Fall 2017

Reduced Gravity Flight: A UAV Approach

Sponsored by:

Northrup Grumman Corp.

Andrew Brahim,EE

Jourdain Francis,CpE

UCF ECE Senior

Design Group 32

Fall 2017

Table of Contents

1.0 Executive Summary

The need for low cost Microgravity (reduced gravity) experimentation platforms exist for researchers who may not have access to drop tower facilities or funds to carry out existing methods of collecting data in reduced gravity environments. Parabolic flight requires modified commercial airline equipment researchers do not have, and renting such a flight or service is not a feasible cost for the average university faculty scholar or researcher. Parabolic flights cost approximately $50,000 U.S depending on the payload size. Various drop towers exist throughout the world that allow reduced gravity experiments over the course of 7-9 seconds, but these facilities are not always feasible for researchers due to travel costs and expenses, coupled with difficulty in securing a reservation for tower use. Lastly, sending up a payload to the ISS for reduced gravity conditions incurs great costs and logistics, not always working within the timeframe of the researcher.

Ergo, the proposal is made for a drone based system to carry a payload (comprised of a vacuum chamber and sensors) for experimentation at predetermined altitudes for free fall drops or parabolic flight. This project seeks to create reduced gravity conditions via drone flight for a length of time necessary for payload experiments to be recorded. The payload environment must also be recorded and instrumentation such as drag, acceleration, temperature, and velocity are necessary (see section 3.1 for specific design requirements, House of Quality). Specifically, Electrical and Computer Engineers will be designing the power distribution circuit, implement an embedded system required for sensor data logging, and develop the control system. In section 3.1, the individual team member responsibilities outline flight dynamics and controls, telemetry, receiver/transmitter, power distribution, and onboard instrumentation components. Interdisciplinary team effort may be necessary to develop the power distribution system of the drone, depending on DC motor selection. Mechanical and Aerospace Engineering (MAE) team will be in charge of aircraft design to meet the stringent altitude and drag requirements of the project. Computer science students will be tasked with creation of graphical user interface in addition to applying their background and knowledge to algorithms for control systems. The ECE and CS teams will combine efforts to develop a robust control system that allows for clean reduced gravity results. The end objective of this project is to give the researcher an inexpensive, easy to use, near on demand means of conducting reduced gravity experiments.

The interdisciplinary team's goal in this project is to carry a payload minimum 10 pounds (roughly the size of a shoebox) via drone and perform flight maneuver to simulate reduced gravity within 10^-1 to 10^-4 g's for 2.5 seconds or more. The data may be recorded and sent via live telemetry or stored on an onboard device for later interpretation. It has been through research and analysis of fixed wing vs multirotor vs lighter than air options that we have decided on a quadrotor platform, due to the complexity of lighter than air options and fixed wing recovery systems residing outside of the scope of this project.

Due to the altitude, payload, and air speed variables in this project, it is highly possible that our design may be at the upper limit of, or exceed Federal Aviation Administration (FAA) regulations for unmanned aircraft. An aircraft weighing less than 55 lbs falls under part 107 of the Federal Code of Regulation, which imposes regulations on model/hobby or small industrial/commercial grade unmanned aircraft. Part 107.51 of FAA regulation limits air speed of such craft to 100 mph, and imposes a maximum flight ceiling of 400 feet above ground level. In later technical sections of this report, we may discuss further the implications of these height, payload, and air speed constraints as they may influence our microgravity results and flight dynamics. However, there is the option of requesting permission through waiver process to exceed these regulation limits. This process may exceed 90 days and waiver is not guaranteed. Another potential workaround is to utilize military training routes on military installations, via requesting permission from military air traffic control to fly our craft. Further analysis of these options will be discussed in later technical sections.

Previous research projects have incorporated various design platforms to replicated reduced gravity environments using drone technology. One such platform, a multi-rotor drone, sought to “provide a minimum of 4 seconds of microgravity at an accuracy of 10^-3 g’s..” (Afman, et al.). Afman, a faculty research member at Georgia Institute of Technology, utilized a variable pitch quadrotor, modeled a simulation (accounting for forces, masses, drag, etc.). This simulation incorporated a PID cascaded system controller and used an automated flight control strategy in order to obtain clean microgravity results. Further complexities arouse in the project safety standards, which required advanced control algorithms and geofencing (to prevent the drone from flying above FAA limited altitudes). In the end, the researchers chose a physical drone platform, which consisted of a Pixhawk flight control board, a dual IMU comprised of a Invensense MPU 6000 and ST Micro LSM303D accelerometer (Afman, et al.). Unfortunately, the test platform was never flown due to unbalanced systems vibrations, however the drone was able to hover, it was not operational to the level required to produce microgravity conditions.

2.0 Impact of realistic design constraints

2.1 Economic:

A project of this scope requires precise hardware specifications and or custom fabricated parts. Sponsorship by Northrup Grumman has given our team primary funding for the project. For the entire interdisciplinary team, a budget limit of $2000 is suggested. This budget may be subdivided amongst the individual teams. Component selection and testing is crucial to maintaining this budget.

UAV hobby grade DC motors widely available may not produce an adequate thrust due to the ten-pound payload requirements (in addition to the aircraft weight). To power the motor(s), a proper evaluation is necessary to account for continuous current drawn by the motor loads. As such, a Lithium Polymer battery is ideal due to it’s high discharge rate, however a large enough Li-Po battery is relatively expensive compared with the average hobby grade batteries. Based on motor selection, specifically the maximum continuous current drawn by the motor, an electronic speed control unit (ESC) must be purchased and installed for each DC motor on board. Again, these stringent hardware requirements reflect the nature of the project: a heavy lift UAV capable of maneuvering in a parabolic flight safely without completely discharging the batteries and it must safely engage in a parabolic or dive flight with enough thrust to overcome drag forces acting on the craft.

An alternative power source, albeit less economical, is a fuel powered drone via combustion engine or jet turbine. Although any engine induces vibration, there is the option of a specific engine layout or aftermarket motor mounts to reduce vibration. Specifically, Zenoah makes a single cylinder engine capable of powering fixed wing model aircraft, costing upwards of $300 for the G26 model. Comparable hobby grade engines (potentially suitable for this experiment) may cost anywhere from $100-$1000 dollars U.S. It should be noted that combustion engines and quadrotor drones lie outside the scope of this project due to the complexity in multiple combustion engines in a multi-rotor configuration. These drones are usually carbureted, and offer displacements of 10-1000+ cubic centimeters (cc).

Additionally, industrial grade manufacturers exist who sell UAV engines to public service departments or large-scale agriculture/industrial operations. These propulsion systems are generally multi-fuel, fuel injected, have large displacements, and are costlier. The last propulsion system option on the table is potentially the most potent, but also the most expensive: mini jet turbine engines. Manufacturers such as JetCat produce a relatively lightweight, high thrust engine capable of 13+ pounds of thrust. The lower thrust models cost approximately

$2000 dollars U.S. These mini jet turbines have high thrust to weight ratios, however their cost exceeds the budgetary constraints of this project.

Electric motors in a quadrotor configuration may be the most economically feasible approach: four industrial grade brushless DC motors may cost roughly the same as one gas engine combustion engine (capable of enough thrust, given an appropriately selected prop). In other words, it is feasible to design and build either a brushless DC motor quadrotor platform or a fixed wing gas combustion engine drone. A third option is to build an electrically driven fixed wing craft. Again, we have brushless DC motors with this build, roughly costing $300 dollars U.S.

For the actual aircraft build itself, it is necessary to either build a custom craft to accommodate the microgravity payload (roughly a shoebox sized enclosure) or retrofit an existing quadrotor/fixed wing craft to affix the payload. It is more financially feasible to retro fit an existing craft or aircraft kit, however UCF's engineering facilitates custom machining, leaving only the cost of materials in the case of a fabricated build.

Outside of aircraft build and motor and battery selection, there are very little costs incurred. Open source flight automation platforms exist, giving the user a cheap platform to build on whilst maintaining customization and variability. Autopilot Flight controllers such as ArduPilot and Pixhawk are easy to use and cheap, while manual control flight boards such as SP Racing or Naze32 have more open-ended hardware and GUI applications for acrobatics despite sacrificing autopilot precision. Software for flight programming is generally free and open source. A free and open source simulation platform, FlightGear, allows real time control system tuning and 3D simulation of aircraft, however these PID tuning features are only available when FlightGear is used in conjunction with Matlab. Matlab is not free to students and costs $120 with the required aerospace block set and tool box necessary to create a control system and tune it.

Other additional economic considerations involve electronic speed control units, power distribution blocks, and miscellaneous hardware. It is important to simulate the microgravity drone first in FlightGear and then open Matlab Simulink to fine tune the control system: the importance of this step derives from the nature of parabolic flight and microgravity trajectories which will be covered in the design overview and testing sections. Simulations provide insight as to un-accounted for disturbances in the control system and indicate instabilities in the flight patterns. In manipulating the drone to project itself on a parabolic path, there may be unforeseen consequences which result in catastrophe to the drone. As such, our build may exceed $2000 and replacement of parts in the event of a crash would be detrimental to our team's objectives and milestones. In summary of economic design constraints, propulsion system, battery, aircraft platform, and several other design components must be carefully specified to get it right the first time.

2.2 Environmental and Operational:

In testing and flying our microgravity drone, we must prepare for and be alert to objects in our surrounding to avoid collisions. For example, low hanging powerlines, cars, and humans. It is necessary to choose a large open field to test the drone, and per FAA guidelines Public Law 112-95, Section 336 and Title 14 of the Code of Federal Regulation (14 CFR) Part 107 restricts flying unmanned aircraft over humans or from a move vehicle. Additionally, the pilot must be aware of and yield to manned aircraft. Propulsion systems that are fueled must be capped off properly to avoid spilling fuel in the surroundings, while in flight, or when re-fueling. Lithium polymer battery cells that fail must be properly disposed of, in addition to other battery types including: Ni-Cad, Nickel Metal Hydride, Small sealed lead acid, Nickel Zinc, and Lithium-Ion. Under Florida law, it is illegal to dispose of Small sealed lead acid and Ni-Cad batteries. For the purposes of this project, we will focus solely on Lithium Polymer batteries. Since these batteries are becoming popular in hobby and RC, and have similar properties with Lithium ion (commonly used in cell phones), there is a growing concern of improper disposal. Daniel Hsing Po Kang et. al state that due to the lack of disposal regulations and the widespread use of these batteries, they "contribute substantially to environmental pollution and adverse human health impacts due to potentially toxic materials" (Po Kang, et. al). He further expresses his concern, stating "defunct Li-ion batteries are classified hazardous due to their lead (Pb) content (average 6.29 mg/L; $\sigma = 11.1$; limit 5)" according to U.S. Federal regulations.

Lithium batteries may present a substantial level of human toxicity and ecotoxicity. Given this hazardous waste concern, it is imperative our team properly dispose of swollen, and/or defunct batteries. Several large chain electronics store retailers exist nearby the university and may be our selected site for battery waste disposal if needed. In the event one or more of our lithium polymer batteries become defunct, we will need to safely discharge the cells to 1/10th the capacity using resistors or lightbulbs. Visibly damaged or swollen lithium polymer batteries should be placed in a bucket of sand or in a fireproof container to prevent fire or explosion from harming living organisms or damaging the environment.

In the event our battery is swollen or visibly damaged, we would not discharge the battery but immediately isolate the battery to prevent flammability. Additionally, our team will have on hand an ABC fire extinguisher, which is the proper fire extinguishing agent for lithium batteries, and a bucket of sand on site for battery disposal while testing our aircraft. Once disposed of in a fire proof enclosure, we may bring it to a disposal center for proper handling of hazardous waste. Our greatest measure in preventing a defective or swollen battery is to keep the lithium polymer cells charged properly, using a lithium polymer balance charger. The cells must be charged and discharged using only this type of charger due to the nature of lithium polymer cells. Proper care must be taken when discharging the battery for storage so that the cells do not fall below 1 volt each. Storage of the battery must be in a cool location, away from heat sources and direct sunlight.

Environmental considerations must also be given to usage of the machine shop, microgravity lab, the innovation lab, and other areas where fabrication or prototype design is under way. Waste materials must be properly disposed of according to state, federal, and university requirements. Hazardous materials must not be left in any facility, household, or test site. If testing other propulsion systems, such as a combustion engine, the engine must be tested in a well-ventilated area, outdoors, to prevent buildup of carbon monoxide. Carbon monoxide gases are invisible and undetectable by human sent, and can be fatal if inhaled in an indoor area.

Lastly, an FAA guidelines document, AC 107-2, gives pertinent information on weather monitoring and specifically lists recommendations under it's appendix b.3. The document gives a link [7] a Flight Service application for unmanned pilots to gather local weather information before flights. This application will be crucial in the flight testing phase of our project. The document also gives a link to the National Weather Services' (NWS) aviation weather site, and touts it as a free and important tool before flights. AC 107-2 also provides a link to it's Temporary Flight Restrictions site, (TFR) [8]: which will also be important during our pre-flight inspection and checks. Sections B.4.1 & B.4.2 give brief descriptions of the effects of wind speed, direction, and surface heat on UAS performance, the gist of which involve wind turbulence and surface heat affecting UAS controllability. Our senior design team will have to take wind speed and direction into account, as well as the terrain we will be flying over.

2.3 Political and Social:

In recent years, lawmakers have attempted to modernize rules and regulations of unmanned aerial systems (UAS). As of 2012, the FAA Modernization and Reform Act set out to implement a timeline of regulating commercial drone use and intending to impose regulations by 2015. Beforehand, the FAA had only regulated model aircraft used only for recreational purposes. The FAA interprets hobby and recreational aircraft as only used for personal purposes, excluding commercial ventures and for-profit operations. As of August 2016, the FAA guidelines changed to allow small UAS to be flown for commercial uses without exemption. This is known as the "small rule", in Part 107, restricts flight ceiling to under 400 feet, daytime line of sight flight only, and imposes a weight restriction of 55 lbs.

Additionally, the operator must obtain an FAA remote pilot's license. He or she is required to pass a basic aeronautical test, be at least 16 years of age, and vetted by the Transportation Security Administration (TSA). Other rules for remote pilots include yielding to other aircraft, airspeed under 100 mph, not flying over people or from moving vehicles, and flying in class G-airspace (permission is required from local air traffic control before flying in B, C, & D airspace).

However, grey areas still do exist in the regulations concerning the differentiation of model aircraft from commercial unmanned aerial systems. These grey areas are contested in the scope of this project, as our senior design team is not operating a “small UAS” for compensation purposes, or for business, but for educational research. Further research and simulation models are required to be certain that the microgravity drone can follow a trajectory under 400 feet and still obtain clean reduced gravity results. If a 400-ft. flight ceiling is too low for our parabolic or dive trajectories, we will need to obtain a waiver from the FAA in order to exceed that flight ceiling.

Our team and faculty have the option of filing for certification of authorization or waiver from the FAA: a process which may exceed 90 days. Fortunately, the FAA views educational use of UAS as for hobby or recreational use, due to the inherent learning potential of students to study flight dynamics. Although students my graduate and obtain jobs or careers involving unmanned flight systems, the FAA chooses to still view educational use of UAS as strictly “hobby or recreational”. The FAA sees the value of safety and learning through academia. Under the special rule for model aircraft, operators have the same altitude, weight, and speed restrictions under commercial UAS small rules. It is fortunate that our team has been given the benefit of the doubt and may fly our microgravity drone without registering for a remote pilot’s license.

Additional flight restrictions may apply through city and state airspace regulations. We will plot out our testing area beforehand, and ensure it is an open area, in class G airspace, away from traffic and people. An alternative airspace testing site may be arranged through our sponsor, Northrup Grumman corp., who may contact the proper channels to allow us to fly in military training routes. This alternative is highly attractive if the waiver process is not successful, or no successful test site is found in civilian airspace in the central Florida area and it’s surrounding cities. As for the weight of the aircraft, there is a noted exception in FAA regulations for aircraft over 55lbs: The craft must undergo inspections and testing from an operational safety program given by a community based organization.

Our microgravity platform will be constructed and programmed, using open source hardware and firmware, as well as documented calibration, such that the prototype and final builds will have a basic, beginner level arming procedure. Second to the arming procedure, the mission flight path (if autopilot is chosen) shall be plotted using a beginner level mission flying firmware/software (e.g. point and click waypoints, angle of attack, etc)

A UAS arming, speed control calibration, flight control board setup and configuration, motor plus propeller setup and direction, mission flying, and wiring procedure will be documented and appended in a later section of this report under Project Operation. One member from each of the disciplinary teams (electrical and computer engineering, mechanical and aerospace engineering, and computer science) will go line by line of each procedure and complete a successful flight with useful reduced gravity results. This test from each discipline, plus the documented procedures, will aid researchers and scientists who wish to conduct their own reduced gravity experiments using UAS technology while avoiding complexities and uncertainties.

2.4 Ethical:

Due to the nature of our sponsored senior design project., we must put our best ethical practices forward to represent both our university, engineering departments, and Northrup Grumman Corp. It is imperative that our team set the bar high in abiding by ethical practices to bring credit upon ourselves, and future senior design projects at University of Central Florida. Our primary ethical design constraints lie in managing our sponsored funds wisely: managing budgets to individual disciplines within our team, sections within our project, and overall funding. We must limit funds used and choose our hardware only when we are thoroughly researched specifications.

It is our responsibility to use the sponsor's funds for only what they intended it for: a microgravity research platform unmanned aerial system. Secondly, due to the flight testing constraints and availability of our professors at UCF whom must grade us by our house of quality product capabilities, we must present our data in an honest manner, and backed up by either live recorded data or video footage of data post-flight. We mustn't fudge the numbers, we mustn't lie about our data. If we do fall short of our product capabilities, we must be truthful and honest. Ideally, our senior design faculty will be on site on the final day of testing and able to witness data either live through telemetry link or presented immediately after flight trials. However, this is highly unlikely due to either a remote testing facility or unavailability of professor/faculty to attend the flight testing. Our team will make attempts to get all faculty to witness flight trials live: if this is not feasible, we may use live video recording of the flight and resulting microgravity data.

2.5 Health & Safety:

Several important safety aspects arise due to the nature of this senior design endeavor. These safety aspects can be broken down into three distinct categories: hardware & materials, in air collisions, and manufacturing. As mentioned in environmental design constraints, there are indications that lead may be present in quantities toxic to humans in damaged or swollen leaking lithium polymer batteries. This lead risk overlaps both environmental and health and safety design constraints. Proposals for battery disposal have been previously mentioned in the environmental section also.

FAA part 107 does not require routine maintenance or inspection of craft. To ensure safety to all members of our senior design team, and faculty, bystanders, and advisors, we will comply with manufacturer recommended maintenance and inspections. AC 107-2 section 7.3 advises unmanned aerial systems operators to devise their own pre-flight inspection in lieu of a manufacturer's pre-flight manual. For this project, we will have various systems in place: Autopilot, flight control, lithium polymer battery, power module, electronic speed controllers, brushless DC motors, wireless telemetry, and a backup receiver transmitter.

These systems must all be functional for safe test flights. It is our responsibility to devise our own pre-flight inspection to avoid damage to the aircraft, other people, the environment, or other manned or unmanned aircraft. Physical inspection is necessary for all components: voltage testing of the battery, inspection of all electronic components, and bench testing the aircraft to ensure these pre-flight inspections are met. A formalized pre-flight inspection list may be found in term 2 documentation. It is known that helicopters fly above a 500-ft. threshold, which would require careful altitude monitoring and autopilot parameters as our flight threshold and FAA regulations allow only a 100-foot buffer between small UAS and helicopter flight ceilings. The FAA did release an advisory circular, AC-107-2, in which appendix c gives guidance on small UAS maintenance and inspection.

Based on this circular's guidelines, our team has provided similar risk assessment outlined in term 2 documentation and reports, solely for flight testing procedures and during flight tests to identify hazards, assess those hazards to determine risk level, develop controls, and implement those controls. The last step in the risk management process is to supervise and evaluate the process to make in changes or updates as needed.

2.6 Standards:

As presented earlier in this report, there are dangers in handling defunct or damaged lithium batteries. Standards exist to test for battery manufacturer quality control, packaging, and reaction to extreme discharges or overcharges to the cells. Several standards bearers line the forefront of lithium battery testing and safety measures, although not all will be listed in this report. Of those mentionable include Underwriters Laboratory (UL), who independently test products and establish safety standards. UL 1642 governs testing of lithium battery cells. UL 2054 establishes household and commercial battery testing, but specifically delegates lithium battery cell components to UL 1642 [9], (copyright approval granted, see Appendix C).

The International Electrotechnical Commission sets forth several standards including IEC 60086-4 regarding safety primary cells-lithium, IEC 61960 applicable to secondary lithium cells and portable application, and IEC 62281 which regards primary and secondary lithium cell transportation. Further testing standards include United Nations UN/Dot 38.3, which rigorously tests lithium batteries and cells during transport in a series of 8 tests (T1-T8 tests). These tests focus strictly on induced hazards during transportation. A companion standard, US DOT

(Department of Transportation), Section 173.185, “specifically addresses specifications and exceptions and packaging for lithium batteries; section 172.101 covers shipping.

Together, the UN and DOT guidelines define test requirements for the safe packaging and shipment of lithium metal and lithium ion batteries” [9]. While the scope of our project may not include shipping or packing of lithium batteries, it is important to note that our vendor should comply with some lithium battery standards to ensure the products we buy from them arrive undamaged. Damage to the battery due to improper packing or shipping method could impede our project’s timeline. The United Parcel Service defines two lithium-ion battery classifications for shipping purposes: primary lithium metal (non-rechargeable) and secondary (rechargeable). The secondary classification also includes lithium polymer batteries. UPS sets out specific criteria for lithium-ion batteries that must be air or ground shipped as UPS dangerous goods and be accompanied by warning labels.

As previously mentioned, FAA regulations tightly control operation of unmanned aerial systems in U.S. airspace. FAA regulations such as the small rule, 14 CFR Part 107, section 331_336 definition of unmanned aerial systems, and Public Law 114-190 subtitle B “UAS” safety all give explicit guidance and regulation to licensing, flight restrictions, and definition of UAS and aircraft for hobby or recreational uses. A memorandum from Reginald C. Govan, Chief counsel of AGC-1 to the Director of Unmanned Aerial Systems Integration Office and Director John Duncan of Flight Standards Services details the Educational use of unmanned aerial systems in public university and classroom settings. This memorandum, dated May 4th, 2016, accounts for students using unmanned aerial systems for educational purposes and not for profit services, thus allowing them to operate aircraft for hobby and recreational use IAW (In accordance with) Section 336 of FAA Modernization and Reform Act of 2012. This allows students to operate without a license or certificate of authorization from the FAA as long as they do not seek monetary gain from services utilizing unmanned aerial systems. However, there are still restrictions on small model/hobby aircraft. These specific flight constraints can be found in sections above. This memorandum may be found in in Appendix A.

One other applicable standard that plays a large role in custom fabrication of printed circuit boards is IPC-2152, or the Standard for Determining Current-Carrying Capacity in Printed Circuit Design. This standard guides printed circuit designers and hobbyists alike in determining the feasible current carrying capacity of printed circuits. Terms like thermal convection, conduction, and heat transfer may be foreign to some Electrical Engineers who have not taken a heat transfer or thermo-fluids class, but nonetheless, these concepts are important when designing circuits due to the amount of heat a conductor in a printed circuit transfers to the environment and the rest of the board. IPC-2152 gives graphs and charts containing variables such as copper weight (oz), trace width, and temperature affects to allow the designer to make an educated decision when setting forth parameters for their circuit board.

Other factors that play into PCB design are power dissipation, resistance, and temperature dependence. This standard's shortcomings (especially in light of this project) stem from the lack of data for high current carrying capacity printed boards, such as power distribution boards: it seems that the standard offers some guidance, but assumes you will not be using high current applications in printed boards. Where the standard ends in its guidance, our team must determine the best course of action for our printed board.

2.7 Manufacturability:

Design for manufacturing (DFM) is an applicable engineering design concept to our microgravity drone senior design. The ease of building a product that can easily be manufactured beyond prototypes is at the fore-front of our senior design objectives. In building a microgravity drone platform, we must minimalize size and select components that are readily available through suppliers. In using open source autopilot software, we must carefully choose still-relevant and tech supported firmware and software.

Our entire multi-disciplinary budget is limited to roughly $1500 for the prototype design and an extra $500 to build on the prototype to finalize our product. Careful selection of compatible components and high quality, relatively inexpensive products that meet our design requirements is vital. ECE members have chosen off the shelf commercial components that use open source hardware and software for flight control, electronic speed control, and battery power. Pixhawk and ArduPilot are our two-main autopilot flight controller autopilot selections due to their sensor's capabilities and reliability, coupled with open source hardware and software. A 3DR (3-D Robotics) manufactured Pixhawk is an attractive autopilot option to their quality customer service support and on-going manufacturer updates. However, Pixhawk and Ardupilot compatible boards generally do not have support for bidirectional speed controllers, an issue that is further explored in the explicit design section of this report along with a detailed comparison of two separate building platforms. Additional flight boards for consideration that do not fall under Ardupilot module software will be considered and researched. One such flight control firmware of interest is Clean Flight, which is still relevant and supported by an emerging Github database. The flight control firmware we use shall be continually supported via developers, ensuring researchers planning to replicate our design will not have to resort to legacy software no longer used.

Our Mechanical and Aerospace team members have compiled Solid-works/CAD models of the drone's tear drop shaped shell and have begun the process of selecting either 3-D printing or a machine shop service to produce the completed shell. 3-D models can help researchers in the future reproduce the design in terms of scale, and may use it for simulation models. The final product must be used for repeatable tests to collect microgravity data, so our structure will be re-usable. Also, our power source must be either refueled or recharged: in choosing lithium polymer batteries, we gift the platform a means of powering the quadcopter for repeated flights to comply with researcher's experimental constraints.

2.8 Sustainability, Reliability, and Availability:

Due to the repeatable testing design requirements of this senior design project, we must factor in a sustainable way of achieving microgravity results at a fraction less than previously used methods. The propulsion system, payload shell, sub-frame, flight controller hardware, and controller area networks must be robust, protected from the elements, and able to withstand repeated use. Dampeners may be utilized under the flight controller to prevent shock and vibration from both damaging the unit and reduce noise and vibration from affecting reduced gravity data reading.

There exists a redundancy in accelerometer/IMU sensors in both autopilot/flight control boards selected in the Bill of Materials (BOM) section: allowing for sub-component reliability in a parallel system. As such, modeling the UAS microgravity platform reliability prediction takes into account failure rate, or lambda of t. Considering the large mechanical device components such as rotors, UAS shell, sub-frame, and rotor arms, these devices will experience a wear-out period where failure increases (Ford R., pg.161). The flight controller, power distribution board, and speed controllers, along with power modules will not experience this wear-out period but persist at a constant failure rate. A systems series reliability table is given below in table 1, using the U.S. military MIL-HDBK-217F [4] failure rate chart for analog and digital electrical components:

No.	User Defined	Part Type	Env.	Qty.	FPMH	pi-Q Description	pi-Q	Part Failure Rate (FPMH)
1	Flight Controller	Microcircuits,MOS, Microprocessors Up to 32 Bits	GM	1	0.49	Industrial,pi-Q=3.0	3	1.47
2	Electronic speed cntrl.	Microcircuits,MOS, Microprocessors Up to 32 Bits	GM	4	0.49	Industrial,pi-Q=3.0	3	5.88
3	Power Dis. Board	Microcircuits, MOS, Linear, 1 to 100 Transistors	GM	1	0.039	Industrial,pi-Q=3.0	3	0.117
4	Motors	Rotating Devices, Motors, General	GM	4	8.3	MIL-SPEC	1	33.2
Total Failure Rate (FPMH)								40.667
MTBF (Hours)								24589.96

Table 1: Mean time between failure and Failure rates of basic quadrotor electrical/electromechanical components (excluding peripherals)

Utilizing an online MTBF (mean time between failure) and FPMH (failure per million hours), *MIL-HDBK-217F(N2) Parts Count Prediction Calculator* [4] for the basic electronic components of the quadcopter. In the table above, a total of 40.667 units will fail per million hours, and the MTBF of 24589.96 hours will allow for many microgravity flight tests. A separate mechanical parts calculation may be found in subsequent versions of this report. The end product shall be operational maximum of forty minutes at a time given adequate batteries on hand (for our testing purposes, we will have two batteries on hand, allowing 20-40 minutes of flight time). Hence, the system shall be operational during daylight hours only, from 8:00am to 6:00pm, 365 days out of the year.

