Enhancing Digital Control for DC/DC Converters

Simon Effler

Submitted for the award of Doctor of Philosophy

Department of Electronic & Computer Engineering,
University of Limerick,
Limerick, Ireland

Supervisors: Dr. Karl Rinne, Dr. Mark Halton

Submitted to the University of Limerick,
2010
Enhancing Digital Control for DC/DC Converters

Simon Effler

The continuing trend in switched-mode power supplies clearly leads to the delivery of more output power with an ever increasing number of voltages more efficiently at less cost. Increasing power management requirements, i.e. communication functionalities, contribute significantly to the need for digitally controlled power supplies.

While most of these designs still use simple PID compensators, the introduction of more advanced compensation techniques will improve the system performance considerably. However, one reason for the slow adoption of digital power converters is the lack of proper design tools for advanced digital controllers. This issue is addressed in this thesis where an automated design procedure based on Generalized Predictive Control (GPC) has been developed. Real-life design criteria are used to optimize compensators for application-specific requirements without the need for extensive control expertise.

Another issue in today’s digital power systems is the choice of system architecture. Two different concepts are widely used. For high output power demands and fast transient response, multi-phase converters are used which modulate a number of parallel power converters (phases). In this thesis, a new modulator concept is proposed which enables the use of efficiency-improving techniques without increasing the design complexity. The proposed modulator features an internal duty cycle redistribution without complex management overhead. With this modulator, phase shedding can be accommodated together with improved transient behaviour.

An alternative concept commonly used today is an intermediate bus architecture with multiple separate point-of-load converters; each controlled by its individual digital core. Digital communication signals between the converters are required to implement advanced control features, such as phase shedding or current sharing. In this thesis, new techniques are proposed which enable the use of these advanced control features without the need for digital communication signals. Firstly, a new algorithm for phase alignment and frequency synchronization between the individual converters is detailed which uses the perturbation generated by the individual power supplies on the input voltage. Secondly, a novel current distribution principle based on smart power converters is presented. It enables the distribution of the output current over the individual power converters to optimize overall efficiency. Each converter optimizes its output current based on predefined optimal currents without requiring information about the total output current.
“The important thing is not to stop questioning. Curiosity has its own reason for existing.”

“Das Wichtige ist, dass man nicht aufhört zu fragen. Neugierde hat ihren eigenen Existenzgrund.”

– Albert Einstein (1879 - 1955) –
Acknowledgements

I would like to express my gratitude to Karl Rinne as academic supervisor for the valuable guidance, feedback, discussions and the opportunity to independently pursue my own research interests.

I would like to acknowledge Mark Halton for his feedback on the manuscript and the hours spent proofreading this thesis.

I would like to thank all people who have directly or indirectly contributed to this thesis. All current and former colleagues at the Circuits and System Research Centre, University of Limerick. Special thanks to Anthony Kelly, Tilmann Krüger at the University of Mannheim, Aleksandar Prodić and the members of his research group at the University of Toronto, . . .

I like to acknowledge the support of the Irish Research Council for Science, Engineering and Technology: funded by the National Development Plan for the scholarship granted.
Declaration

I hereby declare that this thesis is my own work. References have been cited, as appropriate. It has not been submitted in whole or in part to any other university.

Simon Effler
# Contents

Abstract                                  II
Acknowledgements                          IV
Declaration                                V
Table of Contents                         VIII
List of Figures                           XII
Glossary                                  XIII

1. Introduction and Motivation            1
   1.1. Fuelling today’s digital loads ................. 1
   1.2. Thesis structure ................................ 10
   1.3. Contribution .................................... 10

2. Automated Compensator Design using Real-life Design Criteria 12
   2.1. Introduction .................................... 13
       2.1.1. Digital control of SMPS ................. 13
       2.1.2. Compensator design procedures .......... 14
   2.2. GPC Compensator Design for Linear Unconstrained Systems . 16
       2.2.1. Generalized Predictive Control – An overview ..... 16
       2.2.2. Mathematical background .................. 20
   2.3. Automated Design Process using GPC ............. 24
       2.3.1. Selection of a prediction model .......... 25
       2.3.2. Selection of a suitable performance index .... 30
       2.3.3. Automated design procedure / search for the optimal GPC parameter set ................. 30
2.4. Performance Indices based on Real-life Design Criteria . . . . 32
   2.4.1. General considerations on performance indices . . . . . . 33
   2.4.2. Settling time ........................................ 33
   2.4.3. Maximum deviation ................................... 37
   2.4.4. Over-/Undershoot .................................... 38
   2.4.5. Area ..................................................... 39
   2.4.6. Phase margin .......................................... 42
   2.4.7. Combination of different performance criteria ............. 43
   2.4.8. Discussion of suitable performance indices ............... 45

2.5. Solution to the Optimization Problem .......................... 45

2.6. Practical Design Example .................................... 46
   2.6.1. Case 1: Using the settling time ........................ 48
   2.6.2. Case 2: Using a combined index ....................... 53

2.7. Discussion and Conclusions ................................ 60

3. Automatic Multi-phase Digital Pulse Width Modulator ... 61
   3.1. Introduction .............................................. 62
   3.2. System Overview ......................................... 64
   3.3. Distribution Principle ................................... 65
      3.3.1. New distribution principle .......................... 65
      3.3.2. Phase shedding .................................... 70
      3.3.3. Alternative modulation schemes .................... 70
   3.4. Implementation .......................................... 73
   3.5. Performance assessment ................................ 75
      3.5.1. Prototype framework system ........................ 76
      3.5.2. Loop compensation .................................. 78
      3.5.3. Current sharing technique .......................... 79
      3.5.4. Performance assessment ............................. 81
   3.6. Discussion .............................................. 85

4. Scalable Digital Power Controller with Automatic Phase Alignment and Frequency Synchronization 86
   4.1. Introduction .............................................. 87
   4.2. Parallel Power Converters ................................ 88
      4.2.1. Benefits and issues ................................. 88
5. Current Distribution Scheme for Independent Power Supplies based on Efficiency Levels 106
5.1. Introduction ................................. 107
  5.1.1. Control architectures ................. 107
  5.1.2. Existing current sharing/distribution techniques ... 109
5.2. Current Distribution based on Optimal Current Levels .... 111
5.3. Theoretical Analysis .......................... 116
5.4. Implementation ............................... 122
5.5. Experimental Testing .......................... 125
5.6. Conclusions ................................. 130

6. Conclusions ................................. 131

Bibliography .................................. 134

A. Automated Compensator Design using Real-life Design Criteria 146
  A.1. Nomenclature for GPC ........................ 146
  A.2. GPC design algorithm ...................... 149

B. AM-DPWM .................................. 151
  B.1. Prototype schematic excerpts .............. 151

C. Current Distribution Scheme for Independent Power Supplies based on Efficiency Levels 157
  C.1. Efficiency measurements .................... 157
List of Figures

1.1. Distributed power architecture ........................................... 3
1.2. Intermediate bus architecture with multiple POL converters .... 3
1.3. Centralized control IC with power blocks ............................. 3
1.4. Buck converter with analogue control loop ......................... 5
1.5. Buck converter with analogue control loop and digital wrapper. 5
1.6. Buck converter with digital control loop ............................ 5
1.7. Comparison of different system concepts: Transient performance vs. efficiency ......................................................... 9

2.1. Digital control loop .......................................................... 13
2.2. Control loop incorporating a GPC derived compensator ........ 16
2.3. Basic GPC cost function .................................................... 17
2.4. Costs of an oscillating mode ............................................. 19
2.5. Abstract view of the GPC algorithm ................................... 19
2.6. GPC design process .......................................................... 25
2.7. Single phase buck converter .............................................. 26
2.8. Transfer function buck model ............................................ 27
2.9. Multi-phase single phase converter transformation: Inductor current waveforms ...................................................... 28
2.10. GPC control loop configuration during design and implementation. 29
2.11. Settling time and output voltage overshoot ....................... 34
2.12. Settling time: 3D-profile mapping with $N_u = 5$ .................. 35
2.13. Settling time: 3D-profile mapping with $N_y = 50$ ............... 35
2.14. Settling time vs $\lambda$ ...................................................... 36
2.15. Maximum deviation ....................................................... 38
2.16. Performance index based on area .................................... 40
2.17. Cancellation of error areas .............................................. 41
2.18. Phase margin vs $\lambda$ ................................. 42
2.19. Phase margin and phase margin index vs. $\lambda$ ........... 43
2.20. Influence of the weighting factor $w$ on the transient response. . 44
2.21. Design example 1: 3D-profile of the settling time. .......... 50
2.22. Design example 1: 2D-profile of the settling time for $N_u = 5$; $N_y = 100$. ......................................... 51
2.23. Design example 1: Open-loop bode plot. .................... 52
2.24. Design example 1: Load transient simulation. .......... 52
2.25. Design example 1: Measured load transient performance. ... 54
2.26. Design example 2: Performance indices based on one criterion. 55
2.27. Design example 2: Combined performance index. .......... 56
2.28. Design example 2: Open-loop bode plot. .................... 58
2.29. Design example 2: Load transient simulation. .......... 58
2.30. Design example 2: Measured step performance. .......... 59

3.1. Standard DPWM modulation for a four-phase buck converter. 62
3.2. AM-DPWM with a four-phase buck converter. ............... 64
3.3. Standard DPWM modulation scheme. ......................... 66
3.4. Duty cycle distribution with the new modulation scheme: summary. .............................................. 66
3.5. Duty cycle distribution with the new modulation scheme: detailed step-by-step guide. ......................... 67
3.6. Alternative modulation scheme where more than one phase can be switched on at the start of each subcycle. ............ 72
3.7. DPWM implementation diagram. .............................. 73
3.8. Phase state machine. ........................................ 74
3.9. Prototype system. ........................................... 76
3.10. Bode plots of the open-loop for different sampling times. ... 79
3.11. Bode plot for control loop and additional filter. ............ 80
3.12. Implementation of a comb-filter with selectable order. ..... 81
3.13. System response of the new DPWM to a 70 A load step. ... 83
3.14. System response of the new DPWM to a 70 A load step with DPWM output signals. .............................. 83
3.15. System response of a standard DPWM to a 70 A load step. .. 84
3.16. Phase shedding operation with the new modulation scheme. 84
4.1. System architecture shown with four parallel DC-DC converters. 88
4.2. Overview of parallelization concepts. 90
4.3. Multi-phase solution with central power controller (Cat. 1). 91
4.4. Multi-converter solution with chain/star communication line (Cat. 2). 91
4.5. Proposed digital core. 93
4.6. Output ripple current cancellation. 94
4.7. Input ripple current cancellation. 94
4.8. Phase alignment principle. 96
4.9. AC coupling circuit. 98
4.10. Digital implementation of phase detection/alignment block. 99
4.11. Phase alignment for a two converter prototype prior to the enabling of the proposed alignment algorithm. 104
4.12. Phase alignment for a two converter prototype after enabling of the proposed alignment algorithm. 104
4.13. Resulting phase alignment for a four converter prototype. 105

5.1. System architecture shown with four parallel DC-DC converters. 107
5.2. Proposed digital core. 111
5.3. Optimization of the output currents. 114
5.4. Flow chart of the current distribution procedure. 115
5.5. Typical efficiency of a power converter. 116
5.6. Typical efficiency of a modular power converter for different numbers of active converters: $V_{in} = 12\, V$, $V_{out} = 1.5\, V$. 118
5.7. Typical efficiency of a traditional multi-phase power converter compared with the new current distribution scheme applied to a system with identical converters: $V_{in} = 12\, V$, $V_{out} = 1.5\, V$. 122
5.8. Implementation of the current distribution scheme. 123
5.9. Finite-state-machine used to implement the current distribution scheme. 124
5.10. Digital control loop with predictive current mode control. 124
5.11. Typical sequence of events for the proposed current distribution scheme. 126
5.12. Measured efficiency of a four converter prototype system compared with a standard current sharing scheme. 127
5.13. Measured efficiency variation for repetitive measurements with
arbitrary load current steps. . . . . . . . . . . . . . . . . . . . . . . . 129

B.1. Schematic of the prototype system: Power stage . . . . . . . . . . . 152
B.2. Schematic of the prototype system: Current sensing . . . . . . . . 153
B.3. Schematic of the prototype system: Output voltage sensing . . 154
B.4. Schematic of the prototype system: Analogue digital converter 155
B.5. PCB of the prototype system: Top view . . . . . . . . . . . . . . . . 156
B.6. PCB of the prototype system: Bottom view . . . . . . . . . . . . 156

C.1. Measured efficiency variation for repetitive measurements with
constant load current steps. . . . . . . . . . . . . . . . . . . . . . . . 158
C.2. Measured efficiency variation for repetitive measurements with
constant load current steps (time lines). . . . . . . . . . . . . . . . 158
C.3. Measured efficiency variation for repetitive measurements with
arbitrary load current steps. . . . . . . . . . . . . . . . . . . . . . . . 159
C.4. Measured efficiency variation for repetitive measurements with
arbitrary load current steps (time lines). . . . . . . . . . . . . . . . 159
**Glossary**

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADC</td>
<td>Analogue digital converter</td>
</tr>
<tr>
<td>AM-DPWM</td>
<td>Automatic multi-phase digital pulse width modulator</td>
</tr>
<tr>
<td>ASIC</td>
<td>Application-specific integrated circuit</td>
</tr>
<tr>
<td>C++</td>
<td>High-level programming language</td>
</tr>
<tr>
<td>CPU</td>
<td>Central processing unit</td>
</tr>
<tr>
<td>Cycle</td>
<td>Switching cycle of a DPWM or SMPS. Refers to the time period in which all phases switch once, i.e. the switching period per phase.</td>
</tr>
<tr>
<td>DAC</td>
<td>Digital analogue converter</td>
</tr>
<tr>
<td>DCR</td>
<td>DC resistance</td>
</tr>
<tr>
<td>DPA</td>
<td>Distributed power architecture</td>
</tr>
<tr>
<td>DPWM</td>
<td>Digital pulse width modulator</td>
</tr>
<tr>
<td>DSP</td>
<td>Digital signal processor</td>
</tr>
<tr>
<td>ESR</td>
<td>Equivalent series resistance</td>
</tr>
<tr>
<td>FPGA</td>
<td>Field programmable gate array</td>
</tr>
<tr>
<td>FSM</td>
<td>Finite state machine</td>
</tr>
<tr>
<td>GPC</td>
<td>Generalized predictive control</td>
</tr>
<tr>
<td>GUI</td>
<td>Graphical user interface</td>
</tr>
<tr>
<td>IBA</td>
<td>Intermediate bus architecture</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>IC</td>
<td>Integrated circuit</td>
</tr>
<tr>
<td>Java</td>
<td>High-level programming language</td>
</tr>
<tr>
<td>LQG</td>
<td>Linear quadratic gaussian</td>
</tr>
<tr>
<td>LSB</td>
<td>Least significant bit</td>
</tr>
<tr>
<td>MATLAB</td>
<td>Software package for various types of simulations and mathematical operations</td>
</tr>
<tr>
<td>MCU</td>
<td>Microcontroller unit</td>
</tr>
<tr>
<td>MIMO</td>
<td>Multi-input multi-output</td>
</tr>
<tr>
<td>MOSFET</td>
<td>metal-oxide-semiconductor field-effect transistor</td>
</tr>
<tr>
<td>MSB</td>
<td>Most significant bit</td>
</tr>
<tr>
<td>PB</td>
<td>Power block</td>
</tr>
<tr>
<td>PCB</td>
<td>Printed circuit board</td>
</tr>
<tr>
<td>Phase</td>
<td>Part of a power converter, including switches, inductor, capacitors, but no controller</td>
</tr>
<tr>
<td>PID</td>
<td>Proportional integral derivative</td>
</tr>
<tr>
<td>PLC</td>
<td>Programmable logic controller</td>
</tr>
<tr>
<td>PLECS</td>
<td>Piecewise linear electrical circuit simulation; a Matlab toolbox</td>
</tr>
<tr>
<td>PLL</td>
<td>Phase-locked loop</td>
</tr>
<tr>
<td>PMBus™</td>
<td>Standardised digital communication bus for power converters</td>
</tr>
<tr>
<td>POL</td>
<td>Point-of-load</td>
</tr>
<tr>
<td>RMS</td>
<td>Root mean square</td>
</tr>
<tr>
<td>Simulink</td>
<td>Graphical environment for the MATLAB tool suite</td>
</tr>
<tr>
<td>SISO</td>
<td>Single-input single-output</td>
</tr>
<tr>
<td>SMPC</td>
<td>Switched-mode power converter; used synonymously to SMPS</td>
</tr>
<tr>
<td><strong>SMPS</strong></td>
<td>Switched-mode power supply</td>
</tr>
<tr>
<td>----------</td>
<td>----------------------------</td>
</tr>
<tr>
<td><strong>SPICE</strong></td>
<td>Simulation program with integrated circuit emphasis</td>
</tr>
<tr>
<td><strong>Subcycle</strong></td>
<td>Time interval between the switching actions of two consecutive phases in a multi-phase power converter. For example, a four-phase converter has four different subcycles.</td>
</tr>
<tr>
<td><strong>VCO</strong></td>
<td>Voltage controlled oscillator</td>
</tr>
<tr>
<td><strong>VRD</strong></td>
<td>Voltage regulator down</td>
</tr>
<tr>
<td><strong>VRM</strong></td>
<td>Voltage regulator module, also called voltage regulator down (VRD)</td>
</tr>
</tbody>
</table>
1. Introduction and Motivation

In this chapter, an overview of the problems arising in the modulation of multiple parallel power converters/multi-phase converters is given. Different modulation and control approaches are reviewed and the need for further improvements in the area is highlighted. The structure of this thesis is outlined and the contribution of the research performed is detailed.

1.1. Fuelling today’s digital loads

Today’s power distribution systems require an increasing number of voltage converters. One common input voltage, typically provided by a switched-mode power supply or a battery, is converted into several different voltage levels (rails) to fit today’s application-specific power requirements. This can be either a single rail with high output power, e.g. CPU voltage regulator modules (VRMs) with more than 100 A, or multiple different power levels, for example found in field programmable gate array (FPGA) applications where up to three different voltage levels are required. Mobile devices, such as cell phones, employ an even higher number of rails (up to 50) to supply the different functional units, such as microprocessors, WiFi or Bluetooth chipsets, with adequate voltage levels. Individual power supplies enable the shut down of the different units when not in use. This technique is extensively used to increase battery life-time. Typically, power-sequencing is also required, which ensures that the different circuit/system components are powered up or down in the correct sequence.
Along with the increasing number of output rails, a change in system architecture has been observable in recent years [1]. In the past, small independent power supplies were predominant on the market, referred to as distributed power architecture (DPA). These supplies convert a common external input voltage, e.g. 48 V, into the desired output voltages in a single step (figure 1.1).

To reduce cost and size, these systems have been substituted with systems employing an intermediate bus architecture (IBA) as shown figure 1.2. One central power converter, referred to as bus converter, transforms the input voltage into an intermediate bus voltage which is then distributed over the printed circuit board (PCB). Small point-of-load (POL) converters convert the bus voltage into the voltage levels required by the individual loads. Each converter requires its own control integrated circuit (IC) (analogue or digital) and may be connected to a power management bus, such as PMBus™ [2].

To further reduce costs, systems without individual control ICs have been developed (figure 1.3) where the control and management functionality of all converters is centralized into one IC providing the control signals for individual power blocks (PBs). Each power block consists of the power switches and passive components of a converter, but does not feature a control IC.

As an alternative trend, physical integration of the control IC with the drivers and power switches into a single IC package can be observed. However, these solutions typically do not provide communication interfaces, but instead prove a reasonable choice for low-cost applications.

The optimal architecture is application-dependent based on criteria such as communication requirements, power management options and physical location of the converters. Along with the changes in architecture, a change in the control ICs is observable where a trend towards digital controllers is clearly evident.

However, current state-of-the-art converters utilize predominantly analogue control loops (figure 1.4) with switching frequencies in the range of 200 to 300 kHz. These offer a good trade-off between transient response, size and
Figure 1.1.: Distributed power architecture

Figure 1.2.: Intermediate bus architecture with multiple POL converters

Figure 1.3.: Centralized control IC with power blocks
efficiency. However, the future development is clearly moving in the direction of higher output power occupying less space; especially as the size of the inductors and capacitors is ever decreasing. With desktop computers and servers running 24 hours a day – 365 days a year – efficiency is now also a very important design factor.

As an additional driver for the use of digitally controlled power converters, system level integration has been identified. Digital communication protocols, such as PMBus™, enable the standardized communication between a host controller and the connected power supplies in order to configure parameters, such as the output voltage, or harvest information such as output current or temperature. Recent developments increase the demand for such management functionality, as for example in server applications, power supplies are now required to monitor the delivered energy precisely, so that the customers can be charged appropriately [3].

To satisfy these demands, hybrid systems, referred to as digital wrappers, have been developed. With reference to figure 1.5, they combine a fast analogue control loop with digital circuitry to handle the communication functionalities. This allows the use of pre-existing control expertise in the analogue world with the advantages of digital system integration. However, hybrid systems are clearly an interim solution.

Consequently, fully digital systems have been developed. With reference to figure 1.6, these integrate analogue digital converters (ADCs), digital control loops and digital pulse width modulators (DPWMs) on a single chip. However, most of the digital control laws currently used are still transformations of analogue control systems and incorporate simple proportional integral derivative (PID) compensators. Special direct-digital design approaches have proven to provide better performance or additional functionality, but their design is not straightforward. Software design tools could aid engineers to design compensators with such advanced digital control schemes without the need for extensive knowledge in digital control.
Figure 1.4.: Buck converter with analogue control loop.

