NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIFORNIA

THESIS

POWER ELECTRONIC BUILDING BLOCK NETWORK SIMULATION TESTBED STABILITY CRITERIA AND HARDWARE VALIDATION STUDIES

by

Michael G. Badorf

June, 1997

Thesis Advisor: Robert W. Ashton

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE June 1997	3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE
POWER ELECTRONIC BUILDING BLOCK NETWORK SIMULATION TESTBED STABILITY CRITERIA AND HARDWARE VALIDATION STUDIES

5. FUNDING NUMBERS

6. AUTHOR(S) Michael G. Badorf

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(maximum 200 words)*

Naval power distribution has principally used an AC network to supply loads. With the advent of new power electronic devices, the focus has shifted to employing a DC distribution system that eliminates large transformers and mechanical switching devices and enhances the survivability of the platform. The Power Electronic Building Block (PEBB) Network Simulation Testbed currently under construction at the Naval Postgraduate School is a study into the feasibility of such DC systems.

The objective of this thesis was to perform theoretical and simulation-based analysis to establish quantitative criteria for PEBB Testbed stability. These criteria were then used to develop a set of hardware studies to investigate the interaction of components within the PEBB Testbed. Finally, the hardware studies were utilized to verify PEBB Testbed performance.

Principal conclusions of this research included that the PEBB Testbed demonstrated stability under all simulated loading conditions. Follow-on testing of the PEBB Testbed confirmed that the simulations correlated well with hardware implementation. In addition, the hardware validation studies revealed that switching harmonics had a considerable effect on the system output.

14. SUBJECT TERMS buck converters, dc-to-dc converters, power electronic building blocks, control of power converters, parallel dc-to-dc converter operation	15. NUMBER OF PAGES 162
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18 298-102

Approved for public release; distribution is unlimited.

POWER ELECTRONIC BUILDING BLOCK NETWORK SIMULATION TESTBED STABILITY CRITERIA AND HARDWARE VALIDATION STUDIES

Michael G. Badorf
Lieutenant, United States Navy
B.S., United States Naval Academy, 1990

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
June 1997

Author: _____
Michael G. Badorf

Approved by: _____
Robert W. Ashton, Thesis Advisor

_____
John G. Ciezki, Second Reader

_____
Herschel H. Loomis, Jr., Chairman
Department of Electrical and Computer Engineering

ABSTRACT

Naval power distribution has principally used an AC network to supply loads. With the advent of new power electronic devices, the focus has shifted to employing a DC distribution system that eliminates large transformers and mechanical switching devices and enhances the survivability of the platform. The Power Electronic Building Block (PEBB) Network Simulation Testbed currently under construction at the Naval Postgraduate School is a study into the feasibility of such DC systems.

The objective of this thesis was to perform theoretical and simulation-based analysis to establish quantitative criteria for PEBB Testbed stability. These criteria were then used to develop a set of hardware studies to investigate the interaction of components within the PEBB Testbed. Finally, the hardware studies were utilized to verify PEBB Testbed performance.

Principal conclusions of this research included that the PEBB Testbed demonstrated stability under all simulated loading conditions. Follow-on testing of the PEBB Testbed confirmed that the simulations correlated well with hardware implementation. In addition, the hardware validation studies revealed that switching harmonics had a considerable effect on the system output.

TABLE OF CONTENTS

I. INTRODUCTION

Naval power distribution has principally used AC networks to supply loads. With the advent of new power electronic devices, the focus has shifted to employing a DC distribution system that eliminates large transformers and mechanical switching devices and enhances the survivability of the platform. The Power Electronic Building Block (PEBB) Network Simulation Testbed currently under construction at the Naval Postgraduate School will be used to study the feasibility of such DC systems.

The proposed architecture for this shipboard power distribution scheme is shown in Figure (1-1). In this distribution network the main feeders are DC. The ship is

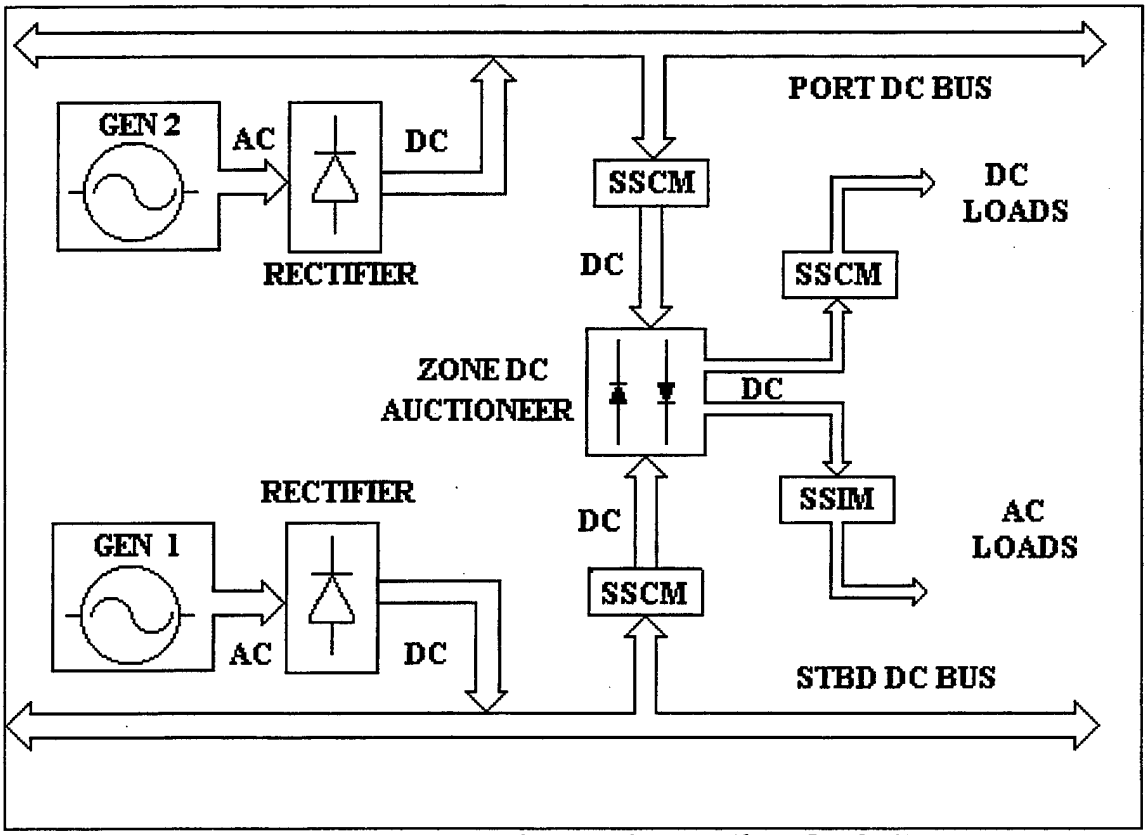


Figure 1-1, Integrated Power System (from [Ref. 1])

divided into zones with each zone containing common energy conversion devices fed from the DC busses. The DC power is distributed via port and starboard busses from the source(s) into the separate zones. Each zone contains a number of Ship Service Converter (SSCM) and Inverter Modules (SSIM). The SSCM is used in each zone to step-down the distribution bus voltage to a regulated level for use in the zone. In this way the SSCM inserts intelligence into the system by acting to buffer, preregulate and fault protect each zone. Electric loads within the zone are fed by either SSCMs or SSIMs depending on the load's requirement for DC or AC power.

The focus of this thesis is on the design of SSCMs and their subsequent integration with SSIMs in a DC distribution network. Coincident to this research is the development of system stability criteria. As a result, theoretical and simulation-based analysis will be performed to establish quantitative criteria for system stability. These criteria will then be used to develop a set of hardware studies to investigate the interaction of components within the PEBB Network Simulation Testbed. Finally, the hardware studies will be conducted to verify system performance.

The documentation detailing this research is organized into six chapters. In Chapter II, the basic design of the SSCM power section is presented. Chapter III deals with the development of SSCM analog and digital closed-loop state-difference control algorithms. Pole placement and gain selection for these controllers are investigated in Chapter IV. With the SSCM gains determined, detailed simulations of integrated SSCM/SSIM operations are documented in Chapter V. This chapter also includes hardware study results for some of the simulation configurations studied.

Circuit operations and schematics for the SSCM power section and controller are covered in Chapter VI. The final chapter contains a summary of the research work, notable conclusions and recommendations for future work.

II. POWER SECTION DESIGN

A. TOPOLOGY

This project focused on the design, implementation and testing of high-bandwidth DC-to-DC buck converters. Figure (2-1) illustrates the basic circuit topology.

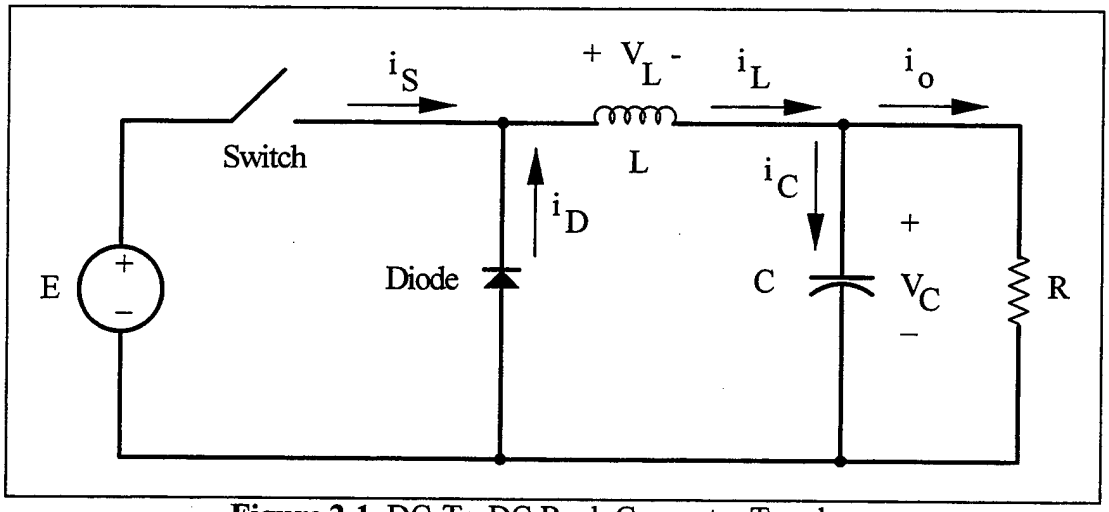


Figure 2-1, DC-To-DC Buck Converter Topology

Following a full development in reference [2], it can be shown that the buck converter acts as a DC step-down transformer. The reduction in voltage is governed by the duty cycle of the switch according to

$$V_C = DE \tag{2-1}$$

where D is the duty cycle. From Equation (2-1), it is evident that any desired output voltage can be attained by controlling the duty cycle.

System design called for the application of this topology as part of the DC Zonal Electrical Distribution (DC ZED) Network envisioned for the Surface Combatant for the twenty-first century. As part of the basic research into DC ZED, a Power Electronic Building Block (PEBB) Network Simulation Testbed is being assembled at the Naval

Postgraduate School. A general topology of this network is pictured in Figure (2-2). From the diagram, it can be seen that two Source Buck Converters condition rectified three-phase power and provide the network with regulated DC power. This power is directed to loads supplied by two Load Buck Converters and four Auxiliary Resonant Commutated Pole (ARCP) Inverters.

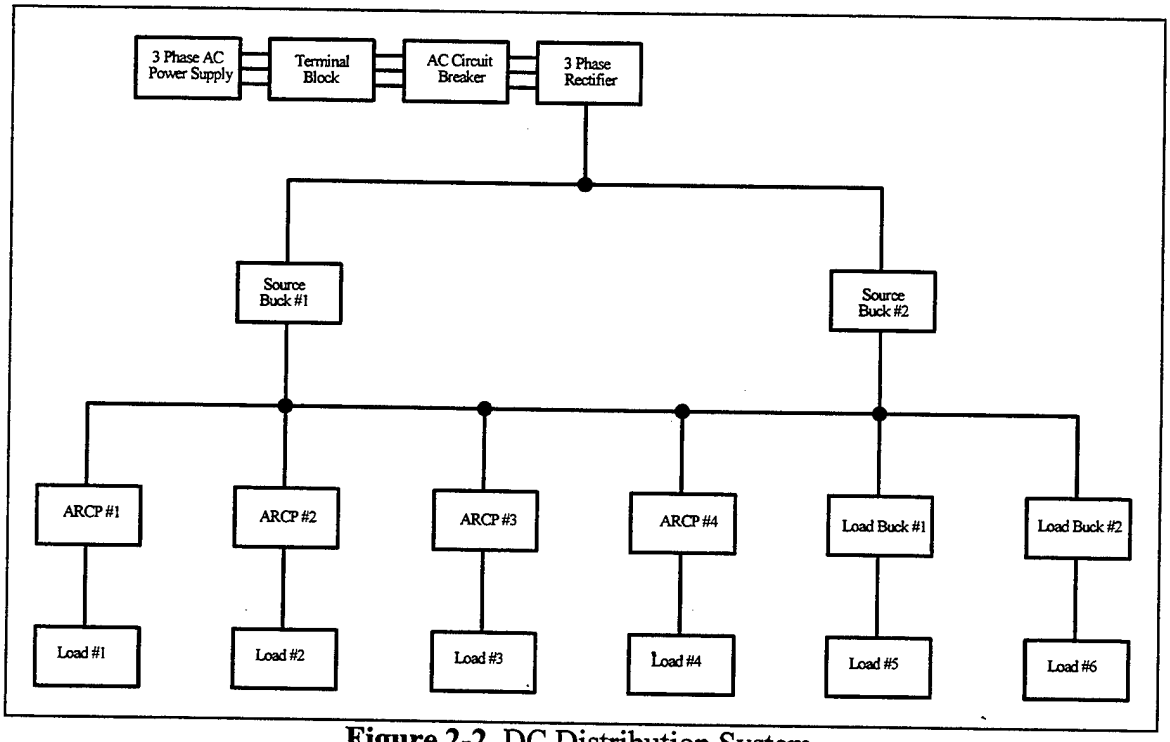


Figure 2-2, DC Distribution System

Specifications for the Source and Load Buck Converters are given below. Detailed design descriptions of the ARCP units are given in reference [3].

B. SPECIFICATIONS AND COMPONENT SELECTION

Minimum design specifications for the Source and Load Buck Converters were provided by the Naval Surface Warfare Center, Carderock Division/Annapolis Detachment. Table (2-1) summarizes the requirements. Using this information,

component sizing and selection was performed. The final results of this analysis are documented in Table (2-2). Analysis of each component selection is given below.

Parameter	Source Buck	Load Buck
Maximum Output Power	9 kW	3 kW
Switching Frequency	5-25 kHz Hard Switched	5-25 kHz Hard Switched
Input (nominal)	400 V_{DC} / 22.5 A	300 V_{DC} / 10 A
Output (nominal)	300 V_{DC} / 30 A	208 V_{DC} / 14.5 A
Continuous Operations	10% to 100% Loading	10% to 100% Loading

Table 2-1, Buck Converter Specifications

Component/Parameter	Source Buck	Load Buck
Switch / Diode	International Rectifier IRGT1090U06 600V/90A IGBT Power Modules	International Rectifier IRGT1090U06 600V/90A IGBT Power Modules
Input Filter Capacitor	Sprague Powerlytic 2000 μF, 450 V_{DC}	Two Sprague Powerlytic 230 μF, 450 V_{DC}, Paralleled
Input Filter Inductor	0.425 mH, hand wound	0.425 mH, hand wound
Output Capacitor	Sprague Powerlytic 400 μF, 450 V_{DC}	Sprague Powerlytic 400 μF, 450 V_{DC}
Output Inductor	0.760 mH, hand wound 0.875 mH, hand wound	1.3 mH, hand wound
Switching Frequency	20 kHz	20 kHz
Output Voltage Ripple	≤ 1 %	≤ 1 %

Table 2-2, Component Selection Results

1. Switch and Diode Selection

For both the Source and Load Buck Converters, International Rectifier 600V/90A IGBT Power Modules were chosen based on their high current density, rugged design, and simple gate drive. In addition, the INT-A-pak "half-bridge" packaging lends itself to the buck converter topology. One power module contains two IGBT / diode pairs

allowing co-location of the switch and diode components. Appendix (A) contains the data sheet for this component.

2. Output Inductor Sizing and Switching Frequency

a. Source Buck Converter

Inductor sizing was done to maintain continuous inductor current over a load range of 10% → 100%. Based on this range, the most limiting condition for continuous inductor current occurs at minimum load. To maintain continuous operations at this point, the inductor must be wound to meet or exceed the critical inductance for the circuit. This value is given in reference [2] by

$$L_{crit} = \frac{T \cdot R}{2} \cdot (1 - D) \tag{2-2}$$

where T is the switching period and R is the resistance at minimum loading.

Analysis of Equation (2-2) shows that two of the input parameters are set. From Equation (2-1), the duty cycle is 75%. In addition, load resistance is 100 Ω based on a nominal output voltage of 300 V. As a result, only the switching period, or conversely the switching frequency, must be set. Utilizing Table (2-1), the switching frequency was chosen to be 20 kHz. This selection corresponds to a switching period of 50 μsec and provides the following advantages:

- minimization of required inductor size
- minimization of the inductor ripple current given by $\Delta I = \frac{E - V_C}{L} \cdot DT$
- elimination of audible switching noise of the buck converter

- maintains a 5 kHz margin to the maximum hard-switched limit of the IGBTs.

Substituting into Equation (2-2), critical inductance was found to be 625 μH. The actual inductances achieved for the two Source Buck Converters were 760 μH and 875 μH respectively.

This sizing was not made any larger than necessary because of the tradeoff between desired inductance and the size required to achieve that inductance. At full load, inductor current has a 30 A dc offset. This dc component drives a single, wound inductor core sized to L_{crit} into saturation. As a result, each inductor was made from several cores wound in series. Cores were wound to minimize saturation effects while maximizing inductance per core. The overall inductance is then given by summing each of the series core inductances. Reference [4] contains core sizing information.

b. Load Buck Converter

Inductor sizing was done to maintain continuous inductor current over a load range of 10% → 100%. A 20 kHz switching frequency was chosen for the reasons mentioned above. Substituting appropriate values into Equation (2-2) yielded a critical inductance of 1.1 mH. In actuality, a 1.3 mH inductance value was achieved for each Load Buck Converter. Contrary to the Source Buck Converters, core saturation due to dc offset was not a concern since the maximum load current is well below saturation levels. As a result each inductor was wound using one core thus saving space in the Load Buck Converter topology.

3. Output Capacitor Sizing

a. Source Buck Converter

Capacitor selection was based on maintaining less than 1% voltage ripple on the Source Buck Converter output. From reference [2], percent voltage ripple is given by:

$$\frac{\Delta V_c}{V_c} = \frac{T^2(1-D)}{8LC} \cdot 100 \qquad (2\text{-}3)$$

Substituting the appropriate values, the minimum output capacitance is found to be 10 μF. Based on this small size, rapid transient response can be expected. Computer simulation and gain selection detailed in Chapter IV, however, shows that the transient voltage response is unacceptably soft at such low output capacitance values. Utilizing these results, 400 μF Sprague electrolytic capacitors were selected. This choice provided the required transient response and built in a design margin for testing above the 10% → 100% → 10% load transient used for analysis in Chapter IV.

b. Load Buck Converter

Capacitor selection was based on maintaining less than 1% voltage ripple on the Load Buck Converter output. Utilizing Equation (2-3), minimum output capacitance was determined to be 7.5 μF. Using computer simulation and gain selection results detailed in Chapter IV, 400 μF Sprague electrolytic capacitors were selected to provide the required voltage transient response.

4. Input Filter Sizing

a. Source Buck Converter

One circuit element not shown in the basic buck converter topology of Figure (2-1) is the input LC lowpass filter. The input filter is documented in Figure (6-1) of Chapter VI. Initial filter design focused on component selection to provide 10 dB of attenuation at 60 Hz. Component sizing and dc attenuation concerns, however, make this design impractical. As a result, ac ripple rejection analysis shifted to a combined filter/controller loop speed method. In this arrangement, the input LC filter decouples the rectifier source from the Source Buck Converter input by at least 10 dB of attenuation for frequencies 360 Hz and above. The controller loop speed of the Source Buck Converter is then tasked with providing ac ripple rejection below 360 Hz. Chapter IV contains a discussion on controller loop speed.

Central to the selection of capacitor and inductor values was the placement of the resonant peak of the input LC filter. The resonance acts to amplify low frequency components in the supply voltage. As a result, peak placement must minimize this amplification for the frequencies of concern. Since the supply voltage is rectified 3-phase power, these frequencies are 60 Hz, 120 Hz, and 360 Hz. Matlab analysis revealed that a resonant peak placed at 180 Hz provides the desired response. This value was obtained using a capacitance of 2000 μF and an inductance of 391 μH. In actuality, a 173 Hz resonant peak was achieved using a 2000 μF capacitor and 425 μH inductor.

b. Load Buck Converter

The input filter to the Load Buck Converters was designed to provide decoupling from the output of the Source Buck Converters. As a result, the LC combination had to be chosen to place the resonant peak of the Load Buck Converter input filter below the slowest controller poles of the Source Buck Converters. Based on the pole placement/gain selections detailed in Chapter IV, the resonant peak of the input LC filter was set at 360 Hz using a capacitance of 460 μF and an inductance of 425 μH.

C. SUMMARY

With the basic power sections of the Source and Load Buck Converters established, the design effort focused on the development and implementation of an appropriate control scheme. As will be shown in Chapter III, open-loop operations of a buck converter are extremely underdamped under transient conditions. Therefore, satisfactory system transient response must be obtained by "closing the loop". Once the required control algorithm has been established, gain selections and simulations can be performed to produce stability criteria for hardware testing.

III. CONTROL ALGORITHM DEVELOPMENT

A. OPEN-LOOP ABCD STATE SPACE MODEL

The first step in controller development required the determination of the open-loop state space equations. In state space analysis, energy storage devices dictate the state variables. Referring to Figure (2-1) in Chapter II, the obvious state variable choices are capacitor voltage and inductor current. With the switch shut, the state equations are:

$$\frac{dv_C}{dt} = \frac{1}{C}\left[i_L - \frac{v_C}{R}\right] \tag{3-1}$$

$$\frac{di_L}{dt} = \frac{1}{L}\left[E - v_C\right] \tag{3-2}$$

With the switch open, the state equations become:

$$\frac{dv_C}{dt} = \frac{1}{C}\left[i_L - \frac{v_C}{R}\right] \tag{3-3}$$

$$\frac{di_L}{dt} = \frac{1}{L}\left[0 - v_C\right] \tag{3-4}$$

Combining like equations into an averaged model yields

$$\frac{dv_C}{dt} = \frac{1}{C}\left[i_L - \frac{v_C}{R}\right] \tag{3-5}$$

$$\frac{di_L}{dt} = \frac{1}{L}\left[dE - v_C\right] \tag{3-6}$$

where the duty cycle, d, is given by:

$$d = \frac{t_{switch\ on}}{t_{switch\ on} + t_{switch\ off}} = \frac{t_{on}}{T_{switch}} \tag{3-7}$$

with T_{switch} = switching period. The resulting ABCD state space representation is:

$$\begin{bmatrix} \dot{v}_C \\ \dot{i}_L \end{bmatrix} = \begin{bmatrix} \frac{-1}{RC} & \frac{1}{C} \\ \frac{-1}{L} & 0 \end{bmatrix} \begin{bmatrix} v_C \\ i_L \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{E}{L} \end{bmatrix} d \tag{3-8}$$

$$\begin{bmatrix} v_C \\ i_L \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_C \\ i_L \end{bmatrix} \tag{3-9}$$

When operated in the open-loop configuration, the buck converter output voltage and inductor current responses are highly underdamped in the presence of load transients. Figure (3-1) illustrates the system response for a buck converter operating at a 75% duty cycle with E = 400 V_{DC}.

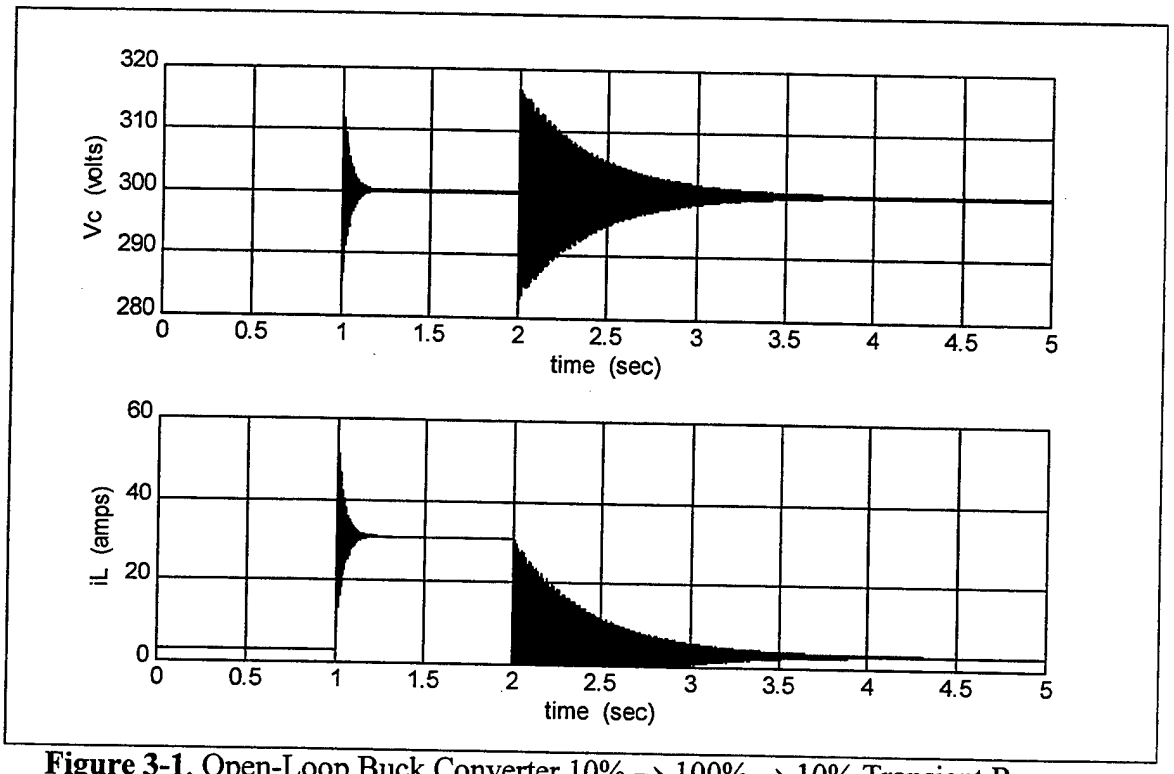


Figure 3-1, Open-Loop Buck Converter 10% → 100% → 10% Transient Response

B. KASSAKIAN ALGORITHM

The large overshoots, long settling times and high-frequency oscillations shown in Figure (3-1) are not acceptable in an interconnected power system. Clearly, a need has been established to "close the loop". The foundation for the final control algorithm comes from reference [5] and is given by:

$$\tilde{d}(t) = -h_i \tilde{i}_L(t) - h_v \tilde{v}_o(t) - h_n \int \tilde{v}_o(\xi) d\xi \qquad (3\text{-}10)$$

This multiloop control law states that duty cycle perturbations are a function of perturbations in the average output voltage and average inductor current. By judicious selection of the constant feedback gains h_i, h_v, and h_n, the desired closed-loop response can be obtained.

The last term in Equation (3-10) represents integral control action. Assuming that the buck converter closed-loop system is stable and driven only by constant signals, all variables must settle to constant values in the steady state. As a result, the integrand in Equation (3-10) must settle to zero to prevent the integral from contributing a time-varying term to the equation. Therefore, so long as the feedback gains are chosen to stabilize the system, zero steady-state error is assured in the output voltage.

With Equation (3-10) serving as a model for the feedback path, the closed-loop state equations were determined. This step was done by performing small-signal analysis of Equation (3-8). The resulting small-signal state space representation is:

$$\begin{bmatrix} \dot{\hat{i}}_L \\ \dot{\hat{v}}_C \\ \dot{\hat{d}} \end{bmatrix} = \begin{bmatrix} 0 & \frac{-1}{L} & \frac{E}{L} \\ \frac{1}{C} & \frac{-1}{RC} & 0 \\ \frac{-h_v}{C} & \left(\frac{h_i}{L} + \frac{h_v}{RC} - h_n\right) & \frac{-h_i E}{L} \end{bmatrix} \begin{bmatrix} \hat{i}_L \\ \hat{v}_C \\ \hat{d} \end{bmatrix} \tag{3-11}$$

By taking the determinant of the system matrix and setting it equal to zero, a third-order characteristic polynomial is obtained. The eigenvalues of this polynomial determine the closed-loop pole positions. As a result, proper selection of the control gains will give the desired pole locations.

C. STATE DIFFERENCE IMPLEMENTATION IN HARDWARE

1. Basic Implementation

Implementation of Equation (3-10) required generating the perturbation terms, $\tilde{v}_o(t)$ and $\tilde{i}_L(t)$, using state difference methods. In this approach, $\tilde{v}_o(t)$ and $\tilde{i}_L(t)$ are represented by difference terms. For example, the output voltage perturbation is given by

$$\tilde{v}_o(t) = v_C - v_{ref} \tag{3-12}$$

where v_{ref} is the desired output voltage of the buck converter. As can be seen from Equation (3-12), $\tilde{v}_o(t)$ is zero in the steady state. If a load transient occurs, however, v_C changes and produces a difference error that is used in Equation (3-10) to affect the duty cycle perturbation term. Likewise, the $\tilde{i}_L(t)$ difference term is

$$\tilde{i}_L(t) = i_O - i_L \tag{3-13}$$

where i_O is output load current and i_L is inductor current. Substituting Equation (3-12) and Equation (3-13) into Equation (3-10) results in the state difference equation:

$$\tilde{d}(t) = -h_i\left(i_O - i_L\right) - h_v\left(v_C - v_{ref}\right) - h_n \int\left(v_C - v_{ref}\right)d\xi \tag{3-14}$$

The development of Equation (3-13) raises an important point. Unlike Equation (3-12), Equation (3-13) does not employ the use of a reference term to act as a baseline from which to measure perturbations. Rather, to generate difference errors, Equation (3-13) relies on the fact that inductor current will lag output current during a load transient. If inductor current is an average value, then Equation (3-13) represents the perturbation in average inductor current as developed in Equation (3-10). Digital controller implementation allows the user to meet this requirement as will be shown in Chapter IV. Analog controller implementation, however, does not provide an adequate method for obtaining the true average inductor current since filtering introduces additional poles and unwanted time delays. As a result, Equation (3-13) actually implements an instantaneous state difference for analog control. This point raises important gain selection issues outlined in Chapter IV.

2. Feedforward Gain

Up to this point, the small-signal duty cycle algorithm has been determined. To drive the switch, however, an expression for the overall duty cycle is required:

$$d(t) = D + \tilde{d}(t) \tag{3-15}$$

Equation (3-15) states that the overall duty cycle is made up of a large-signal component, D, and a small-signal component, $\tilde{d}(t)$, which was given in Equation (3-14). An expression must now be developed for D.

Manipulating Equation (2-1) from Chapter II, it is apparent that D is the ratio of output voltage to input voltage. This ratio could be set as a constant in the controller and methodically added to $\tilde{d}(t)$ every switching cycle, but the full effectiveness of algorithm would not be achieved. In order to enhance the control algorithm, the large-signal duty cycle term is implemented as

$$D_{ss} = \frac{v_{ref}}{e(t)} \tag{3-16}$$

where v_{ref} is the desired output voltage of the buck converter, and $e(t)$ is the time-varying input voltage.

Equation (3-16) illustrates an important point. A feedforward gain is introduced by using $e(t)$ in the calculation of Equation (3-16). This gain compensates for the input voltage perturbations neglected in the development of the small-signal duty cycle algorithm. Since $d(t)$ can instantaneously adjust to input voltage changes due to the feedforward gain, it acts to filter unwanted input noise from the output of the buck converter. The details of this point are expanded in Chapter IV. Incorporating Equations (3-16) and (3-14) into Equation (3-15), the total duty cycle algorithm is given by:

$$d(t) = D_{ss} - h_i\left(i_O - i_L\right) - h_v\left(v_C - v_{ref}\right) - h_n \int\left(v_C - v_{ref}\right)d\xi \tag{3-17}$$

3. Source Buck Converter House Curve

With Equation (3-17) in hand, a controller simulation and implementation can be undertaken for an isolated buck converter. Parallel operations, however, will be required of the Source Buck Converters illustrated in Figure (2-2). Paralleling offers the advantages of supplying higher loads and enhancing reliability of the system. Before this step can be accomplished, Equation (3-17) must be modified to allow load sharing between buck converters. To accomplish this action, a house curve is built into the system control and is depicted in Figure (3-2). As illustrated in the slope of Figure (3-2), every three amp increase in load current causes the output voltage to drop by one volt.

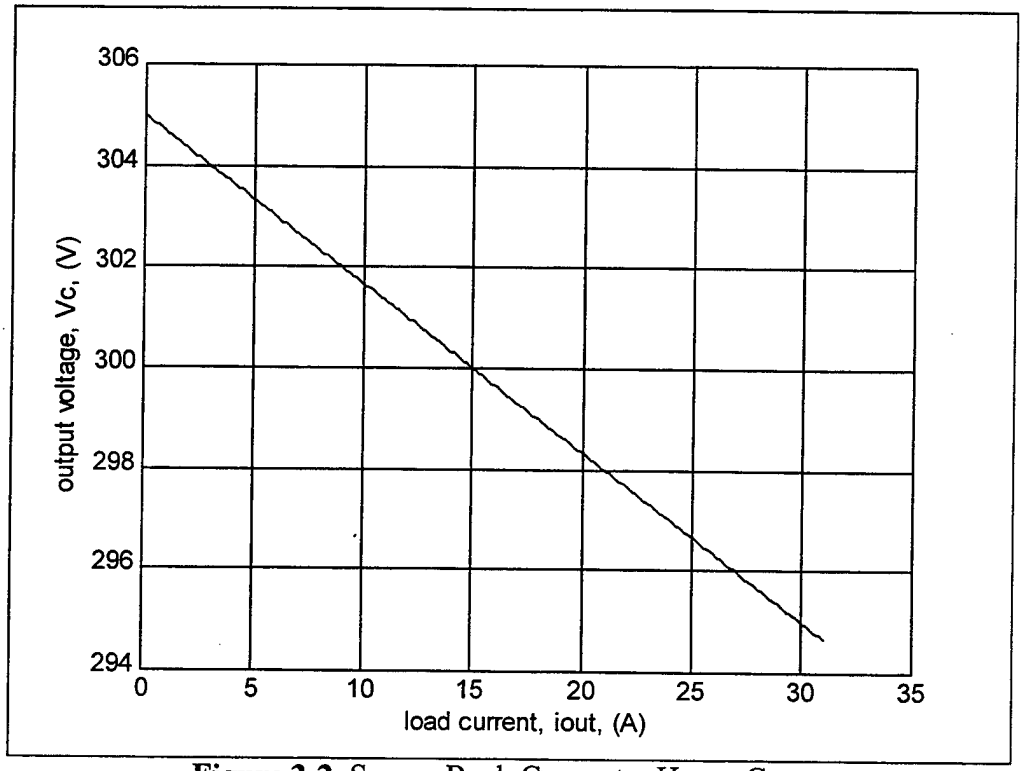


Figure 3-2, Source Buck Converter House Curve

Over the no-load to full-load range, output voltage drops ten volts. The slope of the house curve was selected to provide an adequate load sharing profile while maintaining output voltage sufficiently high to supply system loads.

The house curve is added by introducing the load current, scaled by the desired slope of the house curve, into the voltage and integral gain terms of Equation (3-17). The house curve could also be introduced into the feedforward gain term. Its inclusion, however, makes the overall duty cycle so responsive to load current changes that unacceptable voltage and current overshoots occur during transient load response. Equation (3-18) is the final form of the analog control algorithm:

$$d(t) = D_{ss} - h_i\left(i_O - i_L\right) - h_v\left(v_C - v_{ref} + \frac{i_O}{3}\right) - h_n \int\left(v_C - v_{ref} + \frac{i_O}{3}\right)d\xi \qquad (3\text{-}18)$$

This result will be utilized in Chapter IV to simulate the analog control of the Source Buck Converter. For use with the Load Buck Converter, the house curve is removed from Equation (3-18), and it is modified into the following digital form:

$$d(n) = D_{ss}(n) - h_i\left(i_O(n) - i_L(n)\right) - h_v v_{error}(n) - h_n v_{\text{int}}(n) \qquad (3\text{-}19)$$

where

$$v_{\text{int}}(n) = \frac{T_{sample}}{2}\left(v_{error}(n) + v_{error}(n-1)\right) + v_{\text{int}}(n-1) \qquad (3\text{-}20)$$

and

$$v_{error}(n) = v_C(n) - v_{ref}(n) \qquad (3\text{-}21)$$

D. SUMMARY

This chapter detailed the development of state difference control algorithms for the Source and Load Buck Converters. It was shown that proper selection of the control gains will give the desired pole locations. As a result, the next step in the system design is directed towards determining the gain magnitudes utilizing computer simulations. As

Chapter IV will show, this process involves many tradeoffs to achieve the required system transient response.

IV. CONTROLLER GAIN SELECTION

A. ANALOG CONTROLLER GAIN SELECTION

1. Computer Modeling

Controller gains and operational characteristics were established using computer modeling. First-order gain selection was done using Simulink, MATLAB's dynamic system modeling tool. Detailed analysis was then performed using models developed in the Advanced Continuous Simulation Language (ACSL).

Simulink, in combination with supporting MATLAB m-files, provided the means for iteratively manipulating pole placement and fine tuning gain selection. The closed-loop Source Buck Converter model is pictured in Appendix (B). The open-loop plant is represented by the state space averaged ABCD matrices. It should be noted that the state space averaging process eliminates the switch by "averaging" the modeling equations for the two topologies. As a result, characteristics like ramping inductor current are seen as a dc average. The main control algorithm developed in Chapter III "closes the loop". Utilizing this model and the three MATLAB m-files provided in Appendix (B), Source Buck Converter gain selection performance was measured by running a 10% $\rightarrow$ 100% $\rightarrow$ 10% load transient with a hard input voltage source.

Following initial development in Simulink, the controller gains were evaluated under the same 10% $\rightarrow$ 100% $\rightarrow$ 10% load transient in the detailed ACSL model. Built on the foundation of FORTRAN, ACSL allows modeling of the switched Source Buck

Converter. As a result, the "details" of system performance are brought to the surface. Model assumptions included:

- instantaneous switching
- switch and diode losses modeled as voltage drops
- ideal passive components

The code for the ACSL simulation is provided in Appendix (C).

2. Source Buck Converter Transient Specifications

Specifications for the Source Buck Converter did not include requirements for system transient response. In a general sense, transient response should have short rise and settling times and little or no overshoot. In addition, the loop speed of the controller should be such that low-frequency (< 360 Hz) ac ripple is filtered from the input. This general system performance has been quantified in Table (4-1). The numbers shown are a "first cut" on adequate system response.

Figure of Merit	Value
rise time (t_r)	$\leq$ 1 msec
2% settling time (t_s)	< 5 msec
maximum percent overshoot (M_p)	5 %

Table 4-1, System Transient Response Figures of Merit

3. Closed Loop Model

Before pole placement was done, the Source Buck Converter small-signal closed-loop block diagram was developed. Figure (4-1) gives the final result. The picture shows that two effects can be studied. The first involves analysis of the small-signal output voltage response due to $\hat{d}$. This step is done assuming input voltage is constant.

