Phase-Controlled Rectification for Permanent-Magnet Downhole Alternators

by

Patrick L. Maher

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

February, 2008

©2008 Massachusetts Institute of Technology
All rights reserved.

Author_____
Department of Electrical Engineering and Computer Science
February 1, 2008

Certified by_____
Dr. Albert Hoefel
VI-A Company Thesis Supervisor

Certified by_____
Dr. David Perreault
M.I.T. Thesis Supervisor

Accepted by_____
Arthur C. Smith
Professor of Electrical Engineering
Chairman, Department Committee on Graduate Theses

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
NOV 13 2008
LIBRARIES

ARCHIVES

Phase-Controlled Rectification for Permanent-Magnet Downhole Alternators

by

Patrick L. Maher

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

Abstract

This thesis develops a method of phase-controlled rectification to be used with permanent-magnet downhole alternators. The design faces unique challenges imposed by the downhole environment. The thesis discusses the design, simulation, and construction of the rectifier. A high-side semi-bridge phase-controlled model is proposed. The model is verified in simulation. A prototype rectifier was built and tested using downhole certified components. A controller was designed using the Spectrum Digital TMS320F2812 board. The prototype met all design objectives. The thesis also proposes ways to continue future work.

VI-A Company Thesis Supervisor: Dr. Albert Hoefel
Title: Principal Engineer

M.I.T. Thesis Supervisor: Dr. David Perreault
Title: Associate Professor

Acknowledgements

There are many people without whom this project would have had no chance of successful completion. Chiefly, thanks to Albert Hoefel for his assistance, patience, and expertise in helping to design the rectifier circuit. Without his guidance and knowledge of practical circuit problems this project would not have been possible. Vielen Dank! I would like to thank Professor Dave Perreault of MIT for his assistance in supervising my thesis for MIT, and for his uncommon patience. If only I had sought his advice earlier, my project would have gone much more smoothly!

I'd also like to thank the engineers at Schlumberger for their assistance throughout the project. In particular, Hyung Joon Kim provided frequent help with power circuit design and equipment problems. A very special thanks to Hoang Do for constructing my prototype system, and for having the patience to switch out the many IGBTs that I destroyed. Thanks to Peter Swinburne for his help on the software side.

Thanks to Kathy Sullivan, the MIT VI-A coordinator, and Luis Parra, the Schlumberger VI-A coordinator who made this opportunity possible. This was a terrific learning experience for me, and I am deeply grateful for your efforts.

Finally, I'd like to thank my mother and father for their support, patience, and gentle prodding throughout my life. Without them I would never have made it to MIT. Mom and Dad- thanks for all everything!

Contents

Chapter 1

Introduction

1.1 Background Information on Schlumberger Technology

In oil drilling systems there is often a need for electrical power downhole. Current Schlumberger downhole industrial applications use turbo-alternators for high power levels. The turbo-alternators provide substantial AC power. Turbo-alternators have a wide speed range, which varies with flow speed and load. From this a problem arises: flow and load changes cause a variation of the alternator output voltage. However, Schlumberger technology needs a well regulated output voltage to control tools.

Schlumberger is interested in improving their alternator technologies. Schlumberger wants to use less expensive and smaller alternators. To make these new alternators feasible, Schlumberger needs to develop electronic control circuitry.

1.2 Project Objectives

The objective of this project is to evaluate and test options for electrically controlling new alternators. The goal is to implement a buck or boost function electronically. This control functionality will be achieved through the use of a controlled rectifier circuit.

This project was sponsored by the Schlumberger Concept Project Métier. The goal of this project was not to design the actual controller that would be used in a new tool. The goal was instead to design a prototype that would prove the feasibility of the proposed design. The prototype would later serve as a basis for the design of the production system.

Because Schlumberger's technologies are utilized in an atypical environment, designs must meet many criteria not often considered. In addition to the usual concerns about efficiency, size, and cost, Schlumberger's downhole technologies face extreme temperature and pressure

conditions. Downhole technologies must typically operate at temperatures near 160°C, though temperatures can rise even higher [1.1]. The solution must operate properly at these temperatures. In addition, the desired solution will be able to control the alternator output voltage with low voltage ripple. Finally, the design must avoid hot spots and high voltage or high current transients, which could damage expensive components.

Because of the harsh downhole environment mentioned above, many typical components cannot be used downhole. Downtime on the job is extremely expensive; therefore, Schlumberger will not use components that have not been thoroughly tested for resilience to temperature, pressure, and shock. Because the qualification process takes a long time, parts selection is limited. To balance these many constraints, Schlumberger wanted to investigate the feasibility of a three-phase rectifier with high-side control. The simplicity of this solution makes it extremely desirable.

This thesis will present the three-phase rectifier with high-side control that I designed for Schlumberger. The three-phase rectifier has been designed and simulated, and a prototype has been tested at Schlumberger. This thesis will discuss the design process, simulation, construction, testing and results. Chapter two contains the background needed to understand the proposed design. Chapter three discusses the proposed design and operation of the high-side controlled rectifier. Chapter four provides simulation results and analysis to support the proposed design. Chapter five details the construction of the prototype. Chapter six gives results achieved with the prototype. Finally, chapter seven discusses possible directions for future work.

1.1 Drilling in extreme environments. Downloaded from SLB.com on October 1, 2007. Available at: http://www.seed.slb.com/en/scictr/watch/joides/drilling3.htm

Chapter 2

Background

2.1 Review of three-phase diode rectifier and phase control rectifier topologies.

Rectifiers are commonly used to provide DC output voltage from an AC source. A schematic of a basic six-pulse diode rectifier with inductive filtering is shown in figure 2-1. The AC voltages applied across the lines labeled 'a' 'b' and 'c' are rectified to provide a near DC voltage across the load. Figure 2-2 shows the line-to-line voltages and the unfiltered rectifier voltage [2.1]. A constant current load diode rectifier gives a DC output voltage of approximately:

$$V_{DC1} = \frac{3}{\pi} * \sqrt{2} * V_{LL} \qquad \textbf{Equation 2-1}$$

where V_{LL} is the rms value of the line-to-line voltages. A complete discussion of rectifier operation can be found in many power electronic textbooks (e.g., [2.1]).

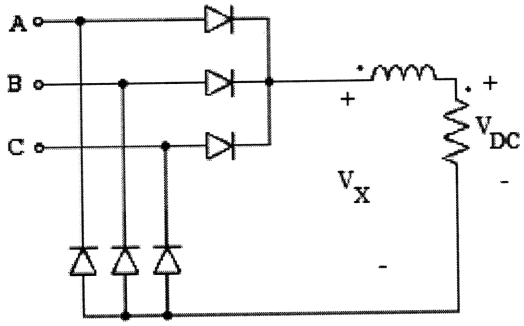

Fig 2-1: Six-Pulse Diode Rectifier

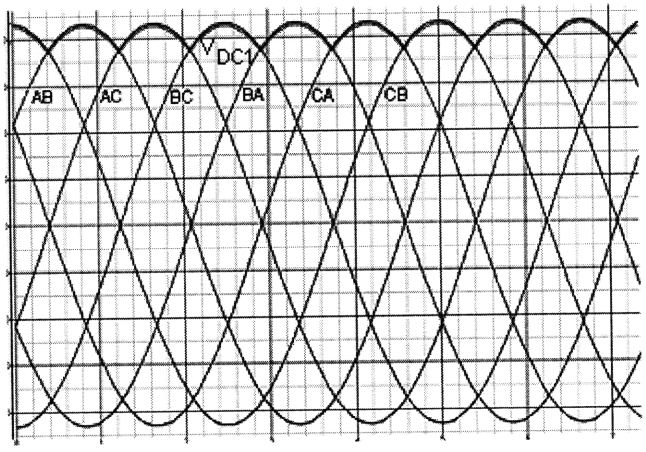

Fig 2-2: Six-pulse diode rectifier waveforms. The labeled traces are the line-to-line input voltages. The load voltage follows the peaks of the line-to-line voltages.

Although the diode rectifier successfully converts AC voltages to DC voltages, the basic diode rectifier does not provide any control over the rectified voltage magnitude. Fortunately, numerous controlled rectification circuit topologies can be used to control output voltages. One such controlled circuit is the phase-controlled rectifier.

If we replace the diodes of the basic rectifier with controllable devices we have a phase-controlled rectifier. A full-bridge phase-controlled rectifier using thyristors can be seen in figure 2-3. The significance of phase control is that it gives the user control over the alternator output voltage. Using switches, only a portion of the source voltage may be applied to the load. By applying a smaller portion, one can lower, or buck, the output voltage. An example of unfiltered output waveforms can be seen in figure 2-4 (2.2).

For a phase-controlled six pulse rectifier with constant current load the average DC output voltage is given in equation 2-2, where α is the firing angle (2.2). As compared to the basic diode rectifier with output from equation 2-1, we can see that the average DC output voltage has been reduced by a factor of $\cos(\alpha_f)$ (2.2).

$$V_{DC2} = \frac{3}{\pi} * \sqrt{2} * V_{LL} * \cos(\alpha_f) = V_{DC1} * \cos(\alpha_f) \qquad \textbf{Equation 2-2}$$

The above discussion concerned inductive filtered rectifiers. This discussion was included to provide some background and intuition as to how basic rectifiers work. Inductive filtered rectifiers are easy to analyze, and have been thoroughly discussed in the literature. The remainder of this paper will propose a variant of a phase-controlled rectifier with a capacitive filter at the load. Capacitively filtered rectifiers, also known as constant voltage load rectifiers, are more difficult to analyze. A discussion of capacitive filtered diode rectifiers and phase-controlled rectifiers can be found in Caliskan [2.3, 2.4].

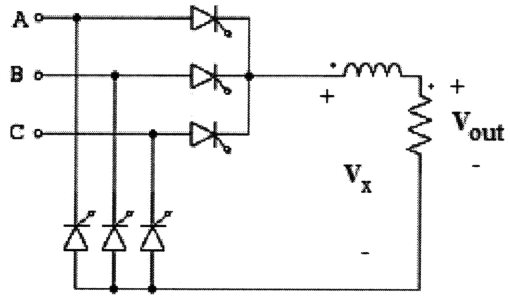

Fig 2-3 Phase-controlled rectifier with thyristors and inductive filtering of the load.

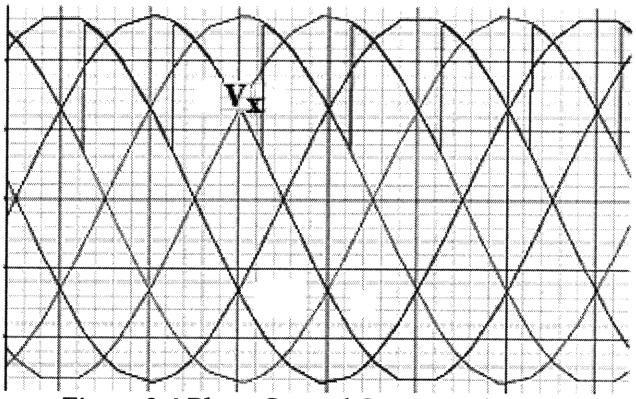

Figure 2-4 Phase Control Output: The six line-to-line voltage sine-waves are shown. The phase control load voltage is superimposed. The load voltage follows the line-to-line voltage until a new switch is turned on.

2.1 Mohan, N., T.M. Undeland, and W.P. Robbins, *Power Electronics: Converters, Applications, and Design,* 2nd edition, John Wiley, 1995.

2.2 Trzynadlowski, Andrzej M. Control of Induction Motors. Boston: Academic Press, 2001. Chapter 4.

2.3 V. Caliskan, D.J. Perreault, T.M. Jahns, and J.G. Kassakian, "Analysis of Three-Phase Rectifiers with Constant-Voltage Loads," 1999 IEEE Power Electronics Specialists Conference, Charleston, SC, pp. 715-720, June 1999.

2.4 V. Caliskan, D.J. Perreault, T.M. Jahns, and J.G. Kassakian, "Analysis of Three-Phase Rectifiers with Constant-Voltage Loads," *IEEE Transactions on Circuits and Systems – I,* Vol. 50, No. 9, Sept. 2003, pp. 1220-1226.

Chapter 3

Design and Operation of the Rectifier

3.1 Specifications and Design Constraints

The goal of the project is to control the output of a permanent magnet alternator driving a resistive load. The alternator characteristics are shown in figure 3-1. The output will be capacitively filtered with a 300uF capacitor bank. To meet load specifications, the output must be controlled within the operational range of 175-350V with a load of 0-7A.

To rapidly demonstrate the principle with a prototype, the feasibility of controlling the system with only high-side phase control was investigated. IGBTs were selected for the high-side switches.

Figure 3-1 shows a schematic of the proposed circuit design. Throughout the design and simulation stages the alternator was modeled as a perfect three-phase sinusoidal voltage generator plus a phase inductance and phase resistance. This alternator model is schematically depicted in figure 3-1. The values utilized for the alternator source inductance and resistance are listed in table 3-1.

Poles	8
Mechanical Speed Range (rpm)	2000---5500
Mechanical Speed Range (Hz)	33.3---91.6
Electrical Speed (Hz)	133---370
Phase Resistance at 25°C (ohms)	1.106
Phase Inductance (mH)	3.133
Voltage Constant phase-to-neutral (V/krpm)	123

Table 3-1: The data in this table summarizes key alternator characteristics used for design and simulation.

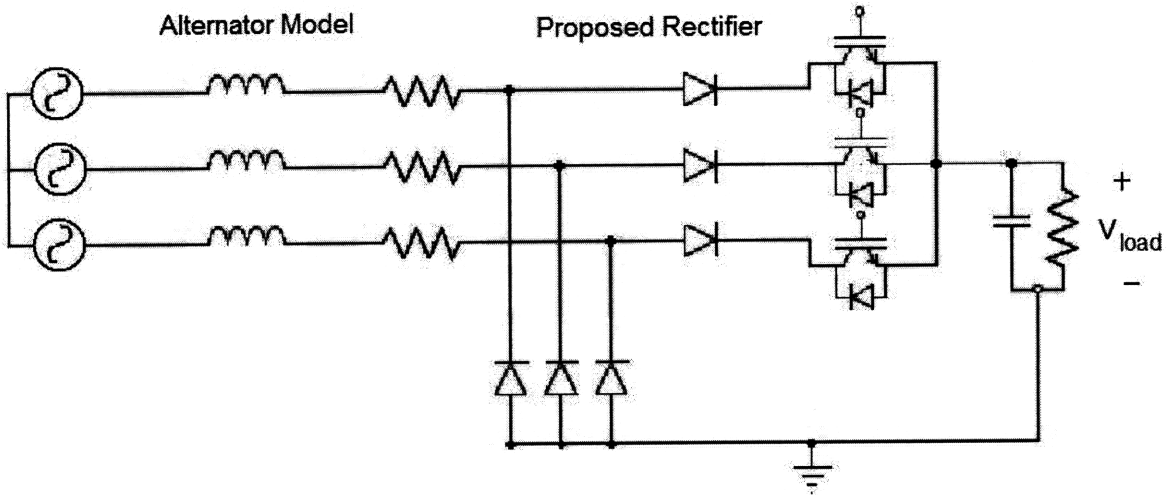

Figure 3-1: Simulation Schematic. The sources on the left are three sinusoids separated by 120 degrees in phase representing the alternator voltage. The source inductance and resistance is shown. Six high voltage diodes are used to create the bridge, in addition to the three phase-controlled IGBTs on the high side.

3.2 Description of Rectifier Functionality

The basic operation of the high-side controlled three-phase rectifier can be likened to the operation of the three-phase diode rectifier, except for one major difference; in the high-side phase controlled rectifier, the given phase cannot conduct until the IGBTs are turned on. The high-side controlled three-phase rectifier must have the highest phase voltage among those with their corresponding IGBT on.

The goal of the project is to have a controllable rectifier that can be used to set the desired load voltage. The remainder of section 3.2 will present an intuitive description of how the high-side phase control system operates. This description will ignore some issues related to switching currents. These issues will be discussed more thoroughly in section 3.3.

The phase controlled rectifier gives us control over the load voltage. The IGBT turn-on delay relative to the phase voltages may be used to control the load voltage. By delaying the switch turn on time, the output voltage can be reduced. A controller will monitor the alternator voltage and the load voltage to determine the appropriate time to turn on the IGBT. After

activating the switch, the switch will remain on until it is safe to turn off, at which time an off signal will be sent.

To make the above explanation clearer figures 3-2, 3-3, and 3-4 depict the system behavior. Figure 3-2 shows alternator waveforms for the ideal case. This figure was included to demonstrate how the phase-to-phase and phase-to-neutral voltages compare. Since both voltages will be important in this system, it is essential to have a clear understanding from the outset. The phase-to-phase voltages are $\sqrt{3}$ times larger than the phase-to-neutral voltage. In addition, the phase-to-phase voltages are shifted relative to the phase-to-neutral voltage. Finally, the phase-to-phase voltages exhibit six pulses (or peaks) per AC cycle.

Figures 3-3 and 3-4 depict typical steady state operation of the high-side-controlled three-phase rectifier. Figure 3-3 shows the phase-to-neutral voltages, denoting the three phases with letters a, b, and c, and neutral with the letter n. Figure 3-3 also shows an example set of IGBT gate drive waveforms, staggered on different axes for ease of viewing. Figure 3-4 contains everything in figure 3-3, plus it also shows a series of current pulses delivering power from source to load.

As can been seen in figure 3-3, a switch is turned on sometime after its respective phase-to-neutral becomes positive, and the switch is turned off sometime after the respective phase-to-neutral voltage becomes zero. Because of the source inductance, some special considerations must be made to switch turn off delay. While a given IGBT is on, and the phase is conducting, the source inductor is storing energy. If an IGBT is shut off while the inductor is still conducting positive current, the inductor will try to maintain the current, thereby destroying the switch. To prevent this, the IGBT gate signal is kept high even after the associated phase voltage is no longer the highest.

During operation, current will be drawn from the source to power the load and charge the filtering capacitor. Figure 3-4 depicts the IGBT control process, and shows a current pulse from each of the three IGBTs. As can be seen in figure 3-4, when a gate signal goes high, a current pulse is immediately transmitted. By delaying the IGBTs turn on time, the time for which the source voltage is applied to the load decreases. The average current delivered to the capacitor will decrease, and the output voltage will correspondingly decrease. This effectively bucks the average output voltage. As was explained above, figure 3-4 show the switch gate signal doesn't go low until after the current has gone to zero. This is done to make certain that there is no residual current through the phase as a result of the phase inductance.

