Automated Thermal Cell-Tester

By

Wonjun Lee (wonjunl2)

Daniel Songer (dsonger2)

John Stimpfl (stimpfl2)

Final Report for ECE 445, Senior Design, Fall 2021

TA: Stasiu Chyczewski

December 9, 2021

Project No. 7

Abstract

The danger that lithium-ion cells can pose, as demonstrated by electric vehicle fires and cell-phones spontaneously exploding, can be mitigated by using more informed battery management systems. A huge part of this is learning a cells individual characteristics, such as a cell's capacity and charge/discharge curve, which change drastically with changing temperatures. Thus, we propose a product that tests cells for the aforementioned characteristics by discharging and charging the battery in a specified way. Because of lithium-ion cells drastically different behavior at different ambient temperatures, we also include a thermal chamber for testing the cells across a range of thermal environments. Our final product was unable to perform many of the desired tasks, but we were successful in discharging a battery at a range of low currents and also controlling the temperature inside a thermal chamber built for a Samsung INR 30 Q lithium-ion cell. We also successfully built an interface between our main controller to the user program for communicating data, errors, and test control. There were various failures in our design which we will address in detail including the specific wiring issues in our printed circuit board as well as the user program's interface with the main controller. Although this product never reached its full potential in this time-frame, it is deserving of a second iteration through the design process. We found that this project was simple enough to implement and would provide a more safe environment for working with lithium-ion cells.

Contents

1 Introduction

Battery applications have been increasing in popularity and stories of terrible battery-related chemical fires along with them. Unfortunately, the causes of these fires is often not discovered until after the fact. The official reports filed after such an event often cite issues such as manufacturing defects in the separator, impurities introduced at production, thermal runaway, design flaws that cause unnecessary current drain, overworking the battery and improper use, and a few others [1]. Not only are these fires dangerous, but they are costly. Like in the case of an entire electric vehicle catching fire, not only is the car destroyed, but also the fire may take as much as 40 times the amount of water an internal combustion engine car fire would take to put out due to the nature of the battery's chemical fire [2]. This has increased some of the stigma in regards to working with batteries which slows the transition towards an electric future. A future that involves many forms of huge energy storage banks made up of lithium-ion batteries that help incorporate more solar and wind power into the grid as well as enable technologies like electric vehicles [3]. Therefore, using lithium-ion batteries can be quite dangerous therefore limiting entry into the industry of battery-powered applications or, at a worse extreme, causing major harm and damage to any individual or application. In many cases, the latter can be avoided by simply having more insight into the characteristics and expected behavior of the battery pack's individual cells. With the proper behavior defined for the battery, the user could theoretically know when a battery is acting out of line with the expected behavior and shut down the application before any fires break out. The reduction in lithium-ion related fires and accidents could possibly result in a broadening of the industries applications as well as ensure safe and efficient use in increasingly popular applications such as electric vehicle and power storage.

To achieve safer use of lithium-ion batteries we are going to provide the user with the necessary insights of an individual cell in any given pack to be able to know when to shutdown safely in order to avoid any destructive chemical fires or harmful battery behavior. To do this we are going to build a cell-tester that is specifically suited for testing a cell's characteristics through different tests and varying thermal conditions. In this way, we will be able to characterize a cell in its application's normal operating conditions and provide the baseline to which all the cells in the pack will be compared. Providing the information from the different tests to the management system responsible for the battery pack will allow the early shutdown desired. Another benefit of having all this information about the cells being used is that they can also be optimized for longevity and efficiency.

1.1 Solution and High-Level Implementation

To accomplish the aforementioned cell testing under various thermal loads, we proposed the automated thermal cell-tester which has the necessary power electronics for battery current drain and charge, as well as thermal control inside a thermally isolated environment. Specifically, our device is able to control different levels of charge and discharge according to a given test while also managing the temperature inside the thermal unit to provide the thermal testing functionality. The device also includes a GUI that helps the user see the batteries current performance as well as analyzes data from previous tests for cell characterization. That is, once a test is performed, the GUI saves and loads all the data transmitted during a test and then analyzes it for various characteristics according to the type of test being analyzed. Our high-level requirements for this device are as follows:

- The device will perform the three necessary tests to accurately characterize a lithium-ion cell's capacity, charge/discharge curve, and RC polarization characteristics which provide a great deal of insight into battery performance.
- The web application will receive results from tests of specific cells and use those results to display the current battery characteristics as well as the cell's performance over time.
- The device will control a thermal unit that will to maintain a constant temperature from 15 °C - 35 °C to within 5 °C throughout any given test allowing for characterization of the cell over a range of thermal conditions.

Thus, we created the block diagram presented in figure 1 as an answer to these design requirements. We made the individual subsystems capable of unique tasks for easy debugging as well as modular builds.

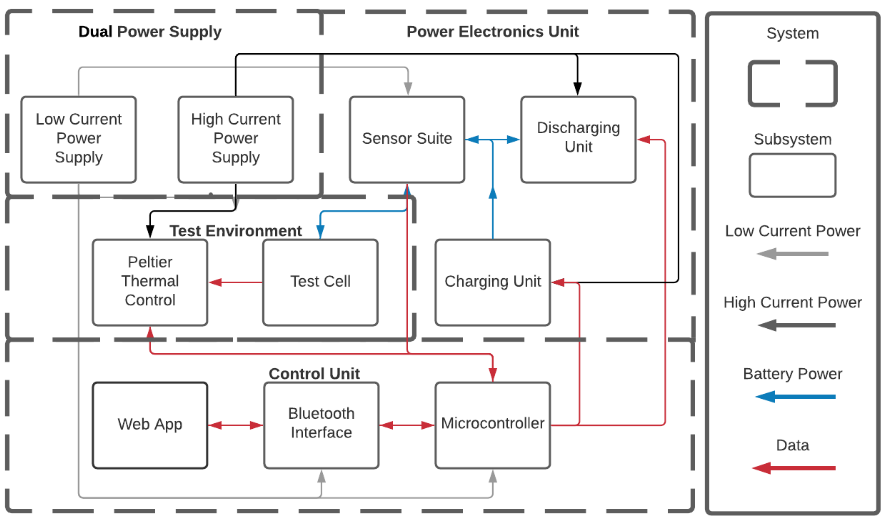

Figure 1: The automated thermal cell tester block diagram.

From top-left, the dual power supply is responsible for handling different current demands, high-current power supply supplies the power electronics while the low-current supply powers the digital portions of the circuit. The power electronics unit has three subcomponents: 1) the sensor suite - responsible for sensing the current and voltage of the cell, 2) the discharging circuit - capable of discharging a wide range of currents at varying precision, 3) the charging unit - single cell 5.5 A charger. The test environment is simply a thermally isolated box that includes a battery tray and Peltier devices which generate the thermal separation according to polarization. Finally, our control unit includes a user interface (web application), the bluetooth interface, and the microcontroller responsible for managing all these low-level electronics as well as communicating with the user interface.

2 Design

2.1 Design Procedure

In this section we will outline the general design choices made and any relevant equations used as well as general circuit outlines.

2.1.1 Discharging Circuit

The discharging circuit design is principally concerned with maintaining large and small constant currents out of the battery. Since current during a given test is required to remain constant regardless of the voltage on the battery, the circuit was designed to be a voltage controlled current source. To accomplish this, a BJT was chosen. Since the voltage from the base to emitter of a BJT remains constant, the base current of the BJT can be controlled proportionally to an input voltage via an intermediate resistance. Since both large and small currents are required, it was decided that CMOS logic would be implemented to allow the selection of two separate resistors sized for large and small currents respectively. Thus, higher accuracy can be obtained for small currents by decreasing the slope of the circuit's IV curve in while operating in low current mode.

2.1.2 Charging Circuit

The charging circuit design relied heavily on the datasheet [4]. We wired a prepackaged IC straight into our board. Although our control scheme was able to talk to the IC at certain points, we never actually were able to demand any charge current to any cells, in fact, communication with the IC was minimal. Because this IC is also built for discharging the battery, we do not believe this particular implementation is the proper application, more on this in section 2.2.2.

