COMPACT AND EFFICIENT POWER ELECTRONICS WITH APPLICATIONS TO BATTERY MANAGEMENT SYSTEMS

BY

DEREK HEEGER

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2016

Urbana, Illinois

Adviser:

Assistant Professor Robert Pilawa-Podgurski

ABSTRACT

This work investigates the use of advanced power electronics techniques for a variety of applications to both improve efficiency and decrease the size. The first area of research is on investigating limitations in high density switched-capacitor converters for voltage step-up applications. The results from this could be particularly useful for pulsed power applications. This work uses techniques such as resonance, advanced control, and interleaving in the Dickson converter to avoid common limitations in switched-capacitor circuits. Another area of research is on fast battery charging using active battery management system topologies. These topologies have been proven before but this work expands upon those by using modern power electronics techniques to minimize the size and maximize the efficiency. This is achieved by using high frequency, GaN switches, planar magnetics, and active core resetting in a forward converter. This system was developed to be isolated and bi-directional so multiple active battery management system topologies could be used from the same design.

To my family and Allison.

ACKNOWLEDGMENTS

I would like to thank my parents for their continual love and support, and for teaching me how to work hard. Thank you to my fiancé Allison for supporting me for the last 7 years and making every day better. Thank you to my adviser, Robert Pilawa, for making my graduate school an enjoyable experience. I have learned tons from him about both the technical and non-technical side of being a good researcher. I would like to thank the Pilawa research group for making this graduate school experience amazing. This group makes working endless hours fun. I would especially like to thank Enver Candan, Ben Macy, and Andrew Stillwell for spending so much time working on projects with me and teaching me how to be a good engineer. I would also like to thank Mike Partridge for the ongoing support at both Sandia and Illinois.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF ABBREVIATIONS

ADC	Analog-to-Digital Converter
BMS	Battery Management System
GaN	Gallium Nitride
OV	Overvoltage
UV	Undervoltage
SC	Switched-Capacitor
SMPC	Switch Mode Power Converter
SOC	State of Charge
ZCS	Zero Current Switching
ZVS	Zero Voltage Switching

CHAPTER 1

INTRODUCTION AND MOTIVATION

1.1 High Energy Density Switched-Capacitor Converters

Power electronics take up an increasing portion of portable systems as other elements such as the transistor shrink at a fast rate. The goal of this work was to develop smaller power electronics circuits without reducing efficiency. This work focuses specifically on converters that step up the voltage which would be applicable to pulsed power applications. Pulsed power applications use high voltages to discharge energy quickly and are commonly used in particle accelerators, fusion research, lasers, water purification, and radar. Commercial capacitors have a higher energy density than inductors at low voltages; therefore, this work investigates the use of switched-capacitor circuits for this application.

Power densities and efficiency in SC converters are predominantly limited by their poor capacitor utilization, switching losses, and capacitor-to-capacitor charging while switching. This work uses techniques to avoid all three of these loss mechanisms. A power converter is built expanding on the work in [1], which achieved an extremely high power density. The key differences are that this converter's conversion ratio is 1:6 instead of 1:4 and it interleaves two modules. Building the higher step-up ratio is important for understanding how this design will expand to even higher conversion ratios.

The converter in this research utilizes a small inductor to achieve resonant zero current switching (ZCS), which reduces switching losses and increases capacitor utilization. Advanced split-phase control [2, 1] operates the switches in a sequence that minimizes capacitor-to-capacitor charging. Implementing split-phase control is a difficult task due to the precise timing requirements of the different switches. This work expands upon this challenge by implement-

ing split-phase control of two interleaved converters, which requires precise synchronization between modules to achieve an accurate 180° phase shift.

Interleaving multiple converters is extremely useful for scaling the power output of a system. In interleaved designs involving any dc-dc converter, current sharing is an issue that needs to be addressed. If one phase can carry substantially more current than the other, the components will need to be overrated, which reduces the power density of the converter. While current sharing in conventional (e.g., buck, boost, etc.) multi-phase power converters is well understood, current sharing in interleaved SC converters (and resonant SC converters) has not been explored.

A preliminary 20 V to 120 V interleaved converter prototype is implemented with the goal of demonstrating interleaved, resonant operation in a split-phase controlled Dickson converter. Moreover, it serves as an experimental verification of the current sharing analysis. Split-phase control with a master and slave micro-controller was successfully implemented at 200 kHz. Resonance was achieved while using this control scheme and allowed for operation at a peak efficiency of 95.7%.

1.2 Active Battery Management Systems

Lithium-ion batteries are becoming more common with applications such as home energy storage, electric cars, and laptop computers. A limitation in using these is their ability to charge very quickly. An individual battery cell is capable of receiving large amounts of current, but the batteries run into issues when they are stacked in series. A common lithium-ion battery cell has a nominal voltage of 3-4 V. Therefore, a high voltage battery pack is many individual cells stacked in series. The issue with charging them in series is that different cells have minor internal differences leading to cells charging faster than other cells, causing voltage imbalances. Excessive voltage on a battery cell degrades its health, and in some cases can cause fires. A similar deviation can also happen during discharge, causing an undervoltage which can be just as damaging to the battery.

These two conditions are avoided by using a battery management system (BMS) which monitors the cell voltages at all times. If an overvoltage (OV) condition is approaching, it disconnects the power and burns energy off the

high voltage cells. If the voltage of a single cell is approaching an undervoltage condition, it simply shuts off the path to the load. These operations are how a passive battery management system operates.

An active battery management system uses power electronics on every single cell. They can intelligently shift power from cell to cell depending on the conditions. For example, when a cell is about to have an OV condition, the power converter will remove energy from the high voltage cell and move it to the lower voltage cells. Similarly during discharge, energy can be shifted to the lowest voltage cells from the highest voltage cells. Although the benefits of these systems are extremely helpful, most commercial packs use passive methods because they are cheaper and easier to implement.

The intent of this work is to develop a compact active battery management system using advanced power electronics techniques. Active balancing systems have been demonstrated to charge battery cells in [3, 4, 5] but the power converter design has not been optimized. The goal of this work was to develop a highly efficient converter capable of battery cell balancing requiring that it is both isolated and bi-directional. A forward converter is used for its power handling capabilities and its compatibility with planar magnetics. Techniques used to optimize this design are high frequency (1 MHz), GaN switches, and planar magnetics.

CHAPTER 2

BACKGROUND OF INTERLEAVED DICKSON CONVERTER

2.1 Switched-Capacitor Converters

2.1.1 Capacitors Versus Inductors

The primary purpose of a switch mode power converter (SMPC) is to convert an input voltage to a different voltage. This is necessary because electronic systems require various voltages for different applications. SMPCs operate by storing the input power into an energy storage mechanism, and then sending the stored energy to the output at a different voltage. There are at least two switches that control the flow of energy into and out of the storage elements, and the duty ratio of these switches controls the input to output conversion ratio. The two common forms of energy storage are capacitors and inductors, and each has its benefits. Figures 2.1 and 2.2 show a basic switched-inductor and switched-capacitor circuit [6, 7, 8].

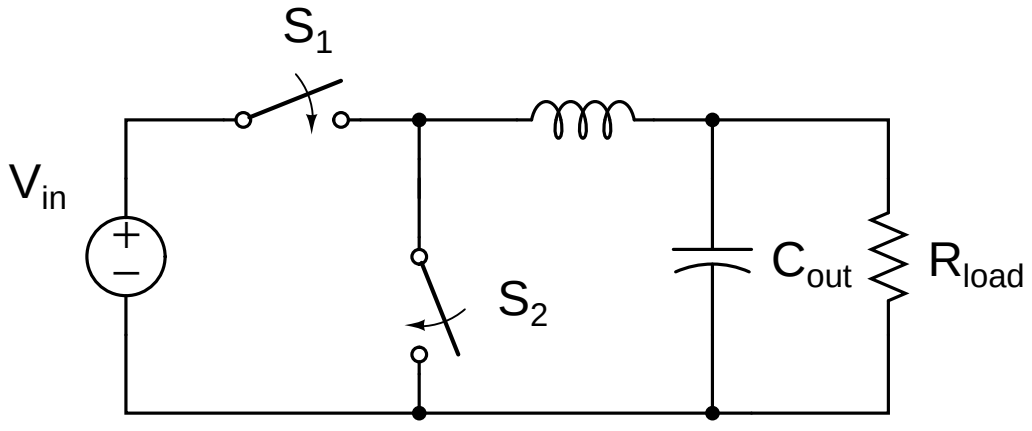

Figure 2.1: Basic switched-inductor circuit.

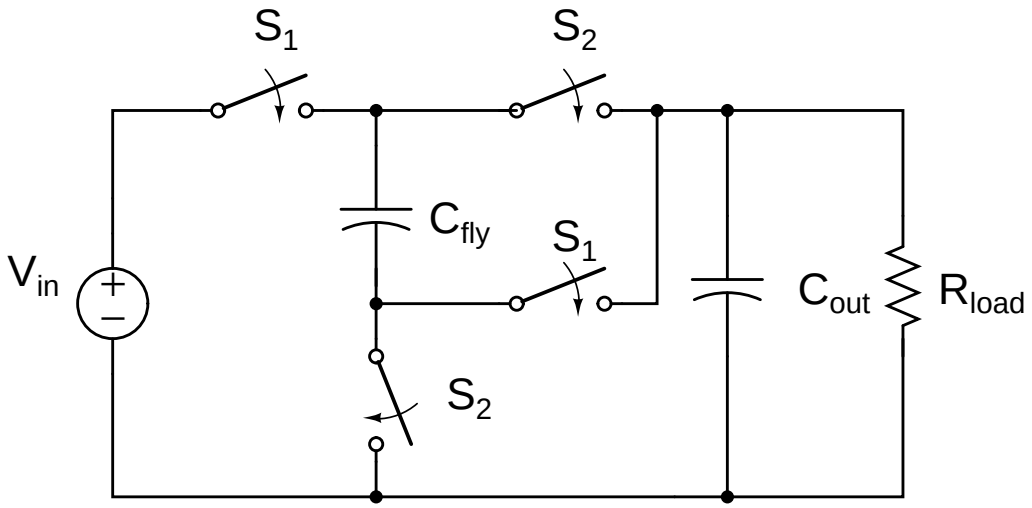

Figure 2.2: Basic switched-capacitor circuit.

Magnetics based converters have low part counts, are easy to implement, and allow for easy control of the conversion ratio. Common examples of these converters are the buck and boost converter, which are widely used in commercial systems. The negatives to using these are that they have high voltage stress on the switches, and magnetics have low energy density relative to capacitors. This increases the size of the converter and requires switches that generate higher conversion losses. In contrast, switched-capacitors can be configured to have low switch and capacitor stress, and capacitors have much higher energy density. The energy density comparison of common commercial inductors and capacitors is shown in Figure 2.3 and is based from data in [9, 10, 11]. This shows that capacitors are more than an order of magnitude more energy-dense, which is the fundamental motivation for optimizing capacitor based converters [12, 13]. The trade-offs to using switched-capacitor converters are that they have high component counts, have fixed voltage conversion ratios, and are not as easy to implement.

2.1.2 Types of Switched-Capacitor Converters

There are various topologies of switched-capacitors that can be implemented such as the series-parallel, ladder, fibonacci, and the doubler. Each topology has a trade-off between the number of switches, voltage stress on switches, number of capacitors, and maximum voltage stress on the capacitors. The

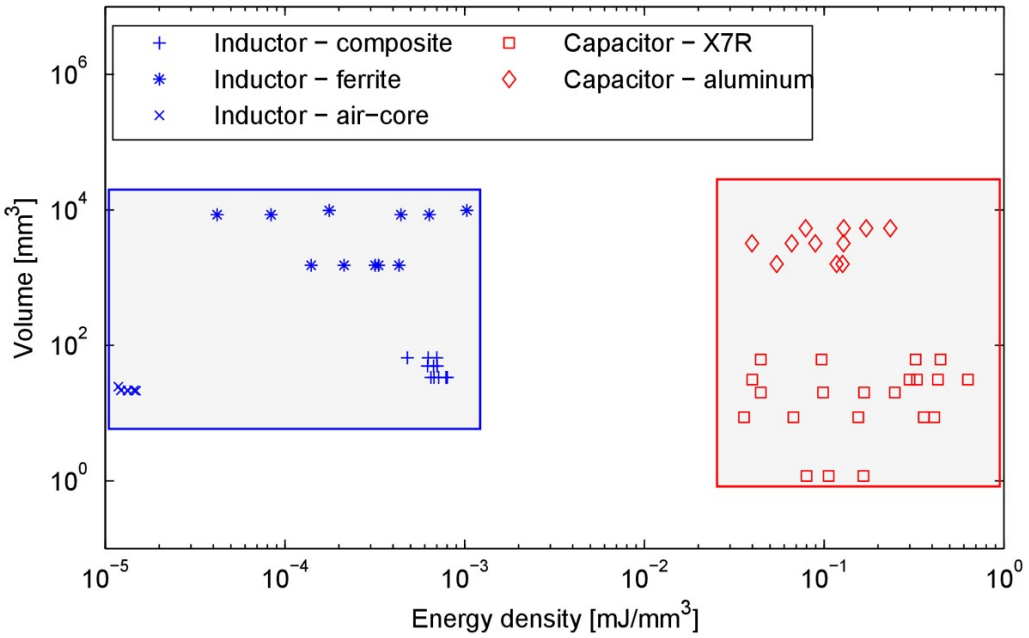

Figure 2.3: Energy density comparison of capacitors versus inductors [9].

summary of the performance of each of these converters can be seen in Table 2.1.

Table 2.1: Comparison of different switched-capacitor converters.

Topology	Switch Count	Switch Voltage	Flying Caps	Cap Voltage
Dickson	$n+4$	$2V_{in}$	$n-1$	$(n-1)V_{in}$
Series-Parallel	$2n+2$	$(n-1)V_{in}$	$n-1$	V_{in}
Ladder	$n+3$	V_{in}	n	V_{in}
Doubler	$4log_2(n)$	$\frac{n}{2}V_{in}$	$2log_2(n)-1$	$\frac{n}{2}V_{in}$
Fibonacci	$3(k-2)+1$	$F(k-1)V_{in}$	$k-2$	$F(k-1)V_{in}$

The rest of this work will focus on the Dickson topology. It was selected for the low voltage stress on switches and its ability to run in resonant operation. Low voltage switches have lower on-resistance and lower switching losses. The benefits of resonant operation will be covered in the resonance section. A negative to this topology is the high voltage stress on the capacitors, but this was viewed as an acceptable trade-off since capacitors have high energy density. Another downside for this converter was the high number of switches, but this does not drastically affect the total volume since small, low voltage switches are being used.

2.1.3 Capacitor Utilization and Efficiency

There is an inherent inefficiency with using capacitors depending on how they are charged [14]. If an uncharged capacitor is tied directly to a voltage source as shown in Figure 2.4, the capacitor will charge to that input voltage. Regardless of the ESR of the capacitor or any other parasitic resistance, half of the energy from the source will be dissipated as heat. This occurs from the I^2R losses by the initial current spike when the capacitor and source connect. Even if the ESR is higher, the peak current spike will be lower but the charging rate will be slower. In contrast, if the capacitor is charged by a current source, it will theoretically lose no energy. In practical conditions, a small loss would result from the parasitic resistances in the design. Therefore, it is desirable to charge capacitors in a switched capacitor converter through a current source, which can effectively be achieved using a small inductor. This will be covered in more detail in the resonance section.

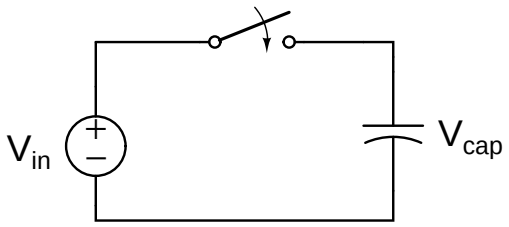

Figure 2.4: Voltage source charging a capacitor.

There is a trade-off between capacitor utilization and efficiency that is useful to understand in switched-capacitor converters. Consider the same situation as Figure 2.4, except the capacitor is already charged up to a voltage less than the input voltage. When the input switch closes, the capacitor will still charge up to the input voltage but the ratio of energy lost from the input will be less than half. Less energy is lost when the change in voltage in the capacitor is reduced because the lower difference in input to capacitor voltage leads to a lower initial current spike.

Two main conclusions can be drawn from the concepts above. The first is that the capacitors need to be charged using a current source which can be achieved by using a small inductor [15, 16]. This inductor does not contribute significantly to the volume of the converter since its only task is to resonate with the capacitors, not store energy. The second is that minimizing the change in voltage seen by the capacitors increases the efficiency. This can be

achieved by switching at a higher frequency or optimizing control.

2.1.4 Switched-Capacitor Converter Modeling

A switched-capacitor converter model that is widely used is shown in Figure 2.5 and comes from [17]. It is a 1:M transformer which is the gain of the converter with a series resistance. The series resistance is frequency dependent at slow switching frequencies due to the effect of capacitor utilization, which is known as the slow switching limit. When high switching frequencies are used, the capacitor utilization effect becomes negligible and the resistance is limited by the parasitic resistances in the components. This is called the fast switching limit.

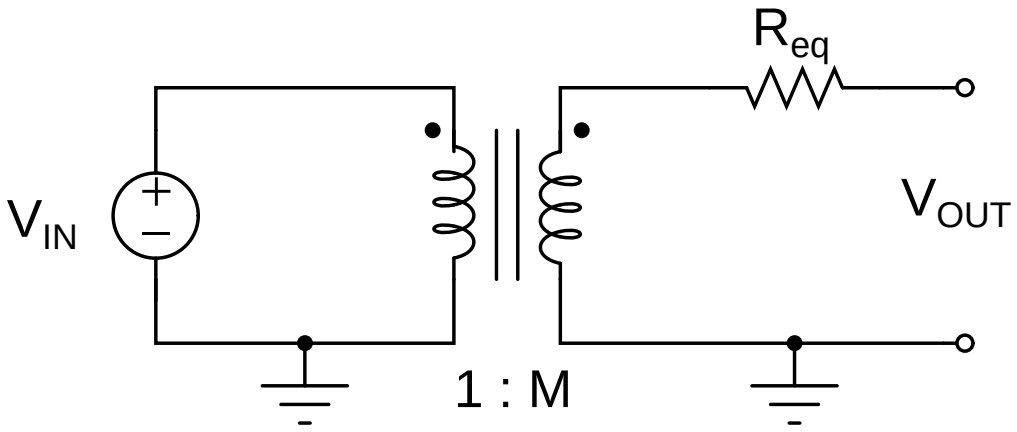

Figure 2.5: Model of an ideal SC converter.

2.1.5 Resonance

Resonance ensures that the capacitors are charged without impulses, which maximizes the efficiency. A basic SC converter without resonance can be seen in Figure 2.6 and a SC converter with resonance can be seen in Figure 2.7. The inductor allows the circuit to be used at lower frequencies with the same equivalent resistance, which reduces switching losses.

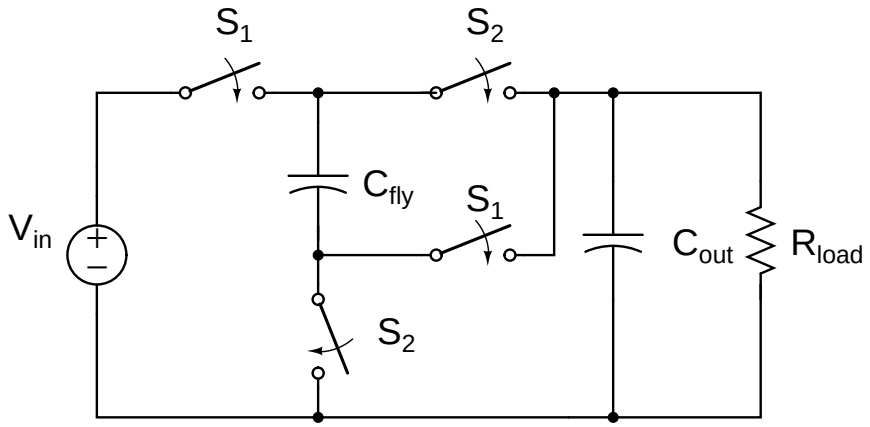

Figure 2.6: Switched-capacitor converter without resonance.

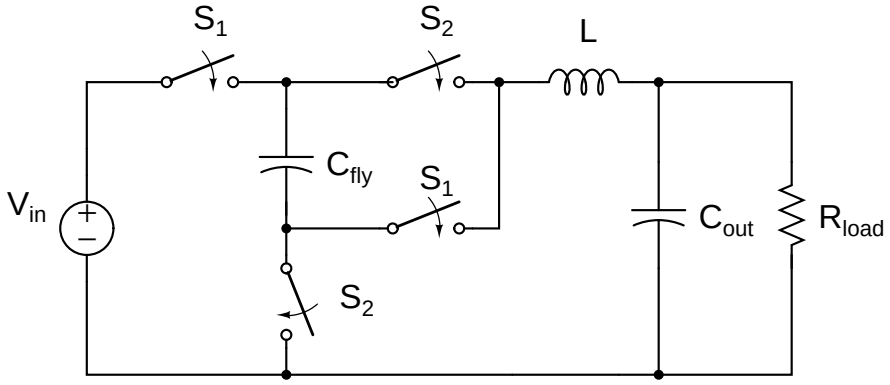

Figure 2.7: Switched-capacitor converter with resonance.

2.2 Resonant Dickson Converter

2.2.1 Dickson Converter

A 1:6 Dickson converter can be seen in Figure 2.8. It has two modes of operation which can be seen in Figures 2.9 and 2.10. It switches between these modes with a 0.5 duty cycle. The voltage stress on each capacitor increases by nV_{in} where n is the capacitor number. The voltage stress on $S_1 - S_5$ and S_{10} is V_{in} and $S_6 - S_9$ is $2V_{in}$.

Adding the inductor to the input of the Dickson allows for resonance [18, 19, 20, 21]. This topology can also achieve lossless regulation as shown in [22]. The equivalent resistance of the Dickson with an inductor can be seen in Figure 2.11. A major benefit to this resonant operation is that it allows for zero current switching (ZCS) at the input, which reduces switching losses. This simulated input current is shown in Figure 2.12.

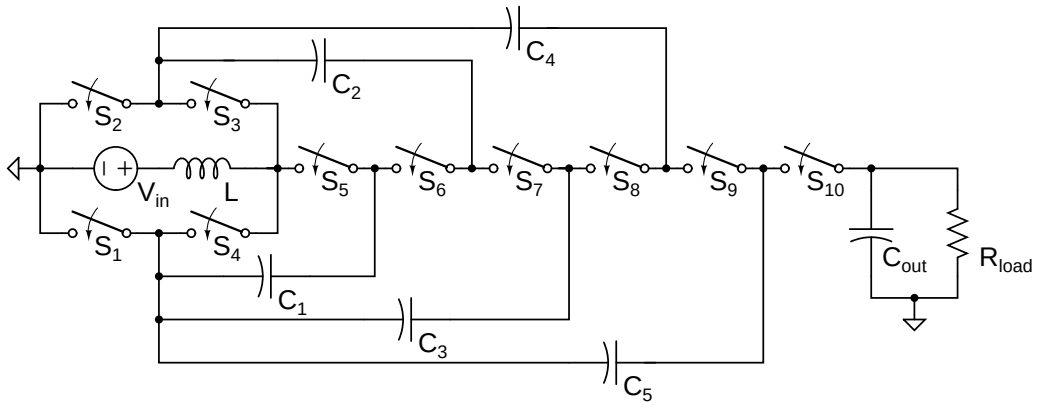

Figure 2.8: 1:6 Dickson converter.

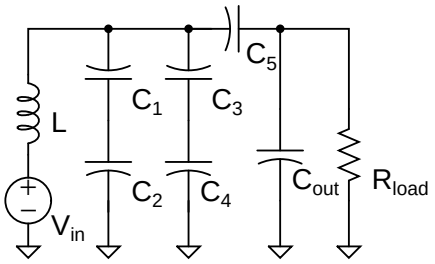

Figure 2.9: First phase of the Dickson.

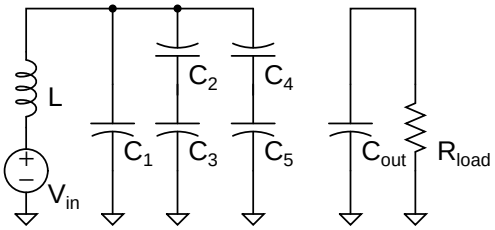

Figure 2.10: Second phase of the Dickson.

2.2.2 Split-Phase Control

While resonance removes the sharp current spikes from the input, it does not remove the internal voltage mismatches. The bottom of Figure 2.12 shows the current spike through S_5 that occurs from an internal capacitor voltage mismatch. This can be avoided in the Dickson with split-phase control [2], [23]. Split-phase control introduces two more states to the converter which effectively act as waiting periods for the voltages to realign.

Typical operation without split-phase control switches between Figures 2.13 and 2.15. The issue with this is that at the end of phase 1a, $V_{C1} \neq V_{C3} - V_{C2}$ and $V_{C1} \neq V_{C5} - V_{C4}$. This introduces a temporary KVL violation in the beginning of phase 2a which leads to the current spike mentioned before. A similar violation occurs when switching from 2a back to 1a.

With split-phase control, the transition from phase 1a to phase 2a is buffered by phase 1b shown in Figure 2.14. This waits to connect V_{C1} until $V_{C1} = V_{C3} - V_{C2}$ and $V_{C1} = V_{C5} - V_{C4}$. A similar operation is done for transition from 2a to 1a by introducing phase 1b in Figure 2.16, except this

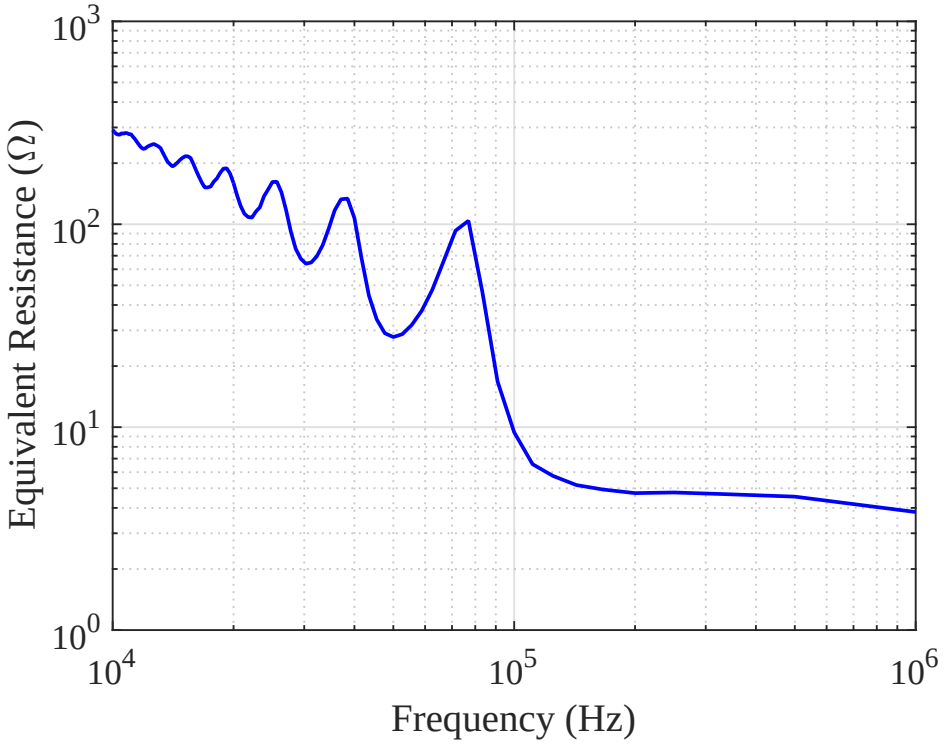

Figure 2.11: Equivalent resistance of Dickson with resonance.

is used to connect V_{C5} at the appropriate time. The result of this operation leads to the current shown in Figure 2.17 versus the current shown in Figure 2.12.

The benefit to introducing these states is an increase in efficiency at the expense of control complexity. Instead of simply having PWM with a 50% duty cycle, a special signal needs to be added to properly connect V_{C1} and V_{C5}. The new switching waveforms can be seen in Figure 2.18 which requires precise timing by the micro-controller or FPGA that is driving the gate signals. The proper time to switch can be calculated by using the work in [2] and does not require closed loop control to switch very close to the nominal time. An interesting side note is that if $S_5 - S_{10}$ are diodes, it inherently operates with split-phase control.

