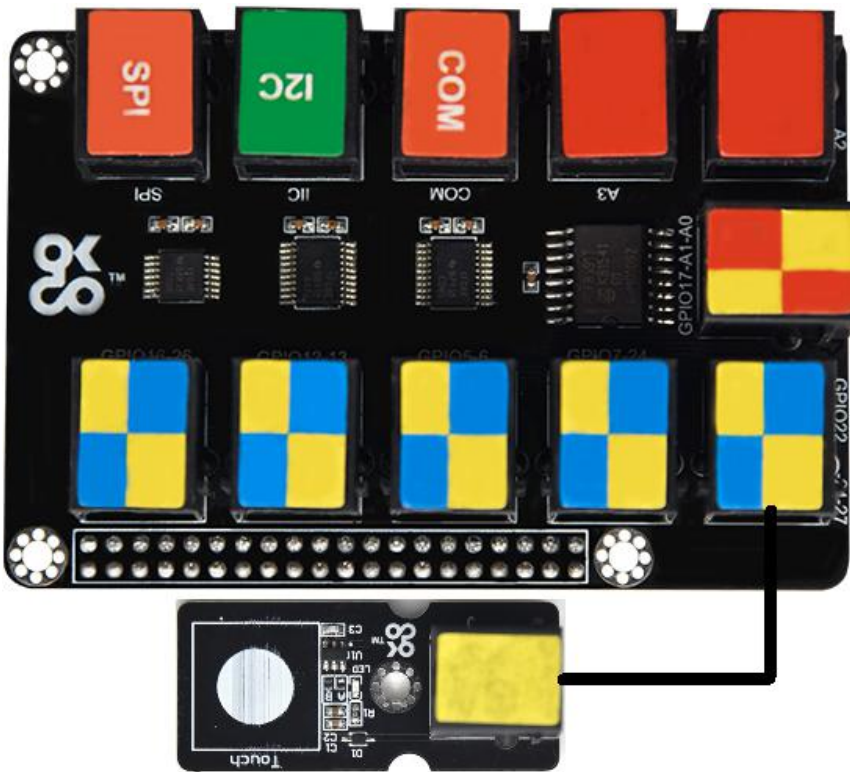
.....①Run the “on start” block to boot the program
.....②the LED matrix of the Micro:bit turns off
.....③The program is run circularly under the command of “forever” block
.....④When P0=1, which means that the sensing area is touched, then execute the program under then
.....⑤the Micro:bit will display “♥”
.....⑥When P0=0, which means that the sensing area is not touched, then execute the program under else.
.....⑦the Micro:bit will show “4x4 grid”

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 you touch the sensing area, the Micro:bit will show“♥”; if not, the image“4x4 grid”will be displayed.

