

Compact DC/AC Power Inverter

Inspired by the Google/IEEE *Little Box Challenge*



ECE 480 Team Seven

Philip Beard
Jacob Brettrager
Jack Grundemann
Stanley Karas
Travis Meade

Executive Summary:

As the world continues further into the 21st century, a need for clean and renewable energy is making itself apparent; the global energy portfolio is calling for more energy and from sources that emit fewer greenhouse gas emissions. With the implementation of solar panels, formidable direct current (DC) voltages can be created with little negative impact on the environment. The voltage from these panels can be used to charge batteries, but their usefulness greatly expands when inverted to alternating current (AC). From here, it can be used to power a large number of devices or put back onto the grid to be used by others.

Group Seven has been tasked with creating a compact power inverter to do just that. The project is inspired by *The Little Box Challenge*, a contest created by Google and IEEE. The group has modified the design constraints slightly due to limited time and resources, yet much of the theory and challenge remain.

The team has been successful in designing the circuit architecture, controlling mechanism, printed circuit board design and layout, enclosure, filtering techniques, and heat dissipation. An industry standard 60 Hz, 120 Vrms sine wave has been produced and used to power devices, and a large amount of testing has been performed; all parts fit into an enclosure only slightly larger than the contest outlined 40 cubic inches.

Acknowledgments:

This project has been a tremendous undertaking that has challenged each group member. Team seven would like to thank the following individuals for their contributions to the progress of the compact power inverter project:

- Professor Aviyente - Professor Selin Aviyente, group facilitator, has assisted the group in presentation preparation, documentation preparation, and timeline management. She made valuable suggestions that helped keep the group on schedule.
- Professor Grotjohn - Professor Timothy Grotjohn served as the group sponsor. In addition, he helped to keep the group focused on their goals with weekly meetings and suggestions.
- Professor Wierzba - Professor Gregory Wierzba loaned the group the infrared thermometer and one of the power supplies used for circuit testing.
- Tom Larter - Tom, an electrical engineering graduate student, has been invaluable to the group's understanding of important concepts over the semester. He was especially helpful in explaining Arduino programming logic.

Table of Contents

Chapter 1: Introduction and Background

Section 1.1: Purpose	4
Section 1.2: Problem	4
Section 1.3: Objectives	5
Section 1.4: Current Technologies	7
Section 1.5: Impact	8

Chapter 2: Solution Exploration and Selection

Section 2.1: FAST Diagram	9
Section 2.2: Critical Customer Requirements	10
Section 2.3: Budget	11
Section 2.4: Schedule	12
Section 2.5: Successful Approach	13

Chapter 3: Technical Work

Section 3.1: Hardware Design	14
Subsection 3.1.1: H-Bridge and Gate Driver Design	14
Subsection 3.1.2: PSpice Modeling	17
Subsection 3.1.3: Microcontroller Programming	22
Subsection 3.1.4: DC/DC Conversion	25
Section 3.2: Hardware Implementation and Photo Documentation	26
Subsection 3.2.1: PCB Design and Enclosure Layout	26
Subsection 3.2.2: Heat Transfer Design	28

Chapter 4: Testing and Proof of Design

Section 4.1: Hardware Testing	31
Section 4.2: Software Testing	35

Chapter 5: Conclusions

Section 5.1: Final Design	36
Section 5.2: Cost	36
Section 5.3: Timeline and Setbacks	36
Section 5.4: Findings	37
Section 5.5: Future Work	38

Appendix 1: Individual Technical Roles and Accomplishments	39
--	----

Appendix 2: References	44
------------------------------	----

Appendix 3: Detailed Technical Attachments	45
--	----

Chapter 1: Introduction and Background

Section 1.1: Purpose

The main goal of design team seven was to research, design, and build a compact power inverter. The inverter was outlined in a contest called *The Little Box Challenge*, which was created jointly by Google and IEEE. The main use of this inverter is to accept DC from either a single solar cell or an array of them and convert it to AC. Group seven is attempting to make a compact power inverter rated at 1000 W. The device is to accept DC from a high powered supply and invert it to an industry standard of 120 Vrms, 60 Hz sine wave. This will be able to either power many household devices or supplement the power grid.

In order to build a high power, compact inverter, the group will put their engineering education skills acquired over the past 4-5 years to use. They will learn a great deal about both new technologies and working professionally as a team. This will be an important experience to have prior to graduation because all of them will be working on multidisciplinary teams for the rest of their lives. The team will work closely with a sponsor and a facilitator to ensure goals are met precisely and on time.

Building an inverter is also important to gain experience performing independent research. Lifelong learning is one of the skills stressed at Michigan State University and learning outside of the scope of classes immediately before graduation will be helpful in forming the habit.

Section 1.2: Problem

Put shortly, the solution team seven is trying to create is a practical power inverter for the masses. As the need for clean, renewable energy sources grows and the price per watt of solar cells continues to fall, energy from the sun is making itself known as a viable solution. Powerful DC/AC inverters have existed for many years, but their large size (figure 1) makes them impractical for a large number of residential uses, particularly with roof-mounted cells that cannot devote the space to a cooler-sized device.

By making a power inverter that is both compact and powerful, solar cells become more viable to the mass market. Instead of having the inverter specially delivered, mounted, and taking up several cubic feet, a device smaller than a box of tissues could be purchased in a store, mounted to the wall of a garage or shed, and be available wherever needed.

Team seven believes that at the correct price, size, and power rating, its power inverter could enable many people who were on the fence about implementing a personal solar array to take the leap.



Figure 1. Standard Solar Power Inverter Alongside Solar Array

http://upload.wikimedia.org/wikipedia/commons/6/65/M%C3%BCllberg_Speyer_-_2.JPG

Section 1.3: Objectives

Group seven has chosen to base their project off of the Little Box Challenge, whose full details can be found here: <https://www.littleboxchallenge.com/>; the major requirements are presented in table 1.

The challenge is open to the public but is strongly focused on a university atmosphere and to be worked on by doctorate students and researchers. The contest organizers have paired with a number of parts manufacturers, including CREE, GaN Systems, and GeneSiC that are showcasing the capabilities of their newest transistors. The existence of these new technologies could help to usher in a new era of compact, but very powerful inverter systems.



Figure 2. Little Box Challenge Logo

http://1.bp.blogspot.com/-qV0Cazd_QF8/U82mucXsHFI/AAAAAAAAATY/fzkr4mnJZJg/s1600/HiResLBCGoogleIEEELogo-UR.jpg

Google and IEEE have outlined the exact parameters required to enter the challenge (shown in table 1). It was decided that these design parameters, which have a \$1 million reward for success, were doable but not with the limited amount of time and resources available for the 15 week semester project.

Parameter	Requirement (Nominal)
Maximum Load	2 kVA
Power Density	> 50 W/in ³
Volume	< 40 in ³
Voltage Input	450 VDC
Voltage Output	240 VAC (RMS)
Frequency Output	60 Hz
Maximum Outer Temperature	< 60 °C
Electromagnetic Compliance	FCC Part 15 B

Table 1. *Little Box Challenge* Design Parameters

Instead, the group had a number of meetings with their sponsor and facilitator which decided that modifying the requirements slightly would allow them to both learn a great deal about power inverters, working as a team, and also have a finished product at the end of the semester. These modifications included cutting the power rating in half, using a lower input voltage and outputting a lower voltage, and are outlined in table 2 below. These requirements are largely self-imposed goals by the group as explained later in section 2.2.

Parameter	Requirement (Nominal)
Maximum Load	1 kVA
Power Density	> 25 W/in ³
Volume	< 40 in ³
Voltage Input	200 VDC
Voltage Output	120 VAC (RMS)
Frequency Output	60 Hz
Maximum Outer Temperature	< 60 °C
Electromagnetic Compliance	FCC Part 15 B

Table 2. Chosen Design Parameters (Modifications in Red)

As shown, the objectives for the project include a power rating of 1 kVA, an enclosure dimension of less than 40 cubic inches, a 200 V DC input, FCC part 15 B electromagnetic compliance, a surface temperature of no more than 60 degrees celsius (140 degrees fahrenheit), and an industry standard 120 Vrms, 60 Hz output.

Section 1.4: Current Technologies

As stated earlier, power inverters have existed for many years, but they are large in size. Input transformers for stepping voltages up or down, DC/DC converters for powering peripherals, large inductors and capacitors for output filters, and isolation/output transformers all take up a significant amount of space, which is part of the reason they are so large.

Obvious limitations to this practice include where they can be used. By making an inverter small enough to be mounted on the wall of a garage by someone with limited experience, it becomes more practical to use by a much larger audience and wider market. Another large limitation of the current practice is price. A fairly standard power inverter used in solar applications today is the Magnum MagnaSine MS-4448PAE. This unit is capable of outputting up to 4 kW of power, but also weighs 55 pounds, is 1400 cubic inches in size, and costs \$2100.



Figure 3. MagnaSine MS-4448PAE Power Inverter

<http://www.wholesalesolar.com/products.folder/inverter-folder/magnum-ms-4448PAE.html>

Depending upon the size, features, and power ratings of inverters on the currently available, the price can fluctuate greatly. Some inverters cost upwards of \$5000. The high price of inverters is another reason why, despite falling prices of solar panels themselves, widespread use of personal solar arrays is not commonplace.

Section 1.5: Impact

The biggest difference that successfully creating a compact power inverter would make is increasing the feasibility of residential solar power. Panels, inverters, and other required technology already exist and are available to the public, but the price and large size of many solutions are a deterrent. Government tax incentives have aided consumers in the past to purchase more energy efficient technology, but the creation of a product without the need of government assistance will be more successful in the market. The creation of a compact and inexpensive inverter could kick off a trend of similarly sized and priced solar peripherals. If adopted by a majority of households, greater solar generation would make a large impact on their carbon footprint. The air would be less polluted and controversial energy harvesting techniques could be reduced.

Chapter 2: Solution Exploration & Selection

Section 2.1: FAST Diagram

The Function Analysis System Technique (FAST) diagram is an important tool to provide a visual aide for a product's functions. The basic function on the far left is supported by the secondary functions to the right. Team seven's FAST diagram is shown in figure 4 below:

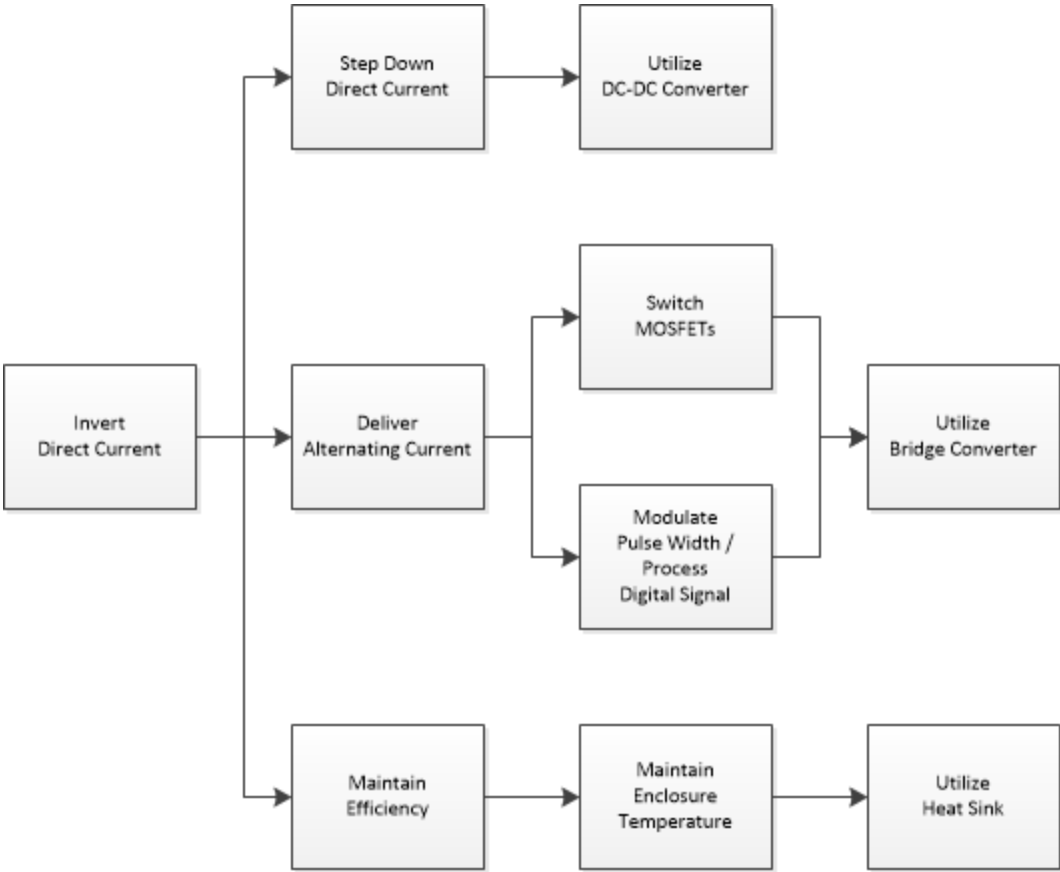


Figure 4. DC/AC Inverter FAST Diagram

The basic function of the design is to invert the DC input into an AC output. Secondary functions include stepping the DC, delivering AC, and maintaining efficiency. To invert DC, the system must first step the DC to the AC peak amplitude. DC can be stepped down with the use of a DC/DC converter. Current is alternated by switching MOSFETs on and off using pulse width modulation (PWM) delivered from a microcontroller, through a gate driver, and to the gate of the transistor. High efficiency will ensure less power is dissipated as heat. This is important because the enclosure will not contain a large excess of free space, so maintaining an acceptably low temperature will be more difficult with a compact heat sink.