2.9 Energy:

Overall energy consumption during flight will vary due to the complex parabolic and dive flight maneuvers. Our interdisciplinary team has determined an adequate battery capacity of 6000-220000 MAH (milli-amp hours) for continuous and burst amperage during flight. The lithium polymer battery discharge rate shall be in the range of 30-65C in order to supply a higher maximum current needed for parabolic maneuvers. For example, a high capacity battery, 2000MAH with a discharge rating of 60C can supply 120 amps maximum current to the application. Also, our battery shall supply energy to the aircraft for 10-20 minutes of flight, and safely land at the end of each flight. The lithium polymer batter must be able to swap out with a fresh battery in a 15-30-minute interval. Additionally, the max continuous current of the battery must be greater than 234 amps and the supply voltage shall be 20-40 volts. These ranges are based on thrust calculations and motor selection covered in the design, test, and build sections of this report. Lastly, lithium polymer batteries supplied by vendors shall comply with shipping, packing, and handling standards outlined in the Environmental and Standards portion of this report.

2.9.2 Legal:

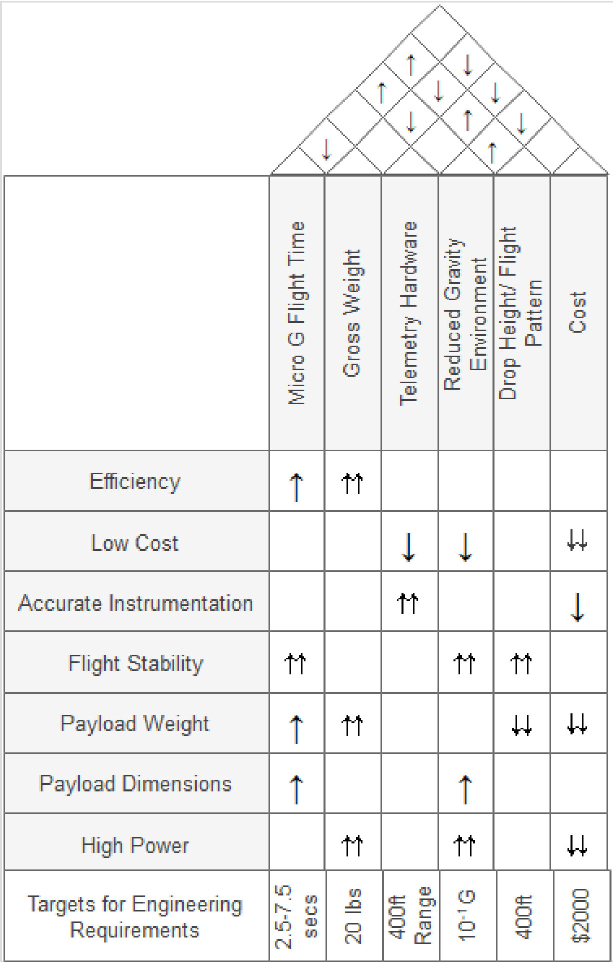
All patents developed during the design process as well as intellectual properties are those of the students, including the Computer Science [CS] team members, Trent Freeman, Noah Headley and Allen Shearin and the Electrical and Computer Engineering team members, Andrew Brahim and Jourdain Francis. Northrup Grumman Corp. is our only officially recognized sponsor, and no other sponsorships (implied) or stated otherwise after this report has been published will be recognized.

3.0 Explicit design w/ BOM, class diagrams, data structures

From a hardware perspective, a drone platform must be built from the ground up to accommodate up to a 20-pound gross weight (including payload) and achieve high levels of thrust to perform dive or parabolic maneuvers while maintaining control of the aircraft. These requirements must be met, in addition to achieving reduced gravity levels of 10^-1 to 10^-4 g's of acceleration. It is the responsibility of electrical and computer engineers to find solutions that work towards building an effective reduced gravity inducing multi-copter unmanned aerial system.

3.1 Technical Objectives

The objectives of this Senior Design project are to create a low-cost drone that can carry a payload high enough so that the drone can achieve a microgravity environment due to freefall. Our group has constructed a list of technical objectives and engineering requirements in order to accurately measure the project's success. In the figure below, we have outlined the generalized requirements that will serve as the basis for testing.



	Micro G Flight Time	Gross Weight	Telemetry Hardware	Reduced Gravity Environment	Drop Height/ Flight Pattern	Cost
Efficiency	↑	↑↑				
Low Cost			↓	↓		↓↓
Accurate Instrumentation			↑↑			↓
Flight Stability	↑↑			↑↑	↑↑	
Payload Weight	↑	↑↑			↓↓	↓↓
Payload Dimensions	↑			↑		
High Power		↑↑		↑↑		↓↓
Targets for Engineering Requirements	2.5-7.5 secs	20 lbs	400ft Range	10^{-1} G	400ft	$2000

Figure 1: House of Quality for Microgravity Quadcopter Drone

The objectives of this Senior Design project are to create a low-cost drone that can carry a payload high enough so that the drone can achieve a microgravity environment due to freefall. Our group has constructed a list of technical objectives and engineering requirements in order to accurately measure the project's

success. In the figure above, we have outlined the generalized requirements that will serve as the basis for testing. These requirements include:

- **Microgravity Flight Time:** The drone will be designed so that microgravity can be achieved for somewhere between three to seven seconds. In this time frame, experiments will be in the condition necessary to record any and all relevant data and be stored to be reviewed for a later time. The success of reaching the desired measurement requirement is highly correlated with the payload weight and dimensions. With more weight and higher dimensions, the drone may have a more difficult time achieving microgravity for such a time frame. The stability of the freefall can also mean the difference between a controlled freefall microgravity environment or a high gravity situation due to uncontrolled circumstances such as rotation and noise from vibrations.
- **Gross Weight:** The weight of the drone will be around twenty pounds, or 9.07 kilograms. This weight is the sum of the physically drone, electrical and mechanical components, and the payload itself. The design and materials used in the creation of the drone are essential in maintaining the desired parameters of the weight. A heavier craft will require higher power output in order to stay within the constraints of the other engineering requirements.
- **Telemetry Hardware:** The hardware selected will have to be able to accurately communicate data between the in-flight drone and the senior design group on the ground. Without this communication, there would be a lack of data being recorded. This makes the telemetry component one of the greatest and most important factors when deciding and setting the height, and henceforth time, that the microgravity drone will be able to stay and record the processes and experimentation of being in a microgravity environment. A good set of telemetry will allow us to receive information from our drone from 1.55 miles (or 2.5 kilometers). Taking into account the distance needed to pull the drone from freefall and reclaim enough control to land, this distance supplies the window of opportunity for calculating and observing the effects of a microgravity environment.
- **Reduced Microgravity Environment:** The reduced microgravity environment is the primary purpose of our senior design project. Our engineering requirement is to create an environment that can maintain 10^{-1} G throughout the duration of freefall. During this time, there drone will hold a payload that will contain small experiments to be carried out. These experiments will be saved, recorded, and transferred while maintaining this state. The gravitational force experienced on the payload will be constantly recorded by an accelerometer. While freefalling, the drone will

mechanically adjust itself to sustain stability and keep the payload at 10^{-1} G.

- **Drop Height:** An appropriate drop height is needed to achieve an optimal duration of freefall and recovery time. The maximum drop height, especially with relation to being able to transfer data while falling, is restricted by the maximum reach of the telemetry. The drop height therefore matches the maximum range of the telemetry hardware at 2.5 kilometers, however we are limited to 400ft via FAA.
- **Flight Pattern:** There are several ways to simulate microgravity on a payload. These different methods, whether it'd be a fixed wing parabolic flight or freefall, require different constraints, configurations, and overall design and implementation by all team branches to simulate the desired results. For the purpose of this project, it has been established that our group will implement freefall by a quadcopter drone. The drone will be programmed to fly up to its maximum height and drop. While freefalling, the drone will be required to not only maintain stability, but also add enough, if any is needed, downward momentum to counteract the drag and maintain the desired measurement of gravitational force on the payload.
- **Cost:** Our cost parameter is based on the budget given to us by Northrup Grumman. This budget is shared among the three branches of our team and will be used for the construction and purchasing of the necessary parts and components for every part of this project. The different components and the complexity of its design drastically change the cost. Finding a balance between quality and cost is essential in the completion of this project.

3.2 Example Control System Hardware Introduction and Materials:

From an electrical and computer engineering perspective, an example control system hardware layout is as pictured in figure 2 (subject to change) Generally, the controller area network (CAN) is integrated into autopilot flight control board, with exception of some sensors. The Pixhawk Mini autopilot board depicted in the example hardware diagram is enclosed in plastic, only leaving I/O pins and ports susceptible to the elements (note that the Pixhawk is only used as an example in this section). It includes an M8N GPS module tethered to the flight controller via bus. Onboard instrumentation and IMU sensors are integrated into the Pixhawk board, which utilizes two IMU's, a primary and secondary, for redundancy and accuracy of data. senior design project

This board's accelerometers will be more than accurate for reduced gravity sensing application. On the following page in table 2.A is listed an explicit bill of materials (BOM) selection example for the electrical and computer engineering component of this project.

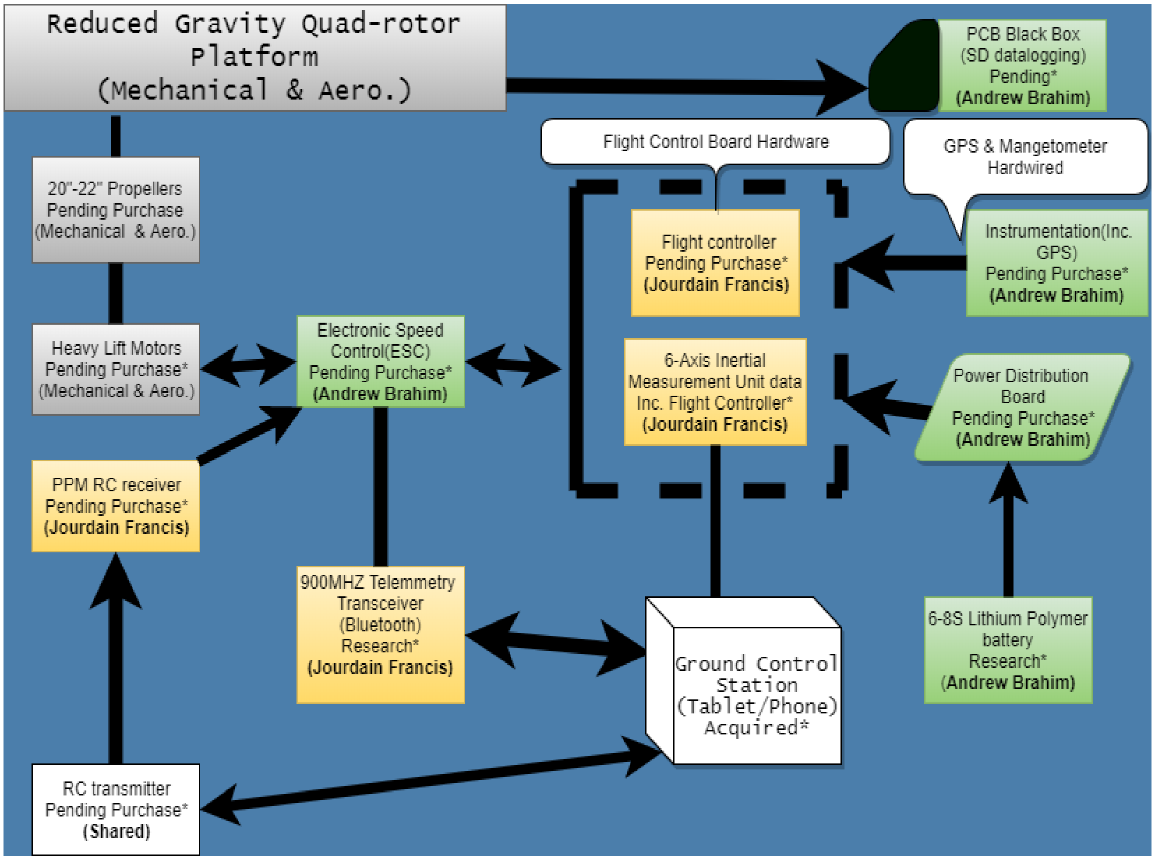


Figure 2: Quadcopter hardware design block diagram shown with team roles and responsibilities

In lieu of, or paralleling the live telemetry link between the transmitter and receiver in the mission planner software, a microSD card will be used (plug in available) in the Pixhawk mini for datalogging of accelerometer and other crucial sensor data needed to evaluate the gravity environment post-flight.

3.3 Control System Hardware and Materials Comparison

One requirement set by Mechanical and Aerospace Engineering team members was that of a bi-directional electronic speed controller, specifically the Lumenier BLheli_32 50A speed controller listed in Table 2.A. Although Pixhawk autopilot hardware and Ardupilot/mission planner software make up a robust platform with many off the shelf options and compatibility, there is no direct support for bidirectional on-the-fly flight control using Pixhawk autopilot. A further decomposition of Pixhawk support for bidirectional esc's are given in the architecture content. A second option is to substitute a car RC speed controller (which allows use of reverse, bidirectional capability) and attempt to modify Pixhawk programming to allow bidirectional speed control support and toggling reverse thrust mid-flight. A diagram of the parabolic flight curve may be found in the Design Overview section.

Component	Description	Cost	URL	Status	Weight(grams)
3DR Pixhawk Mini	Flight controller, 32-bit ARM cortex M4, IMU &GPS incl.	$250	https://www.amazon.com/dp/B071YD56FM	pending	40
10S Power module	Switching voltage reg. Output 5.3v, max 45 volts 90A	$45	https://www.amazon.com/dp/B071Y4CZGZ	pending	22
Power dist. Board	circuit board for cable connections, wire routing	$10	https://www.unmannedtechshop.co.uk	pending	20
Brushless ESC	Electronic speed control, bi-directional/Car RC ESC, Qnt. 4	$100		research	80
3DR telemmetry	ground to aircraft communications, autopilot corrections	$55	https://www.amazon.com/	pending	50
FlySky Trans/Rec.	Radio control trans. and receiver for manual operation	$68	https://www.getfpv.com/flysky-fs-i6	pending	10
Totals		$528.00			222

Table 2.A: Explicit parts list, description, web-link, weight (in grams), and cost (US dollars)

In further research, our team has concluded that another commercial flight controller compatible with Lumenier ESC's could be a substitute platform. An alternate ESC for our application, in the event that our motor and propeller combination will draw more current then the Lumenier 50A can supply, is an Emax BL_heli ESC: This speed controller is approximately the same cost as the Lumenier ESC but has a continuous current rating of 80 A and a burst current rating of 100 A. The Emax ESC is considerable heavier then the lumenier at 81 grams. Another major issue with the Emax 80A speed controller is the lack of information on it's internal (e.g. MOS packages, microcontroller specs etc.) whereas the Lumenier 50A is well documented and specifications even include the microcontroller package and version as well as clock speed, and it's inclusion of a floating point unit. An educated guess on the part of the Electrical and Computer engineering team is that the Emax 80A ESC's use an 8-bit microprocessor to carry out it's calculations: at a lower word length and clock speed, we would sacrifice pure amperage safety netting for system response times, which could prove crucial in the flight envelope needed for parabolic flight.

In changing the platform away from Pixhawk and Ardupilot compatible autopilots, we lose easy access to off the shelf components, in other words, the build becomes more complex. However, in table 2.B we outline such a platform with a completely different hardware supplement, only carrying over the FlySky transmitter/receiver combination for manual flight control. In table 2.B we chose a powerful ARM cortex M4 microprocessor embedded system, built by Seriously Pro Racing (SP racing): the SP Racing F4 EVO flight controller. More details on this controller can be found on a datasheet in Appendix B.

Component	Description	Cost	URL	Status	Weight(grams)
.SPracing Evo F4	Flight controller, 32-bit ARM cortex M4, IMU incl.	$37	https://www.getfpv.com/sp-racing-f4-evo-flight-controller.html	pending	5
Lumenier 6s PDB	Switching voltage reg. Output 5.0v	$13	https://www.getfpv.com/lumenier	pending	9
Lumenier BLHeli / Emax 80 A*	Electronic speed control, bi-directional, 50A max, Qnt. 4	$112	http://www.helipal.com/lumenier-blheli	research	29 / 81
FlySky Trans/Rec.	Radio control trans. and receiver for manual operation	$68	https://www.getfpv.com/flysky-fs-i6	pending	10
GPS module	Ublox M8N or equiv.	$26	https://www.amazon.com/QWinOut	pending	10
Totals		$256.00			63

Table 2.B: Alternative BOM list utilizing bi-directional speed controllers

**Alternate component pending consideration*

The alternative bill of materials build is much more attractive in terms of cost and weight. The flight control board is much cheaper than a true Pixhawk flight controller and even cheaper then several knock-off or copy-cat brands. Electrical and computer engineering research led to this board's compatibility with bidirectional speed controllers through a niche internet segment of drone enthusiasts who brave "3D inverted flight" unmanned aerial systems, expressing acrobatic skills and stunts through inverting the drone midflight and thus, needing the hardware required to quickly switch thrust in reverse to prevent from falling.

The SP Racing F4 Evo flight controller comes integrated with an IMU, an ICM20602 accelerometer/gyro via SPI communication protocol as well as a BMP280 barometer for pressure readings. It features an STM32405 CPU clocking 168 MHZ and includes a floating-point unit. Telemetry support varies, but a close look at the datasheet reveals it is compatible with PPM through a receiver port. The board's underside features a micro-SD card slot with support for 32 gigabytes of data logging. The hardware footprint is small, a 36mm x 36mm square shaped board. This board is recognized as a racing quadrotor flight controller and categorized as such. Its predecessors were the SP Racing F3 boards. One disadvantage to this board is lack of autopilot options. An additional GPS module must be purchased, and the Ublox M8N with integrated compass is supported and compatible with the SP Racing F4 Evo.

The ICM 20602 6 axis motion tracking device manufactured by Invensense: it features 16 bit ADC (analog to digital converter) and accelerometer noise levels of 100ug per square root hertz. The MEMS structure is hermetically sealed and bonded at wafer levels, a feature important for reduced gravity environments where floating particles at the wafer level can inhibit accurate sensor readings. Of importance is the 3-axis accelerometer which supports ranges of +/- 2g and +/-4g which are required in order to pick up low amplitude accelerations in our reduced gravity environment.

Software support for SP racing controllers generally do not include any form of Ardupilot/APM or Mission Planner GUI. Instead, the SP Racing F4 Evo is both supported and built by creators of an open source firmware known as Clean Flight. Clean flight has several extensions, of which two are of interest: Betaflight and iNAV. iNAV utilizes the aforementioned GPS in tandem with an android based application, known as EZ-GUI in addition to Clean flight and it's two extensions to produce a coordinate/waypoint point and click GUI for autonomous flight. Pre-iNAV firmware, boards such as the SP Racing Evo 3 and prior could only use GPS to determine flight coordinates and not for autonomous flight.

This firmware can be used for return-to-home functions, mission flying, and altitude/GPS hold (hover) commands. The issue with iNAV is the lack of autopilot precision a la Mission Planner style coordination in addition to additional hardware purchase, such as magnetometers, GPS, and compass. The reason Clean Flight boards have less options in regards to autopilot and official GPS/Nav firmware hearken back to earlier statements made in this report regarding niche quadcopter racing and acrobatics pilots. In these quadcopter markets, users generally do not need GPS or autopilot as the purpose is to have full manual control of the drone. More detailed information on mission flying in the Clean Flight/iNAV application environment is discussed in the section 4.4.

3.4 Reduced gravity

Reduced gravity is the lack of or minimal presence of a gravitational force. This presence of reduced gravity can be felt while in orbit, or even through a free fall. As long as one can cancel out the force that gravity exerts on them and can stay in equilibrium, there will be a space for reduced gravity. “The properties of reduced gravity conditions make it a valuable environment to conduct research otherwise unfeasible. Fields as diverse as materials science, fluid physics, combustion, biology and biotechnology all have current research questions proposed relating to reduced gravity conditions" [1].

$$H_{initial} = (Max\ Allowable\ Height) - \frac{(Initial\ Velocity)^2}{2(32.2)}$$

Equation 1: Initial Height

$$T_{ff} = \frac{(Initial\ Velocity) + \sqrt{(Initial\ Velocity)^2 - 4(\frac{-32.2}{2})(H_{initial} - H_{critical})}}{32.2}$$

Equation 2 Freefall Time

The equations above detail the process used to find the necessary drop height and subsequent time the drone will have to experience reduced gravity. With so many different applications and benefits that can be applied with the use of reduced gravity, the subject matter has been heavily research to make achieving this environmental state accessible, reliable, and affordable.

- **Problems**
  - In the field of reduced gravity there are many ways to achieve reduced gravity, but each way comes with its own set of flaws and inconveniences. In order to produce and replicate reduced gravity experiments as desired by Northrup Grumman, we must be able to simulate a reduced gravity environment.
  - One way to obtain the amount of gravity, or lack thereof, necessary is by loading up the experiment onto a rocket and shipping it up to the International Space Station. The International Space Station serves as a central hub for man low earth gravity experiments with its expansive gallery of scientific laboratories. Another alternative involves the use of a plane that can fly at certain angles allowing for a parabolic flight. One such vessel, originally owned by Nasa and now currently operated by the Zero Gravity Corporation, is known as the Vomit Comet. "This method has the drawback of high price, which has been recorded to be as high as $3,000 USD/kg. Another major drawback is the lack of repetition, as the aircraft is often booked months in advance and can be very selective" [1]. The Vomit Comet serves its purpose in being far more accessible than sending research material to the International Space Station, but still not a good enough alternative to be considered for this project.
  - Drop towers are also an option typically used in attempting reduced gravity experimentation. The issue with this method is that building

such a large tower takes a high amount of capital. Once constructed, the actual state of reduced gravity is directly limited by the height of the tower.

- **Solutions**
  - For this project, Northrup Grumman has requested the use of a drone to create a reduced gravity environment suitable for experimentation. The Mechanical and Aerospace has decided that the most appropriate approach in order to achieve reduced gravity with a drone is through sessions of freefall in a parabolic fashion. Pictured below is a sketch of the trajectory of the craft. The craft will accelerate upwards to around 400 feet before entering a state of freefall. This freefall state will enable the experiment environment (seen in figure 3 below).

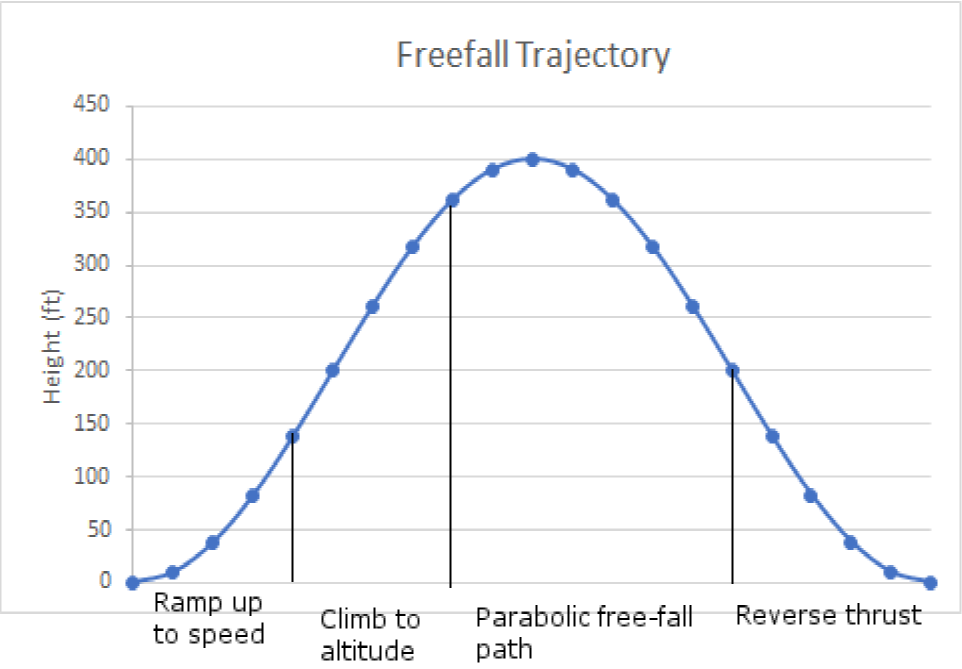


Figure 3 Flight Pattern of the Quadcopter drone to experience reduced gravity

  - The use of a quadcopter drone negates some of the major issues presented in the alternative means of reduced gravity research. Drones as a medium are far more accessible than shipping items up to the International Space Station, renting time on the vomit comet, or constructing a drop tower. The use of a drone also allows for the ease of reuse. Multiple experiments can be run almost one after the other without much hassle. This high accessibility revolutionizes the way that reduced gravity experiments can obtained.

- A few factors to take into account is that due to the relatively low max height, drag and air resistance must be taken into account. Persuasions are taken to ensure the success of experiments. The drone is required to maintain stability during the freefall period. The drone also requires a flight controller that allows for bidirectional motors for the propellers as well as autopilot and basic control capabilities.

3.5 Flight Controller

The flight controller plays an extremely important part in the Electrical and Computer Engineering portion of this senior design project. As a team, there was an ample amount of research implemented into choosing a flight controller that can fit the constraints given to us by the Mechanical and Aerospace department and our own engineering requirements. For the purpose of this project, we have selected to use the Racing F4 EVO Flight Controller. The Racing F4 EVO Flight Controller (board is depicted in figure 4) provides an assortment of useful features that can be coded and modified to work with the components provided by the other branches of our senior design team.

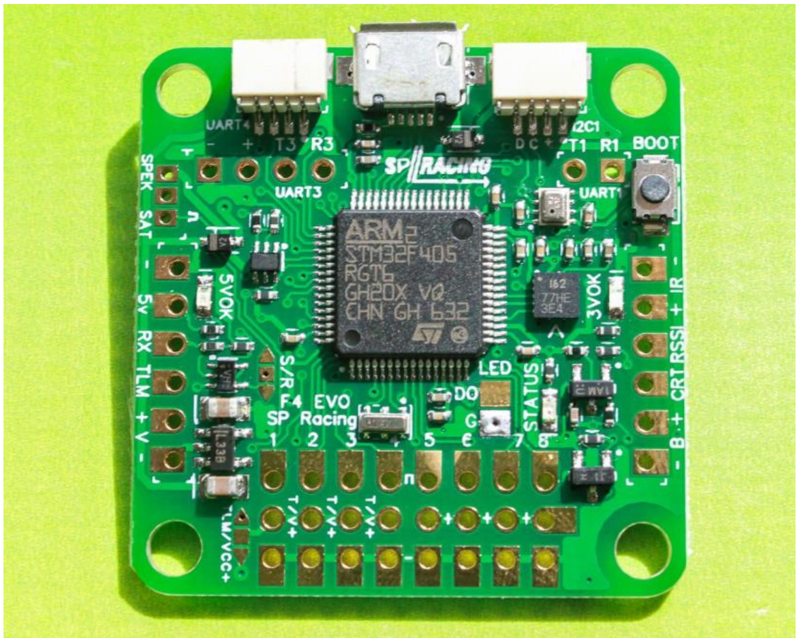


Figure 4: Graphic of the SP Racing F4 Evo Flight Controller

The F4 EVO gives us access to built-in sensors as well as the ability to add our own. With the use of INAV, a branch of Cleanflight the open source flight controller, autopilot and flight control patterns can be implemented and manipulated to incorporate parabolic flight patterns necessary for reduced gravity. The customizable firmware, based in the C Programming language, provides diversability in the electrical components, sensors, and overall capability.

