

# MPC

MULTI PROJEKT CHIP GRUPPE  
BADEN - WÜRTTEMBERG

**Herausgeber:** Hochschule Ulm    **Ausgabe:** 53    **ISSN** 1868-9221    **Workshop:** Esslingen Februar 2015

**1 ASICs für medizinische Geräte und Implantate**

M. Nawito, H. Richter, C. Scherjon, J. N. Burghartz, IMS Chips Stuttgart

**7 Low-Power-SC-Wandler mit hoher variabler Eingangsspannung**

D. Lutz, P. Renz, B. Wicht, Robert Bosch Zentrum für Leistungselektronik, Reutlingen

**13 Entwicklung und Layoutentwurf eines Analog-Digital-Wandlers mit 12 Bit Auflösung in einer 180 nm-CMOS-Technologie**

B. Dusch, M. Bhattacharya, D. Jansen, HS Offenburg

**23 Konzeptstudie eines durchgängig auf parametrisierten Modulgeneratoren basierenden Entwurfsflusses für Analogdesign**

M. Greif, D. Marolt, J. Scheible, Robert Bosch Zentrum für Leistungselektronik, Reutlingen

**31 Hardware to Software Migration and Optimization of a Serial Peripheral Interface**

M. Freier, A. Wenzler, T. Mayer, Robert Bosch GmbH  
K. Schulz, J. Gerlach, HS Albstadt  
W. Rosenstiel, Uni Tübingen

**39 A Flexible Approach on FPGA-based Digital Sensor Interfacing for Cyber Physical Systems**

I. Verdu, K. F. Ackermann, HS Mannheim



Cooperating Organisation  
Solid-State Circuit Society Chapter  
IEEE German Section



# Inhaltsverzeichnis

<b>ASICs für medizinische Geräte und Implantate</b> .....	1
M. Nawito, H. Richter, C. Scherjon, J. N. Burghartz, IMS Chips Stuttgart	
<b>Low-Power-SC-Wandler mit hoher variabler Eingangsspannung</b> .....	7
D. Lutz, P. Renz, B. Wicht, Robert Bosch Zentrum für Leistungselektronik, Reutlingen	
<b>Entwicklung und Layoutentwurf eines Analog-Digital-Wandlers mit 12 Bit Auflösung in einer 180 nm-CMOS-Technologie</b> .....	13
B. Dusch, M. Bhattacharya, D. Jansen, HS Offenburg	
<b>Konzeptstudie eines durchgängig auf parametrisierten Modulgeneratoren basierenden Entwurfsflusses für Analogdesign</b> .....	23
M. Greif, D. Marolt, J. Scheible, Robert Bosch Zentrum für Leistungselektronik, Reutlingen	
<b>Hardware to Software Migration and Optimization of a Serial Peripheral Interface</b> .....	31
M. Freier, A. Wenzler, T. Mayer, Robert Bosch GmbH, K. Schulz, J. Gerlach, HS Albstadt, W. Rosenstiel, Uni Tübingen	
<b>A Flexible Approach on FPGA-based Digital Sensor Interfacing for Cyber Physical Systems</b> .....	39
I. Verdu, K. F. Ackermann, HS Mannheim	
<b>Gefertigte ASICs</b>	
<b>MMIC-Phasenschieber für den Einsatz in phasengesteuerten Arrayantennen im C-Band</b> .....	47
D. D. Aschenbrenner, C. Schick, HS Konstanz	
<b>Operationsverstärker, Bandgap-Referenz, Quarzoszillator und PLL</b> .....	49
M. C. de Andrade, A. Arnold, G. Bratek, C. Eschenbach, N. Koetsier, G. R. Tchouta, B. Vettermann, J. Giehl, HS Mannheim	
<b>Operationsverstärker und On-Chip-Tiefpassfilter 3. und 8. Ordnung</b> .....	51
A. Arnold, F. Bohinsky, I. Ohm, P. K. Fansi, N. Koetsier, B. Vettermann, J. Giehl, HS Mannheim	
<b>Galvanisch getrennter High-Side-Gatetreiber</b> .....	53
A. Seidel, M. Costa, J. Joos, B. Wicht, Robert Bosch Zentrum für Leistungselektronik, Reutlingen	
<b>Hocheffizienter resonanter Spannungswandler bis 30 V mit Maximumsdetektion &lt; 1 ns Auflösung und 50 V High-Speed-Levelshifter</b> .....	55
T. Funk, J. Wittmann, B. Wicht, Robert Bosch Zentrum für Leistungselektronik, Reutlingen	
<b>Synchroner 12 V 10 MHz Abwärtswandler mit Totzeitregelung in 125 ps Auflösung</b> .....	57
J. Wittmann, B. Wicht, Robert Bosch Zentrum für Leistungselektronik, Reutlingen	
<b>Hoch-Volt-Interface für 230 V-Netzbetrieb von ICs</b> .....	59
D. Lutz, B. Wicht, Robert Bosch Zentrum für Leistungselektronik, Reutlingen	
<b>Ultra-Low-Power-Verstärker-ASIC mit Energy Harvesting</b> .....	61
B. Steiner, G. Forster, HS Ulm	

**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

Mitherausgeber (Peer Reviewer):

Jürgen Giehl, Hochschule Mannheim, Paul-Wittsack-Straße 10, D-68163 Mannheim

Frank Kesel, Hochschule Pforzheim, Tiefenbronner Straße 65, D-75175 Pforzheim

Axel Sikora, Hochschule Offenburg, Badstraße 24, D-77652 Offenburg

Alle Rechte vorbehalten

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

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

# ASICs für medizinische Geräte und Implantate

Moustafa Nawito, Harald Richter, Cor Scherjon, Joachim N. Burghartz

**Zusammenfassung**—Aufgrund der Anforderungen an Miniaturisierung und Energieeffizienz werden auch in medizinischen Geräten trotz oft kleiner Stückzahl verstärkt anwendungsspezifische Integrierte Schaltungen (ASICs) eingesetzt. Ein besonderer Schwerpunkt liegt dabei auf mobilen oder sogar implantierten Anwendungen. IMS CHIPS engagiert sich seit langer Zeit auf diesem Gebiet. Im Vortrag wird ein Überblick über aktuelle Einsatz- und Forschungsgebiete gegeben und es werden Aktivitäten an Hand von Beispielen erläutert. Dabei wird sowohl auf die speziellen Anforderungen als auch auf die technische Implementierung eingegangen.

**Schlüsselwörter**—IMS CHIPS, Intelligente Implantate, TENECOR, SMARTImplant, fbEEG.

## I. EINLEITUNG

Hohe Kosten, langwierige Entwicklungszyklen, komplexe Herstellungsverfahren und aufwändige Tests machen die Produktion von ASICs in kleinen Stückzahlen zu einem höchst aufwändigen Produkt. Auch die Tatsache, dass Feldprogrammierbare Gate-Arrays (FPGAs), Mikrocontroller und Standardkomponenten aus dem Regal eine Vielzahl von Funktionen und Mixed-Signal-Verarbeitungsoptionen anbieten, schränkt die Nachfrage nach ASICs ein.

In der Medizintechnik jedoch werden ASICs, trotz kleiner Stückzahl, mehr und mehr eingesetzt. Aufgrund der oft extrem hohen Anforderungen an Miniaturisierung, geringen Energieverbrauch und Integration von analoger Signalvorverarbeitung sowie digitaler Nachverarbeitung müssen für medizinische Geräte, vor allem für mobile und implantierbare Anwendungen, sehr spezielle Lösungen entwickelt werden.

Das Institut für Mikroelektronik Stuttgart - IMS CHIPS - engagiert sich seit langem auf diesem Gebiet; ein Beispiel ist der kürzlich CE-zertifizierte Retina-Implantat-Chip [1]. In dieser Arbeit wird ein Überblick über aktuelle Einsatz- und Forschungsgebiete gegeben. Speziell werden die Aktivitäten von IMS

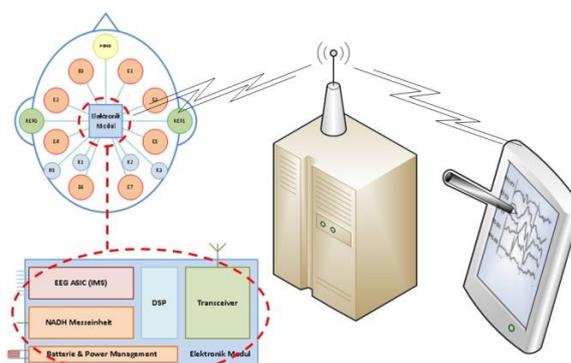


Abbildung 1: TENECOR Gesamtsystem.

CHIPS an Hand zweier Projekte, TENECOR [2] und SMARTImplant [3], erläutert. Dabei wird nicht auf jede einzelne Schaltung in den ASICs eingegangen, sondern auf bestimmte Systementwicklungsmethoden und Schaltungsentwurfsmaßnahmen, die es ermöglichen, spezielle Anforderungen, insbesondere Energieeffizienz, Flexibilität und Programmierbarkeit, Zuverlässigkeit und Signalqualität, zu erfüllen.

## II. TENECOR

TENECOR steht für Telemetrisch- multimodales neonatales Cortexmonitoring. Ziel des Projekts ist es, ein Monitoringsystem zur direkten Überwachung des Gehirns von Frühgeborenen zu entwickeln. In Deutschland gibt es jährlich rund 63.000 Frühgeborene. Eine besondere Gefahr für die Frühgeborenen geht von noch unreifen Nervenzellen aus, welche die Atmung und den Kreislauf steuern. Es besteht deshalb Bedarf an einem Überwachungssystem, das simultan aussagekräftige Parameter der Aktivität und der Versorgung der Zellen des zentralen Nervensystems misst. Dabei sollen mittels eines auf der Kopfhaut des Frühgeborenen befestigten Kombinationssensors, einer Art Messkappe, mehrere Messungen kontinuierlich durchgeführt werden, nämlich:

- Erfassung des NADH (reduziertes Nicotinamid-adenin-dinucleotid) Enzyms, um den zellulären Energiestoffwechsel zu bewerten.
- Breitbandige oder sogenannte „full band“ Elektroenzephalogramme (fbEEG) zur Bewertung der neuronalen Aktivität.

- DC-EEG zur Bewertung der neuronalen Vernetzung und
- Impedanzmessung zum Nachweis von Diffusionsstörungen (Ödeme).

Um dem medizinischen Personal die Möglichkeit zu geben, den Zustand der Hirnentwicklung des Frühgeborenen in Echtzeit zu überwachen, findet eine drahtlose Übertragung der von dem Elektronikmodul gemessenen Daten an das Überwachungsterminal statt. Abbildung 1 stellt das Gesamtsystem dar.

### A. ASIC-Struktur

Die Struktur des IMS ASICs ist in Abbildung 2 gezeigt. Jeder der implementierten 8 differentiellen EEG-Kanäle, die an den entsprechenden Elektroden angeschlossen werden, beinhaltet Schaltungsblöcke für fbEEG und DC-EEG-Messung. Die Impedanzmessseinheit führt eine 4-Punkt-Impedanzmessung durch. Um den gesamten Energieverbrauch zu reduzieren, sind zentrale Blöcke wie der Analog-zu-Digital-Wandler (ADC) nur einmal realisiert. Die analogen EEG und Impedanz-Messwerte werden sequentiell mittels eines Zeitmultiplexverfahrens vom ADC digitalisiert. Digitale Werte werden über eine serielle 4-Draht-SPI-Schnittstelle an die externe Auswertelektronik (AE) gesendet. Das ASIC ist so entworfen, dass interne Parameter wie Verstärkungspegel und Messbereiche von der AE konfigurierbar sind. Die AE gibt auch Takt- und Reset-Signale vor und kann den internen Zustand des ASICs messen. Das ASIC bietet zusätzlich analoge Ausgänge zu Testzwecken.

### B. Qualität der Signalbearbeitung

Trotz der geforderten aufwändigen Funktionalität des ASICs besteht die größte Herausforderung in dem Verhältnis der zu messenden EEG-Signale, die im Bereich  $0.5 \dots 200 \mu\text{V}$  liegen, zu den durch die EEG-Elektroden erzeugten Offsetspannungen, die ca.  $500 \text{ mV}$  betragen. Dies stellt extrem hohe Anforderungen an die analogen fbEEG-Schaltungsblöcke, besonders hinsichtlich rauscharmer Funktion und Offsetunterdrückung. Beispielhaft soll hier auf den Entwurf des Eingangverstärkers eingegangen werden. Als erster analoger Schaltungsblock bearbeitet er die von den EEG-Elektroden abgetasteten Signale. Das bedeutet, dass die Integrität dieser Schaltung für die Leistung des Gesamtsystems von entscheidender Bedeutung ist.

Um rauscharme Funktion und geringere Offsetspannung zu erreichen, wurde von der Schaltung in Abbildung 3 ausgegangen. Diese besteht im Kern aus einem einfachen zweistufigen Miller-kompensierten Transkonduktanzverstärker (OTA) mit kapazitiver Rückkopplung. Die einzelnen Verstärkerstufen  $G_{m1}$  und  $G_{m2}$  sind für eine rauscharme Funktion entworfen, wobei die Versorgungswiderstände  $R_b$  als MOSFETs realisiert sind, um thermisches Rauschen zu reduzieren und um große Widerstandswerte zu realisieren. Der DC-Offset wird durch die Eingangskondensatoren

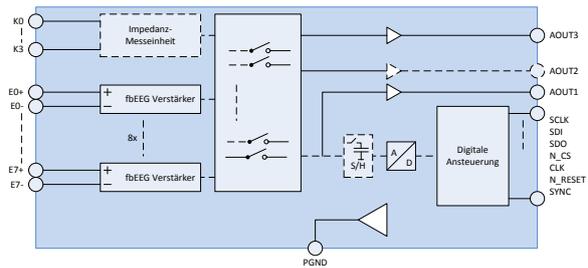


Abbildung 2: IMS ASIC-Struktur.

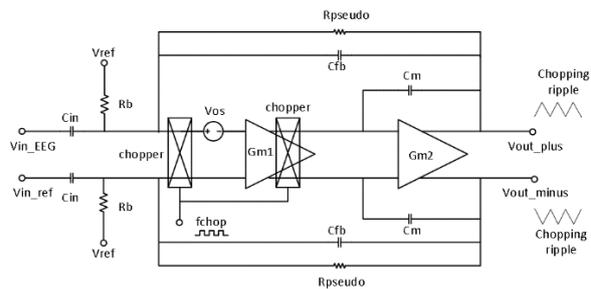


Abbildung 3: Front-end-Verstärker mit Chopping.

$C_{in}$  blockiert, und Chopping wird benutzt, um den internen Offset von  $G_{m1}$  zu eliminieren. Dabei werden die Eingangssignale zuerst auf die Chopping-Frequenz moduliert und anschließend zur ursprünglichen Signalfrequenz demoduliert, während der Offset nur auf die Chopping-Frequenz moduliert ist. Da die Chopping-Frequenz viel größer als die Signalfrequenz gewählt ist, wird die modulierte Offsetspannung durch Tiefpassfilterung entfernt.

Da aber Chopping zu Signalwelligkeit führt, muss die Schaltung erweitert. Abbildung 4 zeigt den Basisverstärker von Abbildung 3 mit sogenannter „ripple reduction“, d.h. Reduzierung der Signalwelligkeit. Diese wird erreicht durch eine Rückkopplungsschleife, wobei der Kondensator  $C_s$  das Ausgangssignal in einen Wechselstrom umwandelt. Weiteres Chopping produziert dann einen DC-Strom. Der Integrator ( $G_3$  und  $C_{int}$ ) wandelt den DC-Strom in eine Kompensations- oder Korrekturspannung, die mittels  $G_{M4}$  als Strom an den ursprünglichen Verstärker zurückgekoppelt wird. Nach mehreren Zyklen werden die Signalwelligkeit sowie die Offsetspannung von  $G_{m1}$  eliminiert.

Wie auch in Abbildung 4 dargestellt, wird „Impedanz Boosting“ oder eine Impedanzsteigerungsschleife eingesetzt, um die Eingangsimpedanz des Verstärkers, die hauptsächlich durch  $C_{in}$  bestimmt ist, zu erhöhen. Die Impedanzsteigerungsschleife ist eine positive Rückkopplungsschleife und besteht aus einem AC-gekoppelten Verstärker ( $G_{m5}$ ) und abstimmbaren Kondensatoren  $C_{tune}$ . Die Wechselspannungsausgangssignale des Kernverstärkers werden zu Strömen über  $C_{tune}$  umgewandelt. Die Schleife liefert dann Ströme proportional zu den Eingangsspannungen des

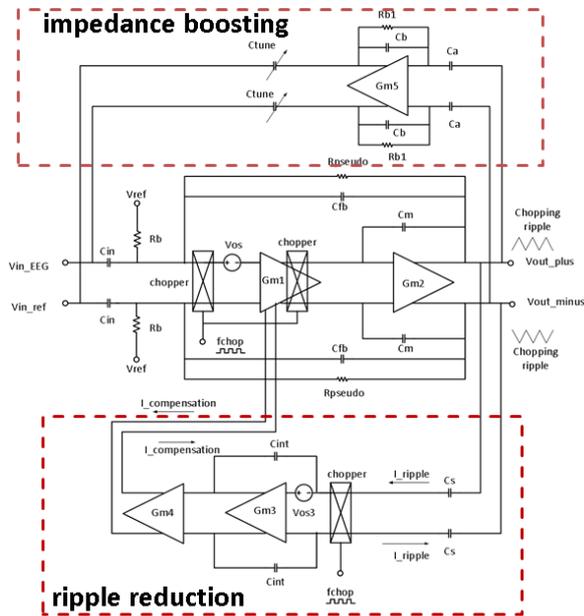


Abbildung 4: Front-end-Verstärker mit Chopping, Ripple Reduction und Impedance Boosting.

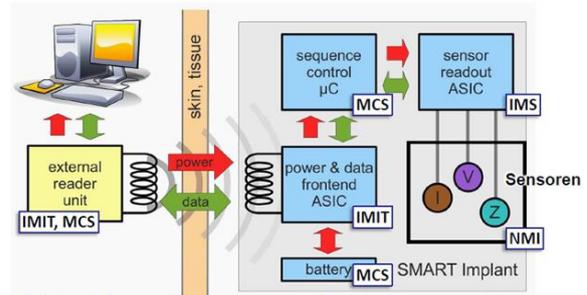
Kernverstärkers und reduziert daher den von den Elektroden empfangenen Strom, was zu einer Steigerung der Eingangsimpedanz führt.

Die Realisierung dieses Konzeptes erfordert eine sehr sorgfältige Untersuchung der Stabilität des Systems. Da mehrere negative und positive Rückkopplungsschleifen in Kombination eingesetzt sind, ist es notwendig, ausführliche Systemsimulationen (z.B. mit MATLAB®) im Designprozess durchzuführen und die Entwicklung nicht nur auf Schaltungsentwurfsebene zu beschränken.

### III. SMARTIMPLANT

Das Design von ASICs für medizinische Implantate stellt besondere Herausforderungen. Obwohl die Signalbearbeitungsqualität eine wichtige Anforderung ist, stehen beim Entwurf von ASICs für Implantate andere Erfordernisse im Vordergrund.

Ein wichtiger Bestandteil aller Arten aktiver medizinischer Implantate, z.B. Hirnimplantate, ist das Auslesechip. Dieses ist für die Stimulation der Messelektroden, Datenerfassung und die Steuerung des Messzyklus zuständig. Für Anwendungen, bei denen große Batterien und die Verkabelung nicht geeignet sind, werden hohe Anforderungen an den Auslesechip in Bezug auf Größe und Energieeffizienz gelegt. Im Rahmen des Projektes SMARTImplant wurde bei IMS CHIPS im Rahmen des Spitzenclusters MicroTEC Südwest eine Familie von ASICs entwickelt, die als Auslesechip von hochintegrierten subkutanen Implantaten zum Einsatz kommen. Dabei werden amperometrische Messungen der Sauerstoffkonzentration,



(a)



(b)

Abbildung 5 (a) SMARTImplant Gesamtsystem, (b) Foto vom Implantat (©NMI).

potentiometrische Messungen des pH-Wertes, Impedanzspektroskopie am umgebenden Gewebe sowie Temperaturmessungen durchgeführt.

#### A. Gesamtsystem und ASIC-Struktur

Wie in Abbildung 5(a) gezeigt, enthält das System den Auslese-ASIC von IMS CHIPS, der die Messelektroden mit dem zu charakterisierenden Gewebe direkt verbindet, stimuliert und ausliest. Ein Mikrocontroller sendet Befehle an den Auslesechip, erhält die Ergebnisse und leitet sie an den Spannungs- und Datenmanagement-ASIC. Der Frontend-Chip ist für die Regulierung der Stromversorgung des Implantats und zur Übertragung von Daten über eine induktive Schnittstelle zu einer externen Leseinheit verantwortlich. Abbildung 5(b) zeigt ein Foto des Implantats, das als eine hermetisch abgedichtete starre Kapsel aus PEEK realisiert ist.

Die Struktur des ASIC ist in Abbildung 6 dargestellt. Die SPI-Schnittstelle empfängt 16-Bit-Messbefehle vom Mikrocontroller und liefert auch 16-Bit-Messergebnisse. Eine digitale Steuerung kontrolliert die analogen Messschaltungen, die die Elektroden treiben. Ein zentralerer 10-Bit-ADC ist dafür zuständig, die gemessenen analogen Werte zu digitalisieren, wobei andere Schaltungsblöcke wie Integratoren, ein 5-Bit-Digital-zu-Analog-Wandler (DAC) und Verstärker während der verschiedenen Messungen eingesetzt werden.

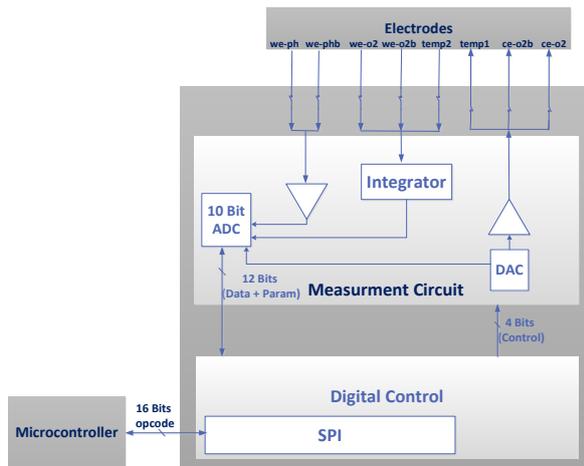


Abbildung 6: Struktur des IMS Auslesechips.

### B. Programmierbarkeit und Flexibilität

Da es eine wesentliche Voraussetzung des Implantats ist, verschiedene Arten von Geweben und Materialien charakterisieren und messen zu können, muss der Auslesechip entsprechend entworfen sein, um einen flexiblen und zuverlässigen Betrieb zu ermöglichen. Zu diesem Zweck wurde ein parametrisierbarer Messablauf implementiert, in dem alle Messungen mit variablen Dauern, Parametern und Einstellungen durchgeführt werden können.

Eine weitere Herausforderung, die insbesondere beim Design von Implantaten für in-vivo-Biosensoren auftritt, ist die Tatsache, dass es keinen direkten Zugriff zur Messumgebung gibt, oder einfach gesagt, es wird „im Dunkeln“ gemessen. Aus diesem Grund ist die Kalibrierung und Voreinstellung des ASICs zwingend notwendig, um eine sinnvolle Interpretation der Ergebnisse zu ermöglichen.

Im Betrieb empfängt der Chip einen 16-Bit-Operationscode oder Opcode, wobei die ersten 4 Bits die Anweisung darstellen (messen, kalibrieren etc.) und die relevanten Parameter in den übrigen 12 Bits verpackt sind. Als ein Beispiel wird in Abbildung 7 der „wakeup“ oder Aufweck-Befehl mit den entsprechenden Parametern dargestellt.

### C. Energieeffizienz und kompaktes Design

Wie erwähnt, ist der 10-Bit-Pipeline-ADC ein zentraler Teil des Auslesepfads. Da aber eine typische Struktur für einen solchen ADC 9 Wandlungsstufen in Serie beinhaltet, wobei jede Stufe 1,5 Bits produziert, wurde ein modifiziertes Pipeline-ähnliches Design verwendet, das Chipfläche und Energie spart. Wie in Abbildung 8 gezeigt, besteht der On-Chip-ADC aus einer Wandlungsstufe, gefolgt von einer Abtast-Halteschaltung. Das analoge Eingangssignal wird einmal gewandelt und die ersten Bits werden erstellt, wonach das Residuum zurück zum Eingang geführt wird, um weitere Bits zu erzeugen. Dieser zyklische

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
wakup	pH, O <sub>2</sub> or T			channel	use bandgap	current range		-							

Abbildung 7: Wakeup Opcode.

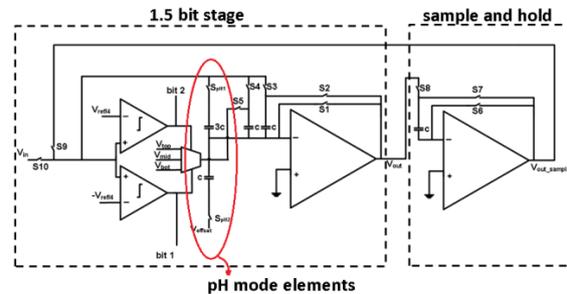


Abbildung 8: ADC-Schaltung mit speziellem pH-Modus.

Wandlungsansatz ermöglicht eine 10-Bit-Funktion und ist gleichzeitig fast zehnmal flächensparender und energieeffizienter als eine typische Implementierung.

Eine weitere Entwicklung des ADCs war erforderlich, um die Auflösung auf 12 Bit zu erhöhen, da die 10-Bit-Ergebnisse, die für O<sub>2</sub>, Temperatur und Impedanz-Messungen ausreichend sind, nicht die Anforderung der pH-Messung erfüllen. Es wurde deshalb ein Offset-Subtraktionsverfahren eingesetzt, das den pH-Messwert zuerst „grob“ in ein 10-Bit-Wort wandelt, einen bekannten Offset-Wert vom Eingang subtrahiert und die Differenz mit dem Faktor 4 multipliziert und nochmals in ein „feines“ 10-Bit-Wort umgesetzt. Das endgültige 12-Bit-Resultat ist die Summe von Offset und „Fein“-Antwort. Wie in Abbildung 8 gezeigt, wurde dieses Wandlungsprinzip mittels Schalter Sph1 und Sph2 und Kondensatoren mit den Werten C und 3C realisiert. Da durch die Wiederverwendung kein zusätzlicher ADC notwendig ist, wurden die Schaltungskomplexität und der Energieverbrauch reduziert.

Eine weitere Designmaßnahme zur Senkung des Stromverbrauchs ist das Ausschalten aller Schaltungsblöcke in den ungenutzten Phasen. Auch die Taktfrequenz der digitalen Schaltung wurde gewählt, um die angestrebte Datenrate zu erreichen und gleichzeitig den dynamischen Energieverbrauch zu reduzieren.

### D. Fertigung und Zuverlässigkeit

Das Design wurde mit Hilfe des GFQ<sup>®</sup> Prozesses, einer 0,5 µm CMOS-Technologie mit 2 Metalllagen, bei IMS CHIPS implementiert. Diese „sea-of-transistors“-Technologie bietet Transistoren und passive Elemente in bestimmten festen Dimensionen an, wobei die digitalen und analogen Schaltungen durch die Verbindung dieser Elemente mit Metallbahnen realisiert werden. Ein Hauptvorteil dieses „semi-custom“-Ansatzes ist, dass die Schaltungsbaulemente sehr gut charakterisiert und modelliert sind, was die Fertigungsausbeute und Zuverlässigkeit der ASICs verbessert. Abbildung 9 zeigt ein Foto des gefertigten

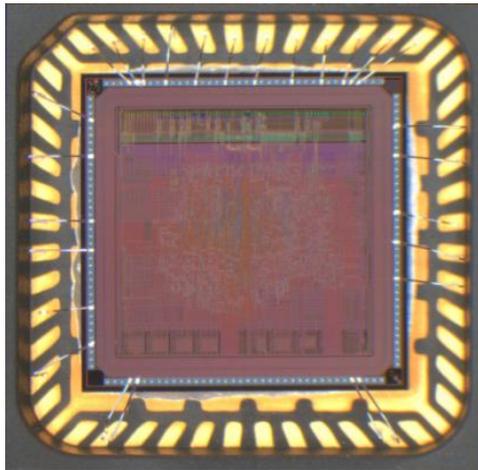


Abbildung 9: Foto vom gefertigten ASIC im Gehäuse.