2.2.3 Interleaving

Interleaving is the process of using multiple converter modules simultaneously to increase the power capabilities of a system. Another benefit of interleaving

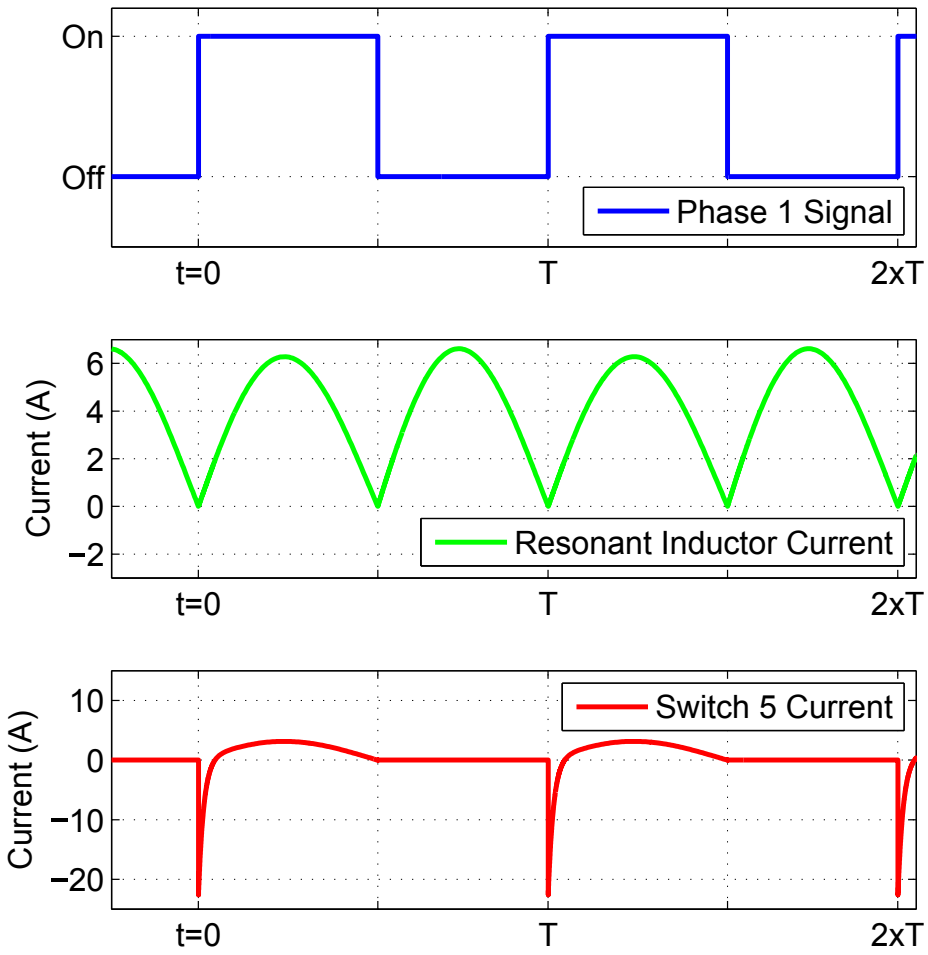

Figure 2.12: Current waveforms in a two-phase Dickson converter.

is that they can be operated out of phase to reduce or completely remove the current ripple. The optimal phase difference depends on the number of parallel converters. Figure 2.19 shows an example of how two interleaved converters would be connected. A concern with interleaving is the current sharing between each phase. If they do not share current efficiently, each interleaved module will need to be overrated which defeats the purpose of interleaving. One of the goals of this work was to analyze and experimentally verify the current sharing mechanism in this converter.

The power loss of the system is minimized when the voltage of each phase is equal for interleaved buck converters [24]. Figure 2.20 shows the DC model of

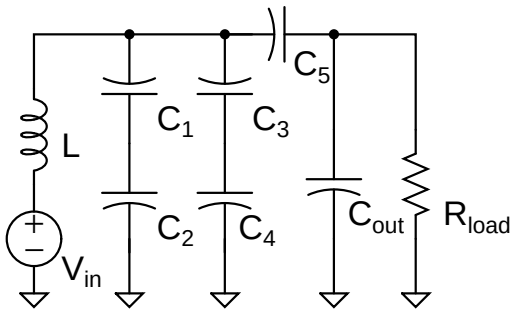

Figure 2.13: Phase 1a.

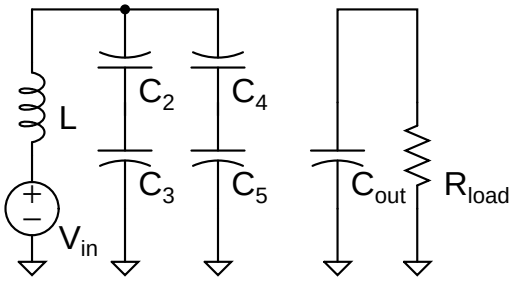

Figure 2.14: Phase 2b.

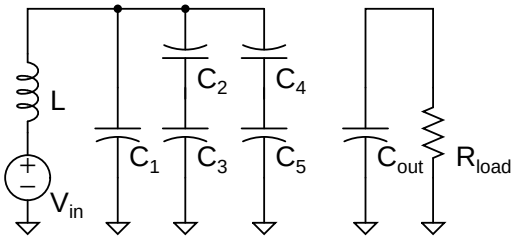

Figure 2.15: Phase 2a.

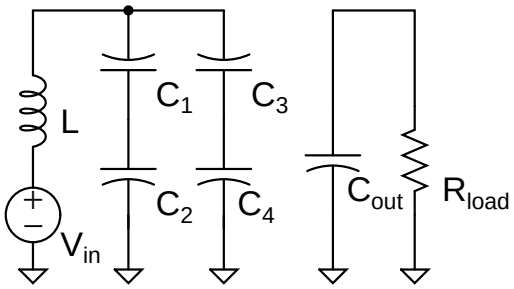

Figure 2.16: Phase 1b.

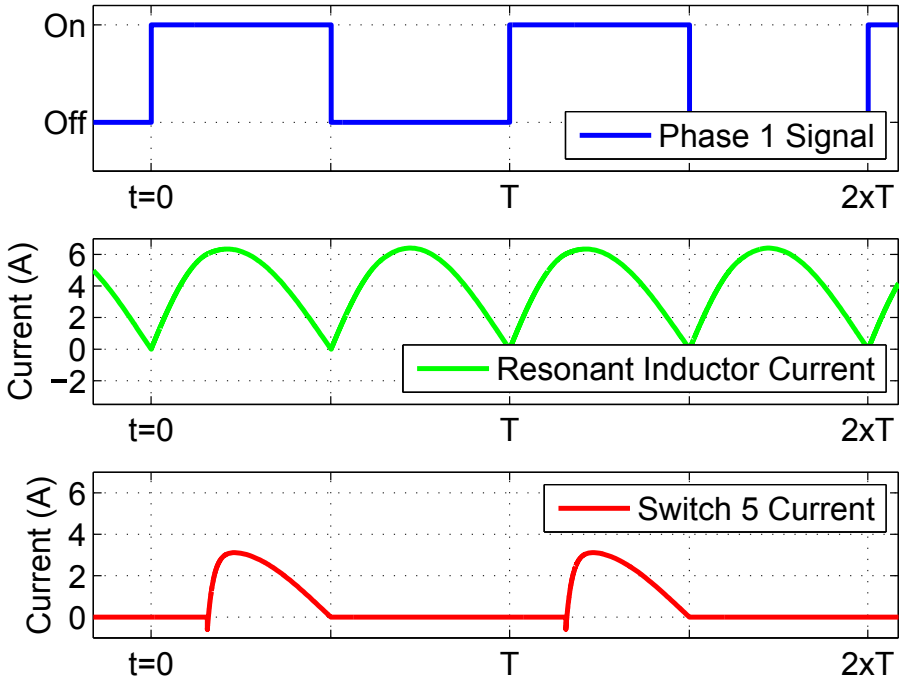

Figure 2.17: Current waveforms in a split-phase Dickson converter.

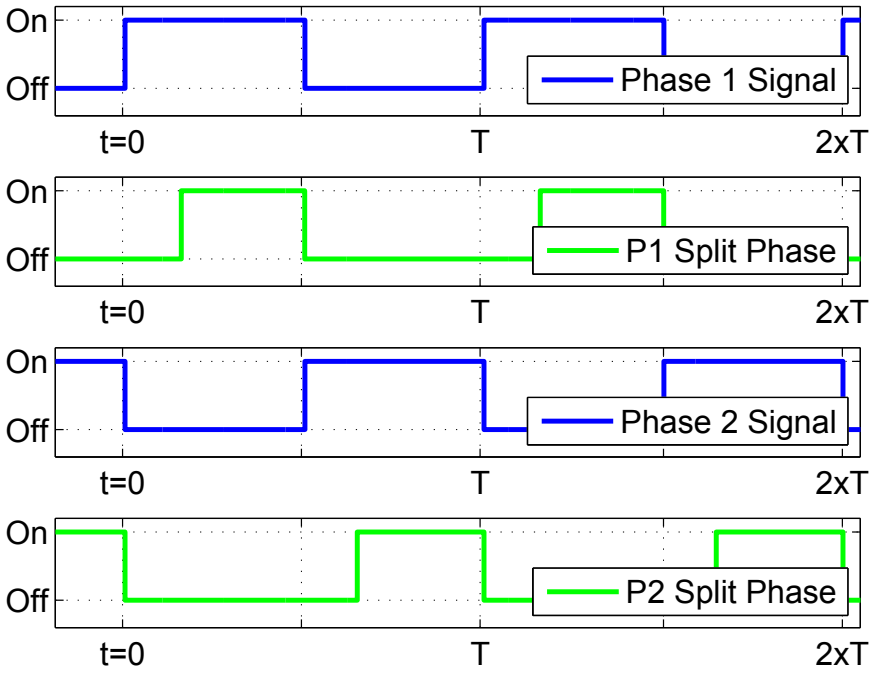

Figure 2.18: Control signals in a split-phase Dickson converter.

k interleaved buck converters. All voltages are equal when the duty cycles of each phase are the same, causing variation in component resistances to cause the mismatch currents. For the majority of SC converters, the conversion ratio is determined by the topology rather than a duty cycle. The standard SC model can be used in conjunction with the model in [24] to analyze the current sharing mechanism of an interleaved Dickson split-phase converter.

The model for current sharing of the Dickson can be seen in Figure 2.21. V_k is equal for each phase due to the inherent topology of the converter and the variation in R_{eq} can be seen in Figure 2.11. In order to minimize the variations in R_{eq}, it is desirable to operate in the region slightly above resonant frequency. Operating at this frequency makes the equivalent resistance a function of the series resistance of the inductor, capacitors and switches. Therefore, these converters should share current well as long as the correct operating frequency is used.

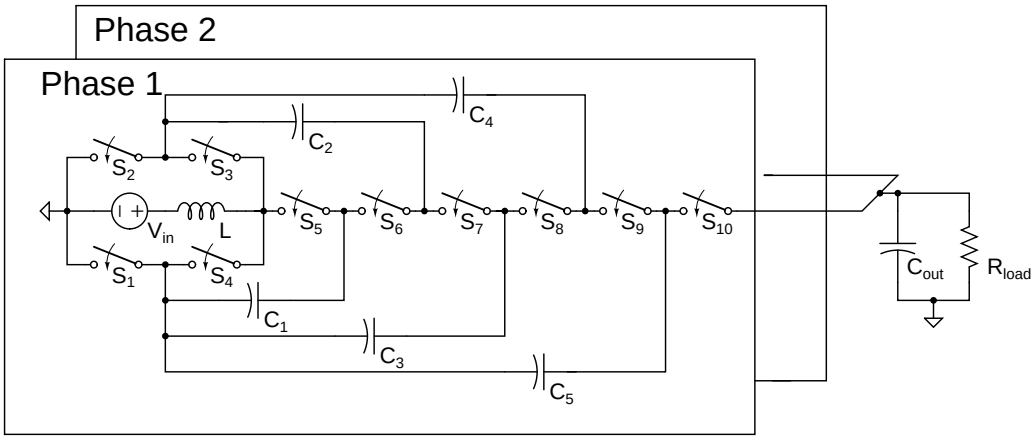

Figure 2.19: Interleaving two resonant Dickson converters.

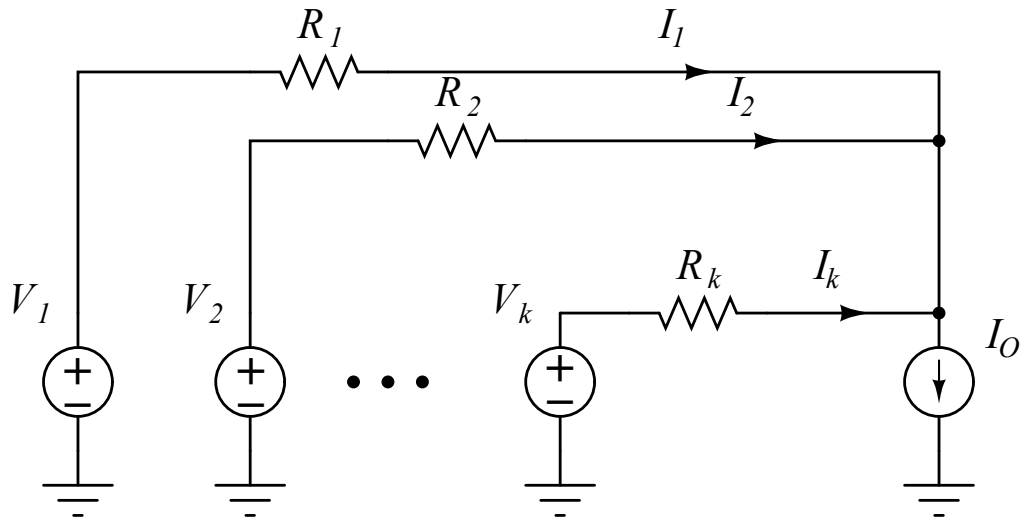

Figure 2.20: DC equivalent circuit of interleaved buck converters.

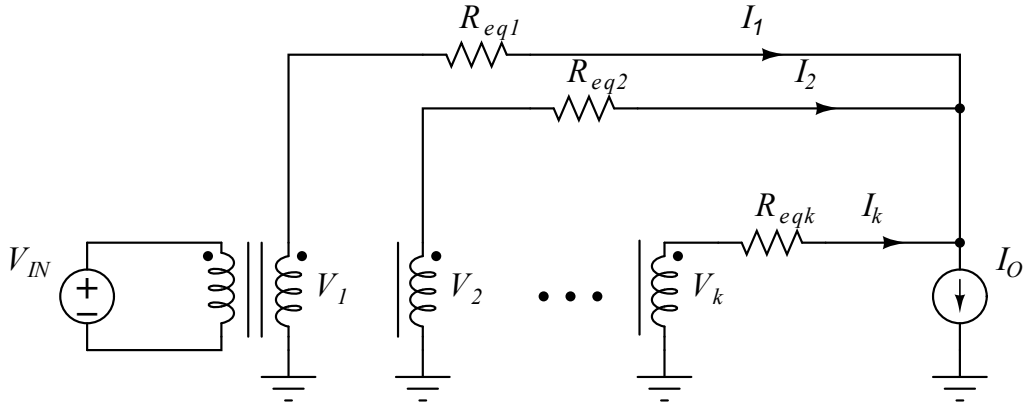

Figure 2.21: DC equivalent circuit of interleaved switched-capacitor converters.

CHAPTER 3

EXPERIMENTAL RESULTS OF INTERLEAVED DICKSON CONVERTER

3.1 Interleaved Dickson Converter

3.1.1 Hardware

The assembled PCB of the interleaved converter can be seen in Figure 3.1. There are three connectors for input power, output power, and 5 V supply power and there are pin headers to connect to the micro-controller. The voltage conversion happens in the power stage which consists of the switches, capacitors, inductor, and gate drivers. The specific components used can be seen in Table 3.1 and an annotated image of a single power stage can be seen in Figure 3.2. This design uses gallium nitride (GaN) switches because they have lower switching losses for a given on resistance. The level shifting circuitry is to power and drive the switches and gate drivers that are not referenced to ground. A detailed schematic and Gerber files from the PCB can be found in Appendix A.

The general layout philosophy of this board was to minimize parasitic inductances. The two main areas to eliminate these are the gate driving paths and the power paths. An example of a gate driving path into a switch is shown in Figure 3.4. The goal is to minimize the area of the loop that I_{gate} flows through because $L_{parasitic}$ is proportional to the loop area. $L_{parasitic}$ depends on the thickness of the trace as well. This total inductance is then reduced by keeping the gate drivers very close to the actual switches and by keeping the traces thick. Minimizing this is important for both reducing the ringing on the gate and allowing the switch to turn on and off faster.

Parasitic inductances on the power path are especially important because they experience a high change in current. Since $V = L\frac{di}{dt}$, even small parasitic inductances can lead to large voltage ringing. The biggest loop in a Dickson

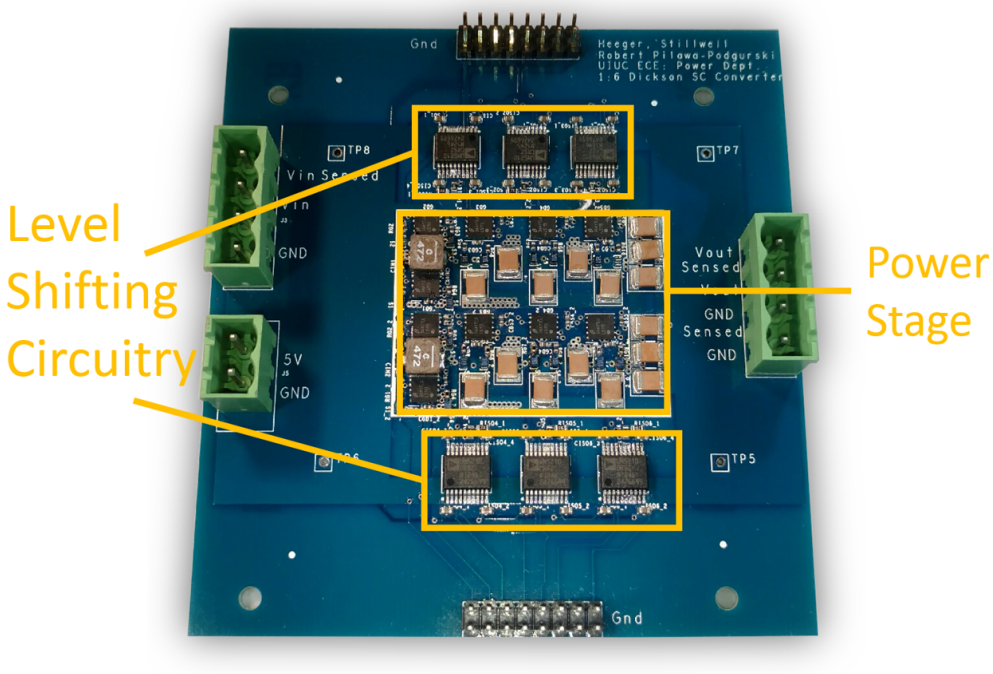

Figure 3.1: The 1:6 interleaved Dickson.

is typically the path through the input, C_4, and C_5. If this area is reduced, it inherently reduces the area for the rest of the current paths. Therefore, a lot of effort was emphasized on reducing this area, which can be seen in the Gerber files in Appendix B.

Most of the switch's sources are not connected directly to ground, so an important hardware consideration is how to implement high side gate driving. The circuit uses a TI LM5113 gate driver that is capable of driving pairs of switches by using a bootstrap capacitor. This method is only sufficient to drive two switches from a single IC. Therefore, $S_1 - S_4$ can be driven directly from ground but $S_5 - S_{10}$ need an isolated 5 V and gate driving signals referenced to the source of S_5, S_7, and S_9. This circuit uses an Analog Devices ADUM5210 which provides both the isolated signals and the isolated power. This can be seen in Figure 3.3.

3.1.2 Control Implementation

Two common ways of controlling power electronics are micro-controllers and FPGAs. The C2000 was selected for this design because it has many useful

Table 3.1: Component listing of the 1:6 interleaved Dickson converter.

Component	Part Number	Parameters
S_1 - S_5, S_{10}	EPC2014	40 V, 10 A
S_6 - S_9	EPC2007	100 V, 6 A
C_{in}	UMK107AB7105	50 V, 1.0 μF
C_1	C1812C104K5RACTU	50 V, 0.1 μF
C_2	C1812C104K1RACTU	100 V, 0.1 μF
C_4	C1812C104K2RACTU	300 V, 0.1 μF
C_5	C1812C104K2RACTU	400 V, 0.1 μF
C_{out}	C1812C104JARACTU	500 V, 0.1 μF
Gate drivers	LM5113	GaN driver
Gate driver caps	C0402C105	1.0 μF
L	XAL5030-472MEB	6.7 A, 4.7 μH
Micro-controller	TMS320F28069	

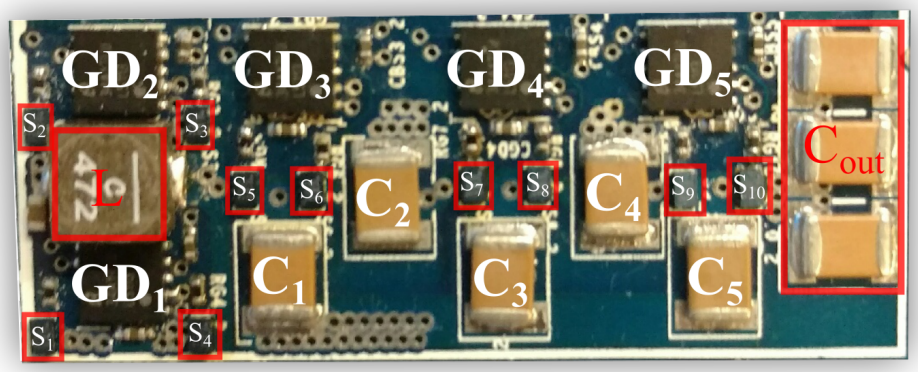

Figure 3.2: Power path of a single module of the 1:6 interleaved Dickson.

features built in for precise control of power converters. The basic methodology for using these micro-controllers is to set a period and a duty cycle referenced to the clock frequency [25]. For example, if this system clock is running at 80 MHz, setting the frequency to 500 kHz is achieved by making the period 160. To control the duty ratio of the converter, a comparison value can be set to trigger an event on the output. Therefore, if a 25% duty cycle is desired, setting the comparator to 40 would accomplish this for a 500 kHz clock. Another important consideration for generating this PWM is using the action control registers which define how the PWM behaves at each event. An event is the start of every period and when the comparator

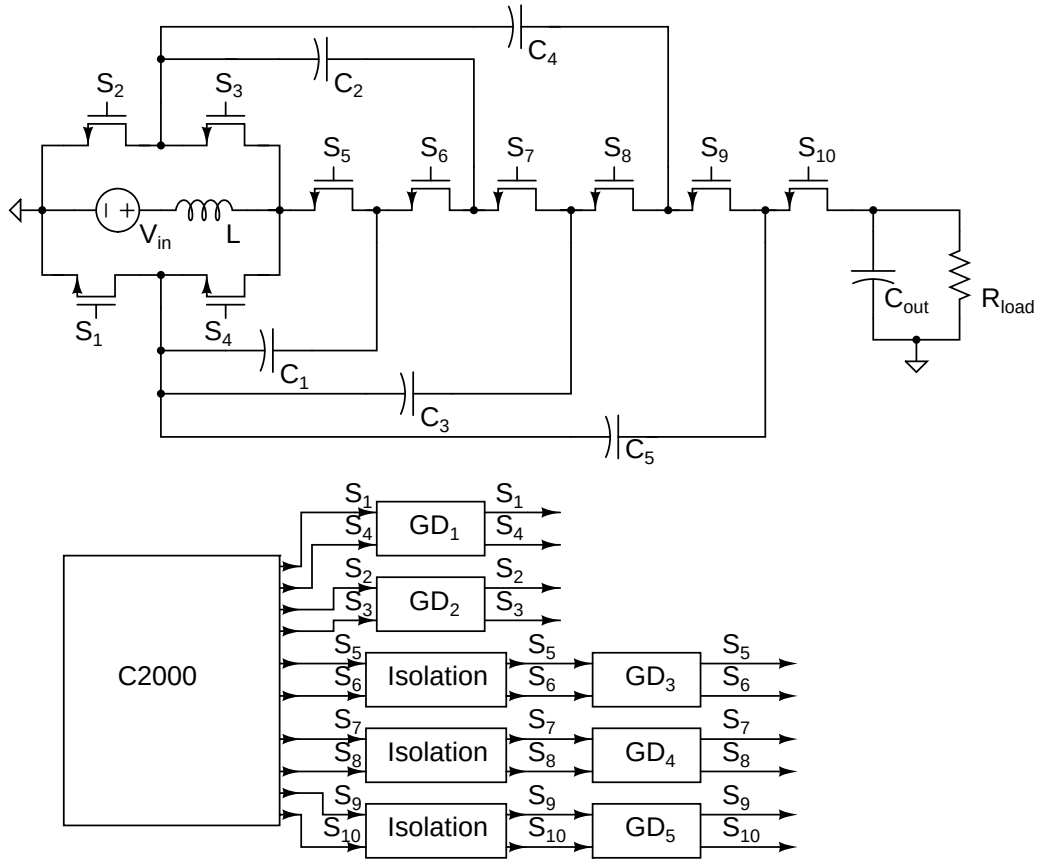

Figure 3.3: Detailed schematic of a single module of the interleaved Dickson.

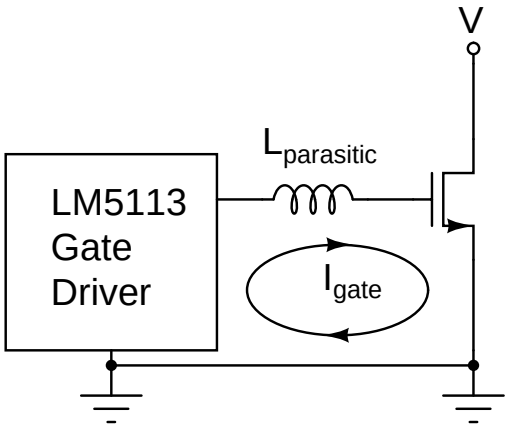

Figure 3.4: An example of the gate driving path.

for the duty ratio is triggered. These can be configured to go high, low, or toggle depending on the desired control operation. Using these registers, in addition to some basic initialization for each PWM channel, is sufficient to

implement split-phase control in the C2000 and the code can be found in Appendix A.

One of the contributions of this work was to implement split-phase control in an interleaved converter instead of just a single converter. Each Dickson converter requires six separate PWM lines, which means there are twelve total PWM channels for two interleaved modules. A single C2000 does not have this many PWM channels so two separate ones need to be used. The challenge associated with this is that each module needs to operate 180° out of phase so they need to be synchronized.

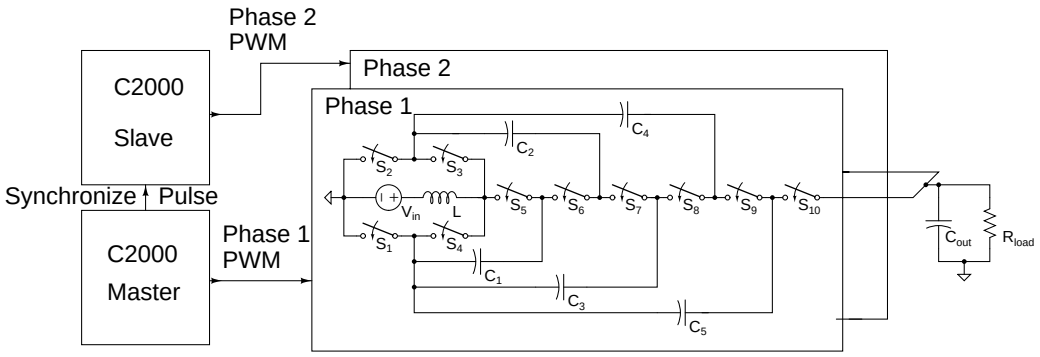

Figure 3.5: Control set up of the interleaved Dickson.

This was accomplished by using the master and slave features in the C2000 PWM peripherals as shown in Figure 3.5. The master generates a pulse at the start of every period which is sent over to the slave micro-controller. The slave signal receives this pulse, and then has to delay its operation by a half of a period. Every PWM channel within a specific micro-controller operates off one of these synchronization pulses except the master. Therefore, the first PWM channel of the slave micro-controller will need to pass this delayed signal through the rest of the PWM channels.

3.2 Results

3.2.1 Efficiency

Figure 3.6 shows the efficiency versus output power of the switched-capacitor converter not factoring in the control power (approximately 0.25 W per module). The blue line is the single phase and the orange line is the interleaved power. As would be expected, the interleaved converter is capable of twice the output power as the single phase. The steep decreases in efficiency that happen at 30 W and 60 W are due to high capacitor utilization as shown in Figure 3.7. This can be avoided by increasing the capacitance or by increasing the switching frequency. Another issue with this design is ringing on the lower voltage rated switches. An example of ringing can be seen in Figure 3.8 with 10 V on the input which causes the device to block twice the nominal voltage.