Numerous receiver peripherals are supported by the F4 EVO including spektrum satellite receivers, PPM (pulse position modulation), and SBUS (serial bus, an 18 channel serial communication protocol). The board can be flashed via micro USB connection port (Micro USB socket) or through a serial port. The ease of use of this board is similar to the Pixhawk Mini autopilot module: board flashing, power distribution module, GPS configuration, sensor configuration, and speed controller and receiver hookup are very similar. A chart comparing both boards is depicted below in table 3:

	GPS & Autopilot incl.	**Onboard IMU**	**Baro.**	**Processor**	**Bidirectional Support**	**Cost**
SP Racing	No, Port available, Autopilot through iNAV & EZ GUI firmware	Yes, ICM 20602 IMU	Yes, BMP280	STM32405, ARM	Yes, BLHeli32 suite & Beta Flight firmware	$37
Pixhawk Mini	Yes, incl. Autopilot	Yes, Dual Redundant ICM 20608 & MPU9250	Yes, MS5611	STM32F427, ARM	No direct support in Arducopter APM software	$250

Table 3: Flight control board comparison Pixhawk Mini and SP Racing F4 Evo

A main feature provided by the Racing F4 EVO Flight Controller is the allowance and compatibility for reverse speed control. Reverse speed control compatibility gives the drone accessibility to bidirectional motor control. This feature provides an extra degree of motion and maneuverability. In accordance with other components such as the gyroscope, barometer, and the accelerometer, bidirectional control of the quad motors will be implemented for a variety of purposes. This downward thrust provides stability as well as a canceling force while traversing along the reduced gravity path of trajectory. “Tracking a reduced gravity trajectory requires the use of both positive and negative thrust forces in order to compensate aerodynamic drag in ascent and descent portions of the parabolic flight; something that a fixed pitch quadrotor cannot do” [1].

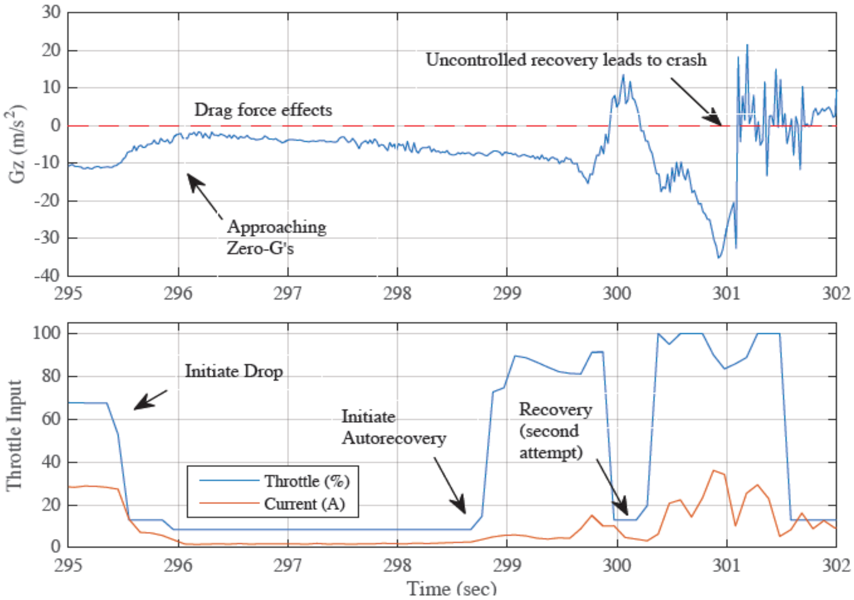


Figure 5: Graph from [1] Simulation of a Fixed Pitch Drone

The research paper [1] presents the graphical representation above in Figure 4 of the problem with using a variable pitch trajectory. The simulation shows that with a variable pitch trajectory, the craft can never fully stabilize at a zero-gravitational force. At a height where drag is still an important factor, the drag causes instability which can prevent the craft from reaching a stable level of reduced gravity. With a varaible pitch, there are problems trying to correct the flight pattern. The drone is unable to correct itself in a way that fully cancels out the forces of drag during freefall. Inability to find a method to have the drone fly steadily will make the project results entirely invalid.

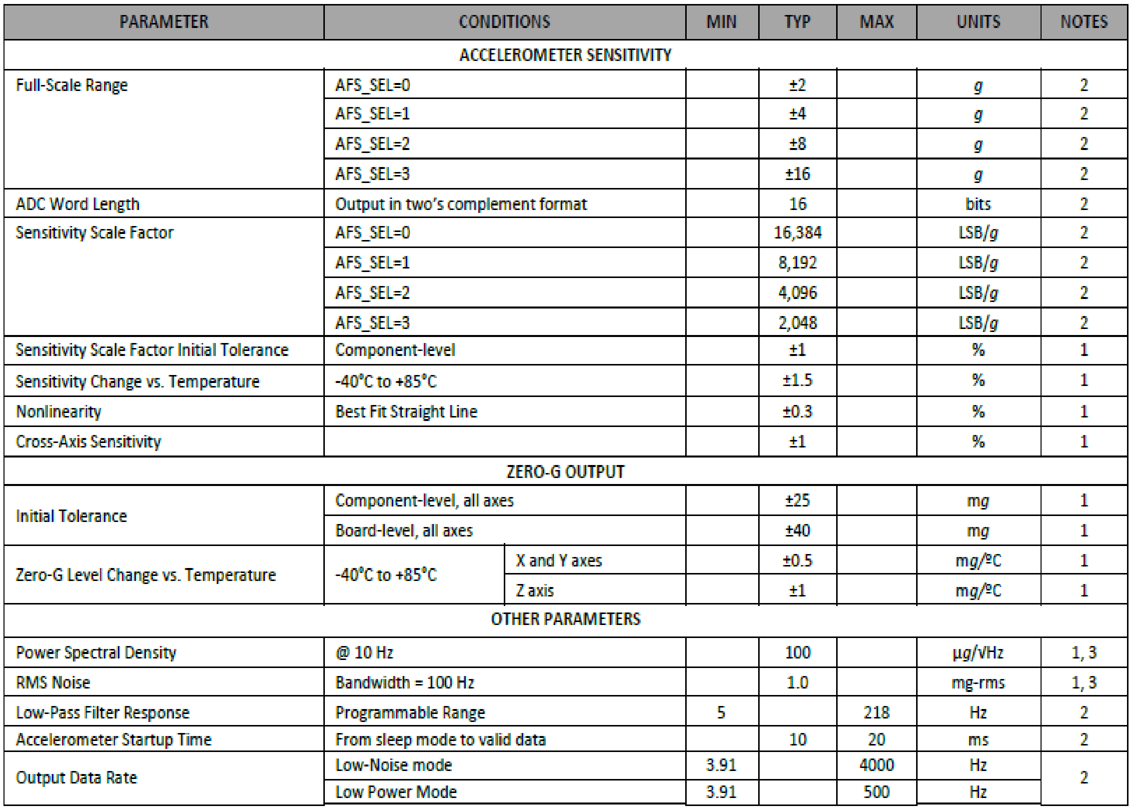
The Racing F4 EVO Flight Controller’s purpose is to directly control the servos of the four rotors of the drone in order to achieve flight and stability throughout the process. The flight control also allows us to plot an optimal route for obtaining reduced gravity on our cargo. The Racing F4 EVO Flight Controller is also equipped to be able to transfer in flight data throughout the duration of the experiment.

This telemetry allows for close and meaningful monitoring of all the flight controls. This feature coupled with the presence of sensors ranging from the accelerometer, global positioning system (GPS), and gyroscope enables those of us on the ground to quickly and accurately make any necessary corrections to better suit the trajectory of the drone as it attempts to reach the desired reduced gravity reading.

3.5.1 Accelerometer

The accelerometer calculates and broadcasts the force of gravity currently acting against the craft. The readings can be used to graph the duration and accuracy of the flight trajectory in regards to the final goal. This component is capable of handling some of the noise caused by the vibrations of the flight throughout the duration of the flight and freefall. These readings as well as the GPS alerts the programming when to begin tracing the desired parabolic arc and when it is necessary to enable the autopilot protocol to slow down enough to allow for a safe and successfully landing of the drone. The drone can also be controlled manually through an RC remote.

- The Racing F4 EVO Flight Controller has a built-in accelerometer to measure the acceleration of the drone from take off to its eventual landing. The accelerometer's data collected from each flight will serve as the best measurement on the success of the quadcopter drone to hit and maintain reduced gravity for a significant duration of time.
- The Racing F4 EVO Flight Controller comes equipped with the ICM-20602. The ICM-20602 provides access to a six-axis motion tracking device. This device is the product of a three-axis gyroscope and a three-axis accelerometer.
- The ICM-20602 is a complementary metal–oxide–semiconductor microelectromechanical system (CMOS_MEMS) that is able to operate using between 1.71 to 3.45V.
- It is expected that in order to satisfy the engineering requirement of achieving a gravitational force of 10^{-1} G we must be able to measure to that degree of accuracy. Depending on the mode set for the accelerometer, there are different levels of application that can change the inherit accuracy of the readings provided by the accelerometer for gravitational forces.
- Setting the accelerometer to AFS_SEL=0, we have access to fourteen-bit resolution with a range of plus or minus 2
- This resolution gives 16,384 least significant bit per g, which allows us to receive measurements to the accuracy of about .244 milli-Gs.
- Table 4 below outlines the specifications of flight controller accelerometer in the different selection modes



PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS	NOTES
ACCELEROMETER SENSITIVITY						
Full-Scale Range	AFS_SEL=0		±2		*g*	2
	AFS_SEL=1		±4		*g*	2
	AFS_SEL=2		±8		*g*	2
	AFS_SEL=3		±16		*g*	2
ADC Word Length	Output in two's complement format		16		bits	2
Sensitivity Scale Factor	AFS_SEL=0		16,384		LSB/*g*	2
	AFS_SEL=1		8,192		LSB/*g*	2
	AFS_SEL=2		4,096		LSB/*g*	2
	AFS_SEL=3		2,048		LSB/*g*	2
Sensitivity Scale Factor Initial Tolerance	Component-level		±1		%	1
Sensitivity Change vs. Temperature	-40°C to +85°C		±1.5		%	1
Nonlinearity	Best Fit Straight Line		±0.3		%	1
Cross-Axis Sensitivity			±1		%	1
ZERO-G OUTPUT						
Initial Tolerance	Component-level, all axes		±25		mg	1
	Board-level, all axes		±40		mg	1
Zero-G Level Change vs. Temperature	-40°C to +85°C X and Y axes		±0.5		mg/ºC	1
	Z axis		±1		mg/ºC	1
OTHER PARAMETERS						
Power Spectral Density	@ 10 Hz		100		µg/√Hz	1, 3
RMS Noise	Bandwidth = 100 Hz		1.0		mg-rms	1, 3
Low-Pass Filter Response	Programmable Range	5		218	Hz	2
Accelerometer Startup Time	From sleep mode to valid data		10	20	ms	2
Output Data Rate	Low-Noise mode	3.91		4000	Hz	2
	Low Power Mode	3.91		500	Hz	

Table 4: ICM-20602 Accelerometer Datasheet

3.6 Simulation

A series of tests are necessary in order to adjust and adapt the firmware changes of the flight controller and the implantation of autopilot directives present in the Racing F4 EVO Flight Controller. In an effort to test the programming of the flight controller before the final development of the final drone project as created by the Mechanical and Aerospace teams, as well as preserve the integrity of the final product, the Electrical and Computer engineering branch of this senior design project has elected the use of simulation software.

For the sake of simulation, we have selected to use FlightGear and Matlab Simulink. FlightGear enables a fully-fledged simulation of our craft in certain simulated scenarios. The use of simulation technology like FlightGear opens the opportunity to continually check any updates made to the design of our drone. As the drone itself is a costly endeavor, this allows for testing without risking any damage to the physical drone that is attached to the electrical components for the end product. Matlab is a widely-used and well supported by Mathworks in addition to 3rd party applications and add-ons. The interface is simple and allows easy access to live scripts, Simulink examples, and help/tutorial guides for many relevant topics pertaining to a quadrotor simulation.

Our reduced gravity unmanned aerial system, as defined by the engineering design constraints, is scheduled to reach nearly 400 feet above the ground and perform controlled and precise movements. A key part of these maneuvers is the presence of freefall. The lack of control or the occurrence of an error in the flight controller would leave the craft spiraling downwards towards a certain destruction. Testing different control system approaches on a physically craft could equate to the accumulation of cost for necessary components that would consistently break upon impact. The motivation behind simulating the aircraft and reduced gravity environment is two-fold: the ability to test various control system approaches to best obtain clean reduced gravity, and ironing out kinks in our flight dynamics, system identification, and reduced gravity trajectories that could cause our quadrotor to become unstable. That latter case is an effort by our interdisciplinary engineering team to prevent collisions to the craft. This practice greatly increases the chance of success on the fun on the final craft. The quadcopter drone needs to be able to remain stable through the freefall while also being able to retain enough control to have a safe landing.

Reusability and cost are two major factors that this reduced gravity drone is tasked to solve. Meticulous testing through simulation ensures the future reusability of the craft by lowering the number of critical failures to the craft through testing. Money is also saved on the craft by not needing to produce another physical drone every time an experiment needs to be run. A system that crashes frequently also runs the unacceptable risk of destroying not only the materials used in the experiment, but also any potential data that could have been recorded throughout the duration of the flight.

FlightGear allows for the simulation of the flight mechanics of the drone and provides a medium to perfect the intricacies of stability and the use of reverse and forward thrust needed for a stable flight path and parabolic trajectory as well as knowing how long and precise the drone will have in reduced gravity.

3.6.1 Equations of Motion, Modelling, Flight Dynamics

A proper aircraft flight dynamics equation set must take information from the actual designed craft. Mechanical and Aerospace have defined an aircraft model in Solidworks and have printed CAD files detailing dimensions and specifications of the brushless DC motor mount, experimental chamber, drone upper assembly, drone center mount, and drone arms. A simple graphic depicting the Green team aircraft’s shape and overall configuration can be found in figure 6 below:

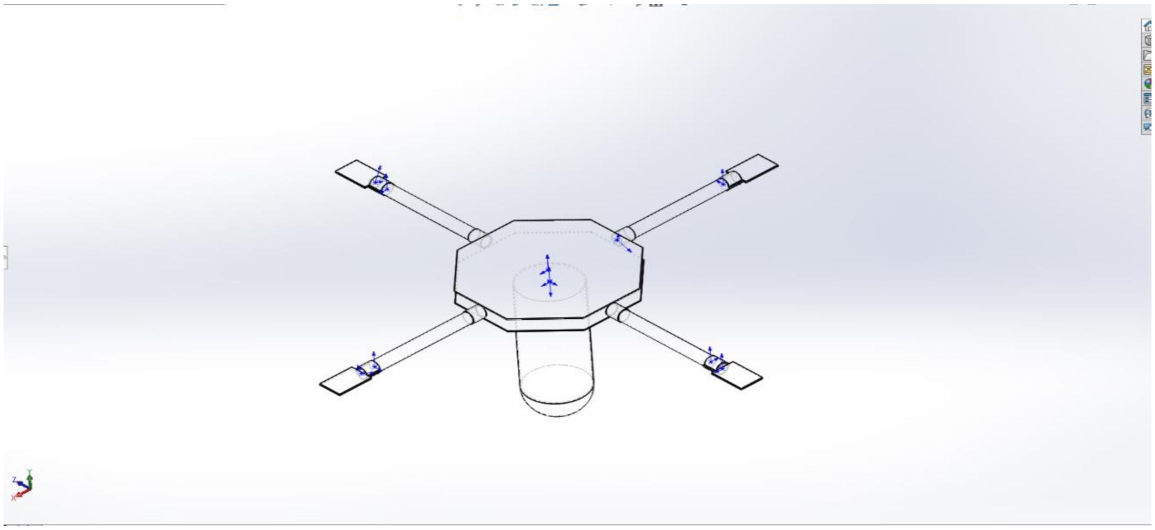
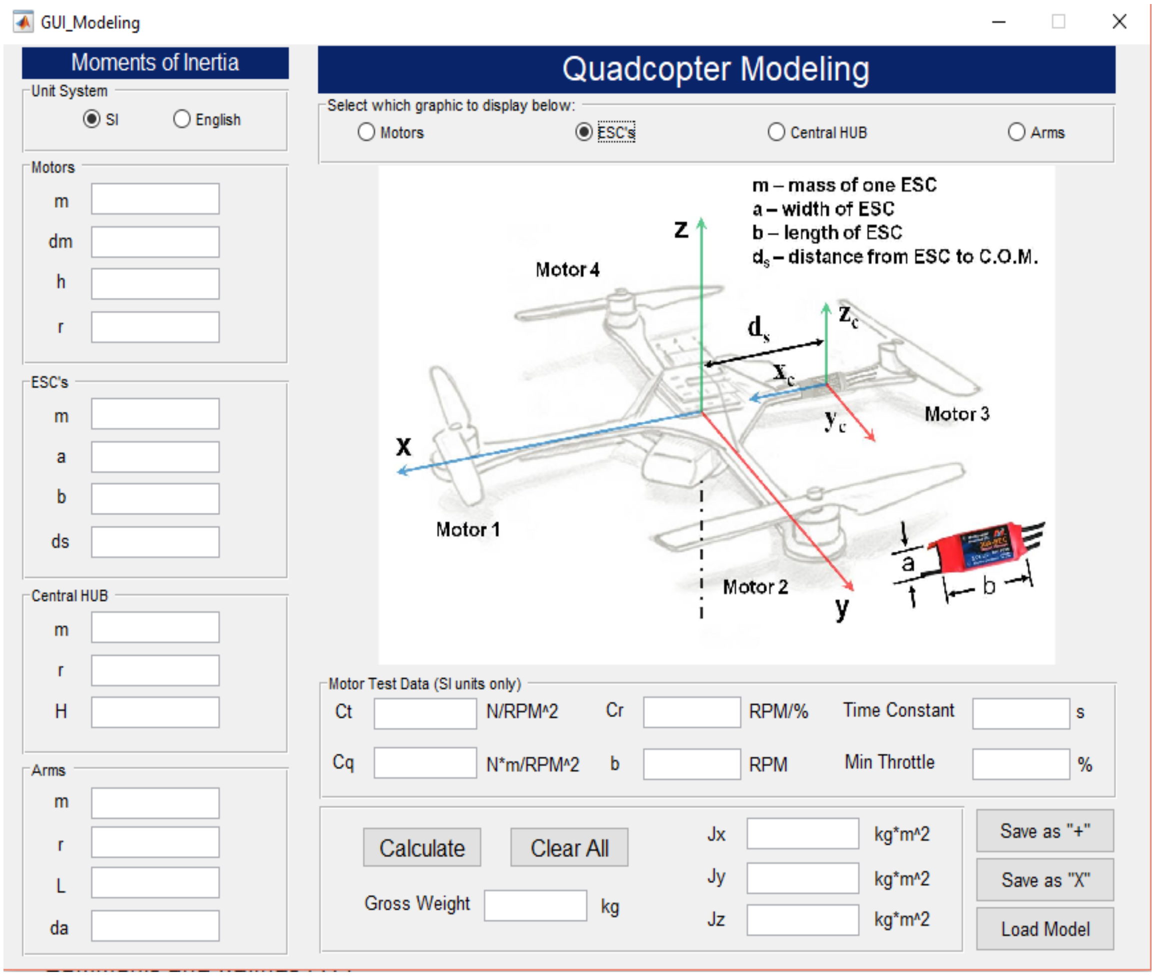


Figure 6: Reduced Gravity Quadrotor Aircraft Structure Design

From M&AE, we can determine dimensions from the CAD files, which are then inputted into a 3rd party GUI, known as Quad Sim [12]. This Matlab Simulink add-on is open source and allows modification to its original contents. Quad Sim includes an easy to use GUI for inertial moments calculations, from which we can obtain equations of motion through Newton-Euler equations of motion for a rigid body. Quad sim provides an inertial moment calculator as shown in figure 7 on the next page: note that the thrust and torque calculations for the motor must be obtained via bench testing and feeding results to the data analyzer in the Quad Sim GUI Matlab application. The application then outputs thrust and torque RPM relationships, percentage throttle to RPM, and time delay constant, finally resulting in the principles of inertial moments of the quadrotor system. Note that both motors and propellers are required for this procedure.



Unit System: ● SI ○ English

Select which graphic to display below: ○ Motors ● ESC's ○ Central HUB ○ Arms

Figure 7: Quad Sim Modeling- Moments of Inertia

After obtaining principle inertial moments, a dedicated mathematical effort is needed to determine direction cosine matrix (overall rotation matrix). This direction cosine matrix consists of Euler rates formulas, which include rotational vectors of the aircraft (yaw, pitch, and roll) and is a compounded rotational matrix. For these calculations, Quad Sim has another tool known as the Attitude Control block (AC). This block contains PID controllers for tuning yaw, pitch, roll, and altitude. This block allows open control system design, but provides a ready-made template for users to configure their quadrotor control system.

Another pertinent feature of Quad Sim includes the quadcopter dynamics block, which converts throttle command to motor RPM: Matlab state equations are calculated here. Matlab code function blocks utilize the throttle input and RPM values of all four motors for analysis. This block includes a 3D simulation that shows the attitude of the quadcopter in response to throttle commands and positioning. A position control system GUI is useful in understanding quadcopter behaviors at different parts of the flight envelope, allowing our team to evaluate any potential instabilities in the flight envelope and developing control system tuning to avoid these instabilities in the flight pattern. Figure 8 shows the attitude control block-sets, where Phi, Theta, and Psi are roll, pitch, and yaw, respectively (attitude controller-blue), along with altitude feed into control mixing. It can be seen in the figure that Attitude controller is in a feedback loop originating at the states and output (y-out). The quadcopter control mixing block (beige) determines which motors are required per the inputs of roll, pitch, yaw, and altitude: e.g. mc1-mc4 are required for altitude inputs. Finally, the previous two blocks feed into the red quadcopter dynamics block.

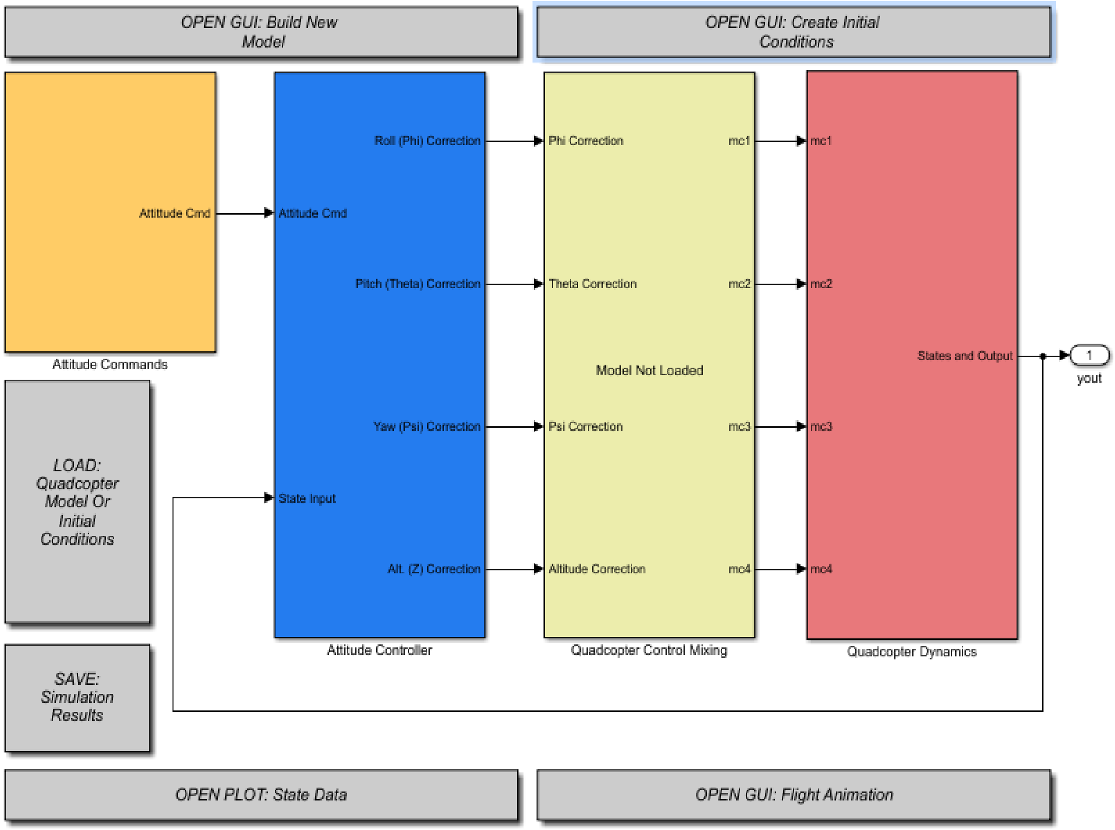


Figure 8: Quad Sim Attitude Control Blocks

4.0 Design Overview

4.1 Design Platform

By researching prior successes and failures in attempts at utilizing UAVs for this purpose, a multi-rotor copter design has been selected as the case for further development. An aerodynamic, bottom-heavy vehicle with concealed testing chamber and limited protruding surfaces is the concept for optimization. The vehicle, while at rest or during takeoff and landing, will utilize an oval base similar to an egg holder. The desired design will offer relatively easy access to the battery and experimental chamber. The desired testing sequence will be fully programmed, omitting the need for a pilot except in emergency circumstances. The most difficult obstacle will be maintaining vehicle stability in dynamic, unpredictable conditions. The need to create repeatable experimental conditions will require a great deal of pre-flight simulation and live flight testing to produce the proper programmed responses to flight perturbations.

Research to date indicates a maximum lifting capacity of 20 pounds due to financial constraints. The team believes it is possible to build a quadcopter with fixed rotors and programmable flight controller within these limiting parameters, however specific product selections have not yet been determined. In an effort to avoid fabrication expenses, a pre-made carbon fiber quadcopter frame retrofitted to carry an aerodynamically designed vehicle body capable of housing a sealed experimental chamber and necessary components is being considered. The body conceptual design is cylindrical with a teardrop formed lower section with plexiglass and aluminum alloy materials being considered. The late addition of the electrical engineering team members has resulted in a recent revelation of additional expenditures and complications not previously considered, but also added much needed clarification to the control system programming requirements. A detailed review of all contact surfaces as well as force analysis is currently being conducted to aid in final material and design selections.

4.2 Black Box (PCB) Design

In addition to an off-the-shelf flight controller, GPS module, and telemetry transceivers, the ECE team has elected to implement a small, lightweight, robust black box flight recorder. The recorder is self-contained and data logs without wireless transmission in order to reduce printed circuit board complexity and wireless interference with other signals (e.g. telemetry, flight controller remote). The design is based around an ATMEGA 328P chip, an 8-bit microcontroller primarily used in the popular Arduino R3 Uno prototyping board. For this application, we will not be using the prototype board, but the microchip the board is based around.

The eagle cad electrical schematic is given in figure below. The AVR microcontroller is a RISC (reduced instruction set count) low power CMOS. It is plenty robust for this small application involving only a 9 axis IMU, an Invensense MPU9250 (accelerometer, gyroscope, magnetometer) and a datalogging unit (SD card storage). However, the microcontroller only contains 32 kilo-bytes of programmable flash memory and 1 kilobyte of EEPROM, leaving few options for long term data storage. Hence, we've added an SD card storage peripheral into the PCB. The SD card schematic is based on an Adafruit breakout board, which can directly connect to the ATmega328P via integrated level shifting and a 5v to 3v regulator.

