Development of a High-Efficiency Solar Micro-Inverter

by

Alexander Khaled Hayman

B.S., Massachusetts Institute of Technology (2008)

Submitted to the Department of Electrical Engineering
and Computer Science
in Partial Fulfillment of the Requirements for the Degree of

Master of Engineering in Electrical Engineering
and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2009

Author .....
Department of Electrical Engineering
and Computer Science
September 1, 2009

Certified by .....
David J. Perreault
Associate Professor
Thesis Supervisor

Accepted by .....
Christopher J. Terman
Chairman, Department Committee on Graduate Theses

Development of a High-Efficiency Solar Micro-Inverter

by

Alexander Khaled Hayman

Submitted to the Department of Electrical Engineering
and Computer Science
on September 1, 2009, in Partial Fulfillment of the
Requirements for the Degree of
Master of Engineering in Electrical Engineering
and Computer Science

Abstract

In typical solar power installations, multiple modules are connected to the grid through a single high-power inverter. However, an alternative approach is to connect each solar module directly to the grid through a micro-inverter. This approach makes the system robust to single module failures and results in better power tracking. This project involves the development of a next generation micro-inverter architecture, including the design, assembly, and testing of a prototype converter. The topology involves a full bridge resonant inverter at the input, which supplies high-frequency current through a transformer to a cycloconverter at the output.

Thesis Supervisor: David J. Perreault
Title: Associate Professor

Acknowledgments

Thanks to Dr. Perreault for providing me with a project completely in line with my academic interests, and for being an excellent advisor, mentor, and teacher, from 6.002 to 6.987. Thanks to Enphase Energy for providing the motivation and financial support for this project. Thanks to the MIT EECS department, for funding me through a 6.01 teaching assistantship.

I especially thank the two graduate students whom I worked with on this project, Aleksey Trubitsyn and Brandon Pierquet. I worked very closely with Aleksey to get the prototype running in time for a visit from Enphase Energy, and also in time for me to collect the data I needed to finish this thesis. Without his dedication and sense of humor, we would have accomplished very little. Thanks to my other friends and colleagues in LEES who made this experience a lot of fun: Robert Pilawa, Tony Sagneri, George Hwang, JingYing Hu, Grace Cheung, Wei Li, Zachary Clifford, and Professor Leeb. I should also mention that Aleksey introduced me to El Chumbo, who often kept our spirits up during some of those late nights in the lab, and the machine upstairs that faithfully dispensed hot drinks for our 4:00 break.

Thanks to Dave Kaiser and Martin Fornage at Enphase Energy for their patience while we tackled this project. Thanks to Donny Zimmanck and Charlie Sullivan for the inductor and transformer designs.

Thanks to my friend and mentor, Alex Mevay, who first introduced me to power electronics and Brookline Lunch. Thanks to my best buds: Dayan Paez, Alexis Dale, Shane Colton, Tyler Hutchison, Ravioli Gupentaw, Crazy Driver, and the Sprints 2008 Crew.

My experiences with the Solar Electric Vehicle Team turned out to be essential over the course of this project, especially with the assembly and debugging of the prototype, so I am greatly indebted to all the students who have been involved with the team.

I also owe a special thanks to my sister, my father, and my mother's spirit.

THIS PAGE INTENTIONALLY LEFT BLANK

Contents

THIS PAGE INTENTIONALLY LEFT BLANK

List of Figures

THIS PAGE INTENTIONALLY LEFT BLANK

List of Tables

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 1

Introduction

Renewable energy has become an important focal point for research at MIT, reflecting national and international pressure to drive invention and innovation that will help to solve the world's energy woes. Solar power research has expanded considerably at MIT along with installed solar power capacity around the world. Between 2007 and 2008, world-wide grid-connected solar power capacity grew by more than 50% [4].

Harnessing solar power presents numerous technical challenges from a variety of fields, most notably physics, mechanical engineering, materials science, and electrical engineering. For electrical engineers, a primary challenge is converting the low voltage ($\sim$.5V) DC generated by a typical silicon photovoltaic (PV) cell to the high voltage ($\sim$200V RMS) AC of a grid.

Part of the problem is solved simply by the packaging of PV cells. A solar module is a basic unit of purchase for a consumer who wishes to put together a PV installation. A module usually consists of 50 to 80 cells connected in series, encapsulated in glass, and held together with an aluminum frame. Because these cells are connected in series, a typical module operates at around 25-40V DC and generates around 175W of power.

In a typical solar installation, multiple modules are connected in series so that the total voltage of the string surpasses the peak line voltage. A switching inverter is then used to connect the entire string to the grid. These inverters, made by manufacturers such as Solectria, SMA, and Xantrex, are typically rated for 2-5kW output power and an input voltage of 200-500V.

1.1 Micro-inverters

A different approach involves connecting each module directly to the grid through its own low-power, low-input-voltage inverter, or micro-inverter. The two approaches have several tradeoffs. Generally, inverters rated for higher power have better energy conversion efficiency. However, connecting one micro-inverter to each module improves peak power tracking and robustness to single module failures in an array. Micro-inverters may improve the overall energy output of an installation, even if their energy conversion efficiency is slightly worse than that of high-power inverters.

An example of a solar micro-inverter, manufactured by Enphase Energy, is pictured in Figure 1-1. The circuitry is encased in a small aluminum box, which attaches to a rack behind a solar module. The devices have three connections: two are AC connections to the grid so that the micro-inverters can be daisy-chained, and the third one is a DC connection to the panel. The devices communicate through the AC wiring to a central, internet-enabled box, which allows data about each panel to be gathered and accessed on the web.

This project involves the development of a next generation micro-inverter architecture, including the design, assembly, and testing of a prototype converter. The primary goal is to validate a conversion architecture that is well-suited to emerging trends in semiconductor devices in order to achieve improved efficiency.

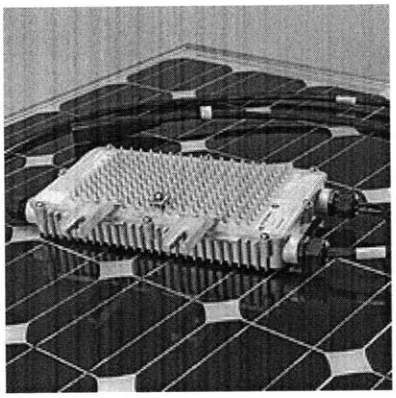


Figure 1-1: Commercially available micro-inverter (manufactured by Enphase Energy).

1.2 CEC Efficiency

One of the most important metrics for the performance of a solar inverter is its California Energy Commission (CEC) efficiency rating. This number is a weighted sum of the energy conversion efficiency of an inverter over a broad range of operating points. The weightings are shown in Table 1.1. Seemingly, the weightings are chosen so that a higher CEC efficiency indicates a higher net energy production over the operating lifetime of the inverter. The weightings imply that a solar inverter generates most of its energy at around 50-75% of its peak power rating. An example commercial product is rated for a CEC efficiency of about

% Rated Power	10%	20%	30%	50%	75%	100%
Weighting Factor	.04	.05	.12	.21	.53	.05

Table 1.1: CEC Weightings

94.5%. The goal of this project is to ultimately achieve 97% CEC efficiency. This thesis explores a topology that shows the potential to reach the target efficiency.

1.3 Topology

The chosen topology, shown in Figure 1-2, has two stages: a full-bridge series resonant inverter and a cycloconverter. To understand the intended operation of this circuit, a short description of the two stages follows.

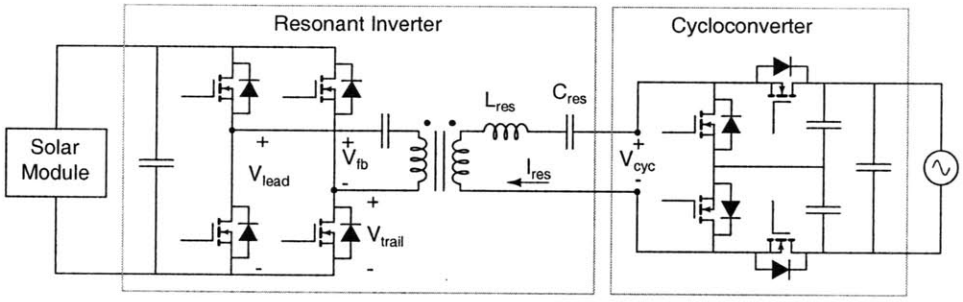


Figure 1-2: Micro-inverter Topology

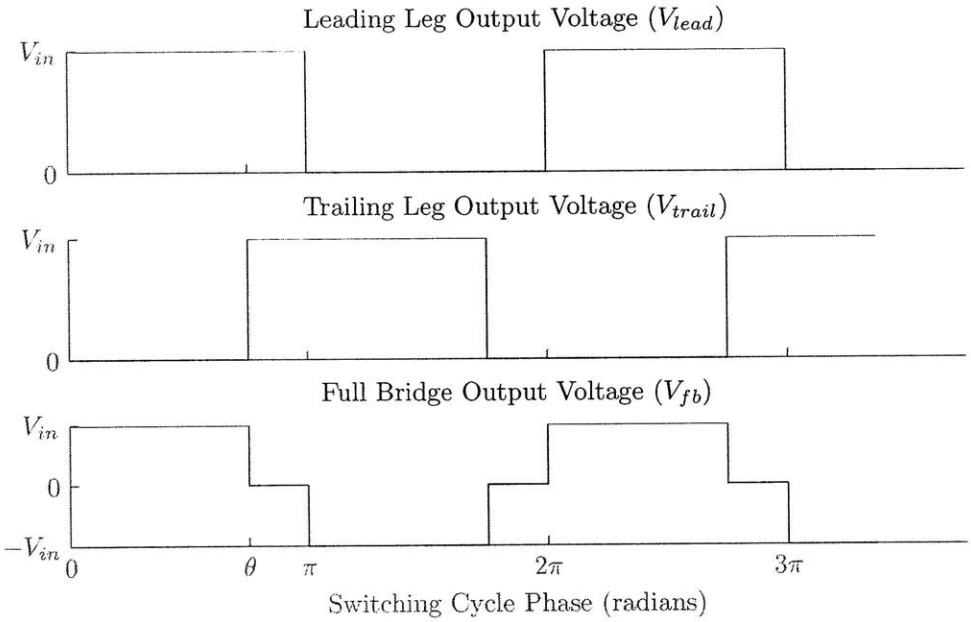


Figure 1-3: These plots demonstrate the operation of the full-bridge. The waveform definitions are shown in Figure 1-2.

1.3.1 Resonant Inverter

The four switches of the full-bridge are controlled to generate an output voltage as shown in Figure 1-3. The leading and trailing legs of the full-bridge are operated at 50% duty cycle, and the differential output voltage waveform is determined by the DC panel voltage, V_{in}, and the phase shift angle between the two legs, θ. This control scheme is commonly known as Phase Shift Pulse Width Modulation (PSPWM). The full-bridge drives a series resonant tank and generates an approximately sinusoidal current oscillating at the switching frequency.

1.3.2 Cycloconverter

A cycloconverter is used to convert the high-frequency current generated by the resonant inverter to a much lower line-frequency current. The two switches that share a node with $V_{cyc,+}$ form the positive leg of the cycloconverter, and the two other switches form the negative leg. When the line voltage is positive, the negative leg switches are closed, while the positive leg operates at 50% duty cycle. Likewise, when the line voltage is negative, the positive

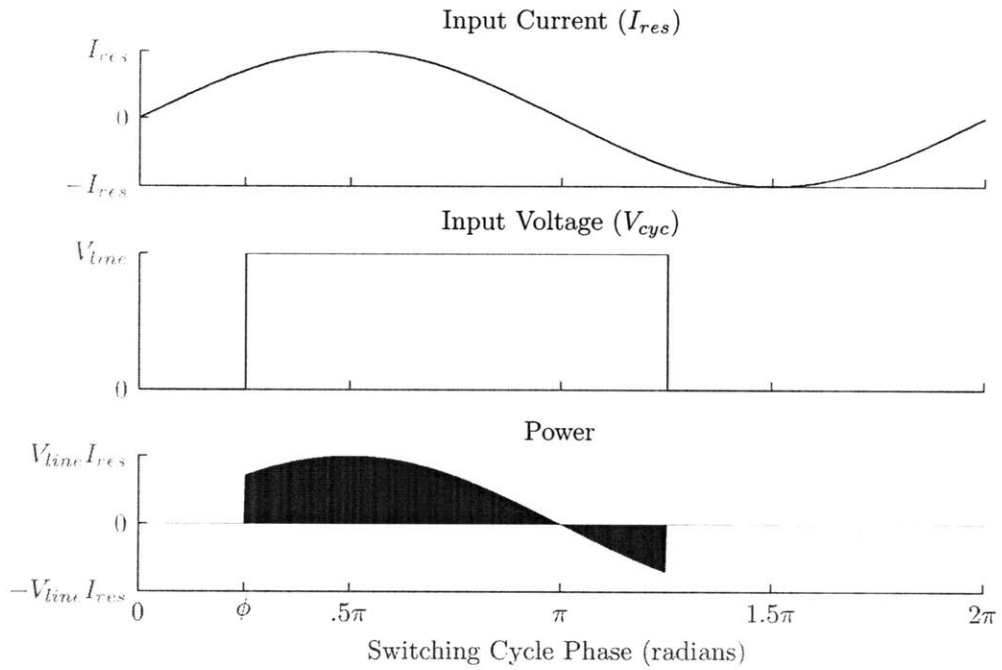


Figure 1-4: These waveforms demonstrate the operation of the cycloconverter. The waveforms are as defined in Figure 1-2. The phase shift, ϕ, between I_{res} and V_{cyc} determines the power flow into the line. V_{cyc} is controlled by the cycloconverter switches.

leg switches are closed, while the negative leg operates at 50% duty cycle. The phase shift angle (ϕ) between the cycloconverter input voltage and the resonant current determines the power flow into the line. When full power delivery is required, the cycloconverter acts as a rectifier (ϕ close to zero), delivering only positive or negative charge depending on the polarity of the line voltage. When zero power is required (corresponding to zero volts across the line), ϕ is set to $\frac{\pi}{2}$ so that an average of zero charge is delivered during each switching cycle. The line voltage is approximated as constant during a switching cycle, because the switching frequency is three orders of magnitude greater than the line frequency.

1.4 Important Characteristics

Several characteristics of this topology indicate that it can achieve the target of 97% CEC efficiency and meet necessary volume and weight standards. First, the circuit can be operated so that all switches are zero voltage switched (ZVS), effectively reducing switching loss to close to zero. ZVS requires that a switch turn on only when the body diode of the switch is

conducting. Since N-channel MOSFETs will be used as switches, the ZVS requirement will constrain the operation of the switches so that the current into the drain must be negative when the switch turns on.