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

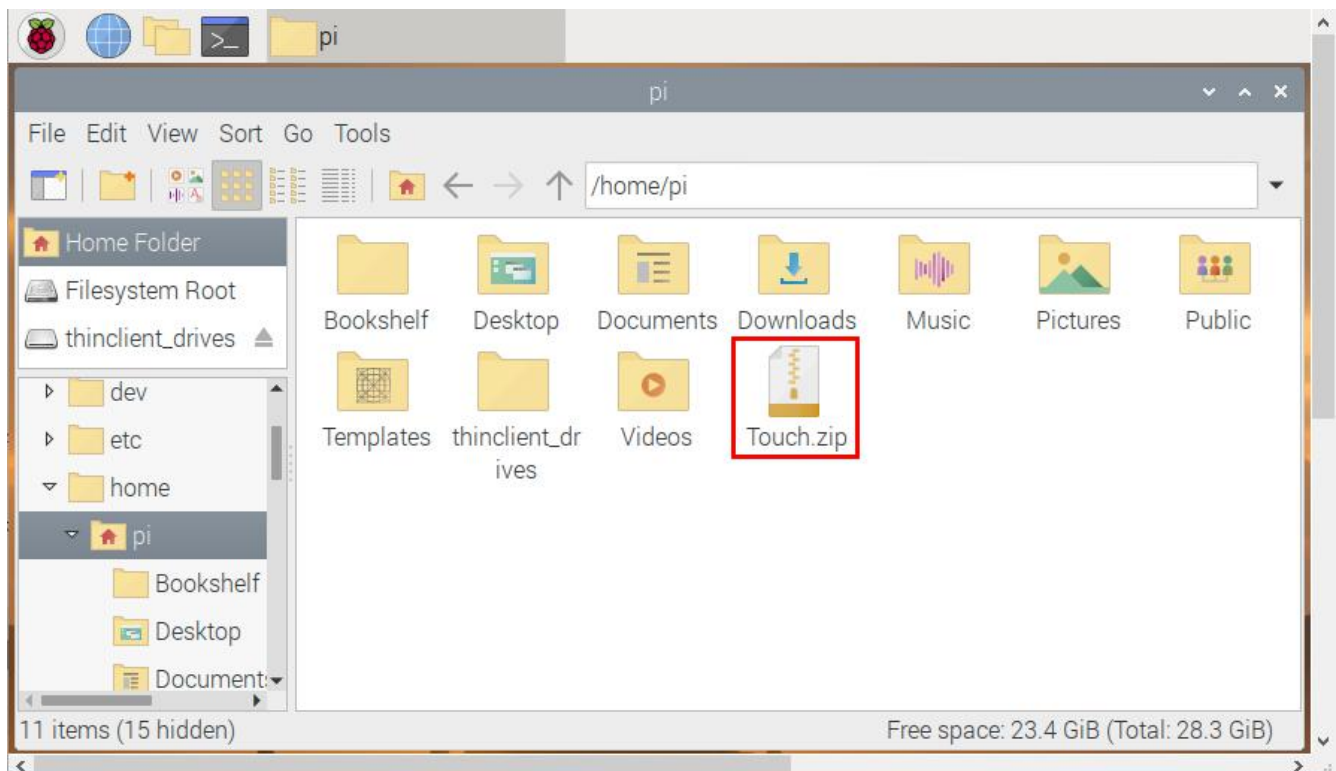
➤ Raspberry Pi Application

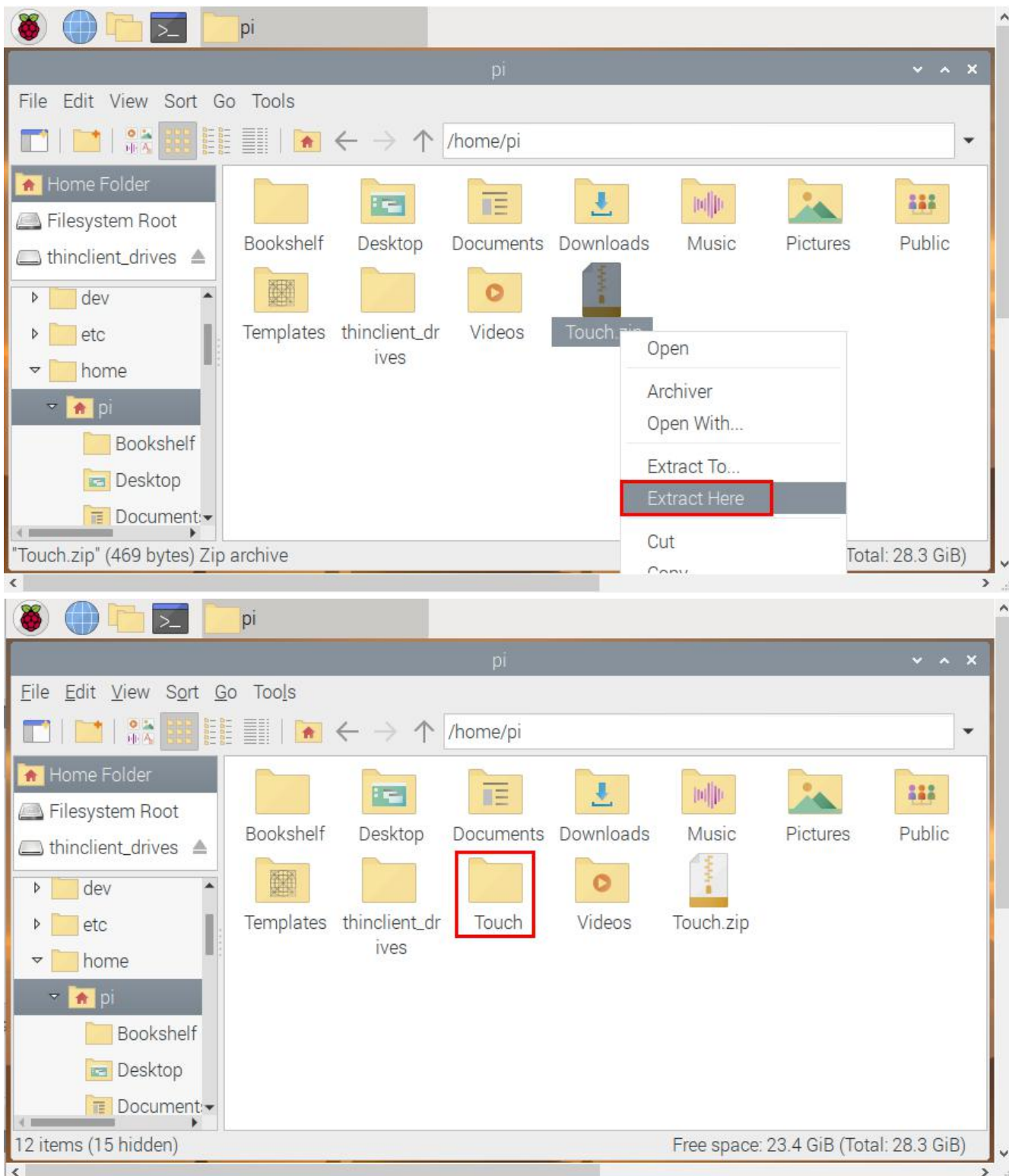


This module is compatible with the Raspberry Pi board and the TS2180 Raspberry Pi shield.