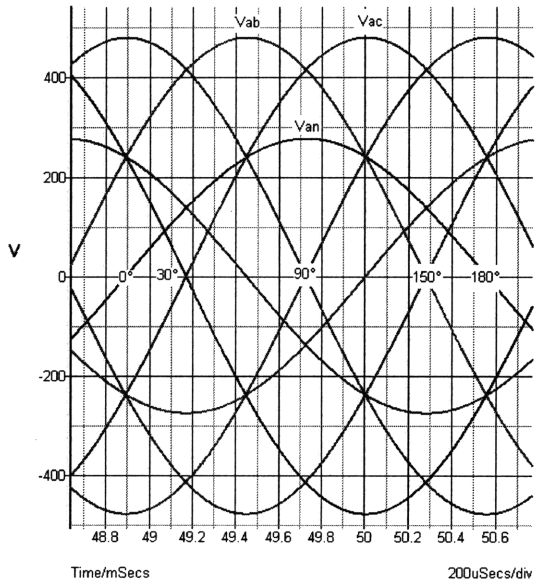

Figure 3-2: Line-to-line voltage and line-to-neutral voltages.

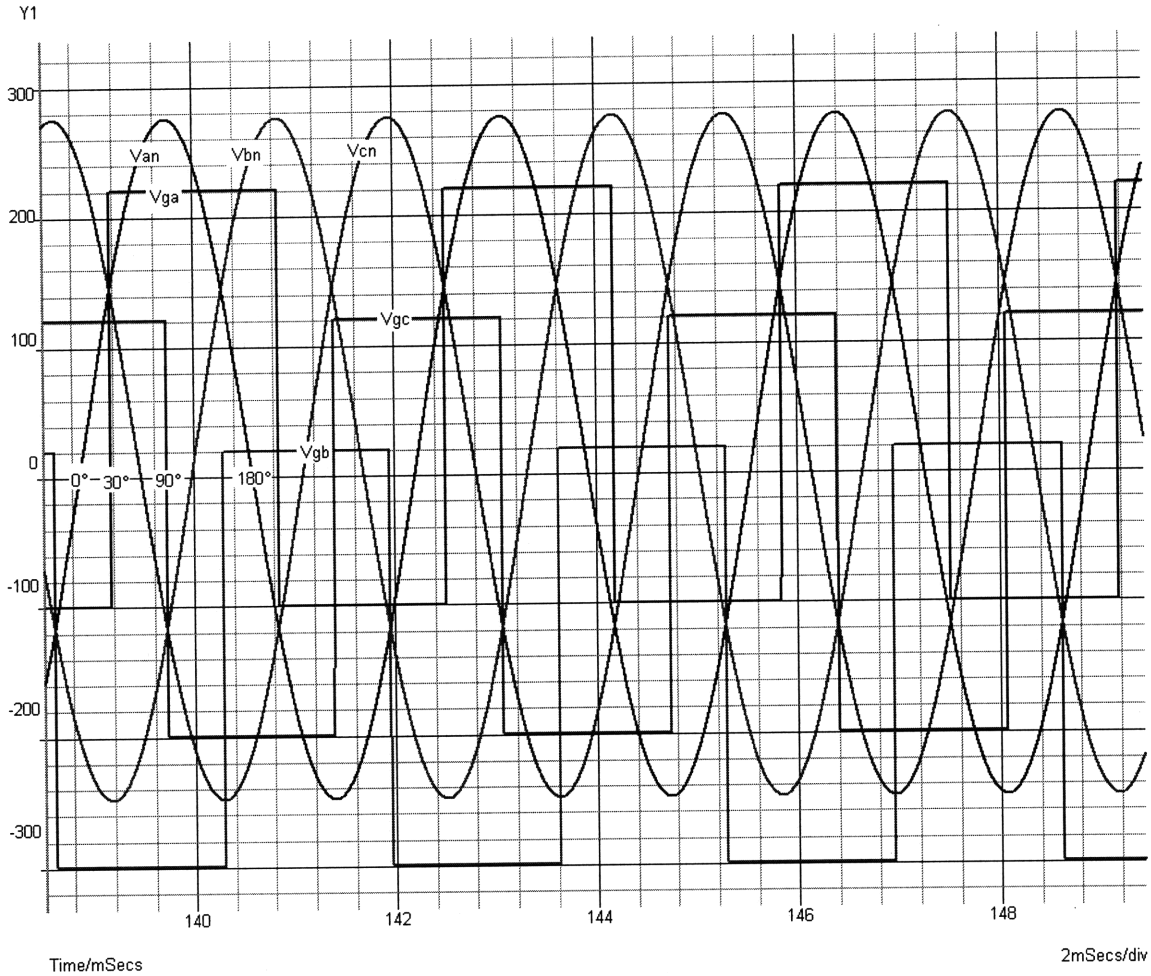

Figure 3-3: The IGBTs can be turned on to apply the source voltage to the load. The phase-to-neutral voltages are taken from before the source inductor and resistor, and therefore appear as perfect sinusoids. The gate signals have been plotted on different axes to make them easy to read. After a phase crosses to become high, the associated gate signal can be raised to turn on the IGBT. In the example shown, the firing angle is 30 degrees, or right at the crossover point between phase voltages.

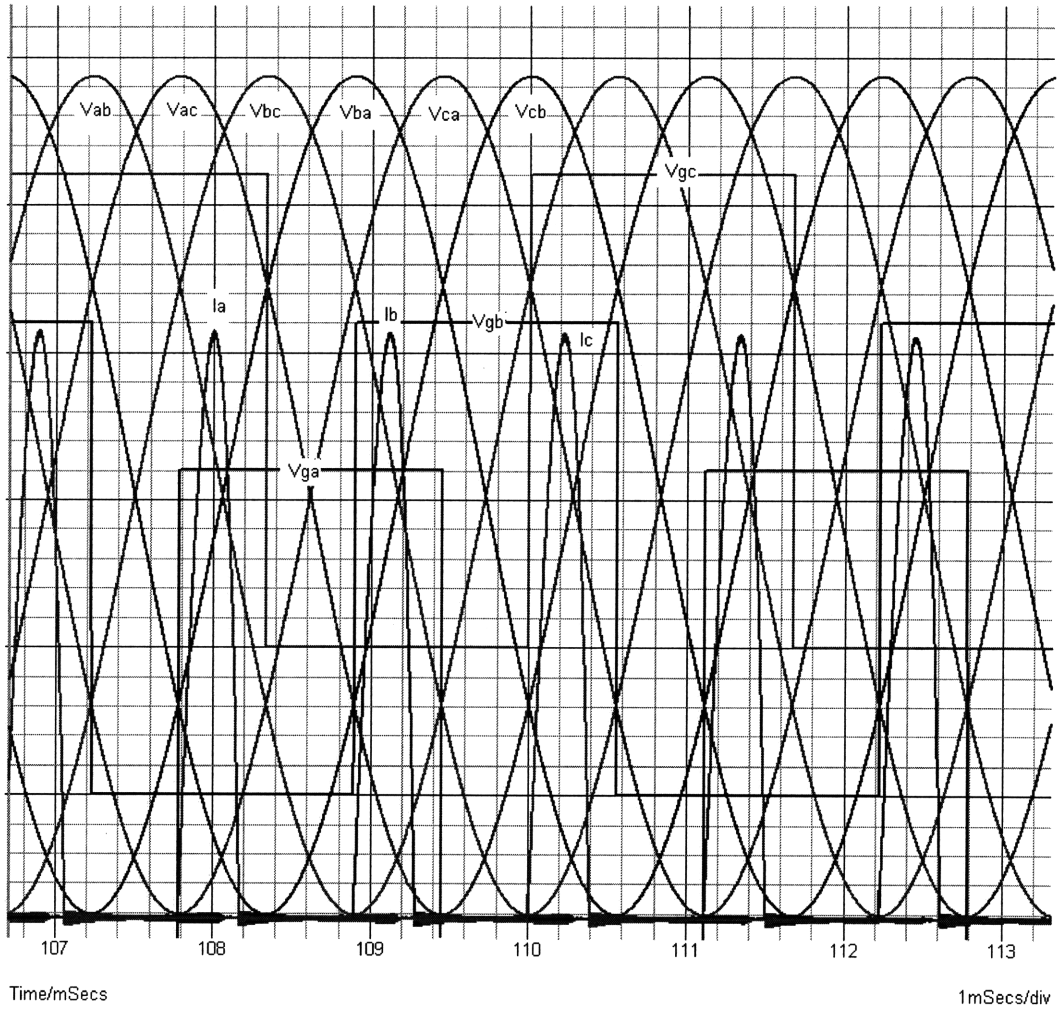

Figure 3-4: Phase Current Waveforms. The six phase-to-phase voltages are shown for reference. The system is run with a firing angle of 120 degrees. Since each phase is highest when its IGBT is turned on, a current pulse is immediately transferred to the load.

3.3 Supplemental discussion of current waveforms

In a diode rectifier with a constant current load, the load draws continuous current from the source. One phase is ceasing to conduct while another is beginning. While the currents are changing, the (AC-side) source inductance will act to preserve currents. The source inductance thus causes a non-zero commutation time (for a more detailed description see [3.1]).

The high-side phase control with constant voltage load system draws pulse-like currents from the source. The currents drawn are discontinuous. An example of the current pulses with a firing angle of 120 degrees can be seen in figure 3-5. The current pulses are drawn from a phase when the respective phase voltage is highest and the associated IGBT is on.

With firing angle less than 90 degrees we would expect to see two current pulses through each IGBT per switching cycle. With a firing angle greater than 90 degrees, we only see one pulse per IGBT per cycle. This can be understood as follows: Let's assume V_{ab} peaks and then V_{ac} peaks. The transition of peaks occurs at 90 degrees. If the IGBT is turned on after the first peak, a current pulse cannot be driven by this peak. This effect can be seen in figure 3-5, where the firing angle is 120 degrees. Figure 3-6 shows current pulses for a firing angle of 60 degrees. Figure 3-7 shows current pulses for a firing angle of 30 degrees. In both figure 3-6 and figure 3-7 we see peaks twice as frequently as in figure 3-5. Also worth noting is that since V_{ab} peaks at 30 degrees, the system could not drive a current before 30 degrees. Consequently, operation with a phase angle between 0 and 30 degrees is identical to the thirty degree case. As we can see in figure 3-7, the discontinuous, one phase at a time, current waveform assumption is still valid.

It is important to note that due to source inductance the current pulses do not decay to zero as rapidly as they otherwise would. However, as can be inferred from examining simulation and prototype results, the effect of the source inductance is small in the system described here.

Non-zero currents are not maintained for long. Since there is significant time between the end of the current pulse and the start of the next cycle there is ample time in which to safely shut off the switches.

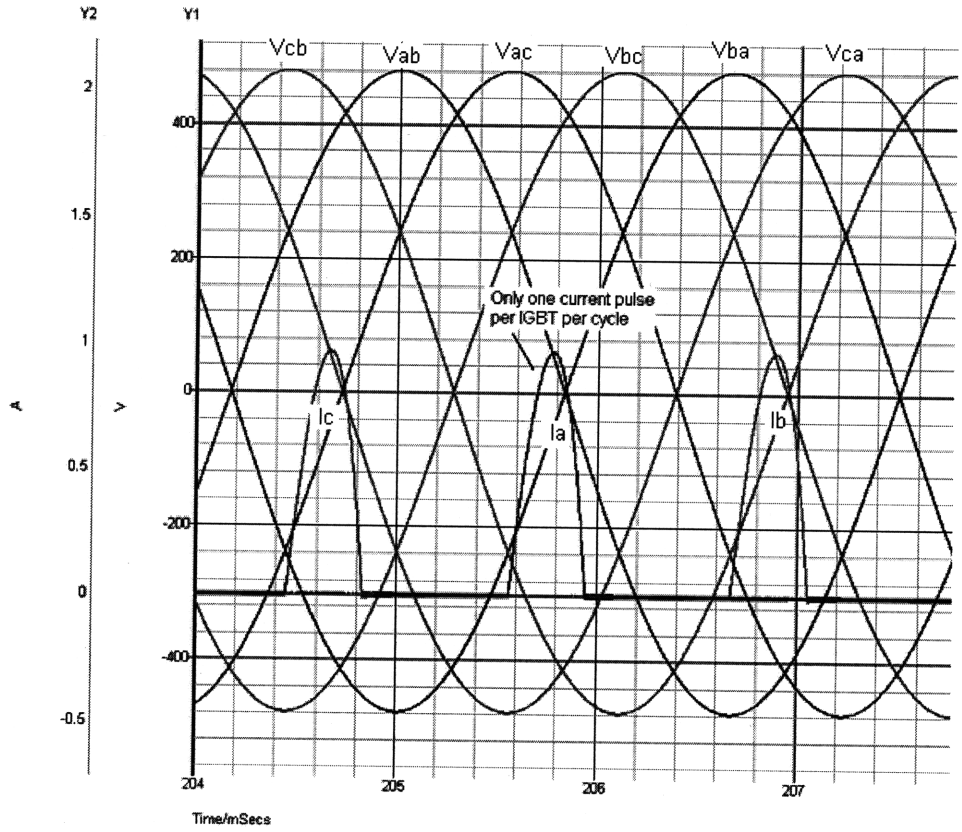

Figure 3-5: Firing angle 120 degrees. Only one current pulse is transmitted per device per cycle. Because the IGBT is turned on too late, the first voltage peak of the phase cannot drive a current pulse. The line-to-line voltages are shown for reference.

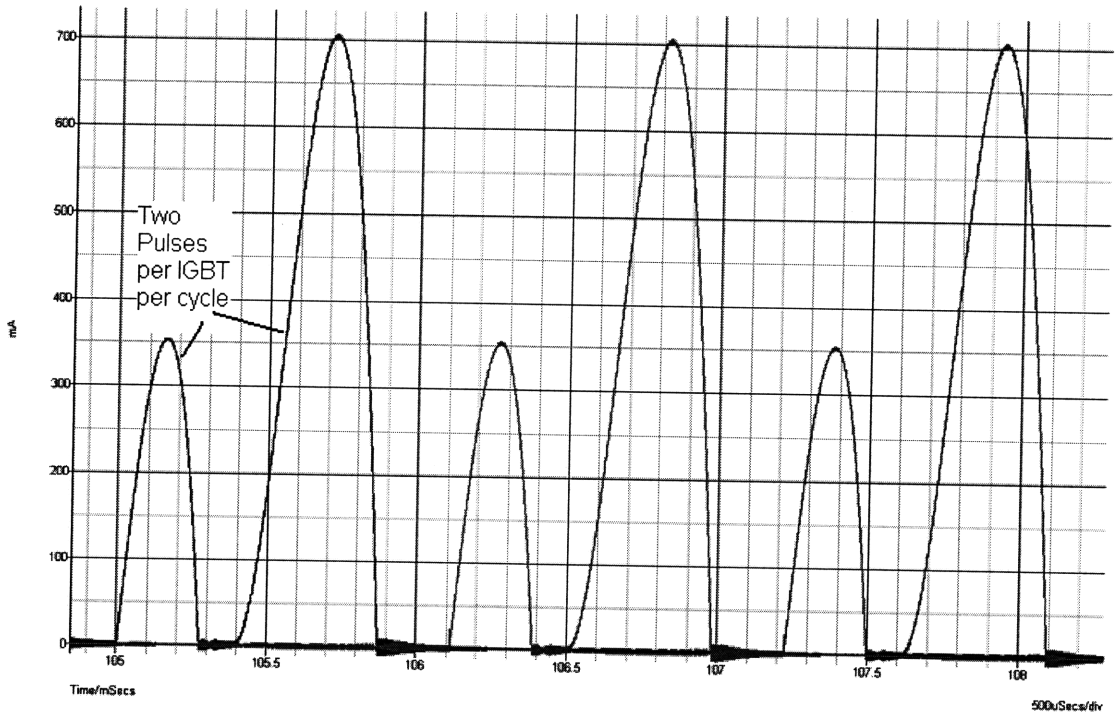

Figure 3-6: Firing angle is 60 degrees. The pulses are asymmetrical because the IGBT is off for part of the first voltage peak.

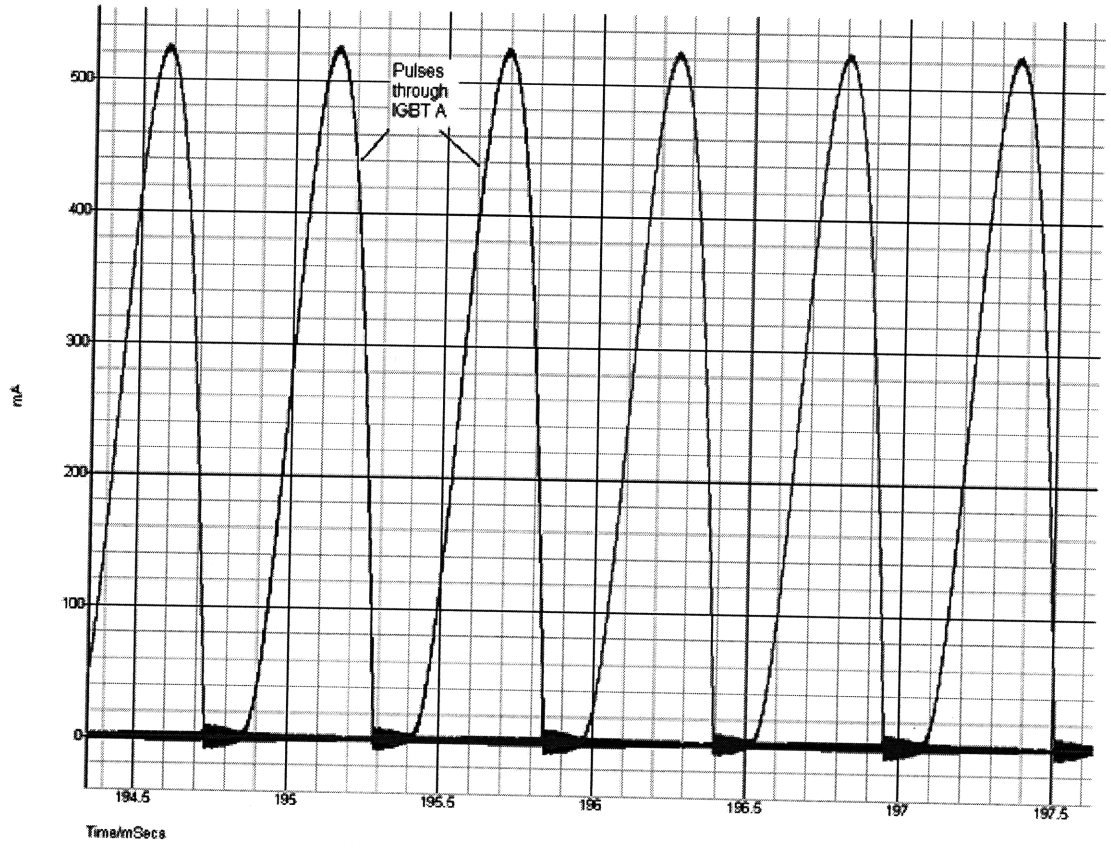

Figure 3-7: Firing angle 30 degrees. Since the IGBT is on for both full cycles, the pulses are symmetrical.

