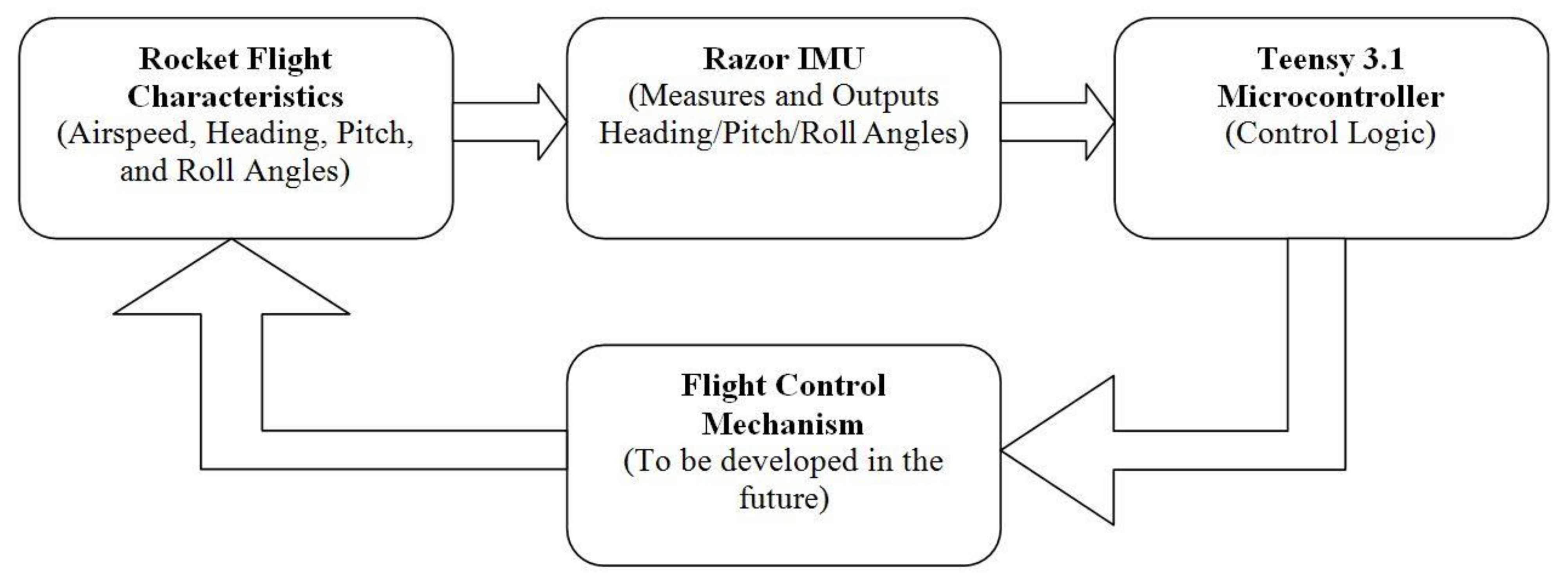
Vertical Attitude Correction Flight Computer for High Power Rockets

Christopher Gosch
Professor John Weyrauch
Undergraduate Research Opportunities Program

Introduction/Motivation

The purpose of this project is to determine whether a cheap, lightweight, reliable flight computer that can be used to keep a rocket's attitude vertical can be developed. In general, high power rockets are designed and built to be aerodynamically stable simply through the sizing and placement of the fins and the mass distribution of the rocket. In order for a rocket to be stable, its center of gravity must be at least one diameter of the rocket ahead of the center of pressure, which is the center of aerodynamic forces on the rocket. This means that whenever a rocket is forced off of a vertical trajectory, the forces of air on the fins will produce a moment that will cause the rocket to rotate back to a vertical trajectory. However, the moment generated by the fins is not always a perfect match to the change in the rocket's attitude, which can result in under-correction or overcorrection The fins will counter changes in attitude such that the rocket tends to fly into the wind at trajectories that are far from vertical. Although this is not usually dangerous, this generally leads to large deviations from a vertical flight trajectory. This poses two major problems. For rockets intended for extremely high flights or even orbital launches very precise trajectories must be followed, and deviation from vertical flight makes it impossible to achieve the proper trajectory. For rockets competing in competitions with a specific altitude goal, this large change in flight trajectory can result in flights that greatly miss their altitude goal [1]. Thus, ensuring that a rocket maintains a vertical attitude throughout its flight is a critical problem, and solving this problem would be a great step forward in University of Minnesota rocket design.