2.1.3 Thermal Unit

The thermal unit was designed around the solid state Peltier coolers. These devices produce a linear relationship between heat pumped through the device and temperature difference at the hot and cold sides when driven with a constant current [5]. By reversing the current with a relay wired as an H-bridge, the direction of the heat flow could be reversed allowing these Peltier coolers to heat or cool the device. The constant current source was reused from the discharge circuit to drive the Peltier coolers as a way to streamline debugging during implementation. In theory a compressor system is much more effective than the Peltier coolers, but has the significant drawback of increased complexity and unidirectional heat flow. The use of the Peltier coolers significantly reduced the complexity of building a thermal unit designed to both heat and cool. To make the unit as effective as possible, decreasing size was made a priority. As we decreased the size the efficiency of the Peltiers went up because there was less work done in cooling the entire unit 1 °C because there was simply less volume of air to cool.

2.1.4 User Interface

The graphical user interface (GUI) was responsible for four main things: alerting the user to any errors, test management (serial interface), data caching, and finally data loading and analysis.
We initially chose to use a web application that communicated via bluetooth, however, as the physical design of the product changed, so did our requirements for semi-long range communication. We also note here that

the GUI realizes the cell characteristics after loading in data stored from a test. For battery capacity, which is measured in Ah, we are interested in calculating the current drained from the battery over time. We use equation (1) in our algorithm to determine a batteries capacity.

$$\begin{aligned} 1\text{A} &= 1\text{C/s} \\ 1\text{Ah} &= 3600\text{C} \end{aligned} \tag{1}$$

Thus we need only multiply each current data point by the interval of the data message interrupt, which is 100 ms. Finally, if we multiply by the amount of hours per second in an hour and add each point, we have added up all the Coulombs that the battery output and are left with the battery capacity in Ah. See equation (2).

$$total\text{Ah} = \frac{1\text{hr}}{3600\text{s}} \times 0.1\text{s} \times \sum_{k=0}^{t_f} i_{measured} \tag{2}$$

2.1.5 Sensor Suite

The design of the sensor suite was focused on isolation and accuracy. Given the danger posed by lithium ion cells in an uncontrolled discharge, the input impedance of some prepackaged current and voltage sensor could not be relied upon to safe guard the system. This prompted the design of our own sensor system, leveraging op-amps with known large input impedance to buffer voltages off of the terminals of an appropriately large shunted shunt resistor. To improve the accuracy of current readings, the voltage difference between the shunt resistor terminals was boosted before measurement.

2.1.6 Dual Power Supply

The design of the dual power supply was intended to isolate sensitive digital electronics from power intensive, and less voltage sensitive, power electronics. By separating these elements onto their own 5v power supplies, the reliability of the digital electronics was bolstered. This separation also allowed for the use of less expensive voltage regulators with lower ratings, thereby saving costs on an already over budget project.

2.1.7 Firmware

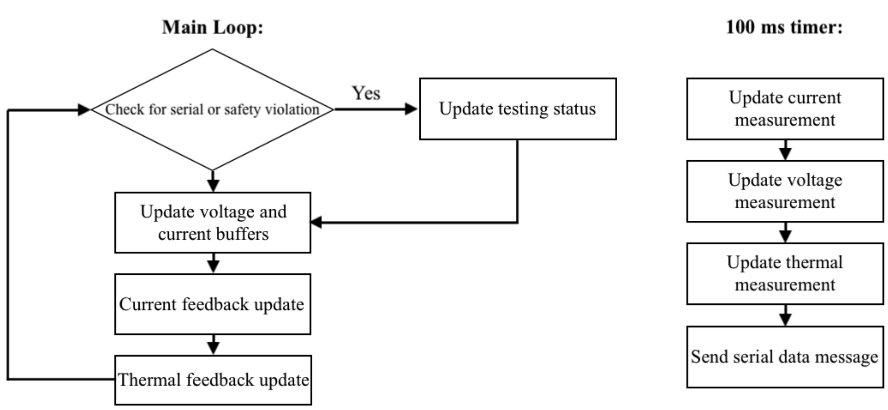

Figure 2: Main loop and ISR flowchart.

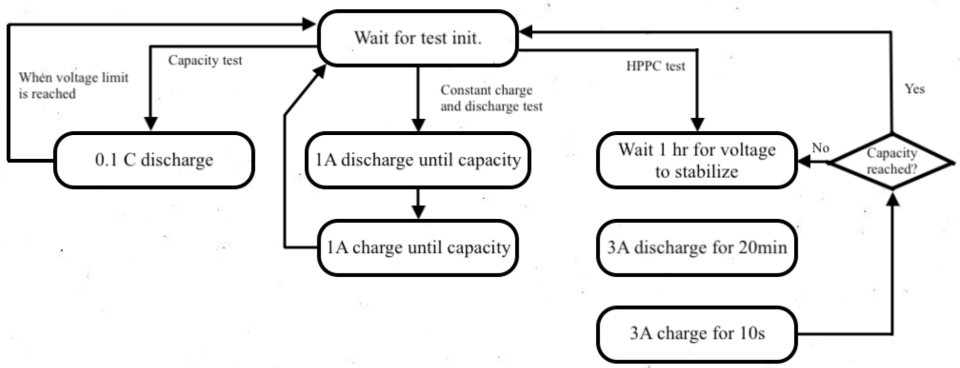

Figure 3: Testing routine finite state machine.

The flowchart in figure 2 shows how straightforward it was to design firmware for this device. The firmware can be described by a fairly simple routine that updates measurements, ensures safe operation and maintains a data structure that can track the testing routines through a finite state machine (figure 3) without wasting essentially any time away from the main loop/measurement updates. There also exists a 100 ms timer interrupt that averages all the measurements taken from the main loop and sends that data over serial to the GUI. One essential routine of the controller was the `safety_stop()` function which first ensured the charger and discharger were not driving or draining current, then opened the relays, disconnecting the battery entirely from the circuit, finally the routine sent an `ERROR_MESG` to the GUI containing a snapshot of the error state of the battery.
The code was so straightforward that we quickly designed but never implemented a feedback P controller for both current and temperature. In the former, the controller executed a more precise current control when the current being measured varied in small amounts from the desired current. For the latter, power output could easily be stepped up or down according to what temperature was being detected. Actual implementation of this temperature feedback loop proved difficult as we were unable to attain thermistors in time due to a lack of availability in sourcing them.

2.2 Design Details

2.2.1 Discharging Circuit

As discussed previously, the discharge circuit functions as a voltage controlled current source. It takes advantage of the current gain and constant base emitter voltage of a BJT to select a discharge current independent of a given cell's charge state. As discussed previously, by applying an input voltage over a given resistance, the current can be controlled by the input at a rate inversely proportional to the resistance. Using CMOS logic, two different resistances can be selected, allowing for a high resistance 'low current mode' and a low resistance 'high current mode'. The full circuit topology can be seen in figure 4. It should also be noted that figure 4 includes a relay to isolate the circuit from the test cell when not in use. With the input voltage separated from the base of the transistor by a resistor and a MOSFET biased in saturation, the full expression for the discharge current (when the input is greater than the base emitter voltage of the BJT) is as seen below in equation (3). The input voltage in this case was supplied by a digital-analog-converter

(DAC) buffered with an op-amp rated for 250 mA (not pictured in figure 4).

$$I_c = (\frac{V_{controlled} - V_{Base}}{R + R_{DS}})\beta \tag{3}$$

In selecting a BJT, care was taken to find a robust device which could handle the high current and power

