ECE 492 Team 5
Topic 1: Lab Power Supply
Faculty Supervisor: Dr. Jens-Peter Kaps, jkaps@gmu.edu

Carlos Martinez: cmarti62@gmu.edu
Juan Alejandro Gaspar-Zapata: jgasparz@gmu.edu
Tarun Singh: tsingh8@gmu.edu
Dean Hosek: dhosek@gmu.edu

Lab Power Supply
Final Report

1. Executive Summary

The COVID-19 pandemic in 2020 resulted in a large push to work from home [1]. Unfortunately, for anyone with engineering tasks at-hand they would now need to search for affordable testbench equipment to aid in them. For example, testbench equipment such as power supplies were now necessary for students in powering their projects, something which was already attractive for hobbyists. Examining the current market, we compared a small sample of multiple channel low-cost power supplies such as the Korad KA3305P ($235), siglent SPD3303C ($299), GW Instek GPE-2323 ($306) [2, 3, 4], In the end we arrived at the assumption of a tendency for prices to be over $100, especially for added functionalities such as programmability or multiple output channels. Our design seeks to fill that void by providing an inexpensive lab power supply that would not only supply DC voltage but provide high-end functionalities at a low $106 cost.

2. Table of Contents

Contents

1. Executive Summary 2
2. Table of Contents 3
3. Requirements 5
    1. Mission Requirements 5
    2. Operational Requirements 5
4. Approach 6
    1. Motivation 6
    2. Identification of Need 6
    3. Market/Application Review 6
    4. Approach based on Conceptual Design 8
    5. Alternative Approaches 9
    6. Individual Contributions 10
5. System Design 11
6. Technical Section 15
7. Experimentation 22
8. Experiment Validation Using Evaluation Criteria 31
9. Additional Issues 42
10. Administrative Part 45
11. Lessons Learned 47
12. References 48
13. Appendix A: Proposal 49
    Executive Summary 49
    Problem Statement 50
    Approach 52
    System Design 56
    Preliminary Experimentation Plan 62
    Preliminary Project Plan 63
    Potential Problems 67

References 68
14. Appendix B: Design Document 69
  Problem Statement 69
  Market/Application Review 71
  Requirements Specifications 72
  Current Part List 73
  Background Knowledge and Phenomenology 75
  System Design 80
  Detailed Design 86
  Prototyping Progress Report 91
  Testing Plan 108
  Tasks for ECE-493 109
  Schedule and Milestones 113
  References 114

3. Requirements

1. ***Mission Requirements***

- The device will operate two independent controllable voltage channels

2. ***Operational Requirements***

Input/Output Requirements

- The output voltage and current limit per channel shall be adjustable using a keypad and a knob (rotary encoder)
- Voltage, current, and current limit input/output per channel shall be displayed on a simple LCD
- Output voltage shall be adjustable between 0-10V
- Power supply will have the ability to set current limits per channel.
- Power supply will be controllable through USB using SCPI
- Power supply will offer two channels

External Interface Requirements

- Power supply shall receive power from Mains power

Functional Requirements

- Maximum Current per Channel shall be 500mA
- If the current limit is reached, the output will switch off
- Digital readouts for voltage, current, and current limit per channel shall be shown if the current limit was reached

Technology and System-wide Requirements

- Custom enclosure will be designed to hold all circuitry
- All components will be set in custom PCB
- Materials cost should be in the range of $50-$100
- A 0.5A fast act fuse shall be used for user and circuit protection

4. Approach

1. *Motivation*

The COVID-19 pandemic in 2020 significantly increased the number of online classes, including those in the electrical and computer engineering field. As many of these classes contained a lab component, students were required to purchase lab power supplies that could be powered and used at home to run experiments for these labs. These lab power supplies would be used in both AC and DC situations, although most labs required only DC voltages/currents to be used. Most power supplies that offer high outputs with high functionality are very expensive, often starting at $300. Cheaper supplies often do not have high end functionality and are quite noisy. Our design seeks to fill this void and provide an inexpensive lab power supply that would offer high end functionality, letting students pay less for the same functionality (for the purposes of their lab).

2. *Identification of Need*

To design a programmable two-channel power supply and meet our project goals and requirements, we must achieve the following:

- The device must be computer controlled
- Output voltage at each channel to be adjustable between 0-10V
- Developed solution must be low-cost ($50-$100)
- A custom front panel to display voltage and current information to the user
- Device must be controllable from a computer (via USB) and using a keypad and a knob
- Measurements and outputs displayed with high accuracy

3. *Market/Application Review*

We took a small sample from the current market as given by Tables 1 and 2 and compared price with specifications and functionalities. We found there is currently an increasing gap in the market for an inexpensive two channel DC power supply, so we aimed to fill that gap. Our design would need to be inexpensive to create but also competitive by adding high-end functions, e.g., programmability, multiple output channels, and intuitive external peripherals. With these goals in mind, we assumed the product would be more favorable especially for students and hobbyists, who may not have the necessary resources for the average power supply. Our system would allow the user to input a voltage between 0 and 10V and a current limit between 0 and 500mA for each of the two channels independently. The power supply would then output said voltage at a current below the limit and display both the user input voltage/current values as well as the actual output voltage/current values for each channel. The power supply would also be connected to Mains voltage as a power source.

Table 1: Dual Channel Programmable DC Power Supplies Comparison

Dual Channel DC Power Supplies			
Model, Programmability	Cost (USD)	Vmax	Vmin
E36233A, programmable [5]	3059	30	0
Series 2220-30-1. Programmable [6]	1320	30	0
SPS5044X, programmable [3]	2269	40	0
DP821A, programmable [7]	731	60	0
GPD-2303S, programmable [4]	545	30	0
GPE-2323, non-programmable [4]	306	32	0

Table 2: Triple Channel Programmable DC Power Supplies Comparison

Triple Channel DC Power Supplies			
Model, Programmability	Cost (USD)	Vmax	Vmin
EDU36311A, programmable [5]	898	30	0
Series 2230-30-1, programmable [6]	1480	30	0
SPD3303X, programmable [3]	549	32	0
SPD3303X-E, programmable [3]	399	32	0
SPD3303C, programmable [3]	299	32	0
DP831, programmable [7]	473	30	0
SPD-3606, non-programmable [4]	759	30	0
KD3305P, programmable [2]	250	30	0
KA3305P, programmable [2]	235	30	0

4. *Approach based on Conceptual Design*

The conceptual design for this project began with the requirement specifications of a 2-channel lab power supply that could be controlled externally via a knob and/or keypad in addition to USB. With the requirements of a keypad and USB access, we immediately knew that the system would need to be controlled by a microcontroller. Another requirement was that the two channels would need to be electrically independent, so controlling both channels with only a single microcontroller would not be possible. With this requirement, we knew early on that we would need at least two microcontrollers to control each channel separately and that they would need to be able to speak to each other in an electrically isolated way, such as with an optocoupler. Since everything would be controlled with microcontrollers, every input needs to be digital, so the knobs will be rotary encoders rather than analog potentiometers, which would also give us more accurate measurements. Furthermore, having to use Mains as the power source would require more safety precautions for the user, so components would have to be chosen meticulously. The rest of the design was based on other lab power supply front panels as well as a sample front panel design given to us by Dr. Kaps.

5. Alternative Approaches

One alternative design was to use a single MCU as a centralized controller, allowing us to use analog sensors along with a corresponding ADC on each channel for communication. The ADC and MCU would communicate through a full-duplex communication protocol such as SPI, but the two would be on separate circuits. The central MCU would be on its own isolated and regulated voltage rail designed to meet the specifications of the MCU for optimal safety and reliability while the sensors and their ADCs would be placed on each voltage channel to measure the voltage and current for that channel. To control the output of the channel, a digital potentiometer, PWM signal, or a DAC would be used to interface with the voltage regulation circuit of that channel, depending on the design of the circuit. These components would similarly communicate through the same full-duplex protocol as the other sensors on its channel. However, since the channel and MCU are electronically separated, an alternative means of communication is needed. This would be accomplished through the optocouplers, as it is in the selected design. This design was not chosen as getting the individual sensors, ADCs, and DACs would end up costing more than a single MCU that has all these features built into it. Furthermore, the central MCU would need higher capabilities and performance since it would have to control all channels simultaneously. Using a single MCU per channel instead would reduce the load on the central MCU, thus allowing for a broader distribution of resources.

An alternative design choice for specifically the voltage regulation circuit in each channel was to use a power electronics microchip instead of assembling individual components to build a regulating circuit. A suitable microchip that can both control and drive the outputs for the voltage channels could not be found.

An alternative design choice for input power was to use a power supply that was designed for use in a desktop personal computer. This choice would eliminate any risk from dealing with Mains voltage as we would not need to open the supply to access the 3.3V, 5V, 12V, and ground pins and it could be treated as a black box. This choice would simplify the circuit slightly by removing the need to rectify AC power, but it comes at an increased cost, as a PC power supply costing around $50 or more while a suitable transformer can be found for under $20.

Several approaches to regulating the supply voltage were considered. The first approach considered was to use a buck-boost converter to achieve a user desired output voltage and the current limit would have to be dealt with by external circuitry. A buck-boost converter is a switching regulator, so the output has a lot of ripples, which is undesirable in the context of lab experiments. To achieve a stable output voltage, we decided that the regulator circuit should at least include a linear regulator at the end. The first approach was further modified as we did not see a need for a boost converter if the transformer can supply the necessary DC voltage after rectification. For this reason, we specifically look for parts with low voltage drops like Schottky diodes and LDO regulators. If the transformer is unable to supply an adequate voltage after rectification, it may be necessary to use a buck-boost or simply a boost converter, but for now, a more specialized buck converter switching regulator with an adjustable current limit seems to be the better choice.

6. *Individual Contributions*

Most of the project was divided as planned in ECE-492, but losing a member caused a few alterations. During prototyping, Tarun assisted everyone, Carlos and Dean handled the linear regulator circuit, and Juan and Carlos undertook the switching regulator circuit. The front panel PCB was mostly handled by Juan and Carlos, while the schematic was done by Tarun and Carlos. The front panel keypad and spacer was designed and printed by Tarun. A front panel cutout design was designed by Dean. The main power supply schematic was handled by Tarun and Carlos, with Carlos designing the PCB. Soldering the final PCB components was primarily handled by Carlos but all members contributed to quick assembly. The front panel assembly was also primarily done by Carlos, but all members contributed to its assembly. Programming the USB was handled by Carlos, while LCD and rotary encoder were done by Juan and Tarun respectively. With those fundamentals in place, user interactions and their effects on the system (PID control and state machines) were handled by Carlos and Tarun, while Dean handled SCPI programmability as well as the design of the front panel laminate in GIMP. All members contributed to presentations and document submissions.

5. System Design

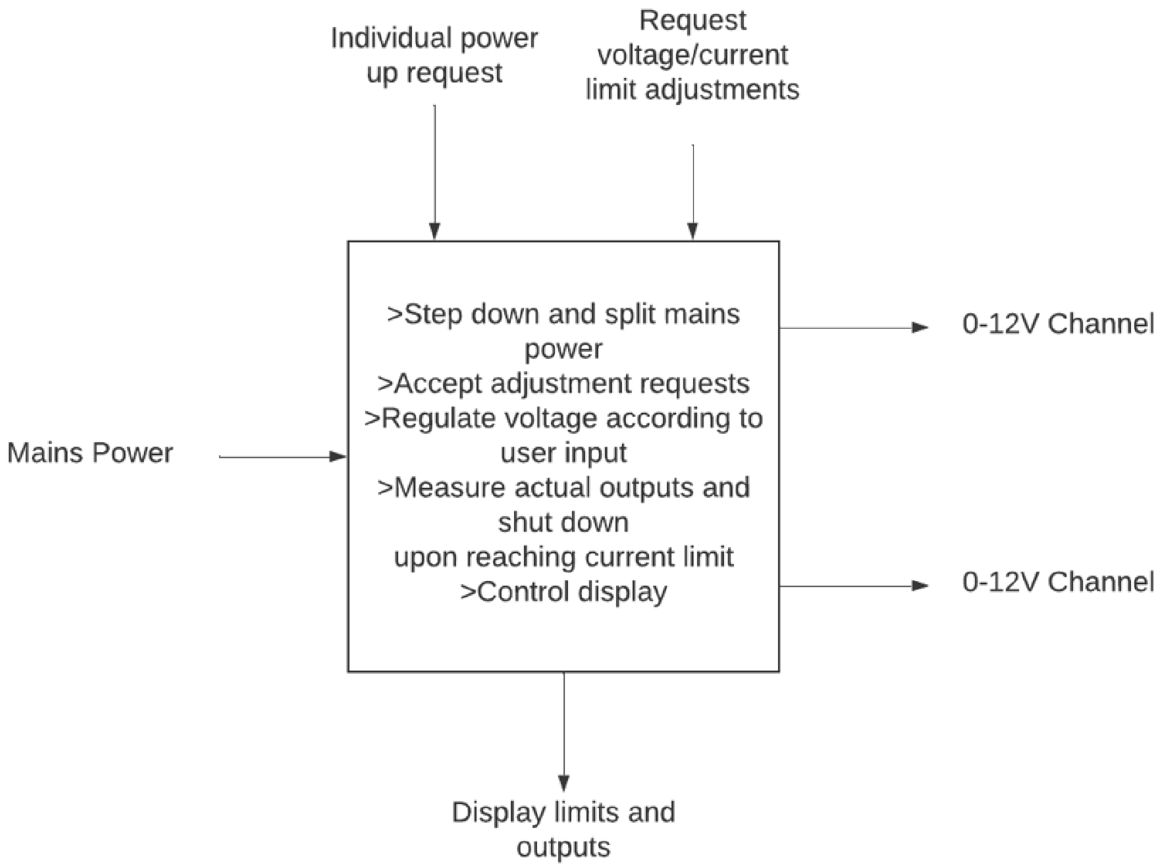


Figure 1: Level 0 Functional Decomposition

Figure 1 shows the inputs and outputs as well as the top-level functions that comprise our system. The inputs and outputs are simple, the system takes mains power and outputs two independent 0-12V channels controllable by the user and display the user set limits and actual outputs.

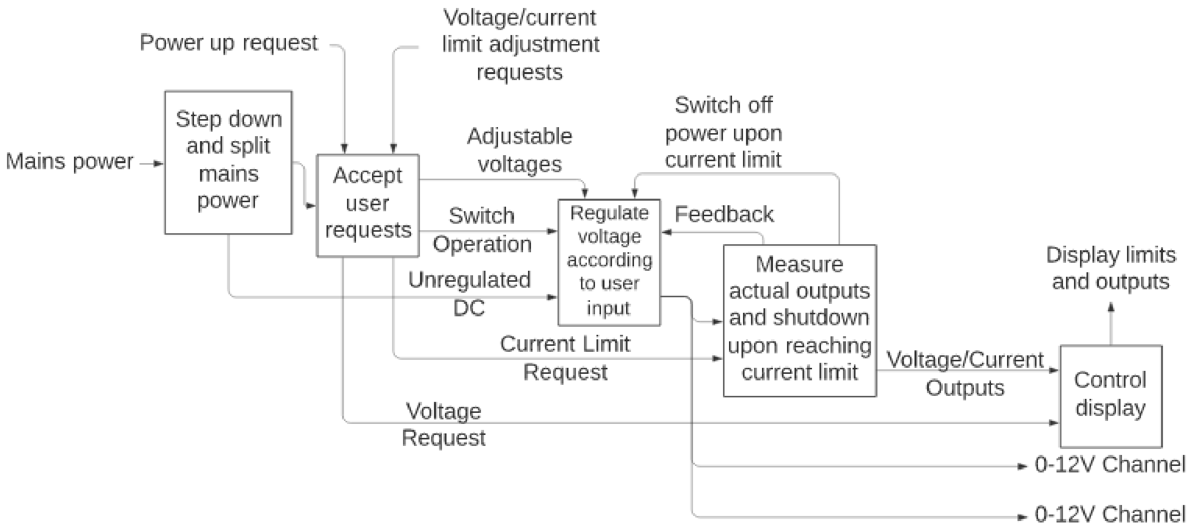


Figure 2: Level 1 Functional Decomposition

Figure 2 shows a more detailed functional decomposition, each function is listed along with how each function interacts with other functions. The first function is responsible for taking mains power and converting it into something manageable by stepping it down with a transformer, splitting the single lane input to two independent outputs, and rectifying it. The second function is responsible for taking input from the user and interacting with the rest of the system to output a user desired voltage on each channel and display said user input. The third function is responsible for taking the unregulated but rectified AC voltage from the transformer and regulate it down to a user specified voltage. The fourth function is responsible for measuring the actual voltage and current outputs of each channel to display to the user and to shut off the switch if the output current is past the user defined limit. The fifth function is responsible for taking the user inputs and the measured outputs and displaying them on a simple LCD.

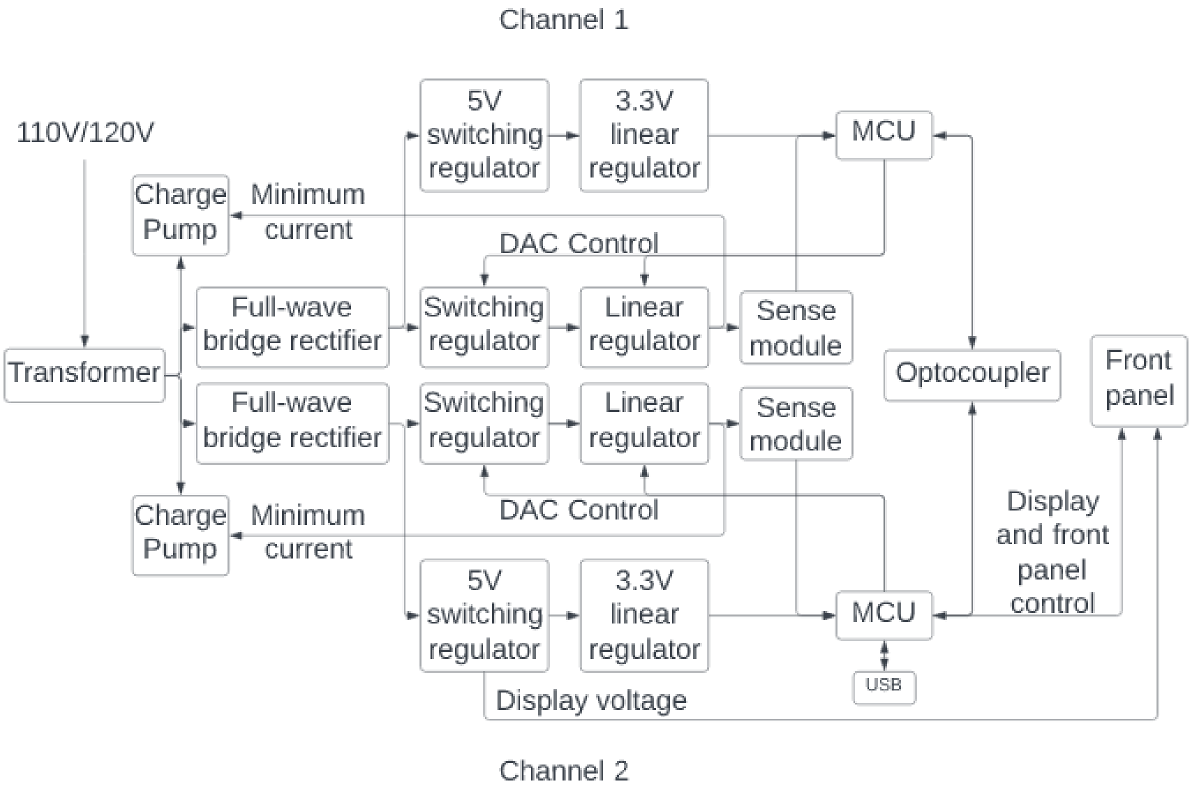


Figure 3: Level 2 System Architecture

Figure 3 shows a detailed block diagram of the individual modules in our system and how they interact with each other. We take mains power and step it down with a transformer that has two secondaries, one for each channel. The secondaries go into a full wave bridge rectifier to supply our system with DC voltage as well as a charge pump to supply us with a negative voltage that we can use to bias our OpAmps and sink the minimum current required by our linear regulators. From the bridge rectifier we preregulate our voltage using a switching regulator for efficiency. From the switching regulator we further drop the voltage to a user specified voltage using a linear regulator for a stable output. Our MCU is powered by its own 3.3V static linear regulator which is preregulated with a 5V switching regulator that is also used to power our display. The MCU reads the output voltage and current using ADC pins and controls the regulators with DAC pins. Each MCU communicates with the other through a pair of optocouplers so that the channels remain isolated. Our main MCU on channel 2 is also responsible for reading inputs from the front panel and USB as well as controlling the front panel display.

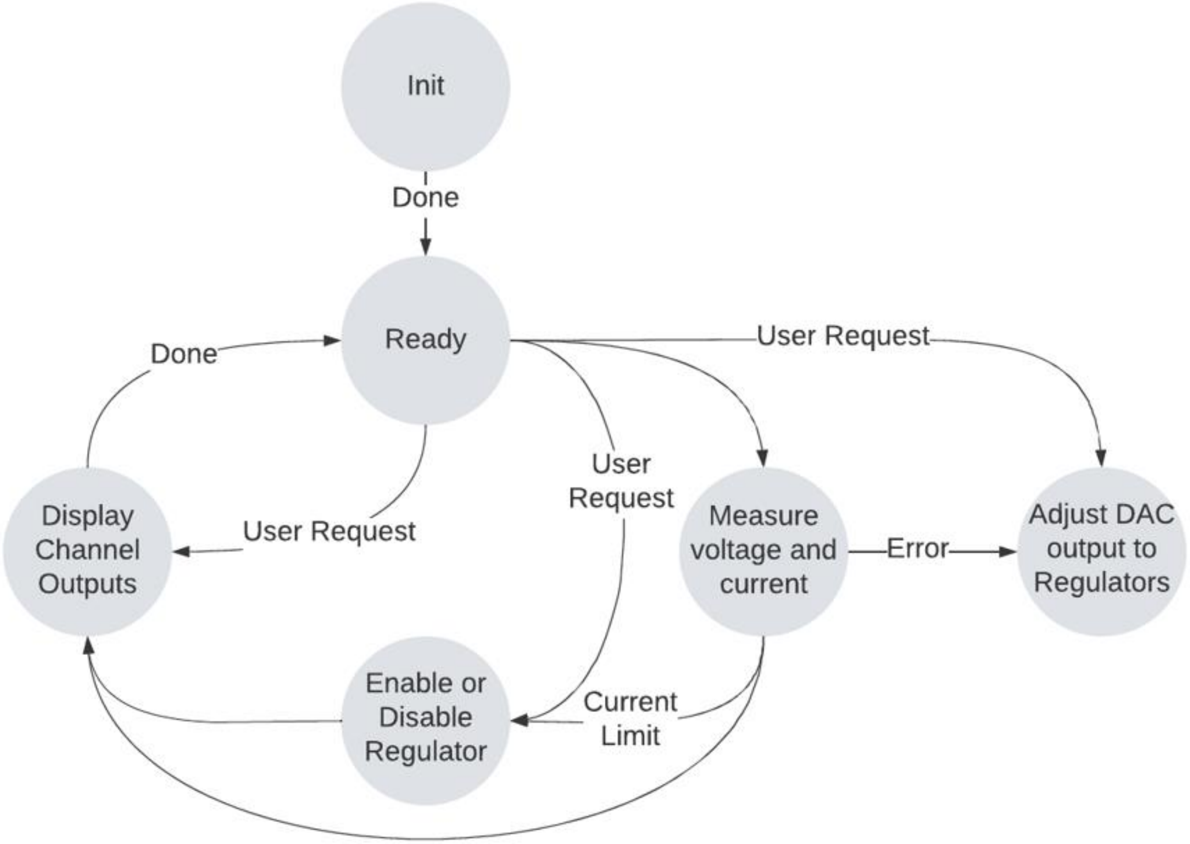


Figure 4: Overall System State Machine

After our system initializes itself, it will automatically begin measuring the output voltage and current and display these results on the LCD. By default, the system will be in shutdown and no current limit will be set so the device should display everything as 0. Once a user requests a voltage or current limit the display will be updated with these settings and the MCU will begin to adjust the DAC to output the specified voltage. The system will continue to read the output voltage and will further adjust the DAC to maintain the user specified voltage. If the user enters a current limit and this current is reached the system will automatically enter shutdown.

6. Technical Section

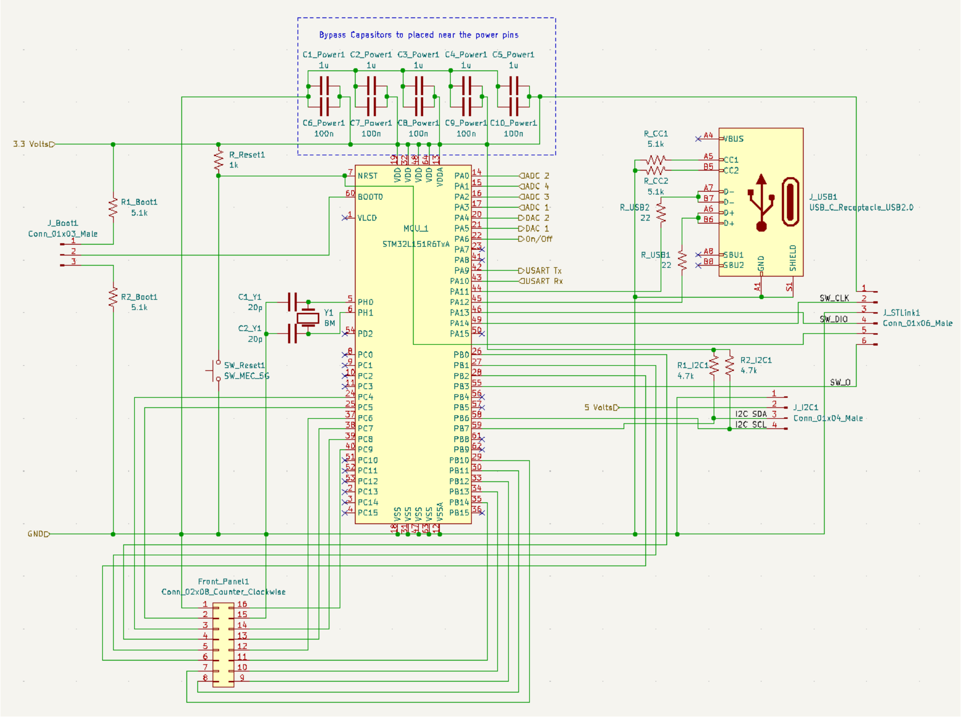


Figure 5: Final MCU Configuration

The brains of the power supply unit we developed are the microcontrollers, which are responsible for handling user inputs, processing them, and controlling the output voltage/currents based on said inputs. The secondary MCU is set up exactly the same except it does not need to be responsible for USB, I2C, or the front panel. The secondary MCU is also slightly smaller, being only 48 pins and having less flash and ram but it only needs to receive information from the primary MCU, send information back, and regulate its channel.

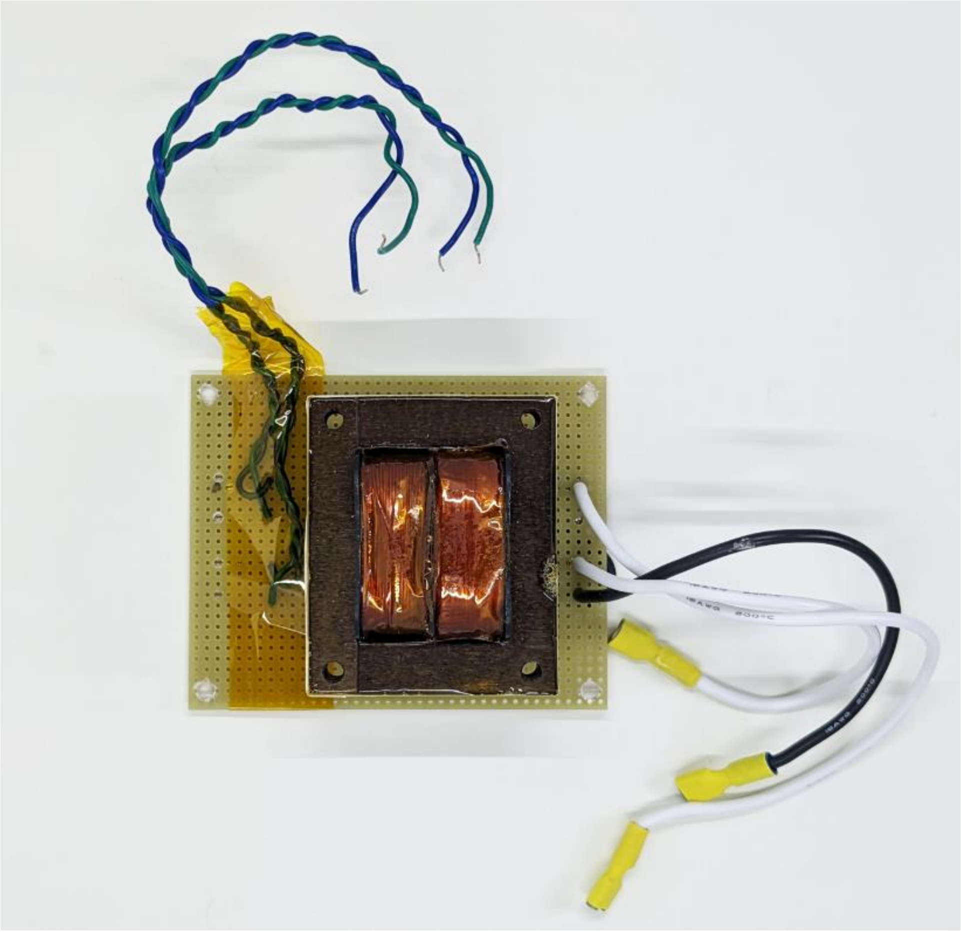


Figure 6: Dual Secondary Transformer

To transform Mains voltage into a voltage that can be safely used by our system, a step-down transformer was used. A step-down transformer is a component used to convert high voltage at a low current into low voltage at a high current, where the secondary voltage is directly proportional to the turns on the secondary coil and the primary voltage, all divided by the number of turns on the primary coil ($V_s = \frac{(V_p \cdot N_s)}{N_p}$).

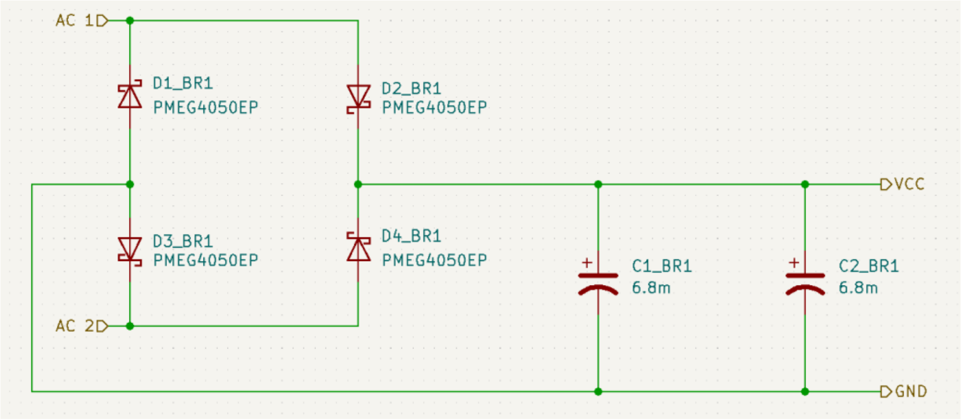


Figure 7: Full Wave Bridge Rectifier

A full-wave rectifier, which is a circuit that converts alternating current voltage into direct current voltage, was used to convert the stepped-down voltage into DC voltage. The full-wave rectifier has a voltage drop equivalent to $2V_D$, where V_D is the voltage drop across one of the diodes in the rectifier. There is an additional drop that we need to account for when flattening a sinusoidal wave, the average output of a bridge rectifier is $2Vp/\pi$ and we get a peak voltage of 12/0.7071=16.97072V with an RMS voltage of 12. Plugging this peak voltage back into the first equation we get an average voltage output of 10.80389V. This assumes a full load however and we did not plan to operate the transformer at max load; however, the bigger and larger display we chose drew 0.1A to 0.2A on its own and in addition to our 0.5A output current draw we saw a big drop on the 0.8A transformer we used in the beginning. This drop in voltage caused our opamp circuit to oscillate until we added a massive capacitance to the bridge rectifier to keep the voltage above 12V. We have since replaced the 0.8A transformer with a larger 1.5A transformer and consequently we have increased our power limits from the original 10V 0.5A to 12V 0.8A.

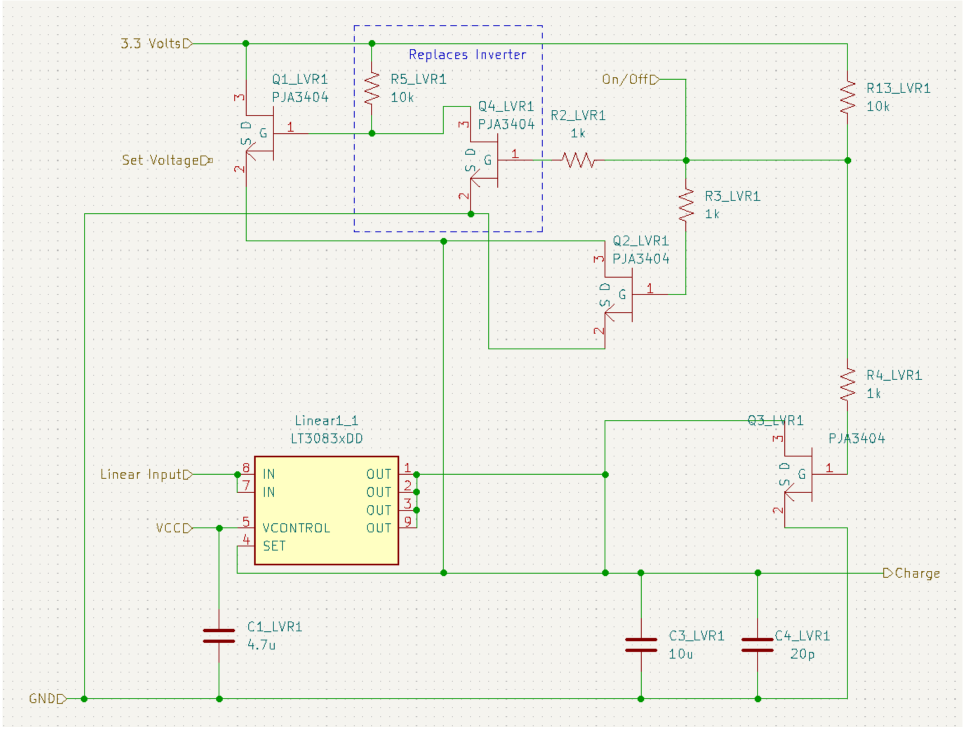


Figure 8: LT3083 Linear Regulator

The regulator circuit that outputs the voltage sourced by the rest of our system will be composed of three types of regulators: linear, switching, and static. Linear regulators are regulators that utilize a linear component, such as a potentiometer, to regulate input voltage. These types of regulators are relatively simple to design and do not output much noise, but they have a low power capacity and are not as efficient as other regulators. We had a lot of problems with the LT3080 latching up or down during operation and we attempted to fix this through advice we received from EEVBlog and while it did solve the latch up problems it caused further problems when trying to operate the device in shutdown [8]. Because of this, we needed to switch to using the more expensive LT3083.

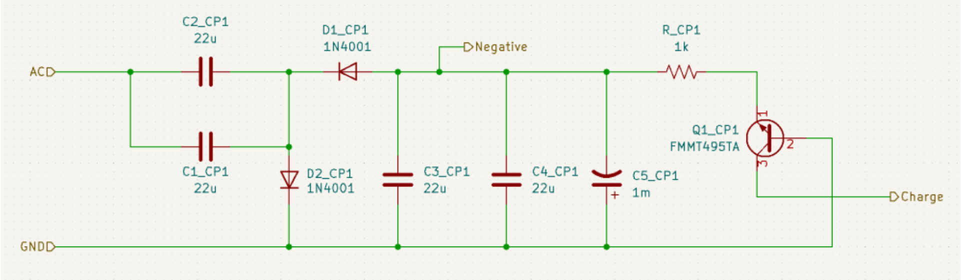


Figure 9: Charge Pump

The linear regulator we chose requires a minimum current draw of 1mA to regulate near 0V but this minimum current must be returned to a negative voltage source. A negative voltage source was also necessary to bias our OpAmp circuit to operate at or near 0V. A negative voltage source from a switching regulator was briefly considered before being dropped in favor of a much cheaper charge pump.

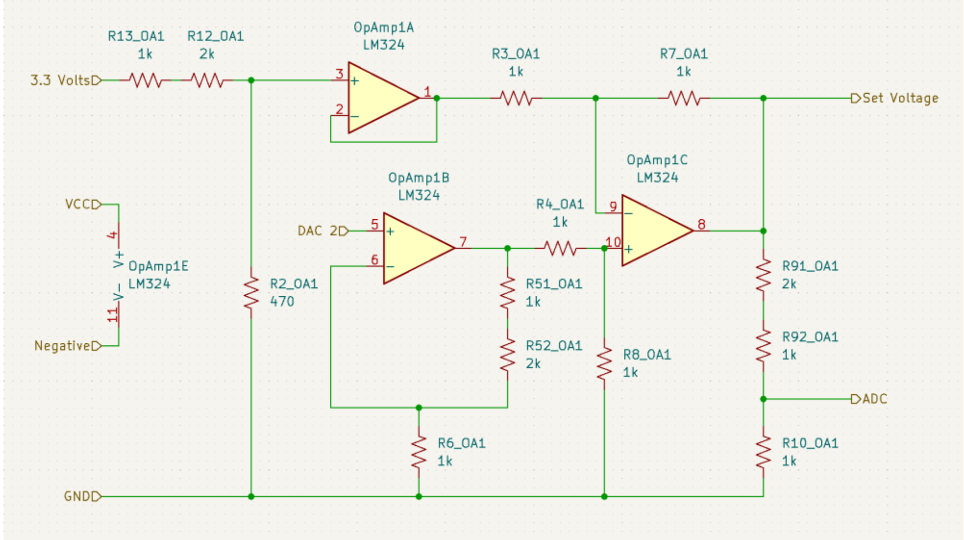


Figure 10: OpAmp Circuit

The linear regulator we chose outputs whatever voltage it sees on its set pin so to control it using a DAC requires us to multiply our DAC by a gain of 4 using a non-inverting Opamp. Both the Opamp and the DAC have a minimum output voltage which would have caused us to have a minimum output voltage of nearly half a volt. In order to remove this offset, we divide a 3.3V voltage reference and buffer it with a voltage follower which is finally subtracted from our non-inverting amplifier using a differential amplifier with a gain of 1. This does mean that we can now output a slightly negative voltage, but we can carefully control the shutdown to prevent this from ever occurring.

