1.0: Executive Summary

The idea behind this project is to create a security drone that will provide the most economical and safe security system for cliental. Traditional security systems consist of security cameras mounted around the building with guards that patrol watch the cameras for any abnormalities. For larger complexes security systems may require the installation of poles or towers to give the cameras the correct angles to properly view the area. The cost for proper security in today's world can be extremely expensive.

The Automatic Flying Security Drone (AFSD) is an attempt to reduce the manpower needed to patrol a large open perimeter. No longer will security teams need to patrol large areas on a regular basis or have the large expense of mounting numerous security cameras around large open areas. The AFSD would instead patrol a route defined by a single member of the security personnel and stream live video data to that one person wirelessly. The idea is to limit the man power and expense of having a traditional security system. Multiple security guards to constantly patrol the area is no longer necessary. The AFSD would navigate via GPS coordinates, regularly comparing its position with its goal and correcting its course automatically. The path followed can be set and the AFSD will repeatedly follow the same path; however, predefined routes can be instantly changed by security personnel monitoring the AFSD, rerouting the AFSD to new paths. If security personnel will have full control of the path the AFSD will follow at all times and if need be can even have the vehicle hover over designated areas for set periods of time.

The AFSD will be spherical in shape, with the propellers and control surfaces located inside the frame. The external frame allows AFSD to bump into obstacles without crashing. It will also support a gyro sensor and will be able to automatically correct its attitude should any outside forces threaten its stability (wind, collision, etc.). This design allows for mobility through both open and dense areas. The AFSD will easily be able to adjust to obstacles it may encounter while flying the defined route. If a tree or building is obstructing the vehicles path, it will be able to adjust to the obstacle and maneuver around or over the obstacle and continue on route. The most important feature of the AFSD, however, is its self-maintenance: Upon completing a route, the AFSD will return to a specially-made docking station, which will charge the device in preparation for its next patrol.

As the world moves to becoming greener one of the key features of the vehicle's design is to make this device as energy efficient as possible. All devices and components of the AFSD will be selected based on their efficiency. A rechargeable Lithium polymer battery will be used to aid in energy efficiency. The most important part of the design relating to energy efficiency and sustainability is the docking station. The docking station will rely on a renewable and sustainable energy source, solar power. It will be surrounded with three 20 watt

solar panels that will produce enough energy to recharge the AFSD's battery. The solar energy will be stored and waiting for when the device lands on the docking station so it can be recharged.
The AFSD Control application will show the user a top-down satellite photo of the docking station's position and surrounding area. Using the mouse, the user will be able to define a route in one of three manners: Clicking individual points on the photo, clicking-and-dragging line segments, or drawing a free-hand route and having the computer interpolate points that resemble the route. The application will allow the user to redefine the route mid-flight, or to issue specific commands (Halt, Emergency Landing, Return to Docking Station) that will change the AFSD's behavior.

2.1: Motivation and Goals

When working on such an extensive project as this, teams should clearly know the motivational factors behind the project being pursued. The main motivation behind the AFSD was to pick a project that was cutting edge and necessary in today's world. Drones are rapidly being implemented in all different ways in today's world using a variety of cutting edge technologies that allow them to work. The design of a drone gives the flexibility of learning many different types of technology including communication, power, navigation, and many more. Being able to learn and work with a large variety of different technologies was a big motivational aspect for this project. Some of the motivations for this project were to learn working in a team environment with group collaboration, implement knowledge gathered from school and apply that knowledge to real world applications, and explore into unknown fields to gain further knowledge about electrical engineering.

The main goal of this project is to learn about relevant technologies in the electrical engineering field. The AFSD proved to be the perfect project idea to satisfy this goal because it incorporates a variety of technologies relevant to the electrical engineering field. Some of the different topics that were be researched and better understood by the end of this project are stability/control systems, communication systems, power supply systems, and navigation systems. The AFSD project idea workout out perfect giving every group member the ability to work in and learn fields they were interested in.

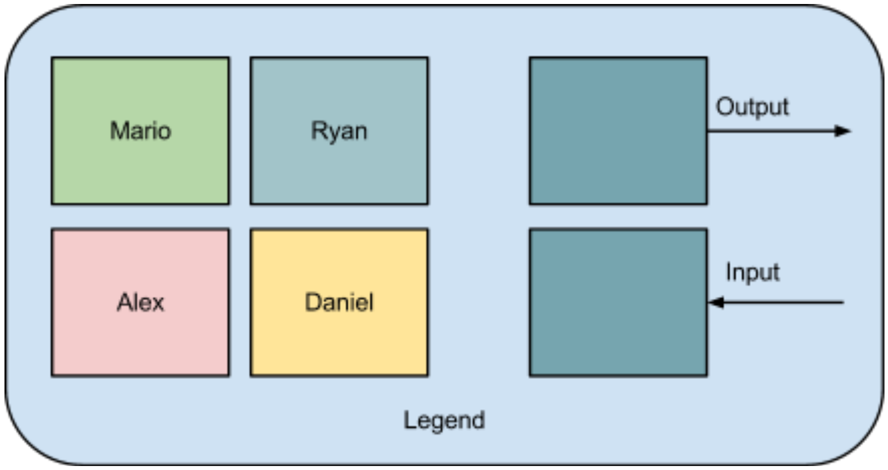


Figure 2.1.1: Project Goals Legend

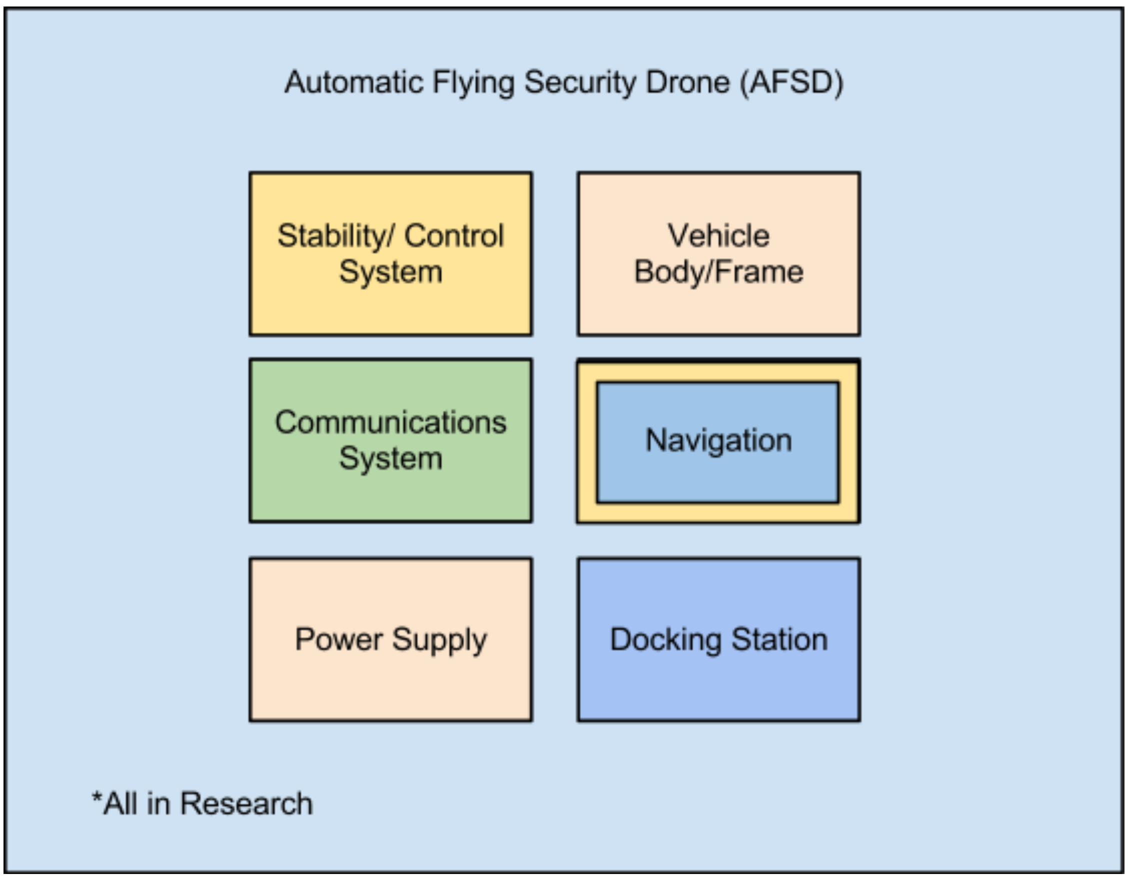


Figure 2.1.2: Major Components in Project Goals

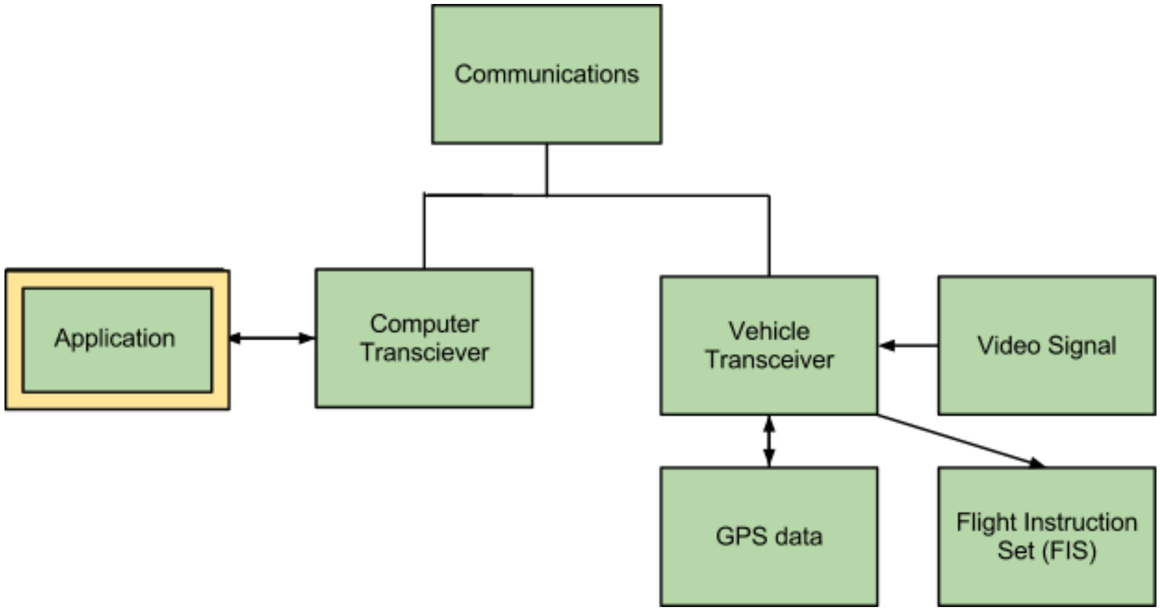


Figure 2.1.3: Breakdown of Communication

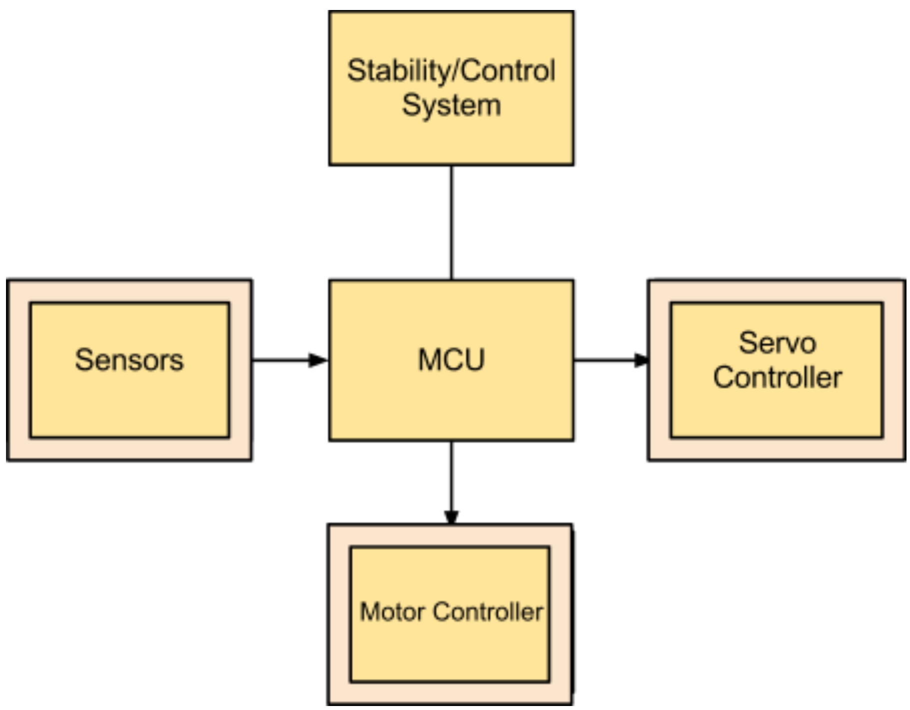


Figure 2.1.4: Breakdown of MCU

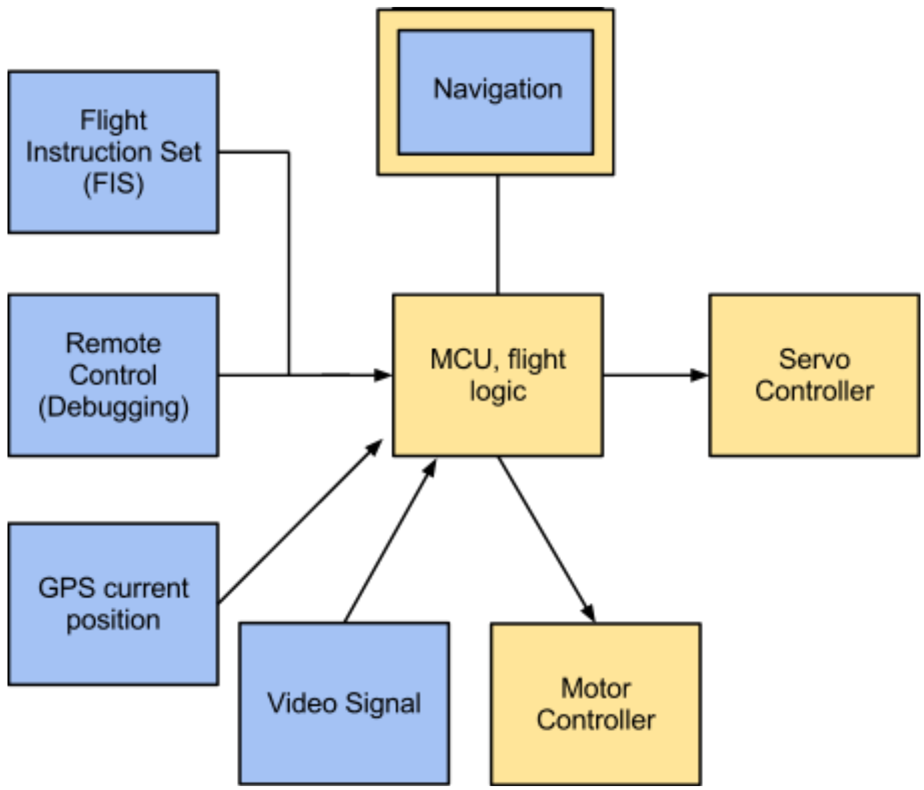


Figure 2.1.5: Breakdown of Navigation

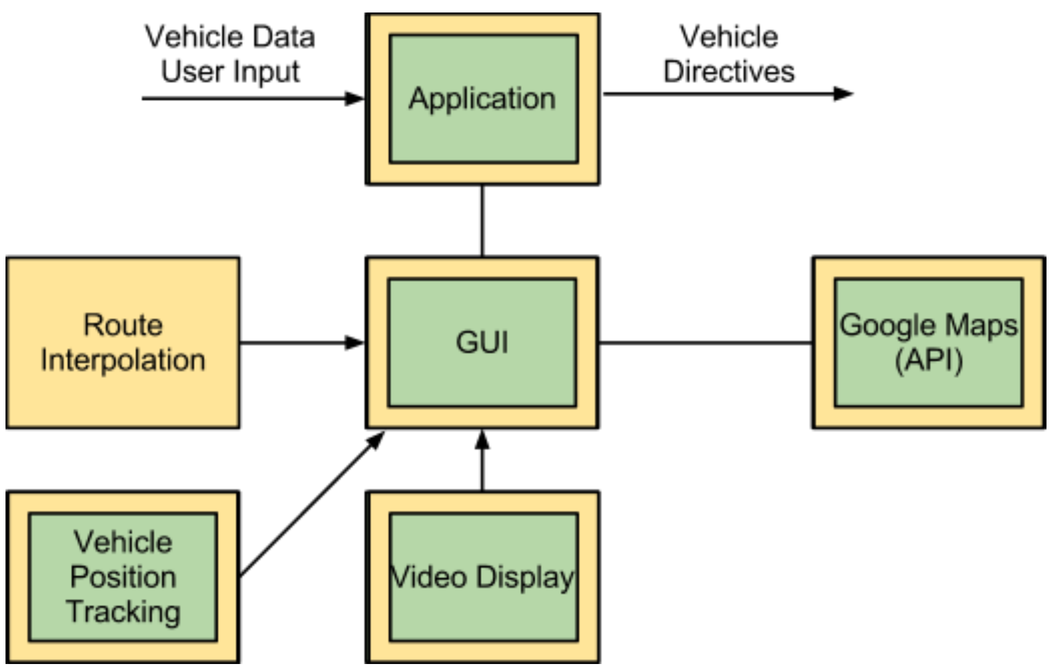


Figure 2.1.6: Breakdown of Application

2.2: Objective

The AFSD is a security drone that patrols a designated area, providing a live camera feed to a computer for security personnel to monitor. The AFSD is designed to reduce the man power needed to patrol a large open perimeter. The intended design is to make the drone completely hands free so that the security personnel won't require any knowledge of how to work the drone. The only thing required of the security personnel would be to monitor the live video feed from the AFSD on a monitor and change the path of the vehicle if need be. The ability to change the path of the vehicle is made very easy for the user. A top-down photo of the docking station's position and surrounding areas is shown on a computer screen for the user. By the simple click of the mouse the user has the ability to change the path of the vehicle. Using the mouse, the user is able to define a route in one of three manners: Clicking individual points on the photo, clicking-and-dragging line segments, or drawing a free-hand route and having the computer interpolate points that resemble the route. The application allows the user to redefine the route mid-flight, or to issue specific commands (Halt, Emergency Landing, Return to Docking Station) that change the AFSD's behavior.

For the vehicle itself, the frame is a spherical design with the propeller and control surfaces located inside the frame. It also support a gyro sensor and w is able to automatically correct its altitude should any outside forces threaten its stability. When navigating through the area being patrolled the AFSD is able to avoid collision with obstacles such as trees or buildings that maybe in its path. The AFSD automatically correct its path and maneuver around the obstacle and return to the path. With the propeller and control surfaces located inside the spherical frame, if the AFSD were to run into something or have forces act against it the vehicle self correct itself and return to a stable flight. This is very important in the design of the vehicle to help with consistency in security. It would be an inconvenience if every time a gust of wind came the AFSD was blown over and crashed.

The last major objective in this design is the self-maintenance. The docking station is the main component in making the AFSD fully hands free and self-maintaining. Upon completing a route, the AFSD returns to a specially made docking station, which charges the device in preparation for its next patrol. Around the docking station solar panels are used to provide an energy efficient way to recharge the batter on the vehicle. The solar energy is stored and waiting for when the device lands on the docking station so it can be recharged. In case the solar panels for some reason are not able to fully charge the battery an additional power supply can be switched on if need be.

2.3: Requirements & Specifications

Vehicle Specifications:

1. The vehicle needs to be light weight so the motor can get the vehicle off the ground. The goals is to keep the total weight of the vehicle under 600 grams.
2. The vehicle has a diameter of 1 foot.
3. There is a total of 8 control surfaces, 4 upper and 4 lower.
4. Propeller and control surfaces need to be inside the spherical frame.

Flight Specifications:

5. The vehicle can take off vertically from the docking station.
6. The vehicle can attain a maximum altitude of 50ft.
7. The vehicle can fly at a maximum speed of 15mph.
8. The vehicle can maintain flight for at least 15 minutes.
9. There is a maximum broadcast range of one mile.

Navigation Specifications:

10. Can use GPS coordinates to determine the vehicle position and heading.
11. Can regularly report the vehicles position to the computer terminal.
12. Can correct its attitude when forces obstruct the vehicles orientation.

Video Specifications:

13. B/W video.
14. 10-15 fps
15. Resolution at least 352x240

Application Specifications:

16. Has an intuitive, user-friendly GUI that requires very little training to properly use.
17. Allows the user to draw a route by dragging line segments, clicking points, or by drawing free-hand with the mouse and having the program determine the number of points along the path.
18. Allows the user to redefine the route mid-flight or issue commands to the vehicle that changes its behavior.
19. Display visuals on a screen from the vehicle mounted camera.
20. Save flight routes for future use.
21. Limits the flight duration to the maximum battery potential of the drone.

Docking Station Specifications:

22. Solar panels mounted around docking station to provide energy efficient power supply.
23. Store solar power energy so that battery can be recharged.
24. Have alternate power supply in case solar panel cannot fully recharge battery.

25. Positive and negative terminals on parabolic bowl shape where vehicle lands to allow onboard battery to recharge hands free.

2.4: Future Objectives

The great thing about this project idea is that once the drone is up and flying and able to send a live video feed to a computer monitor the possibilities of what else could be implemented is virtually endless. Being a security drone, one could add any security feature a person could think of to this vehicle. With a live video feed steaming to a computer for security personnel to view, an object detection system could be added to the drone. If a person were to walk into the camera frame a red box could be shown on the computer screen let the security personnel know that person is there. The security personnel could then have the option to have the drone follow that person if they are suspected of suspicious activity. The drone would have the ability to lock onto the person's location and the route for the vehicle would then be determined by the path the person is walking and would fly at the same pace as the person walking.

Another feature that could eventually be added to the drone to improve it would be lights and speakers to alert and notify people. Say a person was caught by the drone's camera entering a restricted area; the speaker system on the drone could alert the person by saying "You are entering a restricted area" and flash lights to warn the person. Another way this could be used would be for emergencies. If there was an emergency on the facility, the drone could fly around flashing warning lights while sounding an alarm or warn people about the type of emergency at hand. To further the idea of speakers on the drone, the security personnel could have the ability to talk with people the drone locates through a speaker system.

In high security areas key cards are being used more and more often to give people clearance to certain areas. On the drone a scanner could be added so that the drone can scan the barcodes on ID cards for the area they are in. If need be the security personnel could instruct the drone to approach a person and through the speaker system tell them they need to present identification to be scanned. The drone could then scan the person id and on the security personnel's computer a short bio of the person would pop up.

For very large areas that needed to be patrolled and monitored, multiple AFSD's could be used to work together to properly monitor an area. The ability to have multiple drones fly the same path would be possible sharing a docking station. One drone could be charging while another is flying the route. Right before the first drone returns to the docking station to recharge its battery the other drone would take off and fly the route. This way at no time would the area being monitored not have a drone flying around.

The possibilities for future development of the AFSD are endless. There is such a large variety of things that could be add-ons to the design. The drone could be custom designed based on the need of the client and have a variety of different features depending on the clients’ wants and needs.

3.1: Similar Products

When designing the AFSD two similar products were used as the basis for the AFSD design. These two products were the ‘Spherical Flying Machine’ and the ‘CoaX Helicopter.’ While the shape and general features of these other projects were mimicked in the design of the AFSD, the hardware, software, and function of the AFSD was different from these other products. One of the major differences between the AFSD and the other two products is that the AFSD does not use a remote control. The lack of a remote control made the design of the AFSD very different in the end from that of the other two products. These products also had very limited documentation about them available to refer to so other than the shape very little was mimicked in the design.

3.1.1: CoaX Helicopter

The ‘CoaX Helicopter’ appears to be the product that is available for consumer use as an RC helicopter. Although the outer design of this helicopter is not similar to the AFSD being designed, it did implement a camera and wireless connection from the helicopter. The CoaX, like the AFSD design, is small and light weight. The helicopter has a span of 340.8mm, height of 168.4mm, and width of 165.1mm with a weight of 340 grams. The small size and light weight makes it very easy for the helicopter to fly using a smaller motor. The CoaX system was built around the CoaX board which connects all the different electrical components together. The helicopter can be equipped with two types of Wi-Fi modules, external (802.11n) or internal (802.11g) Wi-Fi module. The external Wi-Fi module offers high power and longer range compared to the internal Wi-Fi module and can stream high resolution images at a high frame rate. The external Wi-Fi must however be connected through a router to transmit the images. The internal Wi-Fi module on the other hand requires an external dipole antenna on the helicopter. This module is small so allows for space to be saved but as a result does not deliver a very high data transfer rate.

Two different types of cameras can be connected to the helicopter to send images back to a computer, a Standard USB 2.0 color webcam or a Research grade camera (b/w). For both of the cameras a standard resolution of 640x480 is used and they both have a frame rate of 15 fps. Whichever camera is selected to be used is connected to a USB connector the difference is that the webcam is directly mounted on the USB connector while the research grade camera needs a cable to connect to the USB connector. For the data transfer two different types of wireless communication modules can be equipped onto the helicopter. The first is a Bluetooth module 2.o which offers easy wireless connectivity to any PC, laptop, or Smartphone. The down side to using this type of wireless module is that it can only transfer data over a few meters but has a UART port speed of up to 921.6 kbps. The second wireless communication module offered is the ZigBee module. The benefit of using the Zigbee module instead of Bluetooth is that data can be transferred over a much larger range, up to 400 feet; however with the increase in distance comes the consequence which is that the Zigbee has a lower data transfer rate. The maximum data rate of the Zigbee is 250 kbps which is much less than the Bluetooth.

The physical design of the CoaX helicopter is very different than the design of the AFSD. There is no frame surrounding the helicopter so the propellers are out in the open. The helicopter uses and upper and a lower rotor with a stabilizing bar on top of them. Below the two rotors is a small control surface with a servo motor to keep the helicopter from spinning in circles. Underneath that is a platform where all the hardware, camera, and modules are located. The camera can be positioned in a few different ways allowing horizontal views or downward views. On the bottom of the platform is where the battery is attached to the helicopter. There are four 2mm legs that are attached to the bottom of the platform that allow for the helicopter to land and stay upright.

The battery that is used to power the CoaX helicopter is a Lithium Polymer batter with 11.1 V and 1350 mAh. A charger comes with the helicopter to recharge the battery when needed. The remote control used to pilot the helicopter is a 4 channel, 2.4 GHz remote control unit. Each time the helicopter is turned on the remote must be synced in with the helicopter receiver. The remote control allows the user to control a variety of things on the helicopter. The remote control has a kill switch on it to stop the motors on the helicopter. The user has the ability to use the switches to control the height and move the helicopter up and down. On the remote control there are different sticks to control the height, yaw rate, and roll angle. The last feature on the remote control is an indicator that tells the user the battery life on the helicopter by using different color lights.

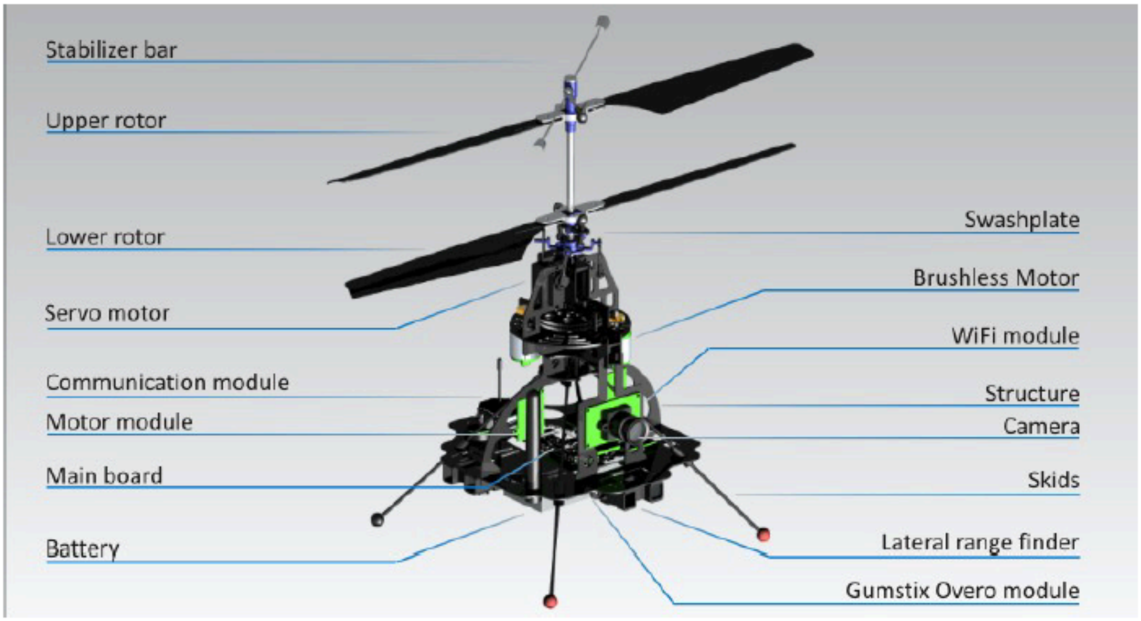


Figure 3.1.1: CoaX Helicopter

3.1.2: Spherical Flying Machine

The spherical flying machine was a prototype developed by the research department at the Japanese Ministry of Defense. The AFSD structure was designed to mimic the design of the spherical flying machine. The spherical frame acts as a shell to all of the control surfaces, motor, and propeller which are located inside of it. The vehicle has three gyro sensors located on it to help stabilize the vehicle. The stability system of the spherical flying machine is very good in that external forces can act on the vehicle shifting the vehicles orientation. Through the use of the gyros sending data saying that the attitude of the vehicle has changed, the control surfaces adjust to stabilize the vehicle. The round design of the vehicle allows for it to land in all types of surfaces and move along the ground. With the help of the gyro sensors and the stabilization the vehicle turns upright when on the ground so that it can take off vertically when the motor is started.

In horizontal flight the propeller provides a propulsion force, while the wings or flaps around the bottom of the frame provide lift. While the propeller rotates it creates buoyancy in the vehicle allowing the control surfaces to provide attitude control. There are a total of 8 control surfaces on the vehicle, 4 upper and 4 lower. The different levels of control surfaces, upper and lower, make it easier for the spherical flying machine to adjust its yaw, pitch, and roll. All the control surfaces are located under the propeller. By having the control surfaces located below the propeller the control surfaces can use the downward thrust from the propeller to their advantage. With the simple rotation of one of the control surfaces the drag created by this shift helps to re-stabilize the spherical flying machine to its original attitude.

Figure 3.1.2: Spherical Flying Machine

The vehicle is controlled by a remote control which allows the user to control the vehicles attitudes through the use of joy sticks. The spherical flying machine has the ability to hover for up to 8 minutes at a time before the battery has to be recharged and has a range of speeds for 0 km/h when it is hovering to up to 60 km/h. The flying machine weighs 350 g and is 42 cm in diameter. The vehicle was design by the Japanese Ministry of Defense in hopes that it would be able to reach places that are difficult for humans to get to or might be dangerous. The hopes are that it is able to help with rescue missions.

3.2: Objective Realization

As per the initial ideals of the project's abilities, a flight time of 15 minutes per run has been deemed the original goal. This was specified as such for the reasoning that it seemed as a reasonable goal and an acceptable amount of time to complete a worthwhile security perimeter. In order to identify if this goal is attainable as far as theoretical goals, the power consumption of each of the individual components on board the UAV. This information was gathered from various manufacturers' data sheets. However, a single current and voltage was not always listed. Usually, there was a range of currents and voltages were documented. This range of information was provided to record the upper and lower limit consumption based on how hard the components may be running. In order to account for the variations, for nearly all the components, an absolute upper limit was recorded for each component and place in the figure below.

