An EMI-Compliant and Automotive-Rated 48 V to Point-of-Load Dickson-Based Hybrid Switched-Capacitor DC-DC Converter

Sahana Krishnan

Electrical Engineering and Computer Sciences
University of California, Berkeley

Technical Report No. UCB/EECS-2024-79
http://www2.eecs.berkeley.edu/Pubs/TechRpts/2024/EECS-2024-79.html

May 10, 2024
An EMI-Compliant and Automotive-Rated 48 V to Point-of-Load Dickson-Based Hybrid Switched-Capacitor DC-DC Converter

by

Sahana Krishnan

A thesis submitted in partial satisfaction of the requirements for the degree of

Master of Science

in

Engineering - Electrical Engineering and Computer Sciences

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Robert Pilawa-Podgurski, Chair
Assistant Professor Jessica Boles

Spring 2024
An EMI-Compliant and Automotive-Rated 48 V to Point-of-Load Dickson-Based Hybrid Switched-Capacitor DC-DC Converter

Copyright 2024
by
Sahana Krishnan
Abstract

An EMI-Compliant and Automotive-Rated 48 V to Point-of-Load Dickson-Based Hybrid Switched-Capacitor DC-DC Converter

by

Sahana Krishnan

Master of Science in Engineering - Electrical Engineering and Computer Sciences

University of California, Berkeley

Professor Robert Pilawa-Podgurski, Chair

With the move to a 48 V distribution rail in data center power delivery architectures and automotive powertrains, high performance hybrid switched-capacitor (SC) converters have become an attractive power delivery solution in both spaces. However, automotive power systems present unique design challenges due to strict electromagnetic interference (EMI) and reliability requirements. This thesis investigates a regulating Dickson-based hybrid SC topology with low inherent EMI, and discusses the incorporation of control-based EMI mitigation techniques such as resonant and above resonant operation, as well as spread spectrum frequency modulation (SSFM). The impact of such techniques on efficiency in hybrid SC converters is explored, as well as utilizing layout techniques and passive filter designs to achieve EMI compliance. A hardware prototype combining a power stage and passive input filter is built to demonstrate the merit of hybrid SC topologies for use in 48 V automotive systems. The proposed filter and modulation schemes enable this converter to meet the CISPR 25, automotive EMI standard. A 150 W hardware prototype is built and tested to demonstrate the merit of hybrid SC topologies for use in 48 V automotive systems. The converter achieves a peak efficiency of 97.1% for 48 V-to-5 V regulated operation at 150 W of output power and meets CISPR 25, Class 5 EMI regulation limits.
To Dad, Mom, Mohana, and Max
# Contents

## List of Figures

iv

## List of Tables

vii

1 Introduction 1

2 Power Electronics in the Automotive Industry 3
   2.1 Background and Motivation ........................................ 3
   2.2 Moving Towards a 48 V System .................................... 4
   2.3 Automotive Industry Challenges ................................... 5

3 Dickson-Based Topologies 7
   3.1 The Buck Converter ................................................. 7
   3.2 The Dickson Converter: Evolution and Topology Comparison ...... 8

4 Hybrid, Interleaved-Input, Single Inductor Dickson (HISID) Converter 14
   4.1 The HISID Converter .................................................. 15
   4.2 Circuit Operation .................................................... 16
   4.3 Circuit Operation - Output Voltage Regulation .................... 19

5 EMI Mitigation Techniques 21
   5.1 EMI Background and Test Setup .................................... 21
   5.2 HISID Layout Considerations ....................................... 24
   5.3 Passive Front-End EMI Filter ....................................... 28
   5.4 Spread Spectrum Frequency Modulation (SSFM) ..................... 30

6 Experimental Results 33
   6.1 Experimental Prototype ............................................. 33
   6.2 Converter Operation ................................................. 36
   6.3 Converter Efficiency and Power Loss Breakdown ................... 37
   6.4 Implementation of EMI Mitigation Techniques - Filter Size Reduction (Rev 0) 39
# List of Figures

2.1 Diagram of an internal combustion engine (ICE) power delivery architecture, highlighting potential use cases of the power converter presented in this thesis. ...................... 4

2.2 Diagram of an electric vehicle (EV) power delivery architecture, highlighting potential use cases of the power converter presented in this thesis. .......................... 5

3.1 Schematic of a conventional synchronous buck converter. ................................. 7

3.2 Historical Dickson evolution: a) voltage multiplier configuration, b) purely capacitive Dickson converter, c) hybridized, single-ended Dickson converter. ................. 9

3.3 Dickson topology evolution. .................................................................................. 10

3.4 4:1 HISID converter charge flow analysis. .............................................................. 11

3.5 Four Dickson variants including the a) single-ended, hybrid Dickson, b) switched-tank, c) stacked-ladder ReSC, and d) HISID converters for a 4:1 conversion ratio. 11

4.1 Schematic drawing of an 8-to-1 HISID converter with switches color-coded according to their control signals. ................................................................. 14

4.2 Input current, $I_{HI}$, and inductor current, $I_L$, in both switching phases of the HISID converter. ................................................................. 15

4.3 Exemplar switch node waveforms for the HISID converter. ................................. 16

4.4 Switching scheme and exemplar converter waveforms for output voltage regulation of the HISID converter operating at a switching frequency faster than resonance. 17

4.5 Equivalent circuits for each sub-phase of a regulating 8-to-1 Dickson converter, with split-phase switching and regulating sequence as ordered a-f: Phase 1a $\rightarrow$ Phase 1b $\rightarrow$ Phase 1c $\rightarrow$ Phase 2a $\rightarrow$ Phase 2b $\rightarrow$ Phase 2c. ................................. 18

5.1 CISPR 25, Class 5 peak, quasi-peak, and average conducted emissions limits. . . 22

5.2 Pre-compliant lab setup for measuring conducted emissions. ............................ 23

5.3 Common Mode and Differential Mode noise paths through the system. ............. 23

5.4 Commutation loop comparison for two different layout configurations of the HISID converter. ................................................................. 25

5.5 Single, zoomed in commutation loop shown in schematic and PCB implementation. 26

5.6 PCB trace resistance comparison of Rev 0 and Rev 1 layouts. ............................ 27

5.7 Schematic of the front-end EMI filter. ................................................................. 28
5.8 Insertion loss plot showing attenuation across frequency for different degrees of impedance mismatch. ................................................................. 30
5.9 Modulated switching frequency over time and key parameters for various SSFM schemes: a) triangular, b) right-triangular, c) sinusoidal, d) trapezoidal, and e) pseudo-random. ................................................................. 31
6.1 Image of HISID prototype board (with EMI filter included on bottom side) .... 33
6.2 Image of top and bottom sides of prototype board with key classes of components labeled. ......................................................................................................................... 34
6.3 Image of top and bottom sides of prototype board with all components labeled. 34
6.4 (a) Above resonant, and (b) regulating inductor current, $i_L$, and switch-node voltage, $v_{sw}$, measured waveforms for the 8-to-1 discrete hardware prototype at 30 A load current. ................................................................. 37
6.5 Measured efficiency of 8-to-1 hardware prototype at 40-54 V input voltage range and both regulated 5 V and 3.3 V outputs, switching at 122 kHz (2.8× faster than resonance). ................................................................. 38
6.6 Power loss breakdown for full-load (150 W) regulating (48 V-to-5 V) operation of the HISID hardware prototype. ................................................................................................................................. 39
6.7 Photograph of EMI filter daughter board mounted on Rev 0 power stage prototype. Power stage components are mirrored on the top and bottom side of the PCB. ................................................................................................................................. 40
6.8 Peak CM and DM emissions plots for above-resonant (∼106 kHz) regulating operation with and without EMI filter (22 µH DM inductors) and no SSFM. ................................................................................................................................. 41
6.9 Peak and average CM and DM emissions plots for above-resonant (∼106 kHz) regulating operation with reduced filter size (15 µH DM inductors) and SSFM enabled. ................................................................................................................................. 43
6.10 Peak conducted emissions plots for above-resonant (∼122 kHz) regulating operation with no EMI mitigation operating at 48 V input, 5 V output and 150 W output. ................................................................................................................................. 45
6.11 Peak conducted emissions with both EMI filter and SSFM employed, measured on positive terminal of converter. ................................................................................................................................. 46
6.12 Average conducted emissions with both EMI filter and SSFM employed, measured on positive terminal of converter. ................................................................................................................................. 46
B.1 Top level circuit schematic. ................................................................................................................................. 60
B.2 Output inductor circuit schematic. ................................................................................................................................. 61
B.3 EMI filter circuit schematic. ................................................................................................................................. 62
B.4 Power stage circuit schematic. ................................................................................................................................. 63
B.5 GaN gate driver circuit schematic (for switches S19 and S20 in the design). ................................................................................................................................. 64
B.6 GDCP Si gate driver circuit schematic (for switches S3, S4, S18, and S17 in the design). ................................................................................................................................. 65
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>B.7</td>
<td>CBS Si gate driver circuit schematic (for switches S5-S8 and S11, S12, S15, and S16 in the design).</td>
<td>66</td>
</tr>
<tr>
<td>B.8</td>
<td>GDCP Si gate driver circuit schematic (for switches S9, S10, S13, and S14 in the design).</td>
<td>67</td>
</tr>
<tr>
<td>B.9</td>
<td>Half bridge gate driver circuit schematic (for switches S1 and S2 in the design).</td>
<td>68</td>
</tr>
<tr>
<td>B.10</td>
<td>Voltage sensing circuit schematic.</td>
<td>69</td>
</tr>
<tr>
<td>B.11</td>
<td>Top layer of PCB.</td>
<td>71</td>
</tr>
<tr>
<td>B.12</td>
<td>First inner layer of PCB.</td>
<td>72</td>
</tr>
<tr>
<td>B.13</td>
<td>Second inner layer of PCB.</td>
<td>73</td>
</tr>
<tr>
<td>B.14</td>
<td>Third inner layer of PCB.</td>
<td>74</td>
</tr>
<tr>
<td>B.15</td>
<td>Fourth inner layer of PCB.</td>
<td>75</td>
</tr>
<tr>
<td>B.16</td>
<td>Bottom layer of PCB.</td>
<td>76</td>
</tr>
</tbody>
</table>
List of Tables

3.1 Topology Comparisons for 48-6 V (8:1) conversion at 20 A output current . . . . 13
5.1 Conducted Noise Limits for CISPR 25, Class 5 EMI Standards [1] . . . . . . . . 21
6.1 Component Listing of the Hardware Prototype . . . . . . . . . . . . . . . . . 35
6.2 Converter Operating Parameters . . . . . . . . . . . . . . . . . . . . . . . . . 36
Acknowledgments

First and foremost, I want to thank my graduate research advisor, Professor Robert Pilawa-Podgurski. I am so grateful that you encourage each of your students to find practical, real-world applications for our research as this is what motivates me every day. Thank you for guiding my research direction and for giving me opportunities to participate and grow in activities outside of research as well.

I would also like to thank Professor Jessica Boles for being on my Masters Committee. I am so happy that you joined the EECS faculty at Berkeley, and I look forward to working more with you in the future.

I want to give a special thank you to Ryan Manack and Pradeep Shenoy. Ryan, thank you for being a great first manager at Texas Instruments and for pushing me to step out of my comfort zone in my career. Pradeep, I am eternally grateful for your mentorship and friendship, and would not be where I am today without your encouragement. Thank you both for introducing me to the world of power electronics!

Next, I would like to thank the Pilawa Research Group. Power electronics on its own is a lot of fun, but it is truly the people in this group that make coming to work each day so enjoyable. A huge thank you to: Logan Horowitz, Rod Bayliss III, Elisa Krause, Yicheng Zhu, Jiarui Zou, Ting Ge, Nathan Ellis, Joseph Schaadt, Nagesh Patle, Nathan Bisterfeld, Tahmid Mahbub, Anya Shah, Ben Liao, Khalid Durani, Marrin Nerenberg, Ivan Petric, Kelly Fernandez, Nathan Brooks, and Sam Coday. Each person in this research group brings a unique perspective to the field of power electronics, and I feel so lucky to have such technical expertise and general positive energy around me every day.

The first year of any new life chapter is always challenging. Haifah Sambo, I will always be grateful that you joined the group with me. From taking classes together and finding our way through research to shopping trips and celebrity gossip, I could not have picked a better person to go on this journey with.

Rahul Iyer, our walks are the best way to break up long days at work. Thank you for being there to talk about research ideas, life, and everything in-between.

To Francesca Giardine and Jessica Keast – my home away from home. You have been so warm and welcoming from my first day of graduate school, and I’ll never be able to thank you enough for all that you have done for me. Francesca, thank you for making stressful work days lighthearted and filled with caffeine!

Rose Abramson, when I first emailed you as a prospective student, I never thought you would become one of my favorite people to collaborate and hang out with. I have had the best time with you working on topics ranging from converter control to magnetics design, and you have given me so much confidence in myself. Thank you for being such a bright, kind, and goofy presence in my life, bringing the laughs when I need them most.

Maggie Blackwell, where do I even begin? The two years that we spent working together were the most fun I have ever had designing, building, and debugging power converters. Your diligence and attention to detail are qualities that I admire, and I will carry the skills
I learned from you throughout my career. I am so fortunate to continue to have you in my life as a wonderful friend. This work would not have been possible without you.

Finally, I would like to give the biggest thank you to my family: Dad, Mom, my sister, Mohana, and my cutie terrier, Max. You are the reason for it all. There aren’t words to express how thankful I am for your endless support and guidance, and I love you so much. This thesis is for you.
Chapter 1

Introduction

Hybrid switched-capacitor (SC) converters can achieve high efficiencies and power densities due to their better utilization of passive components compared to conventional switched-inductor topologies [2,3]. Particularly, high conversion ratio step-down hybrid SC converters designed for data center power delivery have demonstrated high performance against efficiency and power density metrics [4–6]; however, these converters are not yet commonly used in other high-energy-consumption industries, such as the automotive industry. Though these hybrid SC topologies boast high performance and passive component utilization, they often are constructed with a large number of switching elements, which can be a drawback from a complexity point-of-view. However, due to their multi-level structure, hybrid SC converters can have reduced switch blocking-voltages and lower $dv/dt$ at the switch nodes compared to conventional two-level topologies, resulting in reduced electromagnetic interference (EMI). This potentially makes them strong candidates for use in automotive applications [7], where the EMI requirements are more stringent. Therefore, it is necessary to further investigate hybrid SC converter performance within these harsher environmental constraints.

Moreover, as the automotive market shifts towards more hybridized and fully electric vehicles (EVs), internal combustion engine (ICE) vehicles are also beginning to adopt 48 V batteries in place of or in conjunction with the legacy 12 V battery for partial hybridization of the powertrain [8]. This allows ICE vehicles to align themselves with EVs where 48 V is emerging as a low-voltage dc bus for auxiliary electrical loads [9]. The use of a higher bus voltage decreases $I^2R$ transmission losses as power demands increase and allows for lighter-weight cabling systems to be employed within the vehicle. The higher power subsystems may be powered from a 48 V bus directly, while the lower-voltage subsystems may be driven using high-density point-of-load (PoL) converters [10–14]. Following similar trends to data-center power delivery [4–6, 15–17], automotive power delivery can eliminate a voltage conversion step by completely removing the intermediary bus.

The similarity in power delivery architecture indicates an opportunity to apply the advanced power converter designs used in data center applications to automotive power solutions, with some additional considerations. First, in the automotive powertrain, the 48 V nominal battery voltage can vary above and below the nominal 48 V [18], which the power
CHAPTER 1. INTRODUCTION

Converter must regulate to the desired output voltage. Second, the power converters themselves must meet industry EMI requirements so that they do not interfere with any other electrical subsystems in the vehicle.

This thesis presents a regulating hybrid, interleaved-input, single-inductor Dickson (HISID) converter [19–21]. The HISID converter in this work also has an added custom front-end EMI filter to demonstrate the ability of hybrid SC converters to meet CISPR 25, Class 5 EMI standards [1], the most stringent class for on-vehicle applications. The regulation capabilities and the inherent EMI benefits of the hybrid SC converter are explored along with three of the most common techniques to mitigate EMI: 1) layout considerations, 2) front-end conducted EMI filter design and 3) spread spectrum frequency modulation (SSFM) [22,23]. The impacts of the EMI mitigation techniques on system size and efficiency are also analyzed.

The remainder of this thesis is organized as follows: Chapter 2 reviews current trends in the automotive industry and motivates the shift from the legacy 12 V architecture to 48 V. Chapter 3 discusses the evolution of the Dickson converter as well as its merits compared to other power converter topologies in high conversion ratio, step-down applications. Chapter 4 describes the HISID topology used in this work and details its operation. Chapter 5 discusses EMI implications associated with this topology as well as possible mitigation techniques, most notably converter layout, passive filter design and SSFM. Chapter 6 includes specifications for the experimental prototype as well as measured waveforms, efficiency, and conducted EMI results. Finally, Chapter 7 concludes the thesis and proposes possible future directions for the work.
Chapter 2

Power Electronics in the Automotive Industry

2.1 Background and Motivation

The transition from internal combustion engine (ICE) vehicles to battery electric vehicles (EVs) is well underway. The United States Federal Government has set a goal that half of all vehicles sold in 2030 will be zero-emission vehicles and has plans to increase the convenience and equity of charging infrastructure across the country [24]. Other countries across the world are setting similar targets, and countries such as Norway and China [25] are increasing EV sales and setting goals to add more charging infrastructure.

With the growing number of EVs, it is apparent that the electrical power requirements within these vehicles is increasing as well. Each subsystem and its components within a car (air conditioning, driver assistance systems, infotainment systems, etc.) require power. And adding more powertrain capability, driver and passenger amenities, and computing features expands the number of loads in any one vehicle. Supporting these loads and, therefore, higher power requirements requires that the cabling that supplies this power within a vehicle be large enough to carry the increased amount of current – thicker wires correspond to lower resistance which enables higher current carrying capabilities. In a similar way, thinner and smaller diameter wires have lower current carrying capability. However, larger wire thickness comes with increased cabling weight, and these wires can take up more space within the vehicle. If the power, $P$, demanded by a set of loads needs to be supported without compromising on vehicle weight, it is a better decision to increase the voltage, $V$, rather than the current, $I$. This is because power is related directly to the product of current and voltage ($P = IV$), and thus an increase in voltage will reduce the current required to support a given power level.
2.2 Moving Towards a 48 V System

The legacy architecture in vehicles has been a 12 V-based automotive electrical system. With the changing power requirements of today's cars, the current flowing through these 12 V systems has increased significantly in recent years. Many original equipment manufacturers (OEMs) have targeted a 48 V system as the next-generation voltage architecture in cars as it increases the voltage while reducing the current for a given power level. More precisely, the 48 V technology can increase power capabilities up to four times that of the legacy 12 V bus. Furthermore, the 48 V level is still below the 60 V limit of what is generally deemed safe for protection against shock hazard [26].

![Diagram of an internal combustion engine (ICE) power delivery architecture, highlighting potential use cases of the power converter presented in this thesis.](image)

The 48 V architecture can be implemented in a few different ways for each of the vehicle types. The LV148 standard [18] has been developed for ICE and mild hybrid vehicles in order to merge the 48 V bus with an existing 12 V system, as shown in Fig. 2.1. The 12 V rail will continue to power infotainment and audio and passenger safety systems while the 48 V rail will directly supply systems such as adjustable suspensions and air conditioning compressors. Having both 48 V and 12 V batteries in the on-board system also presents the opportunity
to include a bi-directional power converter between the 48 V and 12 V batteries [27]. This architecture allows the 48 V bus to supply the more power-demanding loads that are inherent to hybrid electric vehicles while the traditional lower power loads of an ICE vehicle can still be powered from the 12 V battery.

In the case of a fully electric vehicle, the 48 V distribution works well in conjunction with a 400 V or even 800 V high-voltage battery. In this case, instead of having the 48 V as a physical battery voltage, the 400 or 800 volt battery is converted down to an intermediate 48 V bus within the vehicle to power downstream loads [28]. This elimination of the physical 48 V battery enables smaller weight and size as well as higher power density. A general system architecture for electric vehicles is shown in Fig. 2.2 below.

Figure 2.2: Diagram of an electric vehicle (EV) power delivery architecture, highlighting potential use cases of the power converter presented in this thesis.

### 2.3 Automotive Industry Challenges

There are many challenges that come with a shift in voltage architecture such as this one. As previously mentioned, the 48 V-based network still needs to support and power the legacy 12 V loads. Managing increased power requirements comes with challenges related to keeping vehicles light, high performance, and low cost. Therefore, highly dense and efficient power converters are required to support the increasing number of loads.
Furthermore, increasing the bus voltage from 12 to 48 V presents additional challenges related to system safety, noise, and reliability for power converters. Putting many electrical subsystems in close proximity to each other means that power converters within the vehicle must also meet industry electromagnetic interference (EMI) requirements so that they can be implemented in the vehicle without interfering each other. In many industry solutions, these power converters must also be designed using automotive components that are qualified for withstanding more severe temperatures, shock and vibration, impacting component selection of power solutions. Due to each of these challenges, the advent of this new 48 V vehicle architecture strongly motivates the investigation of novel power converter topologies beyond the conventional that enable high power density and efficiency while adhering to requirements for implementation in vehicles.
Chapter 3

Dickson-Based Topologies

This work focuses on a high-conversion ratio, step-down Dickson power converter for automotive systems. To motivate the power converter topology selection for this application, we will begin by describing a conventional solution for step-down applications as well as its limitations, followed by the evolution of the Dickson converter and how we arrive at the hybrid, interleaved-input, single inductor Dickson (HISID) converter topology.