Section 2.2: Critical Customer Requirements

As a part of quality function deployment it was necessary to develop the best solution of how to invert DC/AC by identifying the critical customer requirements (CCRs). For this project the customer was team seven's sponsor, Dr. Grotjohn. He indicated that the main priorities for the inverter were its power density and efficiency of the DC/AC, DC/DC for the microcontroller and gater driver losses were deemed acceptable. All other Little Box Challenge requirements were secondary and could be adjusted by the group to meet the 15 week deadline. The requirements set in table 2 of section 1.3 were therefore set as goals by the team. After discussion with Dr. Grotjohn, he agreed that the requirements set by the team were an appropriate challenge for the semester.

The best solution was largely based upon the input and output voltage as well as the topology. For maximum efficiency provided the given budget, an H-bridge topology would prove the most effective. Standard silicon components were immediately found to be inadequate for the power demands and were replaced with silicon carbide. With an AC output of 120 Vrms and 60 Hz, testing and demonstration could be completed with standard devices found around a house or apartment. The voltage peak for the output of 170 V was necessitate as the input to the rail of the H-bridge. The 200 V input was set to allow for research of DC/DC conversion, but was not feasible due to time constraints as discussed later in section 5.3.

Section 2.3: Budget

Each design team was provided with a standard budget of \$500 that could be exceeded only with sponsor approval. As cheap, low powered inverters are readily available on the market, team seven worked to avoid exceeding its budget even when other high power inverters are very expensive. The budget for the final inverter is detailed in table 3:

Item	Price	Quantity	Total	Description
C2M0025120D	\$71.99	4	\$287.96	Silicon Carbide MOSFETs
FAN7382	\$3.75	2	\$7.50	600 V Gate Driver
DC/DC Converters	\$2.02	2	\$4.04	Supply Rail for Micro Controller
LM2940	\$1.65	1	\$1.65	5V regulator
Enclosure	\$24.37	1	\$24.37	Aluminum Enclosure
Arduino Microcontroller	\$4.95	1	\$4.95	ATMega328
PCB	\$0	1	\$0	Printed Circuit Board from the ECE shop
		TOTAL:	\$330.47	

Table 3. Silicon Carbide, high power inverter budget

The Silicon Carbide MOSFETs for the final inverter were by far the largest cost items of the project and required that all further purchases be scrutinized. For a professional appeal, but more importantly efficiency, an aluminum enclosure was selected. All other components were necessary for the secondary functions necessary to invert power. The majority of small components were available freely through the ECE shop and would be insignificant to the final cost if purchased in bulk.

Section 2.4: Schedule

Developed by Henry Gantt in the 1910's, a Gantt chart is used to illustrate a project timeline. Undoubtedly, there were setbacks and issues during the semester that meant that the original timeline was to be pushed back. However, the group was diligent to stick to the original timeline as closely as possible. Table 6 in appendix 1 was used to develop the Gantt chart seen below in figure 5:

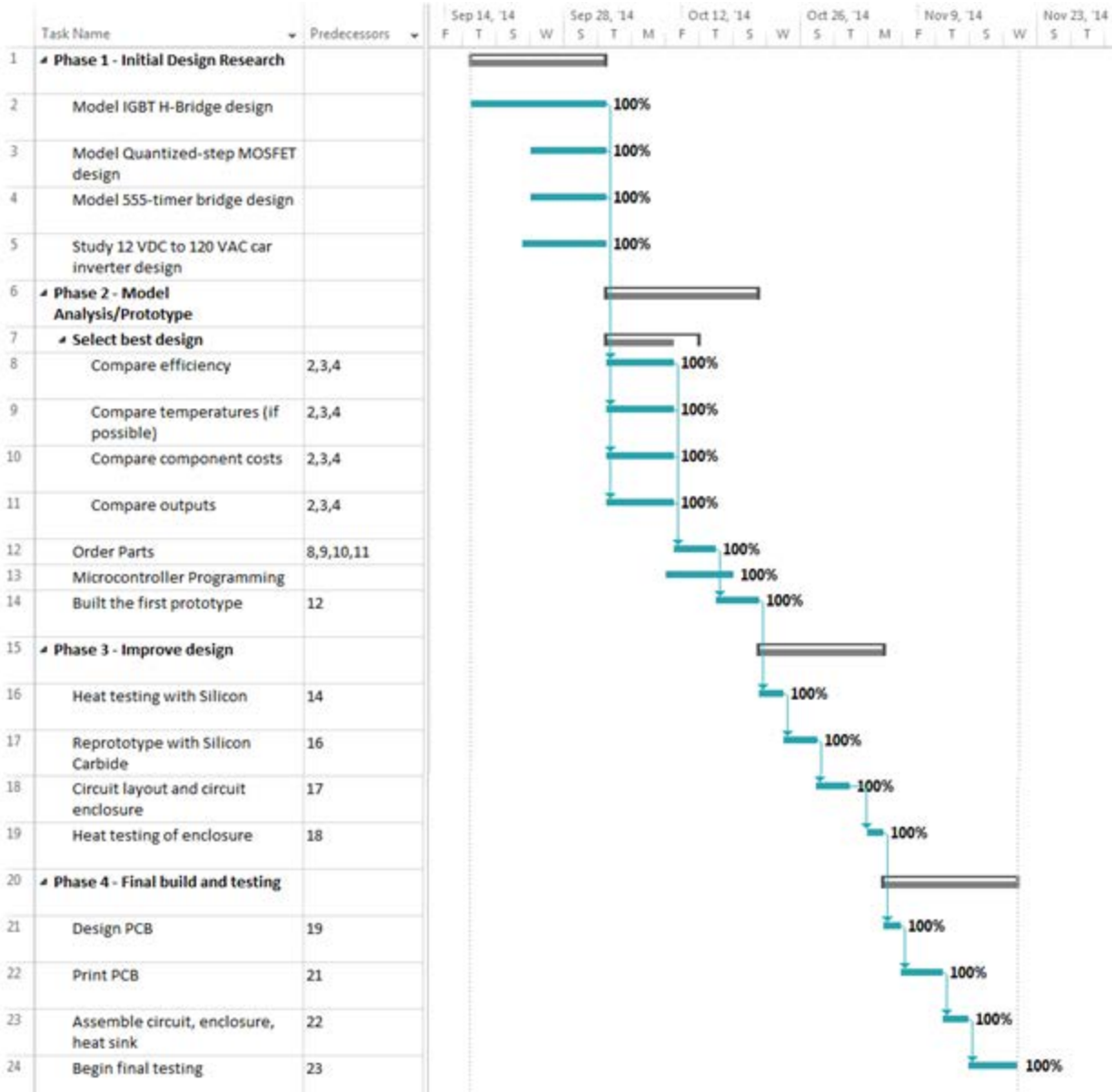


Figure 5. Gantt Chart

Section 2.5: Successful Approach

There are a number of differences between team seven's approach and the current practice, including the triggering method, material type, absence of transformers, and the aluminum enclosure itself.

First, it was decided that the use of a fully-rectified H-bridge was to be the basis of the design. Four transistors would be arranged in such a way that when a load is placed between the two rails of the H Bridge current will change directions 120 times per second, essentially emulating a 60 Hz sine wave whose voltage is capable of swinging positive and negative. This way only a positive power rail is needed, rather than requiring a negative power rail as well. Second, the group decided that the use of PWM was the best way to trigger the transistors in the H-bridge. PWM, as its name suggests, is the adjustment of the width of signals with identical amplitudes. When a time average is taken, the actual power delivered will be somewhere between 0 and 100 percent of the input, depending upon the width of the pulse. Using PWM enables the transistors to act as switches that are either on or off; the timing is modified such that a sine wave is apparent after filtering. Third, team seven decided that using alternative transistors was necessary to meet the power requirement. Silicon metal-oxide-semiconductor-field-effect-transistors (MOSFETs) were standard material used throughout the teams undergraduate experience, but would not be able to withstand the high power needed for this design. Instead, both silicon-carbide and gallium-nitride technologies were researched. Although gallium-nitride transistors are generally accepted as having a better frequency response and power handling, silicon-carbide was chosen for its ease of finding parts and slightly lower cost.

The next thing the group did differently was make the decision not to use traditional transformers in their design. This decision was an obvious one in that the group had to choose either a very expensive transformer technology that did not fit into the budget or a large, more traditional transformer that would not physically fit within the space constraint. Instead, it was decided that DC/DC converters would be the correct tradeoff of space occupied and price. These brought their own complications though, as will be described later. The final large difference between what currently exists on the market and the design of team seven was cost and weight savings. Although a more powerful inverter would require extra components and heat considerations, it is unlikely that doubling or even quadrupling the power rating of the team's inverter would cause the size and weight to come close to some of the options on the market today, particularly the inverter mentioned in chapter 1, section 4. In addition to being smaller and lighter, the groups inverter is significantly less expensive, albeit with a smaller power rating. The prototype costs less than \$500 to build.

Chapter 3: Technical Work

Section 3.1: Hardware Design

Subsection 3.1.1: H-Bridge and Gate Driver Design

An H-bridge is a set of four switches that are assembled in such a way that an arbitrary load impedance is decoupled from a DC power rail and ground. For a DC/AC inverter this circuit architecture is utilized to control the direction of current across an arbitrary load by manipulating the four switches in the bridge. Each of the four switches in the H-bridge are independently controlled and work to divert current across the load at a frequency of 60 Hz. An example of a simple H-bridge with four switches and single load impedance is shown in figure 6:

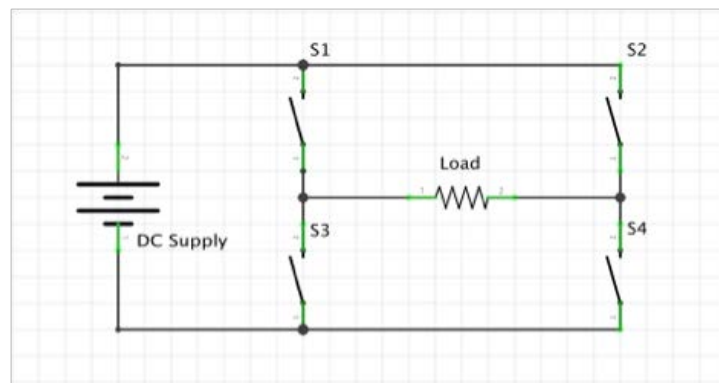


Figure 6. Example H-bridge

Each of the switches shown in figure 6 have different roles for typical operation of an H-bridge. The first important distinction between the different switches within the circuit is that the top two switches are referred to as the high side and the bottom two switches are referred to as the low side. The high side switches are responsible for controlling the availability of the DC rail voltage across the load impedance while the low side switches are responsible for controlling the connection between the load impedance and ground.

Once this basic architecture was decided upon, the next step was for the group to start selecting what part type would act as each of the four switches within the H-bridge. After reviewing several options the group decided on using four N-channel silicon carbide MOSFETs (figure 7) in each of the switch positions. Silicon carbide has a very high heat tolerance which makes it an ideal candidate for a high power inverting application. These parts also can operate up to 1200 V from the source to the drain thus allowing a large amount of flexibility in changing the rail voltage that is available to the inverter.



Figure 7. SiC MOSFETs

After selecting the final MOSFET to use in the design, the next step was to test the initial inverter design using cheaper silicon parts that could be burnt out with less worry from the group if replacement of a burnt out transistor was necessary. For these initial tests, the IRF530 N-channel MOSFET was used in place of the silicon carbide MOSFETs. The IRF530 is limited to 100 V from its source to drain, so these initial tests were done at lower voltages than the final design would be operating at. The first thing the group realized in prototyping the H-bridge design was the obstacle that high side switching devices present to the control interface. In the case of the inverter designed by group seven, the high side MOSFETs are connected with the source tied to the DC rail and the drain tied to the high side of the load impedance. For the case of the IRF530 the gate-to-source threshold voltage is on average 5 V, or more simply put the average voltage that must be applied to the gate of the MOSFET in order for it to conduct between the source and drain is 5 V above the source voltage. After making this realization the group quickly determined the proper fix to this problem was to utilize gate drivers that could momentarily apply a voltage high enough to properly activate the MOSFETs.

The gate drivers work by using a bootstrap capacitor that charges each time the low side MOSFET is conducting to ground. The bootstrap capacitor is tied to the gate driver supply voltage (further referred to as VCC) through a diode so that the voltage from the DC rail of the H-bridge can not bleed back into VCC. When the low side signal is conducting to ground the capacitor is connected to ground and the diode allows the voltage at VCC to charge up the capacitor. When the high side signal from the microcontroller is present at the high side input of the gate driver the bootstrap capacitor is now charged to VCC. This voltage is added to the DC rail present at the input of the gate driver so the supplied high side gate voltage is now the DC rail plus VCC. This is enough to exceed the threshold voltage of the transistor and make it begin conducting from the source-to-drain.