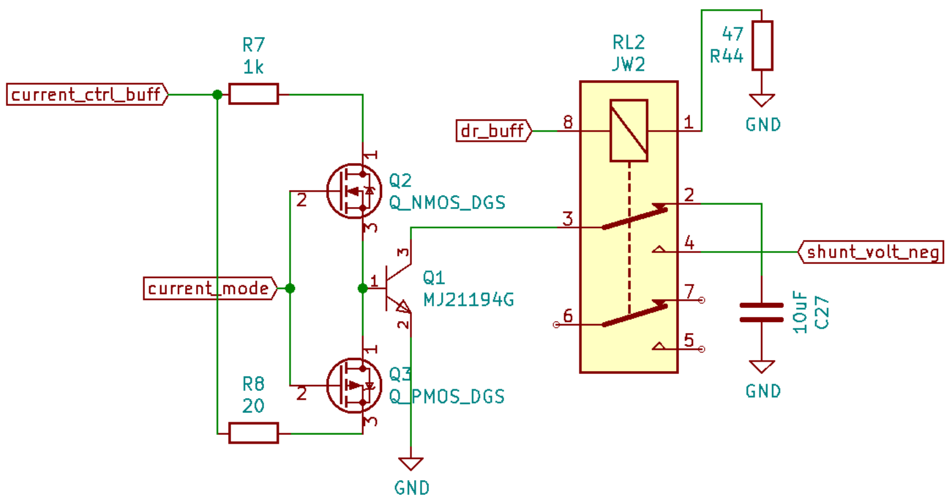

Figure 4: Voltage input is labeled current ctrl buff, Discharge flows from shunt volt neg to ground.

dissipation requirements of this application. The ON Semiconductor MJ21194G was selected for its max current rating of 16 A and max power rating of 250 W [6]. Additionally this BJT comes in a package with a large metal casing, ideal for radiating the heat generated by the large power dissipation. This BJT far exceeds the 5 A and 25 W dissipation per the original circuit requirements. Few other BJTs meeting these requirements were available, and of those only the MJ21194G was the only option with a reasonably large DC current gain (∼50). Lowering this current gain would tax the other components of the circuit requiring more specialized parts. The resistances chosen for the high and low current modes were 20 Ω and 1 k Ω respectively. In the high current mode, this places the max target current of 5A near the center of the voltage range of a 5 V DAC (see figure 14 in appendix A). In the low current mode, this places the max current for low current tests (ex. capacity test) of approximately 200 mA at the upper end of a 5 V DAC (see figure 15 in appendix A).

2.2.2 Charging Circuit

Figure 5 contains our implementation of the MAX7797EFD+, the charging IC. This lithium-ion charger chosen for our design, we later determined was being used in the wrong application. We realized that this charger was not necessarily made for providing an ideal current source which is more along the lines of what we needed for this particular application. Specifically, this charging IC was built to charge a single cell as well as discharge it and power the rest of a circuit. Due to this, we realized that this IC was actually meant for implementation as sort of a single cell management system and would not have provided us the desired current but rather would likely have tried to start an entire charging routine of constant current then constant voltage. Because of this we actually believe that the designs we have for this device are almost superfluous and hold little water when thinking about another iteration of this design.

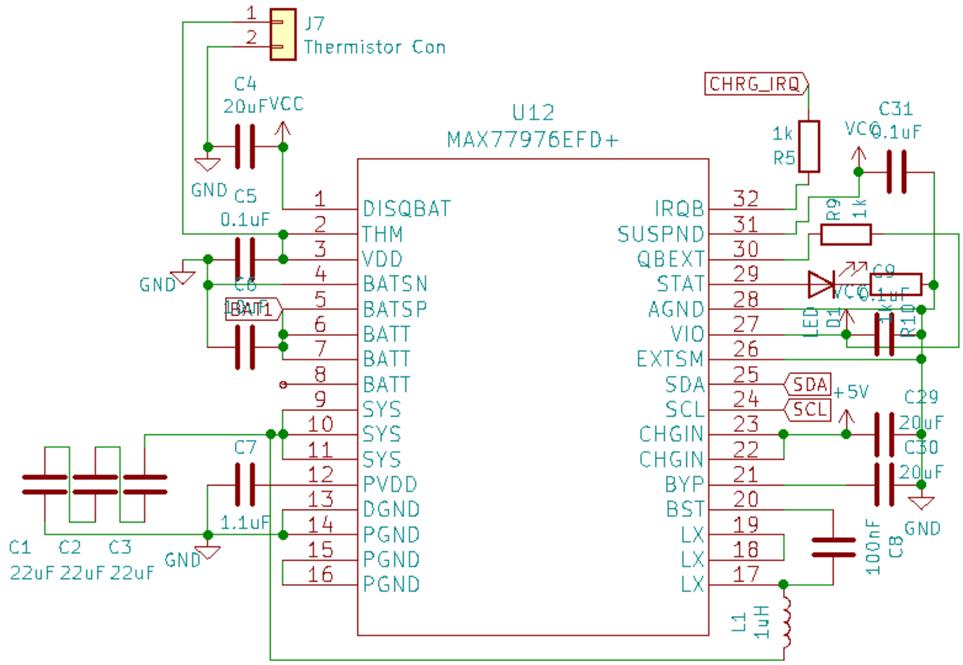

Figure 5: In-circuit implementation of the charging IC which shows the necessary inductor as well as the diode that indicates the charger is on.

2.2.3 Thermal Unit

As discussed previously, the thermal unit is designed around the use of two Peltier coolers. These devices were driven with an identical circuit to the high current mode of the discharge circuit. Due to the relatively high voltage across these Peltier devices when in operation, this circuit was driven at 12 V as opposed to the standard 5 V of the rest of the project. The bulk of the design work on the thermal unit concerns the thermal circuit present in the physical design of the project. The construction of the thermal unit itself (embedded within the thermal chamber) consists of two flat Peltier devices mounted on a large exterior heat sink. This heat sink is rated for 0.90 °C/W at 250 LFM airflow [7]. Smaller heat sinks were mounted to the interior side of the Peltier devices these are rated for 2.50 °C/W at 200 LFM airflow [8]. The Peltier devices themselves also experience Ohmic losses with a resistance of 4.43 Ω [5]. By examining the linear temperature characteristics of the Peltier devices as seen in figure 6, the specifications above can be used to create an equation relating the depression in the air temperature inside the thermal unit to the rate at which heat seeps in to the unit.

$$\Delta T = -\frac{P_{in}(2.5(C/W))}{2} + g(\frac{P_{in}}{2}, I) - 0.9(C/W)(2I^2 4.43\Omega + P_{in}) \quad (4)$$

In the above equation (4), g(x) is the temperature difference across the Peltier device as a function of heat pumped and the current through the device. Making a few assumptions a reasonable expectation of temperature depression can be calculated. Estimating that the power into the thermal unit is 7 W and the Peltier devices are run at 1.44 A, the total temperature depression below room temperature can be found to be approximately 8.4 °C. With the target temperature of 15 °C being 7 K below room temperature, the

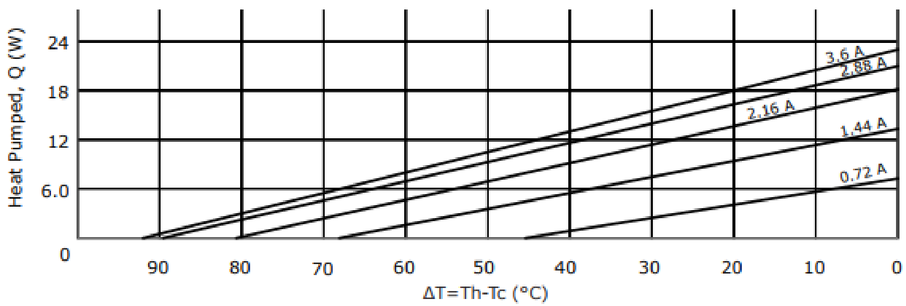

Figure 6: Family of curves for Peltier device thermal performance.

thermal circuit is well equipped to handle the desired temperature depression. The heating mode of the thermal unit is far less interesting and had minimal design effort. The target max temp is 35 °C. Which is only 13 °C above room temperature. Considering that Ohmic losses work in favor of heating the thermal unit, there is no question that 35 °C is achievable with a reversed current. For the heat sinks of the thermal unit to function at their reported specifications, they require airflow. The exterior heat sink was equipped with two fans, each capable of pushing 300 LFM of airflow. These were able to more than satisfy the 250 LFM requirement of the exterior heat sink within their cross-sectional area. These fans were mounted to provide airflow directly over the radial fins of the heat sink for greatest effect. The interior heat sinks were equipped with a single 300 LFM fan. This fan circulated air in a loop throughout the thermal chamber, including past the internal heat sinks, satisfying their airflow requirement.

