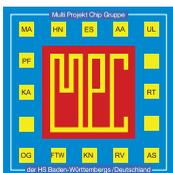


MPC

MULTI PROJEKT CHIP GRUPPE
BADEN - WÜRTTEMBERG

Herausgeber: Hochschule Ulm Ausgabe: 48 ISSN 1868-9221 Workshop: Aalen Juli 2012

- 1 Processor Solutions for Smart Mobility**
J. Becker, O. Sander, C. Roth, Karlsruhe Institute of Technology (KIT)
- 9 Entwicklung einer CMOS Pixelzelle zur Charakterisierung von Peptid-basierten Fotodioden**
D. Seiz, H. Richter, IMS Chips Stuttgart
- 17 Gate Driver Design for Fast Switching DC-DC Converters**
J. Wittmann, T. Funk, B. Wicht, HS Reutlingen
- 25 Realization of an RFID Front End IC for ISO 15693 Standard in UMC CMOS 0.18 μm Technology**
M. Bhattacharyya, T. Volk, A. Kreker, B. Dusch, D. Jansen, HS Offenburg
- 33 Hardware-Entwurf einer flexiblen Fehlerkorrekturereinheit für Flashspeicher**
J. Spinner, J. Freudenberger, HTWG Konstanz
- 41 Measurement Unit for Energy Efficient M2M Mobile Communication**
M. Klemm, A. Sikora, HS Offenburg
- 47 Soft-Core-Prozessor mit Multithreading für DSP-Anwendungen**
C. Kielmann, I. Schoppa, HTWG Konstanz
- 53 TCL-Skript basiertes Backend Design mit Cadence Encounter**
S. Thiel, H.-P. Bürkle, HS Aalen
- 57 Parametrisierte Layout-Module im analogen IC-Entwurf**
S. Gohm, D. Marolt, J. Scheible, HS Reutlingen



Cooperating Organisation
Solid-State Circuit Society Chapter
IEEE German Section



Inhaltsverzeichnis

Processor Solutions for Smart Mobility	1
J. Becker, O. Sander, C. Roth, Karlsruhe Institute of Technology (KIT)	
Entwicklung einer CMOS Pixelzelle zur Charakterisierung von Peptid-basierten Fotodioden	9
D. Seiz, IMS Chips Stuttgart H. Richter, Universität Stuttgart	
Gate Driver Design for Fast Switching DC-DC Converters	17
J. Wittmann, T. Funk, B. Wicht, HS Reutlingen	
Realization of an RFID Front End IC for ISO 15693 Standard in UMC CMOS 0.18 μm Technology	25
M. Bhattacharyya, T. Volk, A. Kreker, B. Dusch, D. Jansen, HS Offenburg	
Hardware-Entwurf einer flexiblen Fehlerkorrekturereinheit für Flashspeicher	33
J. Spinner, J. Freudenberger, HTWG Konstanz	
Measurement Unit for Energy Efficient M2M Mobile Communication	41
M. Klemm, A. Sikora, HS Offenburg	
Soft-Core-Prozessor mit Multithreading für DSP-Anwendungen	47
C. Kielmann, I. Schoppa, HTWG Konstanz	
TCL-Skript basiertes Backend Design mit Cadence Encounter	53
S. Thiel, H.-P. Bürkle, HS Aalen	
Parametrisierte Layout-Module im analogen IC-Entwurf	57
S. Gohm, D. Marolt, J. Scheible, HS Reutlingen	

Tagungsband zum Workshop der Multiprojekt-Chip-Gruppe Baden-Württemberg
Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie.

Die Inhalte der einzelnen Beiträge dieses Tagungsbandes liegen in der Verantwortung der jeweiligen Autoren.

Herausgeber: Gerhard Forster, Hochschule Ulm, Prittwitzstraße 10, D-89075 Ulm

Alle Rechte vorbehalten

Diesen Workshopband und alle bisherigen Bände finden Sie im Internet unter:

<http://www.mpc.belwue.de>

Processor Solutions for Smart Mobility

Jürgen Becker, Oliver Sander, Christoph Roth

Abstract—The term **Smart Mobility** denotes the vision of extensive information dissemination between mobile entities as well as the environmental infrastructure with the goal of making mobility more safe, efficient, environmentally friendly and comfortable. Therefore, relevant information of current situations will be deployed and processed in realtime by all traffic participants. Realizing future smart mobility applications is in fact directly and technological inevitably linked to the provision of sufficient processing power within the individual sub-systems. Within this paper, we class smart mobility into the large research field of cyber-physical systems and discuss requirements and challenges for realizing smart mobility by means of appropriate hardware/software architectures. We present and excert of completed and ongoing research projects at ITIV in the field of heterogeneous system architecture development and give an outlook to worthwhile research directions.

Index terms—Cyber-physical systems.

I. INTRODUCTION

Smart Mobility solutions are essential to meet the increasing demands on travel and transportation systems. Green mobility, accident-free driving, support for elderly people, adaptive route planning to the point of 4D harmonization, all are catchphrases that can be allocated to the topic of smart mobility. By integrating entities from different mobility domains like automotive, railway or avionics into a spanning communication network that connects them with each other as well as the environmental ICT infrastructure, safety and non-safety relevant information can be disseminated, which promises improvements in situation awareness, traffic management and active safety. E.g. looking at the automotive domain, vehicle-to-vehicle communication (V2VC) technology promises to be a next major step towards realization of smart mobility and thus safer and more efficient road traffic. V2VC

uses so called Vehicular Ad hoc NETWORKS (VANETs), that are highly dynamic sensor-actuator-networks for information exchange with vehicles as nodes. Considering the vision of the so called Cyber-Physical Systems (CPS) this trend of intercommunication and cross-linkage will continue or rather become much more intense in future.

In general, due to the increasing complexity of applications and functions for smart mobility, power and performance requirements on the underlying hardware/software system also increase. Feasibility of smart mobility is in fact directly and technological inevitably linked to the realizable processing power of the individual mobile subsystems. One major trend that can be observed in this context, is the emerging popularity of embedded multicore systems which is driven by the need to handle complexity and optimize performance per watt through integrating evermore functionality directly into a single device. Especially, heterogeneous architectures based on reconfigurable hardware structures can offer an alternative to common standard solutions with respect to power and performance. However, the integration of applications and functions of mixed criticality onto a single multicore device with respect to certifiability is a non-solved problem up to now.

In this contribution, the necessity for novel system architecture solutions in the context of smart mobility is outlined. Thereby, different examples of innovative heterogeneous system architectures that resulted from research projects conducted at ITIV or that are still under development at ITIV, are presented. These illustrate the large solution space in architecture design and motivate the development of architecture patterns which allow classifying architectural characteristics according to their necessity for certain functions.

The remainder of this paper is organized as follows: Section II classes the term smart mobility into the larger context of cyber-physical systems. In section III first, key technologies are identified that are necessary for realizing CPS and smart mobility. After that, examples of heterogeneous system architectures in general as well as in the automotive context are presented. Also, methods for verification and validation of smart mobility systems are shortly outlined. Finally, in section IV we conclude and give an overview on current and future research directions.

Prof. Dr. Jürgen Becker, becker@kit.edu; Dr. Oliver Sander, oliver.sander@kit.edu; and Christoph Roth, christoph.roth@kit.edu are with Karlsruhe Institute of Technology (KIT), Institute for Information Processing Technology (ITIV), Engesserstraße 5, D-76131 Karlsruhe, Germany.

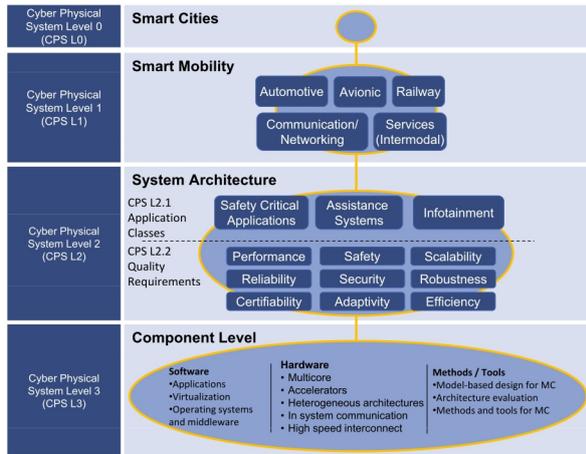


Fig. 1: Cyber-physical system levels.

II. THE CYBER-PHYSICAL SYSTEM CHALLENGE IN THE CONTEXT OF SMART MOBILITY

A. Definition of CPS

Although cyber-physical systems are being discussed for some years now, no broadly accepted definition exists. However two major streams can be recognized. In the U.S. the discussion focuses on the understanding of physical and system interactions, while in Europe the discussion is more centered around the networking aspect of such systems.

In the following we refer to the CPS description which has been proposed by the ACATECH initiative AgendaCPS [1], [2].

Cyber-physical systems are systems with embedded software (as part of devices, buildings, means of transport, transport routes, production systems, medical processes, logistic processes, coordination processes and management processes), which:

- directly record physical data using sensors and affect physical processes using actuators;
- evaluate and save recorded data, and actively or reactively interact both with the physical and digital world;
- are connected with one another and in global networks via digital communication facilities (wireless and/or wired, local and/or global);
- use globally available data and services;
- have a series of dedicated, multimodal human-machine interfaces.

B. Levels in Smart Mobility

Cyber-physical systems can be seen hierarchical with several CPS forming a more complex CPS. In the other direction a CPS can be split up into its building blocks. This is done in the following for the Cyber-Physical System Smart City (see also Fig. 1). We

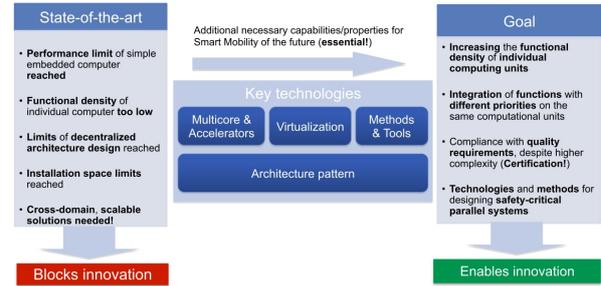


Fig. 2: Incorporation of key technologies.

define Smart Cities as CPS Level 0. One of the CPSes which is a part of Level 0 is the CPS Smart Mobility (CPS Level 1) which forms the next layer. On this level there is a part where all transport possibilities that belong to the most efficient route are actively used (intermodal transportation). This includes trains, trams, buses, planes, cars, car-sharing and the like. This can be seen as integrated system on CPS Level 1.1. CPS Level 1.1 comprises different domains like avionics, automotive or railway. These are denoted as Level 1.2.

On the next layer (CPS Level 2) the system architecture and its aspects are defined. Level 2.1 describes the different application classes (Safety Critical, Assistance Systems and Infotainment). Naturally all applications have different functional and quality requirements which need to be fulfilled in order to successfully perform the functionality. Fig. 1 gives some examples (performance, safety, security etc.). Finally on CPS Level 3 the components of the system are defined. These include hardware, software as well as methodologies and tools.

Of course not all aspects which are necessary to build the aforementioned CPSes are shown in Fig. 1. However considerable computing performance is needed when ad-hoc networking, realtime and security requirements are considered. Hence hardware platforms provide the basis for the realization of smart mobility and smart cities or even more complex CPS.

III. TOWARDS CPS REALIZATION ON NODE LEVEL

A. Key Technologies on Node Level

Currently, state-of-the-art technologies in the embedded domain are characterized by different shortcomings that essentially contribute in blocking the innovation potential (see Fig. 2). E.g., the evermore increasing complexity of applications in combination with the low functional density of current embedded computers results in the attainment of performance limits of current embedded system architectures. Beside that, limits of decentralized architecture design are more and more exhausted, due to reasons like the increasing effort for managing execution of distributed functions in a timely correct manner or the attainment

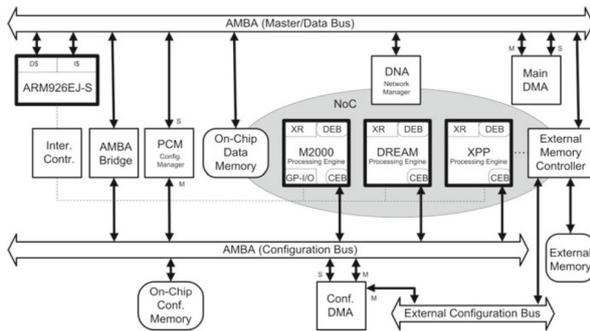


Fig. 3: MORPHEUS architecture.

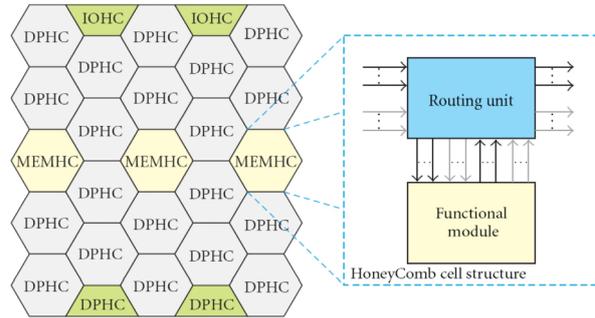


Fig. 4: HoneyComb architecture.

of installation space limits. Therefore, goal of current research is the development of technologies and methods, that allow bridging the gap between the high performance requirements of new functions and the shortcomings of currently available technologies. The key technologies which have to be incorporated are heterogeneous multi-core systems, virtualization technology as well as design methods and tools. Successful incorporation of key technologies finally leads to the goal of highly integrated systems which are capable of running mixed critical functionalities in terms of safety and security.

The approach we are targeting, is the development of so called *architecture patterns* in the context of distributed embedded computing nodes that fulfill the requirements which arise from future CP Systems. The major challenge is to find patterns which are abstract enough to link very different applications to a wide variety of concrete hardware/software platforms. This way a methodology arises, that forms the fundament for combining different architectural components of different granularity to new heterogeneous system architectures (e.g. heterogeneous node, network of heterogeneous nodes, etc.) while complying with quality requirements for example regarding safety.

B. Heterogeneous Architectures Research

From the CPS perspective, different architecture components on CPS L3 are combined to a complex system architecture on CPS L2 for execution of safety critical or non-safety critical applications. Components can be either hardware or software. In both cases, various granularities are imaginable, e.g. a conceivable classification of software artifacts according to the possible algorithmic complexity could be the division into instructions, tasks, applications or operating systems. Similarly, hardware artifacts could be divided into different subcomponents. Probably the most important part are processing elements like singlecore, multicore, reconfigurable hardware, GPUs or complex System-on-Chip devices.

A main aspect of the research at ITIV is the development and investigation of novel domain-specific as well as domainspanning System-on-Chip architectures made up of diverse homogeneous and heterogeneous

(meaning fine-grained and/or coarse-grained) components as well as their evaluation with regard of their benefit for different functions and applications. Thereby, a special focus lies on the consideration of reconfigurable and dynamically reconfigurable architectures. From this point of view, several research projects funded by the European Union, the Federal Ministry of Education and Research as well as the German Research Foundation have been conducted and participated.

Within the European MORPHEUS [3] project new concepts for flexible *domain focused platforms* have been investigated which are positioned between general purpose flexible hardware and general purpose processors, and provide breakthroughs in performance and cost-effectiveness for embedded computing systems. The project focused on the challenges that come along with rising complexity in combination with the enlarging design productivity gap. The goal was the development of a global solution based on a modular heterogeneous System-on-Chip (SoC) platform (see Fig. 3) providing the disruptive technology of dynamically reconfigurable computing and completed by a software oriented design flow and a consistent tool-set. The project provided a modular silicon demonstrator in 90nm technology composed of complementary run-time reconfigurable building blocks.

The goal of the research project AMURHA was the development of an adaptive multi-grained reconfigurable hardware architecture for dynamic function patterns. The outcome was a platform called HoneyComb (HC) [4] that consists of a coarse-grained reconfigurable array designed to compute stream-based as well as control-based applications. Therefore, in addition to coarse-grained components, the architecture includes fine grained components, which are used to compute Boolean operations and control program execution. To support design-time adaption of the architecture to application requirements, in addition to the dynamic reconfiguration, compiler-supported application-tailored hardware reduction/RTL adaption techniques have been developed and implemented. The HC architecture has a hexagonal cell layout (see Fig. 4). Each cell is either a data-path module (DPHC), a I/O module (IOHC) or a memory module (MEMHC). *Invasive*

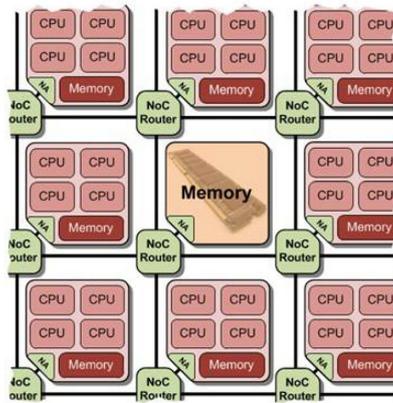


Fig. 5: Invasive Network-on-Chip (iNoC).

Computing is a novel paradigm for designing and programming future parallel computing systems. The main idea of invasive computing is to introduce resource-aware programming support in the sense that a given program gets the ability to explore and dynamically spread its computations to neighbor processors in a phase called invasion, then to execute portions of code of high parallelism degree in parallel based on the available invisable region on a given multi-processor architecture. Afterwards, once the program terminates or if the degree of parallelism should be lower again, the program may enter a retreat phase, deallocate resources and resume execution again, for example, sequentially on a single processor. In order to support this idea of self-adaptive and resource-aware programming, not only new programming concepts, languages, compilers and operating systems are necessary but also revolutionary architectural changes in the design of MPSoCs (Multi-Processor Systems-on-a-Chip) must be provided so to efficiently support invasion, infection and retreat operations involving concepts for dynamic processor, interconnect and memory reconfiguration [5]. Invasive computing is treat in the DFG Transregio Project InvasIC. Thereby, ITIV is responsible for the investigation of mechanisms that allow utilization of adaptivity based on runtime reconfiguration of the instruction set architecture (ISA) as well as the microarchitecture (uArch). Beside that, ITIV is involved in the development and exploration of an adaptive and invasive on-chip communication network (a so called iNoC, see Fig. 5) focussing on the three main aspects a) protocol definition b) runtime prediction and c) decentralized strategies for application specific communication patterns.

The goal of the BMBF project ARAMiS (Automotive, railway, and avionics multicore systems) [6] is to provide the technological basis for a further increase of safety, traffic efficiency and comfort in the mobility domains automobile, avionics and railway through the exploitation of multicore technology in combination with virtualization technology. One of the major challenges is the deployment of such systems with respect

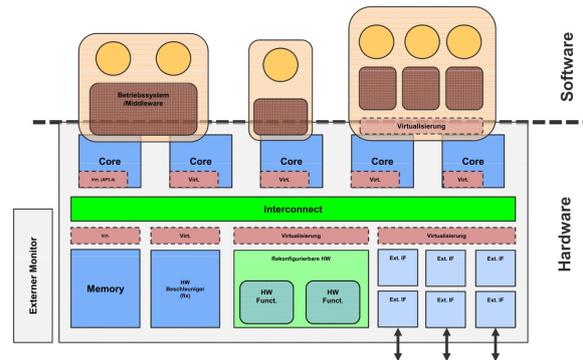


Fig. 6: Virtualized multicore platform.

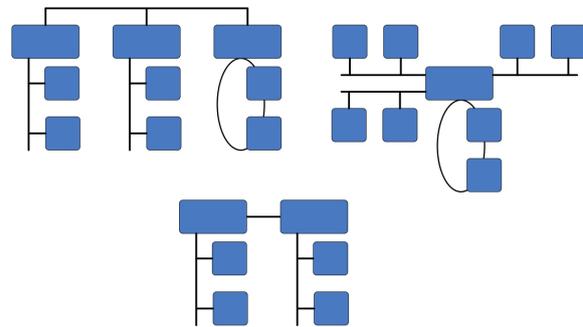


Fig. 7: Backbone architecture (upper left), central architecture (upper right) and front/rear architecture (below).

to safety critical and mixed critical applications. Fig. 6 exemplarily shows an embedded platform architecture that uses hardware and software virtualization in combination with multiple processing cores. A major role within such heterogeneous platforms play so called monitors which allow observing the current system state and the initiation of emergency routines in the case that e.g. realtime requirements of a function could not be fulfilled. The insights gained through the ARAMiS project form the inevitable fundament for the successful interconnection of embedded systems to cyber-physical systems. Against this background the project contributes one's share to the preservation and reinforcement of the worldwide competitiveness of the German industries within the domains automobile, avionics and railway.

C. Examples of Automotive E/E Architectures

Looking at the automotive domain, a wide spectrum of E/E architecture designs is conceivable. Their manifestation or the manifestation of one of their sub-networks depends on specific requirements of the targeted application/function as well as given environmental conditions of the vehicle in which the E/E architecture is embedded. Specifically, timing behavior is one of the key criteria for sophisticated design decisions. Typically, three major characteristic forms

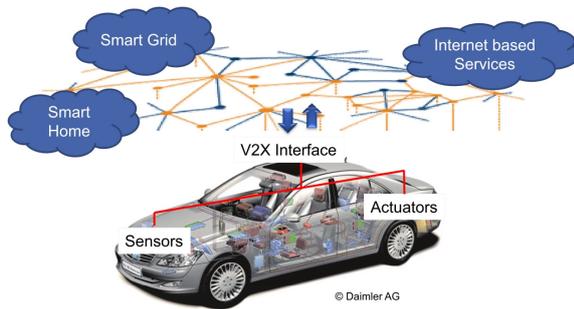


Fig. 8: Vehicle as a complex sensor/actuator node.

are distinguished on the E/E architecture toplevel, *backbone architecture*, *centralized architecture* and *front/rear architecture* (see Fig. 7). The larger blocks represent domain gateways which route information between the different bus segments that are existent in the overall E/E architecture. In the case of a centralized architecture only a single gateway is available that connects all bus segments. The central gateway approach has the advantage of information from all bus segments being directly accessible within a single control unit without additional latency, introduced e.g. by a backbone connection. As described below, this kind of topology is advantageous when opening the internal vehicular network to an outer Vehicular Ad hoc NETWORK (VANET).

D. E/E Architecture Design for V2XC

Vehicle-to-X Communication (V2XC) based on a VANET forms the next major step for extending the automotive domain towards smart mobility. V2XC promises improvements in situation awareness, traffic management and active safety based on disseminated information. A VANET is a highly dynamic sensor-actuator-network with vehicles as nodes. Depending on the density of equipped vehicles, a varying number of geographically distributed VANETs may merge or separate depending on the participant's movements.

Up to now, most research in the V2XC domain ends up in describing protocols, application analysis and standardization. Only limited effort is put into realization and system architecture aspects. However, considering vehicles as a simple network node while neglecting the fact of up to around 100 devices communicating inside which cause highly diverse internal latency times and throughput rates, is not enough for getting a holistic view. Beside predictable latency times, trustworthiness of messages is a major aspect since safety relevant decisions have to be based on remotely collected information to overcome line-of-sight limitations. Security measures based on cryptographic algorithms have to be employed which typically demand for a huge amount of computational power. In Fig. 8 these facts are illustrated by means of

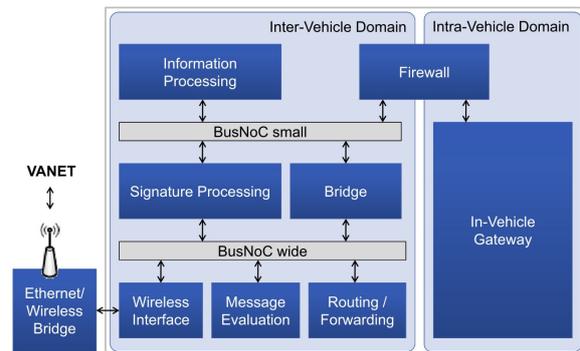


Fig. 9: V2X-gateway architecture.

a single vehicle that is connected through a V2X interface to a smart environment which allows data exchange between external and internal network consisting of a complex interconnection structure of sensor and actuator systems.

In this context we developed a FPGA-based V2X gateway architecture (see Fig. 9) [7] which follows the last North American and European standardization efforts [8]-[10]. By integration of V2X functionality into a central automotive gateway, a direct link between the VANET network and the internal vehicular bus network is provided. The approach opens the internal automotive E/E architecture to the outer VANET. Therefore, the central gateway is divided into an Inter-Vehicle-Networking-Domain (InterVND) and an Intra-Vehicle-Networking-Domain (IntraVND). The main aspects which make this approach different are as follows:

- *Modularization*: Typically occurring tasks are distributed on specific modules. These work in parallel and allow processing of V2X messages in discrete steps. Each of the modules consists of different hardware and software components and fulfill a dedicated task like e.g. message evaluation, routing, signature generation/verification or information processing. Thus the overall gateway itself forms a heterogeneous multicore device. Self-X characteristics have been embedded into the system wherever possible e.g. by utilizing partial dynamic reconfiguration.
- *Communication Architecture*: Packets are passed from one module to the successor in the processing chain by means of a realtime capable on-chip packet-based transmission bus system [11]. Especially in situations like traffic jam when the own vehicle is surrounded by dozens of other vehicles and the system runs near maximum capacity, a fast forwarding of crucial information must be guaranteed in order to comply with the strict latency requirements. Accelerating packet propagation is achieved by prioritized packet forwarding through prioritized scheduling and prioritized re-

ception buffers. This allows higher priority packets to overtake less important ones.

- **Signature Processing:** As shortly introduced above, V2XC calls for trustworthy information. As we cannot assure the correctness of sensor values obtained by remote vehicles, we assure at least trustworthiness of the data by checking properties of the issuing platform, i.e. the sending on-board unit (OBU). This is done using signatures and certificates. Security functions are pooled in a special security module [12],[13]. It performs signature and certificate verifications for all incoming packets and generates signatures for all outgoing ones. Based on the IEEE WAVE security standard [8] the Elliptic Curve Digital Signature Algorithm (ECDSA) is implemented. To achieve the required throughput, the processing is completely done in specialized cryptographic hardware and encapsulated in the module that can easily be duplicated if there is need for more throughput.
- **Separation of Domains:** By principal, V2XC opens the internal architecture to outside communication. Hence a malicious attacker may attempt Denial-of-Service (DoS) attacks to influence the overall vehicle's behavior. Considering safety relevant functions this may result in serious accidents or even dead passengers. So protecting the cars functionality is mandatory for each V2XC OBU. FPGAs offer the unique opportunity to physically divide Inter-VND and IntraVND and connect them via a distinguished interface (fire-wall). In our approach a dual ported memory block serves as connection between both domains. The memory block is split into two disjoint regions. The two domains have write access to one of the not-overlapping blocks but are able to read both blocks. On each side the respective module decides whether to process the incoming data or not. Especially the IntraVND decides when to take data offered by the InterVND. Additionally high level protocols can pass the firewall in another region. As the in-car environment is considered to be trustworthy, information from this side is directly given to the InterVND without additional check.

E. Simulation-based Validation of V2XC Systems

Since V2XC systems have to fulfill a lot of different requirements concerning high performance and functionality, an intensive evaluation and investigation is necessary in order to be able to optimize the overall system architecture. From a more abstract perspective, a V2XC system interacts with the two domains *network* and *environment*. The network domain provides stimulus in terms of network packets and the environmental domain provides stimulus in terms of sensor data.

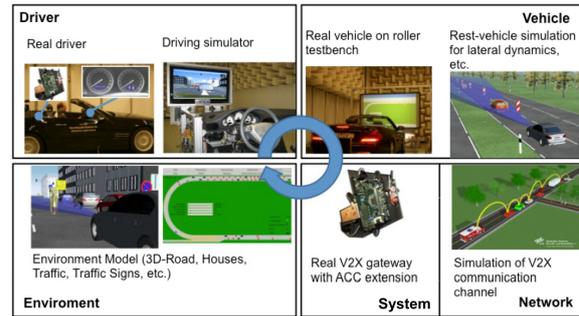


Fig. 10: V2X-ACC-in-the-loop.

In this context, the simulation-based approach proposed in [14] allows a flexible adaption to the test case by being able to interconnect an arbitrary number of simulators of the different domains system, network and environment. It targets comprehensive design space exploration, verification, and test of V2XC system models. Therefore several SystemC based architecture models can be integrated into a simulation framework that allows observation of system behavior and interaction.

In [15] the approach is extended towards capabilities for more continuous and holistic considerations. The ideas from [14] are included into the so called V2X-in-the-Loop platform. The base frame of the platform is given by the so called X-in-the- Loop framework [16] which describes a consistent and integrated development environment for drive systems. Thereby, "X" stands for the unit under test (UUT) which is in our case the complete vehicle equipped with the V2XC system. The V2X-in-the-Loop platform extends the environmental domain by the domains vehicle and driver. Generally, each of them can be either virtual or real which allows evaluating and investigating interactions and interaction chains between real as well as virtual and real systems and sub-systems.

In Fig. 10 an instance of the platform is exemplarily illustrated that is used for evaluating the impact of a V2XCbased adaptive cruise control system (ACC) on the braking behavior of a real driver. Thereby, the driver is sitting in the real vehicle which is running on a roller test-bench or controls it via a driving simulator. Both, real driver and real vehicle are embedded into a virtual environment that simulates surrounding traffic on microscopic level. Furthermore, the real wireless network channel of the vehicle is connected to a virtual wireless network communication channel that allows V2XC between real and virtual vehicles.

IV. CONCLUSION AND OUTLOOK

Single-core based processor solutions are not enough in order to cope with the performance requirements of future general purpose and safety-critical applications. New multicore solutions are necessary to accelerate future applications. However,

simply raising the number of cores is not enough in order to cope with performance and safety requirements. Instead, new heterogeneous multicore systems that provide dedicated programmable hardware units and structures are essential. Beside that, the combination of mixed-critical applications on single platforms demands for certifiable architecture designs. IP-cores and corresponding systems have to be reliable and self-adaptive due to possible errors and malfunctions during runtime. Still an open question is, how to combine all available *key technologies* best? The development of novel architecture patterns may help to solve this question and will possibly be the basis for future hardware/software co-design standards and methodologies.

REFERENCES

- [1] E. Geisberger and M. Broy, editors. *agendaCPS: Integrierte Forschungsagenda Cyber-Physical Systems*. acatech Studie. Springer, Berlin, March 2012.
- [2] O. Sander, B. Glas, C. Roth, J. Becker, and K. D. Müller-Glaser. Cyber-physical systems, driving force for innovation in mobility, health, energy and production. *ACATECH Position Paper, ACATECH, 2011*, Dec. 2011.
- [3] A. Grasset, P. Millet, P. Bonnot, S. Yehia, W. Putzke-Roeming, F. Campi, A. Rosti, M. Huebner, N. Voros, D. Rossi, H. Sahlbach, and R. Ernst. The morpheus heterogeneous dynamically reconfigurable platform. *International Journal of Parallel Programming*, 39:328–356, 2011.
- [4] A. Thomas, M. Ruckauer, and J. Becker. Honeycomb: A multi-grained dynamically reconfigurable runtime adaptive hardware architecture. In *SoCC*, pages 335–340, 2011.
- [5] J. Teich, J. Henkel, A. Herkersdorf, D. Schmitt-Landsiedel, W. Schröder-Preikschat, and G. Snelting. Invasive computing: An overview. In M. Hübner and J. Becker, editors, *Multiprocessor System-on-Chip – Hardware Design and Tool Integration*, pages 241–268. Springer, Berlin, Heidelberg, 2011.
- [6] Automotive, Railway and Avionics Multicore Systems - ARAMiS. <http://www.projekt-aramis.de/index.php>.
- [7] O. Sander, B. Glas, C. Roth, J. Becker, and K. Müller-Glaser. Design of a vehicle-to-vehicle communication system on reconfigurable hardware. In *Field-Programmable Technology, 2009. FPT 2009. International Conference on*, pages 14–21, Dec. 2009.
- [8] IEEE. Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE). *IEEE Std 1609*, 2006.
- [9] SAE. Dedicated Short Range Communications (DSRC) Standard Draft. *SAE Standard J2735*, 2006.
- [10] COMeSafety Project. European ITS Communication Architecture - Overall Framework, 2008. available at www.comesafety.org.
- [11] O. Sander, B. Glas, C. Roth, J. Becker, and K. D. Müller-Glaser. Priority based packet communication on a bus-shaped structure for FPGA-systems. In *DATE*, 2009.
- [12] B. Glas, O. Sander, V. Stuckert, K. D. Müller-Glaser, and J. Becker. Car-to-car communication security on reconfigurable hardware. In *VTC2009-Spring: Proceedings of the IEEE 69th Vehicular Technology Conference, Barcelona, Spain, 2009*.
- [13] B. Glas, O. Sander, K. Müller-Glaser, and J. Becker. Echtzeitfähige car-to-x-Kommunikationsabsicherung und e/e-Architekturintegration. In *ATZ Elektronik, Ausgabe 01/2012*, 2012.
- [14] C. Roth, O. Sander, M. Hübner, and J. Becker. Car-to-x simulation environment for comprehensive design space exploration verification and test. In *SAE Int. J. Passeng. Cars - Electron. Electr. Syst.* 3(1):17-26, 2010, doi:10.4271/2010-01-0451, 2010.
- [15] A. Albers, T. Düser, O. Sander, C. Roth, and J. Henning. X-in-the-loopframework für Fahrzeuge, Steuergeraete und Kommunikationssysteme. In *ATZ Elektronik, Ausgabe 05/2010*, 2010.
- [16] A. Albers, T. Düser, and S. Ott. X-in-the-loop als integrierte Entwicklungsumgebung von komplexen Antriebssystemen. In *8. Tagung Hardware-in-the-Loop-Simulation, Haus der Technik, Kassel*, 2008.

Entwicklung einer CMOS Pixelzelle zur Charakterisierung von Peptid-basierten Fotodioden

Daniel Seiz, Harald Richter

Zusammenfassung—Im Rahmen des von der Europäischen Union geförderten Projekts „Peptide-based diodes for solar cells“ sollen eine große Anzahl verschiedener Peptid-Strukturen mit Hilfe eines Chips hinsichtlich ihrer Eigenschaften als Diode charakterisiert werden [1]. Um möglichst viele unterschiedliche Peptide untersuchen zu können, muss die Möglichkeit bestehen, die Messungen zu parallelisieren. Hierzu wird eine Matrix-Anordnung der Sensoren verwendet. Für den geeignetsten Lösungsansatz wird eine vollständige Schaltung sowie ein entsprechendes Layout vorgestellt.

Schlüsselwörter—CMOS, Peptid, Diode, Strommessung, Operationsverstärker.

I. GRUNDLAGEN

Um eine Solarzelle zu erzeugen, bei der die Energieumwandlung genauso effizient vonstattengeht wie bei der pflanzlichen Photosynthese, sollten sich die Elektronen über sehr kurze Distanzen durch hochgeordnete Strukturen ohne Haftstellen bewegen. Angeregt durch dieses Prinzip sollen am Karlsruher Institut für Technologie (KIT) viele verschiedene hochgeordnete Peptidstrukturen (organische Verbindung aus mehreren Aminosäuren [2]) erzeugt und auf ihre Funktionstüchtigkeit als Diode untersucht werden. Verbunden mit lichtempfindlichen organischen Bausteinen soll eine neuartige Solarzelle entstehen, die auf biologisch inspirierten Prinzipien basiert, sehr effizient arbeitet und kostengünstig herzustellen ist. [1]

Um diese Dioden herzustellen, werden verschiedene Peptid-Bausteine kombiniert. Anschließend wird die Richtungsabhängigkeit der Elektronenleitfähigkeit untersucht, um zu ermitteln, welche Peptidstrukturen eine Diodenfunktion aufweisen. Kernstück ist dabei ein Elektronen-tunnelnder Baustein (in Abbildung 1 grün dargestellt). Wird dieser durch lichterntende oder -aussendende Bausteine ergänzt, können auch peptid-basierte Fotodioden bzw. LEDs gebaut werden [3] - [5].

