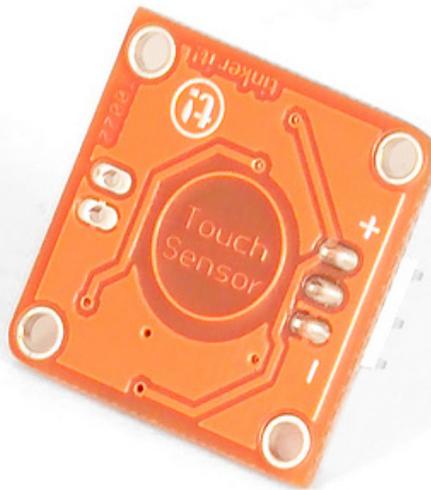




TinkerKit!

TinkerKit Touch Sensor



Overview

The **Touch Sensor** is sensitive to skin contact.

Output: This module normally outputs 0v, but when touched, sends 5v. When connected to an input on the Arduino using the TinkerKit Shield, you will see 0 when there is no touch, and 1023 when touched.

Module Description: on the back of the module you can find a signal amplifier, a capacitor, a [QT 100A single touch controller](#), a green LED that signals that the module is correctly powered and a yellow LED whose brightness depends on the values output by the module.

Please note this device performs an auto calibration when it is turned on, so if someone is touching the switch surface when it is turned on it will not work. To reset, cycle power and make sure no one is touching it as you restart.

This module is a **SENSOR**. The connector is an **OUTPUT** which must be connected to one of the **INPUT** connectors on the **TinkerKit Shield**.

Datasheet: [QT100A](#)

Code Example

```
/*
Analog input, analog output, serial output

Reads an analog input pin; a T000220 Touch Sensor connected to I0, maps
the result to a range from 0 to 255
and uses the result to set the pulsewidth modulation (PWM) on a T010111
LED Module connected on O0.
Also prints the results to the serial monitor.

created 29 Dec. 2008
Modified 4 Sep 2010
by Tom Igoe
modified 7 dec 2010
by Davide Gomba

This example code is in the public domain.

*/

#define O0 11
#define O1 10
#define O2 9
#define O3 6
#define O4 5
#define O5 3
#define I0 A0
#define I1 A1
#define I2 A2
#define I3 A3
#define I4 A4
#define I5 A5

// These constants won't change. They're used to give names
// to the pins used:
const int analogInPin = I0; // Analog input pin that the Touch Sensor is
attached to
const int analogOutPin= O0; // Analog output pin that the LED is attached
to

int sensorValue = 0; // value read from the Touch Sensor
int outputValue = 0; // value output to the PWM (analog out)

void setup() {
// initialize serial communications at 9600 bps:
Serial.begin(9600);
}
```

```
void loop() {  
  // read the analog in value:  
  sensorValue = analogRead(analogInPin);  
  // map it to the range of the analog out:  
  outputValue = map(sensorValue, 0, 1023, 0, 255);  
  // change the analog out value:  
  analogWrite(analogOutPin, outputValue);  
  
  // print the results to the serial monitor:  
  Serial.print("sensor = " );  
  Serial.print(sensorValue);  
  Serial.print("\t output = ");  
  Serial.println(outputValue);  
  
  // wait 10 milliseconds before the next loop  
  // for the analog-to-digital converter to settle  
  // after the last reading:  
  delay(10);  
}
```