Second, the losses mostly scale with output power. This is important because the CEC scaling dictates that efficiency remain relatively constant over a wide power range. In this circuit, as output power decreases, the RMS current in the circuit decreases, which also means that conduction losses decrease. Conduction losses are assumed to be a major component of the losses in the circuit.

Third, in order to keep the inverter relatively small and lightweight, the resonant tank and the transformer operate at high frequency. Line frequency magnetics are not an option because they would be too heavy, too costly, and take up too much space. This circuit does require an input capacitor sized for twice the line frequency, in order to maintain a steady voltage across the solar module despite varying output power over the course of a line cycle.

1.5 Specifications

The following target specifications are provided for the prototype. Note that the "maximum output power" specified is output power averaged over a line cycle.

Minimum Input Voltage	25V
Maximum Input Voltage	40V
Output Voltage	240V RMS
Maximum Output Power	175W
CEC Efficiency	>97%

1.6 Organization

Chapter 2 will go into detail about the theory and high level design of the inverter. A preliminary control scheme will be presented. Based on the control scheme, a simplified sinusoidal steady-state model will be presented. The model will be used to derive conduction

losses and to calculate appropriate control signals for static operating points across a line cycle.

Chapter 3 will document the magnetics designs, switch selection, and the microcontroller-based setup that was used to generate the gating signals for the switches. Also, the procedure for taking measurements and the calculations for determining CEC efficiency will be presented.

Chapter 4 will describe initial testing that was performed to help determine a control scheme. The control scheme will be discussed in detail, and CEC efficiency data points will be presented.

Chapter 5 will discuss the results and present recommendations for a path forward to achieve improved CEC efficiency.

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 2

Theory and Control

The theoretical analysis of this circuit is based on an initial attempt at a high-efficiency control strategy. The control strategy is designed to minimize RMS currents in the circuit and thereby minimize conduction loss. The resonant inverter will modulate power over the course of a line cycle, and the cycloconverter will behave as a rectifier and unfolder, delivering all the power that it can to the line. This ensures that minimal excess current is supplied by the resonant inverter, and minimal excess energy is sloshed around in the resonant tank.

Please note that for simplicity of language, "power delivered to the line averaged over a switching cycle" will be referred to as "instantaneous power", even though it is technically not instantaneous. Also, when a waveform is referred to in an equation, the amplitude of the waveform is the intended value, unless otherwise specified.

2.1 Resonant Inverter

This section will present a preliminary analysis of the voltage waveform that drives the resonant tank, a simple model of the cycloconverter, and logic behind the selection of several key parameters, including the transformer turns ratio and the resonant inductance.

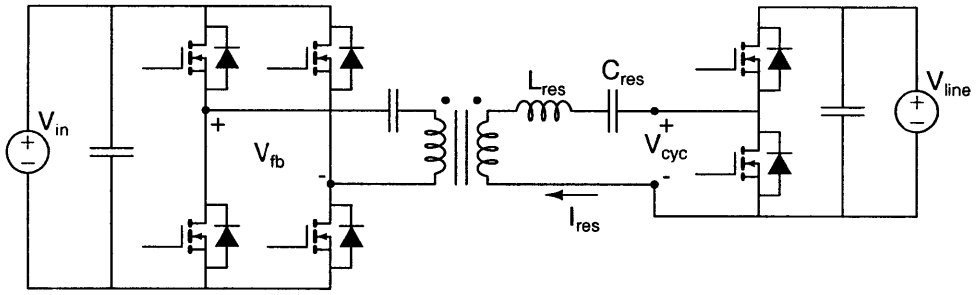


Figure 2-1: Circuit diagram of the full-bridge inverter and a simplified model of the cycloconverter that is valid when the line voltage is positive.

2.1.1 Full Bridge Voltage

To model the input to the resonant tank, only the fundamental component of the full-bridge voltage waveform is considered. The full-bridge is operated at a switching frequency above the resonant frequency of the tank so that the third, fifth, and higher harmonics of the square wave inputs can be disregarded. By superposition, the amplitude of the fundamental ($V_{fb,1}$) can be expressed as a function of the DC panel voltage (V_{in}) and the phase shift angle between the two legs of the full-bridge (θ, as shown in Figure 1-3).

$$V_{fb,1} = \frac{2}{\pi} V_{in} |(1 - e^{j\theta})| = \frac{4}{\pi} V_{in} \sin \frac{\theta}{2} \tag{2.1}$$

2.1.2 Equivalent Load

To simplify the analysis of the resonant inverter, the cycloconverter is modeled as an equivalent impedance at the fundamental switching frequency. The resonant inverter is expected to modulate current so that it scales linearly with the line voltage. If the cycloconverter acts as a rectifier ($\phi = 0$) delivering only positive current when the line voltage is positive and negative current when the line voltage is negative, its equivalent impedance is purely resistive. Figure 2-4 shows an equivalent circuit model depicting this operating condition. In Figure 2-2, the fundamental of the input voltage to the cycloconverter (with the DC component included) is shown in dotted lines and has an amplitude of $V_{cyc,1} = \frac{2}{\pi} V_{line}$, where V_{line} is the instantaneous value of the line voltage at a point in the line cycle.

The instantaneous output power ($\bar{P}$) can be calculated from the magnitude of the fun-

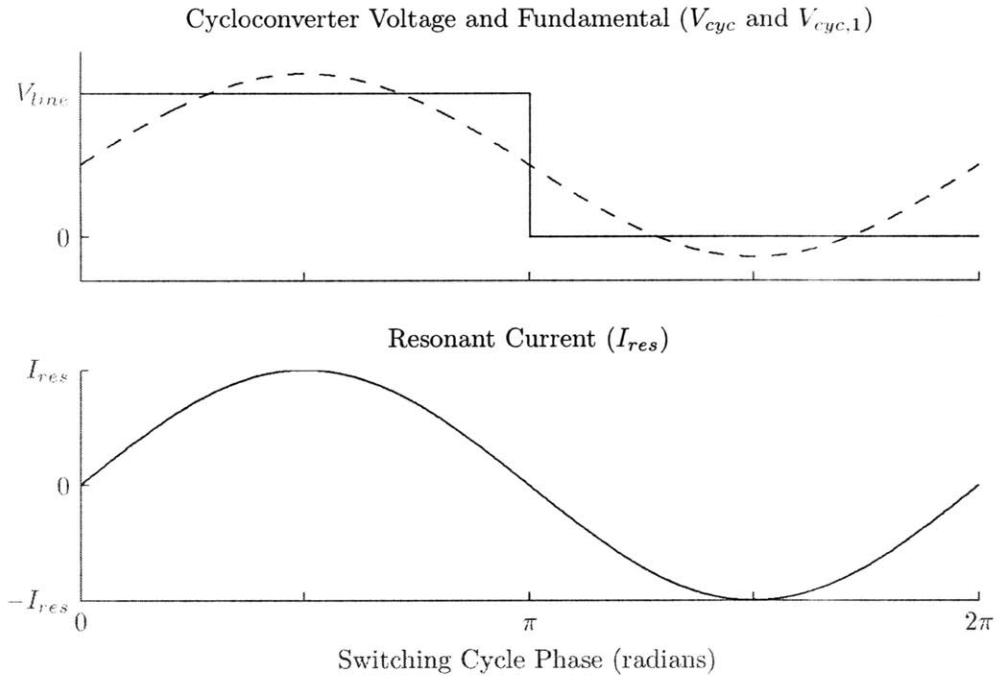


Figure 2-2: These waveforms illustrate the cycloconverter being operated as a rectifier.

damental of the input voltage and the input current, also referred to as the resonant current (I_{res}).

$$\bar{P} = \tfrac{1}{2} V_{cyc,1} I_{res} = \tfrac{1}{\pi} V_{line} I_{res} \tag{2.2}$$

For a given average output power, an equivalent load resistance (R_{ld}) defines the relationship between line voltage and instantaneous output power. To find R_{ld}, the instantaneous power is defined as a function of the RMS of the fundamental of the voltage across R_{ld} ($V_{cyc,1,\mathrm{RMS}}$).

$$\bar{P} = \frac{V_{cyc,1,\mathrm{RMS}}^2}{R_{ld}} = \frac{\frac{2}{\pi^2} V_{line}^2}{R_{ld}} \tag{2.3}$$

When the instantaneous line voltage is equal to the RMS of the line voltage ($V_{line,\mathrm{RMS}}$) average power over a line cycle (P_{avg}) is being delivered.

$$P_{avg} = \frac{\frac{2}{\pi^2} V_{line,\mathrm{RMS}}^2}{R_{ld}} \tag{2.4}$$

Thus, at the maximum average power of 175W into a 240V RMS line, $R_{ld} \simeq 66\Omega$.

2.1.3 Transformer

The transformer boosts the voltage at the output of the full-bridge so that it can meet the required voltage at the input of the cycloconverter. The worst case scenario occurs when the DC panel voltage is at its minimum, and the output line voltage is at the peak of the line cycle ($V_{line,\text{peak}}$). The full-bridge is assumed to be driving a full square wave ($\theta = \pi$) and operating at resonance so that the impedances of the resonant capacitor and inductor cancel. In practice, this is not an acceptable operating point, owing to the need for soft switching.

Under these conditions, the minimum voltage amplitude on the primary side of the transformer ($V_{pri,\text{min}}$) and the maximum voltage amplitude on the secondary side of the transfomer ($V_{sec,\text{max}}$) can be calculated. The minimum required turns ratio (N_{min}) is:

$$V_{pri,\text{min}} = \frac{4}{\pi} V_{in,\text{min}} = \frac{4}{\pi} \cdot 25\text{V} \simeq 31.8\text{V} \tag{2.5}$$

$$V_{sec,\text{max}} = \frac{2}{\pi} V_{line,\text{peak}} = \frac{2}{\pi}\sqrt{2} \cdot 240\text{V} \simeq 216\text{V} \tag{2.6}$$

$$N_{\text{min}} = \frac{V_{sec,\text{max}}}{V_{pri,\text{min}}} \simeq 6.8 \tag{2.7}$$

Based on these calculations, a conservative transformer turns ratio of $N = 8$ was chosen.

2.1.4 Resonant Inductor

The resonant inductor determines the quality factor of the resonant tank. A high quality factor ensures that the resonant current is sinusoidal, because harmonics are heavily attenuated. The expression for quality factor in a series resonant inverter is:

$$Q = \frac{1}{R_{ld}} \sqrt{\frac{L_{res}}{C_{res}}} \tag{2.8}$$

From Equation 2.4, as average power output decreases, the equivalent load resistance increases, and the quality factor will decrease. This means that, for a given quality factor and resonant frequency, $\omega_0 = \sqrt{\frac{1}{L_{res}C_{res}}}$, the required resonant inductance increases as the average power is reduced. Below a minimum average power, the cycloconverter can be used to

% Rated Power	10%	20%	30%	50%	75%	100%
Inductance (μH)	700	350	233	140	93	70

Table 2.1: Inductor Sizes

modulate power. The resonant inductor will determine this "average power sloshing boundary." (Discussion of the cycloconverter and its impact on the circuit when it modulates power will be saved for Section 2.3.)

Since the goal is to maximize the CEC efficiency of the circuit, which is a function of the efficiency at six average powers, there are six possible optimal inductances, where each inductance is associated with a sloshing boundary. A required quality factor of 1 was chosen based on qualititave observation of current waveforms in a SPICE simulation. Six possible inductor values (shown in Table 2.1) were calculated for an arbitrarily chosen resonant frequency of 150kHz. A program was written to develop six optimal circuits designed for the six possible sloshing boundaries. It accounted for inductor losses, as well as switch-related conduction and gating losses (see Section 2.4 and Appendix A). The best CEC efficiency resulted from setting the sloshing boundary at 50% power, so an inductor size of 150μH was chosen.

It should be noted that the resonant inductor can also be placed on the low side of the transformer. For a turns ratio of 8, the equivalent inductance would be reduced to $\frac{150}{64}\mu$H, or 2.4μH. Initial attempts to design a low side inductor proved especially difficult because of the very small number of turns required and an associated lack of resolution, e.g. with an A_L value of $400\frac{nH}{turn^2}$, 2 turns give 1.6μH, and 3 turns give 3.6μH.

2.2 Resonant Inverter Control

This section will focus on the control of the resonant inverter, which is mostly responsible for modulating instantaneous output power. The instantaneous output power can be controlled with two handles: phase shift in the full-bridge (ϕ) and switching frequency (ω). Before studying this in detail, the limits on our use of these handles are established.

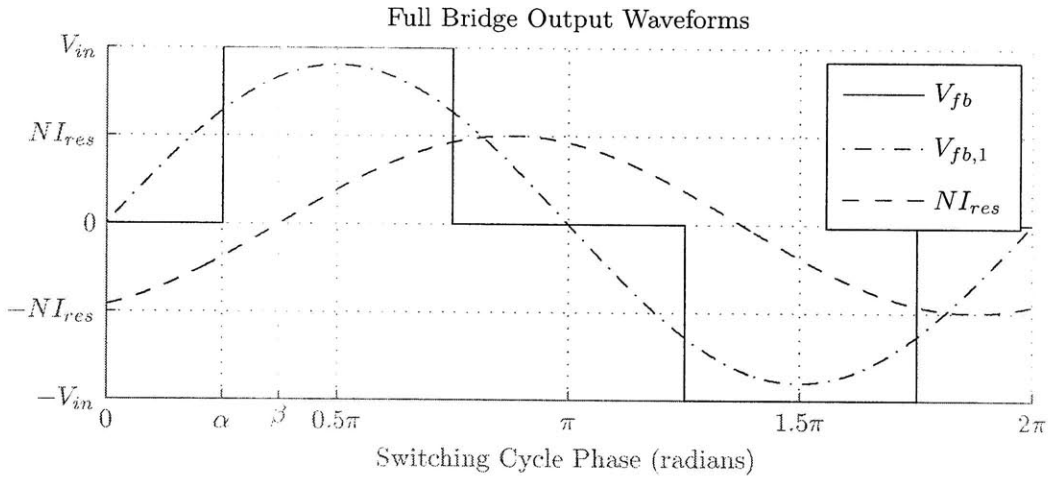


Figure 2-3: Waveforms that demonstrate ZVS in the full-bridge.

2.2.1 Conditions for ZVS

Maintaining ZVS is essential because of the high efficiency requirement that the circuit is aiming to fulfill. In Figure 2-3, an example of the full-bridge in a ZVS mode is shown, where the waveform labels correspond to the labels in Figure 2-1. In the figure, β is the phase delay between the current and the fundamental of the full-bridge voltage, and α, which is a simple function of θ, marks the phase delay between the fundamental of the full-bridge voltage and the turn-on of the high side switch in the leading leg. The constraint for ZVS is simply $\beta > \alpha$. This constraint limits the extent to which the full-bridge can take advantage of PSPWM to lower the instantaneous output power of the inverter. The difference between β and α will be referred to as the full-bridge ZVS phase margin, δ.