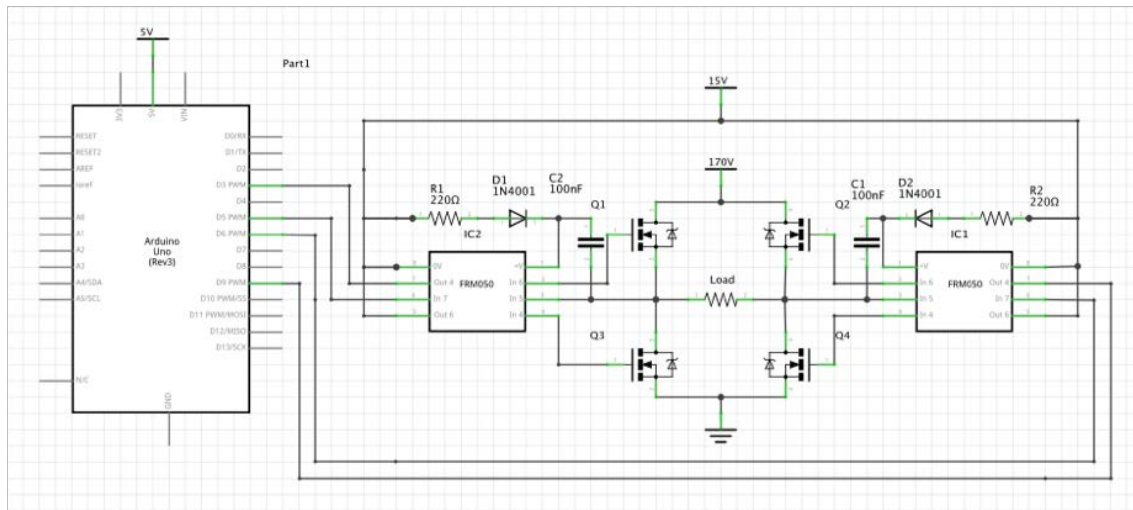


Figure 8. Inverter Schematic

Aside from properly switching the high side drivers in the H-bridge configuration, the group also needed to consider some of the more pressing design considerations of the MOSFETs selected for the circuit. One of the important pieces of information gathered by the group in selecting parts was the relation between the gate-to-source voltage required to activate the part and the current running through the drain of the device. As with most MOSFETs, as the current through the drain is increased the threshold voltage also begins to increase. To account for this increase the group chose a large enough VCC value such that the gate driver voltage would be operating at a high enough value to always meet this threshold based on the rated current value of the inverter. An example of this relationship can be seen in figure 9 below.

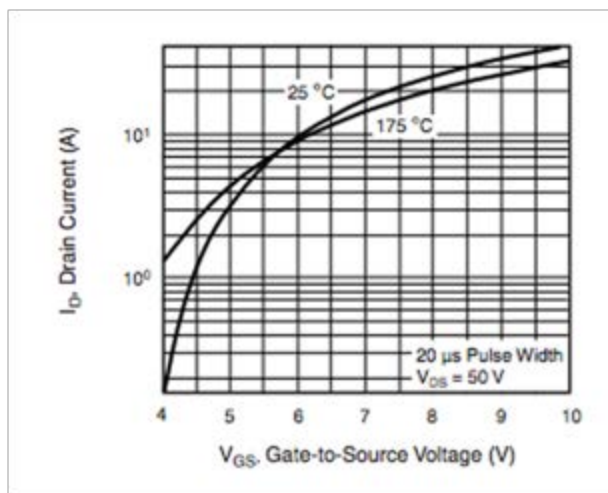


Figure 9. Gate-to-Source Voltage vs. Drain Current

Another important consideration in selecting parts for the inverter is the source to drain voltage of the MOSFET chosen. The average ON resistance of an IRF530 MOSFET is approximately 160 mΩ between the source and drain. Any load that is attached to the output of the inverter will result in a varying amount of current to be pulled through the terminals of the MOSFETs. The maximum amount of current the device is rated for is approximately 6 A, so the maximum power dissipation through the MOSFET at this current rating is:

$$(.16 \Omega * 6 A) * 6 A = \mathbf{5.76 W}$$

This means that at maximum current draw the MOSFETs will have to dissipate is 5.76 W. According to the datasheet for the IRF530 the device can handle up to 88 W dissipated across it, so the 5.76 W of predicted power dissipation should be well within the range of what the part is rated for. The silicon carbide MOSFETs selected by the group have a much higher power rating and a much lower ON resistance, meaning the group will be well within the range of operation for these set of parts as well.

After all of these considerations have taken place the final relevant piece of design information needed in applying an H-bridge to the inverter design was the pull down resistance between the gate driver output and the gates of each resistor. These resistors are included to give the charge applied by the gate driver a path to ground once the gate driver has stopped supplying a signal to the MOSFETs. These pull down resistors must be included in order to dissipate the charge from the gate of the MOSFET and stop the device from conducting between its drain and source when the gate driver is not sending a high signal.

Subsection 3.1.2: PSpice Modeling

After the group had decided on the H-bridge architecture the next step to bringing this design into reality was to model the circuit in PSpice at lower prototyping voltages. The group started by creating a simple H-bridge in PSpice using IRF530s and simple Pulse functions to act as the control signals coming from the gate driver. The results of these tests illustrated the idea behind creating an inverting current using an H-bridge so the went forward with the design and began constructing a prototype to the design created in the first initial PSpice tests. The PWM results can be viewed in figure 10 on the following page.

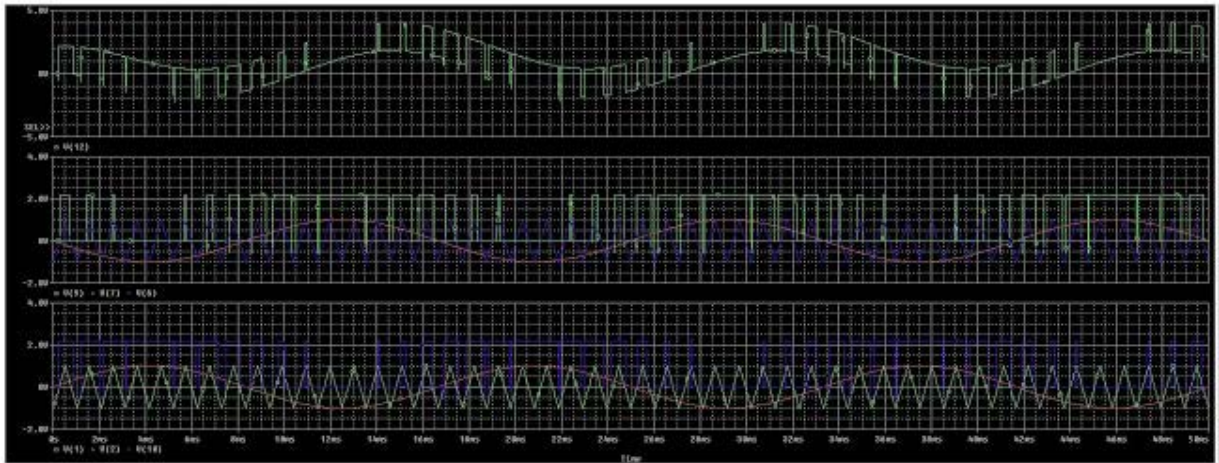


Figure 10. PWM PSPICE Modeling

Once the group had a working prototype, the next step was to start comparing the data gathered in lab to the data in PSpice. Prior to model the LC filter that would be fitted to the output of the inverter, the group worked to recreate as much of the lab data as possible within PSpice. This data was later used as a reference aid for the group in order to ensure the results that were seen in lab. With this lab recreation in PSpice, the group also was able to visualize several problems with the circuit prior to building and testing in lab. One of the more valuable design solutions that came from continuous PSpice testing was a change in the PWM that was being sent to the gate drivers. Initially the group thought to only have one side of the bridge switching at any given time, meaning either the high side or low side would be acting during each 60 Hz cycle. After this was tested in PSpice the group realized that the quality of the Pulse Width Modulated signal observed at the output of the inverter was not very good as a result of this continuous high side switching. A solution was found by the group and entailed alternating the high and low side switches on either side of the bridge in order to create a ground reference in the times where the high side drivers were not activated. This change made a large difference in the quality of the output signal and allowed the group to reduce the size of many of the parts that were initially selected for the much longer high time.

After the group had created a reliable model of the prototyped H-bridge and gate drivers in PSpice the next step was to begin designing an LC filter to demodulate the 60 Hz sine wave encased in the PWM as seen in figure 11. Several initial designs were proposed and tested within the PSpice environment. The main advantage to the group in testing the LC filter this way was the ease with which the group could change component values without having to wind an inductor or search for a properly rated capacitor. With each LC filter modeled by the group, different loads were tested on the output and power absorption ratings were gathered to determine which filter would best fit the desired application.

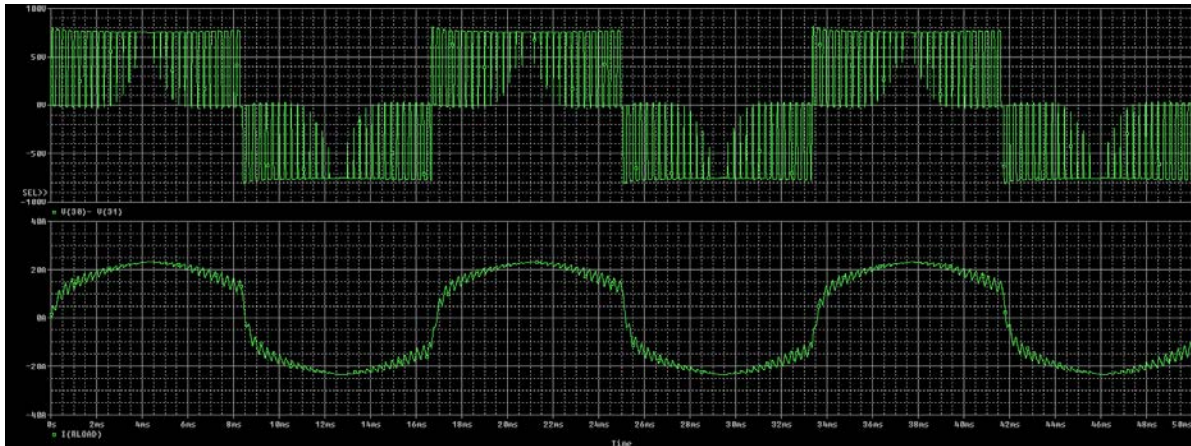


Figure 11. PSPICE Sine Wave Output

Once the final LC Filter design had been chosen the group tested the entire circuit in PSpice and used results from these tests as benchmarks for the final design of the inverter. Utilizing PSpice as a design aid to test the full circuit prior to construction saved the group a significant amount of time that otherwise would have been spent on finding parts to perform time consuming, live tests.

Once the final inductor and capacitor values were chosen for the resonant LC filter and the desired sine wave was demodulated from the PWM signal, the entire filter was characterised in PSpice by sweeping a 340 V peak-to-peak sine wave from 10 Hz to 10 kHz from the input through different loads. The first point of interest that was immediately noticed by the group was the resonant frequency peak that grew significantly as the load impedance was increased. Figure 12 shows that as the load was increased the resonant frequency of the LC filter caused a very noticeable spike in the output voltage supplied to the load. The group also measured the maximum power transferred to the load at each of these different load impedances, which is the first graph in each test. The second graph in each tests displays the voltage across the load impedance. All of the tests done in figure 12 were performed with the same AC sweep input and same magnitude voltage supply.



Figure 12. PSPICE Simulation of changing loads

Using the data illustrated in figure 12, the group began researching different ways to mitigate the undesired resonant peak from the LC filters corner frequency. One of the first designs that was suggested for dissipating the energy spike at the resonant frequency was to add a series damping resistance between the capacitor bank and ground. Several different values were tested to understand how the resonant peak could be dampened at different resistances and the final value chosen was 101 k Ω . This value was chosen because it allows the group to easily mitigate the required resonant voltage at a reasonable power rating that could be realized with two quarter watt resistors.

The graphs shown in figure 13 are the test results of the power and voltage delivered to the load impedance using an AC sinusoidal sweep from 10 Hz to 10 kHz at 340 V peak-to-peak. The resulting data from these tests show that the damping resistor has completely absorbed the resulting resonant peak from the LC filter and delivered a constant level of power to the load. Performing these tests also exemplified the impedance range the inverter can be used before the corner frequency is too large for normal demodulation of the PWM. The last test shown in figure 13 was done using a load impedance of 1000 Ω .

