Boston University
Electrical & Computer Engineering
EC 463 Capstone Senior Design Project

First Semester Report

UAV Laser Tracking

Submitted to

Professor Thomas Little, PhD
8 St. Mary's St., Boston
MA 02215, Room 426
(617) 353-9877
tdcl@bu.edu

by Team 15
LaserTrac

Team Members

Aviva Englander	englanda@bu.edu
Christopher Liao	cliao25@bu.edu
Jeffrey Lin	jlin96@bu.edu
Anton Paquin	paquin@bu.edu
Eduardo Portet	eportet@bu.edu

Submitted: 12/11/2017

Table of Contents

	Executive Summary	2
1.	Introduction	3
2.	Concept Development	4
	2.1. Laser Steering Mechanism	5
	2.2. Receiver	6
	2.3. Tracking Algorithm	7
	2.4. Backlink	7
3.	System Description	8
	3.1. Backwards Link	8
	3.2. Primary Station Controller	8
	3.3. Laser Controller	9
	3.4. Data Controller and Laser	9
	3.5. Photodiode Array	9
	3.6. Motion Controls	10
	3.7. Camera and Coarse Control	10
4.	First Semester Progress	11
5.	Technical Plan	12
	5.1. Establish Backwards Link	12
	5.2. Build Servo Control	12
	5.3. Build MEMS Control	12
	5.4. Tracking Algorithm	13
	5.5. Second Deliverable Testing	14
	5.6. PCB Design	14
	5.7. Tracking Algorithm Testing	14
	5.8. Minimum Deliverable by Functional Testing	15
6.	Budget Estimate	15
7.	Attachments	16
	7.1. Engineering Requirements	16
	7.2. Gantt Chart	16
	7.3. Photodiode Circuit	17
	7.4. Stepper Motor Circuit	18
	7.5. Tracking Algorithm Block Diagram	18
	7.6. Receiver Block Diagram	19

Executive Summary

Laser tracking and Communication with UAVs
Team 15 -LaserTrac

As the radio spectrum becomes more crowded, interference has become an increasingly urgent problem. Wireless Optical Communication (WOC) has emerged as an attractive alternative to radio communication, since it can be aimed at a specific point. Currently WOC is used between stationary objects in many cities, where the transmitter is often mounted on top of skyscrapers to ensure clear line-of-sight. However, WOC with moving targets is still being researched.

The goal of our project is to maintain two-way WOC with an Unmanned Aerial Vehicle (UAV). In order to accomplish this goal, the problem of tracking a moving object must be solved. We intend to use the laser used for communication to track the UAV. While both WOC and laser tracking are well researched, there is little prior work focused on accomplishing both communication and tracking with one laser device. Potential applications for our project include providing internet access to disaster areas, drone racing, and air traffic control.

1. Introduction

Currently the radio frequency spectrum is cluttered and requires a license for use. We want to develop a method of rapid data transmission which doesn't require a license. What differentiates optical communications from radio frequency communications is its "high data rate capabilities and ultra dense access point deployment" (Rahaim and Little 2017). Free space optical communications can potentially transmit data at rates of up to Gbps (Hyperion Project). Which is why optical wireless communication will be a viable alternative to radio frequency communications.

The paramount issue with using wireless optical communications with moving aerial vehicles is there is no good way to keep the visible light focused on the vehicle. Without our project this communication link cannot be maintained for moving aerial vehicles resulting in gaps in communication and lost data.

Our laser tracking system has many applications. It can be used to provide Wi-Fi in remote areas and to communicate with planes in crowded airfields where the RF spectrum is extremely crowded. In addition it will be useful in drone racing scenarios for providing remote control.

Our project's goal is to maintain a constant communication link with aerial vehicles. Our laser tracking system can be used for any kind of rapid information transmission from a ground station to fast-moving aerial vehicles.

2. Concept Development

The problem presented by our client was very general: Professor Little wants technology that can track a fast moving object in three dimensions with a laser beam. While there are numerous useful application for this technology, we and the client have agreed to narrow the scope of the project to a drone racing scenario to reduce ambiguity and facilitate development of engineering objectives. Figure 2.1 illustrates the initial visualization provided by the client. As a drone moves to the right, the base station follows the quadcopter to the right with a laser. The quadcopter also has a laser backlink aimed at the base station. This backlink would presumably have to be a different wavelength (a green laser instead of a red laser for example). After discussion with the client, we made the optical backlink optional due to time constraints.