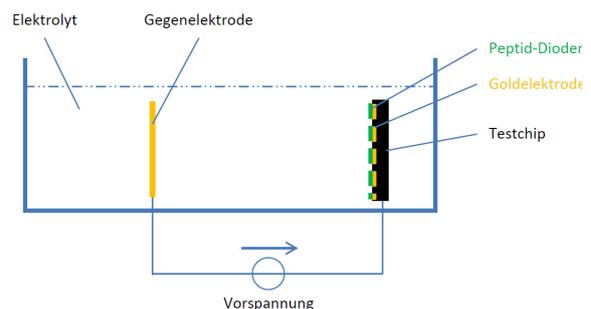


Abbildung 1: Messeinrichtung.

II. ENTWICKLUNG DER SCHALTUNG

Um die Peptidstrukturen auf ihre Eigenschaften zu untersuchen, werden 10.000 Peptide, die nach einem am KIT entwickelten Verfahren synthetisiert werden, in Array-Form auf eine 1 cm² große Chipfläche mit Gold-Pads in einer Matrix-Struktur von 100 x 100 Elektroden aufgebracht. Die auf dem Chip befindlichen Gold-Pads stellen die eine Elektrode dar; die andere wird durch ein Elektrolyt angebunden. Mithilfe des Chips sollen nun die verschiedenen Peptid-Strukturen auf ihre Eigenschaften als Dioden untersucht werden. Hierzu soll es möglich sein, den Dioden eine positive oder negative Vorspannung aufzuprägen und dann den Diodenstrom zu messen. Weiter soll es auch möglich sein, die Veränderungen der jeweiligen Kennlinien unter Lichteinfluss zu erfassen. Abbildung 1 zeigt den vereinfachten Aufbau der Messeinrichtung mit dem Messchip, der Gegenelektrode sowie der Spannungsquelle zur Erzeugung der Vorspannung. [6] Um die einzelnen Peptide charakterisieren zu können, muss auf dem Chip eine individuelle Messung für jedes einzelne Array-Element möglich sein. Hierzu wird eine Struktur verwendet, bei dem für jedes Pixel ein individueller Wert gemessen werden kann. Die Pixel-schaltung muss entsprechend so ausgelegt werden, dass die o.g. Anforderungen erfüllt werden können. Realisiert wird die Schaltung in einem am IMS Stuttgart verfügbaren 0,5 µm CMOS Prozess.

Daniel Seiz; daniel.seiz@gmx.de;
Universität Stuttgart; Postfach 80 11 40; 70550 Stuttgart.

Dr. Harald Richter; richter@ims-chips.de;
IMS Chips; Allmandring 30a; 70569 Stuttgart.

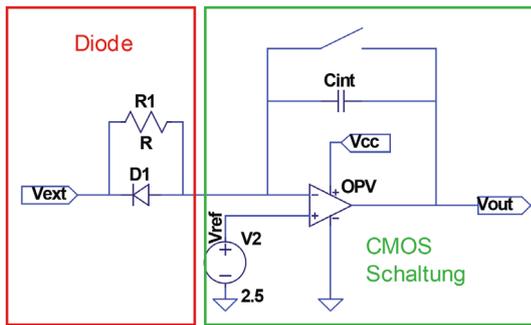


Abbildung 2: Operationsverstärker in Integratorschaltung.

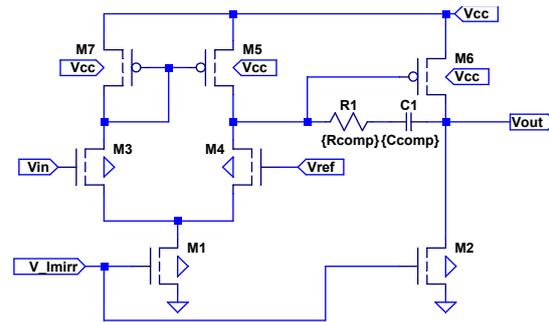


Abbildung 3: Zweistufiger Operationsverstärker.

III. REALISIERUNG DER SCHALTUNG

A. Schaltungsteile

Um die im vorigen Abschnitt genannten Anforderungen zu erfüllen, wurde der in Abbildung 2 gezeigte Schaltungsansatz gewählt. Im Wesentlichen handelt es sich hierbei um einen Operationsverstärker, der als Integrator beschaltet wird. Hierbei wird der Diodenstrom auf einem Kondensator integriert. Die Spannung über der Diode ist über die gesamte Integrationsdauer fest vorgebar. Dies wird durch die Spannungsdifferenz von idealerweise 0 V an den Eingängen des rückgekoppelten Operationsverstärkers ermöglicht. Somit können exakte Messungen durchgeführt werden, auch der Messbereich der Integrationskapazität wird voll ausgenutzt. Für den gemessenen Strom gilt:

$$I_{\text{mess}} = \frac{U_C \cdot C_{\text{int}}}{t_{\text{int}}}$$

Als nachteilig ist bei dieser Schaltung der erhöhte Schaltungsaufwand durch den Operationsverstärker sowie die höhere Stromaufnahme zu benennen (im Vergleich zu einer Active Pixel Sensor Schaltung, wie sie z.B. in Bildsensoren Verwendung findet; vgl. [7]). Auch ist die Integrationskapazität schwieriger zu fertigen als bei der linearen APS-Schaltung, da beide Pole des Kondensators unabhängig von der Versorgungsspannung sein müssen. Dennoch überwiegen die Vorteile dieser Schaltung, so dass auf Grundlage dieses Ansatzes eine entsprechende Schaltung entworfen wurde.

Der Operationsverstärker wurde mit zwei Verstärkungsstufen aufgebaut. Der gegenüber einer dreistufigen Ausführung höhere Ausgangswiderstand und die etwas geringere Verstärkung sind für diesen Anwendungszweck nicht problematisch. Der Verstärker besteht aus einer Differenzverstärkerstufe und einer zweiten Stufe in Drain-Schaltung.

Die Differenzverstärkerstufe besteht aus den N-Kanal-Eingangstransistoren M3 und M4. Die P-Kanal-Transistoren M5 und M7, die als Stromspiegel verschaltet sind, bilden eine aktive Lastschaltung und

wandeln die differentielle Ausgangsspannung in eine unsymmetrische Ausgangsspannung [8] – [10]. Die zweite Stufe besteht aus sechs parallel geschalteten P-Kanal-Transistoren in Source-Schaltung. Der Arbeitspunkt der zweiten Stufe wird über eine Stromspiegelschaltung mit fünf parallelen Transistoren eingestellt. Der Stromspiegel benötigt eine externe Referenzspannung, um den Strom durch die Verstärkerschaltung einzustellen. Diese Referenzspannung muss somit nur ein Mal pro Zeile im Array (oder noch weniger) erzeugt werden.

Um eine möglichst hohe Verstärkung zu erhalten, wurde die Transistorgeometrie optimiert. Hierbei wurde die Länge der Transistoren variiert; die Weite blieb an die vorgegebene Grundstruktur aus der GATE-FOREST-Architektur [11] des IMS angepasst. So wurde die Kanallänge der Stromspiegeltransistoren auf 3 μm festgelegt (größere Länge als bei Digitaltransistoren). Die optimale Länge der Transistoren im Differenzverstärker sowie in der Ausgangsstufe wurde bei 1,05 μm ermittelt; hier ergibt sich ein Optimum bei der Verstärkung.

Um einen stabilen Betrieb des Operationsverstärkers zu gewährleisten, muss sichergestellt werden, dass ein Phasenrand (Phasenreserve) von mindestens 45°, besser 60°, eingehalten wird. Deshalb muss die Schaltung kompensiert werden. Dazu wird die Miller-Kapazität bei der zweiten Verstärkerstufe erhöht, um die Polstellen der beiden Verstärkerstufen im Frequenzbereich zu trennen. Der zusätzliche Widerstand in Reihe wirkt einer unerwünschten Verschiebung der Nullstellen entgegen [12], [13].

Die Auswirkung verschiedener Kondensator-Widerstands-Kombinationen auf die Parameter des unbelasteten OPV ist in Abbildung 4 zu sehen. Wie Abbildung 4 zu entnehmen ist, wird im Bereich von $R_{\text{comp}} \approx 60 \text{ k}\Omega$ ein Optimum beim Phasenrand erreicht, während der Phasenrand mit steigendem Kapazitätswert ebenfalls steigt; für die größeren Werte jedoch nicht mehr sehr stark. Die Transferfrequenz (Frequenz, bei der nur noch die halbe Verstärkung erreicht wird) ist nicht vom Widerstandswert abhängig, sondern sinkt lediglich mit steigendem Kapazitätswert.

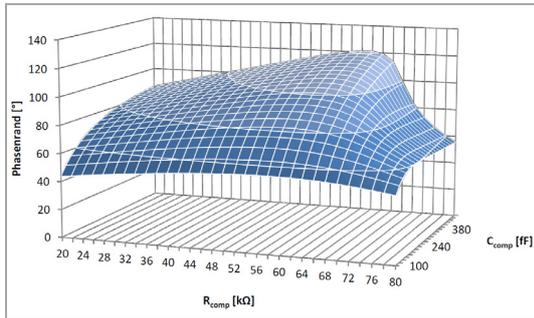


Abbildung 4: Phasenrand des unbelasteten OPV in Abhängigkeit des Kompensationsglieds.

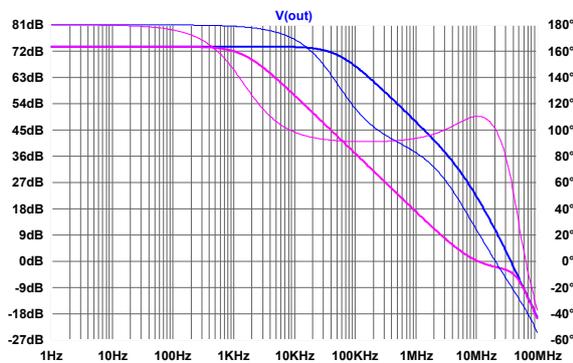


Abbildung 5: Vergleich kompensierter / unkompensierter Operationsverstärker.

Somit gilt es hier, einen Kompromiss zwischen hohem Phasenrand und hoher Transferfrequenz zu finden. Für die letztendliche Auswahl der Kompensationselemente mit $C_{comp} = 375 \text{ fF}$ und $R_{comp} = 57 \text{ k}\Omega$ wurden weitere Simulationen, auch mit belastetem Operationsverstärker (Vgl. Kapitel IV), durchgeführt sowie die Layout-Möglichkeiten berücksichtigt, um eine gute Kompensation des Operationsverstärkers zu erreichen. Bei dem ebenfalls möglichen Kompensationswiderstand von $63 \text{ k}\Omega$ wären zwar der Phasenrand und die Grenzfrequenz bei der reinen Operationsverstärkerschaltung höher, der Phasenrand des belasteten OPV ist bei $57 \text{ k}\Omega$ jedoch höher. In Abbildung 5 ist der Frequenzgang (gestrichelt) und die Verstärkung für den kompensierten (pink) und unkompensierten (blau) OPV aufgetragen (nach [14] S. 513ff).

Bei der Wahl des Stromwertes muss zusätzlich beachtet werden, dass dieser bei 10.000 Messzellen gleichzeitig anfällt und somit eine entsprechende Verlustleistung verursacht. Für die weiteren Simulationen wurde ein Strom von $2 \mu\text{A}$ gewählt; dies stellt einen guten Kompromiss zwischen Verstärkung und Phasenrand dar. Dies ist der Strom durch die Differenzverstärkerstufe; für die Ausgangsstufe fällt zusätzlich der fünffache Strom an). Neben den Transistorparametern hat auch der Strom durch die Verstärkerstufen Einfluss auf Verstärkung (gering) und Phasenrand.

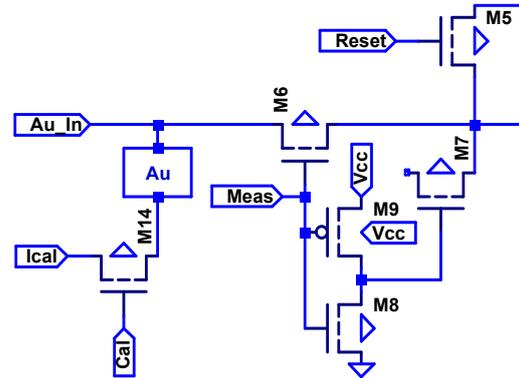


Abbildung 6: Eingangsstufe.

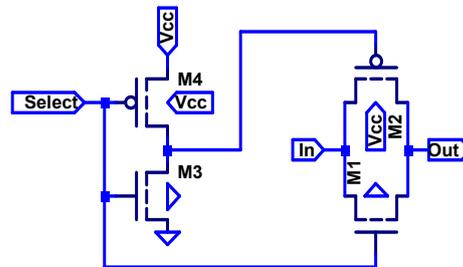


Abbildung 7: Ausgangsstufe.

Als Querstrom wurden als Kompromiss zwischen Verstärkung und Phasenrand $2 \mu\text{A}$ festgelegt. Mit der gewählten Kompensation von 375 fF und $57 \text{ k}\Omega$ ergeben sich somit folgende Kenndaten für den unbelasteten Operationsverstärker:

Verstärkung:	73,6 dB
Transferfrequenz:	2,5 kHz
Grenzfrequenz:	10,7 MHz
Phasenrand:	110°

Um sowohl positive als auch negative Ströme erfassen zu können, wird die Referenzspannung auf

$$U_{ref} = \frac{V_{DD}}{2} = 2,5 \text{ V}$$

festgelegt, um eine möglichst große Ausgangsamplitude zu erreichen:

$$U_{Amplitude,+} < V_{DD} - U_{ref} = 2,5 \text{ V}$$

$$U_{Amplitude,-} > -U_{ref} = -2,5 \text{ V}$$

Da der Operationsverstärker als Integratorschaltung betrieben wird, ist zusätzlich ein Kondensator erforderlich. Die Größe dieses Kondensators bestimmt mit der Integrationszeit den möglichen Messbereich der

Schaltung. Aufgrund der begrenzten Fläche kann die Kapazität nicht beliebig groß gewählt werden. In diesem Fall wurde eine Kapazität von 730 fF gewählt.

Der maximal messbare Strom ist somit durch die minimale Integrationszeit begrenzt. Nimmt man diese mit

$$t_{\text{int,min}} = 1,0 \mu\text{s}$$

an, ergibt sich mit einer Ausgangsamplitude von 2,5 V folgender Maximalstrom:

$$I_{\text{max}} = \frac{U_C \cdot C}{t_{\text{int,min}}} \cdot V \cdot 730 \text{ fF}$$

$$I_{\text{max}} = \frac{2,5 \text{ V} \cdot 730 \text{ fF}}{1 \mu\text{s}}$$

Neben dem Operationsverstärker werden für den Betrieb der Messschaltung zusätzliche Schaltungsgruppen benötigt, siehe Abbildung 6 und 7.

Die Eingangsstufe dient dazu, die Messung zu starten bzw. zu beenden. Hierzu wird der Strom von der Gold-Elektrode (Au) über den Transistor M6 geführt, der von einem Steuersignal entsprechend durchgeschaltet oder gesperrt werden kann. Da beim An- oder Abschalten der Messung der Spannungssprung der Steuerleitung über die Kapazitäten im Transistor (C_{GD}) auf die Messleitung überkoppelt, wird zur Kompensation ein gleichartiger Transistor (M7) an die Messleitung angeschlossen und mit dem invertierten Steuersignal angesteuert; der Drain-Anschluss des Transistors bleibt floatend. Bei entsprechender Ansteuerung kann dieser Transistor auch für eine Referenzmessung genutzt werden, um ggf. unerwünschte Lichteinflüsse zu ermitteln und von der eigentlichen Messung zu subtrahieren.

Um die Störung des Messsignals so klein wie möglich zu halten, wurden die Transistoren (und damit die parasitären Kapazitäten) ebenfalls klein gehalten.

Über den Transistor M14 kann ein Kalibrierungsstrom auf die Goldelektrode und anschließend in die Messzelle geleitet werden, um die genaue Charakteristik der Zelle zu ermitteln. Gleichzeitig kann so überprüft werden, ob die Goldelektrode richtig kontaktiert ist. Außerdem ist ein Transistor (M5) vorhanden, über den die Integrationskapazität vor dem Beginn einer Messung entladen werden kann, um eine definierte Ausgangsbedingung zu erzeugen. Die Anordnung der Messzellen in einem Array erfordert eine gemeinsame Datenleitung für eine Spalte des Arrays. Um die einzelnen Messzellen nacheinander auf die Leitung zu schalten, wird ein Transmission Gate [15] zwischen dem Ausgang des Operationsverstärkers und der Spaltenleitung eingesetzt (Abbildung 7). Da ein Transmission Gate sowohl mit einem invertierten und nichtin-

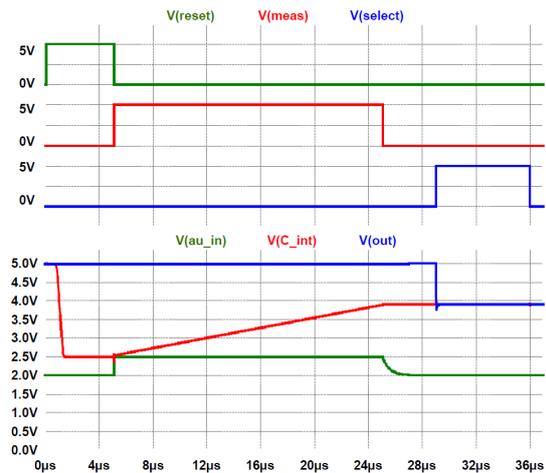


Abbildung 8: Ablauf eines Messzyklus.

vertierten Steuersignal angesteuert werden muss, kommt hier ebenfalls der oben beschriebene Inverter zum Einsatz. Auch hier wurden die Transistoren möglichst klein gehalten, um die Lastkapazität für die Messschaltung zu verringern. Dies ist hier umso wichtiger, da die Kapazitäten auch bei abgeschaltetem Transmission Gate auf die Datenleitung wirken.

B. Ablauf einer Messung

Zuerst wird die Integrationskapazität entladen (Reset), danach sofort die Messung gestartet (Meas). An der Spannung $V(\text{C_int})$ kann man sehen, wie sich die Spannung über der Integrationskapazität aufbaut. Die Spannung am Eingang (Au_In) wird dabei für die Dauer der Messung auf Höhe der Referenzspannung gehalten. Nach Beendigung der Messung bleibt die auf dem Kondensator gespeicherte Spannung erhalten. Soll das Ergebnis nun ausgelesen werden, wird, ausgelöst durch das (Select)-Signal, die Spannung über das Transmission Gate auf die Datenleitung übertragen (Out), von wo aus sie einem AD-Wandler zugeführt werden kann. Die Dauer der Übertragung der Spannung auf die Datenleitung (Wartezeit, bis die Spannung nach Anlegen des Select-Signals ausgelesen werden kann) variiert; sie ist abhängig insbesondere von der Ausgangskapazität (Transmission Gates und Datenleitung) sowie von der vorhergehenden Spannung der Datenleitung, und dem Strom, der über den Operationsverstärker abfließen bzw. von diesem aufgebracht werden kann. Nach dem Abschalten des Transmission Gates kann die nächste Zelle in der Spalte ausgelesen werden.

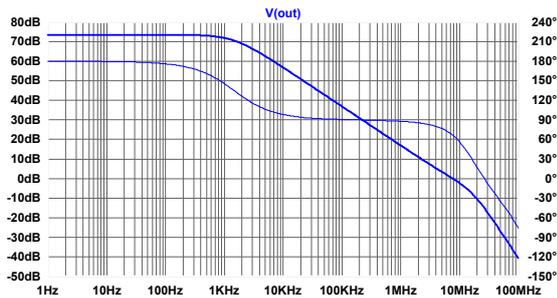


Abbildung 9: Frequenzgang der Messschaltung.

IV. CHARAKTERISIERUNG DER SCHALTUNG

A. Frequenzgang

Zunächst soll der Frequenzgang des Operationsverstärkers (mit Lastkapazitäten; Transmission Gates durchgeschaltet) untersucht werden. Der Frequenzgang ist ein wesentliches Kriterium für die Stabilität der Schaltung. Für einen stabilen Betrieb wird ein Phasenrand von 60° angestrebt, d.h. die Phase sollte bei 0 dB Verstärkung noch 60° betragen. Diese Bedingung wird erfüllt: der Phasenrand beträgt $66,5^\circ$, die DC-Verstärkung $73,6$ dB (4800-fach) (vgl. Abbildung 9). Variiert man die Versorgungsspannung um 10 % und die Temperatur im Bereich von 0°C bis 80°C , bleibt die Schaltung trotzdem stabil, der Phasenrand sinkt nicht unter $63,8^\circ$.

Um den Einfluss von Prozessschwankungen bei Widerstands- und Kapazitätswerten auf den Frequenzgang zu ermitteln, wurde eine Monte-Carlo-Simulation mit 1000 Durchläufen durchgeführt. Dabei wurden die Werte zufällig um bis zu 10 % variiert und die jeweiligen Minimal- und Maximalwerte ermittelt. Dies wurde für die verschiedenen Eckparameter der Transistoren wiederholt. Hierbei ergaben sich ein minimaler Phasenrand von 59° sowie eine minimale Verstärkung von 71 dB.

B. Linearität

Neben dem Frequenzgang spielt die Linearität der Schaltung eine wichtige Rolle. Diese gibt an, wie linear der Strom am Eingang in eine entsprechende Ausgangsspannung umgesetzt wurde.

Simuliert wurden in diesem Fall Ströme von ca. 20 nA bis $1,7\ \mu\text{A}$. Die Ströme werden dabei linear in eine entsprechende Spannung umgesetzt. Um die Abweichung von der Linearität zu ermitteln, wurde der mittlere Wert aller Steigung aus den simulierten Werten berechnet und die Abweichung der einzelnen Steigungen hiervon in Abbildung 10 aufgetragen.

Dabei ist festzustellen, dass die Linearität zu den Extremwerten der Ausgangsamplitude hin abnimmt, sich jedoch mit einer Abweichung von $< 2\%$ in einem eng begrenzten Rahmen bewegt. Würde ein Kompensationswiderstand von $63\ \text{k}\Omega$ eingesetzt, ergäben sich bei

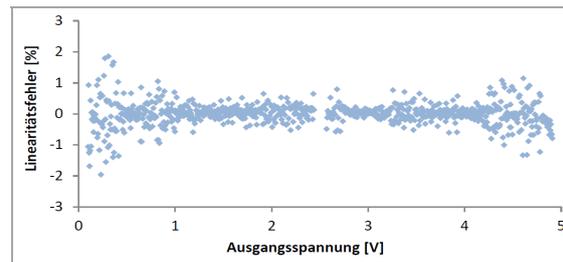


Abbildung 10: Frequenzgang der Messschaltung.

dieser Simulation minimal bessere Werte in den extremen Randbereichen auf Kosten eines etwas geringeren Phasenrandes. Auch unter Berücksichtigung der Transistor-Eckparameter ergeben sich immer Linearitätsabweichungen $< 2,5\%$. Ebenso beeinflussen Temperaturschwankungen von 0°C bis 80°C die Linearität kaum; der Linearitätsfehler bleibt stets $< 2,5\%$. Bei Schwankungen in der Versorgungsspannung ist zu beachten, dass der Aussteuerbereich nicht mehr vollständig genutzt werden kann.

C. Genauigkeit

Außer der Linearität ist natürlich auch die absolute Genauigkeit der Schaltung von Interesse. Hierzu wurden ebenfalls verschiedene Simulationen durchgeführt. Der Strom und die Integrationszeit wurden so variiert, dass sich eine Ausgangsspannung von ca. 4,0 V bzw. 1,0 V einstellt, also 1,5 V Differenz zu der Referenzspannung von 2,5 V. Bei sehr kleinen Strömen ($< 10\ \text{pA}$) und damit langen Integrationszeiten ($> 100\ \text{ms}$) wirken sich Leckströme stark aus, so dass die Spannung an der Integrationskapazität wegdriftet und schließlich die Versorgungsspannung erreicht.

Für den relativen Fehler bei Berechnung des Stroms nach

$$I_{\text{mess}} = \frac{(V_{\text{out}} - V_{\text{ref}}) \cdot C_{\text{int}}}{t_{\text{int}}}$$

ergeben sich für kleine Werte (10 pA) Fehler von ca. 20 %. Um die Genauigkeit weiter zu steigern, wurde die Berechnungsformel für den gemessenen Strom um Korrekturfaktoren und eine weitere Messung (Simulation) ergänzt. Berechnet man nun den Strom nach dieser Formel erneut, erhält man die in Abbildung 11 dargestellten Werte für die Genauigkeit. Der relative Fehler liegt nun im Bereich von 10 pA bis $1\ \mu\text{A}$ durchweg unter 1 % (unter Annahme der genauen Kenntnis der Integrationskapazität von $730\ \text{fF}$).

D. Maßnahmen zur Steigerung der Genauigkeit

Um die Genauigkeit der Messung weiter zu steigern, kann jede einzelne Messzelle über einen separaten Eingang kalibriert werden, um beispielsweise durch Fertigungstoleranzen unterschiedlich große Integrati-

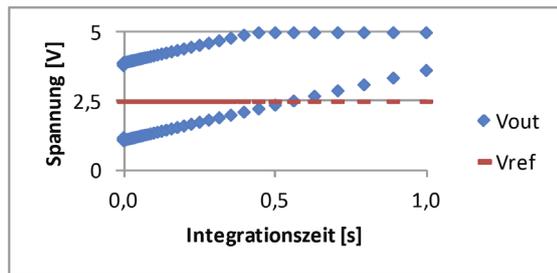


Abbildung 11: Ausgangsspannung über Integrationszeit.

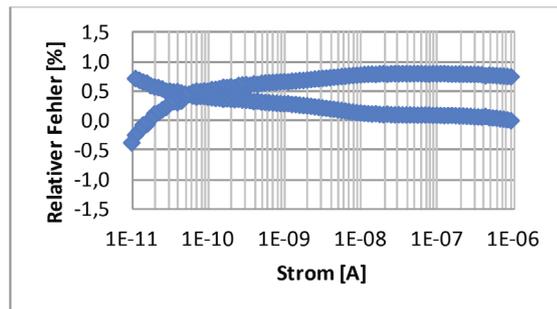


Abbildung 12: Relativer Fehler nach Korrektur.

onkapazitäten, die einen hohen Einfluss auf das Ergebnis haben, auszugleichen. Somit kann bei Bedarf eine individuelle Kennlinie für jede einzelne Messzelle erstellt werden. Das Prinzip hierzu ist in Abbildung 13 gezeigt.

Zusätzlich kann bei Messungen unter Lichteinfluss die Lichtempfindlichkeit der Auswerteschaltung bestimmt werden. Nach der eigentlichen Messung wird wie in Abbildung 14 dargestellt eine zweite Messung vorgenommen, bei der die Elektrode abgetrennt ist. Das Ergebnis dieser Messung ist der Strom, der durch die Messschaltung erzeugt wird und kann zur Korrektur vom ersten Messergebnis subtrahiert werden.

V. LAYOUT

Für die entwickelte Schaltung wurde auch ein entsprechendes Layout entworfen. Randbedingung hierbei war, die Rasterstruktur des GATE-FOREST-Systems, welches am IMS für ASIC-Entwicklung eingesetzt wird, einzuhalten. Die kleinste Einheit in diesem System bildet die sog. Mastergrundzelle mit $9,6 \times 33,6 \mu\text{m}$ Kantenlänge. Da die Messschaltung auf $100 \times 100 \mu\text{m}$ Platz finden soll, sind für das Layout $10 \times 3 = 30$ solcher Mastergrundzellen für das Layout verfügbar. Da jedoch kein passender fertiger Master zur Verfügung stand, konnten diese Zellen frei aufgeteilt werden.

Für den Operationsverstärker wird zunächst die Schaltung ohne Kompensationskapazität und -widerstand umgesetzt. Hierfür werden insgesamt 3 verschiedene Transistoren benötigt. Diese werden auf 2 verschiedene Mastergrundzellen verteilt, wobei der P-Kanal-Transistor auf beiden Zellen integriert



Abbildung 13: Erstellung einer Kennlinie zur Steigerung der Genauigkeit.

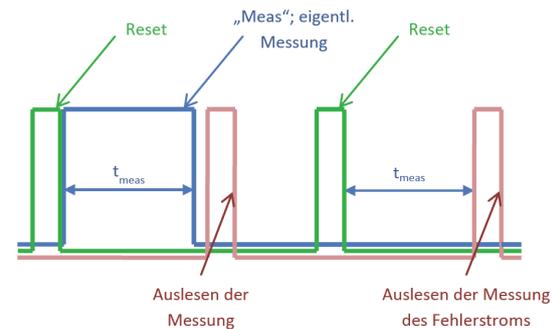


Abbildung 14: Ablauf Kompensation Lichteinfluss.

wird, da dieser sowohl in der Differenz- als auch in der Ausgangsstufe Verwendung findet.

Für die Eingangsstufe und das Transmission Gate am Ausgang wurde eine Mastergrundzelle mit sehr kleinen Transistoren entworfen, um die parasitären Kapazitäten möglichst klein zu halten. Dabei sind die linken P-Kanal-Transistoren etwas weiter ausgeführt, um einen symmetrischen Inverter zu ermöglichen.

Für die Ausgangsstufe kommt dieselbe Mastergrundzelle wie für die Eingangsstufe zum Einsatz, um den Inverter und das Transmission Gate zu realisieren. Für die Integrationskapazität und zur Kompensation des Operationsverstärkers werden auch Kondensatoren benötigt. Diese werden als Plattenkondensatoren zwischen zwei Polysilizium-Schichten gebildet. Die Integrationskapazität wird hierbei in zwei Blöcke zu je ca. 365 fF aufgeteilt; die Kompensationskapazität kommt auf ca. 375 fF (Berechnung über Kapazitätsbelag und Randkapazität). Diese ist etwas anders geformt, da zwischen Kondensator- und Widerstandsstrukturen ein bestimmter Abstand eingehalten werden muss.

Um eine möglichst gute Kompensation des Operationsverstärkers zu erreichen, ist ein Widerstand im Bereich von $60 \text{ k}\Omega$ (vgl. Kapitel I) erforderlich. Die Widerstandsbahnen wurden mit der kleinsten Strukturweite von $1,0 \mu\text{m}$ gezeichnet, um einen möglichst hohen Widerstandswert zu erhalten und damit den Flächenverbrauch zu reduzieren. Lediglich die unteren horizontalen Bahnen wurden etwas breiter gezeichnet, um an diesen Stellen verschiedene Kontaktierungsmöglichkeiten zu erhalten, um bei Bedarf den Widerstandswert nur durch Änderung einer Metallmaske einfach anpassen zu können. Das Layout der Pixelzelle ist in Abbildung 15 dargestellt.

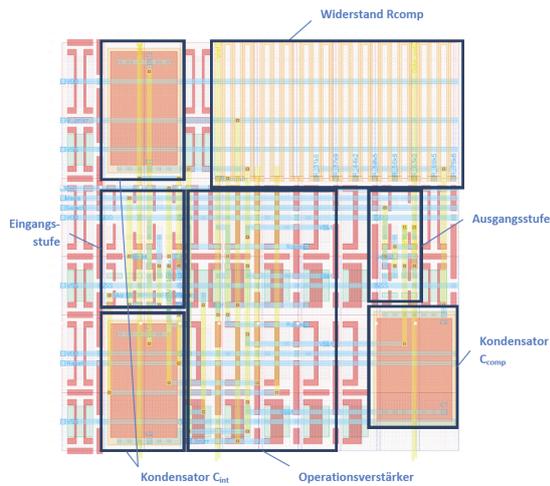


Abbildung 15: Layout der Messzelle.

Nachdem die einzelnen Schaltungsteile in ein entsprechendes Layout umgesetzt waren, mussten diese noch zu einer Gesamtschaltung zusammengefügt werden. Um die Schaltung möglichst vor unerwünschtem Lichteinfall zu schützen, wurde der Operationsverstärker im Zentrum angeordnet, da dieses (in bestimmten Grenzen) von der darüber liegenden Gold-Elektrode verdeckt wird. Links davon ist die Eingangsstufe platziert; auf der rechten Seite die Ausgangsstufe mit Transmission Gate, wobei zwischen Operationsverstärker und Ausgangsstufe noch eine zusätzliche Reihe Transistoren eingefügt wurde, um Flexibilität für eine eventuelle Anpassung der Ausgangsstufe zu schaffen. Ober- und unterhalb der Eingangsstufe wird die in zwei Einheiten aufgeteilte Integrationskapazität platziert. Der Widerstand für die Kompensation des Operationsverstärkers ist rechts oben angeordnet; die Kompensationskapazität rechts unten. Der Kondensator kann hier nicht ganz bis zum unteren Ende der Zelle reichen, da zwischen Kondensator und Widerstand ein Abstand von $6\ \mu\text{m}$ eingehalten werden muss. Da die Zelle in einer Array-Struktur angeordnet werden soll, muss dies hier berücksichtigt werden.

Die nicht von Schaltungsteilen belegten Plätze wurden mit Standard-Transistoren aufgefüllt. Um die Gold-Elektroden kontaktieren zu können, wurden vertikale Kontaktierungsstreifen vorgesehen, die die rasterbedingten Verschiebungen der Goldelektroden ausgleichen können. Da die Goldelektroden in einem $100\ \mu\text{m}$ -Raster angeordnet werden sollen, die Messzelle jedoch im Standard-Raster aufgebaut ist, müssen beim Aufbau des Arrays einige Besonderheiten beachtet werden.

Die Messzelle ist $100,8\ \mu\text{m}$ hoch und $96\ \mu\text{m}$ breit; die quadratische Goldelektrode hat eine Kantenlänge von $70\ \mu\text{m}$. Um die Differenz möglichst gut ausgleichen zu können, werden die Mittelpunkte der jeweiligen Rasteranordnungen aufeinandergelegt, also zwischen der 50. und 51. Zelle (Nr. 49 und 50; Numme-

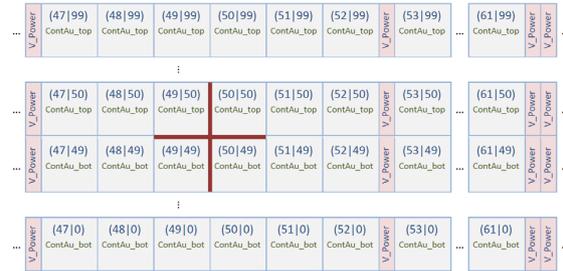


Abbildung 16: Aufbau des Pixel Arrays.

rierung ab 0; vgl. Abbildung 16). In vertikaler Richtung können die Messzellen direkt aneinandergestoßen werden; dies führt zu einem Versatz von $50 \times 0,8\ \mu\text{m} = 40\ \mu\text{m}$. Dies ist (auch nach Addition der $15\ \mu\text{m}$ -Abstände um die Goldelektrode) weniger als eine Zellhöhe, sodass hier eine direkte Kontaktierung der Zelle möglich ist. Es müssen jedoch für die obere und untere Hälfte der Array-Anordnung unterschiedliche Kontaktpositionen (in vertikaler Richtung) vorgesehen werden, da der gesamte Versatz $80\ \mu\text{m}$ beträgt, die Goldelektrode jedoch nur eine Kantenlänge von $70\ \mu\text{m}$ aufweist.

In horizontaler Richtung jedoch müssen weitere Ausgleichsmaßnahmen getroffen werden, da hier der Versatz deutlich größer ist und eine Kontaktierung der äußeren Elektroden schwierig wäre. Daher wird nach 3 Messzellen (die jeweils $4\ \mu\text{m}$ zu schmal sind) eine vertikale Reihe Standardzellen ($9,6\ \mu\text{m}$ breit) eingeschoben. Diese enthalten zusätzlich vertikale Versorgungsspannungsleitungen. Da diese den Versatz nicht vollständig ausgleichen ($2,4\ \mu\text{m}$ Rest) wird nach 4 Dreierblöcken noch eine weitere vertikale Reihe eingefügt, um den Versatz vollständig auszugleichen ($4 \times 2,4\ \mu\text{m} = 9,6\ \mu\text{m}$).