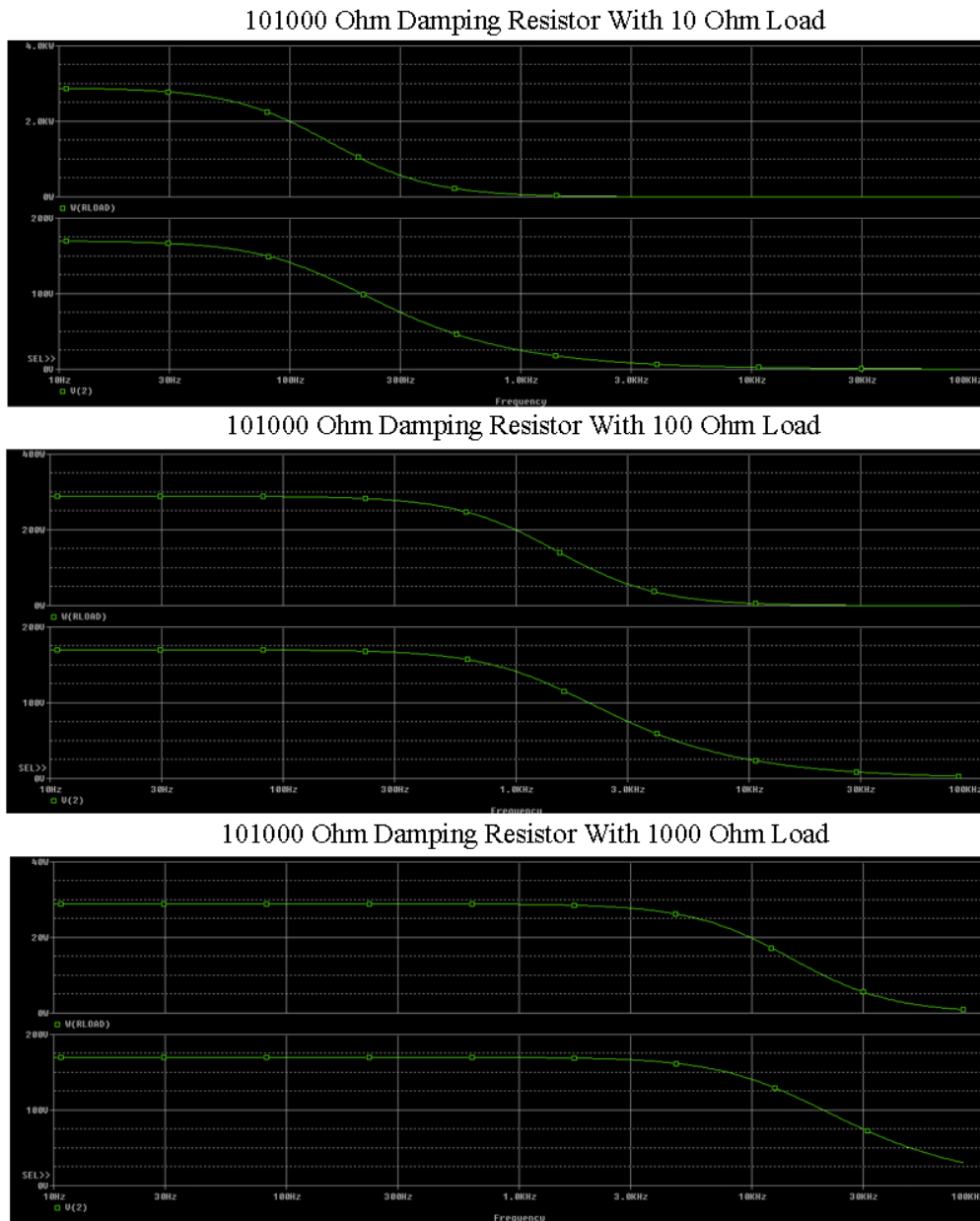


Figure 13. Changing Load with Damping Resistor

Using the PSpice data as a benchmark for the actual design of the final circuit the group has sped up a lot of troubleshooting because they now have the ability to reference a working design and compare the desired signal to the lab signal being recreated. This has helped the group overcome any obstacles much faster than what they would have been capable of if there was only the data available from the lab tests and theoretical situations. Utilizing PSpice as a tool for both design issues and for testing the final circuit helped the group visualize all of the other possible issues that they may encounter while working with the circuit in an actual lab setting.

Subsection 3.1.3: Microcontroller Programming

For controlling the gate driver and H-bridge, the team knew that they would need a microcontroller. The Arduino Uno was chosen due to its 5 V operating voltage and because it is a fairly easy microcontroller to learn and program. The H-bridge design requires a PWM signal in order to vary the output and eventually achieve a sine wave. There are a few possibilities to create a PWM signal using the Arduino.

Before discussing the methods used it is important to understand how to use the Arduino interactive development environment (IDE). The code is composed of three main areas. These code areas can be seen in figure 14 below. The first area, located at the top of the program, is where the variables are initialized. This allows the variables to be identified and used later in the program. The second area is the void setup. This is where Arduino pins are declared as either an input or an output. This indicates to the Arduino whether or not it needs to be storing the information given from the pin or outputting a signal. The variables initialized above this section are able to be placed in place of a specific pin number. This allows for the program to be coded and followed easier. The void setup may also have code that needs to be run once. The last area is the void loop. This is where the majority of the code resides. The Arduino continually runs this loop until it is no longer powered.

```

int led = 9;           // the pin that the LED is attached to
int brightness = 0;   // how bright the LED is
int fadeAmount = 5;   // how many points to fade the LED by

// the setup routine runs once when you press reset:
void setup() {
  // declare pin 9 to be an output:
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  // set the brightness of pin 9:
  analogWrite(led, brightness);

  // change the brightness for next time through the loop:
  brightness = brightness + fadeAmount;

  // reverse the direction of the fading at the ends of the fade:
  if (brightness == 0 || brightness == 255) {
    fadeAmount = -fadeAmount ;
  }
  // wait for 30 milliseconds to see the dimming effect
  delay(30);
}

```

Figure 14. Example Arduino Code

The first method the team utilized was a premade function within the Arduino called `analogWrite`. Example code of this method is shown above in figure 14. The `analogWrite` function generates a square wave that is varied in the void loop. The two things the function needs to run are a pin variable and a value to set the duty cycle. The values for this function must be between 0 and 255. A duty cycle of 100 percent occurs if the value is set at 255 and a value of 0 gives a duty cycle of 0 percent. To obtain a specific duty cycle, a value for `analogWrite` needs to be calculated as:

$$\text{duty cycle} * 255 = \text{analogWrite value}$$

The example code in figure 14 lines that change the duty cycle each time it completes one loop. The output of this code is shown in figure 15 below. Each square picture shows a different time. The pictures from left to right prove the functionality of the code changing the pulse width.

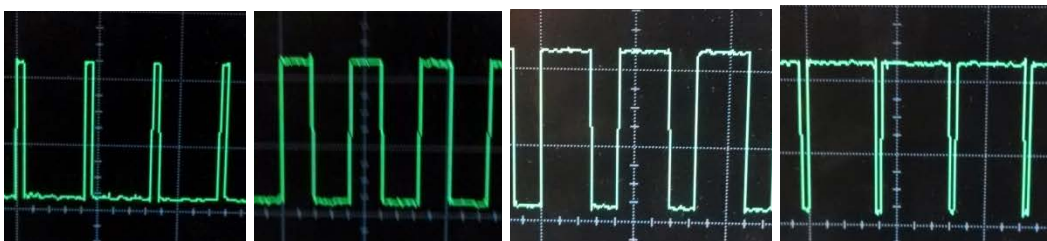


Figure 15. Changing Pulse Widths

The problem with the analogWrite function is that the frequency is at a fixed 490 Hz output. The compact power inverter needs to output at 60 Hz. Another problem with analogWrite is that when multiple signals are being output at once, the timing is off; the signals end up overlapping. This causes a short to ground on the H-bridge. This method was not used in the end, but was an important step in better understanding programming the Arduino.

The next method that was used was digitalWrite. This is another premade function within the Arduino. Unlike the analogWrite function, the digitalWrite function is only able to set a pin as a high or a low. This means that the digitalWrite also needed to use the delay function. Sample code showing this is provided in figure 16 below.

```
int led = 13;
// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);             // wait for a second
  digitalWrite(led, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);             // wait for a second
}
```

Figure 16. digitalWrite Arduino Code

The above code would output a set 50 percent square wave. To obtain a desired duty cycle, the two delays would have to be calculated as:

$$\text{delay time HIGH} / (\text{delay time HIGH} + \text{delay time LOW}) = \text{desired duty cycle}$$

This makes setting the pulse width more difficult and where the problem with digitalWrite is identified. The delays for both the HIGH and LOW sides need to be calculated. Taking the inverse of the the sum of these two times results in the frequency. With a changing pulse, the frequency still needs to remain the same. This is the method that the team used for the majority of the design process. The downside to using digitalWrite is that all of the delays had to be calculated and then each of the pulses needed to be implemented individually. Due to the number of pulses needed to create a cleaner signal, the code was quite long.

Using the digitalWrite code the team learned a significant amount about the gate drivers. While testing, the group learned about a few changes that had to be made to the code in order to achieve the output that was desired. The first problem that was discovered was that the digitalWrite function takes time to run. There was a four microsecond delay that occurred each

time the function ran. Adding a delay each time the function is run causes the effective output frequency to change. Fortunately, this delay was linear in nature and was easily corrected by recalculating the delay values for the pulses. The next problem faced by the team came as a result of solving the first problem. To correct the frequency, the team had to reduce all of the delay times. By doing this, some of the delays had become too small. The `digitalWrite` command had the four microsecond delay and with the smaller delay the `digitalWrite` functions were overlapping. This overlapping caused something called “shoot through”, whereby the high side and low side MOSFETs are both in the on position at the same time. The result of this is a direct path for current to travel from the power rail, through the MOSFETs, and to ground, which burns up the H Bridge. Adding “dead time” was the way to correct this problem. Dead time is a small amount of time between each pulse coming from the Arduino. Adding a small amount of time between changing signals helps ensure that even when a delay is required to turn off, it will not be on at the same time as another pin. This unfortunately changed the timing again, which had to be fixed.

The last problem encountered by the group was an issue related to the gate drivers. Upon testing the initial prototype it was found that the bootstrap capacitor intended to charge up to the VCC voltage and then be used for switching on the high side MOSFET was not able to store enough charge. Due to the quickly changing PWM signal driving the high side MOSFET the bootstrap capacitor needs to be discharged quite often. This bootstrap capacitor was losing its charge before all of the PWM signals had actually dropped to the low voltage. It turned out that when the high side signal goes low, the low side signal needed to turn on to recharge the capacitor. By manually switching all of the high and low side signals with `digitalWrite`, the code would not be quick enough to have the extra low side switching added in. Thus `digitalWrite` was not going to work for the compact power inverter.

The final programming method pursued was also the most computationally efficient. Port manipulation is very fast and allows for multiple pins to change in the same instruction. By assigning values in an array for the delay and dead time as well as an array to control the pin outputs, the length and size of code can be greatly reduced. This final coding scheme can be viewed at the beginning of appendix 3.

Subsection 3.1.4: DC/DC Conversion

In order to power the Arduino and the gate drivers, DC voltage from the 170 V supply needed to be stepped down to a more widely accepted DC voltage of 15 V to power the gate drivers and a different 5 V to power the Arduino. This was completed by using two linear voltage regulators; the adjustable TL783 and the static 5 Volt LM2940. These devices were used to step the 170 V input to 15 V and 5 V, respectively. The TL783 is able to adapt to several

different voltages of regulation by using a feedback resistor network between the output and adjust pins of the regulator. This feedback system allows the regulator to work within a large range of voltages that are very customizable for a specific application. The group chose to use two of these devices because the maximum voltage amount that can be dropped regulated between the output and input pins of the regulator is 125 V. In order for the group to get from 170 V to 15 V, the regulators needed to shed 155 V which exceeds the minimal drop rating for any single part. Because of this the group chose to use two regulators in series, the first regulator drops the 170 V input to a 100 V output and this is then routed into the second regulator that drops the 100 V to 15 V. Dropping this large voltage across the regulators causes the device to switch quite quickly and in turn causes it to get very hot very quickly. To mitigate this, heat sinks combined with thermal grease were applied to the TL783 regulators. A schematic of both TL783 regulators and the LM2940 can be seen in figure 17 below.

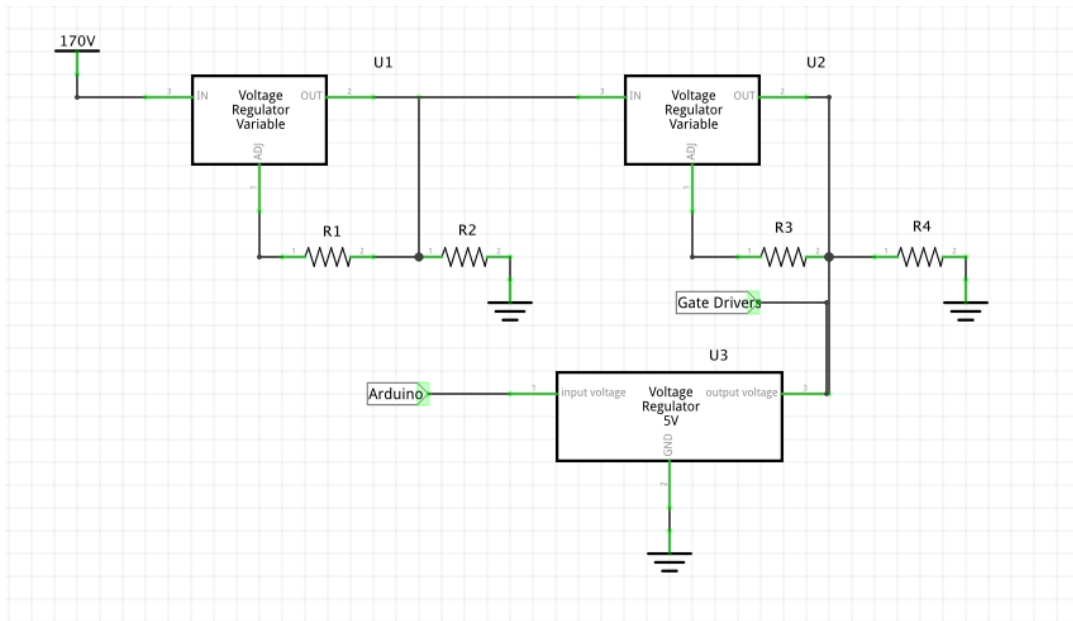


Figure 17. DC/DC Conversion Schematic

Section 3.2: Hardware Implementation and Photo Documentation

Subsection 3.2.1: PCB Design and Enclosure Layout

The enclosure was set up in a way to optimize the placement of the MOSFETs, filter inductor, and DC/DC converters for heat transfer and to ensure that inputs and outputs could easily connect to circuitry inside through the enclosure. The aluminum enclosure used by the group was 4.75 in x 4.75 in x 2.33 in. Working with a square base made fitting the PCB inside the case significantly easier. Upon searching vendor websites and catalogs, it was found that

there were few enclosures available that met the 40 in³ design constraint. This is the primary reason that the enclosure used in the final design was 53 in³.

