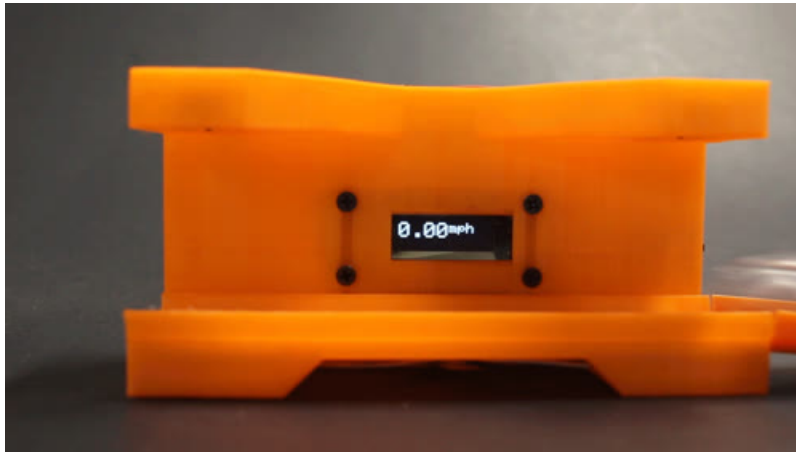




Toy Car Speed Timer

Created by Kirby Griese



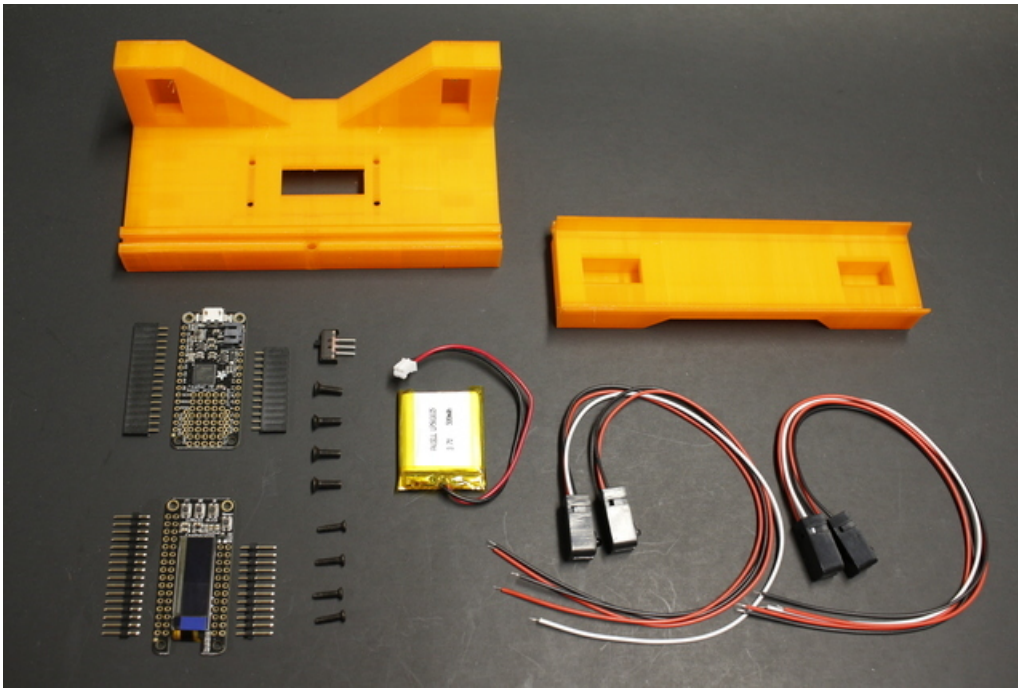
Last updated on 2018-08-22 03:58:53 PM UTC

Guide Contents

Guide Contents	2
Overview	3
Parts needed	3
Prerequisites	3
3D Printing	5
Assembly	6
Wiring	11
Software	16
Use It	19

Overview

I like collecting 1/64 scale cars and was wondering how fast they are on my track. Using two IR Break Beam Sensors I can measure the time between both sensors being triggered and determine the speed. I can then also see if a heavier car is faster and if rubber wheels are better than plastic.



Parts needed

The following parts were used to build this

- [Feather M0 Basic Proto \(http://adafru.it/2772\)](http://adafru.it/2772)
- [FeatherWing OLED \(http://adafru.it/2900\)](http://adafru.it/2900)
- [Feather Header Kit \(http://adafru.it/2886\)](http://adafru.it/2886)
- [2x IR Break Beam Sensor 3mm \(http://adafru.it/2167\)](http://adafru.it/2167)
- [Breadboard-friendly SPDT Slide Switch \(http://adafru.it/805\)](http://adafru.it/805)
- [Lithium-Ion Polymer Battery \(optional\) \(https://adafru.it/dRL\)](https://adafru.it/dRL)
- 4x 4-40 3/8" Machine Screws
- 4x 2-56 3/8" Flat Head Machine Screws

The Feather M0 can also be substituted with a [Feather 32u4 Basic proto \(http://adafru.it/2771\)](http://adafru.it/2771)

