High Power Density Drivetrain Integrated Electric Vehicle Charger

Usama Anwar
University of Colorado at Boulder, usama.anwar@colorado.edu

Follow this and additional works at: https://scholar.colorado.edu/ecen_gradetds

Part of the Power and Energy Commons

Recommended Citation
https://scholar.colorado.edu/ecen_gradetds/150

This Thesis is brought to you for free and open access by Electrical, Computer & Energy Engineering at CU Scholar. It has been accepted for inclusion in Electrical, Computer & Energy Engineering Graduate Theses & Dissertations by an authorized administrator of CU Scholar. For more information, please contact cuscholaradmin@colorado.edu.
High Power Density Drivetrain Integrated Electric Vehicle Charger

by

Usama Anwar

B.Sc., Lahore University of Management Sciences (LUMS), 2014

A thesis submitted to the
Faculty of the Graduate School of the
University of Colorado in partial fulfillment
of the requirements for the degree of
Masters of Science
Department of Electrical, Computer, and Energy Engineering

2016
This thesis entitled:
High Power Density Drivetrain Integrated
Electric Vehicle Charger
written by Usama Anwar
has been approved for the Department of Electrical, Computer, and Energy Engineering

______________________________
Prof. Khurram Afridi

______________________________
Prof. Dragan Maksimovic

Date ______________

The final copy of this thesis has been examined by the signatories, and we find that both the content and the form meet acceptable presentation standards of scholarly work in the above mentioned discipline.
Abstract

This thesis presents a new architecture for an isolated level 2 on-board electric vehicle (EV) battery charger which is integrated with the EV’s drivetrain. The integration with the drivetrain is done by leveraging many of the existing stages of a highly efficient and power-dense composite-architecture-based drivetrain boost converter that acts as an interface between the traction battery and the motor drive and allows the input voltage of the motor drive to be adjusted dynamically to maximize drivetrain efficiency. This composite boost converter comprises a buck stage, a boost stage and a dc transformer (DCX) stage, providing multiple means to integrate on-board charger functionality. In this thesis four alternative approaches to drivetrain integration are identified, analyzed and compared quantitatively in terms of added weight and charging losses. Out of the considered approaches, the selected charging architecture provides an effective tradeoff between added weight and charging losses. This drivetrain-integrated charger leverages the buck stage and part of the DCX stage of the composite boost converter, and adds only a bridgeless-boost based power factor correction (PFC) ac-dc stage, an H-bridge and a single winding to the composite boost converter, to achieve high-power on-board charging functionality without substantial additional weight. Hence, the proposed charger architecture comprises a boost PFC rectification stage, a dual-active bridge isolation stage and a boost current regulation stage.

This thesis also presents detailed power stage and controller design of the boost converter based PFC ac-dc stage and introduces a generalized methodology for developing hybrid feedforward control for power converters. Two PFC bridgeless-boost implementations are presented: the first utilizes traditional feedback control, while the second utilizes hybrid feedforward control. Zero
crossing distortion effects observed in PFC converters are studied and methods to mitigate these effects are developed and implemented. In case of traditional feedback control, it is shown that the new control, introduced in this thesis, mitigates zero crossing distortion and allows the PFC converter to achieve near-ideal PFC rectifier performance. In the case of hybrid feedforward control architecture, good boost PFC rectifier performance is also demonstrated. The thesis also presents control architectures for the dual-active bridge stage acting as an isolation and voltage regulation stage, and the boost stage acting as a power regulation stage.

Finally, two 6.6 kW prototypes of the proposed charger’s PFC stage have been designed, fabricated and tested. One prototype utilizes Silicon devices, while the other utilizes Silicon Carbide devices. It is shown that the proposed charger architecture’s PFC stage achieves greater than 97% peak efficiency and the on-board charging functionality can be added with only 34% increase in the weight of the composite boost converter.
Dedication

To my parents, my siblings and my guide
Acknowledgements

The author would like to acknowledge the assistance, guidance, and support from my supervisor Professor Khurram Afridi who helped me throughout my research at the university and made it possible for me to explore my interests under his supervision. His advise has been very helpful for me. Beyond research, I have learnt invaluable life lessons from him. I would also like to thank my co-supervisors Professor Dragan Maksimovic and Professor Robert Erickson for guiding me for my years at the university. The guidance of my supervisors and their encouragement throughout my research has pushed me to pursue my ideas and explore them deeply. Through them, I have learnt to develop fundamental approaches to understanding ideas. My special thanks to Professor Dragan Maksimovic, whose small bits of insights contain profound knowledge and has led me to develop some of the ideas discussed in this thesis. My research is and will be inspired by what I learnt at CoPEC and for this I am grateful.

I would like to acknowledge the dedication of Professor Asad Ali Abidi (UCLA) in making an effort to uplift the engineering education in Pakistan and delivering workshops in fundamental areas of electrical engineering. These workshops helped me develop insights in the area of control systems, from which some of this work benefits. I had a chance to learn from him, firstly in my sophomore year of undergraduate which exposed me to my excitement in doing science. This journey is still on its way and hopefully will continue!

I would also like to acknowledge my undergraduate adviser and a good friend, Sir Nauman Zaffar (LUMS). He has helped me throughout my later years of undergraduate studies and has remained consistent in being a primary motivator to pursue my studies. I think, he too, like me
finds excitement in doing science and teaching and I have learnt greatly from him. I feel indebted with his greatness in always being there to listen to my problems and in guiding me, among many other students that keep him busy all the time.

The ‘real hard work’ required to demonstrate the research work presented in this thesis was made possible with the help of graduate students, who helped out with the hardware implementation and debugging of circuits. In this regard, I would like to thank Dr. Hyeokojin Kim, Dr. Hua Chen, Sreyam Sinha and Dr. David Jones for taking out time to discuss ideas and help me in debugging circuits.

My special thanks to my parents, Anwaar ul Haq and Samina Anwaar and my siblings, Anum Anwaar and Talha Anwaar for their continued support and it is through them that I have passed difficult phases of my Masters. Lastly I would like to express my deep gratitude to Mr. Asif Mehmood, whose words of wisdom are for me like pearls in an ocean. He has helped me find my direction in life.
## Contents

### Chapter 1

1 Introduction

1.1 Motivation

1.2 Drivetrain Architecture and Charger Integration Approach

1.3 Thesis Contributions

1.4 Thesis Organization

### Chapter 2

2 Alternative Drivetrain-Integrated Electric Vehicle Charger Architectures

2.1 DC-DC Composite Boost Converter Architecture

2.2 Charger Integration Approaches

2.3 Quantitative Comparison of the Alternative Drivetrain-Integrated Charger Architectures

### Chapter 3

3 Drivetrain-Integrated Charger Design and Control

3.1 Selected Drivetrain-Integrated Electric Vehicle Charger Architecture

3.2 AC-DC Bridgeless Boost Converter

3.2.1 Bridgeless Boost Converter Operating in DCM

3.2.2 Bridgeless Boost Converter Operating in CCM

3.3 Dual-Active Bridge Converter

3.4 Power Regulation Boost Converter
# Zero Crossing Distortion in Power Factor Correction Rectifiers

4.1 AC-DC Bridgeless Boost Converter Operating in DCM ............................... 18

4.2 AC-DC Bridgeless Boost Converter Operating in CCM ............................... 21

4.2.1 Traditional Control Architecture for Single-Phase AC-DC Converter .......... 22

4.2.2 Control Architecture that Mitigates Zero-Crossing Distortion in Boost PFC Rectifiers ................................................................. 24

4.2.3 Phase Shift due to Input Voltage Feedforward Effect ......................... 26

# Prototype Design and Experimental Results

5.1 AC-DC Bridgeless Boost Converter Utilizing Si Devices and Operating in DCM ... 30

5.2 AC-DC Bridgeless Boost Converter Utilizing SiC Devices Operating in CCM .... 33

5.3 DAB Simulation Waveforms ................................................................. 35

5.4 Boost Converter Simulation Waveforms ............................................... 37

# Conclusions

6.1 Summary and Conclusions ................................................................. 38

6.2 Recommendations for Future Work .................................................... 40

# Bibliography

42

# Appendix

A Generalized Hybrid Feedforward Control of Pulse-Width Modulated Switching Converters 44

A.1 Introduction ....................................................................................... 44

A.2 Example of Hybrid Feedforward Control Architecture .......................... 46

A.2.1 Feedforward Output Voltage Regulation in CCM Boost Converters ....... 46

A.2.2 Hybrid Feedforward Control of DCM Boost PFC Rectifiers ............... 47

A.3 General Considerations for Synthesis of Hybrid Feedforward Control ....... 50
List of Tables

Table

2.1 Comparison of proposed integrated charger architectures in terms of added weight and full-power losses for a 6.6 kW on-board EV charger. ............................ 9

5.1 Design Details of the 6.6kW Bridgeless Boost PFC Stage Utilizing Silicon Devices and Operating in Discontinuous Conduction Mode (DCM) .......................... 31

5.2 Design Details of the 6.6 kW Bridgeless Boost PFC Stage Utilizing Silicon Carbide Devices and Operating in Discontinuous Conduction Mode (DCM). .............. 34

A.1 Four switch buck boost converter component parameters. ............................... 56

C.1 Microcontroller ePWM Channel Designation ..................................................... 67
List of Figures

Figure

1.1 Architecture of the dc-dc composite boost converter used in an electric vehicle drive-train between the vehicle battery and the motor drive [1]. The converter is comprised of multiple power modules: a buck module, a boost module and a dc transformer (DCX) module. ................................................................. 2

2.1 Architecture of the dc-dc composite boost converter used in an electric vehicle drive-train between the vehicle battery and the motor drive [1]. The converter is comprised of multiple power modules: a buck module, a boost module and a dc transformer (DCX) module. ................................................................. 6

2.2 Four proposed architectures suitable for integration of charging functionality with the composite boost converter of Fig. 2.1. ................................................................. 7

3.1 Drivetrain integrated charger architecture comprising a bridgeless boost converter, followed by a dual active bridge (DAB) and a buck converter. ................................. 11

3.2 Power factor correction functionality (PFC) implemented using combined hybrid feedforward-feedback control. ................................................................. 12

3.3 Control architecture for bridgeless boost converter. .................................................. 14

3.4 Boost converter acting as a power regulation stage to regulate the battery power: (a) Boost converter topology (b) Large signal model and (c) Controller implemented to achieve input power sink characteristics. .................................................. 16
4.1 Power factor correction functionality (PFC) implemented using combined hybrid feedforward-feedback control. ................................................................. 18
4.2 Sensor error in sensing input voltage zero crossing: (a) Actual zero-crossing; (b) zero crossing sensed when the input voltage sensor detects the zero crossing before it actually happens; (c) zero crossing sensed when the input voltage sensor detects the zero crossing after it actually happens; and (d) proposed solution of overlapping the switching time of the two half-bridge legs around input-voltage zero crossing solves the problem. ................................................................. 19
4.3 Bridgeless Boost PFC rectifier. ................................................................. 21
4.4 Traditional ac-dc conversion stage. Rectified input ac voltage $v_{in}$ is used to generate the input current reference $i_{ref1}$. ................................................................. 23
4.5 Input current reference for bridgeless boost converter topology: (a) Use of rectified input ac voltage to generate the current reference introduces high frequency content in the reference command. (b) Direct use of input ac voltage to generate the current reference command removes high frequency content from the reference command. ... 23
4.6 Proposed control architecture for bridgeless boost converter. ................. 24
4.7 Half bridges transition block internal operation. ......................................... 24
4.8 Current reference, control command $v_c$ and duty cycles $d_1$ and $d_2$ plotted over half line cycle. ................................................................. 25
4.9 Proposed control architecture for bridgeless boost converter. ................ 26
4.10 Phase difference between input voltage and average input current. The phase difference is 25° which matches theoretically predicted phase difference. ............... 26
4.11 Power factor correction stage waveforms. Implementation of zero crossing mitigation controller removes crossover distortion, and using proportional-integral-squared compensator removes phase shift between input voltage and input current. ........ 28
4.12 Power factor correction stage waveforms. Implementation of zero crossing mitigation controller removes crossover distortion, and using proportional-integral-squared compensator removes phase shift between input voltage and input current. 28

4.13 Zoomed view of zero crossing event of the converter. Input current smoothly commutes between two legs of the half bridges. 28

5.1 Input voltage $v_{in}$, input current $i_{in}$ and output voltage $v_{out}$ waveforms of PFC converter with $V_{in} = 80 \ V_{rms}$, $I_{in} = 11.8 \ A_{rms}$ and $V_{OUT} = 144 \ V$. 32

5.2 Input voltage $v_{in}$, input current $i_{in}$ and output voltage $v_{out}$ waveforms of PFC converter with $V_{in} = 240 \ V_{rms}$, $I_{in} = 28 \ A_{rms}$ and $V_{OUT} = 378 \ V$. 32

5.3 Switching waveforms of the bridgeless boost ac-dc converter. 32

5.4 AC-DC bridgeless boost converter behavior around input voltage zero crossing. The waveforms depict natural commutation of inductor current between legs of bridgeless boost converter by the improved switching scheme of the converter. 33

5.5 Measured efficiency of the power factor correction stage of the drivetrain integrated electric vehicle charger. 33

5.6 Input voltage $v_{in}$, input current $i_{in}$ and output voltage $v_{out}$ waveforms of PFC converter with $V_{in} = 120 \ V_{rms}$, $I_{in} = 13.1 \ A_{rms}$ and $V_{OUT} = 388 \ V$. 35

5.7 Input voltage $v_{in}$, input current $i_{in}$ and output voltage $v_{out}$ waveforms of PFC converter with $V_{in} = 210 \ V_{rms}$, $I_{in} = 24.2 \ A_{rms}$ and $V_{OUT} = 400 \ V$. 35

5.8 Switching waveforms of the bridgeless boost ac-dc converter. 35

5.9 AC-DC bridgeless boost converter behavior around input voltage zero crossing. The waveforms depict natural commutation of inductor current between legs of bridgeless boost converter by the improved control architecture of the converter. 36

5.10 Dual-active bridge converter input ($v_{bus}$) and output ($v_{bst}$) voltage waveforms with $V_{bus} = 375 \ V_{nom}$ and $V_{bst} = 200 \ V_{nom}$. 36
5.11 Dual-active bridge converter inductor current waveforms. The waveforms show that the converter achieves zero voltage switching on both primary and secondary H-bridges and switching frequency $f_{sw} = 120$ kHz.

5.12 Boost converter input voltage ($v_{bst}$), output voltage ($v_{batt}$), average input current ($\langle i_{bst} \rangle_T$) and average output current ($\langle i_{batt} \rangle_T$) waveforms with $V_{bst} = 375$ $V_{nom}$, $V_{batt} = 200$ $V$, $I_{bst} = 33$ $A_{nom}$ and $I_{batt} = 36.4$ $A_{nom}$.

A.1 Approaches to control of pulse width modulated (PWM) switching converters: (a) Conventional feedback controller architecture and (b) Hybrid feedforward controller architecture. Instead of sensing the output and comparing it with a reference command, the hybrid feedforward controller senses converter independent variables $v_x$, $i_y$ and processes them directly using the converter open loop characteristics to determine the switch duty cycle $d$ so as to achieve a control objective $u_{out} = u_{ref}$.

A.2 Feedforward output voltage regulation of boost converter. Open loop characteristics of the boost converter are utilized to achieve the output voltage regulation. Independent converter variable, input voltage $v_{in}$ is sensed and processed to generate the converter duty cycle $d$.

A.3 Power factor correction functionality (PFC) implemented using two different control approaches: (a) PFC functionality implemented using conventional average current mode feedback control and (b) combined hybrid feedforward-feedback control. The feedback control architecture is not affected by the converter topology, as the converter topology and its circuit element values are only used to determine the compensator parameters. In contrast, hybrid feedforward control utilizes the converter topology and characteristics in the control architecture. The converter independent variables, input voltage $v_{in}$ and output voltage $v_{out}$, are utilized to synthesize the hybrid feedforward controller.
A.4 Input voltage ($v_{in}$), input current ($i_{in}$) and output voltage ($v_{out}$) waveforms of a boost converter operating in DCM and acting as a PFC stage with input voltage of 80 $V_{rms}$, input current of 11.8 $A_{rms}$ and output voltage of 144 V.

A.5 Converter for battery charging considered as an example to illustrate the development of a new hybrid feedforward control architecture: (a) converter architecture, and (b) large signal model of the converter.

