Boston University
Electrical & Computer Engineering

EC464 Capstone Senior Design Project

User Manual

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

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: 04/18/2018

Table of Contents

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

Wireless optical communication has many benefits when compared with radio frequency communication. Currently the radio frequency spectrum is cluttered and requires a license for use. Our goal is to implement an alternative method for rapid data transmission, that 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. Free space optical communications can potentially transmit data at rates of up to Gbps. This is why optical wireless communication is a viable alternative to radio frequency communications.

This method of communication has some characteristics which make it difficult to implement for practical purposes. The most pressing issue with using optical wireless communication is that in order for data to be transmitted, the laser must be pointed at a receiver. This means that the azimuth and elevation angles of the target with relation to the transmitter must be known to within an extremely high degree of accuracy. This is not an issue for communicating with stationary receivers. However if optical communication is to replace radio frequency, it must be able to transmit data to moving receivers. In the past, work has been done by another senior design group with two dimensional motion and knowledge of the receivers' trajectories. The challenge with our project compared with the past work was that we were tracking in three dimensions and with no prior knowledge of our receiver trajectory. Our project's goal is to maintain a constant communication link with aerial vehicles. Without our project this communication link cannot be maintained for moving aerial vehicles resulting in gaps in communication and lost data.

We were provided with a MEMS mirror for pointing the laser and some base code for changing the position of the mirror. From that we constructed a feedback loop which we used to automate the reflection of the laser to keep it pointed at our receiver which is attached to a rapidly moving unmanned aerial vehicle.

2. System Overview and Installation

2.1. Overview Block Diagram

The functionality of this project is straightforward; it tracks a device mounted to a drone using visible light. Our laser tracking system has five main components on its outermost layer, the MEMS mirror beaming the laser, the photodiode array, a camera, a LIDAR and a pair of radio frequency transceiver modules.

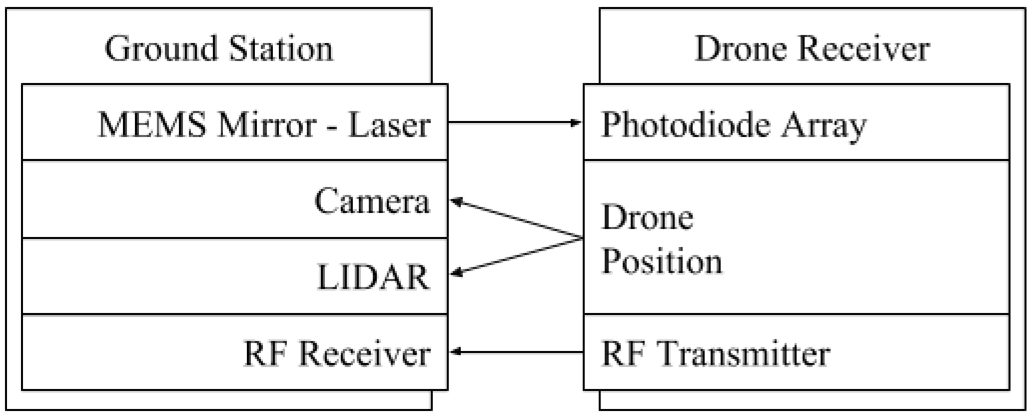


Figure 2.1: System Overview Diagram

The system works as follows, on startup the camera will locate the object and identify its position. Once identified it will address the laser and move it to point at the general direction of the target. Here the laser will scan the area until it makes contact with the photodiode array. The photodiode array will transmit the position of the laser as it hits the photodiodes, and the laser will then be able to readjust moving back to the center of the array with assistance from the LIDAR.

2.2. User Interface

For this project we have little input from the user, but we still allow for some measure of customization. The coarse tracking algorithm is where all of our user interface can be found due to the wide range of scenarios that the product can be used and how they would affect the traceability of the transmitter. To this end we enable the user to tweak the HSV range specifications for the most accurate tracking.

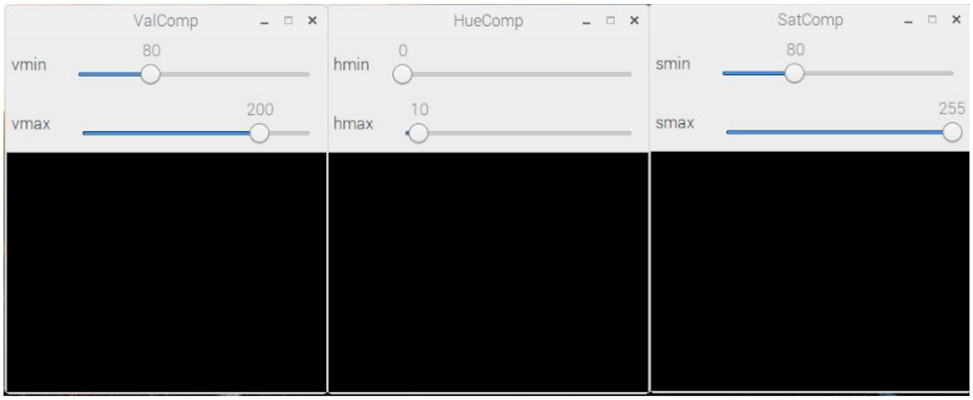


Figure 2.2: HSV Ranges UI, accessible through the Raspberry Pi

2.3. Physical Description

The physical components of our project are separated into the Ground Station and the Drone Receiver.

