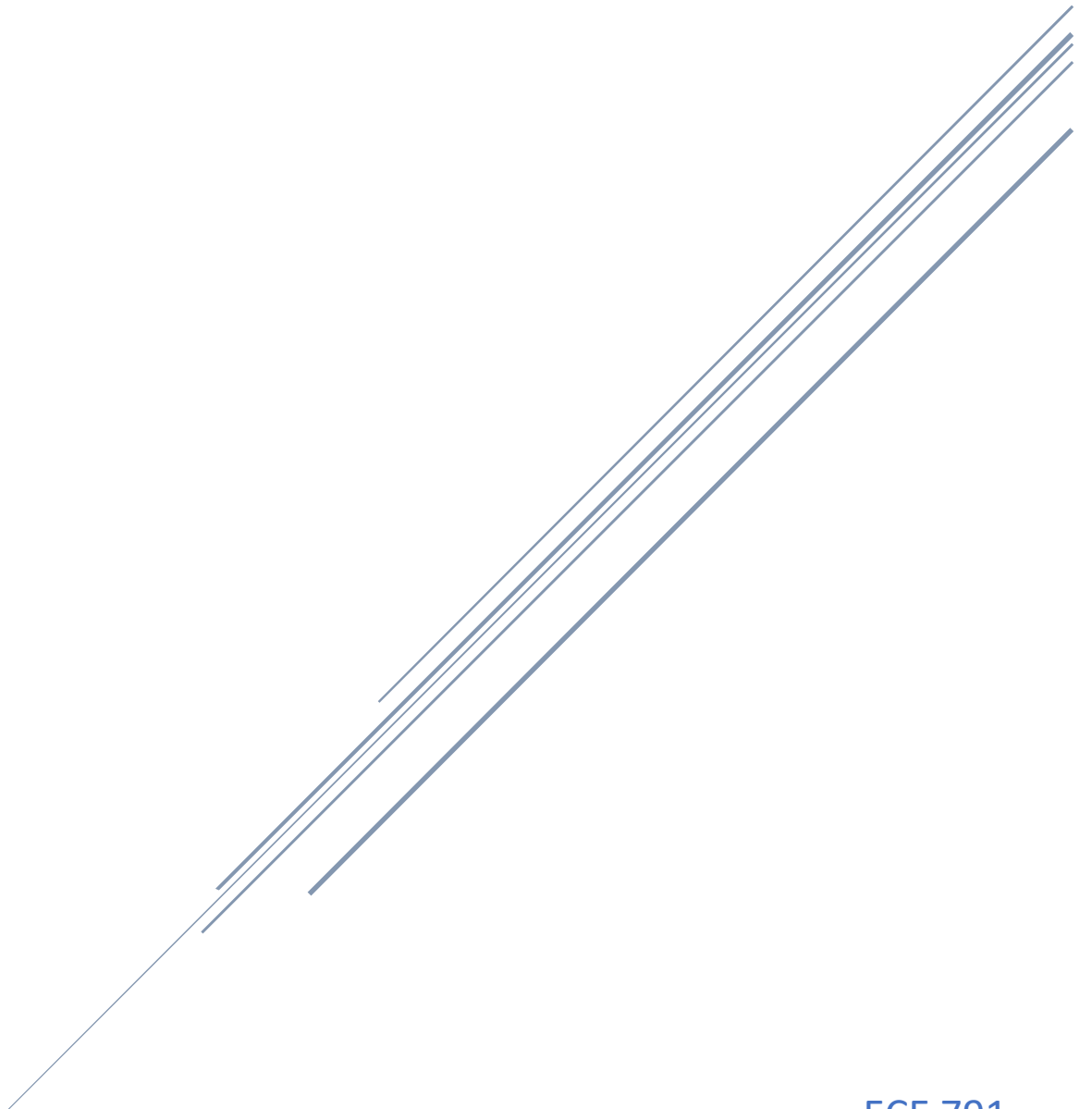


SENIOR CAPSTONE PROGRESS REPORT – ET NAVSWARM ESC

Nicholas Snyder, Qiaoyan Yu Ph.D., Md Shaad Mahmud Ph.D.



