LOW COST DISTRIBUTED BATTERY MANAGEMENT SYSTEM

ECE445 Design Document - Fall 2018

Team 1- Daksh Saraf and Logan Rosenmayer
TA: Evan Widloski

Contents

1 Introduction

1.1 Objective

While lithium batteries are incredible for their energy density and longevity, they also bring with them many downsides. Lithium Ion (Li-ion) batteries are very sensitive to being out of their temperature, current or voltage specification. Violating said properties can often lead to battery pack degradation or even the pack catching fire. Additionally, many applications that use lithium batteries require much higher voltages than their cell average of 3.6/3.7V such as electric cars. This demand for higher voltages has forced manufacturers to place batteries in series to create desired voltages without the need of expensive and relatively inefficient DC/DC converters. Even if the batteries placed in series initially have the same charge, over time the battery cells' charges will begin to drift away from one another and must be brought back together through balancing to prevent damage to the battery and to maintain battery capacity.

To tackle this problem, we plan to take advantage of low-cost microcontroller units (MCUs) and a variety of peripherals to make a low cost distributed BMS. Our goal is to make the LEGOs of BMSs in terms of easiness to assemble and operate. To accomplish this, we will utilize a stackable PCB design with a master that can easily adjust to the addition and removal of modules with the touch of a button.

1.2 Background

In the market today exist many Battery Management Systems (BMSs) that keep batteries balanced and operating in spec, however they are either centralized approaches that offer little flexibility on changes in system configuration, require many wires, and large amounts of effort to implement or distributed systems that can cost over $600 for a 16S battery[1]. This causes many people who are looking to create one-off projects or prototypes with lithium batteries in series to either face large unneeded complexity and messy wiring or face the exorbitant cost of current distributed BMSs.

These hurdles can keep many from making the jump to lithium ion batteries or worse lead to battery packs being made without proper protection. Such battery packs can be very detrimental as they can cause a lot of damage by catching fire, which in turn, can burn down entire buildings and take lives. A common example of this are the fires caused by e-bikes that often utilize cut-rate BMSs or none at all[2].

1.3 High-level requirements list

- Each slave module must be able to bring its batteries within 50mV of its target voltage
- Slave and master modules must have bidirectional communication for reporting battery data and setting target voltage for balancing.
- The BMS must monitor cells and protect all cell from enter a state violating their specification per there data sheet (except under voltage due to BMS power consumption and battery self-discharge).