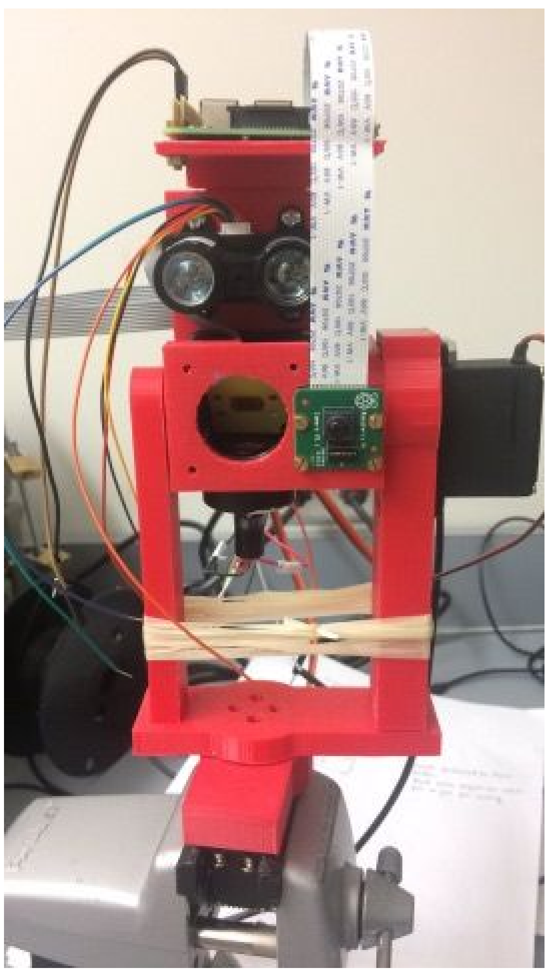


Figure 2.3a

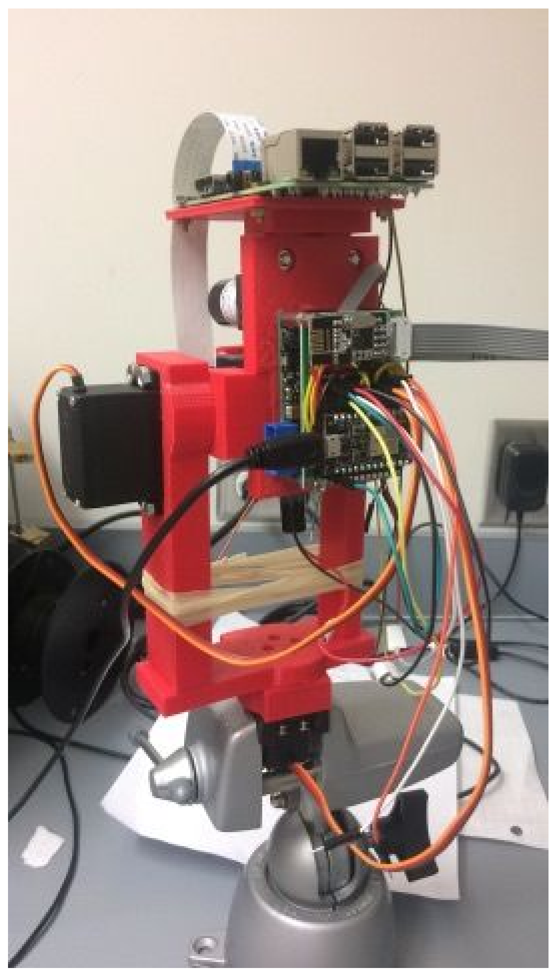


Figure 2.3b

Figure 2.3a: Ground Station Showing Lidar, MEMS Mirror and Pi Camera

Figure 2.3b: Shows Particle Photon with RF Transceiver Module

The Ground Station is composed of a base that holds everything in place and ensures minimal movement of the parts. Mounted on the metal base is a 2-part 3D printed encasing. Each part of the case holds a Servo motor to enable pan and tilt movement. The tilting section of the case contains the Raspberry Pi, the Pi Camera Module, the LIDAR, the Photon and of course the MEMS mirror with the laser attached to it.

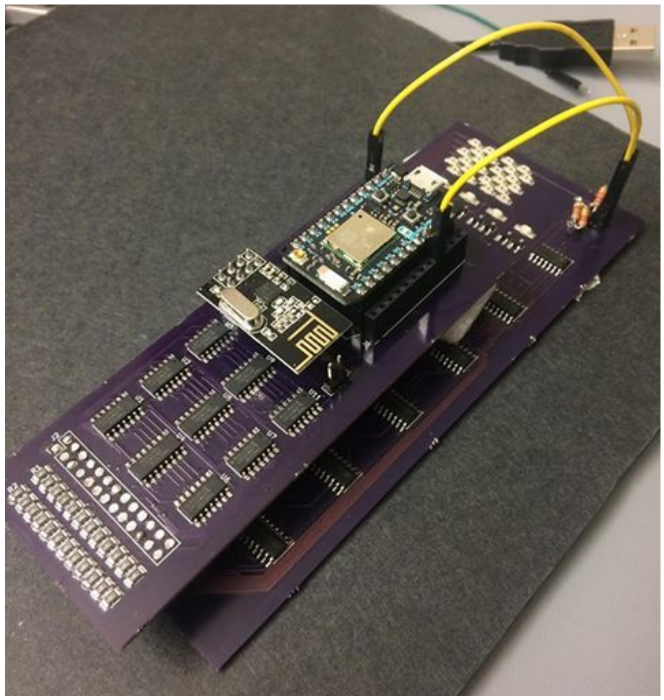


Figure 2.4: Drone Receiver PCB with Particle Photon and RF Transceiver

The Drone Receiver consists of two printed circuit boards, a particle photon and a RF transceiver. The sensor board and the mux board are designed to connect vertically via the 2x13 connector near the bottom edge of each board. This connector is designed so that it cannot be connected in the wrong orientation. The mux board connects to the particle photon via two 1x12 headers. The photon must be attached in the proper orientation, with its micro USB port towards the edge of the board. The nRF24L01 transceiver attaches to the 2x4 header on the mux board, with the body of the transceiver facing in towards the rest of the PCB.