2.2.4 User Interface

The GUI was required to perform four main tasks which resulted in supporting five different pages.

- Main menu and text data page
- Error page
- Test management page
- Live graph data page
- Test analysis page

The most interesting portion of the GUI code, however was the use of the multi-threaded library to support the serial interface for data updates while the GUI update functions also ran so as to not crash the app altogether. Therefore, all the live data coming in at 100 ms apart could be handled in a timely manner while also keeping the renderings of the app refreshed and live.

Finally, the serial interface itself is made up of four messages. These messages are handled in each their own way according to the `handle_message()` function. The `ERROR_MESG` and `DATA_MESG` contain 10 bytes of data (+1 byte for the message type) which were handled through different conversion functions to translate the format that the controller transmits the values, such as 5 V as 4905, into the appropriate floats. The other two messages, `START_MESG` and `STOP_MESG` contains only one byte in the start message, and that is to communicate to the controller what type of test is to be started and at what temperature.See appendix A for figures that show the live data feed and the test management page of the GUI.

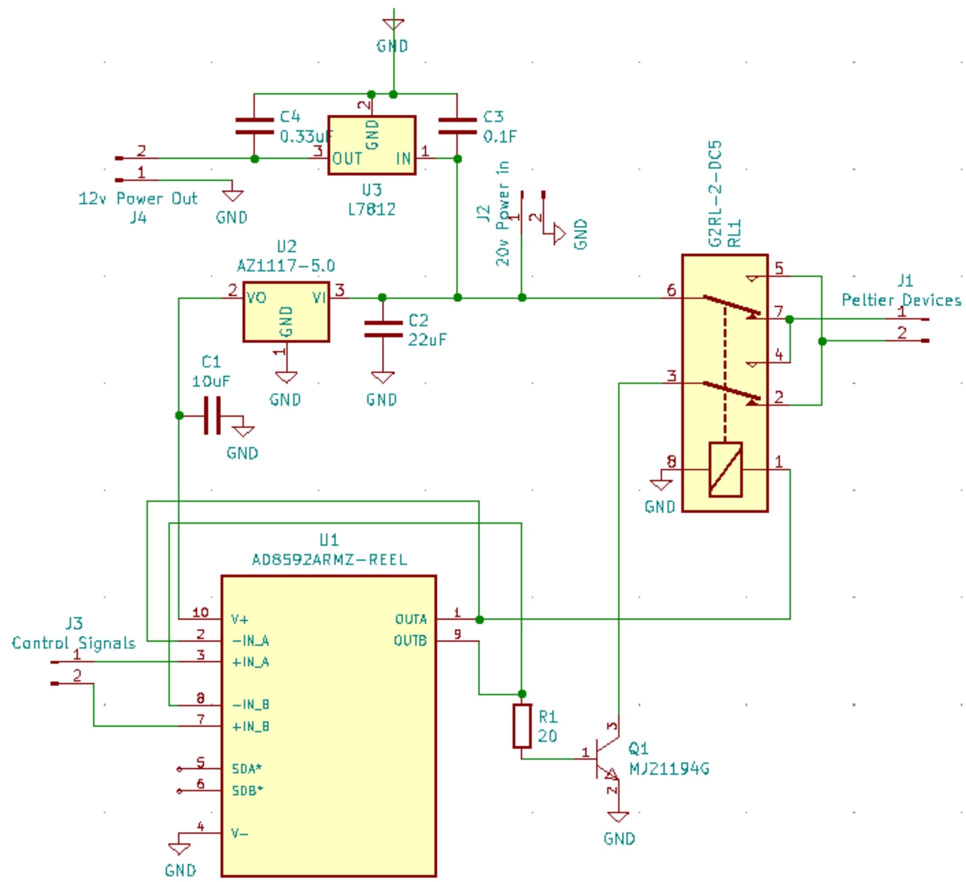

Figure 7: Schematic of the thermal unit drive circuitry.

2.2.5 Sensor Suite

The design of the sensor suite is fairly simple. First, op-amps are used to buffer voltages off of the positive and negative terminals of a low-resistance shunted power resistor connected directly to the fuse that is connected to the positive terminal of the test cell. The positive terminal buffer is then directly fed into an ADC reference to the same ground as the test cell to give its open-circuit voltage. The current through the sensor is slightly more complex as it relies on the small voltage difference generated by current flowing through the shunt resistor. The buffered positive and negative terminal voltages are used as inputs to a differential amplifier referenced to 2.5 V. This will result in an output voltage proportional to the current above 2.5 V for a current out of the test cell and a output voltage below 2.5 V for a current into the test cell. The gain of the amplifier can be adjusted to use the full range of a 5 V ADC at a given max current allowing for increased accuracy. The schematic of the sensor suite is shown in figure 8. The gain of the differential amplifier is given by

$$Gain = \frac{R_4}{R_2} = \frac{R_5}{R_3} \tag{5}$$

The ideal gain can be found with the max expected current and the resistance of the shunt resistor.

$$Gain = \frac{2.5v}{I_{max} R_1} \tag{6}$$

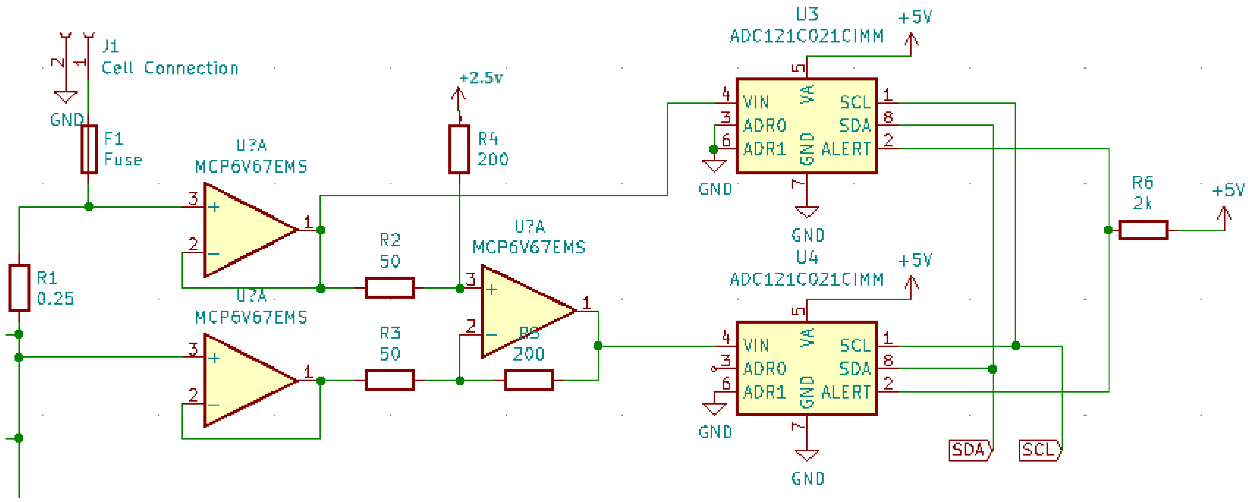

Figure 8: Schematic of the sensor suite.

The schematic in figure 8 below reflects the original shunt resistor and gain values. During implementation the max current was revised down to 3 A and the shunt resistor was changed to 50 mΩ resulting in a revised gain of 16.67. This was rounded to 15 to make use of standard resistor values.

2.2.6 Dual Power Supply

The dual power supply as a whole is rather simple. It is no more than two separate voltage regulators with supporting capacitors for voltage stabilization. The high power voltage regulator is rated for 7.5 A [9]. While the smaller voltage regulator is rated for 1 A [10]. These sizes were chosen based on maximum expected current draw. The larger regulator at time of design had an expected max current draw of 5 A (during a charging pulse). The smaller voltage regulator had an expected current draw of less than 250 mA. The schematic of the dual power supply is seen in figure 9 below.

