Table 1: Revision History

Date	Developer(s)	Change
September 25th, 2022	Abhishek Magdum, Dharak Verma, Jason Surendran, Laura Yang, Derek Paylor	Initial document population

Development Plan
Mechatronics

Team 28, Controls Freaks
Abhishek Magdum
Dharak Verma
Jason Surendran
Laura Yang
Derek Paylor

Our team: Controls Freaks, has had a long love for the world of control systems. Our mission is to develop a control system for McMaster's Formula FSAE team, to learn and better understand how vehicle's and similar systems operate, and to cohesively work in a team to bring together the knowledge we have gained over our university careers into one final project. Our team is primarily mechatronics based, with four members (Laura, Jason, Abhishek, and Derek) being in the program, while our final member, Dharak, brings expertise from the computer engineering program and an understanding of the McMaster team's vehicle sub-systems.

1 Team Meeting Plan

Team meetings will be held once a week on Thursdays at 8:00 pm and will go for roughly one hour. Ad hoc meetings will be held as needed.

2 Team Communication Plan

A discord server has been created for team communication. This will be the primary form of communication for the team meetings as the application has strong functionalities for desktop video calls. For more general messages, a Facebook messenger group chat has been created to update each other on any ideas/issues that arise day to day.

3   Team Member Roles

Derek - Motor Controller & Vehicle Dynamics Specialist Laura - Driver Inputs Specialist Abhishek - Simulation & Validation Specialist Jason - Battery Management System Specialist Dharak - Team Lead and Inertial Sensors Specialist

4   Workflow Plan

The public Github repository will be the location where the project is. Since there are multiple subsystems that will be worked on synchronously, new branches will be created. Our convention used for branch naming will be "Subsystem Name - Major Feature Being Added". An example for this will be "Battery Management System - CAN Signals". Once a significant feature has reached sufficient progress, a pull request will be made and a different team member will review the change. If there are contentions made by the reviewer, a comment will be added to the pull request. If it has no issues it will be merged into the main branch.

To make it easier to track updates to the branches, detailed comments will also be required from each team member on all commits. The frequency of commits will not be strictly monitored but the general rule of thumb is to commit every 30 min (assuming continuous lines of code were being typed), or once a subfeature has been implemented (e.g a bug regarding device connectivity has been fixed).

Subfeature tasks will be tracked and managed on Kanban board (Github project). Every week there will be a meeting to decide on the tasks for the upcoming week (and potentially beyond), Tasks will then be assigned to the appropriate team member. Any issues or questions regarding the tasks can be added as a comment to the specific task on the board.

5   Proof of Concept Demonstration Plan

For proof of concept, we'll be able to demonstrate our vehicle control system through simulating inputs to our overarching state machine, which in turn will control our smaller subsystems and output the current vehicle state. All of this can be visualised through Matlab, using simulation tools already present. There are tools to help test and validate our control system, both through virtual models, which will be used to test our code early and often, and physical models, which we will use at the end to show off our final control system.

6   Technology

Embedded System: C Simulation: Simulink Control Model: Simulink Battery Management System Hardware: Orion BMS Motor Kit: AMK Motor Kit Microcontroller: STM32F767ZI No specific plans for a CI tool

7 Coding Standard

Github will be the version control application we will be using. This will allow us to continuously iterate, track changes, and store code.

For this project, the majority of the codebase will exist as simulink (.slx) files. Due to this, automating testing through Github Actions will be tough, since much of the code will be unrecognisable to Github. To combat this issue, we will be enforcing a PR approval process that will require a minimum of two members to approve. We will be following a 'Scouts Honor' code where each potential PR approver must pull the branch to their local machine and validate the relevant changes before approving the merge. The individual requesting the code review cannot approve their own request.

As always, proper software development practices must be implemented at all points in the software development cycle. This means that code must be properly commented, naming conventions should make sense, code is easily readable, and proper function use should be implemented to ensure unit testing can be completed easily.

8 Project Scheduling

Major milestones will align with those given in the course outline. Project-specific milestones, such as control system features, testing or integration, will be considered for each subsystem, and tasks will be assigned during weekly meetings.