Rocket Flight Characteristics (Airspeed, Heading, Pitch, and Roll Angles)

Razor IMU (Measures and Outputs Heading/Pitch/Roll Angles)

Teensy 3.1 Microcontroller (Control Logic)

Flight Control Mechanism (To be developed in the future)

Figure 1: The Negative Feedback Control Loop Used. Note that the flight control mechanism has not been developed yet.

Methods

It was decided very early on to use existing commercial microcontrollers and sensors to maximize reliability. After that, the initial parts of this project focused on running trade studies to determine what the best possible sensor arrays and microcontrollers were. For the sensors, this involved looking at sensor poll rates, sensor precision, and the ease of use. The trade study resulted in the Razor IMU being chosen to provide sensor data, as it consisted of a nine degree of freedom inertial measurement unit coupled to an Arduino Pro Mini, which performed the sensor data fusion and output it over a serial bus to the microcontroller. The microcontroller chosen was a Teensy 3.1, which was capable of running standard Arduino Sketches, had three Serial Busses and twelve Pulse-Width-Modulation capable pins. It also had the fastest clock speed of any of the microcontrollers used and the smallest overall size, thus making it perfect for use inside of a rocket. Power for the system is provided by two lightweight 3.7V rechargeable Lithium-Ion batteries, which helps to keep the weight to a minimum.

The control algorithm used is a simple feedback loop. The IMU reads in the heading, pitch, and roll angles, and if they are greater than 10 degrees off of the vertical values the controller will command one or more of the controls to activate. An overall safety lock was also built into the system so that it would automatically disable itself if any of the angles exceeded 45 degrees from the vertical. Since there is not a mechanical control system for this structure yet, outputs were represented by tying a single LED to each output. The LED would light up when that output (representing a control surface or thruster) would activate. This entire system was then rigorously tested to ensure it performed properly. However, it will not be fully complete until it has been integrated with a control system and tested with this, which is a future step for this project.

