

Switch Integrated Mode Power Supply

Azra Rangwala

Department of Electrical Engineering
The Cooper Union
New York, USA
azra.rangwala@cooper.edu

Ilona Lameka

Department of Electrical Engineering
The Cooper Union
New York, USA
ilon.lameka@cooper.edu

Ridwan Hussain

Department of Electrical Engineering
The Cooper Union
New York, USA
ridwan.hussain@cooper.edu

Abstract—This paper documents the design and implementation of a physical switch integrated mode power supply (SIMPS) created as a senior capstone project for the Electrical Engineering Department at the Cooper Union for the Advancement of Science and Art. The project consisted of the design of a printed circuit board (PCB), development of firmware for the dual onboard ESP32-WROOM-32 microcontrollers, and a physical enclosure to house the electrical system. The three project members worked on multiple parts of the project, including schematic design, PCB layout and routing, PCB assembly, firmware development, mechanical design, and testing. The design files can be found on the GitHub repository [1].

I. INTRODUCTION

The main purpose of a power supply is to convert power from one type to another for a load, preferably with maximal efficiency to reduce power loss through the process. Bench-top power supplies specifically convert AC power from wall outlets into controllable DC to power circuits in a laboratory setting. This is useful for many common electronics that require a DC input, such as charging a battery, and the adjustable output makes them versatile for many scenarios. Although most bench-top power supplies serve similar functions, their internal circuit topologies can differ significantly. The two most common types are linear power supplies and switch integrated mode power supplies, aka SIMPS. In industry this topology is referred to as switch mode power supply (SMPS), but this paper will refer to the topology as SIMPS. This paper focuses on SIMPS designs due to their advantages over the linear power supply alternatives.

A. Switch Integrated Mode Power Supply Topology

A switch integrated mode power supply is an electrical power converter that transforms the standard 120VAC 60Hz oscillating voltage from U.S. outlets into a range of fixed DC outputs by adjusting the duty cycle of the switching frequency. This conversion is typically performed through a full wave diode bridge rectifier (AC-DC conversion), a flyback converter (DC-DC conversion with isolation), filtering circuits, and a feedback controller.

A diode bridge rectifier functions by using diodes to only allow the positive half cycles of an AC waveform to appear at the output. Capacitors are typically placed at the output to filter the rectified signal and reduce voltage ripple. The greater the capacitance, the smoother ripple but also the longer

the discharge time. This process effectively converts the AC waveform into a near Dc voltage with minimal ripple.

A flyback converter consists of a MOSFET and a transformer to maintain isolation. The switching frequency “chops up” a steady DC voltage into pulsed waveforms for the flyback converter’s transformer to then isolate and regulate the output. This switching frequency is on the order of tens of kilohertz, reducing the size needed for the transformer, where higher frequency devices need lower reactance to be operational. The filtering circuit consists of capacitors and inductors so that the “chopped up” waveform doesn’t appear at the output, and instead, there’s a smooth DC waveform. A smooth DC waveform is defined by low voltage rippling and the lack of a strong harmonic in the output. The feedback controller ensures that the voltage is adjusted in real-time to correct for any voltage drifting in the circuit.

It is also important to note that there is isolation between the AC mains ground and the rest of the circuit to reduce the chance of ground loops and for safety. A ground loop occurs when there are multiple references to the same net with significant resistance in the paths, which can cause unwanted noise in the circuit. With AC mains ground isolated, the potential of shorts occurring decreases, which can be dangerous at the high voltages the circuit operates at.

B. Linear Power Supply Drawbacks

Unlike SIMPS, a linear power supply typically works by having an isolation transformer operate at 120VAC 60Hz. The AC signal is then rectified into DC with a diode bridge rectifier. The DC output is then sent through a BJT to act as a variable resistor via a linear feedback system. There are two main drawbacks for this topology. First is that a low frequency transformer requires a larger reactance to operate effectively, which increases the size and weight of the transformer. The second drawback is that using a BJT as a variable resistor causes significant power losses for high power devices and has lower efficiency. The reduced efficiency in the system causes the thermal temperature of the system to increase as well, which may require more complex designs for a cooling system such as larger heat sinks and fans. For these reasons a SIMPS topology is preferred over a linear power supply.

C. Project Scope

The primary goal for this capstone project was to design and build a functional SIMPS that resembles common industry bench-top supplies. The design specifications included two channels capable of delivering up to 20V and 3A each, while also having the capability to operate in series and parallel mode. Additionally, a third channel provides a fixed 5V and 3A USBC-C output to power and charge devices such as phones. Key targets included maintaining a less than 1% ripple on the output voltages and an efficiency greater than 60%, similar to the tested efficiencies of other lab bench-top power supplies. For this project, the reference switch mode power supply chosen was the Gwinstek SPD-3606, as seen in Fig. 1.

Due to the nature of the SIMPS topology and its source of power, a lot of emphasis was put on maintaining a safe working environment. This included multiple protection circuits and the development of a safe testing procedure to safeguard both the team members and the hardware.

The project was divided into the following milestones:

- 1) Design (schematic and layout) for SIMPS PCB.
- 2) Firmware development to control logic and switching behavior.
- 3) Assembly, safe testing, and performance recording.
- 4) Mechanical enclosure design to house the electrical system.

While all four milestones were completed in the Spring of 2025, not all the design specifications were achieved.

This paper is organized into the following sections: The system overview is discussed in Section II. The design and implementation is presented in Section III. The testing and results data is featured in Section IV. Project takeaways are in Section V. The conclusion and future work are in Section VI.

II. SYSTEM OVERVIEW

Fig. 2 summarizes the system overview of SIMPS. The overall electrical system began with the 120VAC 60Hz output from a wall outlet. This was connected to an EMI filter to remove any noise in the power line and ensure the input signal to the circuit will be clean. To safely mimic the AC power rectification of a typical bench-top power supply, a transformer was required to step down the AC voltage to 24VAC. This was to prevent the team members from working with the 120VAC directly, which can be dangerous due to its high voltage.

The 24VAC was connected to the PCB through an XT30 connector and routed through another set of protection circuits. This lower voltage AC input would then pass through a bridge rectifier to produce the 34VDC voltage used throughout the SIMPS design. The output of the bridge rectifier is 34VDC because 24VAC in this case refers to the V_{rms} value of the AC waveform, while the peak voltage is around 34V.

The 34VDC input first provides the LVDD 5V and 3.3V power rails, which power the onboard ESP32s microcontrollers and the input control circuitry. The 34VDC would then feed into the flyback converters present on each of the three

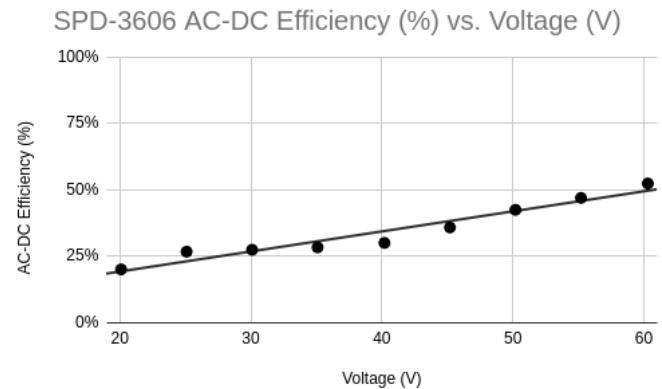


Fig. 1. Efficiency of the Gwinstek SPD-3606 across various loads.

channels. The converters would step down the DC voltage to lower values based on the duty cycle of a signal generated by the ESP32s.

For Channel A and B, a buck converter followed the flyback converter, which is used when lower output voltages are required. The buck converter was selected to handle outputs below 10VDC in order to maintain decent efficiency. Operating the flyback converter at low output levels would require a low duty cycle, leading to poor switch utilization and reduced efficiency. Instead, in the case of a low voltage output, the flyback converter would generate an intermediate 12VDC, which the buck converter steps down as needed. After the buck converter, the output would go through several sensors to monitor the current, temperature, enable, and voltage feedback. Finally, the output would be sent through an XT30 connector for Channel A and B connectors.

Channel C differs in that it does not have a buck converter. Instead, the flyback converter directly generates a fixed 5V output. This 5V output is sent through the over-voltage protection (OVP) circuit before going into the USB controller circuit that delivers a regulated final 5V. A feedback circuit is also present for Channel C for ESP32 monitoring and control.

Throughout the circuit, multiple input controls are provided, such as buttons and potentiometer knobs, along with two liquid crystal displays (LCDs). This allows the user to interact with and control the SIMPS. This includes dictating when the voltage is outputted, adjusting the output voltage, and being informed what the output voltage is.

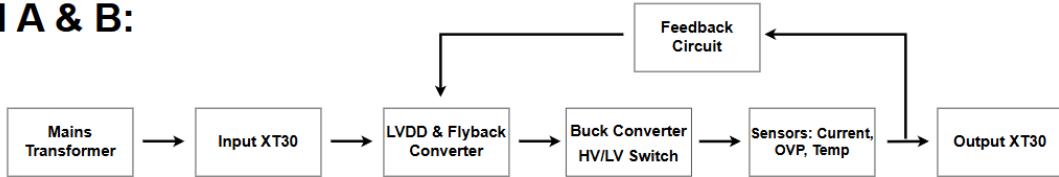
The overall system was designed to output a maximum of $135W (20V \cdot 3A + 5V \cdot 3A = 135W)$. In case the efficiency of the power supply is lower than expected, the maximum power rating was calculated with the assumption of 50% efficiency, so the PCB would be required to handle at least 270W of input power.

III. DESIGN AND IMPLEMENTATION

A. Electrical Hardware Design

This section outlines the electrical hardware design of the PCB. The schematic and PCB layout were created using

Channel A & B:



Channel C:

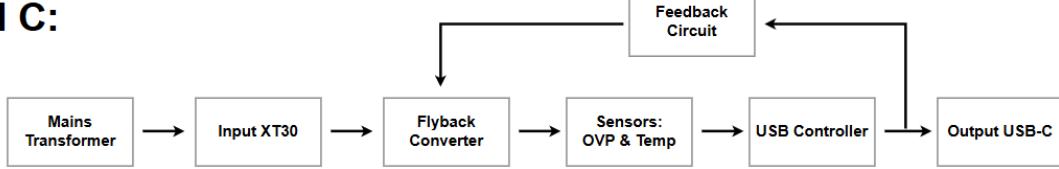


Fig. 2. System Block Diagram

Altium Designer. All design files related to this section can be found in the Altium Design Files repository [1].

The system is divided into multiple functional sub-blocks for Channels A and B: external devices, input circuit, LVDD, microcontroller unit (MCU), flyback converter, sensors, feedback circuitry, and series/parallel configuration. Channels A and B share a duplicated design, with the main differences between them relating to the parallel and series mode. Channel C, however, is much more unique since instead of a variable output, it was designed to output a fixed USB-C charging output.