3.1 The Buck Converter

\[
\begin{array}{c}
\text{V}_{\text{HI}} \quad \text{C}_{\text{in}} \quad S_1 \quad \text{S}_{\text{SW}} \quad \text{C}_{\text{out}} \quad \text{V}_{\text{LO}}
\end{array}
\]

Figure 3.1: Schematic of a conventional synchronous buck converter.

The buck converter is a conventional solution for voltage step-down applications [29], and its circuit schematic is shown in Fig. 3.1. Switches $S_1$ and $S_2$ operate (turn on and off) in a complementary fashion and do not conduct simultaneously. Each switch turns on for a portion of the switching period, $T_{sw}$. The percentage of the period that the switch is on is defined by the duty cycle, $D$. The switch node, $v_{sw}$, sees a square wave ranging from 0 V to the input voltage, $V_{HI}$, and the output $LC$ network filters $v_{sw}$ in order to create a largely DC output. By doing a Kirchhoff’s Voltage Law (KVL) analysis on the circuit, we can see that both switches ($S_1$ and $S_2$) need to block a maximum voltage of $V_{HI}$.

A main advantage of the buck converter is that it is simple and relatively easy to design for a given set of operating conditions. However at higher input voltages, the switches will
have a larger voltage stress, and there will be a large voltage swing on the switch node. As the switch node swings higher, a larger output inductor, $L$, is needed to filter the output voltage sufficiently to get a DC signal at the output. The larger voltage swing on $v_{sw}$ also incurs high $dv/dt$ transitions which can contribute to higher EMI. Furthermore, at higher voltage step-down ratios, the buck converter will be required to operate at very low duty cycles, given by $V_{LO}/V_{HI}$. At extreme conversion ratios, the buck will be forced to operate close to minimum on-time limits of switches. This can impose limitations on gate drive circuitry and have other negative implications for converter control. Finally, the buck converter will also see large efficiency penalties at these extreme conversion ratios. In order to address many of these limitations, we can investigate other circuit topologies for use in high step-down ratio applications.

### 3.2 The Dickson Converter: Evolution and Topology Comparison

The Dickson converter, a variant of which is described in this thesis, can overcome many of the aforementioned limitations of the conventional buck converter in high step-down ratio applications. In order to introduce this topology, we start with some historical context on its origin. In 1976, J.F. Dickson published the voltage multiplier technique (Fig. 3.2a) which was originally on-chip, using diodes to step-up the voltage [30]. It operates similarly to the classical Cockcroft-Walton multiplier [31], and has several advantages. First, at high values of stray capacitance ($C_s$), the circuit can achieve efficient voltage multiplication. Second, the current drive capability does not depend on the number of multiplier stages. This technique has since found use in fully synchronous step-down applications, an example of which is depicted in Fig. 3.2b, which shows a purely capacitive Dickson converter design using switches [32]. However, this Dickson topology may also be hybridized through the introduction of an inductor, and in that way we can remove pulsed inrush currents present in pure switched-capacitor converters (i.e. slow switching limit (SSL) losses) [33]. The single-ended, hybridized Dickson converter is depicted in Fig. 3.2c.

To arrive at the converter used in this work, we begin with the single-ended hybrid Dickson converter and place two of them in parallel. Paralleling two of the single-ended converters provides reduced high-side input current ripple and, therefore, better EMI performance, if the converters are interleaved and operated 180° out of phase relative to each other. After doing this, we can see that both the low-side outputs have identical, inductively loaded full-bridges. Since they are both operating in the exact same manner, we can remove one of them (highlighted in red in Fig. 3.3) and merge the remainder of the two structures together. The resulting converter is an interleaved, hybrid Dickson structure. This entire process is illustrated in Fig. 3.3 for a 4:1 Dickson converter, with the final interleaved Dickson structure shown serving as the converter of interest in this thesis. Half of the switches in the converter are controlled as “Phase 1” (red switches), and the rest are controlled as “Phase 2” (blue
Figure 3.2: Historical Dickson evolution: a) voltage multiplier configuration, b) purely capacitive Dickson converter, c) hybridized, single-ended Dickson converter.

Charge flow analysis [34] can be used to arrive at the conversion ratio for this hybrid, interleaved-input, single inductor Dickson (HISID) converter. This process is briefly shown in Fig. 3.4. The assumptions that we make when deriving the conversion ratio are as follows: we assume that there is a 50% duty cycle between Phases 1 and 2, and also assume that the capacitors are chosen to ensure an equal effective capacitance in each phase. We start on the left of Fig. 3.4 with Phase 1 and denote the charge flowing from \( V_{HI} \) through \( S_{11} \) to be \( q_1 \). Charge \( q_1 \) flows into \( C_{3L} \), and to the output through \( S_3 \) and the output inductor, \( L \). To maintain charge balance, a condition of steady-state operation, on \( C_{3L} \) during Phase 2, the same charge \( q_1 \) must flow out of \( C_{3L} \). In Phase 2, this \( q_1 \) also charges \( C_{2R} \) before continuing to the output through the inductor. To balance this capacitor, charge \( q_1 \) must flow out of \( C_{2R} \) during Phase 1. We can continue to trace \( q_1 \) through the circuits in Phase 1 and Phase 2. We can work through a similar process for \( q_2 \), which is supplied from the input through \( S_{12} \) and into \( C_{3R} \) during Phase 2. Then in Phase 1, \( q_2 \) must flow out of \( C_{3R} \) and so on to define the charge into/out of each capacitor as well as the input and output terminals. Finally, to see how much charge flows into the output inductor in each phase, the charges at the \( V_{LO} \) output can be summed. In this case, the charge delivered to the output in Phase 1, \( q_{\text{out,PH1}} \), is equal to \( 2q_1 + 2q_2 \). The charge delivered to the output in Phase 2, \( q_{\text{out,PH2}} \), is equal to \( 2q_1 + 2q_2 \) as well.
To derive the voltage conversion ratio, we would like to relate the total input charge to the total output charge. The total input charge, $q_{\text{in, total}}$, is equal to the sum of $q_1$ and $q_2$. Assuming Phases 1 and 2 are equivalent, Eqn. 3.1 relates the charges in each phase to the total input charge:

$$q_1 = q_2 = \frac{q_{\text{in, total}}}{2}$$ (3.1)

The total output charge, $q_{\text{out, total}}$, can be found by summing up the charge delivered to the output in each phase ($q_{\text{out, PH1}}$ and $q_{\text{out, PH2}}$). This is shown in Eqn. 3.2 for the presented example:

$$q_{\text{out, total}} = q_{\text{out, PH1}} + q_{\text{out, PH2}} = 4q_1 + 4q_2$$ (3.2)

Finally, the conversion ratio can be determined by looking at the ratio of charge received by the output to the charge supplied by the input (Eqn. 3.3):

$$\frac{q_{\text{out, total}}}{q_{\text{in, total}}} = \frac{8q_1}{2q_1} = \frac{4}{1}$$ (3.3)

Thus, the Dickson topology shown in Fig. 3.4 achieves a 4:1 voltage conversion ratio. This analysis can be applied to any $N$ level Dickson converter, where the conversion ratio will be $N : 1$ and the number of flying capacitors will be $2(N - 1)$.
There are many other Dickson variants including the switched-tank converter [35] and stacked-ladder resonant switched-capacitor (ReSC) converter [36], which have demonstrated merit particularly in data center power delivery. These Dickson-variant converters are shown in Fig. 3.5 alongside the single-ended hybrid Dickson and HISID converters for a 4:1 voltage conversion ratio.

Figure 3.5: Four Dickson variants including the a) single-ended, hybrid Dickson, b) switched-tank, c) stacked-ladder ReSC, and d) HISID converters for a 4:1 conversion ratio.
Table 3.1 shows a comparison between the four topologies for an 8:1 voltage conversion ratio at an example operating condition of 48 V to 6 V at 20 A load current. This operating condition is selected as it is common in both the automotive and data center industries and motivates the work in the following chapters. The comparison is made for a range of criteria detailed below:

- **Number of Switches**: The proposed topology has the fewest number of switches for an interleaved-input topology.

- **Interleaved Input**: The interleaved nature of the input allows charge to flow from the high-side source during portions of both switching phases, rather than just one phase as is typical with many two-phase converters. Because of this quality, the rms value of the input current is reduced as compared to the non-interleaved single-inductor Dickson topology, thus reducing change in current over time. Furthermore it reduces the necessary input capacitance, making this approach advantageous for applications where designing a compact input filter is desired.

- **Regulation**: The topology in this work can easily achieve PWM regulation by adding a “inductor-connected-to-ground phase”. The output voltage can be regulated to any level lower than the fixed-ratio output voltage. While the switched-tank and stacked-ladder topologies can achieve output voltage regulation as well, the regulation range is limited by increased circulating currents.

- **Operation**: This topology can achieve above resonant operation in addition to resonant operation. However, some added control complexity called “split-phase switching” is required for both resonant and above resonant operation, and this will be discussed in more detail in Chapter 4. However, due to the symmetry of this Dickson topology, the split-phase switching scheme can have the exact same timing for both main switching Phases 1 and 2 as well as their corresponding sub-phases.

- **Inductor Placement**: A single inductor at the output serves to both act as an EMI filter and to soft-charge all flying capacitors in the interleaved structure. Additionally, unidirectional current can improve inductor losses, whereas the switched-tank’s inductors must conduct bi-directionally [35]. Moreover, if not using coupled magnetics, fewer inductors are likely more advantageous for solution power density due to magnetics scaling laws [37]. However, for inductor-at-output (“direct”) topologies [38], the switches see some combination of the voltage ripple on the inductor and capacitors. For tank-based converters, the ripple is mostly contained inside the LC tanks.

- **Volt-Amp (V-A) Switch-Stress**: If we ignore ripple and look at nominal operating conditions, all four topologies should have an equal V-A rating. However, if effects due to voltage ripple, rms currents, and split-phase timing are included, the results become more nuanced. When taking these other effects into account for a specific operating condition, the V-A product of the proposed topology is slightly higher than both the
switched-tank and stacked-ladder converters, but it is still much better than that of the Doubler, Series-Parallel, or FCML topologies [39].

After detailing many of the reasons why the HISID converter is an attractive power solution, we will next describe how it can be used in an automotive application in the following chapters.

| Table 3.1: Topology Comparisons for 48-6 V (8:1) conversion at 20 A output current |
|----------------------------------|--------------------------------|---------------------------------|---------------------------------|
| Criterion                        | Single-ended hybrid Dickson | Switched-tank converter (single-phase, $V_{out}$ clamped) | Interleaved two-phase stacked ladder | Interleaved hybrid Dickson (this work) |
| Number of switches (8:1)         | 12                           | 28                             | 32                             | 20                             |
| Interleaved input                | No                           | No                             | Yes                            | Yes                            |
| Regulation                       | PWM regulation               | Phase-shifted PWM              | Phase-shifted PWM              | PWM regulation                 |
| Split-phase operation            | Not required (for odd conversion ratios) | Not required | Not required | Required |
| Above resonant operation         | Yes                          | No                             | No                             | Yes                            |
| Inductor placement               | 1 at output                  | 7 tanks                        | 2 tanks                        | 1 at output                    |
| Voltage ripple imposed on switches? | Yes (large)                  | No (small)                     | No (small)                     | Yes (large)                    |
| Nominal switch voltage stress    | $6 \times 12$ V             | $28 \times 6$ V               | $32 \times 6$ V               | $12 \times 12$ V              |
|                                 | $6 \times 6$ V              |                                |                                | $8 \times 6$ V                |
| Switch current stress ($I_{rms}$ at resonance) | $4 \times 13.7$ A          | $4 \times 13.7$ A             | $4 \times 13.7$ A             | $4 \times 13.7$ A             |
|                                 | $6 \times 4$ A              | $28 \times 3.9$ A             | $28 \times 2$ A              | $12 \times 2$ A              |
|                                 | $2 \times 5$ A              |                                |                                | $4 \times 2.5$ A              |
| V-A Product (48-to-6 V, 20 A output) | 677                         | 660                           | 665                           | 677                           |
Chapter 4

Hybrid, Interleaved-Input, Single Inductor Dickson (HISID) Converter

The theoretical circuit of the HISID converter is presented in [40] along with the circuits of many other Dickson converter variants. This thesis in an extension of prior conference papers [20, 41], and provides an extended circuit analysis, additional design guidelines, and further investigations of EMI mitigation techniques. The HISID topology (Fig. 4.1) is used in this work to demonstrate efficient, compact, and EMI-compliant DC-DC power conversion for use in an automotive environment. This topology is attractive for automotive applications as it features an interleaved input, low switch blocking voltage stress, low-$dv/dt$ switching transitions, and output regulation capability.

![Figure 4.1: Schematic drawing of an 8-to-1 HISID converter with switches color-coded according to their control signals.](image-url)
CHAPTER 4. HYBRID, INTERLEAVED-INPUT, SINGLE INDUCTOR DICKSON (HISID) CONVERTER

4.1 The HISID Converter

The proposed converter [19, 40] evaluated in this work, shown in schematic form in Fig. 4.1, is a variation on the single-ended hybrid Dickson converter [30, 42]. This topology is called the hybrid, interleaved-input, single-inductor Dickson (HISID) converter [19, 20].

We will now re-emphasize a few merits of this HISID topology compared to others from the preceding chapter. The topology takes a similar approach to interleaving as the two-phase interleaved stacked-ladder in [36]. This is beneficial for minimizing current and voltage ripple at the input and output ports, thereby reducing filtering requirements at those ports. However, as compared to the two-phase interleaved stacked-ladder, the base topology in this work does not require the bulky capacitor column of [36] because it is inherently interleaved with a single output inductor instead of two tank-configured inductors in the stacked-ladder topology. The proposed HISID topology only requires $2N + 4$ switches for an $N : 1$ conversion while the interleaved stacked-ladder in [36] requires $4N$ switches. Moreover, the HISID converter is capable of above-resonant operation and continuous forward conduction [33] allowing for output voltage regulation without incurring increased circulating currents [43]. However by doing this, we sacrifice some zero-current switching (ZCS)/zero-voltage switching (ZVS) capability.

The HISID converter topology is also advantageous for EMI, making it an attractive option for high-efficiency and low-cost applications. In the proposed converter, the interleaved input allows charge to flow from the high-side source during portions of both switching phases, rather than just one phase as is typical with many two-phase converters (such as the buck converter). Because of this characteristic, the rms value of the input current is reduced compared to the non-interleaved single-inductor Dickson topology [19], thereby reducing the necessary input capacitance. This is imperative for high power density of the total power conversion system. Fig. 4.2 shows the input current, $I_{HI}$, and inductor current, $I_L$, in both switching phases. In the waveform figure, both main Phases 1 and 2 are divided into sub-phases. For example, Phase 1 includes sub-phases 1a, 1b (the “split” phase), and

Figure 4.2: Input current, $I_{HI}$, and inductor current, $I_L$, in both switching phases of the HISID converter.
1c (the regulation sub-phase). Phase 2 is divided in the same manner. These sub-phases are discussed in more detail in the following sections: split-phase operation in Section 4.2 and regulating operation in Section 4.3.

As described in [2, 3, 44], the family of Dickson-style converters demonstrates minimal total switch stress compared to other hybrid SC topologies. This feature reduces the swing at the switch node, reducing the high $dv/dt$ transitions and yielding better EMI performance. Fig. 4.3 shows example switch node waveforms for the HISID converter. The switch node sees, nominally, the output voltage ($V_{LO}$) as compared to a buck converter’s switch node which sees the entire input voltage. Furthermore, the output side of the HISID converter has similar benefits as the output of a buck converter. The inductor at the low-side port of the converter not only allows us to operate the converter at resonance and above-resonance but serves as an output EMI filter and ensures full soft-charging of the flying capacitors [45, 46].

![Figure 4.3: Exemplar switch node waveforms for the HISID converter.](image)

While there are many facets of this topology that make it attractive for both EMI and automotive applications, it does require additional control complexity to maintain soft-charging of the flying capacitors. Split-phase operation [46] is a control scheme which introduces two additional sub-phases within the two main switching phases. Operating the converter in this way ensures soft-charging of the flying capacitors. As mentioned previously, interleaved converters such as the HISID converter can have reduced input rms currents. However, owing to the requirement for split-phase switching, input switches $S_{19}$ and $S_{20}$ turn off towards the end of primary Phases 1 and 2, respectively, thereby disconnecting the input source. Even though the input current is not fully continuous throughout each period (Fig. 4.2), there is still significant improvement over a single-ended topology, where the input current would be zero for 50% of the switching period.

### 4.2 Circuit Operation

Converter waveforms and gating signals for for an 8:1 ($N = 8$) HISID converter are shown in Fig. 4.4, with equivalent circuits for each phase and corresponding split-phase shown in Fig. 4.5. The converter operates with a 50% duty cycle for Phases 1 and 2. This implementation imposes a twice switching frequency ripple on the switch node voltage and output
inductor current, allowing for smaller-sized magnetics [37]. Assuming two-phase operation, all odd-numbered switches ("bridge" switches $S_1, S_3$ and "string" switches $S_5 - S_{19}$) are on during Phase 1 and all even-numbered switches ("bridge" switches $S_2, S_4$ and "string" switches $S_6 - S_{20}$) are on during Phase 2. Due to the inherent interleaved symmetry of the topology, sizing the flying capacitors such that $C_{iL} = C_{iR}$ for $i \in \{1, N - 1\}$, results in an identical effective capacitance presented at the switch-node $v_{sw}$ during both Phase 1 (Fig. 4.5a) and Phase 2 (Fig. 4.5d). Because the single inductor at the output is engaged with an identical capacitor network during both phases, Phase 1 and Phase 2 exhibit equivalent operation.

Operating in this two-phase manner, however, will result in hard-charging of the flying capacitors. In order to maintain soft-charging, the converter must be operated with split-phase operation. Detailed in [46], split-phase operation describes the introduction of sub-phases within the two main switching phases to ensure soft-charging of the flying capacitors through the output inductor. Without these additional switching states, large current spikes occur at phase transitions due to mismatched loop voltages. These hard-charging events have a negative impact on efficiency due to capacitor charge-sharing losses, and on EMI
CHAPTER 4. HYBRID, INTERLEAVED-INPUT, SINGLE INDUCTOR DICKSON (HISID) CONVERTER

Figure 4.5: Equivalent circuits for each sub-phase of a regulating 8-to-1 Dickson converter, with split-phase switching and regulating sequence as ordered a-f: Phase 1a → Phase 1b → Phase 1c → Phase 2a → Phase 2b → Phase 2c.

performance from increased voltage and current spikes. Furthermore, without soft-charging operation, capacitor voltage ripples may need to be reduced to improve efficiency and EMI performance, thereby decreasing passive utilization.

For the step-down HISID converter in this work, the following split-phase switching scheme is utilized to satisfy voltage loops at phase transitions. Sub-phases 1b (Fig. 4.5b) and 2b (Fig. 4.5e) are inserted between the transition from Phase 1a to 2a and from 2a to 1a, respectively. Since capacitors \{C_{1R}, C_{7L}\} and \{C_{1L}, C_{7R}\} are not series-connected to other capacitors during Phase 1a and Phase 2a, respectively, these capacitors accrue charge more quickly than the other flying capacitors (assuming all capacitors are equally sized). Switches \(S_5\) and \(S_{19}\) (Phase 1b) and Switches \(S_6\) and \(S_{20}\) (Phase 2b) are turned off to remove capacitors \{C_{1R}, C_{7L}\} and \{C_{1L}, C_{7R}\} from the circuit before they are reconnected in a different configuration for the following phase. Correct timing of the “b-phase” durations and placement within the primary Phases 1 and 2 are necessary to achieve full capacitor soft-charging \[46, 47\].
Moreover, selection of desired converter switching frequency is also non-trivial. The resonant frequency is related to the equivalent L-C tanks formed at the switch-node during each phase and corresponds to ZCS operation where the inductor current reaches 0 A at the end of each primary phase. The ratio of switching frequency to resonant frequency \( \Gamma = \frac{f_{\text{sw}}}{f_{\text{res}}} \) can then be tuned for best performance. Considerations for switching losses and conduction losses, as well as the EMI regulatory frequency range [1] and resultant EMI filter size, inform the choice of switching frequency and resonant frequency, and thereby the inductor and capacitor values. Trade-offs for different operating regimes of this Dickson-variant converter topology are explored in detail in [19], specifically with regard to how the choice of switching frequency impacts efficiency and EMI performance. An advantage of above-resonant operation is the ability to operate in continuous conduction mode (CCM) and regulate the output voltage to a value lower than \( \frac{V_{\text{HI}}}{N} \), typically a requirement for PoL converters. Additionally, operation of this hybrid SC converter above resonance provides tolerance to component mismatch, which enables the use of high energy density, Class II ceramic flying capacitors [15, 48, 49]. Because of these benefits and for reasons discussed later, the converter is operated above resonance in this work.

4.3 Circuit Operation - Output Voltage Regulation

The HISID converter is able to achieve a fixed-conversion ratio when operated with 50% duty cycle. However, due to the output configuration of the inductor, \( L \), and quad switches, \( S_1 - S_4 \), this HISID converter can also be viewed as a fixed ratio switched-capacitor network merged with a buck converter at the output. Switches \( S_1 - S_4 \) can be controlled to regulate the output voltage to any value lower than the fixed-conversion-ratio output. This work focuses on validating the converter for operation with a regulated 5 V output or a regulated 3.3 V output. These voltage levels were chosen as 5 V and 3.3 V are important low voltage rails in an automotive subsystem, powering downstream loads such as processors, sensors, and in-vehicle networks.

