

Design and Implementation of Power Electronics for an Electric Bicycle

A Thesis Presented for
The Bachelor of Science
Degree
The University of Tennessee, Knoxville

Kyle Goodrick

May 2017

© by Kyle Goodrick, 2017
All Rights Reserved.

Acknowledgements

I would like to thank my parents for their love and support throughout my life and especially during my undergraduate studies. I would also like to thank my friends for keeping me sane for the last five years and including me on our many adventures. Additionally, I would like to thank Jacob Dyer and Jared Baxter for being excellent teammates during this project. Finally, I would like thank Dr. Costinett for his wonderful guidance over the last year and a half; I have learned so much and grown as an engineer in ways that I would have never imagined. I can not thank him enough for the opportunities he has given me.

Abstract

During the spring semester of 2017 I worked with a team of two other University of Tennessee Students to design and construct an electric bicycle system capable of driving a 3-phase electric motor from a 25 V battery with 250 W of power. The design was implemented on a Printed Circuit Board and the final revision achieved an operating efficiency of 94 % at full power.

This senior honors thesis was completed in order to fulfill the requirements of the University of Tennessee Chancellor's Honors program as well as the Electrical Engineering course ECE 482: Power Electronic Circuits.

Contents

List of Tables	viii
List of Figures	ix
1 Introduction	1
1.1 Thesis Overview	1
1.2 Project Overview	2
1.3 Key Features of System	2
1.4 System Description	2
1.4.1 Functionality Overview	2
1.4.2 Boost Converter	4
1.4.3 Operating Ranges and Capabilities	4
2 Battery and Motor Characterization	7
2.1 Introduction and Objectives	7
2.2 Methods and Results	7
2.2.1 Motor Characterization	7
2.2.2 Battery Characterization	13
2.2.3 Vehicle Dynamics	20
2.3 Discussion and Conclusions	22
3 Microcontroller Programming	25

4	Open Loop Circuit Design	27
4.1	Introduction and Objectives	27
4.2	Methods and Results	27
4.2.1	Boost Converter Construction and Debugging	27
4.2.2	Boost Converter Testing	30
4.2.3	Efficiency Analysis	35
4.2.4	LTSpice Simulations	41
4.3	Discussion and Conclusions	42
5	Closed Loop Circuit Design	46
5.1	Introduction and Objectives	46
5.2	Methods and Results	46
5.2.1	Averaged Switch Modeling and Simulation	46
5.2.2	Switching Model and Simulation	47
5.2.3	Open-Loop LM5121 Implementation in Hardware	52
5.2.4	Closed-Loop Implementation	57
5.3	Discussion and Conclusions	59
6	Printed Circuit Board Layout	66
6.1	Introduction and Objectives	66
6.2	Daughter Boards	66
6.2.1	Si Daughterboard	69
6.2.2	GaN Daughterboard	70
6.3	Additional PCB Design Considerations	70
6.3.1	Capacitors	70
6.3.2	RCD circuit	71
6.3.3	Toshiba Motor Controller	71
6.3.4	Infineon Motor Driver	72

7	Motor Control Design	73
7.1	Introduction and Objectives	73
7.2	Methods and Results	73
7.3	Discussion and Conclusions	74
8	Final System Integration	75
8.1	Introduction and Objectives	75
8.2	Methods and Results	75
8.2.1	Experimental Results	75
8.3	Conclusions and Future Work	77
9	Conclusion	80
A	Appendix	82
A.1	MSP430 Code	82
A.2	Altium Schematics	89
A.3	Altium PCB Layout	96
A.4	Bill of Materials	107
Vita		110

List of Tables

1.1	Recommended Operating Conditions	5
1.2	Absolute Maximum Ratings	6
2.1	Motor Characteristics	8
2.2	Hall Sensor Analysis	16
2.3	Constant Values	22
2.4	Calculation Summary	22
4.1	MOSFET Characteristics	32
4.2	Measured Boost Converter Efficiency Analysis	38
4.3	Loss Summary Table	40
4.4	Simulated Boost Converter Efficiency Analysis	45
5.1	250W Open Loop Results	56
5.2	Compensator Network	60
5.3	Line Regulation	65
6.1	Si MOSFET Characteristics	69
6.2	GaN MOSFET Characteristics	70

List of Figures

1.1	Populated board ready to hit the road!	3
1.2	Detailed Block Diagram	4
1.3	Overview Block Diagram	5
2.1	$\lambda_m = 1.02V/(2\pi * 5.49\text{Hz}) = 29.6 \text{ mWb}$	9
2.2	$\lambda_m = 5.68V/(2\pi * 33.3\text{Hz}) = 27.1 \text{ mWb}$	9
2.3	$\lambda_m = 3V/(2\pi * 19.4\text{Hz}) = 24.6 \text{ mWb}$	9
2.4	$P = 2 * 23\text{Hz} = 46 \text{ poles}$	10
2.5	$P = 2 * 25\text{Hz} = 50 \text{ poles}$	10
2.6	$P = 2 * 24\text{Hz} = 48 \text{ poles}$	11
2.7	$P = 2 * 23\text{Hz} = 46 \text{ poles}$	11
2.8	$P = 2 * 25\text{Hz} = 50 \text{ poles}$	12
2.9	$P = 2 * 24\text{Hz} = 48 \text{ poles}$	12
2.10	Output voltages with $\omega_m = 4.876 \frac{\text{rad}}{\text{s}}$	13
2.11	Output voltages with $\omega_m = 1.38 \frac{\text{rad}}{\text{s}}$	14
2.12	Motor Example LTSpice model	14
2.13	Motor No Load LTSpice model	15
2.14	Motor DQ LTSpice model	15
2.15	Wheel Position θ_m for two revolutions	16
2.16	Six Subintervals for Hall Sensor Analysis	17
2.17	Given LTSpice Battery Model	17
2.18	Complete Discharge of the Battery at 3 Amps	18

2.19	Voltage response when battery current is taken from 3A to 0A	19
2.20	Final LTSpice Battery Model	19
2.21	Measured and Simulated Voltage for a discharge at 3A	20
2.22	Dynamic Performance of Battery	20
2.23	Free Body Diagram	23
3.1	Microcontroller Output Curve	26
4.1	Boost Converter Schematic	28
4.2	Initial Test: V_{out} (Pink), I_L (Yellow), V_{sw} (Blue), V_{PWM} (Teal)	31
4.3	Bootstrap Failure: I_L (Teal), V_{BO} (Yellow), V_{gsL} (Blue), V_{sw} (Pink), $V_{bootstrap}$ (Purple)	31
4.4	Waveform right before the failure: I_L (Blue), V_{HO} (Yellow), V_{BO} (Teal), V_{sw} (Pink), V_{gsH} (Purple)	32
4.5	Full Power: V_{sw} (Pink), I_L (Blue), V_{batt} (Teal), V_{out} (Yellow)	34
4.6	Low Side MOSFET Current: V_{sw} (Pink), I_{dL} (Blue), V_{batt} (Teal), V_{out} (Yellow)	34
4.7	High Side MOSFET Current (after decoupling cap): V_{sw} (Pink), I_{dH} (Blue), V_{batt} (Teal), V_{out} (Yellow)	35
4.8	MOSFET Thermal Image from FLIR Camera when $P_{out} = 250$ W. Left device is high side and right device is low side.	36
4.9	Gate Driver Thermal Image from FLIR Camera when $P_{out} = 250$ W.	37
4.10	Measured Efficiency vs. Input Power at 50 V with added points for 40 V, 34 V, and 30 V	39
4.11	Efficiency Sweep at 30 V output: V_{sw} (Pink), I_L (Blue), V_{hb} (Teal), V_{ho} (Yellow), V_{gsH} (Purple)	40
4.12	Efficiency Sweep at 34 V output: V_{sw} (Pink), I_L (Blue), V_{hb} (Teal), V_{ho} (Yellow), V_{gsH} (Purple)	41
4.13	Efficiency Sweep at 40 V output: V_{sw} (Pink), I_L (Blue), V_{hb} (Teal), V_{ho} (Yellow), V_{gsH} (Purple)	42

4.14	Simulated Efficiency vs. Input Power	43
5.1	Average Boost Converter Schematic	47
5.2	Time domain sweep of Output Voltage	48
5.3	Time domain sweep of Output Current	49
5.4	Frequency Response of converter with no loss modeling	50
5.5	Frequency Response of converter with loss modeling	51
5.6	Switching Boost Converter Schematic	52
5.7	Step Response of converter	53
5.8	Approximately 0 ns dead time with V_{bst} (light blue), V_{gsL} (dark blue), V_{goH} (yellow), V_{sw} (pink), and V_{gsH} (purple)	54
5.9	Approximately 50 ns dead time with V_{bst} (light blue), V_{gsL} (dark blue), V_{goH} (yellow), V_{sw} (pink), and V_{gsH} (purple)	55
5.10	Full power open loop operation I_{ind} (light blue), V_{gsL} (dark blue), V_{goH} (yellow), V_{sw} (pink), and V_{gsH} (purple)	56
5.11	Full Power Boost Operation in Current Loop Control with I_{ind} (light blue), V_{gsL} (dark blue), V_{goH} (yellow), V_{sw} (pink)	57
5.12	Initial Current Surge at Start Up	58
5.13	Initial Compensator Noise Waveforms with I_{ind} (light blue), V_{gsL} (dark blue), V_{goH} (yellow), V_{sw} (pink), and V_{gsH} (purple)	60
5.14	Failure from Switching the CSP and CSN pins with I_{ind} (light blue), V_{gsL} (dark blue), V_{goH} (yellow), V_{sw} (pink), and V_{gsH} (purple)	61
5.15	Failure from Sense Resistor Parasitics with I_{ind} (light blue), V_{gsL} (dark blue), V_{goH} (yellow), V_{sw} (pink)	62
5.16	Same Failure with FB and COMP pins shown in yellow and purple	62
5.17	Full Power Voltage Loop Control Boost Converter with I_{ind} (light blue), V_{gsL} (dark blue), V_{goH} (yellow), V_{sw} (pink)	63
5.18	Converter Reaction to Load Step from 50 Ω to 25 Ω with I_{ind} (light blue), V_{gsL} (dark blue), V_{out} (yellow), V_{sw} (pink)	64

6.1	PCB 3D Model	67
6.2	Daughterboard Connection	68
6.3	Timing chart for the Toshiba TB6551FAG	72
8.1	250W Boost Converter before PCB assembly with I_{ind} (light blue), V_{gsL} (dark blue), V_{goH} (yellow), V_{sw} (pink)	76
8.2	250 W Boost Converter after PCB assembly with I_{ind} (light blue), V_{gsL} (dark blue), V_{goH} (yellow), V_{sw} (pink)	77
8.3	Motor Driver Output with Hall Sensors with H_c (light blue), H_b (dark blue), V_b (yellow), V_a (pink), V_{ab} (purple)	78
8.4	150 W Boost Converter with GaN WBG Semiconductors with I_{ind} (light blue), V_{gsL} (dark blue), V_{goH} (yellow), V_{sw} (pink)	79
8.5	Attempt at ZVS Boost Converter with I_{ind} (light blue), V_{gsL} (dark blue), V_{sw} (pink)	79

Chapter 1

Introduction

1.1 Thesis Overview

Over the last semester I have worked alongside a team of my peers to design and construct an electric bike system capable of traveling at speeds up to 20 mph and of climbing small hills. The project was completed as part of electrical engineering course ECE 482: Power Electronic Circuits.

The project, and this thesis, is separated into seven parts:

1. Battery and Motor Characterization
2. Microcontroller Programming
3. Open Loop Circuit Design
4. Closed Loop Circuit Design
5. Printed Circuit Board Layout
6. Motor Control Design
7. Final System Integration

1.2 Project Overview

We will begin with an overview of the electric bike and then in the following chapters discuss each subsystem in detail.

1.3 Key Features of System

This electric bike system is designed for for a 25 V battery input and a three-phase motor output. Some of the features of the system include:

- Up to 250W Output Power at 94% Efficiency
- Dual Motor Voltage and Motor Duty Cycle Control
- Short Circuit Motor Protection
- Interchangeable Si and GaN Synchronous Boost Converter Daughterboards

1.4 System Description

1.4.1 Functionality Overview

The electric bike system starts with a 25V battery input to be used as the electrical power source. However, the battery itself is not enough to effectively spin the motor. A boost converter must be implemented to boost the battery voltage to drive the motor faster and have better functionality riding the electric bike. This boost converter is designed to boost the voltage anywhere from 26-50V depending on the amount of throttle. This boost converter also needs to be as efficient as possible in order to not lose any power in the transfer from the battery to the motor driver. This boost converter is regulated by a current-mode controller that controls the converter output voltage with two feedback loops: An inner current loop and an outer voltage

loop. Then the boosted voltage is used as the input to a motor driver/inverter to power the motor. This motor driver is controlled with a motor controller that uses the motors hall sensors and a reference input signal from the microcontroller to determine the required duty cycle for the Motor and converter output voltage. The throttle on the bike is the user control used to give the boost converter the reference output voltage and the motor driver reference duty cycle, where both together determine the power sent to the motor. A photo of the overall converter system assembled on a PCB and placed inside the converter box can be seen in Fig. 1.1. A basic block diagram of the power converter system highlighting the key features to turn the battery power into a spinning motor can be seen in Fig. 1.3.

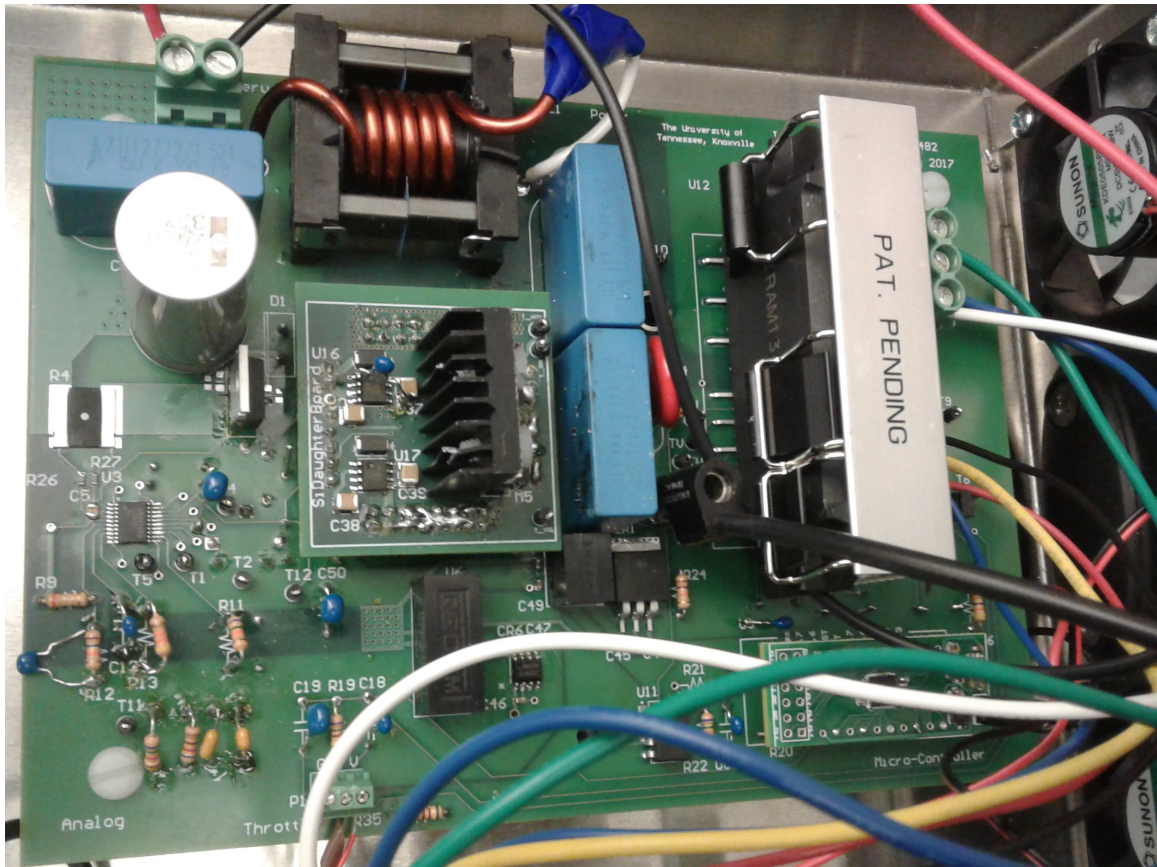


Figure 1.1: Populated board ready to hit the road!

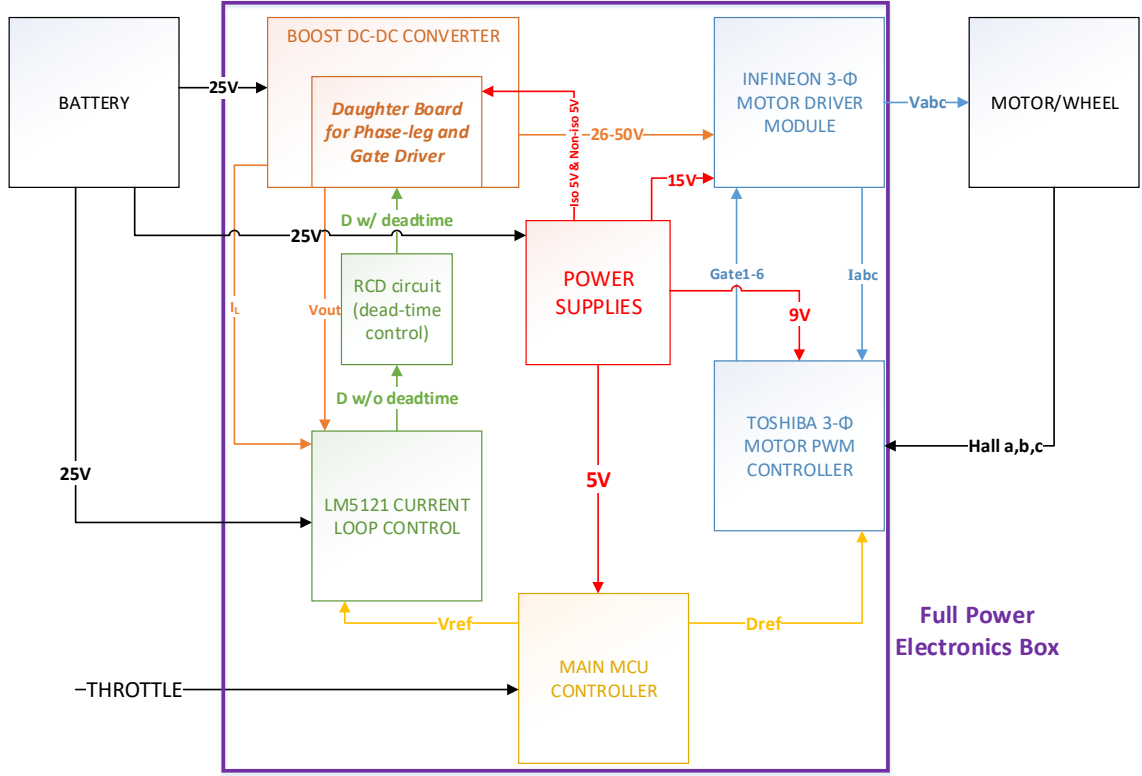


Figure 1.2: Detailed Block Diagram

1.4.2 Boost Converter

As mentioned previously, a boost converter is needed in the electric bike system to bring the battery voltage up to help the motor spin faster. A more detailed block diagram is also included in Fig. 1.2, which shows each major aspect of the converter system that will be explained in more detailed in the following chapters.

1.4.3 Operating Ranges and Capabilities

The recommended operating ranges are listed in Table 1.1 while the absolute maximum ratings for the system are listed in Table 1.2.



Characteristic	Symbol	Min	Typ.	Max	Unit
Input Voltage	V_{in}	20	25	28	V
Input Current	I_{in}	0.1	-	10	A
Input Power	P_{in}	10	-	250	W
Bus Voltage	V_{bus}	26	-	50	V
Temperature	Temp	-	25	-	°C

Table 1.2: Absolute Maximum Ratings

Characteristic	Symbol	Min	Max	Unit
Input Voltage	V_{in}	0	35	V
Input Current	I_{in}	0	15	A
Input Power	P_{in}	0	280	W
Bus Voltage	V_{bus}	-	70	V
Temperature	Temp	-20	75	°C

Chapter 2

Battery and Motor Characterization

2.1 Introduction and Objectives

Experiment 1 was focused on the characterization of the two fixed parts of the electric bike system, the motor and the battery. We performed experiments that enabled us to characterize each and develop simulation models in SPICE. We also performed a mechanical analysis of the system that gave us insight into the required power electronic performance characteristics that will be used later in the semester.

2.2 Methods and Results

2.2.1 Motor Characterization

The motor characterization process is described in detail in the following sections and summarized in table [2.1](#).

Winding Resistance

The winding resistance of the motor is given to be $150\text{ m}\Omega$.

Winding Inductance

The winding inductance of the motor is given to be 130 μH .

EMF Shape

The EMF shape is a sinusoidal as shown in the following figures of the motor electrical voltage.

Flux Linkage

To find the flux linkage, we decided to use the motor winding voltage equation for phase A and use the fact that there was no current to then simplify the equation to [2.1](#).

$$V_a = \lambda_m \omega_r \cos(\theta_r) \quad (2.1)$$

Then if you measure the voltage on the oscilloscope and acquire the peak value, that is the point where $\cos(\theta_r) = 1$. So for this measurement, $\lambda_m = V_{a,peak}/\omega_r$ where ω_r is also acquired on the scope. Figure [2.1](#), Figure [2.2](#), and Figure [2.3](#) show three different flux linkage calculations and the average between the three came out to $\lambda_m = 27.1 \text{ mWb}$.

Table 2.1: Motor Characteristics

EMF Shape	Sinusoidal
P	48
r_w	150 m Ω
L_w	130 μH
λ_m	27.1 mWb



Figure 2.1: $\lambda_m = 1.02V/(2\pi * 5.49Hz) = 29.6 \text{ mWb}$

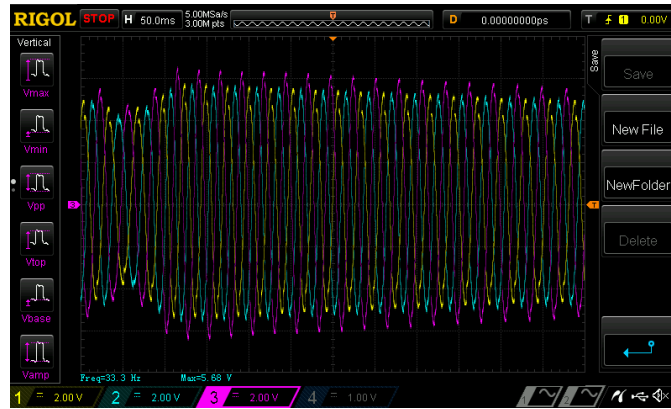


Figure 2.2: $\lambda_m = 5.68V/(2\pi * 33.3Hz) = 27.1 \text{ mWb}$



Figure 2.3: $\lambda_m = 3V/(2\pi * 19.4Hz) = 24.6 \text{ mWb}$

Number of Poles

For calculating the number of poles in the motor, we used equation 2.2 from the lab procedure.



Figure 2.4: $P = 2 * 23\text{Hz} = 46$ poles

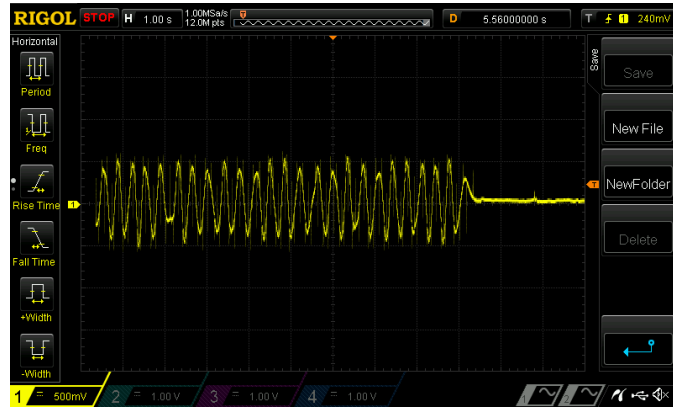


Figure 2.5: $P = 2 * 25\text{Hz} = 50$ poles

$$f_r = \frac{P}{2} f_m \quad (2.2)$$

So, we decided to spin the wheel one full revolution making $f_m = 1$, and measure on the oscilloscope one of the output voltages. Then multiplying the electrical frequency by 2 gives you the number of poles. After averaging three pole tests shown in Figure 2.4, Figure 2.5, and Figure 2.6, we calculated $P = 48$.

Hall Sensor Analysis

We also analyzed the hall sensors to discover the hall sensors can be used as the control to know when the input electrical voltage needs to be positive, zero, and



Figure 2.6: $P = 2 * 24\text{Hz} = 48$ poles

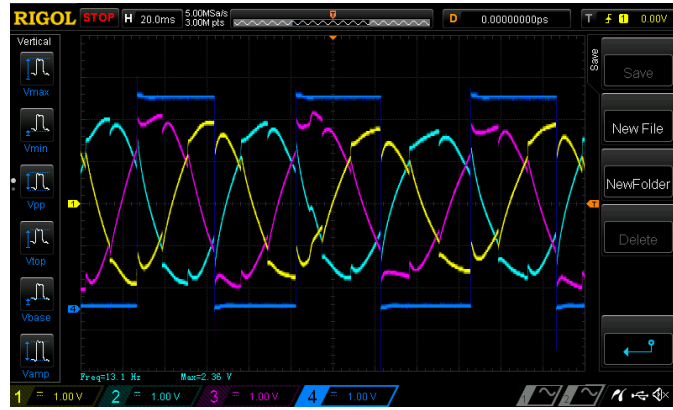


Figure 2.7: $P = 2 * 23\text{Hz} = 46$ poles

negative. It is worth noting that in Figure 2.7, Figure 2.8, and Figure 2.9 the pink waveform is for V_a , the light blue waveform is for V_b , and the yellow waveform is for V_c . Using Figure 7 as the example for V_a and Hall Sensor A, the hall sensor switches high when the voltage becomes the highest positive voltage and goes back low when the voltage becomes the lowest negative voltage. So from the low voltage signal level hall sensors, we can recreate what the input voltages for the motor need to look like.

LTSpice Simulations

To further verify our motor characterization, a LTSpice model was created using the motor parameters found. We used the LTSpice models given to us by Dr. Costinett and updated the parameters to our motor characterization. Figure 2.10,



Figure 2.8: $P = 2 * 25\text{Hz} = 50$ poles

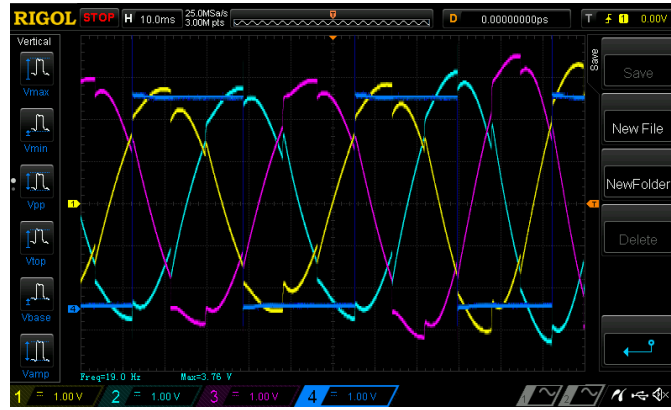


Figure 2.9: $P = 2 * 24\text{Hz} = 48$ poles

and Figure 2.11 are LTSpice outputs of the electrical voltage: V_a , V_b , and V_c as well as the motor position m . Figure 2.10 is comparing the LTSpice results to Figure 2.3 oscilloscope results and Figure 2.11 is comparing the LTSpice results to Figure 2.1 oscilloscope results. The LTSpice voltage values are a little smaller than the oscilloscope values but within a reasonable margin. Figure 2.12, Figure 2.13, and Figure 2.14 shows the LTSpice schematics used. Also, Figure 2.15 shows the wheel position θ_m for two complete revolutions with $\omega_m = 1.38 \frac{\text{rad}}{\text{s}}$. Figure 2.16 shows a LTSpice Simulation looking at V_A , V_B , V_C , H_A , H_B , and H_C that is also divided into six unique subintervals. It is worth noting that the hall sensor waveforms in Figure 2.16 are shifted on the y-axis so they don't overlap, as they normally output either 1 or 0. As shown in Table 2.2, these six subintervals of hall sensors determine

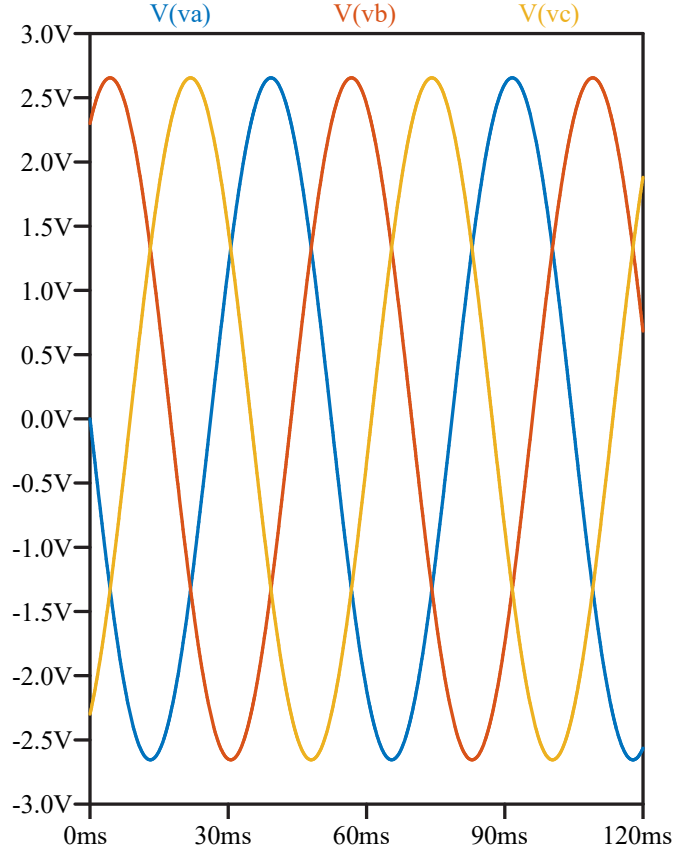


Figure 2.10: Output voltages with $\omega_m = 4.876 \frac{\text{rad}}{\text{s}}$

the trapezoidal input current for the motor drive and this control scheme will be used later in the semester.

2.2.2 Battery Characterization

For this portion of the experiment we looked to fully characterize and make an accurate LTSpice model for a Samsung INR18650-15M Lithium-ion battery cell. An example LTSpice simulation was given to us by Dr. Costinett as seen in Figure 2.17. Our goal was to test the battery and to find the parameters of the simulation were needed to accurately characterize the battery.