ECE 791

Submitted on 8 December 2023

Table of Contents

Project Overview.....	1
Background	1
Problem to solve	2
Design Overview	3
Current Status	3
Progress.....	3
Changes to plan.....	4
Problems encountered	4
Work done/word proposed	5
Active tasks	5
Future Work.....	5
Discussion and Conclusion	5
Two constraints beyond technical	5
Conclusion.....	5
References	6
Appendices.....	6
Timeline.....	6
Parts list.....	7
Schematics/Images	8
Data sheets/user manuals	11

Project Overview

Background

ET NavSwarm is an interdisciplinary group with the end goal of sending a swarm of small autonomous rovers to another planet to prospect for materials. The group is now implementing the particle swarm algorithm to all the rovers to communicate with each other. Some areas for improvements the group have identified are active battery monitoring and hardware longevity.

A brushed motor controller is a device that regulates the speed, direction, and torque of a brushed motor, which is a type of electric motor that uses brushes and a commutator to switch the current in the windings. A custom brushed motor controller can be tailored to the specific characteristics and requirements of the motor, such as the voltage, current, power, and load.

What I hope to learn from completing this project is PCB design and real-world applications of circuits and microcontrollers. PCB design is not something taught in the curriculum but is very valuable to a professional electrical/computer engineer. Embedded circuit design with microcontrollers is another necessary skill to be an effective computer engineer.

Problem to solve

The hardware failures usually stem from the motor controllers. The cause of this is most likely the low-quality components on the board. The most recent design specification uses two HW-039 H-Bridge boards for brushed motor control. One for each side of the rover. The HW-039s are widely known to be unreliable and fail seemingly randomly. Designing a high-quality controller with an IGBT or Power FET would help the longevity of the rovers tremendously.

When testing on Boulder Field it's hard for the group to monitor the states of charge of all the different rovers. This leads to batteries being over discharged and ruined. A way to clearly indicate when a battery's voltage drains too low and eventually turning off the rover before the battery dies would be very helpful. This would also reduce the waste created by bad lithium batteries. A problem is already appearing in the lab. An over discharged lithium-ion battery can be a safety risk as well, making it more susceptible to a thermal runaway event that can result in a large fire or explosion.

Design Overview

My design for the motor controller does away with an H-bridge. This is because the custom airframe loaded into the Pixhawk by QGroundControl does not have capabilities for reverse thrust. The main reason to use an H-bridge is to reverse the motor direction meaning the simplification brought by my design would not lose any functionality. The Pixhawk sends out a PWM signal from its I/O ports that in this design is connected to the base of each IGBT. When a pulse is high the base closes the circuit allowing current to flow.

Most of the cells available for testing are 4s Li-ion pouch cells so I planned for a maximum voltage of 16.8 volts. The motor control is done by a single IGBT with its emitter connecting to each of the four motors. The base is driven by the MCU directly. The MCU is powered by a 3.3-volt output of a custom designed buck converter. My current plan is to read the voltage from each of the balance leads of the pack into the MCU and only allow current to flow to the motors when each of the series cells are above a specified low voltage cut-off. When the cut-off is reached, a red LED will start flashing at the top of the rover's control tower to alert the team. Either an LCD screen or 4 7-segment displays will also be present on the rover to show the cell voltages in real time.

Current Status

Progress

Due to the team last year not progressing much at all and not documenting much. Most of October was spent getting a feel for the current state of the rovers and identifying the immediate problems. One problem was to not slow down the CS team's autonomous integration, we had to get a few rovers up and running as quickly as possible. I led the search to find temporary ESCs to get the rovers moving again. I found some cheap ones on Amazon and connected them to the

rovers once they arrived. We currently have two rovers capable of manual control with a hobbyist RC controller.

I have made a high-level wiring diagram of the rovers to help the CS team show the purpose of every component. When designing the schematic, I started with the motor control section, then moved to the buck converter and then finally to the microcontroller.