2.4. Installation Setup and Support

Given that the microcontrollers are already setup and the required dependencies are installed, the only initialization steps are to power up the devices. The power supply for the ground station should be set to output 5 volts by pressing the output button and twisting the knob until the display reads 5.00 volts.

3. Operation of the Project

In either mode of operation, the receiver boards each require a 5-volt power source. This can be provided by a drone's BEC circuit, or by the AA battery pack provided. The +5V line should be connected to the pin closest to the "J1" label, and the GND line should be connected to the remaining pin. The sensor board has a sensitivity that can be configured by the resistor bridge near the power port. To reduce sensitivity, the user may reduce the resistance across "R1", or increase the resistance across "R2". Doing the opposite will increase the sensitivity of the sensor.

To achieve communication, connect serial Tx to the pin indicated on the transmitter. Connect ground to pin indicated. This pin modulates the laser. Connect Rx to D7 of the the photon on the receiver. Connect ground pin to photon on receiver. The communication can be treated as a UART connection.

There are two separate power supplies needed for the transmitter. All transmitter components are powered through the RasPi. The RasPi should be powered by a wall adapter or laptop USB connection. Use a short micro USB cable to connect the Raspi to the Photon, this serves as serial connection and power supply for Photon, laser, LIDAR, MEMS, and radio. The servos must be powered separately via the jack indicated. This must be a 5V 5A power supply. A smaller power supply will result in erratic servo behavior.

3.1. Operating Mode 1: Manual Operation

Manual operation is fine tracking without coarse tracking integrated and requires that a human manually instantiate the initial laser lock on the target. This works similar to automatic operation in that once the laser is pointed at the receiver on the drone, it stays on the drone using the radio frequency feedback. The biggest difference between manual and automatic operating mode is that the user must initially place the receiver on the drone in front of the laser before it begins moving. This is true anytime the laser loses the drone, the human in the loop must take care of coarse acquisition instead of the raspi camera.

3.2. Operating Mode 2: Automatic Operation (not fully implemented)

There are two main modes of automatic operation. First there is the coarse acquisition which is the mode when the laser is pointed at the drone. The other mode is fine tracking when the laser is pointed at the receiver and the ground station is getting feedback from the drone.

In the first mode of operation the servos controlled by the Raspberry Pi are rotating around until the raspi camera detects the drone. Once the drone is detected the laser is pointed at the drone and the drone sends feedback to the ground station telling it which photodiode has been hit and the product transitions into the second mode of operation.

In the second mode of operation, the camera is no longer trying to find the drone. Instead the laser is being moved to try to keep it centered in the photodiode array. In order for the laser to be kept in the array successfully, the radio transmitter on the drone should be sending feedback which the ground station uses to calculate where the laser should move. If there is a temporary obstruction which causes the laser to lose the drone, we transition back to the first operation mode.

These two modes currently work independently of each other. We are currently working on integrating both modes together.

3.3. Operating Mode 3: Abnormal

Abnormal Operation occurs when the laser loses track of the drone because it is out of range. The system will only operate normally when the drone is in range. When the drone is not in range the laser will move around rapidly and the camera on servos in will move around somewhat spastically until either power is disconnected or the drone comes back in range.

3.4. Safety Issues

The laser used by the laser tracking system is eye safe, although the manufacturer recommends that it not be purposefully pointed into someone's eyes. In addition the effects of staring directly into a laser may be exacerbated by the use of a lense that magnifies the intensity of the laser. Avoid contact with water for outdoor use.

4. Technical Background

Laser tracking can be broken down into two major components, Fine Tracking and Coarse Tracking. We must be able to find the object we are looking to track with our Coarse Tracking Algorithm before adjusting ourselves to aim at center of our target.

4.1. Coarse Tracking Algorithm

The coarse tracking algorithm is run from the Raspberry Pi. This code required extra processing power provided by the Raspberry Pi due to the volume of image manipulation that was is required to track an object constantly. The Raspberry Pi Camera Module V2 was used to access the video stream, however any USB camera with a minimum quality of 720p can also be used.

This algorithm utilizes the library OpenCV for all of its camera related manipulations and video retrieval. The video resolution is set to 320x240 pixels for improved performance on the Raspberry Pi and reducing lag. Once the video stream is introduced to the algorithm it will enter a while loop that will first blur the image using a Gaussian blur to get rid of any sharp edges that will interfere with the image processing aspects. Once blurred, it will separate the image of that instance into 3 different spectrums; hue, saturation and value. Each instance will provide a different visibility range that can be overlapped to obtain a trace of the tracked object. Once we overlap these images we run two iterations of eroding and dilatation masks to discard any unwanted smaller blobs, as OpenCV refers to the overlapping figures. The program will then select the larger visible blob, find its center of mass and record this position as (x,y) where the position represents the pixel location. This process occurs for each frame of streamed footage and is extremely taxing on the Raspberry Pi.