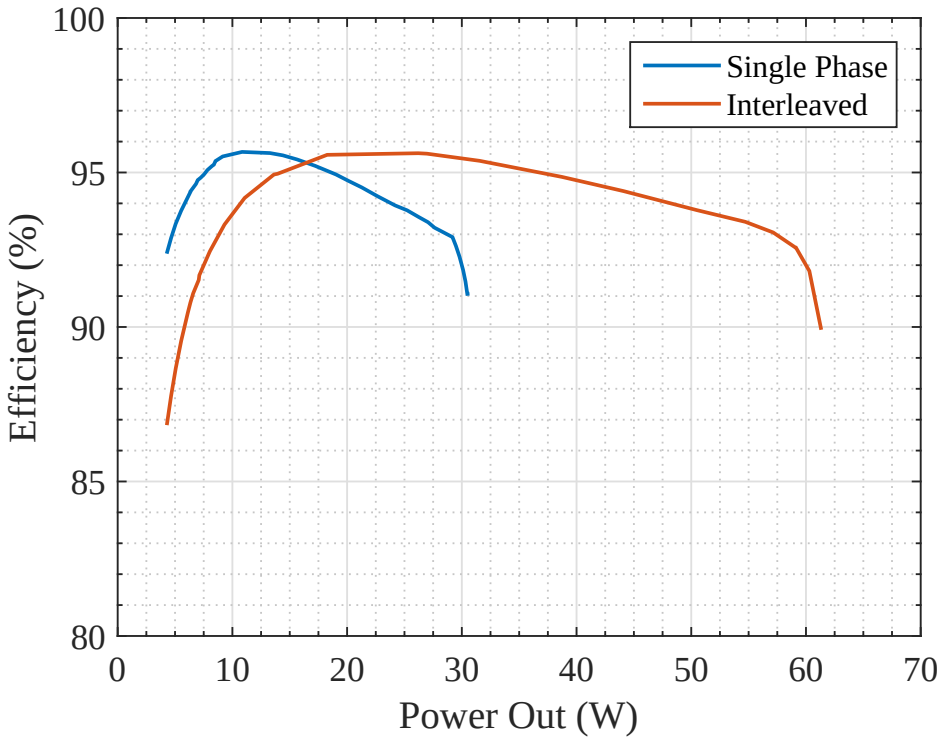

Figure 3.6: Prototype efficiency of Dickson converter with 20 V to 120 V conversion.

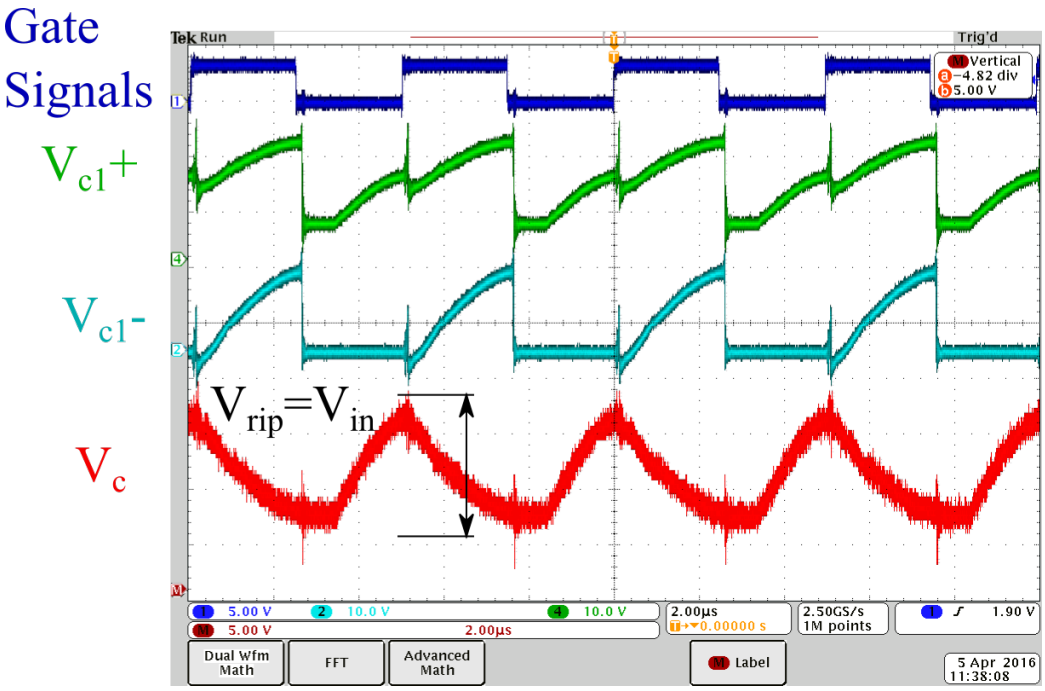

Figure 3.7: The voltage ripple on C_1 with 10 V input and 18 W of output power.

3.2.2 Resonance and Current Sharing

Figure 3.9 shows the resonant currents through the inductors of both phases as well as the switching signals. Note that Figure 3.9 demonstrates excellent current sharing between phases with no special control implementation to balance the current. This asserts that the Dickson is a good candidate for interleaving to build high power converters.

3.2.3 Challenges with the Dickson

The limitations in increasing the output power with the Dickson converter result from two main failure modes. The first is that voltage ringing can cause switches to break and the second is thermal. To optimize the performance of a given Dickson converter, three settings can be tuned: operating frequency, gate resistance, and switch ratings. Increasing the output power is accomplished by switching at a higher frequency, but this contributes to higher switching losses leading to thermal limitations. The main contributor to switching losses is the current and voltage overlap during the transition time of the switch. Therefore, the solution is to decrease the gate resistance

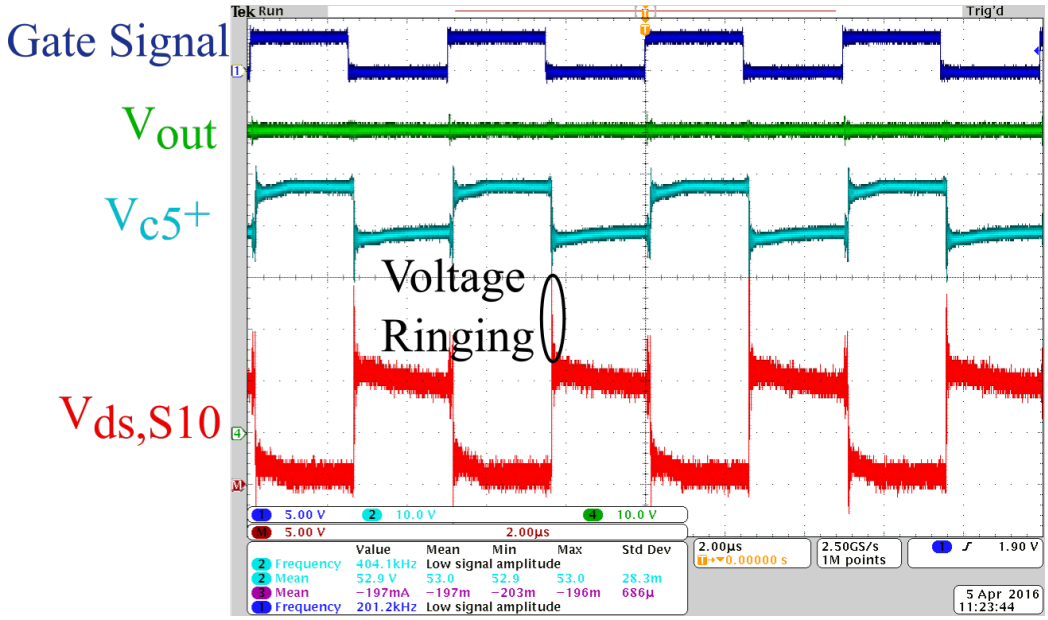

Figure 3.8: This shows the drain to source voltage ringing on S_{10} with a 10 V input.

to switch faster to minimize these losses. The disadvantage to this solution is that a shorter transition time increases the ringing in the circuit because the dt value is reduced in $V = L\frac{di}{dt}$. This limits the input voltage on the converter to something less than the nominal voltage, which necessitates increasing the switch's blocking voltage. Unfortunately, higher voltage rating on switches increases either the switching losses or the conduction losses through the switch (or both at the same time), making it difficult to increase the switching frequency.

These three settings, which are summarized in Table 3.2, have interdependent effects such that no single parameter is the solution. The three main improvements from this design will come from an improved layout, better switches, or better capacitors. An improved layout involves further minimizing the loop areas in the converter. Better switches would have a lower on-resistance for higher blocking voltages and lower gate capacitances. This would allow the converter to be switched faster with switches that could tolerate higher ringing. Better capacitors would have a higher energy density which would allow for a slower switching frequency for a given board area or allow for smaller capacitors that could be used to reduce the converter size and loop area.

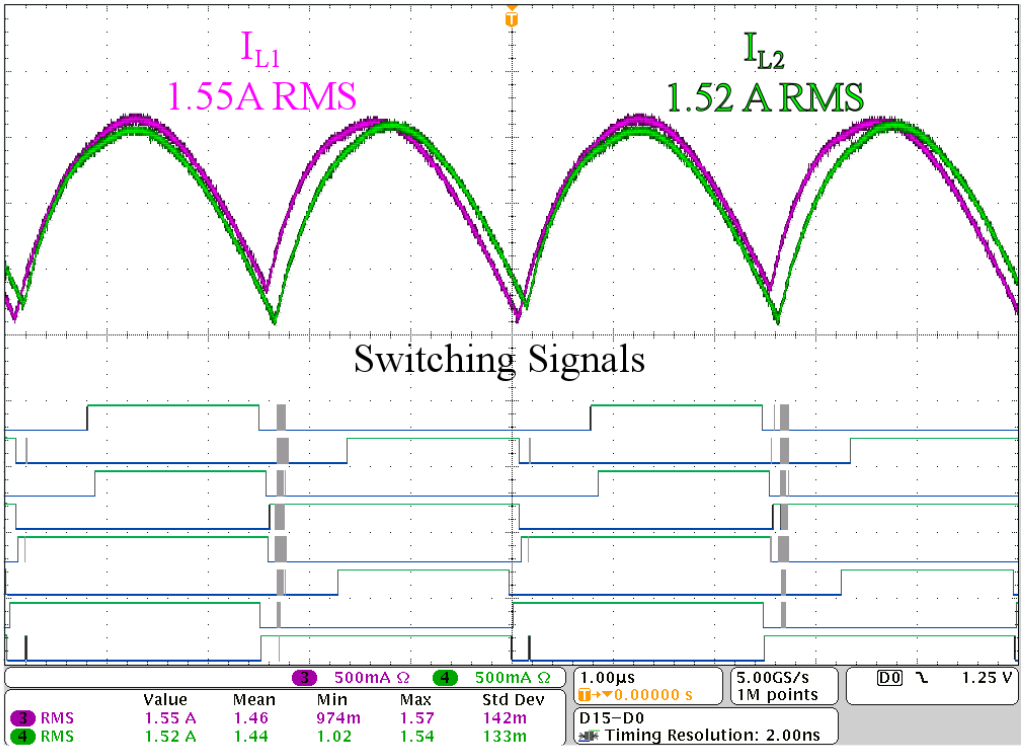

Figure 3.9: Waveform showing the resonant currents of the inductors and control signals.

Table 3.2: Circle of design consideration limitations.

Feature	Positive Effect	Negative Effect
$\Uparrow f$	Increases power density	Increases switching losses
$\Downarrow$ Gate Resistance	Decreases switching losses	Increases voltage ringing
$\Uparrow$ Switch Blocking Voltage	Tolerates more ringing	Increases switch losses

CHAPTER 4

ACTIVE BATTERY MANAGEMENT SYSTEMS

4.1 Features in Battery Management Systems

Battery management systems (BMS) are necessary for any lithium-ion battery pack. The functional requirements of the BMS depend on the size of the battery pack and the application that the battery is being used in.

4.1.1 Overvoltage and Undervoltage Protection

Since the primary purpose of the BMS is to ensure safe operation of the battery pack, it must be able to disconnect the charger when a cell reaches its maximum voltage. Undervoltage (UV) on battery cells can be just as damaging as overvoltage (OV), so it must also be able to disconnect the load when this condition is reached. This protection requires a voltage measurement and some type of switch to disconnect the charger. UV and OV conditions happen very slowly so they do not need to be checked for at a high frequency. The apparent voltage measurements will change depending on current through the battery and the cell's internal impedance, so it is important to factor this into the safety voltage measurements.

The two common switching methods are relays or MOSFETs. Relays are most useful when galvanic isolation is required but are usually bulky, require significant power to operate, and chatter when they connect. A p-FET or n-FET benefits from being small, requiring low power, and not chattering, but does not have the reliability benefits of relays. An important consideration with using FETs is where to place them and how to control them. A p-FET is commonly used on the high side of the battery pack and can be driven by using resistors and a small n-FET referenced to ground. The disadvantage of this is that a p-FET with a low on-resistance is generally very big. An

n-FET, which has lower on-resistance for a given package size, can be used on the high or low side of the battery pack. The low side benefits from being easy to control directly from a micro-controller but creates two separate grounds separated by the FET. This can lead to communication and noise issues especially during periods of high current discharge from the battery. The only negative of the high side n-FET is that it requires high side gate driving circuitry like a charge pump to drive the FET.

4.1.2 Short Circuit Protection

Batteries are capable of supplying significant amounts of current extremely quickly which can damage the cells and their surroundings. Therefore, it is important to recognize short circuit conditions much quicker than OV or UV conditions. The switch used to stop the short circuit will be the same switch to the load that prevents UV conditions. In order to detect this condition rapidly, it is most useful to feed a current sense measurement directly into an external interrupt of the micro-controller. This interrupt would be set with the highest priority and its routine would immediately open the switch. The current spike can be detected by a big sense resistor or by a small sense resistor with an amplifier. It is important to use an amplifier capable of operating at a very high frequency if this option is used.

4.1.3 Cell Voltage Monitoring

Cell voltage monitoring can be difficult because every cell in a pack is at a different voltage so an isolated measurement is required for each cell. Fortunately, there are various integrated circuits for this specific application. Texas Instruments and Linear Technology are two common companies that make these, and the specific chip used depends on the number of cells in the system [26, 27].

Another less desirable method that can be used is resistor dividers across every cell so they can measured with an ADC referenced to ground. This suffers from poor measurement accuracy due to component tolerances in the resistors and the constant passive current draw from each resistor. The benefit of this method is that it is very cost effective.

When measuring cell voltage, it is important to calculate the actual voltage of the cells by adding the drop from the cell's internal impedance. This can be done using equation 4.1.

$$V_{OC} = V_{measured} + I_{discharge} R_{cell} \tag{4.1}$$

4.1.4 State of Charge Estimation

A battery pack that supplies energy to a system with a user interface should be capable of measuring state of charge (SOC), which is generally estimated in two ways. The first is pure coulomb counting which requires a high performance current sensor to measure the total current. The current is integrated over time to keep track of the capacity of the pack. The main drawback to this method is that it is very difficult to create a current sensor that remains accurate over a wide operating temperature. Also, the clock of a microcontroller can vary 3-5% if a standard internal oscillator is used and more expensive external oscillators are required for highly accurate timers.

The second way is by using voltage and mapping this to the SOC characteristics. The issue with this technique is that some chemistries have a very flat profile over their state of charge, making SOC resolution very poor. These curves are also temperature dependent so this needs to be accounted for to minimize SOC error.

4.1.5 Reporting to Host System

Depending on the system the BMS is operating in, it sometimes needs to be capable of reporting information back to the user. The amount and type of data is completely application specific. Common communication protocols that can be used are Ethernet, USB, SPI, and CAN. Each one has various benefits pertaining to data rates, communication wires lengths, data bit error rates, and power consumption.

4.2 Overview of Cell Balancing Mechanisms

The final feature of a BMS, and the focus of this work, is battery cell balancing. This is required when cells have slight mismatches in internal impedances leading to voltage imbalances during charge or discharge. In an ideal pack balancing would not be necessary, and pack manufacturers generally group cells by their impedances to minimize the differences.

Cell balancing allows the pack's capacity to be fully utilized. If a single cell reaches its maximum voltage before the rest of the pack, the rest of the cells cannot be charged without balancing. Similarly, if a single cell reaches the UV threshold before the rest of the pack, the pack cannot fully discharge.

There are two general categories of battery balancing but many different ways to specifically implement them [28]. The first type comprises passive systems which use resistive methods to burn energy off the high voltage battery cells. All the excess energy is dissipated as heat, which is bad for the health of cells. Active systems use power electronics to move the excess energy to a different location in the pack [29]. There are three main sub-methods of this: individual isolated chargers, cell-to-cell balancing, and bus-to-cell balancing. These will be covered in more detail in the following section.

4.2.1 Passive Balancing

A schematic for passive cell balancing can be seen in Figure 4.1. This operates by enabling the charger until a single cell reaches its maximum cell voltage. The next operation depends on the value of the cell balancing resistor used. Low resistance values will balance high currents but will also generate large amounts of heat. The resistor is generally chosen to be the smallest it can be while still being able to remove the heat. If the balancing current is lower than the charge current (which it usually is), the charger needs to be disconnected immediately upon detecting the OV. It will dissipate energy until the cell reduces to a certain voltage and then turn the charger back on. This process repeats until all the cells are fully balanced, causing the end of the charge cycle to be very slow. If the balancing current is higher than or equal to the charge current, the charger does not need to be stopped but the cell balancing resistor needs to be enabled on the high voltage cells.

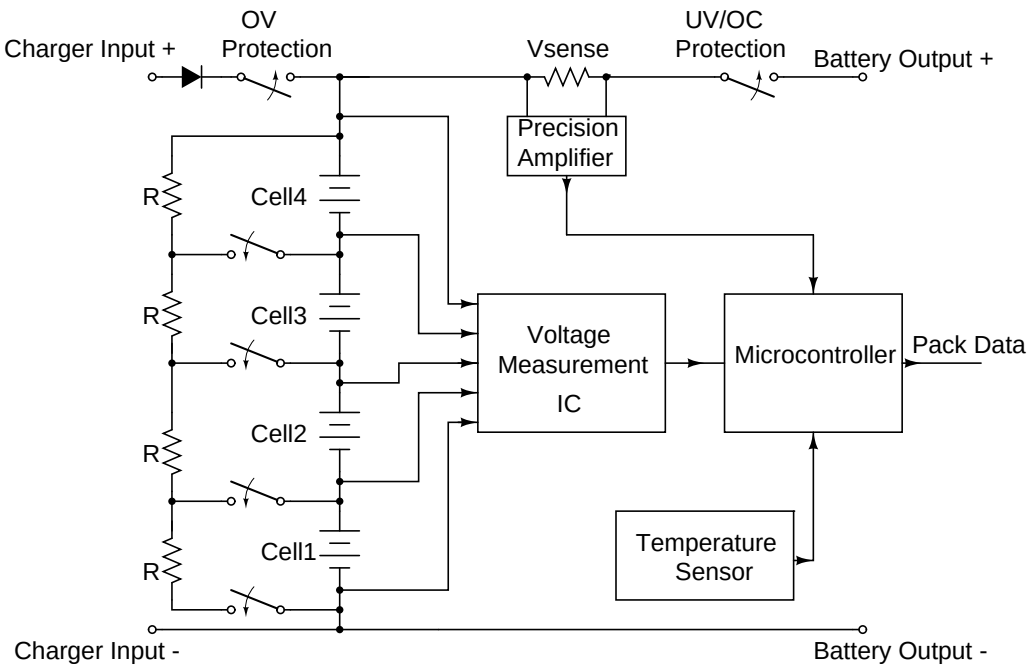

Figure 4.1: A passive battery management system.

4.2.2 Cell-to-Cell

A schematic for the cell-to-cell architecture can be seen in Figure 4.2. This operates by shifting energy from the high voltage cells to the low voltage cells [30, 31, 32]. The amount of energy that gets shifted from cell to cell depends on the specific topology used, but is generally dependent on the cell voltage imbalance, duty cycle, operating frequency, and size of the passive components. The benefit to this method is that it only processes the difference in power among the cells, not the full charge current. This is because the converters only need to operate while there is an imbalance and can remain off when there is not. Another benefit to this method is that the power path of the converter does not need to be isolated, although the communication to the converter and the control power needs to be isolated. This method is better if the cells that have high voltages are close to the cells that have low voltages, which is more likely in smaller packs. Consider the scenario where the top cell has the highest voltage and the bottom cell has the lowest. The energy from the top cell needs to go through all three element-to-element converters, which means the power is being processed three times. If a single converter efficiency of 90% is assumed, about 30% of the energy being processed will be lost as heat. A useful capability of this technique is that it can

shift the energy from cell to cell while the pack is discharging. Therefore, if a cell has a lower voltage while discharging, this cell can be charged from the others which effectively extends the life of the battery.

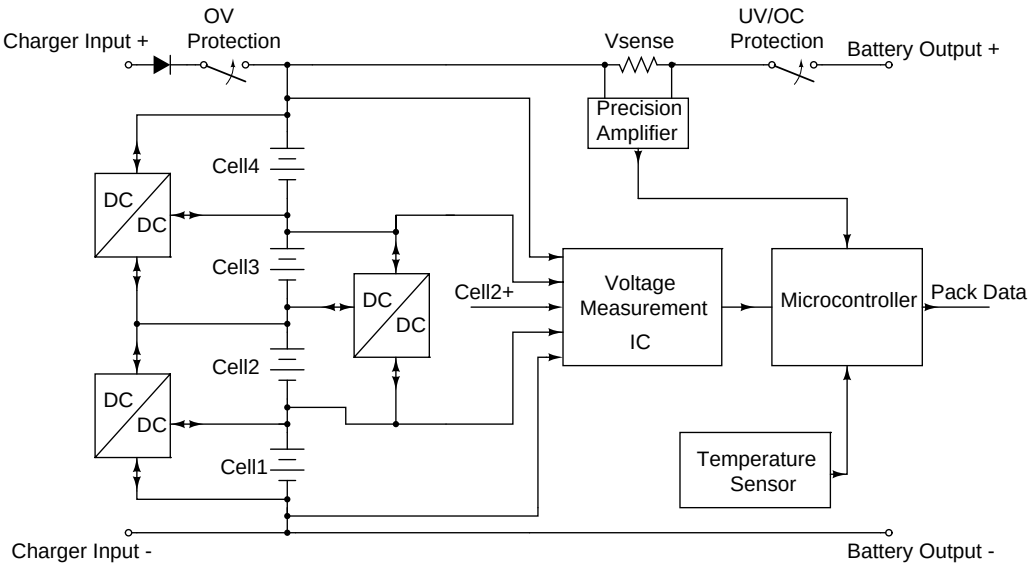

Figure 4.2: A BMS with the cell-to-cell architecture.

4.2.3 Isolated Charger

A schematic for the individual charger architecture can be seen in Figure 4.3. This operates by simply having an isolated charger on every battery cell. The benefits to this method are that the efficiency of the charge is independent of the location of imbalanced cells and controlling the charger is very easy. Every power converter needs to be isolated, but it does not require isolated communication to control it. Having a power converter directly tied to every cell allows enables impedance spectroscopy, which is a technique used to monitor the state of health of the cell by applying sinusoidal inputs at various frequencies [33]. A negative to using this method is that all the energy that comes through the charger is processed. Therefore, if a 90% efficiency converter is used, 10% of the energy will be lost regardless of the mismatch in the cells. This method also does not allow for the cells to balance while discharging.

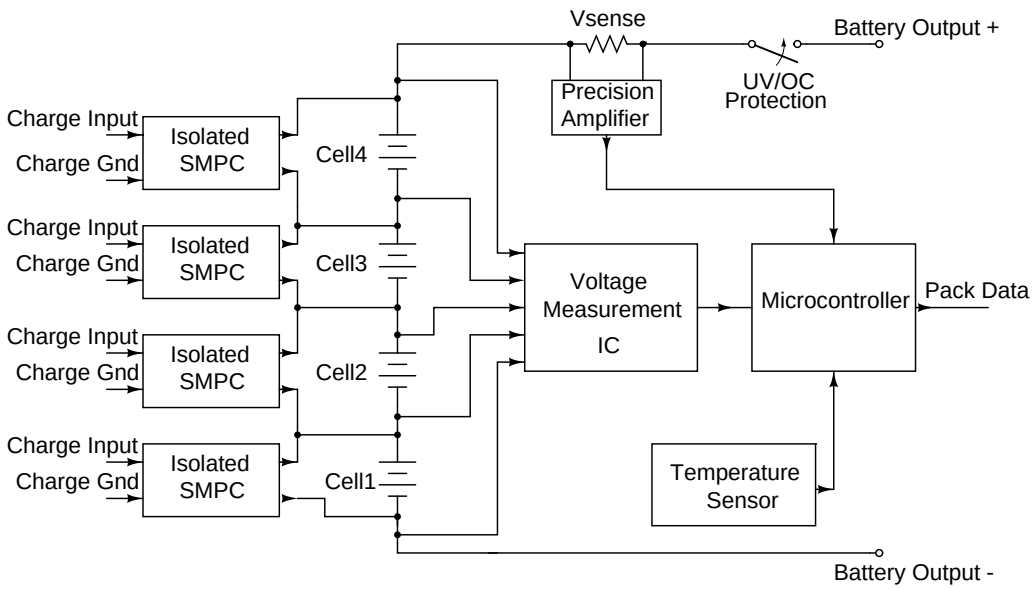

Figure 4.3: A BMS with the individual charger architecture.

4.2.4 Bus-to-Cell

A schematic for the bus-to-cell architecture can be seen in Figure 4.4. This operates by having an isolated bi-directional power converter on every single cell. The main difference between this and the individual charger method is that the input connects to the pack and the charger. The converter only needs to operate when there is a difference in cell voltages so it processes much less power than the individual charger method [34, 35, 36, 37]. The power converter still needs to be isolated, but the communication does not. This method is ill suited to very high voltage packs because that would require a high step-down ratio which is difficult to do efficiently. This design allows for impedance spectroscopy and for the cells to balance while discharging. The control will be more complex than the other methods because every converter can run bi-directionally.

4.3 Isolated Bi-Directional Power Converters

The rest of this work will assume the use of either the individual charger method or the bus-to-cell method. Both require an isolated converter so this section will address the converter in detail. The three main topologies that were evaluated were fly-back, forward, and resonant. The resonant topolo-

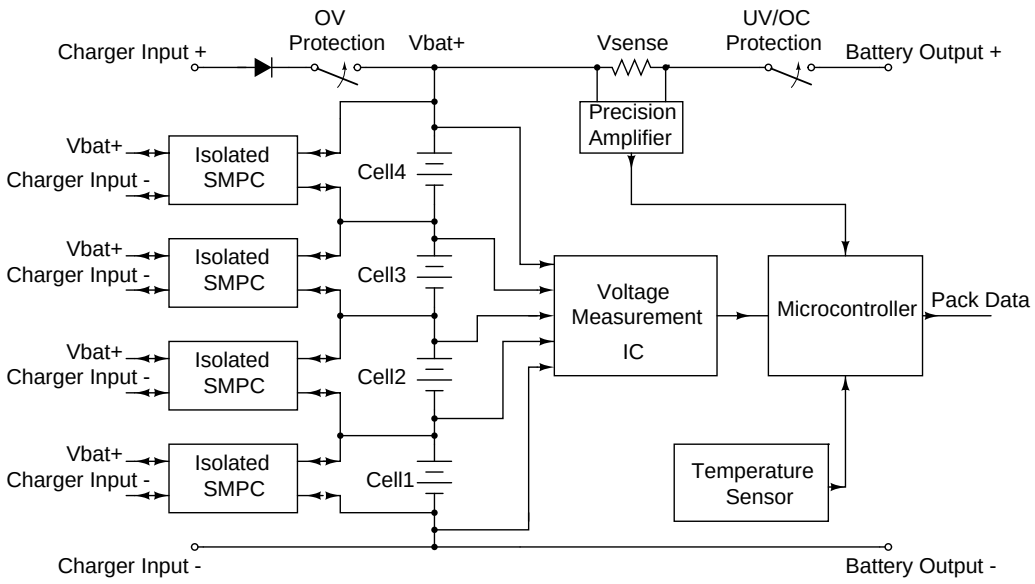

Figure 4.4: A BMS with the bus-to-cell architecture.

gies are capable of high power and high efficiency but were not selected for the high component count. The fly-back only requires two switches and one magnetic element so its low component count and ease of control are desirable. It was not selected because its maximum power capability is limited by the energy that can be stored in the transformer. The forward converter has an additional switch and inductor, but is capable of much more power than the fly-back since the transformer is not an energy storage element. Also, planar magnetics were being used and the benefits of planar magnetics are reduced in a fly-back topology. This will be covered in more detail in the planar magnetics section.