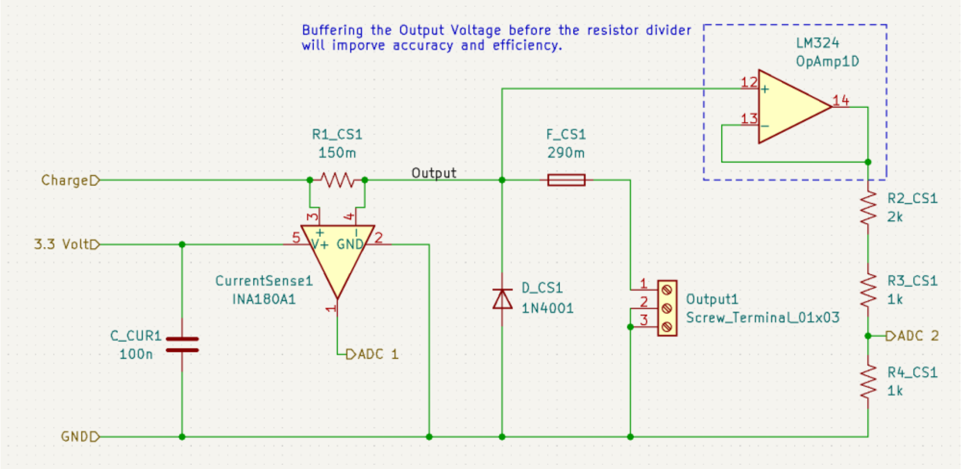


Figure 11: Output Sense Circuit

We measure the output current using a current sense resistor and a current sense amplifier with an ADC probe on it. We measure the output voltage by resistively dividing it with an ADC probe on it. The placement of our resistive divider was not reconsidered after adding a flyback diode and a fuse to prevent reverse polarity and it should have been placed before the current sense or at least buffered with an OpAmp. We chose a flyback diode and fuse instead of just a diode for reverse polarity because we previously could not afford a large voltage drop on the 0.8A transformer but our new 1.5A transformer could easily support a diode in series as well as a MOSFET for reverse current protection.

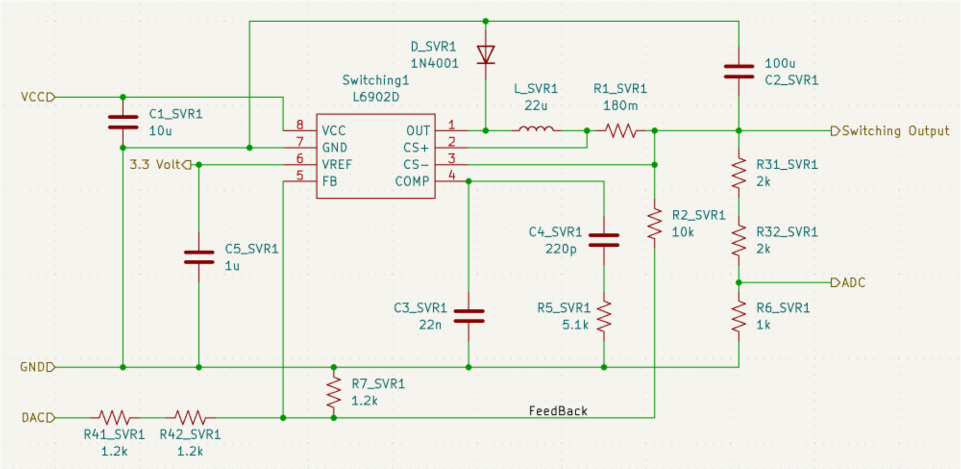


Figure 12: L6902 Switching Regulator

Switching regulators use switching components, such as MOSFETs, and work by taking small chunks of energy from the input voltage source and transferring them to the output. These kinds of regulators are more efficient and can handle excessive amounts of power but are also more complicated to design and have noisy output signals. To control the output of this switching regulator we introduced a third resistor R3 to the node between the first two connected to the feedback pin and drove this new resistor with a voltage from a DAC. Applying KVL at the feedback pin of a switching regulator gives us the following equation:

$$Vfb = \frac{R3R2Vout \ + \ R1R2Vdac}{R3R2 \ + \ R1R3 \ + \ R1R2}$$

This equation can then be used to make a system of equations based on the minimum and maximum output of the regulator as well as the DAC to find the required resistance of any two resistors given any choice of a third resistor. This regulator worked well during prototyping and under a passive load using some 5W resistors, but we later determined that the inductor we chose was the wrong frequency and this caused major problems when regulating down to low voltages while operating at a high current draw. Due to this, we disabled control of this regulator, and we must drop voltage using the linear regulator.

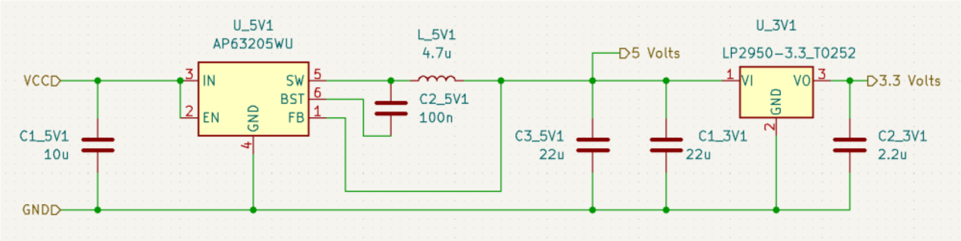


Figure 13: AP63205WU and LP2950 Static Regulators

Static regulators are normal switching or linear regulators that output a constant voltage value that can't be changed, which makes them ideal for powering a device that needs a specific constant voltage, such as a microcontroller, but not for powering something that requires a variety of voltage values, such as the output channels for our PSU. We get a static 5V switching regulated voltage out of the AP63205WU to power the LCD and pre-regulate the static 3.3V linear regulated voltage out of the LP2950 which powers the MCU.

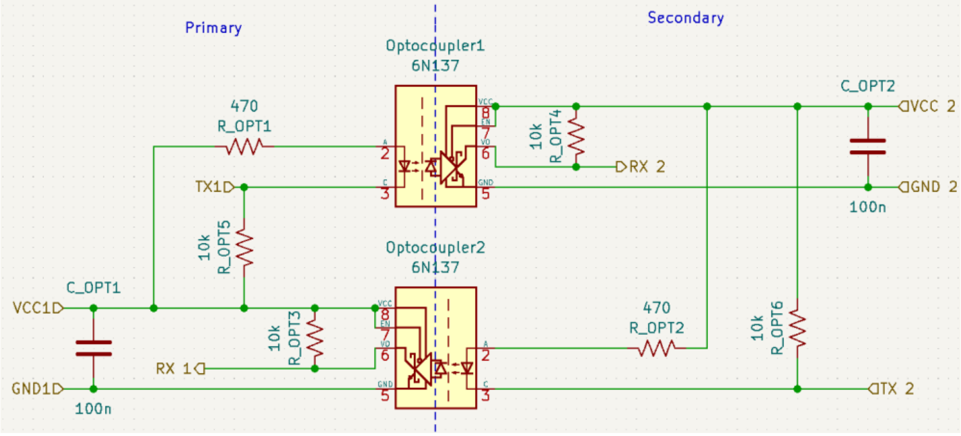


Figure 14: Optocoupler Circuit

For our MCUs to talk to each other without sharing ground we needed to use an isolated medium such as optocouplers. We have two optocouplers for our board so that the MCUs can speak UART with each other and communicate relevant information.

7. Experimentation

The first experiment we ran was to step our power supply from 0-12V as well as 0-0.8A and compare the voltage and current we read out with the voltage and current from a known accurate source. We used the Rigol 3021 load tester that gave us the current load we wanted to get the accurate voltage and current readings. Since both our supply and the Rigol load tester both accept SCPI commands this process was automated through PyVISA so we were able to collect a lot of data. We collected voltage data in 48 steps and current data in 20 steps for channel 2 of our supply which is the main microcontroller because it would be the busiest and offer the worst results. We took data from channel 1 with lower resolution with voltage being in 12 steps and current in 10 steps.

Table 3: Raw data from channel 2 at 0.76A incrementing load

Voltage Requested	Voltage Displayed (Our supply)	Voltage Displayed (Rigol load)	Current Requested	Current Displayed (Our supply)	Current Displayed (Rigol load)
0.25	0.24	0.23528	0.76	0.768	0.758612
0.5	0.5	0.487542	0.76	0.789	0.7587
0.75	0.7	0.733813	0.76	0.761	0.758684
1	0.95	0.984754	0.76	0.761	0.758684
1.25	1.25	1.234762	0.76	0.767	0.758684
1.5	1.49	1.483925	0.76	0.761	0.758708
1.75	1.75	1.738351	0.76	0.764	0.758732
2	2	1.984414	0.76	0.768	0.7587
2.25	2.25	2.232465	0.76	0.765	0.758764
2.5	2.49	2.483495	0.76	0.754	0.758748
2.75	2.75	2.73248	0.76	0.758	0.758716
3	3	2.985557	0.76	0.769	0.758644
3.25	3.24	3.235743	0.76	0.768	0.758572
3.5	3.5	3.482814	0.76	0.767	0.7587
3.75	3.75	3.734037	0.76	0.764	0.75878
4	4	3.982681	0.76	0.768	0.758724
4.25	4.25	4.234275	0.76	0.769	0.758612
4.5	4.5	4.483127	0.76	0.759	0.7587
4.75	4.75	4.733475	0.76	0.755	0.758652
5	5.01	4.98369	0.76	0.755	0.75866
5.25	5.24	5.229931	0.76	0.758	0.7587
5.5	5.5	5.482475	0.76	0.768	0.758652
5.75	5.75	5.73091	0.76	0.77	0.75862

Table 3 continued: Raw data from channel 2 at 0.76A incrementing load

Voltage Requested	Voltage Displayed (Our supply)	Voltage Displayed (Rigol load)	Current Requested	Current Displayed (Our supply)	Current Displayed (Rigol load)
6	6	5.981245	0.76	0.765	0.7587
6.25	6.25	6.233654	0.76	0.762	0.758676
6.5	6.51	6.478887	0.76	0.766	0.758644
6.75	6.76	6.729903	0.76	0.763	0.758628
7	7.01	6.980638	0.76	0.751	0.758628
7.25	7.26	7.228362	0.76	0.76	0.758652
7.5	7.5	7.480949	0.76	0.757	0.758684
7.75	7.73	7.728881	0.76	0.762	0.758684
8	8	7.980415	0.76	0.764	0.758716
8.25	8.25	8.229934	0.76	0.754	0.758732
8.5	8.5	8.47708	0.76	0.768	0.758708
8.75	8.75	8.726731	0.76	0.765	0.758764
9	9.02	8.977258	0.76	0.762	0.75874
9.25	9.24	9.227296	0.76	0.764	0.758724
9.5	9.51	9.477852	0.76	0.762	0.758612
9.75	9.77	9.727341	0.76	0.764	0.758732
10	9.98	9.977096	0.76	0.766	0.758644
10.25	10.23	10.224657	0.76	0.763	0.758652
10.5	10.49	10.477007	0.76	0.759	0.75862
10.75	10.72	10.726792	0.76	0.76	0.758692
11	11	10.974887	0.76	0.759	0.758748
11.25	11.24	11.225384	0.76	0.768	0.758668
11.5	11.51	11.474784	0.76	0.756	0.7587
11.75	11.75	11.723442	0.76	0.768	0.758772
12	12.01	11.975363	0.76	0.756	0.758748

We initially took our data going from 0-12V and 0-0.8A in increasing order, this resulted in our system building up a lot of heat by the time it came for the larger loads due to the failure of our switching pre regulator. This heat then affected our MCU either directly or through the jumper wire we used to move the voltage divider on the output behind our current sense resistor and we saw about 30mV drift at the high end. At 0.8A load our switching regulator failed even harder dropping to around 9V without us attempting to regulate it due to the improper inductor but we include this data in processing because it does not significantly impact our average error.

Table 4: Raw data from channel 1 at 0.72A incrementing load

Voltage Requested	Voltage Displayed (Our supply)	Voltage Displayed (Rigol load)	Current Requested	Current Displayed (Our supply)	Current Displayed (Rigol load)
1	1.01	0.988699	0.72	0.729	0.719231
2	2	1.988255	0.72	0.727	0.719207
3	3	2.992229	0.72	0.725	0.719279
4	4	3.991089	0.72	0.722	0.719319
5	5	4.994826	0.72	0.73	0.719319
6	5.99	5.994264	0.72	0.732	0.719263
7	7	6.995274	0.72	0.724	0.719247
8	8	7.997053	0.72	0.722	0.719319
9	9	8.999694	0.72	0.723	0.719279
10	10	10.000332	0.72	0.717	0.719167
11	11.01	11.001386	0.72	0.73	0.719271
12	12	12.005613	0.72	0.721	0.719175

Since our testing on channel 1 was done in much fewer steps it went by much quicker and the system did not have time to build up a significant amount of heat to disturb our results. We still encountered the same issue as we did with channel 2 where the switching regulator failed at 0.8A but the voltage drop on channel 1 was significantly higher. We still chose to include this data in processing.

Table 5: Raw data from channel 2 at 0.76A decrementing load

Voltage Requested	Voltage Displayed (Our supply)	Voltage Displayed (Rigol load)	Current Requested	Current Displayed (Our supply)	Current Displayed (Rigol load)
0.25	0.25	0.246876	0.76	0.761	0.758732
0.5	0.5	0.497818	0.76	0.76	0.758772
0.75	0.75	0.746906	0.76	0.759	0.758748
1	1	0.996662	0.76	0.762	0.758716
1.25	1.25	1.247559	0.76	0.766	0.75878
1.5	1.5	1.497685	0.76	0.753	0.758668
1.75	1.76	1.748523	0.76	0.761	0.758732
2	2	1.998605	0.76	0.766	0.75874
2.25	2.25	2.246567	0.76	0.761	0.758636
2.5	2.5	2.496248	0.76	0.767	0.758668
2.75	2.75	2.746626	0.76	0.758	0.758676
3	3	2.998828	0.76	0.761	0.758764
3.25	3.26	3.247205	0.76	0.772	0.758804
3.5	3.5	3.496946	0.76	0.768	0.758788

Table 5 continued: Raw data from channel 2 at 0.76A decrementing load

Voltage Requested	Voltage Displayed (Our supply)	Voltage Displayed (Rigol load)	Current Requested	Current Displayed (Our supply)	Current Displayed (Rigol load)
3.75	3.75	3.747813	0.76	0.77	0.758748
4	4.01	3.996546	0.76	0.759	0.758788
4.25	4.25	4.248214	0.76	0.759	0.758764
4.5	4.48	4.498993	0.76	0.749	0.758764
4.75	4.75	4.746005	0.76	0.767	0.7587
5	5	4.999156	0.76	0.771	0.758788
5.25	5.25	5.245412	0.76	0.758	0.758708
5.5	5.49	5.497659	0.76	0.755	0.758716
5.75	5.76	5.746851	0.76	0.768	0.758684
6	5.99	5.993952	0.76	0.756	0.758676
6.25	6.24	6.247919	0.76	0.751	0.758612
6.5	6.5	6.494368	0.76	0.755	0.758668
6.75	6.75	6.745251	0.76	0.762	0.758644
7	7.01	6.995792	0.76	0.754	0.758748
7.25	7.24	7.245592	0.76	0.758	0.758724
7.5	7.49	7.498343	0.76	0.767	0.758684
7.75	7.75	7.743219	0.76	0.76	0.758692
8	8.01	7.996609	0.76	0.772	0.758724
8.25	8.25	8.245667	0.76	0.758	0.758708
8.5	8.49	8.493184	0.76	0.761	0.7587
8.75	8.75	8.748262	0.76	0.763	0.758652
9	9	8.993673	0.76	0.761	0.758628
9.25	9.26	9.244407	0.76	0.764	0.758652
9.5	9.49	9.496254	0.76	0.768	0.7587
9.75	9.75	9.743786	0.76	0.774	0.758684
10	9.99	9.99744	0.76	0.756	0.758684
10.25	10.26	10.244201	0.76	0.766	0.758724
10.5	10.49	10.49707	0.76	0.757	0.758724
10.75	10.75	10.747063	0.76	0.76	0.758732
11	11.01	10.994328	0.76	0.765	0.758812
11.25	11.24	11.245477	0.76	0.756	0.75874
11.5	11.49	11.493453	0.76	0.754	0.7587
11.75	11.76	11.744365	0.76	0.753	0.758692
12	12	11.996997	0.76	0.766	0.758652

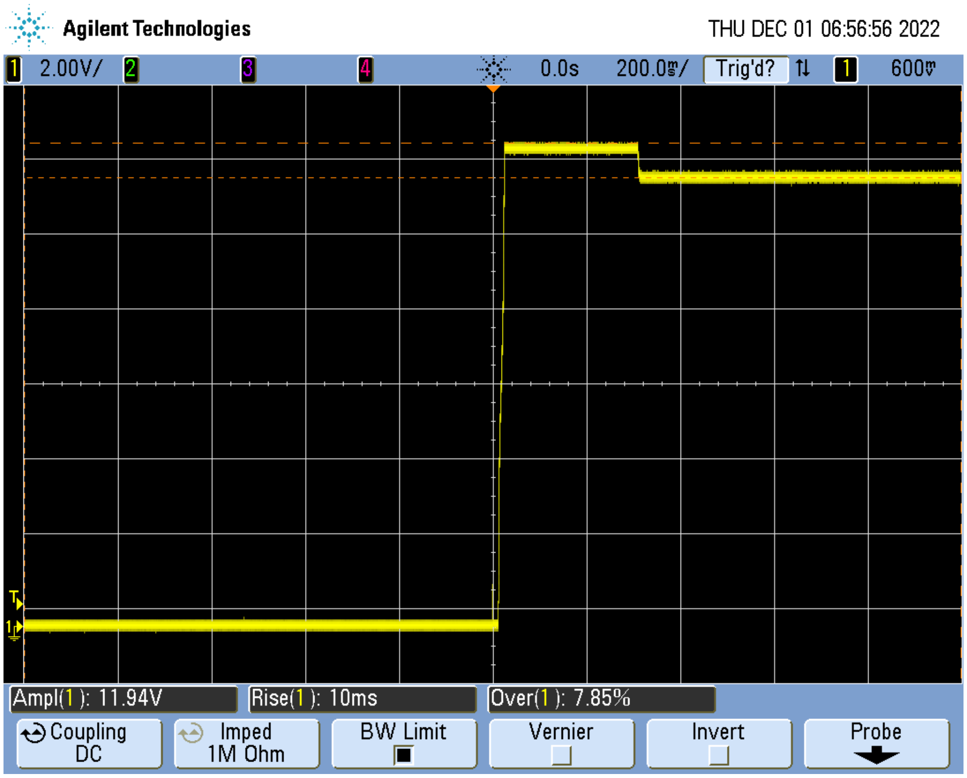
We repeated our first experiment on channel 2 but this time we ran the experiment with the current beginning at 0.8A and dropping to 0A so that the system would start out with the highest load when the device was cool, and the load would only decrease from there. This resulted in us not seeing a significant voltage drift on our ADC, but we still had the issue with the switching regulator at 0.8A.

Table 6: Raw data from channel 1 at 0.72A decrementing load

Voltage Requested	Voltage Displayed (Our supply)	Voltage Displayed (Rigol load)	Current Requested	Current Displayed (Our supply)	Current Displayed (Rigol load)
1	1	0.98778	0.72	0.718	0.719303
2	2.01	1.988581	0.72	0.72	0.719247
3	3	2.992081	0.72	0.721	0.719343
4	4.03	3.992097	0.72	0.73	0.719239
5	10.1	4.995523	0.72	0.732	0.719271
6	6.02	5.99542	0.72	0.725	0.719263
7	6.99	6.996533	0.72	0.728	0.719271
8	8	7.997943	0.72	0.723	0.719335
9	9	9.001621	0.72	0.727	0.719207
10	10.01	10.002334	0.72	0.724	0.719271
11	10.99	11.002735	0.72	0.728	0.719223
12	12.01	12.006443	0.72	0.726	0.719231

We repeated this experiment with channel 1 with decrementing load even though we did not expect it to be any different due to how much faster we were testing this channel.

The next experiment we ran was a qualitative experiment with trial and error to improve the PID response of our PID control of the regulator. The first experiment was done with enough delays that the system reached steady state before measurements were taken and steady state oscillations were similar. Specifically, we ran the first experiment using the same 0.1 coefficient for all PID gains. For this experiment we used an Agilent oscilloscope to trigger on voltage steps so we could see our PID response and get an overshoot and rise time measurement.



THU DEC 01 06:56:56 2022

1 2.00V/ 0.0s 200.0ms/ 1 600mV

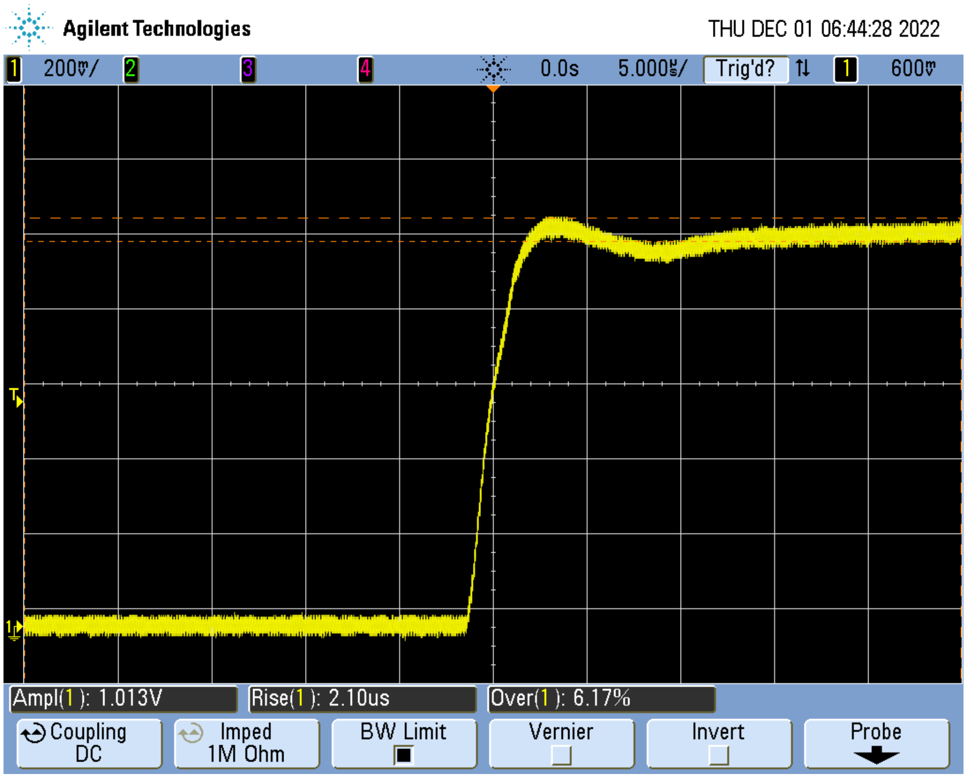
Ampl(1): 11.94V

Rise(1): 10ms

Over(1): 7.85%

Figure 15: PID Response of 12V step at 0.75A

We started out by seeing how our system responded to a 12V step under load, but we could not improve anything significantly. Increasing the P gain much past 0.2 resulted in very visible steady state oscillations. Increasing the I gain much past 0.2 also introduced a small amount of oscillation in steady state but also increased the overshoot. The D gain did nothing to help us but going past 0.5 caused an increase in the overshoot.

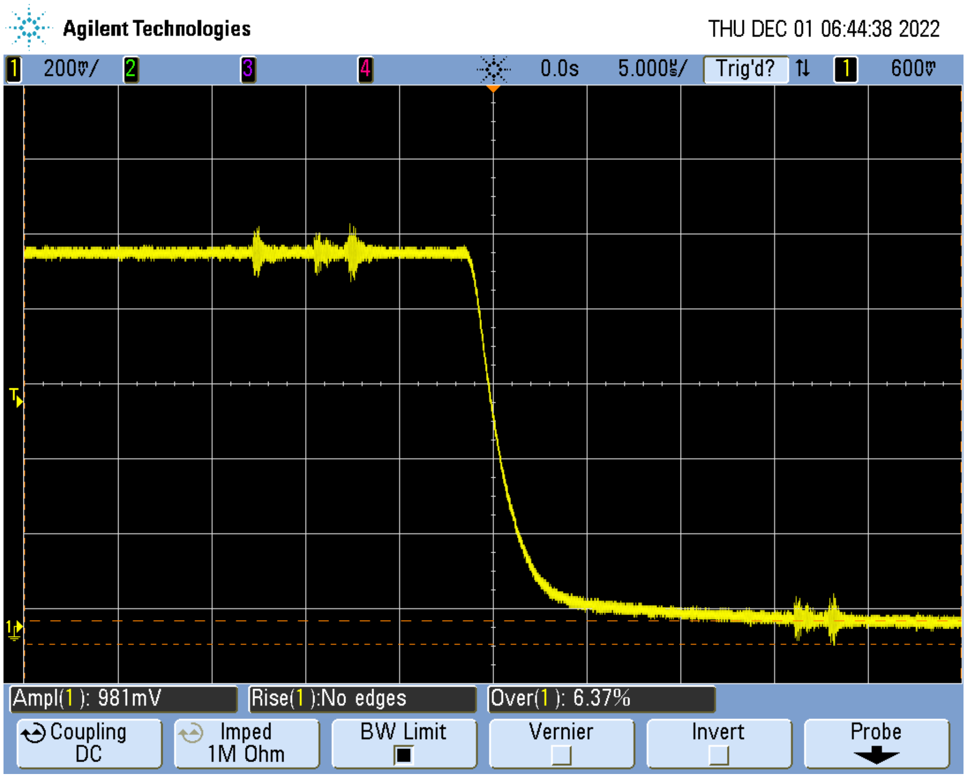


Ampl(1): 1.013V
Rise(1): 2.10us
Over(1): 6.17%

Coupling DC
Imped 1M Ohm
BW Limit ■
Vernier ☐
Invert ☐
Probe

Figure 16: PID Response of 1V step at 0A

We also decided to run this experiment with a 1V step under no load to see how our system would respond without being stressed. Our default 0.1 gain on every term resulted in a large overshoot but not much steady state issues. Removing the D gain entirely removed a good portion of the overshoot and decreasing both the P and the I gains to 0.01 dropped the overshoot to under what our overshoot at full load was as well as decreased the undershoot.



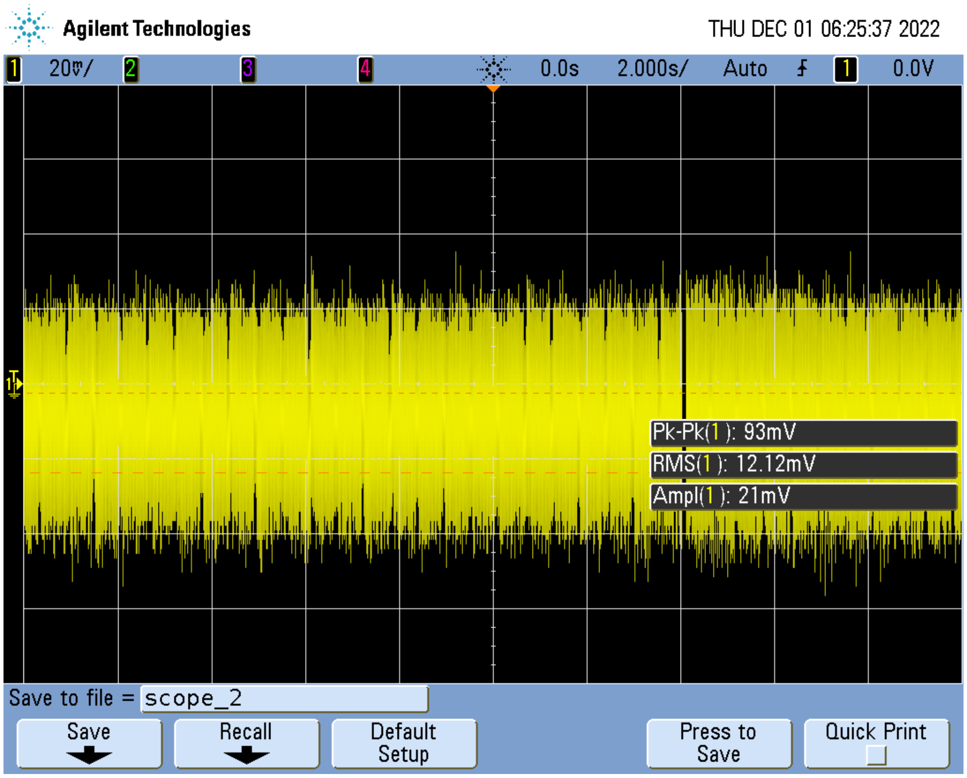
Ampl(1): 981mV
Rise(1):No edges
Over(1): 6.37%

Coupling DC
Imped 1M Ohm
BW Limit ■
Vernier ☐
Invert ☐

Figure 17: Shutdown Response

We also wanted to look at how our system responded to shut down operations. At 12V to 0V steps we saw a nearly perfect transition but at a 1V to 0V step we were able to effectively zoom in and we saw a little bit of noise just before and after shutdown operation. We aren’t entirely sure if this was due to the bad coaxial cable that we were using picking up noise or if this was due to the shutdown transistors themselves.

The final experiment we ran was more of a qualitative measurement as we wanted to measure the output noise of our system. This experiment was done with a non-ideal setup since we only had access to a 10x probe, and we had to hold the probe to our output by hand.



Pk-Pk(1): 93mV
RMS(1): 12.12mV
Ampl(1): 21mV

Save to file = scope_2

Figure 18: Noise levels at 0.75A

We didn’t see a significant difference between the noise levels when running our system under load or not but the noise levels we recorded were much higher than we would expect for our linear regulator.

8. Experiment Validation Using Evaluation Criteria

Our raw data from the first experiment we ran was processed using the matplotlib library in python to create a 3d plot showing how the error rate was affected by the power used by the system. The mean, median, and standard deviation was also calculated for each set of data we took using the same library. The absolute value of the error rate was not taken so that the plots would make sense visually so the average error rates discussed below would effectively be doubled since negative errors would cancel out positive errors.

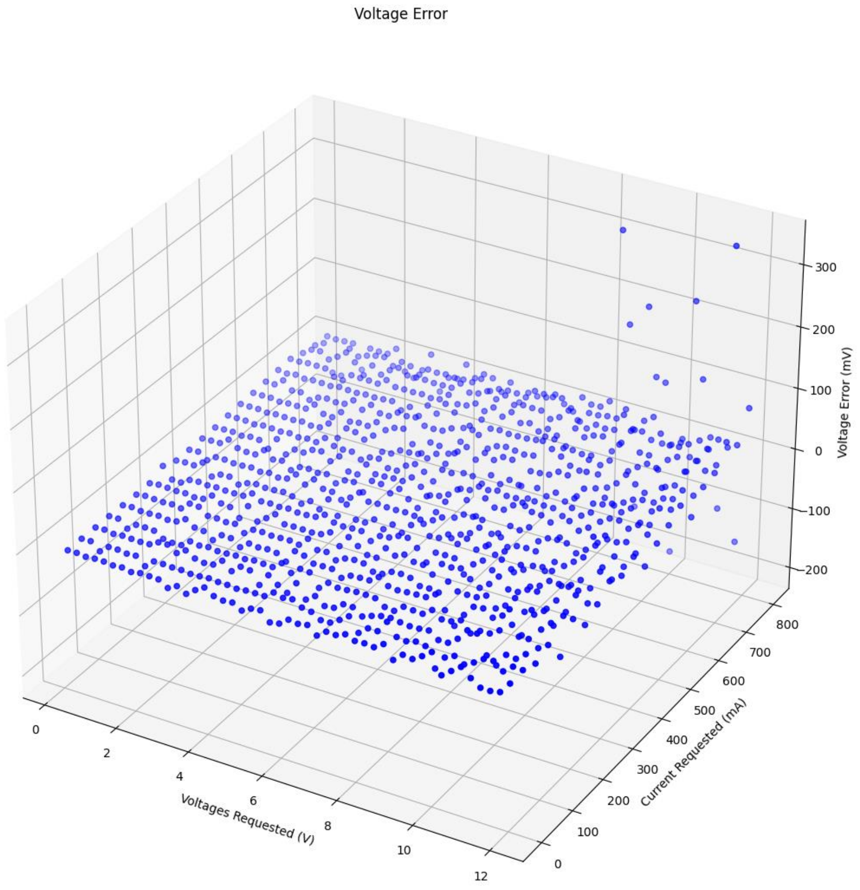


Figure 19: Voltage error with incrementing load on channel 2

Figure 19 shows a nearly flat error surface due to the outliers caused by the failure of the switching regulator at higher loads and the heat built up in the system over the course of this run. Even with these outliers the average error rate was calculated to be 5.56mV with a median error rate of 3.158mV and a standard deviation of 24.57mV. These numbers are acceptable given that the resolution of the voltage we display is 10mV and we remained within around +-10mV accuracy.

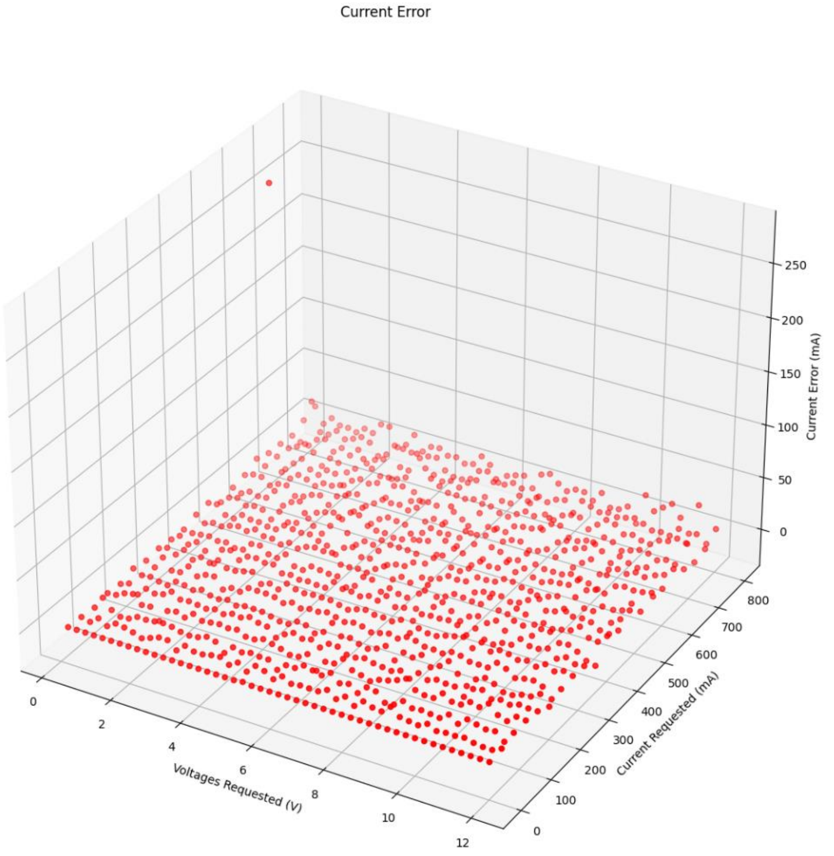


Figure 20: Current error with incrementing load on channel 2

Figure 20 also looks nearly flat just like Figure 15 since there is a strange outlier. The average error rate was calculated to be 2.63mA with a median error rate of 1.81mA and a standard deviation of 9.82mA. These numbers are less acceptable given that the resolution of the current we display is 1mA and we only remained within about +-5mA accuracy.

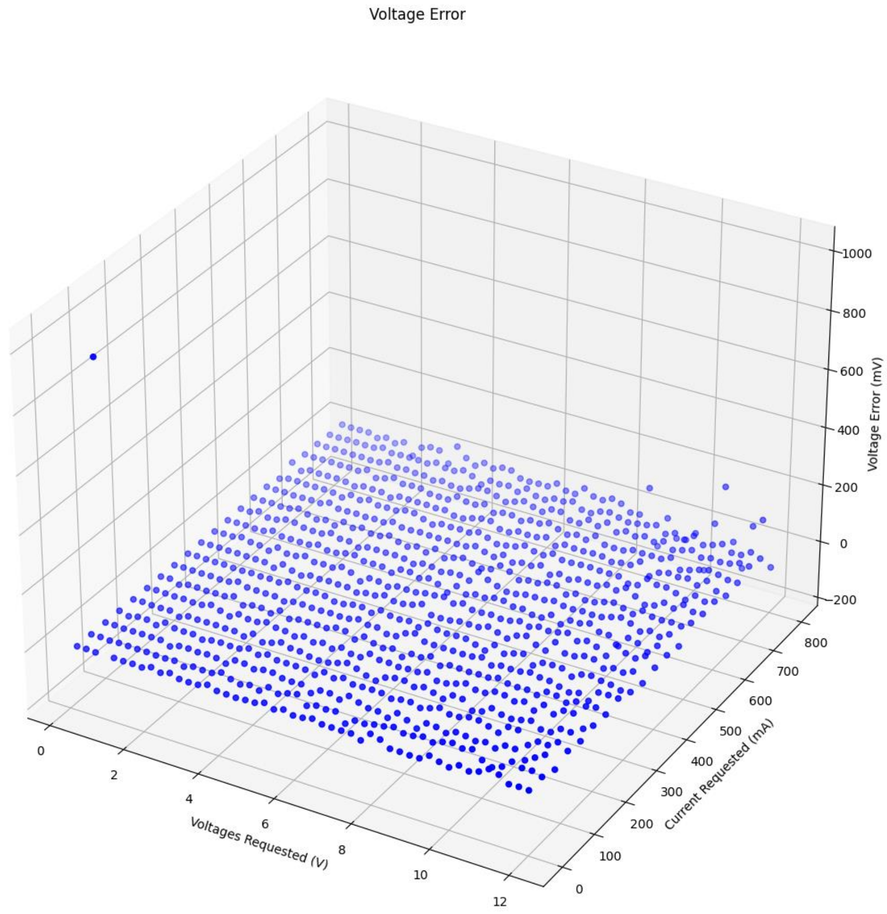


Figure 21: Voltage error with decrementing load on channel 2

Figure 21 looks nearly flat much like Figure 15 due to a strange outlier not caused by the failure of the switching regulator. The failure of the switching regulator can still be seen in Figure 21, but this plot is much flatter since the system did not have a chance to build up a significant amount of heat. The average error rate was calculated to be 0.17mV with a median of –1.47mV and a standard deviation of 34.27mV. These numbers are much better than those of Figure 15, showing that so long as the system does not heat up too much our accuracy is very good. We believe that this is since we moved the voltage divider behind the current sense resistor using a jumper wire that is close to the linear regulator and this wire heating up may change its resistance enough to affect our results.