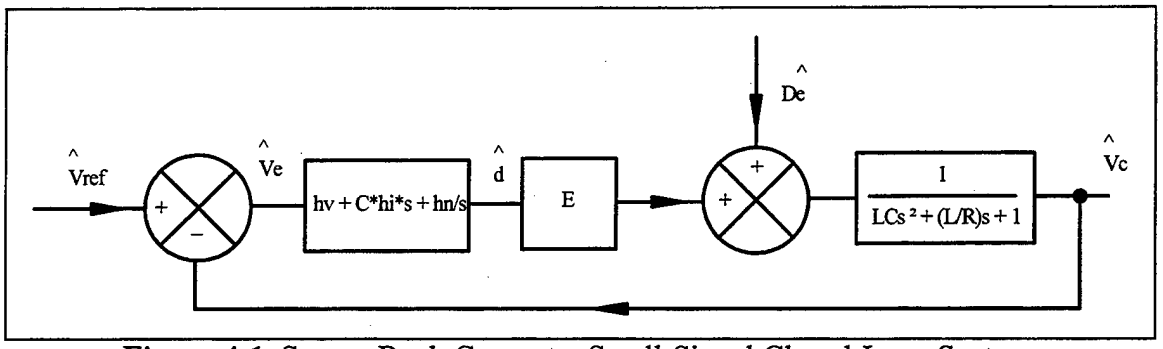


Figure 4-1, Source Buck Converter Small-Signal Closed-Loop System

As a result, $\hat{De}$ is set to zero. Derivation of the closed-loop transfer function for this model resulted in Equation (4-1):

$$\frac{\hat{v}_C}{\hat{v}_{ref}} = \frac{\frac{E}{LC}\left(Ch_i s^2 + h_v s + h_n\right)}{s^3 + \left(\frac{1}{RC} + \frac{Eh_i}{L}\right)s^2 + \left(\frac{1+Eh_v}{LC}\right)s + \frac{E\,h_n}{LC}} \tag{4-1}$$

The second effect involves analysis of the input voltage ripple response. This step is done assuming the duty cycle is constant. As a result, $\hat{v}_{ref}$ is set to zero. Derivation of the closed-loop transfer function for this model resulted in Equation (4-2):

$$\frac{\hat{v}_C}{\hat{e}} = \frac{\frac{D}{LC}s}{s^3 + \left(\frac{1}{RC} + \frac{Eh_i}{L}\right)s^2 + \left(\frac{1+Eh_v}{LC}\right)s + \frac{E\,h_n}{LC}} \tag{4-2}$$

As shown below, both of these cases will be used to determine the overall closed-loop performance of the Source Buck Converter.

4. Pole Placement And Gain Selection

To determine pole placement, the Bessel Prototype model given in reference [6] was employed. The poles for a third-order system are given by

$$s = -0.9420w_o, (-0.7455 \pm 0.7112i)w_o$$

where w_o is the desired closed-loop bandwidth. With the poles established, a third-order polynomial can be formed, and term-by-term matching can be done with the denominator of Equation (4-1) to find the controller gains.

Selection of w_o, and thus the gains, was set by three factors. First, w_o must be sized to ensure the pole farthest in the left half plane is at least a factor of ten smaller than the radian switching frequency to prevent unwanted controller interactions. Second, selection of w_o should not require an excessive duty cycle control effort. This effect causes increased noise in the voltage output due to a beating action set up between the faster duty cycle response and the slower output capacitor voltage response. Third, w_o must be large enough to produce the required response times of Table (4-1).

Based on these criteria, the final pole locations and controller gains were determined using $w_o = 3250$ rad/sec. The results are documented in Table (4-2).

Closed-Loop Pole Locations	$s = -3062, -2423 \pm 2311i$
Proportional Voltage Gain	$h_v = 0.017$
Integral Voltage Gain	$h_n = 26.09$
Proportional Current Gain	$h_i = 0.015$

Table 4-2, Source Buck Converter Closed-Loop Poles and Gains

Substituting these gain values into Equation (4-1), the system closed-loop frequency response shown in Figure (4-2) was produced. From the magnitude response it can be seen that frequencies beyond 7000 rad/sec (~1114 Hz) are attenuated by the 20 dB/decade rolloff. Up to this point, the response shows little or no attenuation. In fact, from 2000 rad/sec to 7000 rad/sec, a maximum gain of 2 dB occurs. The input ripple response of Figure (4-3), however, compensates for this lack of attenuation. This plot shows a

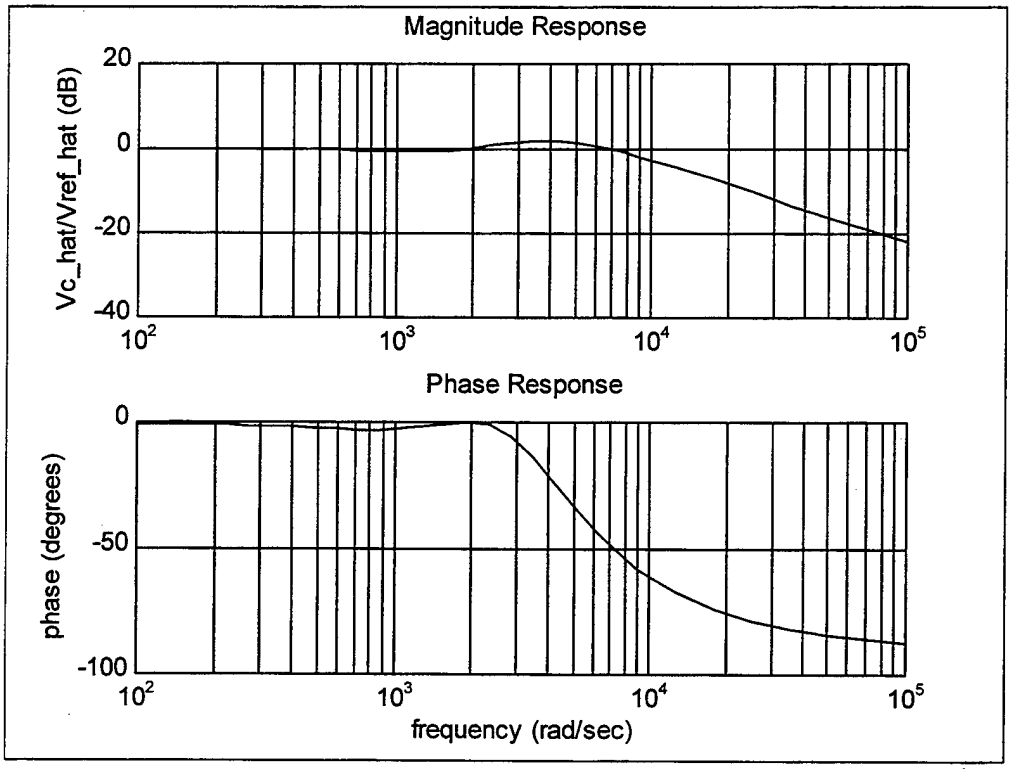


Figure 4-2, Source Buck Converter Closed-Loop Frequency Response: $\hat{v}_C / \hat{v}_{ref}$

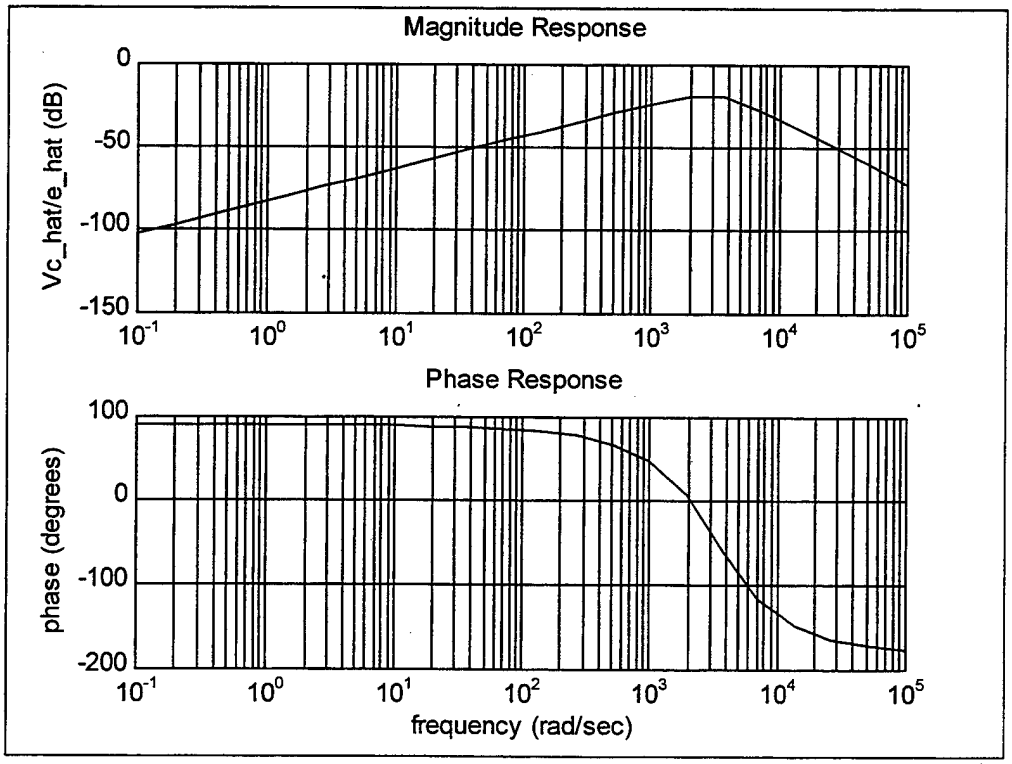


Figure 4-3, Source Buck Converter Closed-Loop Frequency Response: $\hat{v}_C / \hat{e}$

minimum attenuation of 20 dB in the 360 Hz frequency range. As a result, the combined effect of both responses yields proper Source Buck Converter closed-loop operations. The MATLAB transient response is given in Figure (4-4). Analysis of Figure (4-4) and the MATLAB code results shows that all the requirements of Table (4-1) were met. Next, the transient response was verified using the detailed ACSL simulation.

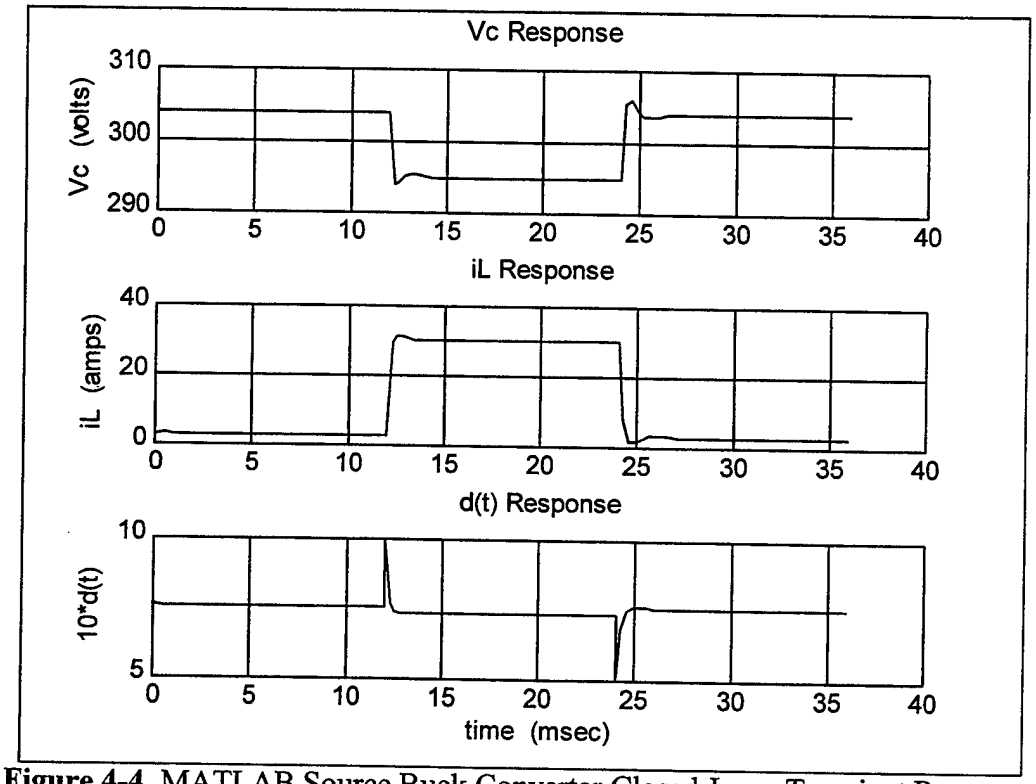


Figure 4-4, MATLAB Source Buck Converter Closed-Loop Transient Response

Substituting the Table (4-2) gains into the ACSL model produced the transient response pictured in Figure (4-5). The most notable feature of the ACSL graphs is the influence of detailed switching on inductor current and duty cycle. The inductor current response was expected. The duty cycle response, on the other hand, was not. In fact, results close to that of the duty cycle plot in Figure (4-4) were anticipated. The reason for the duty cycle band in Figure (4-5) goes back to the fact that the inductor current perturbation term in Equation (3-18) is not a true average. Rather, instantaneous

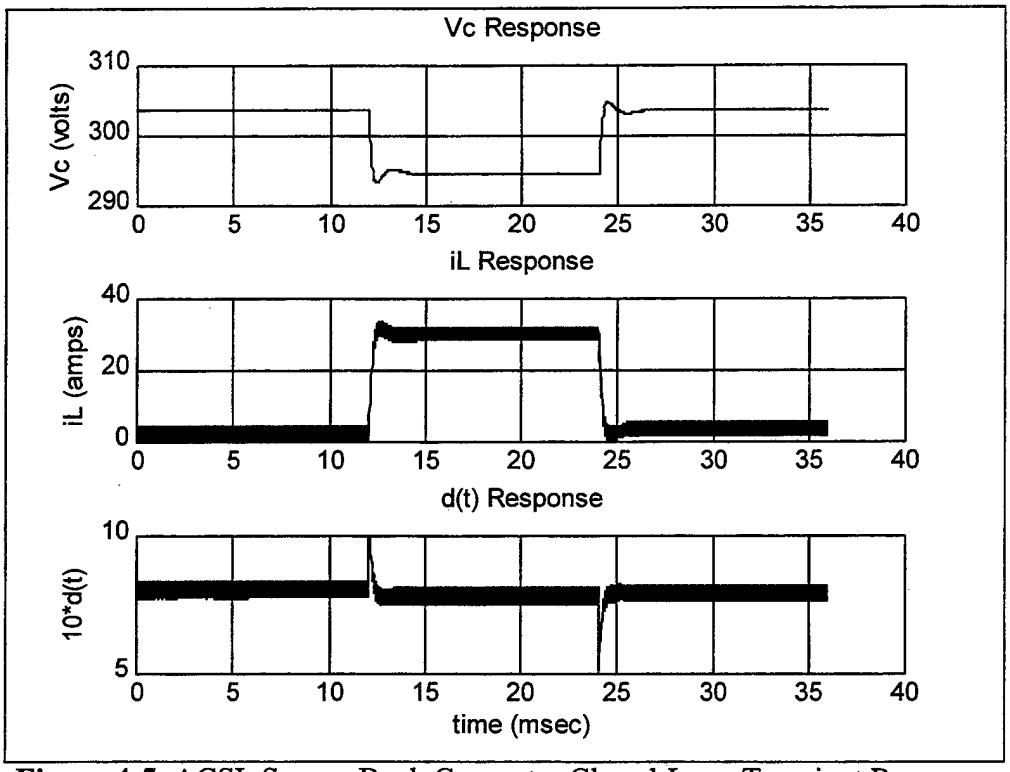


Figure 4-5, ACSL Source Buck Converter Closed-Loop Transient Response

inductor current and load current are subtracted to produce the perturbation term. In steady state, this difference is the inductor ac current ripple. As a result, this ripple is introduced into the duty cycle calculation, and the duty cycle waveform of Figure (4-5) is produced. The end result of this effect is the introduction of low amplitude (< 0.1 V) 20 kHz harmonics into the output. Therefore, it is imperative to minimize the magnitude of h_i while still meeting the response of Table (4-1). Bench testing showed that a value of h_i less than 0.02 gave acceptable results.

5. Effect of Output Capacitor Sizing

To this point, pole placement and gain selection have been done utilizing an output inductance of 760 μH and an output capacitance of 400 μF. It has been shown,

however, that output capacitance can be as low as 10 μF while still satisfying the 1% output ripple requirement. As a result, justification for the 400 μF choice is required.

Using the Bessel Prototype model described above, system transient response was analyzed at additional output capacitance values of 10 μF, 140 μF, 230 μF, and 400 μF. A value of w_o = 3250 rad/sec was used for all cases to maintain the current gain less than or equal to its present value based on the discussion in the previous paragraph.

Table (4-3) gives the gains for each of the output capacitances mentioned above. Figure (4-6) summarizes the transient responses.

Parameter	C = 10 μF	C = 140 μF	C = 230 μF	C = 400 μF
h_i	0.0131	0.015	0.015	0.015
h_v	-0.0021	0.0043	0.0087	0.017
h_n	0.65	9.13	15.00	26.09

Table 4-3, Capacitor Sizing Effects on the Source Buck Converter Gains

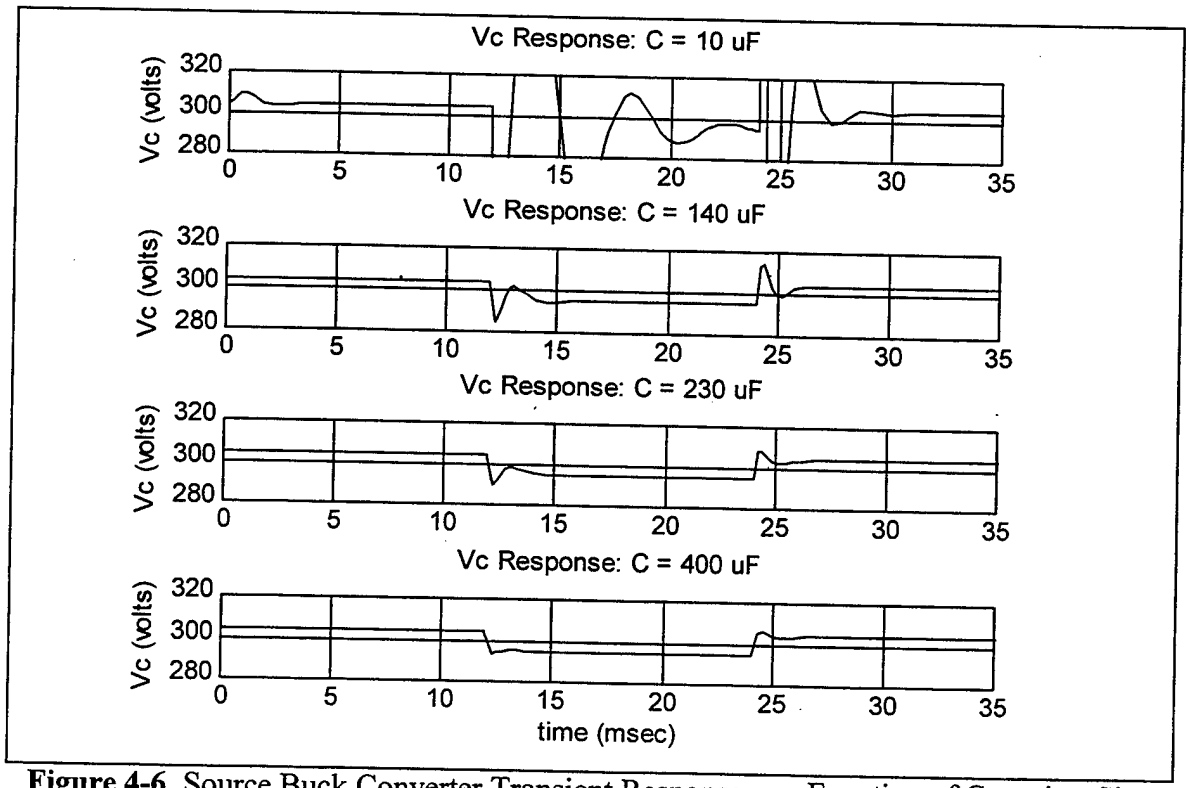


Figure 4-6, Source Buck Converter Transient Response as a Function of Capacitor Size

From the data provided, it can be seen that the 10 μF case was unacceptable. All of the remaining cases, however, were well within the Table (4-1) requirements. Based on overshoot and settling characteristics, however, the 400 μF capacitor was used in the hardware implementation. This selection was made to prevent undesirable buck converter interactions during parallel operations and to ensure adequate system response at load transient testing above the 10% → 100% → 10% scenario used in this analysis.

B. DIGITAL CONTROLLER GAIN SELECTION

1. Computer Modeling

As with the analog controller, the digital controller was modeled using Simulink and ACSL. The simulation limitations discussed above also apply to this section. In the digital simulation, the continuous-time controller inputs were sampled once each switching period using a zero-order hold and applied to the digital control algorithm given by Equation (3-19). The calculated duty cycle is then applied to the IGBT gate drive on the next switching period. The Simulink block diagram and MATLAB m-files are contained in Appendix (B). The code for the ACSL simulation is provided in Appendix (C).

2. Pole Placement and Gain Selection

The design parameters of Table (4-1) and the simulation sequence employed above were also used for the digital controller design. In addition, the limitations on w_o also hold here. Unlike the analog case, however, current gain selection does not play as significant a role. This fact was due to the use of a true average for the inductor current

perturbation term in Equation (3-19). Since the analog inputs to the digital controller are sampled once each switching period, the inductor current sample becomes an average value as long as the sample point occurs at the same place during each switching period. Of course the ramping nature of inductor current may cause the sampled average to be slightly different from the actual average depending where on the ramp the sample is taken. Any offset, however, is compensated for by the digital integrator. This argument is equally applicable to the case where multiple samples are taken and averaged during the switching period.

3. Summary of Digital Controller Gain Selections

Table (4-4) provides a summary of final gain selections for the Load Buck Converter controller. The design centered on using a 400 μF capacitor on the output with w_o = 1900 rad/sec. The value of w_o was set based on a maximum duty cycle limitation of 0.95 for the Universal Controller detailed in reference [7]. Since the controller was built for dual Auxiliary Resonant Commutated Pole (ARCP) Inverter and Buck Converter operations, the duty cycle must incorporate dead time to prevent a direct short between ARCP switches during the switching interval. As a result, duty cycle is limited to 0.95. The w_o selection prevents the duty cycle from exceeding this limit so that linear operation is maintained. Overall, Table (4-1) transient response is achieved. See the analog

Closed-Loop Pole Locations	$s = -942, -746 \pm 711i$
Proportional Voltage Gain	$h_v = 0.000869$
Integral Voltage Gain	$h_n = 1.733$
Proportional Current Gain	$h_i = 0.0105$

Table 4-4, Load Buck Converter Closed-Loop Poles and Gains

controller case above for the detailed gain and capacitor selection procedure. Figure (4-7) documents the ACSL transient load response.

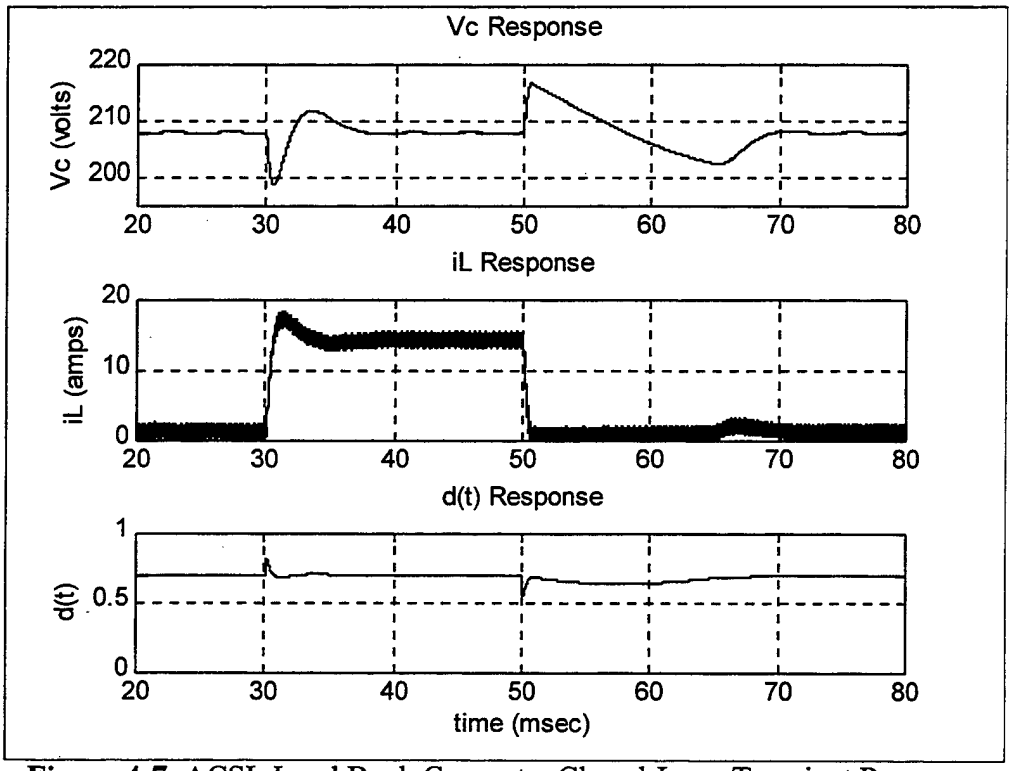


Figure 4-7, ACSL Load Buck Converter Closed-Loop Transient Response

C. SUMMARY

With the controller gains established for the Source and Load Buck Converters, simulation of the PEBB Network can begin. This analysis will be used to identify stability criteria for the network. With this information in place, interesting PEBB Network component interactions can be investigated.

V. SIMULATION AND HARDWARE RESULTS

A. BACKGROUND

With the proper analog and digital controller gains established, detailed PEBB Network simulations were developed and conducted in ACSL. Representative models, identified as "modes" in this chapter, were established based on seven basic PEBB Network hardware configurations. Model assumptions included:

- rectified 3-phase power input to Source Buck Converters
- instantaneous switching
- no switching noise
- switch and diode losses modeled as voltage drops
- inductor and capacitor equivalent series resistance (ESR) modeled
- DC bus resistive-inductive link (RL-link) effects not modeled

The RL-link effect was not considered due to a tradeoff between simulation run time and the simulation time step. The time step could be made small enough to model the RL-links, but the simulation run time would have been unnecessarily large. In addition, the RL-link contribution to the overall system stability is negligible for the components considered based upon the small size of the line inductance compared to the much larger inductance value of the Load Buck Converter input filter.

For each of the seven modes described in this chapter, two transient analysis cases were considered. The first case investigated a step change in load resistance on the output of the Source Buck Converter. Load Buck Converter output response was then

observed to identify if its controller could regulate the desired output voltage and current. No noticeable effects or stability issues were documented for any of the seven topologies (modes), and therefore, this transient study was not considered further. The second transient case performed a load change on the output of the Load Buck Converter and/or a conventional inverter. Overall system response was then observed. The results for this case are documented below for each of the seven modes. Representative ACSL code is provided for Mode 7 operation in Appendix (C).

B. MODE 1 OPERATION

Mode 1 is defined as a Source Buck Converter operating in series with a Load Buck Converter. Figure (5-1) illustrates the overall system model. The only difference

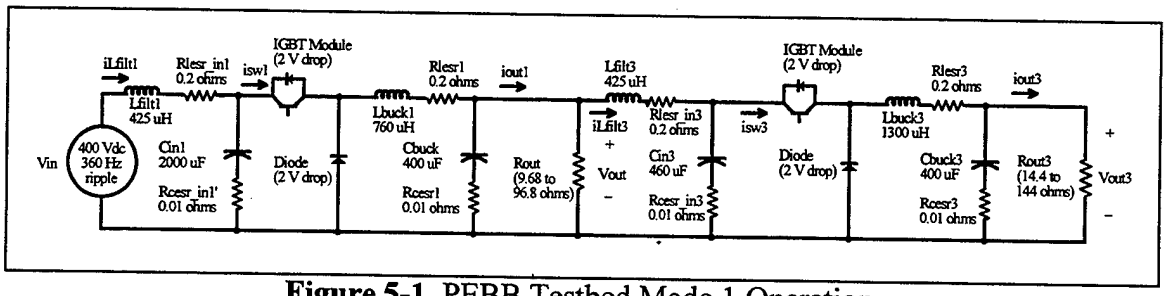


Figure 5-1, PEBB Testbed Mode 1 Operation

component-wise between Figure (5-1) and the ACSL code resulted from the need to represent the input filter to the Source Buck Converter differently in the simulation. Due to an algebraic loop formed between the ACSL main program, the buck converter macro and the analog controller macro, the input filter capacitor and its associated ESR had to be modeled by a capacitor in parallel with an equivalent resistor. As a result, the Rcesr_in1' component in Figure (5-1) is represented by a 2.47 kΩ parallel resistance in the ACSL code.

System transient analysis focused on investigating a step load change from 300 W to 3 kW on the output of the Load Buck Converter. Figure (5-2) documents the results. As can be seen, the output voltage and current of the Source Buck Converter (Vout and iout1) have a somewhat more pronounced oscillation at the transient edges before settling to steady-state values. This result is due to a "ringing" effect set up between the output of the Source Buck Converter and the input LC filter of the Load Buck Converter. In fact, this effect is clearly discernible in iout1 in the steady state. The transient performance of the Load Buck Converter is the same as that predicted by Figure (4-7) in Chapter IV.

With system stability demonstrated, a hardware validation study was conducted. Figure (5-3) shows the actual system response for a 300 W to 2.5 kW load change. The reason for the difference between the simulated 3 kW maximum load value and the actual 2.5 kW load value was based upon the sizing of test load banks available in the laboratory. Comparison of the simulation and hardware trials demonstrates reasonably good correlation between predicted and actual system response. Note that the voltages in Figure (5-3) are ac coupled. The major difference between the plots comes from the switching harmonics and noise of the actual system operation. The principle source of the noise comes from the operation of the IGBT switches.

C. MODE 2 OPERATION

Mode 2 is defined as a Source Buck Converter operating in series with an inverter. Figure (5-4) illustrates the overall system model. A hard-switched inverter was used to model the ARCP unit so that simulation times could be kept to a reasonable

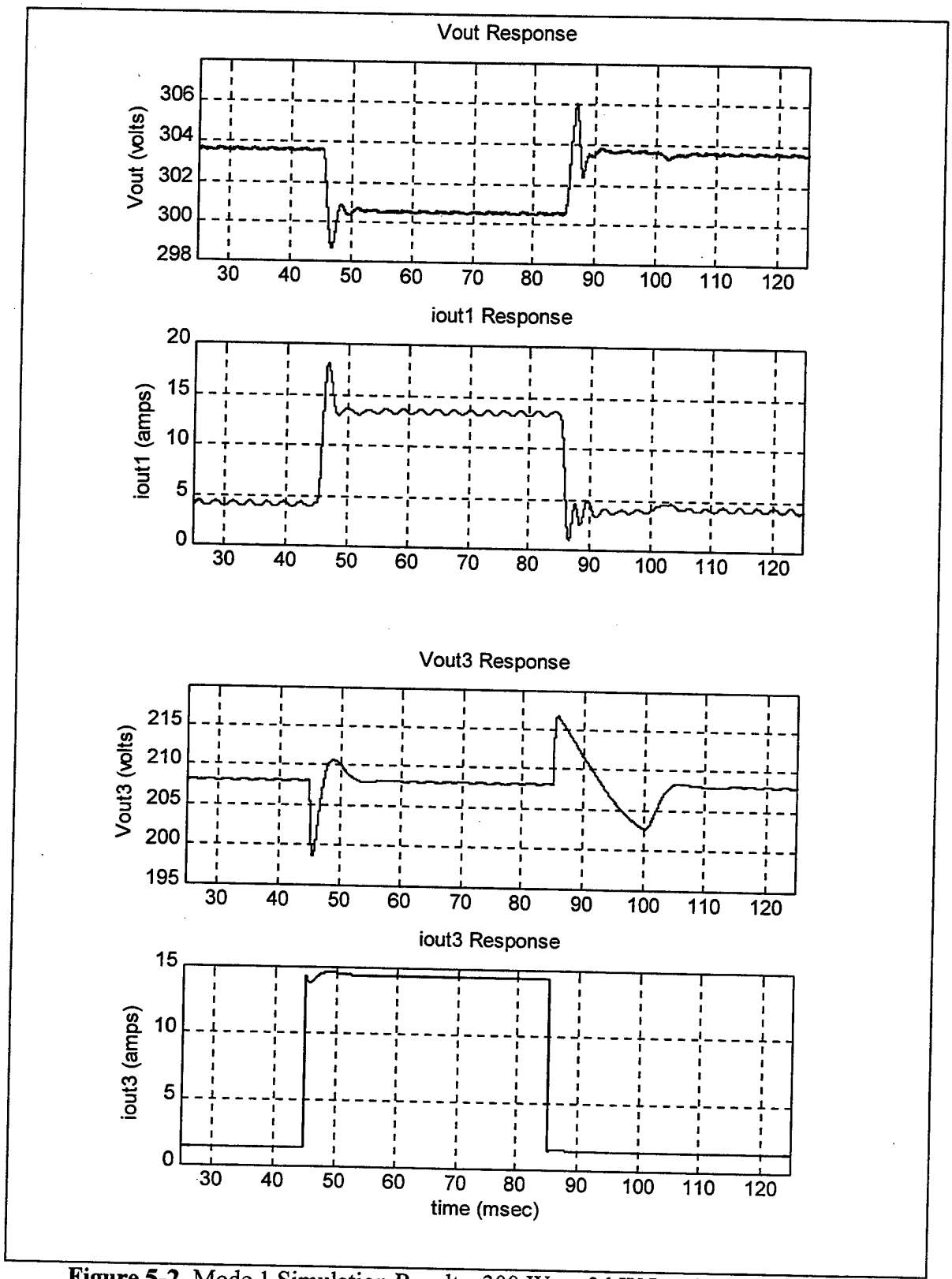


Figure 5-2, Mode 1 Simulation Results: 300 W → 3 kW Load Buck Transient

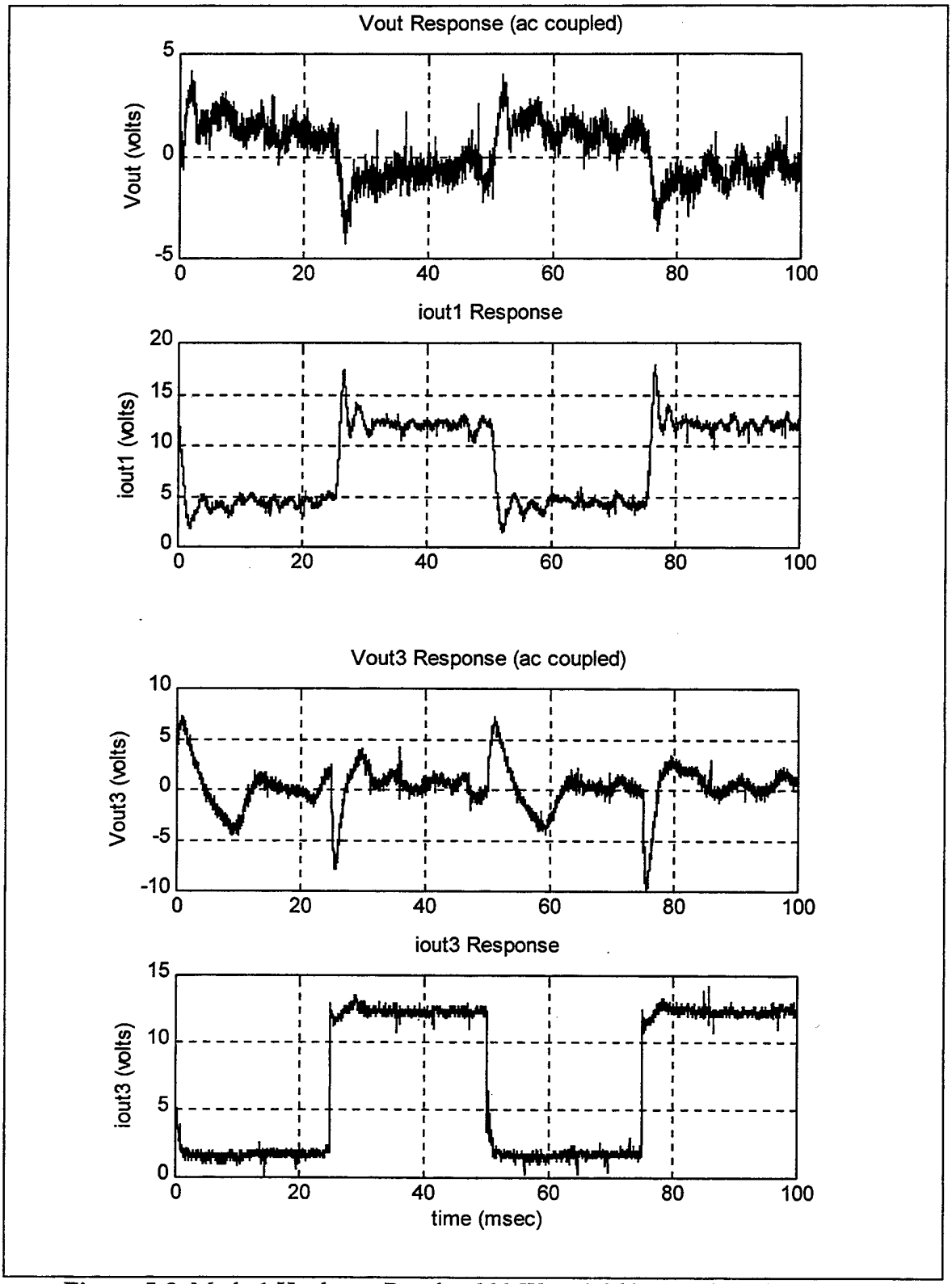


Figure 5-3, Mode 1 Hardware Results: 300 W → 2.5 kW Load Buck Transient

length. Since the gross system dynamics between an ARCP and a conventional inverter are nearly the same this assumption was reasonable to make. The inverter was controlled by a current feedback controller implemented in the stationary reference frame. A complete discussion on the derivation of this model is given in reference [8]. As with Mode 1 operation, the input filter capacitor and its associated ESR had to be modeled as an equivalent parallel combination in the ACSL code.

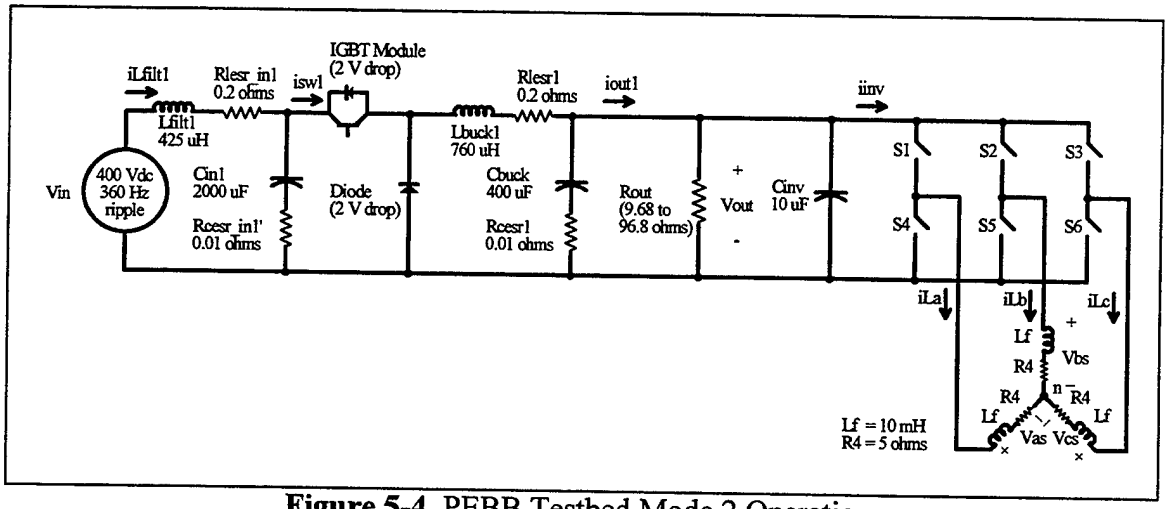


Figure 5-4, PEBB Testbed Mode 2 Operation

System transient analysis focused on investigating a step change in load from 2.5 kW to 6 kW at the inverter output. This was performed by changing the amplitude values of the reference phase currents from 10 A to 24 A. Figure (5-5) documents the results. As can be seen, significant switching harmonics have been introduced into the output voltage and current responses of the Source Buck Converter. In addition, this effect is aggravated at higher levels of inverter loading. The output of the inverter is as expected for a current controlled unit.