4.3.1 Forward Converter

The forward converter used to charge batteries can be seen in Figure 4.5. The forward converter is basically an isolated buck converter and it has two modes of operation. The first is when both S_1 switches are on, in which case energy flows directly through the transformer into the inductor and load. The second is when S_2 is on but the S_1 switches are off. The duty cycle of these two modes decides the output voltage relative to the input voltage. To be able to implement differential power processing, this converter must be bi-directional. The forward converter operates bi-directionally naturally as long

as all the output switches are n-FETs instead of diodes. The current sensor will also need to be able to measure negative current to properly control the circuit.

The forward converter has both an inductor and transformer. The transformer is not the energy storage element, it simply provides the isolation. The inductor is the main energy storage element so it must be sized appropriately.

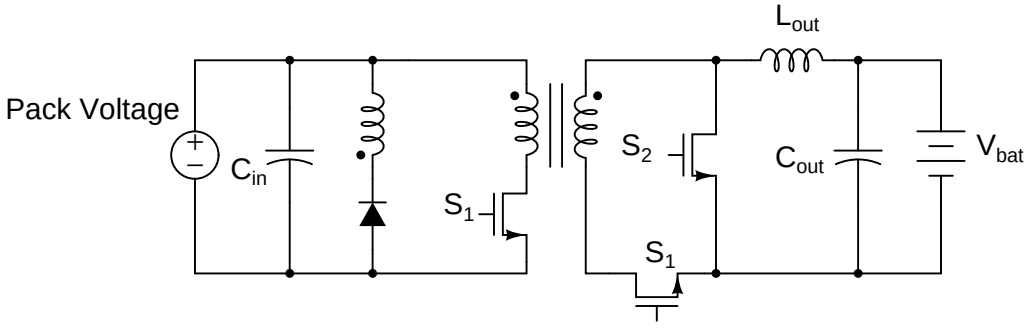

Figure 4.5: The forward converter used to charge batteries.

4.3.2 Magnetizing Inductance

A major design consideration with the forward converter is that the transformer has a magnetizing inductance that is not naturally reset. This is because the volt-second balance on the windings is inherently not zero. It is important to maximize the magnetizing inductance to minimize the energy stored in it. How to maximize this will be covered in more detail in the planar magnetics section. The amount of current built up in the magnetizing inductance can be calculated using equation 4.2 where D_{max} is the maximum duty ratio, V is the input voltage, f is the operating frequency, and L_m is the magnetizing inductance.

$$I_{L,Max} = \frac{D_{max}V}{fL_m} \tag{4.2}$$

This current can then be used to find the total power which is given by equation 4.3. These current and power values are very important for deciding which core reset mechanism to use. The two main parameters that can be controlled in the design process are the operating frequency and the magne-

tizing inductance of the transformer. The input voltage and duty ratio are dependent on the converter specifications.

$$P_{avg} = VI_{avg} = \frac{V}{2} D_{max} I_{L,max} = \frac{V^2}{2} \frac{D_{max}^2}{fL_m} \tag{4.3}$$

There are three common ways to reset the core and each has unique benefits based on the amount of energy being processed. The easiest to implement is a zener diode in series with a diode which can be used to passively burn off the energy to reset the core as shown in Figure 4.6. This is recommended for applications where the amount of energy in the magnetizing inductance is much less than the total power processed. The only critical design implication for this circuit is the value of the zener voltage, which needs to be high enough to fully reset the core. If the voltage is too high, it can increase voltage stress on the switches and capacitors. Choose the minimum V_z according to equation 4.4 to ensure that the core fully resets, and set the maximum V_z value according to equation 4.5 to ensure switches are not overstressed.

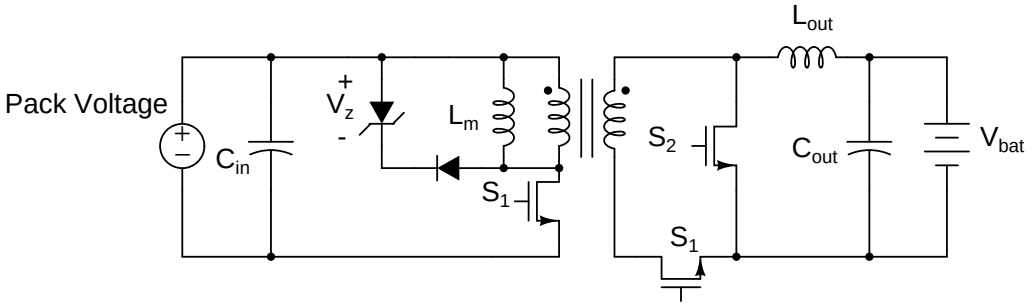

Figure 4.6: Topology for passively removing the magnetizing inductance energy.

$$V_Z > Vin * \frac{D_{max}}{1 - D_{max}} \tag{4.4}$$

$$V_z < V_{sw} - V_{in} \tag{4.5}$$

An active way to reset the core is to use a switch in series with a capacitance to push the energy back into the input. The capacitor value is dependent on the magnetizing inductance and maximum duty cycle at which the converter is operated. This topology can be seen in Figure 4.7. This method is not advantageous at high frequencies because the switching losses can contribute

as much loss as the amount of energy that one is attempting to save.

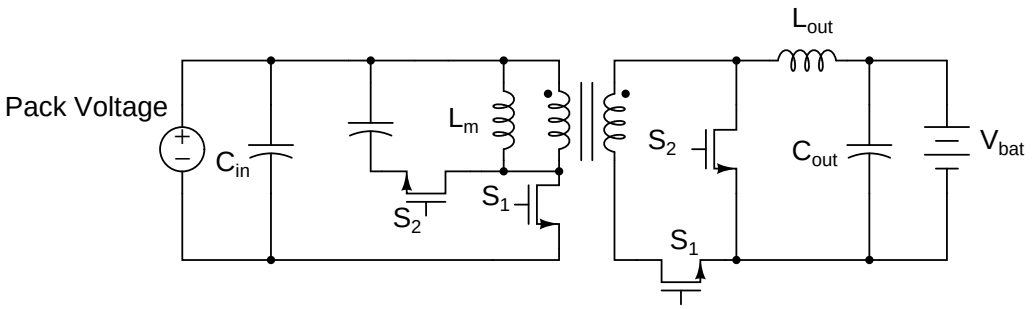

Figure 4.7: Topology for using a capacitor and a switch to remove the magnetizing inductance energy.

Another method, which is used in this work, is having a third winding in parallel with the input as shown in Figure 4.8 to recover the energy. It operates by pushing the energy back into the input when S_1 is off. The main concern with this design at high frequencies is minimizing reverse recovery losses in the diode and minimizing parasitic inductance in the winding. It is recommended to select the diode by emphasizing the reverse recovery time instead of minimizing the forward voltage drop.

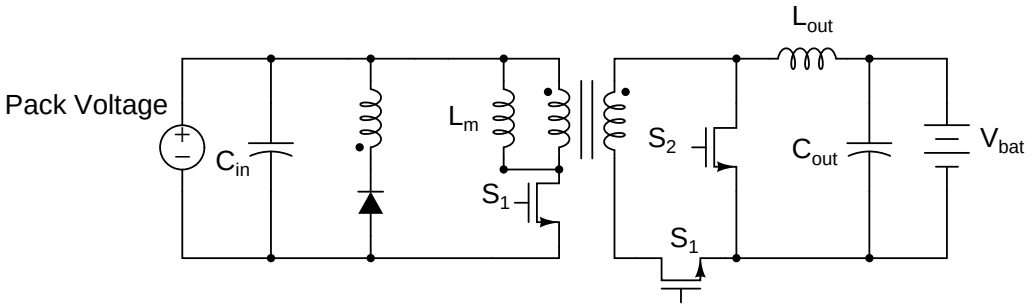

Figure 4.8: Topology for using a third winding to remove the magnetizing inductance energy.

4.3.3 Planar Magnetics

This converter uses planar magnetics in the transformer design. Planar magnetics are transformers where the windings are actually traces on the printed circuit board (Figure 4.9 from Ferroxcube [38]). The core drops into the board and needs to be secured by magnetic resin or a clamp. They are better than conventionally wound magnetics because they have a low profile,

low leakage inductance, and high surface area to volume ratio. The low profile is especially useful for applications like laptops and phones where all the components cannot be much taller than the board. The low leakage inductance occurs because the windings can be carefully optimized to cancel most of the stray magnetic fields, although this is at the expense of additional parasitic capacitance in the windings. The high surface area to volume ratio allows heat to dissipate much easier, which is important because magnetics performance generally suffers heavily over temperature.

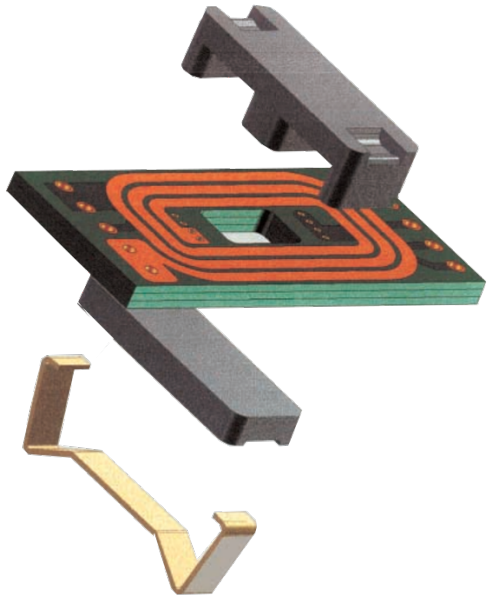

Figure 4.9: Basic planar magnetic configuration [38].

For this particular design, it is desirable to maximize the magnetizing inductance and minimize the leakage inductance since the transformer is not the energy storage element. The forward converter inherently minimizes leakage inductance because the primary and secondary currents flow at the same time. Therefore, they can be designed to cancel most of the parasitic fields. In a design like a fly-back, planar magnetics do not have this benefit because the current goes through the primary and secondary at different times.

Magnetizing inductance for a particular core can be calculated using equation 4.6 where A_l is the single turn magnetizing inductance and n is the number of turns.

$$L_m = A_l n^2 \tag{4.6}$$

Therefore the two ways to maximize the magnetizing inductance are by using a core with a bigger volume or higher permeability, or by adding additional windings. Increasing core area increases the board size but generally improves efficiency. Higher permeability cores tend to have lower resistance which reduces efficiency due to eddy losses in the core. To quantify the loss, the peak magnetic change B_{max} needs to be calculated by first finding λ which is demonstrated in equations 4.7 and 4.8.

$$\lambda = \frac{VD_{max}}{f} \tag{4.7}$$

$$B_{max} = \frac{\lambda}{2nA_c} \tag{4.8}$$

B_{max} along with the operating frequency can then be used to find the core loss in the data sheet on the material of the core. That will give a P_v value which can be multiplied by core volume to find total loss in equation 4.9.

$$P_{Loss} = P_v V_C \tag{4.9}$$

Additional windings increase the magnetizing inductance, but add to winding losses from the extra winding length and add to cost by requiring additional PCB layers. Increasing the winding lengths can significantly contribute to leakage inductance if the windings are not configured properly. This work is going to assume only a single winding is possible per layer given the high current requirements so the potential winding configurations for a four layer board are shown in Table 4.1.

Table 4.1: Potential winding combinations.

Winding Type	Non-Interleaved	Interleaved	Sandwiched
Layer 1	P	P	P
Layer 2	P	S	S
Layer 3	S	P	S
Layer 4	S	S	P

This analysis is done for an 8 layer board in [39] demonstrating the trade-off between winding capacitance and leakage inductance between all the meth-

ods. Non-interleaved has the highest leakage inductance but lowest winding capacitance. Interleaved has the lowest leakage inductance but the most capacitance. Sandwiched has slightly more leakage inductance but much less capacitance. The exact magnitude of these differences is highly dependent on PCB thickness, copper thickness, and trace widths, but the effect of each configuration intuitively still holds. Work in [40] analyzed the winding resistance and leakage inductance of these configurations using a systematic modeling approach. The conclusion is that non-interleaved has the most AC winding resistance, interleaved has less, and symmetric has the least. The conclusions on leakage inductance were also very similar except that the leakage inductance for the sandwiched configuration increases rapidly relative to the others at very high frequencies.

4.4 Converter Specific Calculations

When designing this forward converter, a big concern was how much energy needs to be processed in the magnetizing inductance. The Ferroxcube EQ25 core with the 3F46 material was chosen for its performance at higher frequencies especially around 1-2 MHz. The EQ25 was selected for its larger size to be able to conduct enough current in the internal transformer windings and to allow space for vias.

The first design consideration was the effect of the number of turns on the magnetizing power as seen in Figure 4.10. The general goal was to process less than 1 W of power, which this shows can be achieved by using two turns and switching at 1 MHz or higher. The rest of this analysis is going to assume to use two turns.

The next consideration was the effect on the input voltage on the magnetizing power that can be seen in Figure 4.11, which assumes a highest duty ratio of 0.35 and looks at a range of input voltages. This is relevant because in a bi-directional active balancing system, the input voltage changes as the cells charge which could vary the input from 8 V to 14 V. Operating at 1 MHz keeps the power processed under 1 W in the worst case which shows this frequency is high enough to reach the desired magnetizing power processed goal.

The next area to analyze is the effect of duty ratio on the magnetizing

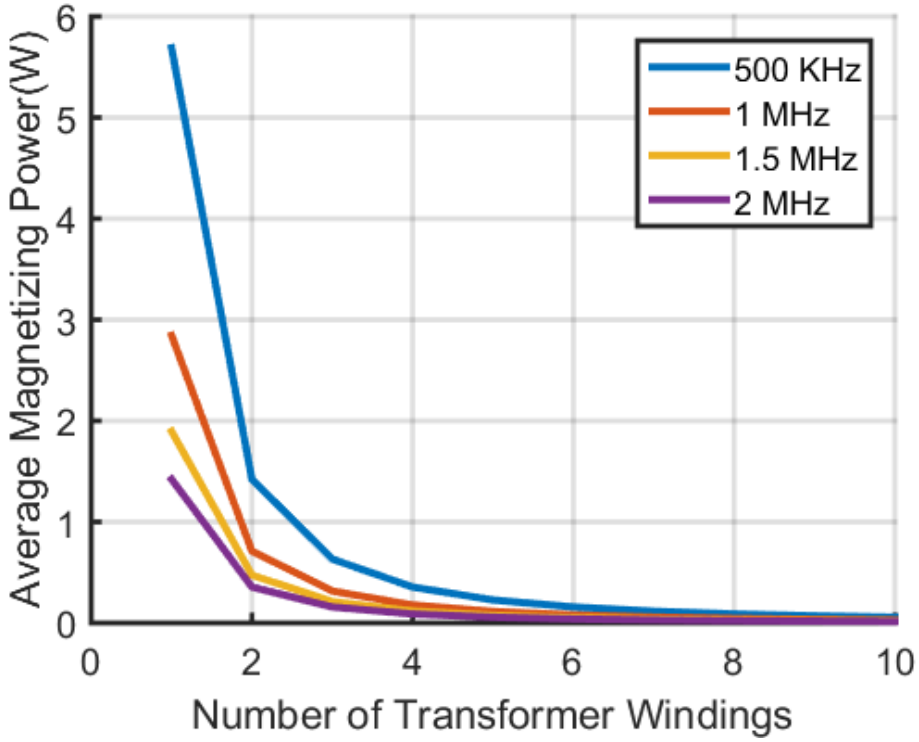

Figure 4.10: Magnetizing power relative to input voltage with D fixed at 0.35.

power which is shown in Figure 4.12. It can be seen from D=0.2 to D=0.3, the range where we are expected to operate, that there are diminishing returns on increasing the frequency much higher than 1 MHz.

The previous plots lead to the choice of two turns in the secondary and primary of the transformer and to operating at 1 MHz. Some critical design information about the transformer can be calculated now. The first is the effective magnetizing inductance which is given in equation 4.10.

$$L_m = A_l n^2 = (3.1\ \mu H)2^2 = 12.4\ \mu H \tag{4.10}$$

The next important values are B_{max} and λ which are given in equations 4.11 and 4.12.

$$\lambda = \frac{VD_{max}}{f} = \frac{(12)0.35}{1MHz} = 4.2 \times 10^{-6} \tag{4.11}$$

$$B_{max} = \frac{\lambda}{2nA_c} = \frac{4.2 \times 10^{-6}}{2 * 2 * 89.7\ mm^2} = 11.7\ mT \tag{4.12}$$

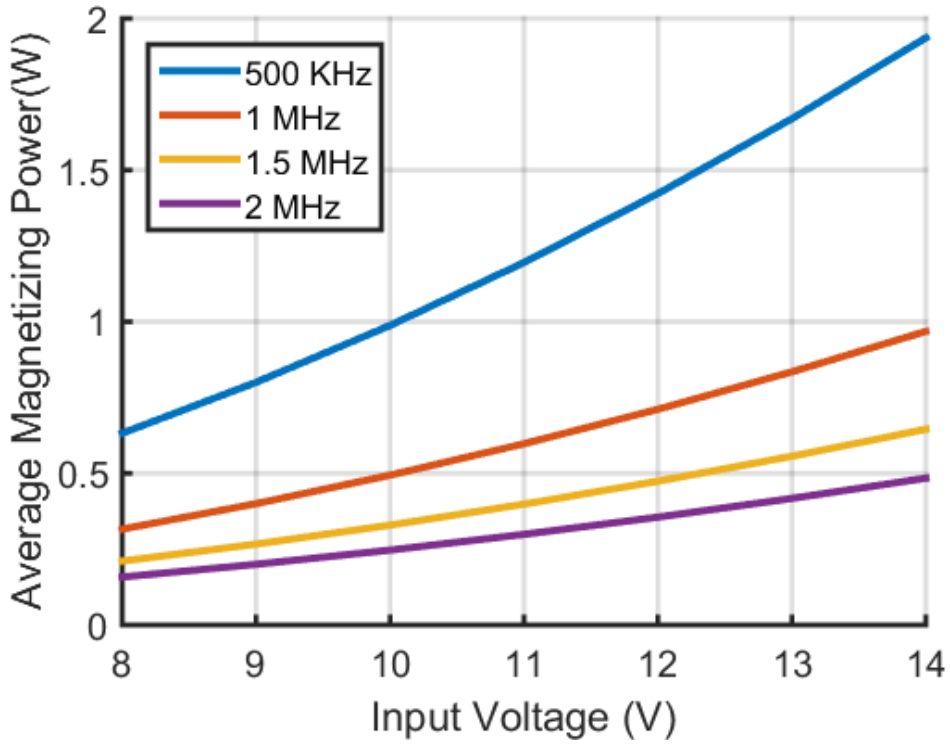

Figure 4.11: Magnetizing power relative to input voltage with D fixed at 0.35 and N=2.

The datasheet can then be used to find P_v for the given flux density which is $10\frac{kW}{m^3}$ [41]. The expected power loss can then be found in equation 4.13 which is very low.

$$P_{Loss} = P_v V_C = 10\ \frac{kW}{m^3} * 2370\ mm^3 = 23\ mW. \tag{4.13}$$

Another important factor in the design is the current ripple in the inductor. This current ripple is expected to be less than 0.5 A. The necessary inductance for this to happen can be calculated using equation 4.14. For the actual converter, a 4.7 μH inductor will be used.

$$L_{min} = \frac{\Delta t V_{out}}{\Delta I} = \frac{(1-D)V_{out}}{f\Delta I} = 4.5\ \mu H \tag{4.14}$$

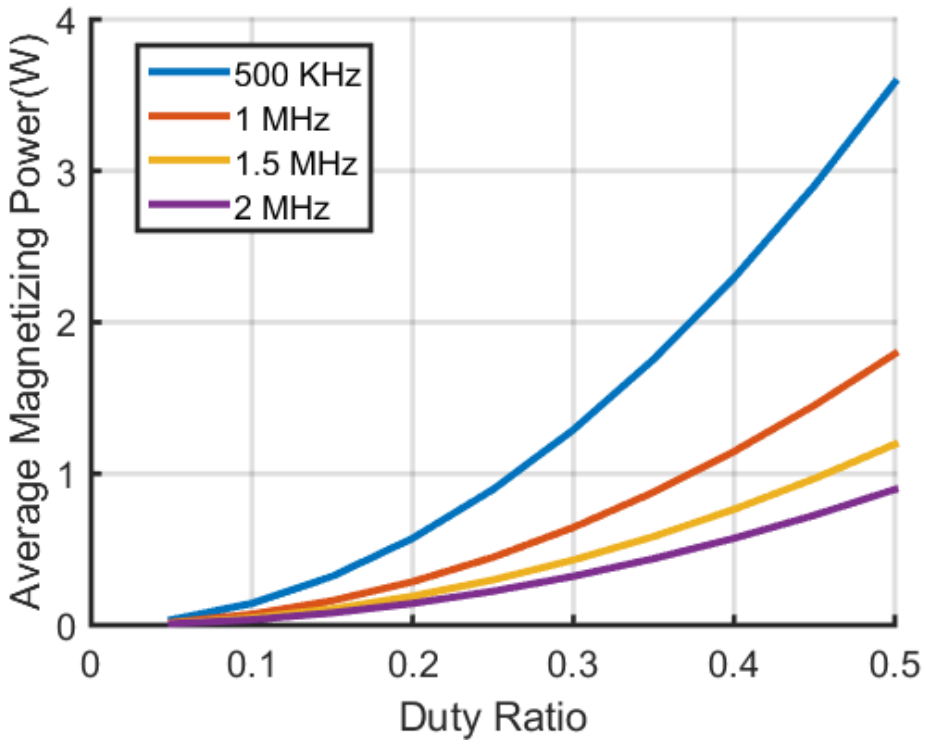

Figure 4.12: Magnetizing power relative to the duty ratio with V_{in} fixed at 12 V and N=2.

CHAPTER 5

EXPERIMENTAL RESULTS OF ACTIVE BATTERY MANAGEMENT SYSTEMS

5.1 Forward Converter

5.1.1 PCB Layout

The forward converter that was built can be seen in Figure 5.1. An annotated version can be seen if Figure 5.2 and the components used can be found in Table 5.1. It has connections for the input, output, and control power, and there are pin headers to communicate to the micro-controller for the PWM channels, digital communication, and voltage sensing. The goal of the layout was to minimize parasitic leakage inductances which lead to voltage ringing. There are two critical paths that experience high $\frac{di}{dt}$. The first is from the input capacitance, through the transformer, and through the low side switch. This was minimized by keeping the input capacitors close the transformer and by keeping this path short with thick traces. The second is on the isolated side by both of the switches and through the transformer which was also minimized with short path lengths and thick traces. The low leakage inductance of the planar transformer windings helps this ringing as well.

Another layout goal was to keep the gate driving paths very short to minimize the parasitic inductances. This is especially important because the converter is operating at a high frequency. This was achieved by putting the gate drivers on the back side of the PCB directly below the switches. This can be seen in Figure 5.3.

An important aspect of the design was maintaining proper isolation between the primary and secondary sides. Isolation was achieved by having two separate ground planes referenced to the primary and secondary ground. The isolated power supply and the digital isolator were placed directly between the two planes to maintain this isolation.

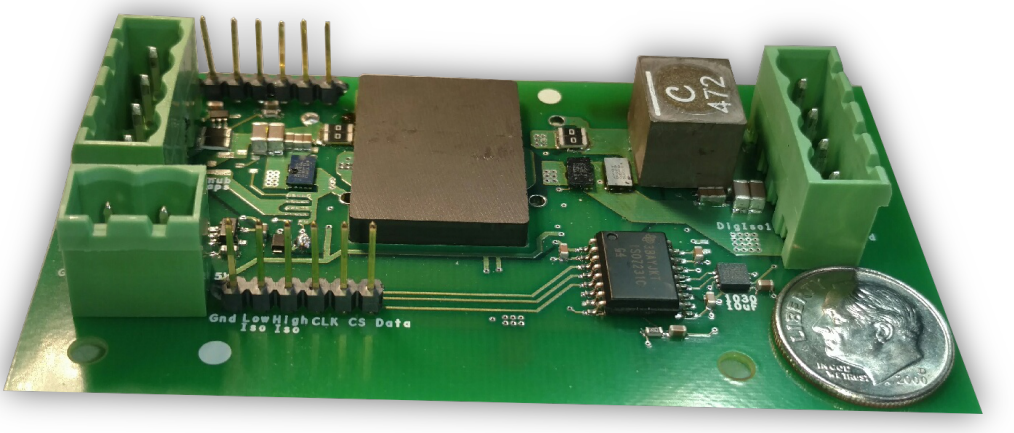

Figure 5.1: The forward converter for charging batteries.

The ADC that measures the output voltage was placed very close to the output but away from the switching nodes so that it did not pick up switching noise. The digital communication lines were also placed far away from the switching nodes and power planes for similar reasons. The current sense amplifier on the input side was placed extremely close to the voltage sensing resistor connected by parallel wires to minimize noise that feeds into the amplifier.

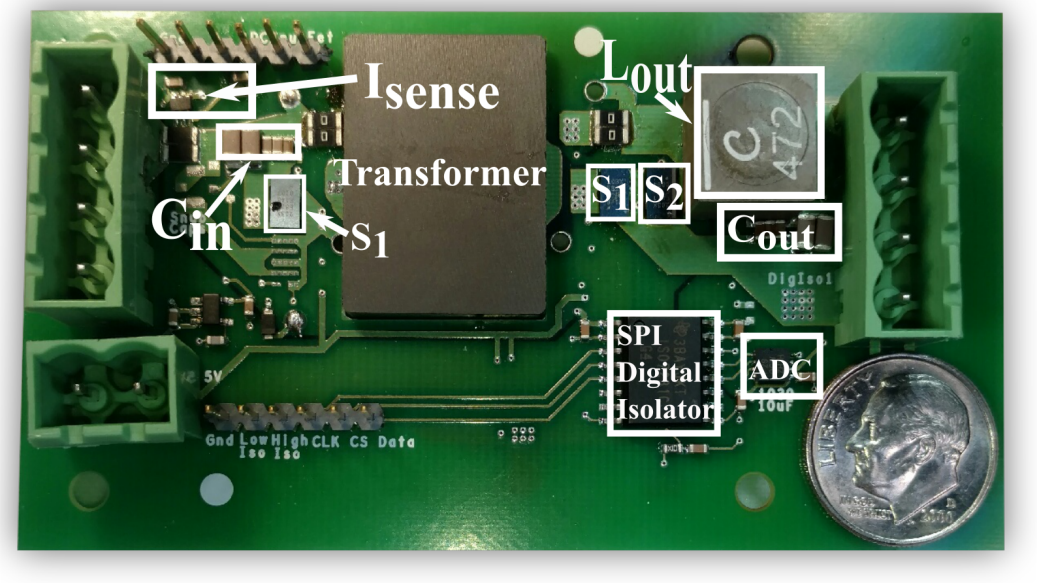

Figure 5.2: Top view of the forward converter.

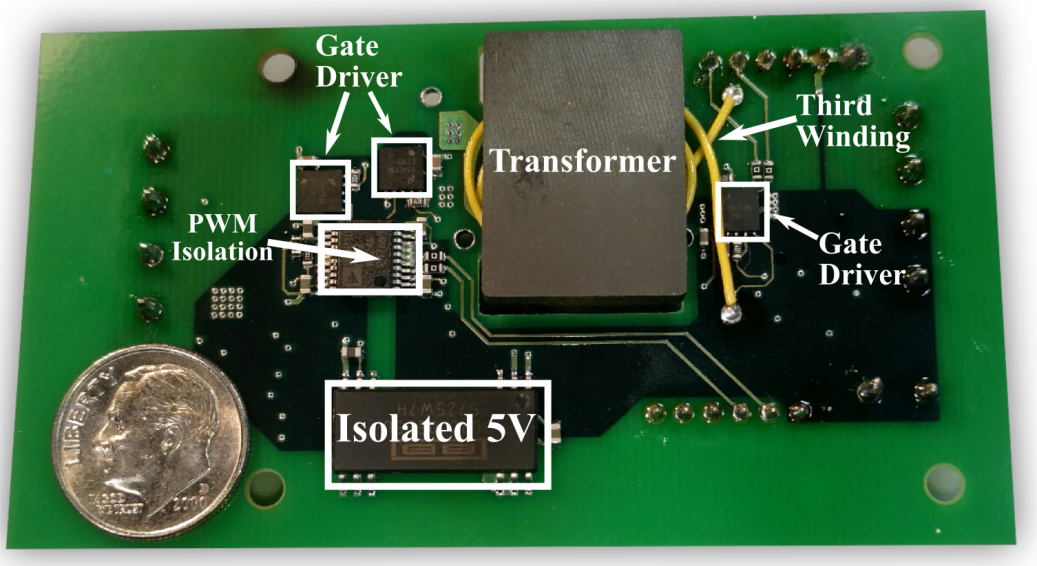

Figure 5.3: Back view of the forward converter.

Table 5.1: Critical component listing of the forward converter.