Changes to plan

Over the course of this project, I have expanded my goals slightly by adding functionality. Some of these improvements have been reading the voltage of each cell and not just the pack. I did this because due to the Lithium-Ion chemistry, each cell has a different internal resistance and therefore one cell could have a slightly lower capacity. This would lead to that individual cell having a lower voltage compared to the other cells in the pack. Draining this cell too low too often can completely kill the cell and compromise the pack's performance. The most critical impact of this is the speed of an electric motor is proportional to the voltage across it.

Problems encountered

When designing the buck converter section, I ran into an issue with the chip to control the switching of the high voltage signal. I found this chip in the component selector menu in Multisim and then searched the part on the internet to find its datasheet. I found the datasheet from Texas Instruments to find the chip was completely different and needed a more complex circuit to be used with it. My plan of action so far is to find a new chip like the designs I have found online. This is a relatively recent problem, so a solution has not yet been found.

Another problem I am dealing with is for some reason, my Multisim license deactivated sometime over Thanksgiving break. This is a serious problem because all my schematics are

Multisim files and prevents me from working on the circuit designs. I had to contact Mrs.

Foxhall and then UNH IT to find out why that happened and how to get my license reactivated. I

was able to get my license activated again with the help of Chris Pycko from UNH IT.

Work done/word proposed

Active tasks

Right now, I am finalizing the schematic as well as starting to decide on what model of chips I want to use. I have been recommended to use the MSP430FR6989 microcontroller from Texas Instruments. I have seen the components inside E-Bike ESCs which can produce upwards of 10kW and found MCP16311 regulator IC and IRF7769 Power FET. I'm choosing to use these because I would prefer to use components that I have seen work with my own eyes in consumer applications.

Future Work

After I complete designing the whole schematic and pick components this December, I will start designing the PCB layout and get a prototype sent out to be fabricated in February. Once I get the fabricated board, I will start programming the MCU. After that it will be ready to test on the rovers. If alterations need to be made to the design, I will still have about a month to redesign and refabricate the PCB.

Discussion and Conclusion

Two constraints beyond technical

Conclusion

One of the reasons behind choosing this as my senior capstone is because it helps my understanding of motor controllers. Starting this summer and an ongoing project of mine is converting my road bike to electric. A brushless ESC is situated between the motor and the

battery with the capability to send the pack voltage at over 200 phase amps. While doing this, it performs perfect calculations of the rotor position using Field Oriented Control (FOC). The ESC has nearly limitless programable parameters using the VESC Project created by Benjamin Vedder. These include motor parameters like inductance and resistance as well as battery and phase current limits, low and high voltage cutoffs, temperature cutoffs, ERPM, sampling rates, and sensor-less algorithms like BLDC and HFI. Understanding each small part of the ESCs many functions can help me choose more capable ESCs in the future when I inevitably upgrade my bike.

Another reason is one I've had for much longer. I have had a fascination with robots and autonomy since I was young. ET NavSwarm is the perfect platform for learning about both the hardware and software design present in cutting edge robots of today. I one day hope to get a job in a field like this. The efficiency gains in autonomous functions are critical to the colonization of extraterrestrial bodies. For example, autonomy can enhance the safety and reliability of the rovers and their systems, as they can operate independently from the colonists and adapt to unforeseen situations.

References

Appendices

Timeline

Oct-Nov-Dec: Design schematic

I hope to have the completed schematics for both the controller and battery monitor by the end of the semester.

Dec-Jan: Fabrication

This stage consists of designing the traces of the board in KiCad and sending it out to be fabricated by JLCPCB.

Jan-Feb: Testing/programming

This is the time it would take to verify the designs and program the microcontroller.

Feb-Mar: Revision

If any designs don't work as intended from the testing phase, this is the time where I would revise the PCB design and/or microcontroller.

Apr: Conference preparation

This is when I would compile my results and create my poster.

May: Final report

This is when I would compile my results into one cumulative report at the end of my project.

