

Alex Shah

Project 9 - Orientation Sensor Data Over SPI

2024-05-03

EN.605.715.81.SP24

## Table of Contents

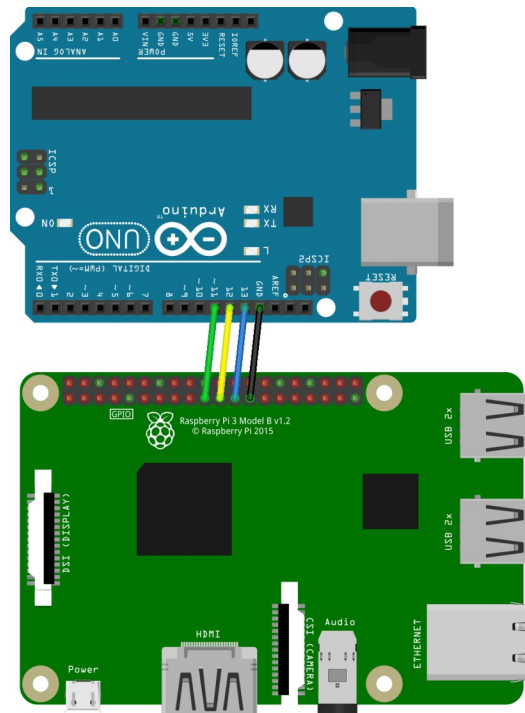
Requirements.....	.1
Design.....	.1
Implementation.....	.3
Demo.....	.5
References.....	.6

NOTE: This project uses SPI because I was unable to get the deprecated Bluetooth libraries working.

## Requirements

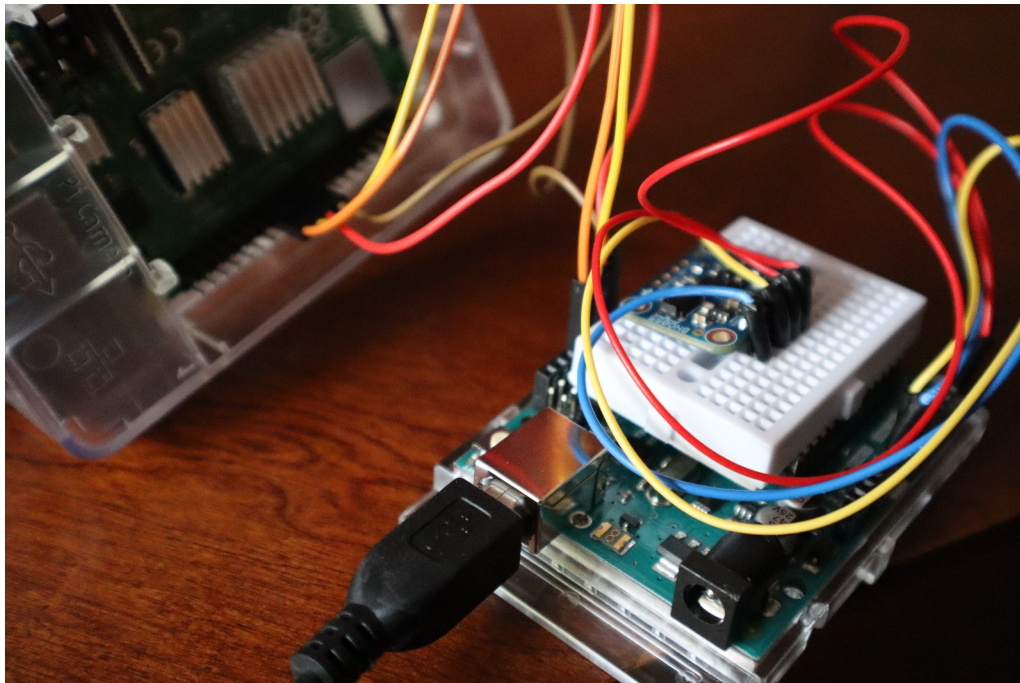
Use an Adafruit BNO055 Sensor with an Arduino, connect the Arduino to a Raspberry Pi using SPI, and transmit the IMU data (roll, pitch, and yaw) from the Arduino to the Pi over SPI.