Component	Part Number	Parameters
S_1 - S_2	EPC2030	40 V, 31 A
Current Sense	INA213AIDCKR	Gain=50
Transformer	EQ25	Ferroxcube 3F46
Gate drivers	LM5113	GaN driver
ADC	ADS7829IDRBT	12-bit, 125KSPS
PWM Isolation	ADUM5210CRSZ	2.5 kV, 100 Mbps
Isolated Supply	DCR010505U/1K	5 V, 0.2 A
Digital Isolation	ISO7231CDW	2.5 kV 25 Mbps
L	XAL7030-472MEB	6.7 A, 4.7 μH
Micro-controller	TMS320F28069	

5.2 Control of the Battery Charger

5.2.1 Component Selection and Operation

In order to safely charge a battery, both the voltage and current into the battery must be monitored. The general methodology used to measure the voltage and current is shown in Figure 5.4. A common process for charging batteries is to use constant current until it reaches a nominal voltage and then use constant voltage to finish off the charge. The current decreases with time during the constant voltage period. One of the challenges associated with

charging a stack of battery cells is that every cell is located at a different potential requiring an isolated measurement on the cell voltage. This was achieved by using an analog-to-digital converter (ADC) on the cell, a digital isolator, and a small 5 V isolated power supply. The 5 V isolated power supply ensures there is enough voltage on the ADC and digital isolator to communicate properly. It is possible to use the battery power to supply these electronics, but this was not used due to the passive current draw that it would take from the cell.

An external ADC was used instead of transferring the analog voltage over isolation because this introduces large measurement error. The ADC is 12 bits which gives 1.22 mV of resolution which is sufficient for this design (equation 5.1). SPI was chosen as the communication protocol because it is easy to implement in both hardware and software and is capable of sufficient data rates. Two wire protocols such as I^2C require extra external components on the communication lines to properly communicate. The ADC operates using three wires which are a serial clock, chip select, and data line. A measurement is initiated by pulling the chip select line low. The measurement takes place for two serial clock cycles and then the ADC begins sending data with the most significant bit first. The C2000 SPI peripheral function was used to initiate and read this data. A consideration with using this function is that every bit during the clock cycle including the initial two serial clock cycles is read. Therefore, a simple bitwise operation needs to be used to eliminate these two useless bits of data. Another step in the voltage measurement process is scaling the ADC measurement to an actual voltage for the control algorithm which can be calculated by using equation 5.2.

$$V_{resolution} = \frac{V_{range}}{2^{ADCBits}} = \frac{5}{2^{12}} = 1.22\ mV \tag{5.1}$$

$$V_{actual} = \frac{5ADC_{measurement}}{2^{12}} \tag{5.2}$$

In order to sense current, it was decided to use a current sense resistor and amplifier. The other reasonable option was a Hall-effect current sensor but these are typically ineffective at higher frequencies. The issue with using the current sense resistor is that it introduces loss into the system, but this can be mitigated by using a very small resistance and a high gain amplifier.

Location of this sensor affects how the data is extracted from it. The first

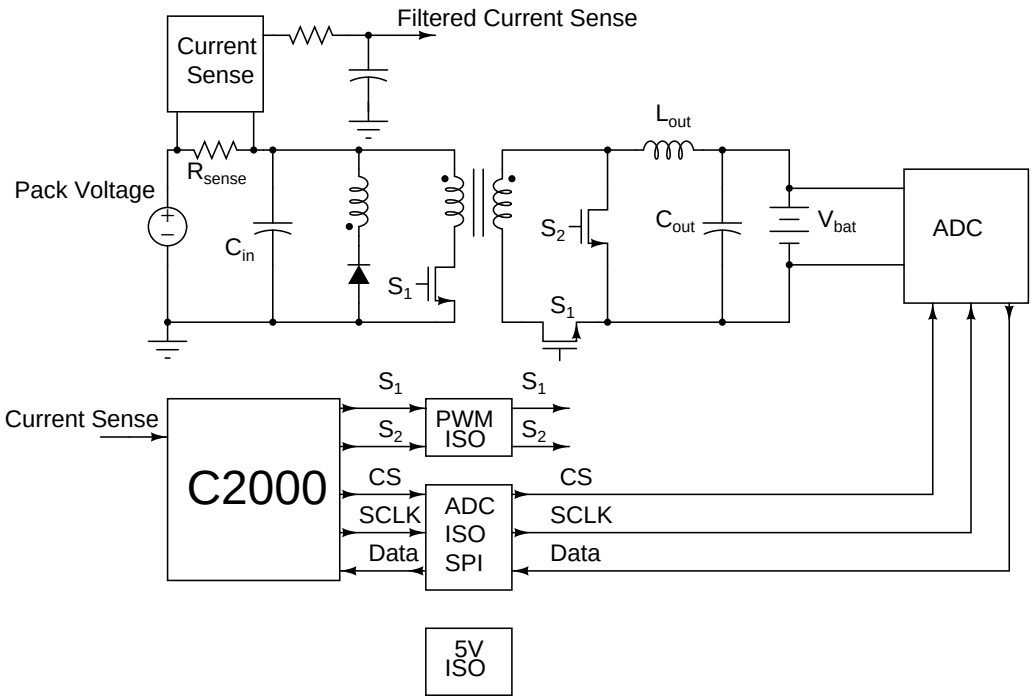

Figure 5.4: Schematic of the control of the battery.

consideration is to put it on the input or output side of the isolation. The benefits to the output side are that it has a relatively constant current, making the voltage on the output cleaner, and that it factors in the inefficiency of the converter. The negatives with this are that the measurement needs to be sent back to the non-isolated side of the circuit, requiring either an external ADC on the isolated side with isolated digital communication (like the voltage measurement) or sending an analog voltage through isolation. It is undesirable to have too many digital lines into a micro-controller, and the error introduced by passing the analog voltage through isolation is not accurate.

The next place to consider is sensing on the input side. This side is advantageous because it does not require isolation, so the current sensing voltage can be passed directly into the micro-controller. The issue with this side is that the current is not continuous; therefore, much more high frequency content enters the current sensor. The current being measured on the input is not the same as the current going into the battery, so the current needs to be scaled by the duty ratio. This also does not factor in the efficiency of the converter reducing the output current, which can be seen in Figure 5.3.

The placement of the current sensor on the high or low side affects its

performance. Low side current sensing is beneficial because it is easy to amplify the voltage, although it creates two ground planes which can introduce communication errors at higher currents. High side current sensing benefits from not disrupting the ground plane, but the amplifier performance suffers at high frequencies due to the required common mode rejection. This application uses high side current sensing on the input side because having a single ground plane and avoiding additional digital communication were the most important factors.

$$I_{out} = \frac{I_{in}}{D}\eta \tag{5.3}$$

5.2.2 Current Sensor

The current sensor was designed with $R = 25\ m\Omega$ and an instrumentation amplifier gain of 50. The maximum input current that can be sensed is calculated in 5.4, which means the maximum output current that can be sensed is around 10 A depending on the duty ratio.

$$I_{In,max} = \frac{V_{max}}{R_{sense}G} = 4\ A \tag{5.4}$$

In order to characterize the current sensor effectively, it was calibrated with the efficiency taken into account. This is achieved by mapping I_{out} to $\frac{I_{in}}{D10}$. The factor of 10 is necessary for limitations with using 16-bit numbers in the processor. The data was taken for V_{out} equal to 3 and 3.5 and can be seen in Figure 5.5.

An important design consideration when sending an analog voltage to a different board that does not share the same ground plane is the effect of switching noise. This happens when the power converter sends the current sensor voltage to the C2000 and can be seen in Figure 5.6. This noise can be avoided by carefully selecting the sampling time and averaging. The cleanest sampling time is at the end of the period as shown in Figure 5.7. This is achieved by moving the C2000 ADC conversion time to the falling edge of the PWM channel and increasing the sampling window. See the code in Appendix B for specific details. Averaging is used over 1024 current samples to decrease the effect of noise on the measurement. This value is then used to update the duty ratio to control the current to the output. The update

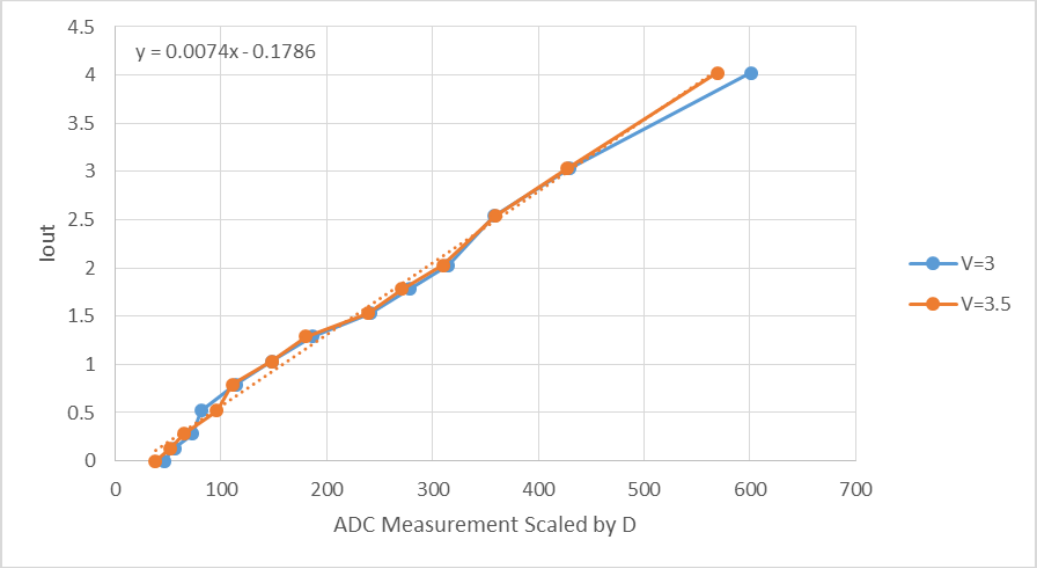

Figure 5.5: Calibration of the current sensor.

time can be calculated using 5.5.

$$T_{update} = \frac{averageSamples}{f_{sample}} = \frac{1024}{1\ MHz} = 1.024\ ms \tag{5.5}$$

The basic control algorithm to charge the battery is to measure the output voltage and see if it has reached the maximum voltage limit. If it has not, then it should stay in constant current mode. When it reaches the maximum battery voltage, it should go into constant voltage mode to trickle charge the battery. The basic methodology can be seen in Figure 5.8.

Constant current and voltage control are accomplished by basic hysteresis control. The threshold to continue improving the duty ratios is limited by the resolution of the PWM channels. If this threshold is set too narrow, the control algorithm will oscillate between two different duty ratios that surround the nominal output voltage or current.

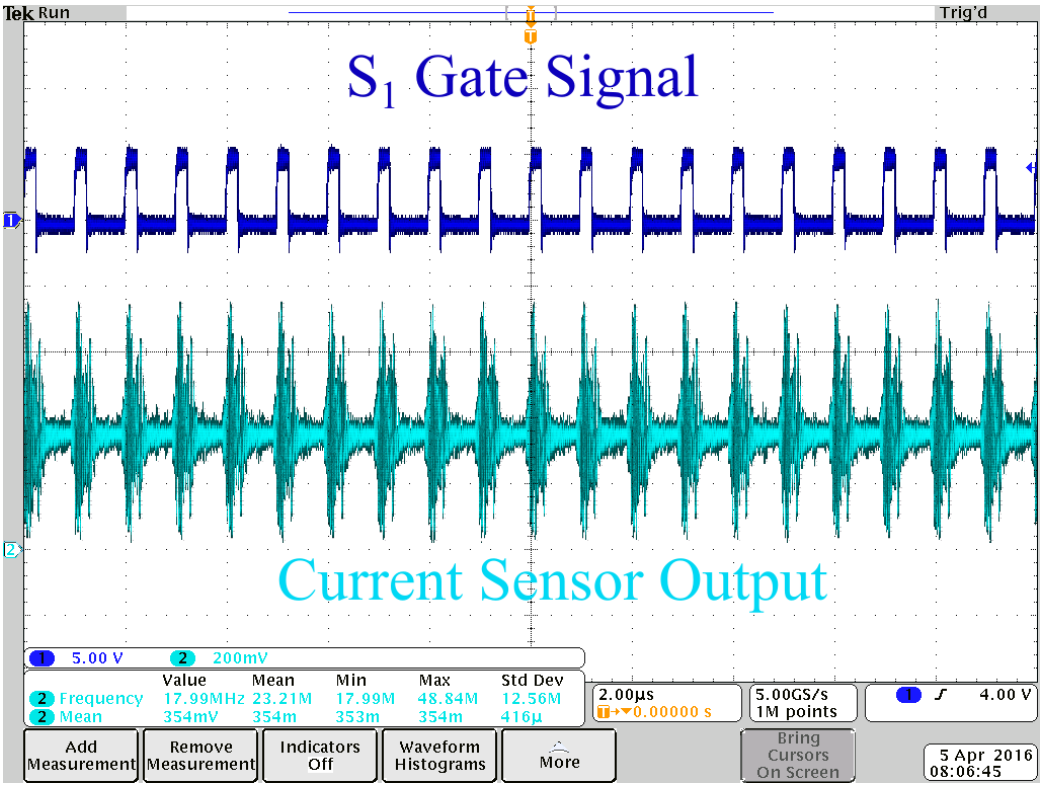

Figure 5.6: Current sensor output due to passing the signal over a poor ground connection.

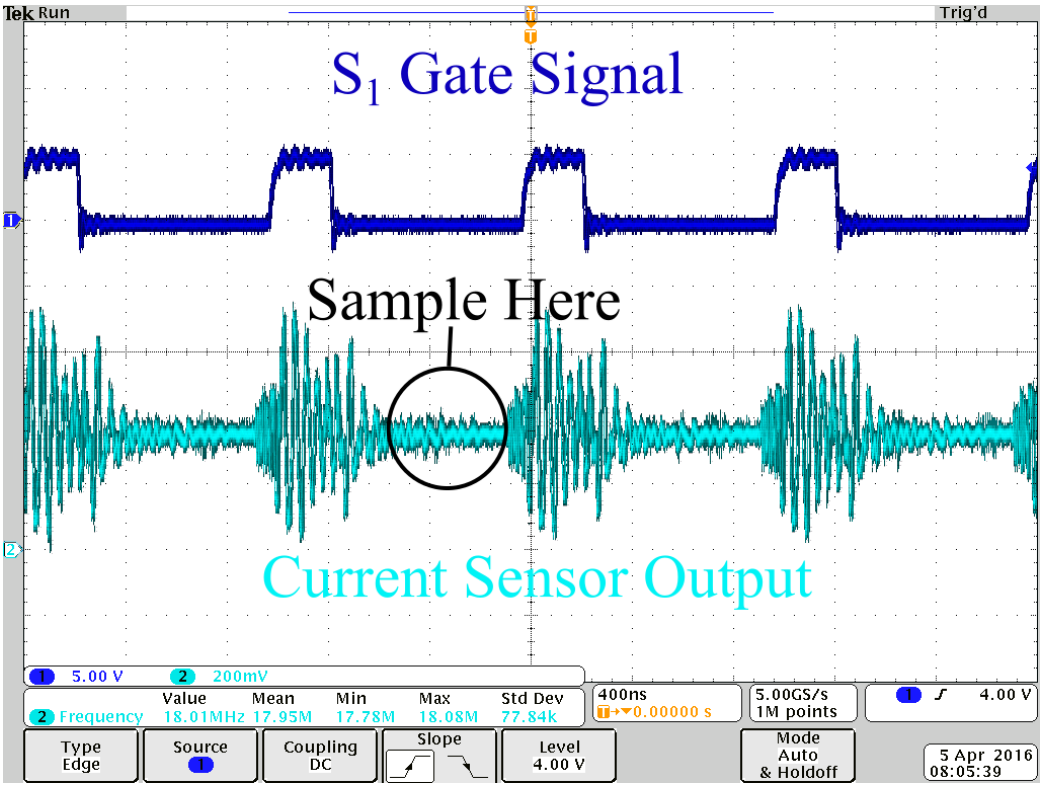

Figure 5.7: Short time period of the current sensor output.

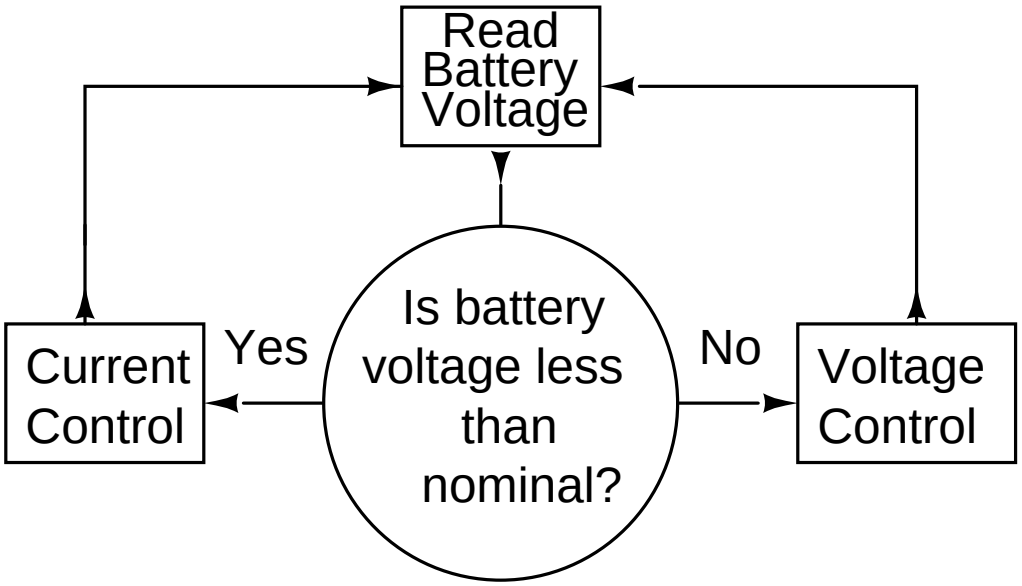

Figure 5.8: Basic control diagram to charge the battery.

5.3 Converter Performance

The efficiency of the power stage can be seen in Figure 5.9 and the efficiency of the whole battery charger can be seen in Figure 5.10. The difference in efficiency between the two is a constant 650 mW that is consumed by the digital isolation and the gate driving circuitry. For an actual system, a voltage measurement IC would be used, removing the digital isolation that consists of 60% of the 650 mW. The other 40% that goes to the gate drivers can be reduced by switching at a lower frequency or using a smaller switch with less gate capacitance.

5.4 Core Reset

An interesting result from this was the core reset mechanism. At higher duty ratios, the reset mechanism operated as expected which can be seen in Figure 5.11. The ringing on the current is attributed to high parasitic inductance in the third winding. This can be reduced in future iterations by embedding the third winding trace in the PCB.

At duty ratios less than 0.3, no reset current actually flows through the third winding. Instead, it is believed to reset through parasitic capacitance in

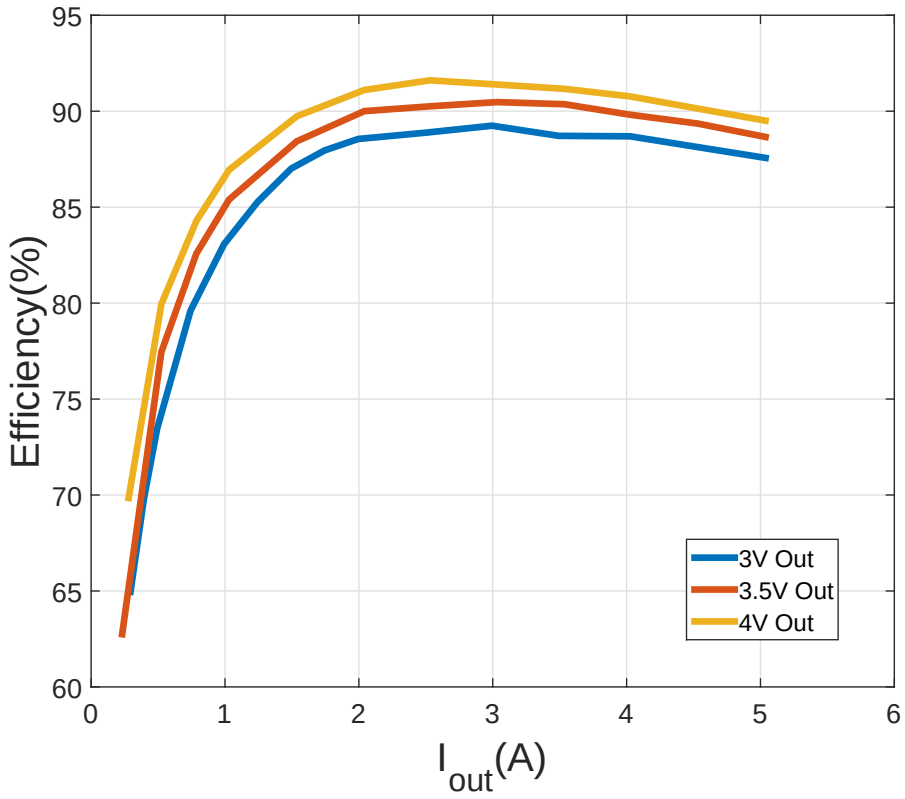

Figure 5.9: Efficiency of the power stage of the battery charger.

the planar magnetics windings. It actually discharges in a resonant manner which can be seen in Figure 5.12. Further modeling and calculations are required to validate this.

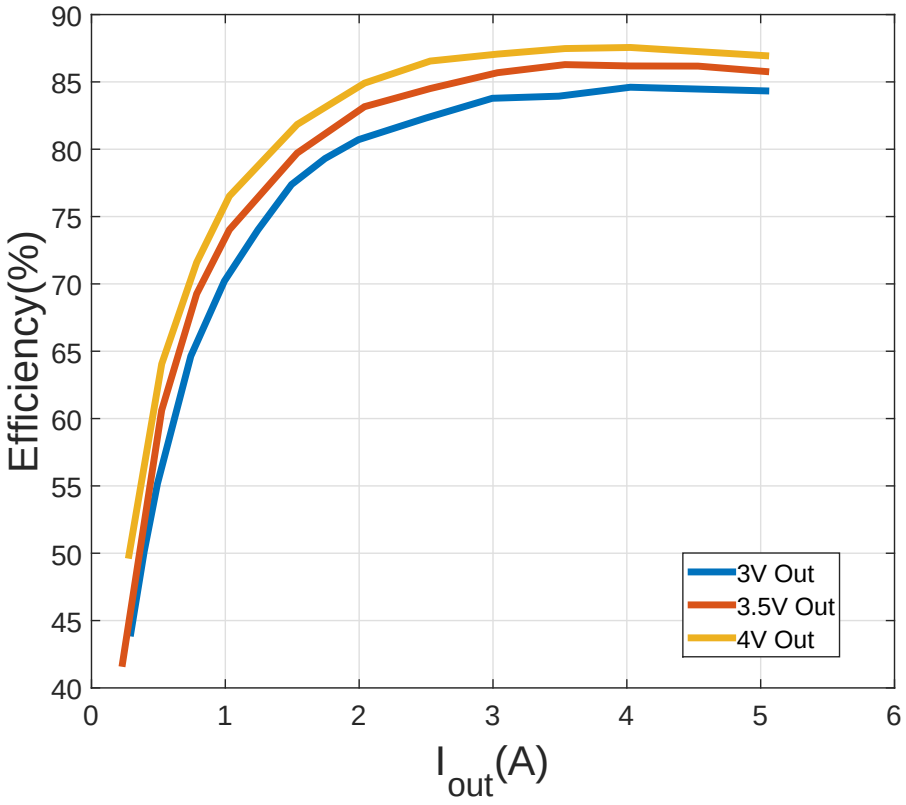

Figure 5.10: Efficiency of the whole battery charger.

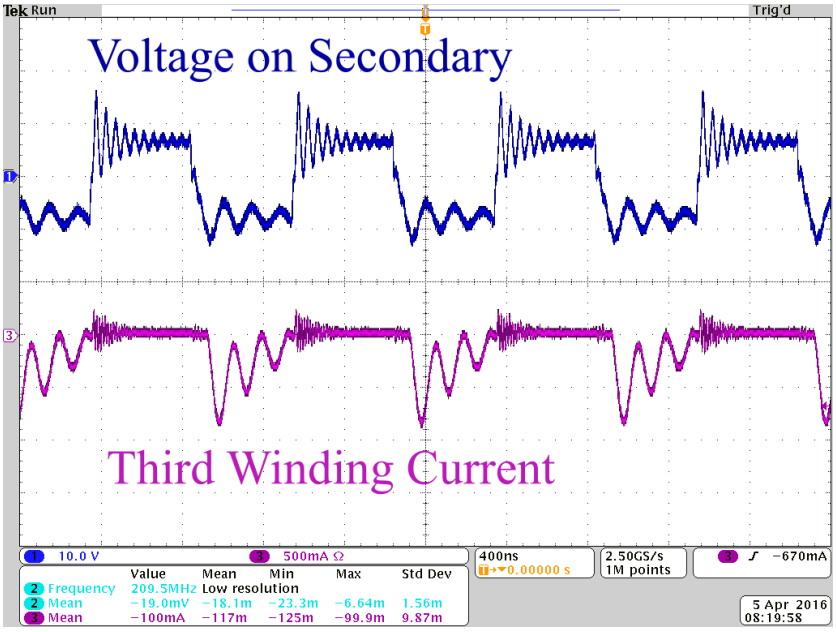

Figure 5.11: The volt-second balance on the secondary winding with a 7 V input and a 0.5 duty ratio.

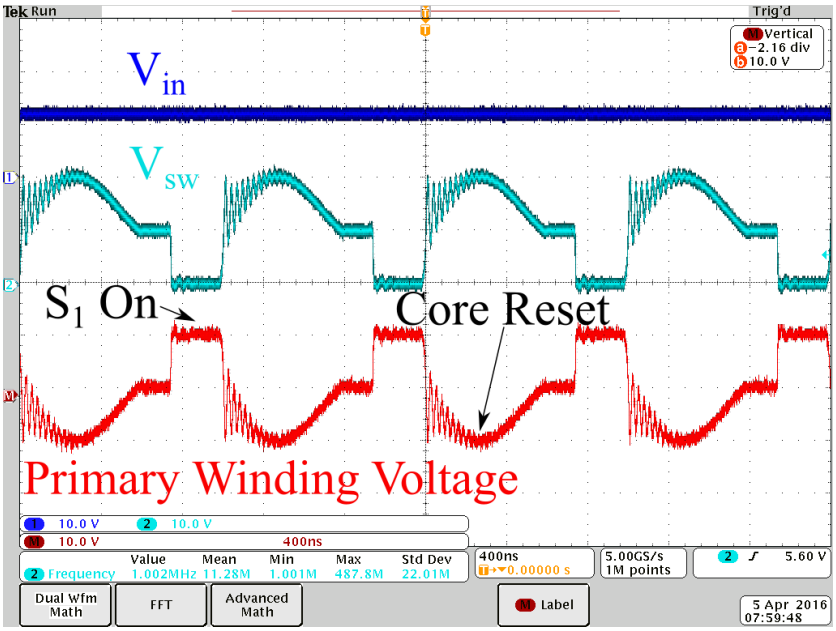

Figure 5.12: The volt second-balance on the primary winding with a 10 V input and a 0.25 duty ratio.

5.5 Charging Batteries

A condition was established for a two cell pack where the high cell has much less voltage than the other. The isolated charger can efficiently charge up this cell without dissipating heat from the other battery cell. The cell was charged using purely constant current and the voltage over time can be seen in Figure 5.13. K2 22650 battery cells were used and it took 70 minutes to charge at 1.9 A. The sawtooth shape of the charge current is caused by the hysteresis control and the limited resolution of the PWM channel at 1 MHz.

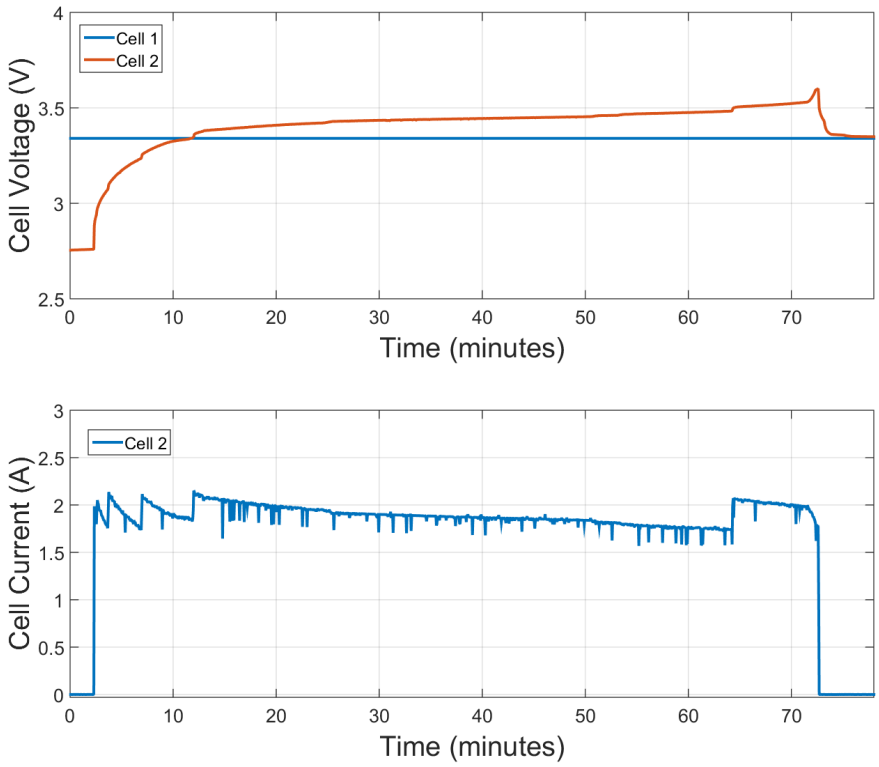

Figure 5.13: Voltage and current of the batteries charging over time.