In order to communicate with the Photon controlling the laser position and servos, we use serial communication through USB to transfer the recovered information from the previous steps.

4.2. Fine Tracking

Fine tracking has several key components, the array of sensors on the receiver and circuit board, the radio frequency transceiver, the lidar, the actual tracking algorithm, the MEMS mirror and the particle photon.

4.2.1. Photodiode Array and Circuit Board Design

The receiver circuit consists of two PCBs, a particle photon, and a transceiver module. The PCBs are the “sensor” board, and the “mux” board.

The primary component of the sensor board is a 6x4 photodiode array which detects the presence and position of a red laser. These photodiodes produce an analog signal, which is then cleaned by comparison with a resistor bridge. The binary on/off data from each photodiode is made available outside the board via a 13x2 header. If used alone, each pin of output requires a pull-up resistor on its output. The sensor board's resistor bridge is configurable to allow the optical sensitivity of the detection circuit to be changed after the board has been built.

The mux board makes the 24 signals from the sensor board available to the particle photon. The primary components of this board are a 32-to-1 multiplexer, and a 24-input OR gate. The multiplexer allows the photon to read each photodiode one at a time, which is used for detecting the laser's position. The OR gate allows the photon to detect if the laser is present, which we use to transmit data across the laser link.

4.2.2. RF Transceiver (nRF24L01)

From the photodiode array, we are able to extrapolate data telling us where the laser is hitting our drone. Ideally we would want to use a laser to establish a backlink from the target/drone to our ground station relaying the relevant information however that would require us to create another Laser Tracker that would end up being far too large and too heavy to be placed on a drone. Our team decided that for the sake of the project, using an RF Transceiver would suit our needs well.

The RF Transceiver uses the 2.4 GHz band to communicate between devices. The transmitter and the receiver modules must be on the same "pipe" in the band in order to communicate properly. These modules are connected to our Particle Photon Microcontrollers via SPI and has a data rate of either 250Kbps, 1Mbps or 2Mbps.

These modules have a overhead of about 2ms from getting the information from the photodiode array and sending it out, to receiving it on the ground station. This model of transceiver is also limited in its range of 100m. Because it operates in the 2.4 GHz band, there may be a lot of noise coming from other comRF mon signals operating in the same frequency such WiFi and Bluetooth.

4.2.3. LIDAR

We use the Lidar to find the range of the drone and use that combined with the azimuth and elevation angles to determine how much we should adjust the angles of the laser in order to move it closer to the center of the photodiode array. The use of the lidar allows us to adjust for the varying range of the drone as the angles will differ greatly when the drone is 1 meter versus when it is 10-15 meters away. When the drone is farther away we want to make smaller adjustments to the angles of the laser and when it is closer bigger changes. Adding the Lidar will improve the range of ranges at which fine tracking will work. The Lidar has two main components, one which outputs a pulse of light and one that receives the reflection. The Lidar uses the time it takes for the light to travel to the target and back as well as the speed of light to calculate the distance between the lidar and the object that the light reflects off of. The lidar transmits the range of the target to the photon using a pulse width modulated logic signal. The photon measures how long the signal is high and uses the conversion rate, 10 microseconds high corresponds to 1 centimeter of distance. The lidar needs to be hooked up to a 5 volt power supply and ground to work.

4.2.4. Tracking Algorithm

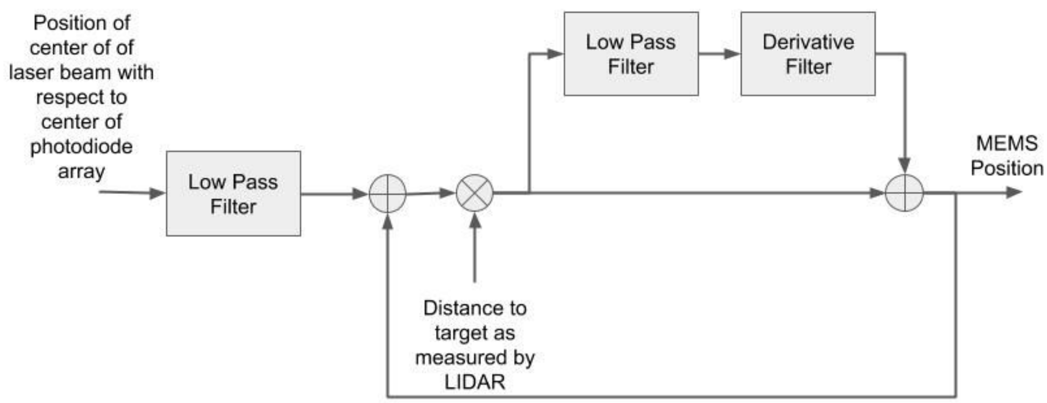


Figure 4.1 Tracking Algorithm Block Diagram

Simply stated, the tracking algorithm we use is a feedback loop where we move the laser in the direction of the center of the photodiode array. The simplest system would adjust the position of the MEMS by some constant times the distance from laser to target times the position from of the laser from the center of the array as measured by the photodiode array. In practice, tracking is complicated by (1) nonlinearity of voltage output to MEMS mirror with respect to distance distance moved by laser, (2) movement of servos, (3) latency of feedback signal, and (3) noise in the feedback signal.