To regulate the output voltage, a regulation sub-phase (Phases 1c and 2c in Fig. 4.4) is inserted within each main switching phase. During each of these sub-phases, the output inductor is connected between the output and ground [49,50]. Phase 1 consists of Phase 1a (Fig. 4.5a), its corresponding split-phase, Phase 1b (Fig. 4.5b), and its regulating sub-phase, Phase 1c (Fig. 4.5c). Similarly, Phase 2 consists of Phase 2a (Fig. 4.5d), Phase 2b (Fig. 4.5e), and Phase 2c (Fig. 4.5f). During the regulating intervals, switches \( S_5 - S_{20} \) are off and the current through inductor \( L \) freewheels via the four bridge switches, \( S_1 - S_4 \). The duration of each regulation sub-phase is set according to the relationship between the switching frequency and resonant frequency as well as the desired output voltage. The switching frequency and resonant frequency are often selected to optimize efficiency and power density. However, in automotive applications, these frequencies can also play a large role in determining EMI performance and compliance with regulatory standards. The next chapter takes a closer look at automotive EMI requirements as well as mitigation techniques and their impacts on the
CHAPTER 4. HYBRID, INTERLEAVED-INPUT, SINGLE INDUCTOR DICKSON (HISID) CONVERTER

HISID converter.
Chapter 5

EMI Mitigation Techniques

Automotive power converters are often placed close to other in-vehicle electronic systems, and many of these systems are susceptible to EMI. The high $di/dt$ and $dv/dt$ associated with the power converter switching transitions must be mitigated. Topologically, the HISID converter has many inherent EMI benefits. This chapter provides some background on automotive EMI regulations and useful mitigation techniques that are incorporated into the HISID power converter solution.

5.1 EMI Background and Test Setup

The allowable EMI automotive noise levels are standardized in CISPR 25, under Class 5 limit requirements [1]. The EMI spectrum is measured over the frequency range of 150 kHz to 108 MHz and there are average, quasi-peak, and peak noise limits that are set within this range. The term “quasi-peak” refers to weighting the signals according to rate at which they are measured by the detector. A summary of the peak and average specifications for the CISPR standard are presented in Table 5.1 and these limits are shown on a plot in Fig. 5.1.

Full compliance testing requires peak, quasi-peak, and average noise levels to be detected, but, both the quasi-peak and average data cannot exceed the peak levels [51]. In this

<table>
<thead>
<tr>
<th>Band</th>
<th>Frequency (MHz)</th>
<th>Limit (dBμV)</th>
<th>Peak</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>LW</td>
<td>0.15-0.3</td>
<td>70</td>
<td>50</td>
<td></td>
</tr>
<tr>
<td>MW</td>
<td>0.53-1.8</td>
<td>54</td>
<td>34</td>
<td></td>
</tr>
<tr>
<td>SW</td>
<td>5.9-6.2</td>
<td>53</td>
<td>33</td>
<td></td>
</tr>
<tr>
<td>FM</td>
<td>76-108</td>
<td>38</td>
<td>18</td>
<td></td>
</tr>
<tr>
<td>TV</td>
<td>41-88</td>
<td>34</td>
<td>24</td>
<td></td>
</tr>
<tr>
<td>CB</td>
<td>26-28</td>
<td>44</td>
<td>24</td>
<td></td>
</tr>
<tr>
<td>VHF I</td>
<td>30-54</td>
<td>44</td>
<td>24</td>
<td></td>
</tr>
<tr>
<td>VHF II</td>
<td>68-87</td>
<td>38</td>
<td>18</td>
<td></td>
</tr>
</tbody>
</table>
work, peak EMI data is reported to quantify “worst case” noise levels for the converter, and average data is reported to demonstrate the positive impact of the spread spectrum frequency modulation (SSFM) technique on noise levels. This technique will be discussed in more detail later in this chapter.

While both conducted and radiated emissions are regulated in CISPR 25, Class 5, only conducted emissions are analyzed in this work for an initial demonstration of emissions performance of this hybrid SC converter. The conducted emissions are measured in a pre-compliance setup (Fig. 5.2), which provides a good preliminary indication of overall EMI performance. The integrity of each pre-compliant EMI setup will vary based on its location as well as ambient noise in the environment. To get a sense of the inherent noise of the EMI setup, a noise floor measurement of the setup can be taken with the power converter and all test equipment off. This can help to account for variation between pre-compliant test setups. Radiated emissions measurements are more intricate, involving the use of a large, fully-shielded anechoic chamber, and are also highly dependent on cable routing and mounting orientation [52]. Therefore, these measurements were not carried out in this work.

Conducted emissions can be broken down into two different kinds of noise: common mode (CM) and differential mode (DM), shown in Fig. 5.3. CM refers to noise in which the direction of the “noise currents” on the positive and negative lines of the power converter have the same direction. DM refers to noise in which the direction of the “noise currents” on the positive and negative lines of the power converter have the opposite direction. CM noise increases with increasing parasitic capacitance in the power stage and larger and/or faster switched-voltage transitions [23,53,54]. On the other hand, DM noise is increased by increasing load current. Higher load current will exacerbate the impact of parasitics in larger
current loops throughout the power stage [23]. In this work, we measure the conducted EMI as well as its CM and DM components at the 48 V input bus. For applications which utilize a distribution bus, it is crucial to shield the bus from noise generated by other switching converters. This work does not examine the impact of bus capacitance on the low-side port.

There are several ways to reduce EMI in switching power converters. EMI mitigation often starts with power converter layout practices such as strategic component placement,
loop area minimization, and effective grounding and shielding techniques [55–59]. EMI filters are also commonly added to the input side of a power stage and are tuned to mitigate noise peaks that exceed limits within the EMI measurement range. While EMI can be reduced with changes to the converter hardware, switching frequency control techniques such as spread spectrum frequency modulation (SSFM) can also be used. SSFM varies the converter switching frequency to spread the noise peaks across a range of frequencies to reduce conducted EMI [22]. Different SSFM schemes and their specific impacts on converter EMI in this hybrid SC topology are explored in more detail in [19]. In this work, each of these mitigation techniques is utilized to balance converter efficiency, size, and EMI compliance.

5.2 HISID Layout Considerations

Good PCB design is imperative to EMI mitigation as parasitics, copper traces, and ground planes can have a major impact on noise sources and magnitudes. Prior work has shown that EMI performance can be improved by minimizing commutation loop parasitic inductance within converter layout [60, 61]. Typically, power converter topologies which comprise half-bridge modules (e.g., Buck [60, 62], Flying Capacitor Multi-Level (FCML) [63], and Series-Capacitor Buck [50] converters) have relatively simple and straightforward commutation loops that are easily minimized in layout. However, for many hybrid SC topologies, which not only have a large number of switches but more complex circuit connections, optimization of commutation loops becomes more difficult. In a hybrid SC converter, each switch of Phase 1 forms a commutation loop with each switch of Phase 2. Topologies such as the Dickson converter have many commutation loops overall, and many loops per switch. For example, the 8:1 HISID converter has \((\frac{N_s}{2})^2 = 10 \times 10 = 100\) commutation loops for \(N_s = 20\) number of switches. Because there are many different converter loops, it is most effective to layout the switch network to create a single small commutation loop per switch. The designer should not attempt to minimize multiple loops per switch. Generally, there is a trade-off between power-stage size, PCB loss, and commutation loop size in layout design. For the HISID converter, minimizing commutation loops reduces PCB loss as well as switching noise and switching loss.

The comparison between two different PCB layouts of the HISID converter is depicted in Fig. 5.4 for two specific commutation loops. The layout configuration for the HISID converter in this thesis is depicted in the bottom row of Fig. 5.4. In this configuration, the converter is folded vertically on itself, which is more advantageous than the lateral fold (top row of Fig. 5.4) implemented in [19, 20] due to the significantly smaller and tighter commutation loops. This can be seen from the comparison between Figs. 5.4a and 5.4c and Figs. 5.4b and 5.4d showing the reduced loop areas and consequent reduced commutation loop inductance. Fig. 5.5 also shows a zoomed in rendering of one specific commutation loop within the converter and illustrates its implementation on the PCB.

Furthermore, for this cross-connected HISID converter, configuring the commutation loops in this way also reduces the PCB trace resistance. As an example, the PCB trace
resistance of the power path through $C_{3R}$ during Phase 2 of Fig. 5.4 (consisting of $V_{HI} \rightarrow S_{12} \rightarrow C_{3R} \rightarrow S_4 \rightarrow L \rightarrow V_{LO} \rightarrow$ ground $\rightarrow V_{HI}$) is significantly reduced with this layout due to the placement of switch $S_{12}$ (top-side) above switches $S_1$ and $S_4$ (bottom-side) rather than the power path of through these switches traversing the full-length of the power-stage. This can be seen in Fig. 5.6, which compares the PCB trace resistance for two different layouts (Rev 0 and Rev 1) of the HISID converter. For the loop mentioned above ($V_{HI} \rightarrow S_{12} \rightarrow C_{3R} \rightarrow S_4 \rightarrow L \rightarrow V_{LO} \rightarrow$ ground $\rightarrow V_{HI}$), the Rev 0 PCB trace resistance is 7.48 $\Omega$. The Rev 1 PCB trace resistance is 1.44 $\Omega$. The DC trace resistance for this power path loop is reduced by 80% compared to the layout in Figs. 5.4a and b. This lower DC resistance is beneficial if we would like to push the converter to higher load currents as conduction loss will increase in this regime.

Figure 5.4: Commutation loop comparison for two different layout configurations of the HISID converter.
Figure 5.5: Single, zoomed in commutation loop shown in schematic and PCB implementation.
Figure 5.6: PCB trace resistance comparison of Rev 0 and Rev 1 layouts.
5.3 Passive Front-End EMI Filter

Passive filters at the input side are commonly used for filtering EMI, post-layout. These filters target noise at specific frequencies that are not already attenuated by the layout practices discussed in the previous section [64]. In this work, only the design of an input filter is demonstrated, though a similar process could be followed for an output filter if necessary. As the main focus of this work is minimizing emissions from the converter on the 48 V bus, only the EMI at the input side of the converter is characterized.

The EMI filter circuit designed for this hardware demonstration is depicted in Fig. 5.7. The filter components are sized based on preliminary conducted EMI measurements of the converter that determine at which specific frequencies the noise peaks occur. It should be noted that to design the filter stages, it is advantageous to measure the CM and DM noise separately. However, the CISPR 25 standard sets limits based on the aggregate EMI, which is a vector combination of CM and DM noise. Therefore, filter design based on separated CM and DM noise should include a safety margin to ensure that the total noise is reduced sufficiently to meet the CISPR 25, Class 5 aggregate EMI standard [65–67]. Here, the safety margin is chosen to be 6 dB.

For switch-mode power supplies, the largest noise peaks typically occur at harmonics of the switching frequency. For this HISID converter, peaks also correspond to the frequency of the switched input current frequency, which is twice that of the switching frequency. These frequencies and the required attenuation guide the selection of the CM and DM filter components. The necessary attenuation is determined from the difference between the noise peak and the EMI standard limit with some safety margin.

![Figure 5.7: Schematic of the front-end EMI filter.](image)
A brief description of the filter design process is as follows [66, 68]. The minimum attenuation needed is determined from the amplitude of the highest noise peak (in this case, the noise peak at the second switching harmonic). The required attenuation, $Atten$, can be found using (5.1), where $A_{pk,noise}$ is the peak amplitude of the noise, occurring at frequency $f_{pk}$, and $A_{pk,limit}$ refers to the CISPR 25, Class 5 peak noise limit. In this case, 1 µV is used as a standard reference voltage for measurement.

$$Atten = A_{pk,noise} - A_{pk,limit} + 6 \text{ dBµV} \quad [\text{dBµV}]$$  \hspace{1cm} (5.1)

The required attenuation can then be converted from dBµV to µV by the following equation (5.2):

$$Atten = 10^{(A_{pk,limit} - 3 - A_{pk,noise})/20} \quad [\mu V].$$  \hspace{1cm} (5.2)

The required cut-off frequency for both the CM and DM filter is then determined by:

$$f_{cutoff} = \frac{f_{pk}}{\sqrt{Atten}} \quad [\text{Hz}].$$  \hspace{1cm} (5.3)

Finally, a combination of the converter input impedance ($Z_{in,conv}$), filter output impedance ($Z_o$), and filter resonant frequency ($f_{cutoff}$, $f_{r,fil} = (2\pi \sqrt{LC})^{-1}$) relationships is used to calculate the filter inductance and capacitance values. For the CM filter, the inductance $L$ refers to $L_{CM}$ and the capacitance $C$ refers to $C_Y$ in Fig. 5.7. Similarly for the DM case, $L$ refers to $L_{DM}$ and the $C$ refers to $C_X$ in the same figure. The “chassis” ground of the CM filter input is the same as that of the line impedance stabilization network (LISN) ground. They are both connected to the large shielding ground plane of the EMI setup. The output of the common mode choke also utilizes a different “signal” ground to improve CM noise filtering and the effectiveness of the choke. The DM filter and converter use a separate power ground to avoid noise coupling into the “chassis” LISN ground [69, 70]. Discrete component values for the CM and DM filter are then chosen.

Both filter impedances and converter input/output impedance impact the filter attenuation. EMI filter and circuit impedances should be mismatched to ensure that the filter can sufficiently attenuate the noise across a large signal frequency range. When the filter component impedance is close to the circuit impedance, the filter may not achieve the required noise mitigation. In this work, a mismatched impedance network between the filter ($Z_o$) and the converter ($Z_{in,conv}$) is assumed where $Z_{in,conv} \neq Z_o$. $Z_R$ is defined as the ratio of $Z_o$ to $Z_{in,conv}$. Adjusting the filter and converter impedance mismatch can also be advantageous when trying to reduce filter passive component volume while keeping the same attenuation at a specific frequency [71]. For example, Fig. 5.8 shows a plot of the relationship between $dB$ attenuation and frequency as the mismatch between source and load impedances is varied. Equation 5.4 is used to calculate the insertion loss for each specific impedance ratio.

$$IL = 20 \cdot \log_{10} \sqrt{(1 - \frac{f}{f_{cutoff}})^2 + (\frac{f}{f_{cutoff}} \cdot Z_R)^2} \quad [\text{dB}].$$  \hspace{1cm} (5.4)
Figure 5.8: Insertion loss plot showing attenuation across frequency for different degrees of impedance mismatch.

The vertical red line on the plot in Fig. 5.8 denotes the target frequency ($f_{pk}$) within the EMI frequency range at which the noise peak of interest occurs. At this target frequency, filter inductance values can be calculated by the following relationship, where $R$ is the effective converter resistance (calculated by input voltage divided by input current, $V_{in}/I_{in}$):

$$L = \frac{Z_R}{f_{cutoff} \cdot 2\pi \cdot R} \text{ [H].}$$  \hspace{1cm} (5.5)

With higher mismatch between source and load impedances, as long as the noise at frequencies of interest is still sufficiently attenuated, the filter inductance can be reduced. It is also well-known that EMI attenuation in practice can also deviate from the calculated value due to additional factors such as parasitic elements, and component de-rating [72]. Therefore, an iterative process is typically required to tune the filter according to measured EMI results.

### 5.4 Spread Spectrum Frequency Modulation (SSFM)

EMI filters do add to overall passive component volume and power solution loss, so we can use clever control techniques to mitigate EMI even further. Spread spectrum, or “dithering”, frequency techniques can be used to further reduce conducted EMI that is generated by fixed-frequency switching schemes [73]. Spread spectrum frequency modulation (SSFM) [22,74–76] is a popular control technique that modulates, or dithers, the converter’s periodic switching
frequency around its original center frequency. This allows us to spread out the original energy of each harmonic about a specified frequency band, providing a wider noise spectrum with lower peak amplitudes. There are many different periodic and random SSFM methods that can be used to achieve this goal and lower EMI [22, 74–77].

In [19], four popular modulation schemes – right-triangular, triangular, trapezoidal, sinusoidal – were analyzed with respect to how they affect the converter’s conducted EMI. Each of these schemes is shown in Fig. 5.9. The fundamental parameters for the frequency modulation profiles are:

- \( f_c \) Center frequency, or nominal frequency about which the switching frequency is dithered.
- \( \Delta f_c \) Step size of frequency dithering.
- \( T_m = \frac{1}{f_m} \) Period/frequency of modulation profile.
- \( A_m \) Maximum deviation of switching frequency from center frequency, \( f_c \).

![Figure 5.9: Modulated switching frequency over time and key parameters for various SSFM schemes: a) triangular, b) right-triangular, c) sinusoidal, d) trapezoidal, and e) pseudo-random.](image)

Each of these modulation profiles follows a periodic pattern. Based on the analysis in [19], this work implements the triangular SSFM scheme and the results are presented in Chapter 6.
The triangular modulation scheme has advantages and disadvantages, which are detailed in [19,77]. Ramping the switching frequency up and down avoids noise spiking at any specific frequency and its harmonics [22]. Triangular SSFM also evenly spreads the energy away from the center frequency, creating a mostly flat energy band which helps to lower noise peaks. Since the switching frequency is being manipulated periodically, one drawback of this method is that both the input and output voltages can acquire a periodic ripple at the modulation frequency. Therefore, the modulation frequency should be chosen to be sufficiently low to avoid too much overlap with the fundamental frequency and its harmonics. Furthermore, a pseudo-random scheme can help overcome the challenges presented by periodic modulation. For the implementation in this work, the dithering step size is ± 0.5 kHz with a maximum deviation of ± 3.5 kHz from the center frequency at a modulation frequency of 700 Hz.

A final consideration is that when implementing any kind of switching frequency modulation, we need to ensure that dithering the switching frequency in this way does not negatively impact hybrid SC converter efficiency. When operating near the resonant frequency, the EMI benefits from SSFM may not outweigh the negative impacts on converter efficiency. However when operating the converter above resonance, the losses are found to be relatively constant with changes in switching frequency [19]. This is because as the switching frequency becomes greater than the resonant frequency, the effective output resistance, $R_{eff}$, approaches its limit, $R_{esr}$, which is the effective series resistance of the power components in the converter. This represents the lowest possible output resistance the converter can achieve [49]. Therefore, switching at a center frequency sufficiently above resonance (at least 50% higher) and with a relatively small dithering band, we can get both the EMI and efficiency benefits.
Chapter 6

Experimental Results

6.1 Experimental Prototype

An 8-to-1 prototype of the HISID converter (Fig. 6.1) was constructed to verify its operation as well as to explore the effectiveness of the different EMI mitigation techniques on the full solution. Fig. 6.2 shows the top and bottom sides of the power stage with key classes of components labeled, and Fig. 6.3 has each of the components labeled in more detail. The experimental prototype measures 84 mm x 41 mm x 8 mm, including gate drive circuitry.

This prototype uses only automotive-qualified components to ensure adherence to the Automotive Electronics Council (AEC) standard (Table 6.1). Within the constraint that all components need to be automotive-qualified, mixed switch technologies (both Si and GaN) are used to optimize for various parameters such as on-state resistance ($R_{DS_{on}}$), drain-to-

Figure 6.1: Image of HISID prototype board (with EMI filter included on bottom side).
source voltage ($V_{DS}$), and gate charge ($Q_G$). Both Cascaded Bootstrap (CBS) and Gate-Driven Charge Pump (GDCP) methods [78] are used for bootstrapping in the gate drive power circuit. Table 6.2 defines the operating parameters for this prototype. The converter flying capacitors ($C_{R1-7}$ and $C_{L1-7}$) and inductor ($L$) values were chosen to give an effective resonant switching frequency of 43 kHz, with the second and third harmonic falling below the lowest relevant EMI frequency band to alleviate passive filter requirements when operating above resonance.

Figure 6.2: Image of top and bottom sides of prototype board with key classes of components labeled.

