Automotive Data Logger

ECE 445 Design Document – Fall 2018

Michael Jang (myjang2) & Jacob Drewniak (jdrewni2)
Team 10

TA: Evan Widloski

1 Introduction

1. Objectives and Background

The primary objective of the electronics subsystem of an FSAE team is collecting useful data in support of the design and validation process. This data can range from lateral and longitudinal acceleration to ignition timing to suspension strain values and is absolutely essential for a finely-tuned race car. Commercial data acquisition systems are extremely expensive and often times use difficult, slow, and proprietary software packages that don't properly parse or format data. The team's current data logging module is made by Vector but is unideal for several reasons. The software required to download the data from this logger is difficult to use and quite slow. Additionally, the files it produces have a maximum size of 5.25 MB and are not parsed correctly for easy data extraction. Post processing is necessary to produce usable data. Finally, data is often completely lost or, at the very least, corrupted if the power supply is severed during downloads. Replacing this data logger with a different commercially available option is not within the team's budget, so the best solution is to build our own with a simpler UI, better extraction methods, and faster speeds.

The goal of this project is to design a data logger that can save all incoming sensor data at a high frequency and then make it quickly and easily available to the user through a simple, streamlined interface. This ideally can be done through either USB or Bluetooth connectivity depending on the scenario and testing session length. The team already employs a custom-built sensor processor that converts analog and digital sensor data into CAN format, so the data logger must simply log incoming CAN messages from the sensor processor [1]. This simplifies the overall architecture to merely storing and transmitting the data to team members at the necessary times with several protection measures built in to prevent data corruption or losses.

1.2 High-Level Requirements

- The system must be able to store all the data from incoming CAN messages into non-volatile memory.
- The data stored in the SD card must be able to be transferred through USB and Bluetooth.
- In case the system loses battery power, it must be able to power itself for long enough to properly power down and ensure that no data is lost or corrupted in the process.

2 Design

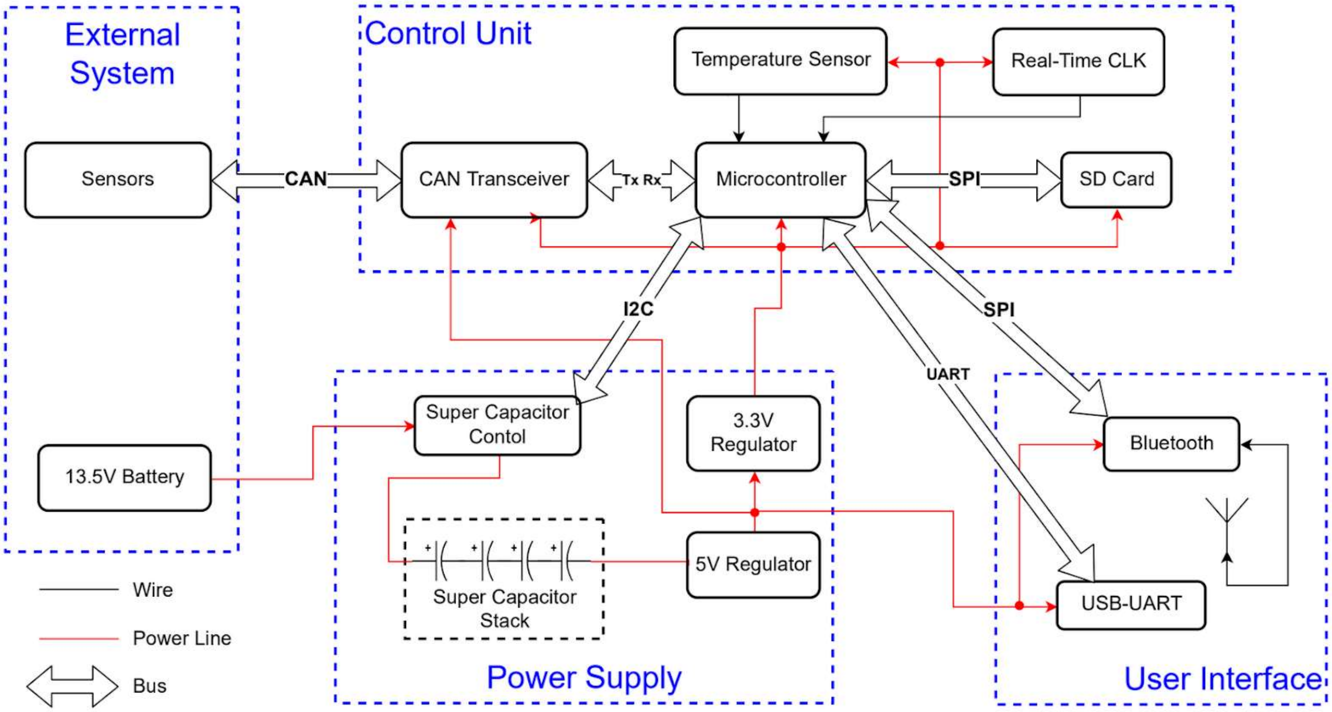

Figure 2.1 Block Diagram

This design has three main components: power supply, control unit, and user interface. The power supply module will drive the entire circuit with the proper voltages (3.3V and 5V). There will also be a feature in which we can temporarily power the entire system in the case that electrical power to the car is lost. The control unit has the microcontroller at the heart of it, and it is mainly responsible for all the data transfer and data storage. Finally, the user interface will allow the team to be able to read the logged data through different means: USB and Bluetooth.

2.1 External System

The external system is the vehicle that the logger will be receiving data from. This design only needs two things from the vehicle: a battery and sensors that communicate through CAN.

2.1.1 13.5V Battery

The car and all of its electrical components, including the proposed data logger, are powered by a lithium-ion battery that is nominally at 12V. Normal operating conditions usually range from 13V-14V. The battery can be charged using a lithium-ion battery charger when the car is off or by the car's alternator while the engine is running.

2.1.2 Sensors

Our car has a large number of sensors that all communicate through CAN. These are the data points that the circuit will be logging. Sensors give the team important data points such as lateral acceleration, battery voltage, large current draws (water pump, radiator fan, etc.), steering angle, wheel speed, etc. All these data points are essential to log in order to calibrate/tune the car, validate mechanical and electrical designs, and make improvements for future designs.

2.2 Power Supply

The power supply subsystem will provide the necessary voltages throughout the board using two regulators with enough current capacity to properly power all circuitry. The main power supply will be provided by the car battery. There will also be a super capacitor stack as a backup powering method in the scenario that battery power is lost.

2.2.2 Super Capacitor Stack

A stack of multiple super capacitors will be used as a power backup for the board if the battery supply is cut off. This is necessary so that the system can properly power down without losing or corrupting any essential data that has been logged. The pack voltage must be close to the supply battery voltage of 12V-14V to power the board for about 2 seconds to ensure that no data is being transmitted or stored when the backup loses charge. Because single supercapacitors can rarely exceed between 2.7V and 3V because of dielectric breakdown at higher voltages, a series stack of four is necessary. The buck converter that supplies power to the rest of the board has a minimum input voltage of 6.5V. Because the current consumption of the board is a relatively constant 500 mA, the capacitance of the stack can be calculated as follows:

Eq. 1)
$$I = C\frac{dV}{dt} = -500mA$$

$$\frac{dV}{dt} = \frac{6.5 - 12}{2} = -2.75 = constant$$

$$C = 0.18F$$

Because the stack has capacitors in series, individual capacitance can be calculated as

Eq. 2)
$$\frac{4}{C_{individual}} = \frac{1}{C_{stack}} \rightarrow C_{individual} = 0.72F$$

Accounting for a safety factor of about 5, each capacitor should be 5+ times larger than calculated, leaving individual capacitance of around 4F.

Requirements	Verification
1. Pack voltage of 10.8V-12V to sufficiently power the board if power is lost **2.** Power the 5V Regulator for 2 seconds at a max current consumption of 500mA **3.** Pack capacitance must be at or above 1F	Use multimeter to test pack and individual voltages. Measure capacitance by measuring voltage on an oscilloscope over time at constant current.

2.2.3 Super Capacitor Control

This IC will be connected to all four super capacitors and will function as a battery management system. It will function as both a monitor and a controller. The monitor functionality will include measuring state of charge, voltage level, and current I/O. The controller aspects of this IC will include cell balancing, pack protection, and microcontroller interfacing [2]. The LTC 3350 supercapacitor backup controller will be used to perform this functionality. It controls a stack of up to 4 supercapacitors. Controller functionality includes performing constant current/constant voltage charging of the supercapacitor stack, monitoring stack currents and voltages with a 14-bit delta-sigma ADC, performing internal cell balancing of up to 10mA with 10mV resolution, programmable overage and underage protection, and input power failure detection. Programmable protection and charging parameters as well as ADC measurements are communicated with the microcontroller over an I^2C bus [3].

Requirements	Verification
1. Programmable overvoltage, overcurrent, and undervoltage protection **2.** Pack voltage and current monitoring with at least 12-bit precision **3.** Cell balancing current of 10-20mA **4.** Input power detection of 13.5V **5.** Must monitor 4 supercapacitors	Compare multimeter measurements with I^2C microcontroller outputs Unplug power supply and measure current discharge Measure pack differential voltage

2.2.4 5V Regulator

A buck converter will be used to step down the 13.5V battery supply to a 5V voltage rail that powers several IC components on the board. A buck converter was chosen because of the wide input voltage range and higher efficiency at large voltage drops from supply to output [4]. This prevents large amounts of power losses and heat dissipation into the board that could cause improper functionality. This 5V regulator will directly power the 3.3V regulator, CAN Transceiver, Bluetooth, and USB-UART modules. The regulator will typically output about 300mA during normal operation. Accounting for worst case current draw and a safety factor, the 5V regulator must output a maximum of 500mA.

Requirements	Verification
1. Output voltage of 5.0V +/- 5% **2.** Must be able to supply at least 500mA	The LT2675 has current output rating of 1A and 1.5% output voltage tolerance [5]. To test, we will measure the tolerance on the output voltage with an oscilloscope while varying the load and input voltage.