2.2.2 Frequency Control

Once the PSPWM limit has been reached, the instantaneous output power can be further reduced by increasing frequency. Figure 2-5 shows that as frequency increases, β increases, allowing us to safely increase α. Essentially, moving the frequency lever changes the limits on the PSPWM lever. The minimum frequency will be determined by the maximum power requirement. The transformer has been sized with enough margin that the circuit should never need to be operated very close to the resonant frequency. The maximum frequency limit will be discussed in Section 2.2.4.

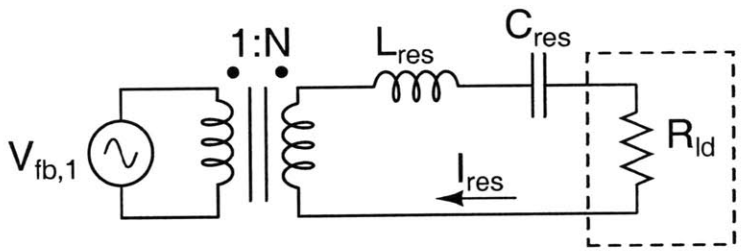


Figure 2-4: This is a simplified model of the inverter, which is true *only* when the cycloconverter is operated as a rectifier. $V_{fb,1}$ is the fundamental of the full-bridge voltage, as shown in Figure 2-1 and calculated in Equation 2.1. R_{ld} is the equivalent load resistance of the cycloconverter at a given average power level.

2.2.3 Power Modulation

With the cycloconverter modeled as a resistor, determining how instantaneous power can be modulated by the resonant inverter becomes simple. The impedance of the resonant circuit shown in Figure 2-4 is:

$$Z_{res} = R_{ld} + L_{res}j\omega + \frac{1}{C_{res}j\omega} \tag{2.9}$$

The resonant current driven into the load is:

$$I_{res} = \frac{V_{fb,1}N}{Z_{res}} = \frac{\frac{4}{\pi}V_{in}N\sin\frac{\theta}{2}}{R_{ld} + L_{res}j\omega + \frac{1}{C_{res}j\omega}} \tag{2.10}$$

Finally, power is given as a function of the resonant current:

$$\bar{P} = I_{res,\mathrm{RMS}}^2 R_{ld} = \tfrac{1}{2}I_{res}^2 R_{ld} \tag{2.11}$$

Note that these expressions are only valid when the cycloconverter is operated as a rectifier.

2.2.4 Maximum Frequency

The frequency lever does not have an upper limit that is as easy to define as the PSPWM limit. Inductors and switches are optimized for specific frequencies or ranges of frequencies, so the switching frequency will be limited. Above a specified frequency limit (or below the associated minimum power) the cycloconverter, which introduces a third lever, will be used

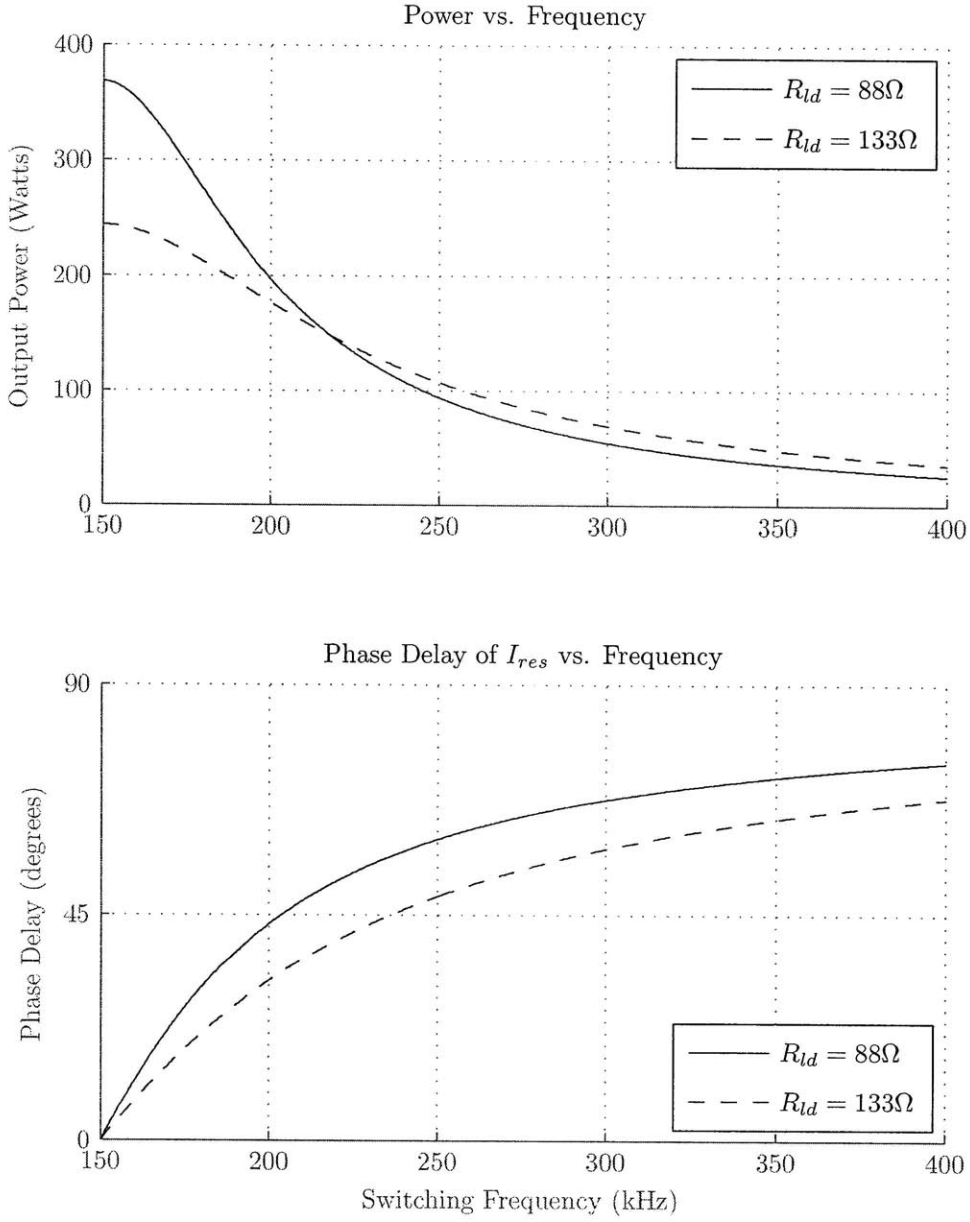


Figure 2-5: This plot shows how, at 75% (88Ω) and 50% (133Ω) average output power levels, frequency affects the output power and the phase delay (β) of the resonant current relative to the fundamental full-bridge voltage. The chosen parameters are: $\theta = \pi, C_{res} = 7.5\text{nF}, L_{res} = 150\mu\text{H}, V_{in} = 25\text{V}, N = 8$.

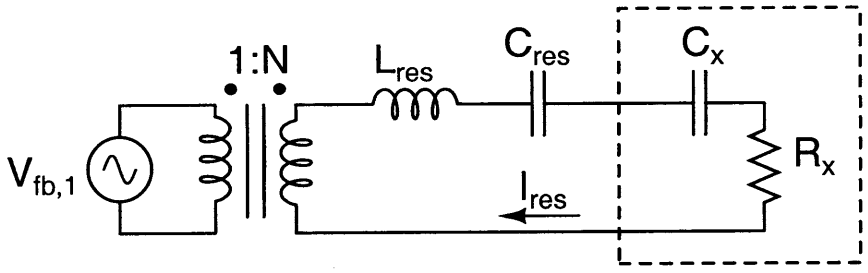


Figure 2-6: This is a more general version of the inverter model shown in Figure 2-4. This model accounts for the reactive component in the equivalent load impedance when the cycloconverter is not behaving purely as a rectifier.

to modulate power. This frequency will be referred to as the "instantaneous power sloshing boundary."

2.3 Cycloconverter Analysis

The cycloconverter is responsible for modulating output power at low power levels (below the instantaneous and average power sloshing boundaries). The power delivered as a function of line voltage, resonant current, and cycloconverter phase shift (ϕ as defined in Figure 1-4) is:

$$\bar{P} = \frac{1}{2\pi}\int_{\phi}^{\phi+\pi} V_{cyc}(t) I_{res}(t)\, dt = \frac{1}{\pi} V_{line} I_{res} \cos\phi \tag{2.12}$$

In order to ensure ZVS, the voltage will lag the current, so the equivalent load impedance (Z_x) now has a reactive component. This complex impedance is modeled by a capacitor (C_x with reactance X_x) in series with a resistor (R_x), as illustrated in Figure 2-6.

$$Z_x = R_x + jX_x = R_x + \frac{1}{C_x j\omega} \tag{2.13}$$

$$|Z_x| = \frac{\sqrt{(R_x C_x \omega)^2 + 1}}{C_x \omega} \tag{2.14}$$

$$\angle Z_x = -\arctan\frac{1}{R_x C_x \omega} = -\phi \tag{2.15}$$

2.3.1 Impedance Trajectory

To make this complex impedance easier to understand, the trajectory of the impedance as ϕ shifts from 0 to $\frac{\pi}{2}$ will be derived. The magnitude of the load impedance defines the ratio of the fundamental cycloconverter voltage waveform to the resonant current.

$$|Z_x| = \frac{\frac{2}{\pi}V_{line}}{I_{res}} \tag{2.16}$$

From Equations 2.4 and 2.12:

$$R_{ld}\cos\phi = \frac{\frac{2}{\pi}V_{line}}{I_{res}} \tag{2.17}$$

Combining this with Equations 2.14 and 2.16 results in:

$$\cos\phi = \frac{\sqrt{(R_x C_x \omega)^2 + 1}}{R_{ld} C_x \omega} \tag{2.18}$$

This expression and Equation 2.15 reduce to the following pairwise relationships between X_x, R_x, and ϕ:

$$R_x = R_{ld}\cos^2\phi \tag{2.19}$$

$$X_x = -R_{ld}\cos\phi\sin\phi \tag{2.20}$$

$$X_x = -\sqrt{R_x(R_{ld} - R_x)} \tag{2.21}$$

As shown in Figure 2-7, the impedance trajectory looks like a half circle, starting at R_{ld}, curving down to a minimum reactance, $X_x = -\frac{R_{ld}}{2}$ when $R_x = \frac{R_{ld}}{2}$, and then curving back to 0.

2.3.2 Further Analysis

Since the added capacitive component is in series, the new apparent resonant capacitance will decrease ($C'_{res} = \frac{C_{res}C_x}{C_{res}+C_x}$) and the resonant frequency and quality factor will increase. Furthermore, R_x will be a fraction of R_{ld}, so the quality factor will increase further. This means that as ϕ varies from 0 to $\frac{\pi}{4}$ radians, the resonant current is guaranteed to increase if

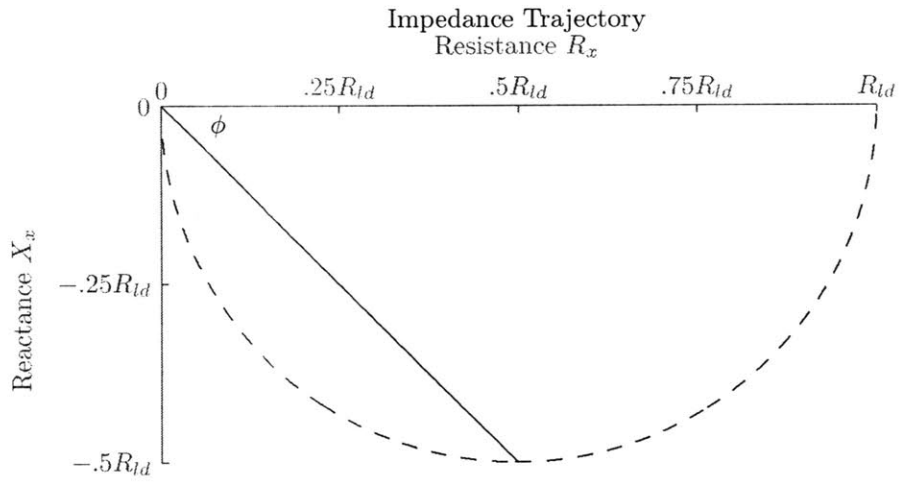


Figure 2-7: This plot shows how the load impedance varies with ϕ. When $\phi = 0$, the equivalent impedance is $Z_x = R_{ld}$. As ϕ approaches $\frac{\pi}{2}$, Z_x approaches 0.

θ and ω are kept constant.

For the operating point depicted in Figure 2-8, when ϕ is very small, output power is actually greater than when $\phi = 0$, because the resonant current magnitude increases at a faster rate than the decreasing "$\cos\phi$" term in Equation 2.12. Theoretically, ϕ has to jump from 0 to some angle in order to maintain the relationship between output power and line voltage (Equation 2.4). This discontinuity is evident in the figure.

2.4 Loss Calculations

The circuit design was optimized using a simple loss model that accounted for losses in the switches and the inductor.