2 Design

2.1 Block Diagrams

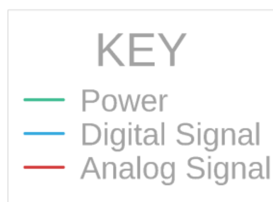

Figure 2.1.1: Block Diagram Key

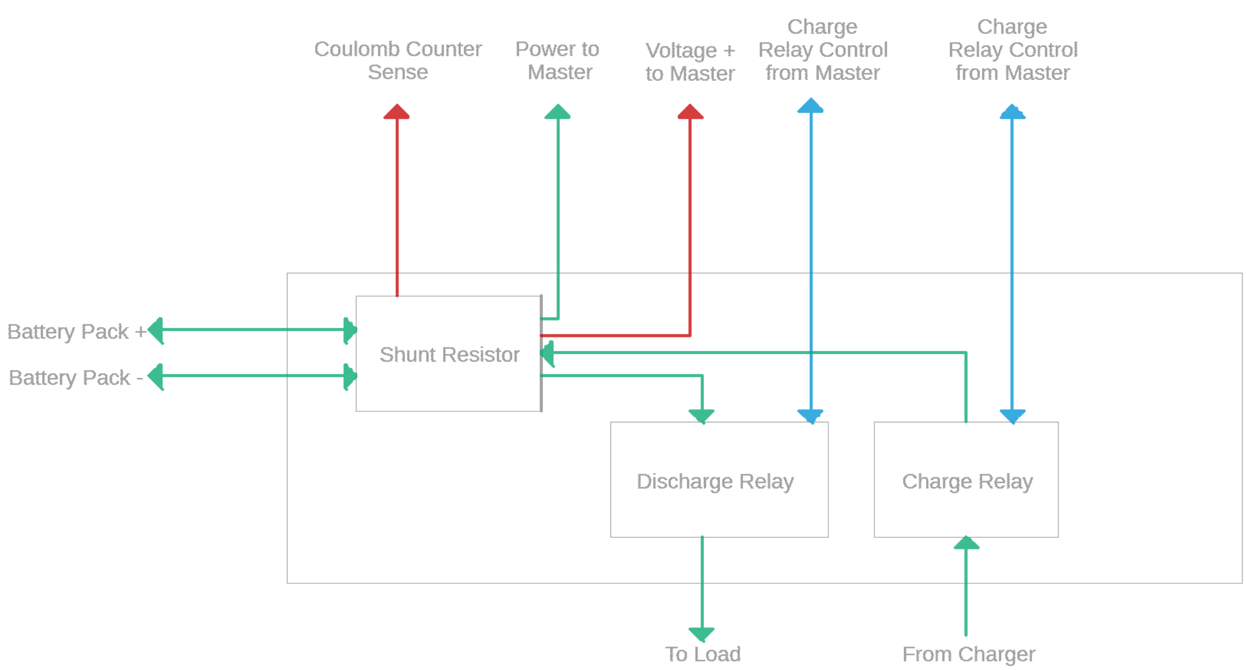

Figure 2.1.2: Block Diagram of Power Components

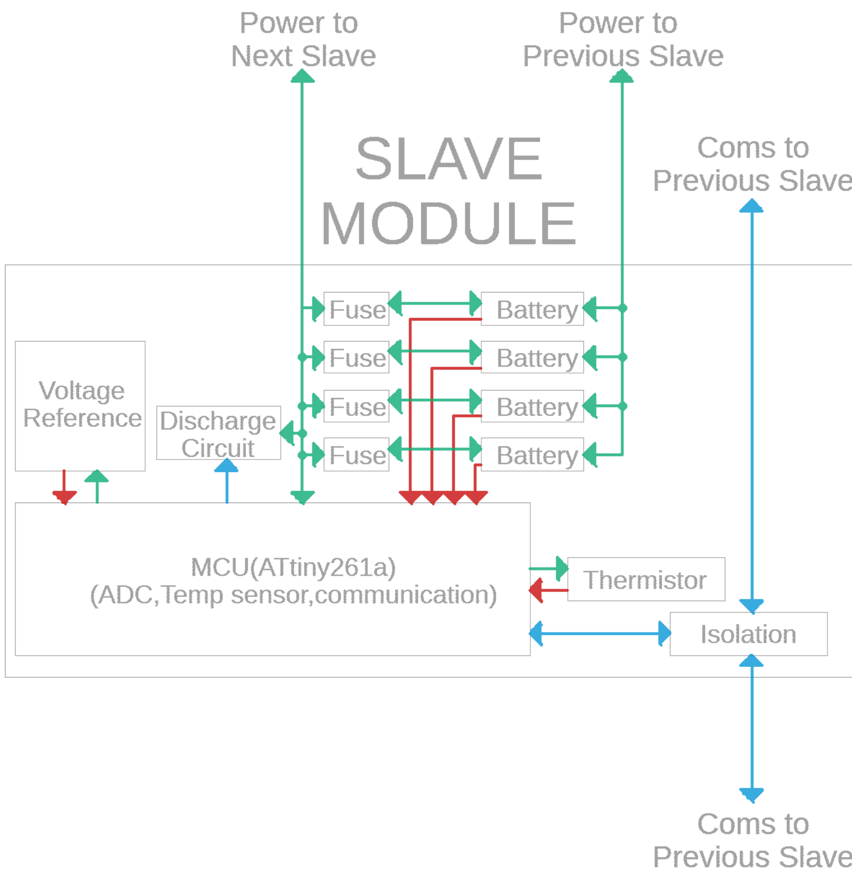

Figure 2.1.3: Slave Block Diagram

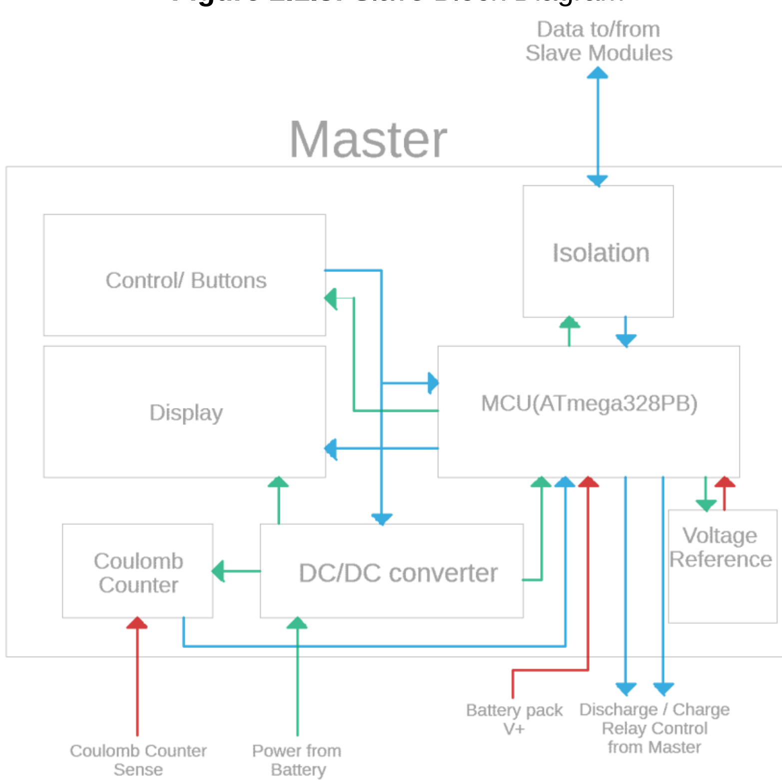

Figure 2.1.4: Master Block Diagram

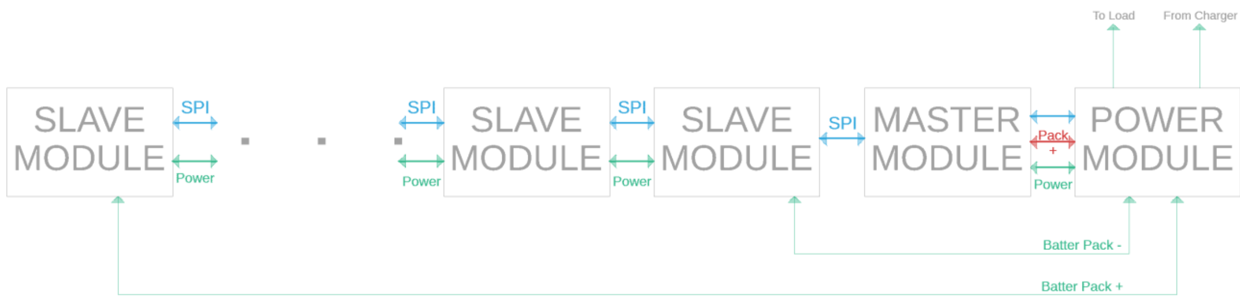

Figure 2.1.5: BMS High Level Block Diagram

2.2 Physical Design

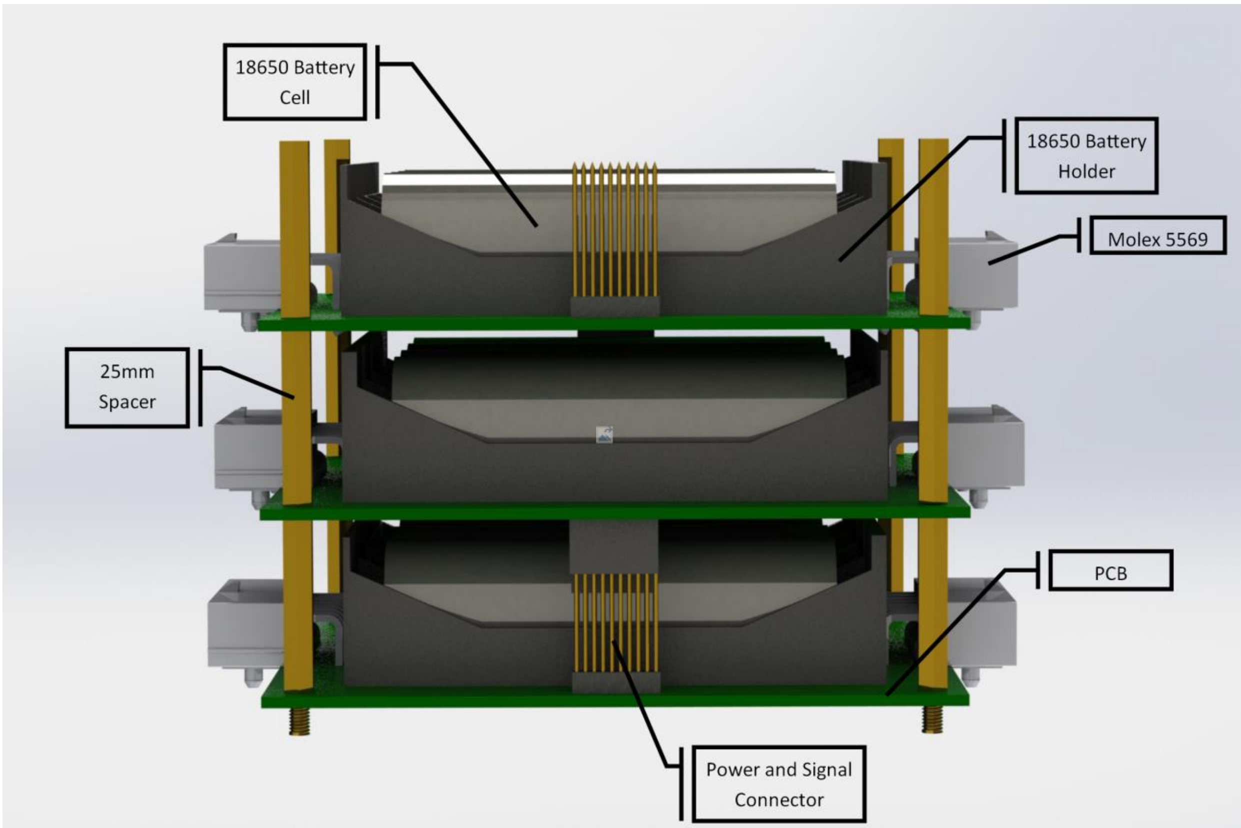

Figure 2.2.1: Basic Slave Module Stack Render

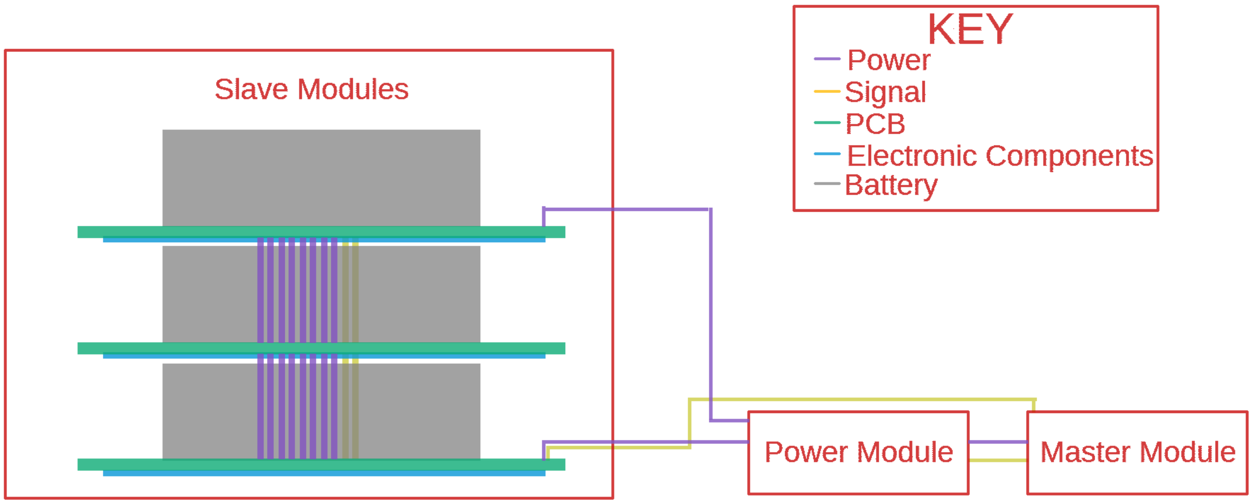

Figure 2.2.2: BMS Physical Layout Diagram

Each slave module can hold up to four 18650 cells that are held in place with leaf springs. To make the BMS easy to configure we opted to make it stackable. The 25mm brass spacers on each corner provide structural support while allowing for quick and easy reconfiguration. Module to Module power and communication signals are sent through the 25mm tall 2x10 pin headers. To stack 2 modules, one must be rotated 180° with respect to the other. To allow for many modules to be used without creating towers of unstable height multiple towers can be daisy chained using a Molex 5569 connector for power and communication by connecting the top of one tower (+) to the bottom of the next tower (-). Likewise, this connection can be used to connect the master and power components to the 1st slave modules. The master module and power components have mounting that cannot be a part of the towers.

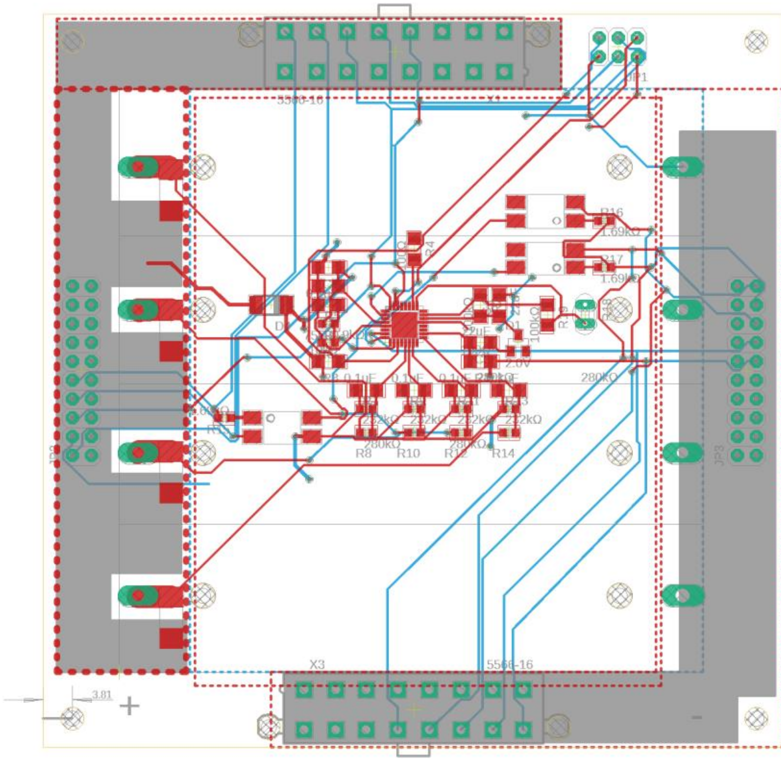

Figure 2.2.3: Slave Board Layout Rev 1