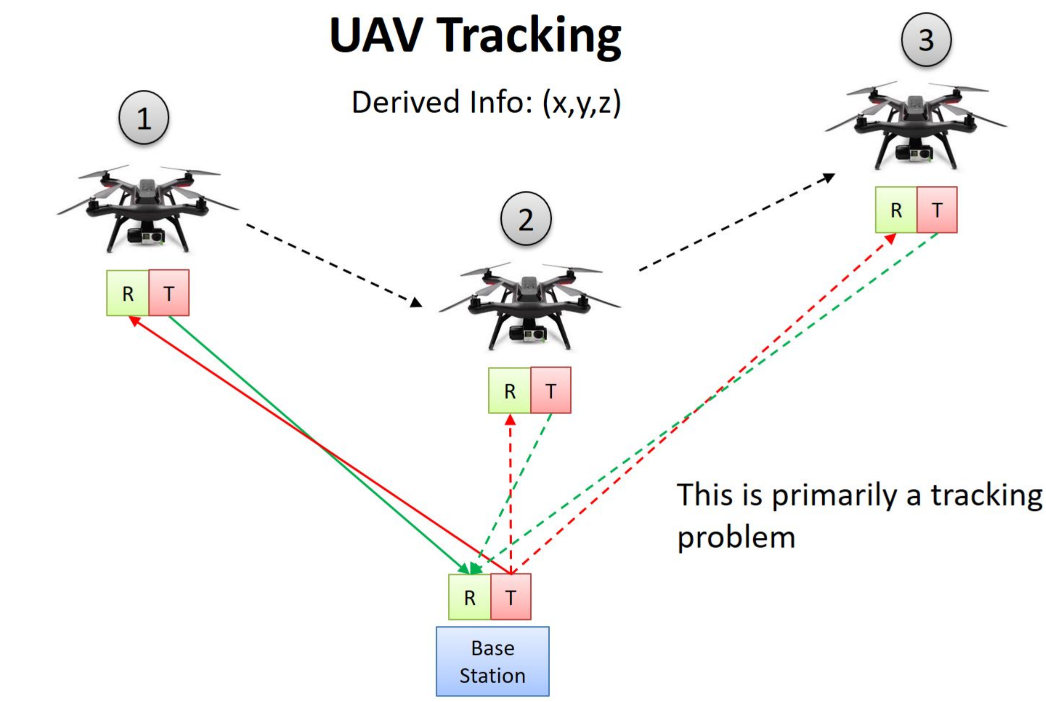


Figure 2.1. Visualization of Laser Tracking

From the initial discussion with the client, we came up with the following concept. The laser steering mechanism would be implemented using a MEMS mirror. The advantage of a MEMS mirror is threefold: 1) it can rotate at up to 4 kHz and thus can quickly acquire signal from the drone's receiver, 2) it is precise, which makes tracking of far-away objects viable, 3) unlike any servo system made of motors, the mirror has negligible inertia, allowing high acceleration. The MEMS mirror setup was already provided by the client. The setup consists of a laser beam routed through optical fiber and aimed at the MEMS mirror. Currently the laser's line of sight is constrained by the bracket holding it. For the final design, we will have to modify the mount of the laser and the MEMS mirror so that the laser can be aimed anywhere above the horizontal plane. The client emphasized that the drone itself is not a deliverable for the project.

We are only responsible for delivering the receiver, but a drone will be provided for testing. The **receiver assembly** and **tracking method** are the two most important design decisions we have to make. The initial concept for the receiver was an array of photodiodes on a circuit board. The tracking algorithm will be a form of feedback control. We agreed that the final control mechanism must be closed loop, but the client has IR positioning equipment available for testing. The initial tracking algorithm built off of the following simple scenario. Assume that the receiver consists of two photodiodes and is constrained to move along a line. When the laser hits the left photodiode, it laser is moved right and vice versa. Using this rudimentary tracking algorithm, we would theoretically be able to track the receiver regardless of its motion assuming that the motor can accelerate at any rate and move at any speed and that there is no latency in the feedback signal. This rudimentary tracking method was the starting point for development of a tracking algorithm.

Given the complexity of this project, we chose a divide-and-conquer approach to concept development. We focused on designing one component at a time. The following paragraphs go into more detail about development of each component.