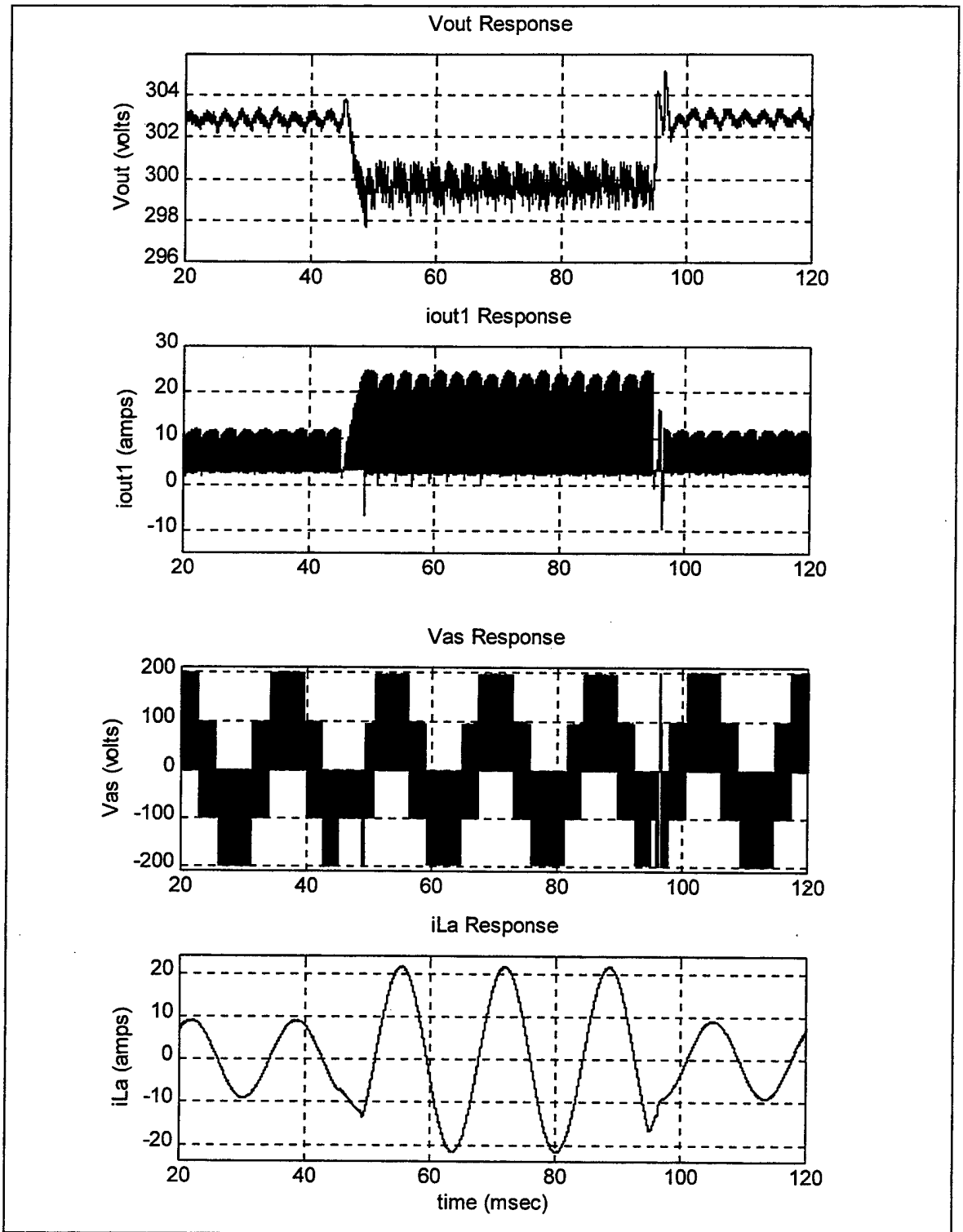


Figure 5-5, Mode 2 Simulation Results: 2.5 kW → 5.3 kW Inverter Transient

One point to note is that the inverter current response does not achieve the desired steady-state values. This result stems from the use of stationary reference frame control. Although it is easy to implement, reference [8] shows that steady-state errors of 11% can be expected. A hardware validation study was not conducted for this case because the ARCP closed-loop control has not been fully developed at this time. It is anticipated that future research efforts will provide the hardware study results to validate these waveforms.

D. MODE 3 OPERATION

Mode 3 is defined as a Source Buck Converter supplying both a Load Buck Converter and an inverter. Figure (5-6) illustrates the overall system model. The same inverter model described in Mode 2 operation was used for this case. As with Mode 1 operation, the input filter capacitor and its associated ESR had to be modeled as an equivalent parallel combination in the ACSL code.

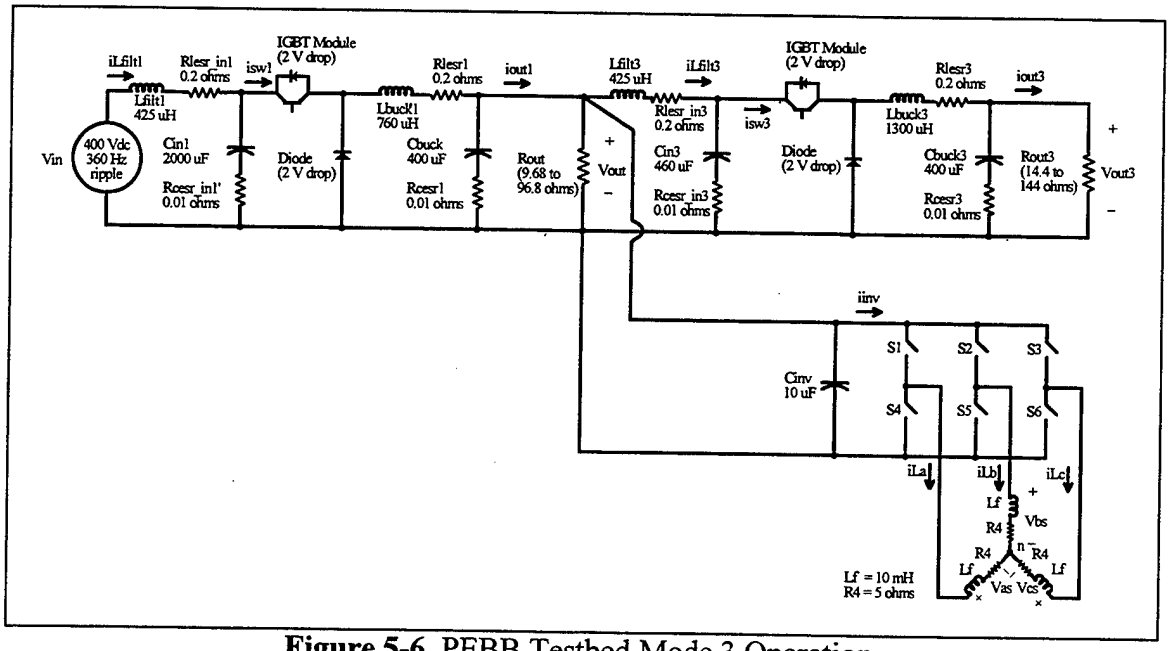


Figure 5-6, PEBB Testbed Mode 3 Operation

Transient analysis focused on investigating system response to the load changes discussed in Mode 1 and Mode 2 operation. The step change for the Load Buck Converter occurred at 45 msec while the step change in inverter loading was specified at 125 msec. Figure (5-7) documents the results. The plots show a system response that is the composite of the results achieved in the previous two modes of operation. As with Mode 2 operation, a hardware validation study was not conducted since the ARCP closed-loop control algorithm is still under development.

E. MODE 4 OPERATION

Mode 4 is defined as parallel Source Buck Converter operations. Figure (5-8) illustrates the overall system model. As described in Mode 1 operation, the input filter capacitors and their associated ESRs had to be modeled as equivalent parallel combinations in the ACSL code.

Transient analysis for this case focused on investigating a load change from 2.25 kW to 18 kW and observing system response. Figure (5-9) documents the results. From the plots, it can be seen that the parallel units share the total load, iout, equally. In addition, the output voltage response covers the full range of house curve operation as expected.

An interesting occurrence to note in the individual buck converter current responses, iout1 and iout2, is the "mirror" effect seen at the transient edges. This effect is best observed in the system transient at the 85 msec mark. At this point, iout1 settles to

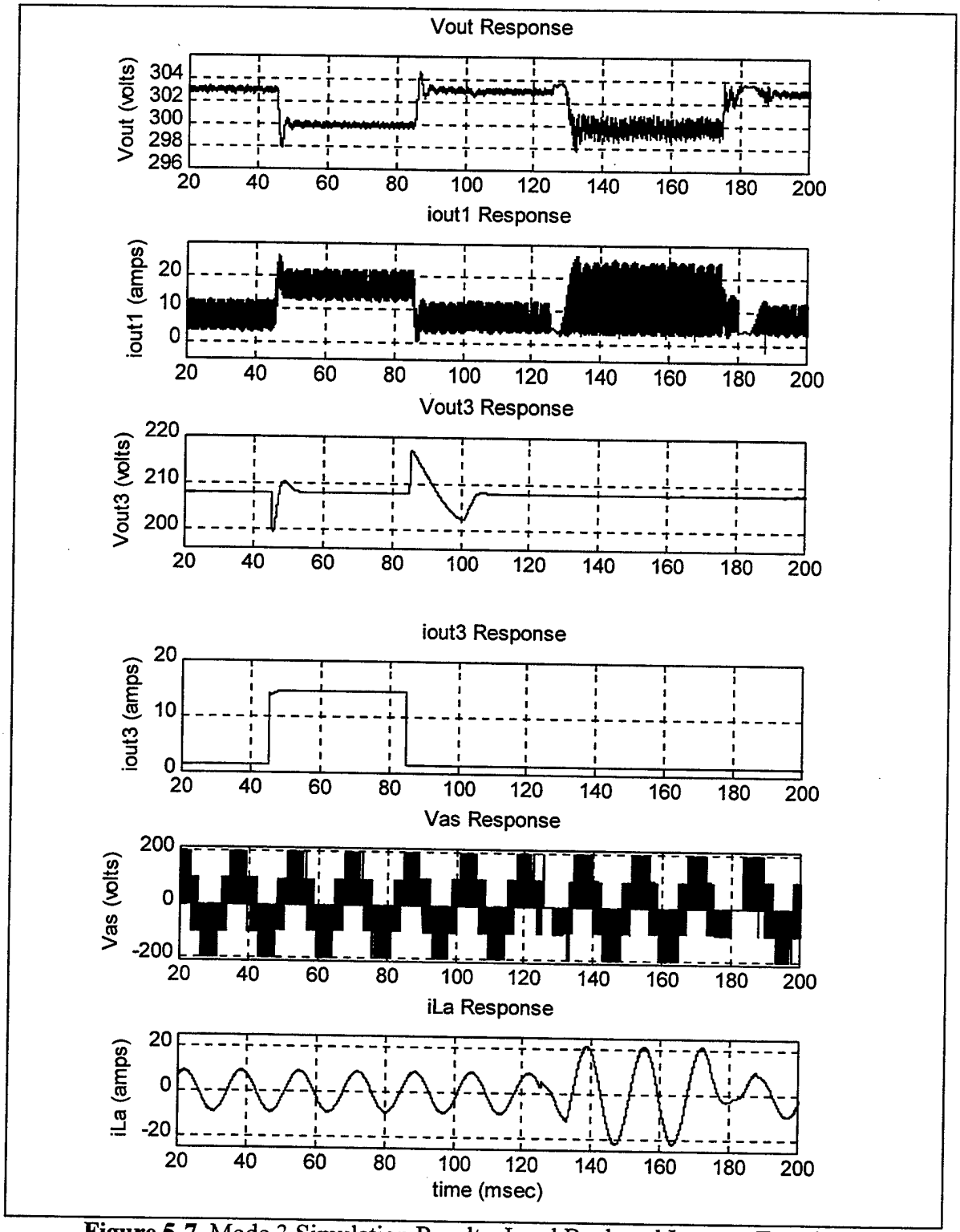


Figure 5-7, Mode 3 Simulation Results: Load Buck and Inverter Transients

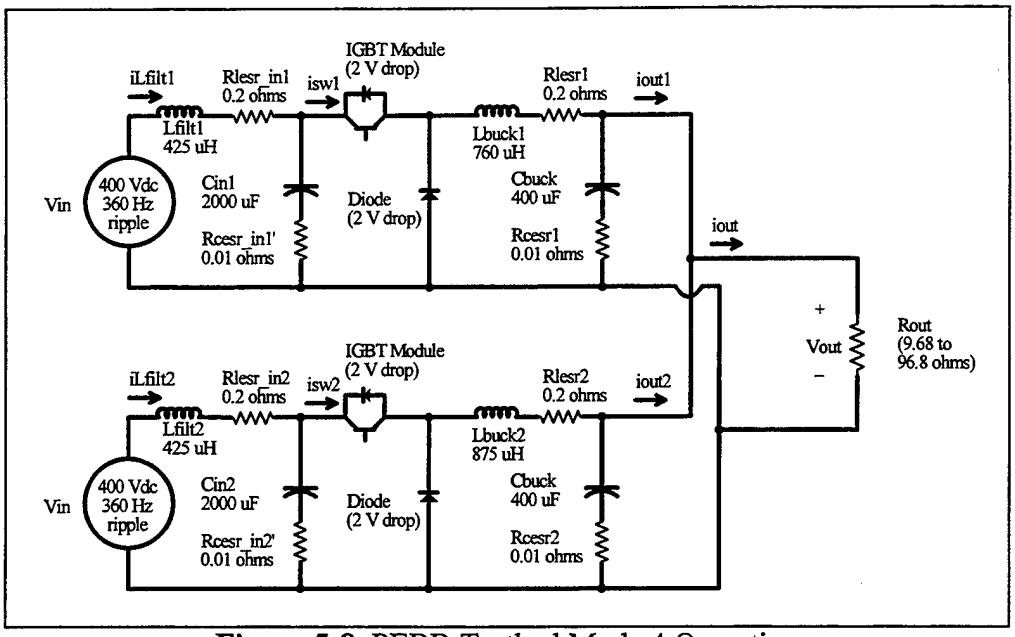


Figure 5-8, PEBB Testbed Mode 4 Operation

its final steady-state value following a slight undershoot. The response for iout2, on the other hand, settles to its final steady-state value after experiencing an overdamped effect. When observed on an expanded scale, the currents are mirror responses of each other. This result occurs due to the difference in the output inductor sizing of the Source Buck Converter units. In essence, a small AC current is established and reflects between the paralleled capacitors based upon the difference in the inductor current waveforms.

With system stability demonstrated and the transient excursion of the variables well within device limitations, a hardware validation study was conducted for Mode 4 operation. Figure (5-10) shows the actual system response for a 2.25 kW to 7.5 kW load change. Although the overall transient was smaller in magnitude, a general comparison of the simulation and hardware trials demonstrates reasonably good correlation between predicted and actual system response. As with the Mode 1 hardware validation, the major

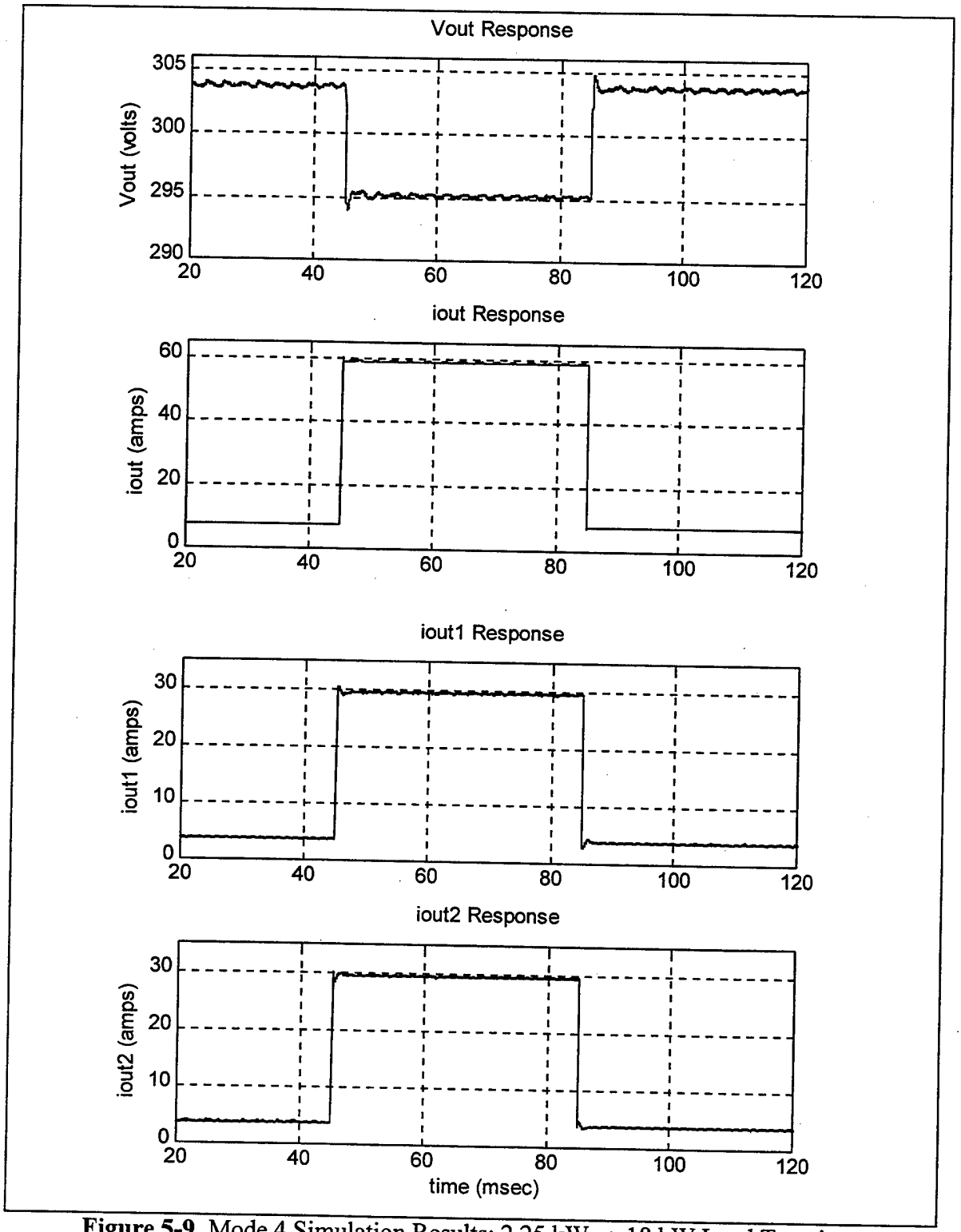


Figure 5-9, Mode 4 Simulation Results: 2.25 kW → 18 kW Load Transient

difference between the simulated and actual results is due to the presence of switching harmonics and EMI noise in actual system operation.

The current "mirror" effect discussed previously is demonstrated on an expanded scale in Figure (5-11) for actual circuit operation. It can be seen that the effect occurs during steady state as well as transient operations. The steady-state effect was not as noticeable in the simulation because switching noise was not modeled. Follow-on research may consider incorporating a Gaussian white noise source into the sensed variables in the simulation.

F. MODE 5 OPERATION

Mode 5 is defined as two parallel Source Buck Converters operating to supply one Load Buck Converter. Figure (5-12) illustrates the overall system model. As described in Mode 1 operation, the input filter capacitors and their associated ESRs had to be modeled as equivalent parallel combinations in the ACSL code.

System transient analysis focused on investigating a step change in resistive load from 300 W to 3 kW on the output of the Load Buck Converter. Figure (5-13) documents the simulation results. As can be seen, the output voltage and current of the parallel Source Buck Converters have a somewhat more pronounced oscillation at the transient edges before settling to their steady-state values. This result is due to a "ringing" effect set up between the output of the Source Buck Converter and the input LC filter of the Load Buck Converter. During load transients, this effect dominates over the current "mirror" effect documented in Mode 4 operation. This result is shown in

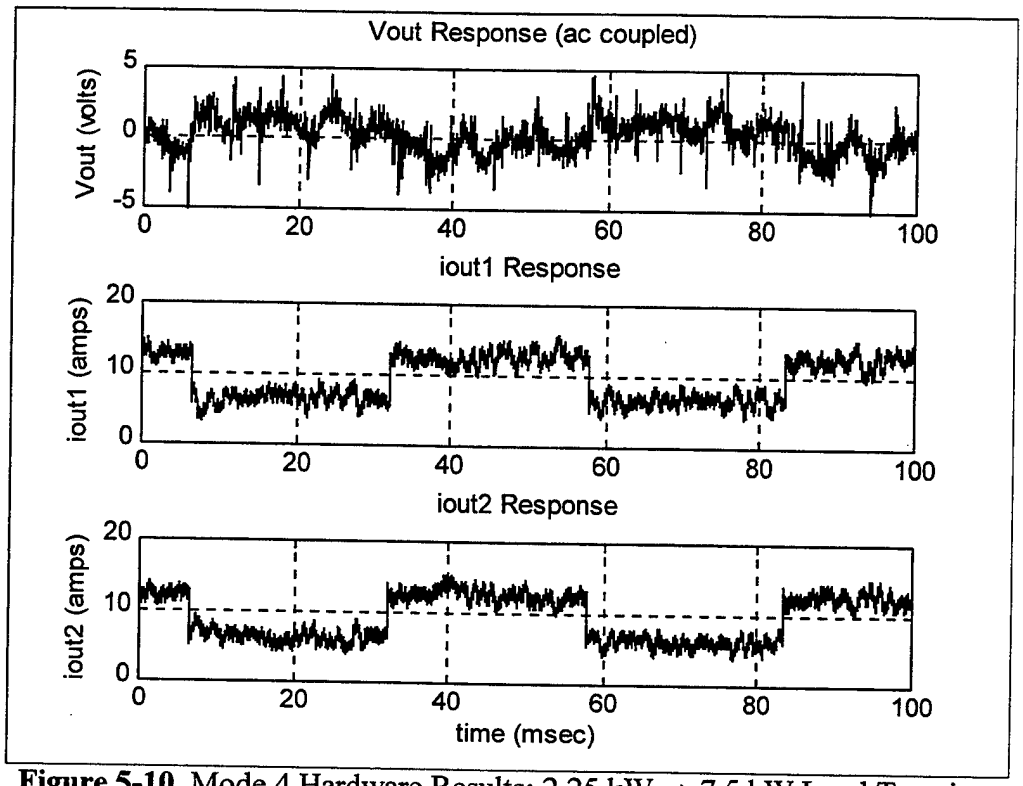


Figure 5-10, Mode 4 Hardware Results: 2.25 kW → 7.5 kW Load Transient

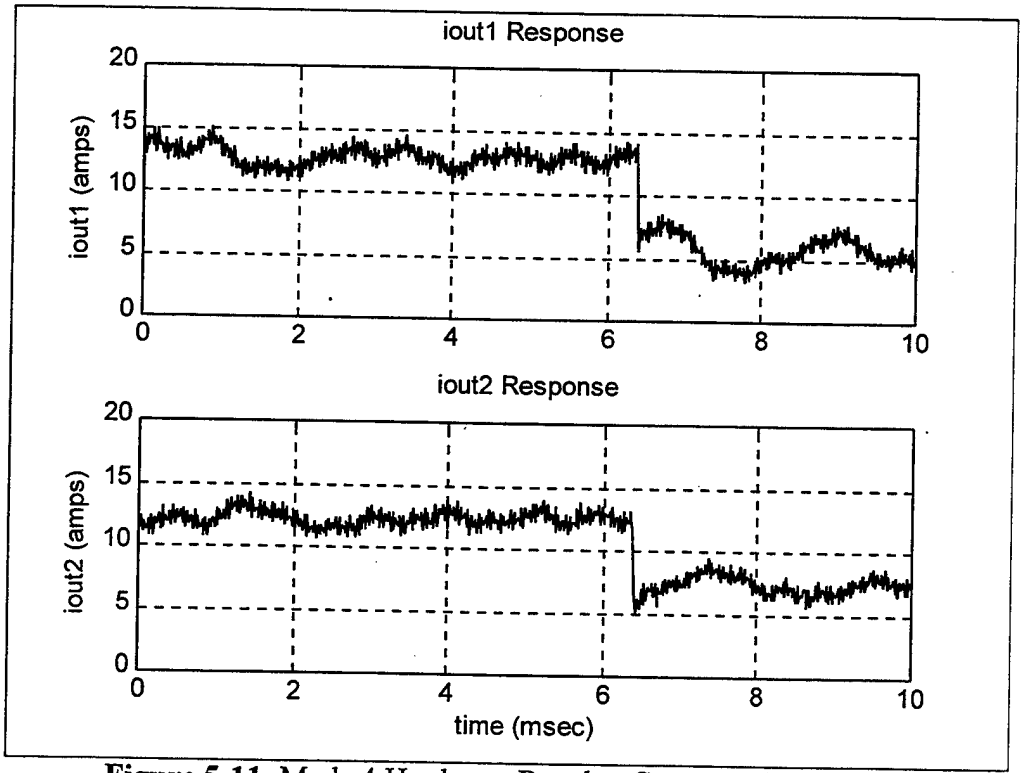


Figure 5-11, Mode 4 Hardware Results: Current Mirror Effect

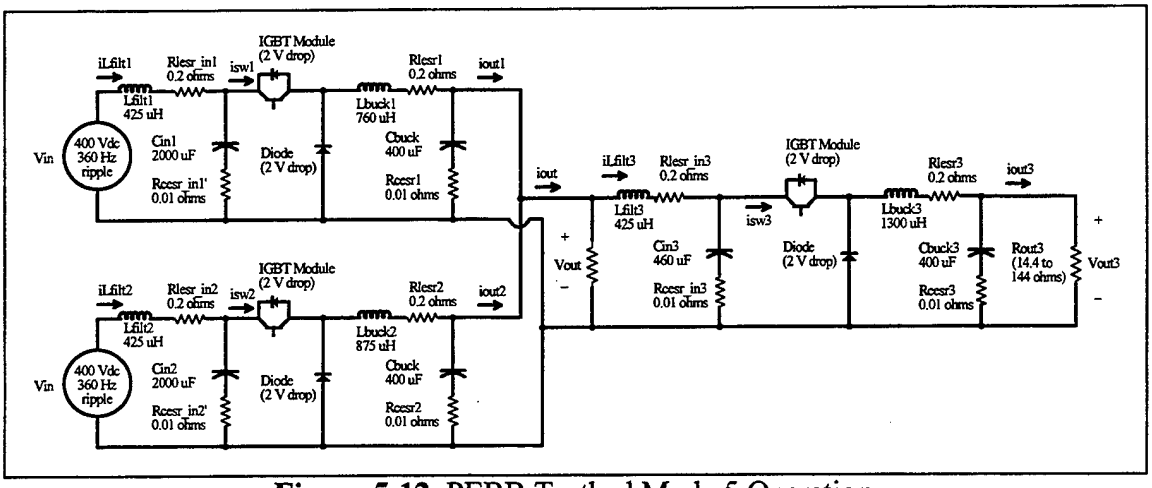


Figure 5-12, PEBB Testbed Mode 5 Operation

Figure (5-13) by observing that the transient portions of the output currents, iout1 and iout2, are "in phase" with one another during load changes.

With system stability demonstrated, a hardware validation study was conducted for Mode 5 operation. Figure (5-14) shows the actual system response. From the plots, it can be seen that the actual Load Buck Converter output correlated with the simulation results. Actual current values out of the parallel Source Buck Converters were about 2.5 A higher than shown in the simulation results because a 25 Ω static load was used during hardware validation. In the ACSL simulation, this value was 40 Ω. The voltage output response of the parallel Source Buck Converter is harder to compare due to the switching noise present on the waveform. Note that all voltage responses are ac coupled.

The "ringing" effect versus the current "mirror" effect is shown in Figure (5-15). This plot shows how the mirror effect dominates during steady-state operation but shifts "in phase" during load changes. Overall, comparison of the simulation and hardware

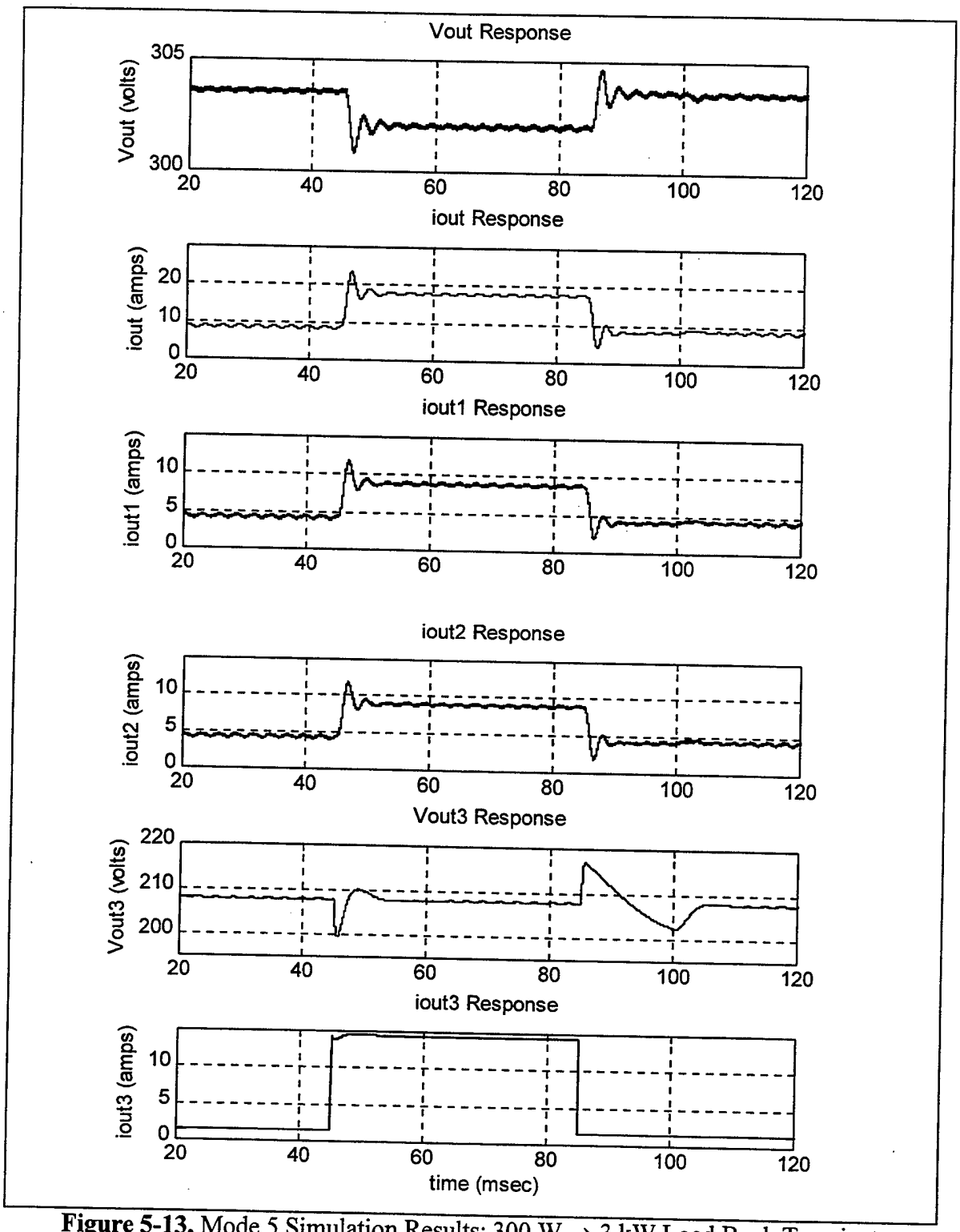


Figure 5-13, Mode 5 Simulation Results: 300 W → 3 kW Load Buck Transient

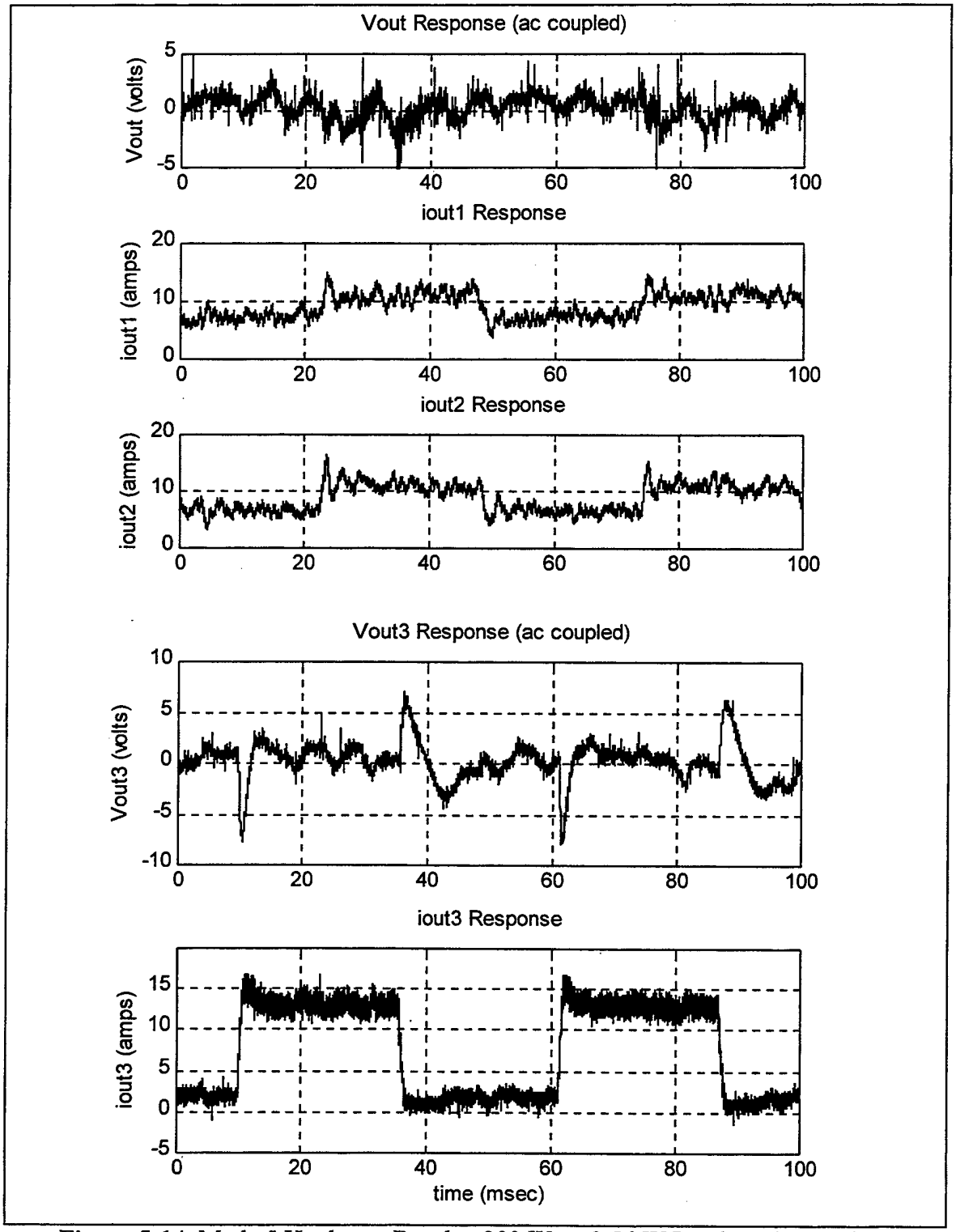


Figure 5-14, Mode 5 Hardware Results: 300 W → 2.5 kW Load Buck Transient

trials demonstrates reasonable correlation between predicted and actual results. As with the previous hardware studies, switching noise constitutes the major difference between simulated and actual operations.

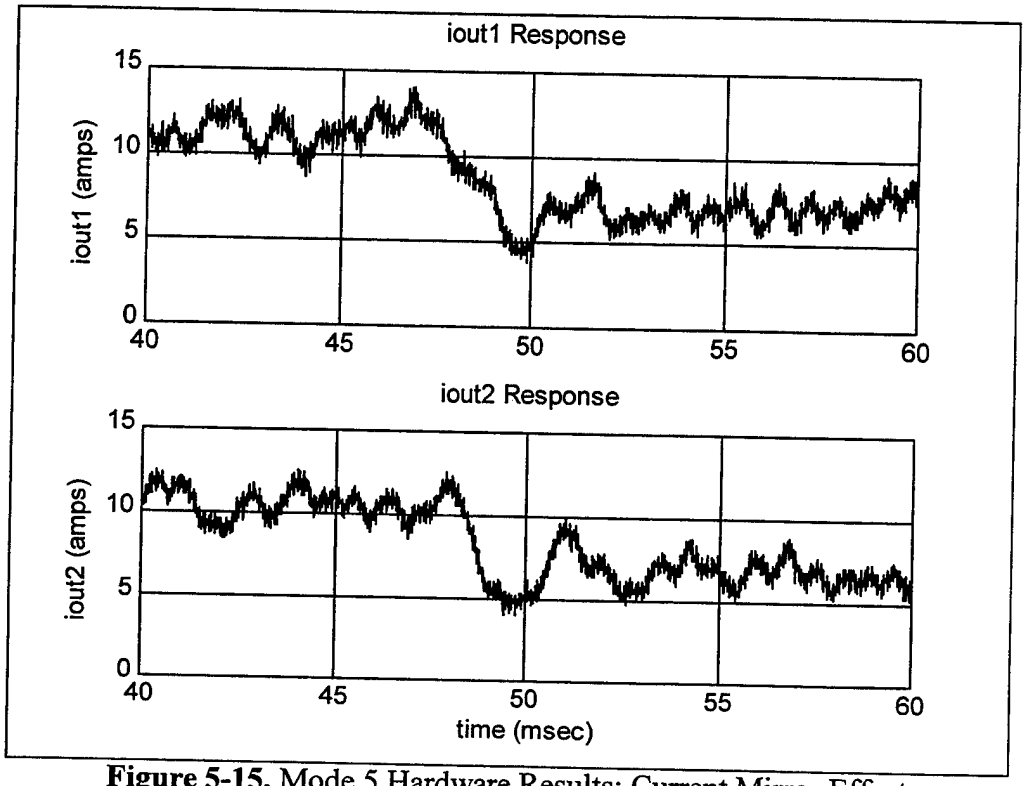


Figure 5-15, Mode 5 Hardware Results: Current Mirror Effect

G. MODE 6 OPERATION

Mode 6 is defined as two parallel Source Buck Converters supplying an inverter. Figure (5-16) illustrates the overall system model. The same inverter model described in Mode 2 operation was used for this case. As described in Mode 1 operation, the input filter capacitors and their associated ESRs had to be modeled as equivalent parallel combinations in the ACSL code.

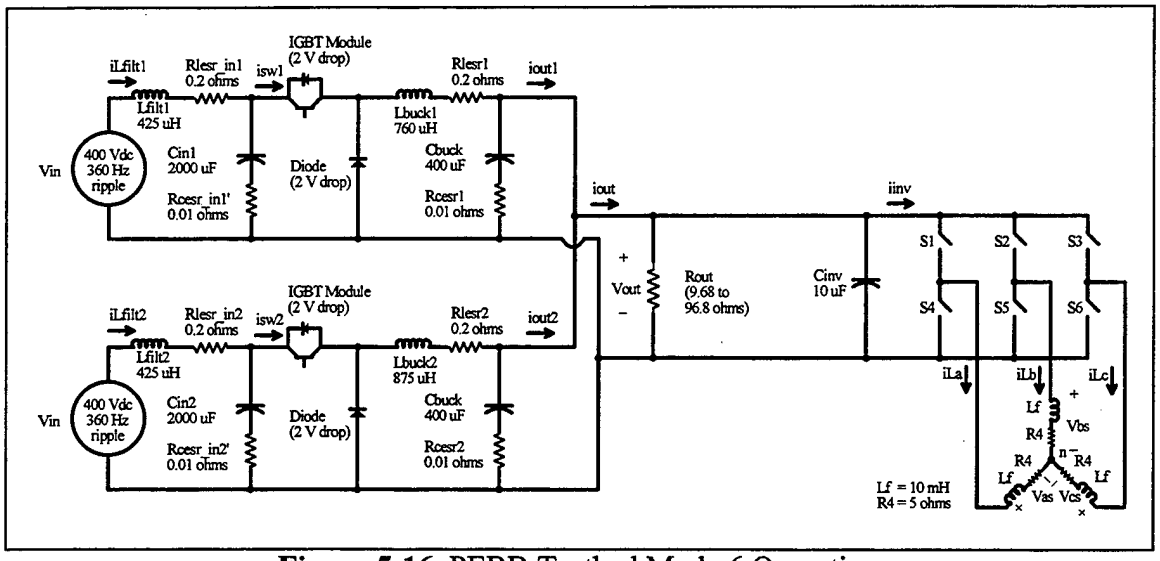


Figure 5-16, PEBB Testbed Mode 6 Operation

System transient analysis focused on investigating a step change in inverter load from 2.5 kW (10 A peak current per phase) to 6 kW (24 A peak current per phase) and observing system response. Figure (5-17) documents the results. As can be seen, significant switching harmonics have been introduced into the output voltage and current responses of the parallel Source Buck Converters. In addition, the level of the harmonics is increased with increased in inverter load. The output of the inverter is as expected for a current controlled unit. As described earlier, the inverter current response does not achieve the desired steady state-values due to the use of stationary reference frame control. A hardware validation study was not conducted for this case because the ARCP closed-loop control has not been fully developed.