The system starts with selecting external devices responsible for supplying power to the PCB. The chosen external EMI power line filter, Qualtek 850-20/004, is rated for the U.S. standard of 120VAC at 60 Hz and supports a current rating of 20A. The EMI filter is then connected to an external step-down mains transformer. Selecting the appropriate transformer required much consideration with multiple requirements. The transformer needed to handle a 120VAC input, operate at 60 Hz, and provide a significantly lower output voltage. The selected transformer model, the SolaHD E300E, met all of these specifications by providing an output voltage of 24VAC, along with having a power rating of 300VA.

In addition to the EMI filter and the mains transformer, the other external devices include rocker switch, potentiometers, buttons, LCD screens, and the output receptacles. Most of these devices were connected using JST connectors for simplicity. The user would interact with these external devices to adjust the output of the power supply. The rocker switch powered on the entire circuit while the potentiometer adjusted the output voltage and current. The multiple buttons turned on Channel A, B, and C, and also controlling series/parallel mode. The LCD screen displayed the voltage and current set by the user, in addition to any error state the user encountered.

The input circuit is the first block that exists on the PCB. The output of the rocker switch connects to an XT30 connector on the PCB, which sends the input through a protection circuit. This protection circuit includes a fuse, thermistor, and varistor. The fuse rating was selected based on the power requirements of the PCB, which was assumed to be 270W. Equation (1)

shows the calculation used to find the required fuse rating. Fuse ratings are typically rounded-up to the next available fuse value, which in this case was 12A. The selected inrush current limiter was rated for 14A and had a resistance of 4Ω . At its maximum current rating, the thermistor's resistance would fall to 0.05Ω . For transient voltage protection, a varistor rated at 60.5V with a surge current rating of 180A was selected. Together, these three components formed a simple protection circuit for SIMPS.

$$I_{rms} = \frac{P_{input}}{V_{rms}} = \frac{270W}{24V} = 11.25A \quad (1)$$

Following the protection stage, the input circuit also includes a diode bridge rectifier to convert the AC waveform to a DC waveform. There were two component options populated on the final PCB to evaluate performance differences: 60V and 20A SB2060 Schottky diodes, and a bridge rectifier IC, the GBU3010-BP. These two designs were both tested individually to determine the better option. The output of the bridge rectifier produced 34VDC, which required multiple capacitors for filtering. The schematic design specified five $200\mu F$ capacitors, however, their standard footprint allowed potential capacitor replacements from junior projects lab (JLab). Each capacitor bank also included a $20k\Omega$ bleeder resistor for safety discharge of the capacitors' stored energy, where the charge would fully discharge after a maximum time of one minute. This DC waveform is noted as "UNREG_DC" on the PCB.

The "UNREG_DC" net is what sources the power for Channel A, B, and C. One thing to note is that since the design of the power supply has the option of connecting Channel A and B in series or parallel, these two channels had to maintain isolated grounds to prevent shorts when either of these modes were active. The flyback converter has a transformer in its design which allows isolated grounds between Channel A and B intrinsically. The LVDD (5VDC and 3.3VDC) also needed isolation. To simplify the design, Channel B and C shared the same ground net. As a reminder, the primary ground is also typically kept isolated from the rest of the circuit. So in total, there are three ground planes: primary ground, channel A ground, and Channel B + C ground.

The next block was the LVDD, which maintained isolation through an isolated buck converter for 5VDC and a low dropout (LDO) regulator for 3.3VDC. The isolated buck converter had to handle 34VDC and output a fixed 5VDC. A buck converter with these specifications was hard to find, however, RECOM Power manufactured the model REC10K-2405SAW/H2 that met the specifications, with an output current rating of 2A. The 5VDC was then put through an LDO to produce the 3.3VDC rail. As mentioned before, Channel A has its own set of 5VDC and 3.3VDC, while Channel B and C share their 5VDC and 3.3VDC.

After the LVDD is the MCU, which serves as the main controller of the circuit. The specific MCU chosen was the ESP32 since there was a readily available breakout board for it, the ESP-WROOM-32, and it had sufficient pulse width modulation (PWM) and analog to digital converters (ADC) channels required for the project. There were two MCUs for this project, MCU A and MCU B. MCU A controlled Channel A, series, and parallel mode while MCU B controlled Channel B and C. These ESP32s were powered with 5VDC from their respective power domains. The ESP32 handles the PWM for the flyback converter, buck converter, reads the sensor values, and enables the output. More details on the implementation of the ESP32 is given in the “Firmware Design” section.

The next critical subsystem is the flyback converter, which serves as the primary power conversion stage for SIMPS. As described previously, the flyback converter consists of a flyback transformer, filtering capacitors, and two STP55NF06 MOSFETs, where one is placed on the primary side and the other on the secondary side. This topology is referred to as a synchronous flyback converter, in contrast to an asynchronous design which would utilize a diode on the secondary side in place of the MOSFET. Due to the high switching frequency that is on the order of tens of kilohertz, the reactance required by the flyback transformer is significantly reduced. This explains why the 60 Hz mains transformer, which is external to the PCB, weighs approximately 10 pounds, whereas the high-frequency flyback transformer is light weight enough to be mounted directly onto the PCB.

The primary side of the transformer receives an unregulated DC voltage labeled as “UNREG_DC”, while the secondary side delivers the transformer output. Although both of these waveforms are ideally DC, the transformer can transfer power because of the “chopping” actions of the MOSFETs. The MOSFETs on both the primary and secondary sides are controlled by a PWM signal, which converts the DC signal into a pseudo-AC waveform suitable for transformer operation.

It is important to note that the duty cycles on the primary and secondary MOSFETs are complementary. For example, if the primary side has a 30% duty cycle, then the secondary side operates at a duty cycle of 70%, such that one is HIGH while the other is LOW. The output voltage can be increased by adjusting the PWM duty cycle, up to a maximum of 50%.

To smooth out the output waveform to a clean DC signal, filtering capacitors were added after the flyback transformer to deal with the heavy transient voltage swings. However, due

to the high switching frequency along with the high electrical power, this stage generates the most EMI interference. The output from this stage was expected to range from 20VDC to 10VDC.

The buck converter stage comes after the flyback converter. It is activated when an output voltage less than 10VDC is required. This transition is managed by a set of relays, which are controlled by an MCU. Since relays require more current than an MCU pin can supply, the control signal from the MCU is passed through PMOS transistors with pull-down resistors so that the relays draw power from the 3.3VDC line.

The flyback converter is designed to ideally provide a stable 12VDC output, which serves as the input to the buck converter. A 12VDC input was chosen so that the buck converter doesn’t require a high efficiency design to reach the maximum voltage (maximum efficiency needed is $10\text{VDC}/12\text{VDC} = 83.44\%$). A synchronous buck converter topology was chosen for its superior efficiency compared to the asynchronous variant. This topology typically consists of two MOSFETs (two NMOS or a PMOS/NMOS pair), an inductor, and a capacitor. For this design, two NMOS transistors were used where they were controlled by complementary PWM signals.

Since the ESP32 logic level was 3.3VDC, this wasn’t sufficient to reduce the $R_{ds,on}$ of the MOSFET, so a logical level shift circuit was utilized to step up the PWM logic level from 3.3VDC to 5VDC. By minimizing the $R_{ds,on}$, there would be less loss in the circuit and lower heat. This logical level shift consists of two NMOS and two pull-up resistors. In the buck converter design, there were two options for the NMOS transistors. The first choice was the STP55NF06, which has a gate charge of 60nC. Due to concerns that the high gate capacitance could affect switching performance, an alternative, the IRLHM630TRPBF, was chosen since it had a lower gate charge of 41nC. If the relays are de-energized, the output of this stage is 10VDC-20VDC. Otherwise, if the relays are energized, the output of this stage is 1VDC-10VDC.

The next subsystem includes various sensors and protection components that include a temperature sensor, current sensor, over current protection (OCP), OVP, in-line diode, TVS diode, and a circuit enabling mechanism. The temperature sensor was implemented with the TMP708AIDBVR, which would send a signal to the MCU when the temperature threshold was reached. This temperature threshold was set to 100°C . The current sense was done using the ACS712ELCTR-05B-T, which utilizes a Hall effect current sensor. The OCP was another in-line fuse, which was rated at 3A. This fuse serves as extra protection, by ensuring each channel has its own current protection.

The OVP circuit for each channel consists of a zener diode and two PNP transistors, along with some resistors. Fig. 3 shows the protection circuit for Channel A, where a zener diode rated with a clamping voltage of 21.2V was used. When the circuit is operating below the threshold voltage, the base of the Q15 PNP is HIGH, meaning the transistor is off. This pulls down the gate of Q13 to ground, which keeps the Q13 transistor on for voltage to go through. Once the voltage

exceeds the limit, the zener diode starts conducting to ground, turning on the base of Q15. This results in Q13 being off, effectively preventing the voltage from reaching the output.

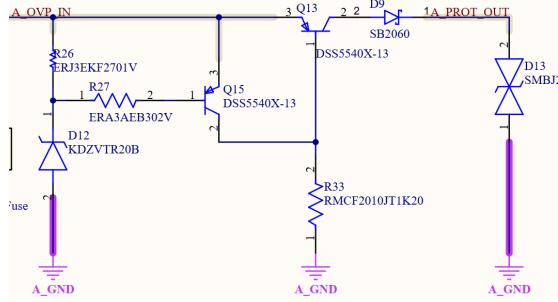


Fig. 3. Over Voltage Protection Circuit

Following the OVP stage, an in-line diode provided reverse polarity protection during series/parallel states to safeguard against voltage imbalances. There is also a TVS diode for additional OVP. The last part for this subsystem is the enable circuit, which is controlled with a NMOS that connects the output of the protection circuit to the channel output. The channel output goes from a XT30 to the output receptacles, giving the user access to the voltage and current requested.

The next block is the feedback, which senses how much voltage is being output from the protection circuit, and adjusts it based on the voltage the user wanted. The voltage is read from the output of the OVP and is sensed through a difference OpAmp which maps 0VDC-20VDC to roughly 0VDC-3.1VDC (as per ESP32 ADC reading). Based on the voltage set by the user, and what the voltage sensed from the feedback is, the ESP32 adjusts the PWM for the flyback converter or the buck converter accordingly. Since the buck converter ideally has a fixed 12VDC input from the flyback converter, only the buck converter PWM is adjusted when the power supply is outputting less than 10VDC.

For the ESP32 to adjust the PWM for the flyback converter, a feedback transformer was needed. This is because the ESP32 PWM exists on a channel's ground, while the primary side of the flyback converter exists on the primary ground. Before the flyback converter PWM was sent through the feedback transformer, and a logical level shift circuit was used to reduce the $R_{ds,on}$ of the switching MOSFETs, maximize efficiency and reduce the heat loss for the MOSFET.