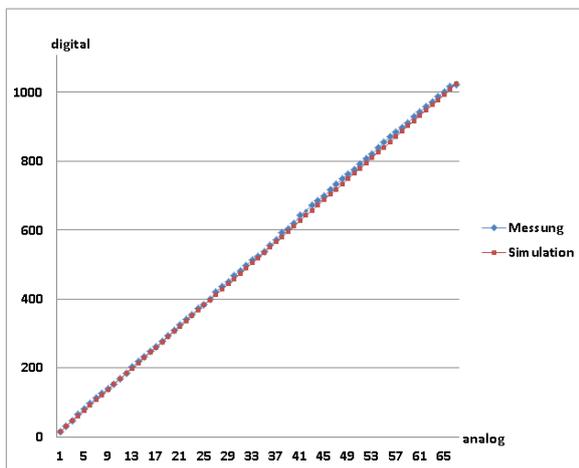


Abbildung 10: Gemessenes und simuliertes Verhalten des On-Chip-ADCs.

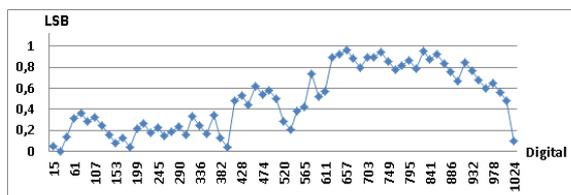


Abbildung 11: DNL-Messung des On-Chip-ADCs.

ASICs, aufgebaut in einem QFN48-Gehäuse. Die gute Übereinstimmung der Messergebnisse mit den Simulationen des On-Chip-ADCs ist in Abbildung 10 gezeigt. Abbildung 11 zeigt auch die gemessene Differenzielle Nichtlinearität (DNL) des ADCs, die wie gewünscht unter 1 LSB liegt.

#### IV. FAZIT

Die erfolgreiche Konzeption und Umsetzung von programmierbaren ASICs für Medizintechnik für den Einsatz in aktiven Miniaturimplantaten mit hoher Funktionalität ist ein interessantes und vielseitiges Unterfangen. Die Schaltungsentwicklung erfolgt unter Berücksichtigung des Energiebedarfs und des flexiblen Einsatzes mit hoher Signalqualität nicht nur auf Schaltungsebene, sondern auch für das Gesamtsystem. Mit dem Einsatz von ASICs können sehr spezielle Funktionen implementiert werden, die die Qualität der medizinischen Prozesse deutlich erhöhen und verbessern.

#### DANKSAGUNG

TENECOR wird im Förderprogramm „KMU-innovativ: Medizintechnik“ vom Bundesministerium für Bildung und Forschung (BMBF) unter der Projektnummer 13GW0033E gefördert. SMARTImplant ist ein Projekt der SSI Plattform im microTEC Südwest Spitzencluster und wird vom BMBF unter der Projektnummer 16SV5979K gefördert.

Die Autoren bedanken sich herzlich bei allen beteiligten Projektpartnern und den IMS CHIPS Kollegen, ohne deren Unterstützung die vorgestellte Arbeit nicht möglich gewesen wäre.

#### LITERATURVERZEICHNIS

- [1] J. N. Burghartz et al., "CMOS Imager Technologies for Biomedical Applications", in *Dig. of Tech. Papers ISSCC 2008*, San Francisco, CA, pp. 142-143.
- [2] [https://www.medizintechnologie.de/fileadmin/pdfs/projektsteckbriefe/13GW0033\\_Tenecor-Projektsteckbrief.pdf](https://www.medizintechnologie.de/fileadmin/pdfs/projektsteckbriefe/13GW0033_Tenecor-Projektsteckbrief.pdf).
- [3] A. Stett et al. "SMART Implant: Electronic Implants for Diagnosis and Monitoring", *Proceedings of 7th GMM Workshop on Energy self-sufficient Sensors*, Magdeburg, Germany, February 2014.



Moustafa Nawito erhielt den B.Sc. in Elektronik und Nachrichtentechnik von der Kairo Universität in 2002 und den M.Sc. in Elektronik von der German University in Cairo (GUC) in 2008. Seit 2010 ist er am Institut für Mikroelektronik in Stuttgart (IMS-Chips) als Analog und Mixed-Signal-Chip-Designer tätig, wobei er gleichzeitig an der Universität Stuttgart promoviert



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.



Cor Scherjon erhielt das M.Sc. Diplom in 1997 von der Universität Twente, Niederlande. Er leitet die Abteilung ASIC-Entwicklung am Institut für Mikroelektronik Stuttgart. Er hat mitgewirkt in verschiedenen nationalen und internationalen Projekten im Bereich Embedded Systems, hochdynamischen Bildsensoren und Entwicklung von Mixed-Signal ASICs.



Joachim N. Burghartz ist seit 2005 Direktor des Instituts für Mikroelektronik für Mikroelektronik Stuttgart und Professor für Mikroelektronik an der Universität Stuttgart. Er erwarb den Titel Dipl.-Ing. von der RWTH Aachen in 1982 und den Dr.-Ing. an der Universität Stuttgart in 1987. Bevor er nach Stuttgart kam, arbeitete er für 11 Jahre bei IBM Research in den USA und für 8 Jahre als Professor und Wissenschaftlicher Direktor des Instituts DIMES an der TU Delft in den Niederlanden.

# Low-Power-SC-Wandler mit hoher variabler Eingangsspannung

Daniel Lutz, Peter Renz, Bernhard Wicht

**Zusammenfassung**—Die Nachfrage nach kompakten Spannungsversorgungen ist in den letzten Jahren stark gestiegen. Vor allem im Bereich der mobilen Geräte wachsen die Anforderungen an die Spannungsversorgung hinsichtlich Bauvolumen und Batterielaufzeit. Für die Vollintegration von DC-DC-Wandlern als „Power Supply on Chip“ ist der SC-Wandler (Switched-Capacitor-Wandler) besonders geeignet. Insbesondere für Low-Power-Anwendungen im Bereich 10 mW kann ein SC-Wandler sehr gut, ohne externe Bauelemente, integriert werden. Während es für niedrige Eingangsspannungen (bis zu 5 V) eine Vielzahl an Topologien und Konzepten gibt, wurden SC-Wandler für höhere Eingangsspannungen (> 8 V) bisher nur wenig untersucht. Dieser Beitrag untersucht die wichtigsten Grundlagen für SC-Wandler mit Schwerpunkt auf hoher und zugleich variabler Eingangsspannung im Bereich 5 - 20 V. Am Beispiel eines Multi-Ratio-Wandlers (Wandler mit mehreren Übersetzungsverhältnissen), dem rekursiven SC-Wandler (RSC-Wandler), werden die Anforderungen eines SC-Wandlers für hohe Eingangsspannungen herausgearbeitet und diskutiert.

**Schlüsselwörter**—Low-Power, DC-DC-Wandler, SC-Wandler, rekursiver SC-Wandler, großer Eingangsspannungsbereich, variable Eingangsspannung.

## I. EINLEITUNG

Der Bedarf an kompakten Spannungsversorgungen ist in den letzten Jahren stark gestiegen. Verbraucher wie z.B. ein Mikrocontroller oder ASIC (Application-specific Integrated Circuit) weisen dabei einen sehr geringen Leistungsverbrauch auf (wenige Milliwatt). Für die Versorgung dieser Low-Power-Schaltungen können verschiedene Energiequellen wie Batterien, Energy-Harvesting-Techniken oder die Netzspannung verwendet werden. Die Bereitstellung der nötigen

Daniel Lutz, daniel.lutz@reutlingen-university.de, Peter Renz, peter.renz@student.reutlingen-university.de und Bernhard Wicht bernhard.wicht@reutlingen-university.de sind Mitglieder der Hochschule Reutlingen, Alteburgstraße 150, 72762 Reutlingen.

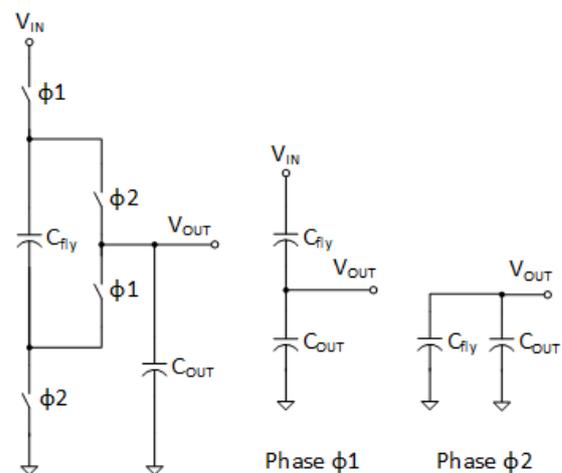


Abbildung 1: 2:1 SC-Serien-Parallel-Wandler.

Energie spielt eine zunehmend wichtig werdende Rolle. Für die Versorgung von integrierten Schaltkreisen (ICs) werden Netzteile benötigt, welche die Versorgungsspannung auf eine der Schaltung angepasste Eingangsspannung reduzieren. Diese sind für Low-Power Anwendungen oft überdimensioniert. Deshalb wird versucht, die Spannungsversorgung direkt auf dem Chip zu implementieren, um Platz und somit Kosten einzusparen. Dieser Ansatz wird unter dem Begriff „Power Supply on Chip“ (PwrSoC) zusammengefasst. Durch die Vollintegration sind keine externen Bauelemente nötig und es bietet sich die Möglichkeit, Spannungsversorgung und weitere Funktionsblöcke in einem IC zu realisieren.

Für die Vollintegration eignen sich kapazitive SC-Wandler besonders gut. Diese bestehen prinzipiell nur aus Kapazitäten und Schaltern, welche sich beide gut auf dem Chip integrieren lassen. Die Integration von Induktivitäten ist wesentlich schwerer und meist nur mit aufwendigen Prozessoptionen oder durch Postprozesse möglich. Dadurch ist der Einsatz in vollintegrierten DC-DC-Wandlern nur bedingt möglich.

Der kapazitive Wandler besitzt ein diskretes Übersetzungsverhältnis und kein einstellbares Übersetzungsverhältnis wie der induktive Wandler. Dies ist ein Nachteil des kapazitiven Wandlers, der dadurch gelöst werden kann, dass mehrere Topologien in einem

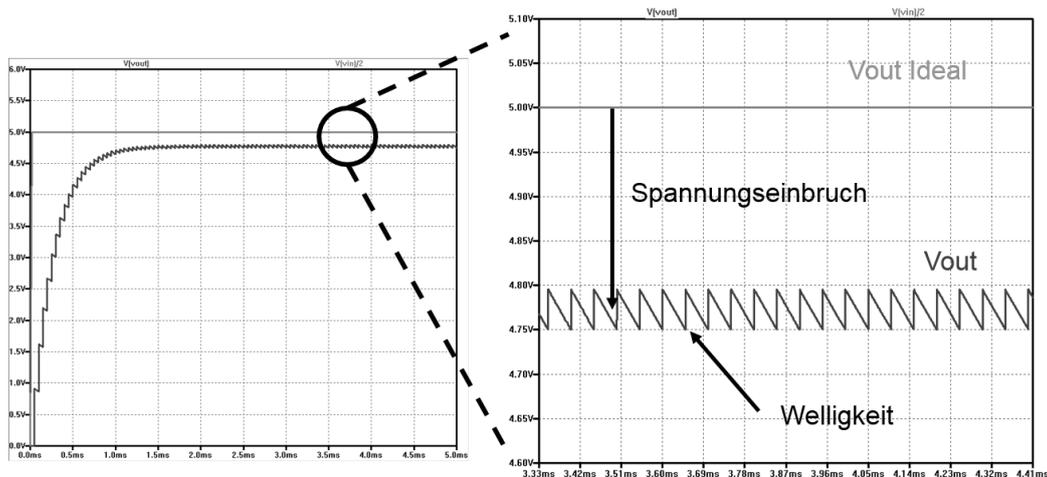


Abbildung 2: Spannungseinbruch und Welligkeit der Ausgangsspannung eines 2:1-Serien-Parallel-Wandlers.

Wandler realisiert werden. Sogenannte Multi-Ratio-Wandler bilden die Voraussetzung dafür, einen großen variablen Eingangsspannungsbereich zu unterstützen. Dieses Thema wird nachfolgend behandelt.

## II. GRUNDLAGEN SC-WANDLER

Ein kapazitiver DC-DC-Wandler besteht prinzipiell aus zwei verschiedenen Kapazitätstypen: Fliegende Kapazitäten (Potential an der Kapazität springt) und Pufferkapazitäten (Potential an Kapazität ist fest). Die fliegenden Kapazitäten sind für den Ladungstransport zuständig, während die Pufferkapazitäten für die Stabilisierung und Zwischenspeicherung der Ausgangsspannung verantwortlich sind. Das Übersetzungsverhältnis bei einem SC-Wandler wird durch die Verschaltung der fliegenden Kapazitäten (Topologie) bestimmt. Es gibt verschiedene Grundschaltungen (Topologien) aus denen SC-Wandler aufgebaut sind. Einen Überblick und Vergleich gibt hierbei Referenz [1].

### A. Analyse von SC-Wandlern

Das Funktionsprinzip eines SC-Wandlers wird mit Hilfe der Abbildung 1 anhand eines einfachen 2:1-Serien-Parallel-Wandlers erläutert. Die Schalter werden gegenphasig angesteuert und sind in der Abbildung mit  $\phi 1$  und  $\phi 2$  gekennzeichnet. In der ersten Hälfte der Taktperiode werden die Schalter  $\phi 1$  eingeschaltet, während die Schalter  $\phi 2$  ausgeschaltet sind. In der zweiten Taktperiode leiten die Schalter  $\phi 2$ , während die Schalter  $\phi 1$  sperren. Das Tastverhältnis der beiden Phasen beträgt hierbei 50 %. Um das Auftreten von Querströmen zu vermeiden muss eine kurze Totzeit zwischen den Phasen eingehalten werden.

Die Energieübertragung zwischen Eingang und Ausgang erfolgt durch die fliegende Kapazität  $C_{fly}$ . In Phase  $\phi 1$  werden  $C_{fly}$  und  $C_{out}$ , in Reihe zur Eingangsspannung  $V_{in}$  geladen. In Phase  $\phi 2$  wird  $C_{fly}$  parallel zur Ausgangskapazität  $C_{out}$  geschaltet. In beiden Phasen wird

Ladung zum Ausgang transportiert. Durch Ladungsausgleich sind im eingeschwungenen Zustand, beide Kapazitäten  $C_{fly}$  und  $C_{out}$  auf den gleichen Spannungswert geladen. Die Ausgangsspannung  $V_{out}$  entspricht im Idealfall und ohne Last exakt der halben Eingangsspannung  $V_{in}$ . Dadurch entsteht eine 2:1-Wandlung, unabhängig von der Dimensionierung der Kapazitäten. Im Falle eines Laststromes muss Ladung zwischen dem Eingang und dem Ausgang übertragen werden. Die Kapazitäten des Wandlers werden entladen und müssen wiederum geladen werden. Dies führt zu Lade- und Entladeverlusten sowie zu einem Einbruch und einer Welligkeit der Ausgangsspannung. In Abbildung 2 ist im linken Bild die Ausgangsspannung eines 2:1-SC-Serien-Parallel-Wandlers zu sehen. Die graue Linie zeigt den idealen Verlauf der Ausgangsspannung. Die schwarze Linie zeigt den realen Verlauf der Ausgangsspannung unter Last. Die Ausgangsspannung nimmt stufenweise zu und nähert sich einem Spannungswert unterhalb der idealen Ausgangsspannung an. Die bleibende Differenz zwischen idealer und realer Ausgangsspannung wird als Spannungseinbruch bezeichnet und ist im rechten Teil von Abbildung 2 vergrößert dargestellt. Dieser Spannungseinbruch hängt vom Laststrom am Ausgang, der Topologie und der Dimensionierung des SC-Wandlers ab. Bei der Dimensionierung spielt die Größe der fliegenden Kapazität und die Schaltfrequenz des Wandlers eine Rolle.

Die Ausgangsspannung lässt sich durch das Aufstellen der Ladungsbilanz berechnen, welche auf dem Gesetz der Ladungserhaltung beruht [2]. Diese Analyse-methode ist allerdings für komplexe Wandler sehr aufwendig, weshalb ein neuer Ansatz, die Ladungsflussanalyse verwendet wird, der in [1] hergeleitet und beschrieben wird. Dabei wird nicht die absolute Ladungsmenge betrachtet, sondern die Ladungsänderung an den Komponenten der Schaltung. Für jeden Zustand eines Wandlers können hierbei Vektoren definiert werden, welche die Topologie mithilfe des Ladungsflusses durch die Komponenten beschreibt. Wird diese

auf den in Abbildung 1 gezeigten 2:1-SC Serien-Parallel-Wandler angewendet, ergibt sich mit der Bedingung, dass die Ausgangskapazität  $C_{out}$  viel größer ist als die fliegende Kapazität  $C_{fly}$ , folgender Ausdruck für die Ausgangsspannung  $V_{out}$ :

$$V_{out} = \frac{1}{2} \cdot V_{in} - \frac{1}{4} \cdot \frac{I_{Load}}{f_{sw} \cdot C_{fly}}$$

In die Berechnung gehen die Parameter Ausgangsstrom  $I_{load}$ , die Schaltfrequenz  $f_{sw}$  und die Größe der fliegenden Kapazität  $C_{fly}$  ein. Zusätzlich kommt für den 2:1-SC-Serien-Parallel-Wandler noch der von der Topologie abhängige Faktor  $1/4$  dazu. Der Subtrahend in der Gleichung entspricht dem Spannungseinbruch. Die Welligkeit der Ausgangsspannung ist in Abbildung 2 ebenfalls zu sehen. Die Welligkeit lässt sich unter der Bedingung, dass  $C_{out}$  viel größer ist als  $C_{fly}$ , wie folgt berechnen:

$$\Delta V = \frac{1}{2} \cdot \frac{I_{Load}}{f_{sw} \cdot C_{out}}$$

Die Welligkeit hängt von der Größe des Ausgangsstroms, der Größe der Ausgangskapazität und der Schaltfrequenz ab. Der Faktor  $1/2$  ist topologieabhängig. Für den 2:1-SC-Wandler nach Abbildung 1 beträgt dieser Faktor  $1/2$ , da in beiden Phasen die Ausgangskapazität nachgeladen wird und dadurch die Welligkeit halbiert wird. Da  $C_{out}$  gerade bei vollintegrierten SC-Wandlern aus Platzgründen nicht beliebig groß gewählt werden kann, wird oft die Multi-Phase-Interleaving-Methode angewendet [3]. Dabei wird der Wandler mehrfach aufgebaut, der Gesamtkapazitätswert der fliegenden Kapazitäten bleibt jedoch gleich. Durch eine Phasenverschiebung der Taktsignale der einzelnen Wandler wird erreicht, dass auf die Periode verteilt öfters Ladung zum Ausgang transportiert wird. Dies führt oft dazu, dass die Ausgangskapazität stark reduziert oder sogar ganz entfallen kann.

### B. SC Modell

Ein SC-Wandler kann durch ein idealisiertes Modell entsprechend Abbildung 3 beschrieben werden. Das Modell besteht aus einem idealen DC-Transformator, welcher das entsprechende ideale Übersetzungsverhältnis  $1:n$  realisiert, einem Ausgangswiderstand  $R_{out}$  und einem Parallelwiderstand  $R_p$ . Im Falle eines Laststroms  $i_{out}$  entsteht ein Spannungsabfall am Ausgangswiderstand  $R_{load}$  welcher die ideale Ausgangsspannung  $V_{nl}$  (nicht belastet, engl. no load) reduziert und Verluste verursacht. Der Ausgangswiderstand  $R_{out}$ , welcher die Lade- und Entladeverluste der Kapazitäten sowie die resistiven Leitungsverluste berücksichtigt, kann mit der bereits erwähnten Ladungsflussanalyse aus [1] bestimmt werden. Der Ausgangswiderstand  $R_{out}$  ist hierbei frequenzabhängig und kann auch zur Feinregelung der Ausgangsspannung des Wandlers verwendet wer-

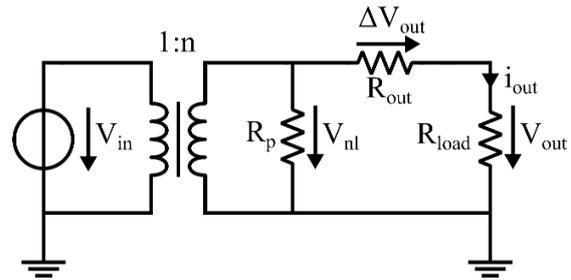


Abbildung 3: SC-Wandler-Modell.

den. Extrinsische Verluste wie beispielsweise die Ansteuerungsverluste der Schalttransistoren und Verluste durch die Regelung werden durch den Parallelwiderstand  $R_p$  modelliert.

Der Ausgangswiderstand  $R_{out}$ , welcher die maximale Ausgangsleistung begrenzt, setzt sich aus zwei asymptotischen Bereichen, dem Slow-Switching Limit (SSL) und dem Fast-Switching Limit (FSL) zusammen. Die Slow-Switching-Limit-Impedanz  $R_{SSL}$  wird unter der Annahme berechnet, dass alle Schalter und leitfähigen Verbindungen ideal und somit vernachlässigbar sind. Die Ströme zwischen den Kapazitäten verlaufen hierbei impulsförmig. Die Bezeichnung Slow-Switching Limit kann man damit erklären, dass die Zeitkonstante des RC-Glieds zwischen Schalter und Kapazität hierbei wesentlich kleiner als die Periodendauer der Schaltfrequenz ist. Die Kapazitäten werden somit vollständig umgeladen. Die SSL-Impedanz ist umgekehrt proportional zu der Schaltfrequenz. Das Fast-Switching-Limit tritt auf, wenn der Widerstand der Schalter und der Verbindungen dominiert. Die Lade- und Entladeverluste der Kapazitäten werden somit vernachlässigt. Die Bezeichnung Fast-Switching Limit lässt sich dadurch erklären, dass die Zeitkonstante zwischen Schalter und Kapazität größer als die Periodendauer der Schaltfrequenz ist und somit die Kapazitäten als feste Spannungsquellen angenommen werden können. Die Kapazitäten werden hierbei nicht vollständig geladen. Die Abbildung 4 zeigt exemplarisch den Verlauf des Ausgangswiderstands  $R_{out}$  in Abhängigkeit der Schaltfrequenz  $f_{sw}$ . Zudem sind die Asymptoten der SSL- und der FSL Impedanz eingezeichnet. Die SSL-Impedanz  $R_{SSL}$  nimmt mit steigender Frequenz ab. Das lässt sich dadurch begründen, dass mit steigender Schaltfrequenz  $f_{sw}$  der Spannungshub an den Kapazitäten kleiner und somit die Lade- und Entladeverluste geringer sind. Die FSL Impedanz ist frequenzunabhängig und zeigt einen konstanten Verlauf. Der Ausgangswiderstand  $R_{out}$  bildet sich bei kleinen Schaltfrequenzen maßgeblich durch  $R_{SSL}$  und bei hohen Frequenzen durch  $R_{FSL}$  und kann prinzipiell durch folgende Gleichung angenähert werden [4].

$$R_{out} = \sqrt{R_{SSL}^2 + R_{FSL}^2}$$

Tabelle 1: Überblick über implementierte SC-Wandler.

Referenz	$V_{in}$ [V]	$V_{out}$ [V]	Übersetzungsverhältnisse	Wirkungsgrad (max.)
[9]	1,2	0,3-1,1	5	70 %
[5]	2,5	0,1-2,18	15	85 %
[11]	2	0,5-1,2	3	79 %
[12]	2,4	1	1	69 %
[10]	1,8	0,9-1,65	5	89 %
[6]	2,8-8	1,2	4	77 %

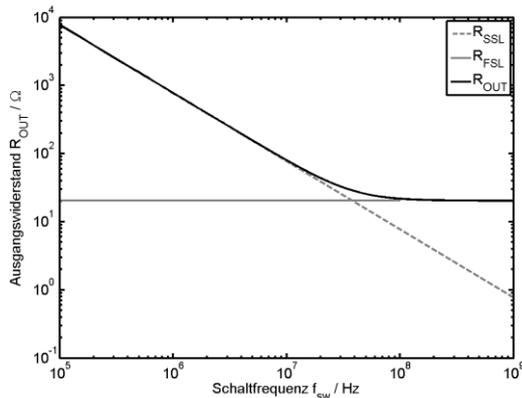


Abbildung 4: Ausgangswiderstand  $R_{out}$  eines SC-Wandlers mit der FSL- und SSL Asymptoten.

Die Gleichung stellt eine Näherung dar, da in der Herleitung der beiden Widerstände  $R_{SSL}$  und  $R_{FSL}$  entsprechend gegensätzliche Annahmen angenommen werden. In [4] wird gezeigt dass die Abweichung gering und somit vernachlässigbar ist.

### III. IMPLEMENTIERUNG VON SC-WANDLERN

#### A. Anforderungen

Eine der Hauptanforderungen an SC-Wandler ist der Flächenbedarf auf dem Chip, der stets minimal zu wählen ist. Dies hat zur Folge, dass SC-Wandler je nach Anforderung entworfen werden. Die Anforderungen an den SC-Wandler bestimmen, welche Topologie zu wählen ist oder ob mehrere Topologien gewählt werden müssen. Die Dimensionierung wird maßgeblich von den Ausgangsparametern bestimmt. In dieser Arbeit wird ein SC-Wandler benötigt, welcher einen großen Eingangsspannungsbereich am effizientesten auf eine konstante Ausgangsspannung wandelt.

#### B. Stand der Technik

Aufgrund des großen Eingangsspannungsbereichs ist eine hohe Anzahl an Wandlungsverhältnissen notwendig, um einen Betrieb mit hohem Wirkungsgrad zu gewährleisten. Das Erhöhen der Anzahl an Wandlungs-

verhältnissen ist mit zwei Herausforderungen verbunden: Kapazitätsnutzung und Flächenverbrauch. Besonders bei einem vollintegrierten SC-Wandler ist es wichtig, dass die Kapazitäten voll genutzt werden. Die Komplexität der Topologien (Anzahl Schalter und Kapazitäten) steigt jedoch mit zunehmender Anzahl an Übersetzungsverhältnissen stark an. Volle Ausnutzung der Kapazitäten bei gleichzeitig optimaler Skalierung ist dabei sehr wichtig. Diese Faktoren begrenzen die Anzahl an Übersetzungsverhältnissen. Die Anzahl kann deshalb nicht beliebig hoch gewählt werden.

Einen Überblick über aktuell implementierte SC-Wandler gibt Tabelle 1. Es fällt auf, dass die SC-Wandler nur sehr kleine Eingangsspannungen und Spannungsbereiche aufweisen. Das liegt zum Teil daran, dass aktuell viele Wandler für batteriebetriebene Anwendungen entwickelt werden. Ein weiterer Grund sind zusätzliche Herausforderungen, die bei der effizienten Wandlung hoher Eingangsspannungen zu lösen sind. Aus der Tabelle 1 treten zwei Architekturen hervor, der rekursive SC-Wandler (RSC-Wandler) [5] und der Folding Dickson SC-Wandler [6]. Referenz [5] realisiert dabei eine sehr große Anzahl an Übersetzungsverhältnissen, während [6] höhere Spannungen und einen relativ großen Eingangsspannungsbereich abdeckt. Diese beiden Ansätze sind für die hier geforderte Anwendung prinzipiell geeignet. Der Folding Dickson SC-Wandler [6] ist jedoch im Übersetzungsverhältnis begrenzt. Der RSC-Wandler [5] bietet durch seinen modularen Aufbau den vielversprechendsten Ansatz für hohe Eingangsspannungen und zugleich einen großen Eingangsspannungsbereich.

#### C. Aufbau des RSC-Wandlers

Im Folgenden wird der Aufbau des RSC-Wandlers am Beispiel eines 2-Bit RSC-Wandlers beschrieben. In Abbildung 5 sind die damit möglichen Konfigurationsmöglichkeiten dargestellt. Ein 2-Bit-RSC-Wandler bietet die drei abgebildeten Übersetzungsmöglichkeiten. Der modulare Aufbau mit 2:1-Zellen kann beliebig erweitert werden, wodurch weitere Übersetzungsverhältnisse möglich sind. Eine 2:1-Zelle besteht aus dem in Abbildung 1 dargestellten Serien-Parallel-Wandler. Eine mögliche Realisierung des 2-Bit-RSC-Wandlers kann wie in Abbildung 6 aussehen. Dieser besteht aus

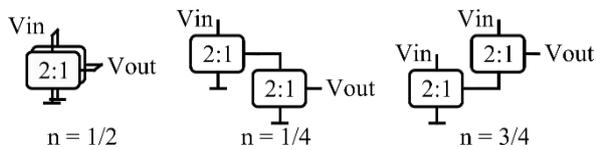


Abbildung 5: Konfigurationsmöglichkeiten eines 2-Bit-RSC-Wandlers.

zwei Grundzellen (nach Abbildung 1) und den zusätzlichen Schaltern S15, S22 und S23, welche zur Umschaltung des Übersetzungsverhältnisses benötigt werden.