Figure 6.3: Image of top and bottom sides of prototype board with all components labeled.
<table>
<thead>
<tr>
<th>Component</th>
<th>Mfr. &amp; Part Number</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dickson Power Stage</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Start-up Switches $S_{19} - S_{20}$</td>
<td>EPC EPC2206</td>
<td>GaN, 80 V, 2.5 mΩ</td>
</tr>
<tr>
<td>String Switches $S_{5} - S_{18}$</td>
<td>ON Semiconductor NVTFS002-N04CL</td>
<td>Si, 40 V, 3.5 mΩ</td>
</tr>
<tr>
<td>Bridge Switches $S_{1} - S_{4}$</td>
<td>Infineon IAUC100-N04S6L014</td>
<td>Si, 40 V, 1.4 mΩ</td>
</tr>
<tr>
<td>Flying Capacitors $C_{1x}, C_{2x}$</td>
<td>Murata GRT188R61H225ME13D</td>
<td>X5R, 50 V, 2.2 µF (x4, x8)</td>
</tr>
<tr>
<td></td>
<td>TDK CGA4J3X5R1H475K125AB</td>
<td>X5R, 50 V, 4.7 µF (x4, x6)</td>
</tr>
<tr>
<td></td>
<td>TDK CGA5L3X5R1H685K160AB</td>
<td>X5R, 50 V, 6.8 µF (x4, x5, x7)</td>
</tr>
<tr>
<td>Inductor</td>
<td>Vishay Dale IHLP4040DZERR56M01</td>
<td>0.56 µH, 49 A $I_{sat}$</td>
</tr>
<tr>
<td><strong>Gate Drive</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GaN Driver</td>
<td>Texas Instruments LM5113QDPRRQ1</td>
<td>90 V, high and low-side</td>
</tr>
<tr>
<td>LDO</td>
<td>Microchip MCP1792T-5002H</td>
<td>5.0 V, 100 mA</td>
</tr>
<tr>
<td>Si Gate Driver (and Charge Pump)</td>
<td>Analog Devices Inc. LTC7062IMSE</td>
<td>Dual high-side driver</td>
</tr>
<tr>
<td>Bootstrap Diodes</td>
<td>Nexperia PMEG6002EJ,115</td>
<td>Schottky, 60 V, 200 mA</td>
</tr>
<tr>
<td>Charge Pump Diodes</td>
<td>Diodes Inc. PD3S230L-7</td>
<td>Schottky, 30 V, 2 A</td>
</tr>
<tr>
<td>Charge Pump Capacitors</td>
<td>Murata GRT188R61H225ME13D</td>
<td>X5R, 50 V, 2.2 µF</td>
</tr>
<tr>
<td><strong>EMI Filter</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CM Choke, $L_{CM}$</td>
<td>Eaton CMS1-6-R</td>
<td>32.8 µH, 3.1 A $I_{sat}$</td>
</tr>
<tr>
<td>CM “Y” Capacitance, $C_Y$</td>
<td>TDK CGA4J3X7S2A105K125AB</td>
<td>X7S, 100 V, 1 µF</td>
</tr>
<tr>
<td>DM Inductors, $L_{DM}$</td>
<td>Vishay Dale IHLP4040DZER150M8A</td>
<td>15 µH, 7.7 A $I_{sat}$</td>
</tr>
<tr>
<td>DM “X” Capacitance, $C_X$</td>
<td>TDK CGA4F3X7S2A224K085AE</td>
<td>X7S, 100 V, 0.22 µF</td>
</tr>
<tr>
<td><strong>Controller Board</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FPGA</td>
<td>Terasic Inc. P0466</td>
<td>DE10-Lite, Max10 FPGA</td>
</tr>
</tbody>
</table>
Table 6.2: Converter Operating Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_{HI}$</td>
<td>48 V</td>
<td></td>
</tr>
<tr>
<td>$V_{LO}$</td>
<td>5 V</td>
<td></td>
</tr>
<tr>
<td>$P_{LO,max}$</td>
<td>150 W</td>
<td></td>
</tr>
<tr>
<td>$f_{sw}$</td>
<td>122 kHz</td>
<td></td>
</tr>
<tr>
<td>$f_{res}$</td>
<td>43 kHz</td>
<td></td>
</tr>
<tr>
<td>$L$</td>
<td>0.56 µH</td>
<td></td>
</tr>
<tr>
<td>$C_{in}^*$ (voltage de-rated)</td>
<td>60 µF</td>
<td></td>
</tr>
<tr>
<td>$C_{fly}^*$ (voltage de-rated)</td>
<td>6 µF</td>
<td></td>
</tr>
</tbody>
</table>

6.2 Converter Operation

Both inductor current, $i_L$, and switch-node voltage, $v_{sw}$, waveforms are shown in Fig. 6.4 for above-resonant fixed-ratio, and regulating operation of the hardware prototype at the conditions listed in Table 6.2. Operating the hybrid SC converter above resonance enables reduced rms currents compared to resonant operation. This is beneficial when the converter needs to support higher load currents, where conduction loss dominates [19]. However, output voltage regulation implemented as discussed in Chapter 4 exhibits higher rms currents compared to operation at a fixed-conversion ratio, incurring greater conduction losses in the switches and magnetics. Furthermore, switches $S_1 - S_4$ conduct for longer durations (depicted in Fig. 4.4), again resulting in increased losses. Higher $di/dt$ transitions in the regulating case (due to the linear ramp-down of the inductor current) can also contribute to higher core loss in the inductor [50].
Figure 6.4: (a) Above resonant, and (b) regulating inductor current, $i_L$, and switch-node voltage, $v_{sw}$, measured waveforms for the 8-to-1 discrete hardware prototype at 30 A load current.

6.3 Converter Efficiency and Power Loss Breakdown

Fig. 6.5 shows efficiency versus load curves measured on the HISID converter for a variety of input and output voltages. The nominal operating condition is a 48 V input with regulated 5 V output where the converter is operated at a nominal switching frequency of $\sim 122$ kHz ($\Gamma = 2.8$) with no SSFM employed. The peak efficiency of the full solution including the EMI filter (but not gate-drive power) at a 48 V nominal input voltage and 5 V output is 97.1%, and the full-load efficiency at 30 A is 93.6%. The work in [19] demonstrated that employing SSFM at frequencies much higher than resonance does not affect the efficiency significantly, regardless of the spread spectrum modulation scheme. To demonstrate the converter’s ability to regulate to other voltage rails present in an automotive system, efficiency measurements are also reported for the conversion from 48 V to 3.3 V. Finally, the 48 V rail in a vehicle may vary about the nominal 48 V value if the 48 V source is a battery [18]. Therefore, converter efficiency data is also reported with input voltages of 40 V and 54 V. Each of these cases has peak efficiencies greater than 94% and a full load efficiency of above 90%.
An approximate loss breakdown for full-load operation at the 48 V-to-5 V conversion is shown in Fig. 6.6. Examining the source of loss within the circuit can inform not only component selection and choice of switching frequency, but also the PCB layout as discussed above. The different circuit losses are characterized as follows:

- **Conduction loss** – Calculated as the sum of $I_{\text{rms}}^2 R_x$ for each of the following:
  - Switches: Where $I_{\text{rms}}$ is the rms current through each switch and $R_x$ is the $R_{\text{DSon}}$ of each switch.
  - Capacitors: Where $I_{\text{rms}}$ is the rms current through each flying capacitor and $R_x$ is the $R_{\text{esr}}$ of each flying capacitor.
  - PCB: Where $I_{\text{rms}}$ is the rms current through each flying capacitor or switch and $R_x$ is $R_{\text{PCB}}$, the PCB trace resistance associated with each capacitor or switch.

- **Inductor loss** – Based on ac winding, dc winding, and ac core loss using manufacturer-provided loss calculators.

- **Switching loss** – Based on overlap losses (turn-on and turn-off), output capacitance losses, and MOSFET body diode reverse recovery [79,80].
Figure 6.6: Power loss breakdown for full-load (150 W) regulating (48 V-to-5 V) operation of the HISID hardware prototype.

- EMI filter loss – Based on magnetics (winding and core) and conduction losses of the CM choke and DM inductors.

For the 48 V-to-5 V regulating converter HISID prototype, the output inductor and switch conduction losses dominate the losses at full-load. Furthermore, the PCB losses only contribute to 14% of the total loss, a 2.5 times reduction over the prototype presented in [20] thanks to improved layout techniques. The EMI filter contributes to about 16% of the overall loss, further enforcing the need to use multiple EMI mitigation techniques, such as SSFM, to optimize filter design.

### 6.4 Implementation of EMI Mitigation Techniques - Filter Size Reduction (Rev 0)

Initially, an EMI filter daughter board was designed, built, and connected at the input side of the Rev 0 HISID prototype board [20] to assess the impacts of the filter design on overall solution size and EMI performance. This Rev 0 prototype with the EMI filter daughterboard is shown in Fig. 6.7. This initial prototype was purely used for filter sizing and to achieve certain levels of noise reduction before building the Rev 1 prototype with integrated EMI filter.
CHAPTER 6. EXPERIMENTAL RESULTS

Figure 6.7: Photograph of EMI filter daughter board mounted on Rev 0 power stage prototype. Power stage components are mirrored on the top and bottom side of the PCB.

As a note, all noise measurements reported in this chapter were taken in a laboratory pre-compliance, semi-shielded environment using the Tektronix RSA306b Real-time RF Spectrum Analyzer. Noise measurements in this section are split into their CM and DM sub-components to assess the contribution of each kind of noise to the overall EMI. Preliminary EMI noise levels, for the Rev 0 converter operating at $2.5 \times$ the resonant switching frequency with no EMI filter are shown in Fig. 6.10. In this figure, the CISPR 25, Class 5 limits are plotted as well to show the sizes of the peaks relative to these limits. These initial measurements inform the EMI filter design. The highest noise peaks within the CISPR frequency range — which the filter targets — occur at the second switching harmonics for both CM and DM noise. Applying the designed CM filter with corner frequency 42 kHz and the DM filter at 72 kHz, the noise peaks for CM and DM conducted emissions are reduced by more than 85%, as seen in Figs. 6.8a and 6.8b.
Figure 6.8: Peak CM and DM emissions plots for above-resonant (∼106 kHz) regulating operation with and without EMI filter (22 μH DM inductors) and no SSFM.
Where passive component volume is a primary concern, SSFM can be employed in lieu of or in conjunction with an EMI filter, depending on the noise levels that must be mitigated. The impacts of a trapezoidal SSFM scheme on both CM and DM peak conducted EMI are demonstrated in Fig. 6.9. The key observation is that when SSFM is implemented, both the CM and DM noise peaks are lower and more spread out. This enables optimization of the EMI filter size, and the DM filter inductance is decreased from the initial 22 $\mu$H calculated value to 15 $\mu$H – a reduction of over 30%. With SSFM, the fundamental switching frequency is changing periodically, so the noise peaks will occur at different harmonics along the EMI measurement range at the time of the measurement. In this case, an average EMI measurement (shown as the orange trace in Fig. 6.9) is useful to showcase the true impact of SSFM. Average detection takes the average amplitude of each noise signal across its period. The results in the next section will demonstrate how the Rev 1 prototype with integrated EMI filter performs.
(a) Peak and average common mode (CM) conducted emissions.

(b) Peak and average differential mode (DM) conducted emissions.

Figure 6.9: Peak and average CM and DM emissions plots for above-resonant (∼106 kHz) regulating operation with reduced filter size (15 µH DM inductors) and SSFM enabled.
6.5 Implementation of EMI Mitigation Techniques - Integrated Solution (Rev 1)

The EMI results in this section are for the Rev 1 prototype board, shown in Fig. 6.1. For this Rev 1 prototype, the integrated input-side EMI filter accounts for 7.5% of the overall converter volume. Preliminary (i.e., with no EMI mitigation employed) aggregate EMI noise levels for the regulating converter operating at $2.8 \times$ the resonant switching frequency are shown in Fig. 6.10a. This aggregate EMI was measured on both the positive and return terminals, however, only results for noise measurements on the positive terminal (where noise peaks were higher) are shown. Similar to the process in Section 6.4, the initial CM and DM measurements shown in Fig. 6.10b inform the Rev 1 EMI filter design discussed in Chapter 5. The highest noise peaks within the CISPR frequency range — which the filter targets — occur at the second switching harmonics for both CM and DM noise. Owing to the requirement of split-phase switching, which results in the input voltage being disconnected from the circuit within each phase, the current ripple at the input source is non-zero. This leads to slightly greater DM noise. Despite the impossibility of eliminating the input current ripple, clever circuit configuration (e.g. the implementation of an interleaved-input in this work) serves to reduce this source of DM noise as compared to a single-ended topology.

All of the EMI mitigation techniques discussed in this work are implemented on the converter prototype, much like a typical industry power converter solution. The EMI filter stages are designed as follows: the CM filter has a corner frequency of 42 kHz and the DM filter has a cutoff at 72 kHz. A triangular SSFM scheme with a dithering step size of $\pm 420$ Hz with a maximum deviation of $\pm 2.5$ kHz from the center frequency at a modulation frequency of 780 Hz is employed.

The final aggregate EMI results for the converter with all EMI mitigation techniques employed is shown in Figs. 6.11 and 6.12. By utilizing these EMI mitigation techniques, the noise peaks for the peak conducted emissions are reduced by more than 37%, as seen in Fig. 6.11. A key observation is that when SSFM is implemented, the noise peaks are lower and more spread out. This enables optimization of the EMI filter size.

Moreover, with SSFM, the fundamental switching frequency is also changing with the modulation scheme, so the noise peaks occur at different harmonics along the EMI measurement range at the time of the measurement. In this case, an average EMI measurement is useful to showcase the true impact of SSFM (Fig. 6.12). Average detection takes the average amplitude of each noise signal across its period. The 48 V converter presented in this work with passive EMI filter and SSFM passes CISPR 25, Class 5 limits\(^1\). However, each of these techniques can be employed in lieu of or in conjunction with one another depending on the noise levels that must be mitigated. In applications where passive component volume is a primary concern, SSFM can be used as the primary EMI mitigation technique as showcased

\(^1\)The high frequency noise between 26 MHz and 108 MHz does not come from the power stage itself. This noise is a measured phenomenon of the pre-compliant setup which contains both an electronic power supply and load, and which is not fully enclosed.
in [19].

(a) Peak conducted emissions with no EMI mitigation techniques, measured on both positive and negative terminals of converter.

(b) Peak CM and DM conducted emissions with no EMI mitigation techniques, measured on positive terminal of converter.

Figure 6.10: Peak conducted emissions plots for above-resonant (∼122 kHz) regulating operation with no EMI mitigation operating at 48 V input, 5 V output and 150 W output.
Figure 6.11: Peak conducted emissions with both EMI filter and SSFM employed, measured on positive terminal of converter.

Figure 6.12: Average conducted emissions with both EMI filter and SSFM employed, measured on positive terminal of converter.
Chapter 7
Conclusions

The development of a 48 V distribution bus in both EVs and ICE vehicles opens opportunities for adapting advancements in high-efficiency, high-power-density data center power conversion techniques to automotive applications. However, in-vehicle power electronics also require both robust component selection and qualification for industry EMI standards. This thesis discussed the construction of a 150 W automotive EMI pre-qualified regulating hybrid Dickson switched-capacitor converter for 48 V-to-5 V conversion. EMI mitigation techniques such as improved converter layout, a passive front-end EMI filter, and spread spectrum frequency modulation (SSFM) are discussed in detail and implemented on the hardware prototype. Finally, conducted EMI results showcasing the benefits of the input filter and SSFM implementation are reported demonstrating the interleaved-input hybrid Dickson converter passing CISPR 25, Class 5 EMI specifications at a peak efficiency of 97.1%.

7.1 Future Work

This work can be expanded and improved upon in many ways. One potential extension of the work presented in this thesis would be improving noise attenuation in the high frequencies of the EMI measurement range. Within the CISPR 25, Class 5 limits, the high frequency limits are quite challenging, particularly in the frequency range of 68 to 108 MHz. The parasitics of the filter components can degrade the EMI filter attenuation at such frequencies.

Furthermore, this is a power converter that is meant to be utilized in automotive systems within a vehicle. Therefore, future work also includes qualifying the solution for startup and shutdown sequences as well as input and output transient response. These sequences and responses can be evaluated against industry specifications as well – for example, ensuring the output voltage does not deviate beyond a certain value under a given load step and slew rate. For these novel hybrid SC topologies, this will likely require investigation into control techniques beyond the conventional. As a final step, this converter can be taken to an EMI compliant setup at a regulatory compliance testing facility to make a comparison with the pre-compliant test setup EMI performance.
On the more theoretical side, many of the multilevel topologies, including the HISID converter, are composed of identical cells of switches and capacitors that make up the different “levels”. It would be useful to develop a law for how common and differential mode EMI scale with HISID converter level count.
Bibliography


Appendix A

Filter Insertion Loss Calculator

Included here is the Matlab file used to calculate the insertion loss for EMI filter design.

```
clc
clear all
close all
fsweep = logspace(3,6,200);
fc = 20000; %corner frequency
ZcR = 0.2; %ratio of impedances
IL = 20*log10(sqrt((1-fsweep/fc).^2+(fsweep/fc*ZcR).^2)); %insertion loss calc
R = 14.5;

hold on; grid on;
ZcR2 = 1; %source and load Z perfectly matched
fc2 = 50e3;
IL2 = 20*log10(sqrt(((1-fsweep/fc).^2+(fsweep/fc*ZcR2).^2))); %insertion loss calc
semilogx(fsweep,IL2, 'LineWidth', 2);
L2 = ZcR2/fc2/2/pi*R

ZcR3 = 0.8;
fc3 = 40e3;
IL3 = 20*log10(sqrt((1-fsweep/fc).^2+(fsweep/fc*ZcR3).^2)); %insertion loss calc
semilogx(fsweep,IL3, 'LineWidth', 2);
L3 = ZcR3/fc3/2/pi*R
```
ftarget = 243e3;
IL_target1 = 20*log10(sqrt((1-ftarget/fc).^2+(ftarget/fc*ZcR).^2));
disp(['IL@160kHz:',num2str(IL_target1)])

L1 = ZcR/fc/2/pi*R
semilogx(fsweep,IL, 'LineWidth', 2);
plot(ftarget, IL_target1, 'o', 'LineWidth', 3, 'Color','r');
xline(243000,'Color','r')
xlabel('Frequency [Hz]')
ylabel('Attenuation [dB]')
h = legend('Perfect Match', '20% Mismatch', '80% Mismatch', 'Location', 'southeast');
set(h,'FontSize',12);
set(gca,'FontSize',16, 'FontName','Times New Roman');
Appendix B

HISID Converter Hardware
Prototype Circuit Schematic and PCB Layout

The schematic and PCB layout for the 8-to-1 HISID converter are included below.

Schematic
Figure B.1: Top level circuit schematic.
Figure B.2: Output inductor circuit schematic.
Figure B.3: EMI filter circuit schematic.
Figure B.4: Power stage circuit schematic.
Figure B.5: GaN gate driver circuit schematic (for switches S19 and S20 in the design).
Figure B.6: GDCP Si gate driver circuit schematic (for switches S3, S4, S18, and S17 in the design).
Figure B.7: CBS Si gate driver circuit schematic (for switches S5-S8 and S11, S12, S15, and S16 in the design).
Figure B.8: GDCP Si gate driver circuit schematic (for switches S9, S10, S13, and S14 in the design).
Figure B.9: Half bridge gate driver circuit schematic (for switches S1 and S2 in the design).
Figure B.10: Voltage sensing circuit schematic.
PCB Layout

The PCB layers for the 8-to-1 HISID converter prototype are shown below.
Figure B.12: First inner layer of PCB.
Figure B.13: Second inner layer of PCB.
Figure B.16: Bottom layer of PCB.
Appendix C

FPGA Code for Converter PWMs and SSFM

The QPF file for the HISID converter control is included here. It has code for PWM generation as well as output voltage regulation and spread spectrum frequency modulation (SSFM) of the converter. The code also implements checks to ensure that no gate signals are overlapping as they are being modulated.

```verbatim
module pwm_test(

//////////// CLOCK //////////
inherited ADC_CLK_10,
inherited MAX10_CLK1_50,
inherited MAX10_CLK2_50,

//////////// SDRAM //////////
output [12:0] DRAM_ADDR,
output [1:0] DRAM_BA,
output DRAM_CAS_N,
output DRAM_CKE,
output DRAMCLK,
output DRAM_CS_N,
inout [15:0] DRAM_DQ,
inout DRAM_LDQM,
output DRAM_RAS_N,
output DRAM_UDQM,
```
output DRAM_WE_N,

//////////// SEG7 //////////
output [7:0] HEX0,
output [7:0] HEX1,
output [7:0] HEX2,
output [7:0] HEX3,
output [7:0] HEX4,
output [7:0] HEX5,

//////////// KEY //////////
input [1:0] KEY,

//////////// LED //////////
output [9:0] LEDR,

//////////// SW //////////
input [9:0] SW,

//////////// Accelerometer //////////
output GSENSOR_CS_N,
inout [2:1] GSENSOR_INT,
output GSENSOR_SCLK,
inout GSENSOR_SDI,
inout GSENSOR_SDO,

//////////// Arduino //////////
inout [15:0] ARDUINO_IO,
inout ARDUINO_RESET_N,

//////////// GPIO, GPIO connect to GPIO Default //////////
inout [35:0] GPIO
);

//=======================================================
// REG/WIRE declarations
//=======================================================
wire [14:0] data;
wire [7:0] address;
wire CLK_200;
APPENDIX C. FPGA CODE FOR CONVERTER PWMS AND SSFM

// regreg1/*synthesispreserve*/;
reg pwm_enable;
reg ss_enable /*synthesispreserve*/;
reg ss_status;
reg reg_enable /*synthesispreserve*/;
reg reg_status;
reg py_flag;
reg [15:0] count1 = 11'b0;
reg [15:0] count = 11'b0;
reg REG_pin = 0;
integer shift_reg = 0/*synthesispreserve*/;