Prerequisites

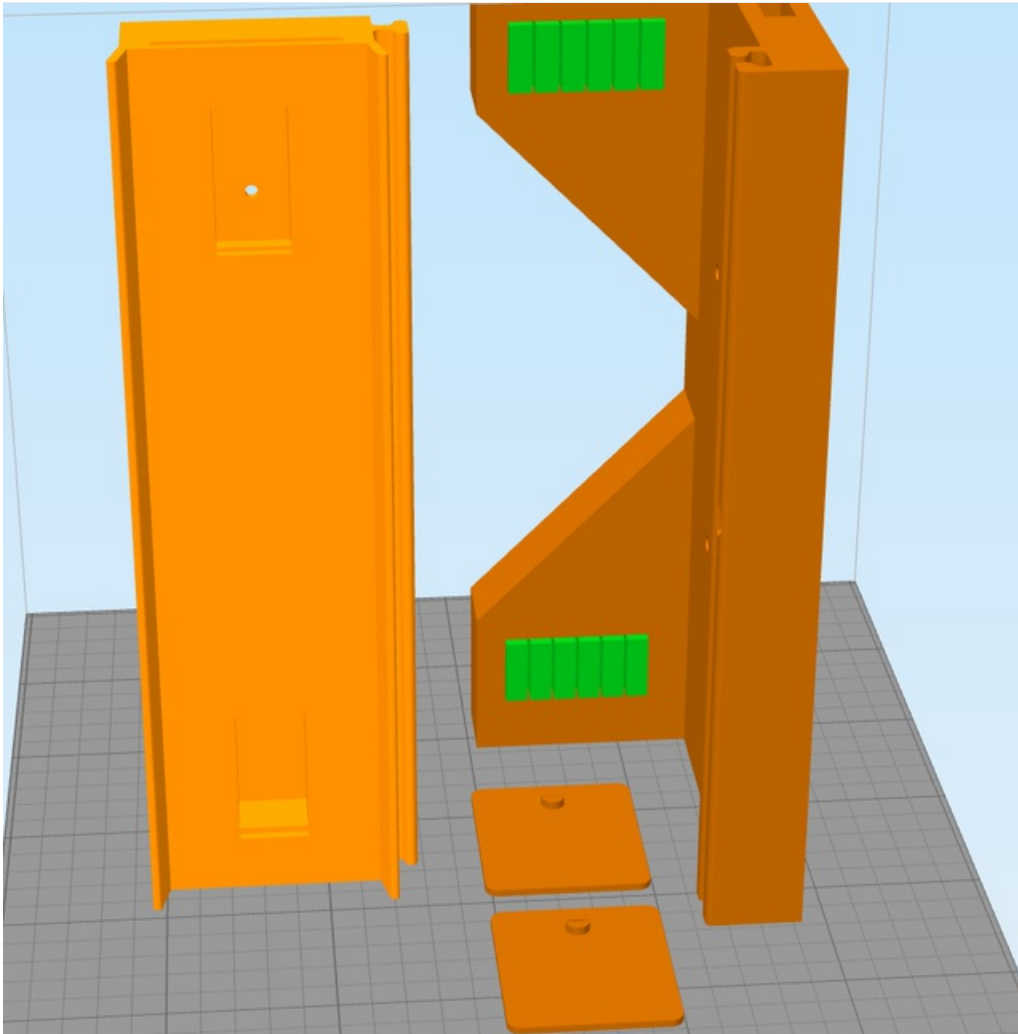
The following two guides should be followed to make sure the necessary drivers and libraries are installed in the Arduino IDE. The guides also show how to solder the headers on.

- [Adafruit Feather M0 Basic Proto \(https://adafru.it/vac\)](https://adafru.it/vac)
- [Adafruit OLED FeatherWing \(https://adafru.it/vad\)](https://adafru.it/vad)

If you havent soldered before this is a great guide to learn how to solder.

- [Adafruit Guide To Excellent Soldering \(https://adafru.it/drl\)](https://adafru.it/drl)

3D Printing



Most of the parts are designed to be printed without supports. The top part is only part that we'll need supports in the sections the sensors go in. I have the required supports shown in green.

The largest part is 68mm x 45mm and 135mm Tall

The STL files are available below along with the Fusion 360 design files if you would like to adjust any of the dimensions.

