Resources

Rocket Dynamics

- MIT lecture on rocket vectoring
- MIT the rocket equation
- MIT dynamics course
- Caltech Harrier Vectoring
- Control lecture notes
- Basic Rocketry

GPS spec

- SBAS Fundamentals
- Software Defined GPS
- GNSS Augmentaion
- RTKLIB wiki

Control and sensing

- Sensor Fusion Paper (GPS and IMU)

Kalman Filter

See more at Kalmanfilter.pdf The Kalman Filter is a mathematical statistical technique that is usually utilised for avionic control systems. The filter is used as an optimal solution for tracking and data prediction. This is extremely useful for something like a rocket as it will help keep track of its position in space. The filter uses a lot of computational power, for that reason, if we wanted to use it for this avionics package we would need to basically start from scratch with the PCB design. This is because we need something stronger than the teensy i.e. we need an FPGA that could be placed on the board. This would've taken us a lot of time that we did not have.

Proportional-Integral-Derivative (PID) Control

A PID controller a type of controller that uses proportional, integration and derivative factors when correcting for errors in the system. Each of these mathematical operations are associated with a coefficient; that is proportional has the coefficient Kp, integration has the coefficient ki and derivative has the coefficient kd. Each of these coefficients are varied until the system being controlled is stable and has the desired response. It is easy to implement but prone to instability especially if the rate in error changing is non-linear. In simple terms this means that if the data passed through the controller has a large variance, then the controller will be unstable. Therefore tuning the controller is crucial when implementing a PID controller.

The process of a PID controller controlling a system is as follows

- Have a fixed set point (i.e. fixed input) into the system.
- Detect errors between the set point and the output (error detection).
- Errors are passed through the controller.

- The new error corrections made by the controller is passed through the system.

The error detection is the feedback loop and this allows the system to try and maintain the ouput as close to the set point as possible.

There are several techniques such as the Ziegler-Nichols method(s) for tuning the PID coefficients but this is only useful when an accurate mathematical model of the system is known. Therefore the most useful way is to perform trial and error operations (manually change the coefficients) until the system has the desired response.

Proportional-Integral-Derivative (PID) Control

PID consists of three basic coefficients; proportional, integral and derivative which are varied to get the optimal control. PID is a popular and robust control algorithm that is used in industry. In a system there's open loop (which is when no feedback is gained and an disturbances will heavily affect the system output an therefore its gain) and closed loop (a feedback loop is present which accounts for disturbances).

Proportional

The proportional component depends on the error i.e. the difference from the set point and the process variable. The proportional gain depends on the ratio of the output to the error. Increasing the proportional gain will increase the speed of the system. Anything above the proportional gain will cause the system to oscillate out of control i.e. become unstable.

Integral

The integral component sums the error over time. The integral response will increase gradually until the error becomes zero; the effect is to drive the steady state error (the error between the target value and the value which the proportional gain has allowed us to reach while still being stable) to zero. When the integral saturates the controller, this is called integral wind up. To fix integral wind up, you can reset the integral component, however, a more effective way is to bound the integral component.

Derivative

The derivative component is proportional to the rate of change of the process variable. The derivative component is usually very small because it is highly sensitive to noise. Its main function is to basically make the control respond faster. If the sensor feedback signal is noisy or the control loop is too slow then the derivative component can make the system unstable.

Launch Regulations

- No launches are permitted within 4km of the airport. Within a 4 to 8km range of the nearest aerodome launches must be below 400ft according to CAA regulations. For higher altitude launches, these must take place at least 8km away from the nearest aerodome
- Specifiying the launch day is not necessary for launching the cone rocket, rather, a launch window should be sent to the Upper Hutt City council to avoid clashes with any events taking place in Trentham park
- The above applies for MVP1 and MVP2
- CAA rules

- Website for checking wind speeds and direction
- [Kelburn average wind wpeeds] (https://www.windfinder.com/windstatistics/chaffers-marina-wellington?fspot=kelburn_wellington)
- NZ Rocketry Association rocket safety
- NZ Rocketry Association safety code (launch site dimensions for classes A- G)

Wellington Region UAV Launch Restrictions

The following is a list of specific regulations put in place by each city council in the Greater Wellington Region if available.

Greater Wellington Region

GWRC Drone/UAV Page. UAV use must conform to CAA regulations. There are some recommended parks for UAV use such as Whitireia Park.

Wellington City Council

WCC Drone Page. As most of Wellington City is within controlled airspace, Wellington Control Tower permission must be attained unless it is a shielded operation. RPA use is permitted in WCC parks so long as they conform to CAA regulations and have the aforementioned permission.