2.2.7 Firmware

The main concern in design of the controller code was timing. Specifically, we were concerned with the main loop being performed in less than one-fourth the time of the timer interrupt (100 ms). That is, we wanted four measurements of voltage, current and temperature to be made so that averaging these measurements in the ISR meant something. Finally, because of the timing, we avoid any floating point operations in the entire script. One particular algorithm with well-defined timing would be our low-pass filters on all measurements to filter out noise in our ADCs, in planning for this we ensured that the low-pass filter was implemented in powers of 2 so that bit-shifting could be performed instead of division, see equation (7).

$$x^{k+1} = (7 * x^k + x_{meas}) >> 3 \tag{7}$$

One timing flaw, that never got verified within the script timing constraints due to lack of measurement data, was the measurement average that took place in the ISR. We believed that with the ISR only firing every 100 ms, we would hardly be able to notice any issues with the device. However, it can easily be reasoned why this delay in taking more measurements from the main loop can result in less accuracy of our device.
The next thing we considered in our design was that the main controller would track progress through a

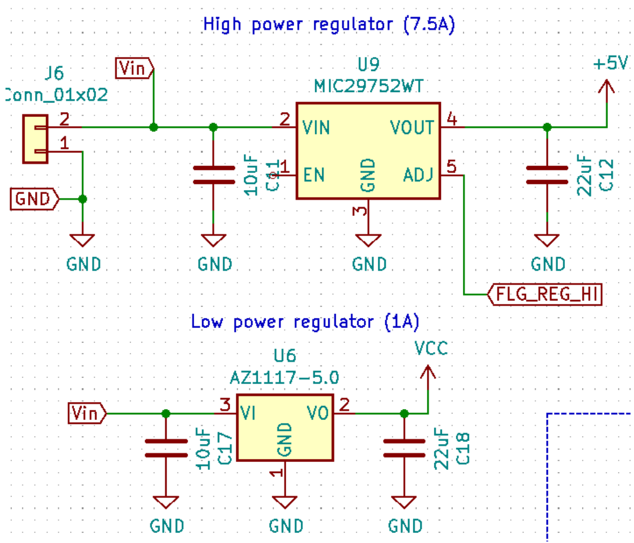

Figure 9: Schematic of the dual power supply.

type byte0	time [31:24] byte1	time [23:16] byte2	time [15:8] byte3	time [7:0] byte4	voltage [15:8] byte5	voltage [7:0] byte6	current [15:8] byte7	current [7:0] byte8	temperature [15:8] byte9	temperature [7:0] byte10

Figure 10: Data contained in a message that the controller transmits.

given test. This was because we didn't necessarily want the GUI to change the testing state asynchronously and also because the only thing needed for tracking is to track time and a computer representation of the testing finite state machine. That said, the design of the computer representation of the finite state machine is implemented as a class in C++ because we used the Arduino®IDE. Thus, it includes a timing construct for knowing when to switch to a new state as well as a `type` and a `stage` variable for tracking what stage we are in for a specific type of test.

Finally, the controller included the implementation of four different communication interfaces for low-level peripherals

1. I^2C peripherals
2. SPI interface
3. ISP (JTAG) interface
4. USB port

Which were used as a means to communicate with the ADCs and charging IC (I^2C), the DACs (SPI), the main controller programming port (ISP), and the GUI serial interface (USB port). The most noteworthy of these is the serial interface which implemented four different message types in which both the `DATA_MESG` and the `ERROR_MESG` contain data which provide a snapshot of the state of the cell in the form described in figure 10. The other two messages, `START_MESG` and `STOP_MESG`, are received by the controller and contain one byte and no data respectively and are easily handled due to the small amount of data contained inside each message.

3 Verification

3.1 Discharging Circuit

Because the final form of this module was to be controlled by a DAC that was controlled by the microcontroller we never verified the full implementation. However, we were able to emulate the voltage input to the power BJT by using a mechanical potentiometer. This controlled the current as high as 1.8 A draining out of the battery. In this particular verification scheme we also emulated a lithium-ion battery by placing AAA batteries in circuit. With this we were able to draw the desired current as well as see a voltage across the power BJT. Throughout this test we varied the potentiometer voltage and saw the current increasing and decreasing proportionally to how much we adjusted the potentiometer. This demonstrated that we were indeed able to control the current drain from the battery by supplying a low voltage input to the base of the power BJT.

3.2 Charging Circuit

Due to the lack of control in our final implementation we never truly got to test or verify that this charging circuit was ever even capable of supplying the desired current to a lithium-ion cell. Specifically, we only ever received an acknowledge signal and wrote into setup registers but never actually demanded any current. Thus, we cannot say, with any certainty, how well this IC worked and if it even provided any current at all.

3.3 Thermal Power Unit

To verify the correct operation of the thermal unit and the fulfilment of the third high level requirement, several tests were performed to plot the temperature of the thermal unit under various operating conditions. The purpose of the test was to determine if the thermal chamber is capable of operating over the full temperature range specified in the high level requirement. Three cooling runs were performed at 1 A, 2 A, and 3 A running through the Peltier devices. One heating run was performed at 2 A. The temperature inside the thermal unit was taken via infrared thermal sensor once every 2 minutes. The results are shown in figure 11 below. As expected there were no issues warming the thermal chamber. The 35° C target was well exceeded and the power had to be disconnected earlier than originally planned. The cooling runs varied in their effectiveness. The 1 A run had a fairly consistent temperature drop before leveling off at around 22 °C, 4 K below the ambient air temperature during the test. The 3 A run saw rapid temperature declines early in the run which leveled off before the temperature actually began to rise. This is likely to due to mounting Ohmic losses when running at the higher current. The 2 A run was much more optimal than the other cooling runs. The temperature continued dropping even at the planned end of the test, so the test was extended until the temperature leveled off at around 16 °C. This is within the allowed 5 °C temperature range of the required 15 °C. Considering the ambient temperature was significantly above normal room temperature during the test, under more favorable conditions the required 15 °C is achievable. Though it was not recorded in this test, the lowest temperature recorded in the thermal chamber was 13 °C. The control aspect of the third high level requirement was, unfortunately, not able to be tested as a result of the loss of our microcontroller.

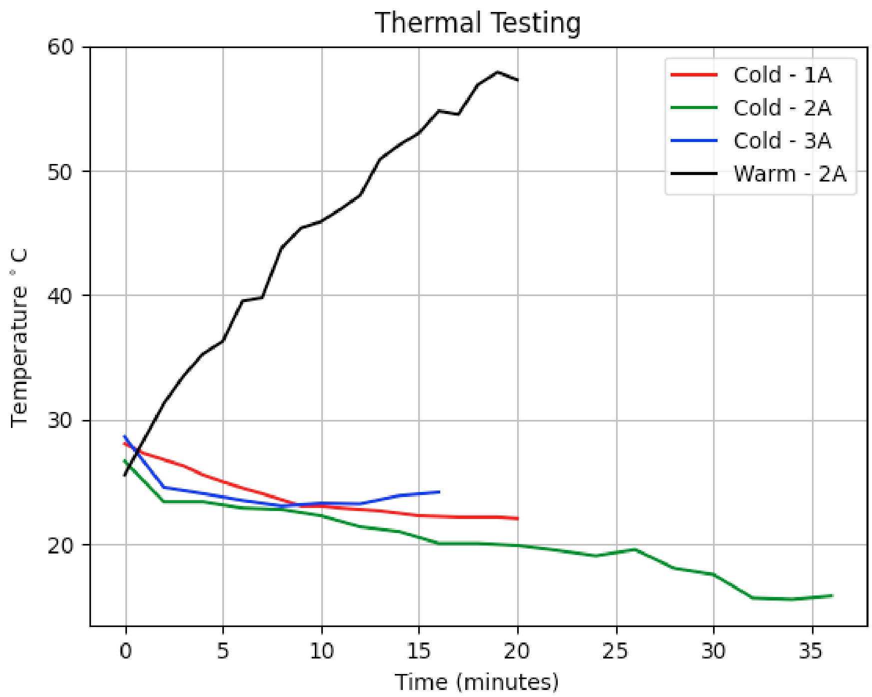

Figure 11: Thermal unit test runs.