3.1 J.G. Kassakian, M.F. Schlecht and G.C. Verghese, Addison-Wesley, 1991. Chapter 4.3.2 Pgs. 69-73.

Chapter 4

Simulation Results and Design Verification

4.1 Simulation Design

To verify the functionality of the proposed system, simulations were completed. Simulations were conducted using two different software packages: Catena's SIMetrix power electronics simulation package [4.1], and Powersim Inc.'s PSIM simulator [4.2]. Both of these packages are excellent programs with unique benefits. In addition, both of these packages have limited versions available freely online.

Figure 4-1 is a screen capture of a model used in SIMetrix. This model was used to simulate the rectifier performance. In the model, the left-most sources represent a perfect three-phase sinusoidal voltage source. These sources are followed by phase inductors and resistors. Together these elements provide a good approximation of the alternator performance. The rectifier software SPICE models for the diodes and IGBTs are included in an appendix [4.3].

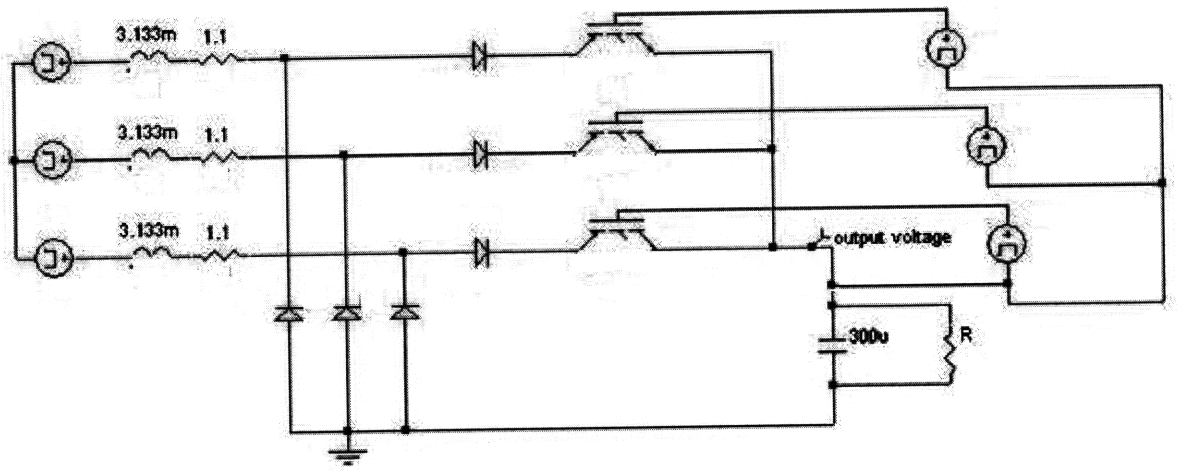

Figure 4-1 Screen capture of simulation circuit used in SIMetrix to generate data for figure 4-2

For the diodes, a SPICE model of BYT30P-1000 diodes is used for the model [4.3]. These are 1000V diodes (The actual prototype used 1200V diodes). The load is approximated by a resistor with capacitive filtering. The right-most sources, which are driving the IGBT gates, provide and idealized model of the gate-drive behavior. The output voltage is also indicated.

4.2 Simulation Results

Throughout this discussion, phase angle will be referenced from the phase-to-neutral voltage. Zero degrees is taken to be when the phase-to-neutral voltage crosses zero volts (See Figure 4-2). Figure 4-3 shows the simulated steady state voltage as a function of phase firing angle for different parameters. Each successive point depicts the output voltage for a 20 degree phase angle change. Results are shown for three different alternator frequencies each with three different load conditions. The results in figure 4-3 show an important fact about the system. As expected, output voltage monotonically decreases as phase firing angle is increased. This makes sense: as we apply a lower portion of the source waveform, the average load voltage should decrease. It is also worth noting that at some angles the output voltage changes little, while at other angles the change is many volts.

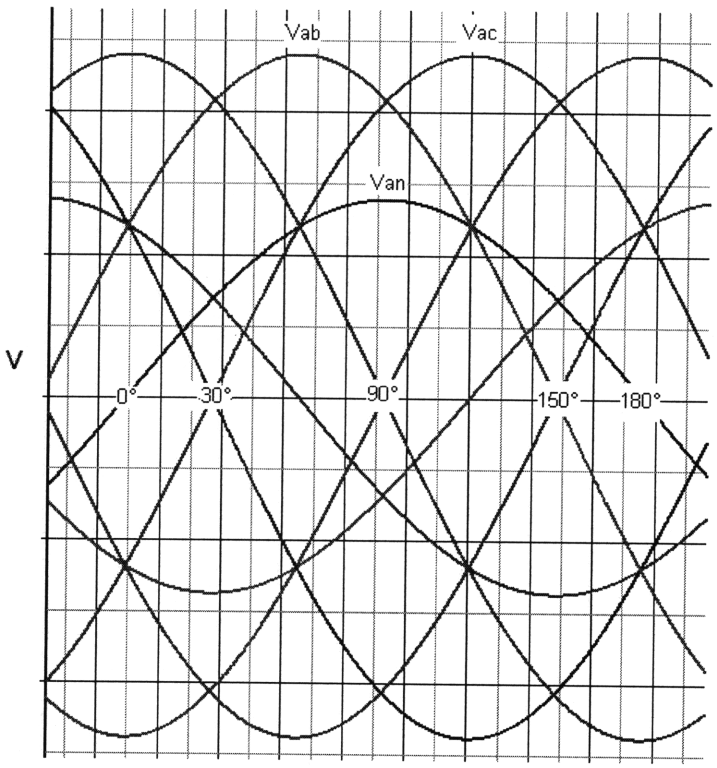

Figure 4-2: Graphical depiction of how phase angles are referenced for phase 'a'.

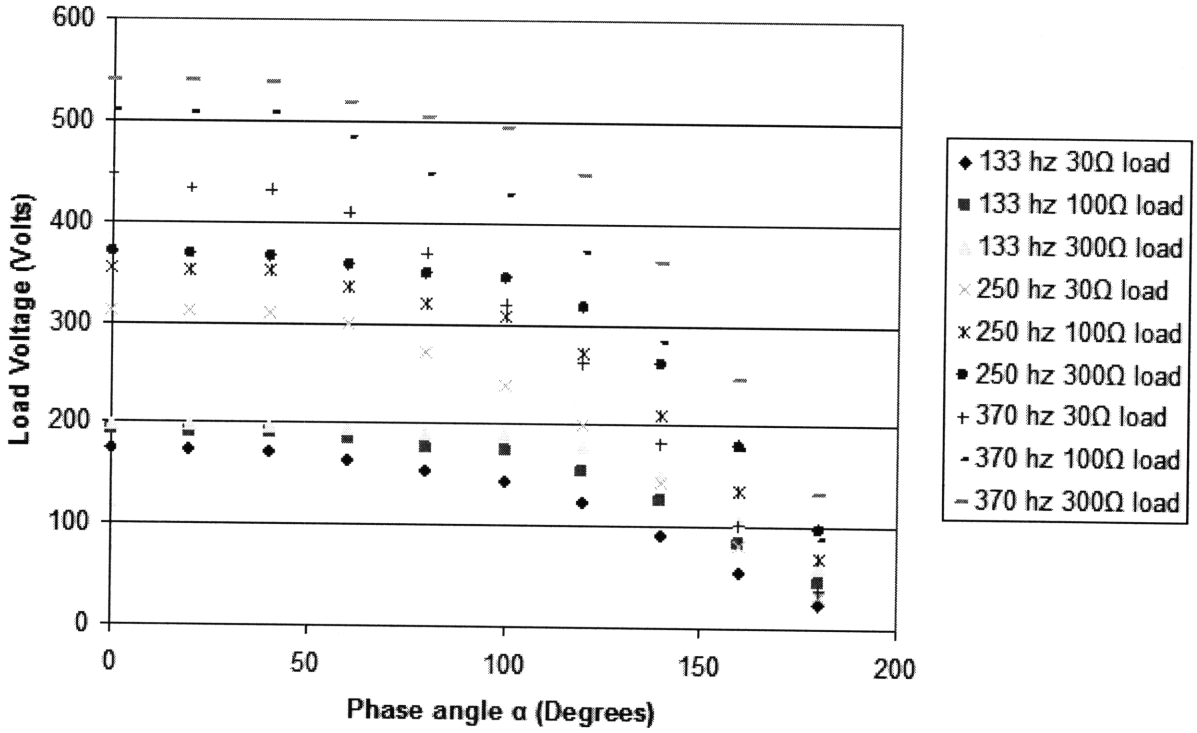

Figure 4-3: Graph of steady state output voltage for different test cases. Phase angle is increased by constant 20 degree increments from 0-180 degrees. As phase angle is increased, output voltage decreases. As expected, the relationship is highly nonlinear.

Two fundamental questions needed to be answered before considering phase control: First, can manipulation of phase maintain the desired output voltage range? Second, will current waveforms be small enough to prevent damaging components? These two questions are addressed in the following paragraphs.

The output voltage can be satisfactorily controlled by manipulation of phase. Figure 4-3 shows output results of simulations at various frequencies, loads, and firing angles. At the lowest operating frequency (133Hz) the output is still within tolerable constraints (> 175V). At these frequencies, operating like a diode rectifier is sufficient. At the higher voltages and frequencies the alternator output voltage must be substantially attenuated to meet the specification. By utilizing phase control, the high-frequency high-voltage source cases can also be controlled.

Chart 4-1 shows additional simulation data for higher frequency test cases. The chart shows peak switch currents, peak-to-peak load ripple voltage, and DC load voltage as frequency, voltage, and load resistance were varied. For each frequency, the load resistances were selected to give three different DC load currents. The chart includes data for higher frequency cases than will be tested with the prototype. Even for these extreme cases the design can meet the required constraints. (The chart also includes some data for higher loads than depicted in figure 4-3. If the future load range is increased, the system will still function correctly.) The chart shows important results. In all cases, the output voltage was controllable below the 350V upper limit. Furthermore, in all cases, output voltage ripple was reasonably small. Since the system can be controlled at both extremes, and since all intermediate values are constrained between the extremes, the system can be controlled at all frequencies of interest.

In answer to the second question above, current waveforms are small enough to protect components. Chart 4-1 includes the data that verifies this result. The peak currents observed were less than 10A. The prototype system will use 1200V diodes which can withstand continuous currents of 30A, or repetitive peak currents of 60A at 175°C. The IGBTs can withstand a continuous collector current of 39A or a pulsed current of 280A at 100°C. Although this low temperature tolerance may not work in the actual high temperature device, it is sufficient for the prototype developed here. Also, in the worst case, the CE voltage on the IGBT was 400V, which is well below the 600V tolerance.

Mechanical rpm	Electrical f (Hz)	$V_{in\ p\text{-}n}$ (V)	$I_{\_IGBT\ p}$ (A)	DC Output V_o (V)	Ripple ΔV_o (V)	Phase	R_{load} (Ω)	DC Iload
3000	200	369	0.6	300	0.6	112.5	2000	0.15
			3.5	277	3	62	150	1.846667
			6.9	254	3	60	50	5.08
3750	250	461.25	0.6	343	0.3	127.5	4000	0.08575
			4.2	335	3.5	82.5	175	1.914286
			6.5	324.5	3	60	70	4.635714
4500	300	553.5	1.5	310	1	150	1000	0.31
			3.9	343	2	127.5	250	1.372
			7.6	328	4	112.5	90	3.644444
5000	333.3333	615	0.35	349	0.1	160	10000	0.0349
			4.5	333	2	135	200	1.665
			7.4	342	2	120	100	3.42
5500	366.6667	676.5	0.5	342	0.1	165	10000	0.0342
			4.8	321	2	140	200	1.605
			7.75	350	4	125	100	3.5

Chart 4-1: Simulation Data

4.3 Controller Design

The results demonstrated above are promising. Simulations results indicate that the system can behave as desired if a suitable controller can be developed. The system control needs to dynamically adjust the turn on delay of the IGBTs to regulate the output voltage while system parameters vary.

A proportional plus integral (PI) controller was used in the system. PI controllers are very common in control circuit design. Prior to the PI controller the load voltage is compared to a desired reference value. The load and reference voltages are subtracted to produce an error signal, which is then passed through the PI controller. A schematic of the controller is presented and discussed below.

Using PSIM, a controller design behavioral model was developed. The controller used feedback around a PI block to find the desired phase angle. Figure 4-4 shows the complete phase control circuit with the feedback behavior model. Figure 4-5 shows a close up of the feedback behavior model.

After the error signal passes through the PI block the resulting signal is naturally sampled. The rising edge of this naturally sampled waveform gives the desired phase turn on delay of the IGBTs. However, the falling edge is incorrect. To ensure the correct turn off delay, a monostable multivibrator (or "one shot") was used. The monostable turns on when it receives the on pulse from the naturally sampled waveform and stays on for the desired amount of time (calculated from the input frequency). Through this action I can guarantee correct turn off delay for simulation (Note: In the prototype a different method was used to control turn off delay). The controller also contains three blocks that implement the delay needed to drive each switch in the correct phase relative to the input sine waves.

The simulation results using the feedback controller looked promising. To verify the dynamic stability of the controller, the load was stepped. Figure 4-6 shows an example of the feedback controller with stepped load, including the start-up transient time. The system settles to the command voltage in a reasonable period of time. To generate figure 4-6, a capacitor in parallel with a DC current source was used as the load. Since the output voltage is nearly DC, the output resistor draws near DC current, and a DC current source is a good approximation. The abrupt changes represent a load change from 6 amps to 2 amps on the current source. As can been seen in the figure, the error voltage shows a peak overshoot of around 100V and a settling time on the order of 1/10 second. This indicates that after step changes in the load the output settles in a reasonable time. Although large, the output peak transient voltage stays within a tolerable range for components. This could likely be improved through a better design of the PI controller. Based on these results, the high-side phase-controlled rectifier should meet the desired system specifications. With simulation results that validate the design, I decided to build a prototype of the system.