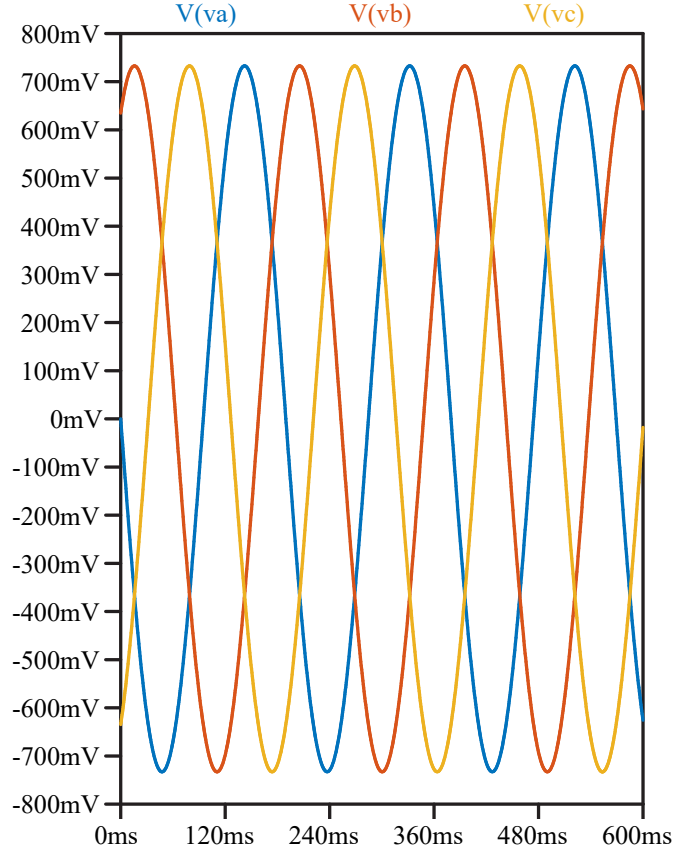


Figure 2.11: Output voltages with $\omega_m = 1.38 \frac{\text{rad}}{\text{s}}$

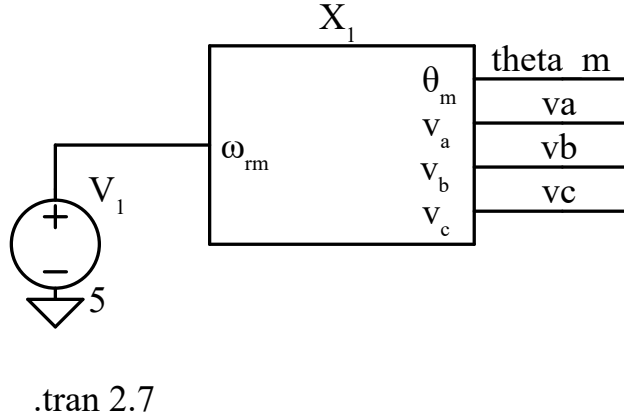


Figure 2.12: Motor Example LTSpice model

Battery Testing

For the battery characterization, we first took the fully charged Samsung INR18650-15M Lithium-ion battery cells and performed a complete maximum discharge at 3

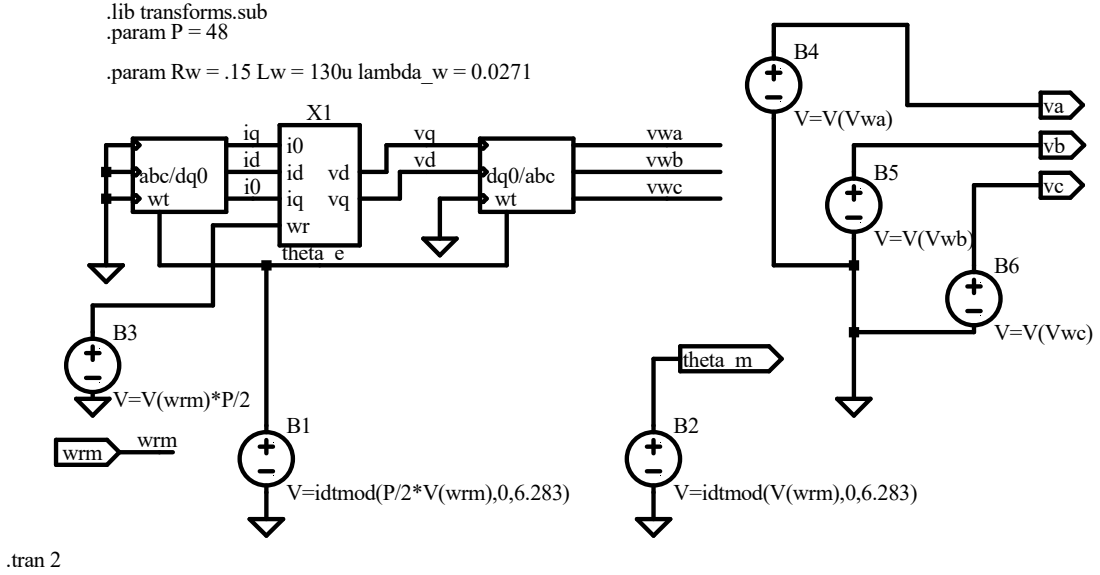


Figure 2.13: Motor No Load LTSpice model

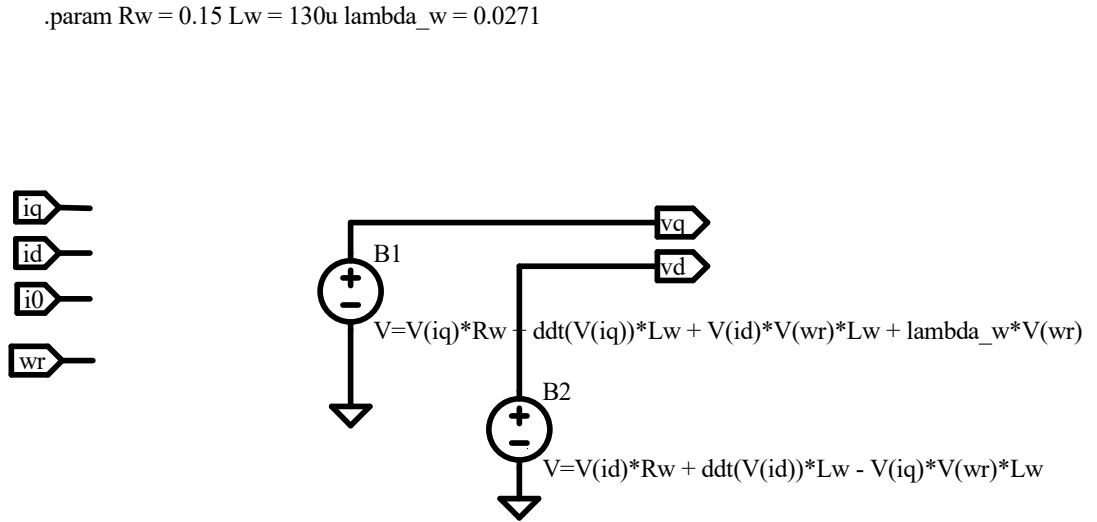


Figure 2.14: Motor DQ LTSpice model

Ampere. This allowed us to obtain the complete Voltage-vs-SOC curve for the battery in a short amount of time and completed the graph as seen in . We then recharged the battery to about 50% SOC and tested the battery's dynamic performance by running it through a current pulse train. We completed the pulse train test by discharging the battery at 3 Amps for 1 minute, 1.5 Amps for 2 minutes, 0.75 Amps for 4 minutes,

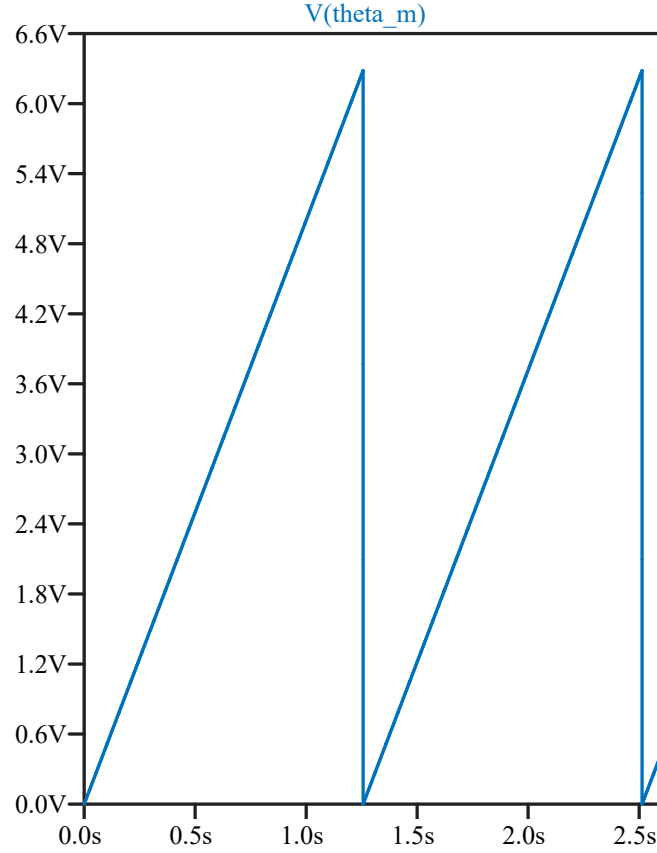


Figure 2.15: Wheel Position θ_m for two revolutions

Table 2.2: Hall Sensor Analysis

Trapezoidal Current						
	H_A	H_B	H_C	A	B	C
t_1	L	H	L	Neg	Pos	Zero
t_2	L	H	H	Neg	Zero	Pos
t_3	L	L	H	Zero	Neg	Pos
t_4	H	L	H	Pos	Neg	Zero
t_5	H	L	L	Pos	Zero	Neg
t_6	H	H	L	Zero	Pos	Neg

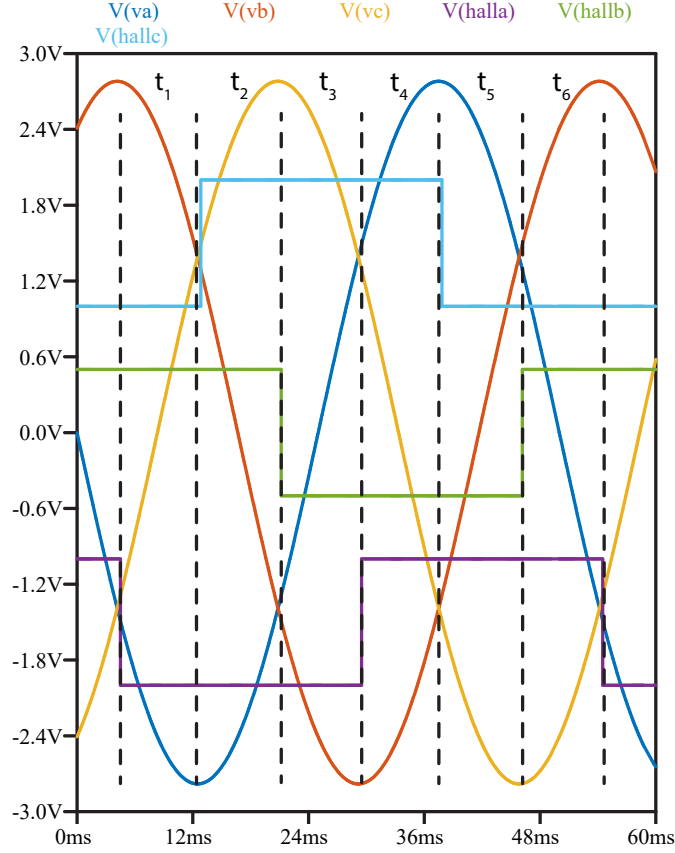


Figure 2.16: Six Subintervals for Hall Sensor Analysis

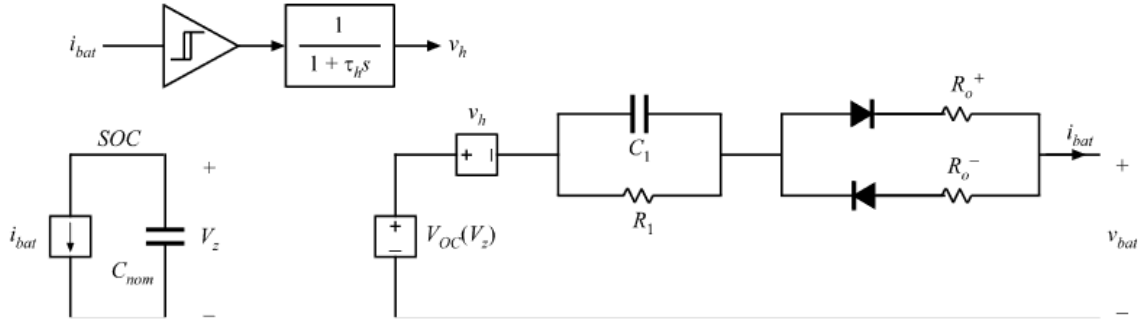


Figure 2.17: Given LTSpice Battery Model

and 0.375 Amps for 8 minutes. From the current pulse train data, we could obtain two different data sets of data points. First, we able to get the actual Voltage-vs-Time data itself. Second, we were able to calculate the series resistance of the battery by applying Ohms Law to the edges of the pulses in the pulse train. By subtracting the

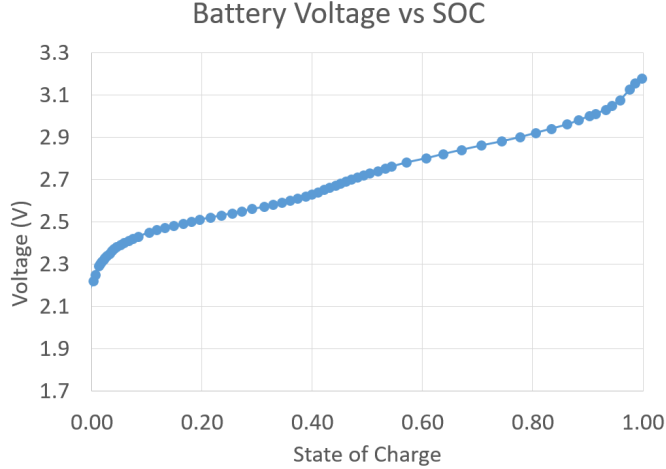


Figure 2.18: Complete Discharge of the Battery at 3 Amps

voltages just before and just after a discontinuity in the pulse train and dividing by the change of current that caused that discontinuity, we were able to estimate the series resistance of the battery to be 323 mOhms.

All of the data above was collected by reading measurements off a multimeter; however, in order to see if there if there was a need for a hysteresis time constant in LTSpice simulation, we then conducted a pulse test while logging the voltage of the battery on an oscilloscope as seen in Figure 2.19. From this test, we concluded that there was no need for the time constant because the battery voltage appeared to not have a large time constant associated with a discontinuity in battery current.

Battery Modeling

We started battery modeling in LTSpice by looking at implementing our results from the SOC and series resistance measurements. By only modeling these two aspects of a battery we were able to get very close to the measured data. The only other edit that we made was adding a small hysteresis magnitude $v_h = 0.3V$ in order to account for an offset in the dynamic performance of the battery. This then resulted in our final model as seen in Figure 2.20. The results of our model in comparison to the measured data can be seen in Figure 2.21 and Figure 2.22. Though this model

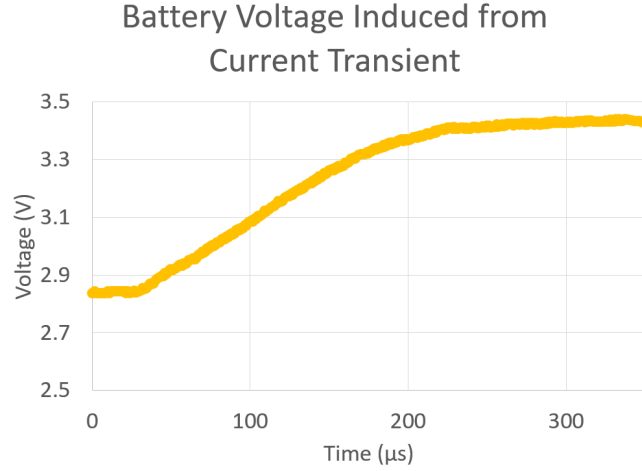


Figure 2.19: Voltage response when battery current is taken from 3A to 0A

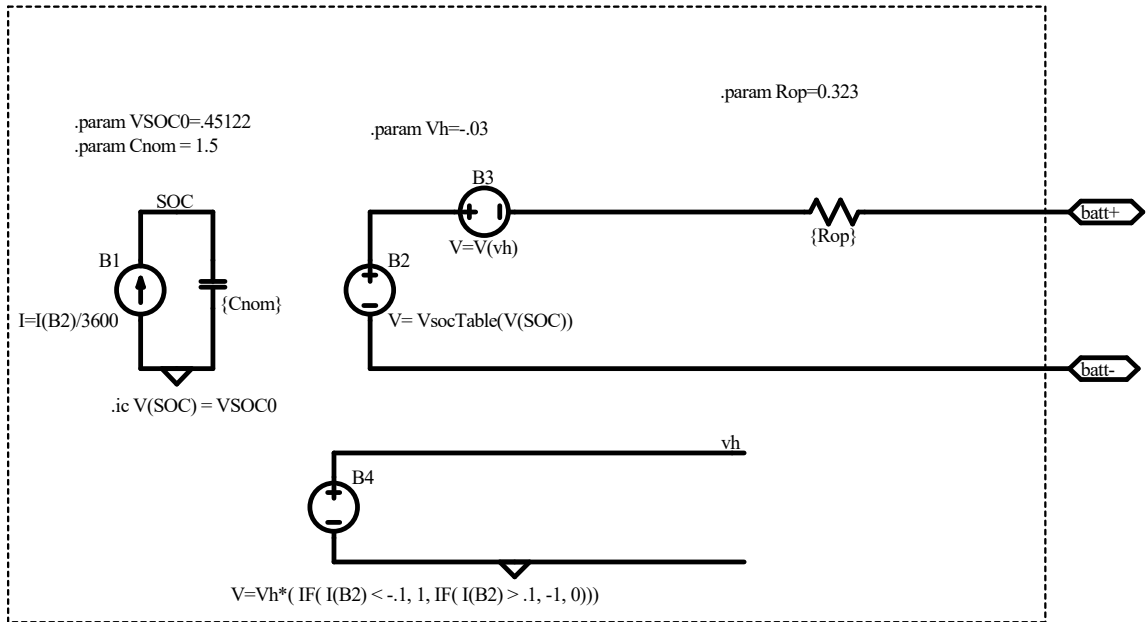


Figure 2.20: Final LTSpice Battery Model

does not include all of the possible parameters that we could have simulated for the battery, the simulations show that it comes close to the measured data. Therefore adding the hysteresis time constant and diffusion parameters would only have added complexity to the without giving a more accurate model.

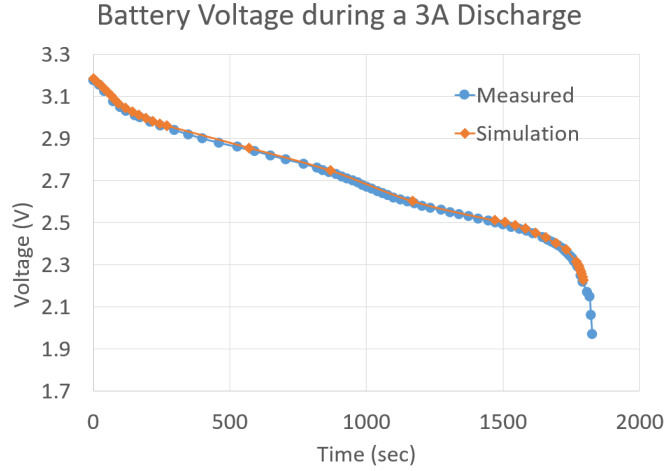


Figure 2.21: Measured and Simulated Voltage for a discharge at 3A

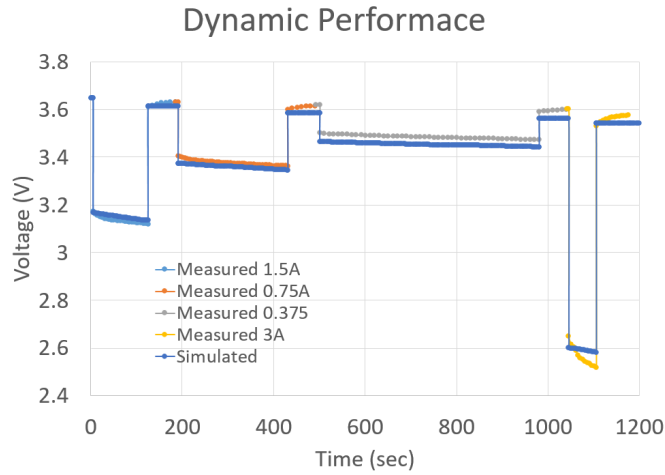


Figure 2.22: Dynamic Performance of Battery

2.2.3 Vehicle Dynamics

In the fourth part of the experiment we looked at the mechanical requirements for the overall system. We examined two separate cases that demonstrate the desirable performance characteristics of the bike. The first is the flat ground situation where the bike should be able to achieve a speed of at least 10 mph and the second is a hill climbing situation where the bike should be able to achieve a speed of 5 mph on a hill with a 3% grade. A basic free body diagram of the system can be seen in [Figure 2.23](#).

In our model we are including the traction force of the bicycle, the air resistance, the rolling resistance, and the weight of the bike and user. At a constant speed the force of traction F_t will be equal to the forces acting in the opposite direction. Setting theses equal to each other results in 2.3.

$$F_t = \frac{1}{2}\rho A_v C_d V^2 + Mg \sin \theta + C_r Mg \cos \theta \quad (2.3)$$

Where the constants are summarized in table 2.3. Once we have found the force required we can find the necessary mechanical torque using 2.4.

$$\tau_m = F_t r_w \quad (2.4)$$

We can then use 2.5 to find the current through the motor.

$$I = \frac{4}{3P} \frac{\tau_m}{\lambda_m} \quad (2.5)$$

The required mechanical power can also be found with 2.6.

$$P_m = \tau_m \omega_m \quad (2.6)$$

Finally we can solve for the peak applied back-emf voltage using 2.7.

$$V_{motor} = r_{motor} I + \lambda_m \omega_r \quad (2.7)$$

Where,

$$\omega_r = \frac{P}{2} \omega_m \quad (2.8)$$

The results of these equations for each of the two required scenarios as well as a third desired scenario can be seen in Table 2.4.

Table 2.3: Constant Values

ρ	$1.204 \frac{\text{kg}}{\text{m}^3}$
A_v	0.5 m^2
C_d	0.65
M	250 lbs
g	$9.81 \frac{\text{m}}{\text{s}^2}$
C_r	0.013
r_w	35 cm

Table 2.4: Calculation Summary

	Hill Climb	Flat Ground	Max Speed
V	5 mph	10 mph	20 mph
Grade	3%	0%	0%
F_t	48.8 N	18.4 N	30.1 N
τ_m	17.1 N·m	6.43 N·m	10.5 N·m
I	17.5 A	6.58 A	10.83 A
ω_m	$6.39 \frac{\text{rad}}{\text{s}}$	$12.8 \frac{\text{rad}}{\text{s}}$	$25.6 \frac{\text{rad}}{\text{s}}$
P_m	109 W	82.1 W	269 W
V_{motor}	6.78 V	9.31 V	18.3 V

2.3 Discussion and Conclusions

The results of this lab give us a good indication of what type of design considerations need to be made for the power electronics in our bicycle. As seen in table 2.4 our power electronics will need to supply, at minimum, 110 W to meet both the flat ground and hill climbing requirements. To meet our top speed stretch goal of 20 mph

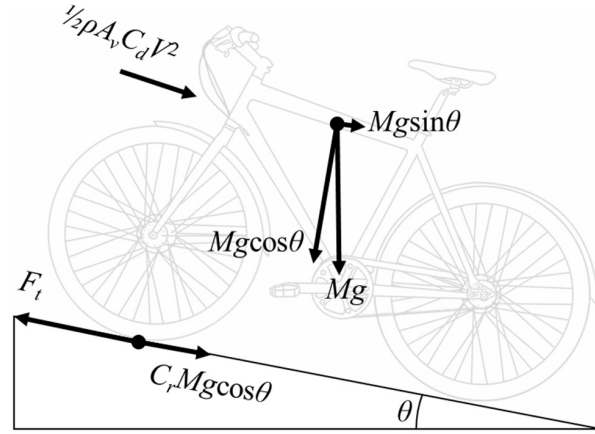


Figure 2.23: Free Body Diagram

on flat ground we will need nearly 270 W, which is slightly outside of the max power specification for the converter. It is quite close though, so a slight reduction in the riders weight and surface area may allow us to reach our desired top speed.

The peak motor voltage requirement will primarily affect the inverter design. In order to meet or speed requirements we will need to design it to have a wide variable output with ranges from 0 V to nearly 20 V.

As seen in section 2.2.2 the battery has a very fast time constant on the order of several hundred microseconds. This is likely to be much faster than any motor time constants as acceleration times will be on the order of several hundred milliseconds to several seconds.

The battery back will have over and under voltage limits that are equal to the number of series cells times the minimum and maximum limits of each individual cell. In our case there are seven series cells so the minimum safe voltage will be 17.5 V and the maximum voltage will be 29.4 V. This means that we will need to design our initial stage to handle a range of input voltages. It also means that as the battery SOC charge decreases the maximum input power will decrease as we can not increase the current to compensate for the decrease in voltage. It could be possible to dynamically modify the switching frequency to decrease the current ripple as the state of charge

decreases which would allow us to increase the average current without exceeding the maximum battery current. This is likely non-trivial though, but it could be an interesting idea to explore in the final experiment.

Chapter 3

Microcontroller Programming

The microcontroller used for this project was the Texas Instruments MSP430F5172. The purpose of this microcontroller is to coordinate the entire electric bike and allow for system level control. The microcontroller reads the throttle's voltage as an input and outputs two PWM signals to control the Motor Duty Cycle and the Boost Output Voltage. These PWM signals are converted to DC voltages by an RC filter network. The throttle voltage ranges from 0.8 V to 4.2 V and the microcontroller outputs follow the curve shown in Figure 3.1. Figure 3.1 can be mathematically described by equations 3.1 - 3.3. The first third of the throttle movement increases the motor duty cycle from 5% to 95% and the last two thirds increase the converter output from 26 V to 50 V. To read the throttle voltage on the MSP a resistor divider is used to bring the 5 V signal down to 3.3 V. A capacitor network is further used to filter the throttle input to prevent rapid changes in the throttle signal. The full MSP430 code can be seen in Appendix A.1.

$$D_{motor} = \begin{cases} 5\% & \text{if } V_{Throttle} < V_{min}, \\ (V_{Throttle} - V_{min}) \left(\frac{90\%}{V_{mid} - V_{min}} \right) + 5\% & \text{if } V_{min} < V_{Throttle} < V_{mid}, \\ 95\% & \text{if } V_{Throttle} > V_{mid}. \end{cases} \quad (3.1)$$

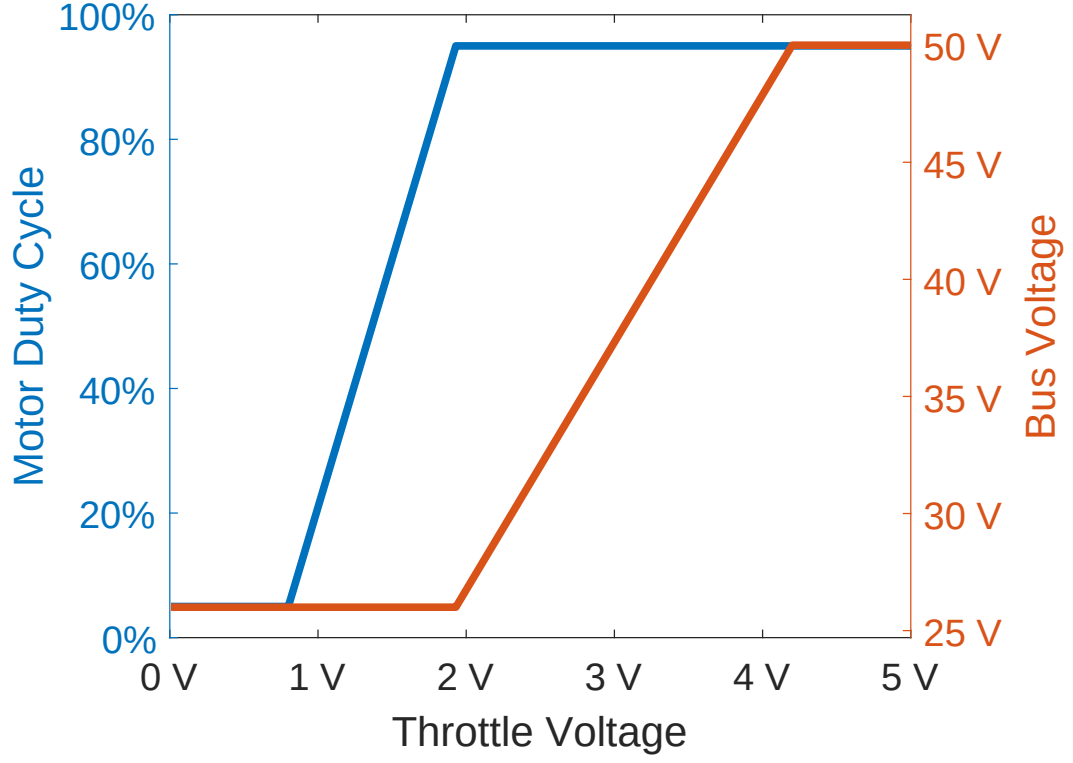


Figure 3.1: Microcontroller Output Curve

$$V_{Bus} = \begin{cases} 26 \text{ V} & \text{if } V_{Throttle} < V_{mid}, \\ (V_{Throttle} - V_{mid}) \left(\frac{24 \text{ V}}{V_{max} - V_{mid}} \right) + 26 \text{ V} & \text{if } V_{mid} < V_{Throttle} < V_{max}, \\ 95\% & \text{if } V_{Throttle} > V_{mid}. \end{cases} \quad (3.2)$$

Where,

$$V_{mid} = \frac{V_{max} - V_{min}}{3} + V_{min} \quad (3.3)$$

Chapter 4

Open Loop Circuit Design

4.1 Introduction and Objectives

Experiment 3's objective was to design and implement the boost stage of the bike's power electronics using open loop, duty cycle control. Through this process we analyzed a synchronous boost converter using average modeling equations and Spice simulations. We also designed an inductor using one of several given cores and chose a MOSFET from a provided list. Once finished with the design process we implemented the converter using basic construction methods. After construction the converter was debugged, and we went through several phases of design iteration. Finally, the converter was tested and characterized.

4.2 Methods and Results

4.2.1 Boost Converter Construction and Debugging

The initial design of the boost converter used soft switching to improve switching efficiency. This design utilized zero voltage switching to reduce reverse recovery and IV Overlap losses when the low side device turns on, shown as ML in Figure [4.1](#). To achieve soft switching at full power, the inductor needed to be very small such

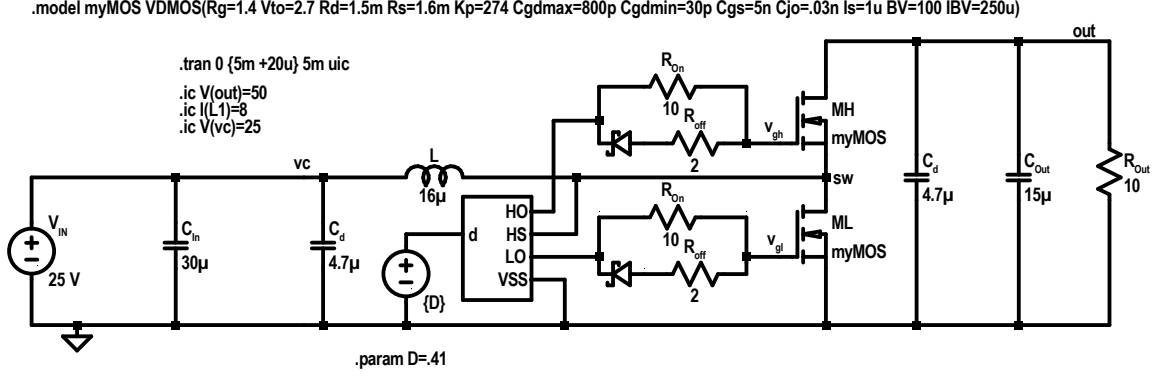


Figure 4.1: Boost Converter Schematic

that the inductor current would barely go below 0 A when discharging. This negative current would discharge C_{oss} after the high side channel is turned off and allow the low side device to switch at zero volts. The goal with this design would also be to optimize dead time such that at the moment the low side body diode would conduct, the channel would be turned on preventing any diode conduction or reverse recovery losses.

Initial Design

The inductor core chosen for the converter was the ETD 29 due to its exceptional k_{core} value provided by the lab specifications. As we intended to use a very small inductance a large core was also unnecessary. The ETD 29 was chosen over the EFD 25 because of the better max power performance at high frequency per the datasheet as well as less winding resistance from the round centerpost. The inductance was then set at $1.81 \mu\text{H}$ using 2 turns of AWG 10 wire and an air gap of $200 \mu\text{m}$.

Once the inductance was chosen we used 4.1 - 4.3 to determine the number of turns and the air gap.

$$L = \frac{N^2}{\mathfrak{R}_g} \quad (4.1)$$

Where,

$$\mathfrak{R}_g = \frac{l_g}{\mu_0 A_c} \quad (4.2)$$

and A_c is a property of the inductor core.

To ensure the inductor did not saturate we must use 4.3:

$$I_{max} = \frac{B_{max} l_g}{N \mu_0} \quad (4.3)$$

Where B_{max} was from the device datasheet and,

$$I_{max} = 10 \text{ A} + D \frac{1}{2f_{sw}} \frac{25 \text{ V}}{L} \quad (4.4)$$

Calculations for the inductor were confirmed using Ferroxcube Soft Ferrites Design Tool and measured using an LCR meter. The MOSFET was chosen to be the CSD19535KCS 100 V N-Channel NexFET Power MOSFET. This MOSFET was chosen based on a $Q_g R_{on}$ FOM as seen in Table 4.1. To reduce the input voltage ripple, a large capacitance of 30 μF along with a decoupling capacitor of 4.7 μF was chosen for C_{in} . C_{out} was then chosen to be 15 μF along with a decoupling capacitor of 4.7 μF . All of these values were confirmed by using simulation.

Debugging and Redesign

After initially ensuring that the gate driver and MOSFETs worked individually, an output resistance of 250 Ω was placed on the converter while the input voltage was brought up to 25 V. The waveforms are shown in Figure 4.2. A quick check of the efficiency of the converter at this point showed that $\eta \approx 25\%$. After turning the converter off, we noticed that the inductor was hot and determined that this was likely caused by a large core loss in the inductor. A new inductor was made using 5 turns instead of 2; however, the air gap was significantly increased to keep $L \approx 2 \mu\text{H}$. After the inductor change was made, we started operating the boost converter again and noticed the bootstrap capacitor was not holding a consistent 12V causing failure

as seen in Figure 4.3. Therefore, we decided to increase the bootstrap capacitance from $0.1\ \mu\text{F}$ to $1.0\ \mu\text{F}$. This corrected the problem and allowed the high side device to properly switch.

As a precaution, we replaced both FETs and the gate driver. However, after getting the input voltage up to around 16 V, the gate driver experienced over-current. Since we had experienced multiple failures at this point, we decided to simplify our design and implement a simple hard-switched boost converter. So, we replaced the inductor once again. This time we went with a much larger inductance therefore causing the boost converter to hard switch. The final inductor value was $16\ \mu\text{H}$ and was constructed using 6 turns of 10 AWG wire with a $254\ \mu\text{m}$ air gap. Also as a precaution, we changed the layout on our Vector-board to minimize any parasitics and create better solder connections. After the inductor was replaced and layout was changed, the converter was once again brought up to full power. Once again, the gate driver blew once the input voltage reached around 16 V as seen in Figure 4.4. We were at a loss for what the problem could be at this point since the waveforms looked as they should. We checked to be sure all of the passives and the two FETs on the boost converter were operating perfectly and then realized that it must be the micro-controller causing the gate driver to fail this whole time. Instead of determining the duty cycle using a resistor pot, we hard set our duty cycle in the micro-controller's programming. After replacing the gate driver, the boost converter worked all the up to a 25 V input. We believe that this micro-controller was receiving an erroneous signal from the resistor network causing the duty cycle to jump, thus causing the gate driver to fail.