The final block for Channel A and B is series and parallel mode. To connect a power supply in series, the ground of one side must be connected to the power output of the other side. To connect a power supply in parallel, both power and ground nets of each side must be connected. As mentioned before, in order to prevent any shorts when utilizing each of these modes, the Channel A and B grounds were kept isolated. When the channels are connected in series or parallel mode, MCU A acts as the primary, and MCU B acts as the secondary. This means that Channel B behaves exactly the same as it does in independent mode, while Channel A is the one controlling the voltage and current for Channel B,

sensing the voltage feedback from Channel B, and dictating when to turn on the output. This requires additional logic to allow Channel A to decide how to connect the outputs, when to enable the output, and when to disable the output. This had to be implemented in hardware instead of allowing the MCU directly communicate with each other due to the isolated grounds. A wireless communication (such as Bluetooth or WiFi) was possible but avoided due to its added complexity in needing to take care of the RF signal and the additional resources the MCU would consume.

When either series or parallel mode is active, Channel A can send its voltage and current values through an opto-isolator. A relay controls whether Channel B reads its voltage and current values from a potentiometer or from Channel A. Channel A can also sense the voltage output from Channel B. This is done through an opto-isolator, and ensures the Channel A and Channel B voltages are both within a good tolerance before connecting the outputs together. If there is a fault, Channel A can detect the fault by seeing what the voltage feedback it is receiving from Channel B, and disable the circuit. If Channel A can't lock its voltage within tolerance in time, Channel A can send a fault signal to Channel B by driving a relay that connects to the fault input for Channel B, which will disable the voltage for Channel B as well.

Channel B determines if it's enabled or not from two sets of relays. The first relay determines if the relay is reading from its button, or from the second relay. The second relay is powered on from the Channel A button so Channel B knows when to be enabled or not based on Channel A button. This effectively makes button B useless. In order to make sure the button lighting up matches what happens internally for the user, the Channel B button has an additional NMOS that needs to be driven high to be turned on. This NMOS gate logic is taken from the output of the two sets of relay.

When Channel A is configured for parallel operation, a relay connects the power and ground nets of both Channel A. This connection allows the output currents to combine, creating a parallel connection. As mentioned before, in the case of a voltage imbalance when the power supply is connected in parallel, there are diodes on the OVP circuit to prevent reverse current flow. When Channel A is set to series mode, there's a relay that connects together Channel A's ground and Channel B's power nets. These connections allow the voltages from both outputs to add up. Since these relays are driven from the MCU and not the buttons directly, there is a check in the code to handle the event when the user enables both series and parallel mode. This concludes the block summary for Channel A and B.

Channel C had much fewer blocks and a simpler design. The input circuit for Channel C is the same, the only external device for Channel C is the button (since it has a fixed output and current rating), and as mentioned before, MCU B is used for both Channel B and C. Channel C's main differences arose from the flyback converter, the sensors, and the USB controller circuit.

For the flyback converter, a transformer with a turns ratio

less than 1:1 was selected so that the 34VDC can be stepped down to 5VDC without requiring a low duty cycle for the PWM signal. The 5VDC was generated from the flyback converter instead of the isolated buck converter due to concerns about the isolated buck converter's output accuracy. In addition, by having control over the output voltage, if the voltage needs to be slightly higher due to voltage losses across the circuit, the flyback converter can increase the PWM while the isolated buck converter output is fixed.

The feedback circuit used was kept the same as Channel A and B, with logical level shifting and sending PWM signals to the MOSFETs. The mapping for the voltage sense and PWM output was different, but the functionality was the same. Channel C had only an in-line fuse and the OVP circuit. Current sense was omitted since Channel C current output is not adjustable by the user, and is instead determined by the USB controller circuit. The feedback for Channel C was done on the output of the OVP circuit.

The input for the USB controller circuit came from the output of the OVP circuit. The USB controller circuit was a series of Texas Instrument ICs that can handle USB 2.0 charging. This is done by using the TPS25810RVCR, which generates a stable 5VDC for the USB voltage bus needed to charge a device from the 5VDC output from the OVP circuit. The TPS25810RVCR also internally sets the CC1 and CC2 pins to appropriate pull-down resistors (in this case $4.7k\Omega$) once the USB voltage bus is stable to indicate to the device that the charger is ready to charge the device. The IC was used to generate the D+ and D- pins needed for USB 2.0 communication, which isn't required but helps with the voltage negotiation between the charger and the device. From the PCB, a USB-C port was connected to the USB controller circuit, and this would be connected to the exterior box of the power supply through a USB-C extension cable.

For the final layout, the board dimensions were 9 inches by 8 inches. From Altium Designer, the final board was a 4-layer PCB with the stackup shown in Fig. 4. This stackup was chosen as per the recommendation of JLCPCB (the fabrication house chosen for the PCB) for fast turnaround 4-layer power boards. The PCB had a total of 352 components, 753 connections, 289 vias, and 89 polygon pours, and copper utilization of 87%, 91%, 92%, and 92% for the top, inner layer 1, inner layer 2, and bottom layer, respectively. Several big components required a large board size, including the flyback transformer, isolated buck converter, MCU header pins, filtering capacitors, inductors for the buck converter, and the relays. Almost all the components were placed on the top side since this board for easier probing during the testing phase.

The first difficulty for the layout was to place all the components optimally to ensure that the path of power was direct and in a single direction. The input circuit and Channel A were easier in this regard because they were on the edge of the board so the power path naturally flowed from one side to another. Channel B was the most difficult since it was in the middle and it was difficult to make sure the power path didn't

#	Name	Material	Type	Weight	Thickness	Dk	Df
	Top Overlay		Overlay				
1	Top Solder	Solder Resist	Solder Mask		0.272mil	3.8	
1	Top Layer	PP-006	Signal	1oz	1.378mil		
2	Layer 1	CF-004	Prepreg		8.284mil	4.4	0.02
2	Dielectric 1	FR-4	Core		41.929mil	4.6	
3	Layer 2	CF-004	Signal	1/2oz	0.598mil		
3	Dielectric 3	PP-006	Prepreg		8.284mil	4.4	0.02
4	Bottom Layer	PP-006	Signal	1oz	1.378mil		
	Bottom Solder	Solder Resist	Solder Mask		0.272mil	3.8	
	Bottom Overlay		Overlay				

Fig. 4. Altium PCB Stackup.

zigzag around the board while still making sure the board was space efficient.

Another key challenge was ensuring that all the power traces had the minimum thickness needed to handle 3A, since the input circuit needed to handle 12A. This was mainly achieved through polygon pours, which is why there were so many planes. In addition to the power traces, most of the ESP32 traces were placed in the internal layer because that made routing them easier, left more room for the power traces in the external layers, and reduced the amount of interferences those traces would experience from external sources.

The last big hurdle was managing the layout for the project with the sheer number of components used. This was addressed through careful planning ahead on what section would go where, importing components block by block, and routing them. To place the components block-by-block in Altium Designer, the "compile mask" feature was utilized to temporarily hide parts of the schematic so that they wouldn't be imported onto the layout.

The general layout procedure began by assigning dedicated board areas to each major circuit block. High current traces were visually distinguished by color coding to prevent errors such as using narrow widths for high current paths. The input circuit, in particular, was designed to make sure 12A could be safely carried through the PCB. After this, each block was imported one by one.

Fig. 5 shows what the signal layout looks like without polygon pours. Fig. 6 shows what the 3D preview looks like from Altium Designer.

B. Firmware Design

The firmware was developed for two ESP-WROOM-32 microcontroller units (MCUs), which controlled the three output channels. MCU A managed Channel A, while MCU B contained firmware that controlled Channels B and C. All of the code for the firmware can be found on GitHub [1].

Each ESP32 MCU contained two cores, enabling dual-core multitasking where each core can run a separate task simultaneously. For MCU A, three tasks were created, with two running concurrently at any given time. The first task was designated as the primary task, which operated continuously while the MCU was powered. This task handled ADC readings from potentiometers that would control the current and voltage levels, as well as monitoring for any fault conditions.

The secondary task handled input and output logic, including controlling the PWM signal, processing button inputs, and

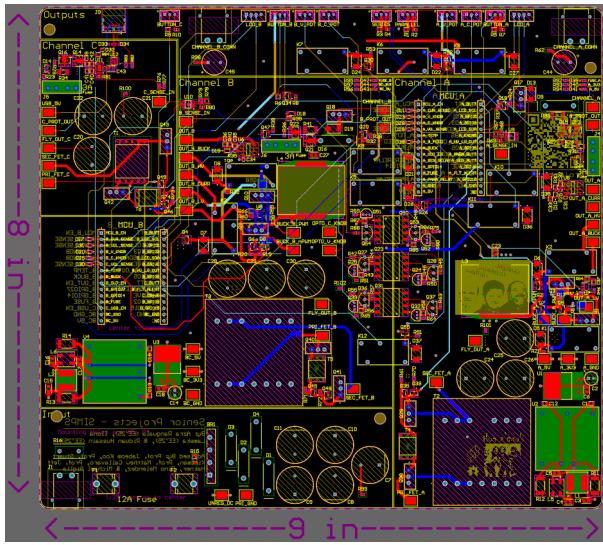


Fig. 5. Altium Designer Layout Signal Layer



Fig. 6. Altium Designer 3D Layout Preview (top layer).

managing I₂C communication with the display. This secondary task was active during normal operation, when no faults were detected by the primary task. Overall, the primary task would live in its own core, while the other core would host a specific secondary task that depended on the fault state of the logic.

If a fault was triggered, the primary task would detect it and terminate the already running secondary task. This new secondary task disabled all of the output logic, preventing the flyback and buck converter from being driven by a PWM signal. The new task also stopped reading any input pins, preventing the user from leaving the fault state of the MCU through any external controls. The firmware was intentionally designed to only leave the fault state through a full power cycle, adding an extra layer of safety in the case of a hardware fault.

Due to the unstable nature of the ADC signals received from the potentiometers and feedback circuits, the primary task averaged multiple samples to obtain more stable readings. Additional mathematical operations were performed to convert the raw ADC integer values into real voltage or current measurements.

Multiple feedback functions were also created for the flyback and buck converter. Since the converters were designed to work under a variety of loads, it was essential for the MCU to adjust the duty cycle of its PWM signals depending on the real-time feedback. These feedback functions relied on a duty cycle-to-voltage/current mapping, which was completed during the testing phase of the project.

While the MCU A code was completed, the MCU B code development was paused until all of the mapping data for MCU A was finalized. Additionally, MCU B required extra tasks for Channel C, so a decision was made to test Channel C first before proceeding with finalizing the firmware.

Multiple test scripts were also written to test each channel, which would be written based on the relevant test being ran. Each test script would control various pins and produce different outputs, with the goal of collecting data to then further the development of the final firmware for both MCU A and B.