Um das entworfene Layout zu verifizieren, wurde ein DRC durchgeführt. Mittels einer LVS-Prüfung gegen die in Cadence erstellten Schaltpläne wurde die Übereinstimmung des Layouts mit den Schaltplänen sichergestellt. Weiterhin wurde eine aus den Layoutdaten extrahierte Netzliste mit SPICE simuliert und die Übereinstimmung dieser Post-Layout Simulation mit den Simulationen auf Schaltungsplanebene nachgewiesen.

VI. ERGEBNIS & AUSBLICK

Im Rahmen dieser Arbeit wurde eine Schaltung entworfen und simuliert, die zur Vermessung von Peptid-Dioden geeignet ist. Es können sowohl negative als auch positive Ströme gemessen werden, wobei den zu untersuchenden Dioden eine positive oder negative Vorspannung aufgeprägt werden kann. Da die Schaltung auch am IMS gefertigt werden soll, wurde ein entsprechendes Layout in Cadence umgesetzt und eine entsprechende Arraystruktur der Messzellen erstellt.



Um die Messzellen zu betreiben, muss noch eine Ansteuerschaltung realisiert werden. Ebenso wird noch eine AD-Wandler-Schaltung benötigt, um die gemessenen Ströme, die durch die Messschaltung in eine proportionale analoge Spannung gewandelt wurden, in einen digitalen Wert umzusetzen, welcher entsprechend weiterverarbeitet werden kann.

Nach der Entwicklung und Produktion des Messchips ist es somit möglich, viele verschiedene Peptid-Kombinationen gleichzeitig auf ihre Dioden-Eigenschaft hin zu untersuchen und somit möglicherweise den Weg für eine neue Generation an kostengünstig zu produzierenden Solarzellen zu ebnet.

DANKSAGUNG

D. Seiz dankt allen Mitarbeitern am IMS, die ihn während seiner Zeit am IMS tatkräftig unterstützt haben. Teile der Arbeit wurden im Rahmen des FP7-ENERGY-2010-FET Forschungsprojektes „Peptide-based diodes for solar cells (PEPDIODE)“, project number 256672 gefördert.



Daniel Seiz erhielt den akademischen Grad des B. Eng. in Elektrotechnik im Jahr 2010 von der Dualen Hochschule Stuttgart. Er studiert Elektrotechnik und Informationstechnik im Masterstudiengang an der Universität Stuttgart.



Harald Richter erhielt den akademischen Grad Diplom-Physiker im Jahre 1980 von der Universität Kaiserslautern und promovierte 1983 an der Universität Stuttgart zum Dr. rer. nat. Seit 1986 arbeitet er am Institut für Mikroelektronik Stuttgart. Hier leitet er den Bereich Systeme.

LITERATURVERZEICHNIS

- [1] http://cordis.europa.eu/search/index.cfm?fuseaction=project.document&PJ_LANG=EN&PJ_RCN=12166864&pid=9 [Zitat vom: 14.02.2012]
- [2] <http://de.wikipedia.org/wiki/Peptid> [Zitat vom: 28.03.2012]
- [3] Breitling, Frank. *Peptidbasierte Dioden als Basis eines Bionik-inspirierten synthetischen Photosystems*. [Präsentation] Karlsruhe : KIT, 2009.
- [4] Ferrocene-functionalized peptides as a model system for solar cells (Cache). http://webcache.googleusercontent.com/search?q=cache:nMZNZkQXG5sJ:www.imt.kit.edu/arbeitsn_938.php+peptide+solar+cell+site:kit.edu&cd=1&hl=de&ct=clnk&gl=de [Zitat vom: 02.04.2012]
- [5] http://www.ptj.de/lw_resource/datapool/items/item_1650/projekte_mol_bionik_web1.pdf. [Zitat vom: 02.04.2012]
- [6] Nesterov-Müller und Schillo. *Peptiddiode*. [Präsentation]. s.l.: KIT.
- [7] Ji, Honghao und Abshire, Pamela A. *Fundamentals of Silicon-Based Phototransduction*. [Hrsg.] Orly Yadid-Pecht und Ralph Etienne-Cummings. CMOS Imagers: From Phototransduction to image processing. Boston : Kluwer Academic Publishers, 2004, S. 1-51.
- [8] Seifart, Manfred. *Analoge Schaltungen*. Heidelberg : Hüthig Buch Verlag GmbH, 1990.
- [9] Manoli, Y. *CMOS Analoge Schaltungen*. Freiburg: Lehrstuhl für Mikroelektronik; Albert-Ludwigs-Universität Freiburg, 2002.
- [10] Blalock, Benjamin J. *Active Loads. Lecture Notes EE 8223 - Analog IC Design*. 2001, S. 82-90.
- [11] http://www.ims-chips.de/content/pdf/text/White_paper_MS_Array_09_11.pdf [Zitat vom: 05.06.2012]
- [12] Grözing, Markus. *Verstärkertechnik I+II*. [Vorlesungsmitschrieb]. Stuttgart 2011.
- [13] Haggemiller, Lenz. *Operationsverstärker*. Regensburg : s.n., 2002.
- [14] Tietze, Ulrich, Schenk, Christoph und Gamm, Eberhard. *Electronic Circuits*. s.l. : Springer, 2007.
- [15] <http://de.wikipedia.org/wiki/Transmission-Gate> [Zitat vom: 28.03.2012]

Gate Driver Design for Fast Switching DC-DC Converters

Juergen Wittmann, Tobias Funk, Bernhard Wicht

Abstract—Fast switching integrated DC-DC converters enable system cost reduction as external components can be reduced in size or eliminated. Fast switching leads to several challenges on circuit side. This is even more critical for automotive applications with input voltages up to 40 V and very short duty cycles. The gate driver for the integrated power FET switches belongs to the most critical blocks. This paper discusses the trade-off between speed and losses in the gate driver. By using cascaded buffer stages with parallel branches with asymmetrical sizing an optimized gate driver could be designed in a 180 nm high-voltage BiCMOS technology. The presented gate driver is suitable for a 40 V step down converter running at 20 MHz switching frequency with a duty cycle as low as 10 %.

Index Terms—DC-DC Converter, gate driver.

I. INTRODUCTION

In several applications the trend to higher switching frequencies in switched mode power supplies can be observed to reduce system costs. The control circuits can be integrated with a very small die size in modern technologies but passive components like inductors or filter capacitors remain the dominating cost factor due to the limited power density. Especially in high efficient systems passives with low losses are required. These passives usually dominate the size. An increasing switching frequency leads to a proportional reduction of the passive components and thus a significant reduction of the system size can be achieved. Several DC-DC converters with switching frequencies with some tens of MHz were proposed for applications with low input voltages down to 5 V. The realization of high switching frequencies at the same time with a high input voltage range is a challenge, because technologies supporting high voltage transistors are limited in bandwidth due to a larger structure size. A higher input voltage is important in e.g. automotive,

Juergen Wittmann, juergen.wittmann@reutlingen-university.de, Tobias Funk, tobias.funk@student.reutlingen-university.de, and Bernhard Wicht, bernhard.wicht@reutlingen-university.de, are members of the Robert Bosch Center for Power Electronics, Reutlingen University, Alteburgstraße 150, 72762 Reutlingen.

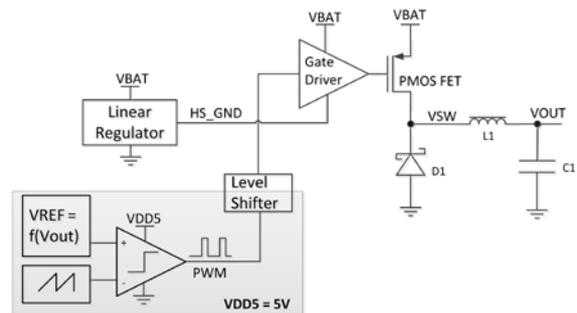


Fig. 1: PWM controlled, asynchronous buck converter.

battery driven applications for e-mobility or solar systems, where the input voltage can go to some hundreds of volts while it is still required to have low voltage sub-regulation. Considering a buck converter (Fig. 1), the conversion ratio between input and output voltage is regulated by the duty cycle of a pulse-width modulated signal (*PWM* signal). A high input-to-output ratio requires a very short on time of the *PWM* signal controlling the switching FETs. In consequence, the gate driver for the FETs requires a shorter delay and an increased driving strength for faster charging and discharging of the gate of the power FETs. At the same time the gate driver needs to provide high power efficiency and thus a low current consumption of the gate driver block.

A buck converter concept based on a fast gate driver for high speed operation is presented in Section II. Section III focuses on requirements for the high side driver for fast switching. The concept of a cascaded driver is discussed and the theory of the design is presented. The driver is implemented such that the current consumption can be minimized. A symmetrical sizing of the driver stages is utilized to reduce cross currents.

In section IV simulation and measurement results are presented. Three driver stages are compared with respect to the switching delay, current consumption and die size.

II. FAST SWITCHING BUCK CONVERTER

The gate driver investigation presented in this paper is related to an asynchronous buck converter as shown

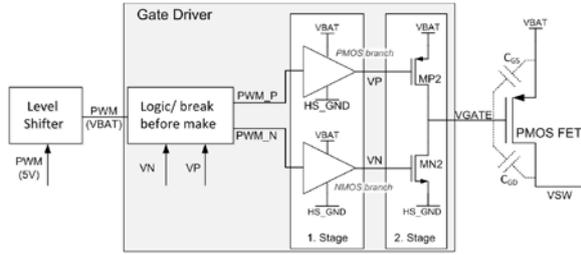


Fig. 2: Principle of a two-stage gate driver with “break before make” circuit.

in Fig. 1. In this case the converter operates from a battery input voltage V_{BAT} . The external output filter consists of the inductor $L1$ and capacitor $C1$. A Schottky diode $D1$ is used to benefit from fast recovery time and low junction capacitance for fast switching and reduced switching losses at higher frequencies. An integrated PMOS MOSFET is used as high side switching device. A NMOS device would achieve lower on-state resistance but complicates the gate control since a boot strap circuit or a charge pump circuit is required to achieve the gate overdrive. The output voltage is controlled by a PWM signal which switches the high side FET. The control circuit is implemented in the low voltage, 5 V domain as depicted in the grey box in Fig. 1. An amplifier generates a reference voltage V_{REF} in dependency of the output voltage. V_{REF} is then compared to a saw-tooth signal which generates the PWM signal. The duty cycle of the PWM signal determines the on/off time of the FET and thus controls the ratio of input to output voltage. The PWM signal is then passed to the high voltage domain by a level shifter to control the input of the high side gate driver.

To turn on the high side PMOS FET, the high side gate driver needs to generate a source-gate voltage of 5 V. This is achieved by generating a high side ground HS_GND which is 5V below the battery input voltage V_{BAT} by a linear voltage regulator (see Fig. 1). The gate voltage of the power FET is switched between V_{BAT} and HS_GND by the gate driver.

To optimize the converter for higher switching frequency, the whole control circuit chain needs to be optimized for speed. This includes the saw-tooth generator, PWM comparator, level shifter and the gate driver.

The maximum delay of the gate driver is determined by the duty cycle and the switching frequency. The duty cycle D is determined by the ratio of output to input voltage $D = V_{OUT}/V_{BAT}$. The duty cycle is minimal if the input-to-output voltage ratio is maximal, i.e. the converter regulates a very low output voltage from a very high input voltage. The input-to-output voltage ratio and hence the minimum duty cycle D_{min} is limited by the minimum on time $t_{on,min}$, at which the regulator should operate.

Based on the relation

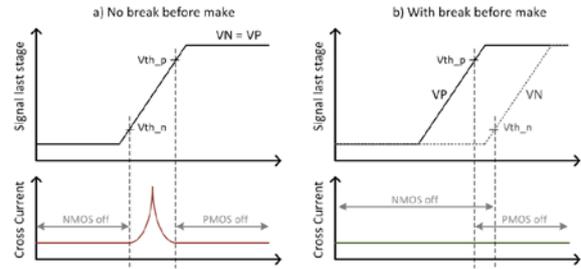


Fig. 3: a) Driver without break before make. b) Cross current reduction with break before make circuit.

$$t_{on,min} = D_{min} \cdot \frac{1}{f_{sw}} = \frac{V_{BAT}}{V_{OUT}} \cdot \frac{1}{f_{sw}}$$

the minimum on time shows a strong dependency on the switching frequency f_{sw} for given input and output voltages.

The minimum on time which a converter can achieve depends mainly on the performance of the gate driver. The design goal for the buck converter presented in this paper was to operate at a switching frequency of 20 MHz while the output-to-input voltage ratio and thus the duty cycle is smaller than $D = 1/10$. This corresponds to a typical automotive requirement with an input voltage of $V_{BAT} = 40$ V and an output voltage below $V_{OUT} = 5$ V, while a small margin for regulation is considered in the duty cycle. The minimum on time at these conditions can be calculated as $t_{on} = 1/(D \cdot f_{sw}) = 5$ ns.

Accordingly, the minimum off time, which is obtained at a maximum duty cycle, could be calculated. This would correspond to an input voltage which is only slightly above the output voltage.

III. HIGH SIDE DRIVER DESIGN

A. High side driver concept

To analyze the switching behavior the principle of a high side driver is explained at the example of a two stage driver shown in Fig. 2. The first stage is driven from standard logic gates which are providing the PWM input signals to the driver. The second stage controls the gate of the PMOS FET at node $VGATE$. The transistors $MP2$ and $MN2$ have to have a large channel width/length (W/L) ratio to drive the large gate capacitance of the PMOS FET, which is in this case in the range of 150 pF. Since the first stage has to be small because the digital gate cannot drive large capacitive loads, also its driver strength is limited.

If the last stage is driven from one particular driver only, i.e. the driver signals for $MP2$ and $MN2$ $V_N = V_P$ are the same, a large cross current would occur as shown in Fig. 3a. In that case $MN2$ will conduct when the increasing gate voltage is higher than the NMOS threshold voltage $V_{th,n}$ but $MP2$ would

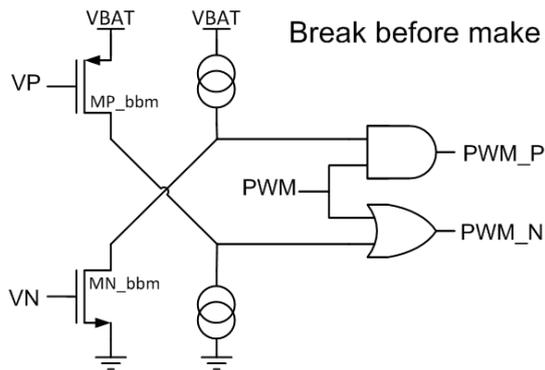


Fig. 4: Digitally controlled “break before make” circuit.

only be off when the gate voltage exceeds $V_{BAT} - V_{th,p}$ with the PMOS threshold voltage $V_{th,p}$ of MP2. The cross current peak increases with increasing driver strength and decreasing slope of the gate signal $V_N = V_P$.

The cross current can be significantly reduced by applying a “break before make” circuit. In the driver of Fig. 2 this is realized by separating the driver branch for MP2 from the driver branch for MN2. An additional circuit observes the signals V_P and V_N and recognizes, if MP2 and MN2 are turned on or off, respectively. An example for a conventional “break before make” is shown in Fig. 4. The logic controls the PMOS driver branch separately from the NMOS driver branch by the signals PWM_P and PWM_N (see Fig. 2). The behavior is illustrated in Fig. 3b. The PWM_N signal is delayed to PWM_P in a way that the signal V_N turns on MN2 only after the signal V_P turned off. MP2 and MN2 are never on at the same time and the cross current is reduced ideally to zero.

While Fig. 3 explains the principle behavior, Fig. 5 depicts a more detailed turn off transition of the PMOS power FET with the different phases marked A to D. The transition starts with a falling PWM signal at the time t_0 and the NMOS branch turns off MN2 during phase A. After t_1 the “break before make” has a delay which adjusts the off time overlap of MP2 and MN2 (B). After t_2 the PMOS branch switches. After the delay of the PMOS driver branch (C) the gate of the PMOS FET is charged with full driver strength of MP2 and the FET turns off.

The drawback of the conventional “break before make” circuit is that it suffers from a large process variation which requires a large margin in the phase B in Fig. 5. Moreover it requires two times the delay of the buffer stage since it propagates once through the PMOS branch and after the feedback again through the NMOS branch. This defines the minimum possible delay and the optimization of the total delay is limited.

The gate signals of the PMOS FET during one switching period are sketched in Fig. 6. In the off transition the gate voltage starts to rise when PWM

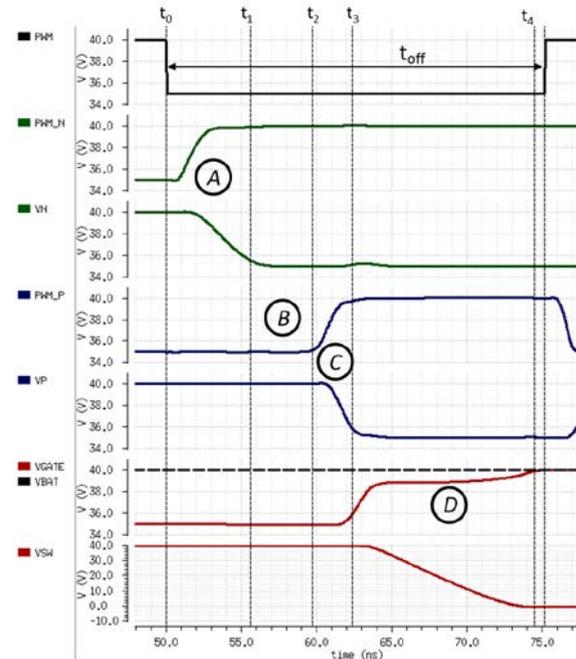


Fig. 5: Signals of the two-stage gate driver with “break before make” circuit during low to high switching transition.

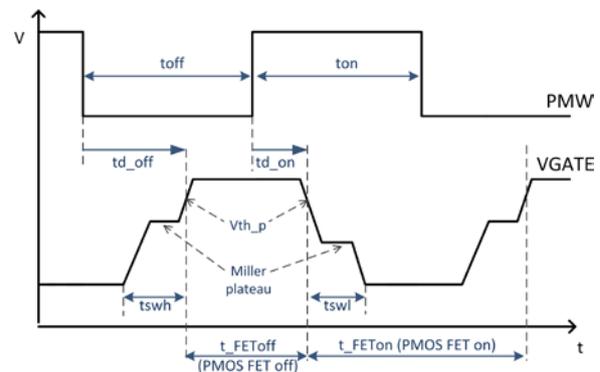


Fig. 6: Delay of a gate driver.

goes low after the delay of the gate driver. The gate voltage takes the time t_{swh} until it crosses the threshold voltage of the PMOS FET $V_{th,pFET}$ and the FET is completely turned off after t_{off} . In the on transition when PWM goes high and after the gate driver delay $VGATE$ starts to drop, it crosses the threshold voltage $V_{th,pFET}$ immediately and the PMOS FET turns on after the time t_{on} . After the time t_{swl} the FET is completely on. This behavior results in a significant asymmetry in the PMOS FET on time t_{FETon} and off time t_{FEToff} (see Fig. 6) which limits the minimum on and off time. Either the maximum or minimum possible input-to-output voltage ratio, or the maximum switching frequency is limited.

The gate charge and discharge times are mainly dominated by the miller plateau which can be seen

during t_{swl} and t_{swh} in Fig. 6. The miller plateau occurs when the drain voltage of the PMOS FET decreases or increases. A significant amount of charge is coupled through the drain-gate capacitance of the PMOS FET, which creates an additional current flowing through the gate driver and thus causes an additional voltage drop at the driving transistor in the last driver stage, even if the driver transistor is fully turned on. If the miller plateau is below the PMOS FET threshold $V_{th,pFET}$, in the off transition, the PMOS FET still remains on during the miller phase (as it is the case in Fig. 6). By further increasing the driver strength of the gate driver the on-resistance of the last driver stage decreases in a way, that during the miller phase the gate voltage V_{GATE} is above $V_{BAT} - V_{th,pFET}$ and thus the PMOS FET is off. In consequence, the transition time can be reduced. The impact of the miller plateau during on transition can be reduced accordingly.

Fig. 5 shows the switching of V_{GATE} for a buck converter with a conventional two stage gate driver with the “break before make” circuit of Fig. 4. The driver was operated at a switching frequency of 20 MHz at $V_{BAT} = 40$ V with a duty cycle of 50 %. The miller plateau duration (D) is in the time range of the gate driver delay. For a fast switching gate driver, it is important, that the PMOS FET is safely turned off during this time. Considering the off time of the *PWM* signal in Fig. 5 at the frequency of 20 MHz, total switching transition is only finished at t_4 which is nearly at the end of the off time. An operation with duty cycles in the range of 10 % is impossible.

B. Optimization for fast switching.

The considerations of Section III.A lead to the requirements for the optimization of a gate driver operation in a converter with a switching frequency of 20 MHz:

- The driver strength needs to be optimized to be able to reduce the switching transition time and eliminate the miller plateau at V_{GATE} to be able to fully turn of the PMOS FET in the miller phase.
- The delay of the gate driver needs to be reduced to be able to operate with an on time which is smaller than $t_{on_{min}} < 5$ ns (see Section II) for a switching frequency of $f_{sw} = 20$ MHz and a duty cycle $D < 10$ %. With such a minimum delay, the effect of asymmetric switching gets acceptable small.
- The cross current needs to be avoided in the last driver stage in the switching transitions to improve the current consumption and the efficiency of the converter. An improved “break before make” technique is required. The delay of the conventional circuit cannot be minimized since the switching signal propagates two times the

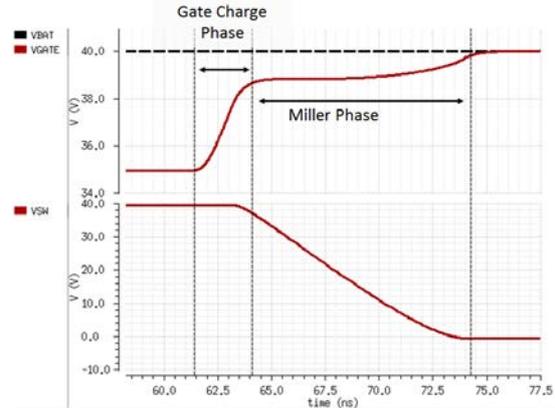


Fig. 7: PMOS FET turn off transition.

buffer in series and a margin is required because the circuit suffers from large process variations. This will be addressed further in Subsection 3 below.

1) Defining the driver strength of the driver

The slope of the gate signal V_{GATE} is defined by the driver strength of the last driver stage (Second stage in Fig. 2) and the gate capacitance of the PMOS FET. To obtain a simple rule for dimensioning the last driver stage, it is assumed that the switching node *VSW* (or drain of PMOS FET) starts only to increase or decrease if V_{GATE} is nearly fully charged or discharged. The nodes *VSW* and V_{GATE} during the turn off transition at time t_3 in Fig. 5 are demonstrated in more detail in Fig. 7. *VSW* only starts to discharge, when V_{GATE} is charged to the level of the miller plateau in the conventional two stage driver. For a stronger and faster gate driver this assumption is even more accurate. This assumption allows neglecting the miller effect during the charging and discharging phase of V_{GATE} . In that way, the turn off and turn on transitions can be separated in two phases, the gate charge phase and the miller phase as shown in Fig. 7.

In the gate charge phase, neglecting the miller capacitance, the effective gate capacitance can be calculated as $C_G = C_{GD} + C_{GS}$, where C_{GD} and C_{GS} are the gate-drain and the gate-source capacitance as depicted in Fig. 2. The gate is charged within the time t_{GC} from low to high with the average gate charge current I_{GC} which can be calculated in first order as

$$I_{GC} = C_G \times \frac{V_{BAT} - HS_GND}{t_{GC}}$$

The PMOS transistor of the last stage has to be designed to provide at least the current I_{GC} when fully turned on. The gate charge time t_{GC} need to be significantly smaller than the minimum on time $t_{on_{min}}$.

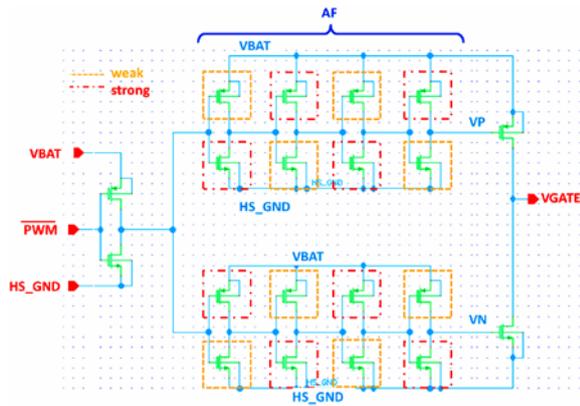


Fig. 8: Implementation of a 6-stage high side driver with separated PMOS and NMOS branch.

The current I_M in the miller phase in Fig. 7 through the last stage is determined by the injected charge of the gate drain capacitance C_{GD} when the switching node voltage V_{SW} is falling. The slope of V_{SW} is determined by the parasitic capacitance at the node V_{SW} , the output current and the driver strength of the PMOS FET. The current I_M can be calculated as

$$I_M = C_{GD} \times dV_{SW}/dt$$

The current I_M is maximal at the maximal possible slope dV_{SW}/dt , which can occur at the converter switching node V_{SW} . The last stage has to be designed strong enough to provide the larger of both currents, the gate charge current I_{CG} or the current through the miller phase I_M .

2) Delay time and driver strength optimization

An optimization of the total delay of the gate driver can be achieved by using several cascaded buffers in series. A schematic of a cascaded driver is shown in Fig. 8. To optimize the gate driver delay, at first a single branch buffer stage with N buffer stages is considered. The load capacitance C_L is equivalent to the gate capacitance C_G of the PMOS FET. The capacitance C_{IN} represents the input capacitance of a minimum sized buffer stage. The buffer stages consist of simple inverter stages with minimum length of the PMOS and NMOS transistors while the width of the transistors is scaling with the driver strength of each stage.

To obtain the number of buffer stages N and the sizing of the driver strength of a particular buffer stage k for the optimal delay, the tapered buffer approach proposed by Jaeger [1] was used. This approach provides the sizing of the driver to obtain an optimal delay and driver strength for a given gate capacitance of the PMOS FET.

Assuming the NMOS transistor in stage k has the width W_k and, subsequently, a width W_{k+1} in the following stage, the factor α defines the increase of the driver strength from stage to stage so that

$$\alpha = \frac{W_{k+1}}{W_k}.$$

The area from stage to stage also increases with the factor of α and thus the input capacitance of each stage. At N number of stages the output capacitance is α^N times larger than the input capacitance

$$C_L = \alpha^N \cdot C_{IN}.$$

The number of stages N can be calculated as

$$N = \frac{\ln\left(\frac{C_L}{C_{IN}}\right)}{\ln \alpha}.$$

According to [1] the minimum gate driver delay is obtained for $\alpha = e = 2.718$. Later publications [2], [3] showed that the optimum of the total delay depends also on the ratio of C_L/C_{IN} and thus the optimal delay is obtained at a moderately higher α . For the PMOS FET used in Fig. 1 the gate capacitor was 145 pF, while the input capacitance of a minimum sized buffer stage is ~ 7 pF. For $\alpha = 2.718$ the optimal number of stage would be 10. Choosing $\alpha \approx 3.5$ the optimum is obtained at $N = 8$. Recalculating the exact value for α at 8 stages results in $\alpha = 3.46$. To further reduce the die size of the driver experiments were also done with a gate driver existing of $N = 6$ buffer stages with $\alpha = 5.24$.

Finally it has to be verified if the last stage of the driver is strong enough to be able to provide the gate charge current I_{CG} and the current I_M in the miller phases according to the calculations in Section III.B.1) to achieve the design goal for fast switching. If the maximal current which the last driver stage can provide is less than the calculated current I_{CG} for the given requirements, the process is limiting the driver speed and the given requirements cannot be fulfilled.

If the driver current from the last stage is smaller than the calculated current I_M , the driver strength of the last stage needs to be increased. Especially at very high input voltages and high output impedances at the drain of the PMOS FET the calculated driver strength could be too weak to eliminate the miller effect on the switching delay. To completely eliminate the miller effect, it would be required to use the effective miller capacitance instead of the physical capacitance C_L of the PMOS FET for dimensioning the driver strength. In this work the optimization according to Jaeger [1] and optimizing for the physical gate capacitance of the PMOS FET was sufficient to achieve the required current in the output stage to provide I_{CG} and I_M .

3) Cross current reduction

The conventional “break before make” circuit suffers from a large variation over process corners and turned out to be not suitable for fast switching applications.

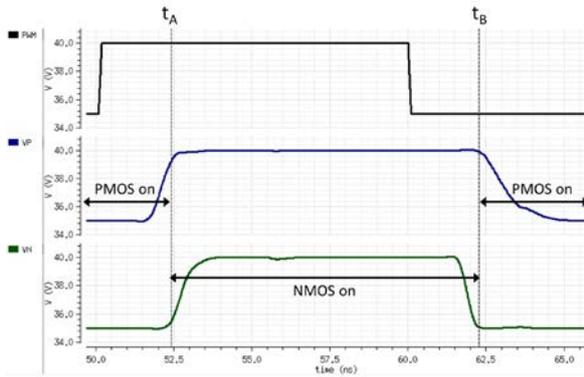


Fig. 9: Signals VP and VN of the 6-stage driver with AF factor.

For the cross current optimization at higher frequency a different concept turned out to be more suitable. The implemented circuit of the split driver is shown in Fig. 8. The buffer stage dimensioned in Section III.B.2) is now split up into a PMOS and NMOS driver branch as depicted already in Fig. 2. The first and the last stage remain as a single inverter, while the stages 2 to 5 are split up into two branches. Since each driver branch has only to drive either the NMOS or the PMOS gate capacitance of the last driver stage, the driver strength of each branch is reduced proportionally to the gate area of MP2 and MN2, respectively. In this case the gate capacitances of MP2 and MN2 are assumed to be equal as an approximation to simplify the design.

For the dead time to prevent cross conduction of the PMOS and NMOS transistors an asymmetry factor AF is applied in the particular buffer stages. By alternating the NMOS and PMOS transistor strengths from stage to stage from weak to strong in both branches as shown in Fig. 8, the rising PWM signal is delayed in the NMOS branch, while a falling PWM signal is delayed in the PMOS branch. The delayed signals are only propagated by the weak transistors. The falling PWM signal in the NMOS and the rising PWM signals in the PMOS branch are only propagated by the strong transistors and thus the delay is less. The asymmetry is created by increasing or decreasing the width W of each transistor by the asymmetry factor AF . For weak transistors the adjusted width is

$$W_{weak} = W \cdot \left(1 - \frac{AF}{100\%}\right),$$

while for strong transistors the width is calculated as

$$W_{strong} = W \cdot \left(1 + \frac{AF}{100\%}\right).$$

Compared to the “brake before make” circuit of Fig. 2 and Fig. 4, the delay created by such an asym-

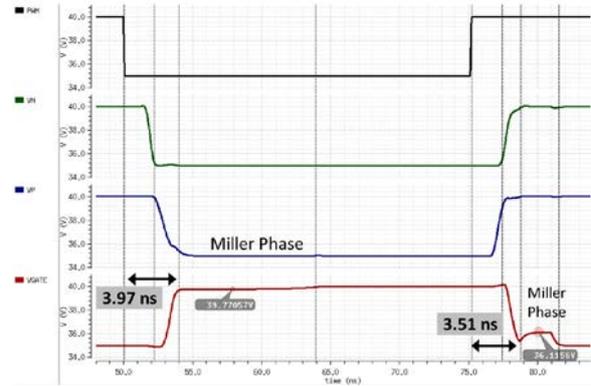


Fig. 10: Signals of the 6-stage high side driver with separated PMOS and NMOS branch and AF factor.

metry factor tracks much better over process corners. Fewer margins in the design are required and thus the delay can be adjusted very accurately. Therefore, the technique is very well suitable for fast switching converters in the Megahertz range.

In Fig. 9 the signals VP and VN of the 6-stage gate driver of Fig. 8 are shown. The optimal delays of VP and VN are obtained when the NMOS of the last stage turns on exactly at the time when the PMOS transistor is completely turned off and vice versa. In that way, the total delay of the driver is minimized while still no cross current occurs in the last stage. Typically an asymmetry factor in the range of $AF = 20 - 30\%$ is suitable [4]. The signals in Fig. 9 are based on an asymmetry factor of $AF = 20\%$ which also turned out to be the best choice for the final design as presented in the next section.

IV. SIMULATION RESULTS

Fig. 10 shows the switching behavior of the optimized 6-stage gate driver with an asymmetry factor $AF = 20\%$ as part of a 20 MHz buck converter at 40 V input voltage. The total delay from the PWM low transition until the charging of the gate of the PMOS FET VGATE is completed is 3.97 ns. The miller plateau is around ~ 39.8 V and well above the threshold voltage of the PMOS FET so that the PMOS FET remains safely turned off during the miller phase. At the rising transition of PWM , the delay of the gate driver is measured as 3.51 ns while the miller plateau is at ~ 36.1 V. The miller plateau is higher when the PMOS FET turns on, because the slope of the switching node V_{SW} in an asynchronous buck converter is higher. The miller plateau of 36.1 V in Fig. 10 is low enough that the PMOS FET can be considered to be nearly fully turned on and thus has a negligible impact on the switching behavior as outlined in Section III.B.1).

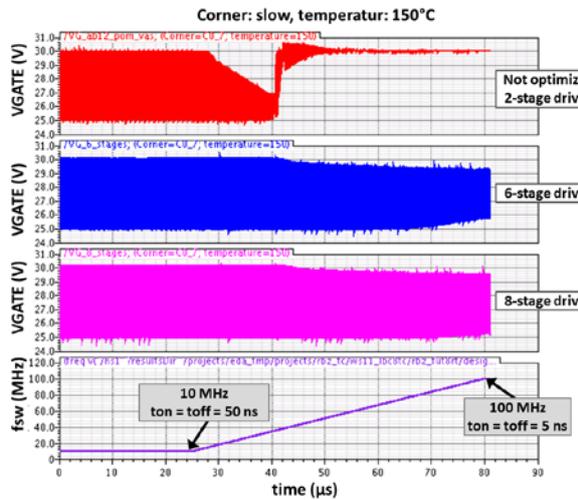


Fig. 11: Comparison of the total current consumption.

In Fig. 11 a simulation of the worst case corner (process corner “slow”, temperature 150 °C) is performed at an input voltage of $V_{BAT} = 30\text{ V}$. The conventional 2-stage gate driver of Fig. 2, and the 6-stage and 8-stage driver with asymmetry factor according to Section III.B.2) are compared while a sweep over the switching frequency is simulated. Instead of decreasing and increasing the duty cycle at 20 MHz to obtain the minimum and maximum on time, the duty cycle is kept at 50 %, while the frequency is further increased to decrease the on and off time of the converter. At 100 MHz, the *PWM* signal thus would have a on and off time $t_{on} = t_{off} = 5\text{ ns}$. This corresponds to a *PWM* signal of 20 MHz with a duty cycle of 10 % or 90 %, respectively.

The 8-stage driver is still able to fully switch the gate *VGATE* of the PMOS FET at 100 MHz. The 6-stage driver is also able to fully operate at 100 MHz, but *VGATE* does not reach its full levels due to remaining switching delays. The level is slightly reduced to the range of 25.6 V to 29.4 V. The PMOS FET can still be fully turned off during the minimum off time. During the minimum on time, the gate driver of the PMOS FET provides only 4.4 V instead of 5 V due to the remaining miller plateau after the turn on transition. This decreases slightly the efficiency of the converter but a proper operation is still possible.

In Fig. 12 the current consumption is compared. The current is obtained by averaging the current in each switching period. Delay optimized gate drivers have an increased current consumption compared to the non-optimized 2-stage driver. The 8-stage driver has a higher current consumption than the 6-stage driver due to the larger total sizes of the transistors and thus the larger total gate capacitances which has to be charged and discharged.