4.2.2 Boost Converter Testing

Full Power Testing

Once a working boost converter was achieved, the load resistance on the electronic load was decreased to obtain full power at the output as detailed in Section III of

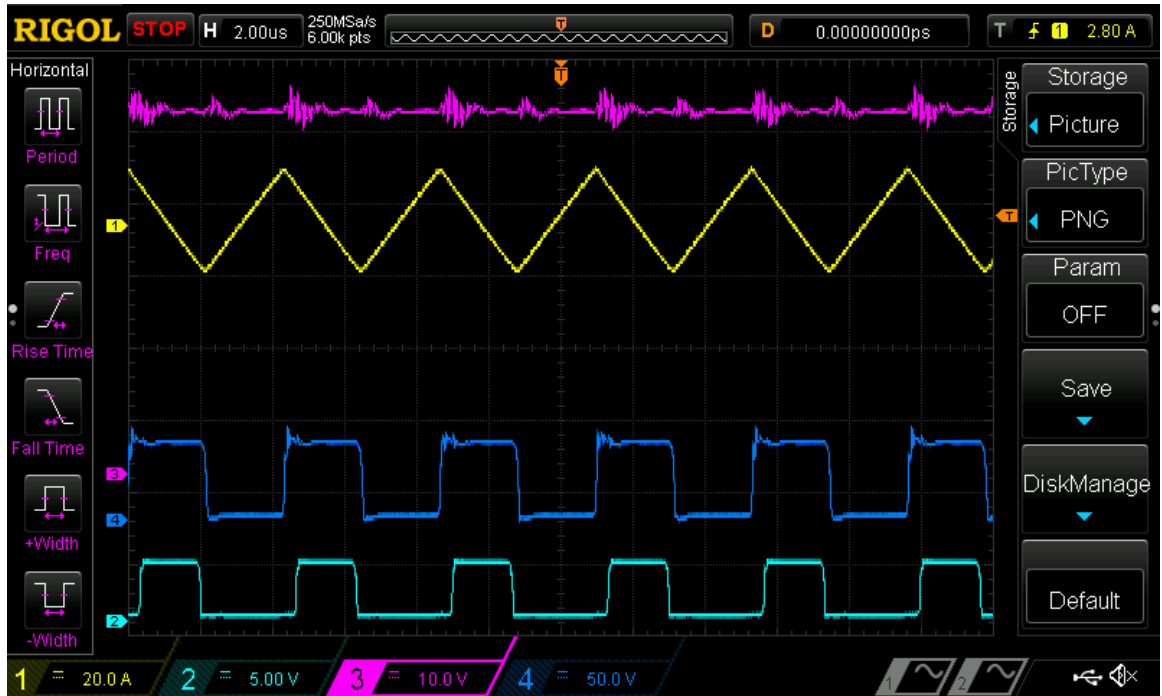


Figure 4.2: Initial Test: V_{out} (Pink), I_L (Yellow), V_{sw} (Blue), V_{PWM} (Teal)

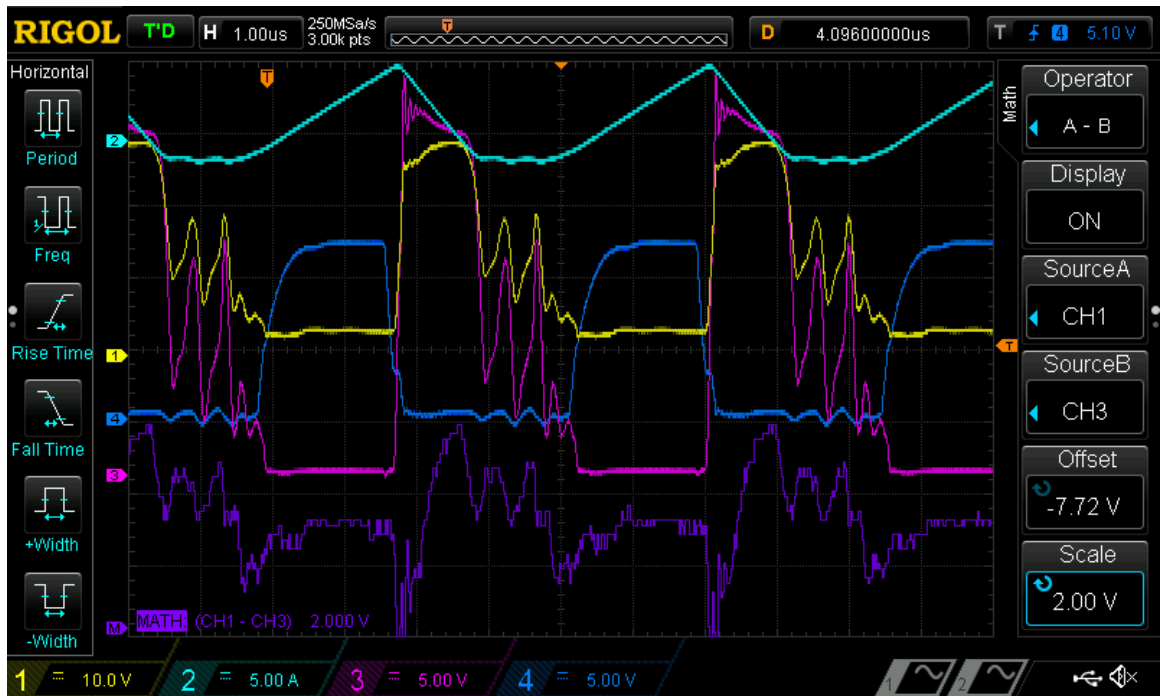


Figure 4.3: Bootstrap Failure: I_L (Teal), V_{BO} (Yellow), V_{gsL} (Blue), V_{sw} (Pink), $V_{bootstrap}$ (Purple)

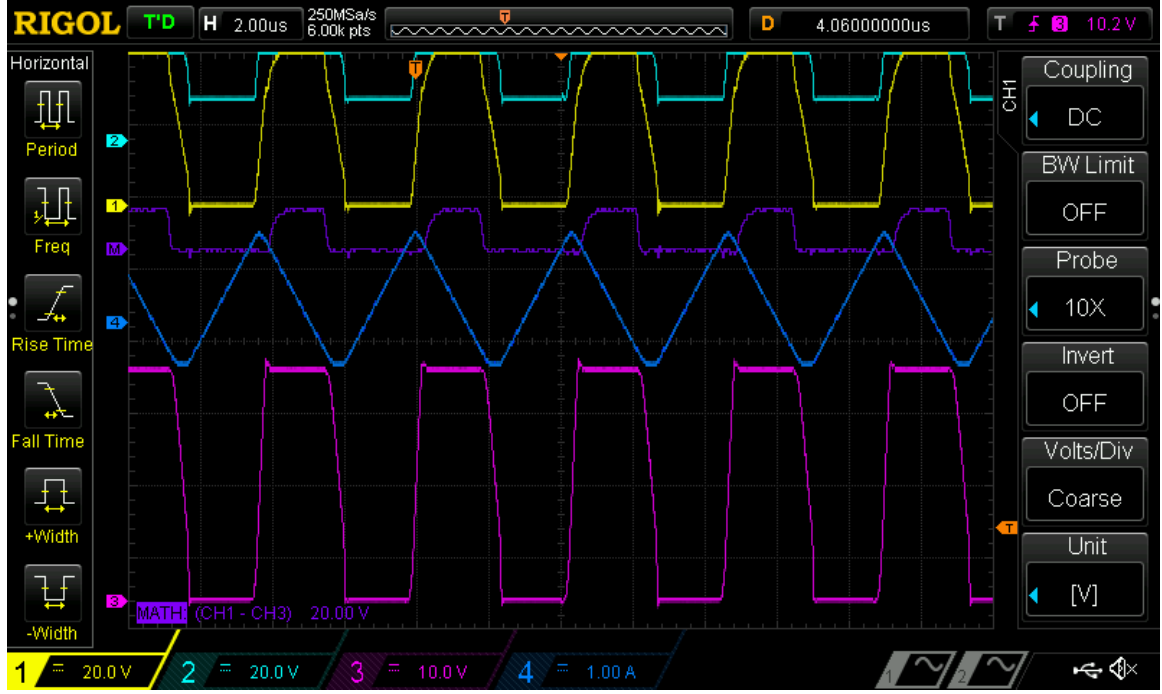


Figure 4.4: Waveform right before the failure: I_L (Blue), V_{HO} (Yellow), V_{BO} (Teal), V_{sw} (Pink), V_{gsH} (Purple)

Table 4.1: MOSFET Characteristics

Device	R_{on} (m Ω)	Q_g (nC)	$\frac{1}{Q_g R_{on}}$ ($\frac{1}{m\Omega \cdot \mu C}$)
CSD19535KCS	3.1	78.0	4.14
IPP023N10N5	2.3	168.0	2.59
FDP083N15A	6.9	64.5	2.26
IPI200N15N3	20.0	23.0	2.17
AOT2500L	6.5	97.0	1.59
IRFB4615PbF	32.0	26.0	1.20

the Lab 3 Procedure. This was also achieved and waveforms from the oscilloscope were captured. Figure 4.5 shows the switch node voltage, inductor current, battery voltage, and output voltage. The main observation from Figure 4.5 is the over-voltage and oscillations on the switch node voltage due to parasitic inductance from our setup

and having to solder to the bottom of the MOSFET leads. The other observation for Figure 4.5 is the inductor current ripple which according to the figure is a peak to peak ripple of 4 A. With $L = 16 \mu\text{H}$, $V_{batt} = 25 \text{ V}$, $f_s = 220 \text{ kHz}$, and $D = 0.66$, the peak to peak ripple current comes out to be 4.68 A. This difference can be attributed to the actual inductance being larger than we expected ($L \approx 18.5 \mu\text{H}$ to obtain 4 A ripple). Figure 4.6 shows the low side MOSFET drain current instead of the inductor current. The observation from this waveform is the spike in current when the low side MOSFET turns on, which is via parasitic capacitance across each device and the reverse recovery from the high side body diode. It is worth noting these parasitics and their negative effects of extra switching losses and potential failures will be mitigated during PCB construction and soft-switching implementation. Last in Figure 4.7, the high side MOSFET current signal was measured. However, this measurement was taken after the decoupling capacitor so the signal is not the true high side MOSFET drain current, and this was done intentionally as we didn't want to place the decoupling capacitor farther away from the devices or change any of our soldering to the pins of the devices and cause additional issues. It is also worth mentioning that the two drain current measurements were at half power (125 W at the output). We also observed the thermal aspect of the converter during max power boost operation and those results can be seen in Figure 4.8 and Figure 4.9. The thermal results are as expected with the power MOSFETs as the hottest component but well within its thermal limits.

Efficiency Sweep

After looking at waveforms, an efficiency analysis of the converter was performed. Table 4.2 shows all the data that was obtained for Section IV of the Lab 3 Procedure. Also, Figure 4.10 shows the efficiency of the converter compared to the output power for different output voltages. The results come out as expected with efficiency rising as output power rose and efficiency rising as output voltage fell while power stays constant. The waveforms captured in Figure 4.11, Figure 4.12, and Figure 4.13 show



Figure 4.5: Full Power: V_{sw} (Pink), I_L (Blue), V_{batt} (Teal), V_{out} (Yellow)

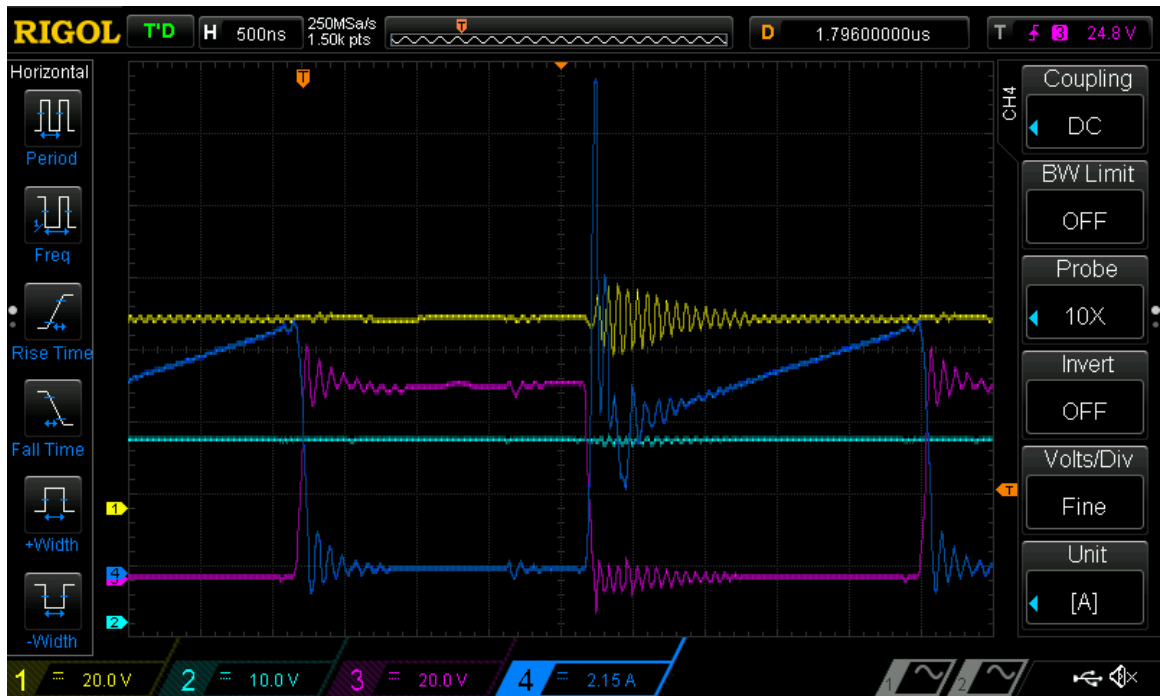


Figure 4.6: Low Side MOSFET Current: V_{sw} (Pink), I_{dL} (Blue), V_{batt} (Teal), V_{out} (Yellow)



Figure 4.7: High Side MOSFET Current (after decoupling cap): V_{sw} (Pink), I_{dH} (Blue), V_{batt} (Teal), V_{out} (Yellow)

these tests at full power but with varying duty cycles to decrease the output voltage for efficiency analysis. In analyzing the efficiency results, we were impressed to get results of $\sim 94\%$ efficiency for a boost converter design we simplified during the debugging process just to obtain functionality. After discussion among the group, we plan to go back and implement the original design of a low inductance soft switched boost converter in hopes to increase this efficiency to the target 98% efficient converter.

4.2.3 Efficiency Analysis

In analyzing the efficiency of our converter there were three main losses that we expect reduced the efficiency of our converter. These losses were device conduction loss, switching loss, and inductor conduction loss. In switching loss we included the reverse recovery loss as well as the IV overlap losses. We choose to neglect core losses in our inductor after FLIR images revealed very little temperature increase of

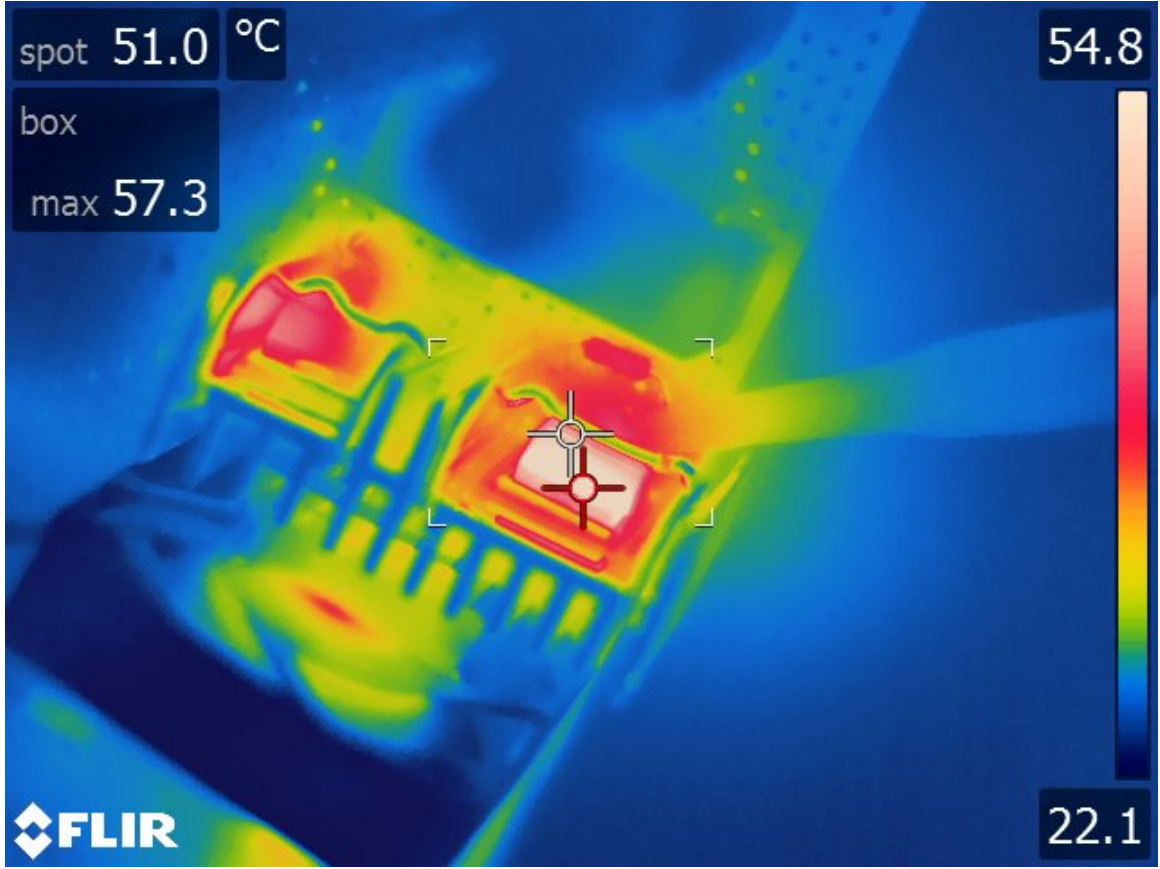


Figure 4.8: MOSFET Thermal Image from FLIR Camera when $P_{out} = 250$ W. Left device is high side and right device is low side.

the inductor during operation. We also found that our analytical calculations were already much higher than observed.

The device conduction loss can be estimated with 4.5:

$$P_{cond} = I_{avg}^2 r_{ds,on} \quad (4.5)$$

We choose to neglect the effect of switching time and dead time on the conduction loss as it was a negligible difference.

For our device conduction loss we also included the loss from the diode conduction during dead time. This can be found using 4.6:



Figure 4.9: Gate Driver Thermal Image from FLIR Camera when $P_{out} = 250$ W.

$$P_{cond_d} = V_d I t_d f_{sw} \quad (4.6)$$

The reverse recovery energy loss can be calculated with 4.7:

$$W_{rr} = V I t_{rr} + V Q_{rr} \quad (4.7)$$

We can then calculate the power loss from reverse recovery with 4.8:

$$P_{rr} = 2W_{rr} f_{sw} \quad (4.8)$$

Where t_{rr} and Q_{rr} were obtained from the device datasheet.

We used the basic overlap loss equation to estimate the device switching losses:

Table 4.2: Measured Boost Converter Efficiency Analysis

V_{out} (V)	V_{in} (V)	I_{in} (A)	I_{out} (A)	P_{in} (W)	P_{out} (W)	η	D
40.3	23.7	9.95	5.60	236	225	0.955	0.528
28.1	23.7	10.0	8.18	237	230	0.970	0.300
34.4	25.0	10.0	7.05	250	242	0.970	0.400
53.2	25.1	10.0	4.45	251	237	0.944	0.659
53.3	25.0	8.00	3.55	201	189	0.945	0.659
53.4	25.0	6.00	2.65	150	141	0.944	0.659
53.8	25.0	4.00	1.74	100	93.7	0.937	0.659
54.7	25.0	2.00	0.83	50.1	45.5	0.908	0.659
70.2	25.1	0.11	0.02	2.76	1.40	0.509	0.659

$$W_{sw} = \frac{VI}{6}(t_f + t_r) \quad (4.9)$$

Where t_f and t_r are the rise and fall time of the device obtained from the datasheet.

The power loss from IV overlap can then be calculated with 4.10:

$$P_{sw} = E_{sw}f_{sw} \quad (4.10)$$

Finally for the inductor conduction losses we can use 4.11:

$$P_{cond_l} = I^2 R_L \quad (4.11)$$

Where R_L was found to be 260 m Ω with an LRC meter.

The result of these are summarized in Table 4.3. As shown in the table the losses sum to approximately 45 W at full power. This results in a calculated efficiency of less than 82%, which is somewhat lower than observed. We suspect that is for a number

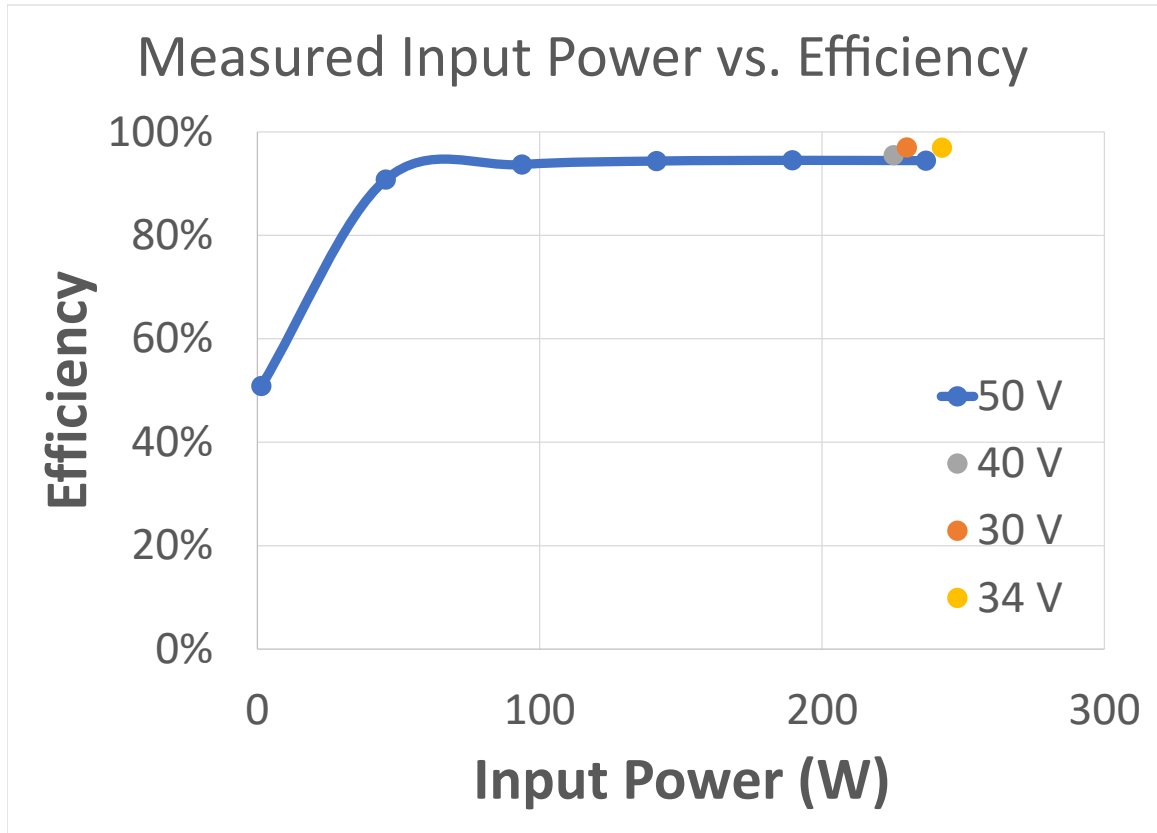


Figure 4.10: Measured Efficiency vs. Input Power at 50 V with added points for 40 V, 34 V, and 30 V

of reasons. The largest analytical loss is from the inductor conduction; however, as mentioned earlier, thermal images revealed very little temperature increase in the inductor which means that there was not significant power loss within. As the calculation relied on a DC resistance found with an RLC meter we believe that this resistance may actually be somewhat lower in reality. It is also likely that our estimated power loss from reverse recovery is less than expected analytically. The values from the datasheet for t_{rr} and Q_{rr} were found at much higher load currents than present in our converter which may cause the datasheet values to be somewhat larger than the circuit values.

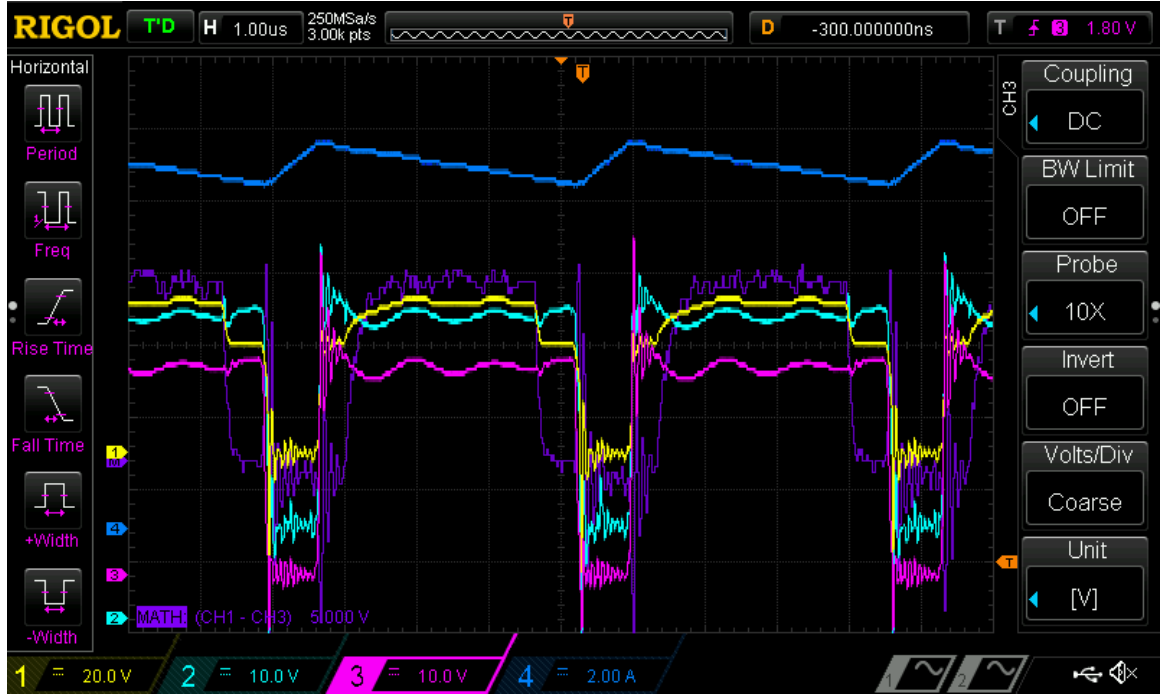


Figure 4.11: Efficiency Sweep at 30 V output: V_{sw} (Pink), I_L (Blue), V_{hb} (Teal), V_{ho} (Yellow), V_{gsH} (Purple)

Table 4.3: Loss Summary Table

Mechanism	Loss (W)
Conduction	.310
Diode Conduction	2.20
Reverse Recovery	16.4
Switching	.367
Inductor Conduction	26.0
Sum	45.3



Figure 4.12: Efficiency Sweep at 34 V output: V_{sw} (Pink), I_L (Blue), V_{hb} (Teal), V_{ho} (Yellow), V_{gsH} (Purple)

4.2.4 LTSpice Simulations

The LTSpice model used to simulate losses can be seen in Figure 4.1. We modeled the devices using a standard VDMOS model using parameters extracted from the device datasheet. There is a PSpice model of the device published by TI; however, it is either encrypted or in a binary like format that make it impossible to port into LTSpice. We modeled the inductor conduction losses internally so they can not be seen in the schematic.

The results of LTSpice simulation can be seen in Figure 4.14 and are summarized in Table 4.4.

Once again we can see that the results do not match exactly with reality, fortunately again in our favour. As with the analytical results it appears that the inductor conduction loss is too dominant in the LTSpice Model. This can be seen in the non-zero power points as a decrease in efficiency as input power increases instead of the expected increase. When input power increases, input current increases linearly;

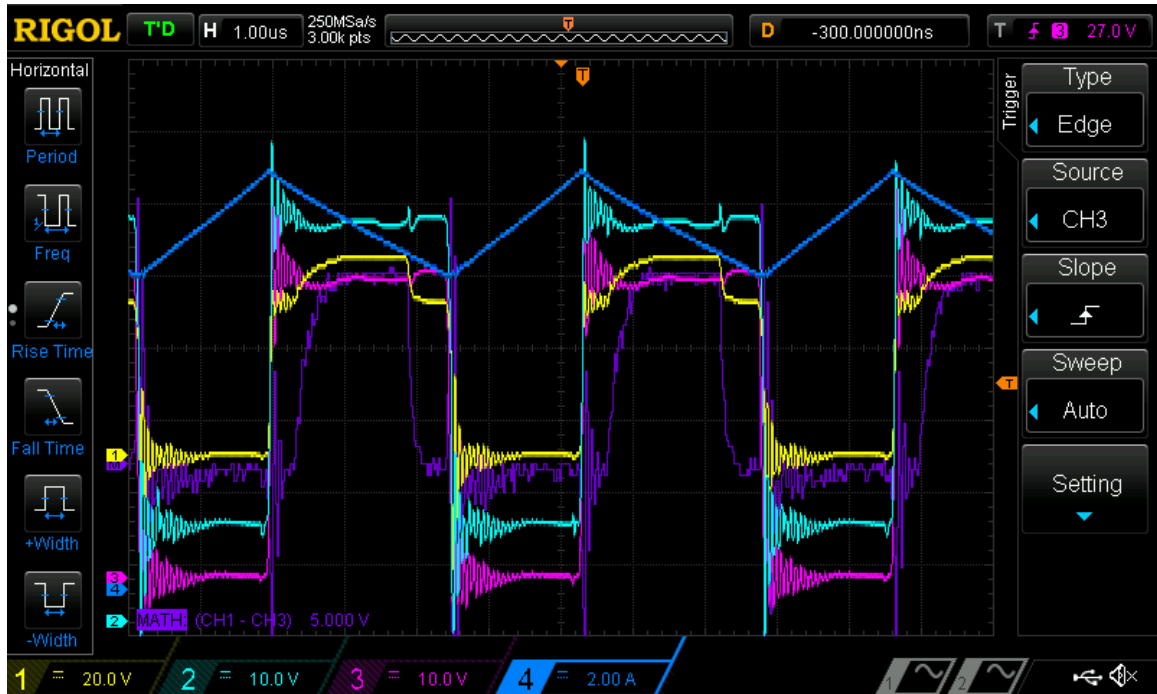


Figure 4.13: Efficiency Sweep at 40 V output: V_{sw} (Pink), I_L (Blue), V_{hb} (Teal), V_{gsH} (Purple)

however, the conduction loss in the inductor increases quadratically and the converter appears to be less efficient. In reality there is likely much less inductor loss and a higher proportion of the losses are reverse recovery which does not scale significantly with input power.

4.3 Discussion and Conclusions

From this lab, we gained valuable insight on how to build a working, efficient boost converter. We met the objectives by building a working converter which according to the lab's performance grading equation comes out to a 20.7% out of 20% of our overall grade showing a good performance in our converter. We discovered and fixed functionality issues from our design along the way: an undersized bootstrap capacitor and duty cycle jitter from our micro-controller. The duty cycle jitter (a so called "jitterbug") from our micro-controller gave us good insight into how significant

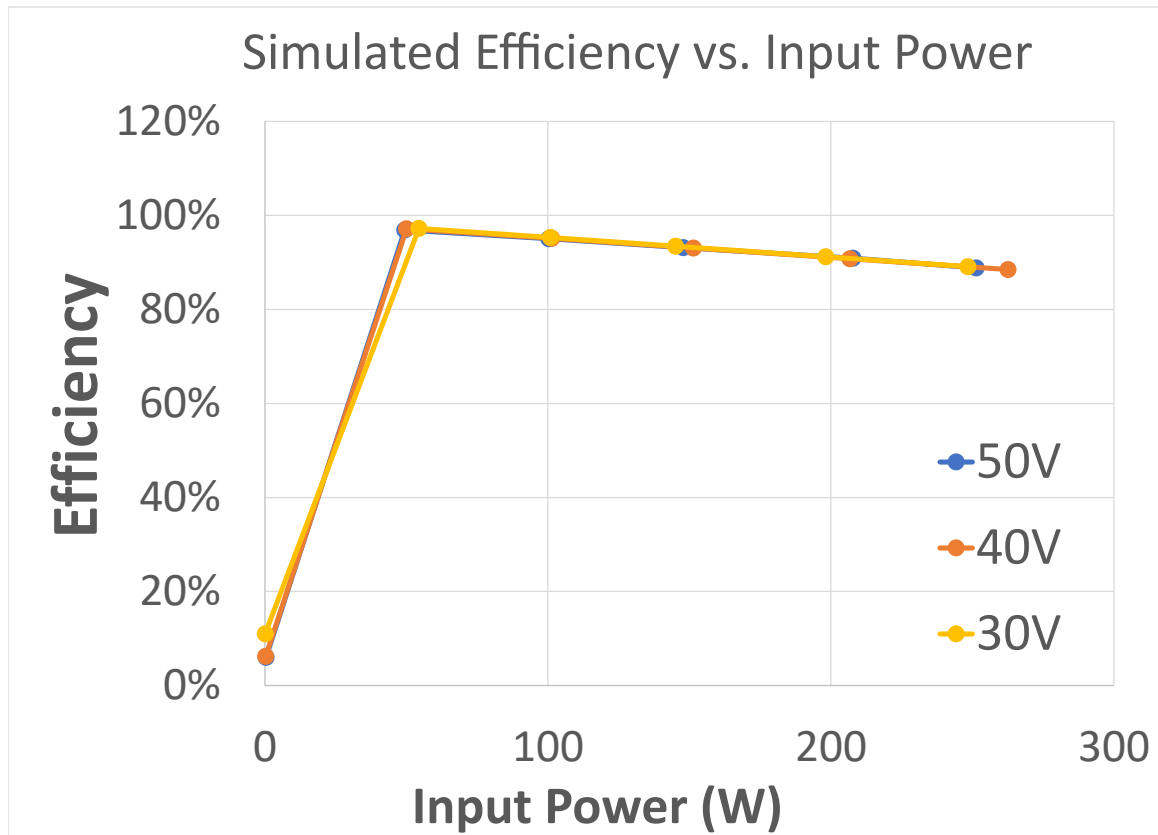


Figure 4.14: Simulated Efficiency vs. Input Power

steps in the duty cycle can be with open loop control. We are also looking to the future in what ways we can improve the converter performance like implementing soft switching.

Soft switching will have several benefits to our converter. The most clear is a reduction in reverse recovery loss, by properly optimizing the dead time for our converter we can entirely eliminate the reverse recovery loss by preventing the diodes from ever turning on. As this is likely the most significant portion of our losses it may greatly increase our efficiency. We will also be able to operate with a smaller inductor and may be able to reduce the conduction losses. We must be careful though to ensure that in downsizing our inductor we do not trade a small reduction in conduction loss for a large increase in core loss.

If we were to improve our current design without going all the way back to soft switching, we might consider decreasing the switching frequency. We originally had it so high (220 kHz) to keep the inductor from saturating, but we have made considerable changes to the inductor since then and we could likely cut our switching frequency in half and still avoid saturation. This change would conceivably half the reverse recovery losses, the switching losses, and the diode conduction losses. It has the potential to create a larger ripple current from the battery and a larger voltage ripple at the output, but we have two large capacitors at the input and one at the output to mitigate this. Additionally, most of our output ripple at the moment is caused by overshoot during switching so a reduction in switching frequency could actually reduce output voltage fluctuations.