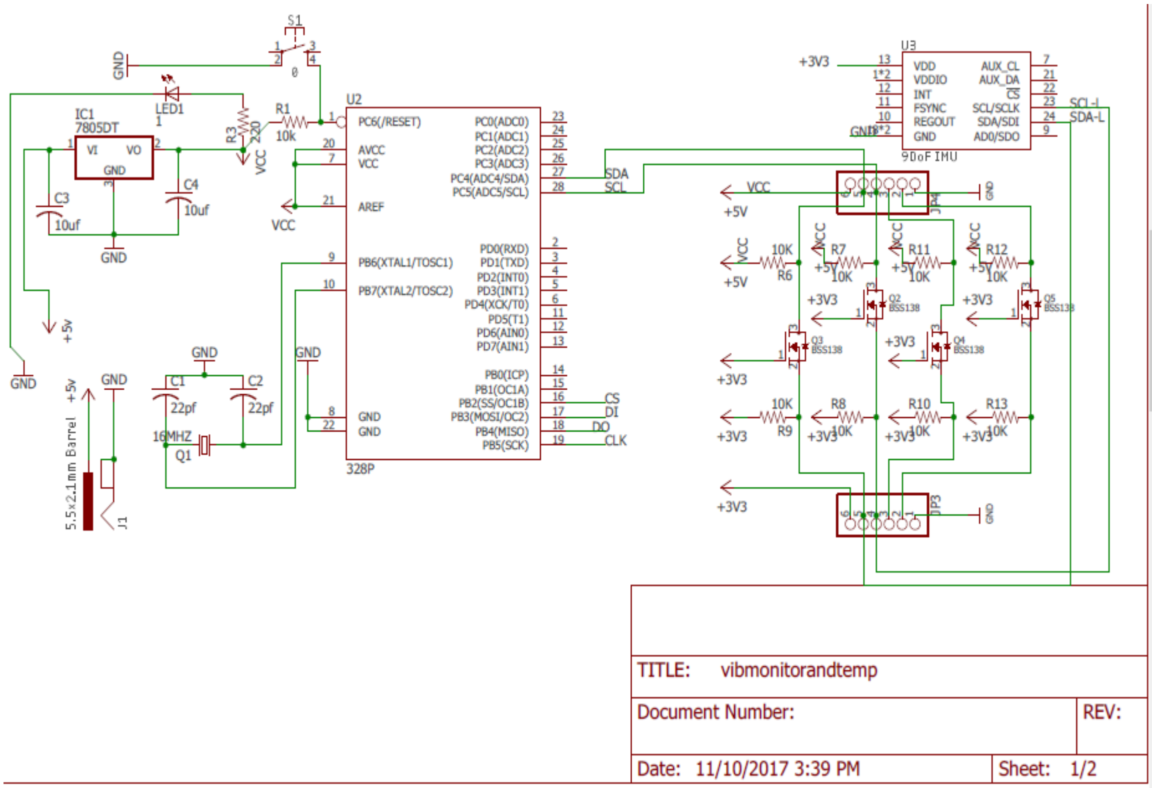


Figure 9: Eagle CAD electrical schematic diagram, black box flight sensors

As shown above, the PCB entails several protection features for the microcontroller and MPU 9250 9 axis inertial measurement unit. The microcontroller is powered via 5 volts, and given the self-contained nature of a black box recorder, ECE has decided to internally power the PCB with a 9-volt non-rechargeable battery, plugged into the PCB protection circuit via DC barrel jack connector. The protection circuit consists of an LM7805 linear voltage regulator which drops the 9 volts down to 5, which can then power the microcontroller. Additionally, an LED is soldered to the PCB protection circuit to signal the circuit is working. Next, a 16 MHZ quartz crystal is added to the microcontroller for better clock timing (the internal Atmega328P does include an internal clock albeit less precise than a quartz crystal). Finally, a reset button wired to ground (pulls reset pin to ground, resets bootloader) and a bridge circuit in between the MPU9250 accelerometer and microcontroller are added. The reason for this bridge circuit is to serve as a bi-directional level shifter between the 3.3-volt sensor board and the 5v atmega328: without this level shifter, the I2C data sent back and forth at different high and low signals will cause either chip to fry. The level shifter is a BSS138 4 channel I2C safe logic level controller, which can be purchased from Adafruit as a breakout board for the Arduino.

Shown in Figure 10 below is the SD & MMC breakout schematic, based on Adafruit's breakout board for prototyping with Arduino boards. However, this circuit's footprint will be combined with the above ATmega328P, protection system, and accelerometer PCB systems.

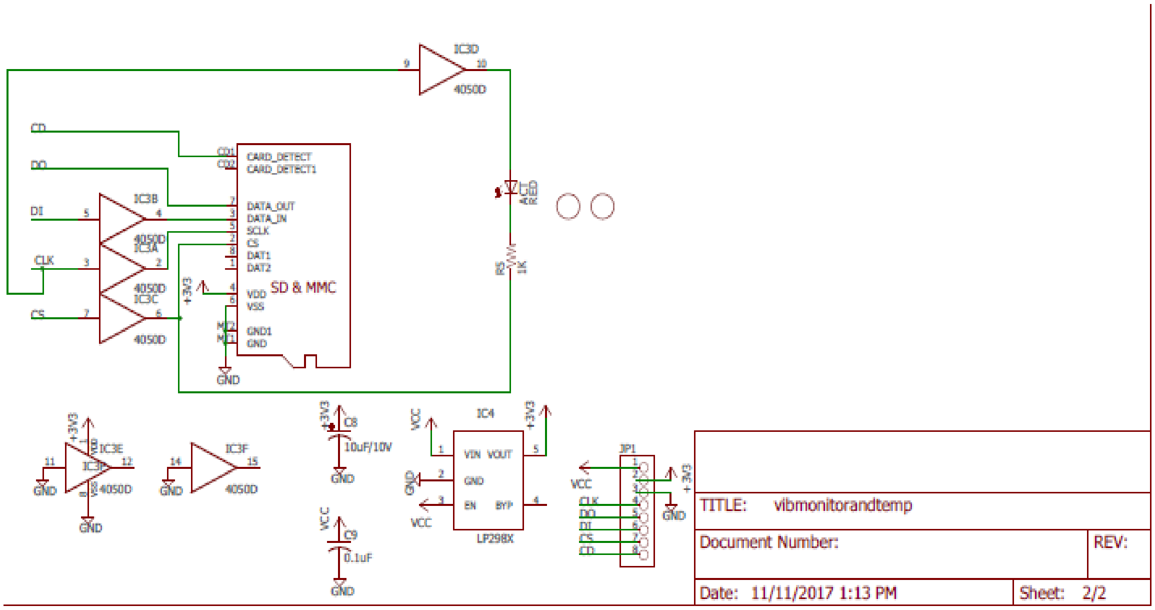


Figure 10: Eagle CAD electrical schematic diagram, black box storage

Adafruit's breakout board electrical schematic conveniently integrates a level shifter in the SD & MMC circuit for 3-volt to 5-volt and 5-volt to 3-volt level shifting between it and the ATmega328P. Adafruit's position on it's hardware and design is generally open-source, with only policies regarding reselling in tandem with modification of the design.

After combining these two schematics in Eagle, and choosing to generate the board, we are faced with placing the components so that they do not overlap and efficient traces can be made (bottom and top tracing). Eagle is generous to provide an auto-routing tool: in some cases it can route at 100% with all traces connected. For the black box circuit, it was much more complicated due to the SD card footprint and level shifting MOS components. The design of the PCB is centered around the 9 axis IMU and ATMEGA 328 microcontroller. Considerations in determining board layout began with the IMU, which should be placed away from large nets and clusters of traces. A design constraint that limited auto-routing was that the IMU could not be placed over traces on the bottom of the PCB due to interference with the magnetometer on board the IMU.

Fortunately the auto-router was able to complete 95.9% of the circuit, and only the connections to the IMU were still in need of manual routing. Via the use of vias, our team was able to route above and below other traces that could cause a short in the circuit. A revision two of this board layout is needed before printing at a fab-house as the JP1 header is further away from the rest of the circuit, creating a larger board area and potentially causing higher PCB fabrication cost. Also, the 5mm barrel jack is extremely close to the SD card holder at the bottom right corner of the PCB (figure 11): care must be taken in revision two of this circuit to create a larger distance between these components.

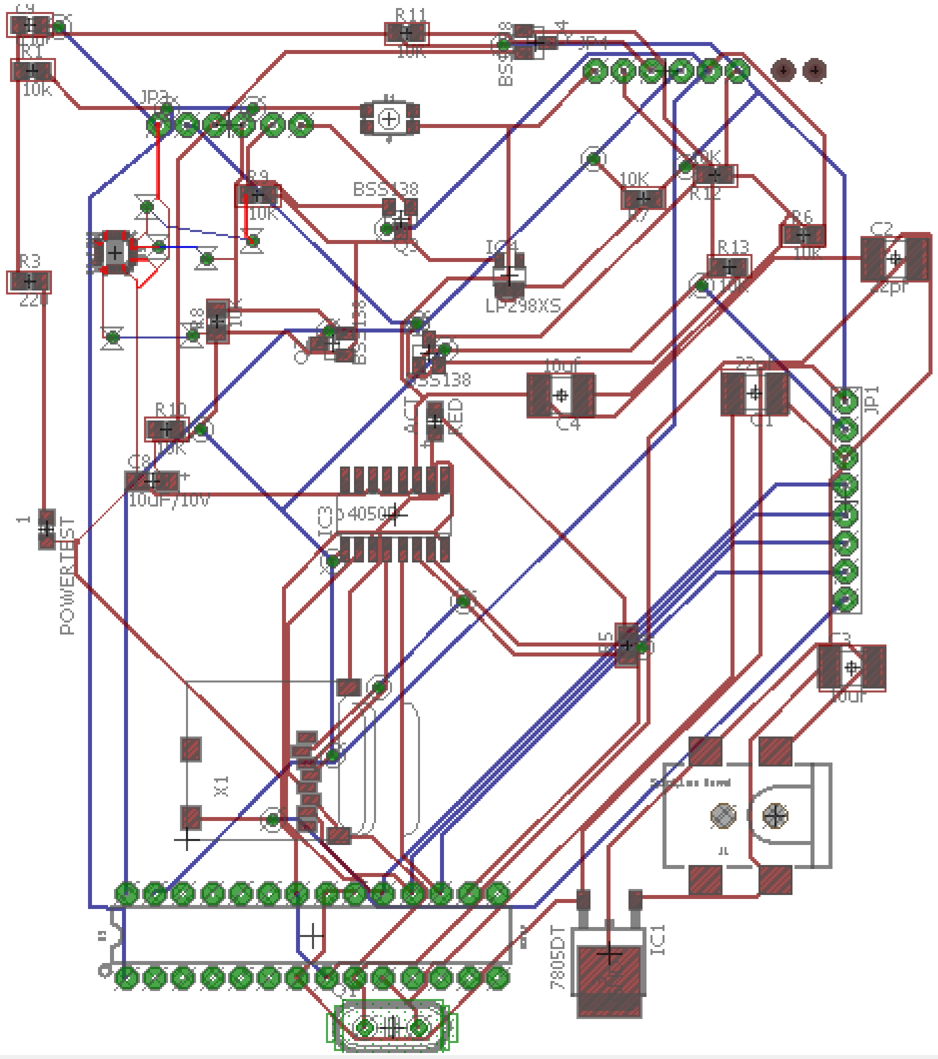


Figure 11 Printed Circuit Board Layout, Black Box Rev. 01

4.3 Hardware Assembly and Engineering Considerations

The board chosen for this project, based on cost, performance, and compatibility factors is the SP Racing Evo F4. This board is supported by the popular Clean Flight/Beta Flight configurator software. The board does not directly support autopilot or mission flying without flashing iNAV firmware onto the board, in addition to wiring in a compatible GPS/Compass to the board.

4.3.1 Instrumentation

The GPS of choice for most Clean Flight compatible boards is the Ublox M7/M8N, which is also primarily used in conjunction with Pixhawk autopilot boards. Fortunately, this GPS is relatively inexpensive and easily found, a graphic of the GPS is shown in Figure 12. This manufacturer's M8N GPS is accurate up to 2.5m and has a heading accuracy of 0.5°, velocity accuracy of 0.1 m/s, and with a maximum altitude ceiling of 50,000 meters. This GPS is perfect for this application, but it's accuracy is dependent on the nearby satellites and may vary based on location of the test flight: our team must carefully select a test flight area so that adequate GPS satellite coverage is available. The flash version of the Ublox M8N model (seen in figure 8) is ideal due to it's support of GNSS (Global navigation satellite system) support.

Figure 12: Ublox M8N GPS module

GNSS has the advantage of mass regional coverage for autonomous positioning systems that include GPS (32 medium Earth orbital satellites), GLONASS, Beidou, and a host of other regional positioning systems. This mass regional coverage ensures that if signal is lost with a particular GPS

satellite, others are available (crucial for mission flying and return to home functions in the Clean Flight configurator). GPS system accuracy and reliability in our reduced gravity platform is crucial in order to map out and follow parabolic ascent and dive maneuvers to facilitate a reduced gravity environment.

With the GPS module being separate from the SP Racing F4 board, it will be wired into UART1 port of the flight control board (5-volt signal compatible). If the GPS module has an integrated magnetometer/compass, it may be selected and configured in Clean Flight as long as it Is not the exact model integrated on the SP racing board. Now that the GPS module has been connected, we proceed to connect the power distribution board (PDB).

Following the wiring of ground and positive terminals of the PDB, it is now necessary to connect each of the four electronic speed controllers to the PDB so that they can be powered from the battery and can send and receive signals from telemetry and/or the transceiver (physical flight controller transmitter). The ESC PWM signal wires originate on the SP racing board (PWM channels 1-8). Additionally, the ESC will connect to its corresponding motor via three-wire connection. Manual control through the flight transmitter will only be necessary for initial low altitude hover of the aircraft or emergency flying situations (GPS signal lost). The SBUS/PPM receiver is connected to the SP Racing board via UART 2/PPM and configured in Clean Flight. Now that the SP flight control board, transceiver, PDB, and ESC’s are connected, it is time to connect the board to the ground station (laptop) and configure the board and its sensors. The update model specific firmware for this specific board must be flashed to the board upon entering the Clean Flight configurator (bootloader mode, Clean Flight is a chrome application GUI). The magnetometer and accelerometer must be calibrated manually, an easy but important step in the process.

A Bluetooth telemetry model is optional, but not required to upload mission flying and autonomous commands pre-flight through the Ez-GUI android application (which synchronizes with Clean Flight and Betaflight), since a micro USB-to-USB cable may be used.

Proper placement of the magnetometer is crucial to heading and compass readings for autonomous flight, return to home, and altitude hold functions. A magnetometer directly measures magnetic field, or the change of magnetic field based on location (dynamic). Since our project is a battery powered, fully electric quadrotor design, there are current carrying wires throughout the platform. The ESC wires draw up to and exceeding approximately 50 amps +/- 10. Our platform speed controller battery wires will be anywhere from 12-18 AWG wiring. We know from electromagnetic theory that a current carrying wire emits a magnetic field that can be represented by curling the right hand with the thumb facing the direction of the current. The magnetic field produced from a current carrying wire results in a cross product, known as the Biot-Savart law:

$$d\boldsymbol{B} = \frac{\mu_0 I d\boldsymbol{L} x \boldsymbol{l}_r}{4\pi r^2}$$

Through applying Ampere’s law to the Biot-savart law, we can estimate the magnetic field produced by a straight current carrying wire:

$$B = \frac{\mu o I}{2\pi r}$$

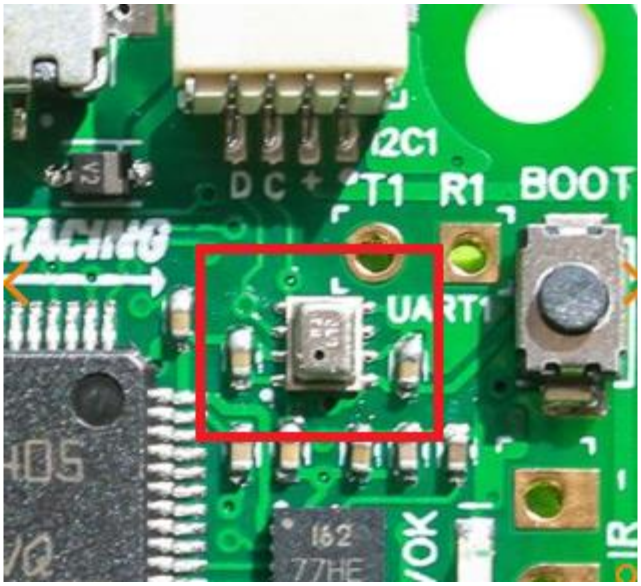
Where mu-naught is the permeability of free space, and *I* is the current in the wire, while *r* is the radius expanding out from the wire. It can be seen through this equation that at a far enough distance *r* from the current carrying wire, the magnetic field strength attenuates. Additionally, the force of a current carrying wire by a magnetic field can be approximated via:

$$\boldsymbol{F} = \boldsymbol{I} L x \boldsymbol{B}$$

Note that a Lithium polymer power source with a discharge ‘C’ rate of 60C, and a capacity of 2200mAH will output a max continuous current of 132 Amps, although this is probably an extreme case, it is necessary to factor in max current draw and it’s effects on magnetometer performance. Unfortunately, shielding is a moot solution due to the shielding interfering with measurement of the earth’s magnetic field (for sensor heading and autonomous flight modes). Distance from the battery and current carrying wires must be calibrated for mitigation of electromagnetic interference: the HMC5983 is an abundant MEMS magnetometer found in M8N model GPS units similar to the proposed unit in the bill of materials list found in section 3. It can measure a field range of -8 to 8 gauss, which is equal to a 1.6x10^-3 tesla range. Given the permeability of free space as 4πx10^-7T*m/A, the max continuous current in the battery wire feeding into the power distribution board, and the approximate range of gauss readings by the magnetometer, we calculate the minimum distance *r*, from the wire, should be greater than or equal to 1.65 cm (roughly 1”) to mitigate interference. We may also work backwards from trace current (e.g. from PDB to GPS & Magnetometer sensor) wire to calculate maximum current in the trace wire to prevent interference.

An easy configuration for the magnetometer integrated in the M8N GPS is situated above the quadrotor on a small diameter cylindrical support made of PVC or plastic materials. In addition to mounting the magnetometer, the sensor must also be calibrated away from large metallic objects or transmitting/receiving equipment in the general vicinity. On the basis of metal objects or other magnetic interference during magnetometer calibration, commercial quadrotor products advise to calibrate the compass outdoors only.

Figure 13 : BMP280 barometer sensor mounted to flight control



Before calibration of the barometer, careful consideration must be given to shielding or baffling the barometer. The barometer used in quadrotor and model plane applications is necessary for altitude readings and the aircraft's positioning in 3D space. The barometer is highly sensitive to light exposure (sun light) and wind gusts or drafts. The barometer on the SP racing F4 Evo is a Bosch made BMP280 piezo-resistive barometer that reads barometric pressure and altitude: it is primarily used as an augmentation to GPS navigation modes. The BMP280 exhibits an absolute accuracy of +/-1 hPa and interfaces via I2c or SPI. The barometer mounted to the SP racing F4 board is pictured above in figure 9. Our solution to protecting this exposed sensor is a miniature Stevenson screen, made of a thin wood or UV rated plastic box enclosure with vent slits on all sides. This enclosure will be glued, covering the sensor open face down with a crafts person hot glue. One vent per side may be adequate given the miniature size of the enclosure. The vent can be either drilled into the side panels or dremeled in.

The BMP280 data manual gives several guidelines for mounting considerations including venting for ambient pressure readings and protection from light exposure (corrupts accuracy of measurement.

4.3.2 Power Distribution (contingency)

In ECE's bill of materials, we've provided a power distribution board spec from an off the shelf, hobby grade retailer. However, in investigating different motor and propeller combinations, M&AE have discovered that our current motor selection in conjunction with a 22" prop will draw over 57 Amps at greater than 85% throttle. This poses an engineering challenge as the selected Lumenier 6S PDB is only rated to carry 46 Amp bursts on each of it's ESC traces for 5 seconds at a time. There is a + 10 A difference, which is significant, considering the entire flight envelope of the unmanned aerial system is 12.43 seconds. If a considerable amount of flight envelope time is spent above 85% throttle, greater than 5 seconds, there is a chance the Lumenier PDB (figure 14) traces can burn up, thus cutting power to the ESC and motors: this will cause a high speed collision impact into the earth with no failsafe.

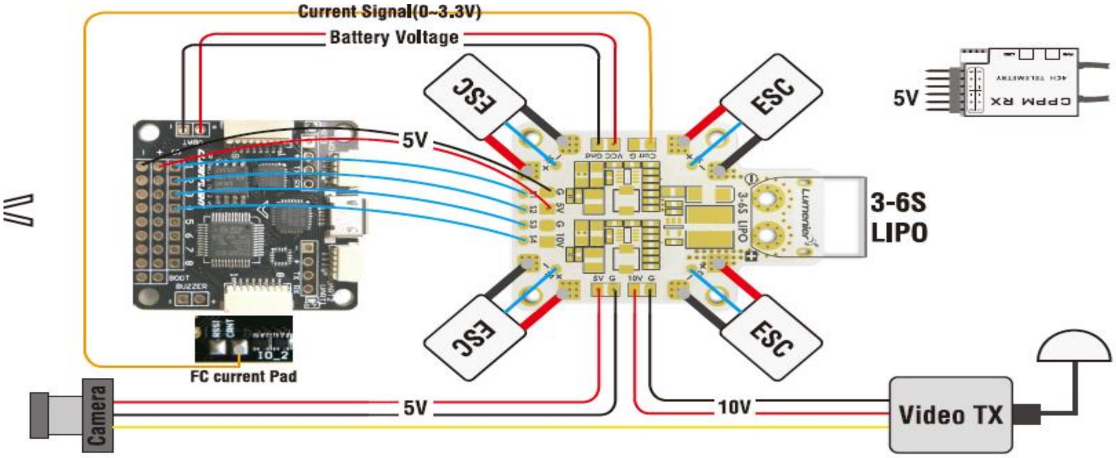


Figure 14: Power distribution Wiring Diagram

ECE has decided on 1) a preventative measure (bench testing motor and props for current draw), which will export motor rpm, max current draw, propeller size, and voltage to a spreadsheet for analysis, and 2) a custom PCB with robust traces to accommodate over 30 A continuous current. This custom fabricated PCB will have an onboard voltage regulator and solder pads for the flight control board 5v and ground lines. It will be of similar design to the Lumenier 6S PDW shown above. The custom will have a PCB of a similar footprint to the SP Racing Evo F4: a 30.5mm mounting hole on all four corners. The board is centered around a Texas Instruments LM25011-Q11 switching voltage regulator capable of 45 volts input: it spits out 5 volts output for logic level powering the SP Racing Evo. The LM25011 features adjustable frequency range up to 2MHZ This IC is an automotive grade component made to withstand harsh conditions and is suitable for distributing power to our reduced gravity drone. It can supply up to 2A of current and features low dropout voltage at high switching frequency. The LM package contains a power good pin for voltage level safety with low logic electronics. Our PCB centered around this board includes an inductor, resistors, and capacitors: the resistor combination on the output determines the output voltage of the circuit. Additionally, the custom PDB has four deans/XT60 connectors on each corner for easy plug and play with speed controllers as well as a deans/XT60 for the lithium polymer battery connection. It has solder pads for 5v pin out and ground pins.

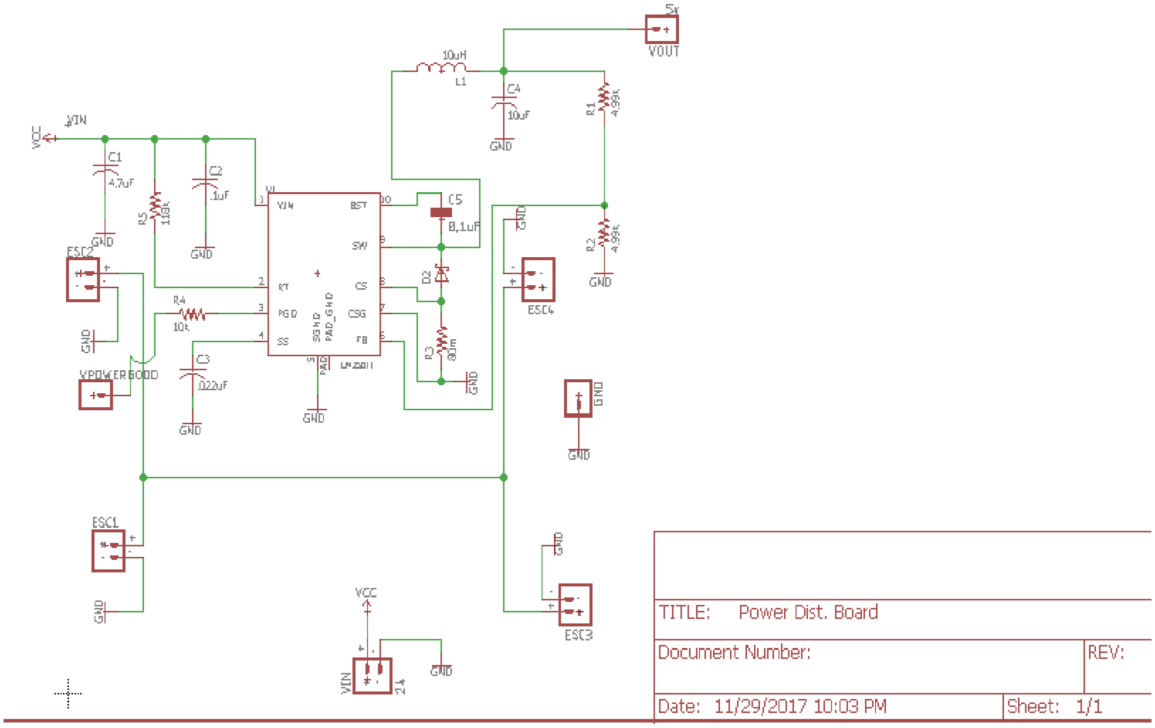


Figure 15: Power Distribution Board– Custom Fabricated-Switching Voltage Reg. Rev01

The board layout of this schematic is shown in the figure below (figure 16):

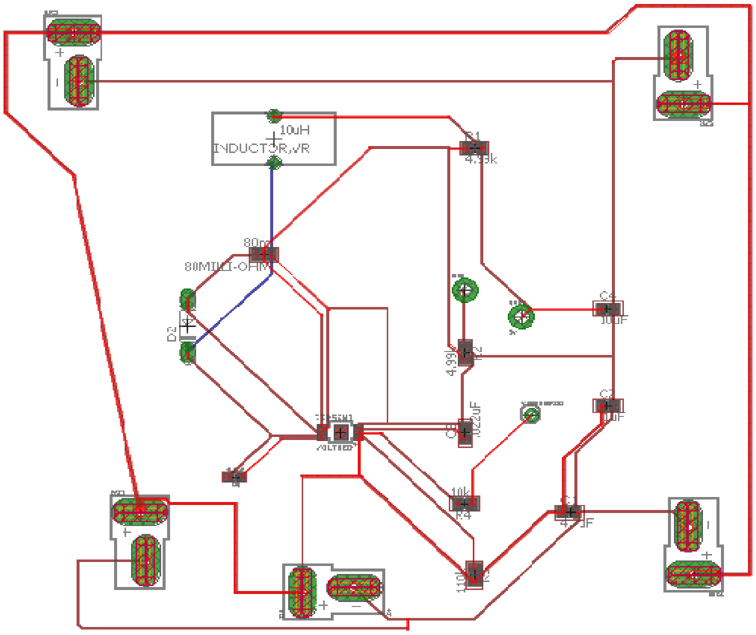
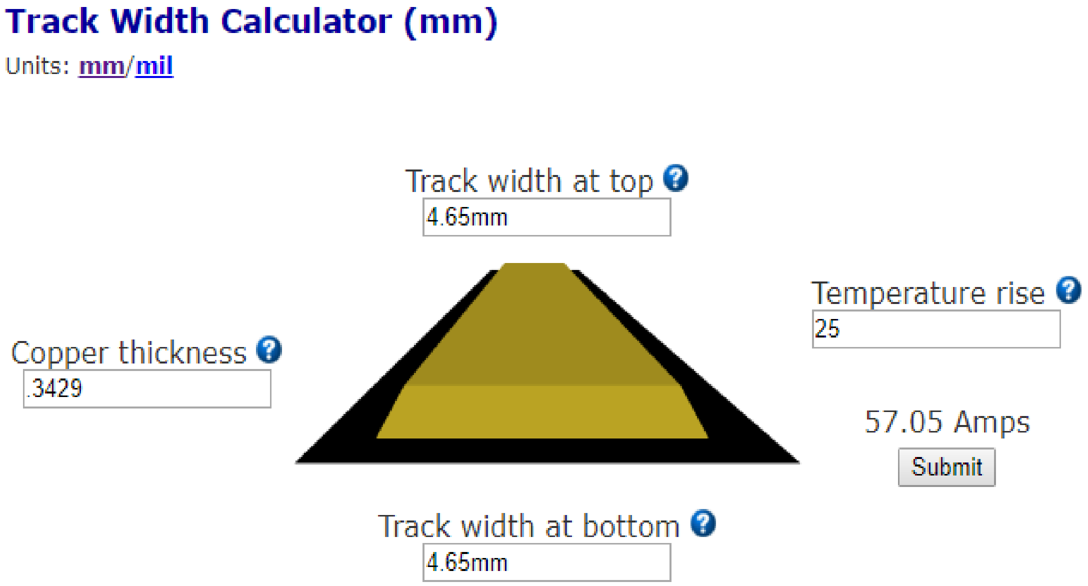


Figure 16 : Power Distribution Board Layout

4.3.2.1 High Current Trace Width Acommodation

As stated in the standards section of this report, IPC-2152 guides designers and engineers in printed circuit board current carrying parameters. For this project, our P60 KV340 motor and 22” propeller (spec’d out by MA&E) is projected to produce over 14lbs of thrust: this comes at a cost of up to and possibly exceeding 56 amps continuously for over 85% throttle. The selected Lumenier power distribution board is only rated for four ESC’s drawing 46 amps, at a burst rate of no more than 5 seconds. This is a delicate calculation and also depends on the flight envelope: hence, our custom printed circuit board is a contingency plan only, to be implemented if found that the Lumenier board is in-adequate during mere bench testing of the propeller, esc, and motors.