2.1. Laser steering mechanism

We went back and forth between using the MEMS mirror provided and a servo mechanism for steering. The advantage of the MEMS mechanism has already been discussed above. The advantages of the servo mechanism is that it is cheaper and more rugged. Motors are also more intuitive to control and may be easier to debug. Adding the MEMS mirror adds another layer of complexity. Right now, we are working on getting the MEMS mirror working and building a stepper motor mechanism in parallel. If we can get the MEMS mirror working, work on building the servo mechanism isn't necessarily in vain, since we have plans to mount a RasPi camera on the servo for coarse tracking (more on this in following sections). The mirror steers the beam by tilting the mirror in different directions. The tilt of the motor is controlled by a voltage in the x direction and a voltage in the y direction. However, the tilt of the angle does not correspond directly with the position of the drone. We will need to write a function that takes the polar coordinates of the drone as input and outputs the tilt of the mirror in the x and y directions, taking into account the angle at which the laser beam is incident. The stepper motor assembly is slightly easier to control, because we can move the laser into position based on a set of polar coordinates.

2.2. Receiver

We agreed in the beginning that the receiver would consist of an array of photodiodes, the design of the photodiode array has changed over the semester. Theoretically, if there were no limits on the laser steering mechanism, and we are able to obtain an analog value from the photodiodes, we could build a receiver with four photodiodes and design a tracking mechanism using PID control. Such a receiver would have a photodiode in the middle to demodulate the laser (for communication). The other four diodes would be positioned in each of the four coordinate directions. In this setup, we would assume that the photodiode is able to output an analog value based on the distance the laser is from the photodiode. For example, if the laser is left-of-center, as the beam moves away from the center photodiode and closer to the left photodiode, the intensity of light the left photodiode sees would increase, and thus its analog value would increase. The analog value outputted by each of the four photodiodes could be used as the error function in the PID control loop. There are two main problems with this solution:

- The photodiode connected in reverse bias configuration can output a analog value, but since it quickly deflects to extremes, it is as good as a digital signal.
- Transmitting the analog signal takes longer, and latency is a critical factor in feedback control loops.

The concept that we settled on is a 4x4 array of photodiodes. Each photodiode would output a digital value. The feedback would consist of 4 bits indicating the photodiode that is on. For this to work, the photodiodes must be close enough to each other that the laser beam can’t slip between adjacent diodes. More specifically, the beamwidth of the laser must never be less than the distance between adjacent photodiodes. The disadvantage of this design is that we can only resolve the position of the laser relative to the center of the receiver down to the distance between the photodiodes. This gives us a very imprecise error function and thus puts a lot of burden on the tracking algorithm.

2.3. Tracking Algorithm

For the tracking algorithm, we decided to use video processing from a RasPi camera to get initial coarse positioning. To make our video processing algorithm robust in situations with multiple moving overhead objects, we decided to add a pattern of leds to the bottom of the drone. Having an easily identifiable pattern will make identifying our drone easier.

For the fine positioning we plan on using a feedback loop with predictive tracking. The ground station will get its initial pointing positioning from the video processing unit and will point in that area until it gets feedback from the drone telling it which way to move. The drone will have a photodiode array, which it will read with an Arduino Leonardo in order to determine which direction the laser will need to move to hit the center photodiode. This information will be transmitted back to the ground station using radio frequency transmitters. Then the laser will move in the direction directed by the signal from the drone. When the laser hits the center photodiode, the azimuth and elevation angles will be used as measurements to update the Kalman filter and to predict where the laser should point next.

2.4. Backlink

As mentioned already, communication back from the drone using laser is optional. Judging from the progress so far, it seems unlikely that this will be a wireless optical link. We will use radio communication instead. The bitrate of this link is critical, since it is the feedback for tracking. We have previously considered using Bluetooth for this, since we have experience with Bluetooth. We decided that Bluetooth is too slow for this application. We have purchased NRF24L01 transceivers. These transceivers are cheap and compatible with Arduinos. The transmission bandwidth and the range are also suitable for our application.