Figure 3.2.1: Power Consumption

Component	Quantity	Current (A)	Voltage (V)	Power Consumption (W)
Camera & Transmitter	1	Separate Battery	Separate Battery	Separate Battery
GPS	1	0.05	5	0.25
MCU	1	0.000001	3.3	0.00
Servos	8	0.2	5	8
IMU	1	0.1	12	1.2
Propeller Motor	1	8	8	64
Transceivers	1	0.22	3.30	0.71
Total Power Consumption		8.57 Amps		80.16 Watts

The figure of interest here is the total amount of Amps drawn from the UAV. From the figure, this has been determined to be 8.57 Amps. The next step in this process is to compare this number to the battery which is powering the UAV.

The battery's charge storage is referred to as its capacitance. The capacitance is the amount of charge the battery can store. The Lithium polymer battery chosen has a capacitance of 2000mAh. This means that this specific battery can provide 2 Amps for one hour before becoming completely depleted. Therefore, an easy conversion can take place from capacitance and load applied to total theoretical flight time. If the power system was pulling 8.57 Amps, a simple division of the battery's 2 Amps provided by the 8.57 Amps, the ratio found would be equal 0.233. When this ratio is converted into minutes of flight time, the conversion is to multiply the ratio by 60 minutes and see that the UAV should be able to fly for 14 minutes.

There are a few flaws in this assumption however. The first is the idea of Peurket's Law. This is an effect that reduces the affective capacitance and time the batter can power the load based on the batteries discharge rate. However, based on the fact that the battery is a lithium battery the effect of Peurket's Law is reduced. On that same note, the Law states that the higher the current, the higher the effect of the reduction in apparent capacity. Since the current draw has been determined to be 'low' at around 8 Amps, the effect of Peurket's Law has been ignored for the time being. After testing, the effect may be readdresses if an issue arises for such attention to be made.

The second flaw is the fact that any battery cannot be completely discharged. If fully discharge is done, irreversible damage can take place. As a convention, a max discharge of 60% should not be exceeded. Therefore, the amount of capacity that is available for the user is 2Ah * 60% which is 1.2Ah effectively. This gives a much more accurate assumption on flight time of the UAV. The calculation would include 1.2 Amps divided by 8.57 Amps, which gives a ratio of 0.14. When this ratio is converted into minutes of flight time, the first step is to multiply the ratio by 60 minutes and see that the UAV should be able to fly for 8.4 minutes. Although this is not the 15 minutes initially desired, it is what seems to be the most logical combination of weight and power to be attained. If a larger battery was chosen, more weight would be installed and then more current would need to be drawn from the on board electronics and no increase in flight time would ensue.

The theoretical application of the power provided by the by the Lithium ion battery is believed to be slightly less desirable than that of the original goals. With the original goal of 15 minutes seeming out of reach, the only thing that is left to attempt is slight modifications on the drag from the control surfaces during flight as well as attempting to find any means of reducing the total weight of the system itself.

3.3: Future Implementation

The future of the charging station exists as a lighter and more compact version of itself. The idea is to increase the mobility in order to make the UAV more convenient to use. With increased mobility, the UAV becomes more available for a broader audience. Slight variations of this, a more mobile security UAV, opens doors for use in the public sector, governmental security purposes, or even for military parameter surveillance.

In order to accomplish a lighter version of the UAV, there must first be variations to the charging station. One proposed way to alter the charging station to adjust more optimum portability is to lower the overall weight of the station to allow for a ‘backpack’ type design to be feasible. This would basically mimic the original power distribution system, but with a few minor adjustments. The heaviest components are the universal battery, and the two solar panels, which in total account for nearly 70% of the total weight of the station.

In order to adjust for a smaller universal battery there must be an increase in average daily power to the battery from the solar panels. The reason such a large battery was chosen was to allow the UAV system to run for a minimum of three hours with no solar contribution. However, there is currently new technology being innovated that greatly outperforms the current solar efficiency rating of polycrystalline panels.

This technology is referred to as Multi-Junction Solar cells and they currently outperform polycrystalline cells by over 20%. With increased efficiency power streams to the universal battery at varying light conditions at high efficiency which allows for a smaller battery. The current battery weighs in at 25lbs but with this new technology, estimates show that a less massive holding battery is within reach. A 9lb 12 Ah Universal Battery would theoretically suffice if a large power supply was installed.

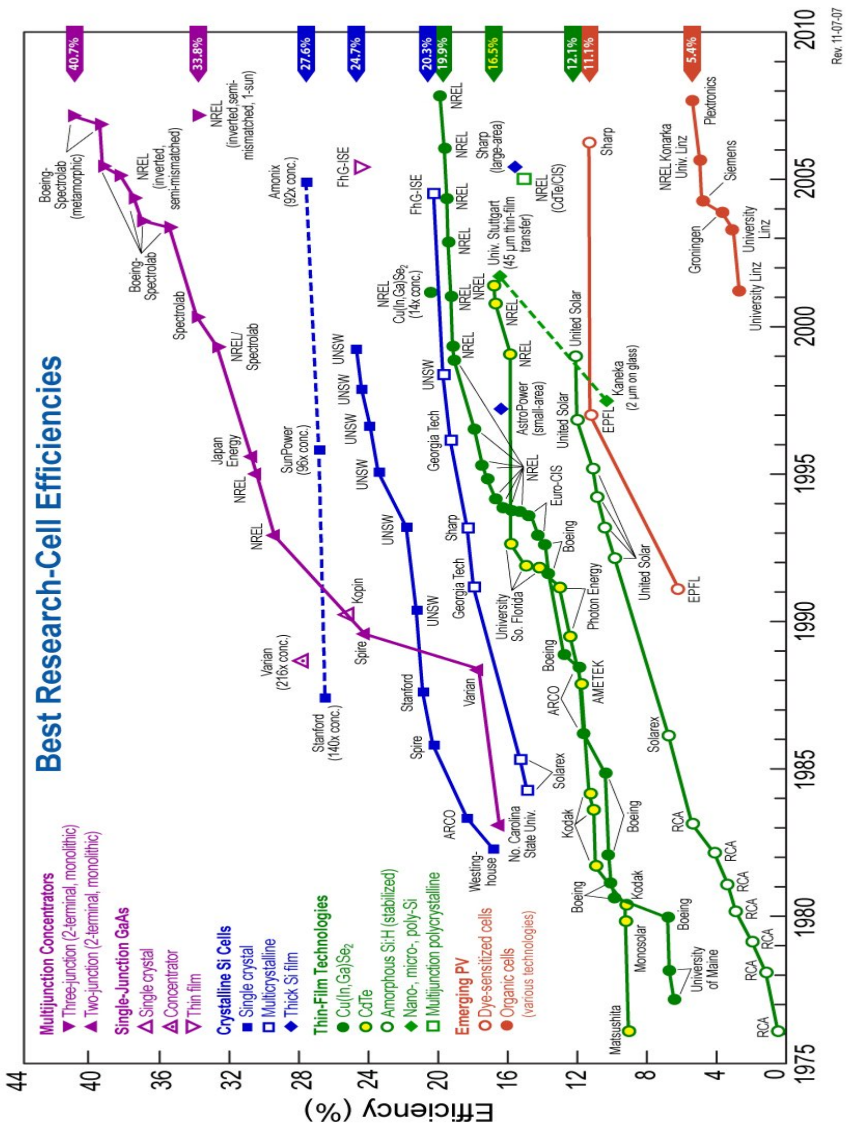


Figure 3.3.1: Cell Efficiency

In the figure above, the Multijunction cell technology is far outperforming the rest with efficiency ranging over 40%. The future mass production and popularity for increased efficiency will eventually make these expensive cells a possible future implementation in the UAV.

In the current charging station design it calls for a laptop to control the UAV. With the increase in tablet technology it would only makes sense for future implementation to include some type of wireless tablet communication to control the UAV. If the vehicle was in fact controlled by a tablet it would drastically reduce the weight and power needed for the system. The laptop of choice pulls 3.42 amps. When compared to a basic tablet at 2.1 amps, the future implementation for the controller will be nearly 40% more efficient, requiring an overall 16 less watts of power during charging.

In the future, another great innovation would be to have the charging station and UAV be able to launch itself remotely by someone in a nearby location. There would need to be three issues addressed in order to make this dream a reality. First, the Top Flap of the charging station would need to be able to open itself. It is already on a hinging mechanism. Therefore, it would be fairly simple to design some sort of mechanism to have a motor activate, upon wireless user command, and slide or push open the Top Flap. This would allow the UAV to be able fly out and dock on its own without any in person user. Secondly, there would need to be a solution to the station only being able to move by a person. If it had some was of moving on its own, then it would be one step closer of provided complete wireless portability of the system. This issue can be resolved with the addition of a second set of wheels attached and motors attached to them. Finally, there would need to be some sort of wireless communication for the UAV to the talk to the charging station and then the station relay the information to the remote location. The information relayed to the remote location needs to be all encompassing. The information transmitted at minimum needs to contain data for the control system, GPS coordinates, and video signal.

The great thing about this project idea is that once the drone is up and flying and able to send a live video feed to a computer monitor the possibilities of what else could be implemented is virtually endless. Being a security drone, one could add any security feature a person could think of to this vehicle. With a live video feed steaming to a computer for security personnel to view, an object detection system could be added to the drone. If a person were to walk into the camera frame a red box could be shown on the computer screen let the security personnel know that person is there. The security personnel could then have the option to have the drone follow that person if they are suspected of suspicious activity. The drone would have the ability to lock onto the person’s location and the route for the vehicle would then be determined by the path the person is walking and would fly at the same pace as the person walking.

Another feature that could eventually be added to the drone to improve it would be lights and speakers to alert and notify people. Say a person was caught by the Drones camera entering a restricted area, the speaker system on the drone could alert the person by saying “You are entering a restricted area” and flash lights to warn the person. Another way this could be used would be for emergencies. If there was an emergency on the facility, the drone could fly around flashing warning lights while sounding an alarm or warn people about the type of emergency at hand. To further the idea of speakers on the drone, the security personnel could have the ability to talk with people the drone locates through a speaker system.

In high security areas key cards are being used more and more often to give people clearance to certain areas. On the drone a scanner could be added so that the drone can scan the barcodes on identification cards for the area they are in. If need be the security personnel could instruct the drone to approach a person and through the speaker system tell them they need to present identification to be scanned. The drone could then scan the person id and on the security personnel’s computer a short bio of the person would pop up.

For very large areas that needed to be patrolled and monitored, multiple AFSD’s could be used to work together to properly monitor an area. The ability to have multiple drones fly the same path would be possible sharing a docking station. One drone could be charging while another is flying the route. Right before the first drone returns to the docking station to recharge its battery the other drone would take off and fly the route. This way at no time would the area being monitored not have a drone flying around the predefined loop.

The possibilities for future development of the AFSD are endless. There is such a large variety of things that could be add-ons to the design. The drone could be custom designed based on the need of the client and have a variety of different features depending on the client’s wants and needs.

4.1: Vehicle Design Architecture

The Automatic Flying Security Drone (AFSD) is an alternative security solution to stationary security cameras. Having a spherical frame with an internal propeller, the AFSD will be an incredibly stable, mobile vehicle capable of broadcasting security footage to a remote computer terminal as it traverses a user-defined route. After completing the route, the AFSD will land in a specialized charging station and charge itself before taking the route again. The design is intended to require only one person to operate in a minimal amount of time with minimal difficulty.

The spherical frame was chosen for a number of reasons. First, it ensures that the architecture has symmetry. The stability of the device is increased by having all of the control surfaces centered on the central axis. The center of gravity--containing the battery, prop motor, and the main board--can also be located on this axis so that the forces created by the control surfaces create torque around the center of the craft.

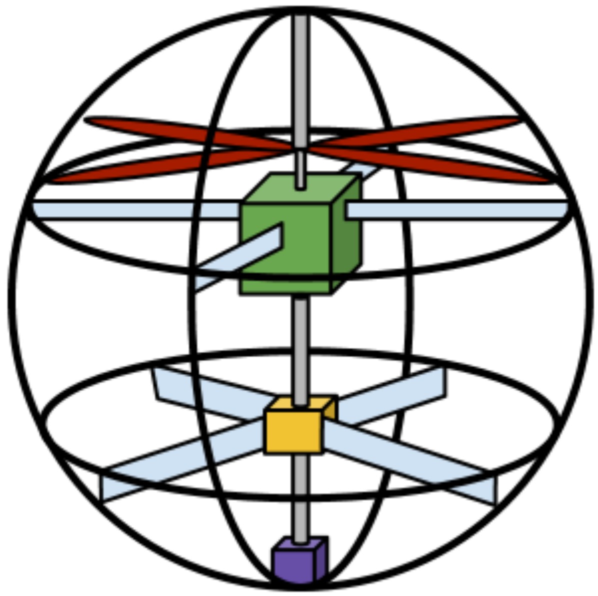


Figure 4.1.1: Simplified Diagram of the AFSD, where major component structures are color-coded

The frame, pictured in Figure 4.1.1 as the spherical network of black lines, will be a lightweight, plastic skeleton with strength sufficient to endure minor stress and impacts. The vehicle will be able to land and roll on the outside of its frame without taking damage. A stronger--likely metallic--central axis, shown in grey, will run through the center of the vehicle to increase rigidity. This axis will be hollow and will be used to run wires between each of the component blocks.

The largest of these component blocks, in green, contains a majority of the electronics. In this housing, all of the navigation and orientation-sensing equipment will keep track of where and how the vehicle is oriented in airspace. This data is processed by a central microcontroller, which decides how the mechanical portions of the vehicle should behave to promote stability and follow the loaded route. In this block is a set of four servos, which operate the four (light blue) control surfaces that protrude from the green box. The motor for driving the prop (red) is also in this block. Both of these mechanical components answer directly to the microcontroller. Finally, the battery for running all of the electronics is also within this block. With all of these components inside, the green block is easily the center of mass for the entire vehicle. This is important for maneuvering, as will be discussed later.

The lower, orange block is a set of servos for controlling the lower set of four control surfaces. Despite being further down on the central axis, these four servos are still controlled by the MCU and powered by the battery contained in the green block. Wires are run through the grey central axis to each of these servos. Both the control surfaces and the servos on the orange block are larger than their green block counterparts, due to the large amounts of force required to cause torque around the vehicle's center of mass.

The purple block on the bottom of the vehicle carries the camera that films the security footage. It also contains an ultrasonic rangefinder, for determining altitude by determining the distance to the ground, and an infrared camera, for detecting a pattern of infrared LEDs on the bottom of the charging station and guiding the vehicle in for a safe landing where it can charge itself.

Nearly all of the components mentioned will either be soldered to or otherwise connected to the vehicle main board: a PCB containing the MCU and possibly a few miscellaneous items such as amplifiers, voltage regulators, and RC components. Most of the larger components will be located off of the board and instead will be connected via plugs and pins. Power supplied to each component by the battery will also run through the board. Finally, in order to be able to reprogram the MCU during the testing phase, the PCB will also include a mini-USB interface. This USB port will only be present in the design prototype. The finished product would not have this interface port.

4.2: Charging Station Design

In any electrical project, there must be some sort of power distribution system in place. The power distribution system is what will provide power and electrical stability for one's specific needs. Usually, this is successfully completed by designing the power system with the overall idea that after production, it will be plugged into a convenient 120V AC friend; "The Wall". In recent years, however, there has been an influx in the amount of interest surrounding the ideals and possibilities of renewable energy. Simply put, renewable energy involves any method of harnessing the world around us in a way that is sustainable to the surrounding environment. Some commonly known sources include wind, hydroelectric, thermal, and solar energy.

The main goal of this charging station is to create a completely energy independent, portable, and simple design. The characteristics of portability and simplicity open the door for the possibility of future mass production purposes as well as a decrease in the amount of effort required for operation. The station will have a completely aluminum cubic frame with two wheels and retractable handle attached. This will allow for a sleek, clean looking design while keeping the station relatively light and easy to maneuver. In addition, the extendable handle and wheels will promote ease of transportation and allow nearly any operator the ability to transport the station.

There will also be two hinged flaps on the station. One flap, located at the rear, is designed for ease of accessibility for maintenance. The second flap, located at the top of the station, is designed for operational purposes. This top flap will manually be opened when it is time to launch the UAV. When the top flap has been opened, the UAV will take its predetermined path, then identify the charging station once more, and dock itself. The most beneficial and convenient addition to the station is the fact that the UAV is self-docking. This leads the UAV to be able to complete multiple and repetitive loops around its parameter without needing the assistance of human monitoring. When the necessity of human monitoring is removed, the possibilities for future implementation into the public and private sector become exponentially greater.

4.3: Navigation Subsystem Design

4.3.1: Narrative Description of Navigation and Positioning Systems

The AFSD performs all of its navigation by storing a series of GPS coordinates in its on-board memory, and travelling to each of those locations in turn. The data for each point will be stored as a structure of two floating-point numbers: one for the north-south coordinate, and one for the east-west coordinate. Since the memory requirement for these coordinates will be very small (approximately 16 bytes for each point), all of the route data will be transmitted to the vehicle and stored prior to flight. The collection of coordinates will be stored in an array structure in the order in which they must be visited. This data remains in memory until the route is changed, at which point it is deleted and replaced by the new route.

In order to travel from one point to another, the vehicle uses its on-board GPS to determine its current position, which it then compares to its desired position. By adjusting its heading, the vehicle simply attempts to minimize the difference between the two positions. However, since the primary application of the vehicle is to follow a distinct path to provide video surveillance of a specific route, the following measures will be implemented to ensure that the vehicle stays on course.

While in flight, the vehicle will consider two points (sets of coordinates) at any given time: The point it most recently visited, and the next point it must travel to in its route. These two points can be connected by a virtual line that will be referred to as the Ideal Flight Path.

Figure 4.3.1: Ideal Flight Path

While flying between two points, there will be deviations from the Ideal Flight Path that must be corrected in-flight. This distance between the Ideal Flight Path and the vehicle is the Flight Path Deviation.

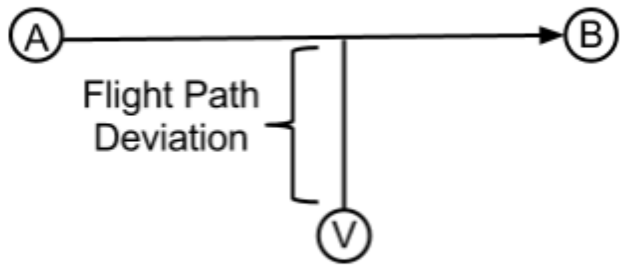


Figure 4.3.2: Flight Path Deviation

The Flight Path Deviation can be determined by the triangle altitude equation, since the three points A, B, and V form a triangle:

$$\text{attitude} = \frac{2\left(\sqrt{(\mathrm{s}-\mathrm{a})(\mathrm{s}-\mathrm{b})(\mathrm{s}-\mathrm{c})}\right)}{a}$$

where s is the semiperimeter of the triangle and a, b, and c are the lengths of the sides. Modifying this equation to suit the purposes of the above diagram, we can define the semiperimeter as

$$s = \frac{\text{dist}(A,B) + \text{dist}(A,V) + \text{dist}(B,V)}{2}$$

The Flight Path Deviation therefore is determined by the equation

$$\text{flight path deviation} = 2\left(\frac{\sqrt{\big(s-\text{dist}(A,B)\big)\big(s-\text{dist}(A,V)\big)\big(s-\text{dist}(\mathrm{B},\mathrm{V})\big)}}{\text{dist}(A,B)}\right)$$

The deviation is measured in feet, as are all the distances between the vehicle and its points. While the deviation is small, the vehicle is said to be in the Acceptable Flight Zone. While in the Acceptable Flight Zone, the vehicle is permitted to continue on its route making only minute corrections to its heading to try to approximate the ideal flight path as closely as possible.

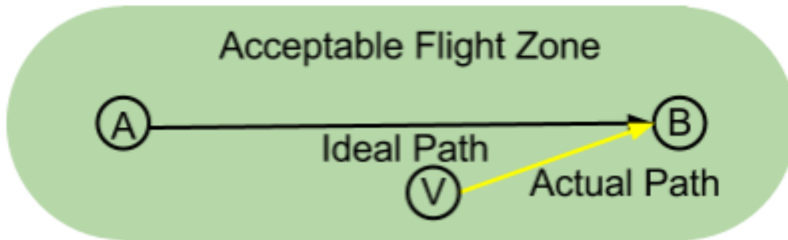


Figure 4.3.3: Acceptable Flight Zone

The Acceptable Flight Zone is the area inside a predefined Maximum Acceptable Deviation around the line and two endpoints. The Maximum Acceptable Deviation will be set by the user, but the default setting will be a 10 foot radius. When the Flight Path Deviation is greater than the Maximum Acceptable Deviation, the vehicle is no longer in the Acceptable Flight Zone and must return to it before continuing forward. This is a process called Route Recovery. A temporary point is created on the Ideal Flight Path that the vehicle travels towards until it is within the Acceptable Flight Zone again. Once within the zone, the vehicle can continue moving forward. The temp point is specifically selected such that the temporary route intersects the Ideal Flight Path perpendicularly.

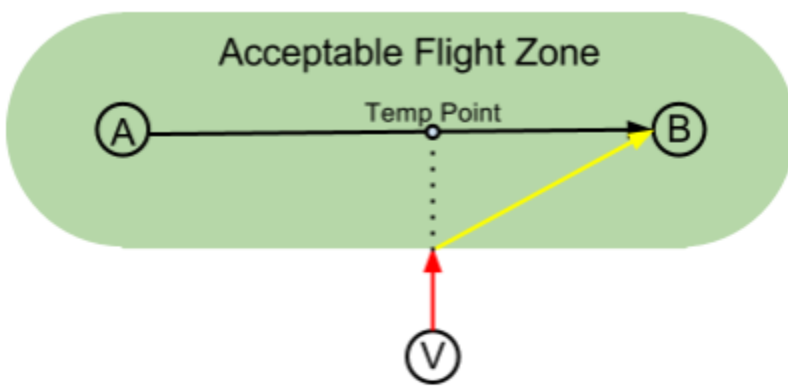


Figure 4.3.4: Route Recovery through creation of a temp point

The temp point could be determined from trigonometry, but in order to avoid having to expend memory on a math function library and to simplify the calculations, the following process will be used:

1. The line equation of the Ideal Flight Path is determined. If x is the east-west coordinate of a point, the y-coordinates of the line can be determined by the equation:

$$y = \frac{y_B - y_A}{x_B - x_A}(x - x_A) + y_A$$

2. For one-hundred x-values between x_A and x_B, the y-values are determined using the equation derived above.

3. The distance is measured between each evaluated point and the vehicle's current position.

4. Since a parallel route between the vehicle and the Ideal Flight Path is also the shortest route to the path, the distance to each point is determined until the distances stop decreasing. The point just before this occurs is a decent approximation of the ideal point mentioned prior.

During flight, the vehicle may be driven so far off-course that it is not advisable to try to return to the Acceptable Flight Zone. At this point, the vehicle will have to try to make an emergency landing. This large distance from the route is called the Extreme Off-Course Boundary. The distance of this boundary from the route may be greater or smaller depending on battery power remaining and how many times the vehicle has been driven off course already.

After crossing the Extreme Off-Course Boundary, the vehicle will slowly descend while emitting a special signal to the computer indicating where it will likely land. An infrared sensor on the bottom of the vehicle will measure distance to the ground. If the vehicle lands in an unstable surface, it can roll on its spherical frame without being damaged. Once it has landed, it will periodically broadcast its location to the controlling computer terminal so that it can be recovered.

4.3.2: Pseudocode for Navigation Logic

```
// prior to flight
if(a route is not currently loaded in memory or there is a new route) {
        download the new route from the computer terminal to the vehicle memory;
        store the p points of the route in the array routePoints[];
}

perform takeoff procedure;

// during flight
for(i from 0 to p) {
        j = i + 1;

        dist_ij =: dist(routePoints[i], routePoints[j]);
        // returns the distance between the two points

        set vehicle heading towards routePoints[j];

        while(current position is != routePoints[j] + E) {
        // E represents the maximum distance from a point before it's "visited"

        currentPos = output of GPS module;
```

```
        report currentPos to the computer terminal;
        dist_vi =: dist(currentPos, routePoints[i];
        dist_vj =: dist(currentPos, routePoints[j];

        s = (dist_ij + dist_vi + dist_vj)/2;  //semiperimeter
        h = sqrt((s-dist_ij)*(s-dist_vi)*(s-dist_vj))/dist_ij;
                // flight path deviation

        if(h > maximumAcceptableDeviation) {
                // the vehicle is too far from the route and must return to an
                // acceptable distance
                initialX = routePoints[i].x; // x-coordinate of first point
                finalX = routePoints[j].x; // x-coordinate of final point
                initialY = routePoints[i].y;
                finalY = routePoints[j].y;
                slope = (finalY - initialY)/(finalX - initialX);
                dX = (initialX + finalX)/100;

                tempPoint = routePoints[i]; // starting from the initial point
                // setting two points equal to one another means copying
                // the x and y coordinates to the other

for(x = initialX; x <= finalX; x += dX) {
                        y = slope * (x - initialX) + initialY;
                        testPoint.x = x;
                        testPoint.y = y;
                        if(dist(currentPos, testPoint) < dist(currentPos, tempPoint)) {
                                tempPoint = testPoint;
                        } else break;
}

                // after the previous for loop, we can be guaranteed that tempPoint
                // is the closest point on the ideal route to our current position

                set vehicle heading towards tempPoint;

                do {
                        currentPos = output of GPS module;
                        report currentPos to the computer terminal;
                        dist_vi =: dist(currentPos, routePoints[i];
                        dist_vj =: dist(currentPos, routePoints[j];

                        s = (dist_ij + dist_vi + dist_vj)/2;
                        h = sqrt((s-dist_ij)*(s-dist_vi)*(s-dist_vj))/dist_ij;
                } while(h > maximumAllowableDeviation AND h < extremeOCBoundary);

        if(h >= extremeOCBoundary) {
                initiate emergency landing procedure;
        } else {
                set vehicle heading towards routePoints[j];
        }
}

        // whether the above if statement was found true or not, the vehicle is
        // assured to be on course to the next point on its route
}
}

// all points along the route have been visited
perform landing procedure;
```

This code is executed repeatedly so long as the vehicle is in operation. There will also be numerous checks for vehicle orientation to ensure stability during flight, but these checks have been omitted from this pseudocode to improve clarity and avoid redundancy.

4.3.3: Navigation Block Diagrams

The simplest expression of the vehicle's navigation can be expressed by a linear feedback control system, where the input is the desired position, the forward path is an adjustment of the various control surfaces that direct airflow from the prop to create movement, and the output is the updated position via the GPS. The backwards loop feeds this new position data back so that it may be compared to the desired position and further adjustments can be made. Information from the IMU is also taken into consideration to ensure that the vehicle will be stable in its orientation.

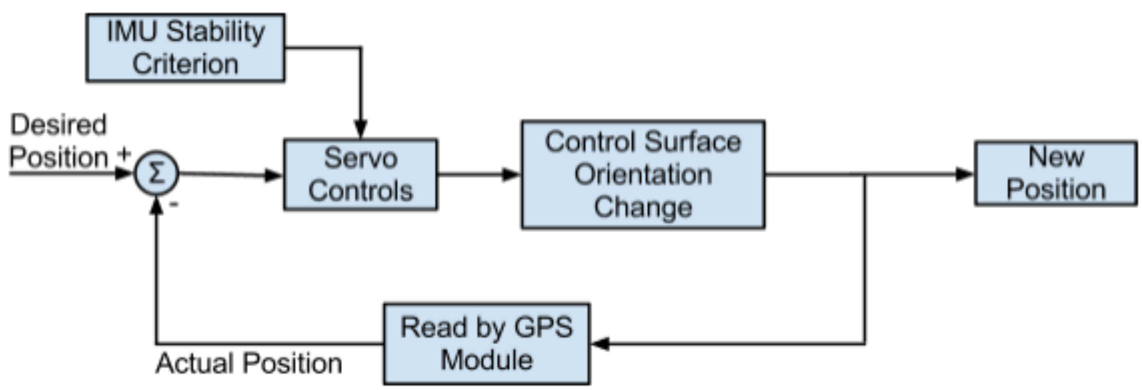


Figure 4.3.5: Feedback loop showing recursive relationship between desired position and actual position

Considering that the desired position is contained within the memory of the microcontroller, it's also possible to create an input-output block diagram to show the data flow between the various devices and the microcontroller. The IMU and GPS feed data to the MCU, which processes the data and outputs it to the servo controls and, by extension, to the control surfaces. The routing antenna is also included, to show the inflow of route data and the outflow of position data.

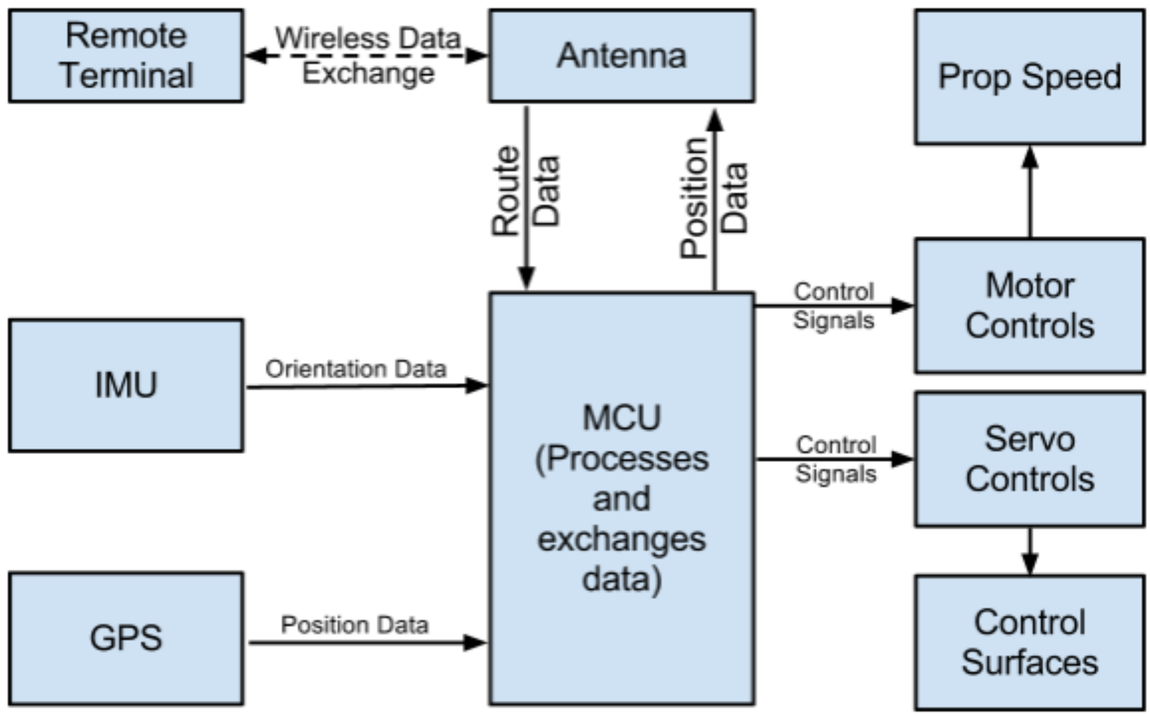


Figure 4.3.6: Data Flow Diagram demonstrating the relationships between each of the components in the navigation structure.