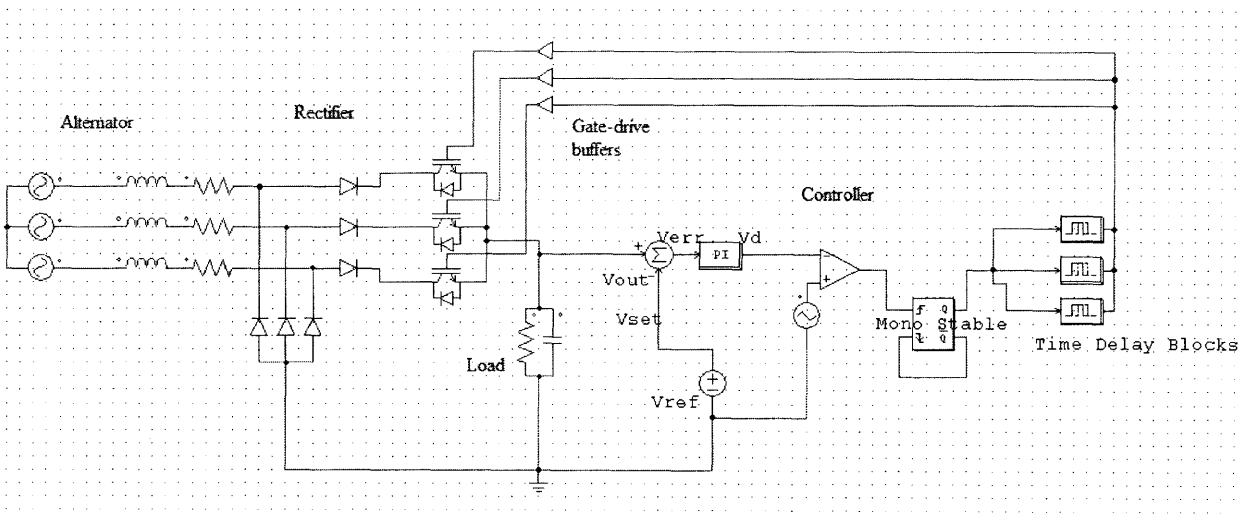

Figure 4-4 Phase-control circuit with feedback loop

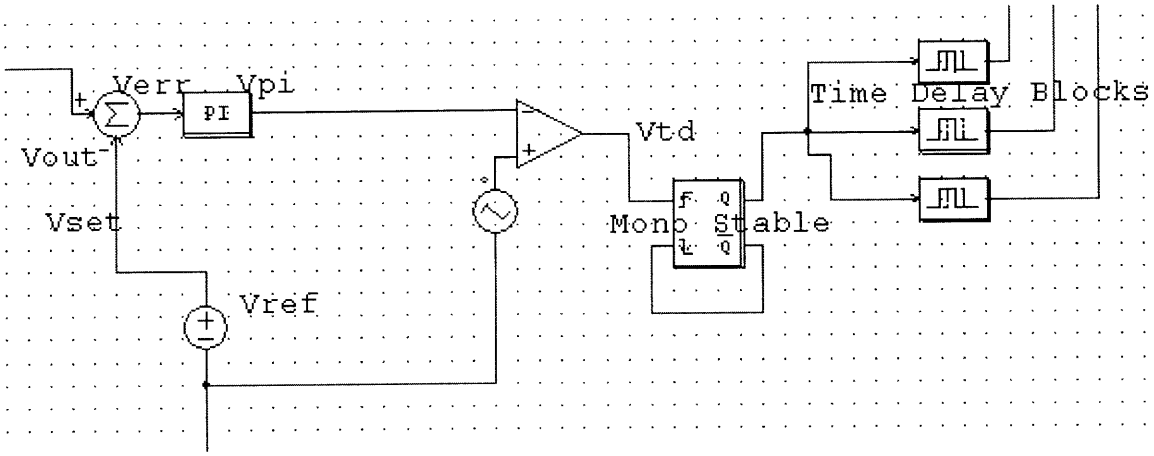

Figure 4-5 Close-up of feedback behavior model block

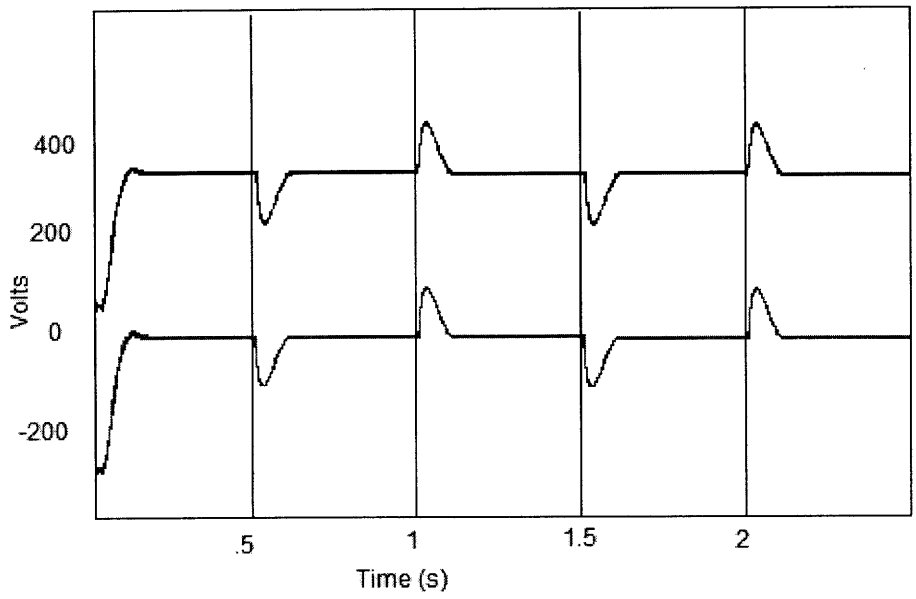

Figure 4-6: Output and Error Voltage. DC Current Load is stepped from drawing 6A to drawing 2A, at a frequency of 1hz. The PI controller has a gain of .003 and a time constant of .001.

4.1 SIMetrix website: http://www.catena.uk.com/site/downloads/download.htm
4.2 PSIM website: http://www.powersimtech.com/
4.3 See appendix A: Spice Models

Chapter 5

Construction and Integration of Prototype Rectifier

5.1 Overview

After validating the design with simulations, a prototype system was built. The prototype system was composed of an alternator model, attenuation and filtering electronics, a controller, gate driver circuits, a rectifier circuit, and a load. The design, construction, integration, and testing of the parts of the prototype is discussed in the sections below.

5.2 Alternator Model Employed in the Prototype

Because this was a desktop prototype, a model had to be used to simulate the downhole alternator being driven by fluid flow. It was important that this model closely resemble the actual downhole alternator. Further, it was imperative to have desktop control over the model's output speed and subsequent voltage. A MOOG G416-204 motor was used to model a downhole alternator [5.1]. This "alternator motor" was driven by a primary motor which had desktop control (Throughout this thesis, the primary motor will be called the "motor" and the secondary alternator motor will be called the "alternator"). This permitted simple robust control of the alternator. The alternator was the Fastact G415-804 by MOOG. The datasheet for both motors are available online [5.1].

Figure 5-1 provides a summary of parameters for the alternator. In the table the inductance and resistance values are given as terminal-to-terminal values; the phase values are therefore half of the values listed in the table. The phase inductance used in simulations is closely in line with the prototype inductance. The resistance listed in the table is less than the

resistance utilized for simulations. The listed resistance will be increased slightly by the resistance of the wiring.

		Metric [English]	G415-8xx (L50)
Continuous stall torque	Mo	Nm [lb-in]	25.0 [221.3]
Continuous stall current	Io	Arms	14.8
Peak torque	Mmax	Nm [lb-in]	60.0 [531]
Peak current	Imax	Arms	43
Nominal speed	nN	Rpm	2200
Nominal power	PN	kW [hp]	4.6 [6.2]
Maximum speed	nmax	Rpm	2400
Torque constant	kT	Nm/Arms [lb-in/Arms]	1.69 [15]
Motor resistance (line-to-line)	Rtt	Ohm	0.56
Motor inductance (line-to-line)	Ltt	mH	5.4
Inertia w/o brake	J	kg cm2 [lb-insec2 x 10-4]	18.4 [163]
Mass w/o brake	m	kg [lb]	16.6 [36.6]

Figure 5-1: Selected properties of the alternator model utilized in the prototype [5.1]

5.3 Gate Driver Boards

The gate driver was created from a Schlumberger proprietary gate drive board. The gate drive board is proprietary technology, so a schematic cannot be included here. Schlumberger wanted to use the existing gate driver board since it was already tested and certified for downhole use. (Because of the aforementioned fears related to downhole temperature and pressure, certification for downhole use is a lengthy process. Whenever possible it is best to re-use existing certified technology.) However, there were many issues related to the existing gate driver that made it difficult to adapt. Most notably the way in which the board turned off had to be modified. Also, capacitance had to be added to the board power supply to correct power supply issues.

After my modifications, the gate driver required distinct pulses to turn on and off. The driver was reset on start-up so that the first pulse would always turn it on. One pulse on the gate-drive input will raise the gate drive output high, which turns the IGBT on. The IGBT will stay on until the gate drive input receives an additional pulse, which will turn the switch off. This simple behavior is captured in figure 5-2.

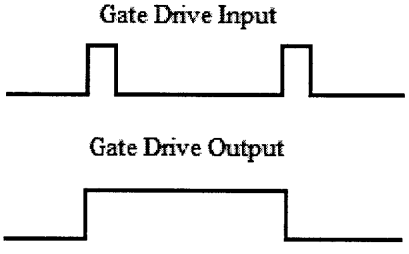

Figure 5-2: Simple gate driver behavior

5.4 Controller design

The original controller was designed for use with a Spectrum Digital TMS320LF2407 evaluation board [5.2]. The evaluation board is built around a Texas Instruments 2407 DSP. In addition the DSP, Spectrum Digital included a D/A converter, convenient connectors for I/O pins, and a simple to use JTAG interface for programming the board. The Spectrum Digital evaluation boards provide an excellent platform for prototyping. In the final stage of the project the 2407 board was replaced with a Spectrum Digital EZDSP TMS320F2812 board [5.3]. The 2812 board, which is similar to the 2407 in many regards, has a faster clock speed than the 2407 (150MHz as compare to 40MHz), allowing my controller to function well at higher alternator frequencies.

The code for the controller was generated with VISSIM DSP code developer [5.4]. VISSIM is a graphical tool that allows for designing systems at block levels. VISSIM is able to generate and compile C-code executing the block level control design. VISSIM has several qualities that make it ideal for rapid prototyping. First, it allows real-time interaction with a personal computer. For example, while reading the target boards ADC, you can watch the ADC value on your PC screen. This feature makes debugging easier. Second, VISSIM takes care of all the DSP specific problems for the user. The user does not have to learn about the DSP timers and registers. You can use simple VISSIM utilities to set control timers and I/O. This is extremely valuable if you want to rapidly prototype a system. It proved an excellent advantage on my project. Modifying my code for the switch from the 2407 to the 2812 board took little time compared to what modifying hand C code would have taken.

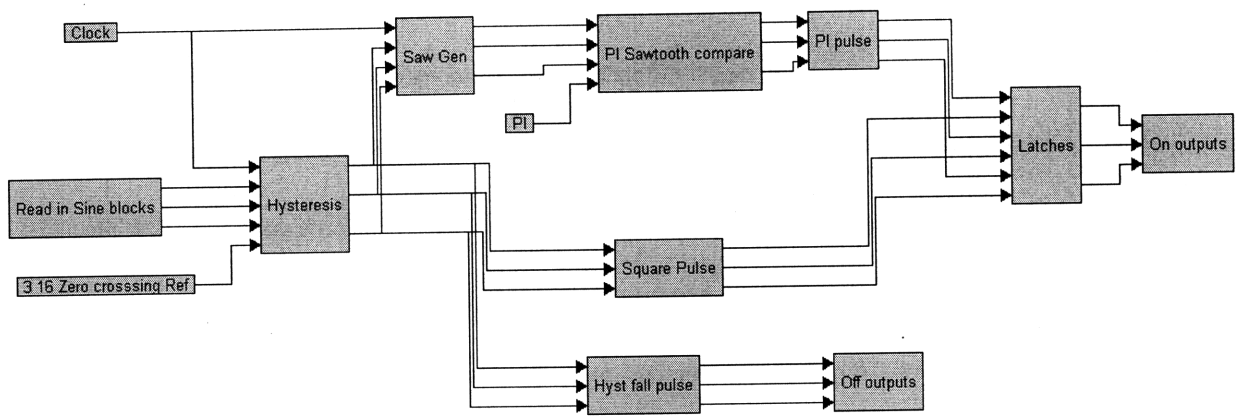

Figure 5-3: Block level illustration of my controller

Figure 5-3 shows the block level controller designed in VISSIM. To make the behavior clear, the functionality is described in the following paragraph. When describing a particular blocks function, the block's name from figure 5-3 will appear in parenthesis.

The behavior is as follows: Using the onboard ADCs, the controller reads in attenuated and scaled version of the alternator sine waves (Read in Sine blocks). These values are compared to a reference to detect when a phase has become positive (providing a reference point for phase). Using a PI controller, natural sampling, the programmed reference voltage, and additional manipulation blocks, the controller determines the desired on and off time for each switch. A more complete description of the controller behavior can be found in Appendix B.

In the system, the controller is followed by a gate drive board capable of driving the high-side IGBTs. To be compatible with the gate driver, the controller was designed to generate a pulse for turn on and turn off. Several of the blocks show in figure 5-3 (Square pulse, Hyst fall pulse, Latches) are needed to implement this functionality.

The switch turn off time must be carefully selected to protect the circuit. Turning off a switch with current flowing through it immediately destroys the switch, and must absolutely be avoided. To accomplish the goal of zero-current turn off, the IGBT turn off pulse was made to

occur near when the sine wave reached its minimum value. At first, the minimum value was determined by comparing successive samples of the sine-wave and waiting until the readings began to rise. In essence, this method found a local minimum and assumed it was the absolute minimum. This method worked well when the input was approximately a perfect sine wave. However, when the full system was run, due to loading and noise issues, the method needed to be revised. In the new system, the off pulse is generated after the input reading falls a certain amount below its median value. This requires calibrating the system with a turn-off Δ. Once the input falls below the mean by Δ, the off pulse is generated.

Although this method was extremely robust against noise problems it included the undesirable effect of requiring calibration to the appropriate levels. This design could be improved upon by using separate hardware (such as a positive phase-current detector) to determine a safe time to turn off the switch.

5.5 Rectifier circuit construction

The rectifier shown in figure 5-4 was built. IGBTs were used for the high side switches. The IGBTs selected had an ultrafast soft recovery diode. These IGBTs have a maximum collector-emitter voltage of 600V, a maximum continuous collector current of 39A, and a maximum pulsed collector current of 280A. Each high side IGBT was preceded by a diode to prevent the IGBT body-diode from conducting in the reverse direction. The diodes used were 1200V, 30A diodes. The diodes could take a nonrepetitive peak current surge of 300A. The diodes had hyperfast soft recovery characteristics (t_{rr} <65ns).

The rectifier circuit was mounted on a one foot by two foot solid metal ground plane. The switches were electrically isolated from the ground plane. The switches were mounted

above aluminum oxide ceramic insulating pads, were bonded to the pads using silicone heat sink compound and were connected using 12 gauge copper wire. Screw terminals were provided for easy connection and access to the alternator phases, as well as both sides of the load. The large metal sheet left room for later mounting the gate drive board, controller board, and the attenuation and low-pass filter blocks.

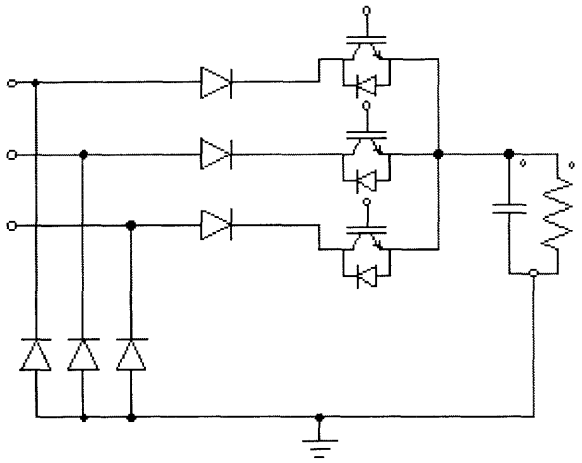

Figure 5-4 Rectifier circuit with load

5.6 Attenuation and Low-Pass Filter Blocks

To integrate the system, several more components were needed. The sine waves coming from the alternator were hundred of volts peak-to-peak. These signals needed to be attenuated to levels that the ADC could handle (0V-3.3V). In addition, a differential amplifier was needed to measure the phase-to-phase voltages. To accomplish both of these feats a differential attenuator was designed. The schematic of this amplifier can be seen in figure 5-5. The frequency response of this differential amplifier can be seen in figure 5-6. As can be seen from the figure, at the frequencies of interest this amplifier has an attenuation of about 285:1. This attenuation

factor will allow a maximum peak-to-peak incoming sine wave of 940V and convert it to maximum peak-to-peak below 3.3V.

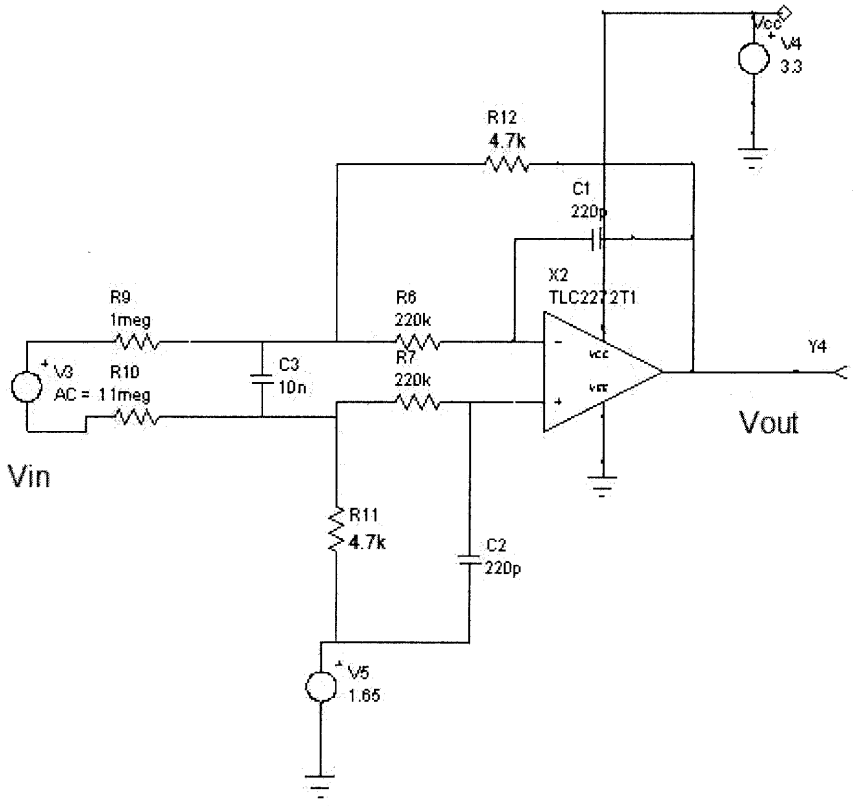

Figure 5-5: Schematic of differential attenuator circuit.

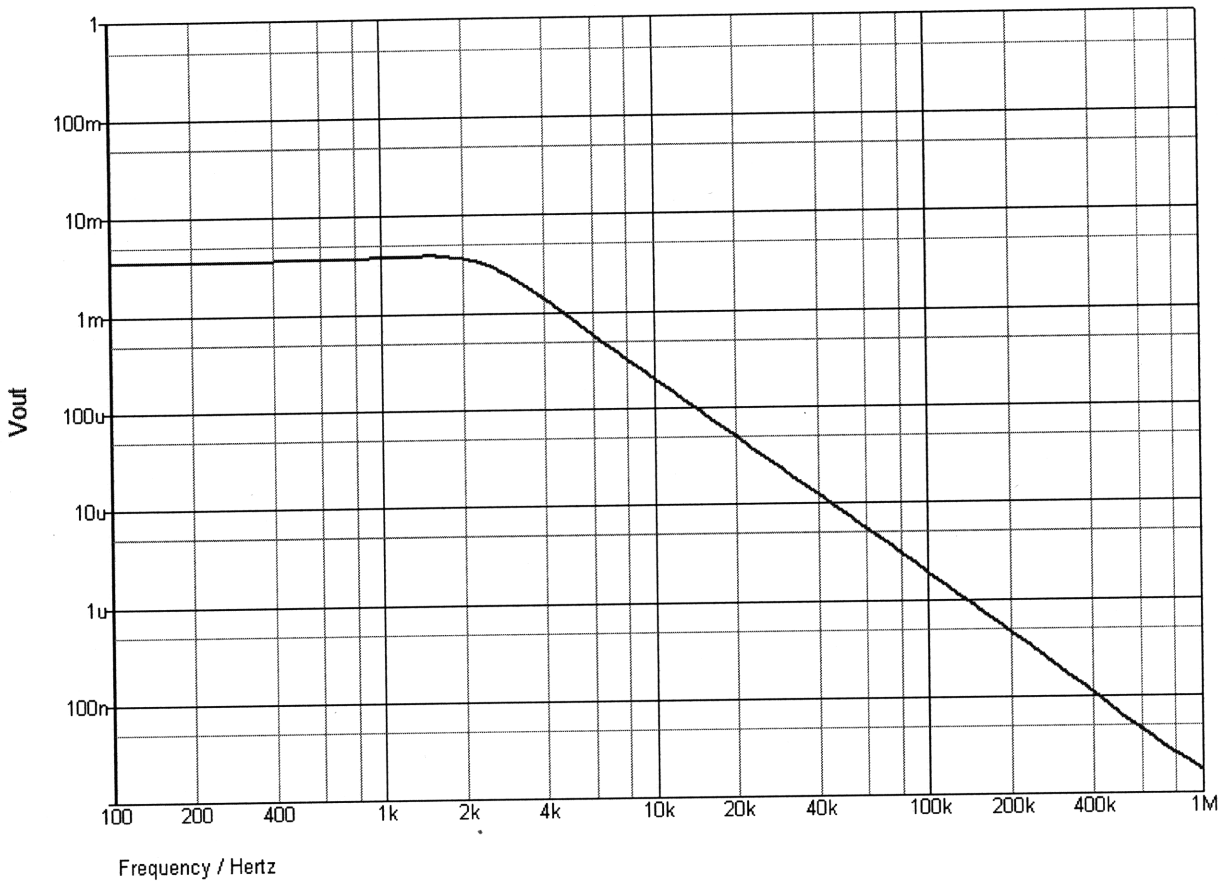

Figure 5-6: Frequency response of the differential amplifier in figure 5-5. At low frequency, the output is attenuated by a factor of 285.