Parts list

MSP430FR6989 Rotary Sensing MCU with extended scan interface 1x\$5.00

MCP16311 30V/1A PFM/PWM Synchronous Buck Regulator 1x\$2.00

IRF7769L1TRPBF DirectFET™ Power MOSFET 2x\$5.00

Schematics/Images

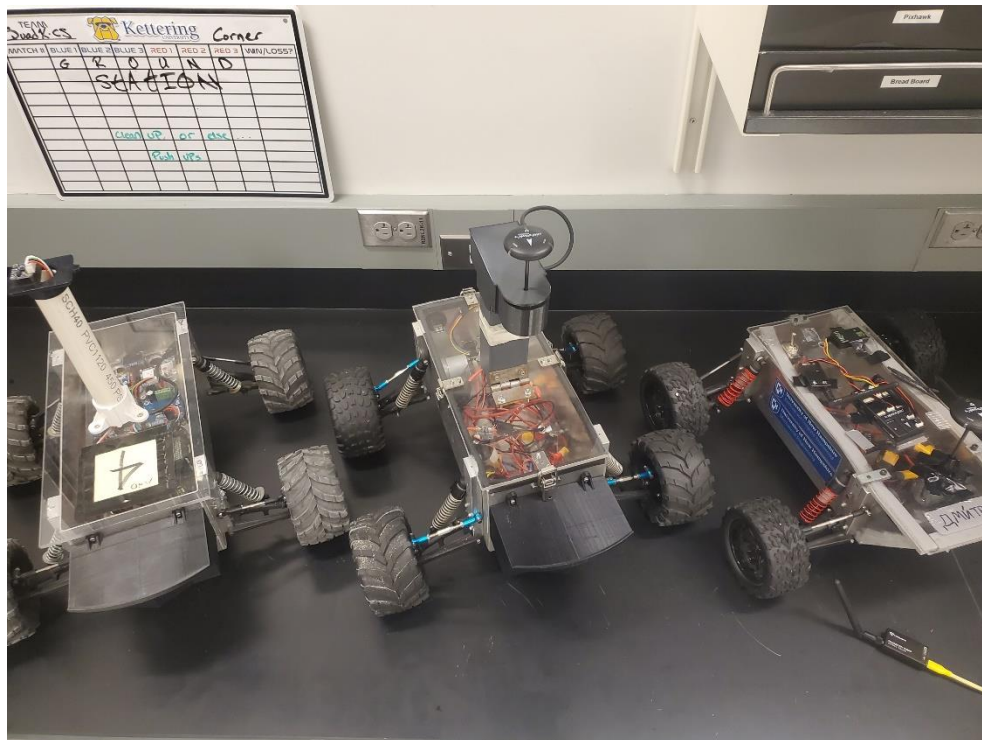