We do not address the first issue (nonlinearity). We justify this by assuming that the MEMS does not move much from rest, so we can approximate a linear relationship between Δd and $\Delta\theta$ where

Δd is the distance between the center of laser beam and center of photodiode array, and $\Delta\theta$ is corresponding difference in angle the laser needs to move.

To address noise in the feedback signal, which is especially a problem with servo movement, we low pass filter the signal from the photodiode array. We average the past four values transmitted, but this filter is tunable. Averaging too many values causes delay in response from MEMS and poor tracking.

To address latency, we use a linear predictor to predict the next position of the photodiode array so that the tracking does not rely solely on feedback from the photodiodes. This linear predictor is implemented by a low pass filter and derivative filter as illustrated in Figure 4.1. The low pass filter is necessary, since the derivative filter is sensitive to noise. The derivative filter approximates the speed of the target over time using the last few positions. The result of the linear predictor is added to the adjustment needed to center the laser.

4.2.5. MEMS Mirror

The MEMS mirror is controlled by two voltages sent over SPI. The voltages control angle of the mirror around x and y axes. The maximal deflection of the mirror is 6 degrees in any of the 4 coordinate directions. Using our setup, the field of view of the MEMS is roughly a rectangle $\pm$ 12 degrees in y direction and ± 6 degrees in x direction. Calculating the relationship between the distance the laser point moves with respect to the target and the voltages applied to the MEMS is a relatively complicated geometric problem. We approximate a linear relationship, since the servo should do a good job keeping the MEMS centered.

4.2.6. Particle Photon

The Particle Photon is used to control both the receiver components and the transmitter components. The microcontroller used by Photon is part of the STM32F20 series microcontrollers. It is part of the Cortex M3 family. The Cortex-M3 programming manual and RM0033 Reference manual was consulted for this project. Major hardware functions needed include Timer counters and SPI. The TCs were programmed manually for the servos, as the Photon library has some issues with controlling two servos on two channels of same TC. TC needs to operated in capture mode for LIDAR and waveform mode for servos.

5. Cost Breakdown

The following parts list has been formulated to create a beta version of this project. The list makes sure to include the items that would make the product the most efficient and functional. The drone is not included on this list as it is not considered part of our product, the user would need to have one or something else to mount the receiver.

Projected Costs for Production of Beta Version				
Item	**Quantity**	**Description**	**Unit Cost**	**Extended Cost**
1	1	Laser Diodes (Pack of 10)	$2.59	$2.59
2	1	Raspberry Pi 3	$40.00	$40.00
3	2	Particle Photon *	$19.00	$38.00
4	1	Sensor PCB (Minimum order: 3)	$180.20	$180.20
5	1	Mux PCB (Minimum order: 3)	$75.09	$75.09
6	1	RF Transmitters/Receivers (Pack of 10)	$11.98	$11.98
7	1	Raspberry Pi Camera Module V2	$22.99	$22.99
8	1	Stepper Motor Driver DRV8825	$11.59	$11.59
9	2	Stepper Motor NEMA 17	$11.99	$23.98
10	1	4 Quadrant Actuator MEMS A1M16.5 Micromirror *	$499.00	$499.00
11	1	LIDAR-Lite V3*	$129.99	$129.99
12	1	3D Printed Encasing	$15.00	$15.00
			Beta-Version Total Cost	$1050.41