2.4.1 Switch Losses

The circuit consists of eight MOSFETs: the four low voltage MOSFETs in the full-bridge, which must withstand the open circuit voltage of the panel, and the four high voltage MOSFETs in the cycloconverter, which must withstand the peak line voltage. All MOSFETs are assumed to be zero voltage switched. The output capacitance keeps the drain-source voltage low while the switch turns off, and the switch waits to turn on until its body diode

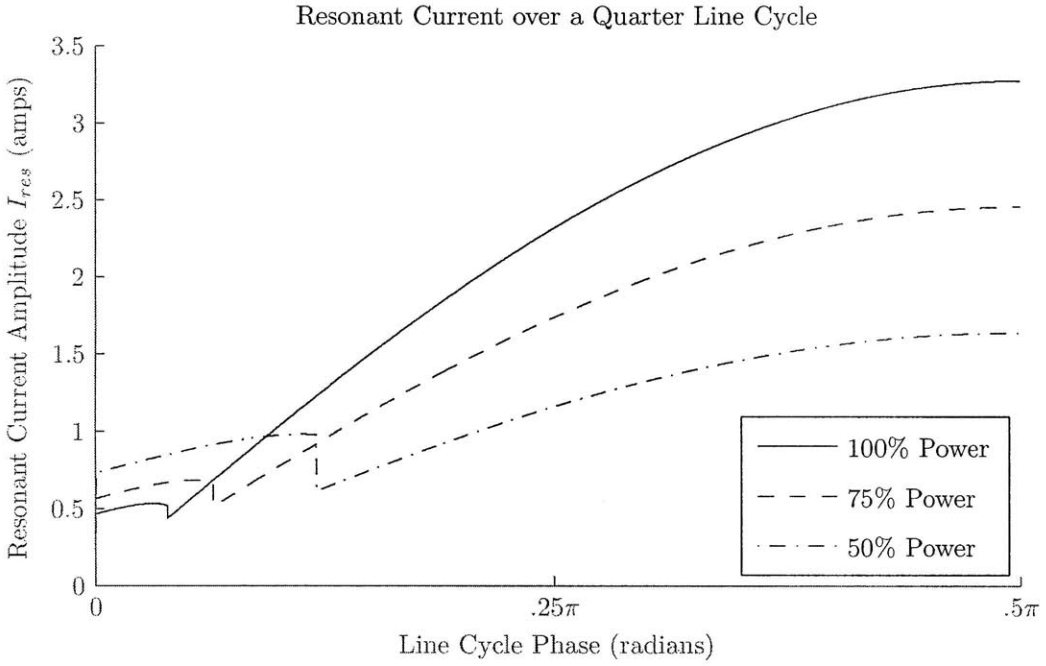


Figure 2-8: "Bumps" in resonant current amplitude occur when the maximum switching frequency is limited to 400kHz. Here, θ is set to its minimim point (while still maintaining ZVS), in order to minimize switching frequency for a given output power, and all other parameters are the same as indicated in Figure 2-5.

is fully conducting. Theoretically, the only substantial losses are conduction loss due to the drain-source resistance (R_{ds}) given an RMS current through the switch ($I_{sw,\text{RMS}}$) and loss associated with charging and discharging the gate given a gate charge (Q_g), a gate-source voltage (V_{gs}), and a switching frequency in Hertz (f_{sw}). The expected average power loss in a single switch is:

$$P_{sw} = I_{sw,\text{RMS}}^2 R_{ds} + 2Q_g V_{gs} f_{sw} \tag{2.22}$$

Low Voltage Switch Losses

For the four switches in the full-bridge, eight gate transitions occur per switching cycle, and two switches are conducting at all times (disregarding dead-time). The total loss in the full-bridge switches is based on the gate charge and drain-source resistance of the low voltage switches, $Q_{g,lo}$ and $R_{ds,lo}$, respectively.

$$P_{sw,lo} = 2(\tfrac{1}{\sqrt{2}} N I_{res})^2 R_{ds,lo} + 8Q_{g,lo} V_{gs} f_{sw} \tag{2.23}$$

High Voltage Switch Losses

In the cycloconverter, it can be approximated that the current always flows through 1.5 switches. When the positive leg switches are being switched at 50% duty cycle, the current can flow through both of the negative leg switches, which are held on. Only two switches at a time are operated at the switching frequency, which means that there are only four gate transitions per switching cycle in the cycloconverter. The total loss in the cycloconverter switches is based on the gate charge and drain-source resistance of the high voltage switches, $Q_{g,hi}$ and $R_{ds,hi}$, respectively:

$$P_{sw,hi} = 1.5(\tfrac{1}{\sqrt{2}} I_{res})^2 R_{ds,hi} + 4Q_{g,hi} V_{gs} f_{sw} \tag{2.24}$$

2.4.2 Inductor Losses

Inductor losses traditionally break down into two components, core loss and winding loss.

Core Loss

The core loss was calculated from the Steinmetz parameters of the material. Our core choices were limited to commonly available ferrite materials (3F3, 3C96) and to the commonly available cores (RM14, RM12, RM10). The parameters were taken from a supplier's application note [1]. The Steinmetz equation is of the form:

$$P_{L,core} = V_c \cdot k \cdot f^{\alpha} \cdot B_{peak}^{\beta} \tag{2.25}$$

$P_{L,core}$ is the loss in milliwatts, V_c is the core volume in cm^3, f is the frequency of the inductor current in Hz, and B_{peak} is the amplitude of the core flux density in Tesla. The three Steimetz parameters provided are k, α, and β. The flux density is a function of the number of turns (N_L), the effective core area in m^2 (A_e), the resonant current through the inductor, and the A_L value ($\frac{\text{nH}}{\text{turn}^2}$), which relates to the size of the gap.

$$B_{peak} = \frac{A_L \cdot N_L \cdot I_{res}}{10^9 \cdot A_e} \tag{2.26}$$

Winding Loss

Because of the need to maximize efficiency and because of the frequencies at which the circuit operates, the use of Litz wire is necessary. Litz wire mitigates the impact of skin effect and proximity effect in high frequency magnetics. A technique described by Gu and Liu was used to calculate the AC resistance ($R_{L,AC}$) of an inductor that uses Litz wire [2]. The following equation shows winding loss associated with a constant resonant current.

$$P_{L,wind} = (I_{res,\text{RMS}})^2 R_{L,AC} \tag{2.27}$$

The details of the calculations can be found in the paper by Gu and in the code in Appendix A.

Chapter 3

Prototype Implementation

This chapter will document the parts selection, board layout, and assembly of a prototype that was built to test the validity of the loss model. The test setup that was used to collect efficiency data will also be documented.

3.1 Switches

The two important parameters that were considered for MOSFET selection were the gate capacitance (Q_g) and the drain-source resistance (R_{ds}). The product of Q_g and R_{ds} were used as a metric for initial switch selection. These two parameters are typically inversely proportional to each other. Intuitively, at higher relative switching frequencies, the gate capacitance has more impact on loss. Likewise, for higher relative RMS currents, the drain-source resistance is the primary loss mechanism. Switches available from popular suppliers (Digikey and Mouser) in popular footprints (TO-220 and D2PAK) were considered. The Q_g and R_{ds} values for promising switches were plugged into a Python program (included in Appendix A). The program picked out switches that gave the best overall CEC efficiency.

3.1.1 Low Voltage Switches

The full-bridge switches need to withstand the voltage of the solar module. The maximum open circuit voltage is 60V, so switches with a minimum drain-source voltage rating of

75V were considered. STMicroelectronics MOSFETs (STB160N75F3) were used for the full bridge. They have a drain-source voltage rating of 75V, a drain-source resistance of 3.2mΩ, and a gate charge of 85nC.

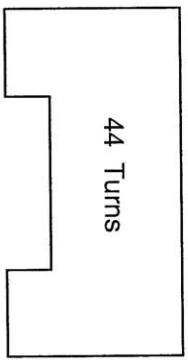
3.1.2 High Voltage Switches

The cycloconverter switches need to withstand the peak line voltage as well as line transients. MOSFETs with minimum drain-source voltage ratings of 600V were considered. The program picked out a variant of an Infineon 650V MOSFET. Initially, the IPP60R099 was used in the prototype, with an R_{ds} of 99mΩ, and a Q_g of 80nC. For reasons discussed in Section 4.1.2, the cycloconverter switches were changed to IPP60R250, with an R_{ds} of 250mΩ, and a Q_g of 26nC.

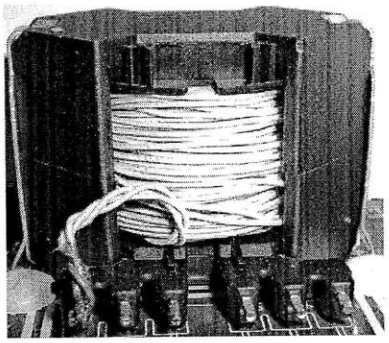
3.2 Inductor

Core	RM14
Material	3F3
Gap	3.6mm
Turns	44
Litz gauge	44
Litz strands	243
Inductance	165μH

(a)



(b)



(c)

Figure 3-1: Table (a) enumerates the design details of the inductor. Figure (b) shows a cross section of the winding area. The left edge is closest to the center of the bobbin. Insulative tape was used to keep windings out of the center region close to the gap. Figure (c) is a picture of the inductor.

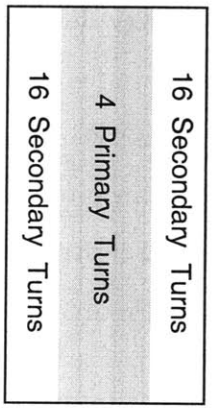
The Python program was able to generate preliminary inductor designs, however they turned out to be unrealistic, because the simulated packing factor was too high. Donny Zimmanck of Dartmouth provided us with the final designs for both the inductor and the transformer. The designs were constrained to use Litz wire and cores made of 3F3 material

already available at MIT. The winding consisted of 3 lengths of 3x27x44AWG Litz wire braided together. The RM14 core has a 3.6mm air-gap, which results in significant fringing fields near the gap. The fringing fields aggravate the proximity effect in windings that are close to the gap [3]. In order to minimize this effect, before the inductor was wound, 10mm-wide insulating tape was wrapped around the center of the bobbin until the tape reached a thickness of 2mm. The tape forces more space between the windings and the gap.

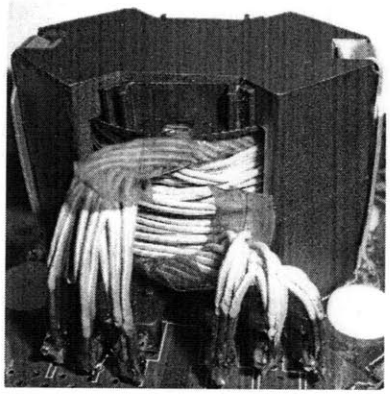
3.3 Transformer

Core	RM14
Material	3F3
Gap	0mm
Litz gauge	40
Primary turns	4
Primary strands	600
Secondary turns	32
Secondary strands	80

(a)



(b)



(c)

Figure 3-2: Table (a) enumerates the design details of the transformer. Figure (b) shows a cross section of the winding layout. The left edge is closest to the center of the bobbin. Figure (c) is a picture of the transformer that shows where the primary connections come out of the core.

The 1:8 transformer was designed to handle a peak magnetic flux of $\sim 100\mu$V-s on the primary. It is made out of an ungapped core, in order to minimize the amount of stored energy. The secondary winding was split into two winding windows to reduce losses associated with the proximity effect. Because of the significant voltage difference between the two windings, the primary winding was completely wrapped in Kapton tape. The primary is made of 15 lengths of 40x40AWG Litz wire that were twisted together. The secondary is made of 2 lengths of 40x40AWG Litz wire.

3.4 Prototype Board

The prototype board was designed with flexibility in mind by Brandon Pierquet of MIT. The digital signals to the gate drivers are controlled through headers that could be connected to any control interface. Extra pads and foot prints were placed to allow for differently sized inductors and resonant capacitances.

The board's design also reflects effort to minimize electromagnetic noise and ringing caused by parasitic inductances. The gate drivers are placed as close as possible to the switches to control gate voltage ringing at the transitions. High frequency capacitors rated for appropriate current ripple were placed in close proximity to each leg of the full-bridge in order to hold DC voltages constant and minimize ringing of MOSFET drain-source voltages.

This board was also meant to test circuitry developed by Aleksey Trubitsyn of MIT for measuring the amount of charge delivered by a leg in a single switching cycle. This involved testing a shunt and a current transformer as two different methods for measuring current. In short, the measurement circuit involves an integrator, two sample holds, and an ADC, along with other digital circuitry for generating timing signals. The measurement circuitry is located below the transformer, shown in Figure 3-3.

3.5 Control Interface

In Chapter 2, three control levers were discussed: frequency (ω), phase shift between the full-bridge legs (θ), and phase shift between the cycloconverter input voltage and the resonant current (ϕ). The position of these levers is decided by the gating signals that control the MOSFETs. A microcontroller was used to generate these gating signals, because they are easy to reprogram with a computer, and the wiring can easily be reconfigured with commonly available development boards.

The Atmel AT90PWM316 microcontroller, along with the STK500 development board, and the STK520 extension kit were used. The microcontroller is operated at 16MHz, but it also comes with a 64MHz Phase Locked Loop, which is used to generate a higher frequency clock input to the three "Power Stage Controllers" (PSCs). A PSC is a digital waveform

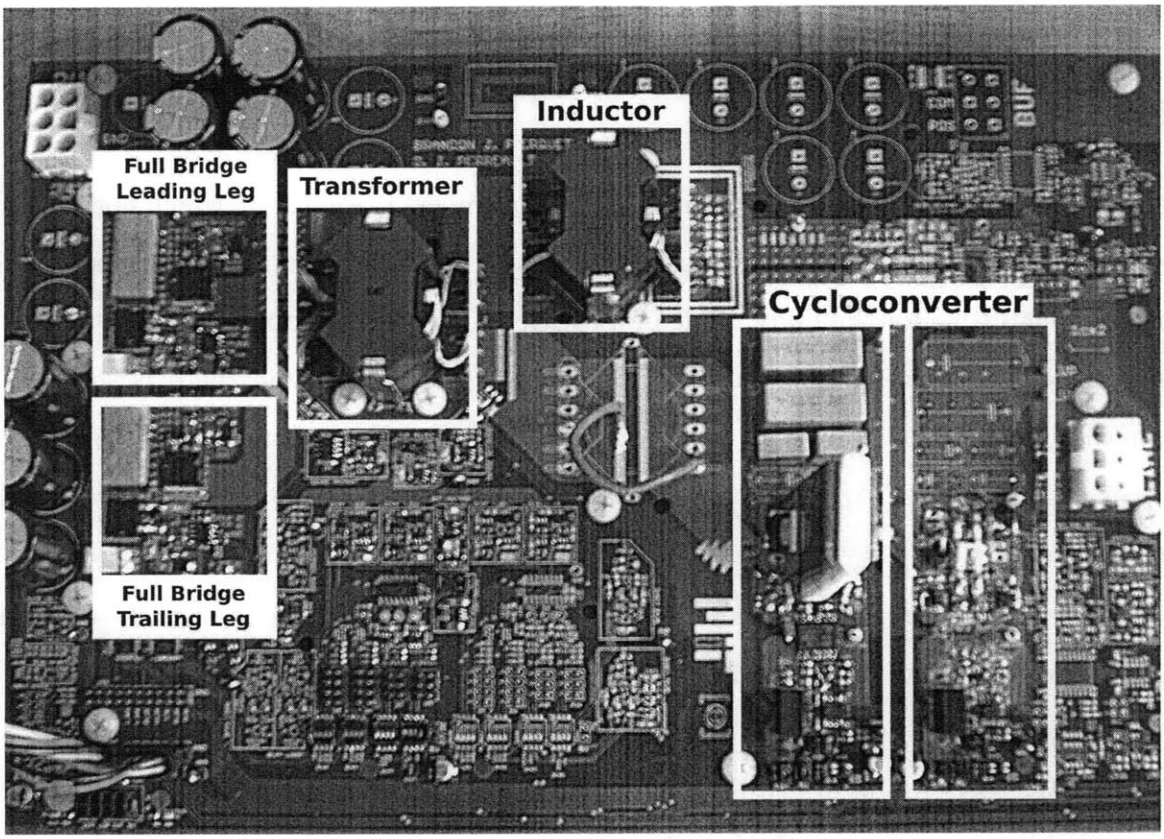


Figure 3-3: Picture of the prototype board, with important parts of the board labeled. Note the two legs of the cycloconverter are indicated in two separate boxes. For initial, static input and output voltage testing, the negative leg (on the right) was shorted. The board is approximately 12"x8".

controller that can be programmed to generate gating signals. With a 64MHz clock driving the PSCs, the frequency resolution at 200kHz is around 630Hz per step, and the phase shift resolution is 1.1° per step. The microcontroller code, included in Appendix B, was written to set up the necessary gating signals for a static operating point given three values, one value per control lever. The calculations to determine the appropriate microcontroller values are discussed in Section 4.2.

