

How to connect Arduino to the Pressure Sensitive Fabric Switch

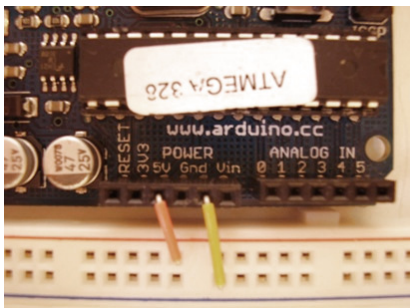
This is one of the ways you can connect the Pressure Sensitive Fabric Switch to an Arduino. Any schematic you can find to connect a switch to Arduino works fine, with the only exception that you need to increase the debouncing time.

In addition to the Pressure Sensitive Fabric Switch, you need:

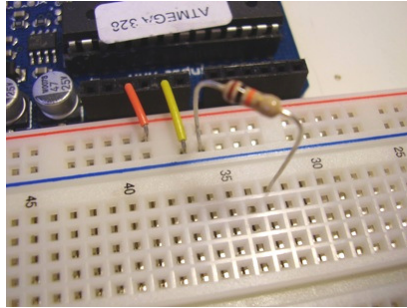
- Arduino
- breadboard
- wires
- an LED
- a 10 kOhm resistor
- a 1 kOhm resistor
- power supply for Arduino, or a PC with a USB port

This schematic uses two resistors, a pull down 10 kOhm resistor and a 1 kOhm resistor.

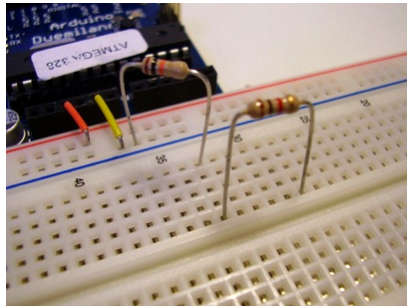
Start by connecting positive and negative breadboard buses to Arduino's 5V and ground, GND:



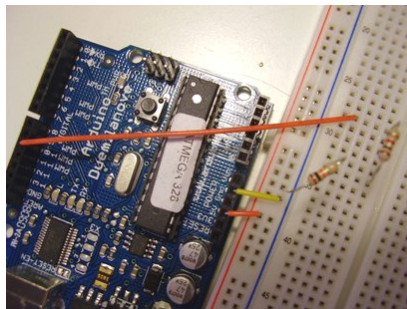
Place the 10 kOhm resistor between the GND and another row on the breadboard, shown here:



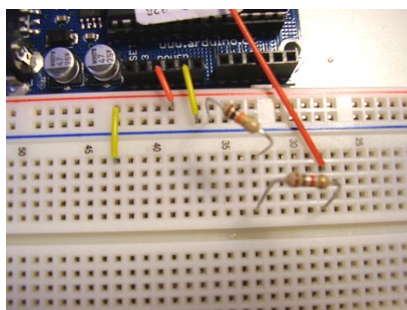
Now place the other resistor as in the following picture:



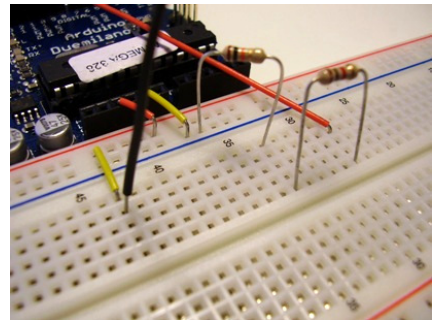
Next, connect the circuit to Arduino's pin 10 with a short wire (because in our sketch it is the 10 input that the Arduino will read):



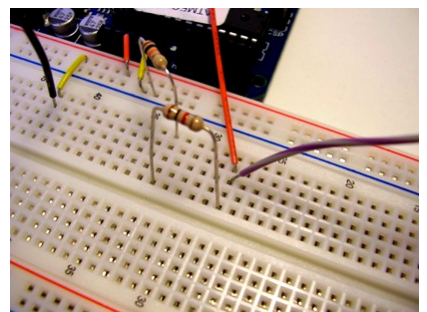
Now, connect 5V to one of the rows (yellow):



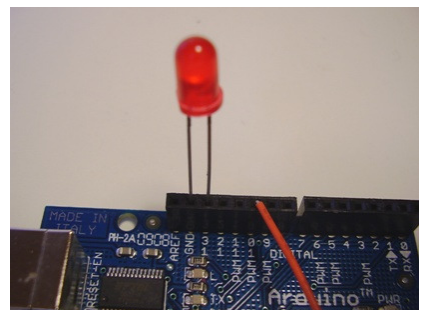
Connect one of the wires of the fabric switch to that same (5V) row, see black wire:



Connect the other wire coming from the switch to the row where the 1k resistor has been placed:



Place the LED with the longer leg in pin 13 of the Arduino and the shortest one in the GND pin. We connect pin 13 because in our sketch pin 13 has been defined as output.



Check that you have done all connections properly and then you can connect Arduino to your PC, copy and transfer the following sketch to Arduino:

```
*
* Each time the input pin goes from LOW to HIGH (e.g. because of a push-button
* press), the output pin is toggled from LOW to HIGH or HIGH to LOW. There's
* a minimum delay between toggles to debounce the circuit (i.e. to ignore noise).
* FOR FABRIC SWITCHES DEBOUNCE TIME SET AT 500 MSEC
*
* NOTE: INPUT PIN = 10, OUTPUT PIN = 13
*
* David A. Mellis
* 21 November 2006
*/

int inPin = 10;    // the number of the input pin
int outPin = 13;   // the number of the output pin

int state = HIGH; // the current state of the output pin
int reading;      // the current reading from the input pin
int previous = LOW; // the previous reading from the input pin

// the following variables are long's because the time, measured in milliseconds,
// will quickly become a bigger number than can be stored in an int.
long time = 0;    // the last time the output pin was toggled
long debounce = 500; // the debounce time, increase if the output flickers

void setup()
{
  pinMode(inPin, INPUT);
  pinMode(outPin, OUTPUT);
}

void loop()
{
  reading = digitalRead(inPin);

  // if the input just went from LOW and HIGH and we've waited long enough
  // to ignore any noise on the circuit, toggle the output pin and remember
  // the time
  if (reading == HIGH && previous == LOW && millis() - time > debounce) {
    if (state == HIGH)
      state = LOW;
    else
      state = HIGH;

    time = millis();
  }

  digitalWrite(outPin, state);

  previous = reading;
}
```

After you finished transferring, the LED on pin 13 should go on. Press the fabric switch and you will see the LED going off and on every time you press it (toggle mode). You can connect a small relay to the output and control a light.