C. Mechanical Hardware Design

The mechanical design for the project involved creating a protective and accessible enclosure for the PCB and its external connections. The enclosure was constructed using a combination of materials: a medium-density fiberboard (MDF) base and 1/4" acrylic sheets for the walls and cover.

The MDF base served as the foundation and included rubber bumpers nailed to its underside, allowing for easier handling when lifting the enclosure. Additionally, the base featured four screw holes precisely aligned with the mounting holes on the PCB. The PCB was secured above the base using 1-inch spacers, providing clearance and mechanical stability.

The acrylic walls had cutouts to accommodate external PCB connections. One of the side walls included an opening for the input AC mains wiring to pass through and connect to the internal circuitry. The front wall was designed with cutouts for all user interface components, including buttons, potentiometers, LCDs, rocker switch, and a USB connector. For better aesthetics, knobs were installed on the potentiometers. The buttons used were of the latching type and featured internal diodes, enabling them to provide a visual indication of whether the button was in a HIGH or LOW state.

For added safety, the enclosure included a hinged acrylic cover, providing full protection between the user and the PCB while allowing easy access when needed.

The acrylic walls were designed in OnShape, and the design files can be found on GitHub [1]. The dimensions of the final box was roughly 12" x 12" x 6". Fig. 7 and Fig. 8 display the front and back walls of the box, respectively.

Throughout the design, there were no additional heat sinks for any of the MOSFETs powering the circuit. Based on an

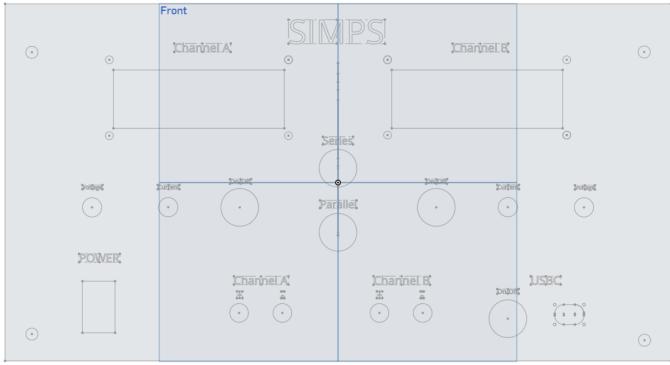


Fig. 7. OnShape Front Wall 3D CAD.

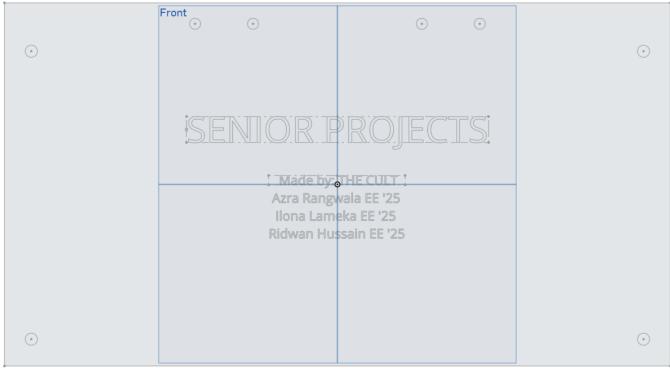


Fig. 8. OnShape Back Wall 3D CAD.

online calculator that determines the heat dissipation needed for a high-power MOSFET, a heat sink was not necessary [3]. This is likely due to the switching voltages between ON/OFF were at 34V, instead of at roughly 120V.

IV. TESTING AND RESULTS

Due to the high-power nature of the project, extra precautions were taken to maintain a safe working environment. High-voltage gloves were always worn during interactions with a live circuit, and remained on until the capacitor banks were fully discharged. Although discharge times were recorded during testing, an additional safety measure was taken by enforcing a one-minute cool-down period before handling the PCB without gloves. Additionally, the rocker switch used to power the circuit was always held by a team member, allowing immediate shutdown when necessary. This rocker switch was connected to the low side of the 60Hz transformer to make sure the operator would have 24VAC in their hands instead of 120VAC. To minimize exposure to high-voltage contacts, the transformer connectors were covered with electrical tape (rated for 600V), along with any other exposed metal contact that carried the mains voltage (such as the EMI filter).

The tests were conducted by following the Testing Procedures document, which would dictate how to conduct the tests in ideal conditions, where the Testing Results would record the results of each test and explain any difficulty or

eworks that needed to be done to get through a particular test. Both of these documents can be found on the GitHub [1]. All oscilloscope readings were done on the Keysight MSO-X 2012A oscilloscope, where the data would be recorded, run stopped/triggered, and the screenshot would be saved on a USB drive. This was to ensure the highest quality data would be recorded and could be viewed later. The probes were almost always set to 100x for all measurements to have consistency.

It is worth noting that the oscilloscope reading seemed to be lower than the true reading after verifying this with a benchtop PSU from JLab and checking the voltage it displayed with a DMM and the oscilloscope. Although this was the case, this oscilloscope model was the only one available, and an oscilloscope is required to measure and view the waveforms, so it was noted that the oscilloscope readings were likely lower than they should be.

The first test involved determining the better bridge rectifier configuration for the circuit. It was determined that the individual schottky diodes (SB2060) resulted in a higher peak-to-peak output voltage of 34.2V, while the bridge rectifier IC (GBU3010-BP) produced 33.6V. After desoldering the bridge rectifier IC, filtering capacitors were added to achieve a stable DC output. Three 200 μ F capacitors yielded a voltage of 33.8V with a ripple of less than 0.5% as shown in Fig. 9.

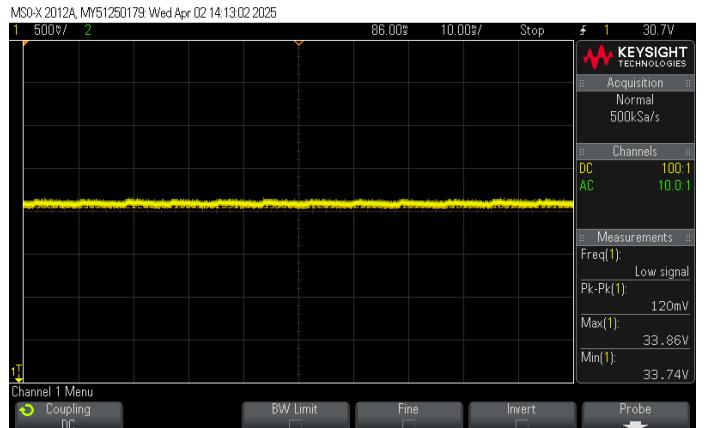


Fig. 9. Input DC Ripple.

After getting the fixed 34VDC, testing pivoted to Channel A. Testing was done by building up individual circuit blocks of Channel A and testing each one. The first circuit tested on Channel A was the LVDD rails. Both the 5V and 3.3V rails had a constant output voltage, which marked them as stable rails for the rest of the circuit.

The next couple of tests came in rapid succession, with the ESP32 providing a good PWM output signal that would later be used in the flyback and buck converter. External connections, such as buttons and LCD screen, were able to function normally. The temperature sensor circuit received a resistor of 51k Ω to set a trigger temperature value of 52°C. The circuit managed to successfully enter a fault state when a hot air gun was pointed at the temperature sensor.

The most intensive testing that was done involved the Channel A flyback converter due to several issues occurring at once. The first issue that was noticed was the failure of the feedback transformer. Intended to provide the PWM signal for the primary side of the synchronous flyback transformer, the feedback transformer never provided a HIGH signal for the primary side MOSFET while the PWM was running. Fig. 10 shows the isolation transformer, where its right side was pulled-up to 5VDC through a resistor.

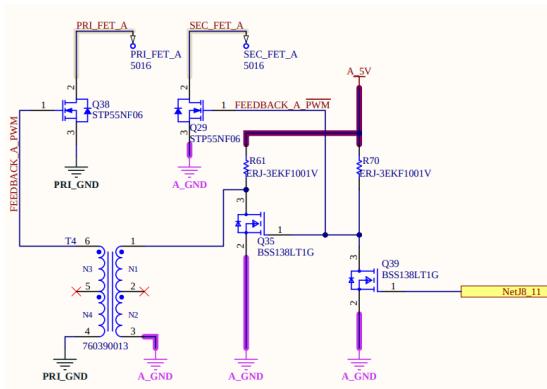


Fig. 10. Feedback Drive Circuit

However, the internal resistance of the coil of the feedback transformer was found to be 0.25Ω , acting like a pull-down resistor to ground, which prevented the pin from ever being pulled up to 5V; instead, it would act as a voltage division that would almost always be near ground. This issue was resolved by obtaining an isolated gate driver, the UCC5304DWV, which accepted the MCU's PWM on the channel side ground and outputted it to the gate of Q38 on the primary side.

However, there were still issues with having an output of the flyback converter. Upon further inspection of the block, it was seen that the dot polarity of the flyback transformer was switched accidentally, which resulted in multiple blown fuses and damaged MOSFETs. To fix the wrongly connected pins, the transformer's pins were swapped by soldering jumper wires between the component and the PCB. However, this unstable wiring created a large amount of "buzzing", which can be described as having an audible humming sound, along with high ripples present on the output voltage. This "buzzing" was caused by magnetostriction, where the transformer's core is internally expanding and contracting, causing the audible buzzing noise, and it is mainly caused by the transformer not being mounted properly (since jumper wire rework had to be done on the transformer). This rework proved to cause many issues for the circuit during testing, and this buzzing will be referred to several times throughout the test procedure.

To reduce the ringing found on the voltage lines, RC snubbers were added to the primary and secondary MOSFETs. By using the frequency of the ringing, the primary transistor had a resistor of 22Ω and a capacitor of 680pF soldered between the drain and source pins. The secondary transistor

received a 15Ω resistor and a 680pF capacitor. The link for the snubber circuit calculator is provided in the references [4].

The pull-up resistor of R70 for the PWM drive circuit was also reduced from $1k\Omega$ to 150Ω , since the switching frequency was increased, so the larger resistance caused too much delay in the rise time ($\tau = RC$) for the PWM signal. This resistor was also a through-hole resistor instead of a surface mount resistor, so it has a higher power rating.

The last roadblock that prevented the flyback converter from outputting a voltage was that the flyback converter would need a soft start before being stepped up to the desired voltage. This was found through experimentation, where if the duty cycle was set low, the flyback converter would output a DC voltage. However, if the duty cycle was set too high, the fuse would pop, and the primary MOSFET would need to be replaced (even with the snubber circuit). To solve this, the flyback converter's PWM would start at 102 (out of 1024), and be stepped up to the desired output voltage.