Upper Hutt City Council

UHCC Drone Page. Drones may be flown over parks and roads in accordance with CAA regulations and so long as it does not harm anyone or interfere with traffic.

Hutt City Council

HCC Drone Policy. Drones may be flown over Hutt City Council owned and managed parks and reserves, with the exception of Taita Cemetery, in accordance with CAA regulations and so long as it does not harm anyone or interfere with traffic **for recreational purposes only**. As the project is a university project, permission must first be obtained from the HCC.

Porirua City Council

PCC Drone Permit. In Porirua, a permit is required to fly drones on council land.

Masterton City Council

Aerodrome Information. The Aerodrome page states that in order to fly RPA's or drones within four kilometres from the aerodrome, permission is needed from the airport manager.

Rocket Simulation

- Open Rocket open source in depth rocket simulation used to test rocket designs
- Thrust Curve simulation of different ranges of rocket motors
- VCP predict rocket performance given design and motor parameters

Communication Resources

- Linx Antennae Catalogue

PCB

Power Management

Battery Elimination Circuits

Battery Elimination Circuits (BEC) are often used in model aeroplanes. They include often include some chips to help monitor and protect the batteries, this is particularly important for LiPo batteries as they can cause fires if not protected properly. They also contain a switching regulator because they are a lot more efficient than other regulators at the cost of space and noise. They also have a circuit that allows them to power parts of a system and not others, this is most commonly used for model aeroplanes so that as the battery runs low power can be cut to the motors to preserve the control system and the plane can still be controlled to glide back to the user.

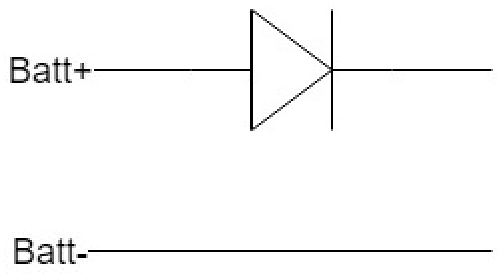
However, in our application, the servos are not regulated as they need the full 7.4V from the batteries which makes the BEC less useful. We are also using Li-ion batteries which are more stable than LiPo, although some battery protection chips are probably still a good idea not many Li-ion BEC's exist.

Instead, a Low Drop Out (LDO) voltage regulator would work better for us to ensure the PCB remains powered as the battery depletes. These are also better than the previous LM7805 voltage regulator that was used as it has a much lower drop out voltage, where the LM7805 had a dropout voltage of only one volt less than the battery outputted which resulted in regular failures. These also come in SMD options which save a lot of space.

Reverse Polarity Protection

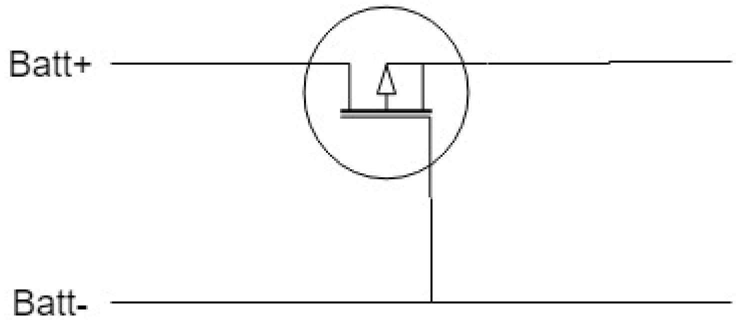
A big risk for any battery operated is that the battery is plugged in the wrong way round and it damages the board. This has happened in this project in previous years and is something that must be considered.

A diode can be used to prevent any current from flowing backwards from the battery as shown below.