3.4 User Interface

Early in the implementation of the GUI we experienced a lot of crashing and were unsure that this would be sustainable for proper functionality. We discovered that creating a separate thread for handling our serial interface would essentially be all we needed to do in order to experience no more issues whatsoever with the stability of the GUI. To verify that the serial interface functionality was working with the GUI, we designed a bit of controller code specially to emulate and handle any interactions with the GUI. That is, we used simulated cell data and transmitted it every 100 ms (often faster) as well as blinked LEDs when certain types of messages were received on the controller from the GUI. This verified that messages were being sent and received properly.

3.5 Sensor Suite

The sensor suite was reliant on a series of three op-amps and two ADCs. All of the op-amps experienced a failure and none of the circuit was able to be tested. The op-amps were rendered inoperable by an active-low shutdown pin which was not tied to a logical high signal on the PCB used for the final assembly. This caused the op-amps to behave unpredictably. Due to how grossly over budget this project was, only two ADCs were purchased. Of the two ADCs purchased, one was broken during assembly. The other was unable to be communicated with after we lost our microcontroller.

3.6 Dual Power Supply

The dual power supply was not verified after the loss of our microcontroller. Enough components were broken off the main board after this point that the capabilities of the dual power supply were immaterial to the overall performance of the project.

3.7 Firmware

The ATmega32u4 as implemented on our board was unable to function as a result of an HB pin which needed to be tied to the reset line. This could not be corrected as the pins were inaccessible once the microcontroller was soldered to the board. We attempted to replace the microcontroller with an Arduino®board, but were unable to when the board unexpectedly broke.
Unfortunately, due to the flaws mentioned in the design of our PCB, we ended up implementing a good amount of our board (including the controller) on an exterior breadboard. Once we did this, however, we were not careful enough in our design and we actually destroyed an Arduino®Mega in the build process. This meant that we had no controller left to implement our class-based testing structure and therefore very little true testing code ever got tested and verified. We did manage to verify that the serial and I^2C interfaces were indeed working as expected as well as ensure the timing of the device would be satisfactory for our purposes. That is, we tested the main loop timing to be < 25 ms by toggling an LED every 400 passes through the main loop and starting and stopping a handheld timer on these toggles. We found that one round of toggles was took much less than 10 seconds and therefore could confirm the timing of the user loop. The timer interrupt was verified similarly although we had no reason to think it would ever fail or vary much from 10 seconds exactly every 100 passes through the ISR.

4 Costs

4.1 Parts

We found that parts were often difficult and expensive to source. Because of this, we largely designed with the parts that we could source easily and this seemed to come with certain drawback as well. In general, the following table includes our costs for this project, including any testing equipment we also had to purchase.

Table 1. Parts cost analysis.

Items	**Cost**
Bluetooth Module (Amazon; HC-05).	$ 10.89
Microcontroller (Digikey; ATmega32u4).	$ 5.28
Power Resistor (Digikey; RSB-500-50).	$ 26.16
2 x Digital to Analogue Converter (Digikey; TLV5618-EP).	$ 31.52
Battery Charger (Digikey; BQ24179).	$ 6.36
Analogue to Digital Converter (Digikey; ADC121C021).	$ 49
4 x Amplifier (Digikey; MCP6V69).	$ 10.12
2 x Diodes (Digikey; AZ1084C)	$ 0.74
5 x N-Channel MOSFET (Digikey; VN10KN3)	$ 2.85
5 x P-Channel MOSFET (Digikey; TP0606N3)	$ 5.25
3 x 12v Regulator (Digikey; 497-12406)	$ 2.43
MicroUSB Port (Digikey; WM11262CT)	$ 1.46
3 x Relay (Digikey; 255-1116)	$ 11.28
2 x Interior Heat Sink (Digikey; 294-1155)	$ 11.58
Exterior Heat Sink (Digikey; 345-1052)	$ 12.70
2 x Peltier Coolers (Digikey; 102-4687)	$ 78.18
3 x Fan (Digikey; P19863)	$ 20.94
1 x Infrared Thermal Sensor (Home Depot)	$ 38.95
3 x Samsung Li-ion 30Q INR (M & A BD Electronics)	$ 29.68
1 x Power supply unit (Amazon)	$ 69.99
Total	$ 425.36

4.2 Labor

Aside from weekly meetings for an hour, we estimate that each member of this team put 15 $\frac{\text{hrs}}{\text{wk}}$ for 14 weeks and 30 $\frac{\text{hrs}}{\text{wk}}$ for 2 weeks. We average our hourly costs to be about $ 40 / hour and calculate our final costs as in equation (8).

$$C = 3 \times \$40 \times (15\frac{\text{hours}}{\text{week}} \times 14\text{weeks} + 30\frac{\text{hours}}{\text{week}} \times 2\text{weeks}) \tag{8}$$

The total labor cost, C, is: $ 32,400.

5 Conclusions

In this section we will summarize the main conclusions of our project. We include discussion on the successes, the failures, any ethical concerns that were raised, and any future work we would like to do regarding this project.

5.1 Accomplishments

In this section we highlight the thermal power electronics, the discharging circuit, and the user interface. These three things were the main portions of the project included in our demonstration.

The thermal power electronics worked almost better than expected. We originally gauged the unit to be able to cause $\pm$ 10 °C deviations from the ambient temperature but upon testing (see section 3.3) we found that we could cause that amount of deviation and a bit more for cooling. The thermal unit was so efficient while heating, in fact, that we had to limit the power consumed by the thermal unit in order to hit our goal of 10 °C deviation from ambient temperature.

The discharging circuit was also a success. We were never able to supply digital-to-analog voltages from 0V-5V but we were able to control current drain from a AAA battery via a mechanical potentiometer input into the base of the power BJT controlling battery discharge. Thus, the only failure of this portion of the device was the control inputs from the on-board DACs.

The final success in the project was the GUI. This portion also seemed like a failure at first because it was so lacking in what we originally envisioned it to be, however, by the end of the project we were able to redesign part of the GUI to work with an entirely new communication protocol (serial) and even implement the necessary multi-threading to avoid any crashing of our user program. The serial interface is in itself a well-designed success, but its most notable manifestation is in the live graph data feed. This page on the GUI updates two graphs, voltage and current, with the last 100 data points received from the controller in a `DATA_MESG`. This part of the GUI was perhaps the most aesthetic part of the whole project and made it seem like full integration would have been possible if it weren't for such a dramatic loss of control leading up to the demonstration.

5.2 Uncertainties

The main uncertainties which we would have to confront immediately before pursuing this project further, would be the charging IC, the battery isolation fault, and the firmware.

The main question we would have to address if we approached this problem again, would be the charging IC. This IC was made for single cell tablets and other smart devices. That is, it was made for charging and discharging a battery. This is a red flag and furthermore an uncertainty because this means this IC is built for battery management and wasn't actually built for supplying steady current exactly when asked for it, in other words it is far from an ideal current source. Thus, our uncertainty lies in the fact that ideal current sources are hard to find and seldom, if ever, fit on a 100 mm $\times$ 100 mm PCB.

We were unable to find the issue that caused what we determined to be an isolation fault from the battery. We believe it had to do with some of the wires and signals that had been broken out of the board when certain elements began to fail in the build process.

The final uncertainty, and perhaps the biggest, is the firmware. Without much control working, and virtually no code truly implemented, the firmware is a relatively large black box that we do not know much about. The only thing certain about the firmware is that its subroutines have all been verified individually in tests ran under ideal conditions for verifying that specific functionality (e.g. ADC reads of 2.5 V supplied by a DC power supply).