A.6 Battery charger circuit. First stage is a four switch buck-boost converter, functioning as PFC stage. The two independent duty cycle commands are marked as $d_{buck}$ and $d_{boost}$. Second stage is a buck converter functioning as power regulation stage.

A.7 Four switch buck-boost duty cycle commands $d_{buck}$ and $d_{boost}$ plotted over a half line cycle. The two duty cycle commands are non-overlapping, which allows the converter to achieve smooth transition between buck and boost modes over its entire range of operation.

A.8 Input voltage ($v_{in}$), input current ($i_{in}$) and output voltage ($v_{out}$) waveforms of four stage buck boost converter acting as PFC stage with Input voltage of 120$V_{rms}$, input current of 7.9$A_{rms}$ and output voltage of 110V. Lag in input current with respect to input voltage is due to filtering capacitor connected at the input to filter out switching current from line current.

B.1 Schematic of architecture A.

B.2 Loss distribution of power converters in architecture A of integrated charger.

B.3 Loss distribution among power converters in architecture A of integrated charger.

B.4 Schematic of architecture B.

B.5 Loss distribution of power converters in architecture B of integrated charger.

B.6 Loss distribution among power converters in architecture B of integrated charger.

B.7 Schematic of architecture C.

B.8 Loss distribution of power converters in architecture C of integrated charger.
B.9  Loss distribution among power converters in architecture C of integrated charger. . . 64
B.10 Schematic of architecture D.  . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 64
B.11 Loss distribution of power converters in architecture D of integrated charger. . . . . 65
B.12 Loss distribution among power converters in architecture D of integrated charger. . . 65

D.1 Photograph of the Si based bridgeless boost PFC converter. . . . . . . . . . . . . . . . . . 86
D.2 Photograph of the SiC based bridgeless boost PFC converter. The converter utilizes
four separate PCB boards: (1) Power board containing major pathway for power flow
(2) Driver board containing gate drivers and switches supporting circuitry (3) Sensor
board containing input voltage and input current sensors, and (4) Planar inductor
board containing windings of the planar inductor. . . . . . . . . . . . . . . . . . . . . . . . . . . . . 87
D.3 Layout of the power board for the SiC based bridgeless boost PFC converter. . . . . 88
D.4 Layout of the gate driver board for the SiC based bridgeless boost PFC converter. . . 88
D.5 Layout of the sensor board for the SiC based bridgeless boost PFC converter. . . . . 88
D.6 Layout of the planar inductor board for the SiC based bridgeless boost PFC converter.
The planar inductor is a coupled inductor structure with 12 layers. . . . . . . . . . . . . 89
Chapter 1

Introduction

1.1 Motivation

A high power on-board charger provides convenience, reduces range-anxiety and is considered vital to accelerate the adoption of electric vehicles (EVs) [2]. However, the additional weight of a high power on-board charger can reduce the range of the vehicle and its volume can impose unfavorable design constraints. Hence, reducing the weight of on-board chargers is an important goal for vehicle manufacturers. Different approaches to implementing on-board electric vehicle chargers have been proposed [3,4]. However, in all of these approaches, the on-board charger is considered a separate power electronic converter. Some attempts have also been made to integrate on-board charging capability with the electric drivetrain by leveraging the windings of the electric motor [2,5,6]. However, this approach introduces complexity into the design of the electric motor and can impact its cost and reliability. Hence, there is a need to explore alternative ways of integrating on-board charging functionality that minimize added weight and maintain high efficiency.

1.2 Drivetrain Architecture and Charger Integration Approach

Recently a new drivetrain architecture for electric vehicles that utilizes a composite boost dc-dc converter between the vehicle battery and the motor drive has been introduced [1]. The proposed composite boost dc-dc converter comprises dissimilar converter modules combined into a power conversion system with superior performance in terms of efficiency and power density compared to the traditional drivetrain boost converter. The architecture of the composite boost dc-
dc converter is shown in Fig. 1.1. This modular approach to boost the battery voltage provides an opportunity to leverage some of its power conversion stages as part of the charging function. Such charger integration can result in minimal additional weight, while providing high power on-board charging capability, increasing the on-board charger power density.

1.3 Thesis Contributions

This thesis identifies, explores and quantitatively compares alternative architectures for integrating the charging functionality with the composite boost converter. The architecture among these that provides an effective trade-off between added weight and charging losses is selected for detailed design and testing. This drivetrain-integrated charger only adds a bridgeless boost based power factor correction (PFC) ac-dc stage, an H-bridge and a single winding to the composite boost dc-dc converter. Hence, it significantly reduces the additional weight required for the on-board charging functionality.

The proposed charger architecture comprises a PFC boost ac-dc stage, a dual-active bridge stage and a boost converter current regulation stage. This thesis presents detailed design and control of the boost converter based PFC stage. Two PFC implementations, using traditional feedback control and hybrid feedforward control are presented. For traditional feedback control, it is shown that the new control architecture, introduced in this thesis, can mitigate cross-over distortions and allow PFC converters to achieve near-ideal PFC rectifier performance. In case of hybrid feedforward control architecture, PFC ac-dc conversion has been achieved using current sensor less technique.
and good boost PFC rectifier performance is demonstrated. Furthermore, zero crossing distortion effects observed in PFC converters and methods to mitigate these effects are also discussed for both cases. This thesis also presents control architectures for the dual-active bridge stage acting as an isolation stage and voltage regulation stage, and the boost stage acting as a power regulation stage.

Finally, two 6.6 kW prototypes of the proposed charger's PFC stage have been designed, fabricated and tested. One prototype utilizes Silicon devices, while the other utilizes Silicon Carbide devices. It is shown that the proposed charger architecture’s PFC stage achieves greater than 97% peak efficiency and the on-board charging functionality can be added with only a 34% increase in the weight of the composite boost converter.

This thesis also presents a generalized approach for the development of hybrid feedforward control architectures and identifies the underlying fundamental principles. New hybrid feedforward control architecture implemented on a two stage battery charging system is also introduced. The architecture comprises of a four-switch buck-boost PFC stage, introduced as part of this thesis, and battery power regulation buck stage. Experimental results have also been presented for the proposed hybrid feedforward control architectures.

1.4 Thesis Organization

The remainder of this thesis is organized as follows. Approaches to integrating the charging functionality with the composite dc-dc converter are discussed and quantitatively compared in Chapter 2. Based upon the quantitative results, a drive-train-integrated charger architecture is identified and discussed in detail in Chapter 3. Chapter 4 discusses the control challenges associated with the PFC stage of the integrated charger, such as zero crossing distortion. Novel ways to overcome these challenges are also discussed in this chapter. The control architectures developed here can be used for both ac-dc conversion and dc-ac conversion, and can process both active and reactive power. Chapter 5 presents the detailed design, simulation results and experimental results of two prototyped chargers; the first uses Si devices and the second uses SiC devices. Finally, the summary and conclusions of this thesis, as well as recommendations for future work are presented in
Chapter 6. This thesis also contains four appendices. The general structure for hybrid feedforward control architectures is introduced in Appendix A. Appendix B documents weight distribution and losses for the charger architectures introduced in Chapter 2. Appendix C contains the microcontroller code used for the implementation of the ac-dc conversion stage of the charger. Finally, Appendix D contains layout of the prototype boards used for testing of ac-dc converters with both Si and SiC devices.
2.1 DC-DC Composite Boost Converter Architecture

The architecture of the composite boost dc-dc converter [1] used as a drivetrain boost converter is shown in Fig. 2.1. This architecture comprises of a buck module, a boost module and a dual active bridge operated as a dc transformer (DCX). The composite boost converter achieves a high voltage gain by connecting the outputs of the boost and the DCX modules in series. The buck module forms a cascade with the DCX stage, and its input is parallel-connected to the input of the boost module. This architecture allows the composite boost converter to achieve high output voltages with relatively low breakdown voltage switches having low on-resistance and fast switching speed. Each module of the composite converter processes only a fraction of the total power processed by the converter. This approach enables efficiency and power density optimization of the overall converter by utilizing a control strategy that maximizes the amount of direct power processed by the converter across a wide range of operating conditions. These factors enable the composite boost converter to achieve greater efficiency and power density than a traditional boost converter. In addition to these benefits, the modular nature of the composite boost converter opens up the possibility of integrating the charging functionality within the drivetrain, as discussed in detail in the next section.

2.2 Charger Integration Approaches

The charging functionality can be integrated with the composite boost architecture in multiple ways. Four of the most promising approaches to integrate an on-board charger into the composite
boost converter are shown in Fig. 2.2. Figure 2.2(a) shows architecture A, a bridgeless boost stage followed by a phase-shifted full-bridge isolated dc-dc converter. The bridgeless boost stage is operated as a power factor correction (PFC) rectifier, and the phase-shifted full-bridge isolated dc-dc converter provides isolation and also regulates the battery charging current. Although this architecture involves only two cascaded power stages, resulting in relatively low losses, the added weight of this architecture is the highest among the four considered charging architectures. This is due to its relatively weak integration with the composite boost converter, as it only utilizes the composite converter’s boost stage filter. Figure 2.2(b) shows architecture B, which comprises a diode rectifier followed by an isolated full-bridge boost stage to achieve power factor correction and isolation. The full-bridge boost converter integrates into the existing drivetrain by adding an additional winding to the transformer of the DCX and utilizing its secondary side H-bridge for rectification. These stages are then followed by the drivetrain buck converter that regulates the battery current. While this architecture provides strong integration with the composite boost converter, increased losses in the isolated boost stage degrades its efficiency.
Figure 2.2: Four proposed architectures suitable for integration of charging functionality with the composite boost converter of Fig. 2.1.
Figure 2.2(c) shows architecture C, which comprises a bridgeless boost converter followed by a dual active bridge (DAB) converter and the drivetrain buck converter. The bridgeless boost converter at the front end is operated as a PFC stage. The DAB is used to provide isolation and the drivetrain buck converter is then used to regulate the battery current. The DAB is implemented by adding a primary side H-bridge and an additional transformer to the existing drivetrain DCX. The secondary side H-bridge of the drivetrain DCX is reused in the charging operation. This architecture can be further integrated with the composite boost converter by integrating the two DAB transformers. This can be achieved by utilizing a three winding transformer in the DAB stage, as shown in architecture D of Fig. 2.2(d). Architecture D achieves strong integration by effectively utilizing the existing drivetrain stages and appears highly promising.

2.3 Quantitative Comparison of the Alternative Drivetrain-Integrated Charger Architectures

In order to quantitatively compare the proposed integrated charger architectures shown in Fig. 2.2, loss models are developed to estimate their losses over a line cycle. These loss models include switch conduction and switching losses, and inductor and transformer core and winding losses. In addition, models to compute the weight of each architecture are also developed. The weight models consider the weight of the magnetic and capacitive components, as well as the heat sinks, since they constitute the largest fraction of the converter weight. The computed weights and losses from these models for the four considered architectures are presented in Table 2.1, details of which can be found in Appendix B. It can be observed that architecture A has the least losses but contributes the most additional weight. On the other hand architecture B adds substantially less weight but has high losses. Note that architecture B adds least components but not the least weight due to increased size of the heat sink. Since in an EV application both weight and efficiency are important considerations, architecture D appears to provide an excellent tradeoff between these attributes as it effectively leverages the existing modules of the composite boost converter. Thus architecture D is selected for detailed design, fabrication and testing.
Table 2.1: Comparison of proposed integrated charger architectures in terms of added weight and full-power losses for a 6.6 kW on-board EV charger.

<table>
<thead>
<tr>
<th>Architecture</th>
<th>Added Weight [kg]</th>
<th>Losses [W]</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>2.67</td>
<td>238</td>
</tr>
<tr>
<td>B</td>
<td>1.97</td>
<td>322</td>
</tr>
<tr>
<td>C</td>
<td>2.07</td>
<td>268</td>
</tr>
<tr>
<td>D</td>
<td>1.90</td>
<td>265</td>
</tr>
</tbody>
</table>
Chapter 3

Drivetrain-Integrated Charger Design and Control

3.1 Selected Drivetrain-Integrated Electric Vehicle Charger Architecture

The schematic of the proposed drivetrain-integrated charger is shown in Fig. 3.1. The first stage of the architecture is a bridgeless boost PFC. The second stage is a dual-active bridge formed by adding an H-bridge plus a winding, and utilizing the existing H-bridge of the composite converter’s DCX stage. The final stage is formed by the drivetrain buck converter which helps regulate the battery power.

3.2 AC-DC Bridgeless Boost Converter

The bridgeless boost topology is well suited for high efficiency ac-dc power conversion since it eliminates the passive rectifier bridge used in traditional PFC rectifiers, resulting in reduced component count and decreased conduction losses [7]. A detailed design of the PFC stage of the proposed drivetrain-integrated charger is conducted and optimizations are performed to minimize the losses over an ac line cycle. Additional details regarding these optimizations are presented in Chapter 5. Based on the results of these optimizations, discontinuous conduction mode (DCM) for operation of the bridgeless boost PFC was selected when utilizing silicon (Si) super-junction FETs, and continuous conduction mode (CCM) of operation was selected when utilizing silicon carbide (SiC) MOSFETs.
3.2.1 Bridgeless Boost Converter Operating in DCM

When the bridgeless boost PFC stage utilizes silicon superjunction MOSFETs, DCM operation results in reduced converter weight and losses due to reduction in inductor size and switching losses. An additional advantage of this topology compared to a totem-pole bridgeless boost (i.e. in which all four switches are actively controlled) is that the active switches are referenced to the same node, simplifying gate driver requirements. Operating the bridgeless boost converter in DCM provides additional benefits related to control implementation. In DCM operation, PFC functionality can be implemented without sensing the inductor current, which typically requires a high-bandwidth sensor. Under DCM operation, with the selected control implementation, only the input and output voltages of the converter need to be sensed. In DCM, the average inductor current can be expressed as [8]:

$$\langle i_l \rangle_{T_s} = \left( \frac{T_s}{2L} \frac{d^2}{1 - \frac{v_{in}}{v_{bus}}} \right) v_{in}.$$  
(3.1)

Here $\langle i_l \rangle_{T_s}$ is the local average of the inductor current, $v_{in}$ is the input line voltage, $v_{bus}$ is the dc bus voltage, $T_s$ is the switching period, $d$ is the duty ratio and $L$ is the bridgeless boost inductance.

The control objective for achieving PFC functionality is:
\[ \langle i \rangle T_s = \frac{v_{in}}{R_e} \]  

where \( R_e \) is the emulated input resistance of the converter. It can be observed from (3.1) that the average input (inductor) current can be made proportional to input voltage, as required to achieve the control objective (3.2), by adjusting the duty ratio such that the term in parenthesis becomes constant.

\[ \frac{T_s}{2L} \frac{d^2}{1 - \frac{v_{in}}{v_{bus}}} = \frac{1}{R_e}. \]  

This can be manipulated to yield the following control law [9,10] :

\[ d = \sqrt{\frac{2L}{R_e T_s} \left(1 - \frac{v_{in}}{v_{out}}\right)}. \]  

The value of the emulated resistance \( R_e \) can be adjusted by a PI compensated outer voltage control loop.

For the bridgeless boost converter, since the converter has two active switches, the control strategy can be implemented as:

\[ d_1 = \begin{cases} 
  d, & v_{in} > 0 \\
  1, & v_{in} < 0 
\end{cases} \]  

Figure 3.2: Power factor correction functionality (PFC) implemented using combined hybrid feedforward-feedback control.
Here \( d_1 \) and \( d_2 \) correspond to duty cycles of the two active switches of the bridgeless boost converter.

The control architecture falls in the category of hybrid feedforward control architecture embedded inside a feedback loop [11]. Hybrid feedforward control architectures are developed more generally as a part of this thesis in Appendix A. A digital implementation of this control strategy using a micro-controller is selected for the PFC stage. The micro-controller senses the input voltage and output voltage of the converter and processes them as given by (3.5) and (3.6) to modulate the duty ratio of the converter in each switching interval. As can be observed from the duty ratio modulation equation, the operations of square root, division, multiplication and subtraction are required to implement the control. This can be easily performed in a modern micro-controller.

The converter outer voltage loop is designed to adjust the emulated input resistance of the converter. The outer voltage control loop is designed in a traditional manner, by assuming that the inner current control loop works ideally at low frequencies and the ideal rectifier model adequately represents the low-frequency system behavior [8]. When the converter outer voltage loop is designed properly, it regulates the converter output voltage \( v_{\text{bus}} \). Any tolerance in the converter inductance value, which may appear to affect the performance of the inner current control loop (as the converter duty ratio given by (3.4) depends on the inductance value), is automatically corrected by the outer voltage loop. Since the output of the outer voltage loop compensator appears as a multiplicative factor with the circuit parameters in (3.4), inaccuracy in any other circuit parameters also get compensated by the action of outer feedback loop.