2.2.5 3.3V Regulator

As mentioned above, this 3.3V supply will be originated from the 5V regulator which will then power most of components throughout the system. Unlike the 5V regulator, this regulator is a Low-Dropout (LDO) Linear regulator. Because the drop from 5V to 3.3V is fairly small, the efficiency of the linear regulator is reasonable and, therefore, it doesn't dissipate much power. Although the efficiency is at a reasonable level, power dissipation from a linear regulator should never be ignored. The power dissipation can be given as:

Eq. 3) $$P_D = (V_{IN} - V_{OUT}) * I_{OUT}$$

Even though power dissipation must be considered, a linear regulator is still beneficial due to its simplicity, size, and cleaner outputs when compared to a switching regulator [6]. As given in the TPS73633 datasheet [7], the output noise equation is given to ensure output voltage tolerances are met:

Eq. 4) $$V_N(\mu V_{RMS}) = 8.5 * V_{OUT}$$

Since the 3.3V regulator powers about two thirds of the components, the necessary max output current can be determined to be 350mA derived from the 500mA requirement of the 5V regulator. From our tolerances on both 3.3V and 5V regulators, the minimum dropout voltage for the linear regulator is given as:

Eq. 5) $$(5 * 0.95) - (3.3 * 1.05) = 1.285V$$

Requirements	Verification
1. Output Voltage of 3.3V +/- 5% **2.** Must be able to supply at least 350mA **3.** Minimum drop out voltage of 1.2V (Eq. 5) **4.** Temperature maintains below 80C	From Eq. 4, the output noise is calculated to be 28Vμ_{RMS}. This is well within the 5% tolerance. The TPS73633 has rated output current of 400mA [7]. Testing with an oscilloscope and DC power supply, the output voltage/current will be monitored while varying loads and input voltage to ensure output requirements and dropout voltage requirements are met. The temperature will also be monitored using an IR thermometer to see how the 0.68W dissipated (Eq. 3) affects the temperature.

2.3 Control Unit

The control unit for this system manages the incoming CAN messages and stores the data into an SD card through the use of a microcontroller and CAN transceiver. When storing data into the SD card, the real-time clock will let the microcontroller what time it is so that when pulling the data, we know what test run the data correlates to. The microcontroller will then transfer the data through Bluetooth and USB for user communication.

2.3.1 Microcontroller

The PIC32 microcontroller (PIC32MZ2048EFM100) will communicate with the CAN transceiver [8]. The microcontroller will then store that data into an SD card through SPI communication. The microcontroller will send the data in the SD card to the Bluetooth IC through a separate SPI bus and the USB-UART IC through a UART bus. The microcontroller is also responsible for communicating with the super capacitor controller to set protection limits and to monitor and make decisions if power loss or overvoltage occurs [3]. This microcontroller has been historically used on all the team's boards, so several libraries have been built, so the requirements has already been determined and satisfied.

Requirements	Verification
1. Flash memory storage of at least 250KB **2.** Includes programmable analog and digital I/O with CAN, SPI, I^2C, and parallel communications protocol functionality **3.** Internal ADC **4.** External oscillator compatibility	Based on previous libraries built by the team, 250KB is a very conservative maximum for flash memory storage. In addition, the PIC32 has 2MB of flash storage [8]. Requirements 2-4 have been made by the team in previous years, and the PIC32 has been chosen because it meets these requirements [8].

2.3.2 CAN Transceiver

The CAN transceiver is a necessary component because the PIC32 does not have any compatibility with CAN's 5V differential signals. The MCP2562 will read all the high-powered CAN messages coming from throughout the car via our CAN bus wire harness (Figure 2.2) and produce low-powered 3.3V Tx/Rx signals that are readable by the microcontroller [9]. Since the CAN transceiver is directly connected to an outside wire harness, the IC must have ESD protection. Our main ESD concern stems from inductive spikes from long wires. ESD is a very difficult parameter to predict, but an ESD rating of +/- 2kV is a safe number to expect from our wire harness.

Requirements	Verification
1. Accurately convert 5V differential CAN signals from other boards and sensors throughout the car to 3.3V single signals that can bidirectionally communicate with the microcontroller **2.** ESD protection of at least +/- 2kV on CAN differential signals	CANalyzer is a software that the team uses that can track real time CAN data throughout the car while a laptop is plugged in. We can test the CAN messages that the transceiver produces and compare them to the real values that are read through CANalyzer. Twisted pair wires with shielding are used in the CAN bus wiring. This will greatly reduce the inductance in the wires to minimize ESD, and the IC has a rated +/- 8kV ESD protection rating [9].

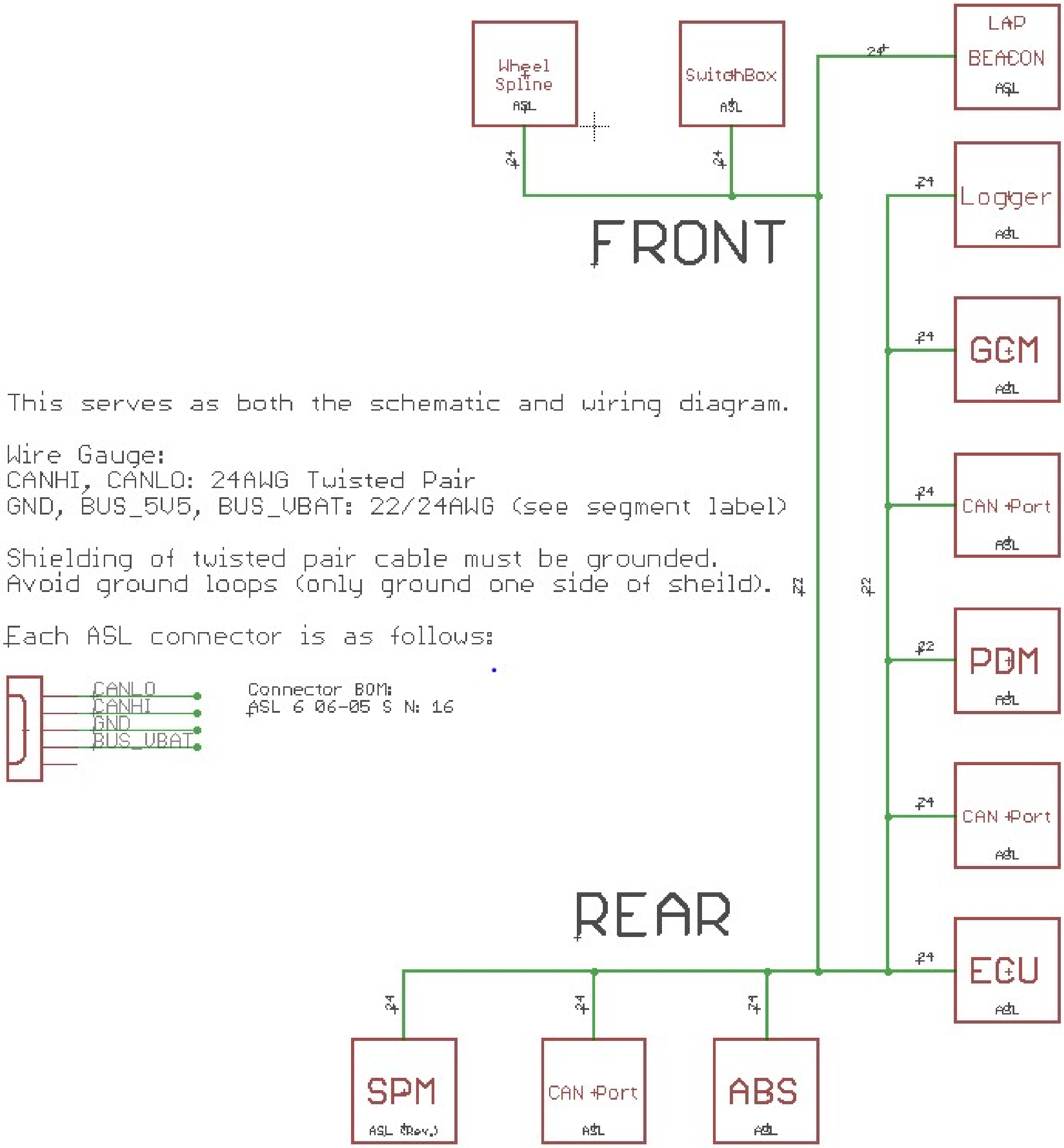

Figure 2.2 CAN Harness Schematic w/ Pin-Out

(Modules roughly oriented to car placement)

2.3.3 SD Card

Some form of non-volatile memory (either a flash chip, SD card, or potentially some other form of NVM) is necessary to hold all the data that the microcontroller and CAN transceiver process. For our system, an SD card was chosen due to its storage capabilities and size/weight. The SD card will communicate back to the microcontroller so that the user can read the data through Bluetooth or USB. Our car runs a 1 Mbps CAN bus. Assuming 8-hour testing sessions (3600 seconds) without pulling data, the SD card must have minimum storage of 3.6 GB with a data transfer rate of at least 1 Mbps.

Requirements	Verification
1. Minimum Storage of 3.6GB **2.** Minimum data transfer rate of 1Mbps (but preferably much faster).	Standard SD card sizes calls for a 4GB storage which give us even more buffer for data storage. We will log data (car standing still in the garage so that useful data isn't lost) for a total of 8 hours, to see how much data is stored into the SD card.

2.3.4 Temperature Sensor

The only sensor in this circuit is the temperature sensor. This will communicate with the microcontroller for circuit diagnostic purposes. The logger will likely be placed in a high-temperature environment near the fuel tank of the car, and this sensor will relay the temperature of the board to ensure that all chips are still within their temperature operating range. Since the board will be placed near high temperature environments, a max temperature of 80C must be met. Also, driving conditions can get as low as -10C, so that will be the lower bound for the temperature rating.

Requirements	Verification
1. Must have temperature read range of -10C-80C	The MAX6610 specs for a -45C to 125C range of operation which is a much wider bound than necessary [10]. This part has historically been used on the team's boards for years, and functionality has been verified over the years.

2.3.5 Real-Time Clock

The real-time clock lets the microcontroller know what the date and the time is so that the data logged can be associated with that time/test run.

Requirements	Verification
1. Must log time accurately to less than a second	The chosen IC (MCP79522) features time logging to the hundredth of seconds [11]. Testing on the logged times can be displayed and compared to today's standard of 'true time', our phones.

2.4 User Interface

The user interface module will make it possible for any user to read the data that has been stored in NVM. There are two methods to communicate the data: USB and Bluetooth. The Bluetooth will be useful for when smaller data sets are desired between test runs. The USB interface will be used for transferring large sets of data from longer testing sessions because the Bluetooth transfer rate is much slower.