5.3 Ethical Considerations

This project in its entirety is motivated by helping people work safely with lithium-ion batteries. As such many ethical and safety considerations were made over the course of the project. The first ethical consideration we made was to read the relative documents provided by the course that were required for us to read in order to work with lithium-ion cells, this meant that we also followed the guidelines for handling and emergencies that can be found in *Safe Practice for Lead Acid and Lithium Batteries* [11]. The next ethical decision we made was to ensure that the device would be a safe place to charge and discharge a battery under varying thermal loads. That is, we always had the battery disconnected from any circuit when not in use, we implemented over-voltage and under-voltage protection as well as over-current protection, the positive terminal of the battery was connected directly to a fuse, and we intended for their to be no isolation faults. These were the considerations we made before much engagement in the design of this device. The final and perhaps most important decision we made was to not put a lithium-ion cell in the device for testing. We had detected what we thought to be an isolation fault the night before the demonstration and rather than force the issue, we decided that other engineers would greatly benefit if we didn't start a fire in the Senior Design Laboratory.

5.4 Future Work

Given that lithium-ion cells are still vastly misunderstood, this project warrants some form of continuation. We think that not only would the next iteration be more thermally efficient at cooling, but also perhaps even less power-hungry. For one, we know that with the right amount of testing we could optimize our Peltier devices for maximum output at a lower input power than even we have tested or replace them altogether with a compressor. A second version of this device would also include testing and implementing the P controller mentioned in section 2.1.7 as well as finishing off all the firmware in general. Other alternative future work might include design of our own battery charger, a more rigorous form of battery isolation as well as more fluid connect and disconnect from the charging and discharging circuit so that tests accuracy can improve.

References

[1] Jhaveri, Jeet. "[Battery Safety] Top 5 Reasons Why Lithium-Ion Batteries Catch Fire." ION Energy, 23 Apr. 2020, https://www.ionenergy.co/resources/blogs/battery-safety/.

[2] Ramanathan, Kumutha. “Firefighters Tackling a Tesla Blaze in Austin Said They Had to Use 40 Times More Water than for a Regular Vehicle Fire Because of the Car’s Lithium Battery Cells.” Business Insider, Business Insider, 17 Aug. 2021, https://www.businessinsider.com/tesla-crash-fire-lithium-battery-austin-more-water-than-regular-2021-8. https://www.bloomberg.com/news/articles/2021-09-24/battery-fires-haunt-the-electric-car-and-clean-power-revolution.

[3] Hull, Dana, and Naureen S Malik. "Battery Fires Haunt the Electric Car and Clean Power Revolution." Bloomberg.com, Bloomberg, 24 Sept. 2021,

[4] M. Integrated, “MAX77975/MAX77976,” Mouser.com, 2021. [Online]. Available: https://datasheets.maximintegrated.com/en/ds/MAX77975-MAX77976.pdf. [Accessed: 09-Dec-2021].

[5] Cui Devices, “Peltier Module,” CP36H-2 datasheet, Oct. 2019

[6] O. Semi, “MJ21193 - Silicon Power Transistors - mouser.com,” Mouser.com, 2000. [Online]. Available: https://www.mouser.com/datasheet/2/308/1/MJ21193_D-2315930.pdf. [Accessed: 09-Dec-2021].

[7] Wakefield-Vette, 641A Wakefield datasheet.

[8] CIS "Series APF Low-Profile Forged Heat Sinks," APF30-30-13CB datasheet, Oct. 2018.

[9] Micrel, "High-Current Low-Dropout Regulators," MIC29752WT datasheet, Jan. 2012.

[10] Diodes Incorporated, "1A Low Dropout Linear Regulator," AZ1117 datasheet, Jan. 2019.

[11] C. Staff, “Safe practice for lead acid and lithium batteries,” ECE 445 Course Webpage, 2016. [Online]. Available: https://courses.engr.illinois.edu/ece445/documents/GeneralBatterySafety.pdf. [Accessed: 09-Dec-2021].

[12] Unknown,. “Peltier Device Cooling Efficiency.” Peltier Effect Cooling Efficiency Test, 2013, https://rimstar.org/science_electronics_projects/peltier_effect_module_cooling_efficiency_test.htm.

6 Appendix A: Extra Figures and Tables

6.1 Extraneous Data

Table 2. Thermal unit testing results tabulated. Note the positive final derivative for 3A test implying that the temperature was starting to rise which indicates that 3A yielded sub-optimal performance.

Test Type	Starting Temperature(°C)	Ending Temperature(°C)	Ending derivative (°C per minute)
1 A Cold	28.06	22.06	-0.055
2 A Cold	26.67	15.83	-0.135
3 A Cold	28.61	24.17	0.139
2 A Hot	25.56	57.28	0.195

6.2 RV Tables

Table 3. Requirements and verification for the Discharging Unit.

Requirements	Verification
1. Isolate the battery and avoid any inrushes of current to the board or power supply. 2. Successfully control current out of the battery to within 1%.	1. Perform exhaustive startup conditions and measure current on oscilloscope over time. 2. Demand constant current at 0 A to 5 A in 1 A increments from the battery and use an oscilloscope to measure current at each step, compare results with expected value. 100%→0%).

Table 4. Requirements and verification for the Charging Unit.

Requirements	Verification
1. Isolate the battery and avoid any inrushes of current to the chip, board, or power supply. 2. Successfully control current flowing into the battery to within 1%.	1. Perform exhaustive startup conditions and measure current on oscilloscope over time. 2. Supply current to the battery at 0 A to 5 A in 1 A increments and use an oscilloscope to measure current at each step, compare results with expected value.

Table 5. Requirements and verification for the Sensor Suite.

Requirements	Verification
1. Isolate the battery and avoid any inrushes of current to the chip, board, or power supply. 2. Measure voltage and current on the battery to within 1 % of the true measurement.	1. Perform exhaustive startup conditions and measure current on oscilloscope over time. 2. Confirm that measurements read out from the main controller align to within 1 % with those of a voltmeter/ampmeter etc. over the course of an entire capacity test (e.g. SoC goes 100 % → 0 %).

Table 6. Requirements and verification for the computer interface.

Requirements	Verification
1. Successfully relay information about the given test being performed or the start/end of a test to the users computer over bluetooth. 2. Web application should store data from tests on the users computer ("save data to file"). 3. Web application should convey information about a cell's history, as well as its current behavior. 4. Web application will have an option for viewing the current state of the cell and information on the test provided that it is currently performing one.	1. Perform at least one of each type of test while having Bluetooth keep alive message time signatures be < 8 ms apart. 2.A Save test results to computer. 2.B Read test results back to ensure they were saved in the proper format. 3. The user-end graphs and data are aesthetic, easy to understand, and encapsulate the data well. 4. The testing page of the web application updates at least every 500 ms with all the data dumped to it since the last update, including current cell information.

Table 7. Requirements and verification for the Peltier Thermal Control.

Requirements	Verification
1. Set the internal temperature of the thermal chamber by causing a change in the polarity or power consumed by the Peltier devices. 2. Maintain a given temperature over long stretches of time (e.g. $\sim$ 10 hours) by controlling the output of fans and Peltier devices to ensure optimal efficiency and minimize needless power consumption. 3. Identify a valid range of testing temperatures in given environment (e.g. outdoors vs. in a refrigerator).	1.A Generate voltage difference to Peltier devices. 1.B Use a thermal camera for ensuring there is a difference. 2.A Using some form of thermistor, ensure that the temperature of the two sides of the Peltier device do not tend towards the average temperature. 2.B Ensure that fans and heatsinks perform the necessary functions well enough by examining Peltier device behavior through a thermal camera over time. 3. Use a thermometer to analyze the exact temperature ranges available to the system.

Table 8. Requirements and verification for the Test Cell.

Requirements	Verification
1. Measure the current temperature of the cell to within 5 °C	1. Use a thermometer to verify the accuracy of the temperature readings over the range of the thermal chamber's operating temperature.

Table 9. Requirements and verification for the low current power supply.