4.3.4: Altitude and Motor Control

Because there are no reasonably priced altimeters precise enough to determine altitude on the level of feet, altitude will be determined using an ultrasonic rangefinder aimed at the ground. This is a reliable way of determining the vehicle's altitude over flat ground, but in the case that the distance to the ground changes--as the vehicle flies over a tree, for example--the distance results could end up very noisy. It is because of this that the new results for altitude will be run through a weighted function so that big changes in distance to the ground do not cause the vehicle to rapidly ascend or descend in order to keep the desired altitude.

4.3.5: Landing Procedure

Once the vehicle has visited every point in memory, it should be above the charging station, and should initiate landing procedures. The vehicle begins by sampling its position several times and averaging its results to obtain a more precise indication of its position. Minor adjustments are made while hovering to approximate the position of the charging station as closely as possible. After a few checks to determine if the vehicle is in a satisfactory position, an infrared sensor on the bottom of the craft will switch on, and a subroutine will be executed. The IR sensor will look for a specific pattern of infrared LEDs arranged on the bottom of the charging station.

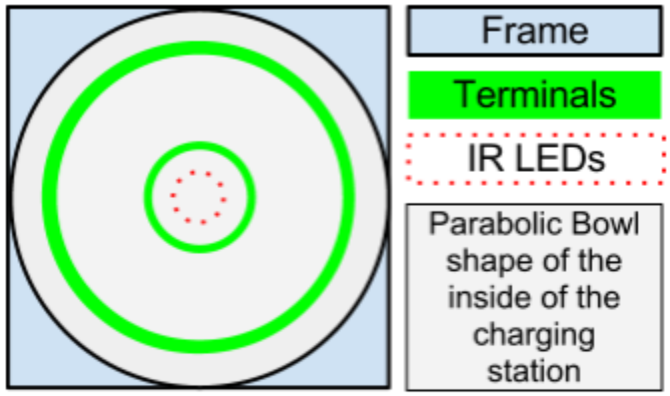


Figure 4.3.7: Graphical Representation of the LED pattern on the bottom of the charging station

Shortly before and during descent, the vehicle will attempt to center the LED pattern in the IR sensor's viewport as part of the landing subroutine. This assures that the landing procedure does not have to rely solely on the GPS device, which won't be precise enough to guide the vehicle into the landing area.

4.3.6: Design Specifications

Because the algorithms the navigation system will be built on rely so much on knowing the current position of the device, it is important that the GPS updates quickly so that precious time is not spent waiting on position data. Most GPS modules update their position at 1Hz, which means in the worst case scenario--the MCU requires position data from the GPS just after it updates--the program could end up idle for nearly a full second before the data comes in. Given how often the position data must be checked, this is an undesirable occurrence that could result in a very sluggish navigation system. For this reason, it is of utmost importance that the GPS module update considerably faster. Fortunately, there are several modules capable of producing data much more quickly, upwards of 5-10 times per second.

Of secondary importance is the accuracy and resolution of the module. Luckily, a minor lack of accuracy can be accounted for by taking a number of position measurements in sequence and attempting to eliminate outliers, but this would once again result in undesirable slowdowns. What cannot be corrected for, however, is resolution. Figure 4.3.8 shows the relationship between resolution in GPS coordinates and resolution in real distance:

Figure 4.3.8: GPS Resolution vs. Real Distance Resolution	
GPS Resolution	**Real Distance Resolution**
0.1 degrees	6.50 miles
0.01 degrees	0.65 miles
0.001 degrees	343 feet
0.0001 degrees	34.3 feet
0.00001 degrees	3.43 feet

It is worth noting that, in order to function on the order of a few feet, the GPS must be capable of reporting it position to the order of 1/100,000th of a degree. Anything less precise will be unsuitable for the purpose of this design.

Signal acquisition time is slightly less important to the design. Most GPS modules have a long initial connection time and then a shorter reconnection time should the power or signal be lost temporarily. The vehicle can stay grounded after being powered on and wait for a GPS signal as part of the boot sequence. If the signal is ever lost while in mid-flight, however, it is important that it be reacquired as quickly as possible. This is achieved by having a high number of acquisition channels that can search at a variety of frequencies and phases for the GPS signals.

4.4: Vehicle Control System Design

4.4.1: Control Surface Configuration

As with any aircraft, there are three axes of rotation to account for: pitch, roll, and yaw. Control over the AFSD's orientation can be achieved with the help of eight control surfaces located below the prop and within the vehicle's frame. Four of the surfaces will be located just below the prop and level with the vehicle's center of gravity. These surfaces will control the yaw of the vehicle. The other four will be located near the bottom of the craft and will control pitch and roll.

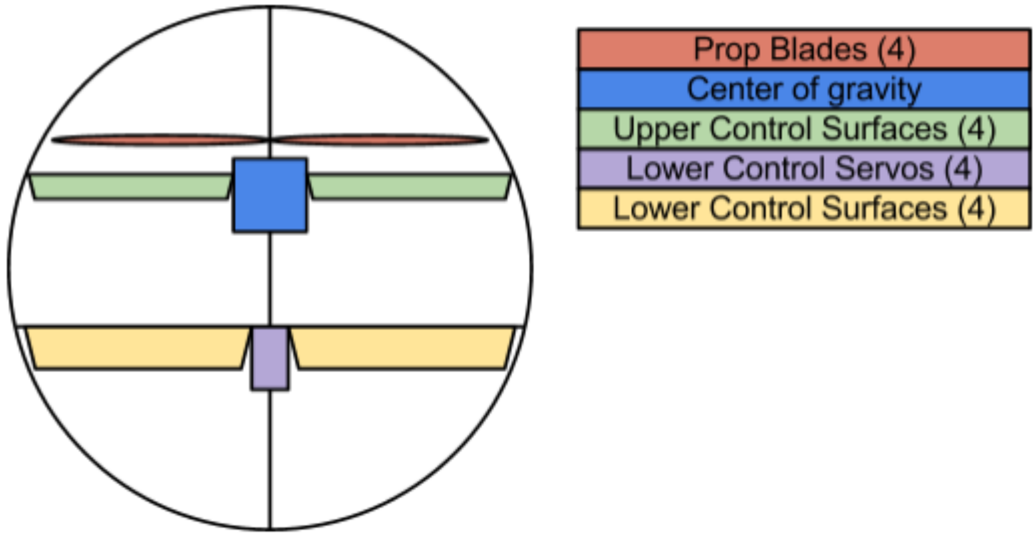


Figure 4.4.1: Simplified graphical representation of the vehicle's control systems.

The blue block representing the center of gravity also contains the 4 servos for controlling the upper control systems and the motor for driving the prop.

Figure 4.4.1 makes a few departures from the actual design in the name of simplicity and clarity. First, the components of the control system that would protrude out of or into the paper have been removed. This includes two prop blades, two of the upper control surfaces, and two of the lower control surfaces. Second, the upper and lower control surfaces are not parallel, but instead are offset from each other by 45 degrees (i.e.: the upper four form a + while the lower four form an x). However, despite being a drastic simplification of the overall control system, it illustrates a very important point that will be explored in the following sections: The vehicle control system can be divided into distinct component subsystems.

4.4.1.a: Motor and Prop Control Subsystem

The 4-bladed prop of the vehicle will be driven by a brushless DC motor. Motors of this type are usually controlled by one of two methods: Either a direct voltage/RPM ratio, or a PWM control signal. Either of these options requires only a slight modification in design, as shown in Figure 4.4.2. If the motor is controlled by voltage, a digital potentiometer, controlled by the microcontroller, will be used to control the voltage across the motor and by extension the speed, as shown in Figure 4.4.2.a. If PWM is required, one of the PWM outputs of the microcontroller can be amplified to provide the required signal. In this case, the voltage necessary would be contributed directly to the motor through a resistor network.

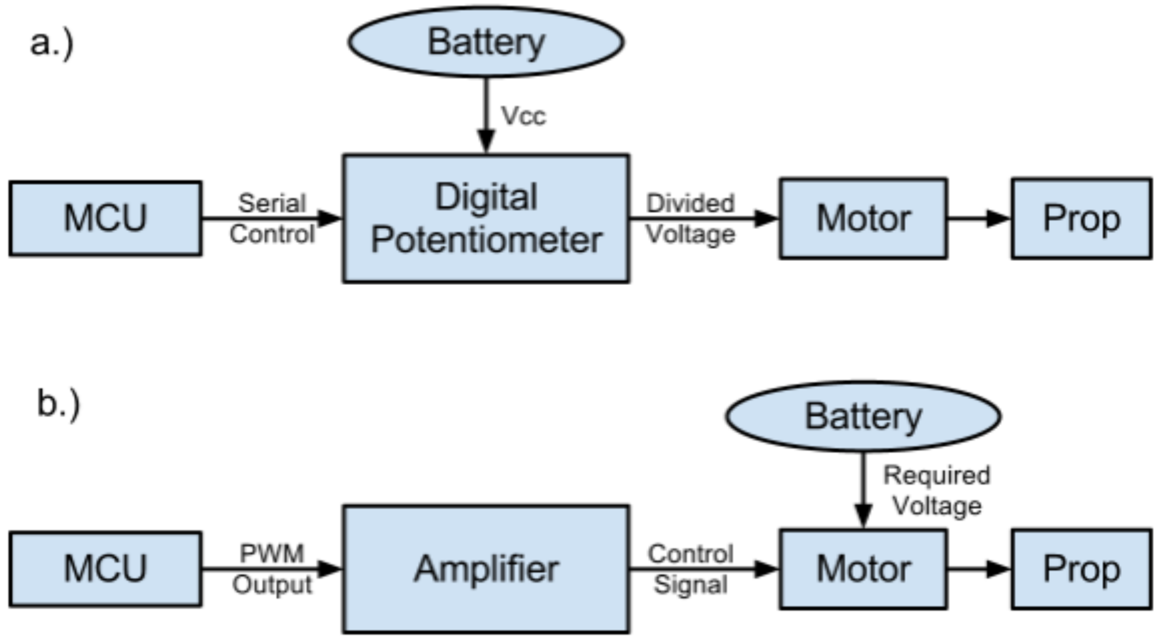


Figure 4.4.2: Block Diagrams of two possible motor control systems

Regardless of the final design, however, there are five basic operation modes for the motor. The motor begins and finishes in the off mode, where no power is supplied to the motor at all. The take-off mode is a quick ascent, likely at maximum RPMs, until the vehicle reaches the desired height. The hover mode supplies enough thrust to keep the vehicle at a somewhat-constant altitude and is the mode the vehicle will be in a majority of the time it's in operation. The recovery mode is initiated when the vehicle drops below the desired altitude during flight, and the motor speed is adjusted to just above the hover mode speed. The landing mode is a slow descent at just under hover mode speed, and is used to land the vehicle either at the end of its route or in case of emergencies. The exact speeds the motor needs to turn for each of these modes will be determined with device testing.

4.4.1.b: Upper Control Servo Subsystem

The four upper control surfaces control the yaw of the vehicle. This is achieved by rotating all four of the surfaces the same direction, either clockwise or counter-clockwise. This causes air to be forced to either side of the surface. Since the surfaces are on level with the vehicle's center of gravity, the moment arm of the force is along the control surface and the axis of rotation vertical through the center of mass. This causes the vehicle to rotate in the opposite direction from the airflow, as shown in Figure 4.4.3:

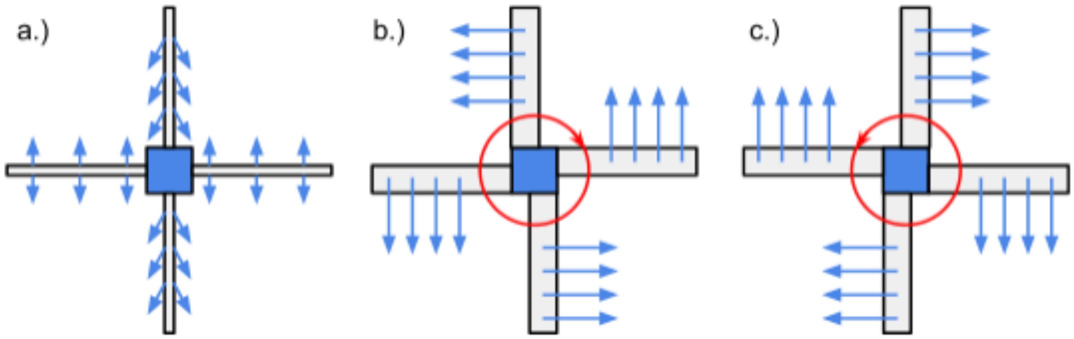


Figure 4.4.3: A top down view of the center of mass and the four upper control surfaces

The control surfaces are shown in gray, the center of mass in dark blue, the direction of diverted airflow in light blue, and the rotational force in red.

As shown above there are three possible configurations of the upper control surfaces. The rest position, shown in Figure 4.4.2.a, is the “no-adjustment” configuration. While illustrated here to appear strictly vertical, this configuration will be, in practice, slightly off-center. This is to force a small amount of airflow to one side; enough to counter the torque of the spinning prop motor. If left unchecked, the motor’s rotational force would cause the vehicle to spin out of control. In order to adjust the yaw of the vehicle, the four servos are turned counter-clockwise from the rest position to create a clockwise rotation (as in 4.4.2.b), or vice versa (as in 4.4.3.c). Because the force required to turn the vehicle around its center of mass is relatively small and is being distributed across 4 control surfaces, a small change in the angle of the surfaces could cause a drastic change in the craft’s orientation. Therefore, the yaw control surfaces will be relatively small to avoid accidental overcompensation.

Any other configurations--specifically, having the four surfaces be any mix-match combination of rest, clockwise, or counterclockwise--would likely compromise stability or at the very least prove useless in practice. Therefore, all the surfaces’ movements will be implicitly synchronized. By sharing one control signal across the four upper servos, it is not only possible to move all four upper surfaces in unison, but also to only expend one PWM channel in doing so.

4.4.1.c: Lower Control Servo Subsystem

The four lower control surfaces work in pairs to control pitch and roll. These surfaces are located near the bottom of the frame so that the force they exert causes a rotation around the vehicle’s center of gravity as shown in Figure 4.4.4. Because they are located further from the prop and are required to exert more force to change the vehicle's orientation, the lower control surfaces are larger than the upper surfaces. The servos must also work in two pairs rather than four all-together to control pitch and roll, so the torque provided by these pairs of servos must be high enough to handle the various forces imposed on them.

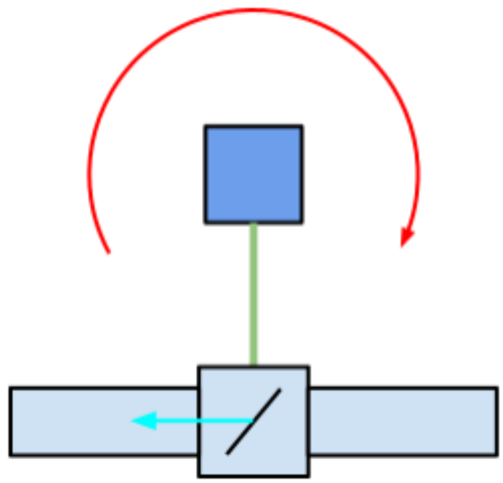


Figure 4.4.4: Simplified diagram of lower servo manipulation and the resultant torque around the center of mass.

In Figure 4.4.4, airflow directed by the lower surface results in a net horizontal force at the surface. Note that there is also a surface on the opposite side deflecting airflow in the same direction. This force, represented by the light-blue arrow, is directed along the moment arm, represented by the green line, resulting in a torque force and rotation, represented by the red arc, around the center of mass, represented by the dark-blue box. A rotation in the opposite direction can be achieved by rotating the servos in the opposite direction.

Unfortunately, unlike the upper control servos, the lower control servos cannot be synchronized with the same control signal. Not only do the pitch and roll pairs have to work separately, but each member of each pair have to rotate in opposite directions in order to deflect air flow in the same direction. Therefore, it is imperative that each of the lower control servos have their own control signal.

4.4.2: Movement in Flight

Beyond changing the vehicle's orientation in mid-flight, it is necessary for these control systems to be able to move the craft on a horizontal plane. By tilting the craft using the lower control surfaces, the airflow of the prop is angled slightly to provide a horizontal component of thrust. This is illustrated more clearly in Figure 4.4.5.

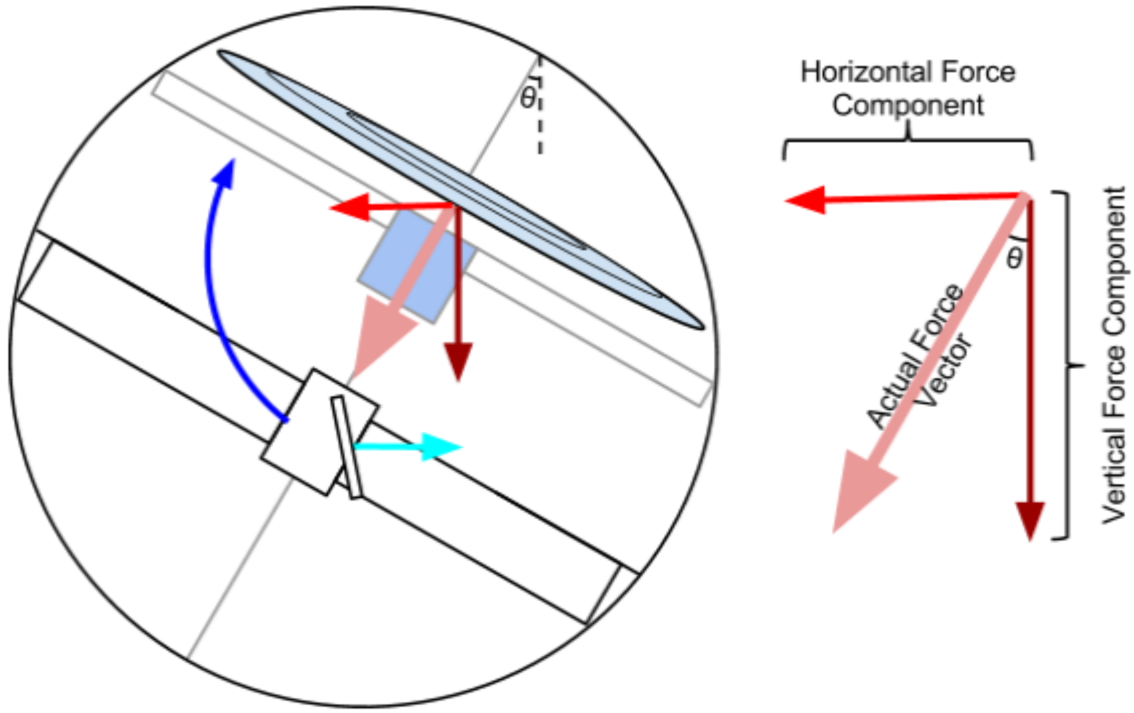


Figure 4.4.5: Creation of a horizontal thrust vector

By using a pair of the lower control servos, airflow from the prop is used to create a net horizontal force which is represented by the light blue arrow. This creates a torque, the dark blue arrow, which tilts the vehicle by an angle θ. This changes the direction of thrust of the prop by θ. The thrust vector can then be envisioned as the combination of two component vectors. The horizontal component of the thrust moves the vehicle forward while the vertical component keeps the vehicle airborne.

If the prop's overall thrust can be specified by a magnitude $|T|$, the horizontal component can be expressed as $|T|\cos\theta$ and the vertical component is $|T|\sin\theta$. Since the vertical component is less than or equal to $|T|$, the thrust must increase when the vehicle is moving so that the vertical component is enough force to keep the vehicle afloat.

When the vehicle has a new point to travel to, it will orientate itself using the upper control surfaces so that one pair of the lower control surfaces are perpendicular to the direction of travel. The perpendicular pair is then used to control the forward movement of the vehicle while the pair parallel to the path is used for course correction and stabilization. In order to slow and stop, the surfaces return to the rest position a few feet short of the desired position. If the vehicle is not stopping fast enough, the surfaces can tilt the vehicle in the direction opposite the motion to direct a small amount of thrust against the vehicle's current momentum.

The onboard IMU helps to assure the vehicle does not reach an unstable orientation. If the vehicle starts to tilt too much in any one direction and is in danger of capsizing, the MCU will initiate an emergency

4.4.3: Design Specifications

The prop is the most criteria-restricted component of the control system: it must fit within the upper-half of the frame, which is a 1-foot diameter sphere. Therefore, the propeller must be significantly less than 12-inches in diameter. The motor must be selected around this restriction on the prop, being able to spin it fast enough to provide lift but also being energy-efficient enough to run off of the vehicle's battery for a significant amount of time. The control surfaces must be large enough to exert control over the vehicle, but shouldn't be unnecessarily heavy. The servos controlling the surfaces should be light enough to not contribute a significant amount to the weight of the vehicle, but should be capable of enough torque to handle the force of the thrust against the surface.

4.5: Vehicle Frame Design

While deciding on the design of the frame for the vehicle and what would best fit the projects goals and objectives a few different factors were considered: shape, weight, and component placement. The first major design requirement was the shape of the frame that was going to best suit the desired objective. The frame had to be aerodynamic while at the same time allow for collision recovery. The design plan was to make the drone so that it could bump into obstacles or be affected by external forces and be able to recover and stabilize.

Traditional plane and helicopter frame designs had to be thrown out. Those traditional designs called for the propeller of the plane or helicopter to be out in the open, making it so that if the vehicle hit a wall the wall would interfere with the spin of the propeller causing the vehicle to crash. Knowing this much research was devoted to finding a frame that would surround the entire vehicle so that when the vehicle were to hit a wall it would be able to recover from the collision and continue to fly. Knowing that the frame was going to surround the vehicle and all the components were going to be inside the frame the next question at hand was to determine the shape of the frame.

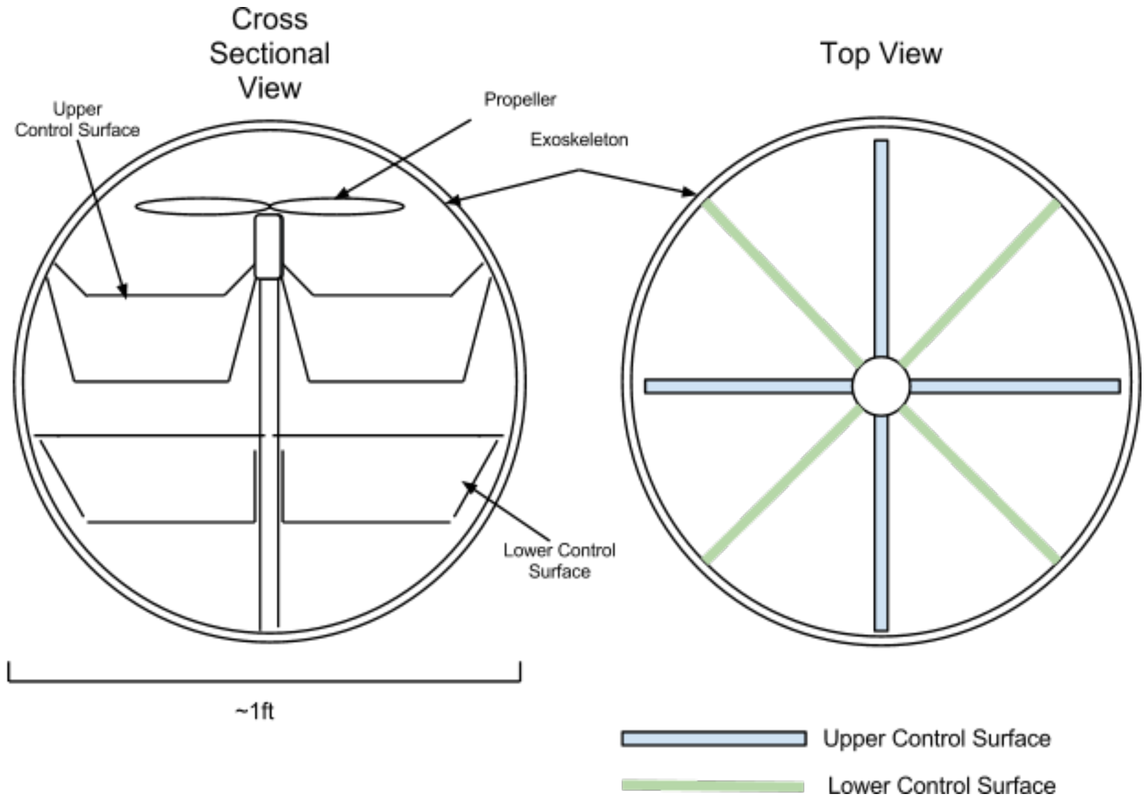


Figure 4.5.1: Cross Section View and Top view of the Initial Frame Design

With the frame surrounding the outside of the vehicle the question of the physical shape of the frame became an issue. Such a large frame on the outside meant lots drag from the vehicle thus making it not very aerodynamic. At the same time the issue how to make the shape of the frame able to land became an issue. To make the frame in a square shape would allow for easy landing on one of the flat sides, however the large amounts of flat surfaces from a square create to large of a drag. Through much research on the issue a design created by the Japanese Ministry of Defense proved to be a perfect design for the vehicle to meet the design needs of the project. The shape of the frame would be spherical.

Figure 4.5.2: Assembled Vehicle Frame

The spherical frame design surround the vehicle solved the initial problems. The rounded shape of the frame helped to make the vehicle more aerodynamic. Although a rounded shape would be more difficult to safely land a spherical frame best suited the vehicle requirements. The docking station became designated to solve the difficult of securely landing the vehicle. The exterior part of the frame consisted of 8 different semicircle pieces that would be clip into a 2 different circular clips located at the top and bottom of the frame. 3 more circular pieces, one with radius of 12 inches and two with radius 8 inches, would clip into the 8 semicircle pieces creating the circular frame that was desired. By creating the design of a bunch of small pieces that were assembled together this allows for easy maintenance of the vehicle. Another benefit to this design is that the frame is mostly open allowing airflow to easily flow through the frame. This was a key aspect in the design in order to create lift. The propeller now has the ability to create a downward thrust not only allowing the vehicle to lift but allowing the control surfaces located below the propeller to control the attitude of the vehicle.

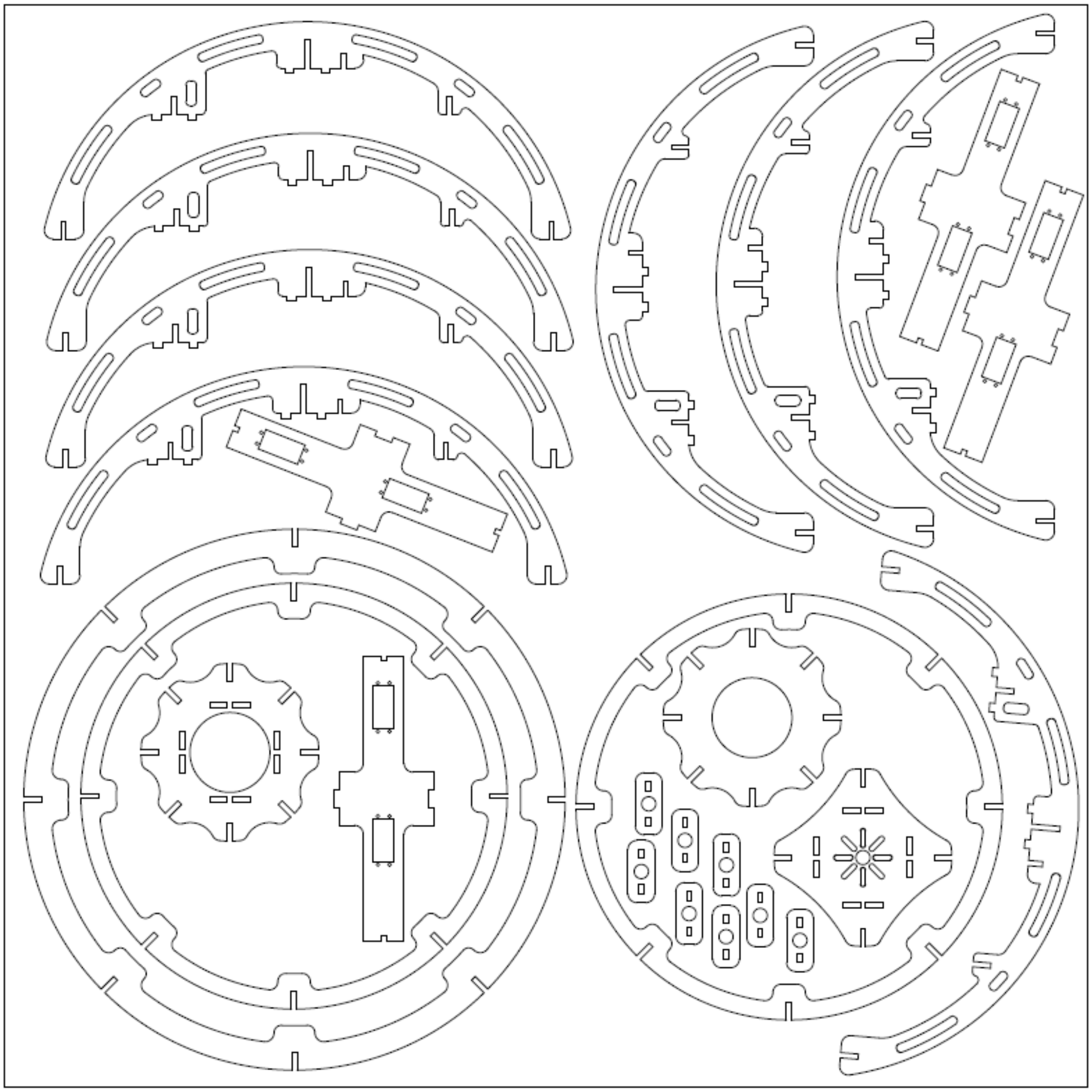


Figure 4.5.3: Individual Frame Parts Design

The two circular clips (4 inch radius) located at the top and bottom of the vehicle allowed for an additional frame to be build on the inside to store the electrical components and mount the control surfaces, propeller, and motor at the desired heights. Attached to the bottom circular clip, a 3.5”x3.5”x7” rectangular frame is used to hold all the different components in place. The frame is hollow allowing for the placement of all necessary components inside of the frame giving the vehicle a clean look while at the same time helping to reduce drag from stray parts. The small frame is also very important to keep the electrical components from interfering with the control surfaces. This rectangular frame allows for 8 different rods to be mounted off of it and connect to the exterior frame. These rods will hold the control surface flaps in place. There will be 8 different control surfaces located on the vehicle extending from each corner of the rectangular interior frame. The upper 4 control surfaces will be placed close to the top of the interior frame at a height of 6 inches from the base of the frame. The lower 4 control surfaces will be places directly below the upper 4 but will be places at a height of 3 inches from the base.