#### D. Design-Herausforderungen für hohe Spannungen

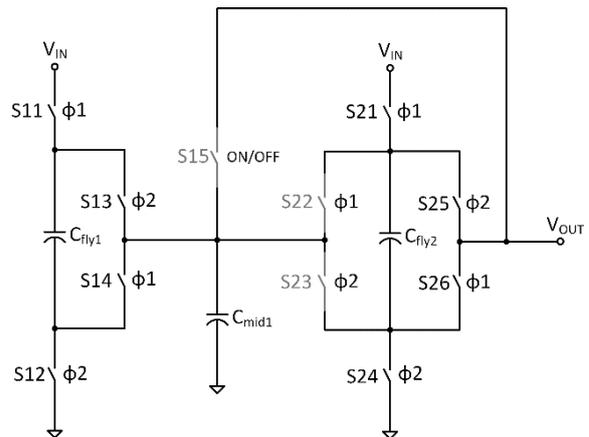
Bei hohen Eingangsspannungen ( $> 5\text{ V}$ ) ergeben sich besondere Herausforderungen, die bei kleinen Spannungen in dieser Art nicht entstehen. Im Folgenden werden die wesentlichen Unterschiede dargestellt. Die Aufzählung erhebt nicht den Anspruch auf Vollständigkeit.

- Hochvolt-Transistoren: Durch die relativ hohen Spannungen müssen Hochvolt-Transistoren verwendet werden. Die Ansteuerverluste für Hochvolt-Transistoren sind wegen der größeren Gatekapazität deutlich größer als bei Niedervolt-Transistoren.
- Levelshifter: Für die Ansteuerung der Transistoren auf fliegendem Potential wird der Einsatz von Levelshiftern notwendig.
- Ladungspumpe: Für jeden Hochvolt-Transistor wird auch eine Ladungspumpe für den Gate-Overdrive (Einschaltspannung) benötigt. Die dadurch entstehenden zusätzlichen Verluste in Levelshiftern und Ladungspumpe wirken sich negativ auf den Wirkungsgrad aus.
- Fliegende Kapazitäten: Durch die erhöhten Spannungshübe entstehen an den parasitären Kapazitäten Umladeverluste, welche mit der Spannung quadratisch skalieren.
- Isolationswannen: Durch Potentialänderungen an den Isolationswannen entstehen Umladeverluste.
- Bodydioden: Durch unterschiedliche Konfiguration der einzelnen Zellen entstehen Potentialsprünge an den Source- und Drainanschlüssen der Schalttransistoren, die in ungünstigen Fällen dafür sorgen, dass eine Back-to-Back-Schalteranordnung nötig wird. Dadurch kann verhindert werden, dass über die Bodydioden unerwünschte Querströme fließen können.

Die beschriebenen Herausforderungen und Effekte führen durch zusätzliche Verluste zu einer Reduzierung des Wirkungsgrades.

#### IV. FAZIT

SC-Wandler eignen sich durch ihre gute Integrierbarkeit sehr gut für vollintegrierte DC-DC-Wandler. Die Ausgangsspannung eines SC-Wandlers wird durch die



Verhältnis	S15	S21	S22	S23	S24
1/2	ON	$\phi 1$	OFF	OFF	$\phi 2$
1/4	OFF	OFF	$\phi 1$	OFF	$\phi 2$
3/4	OFF	ON	OFF	$\phi 2$	OFF

Abbildung 6: 2-Bit-RSC-Wandler mit Konfigurationsschalter.

Parametergrößen Spannungseinbruch und Welligkeit der Ausgangsspannung charakterisiert. Diese Parameter können durch die Dimensionierung des SC-Wandlers entsprechend der Anwendung angepasst werden. Für eine effiziente Wandlung mit hohen Eingangsspannungen und einem weiten Eingangsspannungsbereichen müssen SC-Wandler mit mehreren Übersetzungsverhältnissen aufgebaut werden. Für hohe Eingangsspannungen ( $> 5\text{ V}$ ) ergeben sich weitere Herausforderungen für den Schaltungsentwurf. Bedingt durch die hohen Spannungen müssen Hochvolt-Transistoren verwendet werden, welche zur Ansteuerung sowohl Levelshifter als auch Ladungspumpen benötigen. Die im Wandler auftretenden hohen Spannungshübe bewirken höhere Verluste in den parasitären Elementen und wirken sich negativ auf den Wirkungsgrad aus. Der vorgestellte RSC-Wandler eignet sich besonders gut für vollintegrierte DC-DC-Wandler mit hoher variabler Eingangsspannung.

#### DANKSAGUNG

Diese Arbeit ist gefördert vom Bundesministerium für Bildung und Forschung unter dem Förderkennzeichen IN2013-425-203.

#### LITERATURVERZEICHNIS

- [1] M. Seeman and S. Sanders, "Analysis and Optimization of Switched-Capacitor DC," *IEEE Transactions on Power Electronics*, p. 23, March 2008.
- [2] Y. Tsvetkov, "Principles of operation and analysis of switched-capacitor circuits," *Proceedings of the IEEE*, p. 71, Aug 1983.
- [3] T. Van Breusegem and M. Steyaert, "Multi-Phase Interleaving," *CMOS Integrated Capacitive DC-DC Converters*, New York, Springer Science+Business Media, 2013, pp. 104-122.

- [4] M. Seeman, "A Design Methodology for Switched-Capacitor DC-DC-Converters," *University of California at Berkeley, USA, Diss.*, 2009.
- [5] L. Salem and P. Mercier, "A Recursive Switched-Capacitor DC-DC-Converter Achieving 2 N - 1 Ratios With High Efficiency Over a Wide Output Voltage Range," *IEEE Journal of Solid-State Circuits*, 49, 2014.
- [6] A. Sarafianos and M. Steyaert, "The folding dickson converter: A step towards fully integrated wide input range capacitive DC-DC-converters," in *European Solid-State Circuits Conference (ESSCIRC)*, 2014.
- [7] K. Philips, "Ultra Low Power Short Range Radios - Covering the last mile of the IoT," in *European Solid-State Circuits Conference (ESSCIRC)*, 2014.
- [8] L. G. Salem and P. P. Mercier, "An 85%-Efficiency Fully Integrated 15-Ratio Recursive Switched-Capacitor DC-DC-Converter with 0.1-to-2.2V Output Voltage Range," *International Solid-State Circuits Conference (ISSCC)*, 2014.
- [9] Y. Ramadass and A. Chandrakasan, "Voltage Scalable Switched Capacitor DC-DC-Converters," *Power Electronics Specialists Conference*, 2007.
- [10] L. Su and D. Ma, " Monolithic reconfigurable SC power converter with adaptive gain control and on-chip capacitor sizing," *Energy Conversion Congress and Exposition (ECCE)*, 2010.
- [11] H.-P. Le, S. Sanders and E. Alon, "Design Techniques for Fully Integrated Switched-Capacitor DC-DC-Converters," in *IEEE Journal of Solid-State Circuits*, 46, 2011.
- [12] H. Meyvaert, T. van Breussegem and M. Steyaert, "A 1.65 W Fully Integrated 90 nm Bulk CMOS Intrinsic Charge Recycling Capacitive DC-DC-Converter: Design & Techniques for High Power Density," *Energy Conversion Congress and Exposition*, 2011.



Daniel Lutz erhielt den akademischen Grad des Bachelor of Engineering in Mechatronik/Elektrotechnik im Jahr 2011 von der Hochschule Esslingen und den Grad des M.Sc. in Leistungs- und Mikroelektronik im Jahr 2013 von der Hochschule Reutlingen. Er ist Akademischer Mitarbeiter am Robert Bosch Zentrum für Leistungselektronik an der Hochschule Reutlingen.



Peter Renz erhielt den akademischen Grad des Bachelor of Engineering in Mechatronik im Jahr 2013 von der Hochschule Reutlingen. Seit 2013 studiert er im Masterstudiengang Leistungs- und Mikroelektronik am Robert Bosch Zentrum für Leistungselektronik an der Hochschule Reutlingen.



Bernhard Wicht erhielt den akademischen Grad Dipl.-Ing. im Jahr 1996 von der Technischen Universität Dresden und den Grad Dr.-Ing. im Jahr 2002 von der Technischen Universität München. 2003-2010 war er im Geschäftsbereich Mixed Signal Automotive bei Texas Instruments, Freising, verantwortlich für die Entwicklung von Smart Power ICs. Seit September 2010 ist er Professor für integrierte Schaltungen am Robert Bosch Zentrum für Leistungselektronik an der Hochschule Reutlingen.

# Entwicklung und Layoutentwurf eines Analog-Digital-Wandlers mit 12 Bit Auflösung in einer 180 nm-CMOS-Technologie

Benjamin Dusch, Mayukh Bhattacharya, Dirk Jansen

**Zusammenfassung**—Bestandteil dieser Entwicklung sind das Design sowie das Layout eines 12-Bit Analog-Digital-Wandlers nach der Methode der sukzessiven Approximation in einer 180 nm-CMOS-Zieltechnologie. Das angewandte Verfahren beruht auf dem Konzept von De Venuti mit geschaltetem C-Array, jedoch mit maßgeblichen Verbesserungen. In der Analyse werden geforderte INL und DNL erreicht, die Stromaufnahme liegt bei statischem Betrieb im  $\mu\text{A}$ -Bereich und erfüllt damit die Kriterien eines Low-Power-ADCs. Der Eingangsspannungsbereich beträgt  $\pm 500\text{ mV}$  bei nur  $1,2\text{ V}$  Versorgungsspannung. Der Wandler ist in ein Datenerfassungssystem eingebettet, welches sich noch in der Detailentwicklung befindet. Im Layout wird eine Fläche unter  $0,5\text{ mm}^2$  benötigt.

**Schlüsselwörter**—ASIC, Analog-Digital-Wandler, Sukzessive Approximation, CMOS, Komparator.

## I. EINLEITUNG

Motivation für diese Arbeit ist der Wunsch des Instituts für Angewandte Forschung (IAF) der HS Offenburg, einen leistungsfähigen, hochauflösenden und genauen, aber verlustleistungsarmen Analog-Digital-Wandler als fertige einsetzbare mikro-elektronische Komponente einsatzbereit zu haben. Benötigt wird solch ein Wandler immer wieder in komplexen sensor-signalverarbeitenden Ultra-Low-Power-Systemen, wie sie am IAF in Form von Forschungs- und Projektarbeiten entwickelt werden (Abb. 1). Die Entwicklung erfolgt in der Zieltechnologie des 180 nm-CMOS-Mixed-Signal-Prozesses von UMC (United Microelectronics Corporation), da dieser im IAF bereits seit mehreren Jahren im Digitalbereich erfolgreich eingesetzt und gut beherrscht wird. Des Weiteren soll der für Analogdesign erforderliche Designflow zur Erstellung eines Mixed-Signal-ICs erarbeitet und verifiziert werden.

Der Wandler ist Teil eines Datenerfassungssystems mit Kanalselektor, Instrumentenverstärker und Referenzspannungsquelle, die hier nicht näher darge-

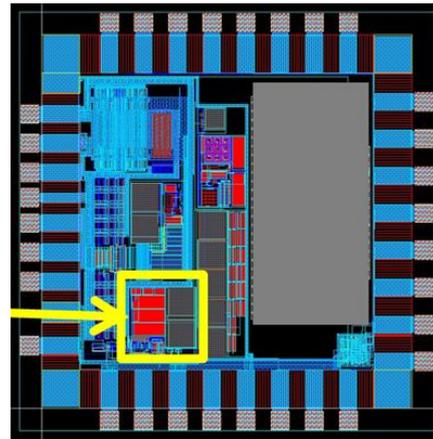


Abbildung 1: Layout eines am IAF entwickelten ASICs mit Integration eines AD-Wandler-Moduls (eingerahmter Bereich).

stellt sind. Er soll im Zusammenwirken mit einem Kanalselektor betrieben werden, so dass eine größere Anzahl von Kanälen (zunächst 16) unterschiedlicher Art erfasst werden können. Auswahl der Kanäle, Abtastrate und jeweilige Verstärkung werden digital programmiert. Die Anwendungen liegen im Bereich der Biosignalverarbeitung, wo Bandbreiten von wenigen Hz bis zu kHz vorliegen, so dass die einzelnen Kanäle bei einer Auslegung der analogen Vorverarbeitung für eine Bandbreite von 20 kHz sicher abgetastet werden können. Diese Vorverarbeitung ist jedoch nicht Gegenstand dieser Darstellung. Für das Wandlerverfahren wird das Prinzip der sukzessiven Approximation angesetzt. Folgende Anforderungen werden an den Wandler gestellt:

- Es wird eine Auflösung von 12 Bit angestrebt bei einem Eingangsspannungsbereich von  $\pm 0,5\text{ V}$ , bezogen auf eine Referenzspannung.
- Sample-Raten bis zu  $100\text{ kS/s}$  sollen möglich sein.
- Die Verlustleistung soll unter  $20\text{ }\mu\text{W}$  liegen, da ein Einsatz in passiv versorgten Anwendungen vorgesehen ist.
- Der Wandler soll mit einer Versorgungsspannung von  $1,2\text{ V}$  arbeiten können.
- Der Wandler soll im Temperaturbereich  $-40\text{ }^{\circ}\text{C}$  bis  $85\text{ }^{\circ}\text{C}$  einsetzbar sein.
- Die vom AD-Wandler-System benötigte Fläche soll weniger als  $0,5\text{ mm}^2$  betragen.

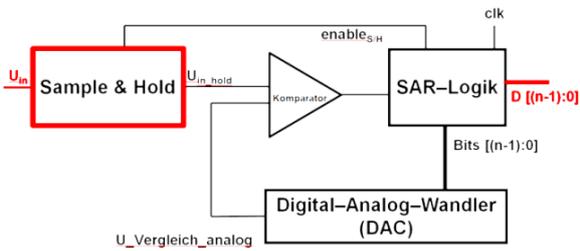


Abbildung 2: Klassischer SAR-Wandler.

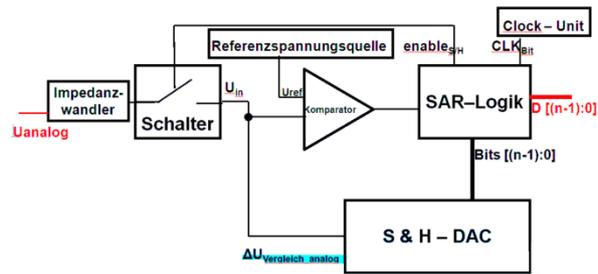


Abbildung 3: SAR-Wandler nach De Venuto.

## II. ANALOG-DIGITAL-WANDLUNG NACH DER METHODE DER SUKZESSIVEN APPROXIMATION

### A. Konzeptvergleich und -referenz

Bei der Recherche zur Umsetzung des Verfahrens der sukzessiven Approximation unter Low-Power-Bedingungen wurden wir auf das von De Venuto beschriebene Verfahren aufmerksam [1]-[3]. Hier wird ein 12-bit-SAR-ADC beschrieben, der eine neue Methode der Umsetzung des Algorithmus der sukzessiven Approximation anwendet, welche entscheidende Vorteile gegenüber dem in Abb. 2 dargestellten Verfahren bietet. Bei dieser Methode übernimmt der DAC auch die Funktion des Sample & Holds der analogen Eingangsspannung, indem er durch Bereitstellen seines Kondensatoren-Arrays als „Abtastkondensator“ arbeitet (Abb. 3). Das Eingangssignal wird als Ladung auf dem Kondensator-Array gespeichert, das dazu in einer definierten Ausgangsstellung steht. Der DAC addiert/subtrahiert nun binär gestufte Ladungsquantitäten auf diesen Knotenpunkt. Verglichen wird die Spannung des Knotens mit einer festen Referenzspannung am Komparator (Abb. 3). Im Gegensatz zum Vergleich der Abtastspannung mit DAC-Ausgangsspannungen beim Standardprinzip (Abb. 2) wird hier auf eine feste Mittenspannung (die Referenzspannung) hin abgeglichen. Die Vorteile dieses modifizierten Prinzips sind das Entfallen einer zusätzlichen S&H-Kapazität sowie die vereinfachte Auslegung des Komparators, der nur noch mit der festen Referenzspannung vergleichen muss, nicht mehr im gesamten Spannungsbereich. Dies ist bei der niedrigen, unipolaren Versorgungsspannung eine große Erleichterung.

Die vorliegende Darstellung baut auf der oben beschriebenen Arbeit De Venutos auf, jedoch wird hier eine andere Technologie verwendet. De Venuto erreicht eine Genauigkeit von 11 Bit bei einer implementierten Auflösung von 12 Bit. Die maximale Abtastrate beträgt nur 6,25 kHz, der verwendete Prozess hat eine Strukturgröße von 140 nm, wodurch eine Chipfläche von 0,35 mm<sup>2</sup> erreicht wird. Der kleinste Kondensator (1 LSB) ist 1 fF groß, versorgt wird das System mit einer stabilisierten Spannung von 1,2 V.

Der Eingangsspannungsbereich beträgt 0 – 700 mV unipolar. De Venuto verwendet zudem einen dynamischen Komparator, eine extreme Low-Power-Lösung, die jedoch im Vergleich zu einem statischen Komparator mit Entwurfsrisiken verbunden ist. Die sehr kleine LSB-Kapazität von 1 fF stellt zudem bei den unvermeidlichen parasitären Kapazitäten im Layout ein weiteres, vermeidbares Risiko dar. Das hier beschriebene Design ist weniger aggressiv und versucht Risiken zu vermeiden. Weiterhin wird eine höhere Bandbreite benötigt, der Eingangsspannungsbereich soll auf möglichst  $\pm 500$  mV vergrößert werden. Die damit absehbare höhere Verlustleistung fällt dagegen nicht ins Gewicht. Auf die weiteren Änderungen gegenüber dem De Venuto-Konzept wird noch eingegangen.

### B. Konzept des Analog-Digital-Wandler-Systems

Der eigentliche Analog-Digital-Wandler wird in der Komponente ADC (Abb. 4) umgesetzt, welche in die übergeordnete Baugruppe mit der Bezeichnung ADC-System eingebettet ist. Die SAR-Logik (successive approximation register logic) initiiert die Abtastung über das Signal `enable_notS&H`. In ihr ist der Algorithmus der sukzessiven Approximation digital implementiert, sie übernimmt die Steuerung des Sample & Hold-DAC und speichert die durch Abarbeiten des Algorithmus errechneten Bits ab, um sie nach der Wandlung als digitales Ausgangswort  $D[(N-1):0]$  auszugeben. Getaktet wird die Logik durch  $CLK_{BIT}$ . Hierbei bedeutet  $N$  die Anzahl der Bits des ADCs, im vorliegenden Fall also  $N = 12$ .

Das digitale Ausgangswort  $D[(N-1):0]$  nach der eigentlichen AD-Wandlung gelangt über den Umweg mehrerer Register kontrolliert auf den externen Datenbus `Data_Ext [7:0]`. Auch die ADC-Statusabfrage sowie die Konfiguration des ADCs von extern erfolgt durch Nutzung des Datenbusses `Data_Ext [7:0]`. Kontrolliert wird der gesamte Datenfluss durch den `Data-Reg Selector`. In ihm wird ein Protokoll implementiert, welches die Arbitrierung des Busses regelt. Der Buszugriff wird durch diese Einheit, welche mittels eines `Chip-Selects` und `Adressbusses` von extern angesprochen wird, organisiert.

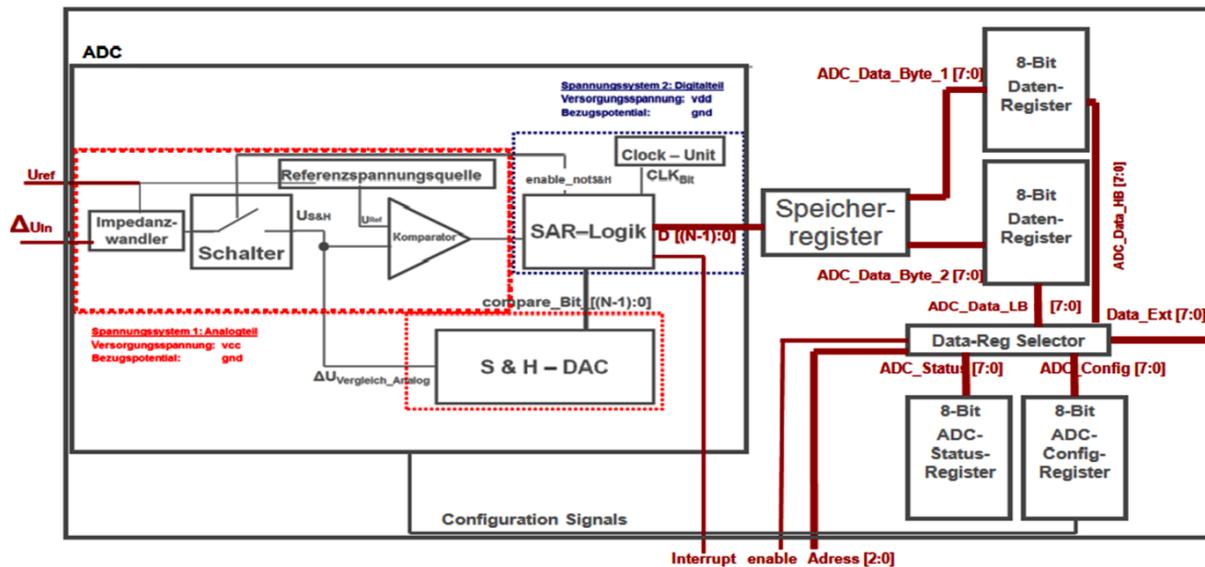


Abbildung 4: Analog-Digital-Wandler (ADC) mit Datenspeicher.

Ein Interrupt-Signal wird ebenfalls aus der SAR-Logik nach außen geführt und signalisiert eine erfolgreiche Wandlung. Es ergibt sich folgende Schnittstelle nach außen:

- Analoge Eingangsspannung  $\Delta U_{in}$ .
- Bezugspotential  $U_{ref}$
- Externer Datenbus  $Data\_Ext [7:0]$
- Externer Adressbus  $Address [2:0]$
- Chip-Select/Enable
- Interrupt

Der Impedanzwandler ist erforderlich, da die Sample-Kapazität mit etwa 16 pF relativ groß ist und von einer hochohmigen Quelle nicht getrieben werden kann. Gleichzeitig kann diese Schaltung auch als Verstärker für kleine Signale dienen (z.B. EKG-Signal mit wenigen mV) und ist daher als Instrumentenverstärker ausgeführt.

Der Vorgang der Messung beginnt mit der Übertragung des Ausgangspotentials des Impedanzwandlers auf den Knoten  $U_{S\&H}$  durch Schließen des Schalters für eine definierte Zeit, gesteuert durch die SAR-Logik. Im einfachsten Fall ist dies eine Taktperiode. Der Komparator gibt mit seinem Ausgangssignal  $compare\_result$  an, ob die Spannung am Messpunkt  $U_{S\&H}$  kleiner oder größer als  $U_{ref}$  ist. Damit wird in der Abarbeitung des Algorithmus ermittelt, ob Ladung hinzugefügt oder entfernt werden muss. Ist die Spannung  $U_{S\&H}$  größer als  $U_{ref}$ , wird Ladung entfernt, ist sie kleiner, wird Ladung hinzugefügt. Die Ladungspakete sind dabei entsprechend der sukzessiven Approximation binär gestuft, mit jedem Vergleichsvorgang wird ein Bit bestimmt. Die Auflösung und Genauigkeit des Komparators muss dabei so groß sein, dass er den Ladungsunterschied einer halben LSB-Ladung noch sicher unterscheiden kann.

Der Algorithmus startet mit '10...0', entsprechend ist der MSB-Kondensator auf  $Gnd$ , alle weiteren Stel-

len auf  $V_{CC}$ . Im weiteren Verlauf wird nun jede Ladungswertigkeit entsprechend den Bits vom MSB-1 bis zum LSB geschaltet und jeweils das Verhalten des Komparators ausgewertet. Die Ladungspakete entsprechen dabei den Wertigkeiten dadurch, dass jeder Kondensator auf der digitalen Seite von einem Inverter getrieben wird, der den Spannungshub von  $V_{CC}$  nach  $Gnd$  oder zurück durchfährt. Die Spannung  $V_{CC}$  ist dabei im analogen Sinne geregelt auf doppelte  $U_{ref}$ -Spannung, so dass das kleinste Ladungspaket

$$Q_0 = C_0 \cdot V_{CC} = 2 \cdot C_0 \cdot U_{ref}$$

beträgt, wobei  $C_0$  die mit dem LSB verbundene Einheitskapazität von 4 fF darstellt, hier viermal so groß gewählt wie bei De Venuto.

Eines der wichtigsten Elemente des ADCs ist neben dem Komparator der bereits erwähnte S&H-DAC. Er stellt ein Array von getriebenen Kapazitäten dar und ist aufgebaut aus  $N$  Kapazitäten, die den Bitwertigkeiten entsprechend, hier also 12 Bit, dimensioniert sind. Die digitalen Eingangssignale  $compare\_Bit [(N-1):0]$  werden über Treiber, deren Funktion hier wahrgenommen wird von einfachen Invertern, "analog" verwendet und den Kondensatoren zugeführt. Die Treiber vermögen die stabilisierten Potentiale  $V_{CC}$  oder  $Gnd$  an die Kondensatoren anzulegen.

Der DAC stellt ein Kapazitäten-Array zum Erzeugen von Ladungspaketen dar, sichtbar als Spannungsdeltas ( $\Delta U_{Vergleich\_Analog}$ ) am Komparator-Eingang. Er generiert den Bitwertigkeiten entsprechende Ladungspakete, wandelt somit die von der SAR-Logik digital gesetzten Bitwertigkeiten  $compare\_Bit [(N-1):0]$  in analoge Spannungsdeltas am Messpunkt um. Da bei Abtastung das  $C_{MSB} = C_{N-1}$  gesetzt ist (auf  $Gnd$  liegt), die anderen Kondensatoren  $C_{N-2}$  bis  $C_0$  nicht gesetzt sind (S&H-DAC ist im Zustand '10...0'), bewirkt ein Um-

schalten der Potentiale an den Kondensatoren, gesteuert durch die SAR-Logik, ein Umladen der Kondensatoren, woraus Spannungsdeltas am Messpunkt hervorgehen. Zieht der DAC sein  $C_{MSB}$  nach der Abtastung auf  $U_{stab} = V_{CC} = 1,2$  V oder bei Verwendung einer externen Versorgungsspannung auf  $U_{stab} = V_{CCext} = 1,8$  V und setzt es damit auf logisch '0', ergibt sich ein Spannungsdelta von

$$\begin{aligned}\Delta U_{Vergleich\_Analog} &= -(C_{MSB} \div C_{ges}) \cdot U_{stab} \\ &= -(C_{N-1} \div C_{ges}) \cdot U_{stab}\end{aligned}$$

Wird hingegen das  $C_{MSB}$  nach der Abtastung auf  $Gnd$  belassen und die nächst kleinere Kapazität  $C_{N-2}$  von  $V_{CC}$  auf  $Gnd$  gezogen, damit auf logisch '1' gesetzt, ergibt sich ein Spannungsdelta von

$$\Delta U_{Vergleich\_Analog} = +(C_{N-2} \div C_{ges}) \cdot U_{stab}$$

Umschalten von Kondensatoren auf  $V_{CC}$  bedeutet demnach ein negatives, Umschalten auf  $Gnd$  ein positives Spannungsdelta. Werden zeitgleich in einem Schritt zwei Kondensatoren bezüglich ihrer Potentiale umgeschaltet, beispielsweise  $C_{MSB} = C_{N-1}$  nach der Abtastung auf  $V_{CC}$  gezogen und  $C_{N-2}$  auf  $Gnd$ , sowie der DAC in den Zustand '01000...' (zum Vergleich der analogen Eingangsspannung  $U_{S\&H}$  mit der dem Zustand '01000...' entsprechenden Spannung) gebracht, stellt sich ein Spannungsdelta gemäß der Summe der durch Umschaltung der beiden Kondensatoren hervorgerufenen Spannungsdeltas, ein:

$$\begin{aligned}\Delta U_{Vergleich\_Analog} &= -(C_{N-1} \div C_{ges}) \cdot U_{stab} \\ &+ (C_{N-2} \div C_{ges}) \cdot U_{stab}\end{aligned}$$

Es lässt sich eine Gesetzmäßigkeit des vom DAC hervorgerufenen Spannungsdeltas am Messpunkt (Eingang des Komparators) wie folgt ausdrücken:

$$\Delta U_{Vergleich\_Analog} = [(C_{1 \rightarrow 0} - C_{0 \rightarrow 1}) \div C_{ges}] \cdot U_{stab}$$

Dabei bezeichnet  $C_{1 \rightarrow 0}$  die Kapazität der Kondensatoren, welche von logisch '1' auf logisch '0' umgeschaltet werden und  $C_{0 \rightarrow 1}$  die Kapazität der Kondensatoren, welche von logisch '0' auf logisch '1' umgeschaltet werden. Da die am DAC-Ausgang generierten Spannungsdeltas  $\Delta U_{Vergleich\_Analog}$  direkt in die Messung einfließen, müssen die Potentiale  $V_{CC}$  und  $Gnd$  präzise und stabil sein und den Versorgungsspannungen am Komparator entsprechen. Somit werden Komparator, S&H-DAC, des weiteren Impedanzwandler und Schalter, da deren Ausgangsspannungen ebenfalls am Messpunkt einfließen, durch eine stabilisierte Spannung  $V_{CC}$  und  $Gnd$  versorgt. Diese stabile Spannung  $V_{CC}$  wird von einer Referenzspannungsquelle  $U_{ref}$  abgeleitet. Es kann jedoch auch eine externe Versorgung ( $U_{stab} = V_{CCext}$ ) erfolgen. Das Potential  $Gnd$  ist

mit dem digitalen Massepunkt verbunden. Der Digitalteil wird durch die Spannung  $V_{DD}$ , bezogen auf das niedrigste Potential  $Gnd$ , versorgt. Analoge- und digitale Versorgung sind getrennt, jedoch bei  $Gnd$  verbunden (im Layout nur an einer Stelle).