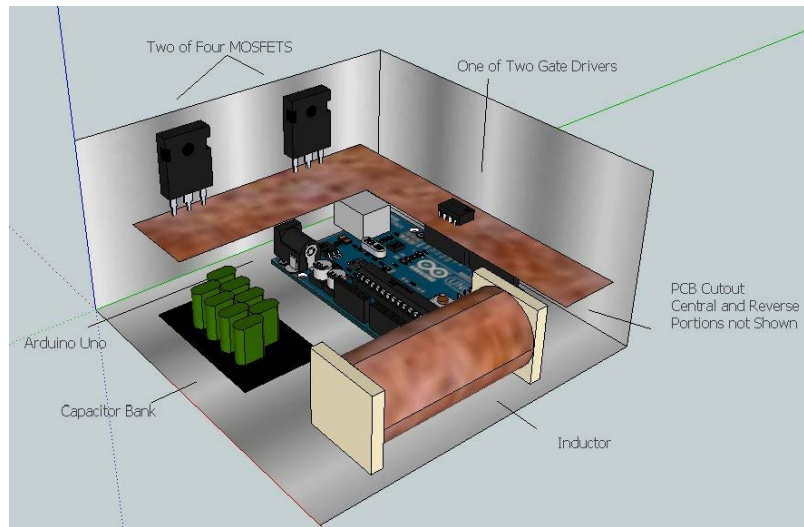


Figure 18. Preliminary Enclosure 3D Model

The bottom layer of the enclosure held the secondary circuitry. These were the output filter, made up of a large inductor and a capacitor bank and the Arduino controlling the pulse width modulation. This secondary circuitry was placed underneath the PCB which sat directly on top. A 3D model was constructed to help in optimizing component placement (figure 18). After adding the DC/DC circuitry it was necessary to remove the Arduino Uno development board and use just the microcontroller and its oscillator for an accurate clock. The final layout can be seen in figure 19 below.

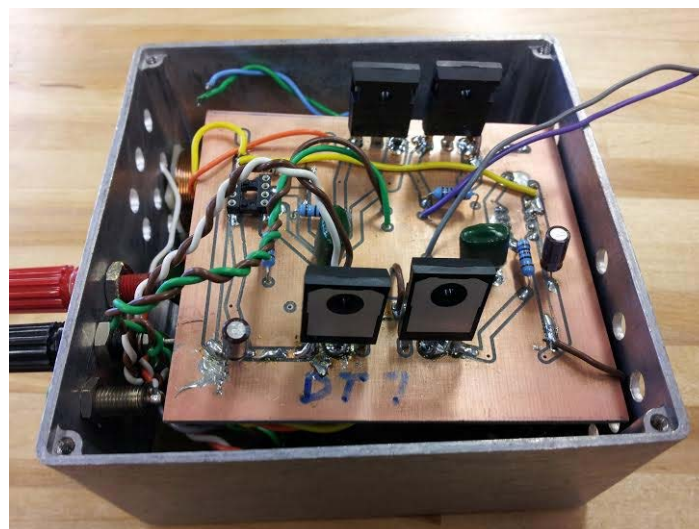


Figure 19. Final Enclosure Layout

Subsection 3.2.2: Heat Transfer Design

Due to the high voltage and high power requirements set in the design constraints, large amounts of heat are generated during inverter operation. To increase efficiency and safety, the team set out to design a system to dissipate the heat generated by the circuit. The largest generators of heat in the power inverter circuit are the MOSFETs of the H-bridge, the DC/DC converter, and the filter inductor. To dissipate this heat, aluminum heat sinks and an aluminum enclosure are used to conduct heat away from the circuitry. Both the heat sinks and aluminum enclosures are examples of passive heat transfer. These are systems that do not use forced convection to move fluids to transfer heat. In the final design, passive heat sinks were chosen for three reasons: price, size, and practicality.

The price of an active heat removal system can range anywhere from around five dollars for small fans commonly seen in computers (figure 20) to just under a hundred dollars for some liquid cooled systems. Compared to the price of a small heat sink of around two dollars, most active systems are at best slightly more expensive. Another reason that these active systems were not implemented in the design were the size constraints of the design. The heat sinks that would be placed on the transistors are significantly smaller than any active system that could be placed inside the enclosure. If active systems were added to the design, a larger enclosure would be needed to enclose the circuit. The last element for choosing the passive heat dissipation devices over the active heat dissipation systems was the practicality of design. The eventual intent of this power inverter is to be used for solar arrays, but the the technology can applied to other applications as well. If using this design in power generation it makes sense to not use any active systems that would be parasitic and use electricity such as fans or pumps. These are common on many active systems. For these three reasons discussed, passive heat dissipation elements were used in the design of the power inverter.



Figure 20. Computer Heat Sink

The aluminum enclosure surrounds the entire circuit and its control components. The MOSFETs used for H-bridge are mounted to the interior of the enclosure. Heat generated by the MOSFETs is passed through the transistor fin and conducted through the enclosure. To increase thermal conductivity, the small gap between the transistor fin and enclosure will be filled with thermal grease before mounting. Finned heat sinks will be placed inside the enclosure for the DC/DC converters. These will conduct heat from the circuit to the surrounding air. To remove this heat from the enclosure, several vents were drilled through two of the enclosure walls (figure 21). This created a convection flow bringing in ambient temperature air into the enclosure while removing heated air.



Figure 21. Convection Vents as Viewed from the Output

Discovered in testing, the inductor in the output filter generates a lot of excess heat when heavily loaded. To dissipate this heat, the inductor was placed inside a separate aluminum enclosure located in the bottom corner of the primary aluminum enclosure as seen in figure 22 below. This enclosure was constructed using available scrap aluminum.



Figure 22. Inductor in Separate Aluminum Enclosure

There are several future improvements for the heat design used in the compact power inverter. The two primary areas for improvement are the materials that are used and the enclosure design.

Aluminum was used for the enclosure used for the design and in the passive heat sinks inside of the enclosure. While aluminum has a thermal conductivity of 205 W/(mK) at room 25 degrees celsius, copper is a better thermal conductor. Having a thermal conductivity of 401 W/(mK) at 25 degrees celsius, copper's use in the design of heat dissipation devices will help dissipate heat more quickly.

Passive circuit components can also use different materials. The MOSFETs for example are primarily made with silicon carbide to increase the temperature operation threshold. These MOSFETs can also use a more efficient materials in their fin such as a gold thermal interface between the silicon carbide in the transistor and the aluminum fin. This will increase the amount of the heat the is conducted from the transistor. The problem with the addition of copper or gold is that it is more expensive than the conventional options in place now.

The second area for heat dissipation improvements is in the design of the enclosure. The aluminum enclosure used in the final power inverter design is only a box. To have the enclosure act as a more effective heat sink, it has to be designed like a heat sink. For the project, the group had a difficult time locating an enclosure that was extruded and had fins designed into the outside of the enclosure.

Chapter 4: Testing and Proof of Design

Section 4.1: Hardware Testing

There were several types of hardware tests that were done to prove the correct operation of the design. These tests were for the gate drivers, output filters, silicon transistors, and silicon carbide transistors. One of the initial tests completed was with the gate drivers. The first major issue faced by the team was outputting the signals that were sent in from the Arduino. On the initial prototype, the gate drivers would only output half of the PWM signals that were sent to it. This was because the capacitors could not store enough charge per cycle to supply the transistor gate for each side, and can be seen below in figure 23.

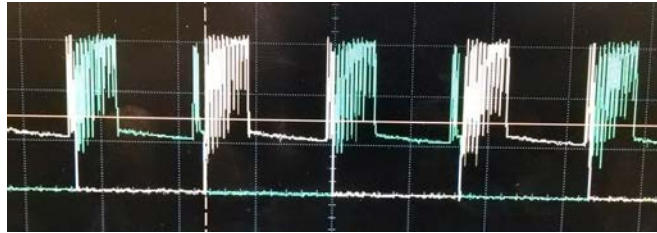


Figure 23. Bootstrap Capacitor Issues

This issue was first remedied by adding a larger bootstrap capacitor, which was only a temporary fix. This problem occurred while the team used the digitalWrite program that was mentioned in section 3.1.3. After switching to a port manipulation programming scheme, the group did not experience any more problems with the gate drivers and were even able to reduce the capacitor back to the original value used. Because port manipulation was used instead, the low side MOSFETs could be turned on, which recharged the bootstrap capacitor.

In order to meet the inverter's output design specifications, the group had to extensively design and test their output filters. The filter had signals coming from the Arduino and passing through the H-bridge to yield an output seen below in figure 24. To test this filter, silicon carbide transistors were put in place; this was important because the two transistors reacted differently when used with the filter.

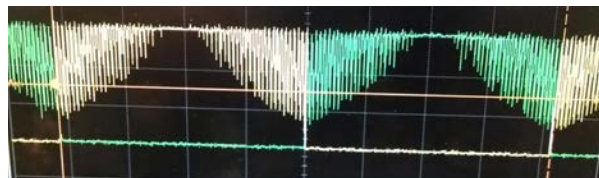


Figure 24. Output Before Filtering

The distorted signal seen in figure 24 is what the output from the H-bridge looks like prior to filtering. Several resonance calculations were performed to choose inductor and capacitor values for the filter. A series of calculations, PSpice modeling, prototyping and testing were performed to ensure correct operation of the filter. A clean, 60 Hz sine wave was apparent after the filter was installed, as shown in figure 25 below.

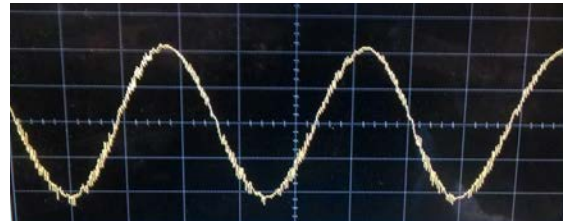


Figure 25. Final Filtered Output

To ensure that the initial prototypes did not damage the silicon carbide components, the group used standard silicon mosfets for most of their testing. The first silicon load test was done at 30 V, with four available 10 W load banks. This 40 W test was run for 60 minutes to determine how the circuit’s temperature responded to a sustained load. This test, seen in figure 26 below, was done without heat dissipation components. Using a handheld infrared thermometer, the temperature of the load and silicon mosfets could be measured. Table 4 contains a summary of these tests results.

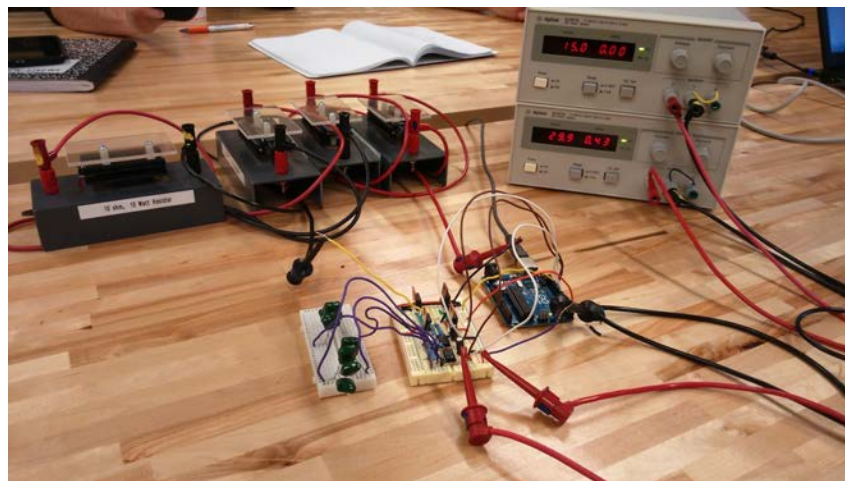


Figure 26. Test Set-up with Stable Resistor Load

Time (Min)	Load Temperature (F)	Si MOSFET Temperature (F)
------------	----------------------	---------------------------

5	112.0	77.6
20	143.0	78.8
35	144.6	78.8
50	142.6	78.6
60	144.6	79.0

Table 4. First Load Test Results

From this first load test, the group learned that circuit operation temperature reaches a steady state after twenty minutes. From this initial knowledge, the group was able to move on to silicon carbide tests inside the enclosure.

During the first test of this circuit, the power supply was set to output 170 V and 2.25 A for 382 W of power. The load used in this first silicon carbide test was a toaster. During this test, the silicon carbide transistors reached 96 degrees fahrenheit on a medium browning setting. From several independent tests, it was determined that at the medium browning setting the load was on for 92 seconds. While the transistors did not heat up too much, it was discovered that the inductor in the output filter severely increased in temperature. Knowing this early on allowed the group to redesign part of the heat dissipation system to better accommodate this unforeseen operation effect.

During initial testing of the DC/DC converters to step voltage down for both the gate drivers and Arduino, a number of observations were made. There are a total of three linear regulators used in this circuit. They step down 170 V to 100 V, 100 V to 15 V, and 15 V to 5 V. All three of these linear regulators rapidly increased in temperature. After five minutes of testing the linear regulators, the high voltage linear regulators had a temperature of 170 fahrenheit (See table 5). To mitigate this, heat sinks were added to these linear regulators (figure 26 below).