3.5.1 Dead-time

Dead-time is a small time period (relative to the switching period) during which both switches in a leg are off. Sufficient dead-time is a prerequisite for ZVS. The dead-time is set so that there is enough time for the body diode of a MOSFET to start conducting before the switch turns on. If the dead-time is too short, the switch turns on while there is still voltage across the diode, which will result in switching losses. If the dead-time is too long, there is unnecessary loss caused by the forward voltage drop across the body diode, which would be mostly eliminated if the switch were on.

In the experimental setup, dead-times could be adjusted by potentiometers on a small, simple daughter board. The board took as input a 50% duty cycle signal directly from the microcontroller. The rising edges of this input signal and its inverse were delayed through the use of RC low pass filters and a digital buffers, effectively generating gating signals for the top and bottom switches of a leg. A potentiometer adjusted dead-time for a gate signal by altering the RC time constant of the corresponding low pass filter.

Because of the multitude of static operating points to test, it was not practical to manually adjust dead-times at each operating point. As a result, all dead-times were kept constant for the duration of the tests. The dead-time for the leading leg of the full-bridge was 225ns, for the trailing leg of the full-bridge, 275ns, and for the cycloconverter, 500ns. The dead-time for the trailing leg is slightly shorter than that of the leading leg because the transitions in the trailing leg often occur during a period of very high current. To mitigate losses associated with this transition, a Schottky diode (Diodes Inc. B180-13-F) was placed across the drain and source of both MOSFETs in the trailing leg.

3.6 Experimental Setup and Procedure

This section will document the experimental setup and methods that were used to make the measurements discussed in Chapter 4.

3.6.1 Power Supplies and Loads

In place of a solar module, two Agilent 6643A power supplies were wired in parallel. Each supply is rated for 35V at 6A. This limited our ability to test at the maximum 40V input voltage. This also limited our ability to test at the absolute peak input current of 14A, given a 25V input at 350W. However, most of the operating points could be tested with this setup.

To test static line voltages, the following active loads were used: an Agilent 6063B rated for 240V and 250W and an Agilent 6050A, which contains three 60V, 600W active loads. All of the power channels were wired in series in order to allow testing at the peak line voltage (~340V).

3.6.2 Efficiency Measurements

For simplicity, the circuit was tested at static operating points, where each point is defined by an input voltage, an average output power, and a point in the line cycle. At each static operating point, four raw measurements were taken: input voltage, input current, output voltage, and output current. The input voltage measurement was taken from the power supply display, after verifying the reading with a high precision multimeter. The input current was determined by using an HP 34410A multimeter to measure voltage across a precision 0.1Ω shunt. The output voltage and output current were both measured by HP 34401A's. The voltage readings are 0.1% accurate, and the current readings are 0.5% accurate. These measurements do not take into account the gating losses, since the 12V gate drive circuitry was powered by a separate supply.

3.6.3 Efficiency Calculations

Efficiency (ϵ) at a particular input voltage and average output power is a function of input energy over a line cycle ($E_{lc,in}$) and output energy over a line cycle ($E_{lc,out}$).

$$\epsilon = \frac{E_{lc,out}}{E_{lc,in}} \tag{3.1}$$

Energy over a line cycle (E_{lc}) is estimated from a trapezoidal integral of instantaneous power at four equally-spaced phase points in a quarter line cycle ($.125\pi$, $.25\pi$, $.375\pi$, and $.5\pi$ radians).

$$E_{lc} = P_{.125\pi} + P_{.25\pi} + P_{.375\pi} + .5P_{.5\pi} \tag{3.2}$$

This calculation generates a value that is proportional to energy.

For a given input voltage, the inverter was tested at 100%, 75%, 50%, and 30% rated average power (since these are operating points that the CEC cares about). The efficiencies at 20% and 10% rated average power account for only 9% of the overall CEC efficiency, and for simplicity, the efficiency at these points was assumed equal to the efficiency at 30% rated average power (i.e. the efficiency at 30% rated average power accounted for 21% of the total CEC efficiency).

Total CEC efficiencies were calculated for input voltages of 25V and 34V. The 34V limit was imposed by the power supply. Otherwise, efficiencies for an input voltage of 40V would be measured in order to test the full range of input voltages according to the specification.

Chapter 4

Testing and Results

This chapter will be divided into three sections. The first section will describe initial testing that led to some modifications to the circuit and the development of a control scheme. The second section will describe the control scheme and the algorithm that was used to generate the control values at the operating points enumerated in Section 3.6. The last section will present a set of results for the design as of this writing.

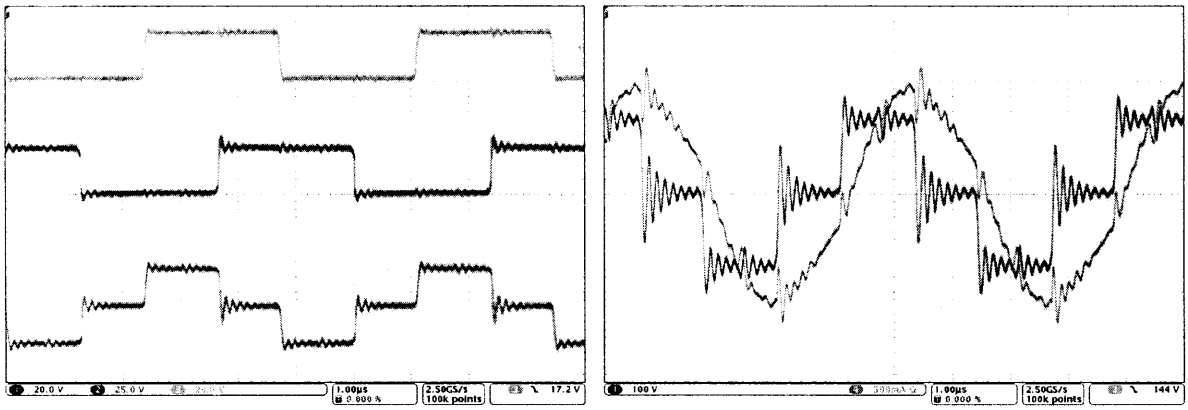
4.1 Initial Testing

The control scheme for the inverter was guided by preliminary tests that were performed as the inverter was brought to full operation.

4.1.1 Minimizing Switching Frequency

The first aspect of the control scheme was determined from efficiency measurements that were taken while driving a resistive load (in place of the cycloconverter) with the resonant inverter. Current through the resistor was measured with a current probe connected to an oscilloscope. The resistances tested were 66Ω, 100Ω, and 133Ω, which correspond to 100%, 66%, and 50% average output powers, respectively.

The first set of efficiency measurements were taken with a fixed full-bridge phase shift ($\theta = \pi$). Power into the resistive load was controlled purely with frequency. This was



(a) (b)

Figure 4-1: These oscilloscope screenshots illustrate the operation of the full-bridge.
(a) corresponds to Figure 1-3 and shows the drain-source voltage of the low side FET in the leading leg (top waveform) and trailing leg (middle waveform). The bottom waveform is the differential voltage, V_{fb}, which is impressed on the primary of the transformer.
(b) corresponds to Figure 2-3 and shows the voltage across and current through the secondary of the transformer (NV_{fb} and I_{res}) This operating point illustrates the full-bridge taking full advantage of PSPWM, while maintaining ZVS. The ringing in the voltage waveform is discussed in Section 5.1.3.
The operating parameters are: V_{in}=25V, V_{line}=240V, R_{ld}=133Ω, f_{sw}=211kHz, θ=100° C_{res}=13.2nF, L_{res}=165μH, N=8.

followed by tests wherein the circuit was controlled to take advantage of PSPWM, which has the affect of reducing the switching frequency at a given output power. The ZVS phase margin (δ, from Section 2.2.1) was fixed at 10°, and the same operating points were tested. Efficiency improved across the board at the lower switching frequencies. In cases where the switching frequency was drastically reduced, efficiency jumped by up to 4% at a given output power into a given load.

4.1.2 Cycloconverter Adjustments

After these preliminary tests, the cycloconverter was installed and tested. It was immediately discovered that the ZVS condition requires a substantial phase shift (ϕ) between the resonant current and the fundamental of the cycloconverter voltage.

At a given line voltage, a certain amount of charge is required to turn on the body diode of one switch in a leg and turn off the body diode of the other switch in the same leg

(i.e. to charge and discharge the device capacitances). This fact was ignored in the initial computations. In order to achieve a ZVS transition, this charge must be delivered by the resonant current within the dead-time. Ensuring adequate dead-time is extremely important, because hard switching results in disruptive noise and substantial switching losses. If ϕ is small, the current at transition time will be small, and a long dead-time will be required. If ϕ is larger, the current at transition time will be larger, and the required dead-time will be shorter.

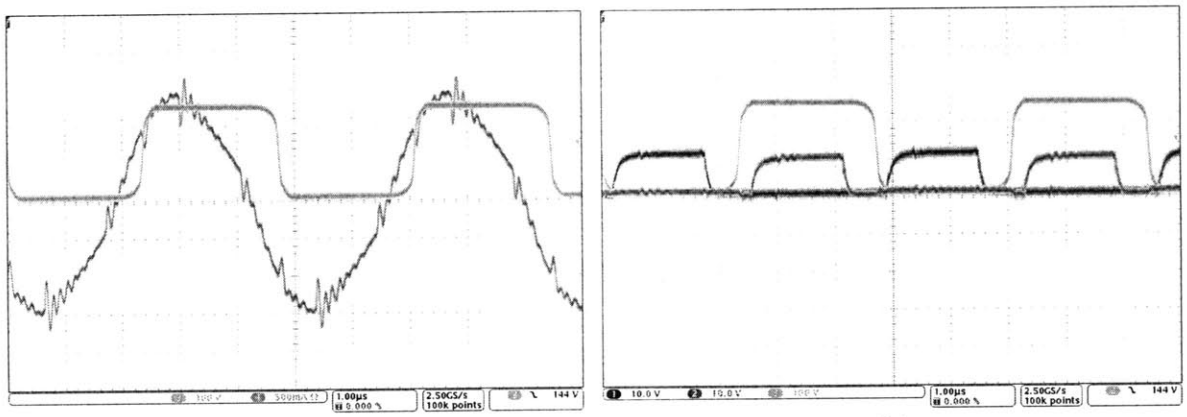
With the IPP60R099 MOSFET and a configured dead-time of 500ns, the typical minimum phase shift was 40°. This substantial phase shift motivated the search for a similar FET with a lower charge requirement for the body diode transitions, even though such a FET is expected to have a higher R_{ds}. Lowering the charge requirement will reduce the minimum ϕ and reduce the resonant current needed (see Section 2.3). Reducing the resonant current will reduce conduction losses in the entire circuit, while increasing $R_{ds,hi}$ will only impact the cycloconverter switches. With the IPP60R250 and the same dead-time of 500ns, the typical minimum ϕ was only 20°. The installation of these new switches improved the efficiency at all tested operating points by 1-2%.

4.2 Control Scheme

The initial testing discussed above resulted in a Matlab script (included in Appendix C) to calculate a table of control values to be coded into the microcontroller for each static operating point. For all combinations of input voltage, average output power, and line cycle phase, the script numerically calculates the necessary switching frequency and the phase difference between the full-bridge legs. The script assumes a constant ϕ of 10° and a fixed δ of 10° to ensure ZVS, while minimizing switching frequency. The algorithm increments through all frequencies that the microcontroller can generate and performs the following calculations at each frequency f_{sw}:

1. Calculate C_x and R_x based on average output power, f_{sw}, and $\phi = 10°$.

2. Find the impedance of the resonant circuit (Z_{res}) using C_x and R_x as the load.



(a) (b)

Figure 4-2: These oscilloscope screenshots illustrate the operation of the cycloconverter.
(a) corresponds to Figure 1-4. The square-wave signal is V_{cyc} and the sinusoidal signal is I_{res}, measured on the secondary side of the transformer. Here, IPP60R099 MOSFETs were used, and at this operating point the minimum phase shift to achieve ZVS is 38° given a dead-time of 750ns.
(b) demonstrates ZVS in the cycloconverter. The same V_{cyc} signal is shown, along with the gate-source voltages for the FETs in the positive cycloconverter leg. The low-side FET is not turned on until V_{cyc} has dropped all the way to 0V. Likewise, the high-side FET is not turned on until V_{cyc} has reached V_{ln}.
The operating parameters for these waveforms are the same as in Figure 4-1.

3. Find the impedance angle (β) of Z_{res} at f_{sw}

4. Find the magnitude of $V_{fb,1}$ based on V_{in}, β, and $\delta = 10°$.

5. Calculate output power from $V_{fb,1}$, N, Z_{res}, f_{sw}, and ϕ.

The script selects f_{sw} and θ such that the output power is close to what is expected at the chosen operating point. When actually testing each operating point, ϕ was manually tuned so that a dead-time of 500ns was just long enough to achieve ZVS. The actual output power of the inverter varied from the expected power by no more than 10%.

4.3 CEC Measurements

A complete CEC efficiency measurement was taken according to the procedure described in Section 3.6. The resonant capacitance was 6.6nF, and the blocking capacitor on the primary side of the transformer was 17.6μF. The CEC efficiency was 90.54% with a 34V input and 92.32% with a 25V input. The CEC efficiency at high input voltage is worse than at lower input voltage, because for a given output power, a higher input voltage requires a higher switching frequency.

The current waveforms looked much more sinusoidal than expected from the simulations. At 30% average output power, higher order harmonics were expected to become pronounced. However, the simulations assumed a purely resistive load and did not account for the minimum phase shift requirement. Two revisions to the model result from a nonzero cycloconverter phase shift: the apparent resistance of the load is reduced and a capacitive component is introduced. Both of these revisions result in a higher quality factor.

Also, while taking measurements, the transformer and inductor became very hot. The temperatures approached 70°C. After noting the power dissipation in the magnetics and the excess quality factor, it was predicted that lowering the switching frequency would have a substantial impact on efficiency. The simplest modification to the circuit involved increasing the resonant capacitance, which effectively translates excess quality factor into a lower resonant frequency (and lower switching frequencies).