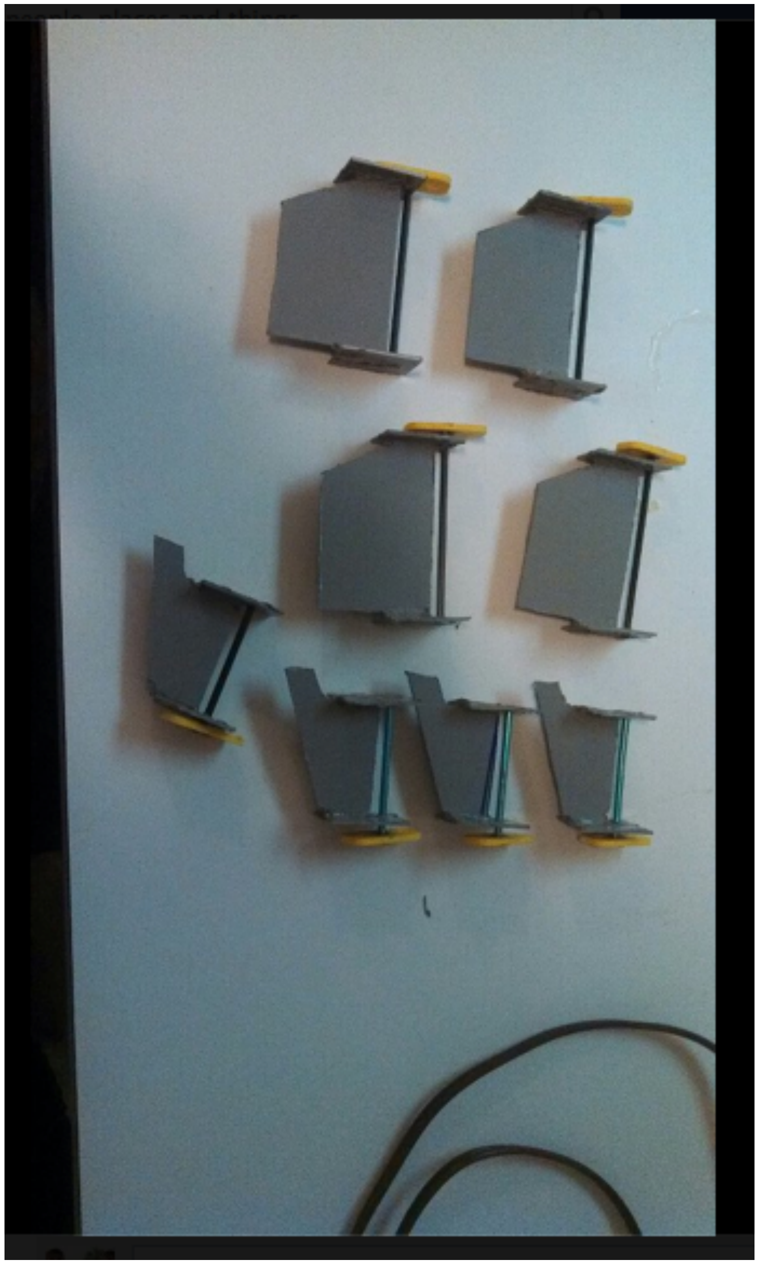


Figure 4.5.4: Vehicle Control Surfaces

Another benefit to having an interior frame build is that it helps to create a base to mount the propeller, motor, and servos on. The motor and propeller can be easily mounted at the top of the base, creating a stable spot for them to be located at the correct height to create the downward draft necessary for the control surfaces to work. The servos can also be securely mounted to the interior frame at the designated heights of the control surfaces. Also the camera can be securely mounted on the bottom of the interior frame to give a clear downward view of the ground below as intended.

Figure 4.5.4: Vehicle Control Surfaces

Now that the frame has been designed and all the components have a secure location to me placed in or mounted too issue of what to make the frame out of arises. In order for the vehicle to properly fly the design called for the vehicle to be as light as possible. Research showed many materials have been used to create similar projects including wood, metal, plastic, and Styrofoam. These other projects though ran into many problems with these types of materials. The vehicle that was build out of wood was too heavy for the prop and motor to lift so the vehicle could never get off the ground. Metal was very similar to the wood in that for material at a reasonable cost it was typically heavy. There were types of metal though such as carbon which were lighter but for the amount needed, the cost outweighed the benefit. Similar designs and prototypes showed that 3mm PVC Foam Board worked best for the frame design. PVC Foam Board allows for a sturdy light weight frame weight just about 230 grams, while at the same time being relatively easy to cut and make adjustments too if need be.

5.1: Microcontroller

When the project first began, the original plan was to have the video data from the camera processed and possibly even encoded by the microcontroller prior to being broadcast. The concern was that the bitrate of uncompressed video would be too high and would require too much bandwidth to be feasible. This is what led to the selection of the C28x Piccolo series of 32-bit microcontrollers. As it turned out, however, the camera selected for the vehicle had its own antenna capable of broadcasting the video data. However, the Piccolo series stuck because its evaluation board happened to be much cheaper than comparable microcontrollers at only $17.00—whereas other evaluation boards were $150 or more.

From there, the selection of Piccolo MCUs was further reduced by the decision to have a total of 6 PWM outputs on the chip. One output would be used to control the upper set of four control servos, four would be used to control each of the lower control servos, and one would be left as an extra option if another control signal was needed, most probably for the motor. Using Texas Instruments' parametric search tool, six Piccolo MCUs were found to have six PWM outputs: the F2803xPAG family.

All members of the F2803xPAG family have the 12 PWM channels the vehicle needs to control its various mechanical parts. All of the family members are low power, requiring only 3.3V of supply voltage. As a matter of fact, they share much of the same architecture across the board. The differences between members of the family are slight, but they are enough to warrant some discussion. Figure 5.1.1 is a table summary of the similarities and differences between members of this family of microcontrollers.

The greatest difference between members of the F2803xPAG family is the amount of flash memory available. Flash memory is very important to the prototype of this project. While determining the various values and limits needed to keep the vehicle stable and on-course, many variables and equations will need to change. It is important that the amount of rewritable memory be sufficient to hold a changing and likely expanding program.

Beyond flash memory is the amount of SARAM within the MCU. The route data will be stored in RAM to be read later, so the amount of RAM available is a direct indicator of how large the routes can possibly be. Some reason will have to be exercised here as the routes will not have an unreasonable number of points to be stored, but more memory gives more flexibility in how those points are actually stored. More RAM also provides more flexibility for storing values from mathematical operations.

Figure 5.1.1: Comparison of Microcontrollers
(Permission for use of data pending)

MCU	28030-PAG	28031-PAG	28032-PAG	28033-PAG	28034-PAG	28035-PAG
Clock Rate	60 MHz	60 MHz	60 MHz	60 MHz	60 MHz	60MHz
Control Law Accel.	No	No	No	Yes	No	Yes
Flash Memory	16K	32K	32K	32K	64K	64K
SARAM	6K	8K	10K	10K	10K	10K
PWM Channels	12	12	12	12	12	12
Hi-Res PWM	No	No	2	2	2	2
Supply Voltage	3.3 V	3.3 V	3.3 V	3.3 V	3.3 V	3.3 V

The Control Law Accelerator, mentioned in Figure 5.1.1, is a specialized component of the Piccolo architecture that speeds up certain floating-point calculations. Since many of the operations that the MCU will be performing will be calculations involving floating-point GPS coordinates, it is most likely beneficial to have the CLA. At the very least, there does not appear to be any significant downside to having CLA in the MCU.

Choosing to include CLA reduces the decisions to two MCUs, the 28033PAG and the 28035PAG, and the only significant difference between the two is the amount of flash memory included in the architecture. Given that, once again, there seems to be no apparent downside to the benefit, and because most computer engineering design guides tend to agree that one should overestimate his or her memory requirements, the 28035PAG, pictured in Figure 5.1.2 will be the MCU flying aboard the AFSD.

Figure 5.1.2: TMS320F28035PAG Microcontroller
(Permission for use of photograph pending)

Figure 5.1.3 shows the pin-out of this microcontroller, which will be necessary for the construction of the PCB and for further design of any schematics.

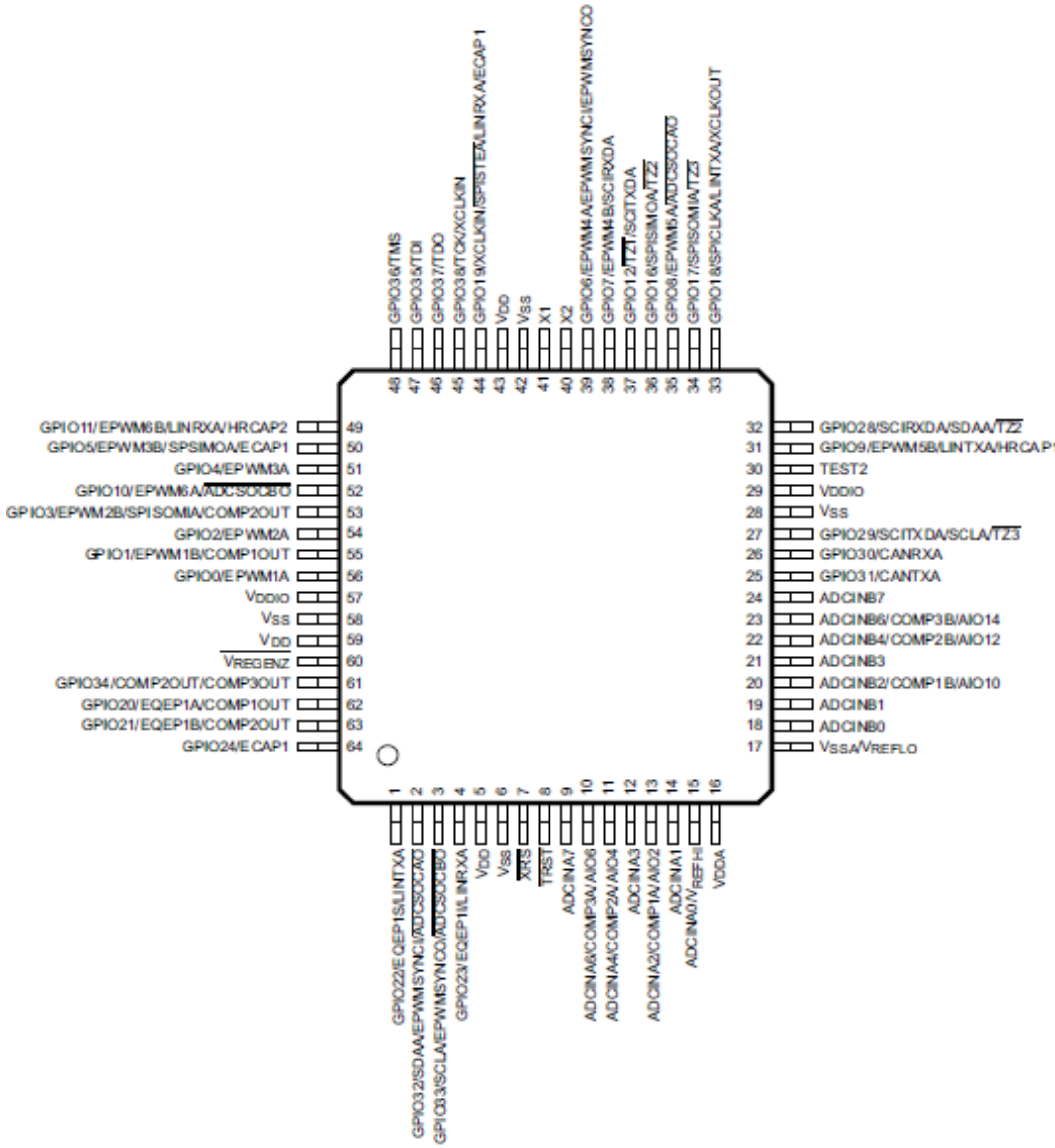


Figure 5.1.3: Pin-Out of the 2803xPAG Family

Figure 5.1.4 shows the functional block diagram of the 2803xPAG family. The 28035PAG simply includes all of the listed features at their maximum amounts and values.

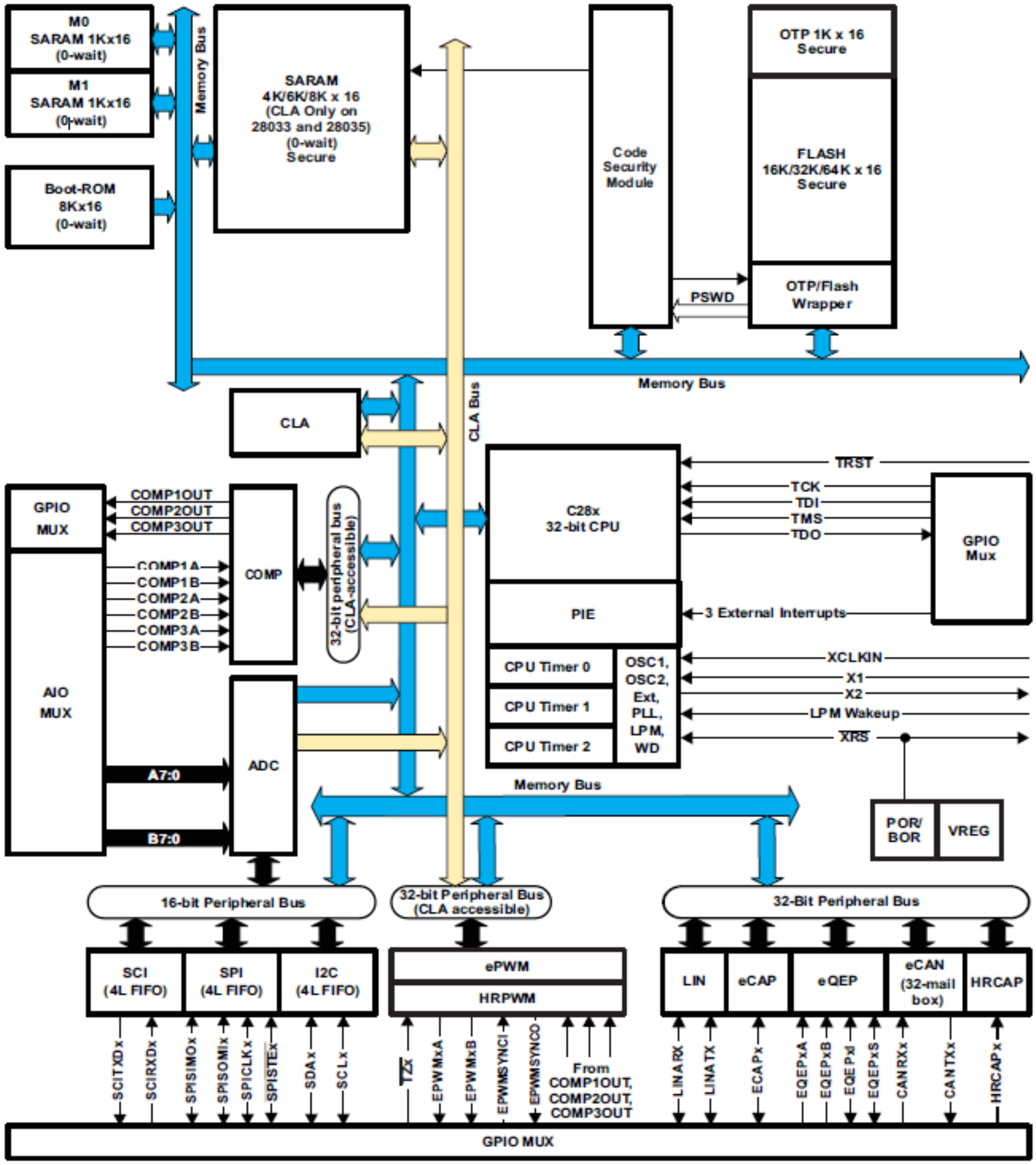


Figure 5.1.4: Functional Block Diagram of F2803xPAG Microcontroller Family

(Permission for use of diagram pending)

This diagram is slightly overwhelming, but shows the intricacies of the various components inside the MCU and will be important during the prototyping phase for understanding exactly how the data flows. For now, the simplified diagram of Figure 5.1.5 will suffice for garnering understanding of how the external components connect to the MCU.

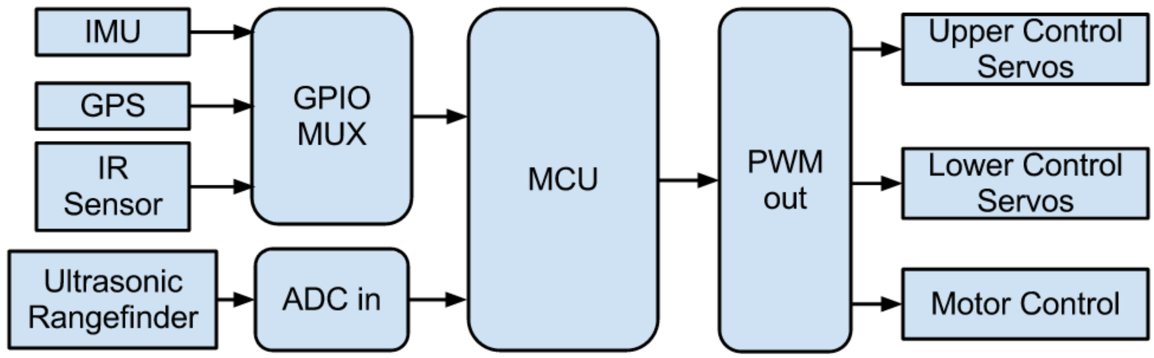


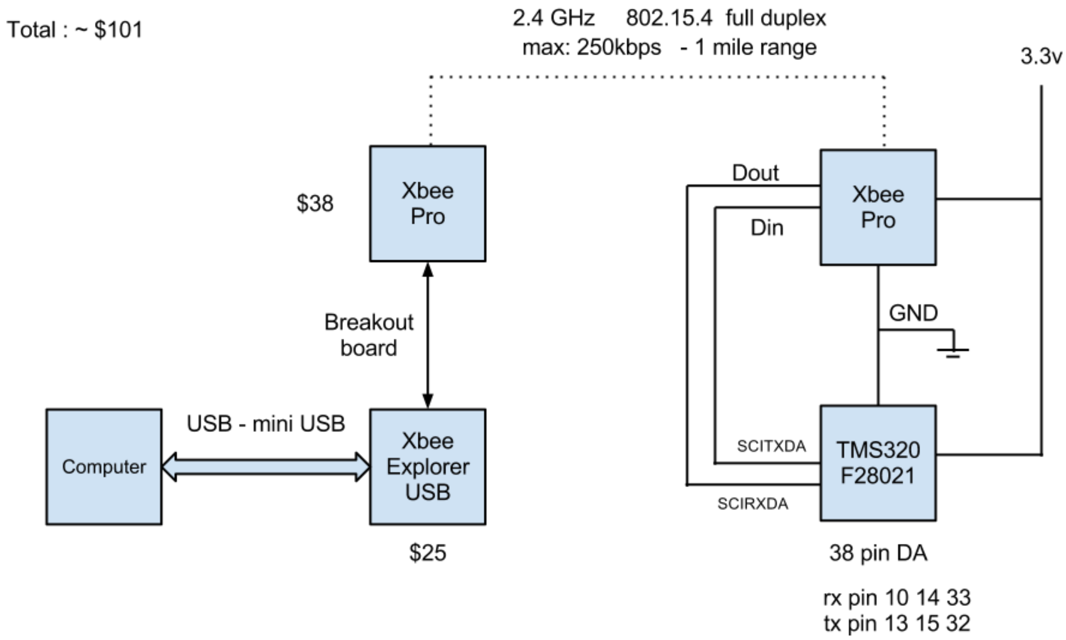
Figure 5.1.5: Simplified I/O Diagram for MCU

The IMU, GPS, and IR sensors all feed through GPIO MUX and are handled by the various serial interfaces to provide useful data to the MCU. The ultrasonic rangefinder, however, outputs in an analog signal that must be measured and converted to a digital quantity before it can be used in any calculations. Similarly, any other devices that output this way must go through the analog-to-digital converter. The MCU outputs to the various mechanical components by outputting specific waves across its various PWM channels.

5.2: Wireless

For the wireless transceivers this design will be using the XBee Pro. The UAV is designed to fly less than one mile away from its docking station due to flight and battery limitations. The XBee Pro allows the design to communicate between the UAV and the computer running the UAV software at a data rate max of 250kbps, and outputs 60mW (+18dBm). Using two XBee Pro Series 1 (802.15.4) we set up a system as seen below in figure 5.2.1.

The XBee module could be configured and operate through a serial port but it made sense to opt for the Explorer USB since it was fairly cheap and any one could easily use it on most computers as serial ports are becoming harder to find on modern computers. USB is standard with most computers and features a higher port line speed.



	USB 3.0	2.0	1.1	Serial	Parallel
port line speed	5 Gbps	480 Mbps	12 Mbps	115 Kbps	115 KBps

Figure 5.2.1 Wireless Setup

The XBee pro module is used to interface with the computer is mounted using an Xbee Explorer USB as an adapter. It was decided that it would be best to use the Xbee explorer USB as opposed to a serial connection to take full advantage of the transmission rate of 250kbps otherwise the communication would be bottlenecked by the port line speed. This did create some confusion as it was believed that the explorer eliminated the serial interface, instead it uses a USB connection as a buffer between the serial connection built into the explorer.

Figure 5.2.2 XBee Explorer
(Permission for use of photograph pending)

5.2.1: Setup

Configuring the XBee modules for the purposes of this project can be broken into two main steps. Configuring the local XBee module, the module which will be connected directly to the computer, and configuring the remote XBee module, the module connected to the UAV. Each will be configured on a computer using the XBee explorer to interface via USB.

First you would configure the module that will be used remotely. In order to properly configure the module it is required to have the X-CTU software visible in figure 5.2.3, provided by the manufacturer and available for download from the manufacturer's website. This software is used to set the appropriate registers in the module to properly configure it as a transmitter. The configuration will be set so that the remote XBee should continuously transmit coordinate data to other XBee's in the area, since this project only intends to have one receiver that will be the local XBee. It is also important to be upgrading the modules to the latest firmware using this software during configuration.

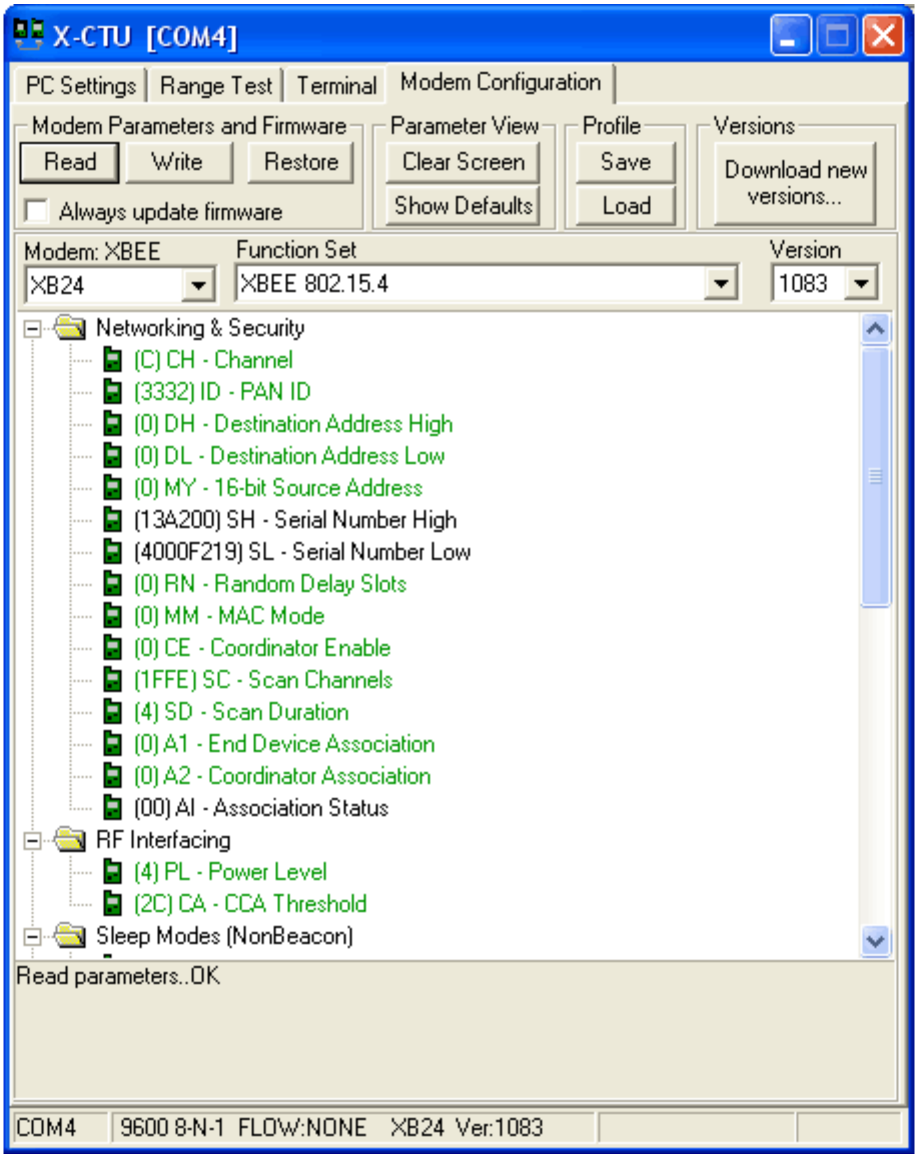


Figure 5.2.3 X-CTU Software Interface

Setting the appropriate baud rate is crucial to ensure the design has enough bandwidth available to send data as quickly as possible and avoid delayed responses. To calculate the appropriate baud rate it is important to consider what data rate would suffice for this project. Considering GPS coordinates are the only thing that will be being transmitted; The size of GPS coordinates can be used to get a rough estimate for the appropriate baud rate.

$$\text{GPS} \approx 32 \text{ bits}$$
$$\text{Sample Rate} = 10 \text{ Hz}$$
$$32 \text{ bits} \times 10 \text{ Hz} = 320 \text{ bits/second}$$

Also taking into account delimiters used for overhead in formatting the data

$$\text{Delimiter} \approx 8 \text{ bits}$$
$$11 \times 8 + 320 = 480 \text{ bits/second}$$

That gives 480 bits/sec just for maintaining current position up to date on the client application. Data transmission during route initial route transmission should not be slow and allow for a quick take off once the route has been plotted. Since we have an upper limit of 250 kbps to this allows us enough elbow room to have at most a route of 580 individual points transmitted in a second. For testing purposes and initial specifications it was decided to use a baud rate of 9600 to avoid data loss and conserve power. Since the baud rate is 9600 and there are two bits per symbol, the number of symbols is $2^2 = 4$. The bit rate is:

$$R = 9600 \times 3.32 \log_2 4 = 9600 \times 2 = 19200 \text{ bits/second}$$

Where R is the data rate. To set the baud rate one could use a hyper terminal interface as a backup if the X-CTU program becomes inaccessible when beginning prototype testing, and the set of AT commands (one could alternatively use puTTy rather than hyperterminal). Ensuring the COM port is set appropriately we would query the module to enter AT command mode with +++ and expect OK in return. To ensure the module is connected and responding appropriately we could query with an AT command which should respond with OK if it is ready to accept commands. The baud rates are set by numbers which correspond to them defined in the table in figure 5.2.4. Alternatively the baud rate is easily set in the X-CTU graphical user interface under the **serial interface -> interface data rate**

Figure 5.2.4: Baud Rates

Number	Baud Rate
0	1200
1	2400
2	4800
3	9600
4	19200
5	38400
6	57600
7	115200

Setting the baud rate to 9600 yields 19.2kbps. This is well below the maximum data transfer rate possible with the XBee Pro modules, but considering the power consumption and limits the project to a reasonable transfer rate that meets the requirements needs without getting greedy.

Once configuration is set on the remote XBee we simply remove it from the XBee explorer and configure the local XBee, which needs to be set with the same baud rate and set to listen and send the data to the computer on a simulated serial port, and transmit data to the remote XBee module.

It is also worth noting that the XBee settings must be made to not use default settings to avoid interference from other XBee's in the area. To avoid this occurrence the XBee network and modules (remote and local) will be set with unique ID's. Once these have been set we can eliminate unintended packet interception by addressing the packets. To address the messages from one module to another we simply set the Destination Address (DL + DH) of the sender to match the Source Address (SL + SH) of the intended destination module. Where SL, SH, DL, DH are hex addresses broken into 2 bytes.

5.2.2: Local XBee - API

Unfortunately the manufacturer does not provide an API but the module does support API command sets. To interface the module with the remote terminal interface application it was decided that the project should use the "xbee-api A Java API for Digi XBee/XBee-Pro OEM RF Modules" which uses a GNU public license. It provides an extensive library of objects and methods to easily interact with the module from within the remote terminal application.
Sending and receiving and even reconfiguring a module from the API is as simple as creating an Xbee object creating a transmission request and sending. For example :

```
XBee xbee = new XBee();
xbee.open("COM4", 9600);
xbee.sendSynchronous(new TxRequest16(new
     XBeeAddress16(16bit address, new int[] {payload})));
```

Would open a connection to the XBee module on COM port 4 at a baud rate of 9600, and then send a transmission request to the module addressed to the 16 bit address with a payload.

5.2.3: Remote XBee - MCU

It will connect to the MCU board simply using SPI interface and is an ideal solution for a low-power, low data rate pin for pin network. Figure 5.2.5 below shows internal data flow of the module. According the XBee data sheet, those modules interface to a host device through a logic-level asynchronous serial port. Through its serial port, the module can either communicate with logic and voltage compatible UART or through a level translator to a USB interface port. Figure 5.2.5 below illustrates the serial bit pattern of data passing through the module. Serial communication depends on the microcontroller's UART and the RF module's UART to configure with compatible settings such as baud rate, parity, start bits, stop bits, and data bits.

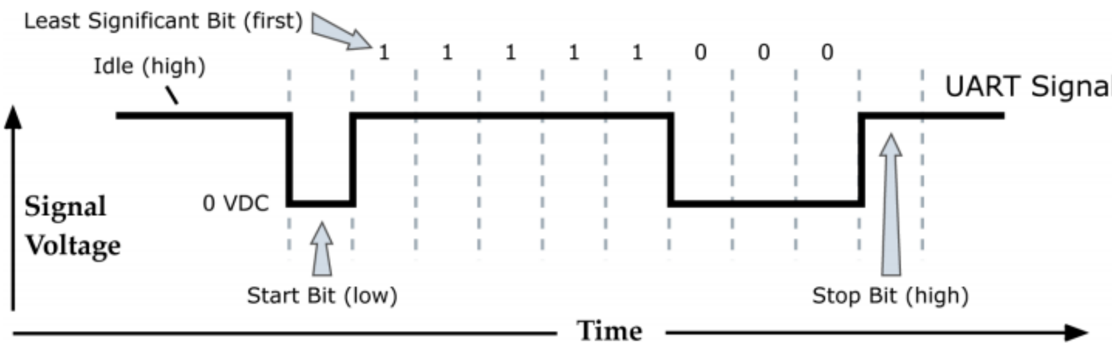


Figure 5.2.5: Serial Data Transfer

XBee modules supports sleep mode. Sleep modes enable the RF module to enter states of low power consumption. The module will be required to enter sleep mode as times to conserve power in the UAV, but never during flight. The SM command is central to setting sleep mode configuration. By default, sleep modes are disabled (SM=0) and the module remains in Idle/ Receive mode, which indicates the module is constantly ready to respond to serial or RF activity. Table 11 below shows the sleep mode configurations.