It is worth noting that when the controller is implemented to switch the duty cycle commands at the input voltage zero crossings of the input voltage as expressed by (3.5) and (3.6), zero crossing distortion may appear. A detailed discussion of the reasons and the nature of this distortion, along with a solution to mitigate this problem is covered in Chapter 4.
3.2.2 Bridgeless Boost Converter Operating in CCM

A totem pole H-bridge structure is used as a bridgeless boost converter to achieve ac-dc conversion while operating the converter in CCM. Bridgeless boost converter can be operated as a PFC ac-dc conversion stage using traditional average current mode control.

The general architecture of traditional average current mode control implemented on bridgeless boost converter is shown in Fig. 3.3. Average current mode control is implemented by sensing the input current, comparing it with a reference command and passing the error through a compensator. The compensator generates duty cycle command for the converter. Input current reference is generated by sensing and scaling input voltage of the converter. Furthermore, output voltage of the converter also needs to be controlled. This is achieved by sensing the converter output voltage, comparing it with the reference voltage and passing the error through a compensator. Output of the compensator modulates the amplitude of the current reference [8].

The control command $v_c$ generated by the inner current loop compensator is processed further to generate two duty cycle commands $d_1$ and $d_2$ for the converter. For the positive half line cycle, the bottom switch in the half-bridge leg receiving the command $d_1$ is turned on and the switches in the half-bridge leg receiving the command $d_2$ are switching, and for the negative half line cycle, the bottom switch in the half-bridge leg receiving the command $d_2$ is turned on and the switches in the half-bridge leg receiving the command $d_1$ are switching. The converter needs to decide when...
to make the transition between these two modes of operation. This leads to issues related to zero crossing distortion observed in the converter, as inaccurate transition time can cause the converter to switch inaccurately between the switching mode of its two half bridge legs. This issue is dealt in Chapter 4, which leads more generally to the use of this structure for ac-dc and dc-ac conversion, as well as giving it the capability to process both active and reactive power. The end result is a half bridge transition block, which is used to effectively rectify the compensator control command and is shown in Fig 3.3.

### 3.3 Dual-Active Bridge Converter

The second stage of the charger is a dual-active bridge (DAB) converter which is integrated with the drivetrain using a three winding transformer, implemented by simply adding one extra winding to the existing drivetrain DCX transformer. One side of the transformer, which corresponds to the output of the DCX is turned off while the charger is charging the battery. The power transfer takes place from the grid side DAB full bridge to the battery side DAB full bridge. For charging purposes, the DAB converter provides isolation and regulates the voltage at its output.

The control of the DAB converter is done by switching the primary and secondary H-bridges at approximately 50% duty cycle and controlling the phase shift between the two H-bridges [12]. The control is based on sensing the converter output voltage, comparing it with a reference and passing the error through a compensator to generate the converter phase shift. This control loop is designed to have much lower bandwidth than twice line frequency. The twice line frequency ripple present on the grid side of the DAB is buffered by both the grid side and the battery side capacitors ($C_1$ and $C_2$) in Fig. 3.1.

### 3.4 Power Regulation Boost Converter

The final stage of the charger is the drivetrain bi-directional buck, which functions as a boost converter during the charging operation. The converter regulates the current flowing into the battery. To realize the controller on this boost converter, inductor current is sensed and used in a
standard feedback architecture to generate the duty ratio command. Since the inductor is present at the input of the converter, current control is not done in a standard fashion by sensing the battery current and comparing it with the reference current. Instead, as the control objective is to regulate the battery power, controller is synthesized to make the converter behave as a power sink at the input port [8]. Since in steady state the converter has no net energy storage, the input power comes out of the converter output port and the output port behaves as a power source, which charges the battery at constant power. The schematic, the large signal model and the control architecture of this power regulating boost stage is shown in Fig. 3.4.

![Figure 3.4: Boost converter acting as a power regulation stage to regulate the battery power: (a) Boost converter topology (b) Large signal model and (c) Controller implemented to achieve input power sink characteristics.](image)

To make the converter act as a power sink at the input port, the input voltage of the boost converter is sensed and used to generate the converter current reference. The converter current reference can be expressed as:
\[ i_{\text{ref}} = \frac{P_{\text{ref}}}{v_{\text{bst}}} \] (3.7)

The converter feedback loop is designed to achieve input current regulation of the converter, which in case of ideal operation of current control loop can be expressed as:

\[ \langle i_{\text{bst}} \rangle_{Ts} = i_{\text{ref}} \] (3.8)

Thus in steady state, the converter input power can be expressed as:

\[ P_{\text{in}} = v_{\text{bst}}i_{\text{ref}} = P_{\text{ref}} \] (3.9)

Ignoring the losses in the converter, as they are small compared to the input power, the converter output power can be expressed as:

\[ P_{\text{out}} = P_{\text{ref}} \] (3.10)

When this control is implemented, the output power of the converter can be controlled without directly sensing the battery current. It can be noted here that the converter input voltage has twice line frequency ripple. The ideal power regulation stage will get rid of this twice line frequency ripple and charge the battery with constant current. Thus bandwidth of the inner current loop is designed to be much larger than twice line frequency to track the ripple in input current reference appearing because of ripple in the input voltage of the converter.
Chapter 4

Zero Crossing Distortion in Power Factor Correction Rectifiers

Zero crossing distortion is a common problem in PFC converters [13]. The problem has been extensively studied and researched upon in literature and different reasons have been identified in the past. This chapter deals with zero crossing distortions observed in PFC converters operating in DCM and CCM, and presents new control architectures that mitigate these zero crossing distortions.

4.1 AC-DC Bridgeless Boost Converter Operating in DCM

The topology and control architecture of the bridgeless boost PFC stage operating in DCM is shown in Fig. 4.2. The control architecture has already been discussed in Chapter 3. Like other PFC converters, this architecture is also prone to zero crossing distortion. The reason for this distortion in the line current near the zero crossing of the line voltage is inaccurate sensing of input voltage and
delays in the control loop and the gate driver. This explanation is illustrated through an example in Fig 4.2. Figure 4.2(a) shows when the actual input line voltage crosses zero volts, and Fig. 4.2(b) shows when the sensor senses the input voltage zero crossing - in this example before it actually happens. If the controller is implemented as given by (3.5) and (3.6), then due to inaccurate input voltage sensing, the controller causes two legs of the H-bridge to switch their operation before it should actually happen. This causes the two switches of the converter to remain on for interval II as marked in Fig. 4.2. Notice that even though the controller commands switch \( Q2 \) to switch in interval II, the transistor remains on due to the conduction of the body diode, as the current only changes polarity when the input voltage changes polarity in DCM. This causes input voltage to appear across the inductor for the time interval II. Since, small inductors are normally employed in DCM converters, even a small input voltage around zero crossing can cause spikes the in inductor.
current, which is also the line current.

In order to mitigate this issue, both transistors of the bridgeless boost converter are switched on and off at the same time around the zero crossing. This prevents a large build-up in the inductor current, as the current is periodically interrupted by the switching of the two transistors and enables smooth commutation of the current from one leg of the converter to the other, as shown in Fig. 4.2(d). Notice that in the operation of the bridgeless boost converter, one of the two switches of the converter remains on for half line cycle. Thus normal operation of the converter remains unchanged by overlapping the switching time of the two switches around the voltage zero-crossings. This zero-crossing mitigation strategy can be expressed in terms of the duty ratios of the two converter legs as:

\[
d_1 = \begin{cases} 
    d, & v_{in} > -\delta V \\
    1, & v_{in} < -\delta V
\end{cases}
\]

\[
d_2 = \begin{cases} 
    1, & v_{in} > \delta V \\
    d, & v_{in} < \delta V
\end{cases}
\]

Here \( \delta V \) is the estimated inaccuracy in the input voltage sensing. Both transistors are switched during the time that the line voltage is between \( -\delta V \) to \( \delta V \), allowing current to naturally commutate between the two legs of the half bridge. This control strategy to overcome sensing inaccuracy causes switching of the two legs of the converter to overlap around the voltage zero crossings, preventing inductor current spikes. Furthermore, this control strategy is simple to implement. In case when the input voltage sensor senses the voltage zero crossing after it actually happens, as shown in Fig. 4.2(c), the effects observed are similar and the proposed control strategy solves the problem. Finally, it is interesting to note that with this control strategy current commutation is natural as the commutation happens automatically without any effort on the part of the controller to accurately detect the input voltage zero crossing.
4.2 AC-DC Bridgeless Boost Converter Operating in CCM

![Bridgeless Boost PFC rectifier](image)

The control objective of ac-dc single phase power factor correction (PFC) rectifiers is to shape input current drawn from ac grid to be proportional to ac grid voltage, thus emulating the behavior of a resistor connected across the grid [8]. Ideally, this results in achieving unity power factor. Modern ac-dc PFC rectifiers approach the control objective closely, except around input voltage zero crossings [13–18]. Around input voltage zero crossing significant distortions in input current can be observed, hereby referred to as zero-crossing current distortion. The current distortions can deteriorate the performance of PFC rectifiers significantly. Previous work in the area [13–18] has identified some underlying reasons for the distortion, which include phase shift between input voltage and input current in average current mode controllers and discontinuous conduction mode of operation of the circuit around input voltage zero crossings, among others.

The objectives of this work are to expose another very significant cause of zero-crossing current distortion in traditional PFC rectifiers. It is identified that high frequency content is introduced in current reference due to use of diode bridge at the converter input which excites dynamic response of converter. Once diode bridge is identified as root cause of zero current distortion, bridgeless ac-dc converters, in particular bridgeless boost converter, as shown in Fig. 4.3, is considered to eliminate the root cause. A new control architecture capable of substantially reducing the zero-crossing distortion in bridgeless boost PFC rectifiers is introduced and experimentally validated.

Furthermore, to achieve high power density, it is desired to reduce the size of the boost inductor, as it constitutes a large proportion of the PFC converter’s volume. Thus it seems advantageous
to reduce the inductance value of the PFC boost inductor. However, a low inductance results in more significant phase shift between input voltage and input current [13], which can degrade the quality of input current waveform. For example, in conventional PFC controllers, an increased phase shift between input ac voltage and ac current is shown [13] to have significant impact on converter performance.

In bridgeless boost ac-dc converter, a PI compensator with bandwidth of 4 kHz would be considered sufficient to track low frequency (50-60 Hz) current reference command. Yet experiments presented here seem to disagree with the understanding, especially in cases where the converter inductance is significantly reduced. The reason for the appearance of the phase shift between input voltage and input current is revisited here and an intuitive explanation is presented for the observed phenomena. Once the phenomena is clearly identified, it is shown that a $PI^2$ compensator can eliminate the problem. Experimental results predict near-ideal unity power factor correction performance with negligible to no zero-crossing distortion.

Traditional ac-dc converter control architecture is first reviewed in Section 4.2.1 and the underlying reasons for zero-crossing distortion are identified. This is followed by development of new control architecture to eliminate zero-crossing distortion in Section 4.2.2. Effects of utilizing small inductors on the performance of PFC converters is analyzed in Section 4.2.3. Experimental results validating the proposed control modifications and verifying near-unity power factor correction performance are presented in Chapter 5.

4.2.1 Traditional Control Architecture for Single-Phase AC-DC Converter

Traditional ac-dc conversion stage is shown in Fig. 4.4 [8]: it rectifies an input ac voltage and then employs a dc-dc conversion stage controlled to shape the input current. Rectified input ac voltage ensures that input of the dc-dc converter remains positive for a complete line cycle. The rectified ac input voltage is typically utilized to generate the current reference. As a result, the current reference has significant high frequency components as shown in Fig. 4.5. As the input ac voltage reaches close to zero volts, current reference suffers from a kink due to rectification. If
Figure 4.4: Traditional ac-dc conversion stage. Rectified input ac voltage \( v_{in} \) is used to generate the input current reference \( i_{ref1} \).

Figure 4.5: Input current reference for bridgeless boost converter topology: (a) Use of rectified input ac voltage to generate the current reference introduces high frequency content in the reference command. (b) Direct use of input ac voltage to generate the current reference command removes high frequency content from the reference command.

the current controller is designed to track line frequency signal, this kink in the rectified signal can excite converter dynamics, and cause oscillations and distortions around input voltage zero crossings [13]. Furthermore, since the converter can potentially go in discontinuous conduction mode (DCM) around input voltage zero crossings, the rectifier performance can additionally deteriorate [14, 18]. Finally, any phase shift between ac line voltage and current can grossly exacerbate the distortion [13]. Bridgeless boost rectifiers eliminate the diode-bridge rectification stage. The ac line input voltage, which is used to generate the current reference, is not explicitly rectified. Yet control architectures are typically synthesized in a manner that an effective rectification is done by actively switching between legs of the converter at the input voltage zero crossings [3, 7]. Thus the zero crossing current distortion phenomena remains essentially the same for the bridgeless boost converters.
4.2.2 Control Architecture that Mitigates Zero-Crossing Distortion in Boost PFC Rectifiers

The starting point in the development of the new control architecture is to use the input ac line voltage in generating a current reference command directly, without rectification, as shown in Fig. 4.6. No effort need be placed to switch between legs of the converter around input voltage zero crossings. Instead, the controller automatically controls the converter in a way that line voltage is tracked for both positive and negative half line cycles. Since high frequency content from the reference command is removed, the controller bandwidth can be eased as the controller is required to simply track a sinusoidal ac line frequency current reference signal.

While the approach applies more generally, details are presented here for the bridgeless boost totem-pole configuration shown in Fig. 4.6. The converter is operated in continuous conduction mode (CCM) over the line cycle by switching the half-bridge switches in complementary manner.
Figure 4.8: Current reference, control command $v_c$ and duty cycles $d_1$ and $d_2$ plotted over half line cycle.

To achieve the control functionality, two half bridges are switched independently. The controller senses the input voltage and scales it to generate the current reference. The sensed input (inductor) line current is compared with the reference command to generate an error, which is then processed by the controller to generate the control command $v_{c_1}$. The controller command $v_{c_1}$ is processed further, as shown in Figs. 4.7 and 4.8, to generate the two duty cycle commands for the converter. Duty cycle commands are generated from the control signal $v_{c_1}$ by adding and subtracting amplitude of the pulse-width modulator ramp $V_m$, and are allowed to saturate naturally when current transitions between the half bridges. It is important to note that when the half bridge transition block is implemented as shown in Fig. 4.7, the feedback loop remains negative over the full line cycle of operation.

Since the half bridge transition block automatically handles the transition between two half bridges of the converter, from the point of view of the controller, the converter appears to be working at line frequency. The control signal is periodic with line frequency, as shown in Fig. 4.8. Figure 4.9 shows the experimental input voltage and current waveforms for the PFC converter with the controller implemented as discussed above, and the compensator designed as a standard proportional-integral (PI) compensator with a current-loop bandwidth of 4 kHz. In spite of the relatively low bandwidth current control loop, and in spite of the noticeable phase shift between voltage and current, no zero-crossing distortion can be observed. The phase-shift issue is addressed further in the next section.
4.2.3 Phase Shift due to Input Voltage Feedforward Effect

With a PI compensator, and the current loop bandwidth of 4 kHz, one would expect the converter to track the 60 Hz line frequency signal very well. However, the waveforms in Fig. 4.9 show a significant phase shift in the input current. A zoomed view of the waveforms is shown in Fig. 4.10, where a phase shift of $25^o$ can be observed. The root cause of the phase shift is related to the input voltage feedforward effect described in [13], which is particularly pronounced when a small boost inductance is employed. To understand the effect of this non-ideality, consider the boost converter averaged dynamics which, neglecting variations in output voltage, can be written as [8]:

$$L \frac{d\langle i_{in}\rangle_T}{dt} = \langle v_{in}\rangle_T - d'(t)V_{out}. \quad (4.3)$$

Separating the dc components from ac components and taking the Laplace transform of 4.3, one can write:
\[ i_{in}(s) = \frac{V_{out}}{sL}d(s) + \frac{1}{sL}v_{in}(s). \] (4.4)

If the input voltage of the converter is used to generate the current reference, then the converter dynamic model is shown in Fig. 4.11. Along with the feedback loop designed to regulate the input current, an additional feedforward path from the input voltage to inductor current is present, corresponding to the second term in (4.4). The closed loop transfer function can be expressed as:

\[ \frac{i_{in}(s)}{v_{in}(s)} = \frac{1}{R_e} \frac{T(s)}{1 + T(s)} + \frac{1}{sL} \frac{1}{1 + T(s)} \] (4.5)

where \( T(s) \) is the loop gain, and \( R_e \) is the emulated resistance of the converter,

\[ T(s) = G_c(s)H_{iin}\frac{V_{out}}{sL}, R_e = \frac{H_{iin}}{H_{v_{in}}v_{c2}}. \] (4.6)

Here it can be noted that the second term in (4.5) acts as a disturbance, which should ideally be rejected by the controller. But, this term becomes large if a small boost inductance is employed.