CHAPTER 6

CONCLUSION AND FUTURE WORK

A resonant interleaved Dickson switched-capacitor converter was developed to convert 20 V to 120 V with a peak power of 60 W. A peak efficiency of 95.7% was achieved and the board had an enclosed box power density of 157 $\frac{W}{in^3}$. This efficiency and power density suggest that this converter is a good candidate for compact voltage step-up applications.

Future work on this converter involves improving the layout and reaching a better optimization between switching frequency, gate resistance, and switch blocking voltage. Another area that would be interesting to investigate would be expanding the design to higher voltage conversion ratios. This converter can also be simplified by using diodes instead of switches on $S_5 - S_{10}$. Future work could involve analyzing the effect of diodes on efficiency and power density along with experimental validation.

A bi-directional isolated forward converter was developed and tested utilizing GaN switches, planar magnetics, and a 1 MHz operating frequency. The converter has a peak efficiency of 91.6% and a peak power of 20 W. It was shown to be capable of charging a battery pack using current control for an extremely imbalanced pack.

Future work involves implementing differential power processing using the bi-directional capabilities of the converter. A useful step would involve embedding the micro-controller and all the power converters on a single board, reducing the current sense measurement noise and setup complexity. This board would be a useful testbed for battery charging optimization. For example, it would be interesting to investigate methods for detecting the weak cells early and optimizing the charge rate to reduce the overall processed power.

APPENDIX A

INTERLEAVED DICKSON

A.1 Schematics

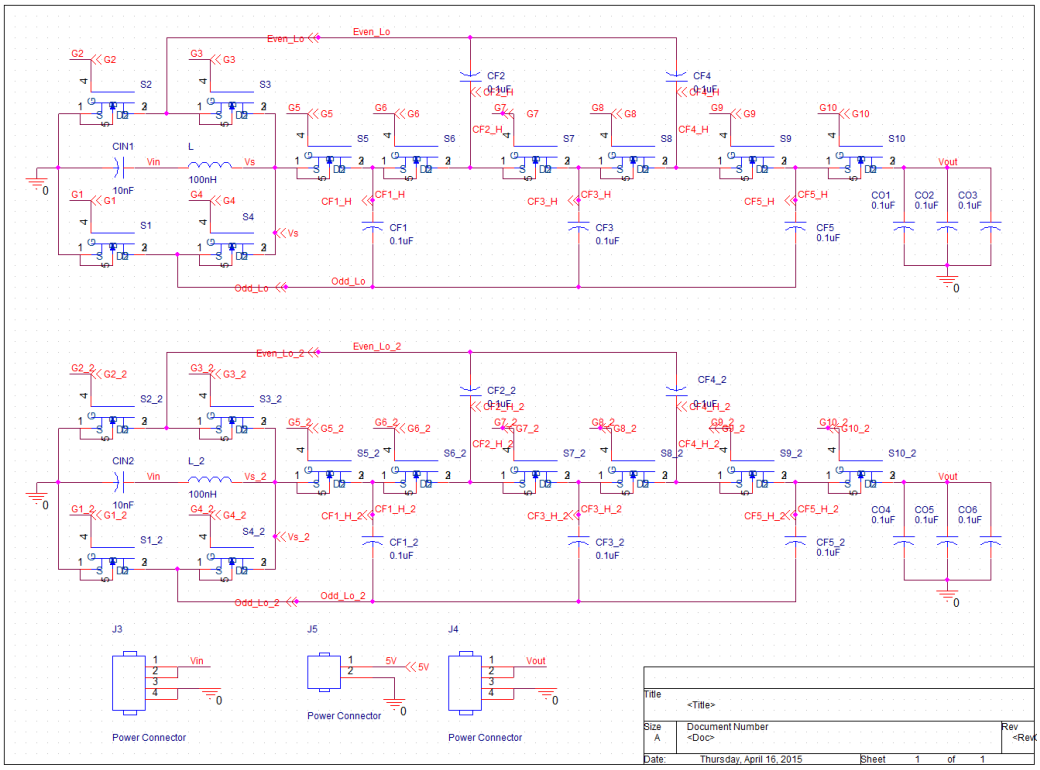

Figure A.1: Schematic of the main power path of the interleaved Dickson.

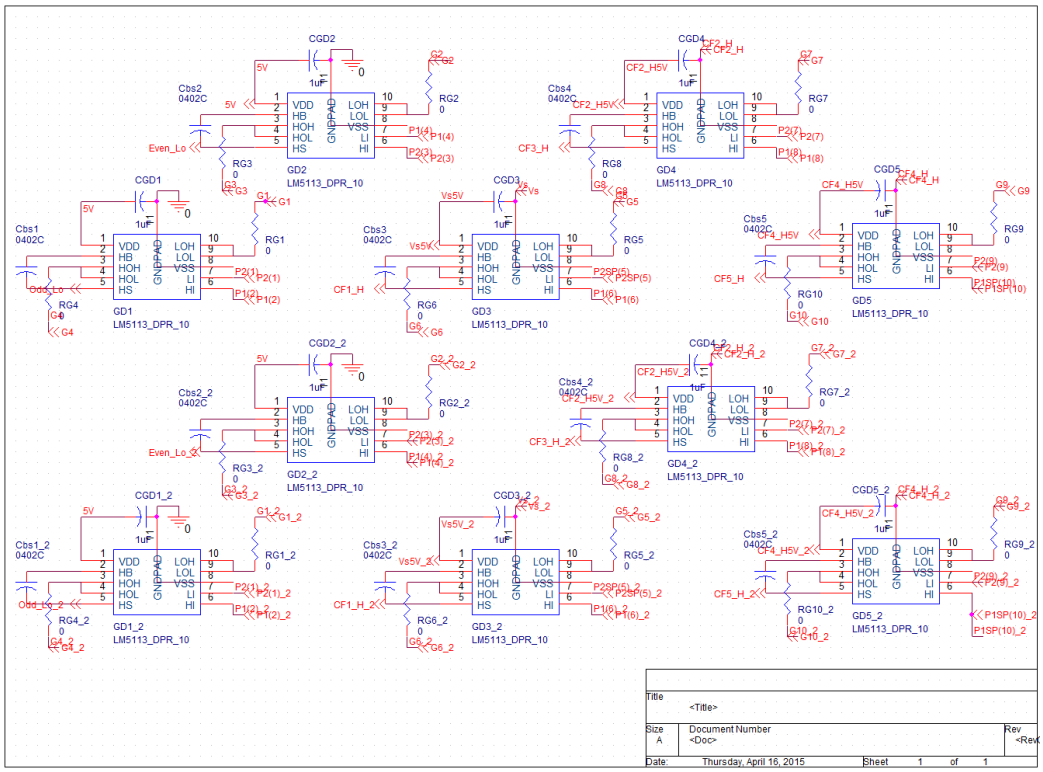

Figure A.2: Schematic of the gate drivers for the interleaved Dickson.

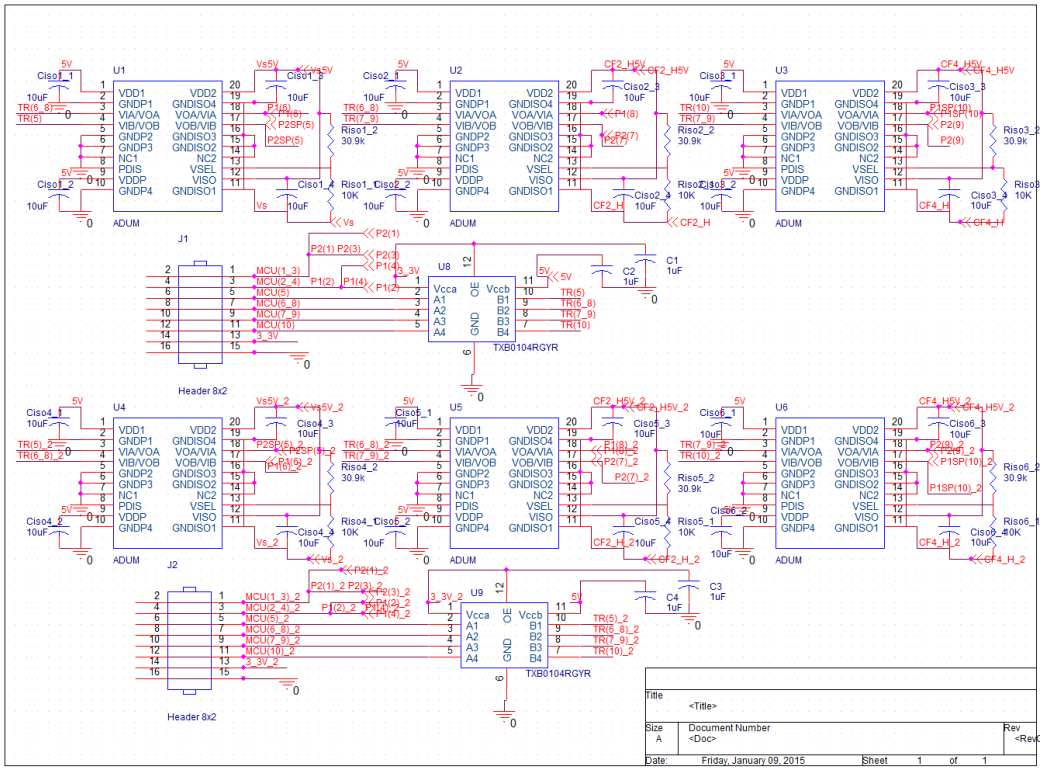

Figure A.3: Schematic of the level shifting circuitry for the interleaved Dickson.

A.2 PCB Layout

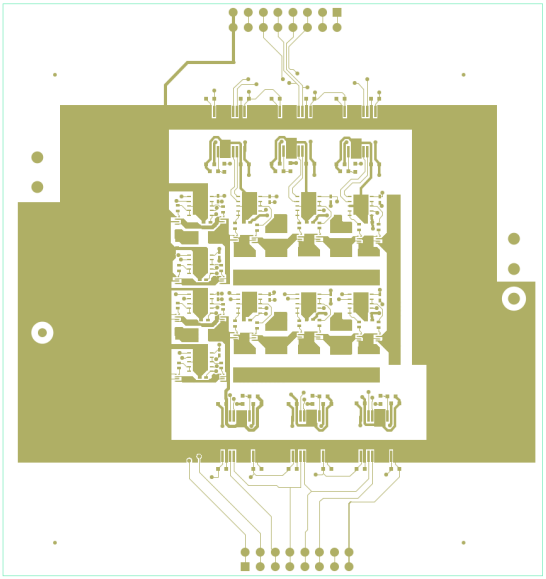

Figure A.4: Top copper layer of the Dickson.

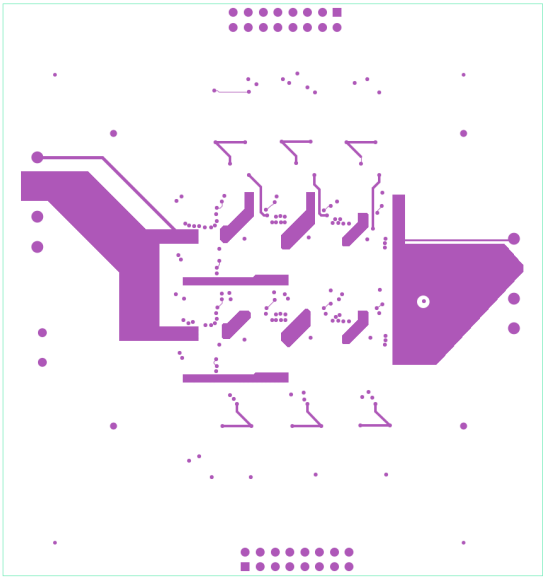

Figure A.5: Bottom copper layer of the Dickson.

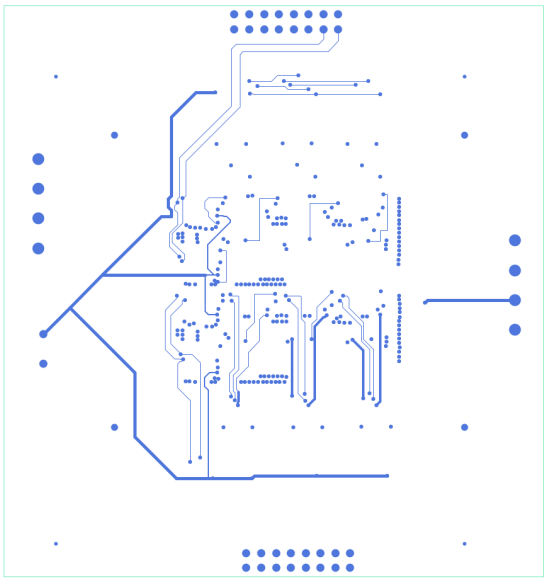

Figure A.6: Internal copper layer of the Dickson.

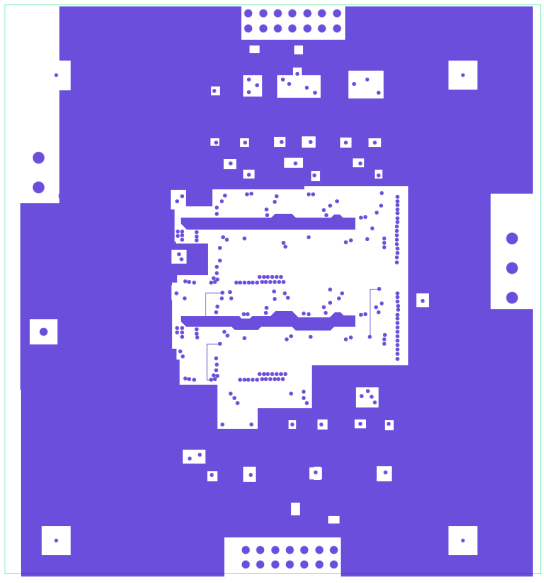

Figure A.7: Internal copper layer of the Dickson.

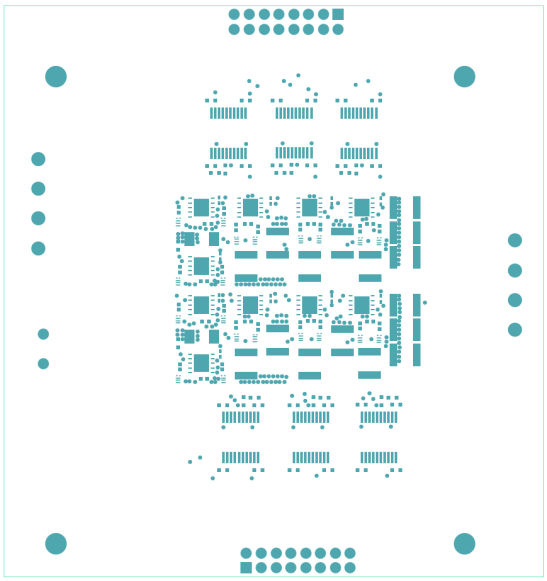

Figure A.8: Top solder mask of the Dickson.

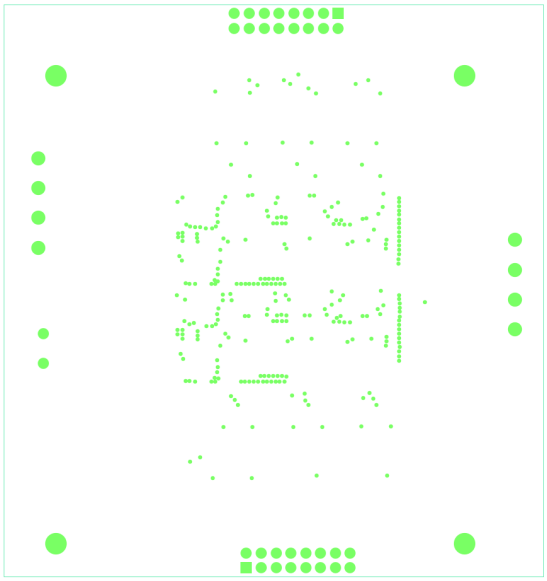

Figure A.9: Bottom solder mask of the Dickson.

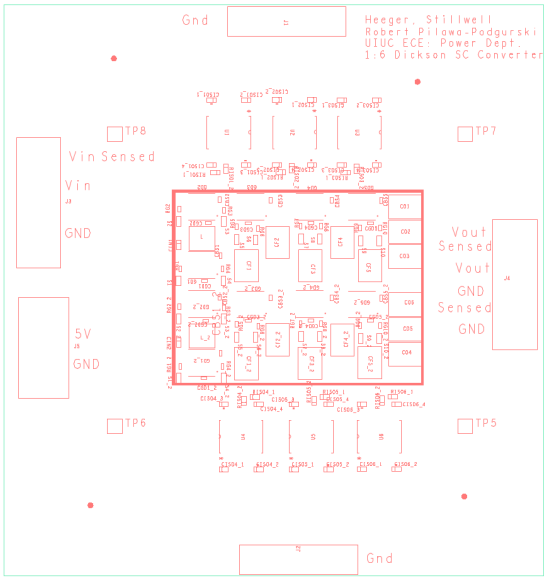

Figure A.10: Silk screen of the Dickson.

A.3 Micro-controller Code

```
1
2 //###########################################################
3 //Derek Heeger, Andrew Stillwell, Ben Macy
4 //Advisor: Robert Pilawa Podgurski
5 //Interleaved Dickson Converter
6 //Design based heavily off of Texas Instrument's HRPWM
      example from code composer studio
7 //###########################################################
8
9 #include "F2806x_Device.h"          // F2806x Headerfile
10 #include "F2806x_Examples.h"       // F2806x Examples
      Headerfile
11 #include "F2806x_EPwm_defines.h"   // useful defines
      for initialization
12
13 // Declare your function prototypes here
14 //-----------------------------------------
15
16 void HRPWM1_Config(Uint16);
17 void HRPWM2_Config(Uint16);
18 void HRPWM3_Config(Uint16);
19 void HRPWM4_Config(Uint16);
20 void HRPWM5_Config(Uint16);
21 void HRPWM6_Config(Uint16);
22
23 // General System nets - Useful for debug
24 Uint16 i,j, DutyFine, n,update;
25
26 Uint32 temp;
27 float D, DP1SP, DP2SP;
28 int delay1, delay2, delay3, delay4, delay5, delay6;
29 int T, MEP;
30 int master;
31
```

```
32
33
34 void main(void)
35 {
36
37 master = 1;
38 InitSysCtrl();
39
40 InitEPwm1Gpio();
41 InitEPwm2Gpio();
42 InitEPwm3Gpio();
43 InitEPwm4Gpio();
44 InitEPwm5Gpio();
45 InitEPwm6Gpio();
46 // Master Code
47
48 if(master==1){
49 EALLOW;
50 EPwm1Regs.TBCTL.bit.PHSEN= TB_DISABLE;
51 EPwm1Regs.TBCTL.bit.SYNCOSEL = TB_CTR_ZERO;
52 GpioCtrlRegs.GPBDIR.bit.GPIO33 = 1;
53 GpioCtrlRegs.GPBMUX1.bit.GPIO33 = 2;
54 EDIS;
55 }
56 if(master==0){
57 //Slave Code
58 EALLOW;
59 EPwm1Regs.TBCTL.bit.PHSEN= TB_ENABLE;      //enable off
      of external signal
60 EPwm1Regs.TBCTL.bit.SYNCOSEL = TB_CTR_ZERO;
      //TB_CTR_ZERO
61 GpioCtrlRegs.GPBDIR.bit.GPIO32 = 0;    // set GPIO to
      input
62 GpioCtrlRegs.GPBMUX1.bit.GPIO32 = 2;   //mux
63 EDIS;
64 }
```

```
65 //Slave Code
66 EALLOW;
67 GpioCtrlRegs.GPBMUX1.bit.GPIO34 = 0;
68 GpioCtrlRegs.GPBDIR.bit.GPIO34 = 1;
69 EDIS;
70
71
72 DINT;
73
74 InitPieCtrl();
75
76 // Disable CPU interrupts and clear all CPU interrupt
       flags:
77 IER = 0x0000;
78 IFR = 0x0000;
79
80 InitPieVectTable();
81
82 // Step 4. Initialize all the Device Peripherals:
83 // This function is found in F2806x_InitPeripherals.c
84 // InitPeripherals();  // Not required for this example
85
86 // For this example, only initialize the EPwm
87 // Step 5. User specific code, enable interrupts:
88
89 update =0;
90 DutyFine =0;
91
92 EALLOW;
93 SysCtrlRegs.PCLKCR0.bit.TBCLKSYNC = 0;
94 EDIS;
95
96 // Some useful Period vs Frequency values
97 //  SYSCLKOUT =      80 MHz
98 //  ----------------------------
99 //  Period           Frequency
```

```
100 //   1000            80 kHz
101 //   800             100 kHz
102 //   600             133 kHz
103 //   500             160 kHz
104 //   250             320 kHz
105 //   200             400 kHz
106 //   100             800 kHz
107 //   50              1.6 Mhz
108 //   25              3.2 Mhz
109 //   20              4.0 Mhz
110 //   12              6.7 MHz
111 //   10              8.0 MHz
112 //   9             8.9 MHz
113 //   8             10.0 MHz
114 //   7             11.4 MHz
115 //   6             13.3 MHz
116 //   5             16.0 MHz
117
118 //===========================================
119 // ePWM and HRPWM register initialization
120 //===========================================
121
122 T=160;          // Period = 65 for .1uF, .1uH. period
      =225 for .2uF, 1uH
123 D = .5;         // duty is time in first position as
      indicated by 'down up' etc
124 DP1SP = .34;       //
125 DP2SP = .34;       //
126
127 delay1 = 4;
128 delay2 = 4;      // was 4
129 delay3 = 4;        // was 5
130 delay4 = 2;
131 delay5 = 5;
132 delay6 = 3;
133
```

```
134 MEP = 40;
135
136 HRPWM1_Config(T);       // ePWM1 target, Period = T,
       down up
137 HRPWM2_Config(T);       // ePWM2 target, Period = T, up
       down
138 HRPWM3_Config(T);       // ePWM3 target, Period = T,
       down up
139 HRPWM4_Config(T);       // ePWM4 target, Period = T, up
       down
140 HRPWM5_Config(T);       // ePWM5 target, Period = T,
       down up
141 HRPWM6_Config(T);       // ePWM6 target, Period = T, up
       down
142
143
144 EPwm1Regs.CMPA.half.CMPAHR = MEP+15<< 8;        // F,
       delay => shift to right
145 EPwm2Regs.CMPA.half.CMPAHR = MEP+20 << 8;       // F
146 EPwm3Regs.CMPA.half.CMPAHR = MEP+12 << 8;       // F
147 EPwm4Regs.CMPA.half.CMPAHR = MEP << 8;         // F
148 EPwm5Regs.CMPA.half.CMPAHR = MEP+12 << 8;       // F
149 EPwm6Regs.CMPA.half.CMPAHR = MEP+10 << 8;       // F
150
151 EALLOW;
152 SysCtrlRegs.PCLKCR0.bit.TBCLKSYNC = 1;
153 EDIS;
154
155
156 GpioDataRegs.GPBSET.bit.GPIO34=1;
157 while (1)
158 {
159 long     i;
160 for (i = 0; i < 8000000; i++) {
161
162 }
```

```
163
164 GpioDataRegs.GPBTOGGLE.bit.GPIO34=1;
165 }
166 }
167 void HRPWM1_Config(Uint16 period)
168 {
169 // ePWM1 register configuration with HRPWM
170 // ePWM1A toggle low/high with MEP control on Rising
      edge
171
172 EPwm1Regs.TBCTL.bit.PRDLD = TB_IMMEDIATE;           //
      set Immediate load
173 EPwm1Regs.TBPRD = period-1;                          //
     PWM frequency = 1 / period
174 EPwm1Regs.CMPA.half.CMPA = period *D-2;
      // set duty 50% initially
175 EPwm1Regs.CMPA.half.CMPAHR = (1 << 8);               //
      initialize HRPWM extension
176 //  EPwm1Regs.TBPHS.all = 0;
177 EPwm1Regs.TBPHS.half.TBPHS = period/2;          //
      Time-Base Phase Register, master's phase = 0
178 EPwm1Regs.TBCTR = 0;
179
180 EPwm1Regs.TBCTL.bit.CTRMODE = TB_COUNT_UP;
181 //  EPwm1Regs.TBCTL.bit.PHSEN = TB_DISABLE;
      // EPwm1 is the Master
182 //  EPwm1Regs.TBCTL.bit.SYNCOSEL = TB_CTR_ZERO;
183 EPwm1Regs.TBCTL.bit.HSPCLKDIV = TB_DIV1;
184 EPwm1Regs.TBCTL.bit.CLKDIV = TB_DIV1;
185
186 EPwm1Regs.CMPCTL.bit.LOADAMODE = CC_CTR_ZERO;
187 EPwm1Regs.CMPCTL.bit.SHDWAMODE = CC_SHADOW;
188
189 EPwm1Regs.AQCTLA.bit.ZRO = AQ_SET;                  //
     PWM toggle low/high
190 EPwm1Regs.AQCTLA.bit.CAU = AQ_CLEAR;
```

```
191
192 EALLOW;
193 EPwm1Regs.HRCNFG.all = 0x0;
194 EPwm1Regs.HRCNFG.bit.EDGMODE = HR_FEP;          //MEP
       control on Rising edge
195 EPwm1Regs.HRCNFG.bit.CTLMODE = HR_CMP;
196 EPwm1Regs.HRCNFG.bit.HRLOAD  = HR_CTR_ZERO;
197
198 EDIS;
199 }
200
201 void HRPWM2_Config(Uint16 period)
202 {
203 // ePWM2 register configuration with HRPWM
204 // ePWM2A toggle low/high with MEP control on Rising
       edge
205
206 EPwm2Regs.TBCTL.bit.PRDLD = TB_IMMEDIATE;          //
       set Immediate load
207 EPwm2Regs.TBPRD = period-1;                        //
      PWM frequency = 1 / period
208 EPwm2Regs.CMPA.half.CMPA = period *D+2;             //
       set duty 50% initially
209 EPwm2Regs.CMPA.half.CMPAHR = (1 << 8);             //
       initialize HRPWM extension
210 //  EPwm2Regs.TBPHS.all = 0;
211 EPwm2Regs.TBPHS.half.TBPHS = delay2;           //
       Time-Base Phase Register, master's phase = 0
212 EPwm2Regs.TBCTR = 0;
213
214 EPwm2Regs.TBCTL.bit.CTRMODE = TB_COUNT_UP;
215 EPwm2Regs.TBCTL.bit.PHSEN = TB_ENABLE;             //
       EPwm2 is the Slave
216 EPwm2Regs.TBCTL.bit.SYNCOSEL = TB_SYNC_IN;       //was
       sync in
217 EPwm2Regs.TBCTL.bit.HSPCLKDIV = TB_DIV1;
```

```
218 EPwm2Regs.TBCTL.bit.CLKDIV = TB_DIV1;
219
220 EPwm2Regs.CMPCTL.bit.LOADAMODE = CC_CTR_ZERO;
221 EPwm2Regs.CMPCTL.bit.SHDWAMODE = CC_SHADOW;
222
223 EPwm2Regs.AQCTLA.bit.ZRO = AQ_CLEAR;
      // PWM toggle low/high
224 EPwm2Regs.AQCTLA.bit.CAU = AQ_SET;
225
226 EALLOW;
227 EPwm2Regs.HRCNFG.all = 0x0;
228 EPwm2Regs.HRCNFG.bit.EDGMODE = HR_FEP;
      //MEP control on Falling edge
229 EPwm2Regs.HRCNFG.bit.CTLMODE = HR_CMP;
230 EPwm2Regs.HRCNFG.bit.HRLOAD  = HR_CTR_ZERO;
231
232 EDIS;
233 }
234
235 void HRPWM3_Config(Uint16 period)
236 {
237 // ePWM3 register configuration with HRPWM
238 // ePWM3A toggle high/low with MEP control on falling
      edge
239
240 EPwm3Regs.TBCTL.bit.PRDLD = TB_IMMEDIATE;          //
      set Immediate load
241 EPwm3Regs.TBPRD = period-1;                        //
     PWM frequency = 1 / period
242 EPwm3Regs.CMPA.half.CMPA = period * (1-DP1SP);
                    // set duty 50% initially
243 EPwm3Regs.CMPA.half.CMPAHR = (1 << 8);              //
      initialize HRPWM extension
244 //  EPwm3Regs.TBPHS.all = 0;
245 EPwm3Regs.TBPHS.half.TBPHS = delay3;     // found
      iteratively, confirmed experimentally.
```

```
246 EPwm3Regs.TBCTR = 0;
247
248 EPwm3Regs.TBCTL.bit.CTRMODE = TB_COUNT_UP;
249 EPwm3Regs.TBCTL.bit.PHSEN = TB_ENABLE;                    //
      EPwm3 is the Slave
250 EPwm3Regs.TBCTL.bit.SYNCOSEL = TB_SYNC_IN;
251 EPwm3Regs.TBCTL.bit.HSPCLKDIV = TB_DIV1;
252 EPwm3Regs.TBCTL.bit.CLKDIV = TB_DIV1;
253
254 EPwm3Regs.CMPCTL.bit.LOADAMODE = CC_CTR_ZERO;
255 EPwm3Regs.CMPCTL.bit.SHDWAMODE = CC_SHADOW;
256
257 EPwm3Regs.AQCTLA.bit.ZRO = AQ_CLEAR;
      // PWM toggle high/low
258 EPwm3Regs.AQCTLA.bit.CAU = AQ_SET;
259
260 EALLOW;
261 EPwm3Regs.HRCNFG.all = 0x0;
262 EPwm3Regs.HRCNFG.bit.EDGMODE = HR_FEP;
      //MEP control on falling edge
263 EPwm3Regs.HRCNFG.bit.CTLMODE = HR_CMP;
264 EPwm3Regs.HRCNFG.bit.HRLOAD  = HR_CTR_ZERO;
265 EDIS;
266 }
267
268 void HRPWM4_Config(Uint16 period)
269 {
270 // ePWM1 register configuration with HRPWM
271 // ePWM1A toggle low/high with MEP control on Rising
      edge
272
273 EPwm4Regs.TBCTL.bit.PRDLD = TB_IMMEDIATE;                 //
      set Immediate load
274 EPwm4Regs.TBPRD = period-1;                               //
      PWM frequency = 1 / period
275 EPwm4Regs.CMPA.half.CMPA = period * D-2;
```

```
    // set duty 50% initially
276 EPwm4Regs.CMPA.half.CMPAHR = (1 << 8);               //
    initialize HRPWM extension
277 //  EPwm4Regs.TBPHS.all = 0;
278 EPwm4Regs.TBPHS.half.TBPHS =delay4;      // Time-Base
    Phase Register, slave's phase = phi
279 EPwm4Regs.TBCTR = 0;
280
281 EPwm4Regs.TBCTL.bit.CTRMODE = TB_COUNT_UP;
282 EPwm4Regs.TBCTL.bit.PHSEN = TB_ENABLE;             //
    EPwm1 is the Master
283 EPwm4Regs.TBCTL.bit.SYNCOSEL = TB_SYNC_IN;
284 EPwm4Regs.TBCTL.bit.HSPCLKDIV = TB_DIV1;
285 EPwm4Regs.TBCTL.bit.CLKDIV = TB_DIV1;
286
287 EPwm4Regs.CMPCTL.bit.LOADAMODE = CC_CTR_ZERO;
288 EPwm4Regs.CMPCTL.bit.SHDWAMODE = CC_SHADOW;
289
290 EPwm4Regs.AQCTLA.bit.ZRO = AQ_SET;                  //
    PWM toggle low/high
291 EPwm4Regs.AQCTLA.bit.CAU = AQ_CLEAR;
292
293 EALLOW;
294 EPwm4Regs.HRCNFG.all = 0x0;
295 EPwm4Regs.HRCNFG.bit.EDGMODE = HR_FEP;          //MEP
    control on Rising edge
296 EPwm4Regs.HRCNFG.bit.CTLMODE = HR_CMP;
297 EPwm4Regs.HRCNFG.bit.HRLOAD  = HR_CTR_ZERO;
298 EDIS;
299 }
300
301 void HRPWM5_Config(Uint16 period)
302 {
303 // ePWM2 register configuration with HRPWM
304 // ePWM2A toggle low/high with MEP control on Rising
    edge
```

```
305
306 EPwm5Regs.TBCTL.bit.PRDLD = TB_IMMEDIATE;           //
       set Immediate load
307 EPwm5Regs.TBPRD = period-1;                         //
      PWM frequency = 1 / period
308 EPwm5Regs.CMPA.half.CMPA = period *D+2;
       // set duty 50% initially
309 EPwm5Regs.CMPA.half.CMPAHR = (1 << 8);               //
       initialize HRPWM extension
310 //  EPwm5Regs.TBPHS.all = 0;
311 EPwm5Regs.TBPHS.half.TBPHS = delay5;       // Time-Base
       Phase Register, slave's phase = phi
312 EPwm5Regs.TBCTR = 0;
313
314 EPwm5Regs.TBCTL.bit.CTRMODE = TB_COUNT_UP;
315 EPwm5Regs.TBCTL.bit.PHSEN = TB_ENABLE;               //
       EPwm2 is the Master
316 EPwm5Regs.TBCTL.bit.SYNCOSEL = TB_SYNC_IN;
317 EPwm5Regs.TBCTL.bit.HSPCLKDIV = TB_DIV1;
318 EPwm5Regs.TBCTL.bit.CLKDIV = TB_DIV1;
319
320 EPwm5Regs.CMPCTL.bit.LOADAMODE = CC_CTR_ZERO;
321 EPwm5Regs.CMPCTL.bit.SHDWAMODE = CC_SHADOW;
322
323 EPwm5Regs.AQCTLA.bit.ZRO = AQ_CLEAR;
       // PWM toggle low/high
324 EPwm5Regs.AQCTLA.bit.CAU = AQ_SET;
325
326 EALLOW;
327 EPwm5Regs.HRCNFG.all = 0x0;
328 EPwm5Regs.HRCNFG.bit.EDGMODE = HR_FEP;
       //MEP control on Falling edge
329 EPwm5Regs.HRCNFG.bit.CTLMODE = HR_CMP;
330 EPwm5Regs.HRCNFG.bit.HRLOAD  = HR_CTR_ZERO;
331
332 EDIS;
```

```
333 }
334
335 void HRPWM6_Config(Uint16 period)
336 {
337 // ePWM4 register configuration with HRPWM
338 // ePWM4A toggle high/low with MEP control on falling
      edge
339
340 EPwm6Regs.TBCTL.bit.PRDLD = TB_IMMEDIATE;           //
      set Immediate load
341 EPwm6Regs.TBPRD = period-1;                          //
     PWM frequency = 1 / period
342 EPwm6Regs.CMPA.half.CMPA = period * DP2SP;
                    // set duty
343 EPwm6Regs.CMPA.half.CMPAHR = (1 << 8);               //
      initialize HRPWM extension
344 //  EPwm6Regs.TBPHS.all = 0;
345 EPwm6Regs.TBPHS.half.TBPHS = period * (1 - D + DP2SP)
      + delay6;      // Time-Base Phase Register, slave's
      phase = phi
346 EPwm6Regs.TBCTR = 0;
347
348 EPwm6Regs.TBCTL.bit.CTRMODE = TB_COUNT_UP;
349 EPwm6Regs.TBCTL.bit.PHSEN = TB_ENABLE;              //
      EPwm4 is the Master
350 EPwm6Regs.TBCTL.bit.SYNCOSEL = TB_SYNC_IN;
351 EPwm6Regs.TBCTL.bit.HSPCLKDIV = TB_DIV1;
352 EPwm6Regs.TBCTL.bit.CLKDIV = TB_DIV1;
353
354 EPwm6Regs.CMPCTL.bit.LOADAMODE = CC_CTR_ZERO;
355 EPwm6Regs.CMPCTL.bit.SHDWAMODE = CC_SHADOW;
356
357 EPwm6Regs.AQCTLA.bit.ZRO = AQ_SET;                   //
      PWM toggle
358 EPwm6Regs.AQCTLA.bit.CAU = AQ_CLEAR;
359
```

```
360 EALLOW;
361 EPwm6Regs.HRCNFG.all = 0x0;
362 EPwm6Regs.HRCNFG.bit.EDGMODE = HR_FEP;
        //MEP control on falling edge
363 EPwm6Regs.HRCNFG.bit.CTLMODE = HR_CMP;
364 EPwm6Regs.HRCNFG.bit.HRLOAD  = HR_CTR_ZERO;
365 EDIS;
366 }
```