Figure 5.2.6 Sleep Mode Configurations from Digi International, Inc					
Sleep Mode Setting	**Transition into Sleep Mode**	**Transition out of Sleep Mode (wake)**	**Characteristics**	**Related Commands**	**Power Consumption**
Pin Hibernate (SM =1)	Assert (high) Sleep_RQ (pin 9)	De-assert (low) Sleep_RQ	Pin/Host-controlled / NonBeacon systems only / Lowest Power	(SM)	< 10μA (@3.0 VCC)
Pin Doze (SM=2)	Assert (high) Sleep_RQ (pin 9)	De-assert (low) Sleep_RQ	Pin/Host-controlled / NonBeacon systems only / Fastest wake-up	(SM)	<50μA
Cyclic Sleep (SM =4)	Automatic transition to Sleep Mode as defined by the SM(Sleep Mode) and ST (Time before Sleep) parameters.	Transition occurs after the cyclic sleep time interval elapses. The time interval is defined by the SP (Cyclic Sleep Period) parameter.	RF module wakes in pre-determined time intervals to detect if RF data is present / When SM =5	(SM), SP, ST	<50μA when sleeping
Cyclic Sleep (SM =5)	Automatic transition to Sleep Mode as defined by the SM (Sleep Mode) and ST (Time before Sleep) parameters or on a falling edge transition of the SLEEP_RQ pin.	Transition occurs after the cyclic sleep time interval elapses. The time interval is defined by the SP (Cyclic Sleep Period) parameter.	RF modules wakes in pre-determined time intervals to detect if RF data is present. Module also wake son falling edge of SLEEP_RQ	(SM), SP, ST	< 50 μA when sleeping

In order to design the PCB board properly for the project, the team had to know the digital electrical characteristics of this module. Width of trace is determined by the current that flows in the circuitry. To sum the total current, electrical characteristics of each electronic unit that were going to be implemented in the PCB board had to be acknowledged. See Figure 5.2.6 below for the detailed module specification.

Figure 5.2.7: XBee Module Specifications

Symbol	Characteristics	Condition	Min	Typical		Max	Unit
V_{IL}	Input Low Voltage	All Digital Inputs	-	-		$0.35*V_{CC}$	V
V_{IH}	Input High Voltage	All Digital Inputs	$.7*V_{CC}$	-		-	V
V_{OL}	Output Low Voltage	I_{OL} = 2 mA, V_{CC} >= 2.7V	-	-		0.5	V
V_{OH}	Output High Voltage	I_{OH} = -2 mA, V_{CC} >= 2.7V	V_{CC} - 0.5	-		-	V
II_{IN}	Input Leakage Current	V_{IN} = V_{CC} or GND, all inputs, per pin	-	0.025		1	µA
II_{OZ}	High Impedence Leakage Current	V_{IN} = V_{CC} or GND, all I/O High-Z, per pin	-	0.025		1	µA
TX	Transmit Current	V_{CC} = 3.3V	-	45 (Xbee)	215 , 140 (PRO, Int)	-	µA
RX	Receive Current	V_{CC} = 3.3V	-	50 (Xbee)	55 (PRO)	-	µA
PWR-DWN	Power-down Current	SM parameter = 1	-	<10		-	µA

5.3 Camera and Receiver

The project will be using KY-2.4GR01+C-203A (receiver, cam respectively) to meet the visual requirements of our project.

Figure 5.3.1: Camera + Receiver
(Permission for use of photograph pending)

The KY-2.4GR01+C-203A set functions in 2.4GHz frequency, with a transmission power of 10mW. The miniature color camera is less than a cubic inch and allows itself to be easily mounted on the UAV. The camera has a built in transmitter, built-in rechargeable Li-Battery that operates for 3 hours continuously. The camera has a horizontal resolution of 380, a viewing angle of 62 degrees, a bandwidth of 18MHz, and a consumption current of 80mAh. The receiver has 4 channels, is supplied by an AC/DC adaptor (12V, 500mA), and a consumption current of .3W. The receiver has RCA outputs which can be easily connected to an alternate monitor or fed through an adapter to view on a computer making it ideal for this project.

5.3.1: Setup

To set up the camera + receiver simply twist the detachable receiver antenna into the back of the receiver. Connect the receiver to the monitor with AV cable or feed through an adapter to USB. Plug the DC 12V 500mA adaptor into the power jack of the receiver. Then insert the DC 8V 200mA adaptor into the power jack of the camera. Adjust the lens of the camera to the best position; mount the camera to the UAV with the screw. Since the camera has a built in Li-Battery of the camera needs to be recharged every 3 hours of use it is necessary to account for charging it via charging station

Since there will be no image processing in the onboard system and video transmission is very bandwidth expensive it made sense to eliminate video from being handled by the MCU onboard all together. This allows the freedom to use a dedicated transmitter for video which allows oversight of the near impossibility of attempting to transmit moderately decent quality video over the very limited bandwidth readily available on most XBee RF modules.

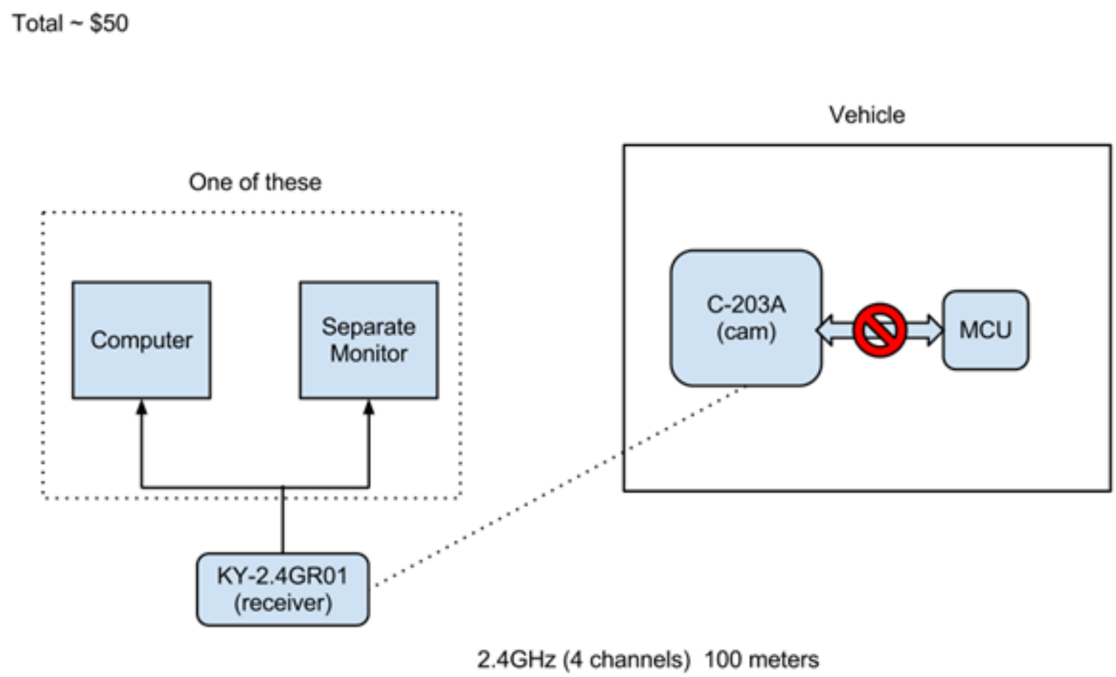


Figure 5.3.2: Camera Integration

5.3.2: CMOS vs. CCD

Deciding on the camera was very difficult because when comparing technologies available for the application there were many factors to consider; the image quality, device cost and size.

CMOS technologies were a clear cut winner for this application when it came to size because they are much smaller than their CCD counter parts. Having a small camera to avoid weighing down the UAV was a big concern just like the main priority across the board was to cut down weight on components where ever possible. Unfortunately the size advantage came at a cost of price and quality. As size went down so did quality and only slightly affected price.

It was also found that CMOS were almost always cheaper than CCD cameras. The price difference was presumably based on the fact that “CCDs have traditionally provided the performance benchmarks in the photographic, scientific, and industrial applications that demand the highest image quality” (TeleDyne , CCD vs CMOS). The table in figure 5.3.3 shows many of the differences between CCD cameras and CMOS which were considered.

Figure 5.3.3: CMOS vs. CCD
Permission for use of data pending

Feature and Performance Comparison	CCD	CMOS
Signal out of pixel	Electron packet	Voltage
Signal out of chip	Voltage (analog)	Bits (digital)
Signal out of camera	Bits (digital)	Bits (digital)
Fill factor	High	Moderate
Amplifier mismatch	N/A	Moderate
System Noise	Low	Moderate
System Complexity	High	Low
Sensor Complexity	Low	High
Camera components	Sensor + multiple support chips + lens	Sensor + lens possible, but additional support chips common
Relative R&D cost	Lower	Higher
Relative system cost	Depends on Application	Depends on Application
Performance	CCD	CMOS
Responsivity	Moderate	Slightly better
Dynamic Range	High	Moderate
Uniformity	High	Low to Moderate
Uniform Shuttering	Fast, common	Poor
Uniformity	High	Low to Moderate
Speed	Moderate to High	Higher
Windowing	Limited	Extensive
Antiblooming	High to none	High
Biasing and Clocking	Multiple, higher voltage	Single, low-voltage

Although a CMOS camera was ultimately decided on because it more closely aligned with the proejcts requirements (price, weight), the decision left the design with lower image quality in low light situations. Even so the defect is easily remedied with an LED flash light if necessary and will be further analyzed and if necessary implemented during the testing phase.

The one concern with having settled on a CMOS camera is related to a future implementation that would most likely require the design to change to a CCD camera. Since CMOS cameras generally don't update an entire frame at a time but rather scan lines at a time this gives a warped image (jello effect) when capturing objects in motion. An example of the jello effect on an object in motion can be seen in figure 5.3.4. A future implementation we are considering of adding; image processing to allow the UAV to track objects of interest may suffer from this updating method. The rolling shutter which causes this effect is not present in CCD cameras which give a clearer image due to its “global shutter” which updates entire frames at once.

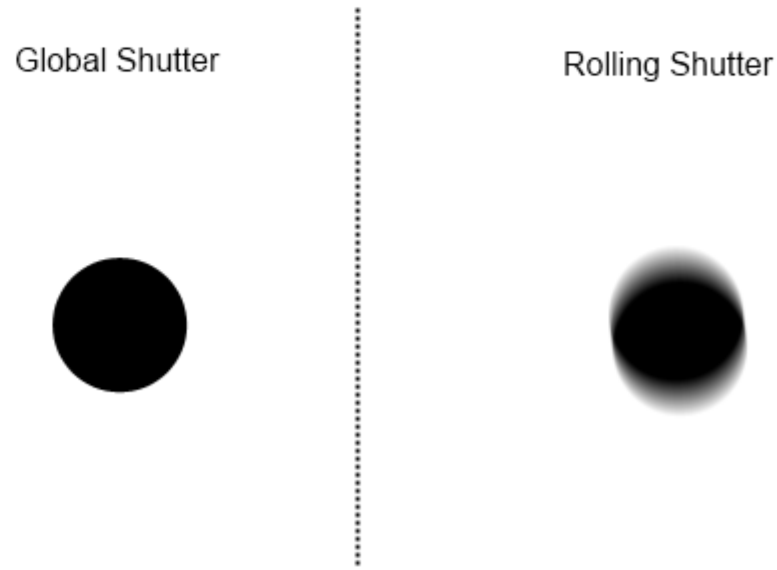


Figure 5.3.4: Global Shutter vs. Rolling Shutter

5.4: Motor and Servo Control

Control of the mechanical components is achieved by generation and manipulation of control signals by the MCU. However, which servos and motors are actually used plays an important role in how the craft maneuvers. The original plan of action was to minimize the amount of weight contributed by the servos and motor, but as time went on this became a less viable plan of action. The servo chosen based on this criteria, the Futaba S3117 pictured in Figure 5.4.1, was eventually regulated to controlling only the top four control surfaces due to concerns over the amount of torque the micro-sized servos would be able to provide.

Figure 5.4.1: Futaba S3117 Micro-Servo
(Permission for use of photograph pending)

In terms of size and weight, nothing out-minimizes the S3117. Roughly the size of a quarter, as shown in Figure 5.4.2, the S3117 is only 6 grams in weight. This made it ideal for controlling the low-torque-required upper control surfaces.

Figure 5.4.2: S3117 Size Comparison
(Permission for use of photograph pending)

For the lower control surfaces, a more powerful servo was desired. Concerns were that the amount of torque exerted by these surfaces may cause the S3117 to stall and become damaged. For this reason, the lower servos were upsized to the S3115, pictured in Figure 5.4.3.

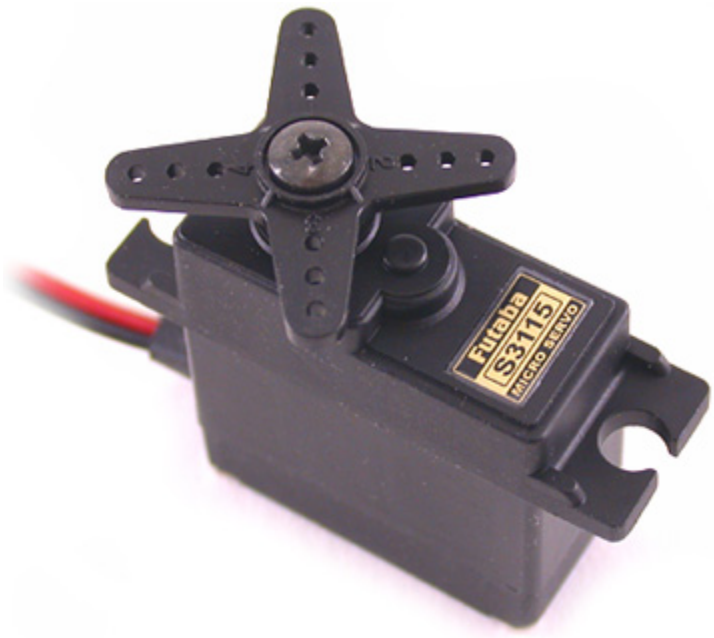


Figure 5.4.3: Futaba S3115 Servo
(Permission for use of photograph pending)

While only slightly larger than the S3117, the S3115 provides 38.9 oz-in of torque to the 20.8 oz-in of the S3117. However, the weight of the S3115 is nearly three times that of the S3117, weighing in at 17 grams. It is for this reason that S3115's are not used for all eight servos: The power isn't required so the weight isn't justified.

Both the S3117 and the S3115 are analog servos. They are powered by a DC voltage applied to the red wire. The S3117's torque varies with the amount of DC voltage applied—20.8 oz-in at 4.8 V, 23.6 oz-in at 6 V—but uses more power with higher voltage. The S3115, however, only takes 4.8 V. The power supplied to all servos will be 4.8 as a matter of simplicity, and also because the upper control surfaces won't be compromised by a lack of 2.8 oz-in.

The servos are controlled along the white wire with a 50 Hz square wave, oscillating between 0V low and 3-5V high. The control comes from the duty cycle of this wave. By adjusting the pulse between approximately 1 ms and 2 ms, it's possible to rotate the servo from 0 to 180 degrees. The servos will be oriented so that 90 degrees is approximately the "no-adjustment" position for the control surfaces. An example of a control signal is shown in Figure 5.4.4.

Figure 5.4.4: Example of a Servo Control Signal
(Permission for use of information pending)

The motor is similarly controlled by a square wave, where the frequency of the wave controls the speed of the motor while the duty cycle controls the torque. For the purposes of the AFSD, the Exceed RC Helium 450 Brushless Motor, pictured in Figure 5.4.5, was selected.

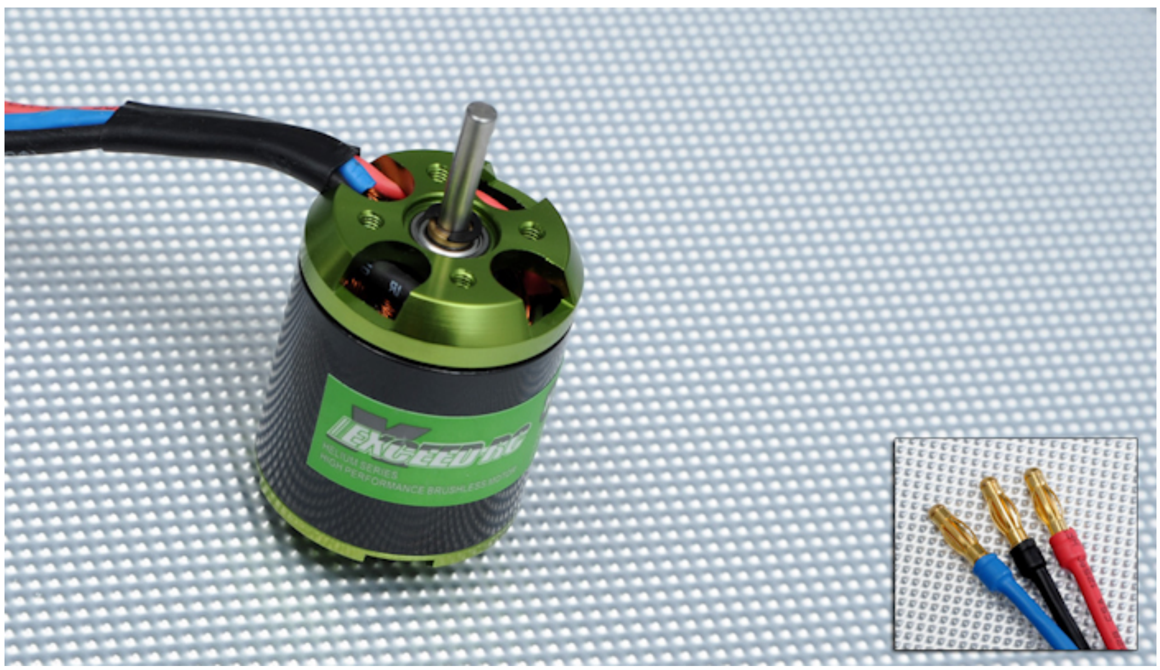


Figure 5.4.5: Exceed RC Helium 450 Brushless Motor
(Permission for use of photograph pending)

This motor is likely much more powerful at max power than will ever be needed for the AFSD, but it was deduced that it would be better to have a powerful motor and under-power it than to have a motor that's too weak even at full power.

This motor will be driving a propeller resembling (if not exactly), the Scale 4-Blade 9-1/2 X 10 Variable Pitch Prop, available from Hobby Lobby. Pictured in Figure 5.4.6, this prop is approximately the size needed for the AFSD.

Figure 5.4.6: Scale 4-Blade 9-1/2 X 10 Variable Pitch Prop
(Permission for use of photograph pending.

5.5: IMU

One of the main design criteria for the AFSD is the in-flight stability of the vehicle. If the vehicle is obstructed by any external force during flight, the AFSD should be able to to automatically correct its attitude and continue flying the desired path. For the vehicle to be able to correctly monitor its initial attitude and stabilize from interference from external forces, an IMU (Inertial Measurement Unit) is necessary to accomplish this. The IMU provides platform stability in pitch, yaw, and roll, helping to correct or align the AFSD with the correct attitude relative to the horizon. Providing the necessary data to maintain stability, the IMU becomes the core component in the vehicles autopilot system.

In researching the IMU's to find the best one that would meet the design need and best work with the AFSD's design became a rather difficult task. The selection was narrowed down as a IMU with 9 degrees of freedom decided to be selected. A few of the IMU's that had 9 DoF had preset ranges for the accelerometer and the gyroscope and for many of the ones that were preset the range was not sensitive enough for vehicles design. To provide options for the IMU, three different ones have been selected and are being compared together to see which will be best in the projects design. The three IMU's being considered are the IMU3000, ArduIMU V3, and MPU 6050. The two IMU's are compared in Figure 5.5.1 below.

Figure 5.5.1: IMU Comparison

IMU	ArduIMU V3	ADXL345 & IMU3000	MPU 6050
Price	$49.99	$59.99	$39.99
Accelerometer Range	+/-2g, 4g, 8g, 16g	+/-2g, 4g, 8g, 16g	+/-2g, 4g, 8g, 16g
Gyroscope Range	±250, 500, 1000, 2000°/s	±250, 500, 1000, 2000°/s	±250, 500, 1000, 2000°/s
Interface	Serial, I2C	I2C	I2C
Accelerometer Axes	3	3	3
Gyroscope Axes	3	3	3
Power	6-12VDC	3.3-16VDC	2.3-3.4VDC
Magnetometer	Yes	No	No
Self-Test	Yes	Yes	No

For the design of the AFSD, the ArduIMU V3 will most likely be selected to provide the vehicle with an orientation solution to keep it stable. The ArduIMU V3 provides a 9 degree-of-freedom motion tracking system by incorporating a 3-axis gyro, 3-axis accelerometer, and 3-axis magnetometer. For the IMU to properly work there are three major components that make up the chip: Invensense MPU-6000 with built in 3-axis gyro and accelerometer, 3-axis I2C magnetometer HMC-5883L, and Arduino Atmega328 microcontroller.

Figure 5.5.2: ArduIMU V3 Board
(Permission for use of photograph pending)

The MPU-6000 has six 16-bit analog to digital converts, three or which are for the gyro outputs and three for the accelerometer outputs. Both the gyroscope and accelerometer allow for the user to program the parts range, allowing for accurate tracking of both fast and slow vehicle motions. For the gyroscope the user has the ability to program a full scale range of ±250, ±500, ±1000, or ±2000 dps and for the accelerometer a range of ±2g, ±4g, ±8g and ±16g. For the purpose of the AFSD design, fast vehicle motion precision tracking will be necessary. In order to get the most accurate sensitivity rating from the gyro the maximum angular velocity should not exceed that of the gyro but should be close for precision purposes, so to make the sensitivity a close as possible a ±250 dps will be used. When the gyro senses an unwanted movement it commands the control surfaces to move in the opposite direction to compensate for the movement. The amount the gyro’s tell the servo to move is the gain. Ideally the amount of gain should match how much the helicopter was rotated so that it stays stable and facing the same direction. If the gyro tells the servo to large of a gain the control surfaces will overcompensate, thus sending the AFSD bouncing back and fourth trying to stabilize. For the accelerometer the a lower scale range will also be used in order to get a more precise reading. As a result the accelerometer will most likely have a range of ±2g. The HMC-5883L magnetometer is designed for low field magnetic sensing. It uses a 12-bit ADC coupled with Low Noise AMR Sensors that enables a compass heading accuracy of 1-2 degrees.

In selecting all the electrical components that would be used in the design of the AFSD, size was a big issue because of the limited space provided by the interior frame to store all the different components. The ArduIMU V3 ended up being one of the smallest yet still powerful and accurate IMU's on the market with a size of 1.5"x1.0". For the purpose of simplicity and a clean look the ArduIMU has a GPS port with FTDI autoswitch located on chip. The only thing necessary to connect it to a GPS is a FTDI 3.3V cable. While researching IMU's one of the main errors that IMU's seem to have is a biasing error, where the signal output from the gyro when it is not experiencing rotation. It would be a real problem if while in flight the IMU started issuing bias errors causing the AFSD to try and correct its already stable flight thus causing instability. The ArduIMU V3 reduces settling effects and sensor drift by the elimination of board-level cross axis alignment errors between the accelerometers and gyroscopes. In selecting all the electrical components that would be used in the design of the AFSD, size was a big issue because of the limited space provided by the interior frame to store all the different components. The ArduIMU V3 ended up being one of the smallest yet still powerful and accurate IMU's on the market with a size of 1.5"x1.0". For the purpose of simplicity and a clean look the ArduIMU has a GPS port with FTDI autoswitch located on chip. The only thing necessary to connect it to a GPS is a FTDI 3.3V cable. While researching IMU's one of the main errors that IMU's seem to have is a biasing error, where the signal output from the gyro when it is not experiencing rotation. It would be a real problem if while in flight the IMU started issuing bias errors causing the AFSD to try and correct its already stable flight thus causing instability. The ArduIMU V3 reduces settling effects and sensor drift by the elimination of board-level cross axis alignment errors between the accelerometers and gyroscopes.

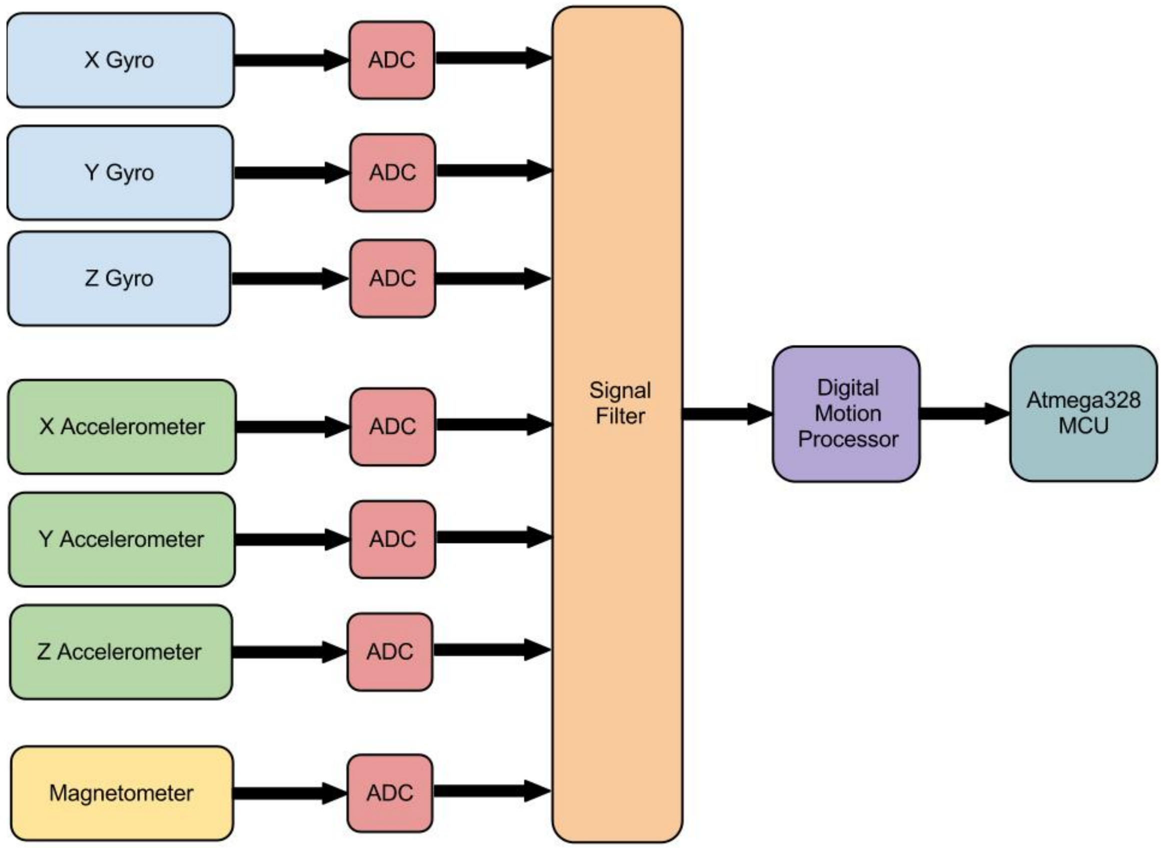


Figure 5.5.3: IMU Block Diagram

One of the very important things offered in the IMU is the ability for user self test. The board comes configured with four different LED lights for testing purposes: green, yellow, blue, and red. The green LED turns on when the IMU power is on, the yellow LED indicates the speed filter is experiencing yaw drift correct meaning the vehicle is under the threshold speed, and the red LED turns on after the IMU finishes calibration. The IMU has the predefined parameters for both an air start and ground start. If for some reason the IMU decides to restart due to power glitch or whatever other reason, the IMU will restart and recalibrate itself. When the IMU is experiencing an air start the red, blue, and yellow LED will begin flashing during the IMU startup calibration. One of the very important things offered in the IMU is the ability for user self test. The board comes configured with four different LED lights for testing purposes: green, yellow, blue, and red. The green LED turns on when the IMU power is on, the yellow LED indicates the speed filter is experiencing yaw drift correct meaning the vehicle is under the threshold speed, and the red LED turns on after the IMU finishes calibration. The IMU has the predefined parameters for both an air start and ground start. If for some reason the IMU decides to restart due to power glitch or whatever other reason, the IMU will restart and recalibrate itself. When the IMU is experiencing an air start the red, blue, and yellow LED will begin flashing during the IMU startup calibration.

The ArduIMU outputs serial data in either binary or ASCII formats depending on the user configuration. The user has the ability to pick the flags that appear in the output on the computer. When the IMU is coupled with a GPS the output unit can display the following:

1. Roll: Measured in degrees with positive and increasing as the right wing drops
2. Pitch: Measured in degrees with positive and increasing as the nose rises
3. Yaw: Measured in degrees with positive and increasing as the nose goes right
4. Latitude: Measured in decimal degrees times 10^7
5. Longitude: Measured in decimal degrees times 10^7
6. Altitude: Measured in meters above sea level times 10^1
7. Course over ground:
8. Speed over ground:
9. GPS Fix: A binary indicator of a valid gps fix
10. Satellite Count: The number of GPS satellites used to calculate this position
11. Time of week: Time of week is related to GPS time formats.

5.6: GPS

The GPS module in the AFSD is a very important part in the navigation system for the vehicle to be able to maintain on the route defined by the user. In order for the vehicle to stay as close to the path defined by the user, the GPS module must have a relatively fast refresh rate so that the vehicle does not deviate from the path. In researching and searching for a GPS that had quick refresh rate, it was found that most modules in the desired price range would update their position at 1Hz. In order to maintain a rapid update rate to keep the vehicle from deviating from the route a much higher refresh rate was desired. According to the design specifications set forth for the navigation system a module with a data update rate of 5-10 times per second was desired.

In researching the GPS modules that were to be used on the AFSD, three different one were considered: Fastrax UP501, Parallax RXM-SG, and MediaTek MT3329 GPS modules. In comparing the three different GPS modules all three had many of the same features as shown in figure 5.6.1. What helped to separate one GPS from the other was how well they were able to satisfy the two main goals necessary for the GPS in the navigation system. The first was being able to have an update rate of 5-10 times per second. The MediaTek and Fastrax GPS modules both satisfied this condition with having update rates of up to 10

Hz, however the Parallax GPS only had an update rate of 1 Hz so this module was taking out of consideration. The next major goal was the accuracy of the module. Due to the price range presented by the projects budget, all reasonably GPS modules accuracy was rather large ranging from 3-5 meters of accuracy. With this in consideration the MediaTek had the better position accuracy of the two modules.

Figure 5.6.1: GPS Module Comparison
(Permission for use of data pending)

GPS Module	MediaTek MT3329	Fastrax UP501	Parallax RXM-SG
Price	$37.99	$49.99	$39.99
Antenna	Built-in Patch	Built-in Patch	External
Connections	USB	6-pin SIP	11-pin SIP
Receiver (L1-1572.42	22 tracking channels	22 tracking channels	20 tracking channels
Position Accuracy	< 3 m	5 m	5 m
Velocity Accuracy	0.1 m/s	0.1 m/s	0.1 m/s
Time Accuracy	100 ns	+/- 1 µs	+/- 1 µs
Acquisition Time	Cold: 35 s Warm: 34 s Hot: 1 s		Cold: 35 s Warm: 15 s Hot 2 s
Acquisition Sensitivity	-148 dBm	-148 dBm	-148 dBm
Tracking Sensitivity	-165 dBm	-165 dBm	-159 dBm
Max. Altitude	18,000 m	18,000 m	18,000 m
Max Velocity	515 m/s	515 m/s	515 m/s
Max. Acceleration	+/- 4 g	+/- 4 g	+/- 4g
Update Rate	up to 10 Hz	up to 10 Hz	1 Hz
Baud Rate	9600 bps	9600 bps	9600 bps
Power Supply	3.3V-5V	3.3V-4.2V	3.3V-5V
Power Consumption	48 mA@5V	75 mW @ 3.0V	50 mA@5V
Size	38x38x7.8 mm	22x22x8 mm	1.7x1.6x0.6 in
Serial Data Format	8 bits, no parity, 1 stop bit	8 bits, no parity, 1 stop bit	8 bits, no parity, 1 stop bit

To satisfy this major design specification the MediaTek MT3329 GPS seems to be the best GPS for the navigation system of the vehicle. This GPS module allows for a maximum update rate of up to 10Hz and can be changed to a lower update rate in the firmware if desired. The serial baud rate can also be changed in the firmware if desire otherwise the module has a default baud rate of 38400 bps. With a relatively quick update rate the MediaTeck MT3329 GPS allows for the vehicle to update its current position quick enough and compare it to the position of the desired path so that it can stay on route.