The phase difference between input voltage and input current can be determined by looking at the phase response of the transfer function (4.5) at the line frequency, which comes out to be 25° in the experimental prototype (\( L = 40\mu H, f_s = 120kHz \)), matching the experimentally measured phase shift in Fig. 4.10.

One method to reduce the effect of this disturbance is to increase the loop gain at line frequency. This can be done using a \( PI^2 \) compensator with two integrators instead of a standard PI compensator:

\[ G_c(s) = G_\infty(1 + \frac{w_{z1}}{s})(1 + \frac{w_{z2}}{s}). \] (4.7)

When the \( PI^2 \) compensator is employed, the converter performance based on the control architecture of Section 4.2.2 is improved further. The phase shift is reduced down to 0.2° and the waveforms are as shown in Fig. 4.12, with a zoomed view of the zero-crossing shown in Fig. 4.13. There is no visible zero-crossing distortion or phase shift. Chapter 5 presents more experimental results at various power levels.
Figure 4.11: Power factor correction stage waveforms. Implementation of zero crossing mitigation controller removes crossover distortion, and using proportional-integral-squared compensator removes phase shift between input voltage and input current.

Figure 4.12: Power factor correction stage waveforms. Implementation of zero crossing mitigation controller removes crossover distortion, and using proportional-integral-squared compensator removes phase shift between input voltage and input current.

Figure 4.13: Zoomed view of zero crossing event of the converter. Input current smoothly commutes between two legs of the half bridges.

Lastly, it is noted that the theory and the control approach developed for mitigating zero crossing distortion in ac-dc converter is generally applicable for bridge less ac-dc converter topologies by effective rectification of the control signal, instead of sensed signals. Furthermore, the approach for mitigating zero crossing distortion, introduced here, also enables the converter to have any phase shift between the input voltage and input current of the converter (with an appropriate current reference command), without producing any zero crossing distortion effects. Thus the control architecture introduced here possesses the capability to process active as well as reactive
power. It can be observed from Fig. 4.9 that the converter is processing both active and reactive power without producing any zero crossing distortion, thus validating the idea. Furthermore, the control architecture is fully bi-directional in nature, making it capable of dc-ac conversion as well, if the input and output of the converter are exchanged. Thus to summarize, the theory developed here allows control architectures to be synthesized that enable power converters to become bi-directional and process active and reactive power.
Chapter 5

Prototype Design and Experimental Results

Two 6.6 kW electric vehicle chargers meant to be integrated with a 30 kW composite boost dc-dc converter have been designed and their PFC stages built and tested. One charger is designed to utilize silicon (Si) devices, while the other is designed to utilize silicon carbide (SiC) devices. This chapter presents the design details and the experimental results for these Si and SiC based bridgeless boost PFC stages. Near ideal rectifier performance is achieved in both PFCs, including the mitigation of the zero-crossing distortion discussed in Chapter 4. The control architectures introduced for the second stage dual active bridge (DAB) and the final stage boost converter are also verified in this chapter using computer simulations.

5.1 AC-DC Bridgeless Boost Converter Utilizing Si Devices and Operating in DCM

A detailed design of the PFC stage utilizing silicon MOSFETs is conducted. The design variations considered span a large design space, including continuous and discontinuous conduction modes of operation, switching frequencies spanning 20 kHz to 80 kHz, wide range of component values and different component types. For example, within each system level design iteration, optimal inductors are designed for the PFC boost stage by iterating over various core materials, core geometries and winding parameters. The optimization is performed using an exhaustive search based numerical approach that computes the overall converter weight and line-cycle average losses for each considered design. The final design is selected by searching for solutions that minimize
losses while staying within specified weight limits. As a result of this optimization, a switching frequency of 20 kHz and discontinuous conduction mode (DCM) operation are selected for the bridgeless boost PFC stage prototype. The components used in this optimized prototype of the bridgeless boost PFC stage are listed in Table 5.1. Table 5.1 also provides the design details for the optimized inductor used in this prototype. The active switches in the bridgeless boost stage are implemented using silicon super-junction MOSFETs, while the rectifier diodes are realized using silicon carbide Schottky diodes. The details of the other components used in this prototype are also presented in Table 5.1.

Table 5.1: Design Details of the 6.6kW Bridgeless Boost PFC Stage Utilizing Silicon Devices and Operating in Discontinuous Conduction Mode (DCM)

<table>
<thead>
<tr>
<th>Component</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOSFETs</td>
<td>IPW65R041CFD2 650 V, 68.5 A Silicon Super-Junction MOSFET</td>
</tr>
<tr>
<td>Diodes</td>
<td>C5D50065D 650 V, 50 A SiC Schottky Diode</td>
</tr>
<tr>
<td>Output capacitor</td>
<td>100 µF x 4, 500 V Film Capacitor</td>
</tr>
<tr>
<td>Coupled Inductor</td>
<td>23 µH Amorphous alloy core 10 turns of 1000-strands 38 AWG litz wire</td>
</tr>
<tr>
<td>Microcontroller</td>
<td>TI-TMS320F28069 32 Bit CPU, 90 MHz</td>
</tr>
</tbody>
</table>

The prototyped PFC stage has been tested up to the full power of 6.7 kW. Figures 5.1 and 5.2 show the waveforms of the bridgeless boost stage operating at power levels of 950 W and 6.7 kW, respectively. As can be seen in both figures, the input current is fairly sinusoidal. The phase shift in input current relative to input voltage is due to capacitive filters employed at the converter input to filter out switching ripple from the line current. The peak-to-peak dc bus voltage ripple is designed to be 30%, which can also be seen from Fig. 5.2. Figure 5.3 shows the switching waveforms for the switch-node voltage, the inductor current and the MOSFETs gate-to-source voltage.

The effectiveness of the zero-crossing distortion mitigation strategy can be seen from Fig. 5.4.
Figure 5.1: Input voltage $v_{in}$, input current $i_{in}$ and output voltage $v_{out}$ waveforms of PFC converter with $V_{in} = 80\, V_{rms}$, $I_{in} = 11.8\, A_{rms}$ and $V_{OUT} = 144\, V$.

Figure 5.2: Input voltage $v_{in}$, input current $i_{in}$ and output voltage $v_{out}$ waveforms of PFC converter with $V_{in} = 240\, V_{rms}$, $I_{in} = 28\, A_{rms}$ and $V_{OUT} = 378\, V$.

Figure 5.3: Switching waveforms of the bridgeless boost ac-dc converter.

Figure 5.4 shows that line current transitions across the line zero crossing without any visible distortion. Both active switches of the converter switch around the zero crossing to allow for this seamless zero crossing.

The efficiency of the PFC stage across a more than 10:1 output power range with an input line voltage of 240 Vrms has been measured and is shown in Fig. 5.5. It can be seen that the PFC
stage achieves a peak efficiency of 97.7% and maintains efficiencies above 97% across a 6:1 output power range. The worst case measured efficiency is 96.3%, which occurs at the lowest measured output power level.

Figure 5.5: Measured efficiency of the power factor correction stage of the drivetrain integrated electric vehicle charger.

5.2 AC-DC Bridgeless Boost Converter Utilizing SiC Devices Operating in CCM

A detailed design of the PFC stage with Silicon Carbide (SiC) MOSFETs is conducted. The design variations considered span a large design space, including continuous and discontinuous conduction modes of operation, switching frequencies spanning 20 kHz to 300 kHz, wide range of component values and different component types. For example, like in the case of the Si based PFC stage, within each system level design iteration, optimal inductors are designed for the PFC boost stage by iterating over various core materials, core geometries and winding parameters. The optimization is performed using an exhaustive search based numerical approach that computes the overall converter weight and line-cycle average losses for each considered design. The final design is
selected by searching for solutions that minimize losses while staying within specified weight limits. As a result of this optimization, a switching frequency of 120 kHz and continuous conduction mode (CCM) operation are selected for the bridgeless boost PFC stage. The components used in this optimized SiC based prototype of the bridgeless boost PFC stage are listed in Table 5.1. The design details of the optimized inductor and the SiC MOSFETs utilized in this bridgeless boost stage are also given in Table 5.2. The details of the other components used in bridgeless boost stage are also presented in Table 5.2.

Table 5.2: Design Details of the 6.6 kW Bridgeless Boost PFC Stage Utilizing Silicon Carbide Devices and Operating in Discontinuous Conduction Mode (DCM).

<table>
<thead>
<tr>
<th>Component</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOSFETs</td>
<td>HT-4201 module</td>
</tr>
<tr>
<td></td>
<td>1200 V, 25 mΩ, 68 A</td>
</tr>
<tr>
<td></td>
<td>Silicon Carbide MOSFET</td>
</tr>
<tr>
<td>Output capacitor</td>
<td>80 µF x 5, 850 V Film Capacitor</td>
</tr>
<tr>
<td>Coupled Inductor</td>
<td>40 µH, Ferrite core</td>
</tr>
<tr>
<td></td>
<td>Planar PCB inductor with 12 turns across 12 layers</td>
</tr>
<tr>
<td>Microcontroller</td>
<td>TI-TMS320F28069</td>
</tr>
<tr>
<td></td>
<td>32 Bit CPU, 90 MHz</td>
</tr>
</tbody>
</table>

The PFC stage has been tested up to close to its full power of 6.6 kW. Figures 5.6 and 5.7 show waveforms of the bridgeless boost stage operating at power levels of 1.6 kW and 5 kW, respectively. It can be seen that the input current is fairly sinusoidal and in phase with input voltage. The PFC stage achieves a power factor of 0.99 at 1.6 kW, demonstrating good performance as a result of new control architecture introduced and implemented in this thesis. Figure 5.3 shows the switching waveforms for the switch-node voltage, the MOSFETs drain-to-source voltage, the inductor current and the MOSFETs gate-to-source voltage. From these switching waveforms, it can be clearly observed that the converter exhibits minimal ringing and overshoot. This is due to good circuit layout and use of SiC devices packaged in modules with low lead inductances.

The zero-crossing distortion mitigation strategy presented in Section 4.2.2 is validated by the
Figure 5.6: Input voltage $v_{in}$, input current $i_{in}$ and output voltage $v_{out}$ waveforms of PFC converter with $V_{in} = 120 \, V_{rms}$, $I_{in} = 13.1 \, A_{rms}$ and $V_{OUT} = 388 \, V$.

Figure 5.7: Input voltage $v_{in}$, input current $i_{in}$ and output voltage $v_{out}$ waveforms of PFC converter with $V_{in} = 210 \, V_{rms}$, $I_{in} = 24.2 \, A_{rms}$ and $V_{OUT} = 400 \, V$.

Figure 5.8: Switching waveforms of the bridgeless boost ac-dc converter.

experimental results presented in Fig. 5.9. As can be seen from Fig. 5.9, the line current transitions across the line zero crossing without any visible distortion. The new control architectures presented in this thesis make this seem less zero crossing possible.

5.3 DAB Simulation Waveforms

Simulated input and output voltage waveforms for the dual-active bridge (DAB) converter are shown in Fig. 5.10. The slow output voltage ($v_{bst}$) regulation loop allows twice line frequency
ripple present at the input energy buffering capacitor of the DAB converter to appear at the output of the DAB. Hence, the capacitor connected at the output of the DAB converter also plays a role in energy buffering. Inductor current waveforms of the DAB converter are shown in Fig. 5.11 when the converter is operated at a switching frequency of 120 kHz. The waveforms show that the converter switches in both the primary and the secondary side H-bridges achieve ZVS.

Figure 5.10: Dual-active bridge converter input \((v_{bus})\) and output \((v_{bst})\) voltage waveforms with \(V_{bus} = 375\ V_{nom}\) and \(V_{bst} = 200\ V_{nom}\).

Figure 5.11: Dual-active bridge converter inductor current waveforms. The waveforms show that the converter achieves zero voltage switching on both primary and secondary H-bridges and switching frequency \(f_{sw} = 120\ kHz\).
Simulated input and output voltage and current waveforms of the boost stage are shown in Fig. 5.12. It can be observed that the input voltage and input current of the boost converter has twice line frequency ripple, while the output current is near constant. The fast current regulating loop of the boost converter allows the converter input port to act as a constant power sink which appears at the converter output port as a constant power source, thus eliminating twice line frequency current ripple from the output current.

Figure 5.12: Boost converter input voltage ($v_{bst}$), output voltage ($v_{batt}$), average input current ($\langle i_{bst} \rangle T_s$) and average output current ($\langle i_{batt} \rangle T_s$) waveforms with $V_{bst} = 375 \text{ V}_{\text{nom}}$, $V_{batt} = 200 \text{ V}$, $I_{bst} = 33 \text{ A}_{\text{nom}}$ and $I_{batt} = 36.4 \text{ A}_{\text{nom}}$. 
Chapter 6

Conclusions

6.1 Summary and Conclusions

This thesis presents a new architecture for an isolated level 2 on-board electric vehicle (EV) battery charger which is integrated with the EV’s drivetrain. The integration with the drivetrain is done by leveraging many of the existing stages of a highly efficient and power-dense composite-architecture-based drivetrain boost converter that acts as an interface between the traction battery and the motor drive and allows the input voltage of the motor drive to be adjusted dynamically to maximize drivetrain efficiency. This composite boost converter comprises a buck stage, a boost stage and a dc transformer (DCX) stage, providing multiple means to integrate on-board charger functionality. This is discussed in more detail in Chapter 1. This is followed by looking at multiple approaches to integrate charging functionality with the drivetrain architecture discussed in Chapter 2. Four alternative approaches to drivetrain integration are identified, analyzed and compared quantitatively in terms of added weight and charging losses. Out of the considered approaches, the selected charging architecture provides an effective tradeoff between added weight and charging losses. This drivetrain-integrated charger leverages the buck stage and part of the DCX stage of the composite boost converter, and adds only a bridgeless-boost based power factor correction (PFC) ac-dc stage, an H-bridge and a single winding to the composite boost converter, to achieve high-power on-board charging functionality without substantial additional weight. Hence, the proposed charger architecture comprises a boost PFC rectification stage, a dual-active bridge isolation stage and a boost current regulation stage.
Chapter 3 discusses design approach and control architecture of the selected architecture stages. PFC ac-dc conversion stage of the architecture is discussed in detail and is designed to operate in DCM when Si devices are used and in CCM when SiC devices are utilized. Hybrid feedforward controller is used to control the converter in DCM and feedback controller is designed for converter operating in CCM. Control architectures for dual-active bridge isolation and voltage stage and boost current regulation stage are also developed. A slow voltage regulation loop is proposed for controlling output voltage of the dual active bridge stage and the boost converter is controlled to regulate power flow into the battery.

Zero crossing distortion in the line current is a common problem experienced in single phase ac-dc converters. A control architecture that eliminates zero-crossing distortion in single-phase bridgeless boost PFC rectifiers is presented in Chapter 4. Zero-crossing distortion for bridgeless boost converter is discussed for the PFC converter operating in DCM and CCM. In DCM mode of operation, the issue is avoided by switching the converter’s active switches with the same duty cycle around the zero crossings of the line voltage. In CCM, zero crossing distortion is mitigated by sensing the ac line voltage and generating a non-rectified current reference. This eliminates zero-crossing distortion and eases requirements for the controller bandwidth. Experimental results on a 6.6 kW bridgeless boost PFC rectifier presented in Chapter 5 show very low line current distortion even in the case when a small boost inductance results in significant phase shift between the input ac line voltage and the ac line current. This is in sharp contrast with conventional controllers where such phase shift results in substantial zero-crossing distortion. The small-inductance effect is also analyzed, and its impact on the phase shift between ac line voltage and current is explained. It is shown that in the new controller architecture the phase shift can be easily mitigated using a proportional-plus-integral-squared (PI²) compensator, resulting in near-ideal rectification. The approach is verified by experimental results on a 6.6 kW single-phase bridgeless boost PFC rectifier presented in Chapter 5. Simulation results for dual-active bridge isolation stage and boost current regulation stage are also presented in Chapter 5.