Die von außen zugeführten Biosignale werden in der Signalaufbereitung differentiell auf  $U_{ref}$  als virtuelle Masse bezogen, die über einen Treiber nach außen geführt wird. Die Potentiale  $V_{DD}$ ,  $V_{CC}$  wie auch  $Gnd$  sind deshalb als schwimmende Potentiale zu bewerten, deren absolute Größe nur in die Verstärkung eingeht.

### III. ENTWICKLUNG DES ANALOG-DIGITAL-WANDLER-SYSTEMS

#### A. Einführung in die verwendete Technologie

Die Arbeit basiert auf einem Mixed-Signal-Triple-Well-RFCMOS-Prozess des Herstellers UMC mit der Strukturgröße 180 nm. Verfügbar im Prozess sind neben dem Standard-FET mit etwa 0,4 V Schwellenspannung auch Transistoren mit kleinerer Schwellenspannung nahe 0 V, die den Entwurf von Analogkomponenten mit niedriger Versorgungsspannung erleichtern. Für den Analogteil des ADCs werden weiterhin Kondensatoren und Widerstände benötigt. Kondensatoren werden als M2M-Kapazitäten zwischen Metal-Layer 5 und 6 realisiert, sind also Teil der Verdrahtung und relativ genau. Widerstände werden als Polysilizium-Widerstände ausgeführt, wobei hier das für die Gates verwendete Polysilizium verwendet wird, ein Schutzlayer verhindert die Dotierung. Damit können Widerstände mit etwa 20 % absoluter, aber hoher relativer Genauigkeit erzeugt werden. Wie in allen CMOS Technologien sind Widerstände aber sehr flächenintensiv und möglichst zu vermeiden.

Die Umsetzung des Digitalteils erfolgt durch Verwendung einer Standardzellen-Bibliothek, welche alle für die Synthese grundsätzlich benötigten Gatter und Register beinhaltet. Die verwendeten Standardzellen werden vom IMEC/UMC in das Design integriert.

Die Digitalversorgungsspannung ist mit  $1,8 \pm 0,18$  V toleriert. Da der Analogteil des ADCs mit einer stabilisierten Versorgungsspannung  $V_{CC} = 1,2$  V betrieben werden soll, liegt es nahe, den Digitalteil ebenfalls mit 1,2 V zu versorgen. Nach Recherche funktionieren die im Digitalteil verwendeten einfachen Gatter und Register auch noch bei dieser unter dem zulässigen Toleranzbereich des Herstellers liegenden reduzierten Spannung. Die mit der Versorgungsspannungsreduktion verbundenen größeren Laufzeiten und Verzögerungen sind hier ohne Bedeutung.

#### B. Umsetzung des Digitalteils

Nach dem Entwurf eines schaltungstechnischen Grundkonzepts der Logik des ADC-Systems erfolgte die programmieretechnische Umsetzung der Logik in der Hardwarebeschreibungssprache VHDL.

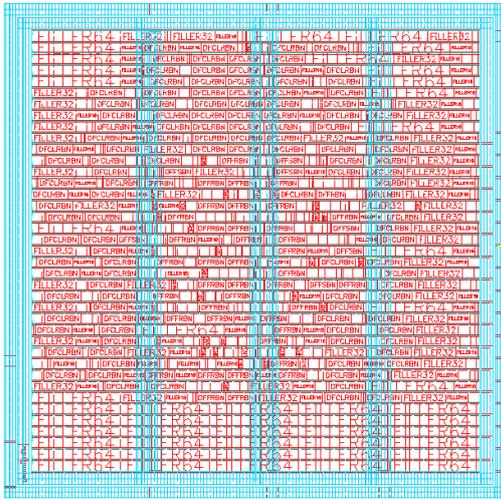


Abbildung 5: Layout des Digitalteils des ADC-Systems im Programm Cadence Encounter: Ansicht der Standardzellen mit zusätzlichen Füllzellen (Filler) zum Auffüllen des Floorplans.

Entworfen wurden eine SAR-Logik, eine Clock-Einheit, die zum gewünschten Betrieb mit der gegenüber der externen Taktversorgung reduzierten Frequenz sowie zur Erzeugung aller im ADC-System benötigten Taktsignale notwendig ist, eine Einheit zum kontrollierten Aktivieren und Deaktivieren der AD-Wandlung im gewünschten ADC-Betriebsmodus, eine Einheit zur Steuerung der Buskommunikation und mehrere Register zum Abspeichern von Status, Konfiguration und digitaler Ausgangsworte des ADCs.

Die Schaltungssynthese erfolgte automatisiert und skriptgesteuert. Dazu wurden eigene Syntheseskripte in der Sprache Tcl auf das Design angepasst. Über mehrere inkrementale Synthesedurchgänge bei Verwendung des Syntheseprogramms von Synopsys wird die Netzliste des Designs erzeugt und optimiert. Die Netzliste wird zieltechnologiespezifisch in Verilog herausgeschrieben.

Mit der generierten Netzliste wurde im Tool Encounter von Cadence das Layout synthetisiert. Das Design kann auf einer Fläche von  $200\ \mu\text{m} \times 100\ \mu\text{m}$  geroutet werden. Um flexibel in der Ausführung der Speicherregistergröße zu sein, ist der Floorplan hier  $200\ \mu\text{m} \times 200\ \mu\text{m}$  groß ausgelegt (Abb. 5). So kann der Floorplan wiederverwendet werden, falls das Speicherregister in seiner Größe (im Layout aktuell 10 ADC-Datenausgangswörter, also  $10 \times 12\ \text{Bit} = 120\ \text{Bit}$ ) nochmals geändert wird. Im Floorplan hätten theoretisch bis zu 60 ADC-Datenausgangswörter, also  $60 \times 12\ \text{Bit} = 720\ \text{Bit}$ , Platz. Abb. 5 zeigt die platzierten Standardzellen im Floorplan inklusive der Füllzellen. Umrahmt wird der Floorplan durch die Power-Ringversorgung, welche  $Gnd$  und  $V_{DD}$  jeder Standardzelle über Querverbindungen zuführt. Drei vertikal verlaufende Power-Bars stützen die Versorgung der Standardzellen in der Floorplanmitte.

### C. Umsetzung des Analogteils

Der Schaltungsentwurf des Analogteils wurde mit der Plattform Virtuoso von Cadence durchgeführt. Im Folgenden sollen die wichtigsten Komponenten Instrumentenverstärker, Komparator und S&H-DAC näher erläutert werden.

#### 1) Instrumentenverstärker

Wie schon zuvor dargestellt, kann der Impedanzwandler mit einer Verstärkung versehen werden, am besten wird er als Instrumentenverstärker mit differentiellem Eingang ausgeführt. Ein Rail-to-Rail-Eingangsspannungsbereich, der vom niedrigsten Potential der Schaltung ( $Gnd, 0\ \text{V}$ ) bis zum höchsten ( $U_{stab} = V_{CC} = 1,2\ \text{V}$  oder bei Verwendung einer externen Versorgungsspannung  $U_{stab} = V_{CCext} = 1,8\ \text{V}$ ) ist nicht zu realisieren:

- Parasitäre Eingangs- und Ausgangskapazitäten innerhalb der Schaltung lassen ein Erreichen dieser Potentiale nicht zu. So verringern die Eingangskapazität des Komparators ebenso wie die Ausgangskapazität des geschlossenen Abtastschalters sowie die parasitären Kapazitäten den Bereich, den das Kapazitätsarray aussteuern kann.
- Der Instrumentenverstärker arbeitet nichtlinear in den Grenzbereichen der Aussteuerung des Verstärkerausgangs, ist also nicht für Rail-to-Rail ausgelegt, der Ausgangsspannungsbereich kann also nicht voll ausgenutzt werden.

Als Konsequenz der beiden genannten Aspekte wird der Aussteuerbereich des Instrumentenverstärkers um 16,6 % verringert von maximal 1,2 V auf 1,0 V. Der Bereich wird folglich von minimal 100 mV bis maximal 1,100 V genutzt, bei höherer Versorgungsspannung ( $V_{CCext} = 1,8\ \text{V}$ ) entsprechend höher. Bei einer gewählten Verstärkung von 10 beträgt damit der Eingangsspannungsbereich  $\pm 50\ \text{mV}$  bezogen auf  $U_{ref}$ . Sollte der Eingangsspannungsbereich des AD-Wandlers kleiner als  $\pm 500\ \text{mV}$  ausfallen, kann über die Verstärkung des Instrumentenverstärkers hier nachjustiert werden. Der Verstärker muss in der Lage sein, die hohe Eingangskapazität von fast 20 pF zu treiben. Auf die Detail-Schaltung des Instrumentenverstärkers wird hier nicht näher eingegangen.

#### 2) Komparator

Die Schaltung des Komparators muss temperaturunabhängig die größte Stabilität und Genauigkeit aller Schaltungen des ADC-Systems aufweisen, da von ihr die maßgebenden Entscheidungen zur Ermittlung der Ergebnis-Bits gefällt werden. Wegen möglicher Abtastraten des Systems von bis zu 100 kS/s ( $T_{Bit} = 714\ \text{ns}$ ) sind Durchlaufzeiten im niedrig dreistelligen Nanosekundenbereich gefordert. Die Auslegung des Komparators wird dadurch erleichtert, dass er nur in einem Betriebspunkt ( $U_{ref}$ ), der zugleich auch in der Mitte zwischen den Versorgungsspannungen  $V_{CC}$  und

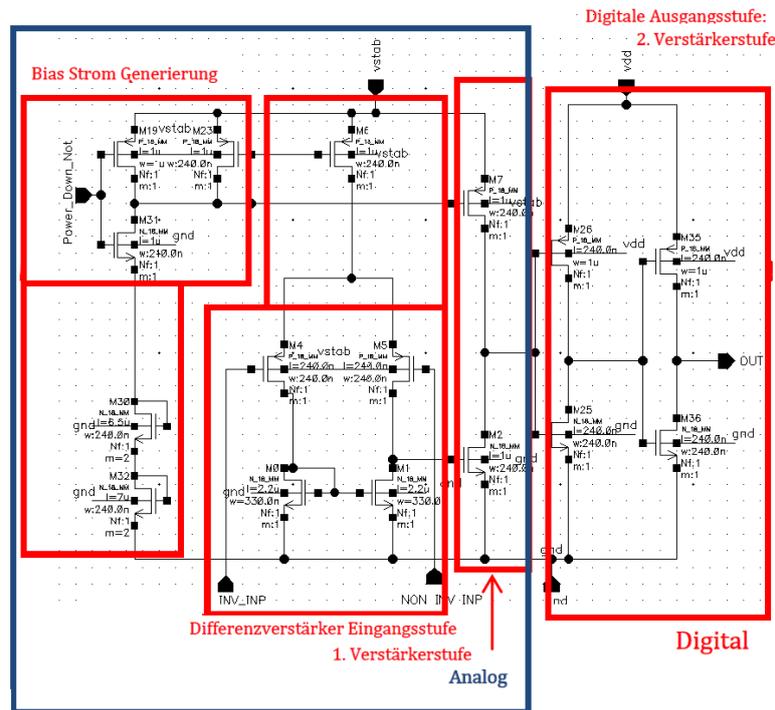


Abbildung 6: Schaltplan des Komparators.

Gnd liegt, Vergleichsentscheidungen fällen muss. Auch schnelles Schaltverhalten wird hierdurch begünstigt.

Aufgebaut ist die Schaltung aus einem Differenzverstärker, welcher eine weitere Verstärkerstufe ansteuert, deren Ausgang einen Inverter ansteuert (Abb. 6). Die Biasstrom-Erzeugung ist in die Schaltung integriert. Der Biasstrom wird über M23, M30 und M32 auf 25,6 nA festgelegt und über M6 und M7 in die Eingangsstufe und die Endstufe eingekoppelt. Bei Power Down des ADC-Systems werden die Gate-Potentiale von M6 und M7 durch einen Inverter auf das Potential der Versorgungsspannung  $U_{stab}$  gebracht, womit der Bias-Strom zu Null wird und der Komparator abgeschaltet wird.

Die PMOS-Eingangstransistoren der Differenzverstärker-Eingangsstufe sind unter Berücksichtigung des Offsetrisikos minimal gewählt, um die parasitäre Eingangskapazität möglichst klein zu halten. Durch die dem Verstärker nachgeschalteten Inverter wird das Signal nochmals verstärkt, was die notwendige Flankensteilheit des Ausgangssignals des Komparators herstellt. An dieser Stelle erfolgt auch die Trennung zwischen Analogversorgung  $V_{CC}$  und Digitalversorgung  $V_{DD}$ . Ist  $V_{CC}$  höher als  $V_{DD}$  bei externer Versorgung, erfolgt hier ein gewisser Pegelshift. Allerdings darf der Unterschied nicht zu groß sein, sonst müsste ein echter Pegelshifter hier eingesetzt werden. Durch die Minimalauslegung der Transistoren können schnelle Schaltzeiten der Inverter erreicht werden. Die Schaltzeit des Komparators von 10 % auf 90 % des Ausgangspegels beträgt weniger als 50 ns und ist

damit ausreichend kurz. Die simulierte Temperaturdrift von ca.  $\pm 10 \mu\text{V}$  der Schaltschwelle des Komparators über den Temperaturbereich von  $-40 \text{ }^\circ\text{C}$  bis  $85 \text{ }^\circ\text{C}$  ist bei einem LSB von  $244 \mu\text{V}$  ohne Relevanz. Tabelle 1 zeigt die Daten des Komparators.

### 3) S&H-DAC

Beim Entwurf des S&H-DAC kommt ein neuartiges Prinzip hinsichtlich der Art, wie die Kondensatoren getrieben werden, zum Einsatz. Die Verwendung von Einheitskondensatoren ist üblich, jedoch nicht die Verwendung von Einheits treibern. Um nach Möglichkeit eine optimale Linearität in der Erzeugung der Ausgangssignale zu erzielen, wird das Kapazitäten-Array des S&H-DAC aus Einheitskondensatoren und Einheits treibern aufgebaut.

Üblich ist es, jede einer Bitwertigkeit entsprechende Kapazität im DAC durch einen eigenen Treiber anzu steuern. Da diese Treiber in ihrer Treiberstärke nicht den Bitwertigkeiten gemäß proportional zueinander dimensioniert sein können, ergibt sich durch die unterschiedlichen Kanalwiderstände unsymmetrisches Lade/Entlade-Verhalten. Setzen die getriebenen Kapazitäten sich dann aus parallelgeschalteten Einheitskapazitäten zusammen, kommt eine nicht zu vernachlässigende parasitäre Kapazität durch die Leitungsverbindungen der parallelen Einheitskondensatoren hinzu. Alle diese Effekte führen zu nicht exakt binär gestuften Kapazitäten und damit zu ungenauen Bitwertigkeiten. Hieraus resultieren Wandlungsfehler bzw. Bitfehler, die die Auflösung begrenzen.

Tabelle 1: Kenndaten des Komparators.

Versorgungsspannung $U_{stab}$	1,2 V
Stromverbrauch $I_{CC}$ (statisch)	77 nA
Referenzspannung $U_{ref}$	600 mV
Anstiegs-/Abfallzeit $t_{10/90}$	< 63 ns
Verzögerungszeit $t_d$	< 650 ns
Temperaturbereich $\Delta T$	-40 – 85 °C

Sind die parallelgeschalteten Einheitskondensatoren aber durch jeweils eigene, gleich dimensionierte parallelgeschaltete Einheitstreiber angesteuert, entfallen die parasitären Kapazitäten der Leitungsverbindungen, da die Kopplung nun auf der digitalen Seite erfolgt, wo parasitäre Kapazitäten unwirksam sind. Die größte Kapazität  $C_{N-1}$  setzt sich damit aus 2048 Kondensatoren zusammen und wird von 2048 Treibern geschaltet. Der Nachteil eines größeren Flächenbedarfs dieses Prinzips im Layout ist gering, da klein ausgelegte Inverter mit einem Flächenbedarf von nur 20  $\mu\text{m}^2$  zum Einsatz kommen können. Die Platzierung der Einheitstreiber erfolgt in dieser Arbeit aus Gründen der Veranschaulichung des Prinzips sowie aus Sicherheitsgründen (der 180 nm-Prozess ist uns noch zu unbekannt, um „riskante“ Speziallayouts auszuführen) neben den Einheitskondensatoren. In zukünftigen Designs kann nach erfolgreicher Verifikation des neuartigen Prinzips durch einen Test-ASIC das Platzieren der Einheitstreiber unter den Einheitskondensatoren erfolgen. Dies ist möglich, da die verwendeten Kondensatoren Metall-zu-Metall-Kondensatoren sind, welche zwischen den Metalllagen 5 und 6 angeordnet werden. Werden nun für das Routing des Arrays nur die Metalllagen 1 bis 4 verwendet und die Inverter unter den Kondensatoren platziert, benötigt die Treiberbaugruppe keinen zusätzlichen Raum. Layout-technisch wird dadurch kein zusätzlicher Flächenbedarf gegenüber dem Standardprinzip benötigt.

#### 4) Auslegung des Offset-Kondensators

Die verwendete schaltungstechnische Umsetzung des SAR-Algorithmus macht es leicht möglich, den gewünschten Eingangsspannungsmessbereich einzustellen. Dies geschieht durch Einfügen eines Kondensators  $C_{Offset}$  (Abb. 7) am Messpunkt der Schaltung. Dieser bewirkt, in gleicher Weise wirkend wie parasitäre Kapazitäten und wie die Eingangskapazität des Komparators, eine Verschiebung des maximal wie minimal durch Aussteuerung  $\Delta U_{Vergleich\_Analog}$  am DAC-Ausgang und damit am Messpunkt der Schaltung erreichbaren Spannungshubs.

Wird ein Eingangsspannungsbereich von  $\Delta U_{in} = U_{in\_max} - U_{in\_min} = 1 \text{ V}$  bei stabilisierter Versorgungs-

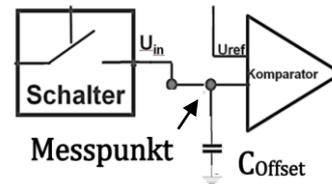
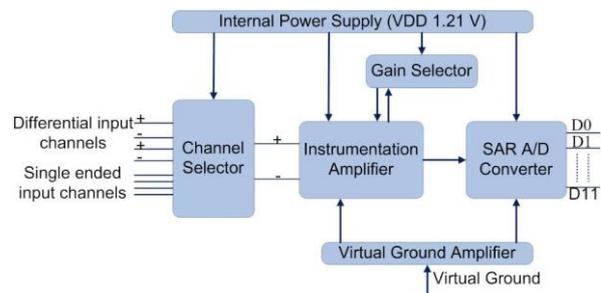
Abbildung 7: Begrenzung des Eingangsspannungsmessbereichs durch Einfügen eines Kondensators  $C_{Offset}$  am Messpunkt der Schaltung.

Abbildung 8: Gesamtschaltung des ADC-Systems.

spannung  $U_{stab} = 1,2 \text{ V}$  gefordert, ergibt sich folgende Kapazität des Offset-Kondensators:

$$C_{Offset} = (U_{stab} - \Delta U_{in}) \div U_{stab} \cdot C_{ges}$$

$$= (1,2 \text{ V} - 1 \text{ V}) \div 1,2 \text{ V} \cdot C_{ges} = 1/6 \cdot C_{ges} = 3,272 \text{ pF}$$

Die Gesamtkapazität des ADC mit  $N = 12$  und  $C_0 = 3,9975 \text{ fF}$  berechnet sich zu:

$$C_{ges} = (2^N - 1) \cdot C_0 = (2^{12} - 1) \cdot 3,9975 \text{ fF}$$

$$= 4095 \cdot 3,9975 \text{ fF} = 16370 \text{ fF}$$

Unter Berücksichtigung von  $C_{Offset}$  ergibt sich rückgerechnet folgender Eingangsspannungsmessbereich von 1 V, wie zuvor auch angenommen:

$$\Delta U_{in} = C_{ges} \div (C_{ges} + C_{Offset}) \cdot U_{stab}$$

$$1 \text{ V} = 16370 \text{ fF} \div (16370 \text{ fF} + 3272 \text{ fF}) \cdot 1,2 \text{ V}$$

Die ideale Gesamtkapazität des ADCs beträgt somit:

$$C_{ges} = (2^N - 1) \cdot C_0 + C_{Offset}$$

$$= 4095 \cdot 3,9975 \text{ fF} + 3272 \text{ fF} = 19642 \text{ fF}$$

Der Offset-Kondensator wird in der realen Schaltung kleiner ausgelegt. Parasitäre Kapazitäten wirken wie die Offsetkapazität, sind also hier abzuziehen. Aus der Layout - Extraktion ergibt sich ein optimaler Wert von 3,250 pF, also 22 fF weniger. In der tatsächlichen Schaltungsauslegung ergibt sich damit die Gesamtkapazität des ADCs zu  $C_{ges} = 19620 \text{ fF}$ .

Unter Berücksichtigung des Offsetkondensators  $C_{Offset}$  sowie der parasitären Kapazitäten  $C_{Para}$  ergibt sich schließlich der Eingangsspannungsbereich  $\Delta U_{in} = 1$  V. Die notwendige Kapazität  $C_{Offset}$  zur Realisierung des reduzierten Eingangsspannungsmessbereichs ist mit in den DAC integriert und am Messpunkt des ADCs angeschlossen.

#### D. Gesamtschaltung des ADC-Systems

Durch Zusammenführen der Schaltung des Digitalteils mit der des Analogteils ergibt sich die Gesamtschaltung des ADC-Systems (Abb. 8). Die Pufferkapazität von  $C_{VDD} = 100$  nF zur Unterstützung der Ausgangsspannung  $U_{stab} = 1,2$  V des LDO-Spannungsreglers ist extern und nicht mit abgebildet. Die aus der Synthese stammende Netzliste des Digitalteils wurde gemeinsam mit der Analogschaltung einer funktionalen Verifikation unterzogen. Im Programm Virtuoso gibt es die Möglichkeit, Digitaltechnik, beschrieben durch Verilog-Code, mit Analogtechnik, beschrieben durch SPICE-Modelle, symbiotisch zu simulieren. Zahlreiche Simulationen durch Stimulation des Systems mit realistischen Eingangssignalen führen zu Ergebnissen, welche das ADC-System bezüglich seiner grundlegenden Merkmale beschreiben.

Von besonderer Bedeutung sind die integrale Nichtlinearität INL [4] und die differentielle Nichtlinearität DNL [5]. Um die gewünschte Wandlungsgenauigkeit zu erhalten, darf der DNL-Wert maximal  $1 U_{LSB}$  groß sein. Bei 12 Bit Auflösung ergeben sich 4095 Stufen, wobei sich die simulierte Übertragungsfunktion (Abb. 9) hier aus 20 gleichmäßig verteilten Testpunkten, welche dann interpoliert wurden, ergibt. Gemessene und ideale Übertragungsfunktion stimmen danach überein. Zur Untersuchung der Auflösung wurde das ADC-System mit einem sehr kleinen veränderlichen differentiellen Eingangssignal stimuliert und der Verlauf der gewandelten digitalen Ausgangsworte  $D[(N-1):0]$  analysiert. Die Verifikation führt zu dem Ergebnis, dass die DNL des ADC-Systems stets kleiner ist als 0,58. Die Simulation ergibt außerdem ein INL kleiner als 1 LSB, was einer unverzerrten Wandlung entspricht. Ein gutes dynamisches Verhalten des ADC-Systems ist damit zu erwarten.

Diese Ergebnisse sind zu erwarten, da in der Simulation praktisch keine parasitären Elemente enthalten sind und als einzige, halb-analoge Komponente der Komparator. Alle übrigen Elemente sind digital und werden deshalb die INL bzw. DNL nicht negativ beeinflussen. Wieweit dies in der Praxis eintritt, wird ein Testchip zeigen. Das Layout der Gesamtschaltung des ADC-Systems ergibt eine kompakte Komponente der Größe  $0,425 \text{ mm}^2$  (Abb. 10).

#### IV. ERGEBNISSE DER ENTWICKLUNG

Ein Ultra-Low-Power-AD-Wandler konnte im Rahmen dieser Arbeit erfolgreich entwickelt werden.

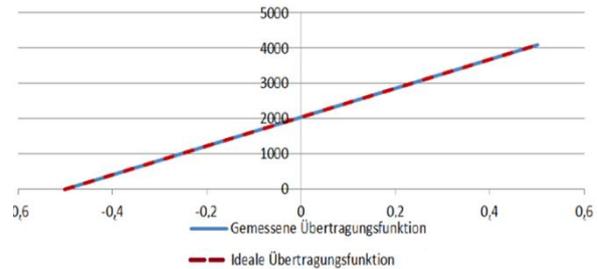


Abbildung 9: Vergleich der simulierten und idealen Übertragungsfunktion des ADC-Systems.

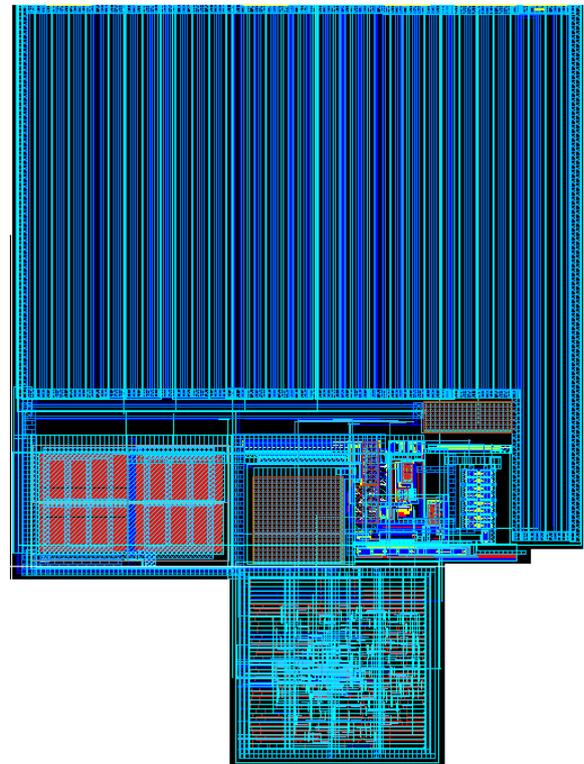


Abbildung 10: Layout des ADCs mit Digitalteil.

Die statische Stromaufnahme des Komparators liegt unter  $1 \mu\text{A}$ . Während des Betriebs ist die Stromaufnahme durch Impulsstrombelastung des Kapazitäten-Arrays bestimmt und damit abhängig von Frequenz und Amplitude des Eingangssignals. Die Impulsstrombelastungen entstehen bei den Auf- bzw. Entladungen der Kondensatoren während der Wandlung. Die Leistungsaufnahme des Digitalteils ist vernachlässigbar. Der größte Teil der elektrischen Verlustleistung des Gesamtsystems rührt von der dem AD-Wandler vorgeschalteten analogen Signalverarbeitung her, die hier nur bezüglich des Instrumentenverstärkers angesprochen werden konnte. Die Tabelle 2 enthält die Daten aller Komponenten, die im gesamten Datenerfassungssystem zum derzeitigen Entwicklungsstand enthalten sind. Hier gibt es zahlreiche

Tabelle 2: Kenndaten des ADC-Systems.