APPENDIX B

ISOLATED BATTERY CHARGER

B.1 Schematics

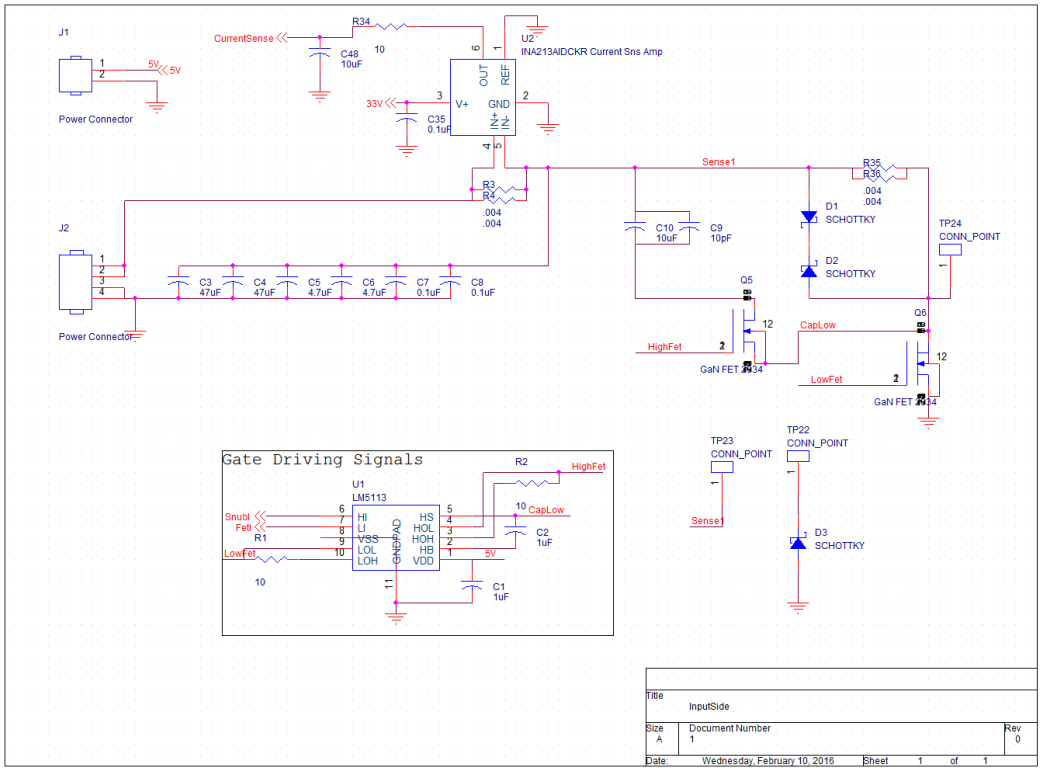

Figure B.1: Schematic of the non-isolated side of the battery charger.

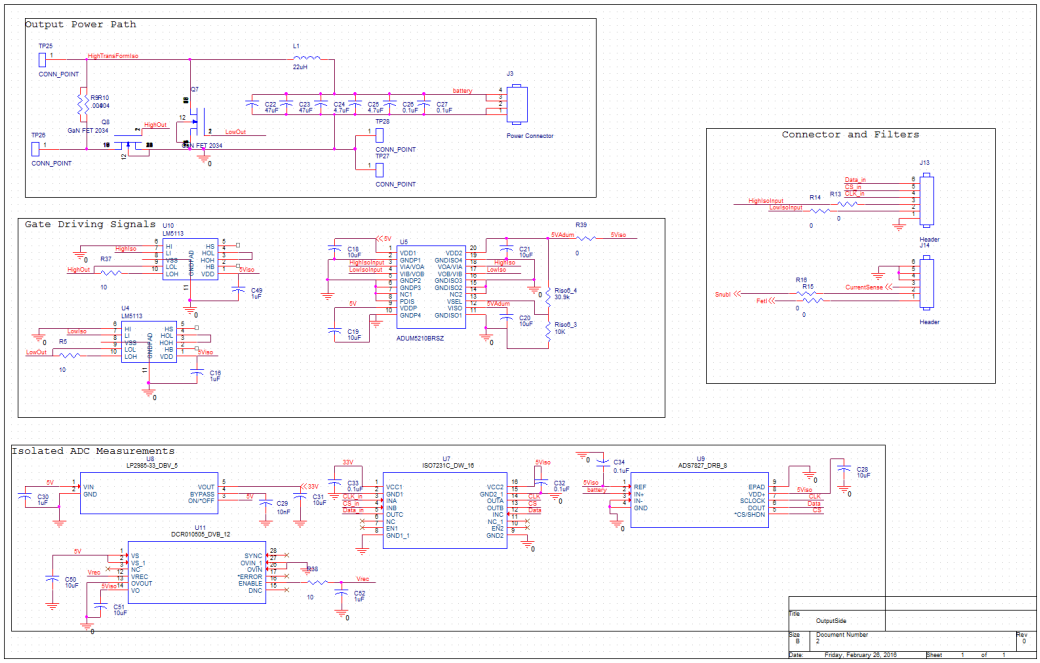

Figure B.2: Schematic of the isolated side of the battery charger.

B.2 PCB Layout

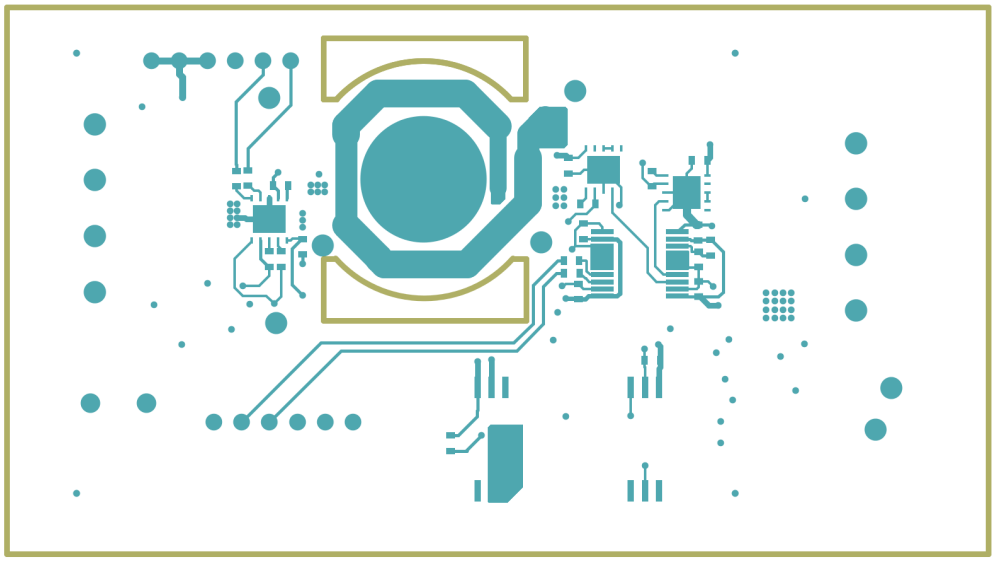

Figure B.3: Bottom copper plane of the isolated battery charger.

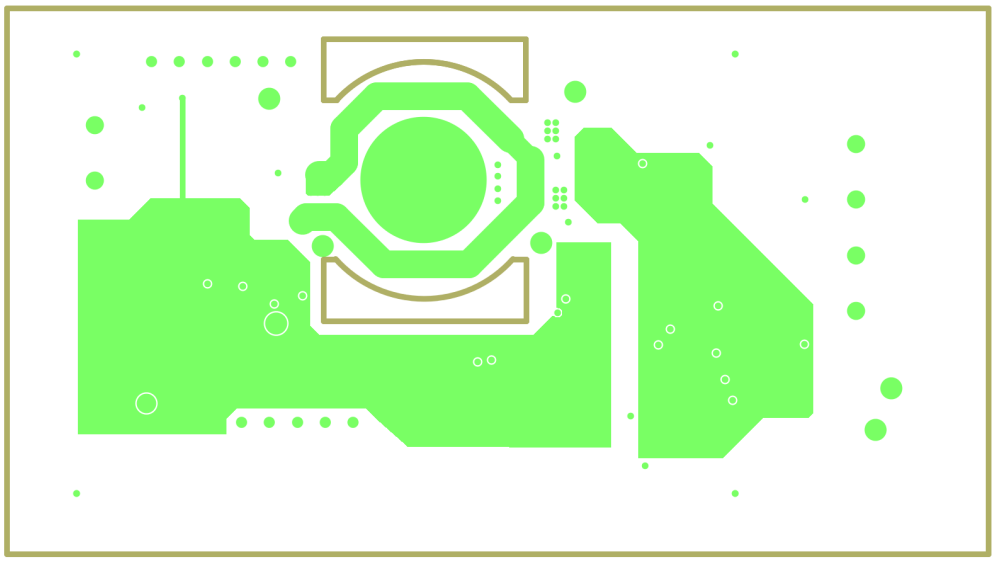

Figure B.4: Top copper plane of the isolated battery charger.

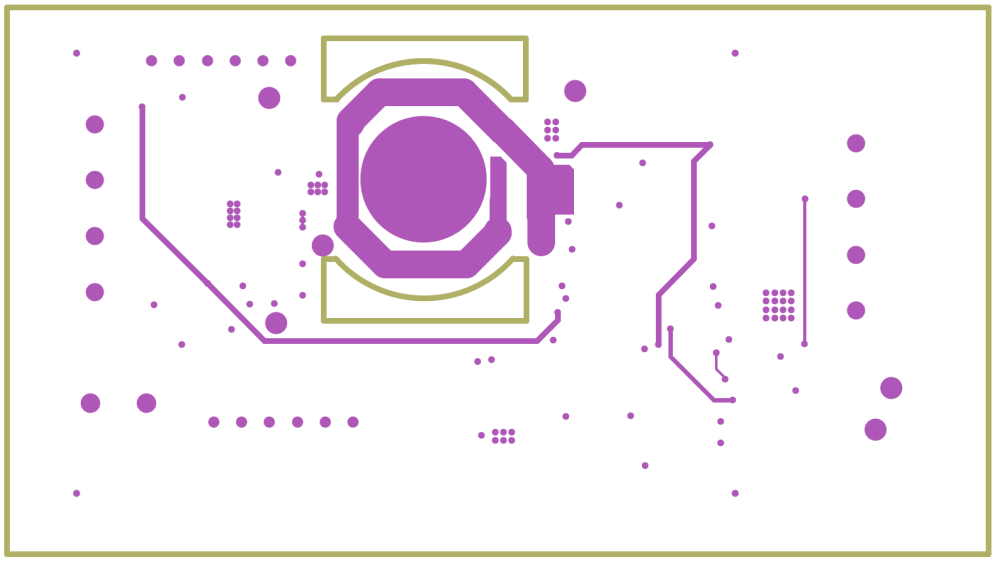

Figure B.5: Ground plane of the isolated battery charger.

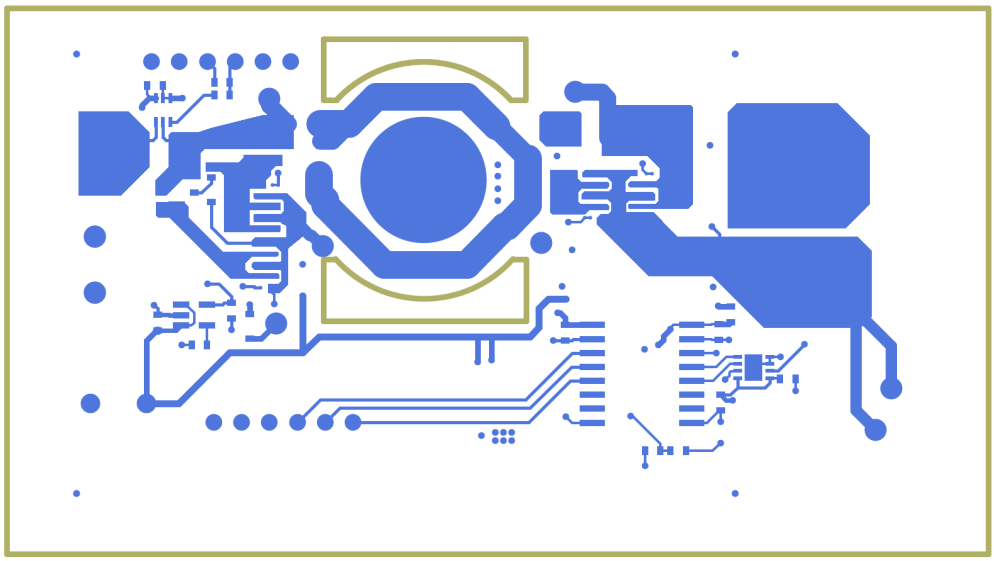

Figure B.6: Signal layer of the isolated battery charger.

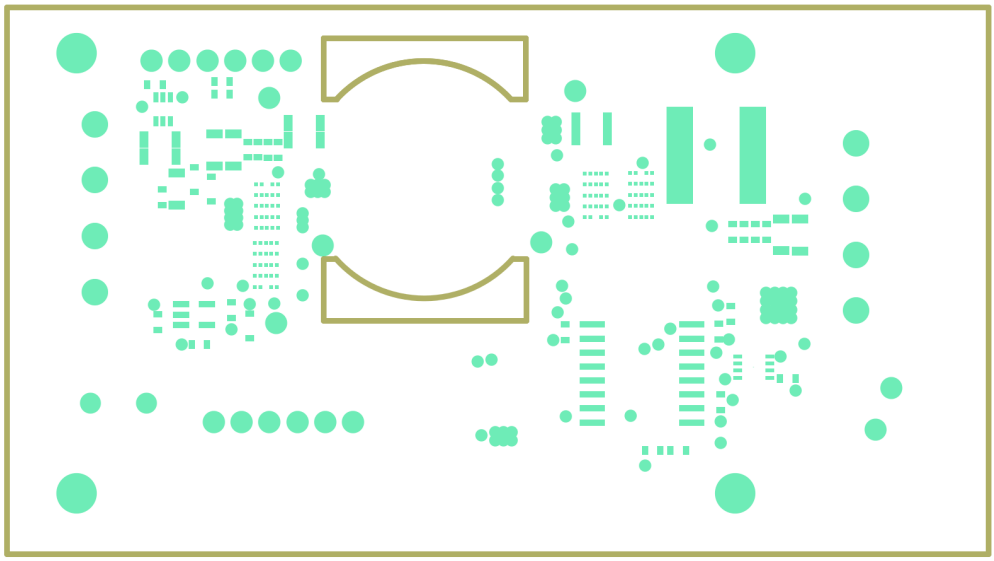

Figure B.7: Top solder mask layer of the isolated battery charger.

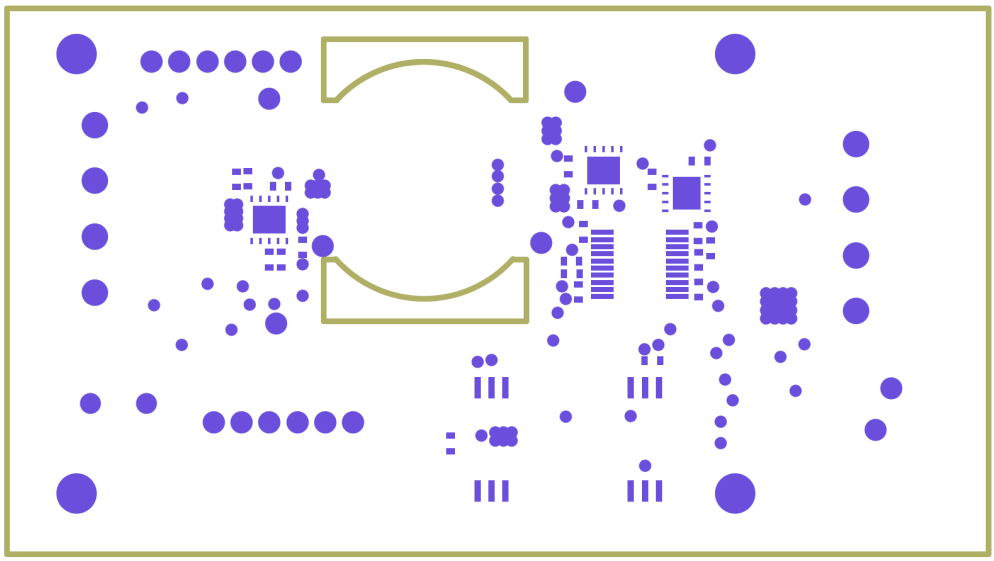

Figure B.8: Bottom solder mask layer of the isolated battery charger.

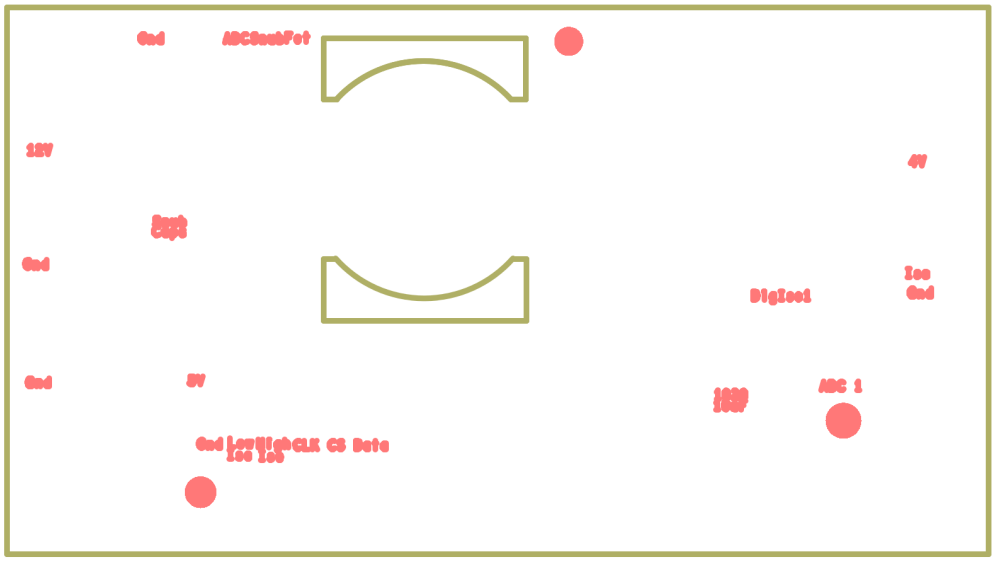

Figure B.9: Silk screen of the isolated battery charger.