Finally hybrid feedforward control architectures are developed more generally as part of this
thesis. General considerations for the synthesis of hybrid feedforward control architectures for PWM converters are developed in Appendix A. The control strategy integrates converter conversion characteristics in the synthesis of the controllers. In general, certain converter independent variables are sensed and utilized to determine the duty cycle to achieve a control objective based on the converter open-loop characteristics. The synthesis process presented in this thesis provides guidelines for selecting the converter independent variables and processing them to yield duty cycle modulation equations. The modulation equations can be implemented in the digital or analog controllers to achieve the desired control objectives.

Compared to standard feedback control techniques, the hybrid feedforward control strategy achieves certain advantages. These include simpler controller implementation, more convenient sensing and improved static and dynamic responses. Furthermore, the number of sensors employed to sense converter variables can be reduced in some situations using hybrid feedforward control architectures relative to traditional feedback control.

The hybrid feedforward approach is illustrated first through previously known examples, including feedforward control of boost PWM converters operating in CCM, and hybrid feedforward control of boost PFC rectifiers operating in DCM. The synthesis principles are then used to realize new hybrid feedforward control architectures for a battery charging application, which comprises a four-switch non-inverting buck-boost converter to implement the PFC functionality, followed by a buck converter to regulate battery charging power. The simplicity of implementing the proposed hybrid feedforward control on the four switch buck-boost converter to achieve PFC functionality is demonstrated. It is shown that the converter naturally transitions between buck and boost modes of operation by employing the developed control strategy. Experimental results are also provided for the considered hybrid feedforward controlled converters.

6.2 Recommendations for Future Work

Some aspects of the ideas introduced here can be taken further and developed more completely as an extension of this work. The charger architecture introduced in this thesis is predicted to have
high efficiency and minimal added weight based on the loss and weight models developed in this thesis. However, only the PFC stage of the charger was built and tested. Therefore, it would be valuable to integrate the PFC stage with the composite boost drivetrain converter and test the performance of the overall charging system. This would also test out in hardware the control architectures developed in this thesis for the two follow-on charging stages: the dual active bridge (DAB) stage used for isolation and the boost stage used for regulating the battery charging power.

The charger architecture introduced here uses a three winding transformer to integrate the charger add-on module with the drivetrain composite boost converter. When the drivetrain is operating, the charger module is turned off. Similarly, when the charger is charging the battery, the part of the composite boost converter powering the motor drive’s inverter is turned off. Hence, when using a three winding transformer, it becomes important to properly control the power flow between the three windings, so that the power only flows where intended. New switching schemes will probably need to be developed to achieve this power flow control objective. This can be a very interesting and useful area for future work.

The above described suggestions for future work will help demonstrate the effectiveness of drivetrain chargers that add very little extra weight and are highly efficient. Such technology would enable on-board level 2 chargers and help accelerate the adoption of electric vehicles.
Bibliography


Appendix A

Generalized Hybrid Feedforward Control of Pulse-Width Modulated Switching Converters

This appendix introduces hybrid feedforward control architectures more generally [11]. Fundamental theory underlying the control architectures is developed and is supported with practically important new control architectures to show how the theory can be applied to synthesize hybrid feedforward controllers.

A.1 Introduction

Block diagram of a pulse-width modulated (PWM) switching power converter is shown in Fig. A.1a, where the converter has an input $u_{in}$, an output $u_{out}$, and is controlled by duty-cycle command $d$. Conventionally, a feedback controller is designed around the power converter to regulate the output and achieve a control objective $u_{out} = u_{ref}$, where the reference command $u_{ref}$ can be an independent signal or can depend upon converter averaged voltage(s) or current(s). In the

![Diagram](image)

Figure A.1: Approaches to control of pulse width modulated (PWM) switching converters: (a) Conventional feedback controller architecture and (b) Hybrid feedforward controller architecture. Instead of sensing the output and comparing it with a reference command, the hybrid feedforward controller senses converter independent variables $v_x, i_y$ and processes them directly using the converter open loop characteristics to determine the switch duty cycle $d$ so as to achieve a control objective $u_{out} = u_{ref}$. 
conventional feedback controller, output $u_{out}$ is sensed and compared with the reference $u_{ref}$. The error between the two is then passed to a compensator $G_c(s)$. This then generates the duty cycle command $d$ for a pulse-width modulator, which produces the converter switch control signals. While the compensator parameters are typically determined based on a converter averaged small-signal model and standard frequency-domain control-loop design techniques, the conventional controller architecture does not make use of the converter intrinsic characteristics or topological relationships.

In contrast, in feedforward controller architectures such as the well-known feedforward control of buck converters [19, 20], the duty cycle is determined, at least in part, based on the converter topology and conversion characteristics. This concept can be generalized as shown in Fig. A.1b. In contrast to the conventional feedback control architecture shown in Fig. A.1a, in Fig. A.1b the converter independent variables $\{v_x, i_y\}$ are sensed and processed directly to generate the duty cycle command $d = f(v_x, i_y, u_{ref})$ so as to achieve the desired control objective $u_{out} = u_{ref}$. The relationship $d = f(v_x, i_y, u_{ref})$ is derived based on the converter open loop steady-state characteristics.

The control strategy involves sensing the converter independent variables, which may involve duty-cycle dependent as well as duty-cycle independent variable(s). As a result, the control architecture shown in Fig. A.1b may include feedforward and feedback loops, and is therefore named hybrid feedforward control architecture. The control architecture is differentiated from pure feedforward control architecture, in which the duty cycle command is determined solely based on input(s) (duty cycle independent converter variable(s)). In this thesis, pure feedforward control is considered a special case of hybrid feedforward control.

Several previously reported control approaches can be identified as examples of the hybrid feedforward control architecture for PWM converters. Examples include feed-forward control of dc-dc PWM converters [21], non-linear carrier control for power factor correction (PFC) converters operating in continuous conduction mode (CCM) [22, 23], open loop control of boost PFC converters operating in discontinuous conduction mode (DCM) [9, 10], as well as constant duty-cycle control of buck-boost, flyback and Cuk PFC converters operating in DCM [24]. All of these control strategies can be considered particular implementations of the architecture shown in Fig. 1(b). The hybrid
feedforward control architectures can achieve certain advantages over the conventional feedback architecture. Since there is a degree of freedom in choosing converter independent variables, reduced or more convenient sensing can be employed \[9,10,21–24\]. Furthermore, hybrid feedforward approaches can lead to simpler controller implementations, and improved static and dynamic regulation. Additionally, it is also possible to embed hybrid feedforward control loops inside feedback loops, which can achieve benefits of both.

The purpose of this work is to show how the hybrid feedforward control architecture can be generalized and synthesized systematically. To introduce and illustrate the concepts, Section A.2 describes two known examples of hybrid feedforward control architecture. This is followed in Section A.3, by general considerations for synthesis of hybrid feedforward control architectures for PWM converters. The proposed methodology is then used to develop a new hybrid feedforward control architecture for an offline battery charging system in Section A.4. Finally, Section A.5 concludes this paper.

A.2 Example of Hybrid Feedforward Control Architecture

This section briefly reviews two previously reported examples of control approaches that can be classified as hybrid feedforward control architectures: feedforward control of boost PWM converters operating in continuous conduction mode \[21,25\] in Section A.2.1, and hybrid feedforward control of boost power factor correction (PFC) rectifiers operating in discontinuous conduction mode \[9,10\] in Section A.2.2.

A.2.1 Feedforward Output Voltage Regulation in CCM Boost Converters

A simple example of pure feedforward control, which can be considered a special case of hybrid feedforward control, is the output voltage regulation in a boost converter operating in CCM. The control objective is \(v_{out} = V_{ref}\). The ideal open-loop conversion ratio of the boost converter is:

\[
\frac{v_{out}}{v_{in}} = \frac{1}{1 - d}. \tag{A.1}
\]
Figure A.2: Feedforward output voltage regulation of boost converter. Open loop characteristics of the boost converter are utilized to achieve the output voltage regulation. Independent converter variable, input voltage $v_{in}$ is sensed and processed to generate the converter duty cycle $d$.

This expression can alternatively be written as:

$$d = 1 - \frac{v_{in}}{v_{out}}. \tag{A.2}$$

If the controller is designed so that in each switching interval the duty cycle command is determined as:

$$d = f(v_{in}, V_{ref}) = 1 - \frac{v_{in}}{V_{ref}}. \tag{A.3}$$

it follows from A.2 and A.3 that the output voltage $v_{out} = V_{ref}$ is ideally obtained in steady state. This simple hybrid feedforward control architecture is shown in Fig. A.2, where it can be observed that the controller utilizes the converter open loop characteristic to determine the duty cycle that results in the desired output voltage. A simple circuit implementation of the control relationship A.3 has been described in [25]. Note that the functional relationship A.3 is based on the ideal converter steady-state characteristic, neglecting losses and converter dynamics. Since only a duty-cycle independent variable $V_{in}$ is sensed, this is an example of pure feedforward control. The control architecture can be embedded inside a feedback loop to achieve better output voltage regulation, while having improved input voltage disturbance rejection.

A.2.2 Hybrid Feedforward Control of DCM Boost PFC Rectifiers

A second example considered here is a hybrid feedforward controller for a boost PFC rectifier. This section also contrasts it with a traditional feedback control implementation. The conventional
Figure A.3: Power factor correction functionality (PFC) implemented using two different control approaches: (a) PFC functionality implemented using conventional average current mode feedback control and (b) combined hybrid feedforward-feedback control. The feedback control architecture is not affected by the converter topology, as the converter topology and its circuit element values are only used to determine the compensator parameters. In contrast, hybrid feedforward control utilizes the converter topology and characteristics in the control architecture. The converter independent variables, input voltage $v_{in}$ and output voltage $v_{out}$, are utilized to synthesize the hybrid feedforward controller.

feedback control architecture is shown in Fig. A.3a. The control objective is to shape average input current to follow the input voltage. Hence, traditional average current mode control is realized by sensing the input voltage and using it to generate a reference for the sensed average input current. The error between the two signals is processed by a current-loop compensator, which generates the duty cycle command for the power converter. Output dc voltage regulation is realized by sensing the output voltage, comparing it with a reference, and passing the error through a voltage-loop compensator. The output of the voltage-loop compensator slowly modulates the amplitude of the input current. As illustrated in Fig. A.3a, this feedback control architecture remains unchanged, regardless of the converter topology employed. The converter topology and its circuit elements are only used to determine the compensator parameters. Furthermore, one can observe that although input voltage, input current and output voltage of the converter are related to each other by converter intrinsic relationships, these relationships are not utilized within the control architecture. This results in greater than minimum number of sensors required to implement the control functionality.

Figure A.3b shows an implementation of the hybrid feedforward control architecture for a DCM boost converter, where the converter intrinsic relationships are used to achieve the control objectives while eliminating the need to sense the input current [9,10]. In discontinuous conduction
mode, average input (inductor) current can be expressed as [8]:

\[
\langle i_{\text{in}} \rangle_{T_s} = \frac{T_s}{2L} \frac{d^2}{1 - \frac{v_{\text{in}}}{v_{\text{out}}}} v_{\text{in}}
\]  

(A.4)

where \( \langle i_{\text{in}} \rangle_{T_s} \) is the average input current, \( v_{\text{in}} \) is the full-wave rectified input line voltage, \( v_{\text{out}} \) is the output voltage, \( T_s \) is the switching period, \( d \) is the duty cycle, and \( L \) is the boost inductance. From (A.4), one can observe that average input current can be predicted by sensing input voltage and output voltage of the converter, which removes the necessity to sense the input current. To implement this control functionality, one can modulate the duty cycle such that the first term (in the parenthesis) of (A.4) becomes constant, making input current proportional to input voltage \( \langle i_{\text{in}} \rangle_{T_s} = v_{\text{in}}/R_e \). Hence, the required duty cycle command can be determined from A.4 and expressed as:

\[
d = \sqrt{\frac{2L}{R_e T_s}} (1 - \frac{v_{\text{in}}}{v_{\text{out}}}),
\]  

(A.5)

where \( R_e \) is the emulated input resistance of the converter. From (A.5), one can observe that sensing only input voltage and output voltage of the converter are required to achieve desired modulation of duty cycle. To regulate the output voltage, one can utilize the sensed output voltage to implement conventional outer voltage control loop, which modulates amplitude of the input current. The resulting control architecture is shown in Fig. A.3b, highlighting the idea that known converter topology and converter characteristics can be used in synthesizing a controller. Figure A.4 shows experimental results for this control approach demonstrating good control performance.

Some benefits of combining hybrid feedforward and conventional feedback architectures are also illustrated by this example. Hybrid feedforward controller results in reducing the number of sensors and simplifying the hardware implementation of the control circuit, while the outer conventional feedback architecture results in tight output voltage regulation, correcting for any inaccuracies in extracting converter parameters \((L, T_s)\).
A.3 General Considerations for Synthesis of Hybrid Feedforward Control

This section presents general considerations for synthesis of hybrid feedforward control architectures. The DCM boost PFC example shown in Fig. A.3b of Section A.2.2 is used to illustrate the main points. A general block diagram for a hybrid feedforward controller around a switching power converter is shown in Fig. A.1b. The control objective is $u_{\text{out}} = u_{\text{ref}}$, where the converter output $u_{\text{out}}$ is the converter variable and is typically an average voltage or current of the PWM converter, and the reference command $u_{\text{ref}}$ can be either be an independent signal or can depend upon converter averaged voltage(s) or current(s). As an example, in the boost PFC converter shown in Fig. A.3b, output $u_{\text{out}}$ corresponds to average input current $\langle i_{\text{in}} \rangle_T$ and the control objective is:

$$u_{\text{out}} = \langle i_{\text{in}} \rangle_T = u_{\text{ref}} = \frac{v_{\text{in}}}{R_e}. \quad (A.6)$$

The hybrid feedforward control synthesis amounts to finding a suitable functional relationship \( d = f(v_x, i_y, u_{\text{ref}}) \), i.e., a way to determine the duty cycle command that achieves the control objective. Depending on the nature of the control objective, and the converter topology and its conversion characteristics, the solution may not be unique. In general, it is desirable to achieve additional objectives such as minimization of sensing requirements or simplicity of analog or digital controller implementation.

A set of converter independent variables (voltage(s) and/or current(s)) \( u^* = \{v_x^*, i_y^*\} \) averaged over a switching interval is identified first. A variable is considered independent if it cannot be determined from converter open-loop characteristics based on the knowledge of other independent
variables and duty cycle $d$. Thus in steady state any other converter variable can be expressed in terms of this necessary and sufficient set of converter independent variables $u^*$, along with the converter duty cycle $d$. As an example, in an ideal converter operating in CCM, converter voltages and currents are independent from each other, as the converter conversion ratio is independent of load current. Thus at least one converter voltage and one converter current must be identified as independent. As seen in the example of Section A.2.1, in an ideal boost converter operating in CCM, one has the freedom to choose the independent variable from input or output voltage, while the other voltage variable can then be considered dependent. If, for example, the input voltage is identified as independent, then the output voltage can be predicted using the converter conversion ratio as discussed in Section A.2.1. With the knowledge of duty cycle and input voltage, one can still say nothing about any currents in an ideal converter operating in CCM. Thus one of the converter currents, for example input current $i_{in}$, must be identified as another independent variable.

As another example, in an ideal converter operating in DCM, converter voltages and currents are not independent from each other as the converter conversion ratio depends upon the load current. Thus it is possible to identify a set of independent variables consisting of only voltage(s) or only current(s), or a mix of voltages and currents. In the example of Section A.2.2, input voltage $v_{in}$ and output voltage $v_{out}$ of the converter are identified as independent variables $u^* = \{v_{in}, v_{out}\}$. It should be further noted that the independent variables need not necessarily be converter state variables. For example, average switch current can also be identified as an independent variable, as is usually done in non-linear carrier control [22]. This provides more freedom in making a choice of the independent variables, as sensing some of these is necessary to implement a hybrid feedforward controller, allowing designers to employ more convenient sensing.