As the delay difference between the 6-stage and 8-stage driver is negligible, the 6-stage high side driver

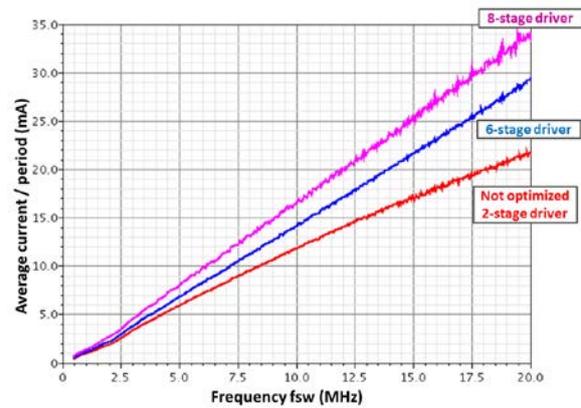


Fig. 12: Comparison of the total current consumption.

turns out to be the better choice due to the smaller die area and the smaller current consumption.

V. CONCLUSION

The design of a fast switching high side gate driver for a buck converter with up to 40 V input voltage, a switching of 20 MHz and a minimum duty cycle of down to 10 % is presented. Speed optimization requires improving the driver strength and optimizing the delay of the driver stage. Conventional “break before make” circuits reduce cross currents but have significant negative impact on the delay of the gate driver and turned out to be not suitable for fast switching gate drivers.

To achieve the required minimum on time of $< 5\text{ ns}$ a tapered buffer was used. The driver strength of the last stage driving the PMOS FET was designed such that the switching transitions are in the range of 1 ns in the worst case condition. The improved driver is strong enough to eliminate the influence of the miller plateau on the switching behavior and the delay time.

Asymmetry was applied to the driver branches to shape the driving signals in a way that almost no cross currents occur but at the same time the delay is kept at minimum.

A tapered buffer with 6 stages and an asymmetry factor of 20 %, designed in a 180 nm high-voltage BiCMOS process, is able to turn on and off the PMOS FET with a delay less than 5 ns. The converter can operate with a frequency of 20 MHz and a duty cycle of less than 10 % while the conversion of a 40 V input voltage to a 4 V output voltage is possible.

ACKNOWLEDGEMENT

The work in this paper was sponsored by the Robert Bosch GmbH. Thanks to T. Rosahl and F. Sievers for the efficient cooperation and the constructive support.

REFERENCES

- [1] R. C. Jaeger, "Comments on 'An optimized output stage for MOS integrated circuits'," IEEE J. Solid-state Circuits, vol. SC-10, no. 3, pp. 185-186, June 1975.
- [2] Li, N.C.; Haviland, G.L.; Tuszynski, A.A.; , "CMOS tapered buffer," Solid-State Circuits, IEEE Journal of , vol.25, no.4, pp.1005-1008, Aug 1990
- [3] Ivan Sutherland, Bob Sproull, and David Harris. "Logical Effort." Morgan Kaufmann, Publishers, 1999.
- [4] F. Stettner, "Half-Bridge Powerstage for Integrated Synchronous Buck Converter." Diploma Thesis, Texas Instruments Deutschland, Technical University Munich, May 2009.



Juergen Wittmann received his Dipl.-Ing. Degree from the Technical University of Munich, Germany, in 2006. From 2006 until 2011, he was with the Mixed-Signal Automotive business unit of Texas Instruments, in Freising, Germany, as an Analog Design Engineer. In February 2011, he joined the Robert Bosch Center for Power Electronics at Reutlingen University as a research assistant.

He is working towards his Ph.D. degree in the area of power- and microelectronics.



Tobias Funk received the B. Eng. from the University of Applied Science Esslingen, Germany in 2011. From 2011 until 2012, he is a student at the master's program "Power Electronics and Microelectronic" at the Robert Bosch Center for Power Electronics.



Bernhard Wicht received the Dipl.-Ing. Degree from the Technical University Dresden, Germany in 1996 and the Ph.D. degree from the Technical University Munich, Germany, in 2002.

From 1996 to 1998, he was with MAZ Hamburg GmbH as a designer of analog ASICs for optical measurement systems. In 1998, he joined the Institute for Technical Electronics of the Technical University Munich, Germany, as a Research Assistant, where he was working on sense amplifiers until 2002 in cooperation with Infineon Technologies. From 2003 until August 2010, he was with the Mixed Signal Automotive business unit of Texas Instruments, in Freising, Germany, responsible for the development of automotive smart power ICs. In September 2010 he became a professor for integrated circuits at Reutlingen University, Robert Bosch Center for Power Electronics. His research interests include IC design with focus on power management, gate drivers, energy efficiency, low-power, ESD, EMC.

He invented 12 patents with several more patents pending. Prof. Wicht is member of IEEE and VDE. He also serves as a member of the Technical Program Committee of the European Solid-State Circuits Conference (ESSCIRC).

Realization of an RFID Front End IC for ISO 15693 Standard in UMC CMOS 0.18 μm Technology

Mayukh Bhattacharyya, Tobias Volk, Andreas Kreker, Benjamin Dusch, Dirk Jansen

Abstract—The RFID Front End manages communication and protocol according to the requirements of the ISO 15693 standard. It is developed for passive as well as semi-passive applications and interfaces to standard microprocessors, which can be programmed to capture data or execute commands in a general manner. The current consumption of the front end is below 50 μA , making it suitable for low power applications. This paper covers the analog part only. The reader generates an ASK-modulated signal, which is demodulated and provided as a digital signal. The 13.56 MHz clock is recovered and a signal, which detects the existence of RF field, is used. For the back channel, classical load modulation is implemented. There are further elements for passive applications integrated, a low drop out regulator LDO with adjustable output voltage and a reference voltage source with a bandgap circuit. This makes the front end IC suitable for energy harvesting sensor applications.

Index Terms—RFID (Radio frequency identification device), RF-field passive system, energy harvesting, bandgap reference, low drop out regulator, ASK demodulator.

I. INTRODUCTION

RFID technology is regarded as one of the evolutionary technology of the 21st century. It was initially used as a replacement for the existing barcode systems, over the years RFID has evolved as one of the key technologies for device identification. In addition to its original intention, the technology of bidirectional data communication available with RFID can again be used for ultra-low power or even passive micro sensor systems. There are applications in transportation, manufacturing, biomedical [1], environmental management, safety and security systems. The advantage of using an established standard, with low cost reader systems available in the market or even integrated into the newest mobile phones makes short range telemetry

Mayukh Bhattacharyya, mayukh.bhattacharyya@hs-offenburg.de, Tobias Volk, tobias.volk@hs-offenburg.de, Andreas Kreker, andreas.kreker@hs-offenburg.de, Benjamin Dusch, benjamin.dusch@hs-offenburg.de, and Dirk Jansen, d.jansen@hs-offenburg.de, are with University of Applied Sciences-Offenburg, Badstrasse 24, 77652 Offenburg.

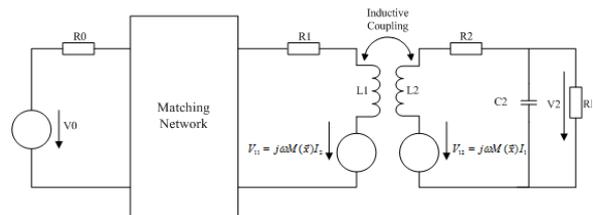


Figure 1: Inductive coupling for an RFID reader transponder system.

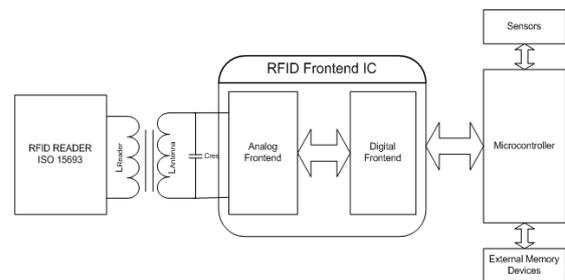


Figure 2: Front end with external devices.

cheap and ubiquitous available.

A typical RFID system consists of a reader and a transponder; the reader may be a smart phone or a host computer, connected to a network [2]. The tag consists of the front end (FE), which has an analog part and a digital part as well as a back end, which may be a data capturing system like a microprocessor, coupled with a sensor and external memory. In a fully developed tag, all these units will be integrated into one IC with only the sensors separated, making a so called “smart tag”, running optimally by harvesting the supply energy from the RF field.

Semi-active sensor tags contain an own supply, mostly a battery, because they have to be active without the connection to the reader, i.e. for collecting data at predefined intervals. These kinds of sensors have again several applications [3], the RFID interface is only used for convenient and extreme low power read out. The RFID reader/ transponder system can be typically defined as a loosely coupled transformer, where the antenna of the RFID reader acts as the primary and the antenna of the transponder acts as the secondary winding as shown in Figure 1. A typical RFID front end with external devices is shown in figure 2.

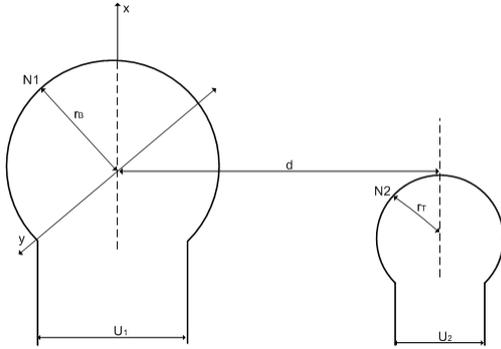


Figure 4: Model to calculate coupling factor between reader and tag antenna, both aligned.

The high power part consists of a power rectifier and a LDO regulator, both get activated when enough RF energy is available. The LDO provides V_{Out} , which can be adjusted by an external feedback resistor divider (not shown), connected to the analog input FB (feedback).

The antenna inputs are directly connected to the overvoltage protection circuit. This circuit is very critical because there are absolute limits for the voltage at the pads defined by the technology (breakdown voltages of the gates, about 3.6 V for the referenced technology). This circuit shall not be active under normal operation; it is only used in case of overvoltage input, which may happen because of the undamped resonance of the antenna, if the load at the output is switched off unintentionally.

The communication part consists of the passive rectifier, low pass filter, Schmitt-trigger, high pass filter, comparator and modulator. For clock recovery a RS flip flop is used. The circuit blocks are further described later in detail.

III. PROTECTION CIRCUIT

The protection circuit consists of the ESD protection of the pad cells (library) and additional circuits. The operation of a transponder in the magnetic field depends mainly upon two factors which are,

- Coupling factor (K)
- Quality factor (Q)

The coupling factor between the reader antenna and the tag antenna depends on geometrical parameters as shown in figure 4. The coupling factor K can be calculated by equation 1,

$$K(d) = \frac{\mu_0 \pi \cdot N_1 N_2}{2\sqrt{L_1 L_2}} \cdot \frac{r_B^2 \cdot r_T^2}{\sqrt{r_B^2 + d^2}} \quad (1)$$

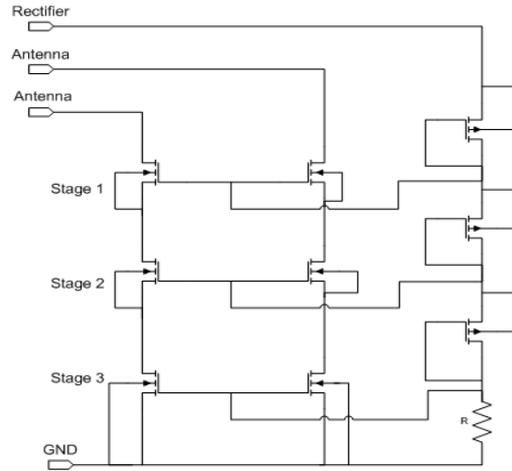


Figure 5: Overvoltage protection circuit.

Equation 1 shows that the coupling factor K depends on the antenna diameter of the reader r_B and tag r_T as well as the distance d in between them. Normally the value of K varies from 0.01 to 0.1. The higher the coupling factor is, the stronger is the field. The tag has to operate from field strength of 150 mA/m to 5 A/m, with 8 A/m maximum sustainable field strength (ISO 15693). This is not a problem, if the energy is consumed by a load.

Resonance of the antenna leads to an increase in amplitude at the input pins which is proportional to the quality factor (figure of merit) Q of the inductance/capacitance/load-circuit.

The quality factor Q is given by the relationship

$$Q = \frac{R_L}{2\pi f \cdot L_2} \quad (2)$$

where R_L is the equivalent load resistance of the transponder and L is the equivalent inductance of the transponder antenna. The read range increases with the Q of the antenna. It is recommended to be adjusted to ca. $Q = 15$ for good sensitivity. The voltage is directly proportional to the Q factor of the circuit [5].

Figure 5 shows the protection circuit. There are three stages of staggered transistors. The input voltage is divided by a resistive divider, providing the gate voltages for each stage. It is designed in such a way so that when the voltage on the antenna is beyond the maximum allowed limit, the transistors will switch on and hence draw power from the input, so the antenna is damped and the input voltage clamped to a safe value below the critical breakdown voltage of the transistors (ca. 3.7 V).

To protect the rest of the circuit, these transistors must be able to carry milliamps of current. To distribute the voltage load, the clamp transistors are placed in separate wells, so each transistor carries only a third of the input voltage. The switching thresholds are taken from a voltage divider, which is supplied by the power rectifier.

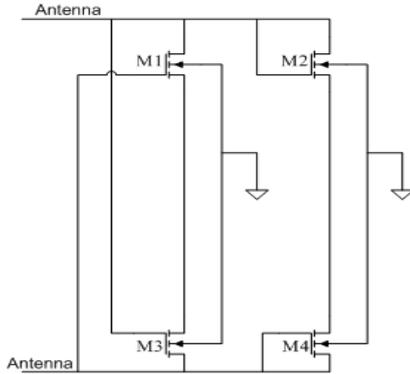


Figure 6: Rectifier with cross coupled NMOS transistors.

The power rectifier must be kept in the allowed operational area by the above circuit. The output of the rectifier is also used to supply the ESD protection circuits in the pads, which are MOS diodes connected to ground on one side and to the output of the power rectifier on the other side. The input voltages can by this never be larger than the output of the power rectifier plus the threshold of the diodes (ca. 0.7 V). So the output of the power rectifier is the highest voltage in the circuit.

Clamping of the antenna signals must be hard enough to protect the circuit against breakdown and disturbance. It is not expected that the demodulation works under such overvoltage conditions. In normal operation the protection circuit must be inactive and shall not draw any current. The circuit is designed for a hard clamping, starting at 3.5 V and limiting the voltage to 3.7 V under normal conditions.

IV. POWER SUPPLY

The power supply unit consists of the rectifiers, bandgap reference circuits and LDO regulator.

A. Rectifier

The power rectifier shown as block B10 in figure 3 is used to supply the low drop out regulator connected to it, which in turn is used to drive any external circuitry. The output is also used for load modulation, so for short intervals, the output voltage is loaded and this small interference has to be suppressed by the LDO as far as possible. A load capacitance at the external supply helps to bridge these intervals.

The communications chain has its own rectifier, supplying the bandgap-circuitry, clock recovery circuit, demodulation and field detection circuit. The comparator, used for demodulation, is supplied by the bandgap regulated voltage and by this independent on range and temperature influences.

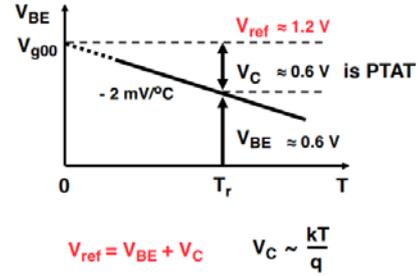


Figure 7: Curve showing PTAT and CTAT voltages [6].

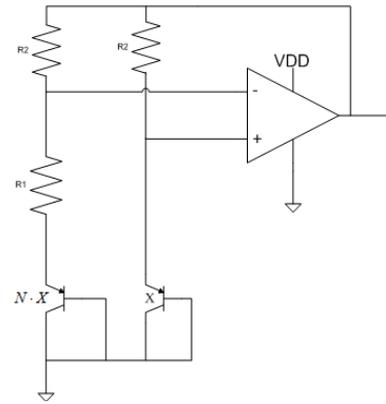


Figure 8: Band Gap reference (V_{ref}).

Both the rectifiers used are typical full wave bridge rectifiers which are built by using four cross connected NMOS transistors as shown in figure 6.

The (W/L) ratio of the rectifier transistors are made large (800:1 for block B10 in figure 3 and 200:1 for block B3 in figure 3 respectively) in order to get a small voltage drop and a good efficiency.

B. Bandgap reference

The purpose of a bandgap voltage reference is to deliver a fixed output voltage which is almost supply voltage and temperature independent. As shown in figure 7 the reference voltage is generated by adding two voltages, one having a positive and the other a negative temperature coefficient. When temperature increases, V_{BE} linearly decreases and V_T (thermal voltage) linearly increases, which makes the V_{ref} almost constant. The voltage obtained is first order independent of temperature. Figure 8 shows the type of band gap reference circuit used here. The reference voltage V_{ref} is given by the following equation,

$$V_{ref} = V_{BE} + \frac{V_T \cdot \ln(N)}{R_1} \cdot R_2 \quad (3)$$

where V_T is the thermal voltage which is 26 mV at 300 °K and N denotes the number of BJTs in parallel which is 8 in this design. Now if we consider a current I_x is flowing through the BJT X . The same current I_x is also flowing through N transistors connected in parallel, identical to X .

Table 1: Characteristic data of the band gap circuit.

Output voltage	$1.2 \pm 0.02 \text{ V}$
Load Current	$\leq 75 \mu\text{A}$
PSRR @ 10KHz	-26 dB
Current Consumption @1.9V	$4 \mu\text{A}$

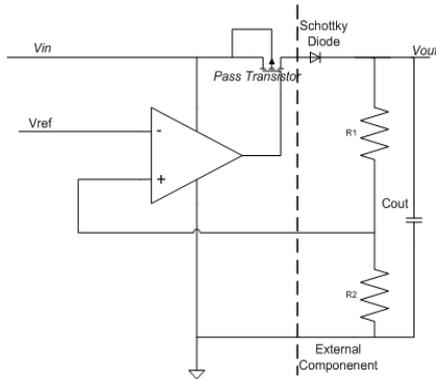


Figure 9: Low drop out rectifier (LDO).

So, the current of X is N times more than the current of the N BJT's connected in parallel which are identical to X . The value of R_2 depends upon the value of R_1 chosen.

Now in order to keep the current consumption small, the value of R_1 is set to $60 \text{ k}\Omega$ and the value of R_2 is $600 \text{ k}\Omega$ in order to get a factor of 10.

For a successful bandgap design there must be a compromise made between the current and the number of passive elements. Also the PSRR is important for a bandgap, because in an RFID application there are big changes in the available supply voltage. Further the PSRR must suppress the RF variations, because there is no space for filtering the input. The bandgap concept of figure 8 is preferred due to its high PSRR value in comparison to the other circuits, see [7]. Some characteristic data of the bandgap circuit is listed in table 1. Current consumption is only $4 \mu\text{A}$ at 1.9 V supply. The circuit can deliver a load of maximal $75 \mu\text{A}$ to the demodulation circuit. The output voltage is stable over temperature range of $-40 \text{ }^\circ\text{C}$ to $125 \text{ }^\circ\text{C}$ with a minimum V_{DD} supply of 1.9 V to a maximum V_{DD} of 3.6 V .

C. Low Drop Out Regulator

The low drop out regulator (LDO) provides a stable voltage supply for the external circuitry connected to the frontend. Figure 9 shows the standard configuration used for the LDO with external components separated by the dotted line. The voltage V_{in} is taken from the output of the power rectifier, and the reference voltage is taken from the bandgap reference. The LDO used is made with a PMOS pass transistor with a minimum dropout voltage of 150 mV . The output voltage is controlled by the resistor divider R_1 and R_2 ,

Table 2: Low drop out circuit.

Output Voltage Range	$1.2 \text{ to } 2.2 \text{ V}$
Quiescent Current	$1 \mu\text{A}$
Max Load Current	4 mA
Drop Out Voltage	150 mV
PSRR @ 4mA Load and 10KHz	-32 dB

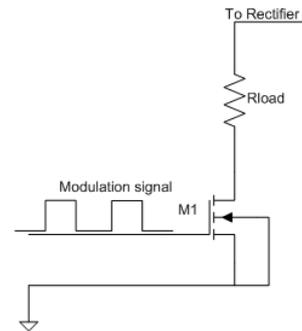


Figure 10: Load modulation circuit.

which are discrete devices to stay flexible for different applications. The external Schottky-diode is only used with semi-passive supply, where an external voltage is available.

The output voltage of the LDO can be adjusted between 1.2 V to 2.2 V by selecting the values of the resistors R_1 and R_2 .

Usually, the values of R_1 and R_2 are in mega ohms, so that the quiescent current of the LDO is kept as small as possible.

The PSRR of the error amplifier must be better than 32 dB in order to suppress any noise in the supply voltage [8]. The output voltage V_{out} is given by the equation

$$V_{out} = V_{ref} \cdot \left(1 + \frac{R_2}{R_1}\right) \quad (4)$$

The external capacitor C_{out} is needed to bridge and filter the output. The external Schottky diode prevents any reverse current flow in case V_{out} is greater than V_{in} . This may happen when a battery or a double layer capacitor ("gold cap") is used at the output for intermediate storage. Some characteristics data of the LDO are listed in table 2.

V. COMMUNICATION

The communication circuits are modulator, field detector circuit and demodulator.

A. Modulator

The modulator circuit used is a load modulation (on-off amplitude shift keying) circuit for sending information back to the reader as shown in figure 10. The

Table 3: Load modulation circuit.

Series Resistance	300 Ω
Current (max)	4 mA
ON Time	< 10 ns
OFF Time	< 10 ns
Leakage Current	< 1 μA

Table 4: Schmitt trigger circuit.

Current Consumption (static)	780 nA @ 1.9 V
ON Time	300 \pm 50 μs
OFF Time	300 \pm 50 μs
Threshold	0.9 V \pm 0.1 V

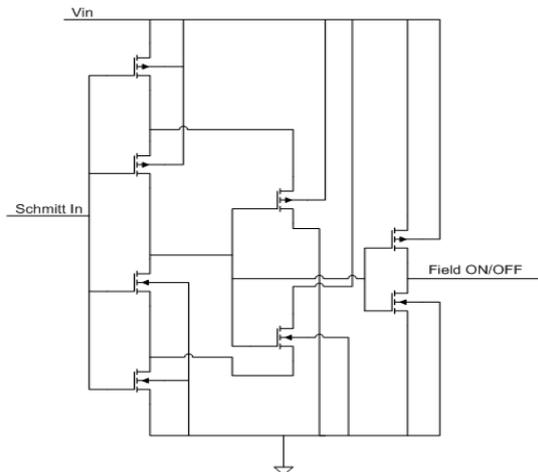


Figure 11: Schmitt trigger circuit.

modulation signal is obtained from the digital part of the front end. Transistor M_1 , in figure 10 acts as a switch. The load resistor R_{load} works as a damping to the antenna via the rectifier. This can be sensed by the reader because of the coupling via the reader tag coils. Table 3 contains some characteristic data regarding the load modulation circuit. The currents may be quite large (up to 4 mA) by direct coupling (small distances), so the layout of the circuit must be accomplished with adequate dimensions of resistor and routing.

B. Field detector circuit

The Schmitt trigger circuit is used to detect the field. A low pass filter is used along with the Schmitt trigger as shown in figure 3. The low pass filter is used to suppress noise and short spikes, so that a minimum of field energy must be available before the output is going high. The circuit used here, as shown in figure 11, is a classical Schmitt trigger circuit, where the threshold is controlled by W/L relations.

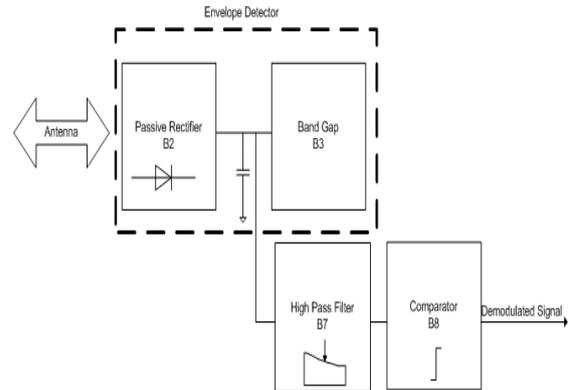


Figure 12: Block diagram showing the envelope detection for demodulator.

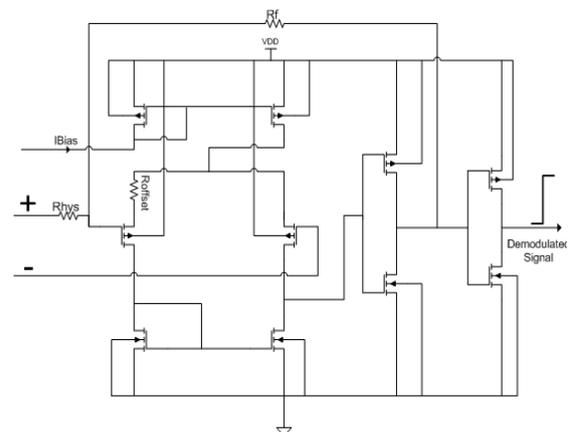


Figure 13: Comparator Circuit.

The threshold is adjusted in such a way that the output only signals a field if there is enough energy to supply the bandgap circuit and the demodulator. The static current consumption is below 1 μA . Table 4 shows the characteristic data.

C. Demodulation

The circuit demodulates the ASK signal received from the reader. The circuit comprises of an envelope detector with a load capacitor, followed by a high pass filter and a comparator. The rectifier shown in figure 12 and the load capacitance make the envelope detector. The needed resistive load for the envelope detector comes from the power consumption of the bandgap circuit. The high pass filter strips the modulation from the envelope and the comparator generates the digital signal.

The comparator circuit used is a standard PMOS single-stage comparator as shown in figure 13. The comparator is supplied by the internal power supply of 1.2 V, available from the bandgap reference. The input is referenced to ground, so the comparator must be designed in such a way that ground is inside the allowed input range. A hysteresis of 4 mV is provided to suppress noise. An offset voltage of 25 mV is built

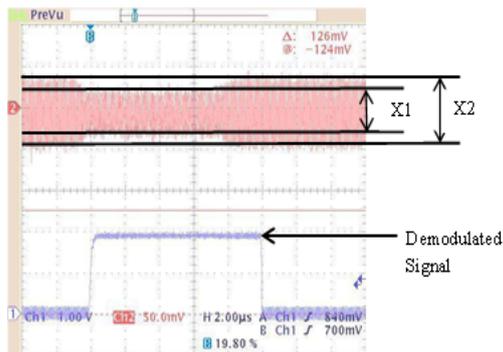


Figure 14: Signal showing modulation and demodulation.

Table 5: Demodulator circuit.

Current Consumption at 1.2 V	15 μ A
Offset Voltage	25 mV
Hysteresis	4 mV
Propagation Delay	490 ns

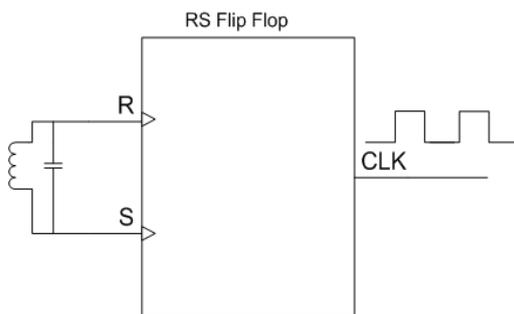


Figure 15: Simplified diagram showing clock recovery using RS flip flop.

in by using a resistor in the source connection of one of the transistors of the differential pair; this generates a definite zero output, if there is no field or only a very low field. In some way this offset is responsible for the sensitivity of the receive chain and defines the maximum distance the tag is able to work.

A typical received signal with 10% ASK modulation and the related demodulated signal is shown in figure 14. The propagation delay of the circuit measured is 0.49 μ s. The characteristic data of the comparator circuit are enlisted in table 5.

VI. CLOCK RECOVERY

The clock recovery circuit consists of an RS flip flop which is used to extract the 13.56 MHz clock from the input signal as shown in a simplified manner in figure 15. The inputs of the clock generator are two phase inversed antenna signals as shown in figure 3 which are ANT_P and ANT_N . The clock signal can be obtained from the input signal by the RS flip flop. The carrier frequency is 13.56 MHz, so the current consumption is relatively large.

Table 6: Clock recovery circuit.

Operating Voltage	1.2 V \pm 0.2 V
Duty cycle (expected 50 \pm 5 %)	53.5 %
Current Consumption	2.36 μ A

Table 7: Overall feature of the frontend.

Technology used	UMC CMOS 0.18 μ m
Size of the chip	1.5 mm \times 1.5 mm
Bandgap reference voltage	1.2 V
Temperature range	-40 – 125 $^{\circ}$ C
Output range of Low drop out	1.2 – 2.2 V
Low power block current consumption	22 μ A
Total power consumption (at no external load)	30 μ A

As shown in table 6, a duty cycle of 53.5 % is obtained by using this circuit which is well within the tolerance of (50 \pm 5%).

VII. CONCLUSION

An analog front end circuit for ISO/IEC 15693 is proposed with very low current consumption below 50 μ A and a separate power supply chain for passive sensor supply. Table 7 includes all the important features of the proposed frontend. A bandgap reference provides a stable operating condition for demodulator and LDO. The circuit discussed here, is designed to work with active, semi-active or passive supply sensor applications. The analog front end, along with the digital part, integrated with a processor and some sensors, is well suited for several industrial or medical applications [9], [10].

ACKNOWLEDGEMENT

We like to thank all the colleagues from ASIC design center and Institute for Applied Research Offenburg, for their support and coordination in due course of this work.

REFERENCES

- [1] W. Jeon, J. Melngailis, R. W. Newcomb, "Disposable CMOS passive RFID transponder for patient monitoring", *ISCAS, Greece*, May 2006.
- [2] icarte, RFID reader for iphone, *Wireless Dynamics Inc*, November 2009.
- [3] R. Jedderman, L. R. Garcia, W. Lang, "Spatial temperature profiling by semi-passive RFID loggers for perishable food transportation", *Journal Elsevier*, August 2008.
- [4] L. D. Sheng, Z. X. Cheng, Y. Q. ping, X. T. wen, "An analog front-end circuit for ISO/IEC 15693-compatible RFID transponder IC", *Journal of Zhejiang University Science A*, May 2006.
- [5] C. Klappf, A. Missoni, G. Hofer, G. Holweg, W. Kargl, "Improvements in operational distance in passive HF RFID transponder systems", *RFID IEEE international conference Las Vegas*, April 2008.
- [6] W. Sansen, "Analog design Essentials", ISBN 0-387-25746-2, Year 2006.
- [7] A. Monteiro, M. Santos, B. Borges, F. Lima, "High PSRR, Low Power Bandgap", *Chipidea MIPS*, (Year not available).
- [8] J. Heidrich, D. Brenk, J. Essel, M. Heinrich, M. Jung, G. Hofer, G. Holweg, R. Weigel, G. Fischer, "Design of a low-power voltage regulator for RFID applications", *IEEE Region 8 SIBIRCON-2010, Irkutsk Listvyanka, Russia*, July 2010.
- [9] T. Feldengut, "Energieübertagung für passive Sensor-Transponder mit hoher Reichweite", *PhD thesis, University of Duisburg-Essen*, October 2010.
- [10] B. Jiang, J. R. Smith, M. Philipose, S. Roy, K. S. Rajan, A. V. Mamishev, "Energy Scavenging for Inductively Coupled Passive RFID systems", *IEEE Transactions on instrumentation and measurement*, February 2007.



Mayukh Bhattacharyya was born in the year 1985. He received his Bachelor of Engineering degree in Electrical and Electronics Engineering from Visvesvaraya Technological University (Belgaum, India) in the year 2007. He finished his Master of Science in Communication and Media Engineering in University of Applied Sciences, Offenburg (Germany) in the year 2010. Since 2010 he is working as a Research Assistant with Prof Dr.-Ing Dirk Jansen in IAF Offenburg. His main area of interest for his work is development of ultra-low power systems using RFID technology. At present he is preparing his Ph.D. thesis with University of Freiburg (IMTEK) under a program called KleE.



Tobias Volk was born in the year 1981. He received his diploma and master degree in electrical engineering from the University of Offenburg (Germany). Since 2006, he is working as development engineer for the IAF Offenburg. His research interests include low power electronics and RFID devices in medical environments. Now he is preparing his Ph.D. thesis in cooperation with the University of Freiburg.



Dirk Jansen was born in the year 1948. He received his diploma degree in electrical engineering from the TH Darmstadt (Germany) 1972 and his Ph.D. from the University of Stuttgart 1979. Since 1986, he is professor at the University of Applied Sciences Offenburg. In 1995 he became director of the Institute for Applied Research Offenburg. He is IEEE, EURO PRACTICE and VDE member. His research interests include ASIC design (digital and analogue), system design in VHDL and inductive data transmission techniques.

Hardware-Entwurf einer flexiblen Fehlerkorrektureinheit für Flashspeicher

Jens Spinner, Jürgen Freudenberger

Zusammenfassung—Die Bedeutung von Flash-Speicherchips als Permanentspeicher nimmt stetig zu. Dabei sinkt jedoch mit zunehmender Speicherdichte die Zuverlässigkeit der Speicherzellen, so dass eine Codierung zur Fehlerkorrektur ein unabdingbarer Bestandteil eines Flash-Controllers ist. Die Anforderungen an die Fehlerkorrektur hängen dabei stark von der Flash-Technologie (Multi-Level-Cell oder Singel-Level-Cell) ab. Daher muss beim Hardware-Entwurf der Korrektur-einheit auf eine möglichst große Flexibilität der Codierung geachtet werden. In diesem Beitrag wird eine Decoder-Architektur beschrieben, die effizient die unterschiedlichen Ansprüche an die Codierung berücksichtigt und so eine hohe Flexibilität und große Fehlerkorrekturfähigkeit bei einem moderaten Flächenbedarf ermöglicht. Die beschriebene Fehlerkorrektureinheit wurde in Verilog umgesetzt und auf einem FPGA getestet.

Schlüsselwörter—BCH-Codes, Flashspeicher, Berlekamp-Massey, flexibler Decoder.

I. EINLEITUNG

Seit einigen Jahren existieren Flashspeicher auf der Basis von Single-Level-Cells (SLC), die geringe Anforderungen an die Fehlerkorrektur stellen. Zwischenzeitlich wächst die Speicherdichte durch die Verkleinerung der Strukturgröße und das Speichern mehrerer Zustände pro Speicherzelle (MLC, TLC) in den Flashspeichern an, wodurch auch die Bitfehlerraten ansteigen. Diese Fehler werden heute üblicherweise mithilfe von leistungsfähigen Codes wie Bose-Chaudhuri-Hocquenghem (BCH) oder Reed-Solomon (RS) Codes [1] korrigiert. Die Fehlerkorrektureigenschaften eines Error-Correction-Codes (ECC) muss an die Anforderungen der Anwendung und die Eigenschaften des verwendeten Flashspeichers angepasst werden.

Jens Spinner, jens.spinner@htwg-konstanz.de und Jürgen Freudenberger, juergen.freudenberger@htwg-konstanz.de, sind Mitglieder der HTWG-Konstanz, Institut für Systemdynamik, Braunerstrasse 55, 78462 Konstanz.

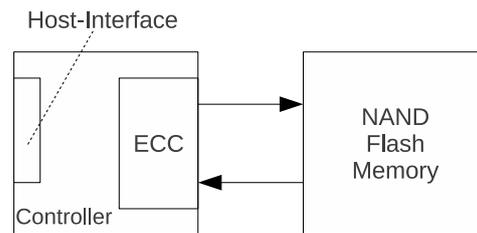


Abbildung 1: ECC-Einheit und NAND-Flash.

Die ECC-Einheit ist Teil eines Flash-Controllers, der meist noch andere Module beinhaltet, so zum Beispiel Datenschnittstellen wie USB, SATA und Verschlüsselungsalgorithmen wie AES. Die Beziehung zwischen NAND-Flash-Speichern und ECC-Unit ist Abbildung 1 zu entnehmen. Die ECC-Einheit kann weiter in Encoder und Decoder aufgegliedert werden. Der Encoder schreibt die Information in den Flashspeicher und fügt Redundanz für die Fehlerkorrektur hinzu. Beim Auslesen werden im Decoder fehlerhaft gelesene Informationen mithilfe dieser Redundanz rekonstruiert.