Time (Min)	170 → 100V Temp. (F)	100 → 15V Temp. (F)	15 → 5V Temp. (F)
5	170	174	96
10	185	186	98

Table 5. First Linear Regulator Test

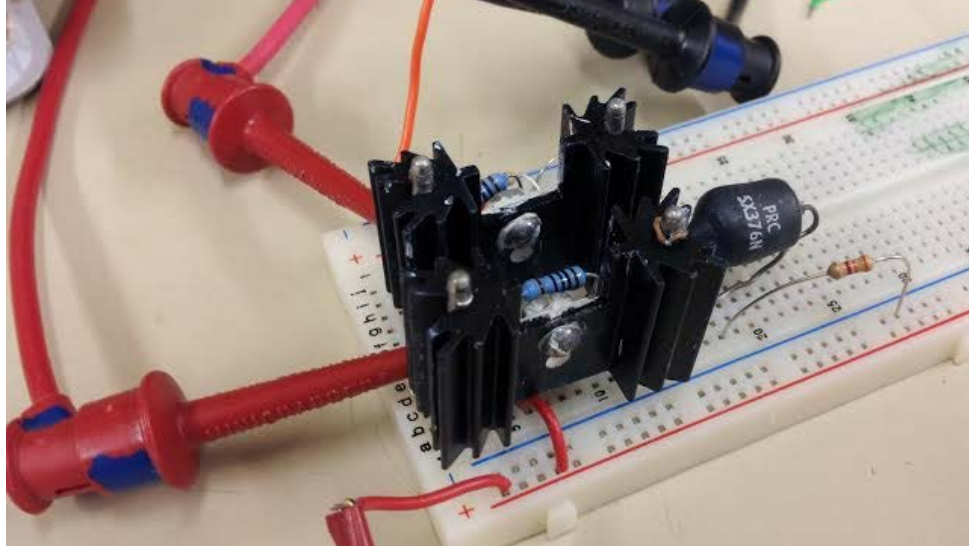


Figure 27. Linear Regulators with Heat Sinks Attached

As with all hardware tests, there was a variety of troubleshooting that took place. The first troubleshooting that occurred was with the bootstrap capacitors. The output from the gate drivers used for PWM was being clipped. This was a result of the capacitor being too small and the capacitor not being able to fully charge. To solve this problem, the value of the bootstrap capacitor was increased. This allowed the H-bridge to receive the correct size signal it needed.

The next problems that occurred in testing was with the transistors. The two primary issues that presented themselves in testing were shorting leads and the transistor becoming stuck in a high/on position. The first problem was first solved by examining the PCB. It was discovered that a specific trace had been damaged when soldering and this shorted two pins together. When testing, this resulted in the transistor becoming hot very quickly and faster than the other transistors. The second problem occurred as a result of the Arduino's program being interrupted when sending a high signal to a particular transistor. This high signal left the gate inside the transistor stuck in the high position so it needed to be reset. Changes in the operation of the Arduino solved this problem. A third problem occurred with the output filter. The first iteration of the output filter was too large to implement inside an enclosure, so the filter design was investigated again. A new design was made with a smaller size and once it was tested, the output was very distorted. So a third final filter was designed to better handle the types of load that would be connected to the power inverter.

The last major issues that commonly occurred in hardware testing were problems with the PCB. At the initial start of testing, the holes in the PCB used for mounting transistors were far too small for the lead width. These PCB ports were enlarged using a drill press. The next

problem that occurred with the PCB were the traces on the copper surface. During printing, the placement of these traces on some parts of the board were too close. This made soldering very difficult and some leads ended up soldered together for the first several tests until they were troubleshooted and the shorts were removed.

Section 4.2: Software Testing

Part of the danger associated with early versions of prototypes is their affinity for error. Because of this, IRF530N (silicon) transistors were used in place of the final silicon carbide transistors. The first test of the Arduino code functionality was done by outputting signals to light emitting diodes (LEDs). This provided a simple, visual representation of the signals destined for the gate drivers. A diagram of the layout can be seen in figure 28 below:

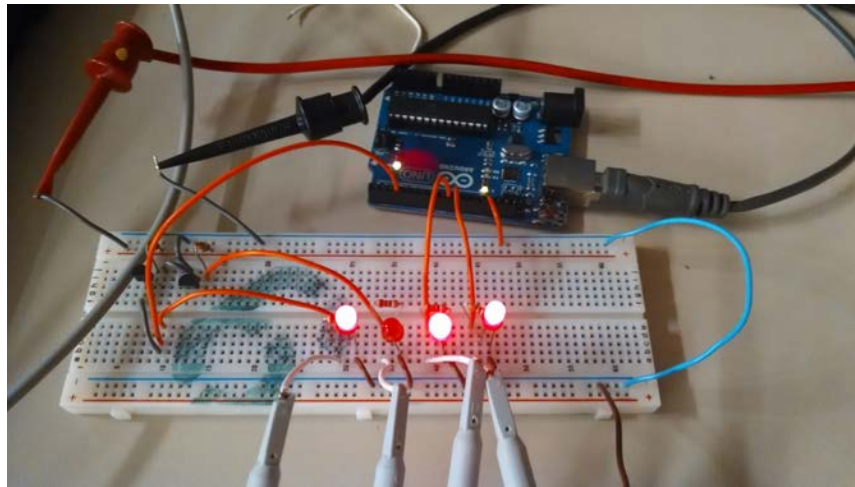


Figure 28. LED test

The LEDs to the outputs of the Arduino were connected to show that the correct code was in place. The group slowed the frequency to show that the signals were perceivable by the human eye; this helped to ensure proper timing of pins relative to one another. This also allowed for precise timing of the outputs. The two LEDs that corresponded with the low side signals were to be alternates of each other. They were either entirely on or off. The other two signals were for high side signals. These both varied in brightness according to the PWM scheme. The group was able to tune the timing by doing using this method and then continued with the rest of the circuit.

Chapter 5: Conclusions

Section 5.1: Final Design

The final design choice consisted of an H-bridge architecture that was built from 4 high power MOSFETs. These transistors act as switches that run current from a 170 V power rail back and forth through a filter and attached load; this simulates a positive and negative voltage swing. The transistors are switched on and off at 5.4 kHz in accordance with a pulse width modulation scheme. Once filtering is completed, a 60 Hz sine wave can be seen at the output. These pulses are created using an ATmega328 chip contained within the enclosure. Before reaching the transistors, these pulses are amplified and isolated at the gate driver; this prevents turning on 2 series transistors at once and melting parts of the circuit. Circuit components are mounted to a double-sided printed circuit board which was milled out by the ECE shop. Its dimensions were determined by enclosure and component sizes.

Section 5.2: Cost

From the budget provided in chapter 2 (table 3) it is evident that a reduction in size comes with a cost. Transitioning from silicon to silicon carbide MOSFETs results in a larger price. Silicon carbide components are relatively new to the market, and if compact, high power amplifiers and other high power uses become commonplace, their cost will greatly decrease; mass production also allows for the bulk purchasing power of all necessary circuit components. In the assembly of the inverter components there were some pieces of wire and small component wasted in prototyping, but nothing was wasted in the final assembly. The inverter was built to be green in its assembly, friendly for an individual performing maintenance, and the majority of the components capable of recycling at end of their life. Moving forward the monetary cost will decrease, human involvement time will decrease, and the environmental impact will be minimized.

Section 5.3: Timeline and Setbacks

The allotted 15 weeks of time to research, design, prototype, test, and perform a final build meant that timing was of the utmost importance. The original plan for the four phases of the project was elongated due to unforeseen issues. The initial goal was to have the final build completed in early November, but a buffer was included so that issues could appropriately be dealt with. After each setback, the GANTT chart (chapter 2, figure 5) was updated for appropriate time resource allotment. The final build was completed still in the month of November so that thorough final testing could be completed. In addition to an inability to obtain a 200 VDC supply, there was no time to build a switching DC/DC 200 V to 170 V converter;

however, the converter was designed and modeled for future implementation as shown in figure 29 below. The fundamental circuitry and testing was the priority and was completed.

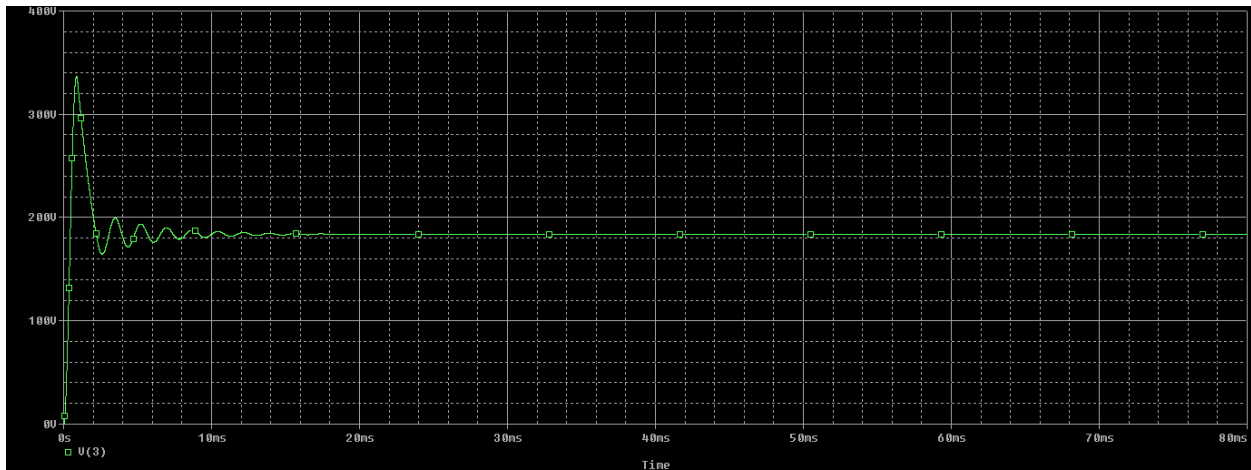


Figure 29. 200 Volt - 170 Volt DC-DC Converter Output

Section 5.4: Findings

Over the course of this semester, the team learned a great deal about the design process, working professionally as a team, what it takes to make a working consumer product, and inverters in general. The group also learned a number of valuable lessons pertaining to time management, the interconnected nature of projects, and what makes for a good presentation.

The group had a number of “failures” that are viewed by the group as lessons in and of themselves. The largest of these was having difficulty in predicting how much time certain parts of the project would take. Towards the end of the it was realized that there was not going to be enough time to build and test DC/DC converters. Similarly, FCC regulations were not thoroughly looked into, and the inverters electromagnetic effects on outside devices was not extensively tested. The final large failure was not being able to achieve the goal enclosure size of 40 cubic inches; instead, 53 cubic inches was used. The team felt pressed for time for the duration of the semester, but better planning towards the beginning could potentially have allowed for a mature PCB layout and smaller enclosure.

Many successes were also had, which included the completion of the project by design day, being able to power a device with the inverter, and having a safe product in the end. In addition, the large amount of learning that occurred alongside the extensive professional writing and presentation practice can be viewed as a success.

Section 5.5: Future Work

Several improvements could be made to this power inverter in terms of heat design. As discussed in detail in heat design, the use of different materials with higher thermal conductivity would greatly help dissipate heat. Also discussed in heat design is having an enclosure milled for our circuit. This would allow the circuit to better fit into an enclosure and more effectively dissipate heat.

Other improvements could be made to the parts chosen for the printed circuit board that would allow for a smaller and more compact footprint of the final circuitry. A majority of the parts chosen for this initial prototype were completed using DIP package devices or other larger and more user friendly parts. If a proper copper-milled circuit board were to be printed a full board could be populated using much smaller surface mount parts that could change the current size quite significantly. Using surface mount parts for the gate drivers that also included internal bootstrapping logic would allow for a significantly smaller footprint that would utilize more of what the current state-of-the-art has to offer for H Bridge gate driving.

The inverter could be significantly improved by using more efficient techniques for filtering the PWM output of the H-bridge. The resonant filter designed by the group is intended to meet the current spec set in place for maximum power transfer at the output of the inverter. While the current filter is a well designed device there could be a better match of a single large capacitor to a smaller, high wire diameter inductor than the one currently being implemented in the inverter output stage. With the output filter, the addition of an isolation transformer across the output leads of the inverter leading to the load ties could be considered for future work.

Appendix 1: Individual Technical Roles and Accomplishments

Philip Beard



Philip's technical role for this project started with the initial design phase of the inverter. As the group was beginning to research different inverter architectures Philip aided in the PSpice testing and modeling of different industry standards. After the initial testing was completed he then provided the group with relevant data to the design of the inverter and helped guide the group in selecting appropriate parts that would meet the needs of the proposed H-bridge design. As the group began prototyping different designs for the inverter Philip played a key role in keeping the PSpice models for the groups design up to date so that a working reference could be used as a benchmark for the prototyped models. Throughout the prototyping stage Philip aided the group by proposing different solutions to design problems and helping the group problem solve together.

Once the first prototype had been completed Philip worked with the group to come up with feasible solutions to utilizing gate drivers in the final design. After the group determined that gate drivers would solve some of the problems presented by the first prototype Philip selected the best part for this application and ordered them for the group. Once the group had completed the first working prototyped H-bridge the design process began for the distribution of power to the arduino and gate drivers within the circuit. Philip aided in selecting the proper parts for the linear regulators that could handle creating the desired line voltage to power all of the logic chips and created designs for all of the power distribution within the inverter. Using the final PSpice model Philip also aided in designing and prototyping the final LC filter that was implemented in the final design. Amongst the different design responsibilities in the group Philip was also present at group prototyping sessions to help contribute to troubleshooting any problems that may arise through the initial testing stages.