2.4.1 USB-UART

This IC will communicate with the microcontroller through UART to transfer data stored in the SD card. The IC will then convert these messages to the standard USB protocol so that the data can easily be transferred to a laptop with a simple cable.

Requirements	Verification
1. Minimum data transfer rate of 2MBps	The MCP79522 has a max data transfer rate of 3MBps [12]. The time it takes to upload the data from the SD card through USB will be documented and determined if the upload time is reasonable.

2.4.2 Bluetooth IC

The Bluetooth IC will receive 2 Mbps SPI signals containing relevant data from the microcontroller (originating from the NVM) and convert them to the necessary signals for the antenna to be able to transmit the data wirelessly to an external device. The Nordic Semiconductor NRF51822 was chosen as the Bluetooth IC for this system because it meets all the requirements and has a plethora of additional capabilities and GPIO pins that can be used if necessary such as a temperature sensor, 10-bit ADC, and ARM processor. It even has a low-powered off mode that can be used to reduce power consumption while data is not being transmitted. [13] Nordic also provides a software development tools for its Bluetooth devices that facilitate implementation of the BLE stack firmware. [14]

Requirements	Verification
1. Minimum data transfer rate of 2 Mbps and a power of >= 4 dBm **2.** Must support Bluetooth v4.0 or newer **3.** Must act as an SPI slave device **4.** Must operate at a frequency at or below that of the microcontroller clock frequency **5.** Must be able to send data packets greater than 12 bytes	Connect to another device. Send known amount of data and measure transfer time.

2.4.3 Antenna and Impedance Matching

The antenna will transmit logged data from the SD card to the user through Bluetooth. It must transmit in the 2.4 GHz to 2.484 GHz band with a signal strength capable of being recognized at a range of 10m. An inverted-F quarter wave trace antenna will be used to minimize power loss and facilitate impedance matching. [15] It will be connected to the main PCB by a coaxial cable. The impedance matching scheme must match the optimal input impedance of the Bluetooth IC with that of the antenna.

Requirements	Verification
1. Antenna must have a 50Ω 5% input impedance in the operational frequency range of 2.4GHz - 2.484GHz	Attach the antenna to a network analyzer and measure reflection coefficient at the antenna over the bandwidth range.
2. Matching network must produce an input impedance of 15Ω + j85Ω 5% at Bluetooth IC	Attach the Bluetooth IC to a network analyzer and measure reflection coefficient. Back calculate input impedances from reflection coefficients.
3. Must produce a signal strength of approximately -90 dBm at a distance of 10m	Measure s-parameters of network. Calculate power degradation with distance.
4. Antenna must radiate omnidirectionally parallel with the ground and losses of no more than 4 dB at a 45 degree zenith angle	Calculate power losses based on measured parameter values.
5. Antenna and matching network must have losses of less than 1dB for the entire bandwidth	

2.5 Software

Because there is a lot of data processing in this design, it is inherent that there is a large software portion to the system. All of the software and brains of the system lies within the microcontroller. The developed code will be stored into the microcontrollers 2MB of flash memory [8]. From a very high level, the microcontroller has to parse through all the CAN messages and format them in a convenient and readable way. The team has a CAN spec that has allocated a list of IDs. Within each ID, there are multiple data points because each data point is typically two bytes. Each byte in the ID is also assigned to a specific data point that is relevant to the respective ID. Figure 2.3 shows an example of the structure of one of our CAN Spec IDs (Motec is our engine control unit):

Origin <Destination>	ID	DLC	Byte Order	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Motec	0x100	8	MSB	RPM (rpm) Signed - 1.0 - 0x0		Throttle Position (%) Signed - 0.1 - 0x0		Lambda (N/A) Signed - 0.001 - 0x0		ECU Battery Voltage (V) Signed - 0.01 - 0x0	

Figure 2.3 Example CAN Spec ID

To begin a more detailed description on the data processing: once the car is powered on, and the board is also powered, there will be a constant check to see if the battery voltage and capacitor voltages are at acceptable levels. This signal will originate from the supercapacitor controller and be sent over an I^2C bus [3]. If the voltages aren't good, the system will go into a power down mode. If the voltages are good, then there will be a check to see if the engine is on because the only relevant data that we want to log is when the car is actually running. This data can be checked by parsing through ID 0x100, Byte [1:0]. This data point is associated with the engine RPM as seen in Figure 2.3. If RPM is nonzero, then we know the engine is running. If the engine is on, the process of receiving, parsing, and storing the CAN messages begins. If the engine is off, the microcontroller decides which method to transmit the data by reading the memory from the SD card and sending the data to the correct IC. When either data storage or data transmission are done, it loops back to check if the battery voltage is good in order to repeat the process.