Auflösung	12 Bit
Eingangssignalbandbreite $B$	48,428 kHz
Abtastfrequenz $f(f_{clk\_ext} = 6,78 \text{ MHz})$	96,857 kHz
Eingangskanäle (unipolar/bipolar)	16/8
Versorgungsspannung Digitalteil $V_{DD}$	1,2/1,8 V
Versorgungsspannung $V_{CCext}$	1,2/1,8 V
Versorgungsspannung $U_{stab}$ (intern stabilisiert)	1,2 V
Eingangsspannungsbereich $\Delta U_{in}$ ( $U_{stab} = 1,2 \text{ V}$ )	1 V
Eingangsspannungsbereich $\Delta U_{in}$ ( $U_{stab} = 1,8 \text{ V}$ )	1,5 V
Einstellbare Verstärkung von $U_{in}$	1 – 100
Ausgangssignal $U_{ref}$ ( $U_{stab} = 1,2 \text{ V}$ )	0,6 V
Ausgangssignal $U_{ref}$ ( $U_{stab} = 1,8 \text{ V}$ )	0,9 V
Genauigkeit (ohne Offsetspannung)	10 Bit
Integrale Nichtlinearität INL	< 1 LSB
Differentielle Nichtlinearität DNL	< 0,58 LSB
Stromaufnahme LDO-Regler	780 $\mu\text{A}$
Stromaufnahme Instrumentenverst.	490 $\mu\text{A}$
Stromaufnahme Komparator	77 nA
Stromaufnahme Virt. Ground Verst.	121 $\mu\text{A}$
Stromaufnahme Biasing	38 $\mu\text{A}$
Chipfläche	0,425 mm <sup>2</sup>

Optimierungsmöglichkeiten, an denen noch zu arbeiten ist. Der Stromverbrauch bei deaktiviertem System konnte auf wenige  $\mu\text{A}$  limitiert werden, wodurch im Power-Down-Modus ein beinahe passives System erzielt wird. Die vom AD-Wandler-System benötigte Fläche wurde mit weniger als 0,5 mm<sup>2</sup> sehr klein gehalten.

Beim Vergleich der Kennwerte mit den Kennwerten aus der Referenz kann die vorliegende Arbeit als Fortschritt gewertet werden. Die Auflösung von 12 Bit wird erreicht, die simulierten Werte für INL und DNL sind vielversprechend, allerdings steht noch eine Verifikation durch Messungen am Silizium aus. Die maximale Abtastrate beträgt in dieser Arbeit 50 kHz gegenüber berichteten 6,25 kHz; höhere Raten sind möglich, so die Signalvorverarbeitung das erlaubt. Trotz der um 30 % größeren Strukturen des angesetzten CMOS-Prozesses, erweiterter Funktionalität und

um Faktor vier größeren Kondensatoren des DACs ist der hier entworfene AD-Wandler nur um ca. 20 % größer als der referenzierte. Ein weiterer Vorteil ist auch der große in dieser Arbeit erzielte Eingangsspannungsmessbereich, wodurch nur ungefähr 17 % der maximal möglichen Aussteuerung verloren gehen im Vergleich zu 41 % in der referenzierten Arbeit. Auch hier sind die Messungen am Chip abzuwarten. Die wichtigen Vergleichskriterien Wandlungsgenauigkeit, Flächenbedarf und Eingangssignalbandbreite konnten demnach durch diese Arbeit verbessert werden. Wenn die Messergebnisse an einem Testchip diese Daten bestätigen, steht damit ein wichtiger Baustein für weitere Projekte zur Verfügung.

## V. AUSBLICK

Es verbleibt für die Zukunft noch eine Reihe von Arbeiten. Bei der Schaltungsentwicklung betrifft dies die Entwicklung der Bussteuerungslogik. Das digitale Busprotokoll muss an das Zielsystem angepasst werden, was durch Re-Synthese aus modifiziertem Code leicht möglich ist. Die genannte Entwicklungsarbeit wird im Rahmen der Einbindung des ADC-Systems in das einbettende Gesamtsystem geschehen. Die Post-Layout-Extraktion und -Simulation konnte im zeitlichen Rahmen dieser Arbeit nicht durchgeführt werden. Die Gesamtgenauigkeit und auch der Leistungsverbrauch werden durch die hier nur angedeuteten analogen Signalverarbeitungsblöcke maßgeblich mitbestimmt, deren Entwicklung noch nicht abgeschlossen ist. Schließlich sind die simulierten Werte durch eine Implementierung in Silizium, also die Einbettung des ADC-Systems in ein Test-ASIC und dessen messtechnische Untersuchung zu verifizieren.

## LITERATURVERZEICHNIS

- [1] D. De Venuto, E. Stikvoort, D.T. Castro and Y. Ponomarev, "Ultra low-power 12-bit SAR ADC for RFID applications", *IEEE Design, Automation & Test in Europe Conference & Exhibition (DATE)* 2010, pp. 1071-1075, 2010.
- [2] D. De Venuto, E. Stikvoort, D.T. Castro and Y. Ponomarev, "Low power 12-bit SAR ADC for autonomous wireless sensors network interface", *IEEE Advances in sensors and Interfaces IWASI*, pp. 115-120, 2009.
- [3] D. De Venuto, E. Stikvoort, D.T. Castro and Y. Ponomarev, "Novel low-power 12-bit SAR ADC for RFID tags", *Quality Electronic Design (ISQED)* 2010, pp. 532 – 537, 2010.
- [4] [www.de.wikipedia.org/wiki/Integrale\\_Nichtlinearität](http://www.de.wikipedia.org/wiki/Integrale_Nichtlinearität).
- [5] [www.de.wikipedia.org/wiki/Differenzielle\\_Nichtlinearität](http://www.de.wikipedia.org/wiki/Differenzielle_Nichtlinearität).



# Konzeptstudie eines durchgängig auf parametrisierten Modulgeneratoren basierenden Entwurfsflusses für Analogdesign

Matthias Greif, Daniel Marolt, Jürgen Scheible

**Zusammenfassung**—Im Bereich integrierter Schaltungen (ICs) für die Fahrzeugelektronik ist in den letzten Jahren ein Trend zum Einsatz komplexer Mixed-Signal-Komponenten erkennbar. Dies führt dazu, dass ein altes Problem zunehmend in den Fokus der EDA-Entwickler rückt: Während der digitale Entwurfsfluss hoch automatisiert ist, findet der Entwurf analoger Komponenten überwiegend in einem manuellen, zeitaufwändigen und interaktiven Entwurfsstil statt. Die folgende Arbeit beschreibt ein Konzept, diesen Mangel mit Hilfe eines durchgängigen analogen Entwurfsflusses unter Verwendung so genannter Modul-Generatoren zu mildern. Der vorgestellte Ansatz zur Erzeugung von Schaltkreis-Automatismen berücksichtigt die implizite Nutzung von Erfahrungswissen des Designers, bietet eine volle Topologie-Flexibilität und steigert die Wiederverwendung („re-use“) gängiger Schaltungstopologien. Die erreichten Zwischenergebnisse lassen einen erheblichen Nutzen erkennen und zeigen das Potenzial sogenannter „Parametrisierter Schaltkreise“ auf, den Automatisierungsgrad des analogen Schaltungsentwurfs zu steigern.

**Schlüsselwörter**—Electronic Design Automation EDA, Generatoren, IP, re-use, Expertenwissen, Analog Design, Analoger Entwurfsfluss, Constraints, Dimensionierung.

## I. EINLEITUNG

Electronic Design Automation (EDA) ist im Bereich integrierter Schaltungen nicht weg zu denken. Seit mehreren Dekaden wird – parallel zu den Weiterentwicklungen im Bereich der Technologie – die Funktionalität der Hilfsmittel für den Schaltungsdesigner vorangetrieben. Hierbei hat sich ein erheblicher Unterschied zwischen den Automatisierungsgraden im Bereich von analoger und digitaler Welt ergeben. Mit der Einschränkung von Entwurfsfreiheitsgraden (z.B. Einsatz von Standardzellen und layerspezifische Vorzugs-

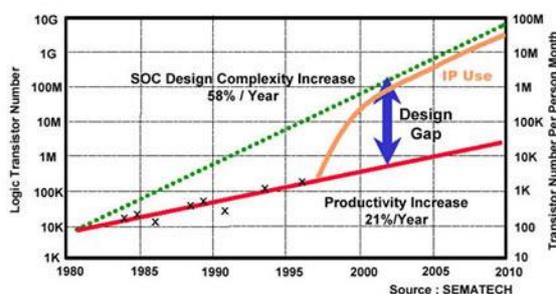


Abbildung 1: Design-Komplexität und -Produktivität über die letzten zwei Dekaden. (Quelle: [http://www.eetimes.com/document.asp?doc\\_id=1279120](http://www.eetimes.com/document.asp?doc_id=1279120)).

richtungen der Verdrahtung) wurde der Automatisierungsgrad im Entwurfsfluss (engl. Design-Flow) von digitalen integrierten Schaltkreisen erheblich gesteigert. Durch die Fokussierung der EDA-Entwickler auf die digitale Welt ist dabei jedoch leider die Entwicklung eines automatisierten analogen Design-Flows in den Hintergrund gerückt. Bis heute ist der analoge IC-Entwurf größtenteils manuell geprägt. Sicher ist ein guter Handentwurf (manueller Entwurfsfluss) – ohne auf die Entwicklungszeit zu achten – einer algorithmisch entworfenen Schaltung vorzuziehen, jedoch spielen Forderungen nach möglichst schnellem Markteintritt („time-to-market“) und dadurch getrieben auch nach einer hohen Wiederverwendbarkeit („re-usability“) von Designs in Zeiten großen Preis- und Konkurrenzdrucks, denen industrielle IC-Unternehmen ausgesetzt sind, eine wichtige Rolle.

Abbildung 1 zeigt die Kluft zwischen der rasch ansteigenden Design-Komplexität und der nachhinkenden Design-Produktivität (Design Gap). Daraus lässt sich schlussfolgern, dass die Entwicklung von EDA-Software für analoge ICs deutlich hinter dem Bedarf der industriellen Anwender zurückbleibt. Nun ist es an der EDA-Community, die Herausforderung anzunehmen, diese Kluft mit Hilfe neuer innovativer Automationsansätze für die Design-Werkzeuge zu verringern. Die bisher verfolgten Ansätze im Analogbereich blieben meist im akademischen Stadium stecken. Als Beispiel kann hier das Optimierungstool OPTIMAN zusammen mit ISAAC, einem grafischen Simulator, genannt werden [1]. Hierbei übernimmt ISSAC die Modellie-

Matthias Greif, Daniel Marolt, Jürgen Scheible, {matthias.greif; daniel.marolt; juergen.scheible}@reutlingen-university.de, Robert Bosch Zentrum für Leistungselektronik, Hochschule Reutlingen, 72768 Reutlingen.

rung der Schaltung, anschließend wird die Dimensionierung des entwickelten Schaltungs-Modells in OPTIMAN mittels eines *simulated annealing*-Verfahrens optimiert. Weitere akademische Beispiele sind unter anderem STAIC [2], oder IDAC [3].

Auch kommerzielle Tools wie zum Beispiel ANALOG INSYDES, ISTONE von IPGEN oder ANACONDA von SAGANTEC werden auf dem Markt angeboten, stoßen aber auf wenig Akzeptanz beim Analog-Designer. Gründe hierfür sind von unterschiedlicher Natur, wie etwa das Misstrauen in Automatisierungsansätze eines fachfremden Programmierers, oder die Abneigung, die bewährte grafisch-manuelle Eingabemethode durch eine kryptische Konsoleneingabe zu ersetzen. Unser Ansatz unterscheidet sich hier von den bisherigen. Bis zum heutigen Zeitpunkt sind die meisten Ansätze zur Automatisierung des Analog-Flows algorithmisch und simulationstechnisch geprägte Optimierungsverfahren für den Designschritt der Dimensionierung einer Schaltung [4]. Andere Ansätze definieren abstrakte Beschreibungssprachen zur Eingabe von hierarchischen Schaltungsbeschreibungen, welche zusammen mit einem symbolisch/numerischen Gleichungssimulator zu „flachen“ Modell-Lösungen, entsprechend der durch den Anwender definierten Topologie- und Performance-Definition, führt [2]. Das hier beschriebene Konzept der parametergesteuerten Generatoren für den Schaltungsentwurf („Frontend“) auf Modulebene bietet dabei einen praktikablen und bisher nicht explizit verfolgten Ansatz, die Design-Effizienz zu erhöhen.

Die vorliegende Arbeit baut sich wie folgt auf: Abschnitt II zeigt den Stand der Technik parametergesteuerter Generatoren und gibt einen Einblick in deren Eigenschaften und Vorteile. Kapitel III gibt eine Übersicht zum Gesamtkonzept parametergesteuerter Schaltplan-Generatoren („Schematic Modul-PCells“) und stellt den Gesamtentwurfsfluss auf Modulebene dar. Hierbei unterscheiden wir zwischen dem Entwurf aus Sicht des PCell-Entwicklers und aus Sicht des späteren Anwenders der erstellten PCell. In IV wird der Entwurfsprozess eines Schaltplan-Modul-Generators aufgezeigt. Anschließend folgt ein Implementierungsbeispiel eines parametrisierten Schaltplans mit einem abschließenden Fazit.

## II. PARAMETERGESTEUERTE GENERATOREN

Mit parametergesteuerten Generatoren lässt sich nicht nur die Design-Effizienz, sondern auch die Qualität des Design-Ergebnisses signifikant steigern. Bei den parametergesteuerten Generatoren handelt es sich um Ablaufskripte, welche auf Grundlage von Expertenwissen erstellt werden und das Vorgehen des menschlichen Designers nachahmen. Das bedeutet, ein Experte – hier auf dem Gebiet des Analog-Designs – nutzt seine Expertise, dafür einen Code-basierten Generator zu erstellen, welcher die Eigenschaft besitzt, das inkludierte Expertenwissen jederzeit und unabhängig von der Designerfahrung des Anwenders reproduzierbar

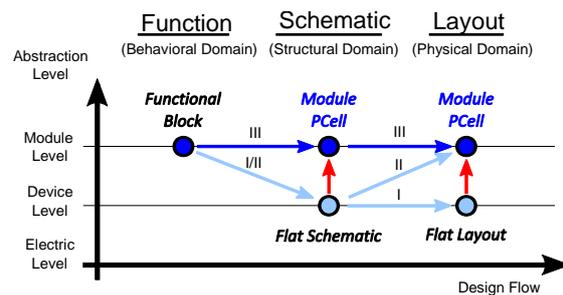


Abbildung 2: Schematic-Driven-Layout (SDL) Design Flow; (I) Konventioneller SDL-Flow; (II) Entwurfsfluss mit Layout-Modul-PCell; (III) Einheitlicher SDL-Flow auf Modulebene.

wiederzugeben. Parametergesteuerte Generatoren sind bereits heute gängiger Gebrauch in der Layout-Erstellung. Ein sehr weit verbreiteter Ansatz ist das sogenannte *PCell*-Konzept des US-amerikanischen EDA-Software-Anbieters CADENCE DESIGN SYSTEMS®, basierend auf der werkzeugspezifischen Skriptsprache SKILL. Andere EDA-Firmen setzen auf allgemein verfügbare Programmiersprachen, wie zum Beispiel SYNOPSIS®, deren sogenannte „PyCells“ in der Open-Source-Sprache Python implementiert werden. Aufgrund der marktbeherrschenden Stellung der Cadence-Werkzeuge im Bereich des analogen IC-Entwurfs hat sich in der Fachsprache der Name PCells als Synonym für parametergesteuerte Generatoren eingebürgert und wird deshalb im Folgenden auch hier verwendet.

Im Layout (häufig auch als „Backend“ bezeichnet) ist die Verwendung von PCells Stand der Technik, allerdings zielt die Anwendung meist auf einzelne parametrisierte Bauelemente ab. Weitere Vorteile des PCell-Konzepts sind:

- Erhöhter Automatisierungsgrad, somit Zeitersparnis bei der Layout-Erstellung
- Reproduzierbare, gute Layout-Ergebnisse
- Inkludiertes Expertenwissen, Ergebnis ist nicht von der Expertise des Benutzers (Anwenders) abhängig
- Ergebnis ist vergleichbar mit gutem Handentwurf (abhängig von der Expertise des PCell-Entwicklers)
- Implizite (verlustfreie) Umsetzung von Constraints (schaltungstechnische Randbedingungen).

Moderne Entwicklungen, wie der PCell Designer bieten dem Layout-Entwickler eine grafische Benutzeroberfläche, welche ihm die Entwicklung von Modul-PCells entscheidend erleichtert, da diese die grafikbasierte Arbeitsweise der Layouter unterstützt und damit ihrer Denkweise entgegenkommt. Als Folge daraus ergibt sich nun die Möglichkeit, das Konzept der parametergesteuerten Generatoren auch auf ganze Module, also komplette Schaltungen oder Schaltungsteile im Layout anzuwenden. Diese Module im Backend enthalten eine Vielzahl an wiederum parametrisierten Bauelementen und deren Verbindungen. Die Weiterentwicklung zur Modul-PCell im Design-Flow integrierter

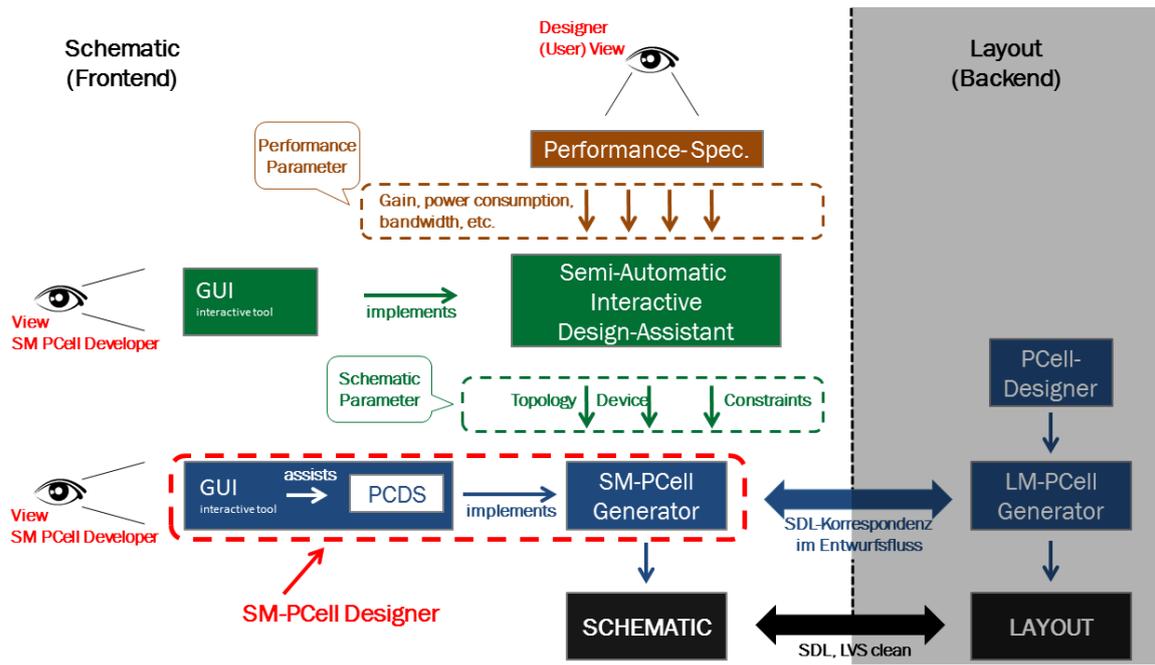


Abbildung 3: Gesamtkonzept Schematic-Modul-PCell aus Designer- (User) und SM-PCell-Entwickler-Sicht.

Schaltungen führt allerdings zu einem Nachteil. Es ergibt sich ein Hierarchiebruch im Designfluss. Nach dem Gajski-Diagramm [5] (auch als Y-Diagramm bekannt) ergeben sich die Abstraktionsebenen Verhalten, Struktur und Geometrie (in Abbildung 2 als *Behavioral*, *Structural* und *Physical Domain* beschrieben). In der Abbildung sind auf der x-Achse die Domänen („Sichten“) und auf der y-Achse die Hierarchiestufen abgetragen. Durch die Einführung einer Layout Modul-PCell (LM-PCell) resultiert ein Wechsel der Hierarchieebenen am Übergang zwischen den Domänen (Abbildung 2: Entwurfsfluss II). Hierarchiebrüche verhindern die Durchgängigkeit des Entwurfsflusses. Eine flache Schaltung auf Strukturebene verhindert die Korrespondenz des SDL-Flows, welche nur innerhalb einer Abstraktionsebene funktioniert. Ziel der Weiterentwicklung ist daher eine grundsätzliche Vermeidung von Hierarchiebrüchen mit dem Ziel eines durchgängigen SDL-Entwurfsflusses auf Modul-Ebene. Hierfür ist für die LM-PCell ein korrespondierendes Design-Element in der Strukturdomäne zu schaffen. Diese besteht in einem parametrisierten Schaltplan, den wir ebenfalls mit dem PCell-Konzept realisieren und daher als Schematic Modul-PCell („SM-PCell“) bezeichnen. Dieser Ansatz ist in Abbildung 2, Entwurfsfluss III dargestellt.

### III. KONZEPT DER SM-PCELL

Ein wichtiger Punkt, die Akzeptanz des Designers gegenüber dem analogen Automationstool zu erhöhen, ist, dem Designer während des gesamten Design Flows die vollständige Kontrolle über die Schaltungsentwicklung zu gewährleisten. Dieser Punkt wurde bei vielen bisherigen Ansätzen nicht explizit berücksichtigt. Ein

nicht-algorithmischer Generator-Ansatz, beruhend auf Expertenwissen, zusammen mit der Einbindung eines Tools zur Generatorentwicklung in eine dem Designer bekannte Entwurfsumgebung führt unserer Meinung nach zu einer höheren Akzeptanz durch den Analog-Entwickler. Der im Folgenden beschriebene methodische Bottom-Up-Ansatz, welcher gleichzeitig eine Erweiterung des bewährten SDL-Flows darstellt, soll den Designer nicht zu einer Abkehr von Altbewährtem bewegen, sondern eine evolutionäre Weiterentwicklung der bisherigen analogen Design-Methodik darstellen.

Das SM-PCell Konzept soll dem Analog-Entwickler ein, seiner Entwurfsmethodik nachempfundenen, Werkzeug zur Schaltungsgenerierung bieten. Es geht dabei nicht um einen weiteren Versuch einer vollautomatischen Schaltungssynthese. Vielmehr soll ein Verfahren entwickelt werden, welches den Entwickler unter Einbeziehung von Expertenwissen unterstützt. Das Gesamtkonzept gliedert sich in zwei verschiedenen Entwurfsflüssen und sollte zwischen den Sichtweisen des PCell-Entwicklers und des PCell-Anwenders differenzieren. Abbildung 3 zeigt die beiden Entwurfsflüsse, inklusive des Backend-Entwurfsflusses mit dargestellter Layout Modul-PCell. (Auf den Layout-Entwurf wird im Rahmen dieses Artikels nicht weiter eingegangen). Horizontal dargestellt ist der Entwurfsfluss für den SM-PCell-Generator aus der Sicht des PCell-Entwicklers. Vertikal dargestellt ist der Entwurfsfluss für die Eingangsparameter der SM-PCell aus Anwendersicht. Beide Entwurfsflüsse enthalten Schaltungsklassen-spezifische Eigenschaften und müssen für jede Schaltungsklasse individuell erstellt werden. Das Konzept begründet zwar einen einmaligen Mehraufwand bei der Erzeugung der spezifischen Komponenten

(Aufgabe des PCell-Entwicklers), ergibt aber einen immensen Vorteil bei der Wiederverwendung (re-use) auf Grund des Generator-Charakters (PCell-Anwender). Im Folgenden sollen die zwei angedachten Entwurfsflüsse aus Anwender- und Entwicklersicht konkretisiert werden.

#### A. Die SM-PCell aus Anwendersicht

Wir betrachten zunächst den Entwurfsfluss aus Sicht des Anwenders der SM-PCell. Dieser nutzt die in Teil B beschriebenen unterstützenden Entwurfswerkzeuge. Allgemein besteht die Aufgabe darin, aus den für den Schaltungsentwickler relevanten Schaltungseigenschaften (im Folgenden als „Performance-Parameter“ bezeichnet und in Abbildung 3 oben dargestellt) die Schaltplanparameter („Schematic-Parameter“) zu generieren. Dies ist gleichbedeutend mit der Übertragung einer Spezifikation in einen elektrischen Schaltplan und stellt damit nichts anderes dar als die originäre Aufgabe der Schaltungsentwicklung selbst. Dabei soll dem Anwender ein interaktives und teilautomatisiertes Werkzeug zur Verfügung stehen (im weiteren Verlauf auch als Design-Assistent bezeichnet). Die aus diesem Entwurfsfluss ermittelten Schematic-Parameter werden an die SM-PCell weitergegeben, welche daraus den fertigen Schaltplan generiert.

Konkret soll der Entwurfsfluss wie folgt aussehen: Der Anwender definiert – auch auf Grundlage der Schaltungsspezifikation – die spezifischen Performance-Parameter, wie zum Beispiel *gain*, *power consumption*, *bandwidth*, uvm.. Diese werden dem Design-Assistenten (in Abbildung 3 als *Semi-Automatic interactive Design-Assistant* bezeichnet) als Eingangswerte übergeben. Der Design-Assistent soll anhand algorithmisch-optimierender Verfahren, unter Einbeziehung des enthaltenen (formalisierten und nicht-formalisierten) Expertenwissens und durch Interaktion mit dem Anwender, diesen dabei unterstützen, die entsprechenden Schaltplanparameter zu Topologie, Dimensionierung und Constraints zu ermitteln. Die Schaltplanparameter werden als Eingangsparameter an den Schaltungsgenerator (SM-PCell) weitergereicht. Der Generator entwickelt daraus den dimensionierten Schaltplan. Zusammengefasst erzeugt der Design-Assistent aus den Performance-Parametern die für die SM-PCell benötigten Schaltplanparametersätze zu Topologie, Bauteildimensionierung und Constraints.

#### B. Die SM-PCell aus Entwicklersicht

Bei der Beschreibung der Vorgehensweise aus Sicht des Anwenders wurden sowohl der Schaltungsgenerator (SM-PCell) als auch der Design-Assistent als gegeben betrachtet. Dies ist – zumindest bei der Erstimplementation der Schaltungsklasse – nicht der Fall. Daher soll im Folgenden der Entwurfsfluss aus Sicht des PCell-Entwicklers näher erläutert werden. Bei der Entwicklung des Schaltungsklassen-spezifischen Design-Assistenten soll das Erfahrungswissen eines Designers

effizient erfasst und verfügbar gemacht werden. Hierfür sollten dem Entwickler unterstützende Methoden angeboten werden, wie etwa interaktive Werkzeuge, deren spezifische Oberfläche der Arbeits- und Denkweise der Schaltungsentwickler so nahe wie möglich kommt (vgl. hierzu Abbildung 3). Je besser dies gelingt, umso eher sind die Designer bereit, mit angebotenen Werkzeugen zu arbeiten und umso umfangreicher wird das im Schaltplangenerator eingebaute Expertenwissen sein. Daneben sollten Möglichkeiten der Automatisierung in Teilschritten soweit als möglich genutzt werden, so dass im Ergebnis ein teilautomatisiertes interaktives Verfahren entsteht, das den Designer in seiner Entwicklungsarbeit unterstützt. Wichtig ist hierbei, den Grad der geplanten Generalisierung nicht zu groß zu wählen, um die Realisierbarkeit des Vorhabens nicht zu gefährden.

Wie bereits erwähnt, ist in Abbildung 3 horizontal der Entwurfsfluss für den Schaltplangenerator (SM-PCell) dargestellt. Der SM-PCell Designer soll in Anlehnung an den von CADENCE® entwickelten PCell Designer eine Designumgebung mit entsprechenden Werkzeugen zur Verfügung stellen, welche den SM-PCell-Entwickler bei der Erstellung der Schaltungsklassenspezifischen SM-PCell unterstützt. Wie in der Darstellung zu erkennen ist, besteht die Designumgebung aus einer grafischen Benutzeroberfläche (GUI) und einer „darunterliegenden“ code-basierten Beschreibungssprache (PCDS) zur Umsetzung der GUI-gestützten Benutzereingaben in den entsprechenden Generator-Code. Grundlage für die Erstellung des Generators bildet die dem Schaltungs-Designer bekannte Schaltplan-Entwurfsumgebung (schematic editor). Die in die GUI des zu entwickelnden SM-PCell Designers eingegebenen Informationen werden im schematic editor als (parametrisierter) Schaltplan wiedergegeben. Die bildliche Darstellung entspricht dem aktuellen Generator-Ergebnis.

Ziel des Konzeptes ist, für jede Schaltungsklasse einen spezifischen Generator bereitzustellen, der auf Basis der Eingangs-Parametersätze (Topologie, Bauteildimensionierung und Constraints) des Design-Assistenten die entsprechende Modul-Instanz (Architecture) ausgibt (vgl. hierzu Abbildung 4). Eine Modul-PCell kann mehrere Schaltungs-Architekturen enthalten, eine Schaltungs-Instanz ist aber nur einer Modul-PCell zugehörig.