A trace width calculator was used (figure 17) to determine trace and copper dimensions needed to carry approximately 57 amps on our PCB. All dimensions are in mm, with the exception of temperature rise in °C. Copper weight/thickness is approximately 10 oz. Note that traces as wide as these may cause overlap without elements of the circuit and other traces, so care must be taken in the board layout section of Eagle.



Track width at top: 4.65mm
Copper thickness: .3429
Temperature rise: 25
57.05 Amps
Track width at bottom: 4.65mm

Figure 17: Trace Width Calculation for 57 A Carrying Capacity [13]

Issues arise when PCB fab houses can only facilitate certain copper weights and trace widths. Additionally, due to the large width and thickness requirements, the physical PCB will be have more depth than an average printed board. It will also carry a heavier weight than standard PCB’s. Contact has been made to two fab houses to ascertain their manufacturing capability in regards to large trace width and high copper thickness. The selected fab house details may be found in the Consultants, subcontractors, and supplier section of this report.

4.4 Flight Software, Mission Waypoints, Ground Control Station (GCS)

Following mounting of the sensors, it is necessary to configure the quadrotor (4 rotor) in Clean Flight as well as connect the motors (without propellers). Next, the receiver must be selected in the UART “ports” section of Clean Fight: RX_Serial Receiver must be checked, and the specific serial provide muster be selected in the drop-down menu. Now selecting the receiver tab will yield values for roll, pitch, yaw, throttle, etc. as depicted in figure 18 below [10]:

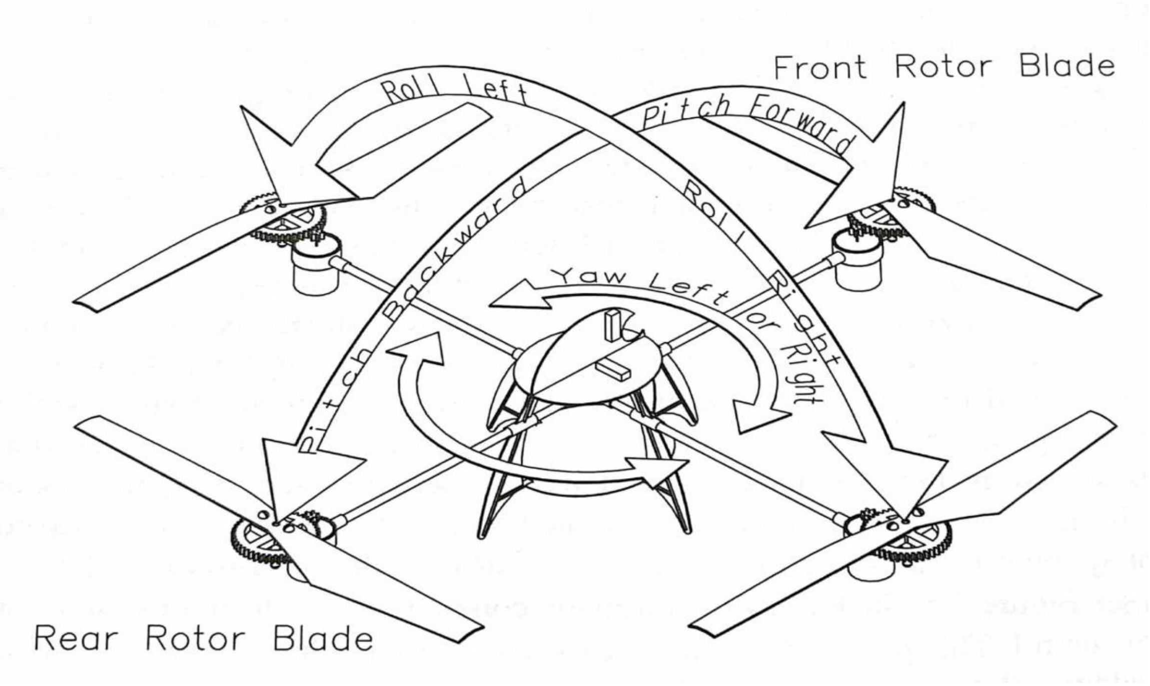


Figure 18: Pitch, Roll, and Yaw on a quadcopter frame [10]

Software implementation will facilitate many of the processes of the project and is a necessary backbone in the operation of the required tasks. For the purpose of this project, there will be the utilization of several different types of software that will interact with each other in a synchronized matter to manage the different hardware components, both electrical and mechanical in nature.

The development of the software is largely based from an assortment of already established firmware for the different electronic components that command the assortment of mechanical motors and actuators. There are also pockets of development to manage the tracing and mapping and returning specified data through the telemetry.

There is also a need for fail safes in the case that some feature of the airborne craft malfunctions or acts differently than planned during a mission. The quadcopter has the ability to abort when the drone is registering to be unstable. The programmed flight will also trigger the abortion of the mission if the craft drifts beyond the predefined flight window established by the autopilot. In the case of a big failure in navigation and control systems, the flight has an override switch where autopilot protocol is terminated and the drone can be manually piloted by a person on the ground.

The combination of these three programs on the flight controller creates a valid stability control system in which the drone is to operate. The stability control system must be able to receive the altitude from the altimeter sensor, receive position based on the global position-monitoring system [GPS], receive the acceleration from the accelerometer or the inertial measurement unit [IMU] at a high resolution, measure the rate of angular change in up to three axes, and measure velocity. All these measurements lead to the development of a system that gives the quadcopter access to a protocol which enables rapid stability response, knowing where and how to land safely, stay on a programmed trajectory, and properly prioritize stability response to ensure constant stability and acceleration during experimental maneuver.

4.4.1 Flight Controller Firmware

The SP Racing F4 Evo flight controller requires the cooperation and synchronization between three different flight navigation programs. During the development of the SP Racing F4 Evo, the initial firmware used to control the craft allowed for certain features over others. With the development of new software, certain features brought upon by the use of the firmware has been altered throughout its course. To achieve maximum usability during the implementation of the board with our product, the SP Racing F4 Evo board will simultaneously be running three different firmware.

4.4.1.1 CleanFlight

The first of these firmware is the open source CleanFlight firmware. This piece of software will be the base firmware in processes regarding flight performance, sensor monitoring, telemetry logging and protocol, and third-party support. CleanFlight allows for an arrangement of tools to be integrated a modified to accommodate the motors, sensors, and PCB involved in the design of the quadcopter.

The basic firmware code is structured in the C programming language. Changes for the firmware is necessary to allow for the implementation of the other two firmware in parallel with CleanFlight as well as creating and allocating the functionality of the various other components.

CleanFlight allows for the configuration of the flight mechanics that will need to be set and adjusted before and during the flight. The program will allow for the control of the motors and the rudders. While working in conjunction with other firmware, the quadcopter drone will autonomously control the thrust and angle that has been planned prior to launch. Running with this firmware on the flight controller enables the craft to follow the planned trajectory to achieve a reduced gravity environment suitable for testing.

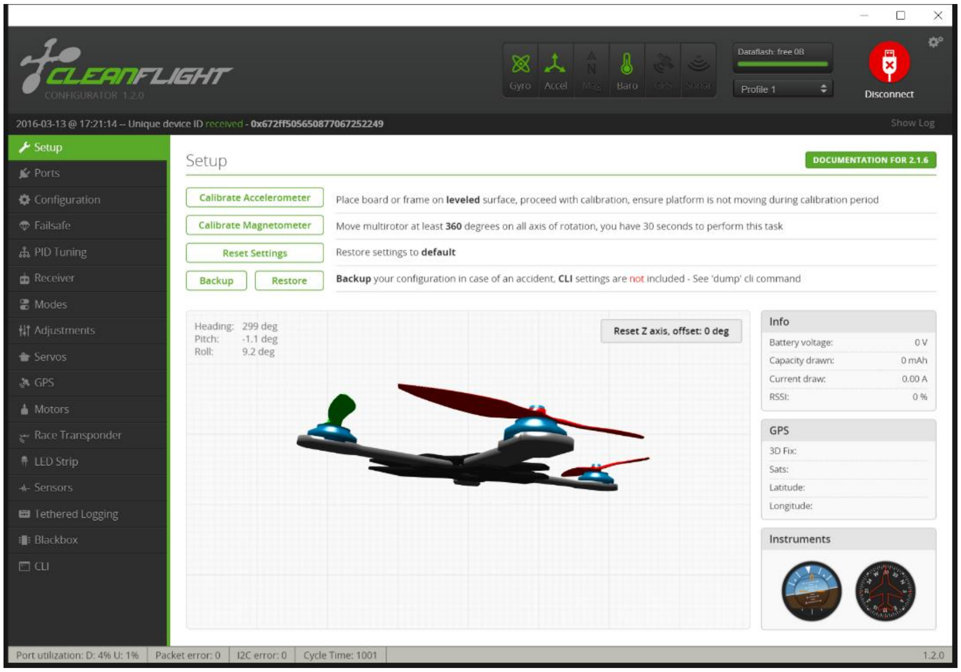
A feature present in this drone, as well as certain classes of racing drones, require the use of reverse thrust and control for the motors. As stated in Section 3.2, reverse thrust is a unique mechanic that increases the stability of the flight and quality of the reduced gravity environment required for experimentation in this project. By manipulating different sections of the configuration, reverse thrust will be implemented and adjusted in accordance with the readings from the sensors.

The different sensors present in the flight controller can be read and logged by CleanFlight and sent back down to the front-end GUI on the ground. The sensor readings will log information from the flight, but also serve as a trigger for multiple ways in which the way the drone handles will be adjusted to achieve the desired results.

In order to accurately test the use of CleanFlight for our drone, we must first step through the initial measures for configuration. The initial set up when dealing with a flight controller that incorporates this software is to loosen the prop nuts so that the prop can spin with the motor without the need for tuning the motor or worrying that the drone may take flight during configuration.

Once the battery is attached, the receiver for the RC telemetry will be bind to the drone and configured to allow for manual operation if needed. The quadcopter will need to be configured with the CleanFlight configurator. The configurator, along with other features, allows for a version of Betaflight (more detail on this software program stated in the section below) to be loaded onto the flight controller. These firmware integrate and work together on the flight controller to allow for the most stable flight. The Betaflight firmware is flashed onto the board through the use of the “Flash Firmware” feature present in the CleanFlight software.

The next step is connecting the quadcopter to the configurator in order to verify the correctness and responsivity of the gyroscopes present on the drone. The figure below, as shown on Prop-washed [14], will appear. This 3D model, should the gyros be responding as expected, will rotate and tilt in accordance to manipulation with the physical craft.



Dataflash: free 0B

2016-03-13 @ 17:21:14 -- Unique device ID received - **0x672ff505650877067252249**

Place board or frame on **leveled** surface, proceed with calibration, ensure platform is not moving during calibration period

Move multirotor at least **360** degrees on all axis of rotation, you have 30 seconds to perform this task

Restore settings to **default**

Backup your configuration in case of an accident, **CLI** settings are not included - See 'dump' cli command

Heading: 299 deg
Pitch: -1.1 deg
Roll: 9.2 deg

Battery voltage: 0 V
Capacity drawn: 0 mAh
Current draw: 0.00 A
RSSI: 0 %

3D Fix:
Sats:
Latitude:
Longitude:

Port utilization: D: 4% U: 1% | Packet error: 0 | I2C error: 0 | Cycle Time: 1001 | 1.2.0

Figure 19: 3D model of Drone during Gyro Configuration

An important step after the configuration of the gyros is the configuration of the RC receiver. The UART port of the flight controller that is connected to the RC receiver will have to be set to “RX”, while all other switches on that slot are switched off. The receiver mode and serial receiver provider must be set to the appropriate settings to allow for the communication between the drone and the manual control of those on the ground. Each portion of the radio needs to be mapped to a different function or feature of the drone through the use of the channel map mode. The throws for the radio must go from 1000 to 2000 or adjusted until the previous statement is made true. For the sake of testing, the mid and high end of the throttle can be recorded for use later. A flight mode, preferable Air Mode, needs to be configured to allow for the disbarment of the RC controller when not needed.

The drone is going to need to have the power systems configured in order to meet the demands of this drone. This process will take into account of the motors that need to be reversed. The placement of the motors has to be outlined in the CleanFlight program. Each motor can be tested one at a time by sliding the bar by the indicated numbered motor. This process clarifies that not only are the motors able to be controlled by the flight controller, but also if they line up with the numbered motors in the 3D diagram.

The general configuration portion in terms of CleanFlight are then completed. The drone will then have to go through additional setup through the use of the BLHeli software. This software sets up the correct configurations and parameters for the speed controller for the multiple rotors. This includes thrust, normal and reversed, and any needed adjustment to the overall power of the drone.

4.4.1.2 BLHeli

The BLHeli program, though separate from the CleanFlight configurator, can be used to connect and configure the onboard speed controller using a CleanFlight passthrough. BLHeli is a program that allows for flashing and configuring onto a speed controller. These configurations allow for certain settings being set, such as the capability for reversed motor function.

Once the program is opened, the user must select the correct interface that will properly interact with the CleanFlight software. This will lead to the selection of the “SILABS BLHeli Bootloader (Cleanflight).” The correct COM port will then be selected in order to communicate with the plugged-in speed controller.

The damped light mode needs to be enabled to allow for active braking. This setting makes it that if any of the speed controllers are operating a motor at full throttle and then receive a signal for half throttle then the motor will be halted using reverse thrust to quickly hit the desired half throttle threshold. This method is in contrast to letting drag slow the motor down, which in turn allows for a significantly responsive result. This mode has the possibility of causing a significant voltage jump when active and needs to be regulated. To make sure that all the motors are oriented correctly, any motor that was spinning the wrong way in the last step can be set to reverse thrust in the BLHeli program.

At this point, it’s important to disconnect from the BLHeli program and reconnect with the CleanFlight software in order to verify that the speed controller properly configured the motors to fly in the correct direction.

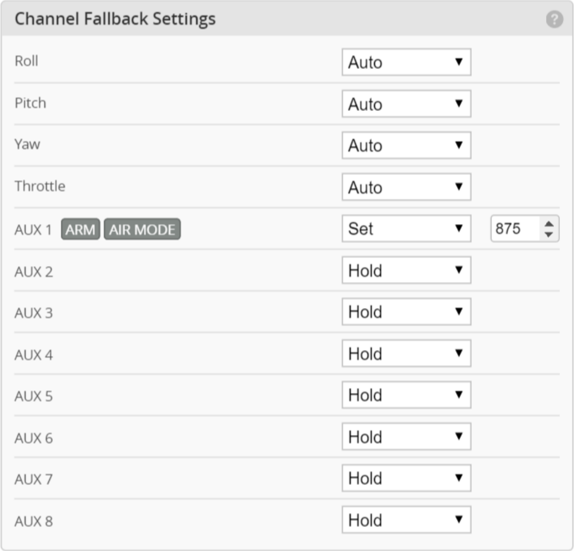
The throttle range will be set through the CleanFlight configurator. The minimum throttle is set to ten above the spool-up start point, while the middle and max point are set to their relative positions on the throttle stick.

The sensitivity rates of the quadcopter control how much the throttle and turn sticks of the RC remote have to be manipulated to achieve a certain degree of activation in the quadcopter. For quicker and more sensitive flipping, rolling, and yawing, the settings in the PID Tuning tab can be adjusted for the sensitivity and rotational speed of each axis of the drone. The initial PID values does not need to differ from the default values given to us by the Betaflight software.

The last setting that needs to be configured by the BLHeli configurator is the presence of a failsafe. This mechanism ensures a failsafe throttle pulse range value is set in order to disarm the system. This setting is adjusted through the failsafe tab present in the configurator.

The props will need to be removed in order to configure the failsafe to ensure a safer environment. The default minimum pulse value is set at 875. This value is the lowest possible pulse value. By shutting off the transmitter, the desired minimum pulse and be observed. This value set into the minimum pulse setting will be higher than the value shown when the transmitter is turned off, but lower than the lowest possible throttle value.

The quadcopter can also be configured to activate the failsafe if the drone ever loses connection with our RC connection. This functionality allows for the ground team to always have the ability to take control of the quadcopter when needed and have ensure that the quadcopter does not go too far outside the realm of connecting with the senior design team. The figure below shows how the channel fallback settings can be configured to allow for this feature to be enables.



Channel Fallback Settings

Roll	Auto	
Pitch	Auto	
Yaw	Auto	
Throttle	Auto	
AUX 1 ARM AIR MODE	Set	875
AUX 2	Hold	
AUX 3	Hold	
AUX 4	Hold	
AUX 5	Hold	
AUX 6	Hold	
AUX 7	Hold	
AUX 8	Hold	

Figure 20: Quadcopter Channel Fallback Settings [14]

4.4.1.3 Betaflight

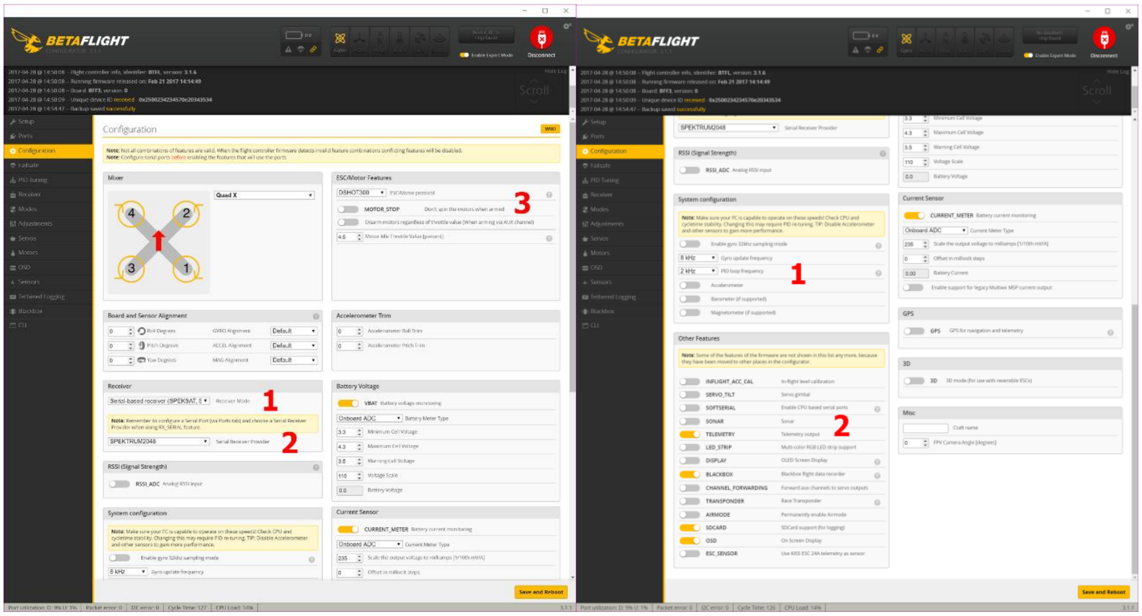
Betaflight is a firmware that is initially based from the CleanFlight program. Betaflight is hailed as far more experimental and therefore can have bugs and glitches that leave the program unstable. With this in mind, there are stable versions available that provide a more polished and finetuned look at what the original CleanFlight has to offer.

The benefits given by the use of Betaflight is the increased stability of the inflight drone. With an ease of maintaining a stable craft, integrating this firmware into the original CleanFlight software greatly increases the quality of flights and results from experimentation.

Betaflight’s robust on-screen display makes adjustment extremely easy for configuration purposes. The display itself can be configured and modified to display different aspects of the craft, from internal measurements of voltage, RPM of the motors, or the data drawn in from the sensors.

The Betaflight is the last of the programs that needs to be set up and configured to the flight controller. This process is after the configuration of both CleanFlight and BLHeli, and there for we make a backup in order to save the previous settings given to the flight controller by the previously named software.

The software will need to be toggled into expert mode to enable further access to some of the other features offered by Betaflight. The first step after this is setting the Serial Rx to be enabled on the UART3 port and save and reboot the system. For the remainder of the basic configurations needed to get off the ground, the figures below show the various of tweaks in the default settings that are needed.



2017-04-28 @ 14:50:08 -- Flight controller info, identifier: BTFL, version: 3.1.6
2017-04-28 @ 14:50:08 -- Running firmware released on: Feb 21 2017 14:14:49
2017-04-28 @ 14:50:08 -- Board: BFF3, version: 0
2017-04-28 @ 14:50:09 -- Unique device ID received - 0x2500234234570e20343534
2017-04-28 @ 14:54:47 -- Backup saved successfully

Configuration

Note: Not all combinations of features are valid. When the flight controller firmware detects invalid feature combinations conflicting features will be disabled.
Note: Configure serial ports before enabling the features that will use the ports.

Mixer: Quad X

ESC/Motor Features
DSHOT300 ESC/Motor protocol
MOTOR_STOP Don't spin the motors when armed
Disarm motors regardless of throttle value (When arming via AUX channel)
4.5 Motor Idle Throttle Value [percent]

Board and Sensor Alignment
0 Roll Degrees — GYRO Alignment: Default
0 Pitch Degrees — ACCEL Alignment: Default
0 Yaw Degrees — MAG Alignment: Default

Accelerometer Trim
0 Accelerometer Roll Trim
0 Accelerometer Pitch Trim

Receiver
Serial-based receiver (SPEKSAT, S Receiver Mode
Note: Remember to configure a Serial Port (via Ports tab) and choose a Serial Receiver Provider when using RX_SERIAL feature.
SPEKTRUM2048 Serial Receiver Provider

Battery Voltage
VBAT Battery voltage monitoring
Onboard ADC Battery Meter Type
3.3 Minimum Cell Voltage
4.3 Maximum Cell Voltage
3.5 Warning Cell Voltage
110 Voltage Scale
0.0 Battery Voltage

RSSI (Signal Strength)
RSSI_ADC Analog RSSI input

System configuration
Note: Make sure your FC is capable to operate on these speeds! Check CPU and cycletime stability. Changing this may require PID re-tuning. TIP: Disable Accelerometer and other sensors to gain more performance.
Enable gyro 32khz sampling mode
8 kHz Gyro update frequency
2 kHz PID loop frequency
Accelerometer
Barometer (if supported)
Magnetometer (if supported)

Current Sensor
CURRENT_METER Battery current monitoring
Onboard ADC Current Meter Type
235 Scale the output voltage to milliamps [1/10th mV/A]
0 Offset in millivolt steps
0.00 Battery Current
Enable support for legacy Multiwii MSP current output

Other Features
Note: Some of the features of the firmware are not shown in this list any more, because they have been moved to other places in the configurator.
INFLIGHT_ACC_CAL In-flight level calibration
SERVO_TILT Servo gimbal
SOFTSERIAL Enable CPU based serial ports
SONAR Sonar
TELEMETRY Telemetry output
LED_STRIP Multi-color RGB LED strip support
DISPLAY OLED Screen Display
BLACKBOX Blackbox flight data recorder
CHANNEL_FORWARDING Forward aux channels to servo outputs
TRANSPONDER Race Transponder
AIRMODE Permanently enable Airmode
SDCARD SDCard support (for logging)
OSD On Screen Display
ESC_SENSOR Use KISS ESC 24A telemetry as sensor

GPS
GPS GPS for navigation and telemetry

3D
3D 3D mode (for use with reversible ESCs)

Misc
Craft name
0 FPV Camera Angle [degrees]

Port utilization: D: 9% U: 1% | Packet error: 0 | I2C error: 0 | Cycle Time: 127 | CPU Load: 14% 3.1.1

Figure 21: Betaflight Configuration Tab [15]

4.4.1.4 INAV

INAV is a program that adds certain navigational features that the previously mentioned firmware lack. The presence of a strong navigational component provides the possibility for autopilot. INAV includes a waypoint system. The waypoint system lets us, in accordance with the flight controller, physically map and trace out the course of action for the quadcopter drone. With mission planning in place, the drone will be able to take off, complete the experiment, and land safely without any and all external interactions.

The INAV program waypoint system is accessed and programmed by a non-open source android app named EZ GUI; this app will be discussed at a later time. INAV includes several other features, such as access to in flight adjustment.

4.4.1.5 EZ-GUI

EZ-GUI is an app that can be downloaded on android phones that connects with the quadcopter UAV. This app then records the flight pattern of the quadcopter and loads this data into log files. These log files are kept as either KML, CSV, WinGUI, or GPX format. EZ-GUI provides a log converter functionality that can convert the log files for a better graphical representation of the flight. File types such as KML can overlay important flight information, as shown in the figure below, and integrate it with a geographical representation of the flight area drafted by Google Earth.