Figure 1.5.: Buck converter with analogue control loop and digital wrapper.

Figure 1.6.: Buck converter with digital control loop.
The lack of practical design tools is addressed in this thesis (chapter 2), where an automatic design procedure for direct digital compensators has been developed. This enables practising engineers to design application-specific, high performance digital compensators without requiring extensive knowledge of the discrete (digital) control domain. During the optimization process, real-life performance indices allow users to specify their application requirements and hence generate suitable compensators.

The application of power supplies spans from low output power, such as in mobile phones, to high output power, such as CPU power supplies. The different applications may have contradictory design requirements and challenges. Some of these challenges are now analysed and addressed with new system concepts proposed. However due to the contradicting requirements, a “universal power controller” has not been developed.

One major design challenge is fast transient performance, especially for today’s high-speed digital loads, such as CPUs. As this is the most dominant design factor, it has gained significant research interest and has been extensively investigated in the literature. Various different solutions have been proposed; a brief overview is now given.

Charge-balance and time-optimal control [4–9] utilize a single corrective step to compensate for a load transient in the minimal possible settling time. This is achieved using a fast analogue loop in parallel with a digital loop to reduce sampling delay. However, these concepts are highly sensitive to parameter changes, e.g. due to system tolerances or temperature variations, and rely on precise information about output voltage and current.

An alternative control scheme, referred to as “Linear–Non–Linear” control, uses a fast hysteretic loop in parallel with a PID control loop which improves the transient performance significantly [10–14]. The design challenge is the selection of the comparator thresholds controlling the additional loop. An analytical method for their calculation has not been published to date.
In [15], an “event-based” system is demonstrated which utilizes an asynchronous ADC to trigger the control loop every time the output voltage changes. The control loop is modified in such a way that the compensator can be executed with a non-fixed sampling frequency by dynamically scaling the coefficients to “simply emulate” an analogue (continuous-time) compensator. The system reduces the latency time, but can suffer from high-frequency noise as there is no limitation of the sampling frequency by the ADC. Also, asynchronous ADCs are not common in power designs (yet) and can provide a potential hurdle in practical applications due to their power consumption and noise sensitivity.

Oversampled compensators have also been proposed which reduce the sampling delay of the digital loop by sampling with frequencies higher than the switching frequency [16–20]. Compared with time-optimal control, oversampled compensators tend to be more robust to system parameter variations as the single step control action is split into several independent action steps. Unlike time-optimal control, the oversampled feedback loop is typically linear and stability can be proven. However with the use of faster digital compensators, the DPWMs, i.e. the interfaces between the digital and the analogue world, are becoming a limitation on system performance.

Another design challenge is efficiency over the full operation range. While previous standards have defined efficiency levels mostly for full output power, recent standards require high efficiency over the full operation range; typically measured at multiple predefined power levels [21]. Different techniques to improve efficiency have been developed, such as dead-time optimization, alternative circuit architectures and phase shedding.

Phase shedding, a technique to dynamically change the number of phases used in multi-phase power converters depending on the current output current, has been proven to be effective and implementable [22]. Therefore, it is widely used in today’s power converter system architectures. However with an increasing number of phases, existing modulator concepts are not able to fulfil these requirements optimally.
The issues of transient performance, efficiency and scalability are addressed in this thesis with the investigation of a new automatic multi-phase digital pulse width modulator (AM-DPWM) in chapter 3. The proposed modulator features an internal redistribution of the duty cycle commands over the available phases in the fastest possible way with respect to the number of switching actions per phase. This reduces the delay in the modulator and simultaneously enables the use of phase shedding with any number of phases. The proposed closed-loop system utilizes a simple digital compensation scheme independent of the number of active phases.

While the systems discussed so-far have been focused on systems with one central controller, another design criterion is modularity. In some applications it is advantageous to have modular control systems without a central controller. For example, high-reliability systems require redundancy and hot-swapping so that a central control architecture is rendered unsuitable. Additionally, such modular systems can be fully integrated with the power semiconductors into one single IC package to provide a reduction in PCB area and cost.

Present modular systems require communications lines between the individual converters to enable the use of advanced control features such as phase-shedding. In this thesis, new approaches how such advanced control features can be integrated without the need for communication signals are investigated. In chapter 4, the issues of phase alignment and frequency synchronization are addressed. The proposed system uses the perturbances created by the switching actions of the individual power converters on the input voltage to harvest information about its counterparts. This information is used to synchronize the switching frequencies and align the phases.

In chapter 5, a current distribution concept of independent power supplies is proposed which does not require communication lines. The output power of the different converters is optimized based on their individual efficiency curves and therefore allows for a more optimal output power delivery. Additionally, this technique enables the use of phase shedding techniques by the individual controller without knowledge of the total output power.
To summarize, the properties of the different system architectures are illustrated in figure 1.7 where transient performance and efficiency of several different existing solutions are compared with the approaches presented in this thesis.

**Figure 1.7.:** Comparison of different system concepts: Transient performance vs. efficiency
1.2. Thesis structure

The thesis is structured so that the different chapters cover the different contributions and publications. The individual chapters are written in such a way that they stand on their own. Chapters can provide more insights into the discussed topics when read together.

As outlined this thesis covers three different topics in four chapters. Chapter 2 presents an automated design procedure to automatically design compensators for digitally controlled power supplies. Chapter 3 focuses on the modulation of multi-phase converters where a new modulation principle is proposed. Chapter 4 focuses on phase alignment and frequency synchronization for independent, parallel power converters. Chapter 5 proposes a new current distribution principle to improve efficiency. Chapter 6 summarizes the contributions of this work with future work objectives outlined.

1.3. Contribution

The research performed during the course of this thesis has been published in several publications and one pending patent:


Two journal papers have been submitted for the consideration in the IEEE Transaction of Power Electronics with the review currently pending.
2. Automated Compensator Design using Real-life Design Criteria

The design of compensators is a critical task in the application of today’s digital power converters. In this chapter an automated direct-digital design process is proposed which allows the automated design of compensators specifically optimized for user-defined, application-specific design parameters. Real-life design criteria are used to ease the applicability of the concept. Firstly, a short introduction in the area of Generalized Predictive Control is given, followed by the derivations required for the proposed compensator design procedure. Several different real-live design parameters are analysed in the next section and their applicability assessed using an experimental setup. This is followed by the assessment of different optimization strategies for the proposed automated design process. In section 2.6, two design cases are detailed as a step-by-step guide and used to give additional insights into the design process. An experimental prototype is used to investigate the applicability of the derived compensators in the real world.
2.1. Introduction

2.1.1. Digital control of SMPS

Digital control of switched-mode power supply (SMPS) is a continuing trend in research and industry. With analogue controllers predominant on the market, the use of digital power controller has increased in recent years. However, most advantages of digital controllers can only be utilized when new compensator design methods are employed. On the other hand, most existing digital designs still use classic compensator design methods well-known in the continuous-time domain and converted into the digital domain. Consequently, the compensators are discretized with the sampling time and implemented in digital circuits. However, direct-digital design methods such as direct-pole-placement or Generalized Predictive Control (GPC) provide significant advantages over existing methods.

With reference to figure 2.1, a digital control loop consists of a power stage, an ADC, a digital control law \( \left( \frac{N(z)}{D(z)} \right) \) and a DPWM. The ADC converts the analogue system variables, such as output voltage and inductor current, into the digital domain, where the digital control law computes the next duty cycle value. The duty cycle is converted back into the analogue domain by the DPWM which provides the control signals for the power switches. The digital control law can be implemented on several different platforms, such as digital signal processors (DSPs), field programmable gate arrays (FPGAs) or application-specific integrated circuits (ASICs).

![Digital control loop](image)

**Figure 2.1:** Digital control loop.
Users of power controllers, analogue and digital, require simple design procedures to ensure well-designed compensators and to speed-up the design process. Analogue compensators are typically designed in the frequency domain using techniques such as bode plots, loop-shading or predefined design guidelines. Today, the same techniques are also used in the digital domain as they are widely established and well understood. However, direct-digital design methods provide advantages over existing schemes, but their practical design is more complicated.

2.1.2. Compensator design procedures

A need for automated design methods to ensure the “quick-and-easy” design of digital compensators for the practising engineer is clearly evident. Whereas existing methods are largely based on trial-and-error, know-how and design experience, the proposed method allows an automated design without extensive knowledge in control engineering.

Most practical designs still use simple PID compensator designs with simple design rules. Such design rules typically recommend the positioning of the pole and zeros of the compensator relative to the power stage’s poles and zeros; examples can be found in [27, 28]. The values of the components used to implement the analogue compensator (resistors and capacitors) are derived from these poles and zeros. However, these design procedures often require iterative, time-consuming fine tuning. Such compensators can be converted into the digital domain using standard discretization techniques, but do not provide the benefits of direct-digital design methods.

To expand, one common issue with such design methods is dealing with low equivalent series resistance (ESR) power stages which are becoming more dominant due to the increasing use of ceramic capacitors. Power converters with ceramic capacitors are gaining more and more market share, as they are generally more efficient, due to lower ESR, have relatively long life-times and consume less space than their electrolytic counterparts.
Compared with the analogue world, the digital world is different. Dedicated software tools are used to configure digital power converters via serial communication interfaces, such as PMBus™. Examples for these software tools are Texas Instruments’s Digital Power Software [29], Powervation’s Digital Power Interface Center [30] or Exar’s Digital Power Design Studio [31]. These design interfaces enable the user to configure the IC for their application-specific needs using a graphical user interface (GUI). A GUI provides an abstract view of the configuration without the need to interfere with the actual bits in the configuration registers. These configuration registers, typically flash memory, store all non-volatile information required by the power controller, such as compensator coefficients, reference voltages, fault-behaviour, etc. The design GUI provides the ideal platform to implement automated design procedures which enable the user to derive optimized compensators.

Several different design procedures can be used for such automated systems. For example in [32], a direct-digital design approach has been analysed which derives a compensator such that direct placement of the closed-loop poles is possible. While the position of the poles can be optimized with an automated design process, one of the problems with this approach is the appropriate positioning of the poles. A wrong selection can result in poor performance or even an unstable system.

Often in automated design methods, optimal control approaches are chosen [33, 34]. For example in linear-quadratic-gaussian (LQG), an optimal compensator is derived minimizing a predefined optimization problem which rates the error signal quadratic. For an infinite optimization, this results in a fixed linear compensator. However, this generally requires the solution of the Ricatti-equation which is computationally inefficient.

Generalized Predictive Control is a dedicated time-domain approach from the family of optimal control techniques. It is computationally more efficient as it can be solved over a finite horizon, can guarantee stability and allows the incorporation of constraints. This is explained in more detail in the following section.
2.2. GPC Compensator Design for Linear Unconstrained Systems

2.2.1. Generalized Predictive Control – An overview

In this section, the reader is introduced into the general area of Generalized Predictive Control.

In contrast to other design methods (which are often based in the frequency-domain), GPC is a time-domain approach. A control loop containing a GPC derived compensator is shown in figure 2.2 while the basic concept is illustrated in figure 2.3. The algorithm predicts the future output values of the system at each step $n$ using a reference plant model $P^*(z)$ over the output prediction horizon $N_y$. The optimal sequence of control values $\Delta u$ is then calculated based upon the predicted output values over the control horizon $N_u$ by optimizing/minimizing a given cost-function.

This cost function is based on time-domain criteria and therefore has the ability to optimize the transient response of a power converter. A typical cost function is

$$
\min J = \sum_{i=0}^{N_y} \| r_{k+i} - y_{k+i} \|^2 + \lambda \sum_{i=0}^{N_u} \| \Delta u_{k+i} \|^2.
$$

(2.1)

It is based on the sum of the squares of the error $e$ between the reference value $r$ and the predicted output value $y$ over a defined number of steps (called prediction horizon $N_y$) and the sum of the squares of the control actions $\Delta u$.

![Figure 2.2: Control loop incorporating a GPC derived compensator.](image-url)
over the control horizon $N_u$ weighted with $\lambda$. The control action sequence, which minimizes the cost function, is considered the optimal control sequence and hence applied to the power stage.

To ensure that the computed control values are optimal at all times, the GPC optimization process is executed at every step $k$, taking into account the latest information about the system, i.e. the most recent sample. Because the optimization of the cost function can be computationally very intensive, real-time computation of GPC has only been used to date in applications with relatively large time constants. This includes programmable logic controllers (PLCs) in industrial and petro-chemical applications, among others.

The complexity of the resulting optimization problem is also dependent on the constraints of the problem. GPC is able to control constrained systems as long the solver for the optimization problem/cost function, is able to handle these constraints. However, the required optimization during run-time is not practical for power converters to date, due to the limitations in computational power and high switching frequencies. However for unconstrained linear systems, the GPC optimization problem can be solved explicitly.

As DC-DC converters can be modelled as linear, time-invariant systems [35], it is not necessary to solve the optimization problem online; therefore GPC algorithms can be utilized offline. In this case, the cost-function can be solved analytically producing a fixed linear control law that can be implemented online.
With reference to (2.1), the three GPC design parameters $N_u$, $N_y$ and $\lambda$ determine the compensator coefficients and therefore the closed-loop performance. Their selection is one of the key steps in a GPC design method.

The output prediction horizon $N_y$, i.e. the number of steps the optimization takes into account, influences the speed and hence the stability of the resulting compensator. When the horizon is too small, the algorithm is not able to predict the dynamics of the system properly and the resulting compensator delivers poor performance, or is even not stable. If the prediction horizon is much larger than the system dynamics, the mathematical burden increases without performance improvements.

The control horizon $N_u$, i.e. the number of control signals considered during the optimization, is generally much smaller than the output horizon and in the magnitude of a couple of steps. It represents the number of control actions applied to the system to regulate the output back to the reference value. For example, a horizon of one, which allows the system exactly one control step, generally results in very poor performance [36]. If the control horizon matches the plant’s order, the resulting compensator is of a dead-beat type. However, such compensators are typically not suitable for power converters due to their aggressiveness. Based on experience, a control horizon of four to ten steps is recommended for the proposed design procedure.

Broadly speaking, the weighing factor $\lambda$, i.e. the ratio between control action $\Delta u$ and error signal $(r - y)$ in the cost function, can be viewed as a measure of the speed of the compensator. Small $\lambda$s lead to fast compensators, as control action is rated more important in the cost function than the output error. On the other hand, if $\lambda$ is large, the compensator will be very slow, as the cost function is dominated by the control action. In this case, the contribution of the output error is insignificant.

A note on stability should also be given. By design, a GPC designed compensator is always stable for a linear system when “the output horizon $N_y$ is large enough” [36]. This can be explained by the high cost generated by the oscillating-mode of an unstable closed-loop (figure 2.4).
Unfortunately, a DC-DC converter is not a fully linear system due to non-linear effects, such as duty cycle limitation and non-linear component behaviour. However, in the range of interest here, and for compensator designs suitable in practical application, this non-linear behaviour is not a concern and does not influence the compensator’s stability.

In the next section, the mathematical background and the underlining GPC equations are given. It should be noted that the proposed GPC algorithm can be seen as a black box design procedure (figure 2.5) with three input parameters $N_u$, $N_y$ and $\lambda$ (beside the reference plant’s transfer function $P^*(z)$) which returns the transfer function of the GPC derived compensator $\frac{N(z)}{D(z)}$. The MATLAB function implementing this algorithm is detailed in appendix A.2.

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{figure2_4.png}
\caption{Costs of an oscillating mode.}
\end{figure}

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{figure2_5.png}
\caption{Abstract view of the GPC algorithm.}
\end{figure}

19
2.2.2. Mathematical background

In this section the mathematical background of GPC is given. It is a summary of the work of Maciejowski and Rossiter presented in [36, 37]. The notation is consistent with the notation used in the given references. Note that additional information can be found in appendix A.1. Application specific procedures, such as the definition of the used transfer function model etc. are not given in this section as they are given in the next sections.

With reference to [36], an optimal control sequence is considered to be a sequence of input values into a control system which optimizes a given cost function. With reference to figure 2.3 and (2.1), the sum of the squares of the error $e$ between the reference value $r$ and the predicted output value $y$ and the sum of the squares of the predicted control actions $\Delta u$ weighted with $\lambda$ are chosen. In a vectorized form, this can be expressed as

$$
\min J = \left\| r \rightarrow - y \rightarrow \right\|_2^2 + \lambda \left\| \Delta u \rightarrow \right\|_2^2,
$$

(2.2)

where $r \rightarrow$ is the vector of the future reference values $r_n$, $y \rightarrow$ the predicted output values and $u \rightarrow$ the computed control actions. Note that the vectorized form allows a description of the control problem for any model, single-input single-output (SISO) and multi-input multi-output (MIMO). Details on the vectorized notation can be found in appendix A.1.

In order to solve (2.2), a prediction of the future output values of the system is required. For the proposed algorithm, the output of the nominal control system is used for such a prediction. Considering a standard control configuration, as shown in figure 2.2, the system equation of the plant in incremental form is given by

$$
A(z)y_k = b(z)\Delta u_k,
$$

(2.3)

with

$$
A(z) = 1 + A_1 z^{-1} + A_2 z^{-2} + \cdots + A_{n+1} z^{-n-1}
$$

(2.4)

$$
b(z) = b_1 z^{-1} + b_2 z^{-2} + \cdots + b_{n+1} z^{-n-1},
$$

(2.5)
where $A_x$, $b_x$ are the plant parameters. This alternative representation of the well-known difference equation of a digital system proves very useful in this context and allows a short efficient description of the control algorithm. Following from (2.3), the next output of the control system can be predicted as

$$
y_{k+1} = -A_1 y_k - A_2 y_{k-1} - \cdots - A_{n+1} y_{k-n} + b_1 \Delta u_k + b_2 \Delta u_{k-1} + \cdots + b_{n-1} \Delta u_{n-k+1}.
$$

This output value consists of two superimposed parts. The free response describes the behaviour of the output due to the previous system inputs, while the controlled response is the contribution caused by the current control signal, $\Delta u_k$ which is the degree of freedom in this case. Keeping this in mind, (2.6) can be reformulated as

$$
y_{k+1} = -[A_1, A_2, \ldots, A_{n+1}] y + [b_2, b_3, \ldots, b_{n-1}] \Delta u_{k-1} + b_1 \Delta u_k.
$$

While (2.7) is the prediction of the next output value, the prediction of all $N_y$ future output values is required to compute the cost function and the compensator. This prediction can be obtained by an iterative use of (2.6) to obtain $y_{k+2}, y_{k+3}, \ldots, y_{k+N_y}$. To simplify the prediction equation used in the next section, a compact matrix notation can be defined as
\[
\begin{bmatrix}
1 & 0 & \cdots & 0 \\
A_1 & 1 & \cdots & 0 \\
A_2 & A_1 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
\end{bmatrix}
\begin{bmatrix}
y_{k+1} \\
y_{k+2} \\
\vdots \\
y_{k+N} \\
\end{bmatrix}
+ \begin{bmatrix}
A_1 & A_2 & \cdots & A_{n+1} \\
A_2 & A_3 & \cdots & 0 \\
A_3 & A_4 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
\end{bmatrix}
\begin{bmatrix}
y_k \\
y_{k-1} \\
\vdots \\
y_{k-n} \\
\end{bmatrix}
= \begin{bmatrix}
y_k \\
y_{k-1} \\
\vdots \\
y_{k-N} \\
\end{bmatrix}
\]
\]

\[
\begin{bmatrix}
b_1 & 0 & \cdots & 0 \\
b_2 & b_1 & \cdots & 0 \\
b_3 & b_2 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
\end{bmatrix}
\begin{bmatrix}
\Delta u_k \\
\Delta u_{k+1} \\
\vdots \\
\Delta u_{k+N-1} \\
\end{bmatrix}
+ \begin{bmatrix}
b_2 & b_3 & \cdots & b_n \\
b_3 & b_4 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
\end{bmatrix}
\begin{bmatrix}
\Delta u_{k-1} \\
\Delta u_{k-2} \\
\vdots \\
\end{bmatrix}
= \begin{bmatrix}
\Delta u_{k-1} \\
\Delta u_{k-2} \\
\vdots \\
\Delta u_{k-n+1} \\
\end{bmatrix}
\]
\] (2.8)

Now with the use of Hankel and Toeplitz notations (see appendix A.1 for details) for the matrices and the prediction notation of the vectors, (2.8) can be written in a very compact form as

\[
C_A y_k + H_A y_k = C_{zb} \Delta u_k + H_{zb} \Delta u_k .
\] (2.9)

From this the prediction outputs can be obtained as

\[
y_k = C_A^{-1}[C_{zb} \Delta u_k + H_{zb} \Delta u_k - H_A y_k].
\] (2.10)

With the definition of \( H = C_A^{-1}C_{zb}, \ P = C_A^{-1}H_{zb} \) and \( Q = -C_A^{-1}H_A \), this is further simplified to

\[
y_k = H \Delta u_{k-1} + P \Delta u_{k-1} + Q y_{k-1}.
\] (2.11)

With this, the cost function given in (2.2) can be solved by substituting (2.11) in (2.2). The value of \( \Delta u_k \) for minimal costs is computed via the first derivative of \( J \) with respect to \( \Delta u_k \) which represents the optimal sequence of control signals taking into account the information available at step \( k \).

\[
\frac{\delta J}{\delta \Delta u_k} = 0
\] (2.12)
Note that the computed vector $\Delta u \rightarrow$ contains the next $N_u$ (all) control actions as generated by the control algorithm. In practice only the first element of this vector is applied to the system and the optimization repeated at the next time step $k + 1$. This ensures up-to-date control actions taking into account all available information.