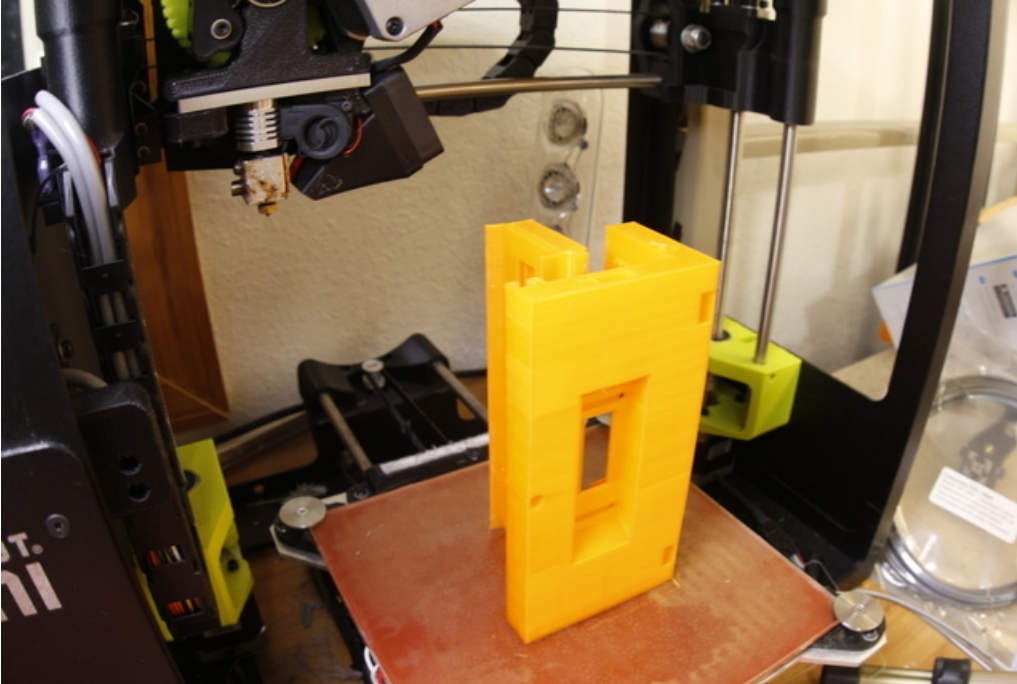
<https://adafruit.it/vae>

<https://adafruit.it/vae>

<https://adafruit.it/van>

<https://adafruit.it/van>

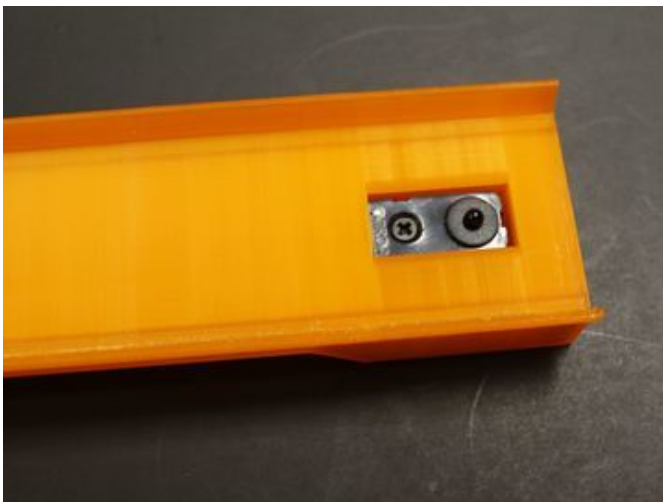
Assembly

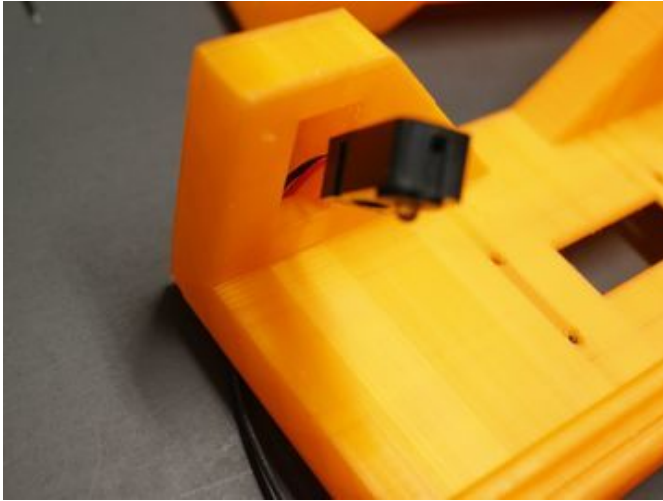




Insert the sensor with 3 wires in the base with the wires routed to the center. Then screw it down using the a 2-56 3/8" Flat Head Machine Screw.

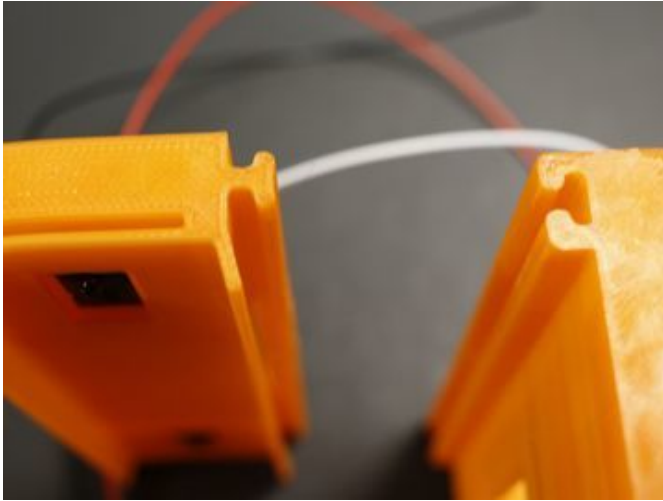
Repeat this for the left side also.