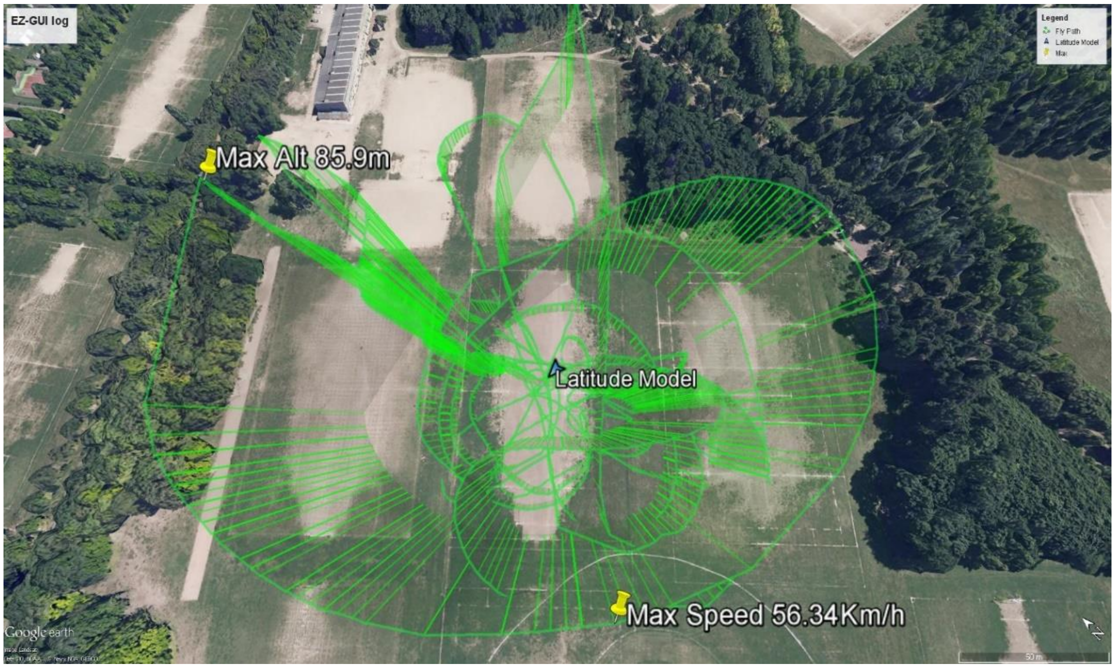


Figure 22: EZ-GUI KML converter flight path [16]

On the first run of the EZ-GUI app, the app will have to be configured to the metrics already established with the flight controller. For the purposes of this project, the EZ-GUI will be configured to allow the connection between the phone application and the quadcopter by USB port connection. The EZ-GUI will then be set to interface with the CleanFlight firmware that is already installed and configured to the quadcopter UAV. The app also has to be able to communicate with the drone and multiple satellites. The satellite connectivity is essential to drafting and following a desired flight path and record the data given from the flight path. Without a strong connection, it is impossible for the drone to construct the log files need for UAV mission planning.

Once connected, the drone can be programmed for previously set missions. Missions are the set path that the quadcopter is able to autonomously follow. These missions can be ported in from a previously recorded flight log or can be drafted in the moment using the EZ-GUI interface. Configuring the reduced gravity UAV to this mission controller gives the presence of a more precise flight pattern. With the press of a button, the drone will be able to transition from a hover over mode into a mission that incorporates the appropriate number of parabolic arches at the correct angle to achieve a reduced gravity flight time for experimentation with a far more reduced risk of human caused error.

After each successful run of the of the mission the drone can be automatically escorted back to the position set at home and safely land. The use of the EZ-GUI allows for repetitive and accurate flight will storing important flight path data that can be used for any necessary modifications or use in other measurements.

4.5 Flight Dynamics, System Identification, PID Tuning

Hardware configuration and test building require many trial and error runs in order to come to a reliable, functional, and working prototype. However, no amount of hardware or software alone will navigate a quadrotor unmanned aerial system through a parabolic flight to record reduced gravity. Rotors on a quadrotor are regulated by electronic speed controllers, which receive servo signals from the main flight control board. Brushless DC motors send a back-EMF feedback signal (position) so that speed controllers are aware of motor position and can “establish a switching strategy to control the rotor speed” [10].

Pitch, roll, yaw, and altitude degrees of freedom are controlled by motor speed and direction, with two motors spinning clockwise diagonal to each other, and two motors spinning counter clockwise spinning diagonal to each other as well.

In an open loop mode, the motor position feedback will be absent: thus the rotor’ speed is the output response, with no position or feedback loop to servo control.

Attitude indicator in traditional flight applications show the aircraft’s relative position to the earth’s horizon. It is a concept akin to the human ear, which helps a person balance and determine his or her orientation. Attitude indicators give live orientation information such as pitch and roll(bank). Attitude stability is important for quadrotor autonomous and manual flight modes when precision maneuvers, such as a reduced gravity inducing parabolic flight is required.

Attitude stabilization is one form of a control loop implementation that utilizes motor current feedback to minimize error due to disturbances. This allows steady hovering and stabilized flight envelopes. This control algorithm adds an “internal control loop on each driver in such a way that for any given control input the four motors turn at almost the same speed"[10]. This control algorithm approximates manual flight controller signals and motor DC signals, utilizing a PD (proportional and derivative) controller. It uses minimal hardware (a digital signal processor) and is based on a traditional quadrotor design: this control theory is under consideration due to it's robustness. At its core, a quadrotor has six degrees of freedom in space, which can be modeled in a multiple input, multiple output block diagram as shown in figure 12.

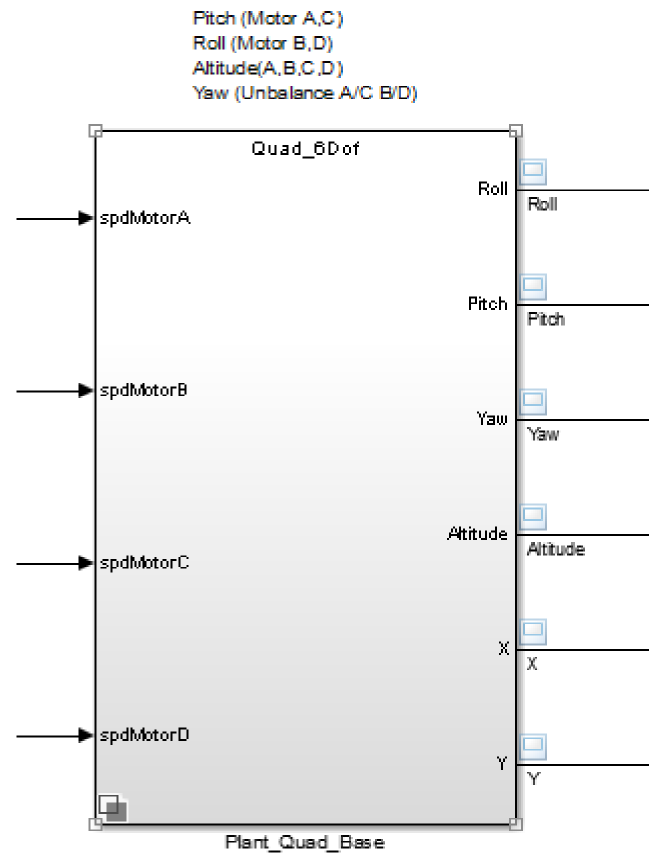


Figure 23: MIMO block

Through first-principles modeling, via Newton's laws of motion involving rigid body dynamics provides an accurate dynamics model of a quadrotor over all flight envelopes, it requires time consuming calculations and solving for many variables. In addition, modeling thrusts and forces acting on the quad rotor craft prove complicated and do not account for perturbations in a real environment. Instead of first-principles modeling, a secondary approach to flight dynamics modeling and design is system identification.

System identification uses real data from a model to provide information and data for flight dynamics models. One such approach is to use system identification models to implement robust control strategies to improve attitude stabilization and system responses to errors (perturbations due to wind gusts, etc.). A proven method of system identification is through analyzing the frequency response of the quadrotor functions and then comparing the differences between output gain and phase margins with input gain and phase margin.

One such tool to aid system identification is CIFER(Comprehensive Identification from FrEquency Responses) a US Army/NASA developed and supported system identification software used in real world full-scale helicopter applications. Our team corresponded via email with Kenny Cheung, Chief Software Architect, Technical Area Lead, Aero flight dynamics at NASA Ames research center regarding CIFER software applications. Mr. Cheung advised that CIFER may be used obtain identification of the aircraft, resulting in matrices blocks A,B,C, & D which may be input into Matlab Simulink block diagrams. From these matrices in the control design block diagram, a control system may be developed and tested (ran).

Additionally, a 3rd party flight simulation application called Flight Gear may be used (along with a 3-D imported model) for test flights within the simulation and control system tuning without running the risk of damaging our real prototype aircraft. More information on flight gear and Matlab Simulink may be found in the simulation section of this report. Mr. Cheung also advised that CIFER tool can be used to generate Bode plots of frequency response and calculate stability margins based on broken loop responses. Additionally, he stated that matrices M, F, G, & H can be derived along with optimized derivatives in the end-state system identification tool, used in conjunction with Matlab Simulink.

System identification may be used for PID tuning of the flight control board in order to illicit a desired response from the quadrotor design. An alternate control algorithm can be implemented that uses linear accelerometer readings from the flight control board as a feedback loop to the electronic speed controllers and servo outputs on the flight control board: this is a control loop based on reduced gravity trajectory and can be implemented in multiple ways. One way is via a body axis coordinate origin for the control system, and a possibly more effective acceleration component feedback system is based in the wind axis. Control algorithms and tuning based on both methods will be proposed, however the flight control configurators and GUI's may have restrictions for control system tuning. More information may be found in subsequent sections in the design portion of this report, under Firmware/software.

5.0 Prototype

In order to achieve success for this reduced gravity project, there is extensive prototyping necessary. Different prototypes span between the work done by the Electrical and Computer engineering, Mechanical and Aerospace, and Computer Science teams to test the different features of this project.

The mechanical and aerospace team will engage in a large amount of prototyping to ensure the stability, strength, flight, and practicality of the drone. They will prototype a quadcopter, which carries a payload, that will be used to conduct the research portion of this assignment. This drone will serve as the main base for the other parts of the project.

In order to test the design of the drafted drone, the prototype will need to be tested for the different aspects of flight mechanics. Mentioned briefly earlier, the prototype design will undergo extensive testing through virtualized simulation. The simulations granted by programs like Matlab will give access to cheaply adjust and manipulate the design and specifications of the initial drone design until a product is drafted that can handle the desired course of action provided by the experiment. Using Matlab Simulink coupled with the Quad Sim Add-on, a prototype can be tested and configured (via bench tests). A simple Arduino Uno can be used to bench test motor and propeller performance: output data is then analyzed by quad sim and used for inertial moment calculations.

A CIFER (Comprehensive Identification from Frequency Responses) model is an identification tool approach that is built to test and handle difficult problems in fight mechanics. Although physical drone will need to undergo time spent in a wind tunnel to measure for drag, our Mechanical and Aerospace team members have calculated the coefficient of drag to be 0.7-0.9 based on CAD models depicting upward motion of the quadrotor. Further testing is needed using Matlab Simulink, bench tests, and hardware configurations (Clean Flight/Beta Flight & iNav GUI). The end result of this testing will lead to the development of the final drone, capable of a 12.4 second flight (flight time required for reduced gravity maneuver). The prototype should have functionalities in place such as return to home and altitude hover hold.

The computer science team will develop an adequate system that will help produce, monitor, record, and store data on reduced gravity. The Computer Science team will also prototype the creation of an interface in which there can either pre-program the movement of the drone, or send impromptu controls through a signal. They will integrated a live video receiver transmitter system to record video of the internal payload chamber (as proof of concept for reduced gravity experiments). This prototype will also test the benefits and risks of using automated versus human based control.

As for the electrical and computer engineering team, prototyping will take shape in several different forms. The initial prototype will involve setting up the required components that can be tested while on the ground. This set up will include making sure that the flight controller can be programmed to control the direction and power of the motor.

Bidirectional thrust is an important feature that the drone will need to access. From the ground, certain sensors and GPS components can be view and tested for accuracy on the ground. This prototype is less about the use of these components once integrated into the final product, but rather the connectivity and compatibility of all of the parts.

Phase two of the electrical and computer engineering based components strives to integrate the technological sensors printed on a custom circuit board via Eagle CAD onto the proposed drone. This final product will mate the existing SP Racing Evo flight control board platform with the PCB black box. The custom PCB will be self-containing and not communicate with the flight control board, but will store data on an SD card. Though it would be preferable to fabricate our own flight control hardware and integrate with the platform created by the mechanical and aerospace team, this approach is not as plausibly due to the time constraints in this project. This prototype will be nearly exactly the same as the final product, minus the custom fabricated PCB black box attached to the platform.

A test quadcopter drone, similar to that used in the final project, has been acquired in order to serve as a medium for testing. This prototype drone will be used for test flights and allow our team members to hone their skills for manual flight maneuvers. The test quadcopter platform is an off the shelf DJI Phantom 3 standard. This quadcopter is durable and its battery allows up to 27 minutes of flight time. Additionally, exponential values (flight controller stick sensitivity) and gains may be adjusted so that our team members can form a basic understanding of dynamics behind the PID (proportional, integral, derivative) controllers used in a consumer grade quadcopter.

This test quadcopter drone will be used to test the thrust, autopilot controls such as return to home and altitude hover, as well as the sensor's readouts. An integral part of this prototype is finding the rough values, based on the equations and numbers provided by the mechanical and aerospace team, to produce a smooth flight on a desired test trajectory. Although the DJI Phantom 3 will not be able to reverse thrust midflight, it is still a useable platform for the interdisciplinary members to acclimate themselves to autonomous drone flying. The DJI GUI software is intuitive and shows a lot of features that will translate over into the Clean Flight environment when the SP Racing F4 is acquired. Learning on a similar hardware control and software interface will provide members with context and background information needed to build, test, and fly a custom made unmanned aerial system.

6.0 Consultants, Subcontractors, and Suppliers

This project is sponsored by Northrup Grumman. This project was designed with the thought in mind to advance the field of microgravity research. This section will outline our sponsor's donation and guidance, as well as contractors, support members, etc.

With the use of a quadcopter drone, as a drone is required by Northrup Grumman, achieving microgravity can be a reusable, easily accessible, and much cheaper affair than the current standard methods of achieving microgravity. Finding new ways for experimentation, even on a smaller scale like this, opens up the door for getting an upper edge and being the catalyst for development within the microgravity spectrum. Northrup Grumman provides aid in the specifications and requirements of the project. This also includes having the ability to contact them directly for advice dealing with the project at hand and ideas on how to handle certain problems we may come across. The financial support that stems from Northrup Grumman enables our team to purchase the items necessary to create our final project. In terms of financing, the budget is given to the group and budgeted to the different disciplinary teams: a full break-down of budget is given in the Admin section of this report.

The electrical and computer engineering team faces challenges in building a UAS in an interdisciplinary team, with Mechanical, Aerospace, and computer science teammates. To aid in in seamlessness, Eliot Ramey, faculty member at UCF and former Air Force pilot, has filled the role of our systems integration expert. Mr. Ramey has also been a flight instructor for over seven years, and brings relevant experience to this project. He also currently works as an air force modeling and simulation service support manager at Camber Corp.

OSH Park pcb services is a small batch community driven PCB fab house, based in and manufactured in the United States. They are easy to use, low cost, and offer a lot of options for PCB's. Our black box prototype will be an easy PCB design to print through OSH Park, being a prototype and of standard trace width, drill diameter, and annular ring size. However, for our contingency power distribution board, we may have to contact an alternate PCB fab house due to the large trace width and copper weight involved. Email correspondence has been made to OSH Park to determine if our distribution board can be fabricated with their manufacturing capabilities, or if alternate high current trace options can be implemented into either the Eagle .brd file, or the PCB characteristics and fabrication elements. An alternate PCB fab house for either board application is Saturn PCB design, inc., located in Deltona, FL. This fab house is ideal due to it's convenient location (within driving distance of Orlando, UCF campus). Shipping will be much quicker because of this close proximity. A third and more attractive option is Advanced Circuits, a US based company and third largest PCB manufacturer in the United States. Advanced circuits have a student discount and have decently fast fabrication and shipping times. Another advantage to using this manufacturer is the CAM engineer inspection of the board prior to printing: this process will iron out any board layout, sizing, or layer errors.

A major supplier of electrical components is Newark, a supplier that sells various integrated circuits, semiconductors, and components. We will choose Newark for major component selection such as IC's, resistors, capacitors, etc. They also carry the Step-down DC-DC voltage regulator used in the contingency power distribution board.

7.0 Milestones

The milestones in this project are based on the progress, coordination, and availability between the Electrical and Computer Engineering, Mechanical and Aerospace, and Computer Science teams. Due to the nature of this project and being interdisciplinary, certain aspects of the timing of this process is based heavily on the communication between the teams. Below is a table of the projected milestones achieved and desired for this reduced gravity drone project.

Milestones will be updated frequently with the accomplishment of group among completion of a desired task. Any item label as to be decided (TBD), will be replaced with an accurate timeframe once said information is known. The current milestones are set according to the rules and guidelines given to us by the Electrical and Computer Engineering department as well as the groups personal start and end times for each specified milestone.

Certain criteria of each milestone are led by a specific branch within our senior design team. Due to the inherit nature of an interdisciplinary project, not every branch within the team know the specific mechanics of the other group. With this in mind, not every aspect of the milestones and design is under the direct control of the Electrical and Computer engineering branch of the project.

Milestone	Start Date	End Date	Status	Lead
		Senior Design 1		
Initial Idea Selection	8/25/17	8/31/17	Complete	Group 32
Boot Camp and Assessing Team Dynamic	8/29/17	9/1/17	Complete	Group 32
Initial Project Document	8/31/17	9/22/17	Complete	Group 32
Interdisciplinary Group Selection	9/1/17	10/23/17	Complete	Group 32
Project Topic Research	9/22/17	11/3/17	Complete	Group 32
Component Research	9/22/17	11/18/17	Complete	EE/CpE
Drone Design	10/23/17	11/18/17	Complete	MAE
Circuit Design and Component Selection	11/3/17	11/18/17	Complete	EE/CpE
Draft Paper	8/31/17	11/18/17	Complete	Group 32
Final Paper	11/18/17	12/4/17	Complete	Group 32

		Senior Design 2		
Drone Production	TBD	TBD	-	MAE
PCB Production	TBD	TBD	-	EE/CpE
Circuit and Component Implementation	TBD	TBD	-	EE/CpE
Front end integration	TBD	TBD	-	CS
Mid Semester Demo	TBD	TBD	-	Group 32
Final Presentation	TBD	TBD	-	Group 32

Table 5: Project Milestones

8.0 Budget

The total cost of the quadcopter drone is to be approximately $1,500. This price is based on the financial support given by the Northrup Grumman sponsorship. The table below outlines the different components between all of the branches in order to have a single complete product.

Certain additional fees and costs such as additional batteries, flight simulation software and pre-design prototyping of $500 are taken into account for the calculation of the budget. Certain component costs are not yet known, resulting in an excess of $1500 indicated in the preliminary budget.

The wide range of cost for each component will be weighed and compared between similar pieces to access benefits versus the potential drawbacks. The final list of components will accumulate to a cost under the budget given by Northrup Grumman.

Item	Cost per Unit	Quantity	Total Cost
UAV registration	$5	1	$5
Pilot Fees	$0 - $100	1	$0 - $100
Motors	$80 - $125	4	$320 - $500
Propellers/ Blades	$20 - $130	2	$40 - $260
Batteries	$40 - $140	2	$80 - $280
Wiring	$10	-	$10
XT60 Connectors	$0.25 - $1	4	$1 - $4
Deans Connectors	$0.10 - $0.40	4	$0.40 - $1.60
Chassis	$200 - $300	1	$200 - $300
Propeller Arms	$50 - $150	1	$50 - $150
Supports	$20 - 60	1	$20 - $60
Base Stand	$0 - $10	1	$0 - $10
Experimental Chamber	$25 - $100	1	$25 - $100
LED light	$5	1	$5
Stabilization/ Dampener	$25 - $50	1	$25 - $50
FlySky Trans/Rec.	$68	1	$68
PCB	$20	1	$20 - $50

Bluetooth Telemetry	$60 - $80	1	$60 - $80
Video Receiver/ Transmitter	$40 - $80	1	$40 - $80
CS Team	$0 - $200	1	$0 - $200
MatLab Software for Simulation	$120	1	$120
SP Racing F4 Evo Flight Controller	$37	1	$37
Lumenier 6s Power Distribution Board	$13	1	$13
Lumenier BLHeli Speed Controller	$112	1	$112
GPS module	$26	1	$26
Micro-SD card	$35	2	$70
Grand Total:			**$1347.40 - $2691.60**

Table 6: Preliminary Budget for Quadcopter Drone

References

[1] Afman, J.-P., Franklin, J., Mote, M. L., Gurriet, T., & Feron, E. (2016). *On the Design and Optimization of an Autonomous Microgravity.*

[2] *High Performance 6-Axis MEMS MotionTracking Device.* (2016, 10 03). Retrieved from IvenSense.

[3*] Energizer Product Safety Datasheet: Lithium Polymer)* (2015). Retrieved from Energizer

[4] *MIL-HDBK-217F(N2) Parts Count Prediction Calculator* (2017)

http://reliabilityanalyticstoolkit.appspot.com/mil_hdbk_217F_parts_count

[5] Daniel Hsing Po Kang, Mengjun Chen‡†, and Oladele A. Ogunseitan†

.(2013). *Potential Environmental and Human Health Impacts of Rechargeable*

Lithium Batteries in Electronic Waste

[6] Coulston, Chris, Ford, Ralph. (2007). *Design for Electrical and Computer Engineers*

Published by McGraw-Hill Education

[7] *Flight Weather Monitoring .* (n.d.). Retrieved from Flight Service.

[8] *Temorary Flight Restrictions .* (n.d.). Retrieved from Federal Aviation Administration: http://tfr.faa.gov/tfr2/list.html

[9] Intertek. (n.d.). *Navigating the Regulatory Maze of Lithium Battery Safety.* See Appendix C for copright letter

[10] Sanchez, Anand; Garcia-Carillo, Luis; Rondon Eduardo; Garcia, Octavio. Edited by Rogelio Lozano. *Umanned Aerial Vehicles, Embedded Control: Improving Attitude Stabilization of a Quadcopter Using Motor Current Feedback* (2010).Wiley. Hoboken, New Jersey.

[11] James. (2016, March 16). Cleanflight setup guide & checklist. Retrieved from Propwashed: https://www.propwashed.com/flight-controller-software-set-up-checklist/

[12] Drexel, David. “Quadcopter Dynamic Modeling and Simulation.” *GitHub*, 14 July 2014, github.com/dch33/Quad-Sim.

[13] Ltd, Black Stick. “The PCB Design Professionals.” *PCB Design from Black Stick Ltd*, 2012, www.blackstick.co.uk/pcb-design-calculators/track-width-current-mm.php.

[14] James. “Cleanflight Setup Guide & Checklist.” *Propwashed*, 26 Apr. 2017, www.propwashed.com/flight-controller-software-set-up-checklist/.

[15] James. “Betaflight Configuration Guide.” *Propwashed*, 30 May 2017, www.propwashed.com/betaflight-configuration-guide/.

[16] “What Is It and How to Use It.” *EZ-GUI — Ground Station for Cleanflight/Betaflight/INav and Multiwii*, ez-gui.com/ez-gui-logs-converter-2/ez-gui-logs-converter/.

Appendix A: Educational use of Unmanned Aerial Systems Memorandum

Federal Aviation Administration

Memorandum

Date: May 4, 2016

Earl Lawrence, Director, Unmanned Aircraft Systems Integration Office,

John Duncan, Director, Flight Standards Service, AFS-I

From: Reginald C. Govan, Chief Counsel, AGC-I

Prepared by: Dean E. Griffith, Attorney, AGC-220

Subject: Educational Use of Unmanned Aircraft Systems (UAS)

This interpretation addresses: (1) use of unmanned aircraft for hobby or recreational purposes at educational institutions and community-sponsored events; and (2) student use of unmanned aircraft in furtherance of receiving instruction at accredited educational institutions.

There is uncertainty in the model aircraft community about when an unmanned aircraft is a model aircraft operated for hobby or recreation or is an operation requiring FAA authorization. The FAA has received many inquiries from students and educational institutions offering coursework in the design, construction and operation of small unmanned aircraft with respect to the types of activities in which students and faculty lawfully may engage pursuant to the existing legal framework.

In light of these questions, we are issuing this interpretation to clarify that:

- A person may operate an unmanned aircraft for hobby or recreation in accordance with section 336 of the FAA Modernization and Reform Act of 2012 (FMRA) [1] at educational institutions and community-sponsored events provided that person is (1) not compensated, or (2) any compensation received is neither directly nor incidentally related to that person's operation of the aircraft at such events;

- A student may conduct model aircraft operations in accordance with section 336 of the FMRA in furtherance of his or her aviation-related education at an accredited educational institution.

Pub. L. 112-95,§ 336(a)(1

2 Community-sponsored events would include demonstrations at schools, Boy or Girl Scout meetings, science clubs, etc.

- Faculty teaching aviation-related courses at accredited educational institutions may assist students who are operating a model aircraft under section 336 and in connection with a course that requires such operations, provided the student maintains operational control of the model aircraft such that the faculty member's manipulation of the model aircraft's controls is incidental and secondary to the student's (e.g., the faculty member steps-in to regain control in the event the student begins to lose control, to terminate the flight, etc.).

On June 25, 2014, the FAA published in the Federal Register its interpretation of the Special Rule for Model Aircraft, section 336 ofthe FMRA. 79 Fed. Reg. 36172 (June 25, 2014). [1]Currently, the FAA is reviewing the more than 33,500 comments to that Special Rule. In addition, on February 23, 2015, the FAA published in the Federal Register its Notice of Proposed Rulemaking (NPRM) on the Operation and Certification of Small Unmanned Aircraft Systems (UAS). 80 Fed. Reg. 9544.

Separate from these two actions, the FAA continues to receive a number of questions on the use of model aircraft to conduct demonstrations and on student use of model aircraft in connection with participation in coursework at educational institutions. The FAA finds it necessary to clarify the applicability of section 336 of the FMRA and of the FAA's operating requirements for UAS. The FAA recognizes that UAS increasingly are being used in education, including science, technology, education, and math (STEM) education, which is the focus of President Obama's Educate to Innovate Initiative.[2]

Hobbyist Use of UAS to Conduct Demonstrations

Section 336(a) of the FMRA provides special rules for model aircraft that require the aircraft to be:

1) Flown strictly for hobby or recreational use;

2) Operated in accordance with a community-based set of safety guidelines and within the programming of a nationwide community-based organization;

---

[1] A model aircraft means an unmanned aircraft that is: (1) capable of sustained flight in the atmosphere; (2) flown within visual line of sight of the person operating the aircraft; and (3) flown for hobby or recreational purposes. Pub. L. 112-95, s 336(c).

[2] http://www.whitehouse.gov/issues/education/k- 12/educate-innovate

3) Limited to not more than 55 pounds unless otherwise certified through a design, construction, inspection, flight test, and operational safety program administered by a community-based organization;

4) Operated in a manner that does not interfere with and gives way to any manned aircraft; and

5) When flown within 5 miles of an airport, the operator of the aircraft provides the airport operator and the airport air traffic control tower (when an air traffic confrol facility is located at the airport) with prior notice of the operations (model aircraft operators flying from a permanent location within 5 miles of an airport should establish a mutuallyagreed upon operating procedure with the airport operator and the airport air traffic control tower).

If an unmanned aircraft is operated as a model aircraft in accordance with the above, then it is does not require FAA authorization. A key element in determining whether an operation may qualify as a model aircraft operation is that it must be flown for "hobby or recreational" purposes. The FAA's interpretation of section 336 relies on common definitions of the terms "hobby" and "recreational." Previous agency guidance addressed the parameters of hobby or recreational use:

> Any operation not conducted strictly for hobby or recreation purposes could not be operated under the special rule for model aircraft. Clearly, commercial operations would not be hobby or recreation flights. Likewise, flights that are in furtherance of a business, or incidental to a person's business, would not be a hobby or recreation flight. [6]

The FAA interprets "hobby or recreational" use to include operation of UAS to conduct demonstrations at accredited educational institutions or at other community-sponsored events provided the aircraft is not being operated for compensation, in furtherance of a business or incidental to a business. Therefore, a model aircraft hobbyist or enthusiast lawfully may fly UAS at accredited educational institutions or other community-sponsored events to promote the safe use of UAS and encourage student interest in aviation as a hobby or for recreational purposes provided the hobbyist receives no compensation of any form (including honorarium or reimbursement of costs), or any such compensation neither directly nor indirectly furthers the hobbyist's business or operation of the UAS and he or she follows the provisions of section 336.