Figure 5.6.2: MediaTek MT3329 GPS

(Permission for use of photograph pending)

Another issue that arises in keeping the AFSD on the correct route is the accuracy of the GPS. The navigation design specifications call for the GPS to be able to report its position to the order of 1/100,000 of a degree which is equivalent to about 3.43 feet. This data comes from table 4.5.1, GPS Resolution vs. Real Distance Resolution. The GPS module selected does not quite have the position accuracy necessary of 3.43 feet but has a position accuracy relatively close of less than 3 meters. This was a common issue with all the GPS modules that were of a reasonable price within the projects budget. It seemed that all GPS modules that were under $100 had a very similar position accuracy of around 3 to 5 meters. Since this is a prototype and the budget is limited the lack of position accuracy will have to be compensated and upon perfecting the design of the AFSD a more accurate GPS module will have to be considered.

The acquisition time for the GPS module is rather slow but that is typical for most modules. The cold start acquisition time is 35 seconds which wont be an issue because the GPS will not be needed immediately on the ground as the vehicle powers up. A statistic that is important though is the reacquisition time of less than one second. If the signal is ever lost it is very important for the signal to be reacquired as quickly as possible. A few of the other features that the MediaTek MT3329 has is that:

- It is already pre-mounted on a chip and has an onboard battery to power the chip.
- The chip uses a USB/UART interface.
- Just like the rest of the electrical components weight and size are important. The MediaTek MT3329 is 38mm x 38mm x 7.8mm and weights 9.45 grams.
- The module has low power consumption of 48mA at acquisition and 37 mA at tracking.

One of the convenient things about this GPS is that it comes with a built in patch antenna. With a built in antenna, a relatively high sensitivity signal of -165 dBW level. Since a typical GPS receives signals for the same amount of time as one complete C/A code cycle, resulting in the ability to track signals at a -160 dBW level. This shows that the GPS can receive signals at a slightly faster rate that the C/A code cycle. With a relatively high sensitivity this makes tracking in urban areas like the one the vehicle is tested in ideal.

The GPS has a port located on it so that it can connect with the IMU that has been selected with a FTID 3.3V cable. By connecting these two parts together the vehicle now is able to have a proper navigation system in which the position, velocity, and attitude are able to be determined as shown in figure 5.6.3. By connecting the GPS to the IMU, the GPS sends the position and velocity information to the IMU to be compared. The IMU and GPS then send the data they have through a filter to be compared and for noise to be filtered out so that the correct output of position, velocity, and attitude for the vehicle is displayed.

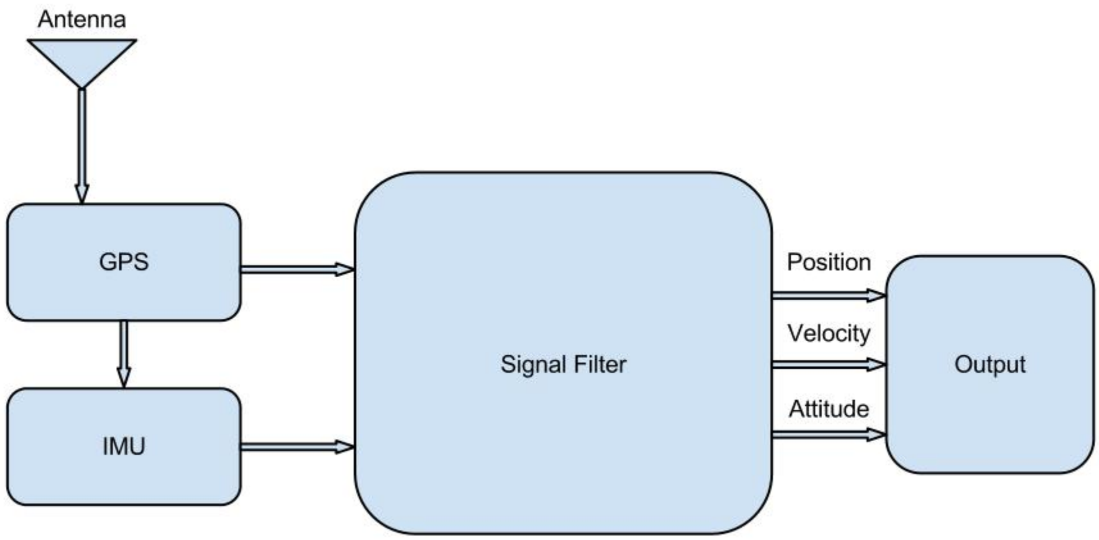


Figure 5.6.3: IMU and GPS Block Diagram

5.7 Remote Terminal Interface

Control of the UAV is managed by user using the client application on a computer connected to the appropriate transceivers. The custom built application was made in Java making it highly portable and uses the native machines look and feel to keep the application from feeling foreign relative to the OS. In figure 5.7.1 we see the application running on a windows 7 machine.

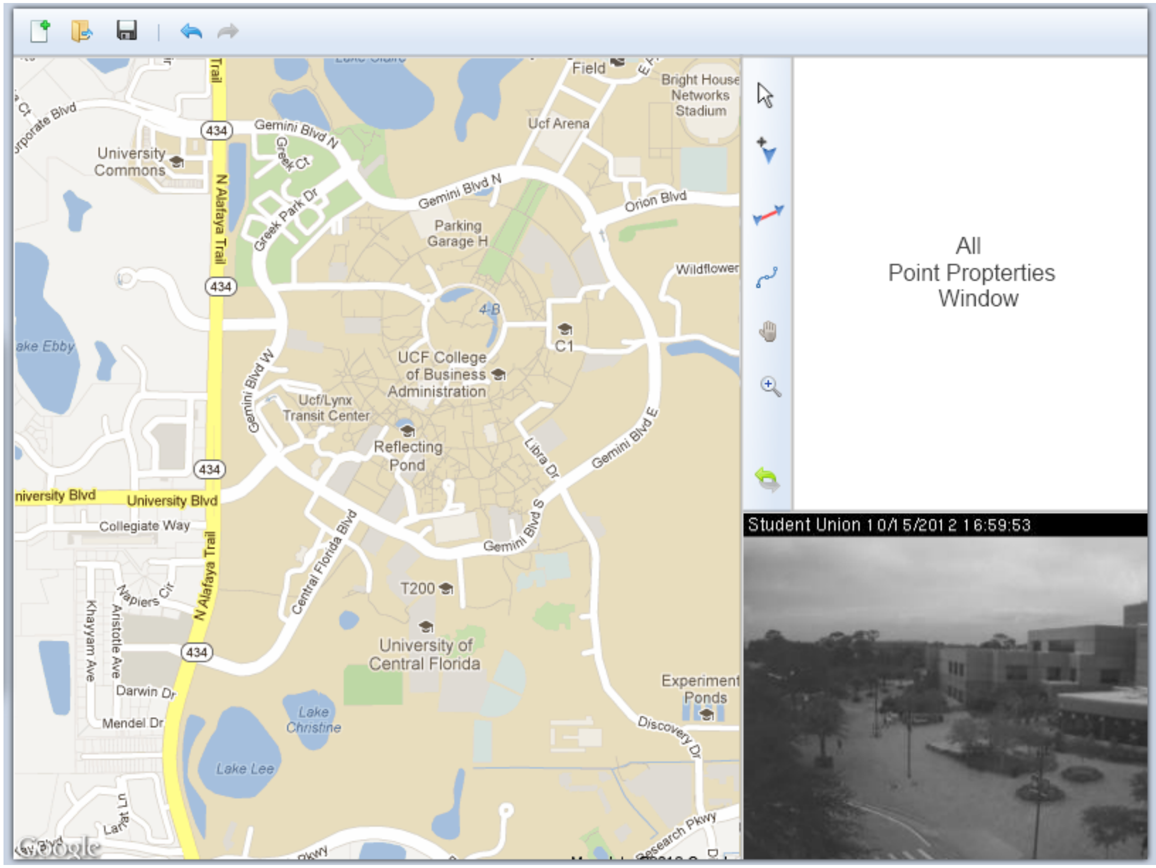


Figure 5.7.1 Remote Terminal Interface

5.7.1: Primary Functions

The GUI can be broken down into functions and displays. The main functions are displayed at the top left. In figure 5.7.1 the main functions are “New”, ”Open”, ”Save”, “undo” and “redo” respectively which are highlighted in figure 5.7.2 .

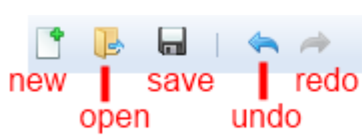


Figure 5.7.2 Primary Functions

“New”, ”Open” and ”Save” allow the user to create a new, open an existing or save the current path drawn in the map display.

5.7.2: Primary Display

The map display (figure 5.7.3) is the primary display and uses the Google Maps API and the GPS data from the UAV to accurately display the current location of the UAV.

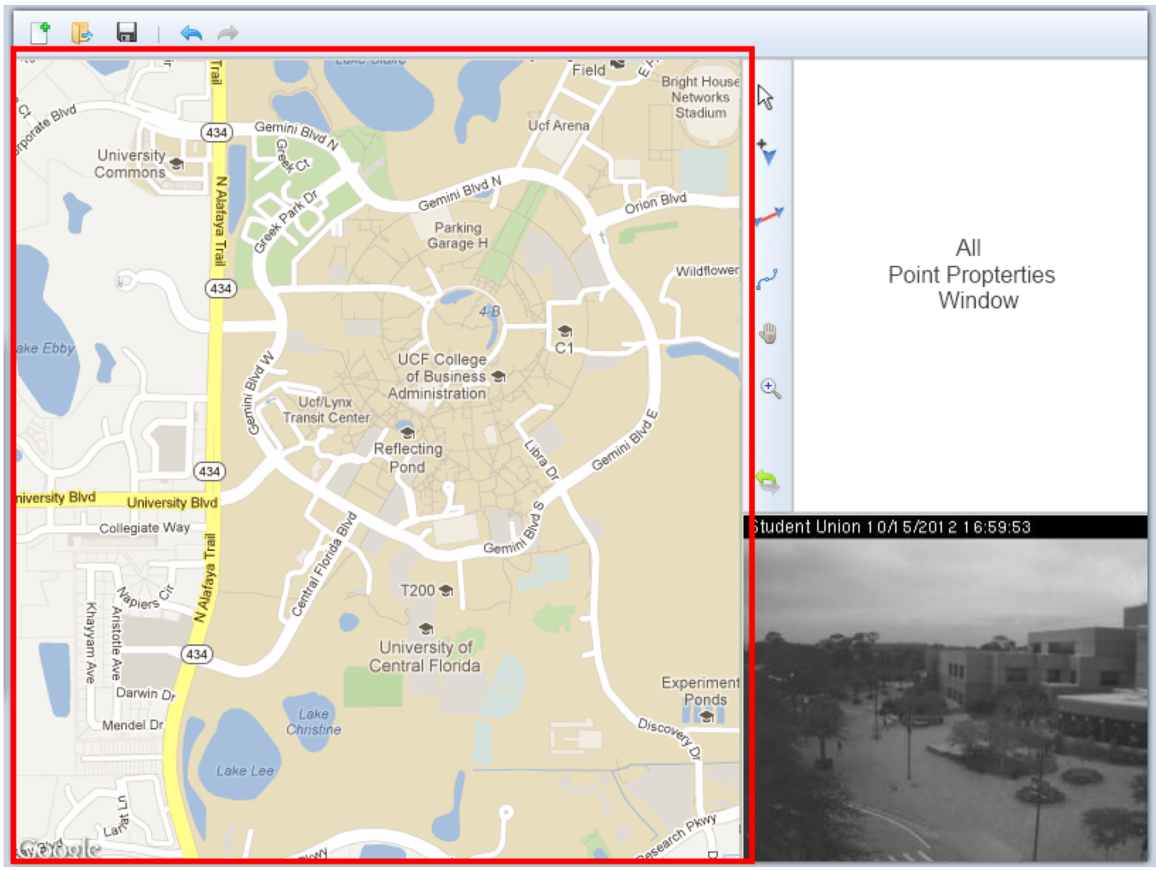


Figure 5.7.3 Map Display

The primary display is where the user will be able to define the path for the UAV using the secondary functions. This is also where "Open"-ed paths will be displayed and what "Save" will reference to save as a path.

5.7.3: Secondary Functions

The secondary functions can be seen in Figure 5.7.4. From top to bottom, they consist of "Select", "Add Point", "Connect ", "Free Form", "Pan", and "Zoom".

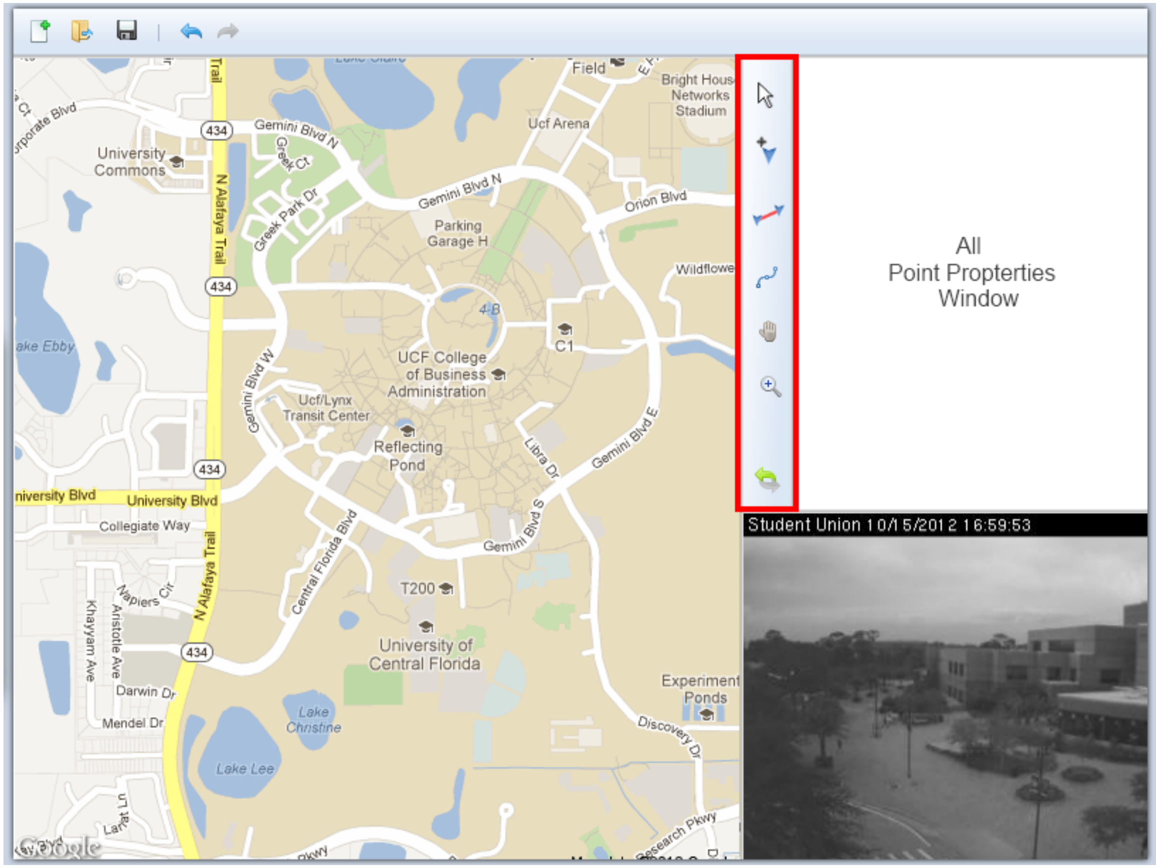


Figure 5.7.4 Secondary Functions

The “Select” function allows the user to select a point on the primary display to view details (which are displayed in the Properties display) about the point such as exact coordinates or time spent at the position.

Figure 5.7.5 Select Function

“Add Point” is the primary method of dropping new points on the map display which will be used to create a path. The user would simply click “Add Point” (figure 5.7.6) then click on the primary display to add a point. The first point on a new is marked by default as the starting location, which is where the UAV is currently docked, and cannot be removed.

Figure 5.7.6 Add Point Function

To connect points drawn on the primary display and create a route for the UAV the user would click “Connect” (figure 5.7.7) then click on begin selecting points to connect which will be connected in the order they are selected, e.g. the first point selected with “Connect” will automatically be connected to the starting point.

Figure 5.7.7 Connect Function

To quickly draw a path the user could select the “Free form” function (figure 5.7.8) and begin drawing a path. When drawing with free form, program drops points and connects them with fixed intervals. The tool draws while the mouse is pressed and when released the path drawn is closed to end the route back at the starting position.

Figure 5.7.8 Free Form Function

The “Pan” function (figure 5.7.9) allows the user, as the name implies, to pan the map to view parts of the map not displayed.

Figure 5.7.9 Pan Function

The “Zoom” function (figure 5.7.10) allows the user to change the scale of the map.

Figure 5.7.10 Zoom Function

5.7.4: Path Translation

When the user creates a point, they create a Point object within the program which contains coordinate variables which are set relative to the location on the primary display as well as the global coordinates. The coordinate variables are used to draw the path on the primary display, and the global coordinates are referenced when sending the path data to the UAV.

The point object also has a Pre-Point and Post-Point objects which are used to create a sort of doubly linked list between points which creates the path. “Points” will also have an x and y offset relative to the location of their origin to create a boundary for the user to click in to allow the user to easily select the point. As seen in the class diagram in figure 5.7.11

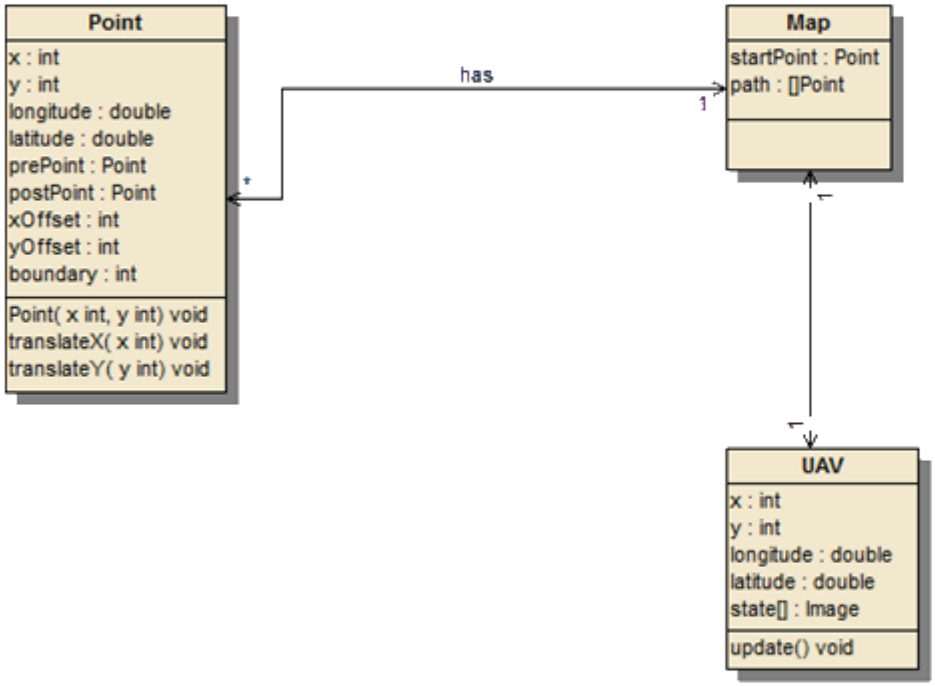


Figure 5.7.11 Class Diagram

The map object will contain a UAV object which will be updated with the current GPS data read in from the XBee updating the image 10 times per second. The image array of the UAV object will be cycled through as a function of time to give the animate the dot as if it were blipping (figure 5.7.12).

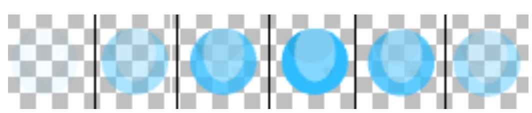


Figure 5.7.12 UAV Sprite Sheet

5.7.5: Application Initialization

When the application first starts it requests GPS data from the UAV. The initialization request is used to set the map accordingly to allow the user to begin drawing a path.

5.7.6: Communication

Since the client application is being written in Java and must be able to communicate with the XBee explorer interface we will be using a GNU licensed API to facilitate a quick and seamless development process for the application. This will allow the program to directly make calls to the XBee as opposed to running a HyperTerminal for interaction. The “xbee-api A Java API for Digi XBee/XBee-Pro OEM RF Modules” contains plenty of classes and methods that allow us to sidestep AT commands and easily integrate our GUI with the XBee transceiver.

The only data to be sent to the UAV will be route directives once the route has been established. This will simply be an array of GPS coordinates marked with a beginning, separator, and end message delimiters to be decoded and understood by the MCU.

5.7.7: Architecture

The application will be using a model-view-controller architecture. This will allow us to easily test and debug the three individually. The view will consist of all of the primary function buttons, the secondary function buttons, viewable images, frames and canvas which would be the primary and secondary display. The interactive objects in the view will invoke the view controller. The data model will handle connections to external sources such as the Google maps API and the XBee. The view controller will sit between the data model and the view responding to the actions performed on the view and making requests to the data model and modifying the view based on the response from the data model if necessary.

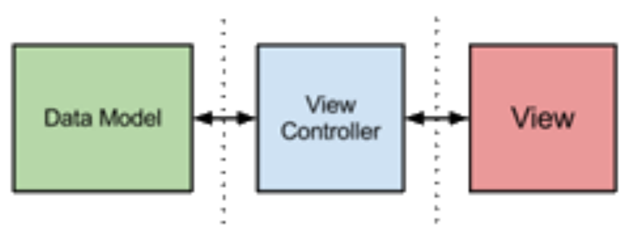


Figure 5.7.13 MVC Architecture

5.8: Charging Station

The charging station will be created with the intention of having the entire UAV and power station encapsulated into an all inclusive cube. It will need to be strong, water resistant, portable, and as light as possible. This will be accomplished by starting with an aluminum frame. It will have an aluminum cube skeleton support structure that has aluminum sheets as clothing. On the back of the cube, that specific aluminum sheet will be on a hinge that opens upwards. The reason for doing so is for maintainability purposes. The back panel will be the only access for the enclosed electronics. Having the design this way, will ensure less debris having the opportunity to enter the station. This will also provide the ability to have the remaining three sides to be riveted closed which will increase the overall durability and stability of the station.

For mobility purposes, the charging station will be able to roll around on two wheels with the operator pulling from an extending arm. This design is very similar to a rolling suit case or backpack. The original hope was to be able to have the entire station carried as backpack. However, due to weight restrictions it will be best left on the ground at all times. The universal battery alone weighs around 25 pounds and it was determined a possible hazard for a person carrying something that heavy. Even though the initial plan was not plausible, the new design's benefit will provide less chance for accidental damage due to dropping. It will also allow a wider array of operators being able to move the station with ease as opposed to someone with the ability to carry a minimum 50 pound load on his or her back.

Figure 5.8.1: Charging Station Weight			
Category	**Component**	**Quantity**	**Total Weight (kg)**
On board Electronics	Camera & Transmitter	1	0.02
	GPS	1	0.00945
	MCU	1	0.01
	Servos	8	0.068
	IMU	1	0.01
	Propeller Motor	1	0.125
	Transceivers	1	0.02
	LiPo Battery	1	0.122
UAV Structure	Frame	1	0.23
	Terminals	1	0.01
	Control Surfaces	1	0.01
	Propeller	1	0.02
Charging Station	Solar Panels	2	5.6
	Charge Controller	1	0.23
	Balancer	1	0
	Holding Battery	1	10.51
	Cigarette Outlet	1	0.0454
	Camera Receiver	1	0.75
	Laptop Charging Cable	1	0.1
	Current Monitor	1	0.135
	Fan	2	0.016
	Aluminum Frame	1	4.54
Total Charging Station Weight			**22.59585**

As one can see from the image above, the estimate for the fully assembled charging station will equate to roughly 22.6kg. For this reason only, the charging station will be left on the ground at all times.

The two 20 watt solar panels will be mounted on opposite sides of each other; on the bare sides without the extending roller arm or the hinged maintenance flap. On either side, both panels will be hinged from the uppermost edge of the station. They will be adjustable with a simple locking mechanism that will be able to lock the panels in multiple positions. The variety of adjustable positions will provide the ability to set the panels in various angels to capture the sun’s rays more efficiently.

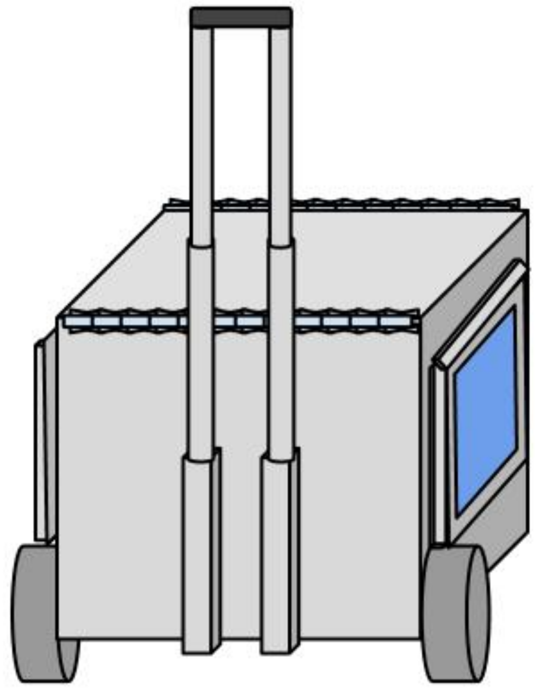


Figure 5.8.2: Charging Station

Connected inside the electronics housing of the station is what may be referred to as ‘the brain’ of the solar system. This device is referred to as a charge controller. The charge controller is responsible for taking in power from the solar panels and distributing it to the appropriate loads. The charge controller chosen has the ability to regulate power to two load lines and maintain proper charging of a 12V holding battery.

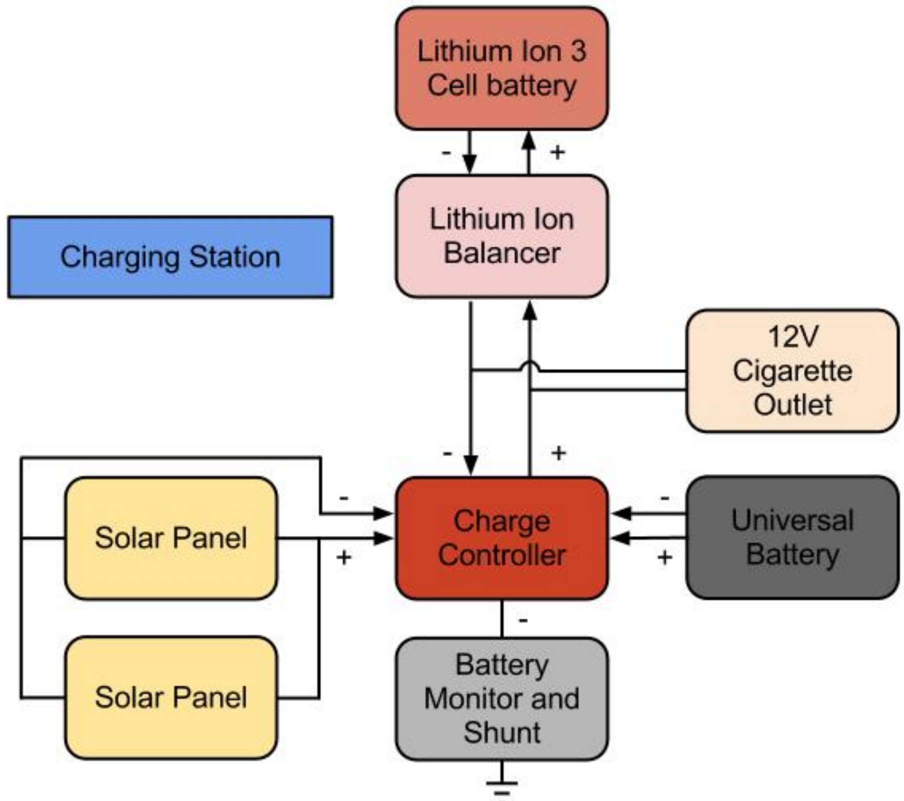


Figure 5.8.3: Charging Station Components

The first load line attached to the system, as you can see in the figure above, is the Lithium ion balancer. The Balancer will be described in more detail later. The Lithium ion balancer is what will be directly responsible for the specific charging needs of the UAV's Lithium Polymer battery. Without the balancer, undesired charging and possible damage can take place to the system as a whole. Also attached to this load line will be a camera receiver. The camera receiver was specifically designed to communicate with the on-board camera and then be able to transmit data to the laptop. It was designed to attach the camera receiver in parallel with the Lipo balancer in order to create a balance, or close to, of current draw from both load lines. Although current will take the path of least resistance, having both load lines drawing near the same amount of current seemed to be the most logical approach.

Figure 5.8.4: Radio Receiver

Image from http://www.chinadigitown.net

(Permission for use of photograph pending)

The second load line attached to the charge controller will be the laptop which has all of the software downloaded to it for the ability to control the UAV's mapping, destinations, and controls. The charge controller will have a water resistant, marine grade cigarette outlet connected directly to it. The outlet will be attached to the bottom edge of the charging station so that only the rubber sealed flap will be exposed to the outside.

Figure 5.8.5: Marine Grade Cigarette Lighter Socket 12VDC

Image from http://www.parts-express.com

(Permission for use of photograph pending)

This will allow for a removable second connection for the laptop. The laptop's second connector will have two male ends; one that plugs directly into the laptop and one that plugs directly into the cigarette outlet on the station.

Figure 5.8.6: Laptop Car Charger- Output 19V-3.42A with 5.5 x 2.1 Tip

Image from http://www.technooutlet.com

(Permission for use of photograph pending)

A separate line built into the charge controller is dedicated for 12 volt storage housing for the power collected from the solar panels. Generally, some sort of battery with a high capacity is chosen to perform this task. The Universal Battery Sla 12V 35Ah was chosen. This particular high capacity battery has many benefits; starting with the fact that it is spill proof, maintenance free, and has a low self discharge rate. It has been determined that discharging the rechargeable battery more than 50% can reduce its life. Therefore, at the point which the battery is fully charged, this battery can easily provide 210 watts (12 volts times 17.5 amps). This will provide more than enough to power the UAV system for the desired amount of time (three hours) with near zero contribution from the solar panels. This battery also has the capability to provide up to 12.1 amps which is more than enough compared to the maximum of 5 amps that are desired.

As seen in the figure below, when it is time to launch the UAV, the top flap is unhinged and the UAV is able to fly directly upwards and out of the vehicle.

