DIGITAL INPUTS (BUTTONS AND SWITCHES)

THE INPUTS ON THE ARDUINO READ VOLTAGE.

ALL INPUTS NEED TO BE THOUGHT OF IN TERMS OF VOLTAGE DIFFERENTIALS.

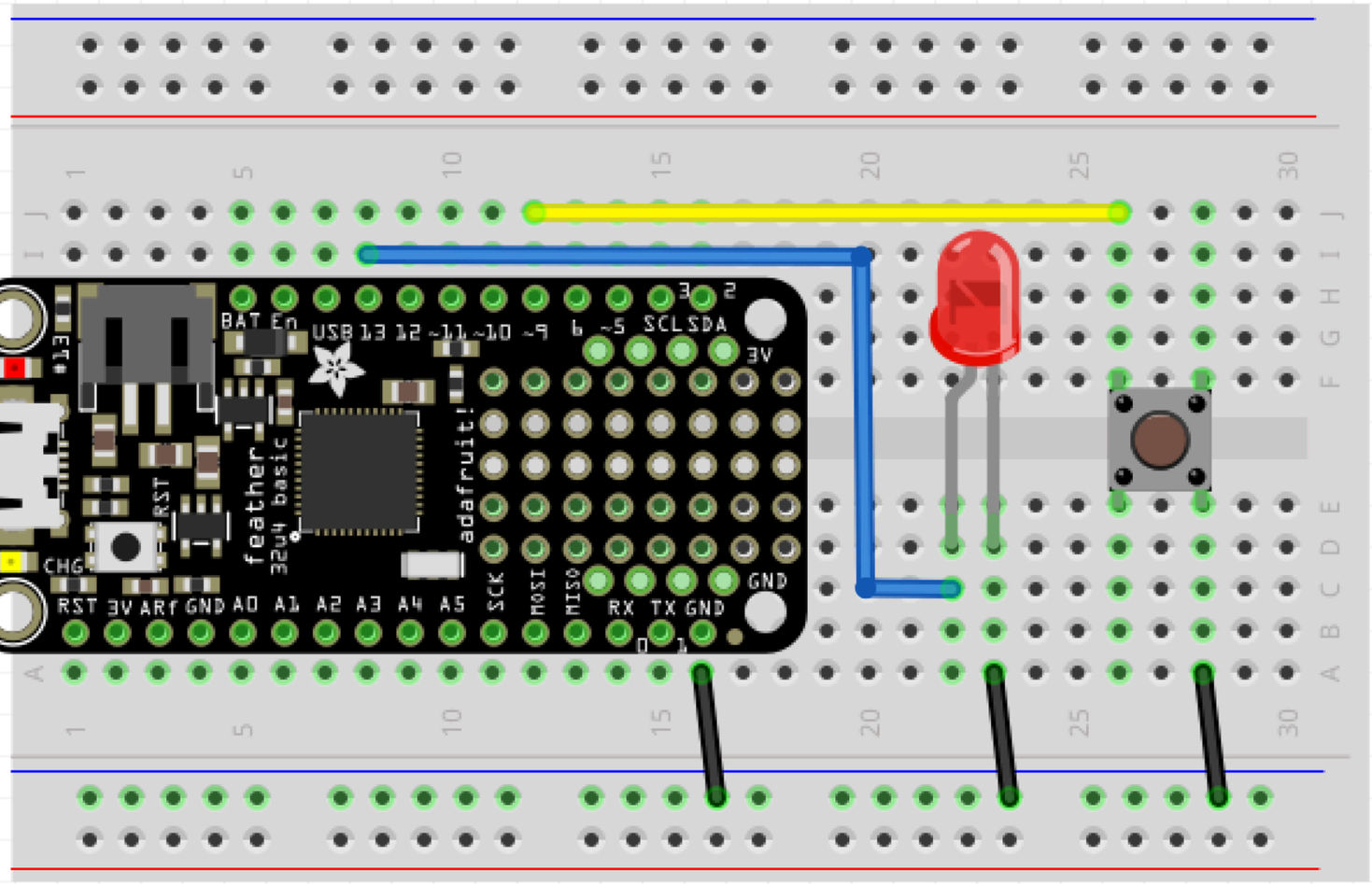
THE ANALOG INPUTS CONVERT VOLTAGE LEVELS TO A NUMERICAL VALUE.

The Arduino always needs to have a good reference for its input values. In the case of a button (digital input) it should always be either `HIGH` or `LOW`. We can do this by setting the pinMode to `INPUT_PULLUP`.

CONNECTING A BUTTON

When the button isn’t pressed, the voltage going to pin is “pulled up” to HIGH.

When the button is pressed, the connection to ground is completed so the pin is LOW.



