

ILR #3

Adam Yabroudi- Team A

October 30, 2015

Teammates: Tushar Agrawal, Pratik Chatrath, Sean Bryan

Individual Progress

This week I worked on helping set up the motor for the obstacle avoidance tests, I worked with BirdsEyeView Aerobotics to handle all the firmware code, I ordered all the parts for the vehicle moving forward, and helped rethink our whole project because of new information and potential risks.

Setting up the motor and ESC

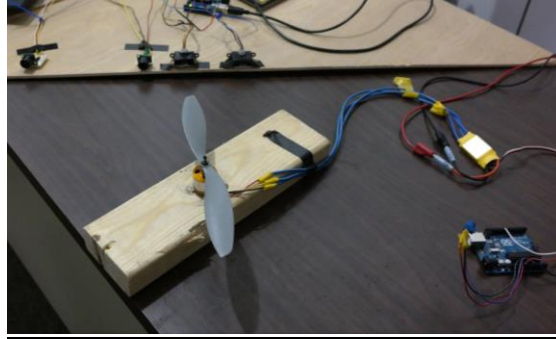


Figure 1: Picture of motor setup used to test sensors in obstacle avoidance system

My first contribution this week was getting the motor set up and running for Pratik and Sean to run their tests. A picture of this setup can be seen in Figure 1. The two main components to this were assembling all the pieces from around the lab and from classmates followed by coding up the ESC setup code. We had to borrow some pieces from Sam so that the propeller fit properly on the brushless motor I had at home and then mount it to a piece of wood to make sure it was safe and easy to manipulate. The code was written in Arduino and used a potentiometer to control the speed of the motor which allowed us to see how noise changed when we pulsed the motors and at various speeds. The two iterations of code can be seen at the end of this report.

Working with BirdsEyeView

This was a large commitment this week because a lot of our previous assumptions changed once we handled the NDA and were able to talk to them. They are allowing us to test their proprietary code that they haven't released yet and so I was able to get that up and running on the Pixhawk in a simple out of the box test. A picture of that can be seen in Figure 2. We also were expecting them to send us a fully equipped vehicle as the CEO had mentioned in earlier emails. After finally talking to him, we realized we weren't getting the parts we needed and so had to find a compromise that worked with both parties and still stay within budget. We finally got everything ordered (hopefully) and should be able to focus on actual engineering moving forward this week.

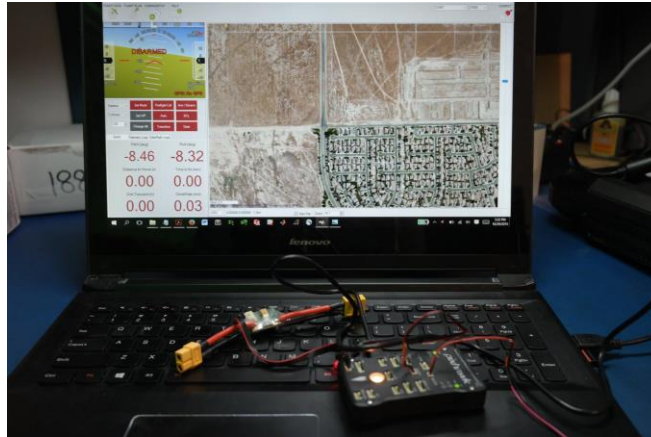


Figure 2: Pixhawk connected to the modified version of Mission Controller from BEVA

Ordering visual processing boards

In the middle of all my procurement, Tushar and I tried to purchase an extra visual processing board for him in case our BeagleBoard-xM failed. There were no more BeagleBoard-xM being sold so we changed to BeagleBone temporarily. After settling on that for less than 24 hours, we changed on to the Odroid. Tushar and I were trying to compare all those boards and reevaluate all the design choices we had assumed for the BeagleBoard-xM.

Designing the PDS

This unfortunately (or fortunately depending on how you view it) caught us unaware and was something we hadn't planned for in our schedule. It is something we are now more conscious of and will probably push us to make decisions on the obstacle avoidance system faster than we expected. I worked to design the system on a high level this week but it has lots of details that need to be resolved before the board is created.

Challenges:

Procurement:

This was mentioned above but this week's problems were really with working with BirdsEyeView Aerobotics and handling the communication between us and getting this project figured out. We hit a lot of realities this week that we had to resolve quickly to get back on pace. Hopefully going forward less of our problems will be about organizational work and more will be technological issues.

Teamwork:

Pratik and Sean handled obstacle detection. They were working on gathering more data about the limits of the sensors in various environments and conditions. I worked with them to set up the motor setup so that they could get the results they needed.

Tushar and I worked together on picking a vision processing board and I also worked with him a tiny bit on visual markers and planning what varieties of markers we wanted to test long range. This was more high level strategy.

Future Plans

Going forward, Tushar is going to continue working on designing a vision system that can detect markers robustly at further distances in different environments. Tushar also wants to try using a better camera with a higher FPS for the algorithms and see how it affects quality of the detection.

Sean is going to start working on assembling the drone if all the parts come in on time. He will also start designing the underbelly of the vehicle. He will also assist Pratik as needed to further test sensors.

Pratik is going to continue working on obstacle avoidance. He will try using different filters in code, new sensors (Lidar Lite), and digital data from the sensors instead of analog all in the attempts to reduce noise and get better data. Ultimately we need to find an “acceptable” threshold to noise. He will also test combinations of sensors including the Hokuyo.

I will continue working on the firmware. I will also be putting the electrical system together for the drone as the parts come in. I will also work on the PDS circuit design and try to have a more detailed design for our project and for future PDS assignments.

Code

Code to run ESC straight with no outside input:

```
#include <Servo.h>
```

```
Servo esc;
```

```
//int throttlePin = 0;
```

```
int throttle;
```

```
void setup()
```

```
{
```

```
    esc.attach(9);
```

```
    throttle=90;
```

```
    esc.write(throttle);
```

```
    delay(3000);
```

```
    //for 1 seconds
```

```
    //throttle=90;
```

```
    //esc.write(throttle);
```

```
    //delay(5000);
```

```
    //for 1 seconds;
```

```
    for (throttle=90; throttle>2; throttle--)
```

```
    {
```

```
        esc.write(throttle);
```

```
        delay(10);
```

```
    }
```

```
    throttle=0;
```

```
    esc.write(throttle);
```

```

        delay(3000);
        for (throttle=0; throttle<70; throttle++)
        {
            esc.write(throttle);
            delay(10);
        }
    }

void loop()
{
    //int throttle = analogRead(throttlePin);
    //throttle = map(throttle, 0, 686, 0, 179);
    throttle=70;
    esc.write(throttle);
}

```

Code to run ESC off of a potentiometer:

```
#include <Servo.h>
```

```
Servo esc;
int throttlePin = 0;
```

```
void setup()
{
    esc.attach(9);
}

```

```
void loop()
{
    int throttle = analogRead(throttlePin);
    throttle = map(throttle, 0, 686, 0, 179);
    esc.write(throttle);
}

```