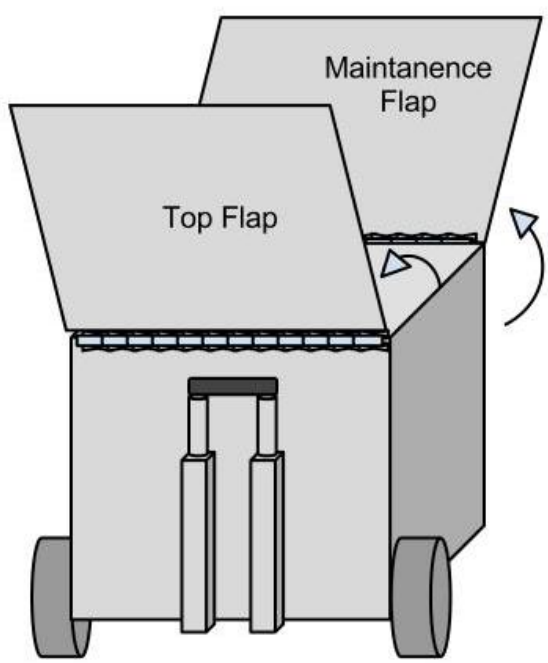


Figure 5.8.7: Charging Station Accessibility

The station will be divided into two halves; the top division will house the UAV and the bottom division will house the charging components and electronics such as the universal battery, charge controller and so on. The UAV will be controlled by a laptop which is charged by the station. There will be three main loads all charged by the two solar panels: the laptop, the UAV's power source, and the

UAV’s camera receiver. All load lines will have quick-disconnect cutoffs and inline fuses to promote safety as well as offer protection to the electronics.

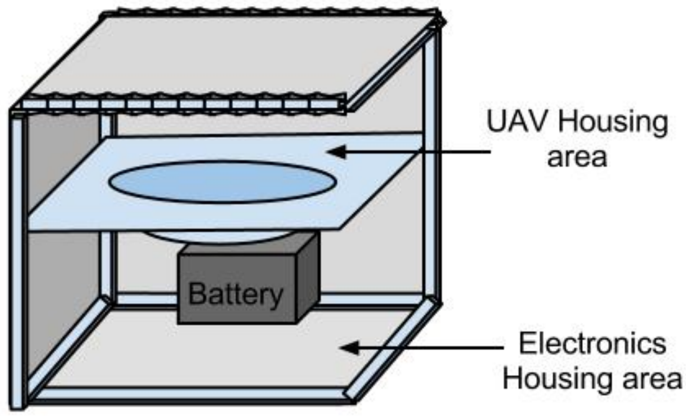


Figure 5.8.8: Charging Station Inner Divisions

By dividing the station into two distinct areas, there are several benefits to be mentioned. First and foremost, during presentation and public launch of the UAV there will be no exposed wires or electronic charging components at all. Looking downward into the station with the top flap opened, there will only be a sleek looking aluminum docking port for the UAV to charge its battery. Another added benefit is that during lift off and docking, there is minimal chance of damaging the electronics housing which may potentially occur due to a bad landing or liftoff.

Along both sides of the station there will be two poly crystalline solar panels attached. They will have the capability of adjusting into multiple locking positions to account for varying solar angels. It was designed such that the panels have the capability of locking at 0, 45, 90, and 135 degree angels. In the provided image below, the solar panels are shown in the 90 degree position.

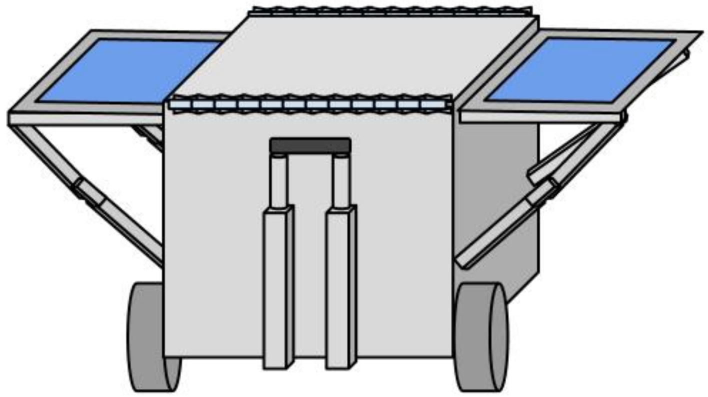


Figure 5.8.9: Solar Panel Extensions

The ability for the vehicle to self dock and charge itself provides an incredible benefit to the user. When the top flap has been unhinged and the UAV is in flight, the UAV housing area will have a bowl like surface. Along the perimeter of the bowl there exists the negative terminal as a metallic band. The negative terminal band will be separated from the aluminum frame with insulation and then connected to the negative end of the charge controller. Within the center of the bowl there exists the positive terminal. The potential of shorts in the design of exposed terminals will be addressed within the Charging Safety section of this paper.

5.8.1 - Charging Station Frame

For a variety of reasons the charging station frame has been chosen to be created solely out of aluminum and created from scratch. The specific skill set which the group is comprised of assured the team that a homemade frame would be manageable and a completely reasonable goal to achieve. Also, it was deemed that creating the station from scratch would ease the impact on the overall project budget which would leave room for add-ons later if desired.

The creation begins with a set of aluminum hollow bars that will be welded together in a cubic form. This will lay the groundwork for stability for the station as well as give a bare canvas to add clothing to later. For areas where reinforcement is necessary, added hollow bars will be added. The floor where the holding battery will be placed, for example, will be given extra reinforcement directly under where is will be placed due to the fact that the battery weights nearly 25 pounds.

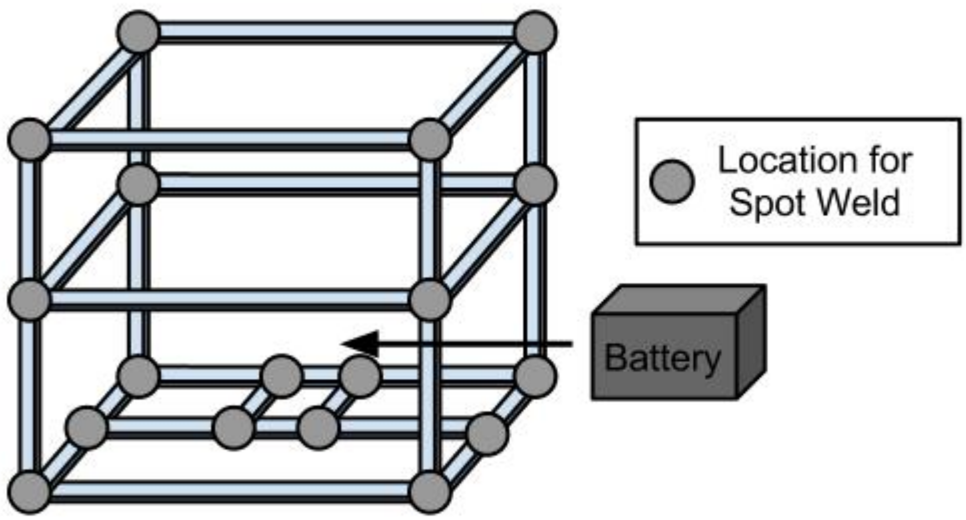


Figure 5.8.10: Charging Station Skeleton

The skeleton will allow for an easy implementation of aluminum covering sheets and a centered division for the two individual sections. The most difficult portion will be creating the bowl in the centered division out of aluminum. The current idea is to bend the aluminum and attempt to leave the center as one single piece. Reinforcements will be applied as needed.

The dimensions of the outer edge of the skeleton are 2ft x 2ft x 2ft. This will provide a six inch clearance for the UAV on either side to allow for ease in docking. The dimensions are also such for the possibility to encapsulate the electronics in the lower portion. The electronics and UAV dimensions have been summarized below.

Figure 5.8.11: Component and Charging Station Dimensions		
Category	**Component**	**Dimension (mm)**
On board Electronics	Camera & Transmitter	23.0 x 25.0 x 20.1
	GPS	38 x 38 x 7.8
	MCU	10 x 10 x 2
	Servos	21.8 x 11 x 23.2
	IMU	38.1 x 25.4
	Propeller Motor	71 x 43 x 43
	Transceivers	24.38 x 32.94
	LiPo Battery	20 x 50 x 65
UAV Structure	Frame	305 x 305 x 305
	Terminals	-
	Control Surfaces	-
	Propellor	241
Charging Station	Solar Panels	550 x 350 x 25
	Charge Controller	152 x 55 x 34
	Balancer	38 x 90 x 14
	Holding Battery	195 x 131 x 181
	Cigarette Outlet	-
	Camera Receiver	140 x 70 x 31
	Laptop Charging Cable	-
	Current Monitor	114 x 114 x 35
	Fan	30 x 30 x 10
	Aluminum Frame	610 x 610 x 610

Figure 5.8.12: Total weight contribution breakdown for the Aluminum in the Charging Station			
Shape	**Quantity**	**Dimensions (in)**	**Aluminum (in^3)**
Side panel	6	24 x 24 x 0.1	28.8
Frame hollow bar	12	24 x 4 x 0.5	57.6
Centered division	1	24 x 24 x 0.1	4.8
Hinges	2	24 x 0.2 x 1	9.6
Fasteners & rivets	20	0.2 x 0.2 x 0.1	0.08
Total Aluminum			100.88
Density of Aluminum			0.098 lb/in^3
Total weight			9.9lbs

5.8.2: Solar Panels

There are several options when it comes to choosing an appropriate method of renewable technology. For a project of this specific size, portability needs, and location, it was chosen to go with utilizing the power of the sun. In this process of harnessing the sun's rays, there are several factors to give various weights to. The main operating location for the UAV project has been determined to be the most influential factor in deciding which type of solar panel to choose for this specific green project.

In Florida, it is no secret that there is a vast availability of sun. This gives a start to an ideal environment for solar power. However, in Florida it is very common for temperatures to reach 90°F or more. This presents a problem to be addressed for efficiency and reliability of the solar panels. The most common array of solar cell technologies available in today's market is thin film, monocrystalline, and polycrystalline. They each have their own benefits as far as efficiency, dimension limitations, and cost.

The main difference between the three options is the grade of the silicon used in the panels. Monocrystalline panels are created from silicon ingots and constructed into wafers. This is the highest purity of all the Silicon panels, and therefore has the highest efficiency per cell; typically 15-20%. Another benefit is that standard monocrystalline panels come with a 25 year warranty, which can be very intriguing to any homeowner. However, monocrystalline panels are the most expensive to manufacture out of the three and creates wastes some of the original silicon due to the way it is cut. Polycrystalline panels are slightly less efficient, typically 13-16%, but the way in which the cells are created is much easier and more cost efficient. Furthermore, polycrystalline panels are nearly comparable in temperature reliability and therefore one cannot be ruled out on that basis alone. Finally, thin film technologies operate at around 9% efficiency and are easier to mass produce. However, thin film they require a large amount of space in order to produce a substantial amount of power.

For this project, it was chosen to go with polycrystalline solar cells. Polycrystalline panels have been proven to operate nearly the same efficiently at higher temperatures than its monocrystalline counterpart and much more efficient than thin film. Cost was a large determining factor in choosing which solar technology to go with. With value and temperature efficiency in mind, polycrystalline was the clear winner.

Figure 5.8.13: Polycrystalline Solar Panel

Image from http://www.gogreensolar.com

(Permission for use of photograph pending)

After choosing the appropriate power supplier to the system, it is time to determine the amount of power needed for the panels to output in order to support the systems electronics. The power system, on the most basic level, is going to have 5 loads attached; the Laptop at 41 watts, the Lithium Ion UAV battery at 22.2 watts, the camera receiver at 6 watts, and the two cooling fans at 0.84 watts each. The loads require a total of roughly 65 watts of power. With a daily run time estimate for the UAV of 3 hours, the power necessity would be the load wattage multiplied by the number of run time hours. This turns out to be 65 watts times 3 hours which is 195 watts.

A solar panel rated a specific wattage will only reach that amount if it is provided enough sun. In Florida, it is estimated that there are roughly 4 optimal sun hours for solar power. If this is extrapolated out, there are 195 watts needed. When divided by 4, this yields 48.75 watts per hour of solar power needed for the system to operate minimum 3 hours. Therefore, in order to ensure enough power and to account for some possible longer run times, it was chosen to provide a total of 40 watts of solar power which is believed to be able to easily store 160 watts of power (40 watts times 4 hours for full sun exposure). Since the amount of power collected and distributed per day is 195 watts, there seems to be a shortage of power of 35 watts. Although this seems to be an issue, the original calculations do not include partial sun exposure nor does it include power previously stored within the holding battery itself. Therefore, it has been determined that the 40 watts solar system will be enough to power the UAV and charging station.

For integration purposes, it was chosen to go with two 20 watt polycrystalline panels. During optimal solar hours these panels will provide 40 watts. The obvious question is why one would choose two 20 watts panels and not simply one 40 watt panel. Although the two panels are cumulatively 53% more expensive than the single 40 watt panel, the single panel is too large in dimensionally to integrate into the proposed charging station. The main goals of the charging station are portability, stability, and a sleek design.

5.8.3: Charge Controller

One of the main difficulties in solar power is regulating the inconsistent power out. The easiest way of taming this issue is adding an appropriate charge controller connected directly to the output of the panels. Charge controllers are a necessity in the regulation of solar power. There are a wide variety of controllers that have a variety of features. Various options and features such as LED indicators and low voltage cut offs are offered. However, the main purpose for each and every one charge controller is to safely provide a steady power stream to the in line electrical components in the system.

The charge controller chosen for this project it the Morningstar SunSaver SS-20L-12V w/LVD, 20 Amp 12V Charge Controller. This will be able to bring an array of benefits to the system.

Figure 5.8.14: SunSaver Charge Controller

Image from http://www.gogreensolar.com

(Permission for use of photograph pending)

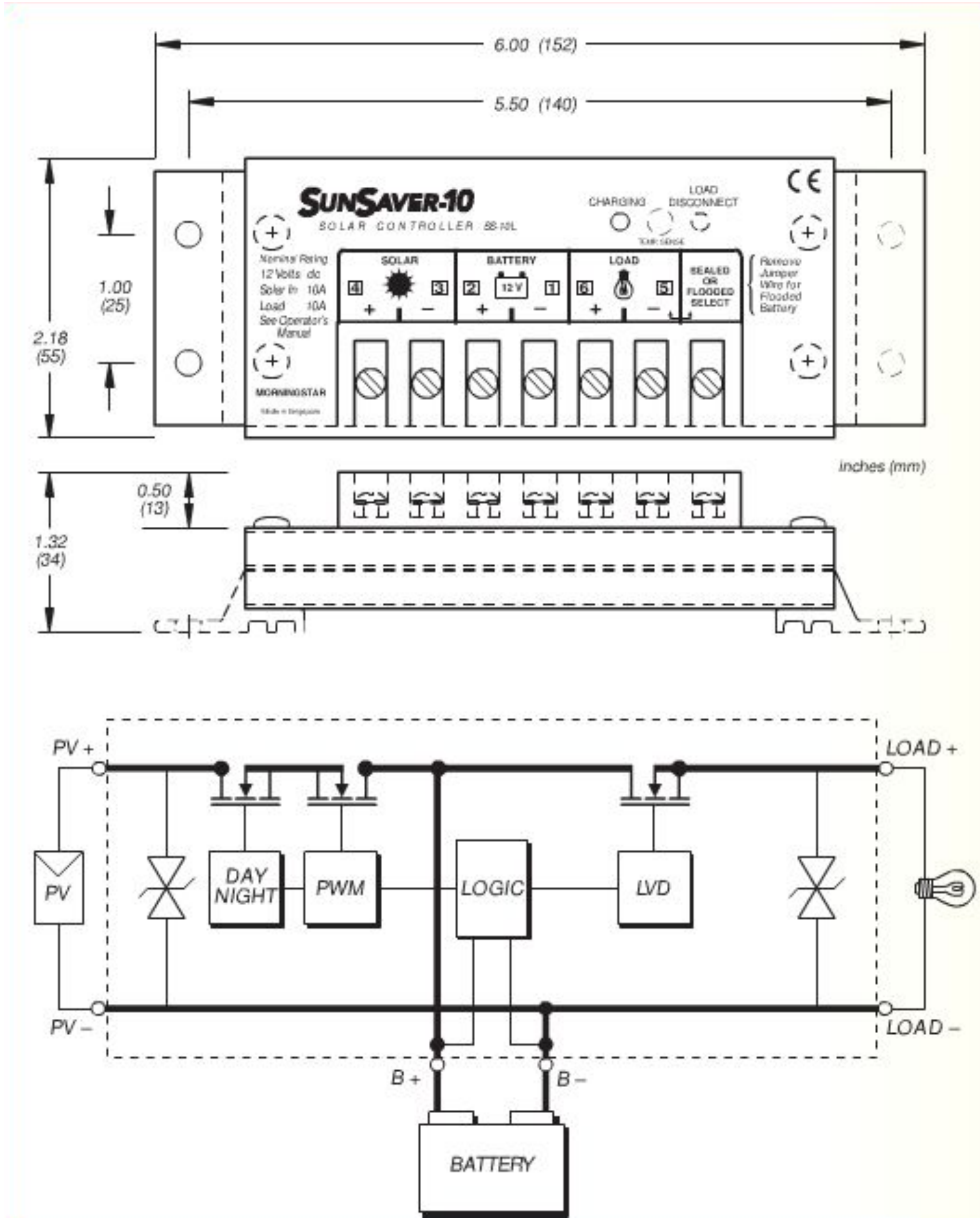


Figure 5.8.15: Charge Controller Schematic

(Permission for use of schematic pending)

First of all, this charge controller will provide a constant 12 volts which is what is to be desired for the type of universal batter that will be used. It also has the capability to run a maximum of 20 Amps through itself. The culmination of components in the power system is 8.57 Amps. Therefore, there is a surplus of nearly 11.5 Amps in this design which has been determined acceptable. The main functionality of the device in the design is to provide a steady stream of power to the universal battery as well as have both loads which will be directly connected. This makes things much simpler as far as connectors, wires, and regulating power to the loads; the controller will handle all of these tasks.

5.8.4 - Universal Battery

The heart of every solar charging system is some type of battery large enough to store the charge collected by the solar panels. There is a rough ratio generally used between the amount of watts generated by all of the solar cells and the size of the battery. If there is not enough storage in the battery compared to the solar cells, then the charge controller will be in the “off” position for too long of a duration due to the fact that no more charge can be help within the battery. This yields a decrease in efficiency of the overall design. On the converse, if too large of a battery is chosen then it is a complete waste of money to purchase something unnecessary. Also, an increase in battery capacitance yields a heavier battery. In this design weight is definitely a concern. Therefore, it must be determined the appropriate size of battery to add the best value to the project. In previous sections the appropriate battery size had been determined to be a 12V 35Ah battery.

The second issue to consider is the type of battery to chose. There are four basic type of solar storage batteries to consider; RV, Flooded, Gel, and AGM. They all have specific benefits and reason for choosing. The benefits and downfalls are listed below.

Figure 5.8.16: Holding Battery Variations

Battery Type	Relative Price	Amount of Maintenance	Unique Drawback	Unique Benefit
RV	Various	Low	Low life time	Economic
Flooded	Medium	Low	Release gas	Good lifetime value
Gel	Medium	Zero	Low lifetime at high temp	No gas release
AGM	High	Zero	Expensive	High efficiency & rigid to vibration

After careful consideration, it was determined to choose an AGM (Absorbed Glass Mats) Universal Battery. The reasons for choosing this type of battery are described in the benefits of this technology listed below.

- Spill proof
- Non hazardous
- Low self discharge rate
- Withstand vibration and shock well
- No maintenance
- Recombining efficiency typically +99%

As far as actual charging is concerned, the charge controller will provide power to the universal battery at 12 volts and at the rate at which the solar panels provide current, until the battery has reached maximum charging. At this point, the controller stops the flow of electricity until it becomes discharged and then charging will resume. A constant monitoring of the universal battery's output voltage will be going on at all times to ensure the battery does not fall below 50% capacity. When it does, the controller will disconnect the loads until it reaches a safe point from being recharged to resume its power distribution responsibilities.

Figure 5.8.17: Universal Battery Sla 12V 35Ah

Image from http://www.ecomelectronics.com

Permission Pending

5.8.5 - Lithium Ion Battery

The UAV drone functionality and practicality has been determined to have one main bottlenecking limitation. This limitation is the amount of time it can remain in the air on one charge. Without air time longevity, there are severe consequences the UAV will be facing. Depending on the length of air time, the UAV may or may not be able to complete its designated mission. Simply put, the longer the UAV can fly the more desirable and effective it ultimately will be.

There are several factors that come into play when it comes to increasing total flight time. Everything from major factors such as the weight of the vehicle or shape of the airframe can dampen the length of flight; even something as small as the climate can mean the difference between whether or not the UAV can surmount the flight time desired.

Figure 5.8.18: Total UAV Weight

Category	Component	Quantity	Total Weight (g)
On board Electronics	Camera & Transmitter	1	20
	GPS	1	9.45
	MCU	1	10
	Servos	8	68.00
	IMU	1	10
	Propeller Motor	1	125.00
	Transceivers	1	20
	LiPo Battery	1	122
UAV Structure	Frame	1	230
	Terminals	1	10
	Control Surfaces	1	10
	Propeller	1	20
Total UAV Weight			**654.45**

For a simple example, if the UAV has a been set up to complete perimeter surveillance about a specific point, the difference between completing a 1/10 mile radius and a 1/5 mile radius is only an extra 7 minutes of flight time. This is assuming the UAV is traveling at a constant 10 miles per hour in a perfect circle. With the additional tenth of a mile of surveillance, it increases the area of the monitored region by 50% which is equivalent to 0.628 square miles.

As per the initial desired qualities of the UAV, the goal as far as flight time needed to be achieved is 15 minutes of flight time. With this amount of time, the

UAV will be able to complete the aforementioned 1/5 mile radius at 10mph. With this in mind, the next step was to look into energy needs in great detail. The current found to be necessary for the UAV to make a full and complete flight is 8.57 Amps. With this current draw in mind, a battery of at least 2 Amp hours was desired in order to be close at achieving the initial goal.

This being the case, the battery chosen was the 3 cell 11.1V 2000mAH Lithium polymer battery; specifically, the Thunder Power RC Pro Lite MS 16C 2000mAh 11.1V 3 Cell LiPo 3SPL 2000 Lipo Battery. The benefits of this battery are in its dimensions, weight, and having a max burst current of 60A. This battery is small enough to be able to fit directly under the propeller motor without adding too much downward drag. Also, the capacity to weight ratio was better than alternatives of the same price range.

Figure 5.8.19: Thunder Power RC Pro Lite MS 16C 2000mAh 11.1V 3 Cell Battery

Image from http://www.rctoys.com

(Permission for use of photograph pending)

5.8.6: Lithium Balancer

This Lithium ion battery is to be the first load connected to the charge controller. However, do to the specific charging needs of the Lithium polymer battery, the battery must be charged by what is referred to as a balancer. Each type of lithium polymer battery has a corresponding balancer that provides accurate, safe, and complete charging. There are several things that can go wrong when it comes to charging the Lithium battery without a balancer. Unequal charging can ensue if a standard charger is connected to the battery due to the fact that this specific battery has multiple cells that must be independently charged. Each cell needs a charge of 3.7 volts to be considered fully charged. With all three cells fully charged, the 11.1 volts will be achieved. Also, each cell must receive a specific amount of current and a specific amount of voltage that varies with respect to time. The following graph is a rough representation of the varying voltage and current flows that must be provided for a proper lithium ion battery charge.

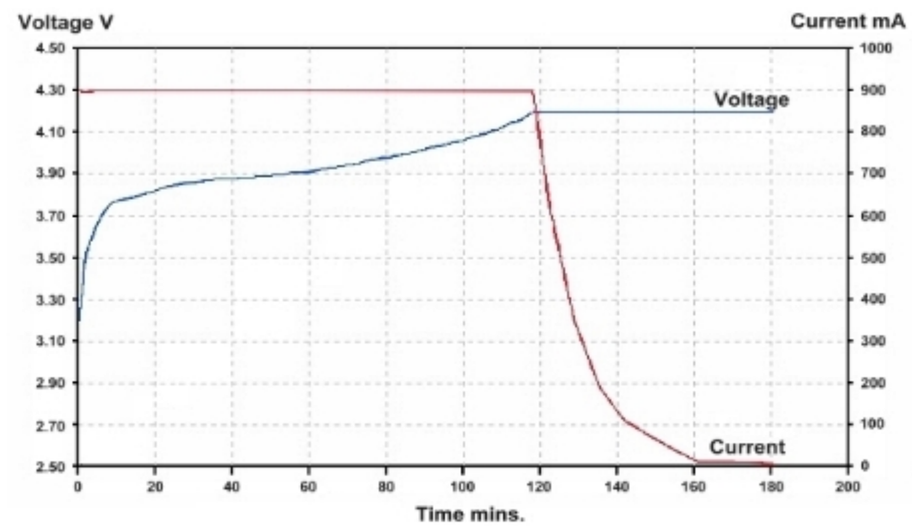


Figure 5.8.20: Typical Lithium Ion Polymer Charging Graph

Image from http://www.ibt-power.com

(Permission for use of data pending)

The balancer chosen for this charging station is the Thunder Power RC 205V Balancer for 2-5 Cell Li Poly Batteries. It is compatible with multiple Lipo batteries and has an auto detect feature to identify the amount of cells the battery has. Other features include check and display battery conditions with LED visual indicators, low voltage discharge warnings, over voltage warnings, full charge alert, and red LED display for each cell in bleeding and balancing indicators. All of these features will provide accurate and safe charging of the Lithium battery.

Figure 5.8.21: Thunder Power RC 205V Balancer

From http://www.rctoys.com

(Permission for use of photograph pending)

5.8.7 - Charging Station Safety

Safety is a top concern for this UAV project. There is never an excuse to compromise safety. This being said, there will be appropriate fuses in line with any battery connection. Also, there will be quick disconnects in line with both batteries and from the solar panels to the charge controller. Furthermore, in an attempt to reduce potential electric shorts and possible electrocution all terminals and connectors will have rubber heat shrink covering any exposed metal. There will also be two fans installed in the charging station for heat dissipation where the electronics will be located. The fans will be installed where one sucks air inwards and the other on the opposite side of the station blows air outwards. These will keep fresh air circulating through and help keep the station cool. Finally, during testing a safety bag will be used to place the lithium battery in to reduce damage and injuries due to accidental battery rupture.

Figure 5.8.22: DC Cooling Fan

Image from http://www.vpi.us

(Permission for use of photograph pending)

Over charging can lead to puffing of the battery and an eventual fiery eruption. The chance of this possible hazard can be decreased be using an appropriate balancer. However, since there is a chance of this happening there are some specific precautions that can be taken. A safety bag is recommended to be used at the onset of every charge. Safety bags are specifically designed to protect damage to nearby persons and electronics from a battery changing malfunction. During testing of the UAV, a safety bag will be used at all times. The safety bag will only not be used the UAV is fully assembled with the battery inside the vehicle and all persons being a safe distance away.

Figure 5.8.23: LipoSack Fire Retarding Lithium Polymer Battery Charging & Storage Safety Bag

Image from http://www.rctoys.com

(Permission for use of photograph pending)

In order to visually monitor the stability of the system during activity, it has been decided to equip the charging station with LED visual read outs and current monitors in line. When there is an increase in real time visual read outs of what the system is doing without having to do manual testing, the safety of the people participating increases. A battery monitor and shunt will be used for visual inspection of the charging path way to make sure no unwanted behavior is occurring at any point. This read out will be able to be seen from the exterior of the station. The great thing about this monitor that has been chosen is that the monitor has low self power consumption. The monitor only draws a total of 0.015 Amps. Also, the LED indicators from the charge controller will be visible from the exterior of the station. This will indicate to the user whether or not the charging is occurring, the holding battery is at low voltage, or the solar panel's contribution has been disengaged.

Figure 5.8.24: Morningstar Remote Meter, RM-1

Image from http://www.gogreensolar.com

(Permission for use of photograph pending)

With the UAV being able to dock and charge itself, a problem is presented in the possibility of shorting the onboard UAV electronics or possibly shorting the terminals of the charging station itself. This problem has been addressed on the UAV with the inline installation of diodes to prevent a potential closed loop between its own terminals. On the charging station however, the terminals have been insulated in order to prevent shorting to the aluminum station itself.

6.0: Project Prototype Construction

The moment when all the research, design, and testing come together is the building of the prototype. For the construction of the prototype there are three different major physical sections that make up the entire project: the AFSD, the docking station, and the user's computer. All three of these sections are tied together in one way or another to make the entire system complete.

The first major part for the assembly of the prototype is the actual vehicle itself. As mentioned in the section describing the vehicle frame, there are two different parts to the frame: interior and exterior. The interior frame is responsible for holding and storing all the electrical components inside of it. The interior frame creates a visual pleasing frame that all the electrical parts and wires can fit inside of while at the same time creating a smooth surface on the outer part of the interior frame to aid in aerodynamics. All the different electrical components are all connected to microcontroller except for the camera and powered by a lithium polymer battery. The GPS and IMU are connected to each other via a FTID 3.3V cable. The connection between the two allows for the two to compare data about the vehicles position and velocity. The GPS has its own build in patch antenna that allows it to receive its latitude and longitude coordinates. Both the GPS and the IMU are connected to the microcontroller providing input data.

Also connected to the microcontroller is a transceiver so that the coordinates for the pre defined path set for by the user can be wirelessly transmitted back and forth from the microcontroller. The microcontroller is responsible for calculating the difference between the actual position of the vehicle and the route defined by the user and for interpreting the vehicles stability. One output form the microcontroller is to go to the 8 servos so that the control surfaces can be rotated, stabilizing the vehicle. Another output of the microcontroller is connected to the motor which then will spin the propeller faster or slower depending on the data given. The last electrical component that is in the interior frame is the camera. The camera is not connected to the microcontroller because it has a receiver attached to it so sends a live feed back to the user's computer. The camera will be located inside the interior frame like the rest of the electrical components but there will a hole in the bottom of the frame for the camera to get a visual of the ground below.

The interior frame was built with holes around the sides, top, and bottom so that wires could get to parts that were located on the interior frame. Along the sides of the interior frame are mounts for the 8 servos to be attached to. On top of the frame is where to motor is to be placed with the propeller on top of it. With this design for the prototype enough space is left for the propeller to fit inside the exterior frame. Mounted on the servos will be the 8 different control surfaces, 4 upper and 8 lower, as specified in vehicle frame section. The last piece to be put together for the vehicle to be complete is the exterior frame shell. The pieces of the exterior frame are designed to clip together and have mounts for the ends of the control surfaces.

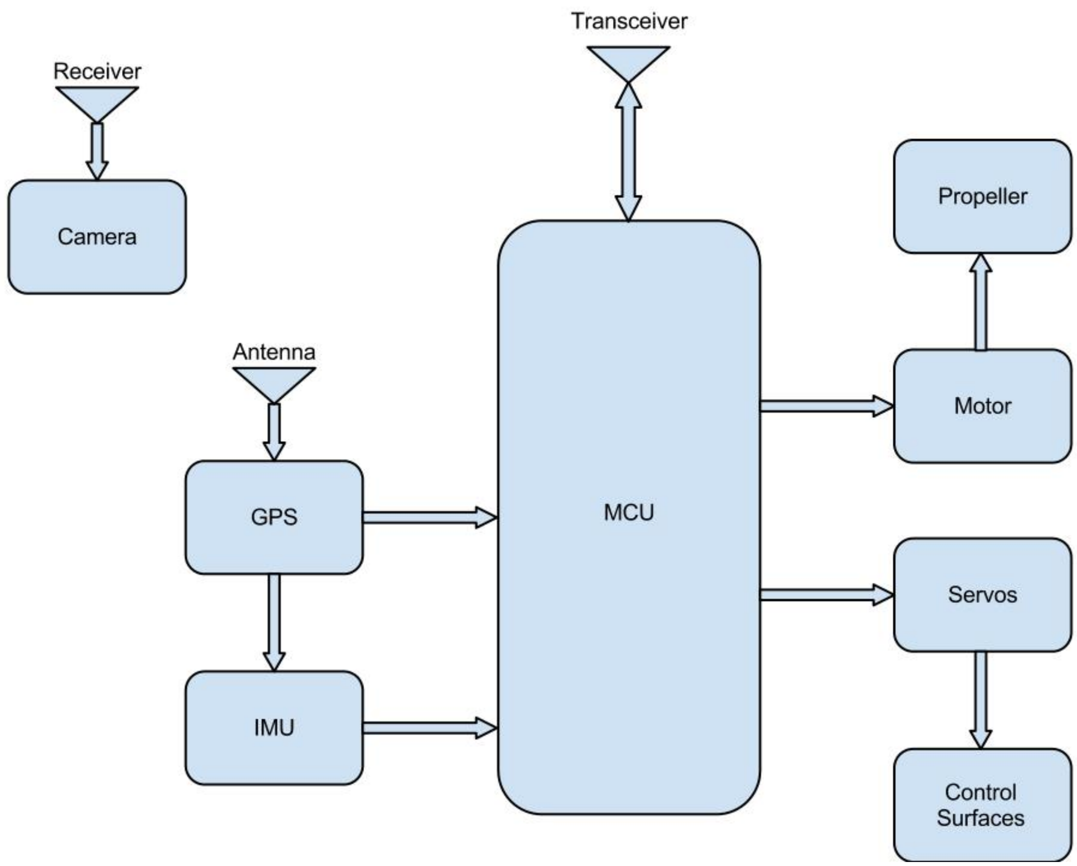


Figure 6.1: Vehicle Electrical Components Block Diagram

The next part of the project prototype to be constructed will the the docking station for the vehicle to land in and recharge. The docking station itself will a constructed for the AFSD to land in and recharge. For this to happen a parabolic bowl shape will be build to allow the vehicle to land securely in the docking station. Probably the main function of the docking station is ability to recharge the vehicles Lithium polymer battery.