Requirements	Verification
1. The power supply must be completely stable for the correct operation of the microcontoller and ICs.	1 Tax the power supply by drawing its full load (1 A). Record and note any voltage differences in our ground compared to the lab's common ground.

Table 10. Requirements and verification for the high current power supply.

Requirements	Verification
1. The power supply must be completely stable as it is our reference and if something were wrong with the supply, we would likely not be able to detect an issue with the battery. 2. The supply will likely need to supply the necessary current to charge the battery so it should be rated for probably about 100 W to accommodate both the charging and the power consumption of the Peltier devices.	1.A Tax the power supply by drawing as much a load as we can (e.g. 5 A discharge and 0 °C). Record and note any voltage differences in our ground compared to the lab's common ground. 1.B Demand a 5 A switching load from the battery and supply 1 A switching current to the Peltier devices. Record and note any voltage differences in our ground compared to the lab's common ground. 2. Tax the power supply by drawing as much of a load as we can. Ensure there are no overcurrents in the ratings of the power supply.

6.3 Figures

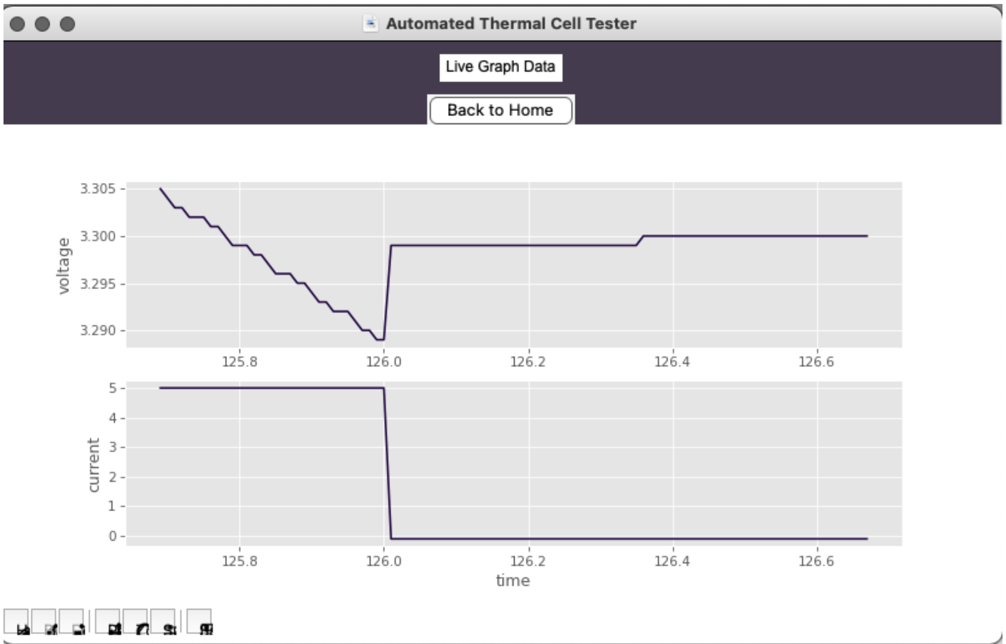

Figure 12: Live data feed of the python GUI.

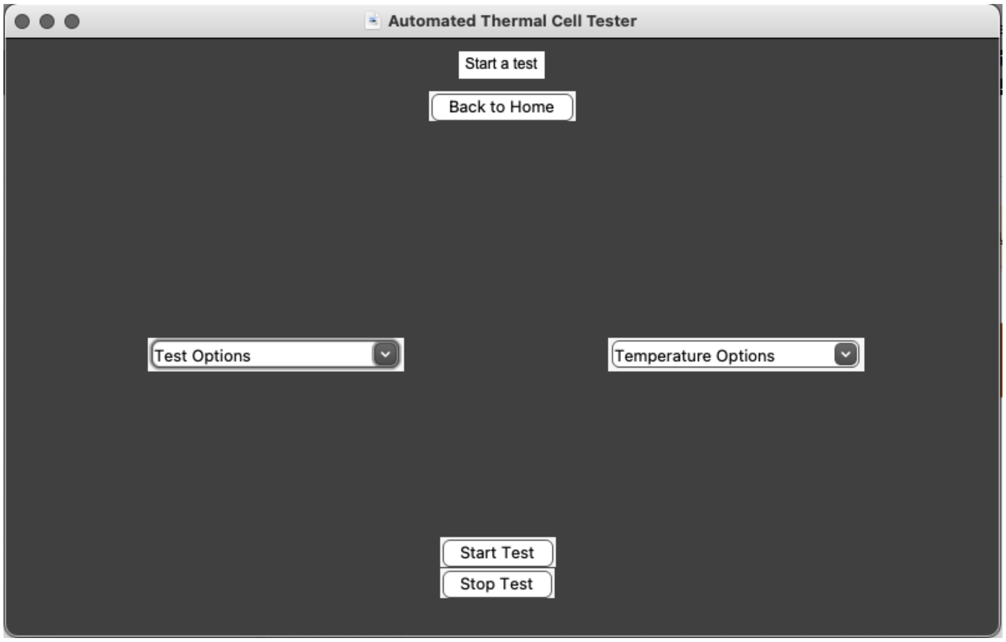

Figure 13: Test start page with options for testing temperature and test type as well as buttons to send messages to the controller.

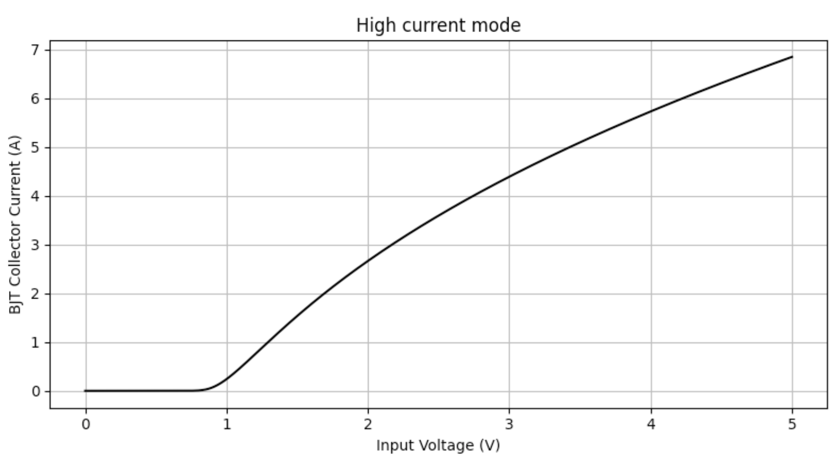

Figure 14: Simulation of discharge circuit in high current mode.

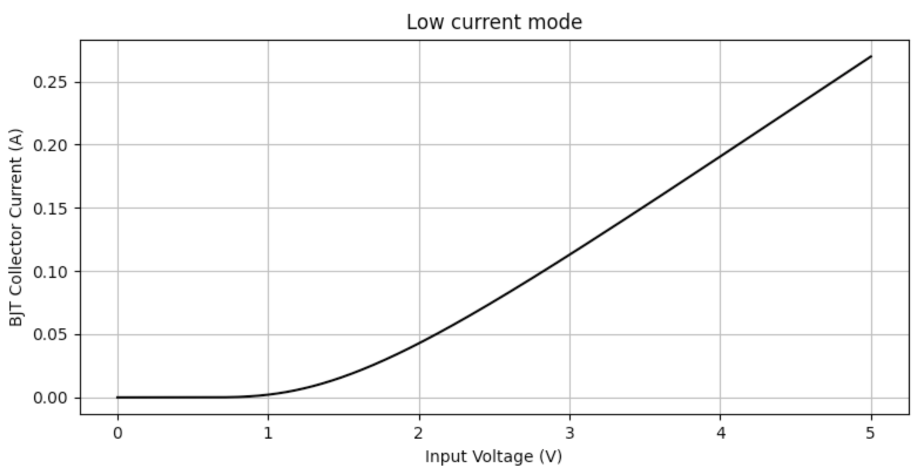

Figure 15: Simulation of discharge circuit in low current mode.