Im weiteren Verlauf soll am Beispiel der interaktiven Entwurfsumgebung für den Schaltplangenerator (SM-PCell) eine Funktionalitätsbeschreibung („Use-Model“) vorgestellt werden. Die Entwurfsumgebung für den Design-Assistenten soll eine ähnliche Funktionalität bereitstellen, wird aber an dieser Stelle nicht weiter erläutert. Nach Abbildung 3 unterscheiden wir bei der SM-PCell zwischen den drei Bereichen der Eingangs-Parametersätze Topologie, Bauteildimensionierung und Constraints, welche im Folgenden nun näher beschrieben werden sollen.

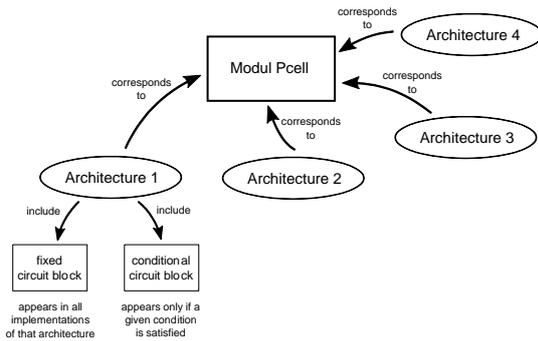


Abbildung 4: Modul-PCell und ihre Instanzen (Architectures).

### 1) Topologie

Jede Schaltungsklasse kann auf eine Grundtopologie zurückgeführt werden. Anhand dieser Grundstruktur entwickeln wir die Freiheitsgrade der PCell und definieren deren Parametersätze und den gültigen Parameterraum. Die Erweiterungen werden mit Hilfe einfacher if/then/else-Funktionen realisiert.

- Definition der Grundstruktur
- Relative Platzierung der Einzelkomponenten, dadurch keine Überlagerung
- Einfacher Wechsel zwischen nMOS und pMOS-Variante (generische Funktion: Auto-Flipping)
- Simple Erweiterbarkeit der Schaltung um z.B.
  - Kaskode- Stromspiegel
  - ENABLE-Funktion (Abbildung 5)
- Setzen von (Layout-)Constraints wie
  - Maching
  - Stromkritische Pfade
  - Leitungsabstände
- Trennung zwischen (Modul-)Symbol und Schaltplan(-Modul)

Als Hilfsmittel stehen neben einfachen Operationen wie das Platzieren einzelner Bauteile auch ganze Bauteil-Module wie etwa Eingangs-, Ausgangs-, oder Differenzstufen-Module, Stromspiegel-Module, etc. zur Verfügung. Die angegebenen Bauteil-Module können ohne Weiteres erweitert werden, Beispiele hierfür findet man unter anderem in [8]. Um eine umfangreiche Bauteil-Modul-Bibliothek übersichtlich zu halten, kann die Generator-Entwurfsumgebung um eine Bibliotheksverwaltung (Library Manager, Browser) inklusive Filter- und Suchfunktion ergänzt werden. Diese Erweiterbarkeit bietet die Möglichkeit, auch unternehmensspezifisches Knowhow in die Modul-Bibliothek des SM-PCell-Entwicklungstools einfließen zu lassen.

Grundsätzlich gilt, dass jeder Designer seinen eigenen Layout-Stil in Bezug auf die Anordnung der einzelnen Bauteilkomponenten eines Schaltplanes (schematic) entwickelt. Bei einer Automatisierung wird dem Anwender meist ein wenig seines Entscheidungsfreiraumes abgenommen. Aus der digitalen Schaltungssynthese sind zum Teil haarsträubende Schaltbilder (Schaltplan-Abbilder) durch Platzierungsalgorithmen

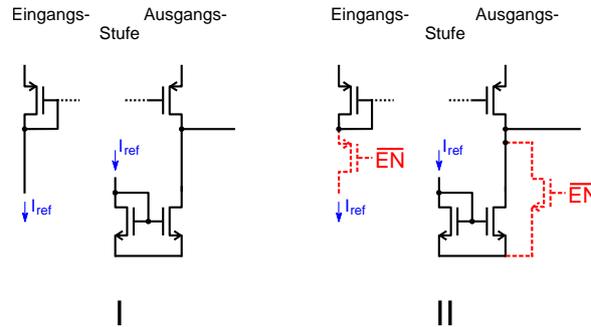


Abbildung 5: ENABLE-Funktionalität.

durchaus bekannt. Da in dem hier beschriebenen Ansatz die Platzierung der Bauelemente nicht mit Hilfe eines Algorithmus erfolgt, sondern von einem erfahrenen Designer vorgegeben wird, ist die Akzeptanz, ein „menschlich“ generiertes Ergebnis anzunehmen, deutlich höher als – im Gegensatz – ein algorithmisch-optimiertes Schaltungs-Design „hinzunehmen“.

### 2) Dimensionierung

Die Dimensionierung der Bauteile erfolgt ausschließlich über die Eingangsinformationen *Bauteile-Dimensionierung* aus dem semi-automatischen Generator (Design-Assistent) des Anwender-Entwurfsflusses (vgl. hierzu nochmals Abbildung 3). Die SM-PCell enthält lediglich Informationen zum gültigen Parameter-Raum, um eine falsche Dimensionierung mittels einer Fehlermeldung abzufangen, sofern diese außerhalb der zulässigen Grenzen liegt.

### 3) Constraint-Handling

Notwendig ist der Einbezug von Layout-Parasiten wie z.B. IR-Drop. Kritische Pfade können mit entsprechenden Leitungswiderständen versehen werden, damit die parasitären Einflüsse bei der Simulation berücksichtigt werden. Im Schaltplan selber sollen diese aber nicht sichtbar sein. Weiter wird der entsprechende Widerstandswert als Randbedingung, oder Constraint (Leitungslänge in DU) an das Layout übergeben und dort beim Ziehen der Leiterbahn interaktiv berücksichtigt. Sollte der Leitungswiderstand bei der Layout-Generierung nun auf Grund der Streckenlänge zu hoch sein, so muss entweder die Leitung verbreitert oder die Leitungslänge verkürzt werden.

Eine weitere nützliche Funktion ist eine Verbindung als Strom-kritische Leitung zu kennzeichnen. Somit kann bei der Simulation die Belastung der Leitung automatisch in die entsprechende Constraint (minimale Leitungsdicke in DU bei entsprechender Strombelastung) umgewandelt und der Leitung angeheftet werden. Auch hier kann bei der Layout-Generierung dieses Constraint explizit miteingebunden und eine Leitungsbreite unter dem gegebenen Wert über eine Fehlermel-

dung abgefangen werden. Die Constraints werden automatisch mit einbezogen und direkt an die entsprechende Layout-PCell weitergereicht. Diese berücksichtigt die entsprechenden Constraints letztlich bei der Layout-Erstellung.

Die vollständige Umsetzung des hier beschriebenen SM-PCell-Konzeptes erfolgt erst nach der Erstellung eines vollständigen Use-Model. Dennoch sollen hier nun beispielhaft einige Eigenschaften aufgelistet werden:

- Hoher Freiheitsgrad
- Mehrere Topologien in einer Modul-PCell
- Freie Transistorauswahl (nMOS, pMOS, NPN, PNP)
- Variable Bauteildimensionierung in einem vordefinierten Parameterraum
- Flatten-Funktion zum Ersetzen der PCell-Instanz durch die entsprechende interne Schaltung
- Differenzierung zwischen Symbol und darunterliegender Schaltung
- Technologieunabhängigkeit
- Einbringung von Expertenwissen
  - Implizit durch die Abbildung von Lösungsstrategien in Generatoren
  - Explizit durch Unterstützung formaler Constraints

#### IV. DESIGN FLOW EINER SM-PCELL

In Abbildung 6 ist schematisch der Entwurfsfluss (Design Flow) einer Schematic-Modul-PCell dargestellt. Wie bereits beschrieben ist die Generator-Umsetzung Schaltungsklassen-spezifisch, daher müssen die im Folgenden beschriebenen Punkte für jede Schaltungsklasse, welche neu als PCell entwickelt werden soll, durchgeführt werden. Somit erfolgt im ersten Schritt zunächst die Festlegung bzw. die Definition der Schaltungsklasse. Anschließend findet die Exploration der Schaltungsklasse statt, bei der alle charakterisierenden Kenngrößen (Performance-Parameter) ermittelt werden und der entsprechende Parameter Raum abgesteckt wird. Es können – praktisch gesehen – nicht alle bestehenden Schaltungstopologien mit Hilfe des hier vorgestellten Generator-Konzeptes abgebildet werden, da der Entwurfsaufwand einfach zu hoch wäre, oder anders gesagt eine Umsetzung aller erdenklichen Schaltungen unmöglich ist. So muss als Grundvoraussetzung eine Auswahl von Topologien erfolgen, welche eine ausreichend hohe Wiederverwendbarkeit (reuse-Faktor) aufweisen [9]. Diese lassen sich dann mittels Kenngrößen charakterisieren (Beispiele solcher Kenngrößen für OTA-Schaltungen finden sich in [5]). Damit sind die einzelnen Schaltungstopologien bekannt und können erstellt werden. Dies führt zum Aufziehen des gültigen Performance-Parameter-Raumes und zur Definition des Device-Parameter-Raums. Bei der anschließenden Exploration der separaten Schaltungstopolo-

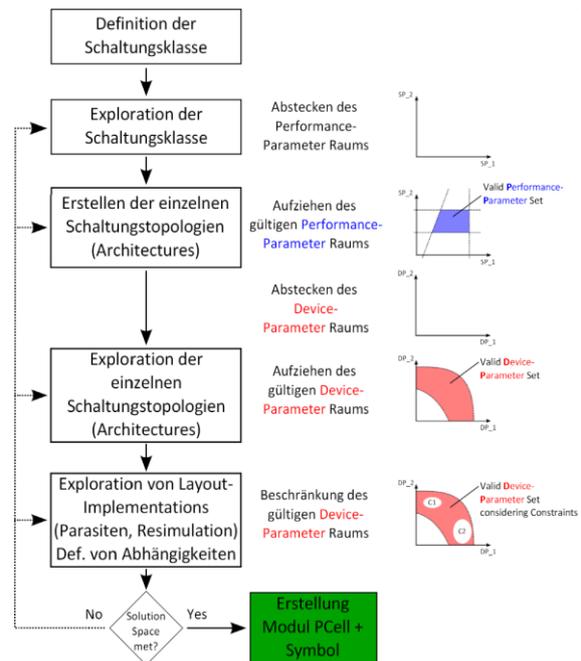


Abbildung 6: Design Flow einer SM-PCell.

gien wird dann der gültige Device-Parameter-Raum bestimmt. Dieser wird im letzten Schritt durch Randbedingungen (Constraints) und die Berücksichtigung von Layout-Parasiten weiter eingeschränkt. Entsprechen die nun gefundenen gültigen Performance- und Device-Parameter-Räume dem geforderten Lösungsraum, werden diese Informationen als Eingangswerte sowohl für den Entwurf des Design-Assistenten (Semi-Automatic interactive Design-Assistant), als auch für die SM-PCell verwendet.

#### V. BEISPIELE PARAMETRISierter SCHALTUNGEN

Das Konzept der SM-PCell wurde zwar noch nicht vollständig umgesetzt, jedoch bestehen bereits einige grundlegende Ansätze. Erste Entwicklungen, den Modul-Charakter auch auf das Frontend (Schaltplan, schematic) zu übertragen, waren z.B. eine automatische Strukturerkennung und ein parametergesteuerter Topologie-Generator, welcher für die OTA-Schaltungsklasse implementiert wurde [9].

Die automatische Strukturerkennung zielt darauf ab, die Teilschaltungen eines Schaltplanes zu gruppieren und von Device-Ebene auf Modul-Ebene zu bringen, ohne die bereits beschriebenen Module erzeugen zu müssen, sondern diese anhand eines Algorithmus eigen und automatisch erkennen zu lassen. Implementiert wurde hierbei eine Erkennung für Stromspiegel, welche die detektierten Stromspiegel mit einer entsprechenden Layout-PCell eines parametrisierten Stromspiegel-Layouts verknüpft.

## VI. FAZIT

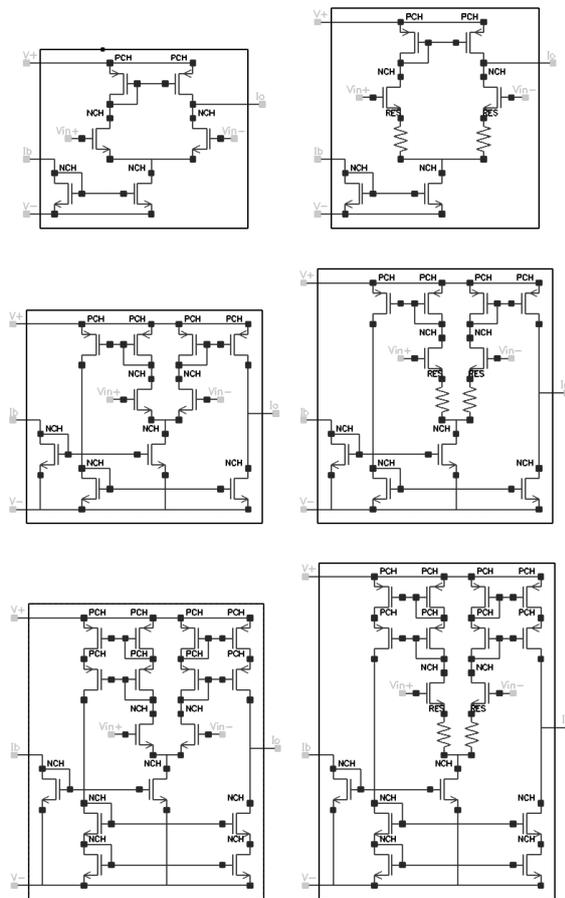


Abbildung 7: Verschiedene Architekturen (Topologien) einer Schematic Modul-PCell.

Abbildung 7 zeigt sechs verschiedene Varianten einer OTA-PCell mit nMOS Eingang, welche mit Hilfe des tools *PCDS* [11] erstellt wurde. *PCDS* ist eine Entwurfsumgebung, welche unter Verwendung einer Metasprache Hilfswerkzeuge zur Erstellung eines parametergesteuerten Schaltplan-Generators bietet. Dem Entwickler wird dabei eine grafische Benutzeroberfläche zur Parameterdefinition und Code-Eingabe zur Verfügung gestellt. Implementiert wurde dieses PCell-Beispiel in der von CADENCE® angebotenen Entwurfsumgebung *Design Framework II (DFII)*. Der Fokus wurde hier auf die Topologie-Auswahl gelegt, die Dimensionierung der Schaltung wurde dabei nicht berücksichtigt. Die Modul-PCell erzeugt – parametergesteuert – verschiedene Topologien (Instanzen), die Auswahl erfolgt über das Property-Menü. Die Implementierung der exemplarisch dargestellten PCell erfolgte mit knapp 80 Zeilen *PCDS*-Code, was übersetzt über 1800 Zeilen *SKILL*-Code ergibt. Die erhebliche Reduzierung der zu implementierenden Code-Zeilen zur Erstellung der PCell zeigt eindrucksvoll den Nutzen eines höheren Automationsgrades und die Notwendigkeit, den Analog-Flow auf höherer Abstraktionsebene zu realisieren.

In der vorliegenden Arbeit wurde ein Konzept eines durchgängig auf parametrisierten Modulgeneratoren basierenden Entwurfsflusses für Analogschaltungen vorgestellt. Hierbei ist es wichtig, die Sichtweise des PCell-Entwicklers und des PCell-Anwenders zu unterscheiden. Nach dem hier gezeigten Vorgehen sollen dem Entwickler bereits unterstützende, interaktive Entwurfswerkzeuge zur Verfügung stehen, mit Hilfe derer er einen Design-Assistenten und einen Schaltplangenerator für den Anwender implementiert. Design-Assistent und Schaltplangenerator ermöglichen dem Anwender einen Werkzeug-unterstützten Entwurfsfluss mit einfacher Wiederverwendung des durch den Entwickler eingebundenen Expertenwissens. Zwar begründet das Konzept einen einmaligen Mehraufwand bei der Erstellung der Schaltungsklassen-spezifischen Werkzeuge, der Generatorcharakter stellt jedoch bei der Wiederverwendung einen erheblichen Vorteil da. Weitere Vorteile sind:

- Hierarchische Konsistenz, d.h. einheitlicher SDL-Flow, ohne Hierarchiewechsel. Damit wird Informationsverlust vermieden.
- Reduzierter Design Aufwand durch einfaches Instanzieren einer PCell, ohne manuelle Schaltungserzeugung oder fehleranfälliges Copy-und-Paste.
- Einheitlicher „Layout-Stil“ in Bezug auf die Anordnung der Bauteile und deren Konnektivität. Dies erhöht die Lesbarkeit der Schaltungsdesigns und damit die Akzeptanz.
- Keine Abweichung von der bisherigen Entwurfsmethodik durch die Einbindung der tools in einer dem Designer vertrauten Entwurfs-Umgebung. Damit wird ebenfalls die Akzeptanz erhöht.

## DANKSAGUNG

Diese Arbeit ist Teil eines Projektes, welches durch das Bundesministerium für Bildung und Forschung (BMBF) gefördert ist.

## LITERATURVERZEICHNIS

- [1] G. Gielen, H. Walscharts, and W. Sansen, “Analog circuit design optimization based on symbolic simulation and simulated annealing,” *IEEE Journal of Solid-State Circuits*, vol. 25, no. 3, pp. 707–713, Jun 1990.
- [2] J. Harvey, M. Elmasry, and B. Leung, “Staic: an interactive framework for synthesizing cmos and bimos analog circuits,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 11, no. 11, pp. 1402–1417, Nov 1992.
- [3] M. Degrauwe, O. Nys, E. Dijkstra, J. Rijmenants, S. Bitz, B. L. A. G. Goffart, E. Vittoz, S. Cserveny, C. Meixenberger, G. Van Der Stappen, and H. Oguey, “Idac: an interactive design tool for analog cmos circuits,” *IEEE Journal of Solid-State Circuits*, vol. 22, no. 6, pp. 1106–1116, Dec 1987.
- [4] G. G. E. Gielen and R. A. Rutenbar, “Computer-aided design of analog and mixed-signal integrated circuits,” *Proceedings of the IEEE*, vol. 88, no. 12, pp. 1825–1852, Dec. 2000.

- [5] D.D. Gajski, "The Structure of a Silicon Compiler", *Proceedings of IEEE ICCD*, 1987, pp. 272-276.
- [6] A. Gerlach, M. Junge, and J. Scheible, "Universal OTA Test Bench," *Workshop der ASIM/GI-Fachgruppe*, Robert Bosch Center for Powerelectronics, Reutlingen, Germany, Feb. 2014, pp. 83–87.
- [7] J. Crossley, A. Puggelli, H.-P. Le et al., "BAG: A designer-oriented integrated framework for the development of AMS circuit generators", *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pp. 74–81, 2013.
- [8] M. Eick, "Structure and signal path analysis for analog and digital circuits," *Ph.D. dissertation*, Technical University of Munich; Department of Electrical Engineering and Information Technology; Institute for Electronic Design Automation, 2013.
- [9] A. Gerlach, M. Junge, J. Scheible, T. Rosahl, "Optimierte, wiederverwendbare OTA-Schaltungen für moderne Power BiCMOS-Technologien", MPC14, 52. Fachtagung der Multi Projekt Chip Gruppe, Künzelsau, Germany, July 2014.
- [10] D. Marolt, J. Scheible, G. Jerke, "The Application of Layout Module Generators upon Circuit Structure Recognition", *CDNLive EMEA 2011*, Munich, Germany, 2011
- [11] D. Marolt, M. Greif, J. Scheible, and G. Jerke, "Peds: A new approach for the development of circuit generators in analog ic design," *22nd Austrian Workshop on Microelectronics (Austrochip)*, Oct 2014, pp. 1–6.



Matthias Greif erhielt den akademischen Grad des B. Eng. in Mechatronik im Jahr 2011 von der Hochschule Reutlingen und den Grad des M. Sc. in Leistungs- und Mikroelektronik im Jahr 2013 von der Hochschule Reutlingen. Er ist wissenschaftlicher Mitarbeiter am Robert Bosch Zentrum für Leistungselektronik (rbz) der Hochschule Reutlingen und befasst sich mit dem Thema Electronic Design Automation (EDA).



Daniel Marolt erhielt den akademischen Grad des B. Eng. in Mechatronik im Jahr 2008 von der Hochschule Reutlingen und den Grad des M. Sc. in Mechatronik im Jahr 2009 von der Hochschule Reutlingen. Er ist wissenschaftlicher Mitarbeiter am Robert Bosch Zentrum für Leistungselektronik (rbz) der Hochschule Reutlingen und befasst sich mit dem Thema Electronic Design Automation (EDA).



Jürgen Scheible erhielt den akademischen Grad des Dipl.-Ing. in Elektrotechnik im Jahr 1987 von der Universität (TH) Karlsruhe und den Grad des Dr.-Ing. in Elektrotechnik im Jahr 1991 von der Universität (TH) Karlsruhe. Er ist Professor für Electronic Design Automation (EDA) am Robert Bosch Zentrum für Leistungselektronik (rbz) an der Hochschule Reutlingen.

# Hardware to Software Migration and Optimization of a Serial Peripheral Interface

Marvin Freier, Kai Schulz, Axel Wenzler, Thomas Mayer, Joachim Gerlach, Wolfgang Rosenstiel

**Abstract**—Today’s sensor ASICs commonly handle signals from various sensors and provide the processed data over several interfaces. A critical issue while designing these interfaces is the hardware/software partitioning. Based on the experience of the designer or characteristic numbers in data sheets, the interfaces are allocated to a certain domain. Besides of the interface requirements, the allocation is driven by constraints such as the availability of a microprocessor, the feasibility of code reuse or the reduction of power consumption. This Paper shows the migration of an existing serial peripheral interface (SPI) taken from an industrial automotive sensor ASIC. Thereby the existing pure hardware solution is migrated to an optimized mixed hardware/software solution. Due to its strict timing requirements, an application specific SPI is mostly realized as hardware. In this paper, we present software-based mechanisms, which enable to accelerate the internal processing as well as to meet the timing requirements and thus, make a software-based solution feasible. The software alternatives are compared to the existing hardware realization based on a system-level model. This multi-abstraction level model is described in SystemC. The results of the exploration study are used in the architecture-finding phase of the next generation of a multi-channel sensor ASIC.

**Index Terms**—ASIC design, hardware/software partitioning, system exploration, SystemC.

## I. INTRODUCTION

Current sensor ASIC designs are driven by the integration of additional functionality, the increasing complexity and a tight time-to-market schedule to stay competitive. Several approaches have been established to design these heterogeneous systems with components from different domains, such as analog hard-

ware, digital hardware and software. Using a top-down approach, a designer starts on a high abstraction layer. This way a designer describes the fundamental behavior on system level according to the ASIC specification. A system-level model also simplifies the examination of the impact on the overall system when a module is integrated. Examples of such modules are intellectual property (IP) modules or already implemented components. Compared to a static examination, a high-level simulation enables an easier detection of unexpected dynamic behavior. During the development the system-level model is gradually extended and refined to a description in a domain-specific hardware description language [1].

A top-down approach is based on design topics which are decided in an early phase. These decisions formatively impact the fundamental system behavior and the resulting overall costs. By a high-level exploration time-consuming design iterations and additional costs of non-optimal solutions can be avoided or at least reduced. However, high-level explorations have to cope with the lack of relevant information which is caused by the abstract system description and the systematic masking of details for faster simulations.

The HW/SW partitioning of interfaces is a typical design topic which is primarily addressed in an early phase. In this paper we focus on alternative mixed HW/SW implementations of an existing serial peripheral interface (SPI) slave originating from an automotive sensor ASIC. Primarily an SPI is utilized for communication with an external microcontroller ( $\mu\text{C}$ ) to request sensor data and to set configuration parameters [2]. In current sensor ASICs, the SPI is usually realized as full-custom hardware to meet the strict timing requirements. In practical use the SPI includes an optimized protocol definition which slightly differs from project to project. Thus, the SPI has a high application-specific implementation and the reuse is only feasible with high design effort. Because of that the usage of IP is often not possible and the SPI has to be redeveloped for each project. In contrast, a software implementation would offer an improved reusability, a shortened development time as well as a higher flexibility. Possible drawbacks are higher power consumption and a more time-consuming internal processing. Furthermore a software implementation could be delayed by interrupts triggered by other applications

M. Freier, A. Wenzler and T. Mayer, {marvin.freier, axel.wenzler, thomas.mayer3}@de.bosch.com, are with Robert Bosch GmbH, P.O.B. 1342, 72703 Reutlingen.

K. Schulz and J. Gerlach, {schulzka, gerlach}@hs-albsig.de, are with Albstadt-Sigmaringen University, Faculty Engineering, Poststrasse 6, 72458 Albstadt.

W. Rosenstiel, rosenstiel@informatik.uni-tuebingen.de, Eberhard Karls University of Tübingen, Department of Computer Science, Sand 13, 72076 Tübingen.

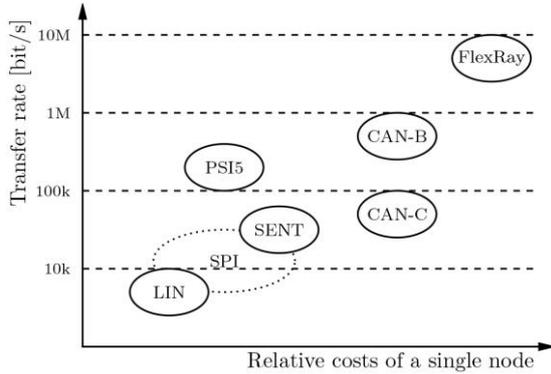


Figure 1: Overview of automotive digital interfaces.

running on the same microprocessor, e.g. applications for safety or maintenance. Additionally only few sensor ASICs provide a microprocessor. To benefit from the mentioned software advantages we first migrate an existing hardware SPI to software. In a second step we implement mechanisms to accelerate the internal processing to meet the timing requirements. Thereby the definition of the underlying SPI communication protocol has to stay unmodified. The results of the high-level exploration contribute to an industrial study.

The paper is organized as follows: Section II gives an overview of typical automotive digital interfaces in ASICs. In Section III the existing hardware reference realization and the developed mixed HW/SW alternatives are described. The experimental results from these scenarios and the comparison to the reference realization are discussed in Section IV. Section V shows related work. The conclusion is given in Section VI.

## II. AUTOMOTIVE DIGITAL INTERFACES

Compared to interfaces for consumer ASICs, automotive digital interfaces have much higher requirements regarding data integrity and data availability [3]. Typically, sensor ASICs have various interfaces to serve multiple applications and to be accessible by different overlying electronic control units. Examples for such interfaces are LIN (local interconnect network) [4], SENT (single edge nibble transmission) [5], PSI5 (peripheral sensor interface) [6], CAN (controller area network) [7] or SPI (serial peripheral interface) [8]. These interfaces are generally characterized by bandwidth, latency, transfer rate and implementation costs of each node. Fig. 1, derived from [3], shows the ratio of transfer rate to relative costs of a single node for each interface type. The SPI is not represented in [3] and can be arranged between LIN and SENT.

Interfaces can be implemented in hardware and in software running on a microprocessor as well. A hardware realization is required if strict timing constraints need to be fulfilled – for example, SPI is typically allocated to digital hardware. Other interfaces,

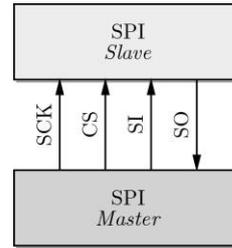


Figure 2: SPI master and SPI slave connections.

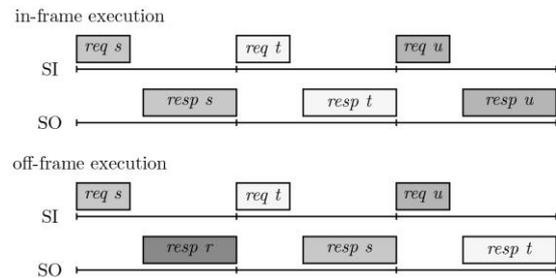


Figure 3: In-frame and off-frame execution.

usually CAN or PSI5, are realized in software, which allows a higher flexibility or better code reusability. An ASIC with an adequate HW/SW partitioning can benefit from both characteristics of the hardware modules and software functions.

### A. SPI Characteristics

An SPI master from an external  $\mu\text{C}$  and an SPI slave are connected with four wires (see Fig. 2). SCK abbreviates “Serial Clock” and sends the master clock signal to the slave. CS stands for “Chip Select” and is used to address a specific slave. The data is transferred over the unidirectional wires SI (Slave In) and SO (Slave Out). The labeling of the wires is not standardized, but the four wires are identically used in different applications for the same functionality.