Figure 1. ET NavSwarm rovers

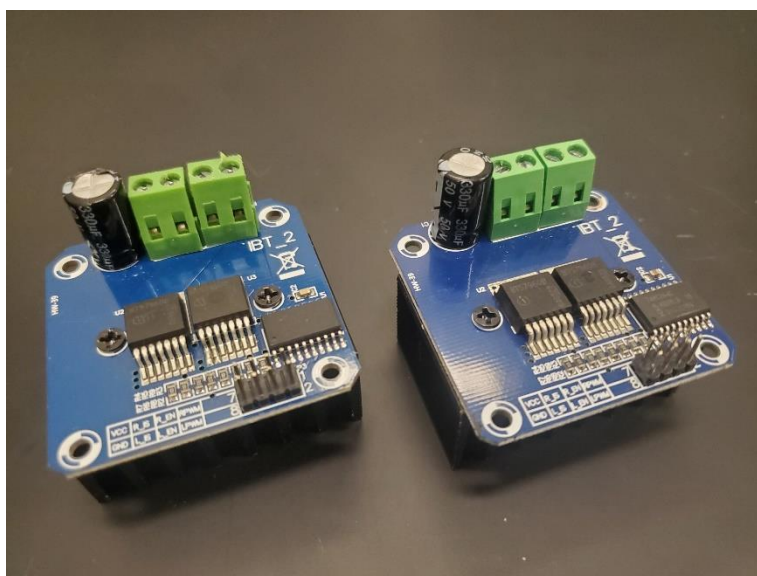


Figure 2. HW-039 H-Bridge ESCs



Figure 3. Drawer full of dead batteries

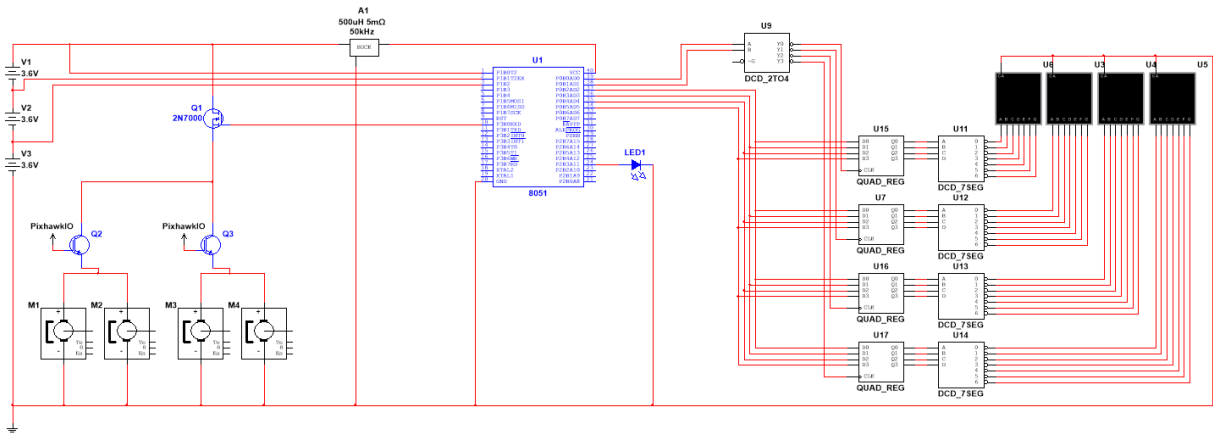


Figure 4. Whole schematic as of 12/1/23