After constructing the differential attenuation circuit, the output still had some undesirable components. In addition to the desired sine wave, some high frequency noise had been picked up. Most of this noise was caused by the high-frequency high-voltage driver that was driving the motor. To correct this, a low-pass filter stage was included after the amplifier stage. The low-pass filter was designed using Texas Instruments FilterPro toolbox. The low-pass filter is a second order Sallen-Key type Butterworth filter. The circuit was designed to have a cut-off frequency of 1KHz. This frequency was selected to be well above the maximum switching frequency of around 300Hz, but well below the higher order noise frequencies. A screen shot of the lowpass filter design from FilterPro can be seen in figure 5-7. The screen shot

includes the circuit schematic, pertinent design info, and a graph of the frequency response of the filter.

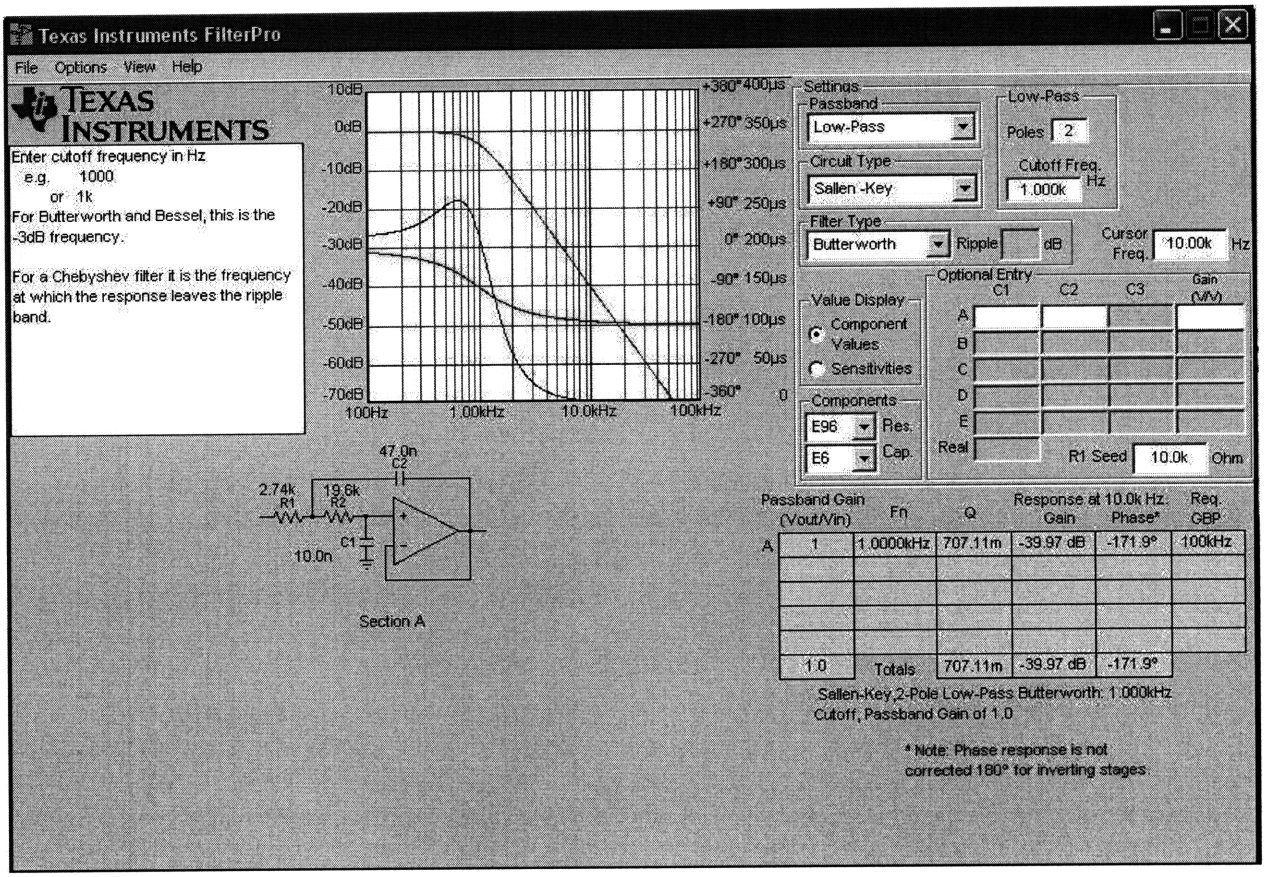

Figure 5-7: Low-pass filter design toolkit. Image generated with TI FilterPro [5.5].

5.7 Capacitor Bank and Load Selection

The output of the rectifier is capacitively filtered with a 300uF bank of capacitors. The capacitor bank is a custom design. It is composed of stacked ceramic capacitors. The stack is comprised of twenty separate 15uF capacitors. This large filter capacitance is used to maintain an excellent voltage filtering result at the load. This ensures a low ripple DC output.

The rectifier was loaded with a bank of 100 watt incandescent light bulbs. The load bank was made with five parallel rows of light bulbs. Each row of bulbs could be switched from three to six light bulbs. This allowed a load of either three bulbs or six bulbs in series, with one to five rows in parallel. This allowed me to run the system from 300W (one row of three bulbs) up to 3000W (five rows of six bulbs).

5.1 Moog Inc. Corporate website. Data sheet for Fastact G/G4xx servomotors available at: http://www.moog.com/Media/1/fasg-g4xxservomotors.pdf

5.2 Spectrum Digital Corporate Website. Information for EVM320LF2407A evaluation module available at: http://www.spectrumdigital.com/product_info.php?cPath=37&products_id=47&osCsid=0a01682a826e163811e917ed8fc0f272

5.3 Spectrum Digital Corporate Website. Information for eZdsp F2812 available at: http://www.spectrumdigital.com/product_info.php?cPath=32&products_id=169&osCsid=81e52daef6766ca1f741b300eb1134dc

5.4 Visual Solutions, Inc. VisSim/Embedded Controls Developer™. Available at: http://www.vissim.com/c2000/c2000.html

5.5 TI FilterPro active filter design application. Available at: http://focus.ti.com/docs/toolsw/folders/print/filterpro.html

Chapter 6

Results

6.1 Measurement Results

The basic operation of the system is captured in figures 6-2 through 6-4. Since each of the three phases exhibits similar behavior, waveforms will only be shown for one phase. In figures 6-2 through 6-4, trace one is the line-to-neutral voltage on one phase (eg. the voltage between the points A1 and N from figure 6-1), trace two is the corresponding gate-drive signal (eg. G1-gnd from figure 6-1), and trace three is the load voltage (L-gnd from figure 6-1). To generate figures 6-2 through 6-4, the controller reference voltage, which is the voltage the controller was trying to create at the load, was set to 135V. Also, the load was attached to a DC power supply. This allowed the load to be externally held to a constant value. For figure 6-2, the DC source on the load was set to 127V (notably, this is below the reference value). Because the load voltage is below the reference, the controller tries to increase the load voltage. To do so, the controller maintains a high duty cycle on the IGBT gates. After taking the screen shot for figure 6-2, the DC power supply voltage was increased to 137V. Since this is above the reference voltage, the controller tries to reduce the load voltage, and therefore reduces the gate-drive duty cycle. Figures 6-3 and 6-4 depict this process. As can be seen in the figures, the gate-drive signal decreases and approaches zero.

Figure 6-5 illustrates this process in the reverse direction. In this figure, the top trace is the line-to-neutral voltage (A1-N from figure 6-1) and the bottom trace is the corresponding gate-drive signal (G1-gnd). To generate the figure, load was again attached to an external DC power supply. This time, however, the DC load voltage was decreased from above the reference

voltage to below the reference voltage. The controller observes the drop of the load voltage, and tries to increase the load voltage by increasing the gate-drive duty cycle.

For the remainder of the tests, the rectifier was loaded with 100W, 120V light bulbs. The load was designed to allow loading with rows of three bulbs in series. There were ten total rows of bulbs, allowing for loading with from one to ten parallel rows of three bulbs in series. Light bulbs were selected because they are a convenient and inexpensive way to get a high power load. Light bulbs have a resistance that depends on the temperature of the filament. The resistance of a light bulb can range from about 9.6Ω at room temperature (300°K) to 144Ω when the bulb is on (3000°K) [6.1]. Throughout this section, the load resistances given are calculated assuming 144Ω per bulb.

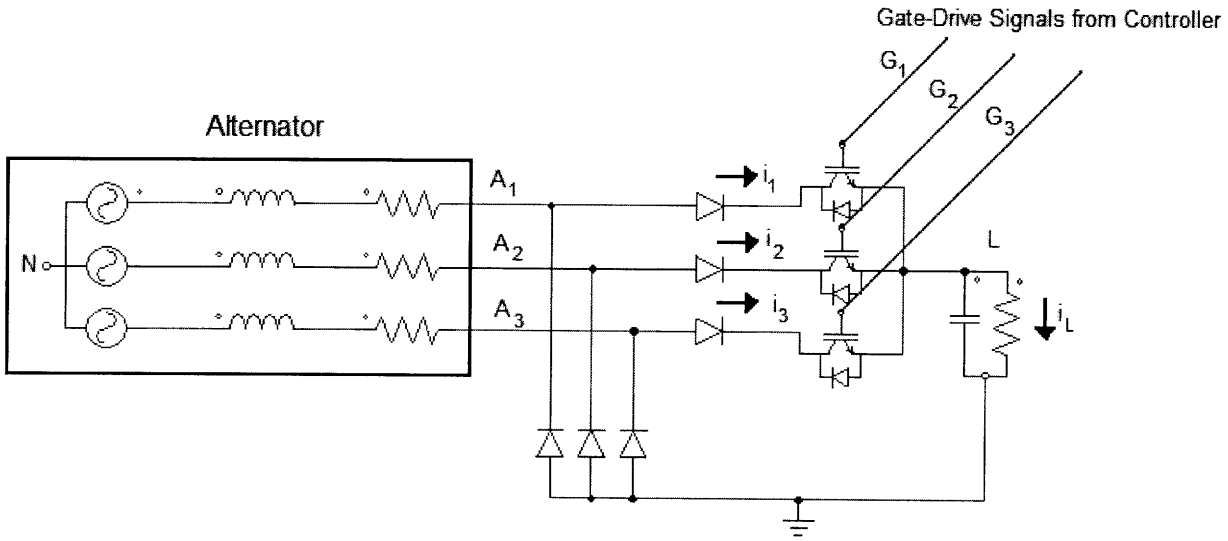

Figure 6-1: Abbreviated Schematic of system for use in explaining measurements. The lines labeled A1, A2, and A3 are the three phases coming from the alternator. The point labeled N is the neutral point. In the actual alternator used, all of these terminals were available for measurement. The lines labeled G1, G2, and G3 are the three gate-drive voltages coming from the controller. G1, G2, and G3 correspond to A1, A2, and A3, respectively. The load is labeled L. Ground is also shown.

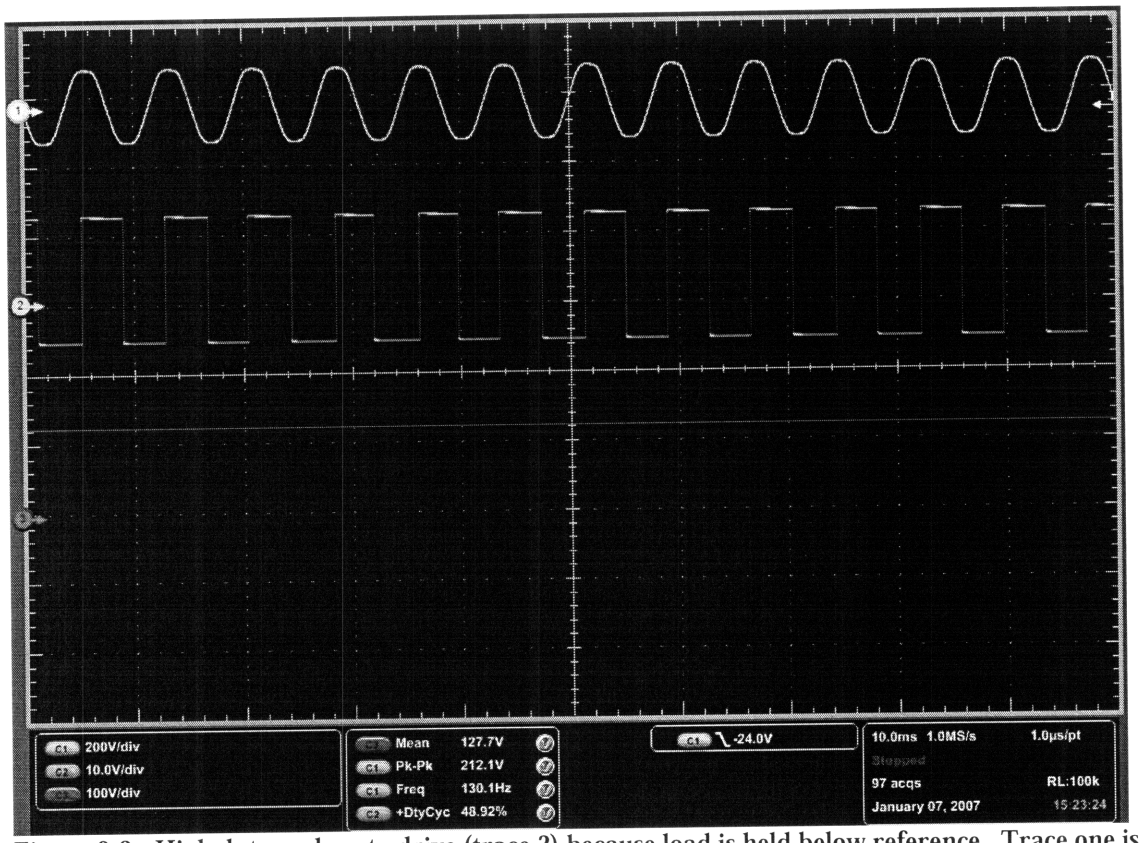

Figure 6-2: High duty cycle gate-drive (trace 2) because load is held below reference. Trace one is the line-to-neutral voltage, trace two is the IGBT gate-drive signal, and trace three is the load voltage.

Figure 6-3: Gate-drive duty cycle (trace 2) is decreasing since the load voltage was increased. Trace one is the line-to-neutral voltage, trace two is the IGBT gate-drive signal, and trace three is the load voltage.

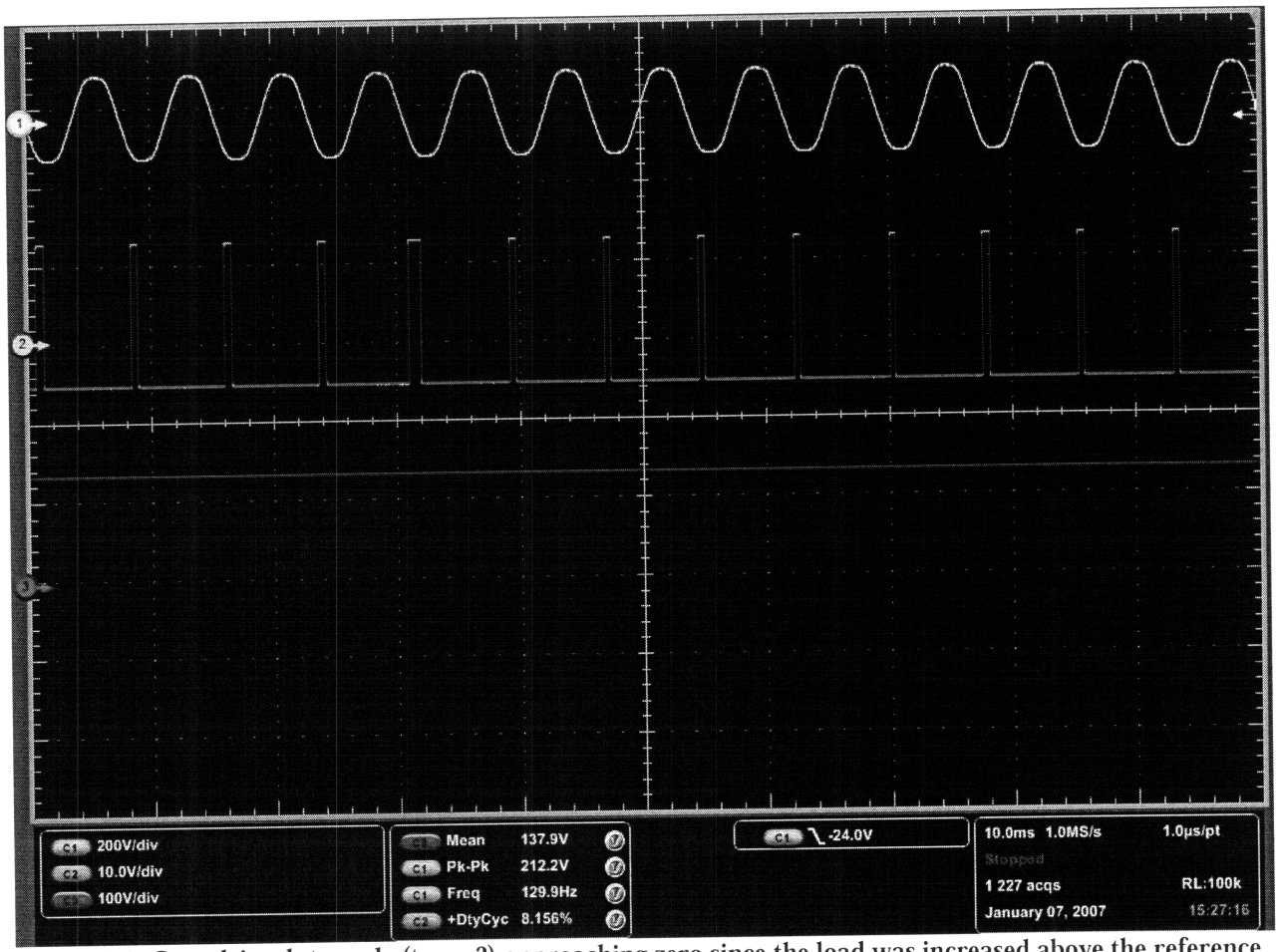

Figure 6-4: Gate-drive duty cycle (trace 2) approaching zero since the load was increased above the reference. Trace one is the line-to-neutral voltage, trace two is the IGBT gate-drive signal, and trace three is the load voltage.