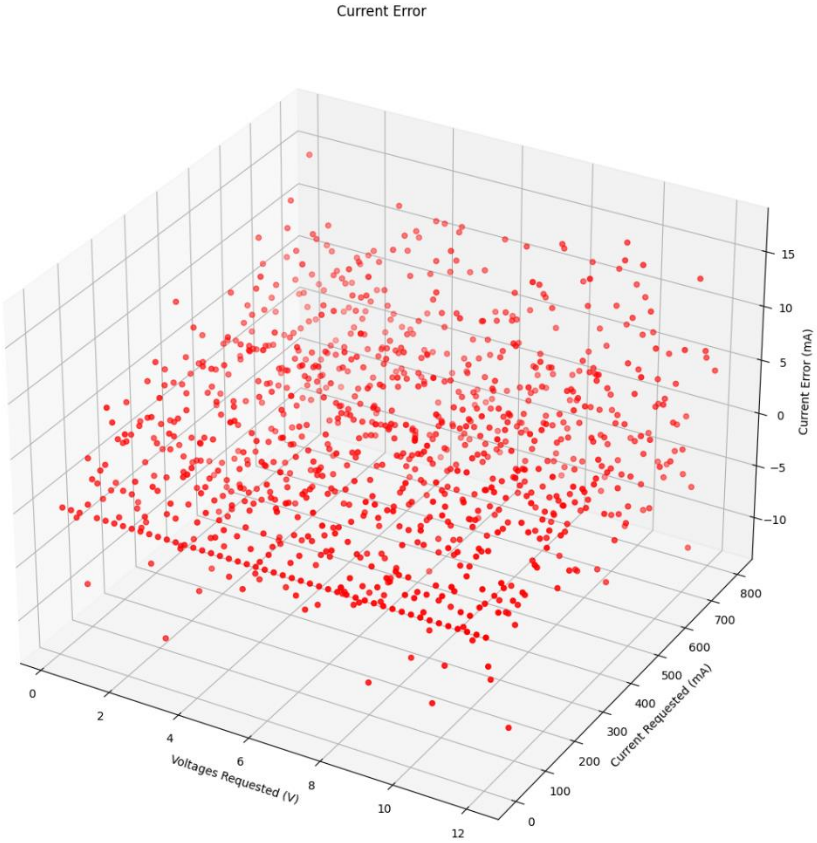


Figure 22: Current error with decrementing load on channel 2

Figure 22 does not look as flat as Figure 16 since there is no strange outlier in the data and therefore error per division is much lower. The average error rate was calculated to be 2.15mA with a median of 1.84mA and a standard deviation of 4.44mA. These numbers are not significantly better than those from Figure 16 for us to believe that heat played a role in the error rate for reading current.

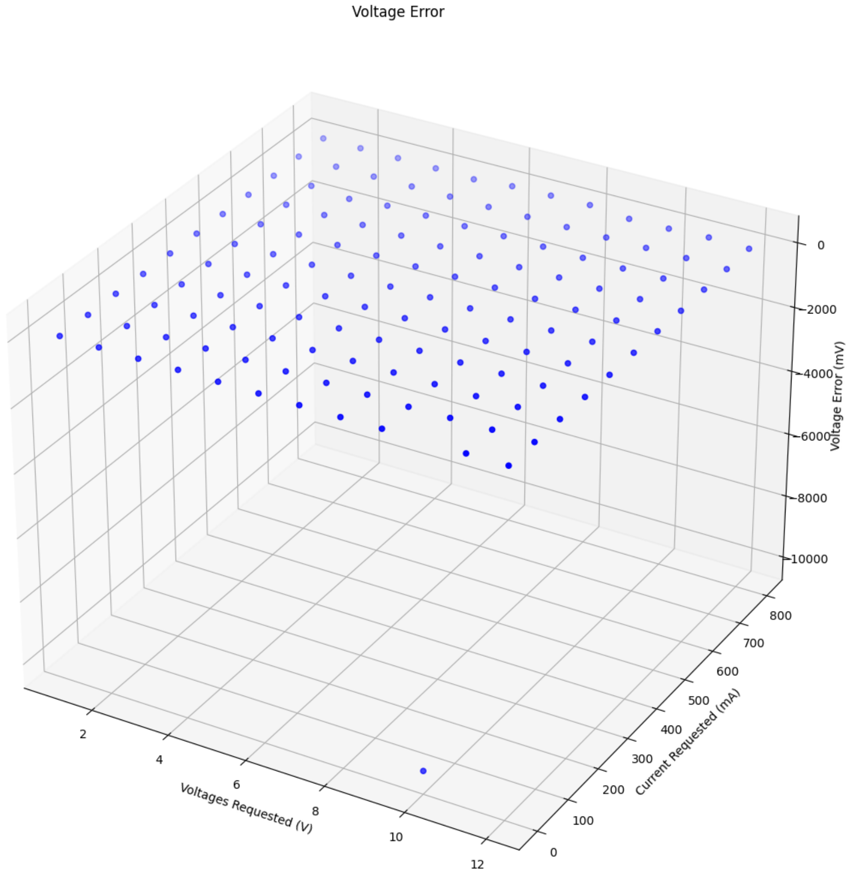


Figure 23: Voltage error with incrementing load on channel 1

Figure 23 looks perfectly flat except for one very bad outlier. We thought we had given the script enough delays to account for the delays in our system, but we found one data point where we read 0V out on our supply instead of 10V. We are not completely sure what the cause of this was, but we believe it to be an intermittent failure in the communication between the two MCUs. With this outlier in place, we have an average error rate of –82.39mV with a median of -7.65mV and a standard deviation of 871.92mV. With this outlier removed we calculated an average error rate of –6.63mV with a median of –7.65mV and a standard deviation of 17.76mV. These numbers aren't too different from those of channel 2 but the error rate trended in the negative direction rather than the positive direction for some reason.

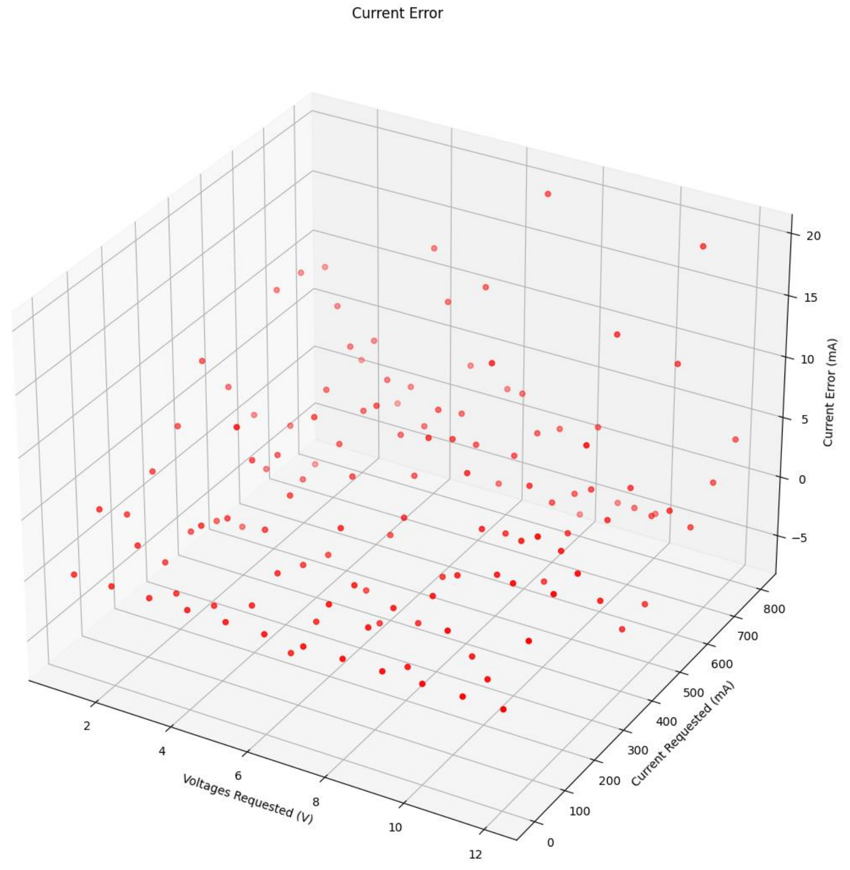


Figure 24: Current error with incrementing load on channel 1

Figure 24 looks much like the previous current error plots with no outliers. The average error rate was calculated to be 2.64mA with a median of 2.16mA and a standard deviation of 4.49mA. These numbers are in line with the numbers we got from channel 2.

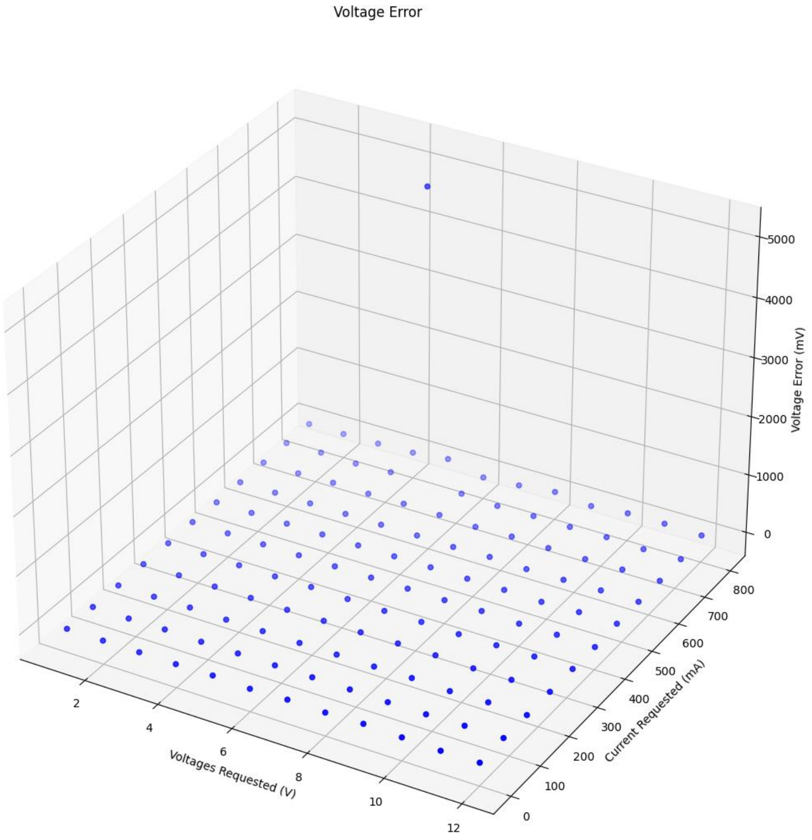


Figure 25: Voltage error with decrementing load on channel 1

Figure 25 looks much like Figure 23 with a strange outlier where the voltage was displayed as 10V instead of the 5V it was at. With this outlier we calculated an average error rate of 37.69mV with a median of –3.10mV and a standard deviation of 444.80mV. With this outlier removed we calculated an average error rate of –0.95mV with a median of –3.10mV and a standard deviation of 19.55mV. Again, these numbers are like those from channel 2 but trended in the negatives.

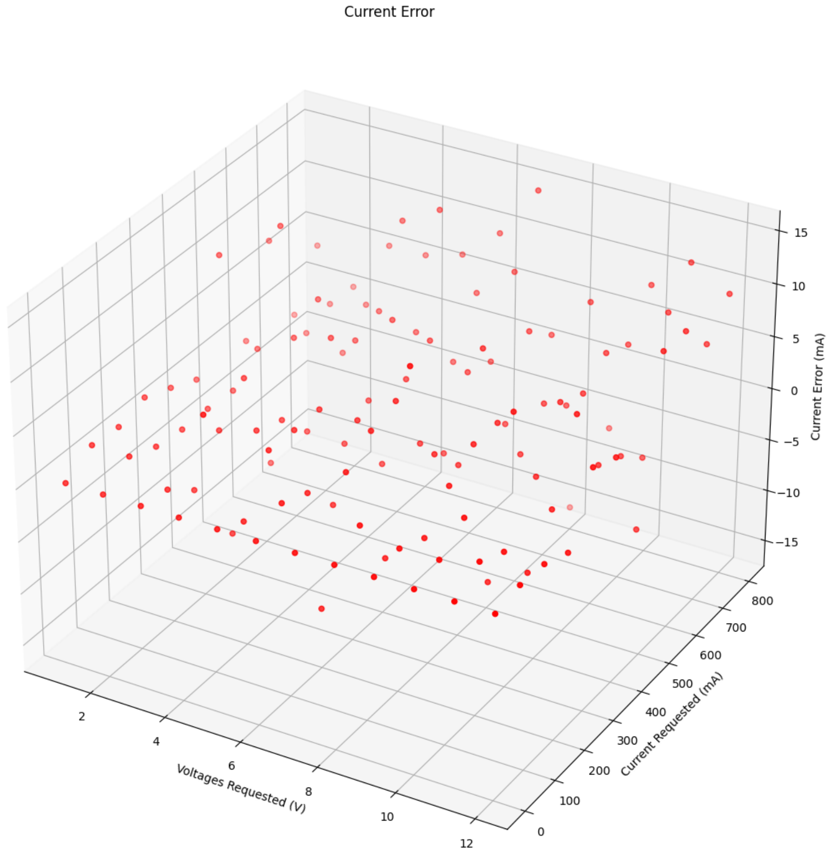


Figure 26: Current error with decrementing load on channel 1

Figure 26 looks just like every previous current plot with no outliers. The average error rate was calculated to be 2.26mA with a median of 1.54mA and a standard deviation of 4.51mA. These numbers remain in line with those from channel 2.

The noise levels we measured on our supply were done with a 10x probe on 10:1 setting with the oscilloscope, but our noise levels were quite high, if we can assume we did something wrong and can divide our results by 10 we end up with a very acceptable 9.3mV peak-to-peak ripple. A peak-to-peak ripple of 93mV would be something we expect out of a switching regulator not a linear regulator. If we did things correctly and the oscilloscope was providing accurate information, then a possible explanation for the noise levels may be the inductor for the switching regulator since it operating outside its frequency range was audible to us. It is also possible that we simply weren’t measuring the noise correctly and were picking up some other noise in the area.

The success of our project is evident from the exceeding of our mission requirements. Initially, our requirements included: the operation of two independent voltage channels, output voltage per channel from 0-10V, maximum current per channel of 500mA, budget of $50-100, custom front panel to display voltage and current settings to the user, control of the device via USB, keypad, and/or knob, and measurements and outputs displayed with high accuracy.

Figure 27: Voltage Chaining for 24V

Our final Output voltage was raised to 0-12V which should allow more flexibility to the user in setting a desired voltage. Additionally, because of channel separation the chaining of the voltage channels would now allow for +-12V to the max or min, higher than our previous +-10V, seen in image 1. Similarly, we extended the range of our max current from 500mA to 800mA per channel, the end goal of this was to allow the user more flexibility by allowing slightly higher max settings.

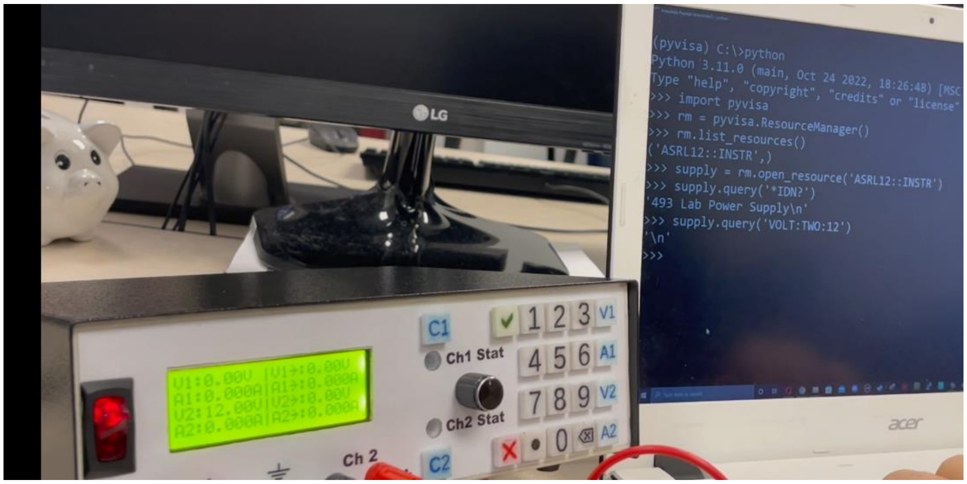
Budgeting was restricted to a $50-100 range, of course because of the semiconductor shortage the likelihood of following those constraints could change at any moment [1]. We observed this from the rise of price in our linear regulators, for example, the LT3083 which rose from ~$8 to ~$12 near the beginning of the Spring 2022 and the end of the Fall 2022 semesters respectively. This is part of the reason we created two design plans for our final design, plan A included the LT3080 with added latch up protection and plan B would only utilize the LT3083. In the end, we calculated the total price of each build using the price for a thousand units for each component. Plan A resulted in $98.88 and plan B $106.07. While plan B had a slightly larger difference to the max $100, we believe it is in respectable range for the requirements, as it did not account for price increases with time and the requirements were exceeded.

Figure 28: Front Panel Display

Our front panel, Figure 24, was designed to display all inputs and outputs required. Initially, we used a smaller enclosure that could barely house outputs and keypad with a tiny LCD and then moved on to a larger enclosure that was more than twice the size. Through this transfer we were able to utilize a larger LCD with a backlight, distribute the space between voltage outputs correctly, display the channel output status via LEDs, and feature a rotary encoder and keypad input. Custom keypads and a spacer were designed to easily integrate them onto the metal plate and maintain functionality. A white laminate was also used to give a clean and easily readable interface for the user. All the holes were drilled then smoothed in the MiX.

```
(pyvisa) C:\>python
Python 3.11.0 (main, Oct 24 2022, 18:26:48) [MSC v.1933 64 b
Type "help", "copyright", "credits" or "license" for more in
>>> import pyvisa
>>> rm = pyvisa.ResourceManager()
>>> rm.list_resources()
('ASRL12::INSTR',)
>>> supply = rm.open_resource('ASRL12::INSTR')
>>> supply.query('*IDN?')
'493 Lab Power Supply\n'
>>> supply.query('VOLT:TWO:12')
```

Figure 29: SCPI Change Output Voltage Prior



```
(pyvisa) C:\>python
Python 3.11.0 (main, Oct 24 2022, 18:26:48) [MSC
Type "help", "copyright", "credits" or "license"
>>> import pyvisa
>>> rm = pyvisa.ResourceManager()
>>> rm.list_resources()
('ASRL12::INSTR',)
>>> supply = rm.open_resource('ASRL12::INSTR')
>>> supply.query('*IDN?')
'493 Lab Power Supply\n'
>>> supply.query('VOLT:TWO:12')
'\n'
>>>
```

```
V1:0.00V |V1→:0.00V
A1:0.000A|A1→:0.000A
V2:12.00V|V2→:0.00V
A2:0.000A|A2→:0.000A
```

Figure 30: SCPI Change Output Voltage Subsequent

Programmability via USB was implemented using SCPI commands with a slight twist on the setting commands so that the channels would be controlled separately. This was not exactly standard, but we saw that many manufacturers also have their own twist on SCPI commands, and we figured it was okay. All the code for accepting SCPI commands is done via string comparisons so modifying them would be simple.

9. Additional Issues

The purpose of our project was to create an inexpensive DC power supply that had the input functionality and programmability of other power supplies currently on the market, but at a more affordable price than said power supplies. This would primarily benefit students by providing a power supply they can use for electronics labs, but it would also benefit hobbyists by giving them a means to power their custom circuits or systems. The impact of our success to create an inexpensive two channel DC power supply (even though we went over budget by a few dollars) would impact students and hobbyists by saving them money that could be spent on more expensive supplies with similar functionality.

Since our finished product works, it could be used by students or hobbyists effectively. As for the next steps we would take, we were planning to make our project open source by making our project's GitHub repository public, allowing anyone to access our schematic/code and therefore build their own version of our power supply. This would benefit students by giving them a blueprint of a working power supply they could assemble at a low cost for their schoolwork while also giving hobbyists the same blueprint as well as the means to modify it for their own desired circuit specifications. Since we would be making our GitHub repository public, anyone could access our project's details from anywhere around the world if they have internet access, giving our project a global range of potential users.

The cost of our finished product was $106.07, but the overall amount of funds spent during this project (for parts we used, parts we did not use, and PCB/Front Panel button fabrication) was approximately $1,087.28. The most expensive components bought for the project were the regulators, as the total cost of all the regulators we bought was about $397.09. As for the cost of labor for this project, assuming everyone was paid $20 per hour for working on the project, the total amount is 867 * 20 = $17,340. Overall, the cost of the project would be $18,427.28.

There were several problems with our implementation of this device that could and should be fixed in a second revision. Most of the mistakes could be fixed on our first revision board using jumper wires, for example we had several pull-up resistors that were accidentally set up as pull-down resistors and we had to jumper those pads to some unused 3.3V pins on the top using a wire with a resistor in it. There were also some mistakes with the pinout of some of the headers having pins reversed causing some confusion when wiring up the display. Including silkscreen labels on each pin would not be a bad idea either. There were also a couple mistakes with the value of resistors on our OpAmp circuits which were easily fixable by replacing said resistors.

Additionally, we made a large mistake with the inductor selection for our L6902D switching regulator and that should be replaced with an appropriate inductor at the minimum.

This mistake caused the switching regulator to fail when operating at low voltages and high currents, so we ended up needing to disable control of the switching regulator and have it simply pass the 16V to the linear regulator which now must do all the work. It would also be a good idea to simply replace our L6902D switching pre regulator entirely since we upgraded our transformer to one that supports up to 1.5A. The L6902D has a current limit of 1A that cannot quite be reached on the output due to other current draws. A switching regulator that supports at least 1.5A output with a programmable current limit should be chosen in its place, this would allow us to output up to 1.2A comfortably with additional changes to resistors and fuses as necessary.

We also placed the resistive divider to read the output voltage after the current sense resistor but did not move it after adding a fuse to the end of the output for reverse polarity protection with a flyback diode, this causes the math that needs to be done to measure the outputs to be needlessly complicated. This divider should either be moved behind the current sense resistor or buffered with an OpAmp. A transistor could also be added in series with the fuse to provide reverse current protection now that we have a more powerful transformer and therefore more voltage to work with.

If the PCB we designed could be shrunk down to fit in a smaller and cheaper enclosure with the large transformer still off to the side to further save on cost and a smaller and cheaper TFT LCD was chosen, then we could be well under our $100 budget while maintaining our higher outputs.

Since the upgrade to our transformer was cheap, it may be possible to find a transformer with an even higher voltage and current limit for not much more money. The LT3083 is a somewhat expensive device with a voltage limit of just over 20V and a current limit of 3A which wouldn't quite be reached due to current draws from the sense module but it is a very reliable and easy device to use. With another LT3083 in parallel you could output 20V 5A if a suitable transformer was selected along with other cost saving measures mentioned above to counteract the price of another regulator per channel and a more powerful transformer.

The maintainability of our project from a hardware perspective is great, as anyone can simply unscrew the sides of our power supply's case to access our circuit and replace any parts that may need maintenance, whether it is a button on the front panel, a regulator, or even the transformer. Additionally, if the internal PCBs or wiring was damaged, the project's schematics could be accessed from our GitHub repository, allowing replacement PCBs to easily be edited and manufactured as needed. Maintainability from a software perspective is also great, as our channel PCB contains pins that allow anyone to use an ST-Link to program both channels' microcontrollers independently to perform any desired updates. Similarly, if either channel's microcontroller breaks and must be replaced, the code for each channel can be found in the Git, so reprogramming a replacement microcontroller would not be difficult.

When our product reaches the end of its lifespan, the components will have to be dealt with in separate ways. The metal case can be taken to a metal recycling facility while the transformer will have to be disassembled to allow the copper wiring to be recycled, with the rest of the transformer being thrown out afterwards. The 3D printed buttons will have to be thrown out as will the individual components on the PCBs, but the PCBs themselves can be melted down for copper that can be recycled. As for the documentation required to create a new instance of our power supply, all that is needed is the PCB footprint and the code, which can be accessed from either the MCUs on the old PCB, or the GitHub repository dedicated to our project. Additionally, a 3D printer and a drill press will be required to create the button caps/stencil for the front panel, and the holes required for the front and back panels.

10. Administrative Part

While we had a few delays affecting this project, namely finding a proper way to interface with the MCU, fabricating our prototype channel circuit, and designing the PCB taking longer than initially expected, we were able to successfully complete not only a working circuit, but a proper prototype within an enclosure that fulfills most of our requirements. We were able to complete all our requirements except for the cost requirement (final cost should be between $50 and $100), as our final cost was $106.07.

We had to change the design several times to account for part failures or parts running out of stock. We changed the regulator we used from the LT3080 to the LT3083 due to the LT3080 latching on either its minimum voltage or its maximum voltage when there is more than a 300mV difference between the set pin and the out pin, which caused issues with our regulation circuit. We also had to change the specific diodes used for the charge pump, as the original diodes we planned to use went out of stock before we ordered them for our PCB, but the only difference between the new and old ones was that the new ones had a slightly higher voltage drop, which had negligible impact on the performance of our circuit. We had planned for most of our components to go out of stock and make sure to order plenty of our MCUs and regulators in advance, but we never expected diodes to go out of stock.

We also had to change the schedule of our project several times due to the delays mentioned before. While we originally planned to start on the code within the first couple weeks of the project since that would be the backbone of our regulation circuit, we were unable to make substantial progress coding until April as we could not get the program STM recommended for programming our microcontroller to work properly. Whenever we tried to program something with STM32CubeIDE on our MCUs they would work momentarily, crash, and restart. We solved this problem by using another IDE, Keil, instead. Keil worked great for us until we ran into the 32KB flash limitation but luckily at that point STM had released an updated version of STM32CubeIDE and a new flash update for the ST-Link which worked for our MCUs. We also planned to finish prototyping before the Spring 2022 semester ended so we could design our PCB over the summer, but we ended up designing the PCB in September and October instead due to the creation and testing of our single channel prototype taking the entire summer as well as the first couple weeks of the semester to finish.

The most significant “unexpected” activity we had to carry out was the creation of a modular protoboard to test our prototype. The initial protoboard we made broke immediately and we could not figure out which regulator or connection was the issue, so we had to remake the whole circuit from scratch with new components in a modular fashion to isolate the problem, which turned out to be connection issues with the switching and linear regulators. Additionally, when we originally tried to drill holes for our case at the MiX, the staff there told us we needed a cutout of the hole positions to put on the case to help us aim the drill press properly rather than the drawn-on holes we made on the case, which took about two hours to design and properly space out and ended up being slightly off anyways due to the inaccurate datasheet provided for the enclosure.

The overall price of our finished product was $106.07, and the funds spent on the project overall can be observed in the following table. These funds include the purchase of parts that were used on the finished product, PCB fabrication costs, and the purchase of parts that ended up not being used in the final design. More funds were spent than originally expected on this project.

Table 5: Funds spent

Member	Funds Spent
Carlos	625.50
Dean	98.75
Juan	131.31
Tarun	231.72

Total man-hours spent on the project can be observed in the following table. Man-hours were calculated both from time spent in meetings as well as time spent working on components on our own time.

Table 6: Man-hours

Member	Man Hours
Carlos	254
Dean	215
Juan	204
Tarun	194
Total	867

11. Lessons Learned

The primary knowledge we needed to learn to make our project work was how to properly interface with STM microcontrollers, since the only microcontrollers most of us have worked with previously was the MSP430 in ECE-447, which uses both different syntax in the code and different programs to interface with the user. Additional knowledge was learned about PCB design, PCB population, and how to interface with a system serially using PyVISA, which was required to fulfill the SCPI requirement. Additional skills learned during this project were time management and persistence. While time management was important for structuring when we would work on components of the project and who would work on said components, persistence was a constant skill that was used throughout all elements of our project, such as when we diligently searched for replacement diodes for our PCB when the ones we originally chose went out of stock or when we tried to find a reliable way of programming our microcontroller when the program that STM recommended (STM32CubeIDE) kept crashing on us. We also had several problems with the HAL library provided by STM, for example, when you want to use the ADC with the DMA to scan multiple channels in a non-blocking way, the HAL library will initialize the ADC and then the DMA but since the ADC relies on the DMA it will not work, and you must manually de-initialize both peripherals and then initialize them in the correct order [9].

Furthermore, there was an ample amount of experience gained from reading datasheets and using them correctly. For example, our linear regulator datasheet laid out what must be done to allow for true-zero output voltage, which we then used to filter out possible solutions till arriving at our final charge pump circuit. Similarly, necessary track widths were detailed on each regulator data sheet which we tuned for our application, and our STM32L1 family datasheet showcased the correct implementation of our IC’s pins like capacitance values. Additionally, The problem we had with our inductor can also be attributed to our datasheet reading, as the inductance value (although being the correct 22uH) would be different depending on the test frequency, and since our switching regulator operated at a fixed 250kHz frequency our inductor value would be erroneous as the value was measured at a test frequency of 2MHz.

The teaming experience was bumpy at first, as we had to produce appropriate meeting times that worked around everyone’s schedule as well as deal with a member who did not show up to our meetings or contribute to our project in anyway, but after a couple of meetings and the member who did not contribute dropping out of our project, teaming went very smoothly. Some members did not come into their niches until the latter half of the project, such as Dean not working on a front panel design halfway through the second semester due to the final case not being selected yet as well as the front panel PCB not being designed yet. However, every team member had a part of the project they specialized in and could communicate with other team members to synergize their respective components of the project for the final product.

12. References

Literature References

[1]

J. Voas, N. Kshetri, and J. F. DeFranco, “Scarcity and Global Insecurity: The Semiconductor Shortage,” *IT Professional*, vol. 23, no. 5, pp. 78–82, Sep. 2021, doi: 10.1109/MITP.2021.3105248.

[2]

“Multiple Channel Power Supplies_Products_SHENZHEN KORAD TECHNOLOGY CO., LTD.” https://www.koradtechnology.com/product/8/ (accessed May 10, 2022).

[3]

“DC Power Supply | Siglent SPD3303X-E DC Power Supplies,” *Siglent*. https://siglentna.com/power-supplies/ (accessed May 10, 2022).

[4]

D. by A. S. | 旭亞系統設計(股)公司 | www.akacia.com.tw, “DC Power Supplies-Gwinstek.” https://www.gwinstek.com/en-US/products/layer/DC_Power_Supply (accessed May 10, 2022).

[5]

Keysight, “DC Power Supplies,” *Keysight*. https://www.keysight.com/us/en/products/dc-power-supplies.html (accessed May 10, 2022).

[6]

“DC Power Supply | Variable Power Supplies | Tektronix.” https://www.tek.com/en/products/keithley/dc-power-supplies (accessed May 10, 2022).

[7]

“DC Power & Loads DC Power Supply & Electronic Load Families | RIGOL.” https://www.rigolna.com/products/dc-power-loads/ (accessed May 10, 2022).

[8]

“LT3080 wierdness - dave’s power supply (eev224) gone mad - Page 1.” https://www.eevblog.com/forum/projects/lt3080-wierdness-dave_s-power-supply-(eev224)-gone-mad/ (accessed Dec. 03, 2022).

[9]

“Set up multiple ADCs on STM32 microcontrollers using DMA - YouTube.” https://www.youtube.com/watch?v=AloHXBk6Bfk (accessed Dec. 03, 2022).

People Contacted

Dr. Jens-Peter Kaps (jkaps@gmu.edu)

13. Appendix A: Proposal

Executive Summary

Since the Covid-19 pandemic, there has been a sharp increase in the price of semiconductor chips and related technology [1]. This has even been experienced in the ECE department at George Mason University where the Analog Discovery 2 (AD2), which became required equipment, saw an increase in price. Another example is the increase in price of high-end power supply units (PSU), which have grown significantly more out of reach; the power of these equipment stems from their ease of use, programmability, number of channels, and resolution. To replicate such equipment at a low cost would require some sacrifice in specifications, especially in channels and resolution, although this will remain a competitive product, should it offer similar functionality at a much lower cost. This proposal seeks funding for a five-member group, to create an affordable dual channel switched-mode power supply which can be controlled physically or via Standard Commands for Programmable Instruments (SCPI), effectively providing similar functionality of an average high-end PSU.

Proposed Solution

This PSU will be built for affordability while maintaining the features of high-end PSU. Therefore, this design will be powered through line voltage, with each channel outputting voltage and current in ranges of 0-10V and 0-500mA respectively. The PSU will be controlled via dual ST microcontrollers (STM32L152RC/B) and feature keypad and/or rotary encoder input. Overcurrent protection (OCP) will be adjustable alongside the voltage through the previously mentioned inputs, within one millivolt/milliamp range. All these settings will be displayed on the front panel either through LED markers or LCD and be available for channels one and two. Additionally, the power supply unit will feature dual polarity, allowing the ability to supply positive and negative voltage simultaneously. Finally, the PSU will also feature control through SCPI by serial input and all of this will be created within a $50-$100 budget in mind.

Problem Statement

Motivation

The COVID-19 pandemic in 2020 significantly increased the number of online classes, including those in the electrical and computer engineering field. As many of these classes contained a lab component, students were required to purchase lab power supplies that could be powered and used at home to run experiments for these labs. These lab power supplies would be used in both AC and DC situations, although most labs required only DC voltages/currents to be used. The power supply model primarily used by GMU, the Digilent Analog Discovery 2, sells for around $400. However, while the lab power supplies on the market were suitable for students working with DC and AC circuits, students only working with DC voltages had no option for a power supply that would output DC at levels suitable for their labs that had a low cost, instead having to purchase more expensive power supplies such as the AD2 that contain functionalities they wouldn't need for that lab. Our design seeks to fill that void and provide an inexpensive lab power supply that would only supply DC voltage, letting students pay less for the same functionality (for the purposes of their lab).

Identification of Need

To design a programmable two-channel power supply and meet our project goals and requirements, we must achieve the following:

- The device must be computer controlled
- Output voltage at each channel to be adjustable between 0-10V
- Developed solution must be low-cost ($50-$100)
- A custom front panel to display voltage and current information to the user
- Device must be controllable from a computer (via USB) and using a keypad and a knob
- Measurements and outputs displayed with high accuracy

Market/Application Review

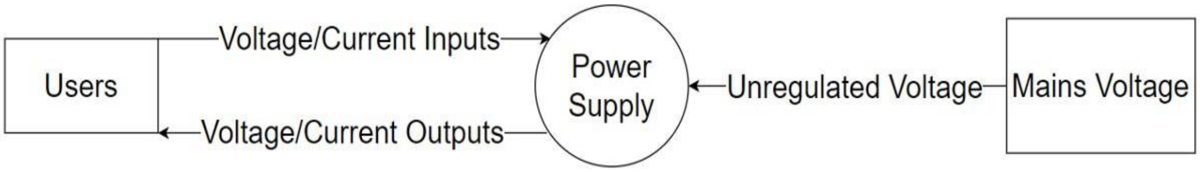
PSU offering SCPI capability, two-channels, and millivolt/milliamp resolution are commonly found at upwards of $200, e.g., the Siglent SPS5044X, SPD3303X (3-channels), Rigol DP821A are priced currently at $2,269, $539, $731 [2, 3, 4], respectively. It must be mentioned that while all feature SCPI, multiple channels, and millivolt/milliamp resolution, each device's respective channel output, wattage, etc. are different. Comparing the lowest price ($259) to the budget (50-$100), it becomes reasonable to consider the proposed solution, which maintains similar features although at less power. Additionally, it must be mentioned that here at George Mason University, most undergraduate Electrical Engineering labs will not exceed 5 watts of power, this is because they were recently designed with the AD2's power supply in mind, i.e., 0.5 to 5V, -0.5 to -5V output voltage, 2.1W maximum power, and 700mA maximum current per channel [5]. Therefore, our PSU can be a viable complementary device to students, especially with both channels outputting simultaneously.

Approach

Problem Analysis

There is currently a gap in the market for an inexpensive two channel DC power supply, so we aimed to fill that gap. We wanted to create a system that would allow the user to input a voltage between 0 and 10V and a current limit between 0 and 500mA for each of the two channels independently. The power supply would then output said voltage at a current below the limit and display both the user input voltage/current values as well as the actual output voltage/current values for each channel. The power supply would also be connected to Mains voltage as a power source.

External System Diagram



Approach based on Conceptual Design

The conceptual design for this project began with the requirement specifications of a 2-channel lab power supply that could be controlled via both a knob and a keypad in addition to USB. With the requirements of a keypad and USB access, we immediately knew that the system would need to be controlled by a microcontroller. Another requirement was that the two channels would need to be electrically independent, so controlling both channels with only a single microcontroller is not possible. With this requirement, we knew early on that we would need at least two microcontrollers to control each channel separately and that they would need to be able to speak to each other in an electrically isolated way, such as with an optocoupler. Since everything is controlled with microcontrollers, every input needs to be digital, so the knobs will be rotary encoders rather than analog potentiometers. The rest of the design was based on other lab power supply front panels as well as a sample front panel design given to us by Dr. Kaps.

Alternative Approaches

One alternative design was to use a single MCU as a centralized controller, allowing us to use analog sensors along with a corresponding ADC on each channel for communication. The ADC and MCU would communicate through a full-duplex communication protocol such as SPI, but the two would be on separate circuits. The central MCU would be on its own isolated and regulated voltage rail designed to meet the specifications of the MCU for optimal safety and reliability while the sensors and their ADCs would be placed on each voltage channel to measure the voltage and current for that channel. To control the output of the channel, a digital potentiometer, PWM signal, or a DAC would be used to interface with the voltage regulation circuit of that channel, depending on the design of the circuit. These components would similarly communicate through the same full-duplex protocol as the other sensors on its channel. However, since the channel and MCU are electronically separated, an alternative means of communication is needed. This would be accomplished through the optocouplers, as it is in the selected design. This design was not chosen as getting the individual sensors, ADCs, and DACs would end up costing more than a single MCU that has all these features built into it. Furthermore, the central MCU would need higher capabilities and performance since it would have to control all channels simultaneously. Using a single MCU per channel instead would reduce the load on the central MCU, thus allowing for a broader distribution of resources.

An alternative design choice for specifically the voltage regulation circuit in each channel was to use a power electronics microchip instead of assembling individual components to build a regulating circuit. A suitable microchip that can both control and drive the outputs for the voltage channels could not be found.