// regcount_vf/*synthesispreserve*/;
//reg [11:0] count_vf = 11'b0 /*synthesispreserve*/;
reg [15:0] count_vf = 11'b0;
reg [1:0] count_sync = 2'b0;
reg flag_update =1'b0;
reg [15:0] period = 13'd1000;
reg [15:0] period_new = 13'd1000;
reg [9:0] shift_1b = 0;
reg [9:0] shift_1a = 0;
reg [9:0] shift_2b = 0;
reg [9:0] shift_2a = 0;
reg [9:0] shift_a_37 = 0;
reg [9:0] shift_b_37 = 0;
reg [9:0] shift_a_47 = 0;
reg [9:0] shift_b_47 = 0;
reg [9:0] shift_a_51 = 0;
reg [9:0] shift_b_51 = 0;
reg [9:0] shift_a_53 = 0;
reg [9:0] shift_b_53 = 0;
reg [9:0] shift_period = 0;
(* preserve *) reg [15:0] Tdelay_1_new /*synthesispreserve*/;
(* preserve *) reg [15:0] Ton_1_new;
(* preserve *) reg [15:0] Tdelay_2_new;
(* preserve *) reg [15:0] Ton_2_new;
(* preserve *) reg [15:0] Tdelay_3_new;
(* preserve *) reg [15:0] Ton_3_new;
(* preserve *) reg [15:0] Tdelay_4_new;
(* preserve *) reg [15:0] Ton_4_new;
(* preserve *) reg [15:0] Tdelay_5_new;
(* preserve *) reg [15:0] Ton_5_new;
APPENDIX C. FPGA CODE FOR CONVERTER PWMS AND SSFM

104 (* preserve *) reg [15:0] Tdelay_6_new;
105 (* preserve *) reg [15:0] Ton_6_new;
106 (* preserve *) reg pwm_enable_new;
107 reg [15:0] Tdelay_1;
108 reg [15:0] Ton_1;
109 reg [15:0] Tdelay_2;
110 reg [15:0] Ton_2;
111 reg [15:0] Tdelay_3;
112 reg [15:0] Ton_3;
113 reg [15:0] Tdelay_4;
114 reg [15:0] Ton_4;
115 reg [15:0] Tdelay_5;
116 reg [15:0] Ton_5;
117 reg [15:0] Tdelay_6;
118 reg [15:0] Ton_6;
119 (* preserve *) reg [15:0] vo_set;
120 reg [15:0] Treg = 0;
121 (* preserve *) reg [15:0] Treg_new = 0;

122 reg [9:0] shift_1ab = 0;
123 reg [9:0] shift_2ab = 0;
124 reg [9:0] shift_1ab_37 = 0;
125 reg [9:0] shift_2ab_37 = 0;
126 reg [9:0] shift_1ab_47 = 0;
127 reg [9:0] shift_2ab_47 = 0;
128 reg [9:0] shift_1ab_51 = 0;
129 reg [9:0] shift_2ab_51 = 0;
130 reg [9:0] shift_1ab_53 = 0;
131 reg [9:0] shift_2ab_53 = 0;
132 reg [9:0] shift_period1 = 0;
133 reg [9:0] shift_period2 = 0;
134 reg [9:0] shift_period3 = 0;
135 reg [9:0] shift_period4 = 0;
136 reg [9:0] shift_period5 = 0;
137 reg [9:0] shift_period6 = 0;
138 reg [9:0] shift_period_37 = 0;
139 reg [9:0] shift_period_47 = 0;
140 reg [9:0] shift_period_51 = 0;
141 reg [9:0] shift_period_53 = 0;

142 reg [14:0] period1_dwn1 = 1000;
APPENDIX C. FPGA CODE FOR CONVERTER PWMS AND SSFM

```
146 reg [11:0] Td2_dwn1 = 500;
147 reg [11:0] Td4_dwn1 = 500;
148 reg [11:0] Td6_dwn1 = 500;
149 reg [11:0] Ton3_dwn1 = 500;
150 reg [11:0] Ton5_dwn1 = 500;
151 reg [11:0] Ton1_dwn1 = 500;
152 reg [11:0] Ton4_dwn1 = 500;
153 reg [11:0] Ton6_dwn1 = 500;
154 reg [11:0] Ton2_dwn1 = 500;
155
156 reg [14:0] period1_dwn2 = 1000;
157 reg [11:0] Td2_dwn2 = 500;
158 reg [11:0] Td4_dwn2 = 500;
159 reg [11:0] Td6_dwn2 = 500;
160 reg [11:0] Ton3_dwn2 = 500;
161 reg [11:0] Ton5_dwn2 = 500;
162 reg [11:0] Ton1_dwn2 = 500;
163 reg [11:0] Ton4_dwn2 = 500;
164 reg [11:0] Ton6_dwn2 = 500;
165 reg [11:0] Ton2_dwn2 = 500;
166
167 reg [14:0] period1_dwn3 = 1000;
168 reg [11:0] Td2_dwn3 = 500;
169 reg [11:0] Td4_dwn3 = 500;
170 reg [11:0] Td6_dwn3 = 500;
171 reg [11:0] Ton3_dwn3 = 500;
172 reg [11:0] Ton5_dwn3 = 500;
173 reg [11:0] Ton1_dwn3 = 500;
174 reg [11:0] Ton4_dwn3 = 500;
175 reg [11:0] Ton6_dwn3 = 500;
176 reg [11:0] Ton2_dwn3 = 500;
177
178 reg [14:0] period1_dwn4 = 1000;
179 reg [11:0] Td2_dwn4 = 500;
180 reg [11:0] Td4_dwn4 = 500;
181 reg [11:0] Td6_dwn4 = 500;
182 reg [11:0] Ton3_dwn4 = 500;
183 reg [11:0] Ton5_dwn4 = 500;
184 reg [11:0] Ton1_dwn4 = 500;
185 reg [11:0] Ton4_dwn4 = 500;
186 reg [11:0] Ton6_dwn4 = 500;
187 reg [11:0] Ton2_dwn4 = 500;
```
APPENDIX C. FPGA CODE FOR CONVERTER PWMS AND SSFM

reg [14:0] period1_dwn5 = 1000;
reg [11:0] Td2_dwn5 = 500;
reg [11:0] Td4_dwn5 = 500;
reg [11:0] Td6_dwn5 = 500;
reg [11:0] Ton3_dwn5 = 500;
reg [11:0] Ton5_dwn5 = 500;
reg [11:0] Ton1_dwn5 = 500;
reg [11:0] Ton4_dwn5 = 500;
reg [11:0] Ton6_dwn5 = 500;
reg [11:0] Ton2_dwn5 = 500;
reg [14:0] period1_dwn6 = 1000;
reg [11:0] Td2_dwn6 = 500;
reg [11:0] Td4_dwn6 = 500;
reg [11:0] Td6_dwn6 = 500;
reg [11:0] Ton3_dwn6 = 500;
reg [11:0] Ton5_dwn6 = 500;
reg [11:0] Ton1_dwn6 = 500;
reg [11:0] Ton4_dwn6 = 500;
reg [11:0] Ton6_dwn6 = 500;
reg [11:0] Ton2_dwn6 = 500;
reg [14:0] period_dwn_37 = 1000;
reg [11:0] Td2_dwn1_37 = 500;
reg [11:0] Td4_dwn1_37 = 500;
reg [11:0] Td6_dwn1_37 = 500;
reg [11:0] Ton3_dwn1_37 = 500;
reg [11:0] Ton5_dwn1_37 = 500;
reg [11:0] Ton1_dwn1_37 = 500;
reg [11:0] Ton4_dwn1_37 = 500;
reg [11:0] Ton6_dwn1_37 = 500;
reg [11:0] Ton2_dwn1_37 = 500;
reg [14:0] period_dwn_47 = 1000;
reg [11:0] Td2_dwn1_47 = 500;
reg [11:0] Td4_dwn1_47 = 500;
reg [11:0] Td6_dwn1_47 = 500;
reg [11:0] Ton3_dwn1_47 = 500;
reg [11:0] Ton5_dwn1_47 = 500;
reg [11:0] Ton1_dwn1_47 = 500;
reg [11:0] Ton4_dwn1_47 = 500;
APPENDIX C. FPGA CODE FOR CONVERTER PWMS AND SSFM

230   reg [11:0] Ton6_dwn1_47 = 500;
231   reg [11:0] Ton2_dwn1_47 = 500;
232
233   reg [14:0] period_dwn_51 = 1000;
234   reg [11:0] Td2_dwn1_51 = 500;
235   reg [11:0] Td4_dwn1_51 = 500;
236   reg [11:0] Td6_dwn1_51 = 500;
237   reg [11:0] Ton3_dwn1_51 = 500;
238   reg [11:0] Ton5_dwn1_51 = 500;
239   reg [11:0] Ton1_dwn1_51 = 500;
240   reg [11:0] Ton4_dwn1_51 = 500;
241   reg [11:0] Ton6_dwn1_51 = 500;
242   reg [11:0] Ton2_dwn1_51 = 500;
243
244   reg [14:0] period_dwn_53 = 1000;
245   reg [11:0] Td2_dwn1_53 = 500;
246   reg [11:0] Td4_dwn1_53 = 500;
247   reg [11:0] Td6_dwn1_53 = 500;
248   reg [11:0] Ton3_dwn1_53 = 500;
249   reg [11:0] Ton5_dwn1_53 = 500;
250   reg [11:0] Ton1_dwn1_53 = 500;
251   reg [11:0] Ton4_dwn1_53 = 500;
252   reg [11:0] Ton6_dwn1_53 = 500;
253   reg [11:0] Ton2_dwn1_53 = 500;
254
255
256   reg [14:0] period1_up1 = 1000;
257   reg [11:0] Td2_up1 = 500;
258   reg [11:0] Td4_up1 = 500;
259   reg [11:0] Td6_up1 = 500;
260   reg [11:0] Ton3_up1 = 500;
261   reg [11:0] Ton5_up1 = 500;
262   reg [11:0] Ton1_up1 = 500;
263   reg [11:0] Ton4_up1 = 500;
264   reg [11:0] Ton6_up1 = 500;
265   reg [11:0] Ton2_up1 = 500;
266
267   reg [14:0] period1_up2 = 1000;
268   reg [11:0] Td2_up2 = 500;
269   reg [11:0] Td4_up2 = 500;
270   reg [11:0] Td6_up2 = 500;
271   reg [11:0] Ton3_up2 = 500;
APPENDIX C. FPGA CODE FOR CONVERTER PWMS AND SSFM

272 reg [11:0] Ton5_up2 = 500;
273 reg [11:0] Ton1_up2 = 500;
274 reg [11:0] Ton4_up2 = 500;
275 reg [11:0] Ton6_up2 = 500;
276 reg [11:0] Ton2_up2 = 500;
277
278 reg [14:0] period1_up3 = 1000;
279 reg [11:0] Td2_up3 = 500;
280 reg [11:0] Td4_up3 = 500;
281 reg [11:0] Td6_up3 = 500;
282 reg [11:0] Ton3_up3 = 500;
283 reg [11:0] Ton5_up3 = 500;
284 reg [11:0] Ton1_up3 = 500;
285 reg [11:0] Ton4_up3 = 500;
286 reg [11:0] Ton6_up3 = 500;
287 reg [11:0] Ton2_up3 = 500;
288
289 reg [14:0] period1_up4 = 1000;
290 reg [11:0] Td2_up4 = 500;
291 reg [11:0] Td4_up4 = 500;
292 reg [11:0] Td6_up4 = 500;
293 reg [11:0] Ton3_up4 = 500;
294 reg [11:0] Ton5_up4 = 500;
295 reg [11:0] Ton1_up4 = 500;
296 reg [11:0] Ton4_up4 = 500;
297 reg [11:0] Ton6_up4 = 500;
298 reg [11:0] Ton2_up4 = 500;
299
300 reg [14:0] period1_up5 = 1000;
301 reg [11:0] Td2_up5 = 500;
302 reg [11:0] Td4_up5 = 500;
303 reg [11:0] Td6_up5 = 500;
304 reg [11:0] Ton3_up5 = 500;
305 reg [11:0] Ton5_up5 = 500;
306 reg [11:0] Ton1_up5 = 500;
307 reg [11:0] Ton4_up5 = 500;
308 reg [11:0] Ton6_up5 = 500;
309 reg [11:0] Ton2_up5 = 500;
310
311 reg [14:0] period1_up6 = 1000;
312 reg [11:0] Td2_up6 = 500;
313 reg [11:0] Td4_up6 = 500;
APPENDIX C. FPGA CODE FOR CONVERTER PWMS AND SSFM

```verilog
reg [11:0] Td6_up6 = 500;
reg [11:0] Ton3_up6 = 500;
reg [11:0] Ton5_up6 = 500;
reg [11:0] Ton1_up6 = 500;
reg [11:0] Ton4_up6 = 500;
reg [11:0] Ton6_up6 = 500;
reg [11:0] Ton2_up6 = 500;
reg [14:0] period_up_37 = 1000;
reg [11:0] Td2_up1_37 = 500;
reg [11:0] Td4_up1_37 = 500;
reg [11:0] Td6_up1_37 = 500;
reg [11:0] Ton3_up1_37 = 500;
reg [11:0] Ton5_up1_37 = 500;
reg [11:0] Ton1_up1_37 = 500;
reg [11:0] Ton4_up1_37 = 500;
reg [11:0] Ton6_up1_37 = 500;
reg [11:0] Ton2_up1_37 = 500;
reg [14:0] period_up_47 = 1000;
reg [11:0] Td2_up1_47 = 500;
reg [11:0] Td4_up1_47 = 500;
reg [11:0] Td6_up1_47 = 500;
reg [11:0] Ton3_up1_47 = 500;
reg [11:0] Ton5_up1_47 = 500;
reg [11:0] Ton1_up1_47 = 500;
reg [11:0] Ton4_up1_47 = 500;
reg [11:0] Ton6_up1_47 = 500;
reg [11:0] Ton2_up1_47 = 500;
reg [14:0] period_up_51 = 1000;
reg [11:0] Td2_up1_51 = 500;
reg [11:0] Td4_up1_51 = 500;
reg [11:0] Td6_up1_51 = 500;
reg [11:0] Ton3_up1_51 = 500;
reg [11:0] Ton5_up1_51 = 500;
reg [11:0] Ton1_up1_51 = 500;
reg [11:0] Ton4_up1_51 = 500;
reg [11:0] Ton6_up1_51 = 500;
reg [11:0] Ton2_up1_51 = 500;
reg [14:0] period_up_53 = 1000;
```
APPENDIX C. FPGA CODE FOR CONVERTER PWMS AND SSFM

```verilog
reg [11:0] Td2_up1_53 = 500;
reg [11:0] Td4_up1_53 = 500;
reg [11:0] Td6_up1_53 = 500;
reg [11:0] Ton3_up1_53 = 500;
reg [11:0] Ton5_up1_53 = 500;
reg [11:0] Ton1_up1_53 = 500;
reg [11:0] Ton4_up1_53 = 500;
reg [11:0] Ton6_up1_53 = 500;
reg [11:0] Ton2_up1_53 = 500;

(* preserve *) reg [5:0] debug = 0 /*synthesispreserve*/;

assign cpu_enable = KEY[0];
assign UART_RXD = GPIO[34];
assign GPIO[35] = UART_TXD;

wire [11:0] count_val;
assign GPIO[0] = PWM1;  //probe pin
assign GPIO[1] = PWM1;
assign GPIO[3] = PWM2;
assign GPIO[5] = PWM3;
assign GPIO[7] = PWM4;
assign GPIO[9] = PWM5;

reg [2:0] TEST;
assign GPIO[14:12] = TEST;

reg [2:0] TEST1;
assign GPIO[30:28] = TEST1;
assign GPIO[23] = REG_write;
assign GPIO[25] = SS_write;
assign GPIO[27] = pwm_update;  //probe pin

wire PWM2, PWM3, PWM1;

// clock bridge
wire sys_clk;

//===----------------------------------------------------------------------==
```
// Structural coding
//=======================================================
sopc u0 (
  .clk_clk (MAX10_CLK1_50),
  .pio_address_external_connection_export (address),
  .pio_data_external_connection_export (data),
  .pio_led_external_connection_export (LEDR),
  .pio_ssen_external_connection_export (SS_write),
  .pio_regen_external_connection_export (REG_write),
  .pio_update_external_connection_export (pwm_update),
  .pio_write_external_connection_export (pwm_write),
  .reset_reset_n (cpu_enable),
  .uart_0_external_connection_rxd (UART_RXD),
  .uart_0_external_connection_txd (UART_TXD),
  .altpll_0_c1_clk (CLK_200),
  .clock_bridge_0_out_clk_clk (sys_clk),
  .adc_vout_command_valid (command_valid),
  .adc_vout_command_channel (command_channel),
  .adc_vout_command_startofpacket (command_startofpacket),
  .adc_vout_command_endofpacket (command_endofpacket),
  .adc_vout_command_ready (command_ready),
  .adc_vout_response_valid (response_valid),
  .adc_vout_response_channel (response_channel),
  .adc_vout_response_data (response_data),
  .adc_vout_response_startofpacket (response_startofpacket),
  .adc_vout_response_endofpacket (response_endofpacket),
);

// command
wire command_valid;
wire [4:0] command_channel;
wire command_startofpacket;
wire command_endofpacket;
wire command_ready;

// continued send command
assign command_startofpacket = 1'b1; // // ignore in altera_adc_control core
assign command_endofpacket = 1'b1; // // ignore in altera_adc_control core
assign command_valid = 1'b1; //
assign command_channel = SW[2:0]+1; // SW2/SW1/SW0 down: map to arduino ADC_IN0
```vhdl
always @ (posedge sys_clk)
begin
  if (response_valid)
  begin
    adc_sample_data <= response_data;
    cur_adc_ch <= response_channel;
    vo_meas <= response_data * 2 * 2500 / 4095;
  end
end

// adc_sample_data: hold 12-bit adc sample value
// Vout = Vin (12-bit x2 x 2500 / 4095)
//assign LEDR[9:0] = vo_meas[12:3]; // led is high active

assign HEX5[7] = 1'b1; // low active
assign HEX4[7] = 1'b1; // low active
assign HEX3[7] = 1'b0; // low active
assign HEX2[7] = 1'b1; // low active
assign HEX1[7] = 1'b1; // low active
assign HEX0[7] = 1'b1; // low active

SEG7_LUT SEG7_LUT_ch (oSEG(HEX5),
  .DIG(SW[2:0]))
);
assign HEX4 = 8'b10111111;
```
APPENDIX C. FPGA CODE FOR CONVERTER PWMS AND SSFM

482
483 SEG7_LUT SEG7_LUT_v (  
484 .oSEG(HEX3),  
485 .iDIG(vo_meas/1000)  
486 );
487
488 SEG7_LUT SEG7_LUT_v_1 (  
489 .oSEG(HEX2),  
490 .iDIG(vo_meas/100 - (vo_meas/1000)*10)  
491 );
492
493 SEG7_LUT SEG7_LUT_v_2 (  
494 .oSEG(HEX1),  
495 .iDIG(vo_meas/10 - (vo_meas/100)*10)  
496 );
497
498 SEG7_LUT SEG7_LUT_v_3 (  
499 .oSEG(HEX0),  
500 .iDIG(vo_meas - (vo_meas/10)*10)  
501 );
502 //=======================================================================
503 // Communicate with PWM module
504 //=======================================================================
505
506 // 00h: OFF
507 // 01h: ON
508 // 02h: period
509 // 11h: Tdelay_1
510 // 12h: Ton_1
511 // 13h: Tdelay_2
512 // 14h: Ton_2
513 // 15h: Tdelay_3
514 // 16h: Ton_3
515 // 17h: Tdelay_4
516 // 18h: Ton_4
517 // 19h: Tdelay_5
518 // 1Ah: Ton_5
519 // 1Bh: Tdelay_6
520 // 1Ch: Ton_6
521 // 1Dh: Tdelay_7
522 // 1 Eh: Ton_7
523 // 1Fh: Tdelay_8
assign count_val = shift_reg;

always @(posedge pwm_write)
begin
  case (address)
    8'h00: pwm_enable <= 1'b0;
    8'h01: pwm_enable <= 1'b1;
    8'h02: period <= data;
    8'h11: Tdelay_1 <= data;
    8'h12: Ton_1 <= data;
    8'h13: Tdelay_2 <= data;
    8'h14: Ton_2 <= data;
    8'h15: Tdelay_3 <= data;
    8'h16: Ton_3 <= data;
    8'h17: Tdelay_4 <= data;
    8'h18: Ton_4 <= data;
    8'h19: Tdelay_5 <= data;
    8'h1A: Ton_5 <= data;
    8'h1B: Tdelay_6 <= data;
    8'h1C: Ton_6 <= data;
    8'h2B: ss_enable <= 1'b0;
    8'h2C: ss_enable <= 1'b1;
    8'h2D: Treg <= data;
    8'h2E: reg_enable <= 1'b0;
    8'h2F: reg_enable <= 1'b1;
    8'h30: vo_set <= data;
  endcase
end
APPENDIX C. FPGA CODE FOR CONVERTER PWMS AND SSFM