Jacob Brettrager



Jacob's technical role was initially identified as interfacing the digital and analog system components. However, he began his technical role in the project with helping to identify which architecture to select through assistance with PSpice modeling of the quantized-step system. After the group discussed and identified the H-bridge as the preferred method for inversion, he assisted with identifying and obtaining the necessary components. This included

obtaining the DC power supplies and infrared thermometer for testing as well as the silicon carbide MOSFETs. Once circuit elements were received, he helped in their assembly for prototyping and for the final build which included the combination of the microcontroller, gate drivers, and bridge. A major part after assembly was confirming each group members' work after placing components on a breadboard or soldering to a protoboard or printed circuit board. Double checking each other's work was very important as to prevent shorts and potential safety issues.

Jacob helped with a portion of the microcontroller programming. The pulse width modulation settings need to be changed to assure a 60 Hz output after adding the large inductor of the LC filter to the system. He was one of the two facilitators of the silicon prototype and silicon carbide build testing and worked to assure the integrity of the system when loaded. Assuring that the power supplies were set at the appropriate voltages and were never current limiting was important for high power testing and use. This was what led to the need to acquire a second, larger DC supply for its higher current output. After the PCB and enclosure were received, he helped to identify the optimal layout of the space demanding components - PCB, Arduino Uno, capacitor bank, and inductor. By 3D modeling the MOSFETs, gate drivers, microcontroller, and filtering components, Jacob identified the best component arrangement for the enclosure. He kept heat constraints in mind when laying out the modeled components.

Jack Grundemann



Jack's technical role in the group as heat transfer designer began in the initial proposal phase. He assisted in discussing the feasibility of each proposed design in regards to efficiency, physical size, and heat dissipation. To properly implement a heat dissipation design, testing of the team's prototype needed to take place. Before the circuitry was completed, Jack worked to bring the microcontroller to completion by assisting in the creation and implementation of the program. Primarily, the assistance was in the stratifying of arrays and delays to create a signal of proper frequency. Upon completion of the circuitry and program used for pulse width modulation, Jack worked extensively with testing. Jack designed tests specifically to examine how silicon and silicon carbide transistors heated when placed under varying loads at varying voltages. From these experiments, it was determined what types of heat dissipation could be implemented in line with design constraints.

In designing the heat dissipation systems, Jack worked to design cost efficient systems. For this reason, both the circuit enclosure and the heat sinks were made of aluminum. After working with PCB design for the physical dimensions of the board, the enclosure was ordered

from suppliers with parameters reflecting design constraints. Transistors then could be mounted to the wall to transfer heat through conduction. Jack worked to design two secondary heat dissipation systems; aluminum transistor finned heat sinks and passive convection vents. The layout of circuitry was carefully determined using Jack's knowledge from researching fluid dynamics and heat transfer. Jack worked with designing the layout inside the enclosure while working with team modeling to optimally place components. Lastly, Jack worked in circuit diagnostics. Through the testing phases, setbacks occurred with the group's power supply, PCB components, and loads. To properly run tests, Jack worked with the group's main circuit designers to quickly address any faults in system testing.

Stanley Karas



Stan's technical role began by working with the group to brainstorm, at a high level, which system architecture was to be implemented. This was done by looking at both existing inverter designs and drawing inspiration from past projects. Once the basic design was chosen, Stan assisted in the formation of programming logic to control the device, particularly the arrays which determined timing and levels. Next, troubleshooting was performed to help find minor issues in the circuit; this was an important step before moving on. The creation of a printed circuit board followed closely after, with close attention paid to board dimensions to fit into the enclosure and trace widths to handle high amperage. After printing, Stan checked each trace and connection to ensure proper translation from the printer; minute changes, such as the diameter of some vias, were made. The assembly and verification of the board with its components came next until, finally, it was operating successfully.

An important part of Stan's technical role was ensuring that all parts of the project would fit together. This was especially crucial in looking at the physical dimensions of the enclosure, printed circuit board, and heatsink design. The process of choosing enclosure dimensions was complicated by the fact that final circuit design was not decided upon until soon before the printed circuit board needed to be printed. Stan assisted in choosing an appropriately sized enclosure that would allow for minor adjustments to the overall project. Placement of the heatsinks was also closely connected with how the PCB was to be laid out.

Finally, Stan assisted the team in assembling and testing the final project. Due to the nature of the project, safety was of utmost concern in testing the final design. High voltages and very hot surfaces meant that each member had to look out for one another to ensure no one was

shocked, burned, or injured in some other way. General best practices, such as not touching the circuit during operation and wearing certified safety glasses, were some of the things Stan stressed most during testing.

Travis Meade



Travis's technical role throughout the duration of the project was extensive. To begin, he was involved in brainstorming initial circuit architecture. Travis assisted in building a number of prototypes; it was important to perform this task in iterations so that testing could be completed and unknown issues could be discovered before delving too deep into any particular design. The construction of these goes hand in hand with the troubleshooting of said prototypes, which is another technical contribution made by him. As a circuit grows in complexity, the level of troubleshooting required also increases; Travis was one of the main troubleshooters of the group and was able to solve issues with the power supplies, connections, and the circuit in general. Two of the major circuit components that required troubleshooting were the implementations of the gate driver and the printed circuit board. The gate driver required a fair amount of experimentation with regards to capacitor and resistor sizes, which is something Travis assisted with. For the printed circuit board, it was important to check each connection with scrutiny to prevent a short circuit when mounting components, particularly for the gate drivers and transistors. This helped to prevent setbacks.

The next major item Travis assisted with was testing, which the Little Box Challenge called for 100+ hours of. He was one of the group members that repeatedly set up the power supplies, microcontroller, and oscilloscope to ensure the device was running properly. Arguably the largest technical item Travis assisted with was programming the Arduino to perform pulse width modulation. This included brainstorming the logic as to how the microcontroller would do this, creating pseudo code to help break down the task into pieces, and physically writing several iterations of the code and loading them onto the Arduino. With this came troubleshooting the code to ensure pulses were the correct width, the frequency was a perfect 60 Hz, and it interfaced properly with the gate drivers.

As necessitated in project planning, role assignments were provided for development of the Gantt Chart. Table 6 on the following page details the design phases, their start and end dates, as well as roles assigned for each task.

1	Task Name	Duration	Start	Finish	Predecessors	Resource Names
2	Phase 1 - Initial Design Research	12 days	Wed 9/17/14	Thu 10/2/14		
3	Model IGBT H-Bridge design	12 days	Wed 9/17/14	Thu 10/2/14		Phil
4	Model Quantized-step MOSFET design	7 days	Wed 9/24/14	Thu 10/2/14		Stan,Jacob
5	Model 555-timer bridge design	7 days	Wed 9/24/14	Thu 10/2/14		Phil
6	Study 12 VDC to 120 VAC car inverter design	8 days	Tue 9/23/14	Thu 10/2/14		Stan,Travis
7	Phase 2 - Model Analysis/Prototype	12 days	Fri 10/3/14	Mon 10/20/14		
8	Select best design	7 days	Fri 10/3/14	Mon 10/13/14		
9	Compare efficiency	6 days	Fri 10/3/14	Fri 10/10/14	2,3,4	Phil,Stan,Travis, Jacob,Jack
10	Compare temperatures (if possible)	6 days	Fri 10/3/14	Fri 10/10/14	2,3,4	Jacob,Jack
11	Compare component costs	6 days	Fri 10/3/14	Fri 10/10/14	2,3,4	Phil,Stan,Travis, Jacob,Jack
12	Compare outputs	6 days	Fri 10/3/14	Fri 10/10/14	2,3,4	Phil,Stan,Travis, Jacob,Jack
13	Order Parts	4 days	Sat 10/11/14	Wed 10/15/14	8,9,10,11	Jacob,Travis
14	Microcontroller Programming	6 days	Fri 10/10/14	Fri 10/17/14		Jacob,Travis
15	Built the first prototype	3 days	Thu 10/16/14	Mon 10/20/14	12	Phil,Stan,Travis, Jacob,Jack
16	Phase 3 - Improve design	11 days	Tue 10/21/14	Tue 11/4/14		
17	Heat testing with Silicon	3 days	Tue 10/21/14	Thu 10/23/14	14	Phil,Stan,Travis, Jacob,Jack
18	Reprototype with Silicon Carbide	2 days	Fri 10/24/14	Mon 10/27/14	16	Phil,Stan,Travis, Jacob,Jack
19	Circuit layout and circuit enclosure	4 days	Tue 10/28/14	Fri 10/31/14	17	Stan,Jacob,Jack
20	Heat testing of enclosure	2 days	Mon 11/3/14	Tue 11/4/14	18	Phil,Stan,Travis, Jacob,Jack
21	Phase 4 - Final build and testing	12 days	Wed 11/5/14	Thu 11/20/14		
22	Design PCB	2 days	Wed 11/5/14	Thu 11/6/14	19	Stan
23	Print PCB	3 days	Fri 11/7/14	Tue 11/11/14	21	Phil,Stan
24	Assemble circuit, enclosure, heat sink	3 days	Wed 11/12/14	Fri 11/14/14	22	Phil,Stan,Travis, Jacob,Jack
25	Begin final testing	5 days	Sat 11/15/14	Thu 11/20/14	23	Phil,Stan,Travis, Jacob,Jack

Table 6. Gantt Chart Dates and Role Assignments

Appendix 2: References

- [1] Paul, Clayton R. *Introduction to Electromagnetic Compatibility*. 2nd ed. N.p.: Wiley, 2006. Print.
- [2]<http://www.cree.com/~media/Files/Cree/Power/Data%20Sheets/C2M0025120D.pdf>
- [3]<http://www.atmel.com/Images/doc8161.pdf>
- [4]http://arduino.cc/en/uploads/Main/Arduino_Uno_Rev3-schematic.pdf
- [5]<https://www.fairchildsemi.com/datasheets/FA/FAN7382.pdf>
- [6]<http://www.wholesolar.com/products.folder/inverter-folder/magnum-ms-4448PAE.html>
- [7]<http://www.circuitstune.com/2011/07/3000w-power-inverter-12v-dc-to-230v-ac.html>
- [8]http://www.entegronik.com/inverter/12v_240v_dc_inverter/2144542959-0.jpg
- [9]<https://www.littleboxchallenge.com/pdf/LBC-InverterRequirements.pdf>
- [10]<http://www.qsl.net/dg5sga/inverter.htm>
- [11]<https://www.littleboxchallenge.com/>
- [12]http://www.wpi.edu/Pubs/E-project/Available/E-project-042507-092653/unrestricted/MQP_D_1_2.pdf
- [13]<http://arduino.cc/en/Tutorial/PWM>
- [14]http://ewh.ieee.org/r5/dallas/sscs/slides/DCDC_Design_Slides_Ver2.pdf
- [15]<http://www.underservice.org/index.php?topic=241.0>
- [16]<http://forum.arduino.cc/index.php/topic,117425.0.html>
- [17]<https://www.fairchildsemi.com/application-notes/AN/AN-9052.pdf>
- [18]<https://www.fairchildsemi.com/application-notes/AN/AN-6076.pdf>
- [19]Eastman, Lester F., and Umesh K. Michra. "The Toughest." *IEEE XPlore*. IEEE, May 2002. Web. 29 Nov. 2014.
<<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=999791&tag=1>>.

Appendix 3: Detailed Technical Attachments

PSpice H Bridge Model Code

ECE480 H Bridge Second Revision (New PWM Controller)

VDCRAIL 1 0 30

VPDC1 16 0 40

** Global Pulse Controller

VG1 28 0 PULSE(1 90 8.3333M 0 0 8.3333M 16.6667M)

** Global Pulse Controller Complement

VG1C 29 0 PULSE(1 90 0 0 0 8.3333M 16.6667M)

** PWM for High Side Driver 1

VAC1 15 0 SIN(0 .99 60 0 0 0)

VSAW1 14 0 PULSE(-1 1 0 136.612U 136.612U 0 273.224U)

R1 16 11 3k

X1 15 14 16 0 11 LM139

** PWM for High Side Driver 1 COMPLEMENT

R3 16 25 3k

X3 29 11 16 0 25 LM139

** PWM for High Side Driver 2

VAC2 20 0 SIN(0 .99 60 0 0 180)

VSAW2 21 0 PULSE(-1 1 0 136.612U 136.612U 0 273.224U)

R2 16 22 3k

X2 20 21 16 0 22 LM139

** PWM for High Side Driver 2 COMPLEMENT

R4 16 27 3k

X4 28 22 16 0 27 LM139

```

** Low Side signal for Low Side Driver 3
VLO3 19 0 PULSE(0 10 8.3333M 0 0 8.3333M 16.6667M)
** Low Side signal for Low Side Driver 4
VLO4 18 0 PULSE(0 10 0 0 0 8.3333M 16.6667M)

M1 11 28 0 0 IRF530
M2 22 29 0 0 IRF530

RGD1 35 11 500
RGD2 36 22 500
RGD3 37 25 500
RGD4 38 27 500

MH1 1 35 30 0 IRF530
MH2 1 36 31 0 IRF530
ML1 30 37 0 0 IRF530
ML2 31 38 0 0 IRF530

LFILTER 31 32 10.67M
CBANK 32 33 3U
RDAMPLW 33 0 101000

RLOAD 32 30 20

.LIB ECE402.LIB
.TRAN 1U 4s0M 0 10u UIC
.OP
.PROBE
.END

```

PSpice LC Filter Characterizing

LC Filter Characterisation

```
VAC1 1 0 AC 170
LFILTER 1 2 10.67M
CBANK 2 3 3U
RDAMPLOW 3 0 101000
RLOAD 2 0 200
```

```
.LIB ECE402.LIB
.AC DEC 100 10Hz 80kHz
.OP
.PROBE
.END
```