Shown above in figure 2.2.3 the Slave module board will have all smd components of the BMS on onside. All components of the BMS on the SMD side are below 3 mm in height which is the allotted space between the bottom of the PCB and the top off the battery case below it.

2.3 Functional Overview and Block Requirements

2.3.1 Slave Module:

The slave module sends battery data to master upon receiving a request and acts on master's command to balance battery.

Requirement	**Verification**
1. Can perform bidirectional communication with other slaves or the master at a baud of at least 9600 2. Can monitor its cells voltage within 50mV 3. Can monitor module temperature within 10°C 4. Can prevent any cell from discharging above it rate current continuously(20A)	1. Test that SPI communication with at least a 9600 Hz clock can be done through the isolation 2. Test that ADC can measure a 4.1V signal with less that 50mV error 3. Measure room temperature and compare against multimeter thermocouple 4. Short battery and confirm that fuse blows 5. Confirm that module operates off

5. Can run off 2.5-4.2V 6. Can dissipate battery charge at a rate of at least 40mA for balancing	external power supply at 2.5V and 4.2V 6. Confirm that resistor value is low enough that V=IR will result in a <40mA current

Balancing Resistor:

The balance resistor is used to burn off excess power on batteries during the balancing phase to bring them all down to the same voltage.

Requirement	**Verification**
1. Balance resistor must be able to handle at least 50mA. 2. Balance resistor must draw > 40mA at 4.2V.	1. Run 50mA through resistor for 5min and confirm it does not break 2. Confirm that resistor value is low enough that V=IR will result in a >40mA current

Battery:

Stores energy to power external devices as well as the BMS itself.