8'h31: REG_pin <= data;
endcase
end
always @(posedge ready) // enable reg
begin
if (REG_pin !=0)
beg
reg_status <= 1;
end
else
begin
reg_status = 0;
end
end

wire checkTon_SS_inc;
wire checkTon_SS_dec;
wire checkTdel_SS_inc;
wire checkTdel_SS_dec;
assign checkTon_SS_inc = 1;
assign checkTdel_SS_inc = 1;
assign checkTon_SS_dec = (Ton_3_new - 106 >= 4) && (Ton_4_new - 106 >= 4);
assign checkTdel_SS_dec = 1;

wire check_SS_inc;
wire check_SS_dec;
assign check_SS_inc = checkTon_SS_inc && checkTdel_SS_inc;
assign check_SS_dec = checkTon_SS_dec && checkTdel_SS_dec;

wire SS_EN;
assign SS_EN = SS_write == 1 && check_SS_inc && check_SS_dec;
always @(posedge ready) // set up SS shifts
begin
if(SS_EN)// spread spectrum enable button from python GUI
begin
// ss_status <= ~ss_status;
shift_1a <= 5;
shift_1b <= 5;
shift_2a <= 5;
shift_2b <= 5;
APPENDIX C. FPGA CODE FOR CONVERTER PWMS AND SSFM

```vhdl
shift_a_37 <= 32; //use for 1a and 2a
shift_b_37 <= 32; //use for 1b and 2b
shift_a_47 <= 47; //use for 1a and 2a
shift_b_47 <= 47; //use for 1b and 2b
shift_a_51 <= 51;
shift_b_51 <= 51;
shift_a_53 <= 51;
shift_b_53 <= 51;

end

else if (SS_write == 0) begin
  // ss_status = 0;
  shift_1a <= 0;
  shift_1b <= 0;
  shift_2a <= 0;
  shift_2b <= 0;
  shift_a_37 <= 0; //use for 1a and 2a
  shift_b_37 <= 0; //use for 1b and 2b
  shift_a_47 <= 0; //use for 1a and 2a
  shift_b_47 <= 0; //use for 1b and 2b
  shift_a_51 <= 0;
  shift_b_51 <= 0;
  shift_a_53 <= 0;
  shift_b_53 <= 0;

end

else begin
  // ss_status = 0;
  shift_1a <= shift_1a;
  shift_1b <= shift_1b;
  shift_2a <= shift_2a;
  shift_2b <= shift_2b;
  shift_a_37 <= shift_a_37; //use for 1a and 2a
  shift_b_37 <= shift_b_37; //use for 1b and 2b
  shift_a_47 <= shift_a_47; //use for 1a and 2a
  shift_b_47 <= shift_b_47; //use for 1b and 2b
  shift_a_51 <= shift_a_51;
  shift_b_51 <= shift_b_51;
  shift_a_53 <= shift_a_53;
  shift_b_53 <= shift_b_53;
```

always @(posedge count_flag) //update pwm reg values from python gui
begin
if (pwm_update)
begin
shift_1ab <= shift_1a + shift_1b;
shift_2ab <= shift_2a + shift_2b;
shift_period1 <= (shift_1ab + shift_2ab);
shift_period2 <= (shift_1ab + shift_1ab +
shift_2ab + shift_2ab);
shift_period3 <= (shift_1ab + shift_1ab +
shift_1ab + shift_2ab + shift_2ab + shift_2ab);
shift_period4 <= (shift_1ab + shift_1ab +
shift_1ab + shift_1ab + shift_2ab + shift_2ab +
shift_2ab + shift_2ab);
shift_period5 <= (shift_1ab + shift_1ab +
shift_1ab + shift_1ab + shift_2ab + shift_2ab +
shift_2ab + shift_2ab + shift_2ab + shift_2ab);
shift_period6 <= (shift_1ab + shift_1ab +
shift_1ab + shift_1ab + shift_2ab + shift_2ab +
shift_2ab + shift_2ab + shift_2ab + shift_2ab + shift_2ab + shift_2ab);
shift_1ab_37 <= shift_a_37 + shift_b_37;
shift_2ab_37 <= shift_a_37 + shift_b_37;
shift_period_37 <= (shift_1ab_37 + shift_2ab_37);
shift_1ab_47 <= shift_a_47 + shift_b_47;
shift_2ab_47 <= shift_a_47 + shift_b_47;
shift_period_47 <= (shift_1ab_47 + shift_2ab_47);
shift_1ab_51 <= shift_a_51 + shift_b_51;
shift_2ab_51 <= shift_a_51 + shift_b_51;
shift_period_51 <= (shift_1ab_51 + shift_2ab_51);
shift_1ab_53 <= shift_a_53 + shift_b_53;
shift_2ab_53 <= shift_a_53 + shift_b_53;
shift_period_53 <= (shift_1ab_53 + shift_2ab_53);
\begin{verbatim}
APPENDIX C. FPGA CODE FOR CONVERTER PWMS AND SSFM

period_up_37 <= period + shift_period_37;
Td2_up1_37 <= Tdelay_2 + shift_1ab_37;
Td4_up1_37 <= Tdelay_4 + shift_1ab_37;
Td6_up1_37 <= Tdelay_6 + shift_1ab_37;
Ton3_up1_37 <= Ton_3 + shift_a_37;
Ton5_up1_37 <= Ton_5 + shift_1ab_37;
Ton1_up1_37 <= Ton_1 + shift_1ab_37;
Ton4_up1_37 <= Ton_4 + shift_a_37;
Ton6_up1_37 <= Ton_6 + shift_2ab_37;
Ton2_up1_37 <= Ton_2 + shift_2ab_37;

period_up_47 <= period + shift_period_47;
Td2_up1_47 <= Tdelay_2 + shift_1ab_47;
Td4_up1_47 <= Tdelay_4 + shift_1ab_47;
Td6_up1_47 <= Tdelay_6 + shift_1ab_47;
Ton3_up1_47 <= Ton_3 + shift_a_47;
Ton5_up1_47 <= Ton_5 + shift_1ab_47;
Ton1_up1_47 <= Ton_1 + shift_1ab_47;
Ton4_up1_47 <= Ton_4 + shift_a_47;
Ton6_up1_47 <= Ton_6 + shift_2ab_47;
Ton2_up1_47 <= Ton_2 + shift_2ab_47;

period_up_51 <= period + shift_period_51;
Td2_up1_51 <= Tdelay_2 + shift_1ab_51;
Td4_up1_51 <= Tdelay_4 + shift_1ab_51;
Td6_up1_51 <= Tdelay_6 + shift_1ab_51;
Ton3_up1_51 <= Ton_3 + shift_a_51;
Ton5_up1_51 <= Ton_5 + shift_1ab_51;
Ton1_up1_51 <= Ton_1 + shift_1ab_51;
Ton4_up1_51 <= Ton_4 + shift_a_51;
Ton6_up1_51 <= Ton_6 + shift_2ab_51;
Ton2_up1_51 <= Ton_2 + shift_2ab_51;

period_up_53 <= period + shift_period_53;
Td2_up1_53 <= Tdelay_2 + shift_1ab_53;
Td4_up1_53 <= Tdelay_4 + shift_1ab_53;
Td6_up1_53 <= Tdelay_6 + shift_1ab_53;
Ton3_up1_53 <= Ton_3 + shift_a_53;
Ton5_up1_53 <= Ton_5 + shift_1ab_53;
Ton1_up1_53 <= Ton_1 + shift_1ab_53;
Ton4_up1_53 <= Ton_4 + shift_a_53;
\end{verbatim}
APPENDIX C. FPGA CODE FOR CONVERTER PWMS AND SSFM

```vhdl
734  Ton6_up1_53 <= Ton_6 + shift_2ab_53;
735  Ton2_up1_53 <= Ton_2 + shift_2ab_53;
736
737  period1_up1 <= period + shift_period1;
738  Td2_up1 <= Tdelay_2 + shift_1ab;
739  Td4_up1 <= Tdelay_4 + shift_1ab;
740  Td6_up1 <= Tdelay_6 + shift_1ab;
741  Ton3_up1 <= Ton_3 + shift_1a;
742  Ton5_up1 <= Ton_5 + shift_1ab;
743  Ton1_up1 <= Ton_1 + shift_1ab;
744  Ton4_up1 <= Ton_4 + shift_2a;
745  Ton6_up1 <= Ton_6 + shift_2ab;
746
747  period1_up2 <= period + shift_period2;
748  Td2_up2 <= Tdelay_2 + shift_1ab + shift_1ab;
749  Td4_up2 <= Tdelay_4 + shift_1ab + shift_1ab;
750  Td6_up2 <= Tdelay_6 + shift_1ab + shift_1ab;
751  Ton3_up2 <= Ton_3 + shift_1a + shift_1a;
752  Ton5_up2 <= Ton_5 + shift_1ab + shift_1ab;
753  Ton1_up2 <= Ton_1 + shift_1ab + shift_1ab;
754  Ton4_up2 <= Ton_4 + shift_2a + shift_2a;
755  Ton6_up2 <= Ton_6 + shift_2ab + shift_2ab;
756  Ton2_up2 <= Ton_2 + shift_2ab + shift_2ab;
757
758  period1_up3 <= period + shift_period3;
759  Td2_up3 <= Tdelay_2 + shift_1ab + shift_1ab + shift_1ab;
760  Td4_up3 <= Tdelay_4 + shift_1ab + shift_1ab + shift_1ab;
761  Td6_up3 <= Tdelay_6 + shift_1ab + shift_1ab + shift_1ab;
762  Ton3_up3 <= Ton_3 + shift_1a + shift_1a + shift_1a;
763  Ton5_up3 <= Ton_5 + shift_1ab + shift_1ab + shift_1ab;
764  Ton1_up3 <= Ton_1 + shift_1ab + shift_1ab + shift_1ab;
765  Ton4_up3 <= Ton_4 + shift_2a + shift_2a + shift_2a;
766  Ton6_up3 <= Ton_6 + shift_2ab + shift_2ab + shift_2ab;
767  Ton2_up3 <= Ton_2 + shift_2ab + shift_2ab + shift_2ab;
768
769  period1_up4 <= period + shift_period4;
770  Td2_up4 <= Tdelay_2 + shift_1ab + shift_1ab + shift_1ab + shift_1ab;
771  Td4_up4 <= Tdelay_4 + shift_1ab + shift_1ab + shift_1ab + shift_1ab;
772  Td6_up4 <= Tdelay_6 + shift_1ab + shift_1ab + shift_1ab + shift_1ab;
773  Ton3_up4 <= Ton_3 + shift_1a + shift_1a + shift_1a + shift_1a;
```
APPENDIX C. FPGA CODE FOR CONVERTER PWMs AND SSFM

Ton5_up4 <= Ton_5 + shift_1ab + shift_1ab + shift_1ab + shift_1ab;
Ton1_up4 <= Ton_1 + shift_1ab + shift_1ab + shift_1ab + shift_1ab;
Ton4_up4 <= Ton_4 + shift_2a + shift_2a + shift_2a + shift_2a;
Ton6_up4 <= Ton_6 + shift_2ab + shift_2ab + shift_2ab + shift_2ab;
Ton2_up4 <= Ton_2 + shift_2ab + shift_2ab + shift_2ab + shift_2ab;

period1_up5 <= period + shift_period5;
Td2_up5 <= Tdelay_2 + shift_1ab + shift_1ab + shift_1ab + shift_1ab + shift_1ab;
Td4_up5 <= Tdelay_4 + shift_1ab + shift_1ab + shift_1ab + shift_1ab + shift_1ab;
Td6_up5 <= Tdelay_6 + shift_1ab + shift_1ab + shift_1ab + shift_1ab + shift_1ab;
Ton3_up5 <= Ton_3 + shift_1a + shift_1a + shift_1a + shift_1a + shift_1a;
Ton5_up5 <= Ton_5 + shift_1ab + shift_1ab + shift_1ab + shift_1ab + shift_1ab;
Ton1_up5 <= Ton_1 + shift_1ab + shift_1ab + shift_1ab + shift_1ab + shift_1ab;
Ton4_up5 <= Ton_4 + shift_2a + shift_2a + shift_2a + shift_2a + shift_2a;
Ton6_up5 <= Ton_6 + shift_2ab + shift_2ab + shift_2ab + shift_2ab + shift_2ab;
Ton2_up5 <= Ton_2 + shift_2ab + shift_2ab + shift_2ab + shift_2ab + shift_2ab;

period1_up6 <= period + shift_period6;
Td2_up6 <= Tdelay_2 + shift_1ab + shift_1ab + shift_1ab + shift_1ab + shift_1ab;
Td4_up6 <= Tdelay_4 + shift_1ab + shift_1ab + shift_1ab + shift_1ab + shift_1ab;
Td6_up6 <= Tdelay_6 + shift_1ab + shift_1ab + shift_1ab + shift_1ab + shift_1ab;
Ton3_up6 <= Ton_3 + shift_1a + shift_1a + shift_1a + shift_1a + shift_1a;
Ton5_up6 <= Ton_5 + shift_1ab + shift_1ab + shift_1ab + shift_1ab + shift_1ab;
Ton1_up6 <= Ton_1 + shift_1ab + shift_1ab + shift_1ab + shift_1ab + shift_1ab;
Ton4_up6 <= Ton_4 + shift_2a + shift_2a + shift_2a + shift_2a + shift_2a;
Ton6_up6 <= Ton_6 + shift_2ab + shift_2ab + shift_2ab + shift_2ab + shift_2ab;
Ton2_up6 <= Ton_2 + shift_2ab + shift_2ab + shift_2ab + shift_2ab + shift_2ab;

period_dwn_37 <= period - shift_period_37;
Td2_dwn1_37 <= Tdelay_2 - shift_1ab_37;
Td4_dwn1_37 <= Tdelay_4 - shift_1ab_37;
Td6_dwn1_37 <= Tdelay_6 - shift_1ab_37;
APPENDIX C. FPGA CODE FOR CONVERTER PWMS AND SSFM

\begin{verbatim}
818 Ton3_dwn1_37 <= Ton_3 - shift_a_37;
819 Ton5_dwn1_37 <= Ton_5 - shift_1ab_37;
820 Ton1_dwn1_37 <= Ton_1 - shift_1ab_37;
821 Ton4_dwn1_37 <= Ton_4 - shift_a_37;
822 Ton6_dwn1_37 <= Ton_6 - shift_2ab_37;
823 Ton2_dwn1_37 <= Ton_2 - shift_2ab_37;
824 period_dwn_47 <= period - shift_period_47;
825 Td2_dwn1_47 <= Tdelay_2 - shift_1ab_47;
826 Td4_dwn1_47 <= Tdelay_4 - shift_1ab_47;
827 Td6_dwn1_47 <= Tdelay_6 - shift_1ab_47;
828 Ton3_dwn1_47 <= Ton_3 - shift_a_47;
829 Ton5_dwn1_47 <= Ton_5 - shift_1ab_47;
830 Ton1_dwn1_47 <= Ton_1 - shift_1ab_47;
831 Ton4_dwn1_47 <= Ton_4 - shift_a_47;
832 Ton6_dwn1_47 <= Ton_6 - shift_2ab_47;
833 Ton2_dwn1_47 <= Ton_2 - shift_2ab_47;
834 period_dwn_51 <= period - shift_period_51;
835 Td2_dwn1_51 <= Tdelay_2 - shift_1ab_51;
836 Td4_dwn1_51 <= Tdelay_4 - shift_1ab_51;
837 Td6_dwn1_51 <= Tdelay_6 - shift_1ab_51;
838 Ton3_dwn1_51 <= Ton_3 - shift_a_51;
839 Ton5_dwn1_51 <= Ton_5 - shift_1ab_51;
840 Ton1_dwn1_51 <= Ton_1 - shift_1ab_51;
841 Ton4_dwn1_51 <= Ton_4 - shift_a_51;
842 Ton6_dwn1_51 <= Ton_6 - shift_2ab_51;
843 Ton2_dwn1_51 <= Ton_2 - shift_2ab_51;
844 period_dwn_53 <= period - shift_period_53;
845 Td2_dwn1_53 <= Tdelay_2 - shift_1ab_53;
846 Td4_dwn1_53 <= Tdelay_4 - shift_1ab_53;
847 Td6_dwn1_53 <= Tdelay_6 - shift_1ab_53;
848 Ton3_dwn1_53 <= Ton_3 - shift_a_53;
849 Ton5_dwn1_53 <= Ton_5 - shift_1ab_53;
850 Ton1_dwn1_53 <= Ton_1 - shift_1ab_53;
851 Ton4_dwn1_53 <= Ton_4 - shift_a_53;
852 Ton6_dwn1_53 <= Ton_6 - shift_2ab_53;
853 Ton2_dwn1_53 <= Ton_2 - shift_2ab_53;
854 period1_dwn1 <= period - shift_period1;
855 Td2_dwn1 <= Tdelay_2 - shift_1ab;
\end{verbatim}
APPENDIX C. FPGA CODE FOR CONVERTER PWMS AND SSFM

\[ Td4\_dwn1 <= Tdelay\_4 - shift\_1ab; \]
\[ Td6\_dwn1 <= Tdelay\_6 - shift\_1ab; \]
\[ Ton3\_dwn1 <= Ton\_3 - shift\_1a; \]
\[ Ton5\_dwn1 <= Ton\_5 - shift\_1ab; \]
\[ Ton1\_dwn1 <= Ton\_1 - shift\_1ab; \]
\[ Ton4\_dwn1 <= Ton\_4 - shift\_2a; \]
\[ Ton6\_dwn1 <= Ton\_6 - shift\_2ab; \]
\[ Ton2\_dwn1 <= Ton\_2 - shift\_2ab; \]
\[ \]
\[ period1\_dwn2 <= period - shift\_period2; \]
\[ Td2\_dwn2 <= Tdelay\_2 - shift\_1ab - shift\_1ab; \]
\[ Td4\_dwn2 <= Tdelay\_4 - shift\_1ab - shift\_1ab; \]
\[ Td6\_dwn2 <= Tdelay\_6 - shift\_1ab - shift\_1ab; \]
\[ Ton3\_dwn2 <= Ton\_3 - shift\_1a - shift\_1a; \]
\[ Ton5\_dwn2 <= Ton\_5 - shift\_1ab - shift\_1ab; \]
\[ Ton1\_dwn2 <= Ton\_1 - shift\_1ab - shift\_1ab; \]
\[ Ton4\_dwn2 <= Ton\_4 - shift\_2a - shift\_2a; \]
\[ Ton6\_dwn2 <= Ton\_6 - shift\_2ab - shift\_2ab; \]
\[ Ton2\_dwn2 <= Ton\_2 - shift\_2ab - shift\_2ab; \]
\[ \]
\[ period1\_dwn3 <= period - shift\_period3; \]
\[ Td2\_dwn3 <= Tdelay\_2 - shift\_1ab - shift\_1ab - shift\_1ab; \]
\[ Td4\_dwn3 <= Tdelay\_4 - shift\_1ab - shift\_1ab - shift\_1ab; \]
\[ Td6\_dwn3 <= Tdelay\_6 - shift\_1ab - shift\_1ab - shift\_1ab; \]
\[ Ton3\_dwn3 <= Ton\_3 - shift\_1a - shift\_1a - shift\_1a; \]
\[ Ton5\_dwn3 <= Ton\_5 - shift\_1ab - shift\_1ab - shift\_1ab; \]
\[ Ton1\_dwn3 <= Ton\_1 - shift\_1ab - shift\_1ab - shift\_1ab; \]
\[ Ton4\_dwn3 <= Ton\_4 - shift\_2a - shift\_2a - shift\_2a; \]
\[ Ton6\_dwn3 <= Ton\_6 - shift\_2ab - shift\_2ab - shift\_2ab; \]
\[ Ton2\_dwn3 <= Ton\_2 - shift\_2ab - shift\_2ab - shift\_2ab; \]
\[ \]
\[ period1\_dwn4 <= period - shift\_period4; \]
\[ Td2\_dwn4 <= Tdelay\_2 - shift\_1ab - shift\_1ab - shift\_1ab - shift\_1ab; \]
\[ Td4\_dwn4 <= Tdelay\_4 - shift\_1ab - shift\_1ab - shift\_1ab - shift\_1ab; \]
\[ Td6\_dwn4 <= Tdelay\_6 - shift\_1ab - shift\_1ab - shift\_1ab - shift\_1ab; \]
\[ Ton3\_dwn4 <= Ton\_3 - shift\_1a - shift\_1a - shift\_1a - shift\_1a; \]
\[ Ton5\_dwn4 <= Ton\_5 - shift\_1ab - shift\_1ab - shift\_1ab - shift\_1ab; \]
\[ Ton1\_dwn4 <= Ton\_1 - shift\_1ab - shift\_1ab - shift\_1ab - shift\_1ab; \]
\[ Ton4\_dwn4 <= Ton\_4 - shift\_2a - shift\_2a - shift\_2a - shift\_2a; \]
\[ Ton6\_dwn4 <= Ton\_6 - shift\_2ab - shift\_2ab - shift\_2ab - shift\_2ab; \]
\[ Ton2\_dwn4 <= Ton\_2 - shift\_2ab - shift\_2ab - shift\_2ab - shift\_2ab; \]
period1_dwn5 <= period - shift_period5;
Td2_dwn5 <= Tdelay_2 - shift_1ab - shift_1ab - shift_1ab - shift_1ab;
Td4_dwn5 <= Tdelay_4 - shift_1ab - shift_1ab - shift_1ab - shift_1ab;
Td6_dwn5 <= Tdelay_6 - shift_1ab - shift_1ab - shift_1ab - shift_1ab;
Ton3_dwn5 <= Ton_3 - shift_1a - shift_1a - shift_1a;
Ton5_dwn5 <= Ton_5 - shift_1ab - shift_1ab - shift_1ab - shift_1ab;
Ton1_dwn5 <= Ton_1 - shift_1a - shift_1a - shift_1a - shift_1a;
Ton4_dwn5 <= Ton_4 - shift_2a - shift_2a - shift_2a - shift_2a;
Ton6_dwn5 <= Ton_6 - shift_2ab - shift_2ab - shift_2ab - shift_2ab;
Ton2_dwn5 <= Ton_2 - shift_2ab - shift_2ab - shift_2ab - shift_2ab;

period1_dwn6 <= period - shift_period6;
Td2_dwn6 <= Tdelay_2 - shift_1ab - shift_1ab - shift_1ab - shift_1ab - shift_1ab - shift_1ab;
Td4_dwn6 <= Tdelay_4 - shift_1ab - shift_1ab - shift_1ab - shift_1ab - shift_1ab - shift_1ab;
Td6_dwn6 <= Tdelay_6 - shift_1ab - shift_1ab - shift_1ab - shift_1ab - shift_1ab - shift_1ab;
Ton3_dwn6 <= Ton_3 - shift_1a - shift_1a - shift_1a - shift_1a - shift_1a - shift_1a;
Ton5_dwn6 <= Ton_5 - shift_1ab - shift_1ab - shift_1ab - shift_1ab - shift_1ab - shift_1ab;
Ton1_dwn6 <= Ton_1 - shift_1a - shift_1a - shift_1a - shift_1a - shift_1a - shift_1a;
Ton4_dwn6 <= Ton_4 - shift_2a - shift_2a - shift_2a - shift_2a - shift_2a - shift_2a;
Ton6_dwn6 <= Ton_6 - shift_2ab - shift_2ab - shift_2ab - shift_2ab - shift_2ab - shift_2ab;
Ton2_dwn6 <= Ton_2 - shift_2ab - shift_2ab - shift_2ab - shift_2ab - shift_2ab - shift_2ab;
end
end