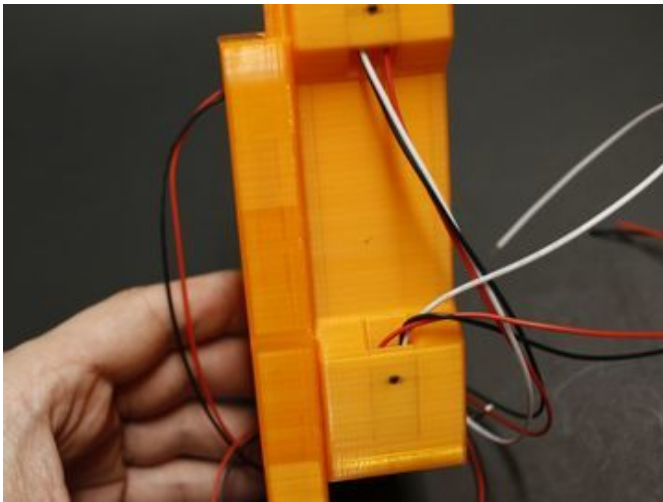
Similar to the last step we now will insert the transmitters in the top. These each have two wires. Then screw down with 2-56 3/8" Flat Head Machine Screw to secure them in.

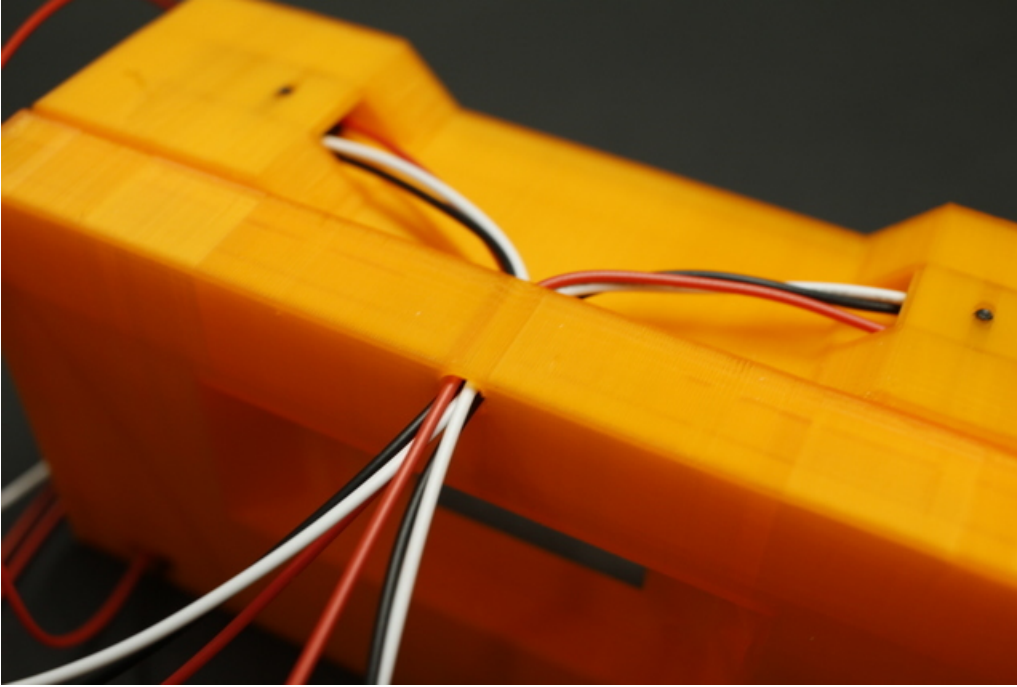




Now that the sensors are installed we will put the two pieces together. Make sure the sensors are all secured before this since after it will be hard to reach the screws to tighten them down.

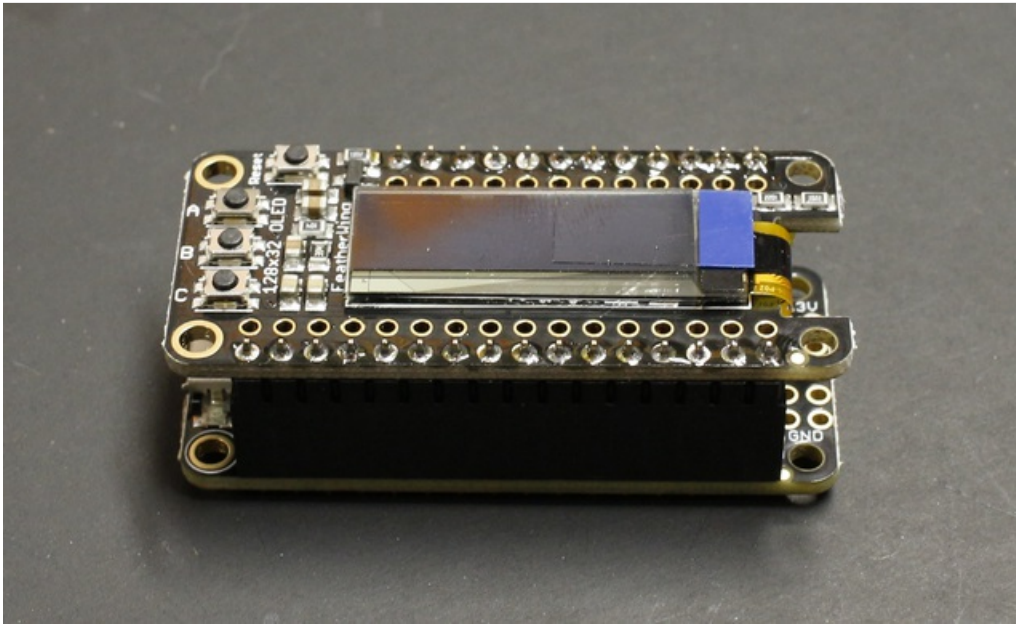
Move the wires to the side and slide the two parts together. This might require some light sanding of the parts depending on how well it printed.