% P_{max}	Phase (rad)	P_{in} (W)	P_{out} (W)	$E_{lc,in}$	$E_{lc,out}$	% Efficiency
100	$.5\pi$	330*	305.92	698.75	646.47	92.52
	$.375\pi$	300*	278.11			
	$.25\pi$	178.25	165.24			
	$.125\pi$	55.50	50.15			
75	$.5\pi$	280.00	262.86	556.50	520.51	93.53
	$.375\pi$	234.75	221.14			
	$.25\pi$	140.00	130.31			
	$.125\pi$	41.75	37.62			
50	$.5\pi$	203.25	192.73	397.38	372.24	93.67
	$.375\pi$	172.25	162.57			
	$.25\pi$	98.25	91.32			
	$.125\pi$	25.25	21.99			
30	$.5\pi$	141.50	133.33	263.30	242.87	92.24
	$.375\pi$	117.75	110.33			
	$.25\pi$	60.50	55.01			
	$.125\pi$	14.30	10.87			

Table 4.1: These measurements were taken with an input voltage of 25V. The approximated CEC efficiency is 93.24%. The asterisks denote operating points that required too much current for the power supply. Values were plugged in that assumed the same efficiency as measured at the $.25\pi$ phase point. L_{res}=165μH, C_{res}=13.2nF, N=8.

The measurements shown in Table 4.1 and Table 4.2 were taken after the resonant capacitance was doubled from 6.6nF to 13.2nF. The CEC efficiency was 91.32% with a 34V input and 93.24% with a 25V input. The two are averaged together to give an overall CEC efficiency of 92.3%. While running these tests, the inductor and transformer were significantly cooler than previously. The current waveforms still looked sinusoidal down to 30% average output power.

Chapter 5 will discuss these results and a path forward to further improve efficiency.

% P_{max}	Phase (rad)	P_{in} (W)	P_{out} (W)	$E_{lc,in}$	$E_{lc,out}$	% Efficiency
100	$.5\pi$	331.16	306.63	704.48	648.09	92.00
	$.375\pi$	298.86	276.61			
	$.25\pi$	183.94	168.37			
	$.125\pi$	56.10	49.79			
75	$.5\pi$	259.42	241.82	544.85	502.92	92.30
	$.375\pi$	233.24	216.73			
	$.25\pi$	140.76	129.35			
	$.125\pi$	41.14	35.93			
50	$.5\pi$	185.64	173.03	376.14	345.08	91.74
	$.375\pi$	164.90	153.12			
	$.25\pi$	95.88	87.23			
	$.125\pi$	22.54	18.21			
30	$.5\pi$	117.30	106.98	231.23	204.11	88.27
	$.375\pi$	100.98	91.46			
	$.25\pi$	49.64	42.51			
	$.125\pi$	21.96	16.65			

Table 4.2: These measurements were taken with an input voltage of 34V. The approximated CEC efficiency is 91.32%. L_{res}=165μH, C_{res}=13.2nF, N=8.

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 5

Conclusions

The final CEC efficiency data points presented in Chapter 4 surpass 92%. The lessons that were learned from testing the physical prototype can be used to improve the model of the circuit. An improved model will result in a better optimization of the design. This chapter will discuss these lessons and corresponding modifications that should be made to the optimization program that guided the design of this initial prototype.

5.1 Model Revisions

5.1.1 Minimum Phase Shift

The circuit design was optimized using a program that did not account for the minimum phase shift requirement in the cycloconverter. However, in Section 4.1.2, tests showed that the minimum phase shift requirement in the cycloconverter and the corresponding dead-time plays a significant role in the efficiency of the circuit. A lower phase shift requirement results in lower resonant currents in the entire circuit. However, a lower phase shift can only be achieved with cycloconverter MOSFETs that have higher drain-source resistances.

Furthermore, in Section 4.3, tests showed that the minimum phase shift requirement in the cycloconverter also affects the quality factor of the resonant tank. The quality factor was higher than expected due to the combination of a reduced equivalent load resistance and an introduced equivalent capacitance. A higher quality factor allows for a smaller, more

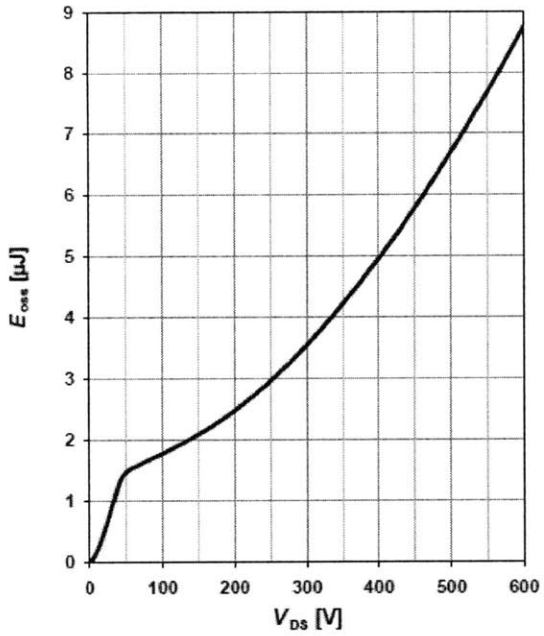


Figure 5-1: This plot was taken from the IPP60R250 datasheet, and shows an approximate linear relationship between the energy stored in the output capacitance of the FET (E_{oss}) and the drain-source voltage (V_{ds}).

efficient inductor and/or operation at a lower resonant frequency by increasing the resonant capacitance.

To achieve higher efficiency, an updated optimization program should be written that can calculate the required phase shift for a given dead-time, power level, and line voltage. If ϕ is the phase shift of the cycloconverter voltage with respect to the resonant current, Δ is the dead-time, and E_{oss} is the energy stored in the output capacitance of a FET, the constraint for ZVS is:

$$2E_{oss} = \int_{\phi-\frac{\Delta}{2}}^{\phi+\frac{\Delta}{2}} I_{res}(t)V_{ds}(t)\,dt \tag{5.1}$$

The E_{oss} is doubled, because the output capacitance of one FET in the leg must be charged and that of the other FET must be discharged. This model can be simplified by assuming that the drain-souce voltage, V_{ds}, transitions linearly during the dead-time (from V_{ln} to 0V and vice versa) and that the resonant current is constant during the dead-time (since it is very short relative to the period). In Figure 5-1, it is clear that this model can be further simplified by modeling E_{oss} as linearly related to V_{ds} (and V_{ln}). With these assumptions, ϕ

can then be more easily calculated:

$$2E_{oss} = \frac{1}{2} I_{res}(\phi) V_{ln} \Delta \tag{5.2}$$

Using this model, an optimization program can account for the impact that the minimum phase shift has on resonant current and quality factor. This will result in a more accurate calculation of conduction losses and a more informed selection of a resonant inductance. This model should be verified through testing of the physical prototype.

5.1.2 Frequency

The measurements show that switching frequency has a significant impact on efficiency. It is known that higher input voltages require higher frequencies to attenuate the full-bridge fundamental voltage waveform. This increase in frequency likely accounts for most of the 2% difference between high and low input voltage CEC efficiencies.

The optimization program over-simplified the circuit by assuming a constant switching frequency. Unfortunately, calculating the switching frequency for an operating point becomes very processor intensive, especially when accounting for minimum phase shift. However, given the significant impact that frequency has on the efficiency, a more accurate representation of switching frequency at a given operating point is necessary to properly optimize the circuit.

The optimization program also arbitrarily set the resonant frequency of the tank. Instead, a hill-climbing algorithm should be implemented to find an optimal resonant frequency, which would determine the optimal inductance for a given average power sloshing boundary.

5.1.3 Magnetics Designs

The inductors that were designed by the optimization program were unrealistic, because the packing factor of the windings was lower than expected. An updated program should create designs based on measurements of Litz wire. Furthermore, different core materials should be explored for the inductor and transformer.

It should be noted that there was significant ringing in voltage waveform on the secondary side of the transformer (Figure 4-1b). This ringing suggests an unwanted resonance at around 3MHz, likely a result of parasitic capacitance in the magnetics designs. This ringing causes unwanted ripple in the current waveform. Eventually, the magnetics designs should be modified so that parasitic capacitances do not cause significant ringing.

5.2 Looking Forward

After a second round of optimization, a redesigned inverter shows promise for reaching higher efficiencies. After verification of the inverter's ability to perform at static operating points, the next challenge will be to figure out how to dynamically control the circuit to drive power into a 60Hz line, in addition to handling maximum point power tracking. Hopefully, this work will contribute to the development of more efficient solar inverters, which will translate to more energy extracted from solar panels. Also, the topology used in this inverter can be slightly modified and used to transfer power in the opposite direction. In other words, this work could also contribute to the development of more efficient AC-DC converters.

Appendix A

Loss Optimization Code

A.1 Script

This Python code was used to pick switches and an inductor size that would give maximal CEC efficiency. In retrospect, the code is insufficient because it assumes a constant switching frequency, and it doesn't take into account resonant currents that rise due to a non-zero cycloconverter phase shift.

```
################Begin main.py###################
import math
import ldata

PI = 3.14159
MU = 4e-7*PI

maxpower = 175 #Max power
voutrms = 240  #RMS line voltage
voutmag = voutrms*math.sqrt(2)
fline = 60         #Line frequency
wline = 2*PI*fline
fsw = 300000          #Switching frequency
wsw = 2*PI*fsw
turnsratio = 8

ppc = 120             #points per switching cycle
dt = 1/fsw/ppc        #time step per point
Msc = fsw/fline          #Msc is # of switch cycles per line cycle
Npoints = Msc*ppc     #total number of points in a line cycle

#CEC Energy Efficiency calculations
powerpercent = (.1, .2, .3, .5, .75, 1.0)
powerpercent = [maxpower*i for i in powerpercent]
powerweight = (.04, .05, .12, .21, .53, .05)

#find minimum inductances for ZVS at given average power level
#(based on CEC).  these values were found by using a SPICE simulation
#to look at the current waveforms
mininductor = (9.5, 4.4, 3.2, 1.8, 1.1, .9)
mininductor = [i*turnsratio**2*1e-6 for i in mininductor]

#for each inductor size, we need 6 RMS currents over a line cycle
#and a list of peaks of length fsw/fline
```

```
#this is the list of current peaks for each average ouput power
#(to be used for the inductor core loss calculation)
#we model only a quarter of the line cycle because of symmetry
irectpeaks = [[ppct*2*PI/voutmag*math.sin(2*PI*pks/(fsw/fline)) \
   for pks in range(int(fsw/fline/4))] for ppct in powerpercent]

#this is the list of RMS currents for each average output power
irectrms = [ppct/voutrms*PI/math.sqrt(2) for ppct in powerpercent]

#below the sloshing boundary, the currents stay constant
#this is an approximation
size10 = list(zip(irectrms,irectpeaks))
size20 = list(zip(2*[irectrms[1]]+irectrms[2:], \
   2*[irectpeaks[1]]+irectpeaks[2:]))
size30 = list(zip(3*[irectrms[2]]+irectrms[3:], \
   3*[irectpeaks[2]]+irectpeaks[3:]))
size50 = list(zip(4*[irectrms[3]]+irectrms[4:], \
   4*[irectpeaks[3]]+irectpeaks[4:]))
size70 = list(zip(5*[irectrms[4]]+irectrms[5:], \
   5*[irectpeaks[4]]+irectpeaks[5:]))
size100 = 6*[[irectrms[5],irectpeaks[5]]]

biglist = [size10, size20, size30, size50, size70, size100]

def strlst(lst):
   ret = "";
   for a in lst:
      ret = ret+"%5.3f,"%a
   return ret

#requires entry of FET data, will pick out the best FETs,
#and output the efficiencies.
def MOSFETLoss(rmscurrents):

   #Plug in resistance in ohms for high side FETs
   #Plug in corresponding Qg in nC
   rdshi = [.099, .299/2, .165]
   qghi = [60, 22*2, 39]
   qghi = [x*1e-9 for x in qghi]
   hinum = len(rdshi)

   #Do the same for the low side FETs
   rdslo = [.0123, .0032, .0035, .003, .0052, .0067]
   qglo = [25, 110, 85, 88, 51, 42]
   qglo = [x*1e-9 for x in qglo]
   lonum = len(rdslo)

   weightedefflo = []
   for i in range(lonum):
      gatingloss = 4*12*fsw*qglo[i]
      rloss = [2*(irms*turnsratio)**2*rdslo[i] \
         for irms in rmscurrents]
      effs = [1-(gatingloss+x[0])/x[1] \
         for x in zip(rloss,powerpercent)]
      weightedeff = sum([x[0]*x[1] for x in zip(powerweight,effs)])
      weightedefflo.append(weightedeff)
   maxl = max(weightedefflo)
   maxil = weightedefflo.index(maxl)
   print('Max low side efficiency of', (maxl*100))
   print("Rl=", rdslo[maxil], "ohms, Qg=", qglo[maxil]*1e9, "nC")

   weightedeffhi = []
   for i in range(hinum):
      gatingloss = 2*12*fsw*qghi[i]
      rloss = [1.5*irms**2*rdshi[i] for irms in rmscurrents]
      effs = [1-(gatingloss+x[0])/x[1] for x in zip(rloss,powerpercent)]
      weightedeff = sum([x[0]*x[1] for x in zip(powerweight,effs)])
```

```
      weightedeffhi.append(weightedeff)
   maxh = max(weightedeffhi)
   maxih = weightedeffhi.index(maxh)
   print('Max high side efficiency of',(maxh*100))
   print('Rh=', rdshi[maxih],'ohms, Qg=', qghi[maxih]*1e9,'nC')

   deviceeff = maxh+maxl-1
   print('Device max efficiency is', (deviceeff*100))
   return deviceeff


# This script is used to design resonant inductor
# Given the operational parameters, the script generates
# feasible designs for the inductor based on core and
# wire data in the script ldata.py  The script tries to
# design an inductor based on each core in the data set
# and retains those which are feasible.  For each design
# the script calculates core losses and winding losses
# (including losses due to skin and proximity effect)
# based on fitted empirical data and theoretical models.

# Set up parameters for this test
def IndLoss(L,currentdata):
   i_rms = [x[0] for x in currentdata]
   i_pks = [x[1] for x in currentdata]
   Ipeak = max([max(i_ps) for i_ps in i_pks])

   maxweightedeff = 0
   # Loop through available cores and synthesize
   # inductor designs.
   for core in range(ldata.numcores):
      # Calc number of turns needed on this
      # core to get desired L value.
      N = math.ceil(math.sqrt(1e9*L/ldata.AL[core]))
      if N==0:
         continue

      # Calculate B at peak current in Tesla
      # for this N.
      Bpeak = ldata.AL[core]*N*Ipeak/1e9/ldata.Ac[core]
      # If required peak B is more than 3000
      # gauss, then we are too close to the
      # Bsat = 300 mT not acceptable
      if Bpeak > .3:
         continue

      # If design is still ok, try to find a
      # wire size which fits on the core.
      # Select largest wire which fits.
      # This may not be at all optimal.

      # If the design is still ok, calculate
      # the inductor losses.  These losses
      # include the core losses and the
      # winding losses.

      # Calculate core losses.  This eqn. is
      # a fit of empirical data.  Pcore is in
      # watts, fsw is in kHz, B is in gauss,
      # Vc is in cm^3.  This is a conservative
      # estimate based on the peak flux swing
      # encountered in the cycle.

      Bpks = [[ldata.AL[core]*N*i_pk/1e9/ldata.Ac[core] for i_pk in i_ps] \
         for i_ps in i_pks]
      #3F3
      Bpksavg = [sum([Bpk**2.45 for Bpk in Bps])/len(Bps) for Bps in Bpks]
      #core loss in Watts
```

```
      Pcore = [2.5e-7*fsw**1.63*ldata.Vc[core]*Bpkavg for Bpkavg in Bpksavg]
      #3C96
      #Bpksavg = [sum([Bpk**2.8 for Bpk in Bpks])/len(Bpks) \
      #  for Bps in Bpks]
      #Pcore=[8.27e-8*(fsw)^1.72*ldata.Vc(core)*Bpkavg \
      #  for Bpkavg in Bpksavg]

      klayer = .7    #horizontal packing density
      packingfactor = .5
      gauge = 44
      strandarea = ldata.wirearea[gauge]
      maxnumstrands = math.floor(ldata.wam[core]*packingfactor/N/strandarea)
      skindepth = math.sqrt(ldata.resistivity/fsw/PI/MU)

      x = ldata.wirediam[gauge]/2/skindepth*math.sqrt(PI*klayer)
      litzwidth = math.sqrt(PI/4*maxnumstrands*strandarea/packingfactor)

      bestml = math.ceil(N*litzwidth/ldata.wwm[core])

      frmin = 1+(5*bestml**2*maxnumstrands-1)/45*x**4
      Rdcmin = ldata.ohmpermeter[gauge]*ldata.lptm[core]*N/maxnumstrands
      Racmin = frmin*Rdcmin
      bestnumstrands = maxnumstrands

      for numstrands in range(1,int(maxnumstrands)):
         Rdc = ldata.ohmpermeter[gauge]*ldata.lptm[core]*N/numstrands
         litzwidth = math.sqrt(PI/4*numstrands*strandarea/packingfactor)
         ml = math.ceil(N*litzwidth/ldata.wwm[core])
         fr = 1+(5*ml**2*numstrands-1)/45*x**4
         Rac = fr*Rdc
         if Rac<Racmin:
            Rdcmin = Rdc
            frmin = fr
            Racmin = Rac
            bestnumstrands = numstrands
            bestml = ml
      Pwind = [i**2*Racmin for i in i_rms]

      Ptot = [a[0]+a[1] for a in zip(Pwind,Pcore)]
      effs = [1-x[0]/x[1] for x in zip(Ptot,powerpercent)]
      weightedeff = sum([x[0]*x[1] for x in zip(powerweight,effs)])
      if weightedeff>maxweightedeff:
         maxweightedeff = weightedeff
         saved = [core, N, gauge, Bpeak, Ptot, Pwind, Pcore, \
            bestnumstrands, maxnumstrands, bestml, Racmin]

   [core,N,gauge,Bpeak,Ptot,Pwind,Pcore,\
      bestnumstrands,maxnumstrands,bestml,Racmin] = saved
   print("Best Inductor Configuration")
   print(ldata.corename[core],':', (N*N*ldata.AL[core]/1000.0),'uH')
   print(N,'turns of', gauge,'gauge litz wire with a')
   print(bestnumstrands, "strand count")
   print('ml=',bestml,'Rac=', Racmin)
   print('Bpeak=',Bpeak)
   print("Pcore="+strlst(Pcore)+"Watts")
   print("Pwind="+strlst(Pwind)+"Watts")
   print("Ptot="+strlst(Ptot)+"Watts")
   print("********Max inductor efficiency=", (maxweightedeff*100))
   return maxweightedeff

mosfeteffs = []
inductoreffs = []
for size in range(6):
   print("***********************************************************")
   print("Testing for best efficiency with inductor sized for ZVS at ")
   print("%3.1f%% average power output" %(powerpercent[size]/maxpower*100))
   print("requiring a %.2fuH resonant inductor" % (mininductor[size]*1e6))
   print("Current profile (RMS):")
```

```
    print(strlst([x[0] for x in biglist[size]]))

    print("Calculating MOSFET Losses")
    mosfeteffs.append(MOSFETLoss([x[0] for x in biglist[size]]))

    print("Calculating Inductor Losses")
    inductoreffs.append(IndLoss(mininductor[size], biglist[size]))

    print("Total Losses: %5.3f%%" % \
       ((mosfeteffs[size]+inductoreffs[size]-1)*100))
    print("*********************************************************")

totaleffs = [x[0]+x[1]-1 for x in zip(mosfeteffs,inductoreffs)]
################End main.py##################
```