// overlap signal
assign GPIO[15] = (PWM1&PWM4) || (PWM1&PWM6) || (PWM2&PWM3) || (PWM2&PWM5) || (PWM3&PWM4) || (PWM3&PWM6) || (PWM4&PWM5) || (PWM5&PWM6);
wire pre_update;
assign pre_update = count_flag;
wire update;
assign update = ready;
wire count_flag;
wire ready;

sixpwm pwm_all (
  .clk (CLK_200),
  .reset (KEY_reset),
  .enable (pwm_enable),
  .pre_update (pre_update),
  .update (update),
  .period (period_new),
  .pwm1_delay (Tdelay_1_new),
  .pwm1_ontime (Ton_1_new),
  .pwm2_delay (Tdelay_2_new),
  .pwm2_ontime (Ton_2_new),
  .pwm3_delay (Tdelay_3_new),
  .pwm3_ontime (Ton_3_new),
  .pwm4_delay (Tdelay_4_new),
  .pwm4_ontime (Ton_4_new),
  .pwm5_delay (Tdelay_5_new),
  .pwm5_ontime (Ton_5_new),
  .pwm6_delay (Tdelay_6_new),
  .pwm6_ontime (Ton_6_new),
  .PWM1 (PWM1),
  .PWM2 (PWM2),
  .PWM3 (PWM3),
  .PWM4 (PWM4),
  .PWM5 (PWM5),
  .PWM6 (PWM6),
  .count_flag (count_flag),
  .ready (ready)
);

(* preserve *) reg [3:0] counter_regu;
(* preserve *) reg clockout_regu = 0;
(* preserve *) reg clockout_regu_check = 0;

wire check_vo_inc;
assign check_vo_inc = vo_meas > vo_set;
wire check_vo_dec;
assign check_vo_dec = vo_meas < vo_set;
wire check_min_inc;
wire check_min_dec;
wire check_1_inc;
wire check_1_dec;
wire check_2_inc;
wire check_2_dec;
wire check_3_inc;
wire check_3_dec;
wire check_4_inc;
wire check_4_dec;
wire check_5_inc;
wire check_5_dec;
wire check_6_inc;
wire check_6_dec;
wire check_4d_inc;
wire check_4d_dec;
wire check_6d_inc;
wire check_6d_dec;

//***** constant period ***//
wire signed [15:0] T_reg1;
wire signed [15:0] T_reg2;
reg [15:0] T_reg_abs;
assign T_reg1 = Ton_2_new + Tdelay_2_new - period_new;
assign T_reg2 = Ton_1_new - Tdelay_2_new;

always @(posedge ready) begin
  if (Ton_2_new > period_new/2) begin
    T_reg_abs = Ton_2_new - (period_new/2);
    TEST1 <= 1;
  end 
else begin
  T_reg_abs = (period_new/2) - Ton_2_new;
  TEST1 <= 2;
end 
end 

assign check_min_inc = (Ton_3_new - 2 > 1) && (Ton_4_new - 2 > 1)
&& (Ton_5_new - 2 > 1) && (Ton_6_new - 2 > 1);
assign check_min_dec = (Ton_1_new - 1 > 1) && (Ton_2_new - 1 > 1)
&\& (\text{Ton}_3\text{\_new} + 2 < \text{Ton}_1\text{\_new}) &\& (\text{Ton}_4\text{\_new} + 2 < \text{Ton}_2\text{\_new})
&\& (\text{Ton}_5\text{\_new} + 2 < \text{Ton}_1\text{\_new}) &\& (\text{Ton}_6\text{\_new} + 2 < \text{Ton}_2\text{\_new});

assign \text{check}_1\text{\_inc} = (\text{Ton}_1\text{\_new} + 1 < \text{period\_new});
assign \text{check}_1\text{\_dec} = 1;///<(\text{Ton}_1\text{\_new} - 1 >= \text{period\_new}/2);
assign \text{check}_2\text{\_inc} = (\text{Ton}_2\text{\_new} + 1 < \text{period\_new});
assign \text{check}_2\text{\_dec} = 1;///<(\text{Ton}_2\text{\_new} - 1 >= \text{period\_new}/2);
assign \text{check}_3\text{\_inc} = (\text{Ton}_3\text{\_new} - 1 < \text{Ton}_5\text{\_new} - 1 + \text{Tdelay}_5\text{\_new} + 1 - (\text{Tdelay}_3\text{\_new} + 1));
assign \text{check}_3\text{\_dec} = (\text{Ton}_3\text{\_new} + 1 < \text{Ton}_5\text{\_new} + \text{Tdelay}_5\text{\_new} - 1 - (\text{Tdelay}_3\text{\_new}));
assign \text{check}_4\text{\_inc} = (\text{Ton}_4\text{\_new} - 1 < \text{Ton}_6\text{\_new} - 1 + \text{Tdelay}_6\text{\_new} + 1 - (\text{Tdelay}_4\text{\_new} + 1));
assign \text{check}_4\text{\_dec} = (\text{Ton}_4\text{\_new} + 1 < \text{Ton}_6\text{\_new} + 1 + \text{Tdelay}_6\text{\_new} - 1 - (\text{Tdelay}_4\text{\_new} - 1));
assign \text{check}_5\text{\_inc} = (\text{Ton}_5\text{\_new} - 1 < \text{Tdelay}_2\text{\_new} - (\text{Tdelay}_5\text{\_new} + 1));
assign \text{check}_5\text{\_dec} = (\text{Ton}_5\text{\_new} + 1 < \text{Tdelay}_2\text{\_new} - (\text{Tdelay}_5\text{\_new} - 1));
assign \text{check}_6\text{\_inc} = (\text{Ton}_6\text{\_new} - 1 < \text{period\_new} - (\text{Tdelay}_6\text{\_new} + 1));
assign \text{check}_6\text{\_dec} = (\text{Ton}_6\text{\_new} + 1 < \text{period\_new} - (\text{Tdelay}_6\text{\_new} - 1));
assign \text{check}_4\text{\_d}\text{\_inc} = (\text{Tdelay}_4\text{\_new} + 1 >= \text{Ton}_1\text{\_new} + 2)
&\& (\text{Tdelay}_4\text{\_new} + 1 < \text{Ton}_6\text{\_new} - 1 + \text{Tdelay}_6\text{\_new} + 1);
assign \text{check}_4\text{\_d}\text{\_dec} = (\text{Tdelay}_4\text{\_new} - 1 >= \text{Ton}_1\text{\_new} - 2)
&\& (\text{Tdelay}_4\text{\_new} - 1 < \text{Ton}_6\text{\_new} + 1 + \text{Tdelay}_6\text{\_new} - 1);
assign \text{check}_6\text{\_d}\text{\_inc} = (\text{Tdelay}_6\text{\_new} + 1 >= \text{Ton}_1\text{\_new} + 2)
&\& (\text{Tdelay}_6\text{\_new} + 1 < \text{period\_new});
assign \text{check}_6\text{\_d}\text{\_dec} = (\text{Tdelay}_6\text{\_new} - 1 >= \text{Ton}_1\text{\_new} - 2)
&\& (\text{Tdelay}_6\text{\_new} - 1 < \text{period\_new});
assign \text{check}_3\text{\_d}\text{\_inc} = (\text{Tdelay}_3\text{\_new} + 1 >= \text{T\_reg\_abs} + 2)
&\& (\text{Tdelay}_3\text{\_new} + 1 < \text{Tdelay}_2\text{\_new});
assign \text{check}_3\text{\_d}\text{\_dec} = (\text{Tdelay}_3\text{\_new} >= \text{T\_reg\_abs} + 2);
assign \text{check}_5\text{\_d}\text{\_inc} = (\text{Tdelay}_5\text{\_new} + 1 >= \text{T\_reg\_abs} + 2)
&\& (\text{Tdelay}_5\text{\_new} + 1 < \text{Tdelay}_2\text{\_new});
assign \text{check}_5\text{\_d}\text{\_dec} = (\text{Tdelay}_5\text{\_new} >= \text{T\_reg\_abs} + 2)
&& (Tdelay_5_new < Tdelay_2_new + 1);

wire check_shift_max;
wire check_shift_min;
assign check_shift_max = shift_reg + 1 < 510;

wire checkTon_SS_reg_inc;
wire checkTon_SS_reg_dec;
assign checkTon_SS_reg_dec = (Ton_3_new - 1 - 106 >= 4) && (Ton_4_new - 1 - 106 >= 4) && (Tdelay_3_new - 1 - 106 >= 4);
assign checkTon_SS_reg_inc = (period_new + 212 <= 32000);

wire check_inc /* synthesis preserve */;
assign check_inc = check_min_inc && check_1_inc && check_2_inc && check_3_inc && check_4_inc && check_5_inc && check_6_inc && check_4d_inc && check_6d_inc && check_3d_inc && check_5d_inc && check_vo_inc && check_shift_max && checkTon_SS_reg_inc && checkTon_SS_reg_dec;

wire check_dec /* synthesis preserve */;
assign check_dec = check_min_dec && check_1_dec && check_2_dec && check_3_dec && check_4_dec && check_5_dec && check_6_dec && check_4d_dec && check_6d_dec && check_3d_dec && check_5d_dec && check_vo_dec && checkTon_SS_reg_inc && checkTon_SS_reg_dec;

always @(posedge clockout_regu) begin
if (~reg_status) begin
TEST <= 7;
shift_reg <= -0;
end
else begin
if (check_inc) begin
shift_reg <= shift_reg + 1;
TEST <= 4;
end
else if (check_dec) begin
shift_reg <= shift_reg - 1;
TEST <= 3;
end
end
else begin
TEST <= 5;
end
end
end

always @(posedge ready)
begin
if (counter_regu == 4'h9) begin
counter_regu <= 4'h0;
clockout_regu <= ~clockout_regu;
end
else begin
counter_regu <= counter_regu + 1;
end
end

always @(posedge ready)
begin
if (count_vf <= 1)
begin
period_new <= period; // + 2*shift_reg;
Tdelay_1_new <= Tdelay_1;
Tdelay_3_new <= Tdelay_3 + 1*shift_reg; //0*shift_reg;
Tdelay_5_new <= Tdelay_5 + 1*shift_reg; //0*shift_reg;
Tdelay_2_new <= Tdelay_2;
Tdelay_4_new <= Tdelay_4 + 1*shift_reg;
Tdelay_6_new <= Tdelay_6 + 1*shift_reg;
Ton_4_new <= Ton_4 - 1*shift_reg; // + 0*shift_reg;
Ton_6_new <= Ton_6 - 1*shift_reg; // + 2*shift_reg;
Ton_2_new <= Ton_2 + 1*shift_reg; // + 2*shift_reg;
Ton_3_new <= Ton_3 - 1*shift_reg; //0*shift_reg;
Ton_5_new <= Ton_5 - 1*shift_reg; // + 1*shift_reg;
Ton_1_new <= Ton_1 + 1*shift_reg; // + 2*shift_reg;
count_vf <= (count_vf + 1'b1);
end
else if (count_vf >= 150)
begin

end
else begin

end
APPENDIX C. FPGA CODE FOR CONVERTER PWMS AND SSFM

1154  count_vf <= 12'b1;
1155  period_new <= period; // + 2*shift_reg;
1156  Tdelay_1_new <= Tdelay_1;
1157  Tdelay_3_new <= Tdelay_3 + 1*shift_reg; //0*shift_reg;
1158  Tdelay_5_new <= Tdelay_5 + 1*shift_reg; //0*shift_reg;
1159  Tdelay_2_new <= Tdelay_2;
1160  Tdelay_4_new <= Tdelay_4 + 1*shift_reg;
1161  Tdelay_6_new <= Tdelay_6 + 1*shift_reg;
1162  Ton_4_new <= Ton_4 - 1*shift_reg; // + 0*shift_reg;
1163  Ton_6_new <= Ton_6 - 1*shift_reg; // + 2*shift_reg;
1164  Ton_2_new <= Ton_2 + 1*shift_reg; // + 2*shift_reg;
1165  Ton_3_new <= Ton_3 - 1*shift_reg; // + 0*shift_reg;
1166  Ton_5_new <= Ton_5 - 1*shift_reg; // + 1*shift_reg;
1167  Ton_1_new <= Ton_1 + 1*shift_reg; // + 2*shift_reg;
1168  end
1169
1170  else if (count_vf == 5)
1171    begin
1172    period_new <= period1_up1; // + 2*shift_reg;
1173    Tdelay_1_new <= Tdelay_1;
1174    Tdelay_3_new <= Tdelay_3 + 1*shift_reg; //0*shift_reg;
1175    Tdelay_5_new <= Tdelay_5 + 1*shift_reg; //0*shift_reg;
1176    Tdelay_2_new <= Td2_up1;
1177    Tdelay_4_new <= Td4_up1 + 1*shift_reg;
1178    Tdelay_6_new <= Td6_up1 + 1*shift_reg;
1179    Ton_4_new <= Ton4_up1 - 1*shift_reg; // + 0*shift_reg;
1180    Ton_6_new <= Ton6_up1 - 1*shift_reg; // + 2*shift_reg;
1181    Ton_2_new <= Ton2_up1 + 1*shift_reg; // + 2*shift_reg;
1182    Ton_3_new <= Ton3_up1 - 1*shift_reg; // + 0*shift_reg;
1183    Ton_5_new <= Ton5_up1 - 1*shift_reg; // + 1*shift_reg;
1184    Ton_1_new <= Ton1_up1 + 1*shift_reg; // + 2*shift_reg;
1185    end
1186
1187  else if (count_vf == 10)
1188    begin
1189    period_new <= period1_up2; // + 2*shift_reg;
1190    Tdelay_1_new <= Tdelay_1;
1191    Tdelay_3_new <= Tdelay_3 + 1*shift_reg; //0*shift_reg;
1192    Tdelay_5_new <= Tdelay_5 + 1*shift_reg; //0*shift_reg;
1193    end
APPENDIX C. FPGA CODE FOR CONVERTER PWMS AND SSFM

1196  Tdelay_2_new <= Td2_up2;
1197  Tdelay_4_new <= Td4_up2 + 1*shift_reg;
1198  Tdelay_6_new <= Td6_up2 + 1*shift_reg;
1199  Ton_4_new <= Ton4_up2 - 1*shift_reg;  // + 0*shift_reg;
1200  Ton_6_new <= Ton6_up2 - 1*shift_reg;  // + 2*shift_reg;
1201  Ton_2_new <= Ton2_up2 + 1*shift_reg;  // + 2*shift_reg;
1202  Ton_3_new <= Ton3_up2 - 1*shift_reg;  // + 0*shift_reg;
1203  Ton_5_new <= Ton5_up2 - 1*shift_reg;  // + 1*shift_reg;
1204  Ton_1_new <= Ton1_up2 + 1*shift_reg;  // + 2*shift_reg;
1205
1206  count_vf <= (count_vf + 1'b1);
1207  end
1208  else if (count_vf == 15)
1209    begin
1210      period_new <= period1_up3;  // + 2*shift_reg;
1211      Tdelay_1_new <= Tdelay_1;
1212      Tdelay_3_new <= Tdelay_3  + 1*shift_reg;  //0*shift_reg;
1213      Tdelay_5_new <= Tdelay_5  + 1*shift_reg;  //0*shift_reg;
1214      Tdelay_2_new <= Td2_up3;
1215      Tdelay_4_new <= Td4_up3 + 1*shift_reg;
1216      Tdelay_6_new <= Td6_up3 + 1*shift_reg;
1217      Ton_4_new <= Ton4_up3 - 1*shift_reg;  // + 0*shift_reg;
1218      Ton_6_new <= Ton6_up3 - 1*shift_reg;  // + 2*shift_reg;
1219      Ton_2_new <= Ton2_up3 + 1*shift_reg;  // + 2*shift_reg;
1220      Ton_3_new <= Ton3_up3 - 1*shift_reg;  // + 0*shift_reg;
1221      Ton_5_new <= Ton5_up3 - 1*shift_reg;  // + 1*shift_reg;
1222      Ton_1_new <= Ton1_up3 + 1*shift_reg;  // + 2*shift_reg;
1223
1224  count_vf <= (count_vf + 1'b1);
1225  end
1226  else if (count_vf == 20)
1227    begin
1228      period_new <= period1_up4;  // + 2*shift_reg;
1229      Tdelay_1_new <= Tdelay_1;
1230      Tdelay_3_new <= Tdelay_3  + 1*shift_reg;  //0*shift_reg;
1231      Tdelay_5_new <= Tdelay_5  + 1*shift_reg;  //0*shift_reg;
1232      Tdelay_2_new <= Td2_up4;
1233      Tdelay_4_new <= Td4_up4 + 1*shift_reg;
1234      Tdelay_6_new <= Td6_up4 + 1*shift_reg;
1235      Ton_4_new <= Ton4_up4 - 1*shift_reg;  // + 0*shift_reg;
APPENDIX C. FPGA CODE FOR CONVERTER PWMS AND SSFM

Ton_6_new <= Ton6_up4 - 1*shift_reg; // + 2*shift_reg;
Ton_2_new <= Ton2_up4 + 1*shift_reg; // + 2*shift_reg;
Ton_3_new <= Ton3_up4 - 1*shift_reg; // + 0*shift_reg;
Ton_5_new <= Ton5_up4 - 1*shift_reg; // + 1*shift_reg;
Ton_1_new <= Ton1_up4 + 1*shift_reg; // + 2*shift_reg;

count_vf <= (count_vf + 1'b1);
end
else if (count_vf ==25)
begin
period_new <= period1_up5;// + 2*shift_reg;
Tdelay_1_new <= Tdelay_1;
Tdelay_3_new <= Tdelay_3 + 1*shift_reg; //0*shift_reg;
Tdelay_5_new <= Tdelay_5 + 1*shift_reg; //0*shift_reg;
Tdelay_2_new <= Td2_up5;
Tdelay_4_new <= Td4_up5 + 1*shift_reg;
Tdelay_6_new <= Td6_up5 + 1*shift_reg;
Ton_4_new <= Ton4_up5 - 1*shift_reg; // + 0*shift_reg;
Ton_6_new <= Ton6_up5 - 1*shift_reg; // + 2*shift_reg;
Ton_2_new <= Ton2_up5 + 1*shift_reg; // + 2*shift_reg;
Ton_3_new <= Ton3_up5 - 1*shift_reg; // + 0*shift_reg;
Ton_5_new <= Ton5_up5 - 1*shift_reg; // + 1*shift_reg;
Ton_1_new <= Ton1_up5 + 1*shift_reg; // + 2*shift_reg;