Once a suitable set of independent variables $u^* = \{v_{x^*}, i_{y^*}\}$ is identified, any other converter variable can be expressed based on the converter steady-state characteristics in terms of the variables in $u^*$ and the duty cycle $d$. The next step is to represent the converter output $u_{out}$ as a function of a subset of independent variables, $\{v_x, i_y\} \subseteq \{v_{x^*}, i_{y^*}\}$ and duty cycle $d$ using the converter intrinsic
conversion characteristics:

\[ u_{\text{out}} = g\left(\{v_x, i_y\}, d\right), \tag{A.7} \]

and combine this with the control objective:

\[ u_{\text{out}} = u_{\text{ref}}, \tag{A.8} \]

to arrive at the relationship:

\[ g\left(\{v_x, i_y\}, d\right) = u_{\text{ref}}, \tag{A.9} \]

which determines how duty cycle \( d \) should be modulated in order to achieve the desired control objective (A.8). Solving (A.9) for \( d \) yields the duty cycle command:

\[ d = f(\{v_x, i_y\}, u_{\text{ref}}), \tag{A.10} \]

which is shown in the hybrid feedforward controller block diagram in Fig. A.1b.

As already mentioned, the reference command \( u_{\text{ref}} \) can either be an independent signal or dependent upon converter variables. If \( u_{\text{ref}} \) is an independent signal, then \( u_{\text{out}} \) in (A.7) must involve dependence on duty cycle \( d \). In the case \( u_{\text{ref}} \) is dependent upon converter variables, possibly including \( d \), this dependence should be included in (A.9) and (A.10). In all cases, the final expressions (A.9) and (A.10) must involve a subset of converter independent variables, and duty cycle \( d \). In general, the independent variables that appear in the duty cycle modulation expressions (A.9) and (A.10) are the variables that need to be sensed.

To illustrate the hybrid feedforward synthesis method represented by (A.7)-(A.10), consider again the DCM boost example of Section A.2.2. In this case, \( u^* = u = \{v_{\text{in}}, v_{\text{out}}\} \), \( u_{\text{out}} = \langle i_{\text{in}} \rangle_{T_s} = u_{\text{ref}} = \frac{v_{\text{in}}}{R_e} \), and (A.9) becomes:

\[ \langle i_{\text{in}} \rangle_{T_s} = g(v_{\text{in}}, v_{\text{out}}, d) = \left( \frac{T_s}{2L} \frac{d^2}{1 - \frac{v_{\text{in}}}{v_{\text{out}}}} \right) v_{\text{in}} = \frac{v_{\text{in}}}{R_e}, \tag{A.11} \]

which yields the following duty cycle modulation equation.

\[ d = f(v_{\text{in}}, v_{\text{out}}, \frac{v_{\text{in}}}{R_e}) = \sqrt{\left( \frac{v_{\text{in}}}{R_e} \frac{2L}{T_s v_{\text{in}} (1 - \frac{v_{\text{in}}}{v_{\text{out}}})} \right)}. \tag{A.12} \]
It can be observed that the duty cycle expression (A.12) depends upon converter input voltage, output voltage and the reference command. The dependence of duty cycle command on the reference command can be further simplified here, as the reference command depends upon one of the converter variables (input voltage),

\[ d = f(v_{in}, v_{out}, \frac{1}{R_e}) = \sqrt{\frac{2L}{R_e T_s}} \left(1 - \frac{v_{in}}{v_{out}}\right). \]  

(A.13)

It should be noted here that the duty cycle modulation equation (A.9) or (A.10) can be implemented in multiple ways, which may involve analog or digital implementations. Direct duty cycle modulation is possible by programming the duty cycle modulation equation (A.10) into a microcontroller. Also relatively simple analog circuits can be designed to solve (A.9), as demonstrated in [9,10,21–24].

A.4 New Hybrid Feedforward Control Architecture

The general considerations discussed in Section A.3 can be used to synthesize new hybrid feedforward control architectures. Consider an offline battery charger application shown in a block diagram form in Fig. A.5a. The first stage is an ac-dc PFC rectifier, followed by a second stage that regulates the battery charging power. Figure A.5b shows a large signal averaged model of this architecture [8], where the ac-dc conversion stage is modelled by a loss-free resistor, followed by the power regulation stage which behaves as a power sink at its input port and a power source at its output port.

To realize the PFC ac-dc conversion stage, a four switch buck-boost converter is selected as shown in Fig A.6. This particular converter topology enables the output voltage of the PFC stage to
be lower than the peak input voltage, as the converter operation can switch between buck or boost
modes during different periods of the line cycle. Hence, the step-down conversion ratio required
from the second stage can be reduced. This can be advantageous if the battery voltage is much
lower than the peak input voltage.

The PFC converter is designed to operate in CCM over its entire operating range. To realize
hybrid feedforward control, the inductor current $i_{l_1}$ and the output (bus) voltage $v_{bus}$ are selected
as the converter independent variables. The control objective for achieving PFC functionality is
$u_{out} = \langle i_{in} \rangle_{T_s} = u_{ref} = \frac{v_{in}}{R_e}$, where $R_e$ is the emulated resistance of the converter. When
the circuit is operating in buck mode, the inductor current can be represented in terms of the
independent variables as:

$$u_{out} = \langle i_{in} \rangle_{T_s} = g(\langle i_{l_1} \rangle_{T_s}, d_{buck}) = d_{buck} \langle i_{l_1} \rangle_{T_s}.$$  \hspace{1cm} (A.14)

Here, $d_{buck}$ is the duty cycle of the buck stage of the converter, as shown in Fig. A.6. This can be
equated to the reference command, following the guidelines presented in Section A.3:

$$d_{buck}.\langle i_{l_1} \rangle_{T_s} = u_{ref} = \frac{v_{in}}{R_e} = \frac{v_{bus}}{d_{buck}R_e}.$$  \hspace{1cm} (A.15)

Note that in this case the reference command ($\frac{v_{in}}{R_e}$) is expressed in terms of converter independent
variable ($v_{bus}$). Equation (A.15) can be used to derive the desired duty cycle modulation equation
(A.10) for buck mode operation:

$$d_{buck} = f(\langle i_{l_1} \rangle_{T_s}, \frac{1}{R_e}) = \sqrt{\frac{v_{bus}}{R_e \langle i_{l_1} \rangle_{T_s}}}.$$  \hspace{1cm} (A.16)
Similarly, when the converter is operating in boost mode, the inductor current can be represented in terms of the independent variables as:

\[ u_{\text{out}} = \langle i_{\text{in}} \rangle T_s = g(\langle i_{\text{1}} \rangle T_s, d_{\text{boost}}) = \langle i_{\text{1}} \rangle T_s. \]  

(A.17)

Here, \( d_{\text{boost}} \) is the duty cycle of the boost stage of the converter, as shown in Fig. A.6. This can be equated to the reference command:

\[ \langle i_{\text{1}} \rangle T_s = u_{\text{ref}} = \frac{v_{\text{in}}}{R_e} = \frac{d_{\text{boost}} v_{\text{bus}}}{R_e}. \]  

(A.18)

Equation (A.18) yields the desired duty cycle modulation equation (A.10) of the form:

\[ d_{\text{boost}} = f(\langle i_{\text{1}} \rangle T_s, v_{\text{bus}}, \frac{1}{R_e}) = 1 - \frac{R_e}{v_{\text{bus}}} \langle i_{\text{1}} \rangle T_s. \]  

(A.19)

If saturation limits of 0 and 1 are imposed on the duty cycle expressions (A.16) and (A.19), and these two duty cycles are used to control the buck and boost stages, then the converter automatically transitions between the buck and boost modes over its entire range of operation, as shown in Fig. A.7.

![Figure A.7: Four switch buck-boost duty cycle commands \( d_{\text{buck}} \) and \( d_{\text{boost}} \) plotted over a half line cycle. The two duty cycle commands are non-overlapping, which allows the converter to achieve smooth transition between buck and boost modes over its entire range of operation.](image)

A comparison of the hybrid feedforward control architecture with the traditional feedback control architecture implemented on this converter reveals several benefits. Significant challenges would appear when one attempts to achieve buck and boost mode transition using traditional feedback architecture. The compensator needs to be designed for two different modes of operation. In contrast, the designed hybrid feedforward controller naturally transitions between the buck and
boost modes of operation. Furthermore, the traditional feedback controller implementation requires employing an input voltage sensor, an output voltage sensor and an inductor current sensor. On the other hand, the hybrid feedforward controller implementation removes the need for input voltage sensing; thus easing hardware implementation. Hence, the hybrid feedforward control architecture in this application simplifies both the controller and the hardware implementation.

Experimental results for this hybrid feedforward controller implementation are shown in Fig. A.8. Component values and switching frequency of the converter are listed in Table A.1. It can be observed that the converter transitions smoothly between the buck and boost modes, and the hybrid feedforward control implementation achieves good performance.

Table A.1: Four switch buck boost converter component parameters.

<table>
<thead>
<tr>
<th>Switching Frequency</th>
<th>Inductance [L]</th>
<th>Capacitance [C]</th>
</tr>
</thead>
<tbody>
<tr>
<td>100 kHz</td>
<td>300 µH</td>
<td>250 µF</td>
</tr>
</tbody>
</table>

Figure A.8: Input voltage ($v_{in}$), input current ($i_{in}$) and output voltage ($v_{out}$) waveforms of four stage buck boost converter acting as PFC stage with Input voltage of 120V$_{rms}$, input current of 7.9A$_{rms}$ and output voltage of 110V. Lag in input current with respect to input voltage is due to filtering capacitor connected at the input to filter out switching current from line current.

Hybrid feedforward control can also be implemented for the power regulation stage of the battery charger. To realize this stage, a synchronous buck converter operating in CCM is selected, as shown in Fig. A.6. The input voltage of the buck converter has twice line frequency ripple because of the finite size of the energy buffering capacitor at the output of the previous stage. Furthermore,
battery voltage is also subject to change depending on the state of charge of the battery. An ideal power regulator should maintain constant charging power in the presence of these disturbances. Here, the input voltage of the buck converter $v_{bus}$ (output voltage of the PFC stage) and its inductor current $\langle i_l \rangle_{T_s}$ are chosen as the independent variables. The desired control objective in this case is $P_{batt} = v_{batt}i_{batt} = P_{ref}$, where $P_{ref}$ is the desired battery charging power, $v_{batt}$ is the battery voltage and $i_{batt}$ is the battery current. To achieve this control objective, following the guidelines presented in Section A.3, output can be represented in terms of the converter independent variables as:

$$u_{out} = P_{batt} = v_{batt}i_{batt} = g(v_{bus}, \langle i_l \rangle_{T_s}, d_{batt}) = d_{batt}v_{bus}\langle i_l \rangle_{T_s}. \quad \text{(A.20)}$$

Here, $d_{batt}$ is the duty cycle of the power regulation buck converter. Expression (A.20) can be equated with the reference command:

$$d_{batt}v_{bus}\langle i_l \rangle_{T_s} = u_{ref} = P_{ref}, \quad \text{(A.21)}$$

which yields the controller duty cycle modulation equation (A.10) in the following form:

$$d_{batt} = f(\langle i_l \rangle_{T_s}, v_{bus}, P_{ref}) = \frac{P_{ref}}{v_{bus}\langle i_l \rangle_{T_s}}. \quad \text{(A.22)}$$

If the duty cycle of the converter is modulated as given by (A.22), then the desired power regulation capability can be achieved by the converter.

It can be observed from (A.3), (A.13), (A.16), (A.19) and (A.22) that operations of addition, subtraction, multiplication, division and square root may appear in duty cycle modulation equations. These operations can be implemented either digitally or using analog circuits. For digital implementations, modern microcontrollers possess the capability to process all of these operations. For analog implementations, various circuit design techniques can be employed, for example as discussed in [9,10,21–24].

A.5 Summary

This section presents a set of general considerations for the synthesis of hybrid feedforward control architectures for PWM converters. The control strategy integrates converter conversion
characteristics in the synthesis of the controllers. In general, certain converter independent variables are sensed and utilized to determine the duty cycle to achieve a control objective based on the converter open-loop characteristics. The synthesis process presented in this paper provides guidelines for selecting the converter independent variables and processing them to yield duty cycle modulation equations. The modulation equations can be implemented in the digital or analog controllers to achieve the desired control objectives.

Compared to standard feedback control techniques, the control strategy achieves certain advantages. These include simpler controller implementation, more convenient sensing and improved static and dynamic responses. Furthermore, the number of sensors employed to sense converter variables can be reduced in some situations using hybrid feedforward control architectures relative to traditional feedback control.

The hybrid feedforward approach is illustrated first through previously known examples, including feedforward control of boost PWM converters operating in CCM, and hybrid feedforward control of boost PFC rectifiers operating in DCM. The synthesis principles are then used to realize new hybrid feedforward control architectures for a battery charging application, which comprises a four-switch non-inverting buck-boost converter to implement the PFC functionality, followed by a buck converter to regulate battery charging power. The simplicity of implementing the proposed hybrid feedforward control on the four switch buck-boost converter to achieve PFC functionality is demonstrated. It is shown that the converter naturally transitions between buck and boost modes of operation by employing the developed control strategy. Experimental results are also provided for the considered hybrid feedforward controlled converters.
Appendix B

Loss and Weight Modeling of Alternative Drivetrain Integrated Charger Architectures

This appendix discusses the loss and weight models used to determine the efficiency and added weight of the drivetrain integrated charger architectures considered in this thesis. Four different architectures are evaluated in this thesis: A, B, C and D as introduced in Chapter 2. The losses considered in the case of each architecture are detailed in the subsections below. To determine the weight of an architecture, only the weight of the capacitors, inductors and heat sinks is considered, as they are the heaviest components in the converter. The weight of capacitors and inductors is determined from data sheets. The weight of heat sinks is estimated assuming that a heatsink weighing 5.1 g is needed to handle 1 W of power loss.

Figure B.1: Schematic of architecture A.
Architecture A

Architecture A is shown in Fig. B.1. The losses considered in its bridgeless boost PFC stage are:

a) Hard-switching losses in the MOSFETs
b) Conduction losses in the MOSFETs and the diodes
c) Core and winding losses in the inductor

The losses considered in the phase-shifted full bridge stage of architecture A are:

a) Conduction losses in the MOSFETs and the diodes
b) Core and winding losses in the filter and resonant inductor and the transformer

Switching losses in MOSFETs is not considered as they are assumed to be soft switching.

The loss distribution of the bridgeless boost stage and phase-shifted full bridge stage for Architecture A is given in Fig. B.2 and B.3.

To determine the weight of architecture A, the weight of the following components was con-
considered:

a) Filter Inductor: AMCC020 (Metglas)
b) Transformer: EE 80/38/20 (Ferrite)
c) Resonant Inductor: PQ 35/35 (Ferrite)
d) Filter Inductor: AMCC008 (Metglas)
e) Metallized Polypropylene Film Capacitors 360uF, 575V
f) Heat sink

**B.2 Architecture B**

Architecture B is shown in Fig. B.4. The losses considered in its rectifier stage is:

a) Conduction loss of the diodes

The losses considered in its isolated boost PFC stage are:

a) Hard-switching losses in the MOSFETs
b) Conduction losses in the MOSFETs and diodes
c) Core and winding losses in the inductor and the transformer

The losses considered in boost stage are:

a) Hard switching losses in the MOSFETs
b) Conduction loss in the MOSFETs
c) Core and winding losses in the inductor
Figure B.5: Loss distribution of power converters in architecture B of integrated charger.

Figure B.6: Loss distribution among power converters in architecture B of integrated charger.

The loss distribution of the rectifier, isolated boost PFC stage and boost stage is given in Fig. B.5 and B.6.

To determine the weight of the architecture B, the weight of following components was considered:

a) Filter Inductor: AMCC020 (Metglas)
b) Transformer: EE 80/38/20 (Ferrite)
c) Resonant Inductor: PQ 35/35 (Ferrite)
d) Metallized Polypropylene Film Capacitors 360uF, 575V
f) Heat sink

B.3 Architecture C

Architecture C is shown in Fig. B.7. The losses considered in its bridgeless boost PFC stage are:
a) Hard-switching losses in the MOSFETs
b) Conduction loss in the MOSFETs and the diode
c) Core and winding losses in the inductor

Losses considered in DCX are:

a) Conduction loss in the MOSFETs and the diode
b) Core and winding losses in the transformer and the resonant inductor

Switching losses in MOSFETs is not considered as they are assumed to be soft switching.

Losses considered in boost stage are:

a) Hard switching losses in the MOSFETs
b) Conduction losses in the MOSFETs and the diodes
c) Core and winding losses in the filter inductor

The loss distribution of bridgeless boost PFC stage, DAB stage and boost stage of the architecture C is given in Fig. B.8 and B.9.