Requirement	**Verification**
1. Can handle voltages from 4.1-3.0V and continuous currents of 15A.	1. Discharge to 3.0V and charge up to 4.1V a. Use spec sheet to confirm safe voltage range

Fuse:

Prevents internal shorts from taking down the whole battery and possibly causing a fire and prevents external shorts from over currenting batteries.

Requirement	**Verification**
1. Prevent continuous currents above battery rated value (20A) by opening the circuit.	1. Short battery and confirm that fuse blows

Isolation:

Allows for communication between modules at different voltage level to allow for daisy chaining communication. This allows slaves to talk to one another, thus allowing for daisy chaining.

Requirement	Verification
1. Can support a baud of at least 9600b/s	1. Test that SPI communication with at least a 9600Hz clock can be done through the isolation

MCU:

The MCU does most of the work on the module. The chip will take voltage measurements of the batteries and thermistor to ensure that all cells are within there operational area. Additionally, the MCU sends and receives communication through its communication busses. The IC must also manage power with its GPIO pins for the thermistor and the voltage reference.

Requirement	Verification
1. Can support a baud rate of at least 9600 b/s on both busses (not simultaneously) 2. Measure battery voltages within 50mV 3. Manage power consumption	1. Test that SPI communication with at least a 9600 Hz clock can be done through the isolation 2. Test that ADC can measure a 4.1V signal with less that 50mV error 3. Can power all devices whose power it is to manage with GPIO pins

Thermistor:

Used to convert temperature into an analog voltage by using thermistor in voltage divider.

Requirement	Verification
1. Analog voltage must represent temperature within 10°C	1. Measure room temperature and compare against multimeter thermocouple

Voltage Reference:

The MCU’s internal analog reference has poor accuracy, to improve ADC readings a more accurate voltage reference must be used.

Requirement	**Verification**
1. Voltage reference produces a 2.048 V ±1mV rail under ADC reference load.	1. Measure diode reverse voltage with multimeter

2.3.2 Master:

Controls slaves and power components to ensure proper BMS functionality.

Requirement	**Verification**
1. Can display information to user without additional hardware 2. Can accept user inputs to adjust configuration / change state 3. Master must be able to track charge in and out of the battery with less than 5% error 4. Can communicate with a slave at 9600 b/s 5. Master can stop charging or discharging of battery	1. Display battery data with LCD 2. Test button presses can be detected by MCU 3. Run current through shunt resistor and confirm that coulombs that the theoretical value of coulombs are within 5% of the theoretical 4. Send and receive test messages through SPI to a slave with a 9600Hz SCK 5. Test that master can control relay

Control:

Allows user to easily change basic parameters of the BMS and see battery status without need of changing and uploading code.

Requirement	**Verification**
1. Allows humans to interface with BMS without additional equipment	1. Test that buttons can be used to interact with master to display different data

Coulomb Counter:

Allows for a simple way of tracking battery SOC with good accuracy by integrating current to track “how many electrons are left”. Additionally, allows for an “battery odometer” which is useful for measuring energy consumption during use and battery lifetime ware.

Requirement	Verification
1. Track battery SOC within 5%	1. Run current through shunt resistor and confirm that coulombs that the theoretical value of coulombs are within 5% of the theoretical

DC/DC Converter:

Converts battery output voltage (7.5-100V) to 5V for MCU, other master components, and relay power.

Requirement	Verification
1. Create 5V ±0.5V average voltage under 300mA load	1. Measure output of converter when powered by battery

Display:

Let’s BMS communicate battery information and fault conditions to users without need of extra equipment.

Requirement	Verification
1. Display data sent from master MCU	1. Create fault conditions and confirm that BMS detects them and displays the fault

Isolation:

Isolation allows for communication between modules at different voltage level to allow for daisy chaining communication.

Requirement	**Verification**
1. Can support a baud rate of at least 9600b/s.	1. Test that SPI communication with at least a 9600 Hz clock can be done through the isolation

MCU:

The master MCU main job is to query the slave modules and coulomb counter for data and controls the charge and discharge relays to prevent the batteries from breaking their specified ratings. The master should also be able to take battery data and send out a command that specifies to the slaves which voltage to reduce themselves to. Additionally, the master should monitor for user inputs and respond accordingly.

Requirement	**Verification**
1. Can support a baud of at least 9600b/s over SPI. 2. Can support a baud rate of at least 9600b/s over I2C. 3. Can support dallas one wire at standard speed 4. MCU has: >500B SRAM, >1KB Flash and >13GPIO pins without need of other ICs.	1. Test that SPI communication with at least a 9600Hz clock can be done through the isolation 2. Test that I2C communication with at least a 9600Hz clock can be done through the isolation 3. Read data from coulomb counter registers with one wire 4. Use spec sheet to confirm

Voltage Reference:

The MCU's internal analog reference has poor accuracy, to improve ADC readings a more accurate voltage reference must be used.

Requirement	**Verification**
1. Voltage reference produces a 2.048V ±1mV rail under ADC reference load.	1. Measure diode reverse voltage with multimeter

2.3.3 Power Components

Power components are off board components since there power rating it to high to be put on the master, additionally the master is agnostic to the exact components allowing for greater flexibility.

Shunt Resistor:

The shunt resistor is a high current resistor whose voltage can be used to calculate current through it with I=V/R.

Requirement	Verification
1. Shunt must have a 60A continuous current rating 2. Shunt voltage must be below 50 mV at 60A.	1. Use spec sheet to confirm rating 2. Measure resistance and use V=IR to confirm the voltage across the resistor will be below 50mV at 60A

Discharge Relay:

The relay is used to prevent load from over discharging battery and is controlled with a digital pin from MCU.

Requirement	Verification
1. 60A continuous current rating 2. Can be controlled with MCU.	1. Use spec sheet to confirm rating 2. Test MCU can turn on and off relay with GPIO

Charge Relay:

The relay is used to prevent charger from overcharging battery and is controlled with a digital pin from MCU.

Requirement	Verification
1. 16A continuous current rating 2. Can be controlled with MCU	1. Use spec sheet to confirm rating 2. Test MCU can turn on and off relay with GPIO

2.4 Supporting Materials

2.4.1 Circuit Schematics:

Figure 2.4.1 shows the schematic for the master of the BMS. The master is composed of three main ICs; the MCU which handles communication and data processing, the Coulomb counter that integrates current flow in and out of the battery, and the DC/DC converter that allows the Master to run off the high voltage output of the battery pack. The slave schematic in figure 2.4.2 is composed of only one major IC; the MCU that handles communication, data collection, and power management for the slave.