Die Analyse einer großen empirischen Datenbank [2] von Fehlermustern zeigt, dass Flashspeicher nur über eine geringe Burstfehlerrate verfügen. Es kann davon ausgegangen werden, dass die Fehler unabhängig auftreten. Aus diesem Grund verwendet man als Grundlage für die Analyse den binären symmetrischen Kanal (BSC) als Kanalmodell. Auf dem BSC wird ein Bit mit der Wahrscheinlichkeit ε verändert, wobei der Übergang von einer Eins zu einer Null dieselbe Wahrscheinlichkeit besitzt, wie der Übergang von einer Null zu einer Eins. Für diesen Kanal sind BCH-Codes eine effiziente Lösung, da diese unabhängige Bitfehler besser korrigieren als Burstfehler.

Die algebraische Decodierung von BCH-Codes kann in vier Schritte unterteilt werden, die üblicherweise in der Hardwareimplementierung in unterschiedlichen Modulen realisiert werden, siehe [3].

1. Syndromwertberechnung.
2. Berechnung des Fehlerstellenpolynoms.
3. Evaluation des Fehlerstellenpolynoms.
4. Korrektur der Fehler anhand der ermittelten Fehlerstellen.

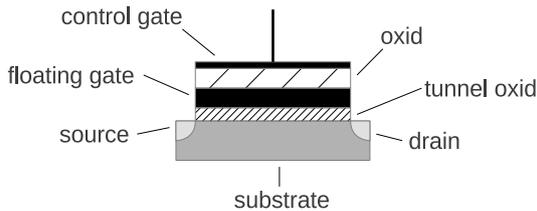


Abbildung 2: Floating Gate.

Den größten Rechenaufwand und Flächenbedarf benötigen die Schritte 2 und 3. Jeder Decodierschritt kann jeweils seriell oder parallel implementiert werden. Eine parallele Implementierung erhöht den Durchsatz aber auch den Flächenbedarf, wohingegen eine reine serielle Implementierung für viele Anwendungen zu langsam wäre. Die Chien-Search eines BCH-Codes der Länge n benötigt n Schritte, in denen jeweils das gesamte Fehlerstellenpolynom ausgewertet wird. Diese Nullstellensuche kann beliebig parallelisiert werden. Das bedeutet, dass durch den Parallelisierungsgrad der Implementierung eine geeignete Wahl zwischen dem Durchsatz und dem Flächenverbrauch getroffen werden kann.

Die Berechnung des Fehlerstellenpolynoms wird üblicherweise mit dem Berlekamp-Massey-Algorithmus (BMA) durchgeführt, dieser hat mit $O(t^2)$ eine geringere Komplexität als das Gaußverfahren mit $O(t^3)$, wobei t die maximale Anzahl korrigierbarer Fehler angibt. Eine Zusammenstellung von verschiedenen BMA-Implementierungen ist in [4] beschrieben. In [4] sind darüber hinaus auch die Komplexitäten im Bezug auf die Zyklenzahl und der Anzahl von Multiplizierern und Addierern aufgeführt. Die erste serielle Implementierung für einen BMA wurde von Blahut [5] vorgestellt. Die serielle Implementierung benötigt lediglich 3 Multiplizierer aber $2t^2$ Iterationen. Nach unserem Kenntnisstand existiert bislang keine gemischte serielle/parallele Implementierung eines BMA.

In diesem Artikel beschreiben wir, wie eine flexible ECC-Einheit bezüglich der Parameter Fläche, Durchsatz und Fehlerkorrekturfähigkeit optimiert werden kann. Insbesondere betrachten wir dabei eine BMA-Implementierung, die einen guten Kompromiss zwischen Durchsatz und Fläche bietet. Dabei wird eine Mischung aus einer parallelen und seriellen Implementierung vorgeschlagen. Die parallele Implementierung korrigiert maximal $t_1 < t$ Fehler. Hierfür benötigt der parallele BMA lediglich $2t_1$ Multiplizierer und t_1 Iterationen. Wird die Fehleranzahl t_1 überschritten, so führt der serielle BMA die Decodierung bis zu seinem Limit t fort.

II. FLASHSPEICHER

Flashspeicher bestehen aus einer Vielzahl von Speicherzellen, den sogenannten Floating-Gate-Zellen (siehe Abbildung 2). Floating-Gate-Zellen sind in ihrer Grundstruktur MOSFETs, die allerdings durch ein Floating-Gate (FG) zwischen dem Control-Gate (CG) und dem Substrat erweitert sind. Dieses Floating-Gate ist vollkommen von einem Dielektrikum umgeben. Der Raum zwischen Control- und Floating-Gate ist mit einem breiten Dielektrikum ausgefüllt. Zwischen dem Floating-Gate und dem Substrat allerdings ist ein relativ feines Dielektrikum. Dadurch, dass das Floating-Gate vollkommen von Dielektrikum umgeben ist, können im Normalfall keine Ladungsträger in das Floating-Gate hinein oder heraus gelangen. Das Floating-Gate behält also seinen Ladungszustand, ohne dass Energie aufgewendet werden muss.

Durch das Anlegen einer Programmier- bzw. Löschspannung am Control-Gate und dem Substrat zwischen Source und Drain entsteht ein elektrisches Feld, das die Elektronen veranlasst, die Barriere der Oxidschicht zu überwinden. Diesen Effekt nennt man auch Fowler-Nordheim Tunneleffekt.

Die Ladung des Floating-Gates beeinflusst den Stromfluss zwischen Source und Drain, wodurch der Zustand ausgelesen werden kann. Dieser Strom wird mithilfe eines A/D-Wandlers in Bitwerte gewandelt. Dabei hat der A/D-Wandler je nach Technologie eine Auflösung von einem (bei SLC), zwei (MLC) oder drei (TLC) Bit.

Es ist jedoch nicht immer gewährleistet, dass der ins Floating-Gate geschriebene Wert dem später ausgelesenen Wert entspricht. Dafür gibt es eine Reihe von Gründen. Die wesentliche Ursache für Fehler ist die Abnutzung des Speichers durch Schreib- und Löszyklen. Beim Schreiben und Löschen kann die Oxidschicht dauerhaft beschädigt werden. Die Fehleranfälligkeit steigt daher kontinuierlich mit zunehmender Anzahl an Schreib- und Löszyklen. Auch durch äußere Strahlungseinflüsse und durch Lesevorgänge können die ursprünglichen Ladungszustände verändert werden [6].

Die Speicherzellen sind in einer NAND-Struktur in Pages und Blöcke in einer Matrixform (siehe Abbildung 4) angeordnet. Dabei sind mehrere Zellen von Drain nach Source in einer Zeile in Serie geschaltet und bilden eine Bitline (BL). Die Control-Gates derselben Zeile sind mit einer Wordline (WL) verbunden, die über einen Row-Decoder adressiert werden. Der Page-Buffer beinhaltet die Mechanismen zum Beschreiben und Lesen einer Page. Pages können einzeln beschrieben werden. Bei einem Löschvorgang wird allerdings immer ein kompletter Block zurückgesetzt.

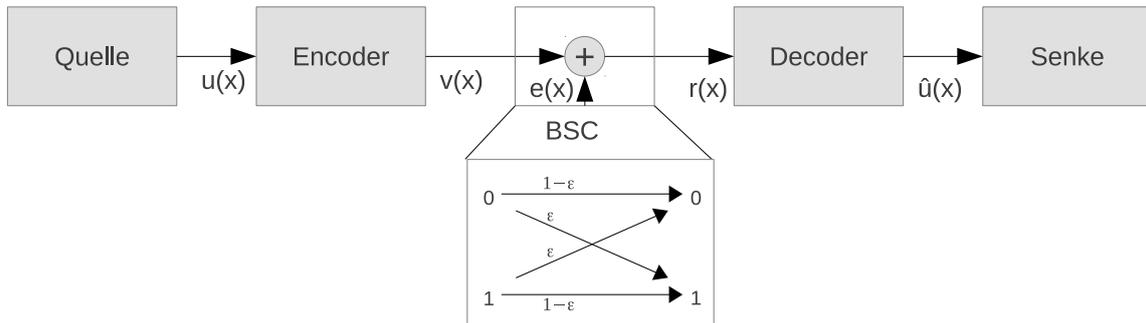


Abbildung 3: Kanalmodell.

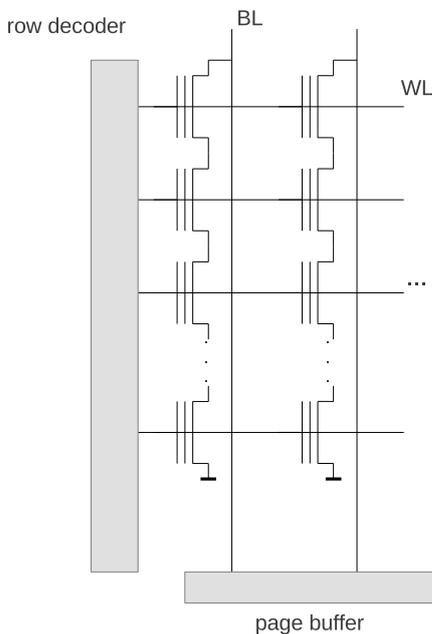


Abbildung 4: Anordnung der Floating-Gates in NAND-Speichern.

III. KANALCODIERUNG

In Abbildung 3 ist das Kanalmodell dargestellt. Ein Codewort $v(x)$ besteht aus n Bits, die aus k Informationsbits erzeugt werden. Bei zyklischen Codes werden diese Vektoren als Polynom dargestellt. Üblicherweise wird das Codewort aus

$$v(x) = u(x)x^{n-k} + u(x)x^{n-k} \bmod g(x)$$

berechnet, wobei die Information $u(x)$ unverändert durch $u(x)x^{n-k}$ enthalten ist und die Redundanz mit $u(x)x^{n-k} \bmod g(x)$ angehängt wird. Da die Information durch diese Methode direkt auslesbar ist, spricht man von einer systematischen Codierung. $g(x)$ ist das Generatorpolynom über dem Galois-Feld $GF(2)$, dessen Eigenschaft es ist, dass es $\alpha, \alpha^2, \dots, \alpha^{2^t}$ als Nullstellen besitzt. α ist das primitive Element des Galois-Feld $GF(2^m)$. t ist die Anzahl der korrigierba-

ren Bitfehler. Bei dem BCH-Code bewirkt das Generatorpolynom

$$S_i = v(\alpha^{i+1}) = 0, i = 0, \dots, 2t - 1.$$

Wenn Fehler auftreten, ist das ausgelesene Wort $r(x)$ meist kein gültiges Codewort. Dies drückt sich in der Verletzung von $S = 0$ aus. Diese Werte sind die sogenannten Syndrome. Diese Syndrome haben die Eigenschaft, dass sie lediglich vom Fehler abhängen. Nun besteht die Idee darin, dass ein $\sigma(x)$ mit dem kleinstmöglichen Grad bestimmt wird, das multipliziert mit $S(x)$ Null ergibt. Dies wird bestimmt durch

$$\begin{aligned} \sigma(x) &= \sigma_0 + \sigma_1 x + \dots + \sigma_e x^e \\ &= (1 - \beta_1 x)(1 - \beta_2 x) \dots (1 - \beta_e x) \end{aligned}$$

Diese sogenannte Schlüsselgleichung ergibt ein lineares Gleichungssystem, das eindeutig gelöst werden kann, wenn t oder weniger Fehler aufgetreten sind. Die Lösung $\sigma(x)$ wird Fehlerstellenpolynom genannt. Die Nullstellen des Fehlerstellenpolynoms zeigen wiederum die erkannten Fehlerstellen an. Die Schlüsselgleichung wird mit dem BMA gelöst. Die Evaluation des Fehlerstellenpolynoms geschieht in der Chien-Search. Dieser Ablauf ist in Abbildung 5 schematisch dargestellt.

IV. VARIABLER-BCH-ENCODER

Das Codewort wird durch

$$v(x) = u(x)x^{n-k} + u(x)x^{n-k} \bmod g(x)$$

erzeugt. Implementiert wird dies mithilfe eines rückgekoppelten Schieberegisters. Die Anzahl der im Generatorpolynom $g(x)$ enthaltenen aufeinanderfolgenden irreduziblen Polynome bestimmt die Anzahl der korrigierbaren Fehler t und die Anzahl der benötigten Redundanzstellen. Da die Page-Größen der Flashspeicher variieren, möchte man zwischen verschiedenen Generatorpolynomen wählen können, um die Speicherzellen möglichst effizient zu nutzen und optimal gegen Fehler zu schützen. Die in [7] präsentierten Logikelemente, die verantwortlich für die Abgriffe des Schieberegisters sind, können in einer Implementierung für eine Auswahl von Generatorpolynomen realisiert werden. Die Grundstruktur des Schieberegisters muss dabei nicht vervielfältigt werden. Die Umschal-

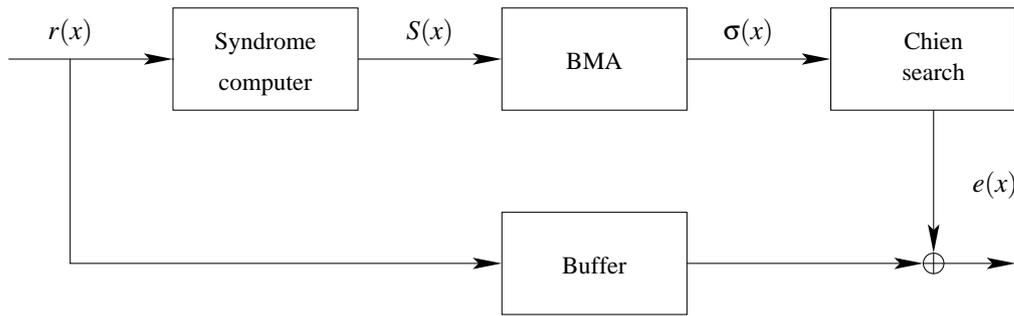


Abbildung 5: Decodierschema eines BCH-Codes.

tung zwischen den Generatorpolynomen geschieht durch Multiplexer, die für jede Fehlerkorrekturfähigkeit die entsprechenden Logikelemente in der Rückkopplung dazuschalten.

V. ALGEBRAISCHE-DECODIERUNG

Im Folgenden wird der in Abbildung 5 dargestellte Ablauf bestehend aus der Syndromberechnung, der Lösung der Schlüsselgleichung mithilfe des BMA und der Bestimmung der Fehlerstellen durch die Chien-Search beschrieben.

A. Syndromberechnung

Die Syndromberechnung kann als eine diskrete Fouriertransformation (DFT) über dem GF verstanden werden. Durch das Einsetzen der Werte $\alpha, \alpha^2, \dots, \alpha^{2t}$ in $S_i = r(\alpha^i)$, werden die einzelnen Syndrome S_i ermittelt. In Matrixform kann die parallele Implementierung veranschaulicht werden. Sie lautet

$$S = r \cdot H^T$$

mit

$$H^T = \begin{bmatrix} 1 & 1 & \dots & 1 \\ \alpha & \alpha^2 & \dots & \alpha^{2t} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha^{n-1} & \alpha^{2(n-1)} & \dots & \alpha^{2t(n-1)} \end{bmatrix}$$

In diesem Fall werden alle Syndromwerte parallel berechnet. Die Hardwareimplementierung verknüpft je Zyklus ein Empfangsbit mit einer vollständigen Zeile und benötigt durch diese Methode genau n Iterationen. Je Zyklus können auch mehrere Zeilen gleichzeitig berechnet werden, üblich sind z.B. 8-Bit. Die Anzahl der Zyklen reduziert sich dadurch auf $\left\lceil \frac{n}{8} \right\rceil$. Das t

richtet sich nach der maximalen Fehlerkorrekturfähigkeit. Eine Berücksichtigung der tatsächlichen Anzahl von korrigierbaren Fehlern findet nicht im Modul der

Syndromwertberechnung sondern im BMA und in der Chien-Search statt.

B. Zweistufiger BMA

Die Grundidee des zweistufigen BMA ist die Implementierung eines parallelen BMA der bis $t_1 < t$ Fehler korrigiert. Für den parallelen BMA werden lediglich die ersten $2t_1$ Koeffizienten des Syndroms benötigt. Ist die Anzahl der Fehler e kleiner oder gleich t_1 , so liefert diese erste Stufe ein gültiges Codewort $v(x)$. Ist $t_1 < e \leq t$, wird der parallele BMA üblicherweise keine Lösung finden, und die Berechnung des Fehlerstellenpolynoms wird in einer seriellen Implementierung fortgeführt. In diesem Fall muss der parallele BMA um eine Prüfung erweitert werden, die erkennt, ob ein gültiges Codewort gefunden wurde. In Abbildung 6 ist die Architektur eines zweistufigen BMA dargestellt.

Findet der parallele BMA eine Lösung für die Schlüsselgleichung, so ist die Bedingung

$$S_i = v(\alpha^{i+1}) = 0, i = 0, \dots, 2t_1 - 1$$

erfüllt. Die gemischte Implementierung benötigt zusätzliche Informationen, um sicherzustellen, dass $\hat{v}(x)$ ein gültiges Codewort ist. Im Folgenden werden wir hierfür zwei Verfahren vorstellen.

Eine einfache Methode ist die Implementierung einer Syndromberechnung für die verbleibenden Syndrome. Die Syndromberechnung prüft

$$S_i = \hat{v}(\alpha^{i+1}) = 0, i = 2t_1, \dots, 2t - 1.$$

Ist diese Bedingung erfüllt, so ist $\hat{v}(x)$ ein korrektes Codewort. Wenn nicht, wird der serielle BMA hinzugezogen, um das komplette Syndrom zu lösen.

Die zweite Methode ist eine Modifizierung des BMA. Während jeder Iteration wird im BMA die sogenannte Diskrepanz

$$\Delta^{(j)} = \sum_{i=0}^{L^{(j)}} S_{j-i} \sigma_i$$

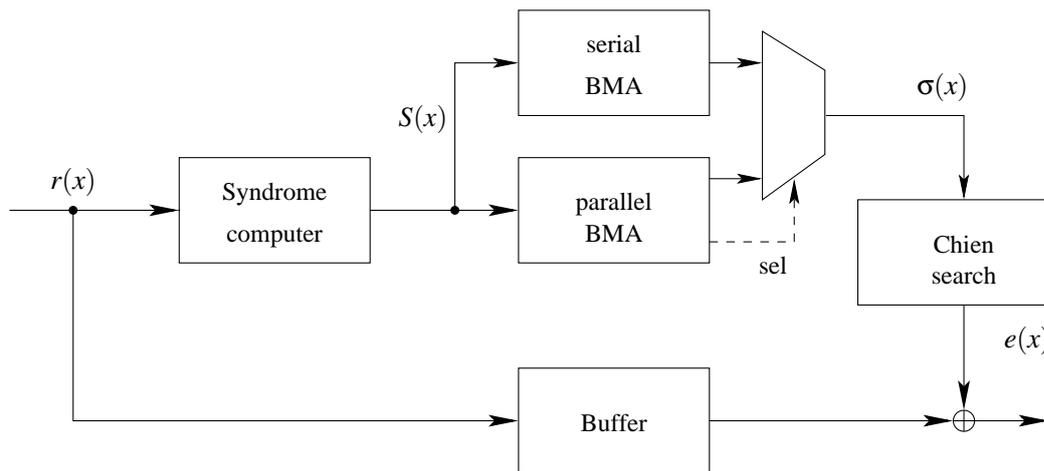


Abbildung 6: Decoder Architektur für zweistufigen BMA.

berechnet. $j = 0, 1, \dots$ ist ein Iterationszähler und $L(j)$ ist der Grad des Fehlerstellenpolynoms in der j -ten Iteration. Ist die Anzahl der Fehler kleiner oder gleich t_1 , wird die Diskrepanz Null für alle Iterationen $j \leq 2t_1 - 1$. Ist die Diskrepanz $\Delta(j)$ Null für alle Iterationen $j = 2t_1 - 1, \dots, 2t - 1$ so haben wir ein gültiges Codewort $\hat{v}(x) \in C$.

Da die Berechnung der Diskrepanz Teil des BMA ist, kann die parallele Implementierung auch für die Prüfung in den Iterationen $j \geq 2t_1 - 1$ verwendet werden. Die parallele Implementierung des BMA benötigt daher lediglich $2t_1$ Multiplikationen, die bei einem $GF(2^{14})$ jedoch eine sehr hohe Komplexität besitzen.

C. Durchschnittliche Anzahl von Iterationen

Die Anzahl der Iterationen eines parallelen oder seriellen BMA ist konstant und wird durch die Fehlerkorrekturfähigkeit festgelegt. Bei der gemischten parallelen und seriellen Implementierung hängt die Anzahl der Iterationen von der tatsächlichen Anzahl der Fehler ab. In diesem Abschnitt beschreiben wir, wie die durchschnittliche Anzahl von Iterationen bestimmt werden kann. Des Weiteren zeigen wir, wie man ein optimales t_1 bestimmt.

Sei $p(t_1, \varepsilon)$ die bedingte Wahrscheinlichkeit, dass die Anzahl der aufgetretenen Fehler größer als t_1 ist, gegeben dass mindestens ein Fehler auftrat. Die bedingte Wahrscheinlichkeit hängt von t_1 und der Bitfehlerwahrscheinlichkeit ε des BSC ab. Die durchschnittliche Anzahl der Iterationen ist dann

$$\bar{N} = \left\lfloor \frac{t + t_1}{2} \right\rfloor + p(t_1, \varepsilon)t^2$$

für den binären BCH-Code und

$$\bar{N} = t + t_1 + 2p(t_1, \varepsilon)t^2$$

für den nicht binären Code. Für den BSC kann $p(t_1, \varepsilon)$ angenähert werden durch

$$p(t_1, \varepsilon) \approx \frac{\sum_{i=t_1+1}^n \binom{n}{i} \varepsilon^i (1-\varepsilon)^{n-i}}{1 - (1-\varepsilon)^n}$$

wobei der Zähler die Wahrscheinlichkeit ist, dass die Anzahl der Fehler größer t_1 ist. Der Nenner ist die Wahrscheinlichkeit, dass mindestens ein Fehler auftritt.

Die mittlere Anzahl von Iterationen sollte die Bedingung $\bar{N} \leq 2t$ erfüllen, um zu gewährleisten, dass der mittlere Durchsatz nicht reduziert wird.

Um die Performanz einer parallel/seriellen BMA Implementierung darzustellen, betrachten wir ein typisches Beispiel für Flashspeicher. Um eine Sektorgröße von 1kByte zur Verfügung zu stellen, benötigen wir eine Codedimension $k = 8288$. Dies beinhaltet die Datenbits sowie zusätzliche Bits, die vom Flash-Controller benötigt werden. Für die Zuverlässigkeit gilt die Anforderung, dass ein Sektor mit einer Wahrscheinlichkeit von höchstens $p_{block} \leq 10^{-16}$ fehlerhaft sein darf. In diesem Beispiel betrachten wir nun drei verschiedene BCH-Codes mit unterschiedlichen Fehlerkorrekturfähigkeiten. Die Parameter sind in Tabelle 1 zusammengefasst. Da die Bitfehlerwahrscheinlichkeit mit zunehmender Anzahl der Schreib- und Löschyklen zunimmt, betrachten wir die höchste tolerierbare Bitfehlerwahrscheinlichkeit. In Tabelle I ist der Wert ε_t die maximale Bitfehlerwahrscheinlichkeit, bei der die Blockfehlerwahrscheinlichkeit von 10^{-16} eingehalten werden kann. Der Wert $t_{1,opt}$ ist der optimale Wert für t_1 , bei dem die mittlere Iterationsanzahl minimal ist. Die Verbesserung im Vergleich zu einer parallelen Implementierung ist in Abbildung 7 dargestellt. Diese zeigt die Reduktion der Iterationsanzahl $t - \bar{N}$ für verschiedene Werte von t_1 und t .

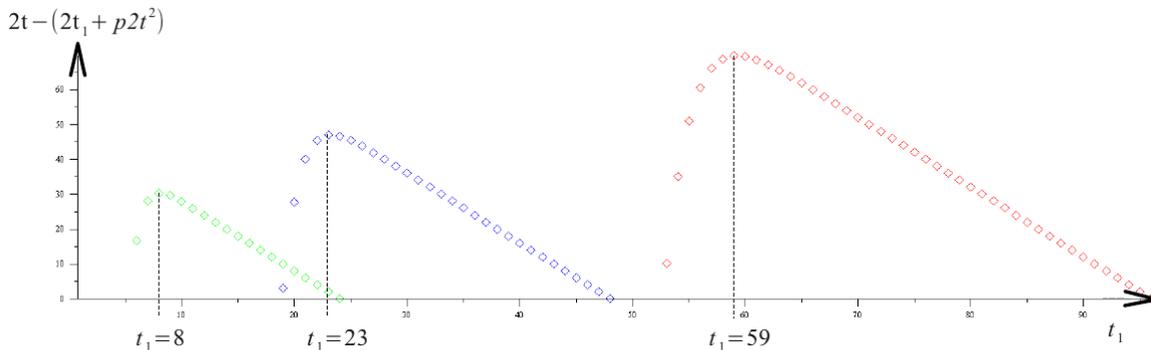


Abbildung 7: t - N für die drei BCH-Codes aus der Tabelle.

Tabelle 1: Parameter dreier verschiedener BCH-Codes für Flashspeicher mit 1 kByte Sektorgröße.

t	24	48	96
n	8624	8960	9623
ε_t	$3 \cdot 10^{-4}$	$1,3 \cdot 10^{-3}$	$3,8 \cdot 10^{-3}$
$p(t_1, \varepsilon_t)$	$1,4 \cdot 10^{-3}$	$6,5 \cdot 10^{-4}$	$2,3 \cdot 10^{-4}$
$t_{t_1, opt}$	8	23	59

D. Chien-Search

Die Fehlerpositionen sind durch das Fehlerstellenpolynom $\sigma(x)$ bestimmt. Die Aufgabe der Chien-Search ist, die inverse DFT auf dieses Polynom anzuwenden. Dies geschieht durch

$$\sigma(\alpha^l) = \sum_{i=0}^t \sigma_i \alpha^{il}.$$

In einer Hardwareimplementierung kann man über die Fehlerpositionen l iterieren, wobei je Iteration jede Stelle i des Fehlerstellenpolynoms $\sigma(\alpha^i)$ berücksichtigt wird. Ist das Ergebnis Null, so ist an dieser Stelle ein Fehler aufgetreten und das entsprechende Informationsbit wird invertiert. Da sich der Grad des Fehlerstellenpolynoms mit t verändert, muss der Term

$$\sum_{i=0}^t \sigma_i \alpha^{il}$$

an den Code angepasst werden. Eine Multiplexerschaltung wählt die jeweilige Fehlerkorrekturfähigkeit t .

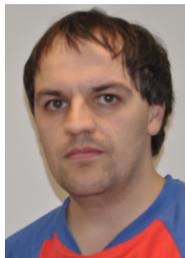
VI. FAZIT

Das komplette Design eines BCH-Encoders und Decoders wurde in Verilog HDL implementiert und auf einem Xilinx Virtex4 (xc4vlx200) FPGA verifiziert. Für diesen FPGA benötigt ein 24-bit korrigierender paralleler BMA 11069 Lookup Tables (LUT), während der serielle 48-bit korrigierende BMA 3462 LUT belegt. Insgesamt werden für eine serielle/parallele Mischimplementierung 14531 LUT benötigt. Eine rein parallele Implementierung für einen 48-bit korrigierenden BMA benötigt 29655 LUT.

In diesem Artikel haben wir den Encodier- und Decodierablauf des BCH-Codes erläutert. Eine wichtige Anforderung dabei war, mehrere Codes in einer Hardwareimplementierung zu realisieren. Diese Flexibilität wurde durch eine gemischte seriell/parallel Implementierung des BMA erreicht. Diese Implementierung erlaubt es, ohne Reduktion des mittleren Durchsatzes den Flächenbedarf signifikant zu verringern.

LITERATURVERZEICHNIS

- [1] R. Micheloni, A. Marelli, R. Ravasio, „Error Correction Codes for on-Volatile Memories“, Springer, 2008.
- [2] E. Yaakobi, J. Ma, L. Grupp, P. Siegel, S. Swanson, and J. Wolf, „Error characterization and coding schemes for flash memories“, in IEEE GLOBECOM Workshops, Dec. 2010, pp. 1856-1860.
- [3] R. E. Blahut, „Algebraic Codes for Data Transmission.“, Cambridge Univeristy Press, 2003.
- [4] W. Liu, J. Rho, and W. Sung, „Low-Power High-Throughput BCH Error Correction VLSI Design for Multi-Level-Cell NAND Flash Memory“, in Signal Processing Systems Design and Implementation, 2006. SIPS '06. IEEE Workshop on, oct. 2006, pp. 303-308.
- [5] R. E. Blahut, „Theory and Practice of Error Control Codes.“, Addison-Wesley, 1983.
- [6] M. Bagatin, „Effects of Ionizing Radiation in Flash Memories“, University of Padova, 2010.
- [7] J. Spinner, J. Freudenberger, „Generierung von Codekomponenten für BCH-Encodierer“, MPC Workshop, 2012.



Jens Spinner erhielt den akademischen Grad des BSc in Technischer Informatik im Jahr 2009 sowie den MSc in Informatik im Jahr 2011 von der HTWG-Konstanz. Er ist wissenschaftlicher Mitarbeiter der Hochschule Konstanz.



Dr. Jürgen Freudenberger ist seit 2006 Professor für eingebettete Systeme an der Hochschule Konstanz. Dort leitet er das Institut für Systemdynamik. Seine Forschungsarbeit beschäftigt sich vorrangig mit der Entwicklung von Algorithmen im Bereich der Signalverarbeitung und der Codierung für zuverlässige Datenübertragung sowie mit der effizienten Umsetzung der Verfahren in Hard- und Software.

Measurement Unit for Energy Efficient M2M Mobile Communication

Martin Klemm, Axel Sikora

Abstract—Machine-to-machine communication is continuously extending to new application fields. Especially smart metering has the potential to become the first really large-scale M2M application. Although in the future distributed meter devices will be mainly connected via dedicated primary communication protocols, like ZigBee, Wireless M-Bus or alike, a major percentage of all meters will be connected via point to point communication using GPRS or UMTS platforms. Thus, such meter devices have to be extremely cost and energy efficient, especially if the devices are battery based and powered several years by a single battery. This paper presents the development of an automated measurement unit for power and time, thus energy characteristics can be recorded. The measurement unit includes a hardware platform for the device under test (DUT) and a database based software environment for a smooth execution and analysis of the measurements.

Index Terms—Machine to machine communication, energy optimization, measurement unit, energy consumption of GSM modules.

I. INTRODUCTION

The number of M2M-applications is rapidly increasing. Whereas development and installation of standard mobile communication systems is a mere development task, the implementation of battery-powered sensor systems comes with severe challenges, especially if long operation times in the range of years should be achieved. For these systems, optimization not only from hardware, but also from firmware side is required. Mobile communication protocol usage must also be optimized.

For this analysis, an automated measurement unit was developed, which is presented in this contribution. In chapter II, the general challenges for the power and energy measurement of communication equipment are described. Ch. III describes some basics about current

measurement and the hybrid layer capacitor. These challenges lead to the use of a flexible, off-the-shelf measurement device, and to the development of an own portable and optimized measurement unit. Both are described in chapter IV. An important part of the automation process comes from the software environment, which is presented in ch. V. Finally, chapter VI shows results of the representative measurements.

II. CHALLENGES

For power consumption measurement of a GSM/GPRS module the main challenge is the high dynamic range of the current that the module consumes. During communication, a module may consume peak currents of up to 2 A during a TDMA time slot of 577 μ s every 4.615 ms. During the remaining (passive) time slots, the power consumption is typically in the range of 10 mA. During power down modes, the power consumption goes down to around 50 μ A. There exist also certain sleep modes where power consumption still is around 1...2 mA. This high dynamic range needs to be taken into account in terms of current as well as timing resolution. Thus, the following requirements for the measurement unit can be derived:

- Current range of 0...2 A
- Current resolution of 100 μ A for a lower range
- Current resolution of 1 mA for a higher range
- Measurement sample rate of up to 20 kS/s to have approximately 10 samples of current peaks

III. BASICS

A. Current measurement

Typically a current is measured over a shunt resistor via the voltage drop across the resistor. The voltage is measured behind the current measurement. This is also the approach that was followed for the measurement unit. Fig. 1 shows the block diagram for such a typical high side current and voltage measurement.

On the one hand the shunt resistor must be sufficiently small so that the voltage drop at the maximum current is still within the tolerance for the communication module. On the other hand it must be sufficiently

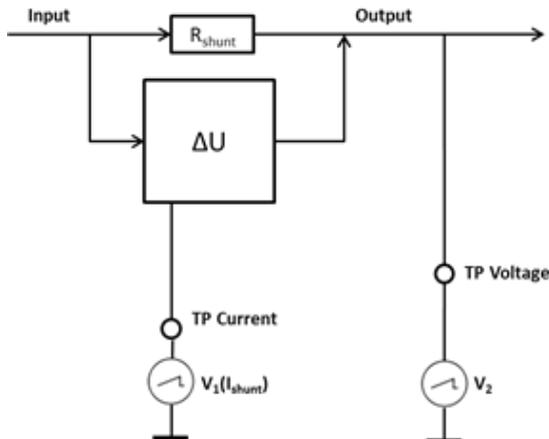


Figure 1: Typical current measurement setup.

large to achieve the required accuracy at small currents.

Typically, the voltage drop across the shunt is amplified by a high-side current sense amplifier and measured by an ADC as voltage level proportional to the current. To achieve higher precision of the power measurement, the voltage is measured after the shunt. Thus the voltage drop of the shunt resistor is taken into account.

B. Hybrid Layer Capacitor

Typically the DUT has its own power supply for the GSM/GPRS module. But since the GSM/GPRS module consumes a high current in such a short time this power supply is supported by a so called Hybrid Layer Capacitor (HLC). This capacitor prevents a voltage drop the power supply would generate at the rising of the peak current. This has to be taken into account in such way that the measurement of the GSM/GPRS module has to be between the HLC and the module itself and not between the power supply and the HLC or even for the whole DUT.

IV. MEASUREMENT UNIT

A. Off-the-shelf power analyzer

There are devices on the market that allow exactly the type of measurements required for this application. Such a device is the Source/Measurement Unit (SMU) provided as module of the DC Power Analyzer [1]. The seamless measurement ranging allows the measurement of low currents down to 100 nA as well as peak currents of up to 3 A, both with full resolution of 28 bits. Seamless ranging enables range changes without losing any reading and keeping the output voltage stable. Measurements can be displayed in the form of meter, scope waveform or data logged over extended periods of time.

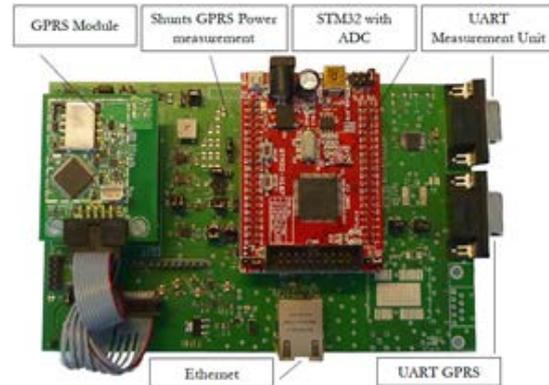


Figure 2: Portable Measurement Unit.

B. Portable measurement unit

Whereas power analyzers available on the market allow very precise measurements, a Portable Measurement Unit (PMU) was developed and produced for distributed and mobile measurements. It is shown in Fig. 2.