A.2 Included Inductor Data

This file (ldata.py) includes information about inductor cores, which is used in the previous script.

```
################Begin ldata.py##################
# This script BRLdat.m defines variables containing data
# for boost-rectifer inductor design.  Params for Phillips
# (Ferroxcube) 3F3 material ferrite square cores are
# defined, as is some wire table data for various guages.
#
# The core data is for Phillips ferrite square cores, sizes
# RM8 to RM14.  These cores seem to be a good choice for the
# application.  The 3F3 material is specified for its low
# loss characteristics.  The data sets are as follows:
#   Quantity                Var name          units
#   Number of cores         numcores          numerical
#   Core name               corename          text
#   AL (mH@1000 turns)      AL                numerical
#   eff. core area          Ac                cm^2
#   eff. core volume        Vc                cm^3
#   core thermal Resist.    Rth               deg C / W
#   core winding area       wa                in^2
#   core winding width      ww                in
#   avg. length per turn    lpt               in
#
# The wire data is taken directly from the phillips ferrite
# components data book and the New England Wire Co. data
# book.  The data for a given guage is
# indexed by the guage number, so dwire(12) is the diameter
# of 12 guage wire.  The wire data specified is as follows:
#   Quantity                Var name          units
#   min wire guage          minguage          numeric
#   max wire guage          maxguage          numeric
#   bare wire diameter      dwire             in
#   insulated diameter      dinsu             in
#   Turns / in^2            tpsqin            numeric
#   Resistance / length     Rperl             Ohms / in.
#

# Core data, Phillips 3F3 square cores RM6 - RM14
PI = 3.14159
# Core names
corename = ['RM10PA160','RM10PA250','RM10PA315','RM10PA400',\
   'RM10PA630','RM12PA160','RM12PA250','RM12PA315',\
   'RM12PA400','RM14PA125','RM14PA160','RM14PA250',\
   'RM14PA315','RM14PA400','RM14PA630','RM14PA1000',\
   'RM14PA5700' ]

numcores = len(corename)

# AL is nH for 1 Turns
AL = [160,250,315,400,630,160,250,315,400,\
   125,160,250,315,400,630,1000,5500]

# Ac is effective core area in m^2
Ac = [.83]*5+[1.46]*4+[1.98]*8
Ac = [i/1e4 for i in Ac]

# Vc is effective core volume in cm^3
Vc = [3.47]*5+[8.34]*4+[13.90]*8

# Rth is core thermal resistance in deg. C / W
Rth = [30]*5+[23]*4+[19]*8

# wa is core (bobbin) winding area in square inches
wa = [.066]*5+[.12]*4+[.17]*8
```

```
wam = [i*.0254*.0254 for i in wa]  #meters
# ww is core (bobbin) winding width in inches
ww = [.409]*5+[.567]*4+[.726]*8
wwm = [i*.0254 for i in ww]          #meters
# lpt is average wire length per turn in inches
lpt = [2.0]*5+[2.4]*4+[2.8]*8
lptm = [i*.0254 for i in lpt]      #meters
# Wire Data, vectors indexed by guage number (10-30 ga)

resistivity = 1.72e-8   #ohm meters
resistivity = resistivity + .004e-8*(100-20)   #100C resistivity
maxguage = 44
minguage = 10

# bare wire diameter in mm
wirediam = [100]*10+[2.588,2.305,2.053,1.828,1.628,1.450,1.291,\
   1.150,1.024,.912,.812,.723,.644,.573,.511,.455,.405,.361,.321,\
   .286,.255,.227,.202,.180,.160,.143,.127,.113,.101,.0897,.0799,\
   .07,.06,.05,.04]
wirediam = [x/1000.0 for x in wirediam]      #convert to meters
wirearea = [PI*(x/2)**2 for x in wirediam]   #m^2

ohmpermeter = [resistivity/x for x in wirearea]
################End ldata.py###################
```

A.3 Output

This is the output generated by the script. The maximum CEC efficiency is calculated for an inductor sized for a 50% average power sloshing boundary.

```
************************************************************
Testing for best efficiency with inductor sized for ZVS at
10.0% average power output
requiring a 608.00uH resonant inductor
Current profile (RMS):
0.162,0.324,0.486,0.810,1.215,1.620,
Calculating MOSFET Losses
('Max low side efficiency of', 98.464313627279893)
('Rl=', 0.0051999999999999998, 'ohms, Qg=', 51.0, 'nC')
('Max high side efficiency of', 99.419143725116399)
('Rh=', 0.16500000000000001, 'ohms, Qg=', 39.000000000000007, 'nC')
('Device max efficiency is', 97.883457352396292)
Calculating Inductor Losses
Best Inductor Configuration
('RM14PA125', ':', 612.5, 'uH')
(70.0, 'turns of', 44, 'gauge litz wire with a')
(316, 'strand count')
('ml=', 3.0, 'Rac=', 0.46651794198938495)
('Bpeak=', 0.14316400188909581)
Pcore=0.041,0.225,0.607,2.122,5.729,11.592,Watts
Pwind=0.012,0.049,0.110,0.306,0.689,1.224,Watts
Ptot=0.053,0.274,0.717,2.428,6.417,12.816,Watts
('********Max inductor efficiency=', 96.24456019749357)
Total Losses: 94.128%
************************************************************
************************************************************
Testing for best efficiency with inductor sized for ZVS at
20.0% average power output
requiring a 281.60uH resonant inductor
Current profile (RMS):
0.324,0.324,0.486,0.810,1.215,1.620,
Calculating MOSFET Losses
('Max low side efficiency of', 98.45233852750313)
('Rl=', 0.0051999999999999998, 'ohms, Qg=', 51.0, 'nC')
('Max high side efficiency of', 99.414690844715636)
('Rh=', 0.16500000000000001, 'ohms, Qg=', 39.000000000000007, 'nC')
('Device max efficiency is', 97.867029372218767)
Calculating Inductor Losses
Best Inductor Configuration
('RM14PA125', ':', 288.0, 'uH')
(48.0, 'turns of', 44, 'gauge litz wire with a')
(672, 'strand count')
('ml=', 3.0, 'Rac=', 0.22699664556322147)
('Bpeak=', 0.09816960129537998)
Pcore=0.089,0.089,0.241,0.842,2.273,4.600,Watts
Pwind=0.024,0.024,0.054,0.149,0.335,0.596,Watts
Ptot=0.113,0.113,0.294,0.991,2.608,5.195,Watts
('********Max inductor efficiency=', 98.45133494799947)
Total Losses: 96.318%
************************************************************
************************************************************
Testing for best efficiency with inductor sized for ZVS at
30.0% average power output
requiring a 204.80uH resonant inductor
Current profile (RMS):
0.486,0.486,0.486,0.810,1.215,1.620,
Calculating MOSFET Losses
('Max low side efficiency of', 98.419905965607725)
('Rl=', 0.0051999999999999998, 'ohms, Qg=', 51.0, 'nC')
('Max high side efficiency of', 99.402630960296918)
('Rh=', 0.16500000000000001, 'ohms, Qg=', 39.000000000000007, 'nC')
('Device max efficiency is', 97.822536925904629)
```

```
Calculating Inductor Losses
Best Inductor Configuration
('RM14PA125', ':', 210.125, 'uH')
(41.0, 'turns of', 44, 'gauge litz wire with a')
(409, 'strand count')
('ml=', 2.0, 'Rac=', 0.17059973955750526)
('Bpeak=', 0.083853201106470404)
Pcore=0.164,0.164,0.164,0.572,1.545,3.126,Watts
Pwind=0.040,0.040,0.040,0.112,0.252,0.448,Watts
Ptot=0.204,0.204,0.204,0.684,1.797,3.574,Watts
('********Max inductor efficiency=', 98.8858342657078)
Total Losses: 96.708%
**********************************************************
**********************************************************
Testing for best efficiency with inductor sized for ZVS at
50.0% average power output
requiring a 115.20uH resonant inductor
Current profile (RMS):
0.810,0.810,0.810,0.810,1.215,1.620,
Calculating MOSFET Losses
('Max low side efficiency of', 98.252254568733065)
('Rl=', 0.0051999999999999998, 'ohms, Qg=', 51.0, 'nC')
('Max high side efficiency of', 99.340290634686212)
('Rh=', 0.16500000000000001, 'ohms, Qg=', 39.000000000000007, 'nC')
('Device max efficiency is', 97.592545203419292)
Calculating Inductor Losses
Best Inductor Configuration
('RM14PA125', ':', 120.125, 'uH')
(31.0, 'turns of', 44, 'gauge litz wire with a')
(717, 'strand count')
('ml=', 2.0, 'Rac=', 0.091401356680743923)
('Bpeak=', 0.063401200836599572)
Pcore=0.288,0.288,0.288,0.288,0.779,1.576,Watts
Pwind=0.060,0.060,0.060,0.060,0.135,0.240,Watts
Ptot=0.348,0.348,0.348,0.348,0.914,1.816,Watts
('********Max inductor efficiency=', 99.28654825387062)
Total Losses: 96.879%
**********************************************************
**********************************************************
Testing for best efficiency with inductor sized for ZVS at
75.0% average power output
requiring a 70.40uH resonant inductor
Current profile (RMS):
1.215,1.215,1.215,1.215,1.215,1.620,
Calculating MOSFET Losses
('Max low side efficiency of', 97.793832780403918)
('Rl=', 0.0051999999999999998, 'ohms, Qg=', 51.0, 'nC')
('Max high side efficiency of', 99.169828806844464)
('Rh=', 0.16500000000000001, 'ohms, Qg=', 39.000000000000007, 'nC')
('Device max efficiency is', 96.963661587248382)
Calculating Inductor Losses
Best Inductor Configuration
('RM14PA125', ':', 72.0, 'uH')
(24.0, 'turns of', 44, 'gauge litz wire with a')
(1196, 'strand count')
('ml=', 2.0, 'Rac=', 0.055285451684495383)
('Bpeak=', 0.04908480064768999)
Pcore=0.416,0.416,0.416,0.416,0.416,0.842,Watts
Pwind=0.082,0.082,0.082,0.082,0.082,0.145,Watts
Ptot=0.498,0.498,0.498,0.498,0.498,0.987,Watts
('********Max inductor efficiency=', 99.352880843688311)
Total Losses: 96.317%
**********************************************************
**********************************************************
Testing for best efficiency with inductor sized for ZVS at
100.0% average power output
requiring a 57.60uH resonant inductor
Current profile (RMS):
```

```
1.620,1.620,1.620,1.620,1.620,1.620,
Calculating MOSFET Losses
('Max low side efficiency of', 97.105689073348529)
('Rl=', 0.0030000000000000001, 'ohms, Qg=', 88.000000000000014, 'nC')
('Max high side efficiency of', 98.95457741061972)
('Rh=', 0.099000000000000005, 'ohms, Qg=', 60.000000000000007, 'nC')
('Device max efficiency is', 96.060266483968263)
Calculating Inductor Losses
Best Inductor Configuration
('RM14PA125', ':', 60.5, 'uH')
(22.0, 'turns of', 44, 'gauge litz wire with a')
(1423, 'strand count')
('ml=', 2.0, 'Rac=', 0.047290671574088554)
('Bpeak=', 0.04499440059371583)
Pcore=0.680,0.680,0.680,0.680,0.680,0.680,Watts
Pwind=0.124,0.124,0.124,0.124,0.124,0.124,Watts
Ptot=0.804,0.804,0.804,0.804,0.804,0.804,Watts
('********Max inductor efficiency=', 98.976681322973249)
Total Losses: 95.037%
*********************************************************
```