Student Operation of Model Aircraft for Educational Purposes

If not operated as "model aircraft" under section 336 of the FMRA, currently there are three ways to lawfully conduct unmanned aircraft operations in the United States: (1) as public aircraft operations pursuant to the requirements of the public aircraft statute and under a Certificate of Waiver or Authorization (COA) from the FAA; (2) as limited commercial operations by type certificated UAS, provided the operator obtains a COA from the FAA; or (3) pursuant to a Section 333 of the FMRA grant of exemption based on the Secretary of Transportation's determination that a certificate of airworthiness is not required, and provided the operator obtains a COA from the FAA. [7]

Each of these three methods is available to educational institutions (including their faculty and students) that want to operate UAS, including for commercial, research and development, and any other non-hobby or non-recreational purpose. Each, however, requires the educational institution or its faculty and students to meet statutory prerequisites and obtain from the FAA the requisite approvals (in the form of exemptions and COAs).

Many educational institutions are keenly interested in having students operate unmanned aircraft as model aircraft under section 336 of the FMRA in connection with their academic coursework at those schools. The educational community contends that these operations not only meet the definition of model aircraft but also meet the unique need of students, which is learning how to design, construct and operate small unmanned aircraft as a component of a variety of science, technology and aviation-related educational curricula. Students also are interested in operating small unmanned aircraft for other educational purposes such as in connection with television, film, or photography courses.

The FAA has considered whether a student's course work of learning how to operate and use a UAS constitutes a hobby or recreational activity within the meaning of section 336' s definition of model aircraft. The FAA believes students operating UAS as one component of a curricula pertaining to principles of flight, aerodynamics and airplane design and consfruction promotes UAS safe use and advances UAS-related knowledge, understanding and skills. UAS also may provide students a useful tool in other academic curricula such as television, film production or the arts generally. Although it may be argued that the student's knowledge and skills obtained through such coursework are necessary for a diploma or degree, which subsequently can lead to an aviation-related job or increased earning potential, the FAA finds this link simply too attenuated to transform student UAS use, as a component of an accredited educational curriculum, into a non-hobby or non-recreational use within the meaning of section 336. A person that operates a UAS strictly for hobby or recreation learns about principles of flight, aerodynamics, and airplane construction may subsequently use such knowledge when gainfully employed, but that does not transform what is otherwise a hobby or recreational activity into a non-hobby or non-recreational pursuit.

Therefore, we find that the use of small unmanned aircraft by students at accredited educational institutions as a component of science, technology and aviation-related educational curricula or other coursework such as television and film production or the arts more closely reflects and embodies the purposes of "hobby or recreational" use of model aircraft and is consistent with the intent of section 336 of the FMRA. Accordingly, the FAA concludes that student use of UAS at accredited educational institutions as a component of their science, technology and aviationrelated educational curricula, or other coursework such as television and film production or the arts, is "hobby or recreational use" within the meaning of the FMRA. The student is,

P.L. 112-95,

[6] 79 Fed. Reg. at 36174

[7] On February 23, 2015 the FAA proposed a rule that, when finalized, will provide a framework for small UAS operations. See Operation and Certification of Small Unmanned Aircraft Systems Notice of Proposed Rulemaking, 80 Fed. Reg. 9544.

however, responsible for meeting and complying with all other elements required for lawful model aircraft operations pursuant to Section 336 of the FMRA, including the student not receiving any form of compensation (including reimbursement of costs, honorarium, etc.) directly or incidentally to his or her operation of the model aircraft.

Faculty Use of Model Aircraft

The FAA recognizes that faculty participation in the student's learning experience often is an integral component of the student's educational experience and that faculty should be able to participate in and contribute to the unmanned aircraft activities in which students can engage as hobbyists. However, a faculty member engaging in the operation of an unmanned aircraft, as part of professional duties for which he or she is paid, would not be engaging in a hobby or recreational activity. Rather, the faculty member is being compensated for his or her teaching or research activity, including any UAS operation arising from or related to the faculty member' s teaching a course or conducting research.

Likewise, a student operating UAS for research on behalf of a faculty member is associated with the faculty member's professional duties and compensation and, thus, is not hobby or recreational use by the student pursuant to section 336. Student operation of UAS for the professional research objectives of faculty renders the operation non-hobby or non-recreational. Accordingly, a faculty member conducting research may not rely on section 336's concept of 'hobby or recreational use" ‘ to either operate a UAS or direct student UAS operations in connection with such research.

Nevertheless, faculty teaching a course or curricula that uses unmanned aircraft as a component of that course may provide limited assistance to students operating unmanned aircraft as part of that course without changing the character of the student's operation as a hobby or recreational activity or requiring FAA authorization for the faculty member to operate. The FAA finds that de minimis limited instructor participation in student operation of UAS as part of coursework does not rise to the level of faculty conducting an operation outside of the hobby or recreation construct.

This limited circumstance would apply to courses at accredited institutions where the operation of the unmanned aircraft is secondary to the design and construction of the aircraft, such that the primary purpose of the course is not operating an unmanned aircraft. For example, an instructor teaching an engineering course in which construction and operation of UAS are one part of the curriculum would be able to conduct limited UAS operations. In that case students would fly UAS to test the validity of design or construction methods to show mastery of the principles of the course. The faculty member's UAS operation would be secondary to the purpose of instructing engineering courses. In contrast, this limited circumstance would not apply to a course related to UAS flight instruction. In that case, the student's primary purpose for taking the course is to learn to fly a UAS and flight would be expected to be demonstrated on a regular

---

Small unmanned aircraft are those weighing less than 55 pounds. See FMRA 331(6).

[9] The prohibition on receiving compensation, while broad, does not preclude a student from operating UAS in connection with fulfilling a specific course's requirement while also receiving financial aid, participating in workstudy programs or being a paid research assistant to a faculty member teaching such course.

basis. In that case, the faculty member's UAS operation is closely tied to his or her purpose of instructing how to fly a UAS.

Conclusion

UAS may be used to conduct demonstrations at schools or other community-sponsored events provided the person operating the aircraft is (1) not compensated, or (2) any compensation received is neither directly nor incidentally related to that person's operation of the aircraft at such events.

Students that operate model aircraft in connection with fulfilling an accredited educational institution's curricula lawfully may conduct model aircraft operations for hobby and recreational purposes pursuant to section 336 of the FMRA, provided they do not receive compensation, directly or incidentally, arising from or related to such operations. Faculty at these educational institutions teaching such curricula may assist students with their model aircraft operations under section 336, provided that the operations are used to teach such curricula to students enrolled in those courses and the faculty member's participation is limited to de minimis participation in the student's UAS operations. We emphasize that these operations must be conducted under the provisions of section 336.

The FAA emphasizes that faculty members who wish to operate UAS outside of these parameters must seek authorization though one of the three methods discussed above. We also note that this interpretation was drafted prior to issuance of the final rule for Operation and Certification of Small UAS Rule and this interpretation may need to be revisited depending on its provisions. See 80 Fed. Reg. 9544 (Feb. 23, 2015) (Notice of Proposed Rulemaking).

Please contact my office with any questions about this memorandum.

Appendix B: SP Racing Evo F4 Flight Control Board Data Sheet

Seriously Pro

Racing F4 EVO Flight Controller

Thank you for directly supporting the Cleanflight project with your purchase.

Seriously Pro Racing F4 EVO Flight Controller Manual (Version 2)

Copyright © 2017 Dominic Clifton

About

The Seriously Pro Racing F4 EVO Flight Controller (SPRacingF4EVO) was designed to give awesome flight performance in a stackable race-ready package. It has a high-performance CPU, the latest sensors, race timing & logging technology backed by excellent connectivity options at a wallet-friendly price.

Featuring a race timing transponder system the SPRacingF4EVO is truly designed for racers. Analyze your race and flight telemetry/blackbox logs using the built-in MicroSD card socket.

The SPRacingF4EVO gives you all the features you need for the heart of your aircraft, whether you're into FPV racing, acrobatic flying or aerial photography it's perfect.

Features

- Next-generation STM32 F4 processor with hardware floating point unit for efficient flight calculations and faster ARM-Cortex M4 core.
- MicroSD-Card socket for black box flight log recorder - optimize your tuning and see the results of your setup without guesswork.
- Race transponder built in - just turn up at a race and have your lap times recorded.
- Features the latest Accelerometer, Gyro and Baro/Altitude sensor technology connected via the fast SPI bus.
- Wire up using pin headers for all major connections for excellent crash-durability. Use either right-angled or straight pin-headers.
- No compromise I/O. Use all the features all the time; e.g. Connect your USB + OSD + SmartPort + SBus + LED Strip + Battery Monitoring + 8 motors + Transponder LED - all at the same time!
- 8 DSHOT/OneShot/PWM output lines for ESCs and Servos. Arranged for easy wiring on standard pin headers.
- Supports direct connection of SBus, SumH, SumD, Spektrum1024/2048, XBus receivers. No external inverters required (built-in). PWM 1 wire per channel receivers not supported.
- Supports direct connection of 3.3v Spektrum Satellite receivers via 3 pin through-hole JST-ZH connector.
- 5 Serial Ports - NOT shared with the USB socket.
- Micro USB socket.
- Dedicated output for programmable LEDs - great for orientation, racing and night flying.
- Dedicated I2C port for connection of OLED display without needing flight battery.
- Battery monitoring for voltage and current.
- RSSI monitoring (analog or PWM).
- Buzzer port for audible warnings and notifications.
- Developer friendly debugging port (SWD) and boot mode selection, unbrickable bootloader.
- Boot button for easy firmware updating.
- Symmetrical design for a super tidy wiring.
- JST-SH sockets for I2C, UART4 and SWD.
- Flashing via USB or serial port.
- Stackable design - perfect for integrating with OSDs and power distribution boards.
- Standard 30.5mm mounting holes.
- LEDs for 3v, 5v and Status for easy diagnostics.
- Cleanflight logo.

Software

The SPRacingF4EVO runs the open-source Cleanflight flight control (FC) software which has an ever-growing community of friendly developers and users. Being open-source means that you too can contribute to the system.

Cleanflight comes with a detailed manual that is reviewed and maintained by the Cleanflight developers and community. No more out-of-date wiki pages and second-hand information.

See http://cleanflight.com for links to the manual. PDF copies can be downloaded from the github releases pages. Ensure you reference the manual that is appropriate to your firmware version.

History

The hardware was designed by the lead developer of Cleanflight to be more capable than the previous-generation STM32F3-based boards and to set the benchmark for a wallet-friendly STM32F4 based board with 8 DSHOT outputs and simultaneous Transponder, LED Strip and Telemetry support.

The F4EVO uses a similar layout to the SP Racing F3 EVO boards; the stack-pins, ESC/Servo outputs and connectors are in the same location on the F4 EVO as they are on the F3 EVO, F3 Acro and Deluxe boards for maximum mounting compatibility with existing products such as the SPRacingF3OSD/PDB board.

Compared to the F3 EVO and F3 Acro/Deluxe the new F4EVO uses newer CPU, newer sensor technologies, adds MicroSD and Transponder functionality and drops PWM receiver (1 wire per channel) support.

WARNINGS

Failure to adhere to these warnings will void your warranty and destroy your flight controller.

- When using ESCs that support active braking or battery regeneration ensure the FC is protected with a power supply filter. Damage to the FC caused by lack of protection is not warrantable.
- Observe polarity at ALL TIMES. Check and DOUBLE CHECK before applying power. Do not rely on wire color-coding alone.
- POWER OFF before unplugging, plugging in or making any connections.
- Connect only one 5.0v SOURCE of power to the VIN pins / Do not connect more than one SOURCE of power to two or more of the VIN pins. e.g. If you are using ESCs with BECs then remove the center RED wire from all but one ESC connector.

- Do not connect SOURCES of power to the pins marked with 5v. They are OUTPUTS for supplying power to other devices.
- Do not connect a LiPo to the 5.0v VIN pins.
- The 3.3v supply is for low-current use only. 100mA MAX.
- Do not use a VIN voltage source higher than 5.0v. Check your 5v supply is actually outputting 5.0 volts, check it before powering the FC via VIN. - Do not just assume it is 5v.
- Do not connect GND, 5v, VIN or 3.3v to each other (short circuit).
- Do not connect GND, 5v, VIN or 3.3v to any inputs or outputs unless specifically stated.
- Do not connect any input or output to any other input or output unless specifically stated.
- Do not allow dirt/dust/glue/etc into the pressure sensor (barometer).
- Keep magnets away from the flight controller.
- Do not use excessive force when inserting or removing MicroSD cards.
- Always align USB plug and socket when inserting/removing USB cables to prevent damage to the USB socket.

GENERAL ADVICE

Follow the advice below for best performance and long-life of your flight controller:

- Apply resin/glue to reinforce JST-SH connectors - helps if you crash your aircraft.
- To further protect the board from crashes you can add a some additional solder to the edges of the JST-SH and USB sockets to reinforce them.
- Support JST connector sockets when inserting cables.
- Using an enclosure/box for the flight-controller is recommended.
- Install open-cell foam on the pressure sensor.
- Do not cover the hole in sensor (e.g. with glue, resin, etc) or allow foreign object to enter it.
- Route motor/battery wires as far away from the compass sensor (magnetometer) as possible.
- Using color-coded pin headers is recommended (not supplied), especially for BATTERY connections.
- For optimum performance do everything you can to prevent vibrations reaching the accelerometer/gyro sensor. e.g. balance motors, props use rubber isolation grommets and secure everything. If twitching is observed check motor bearings and other sources vibrations.

UART3

Use for GPS, Spektrum Satellite RX, SmartPort Telemetry, HoTT telemetry, etc.

IMPORTANT: 3.3v ONLY signals.

UART1 (4)	
1 ■	RXD (R1)
2 ●	TXD (T1)

UART1

Use for connecting to OSD/GPS/BlueTooth. 5v signals OK.

Spektrum Satellite socket holes.

Holes ready to install a top-mount or side-mount through hole JST-ZH 3 pin socket for connecting to a Spektrum Satellite receiver. PIN 1 is nearest the RX/TLM pads, PIN3 is nearest the mouning hole at the top-left. IMPORTANT SEE S/R BRIDGE INSTRUCTIONS.

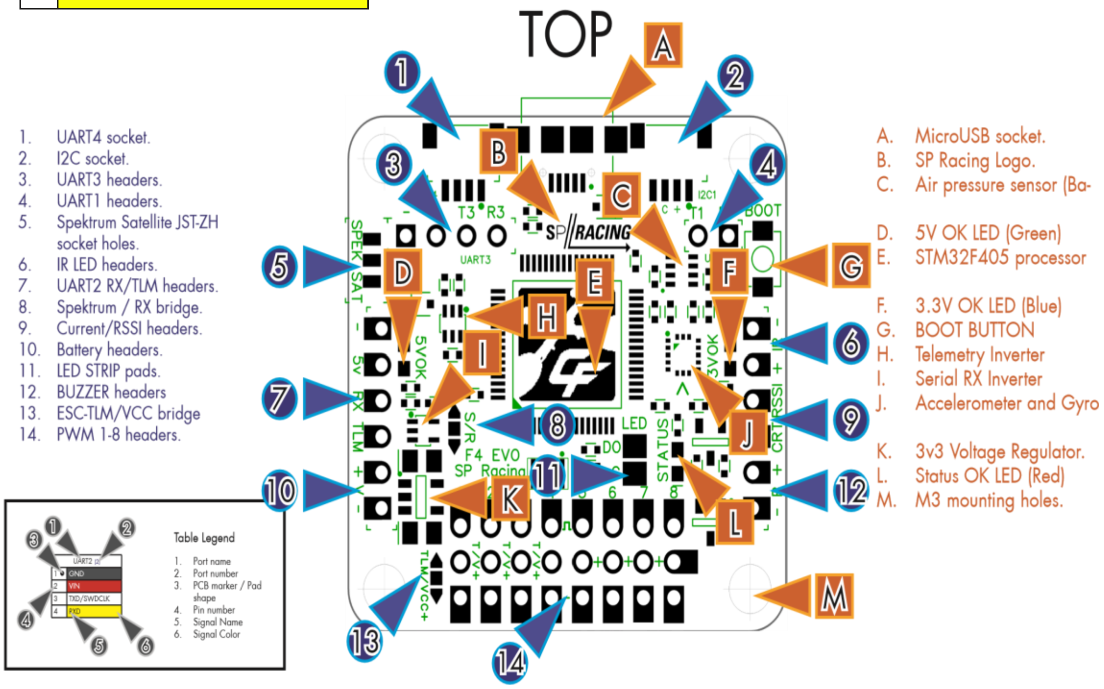
Spektrum Satellite (5)	
1 ●	RXD (R3)
2 ●	GND
3 ●	3.3v

UART3 (3)	
1 ■	GND
2 ●	VIN
3 ●	TXD (T3)
4 ●	RXD (R3)

IR LED

IR + (Pin 2, ROUND) is used to a single IR LED - either direct-solder the LED or attach the LED via a cable. The LONG leg of the LED goes in the ROUND hole. The SHORT leg of the LED goes in the SQUARE hole.

IR LED (6)	
1 ■	IR -
2 ●	IR + / LED Strip



LED STRIP (11)	
1 ■	Data Out (DO)
2 ■	Ground (G)

LED STRIP

Connect the DO pad to the DIN (Data DIN) on a WS2812 RGB LED strip. Connect the G pad to the GND pin on a WS2812 LED Strip. Supply power to the WS2812 LED strip via an external voltage regulator.

RX/TLM (7)	
1 ■	GND
2 ●	5v

3 ●	RX - U2/RXD -- Serial RX/PPM
4 ●	TLM - U2/TXD or U5/S.PORT/HoTT

UART2/PPM - Serial RX or PPM RX + Telememetry See the receiver connection guide below.

Pin 4 is EITHER connected to UART2 TXD or to a bi-directional inverter on UART5.

Use UART2 for telemetry when the T2/TLM bridge is set to T2.

Use UART5 for telemetry when the T2/TLM bridge is set to TLM.

CURRENT / RSSI (9)	
1 ●	CURRENT (CRT)
2 ●	RSSI

CURRENT / RSSI

Current - 0 - 3.3v input from external current sensor for current monitoring.

RSSI is for 0 - 5v PWM RSSI or 0 - 3.3v Analog RSSI - Disabled until RSSI select pads (bottom) are set.

BATTERY (10)	
1 ●	+V
2 ■	V-

BATTERY - IMPORTANT: DOUBLE CHECK and TRIPLE CHECK POLARITY!

For connecting a 2-6S LiPo battery (25V MAX). Used for voltage monitoring ONLY. Will NOT power the board or accessories.

PWM 1-8 ESC/SERVO OUTPUTS (14)	
1 ●	PWM/ONESHOT125
2 ●	Pads 1-4 are Switchable, 5-8 always VIN
3 ■	GND

ESC/SERVO OUTPUTS.

8x3 Pin headers for connecting servos/motors. ALL VIN holes are connected to each other. VIN can be used to power the board. When VIN-5v Pads are bridged only connect OUTPUTs to VIN holes.

IMPORTANT: The first 4 round CENTER pads (on the left) are NOT CONNECTED by default. Set the TLM/ VCC bridge ONLY if you need to use them.

If you have ESCs that have TELEMETRY OUT then set the TLM/VCC bridge to TLM and then connect each ESC's TELEMETRY OUT signal to the first four round center pads. Use UART4 for ESC TELEMETRY.

BUZZER (12)	
1 ■	BUZZER- (B-)
2 ●	BUZZER+ (B+) / 5v

BUZZER

Use 5V and BUZZER- to connect to an external buzzer. 5.0v is also supplied when powering via USB.

UART4 socket (1)	
1 ●	GND
2	5v
3	TXD/SWDCLK
4	RXD/ESC-TELEMETRY

UART4 connector - Used for 5.0v Serial IO. (GPS, etc.) TXD MUST NOT be used when SWD port is in use.

5.0v is always supplied via the on-board voltage regulators, even when powering via USB.

RXD signal also available on the middle of the first 4 motor outputs when the ESC-TLM bridge is set to TLM.

I2C1 socket(2)	
1 ●	GND
2	5v
3	SCL
4	SDA

I2C1 connector - Used for external sensors and OLED displays. The SCL and SDA are 3.3v signals.

5.0v is also supplied when powering via USB.

IMPORTANT: logic level converters are REQUIRED if your sensors use 5.0v signals.

TLM/VCC (13)	
1▲	ESC-TELEMETRY (UART4 RX)
2 ■	OUTPUT
3▼	VIN

ESC-TELEMETRY/VCC select bridge pads

Create a solder bridge between TWO PADS ONLY to select the function of the OUTPUT pad. a) bridge ▲ and ■ to set the first 4 CENTER ESC pads to ESC-TELEMETRY (UART4) b) bridge ■ and ▼ to set the first 4 CENTER ESC pads to VCC/VIN.

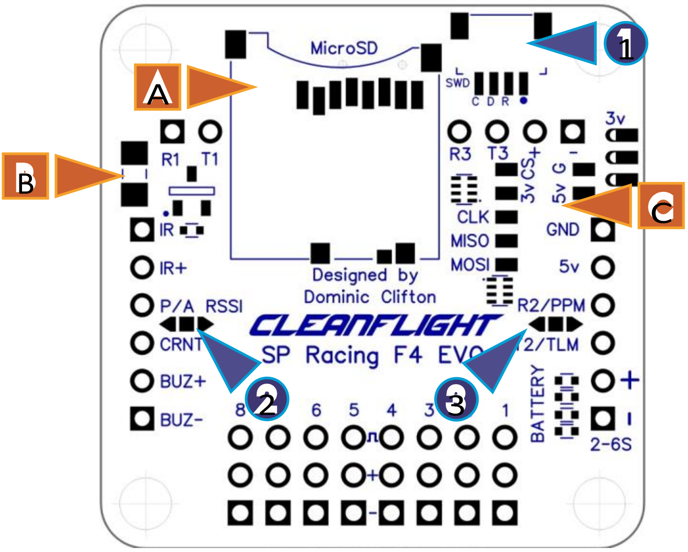
See 'Receiver Connections' section below.

NOTE:

You can check the transponder LED is working by using a mobile phone camera pointed straight at the LED when the transponder is enabled via the board and in the software. The LED will pulse an infrared signal which usually shows up purple via a mobile phone camera.

BOTTOM

1. SWD debugging socket.
2. RSSI Analog/PWM bridge.
3. T2/TLM bridge.



A. MicroSD card socket
B. IR Transponder circuit.
C. SPI3/VTX pads.

SWD/DEBUG (1)	
1 ●	GND
2 R	NRST
3 D	SWDIO
4 C	SWDCLK

SWD/DEBUG connector - Used for software development or flashing via SWD

Cannot be used when UART2 is enabled. Use an ST-Link debugger with OpenOCD or a J-Link debugger.

If for some reason you cannot flash using USB DFU or UART1 you can reinstall firmware using this port.

T2/TLM (3)	
1◀	T2
2■	OUTPUT
3▶	TLM

T2/TLM select bridge pads

Create a solder bridge between TWO PADS ONLY to select the function of the OUTPUT pad. a) bridge ◀ and ■ to set the TLM pin to T2. b) bridge ■ and ▶ to set the TLM pin to TELEMETRY. See 'Receiver Connections' section below.

RSSI PWM / ANALOG SELECT (3)	
1◀	PWM RSSI
2■	INPUT
3▶	ANAGLOG RSSI

RSSI PWM / ANALOG select bridge pads

Create a solder bridge between TWO PADS ONLY to select the function of the INPUT pad. a) bridge ◀ and ■ to use the "RSSI" pad for PWM RSSI - for 0 - 5v PWM signals.
b) bridge ■ and ▶ to use the "RSSI" pad for ANALOG RSSI - for 0 - 3.3v Analog Signals.

NOTE:

When bridging select pads with solder, put a small blob of solder on two pads, let the solder cool, then bridge them together with a little more solder.

IMPORTANT: DO NOT CONNECT ALL THREE PADS TOGETHER.

Receiver Connections

The F4 EVO has two receiver sockets (RX/TLM and Spektrum Sat) and two receiver configuration bridges (S/R and T2/TLM).

Examples of PPM receivers: FrSky D4R, FrSky D8R

Examples of Serial RX receivers: FrSKY X4RSB, XS4 (SBus), Spektrum AR6270T (Spektrum), Graupner GR24 (HoTT), Futaba R3008SB
(SBus)

Examples of Spektrum Satellite receivers:Spektrum Quad Race Serial Receiver w/Diversity - SPM4648, DSMX Remote Receiver - SPM9645

The following table indicates where different types of receivers should be connected and what the bridges should be set to.

RX	Telemetry	T2/TLM Bridge	S/R Bridge	RX Port Pin 3 TLM/R2	RX Port Pin 4 TLM/T2	Spektrum Sat Port	RX	TELEMETRY
PPM	FrSky/LTM/etc	T2	R *1	RX Out	Telemetry Out	UNUSED	PPM	UART2
FrSky Serial RX	S.PORT	TLM	R *1	RX Out	S.PORT	UNUSED	UART2	UART5
Graupner HoTT RX	HoTT	-	R *1	RX Out	UNUSED	UNUSED	UART2	UART1/3/4 *3
Single-Wire RX+TLM	IBUS/SRXL/CRSF	T2	R *1	UNUSED	RX out/TLM in *4	UNUSED	UART2	UART2 *4
Dual-Wire RX+TLM	IBUS/SRXL/CRSF	T2	R *1	RX Out *5	TLM in *5	UNUSED	UART2	UART2 *5
Spektrum Satellite RX	-	-	S *2	UNUSED	UNUSED	RX OUT	UART3	-

*1 Default - no soldering needed.

*2 Cut trace between SQUARE and ‘R’ TRIANGLE then solder ‘S’ TRIANGLE to SQUARE - See below.

*3 UART5, when used, is ALWAYS INVERTED. S.PORT is inverted. HoTT is NOT and cannot be used on the same UART port as the receiver signal. UART2 is used for receiver signals and UART5 cannot be used for HoTT so use any other free serial port for HoTT telemetry - i.e UART1,3 or 4.

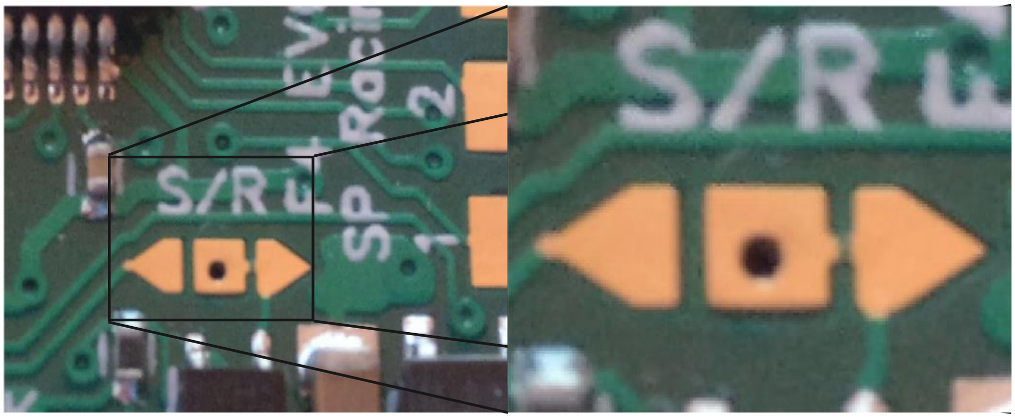
*4 Some receivers support one-wire bi-directional transfer of RX channels to the FC and Telemtery from the FC to the Receiver. e.g. IBUS/SRXL/CRSF. If you receiver supports this use the UART2 TX pin only.