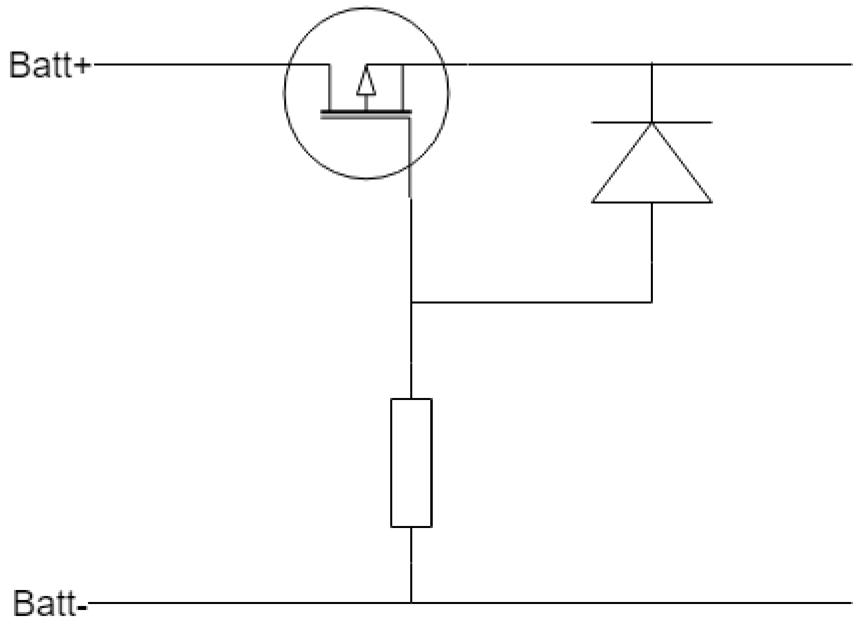
The big drawback of this is the built-in voltage drop across the diode. The biggest current draw in the system is the servos which can draw roughly 1 Amp each. Even assuming a Schottky diode is used which only has a 0.6V voltage drop across it the diode will still need to dissipate 1.2 Watts(0.6V*2A) which can generate a lot of heat on the board as well as reducing the power available to the rest of the components.

Another option is using a P-type MOSFET connected as shown below. in this configuration the transistor will act as an open circuit if Batt- is greater than Batt+ and not let any current flow, this configuration has a much smaller power draw than a diode but is slightly more complex and takes up more space.



Another concern is if the Drain-Source breakdown voltage of the transistor is far greater than the battery used otherwise it will break down and not protect the circuit. The Drain-Source resistance when on needs to be as small as possible to reduce the power dissipation. The Gate-Source voltage also needs to be quite large to work.

If the Gate-Source voltage is lower than the battery being used an extra circuit is needed to reduce the voltage across the gate and source so that the transistor will not break down, as shown below.



The important thing to consider here is that the voltage drop across the diode is less than the Gate-Source voltage of the transistor.

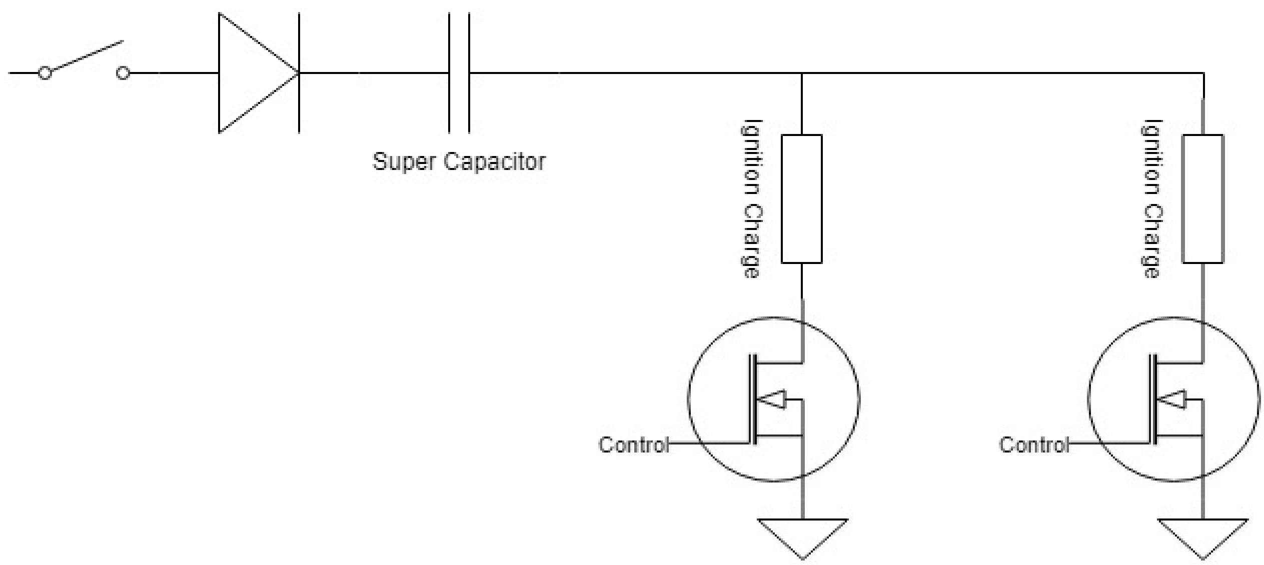
Ignition System

The igniter charges used require roughly 1Amp to trigger. There are three separate charges that need to be ignited; launch, primary parachute and secondary parachute. Two parachutes are needed for a safe landing, the first parachute slows it down enough for the second to work. If a big parachute was used initially the deceleration would likely damage the rocket or rip the parachute off entirely. A backup ignition charge is also required for each charge because although it is very unlikely that an ignition charge will fail the cost of such a failure is most likely the complete destruction of the rocket, therefore backups are required. The backups will be placed within range of the same charge and fired sequentially, if fired at the same time they will draw a lot more current and could cause damage to the motor. When fired one after the other if the first one succeeds it should burn up the second charge so that it will not ignite at all.