3. System Description

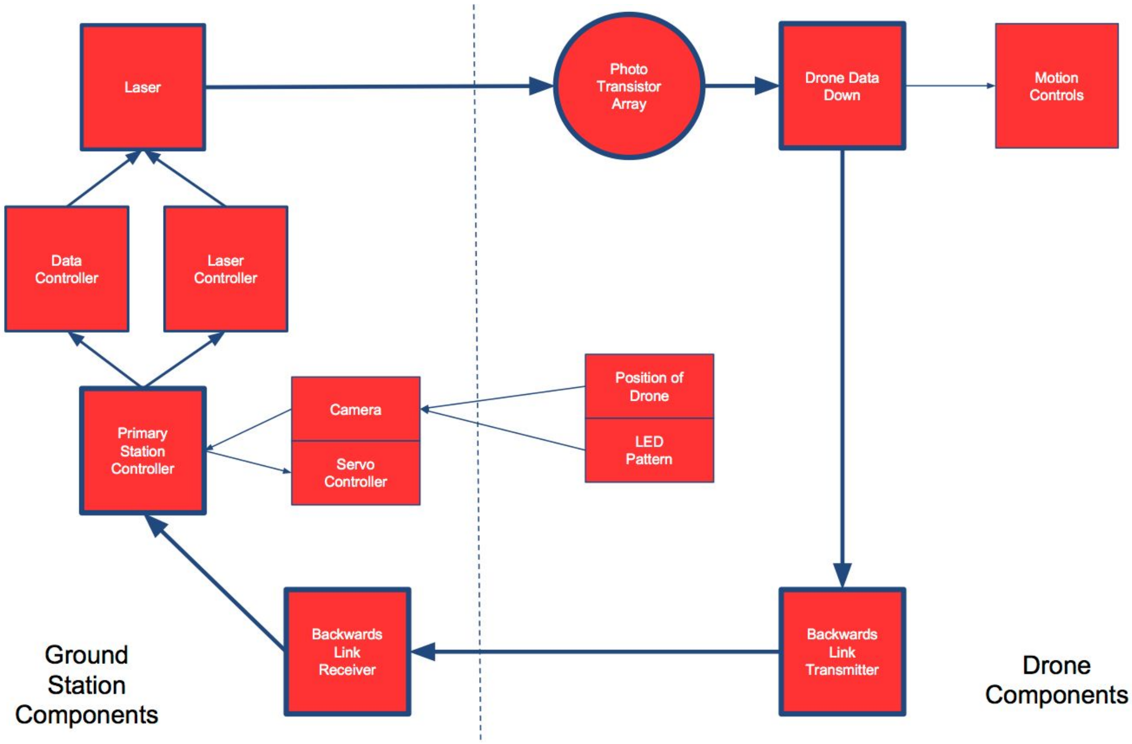


Figure 3.1. Block Diagram for System

The system diagram above shows the major components in our design. The bolded path represents the critical path we have to build in order to have the bare minimum on ECE Day. The other parts are optional. In this section, the critical components will be developed, then the optional components.

3.1. Backwards link

The backwards link will use radio communication. The bandwidth of the backlink is critical. We will use nRF24L01, which can communicate up to 2 mbps. This transceiver communicates with the Arduino through SPI. At the drone end, there will be an Arduino which takes input from the photodiode array (4 bits in parallel) and outputs it to the transceiver using SPI interface. The base station's "Primary Station Controller" will receive data from the radio transceiver on the ground through SPI. The nRF24L01 has capabilities of communicating at 2 Mbps which is more than enough for our customer's requirement of 250 Kbps.

3.2. Primary Station Controller

The station controller will be a RasPi. This will be where the tracking algorithm lives. The tracking algorithm will take the 4 bit feedback from the RF transceiver as input and output signals to the laser and servo controllers. From the 4 bits transmitted by the drone's photodiode array, the tracking algorithm can determine how off centered the laser beam is from the center of the array. Using this information, the Kalman Filter will update its model of the drone's trajectory and predict the drone's position and speed at the next time step. This information is passed to both the laser controller and the servo controller. The specifics of the Kalman Filter will be discussed in Section 5.

3.3. Laser Controller

The Laser Controller will be an Arduino. The job of the laser controller is to take the position prediction from the Kalman Filter as input and output control signals to the laser steering mechanism. Using the position (given in rectangular coordinates), the polar coordinates (angle, pitch and distance) are calculated. If the laser were mounted on a servo mechanism with two stepper motors, this information can be inputted directly into the motor controllers. If the laser is steered using the MEMs mirror, another function must convert the position vector in polar coordinates to degree of tilt (of the mirror) with respect the x and y axis. This function is also a function of the angle of incidence of the laser on the mirror (which is constant).

3.4. Data Controller and Laser

This is the simplest part of the critical loop. The Laser will be modulated using on/off keying. The data controller will turn the laser on and off using the the UART hardware in the Arduino. From the Data Controller's perspective, this is no different than sending Serial communication over a wired connection.