count_vf <= (count_vf + 1'b1);
end
else if (count_vf ==30)
begin
period_new <= period1_up6;// + 2*shift_reg;
Tdelay_1_new <= Tdelay_1;
Tdelay_3_new <= Tdelay_3 + 1*shift_reg; //0*shift_reg;
Tdelay_5_new <= Tdelay_5 + 1*shift_reg; //0*shift_reg;
Tdelay_2_new <= Td2_up6;
Tdelay_4_new <= Td4_up6 + 1*shift_reg;
Tdelay_6_new <= Td6_up6 + 1*shift_reg;
Ton_4_new <= Ton4_up6 - 1*shift_reg; // + 0*shift_reg;
Ton_6_new <= Ton6_up6 - 1*shift_reg; // + 2*shift_reg;
Ton_2_new <= Ton2_up6 + 1*shift_reg; // + 2*shift_reg;
Ton_3_new <= Ton3_up6 - 1*shift_reg; // + 0*shift_reg;
Ton_5_new <= Ton5_up6 - 1*shift_reg; // + 1*shift_reg;
Ton_1_new <= Ton1_up6 + 1*shift_reg; // + 2*shift_reg;
count_vf <= (count_vf + 1'b1);
end
else if (count_vf == 50)
begin
period_new <= period1_up5; // + 2*shift_reg;
Tdelay_1_new <= Tdelay_1;
Tdelay_3_new <= Tdelay_3 + 1*shift_reg; //0*shift_reg;
Tdelay_5_new <= Tdelay_5 + 1*shift_reg; //0*shift_reg;
Tdelay_2_new <= Td2_up5;
Tdelay_4_new <= Td4_up5 + 1*shift_reg;
Tdelay_6_new <= Td6_up5 + 1*shift_reg;
Ton_4_new <= Ton4_up5 - 1*shift_reg; //+ 0*shift_reg;
Ton_6_new <= Ton6_up5 - 1*shift_reg; // + 2*shift_reg;
Ton_2_new <= Ton2_up5 + 1*shift_reg; // + 2*shift_reg;
Ton_3_new <= Ton3_up5 - 1*shift_reg; //+ 0*shift_reg;
Ton_5_new <= Ton5_up5 - 1*shift_reg; // + 1*shift_reg;
Ton_1_new <= Ton1_up5 + 1*shift_reg; // + 2*shift_reg;
end
else if (count_vf == 55)
begin
period_new <= period1_up4; // + 2*shift_reg;
Tdelay_1_new <= Tdelay_1;
Tdelay_3_new <= Tdelay_3 + 1*shift_reg; //0*shift_reg;
Tdelay_5_new <= Tdelay_5 + 1*shift_reg; //0*shift_reg;
Tdelay_2_new <= Td2_up4;
Tdelay_4_new <= Td4_up4 + 1*shift_reg;
Tdelay_6_new <= Td6_up4 + 1*shift_reg;
Ton_4_new <= Ton4_up4 - 1*shift_reg; //+ 0*shift_reg;
Ton_6_new <= Ton6_up4 - 1*shift_reg; // + 2*shift_reg;
Ton_2_new <= Ton2_up4 + 1*shift_reg; // + 2*shift_reg;
Ton_3_new <= Ton3_up4 - 1*shift_reg; //+ 0*shift_reg;
Ton_5_new <= Ton5_up4 - 1*shift_reg; // + 1*shift_reg;
Ton_1_new <= Ton1_up4 + 1*shift_reg; // + 2*shift_reg;
count_vf <= (count_vf + 1'b1);
end
else if (count_vf == 60)
begin
period_new <= period1_up3; // + 2*shift_reg;
Tdelay_1_new <= Tdelay_1;
Tdelay_3_new <= Tdelay_3 + 1*shift_reg; //0*shift_reg;
Tdelay_5_new <= Tdelay_5 + 1*shift_reg; //0*shift_reg;
Tdelay_2_new <= Td2_up3;
Tdelay_4_new <= Td4_up3 + 1*shift_reg;
Tdelay_6_new <= Td6_up3 + 1*shift_reg;
Ton_4_new <= Ton4_up3 - 1*shift_reg; //+ 0*shift_reg;
Ton_6_new <= Ton6_up3 - 1*shift_reg; //+ 2*shift_reg;
Ton_2_new <= Ton2_up3 + 1*shift_reg; //+ 2*shift_reg;
Ton_3_new <= Ton3_up3 - 1*shift_reg; //+ 0*shift_reg;
Ton_5_new <= Ton5_up3 - 1*shift_reg; //+ 1*shift_reg;
Ton_1_new <= Ton1_up3 + 1*shift_reg; //+ 2*shift_reg;
count_vf <= (count_vf + 1'b1);
end
else if (count_vf == 65)
begin
period_new <= period1_up2; // + 2*shift_reg;
Tdelay_1_new <= Tdelay_1;
Tdelay_3_new <= Tdelay_3 + 1*shift_reg; //0*shift_reg;
Tdelay_5_new <= Tdelay_5 + 1*shift_reg; //0*shift_reg;
Tdelay_2_new <= Td2_up2;
Tdelay_4_new <= Td4_up2 + 1*shift_reg;
Tdelay_6_new <= Td6_up2 + 1*shift_reg;
Ton_4_new <= Ton4_up2 - 1*shift_reg; //+ 0*shift_reg;
Ton_6_new <= Ton6_up2 - 1*shift_reg; //+ 2*shift_reg;
Ton_2_new <= Ton2_up2 + 1*shift_reg; //+ 2*shift_reg;
Ton_3_new <= Ton3_up2 - 1*shift_reg; //+ 0*shift_reg;
Ton_5_new <= Ton5_up2 - 1*shift_reg; //+ 1*shift_reg;
Ton_1_new <= Ton1_up2 + 1*shift_reg; //+ 2*shift_reg;
count_vf <= (count_vf + 1'b1);
end
else if (count_vf == 70)
begin
period_new <= period1_up1; // + 2*shift_reg;
Tdelay_1_new <= Tdelay_1;
Tdelay_3_new <= Tdelay_3 + 1*shift_reg; //0*shift_reg;
Tdelay_5_new <= Tdelay_5 + 1*shift_reg; //0*shift_reg;
Tdelay_2_new <= Td2_up1;
Tdelay_4_new <= Td4_up1 + 1*shift_reg;
Tdelay_6_new <= Td6_up1 + 1*shift_reg;
Ton_4_new <= Ton4_up1 - 1*shift_reg; // + 0*shift_reg;
Ton_6_new <= Ton6_up1 - 1*shift_reg; // + 2*shift_reg;
Ton_2_new <= Ton2_up1 + 1*shift_reg; // + 2*shift_reg;
Ton_3_new <= Ton3_up1 - 1*shift_reg; // + 0*shift_reg;
Ton_5_new <= Ton5_up1 - 1*shift_reg; // + 1*shift_reg;
Ton_1_new <= Ton1_up1 + 1*shift_reg; // + 2*shift_reg;

count_vf <= (count_vf + 1'b1);
end
else if (count_vf == 75)
begin
period_new <= period; // + 2*shift_reg;
Tdelay_1_new <= Tdelay_1;
Tdelay_3_new <= Tdelay_3 + 1*shift_reg; // 0*shift_reg;
Tdelay_5_new <= Tdelay_5 + 1*shift_reg; // 0*shift_reg;
Tdelay_2_new <= Tdelay_2;
Tdelay_4_new <= Tdelay_4 + 1*shift_reg;
Tdelay_6_new <= Tdelay_6 + 1*shift_reg;
Ton_4_new <= Ton_4 - 1*shift_reg; // + 0*shift_reg;
Ton_6_new <= Ton_6 - 1*shift_reg; // + 2*shift_reg;
Ton_2_new <= Ton_2 + 1*shift_reg; // + 2*shift_reg;
Ton_3_new <= Ton_3 - 1*shift_reg; // + 0*shift_reg;
Ton_5_new <= Ton_5 - 1*shift_reg; // + 1*shift_reg;
Ton_1_new <= Ton_1 + 1*shift_reg; // + 2*shift_reg;

count_vf <= (count_vf + 1'b1);
end
else if (count_vf == 80)
begin
period_new <= period1_dwn1; // + 2*shift_reg;
Tdelay_1_new <= Tdelay_1;
Tdelay_3_new <= Tdelay_3 + 1*shift_reg; // 0*shift_reg;
Tdelay_5_new <= Tdelay_5 + 1*shift_reg; // 0*shift_reg;
Tdelay_2_new <= Td2_dwn1;
Tdelay_4_new <= Td4_dwn1 + 1*shift_reg;
Tdelay_6_new <= Td6_dwn1 + 1*shift_reg;
Ton_4_new <= Ton_4_dwn1 - 1*shift_reg; // + 0*shift_reg;
Ton_6_new <= Ton_6_dwn1 - 1*shift_reg; // + 2*shift_reg;
Ton_2_new <= Ton_2_dwn1 + 1*shift_reg; // + 2*shift_reg;
Ton_3_new <= Ton_3_dwn1 - 1*shift_reg; // + 0*shift_reg;
APPENDIX C. FPGA CODE FOR CONVERTER PWMS AND SSFM

```
1406 Ton_5_new <= Ton5_dwn1 - 1*shift_reg; // + 1*shift_reg;
1407 Ton_1_new <= Ton1_dwn1 + 1*shift_reg; // + 2*shift_reg;
1408 count_vf <= (count_vf + 1'b1);
1409 end
else if (count_vf == 85)
begin
  period_new <= period1_dwn2; // + 2*shift_reg;
  Tdelay_1_new <= Tdelay_1;
  Tdelay_3_new <= Tdelay_3 + 1*shift_reg; //0*shift_reg;
  Tdelay_5_new <= Tdelay_5 + 1*shift_reg; //0*shift_reg;
  Tdelay_2_new <= Td2_dwn2;
  Tdelay_4_new <= Td4_dwn2 + 1*shift_reg;
  Tdelay_6_new <= Td6_dwn2 + 1*shift_reg;
  Ton_4_new <= Ton4_dwn2 - 1*shift_reg; // + 0*shift_reg;
  Ton_6_new <= Ton6_dwn2 - 1*shift_reg; // + 2*shift_reg;
  Ton_2_new <= Ton2_dwn2 + 1*shift_reg; // + 2*shift_reg;
  Ton_3_new <= Ton3_dwn2 - 1*shift_reg; // + 0*shift_reg;
  Ton_5_new <= Ton5_dwn2 - 1*shift_reg; // + 1*shift_reg;
  Ton_1_new <= Ton1_dwn2 + 1*shift_reg; // + 2*shift_reg;
  count_vf <= (count_vf + 1'b1);
end
else if (count_vf == 90)
begin
  period_new <= period1_dwn3;
  Tdelay_1_new <= Tdelay_1 + 0*shift_reg;
  Tdelay_3_new <= Tdelay_3 + 1*shift_reg; //0*shift_reg;
  Tdelay_5_new <= Tdelay_5 + 1*shift_reg; //0*shift_reg;
  Tdelay_2_new <= Td2_dwn3 + 0*shift_reg;
  Tdelay_4_new <= Td4_dwn3 + 1*shift_reg;
  Tdelay_6_new <= Td6_dwn3 + 1*shift_reg;
  Ton_4_new <= Ton4_dwn3 - 1*shift_reg; // + 0*shift_reg;
  Ton_6_new <= Ton6_dwn3 - 1*shift_reg; // + 2*shift_reg;
  Ton_2_new <= Ton2_dwn3 + 1*shift_reg; // + 2*shift_reg;
  Ton_3_new <= Ton3_dwn3 - 1*shift_reg; // + 0*shift_reg;
  Ton_5_new <= Ton5_dwn3 - 1*shift_reg; // + 1*shift_reg;
  Ton_1_new <= Ton1_dwn3 + 1*shift_reg; // + 2*shift_reg;
  count_vf <= (count_vf + 1'b1);
end
else if (count_vf == 95)
begin
```

APPENDIX C. FPGA CODE FOR CONVERTER PWMS AND SSFM

```
1448 period_new <= period1_dwn4; // + 2*shift_reg;
1449 Tdelay_1_new <= Tdelay_1 + 0*shift_reg;
1450 Tdelay_3_new <= Tdelay_3 + 1*shift_reg; //0*shift_reg;
1451 Tdelay_5_new <= Tdelay_5 + 1*shift_reg; //0*shift_reg;
1452 Tdelay_2_new <= Td2_dwn4 + 0*shift_reg;
1453 Tdelay_4_new <= Td4_dwn4 + 1*shift_reg;
1454 Tdelay_6_new <= Td6_dwn4 + 1*shift_reg;
1455 Ton_4_new <= Ton4_dwn4 - 1*shift_reg; // + 0*shift_reg;
1456 Ton_6_new <= Ton6_dwn4 - 1*shift_reg; // + 2*shift_reg;
1457 Ton_2_new <= Ton2_dwn4 + 1*shift_reg; // + 2*shift_reg;
1458 Ton_3_new <= Ton3_dwn4 - 1*shift_reg; // + 0*shift_reg;
1459 Ton_5_new <= Ton5_dwn4 - 1*shift_reg; // + 1*shift_reg;
1460 Ton_1_new <= Ton1_dwn4 + 1*shift_reg; // + 2*shift_reg;
1461 count_vf <= (count_vf + 1'b1);
1462 end
1463 else if (count_vf == 100)
1464 begin
1465 period_new <= period1_dwn5; // + 2*shift_reg;
1466 Tdelay_1_new <= Tdelay_1 + 0*shift_reg;
1467 Tdelay_3_new <= Tdelay_3 + 1*shift_reg; //0*shift_reg;
1468 Tdelay_5_new <= Tdelay_5 + 1*shift_reg; //0*shift_reg;
1469 Tdelay_2_new <= Td2_dwn5 + 0*shift_reg;
1470 Tdelay_4_new <= Td4_dwn5 + 1*shift_reg;
1471 Tdelay_6_new <= Td6_dwn5 + 1*shift_reg;
1472 Ton_4_new <= Ton4_dwn5 - 1*shift_reg; // + 0*shift_reg;
1473 Ton_6_new <= Ton6_dwn5 - 1*shift_reg; // + 2*shift_reg;
1474 Ton_2_new <= Ton2_dwn5 + 1*shift_reg; // + 2*shift_reg;
1475 Ton_3_new <= Ton3_dwn5 - 1*shift_reg; // + 0*shift_reg;
1476 Ton_5_new <= Ton5_dwn5 - 1*shift_reg; // + 1*shift_reg;
1477 Ton_1_new <= Ton1_dwn5 + 1*shift_reg; // + 2*shift_reg;
1478 count_vf <= (count_vf + 1'b1);
1479 end
1480 else if (count_vf == 105)
1481 begin
1482 period_new <= period1_dwn6; // + 2*shift_reg;
1483 Tdelay_1_new <= Tdelay_1 + 0*shift_reg;
1484 Tdelay_3_new <= Tdelay_3 + 1*shift_reg; //0*shift_reg;
1485 Tdelay_5_new <= Tdelay_5 + 1*shift_reg; //0*shift_reg;
1486 Tdelay_2_new <= Td2_dwn6 + 0*shift_reg;
1487 Tdelay_4_new <= Td4_dwn6 + 1*shift_reg;
1488 Tdelay_6_new <= Td6_dwn6 + 1*shift_reg;
1489 Ton_4_new <= Ton4_dwn6 - 1*shift_reg; // + 0*shift_reg;
```
APPENDIX C. FPGA CODE FOR CONVERTER PWMs AND SSFM

1490 Tor_6_new <= Tor6_dwn6 - 1*shift_reg; // + 2*shift_reg;
1491 Tor_2_new <= Tor2_dwn6 + 1*shift_reg; // + 2*shift_reg;
1492 Tor_3_new <= Tor3_dwn6 - 1*shift_reg; // + 0*shift_reg;
1493 Tor_5_new <= Tor5_dwn6 - 1*shift_reg; // + 1*shift_reg;
1494 Tor_1_new <= Tor1_dwn6 + 1*shift_reg; // + 2*shift_reg;
1495 count_vf <= (count_vf + 1'b1);
1496 end
1497 else if (count_vf == 125)
1498 begin
1499 period_new <= period1_dwn5; // + 2*shift_reg;
1500 Tdelay_1_new <= Tdelay_1 + 0*shift_reg;
1501 Tdelay_3_new <= Tdelay_3 + 1*shift_reg; // 0*shift_reg;
1502 Tdelay_5_new <= Tdelay_5 + 1*shift_reg; // 0*shift_reg;
1503 Tdelay_2_new <= Td2_dwn5 + 0*shift_reg;
1504 Tdelay_4_new <= Td4_dwn5 + 1*shift_reg;
1505 Tdelay_6_new <= Td6_dwn5 + 1*shift_reg;
1506 Tor_4_new <= Tor4_dwn5 - 1*shift_reg; // + 0*shift_reg;
1507 Tor_6_new <= Tor6_dwn5 - 1*shift_reg; // + 2*shift_reg;
1508 Tor_2_new <= Tor2_dwn5 + 1*shift_reg; // + 2*shift_reg;
1509 Tor_3_new <= Tor3_dwn5 - 1*shift_reg; // + 0*shift_reg;
1510 Tor_5_new <= Tor5_dwn5 - 1*shift_reg; // + 1*shift_reg;
1511 Tor_1_new <= Tor1_dwn5 + 1*shift_reg; // + 2*shift_reg;
1512 Tor_1_new <= Tor1_dwn1_51 + 1*shift_reg; // + 2*shift_reg;
1513
1514 count_vf <= (count_vf + 1'b1);
1515 end
1516 else if (count_vf == 130)
1517 begin
1518 period_new <= period1_dwn4; // + 2*shift_reg;
1519 Tdelay_1_new <= Tdelay_1 + 0*shift_reg;
1520 Tdelay_3_new <= Tdelay_3 + 1*shift_reg; // 0*shift_reg;
1521 Tdelay_5_new <= Tdelay_5 + 1*shift_reg; // 0*shift_reg;
1522 Tdelay_2_new <= Td2_dwn4 + 0*shift_reg;
1523 Tdelay_4_new <= Td4_dwn4 + 1*shift_reg;
1524 Tdelay_6_new <= Td6_dwn4 + 1*shift_reg;
1525 Tor_4_new <= Tor4_dwn4 - 1*shift_reg; // + 0*shift_reg;
1526 Tor_6_new <= Tor6_dwn4 - 1*shift_reg; // + 2*shift_reg;
1527 Tor_2_new <= Tor2_dwn4 + 1*shift_reg; // + 2*shift_reg;
1528 Tor_3_new <= Tor3_dwn4 - 1*shift_reg; // + 0*shift_reg;
1529 Tor_5_new <= Tor5_dwn4 - 1*shift_reg; // + 1*shift_reg;
1530 Tor_1_new <= Tor1_dwn4 + 1*shift_reg; // + 2*shift_reg;
APPENDIX C. FPGA CODE FOR CONVERTER PWMS AND SSFM

```vhdl
1532
1533
1534 count_vf <= (count_vf + 1'b1);
1535 end
1536 else if (count_vf == 135)
1537 begin
1538 period_new <= period1_dwn3;
1539 Tdelay_1_new <= Tdelay_1 + 0*shift_reg;
1540 Tdelay_3_new <= Tdelay_3 + 1*shift_reg; //0*shift_reg;
1541 Tdelay_5_new <= Tdelay_5 + 1*shift_reg; //0*shift_reg;
1542 Tdelay_2_new <= Td2_dwn3 + 0*shift_reg;
1543 Tdelay_4_new <= Td4_dwn3 + 1*shift_reg;
1544 Tdelay_6_new <= Td6_dwn3 + 1*shift_reg;
1545 Ton_4_new <= Ton4_dwn3 - 1*shift_reg; // 0*shift_reg;
1546 Ton_6_new <= Ton6_dwn3 - 1*shift_reg; // 2*shift_reg;
1547 Ton_2_new <= Ton2_dwn3 + 1*shift_reg; // 2*shift_reg;
1548 Ton_3_new <= Ton3_dwn3 - 1*shift_reg; // 0*shift_reg;
1549 Ton_5_new <= Ton5_dwn3 - 1*shift_reg; // 1*shift_reg;
1550 Ton_1_new <= Ton1_dwn3 + 1*shift_reg; // 2*shift_reg;
1551
1552 count_vf <= (count_vf + 1'b1);
1553 end
1554 else if (count_vf == 140)
1555 begin
1556 period_new <= period1_dwn2; // 2*shift_reg;
1557 Tdelay_1_new <= Tdelay_1;
1558 Tdelay_3_new <= Tdelay_3 + 1*shift_reg; //0*shift_reg;
1559 Tdelay_5_new <= Tdelay_5 + 1*shift_reg; //0*shift_reg;
1560 Tdelay_2_new <= Td2_dwn2;
1561 Tdelay_4_new <= Td4_dwn2 + 1*shift_reg;
1562 Tdelay_6_new <= Td6_dwn2 + 1*shift_reg;
1563 Ton_4_new <= Ton4_dwn2 - 1*shift_reg; // 0*shift_reg;
1564 Ton_6_new <= Ton6_dwn2 - 1*shift_reg; // 2*shift_reg;
1565 Ton_2_new <= Ton2_dwn2 + 1*shift_reg; // 2*shift_reg;
1566 Ton_3_new <= Ton3_dwn2 - 1*shift_reg; // 0*shift_reg;
1567 Ton_5_new <= Ton5_dwn2 - 1*shift_reg; // 1*shift_reg;
1568 Ton_1_new <= Ton1_dwn2 + 1*shift_reg; // 2*shift_reg;
1569 count_vf <= (count_vf + 1'b1);
1570 end
1571 else if (count_vf == 145)
1572 begin
```
APPENDIX C. FPGA CODE FOR CONVERTER PWMS AND SSFM

1574 period_new <= period1_dwn1; // + 2*shift_reg;
1575 Tdelay_1_new <= Tdelay_1 + 0*shift_reg;
1576 Tdelay_3_new <= Tdelay_3 + 1*shift_reg; //0*shift_reg;
1577 Tdelay_5_new <= Tdelay_5 + 1*shift_reg; //0*shift_reg;
1578 Tdelay_2_new <= Td2_dwn1 + 0*shift_reg;
1579 Tdelay_4_new <= Td4_dwn1 + 1*shift_reg;
1580 Tdelay_6_new <= Td6_dwn1 + 1*shift_reg;
1581 Ton_4_new <= Ton4_dwn1 - 1*shift_reg; /// 0*shift_reg;
1582 Ton_6_new <= Ton6_dwn1 - 1*shift_reg; /// + 2*shift_reg;
1583 Ton_2_new <= Ton2_dwn1 + 1*shift_reg; /// + 2*shift_reg;
1584 Ton_3_new <= Ton3_dwn1 - 1*shift_reg; /// + 0*shift_reg;
1585 Ton_5_new <= Ton5_dwn1 - 1*shift_reg; /// + 1*shift_reg;
1586 Ton_1_new <= Ton1_dwn1 + 1*shift_reg; /// + 2*shift_reg;
1587 count_vf <= (count_vf + 1’b1);
1588 end
1589
1590 else
1591 begin
1592 count_vf <= (count_vf + 1’b1);
1593 end
1594 end
1595
1596 endmodule
1597