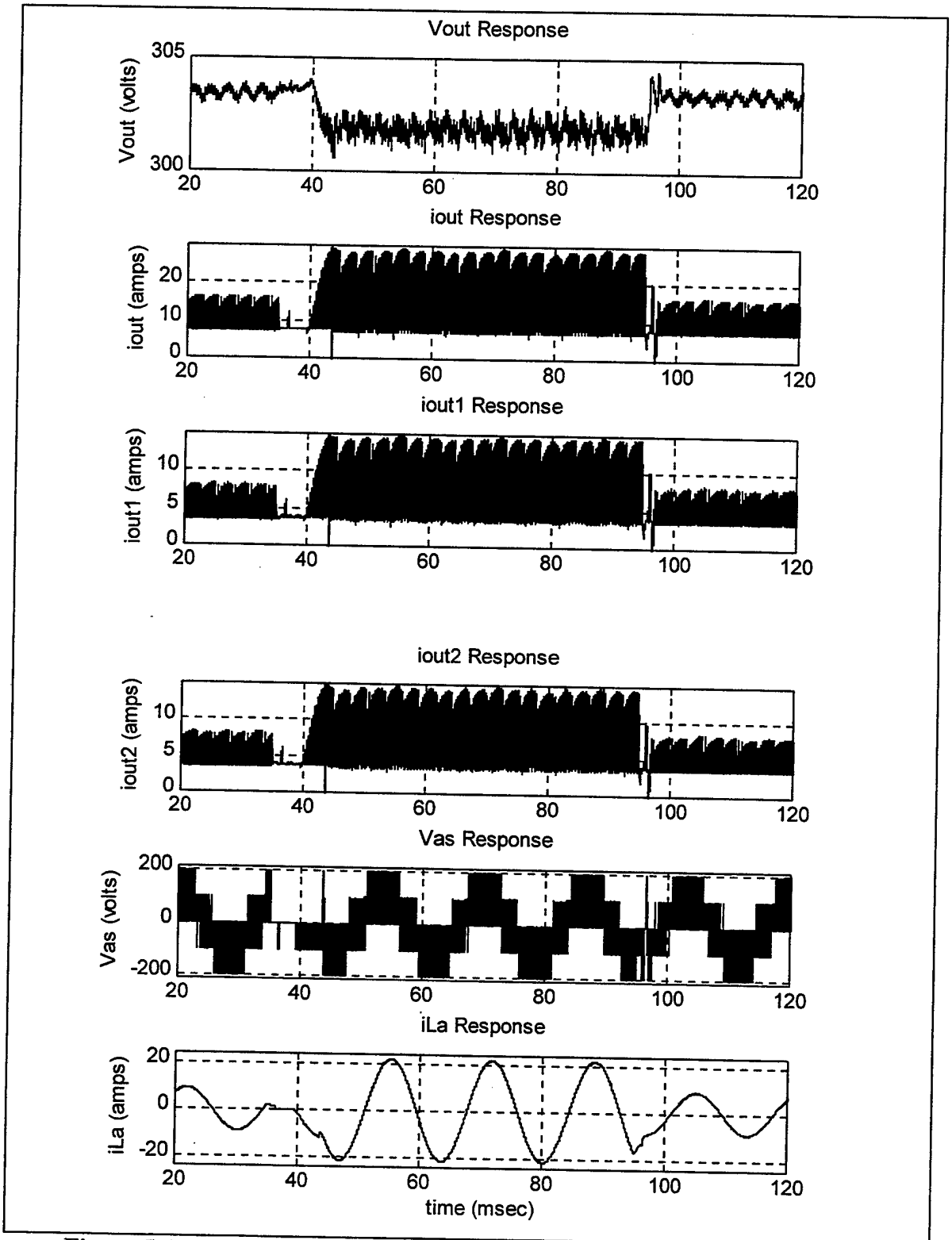


Figure 5-17, Mode 6 Simulation Results: 2.5 kW → 5.3 kW Inverter Transient

H. MODE 7 OPERATION

Mode 7 is defined as two parallel Source Buck Converters supplying both a Load Buck Converter and an inverter. Figure (5-18) illustrates the overall system model. The same inverter model described in Mode 2 operation was used for this case. As described in Mode 1 operation, the input filter capacitors and their associated ESRs had to be modeled as equivalent parallel combinations in the ACSL code.

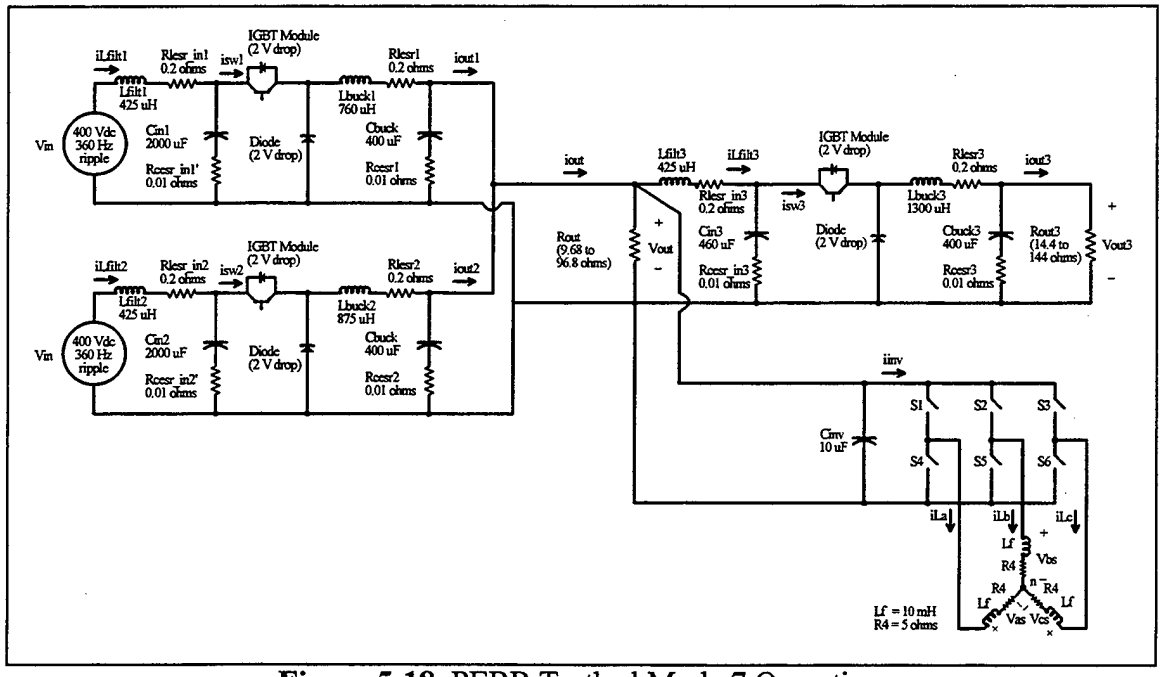


Figure 5-18, PEBB Testbed Mode 7 Operation

System transient analysis focused on investigating the transients discussed in Mode 5 and Mode 6 operation and observing system response. Figures (5-19) and (5-20) document the results. The step change for the Load Buck Converter occurred at 45 msec while the step change in inverter loading was specified at 125 msec. These plots show a system response that is the composite of Mode 5 and Mode 6 results.

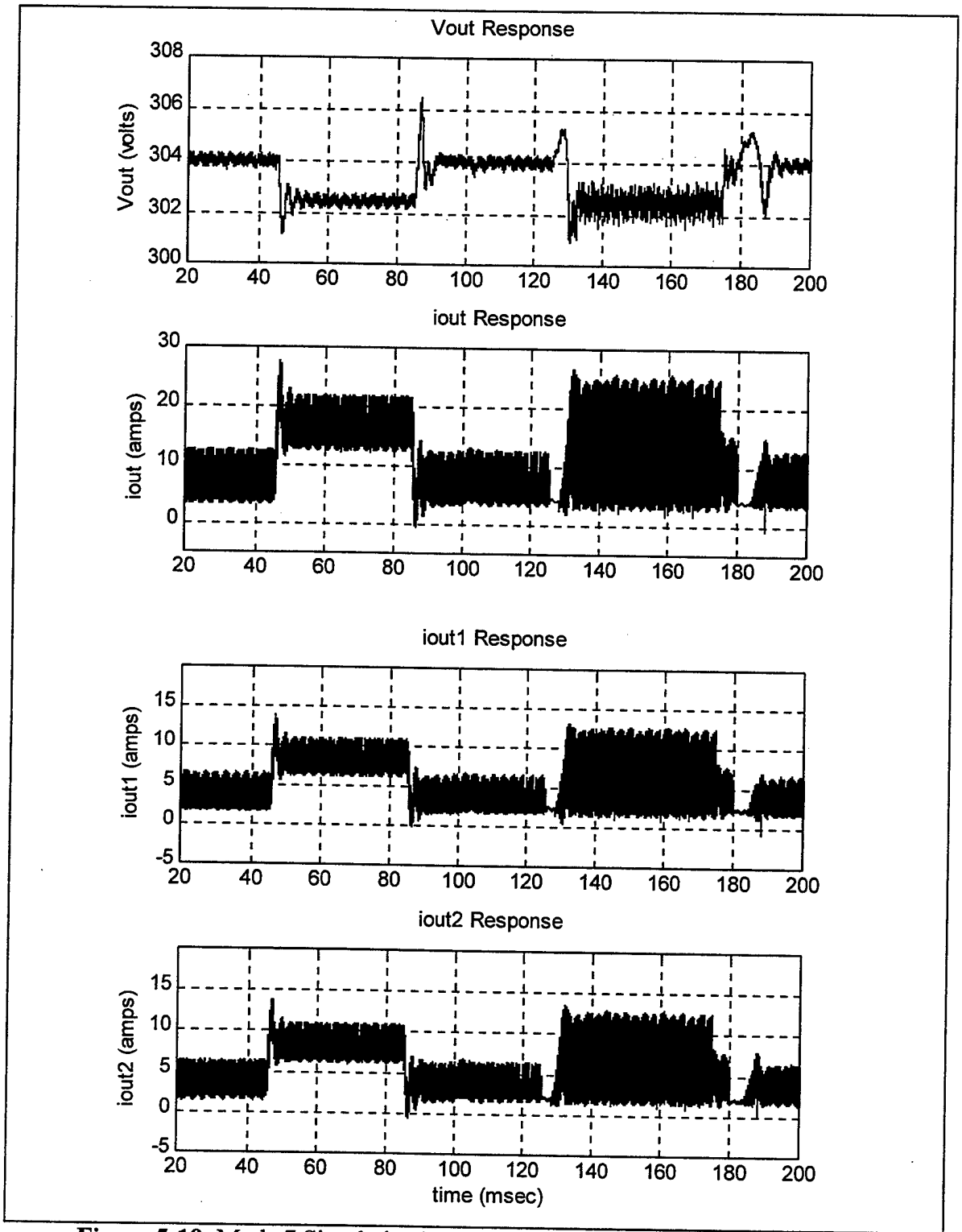


Figure 5-19, Mode 7 Simulation Results: Load Buck and Inverter Transients

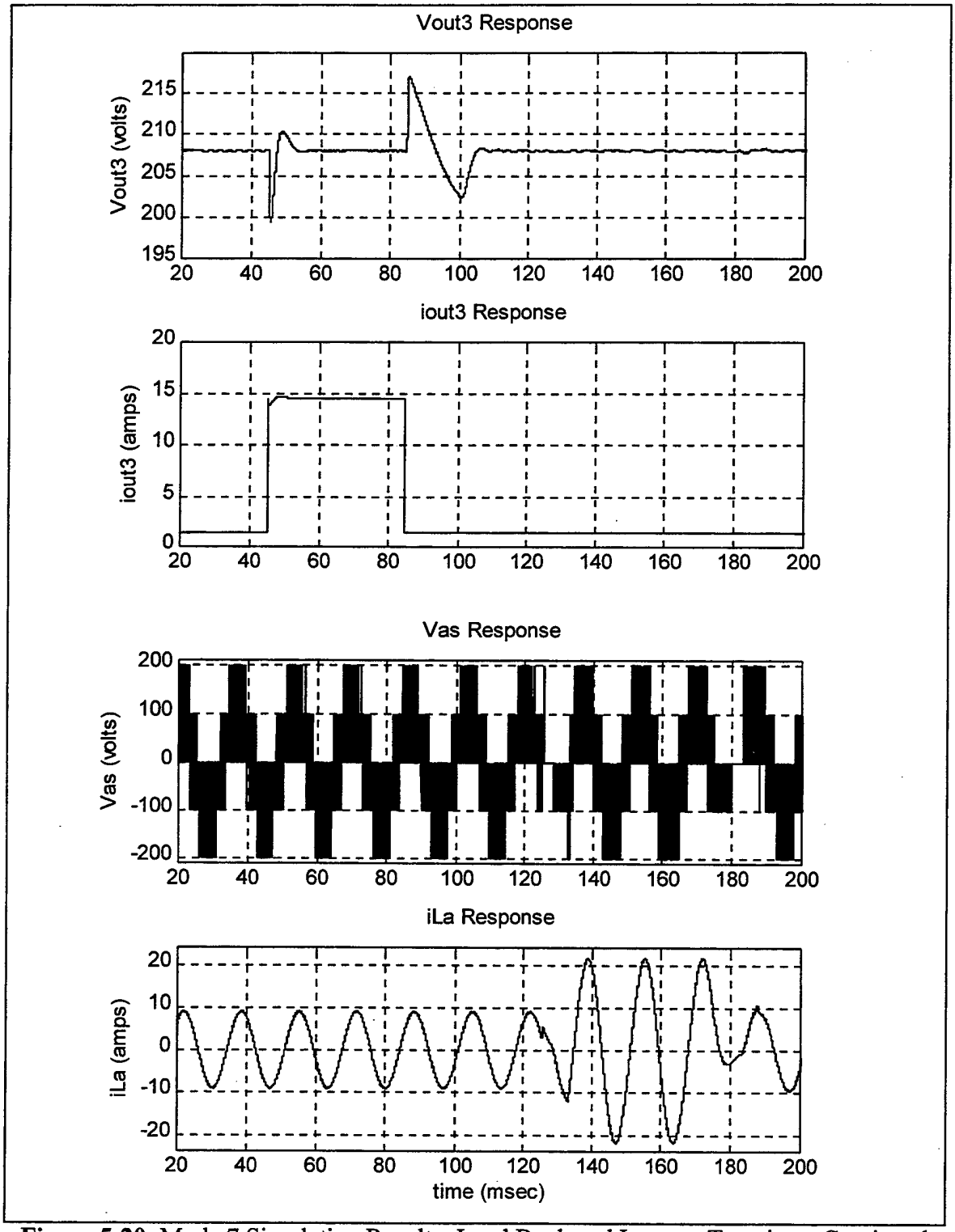


Figure 5-20, Mode 7 Simulation Results: Load Buck and Inverter Transients Continued

I. SUMMARY

This chapter has included a discussion of the ACSL simulations performed for seven basic PEBB Network configurations. The results of these simulations did not yield any gross system instabilities. At most, only "interesting" system interactions were identified. In addition, hardware validation studies were conducted for Modes 1, 4 and 5. The results of these tests verified that the ACSL simulations correlated well with hardware implementation. The hardware validation results also showed that switching noise was a major issue in actual circuit operation. Chapter VII addresses methods for eliminating or minimizing the switching noise and suggests follow-on ACSL simulations and hardware validation studies.

VI. SCHEMATICS

A. CIRCUIT BOARD BACKGROUND

1. Easytrax Overview

The development of printed circuit boards (PCBs) was done using Protel Easytrax, version 2.06. This software allowed PCB definition, layout, and component interconnection on two layers. All PCB layouts were done so that there were no isolated ground sections. Once a PCB was developed, a netlist was compiled. The basic function of the netlist is to associate integrated circuits with resistors, capacitors, and diodes so that connections can be verified. With the PCB layout completed, a Gerber file was made to direct the required circuit milling. The milling work was done at the Naval Postgraduate School by DP1 Daly. Finally, hardcopies of circuit traces and component layouts were made for future reference.

2. Circuit Labeling

For the purpose of more easily locating components and understanding their function in the circuit, the following format was chosen:

The first digit represents the type of component.

- R: resistor
- C: capacitor
- D: diode

The second digit represents the IC with which the component is best associated.

- 0: U10
- 1: U1
- 2: U2, etc.

The third and fourth digits are used for IC pin location.

- 03: pin 3
- 18: pin 18, etc.

When a prefix is used, it indicates an interface between circuits in the overall system. The prefix may have an actual number associated with it or just the '#' sign.

- S1: Sensor board interface for buck converter 1
- D#: Driver board interface for buck converter being referenced
- J25: 25 pin D-sub connector interfacing controller with other boards

If a suffix is used, then there are generally more than one of the same components connected to the same pin.

- A,B,C: Multiple components of same type are connected to same pin where A = 1 component, B = 2 components, etc.
- G: Ground
- H: Vcc, Vdd: positive source voltage
- L: Vss: negative source voltage
- Q: Transistor

Specific examples of circuit labeling include:

- C#-J25-9: Control board interface at J25 pin 9 for buck converter being referenced.

- R113C: Resistor connected to U1 pin 13 and there are at least two more resistors connected to that same node.
- C006: Capacitor connected to U10 pin 6.
- RJ4036Q: Resistor connected to J40 pin 36 and a transistor.

B. CIRCUIT DIAGRAMS AND DESCRIPTIONS

This section documents the hardware implementation of the Source and Load Buck Converters as well as their associated sensor, controller, and driver boards. Each subsection is layed out so that it contains a detailed line diagram and description of the circuit. If further information is needed, Appendix (D) contains all the Easytrax PCB schematics and netlists.

1. Source and Load Buck Converter Topologies

Figures (6-1) and (6-2) document the layout of the Source and Load Buck Converters. As seen from the diagrams, the topologies are identical. Input/output characteristics, inductor sizing, and the voltage sensing circuits represent the only differences. The reason for the different sensing circuits goes back to the choice of analog control boards for the Source Buck Converters and digital control boards for the Load Buck Converters. As the diagrams indicate, the analog controller requires a voltage reduction of (1/50) whereas the digital controller requires a voltage reduction of (1/100). The general flow of overall system operation is as follows:

- Sensed voltage from the voltage divider network is sent to the sensor board via twisted pair.

- Sensed currents are sent to the sensor board via the Hall Effect sensors.
- The sensor board, acting as an isolator, provides the appropriately scaled voltages and currents to the analog/digital control board.
- The control board processes the voltages and currents according to the developed control algorithm.
- The generated driver signal from the control board is sent to the IGBT driver board via twisted pair (analog control) or optical signal (digital control).

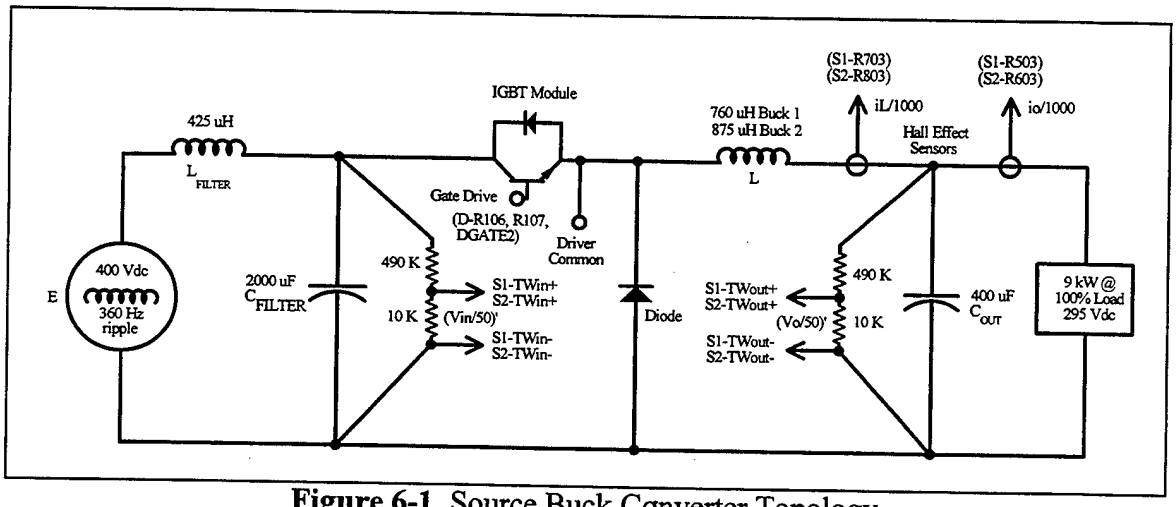


Figure 6-1, Source Buck Converter Topology

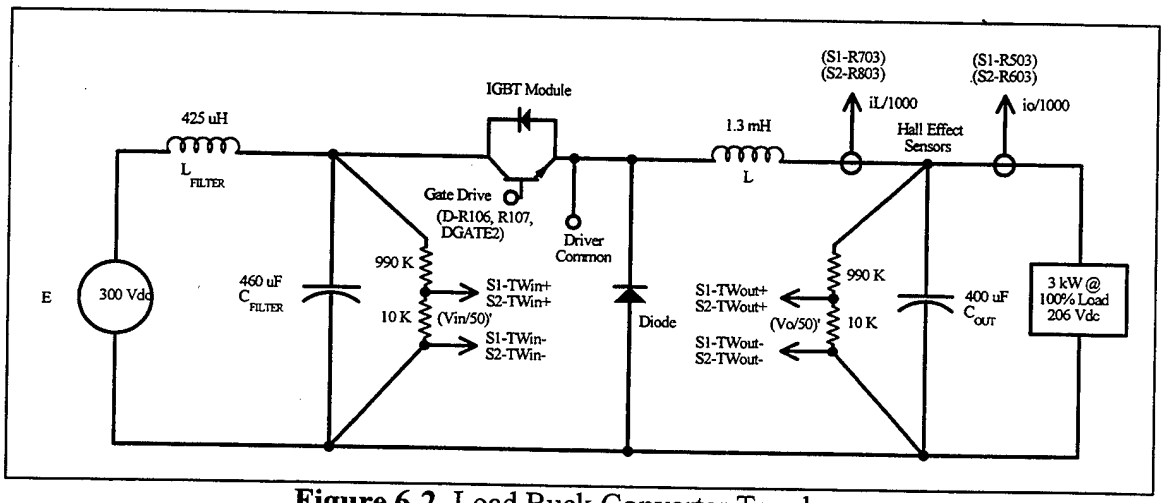


Figure 6-2, Load Buck Converter Topology

- The driver signal controls the IGBT switching frequency and duty cycle.

With this overview in hand, a more detailed description is presented next.

2. Sensor Board

Pictured in Figure (6-3), the sensor board provides isolation between the buck converter and the analog/digital control board. Sensed voltages from the voltage divider network on the buck converter are sent via twisted pair to the input of wideband AD215 isolation units. Configured as a unity gain buffers, the AD215s output the sensed voltages and send them to the control board via twisted pair. Current sensing is done by CL50 Hall Effect sensors mounted directly on the sensor board. Instantaneous currents proportional to (i_{sensed}/1000) are output by the Hall Effects. These currents are converted to a proportional voltage by 100 Ω precision resistors to yield (i_{sensed}/10). These voltages are the sent to the control board via twisted pair.

3. Analog Control Board

The implementation of the analog control board can be broken down into five subsections:

- J25 - 25 Pin Connector and Buffer Stage
- Main Control Stage
- Pulse Width Modulation Stage
- Protection and Startup Circuitry Stage
- Power Supply Stage

A diagram and discussion of each of the subsections is provided below. Appendix (D) contains detailed PCB information on the analog control board.

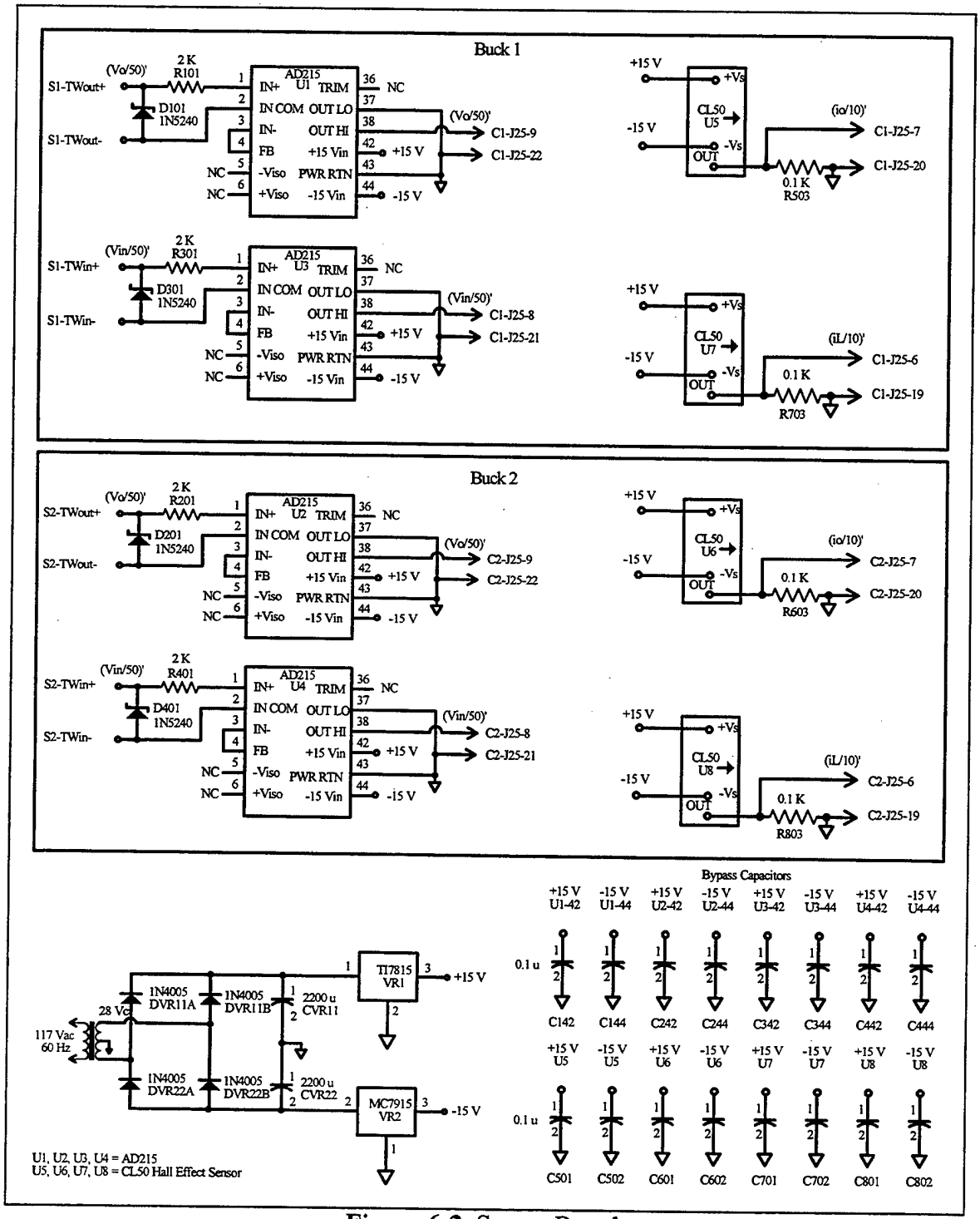


Figure 6-3, Sensor Board

a. Buffer Stage

From the twisted pair output of the sensor board, the sensed and scaled voltages and currents go to the 25 pin D-sub connector of the control board as shown in Figure (6-4). From J25, the inputs are buffered utilizing a LM324 quad op amp (labeled U2). Each buffer stage input contains an RC lowpass filter designed to remove high-frequency components from the measured voltages and currents. The cutoff frequency for each filter is at 8 kHz. The LM324 outputs are fed to the Main Control Stage.

b. Main Control Stage

The Main Control Stage is where Equation (2-18) is formulated. The output of this stage is the analog duty cycle scaled by a factor of ten. In addition to the Buffer Stage outputs, the signal representing the desired output voltage is fed into the Main Control from the Protection and Startup Circuitry Stage. As shown in Figure (6-5), an AD534 configured as a divider provides the base duty cycle, D_{ss}, by dividing the reference voltage by the input voltage. Note that this arrangement implements the feedforward term given in Equations (3-18) and (3-19) of Chapter III.

The small-signal control algorithm is performed using a LF347 quad op amp. The first three op amps in the package form the proportional voltage ($U1_1$), integral voltage ($U1_2$)and current response ($U1_3$) terms. Note that a limited integrator has been used to generate the integral voltage term. This implementation prevents integrator windup. Without the limiting zener diodes in place, C107 is free to charge up during extended transients to values well outside the range in which its voltage affects duty

cycle. As a result, the voltage on C107 locks out the remaining terms in the duty cycle control algorithm. This lockout persists until C107 come backs within its operating range. The final op amp ($U1_4$) is used to scale and sum the outputs of the first three op amps. Scaling is performed such that the analog gain values developed in Chapter IV were obtained, scaled by a factor of ten. Output feeds the Pulse Width Modulation Stage.

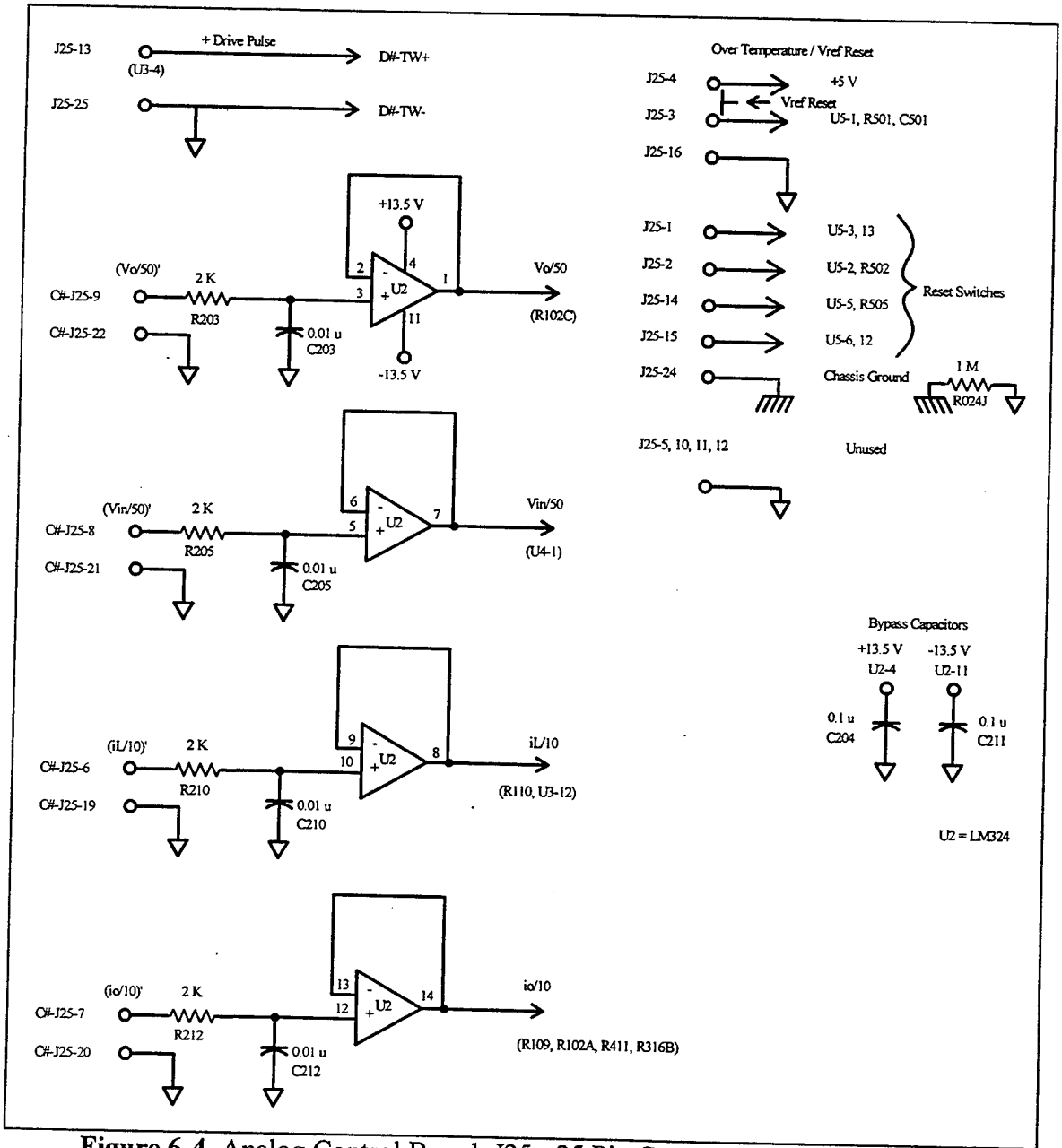


Figure 6-4, Analog Control Board: J25 - 25 Pin Connector and Buffer Stage

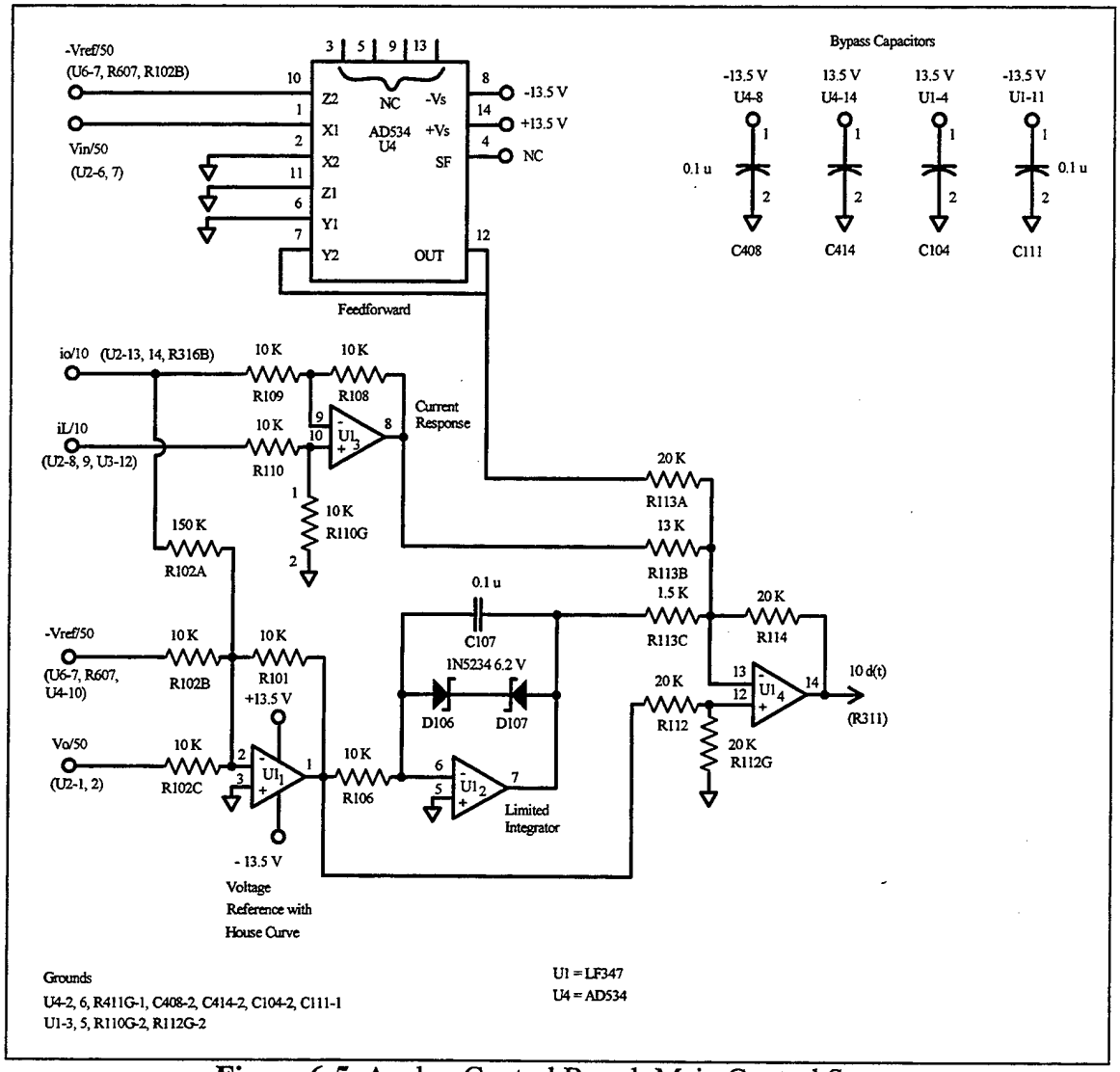


Figure 6-5, Analog Control Board: Main Control Stage

c. Pulse Width Modulation Stage

Pictured below in Figure (6-6), the Pulse Width Modulation (PWM) Stage is comprised of the UC3637 chip. It receives the analog duty cycle from the Main Control Stage and produces a square wave driver signal with a frequency of 23 kHz based upon R318 and C302. This frequency can be calculated utilizing formulas given in the UC3637 data sheet in Appendix (A). Actual frequency runs about 20.5 kHz due to

interactions with the phase-locked loop circuit described below. The duty cycle of the pulses is equal to the Main Control Stage analog duty cycle signal divided by ten. From the Pulse Width Modulation Stage, the driver signal is fed to a driver board mounted on the IGBT.

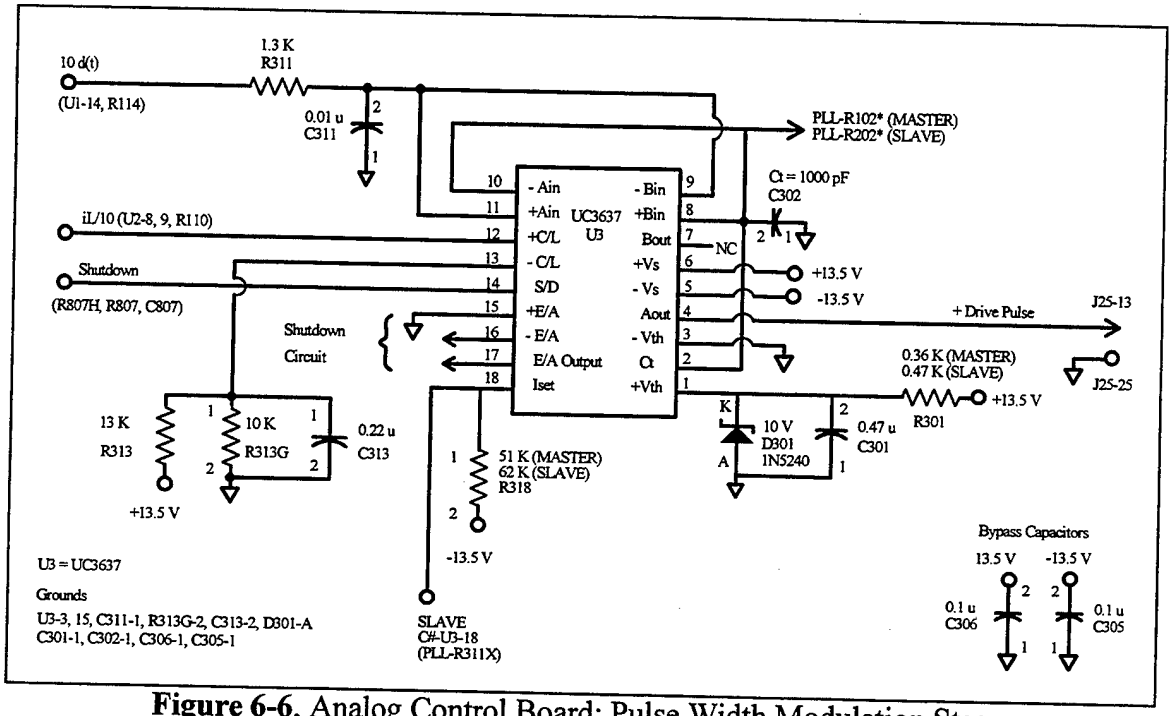


Figure 6-6, Analog Control Board: Pulse Width Modulation Stage

d. Phase Locked Loop

Analysis of Figure (6-6) shows MASTER and SLAVE connection points for implementation of the Phase-Locked Loop Board shown in Figure (6-7). This board synchronizes the switching frequencies of both Source Buck Converter controllers so that they can operate in parallel. Without proper synchronization, load sharing between paralleled units cannot be achieved due to inductor current oscillations. As shown in

reference [7], digital control precludes the need for this circuit since frequency can be directly set by timer circuits.