For unconstrained linear systems, this optimization is explicitly solvable and can be expressed in analytical form. In such cases, the GPC control law can be defined as the first element of the vector $\Delta u_k$.

$$\Delta u_k = P_t r - N_k y - \tilde{D}_k \Delta u$$

with

$$P_t = \varepsilon^T_1 (H^T H + \lambda I)^{-1} H^T$$

$$N_k = \varepsilon^T_1 (H^T H + \lambda I)^{-1} H^T Q$$

$$\tilde{D}_k = \varepsilon^T_1 (H^T H + \lambda I)^{-1} H^T P$$

(2.13)

While (2.13) describes the system in matrix description, a transfer function representation is more applicable to the system in mind. By substituting the transfer function representation of the matrices $P$, $Q$ and $H$ into (2.13), we can obtain the transfer function representation of the unconstrained GPC control law as

$$D_k(z) \Delta u_k = P_t(z) r_k - N_k(z) y_k,$$

(2.17)

where the polynomial vectors, $D_k(z)$, $P_t(z)$, $N_k(z)$, are defined as the polynomial vectors build out of the individual elements of the original matrices (see appendix A.1 for details).

Equation (2.17) has been implemented as MATLAB routine (see appendix A.2) which takes the three design parameters $N_u$, $N_y$ and $\lambda$ (and the plant model) as inputs and returns the transfer function of the GPC derived compensator.
2.3. Automated Design Process using GPC

The proposed automated design process using the GPC algorithm is now outlined and can be quantified in four fundamental steps:

1. Selection of a plant model
2. Selection of a suitable performance index
3. Execution of the automated design procedure to search for the optimal set of GPC design parameters
4. Evaluation of the designed compensator

The selection of a plant model used as prediction model during the design process is a crucial step in the development of the design procedure. The model has to fulfil GPC specific requirements and also represent the power stage accurately. However, as soon as a suitable model has been chosen, the user’s only remaining task is the input of the power stage parameters, such as component values and sampling frequency.

The selection of a suitable performance index is the second step during the design procedure. The performance index is used to rate the performance of the compensator during the design process. Its selection is of important influence on the resulting compensators as the compensator with the highest performance rating is chosen by the design process. During the course of this work, several different performance indices have been evaluated and their application-specific implications have been identified. It is up to the user to decide whether a predefined or a user-defined performance is chosen.

As a third step, the user has to “press a button” and the design routine searches for the optimal set of GPC design parameters. This process is fully automated and allows the quick design of a compensator. It is based on the flow chart shown in figure 2.6 where the performance index and the plant model are considered as inputs. The procedure selects the three GPC design parameters, i.e. $N_y$, $N_u$, $\lambda$, and designs a compensator using the GPC algorithm detailed in subsection 2.2.2. The closed-loop performance of the resulting system is then assessed using the previously defined, user-specific performance index.
Based on this assessment, the GPC design parameters are varied until an “optimal solution” is found.

Finally, it is highly recommended that the user evaluates the automatically designed compensator. Note that the compensator is optimal for the chosen performance index. If it does not fulfil the application-specific requirements, the performance index has to be changed and a new compensator designed.

2.3.1. Selection of a prediction model

As with any compensator design process, GPC requires an accurate model of the plant which is used to predict future output values based on previous output values, previous input values and future control actions. A typical control loop, shown in figure 2.2, consists of a compensator $\frac{N_k(z)}{D_k(z)}$, a plant model $\frac{\hat{B}}{\hat{A}}$ and an optional reference filter $\frac{P_r(z)}{D_k(z)}$.

Several different suitable DC-DC converter models are proposed in the literature, e.g. [38–40]. This work focuses on buck converters with the basic schematic shown in figure 2.7. The chosen transfer function (2.18) is a standard continuous-time transfer function which is based on the averaging of

\[ \frac{N_k(z)}{D_k(z)} \]

\[ \frac{\hat{B}}{\hat{A}} \]

\[ \frac{P_r(z)}{D_k(z)} \]

Figure 2.6.: GPC design process.
the two switching states of the DC-DC converter over a switching cycle, [35].

For obvious reasons, the required discretization is generally not performed analytically but instead is calculated numerically using software tools, such as MATLAB.

\[
G_{d,V}(s) = \frac{V_{\text{out}}(s)}{d(s)} = V_{\text{in}} \cdot \frac{R_C Cs + 1}{L C \cdot s^2 + (R_L + R_C) C \cdot s + 1}
\]  

While (2.18) gives the relationship between duty cycle and output voltage, the output current is neglected. However for power converter applications, load transient performance is usually of more interest than output voltage tracking. It is possible to take the output current into account during the design of a GPC-based compensator, but in practice there is generally too little information to do so. Hence, the output current is not taken into account for the design of the compensator within the GPC algorithm, but for the computation of its performance index. The standard transfer function model is augmented by the load current represented by an unknown disturbance input. During the performance evaluation of the designed compensator, i.e. the computation of the performance index, this current input is used to apply a predefined load step to the power stage. The obtained response is then used to measure the performance of the compensator.

The load current can be introduced in different ways into the model, either as a load resistance, \(R_{\text{load}}\), or as a current source, \(I_{\text{out}}\). To date, a load resistance is used in most converter transfer function representations. In any case, both parameters are constant and represent the dynamics caused by

![Figure 2.7.: Single phase buck converter.](image-url)
the load current. In this work a separate transfer function, \( G_{I_{\text{out}}, V} \), is used to
dynamically represent the output current, similar to the technique outlined
in [41]. The continuous-time transfer function from load current to output
voltage response is given by

\[
G_{I_{\text{out}}, V}(s) = \frac{V_{\text{out}}(s)}{I_{\text{out}}(s)} = -\frac{R_C L C \cdot s^2 + (L + R_L R_C C) \cdot s + R_L}{L C \cdot s^2 + (R_C + R_L) C \cdot s + 1}
\]  

(2.19)

with the power train components \( C, L, R_C \) and \( R_L \) and is typically referred
to as output-loop output impedance. For the GPC algorithm presented, this
function is discretized numerically. The resulting configuration is illustrated
in figure 2.8 where the duty cycle, \( d \), is the control output. Since the duty
cycle control signal, \( d \), is the control variable in the system, \( G_{d, V} \) is used as
the prediction model. The current input, \( I_{\text{out}} \), is not controllable and is viewed
as a disturbance.

Note that the transfer function model chosen can also be used for multi-phase
converters. It is well-known in the literature that a multi-phase converter with
\( N \) phases can be modelled as a single-phase equivalent with a simple transfor-
mation of the components values [42]. The idea is based on the summation
of all inductor currents at the output capacitor and the analysis of the resulting
current waveform, as illustrated in figure 2.9. The transformation of the resis-
tive circuit elements has been derived and the resulting transformations listed
in table 2.1.

![Figure 2.8: Transfer function model.](image-url)
Figure 2.9.: Multi-phase single phase converter transformation: Inductor current waveforms.

Table 2.1.: Transformation of a multi-phase converter into a single-phase model.

<table>
<thead>
<tr>
<th></th>
<th>multi-phase</th>
<th>single-phase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of phases</td>
<td>N</td>
<td>1</td>
</tr>
<tr>
<td>Switching frequency</td>
<td>$f_{\text{switch}}$</td>
<td>$N f_{\text{switch}}$</td>
</tr>
<tr>
<td>Output voltage</td>
<td>$V_{\text{out}}$</td>
<td>$V_{\text{out}} = \text{const}$</td>
</tr>
<tr>
<td>Input voltage</td>
<td>$V_{\text{in}}$</td>
<td>$\frac{1}{N} V_{\text{in}}$</td>
</tr>
<tr>
<td>Inductor value</td>
<td>L</td>
<td>$\frac{1}{N} L$</td>
</tr>
<tr>
<td>On time</td>
<td>$T_{\text{On}}$</td>
<td>$T_{\text{On}} = \text{const}$</td>
</tr>
<tr>
<td>$R_{\text{DS(On)}}$ top switch</td>
<td>$R_{\text{DS(On)}}$</td>
<td>$\frac{1}{N} R_{\text{DS(On)}}$</td>
</tr>
<tr>
<td>$R_{\text{DS(On)}}$ bottom switch</td>
<td>$R_{\text{DS(On)}}$</td>
<td>$\frac{1}{N^2} R_{\text{DS(On)}}$</td>
</tr>
</tbody>
</table>
As the GPC design process is based on incremental control signals, $\Delta u_k$, instead of absolute control signals, $u_k$, a modification of the existing transfer function model is required. A system with incremental inputs can be realized by introducing an additional integral state into the existing control loop, as shown in figure 2.10. This also ensures steady-state accuracy as the integrator can compensate for discrepancies between the system model and the real system. Variations up to 20% or 30% are not uncommon in practice, due to manufacturing tolerances of the components, drift over time and temperature, etc. As shown in figure 2.10, a separate discrete integrator is introduced into the control loop. While this integrator is incorporated into the plant model for the GPC design process, it is considered part of the final compensator for implementation.

Additionally, the saturation of the control signal is not factored into the linear transfer function representation. Physically, the duty cycle of a DC-DC converter is limited between 1 (all on) and 0 (all off), whereas the duty cycle in the transfer function representation can be of any arbitrary value. As a result, in theory, highly tuned and very “fast” compensators with high gains can be designed. However, such compensators are highly undesirable in practical designs as they are highly sensitive to component variation and will experience saturation of the duty cycle. Physical restrictions can be taken into account

Figure 2.10.: GPC control loop configuration during design and implementation.
in two different ways: either the duty cycle is saturated by the model or the design algorithm has to ensure that the output values of the compensator do not exceed these limitations. In GPC, the cost function weights the control action through the parameter $\lambda$ and therefore avoids saturation.

In contrast a constrained GPC system can take saturation, such as duty cycle limitations or inductor current limitation, into account during the computation of the compensator. However as detailed above, the online solution of constrained GPC problems is not yet possible in the practice. An approach to tackle this problem have been presented recently [43, 44], but the system requirements (high-performance DSP and low switching frequency) are impractical.

Note that the design of the reference filter is generally not of interest for power converters as they typically operate with a constant or slowly changing reference voltage. Subsequently, the reference filter can be set to

$$P_r(z) = 1$$

and a constant reference value $r_k$ can be assumed.

### 2.3.2. Selection of a suitable performance index

The second step in the automated design process is the selection of a suitable performance index. This is detailed in section 2.4.

### 2.3.3. Automated design procedure / search for the optimal GPC parameter set

The third step in the automated design process is the automatic search for the optimal set of GPC parameters; optimal in the sense of the user-selected/user-specified performance index. The procedure follows the flow chart given in figure 2.6 and consists of three internal steps.
An initial set of GPC design parameters is chosen first and a compensator designed using the GPC algorithm. This compensator is “plugged” into a closed-loop system and its performance assessed using the performance index selected. Then in a third step, a new set of GPC design parameters is selected. This procedure is repeated iteratively until a termination criterion is met. This criterion depends on the chosen optimization/search technique and can include a maximum number of search steps, a target threshold or the discovery of a suitable optimal point.

Without any additional knowledge about the cost function or the system, the search space for the optimization algorithm can be viewed as a three-dimensional system with the three GPC design parameters as coordinates. Each point in this space has costs associated given by the value of the cost function for the closed-loop system of the GPC compensator derived for the coordinates. To reduce the computational burden and therefore the design time, a further analysis of the influence of the different design parameters has been done. Consequently, the search space can be reduced into a search over two design parameters, and over one respectively. Further details are given in the design examples in section 2.6.

As most performance indices require the closed-loop system transient response, the selection of the appropriate simulation tool to capture this closed-loop system performance is of critical importance. These are a trade-off between the accuracy of the results and the computational time as the capturing of the closed-loop system performance is the main contributor to the latter. Two different simulation techniques have been assessed during this work:

1. A Simulink simulation based on a transfer function model

A Simulink model of the transfer function, implementing the system shown in figure 2.8, has been investigated. Extended with saturation logic to detect duty cycle saturation during simulation, the closed-loop is very similar to the prediction model used for the design. It allows relatively fast simulation and
measurement of the system performance and provides an input port for the load step which is used for the transient performance assessment.

Closed-loop system simulations based on the PLECS toolbox enable the use of circuit schematics which allow for a much more accurate simulation of the physical system. For example, inductor current saturation can be detected as this can be an issue in tightly designed converters.

Generally speaking, the PLECS simulation requires much more simulation time, but delivers more accurate results. It has been used for most of the designs and assessments of the different performance indices presented in the next sections. As with any compensator, the speed of the compensator and its robustness (versus system variations) are also contradictory here. Faster compensators generally rely on more accurate plant models. The use of different models for prediction (transfer function) and closed-loop assessment (PLECS circuit model) is beneficial as they both model the power converter from a different point of view. However, if the discrepancy between the models (no matter if transfer function or PLECS circuit model) and the real circuit is too large, the resulting compensator may not work in practice. This is taken into account during the design of the performance index where “non-robust” compensators get penalized during the optimization process.

2.4. Performance Indices based on Real-life Design Criteria

The performance index is the most critical design criteria for the automated design procedure. It is the user’s option to interfere with the design process and to specify their application-specific needs. In this section, general considerations on performance indices are given first, followed by a detailed analysis of several difference performance indices assessed. Finally some recommendations are given based on the author’s user experience to date.
2.4.1. General considerations on performance indices

The performance index is the user’s control option during the automated design procedure. The use of real-life design criteria allows the user to easily tune this parameter without requiring deep insights into the GPC algorithm.

Several different considerations should be taken into account for the selection of these indices. One requirement of the optimization procedure is the need for a convex performance index. Without a convex index, optimization is very difficult or even impractical. Additionally, the performance index should be application-specific to allow the tuning of the compensator for a specific application, i.e. DC-DC converters. This allows the user to understand and adapt the performance index more easily.

A word on the integration of the performance indices into the design procedure should be included. The design procedure provides a generic interface to enable the use of any arbitrary performance index quantifiable in a MATLAB function. Namely, the performance index is a simple function, with the time-domain response of the output voltage for a load transient as input signal and a numeric value as performance measure as output.

2.4.2. Settling time

The settling time is the time necessary for the output voltage to successfully recover from a load transient. It is an indication for the loop bandwidth, where generally speaking a higher loop bandwidth results in a shorter settling time. In practice and with reference to figure 2.11, the setting time is commonly defined as the time until the output voltage has recovered into a predefined target zone spanned around the reference value with a defined hysteresis band. A practical implementation, e.g. as a MATLAB function for the optimization process, is relatively simple, with the settling time defined as the time difference between the load transient and the last output voltage value outside the tolerance band.
To illustrate, figure 2.12 shows a typical settling time profile as a function of \((N_y, \lambda)\) subject to a constant \(N_u = 5\). Note that the scale is inverted for illustration purposes.

The constant selection of \(N_u\) and hence the reduction of the problem from three to two design parameters allows the reduction in computational requirements during optimization without significant performance degeneration. The constant selection of \(N_u\) is a valid constraint here, as a variation of \(N_u\) results in only marginal performance changes when \(N_y\) is chosen appropriately [36]. Simulations for varying \(N_u\) confirm this. Figure 2.13 profiles the performance index over \((\lambda, N_u)\) with a constant \(N_y\). The variable \(N_u\) moves the optimal point in the profile to a different location, but still results in approximately the same performance.

Reconsidering figure 2.12, it can be seen that for large values of \(N_y\) (larger than 50 steps), the resulting setting times are relatively constant for a change in \(N_y\) and result in convex profiles. Therefore if the settling time is chosen as a real-life design parameter, the optimization problem can be reduced to an one-dimensional problem by setting \(N_u = 5\) and \(N_y = [50 \ 150]\). These values have been proven to be reasonable in the analysis of several different power converters and provide consistent performance.
Figure 2.12.: Settling time: 3D-profile mapping with $N_u = 5$.

Figure 2.13.: Settling time: 3D-profile mapping with $N_y = 50$. 
Figure 2.14: Settling time vs $\lambda$.

Figure 2.14 shows such a setting time profile for an optimization of $\lambda$ subject to constant $(N_u, N_y)$. The resulting waveform is convex and hence suitable for optimization. However from the author’s experience, the compensator chosen at the minimum settling time tends to be highly tuned and not very robust. Small discrepancies between model and real plant can render these compensators not applicable, or worse even unstable. This is caused by the fact that compensators for linear systems can be designed extremely fast as there is no limitation of the control action and perfect matching between reference and plant model. As stated in the previous section, it is advisable to use a different simulation tool, e.g. PLECS, and also to include duty cycle saturation, to allow a more thorough assessment and verification of the designed compensator.

Even with all these drawbacks, the settling time remains a very useful performance index, because of its practical relevance. Power converter design specifications generally describe the control requirements with a maximum allowed settling time. If this settling time is used in a target value optimization (instead of a minimization), it allows the transformation of the settling time into a more useful performance index. If the required settling time is selected appropriately, the compensator designed is fast and reasonably robust at the
same time. However, the considered setting time can be achieved with two
different values of $\lambda$, located at the left edge and the right edge of the “valley”
in figure 2.14. Generally speaking, the compensator designed with the higher
value of $\lambda$ will be more robust and therefore proves a better choice.

In summary, the settling time proves suitable as a performance index as long
the target value is selected in a suitable range for compensator and power
stage. As in most real-life examples, “a fast control signal will not make a slow
power stage fast.”

2.4.3. Maximum deviation

The next performance criterion considered is the maximum deviation of the
output voltage from the reference and is defined as the maximum difference
between the two. The deviation for a load step is calculated according to
(2.21), while (2.22) is used for the reference step. Two different equations
are required because a reference change will automatically create the largest
deviation when applied to the power stage.

$$V_{\text{dev, max}} = \max(\max(V_{\text{out}}(t)) - \text{V}_{\text{ref}}), |\min(V_{\text{out}}(t)) - \text{V}_{\text{ref}}|)$$  \hspace{1cm} (2.21)

$$V_{\text{dev, max}} = |\max(V_{\text{out}}(t)) - \text{V}_{\text{ref}}(t)|$$  \hspace{1cm} (2.22)

A typical maximum deviation profile as a function of $\lambda$ is shown in figure 2.15. It
can be observed that fast compensators produce smaller deviations. The
maximum deviation is clamped to a certain value (even for small $\lambda$s). In this
case, the maximum deviation is determined by the ESR and the inductance of
the power converter. This can be explained with an effect referred to as critical
inductance in the literature [45, 46]. Consequently for a given inductance,
a critical loop bandwidth can be derived where a further increase in loop
bandwidth does not lead to any reduction in maximum deviation. Note that
depending on the power converter specification, an alternative reason for the clamping can be the saturation of the duty cycle. For reference changes, e.g. start-up procedures, systems with small values of $\lambda$ tend to lead to large deviations which are mainly caused by an overshoot of the output voltage.

In summary, the maximum deviation is a suitable performance index until the point of critical inductance/bandwidth. Any improvements beyond this point, e.g. a reduction in settling time, cannot be quantified with the maximum deviation index and hence cannot be optimized.

### 2.4.4. Over-/Undershoot

The overshoot (for a light-to-heavy load transient) is the difference between the maximum occurred value of the output voltage above the reference voltage as illustrated in figure 2.11. This is not to be confused with the maximum deviation as presented in the previous section which is in this case equal to the minimum output voltage value.

![Figure 2.15: Maximum deviation.](image)

**Figure 2.15:** Maximum deviation.
Overshoots are commonly observed in overcompensated underdamped second-order systems, such as buck converters, where the input signal is changed faster than the output signal can follow. This leads to an overcompensation at the input, which is followed by an overshoot at the output.

The overshoot is of critical importance in some power converter applications and therefore has been assessed as a performance index. While a smaller overshoot is considered a better performance, the overshoot is also clamped to zero for compensators below a certain bandwidth, i.e. no overshoot occurs. Therefore, the performance of compensators below this bandwidth cannot be quantified as they all result in a performance index value of zero. On the other hand, overshoots can be tolerated in some applications and thereby enables the use of higher bandwidth compensators. It is up to the user to make a reasonable choice depending on the system requirements with the trade-off being the restriction in overshoot versus speed of response.

2.4.5. Area

The area between the signal and the reference values is considered as the next performance index. Two different ways to compute this area have been investigated. The area is computed as

\[ J = \int_0^T |V_{\text{ref}} - V_{\text{out}}(t)| \, dt = \sum (t_n - t_{n-1}) |V_{\text{ref}} - V_{\text{out},n}| \]  

and is the real physical area between output and reference signal. This requires the computation of the absolute values which can be a problem in some implementations. For this reason, the squared area has been considered and is calculated as

\[ J = \int_0^T (V_{\text{ref}} - V_{\text{out}}(t))^2 \, dt = \sum (t_n - t_{n-1})(V_{\text{ref}} - V_{\text{out},n})^2 . \]

Figure 2.16 shows typical profiles of the two different area measures as a function of \( \lambda \); each for a load and a reference step. The area is calculated
according to (2.23) and \( \text{area}^2 \) according to (2.24). In (2.24) the error signal is squared which is similar to the GPC algorithm. With this large deviations are overweighted and the computation of the absolute value is omitted which is required to avoid the cancellation of positive and negative areas. This cancellation effect is illustrated in figure 2.17 where a poor compensator performance is shown. If the absolute value of the area is not used, the cancellation of the areas above and below the reference voltage will produce rather low cost, even for this poor performance.