*Provided by customer

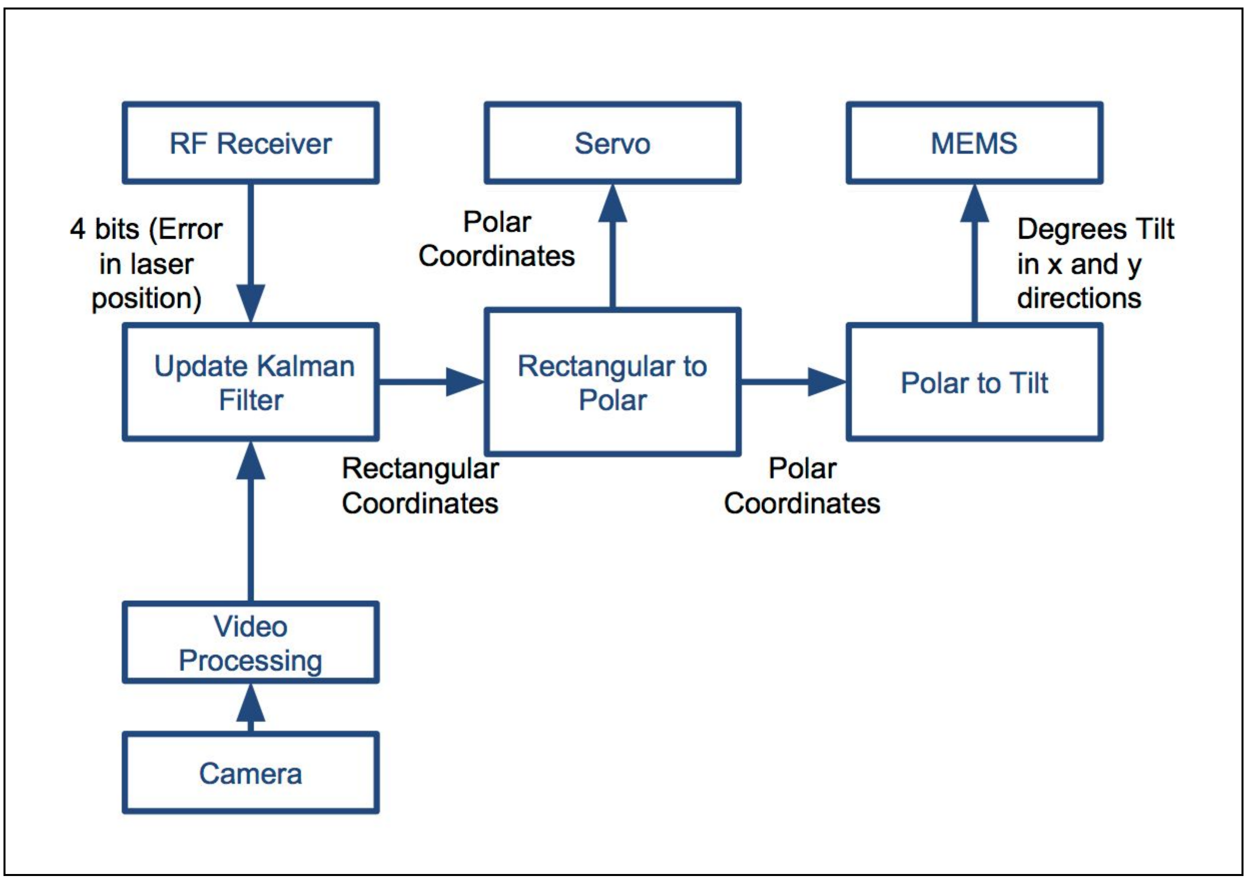
6. Appendices

6.1. Appendix A: Technical Information

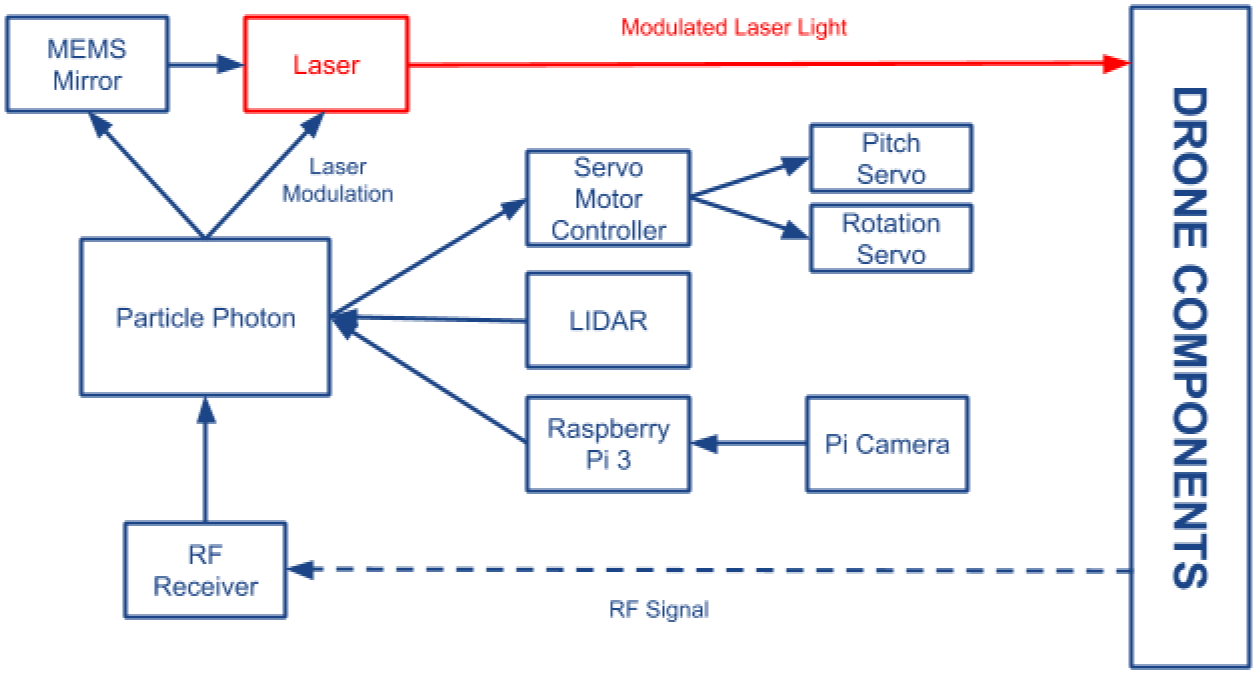
Appendix 1: Engineering Requirements

Requirement	Value, range, tolerance, units
Optical beam type	Class 1 or Class 1M
Weight of receiver on drone	$\leq$ 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	$\geq$ 250 Kb/s