*5 Some receivers support incoming RX data and outgoing Telemetry at the same speed and port settings, using two wires.

S/R bridge.

The S/R bridge is connected by default to the ‘R’ side. ONLY SPEKTRUM SATELLITE RECEIVER USERS should change or set this jumper. There is a small electrical trace between the SQUARE pad and the right-hand side ‘R’ TRIANGLE pad. This trace MUST be cut, carefully and without slipping, and the left-hand side ‘S’ TRIANGLE must be soldered to the center SQUARE pad before connecting your Spektrum Satellite receiver to the Spektrum

Satellite receiver port. After cutting the trace, use a multimeter to check there is no connectivity between the center SQUARE pad and the right-hand ‘R’ TRIANGLE pad.



When set to S the S/R bridge isolates the inverter from the CPU and enables Spektrum Bind support on UART2. When set to R the S/R bridge connects the inverter to the CPU and allows the CPU to enable/disable the inverter as required by the receiver.

Soldering

VERY IMPORTANT! - Do NOT solder your flight controller until you have plugged in a USB cable and checked that the GREEN, BLUE and RED lights operate! Red will flash, GREEN and BLUE must be ALWAYS ON.

IMPORTANT:

- Use a high quality soldering iron and good solder.
- Tin/Lead solder is MUCH easier to use than other Lead-free solder.
- Use solder with flux and remove any flux residue after soldering. AVOID CORROSIVE FLUX!
- Check for and remove solder balls you may have created after soldering - use a magnifying glass.
- Ensure you have sufficient fume extraction when soldering.
- Pin headers that connect to the GND signal will be more difficult to solder because the PCB will sink the heat from your soldering iron. Solder the signal pins, then the VIN pins, then the GND pin - by the time you get to the GND pins you will have heated the board and it will be easier to solder them.
- If you have never soldered before then DO NOT attempt to solder the flight controller, practice on something else first.

The flight controller is supplied with a bag of pin-headers. Some are straight, some are right-angled. Choose very carefully which ones you want to use. You can solder headers to top or the bottom of the board.

Once you have soldered pin headers in place DO NOT attempt to remove them unless you are highly skilled in de-soldering and have the correct tools. Overheating the board or components will destroy it.

Cables

The flight controller is not supplied with any cables. There are sockets for 3 4-pin JST-SH plugs/cables.

IMPORTANT: Ensure that you leave some slack in your cable routing as this will help if you crash your aircraft.

Cables are available from your retailer.

Do not rely on cable color-coding due to manufacturer variations. Always check before applying power.

MicroSD Card

The MicroSD socket allows MicroSD/SDHC/SDXC to be inserted.

HOWEVER, Currently only MicroSD and MicroSDHC are supported. MicroSDXC is NOT currently supported by Cleanflight.

For full details on how to enable logging, SD card speed requirements and how to view logs please see the Cleanflight software manual.

Your aircraft will fly better if you analyze your logs and tune accordingly.

Getting started

Verify flight controller operation via the configuration software.

- Disconnect the flight controller from your computer.
- Disconnect ALL connectors and headers from the flight controller.
- Install latest STM32 Virtual COM Port Driver http://www.st.com/web/en/catalog/tools/PF257938
- Install and launch the Cleanflight Configurator tool https://chrome.google.com/webstore/detail/cleanflight-configurator/enacoimjcgeinfnnnpajinjgmkahmfgb
- Connect flight controller to computer via USB cable.
- Select the correct COM port if it is not automatically detected.
- Click connect, verify that communication is established. (Fig 3)

2015-12-18 @ 00:48:33 -- Unique device ID received - **0x1e00234b32570220343831**

Place board or frame on **leveled** surface, proceed with calibration, ensure platform is not moving during calibration period

Move multirotor at least **360** degrees on all axis of rotation, you have 30 seconds to perform this task

Restore settings to **default**

Backup your configuration in case of an accident, **CLI** settings are not included - See 'dump' cli command

Heading: 18 deg
Pitch: -26.9 deg
Roll: 15.3 deg

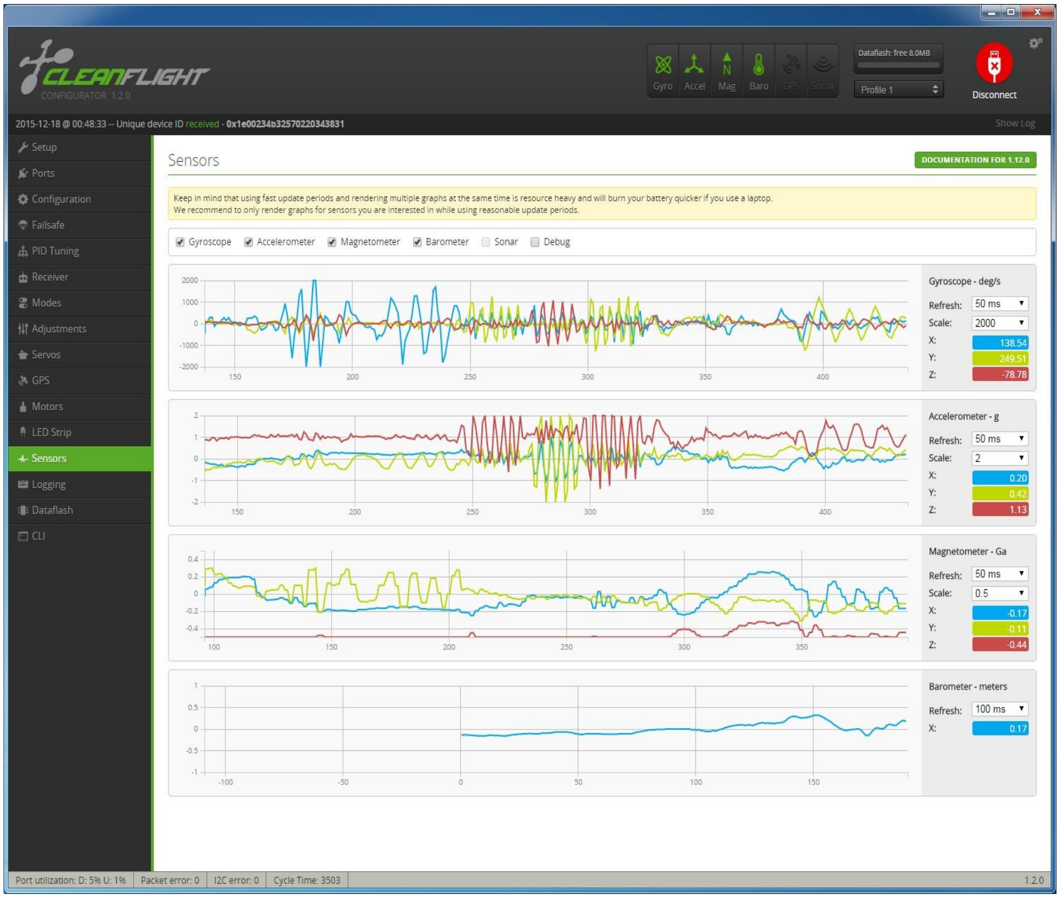
Battery voltage: 0 V
Capacity drawn: 0 mAh
Current draw: 0.00 A
RSSI: 0 %

3D Fix:
Sats:
Latitude:
Longitude:

Port utilization: D: 4% U: 1% | Packet error: 0 | I2C error: 0 | Cycle Time: 3504 | 1.2.0

[Figure 3 - Setup tab after connection established]

- Verify all sensors on your board are giving correct readings. (Fig 4)



2015-12-18 @ 00:48:33 -- Unique device ID received - 0x1e00234b32570220343831

Keep in mind that using fast update periods and rendering multiple graphs at the same time is resource heavy and will burn your battery quicker if you use a laptop.
We recommend to only render graphs for sensors you are interested in while using reasonable update periods.

Gyroscope - deg/s
Refresh: 50 ms
Scale: 2000
X: 138.54
Y: 249.51
Z: -78.78

Accelerometer - g
Refresh: 50 ms
Scale: 2
X: 0.20
Y: 0.42
Z: 1.13

Magnetometer - Ga
Refresh: 50 ms
Scale: 0.5
X: -0.17
Y: -0.11
Z: -0.44

Barometer - meters
Refresh: 100 ms
X: 0.17

Port utilization: D: 5% U: 1% | Packet error: 0 | I2C error: 0 | Cycle Time: 3503 | 1.2.0

[Figure 4 - Sensors tab showing all sensors.

Disconnect and upgrade the firmware using the Cleanflight configurator tool. (See Firmware Upgrade section) For further software configuration see the getting started guide in the Cleanflight manual.

The basic steps, after firmware upgrade, are as follows.

- Choose board alignment - you can mount it in any orientation, not just with the arrow facing forwards.
- Calibrate sensors.
- Configure serial ports.
- Choose model/mixer (default is Quad X) • Enable features.
- Configure receiver, set channel mapping.
- Configure channel mid and endpoints (1000-2000) and trim channels on transmitter.
- Configure voltage monitoring.
- Configure outputs (servos/ESCs)
- Ensure ESC calibration matches ESC configuration, recalibrate ESCs if needed.
- Learn about flight modes and configure channels/switches to activate them as required.
- Learn how to arm/disarm.
- Bench-test failsafe.

- Read safety notes.
- Learn how to download and view your flight logs to help tune your aircraft.
- Insert correctly formatted MicroSD/SDHC Card before your first flight (so you have a log).
- Learn how to recognise un-tuned flight characteristics and the effects of a PID controller. (Watch some videos).
- First flight should be in Acro/Rate mode (the default mode when no other modes are active).
- Tune PIDs.
- Backup settings.
- Contribute to the Cleanflight project with feedback, suggestions, code, etc.

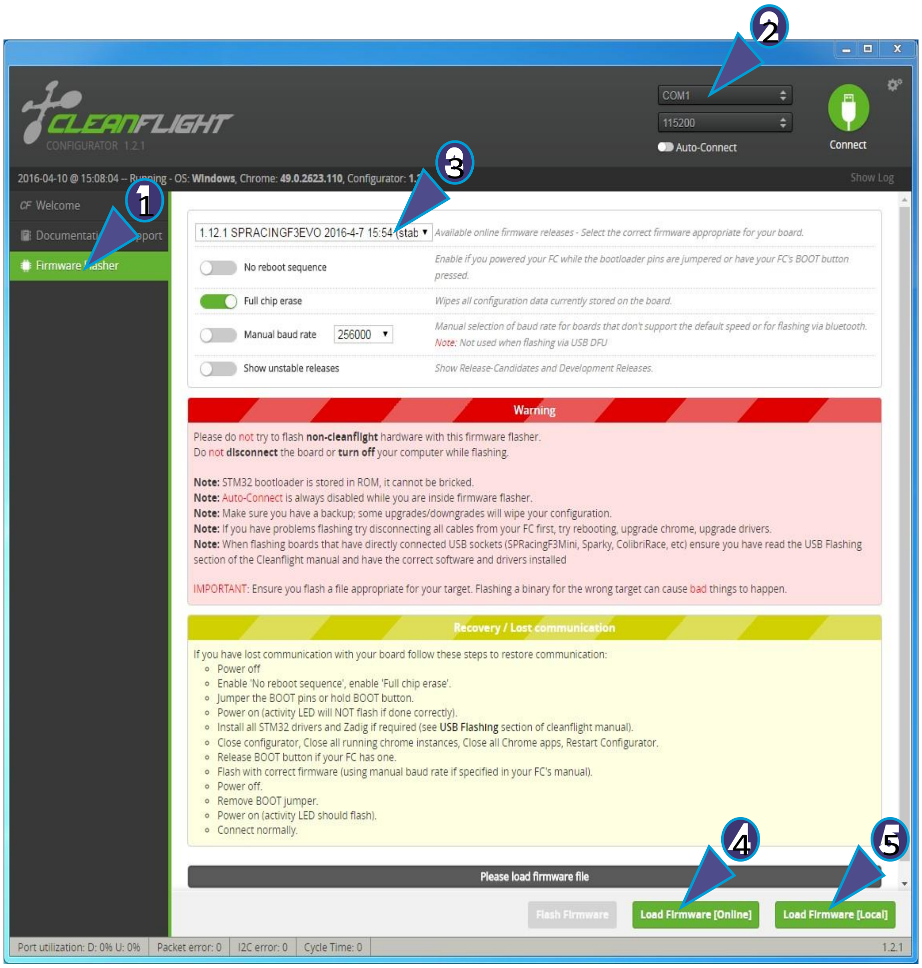
Firmware Upgrade

IMPORTANT:

It is REQUIRED that you immediately upgrade the firmware of the flight controller so that you have the latest features and bug fixes. DO NOT ATTEMPT TO FLY until you have installed the latest CLEANFLIGHT firmware.

On Windows USB DFU drivers must be installed. See the Cleanflight manual for latest installation instructions.

1. Click the Firmware Flasher tab.
2. Select the correct COM port and speed, use the default speed unless you have changed it on the FC.
3. Select the latest “SPRACINGF4EVO” stable release. (Do not use “SPRACINGF3EVO” or other targets). If no firmware is currently available do NOT attempt to flash. NOTE: Flashing the wrong firmware can cause the FC to be permanently damaged.
4. Click ‘Load firmware [Online]’ and wait for firmware to download and read release notes before flashing.
5. Click ‘Flash Firmware’. At this point the FC should reboot into DFU mode. DFU will appear in the list of ports and flashing via DFU will proceed. After flashing the virtual COM port will re-appear.



2016-04-10 @ 15:08:04 -- Running - OS: **Windows**, Chrome: **49.0.2623.110**, Configurator: **1.2.1**

1.12.1 SPRACINGF3EVO 2016-4-7 15:54 (stab ▾ — *Available online firmware releases - Select the correct firmware appropriate for your board.*

No reboot sequence — *Enable if you powered your FC while the bootloader pins are jumpered or have your FC's BOOT button pressed.*

Full chip erase — *Wipes all configuration data currently stored on the board.*

Manual baud rate 256000 ▾ — *Manual selection of baud rate for boards that don't support the default speed or for flashing via bluetooth. Note: Not used when flashing via USB DFU*

Show unstable releases — *Show Release-Candidates and Development Releases.*

Please do not try to flash **non-cleanflight** hardware with this firmware flasher.
Do not **disconnect** the board or **turn off** your computer while flashing.

Note: STM32 bootloader is stored in ROM, it cannot be bricked.
Note: Auto-Connect is always disabled while you are inside firmware flasher.
Note: Make sure you have a backup; some upgrades/downgrades will wipe your configuration.
Note: If you have problems flashing try disconnecting all cables from your FC first, try rebooting, upgrade chrome, upgrade drivers.
Note: When flashing boards that have directly connected USB sockets (SPRacingF3Mini, Sparky, ColibriRace, etc) ensure you have read the USB Flashing section of the Cleanflight manual and have the correct software and drivers installed

IMPORTANT: Ensure you flash a file appropriate for your target. Flashing a binary for the wrong target can cause bad things to happen.

If you have lost communication with your board follow these steps to restore communication:

- Power off
- Enable 'No reboot sequence', enable 'Full chip erase'.
- Jumper the BOOT pins or hold BOOT button.
- Power on (activity LED will NOT flash if done correctly).
- Install all STM32 drivers and Zadig if required (see **USB Flashing** section of cleanflight manual).
- Close configurator, Close all running chrome instances, Close all Chrome apps, Restart Configurator.
- Release BOOT button if your FC has one.
- Flash with correct firmware (using manual baud rate if specified in your FC's manual).
- Power off.
- Remove BOOT jumper.
- Power on (activity LED should flash).
- Connect normally.

Port utilization: D: 0% U: 0% | Packet error: 0 | I2C error: 0 | Cycle Time: 0 | 1.2.1

IMPORTANT: Verify operation using latest CLEANFLIGHT firmware before attempting to use alternative firmware. Not all features may be supported by alternative firmware.

Troubleshooting

Q: Unable to flash firmware via DFU.

A: Likely the correct DFU drivers are not installed. Put the board in BOOTLOADER mode (Press BOOT button, connect USB cable). Then update DFU drivers. See USB Flashing section of Cleanflight manual for details.

Q: The COM port does not show up.

A: Ensure latest USB STM VCP (Virtual Com Port) drivers are installed.

Q: DFU does not appear in the port list when flashing.

A: Try disconnecting the FC, press and HOLD the boot button. Connect the USB cable. Release BOOT, retry flashing.

Q: I have no LEDs on at all.

A: Check 5v supply. (Battery Charged?, USB cable damaged?)

Q: The status LED never lights or is always on.

A: Follow recovery procedure in the configurator. Likely caused by flashing wrong firmware. Could also be caused by stuck-down BOOT switch - check button for dirt, etc.

Q: Unable to connect and a repeating light sequence occurs on the activity LED.

A: Check the Cleanflight manual for how to interpret the error code (count the long flashes).

Q: Unable to connect to the board (and status LED shows non-repeating pattern at boot-up).

A: Close all Chrome browsers and Chome apps, retry. COM Port drivers not installed? Try connecting via different UART OR Reset the board to defaults using buttons OR reflash firmware using ‘full chip erase’. (Likely caused by mis-configuration of ports).

Q: Transponder code not recognised by receiver.

A: Check transponder enable bridge. Check IR LED orientation and light beam exit path. Ensure transponder feature enabled in software. Ensure transponder code correctly configured.

Q: The COM port does not show up after flashing firmware.

A: Wrong firmware was flashed; Use SPRacingF4EVO target and follow recovery steps in the Cleanflight Configurator firmware flasher to restore the firmware.

Q: The receiver tab does not show any activity.

A: Check configuration tab - receiver mode. Center sticks, configure endpoints and trim on transmitter. Check ports tab if using Serial RX. Check wiring. Check S/R bridge, check T2/TLM bridge.

Q: You have a problem not listed here.

A: Reset and/or upgrade the firmware, try again, report issues via the forums - links are in the configurator on the ‘Documentation & Support’ tab. Before contacting your retailer reflash with latest CLEANFLIGHT and double check - perhaps the firmware you are using is old or incorrect?

Q: Only 5V LED lights up, No BLUE 3V LED or RED STATUS LED.

A: Most common cause is misconnection or short of VBAT pins - FC will be destroyed! Check for short in Spektrum Satellite cable if using Spektrum Satellite RX. Always caused by destroyed CPU or overloaded/shorted voltage regulator.

Transponder

The SPRacingF4EVO features a IR LED transponder system. For optimum performance ensure you read the following section.

Mounting - Ensure that the IR LED can shine light from its installed location onto the track-side receiver, unobstructed. e.g. ensure that props, motors, arms, etc, are not in the way of the light shining from the LED.

Orientation - Ensure the LED points outwards from the aircraft towards the track-side receivers. The more receivers you use the better the code reception will be.

Verifying operation - Ensure that the TRANSPONDER feature is enabled in Cleanflight. Ensure that the correct pads of the IR/LED SELECT solder pads are bridged with solder. Ensure that the TRANSPONDER code has been configured via the Race Transponder configuration section in the Cleanflight Configurator. Once this is done you can use a CMOS/CCD camera without IR block pointed directly at the IR LEDs and you should see them pulsing. A mobile phone camera works well for this, your FPV camera and screen/goggles may be OK if it doesn’t block

IR light.

Once you have verified that the IR LED pulses IR light then you can scan your SPRacingF4EVO past one of the iLAP receivers. The iLAP receiver just needs power, no software configuration is required to verify that the code is working.

The first time the iLAP receiver can read the transponder code the LAP light will pulse once, while the iLAP receiver can read the code then the activity light (ACT) will be flashing.

Each time the LAP light flashes the receiver transmits transponder code and timing information via it’s COM port to the computer it is attached to.

By default the iLAP receivers will not transmit the transponder code to the PC twice in a row unless the transponder code has not been received for over one second.

For further iLAP receiver configuration and setup advice refer to the iLAP documentation.

Hardware Specifications

- 36x36mm board with 30.5mm mounting holes
- Weight ~5 grams
- STM32405 CPU, 168Mhz inc FPU
- MicroSD card slot (SD/SDHC, upto 32GB)
- ICM20602 accelerometer/gyro (connected via SPI)
- BMP280 barometer
- Built-in inverters for TELEMETRY and SERIAL RX.
- MicroUSB socket for configuration and ESC programming

- 3 LEDs for 5V, 3V and STATUS (Green, Blue, Red) • Supplied with straight and right-angled pin headers.
- BOOT switch for easy firmware upgrades.
- Stackable with SPRacingF3OSD/PDB.
- IR LED and unique code for race transponder (optional)
- Supports Cleanflight, Betaflight and iNav firmware.
- Cleanflight logo.
- 8x 3pin though-holes for direct-solder or pin headers for ESC/Servo connections
- 1x 2pin though-holes for pin headers for UART1 RX/TX
- 1x 6pin though-holes for direct-solder or pin headers for UART2/PPM/Telemetry/LiPo (for standard SerialRX/ SBus/PPM lead)
- 1x 4pin though-holes for pin headers for UART3
- 1x 6pin though-holes for direct-solder or pin headers for IR LED, RSSI, Current Monitoring, and Buzzer connections
- 1x 3pin though-holes for direct-solder for JST-ZH connector for Spektrum Satellite receivers (connector supplied).
- 1x 4pin top mounted, JST-SH socket for I2C
- 1x 4pin top mounted, JST-SH socket for UART2/PPM
- 1x 4pin bottom mounted JST-SH socket for SWD debugging
- 1x 7pin bottom mounted solder pads VTX connections
- 2x top-mounted direct-solder pads for LED Strip

Credits and acknowledgements

Hardware design by Dominic Clifton.

Manual by Dominic Clifton.

Seriously Pro website by Dominic Clifton.

Cleanflight firmware and GUI tools are maintained by Dominic Clifton.

Blackbox flight logging component, tools and SD Card support by Nicholas Sherlock.

The flight controller software is based on MultiWii by Alex Dubus. The GUI tool was based on software by cTn.

Big thanks to the SPRacingF4EVO test pilots.

Thanks to Painless360 / Lee Schofield for ideas and tutorial videos.

https://www.youtube.com/user/Painless360/playlists

Thanks to everyone who provided feedback on previous SP Racing products.

Software support and contributions from many, many authors. For a complete

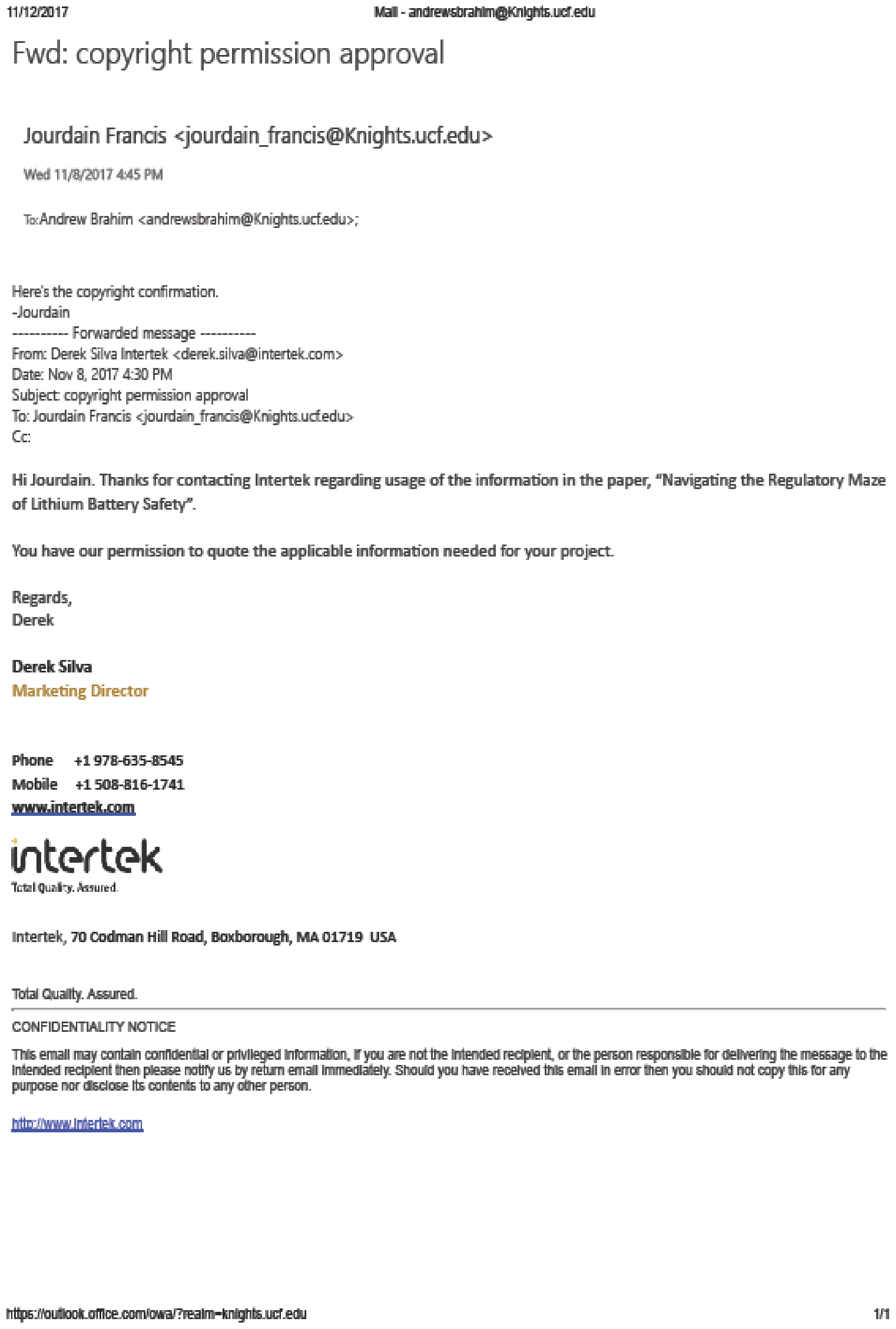
list see github contributors.

https://github.com/cleanflight/cleanflight/graphs/contributors

Community

Thanks to everyone from the AWESOME Cleanflight community for code, artwork, support, ideas, feedback, and everything else. Without you all this product would not exist.

Appendix C: Intertek Copyright Approval Letter



11/12/2017 Mail - andrewsbrahim@Knights.ucf.edu

Fwd: copyright permission approval

Jourdain Francis <jourdain_francis@Knights.ucf.edu>

Wed 11/8/2017 4:45 PM

To:Andrew Brahim <andrewsbrahim@Knights.ucf.edu>;

Here's the copyright confirmation.
-Jourdain
---------- Forwarded message ---------
From: Derek Silva Intertek <derek.silva@intertek.com>
Date: Nov 8, 2017 4:30 PM
Subject: copyright permission approval
To: Jourdain Francis <jourdain_francis@Knights.ucf.edu>
Cc:

Hi Jourdain. Thanks for contacting Intertek regarding usage of the information in the paper, "Navigating the Regulatory Maze of Lithium Battery Safety".

You have our permission to quote the applicable information needed for your project.

Regards,
Derek

Derek Silva
Marketing Director

Phone +1 978-635-8545
Mobile +1 508-816-1741
www.intertek.com

intertek
Total Quality. Assured.

Intertek, 70 Codman Hill Road, Boxborough, MA 01719 USA

Total Quality. Assured.

CONFIDENTIALITY NOTICE

This email may contain confidential or privileged information, if you are not the intended recipient, or the person responsible for delivering the message to the intended recipient then please notify us by return email immediately. Should you have received this email in error then you should not copy this for any purpose nor disclose its contents to any other person.

http://www.intertek.com

https://outlook.office.com/owa/?realm=knights.ucf.edu 1/1