Table 4.4: Simulated Boost Converter Efficiency Analysis

V_{out} (V)	P_{in} (W)	P_{out} (W)	η	D
50	251	223	88.8%	0.66
50	208	189	90.9%	0.64
50	148	138	93.1%	0.63
50	100	95.4	95.0%	0.62
50	49.4	47.9	96.9%	0.61
50	0.415	0.025	6.0%	0.5
40	0.270	0.017	6.2%	0.39
40	50.0	48.6	97.2%	0.49
40	101	96.4	95.1%	0.51
40	151	141	93.1%	0.52
40	207	188	90.8%	0.53
40	263	232	88.5%	0.55
30	248	221	89.1%	0.35
30	198	181	91.2%	0.34
30	145	136	93.5%	0.34
30	101	96.2	95.3%	0.31
30	54.4	52.9	97.2%	0.31
30	0.080	0.009	11.0%	0.17

Chapter 5

Closed Loop Circuit Design

5.1 Introduction and Objectives

Experiment 4's objective was to close the control loop of the boost converter. During this process we began by simply establishing duty cycle control with the LM5121. We then moved on to closing the current loop by populating the sense resistor pins, and reducing the ramp function to its nominal value. The final phase of the experiment involved closing the final loop, the voltage loop. At the conclusion of the experiment we also looked at the feasibility of soft switching during closed loop control.

5.2 Methods and Results

5.2.1 Averaged Switch Modeling and Simulation

We began experiment 4 with a simulation of an average model of the converter. In this model we took into account the losses associated with the resistance of the inductor, the on-resistance of the device, the IV switching losses, and reverse recovery of the body diode. The schematic for this model can be seen in [Figure 5.1](#).

With this model we looked at the stability of our converter with both transient and frequency simulations over a range of operating conditions. The sweep of various

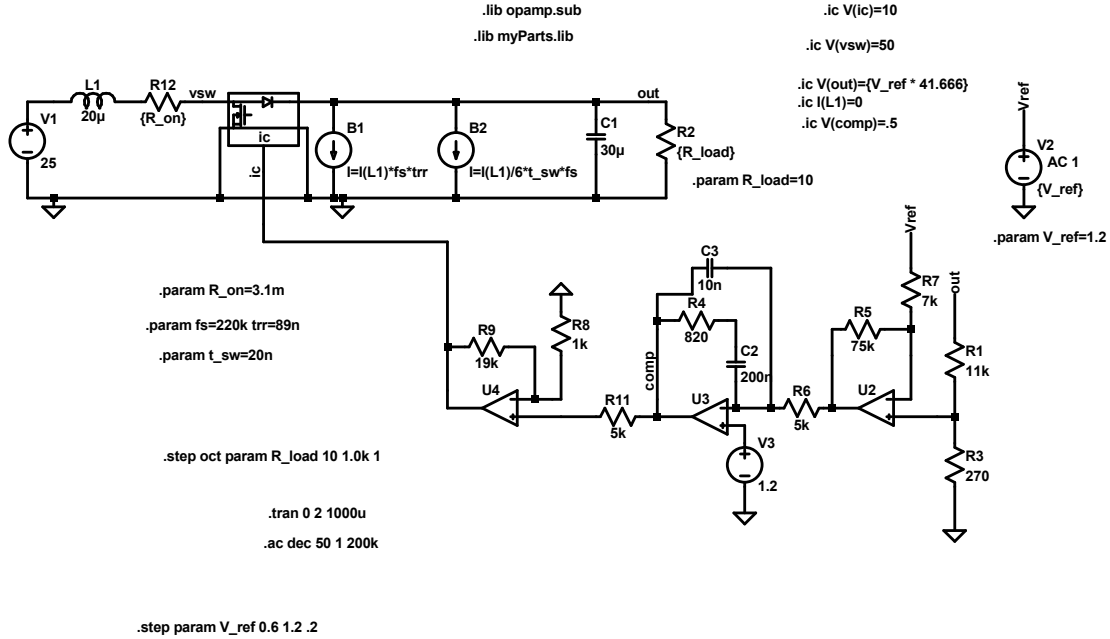


Figure 5.1: Average Boost Converter Schematic

output voltages can be seen in Figure 5.2 and a sweep over currents can be found in Figure 5.3. For the frequency simulation we looked at how a signal at the reference voltage effected the output voltage. We looked at both a lossless and lossy model and found quite different results in both. The lossless frequency sweep can be found in Figure 5.4 and the sweep taking losses into account can be seen in Figure 5.5. These results do not make very much sense, and we have yet to come up with a reasonable explanation for their behavior. We also got unusual results for the DC operating point with this model, so we switched to using a switching model for the remainder of our simulations.

5.2.2 Switching Model and Simulation

In preparation for closing the loop in hardware we also simulated our circuit with a switching model. This switching model used an LTSpice model for the LM5121 that gave us insight into how our converter would preform under load transients. This

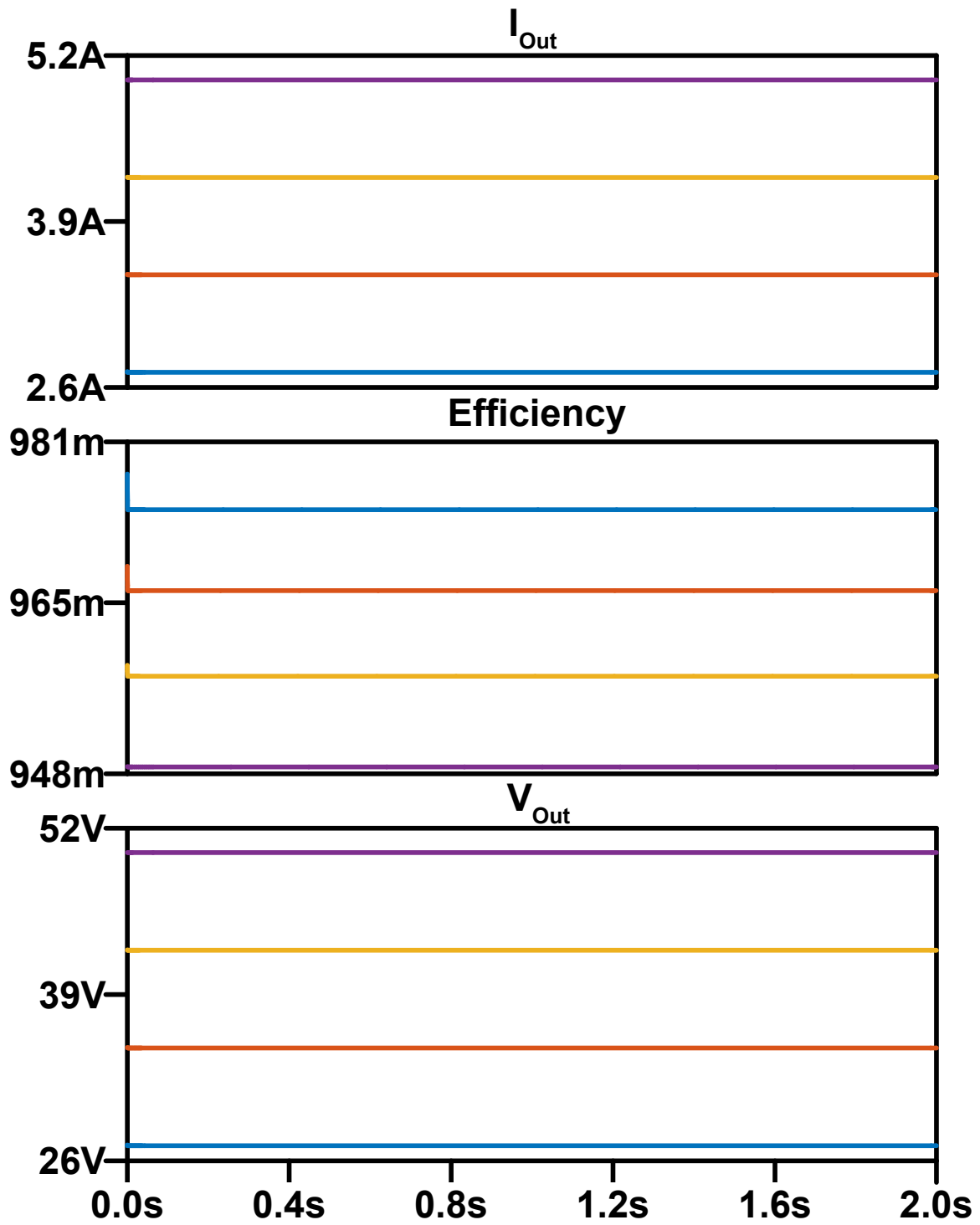


Figure 5.2: Time domain sweep of Output Voltage

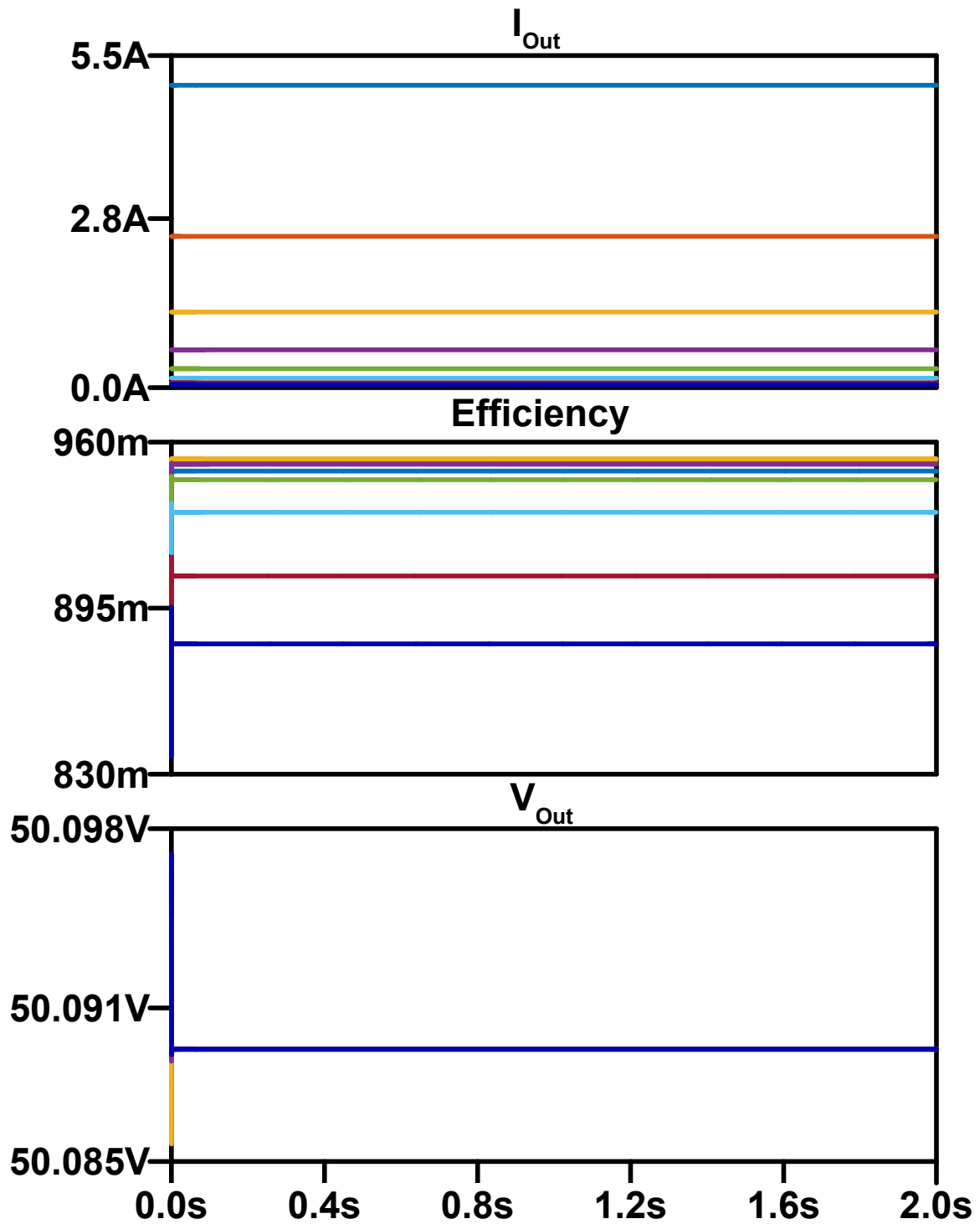


Figure 5.3: Time domain sweep of Output Current

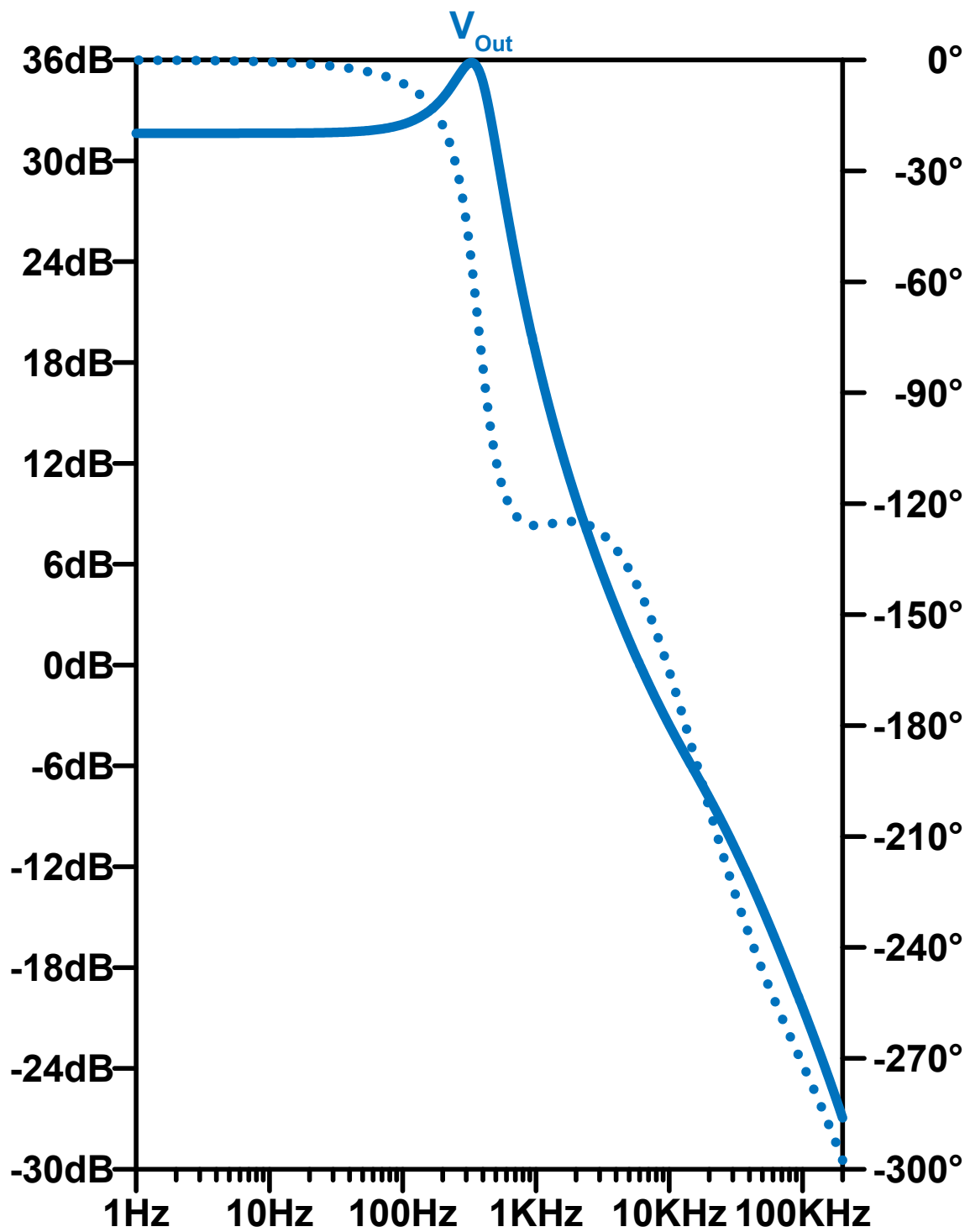


Figure 5.4: Frequency Response of converter with no loss modeling

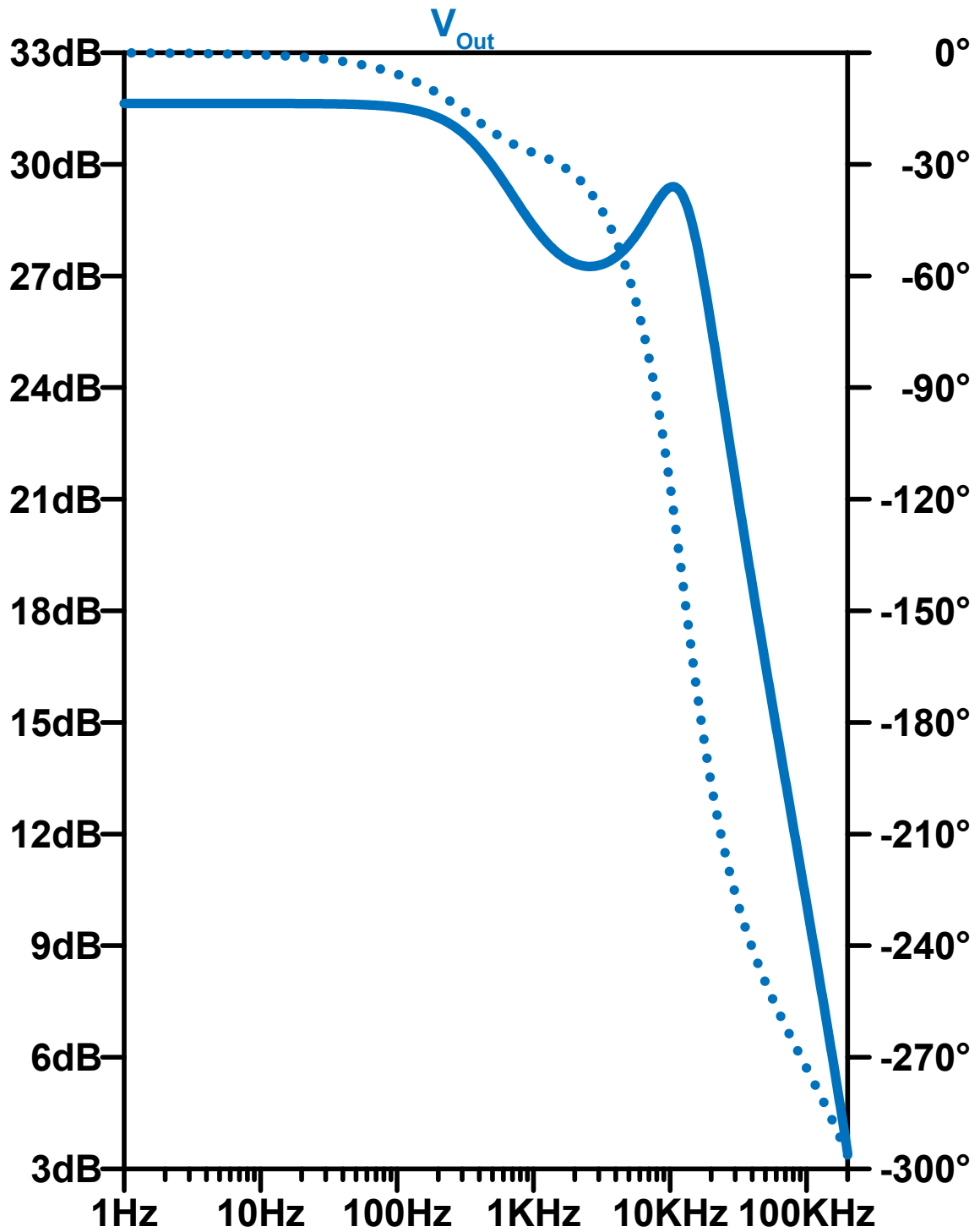


Figure 5.5: Frequency Response of converter with loss modeling

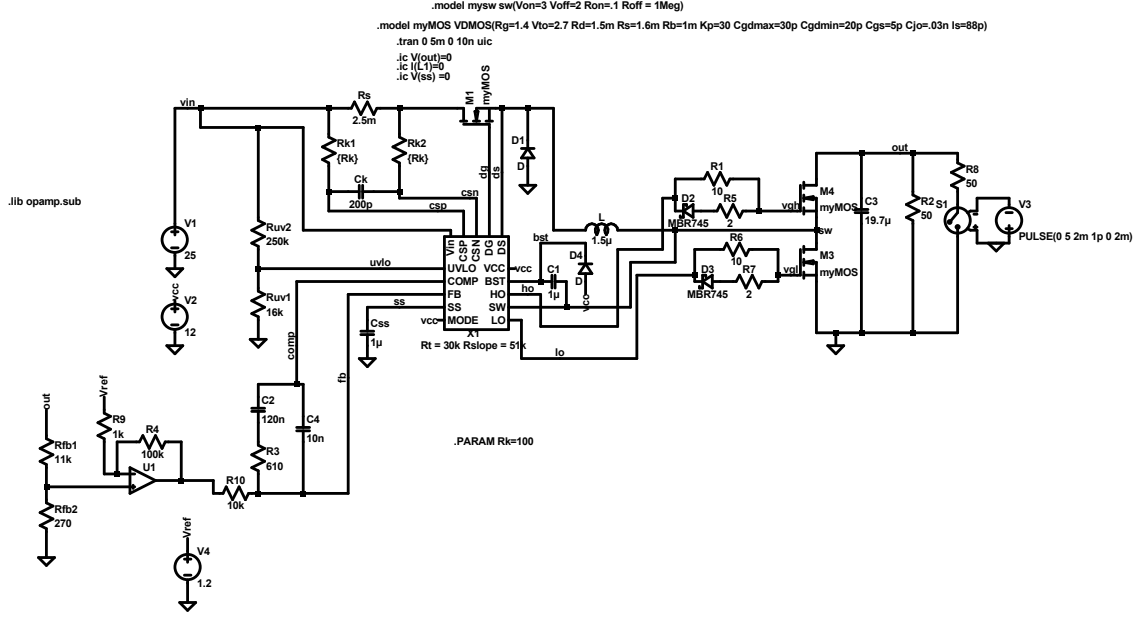


Figure 5.6: Switching Boost Converter Schematic

schematic for this model can be seen in Figure 5.6 and the results of a load step can be seen in Figure 5.7.

5.2.3 Open-Loop LM5121 Implementation in Hardware

Initially, all the components were connected to run in the open loop implementation of the hardware. The UCC27211A was used as the gate driver instead of just using the LM5121 to try to avoid blowing up the controller. This decision was made based on the price difference of blowing up the UCC27211A over the LM5121. A switching test was then performed to confirm that both FETs would switch successfully.

In testing the open loop power state, there was a loose female banana connector on the breakout board that was disconnected while supplying V_{IN} of 25V to LM5121 while an external V_{CC} of 12V was connected. This then caused the power supplies for V_{IN} and V_{CC} to current limited. The LM5121 was then replace, and the loose female banana connector was re-soldered. After re-connecting everything, V_{IN} current limited again. The external gate driver (UCC27211A) was then replaced to see if it

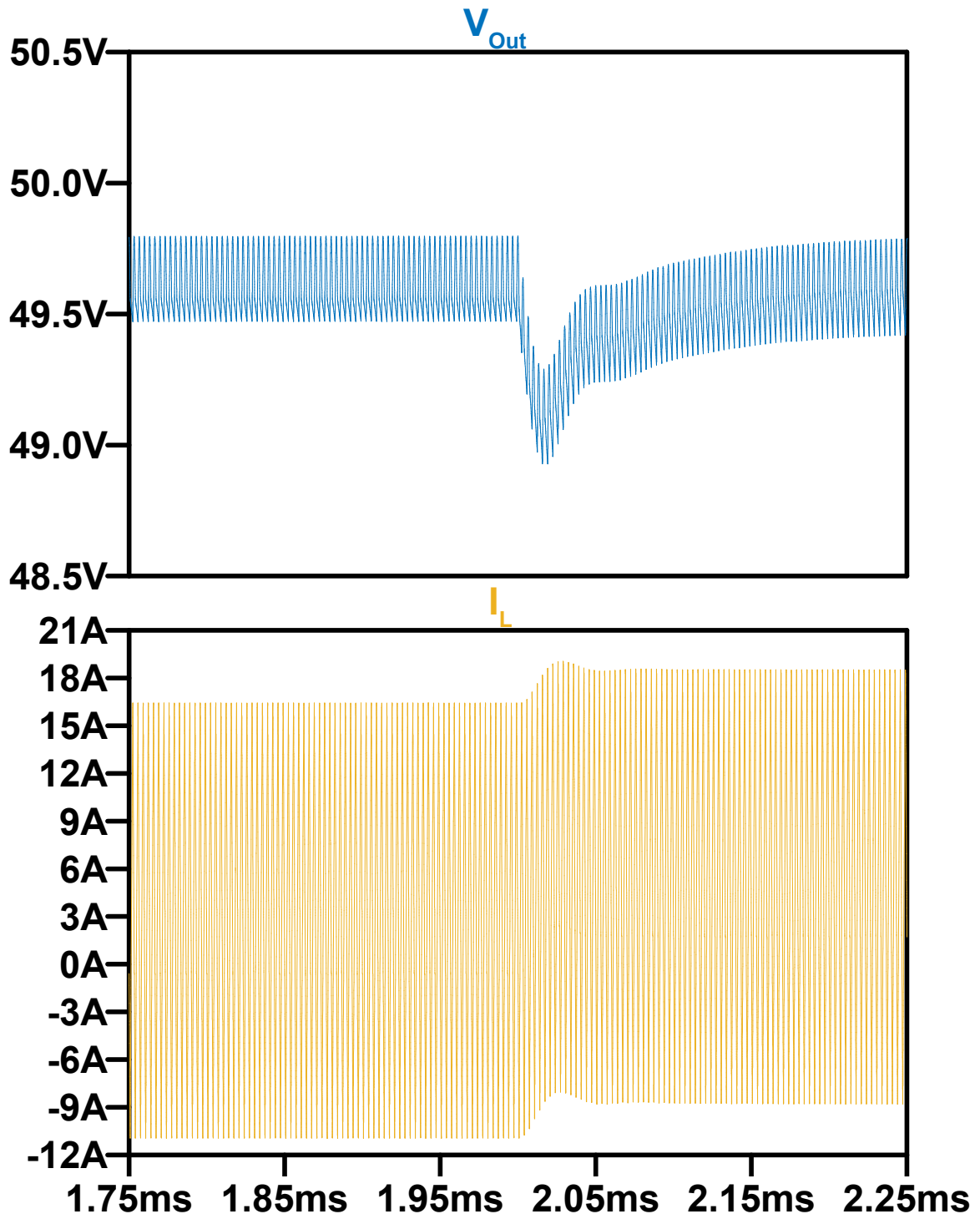


Figure 5.7: Step Response of converter

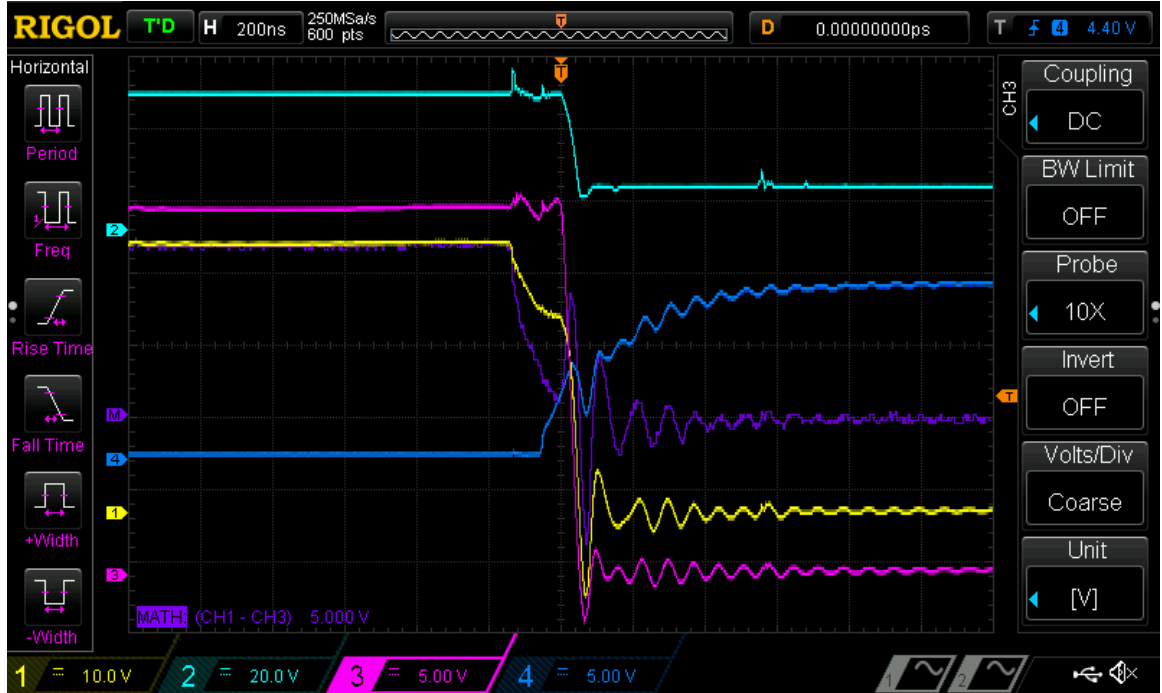


Figure 5.8: Approximately 0 ns dead time with V_{bst} (light blue), V_{gsL} (dark blue), V_{goH} (yellow), V_{sw} (pink), and V_{gsH} (purple)

was the issue. After replacing the external gate driver, the power supply did not current limit and thus successful gate switching was again observed.

The power stage was then connected again and very bad waveforms were observed as seen in Figure 5.8. The dead time was measured to be around 0 nanoseconds. The gate resistance going to the FETs was initially thought to be high enough to extend the dead time to be able to operate without shoot-through. The hypothesis of too small of a dead time was tested by disconnecting the inductor and placing 25V across the output and looked at the gate signals. The dead time was still very small and a lot of ringing was observed.

It was then decided to utilize the LM5121s adapted dead time by disconnecting the external gate driver and simply driving the FETs off of the LM5131. The VSB2200S, a 200V, 2A Schottky diode, was chosen to be the bootstrap diode based on the high voltage of the switch node. The bootstrap capacitance was still chosen to be $1\mu\text{F}$ and based on (5.1). The ripple of the bootstrap voltage was determined to be 80mV.

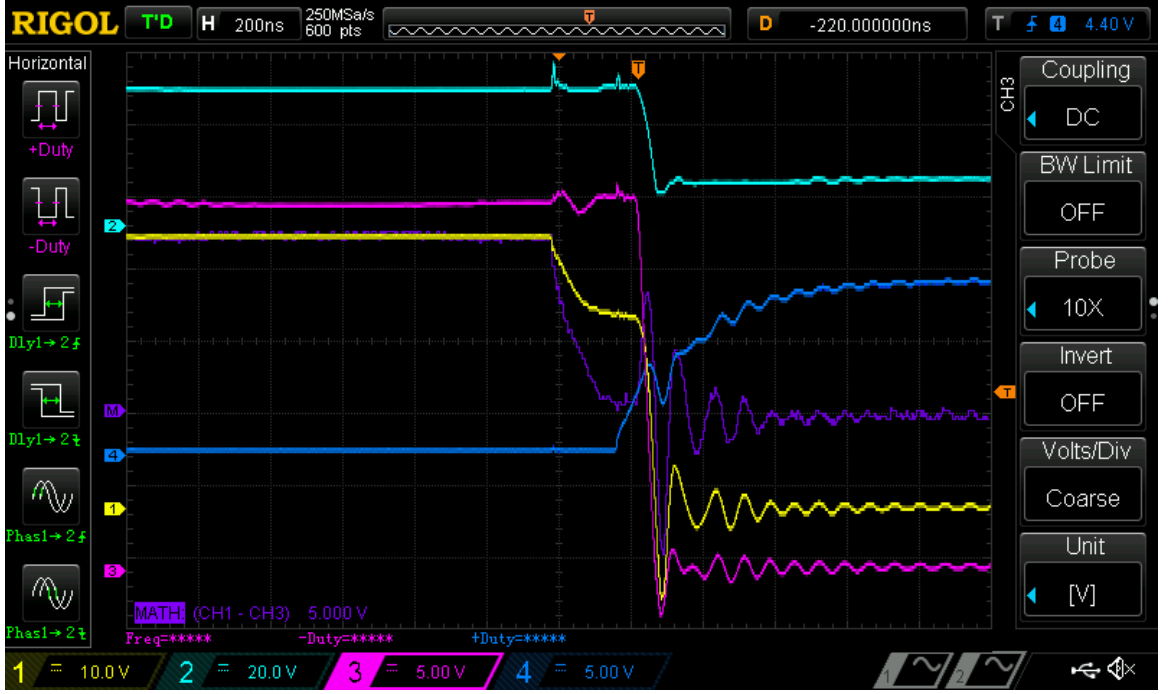


Figure 5.9: Approximately 50 ns dead time with V_{bst} (light blue), V_{gsL} (dark blue), V_{goH} (yellow), V_{sw} (pink), and V_{gsH} (purple)

After this adjustment, a successful test was completed with an increase in dead time to approximately 50 nanoseconds as seen in Figure 5.9.

$$C_{BST} = \frac{Q_G}{\Delta V_{BST}} [F] \quad (5.1)$$

Next, while powering up, there was a spark when the banana jack for V_{IN} was plugged in, and the LM5121 was blown. After replacing the LM5121, the converter could operate in open loop at full power. Table 5.1 shows the DC power parameters along with an overall converter efficiency of 94.11%. This efficiency is slightly down from the results in Experiment 3 (94.4%) and can be contributed to the added resistance caused by the buck converter FET. The waveforms of the converter in open loop full power operation are shown in Figure 5.10 and are similar to what has been observed in the past for an operating boost converter.