It is controlled by an STM32-microcontroller [2] and hosts the DUT. The two ADCs measure with a 12 bit resolution in a range of 0 to 3.3 V. As input for the ADCs 16 different channels can be assigned. The maximum sampling frequency is restricted to 2 MS/s. These two ADCs are used simultaneously to measure voltage and current values at a time [3]. Since multiplexing of the channels costs time the maximum sampling frequency decreases. Another bottleneck is the communication via the serial port. These two facts cause a reduction of the sampling frequency from 2 MS/s down to 3 kS/s. Theoretically it works with a much higher sampling frequency but tests show that 3 kS/s is the highest value that runs stable. Taken a peak of around 500 μ s just one or two samples describe this peak. But taken into account the simple structure during the TDMA timeslots shown in fig. 3 this data can be interpolated by the software environment described in the next chapter. The red graph in fig. 3 shows the used model for the interpolation. This model was chosen after measurements with the DC Power Analyzer described in the previous section. In this model noise was neglected.

To handle at the same time the high dynamic range of the current and the precise resolution in the lower current range, two current measurements are provided, one for the range of 0...200 mA and one for 0...2 A. By this a resolution of 100 μ A in the lower range can be achieved. At the same time the peaks can be measured with a 1 mA resolution. These currents are sampled directly one after another.

However, this is not implemented yet but taken into account that during the whole measurement the GSM/GPRS module is either in an active mode (> 10 mA) or in a passive mode (~ 10 mA) the higher

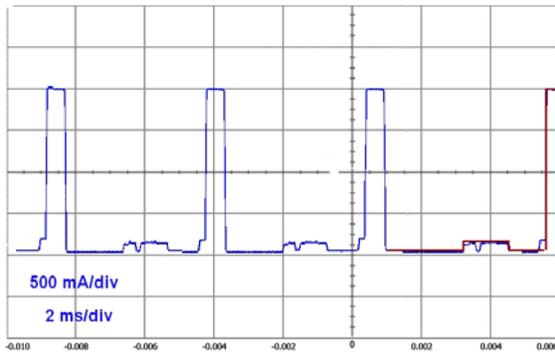


Figure 3: Current drain measurement of a GSM/GPRS module [1] during a TDMA timeslot (blue curve) and its interpolant (red curve).

range suffices. In this higher range it would not be able to measure currents during sleep or power down modes with currents around or even smaller than 1 mA but they do not occur during the measurement.

As described in chapter III.A, the current through the shunts is amplified and then sampled by the ADC. High precision high-side current-sense amplifiers by Maxim [4] are used due to their advantages comprising high precision and temperature stability and their availability in different gains. Different gains are necessary since different shunts are used.

For accuracy reasons, the described measurement ranges can be adjusted by hardware modification, i.e. exchanging the shunt resistors. The power consumption of the portable measurement unit is in the range of 200 mA allowing extended battery operation.

For communication between the STM32 and a PC the measurement unit is equipped with a serial port and for higher data throughput with an Ethernet port. The communication between the DUT and the PC is realized via a second serial port.

V. SOFTWARE ENVIRONMENT

As the measurement results depend on the environment and show a statistical distribution, it is necessary to perform a larger number of consecutive measurement runs. For efficiency reasons, an automated test environment was designed and implemented.

The Software Environment (SE) controls the GSM/GPRS module (DUT) and the power measurement unit as shown in Fig. 4. To perform one measurement iteration the SE sends the command to start the measurement to the PMU. Then the PMU starts the DUT by setting the ON/OFF pin of the GSM/GPRS module. When the SE receives the message from the GSM/GPRS module that it is turned ON it sends the AT commands for initiating the network connection and for transmission of the data. If the transmission was successful the GSM/GPRS module disconnects from the network, the PMU sets the

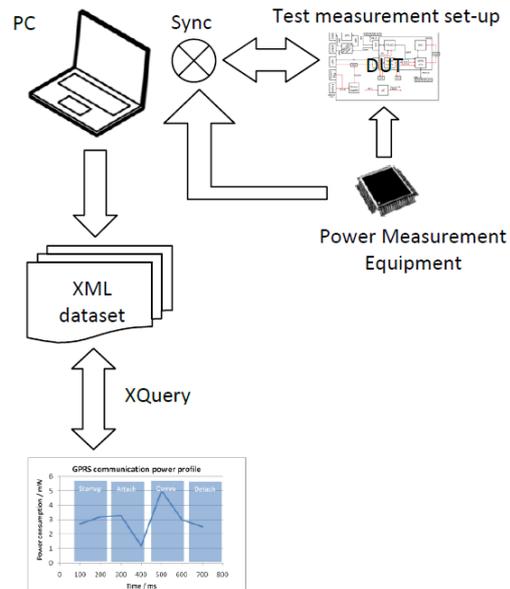


Figure 4: Software Environment.

ON/OFF pin in such way that the module shuts down and it stops the measurement.

In the next step, the data is stored in a Berkeley XML database by Oracle. This allows powerful queries across different measurement runs. The statistical evaluation of the data includes comparisons between

- modules from different manufacturers,
- different locations,
- different GSM providers and
- GSM module antenna configurations.

Finally, reports provide evidence of the energy consumption and the timing under these different conditions.

VI. RESULTS

A. Time and Energy vs. Phases

The resolution in time gives a very informative view on the different phases of the communication. Fig. 5 shows the distribution of time in percentage of the mean of 700 measurements in different locations with two different GSM modules and two different providers. For each configuration the same amount of data is being collected.

For the transmission of 4 kB of data the longest part falls to the attachment process to the network and to the connection establishment. It is possible that the time from starting up the modem until the start of the TCP communication takes 64 % of the total.

In contrast, the energy consumption is slightly different. Data transmission requires the most significant amount of energy whereas the phases from start-up until opening the TCP connection requires only 50 % of the overall energy consumed. Based on these fig-

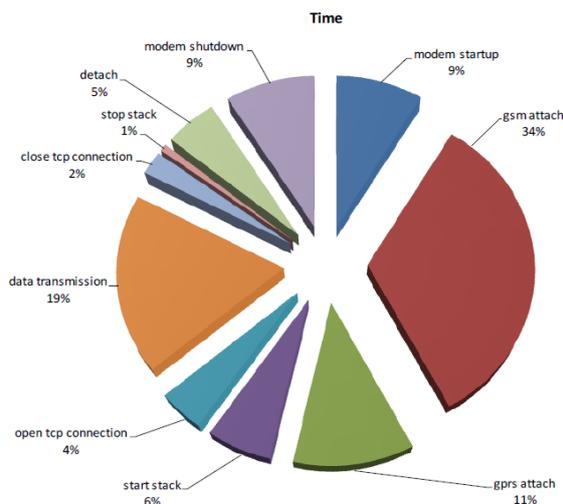


Figure 5: Average distribution of time per phase in percentage of 700 measurements.

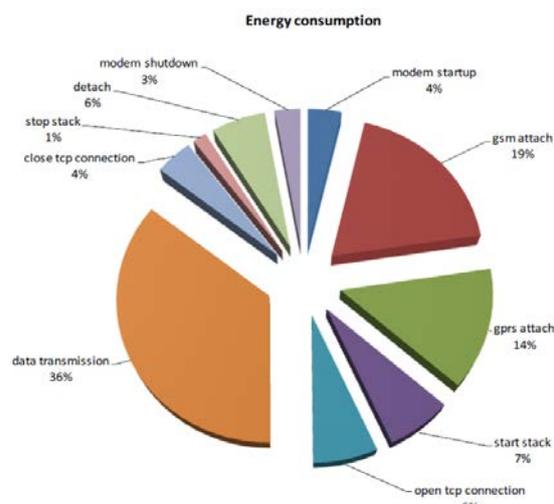


Figure 6: Average distribution of energy consumption per phase in percentage of 700 measurements.

ures the data must be analyzed further in terms of GSM modules, providers and locations.

B. Time and Energy for different GSM modules and providers

For the analysis and comparison of the energy consumption depending on different modules two different modules MOD_A and MOD_B of different manufacturers were used. Both were tested with the same external antenna. Additionally, tests with different providers (T-Mobile and Vodafone) in their home networks and in international networks using roaming were performed.

These tests are based on 657 measurements at three different locations and they are shown in fig. 7 and fig. 8. Each measurement is represented as bar in the diagram. The measurements are grouped by module and provider.

Fig. 7 shows the timing diagram. A critical situation can be observed in the phase "06 data transmission". The combination of module A with the provider T-Mobile shows a significant variance in the time used. Certain test runs use even more than 40 s for the data transmission. However, the same module with a different provider shows a much lower data transmission rate.

Energy measurements shown in fig. 8 are based on the energy consumption of the GSM module. It is calculated as work in mWh per phase. It is based on

the same data as used in the previous chapter but the data on the y-axis is the work consumed by the GSM module.

This illustration shows clearly that a main portion of the energy consumption lies in the data communication. In addition, the attachment process still consumes a significant amount of energy. It is slightly higher for module B compared to module A. In the main parts the energy consumption reflects the time requirements. In [5] different optimization approaches based on this data are shown.

VII. CONCLUSION

General challenges with high dynamic range current measurements on GSM/GPRS modules are discussed and analyzed. By implementing two current measurements, one for the precise resolution in the lower current range and one for the higher current range with less precision, these challenges are overcome.

Instead of having a sampling rate in the range of MS/s, approximately 3 kS/s are sufficient for interpolating the actual energy consumption. With this technique it is possible to show the energy consumed by the modules in each communication phase. Measurements with the DC Power Analyzer described in chapter IV.A show that the inaccuracies of the minor time resolution and the higher range are still in a reasonable dimension.

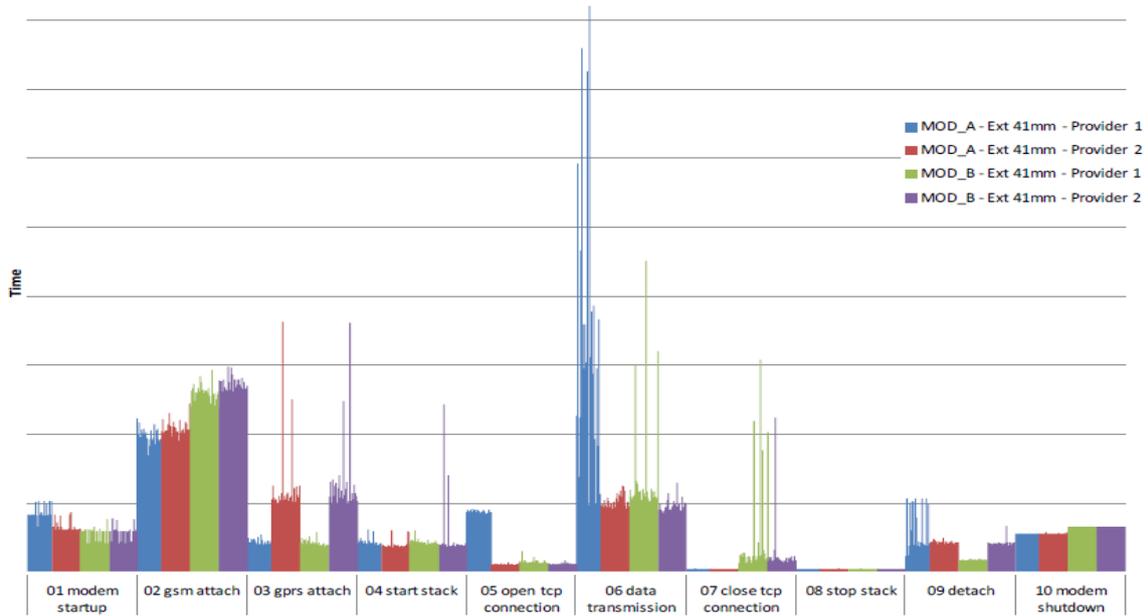


Figure 7: Timing Diagram with two different GSM modules and two different providers (Provider 1 = T-Mobile, Provider 2 = Vodafone).

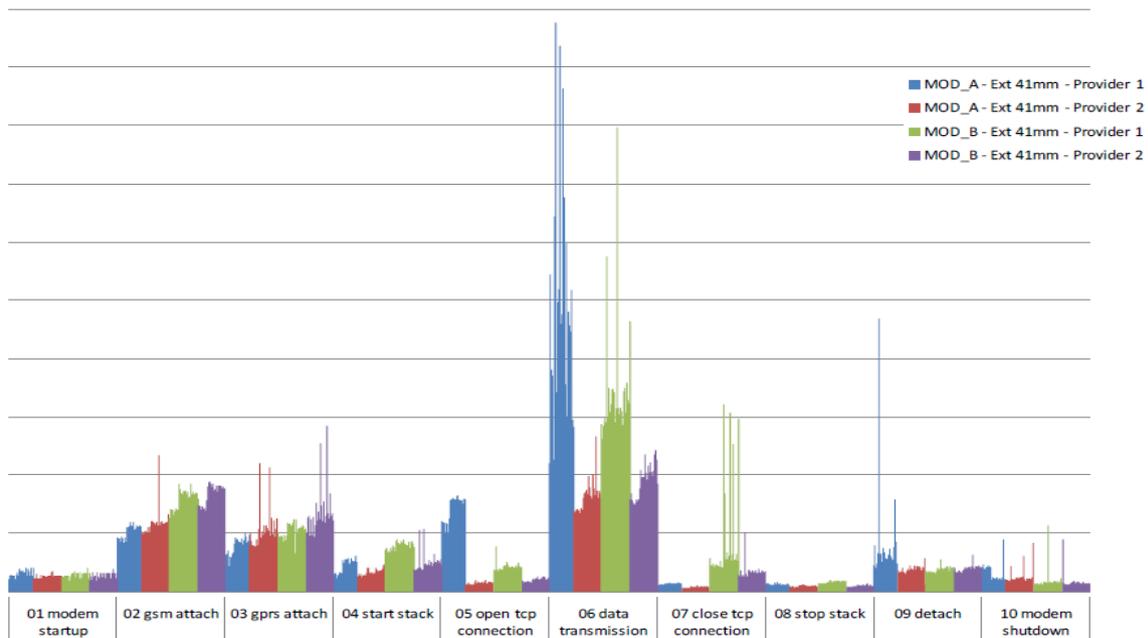


Figure 8: Diagram showing the energy consumption with two different GSM modules and two different providers (Provider 1 = T-Mobile, Provider 2 = Vodafone).

With the described PC software environment it is possible to save all measured data in a database to run powerful queries across different measurements. Finally, reports give evidence that the startup phases (modem startup until TCP connection established) takes 64 % of the time but just 50 % of the energy taking into account that 4 kB of data are transmitted. The transmission of this data takes 19 % of the time but 36 % of the energy. Furthermore two different modules with two providers in numerous locations were checked. Depending on the phase and the mod-

ules there are severe differences in the time and energy consumption that have to be taken into account when dimensioning batteries for M2M applications.

The next steps are implementing the second range in software and the Ethernet connection on the PMU to obtain a higher sampling rate and a higher current resolution.

REFERENCES

- [1] Agilent, „DC Power Analyzer”, www.agilent.com/find/N6781A-EU.
- [2] STMicroelectronics, „STM32F107xx datasheet”, August 2011.
- [3] STMicroelectronics, „AM3116, STM32's ADC modes and their applications”, March 2010.
- [4] Maxim, „Ultra-Precision, High-Side Current-Sense Amplifiers, MAX9922/MAX9923”, January 2010.
- [5] A. Sikora, N. Braun, „Energy Optimization for GPRS-based M2M-communication”, Embedded World Conference, Nürnberg, 2012.



Martin Klemm holds a Bachelor of Engineering in Electrical Engineering from Baden-Wuerttemberg Cooperative State University Loerrach with internship at Testo AG in Lenzkirch. He now pursues at the University of Applied Science in Offenburg a Master program in Electrical and Information Technology Engineering and will finish it in 2013.



Axel Sikora holds a diploma of Electrical Engineering and a diploma of Business Administration, both from Aachen Technical University. He has done a Ph.D. in Electrical Engineering at the Fraunhofer Institute of Microelectronics Circuits and Systems, Duisburg, with a thesis on SOI-technologies. After various positions in the telecommunications and semiconductor industry, he became a professor at the Baden-Wuerttemberg Cooperative State University Loerrach in 1999. In 2011, he joined Offenburg University of Applied Sciences, where he holds the professorship of Embedded Systems and Communication Electronics. His major interest is in the field of efficient, energy-aware, autonomous, and value-added algorithms and protocols for wired and wireless embedded communication.

Dr. Sikora is founder and head of Steinbeis Innovation Center Embedded Design and Networking (sizedn). He is author, co-author, editor and co-editor of several textbooks and numerous papers in the field of embedded design and wireless and wired networking, and head and member of numerous steering and program committees of international scientific conferences.

Soft-Core-Prozessor mit Multithreading für DSP-Anwendungen

Christian Kielmann, Irenäus Schoppa

Zusammenfassung—Die Komplexität der heutigen FPGAs ermöglicht es, Soft-Core-Prozessoren mit applikationsspezifischen Merkmalen effizient zu implementieren. Eine der interessantesten Anwendungen sind spezielle Prozessoren für digitale Signalverarbeitung. Solche Anwendungen zeichnen sich oft durch einen sehr hohen Datendurchsatz aus, der mit fließbandorganisierten Prozessoren effizient verarbeitet werden kann. Die Fließbandorganisation eines Prozessors bringt allerdings wegen Daten- und Steuerflussabhängigkeiten einige Probleme mit sich, die mit steigender Anzahl der Fließbandstufen immer schwerwiegender werden. Eine andere, interessante Technik ist es, eine Applikation in einzelne Tasks (Threads, Befehlsströme) zu zerlegen, und diese überlappend ausführen zu lassen. Dadurch lassen sich Fließbandhemmnisse zwar eliminieren, allerdings auf Kosten eines höheren Hardwareaufwandes für die Verwaltung der einzelnen Befehlsströme. Der folgende Beitrag gibt einen Überblick über die Synthese eines skalierbaren Soft-Core-Prozessors mit Multithreading-Eigenschaften für DSP-Anwendungen auf einem Spartan-FPGA.

Schlüsselwörter—Multithreading, Spartan, Virtex, FPGA, skalierbarer Soft-Core-Prozessor, digitale Signalverarbeitung.

I. EINLEITUNG

Der in diesem Beitrag vorgestellte Soft-Core-Prozessor ist das Resultat einer Masterarbeit [1], die bei der Enclustra GmbH in Zürich, Schweiz, in Zusammenarbeit mit der HTWG Konstanz durchgeführt wurde. Das Ziel der Masterarbeit war es, nach Vorgaben von Enclustra GmbH [2] einen skalierbaren Soft-Core-Prozessor mit Multithreading-Merkmalen für rechenintensive DSP-Algorithmen wie FIR-Filter, DFT/FFT und Matrix-Multiplikationen unter möglichst optimaler Ausnutzung der FPGA-Ressourcen zu realisieren. Der gesamte Vorgang sollte die Entwurfs-, Implementierungs-, Simulations-, Test- und Synthesephasen umfassen.

Ch. Kielmann, kielmann@htwg-konstanz.de, ist Master-Student an der HTWG-Konstanz, I. Schoppa, ischoppa@htwg-konstanz.de, ist Mitglied der HTWG Konstanz, Brauneckerstr. 55, 78462 Konstanz.

Im Kern ist der Soft-Core-Prozessor ein 32-Bit-Prozessor in einer 3-Adress-Architektur mit einem für RISC-Prozessoren typischen reduzierten Befehlssatz, der um DSP-spezifische Befehle erweitert ist, mit mehrstufiger Pipeline im Daten- und Steuerpfad, sowie mit Multithreading-Eigenschaften. Die skalierbare Architektur des Soft-Core-Prozessors sollte so realisiert werden, dass einige Einheiten wie die ALU oder die MAC-Einheit leicht austauschbar sind, um so dem Anwender eine einfache Anpassbarkeit des Soft-Core-Prozessors an spezialisierte Anwendungen zu ermöglichen.

Der Soft-Core-Prozessor unterstützt drei verschiedene Adressierungsmodi Linear, Modulo oder Bit-Reverse, wobei die beiden letzten besonders häufig in DSP-Algorithmen vorkommen. Aus Effizienzgründen erfolgt die Generierung der Adressen für Operanden in drei unabhängig voneinander arbeitenden AGUs. Das Datenformat des Prozessors ist auf die Verarbeitung vorzeichenbehalteter, ganzzahliger 32-bit-Operanden ausgelegt. Zu weiteren wichtigen Merkmalen des Soft-Core-Prozessors gehören die direkt in Hardware realisierte Unterstützung bei der Ausführung von Schleifen (Zero-Overhead-Loops), effiziente Realisierung bedingter und unbedingter Steuerflussbefehle und Unterprogrammaufrufe sowie laufzeitoptimierte Interrupts-Behandlung. Der Soft-Core-Prozessor wurde so entworfen, dass seine Performance sowohl durch die Anzahl der ausführbaren Threads als auch über die Merkmale der ALU skalierbar ist. Die Implementierung des Prozessors erfolgte nach Vorgaben und Richtlinien von Enclustra GmbH in der Hardwarebeschreibungssprache VHDL, und zwar auf der Grundlage der Zwei-Prozess-Methode [3]. Die Struktur des Soft-Core-Prozessors wurde bei der Implementierung auf sieben Entities aufgeteilt: Top-level, Program Flow Unit (PFU), Instruction Decode Unit (IDU), Address Generation Unit (AGU), Stack Control Unit (SCU), ALU und Quad-Port-Speicher. Die gesamte Implementierung umfasst ca. 3000 Lines of Code.

II. QUAD-PORT-SPEICHER

Die Speicherarchitektur ist eines der wichtigsten Merkmale eines Prozessors und hat einen hohen Einfluss auf seine Leistungsfähigkeit. Hinsichtlich der Lage der Operanden wird in der Literatur (z.B. [4], [5]) zwischen drei Architekturvarianten unterschieden:

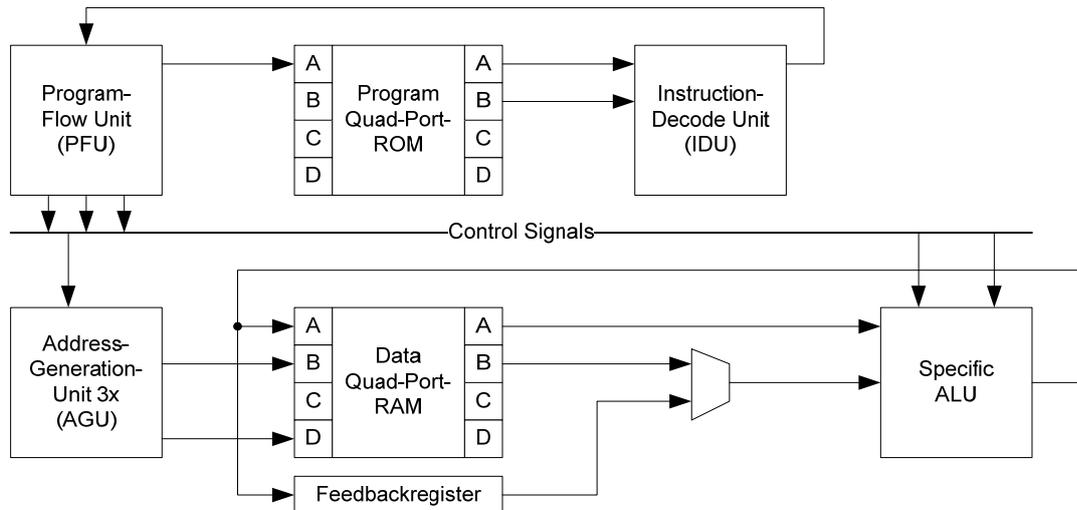


Abbildung 1: Eine auf zwei Quad-Port-Speicher für Daten und Programme basierende Architektur des Soft-Core-Prozessors.

Register-Register-, Register-Speicher- sowie Speicher-Speicher-Architekturen. Bei der für RISC-Prozessoren typischen Register-Register-Architektur werden Operanden üblicherweise mittels Load- und Store-Befehlen aus dem Speicher in den Registersatz transportiert, bevor sie verarbeitet werden können. Ein Nachteil dieses Ansatzes ergibt sich dabei aus der hohen Anzahl von Instruktionen bei Operationen mit Speichervariablen. Dieser Nachteil kann durch eine große Anzahl von Registern kompensiert werden. Bei der Register-Speicher-Architektur, die häufig bei CISC-Prozessoren zu finden ist, kann einer der Operanden direkt im Speicher liegen. Das Resultat einer ALU-Operation kann entweder im Registersatz oder direkt im Speicher abgelegt werden.

Die effiziente Realisierung einer Speicherarchitektur hängt im Wesentlichen von zwei Einflussgrößen ab. Zum einen spielt das Zugriffsverhalten der laufenden Anwendung auf den Speicher eine große Rolle. Zum anderen hat auch die Prozessorarchitektur hier in Form von Multithreading einen entscheidenden Einfluss auf den Entwurf. Die Befehlsverarbeitung benötigt für zwei Operanden auch zwei Lesezugriffe, sowie einen Schreibzugriff für das berechnete Ergebnis. Zum Beispiel setzt die Implementierung einer Prozessorarchitektur mit Multithreading voraus, dass in jeden Takt simultan zwei lesende und ein schreibender Zugriff möglich sind. Mit einem Block RAM in der Dual-Port-Konfiguration kann jedoch nur simultan entweder zweifach gelesen oder geschrieben werden [6]. Die Umsetzung dieser Anforderung kann mittels eines Quad-Port-Speichers erreicht werden. Mit einer solchen Komponente lassen sich simultan zwei Lese- und zwei Schreibzugriffe auf einen Block RAM durchführen.

Jeder der vier Ports kann unabhängig voneinander zum Lesen oder Schreiben genutzt werden. Der Quad-Port-Speicher wird mit zwei Clock-Domänen realisiert. Dabei teilen sich die Ports A und C eine Clock-

Domäne und die Ports B und D die andere. Die beiden Clock-Domänen dienen als Toggle für die Eingangsdaten und Steuersignale des Speichers. Da ein Block RAM als synchroner Speicher ausgelegt ist und ein neues Datum nur bei einer steigenden Taktflanke übernommen wird, muss der Quad-Port-Speicher mit der doppelten Taktrate betrieben werden.

Die Speicherarchitektur des Soft-Core-Prozessors basiert, wie in Abbildung 1 dargestellt, auf zwei Quad-Port-Speichern. Wie bereits erwähnt, muss der Datenspeicher in einer Quad-Port-Konfiguration ausgelegt werden, um eine Multithreading-Prozessorarchitektur zu ermöglichen. Für den Programmspeicher dagegen ist eine Quad-Port-Konfiguration nicht zwingend notwendig. Der Programmspeicher wurde als Quad-Port konfiguriert, um in einer weiteren Ausbaustufe das simultane Auslesen von zwei Instruktionswörtern in einem Takt zu ermöglichen. Das Feedback-Register repräsentiert das letzte ALU-Resultat.

III. MULTITHREADING

Das Multithreading des Soft-Core-Prozessors basiert auf dem Cycle-by-Cycle-Interleaving-Ansatz. Mit dieser Technik wird in jedem Prozessortakt ein Befehl aus einem anderen Befehlsstrom (Thread) in die Befehl-Pipeline des Prozessors geladen. Der wesentliche Vorteil dieses Verfahrens ist die überlappende Ausführung der Befehle in der Pipeline, und zwar ohne besondere Maßnahmen, die aus der sonst für die Pipeline-Verarbeitung typischen Daten- oder Steuerflussabhängigkeiten resultieren würden. Ein Befehl aus demselben Thread kann erst dann in die Pipeline eingespeist werden, nachdem der vorherige Befehl die Prozessorpipeline bereits verlassen hat. Der so realisierte Kontextwechsel zwischen den Threads ist insbesondere durch das zeitgesteuerte Verhalten effektiv in seiner Ausführung realisierbar. Nachteilig in diesem Verfahren ist es, wenn nicht genügend Threads zur Verfü-

gung stehen. Die jeweiligen Ausführungseinheiten des Prozessors können dann nicht mehr vollständig ausgelastet werden und so kommt es zum Leerlauf, zu sog. „Blasen“ in der Pipeline. Für gewöhnlich lassen sich DSP-Algorithmen gut partitionieren und auf alle zur Verfügung stehenden Threads verteilen.

Neben dem Cycle-by-Cycle-Interleaving-Ansatz wurde in [1] auch die Block-Interleaving-Technik hinsichtlich einer möglichen Umsetzung genau untersucht. Bei dieser Technik werden Befehle eines Threads solange sequentiell ausgeführt, bis eine Daten- oder Steuerflussabhängigkeit festgestellt wird, die dann zu Wartezyklen in der Verarbeitung führt. Dieser Ansatz wurde jedoch bei der Implementierung verworfen, da Wartezeiten erst spät in der Prozessorphipeline detektiert werden können und dadurch ein hoher Wechselaufwand notwendig ist.

In Abbildung 2 ist die PFU schematisch dargestellt, die aus einem Schiebe- und einem Pipelineregister besteht und in dieser Konfiguration sechs Threads verwalten kann. Die Register wurden mit Hilfe eines Records definiert und beinhalten zur Laufzeit den gesamten Threadkontext. Wird ein Thread für die nächste Instruktionverarbeitung aktiviert, so wird im ersten Schritt der Programmzähler an den Programmspeicher weitergeleitet. Da die dekodierte Instruktion zu diesem Zeitpunkt noch nicht vorliegt und somit die Instruktion nicht weiterverarbeitet werden kann, wird der aktive Threadkontext in das Pipelineregister überführt. Verlässt der Threadkontext die Pipeline, liegt zeitgleich die dekodierte Instruktion von der IDU an. Auf Basis der dekodierten Instruktion wird der nächste Zustand des Threads bestimmt. Das Schieberegister in der PFU erfüllt zwei Funktionen. Der Soft-Core-Prozessor besteht - über das gesamte System betrachtet - aus mehreren Pipelinestufen, die über das Schieberegister in der PFU ausgeglichen werden. Aus diesem Sachverhalt resultiert auch eine Mindestanzahl an Threads, die der Soft-Core-Prozessor verwalten muss. Außerdem kann mit dem Schieberegister eine flexible Anpassung der Anzahl an Threads für einen bestimmten Anwendungsfall vorgenommen werden.

IV. LOOP CONTROL UNIT

Die Loop-Control-Unit (LCU) ist eine Komponente innerhalb des Steuerpfades und dient zur Unterstützung der iterativen Berechnung von Adressen innerhalb einer Schleife ohne Verlust von Rechenzeit für den Schleifen-Overhead. Die LCU ist in zwei Teilbereiche unterteilt: in einen Instruktionszähler und in einen Schleifenzähler. Der Schleifenzähler wird über ein Loop-Count-Register (LCR) zur Laufzeit konfiguriert. Dieses Register gibt an, wie häufig ein Programmabschnitt abgearbeitet werden soll. Der Instruktionszähler verfügt über zwei Steuerregister, die Loop Start Address (LSA) sowie die Loop End Address (LEA), welche die Start- und Endadressen des Programmabschnitts enthalten. Mit Hilfe des speziellen

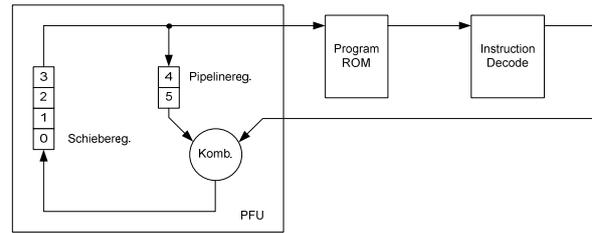


Abbildung 2: Schiebe- und Pipelineregister für Multithreading.

LOOP-Befehls werden alle drei Register aktualisiert. Während der Ausführung zählt die LCU die einzelnen Schleifendurchläufe anhand der Informationen in den Registern, prüft, ob das Ende der Schleife erreicht wurde und signalisiert diesen Zustand gegebenenfalls mit einem Loop-Flag im Statusregister.

V. BIT-REVERSE- UND MODULO-EINHEITEN

Die Bit-Reverse- und Modulo-Einheiten sind Bestandteil der AGU und werden über ein Konfigurationsregister, das CFG-Register, selektiert. Das Bit-Reversing ist für die Berechnung der Fast-Fourier-Transformation (FFT) und ähnlicher Algorithmen vorgesehen. Die Adressmodifikation erfolgt in Hardware und wird einzig in der Adressierungsart Postinkrement um 1 betrieben. Der Inhalt eines Registers wird dabei um den Wert 1 inkrementiert und anschließend in einem Reverse-Addierer zu einem Reverse-Vektor addiert. Der Reverse-Vektor gibt dabei den Adressbereich in der Zweier-Potenz an. Die Reverse-Carry-Arithmetik des Addierers berechnet das Carry-Bit in der zur normalen entgegengesetzten Richtung.

Für die Modulo-Adressierung stehen dem Soft-Core-Prozessor das M0-, M1- und Dn-Register zur Verfügung. Die M0- und M1-Register definieren den Adressbereich durch die Start- und Endadresse. Mit dem Dn-Register wird die Schrittweite der Modulo-Operation angegeben. Die Schrittweite sowie der Adressbereich können beliebig innerhalb des Speichers festgelegt werden. Ferner kann eine freie Einstiegsadresse innerhalb des definierten Adressbereiches gewählt werden. Weitere Realisierungsansätze sind in [7] zu finden.

VI. ASSEMBLER

Um einen Prozessor sinnvoll einsetzen zu können, braucht der Entwickler u.a. eine Programmierung, die im einfachsten Fall aus einem elementaren Assembler bestehen kann. Da aus Zeitgründen nicht möglich war, im Rahmen dieser Masterarbeit noch einen Assembler zu implementieren, wurde hier ein Werkzeug der Firma PohlTronik eingesetzt. Es handelt sich dabei um einen kostengünstigen, mittels Tabellen konfigurierbaren Makroassembler namens HASM. Mit diesem Werkzeug ist es möglich, mit wenig Zeitaufwand und flexibel einen passenden As-

```
.NOP      {255:8 0:5 0:5 0:4 0:10 0:16 0:16 }
.BRK      {00000000b:8 0:5 0:5 0:4 0:10 0:16 0:16 }
.JMP à    {2:8 0:5 0:5 0:4 0:10 Cd:16 @1:16}
.BRA Cà,à {2:8 0:5 0:5 0:4 0:10 @1:16 @2:16}
.CALL Cà,à {3:8 0:5 0:5 0:4 0:10 @1:16 @2:16}
```

Abbildung 3: Auszug aus der HASM-Def. mit Befehlsformaten.

```
Init      NOP
          BRK
          MISC MAGUIAn, 0020h
          LOOP L_End, #3
          NOP
L_End     BRK
          NOP
```

Abbildung 4: HASM Assemblerprogramm.

sembler für beliebige Prozessoren, Mikrocontroller und Signalprozessoren generieren zu können.

Die Generierung des Assemblers für den Soft-Core-Prozessor erfolgt dabei über eine Tabelle, in der die Befehle des Prozessors auf Bit-Ebene definiert sind. Außerdem können in der Tabelle physikalische und virtuelle Speicher-Segmente, Assemblerdirektiven und weitere Anweisungen definiert werden [8]. Der HASM Assembler basiert auf einer Konfigurationstabelle, mit der die Architektur des Soft-Core-Prozessors abgebildet wird. Die Konfigurationstabelle ist zeilenorientiert und wird durch das erste Zeichen in der Zeile festgelegt. In Abbildung 3 ist ein Ausschnitt aus der Konfigurationstabelle zu sehen. Eine Prozessorinstruktion wird über einen Punkt als Präfix eingeleitet und für den Assembler in Form von mnemonischen Abkürzungen definiert. Der Bezeichnung der Prozessorinstruktion folgt eine Sequenz von Argumenten, die das Instruktionsformat repräsentieren.

Die eigentlichen Umsetzungsanweisungen, zusammengesetzt aus den Anweisungen, müssen in geschweiften Klammern gesetzt werden. Alle Anweisungen haben dabei immer die Form <expr>:<bits>. Getrennt durch einen Doppelpunkt steht hierbei auf der linken Seite das Argument und auf der rechten Seite die Anzahl der Bits, die für dieses Argument verwendet werden sollen. Ein variables Argument wird mit ‚@‘ gekennzeichnet und bei der Assemblierung durch einen Wert ersetzt.