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 **Touch.zip** file we provide in the **pi** folder, right-click and click **Extract Here**. As shown below:





(2) Compile and run test code:

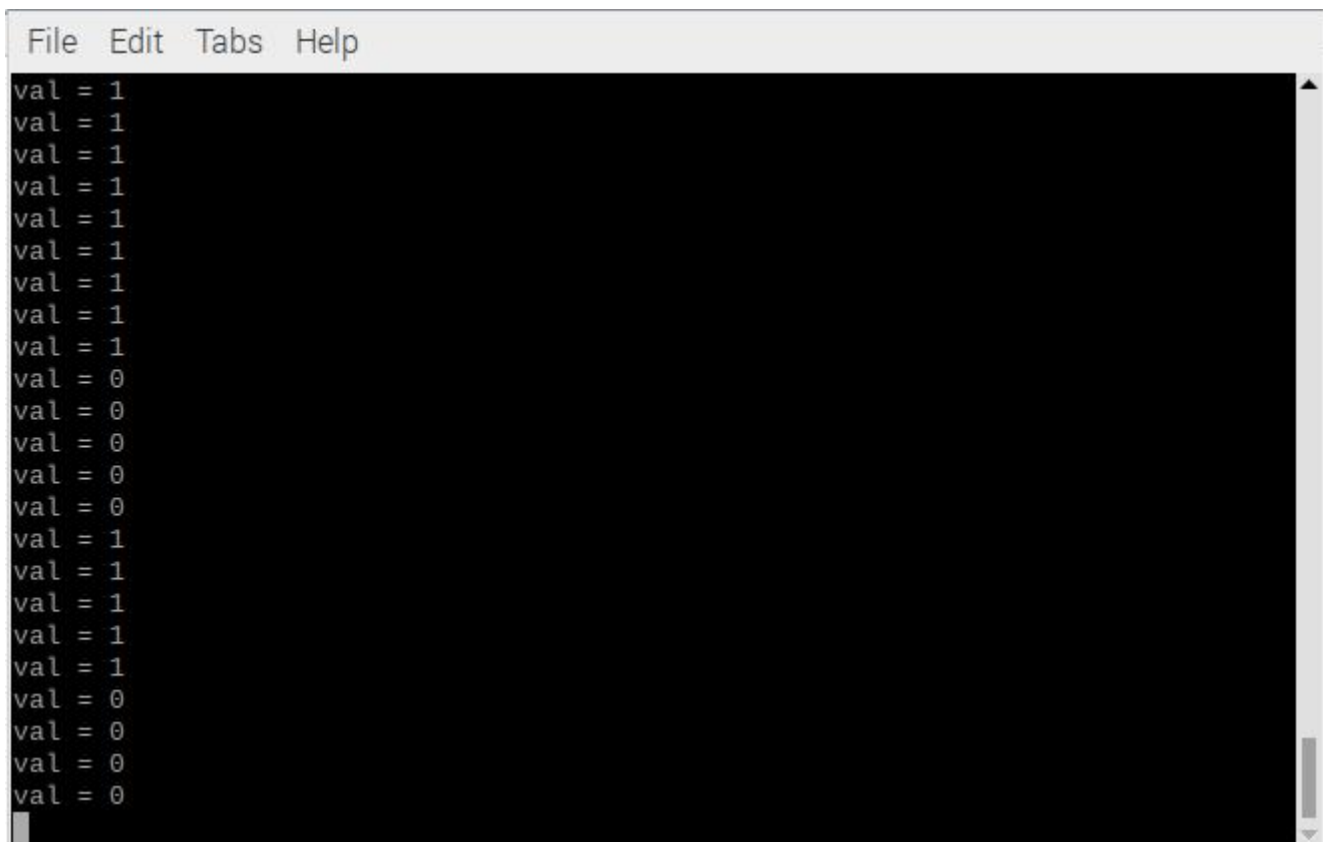
Input the following code and press "Enter"

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

(3) Test Result :

Insert the shield into the Raspberry Pi board. After programming finishes, when you touch the sensing area, the terminal will print 1; on the contrary, the terminal will display 0.

Note: press Ctrl + C to exit code running



```
File Edit Tabs Help
val = 1
val = 1
val = 1
val = 1
val = 1
val = 1
val = 1
val = 1
val = 1
val = 1
val = 0
val = 0
val = 0
val = 0
val = 0
val = 1
val = 1
val = 1
val = 1
val = 1
val = 0
val = 0
val = 0
val = 0
```

Test Code

File name: **Touch.c**

```
#include <wiringPi.h>
#include <stdio.h>
#define touchPin 3 //BCM GPIO 22

int main()
{
    wiringPiSetup();
    char val;
    {
        pinMode(touchPin,INPUT);
    }

    while(1)
    {
        val=digitalRead(touchPin);
        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.

END