Table 5.1: 250W Open Loop Results

V_{IN}	24.94V
I_{IN}	10.05A
P_{IN}	250.6W
V_{OUT}	49.86V
I_{OUT}	4.73A
P_{OUT}	235.8W
η	94.11%

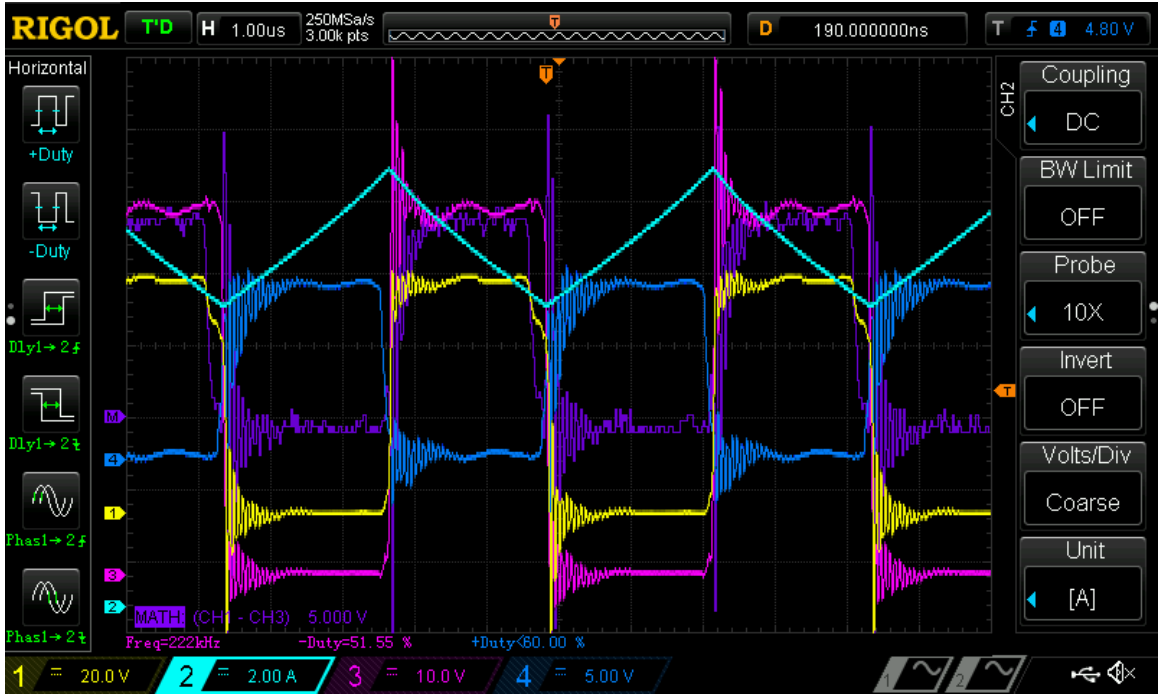


Figure 5.10: Full power open loop operation I_{ind} (light blue), V_{gsL} (dark blue), V_{goH} (yellow), V_{sw} (pink), and V_{gsH} (purple)

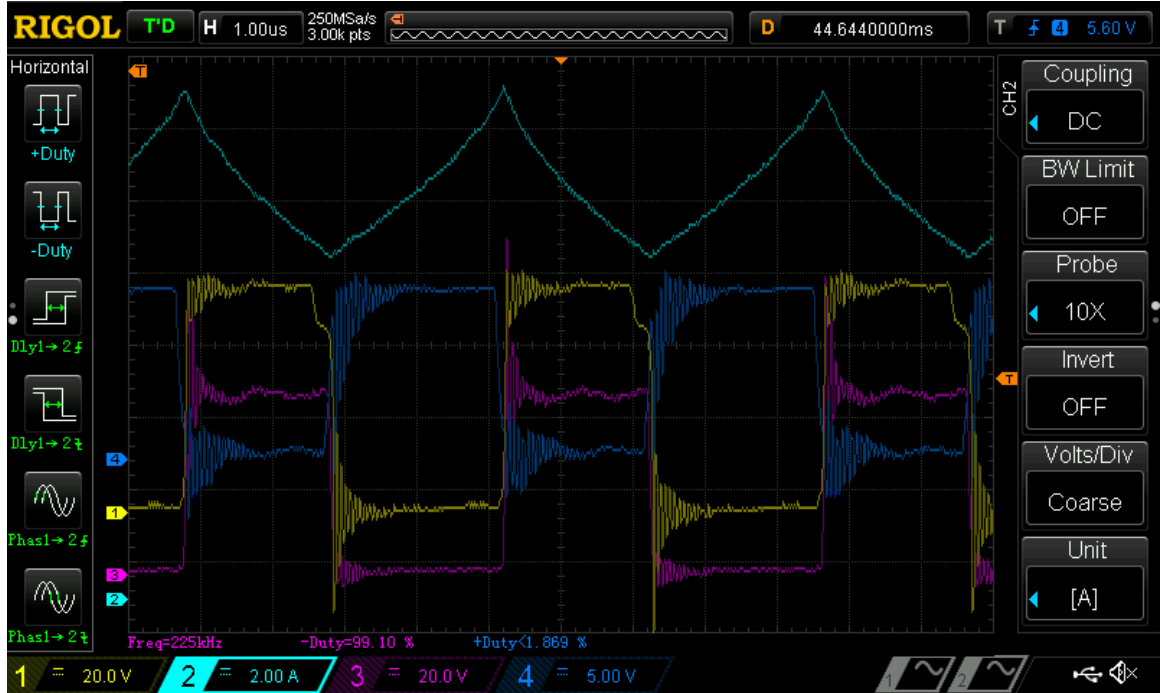


Figure 5.11: Full Power Boost Operation in Current Loop Control with I_{ind} (light blue), V_{gsL} (dark blue), V_{goH} (yellow), V_{sw} (pink)

5.2.4 Closed-Loop Implementation

Current Loop Control

In transitioning to closing the loop for this lab, we started with just closing the current loop part. This section of the lab presented no issues and we quickly were able to confirm functionality as shown in Figure 5.11. We also tested the initial surge current through the inductor at start up of 19.1 V. The results shown in Figure 5.12 show a manageable peak surge current of 10 A.

Voltage Loop Control

The next step was implementing the rest of the close loop control by controlling the output voltage at 50 V. This is done with the compensator network designed earlier in the lab to feedback the output voltage and compare with the sensed inductor current to determine the corresponding duty cycle needed in switching the devices.

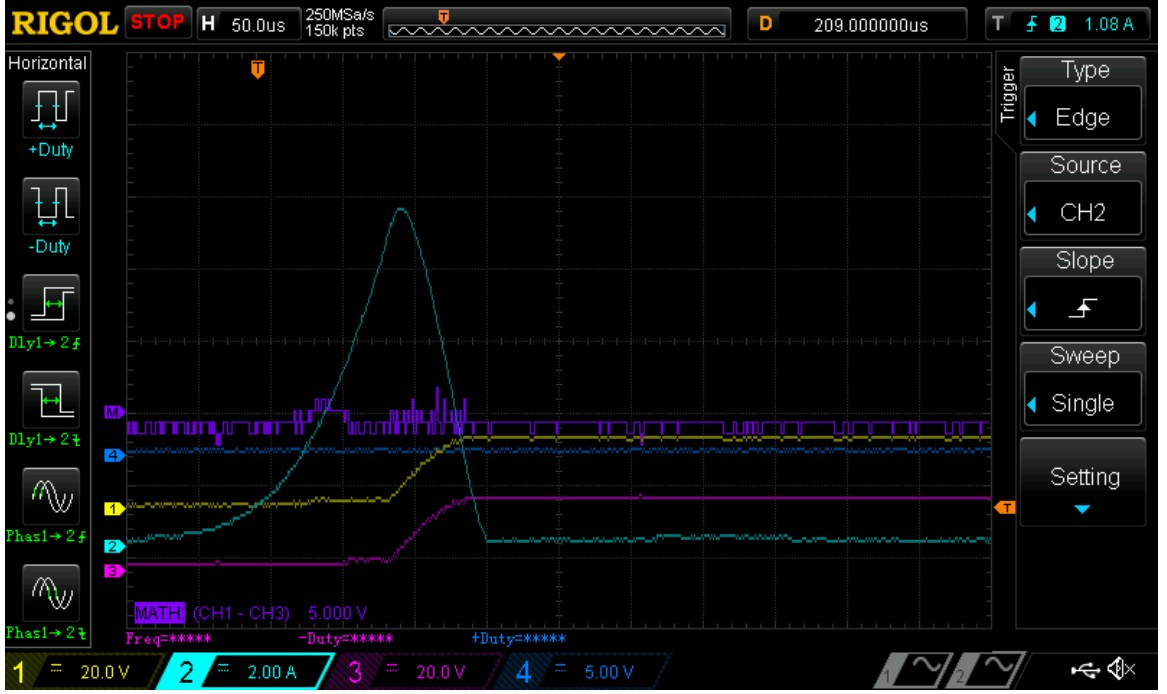


Figure 5.12: Initial Current Surge at Start Up

In beginning this process, we used a compensator network as shown in Table 5.2 and the prelab analysis confirmed this design. However, in the experimentation phase, this compensator network caused there to be audible noise during operation.

To eliminate the compensator audible noise, we changed the compensator network by reducing R_{FB2} & R_{COMP} and increasing C_{COMP} & C_{HF} to keep DC gain similar as before while decreasing crossover frequency and more closely matching the low frequency zero and high frequency pole with the plant transfer function. The network initially used with noise and the replacement network with better performance is detailed in Table 5.2.

After fixing the compensator network issue, we experienced a second issue after start up. The failure waveform capture can be seen in Figure 5.14. It is worth noting that the noise on the inductor current waveform is actually from the current probe used. Eventually it was discovered the CSP and CSN pins of the LM5121 were switched in their connections to the inductor and input voltage.

Then we experienced a third issue in this voltage loop control section. When we got to a peak inductor current of 9 A, the switching would stop, which brought down the inductor current but the switching never came back on. We went through a long process of debugging this issue which involved examining every aspect of the LM5121 operation. Figure 5.15 and Figure 5.16 show some of the different signals probed in an attempt to find the source of the failure. We also probed the LM5121 soft-start pin and UVLO pin but still no answers for this problem. The issue was resolved when we placed a second sense resistor in parallel with the original sense resistor to allow more current to pass through the sense resistor. After this change we were able to achieve a full power closed loop boost converter as shown in Figure 5.17. Normally adding this second sense resistor in parallel would cause a problem because the inductor current limit would increase past the safety value; however, we plan to implement a ZVS boost converter with high current ripple so this decision to reduce the sense resistance is acceptable.

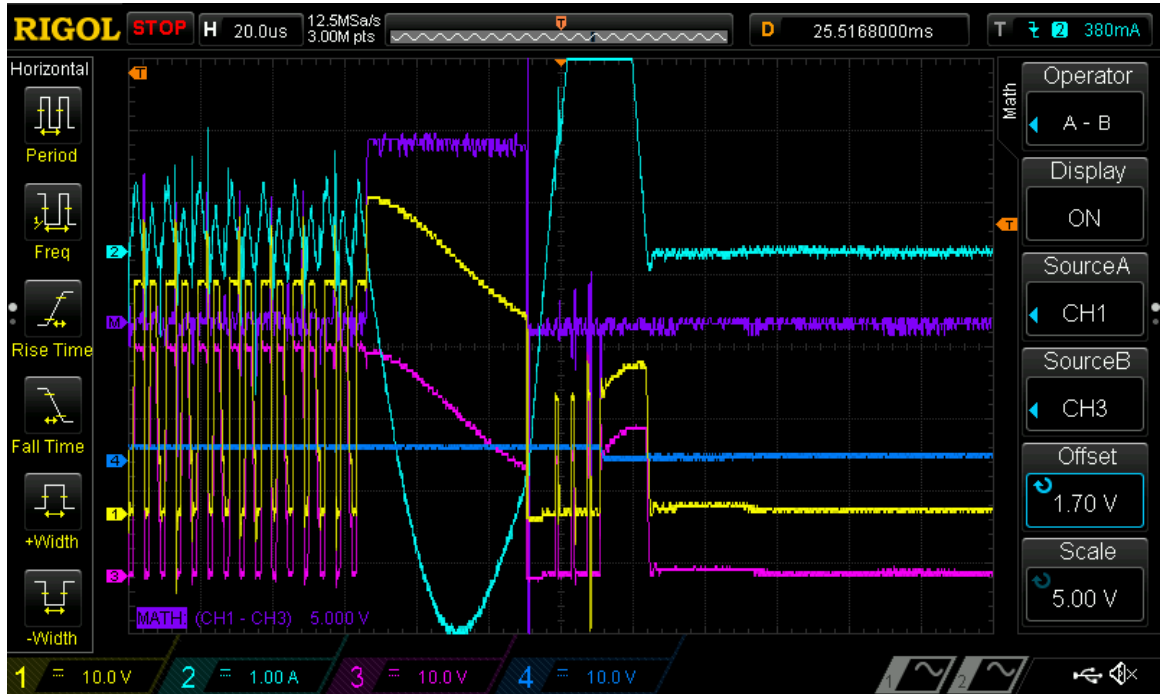
Per the instructions of the lab procedure to test the effectiveness of the voltage loop control, we gathered data detailed in Table 5.3 operating the boost converter at different output voltages, input voltages and load values (i.e. output power values). This table of data shows great output voltage control, but it is worth noting these results do not account for high frequency output voltage ripple. We also tested the step response of the LM5121 by instantly dropping the load from 50 ohms to 25 ohms and the result is presented in Figure 5.18.

5.3 Discussion and Conclusions

Through this lab, we gained valuable insight into the workings of closed loop control of a boost converter. We met all specifications for complete voltage loop control for the full power boost converter and kept the line regulation of the output voltage before 0.6%. We learned how to use every aspect of the LM5121's features and how to best design a Type II feedback compensator network for current-mode control.

Table 5.2: Compensator Network

	Original Compensator	Improved Compensator
R_{FB1}	1 M Ω	11 k Ω
R_{COMP}	150 k Ω	610 Ω
C_{COMP}	1.06 nF	120 nF
C_{HF}	106 pF	10 nF
DC Gain (A_{FB})	896	700
f_{zero}	1 kHz	2.174 kHz
f_{pole}	10 kHz	26 kHz

**Figure 5.13:** Initial Compensator Noise Waveforms with I_{ind} (light blue), V_{gsL} (dark blue), V_{goH} (yellow), V_{sw} (pink), and V_{gsH} (purple)

This compensation scheme adds an RC branch to flatten the gain, and improve the phase response in the mid frequency range. We also have a pole-at-zero for achieving high DC gain, good DC regulation, and low-frequency line injection. It was beneficial

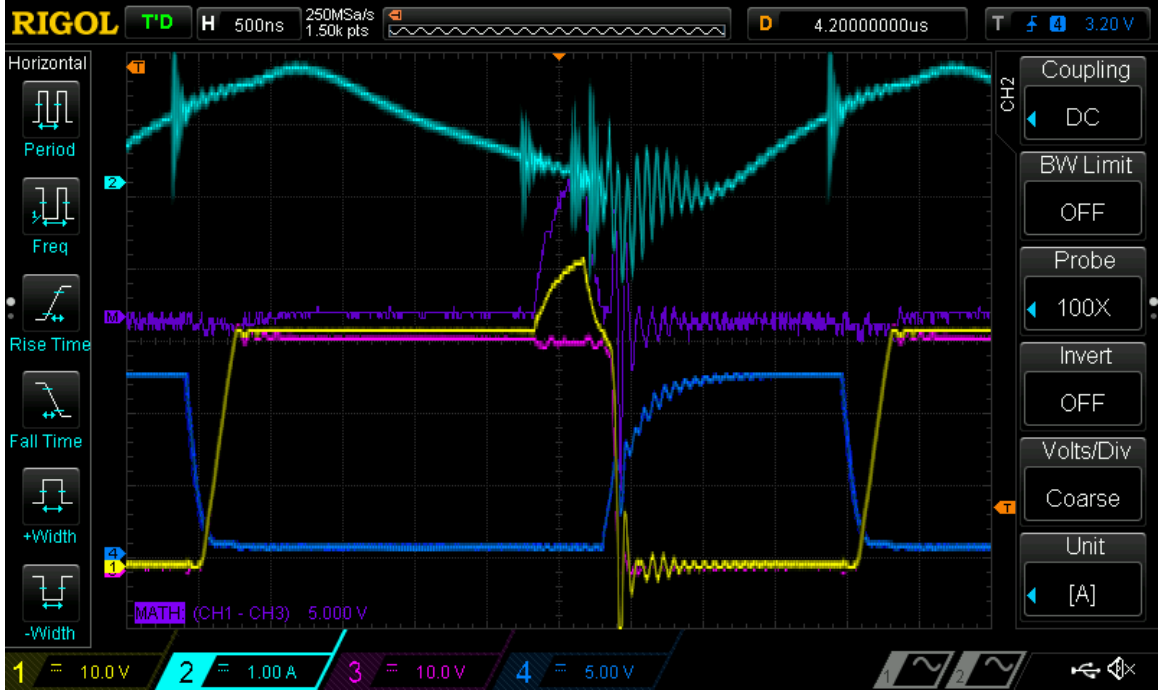


Figure 5.14: Failure from Switching the CSP and CSN pins with I_{ind} (light blue), V_{gsL} (dark blue), V_{gsH} (yellow), V_{sw} (pink), and V_{gsH} (purple)

to break up the process of achieving closed loop control into three parts to both help understand the control process inside the LM5121 and help achieve functionality in stages. We were also able to keep the boost converter efficiency at 94% with closed loop control. The LM5121 implementation into our boost converter also gives us necessary protection features like step response, over-current protection, under-voltage lockout, and soft-start for output capacitor charging.

In efforts to improve in the PCB design phase, we found a surface mount sense resistor that will allow us to have less associated parasitics. This will give more accurate sensing of the inductor current as we learned from this lab that the parasitics can cause the sensed current to be different than the actual current.

From this lab we also learned the impacts of external gate driver vs. internal gate driver of LM5121. Looking ahead, we have to use an specific external gate driver for the GaN devices. As we saw in the results of this lab, we lose the dead time capabilities in doing this. Therefore, we plan to implement RCD circuits for the high



Figure 5.15: Failure from Sense Resistor Parasitics with I_{ind} (light blue), V_{gsL} (dark blue), V_{goH} (yellow), V_{sw} (pink)

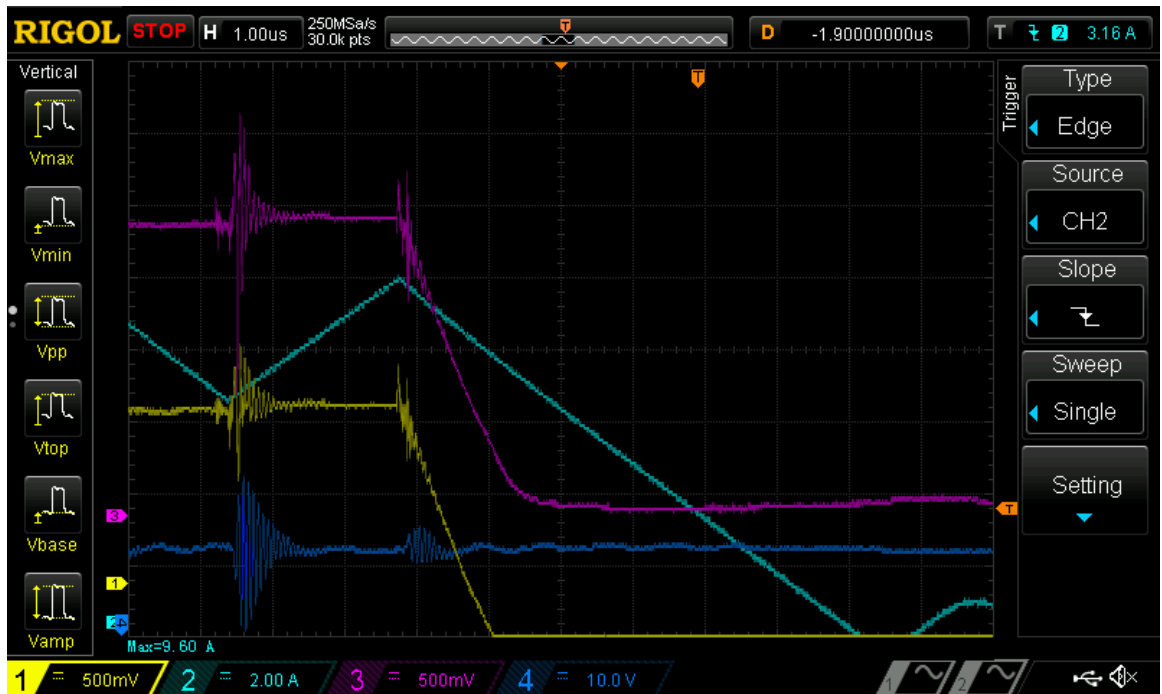


Figure 5.16: Same Failure with FB and COMP pins shown in yellow and purple

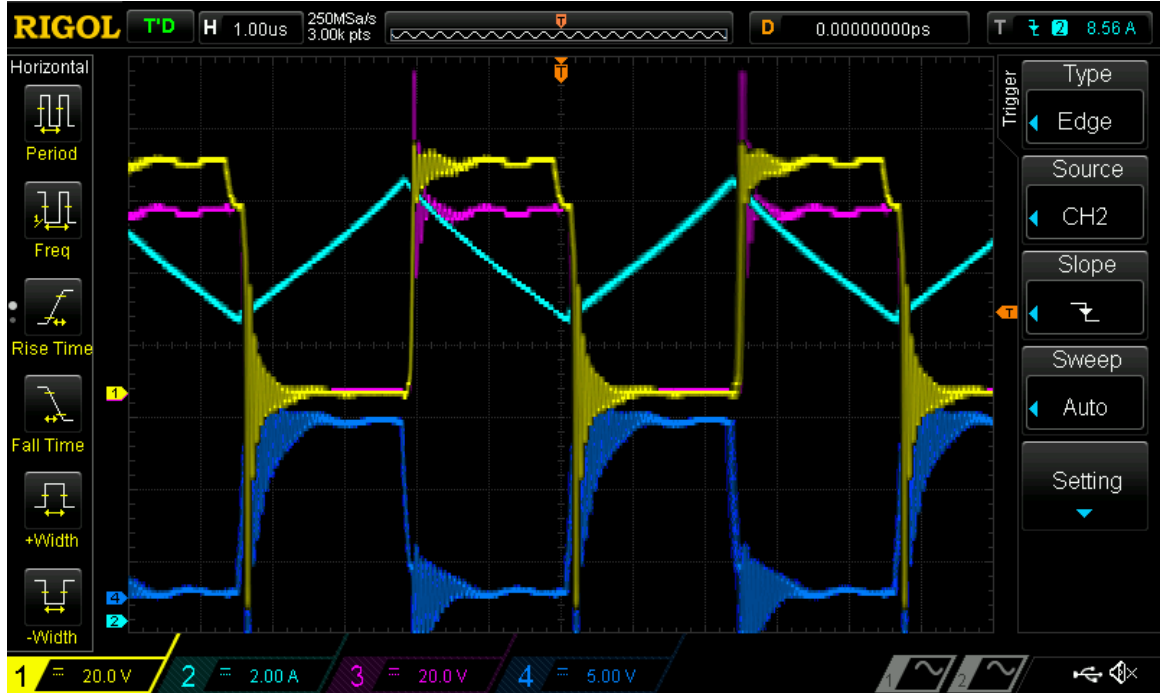


Figure 5.17: Full Power Voltage Loop Control Boost Converter with I_{ind} (light blue), V_{gsL} (dark blue), V_{goH} (yellow), V_{sw} (pink)

and low side gate driving to add a dead time for the driving to avoid shoot-through failures.

We had some interesting issues with the AC closed loop simulation. It was very difficult to get LTSpice to find a reasonable operating point and even with extensive initial conditions there were still some nodes that had very unreasonable DC operating points. There was also a significant impact in the frequency sweeps of what initial conditions were set to. However, the switching model worked mostly as expected and gave us good insight into how our converter would perform in normal operating conditions as well as in transient events.

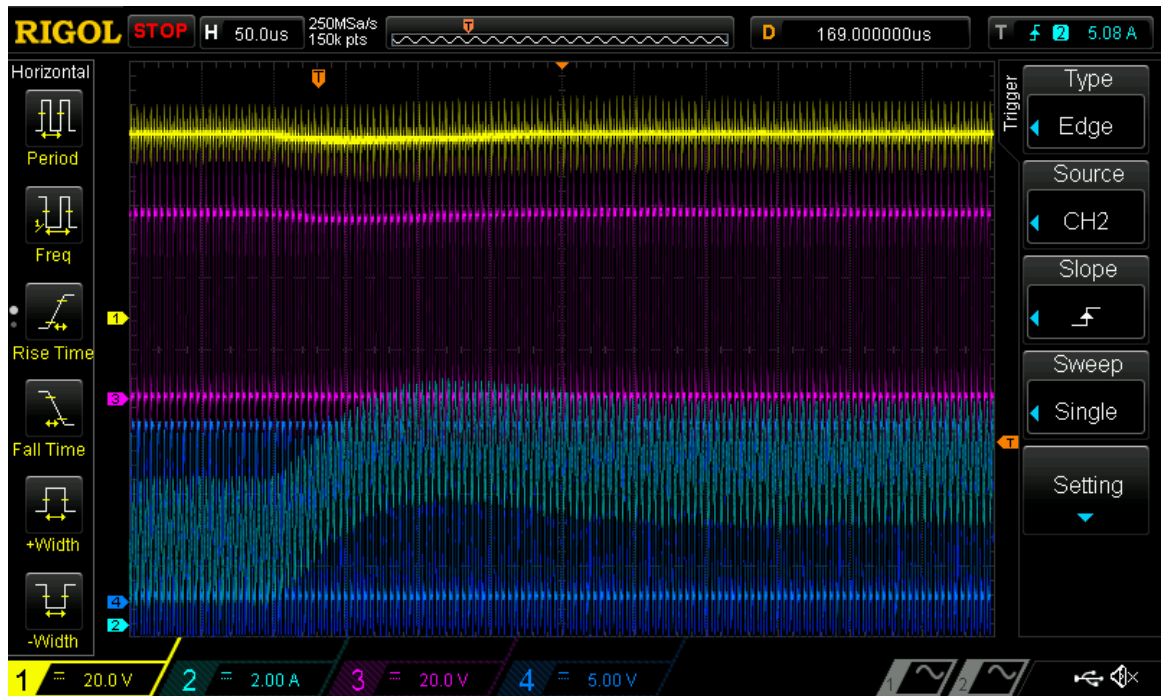


Figure 5.18: Converter Reaction to Load Step from $50\ \Omega$ to $25\ \Omega$ with I_{ind} (light blue), V_{gsL} (dark blue), V_{out} (yellow), V_{sw} (pink)

Table 5.3: Line Regulation

$P_{in}(W)$	$V_{in}(V)$	$I_{in} (I)$	$V_{out}(V)$	$I_{out}(A)$	$P_{out}(W)$	Efficiency	Line Regulation
250	25.0	10.0	50.06	4.70	235	94%	0.00%
100	25.1	4.0	50.33	1.87	94	94%	-0.54%
101	20.0	5.1	50.33	1.87	94	93%	-0.54%
101	22.0	4.6	50.33	1.87	94	93%	-0.54%
100	24.0	4.2	50.33	1.87	94	94%	-0.54%
100	26.0	3.9	50.33	1.87	94	94%	-0.54%
100	28.0	3.6	50.33	1.87	94	94%	-0.54%
100	30.1	3.3	50.33	1.87	94	94%	-0.54%
50	25.0	2.0	50.39	0.89	45	90%	-0.66%
75	25.0	3.0	50.36	1.38	69	92%	-0.60%
125	25.1	5.0	50.27	2.35	118	94%	-0.42%
150	25.0	6.0	50.24	2.82	142	94%	-0.36%
175	25.0	7.0	50.20	3.29	165	94%	-0.28%
200	25.0	8.0	50.15	3.77	189	94%	-0.18%
225	25.1	9.0	50.10	4.25	213	94%	-0.08%

Chapter 6

Printed Circuit Board Layout

6.1 Introduction and Objectives

For the final version of the circuit a Printed Circuit Board (PCB) will be created to replace the previous perfboard construction. The PCB gives us several advantages over the previous construction method, primarily in that it reduces the parasitics of the connections between each component. These reduced parasitics improve the overall converter function and also allow for the use of Wide Band Gap (WBG) GaN Devices, which will be discussed in more detail shortly. The PCB also allows the final design to be constructed in a much more efficient process that is less prone to errors. Finally the PCB Also looks much better than the perfboard construction, an important part of any design. A 3D render of the board can be seen in [Figure 6.1](#) and a picture of the finished product can be seen in [Figure 1.1](#).

6.2 Daughter Boards

For the power stage a motherboard/daughterboard configuration design was chosen. This setup allows an easy change of the power stage should one of the components fail. This configuration also allowed use of multiple power stages with different devices and designs. The daughterboards interface with the motherboard with 56 pins spread

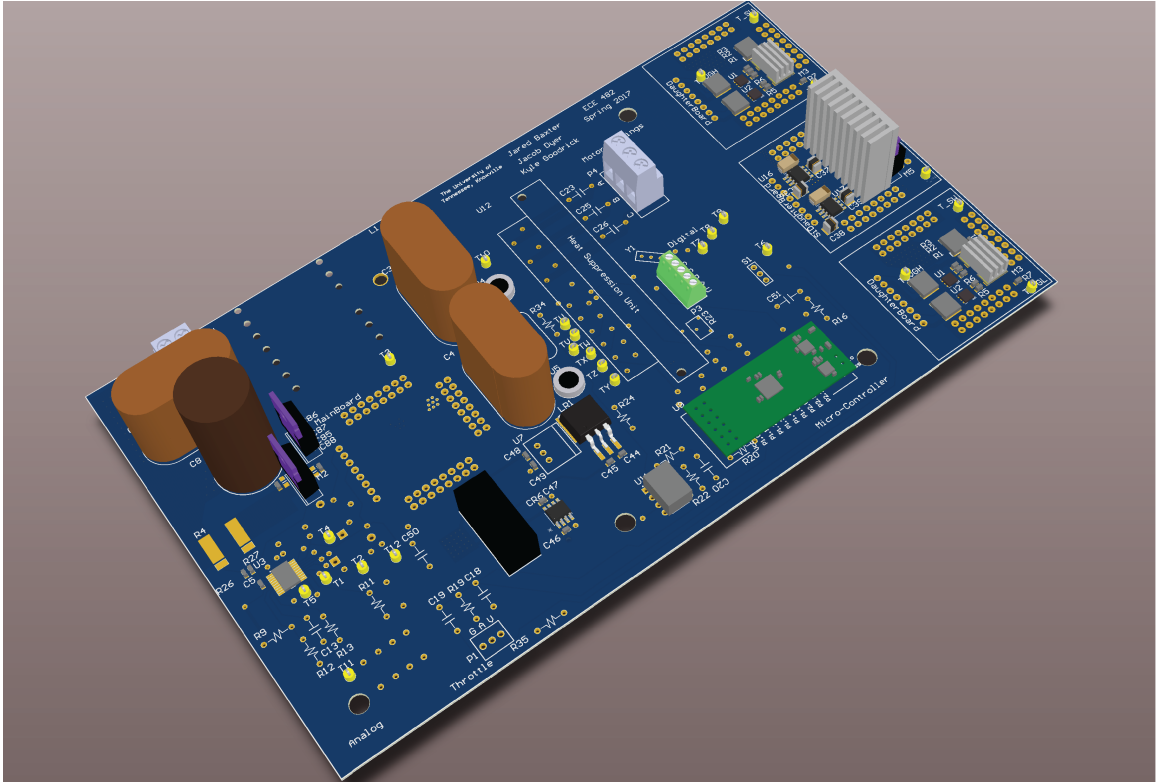


Figure 6.1: PCB 3D Model

throughout four separate header arrays. Three of the header arrays contain 16 pins and the final array contains only 8 pins. This allows the mating in only one possible way, preventing any accidental mis-connections.

Through these connectors the daughterboard receives several signal and power connections:

- High Side Gate Signal
- 5 V Isolated
- High Side Gate Signal Ground
- Low Side Gate Signal
- 5 V Non-Isolated
- Output Bus

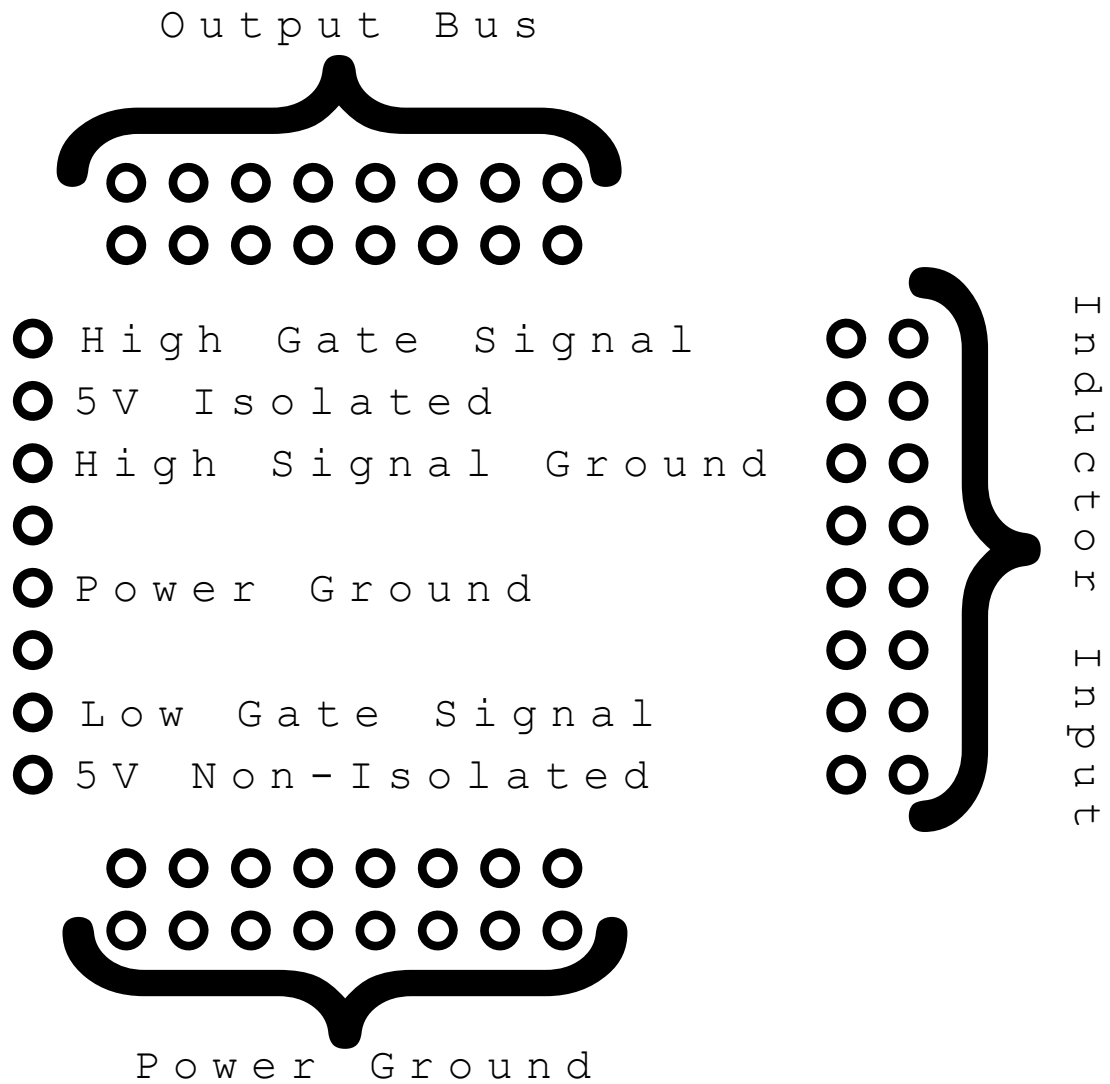


Figure 6.2: Daughterboard Connection

- Inductor Input
- Power Ground

These can be visualized in [Figure 6.2](#).