In order for the vehicles battery to be recharged a power supply must be build into the prototype docking station. The design for the docking station requires the power supply be based on solar panels. Two solar panels will be bought and used as a power supply. One the top docking station there will have to be build two angled mounts for the solar panels to rest on. The two solar panels will be connected to a charge controller that will be build into the docking station underneath the vehicle landing space. In order to store the power from the solar panels a battery will have to be bought and connected to the charge controller. The battery will also be mounted underneath with the charge control. In order to recharge the battery on the vehicle without actually having remove the battery, a positive and a negative terminal strip will build into the parabolic bowl shaped landing surface that the vehicle rests in.

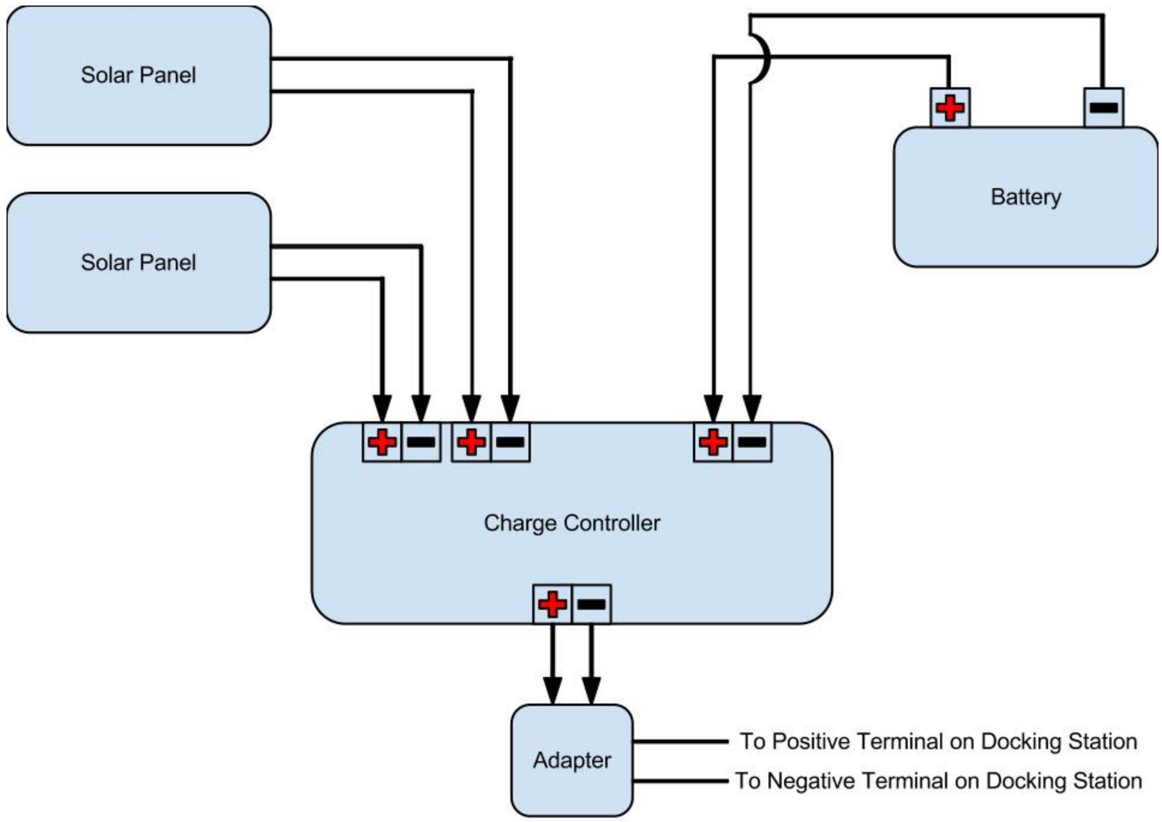


Figure 6.2: Docking station Power Supply Block Diagram

The last part for the prototype construction to be complete is the use of a computer for the security personnel. From this computer the security personnel will be able to monitor the vehicles live video and select the route for the drone to navigate. In order for the security personnel to be able to perform this task though a visual application must be constructed. Below is a prototype picture of what the user application will look like.

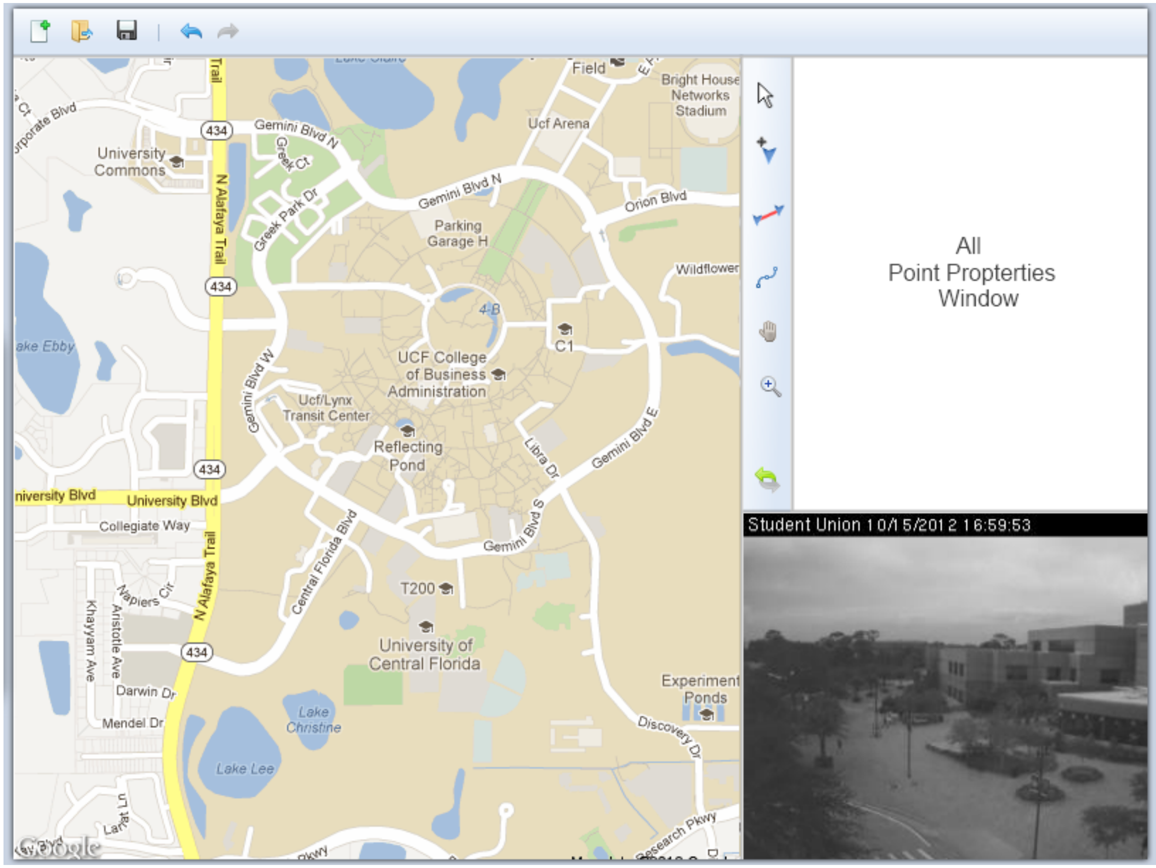


Figure 6.3: User Application Prototype

7.0: Prototype Testing

Since our project has so many different components and subsystems we intend to test each individually prior to integration. Performing modular tests prior to system wide tests will allow us to resolve issues that would otherwise be impossible to pin point. The testing process will be divided by components to ensure each is behaving as expected. Testing our components will also give us a strong understanding of the actual limitations of our devices relative to the packaged limitations. Subsystems will be tested after modular tests such as the wireless communication system, microcontroller and motor as well as servo testing. Each of these subsystems can be tested independently.

We will use the following procedures and conditions of success to determine if the system/component being tested meets our requirements. If the test results do not meet our expected results, then the system will be analyzed to determine the reason for the failure, remedied and subjected again to the test. Failure will result in repeating the cycle until the results meet our expected results.

7.1: Operational Constraints

Due to the number of complex systems being integrated it is advantageous to limit the factors that could detrimentally affect the operation of the entire system. Since the end result is just a prototype and it is not meant to be a final product ready for distribution, this allows the group the freedom to operate and test the system in a relatively stable and predictable environment. When the project is successfully operating under these conditions, then more variables can be introduced.

In the route plotting application the map and the route drawing will be separated into two layers which reside on top of each other; the map drawing underneath the route drawing. Since the longitude and latitude must be calculated depending on the location clicked on the screen it is imperative that the map drawing corner coordinates be known to properly make the translation. It is also the responsibility of the user to ensure the route distance does not exceed the flight capability of the UAV (e.g. a route with a distance well beyond the estimated total flight time given current battery power).

Additionally, environmental concerns such as wind and rain will negatively affect the UAV's ability to fly properly given the size and weight of the vehicle. Another issue would be a cloudy sky not allowing for the GPS module to receive a signal from any of its satellites in orbit. This would likely cause the vehicle to hover until a connection is re-established and if a connection is not made in time for the vehicle to make it back to the docking station with the remaining battery life it would have to perform an emergency landing to avoid falling out of the sky and damaging equipment.

Another constraint is a result of the camera receiver transmission distance limit. If the vehicle exceeds 350 ft. from the docking station, video feed will be lost. It is left to the discretion of the user to avoid exceeding this distance.

7.2: Servo Control

7.2.a: Servos

Purpose: To make sure that the servos perform smooth movement

Procedure:

1. Connect servos with power supply
2. Vary input voltage (4.8-6V)
3. Check the speed of the servos, which should increase as input increases

4. Cut off the power and see if it orients itself back to the center point

Expected Outcome: Servos respond to various inputs

Conditions of Success:

- Smooth movement with various inputs
- Position varies based on duty cycle

7.2.b: Piccolo - Servos

Purpose: To control the servo using the MCU Servo Library to examine if the coding is correct

Procedure:

1. Connect Piccolo board to a PC
2. Connect Piccolo to the servos using wires
3. Make sure the servos are ground on the board
4. Power on the microcontroller
5. Send sets of coordinates to from PC to Piccolo
6. Observe the movement of servos

Expected Outcome: Some servo movements

Conditions of Success:

- Servos move to the designated position
- Servos stay at the position as long as the given rate

7.2.c: Linear Control System

Purpose: To test the linear controller and check if it can properly handle the motor and servos to maintain stability.

Procedure:

1. Connect development board to computer.
2. Load control system to MCU dev board.
3. Disconnect servos and motor.
4. Connect PWMs to oscilloscope.
5. Vary sensor inputs (tilting, move off course)
6. Record the output data

7. Calculate the error percentage and see if it is acceptable. If not, reprogram the control system to more closely match acceptable behavior.

Expected Outcome: Control system gives responses to input

Conditions of Success:

- Motor output properly attempts to maintain altitude
- Closed loop output value is close to its current position value
- The error percentage is small
- Servos signals properly handle tilting and redirection
- Control system is responsive

7.2.d: Linear Control System - Servos

Purpose: To minimize error and avoid overshoot using our linear control system and ensure components are can handle expected loads.

Procedure:

1. Connect Piccolo board to a PC
2. Load control system to MCU dev board.
3. Connect servos and Motor to respective PWM modules.
4. Vary sets of coordinates
5. Observe servo and motor movements and RPM
6. Vary sensor states
7. Observe servo and motor movements and RPM

Expected Outcome: Servos and motor responsive to control system.

Conditions of Success:

- Smooth motor and servo control
- Servos move to appropriate positions based on sensor data
- Bottom servos orientate themselves back to center position when UAV is upright
- Motor responsive to altitude required

7.3: Wireless Communication

7.3.a: XBee – XBee Module Interface

Purpose: To make sure that the local XBee wireless module is successfully connected and communicating with the remote XBee module and they are compatible, also to check that the API is reliable and compatible.

Procedure:

1. Verify configuration settings on each module
2. Connect each XBee to separate computers.
3. Attempt to send data using HyperTerminal from local to remote
4. Check for response on remote module
5. Repeat for API

Expected Outcome: Payload is transmitted

Conditions of Success:

- The modules are compatible and acknowledge transmission
- API is reliable and compatible

7.3.b: Microcontroller – XBee Module Interface

Purpose: To make sure that the XBee wireless module is successfully connected to the microcontroller and those two units are compatible.

Procedure:

1. Connect Piccolo microcontroller to PC
2. Connect XBee to MCU as seen in figure 5.2.1
3. Before powering on the wireless module, make sure it is connected to 3.3 volts shared with the MCU and grounded to the board.
4. Power on the microcontroller
5. Open the IDE software (Code Composer Studio)
6. Make sure the software finds the module

Expected Outcome: The software recognizes the module

Conditions of Success:

- The module and Piccolo are compatible
- The IDE recognizes the module
- User can program communication between the two

7.3.c: Microcontroller – Local Application

Purpose: To use the protocol to transmit data and see if there is anything lost or corrupted data occurs

Procedure:

1. Connect MCU board to a PC
2. Verify wireless module settings
3. Power on both subsystems; local and remote
4. Open and run the IDE software on the PC
5. Open and run the local application on the PC
6. Send a simple digital data from the application to the microcontroller
7. Observe if both input and output appear on the PC

Expected Outcome: Microcontroller responds to application data

Conditions of Success:

- Data is not corrupted or lost through transmission
- Microcontroller receives data signal from the application with minimal delay
- Microcontroller recognizes directives and GPS coordinates respectively

7.3.d: Camera – Receiver

Purpose: To make sure that the camera subsystem can properly transmit video feed from the wireless camera and is displayed with adequate quality without any significant delay.

Procedure:

1. Connect receiver to the AV display.
2. Power on both camera and the receiver.
3. Move camera around and examine video feed to see if there is any significant delay.

Expected Outcome: The video stream coming from the camera is clear and has no significant delay

Conditions of Success:

- Video quality does not show excessive delay or rolling shutter effect.

7.4: Navigation Testing

7.4.a: IMU Testing

Purpose: To make sure that ArduIMU V3's accelerometers and gyroscopes are able to properly read changes in orientation and gravitational forces and properly communicate with servos so control surfaces can fix the vehicle orientation.

Procedure 1:

1. Connect the IMU to a PC, using a serial adapter, to see the serial output in either ASCII or binary.
2. Connect a voltage source to the IMU
3. Begin rotating the IMU in the X, Y, and Z directions along one axis at a time.
4. Rotate in X, Y, and Z direction all at same time.

Expected Outcome: The IMU's gyros and accelerometers correctly pick up the directional movement of the IMU and display the yaw, pitch, and roll measured in degrees on the computer.

Conditions of Success:

- When the IMU is rotated in the X, Y, and Z directions independently the display on the computer for yaw, pitch, and roll shows a major degree change in just on direction at a time.
- When the IMU is rotated in around all three axes the display shows so accordingly by the measured degrees of yaw, pitch, and roll.

Procedure 2:

1. Connect the IMU to one of the servos.
2. Supply a voltage source to the IMU
3. Begin rotating the IMU in the X, Y, and Z directions along one axis at a time.

Expected Outcome: The IMU's gyros and accelerometers correctly pick up the directional movement of the IMU and cause the servo to rotate to compensate for the change in orientation.

Conditions of Success:

When the IMU is rotated in the X, Y, and Z directions independently the servo begins to rotate to compensate for the change in orientation.

7.4.b: GPS Testing

Purpose: To make sure that the MediaTek MT3329 GPS Module can correctly locate its position.

Procedure 1:

1. Connect the GPS module to a PC using a serial adapter.
2. Connect a voltage source to the GPS module.
3. Launch the terminal configuration with the preset parameters: 9600 bauds, 1 stop-bit, no parity bit.
4. Module should display a NMEA output sentence on the screen.

Expected Outcome: The GPS module displays a output NMEA sentence displaying the date, time, latitude, longitude, etc.

Conditions of Success:

The output NMEA sentence data displayed on the screen matches the correct date, time, and location.

7.5: Charging Station

Purpose: To ensure the charge controller properly handles varying voltage levels to avoid battery and load damage.

Procedure:

1. Fully charge charging station battery.
2. Remove solar panels connections to charge controller.
3. Vary input voltage to the charge controller
4. Lower charging station battery voltage to <11 volts
5. Raise battery voltage to >12 volts

Expected Outcome: The charge controller respects the limits of operation to ensure the equipment avoids damage.

Conditions of Success:

- Charge controller stops charging the battery when it exceeds 14.1 volts
- Charge controller disconnects the loads when the battery voltage drops below 11.5 volts
- Charge controller waits until the battery voltage exceeds 12.6 volts before reconnecting the loads.

7.6: Software Testing Environment

The software testing environment will vary depending on the test being conducted and at what stage of testing. Modular tests will be conducted in an isolated environment such that external systems will not be present to ensure the software being tested is behaving as expected and not malfunctioning as a result of external sources. In most cases the client application will be tested in an Eclipse IDE environment on a windows 7 machine connected to the internet. Similarly most of the control logic will be tested and debugged in code composer studio and run on a piccolo development board to ensure proper execution.

When the route plotting software test is successfully completed and debugged on the PC, the next step is to connect the XBee transceivers and attempt to transmit a fully plotted route as a single array of global coordinates. For this the PC will have to be connected to the XBee explorer and mounted with the XBee module.

7.6.a: Application – Route plotting

Purpose: To check that the route plotting program correctly translates relative positions on the display to GPS coordinates and formats appropriately for transmission.

Procedure:

1. Preset current location coordinate to initialize map display
2. Select New route to begin drawing on an initialized map
3. Begin drawing route with new points
4. Watch console output for relative and translated points
5. Check map for accurate API translated coordinates
6. Connect points and close route loop
7. “Initiate flight” and view data output for proper format

Expected Outcome: Route coordinates will be managed translated managed and formatted for transmission.

Conditions of Success:

- Translation of clicked point to actual GPS coordinate does not exceed 20% error.
- Points formatted in the order they will be visited.

7.6.b: Camera – Application Interface

Purpose: To make sure that the application properly receives video feed from the wireless camera system and is displayed with adequate quality without any significant delay.

Procedure:

1. Connect receiver to the PC.
2. Power on both camera and the PC with application.
3. Run the software so that the camera transmission data can be displayed.
4. Move camera around and examine video feed to see if there is any significant delay.

Expected Outcome: The video stream coming from the camera is clear and has no significant delay

Conditions of Success:

- Application recognizes receiver connection and grabs video input stream.
- Video stream is shown on the application as the software is running
- Video quality does not show excessive delay or rolling shutter effect.

7.7: System Test Procedure

7.7.a: Route Flight

Purpose: To check if the coordinate data of designated route is successfully calculated to its GPS equivalent and sent through the wireless protocol without corruption. Also, to check that the flight command is processed by the microcontroller successfully. The team is going to integrate two major subsystems to test the entire system. This test module will be the complete system for the project.

Procedure/ Expected Outcome:

1. Open Docking station and expose solar panels.
2. Power on all subsystems, including camera
3. Run the camera and open the user interface application on the PC
4. Create a new route map.
5. Examine if GPS data has properly updated the map to the current docking station location.

6. Examine if clicked points on the map are correctly translated to coordinates and updated through the Google maps API.
7. Pan, zoom, remove and add points
8. Draw a complete route.
9. Examine if route meets flight criteria
10. Initiate flight from client application and examine UAV marker.
11. The route coordinates and flight command should automatically be sent when the application initiates flight.
12. UAV should respond in less than a few seconds and begin taking off
13. UAV should ascend to flight height while maintaining stability
14. Examine video display from client application to make sure video feed is properly being transmitted.
15. The UAV should then begin visiting the designated coordinates in the order they were routed.
16. Control system should maintain stability so that the vehicle does not spin against the motors torque even when moving forward.
17. Examine UAV marker on client application as it updates to the current location of the UAV.
18. Once all markers have been visited it should begin returning to the docking station.
19. UAV should over the docking station and slowly descend and land to begin charging.
20. Charging station indicators should show if the docking station battery is suitably charged.

Conditions of Success:

- User interface is intuitive
- Camera and map display is clear and objects are well defined
- Application is able to communicate with UAV
- Microcontroller receives data from the PC application
- Servos receive signal from microcontroller
- Servos move to designated position accurately
- No overshoot or swings occur while in flight
- Control loop performs accurately
- No malfunctions occur in any of the subsystem while operation
- No components are burnt during operation.

8.1: Milestone Discussion

One of the most important aspects to make sure that the completion of this project is done on time is to create a timeframe to make sure that major milestones are completed by appropriate dates. By stating dates of upcoming milestones this is also very useful to keep all members of the group informed and on the same track of important dates. The chart below shows some of the major milestones that and the dates for completion to properly complete this project. The first major milestone that starts the day the project is decided upon, is to research how to build the vehicle. The goal was to have all research done completely done by November 30th. The next major milestone that needs to be started is Design. The design of the drone and what is needed to be implemented into the design was an ongoing thing and could be done up until the actual building of the drone. Much of the design and research are to go hand and hand so these to milestones will take part at the same time. The design of the AFSD needs to be completed by January 1st so that building can commence. A parts list of all the different components that are going to be used need to be selected and ready to buy by October 15th.

In order to be able to fund the project and start purchasing some of the necessary parts a week was designated to getting funding from Progress Energy in the beginning of October. The proposal to Progress Energy is to be submitted on October 19th. The biggest milestone for the design was implementing all the research, design, and parts into a paper that is to be finished by Thursday December 6, 2012. In order to make sure there is enough time to put the documentation together for the paper and bind it the paper should be completely written by December 4th. As research, design, and part selection took place the writing of the paper describing the design comes along as these other aspects are completed. In order to start testing the microprocessor that will be selected and and begin programing a development board must be selected rather early. It was set to be researched for a few weeks and selected by November 10th. Starting in the beginning of November another milestone that needs to be accomplished is the purchasing and testing of the different modules. This should be done until the actual building of the vehicle on February 12th. Application testing should ideally be started by the beginning of November and continue until February 12th.

In order to have plenty of time to test and make adjustments to the vehicle, the PCB must be ordered in the beginning of February, no later than the 6th, so that the final product can be put together and tested. To give an ample amount of time to allow for modifications to the vehicle the ideal time to begin flight testing the vehicle would be February 15th. The flight testing will continue until the day of the final presentation but the AFSD should have a successful flight by April 15th to ensure everything works correctly. The final milestone that will lead to the completion of the project is to finish up and turn in a final paper after the presentation. So far all of the major milestones have been started or accomplished by the specified dates. The continuation of meeting these major milestones will prove to be very helpful in having a working final product by the presentation date.

Milestone Completion Dates:

- September 19, 2012: Proposal for Progress Energy Funding
- October 15, 2012: Have generic parts list of all the different parts necessary
- November 10, 2012: Purchase Development Board
- November 30, 2012: Finish research
- December 4, 2012: Complete paper
- December 6, 2012: Submit final copy of the paper
- January 1, 2013: Design
- February 12, 2013: Finish module and application testing so finish vehicle can be assembled and tested.
- February 6, 2013: PCB ordered
- April 15, 2013: Success vehicle flight and landing

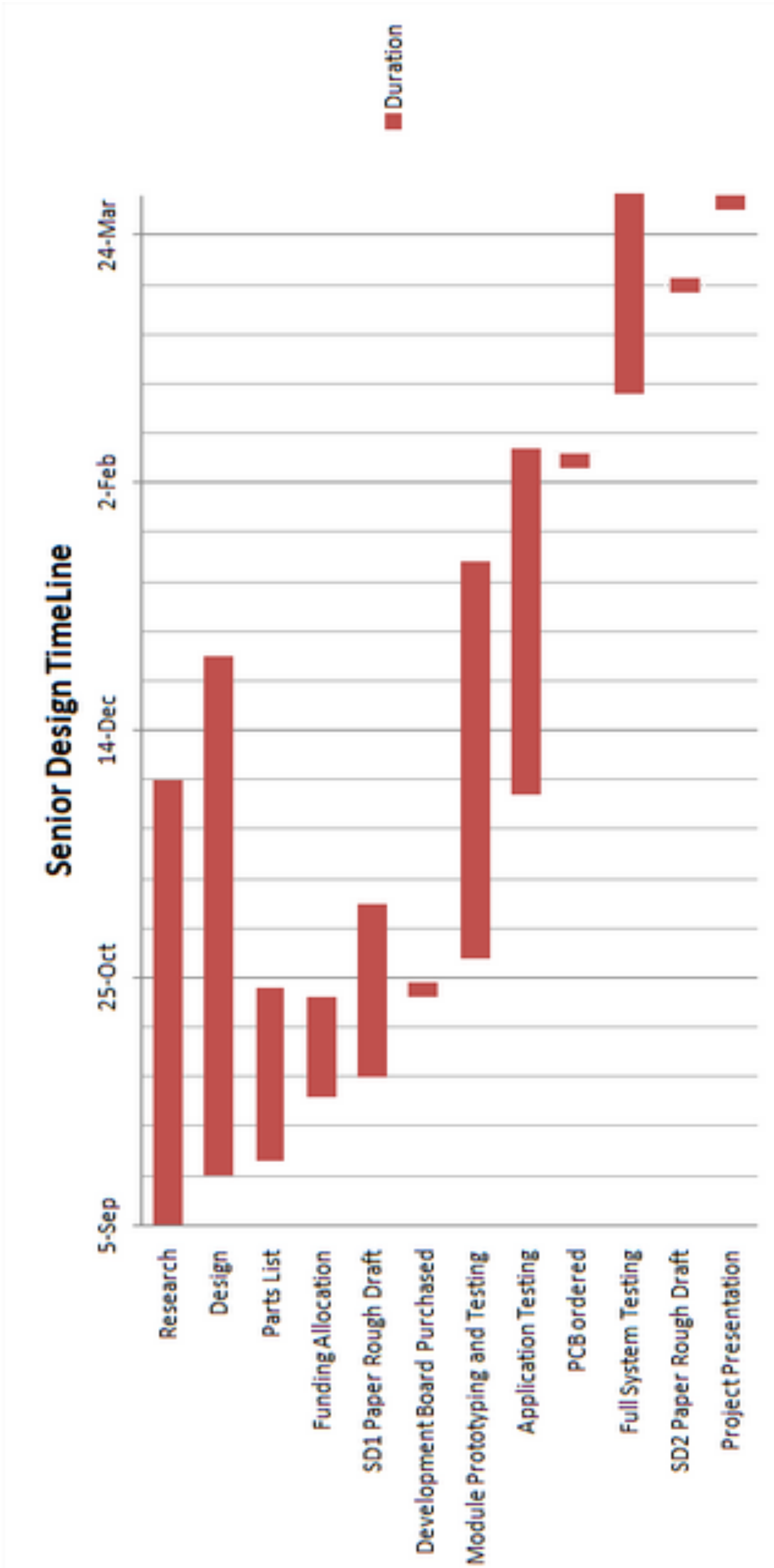


Figure 8.1.1: Senior Design Timeline

8.2 - Budget & Finance

The project has been designed with the intent on maintaining value within all of the components. Throughout the parts list decision making process, there were several instances where there could have better or more efficient part chosen. However, the most efficient parts come with a premium price that would dramatically increase the overall price of the project. When presented with the option, the best value for achieving project goal minimums was always taken.

Project Budget			
*The following figures are our best estimate of which we believe our parts are going to total our UAV Design.			
Category	**Componet**	**Quantity**	**Total Price**
Electronics	Camera	1	$ 55.00
	GPS	1	$ 100.00
	MCU	1	$ 25.00
	Servos	8	$ 152.00
	SD Card	1	$ 20.00
	IMU	1	$ 100.00
	Transcievers	2	$ 60.00
Testing Equipment	Wires	1	$ 20.00
	Harness	1	$ 15.00
	Remote Control	1	$ 150.00
	Dev Board	1	$ 20.00
	Safety Charging Bag	1	$ 22.95
	Bread Board	1	$ 20.00
	Testing Board	1	$ 50.00
Structure	Frame	1	$ 100.00
	LiPo Battery	1	$ 81.19
	Terminals	1	$ 15.00
	Control Surfaces	1	$ 40.00
	Propellor	1	$ 16.99
	Housing	1	$ 15.00
Charging Station	Solar Panels	2	$ 200.00
	Balancer	1	$ 60.00
	Holding Battery	1	$ 69.96
	Aluminum Frame	1	$ 200.00
	Fan Motor	1	$ 20.00
	Control Motors	1	$ 30.00
	Current Monitor	1	$ 71.99
	Fan	2	$ 15.00
Total Projected Cost			$ 1,745.08

Figure 8.2.1: Project Budget

The reasoning behind balancing price versus performance was due to the fact that a competitive proposal process through Progress Energy was attempted. The group wanted to present an effective and realistic project without losing the chance of acquiring funding due to an astronomical budget.

Due to the group's financial status, everything was pending due to the possibility of receiving funding. All four group members were in agreement with each other that the platform of the renewable energy charging station was to be implemented if and only if funding was provided. If funding was not awarded, the conclusion would be to charge the UAV through traditional means of a socket outlet as opposed to the solar power system design. The budget as a whole would drop by nearly $400 and change the personal contribution expectation from $446 down to a more reasonable and manageable $346 per person.

After careful consideration of parts a proposal was made to Progress Energy as a request for full or partial funding for the UAV project. The presentation requirements included a complete parts list, detailed project budget, project development milestones, project block diagrams, project specifications, project motivation, and a statement on the project's impact on energy sustainability and/or renewable energy. When all was completed, the group was awarded the requested funding amount of $1,784.00. This in turn led to a kick start for in depth research into the solar power design. Now knowing the solar system is an option, the renewable energy aspect is finally a reachable goal.

8.3: Conclusion

Through rigorous and diligent research, solutions were found to many of the design questions encountered when the project was conceived. However, many more problems arose when the idealistic assumptions met with harsh reality and a limited budget: Many of the "solutions" were based on components too precise to exist or too expensive to ever afford. A surprising number of compromises and workarounds had to be shoehorned into the final design just to get a working product at the end. This project is, in essence, a real-life example of some very fundamental tenets of human existence: Everything is easier said than done and if something sounds too good to be true, it's probably prohibitively expensive.