After having working output from the flyback converter, there was high voltage ripple on the output, which led to the capacitor values being tweaked. By the end of the testing, the capacitor values used were 10pF, 100pF, 47nF, 1uF, 470uF, 1mF, and 2.2mF. The capacitor values chosen were diverse so that they could target rippling of different frequencies (as seen from the FFT output). The capacitors used were only ceramic capacitors and electrolytic capacitors. The output voltage ripple for the flyback converter was still significant (roughly 3V ripple at an output of 12V), which can be seen in Fig. 11. The PWM was then swept across from 102 to 412 bits, incremented by 10 bits per 100ms at 42kHz in Fig. 12. This displays the high voltage range that Channel A could output, with the maximum voltage being 18V with constant 2V ripple (5.4V ripple including transient spikes).

The next block to test was the HV/LV switch and buck converter. The PWM sweep displays that the flyback converter is capable of having voltages lower than 10V, however, a buck converter was opted instead so the flyback converter wouldn't suffer from low switch utilization. As a note, one procedure used here was to first inject a voltage value from external power supplies onto a spare board, testing the circuit output there, and then after verifying the circuit works, build it on the main board. This was to test the future blocks in isolation since the flyback converter had issues to get accurate measurement results, and if anything went wrong with the board during testing, the reassembly process would be less intensive. Unless noted otherwise, the results of all of these tests refer to the main board.

The first test was to ensure that the HV switches (relays) work as expected, where the flyback converter output would show up if desired. Otherwise, the LV switch would activate, and the output at the end would be from the buck converter. This test worked. The initial buck converter output ripple was 1.1V with an average voltage of 5.692V. After increasing the switching frequency to 35kHz (from 10kHz) of the buck converter and adding more capacitors to the buck converter output (the final capacitors used were 10pF, 10uF, and 47uF),

MSO-X2012A, MY51250182 Fri Apr 26 21:57:05 2025



Fig. 11. 12V Flyback Converter Output on Channel A (yellow).

MSO-X2012A, MY51250182 Tue Apr 15 23:31:03 2025

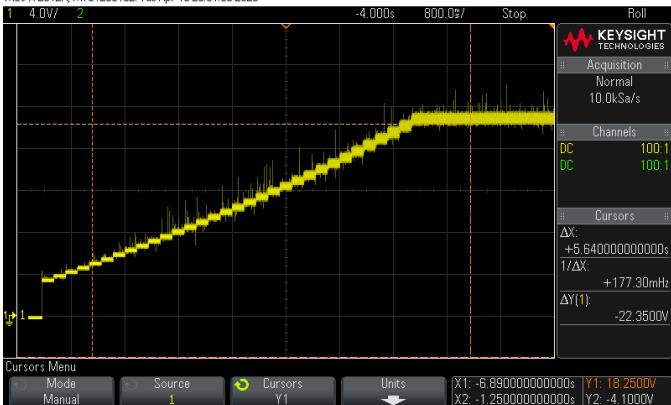


Fig. 12. Flyback Converter Output Sweep at 10bits/100ms (42kHz).

the ripple of the buck converter was lowered to 600mV. This rippling can be seen in Fig. 13.

There was also a test to sweep the buck PWM from a low duty cycle value to a high duty cycle value to map the duty cycles to the output voltages. However, all the screenshots of the output waveform include the buzzing and high-voltage

MSO-X2012A, MY51250182 Fri Apr 25 22:15:49 2025



Fig. 13. 5.6V Flyback Converter Output on Channel A (green).

MSO-X2012A, MY51250182 Tue Apr 22 17:45:36 2025

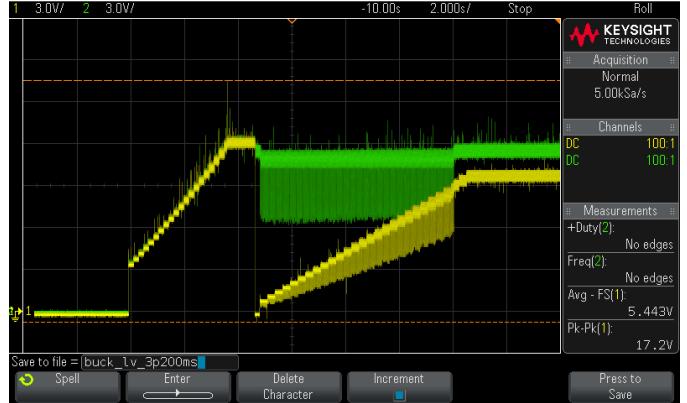


Fig. 14. Buck Converter Sweep Channel A (green), Flyback Converter Buzzing (yellow).

MSO-X2012A, MY51250182 Mon Apr 21 18:12:13 2025



Fig. 15. Buck converter voltage ripple when injected 12V from the lab power supply for the dummy board.

rippling that would appear when the buck converter is immediately switched on. This buzzing was verified to be coming from the flyback converter, however, there was no clear way to remove the massive voltage ripple other than waiting out the circuit. There wasn't enough time to revisit this problem and see if it could be solved. Fig. 14 displays the output waveform of the buck converter (yellow) being swept from 100 to 910, incremented by 10 bits per 100ms at 35kHz. The buzzing occurring immediately after the switch can also be seen in the output of the flyback converter (green).

To display the difference between the ripple of the buck converter when powered from the flyback converter and when powered from a power supply, Fig. 15 displays the ripple of the buck converter with the same input parameters (duty cycle and switching frequency) as the one seen in Fig. 13 on the dummy board.

The next block tested was current sense, where the ACS712ELCTR-05B-T would give an ADC reading based on the current reading it got from the internal Hall effect sensor. The load connected to the output was a 20Ω 10W power resistor to increase the range of current that could be tested.

In order to map current values to the ADC readings from the current sense, the fuse on the circuit was replaced with an ammeter to match the ADC value with the current values. This value would constantly fluctuate, so a sample of 300 values was averaged and displayed. This was a compromise to get fast-changing current values without the screen flickering too fast. This block works fine, however, the range of the ADC was much lower than expected as seen in Fig. 16. To correct this, a subtractor OpAmp circuit can be used to adjust the ADC range to remap the values of 0A to 0.7A from 2023 to 2129 to closer to the range of the ESP32 ADC range (roughly 0VDC-3.1VDC and 0-4095 bit values). The mapping done on the dummy board was decided to be the true values, so the mapping on the main board, with its large voltage ripples, wouldn't need to be recorded.

Afterwards, the OVP circuit was tested on the dummy board to ensure that the OVP works without risking damaging the main board. As a note, by this point in the testing, it was likely that series and parallel mode testing would not be happening, so the schottky diode used to prevent reverse current flow was shorted with a wire. This was to reduce the overall voltage drop across the circuit. The OVP voltage target was 22V, and it was shown that after the injected signal reached or exceeded 22.05V, the OVP circuit would be activated, and there would be no output.

Following the OVP circuit, the Channel A enable block was the next block to test. It was found here that simply using an NMOS as a power switch, as seen in Fig. 19, to enable or disable the output would not work. The reason is that the gate voltage would roughly be 5V, but the source voltage is not ground in this case. Thus, $V_{gs} < V_{th}$, even if the gate voltage is at 5V, so the NMOS would never switch on. To resolve this issue, the NMOS was replaced with a relay circuit, and the channel enable signal would drive the relay to connect the channel A net or disconnect. The future works section mentions a way to implement an enable circuit using PMOS and NMOS instead of relay circuits, but for our testing, soldering on a relay was easier.

The last block for Channel A was the feedback block which would map the 0VDC-20VDC output from Channel A into 0VDC-3.1VDC for the ESP32 ADC to determine if the output being set was the expected value and to adjust the duty cycle of the flyback converter/buck converter otherwise. As with the current sense, it was decided that the mapping of the feedback voltage from the dummy board would be taken as the true value, so the mapping on the main board with the voltage ripples wouldn't need to be recorded. The result of this test was very clean, with the voltage being mapped from 1V to 9.5V, having an $R^2 = 1$. The line of best fit plot can be seen in Fig. 18.

From here, before continuing on and finishing the testing for Channel A, the testing for Channel B began. Since Channel A was mostly finished, it was believed that the testing for Channel B should go much smoother, since Channel B is a copy of Channel A and most of the issues revolving around Channel A should have been encountered and solved. Thus,

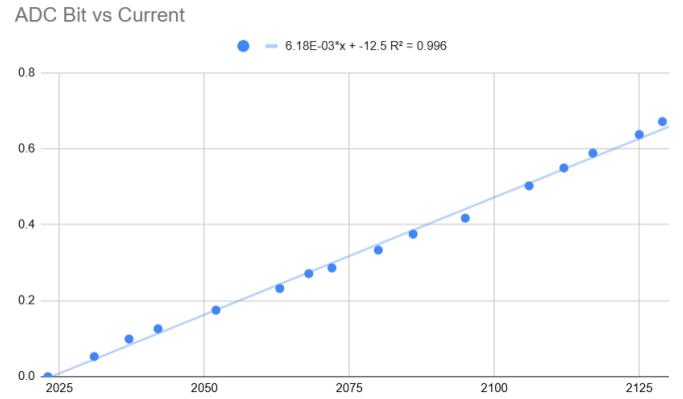


Fig. 16. Current Bits vs Current draw with line of best fit.

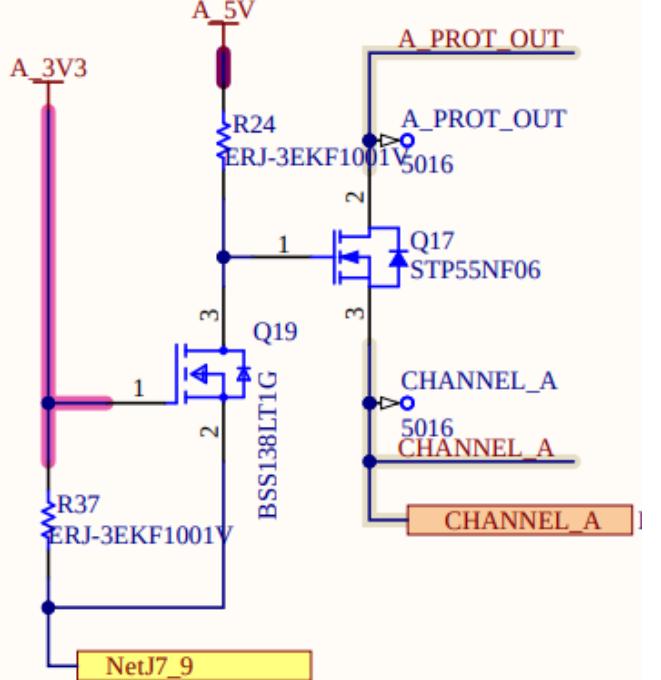


Fig. 17. Enable circuit

the testing for Channel B was assumed to be roughly the same as the testing for Channel A. This ended up being false.