Initially, Darlington arrays were considered for the ignition as they come in compact packages containing up to seven Darlington pairs which are convenient as six are needed. Darlington pairs work by placing two bipolar junction transistors in series to increase the gain of the system at the cost of a higher threshold voltage. The problem, however, is that Darlington pairs cannot provide the current that the ignition charges require and are therefore not suitable for this application.

A reservoir capacitor is also recommended as when the ignition charges draw their current it has the potential to draw more current than the battery is able to provide which means that other components could not receive the current they need to function, this is called browning out. To ensure the ignition charges always have access to the current they need a diode is needed to stop the capacitor from feeding back into the rest of the circuit.

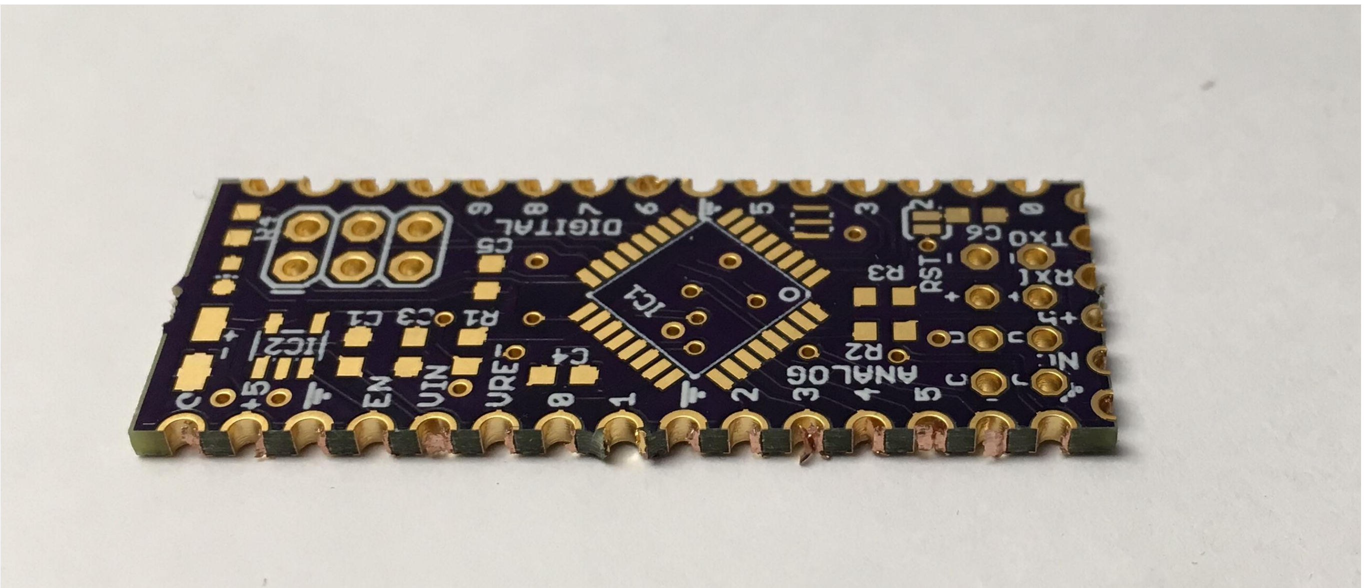
A safety switch or jumper should also be used in series with the ignition system as an extra safety precaution so that no ignition charges can possibly ignite prematurely.



Teensy 3.6

Several times it was considered to surface mount the Teensy to the PCB, this would save a lot of space as the pins would not interfere with components on the underside of the board. Two methods were researched for this project, firstly surface mount headers could be used which are similar to standard through-hole headers but instead of going through the board they turn at a right angle and are soldered along the surface of the board. The biggest drawback of this option is that they are usually slightly taller than normal headers and the footprints required for them can take up more space than the through-hole option. This limits the surface area that can be used for other components.

The other option was castellation which is used on a lot of surface mount chips, it is the best way to solder one silicon board to another. In order to guarantee a good connection to the copper pads, they are put along the edge of the board as seen in the image below. This allows the board to be soldered directly to the surface of the main PCB.



source:https://docs.oshpark.com/tips+tricks/castellation/

The problem with this approach is that in order to castellate the Teensy the edges would need to be cut off which is a risky process as any of the pads could come off from the heat of cutting, many of the pads are not needed and this may not be catastrophic but it could be a big problem.