Implementation of the phase-locked loop requires one controller to be designated the MASTER so that it can set the switching frequency. The other board, by default, is the SLAVE. In actual operation, the phase-locked loop generates a DC current depending on the difference in frequencies of the two controllers. This current sums with the set current of the SLAVE PWM chip, directly influencing the switching frequency of the SLAVE controller. This cycle continues until frequency capture and lock results. In addition to the functions and interfaces mentioned above, the PWM Stage also

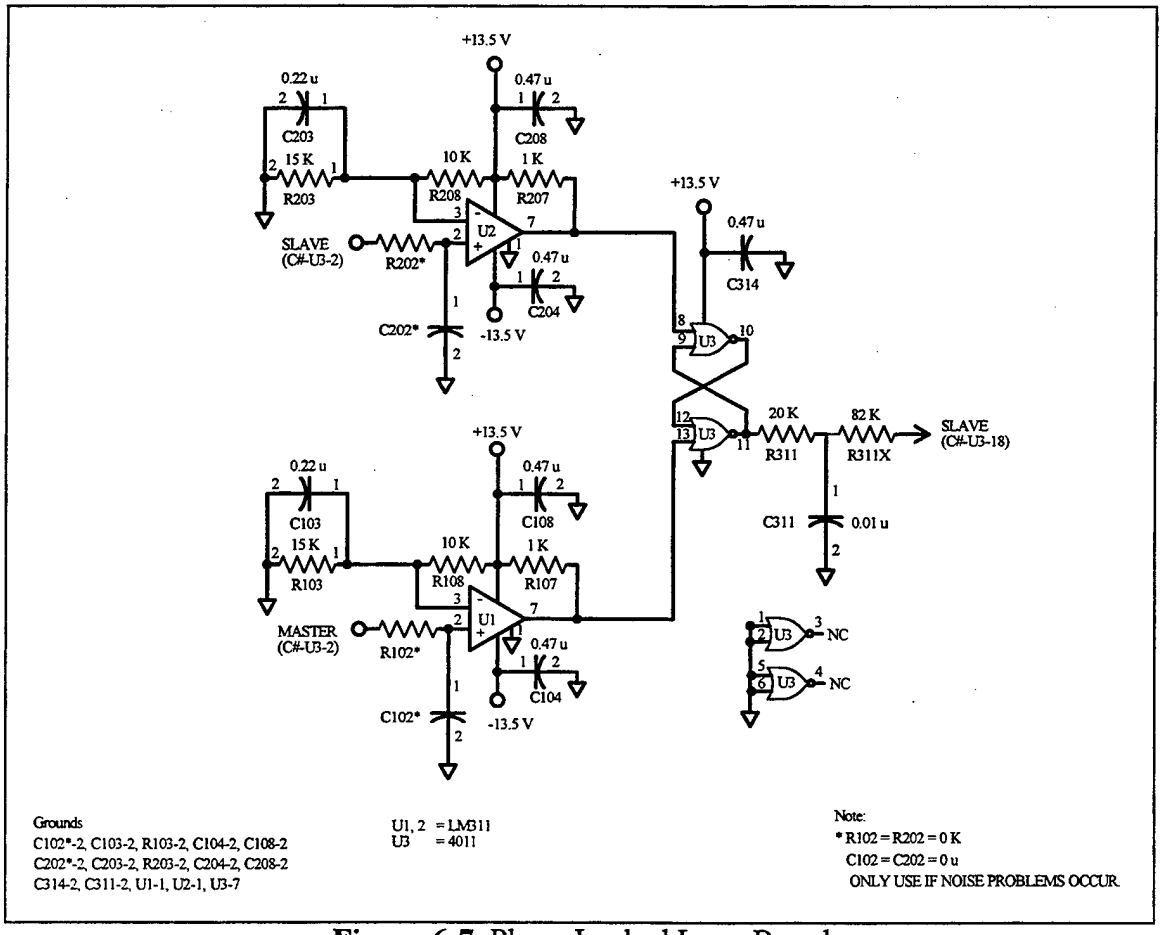


Figure 6-7, Phase-Locked Loop Board

provides pulse-by-pulse and overcurrent protection circuitry. These circuits are documented in the following Protection and Startup Circuitry Stage discussion.

e. Protection and Startup Circuitry Stage

The analog controller provides the Source Buck Converters with two forms of protection. The first is the pulse-by-pulse current limiting feature of the UC3637 chip. This circuit protects the IGBT from exceeding its 90A current rating. Referring to Figure (6-6), pins 12 and 13 make up this circuit. If a fault occurs in the buck converter, $i_L/10$ will rise. If its peak value gets to ~6.75V, the driver output of U3 goes low, and the IGBT remains open until peak inductor current goes below the 6.75V threshold.

The second protection circuit is overcurrent time-out shown in Figure (6-8). This circuit protects components from thermal damage when $i_o/10$ exceeds 150% of its rated value for more than 1 second. Utilizing the op amp in U3 (pins 15-17), an integrator was designed such that $i_o/10 = 4.5V$ causes pin 17 to reach -10V in 1 second. This voltage trips comparator U7 high and causes $U5_2$ and $U5_4$ to go high (~ +V). As a result, comparator U8 goes high initiating a shutdown signal to pin 14 of U3. This protective action has the end effect of halting Source Buck Converter operation.

The startup circuitry allows duty cycle to ramp up to its steady-state value from zero initial conditions. When the controller is initially energized, $U5_1$ goes high due to a +5V pulse generated by the RC circuit at input pin 1 of the OR gate. As a result, $U5_4$ and U_8 go high and Q1 turns "on". This action causes pin 3 of U6 to go low (~ -V)

and prevents a reference signal from being generated. The end result is a voltage lockout. This action prevents a duty cycle waveform from being generated instantaneously. If such an occurrence happened from zero initial conditions, large current and voltage

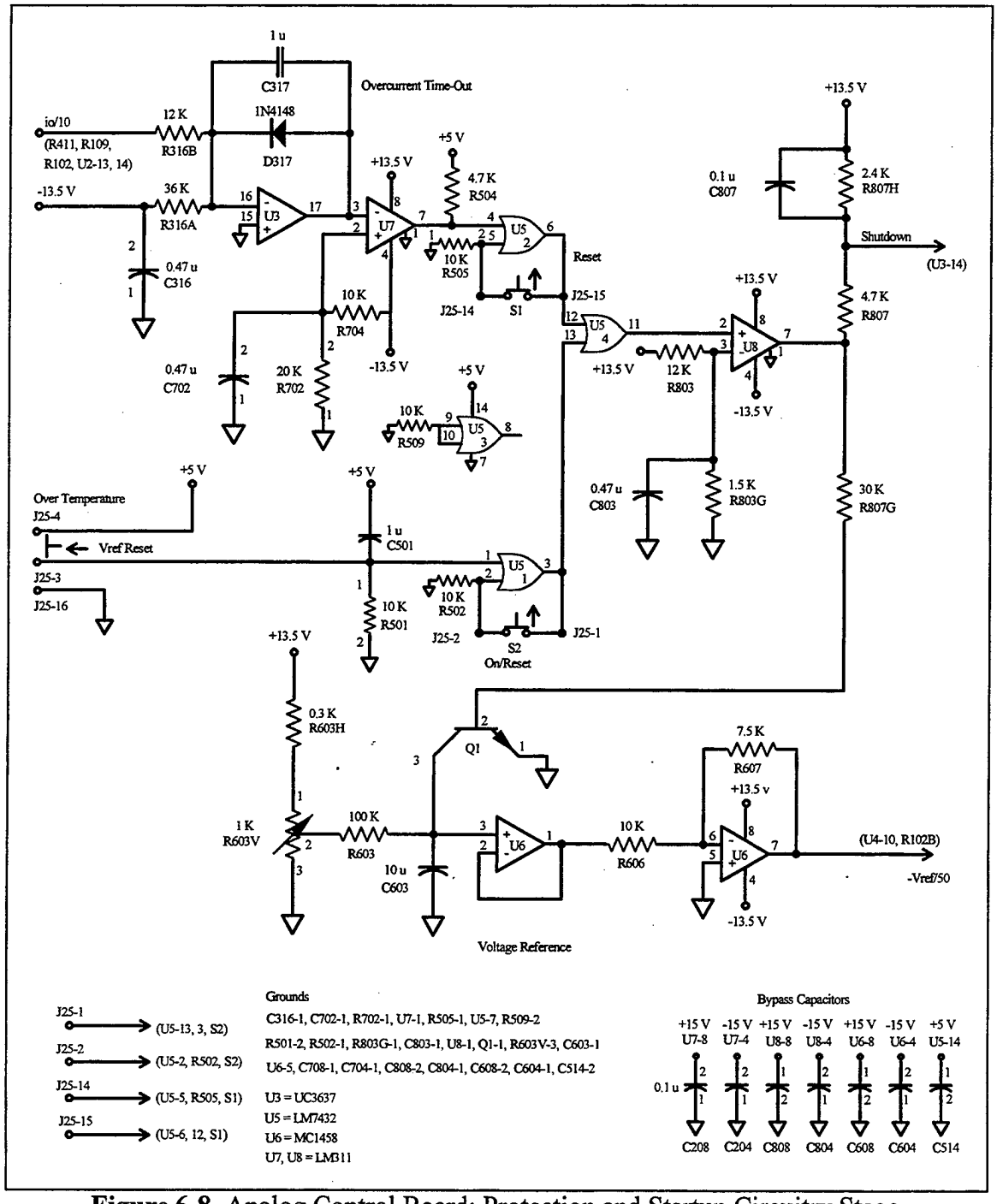


Figure 6-8, Analog Control Board: Protection and Startup Circuitry Stage

oscillations would result with possible damage to the buck converter. As it stands, however, d(t)=0. By resetting momentary switch S2, Q1 is turned "off" and voltage is applied to pin 3 of U6. Reference voltage ramps up to its final setpoint through the action of the RC time constant established by R603 and C603 (τ = 1 sec). Raising the reference voltage in this way causes duty cycle to ramp to its operating value thus preventing the aforementioned current and voltage oscillations on startup. After startup, the reference voltage continues to be supplied through U6.

f. Power Supply Stage

Shown in Figure (6-9), the power supply takes 117 VAC, 60 Hz receptacle power, and rectifies it into 11.5 VDC for input to the DATEL TWR-5/1000-15/200-D12

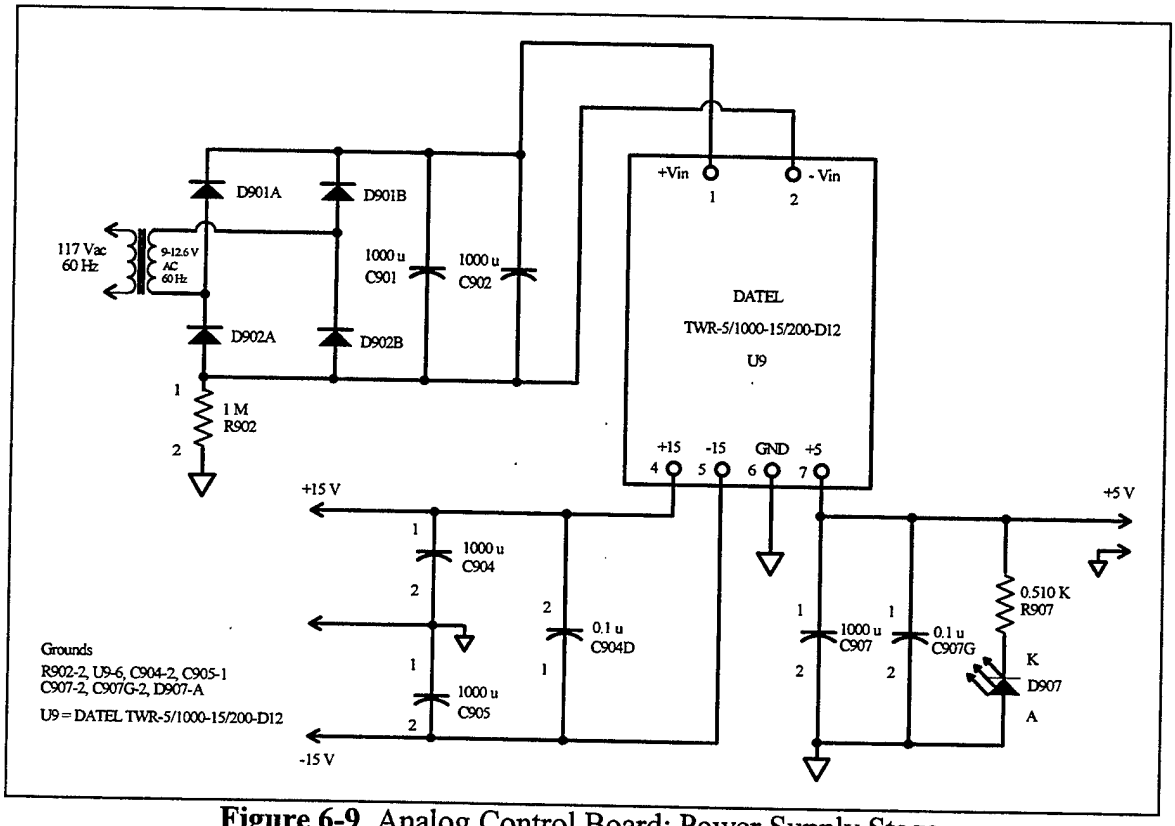


Figure 6-9, Analog Control Board: Power Supply Stage

voltage regulator. The output of the DATEL module is +5 V_{DC} and $\pm$13.5 V_{DC} supply power for the control board. The $\pm$15 V_{DC} is not obtained due to minimal circuit loading. The lower $\pm$13.5 V_{DC} values meet control board component power supply requirements.

4. IGBT Driver Board

The IGBT driver board completes the feedback loop. Input to the board can be made using twisted pair (analog control) or optical interface (digital control). Both configurations are shown in Figure (6-10). The input driver signal is amplified to $\pm$ 10 V by the optically isolated push-pull transistor pair in U1. Following amplification, the driver signal is applied to the IGBT gate. The back-to-back 1N4744 zener diodes (V_z = 15 V) were installed across the gate to prevent overvoltage damage.

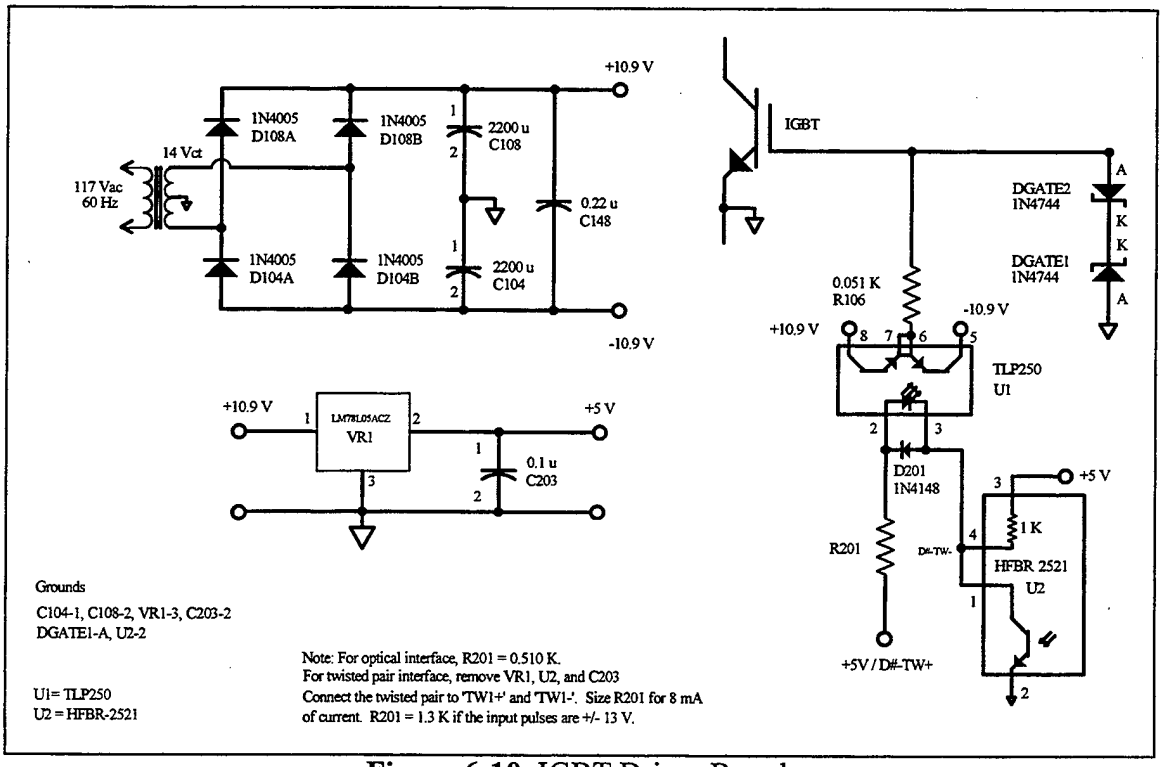


Figure 6-10, IGBT Driver Board

5. Digital Control Board

Reference [7] contains a detailed description on the layout and operation of the Load Buck Converter digital control board.

VII. CONCLUSIONS

This research documented the design of SSCM units and their controllers for the PEBB Network Simulation Testbed being built at the Naval Postgraduate School. In addition, baseline simulations were developed to identify system instabilities and interesting controller interactions. Finally, hardware validation studies were conducted to confirm the PEBB Network simulations.

The SSCM design process began with the selection of components for the buck converter power sections. The initial selections were based upon system specifications, theoretical calculations, and physical limitations. The process continued with the development of state-difference analog and digital control algorithms. Next, gain selection was performed for the Source Buck Converter analog controller and the Load Buck Converter digital controller. Chapter IV showed that to achieve the desired response, proper output capacitor sizing and current gain selection had to be achieved to minimize unwanted harmonic effects in the system output.

With the proper gain values determined, detailed ACSL simulations were performed as described in Chapter V. The results showed that the PEBB Network is stable under the seven basic operational modes. In addition, hardware validation studies were conducted for Modes 1, 4 and 5. The results of this testing verified that the ACSL simulations correlated well with hardware implementation. From this process, it was shown that the SSCM units operated to their design specifications.

One issue not considered in the initial design process, however, dealt with shielding. As shown in Chapter V, switching noise had a considerable effect on the system. This problem will only grow worse with the addition of more SSCM and SSIM units. As a result, the next generation of SSCMs need to employ methods to eliminate switching noise. Several recommendations include:

- soft switching
- placement of an enclosure about the buck converter power sections
- relocation of the buck converter controllers to a separate enclosure
- use of fiberoptic connections between controllers and feedback paths.

An additional system improvement deals with the replacement of the Source Buck analog controllers with digital controllers. This step will greatly enhance the flexibility of the PEBB Network since frequency and gain modifications can be performed in software for the digital controller. As a result, a wide variety of system interactions can be investigated "on the fly" rather than having to wait for resistor replacements on the analog control boards. In addition, the digital control provides another layer of noise immunity.

Recommendations for future research focus on further development of the ACSL simulation models. Areas for investigation include:

- effect of buck converters operating at different frequencies
- implementation of an ARCP synchronous reference frame current controller
- incorporation of Gaussian white noise into the sensed variables to model the effects of switching noise.

Ideally, these detailed simulations can then be used to direct hardware testing towards interesting PEBB Network interactions and stability bounds.

APPENDIX A. DATA SHEETS

A. INTERNATIONAL RECTIFIER IRGT1090U06 IGBT

PD-9.954B

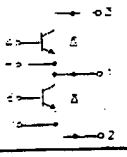
International IOR Rectifier

IRGTI090U06

"HALF-BRIDGE" IGBT INT-A-PAK

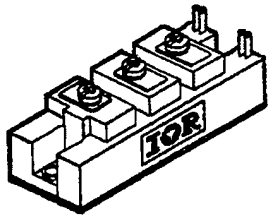
Ultra-fast™ Speed IGBT

Half-Bridge



V_{CE} = 600V
I_C = 90A
V_{CE}(ON) < 3.0V

- Rugged Design
- Simple gate-drive
- Ultra-fast operation up to 25KHz hard switching, or 100KHz resonant
- Switching-Loss Rating includes all "tail" losses



INT-A-PAK case

Description

IR's advanced IGBT technology is the key to this line of INT-A-pak Power Modules. The efficient geometry and unique processing of the IGBT allow higher current densities than comparable bipolar power module transistors, while at the same time requiring the simpler gate-drive of the familiar power MOSFET. This superior technology has now been coupled to state of the art assembly techniques to produce a higher current module that is highly suited to power applications such as motor drives, uninterruptible power supplies, welding, induction heating and ultrasonics.

Absolute Maximum Ratings

Parameter	Description	Value	Units
V_{CES}	Continuous collector to emitter voltage	600	V
I_C @ T_C = 25°C	Continuous collector current	90	A
I_C @ T_C = 85°C	Continuous collector current	60	
I_C @ T_C = 100°C	Continuous collector current	50	
I_{LM}	Peak switching current	180	
I_{FM}	Peak diode forward current (1) 225		
V_{GE}	Gate to emitter voltage	± 20	V
V_{ISOL}	RMS isolation voltage, any terminal to case, t = 1 min	2500	
P_D @ T_C = 25°C	Power dissipation	298	W
T_J	Operating junction temperature range	-40 to 150	°C
T_{STG}	Storage temperature range	-40 to 125	

(1) Duration limited by max junction temperature.

Electrical Characteristics - T_J = 25°C, unless otherwise stated

Parameter	Description	Min	Typ	Max	Units	Test Conditions
BV_{CES}	Collector-to-emitter breakdown voltage	600	—	—	V	V_{GE} = 0V, I_C = 1mA
$V_{CE}(ON)$	Collector-to-emitter voltage	—	—	3.0		V_{GE} = 15V, I_C = 90A
		—	3.1	—		V_{GE} = 15V, I_C = 90A, T_J = 150°C
V_{FM}	Diode forward voltage - maximum	—	—	2.8		I_F = 90A, V_{GE} = 0V
		—	2.6	—		I_F = 90A, V_{GE} = 0V, T_J = 150°C
V_{GEth}	Gate threshold voltage	3.0	—	5.5		I_C = 500µA
ΔV_{GEth}	Threshold voltage temperature coeff.	—	-11	—	mV/°C	V_{CE} = V_{GE}, I_C = 500µA
g_{fe}	Forward transconductance	34	—	58	S(℧)	V_{CE} = 25V, I_C = 90A
I_{CES}	Collector-to-emitter leakage current	—	—	1	mA	V_{GE} = 0V, V_{CE} = 600V
		—	—	10		V_{GE} = 0V, V_{CE} = 600V, T_J = 150°C
I_{GES}	Gate-to-emitter leakage current	—	—	±1	µA	V_{GE} = ±20V

Dynamic Characteristics - T_J = 150°C

Parameter	Description	Min	Typ	Max	Units	Test Conditions
E_{on}	Turn-on switching energy	—	0.05	—	mJ/A	R_{G1} = 47Ω, R_{G2} = 0Ω
E_{off} (1)	Turn-off switching energy	—	0.05	—		I_C = 90A, L_S = 100nH
E_{ts} (1)	Total switching energy	—	—	0.12		V_{CC} = 360V, V_{GE} = ± 15V
$t_{d(on)}$	Turn-on delay time	—	70	—	ns	R_{G1} = 47Ω, R_{G2} = 0Ω
t_r	Rise time	—	90	—		I_C = 90A
$t_{d(off)}$	Turn-off delay time	—	180	—		V_{CC} = 360V, V_{GE} = ± 15V
t_f	Fall time	—	250	—		L_S = 100nH
I_{rr}	Diode peak recovery current	—	52	—	A	R_{G1} = 47Ω, R_{G2} = 0Ω
t_{rr}	Diode recovery time	—	110	—	ns	I_C = 90A
Q_{rr}	Diode recovery charge	—	3.0	—	µC	V_{CC} = 360V, V_{GE} = ± 15V
Q_{ge}	Gate-to-emitter charge (turn-on)	150	—	280	nC	V_{CC} = 360V
Q_{gc}	Gate-to-collector charge (turn-on)	70	—	140		I_C = 90A
Q_g	Total gate charge (turn-on)	26	—	42		V_{GE} = 15V
C_{ies}	Input capacitance	—	5800	—	pF	V_{GE} = 0V
C_{oes}	Output capacitance	—	660	—		V_{CC} = 30V
C_{res}	Reverse transfer capacitance	—	80	—		f = 1MHz

(1) Includes tail losses

Thermal and Mechanical Characteristics

Parameter	Description	Typ	Max	Units
R_{thJC} (IGBT)	Thermal resistance, junction to case, each IGBT	—	0.42	°C/W
R_{thJC} (Diode)	Thermal resistance, junction to case, each diode	—	0.7	
R_{thCS} (Module)	Thermal resistance, case to sink	0.1	—	
Wt	Weight of module	140	—	g

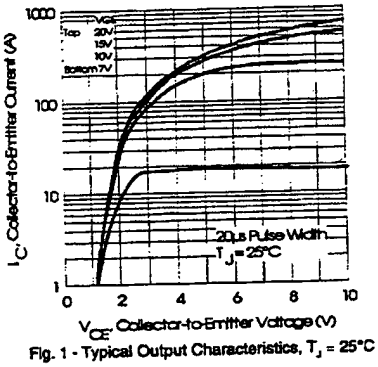


Fig. 1 - Typical Output Characteristics, T_J = 25°C

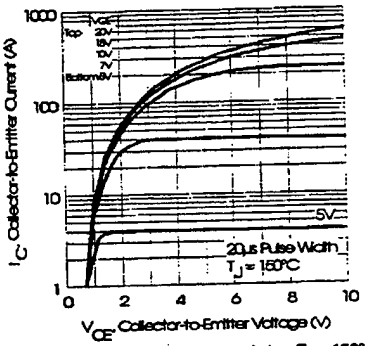


Fig. 2 - Typical Output Characteristics, T_J = 150°C

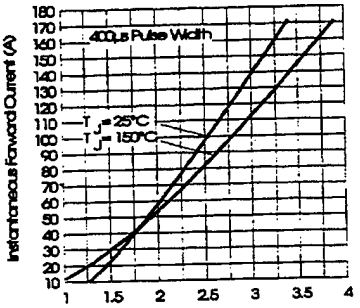


Fig. 3 - Typical Output Characteristics

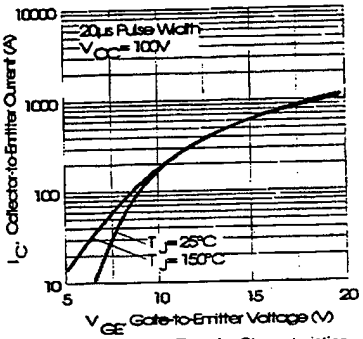


Fig. 4 - Typical Transfer Characteristics

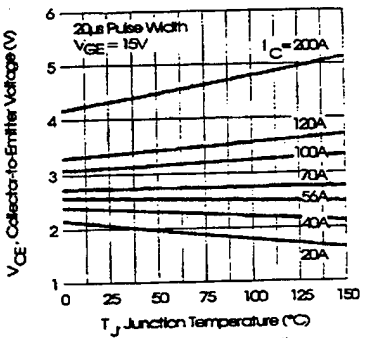


Fig. 5 - Collector-to-Emitter Saturation Typical Voltage vs. Junction Temperature

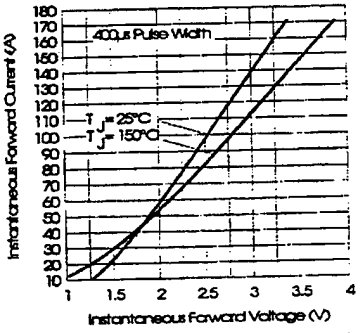


Fig. 6 - Forward Voltage Drop Characteristics

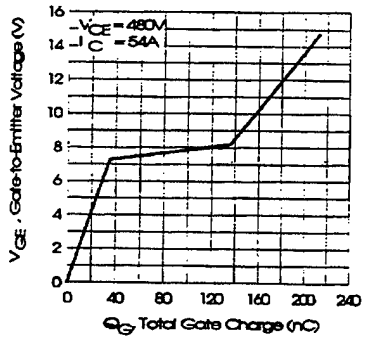


Fig. 7 - Typical Gate Charge vs. Gate-to-Emitter Voltage

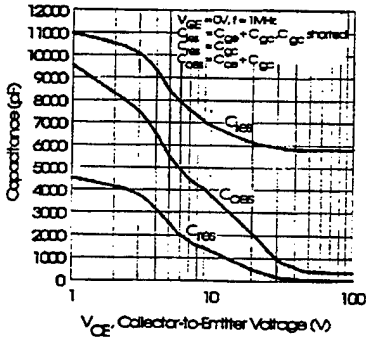


Fig. 8 - Typical Capacitance vs. Collector-to-Emitter Voltage

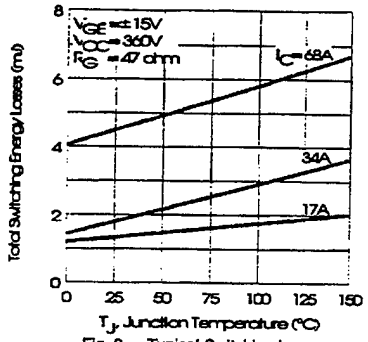


Fig. 9 - Typical Switching Losses vs. Junction Temperature

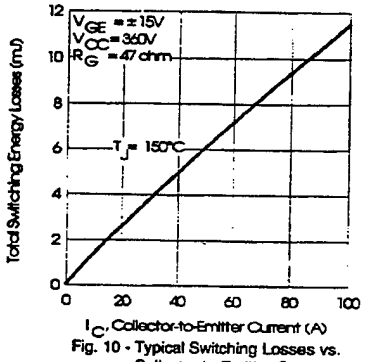


Fig. 10 - Typical Switching Losses vs. Collector-to-Emitter Current

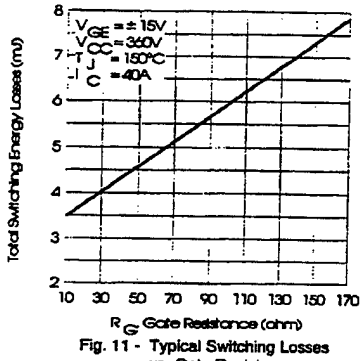


Fig. 11 - Typical Switching Losses vs. Gate Resistance

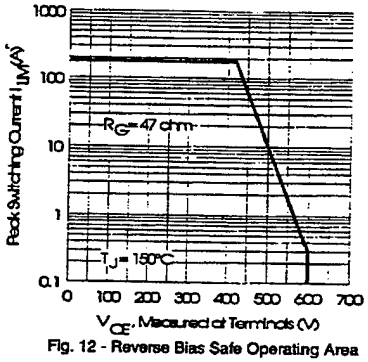


Fig. 12 - Reverse Bias Safe Operating Area

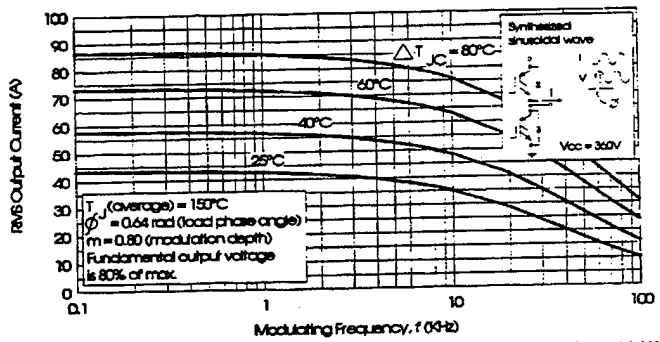


Fig. 13 - Typical RMS Output Current per phase vs. Frequency (Synthesized Sinusoidal Wave)

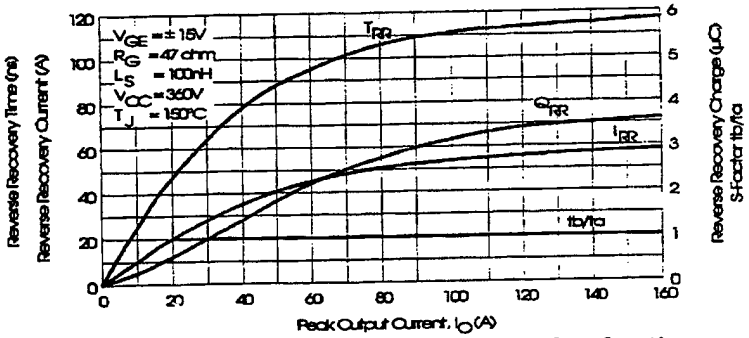


Fig. 14 - Typical Diode Recovery Characteristics as Function of Output Current I_o

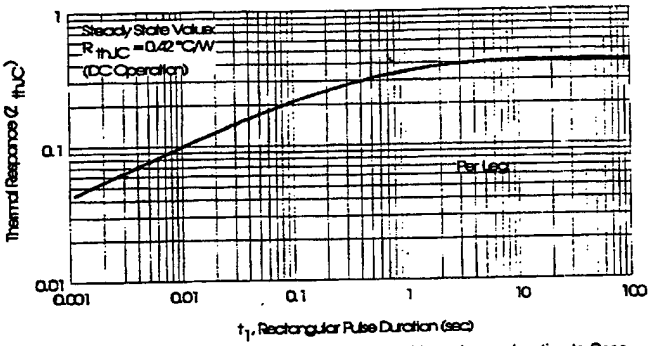


Fig. 15 - Maximum Effective Transient Thermal Impedance, Junction-to-Case

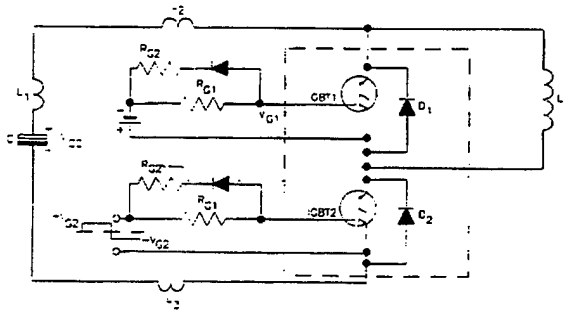


Fig. 16 - Test Circuit for Measurement of I_{LM}, E_{ON}, E_{OFF}, Q_{RR}, I_{RR}, $t_{D(ON)}$, t_r, $t_{D(OFF)}$, t_f

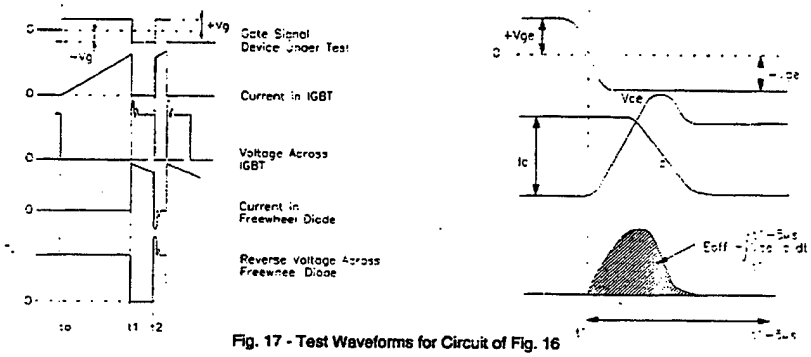


Fig. 17 - Test Waveforms for Circuit of Fig. 16

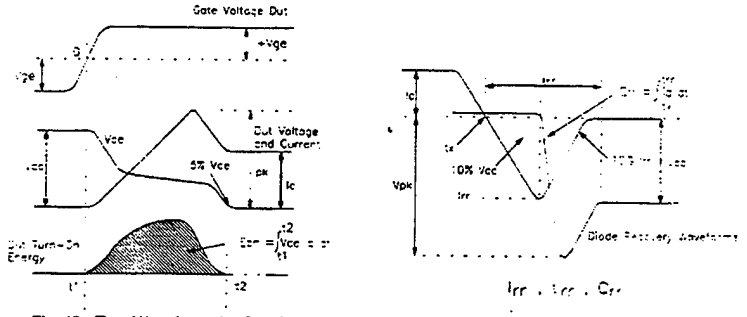


Fig. 18 - Test Waveforms for Circuit of Fig. 16, Defining E_{ON}, E_{REC}, $t_{D(ON)}$, t_r, I_{RR}, t_{RR}, Q_{RR}

B. UNITRODE UC3637 PWM DRIVER IC

DESCRIPTION

The UC1637 is a pulse width modulator circuit intended to be used for a variety of PWM motor drive and amplifier applications requiring either uni-directional or bi-directional drive circuits. When used to replace conventional drivers, this circuit can increase efficiency and reduce component costs for many applications. All necessary circuitry is included to generate an analog error signal and modulate two bi-directional pulse train outputs in proportion to the error signal magnitude and polarity.

This monolithic device contains a sawtooth oscillator, error amplifier, and two PWM comparators with ±100mA output stages as standard features. Protection circuitry includes under-voltage lockout, pulse-by-pulse current limiting, and a shutdown port with a 2.5V temperature compensated threshold.

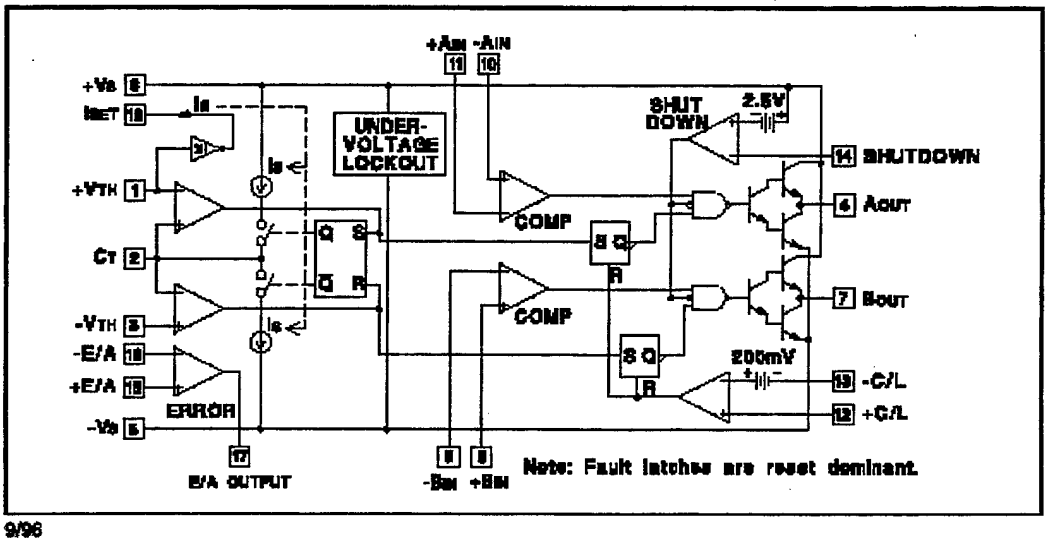
The UC1637 is characterized for operation over the full military temperature range of -55°C to +125°C, while the UC2637 and UC3637 are characterized for -25°C to +85°C and 0°C to +70°C, respectively.

ABSOLUTE MAXIMUM RATINGS (Note 1)

Supply Voltage (±Vs)	±20V
Output Current, Source/Sink (Pins 4, 7)	500mA
Analog Inputs (Pins 1, 2, 3, 8, 9, 10, 11 12, 13, 14, 15, 16)	±Vs
Error Amplifier Output Current (Pin 17)	±20mA
Oscillator Charging Current (Pin 18)	-2mA
Power Dissipation at TA = 25°C (Note 2)	1000mW
Power Dissipation at TC = 25°C (Note 2)	2000mW
Storage Temperature Range	-65°C to +150°C
Lead Temperature (Soldering, 10 Seconds)	+300°C

Note 1: Currents are positive into, negative out of the specified terminal.
Note 2: Consult Packaging Section of Databook for thermal limitations and considerations of package.