An important SPI characteristic is the word width of an SPI frame. In automotive sensor ASICs, the SPI frame is usually configurable and relevant word widths are 16, 32 or 64 bit. The specific characteristics can be classified into implementation and definition characteristics. Implementation characteristics are strongly influenced by the allocated hardware or software resources, such as the crucial factor between the system clock of the ASIC and the SPI master module clock (SCK in Fig. 2). Mostly, the SPI is implemented in hardware to avoid undesired software interrupts and to achieve a predictable timing behavior. Definition characteristics depend on the application and have to be adapted or reengineered in other projects. An example is the in-frame or off-frame execution of a request. In case of an in-frame execution, the corresponding response to a request is sent within the same SPI frame. Off-frame execution means that the

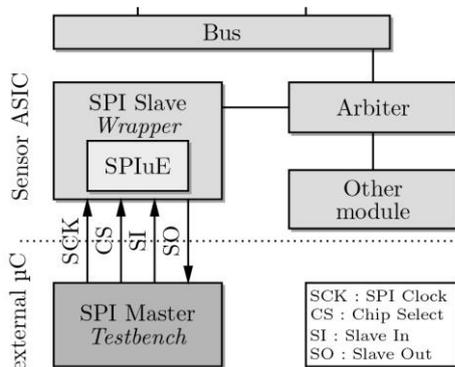


Figure 4: System-level model with relevant parts.

response is transmitted in the subsequent frame. Fig. 3 visualizes the in-frame and off-frame execution in a simplified manner. *req* contains the address for the requested data which is responded in *resp*.

Furthermore, the protocol’s adaption enables the realization of additional features like a cyclic redundancy check (CRC) or a paging mechanism which extends the address space. The realization of additional features also spreads the range of feasible SPI implementations.

### III. REFERENCE IMPLEMENTATION AND SCENARIOS

In this paper, we examine mixed HW/SW alternatives to benefit from the advantages of such implementations, e.g. higher flexibility and easier code reuse. This section describes the existing SPI hardware realization of a current ASIC and the derived software scenarios. For the comparison, the hardware realization is used as reference. The examination takes place in a system-level model which is described in SystemC [9]. Fig. 4 shows the relevant parts of the system-level model. The different SPI implementations are integrated as “SPIuE” (SPI under Exploration) into the “SPI Slave Wrapper” to compare the exploration results. The “SPI Slave Wrapper” is mainly responsible for the communication. In particular for the translation between the TLM (transaction-level modeling, [9]) and RTL communication (register-transfer level). According to Fig. 2, the “SPI Slave Wrapper” is connected via four wires with the SPI master (SPI Master Testbench). The “SPI Master Testbench” is exclusively used to trigger typical test cases from the industrial practice. The other modules are necessary for the SPI on-chip communication, reading register values and writing values to the registers respectively.

The software alternatives run on an in-house developed processor. The word width of the processor is limited to 24 bits. A cycle-accurate SystemC module of the hardware realization exists and the module is integrated into the system-level model as “SPIuE”. The software code has to be written in assembly lan-

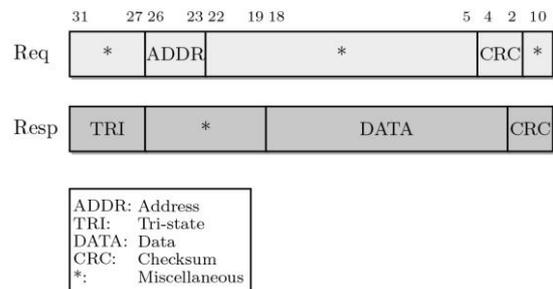


Figure 5: Basic 32-bit in-frame protocol.

guage. During initialization of the system-level model, the assembled code has to be loaded into the memory of the processor module.

As a substitute for the SystemC module of the in-house processor the software scenario could be connected via host-code execution which enables C as programming language and facilitates the implementation [10]. However the effort to reach the necessary timing accuracy based on host-code execution is much higher compared to the chosen approach.

#### A. “HW scenario 0”: Integration of RTL Realization

The existing hardware realization originates from an industrial sensor ASIC and is described in VHDL at RTL. The SPI module serves several applications, supports 32-bit as well as 64-bit in-frame and off-frame execution. The module comes with additional features, like a paging mechanism. The SPI is mainly used to request processed sensor data and to set values in the configuration module. In Fig. 5 a read request of the most time-critical protocol (32-bit in-frame) is shown. The SPI master sequentially sends the single bits of the request frame (Req) according to the defined SPI master module clock. The SPI slave is clocked at the ASIC’s system clock. After receiving the address bits from the request frame (bit 22), the SPI slave has to evaluate the address, collect the requested data from the sensor, prepare the response (Resp) and send the response starting at bit 18. The ratio of system clock to SPI master module clock has to be factored into the calculation of the available system cycles for the processing of the SPI slave. Therefore the ratio has to be multiplied with the number of bits between the end of the address (Req) and the start of the responded data (bit 22 to bit 19) in order to calculate the number of corresponding system cycles. In the regarded industrial application, the factor ranges from 2 to 2.5, where the system clock of the ASIC is faster than the SPI master module clock.

For the evaluation, the VHDL module is transformed into a SystemC module using the described approach in [11]. The resulting SystemC module enables a convincing comparison between the hardware description and the software scenarios.

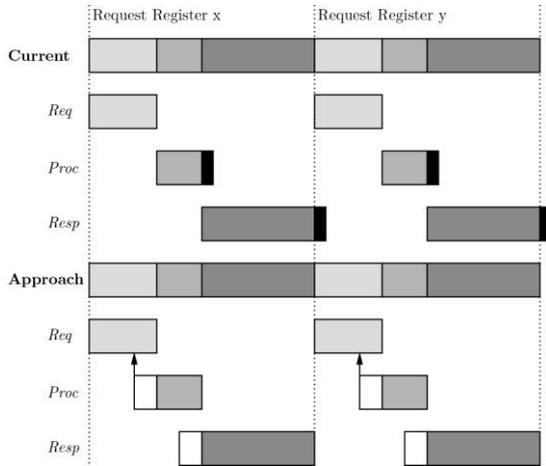


Figure 6: Time-saving through defined request order.

The approach presented in [11] preserves the readability, configurability and the accuracy of the original RTL description. After the transformation, the SystemC module is integrated into the “SPI Slave Wrapper” as “SPIuE” (Fig. 4).

#### B. “SW Scenario 1”: SW-Implementation of RTL-Realization

In “SW scenario 1” the functionality of the hardware description is identically mapped onto the modified SPI slave module. This scenario represents the initial situation for all other mixed HW/SW alternatives and defines the consistent HW/SW partitioning. All SPI submodules are implemented in software except two input/output shift registers and a CRC module. The shift registers allow the in-house processor to read (or write) several bits at once. The CRC module stays in HW enabling the usage of a generic CRC module (as IP). The software parts run exclusively on the processor to avoid delays provoked by interrupts. Such interrupts in automotive sensor ASICs could be triggered by periodic functions, e.g. for safety or maintenance. Furthermore in this scenario, there are no additional mechanisms applied to improve the protocol’s timing behavior. The focus of this implementation is on the pure functionality mapping onto software.

#### C. “SW-Scenario 2”: Prefetching SPI

In this software scenario, the SPI slave bypasses the time-consuming path over several modules, e.g. internal buses or bridges, during processing the request frame. The internal fetching is done earlier and independent of the SPI master requests. Thus the impact of a slow bus communication is reduced. Delays caused by high bus traffic can be avoided as well. In “SW scenario 2” the SPI slave continuously fetches register values and buffers them in the processor memory. If the SPI master requests a register, the SPI slave decodes the register address from the request frame and

responses the value from the mapped memory cell. Probable drawbacks of this approach are higher memory consumption for the processor, more bus traffic and higher power consumption. Another reasonable problem is how to assure the currentness of the sensor data (dead time).

#### D. “SW-Scenario 3”: SPI-Frames in a Defined Request Order

The last software scenario extends the initial situation to improve the timing behavior by means of a predefined frame processing order. The analysis of the use cases from the multi-channel sensor ASIC reveals that the most time-critical 32-bit in-frame protocol is used to request register values in pattern-based sequences. In the customized “SW scenario 3”, we take advantage of these patterns and adapt the frame processing of the SPI slave. The functionality of a random access has to be limited as well. Fig. 6 compares the current use with the suggested software scenario. The figure also emphasizes potential delays and time-savings.

In Fig. 6 the possible delays are marked black and potential time-savings are colored white. Delays generally occur if the SPI slave is blocked and the beginning of the processing (or responding) is delayed. Also the required time for processing or responding could overrun the allocated time. In “SW scenario 3” the available time is increased by an earlier start of the processing. Once an SPI request frame has been recognized, the SPI starts the processing of the request. The SPI slave passes through the defined order, gets the actual register address and has to mark the new position. Thereby the slave saves time by avoiding the address decoding of the request frame.

One drawback is the alleged loss of a random access. An entire random access only copes with the loss of the potential time-savings based on the mechanism of a defined request order. It’s yet possible to configure the request order enabling a random access partially. Another drawback could be the remaining dependence on the ASIC internal communication.

## IV. EXPERIMENTAL RESULTS

The reference and described software alternatives are implemented in SystemC and integrated into the system-level model as SPIuE according to Fig. 4. A scenario is selectable by setting a preprocessor directive. Thereby, both modules “SPI Master Testbench” and “SPI Slave Wrapper” don’t have to be modified and the comparison of the scenarios bases on equal environmental constraints. The “SPI Master Testbench” module contains test cases from the existing VHDL realization for the modes “SPI 64-bit off-frame” and “SPI 32-bit in-frame”. Both test cases have the same structure. At the beginning, a calibration command is sent. In the next phase values are written

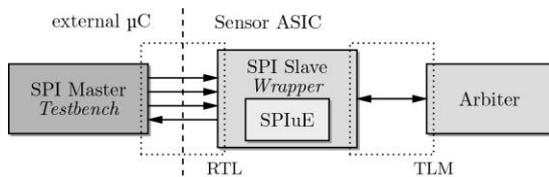


Figure 7: Abstraction levels of the inter-module communication.

to module registers (write requests) and finally register values are requested (read requests). As well as the scenarios, the SPI modes are selectable by a preprocessor directive.

As shown in Fig. 7 the communication between “SPI master Testbench” and “SPI Slave Wrapper” happens at RTL and is less abstract than the inter-module communication between “SPI Slave Wrapper” and “Arbitrer”. Thus the signals on the SPI wires can be traced and the resulting waveforms can be compared bitwise to each other (also with the original VHDL waveform). The most important advantage is the exact identification of delays. Inside the system-level model, the inter-module communication is described with the TLM approach to keep the simulation fast. Potential delays within the system-level model, e.g. caused by high bus traffic, are currently faded out. The exploration focuses on the required internal time of the software alternatives to process SPI Master requests in comparison to the existing hardware solution.

All software alternatives run exclusively on an in-house processor and are implemented in assembly language. The assembler generates two header files which are necessary to be included into the system-level model. The first header file contains an array with the instructions in hexadecimal format code. During the initialization phase, this array is loaded into the processor memory. The other header file defines processor characteristics, amongst others the offset and the memory size.

#### A. HW scenario 0

As reference for the further experiments, the existing VHDL realization is transformed into a SystemC module using the approach in [11]. The resulting module is timing accurate and shows the identical behavior at its ports. In the next step, the SPI module is integrated into the “SPI Slave Wrapper” (see Fig. 4) and the ports SI and SO are directly connected to the corresponding ports of the generated SPI module. In addition, two methods are realized to create outgoing TLM requests for the internal communication of the ASIC and vice versa to unpack incoming TLM packages. The “SPI Slave Wrapper” is verified by comparing the waveforms (32-bit in-frame and 64-bit off-frame) from the SystemC simulation with the original waveforms from the VHDL simulation. Fig. 8 opposes the identical waveforms from a 64-bit off-frame test case, where Fig. 8(a) shows the original waveform and

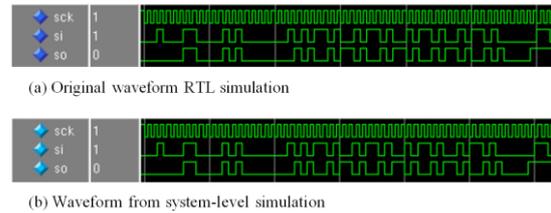


Figure 8: Comparison between RTL and TLM waveform.

Fig. 8(b) presents the results of the system-level simulation. There are also no deviations in the other test cases, independent of a 32-bit in-frame or 64-bit off-frame execution.

#### B. SW scenario 1

In “SW scenario 1”, the protocol-specific tasks are implemented in software, such as the address decode logic for the requested address. Only two input/output shift registers and a submodule for determining the CRC stay in hardware. The shift registers are deployed to read (or write) several bits by the processor and improve the required processing time. Retaining the CRC module in hardware offers the usage of a generic CRC (realized as IP). While mapping the SPI functionality onto software, there are no modifications of the protocol.

#### C. SW scenario 2

The “SW scenario 2” is based on the premise that read requests for sensor data are the primary use of the 32-bit in-frame protocol. In a loop, the modified SPI slave obtains continuously sensor data from their registers and buffers them in the processor memory. For fetching, the addresses of the registers have to be defined in the memory as well. An incoming request triggers an interrupt and the processor starts the decoding of the address. The modified address decode logic refers to the corresponding memory cell of the processor. Once the response is finished, the processor proceeds and fetches sensor data until the next incoming request. The number of requested data affects strongly the dead time and the memory consumption. An increasing number of data extends the time needed to run through the loop and to consequently update the cached data.

#### D. SW scenario 3

The “SW scenario 3”, worked out in two alternative variants, makes use of the fact that the time-critical 32-bit in-frame protocol is mostly used to request sensor data in a repeating sequence. The decoded addresses are defined in internal registers of the processor (in variant 1) according to their request order. Once a request is detected, the processor is triggered by an interrupt signal. Because of the defined order

Table 1: Comparison between the scenarios.

<i>Realization</i>	<i>Resource</i>	<i>Frame Error</i>	<i>#Bits</i>	<i>Min. Ratio</i>
HW scenario 0	HW	no	4	2.0
SW scenario 1	HW+SW	yes	4	4.0
SW scenario 2	HW+SW	yes	4	3.0
SW scenario 3 (Var. 1)	HW+SW	yes	8	2.0
SW scenario 3 (Var. 2)	HW+SW	yes	8	2.0

the processor is already aware of the decoded address from the corresponding register and starts immediately with the on-chip communication. An adaption of the requested addresses and consequently the configuration of the request order are possible. If the processor-internal registers are used, the number of available addresses is limited. Also the usage of processor-internal registers is more expensive than using the memory of the processor. Thus we implemented a variant of this scenario (variant 2) and defined the decoded addresses in the memory. In this variant, the number of addresses is limited by the available memory size. However the used processor needs twice the time to read addresses from the memory compared to read accesses on internal registers.

### E. Comparison

In the following section all SPI realizations are compared to each other. For this purpose, each scenario has been simulated with the defined test cases. The ratio of the ASIC's system clock to the SPI master module clock is defined as 2 according to the present sensor ASIC. Table I lists all realizations in the first column. The first row contains the existing hardware reference which is transformed into a SystemC module ("HW scenario 0"). The next rows show mixed HW/SW implementations without ("SW scenario 1") and with ("SW scenario 2/3") accelerating modifications. In "SW scenario 3 (Var. 1)" the addresses are defined in the processor internal registers, whereas the addresses in "SW scenario 3 (Var.2)" are stored in the memory. The third column shows the occurrence of frame errors when using the original clock ratio. In general a frame error occurs, if the timing requirement hasn't been satisfied. The following column lists the number of bits which are used to determine the available cycles for processing incoming requests (see III-A/ Fig. 5). The last column contains the minimal clock ratio in order that no frame errors occur.

The examination of each mixed HW/SW implementation and the mutual comparison produce the following results:

- In "SW scenario 1" a frame error occurs. Thus the assumption, that a migrated software implementation without modifications can't meet the timing requirements, is proven. In contrast to

present examinations, we quantified the required clock ratio avoiding frame errors to the value 4.0.

- A frame error also occurred in "SW scenario 2". The modifications to prefetch register values and to buffer the values in the processor memory improved the minimal clock ratio from 4 to 3, in comparison with "SW scenario 1". In our application, the ASIC internal communication has a small impact on the processing time as compared to the address decoding. In general the benefit of "SW scenario 2" increases when the impact of the on-chip communication becomes higher.
- "SW scenario 3" is adapted to the specific characteristic of the use case and benefits from a defined request order. The modifications increase the number of bits (and thus the number of available cycles to process the request) from 4 to 8 in both variants and a clock ratio of 2.0 is reached. In each implementation no frame errors have occurred. In variant 1 the available addresses are limited to the number of internal registers. If more addresses are needed, variant 2 can be applied with a slightly longer processing time due to the longer time to read addresses from the processor memory.

Based on these results the usage of a mixed HW/SW SPI in automotive multi-channel sensor ASICs is shown to be feasible and convenient, against the current practice. Thus, a project can benefit from a software implementation, in terms of code reusability and higher flexibility. If the clock ratio is increased at least to 4.0, "SW scenario 1" can be used. The resulting ASIC would benefit from a low memory consumption and no further restrictions due to the functionality. If an unlimited random access and a low clock ratio are required, "SW scenario 2" offers a solution in software. A lower clock ratio can be expected especially in cases where the internal ASIC communication impacts substantially the processing time of a request. If the SPI master module requests sensor data in a cyclic defined order and a limited random access is sufficient, the software alternative "SW scenario 3" can be implemented. One of the most important advantages here is the minimal clock ratio which catches up with the HW realization in terms of the achievable clock rate. Depending on the number of requested addresses variant 1 or variant 2 has to be chosen.

## V. RELATED WORK

In [12] the hardware to software migration of real-time functions is explored and the compiler technique "thread integration" is introduced. In previous applications the time-critical functions could be realized only in dedicated hardware. Thread integration aims to enable multithreading without the overhead of context switching. Thereby time-critical functions could also be implemented in software.

Next to the HW to SW migration the opposite direction from SW to HW for a faster internal processing and less power consumption is applied. In [13] the authors explore the migration of software parts to hardware regarding to the system throughput and estimate the gained performance by means of synchronous data flow graphs. In addition some migration impacts are explained, such as the impact on the hardware execution time or the overhead of inter-task communication.

An optimization for the hardware realization of an SPI to reduce the relatively high degree of intervention from the master controller is described in [14]. The relieved reduction of intervention is reached by more autonomous operations of the serial peripheral interface. This patent modifies the SPI to store serial transfers in memory and execute them autonomously.

Similar to [12] we explore a time-critical application and migrate the hardware realization to software. In contrast using an adapted compiler technology achieving multithreading, we implemented software-based mechanisms to optimize the internal processing of an SPI. [13] also aims to receive a faster internal processing, but in contrast to our approach software parts are migrated to hardware without optimization. Like [14] we modify the internal processing to improve an SPI. However no migration is applied in [14] and the improvements take place in hardware.

## VI. CONCLUSION

In this paper we showed (a) the hardware to software migration of an existing SPI from an automotive sensor ASIC, (b) optimized software implementations of the application-specific SPI with accelerated internal processing and (c) the comparison between the hardware realization and the alternatives in software. We showed that the software implementation without optimization requires a higher clock ratio and we quantified this ratio. The results demonstrate that the proposed optimized software implementations are convincing alternatives to hardware solutions. Thus, subsequent multi-channel sensor ASICs can benefit from a higher flexibility and code reusability of a software-implemented SPI.

## ACKNOWLEDGMENT

The authors would like to thank Ruben Bartholomä, Jan Heinrich, Walter Mack and Jan-Hendrik Oetjens for their helpful support.

## REFERENCES

- [1] A. Mauderer, J. Oetjens, and W. Rosenstiel, "System-level design for automotive mixed-signal-asics: An industrial point of view," in *Methoden und Beschreibungssprachen zur Modellierung und Verifikation von Schaltungen und Systemen* (MBMV), 2011.
- [2] J. Marek, "MEMS for automotive and consumer electronics," in *Solid-State Circuits Conference Digest of Technical Papers* (ISSCC), 2010 IEEE International. IEEE, 2010, pp. 9–17.
- [3] T. Tille, *Sensoren im Automobil II*, Expert Verlag, Ed. Prof. Dr.-Ing. Ulrich Brill, 2007, vol. 2.
- [4] *LIN specification package*, LIN Consortium Std., Rev. Revision 2.0, 2003.
- [5] *Single Edge Nibble Transmission for Automotive Applications*, SAE SENT Task Force Std., Rev. SAE J2716, 2008.
- [6] *Peripheral Sensor Interface for Automotive Applications—Technical Specification*, PSIS Organization Std., Rev. V2.1, 2012.
- [7] *Road vehicles – Controller area network (CAN) – Part 1: Data link layer and physical signaling*, ISO Standard - International Standards Organization Std., Rev. ISO 11898:1993.
- [8] *SPI Block Guide*, Freescale Semiconductor Std., Rev. V03.06, 2003.
- [9] Accellera Systems Initiative, "Homepage accellera systems initiative," 2015. [Online]. Available: <http://www.accellera.org/home/>
- [10] C. Kerstan, N. Bannow, and W. Rosenstiel, "Efficient architecture evaluation using functional mapping," in *Languages for Embedded Systems and Their Applications*, P. D. M. Raetzki, Ed. Springer Netherlands, 2009, pp. 167–182.
- [11] R. Görge, J.-H. Oetjens, and W. Nebel, "Automatic integration of hardware descriptions into system-level models," in *Design and Diagnostics of Electronic Circuits & Systems (DDECS)*, 2012 IEEE 15<sup>th</sup> International Symposium on. IEEE, 2012, pp. 105–110.
- [12] A. G. Dean and J. P. Shen, "Hardware to software migration with realtime thread integration," in *Euromicro Conference*, 1998. Proceedings 24th, vol. 1. IEEE, 1998, pp. 243–252.
- [13] I. Bennour, D. Sebai, and A. Jemai, "Modeling SW to HW task migration for MPSOC performance analysis," in *Design and Technology of Integrated Systems in Nanoscale Era (DTIS)*, 2010 5th International Conference on. IEEE, 2010, pp. 1–6.
- [14] S. C. Hill, J. Jelemensky, M. R. Heene, S. E. Groves, and D. N. DeBrito, "Queued serial peripheral interface for use in a data processing system," United States Patent, Sep. 1990, US Patent 4,958,277.
- [15] M. Freier, A. Wenzler, T. Mayer, J. Gerlach, and W. Rosenstiel, "System-level exploration of design decisions based on indicators," in *50. Workshop on Microelectronics*, pp. 43–47, 2013.



Marvin Freier studied Systems Engineering at Albstadt-Sigmaringen University and received the degree Master of Engineering in 2011. He is currently working toward the Ph.D. degree at the department of computer engineering, University of Tübingen, in collaboration with the Department of Automotive Electronics, Robert Bosch GmbH. His dissertation is supervised by Prof. Dr. Rosenstiel and Prof. Dr. Gerlach.



Kai Schulz studied Computer Engineering at Albstadt-Sigmaringen University and received the degree Bachelor of Engineering in 2014. At present he studies Systems Engineering at Albstadt-Sigmaringen University to earn the degree Master of Engineering.



Thomas Mayer received the degree Master of Science in Computer Engineering from the Case Western Reserve University (CWRU) in 1996. He graduated his Diploma in Informatics from the University of Dortmund in the year 1997. Thomas Mayer is senior expert in the field of inertial sensor ASICs for Automotive Electronics, Robert Bosch GmbH.



Axel Wenzler graduated his Diploma in Electrical Engineering from the University of Stuttgart in 1994. He received the degree of Dr.-Ing. from the University of Stuttgart in the year 1999. Currently he is head of an ASIC design department at Robert Bosch GmbH. He's lecturer at the Baden-Württemberg Cooperative State University (DHBW) and member of the Union Radio Scientifique Internationale (U.R.S.I.).



Joachim Gerlach received his Diploma in Computer Science from the Technical University of Karlsruhe in 1995 and got his Ph.D. from the University of Tübingen in 2000. He worked at the Automotive Electronics division of Robert Bosch for nine years. Since 2009 he is professor for computer and systems engineering at the Albstadt-Sigmaringen University.



Wolfgang Rosenstiel graduated his Diploma in Computer Science from the University of Karlsruhe in the year 1980 and received 1984 the degree of Dr.rer.nat. from the University of Karlsruhe. Since 1990 he is professor for Computer Science at the University of Tübingen and Director of the Department "System Design in Microelectronics" at FZI. Since 2010 he is Dean of the Faculty of Science at the University of Tübingen.

# A Flexible Approach on FPGA-based Digital Sensor Interfacing for Cyber Physical Systems

Isaac Verdu, Kurt Franz Ackermann

**Abstract**—Sensing nodes in Cyber-physical Systems (CPS) share many of the well-known constraints of traditional embedded systems: power efficiency, limited computational resources and close interaction with its environment. Moreover, nodes in CPS are expected to have a certain level of autonomy and adaptability not usually required in deep embedded systems. In this paper, a design concept for the development of adaptive sensor interfaces in CPS is introduced. Our approach takes advantage of the Dynamic Partial Reconfiguration (DPR) capabilities of modern Field Programmable Gate Arrays (FPGA). Thus, on-demand adaptations of interfaces to external sensors are supported, while minimizing the impact of such modifications in both the sensor node itself and the network as a whole. Functional logic blocks are defined and isolated, in an effort to maximize modularity and reusability. The presented framework increases the flexibility of sensor nodes and provides a required level of abstraction by hiding protocol specific details.

**Index Terms**—Field Programmable Gate Arrays, Cyber-physical Systems, Dynamic Partial Reconfiguration, Sensor Networks.

## I. INTRODUCTION

In the traditional sense, an embedded system is a computing system that, unlike general purpose computing systems is tightly integrated with the elements in its environment, which are often fixed, or offer little room for expandability. It is supposed to operate autonomously and reliably for extended periods with severe resource and time restrictions. However, as communication technology continues to move forward, new opportunities arise, and the barriers of what embedded systems could and should do, are pushed even further.

Cyber-physical Systems (CPS) constitute a new paradigm of embedded systems, which are expected to work in collaborative environments, interacting with other embedded systems or even with central control

units (often referred to as “the cloud”). This paradigm shift, together with the miniaturization and improved reliability and efficiency of sensor technology, has enabled the implementation of sensor networks. Containing sensor nodes have the ability of collecting and processing data autonomously, but are also capable of communicating with other sensors within the same network and/or a central control unit. This technology provides a benefit to domains of home and industrial automation, surveillance systems, environmental data collection, etc.

One of the most important issues to be addressed in such architectures is the need to improve the reliability and adaptability of the array. Nodes are expected to transmit data periodically and reliably in real-time and, at the same time, respond to requests from other processing units. They should support various sensor interfaces and provide adequate upgrade mechanisms, minimizing down time and human intervention.

In this paper, state-of-the-art FPGA technology is used in order to describe a design methodology for generic sensor interfaces. Encapsulating the interface logic is mandatory to simplify the connection of various sensor types to a CPS-node. Moreover, applying dynamic partial reconfiguration to such a partitioned design leverages flexibility by providing on-demand adaptability. Thus, CPS-nodes remain continuously functional while sensors may be connected, removed or exchanged by a user. Possible use cases for such a scenario are medical systems giving a surgeon the opportunity to exchange sensors on demand in order to measure different parameters or to acquire data from various image sensors. At this it might moreover be important that a wireless sensor node remains continuously operational and for example holds the connection to its host. Further use cases for run-time adaptations of nodes may be robotic systems in research labs. The complexity and resource requirements of the underlying base design may be extensive, making it desirable to minimize the required interface logic.

However, since data formats and requirements of different interface modules may vary, additional logic is required in order to abstract the communication between data processing units and the interface module.

The rest of this article is divided as follows: Part II discusses the concept of system nodes, the role they play on a network, and how FPGAs contribute to meeting critical real-time requirements, including aspects of reconfigurability. The proposed architecture is described in Part III, where the concept of a Reconfigurable Communication Channel (RCC) is presented. Part IV consists of a case study, which is used to validate the feasibility of the ideas previously introduced. Finally, in Part V, the conclusions are outlined, and ideas for future work are discussed.

## II. ON-DEMAND RECONFIGURATION OF SYSTEM NODES

Nodes in sensor networks usually operate autonomously, while at the same time interacting with other nodes and/or a central control unit. The crucial task of a sensor node is the acquisition of data from external sensors, whereat both, efficiency and adaptability are required. At this, we concentrate on the majority of digital interfaces. Processing elements within nodes establish the configuration of sensors and control data transfers. Such an element must be either hardware or software-based.

### A. Software-based vs. hardware-based solutions

Software-based processing nodes typically make use of microcontrollers (uC) or Digital Signal Processors (DSPs), integrating a set of standardized memory-mapped communication peripherals implemented in silicon. If additional or modified communication protocols are required, a pure software implementation may be feasible, using available General Purpose IOs to interface corresponding sensors. However, such approaches are often inefficient.

Hardware-based solutions, on ASIC or FPGA platforms, are more complex, but enable the implementation of full custom interfaces, best suited for a given application. While ASICs offer the highest power efficiency and performance, they inherently lack flexibility and upgradeability.