The other problem with attaching one PCB to the surface of another, especially such a long board, is that if the PCB flexes during the launch the Teensy would likely break off very easily.

Inertial Measurement Unit

The MPU-9250 was chosen due to the cost and accessability. It has accelerometers, a gyroscope and magnetometer. The biggest problem with these sensors is that the accelerometers and gyroscope are both relative measurements and are subject to drift. Often sensor fusion is used to account for gyroscope drift as it can use Earth's gravity as a constant acceleration, however when accelerating at a rate greater than gravity this is no longer true and cannot be relied on.

Initially the SparkFun breakout board for the MPU-9250 was used as it made the chip breadboard compatible making testing easier. One issue found with the Sparkfun breakout board is that they are shipped configured for I2C communication and in order to change to SPI a jumper must be swapped. This is not documented anywhere but the PCB files show this.

Another problem is the size, the breakout board wastes a lot of space which is an issue with the rocket as space is a concern. To reduce this problem the IMU can be used without the breakout board but this poses it's own problems. Firstly some PCB manufacturers increase the price of their boards dramaitcally in order to produce wires at the pitch required for such small chips. The other problem is soldering, a microscope is reccomended when soldering such a small component. If these problems can be overcome this is the best solution.

Radio Module

LoRa

LoRa is a long rangle modulation technique, it has the longest range and one of the lowest power costs of any modulation. The biggest drawback is that it uses a proprietary modulation which means that it cannot be decoded by any reciever and the appropriate software, it requires another LoRa module. This can increase the cost of the project by quite a bit as they are already quite expensive and at least two are needed.

Antenna

The antenna of the radio module has been found to make little difference at the ranges tested and so a simple wire whip antenna should suffice for most mid-range application.

Breakout Board

The RFM9x radio module can be bought with a breakout board, upon closer examination of the schematic of said board the main purpose is to make it usable in a wider range of applications. The breakout's main attractions is that it makes the radio module breadboard compatible which allows for easy prototyping. The other function of the board is that it allows the module to be used with 5V logic. The main part of the module uses 3.3V logic so the breakout board has a built in voltage regulator and buffer to set all inputs to 3.3V to interface with the module. For the purpose of the avionics package the Teensy runs as 3.3V and provides power rails at this level, the breakout is therefore not necesary for this project.

GPS Module

Antenna

Patch Antennas

Patch antennas use a small plane above a larger ground plane, this configuration then resonates and acts as an antenna. The are relatively compact and have a low profile. The drawback is that they have a very directiona radiation pattern, while they are very sensitive in one direction they have very low sensitivity in most other directions.

Chip Antennas

Chip antennas use a 3D configuration within the chip to act as an antenna, this allows them to provide a more omnidirectional radiation pattern. The biggest downside is that they are still have a few deadzones in the radiation pattern but they are very light and compact.

Helical Antennas

Helical antennas have the most omnidirectional radiation pattern with high gain in nearly every direction. The biggest downside is that they can be quite large and heavy especially when they include a ceramic core.

Testing: Continuous Integration

Simulator

The team researched a simulator for the Teensy 3.6 to use for on-board Continuous Integration testing. While there were some simulators found, none of them were fully documented or implemented:

- http://www.utasker.com/index.html
- https://forum.pjrc.com/threads/25309-Teensy-Simulation (for Teensy 3.1)

Server Board

Another option was to connect another Teensy to a server to run tests automatically whenever a change was made to the GitLab repository. However, this idea was brought up late in the project and as such we did not have the budget to implement the idea. Also, it would mean that our tests would be difficult to run for any third parties.

Google Mocks

The last option the team considered was using Google Mock to fake the external libraries. This way, the tests would be able to track function calls easily and avoid the difficult step of feeding fake data into external libraries.

The issue is that platformio has its own build system which the CMake auto-generated files simply calls as a custom command. To test our code, we need to compile it with the same compiler as GTest.

ARM

The codebase could be compiled in ARM (meaning the code we've written, not including the platformIO libraries) using ARM as it is written in arduino code.

Pros:

- Tests are run on the same environment as the code runs, potentially increasing the reliability of tests

Cons:

- GTest can be compiled in ARM, but requires configuration which could be time-consuming

x86

The codebase could be altered to only use basic C++ and then compile it with x86. This is the approach that was used in the end.

Pros:

- Can use CI and run tests on ECS PCs

Cons:

- Requires changes to the codebase
- May not be able to easily test control code which heavily relies on Arduino maths