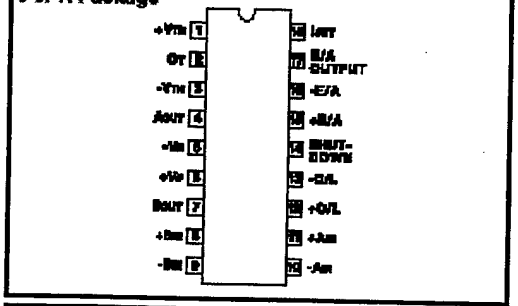
BLOCK DIAGRAM



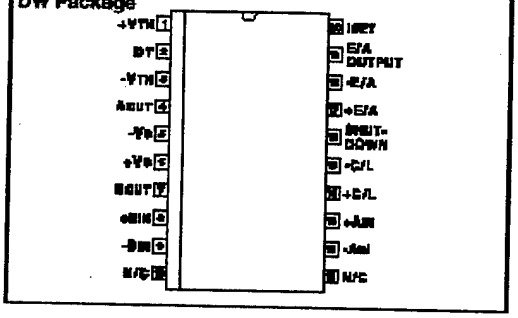
9/96

CONNECTION DIAGRAM

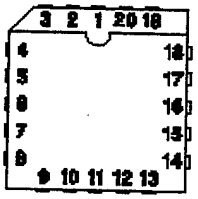
DIL-18 (TOP VIEW)
J or N Package



SOIC-20 (TOP VIEW)
DW Package



PLCC-20, LCC-20
(TOP VIEW)
Q, L Packages



PACKAGE PIN FUNCTION	
FUNCTION	PIN
+VTH	1
CT	2
-VTH	3
AOUT	4
-VS	5
N/C	6
+VS	7
BOUT	8
+BIN	9
-BIN	10
-AIN	11
+AIN	12
+C/L	13
-C/L	14
SHUTDOWN	15
N/C	16
+E/A	17
-E/A	18
E/A OUTPUT	19
ISET	20

ELECTRICAL CHARACTERISTICS:

Unless otherwise stated, these specifications apply for TA = -55°C to +125°C for the UC1637; -25°C to +85°C for the UC2637; and 0°C to +70°C for the UC3637; +VS =+15V, -VS = - 15V, +VTH = 5V, -VTH = -5V, RT = 16.7kW, CT = 1500pF, TA=TJ.

PARAMETER	TEST CONDITIONS	UC1637/UC2637			UC3637			UNITS
		MIN	TYP	MAX	MIN	TYP	MAX	
Oscillator								
Initial Accuracy	TJ = 25°C (Note 6)	9.4	10	10.6	9	10	11	kHz
Voltage Stability	VS = ±5V to ±20V, VPIN 1 = 3V, VPIN 3 = -3V		5	7		5	7	%
Temperature Stability	Over Operating Range (Note 3)		0.5	2		0.5	2	%
+VTH Input Bias Current	VPIN 2 = 6V	-10	0.1	10	-10	0.1	10	µA
-VTH Input Bias Current	VPIN 2 = 0V	-10	-0.5		-10	-0.5		µA
+VTH, -VTH Input Range		+VS-2		-VS+2	+VS-2		-VS+2	V
Error Amplifier								
Input Offset Voltage	VCM = 0V		1.5	5		1.5	10	mV
Input Bias Current	VCM = 0V		0.5	5		0.5	5	µA
Input Offset Current	VCM = 0V		0.1	1		0.1	1	µA
Common Mode Range	VS = ±2.5 to 20V	-VS+2		+VS	-VS+2		+VS	V
Open Loop Voltage Gain	RL = 10k	75	100		80	100		dB
Slew Rate			15			15		V/µs
Unity Gain Bandwidth			2			2		MHz
CMRR	Over Common Mode Range	75	100		75	100		dB
PSRR	VS = ±2.5 to ±20V	75	110		75	110		dB

ELECTRICAL CHARACTERISTICS:

Unless otherwise stated, these specifications apply for TA = -55°C to +125°C for the UC1637; -25°C to +85°C for the UC2637; and 0°C to +70°C for the UC3637: VS =+15V, -VS = - 15V, +VTH = 5V, -VTH = -5V, RT = 16.7kW, CT = 1500pF, TA=TJ.

PARAMETERS	TEST CONDITIONS	UC1637/UC2637			UC3637			UNITS
		MIN	TYP	MAX	MIN	TYP	MAX	
Error Amplifier (Continued)								
Output Sink Current	$V_{PIN\ 17}$ = 0V		-50	-20		-50	-20	mA
Output Source Current	$V_{PIN\ 17}$ = 0V	5	11		5	11		mA
High Level Output Voltage		13	13.6		13	13.6		V
Low Level Output Voltage			-14.8	-13		-14.8	-13	V
PWM Comparators								
Input Offset Voltage	V_{CM} = 0V		20			20		mV
Input Bias Current	V_{CM} = 0V		2	10		2	10	µA
Input Hysteresis	V_{CM} = 0V		10			10		mV
Common Mode range	Vs = ±5V to ±20V	-Vs+1		+Vs-2	-Vs+1		+Vs-2	V
Current Limit								
Input Offset Voltage	V_{CM} = 0V, TJ = 25°C	190	200	210	180	200	220	mV
Input Offset Voltage T.C.			-0.2			-0.2		mV/°C
Input Bias Current		-10	-1.5		-10	-1.5		µA
Common Mode Range	Vs = ±2.5V to ±20V	-Vs		+Vs-3	-Vs		+Vs-3	V
Shutdown								
Shutdown Threshold	(Note 4)	-2.3	-2.5	-2.7	-2.3	-2.5	-2.7	V
Hysteresis			40			40		mV
Input Bias Current	$V_{PIN\ 14}$ = +Vs to -Vs	-10	-0.5		-10	-0.5		µA
Under-Voltage Lockout								
Start Threshold	(Note 5)		4.15	5.0		4.15	5.0	V
Hysteresis			0.25			0.25		mV
Total Standby Current								
Supply Current			8.5	15		8.5	15	mA
Output Section								
Output Low Level	I_{SINK} = 20mA		-14.9	-13		-14.9	-13	V
	I_{SINK} = 100mA		-14.5	-13		-14.5	-13	
Output High Level	I_{SOURCE} = 20mA	13	13.5		13	13.5		V
	I_{SOURCE} = 100mA	12	13.5		12	13.5		
Rise Time	(Note 3) CL = Inf, TJ = 25°C		100	600		100	600	ns
Fall Time	(Note 3) CL = Inf, TJ = 25°C		100	300		100	300	ns

Note 3: These parameters, although guaranteed over the recommended operating conditions, are not 100% tested in production.

Note 4: Parameter measured with respect to +VS (Pin 6).

Note 5: Parameter measured at +VS (Pin 6) with respect to -VS (Pin 5).

Note 6: RT and CT referenced to Ground.

FUNCTIONAL DESCRIPTION

Following is a description of each of the functional blocks shown in the Block Diagram.

Oscillator

The oscillator consists of two comparators, a charging and discharging current source, a current source set terminal, ISET and a flip-flop. The upper and lower threshold of the oscillator waveform is set externally by applying a voltage at pins +VTH and -VTH respectively. The +VTH terminal voltage is buffered internally and also applied to the ISET terminal to develop the capacitor charging current through RT. If RT is referenced to -VS as shown in Figure 1, both the threshold voltage and charging current will vary proportionally to the supply differential, and the oscillator frequency will remain constant. The triangle waveform oscillators frequency and voltage amplitude is determined by the external components using the formulas given in Fig. 1.

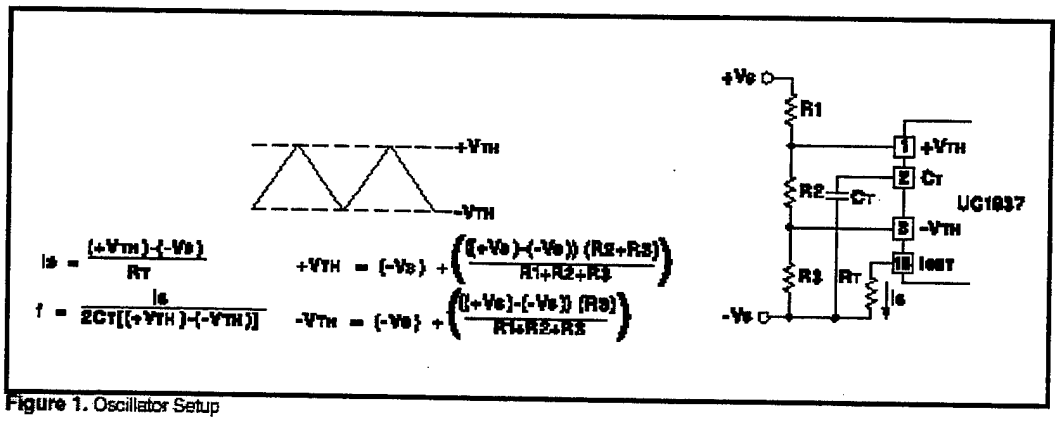


Figure 1. Oscillator Setup

PWM Comparators

Two comparators are provided to perform pulse width modulation for each of the output drivers. Inputs are uncommitted to allow maximum flexibility. The pulse width of the outputs A and B is a function of the sign and amplitude of the error signal. A negative signal at Pin 10 and 8 will lengthen the high state of output A and shorten the high state of output B. Likewise, a positive error signal reverses the procedure. Typically, the oscillator waveform is compared against the summation of the error signal and the level set on Pin 9 and 11.

MODULATION SCHEMES

Case A Zero Deadtime (Equal voltage on Pin 9 and Pin 11)
In this configuration, maximum holding torque or stiffness and position accuracy is achieved. However, the power input into the motor is increased. Figure 3A shows this configuration.

Case B Small Deadtime (Voltage on Pin 9 > Pin 11)
A small differential voltage between Pin 9 and 11 provides the necessary time delay to reduce the chances of momentary short circuit in the output stage during transitions, especially where power-amplifiers are used. Refer to Figure 3B.

Case C Increased Deadtime and Deadband Mode (Voltage on Pin 9 > Pin 11)
With the reduction of stiffness and position accuracy, the power input into the motor around the null point of the servo loop can be reduced or eliminated by widening the window of the comparator circuit to a degree of acceptance. Where position accuracy and mechanical stiffness is unimportant, deadband operation can be used. This is shown in Figure 3C.

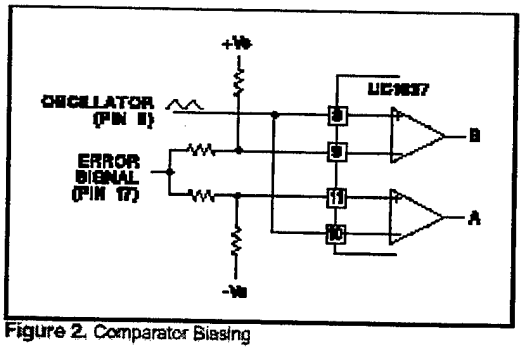


Figure 2. Comparator Biasing

Output Drivers
Each output driver is capable of both sourcing and sinking 100mA steady state and up to 500mA on a pulsed basis for rapid switching of either POWERFET or bipolar transistors. Output levels are typically -VS + 0.2V @50mA low level and +VS - 2.0V @50mA high level.

Error Amplifier
The error amplifier consists of a high slew rate (15V/ms) op-amp with a typical 1MHz bandwidth and low output impedance. Depending on the ±VS supply voltage, the common mode input range and the voltage output swing is within 2V of the VS supply.

Under-Voltage Lockout
An under-voltage lockout circuit holds the outputs in the low state until a minimum of 4V is reached. At this point, all internal circuitry is functional and the output drivers are enabled. If external circuitry requires a higher starting voltage, an over-riding voltage can be programmed through the shutdown terminal as shown in Figure 4.

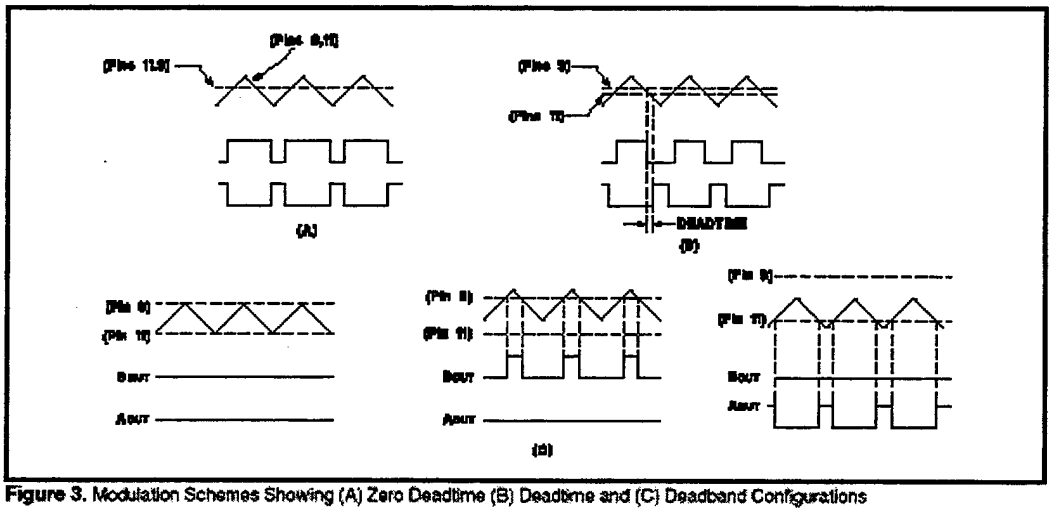


Figure 3. Modulation Schemes Showing (A) Zero Deadtime (B) Deadtime and (C) Deadband Configurations

Shutdown Comparator
The shutdown terminal may be used for implementing various shutdown and protection schemes. By pulling the terminal more than 2.5V below VIN, the output drivers will be enabled. This can be realized using an open collector gate or NPN transistor biased to either ground or the negative supply. Since the threshold is temperature stabilized, the comparator can be used as an accurate low voltage lockout (Figure 4) and/or delayed start as in Figure 5. In the shutdown mode the outputs are held in the low state.

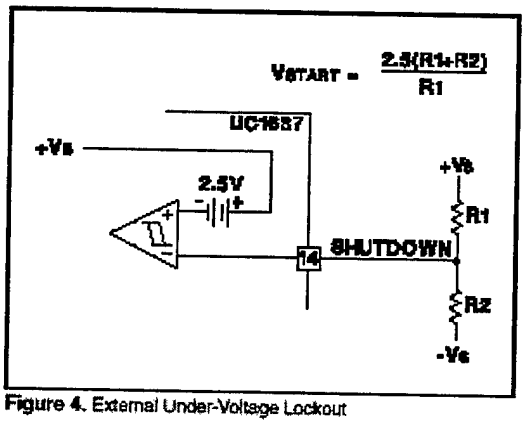


Figure 4. External Under-Voltage Lockout

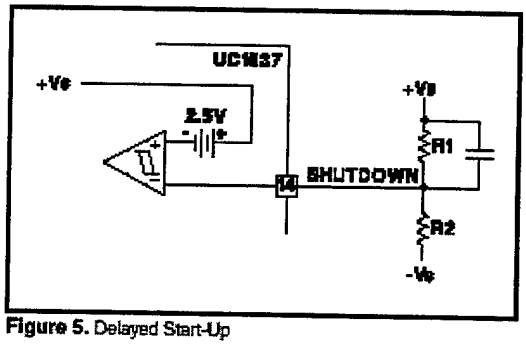


Figure 5. Delayed Start-Up

Current Limit

A latched current limit amplifier with an internal 200mV offset is provided to allow pulse-by-pulse current limiting. Differential inputs will accept common mode signals from -VS to within 3V of the +VS supply while providing excellent noise rejection. Figure 6 shows a typical current sense circuit.

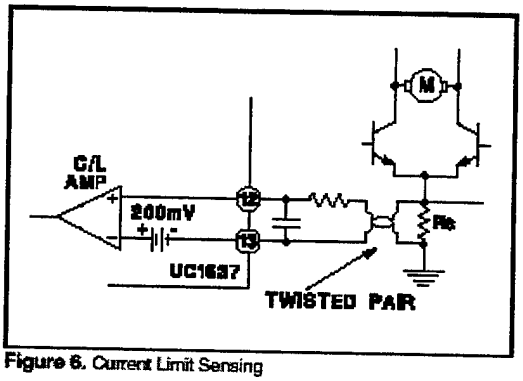


Figure 6. Current Limit Sensing

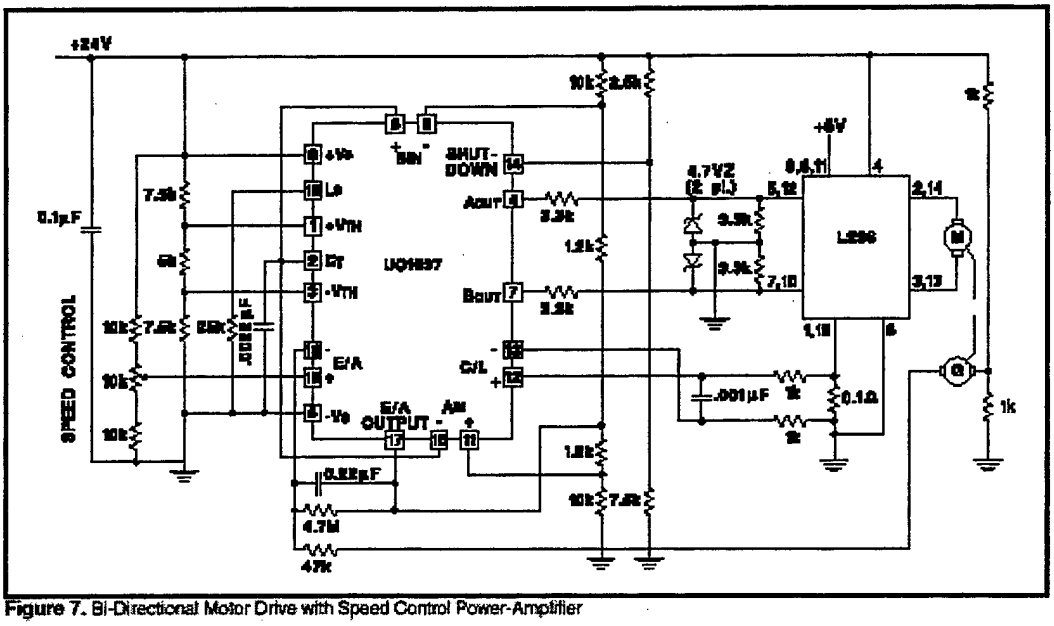


Figure 7. Bi-Directional Motor Drive with Speed Control Power-Amplifier

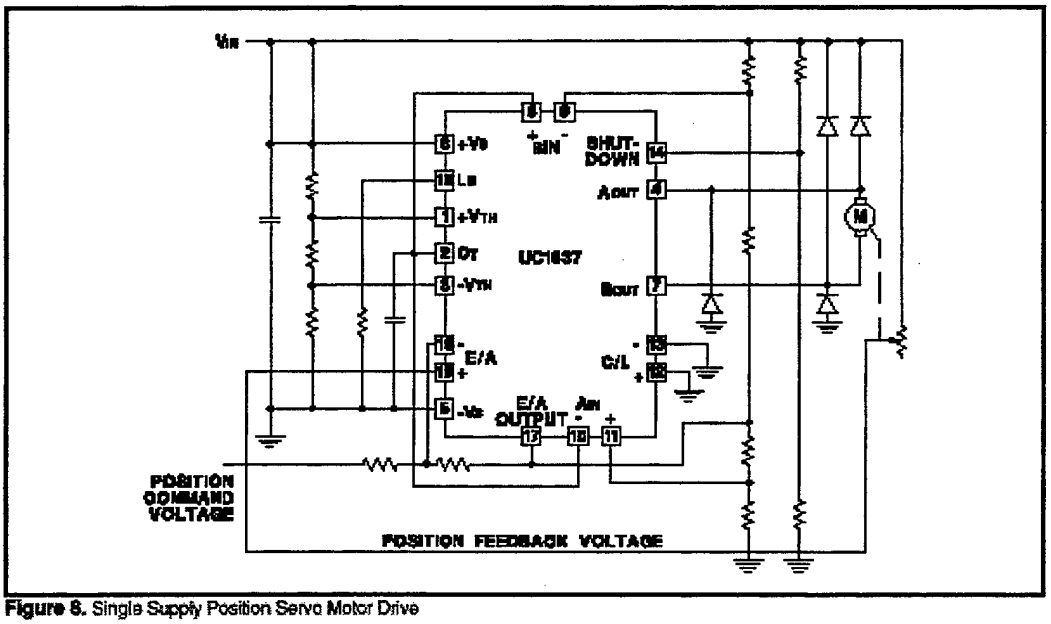


Figure 8. Single Supply Position Servo Motor Drive

APPENDIX B. SIMULINK BLOCK DIAGRAMS AND MATLAB CODE

A. SOURCE BUCK CONVERTER

The Matlab m-files and Simulink block diagram used to generate first-cut gain selections for the Source Buck Converter are provided below.

1. Matlab M-file buckcons.m

```
%************************************************
% buckcons.m
%
% Specifications:
%
% 90 A IGBT's
% Analog controller patterned after SSCM.
%
% Want continuous operations between 10% and
% 100% loading.
%
% Simulation sequence cycles through
% startup -> 10% loading -> full loading.
%
% Written by: Mike Badorf
% Last mod:  2JUN97
%************************************************

%************************************************
% Constant Definitions
%************************************************
E       = 400.0;        % input voltage of buck
Vcd     = 305.0;        % desired output voltage of buck: 10%
Pout    = 9e3;          % output power
Cf      = 400e-6;       % chosen based on fres << fline & parts
Lfilt   = 0.425e-3;     % actual filter inductance value
Cfilt   = 2000e-6;      % actual filter capacitance value
rx_full = 9.68;         % low R value  - full load current
rx_ten  = 96.8;         % high R value - 10% load current
rxstart = rx_ten;       % starting resistance
d       = Vcd/E;        % duty cycle
fline   = 360.0;        % 360 Hz ripple from 6 pulse rectifier
f       = 20e3;         % switching frequency
T       = 1/f;          % switching period
pw      = d*T;          % pulse width for switching noise SS model
L       = 0.760e-3;     % actual inductance value for buck 1
%************************************************
```

```
% Determination of Lcrit: low load R determines
%************************************************
R    = rx_ten;
Lcrit = (T*R/2)*(1-d);    % critical inductance

%************************************************
% Determination of delta_I, Imax, Imin, ILavg
% and I_load at 10% load (ll).
%************************************************
delta_I  = ((E - Vcd)/L)*d*T;
Imax_ll  = d*E*(1/R + (1-d)*T/(2*L));      % ll max L current
Imin_ll  = d*E*(1/R - (1-d)*T/(2*L));      % ll min L current
ILavg_ll  = (Imax_ll+Imin_ll)/2;           % ll avg L current
I_load_ll = Vcd/R;                         % ll current

%************************************************
% Determination of Imax, Imin, ILavg
% and I_load at full load (fl).
%************************************************
R     = rx_full;
Imax   = d*E*(1/R + (1-d)*T/(2*L));        % fl max L current
Imin   = d*E*(1/R - (1-d)*T/(2*L));        % fl min L current
ILavg   = (Imax+Imin)/2;                   % fl avg L current
I_load  = Vcd/R;                           % fl current

%************************************************
% Determination of resonant peak for L / Cf LPF.
%************************************************
fres = 1/(2*pi*sqrt(L*Cf));

%************************************************
% Determination of resonant peak for
% input Lfilt / Cfilt LPF.
%************************************************
fres_filt = 1/(2*pi*sqrt(Lfilt*Cfilt));
```

2. Matlab M-file buckstrt.m

```
%************************************************
% buckstrt.m
%
% Provides startup parameters for 9 kW, 20 kHz
% buck chopper.
%
% Calls buckcons.m, a constant file developed
% for above buck chopper.
%
% Written by: Mike Badorf
% Last mod:  2JUN97
%************************************************

clear          % clear MATLAB memory
buckcons       % establish buck constants

%************************************************
% Bessel Pole specifications
%************************************************
wo = 3250;

poles = wo*[ -0.942 -0.74550+0.7112*i -0.7455-0.7112*i];

%************************************************
% desired characteristic polynomial used to
% determine gains
%************************************************
S    = poly(poles);

%************************************************
% Gain calculations:  Calculated using low
% starting resistance.  Gains are utilized as
% first run starting point.  Entry of manual
% gains can follow to give best overall
% response over all operating conditions.  Use
% board resistors in practice to set gains.
%************************************************
R  = rxstart;
hi = ((L*R*Cf*S(2))-L)/(E*R*Cf)
hn = (L*Cf*S(4))/E + hi*L/2
hv = ((L*R*Cf*S(3))-(E*hi)-R)/(E*R)

%************************************************
% Function to verify poles obtained with gains
% match desired poles.
%************************************************
Am = [ 0        -1/L                E/L
     1/Cf     -1/(R*Cf)             0
    -hv/Cf    (hi/L)+(hv/(R*Cf))-hn    -hi*E/L ];
```

```
closed_syspole = eig(Am)     % check for correct poles

%************************************************
% SIMULINK no load parameters (are in addition
% to constants determined in buckcons.m)
%************************************************

% ABCD state space parameters
A = [ -1/(R*Cf)  1/Cf
    -1/L      0   ];
B = [ 0
    E/L ];
C = [ 1 0
    0 1 ];
D = [ 0
    0 ];

%************************************************
% Update initial conditions for integrators
%************************************************
Vciv    = 304;              % initial Vc integrator value
iLiv    = Vciv/R;           % initial hn integrator value
hnintiv = 0.0;              % initial hn integrator value
dutyiv  = 304/E;            % initial duty cycle value
tstart  = 0.0;              % simulation start time
tstop   = 0.012;            % simulation stop time
```

3. Matlab M-file buckfull.m

```
%************************************************
% buckfull.m
%
% Provides full load parameters for 9 kW,
% 20 kHz buck chopper.
%
% Written by: Mike Badorf
% Last mod:  2JUN97
%************************************************

R = rx_full;    % full load resistance value

%************************************************
% Function to verify poles obtained with gains
% match desired poles.
%************************************************
Am = [ 0        -1/L                E/L
     1/Cf     -1/(R*Cf)             0
    -hv/Cf    (hi/L)+(hv/(R*Cf))-hn    -hi*E/L ];

syspole = eig(Am);    % check for correct poles

%************************************************
% SIMULINK no load parameters (are in addition
% to constants determined in buckcons.m)
%************************************************

% ABCD state space parameters
A = [ -1/(R*Cf)  1/Cf ;  -1/L      0    ]; B = [ 0;     E/L ];
C = [ 1 0;     0 1 ]; D = [ 0;     0 ];

%************************************************
% Data passdown before buckfull.m simulation
% begins.
%************************************************
Vc_response = Vc;
iL_response = iL;
duty_response = duty;
timer = time;

%************************************************
% Update initial conditions for integrators in
% buckfull.m simulation.
%************************************************
Vciv   = Vc(length(time));      % Vc initial value for buckfull.m
iLiv   = iL(length(time));      % iL initial value for buckfull.m
dutyiv  = duty(length(time));    % duty initial value for buckfull.m
tstart  = tstop;              % sim. start time for buckfull.m
tstop   = tstop + 0.012;        % simulation stop time for buckfull.m
```

4. Matlab M-file buckten.m

```
%************************************************
% buckten.m
%
% Provides 10% load parameters for 9 kW, 20 kHz
% buck chopper.
%
% Written by: Mike Badorf
% Last mod:  2JUN97
%************************************************

R = rx_ten;    % 10% load resistance value

%************************************************
% Function to verify poles obtained with gains
% match desired poles.
%************************************************
Am = [ 0        -1/L                E/L
     1/Cf     -1/(R*Cf)             0
    -hv/Cf    (hi/L)+(hv/(R*Cf))-hn   -hi*E/L];

syspole = eig(Am);    % check for correct poles

%************************************************
% SIMULINK no load parameters (are in addition
% to constants determined in buckcons.m)
%************************************************

% ABCD state space parameters
A = [ -1/(R*Cf)  1/Cf;    -1/L     0   ]; B =[ 0;    E/L ];
C = [ 1 0;     0 1 ]; D = [ 0;     0 ];

%************************************************
% Data passdown before buck_ten.m simulation
% begins.
%************************************************
Vc_response   = [ Vc_response; Vc];
iL_response   = [ iL_response; iL];
duty_response = [ duty_response; duty];
timer         = [ timer; time];

%************************************************
% Update initial conditions for integrators for
% buckten.m simulation.
%************************************************
Vciv   = Vc(length(time));       % Vc initial value for buckten.m
iLiv   = iL(length(time));       % iL initial value for buckten.m
dutyiv = duty(length(time));     % duty initial value for buckten.m
tstart = tstop;                  % simulation start time for buckten.m
tstop  = tstop + 0.012;          % simulation stop time for buckten.m
```

5. Matlab M-file buckplot.m

```
%************************************************
% buckplot.m
%
% Plot startup, change to 100% loading, &
% change to 10% loading for 9 kW, 20 kHz buck
% chopper.
%
% Written by:  Mike Badorf
% Last mod:  21SEP96
%************************************************

Vc_response   = [ Vc_response; Vc ];
iL_response   = [ iL_response; iL ];
duty_response  = [ duty_response; duty];
timer        = 1000*[ timer; time ];
datastep = 1:1:length(timer);    % Allows plotting fewer points
                                 % than generated.

figure(1)
subplot(3,1,1), plot(timer(datastep), Vc_response(datastep),'b'), grid
title('Vc Response'),ylabel('Vc  (volts)')
subplot(3,1,2), plot(timer(datastep), iL_response(datastep),'b'), grid
title('iL Response'),ylabel('iL  (amps)')
subplot(3,1,3), plot(timer(datastep), 10*duty_response(datastep),'b'), grid
title('d(t) Response'),xlabel('time  (msec)'),ylabel('10*d(t)')
```

6. Simulink Block Diagram

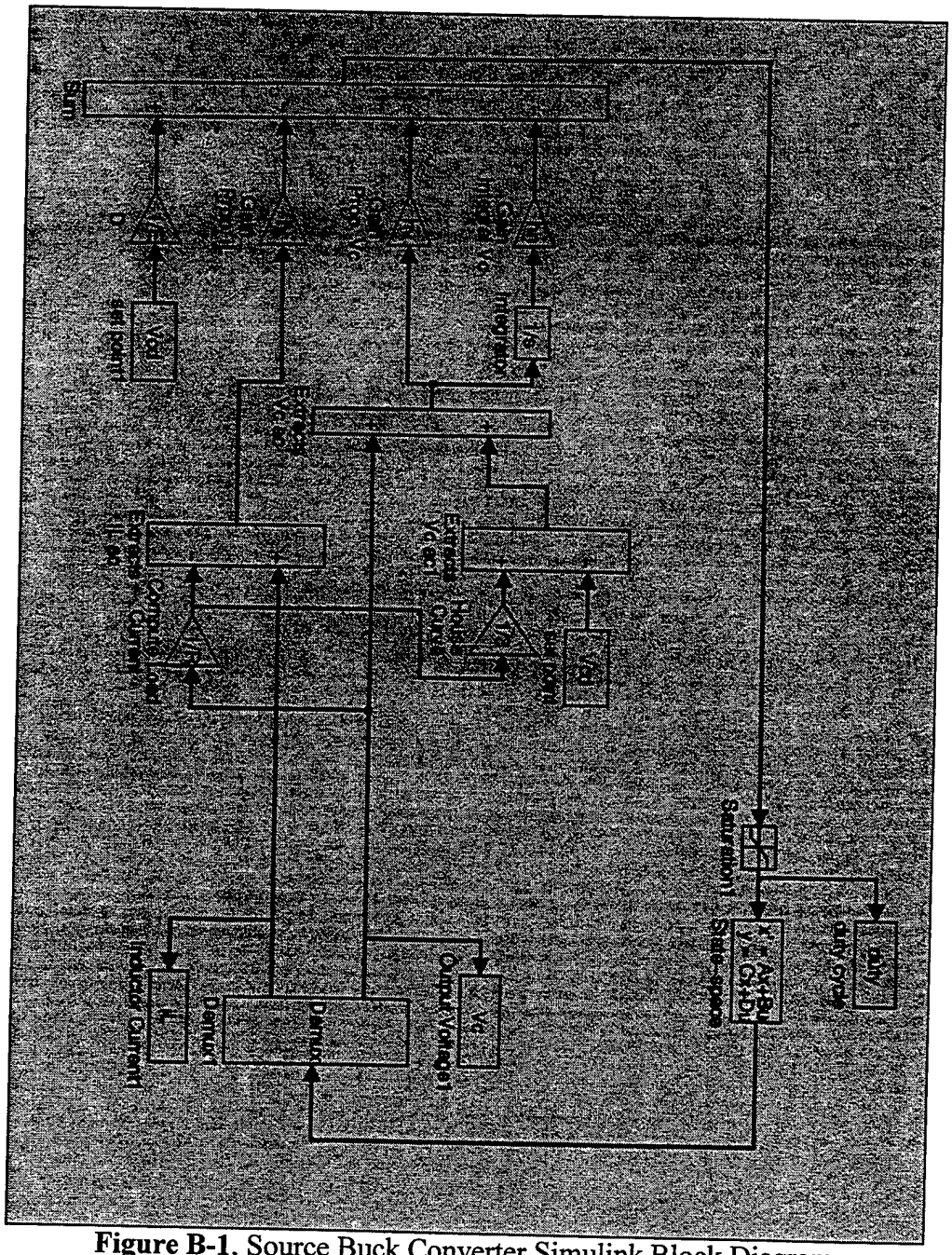


Figure B-1, Source Buck Converter Simulink Block Diagram

B. LOAD BUCK CONVERTER

The Matlab m-file, buckcons.m, and the Simulink block diagram for the Load Buck Converter are provided below. The remaining m-files are provided above since the transient response generation is the same for both the Source and Load Buck Converters.

1. Matlab M-file buckcons.m

```
%************************************************
% buckcons.m
%
% Determining Buck Circuit Parameters
%
% Specifications:
%
% 90 A IGBT's
% Analog controller patterned after SSCM.
%
% Want continuous operations between 10% and
% 100% loading.
%
% Simulation sequence cycles through
% startup -> 10% loading -> full loading.
%
% Written by: Mike Badorf
% Last mod:  21SEP96
%************************************************

%************************************************
% Constant Definitions
%************************************************
E       = 300.0;        % input voltage of buck
Vcd     = 208.0;        % desired output voltage of buck
Pout    = 3e3;          % output power
Cf      = 400e-6;       % chosen based on fres << fline & parts
Lfilt   = 0.425e-3;     % actual filter inductance value
Cfilt   = 460e-6;       % actual filter capacitance value
rx_full = Vcd^2/Pout;   % low R value  - full load current
rx_ten  = 10*rx_full;   % high R value - 10% load current
rxstart = rx_ten;       % starting resistance
d       = Vcd/E;        % duty cycle
fline   = 360.0;        % 360 Hz ripple from 6 pulse rectifier
f       = 20e3;         % switching frequency
T       = 1/f;          % switching period
pw      = d*T;          % pulse width for switching noise SS model
L       = 1.3e-3;       % actual inductance value for buck
```

```
%************************************************
% Determination of Lcrit: low load R determines
%************************************************
R    = rx_ten;
Lcrit = (T*R/2)*(1-d);    % critical inductance

%************************************************
% Determination of delta_I, Imax, Imin, ILavg
% and I_load at 10% load (ll).
%************************************************
delta_I  = ((E - Vcd)/L)*d*T;
Imax_ll  = d*E*(1/R + (1-d)*T/(2*L));      % ll max L current
Imin_ll  = d*E*(1/R - (1-d)*T/(2*L));      % ll min L current
ILavg_ll  = (Imax_ll+Imin_ll)/2;           % ll avg L current
I_load_ll = Vcd/R;                         % ll current

%************************************************
% Determination of Imax, Imin, ILavg
% and I_load at full load (fl).
%************************************************
R     = rx_full;
Imax   = d*E*(1/R + (1-d)*T/(2*L));        % fl max L current
Imin   = d*E*(1/R - (1-d)*T/(2*L));        % fl min L current
ILavg  = (Imax+Imin)/2;                    % fl avg L current
I_load  = Vcd/R;                           % fl current

%************************************************
% Determination of resonant peak for L / Cf LPF.
%************************************************
fres = 1/(2*pi*sqrt(L*Cf));

%************************************************
% Determination of resonant peak for
% input Lfilt / Cfilt LPF.
%************************************************
fres_filt = 1/(2*pi*sqrt(Lfilt*Cfilt));
```

2. Simulink Block Diagram

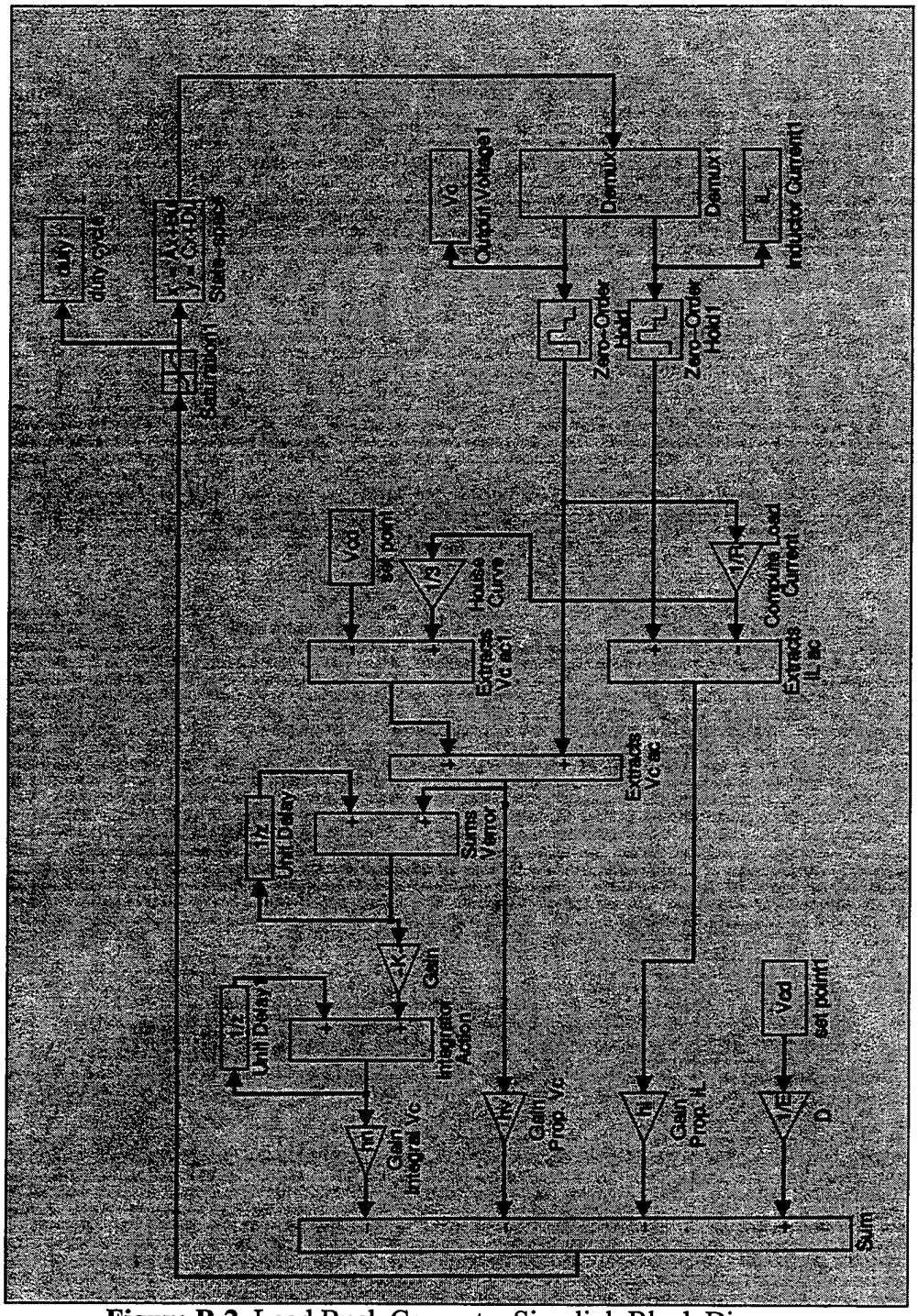


Figure B-2, Load Buck Converter Simulink Block Diagram

APPENDIX C. ACSL CODE

A. SOURCE BUCK CONVERTER GAIN SELECTION CODE

The computer code for this section is comprised of the main ACSL program, two supporting macro files and the command file used to generate transient load studies.