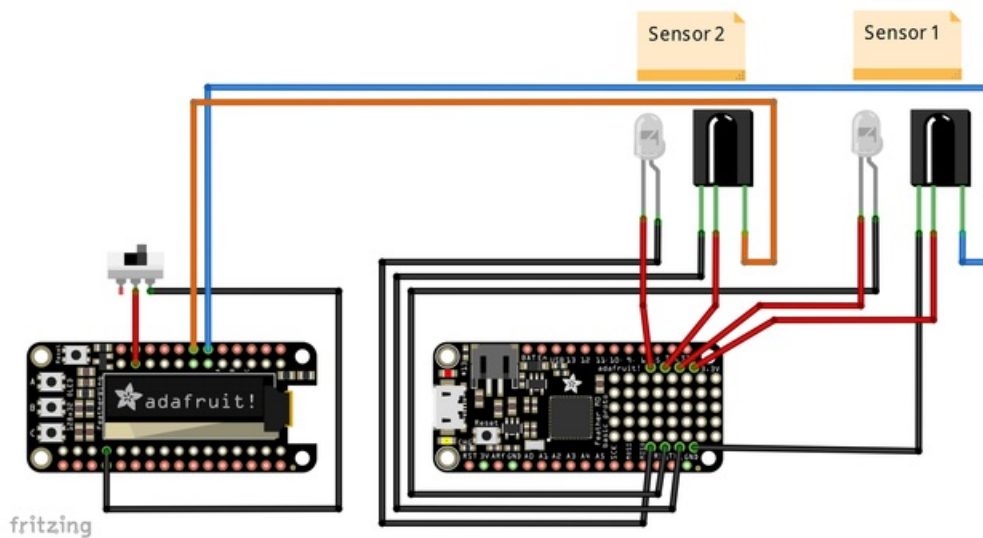


Now we can route the wires through the hole and continue to next step of wiring.

Wiring

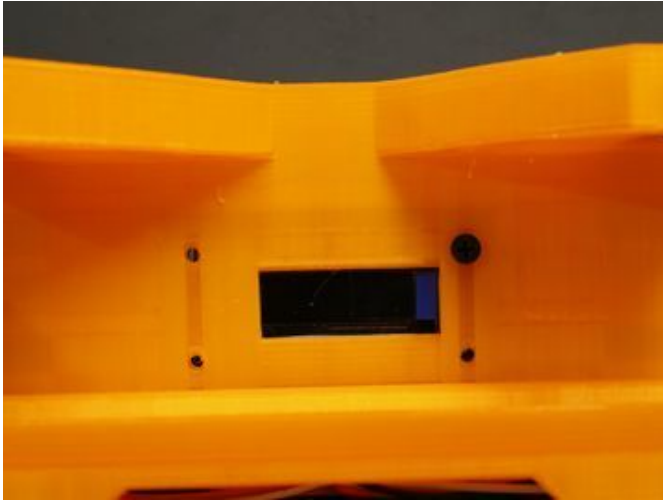


We should now have the Feather M0 and Featherwing OLED assembled and the headers soldered on each



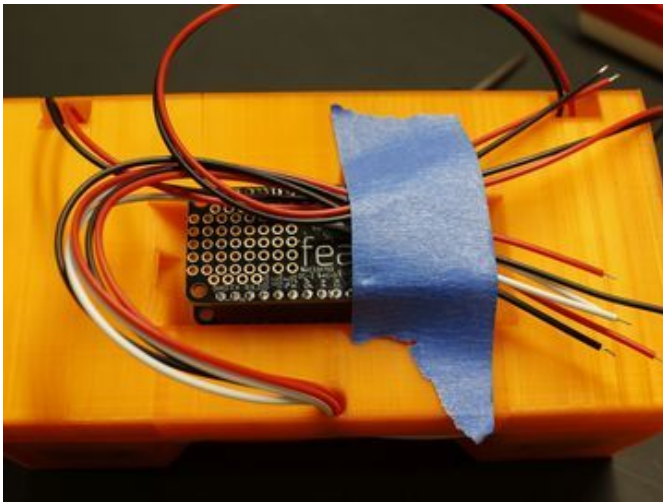
The Fritzing diagram shows where to connect the wires. For power we will be using the **3.3v** connections on the proto section. The ground will be connected the **GND** connections on the proto also.