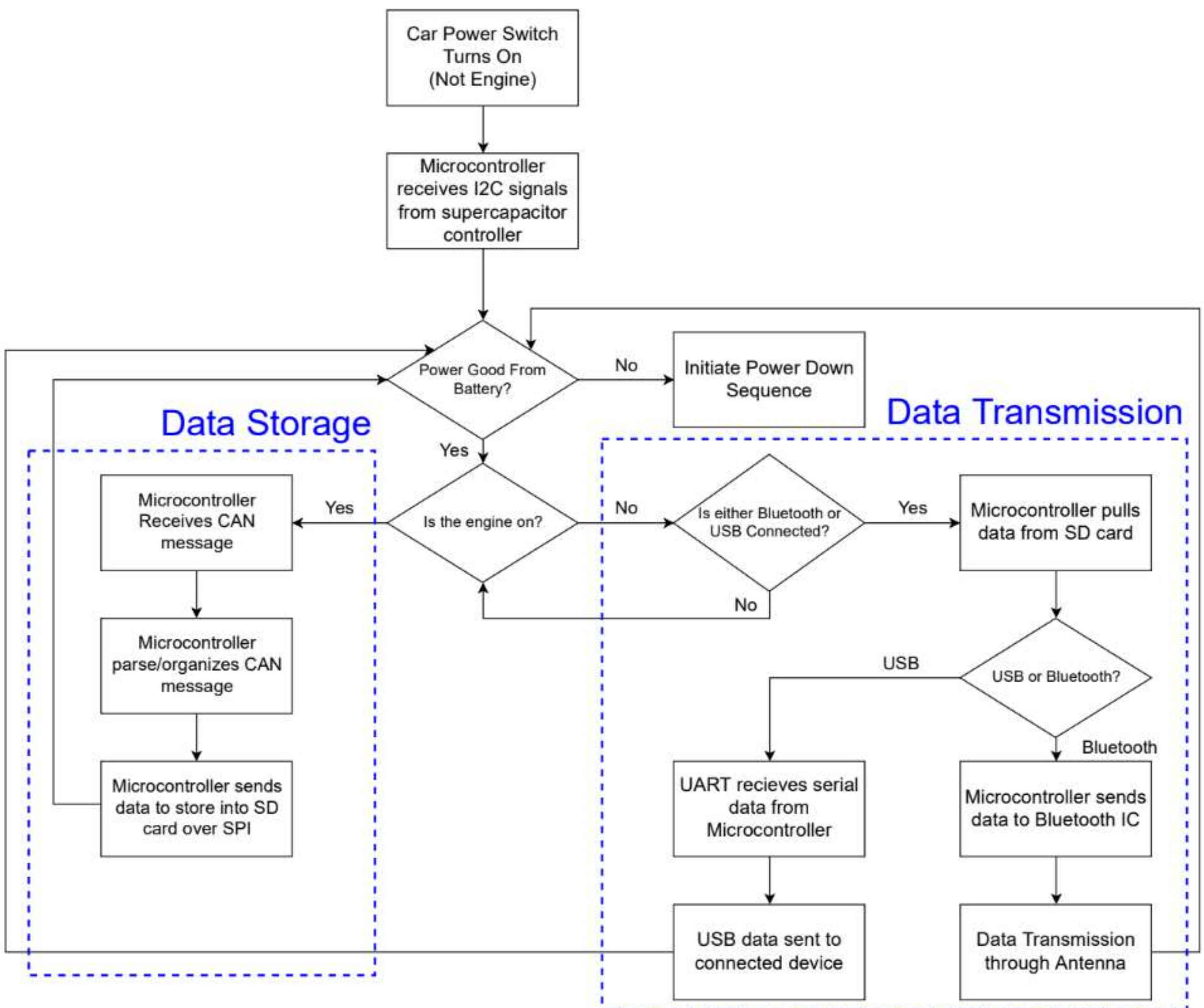

Figure 2.4 Data Flow Chart

2.6 Circuit Schematics

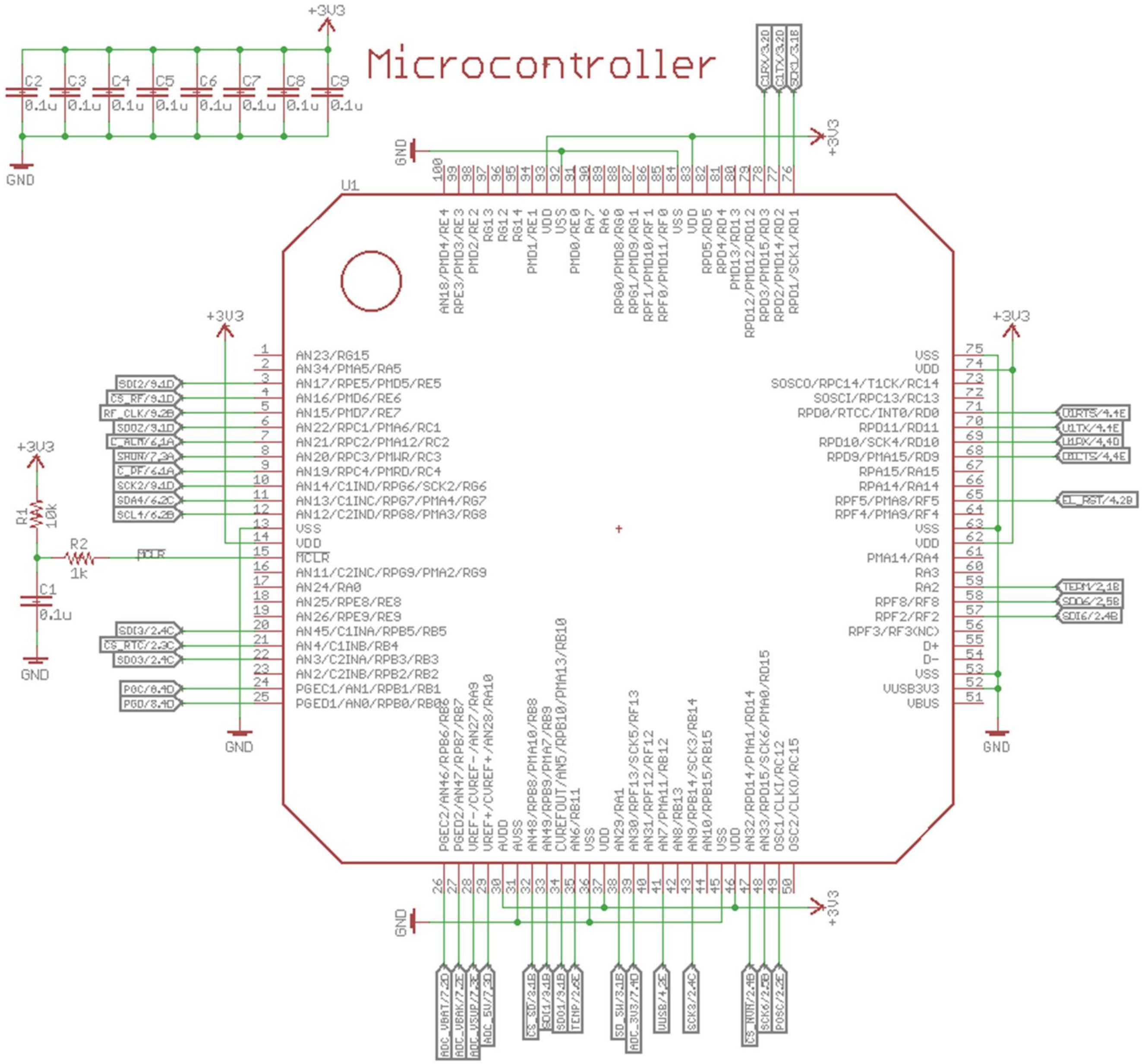

Figure 2.5 Microcontroller Schematic

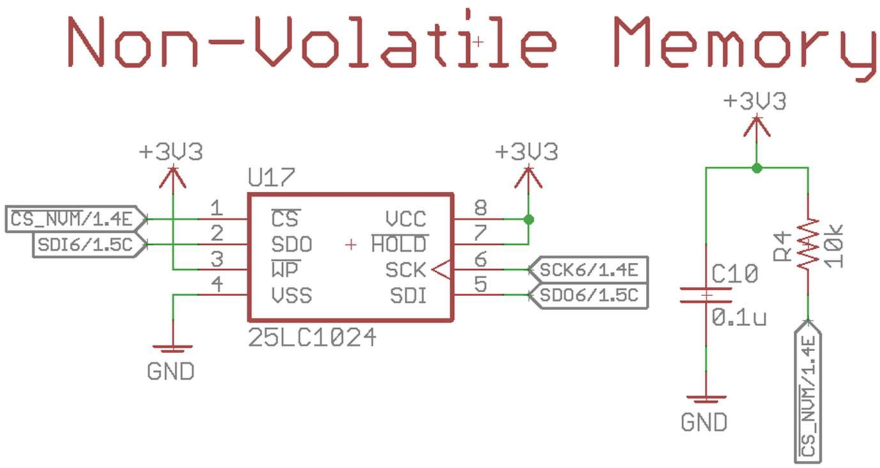

Figure 2.6 Non-Volatile Memory Schematic

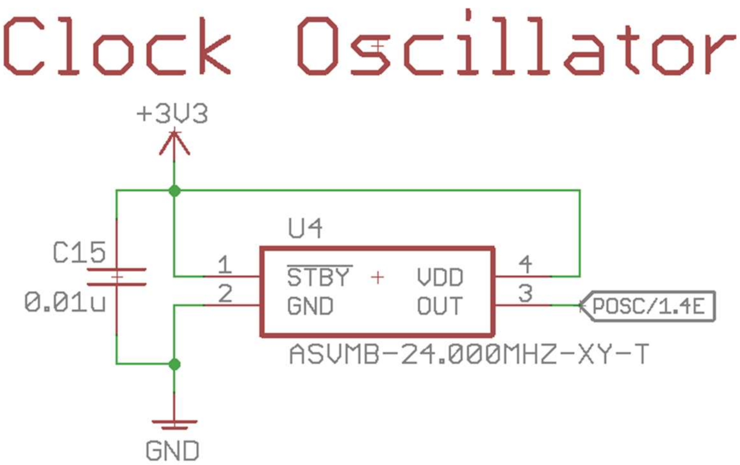

Figure 2.7 Clock Oscillator Schematic

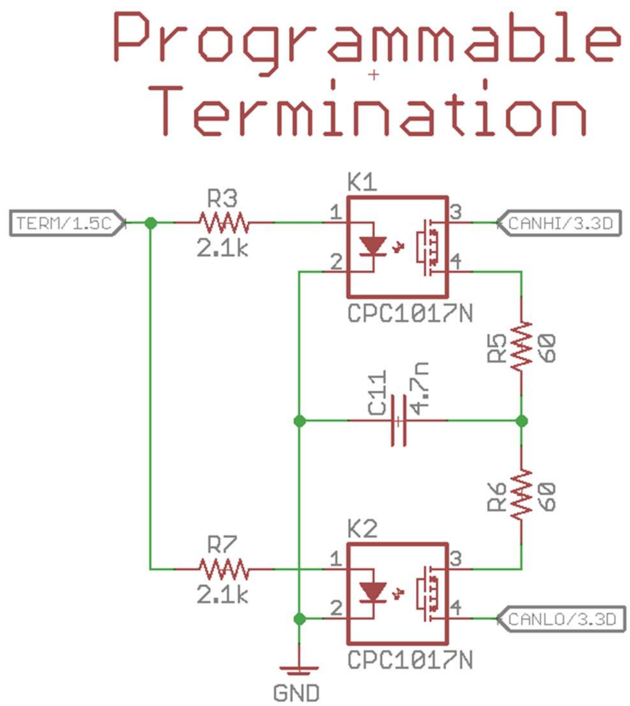

Figure 2.8 Programmable Termination Schematic

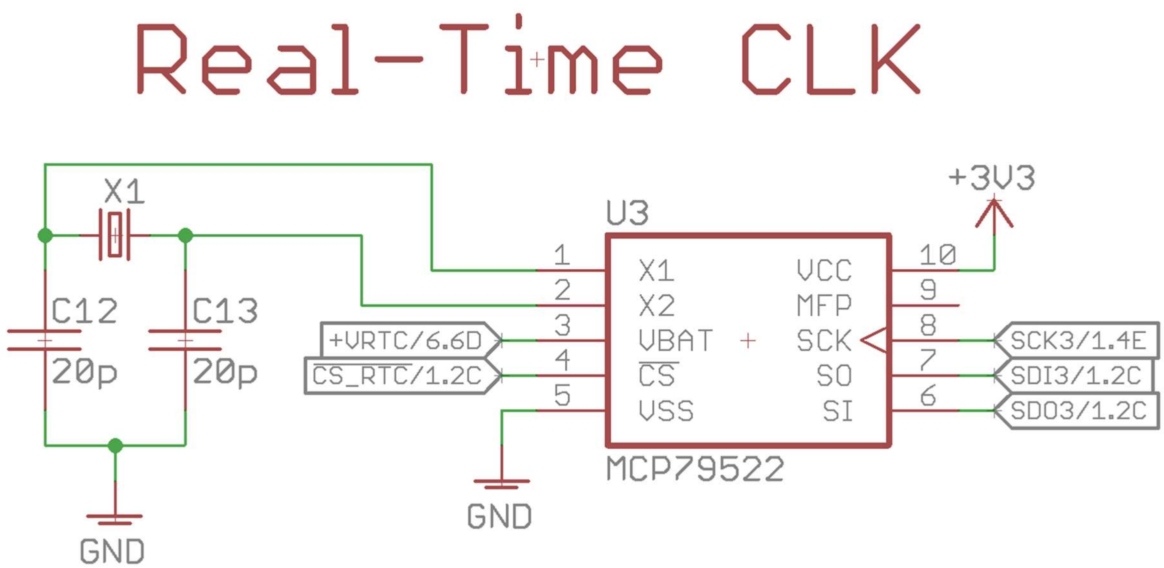

Figure 2.9 Real Time Clock Schematic

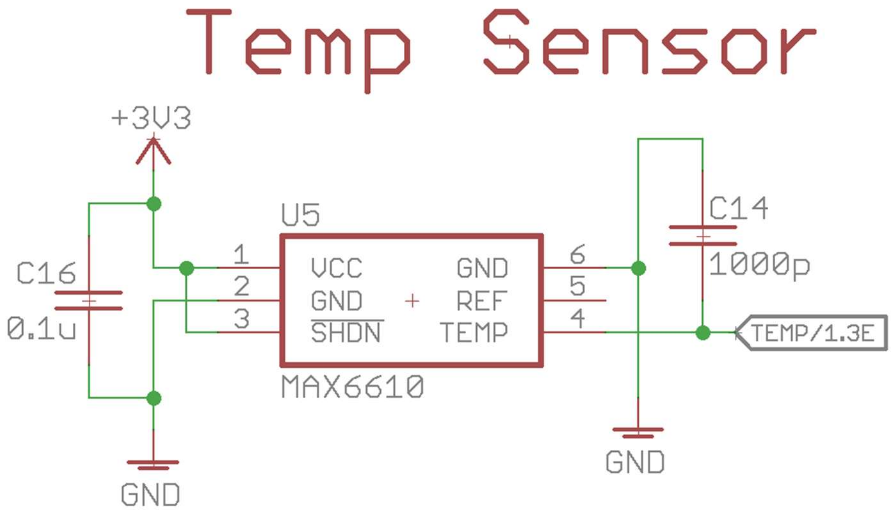

Figure 2.10 Temperature Sensor Schematic

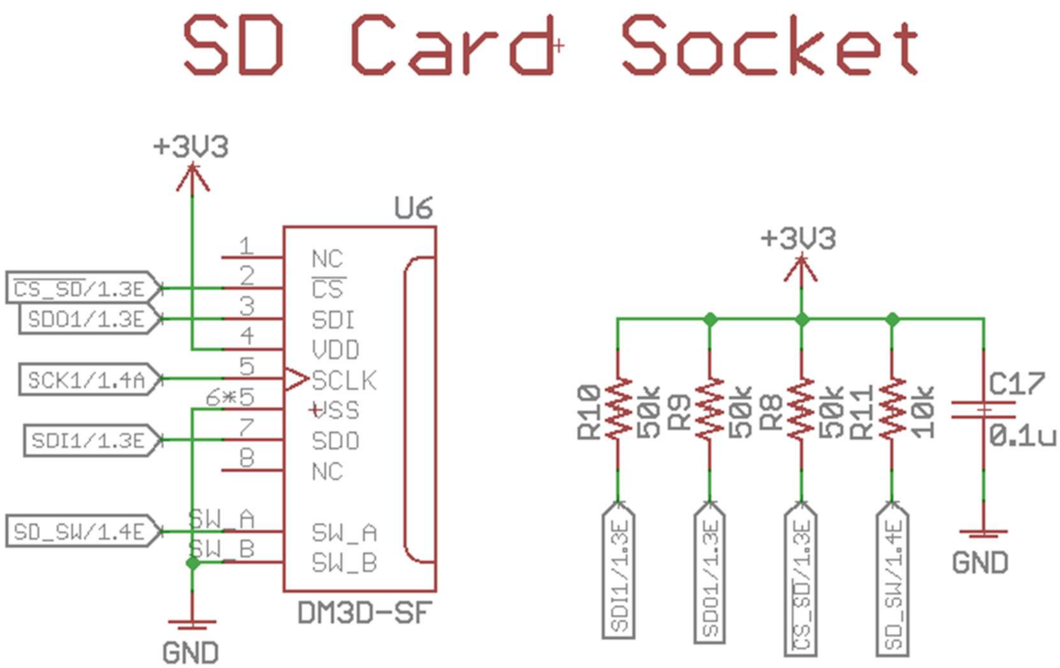

Figure 2.11 SD Card Socket Schematic

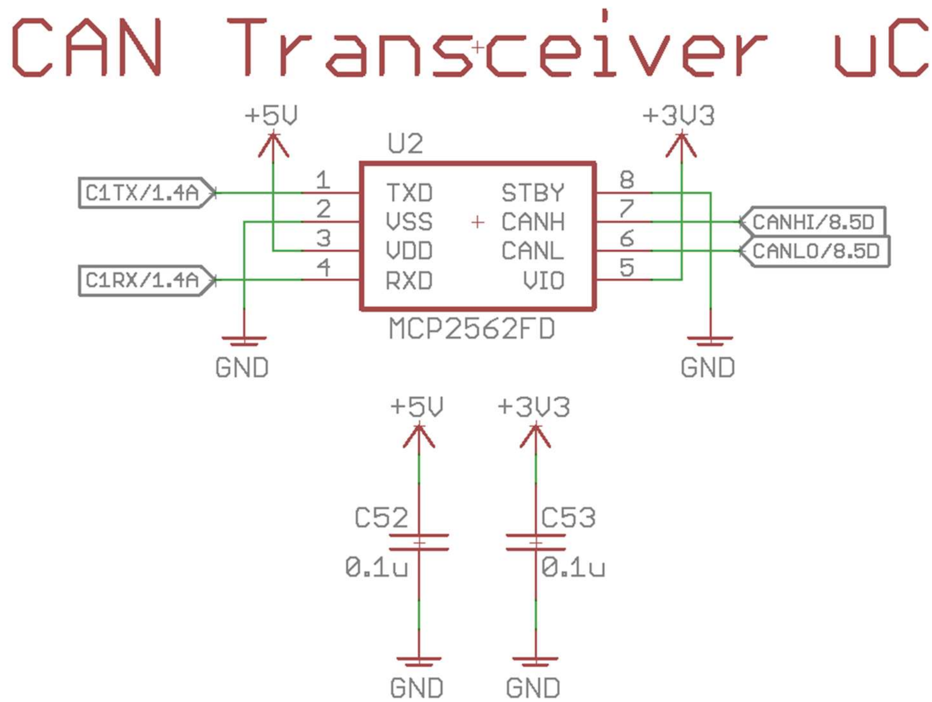

Figure 2.12 CAN Transceiver Schematic

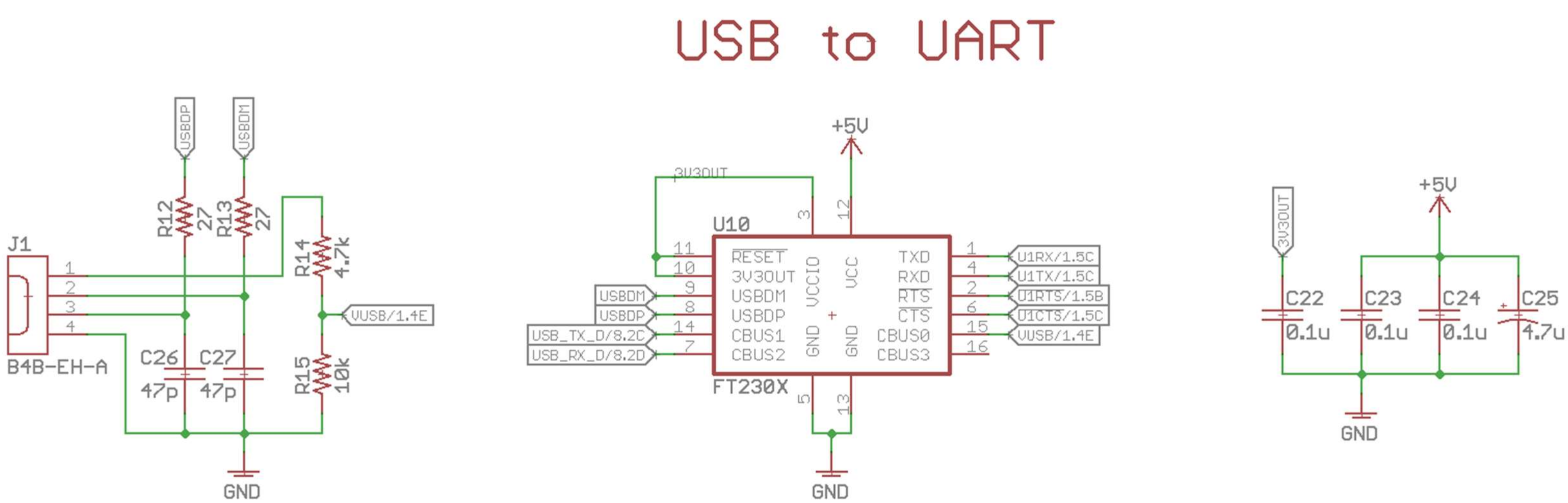

Figure 2.13 USB-UART Schematic

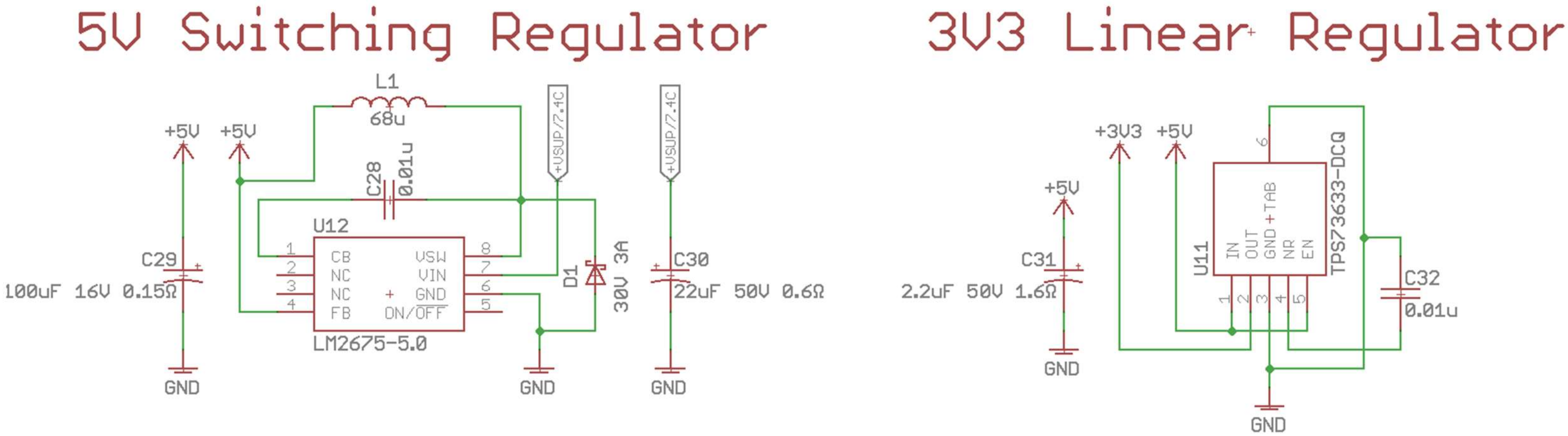

Figure 2.14 Regulator Schematic

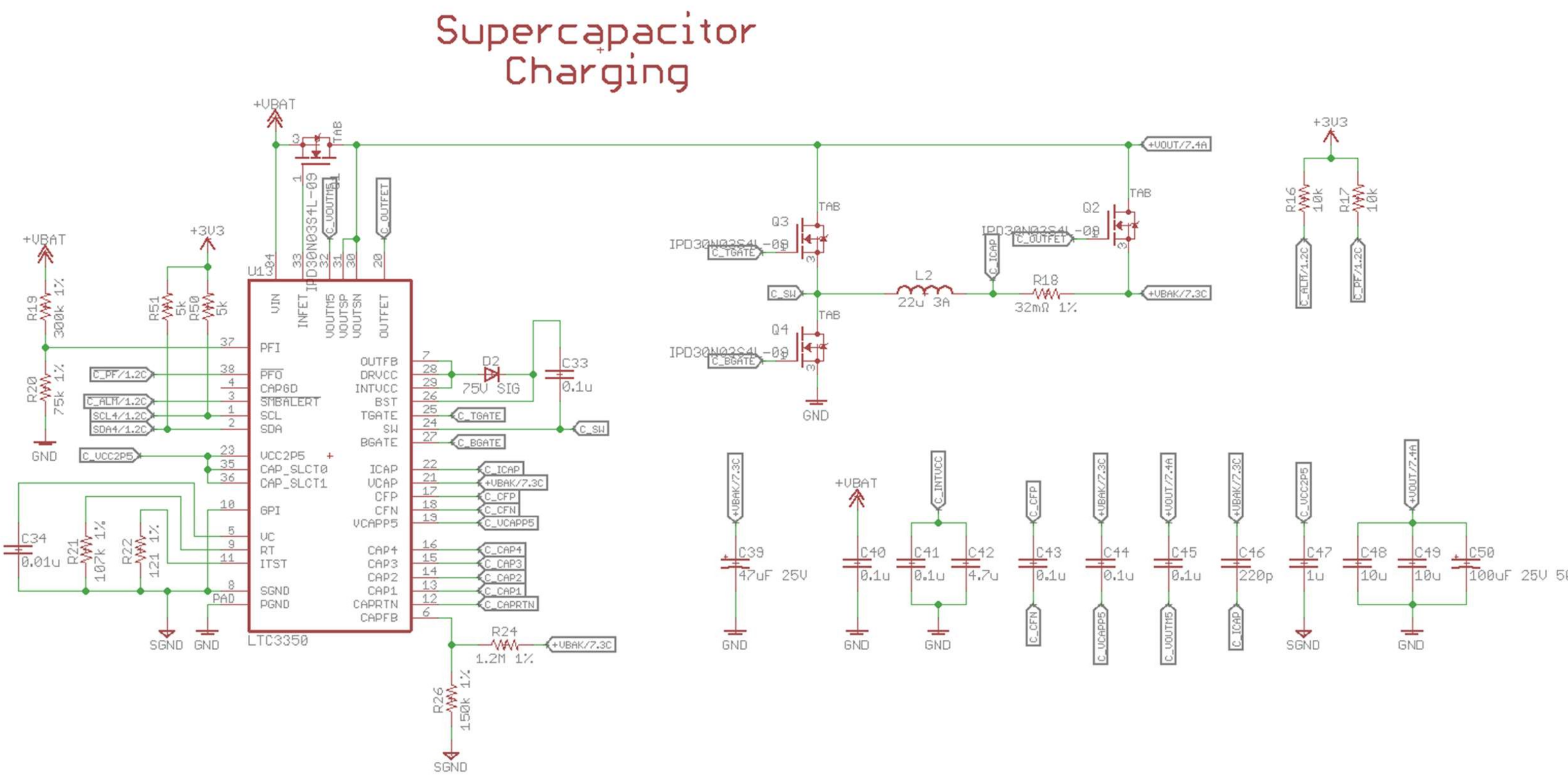

Figure 2.15 Supercapacitor Controller Schematic

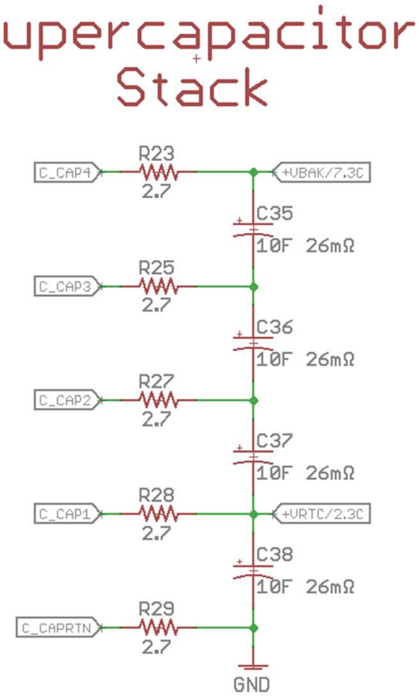

Figure 2.16 Supercapacitor Stack Schematic

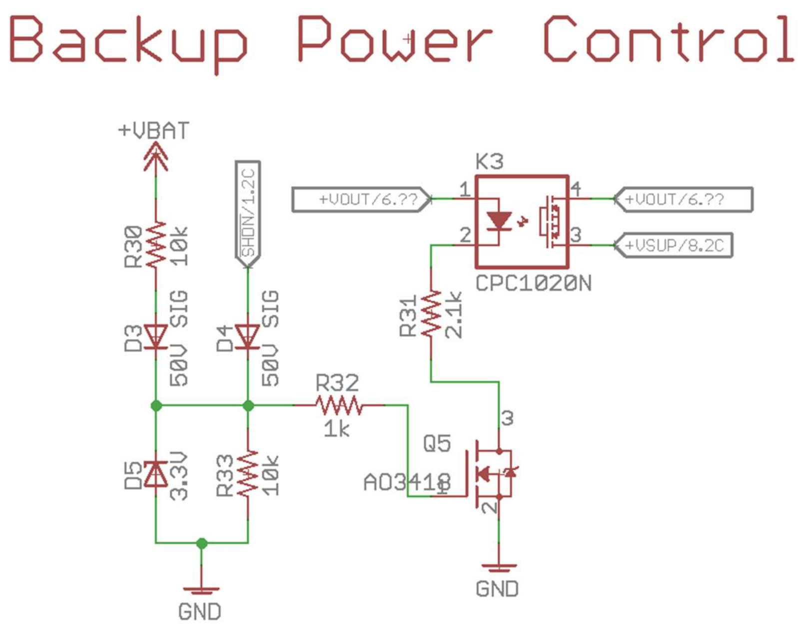