3.5. Photodiode Array

The photodiode array is an array of 4x4 photodiodes. We may expand to 5x5 or even 6x6 next semester. The output of the Photodiode array circuit is a four bit number corresponding to the position of the photodiode that sees the highest intensity of light. The laser beam will most likely turn on multiple photodiodes, since the width of the laser beam will be larger than the distance between adjacent photodiodes by design. We will build a winner-takes-all circuit that will output the 4 bit value of the photodiode seeing the highest intensity from the laser, which will best approximate the location of the laser beam's center.

Aside from tracking, the photodiode array also has to demodulate the message encoded in the laser beam. For this, each of the photodiodes must also output a digital value (using the circuit in Appendix), and connected to an OR gate. The OR gate's output is the data downlink.

3.6. Motion Controls

Using the current setup, the sole purpose of the laser is to communicate data to the drone. Nothing is done with that data. In an application such as drone racing, this data will likely be used to control the drone. If we are able to complete the critical loop, we will try to control a drone using data transmitted by the laser. To do this, we must build our own drone from scratch. Once finished, we would need to connect the data output from the photodiode array to the Rx pin of the flight controller.

3.7. Camera and Coarse Control

Currently, we must assume that the laser is initially aimed at the receiver array, and that the laser never loses track of the drone during flight. This is an unrealistic assumption, and we must have a way to acquire the rough position of the drone. We will do this with a RasPi camera mounted on a servo. There will be an LED pattern mounted on the drone's receiver that the camera will look for.

4. First Semester Progress

In our first semester, we wanted to create minimum viable products for each major component and connection of the critical path shown in the system diagram. For our first round of deliverable testing, we came up with two essential tests for our product. The objective of our first test is to showcase the communication aspect of our project. Since the purpose of the project is to maintain laser communication with a moving target, establishing laser communication is an integral part. A successful test demonstrates that the receptor circuit and laser modulation mechanism work as intended. Our product relies on a closed loop system and the communication between the ground station and the drone is one of the most important components. Once we achieve this first objective, we can focus on the more difficult problem at hand, our tracking mechanism.

Tracking is the second of our two main problems in this project, and also the more difficult. While the tracking setup demonstrated during the test is simple, it is a good baseline to build more sophisticated tracking mechanisms upon. This test demonstrates our ability to control the laser with a servo and to maintain a feedback loop between the receiver and transmitter using a linear photodiode array. The tracking test was very constricted by the lack of equipment to measure distance and positioning of the laser to the photodiode array. However, we managed to demonstrate that the laser would lose sight of the photodiodes within a distance of 20cm, which doesn’t even reach the minimum requirement for the operating range.

Past the first deliverable testing, we have built the 4x4 photodiode array and initiated testing of a tracking algorithm for a two dimensional space. This is a crucial step in the course of the project that will require a lot of time to develop and debug. We have also managed to control the MEMS mirror to point the laser to the locations we desire. We still need to fully test the capabilities of the mirror, and identify how small of a change in position it can achieve to increase precision of the laser at longer distances. In this manner we can reach a decision regarding the use of a stepper motor or the MEMS mirror.

5. Technical Plan

5.1 Establish Backwards Link

Establish a radio link from the receiver to the base station. This link will use the RF transmitters already purchased. They will interface with Arduinos using SPI. The transmission will be 4 serial bits. The four bits are outputted by the photodiode array circuit to indicate the position of the photodiode that is on. Design considerations that need to be ironed out include the following:

- Time between transmissions.
- Transmission rate. Nominal maximum rate is 2 Mbps, but this may be higher than what we can achieve in reality.

5.2 Build Servo Control

This part includes linking the stepper motors with drivers and testing to make sure the stepper motors are able to move at a speed specified and to a position specified to a certain degree of precision. The precision needed has already been calculated, and the components have already been chosen accordingly. The main task remaining is the physical assembly (including 3D printing a bracket) and wiring everything up.

Back-end functions need to be written to interface between the tracking algorithm and the motors. This may include setting up the timer counter in the Arduino to control the step frequency of the motor.

5.3 Build MEMS Control

There are three main tasks to this part: 1) setting up the MEMS mirror 2) writing the polar coordinates-to-tilt function, and 3) physical mounting.

<u>Task 1:</u> This is partly done. The deliverable outcome of this task is to output x and y tilt to the mirror and verifying that the mirror tilts to the correct angle. For example, if the laser beam shines vertically down on the mirror and the mirror is tilted 45 degrees, then the laser should be reflected horizontally. A function should be written that takes two inputs -- tilt in y direction and tilt in x direction -- and outputs the appropriate signals to the MEMS mirror.