In Abbildung 4 ist ein Ausschnitt aus einem Assemblerbeispielprogramm zu sehen. Das Programm beginnt mit einem Init-Label, gefolgt von einer NOP Instruktion. Namen, die in der ersten Spalte des Assemblerprogramms stehen, werden als Labels interpretiert. Dabei ist darauf zu achten, dass zwischen den Labels und den Instruktionen mindestens ein Leerzeichen enthalten ist. Enthält eine Zeile nur Code, muss diese mit einem Leerzeichen beginnen.

Tabelle 1: Auszug aus der Report-Datei mit dem Ressourcenverbrauch des Soft-Core-Prozessors nach der Synthese für die Spartan6-Familie.

Ressourcen	Utilization	Available	Utilization
Register	3021	54576	5 %
LUT	3793	27288	13 %
Slice	1235	6822	18 %
IO	34	218	15 %
RAMB	11	116	9 %
BUFG	2	16	12 %

VII. VERIFIKATION

Die Funktionalität des Designs wurde mittels einer Simulation überprüft. Für die Simulation des Soft-Core-Prozessors wurde eine Testbench implementiert, die alle notwendigen Signalwechsel (Stimuli) an den Eingängen der zu testenden Einheit (DUT - Device Under Test) bereitstellt. Die Ausgangssignale vom DUT wurden anhand einer grafischen Darstellung des zeitlichen Verlaufs überprüft. Mit dieser Testbench lassen sich typische Fehler bei der Abarbeitung von Instruktionen im Prozessor erkennen. Diese Testbench basiert auf einem Testprogramm, welches mit dem HASM-Assembler generiert wurde und verschiedene Testfälle abdeckt, um so die Eigenschaften des Soft-Core-Prozessors zu testen. Das Testprogramm beinhaltet im Wesentlichen einzelne Tests für arithmetisch-logische Operationen, bedingte und unbedingte Sprünge sowie Schleifenanweisungen für die korrekte Adressgenerierung und Interrupts. Des Weiteren wird durch die Testbench die funktionale Korrektheit gewährleistet, wenn der Soft-Core-Prozessor mit einer variablen Anzahl an Threads skaliert wird. Insbesondere stellt die Testbench sicher, dass der Threadkontext des jeweiligen Threads auch bei maximaler Auslastung des Prozessors konsistent bleibt. Weiterführende Informationen zur funktionalen Verifikation sind in [9], [10] zu finden.

VIII. SYNTHESEERGEBNISSE

Die Analyse wurde mit PlanAhead, einer leistungsfähigen Analyse- und Floorplanning Software durchgeführt. Tabelle 1 listet einige wesentliche Kennzahlen nach dem Place & Route (PAR) zur Ressourcenauslastung des DSP auf dem XC6SLX45-CSG324 aus der Spartan6 Familie auf. Bei diesen Werten handelt es sich um einen Mindestbedarf. Dieser Bedarf erhöht sich entsprechend, wenn die Applikation Threads über die Mindestanzahl hinaus benötigt. Der Soft-Core-Prozessor wurde auf einem Spartan6 der Firma Xilinx platziert. Nach dem Place und Route kann er mit einer maximalen Taktrate von 133,245 MHz betrieben werden.

Tabelle 2: Gegenüberstellung der Performance bei verschiedenen Prozessoren.

Prozessor/Implementierung	Taktfreq.	MIPS
Xilinx FPGA, Spartan6	133 MHz	266
Xilinx FPGA, Virtex7	202 MHz	404
ADSP-2196 [11]	160 MHz	160
TMS320C5x	50 MHz	50
TMS320C6x [12]	1200 MHz	9600

Darüber hinaus erfolgte eine Implementierung auf dem Virtex7 Baustein XC7V58-TFFG1157 mit einer maximalen Taktrate von 202,419 MHz. Für die theoretische Leistungsbewertung wurde das MIPS-Maß angewendet. Der MIPS-Wert errechnet sich aus der Taktfrequenz und den Clockcycles per Instructions (CPI). Ein Faktor N repräsentiert die Anzahl der Threads, die der DSP aktiv ausführen kann.

Der Soft-Core-Prozessor benötigt mindestens 8 Zyklen, um einen Befehl abzuarbeiten. Werden alle vorhandenen Threads aktiv ausgeführt, ergibt sich daraus ein MIPS-Wert analog zur Taktfrequenz. Da das Instruktionsformat des Soft-Core-Prozessors eine Verarbeitung von zwei Befehlen gleichzeitig ermöglicht, ergibt sich somit ein maximaler MIPS-Wert aus der doppelten Taktfrequenz, vgl. Tabelle 2. Zum Vergleich erreicht der DSP von Analog Devices aus der ADSP-21 Familie einen MIPS-Wert von 160 [11].

IX. AUSBLICK

Die flexible Architektur des Soft-Core-Prozessors ermöglicht eine partielle Optimierung einzelner Komponenten, die den Ressourcen-Verbrauch reduzieren oder die Performance erhöhen kann. Für den Fall, dass der Soft-Core-Prozessor mit einer hohen Anzahl an Threads betrieben wird, ist es sinnvoll, den Bedarf an Ressourcen weiter zu minimieren. Dies kann zum Beispiel durch Vereinfachungen in der Reset-Phase erreicht werden. In der aktuellen Version sind für die Reset-Phase mehrere Flip-Flops für den Kontext vorgesehen. Das Ziel wäre es, diese Flip-Flops durch ein Schieberegister, als primitive FPGA-Komponente, zu ersetzen. Abschließend ist es überlegenswert, eine Timingoptimierung durchzuführen, damit der Soft-Core-Prozessor mit einer höheren Taktrate betrieben werden kann.

LITERATURVERZEICHNIS

- [1] Ch. Kielmann: „Synthese eines Soft-Core-Prozessors mit Multithreading für DSP-Anwendungen auf einem FPGA“, *Master Thesis, HTWG Konstanz*, 2012.
- [2] Enclustra, „Firmeninterne Spezifikation“, *Enclustra GmbH, Zürich*, 2011.
- [3] J. Gaisler: „A structured VHDL design method“ in „Fault-tolerant Microprocessors for Space Applications“, *Aeorflex Gaisler AG, Goteborg*, <http://www.gaisler.com/doc/vhdl2proc.pdf>
- [4] U. Schneider, D. Werner: „Taschenbuch der Informatik“, *Fachbuchverlag Leipzig, Carl Hanser Verlag*, 2001.
- [5] J. Silic, B. Robic, T. Ungerer: „Processor Architecture“, *Springer Verlag*, 1999.
- [6] Xilinx: „Using Block RAM in Spartan-3 Generation FPGAs“, *Application Note XAPP463, Xilinx Inc.*, 2005.
- [7] L. Dake: „Embedded DSP Processor Design Application Specific Instructionset Computers“, *Morgen Kaufman*, 2008.
- [8] H. Hohe: „HASM HSIM Anleitung/Referenz Tabellenkonfigurierbarer Makro-Assembler mit integriertem Linker und zugehörigem Simulator/Debugger“, www.phltronik.de, 1998,
- [9] W. Bruce, J. Goss, W. Roesner: „Comprehensive Functional Verification, The complete industry cycle“, *Elsevier Morgen Kaufman Publishers*, 2005.
- [10] B. Bailey: „The Functional Verification of Electronic Systems, An Overview from Various Points of View“, *Professional Education International*, 2005.
- [11] Analog Devices: „ADSP-2196M“, *Data Sheet Rev PrA, Preliminary Technical Data DSP Microcomputer, Analog Devices*, 2001.
- [12] Texas Instruments: „TMS320C6455, Fixed Point Digital Signal Processor“, *Technical Data Sheet*, 2012.



Christian Kielmann studierte Technische Informatik an der HTWG Konstanz und erhielt dort im Jahre 2010 den akademischen Grad B.Sc. Im Rahmen einer Bachelor-Arbeit befasste er sich mit Soft-Core-Prozessoren. Neben dem Master-Studium arbeitete er in der Spezialmesstechnik beim Triebwerkshersteller Rolls-Royce Deutschland.



Irenäus Schoppa studierte Informatik an der Technischen Universität Berlin und erhielt dort im Jahre 1993 den akademischen Grad Dipl.-Informatiker. Im Jahre 1998 promovierte er dort zum Dr.-Ing.. Seit dem Jahr 2008 ist er Professor für Hardware-Software Codesign an der HTWG Konstanz.

TCL-Skript basiertes Backend Design mit Cadence Encounter

Stefan Thiel, Heinz-Peter Bürkle

Zusammenfassung—Das Backend Tool Encounter von Cadence ermöglicht die Umsetzung von Gatter-Netzlisten in ein fertiges Layoutdesign. Die für diesen Prozess benötigten Schritte lassen sich durch Skripte automatisieren. In diesem Beitrag wird beschrieben, wie ein einfaches Skript passend zum Designflow in Encounter erzeugt werden kann. Anhand eines Beispiels werden die verschiedenen Schritte veranschaulicht und die Funktionen eines Skriptes verdeutlicht.

Schlüsselwörter—Digital IC Design, Backend Designflow, Cadence Encounter, Tcl-Skripte.

I. EINLEITUNG

Der Entwurfsprozess einer integrierten Digitalschaltung beginnt auf einer abstrakten Ebene mit der Hardwarebeschreibung, häufig auf der Register-Transfer-Ebene (RTL Register Transfer Level). Entsprechende Hardwarebeschreibungssprachen wie VERILOG oder VHDL (VHSIC Hardware Description Language) stehen hierzu zur Verfügung. Eine erste Umsetzung dieser Hardwarebeschreibung in eine Netzliste kann mittels Synopsys Design Compiler vorgenommen werden. Die daraus generierte Netzliste beschreibt, wie die zur Verfügung stehenden Standardzellen einer bestimmten Technologiebibliothek untereinander elektrisch verbunden werden. Im nächsten Schritt sollen die in der Netzliste definierten Strukturen auf der Chipfläche angeordnet und verdrahtet werden. Dieser Entwurfsschritt erfolgt zum Beispiel durch das Backend-Tool Encounter. Der vorliegende Beitrag behandelt den Ablauf zum Layoutentwurf mit Encounter und wie dieser mit Hilfe von Skripten unterstützt und vereinfacht werden kann.

II. TCL-SKRIPTE FÜR CADENCE ENCOUNTER

Cadence Encounter bietet für das Backend Design viele Möglichkeiten zur Verifikation und Optimierung. Daher gibt es auch kein vollautomatisches Ver-

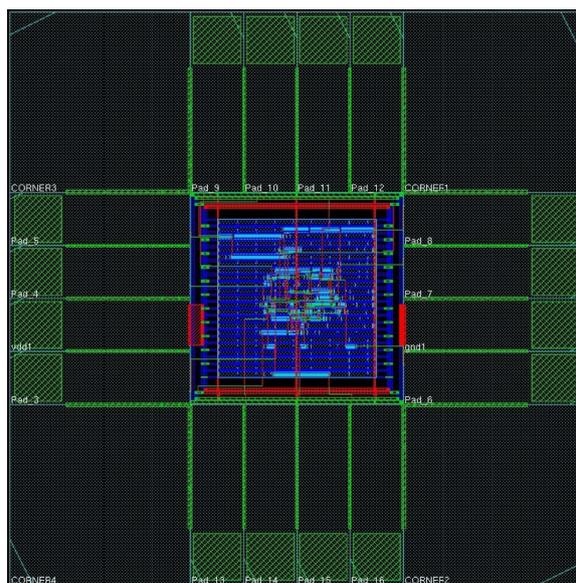


Abbildung 1: Mit Hilfe eines Skripts automatisch erzeugtes Beispiel-Design.

fahren, bei dem der Anwender des Tools (also der Chipdesigner) sozusagen auf Knopfdruck ein fertiges IC-Design bekommt. Trotzdem gibt es bestimmte Verfahren, Einstellungen und Abläufe, die zumindest für eine bestimmte Designumgebung stets gleich sind oder nur kleinere Variationen aufweisen. Dann sollte der Anwender nicht immer neu von vorne beginnen müssen, sondern sollte den bereits durchlaufenen Weg aufzeichnen und durch einfaches textuelles Editieren abändern können. Dazu hat Cadence die Möglichkeit der Nutzung von Skripten vorgesehen und hierzu die Sprache Tcl verwendet. Die Open Source-Skriptsprache Tcl (Tool Command Language) kann am besten durch zwei Eigenschaften, „radikal einfach“ und „alles ist Text“ beschrieben werden.

So ist es möglich, mit Hilfe eines automatischen Befehlsmittschnitts der Eingaben in die Encounter-Benutzeroberfläche ein erstes, einfaches Skript zu erhalten. Hierzu kann der nachfolgend beschriebene Designflow in der Benutzeroberfläche durchgeführt werden. In Abbildung 1 ist ein durch ein Skript automatisch erzeugtes Beispieldesign zu sehen.

```
VDD3ALLP vdd1 ();
GND3ALLP gnd1 ();

ICP Pad_3 (.Y(clk), .PAD());
ICP Pad_4 (.Y(en), .PAD());
ICP Pad_5 (.Y(d[0]), .PAD());
ICP Pad_6 (.Y(d[1]), .PAD());
ICP Pad_7 (.Y(d[2]), .PAD());
ICP Pad_8 (.Y(d[3]), .PAD());
ICP Pad_9 (.Y(load), .PAD());
ICP Pad_10 (.Y(up_down), .PAD());
ICP Pad_11 (.Y(reset), .PAD());

BU1P Pad_12 (.A(q[0]), .PAD());
BU1P Pad_13 (.A(q[1]), .PAD());
BU1P Pad_14 (.A(q[2]), .PAD());
BU1P Pad_15 (.A(q[3]), .PAD());
BU1P Pad_16 (.A(eno), .PAD());
```

Abbildung 2: Auszug der Signalverknüpfung zu IO-Pads in der Top Level Entity.

III. ENCOUNTER DESIGNFLOW

Zunächst werden die prinzipiellen Arbeitsschritte in Encounter gezeigt und erläutert. Zu Beginn eines jeden Designs sollten wichtige Eckpunkte geklärt werden. Hierzu gehören zum Beispiel die Technologiebibliothek, verwendete IP (Intellectual Property) sowie die Anordnung und Art der IO-Pads. Die IO-Pads der jeweiligen Technologiebibliothek müssen zunächst in der Top-Level Entity der Netzliste mit dem entsprechend zugehörigen Signal verknüpft werden. Ein Beispiel hierzu ist in Abbildung 2 zu sehen. In diesem Ausschnitt steht z.B. ICP für das genutzte IO-Pad der Technologiebibliothek. Jedem Pad wird ein eigener Namen zugewiesen und die Kontakte werden definiert. Im Beispiel wird mit dem als Pad_3 benannten Pad das Signal clk nach außen geführt. Die Signalverknüpfungen des Pads werden in der Klammer definiert. Zudem bietet es sich an, eine IO-Datei anzulegen, in welcher die Positionen der IO-Pads beschrieben werden. Ein Beispiel hierzu ist in Abbildung 3 gezeigt.

Jedem Chipdesign liegt in Encounter der in Abbildung 4 gezeigte Ablauf zugrunde. Daher können an dieser Stelle Skripte den weiteren Verlauf optimal unterstützen. So bietet ein Skript die Möglichkeit, die verschiedenen Designparameter und weiteren Variablen eines Designs in textueller Form einzugeben. Die zum Designflow gehörenden Schritte werden anschließend vom Skript mit den zuvor bestimmten Designparametern und Variablen ausgeführt.

Im ersten Schritt, dem Design Import, werden die Pfade zu Netzlisten, Technologiebibliotheken, IO-Dateien und Timing-Dateien angegeben. Die globalen Versorgungsnetze werden definiert. Bei der Verwendung von Skripten bietet es sich an, eine bei jedem

```
Pad: CORNER4 SW CORNERP
Pad: CORNER3 NW CORNERP
Pad: CORNER2 SE CORNERP
Pad: CORNER1 NE CORNERP
```

```
Pad: Pad_3 W
Pad: vdd1 W
Pad: Pad_4 W
Pad: Pad_5 W
```

```
Pad: Pad_6 E
Pad: gnd1 E
Pad: Pad_7 E
Pad: Pad_8 E
```

Abbildung 3: Auszug der IO-Datei zur Positionsdefinition der IO-Pads.

Start von Encounter erzeugte Standardkonfiguration zu bearbeiten. Es werden also die im Kopf des Skripts definierten Pfade und Variablen in diese Standardkonfiguration geschrieben. Diese kann gespeichert und beliebig oft geladen werden. Dadurch wird eine für jedes Design individuelle Konfigurationsdatei erzeugt. In Abbildung 5 wird gezeigt, wie im Skript dieser Schritt umgesetzt werden kann. Dort sind die verwendeten Befehle aufgeführt, mit welchen eine Standardkonfiguration an ein neues Design angepasst wird und als Konfigurationsdatei für dieses Design gespeichert wird. Mit dem letzten Befehl (commitConfig) wird das Design importiert. Zu beachten ist hierbei, dass es sich bei z.B. \$NETLIST_PATH um den im Skript angegebenen Pfad zur Netzliste handelt. Durch die Verwendung von im Skript global definierten Designparametern wird ein Skript wiederverwendbar.

Der nächste Schritt, das Floorplanning, definiert das Layout des Chip. Hierzu gehören zum Beispiel Seitenverhältnis, Flächennutzung sowie Abstand zwischen Kern- und IO-Bereich, in welchem die später erzeugten Versorgungsstrukturen platziert werden. Wenn IP im Design verwendet wird, werden die IP-Blöcke ebenfalls in diesem Schritt platziert. Wichtig hierbei ist, dass keine Zellen oder Verbindungen die vom IP-Block eingenommene Fläche nutzen dürfen.

Im Schritt „Power Planning“ werden die Versorgungsstrukturen definiert und platziert. Hierzu gehören Ringe und Streifen. Bei der Definition müssen die Verbindungen der globalen Versorgungsstrukturen zu den Power- und Ground-Anschlüssen der Zellen der Technologiebibliothek explizit angegeben werden. Die synthetisierte Netzliste enthält hierzu nämlich keine Informationen. Die Namen der Power- und Ground-Anschlüsse sind generell für alle Zellen einer Technologiebibliothek dieselben und können aus dieser entnommen werden. Nach der Definition können die elektrischen Verbindungen geroutet werden. In Abbildung 6 werden die Befehle gezeigt, die zur Definition verwendet werden können. Mit dem Befehl

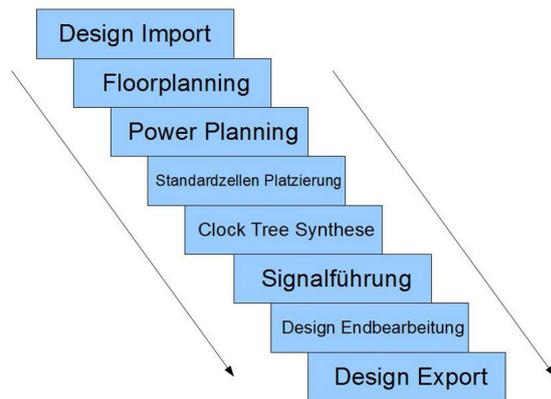


Abbildung 4: Prinzipieller Designflow in Cadence Encounter.

globalNetConnect können einem globalen Netz zum Beispiel die jeweiligen Pins (in der verwendeten Technologie mit vdd! und gnd! bezeichnet) der Standardzellen zugeordnet werden. Das Platzieren der zuvor definierten Strukturen schließt das Power Planning ab.

Im nächsten Schritt werden die in der Netzliste aufgeführten Standardzellen auf dem Floorplan platziert. Hier gibt es bereits verschiedene Möglichkeiten der Optimierung. Zum Beispiel kann das Platzieren mit dem Ziel möglichst optimaler zeitlicher Bedingungen erfolgen (Timing Driven Place and Route). In diesem Fall werden die Leitungslängen zeitlich kritischer Signale angepasst und die zugehörigen Standardzellen entsprechend platziert.

Der Abschnitt Clock Tree-Synthese dient ebenfalls der zeitlichen Optimierung und ist nicht zwingend erforderlich, jedoch im Sinne eines qualitativ hochwertigen Designs sehr zu empfehlen. Hierbei werden in die Taktnetze weitere Standardzellen wie Buffer und Inverter eingefügt, um das Taktsignal gleichmäßig über die gesamte Chipfläche zu verteilen. Wichtig für diesen Teil ist das Erzeugen einer Datei, in welcher die zur Verfügung stehenden Standardzellen spezifiziert sind. Mit dieser Datei kann schließlich ein Clock Tree erzeugt werden. In Abbildung 7 ist zu sehen, wie in einem Skript dieser Schritt erfolgen kann.

Es ist zudem zu sehen, wie Teile des Designflows mittels Steuerflags im Skript übersprungen werden können. Diese Flags werden ebenfalls global definiert. Dadurch wird ein Skript flexibel und es kann vom Nutzer bestimmt werden, welche Schritte ausgeführt werden sollen.

Einen weiteren wichtigen Abschnitt stellt die Signalführung (Signal Routing) dar. In diesem Schritt werden die Standardzellen entsprechend der Vorgaben der Netzliste miteinander verbunden. Hierbei gibt es ebenfalls verschiedene Optimierungsmöglichkeiten. Das Routing selbst kann zum Beispiel wieder nach zeitli-

```
# Import Design:
loadConfig c35b3_std.conf 0
set rda_Input(ui_netlist) $NETLIST_PATH
set rda_Input(ui_topcell) $TOPCELL
set rda_Input(ui_io_file) $IO_FILE_PATH
set rda_Input(ui_timingcon_file) $TIMING_SDC_FILE_PATH
set rda_Input(ui_pwrnet) $POWER
set rda_Input(ui_gndnet) $GROUND
saveConfig $CONFIG_FILE
commitConfig
```

Abbildung 5: Befehle zum Design Import im Skript.

chen Kriterien erfolgen. Eine weitere Möglichkeit stellt die Post Route-Optimierung dar. Hierbei werden verschiedene Routings getestet und schließlich das mit dem besten Zeitverhalten genutzt. Der Aufwand bei diesem Schritt kann sehr hoch sein, da für das Routing sehr viele Lösungen möglich sind.

In der Design-Endbearbeitung werden die Lücken zwischen den Standardzellen und eventuell vorhandenen IO-Pads mit sogenannten Filler-Zellen gefüllt. Zu diesem Schritt zählt auch das Erzeugen von Berichten, um einen Überblick über die Qualität des erzeugten Designs zu erhalten. Es können Berichte zum Beispiel bezüglich der Konnektivität, der Geometrie, des Zeitverhaltens sowie zu verschiedenen Statistiken ausgegeben werden.

Im abschließenden Schritt, dem Design-Export, werden alle für die Weiterverarbeitung benötigten Dateien erzeugt. Hierzu zählen eine neue Netzliste, in der die Verzögerungszeiten aufgrund der Leitungslängen berücksichtigt werden, eine SDF Timing Datei (Standard Delay Format) sowie die Layoutdaten im GDSII (Graphical Design System 2) oder dem neueren OASIS (Open Artwork System Interchange Standard) Format.

IV. FAZIT

Mit einem Skript, welches die dargestellten Schritte abdeckt, kann ein Design weitgehend automatisch erzeugt werden. Für den Anwender ergeben sich dadurch einige deutliche Vorteile. Das Erstellen eines Designs nimmt weniger Zeit in Anspruch. Änderungen können schneller umgesetzt und getestet werden. Ein Skript kann auf verschiedene Projekte angewendet werden. Dabei kann eine einfache Anpassung durch Anwendung von Variablen erzielt werden. Der Designflow ist durch das Skript stärker vorgegeben als bei Arbeiten mit der Benutzeroberfläche und kann somit leichter eingehalten werden. Der Aufwand zur Erstellung eines ersten Skriptes ist aufgrund der automatischen Befehlschnittfunktion in der Benutzeroberfläche von Encounter gering. Zudem sind die Befehle in Encounter ausführlich dokumentiert, so dass der Befehlschnitt leicht editiert und zu einem Skript ausgebaut werden kann.

```

# Power Planning:
clearGlobalNets
# Definition of global Net (vdd! & gnd!)
globalNetConnect $POWER -type pgpin -pin vdd! \
    -inst * -module {} -verbose
globalNetConnect $GROUND -type pgpin -pin gnd! \
    -inst * -module {} -verbose

# Connect VDD-Pad with global Net
globalNetConnect $POWER -type pgpin -pin A \
    -inst vdd1 -verbose
# Connect GND-Pad with global Net
globalNetConnect $GROUND -type pgpin \
    -pin A -inst gnd1 -verbose
    
```

Abbildung 4: Ausschnitt zeigt die Definition der Versorgungsstrukturen.

```

# Clock Tree specification:
createClockTreeSpec \
    -bufFootprint $CTS_BUFFER \
    -invFootprint $CTS_INV \
    -output $CTS_SPEC_FILE

# Clock Tree creation:
if{$CLOCK_TREE}{
    specifyClockTree -file $CTS_SPEC_FILE
    ckSynthesis
    optDesign -postCTS -setup -drv -outdir $TR_PATH
}
    
```

Abbildung 7: Clock Tree Synthese im Skript.

DANKSAGUNG

Die Autoren danken Herrn Dipl.-Ing. J. Hahn-Dambacher für die stetige Unterstützung bei allen Fragen der Rechnerinfrastruktur sowie der Administration von Cadence und der Designkits an der Hochschule Aalen.

LITERATURVERZEICHNIS

- [1] M. Santos, S. Gomes, „Standard Cell Placement and Routing Tutorial“, Juni 2007; https://dSPACE.ist.utl.pt/bitstream/2295/290575/1/tutorial_encounter_ersao2.pdf
- [2] D. Bau, M. Durrenberger, D. Jansen, „ASIC Design Flow from Netlist to Chip with Encounter, IC-Station & Calibre“, Institut für Angewandte Forschung, Offenburg, „Net to Chip Kurs“ an der Hochschule Aalen, Interner Kurs der MPC-Gruppe, 20.11.2006.
- [3] J. M. Emmert, „Cadence First Encounter Tutorial“, August 2007; www.cs.wright.edu/~emmert/tutorials/Cadence_First_Encounter.pdf
- [4] Wei Lai, „SOC Encounter v4.1“, NTU MicroSystem Research Lab, März 2005; <http://www.slideserve.com/neona/soc-encounter-v4-1>



Stefan Thiel strebt den akademischen Grad des B.Eng. in Elektronik und Informationstechnik an und wird diesen vermutlich im Jahr 2013 von der Hochschule Aalen erhalten.



Heinz-Peter Bürkle schloss 1991 das Studium der Elektrotechnik an der Universität Stuttgart mit dem akademischen Grad des Dipl.-Ing. ab und promovierte dort 1997 zum Dr.-Ing. Nach knapp 7 Jahren in Forschung und Entwicklung bei Alcatel erfolgte 2003 der Ruf an die Hochschule Aalen für die Lehrgebiete Halbleitertechnik, Schaltungstechnik und rechnergestützten Schaltkreisentwurf.

Parametrisierte Layout-Module im analogen IC-Entwurf

Simon Gohm, Daniel Marolt, Jürgen Scheible

Zusammenfassung—Ein praktikables Mittel zur Erhöhung des Automatisierungsgrads im analogen IC-Entwurf ist die Verwendung parametrisierter Zellen. Diese sogenannten pCells werden eingesetzt, um determinierte Layouts automatisch zu erzeugen, und zwar in der Regel für einzelne Bauelemente wie Transistoren oder Dioden. Der vorliegende Beitrag zeigt die Potenziale eines erweiterten pCell-Konzepts, mit dem determinierte Layouts als auch Schaltpläne für ganze Schaltungsmodulare automatisch generiert werden können. Als Beispiel wird eine solche Modul-pCell für analoge Stromspiegel beschrieben, die nicht nur die Dimensionierung der Einzeltransistoren, sondern auch verschiedene Transistortypen, beliebige Spiegelverhältnisse und sogar mehrere Topologien sowie weitere Freiheitsgrade implementiert. Das dadurch erzielte Maß an Flexibilität erlaubt es, die zahlreichen schaltungstechnischen Varianten im Analogbereich abzudecken, die ansonsten oftmals Hürden für Automatisierungsansätze darstellen.

Schlüsselwörter—IC-Entwurf, Analog, Layout, Entwurfsautomatisierung, parametrisierte Zellen.

I. EINLEITUNG

In der Geschichte der Mikroelektronik hat sich bis heute die bereits 1965 von Gordon Moore geäußerte Einschätzung bestätigt, dass die Transistorbelegung eines integrierten Schaltkreises exponentiell wächst. So stieg im Durchschnitt der letzten Jahre die Anzahl der fertigmachen Transistoren pro Chip jährlich um etwa 58%, wobei im Gegensatz dazu die Entwurfsproduktivität eines Schaltungsentwicklers, gemessen in der Anzahl umgesetzter Transistoren, lediglich um 21% wächst [1]. Diese Steigerung, die im Wesentlichen der Entwurfsautomatisierung zu verdanken ist, liegt somit weit unter den Möglichkeiten der Fertigungstechnik. Die daraus resultierende Kluft zwischen potenzieller Entwurfskomplexität und tatsächlicher Entwurfsproduktivität wird als Entwurfsschere bezeichnet.

Die Entwurfsproduktivität im Analogbereich ist heute deutlich geringer als bei digitalen Schaltungen. Während im Digitalbereich viele Entwurfsschritte

Simon Gohm, Simon.Gohm@Student.Reutlingen-University.de,
 Daniel Marolt, Daniel.Marolt@Reutlingen-University.de,
 und Jürgen Scheible, Juergen.Scheible@Reutlingen-University.de,
 sind Mitglieder des Robert Bosch Zentrums für Leistungselektronik,
 Hochschule Reutlingen, Alteburgstraße 150, 72762 Reutlingen

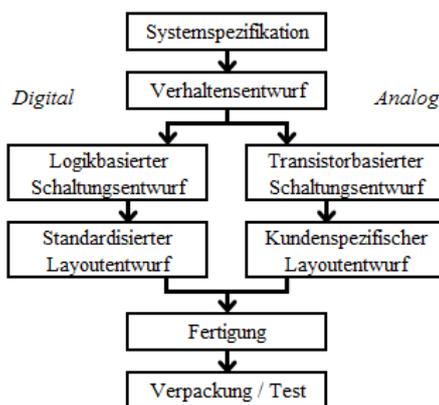


Abbildung 1: Entwurfsprozess eines integrierten Schaltkreises. Diese vereinfachte Darstellung unterscheidet in Bezug auf den Schaltungs- und Layoutentwurf zwischen digital und analog.

durch Synthesealgorithmen weitgehend automatisiert worden sind, ließen sich derartige top-down-gerichtete Methoden bisher nicht erfolgreich auf den Analogentwurf übertragen, sondern scheitern in der Praxis meistens an den komplexen funktionalen Randbedingungen, die typischerweise von analogen Schaltungen ausgehen. Daher ist der Analogentwurf auch heute noch vorwiegend manuell geprägt, wobei insbesondere die Erzeugung des physikalischen Layouts von allen Entwurfsschritten am wenigsten durch kommerziell verfügbare Werkzeuge unterstützt wird [2].

Der vorliegende Beitrag präsentiert einen praxisnahen Ansatz, den Automatisierungsgrad im analogen IC-Entwurf durch parametrisierte Zellen zu erhöhen. Diese sogenannten pCells dienen dazu, determinierte (das heißt „vorgedachte“) Layouts automatisch zu generieren, und zwar meist für elementare Bauelemente wie Transistoren oder Dioden. Der hier vorgestellte Ansatz zeigt, wie mit parametrisierten Zellen ganze Layout- als auch Schaltplan-Module generiert werden können, die wiederum andere pCells enthalten. Dabei wird die pragmatische Philosophie verfolgt, durch schrittweise Automatisierungen den Entwurfsfluss bottom-up zu optimieren und das Expertenwissen des Schaltungsentwicklers durch interaktive Bedienung umfassend in den Automatismus einzubeziehen.

Dieser Beitrag ist wie folgt strukturiert: Kapitel II gibt einen Überblick bisheriger Ansätze zur Entwurfsautomatisierung im Vergleich zwischen digital und analog. Kapitel III beschreibt die Idee, parametrisierte Zellen auf Modulebene einzusetzen und Kapitel IV zeigt als Beispiel die Implementierung einer Stromspiegel-pCell für den Schaltungs- und Layoutentwurf. Kapitel V zieht ein abschließendes Fazit.

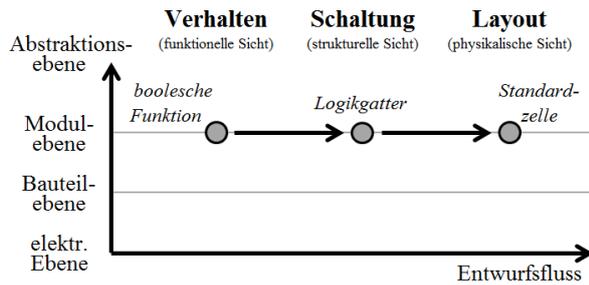


Abbildung 2: Hierarchische Kontinuität im digitalen Entwurfsfluss. Funktionsblöcke (zum Beispiel boolesche Verknüpfungen) werden durchgehend auf Modulebene (mindestens Gatterebene) realisiert.

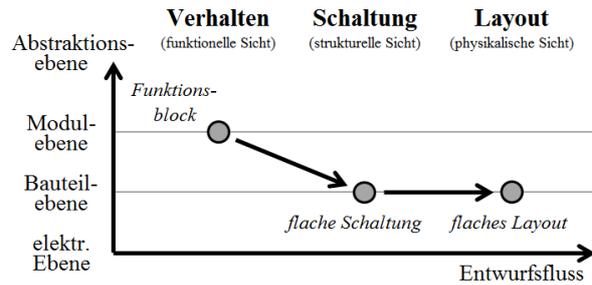


Abbildung 3: Hierarchiebruch im analogen Entwurfsfluss. Funktionsblöcke werden in Schaltung und Layout meistens auf Bauteilebene (Transistorebene) umgesetzt.

II. AUTOMATISIERUNGSANSÄTZE IM IC-ENTWURF

Die zunehmende Komplexität integrierter Schaltungen wäre ohne rechnergestützte Entwurfswerkzeuge mittlerweile nicht mehr beherrschbar und die elektronische Entwurfsautomatisierung (*Electronic Design Automation, EDA*) gilt heutzutage als Schlüsseltechnologie für Industrie, Wirtschaft und Gesellschaft [3]. Ein wesentlicher Fokus von EDA-Entwicklungen in jüngster Zeit lag auf der engeren Verknüpfung von Entwurfsschritten mit dem Ziel, die Durchgängigkeit des Entwurfsflusses zu verbessern. Der Entstehungsprozess einer integrierten Schaltung ist in Abbildung 1 vereinfacht dargestellt. Diesbezüglich soll im Folgenden, besonders in Bezug auf den Schaltungs- und Layoutentwurf, zwischen digital und analog unterschieden werden.

Der Digitalentwurf lebt von der Idee, eine Entwurfsbeschreibung von ihrer physikalischen Implementierung zu trennen. Durch diese Abstraktion ist es möglich, aus einer gegebenen Systemspezifikation automatisch ein entsprechendes IC-Layout zu generieren. Diese top-down-Synthese wird auch als *Silicon Compilation* bezeichnet. Zur Lösung des NP-harten [4] Layoutproblems werden heuristische Algorithmen eingesetzt, welche die einzelnen Layoutschritte wie Platzierung und Verdrahtung in der Regel sukzessive vornehmen. Außerdem werden zur Reduzierung der Komplexität künstliche entwurfsmethodische Randbedingungen eingeführt, so dass ein Algorithmus beispielsweise auf vorgefertigte Layoutelemente mit fester Struktur zurückgreift, die als Standardzellen in einer Bibliothek verfügbar sind. Diese Maßnahmen beeinträchtigen zwar die Optimalität des Entwurfs, können aber dank der diskreten Natur digitaler Signale noch toleriert werden.