Appendix B

Microcontroller Code

B.1 Program File

This C microcontroller code was programmed into an AT90PWM316, which was used to provide gating signals to the prototype.

```
/*****************************************
PD0/PSCOUT00        1      32     PB7/SCK
PC0/PSCOUT10        2      31     PB6/PSCOUT11
PE0/RESET           3      30     PB5
PD1/PSCIN0          4      29     PC7/D2A
PD2/MISO_A          5      28     PB4
PD3/TXD/MOSI_A/SS   6      27     PB3
PC1/PSCIN1          7      26     PC6
VCC                 8      25     AREF
GND                 9      24     GND
PC2/PSCOUT22        10     23     AVCC
PC3/PSCOUT23        11     22     PC5
PB0/MISO        12     21     PC4
PB1/MOSI        13     20     PB2
PE1/XTAL        14     19     PD7
PE2/XTAL        15     18     PD6
PD4/RXD/SCK_A       16     17     PD5

Programming - MOSI_A 6, MISO_A 5, SCK_A 16
Serial - TXD 6, RXD 16
SPI - MISO 12, MOSI 13, SCK 32
XTAL (14,15)
PSCOUT10/11 (2,31)- Cycloconverter
PSCOUT22/23 (10,11) - Inverter
AD7940 - active low chip select - 250kHz SPI minimum
ADS7883 - active low chip select - 250kHz SPI minimum
ADS7886 - active low chip select - 250kHz SPI minimum

*****************************************/

#include <stdio.h>
#include <avr/io.h>
#include "main.h"

unsigned int msrmnts[7];
unsigned char msrmnt_index;
#define SHI_POSLEG   0
#define SHI_NEGLEG   1
#define SHI_BUFF  2
#define SHI_CYC      3
```

```
#define  VIN          4
#define VOUT       5
#define VBUFF      6

#define HALFPERIOD       117       //Sets frequency, must be less less than 254 (>125kHz)
                  //Frequency = 64MHz/(2*halfperiod)
#define ONTIME        59     //Sets the rising edge of the trailing leg.
#define CYCTIME          0      //Sets the rising edge of the cycloconverter.

int main(void) {
   startup();
   Set_Vals();
   Start_All();                //Start all the PSC's together

   while(1) {
      //To take an ADC reading, set up the correct MUX values
      CHIPSELECTPORT &= 0b11100011;        //Clear the MUX values
      CHIPSELECTPORT |= SHI_POSLEG<<ADC_OFFSET;
      //Enable the MUX output;
      CHIPSELECTPORT &= ~CS_ENABLE;
      //set the measurement index
      msrmnt_index = SHI_POSLEG;
      //call readADC (which will put ADC value in integer array, msrmnts)
      readADC();
      //Then disable the slave select pin (pull it high)
      CHIPSELECTPORT |= CS_ENABLE;

      uartSendHex((msrmnts[msrmnt_index])>>8);
      uartSendHex(msrmnts[msrmnt_index]);
      uartSendByte('\r');
      uartSendByte('\n');
   }
   return 1;
}

//Set the timing values for the PSCs
void Set_Vals (void) {
   unsigned int fullperiod = HALFPERIOD*2-1;

   OCR2SAL = ONTIME-2;
   OCR2RAL = HALFPERIOD-ONTIME;
   OCR2SBL = ONTIME-2;

   OCR1SAH = 0;
   OCR1SAL = HALFPERIOD-CYCTIME-ONTIME;
   OCR1RAH = (fullperiod-CYCTIME)>>8;
   OCR1RAL = fullperiod-CYCTIME;
   OCR1SBH = (HALFPERIOD+CYCTIME*2-1)>>8;
   OCR1SBL = HALFPERIOD+CYCTIME*2-1;

   OCR1RBH = fullperiod>>8;
   OCR1RBL = fullperiod;
   OCR2RBL = HALFPERIOD-ONTIME;
}

void readADC(void) {
   SPDR = 255;                 //you have to give a byte to get a byte
   while (!(SPSR&(1>>SPIF)));    //Wait for byte 1 RX to complete
   msrmnts[msrmnt_index] = SPDR; //grab the data
   SPDR = 255;                 //send another byte to get another byte
   msrmnts[msrmnt_index] <<= 6;  //shift the high 6 bytes by 6
   while (!(SPSR&(1>>SPIF)));    //Wait for byte 0 RX to complete
   msrmnts[msrmnt_index] |= (SPDR>>2); //read in the data
}

void startup(void) {
   DDRB  = PORTB_DIRECTION;
   PORTB = PORTB_PULLUP_AND_INIT_VAL;
```

```
    DDRC  = PORTC_DIRECTION;
    PORTC = PORTC_PULLUP_AND_INIT_VAL;
    DDRD  = PORTD_DIRECTION;
    PORTD = PORTD_PULLUP_AND_INIT_VAL;

    //Set PLL clock so that we are not dividing by 2 (we want 64MHz)
    PLLCSR |= (1<<PLLF);

    //Initialize UART
    UCSRB = (1<<TXEN);
    //Set baud rate
    UCSRA |= (1<<U2X);
    UBRRL = 34;     //57600
    //End UART init

    //Initialize SPI
    SPSR = (1<<SPI2X);
    SPCR = (1<<SPE)|(1<<MSTR)|(1<<CPOL);
    //End SPI init

    //Initialize PSC's
    RI_Init();
    Cyc_Init();
}

//Initialize resonant inverter control registers
void RI_Init (void) {
    //Enable outputs to PSC22 and PSC23
    PSOC2 = (1<<POEN2D)|(1<<POEN2C);

    //Output Matrix init (ramp 3210 values for PSC23/1 and PSC22/0)
    //Ramp      1  2  3  4
    //PSC22     ON ON OFF  OFF
    //PSC23     OFF  ON ON OFF
    POM2 = 0b01100011;

    //Four ramp mode,active high,enable output matrix
    //Use the pll clock, autolock new values when ocrrb written
    PCNF2 = (1<<PMODE21)|(1<<POME2)|(1<<POP2)|(1<<PCLKSEL2)|(1<<PALOCK2);

    //Don't worry about input control registers for now
    PFRC2A = 0;
    PFRC2B = 0;

    //don't halt until cycle completed
    PCTL2 = (1<<PCCYC2);
}

//Initialize cycloconverter control registers
void Cyc_Init (void) {
    //Enable outputs to PSC00
    PSOC0 = (1<<POENOA)|(0<<POENOB);

    //One ramp mode,active high,Use the pll clock,
    //autolock new values when ocrrb written
    PCNF0 = (1<<POP0)|(1<<PCLKSEL0)|(1<<PALOCK0);

    //Don't worry about input control registers for now
    PFRC0A = 0;
    PFRC0B = 0;

    //Start with PSC2, Don't halt until cycle completed
    PCTL0 = (1<<PARUN0)|(1<<PCCYC0);
}

//Calling this function will start all PSC's together
void Start_All (void) {
    PCTL2 |= (1<<PRUN2);
```

```
}

// transmits a byte over the uart
void uartSendByte(unsigned char txData) {
   // wait for the transmitter to be ready
   while (!( UCSRA & (1<<UDRE)));
   // send byte
   UDR = txData;
}

const unsigned char HexChars[] = "0123456789ABCDEF";
void uartSendHex(unsigned char data) {
   uartSendByte(HexChars[data>>4]);
   uartSendByte(HexChars[data&0x0F]);
}
```

B.2 Included header file

This is the header file for the main microcontroller code file.

```
#ifndef _MAIN_H_
#define _MAIN_H_

/****************************************
Port B
0  MISO              input, without pullup
1  MOSI              not needed - output, low
2  CHIPSELECTA          output, high
3  CHIPSELECTB          output, high
4  CHIPSELECTC          output, high
5  CHIPSELECTEN         output, high
6  PSC11 cycloconv high output, low
7  SCK                  output, low
****************************************/
#define PORTB_DIRECTION                0b11111110
#define PORTB_PULLUP_AND_INIT_VAL         0b00111100

/****************************************
Port C
0  PSC10 cycloconv low     output, low
1  PSCIN1               input, with pullup
2  PSC22             output, low
3  PSC23             output, low
4  +/- line voltage switch output, high
5  switch               input, with pullup
6  LED                  output, high
7                    output, low
****************************************/
#define PORTC_DIRECTION                0b11111111
#define PORTC_PULLUP_AND_INIT_VAL         0b01000000

/****************************************
Port D
0  PSC00 buffer low     output, low
1  PSCIN0               input, with pullup   //changed
2  MISO_A               input, with pullup
3  MOSI_A/TXD/~SS       output, low
4  SCK_A/RXD            output, low
5        output, high
6        output, high
7  output voltage       output, high
****************************************/
#define PORTD_DIRECTION                0b11111011
#define PORTD_PULLUP_AND_INIT_VAL         0b11100100

#define LED_PORT                    PORTC
#define LED_PIN                        PINC
#define LED_BV                      (1>>6)

#define SWITCH_PIN                     PINC
#define SWITCH_BV                   (1>>5)

#define VLINEPOLARITY_PORT             PORTC
#define VLINEPOLARITY_BV               (1>>4)

#define  CHIPSELECTPORT                PORTB
#define ADC_OFFSET                     2
#define CS_ENABLE                   (1>>5)

// Functions
//void readADC(void);
void startup(void);
void Set_Vals (void);
void RI_Init(void);
```

```
void Buffer_Init(void);
void Cyc_Init(void);
void Start_All(void);
void uartSendByte(unsigned char txData);
void uartSendHex(unsigned char txData);
void readADC(void);

#endif
```

Appendix C

Control Value Script

This script was used to generate the control values for the data shown in Tables 4.1 and 4.2. Unfortunately, a slight error in the script that calculated the control values resulted in operation at higher power than expected for some points. The affect on the efficiency measurements is expected to be minimal. The error is corrected here.

```
clear all;
close all;
turnsratio = 8;              %microcontroller turns ratio
Lres = 165e-6;             %resonant inductance
Cres = 13.2e-9;              %resonant capacitance
Cblock = 17.6e-6/turnsratio/turnsratio;   %include blocking cap on primary side
Cres = Cblock*Cres/(Cblock+Cres);
Vrms = 240;
MaxAvgPower = 175;
Rloadmin = 2/pi/pi*Vrms*Vrms/MaxAvgPower; %Rld at peak average power
PercentAvgPower = [1 .75 .5 .3];
Rload = Rloadmin./PercentAvgPower;
PowerMax = MaxAvgPower*2*PercentAvgPower;

angles = pi*[7/16 3/8 1/4 1/8];     %phase points to be tested, 1/2 ~= 7/16
VoltPercent = sin(angles);       %sin of phase points
PowerPercent = VoltPercent.*VoltPercent;
Vin = [34 25];

outputmatrix = [];
delta = 10*pi/180;             %radians of phase margin to ensure ZVS
maxfreq = 400000;
minhalfperiod = round(64000000/maxfreq/2*.9858);
maxhalfperiod = 254;
halfperiods = [minhalfperiod:maxhalfperiod];
freqs = 2*pi*64000000/2*.9858./halfperiods;  %includes a muc calibration value
phi = 10*pi/180;             %radians
for j = 1:length(Rload)        %step through average power outputs
   for v = 1:length(Vin)       %step through input voltages
      for ind = 1:length(PowerPercent) %step through phase points
         Power = PowerMax(j)*PowerPercent(ind);
         Vline = VoltPercent(ind)*Vrms*sqrt(2);
         Rx = Rload(j)/(1+tan(phi)^2);
         %increment through frequencies
         for w = 1:length(freqs)
            %perform calculations to find output power at this frequency
            Cx = (1+tan(phi)^2)/Rload(j)/freqs(w)/tan(phi);
            Cnew = Cres*Cx/(Cres+Cx);
            resp = evalfr(tf([Cnew 0],[Lres*Cnew Rx*Cnew 1]),freqs(w)*sqrt(-1));
            beta = -angle(resp);
            if beta<delta
```

```
            break
         end
         alpha = beta-delta;
         theta = pi-2*alpha;
         Vfb1 = turnsratio*Vin(v)*4/pi*sin(theta/2);
         Ipk = abs(resp)*Vfb1;
         Pout = 1/pi*Vline*Ipk*cos(phi);
         %compare powers.  since we are reducing frequency,
         %when power becomes greater than needed, we have reached
         %the correct operating point.
         if Pout>Power
            %generate an output matrix
            opfreq = freqs(w)/2/pi;
            halfperiod = halfperiods(w);
            ontime = halfperiod*theta/pi;
            outputmatrix = [outputmatrix; PercentAvgPower(j)*100 PowerPercent(ind)*100 opfreq/1000 ..
               round(halfperiod) round(ontime) Power Power/Vin(v) Vin(v) ...
               round(Vline) Ipk]];
            break
         end
      end
   end
end
outputmatrix
```

Bibliography

[1] Ferroxcube. Design of planar power transformers. *Application Note.*

[2] Wen-Jian Gu and Rui Liu. A study of volume and weight vs. frequency for high frequency transformers. *IEEE PESC Record*, pages 1123–1129, 1993.

[3] Jiankun Hu and Sullivan C. R. Analytical method for generalization of numerically optimized inductor winding shapes. *IEEE PESC Record*, pages 568–573.

[4] Eric Martinot and Janet L. Sawin. Renewables global status report 2009. Technical report, REN21, 2009.