1. Main Program

```
!*******************************************************************************
! sourcebuck.csl
!
! This program simulates the operation of a 9 kW source buck converter unit operating at
! 20 kHz assuming ideal components and no line losses.  The converter has been designed
! to operate as follows:
!                                   400 Vdc input voltage
!                                   300 Vdc nominal output voltage
!                                   continuous ind. current from 10% -> 100% load
!
! The controller for this simulation makes use of the analog control algorithm derived
! in Chapter III.  The file, sourcebuck.cmd, contains the runtime commands utilized to
! exercise this model.
!
! Written by: Michael Badorf
! Last modified: 02JUN97
!*******************************************************************************
INCLUDE 'buck.mac'
INCLUDE 'analog_control.mac'

PROGRAM sourcebuck

INITIAL
        MAXTERVAL maxt = 1.0e-6         !"maximum integration step size"
        MINTERVAL mint = 1.0e-8         !"minimum integration step size"
                                        !"for variable step algorithms "

        CINTERVAL cint = 1.0e-6         !"data communication interval"
        ALGORITHM ialg = 5              !"integration algorithm"
                                        !"4--R.K. 2nd, 5--R.K. 4th"
        NSTEPS    nstp = 1
        CONSTANT tstop = 1.0            !"stop point for integration                "
        CONSTANT Vramppk = 10.0         !"pk value of ramp modulating waveform      "
        CONSTANT swfreq1 = 20000.0      !"switching frequency for buck 1            "
        Tramp1 = 1.0/swfreq1            !"switching period for buck 1               "
END  ! "of initial"
```

```
DYNAMIC

        TERMT (t .GE. (tstop-0.5*cint))

 DERIVATIVE

        Vincap1 = 400.0          !"define a hard input voltage of 400 Vdc"

        !*************************************************************************
        ! source buck converter unit 1
        !*************************************************************************
        tpwm1 = MOD(t,Tramp1)                    !"time elapsed in switching period for buck 1      "
        Vramp1 = Vramppk*tpwm1/Tramp1            !"establish ramp modulating waveform for buck 1"

        !"implement source buck converter unit 1 macro"
        buck(1,Vincap1,iout1,isw1,iL1,Vc,Vramp1,"Lbuck1=760.0e-6","Vdidrop1=2.0",&
             "Vswdrop1=2.0")

        !"implement source buck converter unit 1 multiloop control macro
        analog_control(1,Vincap1,Vc,iout1,iL1,"RCstart1=0.6","hi1=0.015","hv1=0.017",&
                       "hn1=26.09")

        CONSTANT Cbuck1 = 400.0e-6               !"output capcitance  "
        CONSTANT Vcic = 0.0                      !"output voltage IC  "
        pVc = (iL1 - iout1)/Cbuck1               !"output voltage state"
        Vc = INTEG(pVc, Vcic)                    !"output voltage     "

        CONSTANT Rout = 96.8                     !"source buck converter startup load"
                                                 !"10% load value                    "

        iout1 = Vc/Rout                          !"load current"
        !*************************************************************************
        ! end source buck converter unit 1
        !*************************************************************************

 END   !"of derivative"

END  !"of dynamic"

END  !"of program"
```

2. Buck Converter Macro

```
!*****************************************************************************
! buck.mac
!
! This macro governs the conduction status of a buck converter switch and uses this
! condition to develop inductor current utilizing the inductor equation of state.
! Switch and diode losses are modeled as voltage drops.  All other components are ideal.
! The output inductor current is used in the main program to determine output capacitor
! voltage.  Although this step could be done in the macro, the current arrangement
! allows modeling more complex PEBB network configurations.
!
! Written by: Michael Badorf
! Last modified: 02JUN97
!*****************************************************************************
MACRO buck(z,vin,iout,isw,iL,Vc,rampwav,pLbuck,pVdidrop,pVswdrop)

INITIAL

        CONSTANT pLbuck
        CONSTANT pVdidrop
        CONSTANT pVswdrop

        !"assign the initial inductor current"
        CONSTANT iLic&z = 0.0

        !"define and initialize the switch conduction status"
        LOGICAL SW1&z
        SW1&z = .true.          !"T s1 conducts, F s1 blocks"

END

        !"determine conduction status of SW1"
        PROCEDURAL(SW1&z,isw=rampwav,duty&z,iL)

                !"charge inductor when duty GT reframp"
                IF (duty&z .GT. rampwav) THEN

                        SW1&z = .true.
                        isw = iL

                ELSE

                        SW1&z = .false.
                        isw = 0.0

                ENDIF  !"end if (duty&z .GT. rampwav)

        END                     !"of procedural"
```

```
!"Given conduction status of SW1, and whether continuous or not,      "
!" determine derivative of iL                                         "
PROCEDURAL(piL&z=SW1&z,vin,Vc,Vdidrop&z,iL,Vswdrop&z,Lbuck&z)

        "----if SW1 gated, assume can conduct pos. current"
        IF (SW1&z) THEN

                contmode&z = .true.
                piL&z = (vin-Vswdrop&z-Vc)/Lbuck&z

        ELSE

                IF (iL .GT. 0.0) THEN

                        !"if IGBT not gated but iL greater than zero  "
                        !"the diode conducts and the inductor discharges"
                        piL&z = (-Vdidrop&z-Vc)/Lbuck&z

                ELSE

                        !"if IGBT not gated and iL less than or equal"
                        !"to zero, the diode is off and the inductor    "
                        !"current and derivative are zero               "
                        piL&z = 0.0

                ENDIF  !"end if (iL .GT. 0.0)"

        ENDIF  !"end if (SW1&z)"

END             !"of procedural"

!"integrate the inductor state variable"
iL = BOUND(0.0, 100.0, LIMINT(piL&z, iLic&z, 0.0, 100.0))

MACRO END
```

3. Analog Controller Macro

```
!*****************************************************************************
! analog_control.mac
!
! This macro governs the analog control algorithm derived in Chapter III.  In addition
! overcurrent protection is provided to place the controller in pulse-by-pulse
! operations if inductor current exceeds 67.5 amps.
!
! Written by: Michael Badorf
! Last modified: 02JUN97
!*****************************************************************************
MACRO analog_control(z,Vin,Vout,iout,iL,pRCstart,phi,phv,phn)

INITIAL
        CONSTANT pRCstart
        CONSTANT phi
        CONSTANT phv
        CONSTANT phn

        !"assign internal macro constants"
        CONSTANT Vthouseref&z = 305.0           !"top of housecurve"
        CONSTANT houseslope&z = 3.0             !"housecurve slope "
        CONSTANT Vthouseic&z  = 0.0
        CONSTANT Vxic&z       = 0.0
        CONSTANT iLlimit&z    = 67.5            !"max. ind. current before prot. action"
        CONSTANT dutylow&z    = 0.0
        CONSTANT minduty&z    = 0.0
        CONSTANT maxduty&z    = 10.0
        CONSTANT scale&z      = 10.0
END

        pVthouse&z = (Vthouseref&z-Vthouse&z)/RCstart&z     !"reference voltage state   "
        Vthouse&z = INTEG(pVthouse&z, Vthouseic&z)          !"reference voltage         "
        vcref&z = Vthouse&z - iout/houseslope&z
        pVx&z = Vout - vcref&z                      !"voltage state difference   "
        Vx&z = INTEG(pVx&z, Vxic&z)                 !"integral voltage state diff."

        !"control algorithm for source buck converter controller"
        dutyML&z = -hi&z*(iL-iout)-hv&z*pVx&z-hn&z*Vx&z+Vthouse&z/Vin

        !"overcurrent protection for buck converter controller "
         !"if buck converter 1 is operating, implement overcurrent protection "
        IF (iL .GT. iLlimit&z) THEN
                duty&z = dutylow&z                  !"overcurrent"
        ELSE
                duty&z = BOUND(minduty&z, maxduty&z, scale&z*dutyML&z)
        ENDIF           !"end if (iL .GT. iLlimit&z)"

MACRO END
```

4. Command File

```
!*****************************************************************************
! sourcebuck.cmd
!
! This runtime command file exercises the simulation by executing load changes.  In this
! way the transient response of a source buck converter can be analyzed.  The file
! contains a study that allows a source buck converter to start up.  The final values of
! this startup are used as ICs for the transient analysis.  In this way, the transient
! analysis can be run over and over without having to execute a startup every time.
!
! Written by: Michael Badorf
! Last modified: 02JUN97
!*****************************************************************************
s strplt = .t.                  !"one variable per x-axis   "
s calplt = .f.

s devplt = 1                    !"1 for screen output       "
                                !"5 for postscript          "

s ppoplt = .f.                  !"true rotates plot 90 deg  "
s xinspl = 6                    !"x-axis plot units         "
s weditg = .f.                  !"false suppresses data write       "
                                !"each time SCHEDULE occurs     "

s nrwitg = .t.                  !"true enables accumulation of data "
                                !"after a CONTIN                    "

s dpnplt =.f.                   !"no date on plots                  "

!"determines which variables to plot"
prepare t,iL1,iout1,Vc,duty1

!"startup of source buck converter"
proced sourcestartup

        !"source buck startup parameters"
        s Rout = 96.8                   !"10% load"

        !"simulation parameters"
        s ialg = 5
        s maxt = 1.0e-6
        s cint = 1.0e-3
        s tstop = 3.0
        start

        s cint = 1.0e-6
        s tstop = 4.0
        contin

        SAVE /FILE='sourcestartup' /BINARY /STATES /ICS /EVENTS
end
```

```
!"single source buck converter transient operation"
proced trans

        !"initial conditions restoration source converter startup"
        RESTORE /FILE='sourcestartup' /EVENTS

        !"reinitializes states to new ICs"
        REINIT

        !"source buck parameters"
        s Rout = 96.8                                   !"10% load"

        !"simulation parameters"
        s ialg = 5
        s maxt = 1.0e-7
        s cint = 1.0e-6
        s tstop = 0.04
        start

        !"10% -> 100% source buck converter load change"
        s Rout  = 9.68
        s tstop = 0.08
        contin

        !"100% -> 10% source buck converter load change"
        s Rout = 96.8
        s tstop   = 0.12
        contin

        s plt=1
        plot /color=0 /XTAG='(sec)'         &
           duty1                            &
           iout1 /TAG='(amps)'              &
           Vc   /TAG='(volts)'

end
```

B. LOAD BUCK CONVERTER GAIN SELECTION CODE

The computer code for this section is comprised of the main ACSL program and the command file used to generate transient load studies. The buck macro is provided above.

1. Main Program

```
!*******************************************************************************
! loadbuck.csl
!
! This program simulates the operation of a 3 kW load buck converter unit operating at
! 20 kHz assuming ideal components and no line losses. The converter has been designed
! to operate as follows:
!                                300 Vdc input voltage
!                                208 Vdc output voltage
!                                continuous ind. current from 10% -> 100% load
!
! The controller for this simulation makes use of the digital control algorithm derived
! in Chapter III. The file, loadbuck.cmd, contains the runtime commands utilized to
! exercise this model.
!
! Written by: Michael Badorf
! Last modified: 02JUN97
!*******************************************************************************
INCLUDE 'buck.mac'

PROGRAM loadbuck

INITIAL

        MAXTERVAL maxt = 1.0e-6         !"maximum integration step size"

        MINTERVAL mint = 1.0e-8         !"minimum integration step size"
                                        !"for variable step algorithms "

        CINTERVAL cint = 1.0e-6         !"data communication interval"
        ALGORITHM ialg = 5              !"integration algorithm"
                                        !"4--R.K. 2nd, 5--R.K. 4th"

        NSTEPS    nstp = 1
        CONSTANT tstop = 1.0            !"stop point for integration"

        LOGICAL reinitf
        CONSTANT reinitf = .false.
```

```
        !*************************************************************************
        ! Initialization of digital controller values: if reinitializing from previous
        ! run, then this section is skipped by setting reinitf = .true. in the
        ! command file.
        !*************************************************************************
        IF (reinitf) GO TO skipin
                Vthouse_old = 0.0
                Verr_old = 0.0
                Vint_old = 0.0
                count = 0.0
        skipin..CONTINUE

END ! "of initial"

DYNAMIC

        TERMT (t .GE. (tstop-0.5*cint))

 DERIVATIVE

        Vincap3 = 300.0          !"define a hard input voltage of 300 Vdc"

        !*************************************************************************
        ! load buck converter unit 3
        !*************************************************************************

        !"implement load buck converter unit 3 macro
        buck(3,Vincap3,iout3,isw3,iL3,Vc3,count,"Lbuck3=1300.0e-6","Vdidrop3=2.0",&
             "Vswdrop3=2.0")

        CONSTANT Cbuck3 = 400.0e-6          !"output capacitance of load buck   "
        CONSTANT Vcic3  = 0.0               !"output voltage IC                 "

        pVc3 = (iL3 - iout3)/Cbuck3         !"output voltage state              "
        Vc3 = INTEG(pVc3, Vcic3)            !"output voltage                    "

        CONSTANT Rout3 = 144.0              !"load buck converter startup load  "
                                            !"10% load value                    "

        iout3 = Vc3/Rout3                   !"load current"
        !*************************************************************************
        ! end load buck converter unit 3
        !*************************************************************************

        !*************************************************************************
        ! load buck converter unit 3 digital controller overcurrent protection
        !*************************************************************************
        CONSTANT iLlimit3 = 50.0      !"pulse-by-pulse inductor current limit     "
        CONSTANT dutylow3 = 0.0       !"lower admissible duty cycle limit         "
        CONSTANT minduty3 = 0.0       !"lower admissible duty cycle limit         "
        CONSTANT maxduty3 = 475.0     !"upper admissible duty cycle limit         "
        CONSTANT scale3   = 500.0     !"duty between 0-500 counts                 "
```

```
!"if load buck converter operating, implement overcurrent protection"
IF (iL3 .GT. iLlimit3) THEN

        duty3 = dutylow3!"overcurrent"

ELSE
        duty3 = BOUND(minduty3, maxduty3, scale3*dutyML3)

ENDIF  !"end if (iL3 .GT. iLlimit3)"

duty3p = duty3/scale3              !"scales load buck duty cycle to 0 -> 1 range"
!*************************************************************************
! end load buck converter unit 3 digital controller overcurrent protection
!*************************************************************************

END   !"of derivative"

!*************************************************************************
! implement load buck converter digital controller
!*************************************************************************
DISCRETE COUNTER

!"interval statement causes controller to sample every tcount seconds    "
!"set tcount large if load buck is to be idle; when putting on line      "
!"  reset tcount to 1.0e-7, the clock speed of the NSWC digital          "
!"  controller -> perform reset in command file                          "
INTERVAL tcount = 1.0e-7

        count = count + 1                !"clock -> advances   "
                                         !"every 1.0e-7 seconds"

        CONSTANT Vthouseref3 = 208.0          !"top of house curve "

        !"reference voltage with startup "time constant" of 0.06 sec "
        Vthouse3 = Vthouse_old + 700*tcount

        !"memory for old reference voltage value - used on startup   "
        Vthouse_old = Vthouse3

        !"when clock counts to 250 (= 25 usec) update duty cycle for "
        !"application to upcoming switching interval                 "
        IF (count .EQ. 250) THEN

                !"limits reference voltage to top of house curve          "
                IF (Vthouse3 .GE. Vthouseref3) THEN
                        Vthouse3 = Vthouseref3
                ENDIF  !"end if (Vthouse3 .GE. Vthouseref3)

                CONSTANT houseslope3 = 3.0    !"slope of house curve      "
                CONSTANT hi3 = 0.0105         !"current fdbk gain         "
                CONSTANT hv3 = 0.000869       !"voltage fdbk gain         "
                CONSTANT hn3 = 1.733          !"integral voltage fdbk gain  "
```

```
                !"voltage state difference"
                Vprop = Vc3 - Vthouse3

                !"integral voltage state difference"
                Vint = 25.0e-6*(Vprop + Verr_old) + Vint_old

                !"control algorithm for load buck converter controller"
                dutysamp = -hi3*(iL3-iout3)-hv3*Vprop-hn3*Vint+Vthouse3/Vincap3

                Verr_old = Vprop
                Vint_old = Vint

                CALL LOGD(.false.)

            ENDIF                   !"end if (count .EQ. 250)"

            !"apply duty cycle to upcoming switching interval"
            IF (count .EQ. 500) THEN

                CALL LOGD(.false.)

                dutyML3 = dutysamp

                count = 0.0         !"reset clock for next 50 usec cycle"

                CALL LOGD(.false.)

            ENDIF  !"end (count .EQ. 500)

    END                             !"end discrete      "

END  !"of dynamic"

END  !"of program"
```

2. Command File

```
!*****************************************************************************
! loadbuck.cmd
!
! This runtime command file exercises the simulation by executing load changes.  In this
! way the transient response of a load buck converter can be analyzed.  The file
! contains a study that allows a load buck converter to start up.  The final values of
! this startup are used as ICs for the transient analysis.  In this way, the transient
! analysis can be run over and over without having to execute a startup every time.
!
! Written by: Michael Badorf
! Last modified: 02JUN97
!*****************************************************************************
s strplt = .t.                  ! "one variable per x-axis"
s calplt = .f.

s devplt = 1                    ! "1 for screen output"
                                ! "5 for postscript   "

s ppoplt = .f.                  ! "true rotates plot 90 deg"
s xinspl = 6                    ! "x-axis plot units"
s weditg = .f.                  ! "false suppresses data write"
                                ! "each time SCHEDULE occurs"

s nrwitg = .t.                  ! "true enables accumulation of data"
                                ! "after a CONTIN"

s dpnplt =.f.                   !"no date on plots"

!"determines which variables to plot"
prepare t,iL3,iout3,Vc3,duty3p

!"startup of load buck converter"
proced loadstartup

        !"load buck startup parameters"
        s Rout3 = 144.0                 !"10% load"

        !"simulation parameters"
        s ialg = 5
        s maxt = 1.0e-6
        s cint = 1.0e-6
        s tstop = 0.4
        start

        SAVE /FILE='loadstartup' count,Vthouse_old,Verr_old,Vint_old &
                /BINARY /STATES /ICS /EVENTS

end
```

```
!"single load buck converter transient operation"
proced trans

        !"initial conditions restoration from load buck converter startup"
        RESTORE /FILE='loadstartup' /EVENTS

        !"reinitializes state to new ICs and skips digital controller"
        !"initializations in INITIAL section of main program        "
        REINIT
        s reinitf = .true.

        !"load buck parameters"
        s Rout3 = 144.0                          !"10% load"

        !"simulation parameters"
        s ialg = 5
        s maxt = 1.0e-7
        s cint = 1.0e-6
        s tstop = 0.03
        start

        !"10% -> 100% load buck converter load change"
        s Rout3 = 14.4
        s tstop = 0.05
        contin

        !"100% -> 10% load buck converter load change"
        s Rout3 = 144.0
        s tstop   = 0.08
        contin

        plot /color=0 /XTAG='(sec)'        &
           duty3p                          &
           iout1 /TAG='(amps)'             &
           Vc    /TAG='(volts)'

end
```

C. MODE 7 DETAILED SIMULATION CODE

The computer code for this section is comprised of the main ACSL program, three of the four supporting macro files and the command file used to generate transient load studies. The fourth macro file is the analog control macro provided at it was the same for all simulation cases.

1. Main Program

```
!*******************************************************************************
! mode7.csl
!
! This program simulates the operation of parallel source buck converter units (capable)
! of supplying 18 kW) supplying a load buck converter and a hard-switched inverter
! The buck converter models assumes equivalent series resistances for the
! capacitors and inductors. The inverter operates at 6 kHz. The system has been
! designed to operate as follows:
!
!       Source Buck Converter:   400 Vdc input voltage w/ 360 Hz ripple
!                                300 Vdc nominal output voltage
!                                continuous ind. current from 10% -> 100% load
!                                20 kHz switching frequency
!
!       Load Buck Converter:     300 Vdc nominal input voltage
!                                208 Vdc nominal output voltage
!                                continuous ind. current from 10% -> 100% load
!                                20 kHz switching frequency
!
!       Inverter:                300 Vdc nominal input voltage
!                                210 Vac(peak) 3-phase output voltage
!                                24 A(peak) maximum 3-phase current output
!                                6 kHz switching frequency (larger value requires smaller time step)
!
! The buck converter controllers for this simulation makes use of the analog control
! algorithm derived in Chapter III. The inverter is controlled by stationary reference
! frame current feedback as described in reference [8]. The file, mode7.cmd, contains
! the runtime commands utilized to exercise this model.
!
! Written by: Michael Badorf
! Last modified: 02JUN97
!*******************************************************************************
INCLUDE 'buck.mac'
INCLUDE 'sourcefilter.mac'
INCLUDE 'loadfilter.mac'
INCLUDE 'analog_control.mac'
```

```
PROGRAM operational mode 7

INITIAL

        MAXTERVAL maxt = 1.0e-6      !"maximum integration step size    "
        MINTERVAL mint = 1.0e-8

        CINTERVAL cint = 1.0e-6      !"data communication interval      "
        ALGORITHM ialg = 5           !"integration algorithm            "
                                     !"4 -- R.K. 2nd, 5 -- R.K. 4th     "
        NSTEPS   nstp = 1
        CONSTANT tstop = 1.0         !"stop point for integration"

        !*************************************************************************
        ! Electrical constants
        !*************************************************************************
        pi   = 3.14159265
        twopi = 2.0*pi
        pi2o3 = 2.0*3.14159265/3.0   !"120 degrees in radians           "
        we   = 60.0*2.0*pi           !"60 Hz converted to 377 rad/sec   "

        !*************************************************************************
        ! Switch and diode losses
        !*************************************************************************
        CONSTANT Vdidrop = 2.0              !"voltage drop across diodes          "
        CONSTANT Vswdrop = 2.0              !"voltage drop across IGBTs           "

        !*************************************************************************
        ! Buck Converter and PWM settings
        !*************************************************************************
        CONSTANT swfreq1 = 20000.0   !"switching frequency for buck 1"
        Tbuck1 = 1.0/swfreq1         !"switching period for buck 1   "

        CONSTANT swfreq2 = 20000.0   !"switching frequency for buck 2"
        Tbuck2 = 1.0/swfreq2         !"switching period for buck 2   "

        CONSTANT Vpk = 10.0          !"pk value of ramp mod wvfrm"

        CONSTANT finv  = 6000.0      !"switching frequency for inverter  "
        Tinv  = 1.0/finv             !"switching period of inverter      "
        Tinv_over2 = Tinv/2.0        !"half the inverter switching period "
        slope = 4.0*Vpk*finv         !"reference triangle slope for inverter"

        !*************************************************************************
        ! Logical switches used to turn load buck converter on and off
        !
        !       true  = unit on
        !       false = unit off
        !*************************************************************************
        LOGICAL buck1300
        CONSTANT buck1300 = .false.
```

```
        LOGICAL reinitf
        CONSTANT reinitf = .false.

        IF (reinitf) GO TO skipin
        !*************************************************************************
        ! Initialization of digital controller values: if reinitializing from previous
        ! run, then this section is skipped by setting reinitf = .true. in the
        ! command file.
        !*************************************************************************
                Vthouse3 = 0.0
                Verr_old = 0.0
                Vint_old = 0.0
                count    = 0.0

        !*************************************************************************
        ! Initialization of inverter values:  if reinitializing from previous
        ! run, then this section is skipped by setting reinitf = .true. in the
        ! command file.
        !*************************************************************************
                Vag=0.0
                Vbg=0.0
                Vcg=0.0
                i1 = 0.0
                i2 = 0.0
                i3 = 0.0

        skipin..CONTINUE

END ! "of initial"

DYNAMIC

        TERMT (t .GE. (tstop - 0.5*cint))

 DERIVATIVE

        !*************************************************************************
        ! develops 6-pulse voltage input to source buck converters
        !*************************************************************************
        theta = we*t
        theta_s = mod(theta, twopi)         !"statement causes theta_s to vary from 0 -> 2*pi     "
                                            !"then reset back to 0 for next cycle                  "

        CONSTANT Vr_ave = 208.0             !"average value of 6 pulse waveform                    "

        Vrms = pi*Vr_ave/(3.0*SQRT(6.0))          !"rms value of ripple "
```

```
!"generates 6-pulse waveform "
PROCEDURAL(Vr=theta_s,Vrms,we)
  IF ((theta_s .GE. 0.0) .AND. (theta_s .LE. pi/3.0)) THEN
        Vr = -SQRT(6.0)*Vrms*cos(we*t + 5.0*pi/6.0)
  ELSE IF ((theta_s .GE. pi/3.0) .AND. (theta_s .LE. 2.0*pi/3.0)) THEN
        Vr = SQRT(6.0)*Vrms*cos(we*t - pi/2.0)
  ELSE IF ((theta_s .GE. 2.0*pi/3.0) .AND. (theta_s .LE. pi)) THEN
        Vr = -SQRT(6.0)*Vrms*cos(we*t + pi/6.0)
  ELSE IF ((theta_s .GE. pi) .AND. (theta_s .LE. 4.0*pi/3.0)) THEN
        Vr = SQRT(6.0)*Vrms*cos(we*t + 5.0*pi/6.0)
  ELSE IF ((theta_s.GE.4.0*pi/3.0).AND.(theta_s .LE. 5.0*pi/3.0)) THEN
        Vr = -SQRT(6.0)*Vrms*cos(we*t - pi/2.0)
  ELSE IF ((theta_s .GE. 5.0*pi/3.0) .AND. (theta_s .LE. 2.0*pi)) THEN
        Vr = SQRT(6.0)*Vrms*cos(we*t + pi/6.0)
  ENDIF          !"end multiple if-then-else"
END              !"of procedural"

Vin = 2*Vr                !"voltage at source buck converter input filters"
!*****************************************************************************
! end 6-pulse voltage input to source buck converters
!*****************************************************************************

!*****************************************************************************
! source buck converter unit 1
!*****************************************************************************

!"input LC filter for source buck converter unit 1"
sourcefilter(1,Vin,Vincap1,iLfilt1,isw1,"Rlesr_in1=0.2","Lfilt1=425.0e-6",&
          "Rcesr_in1=2046.6","Cin1=2000.01e-6","Vcapic1=400.0")

tpwm1 = MOD(t,Tbuck1)          !"time elapsed in switching period for buck          "
Vramp1 = Vpk*tpwm1/Tbuck1      !"establish ramp modulating waveform for buck        "

!"implement source buck converter unit 1 macro"
buck(1,Vincap1,iout1,isw1,iL1,Vout,Vramp1,"Lbuck1=760.0e-6",&
     "Vdidrop1=2.0","Vswdrop1=2.0")

!"implement source buck converter unit 1 multiloop control
analog_control(1,Vincap1,Vout,iout1,iL1,"RCstart1=0.1","hi1=0.015",&
               "hv1=0.017","hn1=26.09")
!*****************************************************************************
! end source buck converter unit 1
!*****************************************************************************

!*****************************************************************************
! source buck converter unit 2
!*****************************************************************************

!"input LC filter for source buck converter unit 2"
sourcefilter(2,Vin,Vincap2,iLfilt2,isw2,"Rlesr_in2=0.2","Lfilt2=425.0e-6",&
          "Rcesr_in2=2046.6","Cin2=2000.01e-6","Vcapic2=400.0")
```

```
tpwm2 = MOD(t,Tbuck2)          !"time elapsed in switching period for buck   "
Vramp2 = Vpk*tpwm2/Tbuck2      !"establish ramp modulating waveform for buck  "

!"implement source buck converter unit 2 macro"
buck(2,Vincap2,iout2,isw2,iL2,Vout,Vramp2,"Lbuck2=875.0e-6",&
      "Vdidrop2=2.0","Vswdrop2=2.0")

!"implement source buck converter unit 2 multiloop control macro"
analog_control(2,Vincap2,Vout,iout2,iL2,"RCstart2=0.1","hi2=0.015",&
                "hv2=0.017","hn2=26.09")
!*****************************************************************************
! end source buck converter unit 2
!*****************************************************************************

CONSTANT Cbuck = 810.0e-6          !"output capcitance                          "
CONSTANT Rout = 96.8               !"source buck converter startup load         "
CONSTANT Rcesr = 0.02              !"capacitor esr                              "
CONSTANT Vcic = 0.0                !"output voltage IC                          "

pVc = (iL1+iL2-iout)/Cbuck         !"output capacitor voltage state     "
Vc = INTEG(pVc, Vcic)              !"voltage across output capacitor    "

Vout = Vc + (iL1+iL2-iout)*Rcesr   !"output voltage                  "

iout = (Vc+(iL1+iL2)*Rcesr+(iLfilt3+iinv)*Rout)/(Rout+Rcesr)      !"total iout       "

iout1 = iL1 - (Cbuck/2)*pVc        !"iout of buck 1          "
iout2 = iL2 - (Cbuck/2)*pVc        !"iout of buck 2          "

!*****************************************************************************
! load buck converter unit 3
!*****************************************************************************

!"input LC filter for load buck converter unit 3"
loadfilter(3,Vout,Vincap3,iLfilt3,isw3,"Rlesr_in3=0.2","Lfilt3=425.0e-6",&
          "Rcesr_in3=0.01","Cin3=460.0e-6","Vcapic3=0.0")

!"implement source buck converter unit 3 macro"
buck(3,Vincap3,iout3,isw3,iL3,Vout3,count,"Lbuck3=1300.0e-6","Vdidrop3=2.0",&
      "Vswdrop3=2.0")

CONSTANT Cbuck3 = 400.0e-6         !"output capcitance                          "
CONSTANT Vcic3 = 0.0               !"output voltage IC                          "
CONSTANT Rout3 = 144.0             !"source buck converter startup load         "
CONSTANT Rcesr_out3 = 0.01         !"load buck converter output                 "
                                   !"capacitor esr                              "

pVc3 = (iL3 - iout3)/Cbuck3        !"capacitor output voltage state             "
Vc3 = INTEG(pVc3, Vcic3)           !"capacitor output voltage                   "

Vout3 = Vc3 + (iL3-iout3)*Rcesr_out3     !"output voltage                       "
iout3 = (Vc3+iL3*Rcesr_out3)/(Rout3+Rcesr_out3)          !"total iout          "
```

```
!*****************************************************************************
! end load buck converter unit 3
!*****************************************************************************

!*****************************************************************************
! load buck converter unit 3 digital controller overcurrent protection
!*****************************************************************************
CONSTANT iLlimit3 = 50.0              !"pulse-by-pulse inductor current limit "
CONSTANT dutylow3 = 0.0               !"lower admissible duty cycle           "
CONSTANT minduty3 = 0.0               !"lower admissible duty cycle limit     "
CONSTANT maxduty3 = 475.0             !"upper admissible duty cycle limit     "
CONSTANT scale3   = 500.0             !"duty between 0-500 counts"

IF (buck1300) THEN

        !"if load buck converter operating, implement overcurrent protection"
        IF (iL3 .GT. iLlimit3) THEN

                duty3 = dutylow3!"overcurrent"

        ELSE

                duty3 = BOUND(minduty3, maxduty3, scale3*dutyML3)

        ENDIF  !"end if (iL3 .GT. iLlimit3)"

ELSE
        !"if load buck converter is idle, duty cycle is zero"
        duty3 = 0.0
        Vout3 = 0.0
        iout3 = 0.0

ENDIF  !"end if (buck1300)"

duty3p = duty3/scale3                 !"scales load buck duty cycle to 0 -> 1 range"

!*****************************************************************************
! end load buck converter unit 3 digital controller overcurrent protection
!*****************************************************************************

!*****************************************************************************
! inverter model
!*****************************************************************************
CONSTANT Lf   = 10.0e-3       !"inverter phase load inductance          "
CONSTANT R4   = 5.0           !"inverter phase load resistance          "
CONSTANT K    = 0.0005        !"PWM gain                                "
CONSTANT K1   = 1260.0        !"proportional gain                       "
CONSTANT K2   = 3750.0        !"integral gain                           "
CONSTANT A    = 24.0          !"desired current amplitude               "
```

```
!"sine - triangle modulation"
tt = mod(t,Tinv)
IF(tt .LT. Tinv_over2) THEN

  Vtri = Vpk - tt*slope

ELSE

  Vtri =  -Vpk + slope*(tt - Tinv_over2)

ENDIF  !"end if (tt .LT. Tinv_over2)"

!"calculate synchronous reference frame angle"
thetae = integ(we,0.0)

!"calculate desired currents"
Idesa = A*cos(thetae)                    !"desired phase A current  "
Idesb = A*cos(thetae - pi2o3)            !"desired phase B current  "
Idesc = A*cos(thetae + pi2o3)            !"desired phase C current  "

!"transform desired currents to stationary qd0 quantities"
Iqsdes = (2.0/3.0)*(Idesa - 0.5*Idesb - 0.5*Idesc)
Idsdes = (sqrt(3.0)/3.0)*( -Idesb + Idesc)

!"transform desired currents to synchronous qd0 quantities"
Iqsedes = (2.0/3.0)*(Idesa*cos(thetae) + Idesb*cos(thetae - pi2o3) &
          + Idesc*cos(thetae + pi2o3))
Idsedes = (2.0/3.0)*(Idesa*sin(thetae) + Idesb*sin(thetae - pi2o3) &
          + Idesc*sin(thetae + pi2o3))

!"transform actual currents to synchronous qd0 quantities"
Iqse = (2.0/3.0)*(ILa*cos(thetae) + ILb*cos(thetae - pi2o3)  &
       +ILc*cos(thetae + pi2o3))
Idse = (2.0/3.0)*(ILa*sin(thetae) + ILb*sin(thetae - pi2o3) &
       +ILc*sin(thetae + pi2o3))

!"determine error currents in stationary reference frame"
Iqserr = Iqsdes - iqss
Idserr = Idsdes - idss

!"generate commanded currents (PI Controller)"
Iqscom = (K*K1*Iqserr + K*K2*integ(Iqserr,0.0))
Idscom = (K*K1*Idserr + K*K2*integ(Idserr,0.0))

!"transform commanded quantaties to ABC frame"
Iacont = Iqscom
Ibcont = -0.5*Iqscom - (sqrt(3.0)/2.0)*Idscom
Iccont = -0.5*Iqscom + (sqrt(3.0)/2.0)*Idscom
```

```
        !"switching control voltages"
        Vacont = Iacont*R4
        Vbcont = Ibcont*R4
        Vccont = Iccont*R4

        !"schedule inverter switching"
        SCHEDULE hia .XN. (Vtri - Vacont)          !"switch A to HI state     "
        SCHEDULE loa .XP. (Vtri - Vacont)          !"switch A to LO state     "
        SCHEDULE hib .XN. (Vtri - Vbcont)          !"switch B to HI state     "
        SCHEDULE lob .XP. (Vtri - Vbcont)          !"switch B to LO state     "
        SCHEDULE hic .XN. (Vtri - Vccont)          !"switch C to HI state     "
        SCHEDULE loc .XP. (Vtri - Vccont)          !"switch C to LO state     "

        !"calculate line voltages"
        Vas = (2.0/3.0)*Vag - (1.0/3.0)*(Vbg + Vcg)
        Vbs = (2.0/3.0)*Vbg - (1.0/3.0)*(Vag + Vcg)
        Vcs = (2.0/3.0)*Vcg - (1.0/3.0)*(Vag + Vbg)

        !"transform to stationary reference frame"
        Vqss = (2.0/3.0)*(Vas - 0.5*Vbs - 0.5*Vcs)
        Vdss = (sqrt(3.0)/3.0)*( -Vbs + Vcs)

        !"find current derivatives in stationary frame"
        piqss = (Vqss - R4*iqss)/Lf
        pidss = (Vdss - R4*idss)/Lf

        !"calculate stationary currents"
        CONSTANT iqssic = 0.0          !"initial q axis current          "
        CONSTANT idssic = 0.0          !"initial d axis current          "
        iqss = integ(piqss,iqssic)
        idss = integ(pidss,idssic)

        !"transform back to ABC quantaties"
        iLa = iqss
        iLb = -0.5*iqss - (sqrt(3.0)/2.0)*idss
        iLc = -0.5*iqss + (sqrt(3.0)/2.0)*idss

        iinv = i1 + i2 + i3

END     !"of derivative"

        !*************************************************************************
        ! implement load buck converter digital controller
        !*************************************************************************
        DISCRETE COUNTER

        !"interval statement causes controller to sample every tcount seconds "
        !"  set tcount large if load buck is to be idle; when putting on line"
        !"  reset tcount to 1.0e-7, the clock speed of the NSWC digital    "
        !"  controller -> perform reset in command file                    "
        INTERVAL tcount = 10.0
```

```
!"execute if block when load buck converter is to be put on line   "
!"  the buck1300 condition tells the program that the load buck
!"  is to be operating;  the tcount condition ensures that tcount "
!"  has been reset since reset only takes effect on following     "
!"  program cycle -> see ACSL Reference Manual for details      "
IF ((buck1300) .AND. (tcount .EQ. 1.0e-7)) THEN

        count = count + 1                   !"clock -> advances   "
                                            !"every 1.0e-7 seconds"

        CONSTANT Vthouseref3 = 208.0        !"top of house curve "

        !"reference voltage with startup "time constant" of 0.06 sec "
        Vthouse3 = Vthouse_old + 700*tcount

        !"memory for old reference voltage value - used on startup   "
        Vthouse_old = Vthouse3

        !"when clock counts to 250 (= 25 usec) update duty cycle for "
        !"application to upcoming switching interval                 "

        IF (count .EQ. 250) THEN

                !"limits reference voltage to top of house curve             "
                IF (Vthouse3 .GE. Vthouseref3) THEN
                        Vthouse3 = Vthouseref3
                ENDIF  !"end if (Vthouse3 .GE. Vthouseref3)

                CONSTANT houseslope3 = 3.0    !"slope of house curve        "
                CONSTANT hi3 = 0.0105         !"current fdbk gain           "
                CONSTANT hv3 = 0.000869       !"voltage fdbk gain           "
                CONSTANT hn3 = 1.733          !"integral voltage fdbk gain"

                !"voltage state difference"
                Vprop = Vout3 - Vthouse3

                !"integral voltage state difference"
                Vint = 25.0e-6*(Vprop + Verr_old) + Vint_old

                !"control algorithm for load buck converter controller"
                dutysamp = -hi3*(iL3-iout3)-hv3*Vprop-hn3*Vint+Vthouse3/Vincap3

                Verr_old = Vprop
                Vint_old = Vint

                CALL LOGD(.false.)

        ENDIF                       !"end if (count .EQ. 250)"
```

```
                    !"apply duty cycle to upcoming switching interval        "
                    IF (count .EQ. 500) THEN

                        CALL LOGD(.false.)

                        dutyML3 = dutysamp
                        count = 0.0                !"reset clock for next 50 usec cycle"

                        CALL LOGD(.false.)

                    ENDIF  !"end (count .EQ. 500)

            ENDIF                              !"end if (buck1300)...  "

        END                                    !"end discrete          "

        DISCRETE hia
            Vag = Vout - Vswdrop
            i1  = iLa
        END    ! "of discrete"

        DISCRETE loa
            Vag = Vswdrop
            i1  = 0.0
        END    ! "of discrete"

        DISCRETE hib
            Vbg = Vout - Vswdrop
            i2  = iLb
        END    ! "of discrete"

        DISCRETE lob
            Vbg = Vswdrop
            i2  = 0.0
        END    ! "of discrete"

        DISCRETE hic
            Vcg = Vout - Vswdrop
            i3  = iLc
        END    ! "of discrete"

        DISCRETE loc
            Vcg = Vswdrop
            i3  = 0.0
        END    ! "of discrete"

END !"of dynamic"

END !"of program"
```

2. Buck Converter Macro

```
!*****************************************************************************
! buck.mac
!
! This macro governs the conduction status of a buck converter switch and uses this
! condition to develop inductor current utilizing the inductor equation of state.
! Switch and diode losses are modeled as voltage drops. Capacitors and inductors are
! modeled with equivalent series resistance in place. The output inductor current is
! used in the main program to determine output capacitor voltage. Although this step
! could be done in the macro, the current arrangement allows modeling more complex PEBB
! network configurations.
!
! Written by: Michael Badorf
! Last modified: 02JUN97
!*****************************************************************************
MACRO buck(z,vin,iout,isw,iL,Vo,rampwav,pLbuck,pVdidrop,pVswdrop)

INITIAL

        CONSTANT pLbuck             !"inductance      "
        CONSTANT pVdidrop           !"diode  losses   "
        CONSTANT pVswdrop           !"switch losses   "
        CONSTANT iLic&z = 0.0       !"inductor current IC"
        CONSTANT Rlesr&z = 0.2      !"inductor esr          "
        CONSTANT Rcesr&z = 0.01     !"capacitor esr     "

        "----define and initialize the switch conduction status"
        LOGICAL SW1&z
        SW1&z = .true.        !"T s1 conducts, F s1 blocks"

END     !"of initial"

        "----determine conduction status of SW1"
        PROCEDURAL(SW1&z,isw=rampwav,duty&z,iL)

                "----charge inductor when duty GT reframp"
                IF (duty&z .GT. rampwav) THEN
                        SW1&z = .true.
                        isw = iL
                ELSE
                        SW1&z = .false.
                        isw = 0.0
                ENDIF
        END                 !"of procedural"
```

```
        "----Given conduction status of SW1, and whether     "
        "   contin or not, determine deriv of iL             "
        PROCEDURAL(piL&z=SW1&z,vin,Vo,Vdidrop&z,iL,iout,Vswdrop&z,Lbuck&z,&
                        Rlesr&z,Rcesr&z)

                "----if SW1 gated, assume can conduct pos. current"
                IF (SW1&z) THEN
                        piL&z = (vin-Vswdrop&z-Vo)/Lbuck&z - &
                                (Rlesr&z+Rcesr&z)*iL/Lbuck&z + &
                                Rcesr&z*iout/Lbuck&z
                ELSE
                        IF (iL .GT. 0.0) THEN

                                !"if IGBT not gated but iL greater than zero  "
                                !"the diode conducts and the inductor discharges"
                                piL&z = (-Vdidrop&z-Vo)/Lbuck&z - &
                                        (Rlesr&z+Rcesr&z)*iL/Lbuck&z + &
                                        Rcesr&z*iout/Lbuck&z
                        ELSE

                                !"if IGBT not gated and iL less than or equal"
                                !"to zero, the diode is off and the inductor    "
                                !"current and derivative are zero               "
                                piL&z = 0.0

                        ENDIF  !"end if (iL .GT. 0.0)"

                ENDIF  !"end if (SW1&z)"

        END             !"of procedural"

        !"integrate the inductor state variable"
        iL = BOUND(0.0, 100.0, LIMINT(piL&z, iLic&z, 0.0, 100.0))

MACRO END
```