An advantage of using the area as a performance index is the detection of unstable systems during simulation. Note that the GPC algorithm itself can only guarantee stability for linear systems (and suitable design parameters). The non-linear simulation model (duty cycle limitations, etc.) can still result in significant ringing or even be unstable. As illustrated in figure 2.4, such an oscillating-mode will create a considerable amount of costs and consequently will not be chosen during the optimization process. To elaborate further, if the area is chosen as a performance index, it is used twice in the process. The design process uses the area (computed from the simulated closed-loop system response) as a measure of the compensator performance during the search for

![Figure 2.16: Performance index based on area.](image-url)
the optimal set of compensator design parameters. Within the GPC algorithm, the area based on the predicted output values is taken into account during the determination of the optimal compensator coefficients for given design parameters.

Note that the two areas can be significantly different, especially in the presence of duty cycle saturation. In this case, the area based on the linear predicted output voltage is significantly smaller than the area obtained from measurement results. Therefore, the area is an ideal measure to detect non-linear behaviour in the loop and has proven effective in penalising undesirable fast compensators.

With reference to figure 2.16, the area is convex and suitable for optimization. The minimum area is in the range of $\lambda \approx 3$ which is a suitable value of $\lambda$ for the chosen system based on simulation results. Note that the absolute values of the two indices are significantly different, but the minimum value in each case resides at the same value for $\lambda$. This is a rather unexpected result as a difference in the resulting compensators was expected, because the two indices weight the dip (i.e. the large voltage drop as the start of the transient) and the tail (i.e. the long, but small error compensation at the end of the load transient) differently.
2.4.6. Phase margin

In the following, the phase margin of the open-loop system (a commonly used measure of robustness) is considered as a performance index. Generally speaking, the phase margin decreases with smaller values of \( \lambda \), corresponding to faster compensators, as illustrated in figure 2.18. However, the phase margin is not convex by nature and has to be transformed into a convex function prior to its use in the optimization process.

This can be done with the selection of a target value, considered most optimal, beforehand and the minimization of a derived performance index \( J_\varphi \) instead of the actual phase margin \( \varphi \). This performance index expresses the difference between the actual phase margin \( \varphi \) and the target value \( \varphi_T \). The closer the phase margin is to its target value, the better the performance is rated. For example, this can be achieved with

\[
J_\varphi = |\varphi - \varphi_T| ,
\]

(2.25)

Figure 2.18.: Phase margin vs \( \lambda \).
where the result is shown in figure 2.19. Note that for illustration purposes, $J_\varphi$ is shifted slightly to the lower right. However, this performance index requires a preselection of the desired phase margin by the user and is therefore not favourable for an automated design process.

### 2.4.7. Combination of different performance criteria

Most of the performance indices discussed so far have been focused on only one performance criterion, e.g. speed or robustness. They all rely on the user to make a reasonable choice for the target value in order to achieve a fast but also robust system. For example in case of the settling time, if the target value is too large, a more suitable compensator design is possible. Conversely if the target value is too small, the resulting compensator can be too fast for the plant. Hence the resulting compensator may not be very robust or may even be unstable in practice.

In order to optimize a compensator for speed and robustness without user-interaction, a new performance index is now proposed which combines error area, as a measure of speed, and phase margin, as a metric of robustness. To simplify the use of the index during the optimization process, the phase margin component is remapped to a robustness index. From experience, a

![Figure 2.19: Phase margin and phase margin index vs. $\lambda$.](image)

43
phase margin of 20° is suitable as a lower limit and hence rated with a robustness value of $J_\varphi = 1$ (i.e. not very robust). An upper limit of robustness is represented by $J_\varphi = 0$ which corresponds to a phase margin of 100° but other limits may be chosen. Consequently, the overall performance index is given by

$$J_{\text{performance}} = w J_\varphi + \int |V_{\text{out}} - V_{\text{ref}}| \, dt .$$

(2.26)

The ratio between of two terms is weighted by $w$ which is the only design parameter and must be chosen in advance. From simulation and experimental tests, a recommended range for $w$ is between 0.5 and 2.0. To emphasize the influence of $w$, figure 2.20 shows a selection of closed-loop load transient responses for different compensators designed using the GPC algorithm for various weightings in the range $w = [0.5, 2.0]$. As shown in figure 2.20, a higher value of $w$ results in more robust and therefore slower compensators evident from the larger settling times. Note that different values of $w$ can return identical compensators as observed from the systems responses.

Figure 2.20.: Influence of the weighting factor $w$ on the transient response.
2.4.8. Discussion of suitable performance indices

With different performance indices analysed in the previous sections, some concluding remarks on the practical choice are now given. While the final choice remains with the user and should be based on application-specific requirements, some general considerations are now given:

Some performance indices, such as settling time or phase margin, are based on one criterion and require a preselected target value for optimization. Its value is highly application-specific and it is the user’s task to make an informed choice. Because only one criterion is optimized, other design criteria are not taken into account. For example in the case of the settling time, it remains the user’s responsibility to judge, how fast a compensator can be and still provide enough robustness. On the other hand, a performance index based on one criterion enables the user to fine-tune the compensator more easily by adjusting the target value, as the relationship between the performance index, i.e. its target value, and the designed compensator is clearly defined.

More complex performance indices, such as the combined area robustness index, ensure that robustness and speed are considered concurrently. They enable the implementation of a fully automated design process without any user interaction. If required, the user can still tweak the resulting compensator within a given range by adjusting the weighting ratio. However, this does not necessarily result in a different compensator, as previously outlined.

2.5. Solution to the Optimization Problem

The optimization problem arising from the proposed automated design procedure can be considered as an N-dimensional ($N = [1\ 3]$) optimization problem. Several different solvers have been presented in the literature [47–49] and implemented on different platforms. The focus of this work is not on the development of such solvers; rather on their application.
Two properties of the performance indices should be taken into account when choosing a solver. Firstly, some solvers require the derivative of the cost-function, i.e. the chosen performance index, which cannot be derived for the performance indices presented. Secondly, some solvers are not able to handle discrete optimization parameters properly. This may be an issue as two of the three optimization parameters, namely $N_y$ and $N_u$, are discrete values (integers). The choice of the solver is also influenced by the implementation platform. Platforms such as C++ and Java have been chosen for existing digital power GUIs.

When choosing a solver, additional consideration should be given to the accuracy of the results and the computational effort. Most plots in this thesis have been computed with a simple, fixed-resolution grid-search, rather than a solver. This allows the profiling of the entire cost-function and the determination of the optimal point for a chosen resolution. A solver, on the other hand, allows a much faster discovery of the optimal point as only the computation of a subset of data points is required. This is the preferred choice for productive use of the algorithm. However, it must be ensured that the optimal point is reached. The accuracy of the results is also dependent on the resolution of the solver, i.e. the smallest parameter step size allowed. The trade-off between accuracy and resolution remains with the user.

Experimental testing with different solvers provided by MATLAB prove their applicability to the given optimization problem. The results obtained with a grid-search have been compared with the results obtained by existing solvers. In the next section, two case studies using two different performance indices and different solvers are analysed in full detail for the same prototype system.

### 2.6. Practical Design Example

In this section, two practical design examples using the automated design process are presented. The results obtained using selected solvers are analysed and compared with the results from a grid-search. Simulation results for the
derived compensators are compared with the response of a hardware prototype. As an example, a single-phase buck converter is considered which has been implemented on the experimental prototype platform detailed later in this thesis (subsection 3.5.1). Its design parameters are specified in table 2.2.

As a first step in the design procedure, the power converter’s component values have to be converted into a prediction model. To do so, equation (2.18) is discretized numerically with the switching frequency. An additional unit-step delay is added into the transfer function which represents loop delays such as sample-and-hold and conversion time of the ADC, the computation time of the compensator and the DPWM sample time. A MATLAB script is used to convert the prototype parameters into the transfer function model. Using the component values listed in table 2.2, this results in a transfer function of

\[
G_{V,dc} = \frac{V(z)}{d(z)} = \frac{0.04827z + 0.03934}{z^3 - 1.974z^2 + 0.9883z} 
\]

with a sampling time of \( T_s = 2 \mu s \).

In the next step, an appropriate performance index has to be chosen. The following two sections detail two difference scenarios. A simple approach using the settling time as performance index is compared with a combined performance index both computed for a typical 20 A load step.

<table>
<thead>
<tr>
<th>Table 2.2.: Technical details of the prototype system.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of phases</td>
</tr>
<tr>
<td>Input voltage</td>
</tr>
<tr>
<td>Output voltage</td>
</tr>
<tr>
<td>Switching frequency</td>
</tr>
<tr>
<td>Inductance</td>
</tr>
<tr>
<td>Loop impedance</td>
</tr>
<tr>
<td>Total output capacitance</td>
</tr>
<tr>
<td>Output capacitor’s ESR</td>
</tr>
<tr>
<td>ADC reference value</td>
</tr>
</tbody>
</table>
2.6.1. Case 1: Using the settling time

In the first case, the settling time is chosen as a performance index. It has been established during this thesis that the GPC design parameter $N_u$ does not influence the resulting control system significantly when chosen appropriately, e.g. $N_u = 5$. This is illustrated in figure 2.21a and figure 2.21b where the same $(\lambda, N_y)$ search space is shown for two different values of $N_u$, i.e. $N_u = 3$ and $N_u = 5$. The resulting plots are only marginally different. An optimization over an additional third dimension would increase the computation effort excessively without gaining significant advantages. This is further supported by the fact that generally speaking the same performance can be obtained for different values of $N_u$ by selecting a different optimal point, i.e. selecting a (slightly) different $(\lambda, N_y)$.

With two optimization variables remaining, a further analysis of the settling time profile has been performed. Figure 2.21 b shows the settling time as a function of the GPC design parameters $(\lambda, N_y)$ with constant $N_u = 5$. As previously outlined, the prediction horizon $N_y$ should be long enough to cover the system dynamics and to ensure stability. Also the settling time profile varies only marginally for large values of $N_y$. Therefore if $N_y$ is set to a constant value, an optimization over only one parameter, i.e. $\lambda$, is required. Figure 2.22 shows such a profile for $N_y = 100$ in which an automated solver has to search for the optimal settling time.

In an automated design process, a predefined solver is used to minimize a given function (in this case the profile shown in figure 2.22). This can be achieved easily using the optimization toolbox of the MATLAB suite. In order to optimize towards a target value, the function is extended to accommodate for a predefined target value. As discussed in subsection 2.4.2 and evident from figure 2.22, the settling time requirements can be met with two different values of $\lambda$. To ensure the selection of the preferred higher value, the input parameter of the optimization function is remapped, so that the solver approaches the valley from the right.
In this example, the solver \texttt{fminsearch} has been used. This is a generic optimizer which finds the “minimum of unconstrained multi-variable function using a derivative-free method;” in this case a variant of the well-known simplex search algorithm [50, 51].

Note that there is no need during the automated design procedure to compute the entire profile shown. The use of a solver reduces the amount of computational effort significantly. This is evident from the values listed in table 2.3, where the required number of computations and the computation time for this example are compared. It is expected that an implementation in another programming language such as C++ or Java will reduce the computation time significantly. The algorithm implemented in MATLAB has not been fully optimized for computational speed.

Note also that the figures listed in table 2.3 are highly dependent on the resolution of the simulation, i.e. the grid size, where the figures listed correspond to the plots shown in this section. A coarser or finer resolution for the 2D-, 3D-graphs can be also chosen. The optimization time required by the solver is comparable with the time demands of existing solutions which can take approximately one minute to optimize a compensator [29].

The optimization routine returns the point \((\lambda = 12.05, N_y = 100, N_u = 5)\) as the optimal selection for a chosen settling time of \(T_{sett} = 30\) steps. This corresponds to the following compensator transfer function
\[
G(z) = \frac{N(z)}{D(z)} = \frac{13.949 - 24.887z^{-1} + 11.138z^{-2}}{1 - 0.21683z^{-1} - 0.38214z^{-2} - 0.40103z^{-3}}. \tag{2.28}
\]

<table>
<thead>
<tr>
<th>Strategy</th>
<th>Simulation Model</th>
<th># of computations</th>
<th>Computation time</th>
</tr>
</thead>
<tbody>
<tr>
<td>3D grid</td>
<td>TF</td>
<td>50*45=2250</td>
<td>18 min</td>
</tr>
<tr>
<td>3D grid</td>
<td>PLECS</td>
<td>50*45=2250</td>
<td>92 min</td>
</tr>
<tr>
<td>2D profile</td>
<td>TF</td>
<td>50</td>
<td>38 s</td>
</tr>
<tr>
<td>2D profile</td>
<td>PLECS</td>
<td>50</td>
<td>106 s</td>
</tr>
<tr>
<td>\texttt{fminsearch}</td>
<td>TF</td>
<td>19</td>
<td>24 s</td>
</tr>
<tr>
<td>\texttt{fminsearch}</td>
<td>PLECS</td>
<td>19</td>
<td>42 s</td>
</tr>
</tbody>
</table>
Figure 2.21.: Design example 1: 3D-profile of the settling time.
The compensator is of a third-order type with a pole at $z = 1$ as an integrator. Note that the order of the compensator derived with the GPC algorithm is dependent on the order of the reference plant. For a second-order transfer function, e.g. a DC-DC converter, the compensator returned is always of third order.

The respective open-loop bode plot is shown in figure 2.23 with the corresponding load transient waveform shown in figure 2.24. The settling time in the transient simulation matches the design criterion of 30 switching cycles (steps). Note that in this example, the settling time value is defined by the tail instead of the actual load transient. Hence, the angle of intersection between the output voltage and the threshold value is small. A larger angle is preferred as it allows for a better judgement of the actual load transient (rather than the tail). This can be achieved with an increase of the threshold voltage used during the determination of the settling time.

The designed compensator is tested on a developed hardware prototype (specification listed in table 2.2). Figure 2.25 shows the measured transient waveforms for a 20A load step. In figure 2.25 a, the load step response for a system with
Figure 2.23.: Design example 1: Open-loop bode plot.

Figure 2.24.: Design example 1: Load transient simulation.
a compensator derived using the automated design procedure is shown. The transient fulfils approximately the specified settling time requirement while the shape of the waveform is significantly different to the simulation results shown in figure 2.24. This is caused by the discrepancies between model and hardware. It appears from the waveform that the actual output/loop impedance of the hardware is higher than the impedance assumed in the model. This dampens the response, i.e. reduces the overshoot, and causes a larger steady-state offset after the correction of the initial transient. In an optional iterative process, the model can now be adjusted (model matching) and the design procedure repeated.

In figure 2.25b, the transient response is shown for a compensator where the DC gain has been slightly adjusted. This enables a better correction of the “slow” error after the immediate transient and results in an overall better performance.

### 2.6.2. Case 2: Using a combined index

For the second design case, featuring the same prototype system as design case 1, the combined performance index presented in subsection 2.4.7 is used in a two-dimensional optimization algorithm. An optimization algorithm developed around standard solvers provided by MATLAB is detailed.

With reference to (2.26), the performance index consists of two separate contributions, i.e. area and robustness; both illustrated in figure 2.26. The robustness index based on the phase margin is shown in figure 2.26a with the area index for a load transient shown in figure 2.26b. This combined performance index, illustrated in figure 2.27, is used to optimize the compensator design. A weighting ratio value of $w = 0.75$ has been chosen.

Note that the resulting optimization problem is now two-dimensional. A wide range of optimization strategies to solve such problems have been investigated in the literature [48, 49, 51–53]. However, most solvers provided by existing toolboxes, such as MATLAB, prove not very effective for the stated problem.
Figure 2.25: Design example 1: Measured load transient performance.
Figure 2.26.: Design example 2: Performance indices based on one criterion.
Figure 2.27.: Design example 2: Combined performance index.
This is due to the fact that most of these solvers do not provide sufficient control over the parameter optimized within, i.e. the search-direction. For example, a solver might only tune one parameter very effectively while it changes the second parameter only marginally. Subsequently, this can lead to suboptimal solutions.

For the chosen performance index, a simple modification of existing solvers has proven very effective: In a first step, the optimal weighting factor $\lambda$ is computed for a constant, large prediction horizon $N_y$. Then $\lambda$ is kept constant and $N_y$ optimized. This procedure is repeated depending on the required resolution. In practice this technique has proven more efficient as a direct two-dimensional optimization and typically returns the optimal point after two to three iterations. During the course of this research, the results obtained with the use of the modified optimization technique have been cross-checked with the results obtained by a grid-search to ensure the accuracy of the method.

For the prototype considered, the optimization routine returns the optimal point at $(\lambda = 64.70, N_y = 54, N_u = 5)$ which has been confirmed as the global minimum with a grid search over the profile (figure 2.27). The corresponding compensator has a transfer function of

$$G(z) = \frac{N(z)}{D(z)} = \frac{9.189 - 16.81z^{-1} + 7.678z^{-2}}{1 - 0.4476z^{-1} - 0.2760z^{-2} - 0.2764z^{-3}}.$$  \hspace{1cm} (2.29)$$

The respective open-loop bode plot and the time-domain response to a load transient are shown in figure 2.28 and figure 2.29.

For verification purposes, the compensator design has been tested on the prototype system. The response to a 20 A load step is shown in figure 2.30. The compensator provides good performance, meeting the requirements of the design procedure. Because the same model and hardware as in design case 1 has been used, the same observations can be made. The impedance of the power stage is larger than expected, resulting in a significant error immediately after the load step (figure 2.30a). The transient performance after a small adjustment of the DC gain is shown in figure 2.30b. It is superior to the one obtained with a standard PID compensator which is due to the higher order of the compensator. The automatically derived compensator provides good loop performance and an ideal starting point for any manual tuning if required.
Figure 2.28.: Design example 2: Open-loop bode plot.

Figure 2.29.: Design example 2: Load transient simulation.
Figure 2.30.: Design example 2: Measured step performance.
2.7. Discussion and Conclusions

Generalized Predictive Control (GPC) offers real benefits in DC-DC conversion, because of its clearly defined design process, time-domain performance criteria, simple tuning technique and guarantee of stability. An automated-design procedure utilising this algorithm is proposed which enables the automated design of compensators using real-live performance indices. With the choice of a suitable performance index and the selection of a few GPC configuration parameters based on this index, the method presented can generate a stable and highly optimized compensator for any given plant model. Several different real-life performance indices have been analysed and their performance and suitability assessed.

The search for the optimal GPC parameter set has been considerably simplified with the use of dimensional maps profiling the relationship between the parameters. Different optimization strategies have been investigated to reduce the required computation time. It is shown that this method can be further developed for practical designs with the introduction of a robustness metric.

Two different design cases employing different performance indices have been presented. In the first case, the settling time has been chosen as a design parameter to derive a fast, but less robust compensator. In the second design case, an optimal and robust compensator was designed using a combined performance index rating speed and robustness. Both cases have been assessed in simulation and verified on a prototype demonstrator system, i.e. a buck converter. In theory, the automated design method is applicable to other DC-DC converter types, such as a boost or a buck-boost converter. However, this has not been investigated during the course of this work and may be subject to further investigation.
3. Automatic Multi-phase Digital Pulse Width Modulator

Today’s demands for high output power and fast transient performance require the use of multi-phase power converters. Converters of this type comprise of multiple parallel power stages (phases) and a central digital controller which is used to control all phases individually, but as a single power converter.

Digital pulse width modulators (DPWMs) are employed as an interface between the digital world of the control IC and the analogue world of the power stages. Commonly, in multi-phase applications one (single-phase) modulator is required for each controlled phase. While this has been sufficient in the past, it has become more and more a limiting factor due to an increasing number of phases, increasing control performance and the introduction of more advanced modulation requirements, such as phase shedding. To overcome these limitations, new modulator architectures need to be investigated.

In this chapter, a novel automatic multi-phase digital pulse width modulator (AM-DPWM) is proposed which utilizes an internal, intelligent duty cycle distribution scheme. This allows the fastest possible execution of the duty cycle input command with respect to the number of switching actions per cycle and phase. The combination of easy-to-use phase shedding and improved transient performance makes this modulator highly favourable for practical applications.

The chapter is organized as follows: In the first section, the limitations of existing modulators are reviewed and their internal designs are analysed. In the next section, a novel AM-DPWM is proposed and its mathematical background derived. This is followed by implementation details. Experimental assessment on an FPGA prototype and a discussion conclude the chapter.
3.1. Introduction

With SMPSs moving to higher output power and smaller output voltages, the need for multi-phase power converters is continuously growing. With an increasing number of phases, the benefits of a control loop bandwidth closer to or even higher than the switching frequency of the individual phases are well-acknowledged. As simple DPWMs are still used in current multi-phase designs, these tend to be one of the limiting factors of the loop bandwidth, and hence the need for improved “smart” DPWMs is clearly evident. Additionally, most of the existing architectures are not able to drive a varying number of phases which is required in systems with phase shedding operation, e.g. [22, 54].

Standard multi-phase power converters (figure 3.1), comprising of $N$ phases, use $N$ conventional DPWM modulators to generate the control signals for the power stage switches [55–58]. Synchronization between the individual modulators ensures proper phase shift. Naturally sampled DPWMs update the duty cycle once per switching cycle, which allows the update of the inputs of the modulator up to $N$-times the phase switching frequency. However, each

![Digital Control Core](image)

**Figure 3.1.:** Standard DPWM modulation for a four-phase buck converter.
individual DPWM is only updated once per cycle leading to additional problems such as current mismatch during load transients. Most designs overcome these issues by limiting the loop bandwidth, and thereby compromising the system performance.