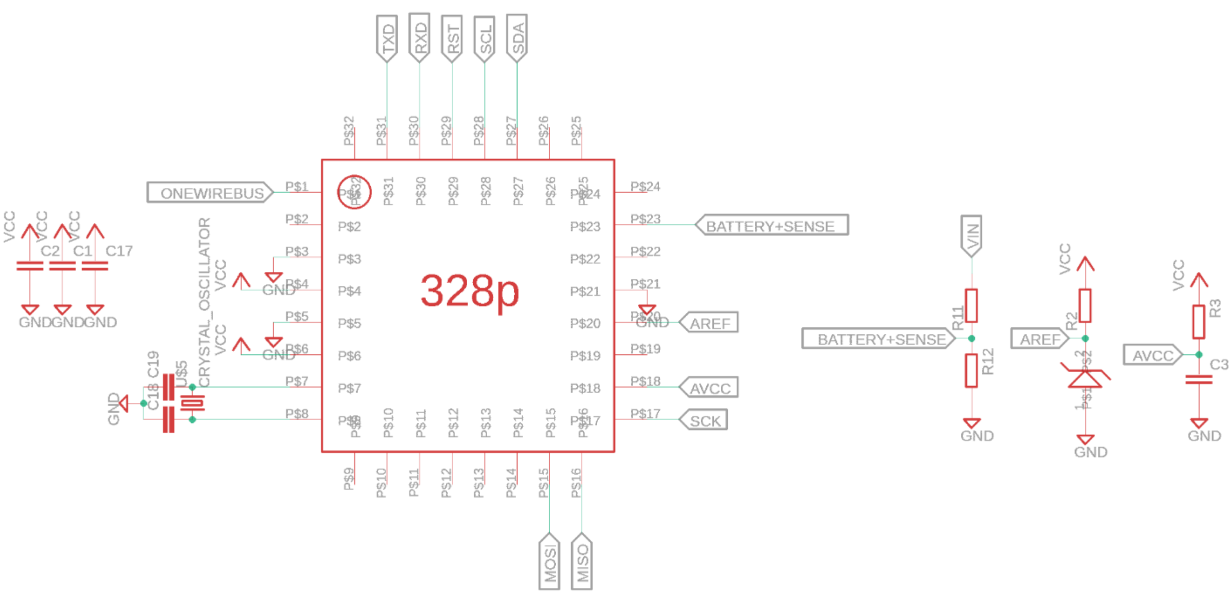

Figure: 2.4.1: Master Module MCU Schematic

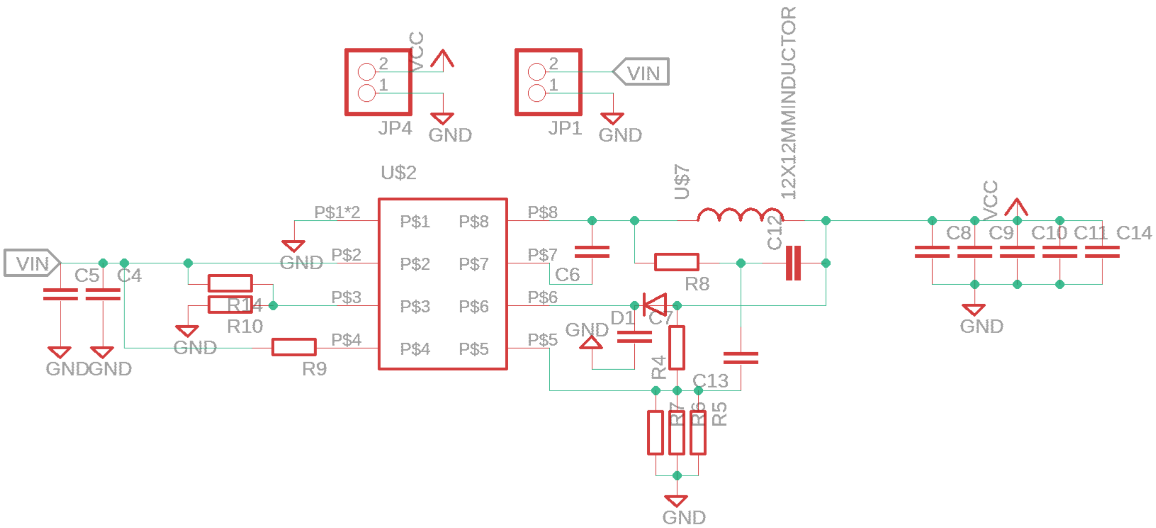

Figure: 2.4.2: Master Module DC/DC Converter Schematic

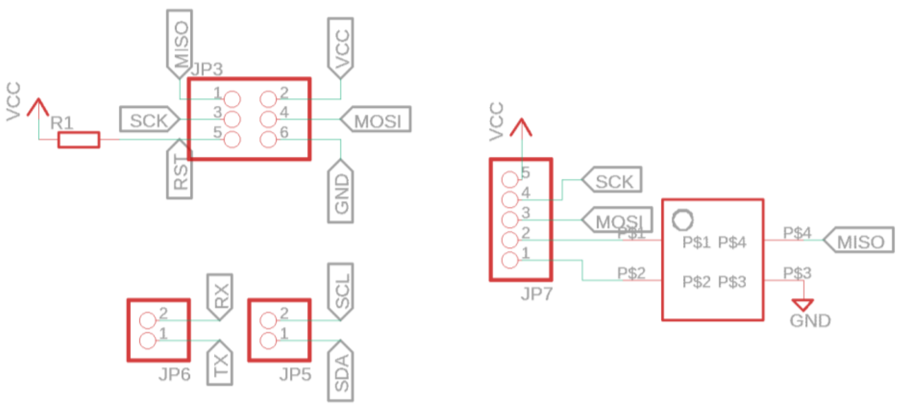

Figure: 2.4.3: Master Module Communication Schematic

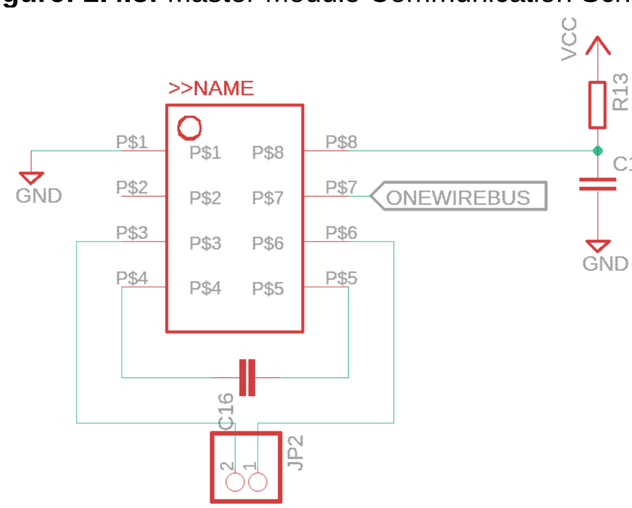

Figure: 2.4.4: Master Module Coulomb Counter Schematic

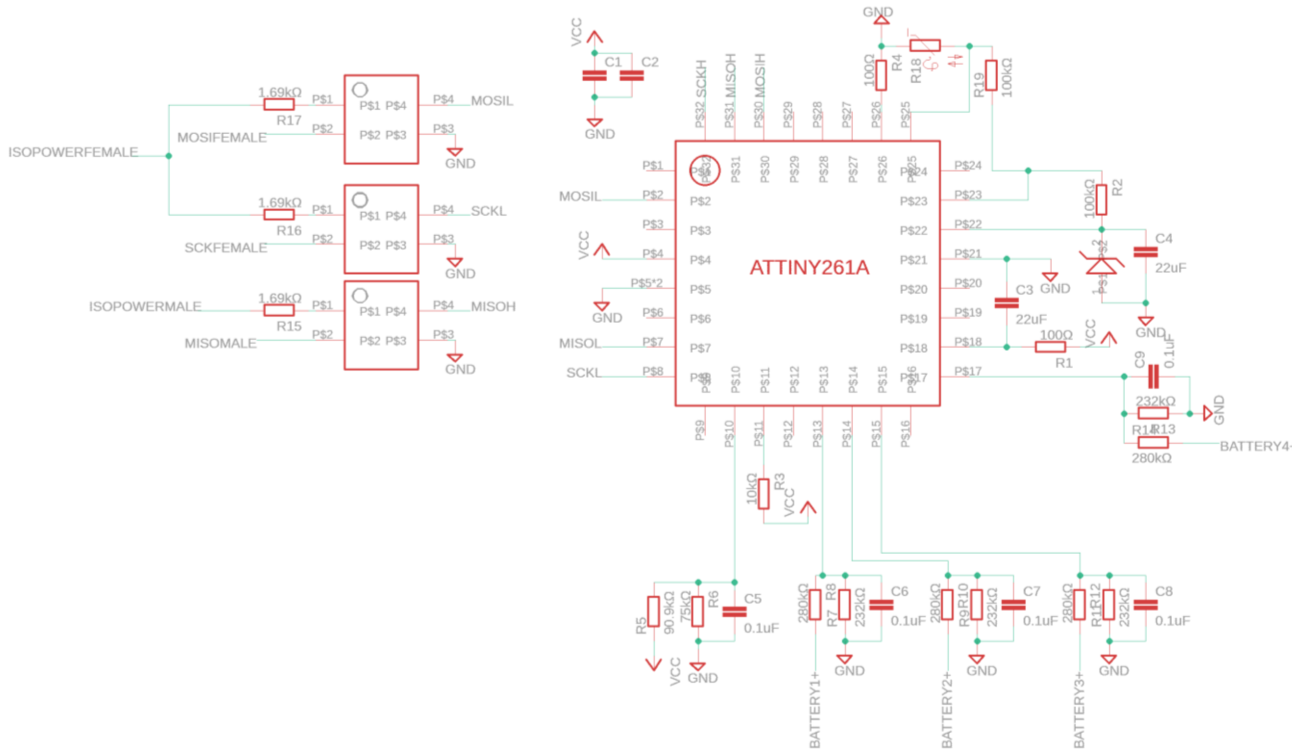

Figure: 2.4.5: Slave Module MCU & Isolation Schematic

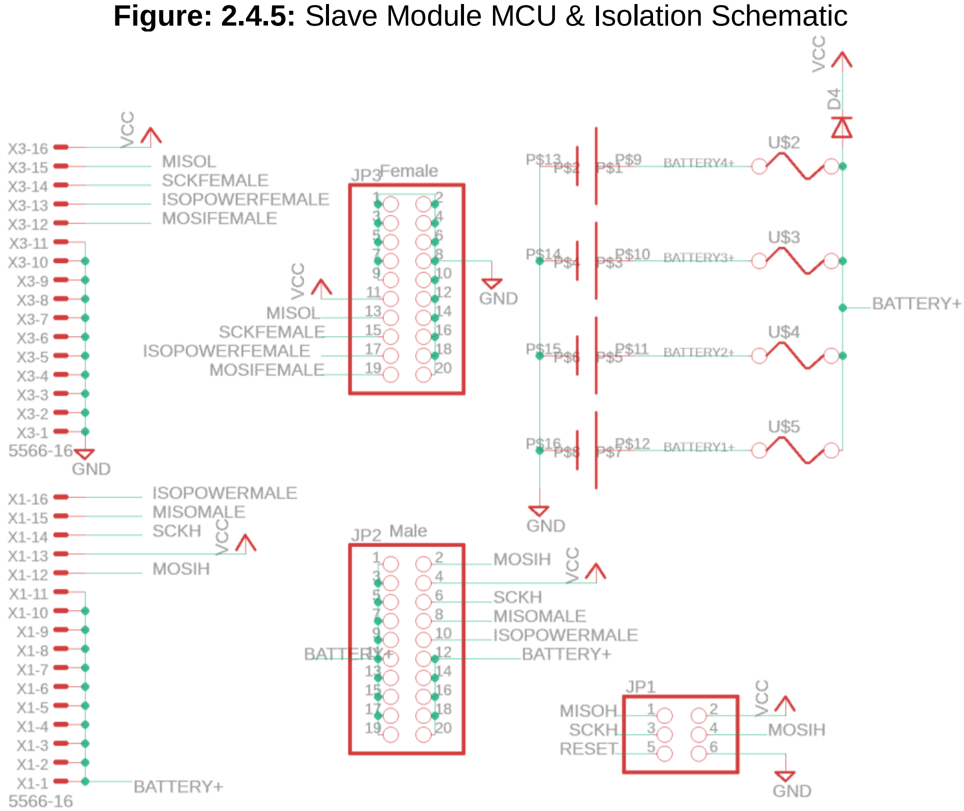

Figure: 2.4.6: Slave Module Connectors Schematic

2.4.2 Flow charts:

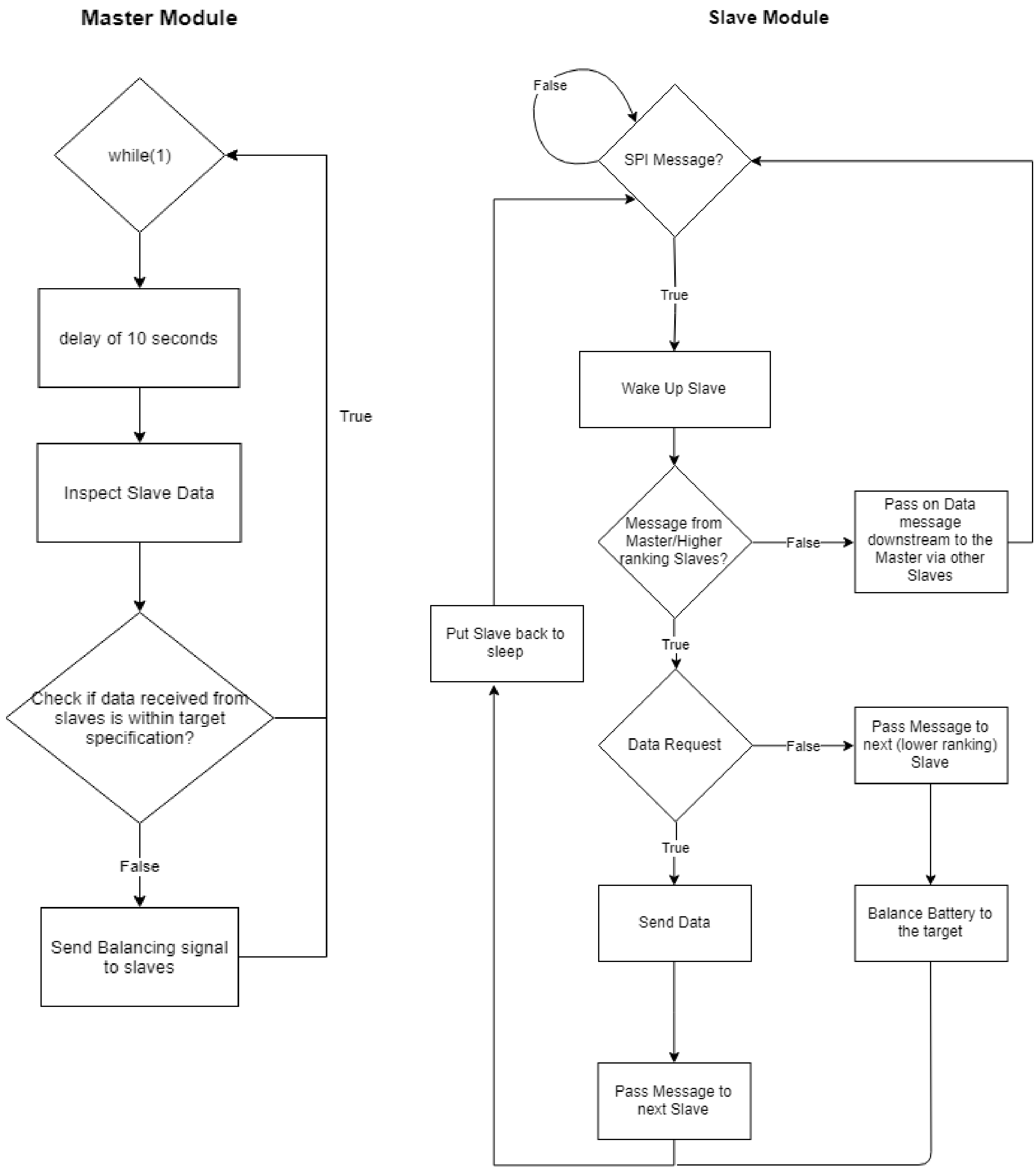

Figure 2.4.7: Flowcharts showing the logical flow of the software on both the Master as well as the Slave MCU

The Master module, as seen above on the flowchart is responsible for polling the slaves for data. This is done every 10 seconds to conserve power. While the master is waiting 10 seconds to poll the slaves again, the slaves go to sleep to enter a low power state to conserve power. The Master on receiving the data from the slaves checks if each slave is within its temperature, voltage and current specifications. If yes, then it will poll the slaves again after a delay of 10 seconds. On the other hand, if the slaves become out of balance, the Master will send a balancing signal to the slaves instructing them to perform certain operations to get balanced.

The Slave modules sleep until they receive a SPI message from the Master. The slaves can generally receive two kinds of signals:

1. From the Master/higher ranking slaves. The signals can be either be for data request or it can be a balancing signal instructing the slaves to get balanced.
2. From low ranking slaves. These signals can be data signals the lower ranking slave wants to send the master, but due to the sequential nature of the design the low-ranking slave must route the signal via the higher-ranking slave.

If the signal is of Type 1, then the slave is woken up. If the signal is for a data request, the slave passes the data downstream to either the Master or higher-ranking slaves and passes the SPI signal upstream to the lower level slaves. On the other hand, if the signal is for balancing, the slave first passes on the signal upstream to the lower level slaves and then starts balancing the battery to the target specification. After this, it goes back to sleep.
If the signal is of Type 2, then the slave merely transfers the data downstream towards the master and goes back to sleep.