An alternative design choice for input power was to use a power supply that was designed for use in a desktop personal computer. This choice would eliminate any risk from dealing with Mains voltage as we would not need to open the supply to access the 3.3V, 5V, 12V, and ground pins and it could be treated as a black box. This choice would simplify the circuit slightly by removing the need to rectify AC power, but it comes at an increased cost, as a PC power supply costing around $50 or more while a suitable transformer can be found for under $20.

Several approaches to regulating the supply voltage were considered. The first approach considered was to use a buck-boost converter to achieve a user desired output voltage and the current limit would have to be dealt with by external circuitry. A buck-boost converter is a switching regulator, so the output has a lot of ripples, which is undesirable in the context of lab experiments. To achieve a stable output voltage, we decided that the regulator circuit should at least include a linear regulator at the end. The first approach was further modified as we did not see a need for a boost converter if the transformer is able to supply the necessary DC voltage after rectification. For this reason, we specifically look for parts with low voltage drops like Schottky diodes and LDO regulators. If the transformer is unable to supply an adequate voltage after rectification, it may be necessary to use a buck-boost or simply a boost converter, but for now, a more specialized buck converter switching regulator with an adjustable current limit seems to be the better choice.

Introduction to Background Knowledge and Phenomenology

The brains of the power supply unit we are developing are the microcontrollers, which are responsible for handling user inputs, processing them, and delivering the output voltage/currents based on said inputs. Microcontrollers are computers on integrated circuits designed for use in embedded systems, which are systems that perform a specific function.

To transform Mains voltage into a voltage that can be safely used by our system, a step-down transformer will be used, which is a component used to convert high voltage at a low current into low voltage at a high current. A rectifier, which is a circuit that converts alternating current voltage into direct current voltage, will be used to convert the stepped-down voltage into DC voltage which will be led into the regulator circuit. The regulator circuit will be used to create a fixed voltage that the microcontrollers and the output channels will draw voltage from.

The regulator circuit that will output the voltage sourced by the rest of our system will be composed of three types of regulators: linear, switching, and static. Linear regulators are regulators that utilize a linear component, such as a potentiometer, to regulate input voltage. These types of regulators are relatively simple to design and do not output much noise, but they have a low power capacity and are not as efficient as other regulators. Switching regulators use switching components, such as MOSFETs, to regulate input voltage. These kinds of regulators are efficient and can handle high amounts of power but are also more complicated to design and have relatively noisy output signals. Static regulators output a constant voltage value that can’t be changed, which makes them ideal for powering a device that needs a specific constant voltage, such as a microcontroller, but not for powering something that requires a variety of voltage values, such as the output channels for our PSU.

Project Requirements Specifications

Mission Requirements

- The device will operate two independent controllable voltage channels

Operational Requirements

Input/Output Requirements

- The output voltage and current limit per channel shall be adjustable using a keypad and a knob (rotary encoder)
- Voltage, current, and current limit input/output per channel shall be displayed on a simple LCD
- Output voltage shall be adjustable between 0-10V
- Power supply will have the ability to set current limits per channel.
- Power supply will be controllable through USB using SCPI
- Power supply will offer two channels

External Interface Requirements

- Power supply shall receive power from Mains power

Functional Requirements

- Maximum Current per Channel shall be 500mA
- If the current limit is reached, the output will switch off
- Digital readouts for voltage, current, and current limit per channel shall be shown if the current limit was reached

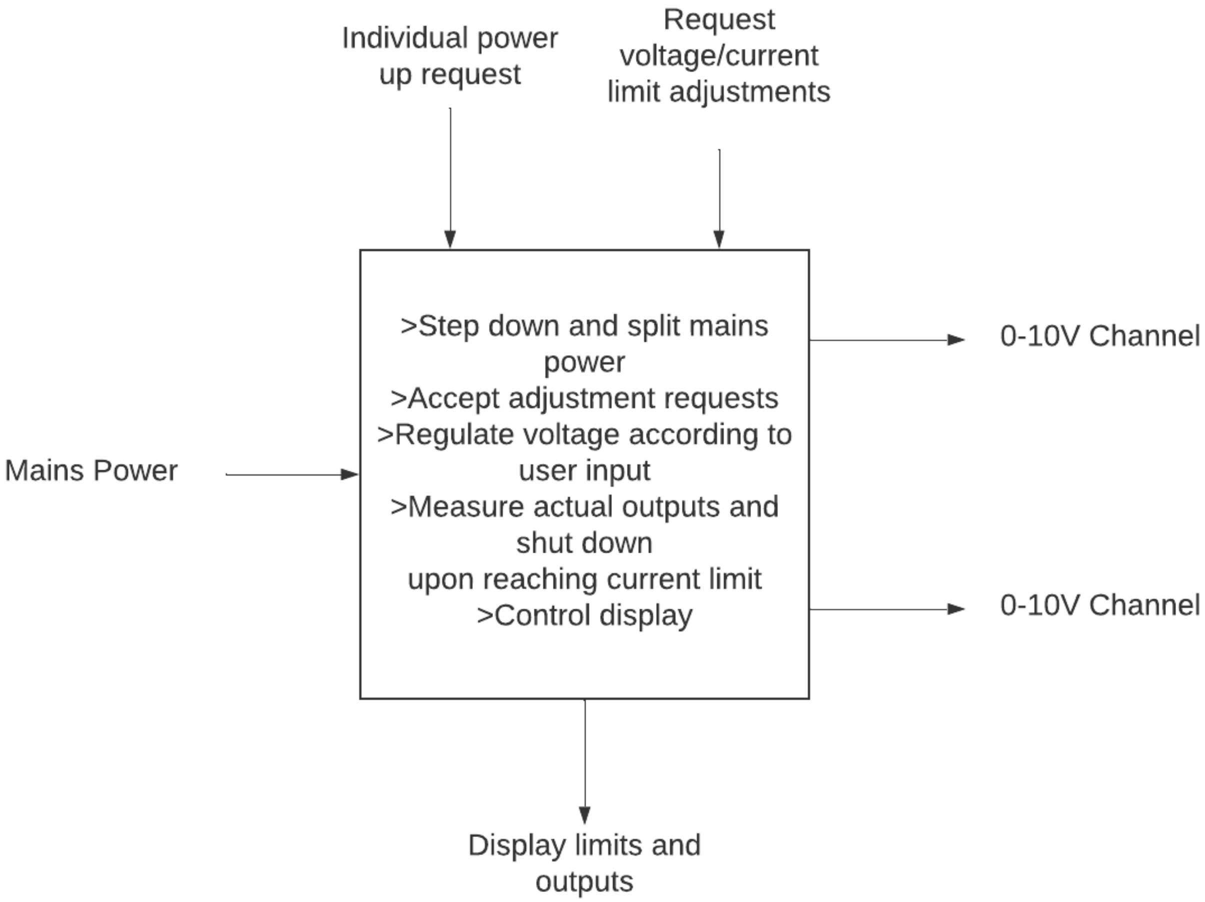
Technology and System-wide Requirements

- Custom enclosure will be designed to hold all circuitry
- All components will be set in custom PCB
- Materials cost should be in the range of $50-$100
- A 0.5A fast act fuse shall be used for user and circuit protection

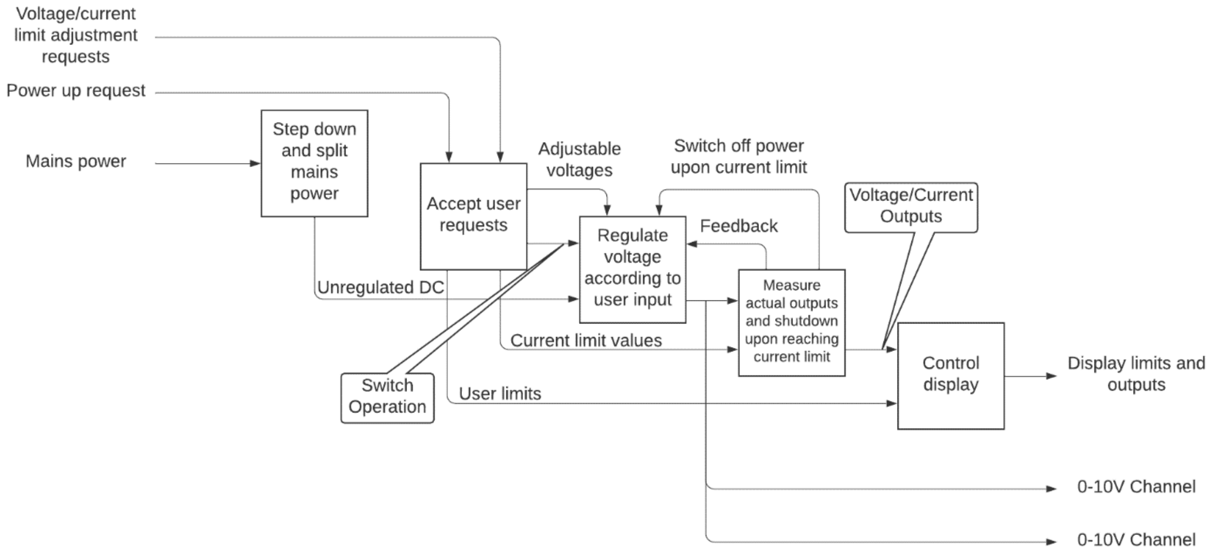
System Design

Functional Decomposition

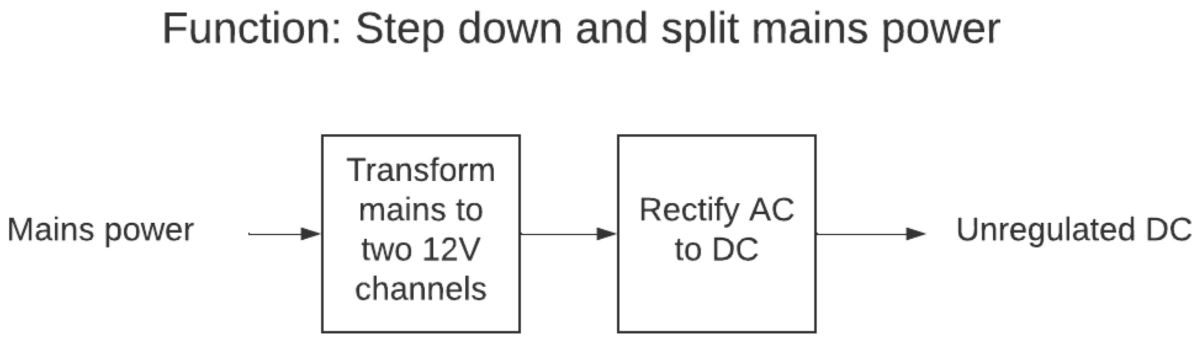
The following figure shows the inputs and outputs as well as the top-level functions that comprise our system. The inputs and outputs are simple, the system will take mains power and output two independent 0-10V channels controllable by the user and display the user set limits and actual outputs.



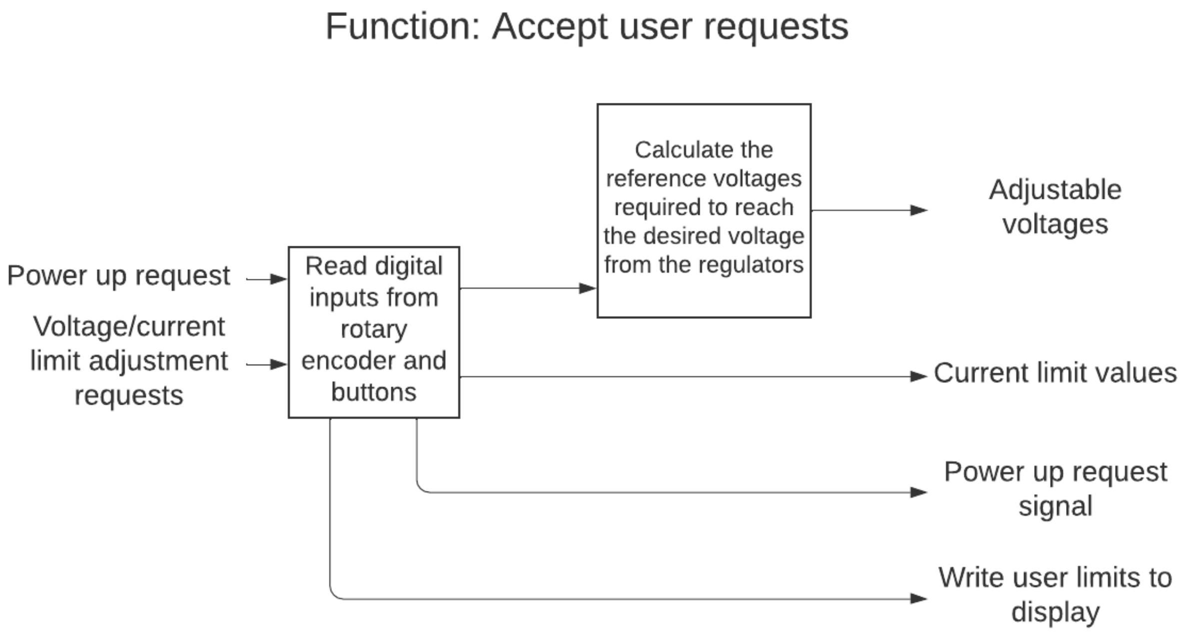
The following figure shows the first level of functional decomposition, each function is listed along with how each function interacts with other functions.



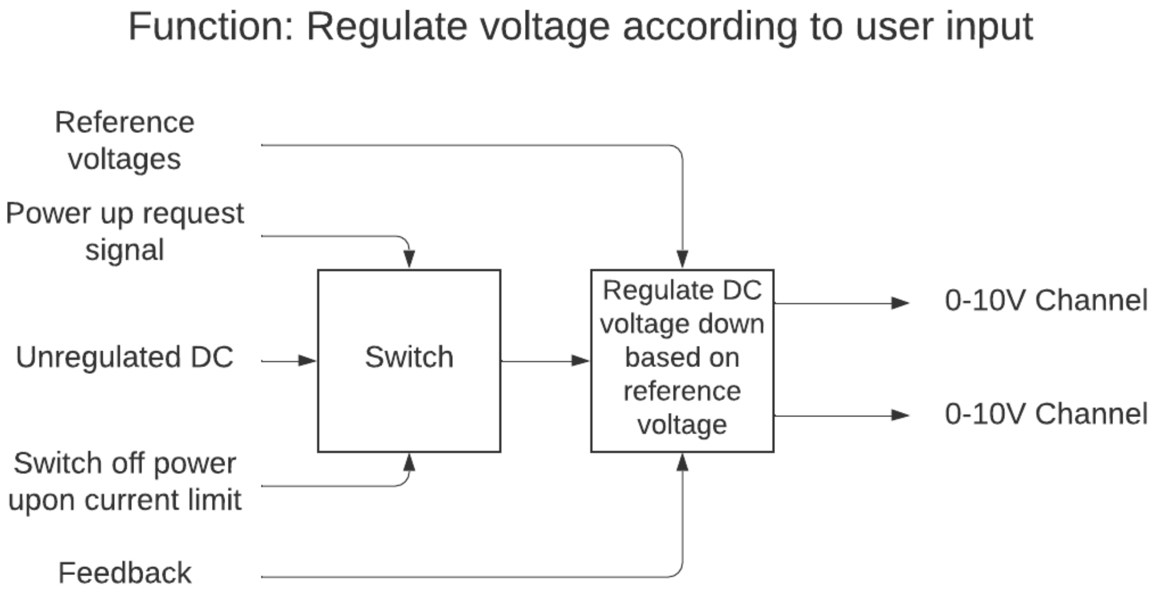
The following figures show the second level of functional decomposition for each of the functions listed in the previous figures. The first function is responsible for taking mains power and converting it into something manageable by stepping it down with a transformer, splitting the single lane input to two independent outputs, and rectifying it.



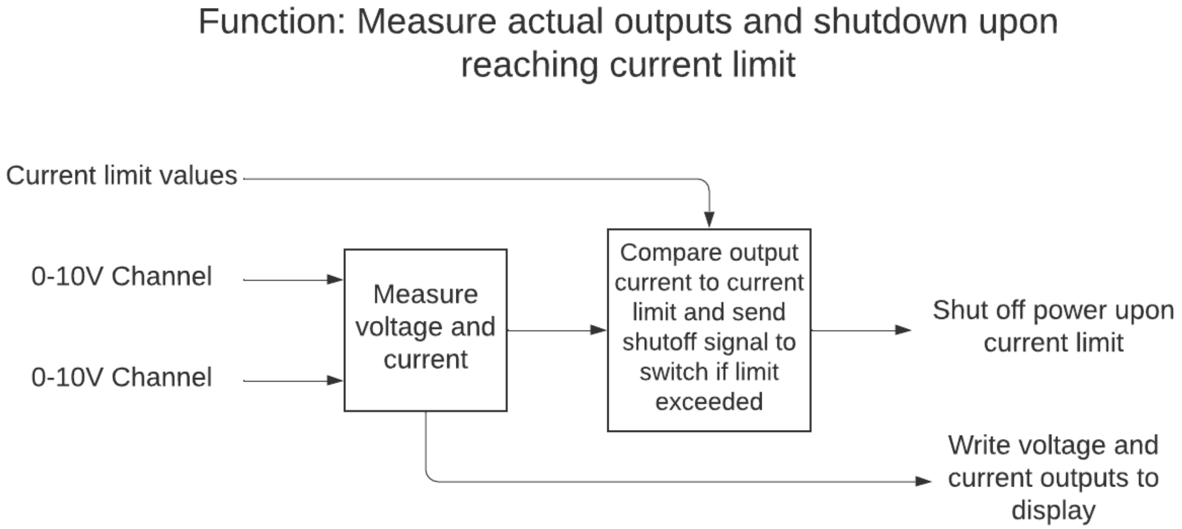
The second function is responsible for taking input from the user and interacting with the rest of the system to output a user desired voltage on each channel and display said user input.



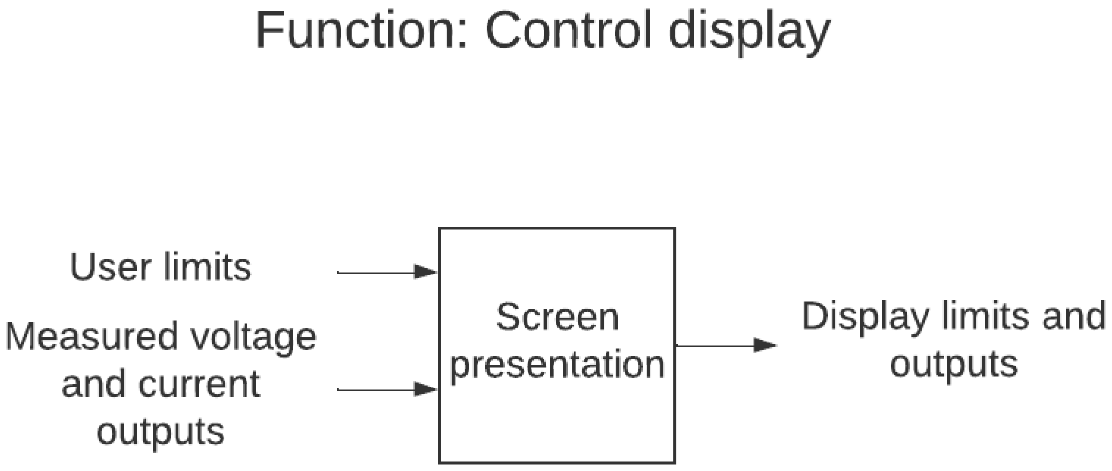
The third function is responsible for taking the unregulated but rectified AC voltage from the transformer and regulate it down to a user specified voltage.



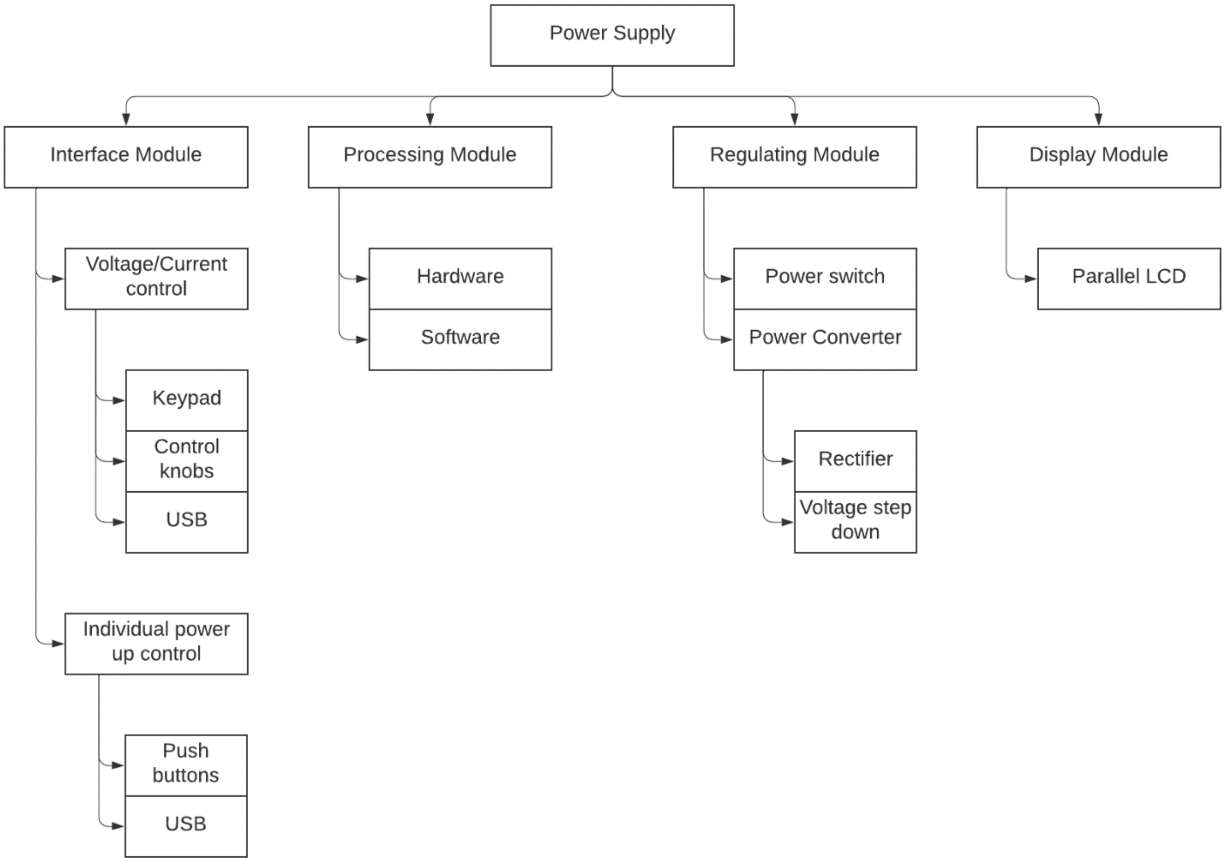
The fourth function is responsible for measuring the actual voltage and current outputs of each channel to display to the user and to shut off the switch if the output current is past the user defined limit. The measurements taken from an ADC would introduce an amount of lag that may be too great to control the regulator switch with so the function of shutting off power past a certain current will be done in hardware.

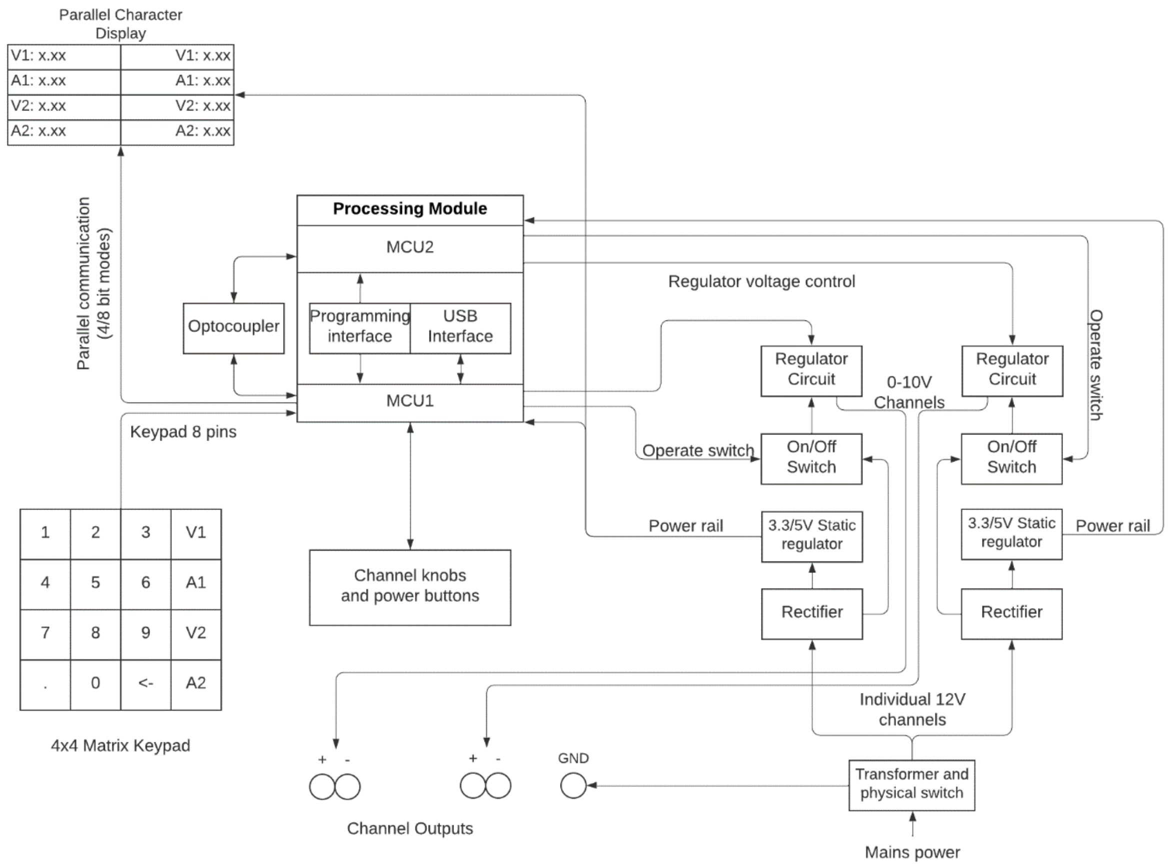


The fifth function is responsible for taking the user inputs and the measured outputs and displaying them on a simple LCD.



System Architecture


Power Supply
Interface Module
Processing Module
Regulating Module
Display Module
Voltage/Current control
Keypad
Control knobs
USB
Individual power up control
Push buttons
USB
Hardware
Software
Power switch
Power Converter
Rectifier
Voltage step down
Parallel LCD



For the two output voltage channels to be independent and retain the ability to chain them together for a single 0-20V channel there cannot be any electrical interference between them. For this reason, each channel's regulator circuit will be controlled by its own MCU and the MCUs will communicate with each other using an optocoupler with a full duplex communication protocol like SPI. Each MCU must be responsible for turning on or off its own channel, measuring its own channels' outputs, and adjusting the voltage output of its own channel. The responsibility of reading and displaying user inputs can be dedicated to one "main" MCU for simplicity and would allow for a lower cost lower functionality secondary MCU. The inputs from the rotary encoder and power button can be dedicated to each channel's MCU to reduce the amount of information that the main MCU needs to communicate with the secondary MCU, but the single keypad, single LCD, and USB input can only be operated by one MCU.

Preliminary Experimentation Plan

Preliminary experimentation will consist of testing and experimentation of individual components. This will be done for all components, from passive components such as resistors and diodes, and for active components like rotary encoders and the character LCD displays. Doing so will ensure correct operation of the components, finding defective components before they are implemented into the circuit. Also doing so will teach us how to use and control these components, making the implementation of these devices go more smoothly. Simultaneously, we will be modeling and simulating our circuit in PSPICE to ensure that all of components will be operating within their specified voltage and current range. This simulation will also be used to optimize our circuit for output accuracy, stability, and efficacy.

Once the individual components and PSPICE experimentation is complete, then an initial version of the channel circuit will be built into a breadboard for real-world testing. Experimenting with different components and layouts will allow us to iteratively improve the circuit design until it is ready to be transferred to perfboard for a semi-permanent solution as the front-plate is made and tested as well as creating the internal mounting within the enclosure. Once that is finished, a custom PCB will be created and built into with our final component choices for validation testing.

Additionally, we will perform experiments to determine that our Microcontrollers can perform in the way we want. This will be achieved by creating test scenarios where we interface with elements of our overall design such as the character display and the rotary encoders using the MCU and make sure we get the desired interactions. By running input and output tests first, we will have confidence that we can command our microcontrollers adequately before we attempt using them in tandem with the regulator circuit to control voltage/current output to each channel.

Experiments To Be Conducted:

- Create a functional regulator circuit simulation in PSPICE that outputs enough voltage to fuel the microcontrollers as well as the maximum possible output voltages for both channels (10V)
- Create a system using the rotary encoder, keypad, character display, and a microcontroller where values are entered using a combination of the rotary encoder and the keypad and are shown properly on the character display
- Create a rudimentary version of one of the output channel circuits on a breadboard that works properly on the defined voltage and current ranges (0-10V, 0-500mA)

Preliminary Project Plan

Short List of Tasks for ECE 493 (Based on System Architecture and Functional Decomposition)

Using the skills learned during early prototyping and modifications that came with it, The PCB design team will create each final PCB, then the components gathered for each module will be soldered onto their respective PCBs. Next, testing will be done on every module (interface, processing, regulating, and display), if the outputs correspond with the preliminary experimentation, then it will be placed into the PSU's enclosure. Finally, the front panel design will be inserted into the enclosure and all necessary connections to the front panel will be made.

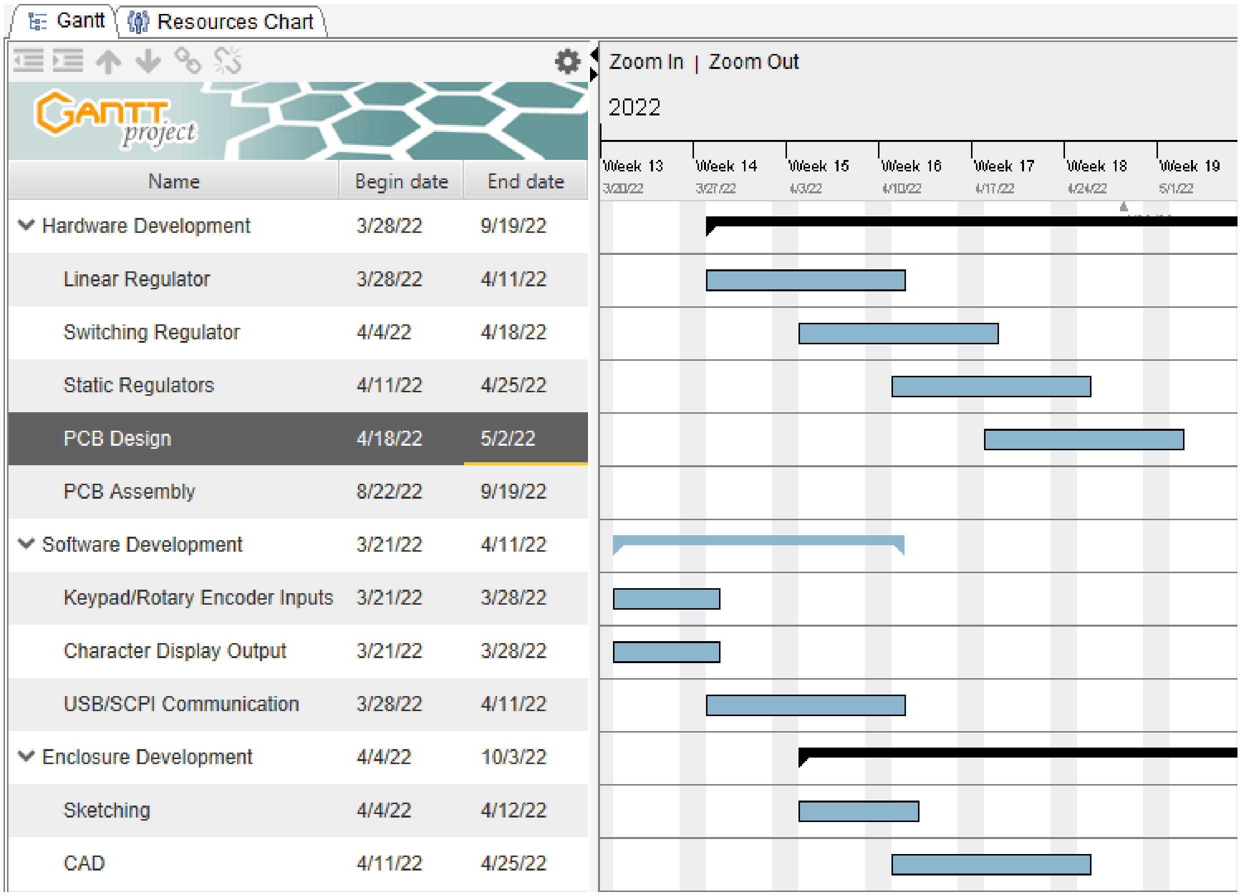
Afterwards, software will be written to implement our PSU onto the hardware, this includes software drivers, software modules, and system integration. Testing the PSU will be done by simple to more rigorous tests, e.g., comparing voltage output per channel with a multimeter, current draw, as well as the dual polarity functionality. Furthermore, LEDs, motors, etc. will be tested and monitored with the PSU as they draw power. Similarly, every front panel display element will be tested to ensure the interface is properly integrated into the system. To test the SCPI functionality, the command prompt window will be used to send SCPI commands to first confirm the SCPI library is working, then the input will be measured and/or observed to check correct output. Lastly, a python script will be used to create various tests allowed through SCPI which will then be measured/observed the same as the previous no script test.

Allocation of Responsibilities

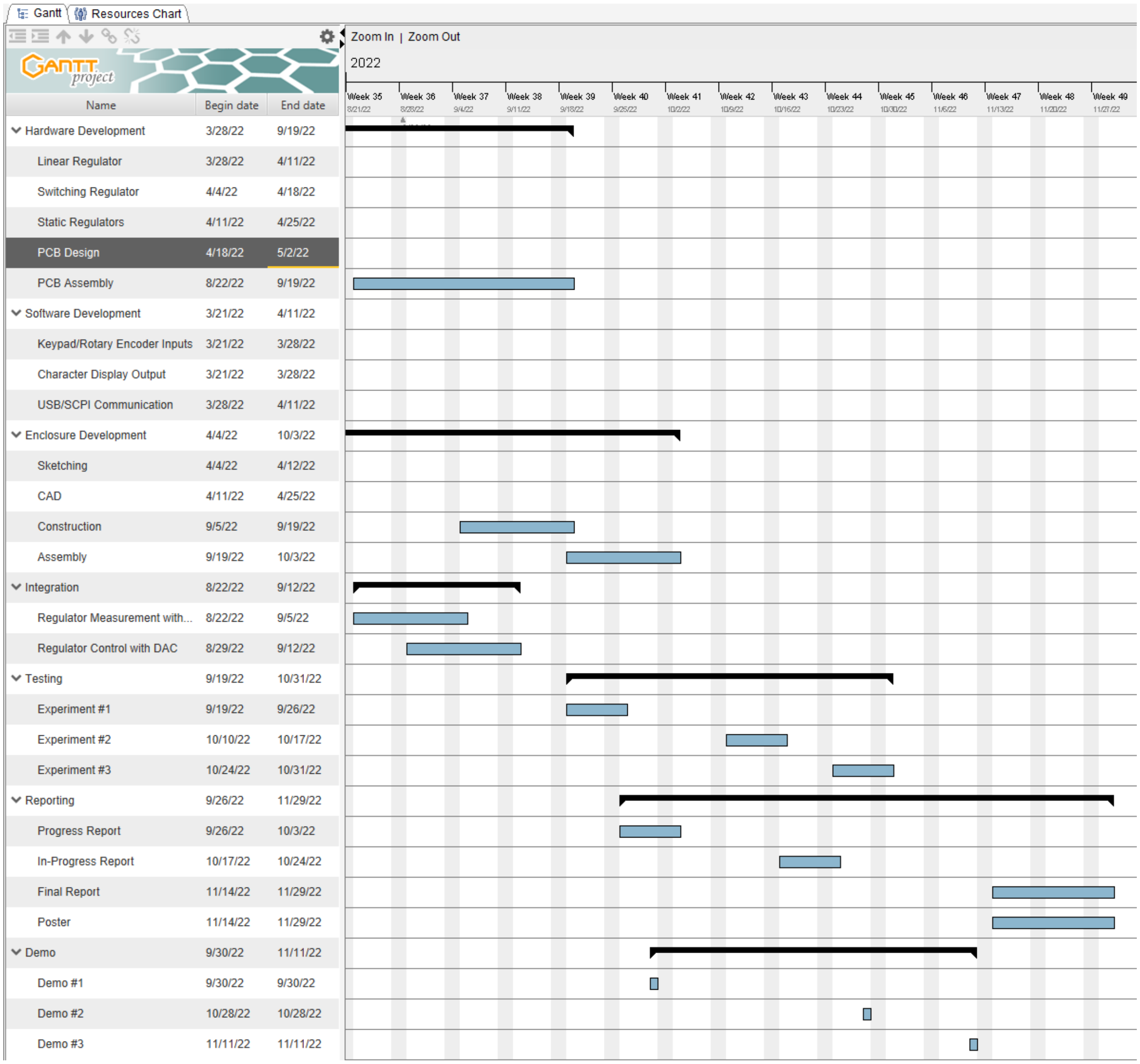
Front panel design will be handled by Tarun, PCB design and soldering by Tarun and Carlos, power electronics will be divided between all members, this involves module simulation and physical testing, of the switching regulator (Zach and Carlos), the linear regulator (Carlos and Dean), the 3.3v static regulator (Juan and Dean) and the 5v static regulator (Dean and Juan) portions. Furthermore, programming the MCU will be split between members, this encloses SCPI commands (Juan and Zach), switching algorithm (Zach and Juan), the LCD and LED markers interface (Tarun and Dean), and the rotary encoder and keypad inputs (Dean and Carlos). These roles are split to ensure a member is never without work. Therefore, every group member can contribute evenly to the group. As a result, roles also will determine who tests which components after finishing the prototype, as these members will have the most practice. Finally, anytime a member finishes a role, they will assist in whichever way they can with any ongoing work.