For single phase applications, this issue has been addressed using several different concepts, such as charged-balanced control [6, 7], linear–non-linear control [10, 11] and multi-sampled DPWMs [16–18, 20]. However to date, most of these concepts have not been applied to multi-phase applications. As a consequence, multi-phase converters are still driven by standard multi-phase modulators.

Additionally, multi-phase DPWMs comprising of conventional single-phase DPWMs require synchronization between the individual phases in order to achieve an optimal phase shift between the individual power stages. This is mandatory to gain full advantage of the parallelization of multiple phases. However, the phase shift is subject to the number of phases. Hence, when the number of phases changes during run-time (phase shedding), a resynchronization of the DPWMs is required which leads to additional hardware overhead.

For multi-phase converters, special scalable solutions have been presented in the literature [59–62], which typically share area consuming hardware resources across the phases. To do so, the design detailed in [59, 60] duplicates the same duty cycle value for all phases and therefore restricts the implementation of current sharing techniques.

In [61, 62], an approach based on a digital analogue converter (DAC) architecture is presented. The duty cycle input command is considered as an input of a DAC representing the delivered output power, which is then distributed over the individual phases. This scheme accommodates the update of the duty cycle with frequencies greater than the actual switching frequency and scales favourably with the number of phases. However, it does not support phase shedding and does not respect the number of switching actions per switching cycle and phase. This can lead to a undesired increase in effective switching frequency.
In the following, a new multi-phase DPWM scheme is proposed addressing the issues highlighted. It provides good hardware utilization, limits the number of switching actions per phase and cycle and supports phase shedding. The system design level is presented, followed by the proposal of the new duty cycle distribution scheme. This is followed by its implementation and verification on an FPGA.

### 3.2. System Overview

The proposed system (figure 3.2) consists of an $N$-phase power stage, an ADC, a digital loop compensator (Loop Comp), current sharing logic, and a “smart” DPWM incorporating a new duty cycle distribution algorithm. This algorithm is based on the fastest possible execution of the duty cycle commands while still ensuring that each phase switches only twice (on/off) per switching cycle. A dynamic number of active phases (phase shedding) is incorporated by design together with an optional sigma-delta functionality to improve the effective resolution.

![AM-DPWM with a four-phase buck converter](image)

**Figure 3.2.** AM-DPWM with a four-phase buck converter.
3.3. Distribution Principle

Before detailing the new modulation scheme, it is first necessary to understand the modulation output of conventional multi-phase DPWMs. As an example, a typical modulation output of a four-phase system is shown in figure 3.3. For each subcycle (latin numbering), the duty cycle (shown below the waveform) is applied to the currently active DPWM. While the system restricts the number of switching operations per cycle intrinsically, the delay in the application of the duty cycle and the resulting distribution of the output signal are not optimal. For the example shown, the phases zero ($S_0$) and three ($S_3$) take most of the transient current leading to a large current mismatch immediately after the transient. Also the delay between the reception of the duty cycle command and the application to the power stage can be up to one full DPWM cycle.

3.3.1. New distribution principle

The proposed scheme distributes the duty cycle over the phases, respecting the following criteria:

- Each phase is exactly allowed to switch twice per cycle (on once, off once).
- Only the next phase in the cycle is turned on.
- If a phase is still on at the end of a subcycle, it can be kept on for “longer”.
- Phase shedding and phase alignment are respected.

Adhering to these criteria leads to a distribution of the duty cycle as shown in figure 3.4, where the standard duty cycle is shaded for comparison purposes. As before, the duty cycle input command for each subcycle is listed below the waveform. The arrows illustrate the redistribution of the duty cycles among the phases compared with a standard modulation scheme. For illustration purposes only the first three redistribution steps (2, 3, 0) are highlighted with the respective subcycle number encircled. To further illustrate the timing, a step-by-step version of the drawing is shown in figure 3.5.
Mathematically, the duty cycle $D_{n,k}$ for each phase $n$ and time instant $k$ can be calculated from the duty cycle input $D_k$ as

$$D_{n,k} = \begin{cases} 0, & \text{if } D_{n,k-1} \neq \frac{1}{N} \text{ and } i \neq 0, \\ \min\left(\frac{1}{N}, \max(0, D_k - \frac{i}{N})\right), & \text{otherwise,} \end{cases}$$

(3.1)

where $i = (k + n) \mod N$.

This mathematical description is suitable for modelling at a high level of abstraction (e.g. using MATLAB) where complex mathematical operations create no computational burden. A further reduction of computational complexity is now described leading to a much simpler mathematical description of the problem suitable for hardware implementation.

---

**Figure 3.3.** Standard DPWM modulation scheme.

**Figure 3.4.** Duty cycle distribution with the new modulation scheme: summary.
Figure 3.5.: Duty cycle distribution with the new modulation scheme: detailed step-by-step guide.
When the distribution principle is analysed, it reveals the following characteristics:

- The duty cycle is distributed over the phases with the highest priority given to the phase turned on last and with the priority dropping with increasing “turn-on” time.
- If more than one phase is on at the same time, the maximum of one phase requires a modulation of the output signal within the current subcycle.

The duty cycle $D_k$ of the interval $k$ can be converted into the priority-encoded signals using

$$P_{n,k} = \begin{cases} 
0 & \text{if } P_{n,k-1} \neq \frac{1}{N} \\
\min\left(\frac{1}{N}, \max(0, D_k - \frac{n}{N})\right) & \text{otherwise}
\end{cases}$$

(3.2)

where the priority is represented by $n = \left\lfloor 0 \quad N - 1 \right\rfloor$ with the highest priority given to $n = 0$.

This equation can be further simplified by analysing a second property of the switching signal. It can be observed that the modulator is allowed to

- turn exactly one phase on at the beginning of the subcycle,
- turn exactly one phase off during the subcycle and
- turn off as many phases as required at the end of the subcycle.

This transforms the description of the problem to a search for the optimal number of active phases during each subcycle. In order to do so, the system assesses the situation at the beginning of each subcycle, decides how many phases need to be active based on the previous number of active phases and the currently required number of active phases extracted from the duty cycle input command.

To ease the calculation of the optimal number of phases, the problem is split into two subproblems, similar to [62]. The number of required “fully turned on” subcycles is determined and the remaining duty cycle is used to calculate
the modulation of the phases turned off during the cycle. This is done by a
simple hardware implementation of the following equations

\[
D_k = D_{in,k} + R_{k-1} \tag{3.3}
\]

\[
F_k = F_{k-1} - G_{k-1} + D_k \div \frac{1}{N} \tag{3.4}
\]

\[
L_k = D_k \mod \frac{1}{N} \tag{3.5}
\]

\[
R_k = L_k - M_k \tag{3.6}
\]

where \(D_{k,in}\) is the current duty cycle input command, \(R_{k-1}\) the residue of
the previous subcycle, \(F_k\), \(F_{k-1}\) the number of required subcycles of the
current/previous interval, \(G_{k-1}\) the applied subcycles during the last interval,
\(L_k\) the subcycle modulation and \(M_k\) the applied subcycle modulation during
the last subcycle.

Although these equations look very complex, their actual implementation (see
section 3.4) is straightforward. If scaling to the power of two is selected and
\(N\) fixed, the modulo operation is equivalent to a truncation of bits. As \(N\) is
not a fixed value here, rather it is a value from a fixed selection, a simple
truncation is not sufficient and a real implementation of the modulo operation
is required. However, this can be implemented as a serial operation, using
a simple recursive subtracter unit, where the number of required cycles is
defined by the maximum ratio between the dividend, iDC, and the divisor,
\(\frac{1}{N}\).

The decision regarding the number of active phases is now dependent on the
inputs \(F_k\), \(L_k\) and on the current number of active phases \(n_{a,k-1}\) at the end
of the previous subcycle. An analysis of the system results in the following
equation used to calculate the number of active phases \(n_{AP,k}\) at the beginning
of each subcycle \(k\).

\[
n_{AP,k} = \begin{cases} 
\min(N - 1, n_{AP,k-1} + 1), & \text{if } n_{AP} \leq F_k \\
F_k, & \text{otherwise.}
\end{cases} \tag{3.7}
\]

If the required number of full subcycles \(F_k\) is fully applied and if a modulation
of the remaining signal is required (which is the case if \(F_k < n_{AP,k}\)), the number
of active phases is reduced by one during the subcycle.
3.3.2. Phase shedding

The support of phase shedding, i.e. the dynamic change in the number of available phases during operation, is a crucial feature for the practical use of multi-phase modulators. Individual phases are commonly shed to improve the efficiency of the overall system during medium and light load conditions [22, 54].

In the presented scheme, several properties change with the number of phases: the maximum number of available phases $N$, the duration of each subcycle $\frac{1}{N}$ and the phase shift between the individual switching signals $\varphi = \frac{1}{N}$. If the underlying equations are implemented in such a way that $N$ (resp. $\frac{1}{N}$) is treated as a parameter, no additional measures are required. Depending on the current operation condition, the number of active phases $N$ is selected from the interval $[1, N_{\text{max}}]$ by an external configuration signal.

3.3.3. Alternative modulation schemes

Along with the presented modulation schemes, alternative modulation schemes are also possible. One such scheme is based on the following consideration: “Is it possible to turn on more than one phase at the start of a subcycle while still limiting the number of switching actions?” The resulting conditions are visualized in figure 3.6 where the switching signals for this case are analysed. In order to turn on more than one phase per subcycle, the system has to ensure that sufficient duty cycle needs to be applied so that the particular phase does not need to be switched off prior to its regular switching edge.

Figure 3.6 illustrates the switching actions for different scenarios each defined by the two parameters. Like in figure 3.4, the rows represent the different phases, while the columns detail the different subcycles. Unlike in figure 3.4, the blocks are not consecutive in time, rather represent different independent scenarios. The first number represents the number of phases fully on at the end of the previous subcycle, while the second number represents the number of
full subcycles of the applied duty cycle in the current subcycle. The subcycles
turned on immediately by the alternative modulation scheme are marked in
green, whereas the cycles required to keep the particular phases on “long
enough” to respect the number of switching actions are shown in blue.

A clear performance increase is visible for the cases, where only a small
number of phases (e.g. one) is active at the beginning of a subcycle. Some
minor advantages can be expected in the cases where one additional phase
can be turned on, which is the case when the number of required subcycles $F_k$
exceeds the number of currently active phases by more than two, i.e. cases
(0,2) and (1,3). These cases can be covered with a minor extension of the
system which is detailed in section 3.4.

Cases where more than two phases can be turned on at the same time
occur very rarely and are subject to tight restrictions on the duty cycle
command. Therefore the additional hardware required does not justify the
small improvement in performance.
Figure 3.6.: Alternative modulation scheme where more than one phase can be switched on at the start of each subcycle.
3.4. Implementation

The overall implementation of the proposed DPWM is based on the block diagram shown in figure 3.7. The system comprises of an input logic, one finite state machine (FSM) for each controlled phase, a cyclic-counter unit and a shared high-resolution DPWM module.

The input logic normalizes the duty cycle input command so that the following logic blocks are independent of the current number of phases and can be implemented more efficiently. Along with a serial integer division and a modulo operation used for the normalization, internal integral action ensures the proper application of the entire incoming duty cycle signal.

One standard high-resolution DPWM block provides the synchronization signal for the subcycle timing depending on the number of available phases and the inter-subcycle modulation for one phase if required. The cyclic-counter provides information about the current subcycle for the individual FSMs.

Figure 3.7.: DPWM implementation diagram.
FSMs control the output modulation of the phases where one FSM is required per phase. Their implementation is relatively simple and independent of the number of available phases as this is handled by the input logic.

With reference to figure 3.4 and figure 3.8, each phase can be in one of three possible states:

- **Off**: The phase is switched off.
- **On**: The phase is switched on for the entire subcycle.
- **Mod**: The phase is modulated during the subcycle.

At the start of each subcycle, each individual FSM evaluates the current situation. Dependent on the current operation conditions, the output of the respective phase is set. It is either turned on, turned off or modulated via the auxiliary high-resolution DPWM module. This is done by a comparison of the number of currently required phases, \( n_{FC} \), with a cyclic counter value, \( CAP \), which allows a hardware effective implementation of (3.1).

![Phase state machine diagram](image-url)

**Figure 3.8.**: Phase state machine.
One additional advantage of the proposed architecture is the scalability with the number of phases. Parallel single-phase DPWMs with common resynchronization logic scale approximately linearly with the number of phases. However, in order to achieve sufficient resolution, most of today’s designs use a hybrid architecture comprising of analogue and digital components. Generally, this increases the required area and usually involves manual layout. The proposed architecture requires only digital blocks, despite one high-resolution DPWM module, and hence is fully synthesizable. The required initial area is larger compared to standard DPWMs due to the auxiliary blocks, but the advantage gained over standard DPWMs increases with the size of the high-resolution DPWM (table 3.1).

3.5. Performance assessment

In order to test the performance of the proposed DPWM in closed-loop operation, some additional digital blocks are required to build a digital control loop (figure 3.2), namely a loop compensator and, due to the nature of multi-phase power converters, current sharing functionality.

In the next subsection, a common prototype system is introduced which is used as a framework to implement and test the digital control designs presented in this thesis. This is followed by the requirements and implementation details for the application-specific blocks, in this case a loop compensator and a current sharing logic.

Table 3.1.: Area vs. number of phases.

<table>
<thead>
<tr>
<th># of phases</th>
<th>3</th>
<th>4</th>
<th>6</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>AM-DPWM</td>
<td>204</td>
<td>242</td>
<td>336</td>
<td>433</td>
</tr>
<tr>
<td>Standard DPWM</td>
<td>155</td>
<td>240</td>
<td>379</td>
<td>503</td>
</tr>
</tbody>
</table>
3.5.1. Prototype framework system

In order to test and debug the digital designs proposed in this thesis, an experimental hardware platform has been developed, built and used. The experimental hardware setup comprises of a PCB containing the analogue system circuitry and an FPGA evaluation board used to implement the digital design. Excerpts from the full schematic and the PCB design can be found in appendix B. A picture of the prototype setup is shown in figure 3.9.

As power stage, a four-phase buck converter design has been chosen. It provides flexibility in terms of system configuration, component selection (e.g. inductor values) and output power. Each phase features a dedicated DC resistance (DCR) current measurement circuit including a dedicated ADC. The digital outputs of two phases are multiplexed to reduced the number of required IO-ports towards the FPGA.

Input and output voltage are measured using high-speed analogue amplifier and ADCs to allow a wide application range of the prototype board, e.g. oversampled applications and emulation of fast ADCs. Lower conversion rates or larger latency times can easily be emulated using different clocking schemes or digital delay lines within the FPGA.

Figure 3.9.: Prototype system.
The power board incorporates two different types of loads: A fast transient load is included directly on the board and controlled via the FPGA. Connectors enable the use of DC loads which are used to sink static currents.

An Altera DE2 evaluation board enables the use of an Altera Cyclone II FPGA to implement the proposed digital blocks. A number of digital blocks have been developed and assessed during the course of this work to enable the implementation and debugging of the digital designs. The range spans from simple mathematical blocks such as special adders or multipliers to complex data paths and special-purpose state machines.

For example, a serial communication interface has been developed which allows the access to hardware registers during run-time. This enables the configuration of parameters, such as compensator coefficients, without requiring a resynthesis of the design. Debug and status information can be read back to a PC to allow for simpler debugging of the design. A multiplexer enables access to most of the signals within the digital design to harvest debug information. This can be done in real-time via a 8-bit debug signal using a mixed-signal oscilloscope or a logic analyser. Alternatively, the information can be stored in a dedicated on-chip memory to allow offline analysis. To do so, the data is written continuously into a ring buffer. When a trigger signal is given, the system stops after recording a predefined number of data sets. A similar principle is used in most digital oscilloscopes today and enables capturing of data prior to the actual trigger event. Access to the memory is given via the serial communication interface using a software program. Recording of precise digital data is very valuable during debugging as it allows further simulation of the system using the recorded data as an input vector set.

Another interesting functional block in the digital framework is the load control unit. It allows the control of the on-board transient load so that the digital design can be thoroughly tested. The generated load step is synchronized with the power switch control signals where the phase shift between the two signals can be configured with a register, so that a load step can be generated at any point in a switching cycle. This allows the sweeping of the load step position within the switching cycle to analyse its influence on the transient
performance. Once the worst-case point is determined, the time instant can be fixed and therefore enables predictable and repeatable experimental conditions which are important to analyse influences such as noise.

### 3.5.2. Loop compensation

Loop compensation techniques for power converters, both in the analogue and the digital domain, are well-developed. A lot of research has been focused on several different techniques. For the AM-DPWM system presented, one specific requirement has to be taken into account, i.e. the dynamic change in the number of phases. The compensator must be able to control all plants arising from a varying number of phases which results in a modification of the power converter’s transfer function. This is caused firstly by a change of the equivalent circuit model and secondly by a change in sampling time.

For demonstration purposes, a PID-based compensator designed using standard design techniques has proven sufficient. This compensator is retimed for different sampling frequencies by changing its clock frequency without a change in its coefficients. This can be viewed as a modification of the compensator in the time/frequency-domain due to a change of its discretization time/frequency. Note that the compensator remains constant in the z-domain as the coefficients are not altered. Generally speaking, a change in the sampling frequency towards lower frequencies “slows” down the compensator and moves the corner frequency to the left. Bode plots of the resulting open-loop systems are shown in figure 3.10.

Note that the described technique does not provide the best possible loop compensation, but proves sufficient for testing the proposed modulator. It is expected that more advanced compensation schemes, e.g. adaptive control techniques, will provide better performance.
Figure 3.10.: Bode plots of the open-loop for different sampling times.

3.5.3. Current sharing technique

In multi-phase power converters current sharing is a mandatory need due to tolerances and mismatches between the individual phases. Different strategies have been investigated in the past and can be broadly categorized into two groups, i.e. active and passive techniques. Active techniques require current sensing of the individual inductor currents and distribute the current equally among the phases. Passive techniques do not require current sensing and use alternative measures to estimate the phase current. The technique employed is based on the concept detailed in [63, 64] where current sharing as a result of duty cycle matching is proposed. This balances the current among the phases based on loss minimization instead of current equalization.

To ensure duty cycle matching among the phases, several different techniques are available. However, pre-existing techniques introduce additional delays into the control loop and hence degrade its performance. Instead, a technique originally proposed for oversampled, single-phase power converters [18] is adapted to achieve passive current sharing without phase delay. A comb-filter is inserted into the system loop which rejects signals (harmonics) at multiples
of the switching frequency without delaying the control loop. The bode plot of the filter’s transfer function is shown in figure 3.11. One advantage of the proposed method is the relatively good decoupling of the design procedures of the current sharing filter and the loop compensator.

Additionally, the proposed comb-filter (figure 3.12) is simple to implement and does not require any dedicated multipliers. Adders and shifters are sufficient in practice. Like the loop compensator, the comb-filter is retimed when the number of phases is changed. Unlike the loop compensator, the comb-filter requires some additional action during retiming.

When the filter is retimed without any modifications, its rejection frequencies are shifted as they are relative to the sampling frequency, e.g. $\frac{1}{4}F_s$ and $\frac{1}{2}F_s$. However in order to ensure proper current distribution, the rejection bands need to match the switching frequency and its harmonics. While four, two and one phase operation can be covered by one filter for harmonics at $\frac{1}{4}F_s$ and $\frac{1}{2}F_s$, different rejection bands are required for three phase operation. For this case, the harmonics are at $\frac{1}{3}F_s$ and $\frac{2}{3}F_s$. Subsequently, a small modification of the comb-filter from fourth-order into a third-order system is introduced to

![Figure 3.11: Bode plot for control loop and additional filter.](image)

---

80
move the (relative) rejection bands into the correct positions. With reference to figure 3.12, this is achieved with five multiplexers and one additional gain stage ($\frac{1}{3}$). When the multiplexer control signal, $T$, is set to one, the filter is in third-order mode, otherwise it is in fourth-order mode.

### 3.5.4. Performance assessment

To verify the proposed concepts, the prototype framework presented in subsection 3.5.1 is employed in a four-phase buck converter configuration with 500 kHz switching frequency per phase. Technical details of the power converters are given in table 3.2.

In figure 3.13 and figure 3.14, the system’s voltage response to a 70 A load transient is shown. Figure 3.15 shows the response for a system using a standard DPWM for comparison. In figure 3.13 and figure 3.15, the respective output
Voltage is shown (blue), together with two of the four phase currents (yellow and magenta). Note that the large noise spikes are due to pure HF-coupling of the measurement probe. In figure 3.14, the respective digital control signals (DPWM outputs) are shown, together with the output voltage as a reference signal. Also the outputs of the standard DPWM (\(D_{12}\) to \(D_{15}\)) and the new modulator (\(D_8\) to \(D_{11}\)) are both shown in the same diagram. Note that only the latter set is used to control the switches.

Using the new distribution algorithm, the transient response is improved by almost 40\% compared to regular DPWMs. The maximum deviation is significantly reduced which is also due to an increase in the maximum loop gain. In this prototype, the gain of the standard DPWM loop is reduced by a factor of two compared with the new modulator, as a higher gain would cause a significant output voltage overshoot.