<u>Task 2:</u> We need to write a function that takes polar coordinates as input and outputs the tilt of mirror in x and y directions. The angle of incidence of the laser on the mirror is

fixed and should be taken as a constant parameter. This part mainly involves doing some basic trigonometry to figure out the relations.

Task 3: The bracket that the MEMS mirror is currently mounted on heavily constrains its line-of-sight. More specifically, the MEMS mirror is set back from the aperture and cannot see from 30 degrees above the horizon. We will need to mount the MEMS mirror so that it can "see" the horizon. We also need to determine the optimal angle of incidence of the laser, and mount the laser in a way that minimizes the blind spot created by the laser.

5.4 Tracking Algorithm

The tracking Algorithm is divided into two separate tasks: Theory and implementation. The deliverable of the theory stage is detailed pseudocode for the Kalman Filter. Implementation involves converting the pseudo code into functional C code. The theory part can be broken down into more parts as follows.

Task 1: Defining a model for the drone's trajectory. As an example, the most obvious way to model a trajectory is using position, acceleration and velocity vectors. Using this model, the Kalman Filter must store three vector values (nine scalar values). The Kalman Filter must output the predicted position to the MEMS or Servo Controller at regular time intervals, which needs to be decided upon. This time interval should be relatively small (~ 1 ms) to avoid jerky movement.

Task 2: Defining the update function for the Kalman Filter. This is somewhat more complicated than the first task. This function will be called every time the 4 error bits are received from the drone's receiver. As an example, if a receiver starts from rest, the Kalman Filter models the position as $x(t) = 0$. When the receiver moves up and the error becomes -1, then the Kalman Filter might change to $x(t) = t$.

Task 3: Making the Kalman Filter more robust. At this stage, the Kalman Filter should work under normal conditions. However, edge cases need to be handled. For example, if the receiver is moved up, turned 180 degrees, and then moved back down, the tracking algorithm might not know that the receiver turned and thus continue moving up.

Task 4: Implement the Coordinate transform function. Convert Rectangular Coordinates to Polar Coordinates. This is trivial, but note that we may find it easier in the end to use an alternate coordinate system.

5.5 Second Deliverables Testing

At this stage, we plan to deliver the following.

- A receiver which is able to receive data at the data rate specified by the Engineering Requirements.
- A tracking algorithm which is able to track the receiver provided that its orientation does not change.
- The receiver circuit should be finished and the feedback loop functional.
- Either MEMS mirror or servos to steer the laser.

Given that the receiver remains in the same orientation, the laser should track the receiver as it is moved up to 10 m/s and accelerated up to the acceleration due to gravity. We will demonstrate this by holding the receiver in front of the laser and moving it in an unpredictable manner.

5.6 PCB Design

Design a PCB for the receiver and have it shipped by at least two weeks before functional testing. The design of the PCB will depend upon the circuit from previous tasks. The photodiode array will probably by 5x5 or 6x6.

5.7 Tracking Algorithm Testing

This will be the least straight forward task for next semester. Many of the details depend upon development of the tracking algorithm up to this point. This is where we must address the cases where the tracking algorithm fails and attempt to address the issue. Foreseeable problems that need to be addressed at this stage include:

- Changing orientation of the receiver
- Tilting the receiver away from the laser
- Momentary loss of signal from the receiver (less important)

An acceptable deliverable from this task will have solved the first two bullet points above by functional testing week.

5.8 Minimum Deliverable by Functional Testing

Redo the test from previous testing with two complications. We can change the orientation of the receiver and tilt it. At this point, we may try to mount the receiver on a drone. However, going from holding the receiver in hand to mounting it on a drone is not difficult. It can be argued that following a hand is much more difficult than following a drone, since a drone's trajectory is more predictable. However, tracking a drone looks cooler.

6. Budget Estimate

For this project, there were no items that would constrain our budget that were not already provided to us. Our customer provided us with the expensive technologies such as the drone and the MEMS micromirror, and also some small microcontrollers. Most of the parts that we had to cover through the budget were components that we used to build the tracking system and receiver device that goes on the drone. We have already obtained the items that will be most crucial for the completion of this project, so the addition to this list will be minimal and of no significant cost.