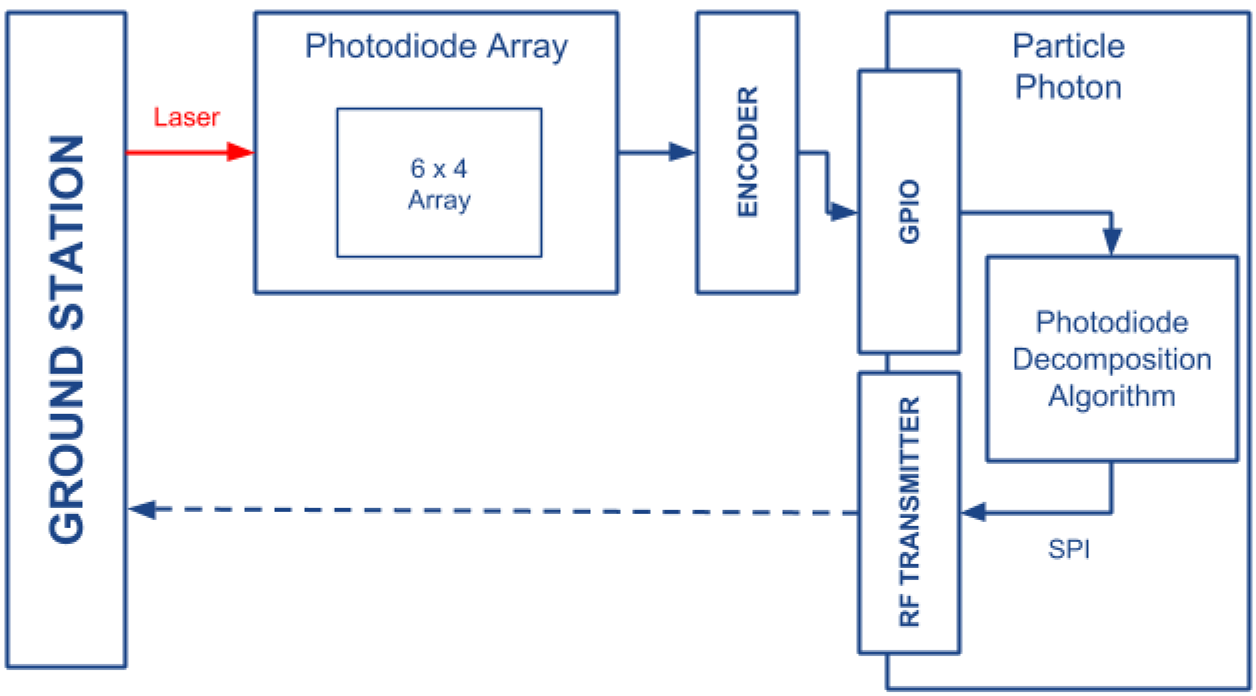
Appendix 2: Tracking Algorithm Block Diagram



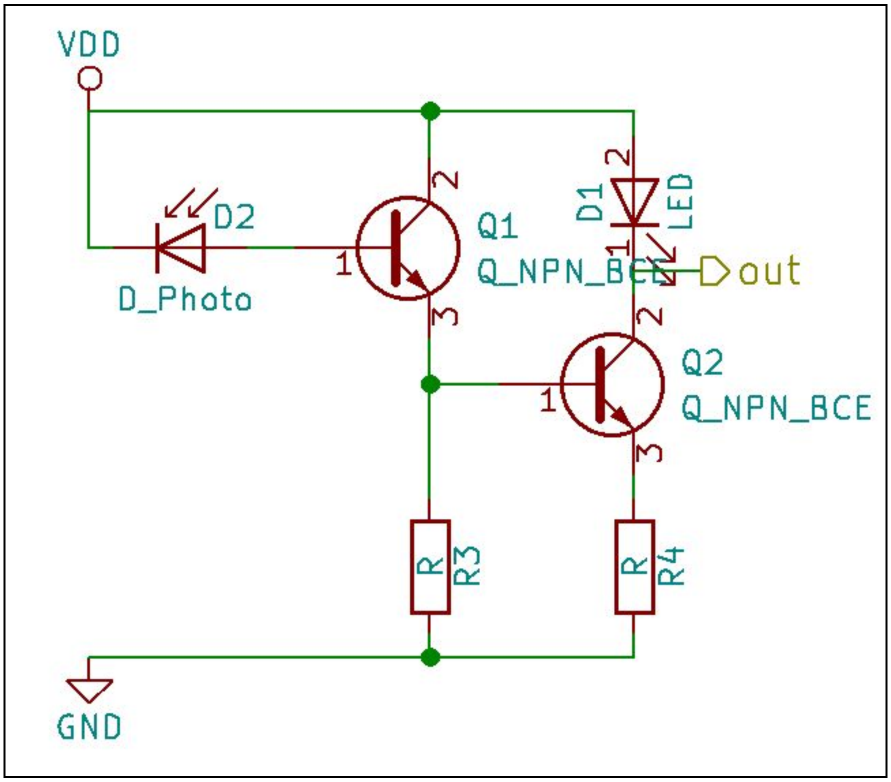
Appendix 3: Ground Station/Receiver Block Diagram



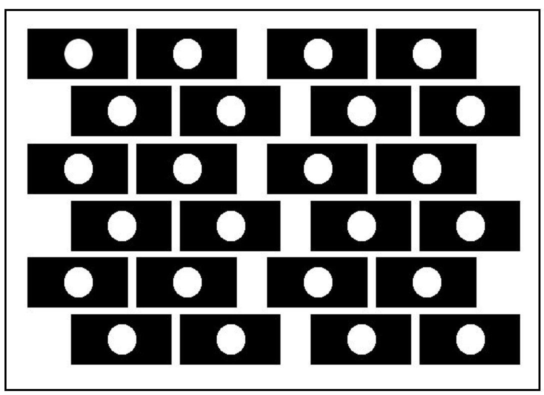
Appendix 4: Drone/Transmitter Block Diagram