Positions									
Front Panel	**PCB & Soldering**	**Switching Regulator**	**Linear Regulator**	**3.3V Static Regulator**	**5V Static Regulator**	**SCPI Controls**	**Switching Algorithm**	**LED/LCD Interface**	**Rotary Encoder & Keypad Inputs**
Tarun	Carlos	Zach	Carlos	Juan	Dean	Juan	Zach	Tarun	Dean
Zach	Tarun	Carlos	Dean	Dean	Juan	Zach	Juan	Dean	Carlos

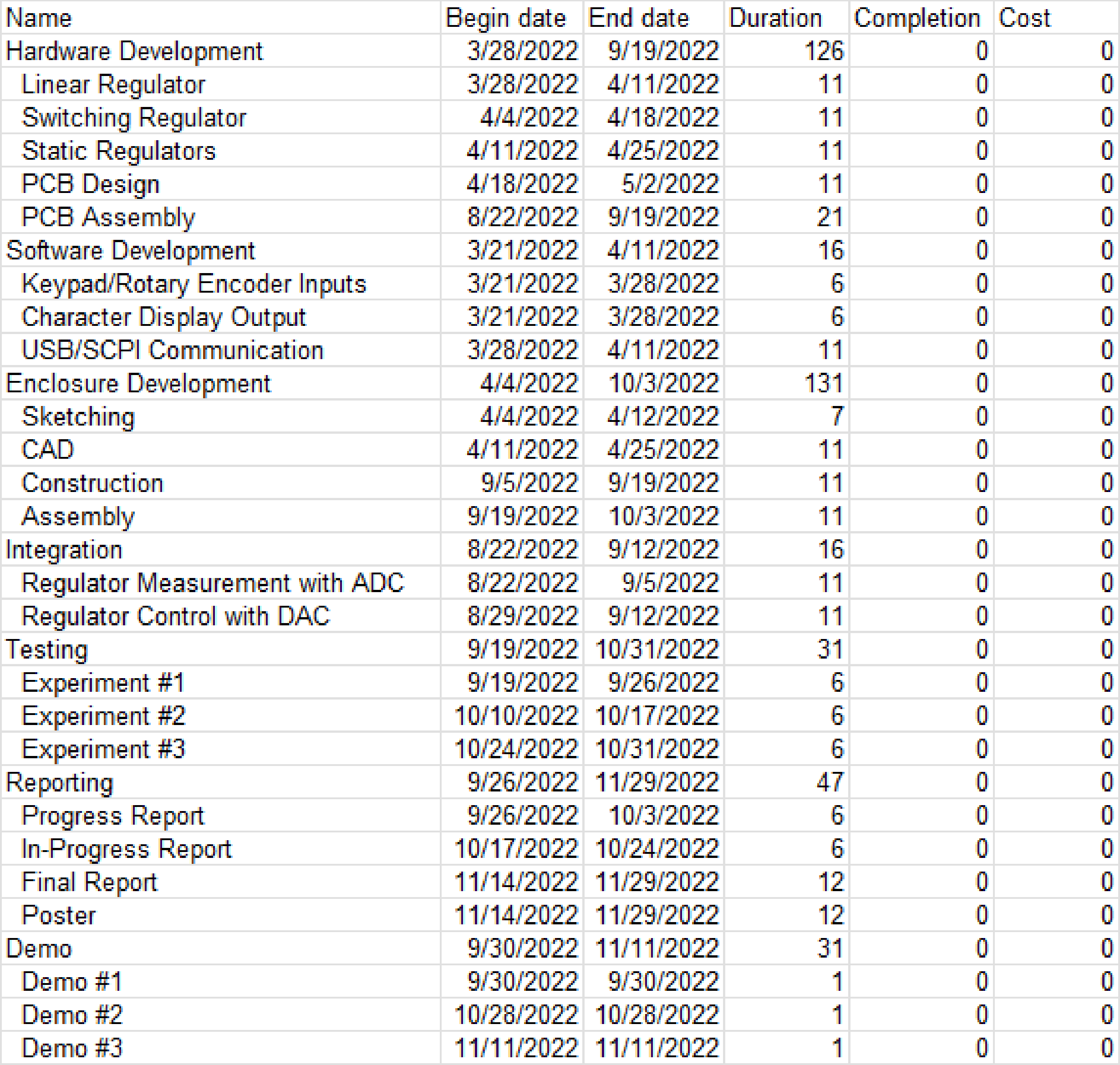
Tentative Project Schedule



Name	Begin date	End date
Hardware Development	3/28/22	9/19/22
Linear Regulator	3/28/22	4/11/22
Switching Regulator	4/4/22	4/18/22
Static Regulators	4/11/22	4/25/22
PCB Design	4/18/22	5/2/22
PCB Assembly	8/22/22	9/19/22
Software Development	3/21/22	4/11/22
Keypad/Rotary Encoder Inputs	3/21/22	3/28/22
Character Display Output	3/21/22	3/28/22
USB/SCPI Communication	3/28/22	4/11/22
Enclosure Development	4/4/22	10/3/22
Sketching	4/4/22	4/12/22
CAD	4/11/22	4/25/22



Name	Begin date	End date
Hardware Development	3/28/22	9/19/22
Linear Regulator	3/28/22	4/11/22
Switching Regulator	4/4/22	4/18/22
Static Regulators	4/11/22	4/25/22
PCB Design	4/18/22	5/2/22
PCB Assembly	8/22/22	9/19/22
Software Development	3/21/22	4/11/22
Keypad/Rotary Encoder Inputs	3/21/22	3/28/22
Character Display Output	3/21/22	3/28/22
USB/SCPI Communication	3/28/22	4/11/22
Enclosure Development	4/4/22	10/3/22
Sketching	4/4/22	4/12/22
CAD	4/11/22	4/25/22
Construction	9/5/22	9/19/22
Assembly	9/19/22	10/3/22
Integration	8/22/22	9/12/22
Regulator Measurement with...	8/22/22	9/5/22
Regulator Control with DAC	8/29/22	9/12/22
Testing	9/19/22	10/31/22
Experiment #1	9/19/22	9/26/22
Experiment #2	10/10/22	10/17/22
Experiment #3	10/24/22	10/31/22
Reporting	9/26/22	11/29/22
Progress Report	9/26/22	10/3/22
In-Progress Report	10/17/22	10/24/22
Final Report	11/14/22	11/29/22
Poster	11/14/22	11/29/22
Demo	9/30/22	11/11/22
Demo #1	9/30/22	9/30/22
Demo #2	10/28/22	10/28/22
Demo #3	11/11/22	11/11/22



Name	Begin date	End date	Duration	Completion	Cost
Hardware Development	3/28/2022	9/19/2022	126	0	0
Linear Regulator	3/28/2022	4/11/2022	11	0	0
Switching Regulator	4/4/2022	4/18/2022	11	0	0
Static Regulators	4/11/2022	4/25/2022	11	0	0
PCB Design	4/18/2022	5/2/2022	11	0	0
PCB Assembly	8/22/2022	9/19/2022	21	0	0
Software Development	3/21/2022	4/11/2022	16	0	0
Keypad/Rotary Encoder Inputs	3/21/2022	3/28/2022	6	0	0
Character Display Output	3/21/2022	3/28/2022	6	0	0
USB/SCPI Communication	3/28/2022	4/11/2022	11	0	0
Enclosure Development	4/4/2022	10/3/2022	131	0	0
Sketching	4/4/2022	4/12/2022	7	0	0
CAD	4/11/2022	4/25/2022	11	0	0
Construction	9/5/2022	9/19/2022	11	0	0
Assembly	9/19/2022	10/3/2022	11	0	0
Integration	8/22/2022	9/12/2022	16	0	0
Regulator Measurement with ADC	8/22/2022	9/5/2022	11	0	0
Regulator Control with DAC	8/29/2022	9/12/2022	11	0	0
Testing	9/19/2022	10/31/2022	31	0	0
Experiment #1	9/19/2022	9/26/2022	6	0	0
Experiment #2	10/10/2022	10/17/2022	6	0	0
Experiment #3	10/24/2022	10/31/2022	6	0	0
Reporting	9/26/2022	11/29/2022	47	0	0
Progress Report	9/26/2022	10/3/2022	6	0	0
In-Progress Report	10/17/2022	10/24/2022	6	0	0
Final Report	11/14/2022	11/29/2022	12	0	0
Poster	11/14/2022	11/29/2022	12	0	0
Demo	9/30/2022	11/11/2022	31	0	0
Demo #1	9/30/2022	9/30/2022	1	0	0
Demo #2	10/28/2022	10/28/2022	1	0	0
Demo #3	11/11/2022	11/11/2022	1	0	0

Potential Problems

Knowledge and Skills to be learned

- Power analysis
- Optimization of heat dissipation (Heat sink design)
- Power electronics for PSU
- PCB design
- 3D printing
- SCPI programming and usage
- Comparing input/output noise
- STM programming and debugging
- SPI, I2C, USB, USART protocols for PSU

Brief Project Risk Analysis

Working with Mains power brings risk— not only to our components—but also to our group members. To circumvent harm to the group, the transformer will be prepared alongside a fast-act 1.5A fuse, a power switch, and a container to hold all these components by our faculty advisor. To ensure components are not damaged it will be necessary to know and understand forward voltages, minimum load currents, etc. so components can run stable, noise be filtered, and temperatures kept within operating range. Before physical prototyping is attempted, the circuit will be simulated in PSpice, to further mitigate any chance of component damage.

References

Literature References

[1]

J. Voas, N. Kshetri, and J. F. DeFranco, “Scarcity and Global Insecurity: The Semiconductor Shortage,” *IT Professional*, vol. 23, no. 5, pp. 78–82, Sep. 2021, doi: 10.1109/MITP.2021.3105248.

[2]

“SPD3303X / SPD3303X-E Series Programmable DC Power Supplies,” *Siglent*. https://siglentna.com/power-supplies/spd3303x-spd3303x-e-series-programmable-dc-power-supply/ (accessed Mar. 03, 2022).

[3]

“SPS5000X Series wide range programmable Switching DC Power Supply,” *Siglent*. https://siglentna.com/power-supplies/sps5000x/ (accessed Mar. 03, 2022).

[4]

“Rigol DP821A Dual Output 140 W Programmable DC Power Supply | TEquipment.” https://www.tequipment.net/Rigol/DP821A/DC-Power-Supplies/Lab-Power-Supplies/ (accessed Mar. 03, 2022).

[5]

“Analog Discovery 2 - Digilent Reference.” https://digilent.com/reference/test-and-measurement/analog-discovery-2/start (accessed Mar. 03, 2022).

People Contacted

Dr. Jens-Peter Kaps (jkaps@gmu.edu)

14. Appendix B: Design Document

Problem Statement

The COVID-19 pandemic in 2020 caused a massive transition to online classes, including those in the electrical and computer engineering field. Furthermore, it heightened the already growing semiconductor shortage causing massive price increases in any chip related product [1]. Here at George Mason University (GMU), many of these classes contained a lab component, as a result, students were required to purchase the Digilent Analog Discovery 2 (AD2) (ADALM2000 being an alternative) which combines multiple lab instruments, therefore, satisfying the typical electrical engineering undergraduate curriculum — the only caveat being these labs would be handicapped. This is because the instrument could be used as a limited power supply source (5V maximum output), although this is just one of its many functions (oscilloscope, logic analyzer, spectrum analyzer, function generator and network analyzer). Unfortunately, as the semiconductor shortage increased prices, both these instruments and lab bench power supplies (PSU) have greatly risen in cost (e.g., the AD2 price grew to $400 [2]). Tables 1 and 2 Compare Commercial DC lab power supplies with 2 channels or more, with the lowest cost being greater than $200 (KA3305P at $235) [8]. Programmable power supplies start at the same price, as shown by the KA3305P. Their capabilities, voltage range and current range, are much above what is required for the ECE undergraduate labs and cover more than just low-power applications. Therefore, our group is proposing a low-cost programmable dual channel power supply that can be used by students and hobbyists, as a standalone or complementary instrument for low-power applications.

Dual Channel DC Power Supplies			
Model, Programmability	Cost (USD)	Vmax	Vmin
E36233A, programmable [3]	3059	30	20
Series 2220-30-1. Programmable [4]	1320	30	1.5
SPS5044X, programmable [5]	2269	40	30
DP821A, programmable [6]	731	60	1
GPD-2303S, programmable [7]	545	30	3
GPE-2323, non-programmable [7]	306	32	3

Table 1: Dual Channel Programmable DC Power Supplies Comparison

Triple Channel DC Power Supplies			
Model, Programmability	Cost (USD)	Vmax	Vmin
EDU36311A, programmable [3]	898	30	1
Series 2230-30-1, programmable [4]	1480	30	5
SPD3303X, programmable [5]	549	32	3.2
SPD3303X-E, programmable [5]	399	32	3.2
SPD3303C, programmable [5]	299	32	3.2
DP831, programmable [6]	473	30	2
SPD-3606, non-programmable [7]	759	30	6
KD3305P, programmable [8]	250	30	5
KA3305P, programmable [8]	235	30	5

Table 2: Triple Channel Programmable DC Power Supplies Comparison

Market/Application Review

PSU offering SCPI capability, two-channels, and millivolt/milliamp resolution are commonly found at upwards of $200, e.g., the Siglent SPS5044X, GW Instek GPD-2303S, and Rigol DP821A are priced currently at $2,269, $545, $731 [5, 7, 6], respectively. It must be mentioned that each device's respective channel output, wattage, etc. are different. Comparing the lowest price (GPD-2303S, $545) to our proposed budget (50-$100), it becomes reasonable to consider the proposed solution, which maintains similar features, with greater accuracy although at less power. Additionally, triple channel PSU with programmability and 10mV/10mA accuracy or greater are represented by the KA3305P, KD3305P, SPD3303, SPD3303X-E at $235, $250, $299, $399 respectively [8, 5]. Comparing these shows our proposal remains more accurate and less costly than all three products, although offering less power and channels. It must be mentioned that here at George Mason University (GMU), most undergraduate Electrical Engineering labs will not exceed 5 watts of power, this is because they were recently handicapped with the AD2's power supply in mind, i.e., 0.5 to 5V, -0.5 to -5V output voltage, 2.1W maximum power, and 700mA maximum current per channel [2]. With that in mind, our PSU will be a budget friendly complementary or standalone instrument for students and hobbyists, with the only constraint limiting it to low power applications.

Requirements Specifications

Mission Requirements

- The device will operate two independent controllable voltage channels

Operational Requirements

Input/Output Requirements

- The output voltage and current limit per channel shall be adjustable using a keypad and a knob (rotary encoder)
- Voltage, current, and current limit input/output per channel shall be displayed on a simple LCD
- Output voltage shall be adjustable between 0-10V
- Power supply will have the ability to set current limits per channel.
- Power supply will be controllable through USB using SCPI
- Power supply will offer two channels

External Interface Requirements

- Power supply shall receive power from Mains power

Functional Requirements

- Maximum Current per Channel shall be 500mA
- If the current limit is reached, the output will switch off
- Digital readouts for voltage, current, and current limit per channel as well as an indication of if the current limit has been reached shall be shown

Technology and System-wide Requirements

- Custom enclosure will be designed to hold all circuitry
- Custom front panel will be designed with clear, readable labels
- All components will be set in custom PCB
- Materials cost should be in the range of $50-$100
- A 1A fast act fuse shall be used for user and circuit protection

Current Part List

We began our part selection with the knowledge that we would need to be controlling at least one linear regulator with a possible addition of another linear or switching regulator and that the overall system would need to be controllable through USB. We would need an ADC on the MCU to measure voltage and a DAC or PWM module would be preferable to control regulator ICs. Our search for MCUs began with the Arduino uno because of the simple dip package but we quickly realized that it was quite underpowered and did not have the peripherals we were looking for. We then began to look at MSP MCUs from TI and came across a development board that had USB, an ADC, and a DAC; however, the MCU chips themselves were out of stock so we decided to keep looking. We then began looking into STM32 development boards and we came across the STM32L152RC-Disco board which has USB 2.0, a 12-bit ADC, and a 12-bit DAC with a fast clock and plenty of ram for our usage. The specific chip on this board was also out of stock but the STM32L152RB was still in stock and was the same chip with slightly less ram. We then began to look for a slightly less powerful MCU that was still in the L1 family and USB was not a requirement. The only other chip we could find in stock was the STM32L100C6 which still had USB but was overall less powerful and was a bit cheaper. At this point even these chips we selected are now out of stock, but we purchased a suitable amount of them and do not expect to need to select new chips.

The transformer was picked out for us by our faculty supervisor, Dr. Kaps, but we originally considered using a computer power supply for our main power in. This was quickly discarded in favor of a split transformer since it was far cheaper and guaranteed no electrical interference. To rectify the sine wave from the transformer we originally wanted to use Schottky diodes because of their low voltage drop but eventually decided to use a regular bridge rectifier IC since it was cheaper and should do the job assuming we have enough voltage at the source when running at full load.

For the linear regulator we were looking for an IC that could be easily controlled by a DAC and we came across the LT3080ET which was also shown to us by our faculty advisor, Dr. Kaps. This chip has given us some problems with its minimum load requirements, making it not as easy to use as originally thought, but it still fits our requirements. We chose the LT1991 IC op-amp since it was recommended by LT as a voltage multiplier to control another chip with a DAC. We may need to switch to the LT3083ET which is slightly more expensive but allows a higher current, specifically a higher current into the SET pin which we believe to be the cause of us breaking two LT3080s to date.

We chose the L6902 as our switching regulator since it was relatively cheap and offered the lowest voltage drop amongst the switching regulators we were looking at. This chip was not as easy to control with a DAC since it could not be directly driven, and we had to do a little bit of math described below to interface the DAC with the feedback resistor divider node.

The current part list only contains the major components of the power supply, and we are currently leaving out various capacitors, resistors, inductors, as well as LEDs and buttons since they are quite cheap, and we do not have a definitive amount needed yet. The price of the enclosure may not be accurate since Jameco is currently down and we may wish to procure a larger enclosure.

Part	# of parts	Individual Price	Price Per 1000	Total per 1000	Total Price:	67.784
STM32L152RBT6A	1	7.66	5.74	5.74		
STM32L100C6U6A	1	5.39	4.84	4.84		
FS24-800-C2	1	16.5	9.82	9.82		
RS207M	2	0.45	0.199	0.398		
LT3080ET	2	6.46	3.79	7.58		
LM2672	2	5.6	3.06	6.12		
L6902	2	3.44	2.92	5.84		
LT1991CMS	2	4.27	2.27	4.54		
6N137	4	0.8	0.269	1.076		
EA DOGS164N-A	1	16.87	9.15	9.15		
EC11 Rotary Encoder	2	3.63	2.34	4.68		
Enclosure	1	8	8	8		

Table 3: Major part list

1. *Microcontrollers*

The brains of the power supply unit we are developing are the microcontrollers, which are responsible for handling user inputs, processing them, and delivering the output voltage/currents based on said inputs. Microcontrollers are computers on integrated circuits designed for use in embedded systems, which are systems that perform a specific function.

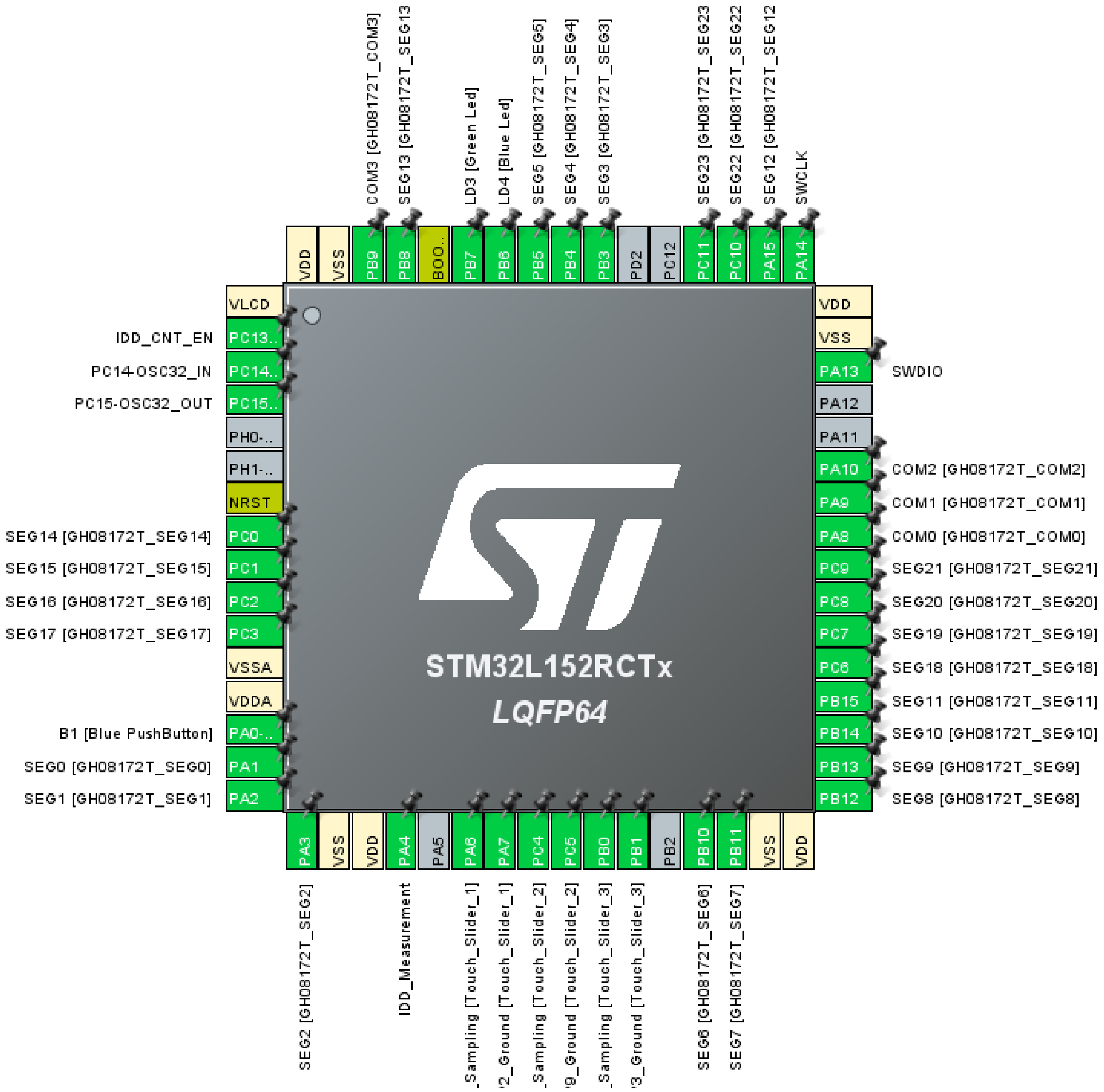


Figure 1: Default MCU configuration

2. *Transformers*

To transform Mains voltage into a voltage that can be safely used by our system, a step-down transformer will be used, which is a component used to convert high voltage at a low current into low voltage at a high current, where the secondary voltage is directly proportional to the turns on the secondary coil and the primary voltage, all divided by the number of turns on the primary coil ($V_S = (V_P * N_S) / N_P$).

$$Vs = \frac{Vp * Ns}{Np}$$

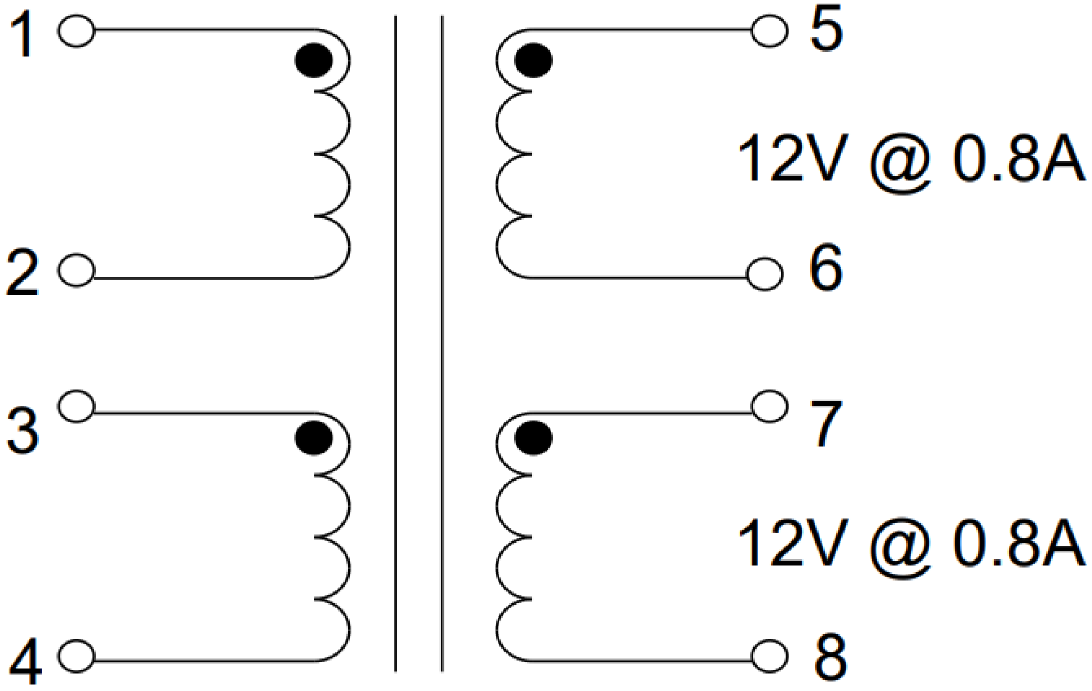


Figure 2: Dual secondary transformer

3. *AC-DC Rectification*

A full-wave rectifier, which is a circuit that converts alternating current voltage into direct current voltage, will be used to convert the stepped-down voltage into DC voltage. The full-wave rectifier has a voltage drop equivalent to $2V_D$, where V_D is the voltage drop across one of the diodes in the rectifier. There is an additional drop that we need to account for when flattening a sinusoidal wave, the average output of a bridge rectifier is $2Vp/\pi$ and we get a peak voltage of 12/0.7071=16.97072V with an RMS voltage of 12. Plugging this peak voltage back into the first equation we get an average voltage output of 10.80389V. This assumes a full load however and we will be limiting our current draw to half an amp, or about 60% load, so we will likely get a higher dc voltage out.

$$Vavg = \frac{2Vp}{\pi}$$

$$Vrms = \frac{Vp}{\sqrt{2}}$$

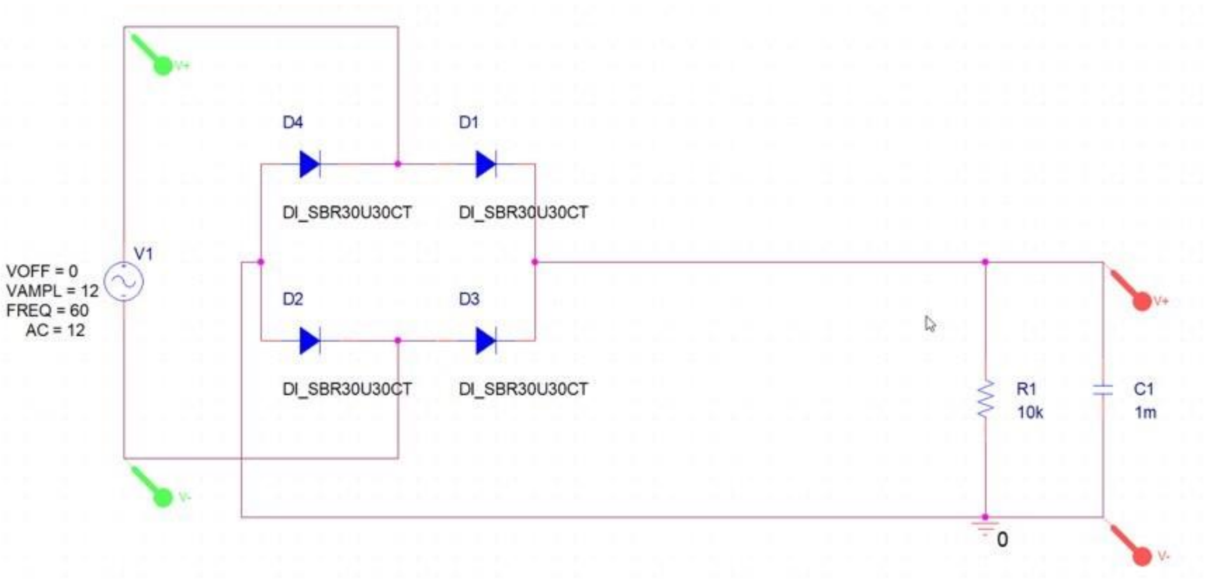


Figure 3: Full wave bridge rectifier

4. *Linear Regulators*

The regulator circuit that will output the voltage sourced by the rest of our system will be composed of three types of regulators: linear, switching, and static. Linear regulators are regulators that utilize a linear component, such as a potentiometer, to regulate input voltage. These types of regulators are relatively simple to design and do not output much noise, but they have a low power capacity and are not as efficient as other regulators.

$$Vout = Vset \pm 0.3V$$

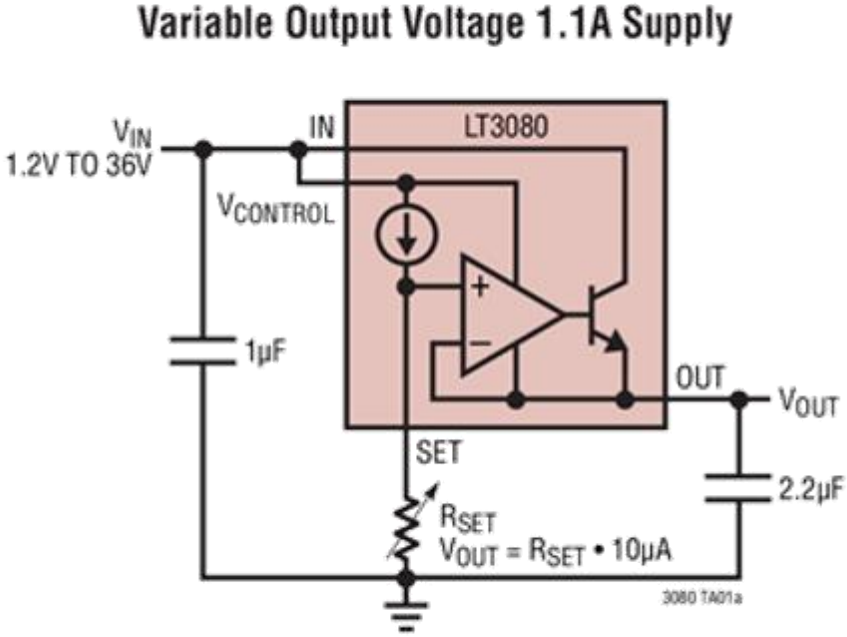


Figure 4: LT3080 Linear regulator

5. *Switching Regulators*

Switching regulators use switching components, such as MOSFETs, and work by taking small chunks of energy from the input voltage source and transferring them to the output. These kinds of regulators are more efficient and can handle high amounts of power but are also more complicated to design and have relatively noisy output signals. In order to control the output of this switching regulator we need to introduce a third resistor R3 to the node between the first two connected to the feedback pin and drive this new resistor with a voltage from a DAC. Applying KVL at the feedback pin of a switching regulator gives us the following equation:

$$Vfb = \frac{R3R2Vout \ + \ R1R2Vdac}{R3R2 \ + \ R1R3 \ + \ R1R2}$$

This equation can then be used to make a system of equations based on the minimum and maximum output of the regulator as well as the DAC to find the required resistance of any two resistors given any choice of a third resistor.

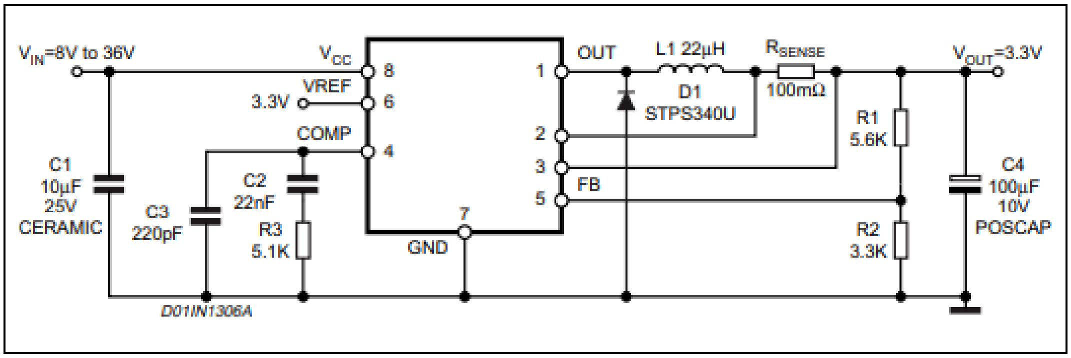


Figure 5: L6902 Switching regulator

Static regulators output a constant voltage value that can’t be changed, which makes them ideal for powering a device that needs a specific constant voltage, such as a microcontroller, but not for powering something that requires a variety of voltage values, such as the output channels for our PSU. Switching regulators have more ripple than linear regulators and this may be an issue for certain components like microcontrollers unless filtered properly.

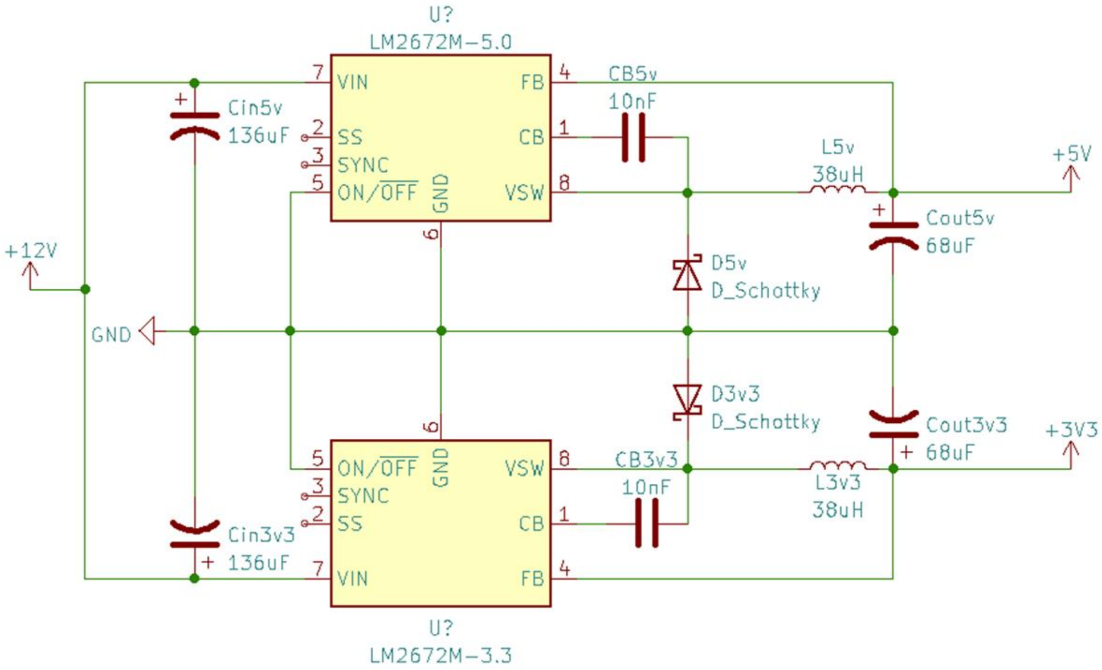


Figure 6: LM2672 Switching regulator

System Design

I. ***Functional Decomposition***

The following figure shows the inputs and outputs as well as the top-level functions that comprise our system. The inputs and outputs are simple, the system will take mains power and output two independent 0-10V channels controllable by the user and display the user set limits and actual outputs.

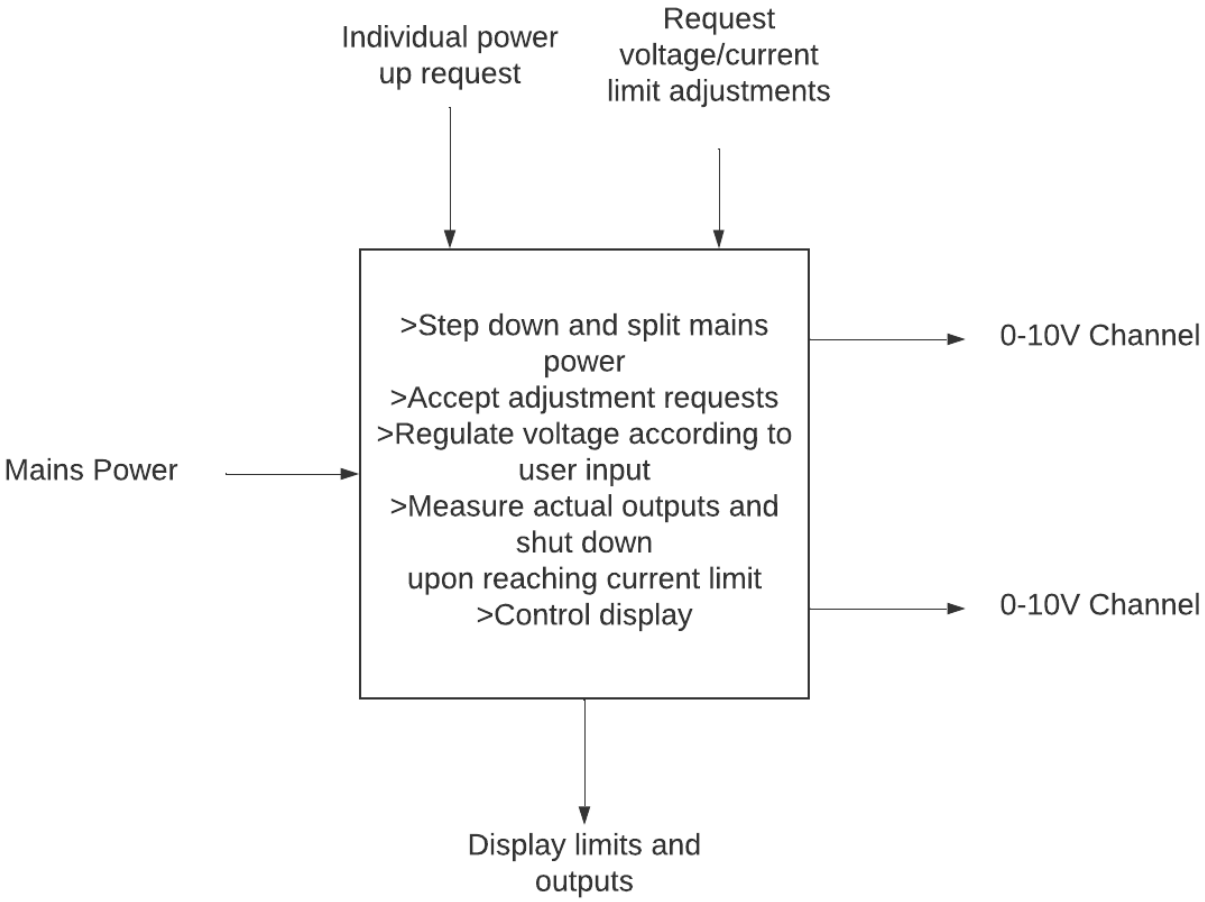


Figure 7: Level 0 decomposition

The following figure shows the first level of functional decomposition, each function is listed along with how each function interacts with other functions.