CONTROL AN LED WITH A BUTTON

button1 | Arduino 1.8.8

button1

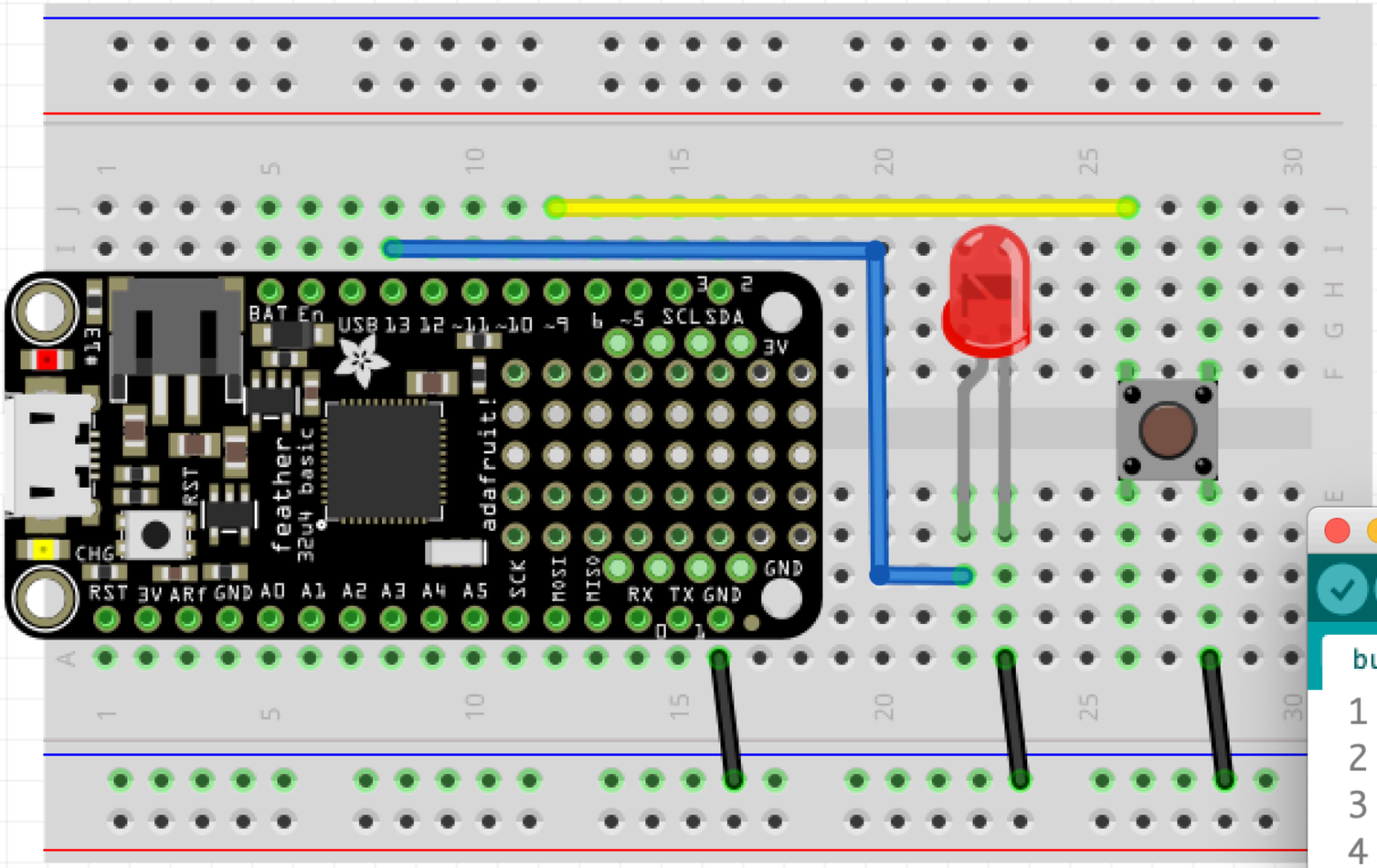
```
const int BUTTON_PIN = 9;
const int LED_PIN = 13;

void setup() {
  pinMode(LED_PIN, OUTPUT);
  pinMode(BUTTON_PIN, INPUT_PULLUP);
}


void loop() {
  int val = digitalRead(BUTTON_PIN);
  if (val == LOW) {
    digitalWrite(LED_PIN, HIGH);
  } else {
    digitalWrite(LED_PIN, LOW);
  }
}
```

Done Saving.

1 Adafruit Feather 32u4 on /dev/cu.usbmodem14101



CONNECTING A BUTTON

When the button isn't pressed, the voltage going to pin is "pulled up" to HIGH.

When the button is pressed, the connection to ground is completed so the pin is LOW.

button1 | Arduino 1.8.8

button1

```
const int BUTTON_PIN = 9;
const int LED_PIN = 13;

void setup() {
  pinMode(LED_PIN, OUTPUT);
  pinMode(BUTTON_PIN, INPUT_PULLUP);
}


void loop() {
  int val = digitalRead(BUTTON_PIN);
  if (val == LOW) {
    digitalWrite(LED_PIN, HIGH);
  } else {
    digitalWrite(LED_PIN, LOW);
  }
}
```

ANYTHING CAN BECOME A SWITCH