The top array of 16 pins connects directly to the converter output bus, the right array of 16 connects to the inductor, the bottom array of 16 connects to power ground,

Table 6.1: Si MOSFET Characteristics

$T_A = 25\text{ }^\circ\text{C}$		Typical Value	Unit
V_{DS}	Drain-to-Source Voltage	100	V
I_D	Maximum Continuous Drain Current	150	A
Q_g	Gate Charge Total (10 V)	78	nC
Q_{gd}	Gate Charge Gate to Drain	13	nC
$R_{DS(on)}$	Drain-to-Source On Resistance	$V_{GS} = 6\text{ V}$	3.4 m Ω
		$V_{GS} = 10\text{ V}$	3.1 m Ω
$V_{GS(th)}$	Threshold Voltage	2.7	V

and six of the eight pins in the left array connect to various gate drive power and signal connections.

6.2.1 Si Daughterboard

The primary daughterboard was one constructed with Si MOSFETs. The intent with this daughterboard was to create a converter as similar as possible to the converter constructed during experiment 3 and 4 to ensure that a functioning converter existed should the GaN board be non-functional. For this reason the same device as in lab 3 was chosen, the TI CSD19535KCS 100 V N-Channel NexFET. The key values for this device can be seen in table 6.1.

As the daughterboards are only supplied with 5 V for the gate drive the Si daughterboard contains two switched capacitor boost ICs (TI LM2662) that double the 5 V to 10 V in order to reduce the Si devices' $R_{ds(on)}$. The Si daughterboard uses one UCC27211 gate driver to drive both gates.

Table 6.2: GaN MOSFET Characteristics

$T_A = 25\text{ }^\circ\text{C}$		Typical Value	Unit
V_{DS}	Drain-to-Source Voltage	100	V
I_D	Maximum Continuous Drain Current	90	A
Q_g	Gate Charge Total (6 V)	4.5	nC
Q_{gd}	Gate Charge Gate to Drain	1.5	nC
$R_{DS(on)}$	Drain-to-Source On Resistance	$T_J = 25\text{ }^\circ\text{C}$	7 m Ω
		$T_J = 150\text{ }^\circ\text{C}$	17.5 m Ω
$V_{GS(th)}$	Threshold Voltage	1.3	V

6.2.2 GaN Daughterboard

The GaN daughterboard makes use of the GaN Systems GS61008T-E01-MR E-HEMT with the goal of using the increased switching speed and lack of reverse recovery to reduce switching losses and improve overall efficiency. The gates are driven with two separate LM5114 gate drivers using the unaltered 5 V supply voltages.

6.3 Additional PCB Design Considerations

6.3.1 Capacitors

The majority of the capacitors in this design are either ceramic or film and were chosen based solely on their voltage rating and desired capacitance. For the input capacitors, this required voltage ratings in excess of 35 V and for the output capacitors voltage ratings in excess of 100 V.

The converter was also designed with soft switching in mind. As soft switching requires a ripple current in excess of the rated battery current, it is necessary to include a large input capacitance to reduce the ripple seen by the battery. To reduce

the ripple sufficiently, a 7500 μF electrolytic capacitor (EGPD350ELL752MM40H) was chosen. This capacitor has a rated ripple voltage of 7.07 A, which is less than the potential 10 A required for soft switching; however, the max 10 A triangle current seen by the capacitor has an RMS value of $I_{RMS} = \frac{I_{pk}}{\sqrt{3}} = 5.8 \text{ A}$ which is within the capacitor specifications.

6.3.2 RCD circuit

One of the unique features implemented for this electric bike is a RCD circuit for both gate inputs to create a sufficient dead-time between the two switching devices. Dead-time is the time when both switching devices are turned off to protect against shoot-through current. With the RCD circuits, the optimal dead-time is set between the two devices to give shoot-through protection, but also keep the reverse conduction losses to a minimum. The three parts incorporated are: a basic 1N5819 Schottky diode, a 1 nF ceramic capacitor, and a 56 Ω resistor, which gives a dead-time of 56 nsec. The 1 nF capacitor was the smallest value we could use as we needed a value larger than the 110 pF parasitic capacitance associated with the Schottky diode.

6.3.3 Toshiba Motor Controller

The motor controller chosen for this electric bike is the Toshiba TB6551FAG. The motor controller takes the three hall sensors outputs from the motor and the D_{ref} output signal from the MSP430 to create the gate signals input to the motor driver. As seen in Figure 6.3, the controller generates a position signal, FG, based on the hall sensors. When this position signal reaches the frequency set by (6.1), the motor is driven by a square-wave.

$$f = \frac{f_{OSC}}{2^{12} \times 32 \times 6} \quad (6.1)$$

f_{OSC} is set by a an external 4.19MHz resonator which also sets the triangular carrier frequency to generate the PWM signals. Once the position signal reaches

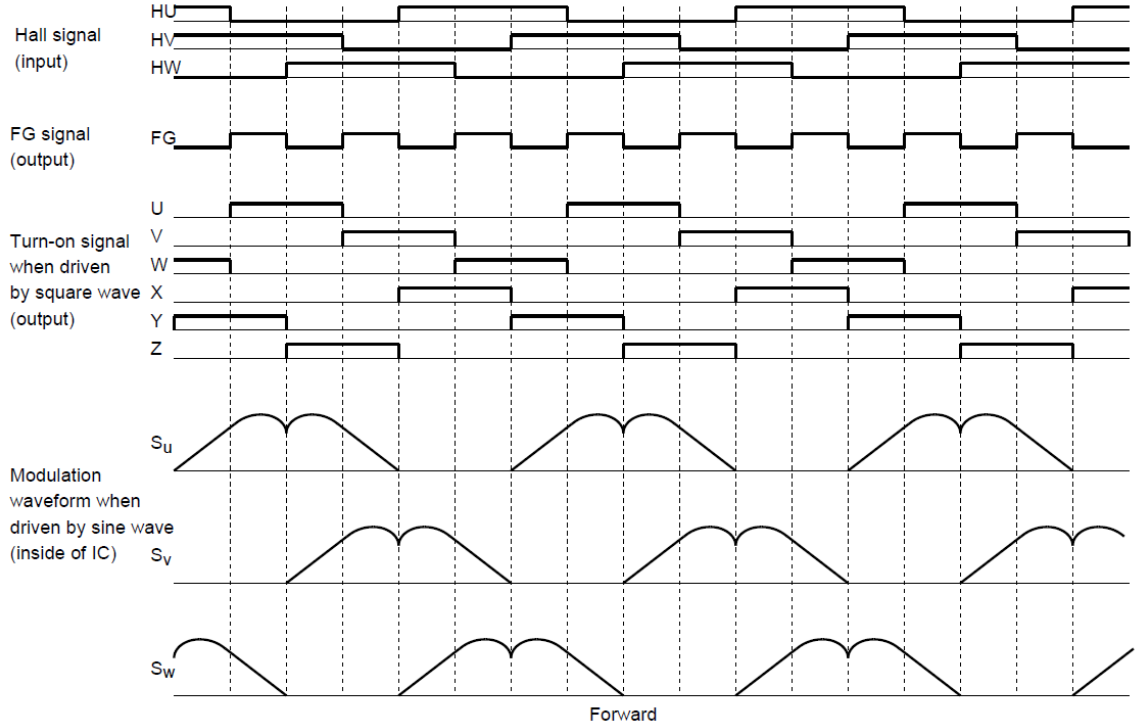


Figure 6.3: Timing chart for the Toshiba TB6551FAG

frequency f , approximately 5Hz for this circuit, the controller then drives the motor by sine-wave PWM. Note that this is the electrical signal frequency, not the mechanical frequency of the bicycle wheel. The motor chosen for this bicycle has 46 poles, so the electrical frequency is much higher than the relative mechanical frequency. An increase in throttle increases the duty cycle or modulation ratio depending on if the converter is in square-wave or sine-wave PWM respectively.

6.3.4 Infineon Motor Driver

The motor driver for this system is the International Rectifier IRAM136-3023B. This motor driver is a 30A, 150V Integrated Power Hybrid IC. The driver takes the 6 gate signals from output from the motor controller, and through an internal IC gate driver, drives the gates of the 6 internal FETs. It is operated by a 15V power supply on the motherboard, and contains features such as short circuit protection.

Chapter 7

Motor Control Design

7.1 Introduction and Objectives

Experiment 6's goal was to assemble the motor control section of the PCB and ensure that the motor control functionality worked.

7.2 Methods and Results

In experiment 6, we faced three major challenges in achieving the goal of controlling and driving the hub motor to spin the bike wheel. The first two challenges came in the central controller coding of the MSP430. We originally designed our two DACs to have a high frequency on the digital side (around 1 MHz) to reduce the time constant and keep the ripple low. However, our ADC input signal that changes the duty cycle had big jumps in duty cycle changes (around 10% shifts), which is bad as we need smooth transitions from the throttle. To fix this issue, we reduced the frequency of the digital PWM signals to 50kHz which caused us to increase the time constant to keep the ripple down, but this is an acceptable trade-off as this throttle response time doesn't need to be in the micro-second range. The second coding issue was that one of our DAC digital outputs was mapped to TimerD1 instead of TimerD0, which caused us to have to remap the output to be able to use the same capture/compare registers.

Once we finished the coding section and finishing assembling the motor section of the PCB, we started by testing the motor controller and saw there weren't any gate signals at the output. After a debug process, it was discovered that the Idc pin used to turn the gate signals off with an overcurrent was floating, and once we tied it to GND, the motor controller was operating correctly. The rest of the lab procedure went smoothly, and we were able to spin the wheel with the control of the throttle. We felt we made a good decision on choosing the motor controller and driver as they were easy to configure once we understood each pin's functionality. The only change we can think of to make in retrospect is changing the MSP430 pin used for the DAC to a pin that is 5V capable but originally mapped to TimerD0 to save us time.

7.3 Discussion and Conclusions

Overall we are very happy with how the motor controller portion of our design functions. The controller was fairly straight forward to connect and design and the motor driver itself was also quite good. Were we to do it again, we would like to add in our own current limiting circuitry to ensure that the average value of current delivered to the motor did not exceed the limitations of the battery.

Chapter 8

Final System Integration

8.1 Introduction and Objectives

Once the motor drive was functioning it was time to bring the entire system together with the battery, converter, and motor drive working simultaneously. The full system could then be used and the bike ridden.

8.2 Methods and Results

8.2.1 Experimental Results

For the experimental testing process, the boost converter section was the first to be assembled and tested. After a debug process for the boost converter and LM5121, successful full power boost was achieved in Fig. 8.1. The efficiency of the boost converter at full power is 94%. Along with confirming functionality and checking waveforms, a line regulation test was done to check the success of the close loop control. The results can be seen in Table 5.3 with very effective DC output voltage regulation.

The next step in the assembly process was putting the boost converter on a PCB. The boost converter operation was similar as before the PCB assembly. The efficiency

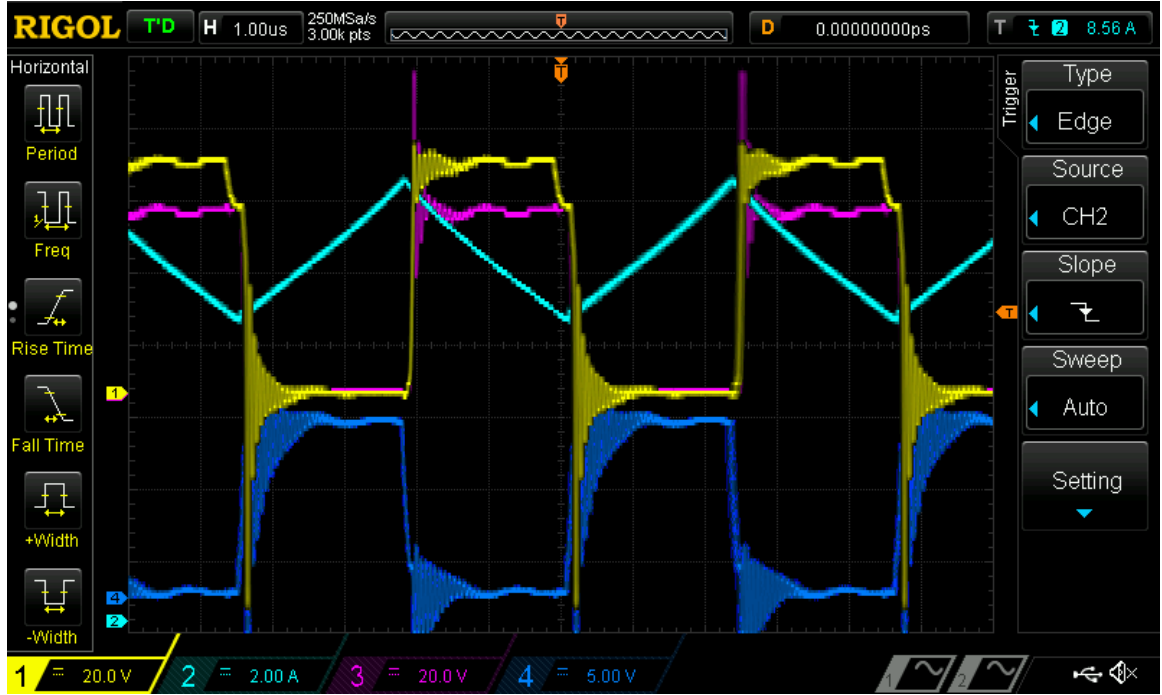


Figure 8.1: 250W Boost Converter before PCB assembly with I_{ind} (light blue), V_{gsL} (dark blue), V_{goH} (yellow), V_{sw} (pink)

at full power stayed at 94% and the ringing on the waveforms slightly decreased from reduced parasitics in PCB operation as shown in Fig. 8.2.

Another aspect of the electric bike that functionality needed to be confirmed for is the motor controller and driver. Functionality was confirmed with Fig. 8.3. The motor control and drive portion is able to take the hall sensors from the motor as an input reference and turn the boosted battery DC voltage into a AC voltage waveform shown in purple in Fig. 8.3.

All other embedded functionality is proven with the success of these results. In the end, the bike wheel could be spun with the control of the throttle and the power from the battery.



Figure 8.2: 250 W Boost Converter after PCB assembly with I_{ind} (light blue), V_{gsL} (dark blue), V_{goH} (yellow), V_{sw} (pink)

8.3 Conclusions and Future Work

Overall this converter design was a success. The design was able to achieve continuous operation at 250 W with an efficiency of 94%. There were some problems achieving this same performance consistently with a rider on the bike, but in terms of building a functional DC-DC Converter, this design achieved our goals. If we were to do this again I think there are a few key changes we would make. The first would be better protection, in our attempt to achieve soft-switching we took away some protection features in the circuit to allow for higher ripple currents. We would add in additional protection circuitry to ensure that the input current never exceeded the maximum battery current. We would also like to have boosted our gate driving voltages slightly, were we to do things again I believe we would have included trimming networks on our power supplies to increase their voltage slightly to improve switching performance in both Si and GaN boards.

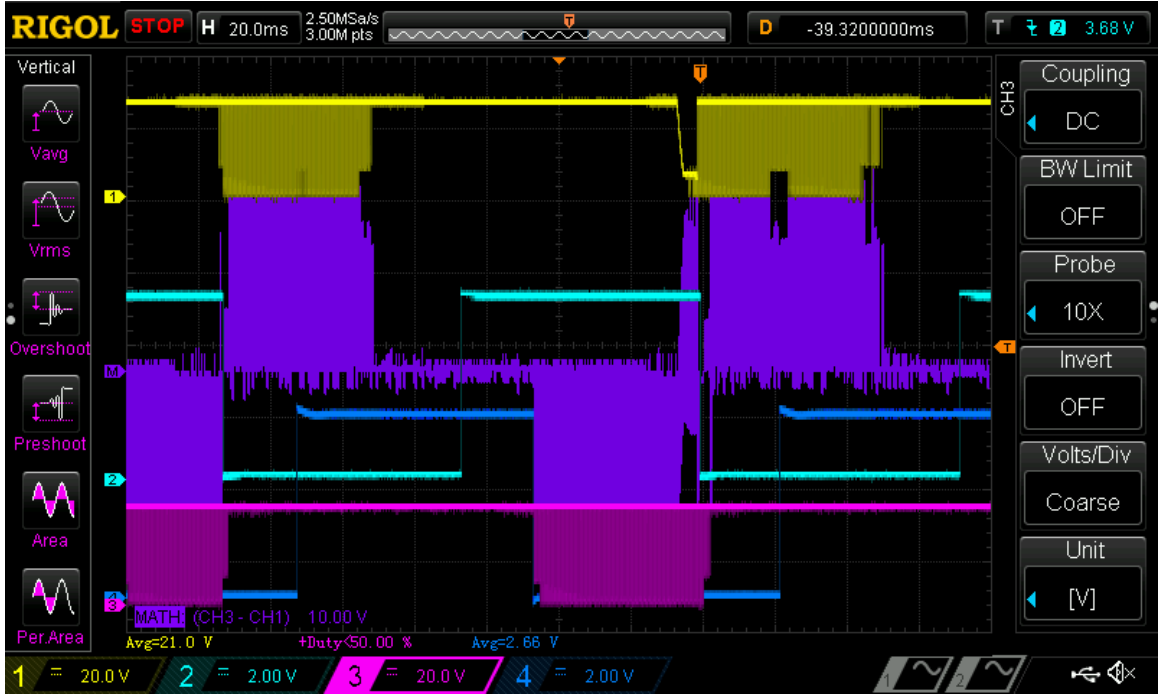


Figure 8.3: Motor Driver Output with Hall Sensors with H_c (light blue), H_b (dark blue), V_b (yellow), V_a (pink), V_{ab} (purple)

In terms of future work for the current system, the final system used Si MOSFETs as the power stage switching devices as we quickly and easily got those devices to work. However, we built GaN WBG semiconductor power stage daughterboards and in future work will implement those devices in the system. This is because the performance is better as shown in Fig. 8.4, seeing cleaner waveforms. Early in the design process, we set a goal to achieve Zero Voltage Switching (ZVS) to increase efficiency and overall performance. The issue we could not overcome was something causing the switching to stop as we were decreasing the load resistance as shown in Fig. 8.5 Another future work would be to revisit this feature to improve the boost converter portion of the system.

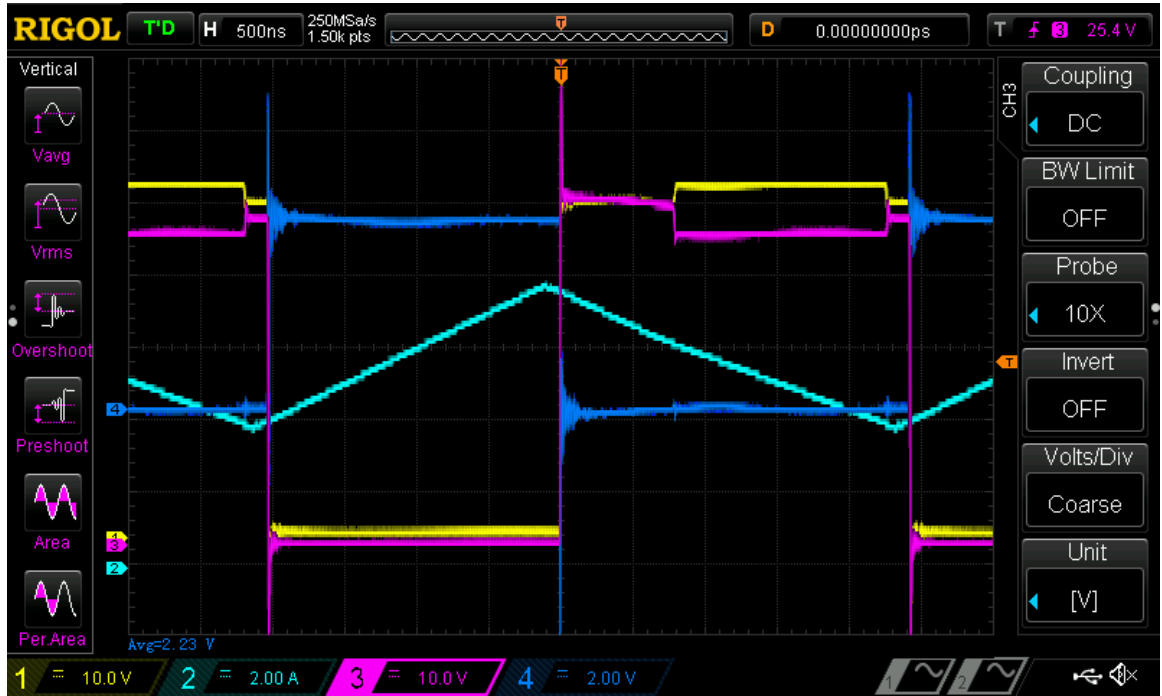


Figure 8.4: 150 W Boost Converter with GaN WBG Semiconductors with I_{ind} (light blue), V_{gsL} (dark blue), V_{goH} (yellow), V_{sw} (pink)

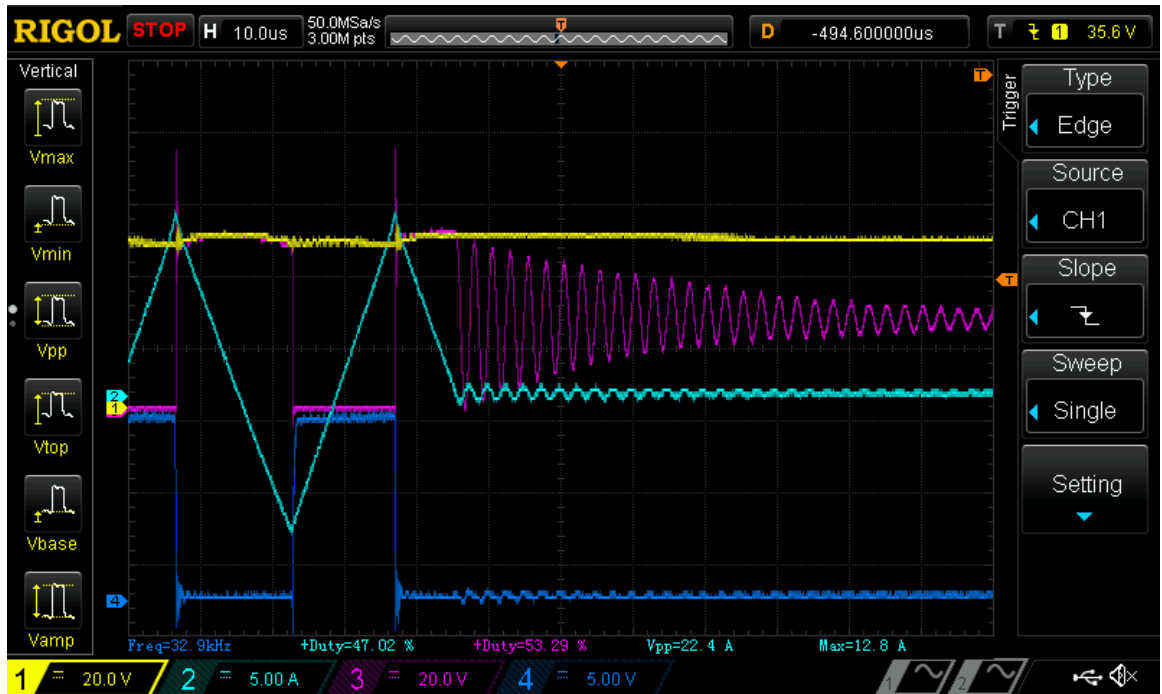


Figure 8.5: Attempt at ZVS Boost Converter with I_{ind} (light blue), V_{gsL} (dark blue), V_{sw} (pink)

Chapter 9

Conclusion

The objective of this project was to build a DC-DC Converter capable of taking a 25 V battery input, boosting it to 50 V, and then driving a three phase motor at 250 W. Now at the conclusion of this project we were able to achieve this goal and built a functioning converter that operated at 94 % efficiency. Our finished bike was rideable; however, only for a brief amount of time. There is still work to be done to increase the reliability of the bike and also to fully implement the GaN daughter board.

Appendix

Appendix A

Appendix

A.1 MSP430 Code

```
1 /* —COPYRIGHT—,BSD_EX
2  * Copyright (c) 2012, Texas Instruments Incorporated
3  * All rights reserved.
4  *
5  * Redistribution and use in source and binary forms, with or without
6  * modification, are permitted provided that the following conditions
7  * are met:
8  *
9  * * Redistributions of source code must retain the above copyright
10 *   notice, this list of conditions and the following disclaimer.
11 *
12 * * Redistributions in binary form must reproduce the above copyright
13 *   notice, this list of conditions and the following disclaimer in the
14 *   documentation and/or other materials provided with the distribution.
15 *
16 * * Neither the name of Texas Instruments Incorporated nor the names of
17 *   its contributors may be used to endorse or promote products derived
18 *   from this software without specific prior written permission.
19 *
20 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
21 * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,
22 * THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
23 * PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR
24 * CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
25 * EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
```

```

26 * PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS;
27 * OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
28 * WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
29 * OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE,
30 * EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
31 *
32 *****
33 *
34 * MSP430 CODE EXAMPLE DISCLAIMER
35 *
36 * MSP430 code examples are self-contained low-level programs that typically
37 * demonstrate a single peripheral function or device feature in a highly
38 * concise manner. For this the code may rely on the device's power-on default
39 * register values and settings such as the clock configuration and care must
40 * be taken when combining code from several examples to avoid potential side
41 * effects. Also see www.ti.com/grace for a GUI- and www.ti.com/msp430ware
42 * for an API functional library-approach to peripheral configuration.
43 *
44 * ---/COPYRIGHT---*/
45 //*****
46 // MSP430F51x2 Demo - ADC10, Sample A1, AVcc Ref, Set P1.0 if A1 > 0.5*AVcc
47 //
48 // Description: A single sample is made on A1 with reference to AVcc.
49 // Software sets ADC10SC to start sample and conversion - ADC10SC
50 // automatically cleared at EOC. ADC10 internal oscillator times sample (16x)
51 // and conversion. In Mainloop MSP430 waits in LPM0 to save power until ADC10
52 // conversion complete, ADC10_ISR will force exit from LPM0 in Mainloop on
53 // reti. If A1 > 0.5*AVcc, P1.0 set, else reset.
54 //
55 // MSP430F51x2
56 //
57 //      /\|      XIN|-
58 //      | |      |
59 //      --|RST    XOUT|-
60 //      |      |
61 //      >---|P1.1/A1      P1.0|--->LED
62 //
63 // F. Chen
64 // Texas Instruments Inc.
65 // Nov. 2012
66 // Built with CCS v5.2.1 and IAR Embedded Workbench Version: 5.51.1
67 //*****
68 #include <msp430.h>

```

```

69
70 #define CALTDH0CTL1_256      *((unsigned int *)0x1A36)
71
72 unsigned int  ADC_Result;
73 unsigned int  period;
74 unsigned int  dead_time_points;
75 double dead_time;
76 unsigned int  on_time;
77 double D;
78 double frequency;
79
80 double v_throttle;
81 double const max_throttle = 3.1;
82 double const min_throttle = 0.62;
83 double mid_throttle;
84
85 struct dutyTuple {
86     double D_Boost;
87     double D_Motor;
88 };
89
90 void Port_Mapping(void);
91 struct dutyTuple VtoD(double);
92 double bit2volt(int);
93
94 int main(void)
95 {
96     volatile unsigned int i;
97
98     WDICTL = WDIPW+WDIHOLD;           // Stop WDT
99     Port_Mapping();
100
101     // P1DIR |= BIT0;                 // Set P1.0/LED to output direction
102     // P1OUT |= BIT0;
103
104     // PMAPWD = 0x02D52;             // Enable Write-access to modify port mapping registers
105     // PMAPCTL = PMAPRECFG;          // Allow reconfiguration during runtime
106     // P1MAP1 |= PMANALOG;           // Modify all PxMAPy registers.
107     // PMAPWD = 0;                   // Disable Write-Access to modify port mapping registers
108                                     // by writing incorrect key
109     // P1SEL|=BIT1;                  // setting the port mapping register PxMAPy to PMANALOG
110                                     // together with PxSEL.y=1 when applying analog signals.