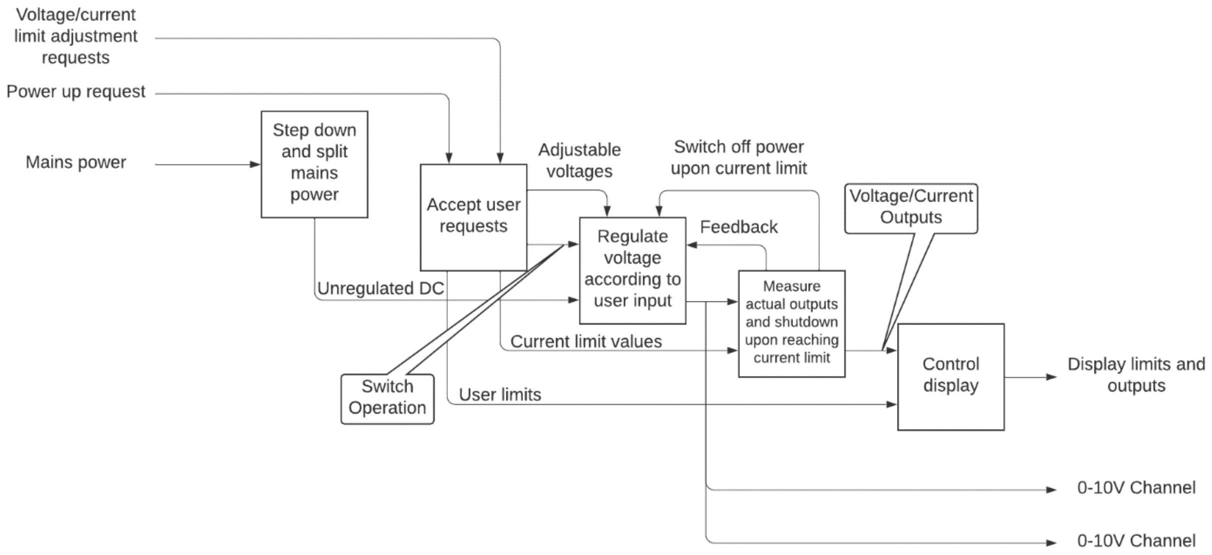


Figure 8: Level 1 decomposition

The following figures show the second level of functional decomposition for each of the functions listed in the previous figures. The first function is responsible for taking mains power and converting it into something manageable by stepping it down with a transformer, splitting the single lane input to two independent outputs, and rectifying it.

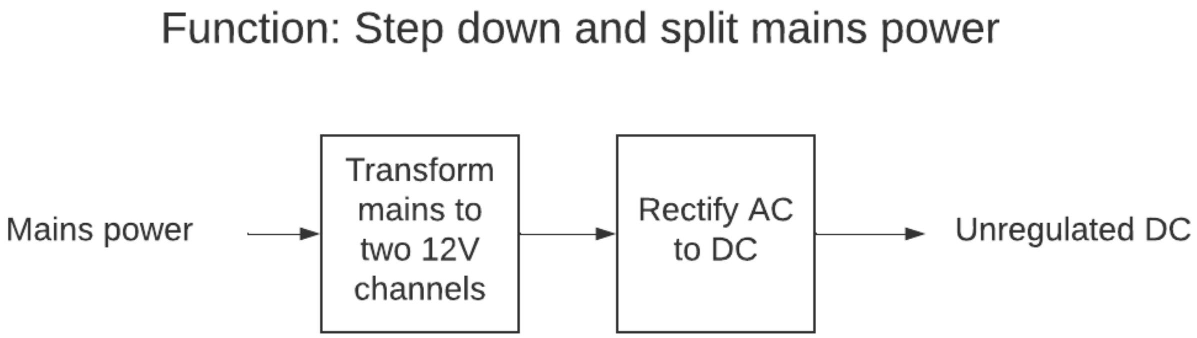


Figure 9: Level 2 decomposition level 1

The second function is responsible for taking input from the user and interacting with the rest of the system to output a user desired voltage on each channel and display said user input.

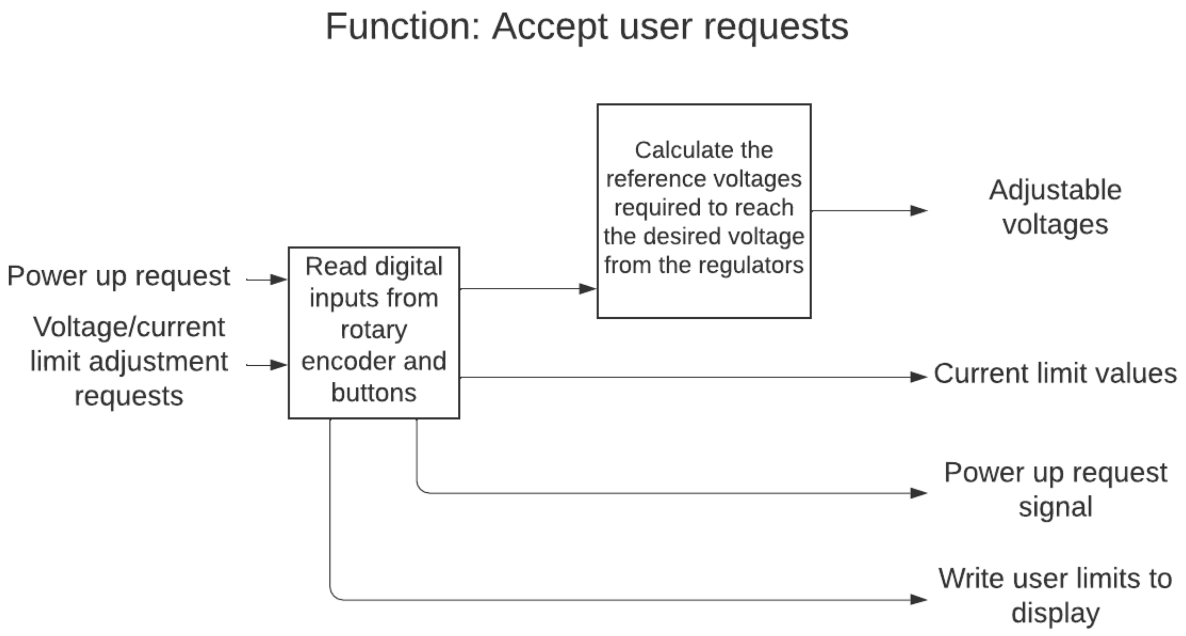


Figure 10: Level 2 decomposition level 2

The third function is responsible for taking the unregulated but rectified AC voltage from the transformer and regulate it down to a user specified voltage.

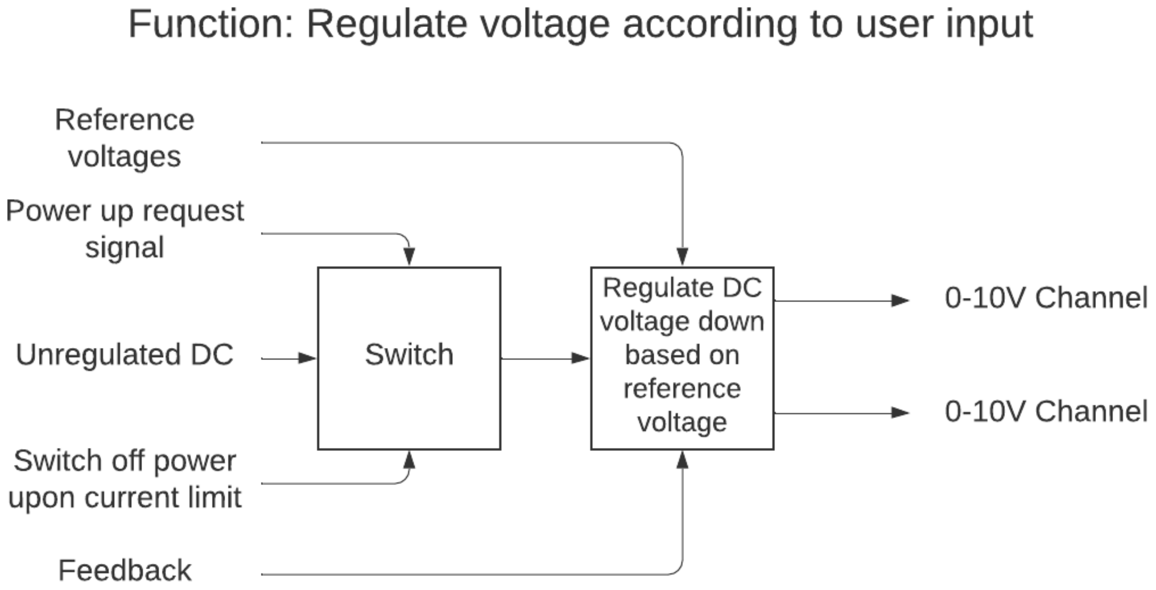


Figure 11: Level 2 decomposition level 3

The fourth function is responsible for measuring the actual voltage and current outputs of each channel to display to the user and to shut off the switch if the output current is past the user defined limit. The measurements taken from an ADC would introduce an amount of lag that may be too great to control the regulator switch with so the function of shutting off power past a certain current will be done in hardware.

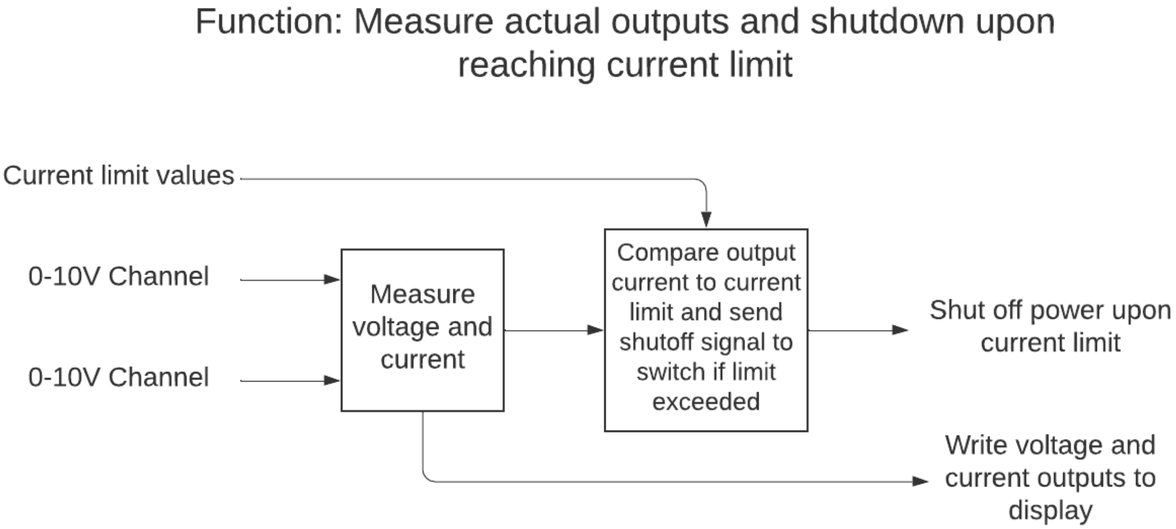


Figure 12: Level 2 decomposition level 4

The fifth function is responsible for taking the user inputs and the measured outputs and displaying them on a simple LCD.

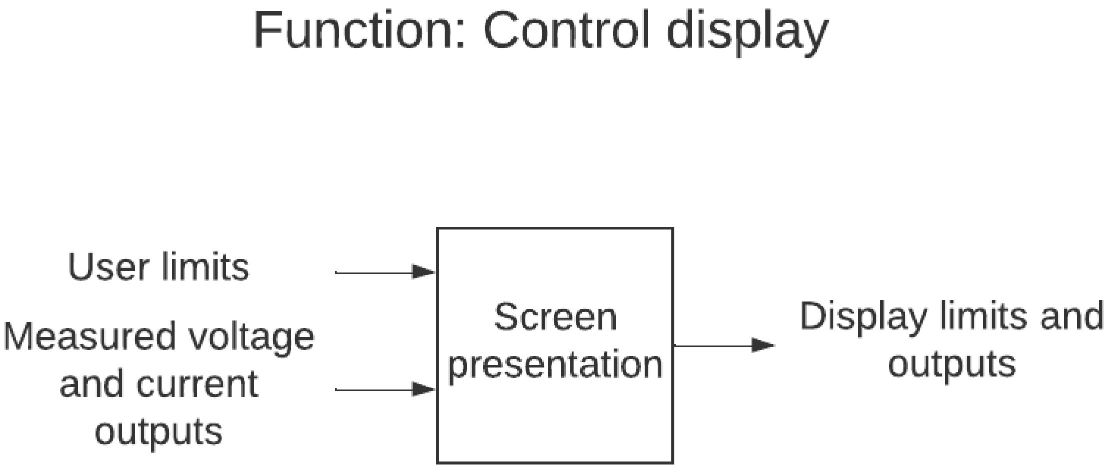


Figure 13: Level 2 decomposition level 5

II. ***System Architecture***

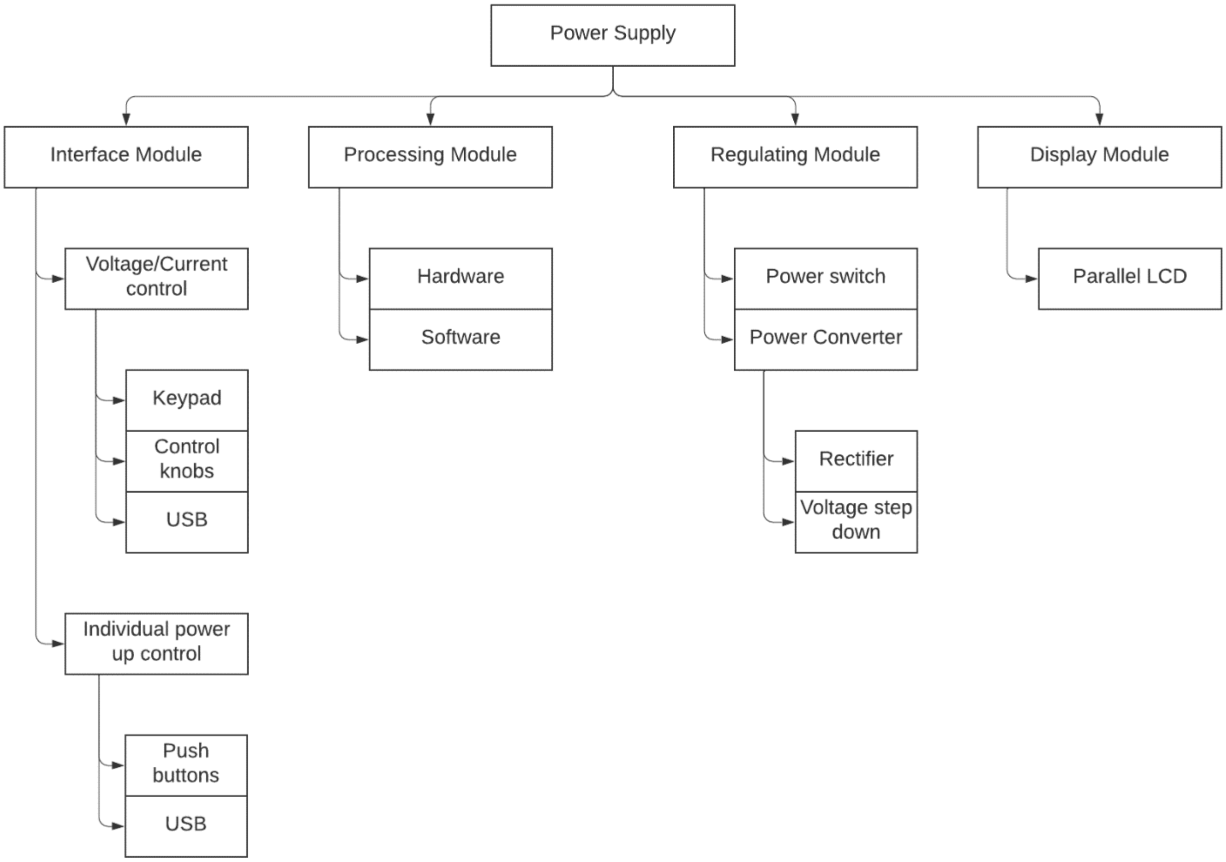


Figure 14: Physical architecture

The following figure shows the various top-level components of our system and how they are connected. For the two output voltage channels to be independent and retain the ability to chain them together for a single 0-20V channel there cannot be any electrical interference between them. For this reason, each channel’s regulator circuit will be controlled by its own MCU and the MCUs will communicate with each other using an optocoupler with a full duplex communication protocol like SPI. Each MCU must be responsible for turning on or off its own channel, measuring its own channels' outputs, and adjusting the voltage output of its own channel. The responsibility of reading and displaying user inputs can be dedicated to one “main” MCU for simplicity and would allow for a lower cost lower functionality secondary MCU. The inputs from the rotary encoder and power button can be dedicated to each channel’s MCU to reduce the amount of information that the main MCU needs to communicate with the secondary MCU, but the single keypad, single LCD, and USB input can only be operated by one MCU.

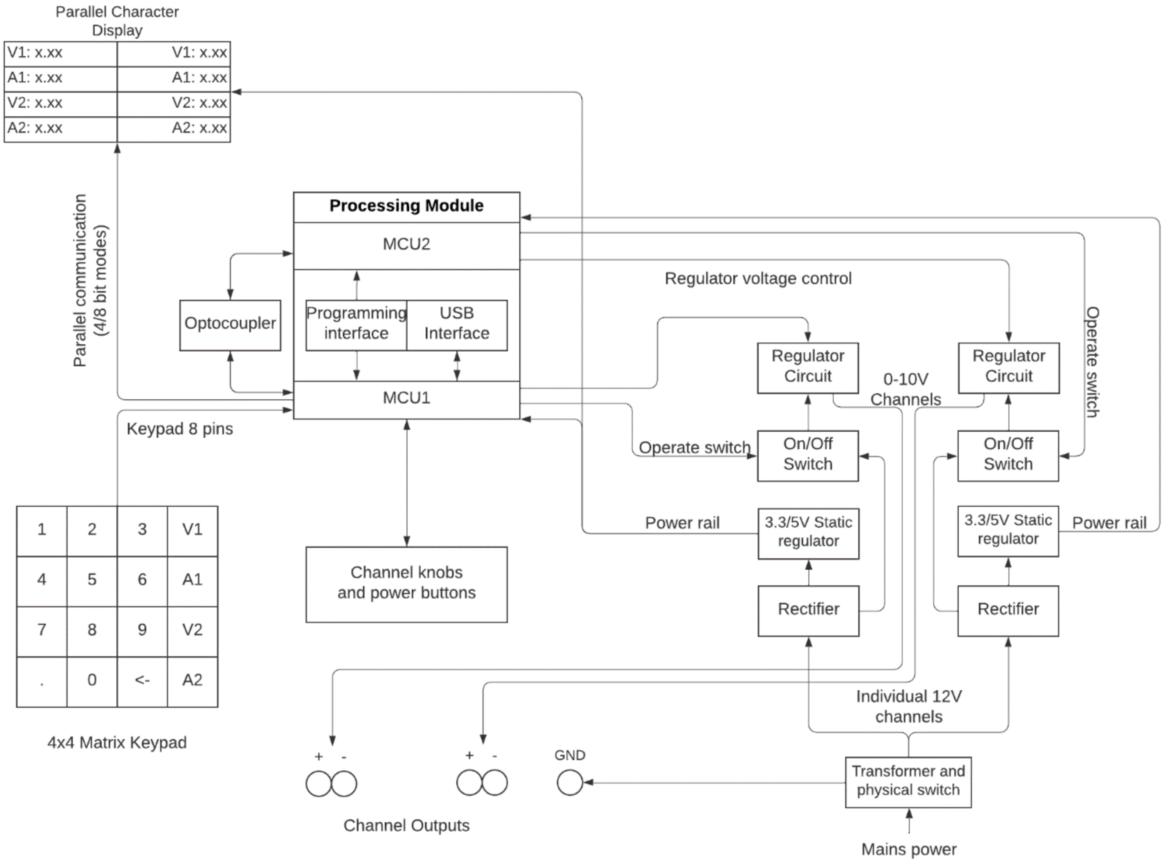


Figure 15: System architecture

Detailed Design

The following figure is a state machine depicting each possible state of our power supply.

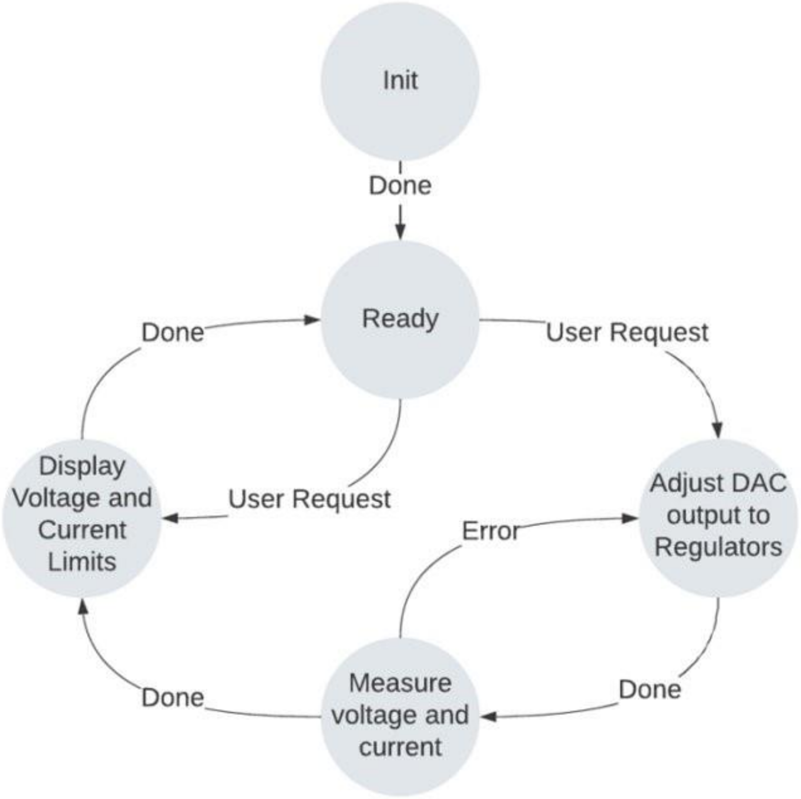


Figure 16: State machine

The following figure depicts the flow of data in the lab power supply, cataloguing inputs from the use as well as internal data flow between components such as the LCD and the MCUs.

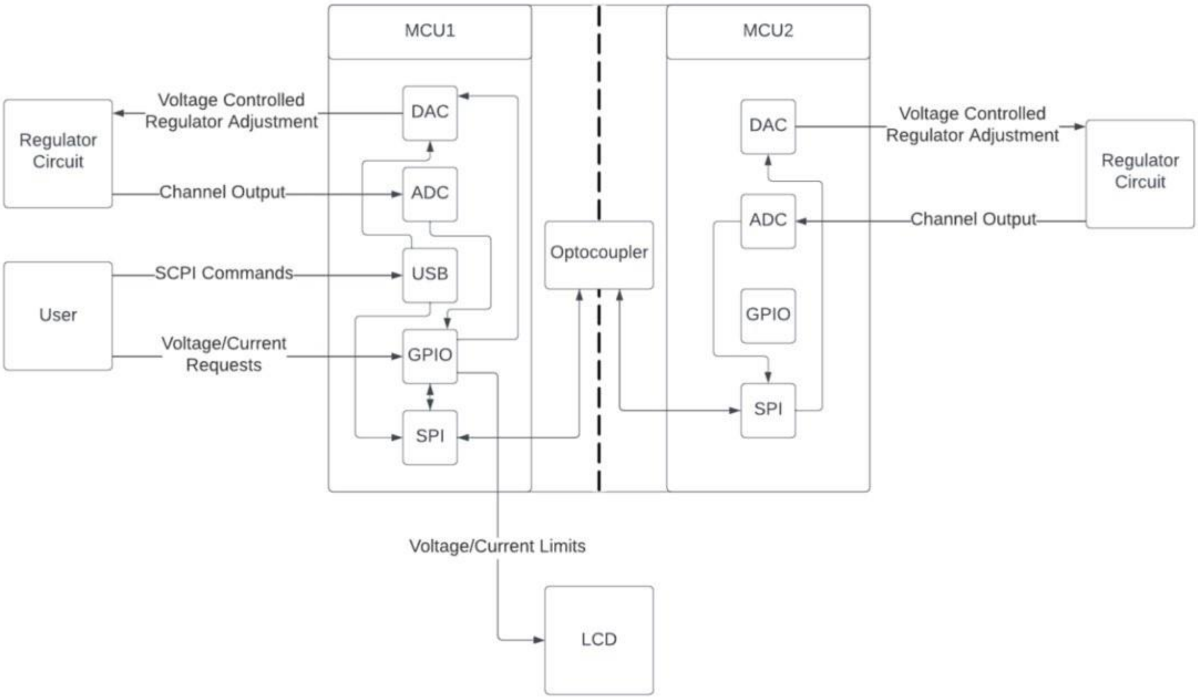


Figure 17: Data flow diagram

The following figure is a flowchart depicting the operations carried out by the MCU when the PSU is turned on.

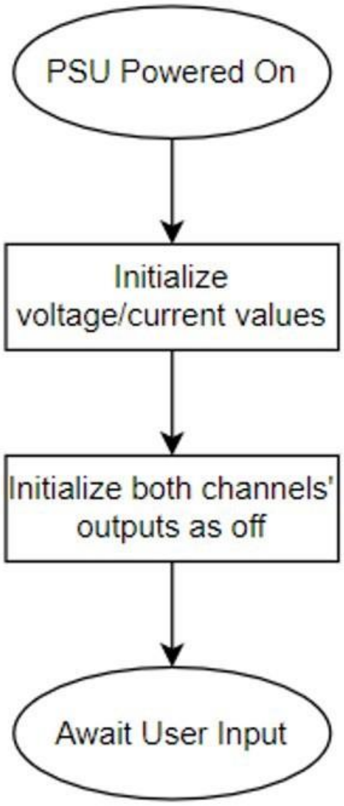


Figure 18: Init flow chart

The following figure is a flowchart depicting the operations that occur when the user inputs voltage/current values using the rotary encoder.

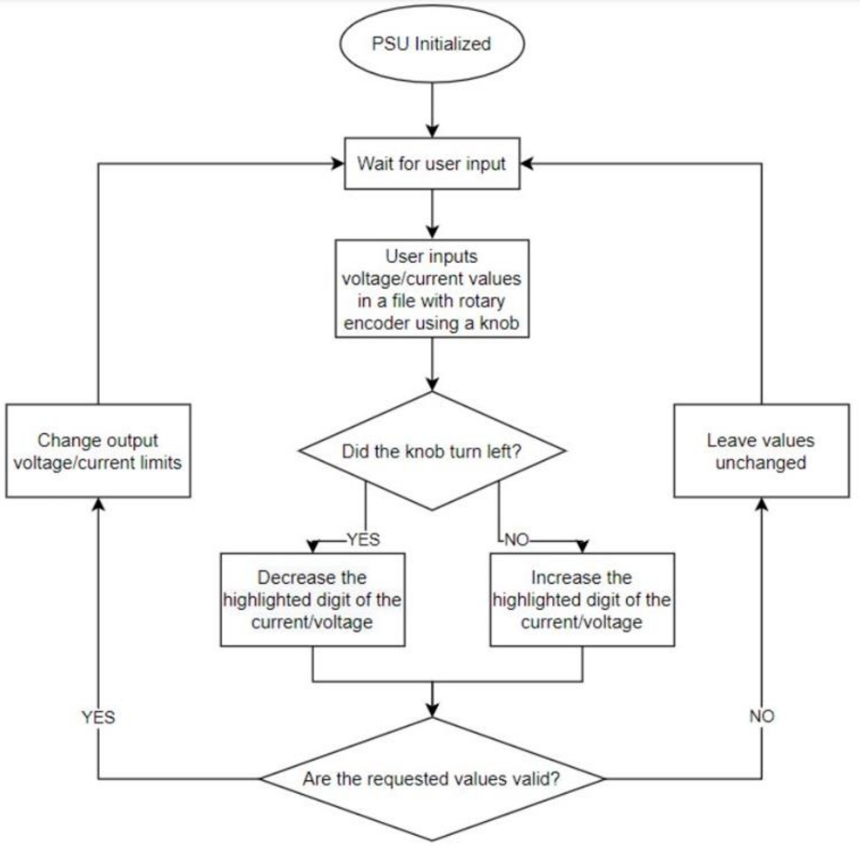


Figure 19: Rotary encoder flow chart

The following figure is a flowchart depicting the operations that occur when the user inputs voltage/current values using the keypad.

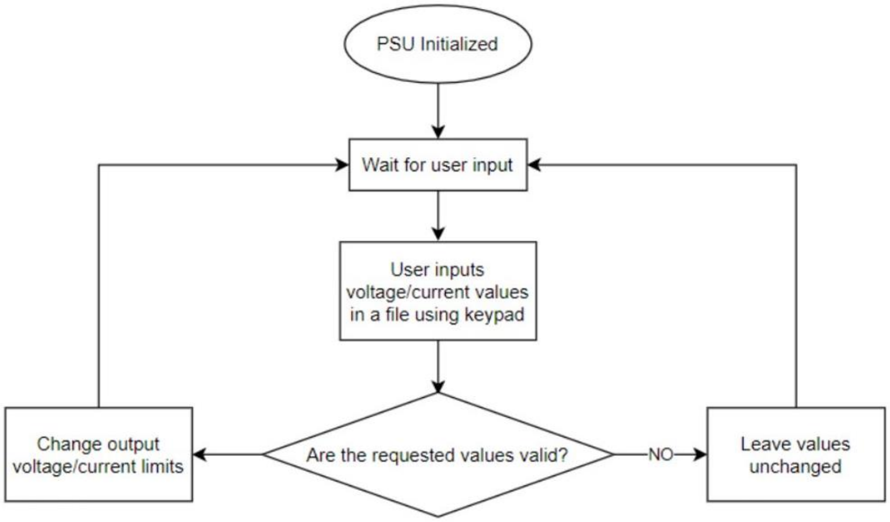


Figure 20: Keypad flow chart

The following figure is a flowchart depicting the software operations carried out when the user inputs SCPI commands to the system using the command line.

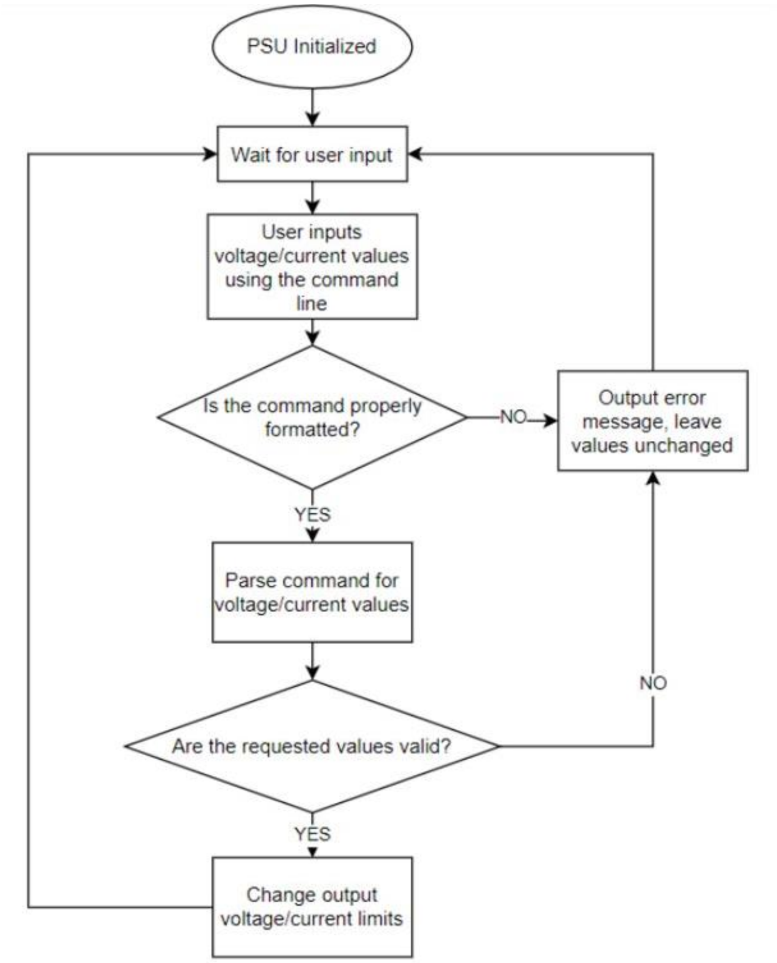


Figure 21: USB flow chart

Prototyping Progress Report

Our power supply needs out output up to 10V using a 12V RMS transformer as the input, so we do not have a lot of room to work with voltage drops. For a bridge rectifier we get an average voltage out of $2Vp\pi$. Our transformer guarantees us 12V rms so we can calculate Vp from that and we get $120.7071=16.97072$. Therefore, our average voltage out from a rectifier should be 10.80389V. This is extremely close to the 10V output we want after regulation but depending on the load this value may be higher. Since we are going to have a current limit of half an amp, the transformer will have a maximum load of around 60% so we should be able to get sufficient voltage. We had begun testing out a full bridge rectifier using Schottky diodes instead of more traditional diodes like the 1N4001 for example because of their comparatively low forward voltage drop since we must account for a voltage drop across two diodes and at least one regulator. We ran into several issues with testing the full wave rectifier because we were using a function generator and an oscilloscope connected to the same power line which gave us a half wave rectification no matter what. Testing the bridge rectifier with the transformer given to us went well since the transformer was properly isolated.

I. **LT3080 Prototyping**

The LT3080 gave us very few problems, the regulator performs as expected until we try to set a voltage less than ~0.8V because the regulator requires a 1ma current to be drawn and as we lower the voltage the resistor dummy load needs to become smaller to maintain regulation. A dummy load resistor of 50 ohms gave us output voltages as low as 14mV but we should produce a better method of sinking the minimum load current. We were previously considering using the LTC3631 to output a negative voltage to sink 4ma of current, but we have shifted our focus to a cheaper alternative. We have had some success in using a charge pump to get a negative voltage supply to return our minimum load to. A charge pump is much cheaper than another regulator circuit, but the current draw through the load resistor will be slightly higher since it will swing up and down and our minimum load must be at least 1ma. Figure 6 shows the output of the regulator in green and the current through the load resistor in blue. The charge pump does require up to 20ms to charge before it begins working for the minimum load and there would be a small amount of voltage generating if the regulator is enabled at the same time. The output voltage of this regulator follows the voltage at the set pin and should be within .3V, this means that for us to control it with a DAC we will need to use a multiplier with a gain of around 4. We are considering the LT1991 op amp for this.