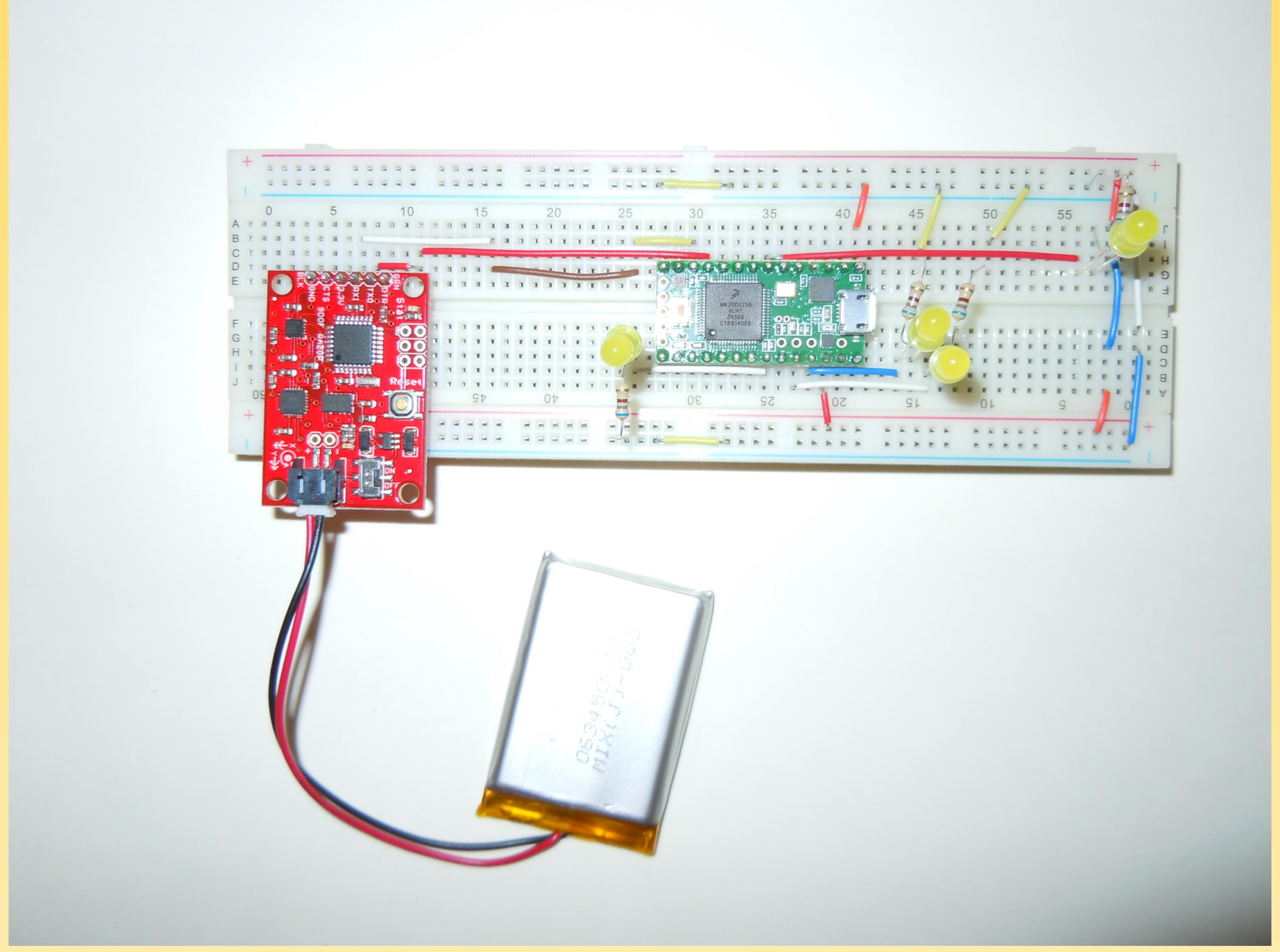


Figure 2: A picture of the system as it was tested on the breadboard.