The first few parts of Channel B testing which included LVDD, ESP32, external connections, and temperature sensor went smoothly. Afterwards, the testing for the flyback converter on Channel B began. The dot polarity for this transformer was also flipped on the schematic, so jumper wires were soldered on. At the beginning, there were issues with the LCD screen and the state MCU B would be in. The LCD screen would appear to flicker between the output enabled and disabled state, the output for the flyback converter would be 0VDC, and there would be audible buzzing from the transformer. This was originally assumed to be some sort of

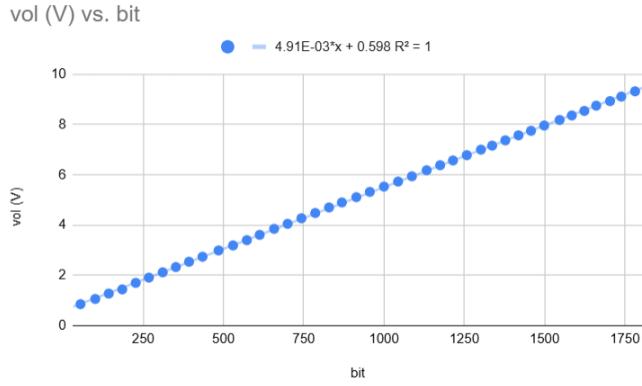


Fig. 18. Feedback Bit vs. Voltage (V) graph with line of best fit.

coupling between I2C (the communication protocol for the LCD screen) and the PWM drive because they are right next to each other on the ESP32-WROOM-32, but this was ruled out after further testing.

The cause of this issue was likely magnetostriction. By this point, the issue of magnetostriction became more and more prominent throughout the design and seemed to only be getting worse with no clear solution on how to correct it for the current spin of the PCB. At this point was determined that the transformers would buzz audibly, which was attributed to magnetostriction, and when the transformer buzzed, the voltage ripple at the output would be very unstable. The reason for the magnetostriction is believed to originate from the soldering rework that needed to be done (to correct the dot polarity of the transformer), and this rework either caused the transformer to be physically unstable so it would vibrate, the EMI interference for the rest of the board for the surrounding components would increase significantly because of the increased flux leakage from the component being tilted in the air, a combination of the two, or something else. It would also appear that as more components were added to the PCB, the transformers would become much more susceptible to start buzzing.

At this point, the flyback converter was able to output 3VDC as seen in Fig. 19. Once the PWM sweep test started, there was no settle time for Channel B and the voltage would always ripple no matter how long the circuit stayed on for (it was left at a fixed PWM for 2 minutes at a certain point), which can be seen in Fig. 20 and Fig. 21. It was seen that the rippling would propagate backward to UNREG_DC, the net used to generate the DC output voltage from the diode bridge rectifier.

From here, Channel A was revisited to finish the design through and get the fully integrated block working with a single test script, however, the magnetostriction was audibly much worst and while waiting to see if the voltage would settle or not, something within the circuit "popped" and the output for the flyback converter on Channel A could not be generated anymore. It is likely that either the transformer or the primary MOSFET popped and needed to be replaced; however,

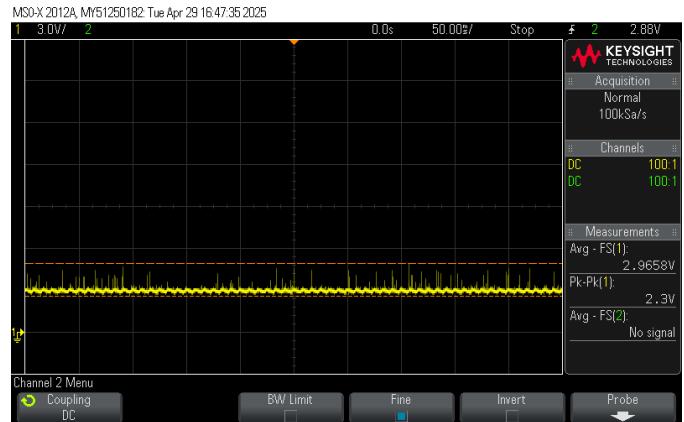


Fig. 19. 3VDC output for Channel B flyback converter.

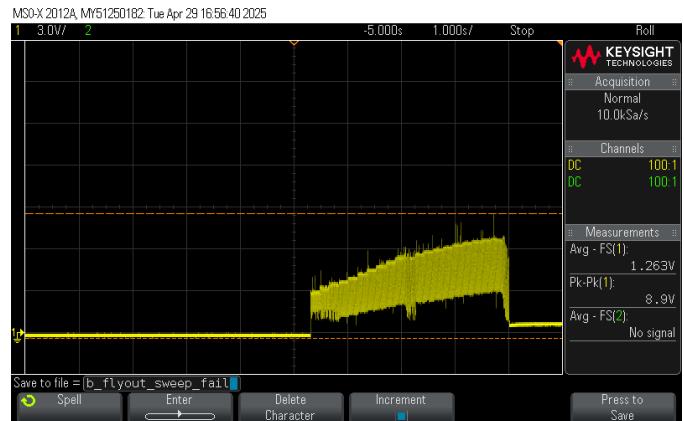


Fig. 20. Channel B flyback converter PWM sweep with constant ripple.

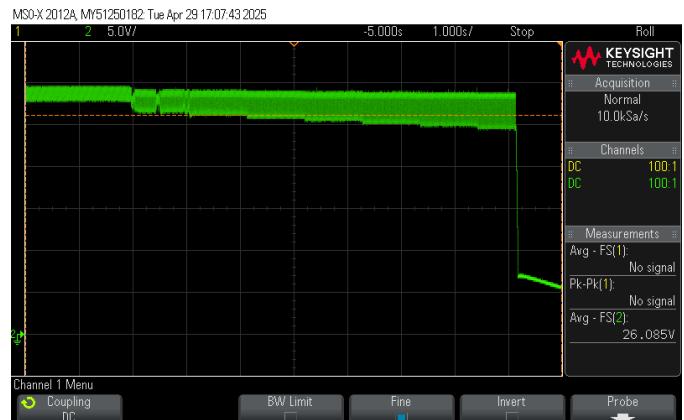


Fig. 21. Channel B flyback converter ripple with fixed PWM by constant ripple until circuit was shut off.



Fig. 22. Channel C charging for the Samsung Galaxy Tab A SM-T350 at 4.74V, 0.97A, 4.6W with Apple 2.4A protocol.

replacing this MOSFET would not solve the fundamental issue of magnetostriction. It was determined that due to a lack of testing time and low supplies, the testing effort would be shifted to Channel C instead.

Testing for Channel C was expedited due to limited time remaining in the development cycle. The flyback transformer seemed to suffer from magnetostriction as well, prompting the decision to bypass the flyback converter. Instead, a regulated 5V signal was injected directly into the “BC_5V” and the output of the flyback converter for Channel C. The OVP circuit and enable circuit works as intended for Channel C.

To evaluate performance, a DROK USB load tester, a Power-Z KM003C USB-C oscilloscope, a Samsung Galaxy Tab A (SM-T350), and a Google Pixel 9A were used. The output of Channel C is delivered through a USB-C port. A USB-C cable connected this port to the USB-C oscilloscope, which allowed for real-time measurement and data logging. The first test output was directed to the USB load tester.

The USB Load Tester verified that the output of Channel C could supply up to 3A on demand, with a peak output power of 12.6W. The voltage on the USB line ended up dropping to 4.2V from the 5.0V, but the output was stable nonetheless. No photos were taken of this event.

The USB load tester was then replaced with the tablet to verify that Channel C can indeed charge devices. The tablet uses a micro USB charging port, so a USB-C to micro USB adapter was utilized. Fig. 22 displays that the tablet was charging at 4.74V, 0.97A, for a total of 4.6W. The USB-C output was then tested on the pixel to make sure the output was reproducible. Fig. 23 displays that the pixel was charging at 4.69V, 1.06A for a total of 5.0W. The two figures also display that Channel C was able to handle different charging protocols.

Thus, this concludes the testing for SIMPS. Table 1 summarizes the results of testing for each block on whether or not they passed/met the minimum requirements on the first try.

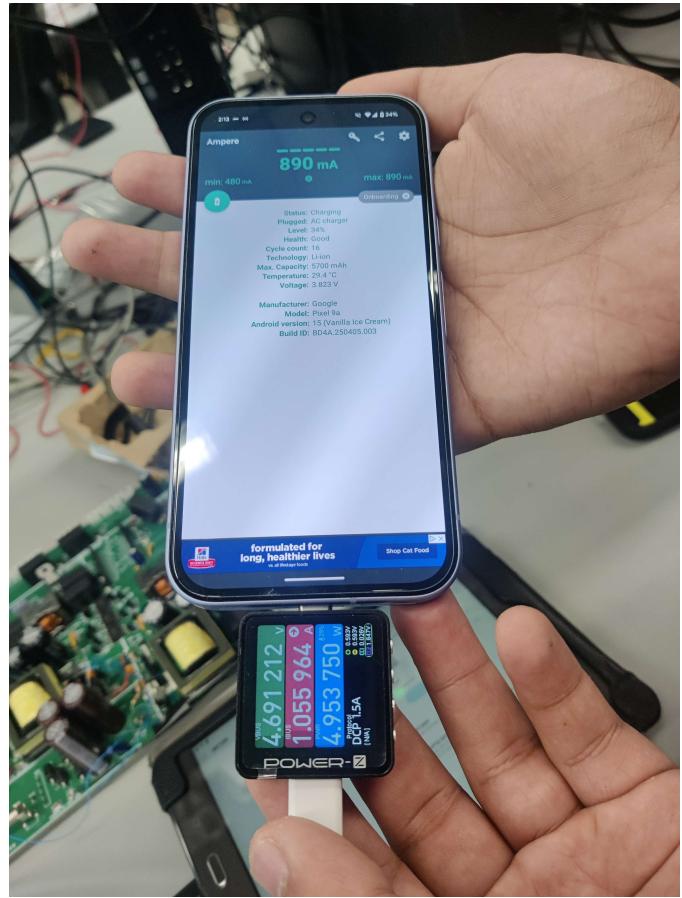


Fig. 23. Channel C charging for the Google Pixel 9A at 4.69V, 1.06A, 5.0W with the DCP 1.5A protocol.

TABLE I
SIMPS BLOCKS PASSED

Input Circuit	Passed
Flyback Converter	Failed
Buck Converter	Passed
Current Sense	Passed
OVP	Passed
Enable	Failed
Temperature	Passed
Feedback	Passed
Channel A	Passed
Channel B	Failed
Channel C	Passed
Parallel Mode	Failed
Series Mode	Failed

V. PROJECT TAKEAWAYS

The major challenges encountered during the testing phase could likely have been mitigated with additional time and resources for a second PCB spin. Since the PCB was ordered during the final semester, only about one month remained for testing each circuit block. Ideally, a shorter planning phase would have allowed the schematic design to be started earlier, allowing more time for testing.