3. Source Buck Converter Input Filter Macro

```
!*****************************************************************************
! sourcefilter.mac
!
! This macro governs the operation of the source buck converter input LC filters. The
! model takes into account equivalent series resistance of the capacitor and inductor in
! the filter. Due to algebraic loop considerations within the controller and buck
! macros, an equivalent parallel capacitance and resistance was found for the series
! capacitance/esr combination. This result gives the seemingly large value of capacitor
! esr shown in the call to this macro in mode7.csl.
!
! Written by: Michael Badorf
! Last modified: 02JUN97
!*****************************************************************************
MACRO sourcefilter(z,vin,Vcap,iLfilt,isw,pRlesr_in,pLfilt,pRcesr_in,&
                   pCin,pVcapic)

INITIAL

        CONSTANT pRlesr_in
        CONSTANT pLfilt
        CONSTANT pRcesr_in
        CONSTANT pCin
        CONSTANT pVcapic

        "----assign the inductor current IC"
        CONSTANT iLfiltic&z = 0.0

END     !"of initial"

        !"buck converter input filter state equations "
        piLfilt&z = (vin-Rlesr_in&z*iLfilt-Vcap)/Lfilt&z

        !"current through input inductor     "
        iLfilt  = INTEG(piLfilt&z, iLfiltic&z)

        pVcap&z = (iLfilt - isw - Vcap/Rcesr_in&z)/Cin&z

        !"input voltage at buck converter IGBT  "
        Vcap  = INTEG(pVcap&z, Vcapic&z)

MACRO END
```

4. Load Buck Converter Input Filter Macro

```
!*******************************************************************************
! loadfilter.mac
!
! This macro governs the operation of the load buck converter input LC filter.  The
! model takes into account equivalent series resistance of the capacitor and inductor in
! the filter.  In contrast to the source filter, no algebraic loops were formed and the
! series capacitance/esr could be modeled directly.
!
! Written by: Michael Badorf
! Last modified: 02JUN97
!*******************************************************************************
MACRO loadfilter(z,vin,Vcap,iLfilt,isw,pRlesr_in,pLfilt,pRcesr_in,&
                 pCin,pVcapic)

INITIAL

        CONSTANT pRlesr_in
        CONSTANT pLfilt
        CONSTANT pRcesr_in
        CONSTANT pCin
        CONSTANT pVcapic

        "----assign the initial inductor current and cap volt"
        CONSTANT iLfiltic&z = 0.0

END     !"of initial"

        !"buck converter input filter state equations  "
        piLfilt&z = (vin-Rlesr_in&z*iLfilt-Vcap)/Lfilt&z

        !"current through input inductor                "
        iLfilt  = INTEG(piLfilt&z, iLfiltic&z)

        pV&z = (iLfilt - isw)/Cin&z
        V&z  = INTEG(pV&z, Vcapic&z)

        !"input voltage at buck converter IGBT  "
        Vcap = V&z + (iLfilt-isw)*Rcesr_in&z

MACRO END
```

5. Command File

```
!*****************************************************************************
! mode7.cmd
!
! This runtime command file exercises the simulation by executing load changes. In this
! way the transient response of the PEBB network can be analyzed. The file contains
! studies that allow system start up. The final values of the startups are used as ICs
! for the transient analysis. In this way, the transient analysis can be run over and
! over without having to execute a startup every time.
!
! Written by: Michael Badorf
! Last modified: 02JUN97
!*****************************************************************************
s strplt = .t.                  ! "one variable per x-axis"
s calplt = .f.

s devplt = 5                    ! "6 for X-windows"
                                ! "5 for postscript"

s ppoplt = .f.                  ! "true rotates plot 90 deg"
s xinspl = 6                    ! "x-axis plot units"
s weditg = .f.                  ! "false suppresses data write"
                                ! "each time SCHEDULE occurs"

s nrwitg = .t.                  ! "true enables accumulation of data"
                                ! "after a CONTIN"

s dpnplt = .f.                  ! "no date"

!"determines which variables to plot"
prepare t,iout,iout1,iout2,iout3,Vout,Vout3,iLa,Vas

!"startup of source buck converters"
proced startup

        !"source buck startup parameters"
        s Rout = 40.0

        !"load buck parameters"
        s Rout3 = 144.0
        s buck1300 = .false.        !"loadbuck off"
        s tcount = 10.0

        !"inverter parameters
        s R4 = 5.0
        s A = 0.0                   !"inverter off"
```

```
        !"simulation parameters"
        s ialg = 5
        s maxt = 1.0e-6
        s cint = 1.0e-3
        s tstop = 0.5
        start

        s cint = 1.0e-6
        s tstop = 0.75
        contin

        SAVE /FILE='startup' count,Vthouse_old,Verr_old,Vint_old,&
              Vout,iout,Vincap3,Vout3,iout3,Vag,Vbg,Vcg,i1,i2,i3,&
             /BINARY /STATES /ICS /EVENTS

end

!"startup of load buck converter"
proced loadbuck

        RESTORE /FILE='startup' /EVENTS
        s reinitf = .true.
        REINIT

        !"source buck startup parameters"
        s Rout = 40.0

        !"load buck parameters"
        s Rout3 = 144.0
        s buck1300 = .true.                !"loadbuck on"
        s tcount = 1.0e-7

        !"inverter parameters
        s R4 = 5.0
        s A = 0.0                          !"inverter off"

        !"simulation parameters"
        s ialg = 5
        s maxt = 1.0e-6
        s cint = 1.0e-6
        s tstop = 0.4
        start

        SAVE /FILE='loadbuck' count,Vthouse_old,Verr_old,Vint_old,&
              Vout,iout,Vincap3,Vout3,iout3,Vag,Vbg,Vcg,i1,i2,i3,&
             /BINARY /STATES /ICS /EVENTS

end
```

```
!"startup of inverter"
proced inverter

        RESTORE /FILE='loadbuck' /EVENTS
        s reinitf = .true.
        REINIT

        !"source buck startup parameters"
        s Rout = 40.0

        !"load buck parameters"
        s Rout3 = 144.0
        s buck1300 = .true.                !"loadbuck on"
        s tcount = 1.0e-7

        !"inverter parameters
        s R4 = 5.0
        s A = 10.0                         !"inverter on"

        !"simulation parameters"
        s ialg = 5
        s maxt = 1.0e-6
        s cint = 1.0e-6
        s tstop = 0.4
        start

        SAVE /FILE='inverter' count,Vthouse_old,Verr_old,Vint_old,&
              Vout,iout,Vincap3,Vout3,iout3,Vag,Vbg,Vcg,i1,i2,i3,&
             /BINARY /STATES /ICS /EVENTS

end

!"parallel source buck converters supplying load buck and inverter"
proced trans

        !"initial conditions restoration from 760 uH source    "
        !"buck converter / load buck converter startup         "
        RESTORE /FILE='inverter' /EVENTS
        REINIT
        s reinitf = .true.

        !"source buck parameters"
        s Rout = 96.8

        !"load buck parameters"
        s Rout3 = 144.0
        s buck1300 = .true.
        s tcount = 1.0e-7

        !"inverter parameters"
        s R4 = 5.0
        s A = 10.0
```

```
!"simulation parameters"
s nrwitg = .true.
s ialg = 5
s maxt = 1.0e-6
s cint = 1.0e-6
s tstop = 0.04
start

!"10% -> 100% load buck converter load change"
!"3 kW 100% power reference                "
s Rout3 = 14.4
s tstop = 0.08
contin

!"100% -> 10% load buck converter load change"
s Rout3 = 144.0
s tstop   = 0.12
contin

!"40% -> 100% inverter load change"
s A = 24.0
s tstop = 0.17
contin

!"100% -> 40% inverter load change"
s A = 10.0
s tstop   = 0.2
contin

s plt=1
plot /color=0 /XTAG='(sec)'                              &
        iout3 /TAG='(amps)'  /lo=0.0 /hi = 16.0          &
        Vout3  /TAG='(volts)' /lo=196.0 /hi=218.0        &
        iout  /TAG='(amps)'  /lo=-5.0 /hi=30.0           &
        Vout  /TAG='(volts)' /lo=300.0 /hi=308.0

s plt=2
plot /color=0 /XTAG='(sec)'                              &
        iout2 /TAG='(amps)'  /lo=-5.0 /hi16.0            &
        iout1 /TAG='(amps)'  /lo=-5.0 /hi=16.0           &
        iLa   /TAG='(amps)'  /lo=-24.0 /hi=24.0          &
        Vas   /TAG='(volts)' /lo=-210.0 /hi=210.0

end
```

APPENDIX D. EASYTRAX PCB DATA

A. ANALOG CONTROLLER PCB DATA

This section is divided into a component parts listing, an EASYTRAX PCB netlist, and a PCB component overlay.

1. Component List

R411G
AXIAL0.4

C603
RAD0.2

R603
AXIAL0.4

R603H
AXIAL0.4

R411
AXIAL0.4

R607
AXIAL0.4

C902
RADO0.2

C904
RB.3/.6

C905
RB.3/.6

C907
RB.2.4

C901
RB.2.4

R112
AXIAL0.4

R807Q
AXIAL0.4

R807H
AXIAL0.4

R606
AXIAL0.4

R803
AXIAL0.4

R803G
AXIAL0.4

C107
RADO.2

D107
DIODE0.4

D106
DIODE0.4

R112G
AXIAL0.4

R114
AXIAL0.4

R108
AXIAL0.4

R110G
AXIAL0.4

R109
AXIAL0.4

R102B
AXIAL0.4

R113A
AXIAL0.4

R113C
AXIAL0.4

R106
AXIAL0.4

R313
AXIAL0.4

R102C
AXIAL0.4

R101
AXIAL0.4

R113B
AXIAL0.4

R110
AXIAL0.4

R102A
AXIAL0.4

C203
RADO.2

C210
RADO.2

C212
RADO0.2

C205
RADO.2

R024J
AXIAL0.4

R205
AXIAL0.4

R210
AXIAL0.4

R212
AXIAL0.4

C904D
RADO.2

C305
RADO.2

C804
RADO.2

C808
RADO.2

C514
RADO.2

C708
RADO.2

C704
RADO.2

C306
RADO.2

C111
RADO.2

C104
RADO.2

C211
RADO.2

C204
RADO.2

C608
RADO.2
C414
RADO.2

C408
RADO.2

C604
RADO.2

D901A
DIODE0.4

D902A
DIODE0.4

D901B
DIODE0.4

D902B
DIODE0.4

R902
AXIAL0.4

R907
AXIAL0.4

D907
RADO.1

R313G
AXIAL0.4

R704
AXIAL0.4

R702
AXIAL0.4

R316B
AXIAL0.4

R316A
AXIAL0.4

R318
AXIAL0.4

R301
AXIAL0.4

R311
AXIAL0.4

C313
RADO.2

C301
RADO.2

C311
RADO.2

C302
RADO.2
C317
RADO.2

C702
RAD0.2

D311
DIODE0.4

D301
DIODE0.4

D317
DIODE0.4

R501
AXIAL0.4

R509
AXIAL0.4

R504
AXIAL0.4

R807
AXIAL0.4

R505
AXIAL0.4

R502
AXIAL0.4

C807
RADO.2

C501
RADO.2

C803
RADO.2

C316
RADO.2

R603V
VR1

J25
DB25RA/F

C907G
RADO.2
U1
DIP14
LF347

U2
DIP12
LM324

U3
DIP18
UC3637

U4
DIP14
AD534

U5
DIP14
7432

U6
DIP8
1458

U7
DIP8
LM311

U8
DIP8
LM311

U9
DATEL
TWR-5/1000-
15/200-D12

Q1
TO-92A
2N2222

2. Net List

NET1
R411G-2
R411-1
U4-11

NET2
C603-2
U6-3
R603-2
Q1-3

NET3
U6-1
U6-2
R606-2

NET4 (-15V)
U6-4
C905-2
C904D-1
C305-2
C804-2
C704-2
C111-2
C211-2
C408-1
C604-2
U3-5
U7-4
U8-4
U2-11
U4-8
U9-5
U1-11
R704-1
R316A-2
R318-2
C316-2

NET5
U6-6
R607-2
R606-1

NET6 (Vref)
U6-7
R607-1
R102B-1
U4-10

NET7 (+15V)
U6-8
R603H-1
C904-1
R807H-2
R803-1
R313-1
C904D-2
C808-1
C708-2
C306-2
C104-1
C204-1
C608-1
C414-1
U3-6
U7-8
U8-8
U2-4
U4-14
U9-4
U1-4
R301-1
C807-1

NET8
R603-1
R603V-2

NET9
R603H-2
R603V-1

NET10 (io/10)
R411-2
R109-1
R102A-1
U2-13
U2-14
R316B-2

NET11
C902-2
C901-2
U9-2
D902A-A
D902B-A
R902-1

NET12
C902-1
C901-1
U9-1
D901A-K
D901B-K

NET13 (+5V)
C907-1
C514-1
U5-14
U9-7
J25-4
R907-2
R504-2
C501-2
C907G-1

NET14
R112-1
R106-2
R101-2
U1-1

NET15
R112-2
R112G-1
U1-12

NET16
R807Q-1
U8-7
R807-2

NET17
R807Q-2
Q1-2

NET18
R807H-1
U3-14
R807-1
C807-2

NET19
R803-2
R803G-2
U8-3
C803-2

NET20
C107-2
D107-A
R113C-2
U1-7

NET21
C107-1
D106-A
R106-1
U1-6

NET22
D107-K
D106-K

NET23
R114-1
R113A-1
R113C-1
R113B-1
U1-13

NET24
R114-2
U1-14
R311-1

NET25
R108-1
R113B-2
U1-8

NET26
R108-2
R109-2
U1-9

NET27
R110G-1
R110-2

U1-10

NET28
R102B-2
R102C-2
R101-1
R102A-2
U1-2

NET29
R113A-2
U4-7
U4-12

NET30
R313-2
U3-13
R313G-1
C313-1

NET31 (vo/50)
R102C-1
U2-1
U2-2

NET32 (iL/10)
R110-1
U3-12
U2-8
U2-9
NET33
C203-2
R203-2
U2-3

NET34
C210-2
R210-2
U2-10

NET35
C212-2
R212-2

U2-12

NET36
C205-1
R205-2
U2-5

NET37
R024J-1
J25-24

NET38
R203-1
J25-9

NET39
R205-1
J25-8

NET40
R210-1
J25-6

NET41
R212-1
J25-7

NET42
U3-1
R301-2
C301-2
D301-K

NET43
U3-2
U3-8
U3-10
C302-2

NET44
U3-4
J25-13

NET45
U3-9
U3-11
R311-2
C311-2
D311-K

NET46
U3-16
R316B-1
R316A-1
C317-1
D317-K

NET47
U3-17
U7-3
C317-2
D317-A

NET48
U3-18
R318-1

NET49
U7-2
R704-2
R702-2
C702-2

NET50
U7-7
U5-4
R504-1

NET51
U8-2
U5-11

NET52 (vin/50)
U2-6
U2-7
U4-1

NET53
U5-1
J25-3
R501-1
C501-1

NET54
U5-2
J25-2
R502-2

NET55
U5-3
U5-13
J25-1

NET56
U5-5
J25-14
R505-2

NET57
U5-6
U5-12
J25-15

NET58
U5-9
U5-10
R509-1

NET59
D901A-A
D902A-K

NET60
D901B-A
D902B-K

NET61
R907-1
D907-2

3. PCB Component Overlay

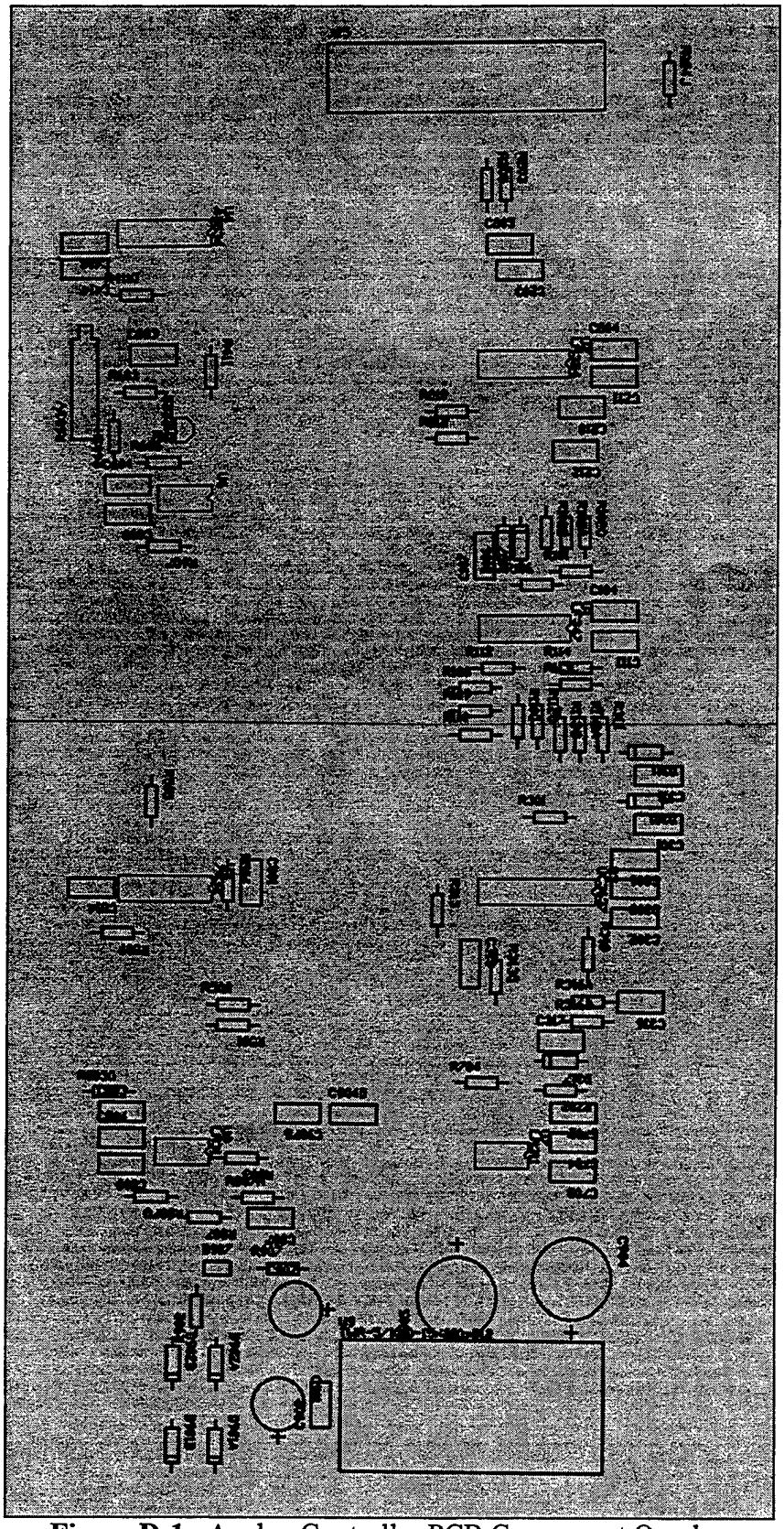


Figure D-1, Analog Controller PCB Component Overlay

B. PHASE-LOCKED LOOP PCB DATA

This section is divided into a component parts listing, an EASYTRAX PCB netlist, and a PCB component overlay.

1. Component List

C202
RAD0.2

C102
RAD0.2

R202
AXIAL0.4

R102
AXIAL0.4

R107
AXIAL0.4

R207
AXIAL0.4

C103
RAD0.2

C203
RAD0.2

C204
RAD0.2

C104
RAD0.2

C108
RAD0.3

C308
RAD0.3

C208
RAD0.3

JUMPER
RAD0.2

R311X
AXIAL0.4

R311
AXIAL0.4

T-15
RAD0.2

T+15
RAD0.2

TI_SET
RAD0.2

TSLAVE
RAD0.2

TMASTER
RAD0.2

U2-LM311
DIP8

U1-LM311
DIP8

U3-4011
DIP14

R103
AXIAL0.4

R203
AXIAL0.4

R108
AXIAL0.4

C311
RAD0.2

R208
AXIAL0.4

2. Net List

NET1
C202-1
R202-2
U2-LM311-2

NET2
C102-1
R102-2
U1-LM311-2

NET3
R202-1
TSLAVE-2

NET4
R102-1
TMASTER-2

NET5
R107-1
R207-1
C108-2
C314-2
C208-2

T+15-2
U2-LM311-8
U1-LM311-8
U3-4011-14
R108-2
R208-1

NET6
R107-2
U1-LM311-7
U3-4011-13

NET7
R207-2
U2-LM311-7
U3-4011-8

NET8
C103-1
U1-LM311-3
R103-2
R108-1

NET9
C203-1

U2-LM311-3
R203-2
R208-2

NET10
C204-2
C104-2
T-15-1
U2-LM311-4

U1-LM311-4
NET11
JUMPER-2
U3-4011-10

NET12
JUMPER-1
U3-4011-12

NET13
R311X-1
TI_SET-1

NET14
R311X-2
R311-1
C311-1

NET15
R311-2
U3-4011-9
U3-4011-11

3. PCB Component Overlay

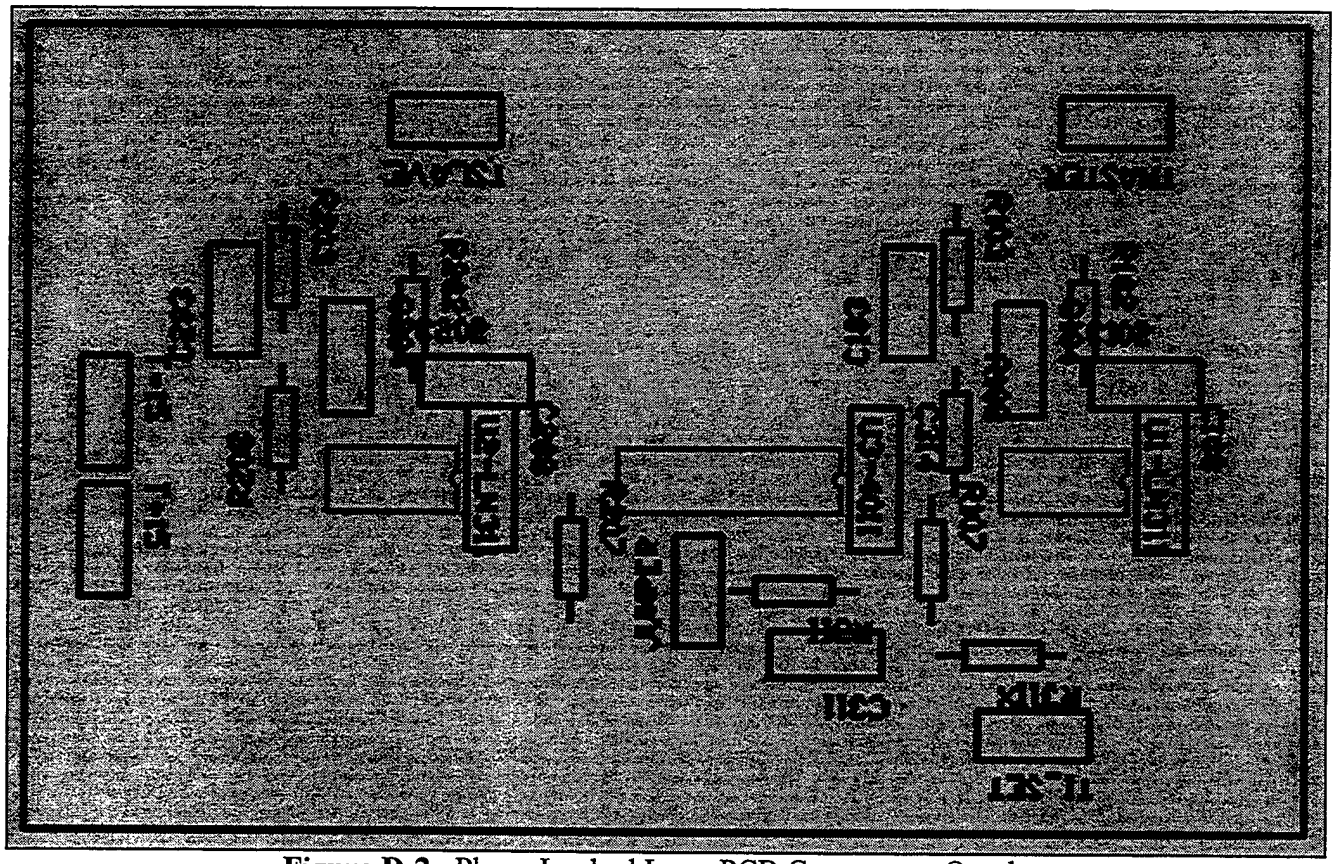


Figure D-2, Phase-Locked Loop PCB Component Overlay

C. IGBT DRIVER CIRCUIT PCB DATA

This section is divided into a component parts listing, an EASYTRAX PCB netlist, and a PCB component overlay.

1. Component List

C104,C108
2200uF-50V
RB.3/.6

C148
0.22uF
RAD0.3

C203
0.1uF
RAD0.2

D108A,D104A,
D108B,D104B
1N4005
DIODE0.4

D123
1N4148 or 1N914
DIODE0.4

DGATE2,DGATE1
1N4744-Zener
DIODE0.4

JUMP
-jumper with opto
-input if no opto
RAD0.1

R106
6.2 - 1/4W
AXIAL0.4

R107
1k-1/4W
AXIAL0.4

R201
-jumper with opto
-1.3k-1/4W if no opto
AXIAL0.4

R213
1k-1/4W
AXIAL0.4

U1
TLP250 (Toshiba)
DIP8

U2
HFBR-2521 (HP)
HFBR 15/25XX

VR1
LM78L05ACZ
TO-92A

2. Net List

NET1
JUMP-1
U1-3
D123-A

NET2
R213-1
U2-3
VR1-2
C203-1

NET3
R213-2
U2-4
U2-1

R201-2

NET4
C148-2
C104-2
D104A-A
D104B-A
U1-5

NET5
C148-1
C108-1
D108A-K
D108B-K
U1-8

VR1-1

NET6
D108A-A
D104A-K

NET7
D108B-A
D104B-K

NET8
U1-7
U1-6
R106-2

NET9
U1-2
R201-1
D123-K

NET10
R106-1
DGATE2-A
R107-2

NET11
DGATE2-K
DGATE1-K

3. PCB Component Overlay

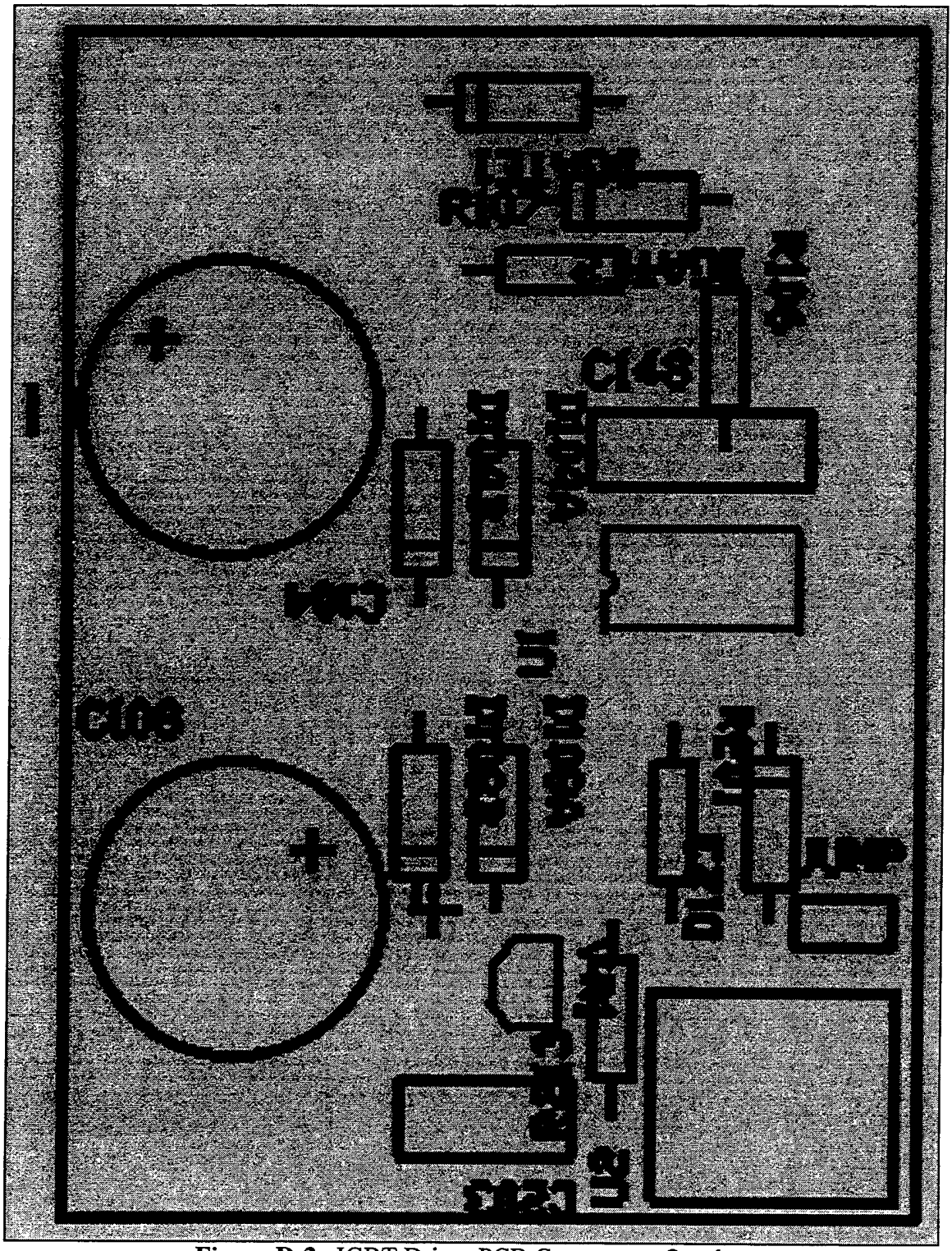


Figure D-3, IGBT Driver PCB Component Overlay

D. SENSOR BOARD PCB DATA

This section is divided into a component parts listing, an EASYTRAX PCB netlist, and a PCB component overlay.

1. Component List

R401
AXIAL0.4

R201
AXIAL0.4

R301
AXIAL0.4

R101
AXIAL0.4

C602
RAD0.2

C601
RAD0.2

C502
RAD0.2

C501
RAD0.2

C801
RAD0.2

C802
RAD0.2

C701
RAD0.2

C702
RAD0.2

J4
DIP6

J3
DIP6

U8
CL50
CL50

U7
CL50
CL50

U6
CL50
CL50

U5
CL50
CL50

VR1
TO-220
TI7815

VR2
TO-220
MC7915

DVR22B
DIODE0.4

DVR11B
DIODE0.4

DVR22A
DIODE0.4

DVR11A
DIODE0.4

CVR11
RB.3/.6

CVR22
RB.3/.6

U1
AD215
AD215

U3
AD215
AD215

U2
AD215
AD215

U4
AD215
AD215

J5
DIP6

J6
DIP6

R703
AXIAL0.4

R503
AXIAL0.4

R803
AXIAL0.4

R603
AXIAL0.4

C144
RAD0.2

C344
RAD0.2

C244
RAD0.2

C444
RAD0.2

C242
RAD0.2

C442
RAD0.2

C142
RAD0.2

C342
RAD0.2

2. Net List

NET1
R401-1
J4-4

NET2
R401-2
U4-1

NET3
R201-1
J4-2

NET4
R201-2
U2-1

NET5
R301-1
U3-1

NET6
R301-2
J3-4

NET7
R101-1
J3-1

NET8
R101-2
U1-1

NET9 (-15 V)
C602-1
C502-1
C802-1
C702-1
U8-2
U7-2
U6-2
U5-2
VR2-3
U1-44
U3-44
U2-44
U4-44
C144-1
C344-1
C244-1
C444-1

NET10 (+15 V)
C601-1
C501-1
C801-1
C701-1
U8-1
U7-1
U6-1
U5-1
VR1-3
U1-42
U3-42
U2-42
U4-42
C242-1
C442-1
C142-1
C342-1

NET11
J4-3
U2-2

NET12
J4-5
U4-2

NET13
J3-2
U1-2

NET14
J3-5
U3-2

NET15
U8-3
J6-6
R803-1

NET16
U7-3
J5-6
R703-1

NET17
U6-3
J6-1
R603-1

NET18
U5-3
J5-3
R503-1

NET19
VR1-1
DVR11B-K
DVR11A-K
CVR11-1

NET20
VR2-2
DVR22B-A
DVR22A-A
CVR22-2

NET21
DVR22B-K
DVR11B-A

NET22
DVR22A-K
DVR11A-A

NET23
U1-38
J1-3

NET24
U1-4
U1-3

NET25
U3-38
J1-4

NET26
U3-4
U3-3

NET27
U2-38
J2-3

NET28
U2-4
U2-3

NET29
U4-38
J2-4

NET30
U4-4
U4-3

3. **PCB Component Overlay**

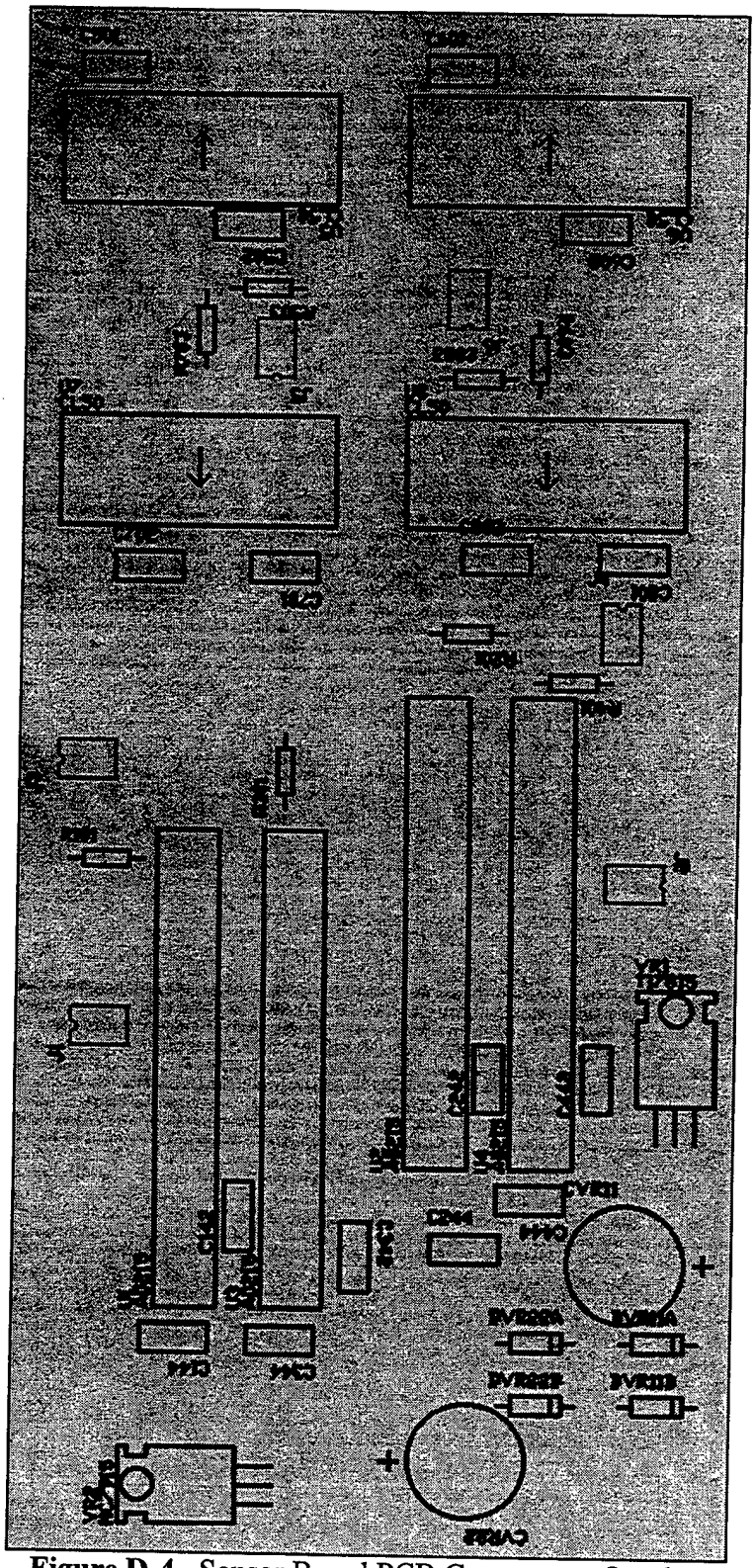


Figure D-4, Sensor Board PCB Component Overlay

LIST OF REFERENCES

1. Blalock, H.C., "Power Electronic Converter Simulation, Real Time Control and Hardware-In-The-Loop Testing for a Shipboard Integrated Power System," Electrical Engineer Thesis, Naval Postgraduate School, Monterey, CA, March, 1995.

2. Mohan, N., Undeland, T.M., Robbins, W.P., *Power Electronics, Converters, Applications and Design*, John Wiley & Sons, Inc., New York, NY, 1989.

3. Mayer, J. and Salberta, F., "High-Frequency Power Electronic Converter for Propulsion Applications," Electrical Engineer Thesis, Penn State University, University Park, PA, January, 1997.

4. Arnold, *PC-104K MPP Cores*, The Arnold Engineering Company, Marengo, IL, September, 1995.

5. Kassakian, J.G., Schlecht, M.F., and Verghese, G.C., *Principles of Power Electronics*, Addison-Wesley Publishing Company, New York, NY, 1991.

6. Ciezki, J.G., "EC2320 Linear Control Systems Class Notes," Naval Postgraduate School, Monterey, CA, October, 1997.

7. Hanson, R.J., "Implementing Closed-Loop Control Algorithms for DC-to-DC Converters and ARCP Inverters Using the Universal Controller," Electrical Engineer Thesis, Naval Postgraduate School, Monterey, CA, June, 1997.

8. Oberley, M.J., "The Operation and Interaction of the Auxiliary Resonant Commutated Pole Converter in a Shipboard DC Power Distribution Network," Electrical Engineer Thesis, Naval Postgraduate School, Monterey, CA, December, 1996.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center.....2
   8725 John J. Kingman Rd., STE 0944
   Fort Belvoir, Virginia 22060-6218

2. Dudley Knox Library.....2
   Naval Postgraduate School
   411 Dyer Rd.
   Monterey, California 93943-5101

3. Chairman, Code EC.....1
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, California 93943-5121

4. Robert Ashton, Code EC/Ah.....3
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, California 93943-5121

5. John Ciezki, Code EC/Cy.....3
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, California 93943-5121

6. Roger Cooley.....2
   Naval Surface Warfare Center
   Carderock Division/Annapolis Detachment
   3A Leggett Circle
   Annapolis, MD 21402-5067

7. Michael G. Badorf.....2
   c/o W.P. Hearn
   R.R. 6 Box 203
   Georgetown, DE 19947