Circuit Diagram

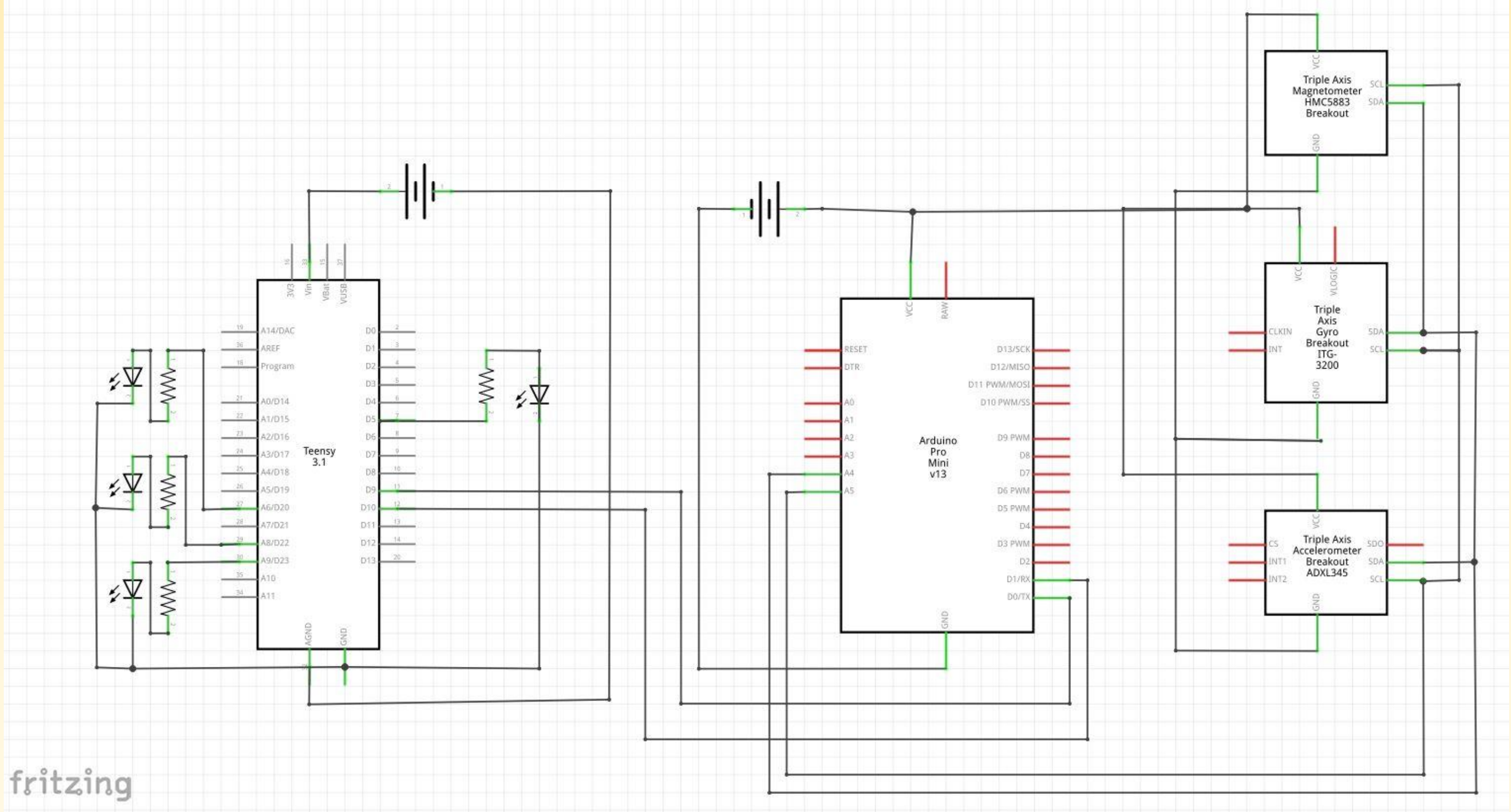


Figure 3: A circuit diagram of the system. The Razor IMU is represented as its individual components.

Results

A working breadboard prototype of the system has been successfully built and tested in the lab. It currently runs with a cyclic rate of approximately 10 Hz, which is about half of the maximum possible cycling rate of 20 Hz. The total system costs approximately $125, is 6.5"x2.1", and weighs about a quarter pound.

Future Work

There are several eventual goals for this project. The first is to mate it with a suitable control system, which will likely be either a thruster system or a set of control surfaces. Once this has been done, substantial testing will need to be done in order to ensure that it works with the new system correctly. Additionally, this will allow the development of a more advanced control algorithm, which will hopefully include roll control in addition to the current pitch and yaw controls. The other major goal is to continue to make the flight computer more robust by designing and building a single printed circuit board with all of the components permanently mounted on it. This will greatly reduce the chances of any type of wiring or structural failure compared to installing everything on a breadboard, and it will also allow the unit to be made even smaller and easier to work with inside of a rocket. Hopefully all of this will result in an easy to use, reliable system which can be used for the active control of future rockets.

Acknowledgments

Special Thanks to Faculty Advisor Professor John Weyrauch, and the Undergraduate Research Opportunities Program.

University of Minnesota
Driven to Discover℠