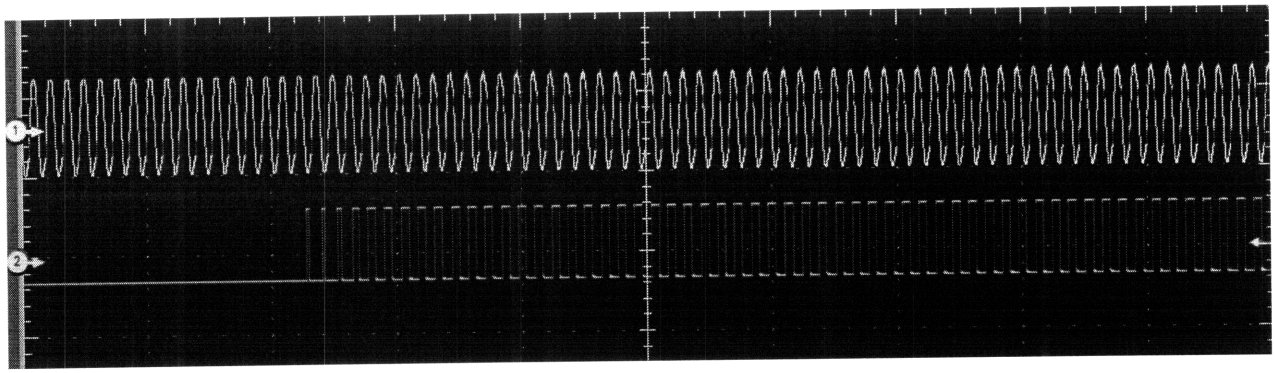

Figure 6-5: Control reaction to load change from above the reference to below the reference. Trace one is the line-to-neutral voltage and trace two is the IGBT gate-drive signal.

As was discussed above, it is necessary to ensure that no current is flowing through an IGBT when it is turned off. Ignoring inductance, current would not be driven through a phase when the line-to-neutral voltage is more than 150 degrees. However, with inductance current may be maintained beyond 150 degrees. It was determined that the inductance never maintains the current beyond 180 degrees. Therefore, after the phase passes 180 degrees the IGBT can be safely shut off. To provide additional margin of safety for turn off, the controller was calibrated to ensure that turn off occurred well after 180 degrees of phase. The controller observes the line-to-neutral voltage, and once it has fallen below the median by the calibrated Δ, the off pulse is generated. Figure 6-6 shows the line-to-neutral voltage (A1-N from figure 6-1) overlaid with the IGBT gate drive voltage (G1-gnd from figure 6-1). As can been seen in the figure, the IGBT is turned off when the line-to-neutral voltage is near its minimum (that is, near 270 degrees of phase). Though this control method sufficed for the prototype, it would be better to have current sensing technology in a production design. This technology could provide absolute certainty of zero current without the need for calibration.

Figures 6-7 and 6-8 depict the full system operation. These figures show actual rectifier operation, with current flowing from the source to the load. In figure 6-7, trace 1 is the line-to-neutral voltage for the observed phase (A1-N from figure 6-1), trace 2 is the gate-drive voltage for the observed phase (G1-gnd from figure 6-1), trace 3 is the load voltage (L from figure 6-1), and trace 4 is the current through the observed phase (i1 from figure 6-1). In figure 6-8, trace 1 is the line-to-neutral voltage for the observed phase, trace 2 is the gate-drive voltage, trace 3 is the load voltage but AC coupled, and trace 4 is the DC current going through the load (iL from figure 6-1). For these figures, the set-up was run with a command of 300V. Trace 3 of figure 6-8 shows that the load voltage has low ripple content (Peak-to-Peak is about 2V).

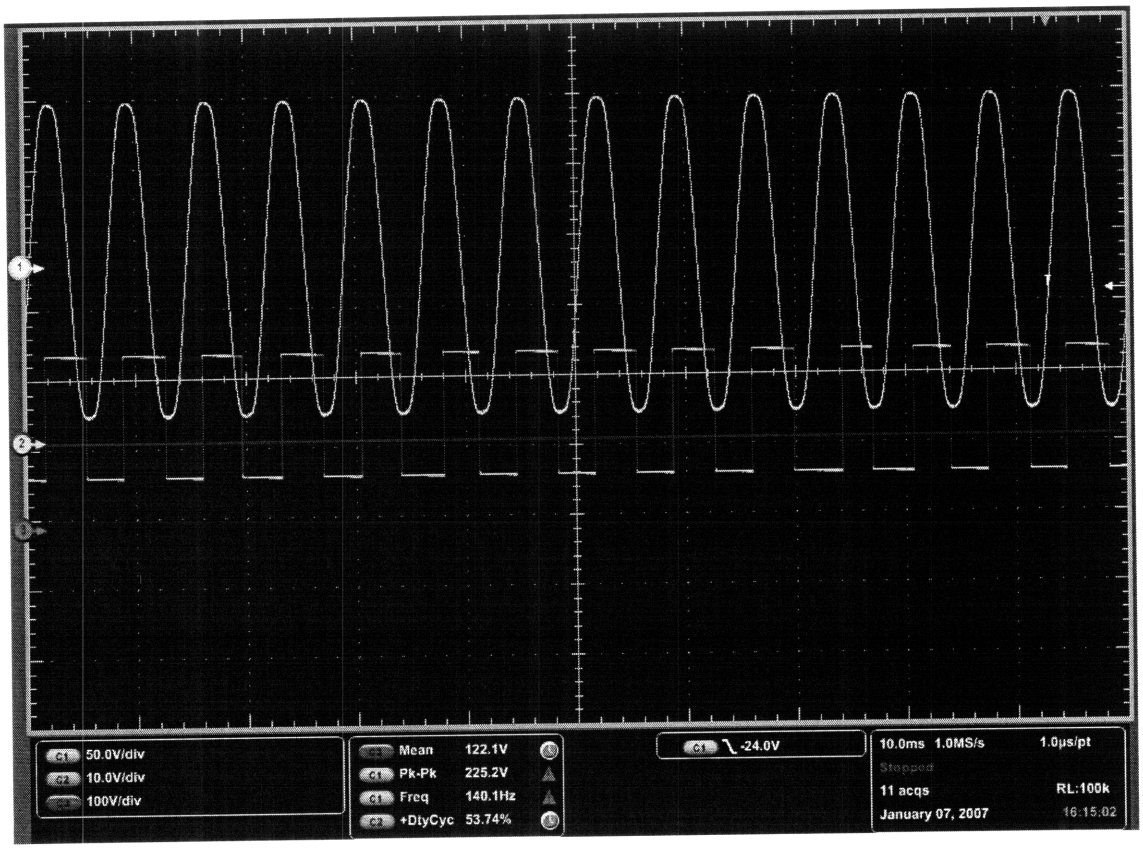

Figure 6-6: Close up where gate off pulse crosses sine-wave. Trace one is the line-to-neutral voltage. Trace two is the IGBT gate drive voltage. When this signal is high the IGBT would be on, when low the IGBT would be off. Trace three is the load voltage.

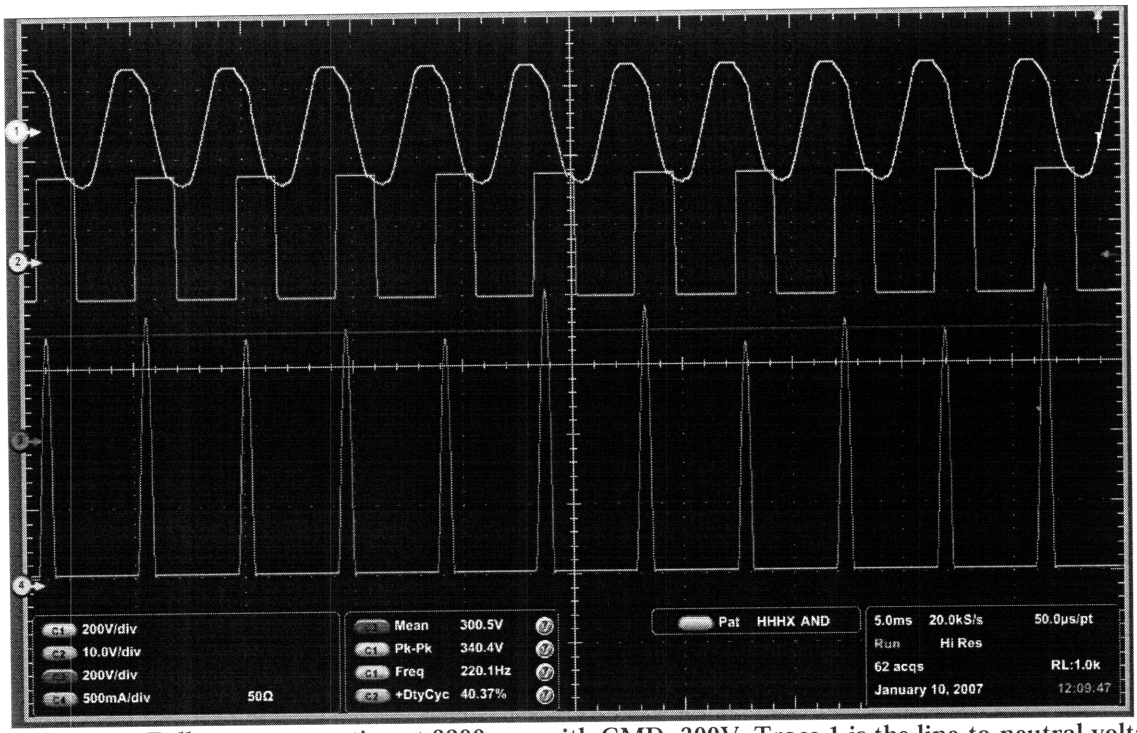

Figure 6-7: Full system operation at 2200rpm with CMD=300V. Trace 1 is the line-to-neutral voltage, trace 2 is the IGBT gate-drive voltage, trace 3 is the load voltage, and trace 4 is the current through the phase.

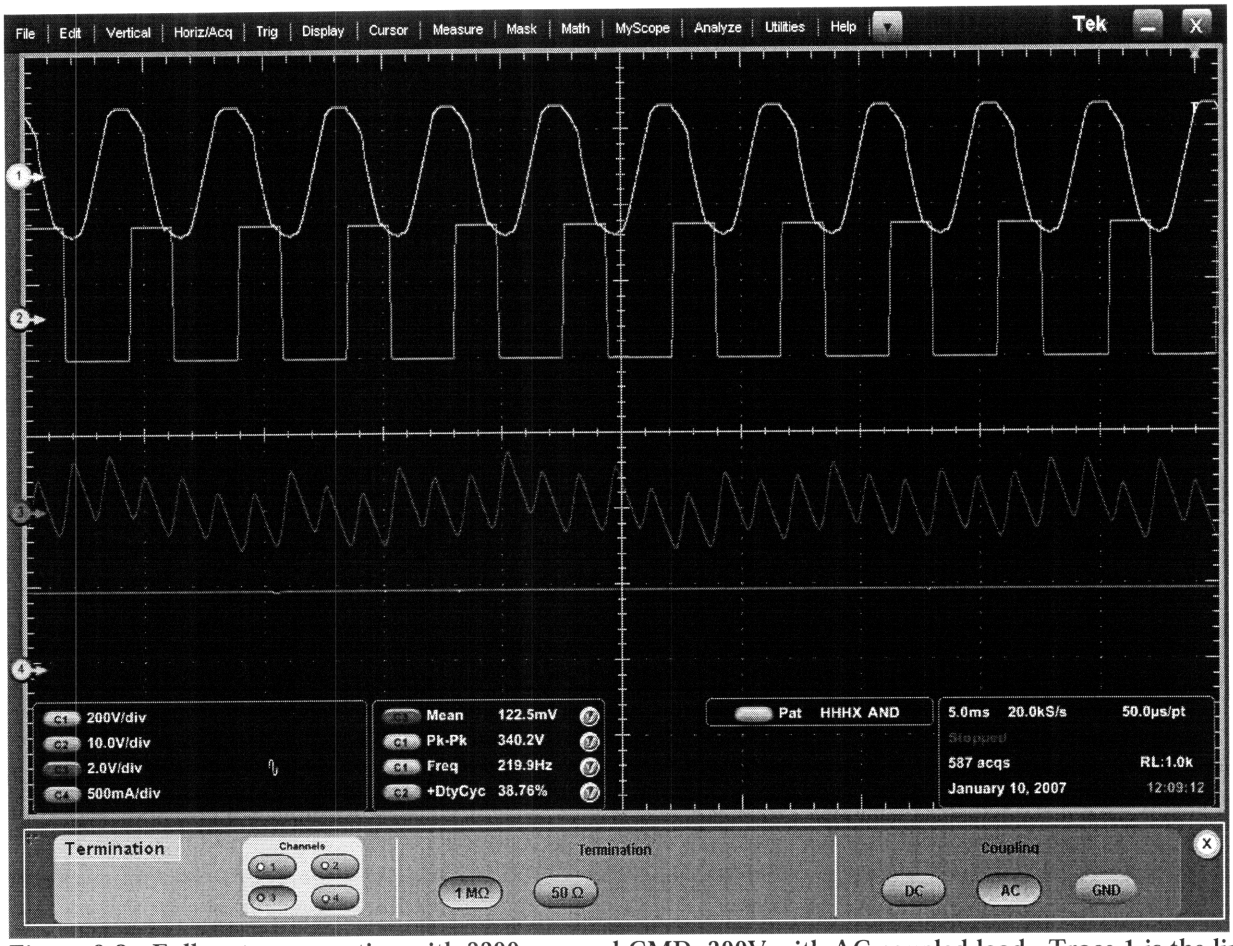

Figure 6-8: Full system operation with 2200rpm and CMD=300V with AC coupled load. Trace 1 is the line-to-neutral voltage, trace 2 is the IGBT gate-drive voltage, trace 3 is the load voltage but AC coupled, and trace 4 is the DC current going to the load.

The system was tested for a number of loads and commands. Figure 6-9 show the output voltage results for a number of different loads and input frequencies. As the figure shows, the output voltage DC level was well controlled. In the case where we are drawing high currents from the source while running the input at low frequency and voltage, the output is pulled down slightly from the desired level (less than two percent). However, the output is still well within the tolerable range.

Figure 6-10 shows the peak-to-peak load ripple voltage at the load for different frequencies and loads. The maximum peak-to-peak ripple observed was 4.5V, which is acceptable.

Figure 6-11 shows the RMS phase current. Figure 6-12 shows the peak switch current for a given frequency and load. Peak currents observed are well below component tolerances.

Figure 6-13 and figure 6-14 shows results for more load test cases. In 6-13 we see the load voltage is pulled down slightly when with low resistance at the load (Voltage is pulled down max of 20V, or about 6 percent). However, even at the worst case the effect is small and well within our constraints. In figure 6-14 we can see current increases almost linearly as the load increases.

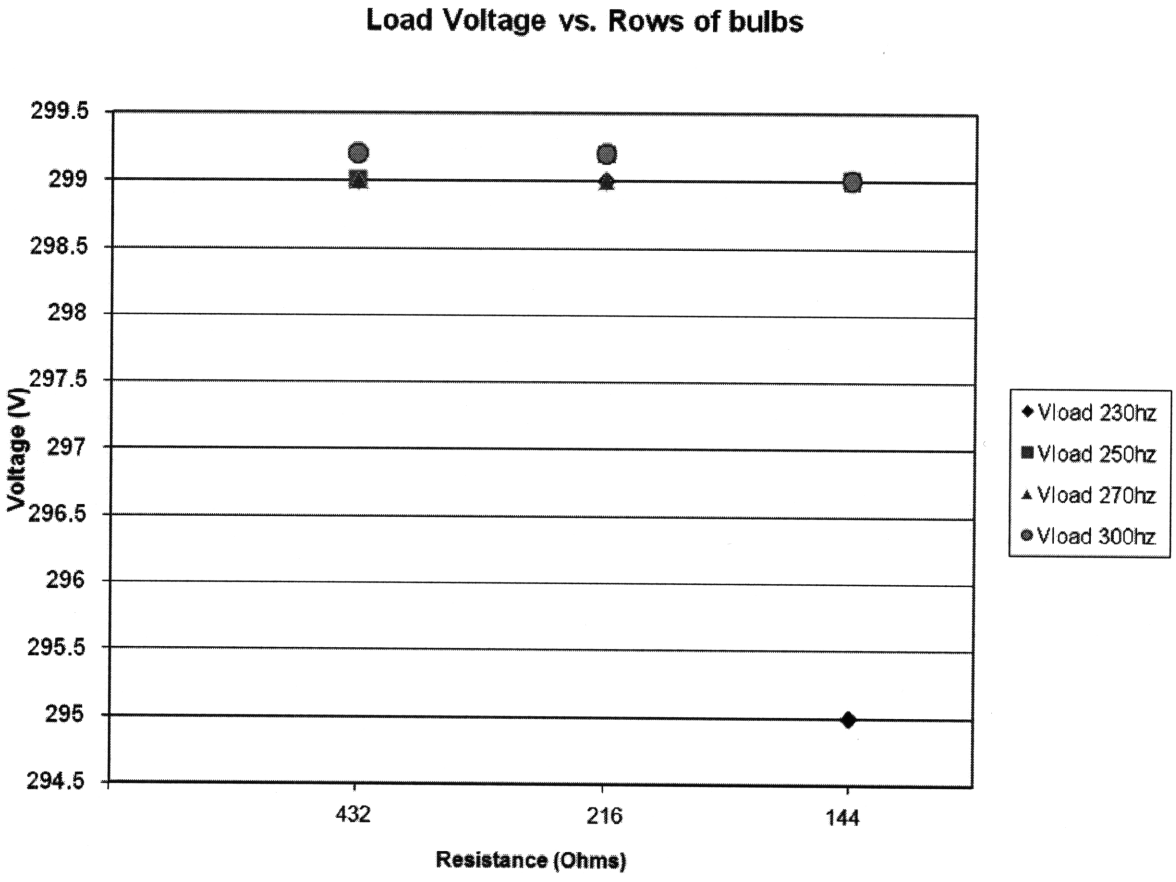

Figure 6-9: Output voltage for several loads. Each load is three bulbs in series, with the number of rows in parallel listed on the chart.