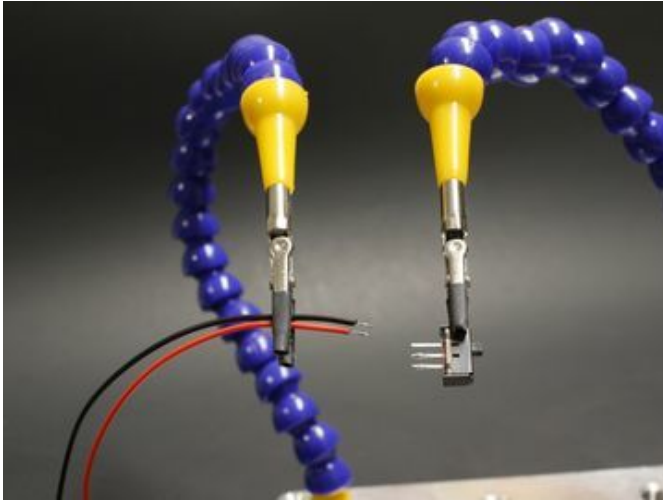
The transmitters just need power and ground. For the receivers they will also need a signal wire. For Sensor 1 we will use pin **10**, Sensor 2 will use pin **11**. Those will be wired to the space provided on the OLED FeatherWing.



To get the correct lengths for the cabling we will temporarily put one 4-40 3/8" Machine Screw in to hold the OLED Featherwing and Feather.

Then use tape to temporarily hold the wires in approximate position to cut them to a shorter length.

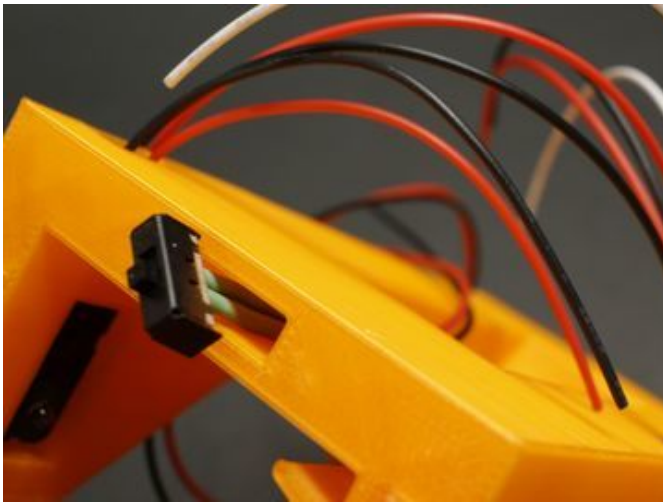


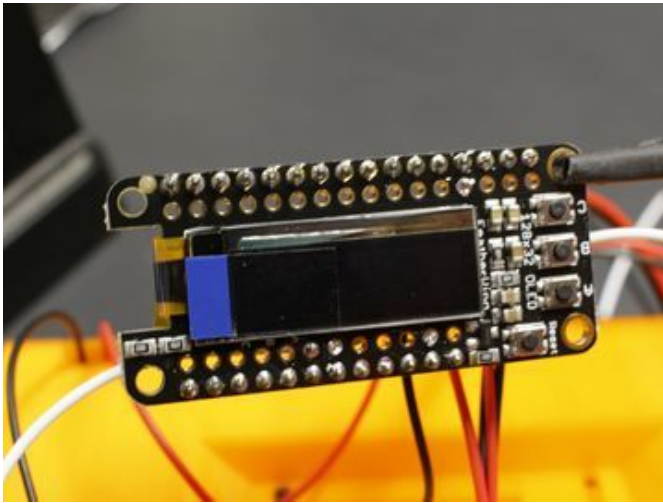


For wiring up the on/off switch we will reuse some of the wire that was cut off in the previous step. Verify that you have enough length to go from side of case with cutout for the switch to the **GND** and **EN** pins on the OLED FeatherWing.

The switch has 3 pins, but we only need the middle and one of the outer pins. The other unused outer pin can be left on or trimmed off.

Once the wires are soldered to the switch insert in to the opening on the right side.