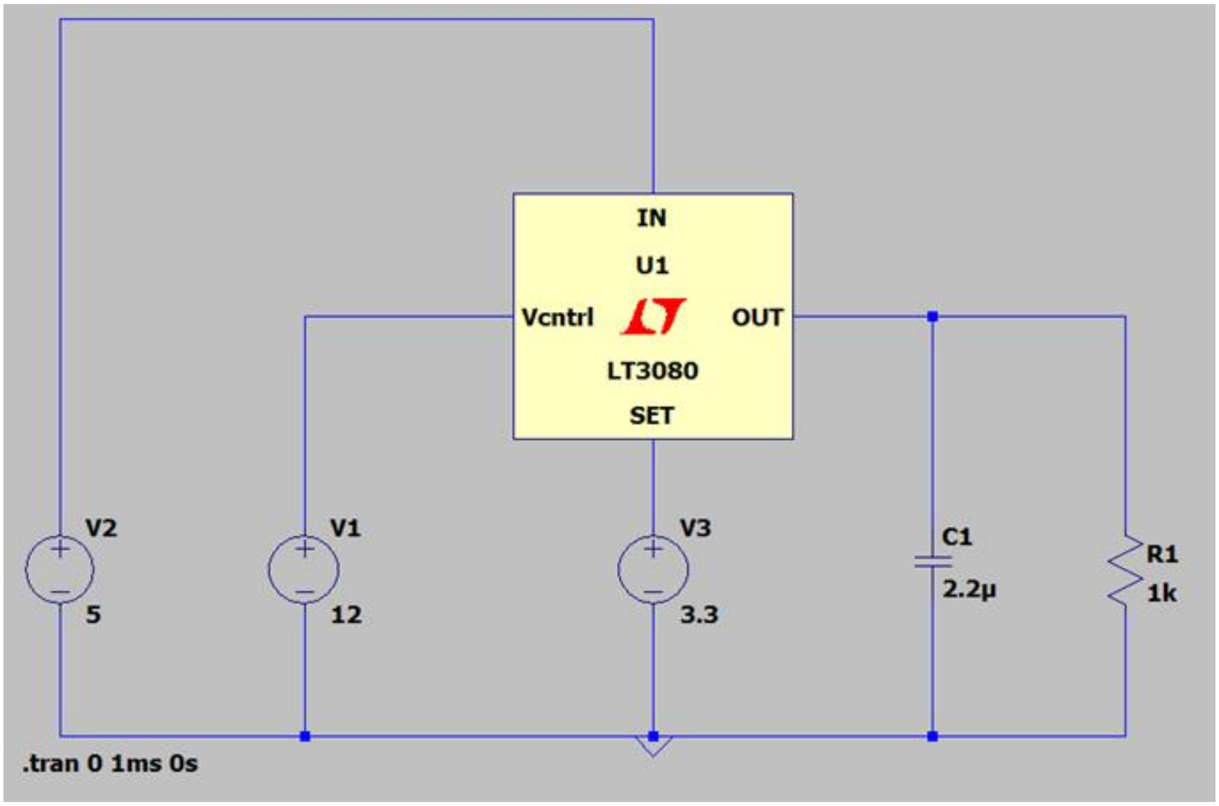


Figure 22: LT3080 demo setup using 3.3V set

Figure 23: LT3080 demo setup output simulation

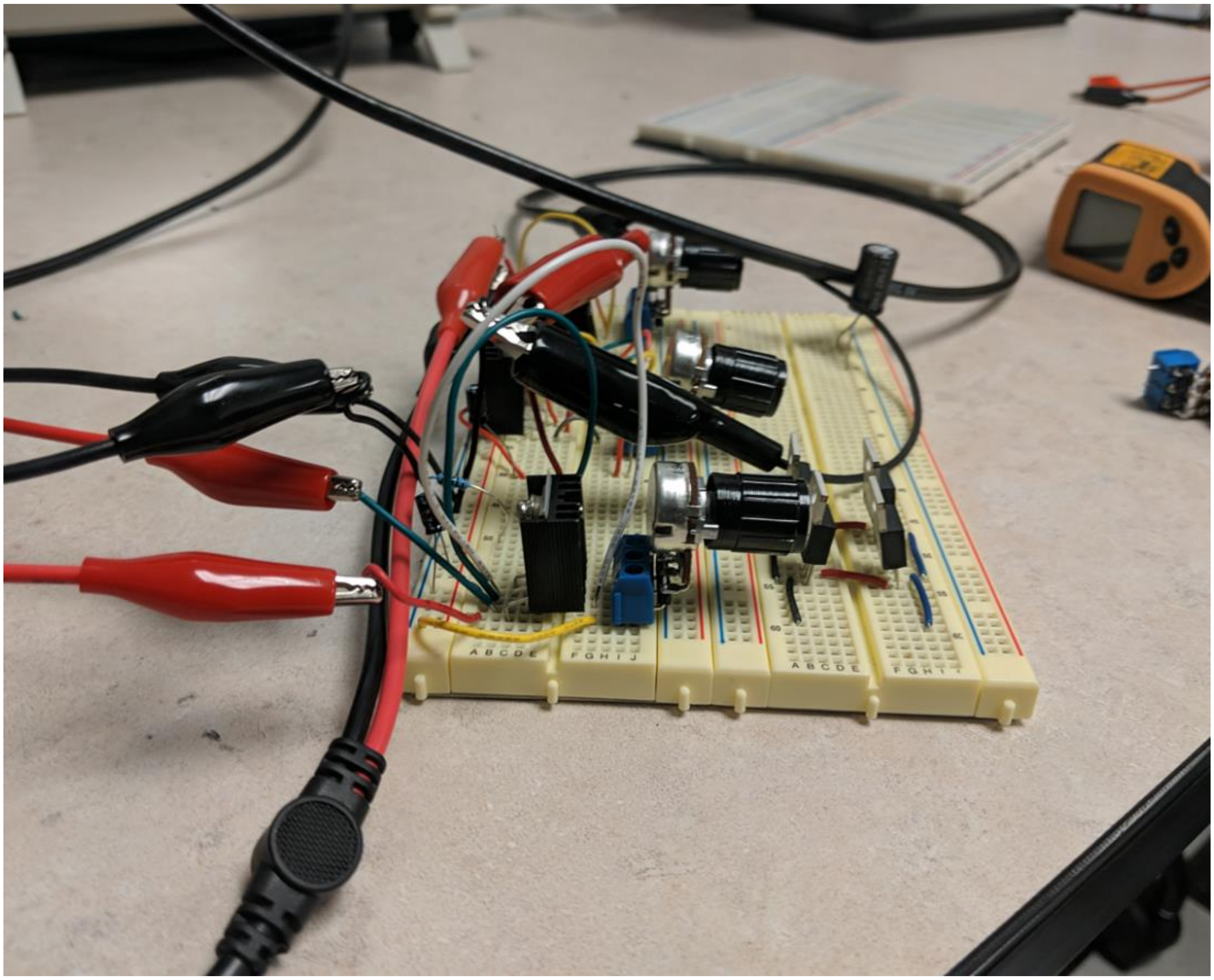


Figure 24: Breadboarded LT3080 demo using 1V set

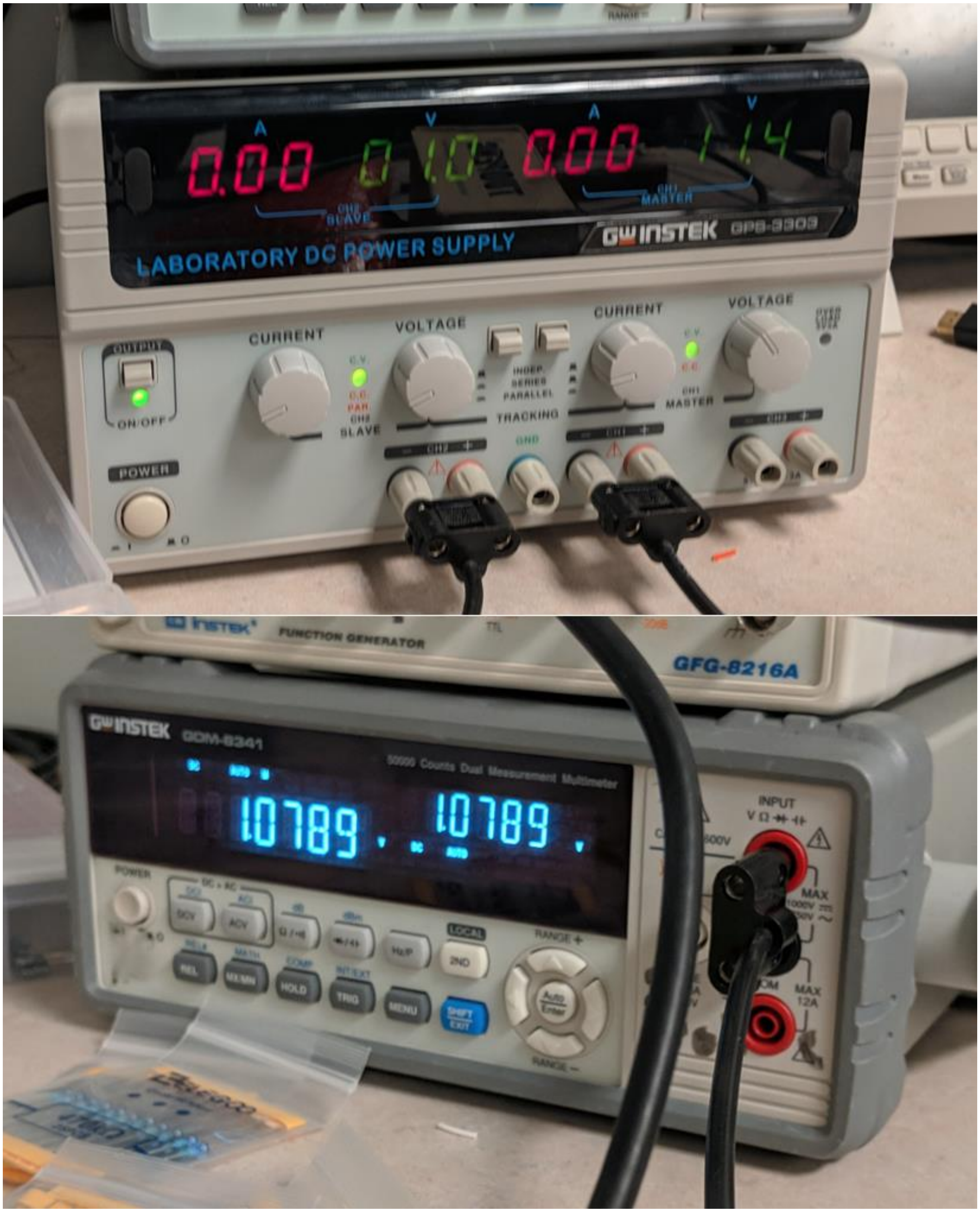


Figure 25: Breadboard demo output

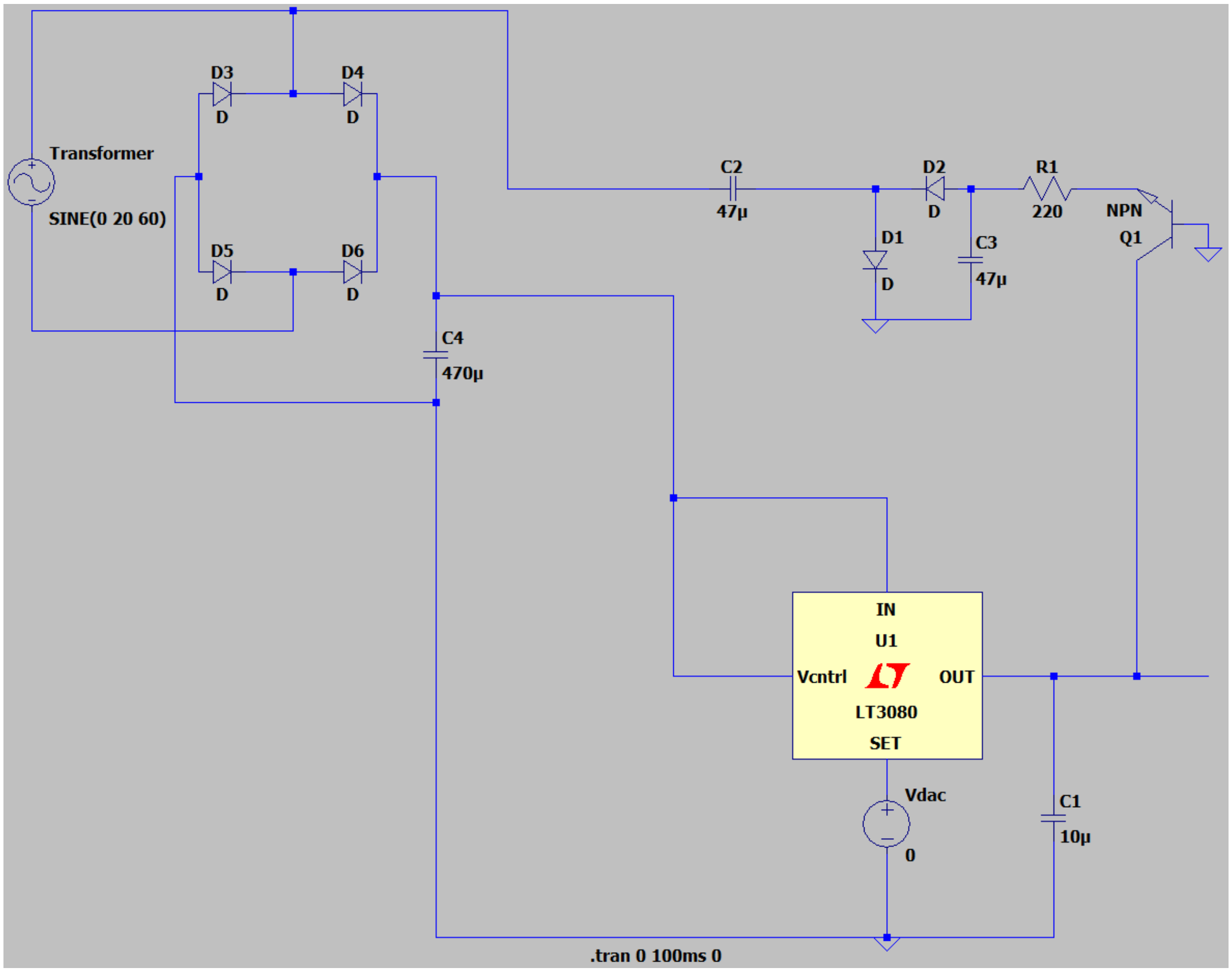


Figure 26: LT3080 charge pump demo

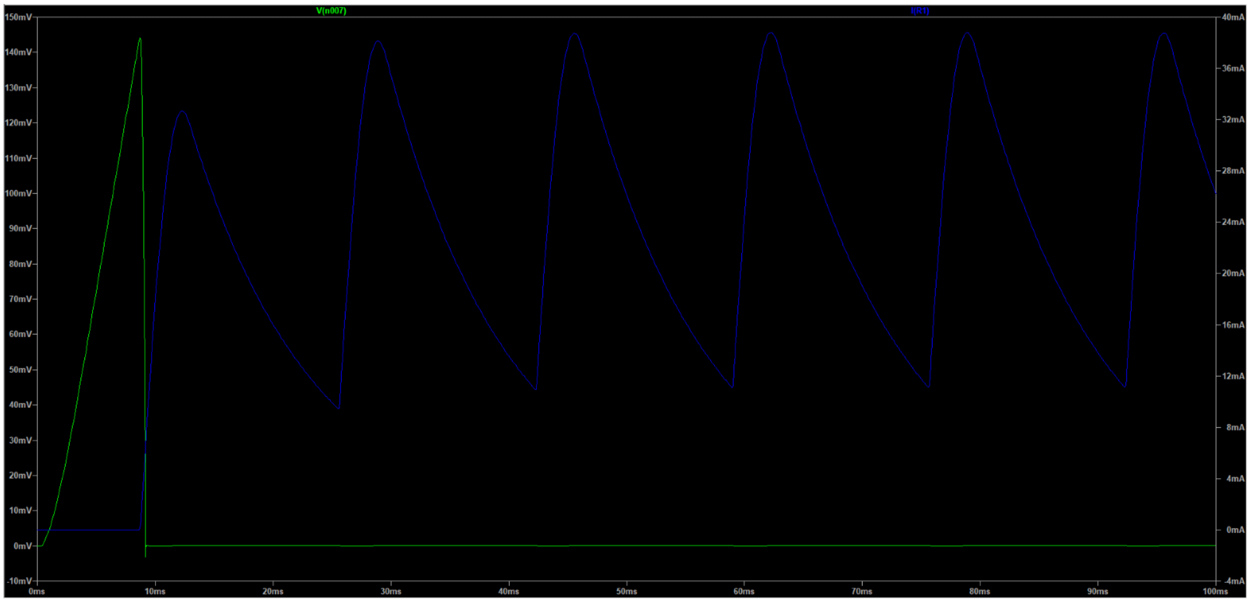


Figure 27: LT3080 charge pump simulation output

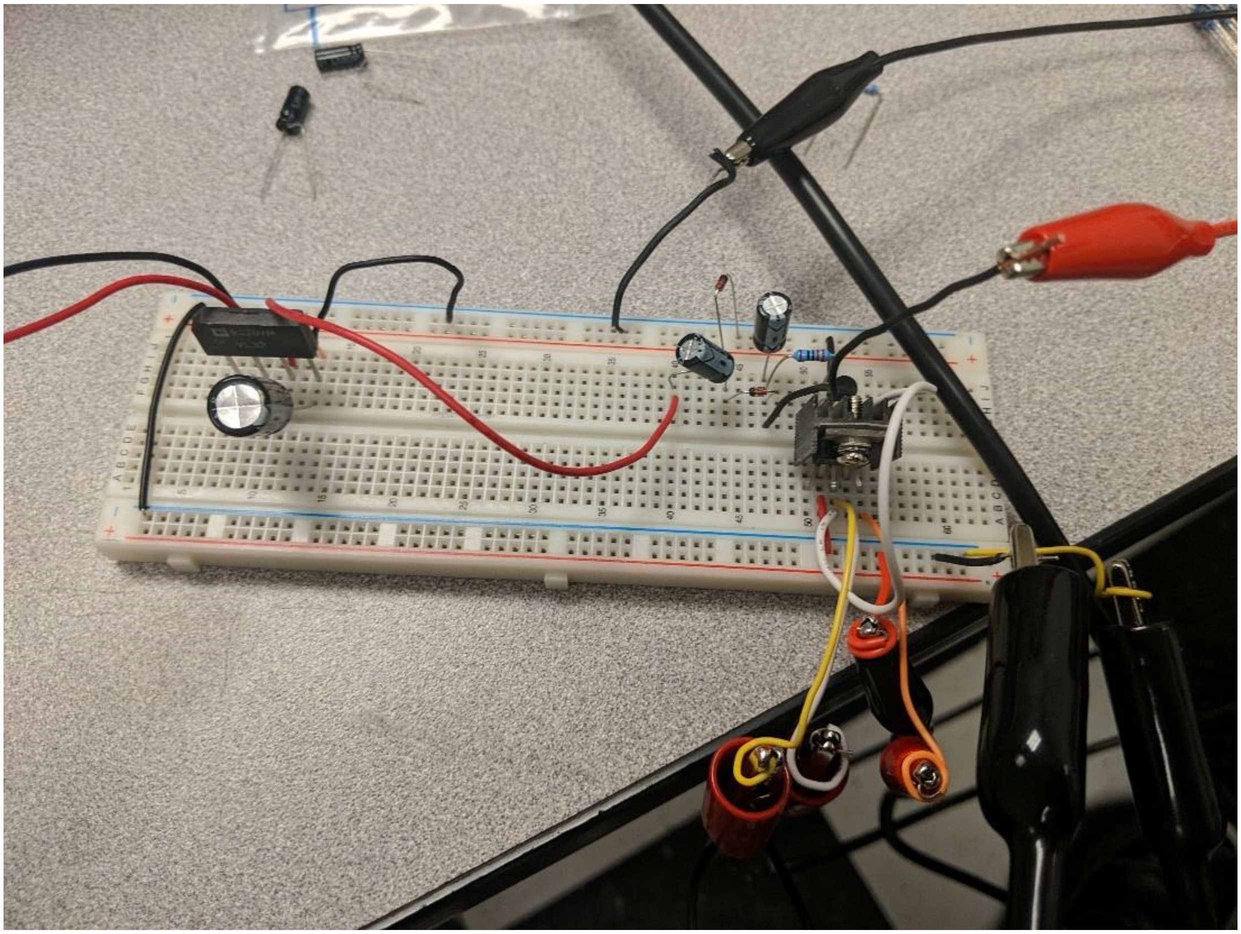


Figure 28: LT3080 charge pump breadboard demo

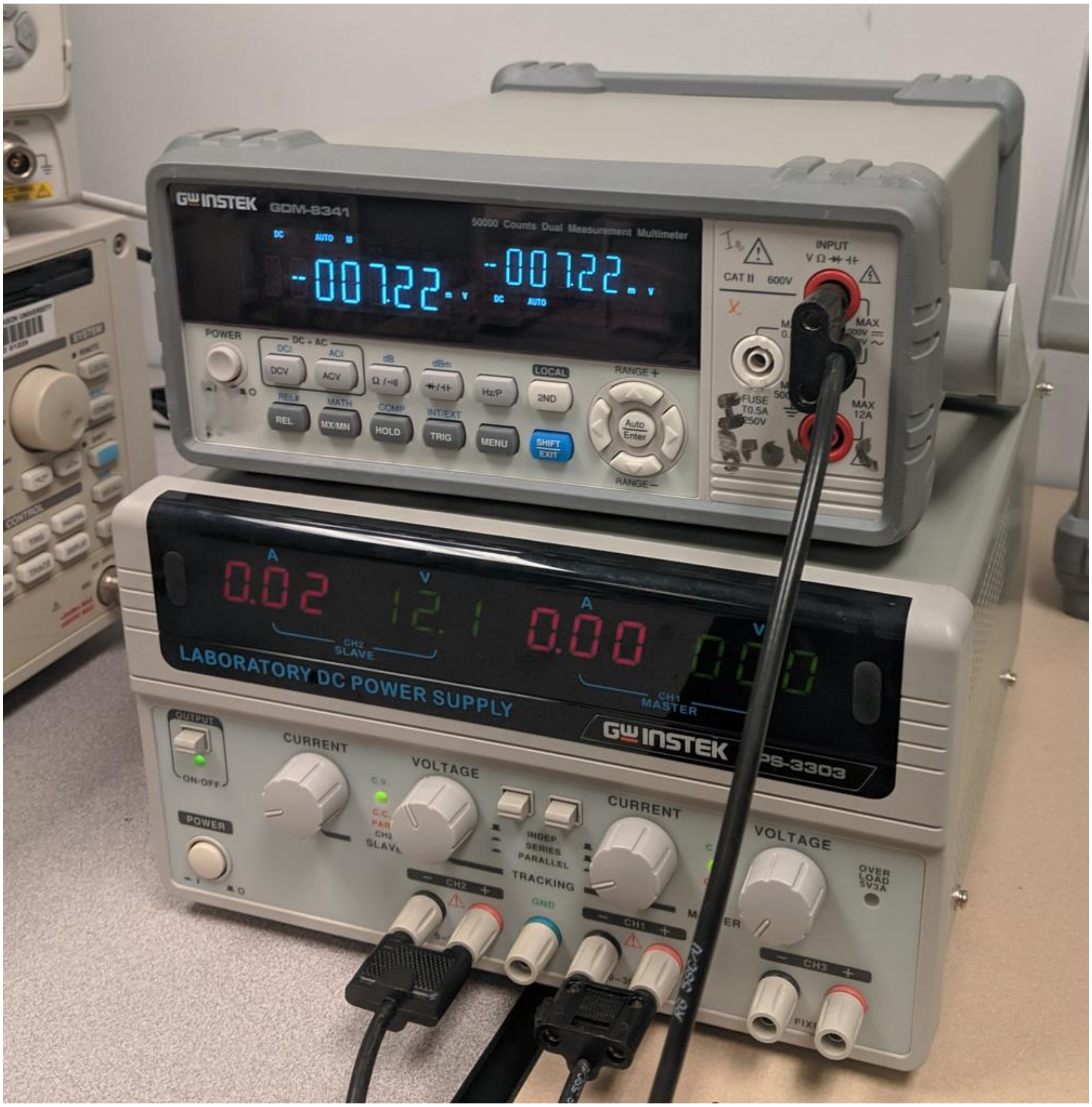


Figure 29: LT3080 charge pump breadboard demo output

II. **L6902 Prototyping**

The L6902 did not give us too much trouble once we figured out how to drive the feedback pin. We initially tried to drive it with a voltage directly and when we set a voltage below the internal reference voltage, we got out the max voltage and could not control it well. We determined that the L6902 could not regulate without feedback voltage from its output pin, so we were left with the resistor divider shown in figure 6 to set the voltage. To control this regulator with a DAC, we would need to add a third resistor called R3 to the node between the resistors going to the feedback pin and drive the DAC voltage though it. Applying KVL at this node gets us $Vfb=R3R2Vout+R1R2VdacR3R2+R1R3+R1R2$, simplifying and inserting the internal reference voltage gets us $Vout=1.2351+R1R2+R1R3-R1R3Vdac$. We can then determine the required resistor values by choosing R1 to be any value we want and solving a system of equations with two different sets of constraints for Vout and Vdac. Since our DAC goes from 0-3.3V those will be the two constraints for Vdac. When the DAC outputs 0V the regulator should be outputting its maximum voltage, we picked 11. When the DAC outputs 3.3V the regulator should be outputting its minimum voltage, we picked 1.5 to give us some room with the internal reference. With this we get $R2=815140984R1$ and $R3=3395R1$. In our demo we chose R1 to be 100k and thus R2 was approximately 20k and R3 was approximately 34k. We saw a voltage range of ~1.48V to ~10.8V which confirms that our math adds up. In the end, we may wish to drop this regulator in favor of a heatsink on the LT3080 to save on the price at the cost of efficiency.

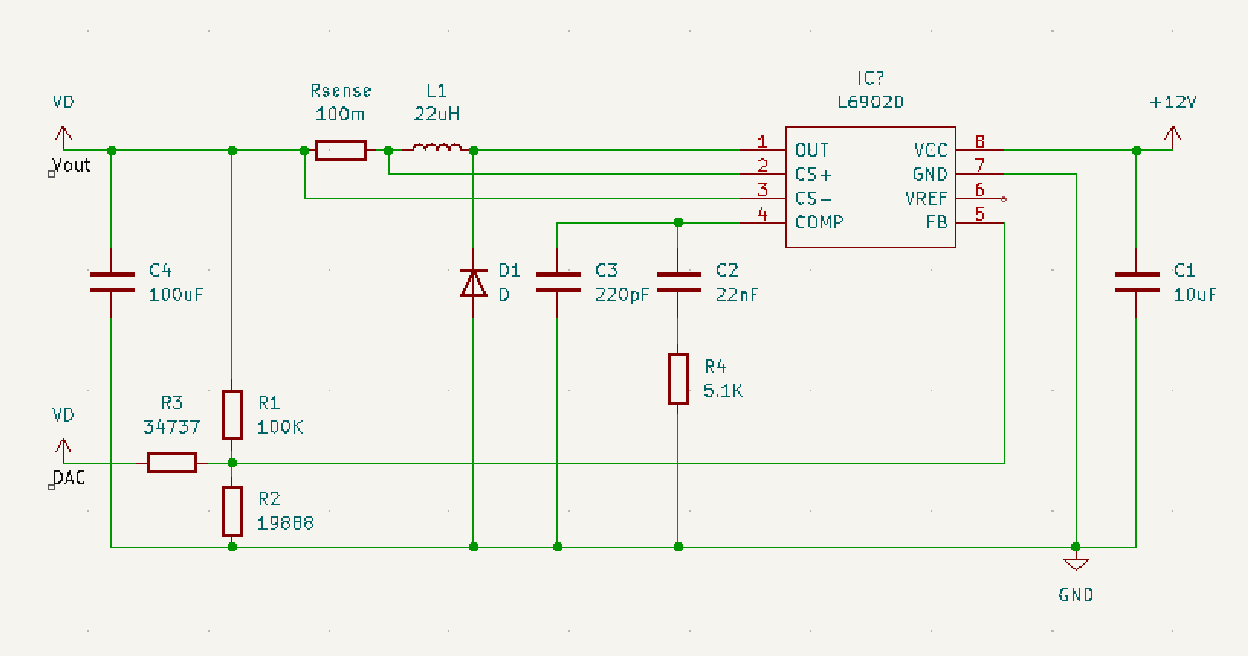


Figure 30: L6902 demo setup using a DAC to control Vout

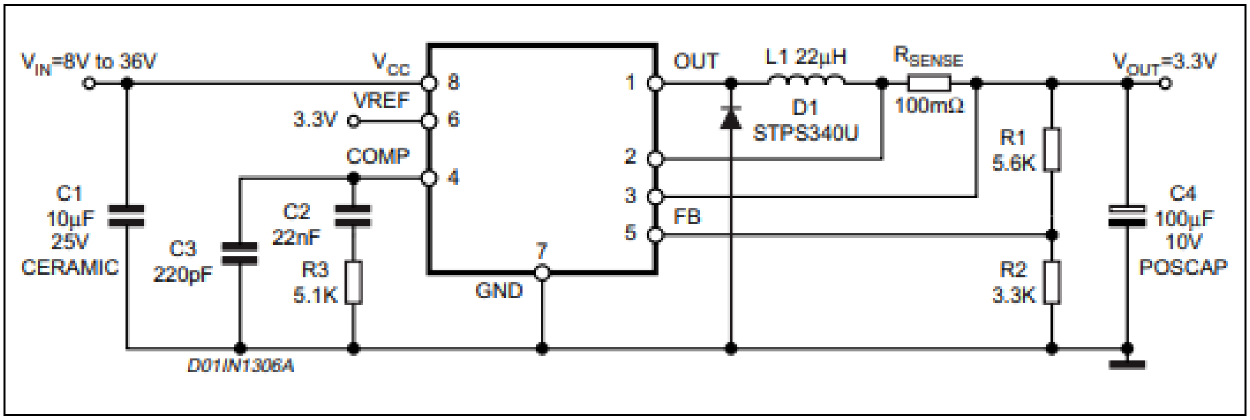


Figure 31: Application circuit from datasheet

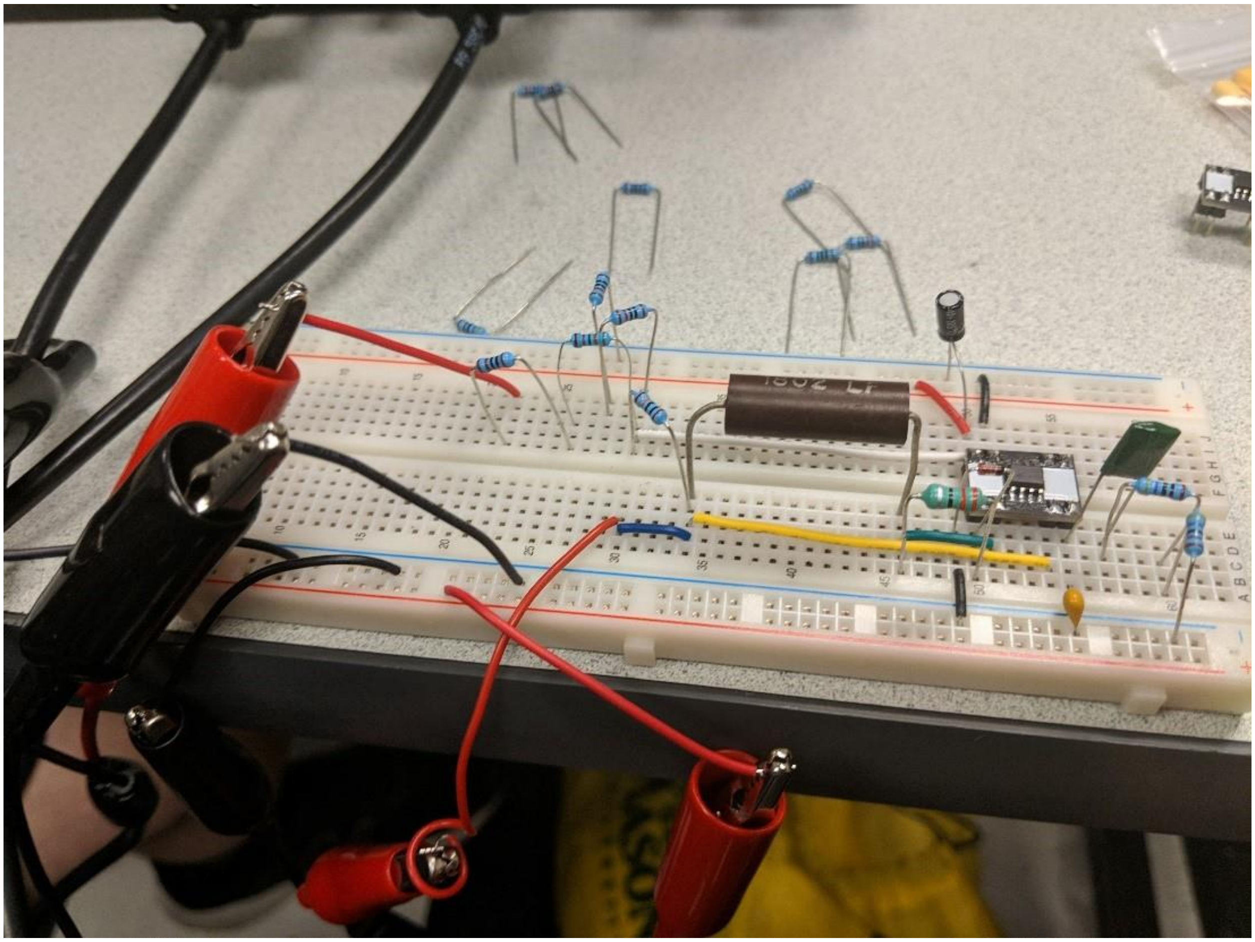


Figure 32: Breadboarded L6902 demo

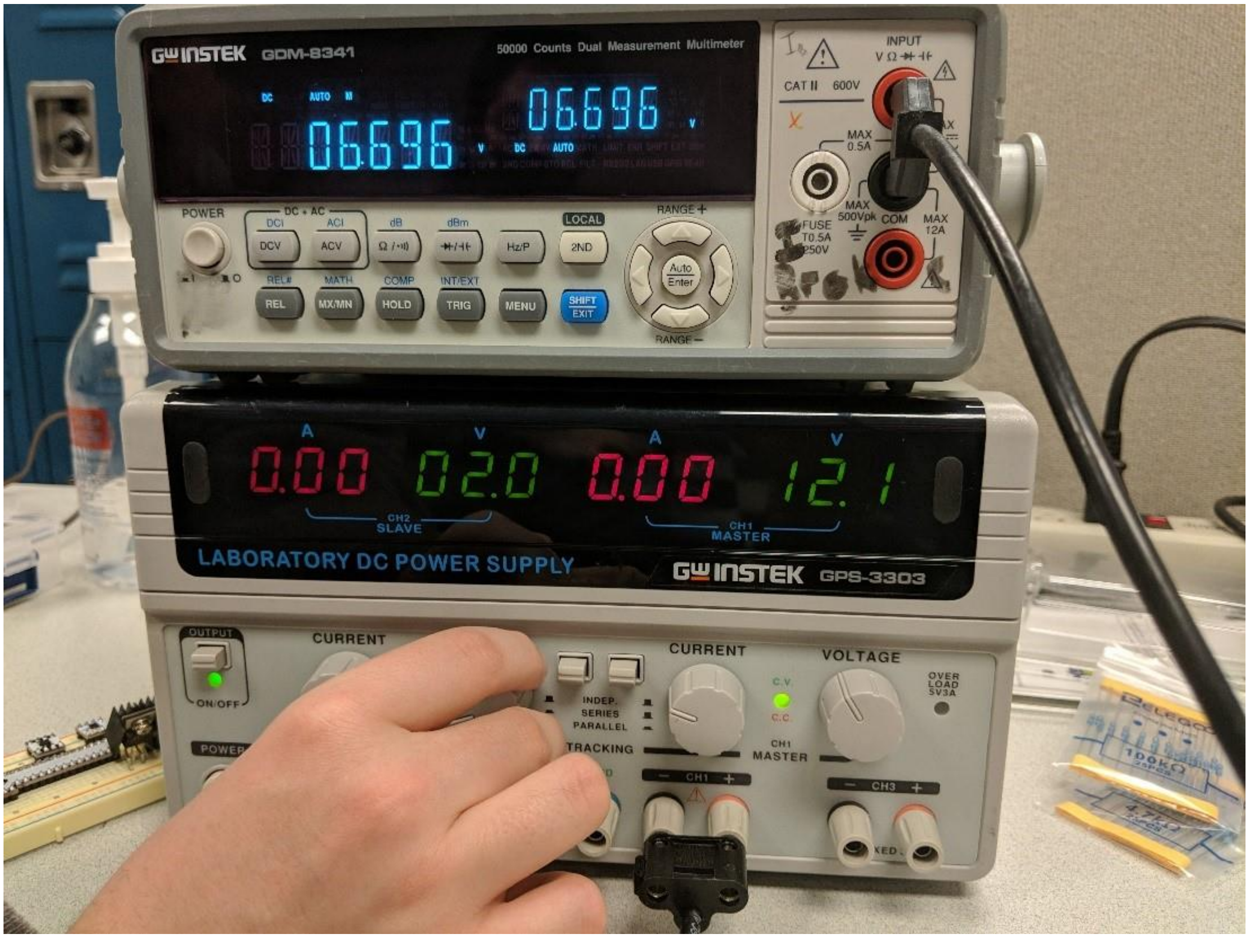


Figure 33: Breadboard demo output ~6.6Vout with 2.0Vfb

III. **Static Regulator Prototyping**

The LM2672M buck converters (Figure 34) we chose were a bit more sensitive to using the proper capacitors but once we got the parts it worked flawlessly (Figures 35, 36). We are planning to use these step-down converters to power the LCD as well as the MCUs. We have not yet tested the ripple of these regulators and if we find that the ripple exceeds the input ripple specifications of the MCUs we will need to switch to using a linear regulator for the 5V rails, either another LT regulator or a more common LM317.