2.4.3 Risk Analysis

One of the main attributes of the project is to save as much power as possible to ensure an efficient battery management system. Hence the slave modules go to sleep, in a low power state when not in use to conserve power. They are woken up by the master whenever needed and this communication protocol between the master and the slave modules is crucial to the project.

Our project also demands our BMS to be cost efficient compared to the existing products in the market, so the chips we are using have hardware limitations, making it hard for communication protocols like SPI to work efficiently. These hardware limitations can be compensated by making SPI more versatile on the software side. This means all communication must be bit banged by the MCU which makes reading the first bit of the message a concern. Depending on the decided speed of the bus the MCU may not be able to wake up in time and read its GPIO pin. This could lead to byte stream misses which in turn would cause the slave module to miss signals from the master when they need to be activated. Also, the communication is done next to high energy wires that introduce large amounts on noise on the system potentially causing bit flipping. These errors can produce false readings that can result in reduce reliability of the BMS and may also lead to slave behavior that can further unbalance the battery. To avoid this a robust communication protocol needs to be developed on the software side which would prevent such undefined behavior of the BMS.

2.5 Tolerance Analysis

For the BMS to be effective it must be able to know the voltage of all its cells. This task falls to the ADC of the slave module MCUs. It is vital that the ADC can take precise and accurate measurements so the BMS can take proper actions to protect and balance its cells. By using the voltage reading to detect over charge or discharge the BMS can keep its cells in a safe range during operation. Additionally, during balancing the readings are used to both decide the balance target and tell when the cells of a module have reached such a target. To ensure accuracy of the ADC we must identify the main sources of error. Gain error from the internal amplifier, system noise, ADC reference voltage, and the external voltage divider are all major sources of error. Gain error can rise to 15 LSB (30mV) typical when using the internal amplifier for differential signal measurements. To minimize the error, we opted to use the setting with the lowest amount of error; single ended reading with a gain of 1x. This setting reduces our gain error to 2.5LSB (5mV).