Item	Description	Cost
1	Laser Diodes (10)	$2.59
2	Raspberry Pi 3	$40.00
3	Arduino Leonardo (2) *	$39.60
4	Photodiodes (25)	$16.69
5	RF Transmitters/Receivers	$11.98
6	Raspberry Pi Camera Module V2	$22.99
7	Stepper Motor Driver DRV8825	$11.59
8	Stepper Motor NEMA 17 (2)	$23.98
9	4 Quadrant Actuator MEMS A1M16.5 Micromirror *	$499.00
10	DJI Phantom 4 Drone *	$1,199.00
	Total Cost	$1,867.42

*Provided by customer

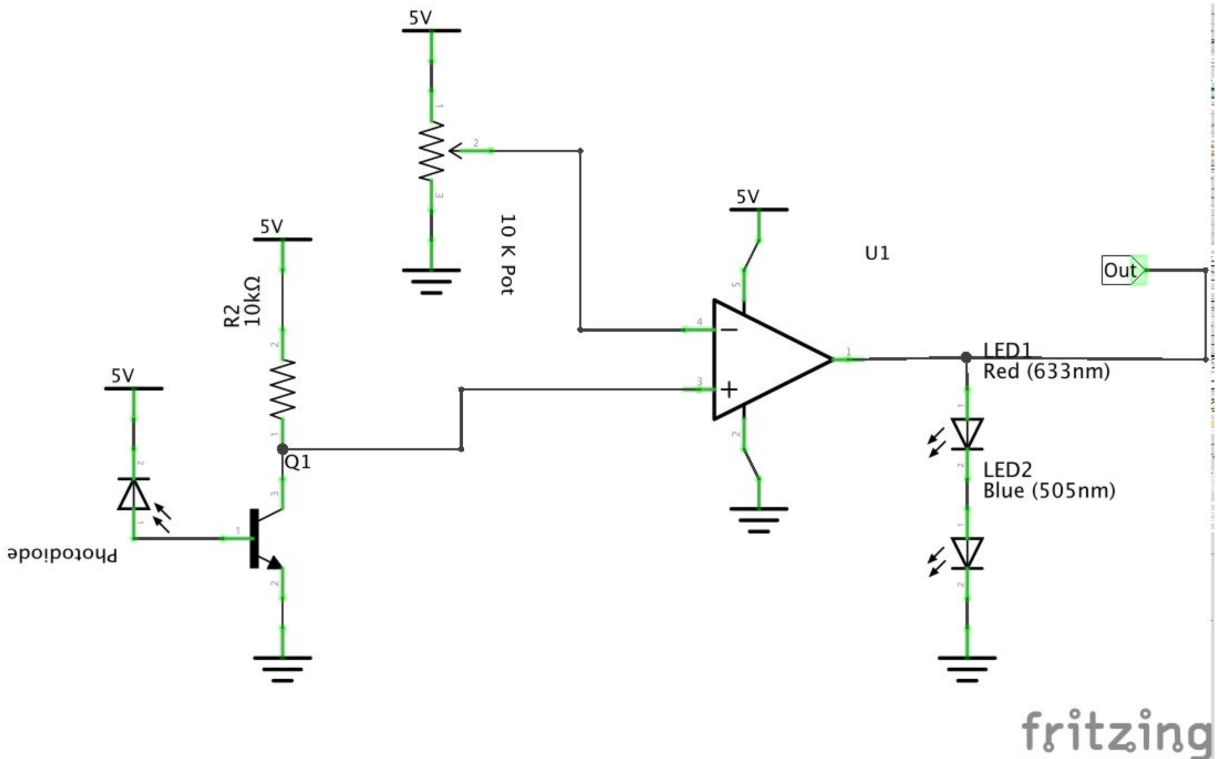
7. Attachments

7.1. Appendix 1 – Engineering Requirements

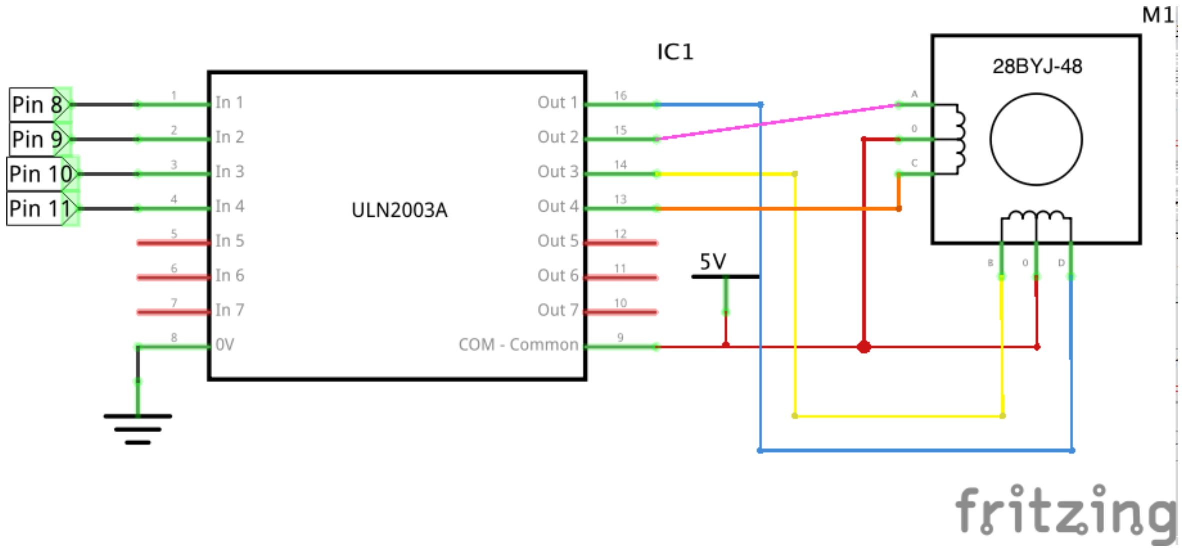
Requirement	Value, range, tolerance, units