Figure 2.17 Backup Power Control Schematic

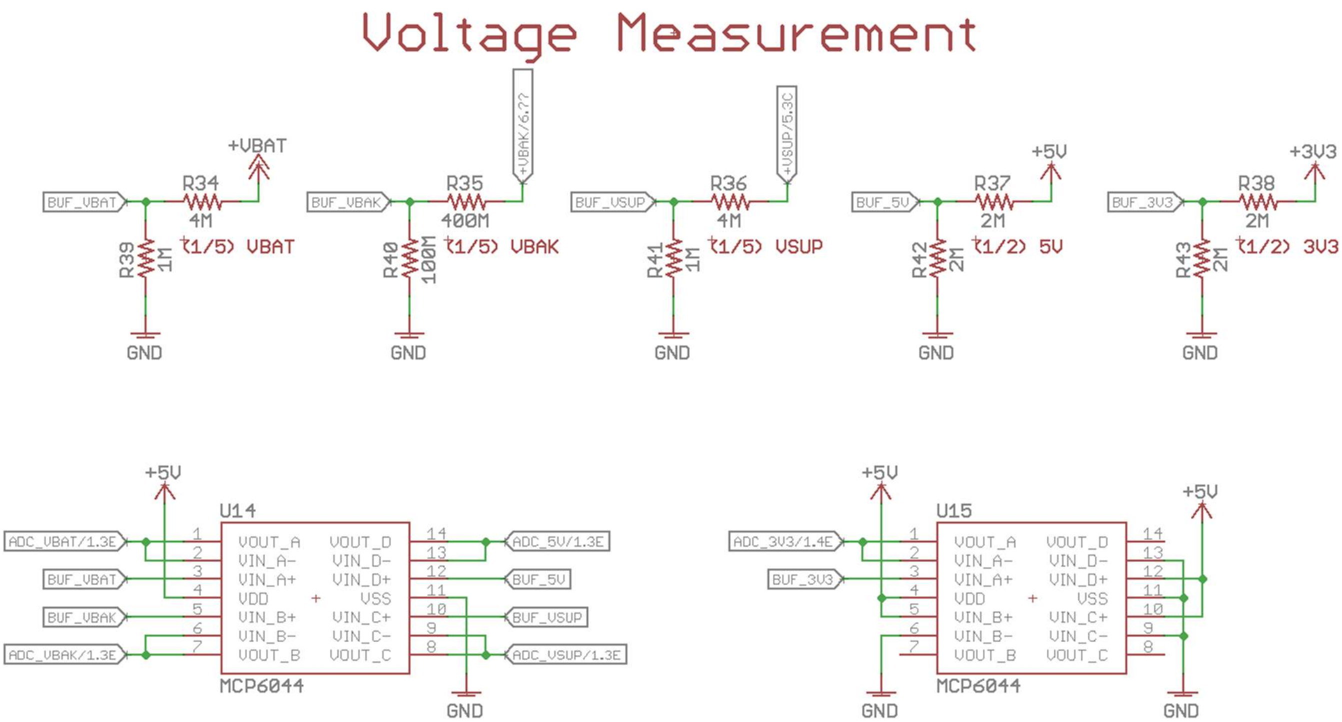

Figure 2.18 Voltage Measurement Schematic

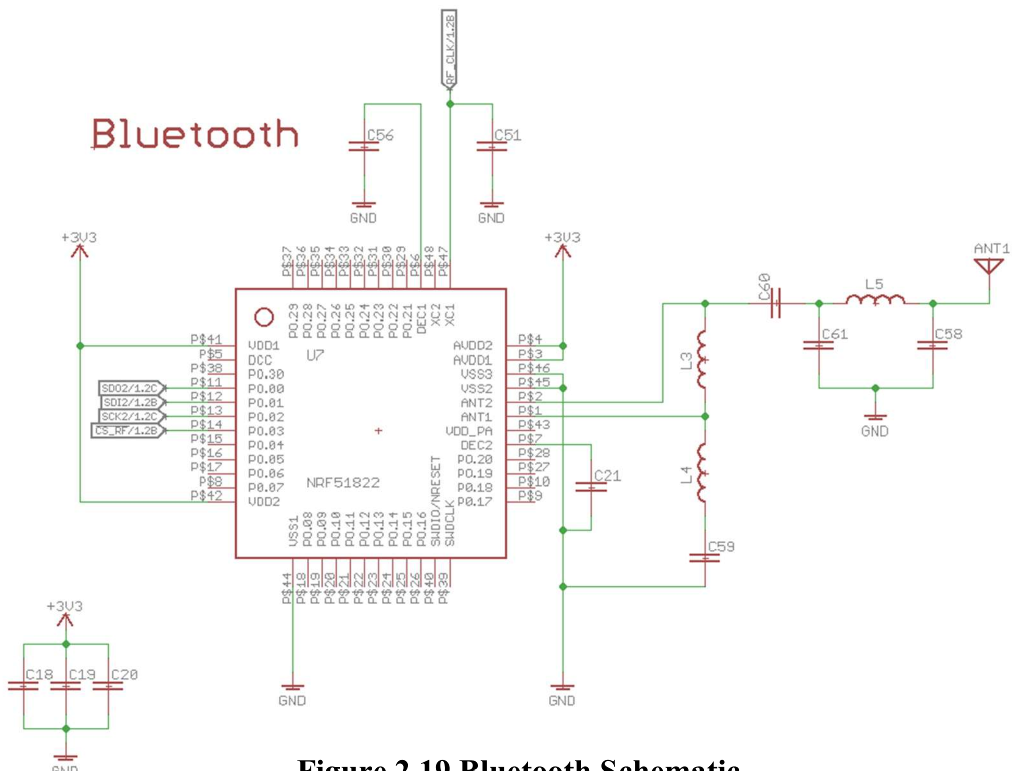

Figure 2.19 Bluetooth Schematic

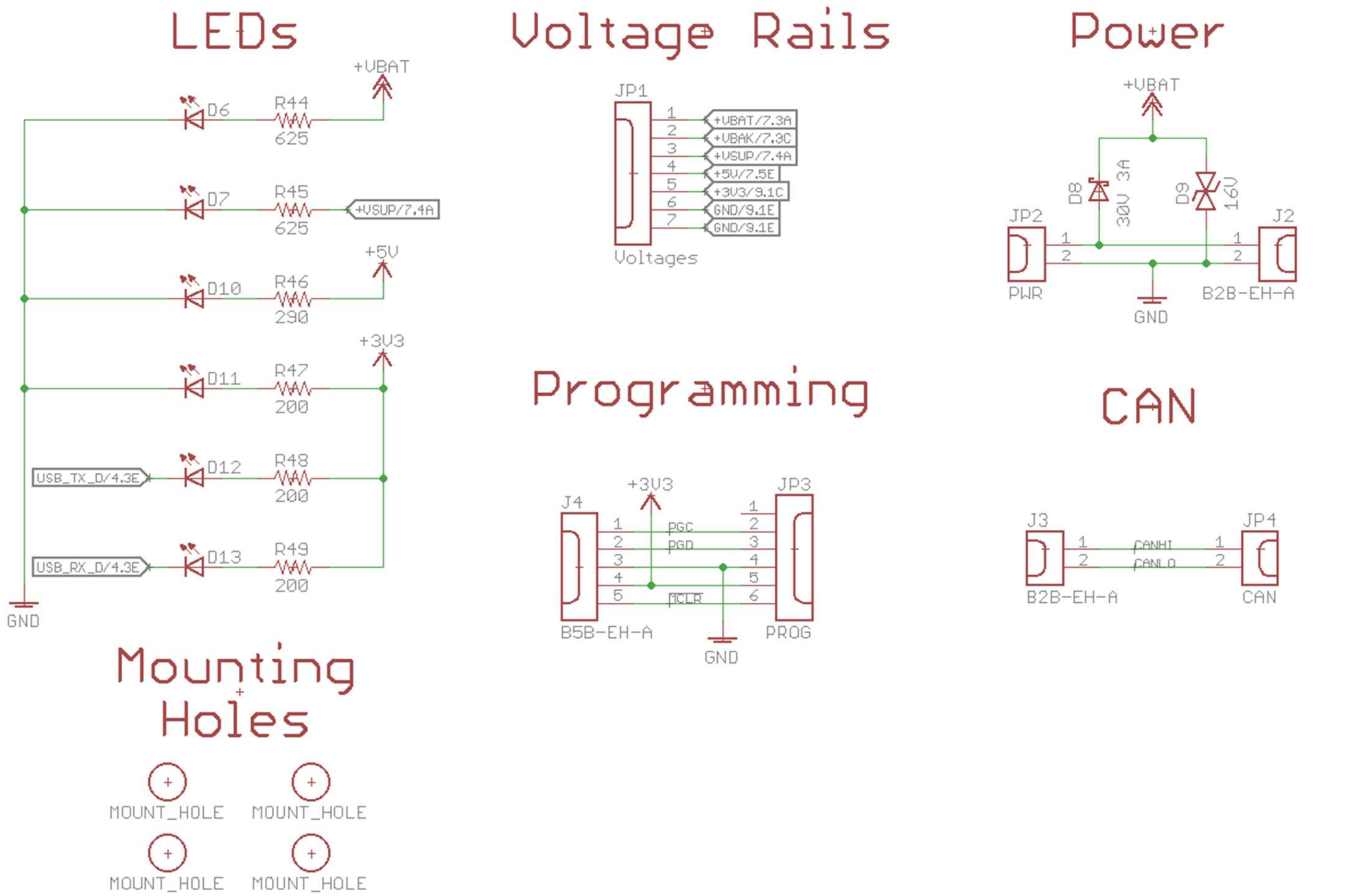

Figure 2.20 Connectors and Debugging Schematic

2.7 Physical Design

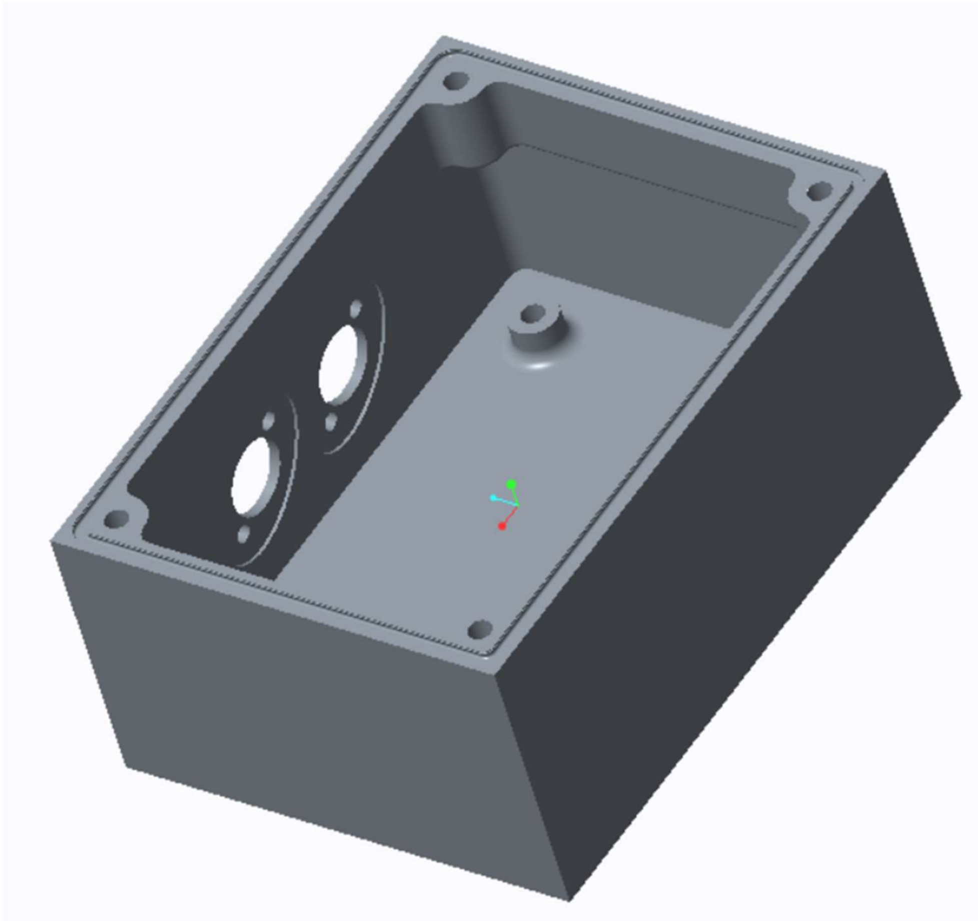

Figure 2.21 Previous Year's PCB Enclosure CAD

When the board is mounted to the car, it is placed in an enclosure for waterproofing and heat shielding purposes. The enclosure body is modeled in Creo Parametric and then 3D printed out of Nylon-12, while the lid is water jetted out of carbon fiber. The PCB will be bolted to the bottom of the enclosure by means of mounting holes placed on its corners. The programming, CAN, and power interfaces will reach the board via Deutsch Autosport connectors mounted to the sides of the enclosure. The USB connector will be attached to the lid. These connectors are then wired directly to headers on the board. The board will likely be placed under the seat of the car and bolted to L-brackets to