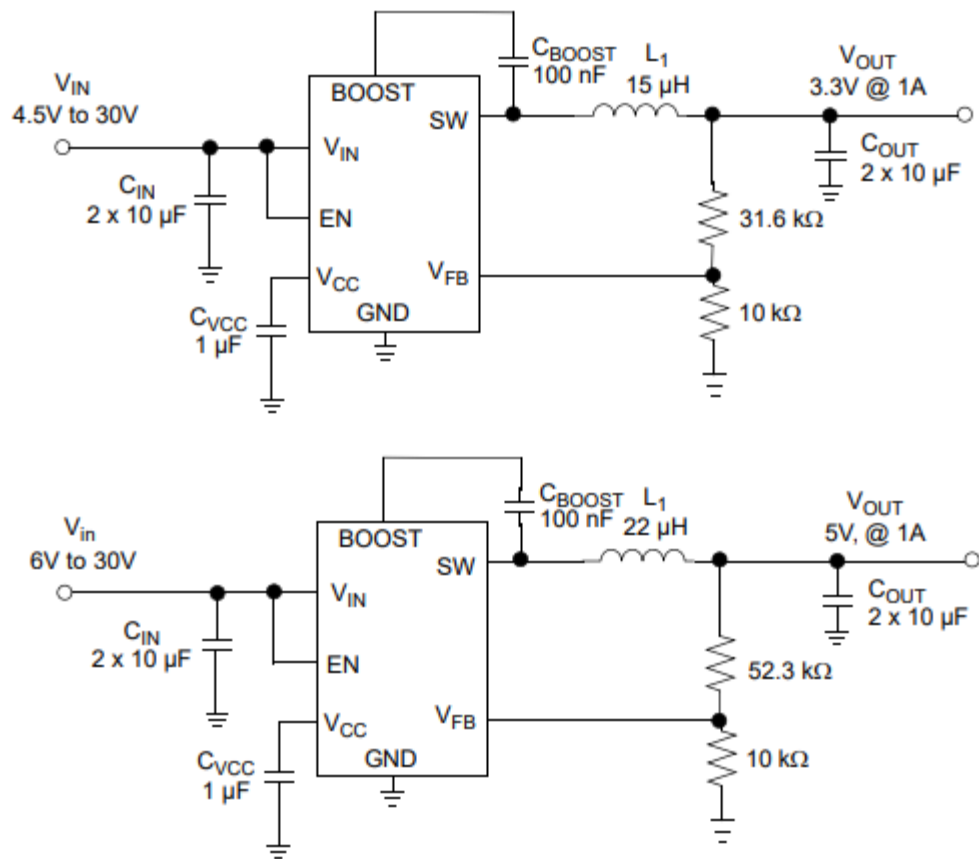


Figure 5. Typical schematics for Buck Controller IC

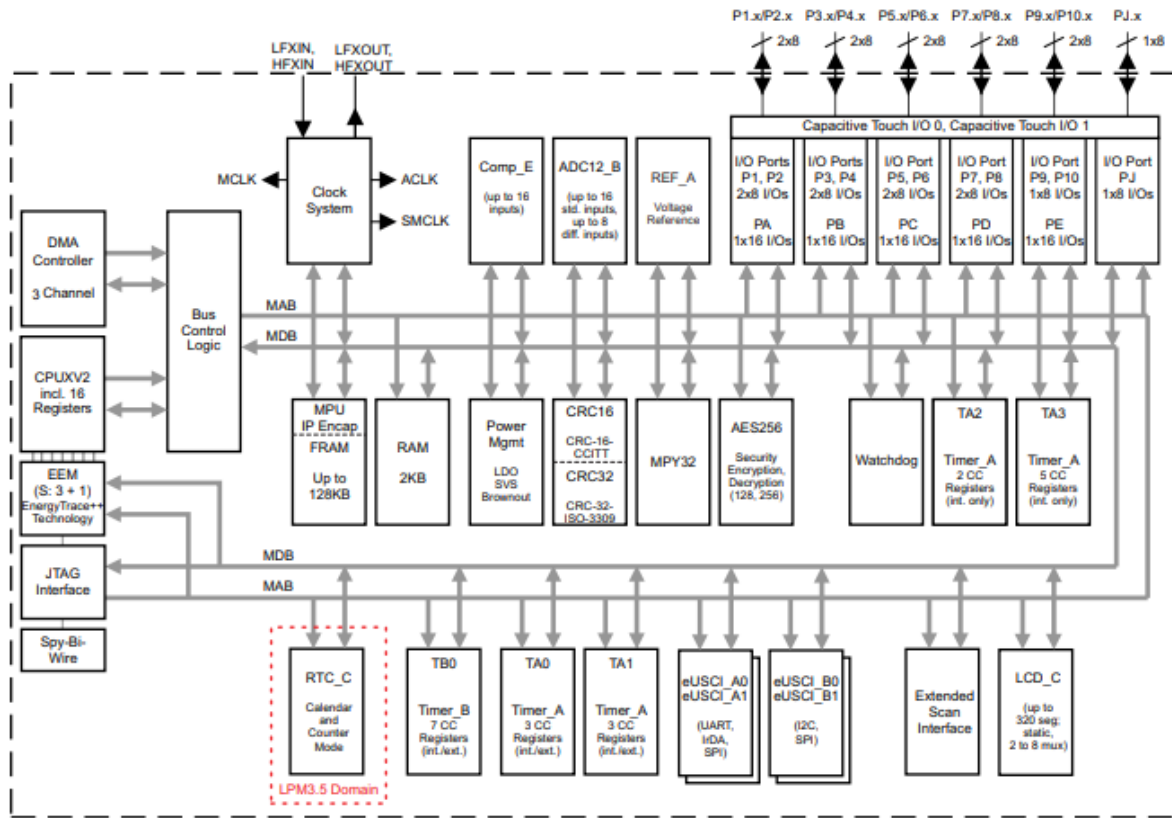


Figure 6. Functional block diagram of microcontroller

Data sheets/user manuals