To determine the weight of architecture C, the weight of the following components was considered:

a) Filter Inductor: AMCC020 (Metglas)
b) Transformer: EE 80/38/20 (Ferrite)
c) Resonant Inductor: PQ 35/35 (Ferrite)
e) Metallized Polypropylene Film Capacitors 360uF, 575V
f) Heat sink

### B.4 Architecture D

Architecture D is shown in Fig. B.10. The losses considered in its bridgeless boost PFC stage are:

![Schematic of architecture D.](image)

Figure B.10: Schematic of architecture D.
a) Hard-switching losses in the MOSFETs  
b) Conduction losses in the MOSFETs and the diodes  
c) Core and winding losses in the inductor  
The losses considered in the DAB stage are:  
a) Conduction losses in the MOSFETs and the diodes  
b) Core and winding losses in the transformer and the resonant inductor  
Switching losses in the MOSFETs is not considered as they are assumed to be soft switching.  
The losses considered in the boost stage of Architecture D are:  
a) Hard switching losses  
b) Conduction losses in the MOSFETs and the diodes  
c) Core and winding losses in the filter inductor  
The loss distribution of the bridgeless boost PFC stage, DAB stage and boost stage is given in Fig. B.11 and B.12.  

To determine the weight of architecture D, the weight of following components was considered:
a) Filter Inductor: AMCC020 (Metglas)

b) Transformer: EE 80/38/20 (Ferrite)

c) Resonant Inductor: PQ 35/35 (Ferrite)

e) Metallized Polypropylene Film Capacitors 360uF, 575V

f) Heat sink
Appendix c

MicroController Code for the SiC based Bridgeless Boost PFC Prototype

The controller used for the SiC based bridgeless boost PFC stage is Texas Instruments C2000 Piccolo 32-bit micro-controller TMS320F28069. This controller has a 90 MHz processor, with 16 channels of 12-bit ADC, and 8 modules of enhanced pulse-width modulator (ePWM).

The PFC comprises an H-bridge with four active switches. Each half bridge is switched in a complementary manner. Separate PWM channels are reserved for the control of each half bridge. The designations of the PWM channels are listed in Table C.1. The remainder of this Appendix contains the source code used to implement the control functionality for the bridgeless boost PFC stage operating in CCM in the selected microcontroller.

<table>
<thead>
<tr>
<th>ePWM1</th>
<th>ePWM2</th>
</tr>
</thead>
<tbody>
<tr>
<td>PFC half-bridge 1</td>
<td>PFC half-bridge 2</td>
</tr>
</tbody>
</table>

Table C.1: Microcontroller ePWM Channel Designation.

C.1 Main Processor Code

The following code segment is the microcontroller’s main processor code.

```c
// SiC PFC Firmware with F28069
// Usama Anwar
// ver. 1.1
// June 2016
```
#include "DSP28x_Project.h"     // Device Headerfile and Examples
    Include File
#include "F28069_example.h"     // Main include file
#include "Serial_comm.h"
#include "ModE_P3.h"
#include "CLAshared.h"
    // Include <stdio.h>
    // Include <stdlib.h>
#include <string.h>

#pragma CODE_SECTION(cpu_timer0_isr, "ramfuncs");
#pragma CODE_SECTION(epwm1_isr, "ramfuncs");

// Prototype statements for functions found within this file.
interrupt void adc1_isr(void);
interrupt void adc2_isr(void);
interrupt void adc3_isr(void);
interrupt void adc4_isr(void);
interrupt void adc8_isr(void);
interrupt void epwm1_isr(void);
interrupt void epwm3_isr(void);
interrupt void cpu_timer0_isr(void);
void init_CL(void);
void init_OL(void);
void Gpio_select(void);

// Global variables
Uint16 LedStatus;
Uint16 LedPeriod;
Uint16 ST;
volatile Uint16 Vinbuf[BUF_SIZE];
volatile Uint16 Vbstdoutbuf[BUF_SIZE];
volatile Uint16 Voutbuf[BUF_SIZE];
volatile Uint16 Ibstdoutbuf[BUF_SIZE];
volatile Uint16 IbkIbuf[BUF_SIZE];
volatile Uint16 Dbuf[BUF_SIZE];
volatile Uint16 irefbuf[BUF_SIZE];
volatile Uint16 Dlimitbuf[BUF_SIZE];
volatile Uint16 bk_offbuf[BUF_SIZE];
Uint16 Sampleno; // Number of samples stores in buffer
Uint16 Samplecnt;
Uint16 Vrefset, DCX_en;
Uint16 softonoff;
Uint16 Vin, Vout, Vbstdout, Ibstdout, IbkI, Iin, Du = 1575, ind = 0;
long i;
long delay = 3000000;
long delay2 = 30000;

// Compensator and reference
#pragma DATA_SECTION(Vref, "CpuToCl1MsgRAM");
#pragma DATA_SECTION(Shutdown, "CpuToCl1MsgRAM");
#pragma DATA_SECTION(start, "CpuToCl1MsgRAM"); // Voltage loop compensator
#pragma DATA_SECTION(CIPI, "CpuToCl1MsgRAM"); // Current loop compensator

int32 Vref;
int32 Shutdown = 0;
int32 start;
float32 CIPI [5] = {2.0L, -1.0L, 0.1882L, -0.3572L, 0.1695L}; // Current loop compensator parameters

// Debugging variables
#pragma DATA_SECTION(Ilsample, "Cl1ToCpuMsgRAM");
#pragma DATA_SECTION(Voutsample, "Cl1ToCpuMsgRAM");
#pragma DATA_SECTION(Vinsample, "Cl1ToCpuMsgRAM");
#pragma DATA_SECTION(debug, "Cl1ToCpuMsgRAM");
#pragma DATA_SECTION(debug2, "Cl1ToCpuMsgRAM");
#pragma DATA_SECTION(debug3, "Cl1ToCpuMsgRAM");

float32 Ilsample, Voutsample, Vinsample, debug, debug2, debug3;
Uint16 x, YY;

void main(void)
{
    start = 0;
    Uint16 cmd;
    InitSysCtrl();
    InitGpio();
    DINT;
    InitPieCtrl();
    IER = 0x0000;
    IFR = 0x0000;
    InitPieVectTable();

    //#ifdef EXAMPLE_FLASH // EXAMPLE_FLASH, if defined, is in CCS project options
    //--- Copy all Flash sections that need to run from RAM (use memcpy() from RTS library)
// Section secureRamFuncs contains user defined code that runs from
// CSM secured RAM
memcpy(&RamfuncsRunStart, &RamfuncsLoadStart, (Uint32)&
   RamfuncsLoadSize);
//--- Initialize the Flash and OTP
InitFlash();    // Initialize the Flash
//#endif

// Interrupts that are used in this example are re-mapped to
// ISR functions found within this file.
EALLOW;       // This is needed to write to EALLOW protected register
PieVectTable.TINT0 = &cpu_timer0_isr;
PieVectTable.ADCINT1 = &adc1_isr;
PieVectTable.ADCINT2 = &adc2_isr;
PieVectTable.ADCINT3 = &adc3_isr;
PieVectTable.ADCINT4 = &adc4_isr;
PieVectTable.ADCINT8 = &adc8_isr;
PieVectTable.EPWM1_INT = &epwm1_isr;
PieVectTable.EPWM3_INT = &epwm3_isr;
EDIS;        // This is needed to disable write to EALLOW protected
    registers

InitEPwmGpio_OL();

InitSci();

InitCpuTimers();  // For this example, only initialize the Cpu
    Timers
// Configure CPU-Timer 0 to interrupt every 500 milliseconds:
    // 90MHz CPU Freq, 0.5 second Period (in uSeconds)
    // LedPeriod = 500;
    // ConfigCpuTimer(&CpuTimer0, 90, (int32)LedPeriod * 1000);

// Enable ADCINT1 in PIE
PieCtrlRegs.PIEIER1.bit.INTx1 = 1; // Enable INT 10.1 in the PIE
PieCtrlRegs.PIEIER1.bit.INTx2 = 1; // Enable INT 10.2 in the PIE
PieCtrlRegs.PIEIER10.bit.INTx3 = 1; // Enable INT 10.3 in the PIE
PieCtrlRegs.PIEIER10.bit.INTx4 = 1; // Enable INT 10.4 in the PIE
PieCtrlRegs.PIEIER10.bit.INTx8 = 1; // Enable INT 10.8 in the PIE
// Enable TINT0 in the PIE: Group 1 interrupt 7
PieCtrlRegs.PIEIER1.bit.INTx7 = 1;
// Enable EPWM1_INT in the PIE: Group 3 interrupt 1
PieCtrlRegs.PIEIER3.bit.INTx1 = 1;
// Enable EPWM3_INT in the PIE: Group 3 interrupt 3
PieCtrlRegs.PIEIER3.bit.INTx3 = 1;
init_cla();

InitAdc(); // Configure ADC
Gpio_select(); ///////////////////////////////////////////////////
LedStatus = ST_LED_BLINK;
ST = OL;
IER |= M_INT1; // Enable CPU Interrupt 1
IER |= M_INT3; // Enable CPU Interrupt 3
IER |= M_INT10; // Enable CPU Interrupt 10
init_OL();
SetDBGIER(IER); // Configure the DBGIER for realtime debug
asm("CLRC INTM, DBGM"); // Enable global interrupts and realtime debug

//CpuTimer0Regs.TCR.all = 0x4000; // Use write-only instruction to set TSS bit = 0, start Timer0
Sampleno = 64;
Samplecnt = 0;
memset((void*)Vinbuf, 0, BUF_SIZE);
memset((void*)Voutbuf, 0, BUF_SIZE);
memset((void*)Vbstoutbuf, 0, BUF_SIZE);
memset((void*)IbstIbuf, 0, BUF_SIZE);
memset((void*)IbkIbuf, 0, BUF_SIZE);
memset((void*)Dbuf, 0, BUF_SIZE);
memset((void*)irefbuf, 0, BUF_SIZE);
memset((void*)Dlimitbuf, 0, BUF_SIZE);
memset((void*)bk_offbuf, 0, BUF_SIZE);

Vrefset = 0;
Vref = 0;
Vin = 0;
Vout = 0;
Vbstout = 0;
IbstI = 0;
IbkI = 0;
Shutdown = 1;
softonoff = 1;
DCX_en = 1;
start_cla();
while(1){
    //cmd = get_cmd16();
//parse_cmd(cmd);
if (start == 0 && Vin > 1344 && Vin < 1544) // initialize compensator at input voltage zero crossing {
    start = 1;
}
else if (start == 0) // turn off {
    OffEPwm1Gpio();
    OffEPwm2Gpio();
}
else if (start == 1 && Vin > 2024 && Vin < 2064) // turn on converter at input voltage zero crossing {
    InitEPwm1Gpio();
    InitEPwm2Gpio();
    start = 2;
}
else if (start == 3) // turn on manually, used for testing {
    InitEPwm1Gpio();
    InitEPwm2Gpio();
    start = 2;
}
}
}

void init_CL(void) {
    // InitEPwmGpio_CL();
    CpuTimer0Regs.TCR.all = 0x4008;
    ConfigCpuTimer(&CpuTimer0, 90, 20000);
    CpuTimer0Regs.TCR.all = 0x4000; // Use write-only instruction to set TSS bit = 0, start Timer0
    ST = CL;
    Shutdown = 1;
    Vrefset = 0;
    softonoff = 1;
    // bk_off = 2250L;
    DCX_en = 0;
    start_cla();
    EPwm1Regs.CMPA.half.CMPA = 0;
EPwm3Regs.CMPA.half.CMPA = DPWMC;
EPwm5Regs.CMPA.half.CMPA = 0;
}

void init_OL(void) {
    InitEPwmGpio_OL();
    CpuTimer0Regs.TCR.all = 0x4008;
    LedPeriod = 500;
    ConfigCpuTimer(&CpuTimer0, 90, (int32)LedPeriod * 1000);
    CpuTimer0Regs.TCR.all = 0x4000; // Use write-only instruction to set TSS bit = 0, start Timer0

    DCX_en = 1;
    stop_cla();
    ST = OL;
    EPwm5Regs.CMPA.half.CMPA = 675;
}

interrupt void cpu_timer0_isr(void)
{
    CpuTimer0.InterruptCount++;

    switch (LedStatus) {
        case ST_LED_BLINK:
            GpioDataRegs.GPATOGGLE.bit.GPIO20 = 1; // Toggle GPIO24 once per 500 milliseconds
            break;
        case ST_LED_ON:
            GpioDataRegs.GPACLEAR.bit.GPIO20 = 1;
            break;
        case ST_LED_OFF:
            default:
            GpioDataRegs.GPASET.bit.GPIO20 = 1;
    }

    // Acknowledge this interrupt to receive more interrupts from group 1
    PieCtrlRegs.PIEACK.all = PIEACK_GROUP1;
}

interrupt void adcl_isr(void) //In sampling
{
    Iin = AdcResult.ADCRESULT6;
    AdcRegs.ADCINTFLGCLR.bit.ADCINT1 = 1; //Clear ADCINT1 flag
    reinitialize for next SOC
    PieCtrlRegs.PIEACK.all = PIEACK_GROUP1; // Acknowledge interrupt to
interrupt void adc2_isr(void) //Vout sampling
{
    Vout = AdcResult.ADCRESULT4;
    AdcRegs.ADCINTFLGCLR.bit.ADCINT2 = 1; //Clear ADCINT1 flag
    PieCtrlRegs.PIEACK.all = PIEACK_GROUP1; // Acknowledge interrupt to PIE
}

interrupt void adc3_isr(void) //Vin sampling
{
    Vin = AdcResult.ADCRESULT5;
    AdcRegs.ADCINTFLGCLR.bit.ADCINT3 = 1; //Clear ADCINT1 flag
    PieCtrlRegs.PIEACK.all = PIEACK_GROUP10; // Acknowledge interrupt to PIE
}

interrupt void adc4_isr(void) //Not used
{
    AdcRegs.ADCINTFLGCLR.bit.ADCINT4 = 1; //Clear ADCINT1 flag
    PieCtrlRegs.PIEACK.all = PIEACK_GROUP10; // Acknowledge interrupt to PIE
}

interrupt void adc8_isr(void) //Not used
{
    AdcRegs.ADCINTFLGCLR.bit.ADCINT8 = 1; //Clear ADCINT1 flag
    PieCtrlRegs.PIEACK.all = PIEACK_GROUP10; // Acknowledge interrupt to PIE
}

interrupt void epwm1_isr(void)
{
}

interrupt void epwm3_isr(void)
{
}

void Gpio_select(void)
{

C.2 ADC Initialization Code

The following code segment is part of the initialization code for the ADC module.

```c
#include "F2806x_Device.h"  // F2806x Headerfile Include File
#include "F2806x_Examples.h"  // F2806x Examples Include File
#define ADC_usDELAY 1000L

// This function initializes ADC to a known state.
// NOTE: ADC INIT IS DIFFERENT ON F2806x DEVICES COMPARED TO OTHER 28X DEVICES
// *IMPORTANT*
// IF RUNNING FROM FLASH, PLEASE COPY OVER THE SECTION "ramfuncs" FROM FLASH
// TO RAM PRIOR TO CALLING InitSysCtrl(). THIS PREVENTS THE MCU FROM THROWING
// AN EXCEPTION WHEN A CALL TO DELAY_US() IS MADE.
```

```c
  GpioCtrlRegs.GPAMUX2.bit.GPIO20 = 0;  // All GPIO
  GpioCtrlRegs.GPADIR.bit.GPIO20 = 1;  // All outputs
  EDIS;
}
```
```c
void InitAdc(void) {
    extern void DSP28x_usDelay(Uint32 Count);
    Uint16 i;

    // *IMPORTANT*
    // The Device_cal function, which copies the ADC calibration values
    // from TI reserved
    // OTP into the ADCREFSEL and ADCOFFTRIM registers, occurs
    // automatically in the
    // Boot ROM. If the boot ROM code is bypassed during the debug
    // process, the
    // following function MUST be called for the ADC to function
    // according
    // to specification. The clocks to the ADC MUST be enabled before
    // calling this
    // function.
    // See the device data manual and/or the ADC Reference
    // Manual for more information.

    EALLOW;
    SysCtrlRegs.PCLKCR0.bit.ADCENCLK = 1;
    (*Device_cal)();
    EDIS;

    // To powerup the ADC the ADCENCLK bit should be set first to enable
    // clocks, followed by powering up the bandgap, reference circuitry,
    // and ADC core.
    // Before the first conversion is performed a 5ms delay must be observed
    // after power up to give all analog circuits time to power up and settle

    // Please note that for the delay function below to operate correctly the
    // CPU_RATE define statement in the F2806x_Examples.h file must
    // contain the correct CPU clock period in nanoseconds.
    EALLOW;
    AdcRegs.ADCCTL1.bit.ADCBGPWD = 1; // Power ADC BG
    AdcRegs.ADCCTL1.bit.ADCREFPWD = 1; // Power reference
    AdcRegs.ADCCTL1.bit.ADCPWDN = 1; // Power ADC
    AdcRegs.ADCCTL1.bit.ADENABLE = 1; // Enable ADC
    AdcRegs.ADCCTL1.bit.ADCREFSEL = 0; // Select internal BG
    EDIS;

    //DELAY_US(ADC_usDELAY); // Delay before converting ADC
```
channels
   for(i=0;i<0xffffa;i++) {
   }
EALLOW;
AdcRegs.ADCCTL2.bit.CLKDIV2EN = 1; // 45 MHz max
AdcRegs.ADCCTL2.bit.CLKDIV4EN = 1; // 45 MHz max
EDIS;