In figure 3.16, the behaviour of the DPWM during a phase shedding operation is shown. The number of phases is changed from three to four via an external control signal (\(D_7\)). The respective phase (\(D_{15}\)) is switched from tri-state mode into standard operation. At the same time, the phases are realigned with a phase shift of 90°. The phase shedding procedure does not cause any perturbation of the output voltage due to the internal distribution scheme. Note that in figure 3.16 (output voltage resolution 20mV/DIV), the unexpected increase in ripple voltage after the change in the number of phases is caused by the switching noise of the just-enabled phase as this is the nearest to the measurement probe.
Figure 3.13.: System response of the new DPWM to a 70 A load step.

Figure 3.14.: System response of the new DPWM to a 70 A load step with DPWM output signals.
Figure 3.15.: System response of a standard DPWM to a 70 A load step.

Figure 3.16.: Phase shedding operation with the new modulation scheme.
3.6. Discussion

A new multi-phase DPWM modulation scheme has been presented based on the fastest possible distribution of the duty cycle input command to the power stage with respect to the number of allowed switching actions per cycle. Utilizing a new distribution algorithm, the system provides superior transient performance and inductor current distribution over systems using conventional modulators. The proposed modulator has been implemented in an FPGA prototype and its performance assessed for a four-phase buck converter. The results presented show clear advantages of the proposed system over standard DPWM modulators with a maximum performance increase of approx. 40%.
4. Scalable Digital Power Controller with Automatic Phase Alignment and Frequency Synchronization

The trend in next-generation switched-mode power supplies will lead to modular, scalable solutions which deliver power efficiently over a wide range of operation. This chapter details a new approach to introduce more advanced control features like phase-alignment and frequency synchronization into such scalable solutions. While these methods have been incorporated into multi-phase converters in the past, they all require the distribution of information among the individual converters. In distributed solutions, dedicated communication signals have been used to share this information. An advantage of the proposed method is that it does not require such communication signals between the individual power supplies and is therefore fully scalable and cost effective. Perturbances generated by the switching actions of the individual converters on the common input/output voltage are used by the each converter to harvest information about the switching actions of its counterparts. An algorithm is proposed to align the individual phases and synchronize the switching frequencies based on this information. This allows a reduction of input/output capacitor ripple currents, similar to techniques used in multi-phase designs. Experimental results for an FPGA prototype implementation are presented.
4.1. Introduction

While today’s power supplies are typically custom-made to fit application requirements, such as output power, a trend to use simple, cost-effective building blocks is clearly evident. With the integration of MOSFETs, drivers and control logic into a single IC package, each building block delivers a certain amount of output power. A number of these blocks can be connected in parallel to fulfill the application requirements. This leads to a topology with multiple independent power supplies connected in parallel, commonly referred to as “scalable power supplies.”

The parallel connection of power supplies (figure 4.1), or the use of multi-phase converters (interleaved phases), provides several benefits such as an increase in effective switching frequency, reduced ripple voltage, improved transient performance and improved efficiency [65–67]. Conversely, issues not present in stand-alone converters arise [68, 69]. For example, converters can act as load on each other, compromising system stability and performance. Consequently, techniques to avoid these issues have been developed, referred to as “current sharing schemes,” where the load current is shared among the different converters improving system performance, efficiency and life-time [70, 71].

Another problem arising is beating of the switching signals between the individual converters, caused by differences in the actual switching frequencies. Even the smallest mismatch in switching frequency, which results in a difference of the switching periods, causes problems as the difference accumulates over time. In practice, clock synchronization schemes are required to counteract any discrepancy between the individual frequencies (caused by component variation, different operation conditions, etc.) [67].

To resolve these issues, an approach is presented which introduces phase alignment and frequency synchronization into distributed power supplies without the requirement for dedicated communication signals between the individual power converters. This chapter is organized as follows: In section 4.2, a brief overview of existing parallel power converters is given. In section 4.3 the pro-
posed system is outlined. In the following section, the phase alignment problem is analysed and a new alignment algorithm is proposed. Implementation details for an FPGA prototype system are also given. In section 4.5, the algorithm is reviewed for frequency synchronization. This is followed by experimental results in section 4.6. A final discussion concludes the chapter.

4.2. Parallel Power Converters

4.2.1. Benefits and issues

With reference to figure 4.1, the benefits of the parallelization of buck converters can be explained via the summation of the individual inductor currents at the common nodes, i.e. input/output voltage. For an ideal system with $N$ converters, the resulting waveform is identical to the waveform of a single power converter operating at $N$-times the switching frequency of the individual converters. This results in reduced capacitor ripple current/voltage (due to ripple cancellation), faster transient performance and the possible reduction of the passive components [42].

Figure 4.1.: System architecture shown with four parallel DC-DC converters.
In a uniform system optimal ripple cancellation can be obtained with an ideal current distribution, i.e. uniformly distributed currents in time and magnitude. Therefore, all converters are operated at equal inductor currents (current sharing) and the individual switching actions are distributed uniformly over time (phase alignment). Consequently, the phase shift for $N$ converters is set to a value of $\varphi = \frac{360^\circ}{N}$.

However, the advantages can only be fully exploited, if phase alignment and current matching can be guaranteed at all times. While phase alignment is relatively simple for centrally controlled systems, current matching can be challenging in practice due to mismatch of the converter impedances, lack of fast current control loops and sensitivity/noise issues in the current sensing [72]. For distributed systems, phase alignment and current sharing is more challenging and generally utilizes communication lines between the individual components [73–75].

With the lack of a common clock domain, distributed systems generally require clock synchronization schemes. While all signals in the central controller are synchronous to one single clock source, the individual controllers in a distributed system are typically driven by their individual clock sources, e.g. internal RC oscillators. Even the smallest differences in switching period caused by a mismatch in operation frequencies accumulate over time and lead to drifting of the switching cycles of the individual power supplies relative to each other. This creates unfavourable effects, such as current beating, which are not acceptable in practice. To avoid these, clock synchronization schemes are used to synchronize the timing of the individual power converters with the use of dedicated communication lines, and hence remove the negative effects.

### 4.2.2. System architectures

To use advanced features in conjunction with parallel power supplies, different concepts have been investigated in the past which can be classified with two
different criteria (illustrated in figure 4.2). The physical domain describes the physical connection between the individual units and includes

1. multi-phase systems with central controllers,
2. parallel power supplies with communication between them via dedicated communication signals and,
3. independent, parallel power supplies.

The logical system architectures can be divided into two different types, i.e.

1. master-slave systems (including central controlled systems) and,
2. master-less systems.

The first and simplest solution (Cat. 1) is the use of a central controller which controls all switches of all connected phases (figure 4.3). In this case, phase alignment between the individual switching signals is trivial as the central controller has all the information required, i.e. the total number of phases and the switching period. With only a single clock domain present in the system, clock synchronization is not required. On the other hand, these systems suffer from scalability issues as the central controller has to generate the respective switch control signals for all converters. This restricts PCB design as all signals need to be connected to the central IC. Also, the central controller is a single point of failure as it is mandatory for the operation of the system.

In an alternative system architecture (Cat. 2), the central controller is split into individual controllers; one for each power supply/ converter. These individual
controllers share information between each other via dedicated communication signals (figure 4.4). Depending on the logical operation, these designs can be further divided into two sub-categories, i.e. master-slave (2a) and master-less systems (2b).

Master-slave systems (Cat. 2a) are functionally similar to systems with central controllers where the physical implementation of the central controller is split into separate ICs; typically one IC per converter. One controller, either a dedicated central controller or alternatively one of the existing controllers,
acts as a master with all other controllers referred to as slaves. The master provides all the features that require central control or information. The slaves provide the features that require only information about the respective power supply. For example, the master can act as a clock reference so that all slaves can synchronize to it. Phase alignment or current sharing can also be provided by the master so that the individual slaves do not require complex control systems. For example in [73], the master provides one common voltage loop while each slave implements its individual current loop (following the master’s current reference). However, master-slave systems suffer from similar drawbacks as systems with central controllers, e.g. the master is a single point of failure. Additionally, two different power supplies types, i.e. master and slave, can lead to increased production and maintenance costs. This issue has been addressed with systems where identical hardware for master and slave is used and the actual functionality is configured dynamically during start-up. A digital implementation of such a system has been presented in [74] where the analogue voltage control loop has been replaced with a digital equivalent.

In contrast, in master-less systems (Cat. 2b) all power supplies are identical in hardware and functionality. A master is not required for their operation. The individual converters share signals between each other to implement advanced control features, such as phase alignment and frequency synchronization, via dedicated communication lines. In [75], a master-less control system has been proposed which enables the implementation of phase alignment and phase-sequencing via a dedicated communication line. During start-up, the system determines the number of phases and uses this information during normal operation. However, the required communication line is a limitation of the system. With the individual supplies connected in a chain-like fashion, it restricts PCB layout and compromises fault tolerance. Also hot-plugging is not supported by this particular system. In [65, 67], analogue master-less control systems have been presented which implement phase alignment via analogue communication lines. Each converter feeds a current into a common wire and uses the voltage generated by the sum of the individual currents across a known impedance as feedback. This feedback voltage allows the individual power supplies to synchronize their switching frequencies and align their respective switching actions with the use of voltage controlled oscillators (VCOs).
The third system architecture listed in figure 4.2 (Cat. 3) are systems without any communication signals between the individual power converters. While such systems have been implemented with current sharing features, such as droop-current sharing [76, 77], phase alignment has not been implemented to date.

4.3. Operational Overview

The proposed system consists of multiple independent power supplies connected to the same input/output voltage (figure 4.1). Each power supply is controlled by an independent digital core (figure 4.5). All converters, including the digital cores, are identical and hence are suitable for scalable solutions. The digital core can be embedded into existing integrated solutions leading to a single chip implementation incorporating MOSFETs, drivers and control.

With reference to figure 4.5, the digital core integrates a standard digital control loop with a new block implementing the phase alignment technique proposed in this chapter. The ADC converts output voltage and inductor current into the digital domain. A digital control law uses this information to compute the duty cycle for the DPWM which controls the power switches. The focus of this chapter is on the additional phase alignment block, which implements the features detailed in the following sections. The block is able to adapt the phase alignment of the individual power supplies relative to each other and therefore interacts with the DPWM block.

![Figure 4.5.: Proposed digital core.](image-url)
4.4. Phase Alignment

4.4.1. Optimal phase distribution

Before defining the optimal distribution of the switching actions, i.e. the phase alignment, it is first necessary to investigate the influence of phase alignment on different system parameters. Two signals are of specific interest, namely the output capacitor ripple current and the input capacitor ripple current. The output capacitor ripple current (figure 4.6) is a direct function of the phase relationship of the converters, but is independent of the actual DC inductor current level, i.e. the current distribution. On the other hand, the input capacitor ripple current is subject to the inductor current distribution and the “overlapping” of the switching pulses. With reference to figure 4.7, this can be explained by the fact that only the magnitude of the current pulses contributes to the root mean square (RMS) value, while their position in time/phase has no influence as long as the pulses do not overlap.

Figure 4.6.: Output ripple current cancellation.

Figure 4.7.: Input ripple current cancellation.
For any current distribution, the optimal phase distribution can be expressed as the distribution with the best ripple compensation, i.e. the smallest RMS ripple currents. As acknowledged in the literature, the optimal phase distribution for an equal inductor current distribution is a uniform pattern with equal phase shift between the converters.

Note that the system considered does not utilize an equal current distribution scheme. As a result, the lowest possible input ripple current cannot be achieved under all operation conditions. However, ripple cancellation can still be used to reduce ripple currents and hence improve system performance.

One possible solution is to measure the RMS capacitor ripple current and to realign the converters to minimize the ripple current. While this theoretically results in an optimal phase distribution, the required design is rather complex. Oversampling techniques or analogue measurement circuits are required to enable real RMS measurement of the small ripple current. Complex optimization techniques are required to determine the optimal distribution. This increases design complexity significantly and also requires communication signals.

A good design trade-off between complexity and benefits is to consider a system without RMS measurement. Instead of a complex measurement and optimization process, the converters are aligned with equal phase shift, similar to an ideal system, even with a non-equal current distribution present. This allows a significant reduction in ripple currents, both input and output, without requiring large implementation overhead or computation power. However, the minimal achievable input ripple cannot be as low as in a system with uniform current distribution and fully optimized phase alignment.

Note that the capacitors in power systems are designed for the worst-case, i.e. maximum output current. For this operation point, the current is shared uniformly between the converters by the current sharing scheme, so that the considered uniform phase alignment is optimal. For any other operation point, the input ripple current is below the maximum value regardless of the actual output current.
With the desired phase relationship established, two problems remain unsolved: Firstly, how can information about the other converters in the system be obtained without communication lines present? Secondly, what optimization algorithm can be used and how can it be implemented effectively?

As the proposed system is of a distributed nature and has no communication signals between the individual converters, the only common signals are the input and the output voltage (figure 4.1). Each switching action of the power switches results in perturbation of these voltages which can be quantified as ripple voltage, voltage spikes, resistive voltage drops, etc. All these effects have a common denominator, i.e. a fixed relationship with the switching action. Therefore, the voltage perturbances can be used to detect the switching of the converters without the need for dedicated communication signals. As the perturbation of the input voltage is generally larger than the perturbation of the output voltage, the input voltage has been chosen as an information source. The circuitry required to extract this switching information from the input voltage is detailed in subsection 4.4.3.

4.4.2. Principle operation of the phase alignment algorithm

For a uniform phase distribution, the time differences between the switching events of the individual converters are equal. While the absolute duration varies with the number of converters, the equality remains constant. This is the basis for the phase alignment algorithm presented.

![Phase alignment principle](image)

**Figure 4.8.** Phase alignment principle.
With reference to figure 4.8 and without loss of generality, the proposed phase-alignment principle is shown for a two converter system. With information about the switching events of the converters gathered from the input voltage, the duration between two switching events can be determined. Each converter measures the duration between the previous switching action and its own action \( (t_1) \) and the duration between its own action and the following action \( (t_2) \). The objective of the algorithm is to equalize these time intervals to achieve a uniform phase distribution. To balance these intervals and hence align the switching actions, the converters use small changes in the actual switching frequencies for single switching cycles. When the switching frequencies of the converters are marginally different, the respective switching actions drift against each other, i.e. their phase alignment changes. As a result, a switching action can be shifted forward relative to other (static) switching actions if the respective switching period is shortened. Conversely, it can be shifted backwards with a longer switching cycle.

One advantage of this algorithm is its simplicity in implementation. The algorithm can run in the background without influencing the control loop or system performance. Additionally during steady-state (when all power supplies have been aligned properly), the duration of the time interval between two converters can be compared to the total switching cycle to obtain the number of converters.

Note that the different power converters will not be aligned during start-up of the system as the converters can have any arbitrary phase relationship. This leads to an increased output ripple voltage ripple until proper alignment is reached. The input voltage ripple is also higher, but is still significantly below the level expected during maximum load operation. Even if the system does not have proper phase alignment during start-up, this is not a concern for two reasons: (a) the input capacitors are defined by the maximum output current (with proper phase alignment assumed), and (b) the system does not start up with maximum output power. To add, applications today use power-good signals so that the power supply can indicate stable output voltage to the load prior to the demand for output power.
4.4.3. Implementation

The implementation of the phase alignment algorithm and the integration into an existing control loop is relatively straightforward and takes place in two domains. Analogue circuitry is required to acquire the switching information from the input voltage of the existing power converters. A digital block is integrated into the existing digital control core and implements the control logic required by the phase alignment algorithm. A small modification of the DPWM is introduced to enable an extension/reduction of the switching period. The algorithm runs in parallel to the existing control loop without influencing the latter.

In order to extract the switching events from the voltage perturbation, analogue filtering and a conversion into the digital domain is required. This can be done in several different ways. The circuit implemented in the prototype system is detailed in figure 4.9 and consists of an AC coupling of the input voltage and a comparator. The AC coupling removes the DC component of the input voltage and feeds the high-frequency perturbation into the input of the comparator. The signal at the output of the compensator changes its value when a perturbation in the input voltage of the power stage is detected. While discrete components have been used for experimental verification, integration into an ASIC is also possible.

In the digital core, the output of the comparator (E) is time-discretized with the system operation clock and used as an input for the phase alignment logic. Two control signals are generated indicating a need for a left-shift or a right-shift.

![AC coupling circuit.](image)

**Figure 4.9.** AC coupling circuit.
right-shift of the converter’s switching action. These signals are fed into the DPWM block and are used to extend or shorten the switching period by one least significant bit (LSB), resulting in a phase shift. This allows a maximum shift of one LSB per switching cycle which has proven sufficient in practice.

With reference to figure 4.10, the implementation of the algorithm comprises of three logic blocks. The event detection block provides digital filtering of the comparator output and identifies the switching events. The control logic block, a small FSM, controls the overall process. The measurement of the time intervals is performed with an up-/down-counter which allows a direct measurement of the time difference. The counter counts up during $t_1$ and counts down during $t_2$. Two digital comparators assess the sign of the difference using a predefined tolerance band ($\pm \epsilon$) which acts as a zero-error-bin for the optimization. If the counter value is positive, i.e. $t_1 > t_2$, the phase switches “too late”. In this case, a left-shift of the switching action will improve the phase alignment and is therefore performed by a reduction of the next DPWM cycle by one LSB. Conversely if the counter value is negative, i.e. $t_1 < t_2$, the next DPWM cycle is extended by one LSB.

As the proposed technique utilizes the perturbation on the input voltage signal, the noise on this signal should be also taken into account. To remove permanent jitter in the phase relationships, due to noise on the input signal, several different methods have been explored. As already detailed, a zero-error-bin is used to reduce hysteretic jitter. However, the zero-error-bin should not

---

**Figure 4.10.** Digital implementation of phase detection/alignment block.
be chosen too large, as it also sets the minimum achievable phase difference. Additionally, a small black-out window is integrated into the event detection logic which suppresses the possible redetection of a switching event directly after its first detection (caused by effects such as ringing). In the presence of extensive noise, an averaging of the corrective actions has proven beneficial. Instead of changing the DPWM period directly after one corrective action has been identified by the control logic, it is only changed when multiple consecutive correction actions have been detected. This removes the influence of corrections based on misleading events, but slows the alignment process down.

4.5. Frequency Synchronization

A common issue with the use of parallel power converters is the mismatch in switching frequencies. In practice, this issue can be observed in all systems with independent power supplies due to unavoidable tolerances, and has to be addressed with additional synchronization schemes. Conversely, if all phases are controlled by one single digital core, frequency matching is not an issue. Unlike sub-optimal phase alignment which leads to enlarged RMS ripple currents, mismatch in the switching frequencies results in input current beating which adds additional stress on the input power system. The beat frequency can be as low as a few of Hertz as it is determined by the difference between the actual switching frequencies. Such low frequencies can cause problems in the respective input power supplies and can lie within the audio range. On a cycle-by-cycle basis, the mismatch in switching frequencies results in an effect called “drifting,” where the switching actions shift in relationship to each other. By design, the proposed system can compensate for drifting as it realigns the converters during run-time up to a maximum tolerance without any additional modifications.

To calculate the maximum acceptable tolerance for the proposed system, consider

\[ \Delta T = \delta T_{sw} \],

(4.1)
as the error introduced during each single switching cycle due to the mismatch of the switching periods between two converters. \( \delta \) represents the tolerance of the switching cycle which is typically the tolerance of the clock source; \( T_{sw} \) represents the switching period. The phase alignment procedure allows a maximum correction of the switching period by one DPWM LSB during each cycle where a corrective action is performed. However, as some (low pass) filtering is required by the signal processing logic, only one switching period can be changed during a certain number of switching periods, referred to as \( N_{LP} \). Therefore, the maximum corrective adjustment is given by

\[
\Delta T = \frac{1}{N_{LP}} \frac{1}{2^{N_{res}}} T_{sw},
\]

where \( N_{LP} \) is the number of switching periods between two corrective actions and \( N_{res} \) the resolution of the DPWM in bits. As a result, the maximum tolerance the system can compensate can be calculated as

\[
\delta_{\text{max}} = \frac{1}{N_{LP}} \frac{1}{2^{N_{res}}}
\]

For a typical system with a 9-bit DPWM and one corrective action every ten switching periods (\( N_{LP} = 10 \)), a maximum tolerance of 195 parts per million can be compensated for. For comparison, the tolerance of typical quartz oscillators is in the range of 10-100 parts per million. However due to their cost, quartz oscillators are not used in digital power converters. Instead RC oscillators prove sufficient in single chip applications where all control signals are generated synchronously within the chip. If RC oscillators are used in distributed systems, clock synchronization is required to compensate for frequency differences between the converters. While this typically requires a separate communication line and continuous resynchronization, the proposed scheme can provide both without the need for an additional communication line.

The tolerance limitation discussed is caused by the limitation of the DPWM to change its switching frequency permanently. If only phase alignment is required, i.e. the tolerance is within the calculated range, a DPWM with
a variation of one LSB per cycle is sufficient for operation. It allows the
extension/reduction of a switching cycle by one LSB for a switching cycle, but
does not support a permanent change of the switching frequency. Consequently,
a small modification of the DPWM is required to enable a permanent change
in order to compensate for larger tolerances. Note that if a change of the
switching cycle by one LSB is too coarse, alternative methods need to be
explored. Possible options are dithering of the switching cycle or a modification
of the controller’s system clock via a VCO or phase-locked loop (PLL).

4.6. Experimental Validation

The proposed system has been implemented and assessed in practice on two
different platforms utilizing the prototype framework presented in subsection 3.5.1. The first system features two parallel converters, while the second
system uses a total of four. The technical specification of the experimental
converters is listed in table 4.1. Both systems are controlled by an Altera
DE2 evaluation board employing a Cyclone 2 FPGA to implement the two,
respectively four, identical digital cores required. In a commercial application,
these cores would be implemented on separate ICs.