PSpice DC-DC Buck Converter

Buck Converter Basic Outline

```
VDCIN 1 0 200
VPDC1 16 0 20
VSAW1 4 0 PULSE(0 210 0 0 0 11.166u 13.272U)
```

```
CIN 1 0 850U
RBUFF1 4 6 10K
MHIGH 1 6 2 0 IRF530
L1 2 3 212.5U
COUT 3 0 356.57U
D1 0 2 D1N967A
RLOAD 3 0 100k
```

```
.LIB ECE402.LIB
.TRAN 1U 80m 0 1u UIC
.OP
.PROBE
.END
```



```
B00001000, B00001001, B00001000, B00001010, B00001000, B00001001, B00001000,
B00001010, B00001000, B00001001, B00001000, B00001010, B00000000, B00001010,
B00000010, B00000110, B00000010, B00001010, B00000010, B00000110, B00000010,
B00001010, B00000010, B00000110, B00000010, B00001010, B00000010, B00000110,
B00000010, B00001010, B00000010, B00000110, B00000010, B00001010, B00000010,
B00000110, B00000010, B00001010, B00000010, B00000110, B00000010, B00001010,
B00000010, B00000110, B00000010, B00001010, B00000010, B00000110, B00000010,
B00001010, B00000010, B00000110, B00000010, B00001010, B00000010, B00000110,
B00000010, B00001010, B00000010, B00000110, B00000010, B00001010, B00000010,
B00000110, B00000010, B00001010, B00000010, B00000110, B00000010, B00001010,
B00000010, B00000110, B00000010, B00001010, B00000010, B00000110, B00000010,
B00001010, B00000010, B00000110, B00000010, B00001010, B00000010, B00000110,
B00000010, B00001010, B00000010, B00000110, B00000010, B00001010, B00000010,
B00000110, B00000010, B00001010, B00000010, B00000110, B00000010, B00001010,
B00000010, B00000110, B00000010, B00001010, B00000010, B00000110, B00000010,
B00001010, B00000010, B00000110, B00000010, B00001010, B00000010, B00000110,
B00000010, B00001010, B00000010, B00000110, B00000010, B00001010, B00000010,
B00000110, B00000010, B00001010, B00000000};
```

```
int i=0; //This is used for a counter to pull the correct values at the correct time.
```

```
void setup() {
```

```
  // put your setup code here, to run once:
```

```
  DDRB = B00001111; //This sets pins 8, 9, 10, and 11 on the arduino as an output.
```

```
}
```

```
void loop() {
```

```
  // put your main code here, to run repeatedly:
```

```
i=0; //Each loop needs to reset the counter variable.
```

```
do //We run a do-while command to run through all of the values for one period of our output.  
Then it gets looped back and does it again.
```

```
{
```

```
PORTB = PARRAY[i]; //This sets the pins either HIGH or LOW based on what was pulled  
from the bit setting array.
```

```
delayMicroseconds(DARRAY[i]); //This delays the HIGH or LOW pulse based on the values  
pulled from our delay array.
```

```
i = i+1; //This increments the counter so that the next pass in the do-while loop will pull the  
correct values from the arrays.
```

```
} while(i<332); //Each array has 332 elements. So as soon as we get above 332, it will break out  
of the do-while loop and then loop back to the beginning of the main loop.
```

```
}
```

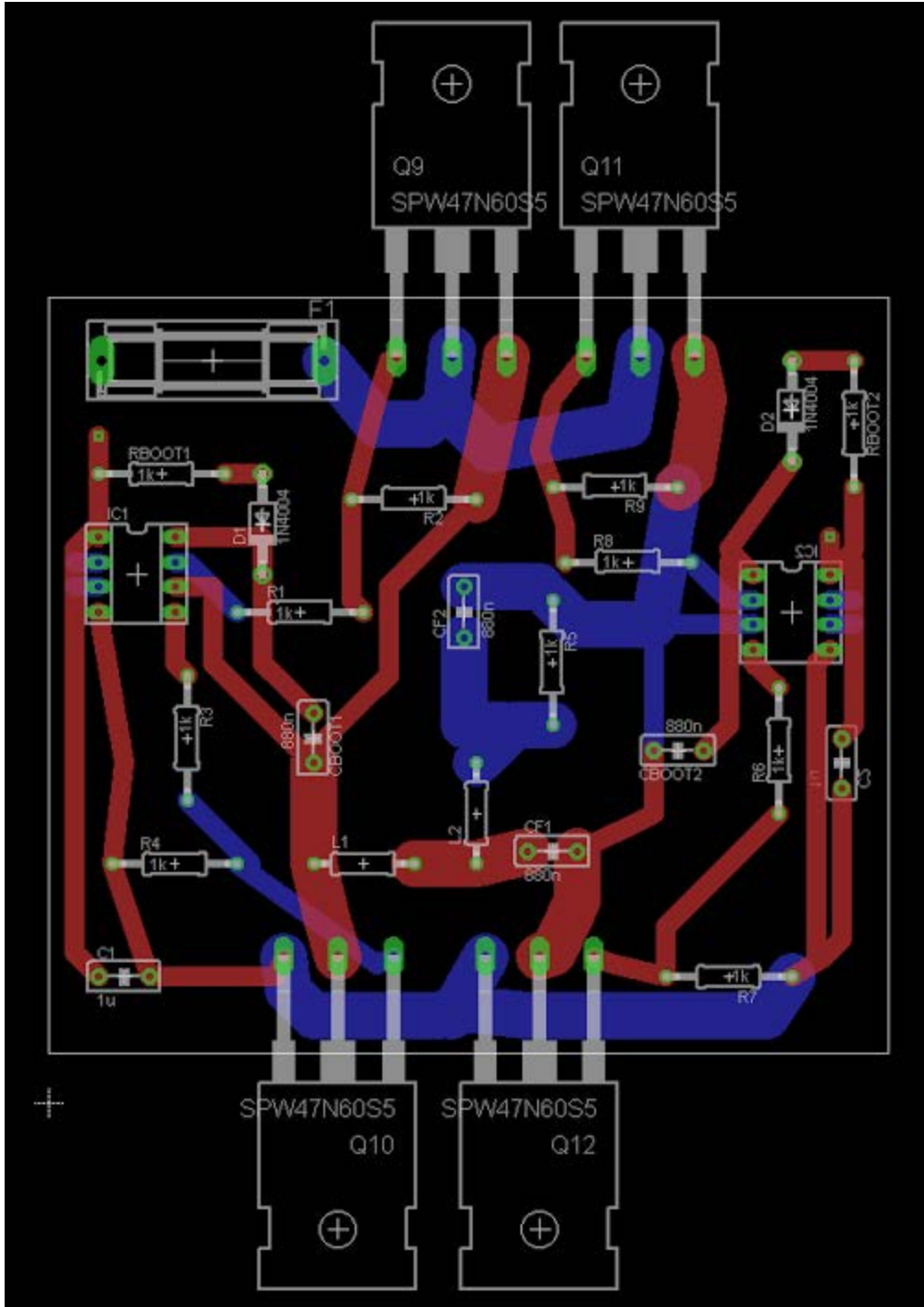


Figure 30. Final PCB Layout, 3" x 3.3"

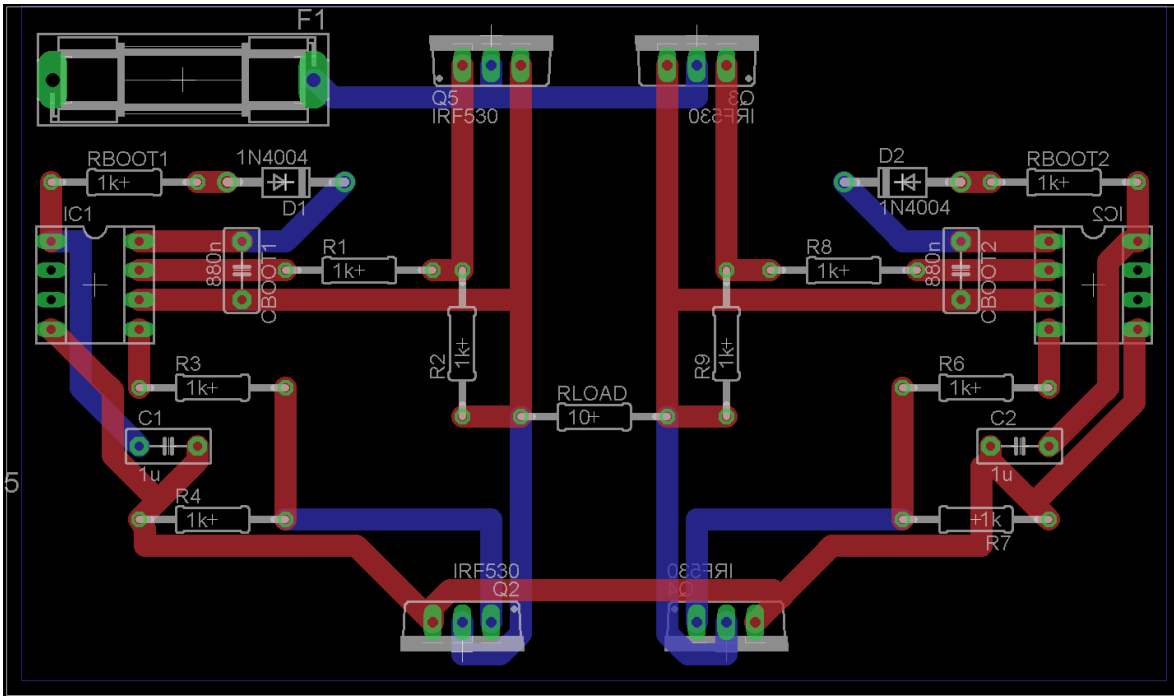


Figure 31. Initial PCB Layout

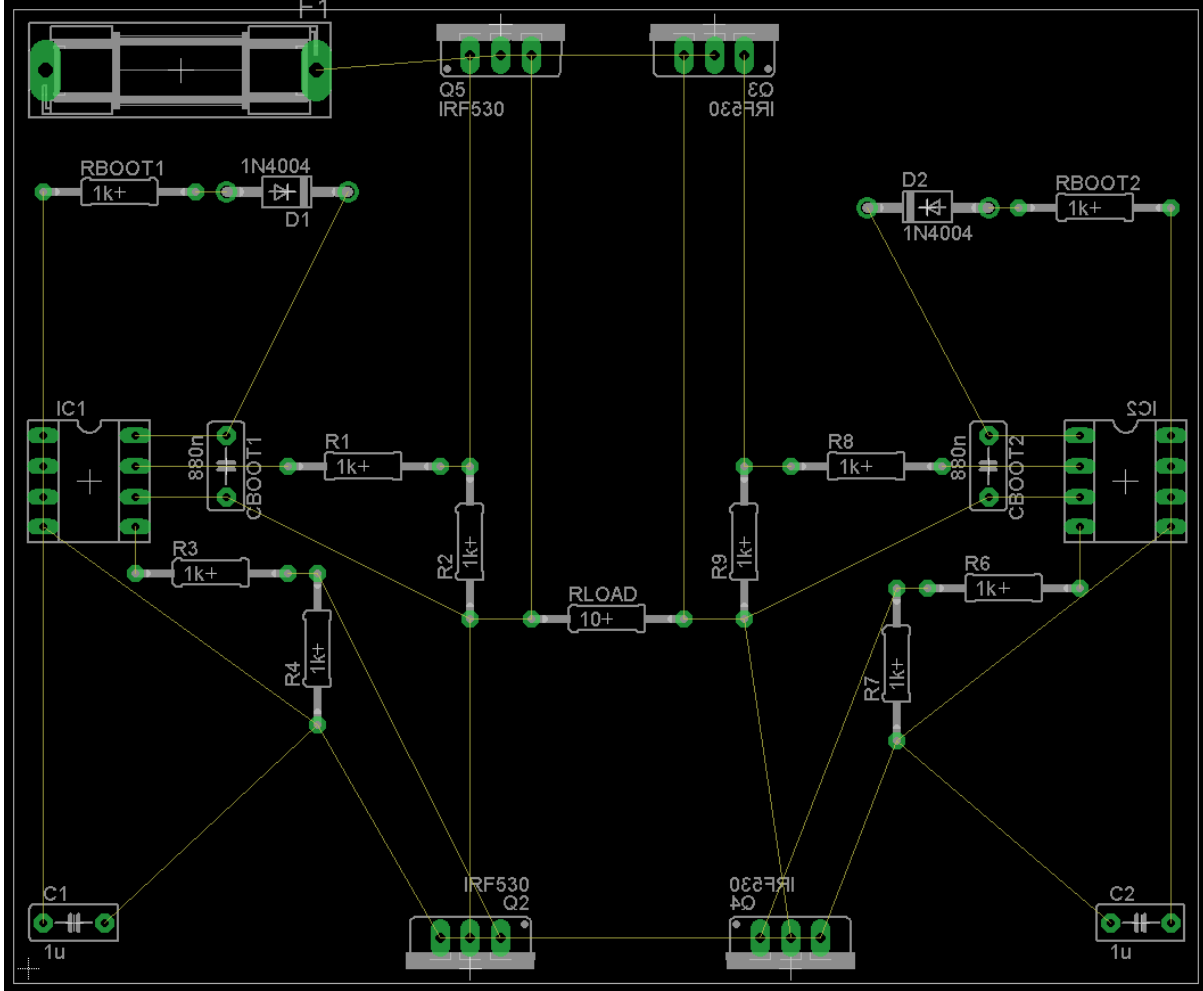


Figure 32. Initial PCB Schematic