//DELAY_US(ADC_usDELAY);  // Delay before converting ADC channels
   for(i=0;i<0xffffa;i++) {
   }
   //AdcOffsetSelfCal();
EALLOW;
AdcRegs.ADCCTL2.bit.ADCNONOVERLAP = 0;  // Enable non-overlap mode
AdcRegs.ADCCTL1.bit.INTPULSEPOS = 0;  // ADCINT1 trips
EDIS;

InitAdcAio();

///////////////////////////////////////////////////////////////Setup ADC interrupts///////////////////////////////////////////////////////////////
EALLOW;

AdcRegs.INTSEL1N2.bit.INT1E = 1;  // Enabled ADCINT1
AdcRegs.INTSEL1N2.bit.INT1CONT = 0;  // Disable ADCINT1 Continuous mode
AdcRegs.INTSEL1N2.bit.INT1SEL = 6;  // setup EOC6 to trigger ADCINT1 to fire

AdcRegs.INTSEL1N2.bit.INT2E = 1;  // Enabled ADCINT2
AdcRegs.INTSEL1N2.bit.INT2CONT = 0;  // Disable ADCINT2 Continuous mode
AdcRegs.INTSEL1N2.bit.INT2SEL = 4;  // setup EOC4 to trigger ADCINT2 to fire

AdcRegs.INTSEL3N4.bit.INT3E = 1;  // Enabled ADCINT3
AdcRegs.INTSEL3N4.bit.INT3CONT = 0;  // Disable ADCINT3 Continuous mode
AdcRegs.INTSEL3N4.bit.INT3SEL = 5;  // setup EOC5 to trigger ADCINT3 to fire

////////////////////////////////////////////////////////////////////////Assign SOC to specific channels////////////////////////////////////////////////////////////////////////

AdcRegs.ADCSOC6CTL.bit.CHSEL = 15;  // Iin // set SOC6 channel select to ADCINB7
AdcRegs.ADCSOC4CTL.bit.CHSEL = 9;  // Vout // set SOC4 channel
C.3 EPWM Initialization Code

The following code segment is part of the initialization code for the ePWM module.

```c
#include "F2806x_Device.h"  // F2806x Headerfile Include File
```
```c
#include "F2806x_Examples.h" // F2806x Examples Include File

#define DB_half 20 //dead band value

void InitEPwmGpio_OL() {
    InitEPwmGpio(); // Initialize ePWM pins
    epwml(); // PFC Half Bridge 1
    epwm2(); // PFC Half Bridge 2
}

void InitEPwmGpio_CL() {
    OffEPwm1Gpio();
    OffEPwm2Gpio();
    OffEPwm3Gpio();
    OffEPwm4Gpio();
    OffEPwm5Gpio();
    OffEPwm6Gpio();
    OffEPwm7Gpio();
    OffEPwm8Gpio();
}

void epwml(void) // PFC Half Bridge 1
{
    EALLOW;
    EPwm1Regs.TBPRD = 375; // Set timer period
    // 120 kHz (2250 for 20kHz, 450 for 100kHz)
    EPwm1Regs.TBPHS.half.TBPHS = 0x0000; // Phase is 0
    EPwm1Regs.TBCTR = 0x0000; // Clear counter

    // Setup TBCLK
    EPwm1Regs.TBCTL.bit.CTRMODE = TB_COUNT_UPDOWN; // Up-down mode, triangular modulation
    EPwm1Regs.TBCTL.bit.PHSEN = TB_DISABLE; // Disable phase loading
    EPwm1Regs.TBCTL.bit.HSPCLKDIV = TB_DIV1; // Clock ratio to SYSCLKOUT
    EPwm1Regs.TBCTL.bit.CLKDIV = TB_DIV1;
    EPwm1Regs.TBCTL.bit.SYNCOSEL = TB_CTR_ZERO; // Synchronize out at CTR = 0

    // Setup compare
    EPwm1Regs.CMPA.half.CMPA = 200; // Arbitrary initial values
    EPwm1Regs.CMPB = 200;

    // Set actions
```
EPwm1Regs.AQCTLA.bit.CAU = AQ_SET; // SET PWM1A on CAU
EPwm1Regs.AQCTLA.bit.CAD = AQ_CLEAR; // RESET PWM1A on CAD
EPwm1Regs.AQCTLB.bit.CAU = AQ_CLEAR; // RESET PWM1B on CAU
EPwm1Regs.AQCTLB.bit.CAD = AQ_SET; // SET PWM1B on CAD
EPwm1Regs.CMPCTL.bit.SHDWAMODE = 0x0;
EPwm1Regs.CMPCTL.bit.LOADAMODE = 0x01; // Load duty cycle register on TBCLK=TBPRD
EPwm1Regs.CMPCTL.bit.SHDWBMODE = 0x0;
EPwm1Regs.CMPCTL.bit.LOADBMODE = 0x01; // Load duty cycle register on TBCLK=TBPRD
EPwm1Regs.DBCTL.bit.OUT_MODE = DB_FULL_ENABLE; //Enable dead band
EPwm1Regs.DBCTL.bit.HALFCYCLE = 1; // enable high resolution DB
EPwm1Regs.DBCTL.bit.POLSEL = DB_ACTV_HIC; // Active high complementary mode
EPwm1Regs.DBCTL.bit.IN_MODE = DBA_ALL; //EPWMA is source for both rising and falling edge delay
EPwm1Regs.DBRED = DB_half; // Dead-band on rising edge (specify value of deadband here)
EPwm1Regs.DBFED = DB_half; // Dead-band on falling edge
EPwm1Regs.ETSEL.bit.SOCAEN = 1; // Enable SOC on A group
EPwm1Regs.ETSEL.bit.SOCASEL = 1; // Trigger sampling when SOC = 0, Trigger SOCA which triggers ADC Sampling
EPwm1Regs.ETPS.bit.SOCAPRD = 1; // Generate pulse on 1st event
EPwm1Regs.ETSEL.bit.INTSEl = 1; // Generate interrupt at zero of counter
EPwm1Regs.ETSEL.bit.INTEN = 1; // Enable Interrupt
EPwm1Regs.ETPS.bit.INTPRD = 1; // Generate interrupt on 1st event
EDIS;
}

void epwm2(void) //PFC Half Bridge {
EALLOW;
EPwm2Regs.TBPRD = 375; // Set switching frequency to 120 kHz
EPwm2Regs.TBPHS.half.TBPHS = 0x0000; // Phase is 0
EPwm2Regs.TBCTR = 0x0000; // Clear counter
// Setup TBCLK
EPwm2Regs.TBCTL.bit.CTRMODE = TB_COUNT_UPDOWN;  // Count up
EPwm2Regs.TBCTL.bit.PHSEN = TB_DISABLE;  // Disable phase
  loading
EPwm2Regs.TBCTL.bit.HSPCLKDIV = TB_DIV1;  // Clock ratio to
  SYSCLKOUT
EPwm2Regs.TBCTL.bit.CLKDIV = TB_DIV1;
EPwm2Regs.TBCTL.bit.SYNCOSEL = TB_CTR_ZERO;  // Synchronize out at
  CTR = 0

// Setup compare
EPwm2Regs.CMPA.half.CMPA = 200;
EPwm2Regs.CMPB = 200;

// Set actions
EPwm2Regs.AQCTLA.bit.CAU = AQ_SET;  // SET PWM1A on CAU
EPwm2Regs.AQCTLA.bit.CAD = AQ_CLEAR;  // RESET PWM1A on Zero
EPwm2Regs.AQCTLB.bit.CAU = AQ_CLEAR;  // RESET PWM1B on
  CAU
EPwm2Regs.AQCTLB.bit.CAD = AQ_SET;  // SET PWM1B on Zero

// Set duty cycle update
EPwm1Regs.CMPCTL.bit.SHDWAMODE = 0x0;
EPwm1Regs.CMPCTL.bit.LOADAMODE = 0x01;  // Load duty cycle register on
  TBCLK=PRD
EPwm1Regs.CMPCTL.bit.SHDWBMODE = 0x0;
EPwm1Regs.CMPCTL.bit.LOADBMODE = 0x01;  // Load duty cycle register on
  TBCLK=PRD

// Active high complementary PWMS - Setup the deadband
EPwm2Regs.DBCTL.bit.OUT_MODE = DB_FULL_ENABLE;
EPwm2Regs.DBCTL.bit.HALFCYCLE = 1;  // enable high resolution DB
EPwm2Regs.DBCTL.bit.POLSEL = DB_ACTV_HIC;  // Active High
  complementary
EPwm2Regs.DBCTL.bit.IN_MODE = DBA_ALL;
EPwm2Regs.DBRED = DB_half;  // Dead band value
EPwm2Regs.DBFED = DB_half;  // Dead band value
EDIS;
C.4 CLA Control Code

The following code segment is the code for the CLA (Control Law Accelerator) which is the main controller code.

```c
// This file carries the code used for control of PFC converter
//Author: Usama Anwar

#include "CLAShared.h" // includes all of the shared variables (shared between
// C28x C source and CLA assembly code)
#include "F28035_example.h" // Main include file
#include "DSP283x_Cla.h"
#include "DSP2803x_Device.h"
#include "DSP2803x_Adc.h" // ADC Registers
#include "DSP2803x_EPwm.h" // Enhanced PWM
#include "F28069_example.h" // Main include file
#include "DSP28x_Project.h" // Device Headerfile and Examples

#include "CLASHared.h"

#pragma DATA_SECTION(VPI, "ClaDataRam0");
#pragma DATA_SECTION(Errold, "ClaDataRam0");
#pragma DATA_SECTION(Errold2, "ClaDataRam0");
#pragma DATA_SECTION(Vcold, "ClaDataRam0");
#pragma DATA_SECTION(Vcold2, "ClaDataRam0");
#pragma DATA_SECTION(Vvold, "ClaDataRam0");
#pragma DATA_SECTION(Errvold, "ClaDataRam0");
#pragma DATA_SECTION(test, "ClaDataRam0");

volatile float32 VPI[3];
volatile float32 test;
volatile float32 Errold, Errold2;
volatile float32 Vcold, Vcold2;
volatile float32 Vvold, Errvold;

__interrupt void Cla1Task1 ( void )
{
    //Define variables (These variables are flashed out after CLA operation)

    float32 Ilsample, Voutsample, Vinsample;
    float32 Il, Vout, Vin;
    float32 Iref, Err, Vc;
    float32 Errv, Vv;
```
float32 Da, Db;

// use debug, debug2, debug3 for debugging. These are shared variables.

// Samples
Ilsample = (float32)AdcResult.ADCRESULT6; // Inductor current
Vinsample = (float32)AdcResult.ADCRESULT5; // Input voltage
Voutsample = (float32)AdcResult.ADCRESULT4; // Output voltage

// Offset Removal
Ii = 2060L - Ilsample;
Vin = 2044L - Vinsample;
Vout = Voutsample - 12L;

if (start == 2) // Start the PFC converter
{
    //%%%% outer voltage loop
    Errv = 1704L - Vout;
    Vv = Vvold + 0.00099708L*Errv - 0.00099703L*Errvold;
    // Saturate compensator output
    if (Vv > 1.7L)
        Vv = 1.7L;
    else if (Vv < 0.4L)
        Vv = 0.4L;
    // Change old values
    Errvold = Errv;
    Vvold = Vv;

    //%%%% Inner current loop
    // Iref = Vv*Vin; // Use with outer voltage loop
    Iref = 0.36L*Vin; // Use without outer voltage loop, constant R_e
    Err = Iref - Ii;
    Vc = 2.0L*Vcold - 1.0L*Vcold2 + 0.03274L*Err - 0.06379L*Errold + 0.03107L*Errold2; // inner current loop compensator for all powers
    // Vc = 2.0L*Vcold - 1.0L*Vcold2 + 0.1882L*Err - 0.3572L*Errold + 0.1695L*Errold2; // compensator for low power
    // Vc = 2.0L*Vcold - 1.0L*Vcold2 + 0.05169L*Err - 0.09965L*Errold + 0.04803L*Errold2; // compensator for medium power
    // Vc = Vcold + 0.02583L*Err - 0.02517L*Errold; // use for dc testing over whole operating range
    // Vc = CIPI[0]*Vcold + CIPI[1]*Vcold2 + CIPI[2]*Err + CIPI[3]*Errold + CIPI[4]*Errold2; // change values

    // Change old values
    Errold2 = Errold;
Errold = Err;
Vcold2 = Vcold;
Vcold = Vc;

// Compute duty cycle from control signal
if (Vc > 375)
    Vc = 375;
else if (Vc < -375)
    Vc = -375;

Da = 375 + Vc;
Db = 375 - Vc;

// Duty cycle saturation limits
if (Da<0)
    Da = 0;
else if (Da>375)
    Da = 375;

if (Db<0)
    Db = 0;
else if (Db>375)
    Db = 375;

// Duty cycle saturation limits
if (start == 0) // Initialize compensator
{
    // initialize inner current loop. Start the compensator from zero crossing
    Vc = 0;
    Vcold = 0;
    Vcold2 = 0;
    Errold = 0;
    Errold2 = 0;

    // initialize outer voltage loop
    Vv = 0.847;
}

//Output duty cycles
// D corresponds to low side duty cycle
//Da: Positive side half bridge
//Db: Negative side half bridge
EPwm1Regs.CMPA.half.CMPA = Db;
EPwm2Regs.CMPA.half.CMPA = Da;

__interrupt void ClalTask2 ( void )
{

__interrupt void ClaiTask3 ( void )
{
}

__interrupt void ClaiTask4 ( void )
{
}

__interrupt void ClaiTask5 ( void )
{
}

__interrupt void ClaiTask6 ( void )
{
}
Appendix D

Bridgeless Boost Converter Prototype Boards

This appendix presents the layouts of the PCB boards used in the silicon and silicon carbide prototypes of the bridgeless boost converter PFC converters. The largest components on the boards include the boost input inductor and the twice-line-frequency energy buffering capacitors.

D.1 Bridgeless Boost Converter with Si Devices

Figure D.1 shows a photograph of the Si based bridgeless boost PFC converter prototype. This prototype uses a boost inductor utilizing an amorphous alloy core.

Figure D.1: Photograph of the Si based bridgeless boost PFC converter.
D.2 Bridgeless Boost Converter with SiC Devices

Figure D.2 shows a photograph of the SiC based bridgeless boost converter prototype. The converter comprises four separate PCB boards: (a) Power board containing the major pathway for power flow shown in Fig. D.3, (b) driver board containing gate drivers and switches supporting circuitry shown in Fig. D.4, (c) sensor board containing input voltage and input current sensors shown in Fig. D.5 and (4) planar inductor board containing windings of the inductor shown in Fig D.6.

Figure D.2: Photograph of the SiC based bridgeless boost PFC converter. The converter utilizes four separate PCB boards: (1) Power board containing major pathway for power flow (2) Driver board containing gate drivers and switches supporting circuitry (3) Sensor board containing input voltage and input current sensors, and (4) Planar inductor board containing windings of the planar inductor.
Figure D.3: Layout of the power board for the SiC based bridgeless boost PFC converter.

Figure D.4: Layout of the gate driver board for the SiC based bridgeless boost PFC converter.

Figure D.5: Layout of the sensor board for the SiC based bridgeless boost PFC converter.
Figure D.6: Layout of the planar inductor board for the SiC based bridgeless boost PFC converter. The planar inductor is a coupled inductor structure with 12 layers.