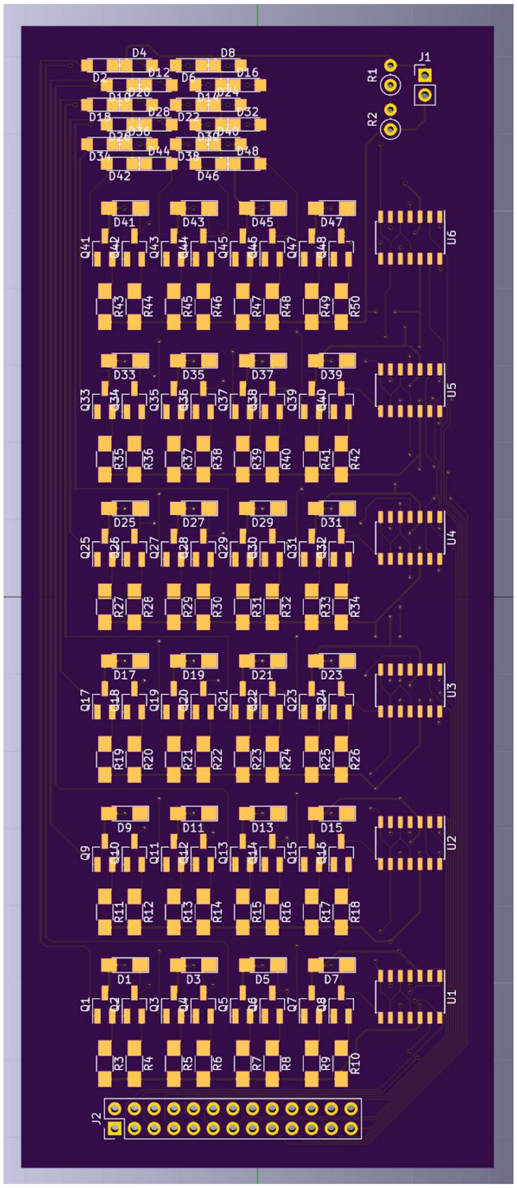
Appendix 5: Photodiode Circuit

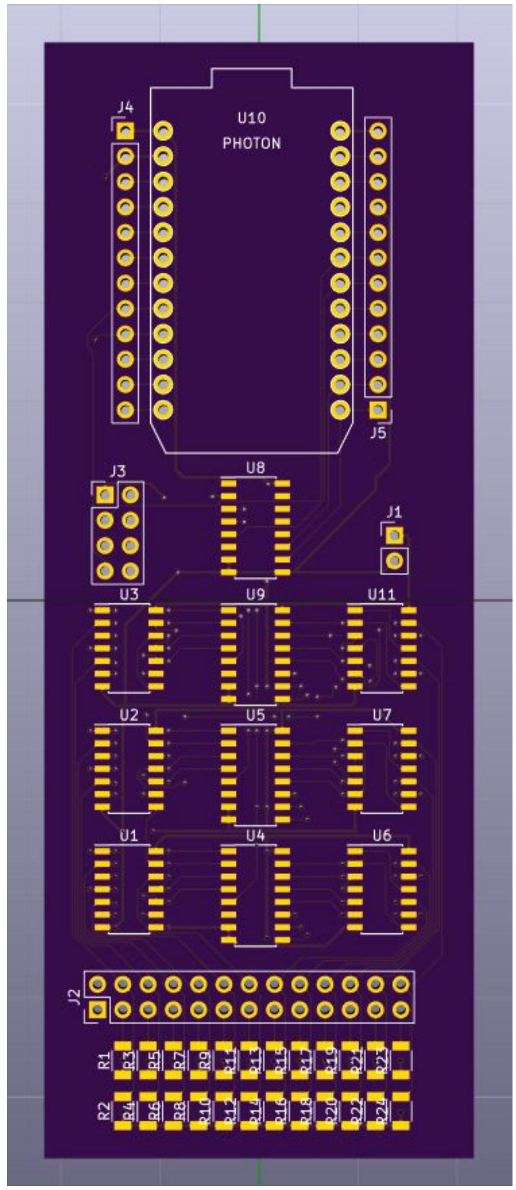


Appendix 6: Photodiode Array Design



Appendix 7: PCB 3D Renders





6.2. Appendix B: Team Information

Aviva Englander englanda@bu.edu

Aviva is a senior in Electrical Engineering. She is responsible for integrating the Lidar into the rest of the fine tracking mechanism. Her interests and experience lie primarily in signal processing and probabilistic graphical models.

Christopher Liao cliao25@bu.edu

Christopher is a senior studying Computer Engineering. Christopher developed the fine tracking algorithm and worked on the software that controls the MEMS mirror and designed and 3-D printed the casing which was necessary to integrate all components of the ground station.

Jeffrey Lin jlin96@bu.edu

Jeffrey is a senior studying Computer Engineering. He is responsible for the RF communication in the backlink to ensure the project has a closed loop. He is interested in software development where his experience in development is primarily in C/C++ and Python.

Anton Paquin paquin@bu.edu

Anton is a senior in Electrical Engineering. He designed the Photodiode Array and the PCBs essential in the project.

Eduardo Portet eportet@bu.edu

Eduardo is a senior in Computer Engineering at Boston University. He designed the coarse tracking algorithm and communication protocols between the Photon and the Raspberry Pi. He is interested in software developing and has development experience with Python, C/C++, C#, Java, Go and Ruby.