$$LSB=2.048\ V/2^{10}=0.002V$$

Equation 2.5.1

The ADC sits below a battery that will be powering a motor controller capable of drawing over a 100A for short periods of time. These large pulses in current create large amounts of electrical noise that can cause errors in readings. To reduce the effects of such noises the low pass filters are used for ADC power, ADC sense pins and voltage reference. The motor controller runs at 60 kHz so low pass filter that can filter out noise above 60 kHz should be used. The internal reference of the ADC is 1.1V with 100mV or error. This error alone violates our requirement of less than 50 mV of error. Additionally, the lower voltage reference requires greater division of the input voltage that can reach up to 4.2V. Since battery voltage is calculated by multiplying the ADC reading by the voltage divider, the larger the divider the greater the error of the ADC will be multiplied as well. To fix these issues an external voltage reference is used instead. The external reference output is typically 2.048V with a maximum error of 2mV. This reduces the voltage division needed as well as the reference's potential error. The voltage divider is needed for the ADC to measure the voltage of the battery cells which are higher than the reference voltage of the ADC. The divider itself can create error by its resistors' values not being as specified. The max voltage divider error can be calculated with equation 2.5.2, using this equation we can see that using resistors with 0.1% error will add 0.2% error to the measurement.

$$E_{total} = E_{R1} + E_{R2}$$

Equation 2.5.2: Voltage divider error equation E_{total} is error of the divider, E_{R1} is the error of resistor 1, E_{R2} is the error of resistor

By looking at all the major sources of error that we can quantitatively predict we can find the total error with equation 2.5.3.

$$E_{total} = E_{ADC} * D + E_{divider}$$

Equation 2.5.3: E_{total}: Total error in reading, E_{ADC}: Internal ADC Error, D: Voltage divider, $E_{divider}$: Voltage Divider Error

$$E_{total\,=}\ (2.5*0.002mV+.002mV) * 2.5 + 0.002 * 4.2V = 25.9mV$$

Equation 2.5.4: Total Error

As shown in equation 2.5.4 the total error of the ADC is at most 25.9 mV which is far below our design requirement of 50mV.