Furthermore, several major issues could have been identified earlier if multiple blocks had been simulated together in LTSpice. Since most of the blocks were built independently, there was limited data on how the blocks would interact when integrated on the final physical PCB. One such block was the logical level shifter being fed into the feedback transformer.

As always, additional design reviews would have helped catch several of the significant issues that arose during the testing phase. This especially applies to the incorrect dot polarity on the flyback transformers, which led to a magnetostriction effect. As mentioned previously, magnetostriction describes the transformer's source of audible buzzing, caused by a fluctuating magnetic field inducing contraction and expansion in the transformer's core. Due to the unstable connection of the transformer with the jumper wires, the magnetostriction vibrations were felt throughout the circuit, hurting the stability of the power rails and the signal integrity.

Despite the project falling short of being successful, all the members were able to learn a lot about power electronics and large system design throughout this project. There were countless hours spent researching the project and going into rabbit holes to discover and design solutions for the many problems that arose from the problem, even if the problems weren't actually encountered. For example, the thought experiment for the series and parallel mode testing and the different considerations needed to make it possible in the real world, between the isolated grounds and how to have the MCUs communicate with each other indirectly, was one worth spending time on.

VI. CONCLUSION AND FUTURE WORK

The total cost of the project was roughly \$1,000. The cost of a single board, assuming no faulty components and ignoring the cost of the additional components soldered on, is approximately **\$484.79**, which can be seen in Table II. As a note, the price for the components is inflated because ordering the components in larger quantities would decrease the unit price of each component. As a reference, many two-channel power supplies with similar output power cost approximately \$200 on Amazon. The Gwinsteek SPD-3606, in particular, costs approximately \$735 (this is due to its higher power output at 60VDC and 3A or 30VDC 6A per channel instead of 20VDC 3A).

The following images display the project from different angles. Fig. 24 shows a blank copy of the PCB with no components while Fig. 25 shows an assembled version of the PCB. For the box, Fig. 26 shows the front wall of SIMPS and Fig. 27 shows a side profile of SIMPS. Fig. 28 shows all the

TABLE II
BOM COST FOR ONE PCB

60Hz Mains Transformer	\$125.85
EMI Filter	\$24.30
External Connectors	\$23.97
ESP32-WROOM x2	\$8.00
PCB Cost (JLC PCB)	\$28.25
PCB Components	\$222.11
Hardware Materials	\$52.31
TOTAL COST PER PCB	\$484.79

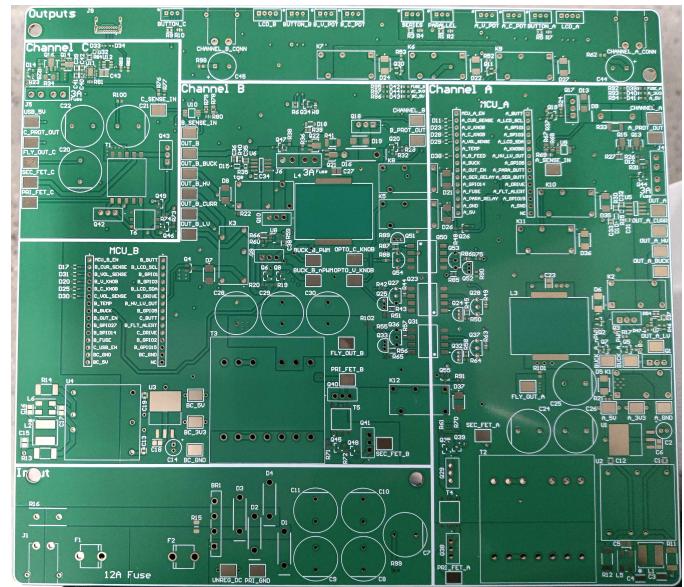


Fig. 24. Blank copy of PCB with no components assembled on.

parts used for testing SIMPS. Fig. 29 shows a dark image of SIMPS with the LCDs and buttons activated.

As a result of there being only one spin to the project, there are several changes and improvements to be made to the design. Some of these designs are to get the project to a much more stable and working point where the blocks from Table I can be marked as passed, while others are more advanced design decisions to bring this design closer to a typical industry-grade power supply.

The first two improvements involve simulating the full design, rather than individual blocks and correcting all the schematic errors. By simulating the entire design, any loading effect that could've been previously missed can be found and corrected in simulation instead of in real life. Once the simulation results are confirmed, the schematic can be designed from the simulation. The current schematic mistakes to correct would be the dot polarity for the flyback transformers, the replacement IC for the isolation transformer, the channel enable for the NMOS, and addition of the snubbers. Correcting the dot polarity issue on the transformer should remove most/all errors related to magnetostriction and thus make testing for the next spin much easier and feasible.

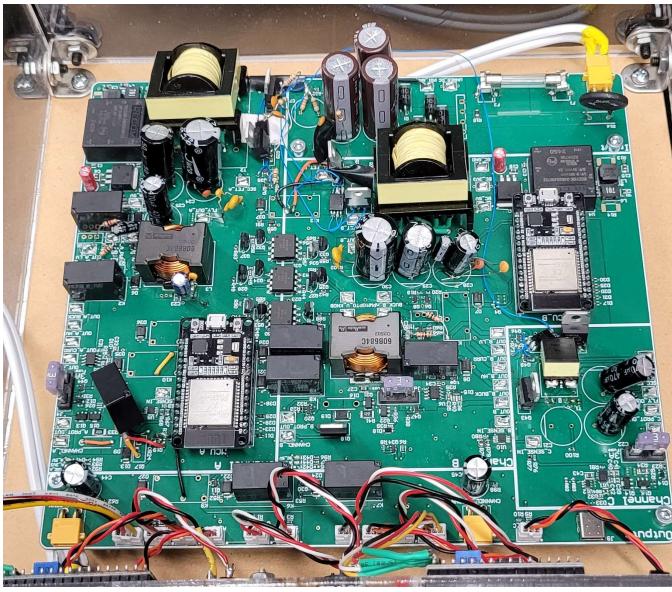


Fig. 25. Fully assembled PCB.

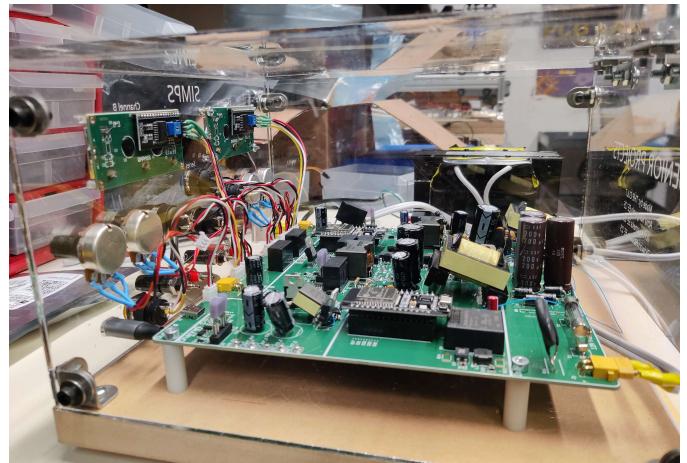


Fig. 27. Side profile of SIMPS to see the components from a side angle.

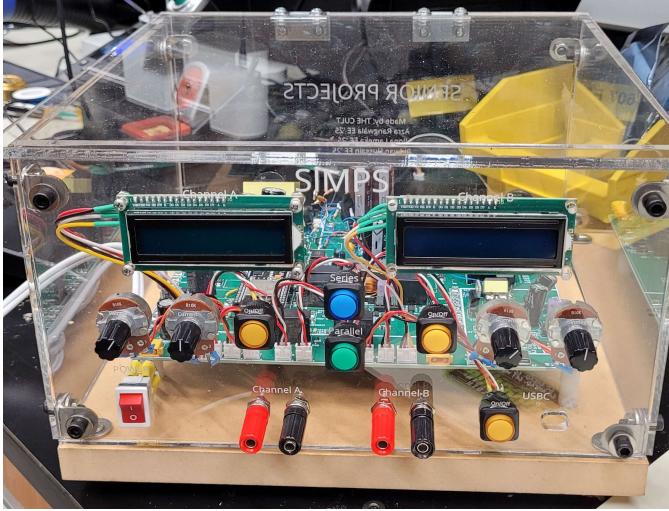


Fig. 26. Front wall design of SIMPS; this is what the user would interact with.

There is also a slight error on the placement for the NMOS for driving the LED for Channel B button on. From Fig. 30, the NMOS would need to be switched positions with the resistor so that the NMOS can be switched on properly.

Several alternative solutions were considered for the isolated gate driver and channel enable circuit. For the feedback transformer to work, a PMOS, such as the DMG2305UX-7 used elsewhere in the schematic, can be used. The $R_{ds, on}$ for the PMOS would be 0.05Ω , which is less resistance than the transformer coil (0.25Ω). The connection would be made on the secondary half of the transformer so that on primary side of the circuit after a 1:2 turns ratio, the voltage on the primary side would appear to be $7.14V$ ($5V \cdot \left(\frac{0.125\Omega}{0.125\Omega + 0.05\Omega} \right) \cdot 2 = 7.14V$), which is sufficient to turn the STP55NF06 ON/OFF.

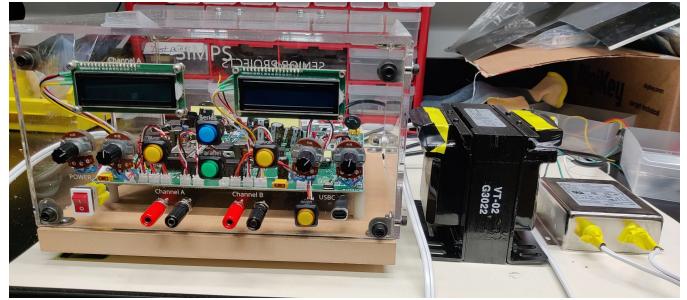


Fig. 28. Showing the parts used to test SIMPS. From left to right: SIMPS, 60Hz transformer, EMI filter.

The biggest concern would be if this would be a short circuit due to the lack of resistance, or if there's enough inductance to withstand this due to the high switching frequency, which can be tested in simulation. For the channel enable, instead of a relay circuit, a possible solution is to have a circuit as seen in Fig. 31 [9], where an NMOS is switching ON/OFF a power PMOS. This works because the PMOS is using a pull-up resistor to be OFF, and the NMOS can turn the PMOS ON.