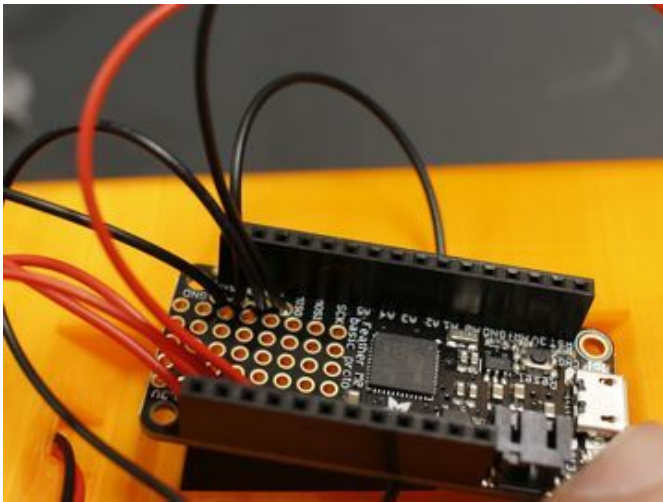




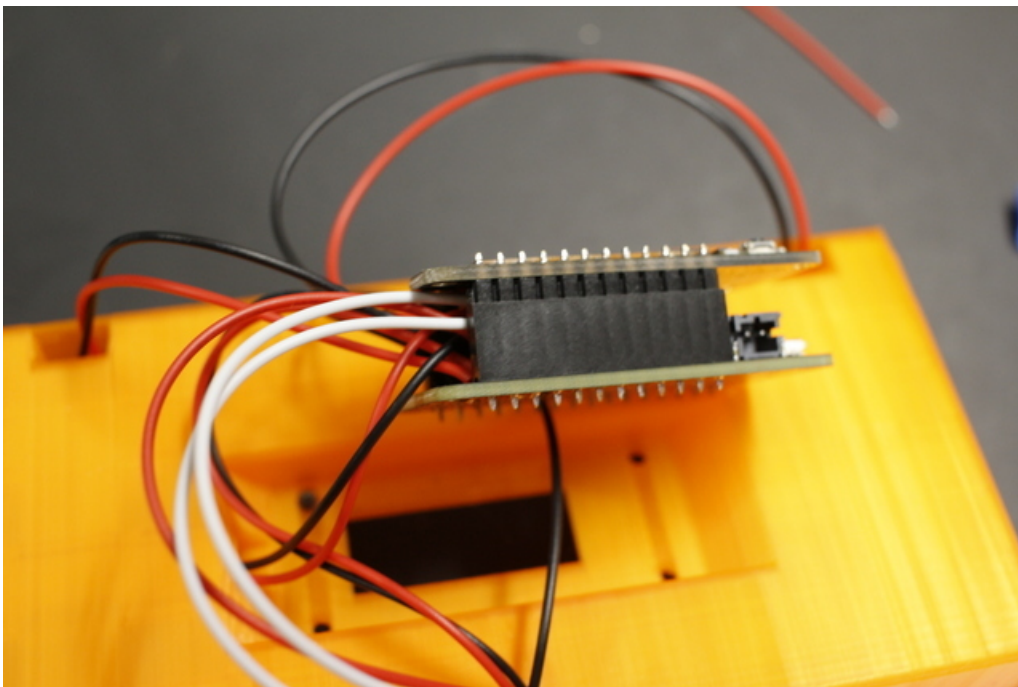
As mentioned previously we will wire the signal wires from the sensors to the OLED FeatherWing. There is a row on each side available for soldering connections.

Sensor 1 is the one on the right of the track and will connect to pin **10**. Have the wire come from under the board so it will not interfere with screwing the display to the 3D printed parts. Then wire Sensor 2 to pin **11**.

The slide switch will wire to the **EN** and **GND** on the OLED FeatherWing. Polarity for this does not matter.



The power and ground will be soldered to the rows provided on the proto section of the Feather board. We will have those wires come from the top of the board.



Now attach the OLED FeatherWing to the Feather board and it should look like the image above. Next using 4x 4-40 3/8" Machine Screws attach the boards to the assembly.

To prevent the reset button from being pressed in, do not fully tighten the screws.

Software

The program will start counting in milliseconds when the beam for Sensor 1 is broken and then stop when beam for Sensor 2 is broken. It will then calculate the speed to miles per hour.

The distance between the two sensors is 106mm. The formula for Millimeter per second to Miles per hour is $1\text{mm/s} = 0.0022\text{mph}$. Since the program returns the value in milliseconds instead of seconds it was easier to move the decimal in the distance from 106 to 106000 rather than convert milliseconds to seconds.

If you change the distance between the sensors you will need to change that value.

```
// convert mm/s to mph
mph = ((106000 / elapsed_time) * 0.0022);
```

The code is also designed that once Sensor 1 is triggered it will not trigger again until Sensor 2 is triggered. Sensor 2 can not be triggered if Sensor 1 is not triggered. So when using the timer always have the car to be timed enter from the right side.