3 Cost Analysis

3.1 Cost of Parts

Part	Part Specification	Cost/Unit	Quantity	Total Cost (prototype)	Supplier
Slave Module					
MCU	ATtiny261a	$0.40	1	$0.40	Digikey
Opto-isolation	PC817	$0.05	3	$0.15	Aliexpress
Fuse	15 A 1808 SMD	$0.09	4	$0.34	Aliexpress
Precision resistor	24.9K 0.1%, 75K 0.1%	$0.16	1	$0.16	Digikey
High tolerance passives	-	$0.05	1	$0.05	Aliexpress
Voltage Reference	LM4040AIM3X	$0.10	1	$0.10	Aliexpress
PCB	-	$0.80	1	$0.80	JLCPCB
			TOTAL	**$2.00**	
Mechanical Module					
Spacers	2mm 3M	$0.10	4	$0.40	Aliexpress
Male connector	2.54x25mm pin header	$0.14	1	$0.14	Aliexpress
Female connector	Pin header	$0.18	1	$0.18	Aliexpress
Locking washer	-	$0.01	4	$0.04	Aliexpress
18650 Holder	Quad bat holder	$0.92	1	$0.92	Aliexpress
			TOTAL	**$1.68**	
Master Module					
MCU	ATmega328pb	$1.26	1	$1.26	Digikey
Opto-isolation	PC817	$0.05	1	$0.05	Aliexpress
Precision Resistors	24.9K 0.1%, 75K 0.1%	$0.16	1	$0.16	Digikey
Colomb Counter	-	$1.91	1	$1.91	Digikey
PCB	-	$0.50	1	$0.50	JLCPCB
Buttons	Momentary	$0.04	5	$0.20	Aliexpress
Passives	-	$0.20	1	$0.20	Aliexpress
DC/DC converter	LM5017MR	$4.00	1	$4.00	Texas Instruments
Display	20x4 character LCD	$2.98	1	$2.98	Aliexpress
			TOTAL	**$11.26**	
Power Components					
Current Shunt	100A 75mV shunt	$3.75	1	$3.75	Aliexpress
			TOTAL	**$3.75**	
			GRAND TOTAL	**$18.69**	

3.2 Cost of Labor

$$\frac{\$71{,}166}{1yr} * \frac{1yr}{2087hrs} = \$34.33/hr$$

Equation 3.2.1

Equation 3.2.1 shows the average amount of hourly salary that an ECE graduate earns. Considering that we have 2 people for this project and we might work for 10 hours a week for 16 weeks; the total cost of labor can be shown using equation 3.2.3 below:

$$\frac{\$34.33}{1\ hr} \times\ 2.5\ \times\ 2\ \times\ 160\ \mathrm{hr}\ = \$\ 27{,}464$$

Equation 3.2.2

4 Schedule

Week	Logan	Daksh
10/05/18	Design Document	Design Document
10/12/18	Master Module Schematic and Board Layout	Practice programming the ATtiny using test codes
10/19/18	Solder proto board, order parts	Hardware validation
10/26/18	Tolerance Testing	Begin testing Interrupts and SPI communication
11/02/18	Manufacture Slave boards	Programming the Slave
11/09/18	Manufacture Master boards	Implementing and fine-tuning SPI for a robust communication protocol
11/16/18	Testing and validation of slave	Programming the Master
11/23/18	Testing and validation of master	Debugging and Testing
11/30/18	Mounting	Preparing for Mock Presentation and Demonstration
12/07/18	Final Paper	Final Paper
12/14/18	Final Paper	Final Paper

5 Safety and Ethics

The biggest safety hazard with our project is the extensive use of batteries. We use lithium ion batteries which are extremely dangerous when they attain high temperatures which can be caused due to higher ambient temperature, overcharging or discharging. At high temperatures the lithium becomes more reactive and can catch fire or even blow up. The lithium ion batteries can also experience a thermal runaway where a positive feedback loop is created leading the batteries to overheat or even explode. A battery management system is precisely created to monitor battery health and eliminate such unforeseen circumstances; a thermistor is connected to the batteries whose resistance varies as a function of the temperature. Hence the battery temperature can be monitored and controlled by monitoring the thermistor's resistance value.

In accordance to the IEEE code of ethics #1, through our project we strive to consider the safety and welfare of public in its highest regard, we ensure to disclose any factors that might endanger the public or environment [3]. A BMS is intended to ensure safe usage of lithium ion batteries. In addition to this, we also commit to be honest in stating claims based on the available data [4], IEEE code of ethics #3.

According to ACM code of ethics 2.7, it is important to foster public awareness about new technologies and its tradeoffs [5]. A more efficient and cost-effective battery management system will be extremely important in the electric vehicles industry. Such a battery management system will greatly influence the advancement of the electric vehicles by making them more cost effective. Hence a technology like this will greatly benefit the environment and the earth by replacing cars fueled by polluting and non-renewable sources. Hence our project can serve to be a crucial step in making electric vehicles more widespread and universal.

6 References

[1] “Starter Kit: WatchMon4 Expansion 14 x Longmon SoC% sensor,” *Batrium*. [Online]. Available: https://www.batrium.com/collections/starter-kit/products/kit-watchmon4-dinexp-14s-longmon-shuntmon2-500a. [Accessed: 15-Sep-2018].

[2] T. Zhuo, “Family of 7 rescued from Woodlands condo fire sparked by e-bike explosion,” *The Straits Times*, 20-Aug-2018. [Online]. Available: https://www.straitstimes.com/singapore/family-of-7-rescued-from-woodlands-condo-fire-sparked-by-e-bike-explosion. [Accessed: 15-Sep-2018].

[3] “IEEE Code of Ethics,” *IEEE Advancing Technology and Humanity*. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 04-Oct-2018].

[4] “IEEE Code of Ethics,” *IEEE Advancing Technology and Humanity*. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 04-Oct-2018].

[5] “ACM Code of Ethics and Professional Conduct,” *Association of Computing Machinery*. [Online]. Available: https://www.acm.org/code-of-ethics. [Accessed: 18-Sep-2018].