Optical beam type	Class 1 or Class 1M
Weight of receiver on drone	≤ 100 grams
Battery powering receiver on drone	~ 10mW
Coarse position of drone acquired	5 seconds
Fine position of drone acquired	1 second
Range of operation	1 - 20 meters
Tracking speed	10 m/s or 36 km/h
Data rate to receiver	≥ 250 Kb/s

7.2. Appendix 2 – Gantt Chart

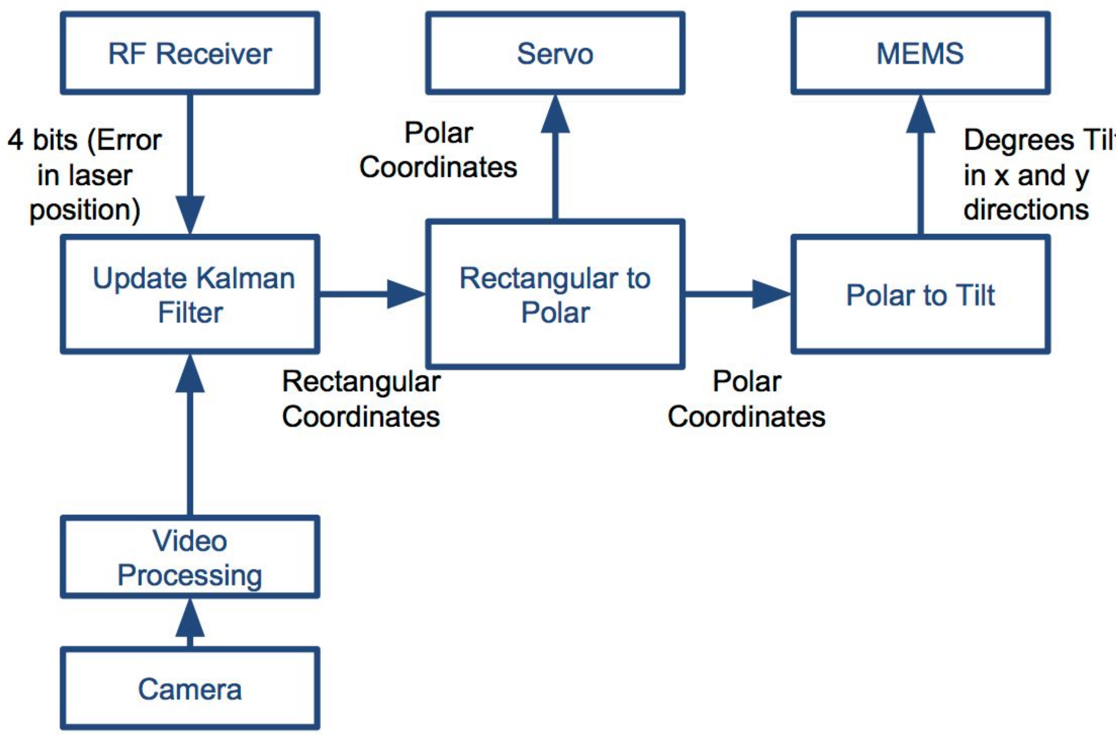
7.3. Appendix 3 – Photodiode Circuit



7.4. Appendix 4 – Stepper Motor Control Circuit



7.5 Appendix 5 – Tracking Algorithm Block Diagram



7.6 Appendix 6 – Receiver Block Diagram