## Design

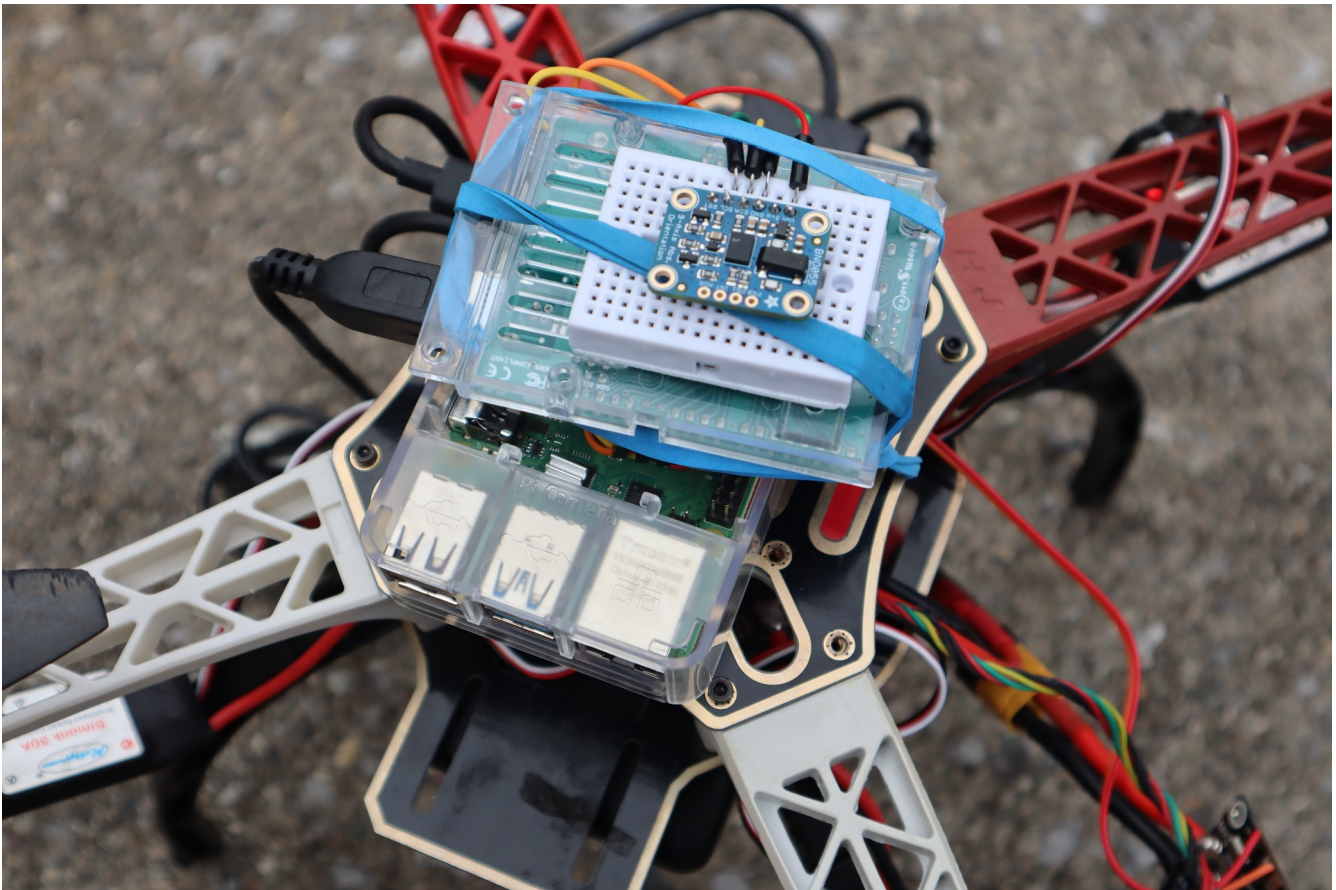


The Arduino is connected to the Pi with 4 wires, GND, Master In Slave Out, Master Out Slave In, and CLK. This corresponds to pins GND, 13, 12, 11 on the Arduino and GPIO pins 19 through 25 on the Pi.

The Adafruit BNO055 is connected to the Arduino with 4 wires, GND to GND, Vin to 5v, SDA to A4, and SCL to A5.



The Pi, arduino, and sensor board are stacked on to the drone for flight.



## Implementation

The Arduino runs code to send IMU data over SPI to the Pi:

...

```
1 #include <Adafruit_BNO055.h>
2 #include <SPI.h>
3
4 Adafruit_BNO055 bno = Adafruit_BNO055(55, 0x28);
5
6 const int slaveSelectPin = 10;
7
8 struct IMUData {
9   float roll;
```

```
10  float pitch;
11  float yaw;
12  };
13
14  IMUData imuData;
15
16  void setup() {
17    bno.begin();
18    SPI.begin();
19    pinMode(slaveSelectPin, OUTPUT);
20    digitalWrite(slaveSelectPin, HIGH);
21  }
22
23  void loop() {
24    sensors_event_t event;
25    bno.getEvent(&event);
26    imuData.roll = event.orientation.x;
27    imuData.pitch = event.orientation.y;
28    imuData.yaw = event.orientation.z;
29
30    sendDataOverSPI();
31    delay(10);
32  }
33
34  void sendDataOverSPI() {
35    SPI.beginTransaction(SPISettings(1000000, MSBFIRST, SPI_MODE0));
36    digitalWrite(slaveSelectPin, LOW);
37
38    SPI.transfer((uint8_t*)&imuData, sizeof(IMUData));
39
```

```
40  digitalWrite(slaveSelectPin, HIGH);
41  SPI.endTransaction();
42  }
...

```

And the Pi runs a python script to receive data over SPI and display it to the console. I am remotely connected to the Pi by Wifi, enabling the data to be sent in flight.

```
...

1  import spidev
2  import struct
3
4  spi = spidev.SpiDev()
5  spi.open(0, 0)
6
7  while True:
8      imu_data = spi.readbytes(12)
9
10     imu_data_bytes = bytes(imu_data) # Convert list to bytes object
11
12     roll, pitch, yaw = struct.unpack('fff', imu_data_bytes)
13
14     print("Roll:", roll)
15     print("Pitch:", pitch)
16     print("Yaw:", yaw)
...

```

## Demo

The drone flies, and the IMU data is sent over SPI to the Pi. I ssh into the Pi and view the console while the Python script is running to view the data wirelessly in flight.

[https://drive.google.com/file/d/1SxN1YSqmU6bnECKhMUgA\\_AkAyXtf2vS0/view?usp=sharing](https://drive.google.com/file/d/1SxN1YSqmU6bnECKhMUgA_AkAyXtf2vS0/view?usp=sharing)

## References

<https://roboticsbackend.com/raspberry-pi-master-arduino-uno-slave-spi-communication-with-wiringpi/>