

```
/******
```

This is a library for the MPR121 12-channel Capacitive touch sensor

Designed specifically to work with the MPR121 Breakout in the Adafruit shop

----> <https://www.adafruit.com/products/>

These sensors use I2C communicate, at least 2 pins are required

to interface Adafruit invests time and resources providing this open source code,

please support Adafruit and open-source hardware by purchasing

products from Adafruit!

Written by Limor Fried/Ladyada for Adafruit Industries.

BSD license, all text above must be included in any redistribution

```
*****/
```

```
#include <Wire.h>
```

```
#include "Adafruit_MPR121.h"
```

```
#ifndef _BV
```

```
#define _BV(bit) (1 << (bit))
```

```
#endif
```

```
// You can have up to 4 on one i2c bus but one is enough for testing!
```

```
Adafruit_MPR121 cap = Adafruit_MPR121();
```

```
// Keeps track of the last pins touched
```

```
// so we know when buttons are 'released'
```

```
uint16_t lasttouched = 0;
```

```
uint16_t currouched = 0;
```

```
void setup() {
```

```
  Serial.begin(9600);
```

```
  while (!Serial) { // needed to keep leonardo/micro from starting too fast!
```

```
    delay(10);
```

```
  }
```

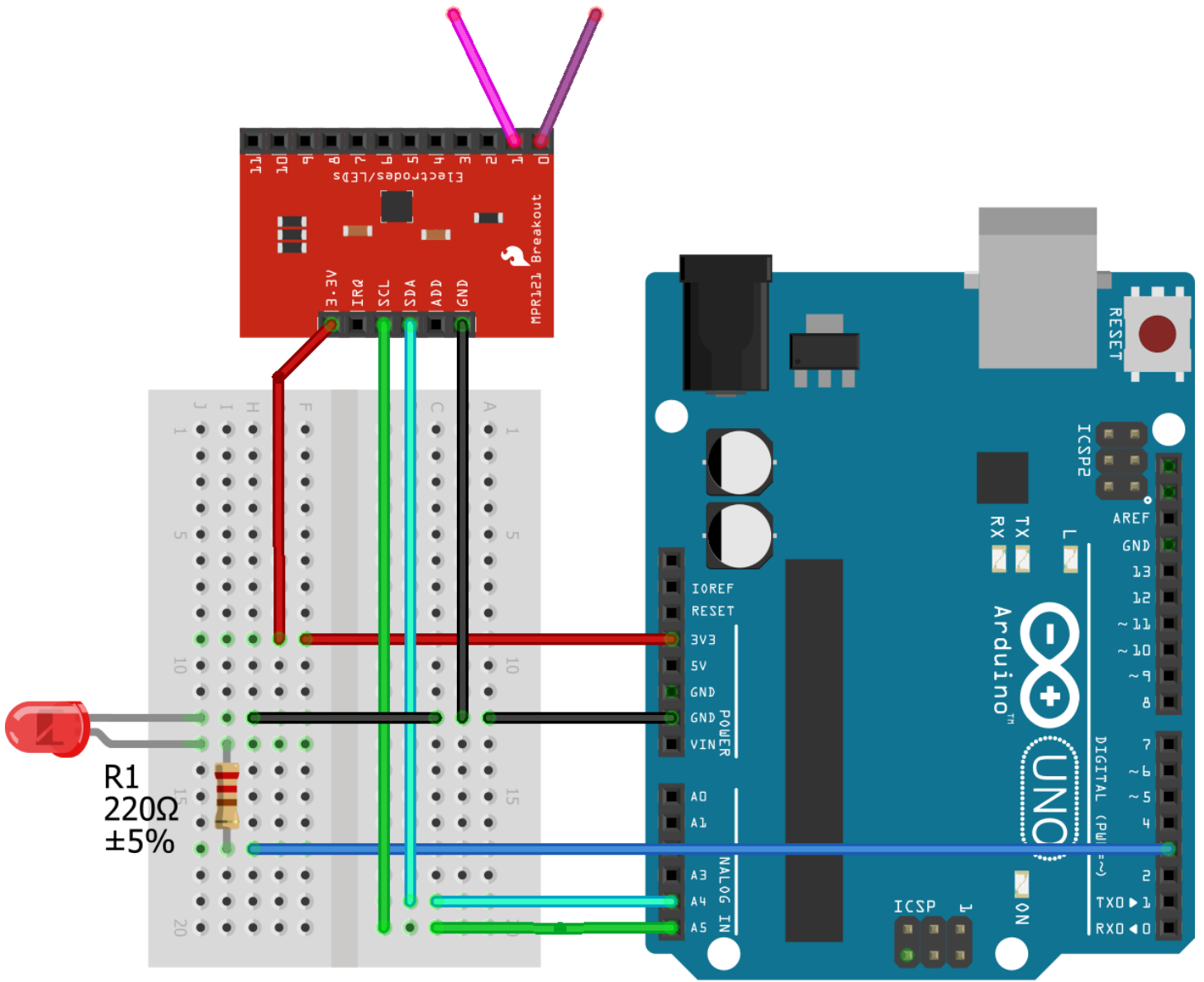
```

Serial.println("Adafruit MPR121 Capacitive Touch sensor test");

// Default address is 0x5A, if tied to 3.3V its 0x5B
// If tied to SDA its 0x5C and if SCL then 0x5D
if (!cap.begin(0x5A)) {
  Serial.println("MPR121 not found, check wiring?");
  while (1);
}
Serial.println("MPR121 found!");
}

void loop() {
  // Get the currently touched pads
  curr_touched = cap.touched();
  for (uint8_t i=0; i<12; i++) {
    // if it *is* touched and *wasnt* touched before, alert!
    if ((curr_touched & _BV(i)) && !(last_touched & _BV(i)) ) {
      Serial.print(i); Serial.println(" touched");
    }
    // if it *was* touched and now *isnt*, alert!
    if (!(curr_touched & _BV(i)) && (last_touched & _BV(i)) ) {
      Serial.print(i); Serial.println(" released");
    }
  }
  // reset our state
  last_touched = curr_touched;
  // comment out this line for detailed data from the sensor!
  return;
  // debugging info, what

```

fritzing