```

```

110 // Configure ADC
111 ADC10CTL0 |= ADC10SHT_2 + ADC10ON;           // ADC10ON, S&H=16 ADC clks
112 ADC10CTL1 |= ADC10SHP + ADC10DIV0 + ADC10SSEL_3;           // ADCCLK =
    MODOSC; sampling timer
113 ADC10CTL2 |= ADC10RES + ADC10PDIV0;           // 10-bit conversion results
114 ADC10MCTL0 |= ADC10INCH_1 + ADC10SREF0;      // A1 ADC input select; Vref=AVCC
115 ADC10IE |= ADC10IE0;                         // Enable ADC conv complete interrupt
116
117 // configure ports
118 // P1SEL |= BIT7+BIT6;                       // P1.7, option select
119 // P1DIR |= BIT7+BIT6;                       // P1.7 output
120 P2SEL |= BIT0;                               // P2.0 options select
121 P2DIR |= BIT0;                               // P2.0 output
122 P2SEL |= BIT7;                               // P2.7 options select
123 P2DIR |= BIT7;                               // P2.7 output
124
125 // Configure TimerD0 in Hi-Res Regulated Mode
126 // TD0CTL0 = TDSSEL_2;                       // TDCLK=SMCLK=12MHz=Hi-Res input clk
    select
127
128 TD0CTL1 |= TDCLKM1;                           // Select Hi-res local clock
129 TD0HCTL0 = TDHM1 + TDHEN;                     // Hi-res clock 16x TDCLK,
130 TD0HCTL1 = CALTDH0CTL1_256;
131
132 // Configure TD0
133 frequency = 50e3;
134 dead_time = 500e-9;
135
136 //dead_time_points = (256e6 / 2) * dead_time;
137
138 period = 256e6 / (2 * frequency);
139
140
141 TD0CTL0 = TDSSEL_2 + MC_3 + TDCLR;           // SMCLK, up-down mode, clear TDR
142 TD0CCR0 = period;                             // PWM Period/2
143 TD0CCTL1 = OUTMOD_2;                          // CCR1 toggle/set
144 TD0CCTL2 = OUTMOD_2;                          // CCR2 toggle/set
145
146 struct dutyTuple DutyCycles;
147
148 for (;;)
149 {
150     __delay_cycles(5000);

```

```

151 ADC10CTL0 |= ADC10ENC + ADC10SC;           // Sampling and conversion start
152 __bis_SR_register(CPUOFF + GIE);           // LPM0, ADC10_ISR will force exit
153 __no_operation();                           // For debug only
154 v_throttle = bit2volt(ADC_Result);          // Set Duty Cycle
155 DutyCycles = VtoD(v_throttle);
156
157 TD0CCR2 = DutyCycles.D_Boost * period;
158 TD0CCR1 = DutyCycles.D_Motor * period;
159
160
161 // on_time = D * period;
162 // if (on_time > 125)
163 // {
164 //     on_time = 0;
165 // }
166 // on_time = on_time + 1;
167 // TD0CCR1 = on_time; //+ dead_time_points;           // CCR1 PWM duty
cycle
168 // TD0CCR2 = on_time; // - dead_time_points;         // CCR2 PWM duty cycle
169 }
170 }
171
172 // ADC10 interrupt service routine
173 #if defined(__TI_COMPILER_VERSION__) || defined(__IAR_SYSTEMS_ICC__)
174 #pragma vector=ADC10_VECTOR
175 __interrupt void ADC10_ISR(void)
176 #elif defined(__GNUC__)
177 void __attribute__((interrupt(ADC10_VECTOR))) ADC10_ISR (void)
178 #else
179 #error Compiler not supported!
180 #endif
181 {
182     switch(__even_in_range(ADC10IV,12))
183     {
184         case 0: break;           // No interrupt
185         case 2: break;           // conversion result overflow
186         case 4: break;           // conversion time overflow
187         case 6: break;           // ADC10HI
188         case 8: break;           // ADC10LO
189         case 10: break;          // ADC10IN
190         case 12: ADC_Result = ADC10MEM0;
191                 __bic_SR_register_on_exit(CPUOFF);
192                 break;           // Clear CPUOFF bit from 0(SR)

```

```

193     default: break;
194 }
195 }
196
197 void Port_Mapping(void)
198 {
199     __disable_interrupt();           // Disable Interrupts before altering
        Port Mapping registers
200     PMAPPWD = 0x02D52;               // Enable Write-access to modify port
        mapping registers
201
202     #ifdef PORT_MAP_REC_CFG
203     PMACTL = PMAPRECFG;              // Allow reconfiguration during runtime
204     #endif
205
206     // P1MAP0 = PM_TA0_0;
207     // P1MAP1 = PM_TA0_0;
208     // P1MAP2 = PM_TA0_1;
209     // P1MAP3 = PM_TA0_1;
210     // P1MAP4 = PM_TA0_2;
211     // P1MAP5 = PM_TA0_2;
212
213     P2MAP7 = 8;
214
215     PMAPPWD = 0;                     // Disable Write-Access to modify port
        mapping registers
216     #ifdef PORT_MAP_EINT
217     __enable_interrupt();            // Re-enable all interrupts
218     #endif
219 }
220
221 struct dutyTuple VtoD(double vt)
222 {
223     // Returns the duty cycles required to set the reference voltages for the
224     // Motor Driver and the boost converter
225
226     struct dutyTuple D_result;
227
228     double V_Bus;
229     double D_motor;
230     double V_boost_ref;
231
232     // Setup Throttle

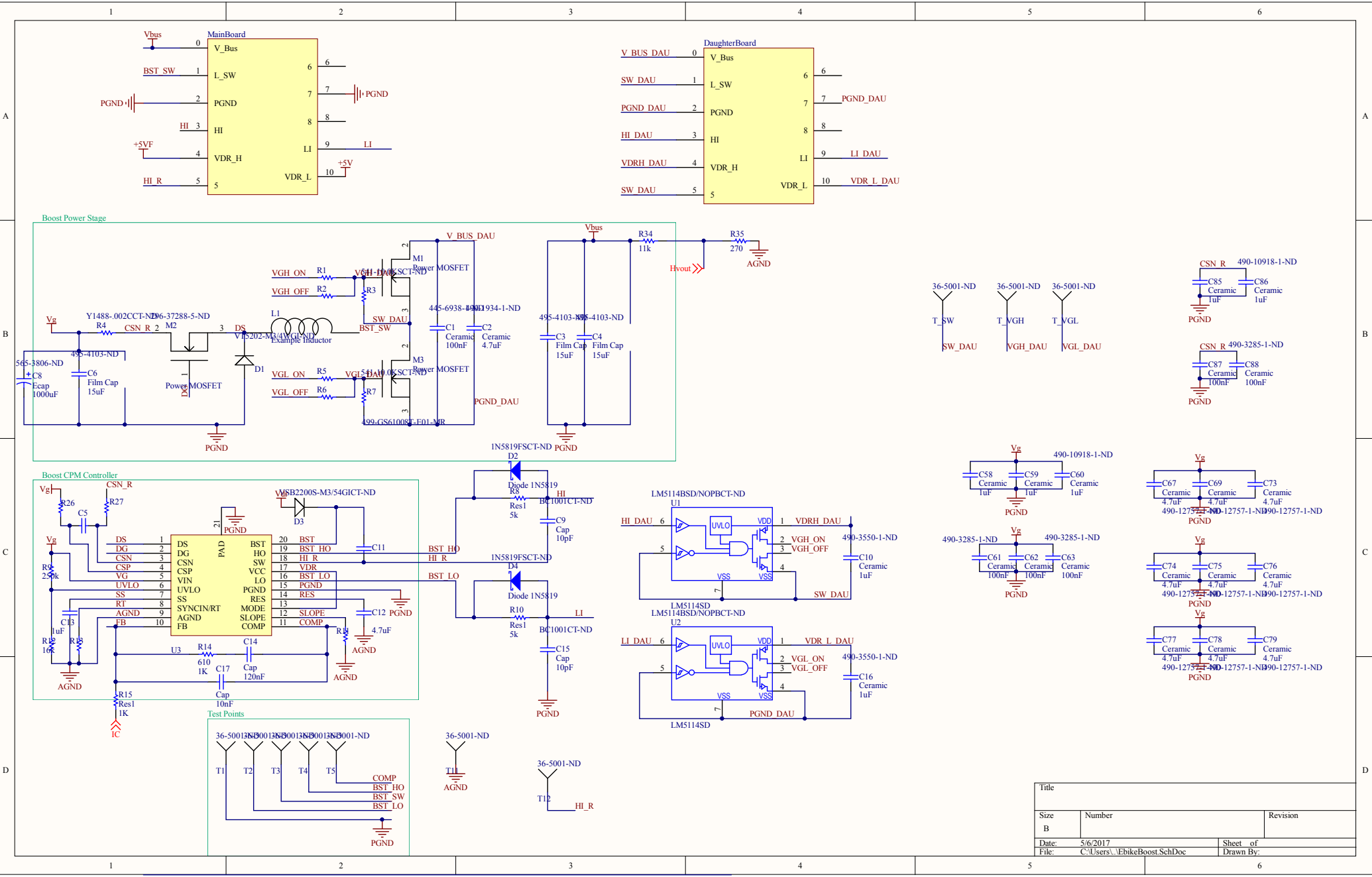
```

```

233     mid_throttle = ((max_throttle - min_throttle) / 3) + min_throttle;
234
235     if (vt < min_throttle)
236     {
237         D_motor = .05;
238         V_Bus = 26;
239     }
240     else if (vt < mid_throttle)
241     {
242         D_motor = (vt - min_throttle) * (0.90 / (mid_throttle - min_throttle)) +
243         0.05;
244         V_Bus = 26;
245     }
246     else if (vt < max_throttle)
247     {
248         D_motor = 0.95;
249         V_Bus = (vt - mid_throttle) * (24 / (max_throttle - mid_throttle)) + 26;
250     }
251     else
252     {
253         D_motor = 0.95;
254         V_Bus = 50;
255     }
256
257     V_Bus = V_Bus * 1.02;
258
259     V_boost_ref = (1.2 / 50.0) * V_Bus;
260
261     D_result.D_Boost = V_boost_ref / 5.0;
262     D_result.D_Motor = D_motor;
263
264     return D_result;
265 }
266
267 double bit2volt(int bitNum)
268 {
269     double voltage;
270
271     voltage = (3.3 / 1024) * bitNum;
272
273     return voltage;
274 }

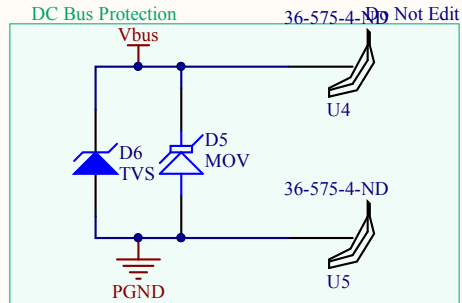
```

A.2 Altium Schematics

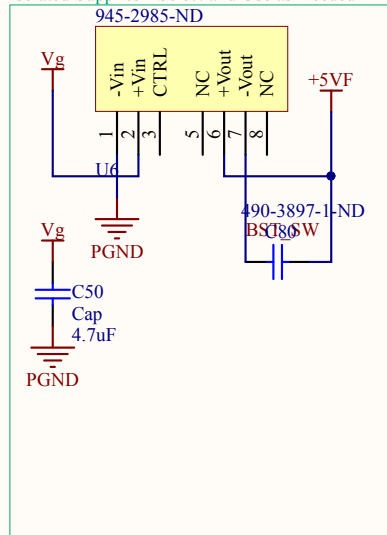


Title		
Size	Number	Revision
B		
Date:	5/6/2017	Sheet of
File:	C:\Users\Ebike\Boost.SchDoc	Drawn By:

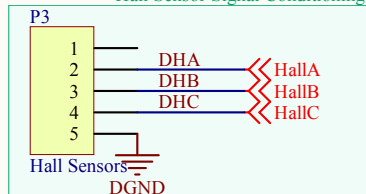
DC Bus Protection



Isolated Supplies - Select and Use as Needed

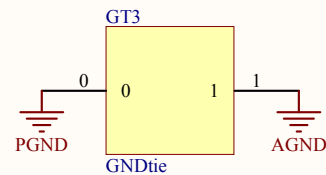
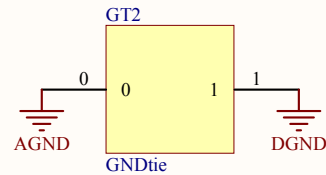
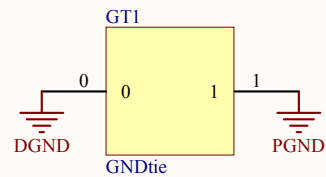
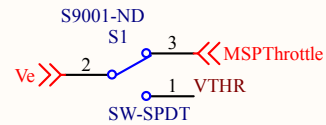
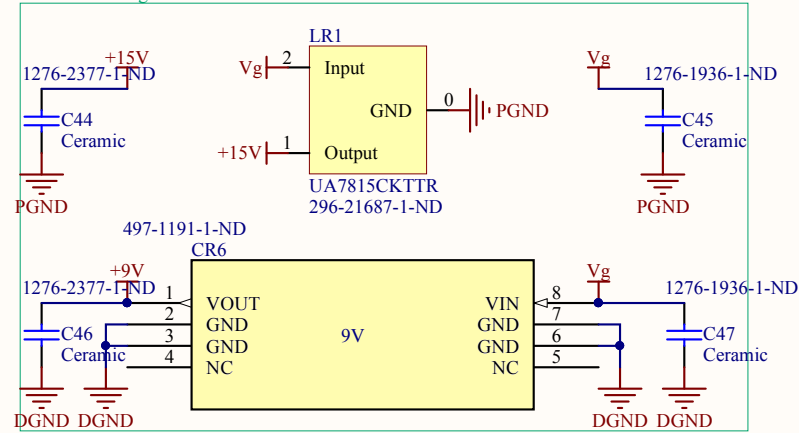


Hall Sensor Signal Conditioning

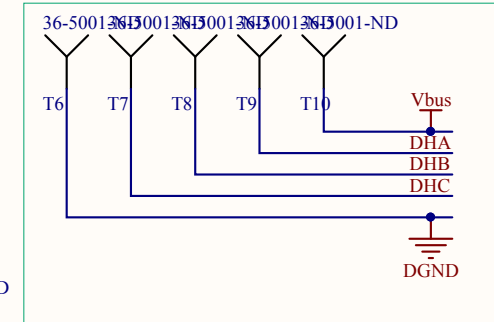


Note: Buffered outputs

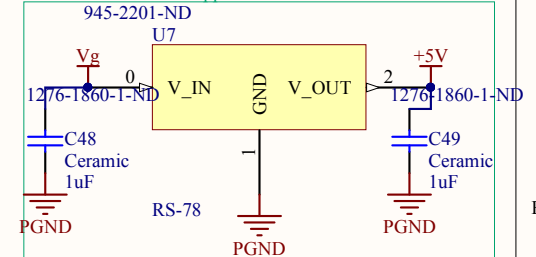
Linear Regulators



Test Points

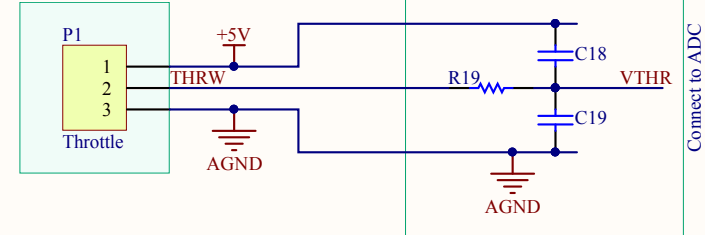


Non-Isolated Power Supplies

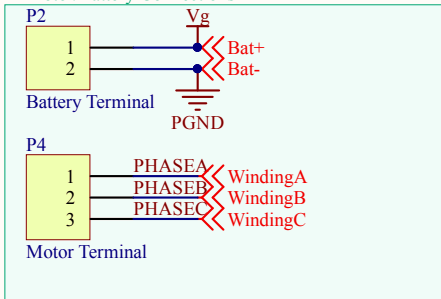


Throttle Signal Conditioning

Throttle Connection



Motor/Battery Connections



Title

Size

A

Number

Date:

5/6/2017

File:

C:\Users\...\EbikeConnectionAndProtectionSchDocBy:

Revision

Sheet of

A

A

B

B

C

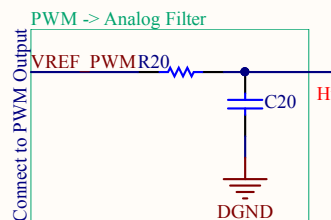
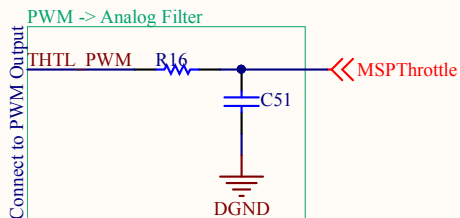
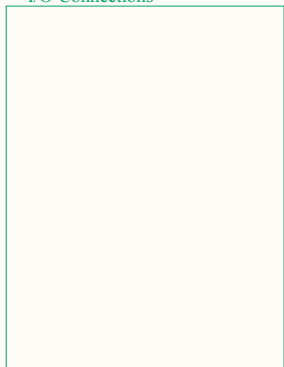
C

D

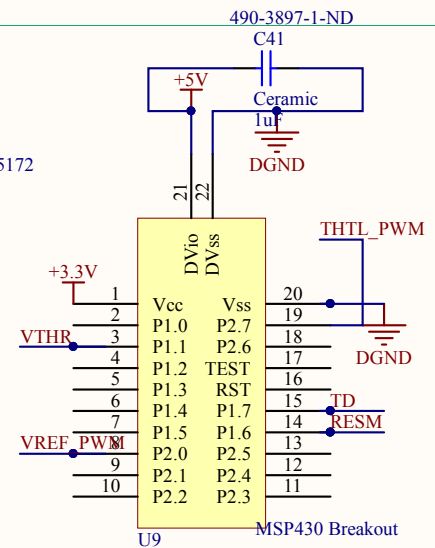
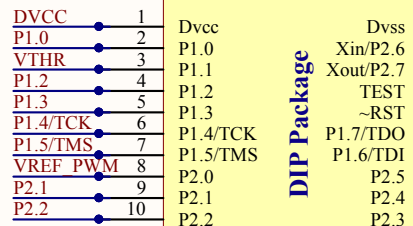
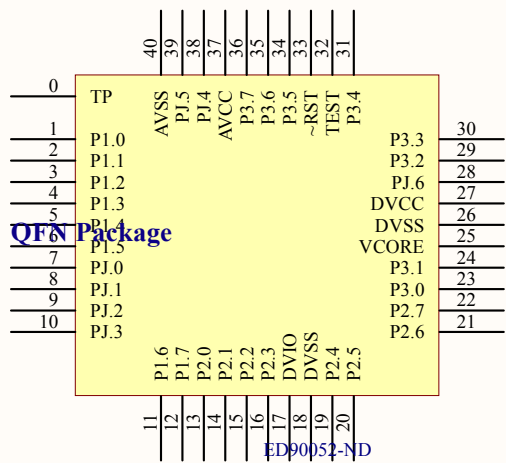
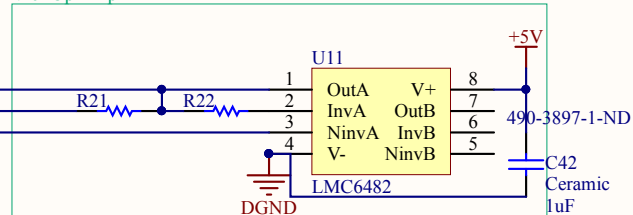
D

Microcontroller -- Choose which one (or more) to use

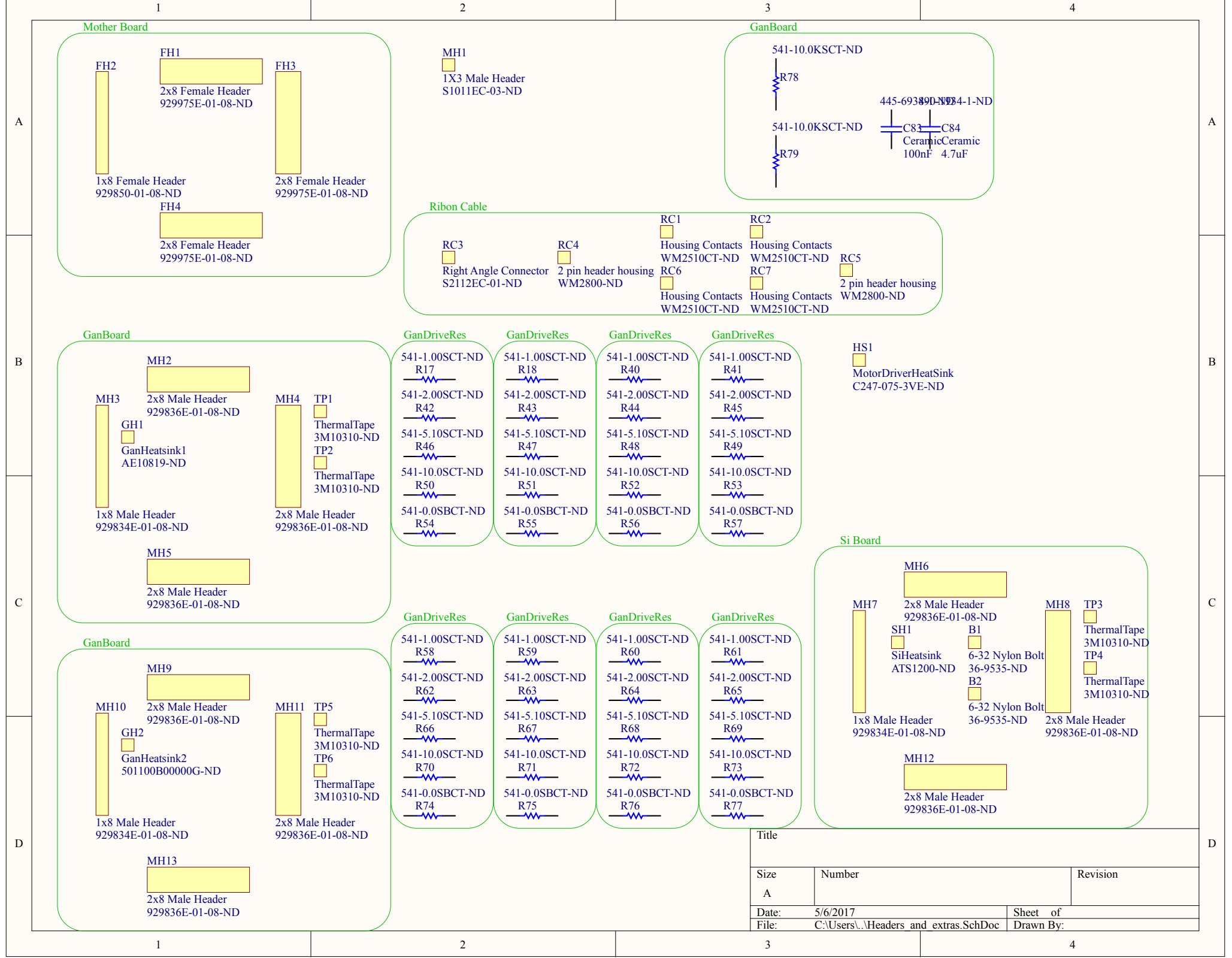
I/O Connections

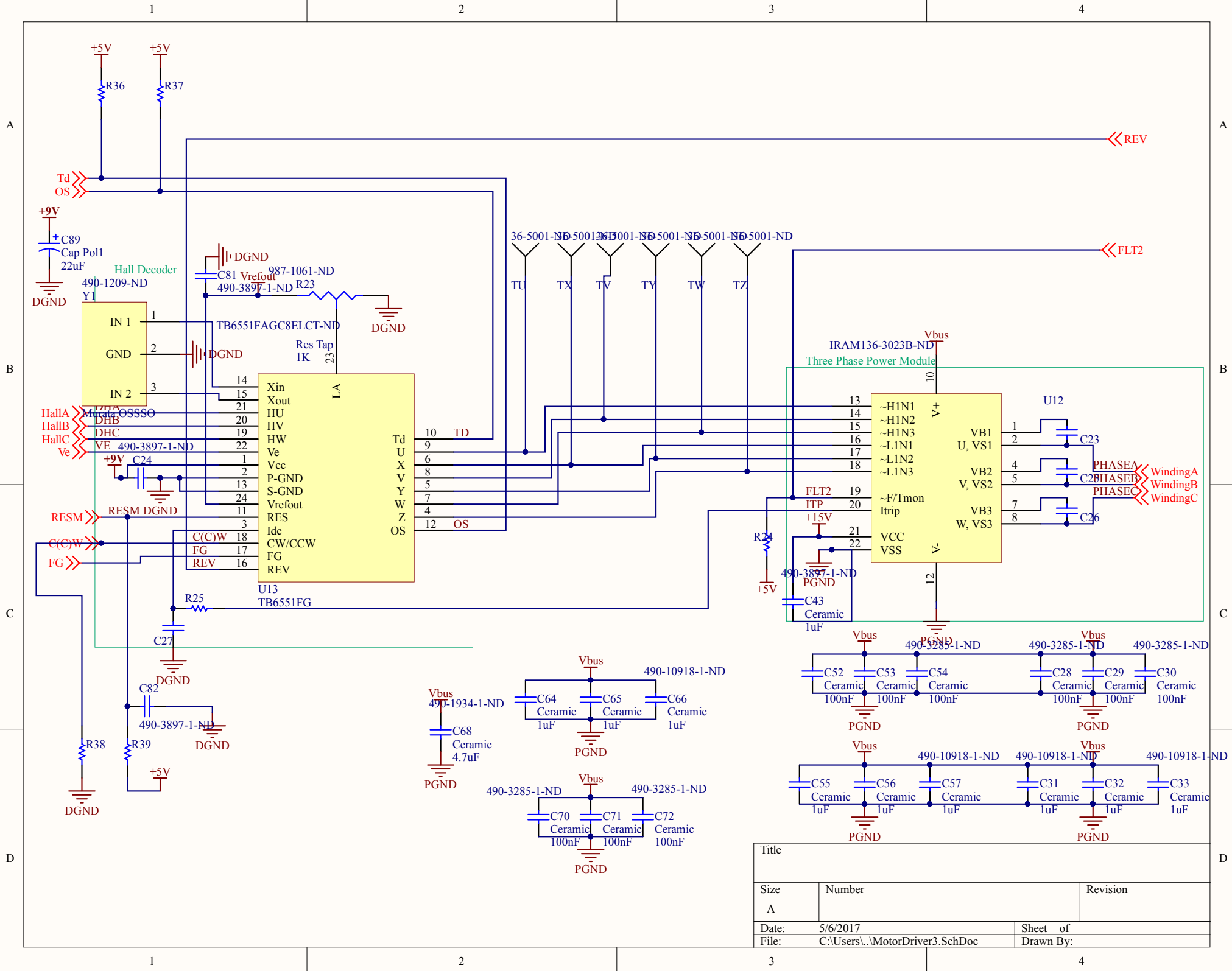


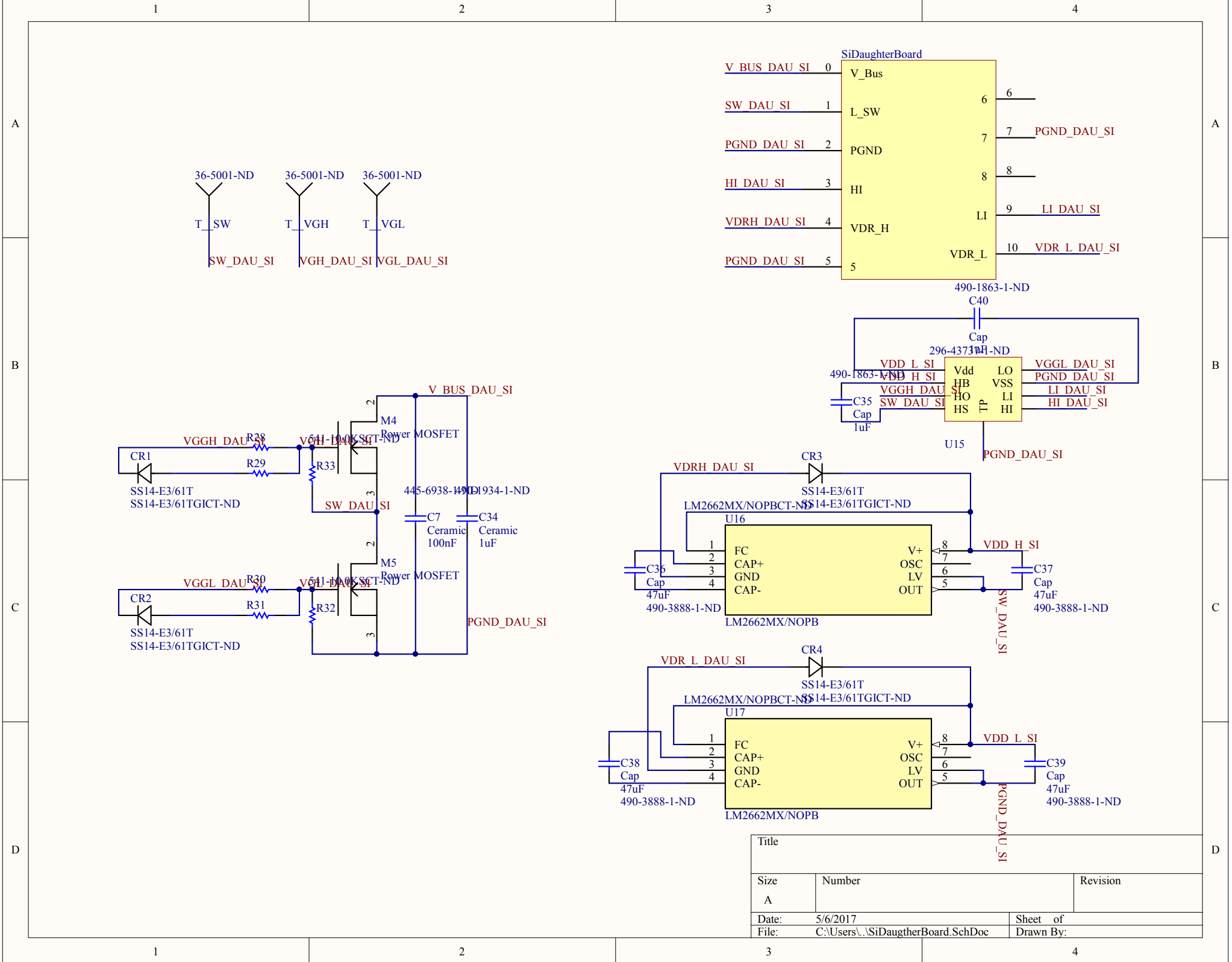
Vref Op Amp



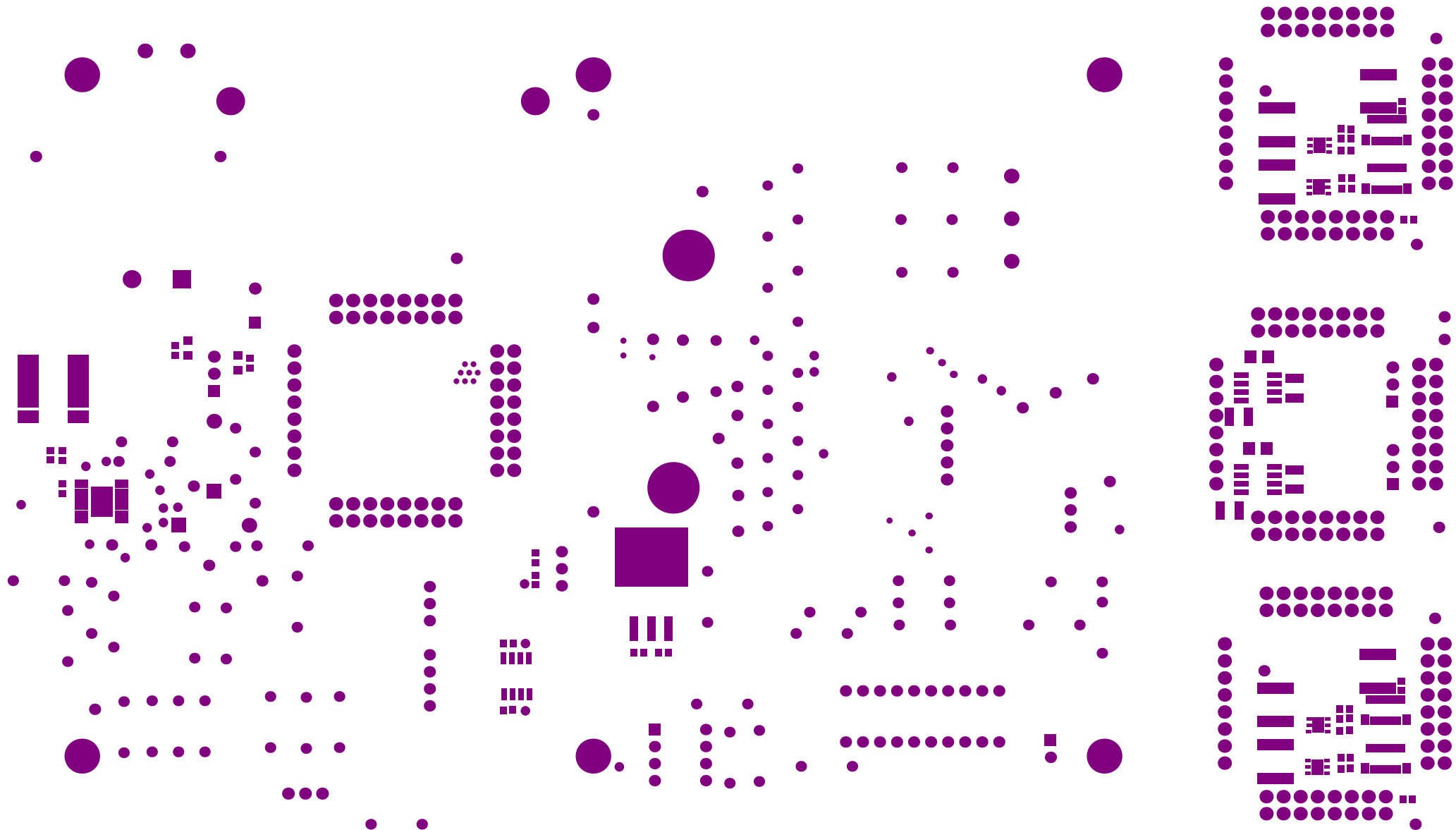
Title		
Size	Number	Revision
A		
Date:	5/6/2017	Sheet of
File:	C:\Users\...\EbikeController.SchDoc	Drawn By:

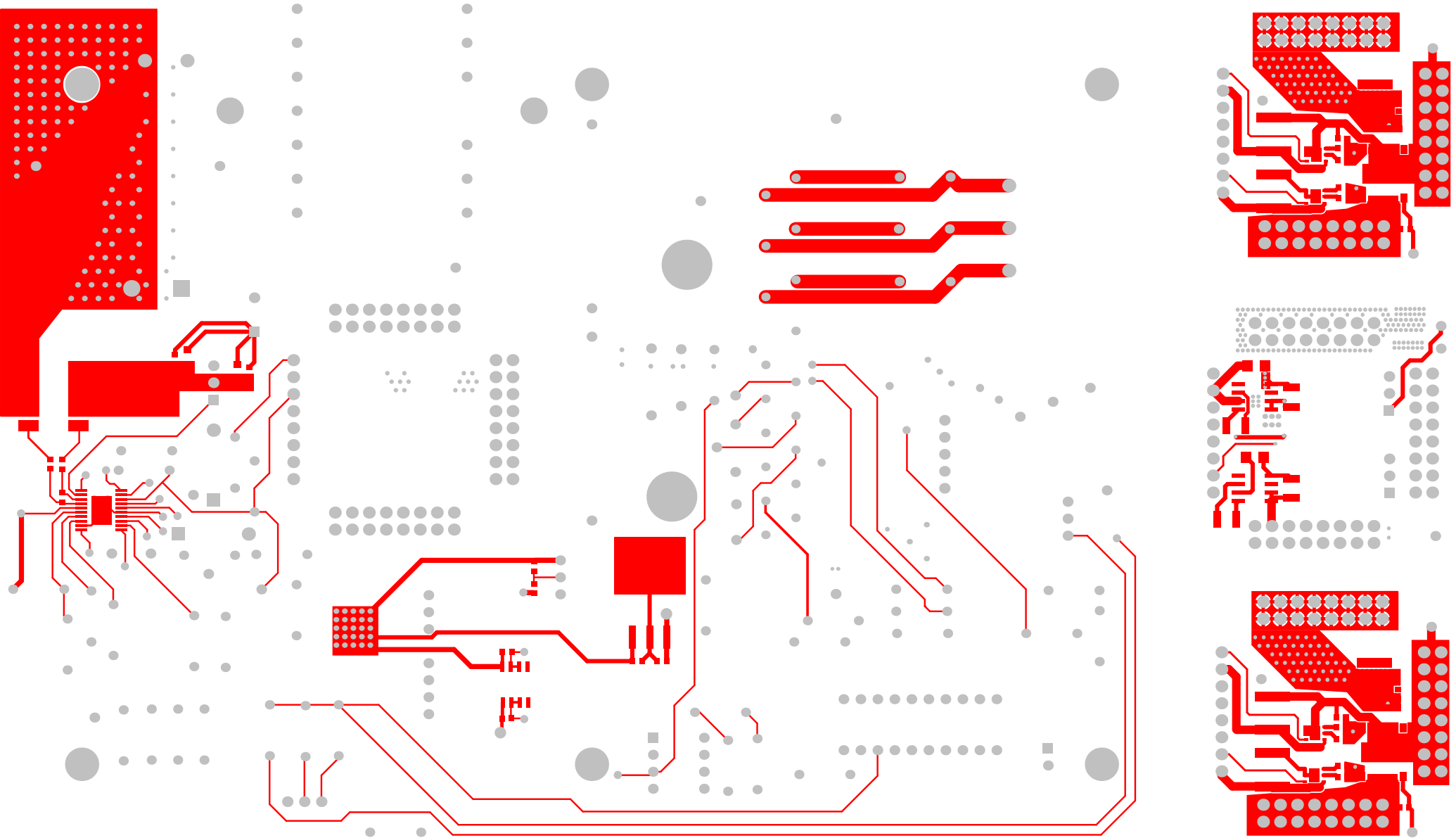


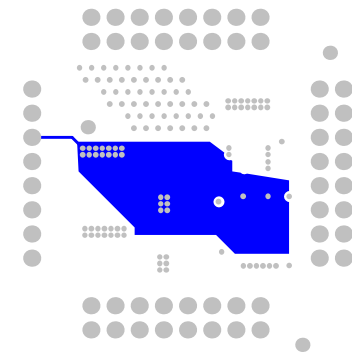
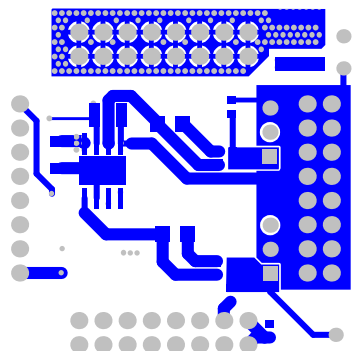
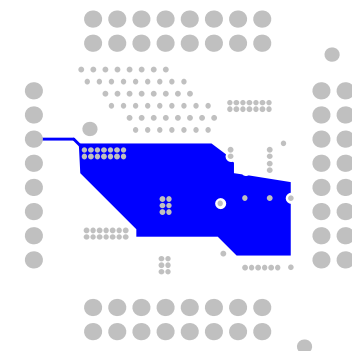
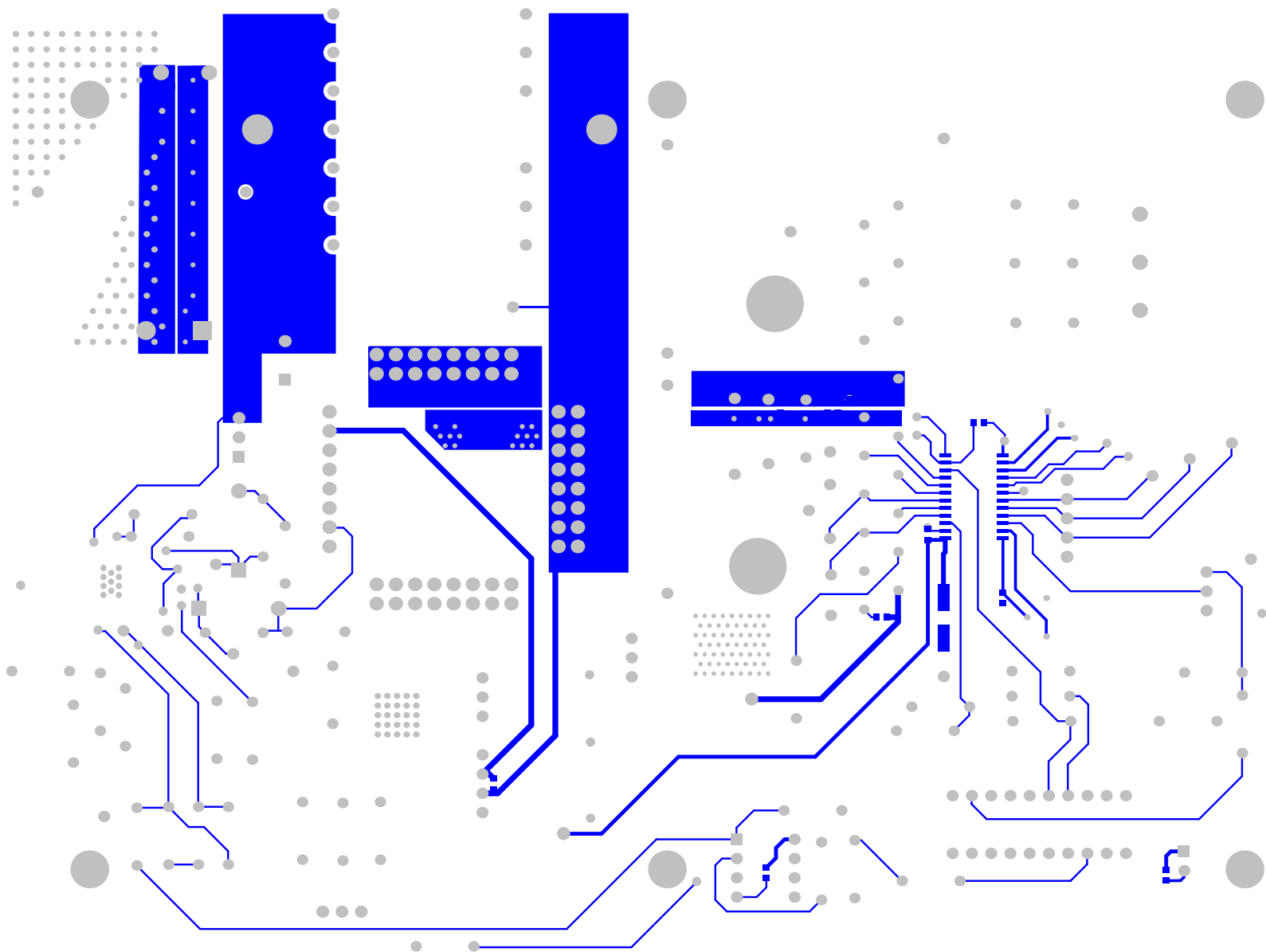