B.3 Micro-controller Code

```
1 //###################################
2 //Derek Heeger
3 //Advisor: Robert Pilawa Podgurski
4 //Battery Charging Code
5 //Design based heavily off of Texas Instrument's
      examples from code composer studio
6 //###################################
7
8 #include "DSP28x_Project.h"     // Device Headerfile
      and Examples Include File
9 //#include "F2806x_Device.h"         // F2806x
      Headerfile
10 #include "F2806x_Examples.h"       // F2806x Examples
      Headerfile
11 #include "F2806x_EPwm_defines.h"   // useful defines
      for initialization
12 // Prototype statements for functions found within
      this file.
13 __interrupt void adc_isr(void);
14 //void Adc_Config(void);
15 __interrupt void cpu_timer1_isr(void);
16 void HRPWM1_Config(Uint16);
17 void HRPWM2_Config(Uint16);
18 void spi_xmit(Uint16 a);
19 void spi_fifo_init(void);
20 void spi_init(void);
21 void disablePWM(void);
22 void enablePWM(void);
23 Uint16 getCellVoltage(void);
24
25 // Global variables used in this example:
26 Uint16 LoopCount, sampleNumber, sampleNumberBits;
27 Uint16 ConversionCount;
28
```

```
29 Uint16 Voltage1[1024];
30 Uint16 Voltage2[1024];
31
32 Uint16 i,j, DutyFine, n,update, period, D, Dmin, Dmax,
     Vmax, VmaxBinary, difference, control,
     VoltageBinary, pwmEnabled, lastCurrent, current,
     Icontrol, Iset;
33 Uint64 tempDutyCurrent;
34 Uint64 dutyCurrent;
35 Uint64 currentTemp;
36
37 float vout, Iout, Dreal;
38
39
40
41 main()
42 {
43
44 // Step 1. Initialize System Control:
45 // PLL, WatchDog, enable Peripheral Clocks
46 // This example function is found in the
     F2806x_SysCtrl.c file.
47 InitSysCtrl();
48
49
50 // Step 2. Initialize GPIO:
51 // This example function is found in the F2806x_Gpio.c
     file and
52 // illustrates how to set the GPIO to it's default
     state.
53 // InitGpio();  // Skipped for this example
54
55 // Step 3. Clear all interrupts and initialize PIE
     vector table:
56 // Disable CPU interrupts
57 DINT;
```

```
58
59 // Initialize the PIE control registers to their
       default state.
60 // The default state is all PIE interrupts disabled
       and flags
61 // are cleared.
62 // This function is found in the F2806x_PieCtrl.c file.
63 InitPieCtrl();
64
65 // Disable CPU interrupts and clear all CPU interrupt
       flags:
66 IER = 0x0000;
67 IFR = 0x0000;
68
69 // Initialize the PIE vector table with pointers to
       the shell Interrupt
70 // Service Routines (ISR).
71 // This will populate the entire table, even if the
       interrupt
72 // is not used in this example.  This is useful for
       debug purposes.
73 // The shell ISR routines are found in
       F2806x_DefaultIsr.c.
74 // This function is found in F2806x_PieVect.c.
75 InitPieVectTable();
76
77 // Interrupts that are used in this example are
       re-mapped to
78 // ISR functions found within this file.
79 EALLOW;  // This is needed to write to EALLOW
       protected register
80 PieVectTable.ADCINT1 = &adc_isr;
81 PieVectTable.TINT1 = &cpu_timer1_isr;
82 EDIS;    // This is needed to disable write to EALLOW
       protected registers
83
```

```
84 // Step 4. Initialize all the Device Peripherals:
85 // This function is found in F2806x_InitPeripherals.c
86 // InitPeripherals(); // Not required for this example
87 InitAdc();  // For this example, init the ADC
88 AdcOffsetSelfCal();
89
90
91
92 // Step 5. User specific code, enable interrupts:
93
94 // Enable ADCINT1 in PIE
95 PieCtrlRegs.PIEIER1.bit.INTx1 = 1;  // Enable INT 1.1
       in the PIE
96 IER |= M_INT1;              // Enable CPU Interrupt 1
97 IER |= M_INT13;
98 EINT;                       // Enable Global interrupt
       INTM
99 ERTM;                       // Enable Global realtime
       interrupt DBGM
100
101 LoopCount = 0;
102 lastCurrent=0;
103 ConversionCount = 0;
104
105 // Configure ADC
106 EALLOW;
107 AdcRegs.ADCCTL2.bit.ADCNONOVERLAP = 1;  // Enable
       non-overlap mode
108 AdcRegs.ADCCTL1.bit.INTPULSEPOS = 1;  // ADCINT1 trips
       after AdcResults latch
109 AdcRegs.INTSEL1N2.bit.INT1E     = 1;  // Enabled
       ADCINT1
110 AdcRegs.INTSEL1N2.bit.INT1CONT  = 0;  // Disable
       ADCINT1 Continuous mode
111 AdcRegs.INTSEL1N2.bit.INT1SEL   = 1;    // setup EOC1
       to trigger ADCINT1 to fire
```

```
112 AdcRegs.ADCSOC0CTL.bit.CHSEL  = 4;     // set SOC0
      channel select to ADCINA4
113 AdcRegs.ADCSOC1CTL.bit.CHSEL  = 2;     // set SOC1
      channel select to ADCINA2
114 AdcRegs.ADCSOC0CTL.bit.TRIGSEL  = 5;     // set SOC0
      start trigger on EPWM1A, due to round-robin SOC0
      converts first then SOC1
115 AdcRegs.ADCSOC1CTL.bit.TRIGSEL  = 5;     // set SOC1
      start trigger on EPWM1A, due to round-robin SOC0
      converts first then SOC1
116 AdcRegs.ADCSOC0CTL.bit.ACQPS  = 10; // set SOC0 S/H
      Window to 7 ADC Clock Cycles, (6 ACQPS plus 1)
117 AdcRegs.ADCSOC1CTL.bit.ACQPS  = 10; // set SOC1 S/H
      Window to 7 ADC Clock Cycles, (6 ACQPS plus 1)
118 EDIS;
119
120 // Assumes ePWM1 clock is already enabled in
      InitSysCtrl();
121 EPwm1Regs.ETSEL.bit.SOCAEN  = 1;     // Enable SOC on A
      group
122 EPwm1Regs.ETSEL.bit.SOCASEL = 1;     // Select SOC from
      CMPA on upcount  was 4 in example
123 EPwm1Regs.ETPS.bit.SOCAPRD  = 1;     // Generate pulse
      on 1st event
124 /*EPwm1Regs.CMPA.half.CMPA  = 0x0080; // Set compare A
      value
125 EPwm1Regs.TBPRD          = 0xFFFF; // Set period for
      ePWM1
126 EPwm1Regs.TBCTL.bit.CTRMODE  = 0;     // count up and
      start
127 */
128 period = 90; //2MHZ is 45
129 D=period*3/4;
130 Vmax = 3;
131 Iset=250;//~1A
132 //VmaxBinary = 4095*Vmax/5;
```

```
133 VmaxBinary = 2866;//3.5 volts out
134 Dmin =period*.9;
135 Dmax = period*.5;
136 control=0;
137 Icontrol=0;
138 sampleNumber=1024;
139 sampleNumberBits=10;
140
141
142 InitEPwm1Gpio();
143 InitEPwm2Gpio();
144 HRPWM1_Config(period);        // ePWM1 target, Period =
      10
145 HRPWM2_Config(period);        // ePWM2 target, Period =
      20
146 pwmEnabled=0;
147
148 EALLOW;
149 GpioCtrlRegs.GPBMUX1.bit.GPIO34 = 0;
150 GpioCtrlRegs.GPBDIR.bit.GPIO34 = 1;
151
152 GpioCtrlRegs.GPBMUX1.bit.GPIO39 = 0;
153 GpioCtrlRegs.GPBDIR.bit.GPIO39 = 1;
154 EDIS;
155
156 GpioDataRegs.GPBSET.bit.GPIO34=1;
157 GpioDataRegs.GPBSET.bit.GPIO39=0;
158
159 InitSpiaGpio();
160
161 spi_fifo_init();     // Initialize the Spi FIFO
162 spi_init();      // init SPI
163
164
165 //configure 1 second interrupt
166
```

```
167 InitCpuTimers();   // For this example, only
      initialize the Cpu Timers
168 // Configure CPU-Timer 0 to interrupt every 500
      milliseconds:
169 // 80MHz CPU Freq, 50 millisecond Period (in uSeconds)
170 ConfigCpuTimer(&CpuTimer1, 80, 500000);
171 CpuTimer1Regs.TCR.all = 0x4001; // Use write-only
      instruction to set TSS bit = 0
172 //      ConfigCpuTimer(&CpuTimer1, 80, 500000);
173
174 enablePWM();
175
176 // Wait for ADC interrupt
177 for(;;)
178 {
179 // VoltageBinary = getCellVoltage();
180 }
181
182 }
183
184
185 __interrupt void  adc_isr(void)
186 {
187
188 Voltage1[ConversionCount] = AdcResult.ADCRESULT0;
189 Voltage2[ConversionCount] = AdcResult.ADCRESULT1;
190 int counter;
191 // If 20 conversions have been logged, start over
192 if(ConversionCount >= sampleNumber-1)
193 {
194 currentTemp=0;
195 ConversionCount = 0;
196 for (counter=1; counter<sampleNumber-1; counter++){
197 currentTemp=currentTemp+Voltage2[counter];
198 }
199 currentTemp= currentTemp>>sampleNumberBits;
```

```
200
201 if(currentTemp>4095){
202 current=lastCurrent;
203 }
204 else{
205 current=currentTemp;
206 lastCurrent=current;
207 }
208
209 Dreal = (100-100*D/period);
210 tempDutyCurrent = current*10;
211 dutyCurrent = tempDutyCurrent/Dreal;
212
213 if (Icontrol==1){
214 if(vout>2.5){
215 difference = abs(dutyCurrent-Iset);
216 if (difference<50)
217 D=D;
218 else if(dutyCurrent>Iset && D<Dmin)
219 D=D+1;
220 else if (dutyCurrent<Iset && D>Dmax)
221 D=D-1;
222 else
223 D=D;
224
225 EPwm1Regs.CMPA.half.CMPA = D;
226 EPwm2Regs.CMPA.half.CMPA = D;                    // set duty
      50% initially
227 }
228 else{
229
230 D=period*3/4;
231
232 }
233 }
234
```

```
235 Iout = dutyCurrent*.0074-0.1786;
236
237 }
238 else ConversionCount++;
239
240 AdcRegs.ADCINTFLGCLR.bit.ADCINT1 = 1;    //Clear
      ADCINT1 flag reinitialize for next SOC
241 PieCtrlRegs.PIEACK.all = PIEACK_GROUP1;    //
      Acknowledge interrupt to PIE
242
243 return;
244 }
245
246 __interrupt void cpu_timer1_isr(void)
247 {
248
249 CpuTimer1.InterruptCount++;
250 GpioDataRegs.GPBTOGGLE.bit.GPIO34 = 1; // Toggle
      GPIO34 once per 500 milliseconds
251 GpioDataRegs.GPBTOGGLE.bit.GPIO39 = 1;
252 // Acknowledge this interrupt to receive more
      interrupts from group 1
253 VoltageBinary = getCellVoltage();
254 vout=VoltageBinary*.00122; //.01953 is 5/4090 which is
      the scaling value to convert bits to real voltage
255
256 if (control==1){
257 difference = abs(VoltageBinary-VmaxBinary);
258 if (difference<60)
259 D=D;
260 else if(VoltageBinary>VmaxBinary && D<Dmin)
261 D=D+1;
262 else if (VoltageBinary<VmaxBinary && D>Dmax)
263 D=D-1;
264 else
265 D=D;
```

```
266
267 EPwm1Regs.CMPA.half.CMPA = D;
268 EPwm2Regs.CMPA.half.CMPA = D;                    // set duty
      50% initially
269 }
270
271 if(vout>4.8){
272 // disablePWM();
273 }
274
275
276 PieCtrlRegs.PIEACK.all = PIEACK_GROUP1;
277 }
278
279 void HRPWM1_Config(Uint16 period)
280 {
281 // ePWM1 register configuration with HRPWM
282 // ePWM1A toggle low/high with MEP control on Rising
      edge
283 unsigned int deadtime = 1;
284
285 EPwm1Regs.TBCTL.bit.PRDLD = TB_IMMEDIATE;          //
      set Immediate load
286 EPwm1Regs.TBPRD = period-1;                        //
     PWM frequency = 1 / period
287 EPwm1Regs.CMPA.half.CMPA = D;                    // set duty
      50% initially
288 EPwm1Regs.CMPA.half.CMPAHR = (1 << 8);              //
      initialize HRPWM extension
289 //EPwm1Regs.CMPB = period / 2;                     //
      set duty 50% initially
290 EPwm1Regs.TBPHS.all = 0;
291 EPwm1Regs.TBCTR = 0;
292
293 EPwm1Regs.TBCTL.bit.CTRMODE = TB_COUNT_UP;
294 EPwm1Regs.TBCTL.bit.PHSEN = TB_DISABLE;          //
```

```
      EPwm1 is the Master
295 EPwm1Regs.TBCTL.bit.SYNCOSEL = TB_CTR_ZERO;
296 EPwm1Regs.TBCTL.bit.HSPCLKDIV = TB_DIV1;
297 EPwm1Regs.TBCTL.bit.CLKDIV = TB_DIV1;
298
299 EPwm1Regs.CMPCTL.bit.LOADAMODE = CC_CTR_ZERO;
300 EPwm1Regs.CMPCTL.bit.LOADBMODE = CC_CTR_ZERO;
301 EPwm1Regs.CMPCTL.bit.SHDWAMODE = CC_SHADOW;
302 EPwm1Regs.CMPCTL.bit.SHDWBMODE = CC_SHADOW;
303
304 //sets dead time and inverts EPWM1B
305 EPwm1Regs.DBCTL.bit.OUT_MODE = DB_FULL_ENABLE;
306 EPwm1Regs.DBCTL.bit.POLSEL = 0;
307 EPwm1Regs.DBCTL.bit.IN_MODE = DBA_ALL;
308 EPwm1Regs.DBRED = deadtime;
309 EPwm1Regs.DBFED = deadtime;
310
311
312 EPwm1Regs.AQCTLA.bit.ZRO = AQ_CLEAR;                    //
      PWM toggle low/high
313 EPwm1Regs.AQCTLA.bit.CAU = AQ_CLEAR;
314 EPwm1Regs.AQCTLB.bit.ZRO = AQ_CLEAR;
315 EPwm1Regs.AQCTLB.bit.CBU = AQ_CLEAR;
316
317 EALLOW;
318 EPwm1Regs.HRCNFG.all = 0x0;
319 EPwm1Regs.HRCNFG.bit.EDGMODE = HR_REP;          //MEP
      control on Rising edge
320 EPwm1Regs.HRCNFG.bit.CTLMODE = HR_CMP;
321 EPwm1Regs.HRCNFG.bit.HRLOAD  = HR_CTR_ZERO;
322 EDIS;
323
324
325 }
326
327 void HRPWM2_Config(Uint16 period)
```

```
328 {
329 // ePWM2 register configuration with HRPWM
330 // ePWM2A toggle low/high with MEP control on Rising
      edge
331
332
333 unsigned int deadtime = 1;
334 unsigned int phaseshift= 0;
335 EPwm2Regs.TBCTL.bit.PRDLD = TB_IMMEDIATE;           //
      set Immediate load
336 EPwm2Regs.TBPRD = period-1;         // PWM frequency = 1
      / period
337 EPwm2Regs.CMPA.half.CMPA = D;                  // set duty
      50% initially
338 EPwm2Regs.CMPA.half.CMPAHR = (1 << 8);               //
      initialize HRPWM extension
339 //EPwm1Regs.CMPB = period / 2;                       //
      set duty 50% initially
340 EPwm2Regs.TBPHS.half.TBPHS = phaseshift;
341 EPwm2Regs.TBCTR = 0;
342
343 EPwm2Regs.TBCTL.bit.CTRMODE = TB_COUNT_UP;
344 EPwm2Regs.TBCTL.bit.PHSEN = TB_ENABLE;             //
      EPwm1 is the Master
345 //EPwm2Regs.TBCTL.bit.PHSEN = TB_DISABLE;
346 EPwm2Regs.TBCTL.bit.SYNCOSEL = TB_SYNC_IN;
347 //EPwm2Regs.TBCTL.bit.SYNCOSEL = TB_SYNC_DISABLE;
348 EPwm2Regs.TBCTL.bit.HSPCLKDIV = TB_DIV1;
349 EPwm2Regs.TBCTL.bit.CLKDIV = TB_DIV1;
350
351 EPwm2Regs.CMPCTL.bit.LOADAMODE = CC_CTR_ZERO;
352 EPwm2Regs.CMPCTL.bit.LOADBMODE = CC_CTR_ZERO;
353 EPwm2Regs.CMPCTL.bit.SHDWAMODE = CC_SHADOW;
354 EPwm2Regs.CMPCTL.bit.SHDWBMODE = CC_SHADOW;
355
356 //sets dead time and inverts EPWM1B
```

```
357 EPwm2Regs.DBCTL.bit.OUT_MODE = DB_FULL_ENABLE;
358 EPwm2Regs.DBCTL.bit.POLSEL = 0;
359 EPwm2Regs.DBCTL.bit.IN_MODE = DBA_ALL;
360 EPwm2Regs.DBRED = deadtime;
361 EPwm2Regs.DBFED = deadtime;
362
363 EPwm2Regs.AQCTLA.bit.ZRO = AQ_CLEAR;                    //
      PWM toggle low/high
364 EPwm2Regs.AQCTLA.bit.CAU = AQ_CLEAR;
365 EPwm2Regs.AQCTLB.bit.ZRO = AQ_CLEAR;
366 EPwm2Regs.AQCTLB.bit.CBU = AQ_CLEAR;
367
368 EALLOW;
369 EPwm2Regs.HRCNFG.all = 0x0;
370 EPwm2Regs.HRCNFG.bit.EDGMODE = HR_REP;          //MEP
      control on Rising edge
371 EPwm2Regs.HRCNFG.bit.CTLMODE = HR_CMP;
372 EPwm2Regs.HRCNFG.bit.HRLOAD  = HR_CTR_ZERO;
373 EDIS;
374
375 }
376
377 Uint16 getCellVoltage(void){
378 Uint16 f8, rdata, sdata;
379 // Transmit blank data to trigger external ADC response
380 //SDI does not need to be connected to ADC. Just need
      CS and SCLK
381 sdata = 0x0000;
382 spi_xmit(sdata);
383 // Wait until data is received
384 while(SpiaRegs.SPIFFRX.bit.RXFFST !=1) { }
385 rdata = SpiaRegs.SPIRXBUF;
386
387 //pull the relevant 8 bits from the ADC
388 f8 = 4095&rdata;
389
```

```
390 return f8;
391
392 }
393
394 void spi_init()
395 {
396 SpiaRegs.SPICCR.all =0x000F;                 // Reset
      on, rising edge, 16-bit char bits
397 SpiaRegs.SPICTL.all =0x0006;                // Enable
      master mode, normal phase,
398 // enable talk, and SPI int disabled.
399 SpiaRegs.SPIBRR =0x007F;
400 SpiaRegs.SPICCR.all =0x009F;                // Relinquish
      SPI from Reset
401 SpiaRegs.SPICCR.bit.SPICHAR=14;
402 SpiaRegs.SPIPRI.bit.FREE = 1;                  // Set so
      breakpoints don't disturb xmission
403 SpiaRegs.SPICCR.bit.SPILBK=0;
404 }
405
406 void spi_xmit(Uint16 a)
407 {
408 SpiaRegs.SPITXBUF=a;
409 }
410
411 void spi_fifo_init()
412 {
413 // Initialize SPI FIFO registers
414 SpiaRegs.SPIFFTX.all=0xE040;
415 SpiaRegs.SPIFFRX.all=0x2044;
416 SpiaRegs.SPIFFCT.all=0x0;
417 }
418
419
420 void disablePWM(){
421 EPwm1Regs.AQCTLA.bit.ZRO = AQ_CLEAR;                //
```

```
      PWM toggle low/high
422 EPwm1Regs.AQCTLA.bit.CAU = AQ_CLEAR;
423 EPwm1Regs.AQCTLB.bit.ZRO = AQ_CLEAR;
424 EPwm1Regs.AQCTLB.bit.CBU = AQ_CLEAR;
425 EPwm1Regs.DBCTL.bit.POLSEL = 0;
426
427 EPwm2Regs.AQCTLA.bit.ZRO = AQ_CLEAR;                    //
      PWM toggle low/high
428 EPwm2Regs.AQCTLA.bit.CAU = AQ_CLEAR;
429 EPwm2Regs.AQCTLB.bit.ZRO = AQ_CLEAR;
430 EPwm2Regs.AQCTLB.bit.CBU = AQ_CLEAR;
431 EPwm2Regs.DBCTL.bit.POLSEL = 0;
432
433 pwmEnabled=0;
434 }
435
436 void enablePWM(){
437 EPwm1Regs.DBCTL.bit.POLSEL = DB_ACTV_HIC;
438 EPwm1Regs.AQCTLA.bit.CAU = AQ_SET;
439 EPwm1Regs.AQCTLB.bit.CBU = AQ_SET;
440
441 EPwm2Regs.DBCTL.bit.POLSEL = DB_ACTV_HIC;
442 EPwm2Regs.AQCTLA.bit.CAU = AQ_SET;
443 EPwm2Regs.AQCTLB.bit.CBU = AQ_SET;
444
445 pwmEnabled=1;
446 }
```

REFERENCES

[1] B. Macy, Y. Lei, and R. C. Pilawa-Podgurski, "A 1.2 MHz, 25 V to 100 V GaN-based resonant Dickson switched-capacitor converter with 1011 W/in3 (61.7 kW/L) power density," in *APEC*, 2015.

[2] Y. Lei, R. May, and R. Pilawa-Podgurski, "Split-phase control: Achieving complete soft-charging operation of a Dickson switched-capacitor converter," in *Control and Modeling for Power Electronics (COMPEL), 2014 IEEE 15th Workshop on*, June 2014, pp. 1–7.

[3] W. C. Lee, D. Drury, and P. Mellor, "Comparison of passive cell balancing and active cell balancing for automotive batteries," in *VPPC*, 2011.

[4] S. Yarlagadda, T. Hartley, and I. Husain, "A battery management system using an active charge equalization technique based on a dc/dc converter topology," *IEEE Transactions on Industry Applications*, vol. 49, pp. 2720–2729, 2013.

[5] M. Einhorn, W. Roessler, and J. Fleig, "Improved performance of serially connected li-ion batteries with active cell balancing in electric vehicles," *IEEE Transactions on Vehicular Technology*, vol. 60, pp. 2448–2457, 2011.

[6] S. Cheong, S. Chung, and A. Ioinovici, "Development of power electronics converters based on switched-capacitor circuits," in *Circuits and Systems, 1992. ISCAS '92. Proceedings., 1992 IEEE International Symposium on*, vol. 4, May 1992, pp. 1907–1910.

[7] S. Cheong, H. Chung, and A. Ioinovici, "Inductorless DC-to-DC converter with high power density," *IEEE Transactions on Industrial Electronics*, vol. 41, no. 2, pp. 208–215, Apr. 1994.

[8] J. Kimball and P. Krein, "Analysis and design of switched capacitor converters," in *Applied Power Electronics Conference and Exposition, 2005. APEC 2005. Twentieth Annual IEEE*, vol. 3. IEEE, 2005, pp. 1473–1477.

[9] R. May, "Analysis of soft charging switched capacitor power converters," M.S. thesis, ECE Department, University of Illinois, Urbana Champaign, Aug. 2013.

[10] C. B. Barth, I. Moon, Y. Lei, S. Qin, and R. C. Pilawa-Podgurski, "Experimental evaluation of capacitors for power buffering in single-phase power converters," in *IEEE Energy Conversion Congress and Exposition*, 2015.

[11] S. Qin, C. Barth, W.-C. Liu, and R. C. Pilawa-Podgurski, "A high-efficiency high energy density buffer architecture for power pulsation decoupling in grid-interfaced converters," in *IEEE Energy Conversion Congress and Exposition (ECCE)*, 2015.

[12] M. D. Seeman, "A design methodology for switched-capacitor dc-dc converters," Ph.D. dissertation, EECS Department, University of California, Berkeley, May 2009. [Online]. Available: http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-78.html

[13] Y. Lei, C. B. Barth, S. Qin, W.-C. Liu, I. Moon, A. Stillwell, D. Chou, T. Foulkes, Z. Ye, Z. Liao, and R. C. Pilawa-Podgurski, "A 2 kW, Single-Phase, 7-Level, GaN Inverter with an Active Energy Buffer Achieving 216 $\frac{W}{in^3}$ Power Density and 97.6% Peak Efficiency," in *Applied Power Electronics Conference and Exposition*, 2016.

[14] C.-K. Cheung, S.-C. Tan, C. Tse, and A. Ioinovici, "On energy efficiency of switched-capacitor converters," *Power Electronics, IEEE Transactions on*, vol. 28, no. 2, pp. 862–876, Feb. 2013.

[15] R. Pilawa-Podgurski, D. Giuliano, and D. Perreault, "Merged two-stage power converter architecture with soft-charging switched-capacitor energy transfer," in *Power Electronics Specialists Conference, 2008. PESC 2008. IEEE*, June 2008, pp. 4008–4015.

[16] Y. Lei and R. C. Pilawa-Podgurski, "A General Method for Analyzing Resonant and Soft-Charging Operation of Switched-Capacitor Converters," *IEEE Transactions on Power Electronics*, vol. 30, pp. 5650–5664, Oct 2015.

[17] M. Seeman and S. Sanders, "Analysis and optimization of switched-capacitor dc-dc converters," *Power Electronics, IEEE Transactions on*, vol. 23, no. 2, pp. 841–851, March 2008.

[18] Y. Lei and R. Pilawa-Podgurski, "Analysis of switched-capacitor dc-dc converters in soft-charging operation," in *Control and Modeling for Power Electronics (COMPEL), 2013 IEEE 14th Workshop on*, June 2013, pp. 1–7.

[19] Y. Lei and R. Pilawa-Podgurski, "Soft-charging operation of switched-capacitor dc-dc converters with an inductive load," in *The Applied Power Electronics Conference and Exposition (APEC), 2014 IEEE*, 2014.

[20] A. Ioinovici, C. Tse, and H.-H. Chung, "Comments on "Design and analysis of switched-capacitor-based step-up resonant Converters"," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 53, no. 6, p. 1403, June 2006. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1643446

[21] K. Law, K. Cheng, and Y. Yeung, "Design and analysis of switched-capacitor-based step-up resonant converters," *Circuits and Systems I: Regular Papers, IEEE Transactions on [see also Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on]*, vol. 52, no. 5, pp. 943–948, May 2005.

[22] Y. Lei, Z. Ye, and R. C. Pilawa-Podgurski, "A GaN-based 97% efficient hybrid switched-capacitor converter with lossless regulation capability," in *IEEE Energy Conversion Congress and Exposition*, 2015.

[23] Y. Lei, R. May, and R. C. Pilawa-Podgurski, "Split-phase control: Achieving complete soft-charging operation of a Dickson switched-capacitor converter," *IEEE Transactions on Power Electronics*, vol. 31, pp. 770–782, Jan 2016.

[24] A. Peterchev, J. Xiao, and S. R. Sanders, "Architecture and IC implementation of a digital VRM controller," *IEEE Transactions on Power Electronics*, vol. 18, pp. 356–364, 2003.

[25] *TMS320x2806x Technical Reference Manual*, Texas Instruments, March 2016.

[26] *bq76PL536A 3-to-6 Series Cell Lithium-Ion Battery Monitor and Secondary Protection IC for Applications*, Texas Instruments, January 2016.

[27] *LTC6804-1/LTC6804-2 Multicell Battery Monitors*, Linear Technology, no date.

[28] J. Cao, N. Schofield, and A. Emadi, "Battery balancing methods: A comprehensive review," in *IEEE Vehicle Power and Propulsion Conference (VPPC)*, 2008.

[29] J. Kimball, P. Krein, and K. Cahill, "Modeling of capacitor impedance in switching converters," *Power Electronics Letters, IEEE*, vol. 3, no. 4, pp. 136–140, Dec 2005.

[30] K. Nishijima, H. Sakamoto, and K. Harada, "A PWM controlled simple and high performance battery balancing system," in *Power Electronics Specialists Conference*, 2000.

[31] A. Stillwell and R. C. Pilawa-Podgurski, "A resonant switched-capacitor converter with GaN transistors for series-stacked processors with 99.8% power delivery efficiency," in *IEEE Energy Conversion Congress and Exposition*, 2015.

[32] S. Qin and R. C. Pilawa-Podgurski, "Sub-module differential power processing for photovoltaic applications," in *Applied Power Electronics Conference and Exposition*, 2013.

[33] E. Din, C. Schaef, K. Moffat, and J. Stauth, "Online spectroscopic diagnostics implemented in an efficient battery management system," in *Control and Modeling of Power Electronics*, 2015.

[34] P. S. Shenoy, B. Johnson, and P. T. Krein, "Differential power processing architecture for increased energy production and reliability of photovoltaic systems," in *Proc. Twenty-Seventh Annual IEEE Applied Power Electronics Conf. and Exposition (APEC)*, 2012, pp. 1987–1994.

[35] E. Candan, D. Heeger, P. Shenoy, and R. C. Pilawa-Podgurski, "A series-stacked power delivery architecture with hot-swapping for high-efficiency data centers," *Energy Conversion Congress and Exposition*, 2015.

[36] E. Candan, P. Shenoy, and R. C. Pilawa-Podgurski, "A Series-Stacked Power Delivery Architecture With Isolated Differential Power Conversion for Data Centers," *IEEE Transactions on Power Electronics*, vol. 31, pp. 3690–3703, May 2016.

[37] S. Qin, C. Barth, and R. C. Pilawa-Podgurski, "Enhancing microinverter energy capture with submodule differential power processing," *IEEE Transactions on Power Electronics*, vol. 31, pp. 357–3585, May 2016.

[38] *Design of Planar Power Transformers*, Ferroxcube, 1997, application note.

[39] Z. Ouyang, A. Cornelius, and M. Andersen, "Optimal design and tradeoffs analysis for planar transformer in high power dc-dc converters," *ECCE ASIA-IPEC-Sapporo*, 2010.

[40] M. Chen, M. Araghchini, K. Afridi, J. Lang, C. Sullivan, and D. Perreault, "A systematic approach to modeling impedances and current distribution in planar magnetics," *IEEE Transactions on Power Electronics*, 2016.

[41] *3F45 Material Specification*, Ferroxcube, 2008, data sheet.