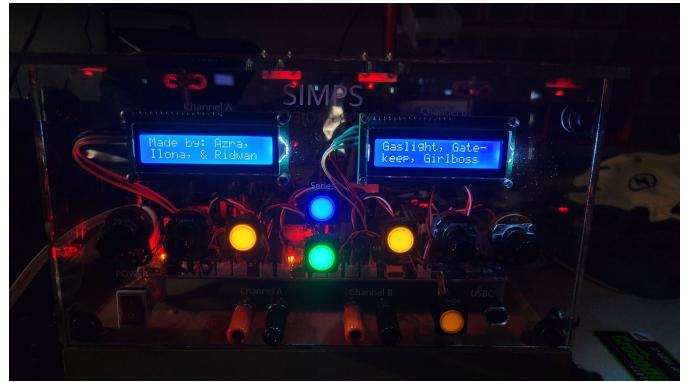


Fig. 29. Cool image displaying the LCD and buttons being active for SIMPS.

ON/OFF BUTTONS

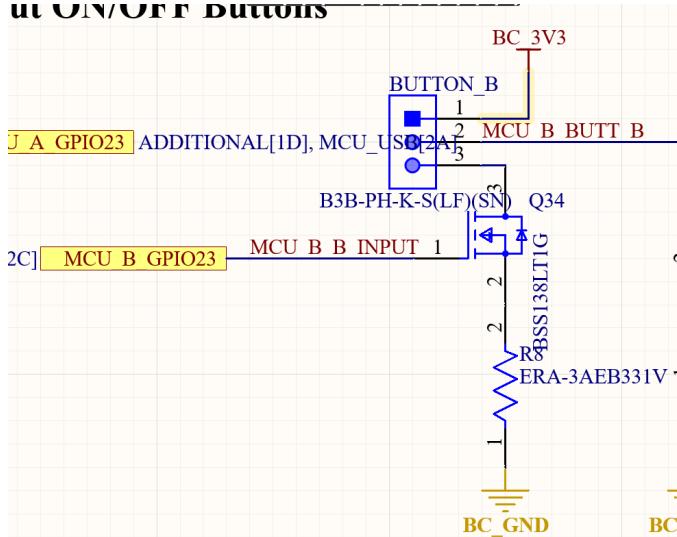


Fig. 30. Image showing the Channel B LED button circuit.

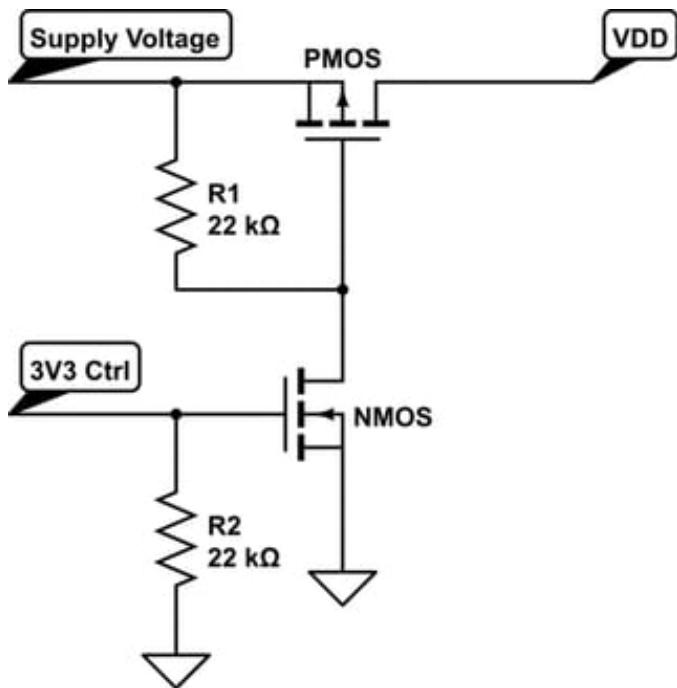


Fig. 31. Channel Enable using MOSFETs.

The next set of improvements are simple changes for making the circuit a cleaner design. The first improvement is to use a power PMOS and NMOS instead of only power NMOS for the synchronous buck converter, since the first MOSFET for the buck converter requires a high-side switch. This would simplify the controller logic for the buck converter. In addition, it would be worthwhile to do better research in determining what the most efficient MOSFET would be based on the application the MOSFET is used for.

Another adjustment is to increase the ADC range for the current sense so that the bits utilize much more of the available

voltage range as needed for the current design needs. This could be implemented through a subtractor OpAmp which adjusts the 0A ADC value closer to 0V and increase the gain of the ADC. Another adjustment for this section is to make sure the MCU position is better placed. The MCU will need to be further away from other tall components, such as the relays, and far away from the transformer to reduce any potential EMI interference. Finally, adding two-pin jumpers as a way to probe the circuit would be helpful since this allows for voltage probing, current probing, and a way to disconnect blocks of the circuit without needing to do additional soldering.

The advancements listed here are additional circuitry designs. The first one is to develop an onboard isolated buck converter instead of buying one commercially off the shelf. An isolated buck converter can be either a simple fly forward converter (DC-DC step down conversion with a transformer) or a slightly more complicated resonant converter. The main benefit of this is that the board would be cheaper, and space would be saved. More protection circuits would also have to be considered to protect the MOSFETs that would get damaged. It is also worthwhile to bring the EMI filter that was external to the board onto the board itself to make the design much more integrated. Finally, having a more active current limiter circuit would provide more safety. In addition, the ATC/glass cylinder fuses can be replaced with an electronic fuse.

The controller circuit can also be enhanced. Choosing a better MCU than the ESP32 to control most of the logic would help if more precision is needed. The additional features of the ESP32 (such as WiFi and Bluetooth capabilities) aren't necessary for this project, so there are other more suitable MCUs, such as the RP2350. The ESP32-WROOM-32 would be pushed to its limit if tested as intended, but the RP2350 has more pins and can do more computations than the ESP32. In addition, instead of using a development breakout board, routing the MCU directly onto the PCB would save space. Another idea is to generate the PWM via hardware (using a comparator and a sawtooth waveform) instead of through firmware, as this is typically done on most industry-grade power supplies. The benefit is that there won't be as much firmware overhead, reducing the computational stress on the MCU, and the PWM can adjust in real time, and won't be limited by bit precision. The PWM could also use a hardware PID control loop to ensure stable output on the feedback. It would also be worthwhile to spend time determining what the ideal switching frequency is, as this is a tradeoff between voltage ripple and the maximum voltage output.

For SIMPS to be much closer to an industry-grade power supply that could be sold on the market, the following changes would have to be made. The 60Hz mains transformer would need to be removed, and the design would need to work with 120Vrms instead of 24Vrms. The settle time between the button press and the output would have to be removed. Reducing the voltage rippling to an acceptable level would also need to be implemented. Finally, measuring and improving the main specs that are seen for a power supply would need to be taken into consideration. These specs include total harmonic

distortion, power factor, and efficiency.

Overall, the SIMPS capstone project has been an incredible learning experience. Countless hours were spent on the design, assembly and testing to try to achieve a working power supply. While the final results were not what was hoped for, the blocks that were successful were still worthy of celebration. The phrase “learn from your mistakes” was proven to be especially true during testing. This memorable project will certainly help the team members in their professional career as they continue on their journeys as engineers.

ACKNOWLEDGMENTS

The authors of this paper would first like to thank Professor Jabeom Koo and Professor Stuart Kirtman for their support during the two semesters of the capstone project. Special thanks to Professor Matthew Cavallaro, Professor Jeff Hakner, Dino Melendez, Michael Giglia, and Max Summers for their guidance through the theoretical and technical issues surrounding the project. Finally, we’d like to thank the Electrical Engineering Department at the Cooper Union for the Advancement of Science and Art for funding this project.

REFERENCES

- [1] R. Hussain, I. Lameka, and A. Rangwala, “Switch-Mode-Power-Supply,” 2025, GitHub Repository. [Online]. Available: <https://github.com/Ridwan-Hussain/Switch-Mode-Power-Supply/tree/main>
- [2] N. Mohan, T. M. Undeland, and W. P. Robbins, Power Electronics: Converters, Applications, and Design, 3rd ed. Hoboken, NJ, USA: Wiley, 2002.
- [3] “MOSFET Power Loss Calculator,” HeatsinkCalculator.com. <https://www.heatsinkcalculator.com/mosfet-power-loss-calculator.html>.
- [4] “Flyback RC Snubber Design,” Biricha Digital Power Ltd. <https://biricha.com/tools-and-downloads/flyback-rc-snubber-design/>.
- [5] “Switched Mode Power Supply (SMPS) - Working & Block Diagram,” Hackatronic.com. <https://hackatronic.com/electronics-tutorial/power-supplies/smmps/>.
- [6] Rohde & Schwarz, “Understanding Switching Mode Power Supplies,” YouTube, Feb. 9, 2024. <https://www.youtube.com/watch?v=MS5vd3LomWg>.
- [7] ElectrArc240, “Every Component of a Linear Power Supply Explained (while building one),” YouTube, Oct. 16, 2024. <https://www.youtube.com/watch?v=UTetQhGyUVg>.
- [8] ElectrArc240, “Every Component of a Switch Mode Power Supply Explained,” YouTube, Jul. 8, 2024. <https://www.youtube.com/watch?v=3FyXqNcqvRM>.
- [9] “Power NMOS used as voltage switch,” Electrical Engineering Stack Exchange, Nov. 18, 2018. <https://electronics.stackexchange.com/questions/407463/power-nmos-used-as-voltage-switch>.
- [10] Z. Peterson, “How to Connect Grounds in an Isolated Power Supply PCB Layout,” Altium Resources, Aug. 23, 2021. <https://resources.altium.com/p/how-connect-grounds-isolated-power-supply-pcb-layout>.
- [11] “Controlled Impedance Calculator,” JLCPCB. <https://jlcpcb.com/pcb-impedance-calculator/>.
- [12] “Is it a bad idea to use 2 vias to change layers in power tracks?” Electrical Engineering Stack Exchange, Jul. 1, 2021. <https://electronics.stackexchange.com/questions/573785/is-it-a-bad-idea-to-use-2-vias-to-change-layers-in-power-tracks>.
- [13] “Looking for advice to prevent reverse current flow,” Electrical Engineering Stack Exchange, Jul. 21, 2022. <https://electronics.stackexchange.com/questions/628298/looking-for-advice-to-prevent-reverse-current-flow>.
- [14] “EMI filter calculation in a SMPS.” Electrical Engineering Stack Exchange, Sep. 27, 2012. <https://electronics.stackexchange.com/questions/38886/emi-filter-calculation-in-a-smps>.
- [15] “Looking to build a signal level converter to reduce 20V input to 3.3V out. Am I doing it right?” Electrical Engineering Stack Exchange, Apr. 3, 2014. <https://electronics.stackexchange.com/questions/84286/looking-to-build-a-signal-level-converter-to-reduce-20v-input-to-3-3v-out-am-i>.
- [16] “Input impedance of ADC and reading voltage from voltage divider,” Electrical Engineering Stack Exchange, Mar. 28, 2023. <https://electronics.stackexchange.com/questions/632065/input-impedance-of-adc-and-reading-voltage-from-voltage-divider>.