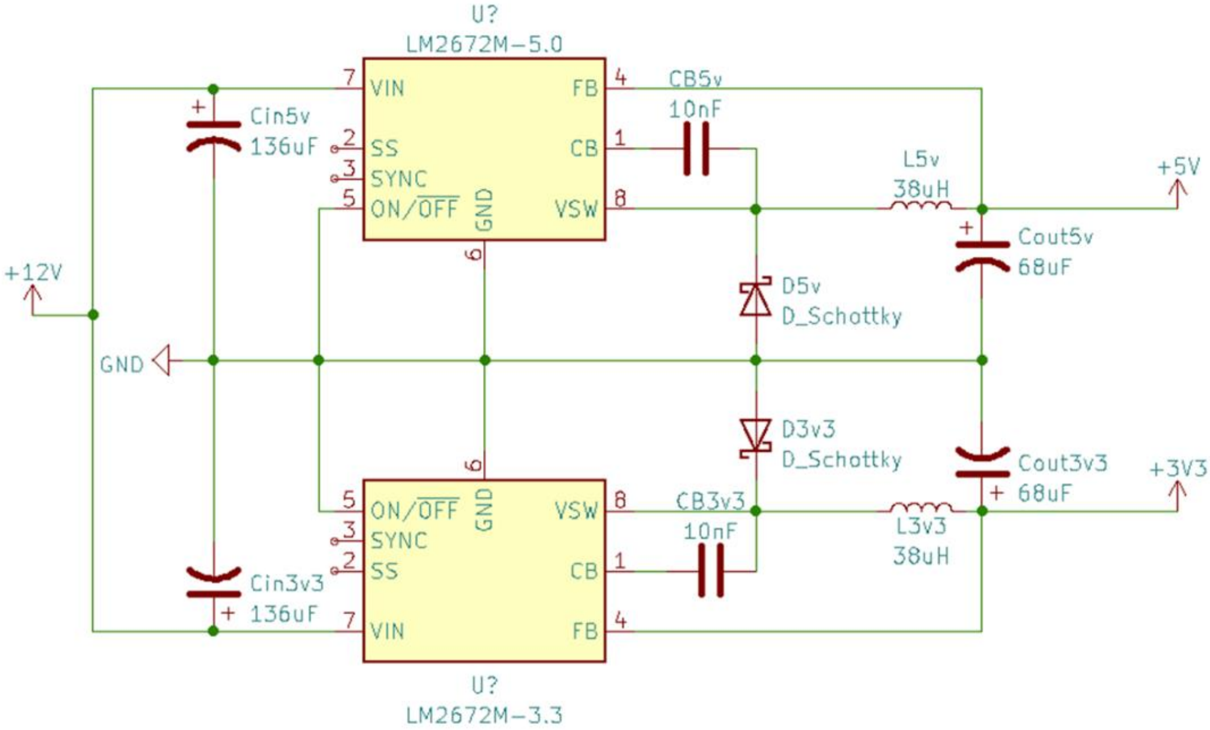


Figure 34: Demo setup for the 3.3V/5V static regulators

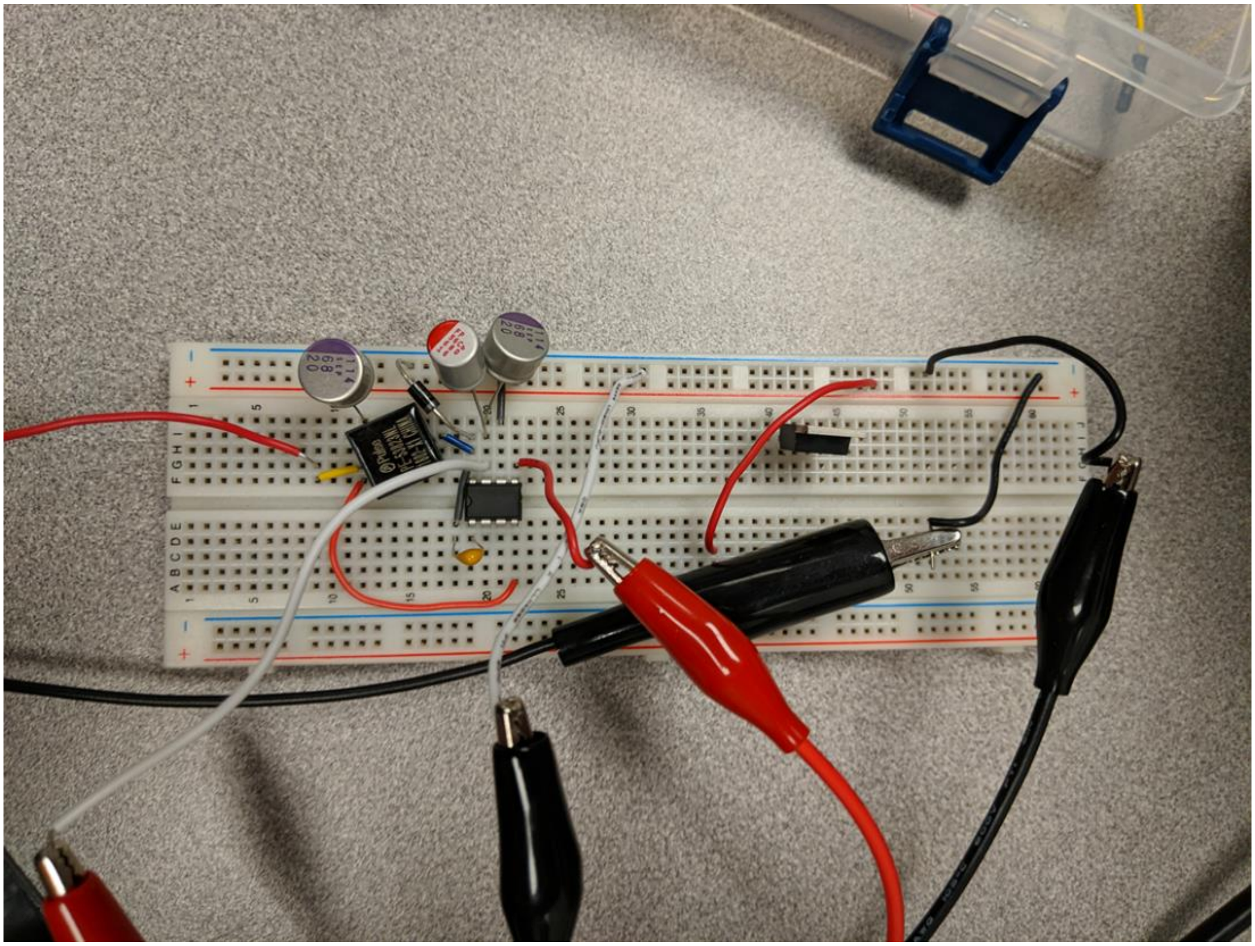


Figure 35: Breadboarded 5V regulator setup

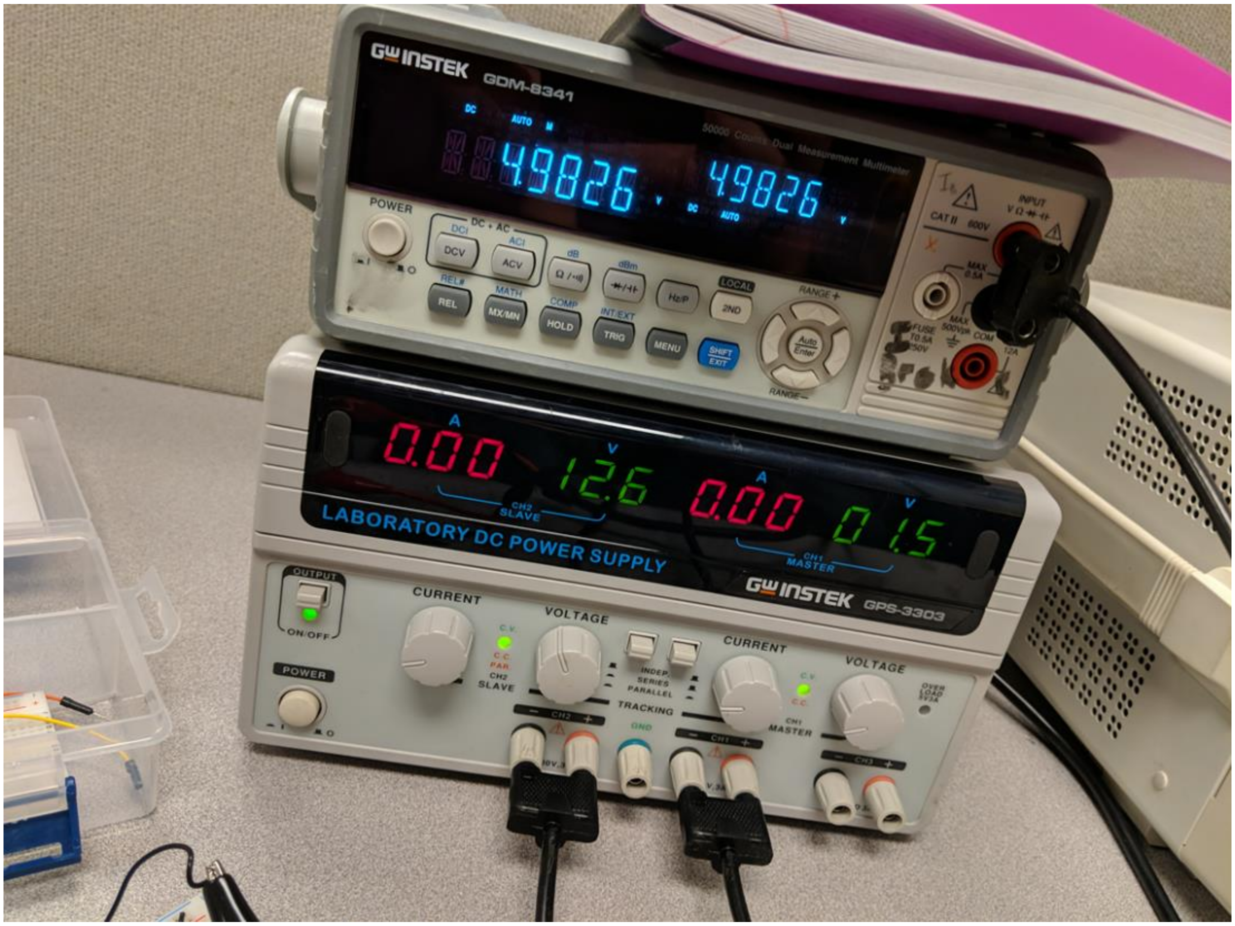
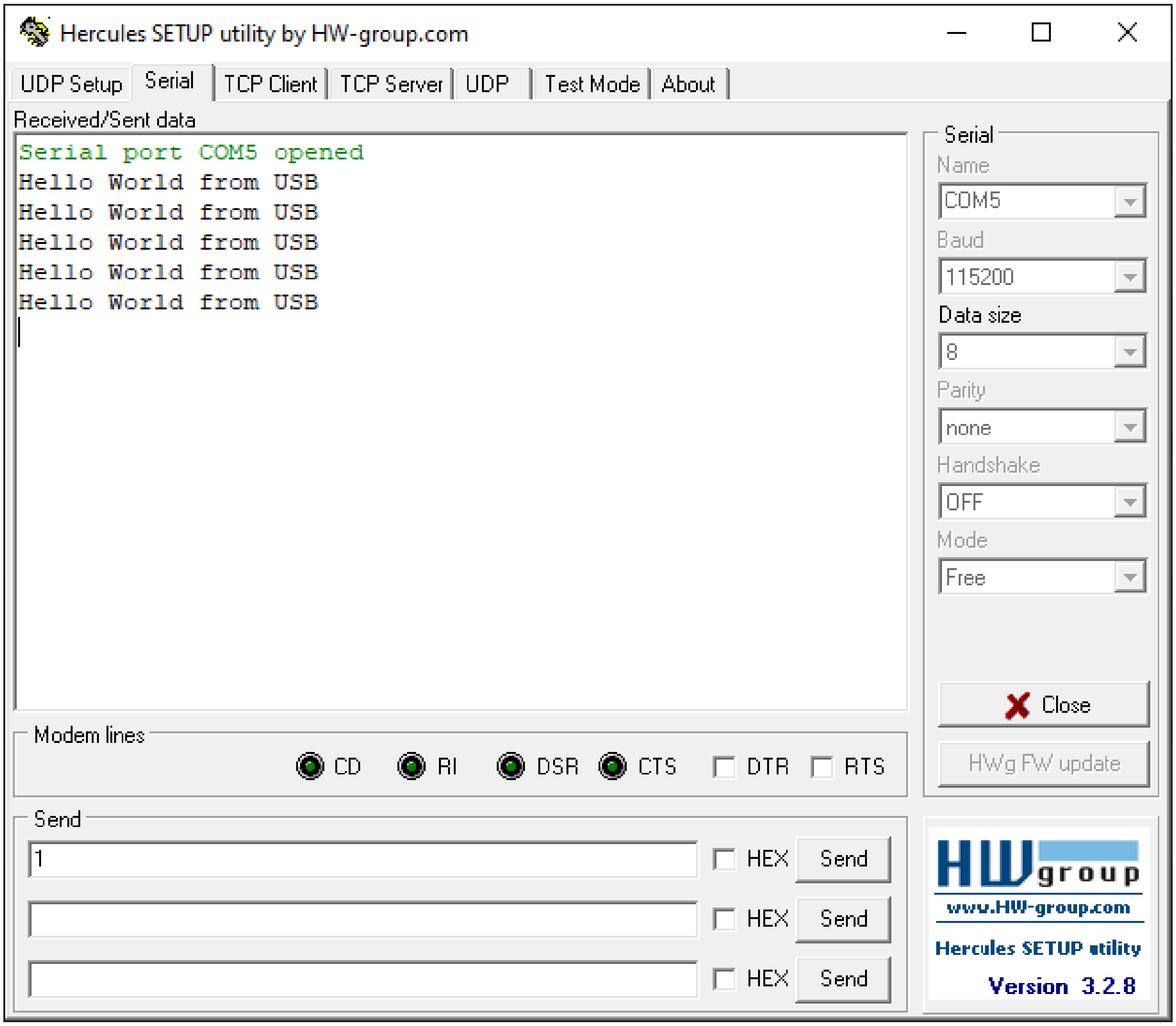


Figure 36: Breadboard demo output of 5V regulator

IV. **MCU Programming**

As for prototyping the MCU, we have been using the STM32L152C-DISCO board and we immediately ran into problems with it. We spent a few weeks using STM32CubeIDE, but we could never get the debugger to work properly, it would die after a few lines of code are executed in the main function. We also had issues where inputs from things like the rotary encoder were not consistent. When we set the MCU to turn on an LED it would constantly flash instead of being a solid light, we believe that the MCU was constantly resetting itself when using this IDE. We recently dropped STM32CubeIDE and went with Keil uVision5 which works flawlessly with the default HAL drivers provided by STM. The debugger works properly, and we get consistent results from inputs, so we believe that something was repeatedly shutting down the MCU when using the other IDE. We have run into an issue with this IDE though where when we use STM32CubeMX to generate the USB drivers, it does not generate the USB_DEVICE driver set, and we cannot compile without it. As it turns out, STM32CubeMX generates another Keil project file further into the project directory which contains all the libraries properly imported saving us the need to manually import the libraries. The final issue we ran into was related to USB. When we enabled USB and set the clocks up for it the MCU would halt on setting up the clocks but when USB was disabled, and the clocks were unchanged it could proceed. The problem was that STM32CubeMX was not generating the code for the external crystal oscillator when USB was disabled even when we set it up in the clock configurator so we could not figure out what the problem was until we noticed this. We had mistaken the crystal oscillator on the board for the external oscillator for our MCU, but it was in fact only for the ST-Link debugger port, so our code was halting because we had no oscillator for the clocks. A quick fix for our demo boards was to solder the pins from the oscillator for the debugger to the pins on our MCU with some wires for its oscillator and just piggyback on the signal. We have since been able to do some simple USB communication as can be seen in figure 16, including the ability to toggle an LED by sending a “1” to the MCU.



```
Serial port COM5 opened
Hello World from USB
Hello World from USB
Hello World from USB
Hello World from USB
Hello World from USB
```

Name: COM5
Baud: 115200
Data size: 8
Parity: none
Handshake: OFF
Mode: Free

Send: 1

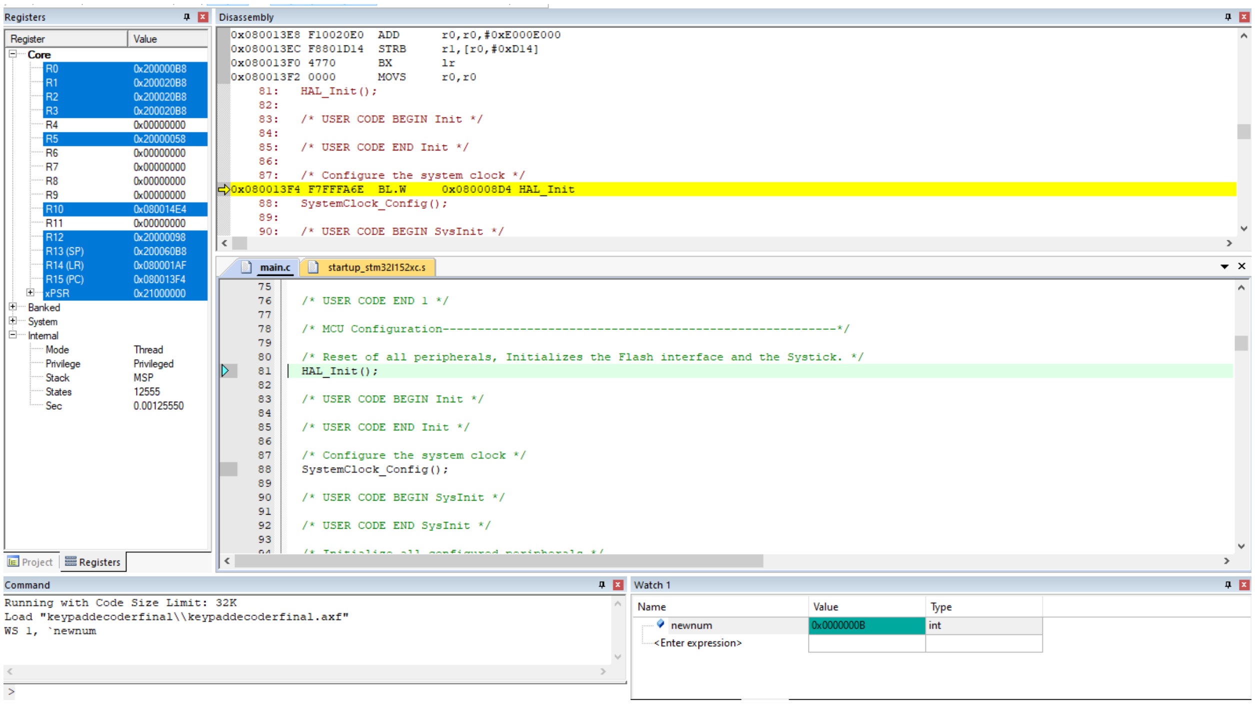
Figure 37: Hercules virtual communication port on MCU

The main front panel display will feature a 16x4 liquid crystal display (LCD), for Figure 38 a 20x4 LCD was used as the part had not arrived. The display here is controlled via a

PCF8574T I2C module and will be used to showcase to the user the chosen output voltage and current limit — as input from the rotary encoder, keypad, or SCPI — as well as the actual output voltage and current limit as read by the MCU. This will be shown for both channels and will have an accuracy in the millivolt/milliamp range. Consequently, non-numerical values, e.g., CH1, representing channel-one and its voltage and current limit settings, will have to be labelled outside of the LCD particularly for the 16x4 to accommodate space. Furthermore, depending on the LCD type used — either an LM2672M-3.3 and/or an LM2672M-5 will have to be used to provide power for the specific VCC.

Figure 38: LCD2004 I2C Demo

We have also successfully interfaced a keypad with our MCU as a prototype for accepting keypad inputs to adjust output currents/voltages on our finished power supply. This was relatively simple to do conceptually as we previously had to decode a keypad for a lab in a previous class, though the actual implementation was more difficult than expected. This was primarily due to the struggle to generate a project file using Keil instead of STM32CubeIDE as well as the interrupts working in a different way compared to MCUs we have used before. GPIO interrupts in the STM32 MCU share a common interrupt value for all pins of the same number (i.e., A0 and B0 both correlate to the same interrupt). This means we only have 16 interrupts to work with (our MCU’s pins are numbered 0-15), though since our input devices are limited to a keypad, two rotary encoders, and a few extra buttons, this lack of interrupts shouldn’t be an issue as we proceed with designing the power supply. Once the interrupts were properly established, the program ran as expected, although the keypad has not been properly debounced yet. The output of this can be observed in the figure above, where the specific key is designated by a variable called “newnum”, and this value ranges from 0-15 corresponding to which button was pressed on the keypad.



Register	Value
Core	
R0	0x200000B8
R1	0x200020B8
R2	0x200020B8
R3	0x200020B8
R4	0x00000000
R5	0x20000058
R6	0x00000000
R7	0x00000000
R8	0x00000000
R9	0x00000000
R10	0x080014E4
R11	0x00000000
R12	0x20000098
R13 (SP)	0x200060B8
R14 (LR)	0x080001AF
R15 (PC)	0x080013F4
xPSR	0x21000000
Banked	
System	
Internal	
Mode	Thread
Privilege	Privileged
Stack	MSP
States	12555
Sec	0.00125550

```
0x080013E8 F10020E0  ADD      r0,r0,#0xE000E000
0x080013EC F8801D14  STRB     r1,[r0,#0xD14]
0x080013F0 4770      BX       lr
0x080013F2 0000      MOVS     r0,r0
    81:   HAL_Init();
    82:
    83:   /* USER CODE BEGIN Init */
    84:
    85:   /* USER CODE END Init */
    86:
    87:   /* Configure the system clock */
0x080013F4 F7FFFA6E  BL.W     0x080008D4 HAL_Init
    88:   SystemClock_Config();
    89:
    90:   /* USER CODE BEGIN SysInit */
```

```

  /* USER CODE END 1 */

  /* MCU Configuration----------------------------------------------------------*/

  /* Reset of all peripherals, Initializes the Flash interface and the Systick. */
  HAL_Init();

  /* USER CODE BEGIN Init */

  /* USER CODE END Init */

  /* Configure the system clock */
  SystemClock_Config();

  /* USER CODE BEGIN SysInit */

  /* USER CODE END SysInit */

```

```
Running with Code Size Limit: 32K
Load "keypaddecoderfinal\\keypaddecoderfinal.axf"
WS 1, `newnum
```

Name	Value	Type
newnum	0x0000000B	int
<Enter expression>		

Figure 39: Debugging Output of Keypad Decoder after B Button is Pushed

Testing Plan

All our prototyping efforts so far have been done using resistors as dummy loads and this is not a good indication of how our circuits will behave under load. Dr. Kaps has told us that he has procured a load tester that will sink any amount of current that we could use for testing purposes. We will have to ensure that our regulator circuit can maintain regulation at the voltages required under heavier loads up to half an amp.

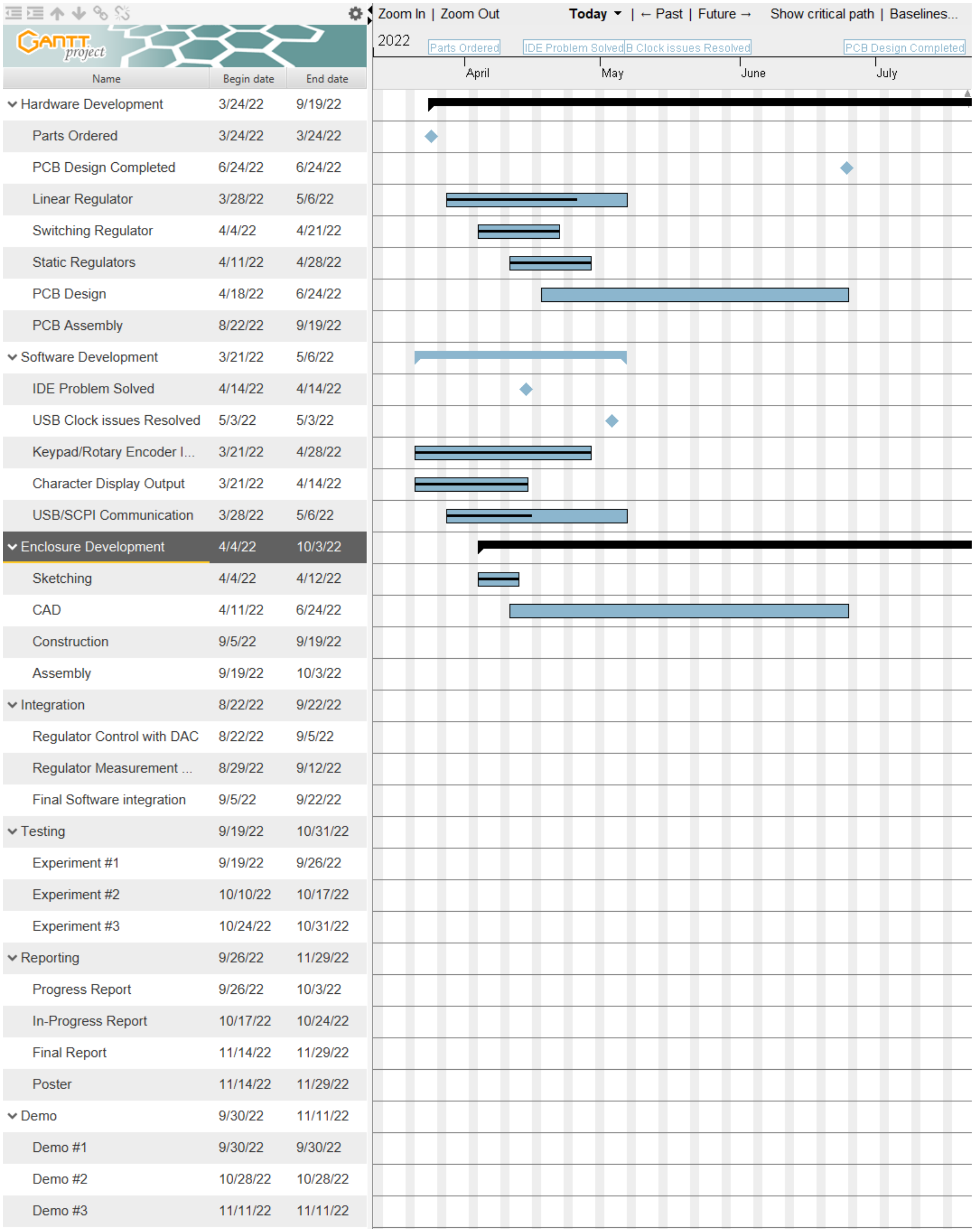
For the linear regulator testing we first need to test the operation of an op-amp based multiplier with a gain of 4 so we can use the DAC to adjust the output of the LT3080. This testing would involve attempting to get out voltages between 0-12V with ~4mv steps since our DAC has a resolution of ~1mv. Once we have achieved this, we would then test the regulator circuit driven by the DAC and attempt to get output voltages from at least 0-10V if not 0-12V. The key here would be 0V operation, or more specifically near 0V operation since for true 0V operation we would turn off the circuit. We have had problems with the regulator achieving 0V or near 0V because of the load issue and this is the most critical section.

For the switching regulator testing we first need to determine what kind of resolution we have on the output voltage using the resistor divider network and we could do this with the AD2 if we do not exceed 5V since waveforms would make it quite easy to get this data. We then need to successfully integrate the L6902 with the LT3080 and make the same tests as with the LT3080. In the end though, we may choose to drop this switching regulator in favor of a heatsink after the integration problems we ran into.

For testing the I/O we will have the MCU output to a serial terminal using USB to test that the keypad, buttons, and rotary encoders are working and are debounced. We then will test having them work with the LCD to set voltage and amperage and we will need to test that they do not allow invalid inputs. Once this is done, we will move on to integrating the MCU with the regulator circuit and performing the same tests as with the LT3080 but using the MCU to drive the set pin.

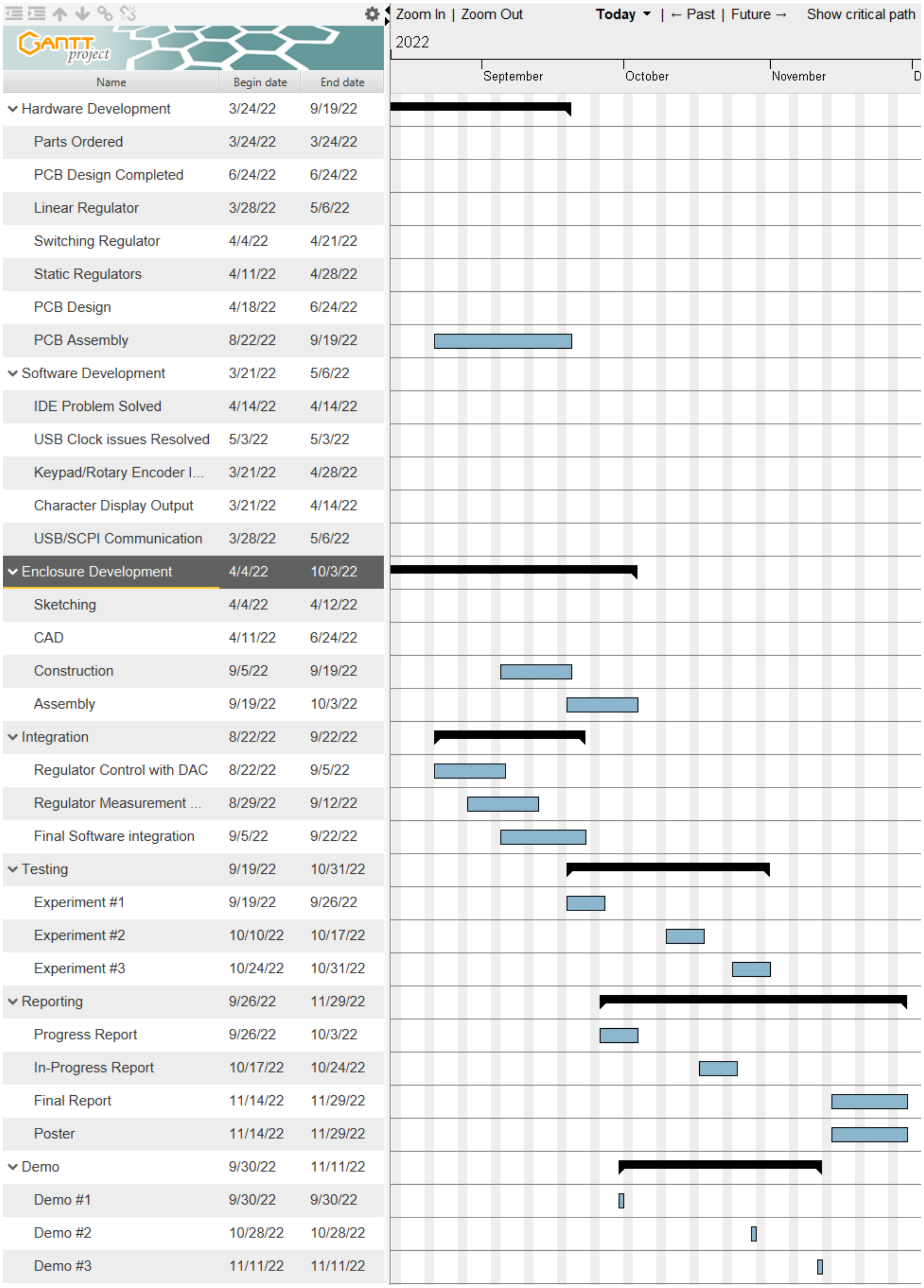
Finally, to fully test the system we will attach a 0.5A DC load to channel 1 and set the output voltage from 0-10V using the keypad, verifying that both the output and the readouts are accurate. We will then test that the channel power button works as expected and provides 0V. We will then repeat this test on channel 2. This test will then be repeated using the rotary encoder instead of the keypad, and then using USB instead of the rotary encoder. Finally, we will repeat this test using both channels interconnected to test from 0-24V to verify that each channel is electrically isolated.

Tasks for ECE-493



Name	Begin date	End date
Hardware Development	3/24/22	9/19/22
Parts Ordered	3/24/22	3/24/22
PCB Design Completed	6/24/22	6/24/22
Linear Regulator	3/28/22	5/6/22
Switching Regulator	4/4/22	4/21/22
Static Regulators	4/11/22	4/28/22
PCB Design	4/18/22	6/24/22
PCB Assembly	8/22/22	9/19/22
Software Development	3/21/22	5/6/22
IDE Problem Solved	4/14/22	4/14/22
USB Clock issues Resolved	5/3/22	5/3/22
Keypad/Rotary Encoder I...	3/21/22	4/28/22
Character Display Output	3/21/22	4/14/22
USB/SCPI Communication	3/28/22	5/6/22
Enclosure Development	4/4/22	10/3/22
Sketching	4/4/22	4/12/22
CAD	4/11/22	6/24/22
Construction	9/5/22	9/19/22
Assembly	9/19/22	10/3/22
Integration	8/22/22	9/22/22
Regulator Control with DAC	8/22/22	9/5/22
Regulator Measurement ...	8/29/22	9/12/22
Final Software integration	9/5/22	9/22/22
Testing	9/19/22	10/31/22
Experiment #1	9/19/22	9/26/22
Experiment #2	10/10/22	10/17/22
Experiment #3	10/24/22	10/31/22
Reporting	9/26/22	11/29/22
Progress Report	9/26/22	10/3/22
In-Progress Report	10/17/22	10/24/22
Final Report	11/14/22	11/29/22
Poster	11/14/22	11/29/22
Demo	9/30/22	11/11/22
Demo #1	9/30/22	9/30/22
Demo #2	10/28/22	10/28/22
Demo #3	11/11/22	11/11/22

Figure 40: Gantt chart of tasks from 492

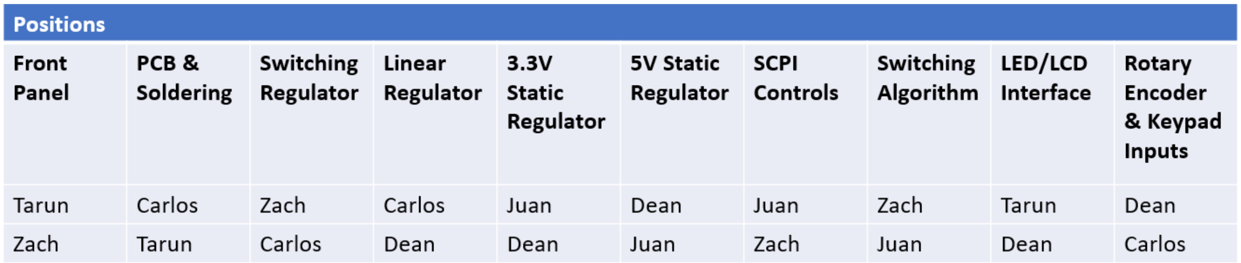


Name	Begin date	End date
Hardware Development	3/24/22	9/19/22
Parts Ordered	3/24/22	3/24/22
PCB Design Completed	6/24/22	6/24/22
Linear Regulator	3/28/22	5/6/22
Switching Regulator	4/4/22	4/21/22
Static Regulators	4/11/22	4/28/22
PCB Design	4/18/22	6/24/22
PCB Assembly	8/22/22	9/19/22
Software Development	3/21/22	5/6/22
IDE Problem Solved	4/14/22	4/14/22
USB Clock issues Resolved	5/3/22	5/3/22
Keypad/Rotary Encoder I...	3/21/22	4/28/22
Character Display Output	3/21/22	4/14/22
USB/SCPI Communication	3/28/22	5/6/22
Enclosure Development	4/4/22	10/3/22
Sketching	4/4/22	4/12/22
CAD	4/11/22	6/24/22
Construction	9/5/22	9/19/22
Assembly	9/19/22	10/3/22
Integration	8/22/22	9/22/22
Regulator Control with DAC	8/22/22	9/5/22
Regulator Measurement ...	8/29/22	9/12/22
Final Software integration	9/5/22	9/22/22
Testing	9/19/22	10/31/22
Experiment #1	9/19/22	9/26/22
Experiment #2	10/10/22	10/17/22
Experiment #3	10/24/22	10/31/22
Reporting	9/26/22	11/29/22
Progress Report	9/26/22	10/3/22
In-Progress Report	10/17/22	10/24/22
Final Report	11/14/22	11/29/22
Poster	11/14/22	11/29/22
Demo	9/30/22	11/11/22
Demo #1	9/30/22	9/30/22
Demo #2	10/28/22	10/28/22
Demo #3	11/11/22	11/11/22

Figure 41: Gantt chart of tasks to do in 493

Name	Begin date	End date
Hardware Development	3/24/2022	9/19/2022
Parts Ordered	3/24/2022	3/24/2022
PCB Design Completed	6/24/2022	6/24/2022
Linear Regulator	3/28/2022	5/6/2022
Switching Regulator	4/4/2022	4/21/2022
Static Regulators	4/11/2022	4/28/2022
PCB Design	4/18/2022	6/24/2022
PCB Assembly	8/22/2022	9/19/2022
Software Development	3/21/2022	5/6/2022
IDE Problem Solved	4/14/2022	4/14/2022
USB Clock issues Resolved	5/3/2022	5/3/2022
Keypad/Rotary Encoder Inputs	3/21/2022	4/28/2022
Character Display Output	3/21/2022	4/14/2022
USB/SCPI Communication	3/28/2022	5/6/2022
Enclosure Development	4/4/2022	10/3/2022
Sketching	4/4/2022	4/12/2022
CAD	4/11/2022	6/24/2022
Construction	9/5/2022	9/19/2022
Assembly	9/19/2022	10/3/2022
Integration	8/22/2022	9/22/2022
Regulator Control with DAC	8/22/2022	9/5/2022
Regulator Measurement with ADC	8/29/2022	9/12/2022
Final Software integration	9/5/2022	9/22/2022
Testing	9/19/2022	10/31/2022
Experiment #1	9/19/2022	9/26/2022
Experiment #2	10/10/2022	10/17/2022
Experiment #3	10/24/2022	10/31/2022
Reporting	9/26/2022	11/29/2022
Progress Report	9/26/2022	10/3/2022
In-Progress Report	10/17/2022	10/24/2022
Final Report	11/14/2022	11/29/2022
Poster	11/14/2022	11/29/2022
Demo	9/30/2022	11/11/2022
Demo #1	9/30/2022	9/30/2022
Demo #2	10/28/2022	10/28/2022
Demo #3	11/11/2022	11/11/2022

Figure 42: Gantt chart task table



Positions									
Front Panel	**PCB & Soldering**	**Switching Regulator**	**Linear Regulator**	**3.3V Static Regulator**	**5V Static Regulator**	**SCPI Controls**	**Switching Algorithm**	**LED/LCD Interface**	**Rotary Encoder & Keypad Inputs**
Tarun	Carlos	Zach	Carlos	Juan	Dean	Juan	Zach	Tarun	Dean
Zach	Tarun	Carlos	Dean	Dean	Juan	Zach	Juan	Dean	Carlos

Figure 43: General allocation of responsibilities

Aside from the tasks we initially scheduled to do in 493, we are currently behind on PCB design and MCU programming, mostly USB. We will be continuing to work on this project over the summer to not only catch up but get ahead since we expect to encounter similar speedbumps next semester. Tarun and Zach are primarily responsible for the front panel design and PCB. Carlos and Tarun are primarily responsible for the regulator circuit PCB design and soldering. Juan and Zach are primarily responsible for implementing SCPI commands as well as the functions to control the output of the switching regulator. Most of the work needed for the regulators and physical inputs is done but we need to begin integrating them.

Schedule and Milestones

Over the summer, we plan to finish integrating our individual pieces into a regulator circuit and design a PCB for it, as well as make more progress with integrating the regulator circuit with our MCU, such as by controlling the linear regulator’s output with our MCU’s DAC. If we manage to get ahead during the summer and procure PCBs, we will have more time to test them and more time to fix any problems that may arise.

During week 0 of next semester, we will aim to have our PCB fully designed if not constructed and will spend successive weeks in the semester focusing on integrating our PCB with the enclosure, we were given by Dr. Kaps, continuing to run tests on our integrated system and provide deliverables to Dr. Kaps as requested, such as Demos and quantitative results from experiments.

References

Literature References

[1]

J. Voas, N. Kshetri, and J. F. DeFranco, “Scarcity and Global Insecurity: The Semiconductor Shortage,” *IT Professional*, vol. 23, no. 5, pp. 78–82, Sep. 2021, doi: 10.1109/MITP.2021.3105248.

[2]

“Analog Discovery 2 - Digilent Reference.” https://digilent.com/reference/test-and-measurement/analog-discovery-2/start (accessed Mar. 03, 2022).

[3]

Keysight, “DC Power Supplies,” *Keysight*. https://www.keysight.com/us/en/products/dc-power-supplies.html (accessed May 10, 2022).

[4]

“DC Power Supply | Variable Power Supplies | Tektronix.” https://www.tek.com/en/products/keithley/dc-power-supplies (accessed May 10, 2022).

[5]

“DC Power Supply | Siglent SPD3303X-E DC Power Supplies,” *Siglent*. https://siglentna.com/power-supplies/ (accessed May 10, 2022).

[6]

“DC Power & Loads DC Power Supply & Electronic Load Families | RIGOL.” https://www.rigolna.com/products/dc-power-loads/ (accessed May 10, 2022).

[7]

D. by A. S. | 旭亞系統設計(股)公司 | www.akacia.com.tw, “DC Power Supplies-Gwinstek.” https://www.gwinstek.com/en-US/products/layer/DC_Power_Supply (accessed May 10, 2022).

[8]

“Multiple Channel Power Supplies_Products_SHENZHEN KORAD TECHNOLOGY CO., LTD.” https://www.koradtechnology.com/product/8/ (accessed May 10, 2022).

[9]

“High Performance Portable DC Bench Power Supply: Save Money and Free Up Bench Real Estate by Building Your Own | Analog Devices.” https://www.analog.com/en/technical-articles/high-performance-portable-dc-bench-power-supply.html (accessed May 05, 2022).

People Contacted

Dr. Jens-Peter Kaps (jkaps@gmu.edu)