secure it to the floor of the chassis. To ensure a quality RF signal free of any negative effects due to being encased in carbon fiber, the antenna will be mounted to the exterior of the car and connected to the board with a coaxial cable.

2.8 Tolerance Analysis of Antenna and Matching Network

To maximize transmission power and minimize network losses, the matching network needs to minimize the reflection coefficient at the antenna. The reflection coefficient is calculated as

Eq. 6) $$\Gamma = \frac{Z_L - Z_0}{Z_L + Z_0}$$

The typical characteristic impedance of a coaxial cable is about 50Ω, so maximum power transmission occurs when the input impedance of the load (i.e. antenna) is a purely resistive 50Ω. Antenna input power can be calculated as

Eq. 7) $$P = \frac{V_L^2}{2Z_0}(1 - |\Gamma^2|)$$

Using this equation, the maximum allowable reflection coefficient to achieve power losses of 1dB or less is 0.454, which seems easily attainable. The change in characteristic impedance due to inhomogeneity of the PCB substrate and its higher permittivity can be mitigated by placing the coaxial cable port as close as possible to the Bluetooth IC one end and the antenna on the other. With proper passive sizing to match the antenna and Bluetooth IC, the microstrip power loss should be fairly negligible.

The major cause of increasing reflection coefficient is the change in wavelength over the range of the bandwidth. Assuming a quarter wave monopole antenna and lossless transmission lines, the input current of the antenna can be calculated from the power of 4dBm = 2.512 mW and a characteristic impedance of 36.5Ω as

Eq. 8) $$P = \frac{1}{2}I^2R \rightarrow I = 11.7mA$$

Because there is an antenna input phase difference between the center of the frequency band and the extrema, the radiation power can drop. But if the transmission line is 1m long (approximately the length necessary for this project), the power only drops by 0.1dBm, which is negligible. This means that if the antenna is properly impedance matched with the 50 Ω coaxial cable and the 15Ω + j85Ω Bluetooth IC, the output power is well within power tolerances, even if some unaccounted losses occur due to the microstrip lines.