The phase alignment principle for the prototype with two converters is shown
in figure 4.11 and figure 4.12. Figure 4.11 shows an arbitrary initial phase
alignment prior to enabling the phase alignment algorithm. The input voltage
with the comparator threshold is shown above the switching signals ($S_0, S_1$) of
the two independent converters, along with the comparator output ($E_0$), and
the digitally filtered equivalent (Filtered $E_0$), for one of the two converters

<table>
<thead>
<tr>
<th>Table 4.1.: Technical details for the experimental converters.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input voltage</td>
</tr>
<tr>
<td>Output voltage</td>
</tr>
<tr>
<td>Inductance</td>
</tr>
<tr>
<td>Switching frequency</td>
</tr>
<tr>
<td>Output power</td>
</tr>
</tbody>
</table>

102
(controlled by $S_0$). When the algorithm is enabled, the phases align to $170^\circ$. This is a difference of approx. 5% from the optimal alignment ($180^\circ$) and proves sufficient in practice. Note that the delay between the switching action and the digital data stream is due to internal delays in the FPGA caused by synchronization and processing. It does not affect the loop performance as the system has been internally compensated for.

To investigate the practical scalability of the system, a second prototype with four converters has been built using the same technical specification as for the two converter system. The respective phase alignment is shown in figure 4.13 where each switching signal ($S_0, S_1, S_2, S_3$) is plotted above the respective comparator output ($E_0, E_1, E_2, E_3$). The resulting phase alignment is not optimal as not all converters are able to extract sufficient information from the input voltage. Therefore, three converters (controlled by $S_0, S_1, S_2$) optimize their phase alignment assuming a three converter system (which they in fact do considerably well). The fourth converter is not taken into account. However, even if the result is not optimal, it is stable which is more important in practice than achieving an optimal result. This is due to the fact that each converter optimizes its respective switching action in such a way that it is centered between the switching events detected. Two main reasons for the sub-optimal information extraction have been identified: (a) the input voltage of the fourth converter (controlled by $S_3$) is not coupled sufficiently with the top two converters (due to the PCB layout) and (b) a simple AC coupling circuit is used in the prototype. An optimization of the coupling circuitry will improve the performance of the coupling filter, allow for better extraction of the switching events, and hence improve the resulting phase alignment.
Figure 4.11.: Phase alignment for a two converter prototype prior to the enabling of the proposed alignment algorithm.

Figure 4.12.: Phase alignment for a two converter prototype after enabling of the proposed alignment algorithm.
Figure 4.13.: Resulting phase alignment for a four converter prototype.

4.7. Conclusions

An approach for digital control of independent, parallel power supplies suitable for today’s independent, scalable power converters has been presented. It has been shown that phase alignment and frequency synchronization without the need for additional communication lines between individual converters is possible. Perturbances generated by the switching actions of the converters on the common input voltage are used by each individual converter to harvest information about its counterparts. An algorithm has been proposed which uses this information to align the switching actions of the different converters without requiring excessive hardware resources. The proposed algorithm has been analysed, implemented and tested on two prototype systems. Good phase alignment is achieved, particularly for the two converter prototype. It is expected that an optimization of the coupling circuitry will further enhance the performance.
5. Current Distribution Scheme for Independent Power Supplies based on Efficiency Levels

The trend in next-generation switched-mode power supplies will lead to modular, scalable solutions which deliver power efficiently over a wide range of operation. This chapter details a new approach to introduce more advanced control features to improve system efficiency into these scalable solutions. While these methods have been incorporated into multi-phase converters in the past, they all require the distribution of information among the individual converters. An advantage of the proposed method is that it does not require such communication signals between the individual power supplies and is therefore fully scalable and cost effective. A system comprising of individual, smart converters is proposed where each converter regulates its respective output power to a level with high efficiency. Converters not required for the delivered output power are shut down. The proposed approach is analysed theoretically. Implementation details for an FPGA experimental prototype system are given. The system performance for a four converter prototype system is analysed and discussed. The efficiency obtained is compared with the efficiency of a multi-phase system with phase-shedding operation and the efficiency of a system with independent power converters without phase shedding support.
5.1. Introduction

With the need for today’s power supplies to deliver more output power for less cost, a trend to scalable digital power solutions has emerged. Standardized power supplies are connected in parallel to satisfy application-specific power demands without the need for different converters (figure 5.1). While parallel power converters provide several benefits, such as reduced ripple currents, higher efficiency and faster transient response [63, 78–81], they also introduce issues non-existent in single-converter applications, such as current distribution among the converters, frequency synchronization, phase alignment and fault behaviour.

5.1.1. Control architectures

In the past, several different control topologies have been presented to control such parallel power converters. They can be quantified into three different categories: central control architectures, distributed control systems with communication and independently controlled systems.

Central control architectures comprise of one central controller which generates the switching signals for all connected power converters, in this case often called phases. An advantage of this concept is the availability of all information at one central point, i.e. the controller, which allows the implementation of more

![Figure 5.1.: System architecture shown with four parallel DC-DC converters.](image-url)
advanced control schemes or efficiency improving techniques. A disadvantage is that the central controller restricts the scalability of the system and is a single point of failure.

Consequently, a second architecture has been developed to address this issue. Distributed controllers do not require a central controller as the control architecture is distributed over the individual power converters. Each converter is controlled by its own controller which is connected to the controllers of the other power converters via dedicated communication lines. These enable the distribution of information among the individual converters and allow the use of more advanced control schemes, such as current sharing.

Two different configurations exist within this group. In master-slave configurations, one converter acts as a master while the remaining converters act as slaves. The master is in charge of all central control features, such as current sharing, while each slave follows the master’s commands. For example in [81, 82], the master controls the output voltage using one common voltage loop. It provides a current reference signal to the individual slaves via a common bus. Each slave follows this current reference signal and effectively acts as a current source. While this scheme provides good current sharing capabilities, all converters are in operation at all times. Individual converters cannot be disabled to improve the overall efficiency. In [74], a similar digital implementation is presented substituting the analogue current reference with a digital signal.

In comparison, master-less configurations, such as [69, 83, 84], use individual, but functionally identical power converters, i.e. no master is present. More advanced control features, such as current sharing, are introduced by sharing information, such as the average output current, among the converters. However, this commonly requires complex connection of the individual converters such as chain-communication lines [83], which limits scalability and fault tolerance. Hot-plugging, i.e. the addition/removal of individual converters during run-time without power-down time, can be an issue, as it can break up the communication signal.
The third system architecture widely used is the parallel connection of multiple independent power supplies where the individual units are not aware of each other and operate individually using passive current sharing techniques. One method to implement such passive current sharing is droop current sharing [76, 77] where the output impedance (natural or artificial) of the power supplies is employed to distribute the current equally among the converters. A non-zero output impedance causes the output voltage to decrease with increasing output current. As a consequence, other converters with lower output currents and therefore higher output voltages take over more current. The system reaches its equilibrium when each individual converter delivers equal output voltage, leading to an even current distribution. To provide sufficient current sharing in the presence of system variation, a rather high output impedance is mandatory. However, a high output impedance is undesirable in most of today’s applications, rendering this scheme no longer applicable. Additionally, this scheme does not disable individual power converters during light-load, which compromises efficiency for this operation mode.

### 5.1.2. Existing current sharing/distribution techniques

While the need for current sharing among parallel power converters is widely acknowledged, several different current distribution patterns and concepts have been investigated in the past. These can be broadly categorized into two groups depending on the availability of the current information. Methods within the first group, such as chain-controlled systems or multi-phase converters, require the sharing of information among the individual converters and therefore require communication signals or a central controller. Methods within the second group, such as passive current sharing, have been developed for parallel power supplies. While they do not require the distribution of the current information, they generally do not accommodate efficiency-improving techniques like phase-shedding.

However, both methods share one common property, i.e. the actual distribution pattern where the current is shared equally among the individual converters. While this is the optimal distribution for identical converters (see section 5.3),
there are some limitations in practice. For example, current-sensing elements suffer from tolerances which can lead to a non-uniform distribution or may require calibration [72, 85]. Also the individual converters are typically not identical, due to effects of component variation, so that an equal distribution is rendered suboptimal in terms of efficiency.

Alternative distribution schemes operate the individual converters with different output currents to improve the overall efficiency of the system. In [63, 86], current distribution schemes for multi-phase converters are detailed where the current is distributed inversely-proportional to the converters’ output impedances by equalising the duty cycle control signals. This distributes the losses equally among the converters (rather than the currents) and improves the overall system efficiency. To achieve this, the system in [86] employs a current sharing procedure, while the system in [63] uses a digital filter.

However, none of the reviewed approaches is able to change the number of active converters; a technique referred to as “phase shedding.” Phase shedding improves the overall efficiency [22] and is widely used in today’s applications especially for light-load operation. It is commonly implemented in parallel with the current sharing scheme using information about the total output power as the current control loop is not able to change the number of active converters.

To combine the advantages of independent converters, such as scalability, with the benefits of multi-phase converters without the need for communication lines, a new technique for current distribution is investigated in this chapter. Control for smart independent power converters is proposed which introduces methods to improve efficiency, currently only employed in multi-phase converters, into independent parallel converters (figure 5.1). The proposed smart control algorithm is implemented in a digital control core; one for each converter (figure 5.2). Information about the total output current (or the average current), i.e. communication signals between the individual converters, is not required by the individual converters.
To further enhance overall system performance, the current distribution concept can be combined with the concept for phase alignment and frequency synchronization proposed in chapter 4. In this case, the system in figure 5.1 is augmented with the coupling circuitry shown in figure 4.1 and the two digital cores (figure 5.2 and figure 4.5) are combined into one design.

This chapter is organized as follows: in section 5.2, a current distribution method suitable for independent, parallel power supplies is proposed which allows the introduction of more advanced features such as phase shedding and/or hot-plugging into existing systems. In section 5.3, a theoretical analysis and optimization of the efficiency for parallel power converters is given. This is followed by implementation details and experimental testing on an FPGA prototype system. A final discussion concludes this chapter.

### 5.2. Current Distribution based on Optimal Current Levels

In the following, a current distribution scheme for distributed digital power controllers is proposed which distributes the current to improve the overall efficiency. Individual converters are shut down during light-load condition improving the overall efficiency. Furthermore, no communication signals are required between the individual converters as each converter only requires information about its local output current.

![Figure 5.2: Proposed digital core.](image)

111
Instead of controlling the inductor currents and the number of active converters separately and distributing the output power equally among the converters, the proposed system is based on individual smart converters with optimum current levels. Each converter regulates its output current independently to a target value with optimal efficiency and is also capable of enabling or disabling its operation independently. Therefore it does not require information about the output current from the other converters nor information about the total number of active converters. The concept is based on the principle that the total output current is the sum of the individual currents so that individual converters can change their respective output currents while the remaining converters compensate for this change.

This is enabled by the efficiency curves of the individual power supplies over the output power range. It is shown in section 5.3 that for any given number of identical power converters, the optimal current distribution is uniform for identical power supplies. To elaborate, for non-identical power converters, the distribution is inversely proportional to the respective output impedance. A method is introduced to determine the number of converters to achieve best efficiency. From the resulting criterion which is a function of the auxiliary losses of the converter, e.g. drive losses, the operation range of an individual power converter is derived. It is shown that all these operation ranges rely within a certain current range which is considered the optimal power level. The task of each individual power converter is to either operate within this range or to reduce its output power and disable its switching action.

With no communication signals present, the question is how this can be achieved by independent converters without causing conflicts of interest. A conflict of interest occurs when different converters try to achieve contradictory goals. For example, if a converter increases its output power level and no other converter in the system is able to reduce its power level, the output voltage will increase undesirably.

To reduce the risks of such conflicts, a prioritization is introduced into the control system which allows the different converters to pursue their respective goals with different precedence. For this reason, the output current is classified
into different zones each with a certain priority assigned. These priorities depend on the efficiency of the total system and the converter operation within a certain zone. The different zones are detailed in figure 5.3. The top figure shows the output current levels with the optimal operation zones highlighted and the direction of optimization indicated. In the middle figure, the respective optimization direction is illustrated with the cost-function $J(I_{out})$, where the system will always try to reduce the cost, i.e. move to the lowest possible value. The bottom figure shows the zone’s priorities $P(I_{out})$, where a larger value corresponds to a higher optimization strength.

The highest priority is given to converters operating with negative output currents, so that they can increase their currents to zero first. Note that, negative output current is highly unfavourable in terms of efficiency, but improves the performance during heavy-to-light-load transients. The second highest priority is assigned to converters operating around the optimal operation area which allows these converters to fix their current levels at the optimal point. This is followed by converters operating with output currents higher than the optimal operation area (zone 6). Consequently, they will reduce their output currents to operation within the optimal range (zone 5). Converters in zone 4 will follow by increasing their output power to operate also within optimal range (zone 5). Converters operating in zone 3 will start their current transfer process. However, the direction for these transfers is not fixed as it is not known whether more or less output power is more beneficial. Therefore, such converters will chose their desired target levels randomly at the beginning of each transfer. Finally, converters operating with very little output power will reduce their output current and will eventually switch off.

Due the lack of communication lines, the entire prioritization scheme is implemented via timing. After a load transient, each converter detects steady-state operation by assessing output voltage and inductor current. The converters reach steady-state at approximately the same time instance and start their respective transfer processes. Shorter wait times within the procedure allow converters with higher priorities to correct their values first.
Figure 5.3.: Optimization of the output currents.

To ensure stability, the proposed current distribution scheme is embedded into a sequence of conditions, illustrated in the (simplified) flow chart in figure 5.4. Prior to enabling the current transfer between the individual converters, the converters ensure that the system is in steady-state operation. Therefore output voltage and inductor current are monitored. If both are stable (indicated by the signals Vss, Iss), the system continues with the current transfer procedure. Within this, each converter waits a time period (depending on priority) prior to enabling the actual current transfer so that it can be ensured that not all converters start their transfer at the same time. During this time, the converters continuously monitor output voltage and inductor current. If any other converter starts its transfer process in the meantime, the wait time is reset and the procedure starts again from the beginning (not shown in the flow chart in figure 5.4).

During the entire process, the stability of the output voltage is monitored closely as it has highest control priority. Note that the output voltage changes
marginally during the current transfer process. This is due to the fact that it is the only way to “communicate” with the other converters. When a converter wants to offload current to other converters, it decreases its reference voltage. This causes its loop compensator to decrease the inductor current which leads to a small decrease in output voltage. The other converters sense the voltage drop and increase their respective inductor currents. When the transfer is complete, the output voltage returns to its nominal value. The value of the output voltage deviation during the current transfer process is proportional to the speed of the transfer. Larger deviation from the reference value allows a faster current transfer.

With reference to figure 5.3, the optimal operation zone spans between two thresholds, $I_{\text{opt\_min}}$ and $I_{\text{opt\_max}}$, around an optimal output current, $I_{\text{opt}}$. The remaining question is the selection of these levels and the level at which the converters are turned off, $I_{\text{down}}$.

![Figure 5.4.](image)

**Figure 5.4.** Flow chart of the current distribution procedure.
5.3. Theoretical Analysis

Figure 5.5 shows a typical efficiency plot for a power converter. The efficiency is low for small output power due to predominant constant losses, and for high output power due to the increasing contribution of resistive losses. It is highest in the mid-range. In figure 5.6, multiple identical converters are connected in parallel and the efficiency is analysed for different numbers of active converters and uniformly distributed current.

Also, it is clearly visible that the optimal number of active converters is a function of the output current. By changing the number of active converters, the overall efficiency can always track the highest efficiency curve for a given output current. To allow for an efficient implementation and to ensure optimal operation, the system has to actively control the number of active converters. In a multi-phase system, where the total output current information is known, this is implemented with static threshold currents where the system adjusts the number of active phases. The following analysis determines these thresholds and is used in this design to establish common properties.

![Efficiency Plot](image)

**Figure 5.5.:** Typical efficiency of a power converter.
The search for the highest efficiency can be transformed into the search for minimal losses for a given output current. Losses in power converters have been analysed in the past [22, 87–89] and can be categorized into three main groups, referred to as resistive losses $P_{\text{res}}$, drive losses $P_{\text{drive}}$ and auxiliary losses $P_{\text{aux}}$. Note that resistive losses include all losses which are a function of the output current even if they are strictly speaking not entirely resistive. A parabola can be used as a good approximation [89], resulting in

$$P_{\text{res}}(I) = V_{\text{loss}}I + R_I I^2.$$  \hspace{1cm} (5.1)

The drive losses summarize all losses dependent on the number of active converters $n$, while the auxiliary losses include all losses constant during operation.

The losses for each converter $k$ can be expressed as

$$P_{\text{loss},k} = P_{\text{aux},k} + P_{\text{drive},k} + P_{\text{res},k}(I_{\text{out}}),$$  \hspace{1cm} (5.2)

so that the total losses of $N$ parallel converters are

$$P_{\text{loss}} = \sum_{k=0}^{N-1} \left( P_{\text{aux},k} + P_{\text{drive},k} + P_{\text{res}}(I_{L,k}) \right).$$  \hspace{1cm} (5.3)

This function is now recast as an optimization problem with the objective of determining the optimal number of active converters $n$ and the respective inductor currents $I_L$ for a given output current $I_{\text{out}}$.

In order to do so, (5.3) is separated as follows

$$P_{\text{loss}} = P_{\text{aux}} + \sum_{k=0}^{N-1} P_{\text{drive},k} + \sum_{k=0}^{N-1} P_{\text{res}}(I_{\text{out},k}),$$  \hspace{1cm} (5.4)

where the three parts represent the constant losses, the losses dependent on the number of active converters and losses dependent on the inductor current.
Figure 5.6.: Typical efficiency of a modular power converter for different numbers of active converters: $V_{in} = 12\,V$, $V_{out} = 1.5\,V$.

As the sum of the individual output currents equals the total output current, (5.4) is subject to the following constraint:

\[ I_{out} = \sum_{k=0}^{N-1} I_{out,k} \]  

(5.5)

To verify these equations, they are now optimized considering $N$ identical converters, i.e. converters with identical loss functions. However, the equation is independent of the actual loss functions and therefore can also be applied to systems with different converters. Such differences can be caused by system tolerances as analysed in [63] or by design as described in [79, 90].

Equation 5.4 can be optimized for a given tuple $(n, I_{out})$ in order to determine the optimal current distribution using well-known optimization methods, e.g. Lagrange multipliers. As a result, the lowest losses can be achieved for an equal current distribution with

\[ I_{L,n} = \frac{1}{n} I_{out} \]  

(5.6)
This is the expected result which states that for a given number of active converters \( n \), the losses are minimal when the current is distributed equally. However, this does not explicitly state the optimal number of converters for a given output current. This is now determined by substituting (5.6) into (5.4) resulting in

\[
P_{\text{loss}}(I_{\text{out}}, n) = P_{\text{aux}} + \sum_{k=0}^{n-1} P_{\text{drive}, k} + \sum_{k=0}^{n-1} P_{\text{res}}(\frac{1}{n} I_{\text{out}}),
\]

which would be minimal for

\[
n = \sqrt[2n+1]{R_L I_{\text{out}}^2} = I_{\text{out}} \sqrt{\frac{R_L}{P_{\text{drive}}}},
\]

if \( n \) is a real number. However, \( n \) is an integer value, so that the standard minimization, i.e. the first-order derivative set to zero, does not apply. While (5.8) reveals the optimal number of active converters \( n \) as a function of the output current, it is more favourable for an efficient hardware implementation to know the actual threshold values. This results in one optimal current interval for each number of active converters and can be implemented via a small look-up table. To calculate the width of each interval, the problem is reformulated into the search for the threshold currents where it is beneficial to increase/decrease the number of active converters. This is equivalent to the computation of the discrete derivative and can be mathematically expressed as

\[
P_{\text{loss}}(n + 1, I_{\text{out}}) - P_{\text{loss}}(n, I_{\text{out}}) < 0,
\]

which can be solved as

\[
I_{\text{out}} > \sqrt{n(n+1)} \sqrt{\frac{P_{\text{drive}}}{R_L}}.
\]

For an easier referral a normalized threshold current, constant for a given power converter, is defined as

\[
I_{\text{th,nom}} = \sqrt{\frac{P_{\text{drive}}}{R_L}}
\]
which leads to
\[ I_{\text{out}} > \sqrt{n(n+1)} I_{\text{th,nom}} \, . \tag{5.12} \]

For example, the current threshold for a change from two to three converters is
\[ I_{\text{out}} = \sqrt{2(2+1)} I_{\text{th,nom}} \approx 2.45 I_{\text{th,nom}} \, , \tag{5.13} \]

If this total threshold value is scaled back into the current per converter, the operation intervals for each of the individual converters can be calculated as
\[ \sqrt{\frac{n-1}{n}} I_{\text{th,nom}} < I_L(n) < \sqrt{\frac{n+1}{n}} I_{\text{th,nom}} \, . \tag{5.14} \]

The resulting scaling factors are listed in Table 5.1. All individual converters operate in an interval around the optimal output current which shrinks with increasing output current. This can be explained with the reduction of the resistive losses, due to smaller inductor currents compared to the drive losses of an additional turned-on converter.