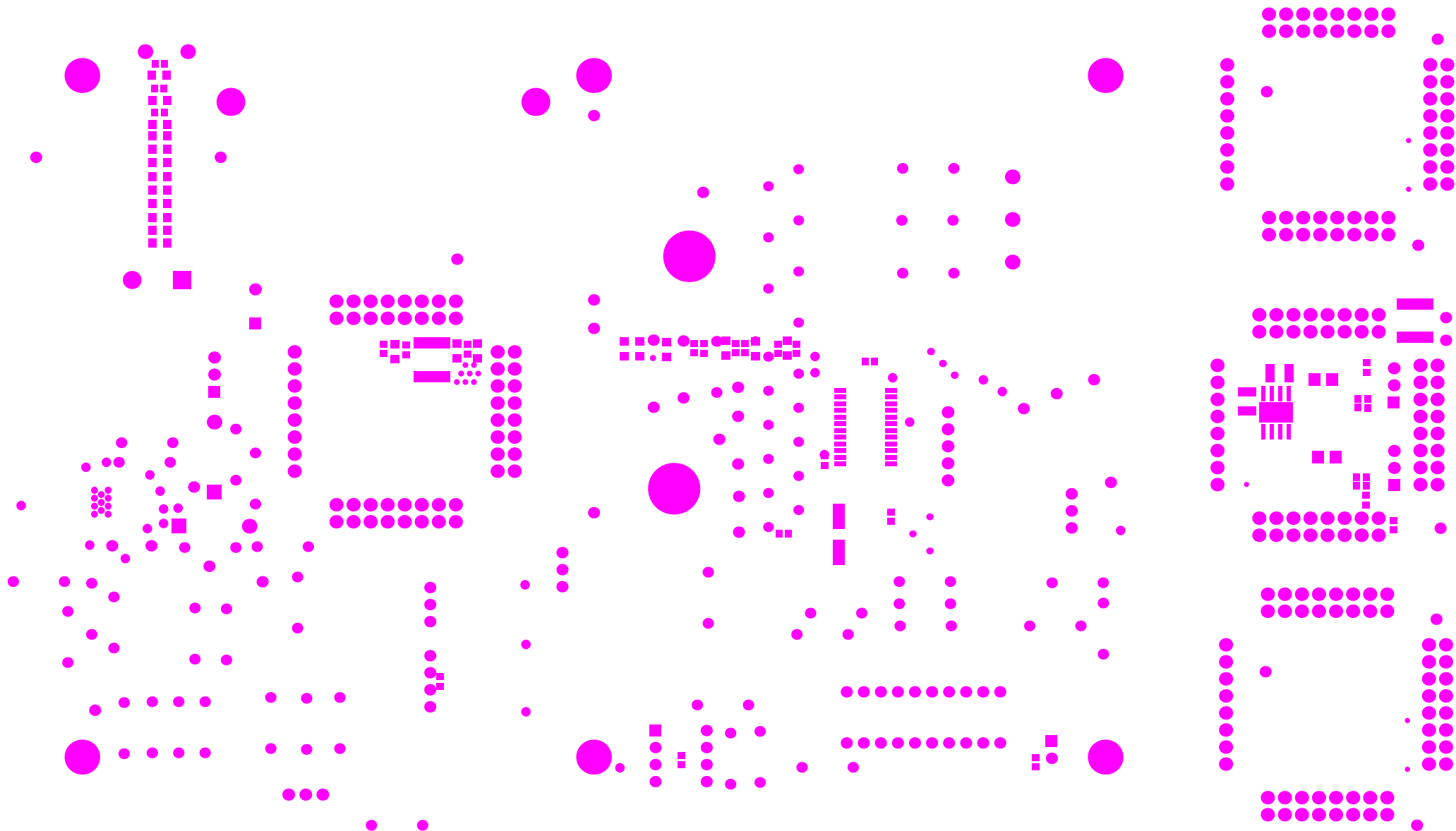


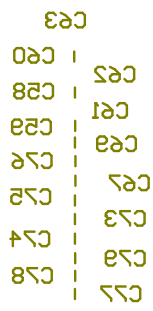
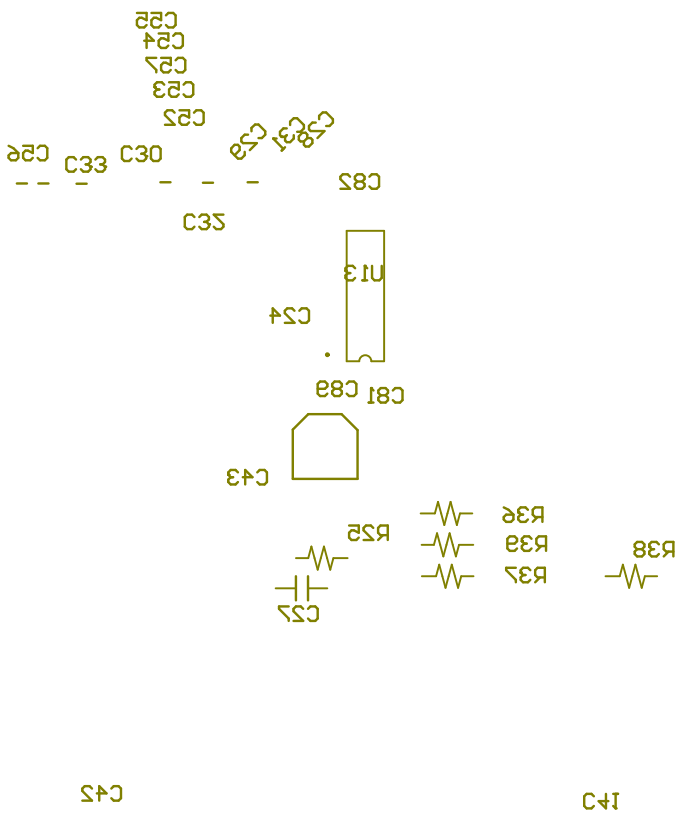
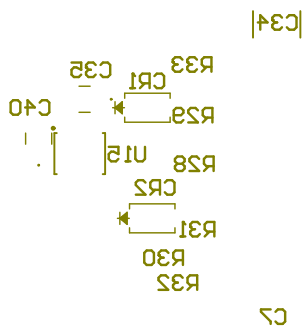
A.3 Altium PCB Layout

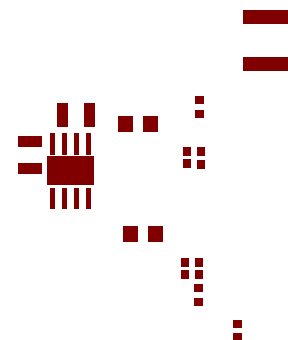
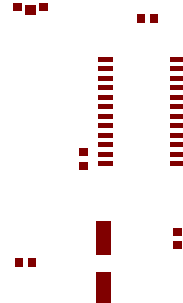


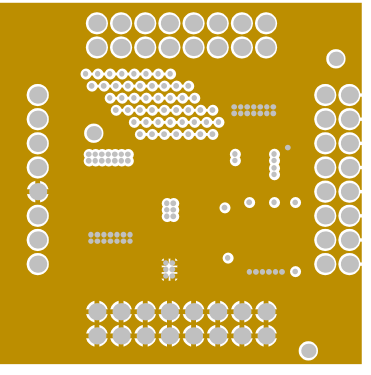
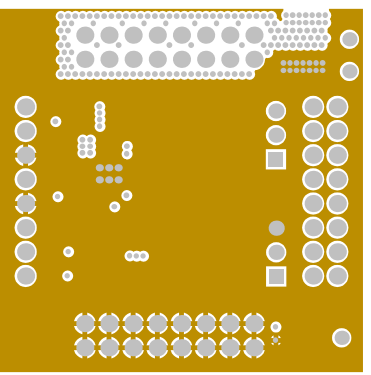
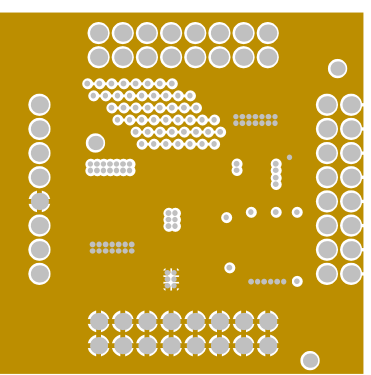
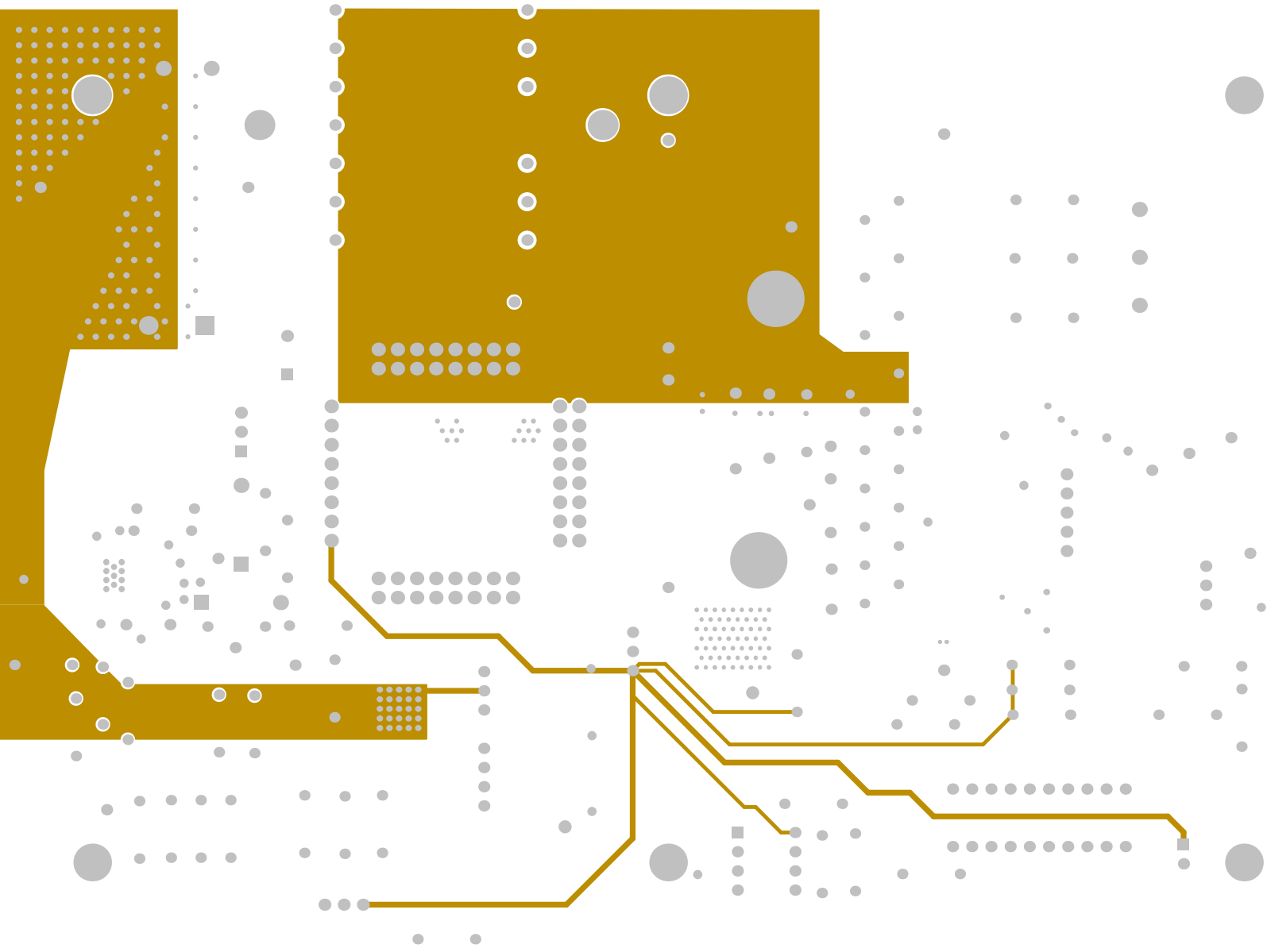


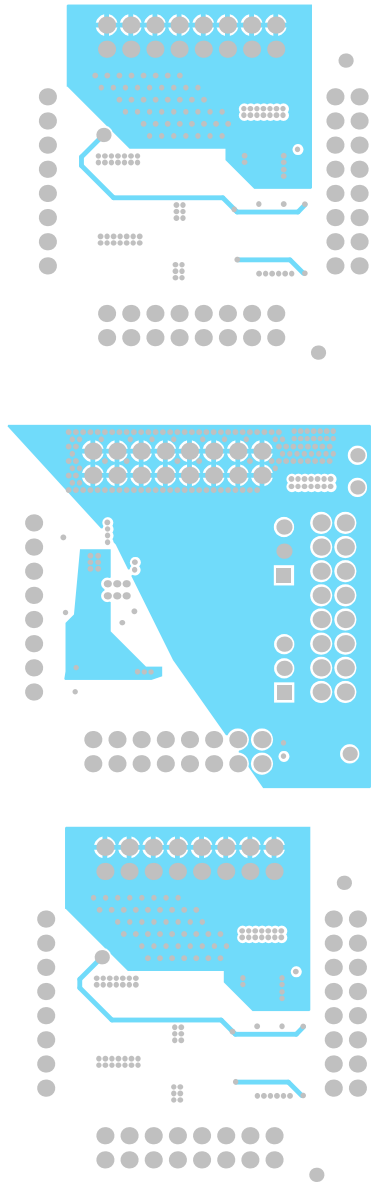
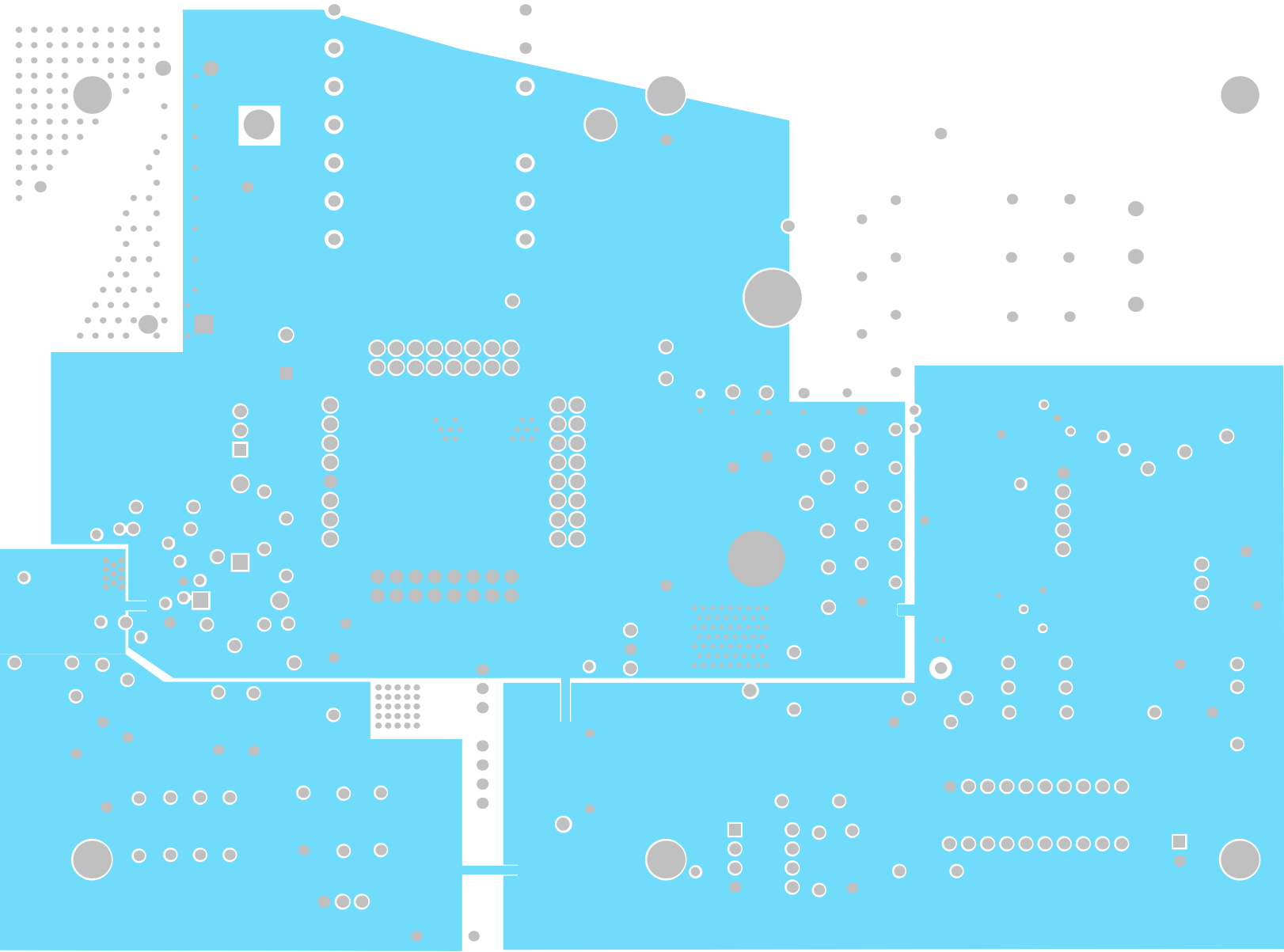












A.4 Bill of Materials

Comment	Description	Designator	Quantity	Supplier 1	Supplier Part Number 1	Supplier Unit Price 1	Manufacturer Part	Supplier Order Qty 1	Supplier Subtotal 1
IRAM136-3023B	3-Phase Motor Driver Module	U12	1	Digi-Key	IRAM136-3023B-ND	29.63	IRAM136-3023B	2	59.26
Film Cap	Capacitor	C3, C4, C6	3	Digi-Key	495-4103-ND	5.71	B32524Q1156K	6	34.26
Iso Supply	RSO-24xxS	U6	1	Digi-Key	945-2985-ND	16.22	RS6-2405S	2	32.44
Ceramic	Capacitor	C2, C34, C68, C84	4	Digi-Key	490-1934-1-ND	2.46	GRM55ER72A475KA01L	8	19.68
2x8 Male Header		MH2, MH4, MH5, MH6	9	Digi-Key	929836E-01-08-ND	0.992	929836-01-08-RK	18	17.856
Power MOSFET	Multiple	M2, M4, M5	3	Digi-Key	296-37288-5-ND	2.97	CSD19535KCS	6	17.82
Res Semi	Semiconductor Resistor	R4	1	Digi-Key	Y1488-.002CCT-ND	8.24	Y14880R00200F9R	2	16.48
MotorDriverHeatSink		HS1	1	Digi-Key	C247-075-3VE-ND	8.07	C247-075-3VE	2	16.14
Net	Keystone 500(0-4) Series	T1, T2, T3, T4, T5, T6, T7	24	Digi-Key	36-5001-ND	0.2298	5001	50	11.49
SiHeatsink		SH1	1	Digi-Key	ATS1200-ND	5.6	ATS-54250D-C1-R0	2	11.2
2x8 Female Header		FH1, FH3, FH4	3	Digi-Key	929975E-01-08-ND	1.78	929975-01-08-RK	6	10.68
LM5121	V/I Boost Controller With Disconnect	U3	1	Digi-Key	296-44178-1-ND	5.04	LM5121MHX/NOPB	2	10.08
UCC27211A	120V Gate Driver	U15	1	Digi-Key	296-43737-1-ND	4.35	UCC27211AQDDARQ1	2	8.7
Ecap	Polarized Capacitor (Radial)	C8	1	Digi-Key	565-3806-ND	4.34	EGPD350ELL752MM40H	2	8.68
LM2662MX/NOPB	Imported	U16, U17	2	Digi-Key	LM2662MX/NOPBCT-ND	2.05	LM2662MX/NOPB	4	8.2
Ceramic	Capacitor	C31, C32, C33, C55, C56, C57, C58, C59, C60, C64, C65, C66, C85, C86	14	Digi-Key	490-10918-1-ND	0.2752	GRJ21BC72A105KE11L	28	7.7056
ThermalTape		TP1, TP2, TP3, TP4, TP5	6	Digi-Key	3M10310-ND	0.63	15MM-15MM-10-8810	12	7.56
RS-78		U7	1	Digi-Key	945-2201-ND	3.62	R-78E5.0-1.0	2	7.24
LM5114SD	eGaN Gate Driver	U1, U2	2	Digi-Key	LM5114BSD/NOPBCT-ND	1.64	LM5114BSD/NOPB	4	6.56
Ceramic	Capacitor	C67, C69, C73, C74, C75, C76, C77, C78, C79	9	Digi-Key	490-12757-1-ND	0.356	GRM21BC71H475KE11L	18	6.408
Cap	Capacitor	C36, C37, C38, C39	4	Digi-Key	490-3888-1-ND	0.634	GRM32ER61C476ME15L	10	6.34
TB6551FG	3-Phase Full-Wave Sine-Wave PWM Brushless Motor Controller	U13	1	Digi-Key	TB6551FAGC8ELCT-ND	3.07	TB6551FAG,C,8,EL	2	6.14
Hall Sensors	Header, 5-Pin	P3	1	Digi-Key	A98336-ND	3.06	282834-5	2	6.12
Motor Terminal	Header, 3-Pin	P4	1	Digi-Key	277-1270-ND	2.1	1714968	2	4.2
Banana Jack	575-8	U4, U5	2	Digi-Key	36-575-4-ND	0.99	575-4	4	3.96
Res Tap	Tapped Resistor	R23	1	Digi-Key	987-1061-ND	1.85	64WR10KLF	2	3.7
1x8 Male Header		MH3, MH7, MH10	3	Digi-Key	929834E-01-08-ND	0.6	929834-01-08-RK	6	3.6
LMC6482	Dual Op-Amp Rail-to-Rail	U11	1	Digi-Key	LMC6482IN/NOPB-ND	1.73	LMC6482IN/NOPB	2	3.46
1x8 Female Header		FH2	1	Digi-Key	929850-01-08-ND	1.72	929850-01-08-10	2	3.44
Ceramic	Capacitor	C10, C16	2	Digi-Key	490-3550-1-ND	0.85	GRM55DR72E105KW01L	4	3.4
Throttle	Header, 3-Pin	P1	1	Digi-Key	A98334-ND	1.62	282834-3	2	3.24
Res Semi	Semiconductor Resistor	R17, R18, R40, R41, R58, R59, R60, R61	8	Digi-Key	541-1.00SCT-ND	0.197	CRCW06031R00FKEAHP	16	3.152
Res Semi	Semiconductor Resistor	R42, R43, R44, R45, R62, R63, R64, R65	8	Digi-Key	541-2.00SCT-ND	0.197	CRCW06032R00FKEAHP	16	3.152
Battery Terminal	Header, 2-Pin	P2	1	Digi-Key	277-1269-ND	1.56	1714955	2	3.12
Ceramic	Capacitor	C28, C29, C30, C52, C53, C54, C61, C62, C63, C70, C71, C72, C97, C98	14	Digi-Key	490-3285-1-ND	0.1088	GRM188R72A104KA35D	28	3.0464
SS14-E3/61T	Imported	CR1, CR2, CR3, CR4	4	Digi-Key	SS14-E3/61TGICT-ND	0.3	SS14-E3/61T	10	3
MSP430G2X33	MCU (Launchpad)	U10	1	Digi-Key	ED90052-ND	1.41	110-44-320-41-001000	2	2.82
Res Semi	Semiconductor Resistor	R28, R50, R51, R52, R53, R70, R71, R72, R73	9	Digi-Key	541-10.0SCT-ND	0.149	CRCW060310R0FKEAHP	18	2.682
Res Semi	Semiconductor Resistor	R46, R47, R48, R49, R66, R67, R68, R69	8	Digi-Key	541-5.10SCT-ND	0.149	CRCW06035R10FKEAHP	16	2.384

Res Semi	Semiconductor Resistor	R54, R55, R56, R57, R74, R75, R76, R77	8	Digi-Key	541-0.05BCT-ND	0.129	CRCW06030000Z0EAHP	16	2.064
Diode 1N5819	Schottky Rectifier	D2, D4	2	Digi-Key	1N5819FSCT-ND	0.45	1N5819	4	1.8
Res Semi	Semiconductor Resistor	R3, R7, R32, R33, R78, R	6	Digi-Key	541-10.0KSCT-ND	0.149	CRCW060310K0FKEAHP	12	1.788
UA7815CKTTR		LR1	1	Digi-Key	296-21687-1-ND	0.86	UA7815CKTTR	2	1.72
Diode		D1	1	Digi-Key	VT5202-M3/4WGI-ND	0.79	VT5202-M3/4W	2	1.58
Cap Pol1	Polarized Capacitor (Radial)	C89	1	Digi-Key	493-10170-1-ND	0.73	UZT1E220MCL1GB	2	1.46
Cap	Capacitor	C35, C40	2	Digi-Key	490-1863-1-ND	0.35	GRM32RR71H105KA01L	4	1.4
2 pin header housing		RC4, RC5	2	Digi-Key	WM2800-ND	0.33	0050579002	4	1.32
GanHeatsink2		GH2	1	Digi-Key	501100B00000G-ND	0.62	501100B00000G	2	1.24
Ceramic	Capacitor	C1, C7, C83	3	Digi-Key	445-6938-1-ND	0.19	CGA3E3X7S2A104K080AB	6	1.14
GanHeatsink1		GH1	1	Digi-Key	AE10819-ND	0.53	V5618A	2	1.06
Diode		D3	1	Digi-Key	VSΒ2200S-M3/54GICT-ND	0.52	VSΒ2200S-M3/54	2	1.04
Housing Contacts		RC1, RC2, RC6, RC7	4	Digi-Key	WM2510CT-ND	0.11	0016020086	8	0.88
Cap	Capacitor	C9, C15	2	Digi-Key	BC1001CT-ND	0.21	K1001J5C0GF5TL2	4	0.84
Right Angle Connector		RC3	1	Digi-Key	S2112EC-01-ND	0.39	PREC001DBAN-M71RC	2	0.78
	Imported	CR6	1	Digi-Key	497-1191-1-ND	0.36	L78L09ACD13TR	2	0.72
6-32 Nylon Bolt		B1, B2	2	Digi-Key	36-9535-ND	0.15	9535	4	0.6
Ceramic	Capacitor	C48, C49	2	Digi-Key	1276-1860-1-ND	0.15	CL10A105KB8NNNC	4	0.6
Cap Semi, Ceramic, Ceramic, Ceramic, Cap Semi, Cap Semi, Cap Semi	Capacitor (Semiconductor SIM Model), Capacitor, Capacitor, Capacitor, Capacitor (Semiconductor SIM Model), Capacitor (Semiconductor SIM Model), Capacitor (Semiconductor SIM Model)	C24, C41, C42, C43, C80, C81, C82	7	Digi-Key	490-3897-1-ND	0.04	GRM188R61E105KA12D	14	0.56
Murata OSSSO		Y1	1	Digi-Key	490-1209-ND	0.22	CSTLS4M19G53-B0	2	0.44
Ceramic	Capacitor	C44, C46	2	Digi-Key	1276-2377-1-ND	0.037	CL10F474ZB8NNNC	10	0.37
1X3 Male Header		MH1	1	Digi-Key	S1011EC-03-ND	0.18	PRPC003SAAN-RC	2	0.36
SW-SPDT	SPDT Subminiature Toggle Switch, Right Angle Mounting, Vertical Actuator	S1	1	Digi-Key	S9001-ND	0.13	SPC02SYAN	2	0.26
Ceramic	Capacitor	C45, C47	2	Digi-Key	1276-1936-1-ND	0.025	CL10B104KB8SFNC	10	0.25
Ceramic	Capacitor	C5	1	Digi-Key	720-1373-1-ND	0.1	VJ0603Y101KXACW1BC	2	0.2
Res Semi	Semiconductor Resistor	R26, R27	2	Digi-Key	311-100GRCT-ND	0.01	RC0603JR-07100RL	10	0.1
Cap, Cap, Cap, Cap, Cap, Cap Semi, Cap Semi, Cap Semi, Cap Semi, Cap Semi, Cap Semi	Capacitor, Capacitor, Capacitor, Capacitor, Capacitor, Capacitor (Semiconductor SIM Model), Capacitor (Semiconductor SIM Model), Capacitor (Semiconductor SIM Model), Capacitor (Semiconductor SIM Model), Capacitor (Semiconductor SIM Model), Capacitor (Semiconductor SIM Model), Capacitor (Semiconductor SIM Model), Capacitor (Semiconductor SIM Model), Capacitor (Semiconductor SIM Model)	C11, C12, C13, C14, C17, C18, C19, C20, C23, C25, C26, C27, C50, C51	14						
DaughterBoard		DaughterBoard, MainBo	3						
Example Inductor		L1	1						
GNDtie		GT1, GT2, GT3	3						
MOV	Tranzorb	D5	1						
MSP430 Breakout	MSP430G5172 breakout from ECE482	U9	1						
MSP430F5172	MCU (HRPWM)	U8	1						
Power MOSFET	Multiple	M1, M3	2						
Res Semi	Semiconductor Resistor	R1, R2, R5, R6, R29, R30	7						
Res1, Res1, Res1, Res1, Res1, Res1, Res1, Res1, Res Semi, Res Semi, Res Semi, Res Semi, Res Semi, Res Semi, Res Semi, Res Semi, Res Semi, Res Semi, Res Semi, Res Semi	Resistor, Resistor, Resistor, Resistor, Resistor, Resistor, Resistor, Resistor, Semiconductor Resistor, Semiconductor Resistor, Semiconductor Resistor, Semiconductor Resistor, Semiconductor Resistor, Semiconductor Resistor, Semiconductor Resistor, Semiconductor Resistor, Semiconductor Resistor, Semiconductor Resistor, Semiconductor Resistor, Semiconductor Resistor	R8, R9, R10, R11, R12, R	21						
TVS	Zener Diode	D6	1						

Vita

Kyle J. Goodrick was born in Knoxville, TN on October 22, 1993. He grew up in Knoxville, TN and attended Hardin Valley Academy graduating in May 2012. Since August 2012 Kyle has attended The University of Tennessee, Knoxville for his Bachelor of Science degree in Electrical Engineering with a minor in German. He will graduate in May 2017 Summa Cum Laude. During his undergraduate studies Kyle held internships with Oak Ridge National Lab and Siemens AG. He also participated in undergraduate research under Dr. Daniel Costinett conducting research focused on characterization of Wide Band Gap Semiconductors. After graduation Kyle plans to pursue a PhD in Electrical engineering at the University of Colorado, Boulder with a focus in Power Electronics.