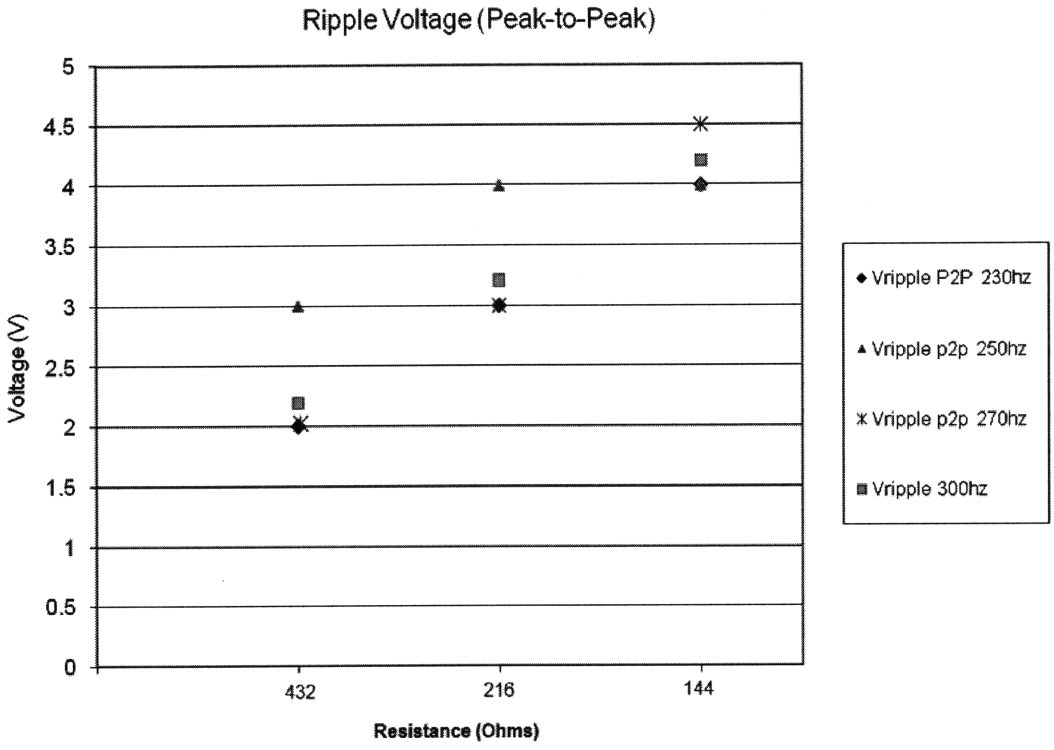

Figure 6-10 Voltage ripple peak-to-peak for different loads

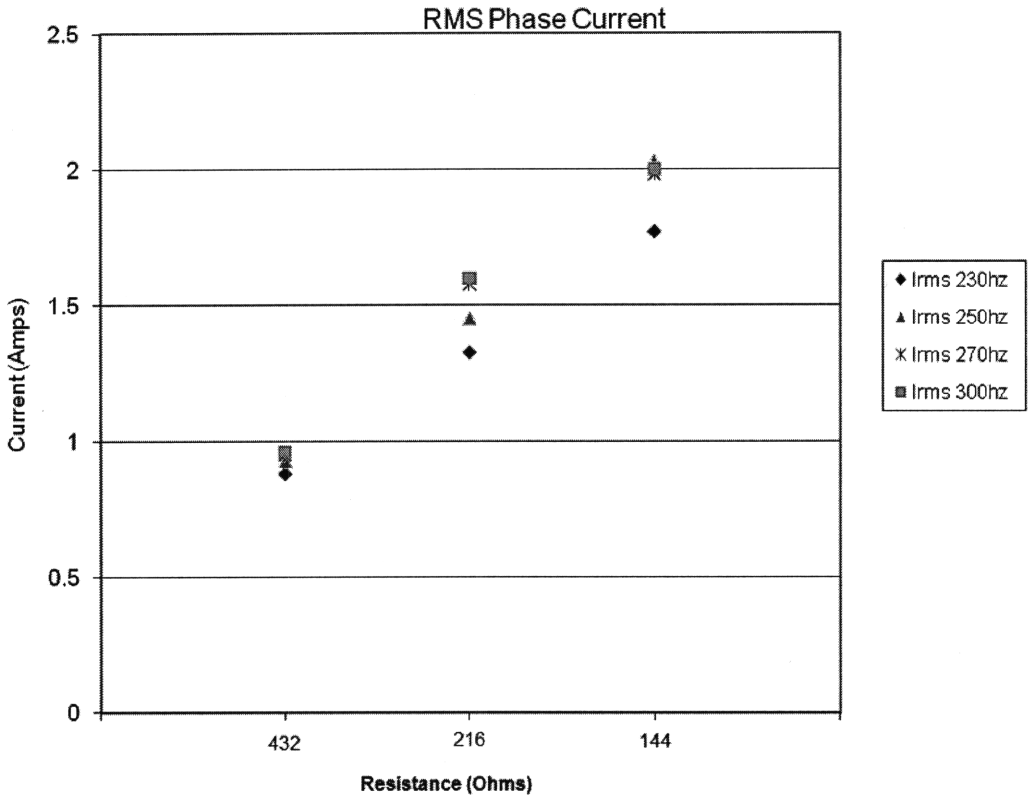

Figure 6-11 RMS current for different loads

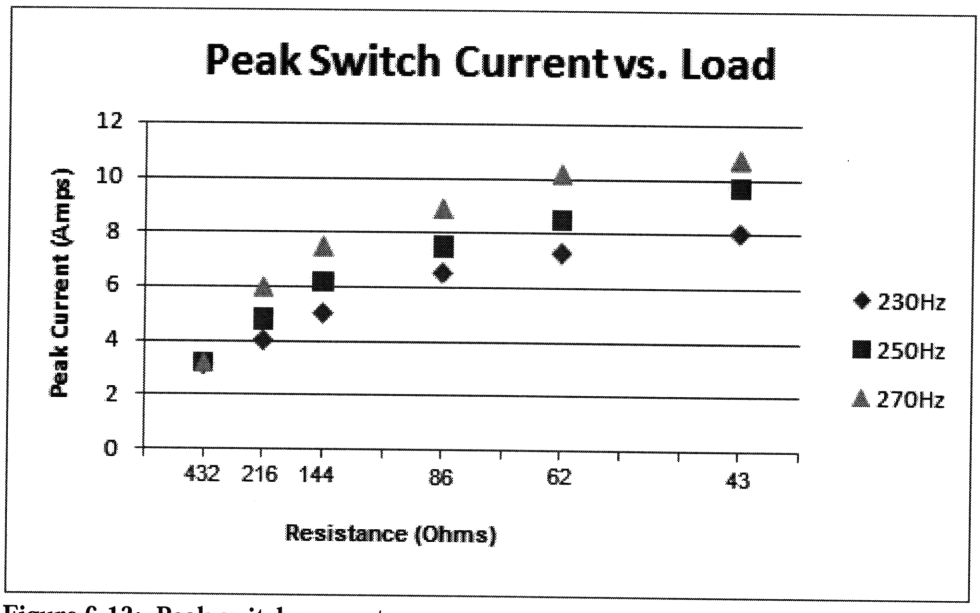

Figure 6-12: Peak switch currents.

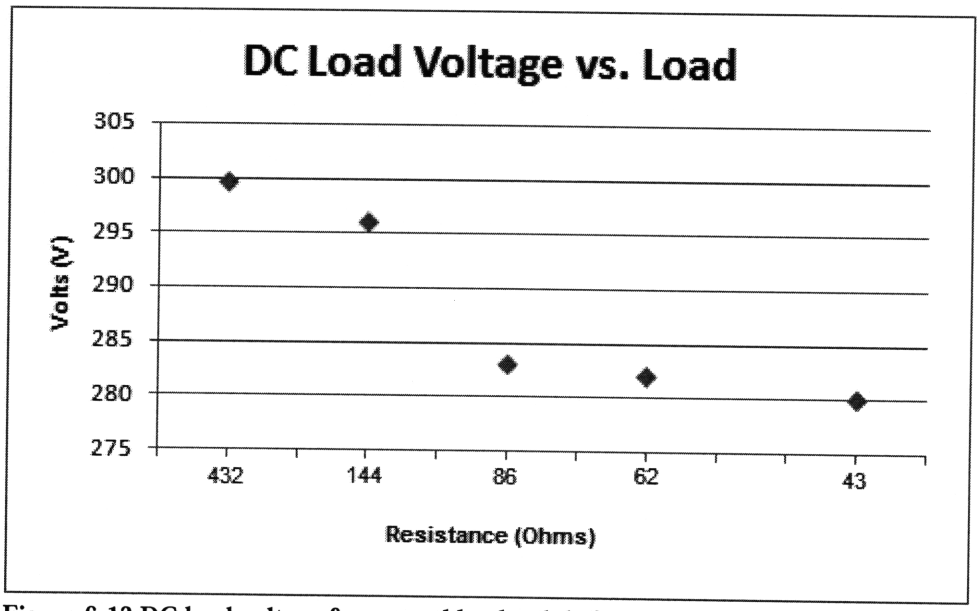

Figure 6-13 DC load voltage for several loads while frequency is held constant at 230Hz.

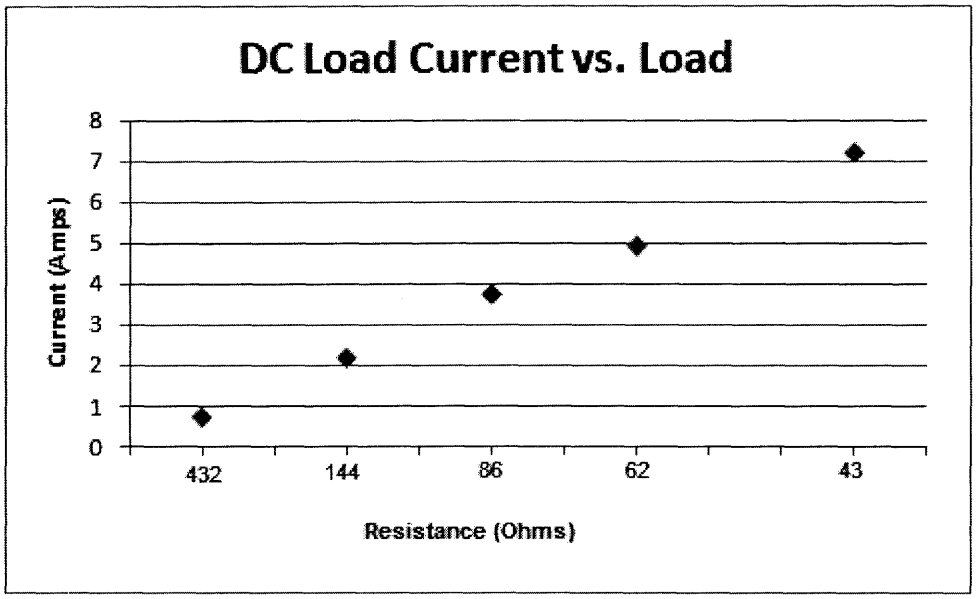

Figure 6-14 DC load current for several loads while frequency is held constant at 230Hz.

6.1 Sciences Education Foundation (General Atomics). "The Physics and Materials Science of the Incandescent Light Bulb." See results from experiment 2, available online at: http://www.sci-ed-ga.org/modules/materialscience/light/pdf/section_08.pdf

Chapter 7

Future work

7.1 Suggested Future Work

There are several key areas in which future work should be conducted. If Schlumberger chooses to proceed with this design, current sensing technology should be utilized to remove the need for calibration. Also, smaller gate driver circuits should be designed to make the whole package much smaller. In addition, the entire package needs to be designed to be packed in a downhole device, and must be pressure, temperature, and shock qualified.

In the above design, a "soft-start" mechanism was utilized. Prior to adding the soft-start, the system was started from rest with zero volts on the load. The system attempted to get up to the command voltage too quickly, driving too large a current and destroying the switches. To alleviate this problem, the system was started with the load pre-charged to 150V. The load was charged using a desktop power supply. This design successfully reduced the initial current transients. Once the system regulated itself up past the 150V, the soft-start generator could be removed. This mechanism was sufficient for demonstrating the concept, however, it would be undesirable in a production system. Future work must change the system to incorporate a commercially viable soft start.

Appendix A

Spice Models

A.1 IGBT Models

The IGBT model used came included in the SIMetrix software package. The model was a model of the actual IGBT used in the circuit (IRG4PC50FD). A model of a BYT30P-1000 was use for the diodes. The SPICE code for each device is found below:

```
*File: g4pc50fd.spi
*
.SUBCKT irg4pc50fd 1 2 3
**************************************
*      Model Generated by MODPEX     *
*Copyright(c) Symmetry Design Systems*
*          All Rights Reserved       *
*    UNPUBLISHED LICENSED SOFTWARE   *
*   Contains Proprietary Information *
*      Which is The Property of      *
*     SYMMETRY OR ITS LICENSORS      *
*Commercial Use or Resale Restricted *
*   by Symmetry License Agreement    *
**************************************
*Model generated on Mar  13, 01
* MODEL FORMAT: SPICE3
*Symmetry IGBT Model (Version 1.0)
*External Node Designations
*Node 1 -> C
*Node 2 -> G
*Node 3 -> E
M1 9 6 8 8 MSUB L=100u W=100u
* Default values used in MSUB:
* The voltage-dependent capacitances are
* not included. Other default values are:
*   RD=0 RS=0 LD=0 CBD=0 CBS=0 CGBO=0
.MODEL MSUB NMOS LEVEL=1
+VTO=6 KP=1.43492 LAMBDA=0.625287 CGSO=3.89883e-05
RD 7 9 0.000977651
RS 4 8 0.000996656
D1 3 1 d4pc50f
.MODEL d4pc50f d
+IS=6.53076e-09 RS=0.0228005 N=2 EG=0.6
+XTI=0.500186 BV=600 IBV=0.00025 CJO=1.16126e-08
+VJ=1.5 M=0.638969 FC=0.5 TT=1.7512e-08
+KF=0 AF=1
```

```
Q1 4 7 1 QSUB OFF
.MODEL QSUB PNP
+IS=1.55191e-17 BF=23.5373 NF=0.85 VAF=131.421
+IKF=1172.87 ISE=9.99937e-12 NE=2.00092 BR=1.04218
+NR=0.999653 VAR=101.051 IKR=999.976 ISC=3.7297e-12
+NC=1 RB=0.00867835 IRB=990.164 RBM=0.00867835
+RE=0.000648409 RC=0.000998116 XTB=0 XTI=3.01201
+EG=1.2 CJC=8.09777e-10 VJC=0.4 MJC=0.340876
+CJE=1.61955e-08 VJE=0.4 MJE=0.9 TF=9.99999e-10
RDS 7 4 1e8
RER 4 3 0.0137988
RG 6 2 2
RL 10 11 1
D2 12 11 DCAP
* Default values used in DCAP:
*   RS=0 EG=1.11 XTI=3.0 TT=0
*   BV=infinite IBV=1mA
.MODEL DCAP D IS=1e-32 N=50
+CJO=6.71681e-09 VJ=0.4 M=0.9 FC=0.5
D3 0 11 DL
* Default values used in DL:
*   EG=1.11 XTI=3.0 TT=0 CJO=0
*   RS=0 BV=infinite IBV=1mA
.MODEL DL D IS=1e-10 N=0.4
VFI2 12 0 0
FI2 6 7 VFI2 -1
EV 10 0 7 6 1
CAP 10 13 6.71681e-09
RCAP 10 14 1
D4 0 14 DL
VFI1 13 14 0
FI1 6 7 VFI1 -1
.ENDS irg4pc50fd
```

BYT30P-1000 code:

```
.model byt30p-1000 D(IS=3e-18 RS=5m CJO=970p M=0.4 VJ=0.75 ISR=790n
+ BV=1000 IBV=100u TT=130n)
```

Appendix B

Controller Design and Code

Figure B.1 shows a block diagram of the complete control system developed in VISSIM. The block labeled "Sine blocks" read in the three ADC's capturing the phase-to-neutral voltages. Because the ADC's can only read voltages from 0V-3.3V, these voltages phase-to-neutral voltages were attenuated and shifted some DC voltage above zero (This was described in chapter 5). The block labeled "Zero Crossing Ref" reads in the DC voltage used in the aforementioned shift. These phase-to-neutral and reference signal are compared in the "Hysteresis" block. This block also includes some noise protection by waiting until the phase-to-neutral voltage has gone sufficiently far above the reference to signal the change (the final buffer voltage used was .05V). This ensures that small amounts of noise will not cause the system to change early. Through this method the system is able to determine when a given phase-to-neutral voltage is positive or negative, and generates the corresponding square wave.