Vergleichbare Ansätze, Syntheseverfahren auch im Analogbereich umzusetzen, konnten sich trotz jüngster Fortschritte bislang nicht in der Praxis etablieren. Für den Digitalentwurf ist die Beherrschung steigender Transistorzahlen die maßgebliche Herausforderung, die sich aus dem Ziel der anhaltenden Miniaturisierung (auch "More Moore" genannt) ergibt. Im Analogen wird dieses Problem nochmals entscheidend verschärft durch den Wunsch, das funktionelle Spek-

trum von Halbleiterprodukten zu erhöhen ("More than Moore"). Analoge Schaltungsteile bilden meist die Schnittstelle eines ICs zu seiner realen Außenwelt und umfassen immer vielfältigere Sensoren und Aktoren wie optoelektronische Komponenten und mikromechanische Systeme. Als solche müssen sie mit hohen Spannungen, kontinuierlichen Signalwerten, verschiedenartigen physikalischen Größen, wechselseitigen Störeinflüssen und parasitären Effekten umgehen. Diese Anforderungen und die daraus resultierenden funktionalen Randbedingungen verlangen im analogen Layout die Ausnutzung einer größeren Vielfalt an Freiheitsgraden. Dies kann momentan nur durch manuellen kundenspezifischen Entwurf erzielt werden, da Heuristiken und entwurfsmethodische Standardisierungen zu inakzeptablen Einbußen der Layoutqualität führen würden. Beispielsweise lassen sich analoge Schaltungsteile nicht auf Standardzellen reduzieren, sondern sind charakterisiert durch die häufig auftretende Notwendigkeit schaltungstechnischer Abwandlungen. Automatisierungsansätze müssen diese Variabilität unbedingt abdecken, um sich in der praktischen Anwendung behaupten zu können.

A. Parametrisierte Zellen (pCells)

Layoutdaten sind in ihrer grundlegenden Form konkrete grafische Daten. Sie bestehen aus geometrischen Primitiven, die in Layoutzellen zusammengefasst werden können. Vor allem früher wurden elementare Bauteile wie Transistoren oder Widerstände als derartige Layoutzellen mit fixer Grafik bereitgestellt. Um andere schaltungsspezifische Geometrien zu erhalten, mussten die Grafikdaten vom Layoutentwickler in Handarbeit angepasst werden. Mittlerweile erfolgt diese Dimensionierung überwiegend automatisch mit Hilfe parametrisierter Zellen. Solche pCells enthalten im Gegensatz zu fixen Zellen keine Grafikdaten, sondern bestehen aus Code, der anhand von benutzerdefinierten Parameterwerten entsprechende Layouts erzeugt. Diese Form der Automatisierung ist im Gegensatz zu Synthesearchgorithmen keine eigenintelligente Lösungsstrategie, sondern vielmehr ein Ablaufskript, das stets eine vom pCell-Entwickler vorab bestimmte Lösung aus einem definierten Parameterraum liefert.

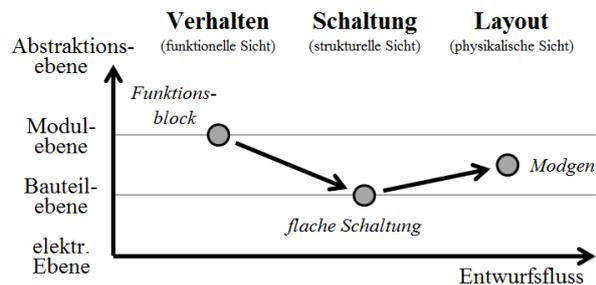


Abbildung 4: Modgen-Konzept im analogen Entwurfsfluss. Durch eine Schaltungsstrukturerkennung wird der Hierarchiebruch im Layout bedingt korrigiert.

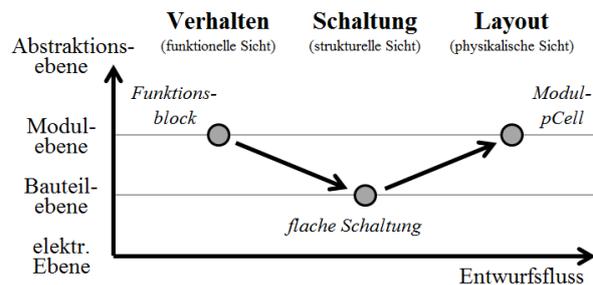


Abbildung 5: Analoger Entwurfsfluss mit flacher Schaltung und Layout-Modul-pCell. Durch eine Schaltungsstrukturerkennung wird der Hierarchiebruch im Layout vollständig korrigiert.

B. Hierarchische Schaltungs- und Layout-Module

Im Entwurf kann man zwischen verschiedenen Abstraktionsebenen unterscheiden. Ein bemerkenswertes Merkmal des Digitalentwurfs ist, dass sich der kreative Teil des Entwurfsprozesses ausnahmslos oberhalb der Abstraktionsebene der Transistoren (Bauteilebene) bewegt, denn die grundlegenden atomaren Bauelemente sind hier nicht Transistoren, sondern Module. Abbildung 2 zeigt beispielhaft, dass dies für alle drei Sichten des bekannten Y-Modells [5] gilt: funktionelle Sicht (Verhalten), strukturelle Sicht (Schaltung) und physikalische Sicht (Layout). Für die funktionelle Sicht wird das Verhalten eines Entwurfs mit Hardwarebeschreibungssprachen ausgedrückt. Die Verhaltensbeschreibung bewegt sich dabei mindestens auf der Gatterebene, die zugleich die niedrigste Modulebene darstellt; oder sie findet auf darüber liegenden Abstraktionsebenen statt, beispielsweise der Registertransferebene. Demgemäß stehen für die elektrische Schaltung (strukturelle Sicht) digitale Bausteine wie logische Gatter oder Speicherblöcke zur Verfügung. Im physikalischen Layout werden für diese Komponenten entsprechende Standardzellen eingesetzt.

Im analogen Entwurfsfluss hält sich eine derartige Kontinuität der Abstraktion nicht (siehe Abbildung 3). Zwar bewegt sich die funktionelle Sicht auch hier vornehmlich auf Modulebene, da ein Entwickler bei der Systemspezifikation meistens in Funktionsblöcken wie Wandlern und Filtern "denkt". Diese Hierarchie wird aufgrund der topologischen Variabilität in der strukturellen Sicht jedoch fast immer aufgelöst und die Funktionsblöcke werden dabei in der Schaltung auf Transistorebene individuell erstellt. Ebenso werden daraufhin auch im Layout üblicherweise keine Module, sondern Einzelbauelemente verwendet.

Im Allgemeinen bringt das Verlassen einer Abstraktionsebene wie in Abbildung 3 mehrere Nachteile mit sich. Zum einen steigt dabei meist der Entwurfsaufwand für Schaltung und Layout, andererseits wird durch entstehende Diskontinuitäten oftmals ein flüssiger Entwurfsprozess verhindert. Vor allem jedoch birgt dieser Bruch in der Hierarchie einen Informationsverlust: die Existenz eines Schaltungsteils als geschlossenes Modul vermittelt dessen Zweck und damit einhergehend auch eventuelle elektrische und geomet-

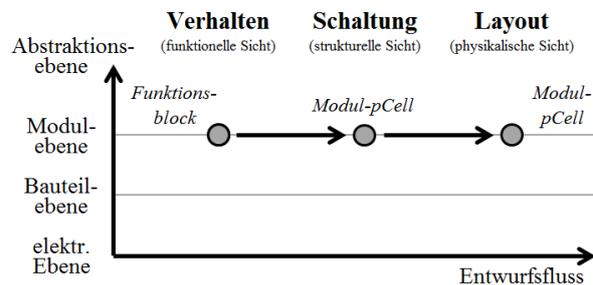


Abbildung 6: Analoger Entwurfsfluss mit Modul-pCells für Schaltung und Layout. Dadurch bleibt ein Funktionsblock ohne Hierarchiebruch durchgehend auf Modulebene erhalten.

rische Randbedingungen für Schaltung und Layout, deren Einhaltung für das korrekte Verhalten des Moduls notwendig ist. Verlässt man eine Modulebene, dann ist die Information über die Funktion des Moduls nur noch implizit in dessen Schaltungstopologie gegeben und wird in der Praxis -wenn überhaupt- meist durch textuelle Kommentare oder verbale Kommunikation ausgetauscht.

Das Modgen-Konzept [6] ist ein Ansatz, den oben beschriebenen Informationsverlust zu revidieren. Abbildung 4 zeigt die resultierenden Abstraktionsniveaus bezogen auf einen spezifischen Funktionsblock, wie beispielsweise einen Differenzverstärker. Dieser wird zwar in der Schaltung auf Transistorebene erstellt, dann aber durch eine Schaltungsstrukturerkennung als funktionelle Einheit detektiert. Auch im Layout besteht die Teilschaltung aus einzelnen Bauelementen, allerdings können diese in der heutigen Entwicklungsumgebung als zusammengehörige Gruppe behandelt werden. Die für das Layout wichtigen funktionalen Randbedingungen können teilweise automatisch deduziert und von verfügbaren Platzierungswerkzeugen berücksichtigt werden. Ähnliche Konzepte, die eine Schaltungsstrukturerkennung auf eine flache Topologie anwenden, werden in [7] und [8] beschrieben.

III. PARAMETRISIERTE ZELLEN AUF MODULEBENE

Wie bereits erwähnt, stammen parametrisierte Zellen ursprünglich aus dem Layoutentwurf und dienen dazu, die Geometrien für einzelne Bauelemente automatisch

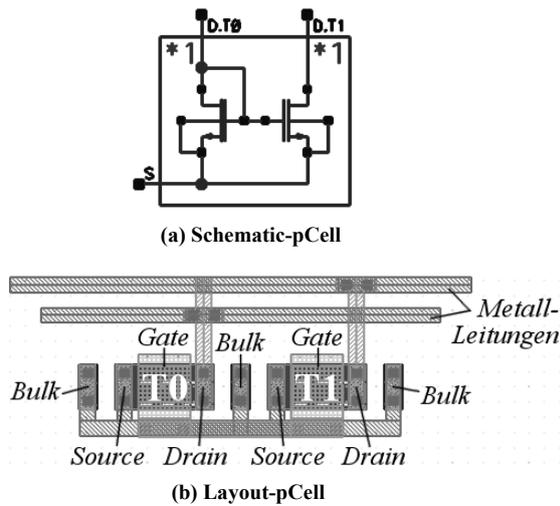


Abbildung 7: Stromspiegel-pCells in Standardkonfiguration jeweils mit NMOS-Transistoren, 1 Ausgang und Spiegelverhältnis 1:1.

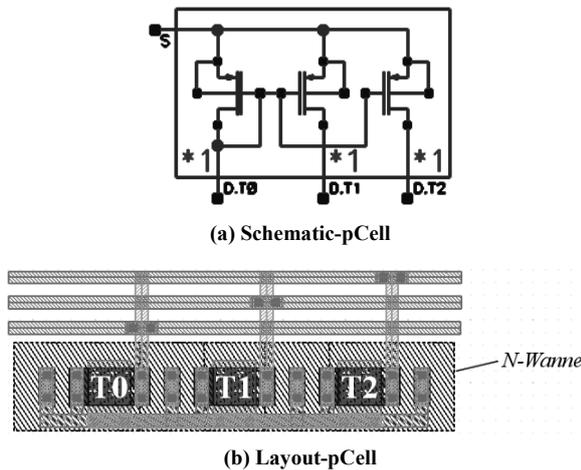


Abbildung 8: Stromspiegel-pCell-Varianten, jeweils mit PMOS-Transistoren, 2 Ausgängen und Spiegelverhältnis 1:1:1.

zu erzeugen. Dieses Kapitel beschreibt, wie solche pCells über ihren grundlegenden Zweck hinaus verwendet werden können, um im analogen Entwurfsprozess die Handhabung von Funktionsblöcken konsequent auf derselben Abstraktionsebene zu halten. Dazu werden mit parametrisierten Zellen komplexe Module realisiert: zunächst für das Layout, anschließend auch für den Schaltungsentwurf.

In [9] wurde ein Ansatz präsentiert, bei dem Modul-pCells im Layout eingesetzt werden, jedoch nicht im Schaltungsentwurf. Dabei deckt die Parametrisierbarkeit der Module die schaltungstechnischen Varianzen ab, die im Analogbereich notwendig sind. Zur Einbindung der pCells in den Entwurfsfluss wird wie beim Modgen-Konzept eine Schaltungsstrukturerkennung durchgeführt, um den Hierarchiebruch zwischen Funktions- und Schaltungssicht für das Layout wieder zu korrigieren (Abbildung 5).

Parametrisierte Zellen können nicht nur für das Layout, sondern als sogenannte Schematic-pCells auch für den Schaltungsentwurf entwickelt werden, wengleich

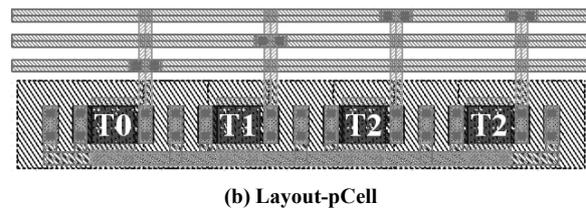
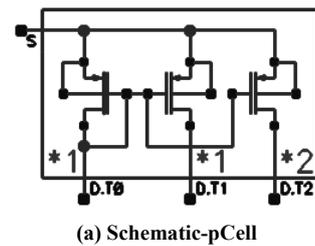


Abbildung 9: Stromspiegel-pCell-Varianten, jeweils mit PMOS-Transistoren, 2 Ausgängen und Spiegelverhältnis 1:1:2.

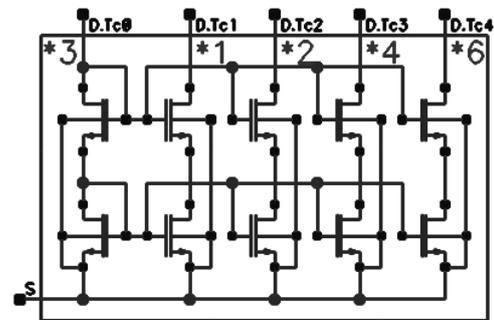


Abbildung 10: Schematic Stromspiegel-pCell-Variante in Kaskode-Topologie mit NMOS-Transistoren, 4 Ausgängen und einem Spiegelverhältnis von 3:1:2:4:6.

dieser Verwendungszweck in bisherigen Ansätzen wie beispielsweise [10] nur begrenzt verfolgt worden ist. Zwar ist die Verwendung hierarchisch aufgebauter Schaltungen gängige Praxis, beschränkt sich bislang aber fast ausschließlich auf fixe Module. Dabei muss jede einzelne topologische Abwandlung als eigenes Modul realisiert werden, was letztlich in einer nicht handhabbaren Variantenexplosion resultiert. Ein entsprechend parametrisiertes Schaltplan-Modul behebt dieses Problem und kann in Kombination mit einer korrespondierenden Layout-Modul-pCell ein konsequentes Abstraktionsniveau im Entwurfsverlauf aufrechterhalten, wie in Abbildung 6 dargestellt ist.

IV. BEISPIEL EINES PARAMETRISIERTEN STROMSPIEGEL-MODULS

Der im vorangegangenen Kapitel diskutierte Ansatz, im Schaltungs- als auch Layoutentwurf parametrisierte Zellen auf Modulebene umzusetzen, wird im Folgenden anhand eines Stromspiegel-Moduls demonstriert. Die nächsten drei Abschnitte beschreiben die Funktionsweise eines Stromspiegels und zeigen dessen Implementierung als Schematic- und Layout-pCell. Die hierbei berücksichtigten Parameter werden im letzten Abschnitt umfassend aufgelistet.

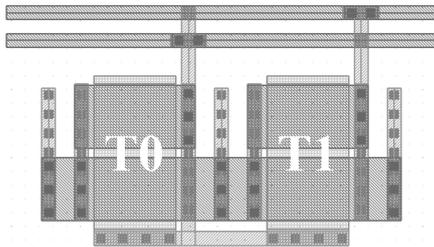


Abbildung 11: Layout Stromspiegel-pCell in Standardausführung wie in Abbildung 7, jedoch mit größeren Transistoren. Sowohl Kanallänge als auch Kanalbreite der Transistoren sind größer.

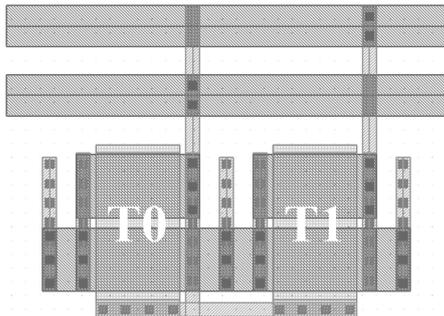


Abbildung 12: Layout Stromspiegel-pCell, identisch zur Variante in Abbildung 11, jedoch mit breiteren Metall-Leitungen und somit einer höheren Stromtragfähigkeit.

A. Verhalten eines Stromspiegels (Funktion)

Ein Stromspiegel ist eine Transistorschaltung, die praktisch in jeder analogen integrierten Schaltung eingesetzt wird. Sie dient dazu, einen gegebenen Referenzstrom ein- oder mehrmals zu reproduzieren. Dabei können über die Dimensionierung der Transistoren beliebige Spiegelverhältnisse erreicht werden.

Die hohe Genauigkeit, die in der Regel für einen Stromspiegel gefordert wird, bedingt, dass dessen Transistoren sich auf dem IC möglichst gleich verhalten und daher im Layout möglichst symmetrisch angeordnet werden müssen. Die Umsetzung dieser Anforderung nennt man *Matching* und wird im Wesentlichen erreicht durch Verwendung gleicher Grundelemente, die in relativer Nähe und gleicher Orientierung symmetrisch angeordnet werden.

Ein gängiges Mittel, um mehrere Transistoren zu *matchen*, besteht darin, diese im Layout aufzuteilen und miteinander zu verschachteln. Dazu werden die einzelnen Transistoren jeweils in Parallelschaltungen mehrerer Teiltransistoren aufgesplittet und im Layout gemäß einem Platzierungsmuster (beispielsweise notiert in der Form ABBA) angeordnet. Zugunsten eines guten Matchings werden sämtliche Teiltransistoren üblicherweise mit derselben Größe dimensioniert.

Wendet man dieses Prinzip auch auf Stromspiegel an, dann werden beliebige Spiegelverhältnisse nicht über die Größe der Teiltransistoren, sondern über ihre Anzahl erzielt. Die nachfolgenden Beispiele veranschaulichen das Prinzip des Verschachtelns.

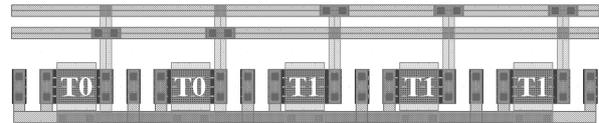


Abbildung 13: Layout Stromspiegel-pCell mit Spiegelverhältnis 2:3. Die Transistoren T0 und T1 sind aufgesplittet, aber nicht verschachtelt. Das AABBB-Muster erzielt kein gutes Matching.

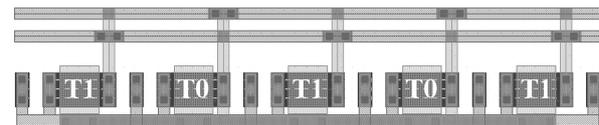


Abbildung 14: Layout Stromspiegel-pCell wie in Abbildung 13, jedoch sind die Transistoren nach dem Muster BABAB angeordnet. Ein solches Common Centroid liefert ein sehr präzises Matching.

B. Umsetzung der Schematic-pCell (Struktur)

Die Implementierung der Stromspiegel-pCell für den Schaltungsentwurf wurde bereits in [11] vorgeführt. Die pCell bietet als Freiheitsgrade mehrere Topologien, eine beliebige Anzahl an Ausgängen, verschiedene Transistortypen und einige weitere Parameter an.

Abbildung 7 (a) zeigt die pCell in ihrer Standardkonfiguration als simplen NMOS-Stromspiegel mit dem Eingang D.T0, einem einzigen Ausgang D.T1 und dem Spiegelverhältnis 1:1. In Abbildung 8 (a) ist die pCell mit PMOS-Transistoren und zwei Ausgängen dargestellt. Abbildung 9 (a) zeigt denselben Stromspiegel, jedoch mit einem 1:1:2 Spiegelverhältnis. In Abbildung 10 ist eine andere Variante der pCell in Kaskode-Topologie zu sehen.

Die Implementierung der hier beschriebenen Stromspiegel-pCell erreicht einen bisher nicht dokumentierten Komplexitätsgrad. Daher repräsentieren die aufgeführten Beispiele nur einen ausgewählten Bruchteil der praktisch unendlich großen Variantenvielfalt.

C. Umsetzung der Layout-pCell (Physik)

Eine Layout-pCell stellt im physikalischen Chip-Entwurf das Gegenstück zur korrespondierenden Schematic-pCell dar. Ihre Implementierung muss für jede Parameterkonstellation gewährleisten, dass die Layout-pCell erstens exakt mit der jeweiligen Schematic-pCell-Variante übereinstimmt und zweitens keine Layoutregeln verletzt, die in der zugrunde liegenden Halbleitertechnologie gelten. Zum Vergleich zeigen die Abbildungen 7 (b), 8 (b) und 9 (b) die Layout-pCells des Stromspiegels, die den oben erwähnten Varianten der Schematic-pCell entsprechen.

Die nächsten Beispiele illustrieren weitere pCell-Parameter, welche die Dimensionierung und layout-technische Umsetzung des Stromspiegel-Moduls veranschaulichen. In Abbildung 11 ist die Layout-pCell in Standardausführung (vgl. Abbildung 7), jedoch mit größeren Transistoren dargestellt. Abbildung 12 zeigt dieselbe Variante mit breiteren Metall-Leitungen und somit einer höheren Stromtragfähigkeit.

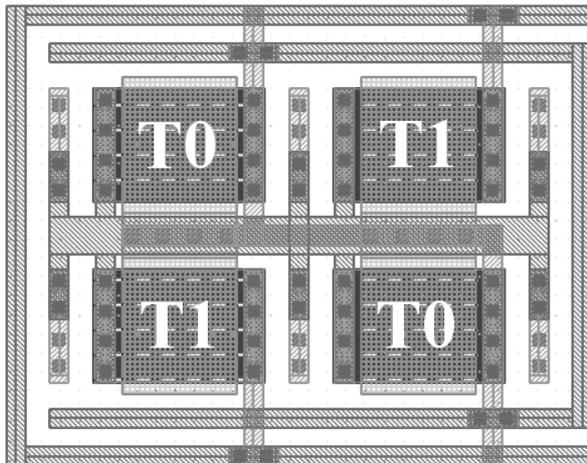


Abbildung 15: Layout Stromspiegel-pCell mit Spiegelverhältnis 1:1 gemäß zweireihigem AB/BA-Muster. Diese Verschachtelung erzielt ein präziseres Matching als eindimensionale Anordnungen.

Abbildung 13 und Abbildung 14 demonstrieren das Prinzip des Verschachtelns anhand eines Stromspiegels mit einem Ausgang. Der Transistor T0 ist in zwei Teiltransistoren, T1 in drei Teiltransistoren aufgesplittet, woraus sich ein Spiegelverhältnis von 2:3 ergibt. In Abbildung 13 sind die Teiltransistoren in einer AABBB-Anordnung ohne Verschachtelung zu sehen und erzielen kein gutes Matching. Per pCell-Parameter kann nun beispielsweise ein BABAB-Muster eingestellt werden, wie es in Abbildung 14 gezeigt ist. Eine derartige Anordnung wird aufgrund des gemeinsamen geometrischen Transistorschwerpunktes allgemein als *Common Centroid* bezeichnet und liefert ein sehr präzises Matching, weil sich Einflüsse linearer Gradienten dadurch vollständig aufheben [12].

Im Allgemeinen kann die Qualität des Matchings durch eine zweidimensionale Verschachtelung noch weiter erhöht werden. Diese Form der Platzierung wurde in der Implementierung der Layout-pCell ebenfalls umgesetzt, sodass bei der Einstellung des Platzierungsmusters durch den Anwender mehrere Reihen angegeben werden können. Abbildung 15 zeigt als Beispiel die häufig eingesetzte AB/BA-Anordnung, die auch als *Cross-Coupled Pair* bezeichnet wird. Je großflächiger eine solche Anordnung mit steigender Transistorgröße wird, desto anfälliger wird sie gegenüber Mismatching. Dieses kann umso besser kompensiert werden, je mehr man die Transistoren in horizontaler als auch vertikaler Richtung streut. Als Beispiel ist in Abbildung 16 ein ABBA/BAAB/ABBA/BAAB-Muster dargestellt. Die wesentliche Problematik derart komplexer Verschachtelungen liegt in der Schwierigkeit ihrer Verdrahtung, erfolgt hier jedoch automatisch durch die implementierte Layout-pCell.

D. Übersicht aller pCell-Parameter

Die nachfolgende Auflistung beschreibt alle Parameter, die gegenwärtig von der pCell unterstützt werden.

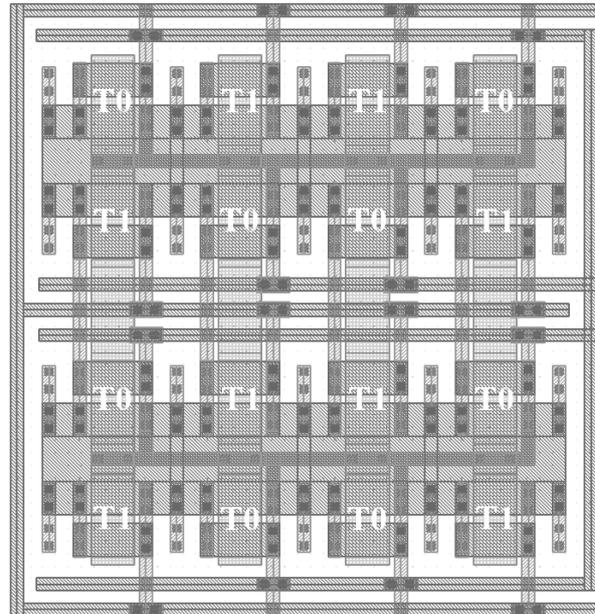


Abbildung 16: Layout Stromspiegel-pCell mit Spiegelverhältnis 1:1 und einem ABBA/BAAB/ABBA/BAAB-Muster. Eine solche Verteilung der Transistoren in horizontaler und vertikaler Richtung erzielt auch bei großflächigen Anordnungen ein präzises Matching.

Topologie:

Für die Schaltungstopologie der Stromspiegel-pCell stehen dem Anwender drei verschiedene Varianten zur Verfügung: Einfacher Stromspiegel, Kaskode-Stromspiegel und Wide-Swing-Kaskode-Stromspiegel.

Transistortyp:

Für die Transistoren des Stromspiegels kann der Anwender zwischen NMOS- und PMOS-Typen mit jeweils verschiedenen Spannungsfestigkeiten wählen. Bei einer Kaskode-Topologie kann für die Kaskodetransistoren eine andere Spannungsfestigkeit als für die Spiegeltransistoren vorgegeben werden.

Transistorgröße:

Zur Dimensionierung der Transistoren ist die Eingabe einer beliebigen Kanalweite und -länge möglich, sowie eine Aufteilung in mehrere sogenannte Gatefinger. Bei einer Kaskode-Topologie können die Kaskodetransistoren anders dimensioniert werden als die Spiegeltransistoren.

Anzahl der Ausgänge:

Darf 1 oder größer sein. Dieser Parameter ist mit den nachfolgend aufgeführten Parametern *Transistorfaktoren* und *Platzierungsmuster* verknüpft. Das bedeutet, die Änderung eines Parameterwertes kann die beiden anderen Werte beeinflussen.

Transistorfaktoren:

Wird als Liste natürlicher Zahlen eingegeben und legt individuell fest, wie viele Teiltransistoren für den Eingang und pro Ausgang eingesetzt werden, um ein gewünschtes Spiegelverhältnis zu erzielen und Verschachtelungen zu ermöglichen.

Platzierungsmuster:

Definiert die Anordnung der Teiltransistoren im Layout. Der Anwender ist hierbei nicht auf vordefinierte Platzierungsmuster beschränkt, sondern kann mit der ABBA/BAAB-Notation beliebige mehrreihige Verschachtelungen erreichen.

Verdrahtung Source/Drain:

Bestimmt eine Verdrahtungsvariante (z.B. über oder neben dem aktiven Gebiet) und die entsprechende Verdrahtungsebene (Metall1/Metall2).

Bulk-Source verbinden:

Legt fest, ob die Substrat-Anschlüsse (Bulks) der Transistoren mit dem gemeinsamen Source-Netz kurzgeschlossen werden sollen. Ansonsten werden die Bulks nur untereinander verbunden.

Leiterbahnbreite und -abstand:

Sind standardmäßig auf den durch die Halbleitertechnologie vorgegebenen Minimalwert gesetzt, können aber für Source und Drain individuell vergrößert werden, um höhere Ströme zu führen.

V. FAZIT

Dieser Beitrag diskutiert den Ansatz, parametrisierte Zellen im analogen IC-Entwurf auf ganze Schaltplan- und Layout-Module anzuwenden. Modul-pCells erlauben es, Funktionsblöcke während des Entwurfsverlaufs konsequent oberhalb der Bauteilebene zu handhaben, wodurch der Entwurfsaufwand erheblich reduziert und die Berücksichtigung funktionaler Randbedingungen gewährleistet oder erleichtert wird.

Die zahlreichen schaltungstechnischen Varianten, die im Analogen häufig Hürden für Automatisierungsansätze sind, können durch die Parametrisierbarkeit abgedeckt werden. Als Beispiel wird die Implementierung einer Schematic- und Layout-pCell für Stromspiegel präsentiert, die mehrere Topologien, verschiedene Transistortypen, beliebige Spiegelverhältnisse und weitere Freiheitsgrade berücksichtigt.

Im Gegensatz zu top-down gerichteten Synthesearchivgorithmen, die sich im Analogbereich bislang nicht behaupten konnten, stellen Modul-pCells einen pragmatischen Ansatz dar, den Automatisierungsgrad im IC-Entwurfsprozess bottom-up zu erhöhen. Der Automatismus beschränkt sich dabei auf rein deterministische Schritte und ermöglicht dem Benutzer überall dort, wo es auf erfahrungsgestütztes Expertenwissen ankommt, interaktive Einflussnahme. Dieser Grundsatz bildet eine wichtige Voraussetzung, damit das Modul-pCell-Konzept in der Praxis weiter an Akzeptanz gewinnen kann.

DANKSAGUNG

Diese Arbeit wurde in Abstimmung mit der Robert Bosch GmbH durchgeführt, daher bedanken wir uns bei Christel Bürzele, Göran Jerke und Vinko Marolt für die fruchtbare Kooperation seitens der Firma.

LITERATURVERZEICHNIS

- [1] International Technology Roadmap for Semiconductors (ITRS): 1999, 2001, 2003 Editions.
- [2] H. Gräß (Hrsg.), "Analog Layout Synthesis: A Survey of Topological Approaches", Springer, New York, 2011.
- [3] T. Meister, "Pinzuordnungs-Algorithmen zur Optimierung der Verdrahtbarkeit beim hierarchischen Layoutentwurf", Dissertation, VDI Verlag, Düsseldorf, 2012.
- [4] J. Lienig, "Layoutsynthese elektronischer Schaltungen", Springer Verlag, Berlin Heidelberg, 2006.
- [5] D. Gajski, R. Kuhn, "Guest Editor's Introduction: New VLSI Tools", IEEE Computer, Dec. 1983.
- [6] Cadence, "Virtuoso Unified Custom Constraints User Guide", Cadence Design Systems Inc., San Jose, 2011.
- [7] V. Meyer zu Bexten, "User-Controlled Layout Synthesis for Analog Integrated Circuits", Dissertation, Shaker Verlag, Universität Dortmund, 1994.
- [8] H. Gräß, S. Zizala, J. Eckmüller and K. Antreich, "The Sizing Rules Method for Analog Integrated Circuit Design", IEEE/ACM International Conference on Computer-Aided Design (ICCAD), pp. 343–349, 2001.
- [9] D. Marolt, J. Scheible, G. Jerke, "The Application of Layout Module Generators upon Circuit Structure Recognition", Proceedings of CDNLive! EMEA, München, Mai 2011.
- [10] P. Bhushan, R. Mitra, "Schematic pCell Implementation in Virtuoso Platform", Proceedings of International Cadence Users Group Conference, Santa Clara, 2004.
- [11] D. Marolt, J. Scheible, "Parameterized Cells in Analog IC Design - Example of a Schematic/Symbol Current Mirror pCell", University Booth at Design Automation and Test in Europe Conference, Dresden, März 2012.
- [12] A. Hastings, "The Art of Analog Layout – Second Edition", Pearson Prentice Hall, New Jersey, 2006.



Simon Gohm erhielt den akademischen Grad des B. Eng. in Elektro- und Informationstechnik im Jahr 2010 von der Hochschule Ravensburg-Weingarten. Derzeit absolviert er ein Masterstudium der Leistungs- und Mikroelektronik am Robert Bosch Zentrum für Leistungselektronik in Reutlingen. In seiner Masterarbeit beschäftigt er sich derzeit mit der Entwicklung parametrisierter Layout-Module im analogen IC-Entwurf.



Daniel Marolt studierte Mechatronik an der Hochschule Reutlingen und erhielt dort im Jahre 2008 den akademischen Grad B. Eng. und 2009 den Abschluss M. Sc. Seit 2010 ist er als wissenschaftlicher Mitarbeiter an der Hochschule Reutlingen tätig, wo er 2011 am Robert Bosch Zentrum für Leistungselektronik Aufgaben in Forschung und Lehre übernahm und nun in Kooperation mit der Robert Bosch GmbH an EDA-Projekten arbeitet.



Jürgen Scheible studierte Elektrotechnik an der TH Karlsruhe, wo er im Jahre 1991 auf dem Gebiet der EDA zum Dr.-Ing. promovierte. Anschließend war er bei der Robert Bosch GmbH im Geschäftsbereich Automotive Electronics tätig, zuletzt als Leiter der Entwicklungsabteilung ASIC Layout Design. Seit 2010 ist er Professor für EDA am Robert Bosch Zentrum für Leistungselektronik der Hochschule Reutlingen. Seine Forschungsinteressen sind Constraint-Driven Design und Automatisierung des analogen IC-Entwurfs.

MULTI PROJEKT CHIP GRUPPE

Hochschule Aalen

Prof. Dr. Bürkle, (07361) 576-2103
heinz-peter.buerkle@htw-aalen.de

Hochschule Albstadt-Sigmaringen

Prof. Dr. Rieger, (07431) 579-124
rieger@hs-albsig.de

Hochschule Esslingen

Prof. Dr. Lindermeir, (0711) 397-4221
walter.lindermeir@hs-esslingen.de

Hochschule Furtwangen

Prof. Dr. Rülling, (07723) 920-2503
rue@hs-furtwangen.de

Hochschule Heilbronn

Prof. Dr. Gessler, (07940) 1306-184
gessler@hs-heilbronn.de

Hochschule Karlsruhe

Prof. Dr. Koblitz, (0721) 925-2238
rudolf.koblitz@hs-karlsruhe.de

Hochschule Konstanz

Prof. Dr. Burmberger, (07531) 206-255
gregor.burmberger@htwg-konstanz.de

Hochschule Mannheim

Prof. Dr. Paul, (0621) 292-6351
g.paul@hs-mannheim.de

Hochschule Offenburg

Prof. Dr. Jansen, (0781) 205-267
d.jansen@hs-offenburg.de

Hochschule Pforzheim

Prof. Dr. Kesel, (07231) 28-6567
frank.kesel@hs-pforzheim.de

Hochschule Ravensburg-Weingarten

Prof. Dr. Siggelkow, (0751) 501-9633
siggelkow@hs-weingarten.de

Hochschule Reutlingen

Prof. Dr. Kreutzer, (07121) 271-7059
hans.kreutzer@hochschule-reutlingen.de

Hochschule Ulm

Prof. Dipl.-Phys. Forster, (0731) 50-28338
forster@hs-ulm.de

www.mpc.belwue.de