3 Cost and Schedule

3.1 Cost

Assuming an estimated starting salary of ≈\$75,000, that translates to \$36.00/hour given the equation:

Eq. 9) $$\$75{,}000 * \frac{1}{52\ weeks} * \frac{1\ week}{40\ hours} \approx \frac{\$36}{hour}$$

Estimating 12 hours per week for each member, for 16 weeks of the semester, the total labor cost comes out to be:

Eq. 10) $$2 * 2.5 * \frac{\$36}{hour} * \frac{12\ hours}{week} * 16\ weeks = \$36{,}560$$

Adding the labor cost and the total parts cost from Figure 3.1, the total cost yields:

$$\$36{,}780.74$$

Device	Manufacturer	Unit Cost	Quantit y	Description
PIC32MZ2048EFM100	Microchip	$12.05	1	IC MCU 32BIT 2048KB FLASH 100TQF
MCP2562FD	Microchip	$1.04	1	IC TRANSCEIVER CAN FLEX 8SOIC
LTC3350	Linear	$11.18	1	Supercapacitor Backup Controller PMIC 38-QFN (5x7)
10F 26mOhm EDLC Capacitor	NessCap	$3.72	4	10F Supercap 2.7V Radial, Can 26 mOhm
SD Card Socket	Hirose Electric	$1.88	1	10 (8 + 2) Position Card Connector Secure Digital - microSD™ Surface Mount, Right Angle Gold
ASVMB-24.000MHZ-XY-T	Abracon	$2.19	1	OSC MEMS 24.000MHZ CMOS SMD
Electron Device	Particle.io	$69.00	1	Electron 3G Kit (Americas/Aus)
MCP2562FD	Microchip	$1.04	1	IC TRANSCEIVER CAN FLEX 8SOIC
MCP79522	Microchip	$1.28	1	Real Time Clock (RTC) IC Clock/Calendar 64B SPI
WRL-00691	SparkFun	$19.95	1	SPARKFUN TRANSCEIVER BREAKOUT -
MCP6044	Microchip	$1.41	2	General Purpose Amplifier 4 Circuit Rail-to-Rail
USB to UART	FTDI	$2.38	1	USB Bridge, USB to UART USB 2.0 UART Interface
ABS25-32.768KHZ-T	Abracon	$0.35	1	32.768kHz ±20ppm Crystal 12.5pF 50 kOhm
MAX6610	Maxim	$2.24	1	SENSOR TEMP RATIOMETRIC SOT23-6
LM2675-5.0	TI	$4.37	1	Buck Switching Regulator IC Positive Fixed 5V
TPS73633DCQR	TI	$2.33	1	Linear Voltage Regulator IC Positive Fixed
1Mb EEPROM	Microchip	$2.94	1	IC EEPROM 1MBIT 20MHZ 8SOIJ
18 Position Female Headers	Sullins Connector	$1.43	2	18 Position Header Connector 0.100"
8 Position Female Headers	Sullins Connector	$0.89	1	8 Position Header Connector 0.100"
8 Position Right-Angle Headers	Sullins Connector	$0.38	1	8 Positions Header, Unshrouded, Breakaway Connector 0.100"
B5B-EH-A	JST	$0.24	1	CONN HEADER EH TOP 5POS 2.5MM
B4B-EH-A	JST	$0.20	1	CONN HEADER EH TOP 5POS 2.5MM
B2B-EH-A	JST	$0.14	2	CONN HEADER EH TOP 2POS 2.5MM
6-pin Header	Molex	$2.17	1	CONN HEADER 36POS .100 VERT GOLD
IPD30N03S4L-09	Infineon	$0.65	4	MOSFET N-CH 30V 30A TO252-3
AO3418	Alpha & Omega	$0.44	1	MOSFET N-CH 30V 3.8A SOT23
CPC1017N	IXYS	$1.08	2	RELAY OPTOMOS SP-NO 100MA 4-SOP
CPC1020N	IXYS	$3.20	1	Solid State Relay SPST-NO
3.3V Zener Diode	NXP Semiconductors	$0.36	1	Zener Diode 3.3V 550mW ±2% Surface Mount SOD-323F
30V Shottky Diode	Diodes Incorperated	$0.51	1	Diode Schottky 30V 3A Surface Mount SMA
20V 1A Diode	Vishay	$0.38	1	DIODE SCHOTTKY 20V 1A DO214AC
16V TVS Diode	ON Semiconductor	$0.47	1	TVS DIODE 16VWM 26VC SMB
75V Signal Diode	Fairchild	$0.13	3	Diode Standard 75V 150mA Surface Mount SOD-323F
30V Shottky Diode	Diodes Incorperated	$0.51	1	Diode Schottky 30V 3A Surface Mount SMA
LTST-C150GKT	Lite-On	$0.30	6	LED GREEN CLEAR 1206 SMD
22u 3A Inductor	Wurth Electronics	$2.76	1	22µH Shielded Wirewound Inductor 3A 50 mOhm
68uH Inductor	Wurth	$2.52	1	68µH Shielded Wirewound Inductor 870mA
All Resistors	Yageo	$8.52	1	RES SMD 1/4W 1206
All Ceramic Capacitors	Kemet	$18.92	1	Ceramic Capacitor 1206
All Tantalum Capacitors	AVX	$16.26	1	Molded Tantalum Capacitors 2917
Total	**$220.74**			

Figure 3.1 Parts BOM

3.2 Schedule

Week	Michael	Jacob
10/7/18	- Design Review Prep - Version 1 & 2 of PCB layout	-Design Review Prep -Finalize Bluetooth and Supercapacitor schematics
10/14/18	- Begin Version 1 PCB Routing - Develop library for USB-UART interface - Begin SD card interface software	-Matching network design - RF/Antenna routing
10/21/18	- Finish PCB Routing	-Generate Gerber files/order PCB (Advanced circuits) -Begin Bluetooth Software
10/28/18	- Solder the PCB - Test peripheral functionality of the PCB - If functionality isn't present, diagnose the issue and reroute Version 2 of the PCB if needed	-Finish Bluetooth Software -Start Supercapacitor I^2C communications software
11/4/18	- Finish USB-UART library development SD card interface	-Finish Supercapacitor I^2C communications software - Test/refine matching network
11/11/18	- Program microcontroller to process CAN messages	- Test Supercapacitor Bank - Test Bluetooth communications
11/18/18	BREAK	BREAK
11/25/18	- Test microcontroller software functionality - Test overall system functionality	- Laptop User Interface Software
12/2/18	- Begin final lab report - Project presentation prep	- Final report - Project presentation prep
12/9/18	- Finish final lab report	- Finish everything

4 Ethics and Safety

The Lithium-Ion car battery is extremely dangerous if the terminals are shorted together. Extreme caution is needed when handling the battery near metal tables or anything capable of being a conductor. Lithium-Ion batteries can also explode if overcharged or if they are exposed to temperatures outside the recommended operating conditions. The team ensures that the battery is well ventilated and never excessively charged under IEEE Code of Ethics #1 (safe, healthy, and environmentally safe) [12].

The purpose of this project is to design a faster, lighter, and safer race car by improving the usage of sensor data. This data can be used to validate mathematical and computational models or provide relevant numbers where models do not exist. This is in direct compliance with IEEE Code of Ethics #5 (to improve understanding through technology) [12]. To reduce the chances of violating IEEE Code of Ethics #3 (to be and realistic with data interpretation), the team always tries to validate all data through both basic hand calculations and complex computational models [12]. If the numbers vastly disagree, we expend every effort to discover why.

References

[1] Kvaser, 'CAN Protocol Tutorial', 2018. [Online]. Available: https://www.kvaser.com/can-protocol-tutorial/

[2] R. Roderick, 'A Look Inside Battery-Management Systems', 2015. [Online]. Available: https://www.electronicdesign.com/power/look-inside-battery-management-systems

[3] Linear Technology, 'High Current Supercapacitor Backup Controller and System Monitor', 2014. [Online]. Available: www.analog.com/media/en/technical-documentation/data-sheets/3350fc.pdf

[4] Renesas, 'Switching Regulators', 2018, [Online]. Available: https://www.renesas.com/us/en/products/power-management/switching-regulators.html

[5] Texas Instruments, 'LM2675 SIMPLE SWITCHER® Power Converter High Efficiency 1-A Step-Down Voltage Regulator', 2016. [Online]. Available: http://www.ti.com/general/docs/lit/getliterature.tsp?genericPartNumber=lm2675&fileType=pdf

[6] D. Knight, 'Introduction to Linear Voltage Regulators', 2016. [Online]. Available: https://www.digikey.com/en/maker/blogs/introduction-to-linear-voltage-regulators

[7] Texas Instruments, 'CAP-FREE NMOS 400 mA LOW-DROPOUT REGULATORS WITH REVERSE CURRENT PROTECTION', 2014. [Online]. Available: http://www.ti.com/general/docs/lit/getliterature.tsp?genericPartNumber=tps73633-ep&fileType=pdf

[8] Microchip Technology, 'PIC32MZ Embedded Connectivity with Floating Point Unit (EF) Family', 2018. [Online]. Available: www.microchip.com/mymicrochip/filehandler.aspx?ddocname=en574992

[9] Microchip Technology, 'High-Speed CAN Transceiver', 2014. [Online]. Available: www.microchip.com/mymicrochip/filehandler.aspx?ddocname=en561044

[10] Maxim Integrated, 'Precision, Low-Power, 6-Pin SOT23 Temperature Sensors and Voltage References', 2003. [Online]. Available: https://datasheets.maximintegrated.com/en/ds/MAX6610-MAX6611.pdf

[11] Microchip Technology, 'Battery-Backed SPI Real-Time Clock/Calendar', 2018, [Online]. Available: www.microchip.com/mymicrochip/filehandler.aspx?ddocname=en557883

[12] Future Technology Devices International, 'FT230X (USB to BASIC UART IC)', 2016. [Online]. Available: www.ftdichip.com/Support/Documents/DataSheets/ICs/DS_FT230X.pdf

[13] Nordic Semiconductor, 'nRF51822: Multiprotocol Bluetooth® low energy/2.4 GHz RF System on Chip', 2018, [Online]. Available: http://www.nordicsemi.com/eng/content/download/13358/214991/file/nRF51822_PS%20v3.1.pdf

[14] Nordic Semiconductor, 'nRF51 Series: SoftDevices', 2018, [Online]. Available:http://infocenter.nordicsemi.com/index.jsp?topic=%2Fcom.nordic.infocenter.s140.sds%2Fdita%2Fsoftdevices%2Fs130%2Fble_protocol_stack%2Fble_protocol_stack.html

[15] Antenna Theory, 'Inverted-F Antenna (IFA)', 2018,[Online]. Available: http://www.antenna-theory.com/antennas/aperture/ifa.php

[16] IEEE, 'IEEE Code of Ethics', 2018. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html