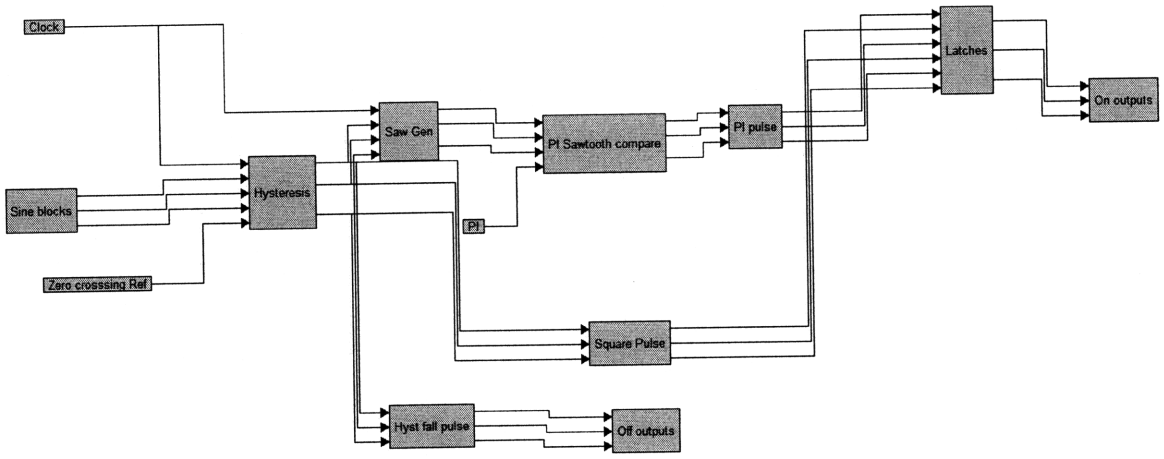

Figure B.1 Block diagram of control system created in VISSIM.

In the "Saw Gen" block, the square wave is integrated to form a sawtooth. This sawtooth has the same frequency as phase-to-neutral voltage, and is also locked in phase for each of the three respective phase-to-neutral voltages. At this point, the attenuated load voltage is read by an ADC ("PI" block) and is naturally sampled in the "PI sawtooth compare" block. The results are used to generate the appropriate on pulses. Additional blocks add additional noise protection, ensuring that the system only gets one on pulse per cycle.

The off pulses are generated through the "Hyst fall pulse" block. This block waits until the phase-to-neutral voltage falls sufficiently below the reference before generating an off pulse (this was done by calibration and experiment, with a final value of .15V being used). As was discussed earlier, anytime the phase-to-neutral voltage is below the reference the system cannot drive current. Therefore, the IGBTs can safely be turned off at any point during this time.

The complete system code was generated by VISSIM. The C code can be found below:

```
/*** VisSim Automatic C Code Generator Version 6.0C12 ***/
/*  Output for c2.vsm at Sun Jan 13 22:42:09 2008 */


#include "math.h"
#include "cgen.h"
#include "c24x.h"
#include "c24x.h"
#include "pidFx.h"
int _crystalMultiple=0xa;
int MHZ=150;

int _maxAnalogInChan=38;
PI_CONTROLLER pid15 = {0,0,10240,32735,0,0,32767,0,0};
extern CGDOUBLE Zed;

static int clock;
```

```
static int __clock_88;
static int __reset_68;
static int __lowlim_89;
static int __highlim_90;
static int __input_91;
static int __clock_228;
static int __reset_201;
static int __lowlim_229;
static int __highlim_230;
static int __input_231;
static int __clock_278;
static int __reset_251;
static int __lowlim_279;
static int __highlim_280;
static int __input_281;

static void cgMain();
static SIM_STATE tSim={0, 0.0001, 1,0,0.0001,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,cgMain,0,0,0,0,0,0,0};
SIM_STATE *sim=&tSim;

static void cgMain()
{
  int t60;
static int _delayOutBuf290=0;
static int _delayInBuf290=0;
static int _delayOutBuf272=0;
static int _delayInBuf272=0;
  int _clock4_252;
static int _delayOutBuf265=0;
static int _delayInBuf265=0;
static int _delayOutBuf155=0;
static int _delayInBuf155=0;
static int _delayOutBuf305=0;
static CGDOUBLE _delayOutBuf311=0;
static CGDOUBLE _delayOutBuf330=0;
static int _delayOutBuf240=0;
static int _delayInBuf240=0;
static int _delayOutBuf222=0;
static int _delayInBuf222=0;
  int _clock4_202;
static int _delayOutBuf215=0;
static int _delayInBuf215=0;
static int _delayOutBuf138=0;
static int _delayInBuf138=0;
static int _delayOutBuf299=0;
```

```
 static CGDOUBLE _delayOutBuf317=0;
 static CGDOUBLE _delayOutBuf336=0;
 static int _delayOutBuf188=0;
 static int _delayInBuf188=0;
 static int _delayOutBuf84=0;
 static int _delayInBuf84=0;
  int _clock4_69;
 static int _delayOutBuf193=0;
 static int _delayInBuf193=0;
 static int _delayOutBuf104=0;
 static int _delayInBuf104=0;
 static int _delayOutBuf5=0;
 static CGDOUBLE _delayOutBuf323=0;
 static CGDOUBLE _delayOutBuf342=0;
 static int _delayOutBuf179=0;
 static int _delayOutBuf173=0;
 static int _delayOutBuf127=0;
  int t58;
  int t269;
  int t286;
  int _reset_252;
  int t257;
  int t267;
  CGDOUBLE t254;
 static CGDOUBLE _sampBuf253=0;
  int t166;
  int t161;
  int t154;
  CGDOUBLE t349;
  int t22;
  int t21;
  int t26;
  int t11;
  int t313;
  int t38;
  CGDOUBLE t37;
 static CGDOUBLE _sampBuf39=0;
  int t219;
  int t236;
  int _reset_202;
  int t207;
  int t217;
  CGDOUBLE t204;
 static CGDOUBLE _sampBuf203=0;
  int t149;
  int t144;
```

```
  int t137;
  CGDOUBLE t348;
  int t10;
  int t319;
  int t32;
  CGDOUBLE t31;
static CGDOUBLE _sampBuf33=0;
  int t81;
  int t96;
  int _reset_69;
  int t74;
  int t195;
  CGDOUBLE t71;
static CGDOUBLE _sampBuf70=0;
  int t115;
  int t110;
  int t103;
  CGDOUBLE t347;
  int t9;
  int t325;
  int t44;
  CGDOUBLE t43;
static CGDOUBLE _sampBuf45=0;
  int t132;
  int t131;
  int t130;
static int _sampBuf61=0;
  t60 = !(int) _sampBuf61 ;
  clock = t60 ;
  __clock_278 = clock ;
if ( __clock_278 ) _delayOutBuf290=_delayInBuf290;
if ( __clock_278 ) _delayOutBuf272=_delayInBuf272;
  _clock4_252 = __clock_278 ;
if ( _clock4_252 ) _delayOutBuf265=_delayInBuf265;
if ( t60 ) _delayOutBuf155=_delayInBuf155;
  __clock_228 = clock ;
if ( __clock_228 ) _delayOutBuf240=_delayInBuf240;
if ( __clock_228 ) _delayOutBuf222=_delayInBuf222;
  _clock4_202 = __clock_228 ;
if ( _clock4_202 ) _delayOutBuf215=_delayInBuf215;
if ( t60 ) _delayOutBuf138=_delayInBuf138;
  __clock_88 = clock ;
if ( __clock_88 ) _delayOutBuf188=_delayInBuf188;
if ( __clock_88 ) _delayOutBuf84=_delayInBuf84;
  _clock4_69 = __clock_88 ;
if ( _clock4_69 ) _delayOutBuf193=_delayInBuf193;
```

```
 if ( t60 ) _delayOutBuf104=_delayInBuf104;
  __input_281 = c2407ReadAnalogVal(6);
  __highlim_280 = 409 /* 0.05@fx3.16 */;
  t58 = c2407ReadAnalogVal(8);
  t269 = ( __input_281 >(int)((long)(( __highlim_280 +(( t58 )<<1)))>>1));
  __lowlim_279 = 1228 /* 0.15@fx3.16 */;
  t286 = ((int)((long)((( t58 )<<1)- __lowlim_279 ))>>1)> __input_281 );
  __reset_251 = MUL_SHIFT16((-( _delayOutBuf272 *0x1000)+( t286 *0x1000)),( t286
*0x1000),12);
  _reset_252 = __reset_251 ;
  t257 = ((int)MUL_SHIFT16((-( _delayOutBuf290 *0x1000)+( t269 *0x1000)),( t269
*0x1000),12)|(int) _reset_252 );
  t267 = MUL_SHIFT16((-( _delayOutBuf265 *0x1000)+( t257 *0x1000)),( t257 *0x1000),12);
  t254 = ( _reset_252 ?0.:1.);
  if ( t267 ) _sampBuf253 = t254 ;
  t166 = (int)( _sampBuf253 * 8192);
  t161 = _delayOutBuf155 ;
  if ( t161 > 27033) t161 = 27033;
  if ( t161 < -819) t161 = -819;
  t154 = (MUL_SHIFT16((int)((long)(4 /* 0.001@fx4.16 */)<<1), t166 ,13)+((( t166
?(int)((long)( t161 )>>13):0))<<13));
  t349 = ( t161 *9.);
  if ( t349 > 0.999) t349 = 0.999;
  if ( t349 < 0) t349 = 0;
  t22 = c2407ReadAnalogVal(10);
  if ( t22 > 13475) t22 = 13475;
  if ( t22 < 40) t22 = 40;
  pid15.ref_reg2 = (int)((long)(2048 /* 0.5@fx4.16 */)<<3);
  pid15.fb_reg2 = (int)((long)(MUL_SHIFT16( t22 ,4959,14)/* 0.3027@fx4.16 */)<<3);
  piControl(&pid15);
  t21 = pid15.out_reg2 ;
  if ( t21 > 32735) t21 = 32735;
  if ( t21 < 327) t21 = 327;
  t26 = (8192 /* 1@fx3.16 */- (int)((long)( t21 )>>2));
  t11 = ( t349 > t26 );
  t313 = ((int)((long)((-( _delayOutBuf311 *0x2000)+( _sampBuf253 *0x2000)))>>13)>0);
  t38 = ((int)((int)((long)((-( _delayOutBuf305 *0x2000)+( t11 *0x2000)))>>13)>0)|(int) t313 );
  t37 = ( t313 ?0.:1.);
  if ( t38 ) _sampBuf39 = t37 ;
  __input_231 = c2407ReadAnalogVal(4);
  __highlim_230 = 409 /* 0.05@fx3.16 */;
  t219 = ( __input_231 >(int)((long)(( __highlim_230 +(( t58 )<<1)))>>1));
  __lowlim_229 = 1228 /* 0.15@fx3.16 */;
  t236 = ((int)((long)((( t58 )<<1)- __lowlim_229 ))>>1)> __input_231 );
  __reset_201 = MUL_SHIFT16((-( _delayOutBuf222 *0x1000)+( t236 *0x1000)),( t236
*0x1000),12);
```

```
  _reset_202 =  __reset_201 ;
  t207 = ((int)MUL_SHIFT16((-( _delayOutBuf240 *0x1000)+( t219 *0x1000)),( t219
*0x1000),12)|(int) _reset_202 );
  t217 = MUL_SHIFT16((-( _delayOutBuf215 *0x1000)+( t207 *0x1000)),( t207 *0x1000),12);
  t204 = ( _reset_202 ?0.:1.);
  if ( t217 ) _sampBuf203 =  t204 ;
  t149 = (int)( _sampBuf203  * 8192);
  t144 =  _delayOutBuf138 ;
  if ( t144 > 27033) t144 = 27033;
  if ( t144 < -819) t144 = -819;
  t137 = (MUL_SHIFT16((int)((long)(4 /* 0.001@fx4.16 */)<<1), t149 ,13)+((( t149
?(int)((long)( t144 )>>13):0))<<13));
  t348 = ( t144 *9.);
  if ( t348 > 0.999) t348 = 0.999;
  if ( t348 < 0) t348 = 0;
  t10 = ( t348 > t26 );
  t319 = ((int)((long)((-( _delayOutBuf317 *0x2000)+( _sampBuf203 *0x2000)))>>13)>0);
  t32 = ((int)((int)((long)((-( _delayOutBuf299 *0x2000)+( t10 *0x2000)))>>13)>0)|(int) t319 );
  t31 = ( t319 ?0.:1.);
  if ( t32 ) _sampBuf33 =  t31 ;
  __input_91 = c2407ReadAnalogVal(2);
  __highlim_90 = 409 /* 0.05@fx3.16 */;
  t81 = ( __input_91 >(int)((long)(( __highlim_90 +(( t58 )<<1)))>>1));
  __lowlim_89 = 1228 /* 0.15@fx3.16 */;
  t96 = ((int)((long)((( t58 )<<1)-  __lowlim_89 ))>>1)> __input_91 );
  __reset_68 = MUL_SHIFT16((-( _delayOutBuf84 *0x1000)+( t96 *0x1000)),( t96
*0x1000),12);
  _reset_69 =  __reset_68 ;
  t74 = ((int)MUL_SHIFT16((-( _delayOutBuf188 *0x1000)+( t81 *0x1000)),( t81
*0x1000),12)|(int) _reset_69 );
  t195 = MUL_SHIFT16((-( _delayOutBuf193 *0x1000)+( t74 *0x1000)),( t74 *0x1000),12);
  t71 = ( _reset_69 ?0.:1.);
  if ( t195 ) _sampBuf70 =  t71 ;
  t115 = (int)( _sampBuf70  * 8192);
  t110 =  _delayOutBuf104 ;
  if ( t110 > 27033) t110 = 27033;
  if ( t110 < -819) t110 = -819;
  t103 = (MUL_SHIFT16((int)((long)(4 /* 0.001@fx4.16 */)<<1), t115 ,13)+((( t115
?(int)((long)( t110 )>>13):0))<<13));
  t347 = ( t110 *9.);
  if ( t347 > 0.999) t347 = 0.999;
  if ( t347 < 0) t347 = 0;
  t9 = ( t347 > t26 );
  t325 = ((int)((long)((-( _delayOutBuf323 *0x2000)+( _sampBuf70 *0x2000)))>>13)>0);
  t44 = ((int)((int)((long)((-( _delayOutBuf5 *0x2000)+( t9 *0x2000)))>>13)>0)|(int) t325 );
  t43 = ( t325 ?0.:1.);
```

```
if ( t44 ) _sampBuf45 =  t43 ;
t132 = !(int) _sampBuf253 ;
t131 = !(int) _sampBuf203 ;
t130 = !(int) _sampBuf70 ;
if (((int)((long)((-( _delayOutBuf342 *0x2000)+( _sampBuf45 *0x2000)))>>13)>0))
  PADATDIR |= 0x40;
else
  PADATDIR &= 0xFFBF;
if (((int)((long)((-( _delayOutBuf336 *0x2000)+( _sampBuf33 *0x2000)))>>13)>0))
  PBDATDIR |= 0x1;
else
  PBDATDIR &= 0xFFFE;
if (((int)((long)((-( _delayOutBuf330 *0x2000)+( _sampBuf39 *0x2000)))>>13)>0))
  PBDATDIR |= 0x4;
else
  PBDATDIR &= 0xFFFB;
if (((int)((long)((-( _delayOutBuf127 *0x2000)+( t130 *0x2000)))>>13)>0))
  PADATDIR |= 0x80;
else
  PADATDIR &= 0xFF7F;
if (((int)((long)((-( _delayOutBuf173 *0x2000)+( t131 *0x2000)))>>13)>0))
  PBDATDIR |= 0x2;
else
  PBDATDIR &= 0xFFFD;
if (((int)((long)((-( _delayOutBuf179 *0x2000)+( t132 *0x2000)))>>13)>0))
  PBDATDIR |= 0x8;
else
  PBDATDIR &= 0xFFF7;

if ( __clock_278 )
  _delayInBuf290= t269 ;
if ( __clock_278 )
  _delayInBuf272= t286 ;
if ( _clock4_252 )
  _delayInBuf265= t257 ;
if ( t60 )
  _delayInBuf155= t154 ;
_delayOutBuf305= t11 ;
_delayOutBuf311= _sampBuf253 ;
_delayOutBuf330= _sampBuf39 ;
if ( __clock_228 )
  _delayInBuf240= t219 ;
if ( __clock_228 )
  _delayInBuf222= t236 ;
if ( _clock4_202 )
  _delayInBuf215= t207 ;
```

```
  if ( t60 )
    _delayInBuf138= t137 ;
  _delayOutBuf299= t10 ;
  _delayOutBuf317= _sampBuf203 ;
  _delayOutBuf336= _sampBuf33 ;
  if ( __clock_88 )
    _delayInBuf188= t81 ;
  if ( __clock_88 )
    _delayInBuf84= t96 ;
  if ( _clock4_69 )
    _delayInBuf193= t74 ;
  if ( t60 )
    _delayInBuf104= t103 ;
  _delayOutBuf5= t9 ;
  _delayOutBuf323= _sampBuf70 ;
  _delayOutBuf342= _sampBuf45 ;
  _delayOutBuf179= t132 ;
  _delayOutBuf173= t131 ;
  _delayOutBuf127= t130 ;
  _sampBuf61 =  t60 ;

}

main()
{
  noIntegrationUsed = 1;
  EALLOW;
  WDCR=0x00ef;       // Disable Watchdog
  asm(" clrc DBGM");
  PCLKCR |= 0x8;
  HISPCP = 0x0;    // HCLK  = 150 MHZ
  EDIS;
 simInit( &tSim );
  startSimDsp();
  installInterruptVec(-2,7,timerFunc );
  TIMER2PRD = 0x3a98; // 32-bit Timer Period Low
  TIMER2PRDH = 0x0; // 32-bit Timer Period High
  TIMER2TCR |= 0x4020; //Interrupt enable, Timer Reset
  EALLOW;
  PIECTRL = 1; // Enable PIE Interrupts
  EDIS;
  IER |= 0x2000; //CPU Interrupt enable
  enable_interrupts();  // Global Start Interrupts
  dspWaitStandAlone();
  return 0;
}
```