Below is the full code to load on to the Feather

```
#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

Adafruit_SSD1306 display = Adafruit_SSD1306();

#define SENSORPIN1 10 //Sensor 1 is on the right
#define SENSORPIN2 11 //Sensor 2 is on the left

long end_time; // When Sensor 2 is triggered
long start_time; // When Sensor 1 is triggered
long elapsed_time; // End time minus start time

float mph; // Speed calculated

int trigger1 = 0; // Sensor 1
int trigger2 = 0; // Sensor 2
int sensor1State; // Sensor 1 status
int sensor2State; // Sensor 2 status

void setup() {
  pinMode(SENSORPIN1, INPUT); // Sensor 1 as input
  digitalWrite(SENSORPIN1, HIGH); // Turn on the pullup
  pinMode(SENSORPIN2, INPUT); // Sensor 2 s input
  digitalWrite(SENSORPIN2, HIGH); // Turn on the pullup

  // by default, we'll generate the high voltage from the 3.3v line internally! (neat!)
  display.begin(SSD1306_SWITCHCAPVCC, 0x3C); // initialize with the I2C addr 0x3C (for the 128x32)

  // Clear the buffer.
  display.clearDisplay();
  display.display();

  display.setTextColor(WHITE);
```



```

}

// Function to determine speed
void speed()
{
  // subtract end time from start time to get total time
  elapsed_time = ((end_time - start_time));

  // convert mm/s to mph
  mph = ((106000 / elapsed_time) * 0.0022);

  // set OLED display text size
  display.setTextSize(3);

  // Clear the buffer.
  display.clearDisplay();
  display.display();

  display.setCursor(0, 0);
  display.print(mph);
  display.setTextSize(2);
  display.print("mph");
  display.setCursor(0, 0);
  display.display(); // actually display all of the above
}

void loop() {
  // Read the state of the IR sensor 1:
  sensor1State = digitalRead(SENSORPIN1);

  // See if IR beam of sensor 1 has been broken
  if (sensor1State == LOW) {

    // Check to make sure both sensors have not triggered
    if (trigger1 == 0 && trigger2 == 0) {

      // Save time when sensor 1 was triggered
      start_time = millis();

      // Prevent sensor 1 from triggering again
      trigger1 = 1;
    }
  }

  // Read the state of the IR sensor 2:
  sensor2State = digitalRead(SENSORPIN2);

  // See if IR beam of sensor 2 has been broken
  if (sensor2State == LOW) {

    // Check to make sure sensor 1 has triggered but not sensor2
    if (trigger2 == 0 && trigger1 == 1) {

      // Save time when sensor 2 was triggered
      end_time = millis();

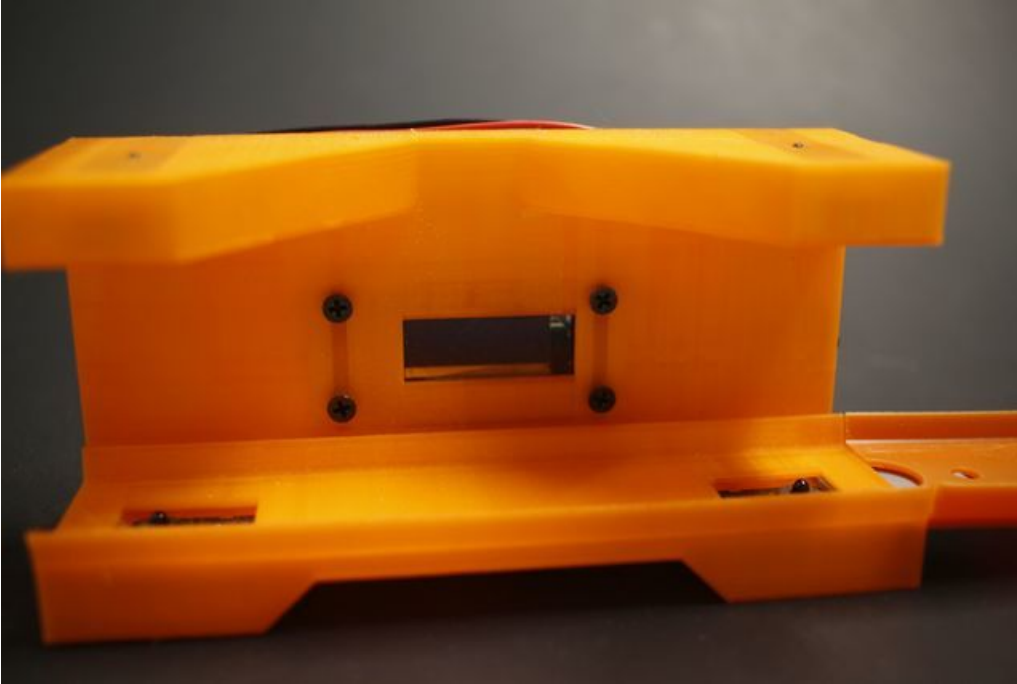
      // Run speed function
      speed();
    }
  }
}

```

```
    // Prevent sensor 2 from triggering again
    trigger2 = 1;
  }
  delay(1000);

  // Reset both sensors
  trigger1 = 0;
  trigger2 = 0;
}
}
```

Use It



You can use the timer by powering the Feather through USB or using a Lithium Ion battery.



Once the car goes past both sensors the speed will be displayed. I did not have another track piece to connect to the left side. But both sides are compatible with Hot Wheels tracks.



If nothing is displaying verify that the display is not too tight to the enclosure causing the reset button to be held in.



I designed the timer to fit taller vehicles but some are too wide for the track and will not fit. This vehicle was an example, it was just a bit too wide.