While the detailed optimization defines clear threshold currents in a typical multi-phase application (where the total output current is known at a central point), it cannot be directly applied to the system detailed as the defined threshold currents are only valid if the output current is distributed equally among all active converters after an additional converter is turned on. If the enabled converter only takes the additional current and the other converters maintain their output currents, the proposed equations are not valid. However, they provide good insights to the optimal operation interval required for the current sharing scheme presented in this chapter.

**Table 5.1.: Normalized current thresholds for varying number of converters**

<table>
<thead>
<tr>
<th>( n )</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>( I_{\text{out}}/I_{\text{th,nom}} )</td>
<td>1.41</td>
<td>2.45</td>
<td>3.46</td>
<td>4.47</td>
<td>5.48</td>
<td>6.48</td>
</tr>
<tr>
<td>( I_{L,\text{min}}/I_{\text{th,nom}} )</td>
<td></td>
<td>0.71</td>
<td>0.82</td>
<td>0.87</td>
<td>0.89</td>
<td>0.91</td>
</tr>
<tr>
<td>( I_{L,\text{max}}/I_{\text{th,nom}} )</td>
<td>1.41</td>
<td>1.22</td>
<td>1.15</td>
<td>1.12</td>
<td>1.10</td>
<td>1.08</td>
</tr>
</tbody>
</table>
The acquired results are now applied to the proposed scheme in order to select the required power levels. With $I_{\text{opt}}$ selected according to (5.11), the levels spanning the optimal zone have to be selected. While this could be performed using (5.14), more considerations should be taken into account. Namely, it has to be ensured that the spanned zone is not too wide.

Consider the following scenario: a certain number of converters are off, a certain number operate in the optimal operation zone and a single converter operates between zero and the optimal operation zone. The question is if the converter should transfer its output current into the optimal zone or if it should be turned off, noting that there exists an optimal number of active converters for a given output current. Also it should be ensured that the converter does not remain on when it is actually not required; nor should it be turned off when it is required.

To elaborate, consider $n$ converters operating at a load current within the optimal operation zone. Each current is subject to a stochastic process and its actual value within this optimal operation zone is unknown. As a result, the current uncertainty per converter equals the width of the optimal zone. With $n$ converters active in the optimal range, a total current uncertainty of

$$\Delta I = n \Delta I_{\text{span}} = n(I_{\text{opt\_max}} - I_{\text{opt\_min}})$$  \hspace{1cm} (5.15)

is expected. The sum of the total uncertainty and the threshold current should not exceed the lower threshold of the optimal operation range

$$\Delta I + I_{\text{down}} < I_{\text{opt\_min}}$$, \hspace{1cm} (5.16)

to ensure that the optimal number of converters is active.

With all design parameters derived, a review of the resulting efficiency can be performed. Analysing the resulting efficiency of the overall system is difficult, due to the statistical distribution of the individual load currents in the optimal operation region. Figure 5.7 shows the expected efficiency for the worst-case scenario with all but one converter operating at the maximum current in the optimal range. The efficiency is less than the efficiency of a multi-phase
converter with ideal current sharing. However compared to existing systems without active current sharing or phase shedding, the efficiency is significantly improved for light-load operation, as these systems will operate always with all converters active. Also it should be noted that the proposed system can also be used for systems with non-identical converters which can further improve the efficiency during light-load operation.

5.4. Implementation

A block diagram of the proposed current distribution scheme for an implementation in discrete logic, feasible for FPGAs, is detailed in figure 5.8. The system consists of four units: The zone-decoder determines the current operation zone of the converter (figure 5.3). The Supervisor logic assesses the converter’s operation state and signals steady-state conditions to the control logic. The control logic acts as overall control unit for the algorithm. An additional auxiliary block implements a timing- and random-generator.

![Block Diagram of Proposed Current Distribution Scheme](image)

**Figure 5.7:** Typical efficiency of a traditional multi-phase power converter compared with the new current distribution scheme applied to a system with identical converters: $V_{in} = 12\,\text{V}$, $V_{out} = 1.5\,\text{V}$. 

122
Figure 5.8.: Implementation of the current distribution scheme.

The current distribution procedure, together with the required prioritization, is implemented using a small FSM. With reference to figure 5.9, the current sharing system can be in one of four states. The system remains in the unsteady state until output voltage and output current have settled. As soon as the outputs have stabilized, the target current is determined using the current output current and the optimization diagram in figure 5.3. The system transfers into a wait state where it remains until the desired wait time has expired. At this stage the state is changed to the active state where the current distribution scheme is enabled and the output current is transferred into the optimal operation zone. The converter remains in the active state until output voltage or current are disturbed by external events, e.g. load transients. When disturbed, the system goes back into the initial state (unstable) and the procedure is restarted. An additional lock state is used to disable the current control procedure in case the converter cannot reach its optimal current value. For example, this is required if the optimal currents have not been selected properly and one converter has to deliver non-optimal current. Without a lock state, this converter will try to reach an optimal current indefinitely and consequently create perturbation on the output voltage.

The current distribution FSM is integrated into an existing digital control loop; in this case a predictive digital current programmed control architecture
Figure 5.9.: Finite-state-machine used to implement the current distribution scheme.

(figure 5.10). Such schemes have been presented in the past [91, 92] and provide direct control of the inductor current via a dedicated current reference value. An inner current loop regulates the inductor current on a cycle-by-cycle basis, while an output voltage control loop provides the current reference value. The control loop is augmented with an additional input to control the inductor current by increasing/decreasing the reference.

When implementing the proposed control scheme, special attention must be given to the prioritization in order to avoid potential control conflicts between converters, as detailed in section 5.2. Therefore, the individual controllers are subject to two types of prioritization. The different zones are prioritized so that the converter operating in the “most undesirable” zone changes its operation state first. Additionally, the converters are prioritized so that two converters do not start their current control operation at the same time when they operate within the same zone. This priority can either be hard-coded, so that particular

Figure 5.10.: Digital control loop with predictive current mode control.
converters are always favoured, or alternatively random. For the latter, all converters have equal priority and a random factor decides about the actual priority. This balances the usage of the individual converters over time and hence increases the life-time of the system compared to a hard-coded (static) prioritization scheme. Note that if the randomness is not induced effectively, different converters can in fact create control conflicts. For example, in a case of two active converters, both converters may decide to change their output currents to higher values. This leads to an increase in output voltage until the converters detect a conflict of interest and resume voltage mode control. Hence, one of the converters has to deliver non-optional current until it can achieve a more optimal level. In this case, the converter restarts the current distribution procedure after a load transient which can be either indicated by a large voltage error (for a fast transient) or a change in operation zone (for a slow load transient). For the implementation detailed, a randomization scheme has been chosen. The random contribution is selected so that the probability for a conflict of interest is small \((< \frac{1}{64})\) when two converters are in the same operation zone.

5.5. Experimental Testing

The proposed system has been implemented and assessed in practice using a total of four parallel converters (technical specification in table 5.2). The prototype system employs the framework presented in subsection 3.5.1 which is controlled by an Altera DE2 evaluation board. The four identical digital cores have been implemented on the Cyclone 2 FPGA. Note that for a real system these cores would be implemented on separate ICs.

<table>
<thead>
<tr>
<th>Table 5.2.: Technical details for the experimental converters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input voltage</td>
</tr>
<tr>
<td>Output voltage</td>
</tr>
<tr>
<td>Inductance</td>
</tr>
<tr>
<td>Switching frequency</td>
</tr>
<tr>
<td>Output power</td>
</tr>
</tbody>
</table>
Figure 5.11 shows a typical current transfer process where the current is transferred from one converter to another after a load transient. For illustration purposes, two active converters have been chosen. When a load step (12A) is applied to the system, the two converters regulate the output voltage back to the reference value. This is followed by the prioritization phase until one of the converters initializes its transfer process. Converter 1 decreases its output current while converter 2 compensates for this change, i.e. it increases its output current. After the transfer converter 1 is switched off (not shown in the figure). The output voltage deviates slightly during this process as it is the only method of communication between the converters. The speed of the transfer is dependent on the output voltage tolerance where a larger tolerance accommodates a faster transfer.

The efficiency curve for a four-converter prototype system is shown in figure 5.12. The measured profile includes power drain loss and drive loss, but does not include the power consumed by the measurement circuits or the FPGA. The operation of the system with different numbers of active converters is clearly visible. The number of active converters over the output current range matches the expected value and therefore validates the functionality of the current distribution scheme. Small discrepancies in the efficiency are due to mismatch between the simulation models and the hardware implementation,

Figure 5.11.: Typical sequence of events for the proposed current distribution scheme.
and measurement tolerances. Note that the random influence of the current behaviour in the optimal zone cannot be modelled. The system efficiency is compared with a standard current distribution scheme which distributes the current equally among the converters. This has been implemented using the same prototype system, where the individual converters are now controlled from one central point. The efficiency of the proposed approach is marginally less than the efficiency of a multi-phase solution with equal current sharing and external phase shedding control.

Note that the efficiency shown is for an “optimal current” in the range of 9 to 11 A. This value has been chosen to illustrate the operation principle and does not relate to the optimal current value for the prototype system which is approximately 6 A.

The current distribution over the individual converters and the respective number of active converters over the load range is shown in Table 5.3. The current is distributed so that the converters operate within the most efficient range, i.e. 9 to 11 A. The random distribution of the load current over the different converters shown is due to the equal prioritization of the individual converter which results in a first-come-first-serve basis, i.e. the converter, which

![Figure 5.12: Measured efficiency of a four converter prototype system compared with a standard current sharing scheme.](image-url)
decides to take the load current first, will get it. Note, the resulting current
distribution also depends on the load transient. If the current increases slowly
within one operation zone, the active converters share the additional current
equally. If the current increases more rapidly, additional converters are enabled
to improve the transient performance. Once the current has settled, they are
turned off again if not required by the load. To elaborate, the two different
cases can be observed in table 5.3: For the 10 A case, the load current has
been increased from 5 A to 10 A, hence converter 1, already active during the
5 A case, takes the additional load current. In contrast, the output current of
15 A has been applied in one single step leading to a different distribution of
the output currents.

The random influence of the proposed current distribution scheme on the
overall efficiency is shown in figure 5.13. The efficiency has been measured for
a random sequence of output currents with multiple measurements for each
current value. This is further illustrated in the “time lines” in appendix C
where measurements for a constant step size are also shown.

Despite the standard measurement variations, it can be observed that the
efficiency varies for constant current values depending on the measurement
case. This can be explained by the fact that different converters and different
current distributions are used at different times. In the zone where only one
or two converters are active, the first effect is clearly visible, where different
converters (with different efficiencies) deliver the output current. Note that the
efficiency difference between the converters has been deliberately enforced for
this measurement to illustrate the effects. These effects are also visible in the
range between two to three converters where two different efficiency patterns
can be observed which are dependent on the selected converters. For high
output currents the spread efficiency is rather small, as the phases operate
close to their maximum output capabilities.
Figure 5.13.: Measured efficiency variation for repetitive measurements with arbitrary load current steps.

Table 5.3.: Current distribution for a four converter configuration

<table>
<thead>
<tr>
<th>$I_{out}$</th>
<th>5 A</th>
<th>10 A</th>
<th>15 A</th>
<th>20 A</th>
<th>25 A</th>
<th>30 A</th>
<th>35 A</th>
<th>40 A</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I_{L0}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>9 A</td>
<td>9 A</td>
<td>10 A</td>
<td>10 A</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$I_{L1}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>5 A</td>
<td>9 A</td>
<td>9 A</td>
<td>11 A</td>
<td>11 A</td>
<td>10 A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$I_{L2}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>6 A</td>
<td>5 A</td>
<td>10 A</td>
<td>9 A</td>
<td>11 A</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$I_{L3}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>11 A</td>
<td>11 A</td>
<td>5 A</td>
<td>9 A</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$N$</th>
<th>1</th>
<th>1</th>
<th>2</th>
<th>2</th>
<th>3</th>
<th>3</th>
<th>4</th>
<th>4</th>
</tr>
</thead>
</table>
5.6. Conclusions

An approach for digital control of parallel power supplies suitable for today’s independent, scalable power converters has been presented. With the introduction of control methods only present in existing multi-phase converters, it has been shown that the system performance can be improved over existing parallel solutions without requiring digital communication signals, i.e. additional wiring between the individual converters. A method to perform current sharing based on individual, smart power converters performing an optimization based on optimal current levels has been proposed, analysed and tested. The resulting stochastic system results in an efficiency comparable to existing multi-phase converters which has not been achieved with independent, parallel power supplies to date.
6. Conclusions

Digitally controlled power supplies are becoming more and more popular in today’s market. Costs, system level integration and more advanced control algorithms have been identified among the main drivers. New technologies and requirements are continuously emerging and push the designs closer and closer to the physical boundaries. Several new design solutions addressing different issues have been investigated during the course of this thesis and are now summarized.

In chapter 2, an automated design procedure for the direct digital design of digital compensators for digitally controlled power supplies has been proposed. It enables the use of real-life design criteria to optimize compensators to application-specific requirements without the need for extensive control expertise. Several different design criteria have been analysed and practical design recommendations are given. The resulting compensators have been tested on an FPGA prototype platform to demonstrate and verify their performance.

In chapter 3, the limitations of existing digital pulse width modulators for multi-phase applications have been analysed. The need for phase shedding and the requirement for faster transient performance have been established. These are addressed with a novel automatic multi-phase digital pulse width modulator (AM-DPWM) which combines improved transient performance, intrinsic limitation of the number of switching actions, phase shedding and scalability. The influence of the proposed modulator on the control loop has been investigated and its performance assessed on a prototype system.

In chapter 4, frequency synchronization and phase alignment for independent parallel power converters without the need for digital communication signals
have been proposed. The perturbances generated by the individual power supplies on the input/output voltage are used by each individual power converter to harvest information about its counterparts. The proposed algorithm utilizes this information to synchronize the switching actions and to align the phases. The concepts has been implemented and tested on an experimental prototype.

In chapter 5, an innovative current distribution scheme for independent power converters has been presented which allows current distribution and phase shedding without the need for digital communication signals. The concept is based on smart power converters where each power converter operates either at an optimal power level or is switched off. This enables the use of phase shedding operation without the knowledge of the total output current. The resulting concept has been analysed in theory and implemented in an FPGA prototype. The efficiency of the proposed system has been measured and compared with existing systems. A good match between predicted and measured efficiencies has been achieved.

While the different chapters cover different contributions, they can also be combined in their application to further enhance the operation of digitally controlled power converters. For instance, the compensator design method detailed in chapter 2 can be combined with either the modulator concept of chapter 3 or the independent converter concepts of chapter 4 and chapter 5. Also, the two concepts for independent power converters presented in chapter 4 and chapter 5 can be integrated into one single system to combine the advantages obtained from phase alignment with the efficiency improvements gained with the use of the proposed current distribution concept.

The findings presented in this thesis have been published in [9, 20, 23–26] and one patent application has been filed.
Opportunities for future work include

- the investigation of the proposed design method for different converter architectures, such as boost and buck-boost converters.
- the extension of the proposed AM-DPWM with active current sharing. More advanced control methods can be investigated to further improve transient performance.
- the investigation of alternative coupling techniques for the phase alignment concept to improve information harvesting and hence the system performance.
- an investigation into the exact limitation of the proposed current distribution scheme. This can be used to improve system performance and efficiency. Additional methods to improve efficiency, such as dead-time optimization or MOSFET gate modulation can also be integrated.
Bibliography


[28] Intersil. ISL6307A Ultra-high bandwidth 6-Phase PWM Controller with 8 Bit VID Code Capable of Precision RDS(ON) or DCR Differential Current Sensing. Feb. 2006.


Lei Hua and Shiguo Luo. “Design Considerations of Time Constant Mismatch Problem for Inductor DCR Current Sensing Method”. In:


Appendix
A. Automated Compensator Design using Real-life Design Criteria

A.1. Nomenclature for GPC

GPC makes extensive use of matrix and vector notations to write polynomials and polynomial equations short and sweet.

The vector notation with underlying arrows describes the vector of consecutive time steps value starting from the given index.

\[ u \leftarrow_k = [u_k, u_{k+1}, u_{k+2}, \ldots]' \]  \hspace{1cm} \text{(A.1)}

\[ u \rightarrow_k = [u_{k+1}, u_{k+2}, u_{k+3}, \ldots]' \]  \hspace{1cm} \text{(A.2)}

Diagonal-constant matrices are commonly called Toeplitz matrices. In general they are quadratic \((n\text{-by-}n)\) and their algebraic definition is \(A_{ii} = a_{i-j}\), leading to

\[ A = \begin{bmatrix}
a_{0} & a_{-1} & a_{-2} & \cdots & a_{-n+1} & a_{-n} \\
a_{1} & a_{0} & a_{-1} & \cdots & a_{-n+2} & a_{-n+1} \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
a_{n-1} & a_{n-2} & a_{n-3} & \cdots & a_{0} & a_{-1} \\
a_{n} & a_{n-1} & a_{n-2} & \cdots & a_{1} & a_{0}
\end{bmatrix}. \]  \hspace{1cm} \text{(A.3)}
To simplify the GPC equation two matrices, $\Gamma_n, C_n$, for a given polynomial

$$n(z) = n_0 + n_1 z^{-1} + n_2 z^{-2} + \cdots + n_m z^{-m} \quad (A.4)$$

are defined as

$$\Gamma_n = \begin{bmatrix} \frac{C_n}{M_n} \end{bmatrix} = \begin{bmatrix} n_0 & 0 & 0 & \cdots & 0 \\
1 & n_0 & 0 & \cdots & 0 \\
n_2 & n_1 & n_0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
n_m & n_{m-1} & n_{m-2} & \cdots & 0 \\
0 & n_m & n_{m-1} & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & \cdots & n_0 \end{bmatrix}. \quad (A.5)$$

For the same polynomial, the Hankel matrix $H_n$ is defined as

$$H_n = \begin{bmatrix} n_1 & n_2 & n_3 & \cdots & n_{m-1} & n_m \\
n_2 & n_3 & n_4 & \cdots & n_{m-1} & n_m \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
n_{m-1} & n_m & 0 & \cdots & 0 & 0 \\
n_m & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & 0 & 0 \end{bmatrix}. \quad (A.6)$$

The elements of these matrices are defined as

$$\Gamma_n(i, j) = C_n(i, j) = n_{i-j} \quad (A.7)$$

and

$$H_n(i, j) = n_{i+j-1} \quad (A.8)$$

where $n_i = 0$ for $i < 0$ or $i > m$. 

147
The dimension of the matrices are not defined with the previous definitions and depend on the requirements of the specific equations they are used in. The following rules should be applied:

1. $\Gamma_n$ should have $m$ more rows than columns, but can have any number of columns.
2. $C_n$ is always square but can be any dimension.
3. $H_n$ has at least $m$ non-zero rows and columns. More zero rows and columns can be added without changing its operation.
A.2. GPC design algorithm

A function is used to implement the GPC design algorithm used in the automated design procedure.

```matlab
function [comp] = GPCDesign(Pz, ny, nu, lambda)

% add integral action to plant model
[num, den, Tsample] = tfdata(Pz,'v');
int = tf([1 0],[1 -1], Tsample);

% get plant and its order
[num, den, Tsample] = tfdata(Pz*int,'v');
ord=length(den);

% build Toeplitz matrices
num_padded = [num(2:end) zeros(1,ny)];
den_padded = [den(1:end) zeros(1,ny)];
Ca = zeros(ny);
Czb = zeros(ny);
for i=1:1:ny
    for j=1:1:i
        Ca(i,j) = den_padded(i-j+1);
        Czb(i,j) = num_padded(i-j+1);
    end
end

% build Hankel matrices
num_padded = [num(2:end) zeros(1,ny)];
den_padded = [den(2:end) zeros(1,ny)];
Ha = zeros(ny);
Hzb = zeros(ny);
for i=1:1:ord
    for j=1:1:ord
        Ha(i,j) = den_padded(i+j-1);
    end
end
```

% [comp] = GPCDesign(Pz, ny, nu, lambda)
% (C) Copyright by Simon Effler, CSRC, University of Limerick
% Implementation of an compensator design method based on
% Generalized Predictive Control
% See Rossiter, Model based predictive control
%
Hzb(i,j) = num_padded(i+j);

end

end

Ha = Ha(:,[1 : ord-1]);
Hzb = Hzb(:,[1 : ord-2]);

GPC design technique
H = Ca^-1*Czb;
P = Ca^-1*Hzb;
Q = -Ca^-1*Ha;
H = H(:,1:nu);
Dk = (H'*H+lambda*eye(nu))^-1*H'*P;
Nk = (H'*H+lambda*eye(nu))^-1*H'*Q;

% extract compensator coefficients
num = Nk(1,:);
den = [1,Dk(1,:)];

% add integral action
comp = tf(num,den,Tsample) * int;

return
B. AM-DPWM

B.1. Prototype schematic excerpts
Figure B.1: Schematic of the prototype system: Power stage
Figure B.2.: Schematic of the prototype system: Current sensing
Figure B.3.: Schematic of the prototype system: Output voltage sensing
Figure B.5.: PCB of the prototype system: Top view

Figure B.6.: PCB of the prototype system: Bottom view
C. Current Distribution Scheme for Independent Power Supplies based on Efficiency Levels

C.1. Efficiency measurements
**Figure C.1.**: Measured efficiency variation for repetitive measurements with constant load current steps.

**Figure C.2.**: Measured efficiency variation for repetitive measurements with constant load current steps (time lines).
**Figure C.3.** Measured efficiency variation for repetitive measurements with arbitrary load current steps.

**Figure C.4.** Measured efficiency variation for repetitive measurements with arbitrary load current steps (time lines).