FPGAs offer a compromise between the flexibility of an uC/DSP and the advantages of ASIC-like hardware implementations. Three characteristics make these fine granular devices particularly suitable for sensor nodes: their hardware architecture, the ability to integrate soft or hard processor cores on the same die, and their reconfiguration features.

Due to massive, ASIC-like parallelization in the data path, very high data rates are supported at clock frequencies of typically less than 200 MHz. By dealing with hardware signals in a parallel fashion [1], FPGAs eliminate part of the processing overhead of uC (e.g. interrupt handling), leveraging performance and potentially reducing the overall power consumption [2].

Table 1: Summary of ASIC, uC and FPGA-based processing features.

Architecture	Features
Firmware (uC/DSP)	<ul style="list-style-type: none"> <li>- Flexible and adaptable through software updates</li> <li>- Lower power consumption than FPGAs</li> <li>- Interfacing capabilities limited by existing peripherals</li> </ul>
Hardware (ASIC)	<ul style="list-style-type: none"> <li>- Highest speed</li> <li>- Lowest power consumption</li> <li>- Fixed hardware configuration</li> </ul>
Hardware (FPGA)	<ul style="list-style-type: none"> <li>- Field-programmable hardware</li> <li>- Partially reconfigurable during run-time</li> <li>- Low latency</li> <li>- Parallel processing</li> <li>- Ability to run software</li> <li>- Highest power consumption (can be improved through synthesis techniques, rapid execution and parallel processing)</li> </ul>

Moreover, modern FPGAs are heterogeneous devices coupling the FPGA fabric with multi-core CPUs as efficiently as never before (Xilinx's Zynq, Altera's Cyclone V, Microsemi's SmartFusion). Thus, HW/SW co-designs may increase a system's flexibility, making it natural to place high-level control logic into the SW part.

The distinctive characteristic of FPGAs, is their reconfigurability. Reconfiguration in FPGAs comes in two flavors: static reconfiguration, in which the whole functionality is disabled while a new bitstream is programmed into the device, and Dynamic Partial Reconfiguration (DPR), in which a portion of the device is programmed while the rest of the logic remains functional [3]. This feature, supported by leading FPGA vendors, stays at the core of the design approach presented in this article. Table 1 summarizes the most relevant characteristics of all three processing architectures, as they apply to digital sensor interfacing.

### B. Dynamic Partial Reconfiguration (DPR)

Depending on the application, Dynamic Partial Reconfiguration (DPR) offers certain advantages compared to the classic static configuration method. Figure 1 illustrates the basic premise of DPR. The ability to configure only a subset of a design during run-time inherently increases a system's flexibility and adaptability [4]. Partial bitfiles are smaller, require less storage memory and reduce the configuration time. Today's devices enable self-reconfigurations through Internal Configuration Access Ports (ICAP) with high bandwidths of 400 MB/s. The DPR principle enables time-division-multiplexing of device resources, breaking classical resource boundaries of hardware designs. Hence, for designs not requiring the complete functionality at a time, smaller FPGAs can be used, reducing the costs significantly.)

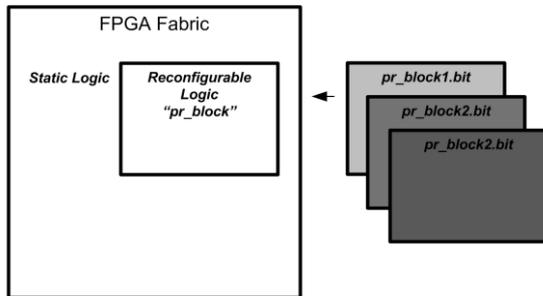


Figure 1. Premise of Partial Reconfiguration.

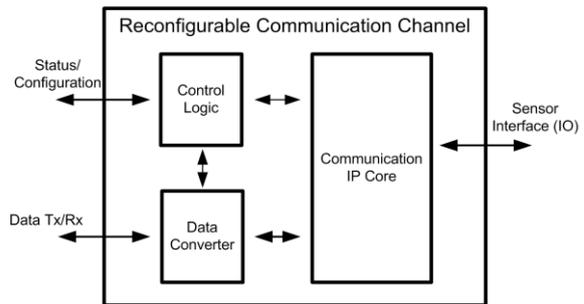


Figure 3: Reconfigurable Communication Channel Architecture.

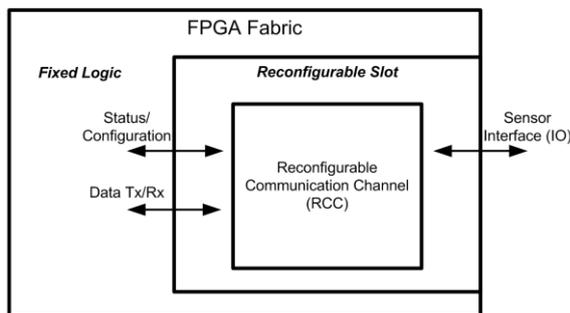


Figure 2: Top-level interfaces of the Reconfigurable Communication Channel.

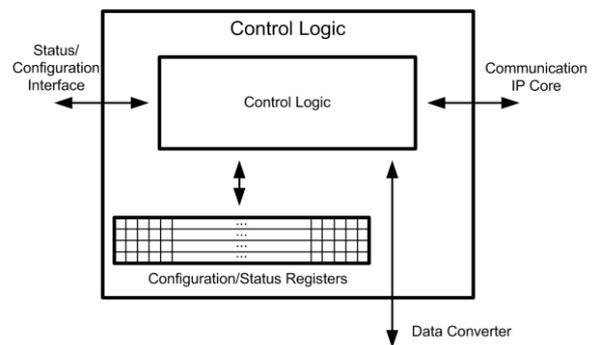


Figure 4: Control Logic Block Architecture.

Nevertheless, the design flow differs from that of traditional FPGA projects [5]. At this, interfaces between static and dynamic regions of a design must be kept consistent [6]. Once the global bitstream has been generated, various partial bitstreams for the existing PR regions may be created.

### III. MODULAR ARCHITECTURE DESCRIPTION

The basic logic unit of the proposed architecture is the Reconfigurable Communication Channel (RCC). Figure 2 shows the top-level interfaces of this module. From an operational point of view, it is the equivalent of the communication peripherals (SPI, I<sup>2</sup>C, etc.) that can be implemented in the programmable logic of an FPGA, or directly in silicon in a microcontroller. Unlike those, however, it features a clear distinction (logically and physically) between the data and control interfaces. The separation of data and control signals allows the standardization of the data input/output logic, thus facilitating the interoperability of different data processing modules, as well as the parallelization of the development process.

An RCC consists of three basic, logic elements, as shown in Figure 3. The Communication IP implements the data-exchange with an externally connected sensor. It is the physical implementation of a communication protocol, which might be taken from an already existing IP-library, or it may have to be specifi-

cally designed for a dedicated solution.

The logical structure of the Control Logic module is depicted in Figure 4. This module is responsible for controlling the operation of the Communication IP. This includes a) the configuration of the general operation parameters of the communication module, and b) the issuing of write/read commands, coupled with any additional information associated with them.

General configuration parameters (i.e. those that usually remain unchanged over several transactions) are stored in internal registers within the Control Unit, and then wired out to the Communication IP Core, which can use such values to adjust its operation accordingly. A good example of such parameter would be the clock speed of a bus: even though it can be changed at any given time, it is more likely that several data transmissions utilize the same frequency. Having a dedicated architecture for the adjustment of such parameters will help reduce the processing overhead, which would be higher if they were to be set every time a data transaction occurs.

Read/Write commands, on the other hand, are issued on a per transmission basis, along with their necessary execution parameters. These must be decoded every time that a data transaction takes place, and may include slave addresses, byte-masks (required for the support of different word lengths), etc. The command itself (i.e. writing to or reading from a sen-

Table 2: AXI4-Stream Command and Data Port Listing.

Port	Description
AXI4-Stream Command Interface	
axi4s_cmd_tdata	Data bus containing the command and configuration options associated with it
axi4s_cmd_tvalid	Indicates that valid information has been placed on the 'tdata' bus
axi4s_cmd_tready	Indicates that the module is ready to receive data
AXI4-Stream Data Interface	
axi4s_data_tdata	Data bus containing the data to be transmitted
axi4s_data_tlast	Not used
axi4s_data_tvalid	Indicates that valid information has been placed on the 'tdata' bus
axi4s_data_tkeep	Not used
axi4s_data_ready	Indicates that the module is ready to receive data

sor) as well as the byte-mask are forwarded to the Data Converter module, which shall use them to control the actual transmission of data, while all other parameters are set directly from the Control Logic unit.

The Control Logic module is, by its very nature, dependent on the specific implementation of a communication protocol (i.e., the Communication IP block). This has two immediate consequences: on one hand, it needs to be adapted to the module that is actually implementing the protocol, and any changes in the latter will cause a revision of the former. On the other hand, the control interface (that is, the one going in/out of the RCC) is solely dependent on the protocol specification, does not have to deal with the details of a particular implementation and remains the same as long as the communication protocol is not changed. It follows logically that while a portion of the Control Logic needs to be developed and paired with a pre-existing communication IP module, the section implementing the Control Interface is highly reusable and effectively abstracts the functional behavior of the RCC from the details of its inner workings.

The structure of the Data Conversion module is depicted in Figure 5. It serves the purpose to handle the actual data that is being transmitted. Thus, it acts as a bridge between the RCC's external data interface and the Communication IP block.

At one end of the Data Conversion module, there is an AXI4-Stream data interface, both in its master and slave variations. This has multiple advantages: in addition to simplifying the design process, we do so by using a proven industry standard that has a great deal of support and popularity, and paves the way for a relatively straightforward integration of the RCC

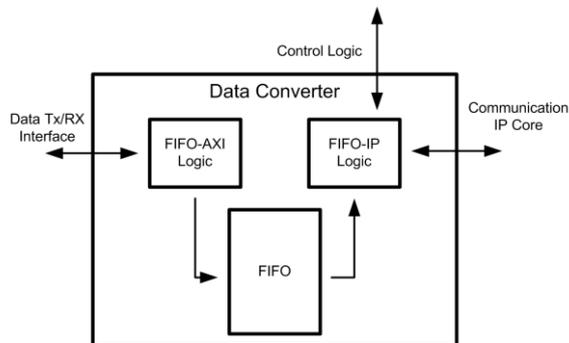


Figure 5: Data Converter Architecture.

with existing IP and both soft and hard processor cores. The data transfer protocol of the AXI4-Stream interface is well documented, and some manufacturers have gone as far as providing reference designs including template FSMs for both data sources and sinks.

The Data Conversion module is also connected to the Communication IP, to which it provides the data to be sent, and from which it reads the received data. As one could expect, this logic is both protocol and implementation dependent, and it is responsible for the actual adaptation of the data from standard registers of a predefined width (typically 32 bits) to the data format required by the specific communication protocol.

Figure 5 also shows a FIFO being implemented inside of the Data Conversion module. This has the goal of effectively decoupling the AXI4-Stream side from the protocol-specific side. This allows easier reusability of the AXI4-Stream logic, and provides a common starting point from which the protocol-specific portion can be developed. It should be noted, however, that there might be some restrictions associated with the use of FIFOs within reconfigurable logic blocks [5] which must be taken into consideration.

With the exception of the Communication IP-Core, all interfaces of the RCC are based on the AXI4-Stream protocol. The port list for both variants (command und data) is described in Table 2. In order to allow the transmission of data in both directions, master und slave instances are used by the Control Logic and Data Converter modules.

#### IV. CASE STUDY

The design approach proposed in this article was applied in the development of a generic communication channel. Due to its simplicity, an IP-core implementation of an I<sup>2</sup>C-compliant protocol [7] was used. The functional description of the I<sup>2</sup>C protocol is detailed in [8]. Figure 6 shows a block diagram of the I<sup>2</sup>C -based

Table 3: Port Listing of the I2C IP-Core.

Port	Direction	Description
clk	Input	System clock
reset_n	Input	Asynchronous active low reset
bus_clk	Input	Speed of the I <sup>2</sup> C bus clock
ena	Input	0: no transaction initiated 1: transaction initiated
addr	Input	Address of target slave
rw	Input	0: write command 1: read command
data_wr	Input	Data to transmit if rw = 0 (write)
data_rd	Output	Data received if rw = 1 (read)
busy	Output	0: I2C master is idle and last read data is available on data_rd 1: command has been latched in and transaction is in progress
ack_error	Output	0: no acknowledge errors 1: at least one acknowledge error occurred during the transaction, ack_error clears itself at the beginning of each transaction
sda	Inout	Serial data line of I <sup>2</sup> C bus
scl	Inout	Serial clock line of I <sup>2</sup> C bus

RCC. Furthermore, the module's port listing and descriptions are outlined in Table 3.

The Control Logic module drives the bus\_clk and the address ports of the I<sup>2</sup>C block. The first parameter is configured separately from the execution of any specific read/write command, while the second is transaction-dependent. In addition to specifying the slave address for a particular data transmission, a byte-mask is required, since the bit width of the I<sup>2</sup>C protocol (8 bits) differs from that of the buses used inside the FPGA (32 bits). Both the byte-mask (bytes-to-transfer, btt) and the command signals (read/write) are forwarded to the Data Converter module, in addition to a 'ready' signal, which indicates that all necessary configurations of the I<sup>2</sup>C-Master have been done, and the data transmission may begin.

The Data Converter module drives all the ports of the I<sup>2</sup>C-Master associated with the execution of a data transaction (except for the 'address' bus). While this logic is specific to the IP being used, the AXI4-Stream-based interface with the data source/sink in the FPGA remains constant, and the use of a FIFO to separate both logical 'domains' increases the reusability potential of the module. Finally, the three modules are instantiated inside the RCC-Top. At this level, the RCC is connected to the data sink/source and to the sensor interface signals.

As previously mentioned, the proposed design approach utilizes a clear separation of the data and con-

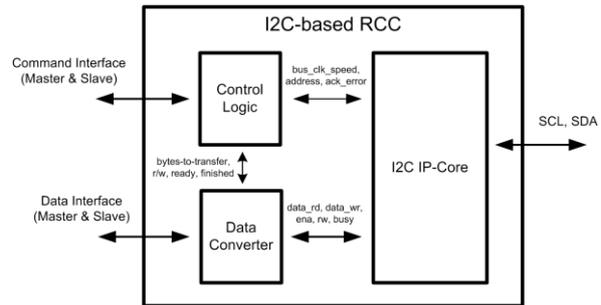


Figure 6: I2C-based RCC.

trol interfaces, in an attempt to maximize modularity, reusability and interoperability. The RCC's drivers fit in this paradigm as well, with an interface handling the configuration and issuing of commands, and the other dealing with the data transmission.

The Configuration/Command interface consists of both Master and Slave AXI4-Stream Command interfaces. The Master bus issues commands to access the contents of the Control Logic module's configuration registers or to initiate a data transaction. The details of each command (bus speed, slave address, byte-mask) are encoded in the 'tdata' bus. The Slave interface is used by the Control Logic module in order to return any configuration data requested by the driver. The Data interface, like the Data Converter inside the RCC, has both a Master and a Slave AXI4-Stream Data interface, allowing bidirectional data transfers.

Figure 7 shows the handshaking sequence of an example 'write' procedure. Before initiating a data transaction, the driver must first setup the configuration parameters of the Communication Core. To do this, the internal registers of the Control Logic module must be properly written (in this case, only the bus speed of the I<sup>2</sup>C interface would need to be specified). Legal values for each parameter, as well as the address of the register that holds it, are agreed in advance, thus the RCC's driver must provide this information through the 'tdata' bus of the AXI4-Stream Command interface, using the 'tvalid' signal to indicate that relevant data is available. The configuration info is stored by the Control Logic module in its internal registers, from where it is made available to the Communication Core. Once the operation is completed an 'acknowledgement' message is sent back to the driver.

Send/Receive operations are initiated in a similar fashion, with the 'tdata' bus that encodes the required command parameters being read in by the RCC when 'tvalid' is enabled. The commands themselves (register read/write, data send/receive) are also transmitted on the 'tdata' bus.

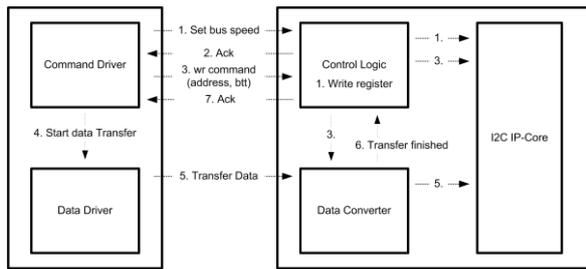


Figure 7: I2C-based RCC Write Sequence.

Once a command is correctly decoded, the Control Logic signals to the Data Converter that a new data transaction can be initiated, and provides the information required by it (byte-mask, read/write) and by the Communication Core (slave address). The Data Converter then asserts its ‘tready’ signal, which informs the driver that the transaction can begin. From this point on, the data is transmitted over the AXI4-Stream data interface, without worrying about how the actual protocol-compliant bus transaction is being carried out. The Data Converter rearranges the 32-Bit data received over the AXI4-Stream interface to a suitable word size, using the byte mask received from the Control Logic module.

Once a data transaction has been completed, the Data Converter asserts a signal for the Control Logic module, which returns an ‘acknowledgement’ message, informing the driver that the requested operation has been processed.

One main advantage of the proposed design strategy is the abstraction of the protocol and implementation details, which greatly improves the adaptability and reusability of the RCC’s driver. If the aforementioned transactions were to be carried out using a different protocol (SPI, for instance), it would only be necessary to change the encoding of the ‘tdata’ bus of the command interface, both when changing configuration parameters and initiating data transactions. The rest of the driver’s logic would remain largely the same.

## V. CONCLUSIONS AND FUTURE WORK

This article describes a design approach for the implementation of a generic reconfigurable communication channel, which can take advantage of the dynamic partial reconfiguration abilities of modern FPGAs in order to deploy protocol-independent data transmission modules that can be dynamically reconfigured. The use of a modular architecture with well-defined interfaces based on widely used industry standards increases the reusability and interoperability potential of each module, and facilitates the definition of boundaries for the reconfigurable regions.

The modular approach introduces a processing overhead, due to the additional computations that must

be carried out inside the RCC module. Further development of this methodology should include measuring the performance overhead of the RCC, as well as proposing alternatives for its reduction.

While the transmission of actual data is independent of the underlying communication protocol being used, it is still required that the configuration details specific to said protocol be known to the RCC’s driver (slave addresses in an I<sup>2</sup>C bus, clock frequency of an SPI transaction, length of the words to be transmitted/received, etc.). This limits the abstraction potential of the driver’s Configuration/Command controller, which still must take into account the operational details of the communication protocol at hand (although not of any particular implementation of a protocol). Design methodologies that further abstract the driver’s architecture from the details of the communication protocol being used are highly desired.

## REFERENCES

- [1] G. Garcia, C. Jara, J. Pomares, A. Alabdo, L. Poggi and F. Torres, "A Survey on FPGA-Based Sensor Systems: Towards Intelligent and Reconfigurable Low-Power Sensors for Computer Vision, Control and Signal Processing," *Sensors*, pp. 6247-6278, 31 March 2014.
- [2] J. Valverde, A. Otero, M. Lopez, J. Portilla, E. de la Torre and T. Riesgo, "Using SRAM Based FPGAs for Power-Aware High Performance Wireless Sensor Networks," *Sensors*, pp. 2667-2692, 2012.
- [3] Y. Krasteva, J. Portilla, J. Carnicer, E. de la Torre and T. Riesgo, "Remote HW-SW Reconfigurable Wireless Sensor Nodes," in *Industrial Electronics, 2008. IECON 2008. 34th Annual Conference of IEEE*, Orlando, 2008.
- [4] F. Liu, J. Zhiping and Y. Li, "A Novel Partial Dynamic Reconfiguration Image Sensor Node for Wireless Multimedia Sensor Networks," in *IEEE 14th International Conference on High Performance Computing and Communications*, Liverpool, 2012.
- [5] Xilinx, "Xilinx's Partial Reconfiguration User Guide," 18 January 2012. [Online]. Available: [http://www.xilinx.com/support/documentation/sw\\_manuals/xilinx13\\_4/ug702.pdf](http://www.xilinx.com/support/documentation/sw_manuals/xilinx13_4/ug702.pdf). [Accessed 20 November 2014].
- [6] J. Rodriguez and K. Ackermann, "Leveraging Partial Dynamic Reconfiguration on Zynq SoC FPGAs," in *9th International Symposium on Reconfigurable and Communication-Centric Systems-on-Chip (ReCoSoC)*, Montpellier, 2014.
- [7] S. Larson, "eewiki - I2C Master (VHDL)," 22 October 2014. [Online]. Available: <https://eewiki.net/pages/viewpage.action?pageId=10125324>. [Accessed 5 January 2015].
- [8] NXP Semiconductors, "I2C-bus specification and user manual," 4 April 2014. [Online]. Available: [http://www.nxp.com/documents/user\\_manual/UM10204.pdf](http://www.nxp.com/documents/user_manual/UM10204.pdf). [Accessed 3 January 2015].



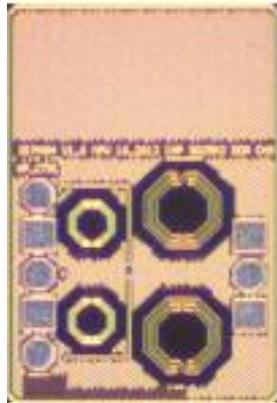
Isaac Verdu received the B.Sc. degree in electrical engineering in 2009 from the Universidad Nacional Experimental Politécnica „Antonio José de Sucre“ in Puerto Ordaz, Venezuela. He is student in the master course of studies of information technology at the University of Applied Sciences Mannheim.



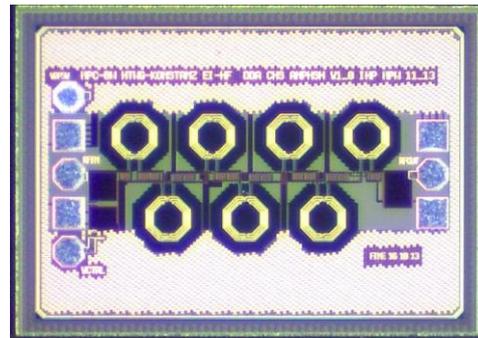
Kurt Franz Ackermann received the academic Dipl.-Ing. degree in electrical engineering and information technology in 2001 and the Dr.-Ing. degree in the year 2011 from the Technical University Darmstadt. He is professor for the design of digital systems and programmable logic at the University of Applied Sciences Mannheim.



## MMIC-Phasenschieber für den Einsatz in phasengesteuerten Arrayantennen im C-Band



(a)

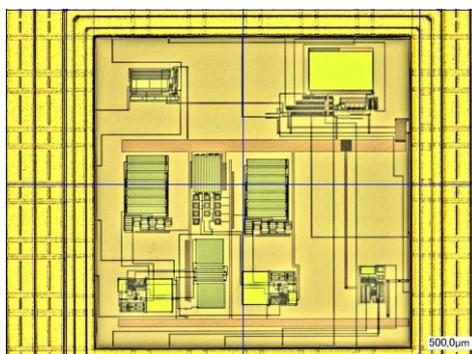


(b)

ENTWURF	Dennis Dario Aschenbrenner, Christoph Schick, Hochschule Konstanz, Fakultät für Elektrotechnik und Informationstechnik, Brauneggerstr. 55, 78462 Konstanz.
ENTWURFSVERFAHREN	Full Custom Design.
TECHNOLOGIE	IHP SG25H3 250 nm BiCMOS 5M.
CHIPFLÄCHE	(a) 0,7 mm x 1,0 mm und (b) 1,1 mm x 0,75 mm.
GEHÄUSE	Bare dies.
FUNKTIONSBLOCKE	(a) Phasenschieber mit schaltbarer Einfügungsphase, (b) Phasenschieber mit variabel einstellbarer Einfügungsphase.
FUNKTION	Bei einer Betriebsfrequenz von 5,4 GHz ermöglicht Chip (a) die spannungsgesteuerte Änderung der Einfügungsphase um 0° und -180° und Chip (b) die variable Änderung der Einfügungsphase um bis zu -180°. Die beiden Chips kommen im Speisetzwerk einer elektronisch schwenkbaren Antenne zum Einsatz.
HERSTELLDATUM	I. Quartal 2014.
KOSTENTRÄGER	MPC-Gruppe Baden-Württemberg.
VORSTELLUNG	D. Aschenbrenner, C. Schick, „Entwurf und Charakterisierung eines MMIC- Phasenschiebers für den Einsatz in phasengesteuerten Arrayantennen im C-Band“, <i>Workshop der Multiprojekt-Chip-Gruppe Baden-Württemberg</i> , Künzelsau, Juli 2014.



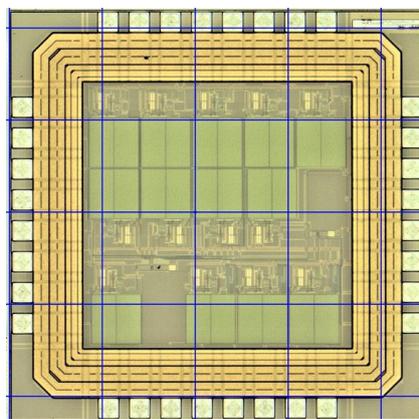
# Operationsverstärker, Bandgap-Referenz, Quarzoszillator und PLL



ENTWURF	Marcio Camoleze de Andrade, Andreas Arnold, Goran Bratek, Christian Eschenbach, Ninja Koetsier, Gaele Ritha Ngassa Tchouta, Bernd Vettermann, Jürgen Giehl, Hochschule Mannheim, Institut für Entwurf integrierter Schaltkreise, Paul-Wittsack-Straße 10, 68163 Mannheim.
ENTWURFSVERFAHREN	Full Custom Design.
TECHNOLOGIE	AMS S35D4M5 0,35 $\mu\text{m}$ CMOS 4M/2P.
CHIPFLÄCHE	2,35 mm <sup>2</sup>
GEHÄUSE	SOIC 20.
FUNKTIONSBLOCKE	Operationsverstärker, Bandgap-Referenz, Strombank, Quarzoszillator, Peaking Current Source, VCO (voltage controlled oscillator), Frequenzteiler mit Teilverhältnis $n = 2 \dots 8$ , Treiberpads.
FUNKTION	Es handelt sich um einen Testchip mit 4 unterschiedlichen Baugruppen. Der Operationsverstärker (OTA) hat eine Leerlaufverstärkung $A_{V0} = 80$ dB und eine Transitfrequenz $f_t = 25$ MHz. Er ist auf maximale kapazitive Last $C_{L\text{max}} = 50$ pF ausgelegt. Die Bandgap-Referenz verfügt über einen Ausgang mit 1.21 V und einen gepufferten Ausgang mit 1.65V. Der Quarzoszillator liefert eine Taktfrequenz $f_{\text{osc}} = 10$ MHz bei einer maximalen Frequenzabweichung von 1300 ppm. Mit der eingebauten PLL lassen sich daraus die Frequenzen $f_{\text{out}} = 80, 40, 26.7, 20, 16, 13.7, 11.3$ und 10 MHz erzeugen.
HERSTELLDATUM	II. Quartal 2014.
KOSTENTRÄGER	MPC-Gruppe Baden-Württemberg.
VERÖFFENTLICHUNG	Goran Bratek, Jürgen Giehl, Bernd Vettermann, Design eines Quarzoszillators in 0,35 $\mu\text{m}$ CMOS-Technologie, Vortrag am 43. MPC-Workshop, 5. Februar 2010, Göppingen, veröffentlicht im Tagungsband ISSN 1862-7102, S. 9-14, Ausgabe 43, Hrsg. Hochschule Ulm.



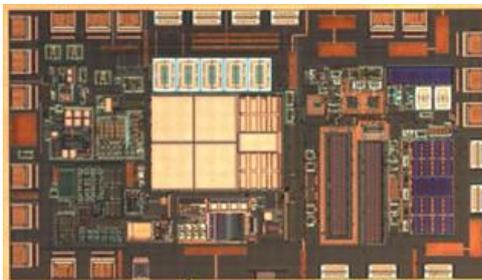
# Operationsverstärker und On-Chip-Tiefpassfilter 3. und 8. Ordnung



ENTWURF	Andreas Arnold, Frauke Bohinsky, Irina Ohm, Paul Kamou Fansi, Ninja Koetsier, Bernd Vettermann, Jürgen Giehl, Hochschule Mannheim, Institut für Entwurf integrierter Schaltkreise, Paul-Wittsack-Straße 10, 68163 Mannheim.
ENTWURFSVERFAHREN	Full Custom Design.
TECHNOLOGIE	AMS S35D4M5 0,35 $\mu\text{m}$ CMOS 4M/2P.
CHIPFLÄCHE	2,04 x 2,04 mm <sup>2</sup> .
GEHÄUSE	QFN 32.
FUNKTIONSBLOCKE	Operationsverstärker, Strombank, Peaking Current Source, Filter 3. und 8. Ordnung.
FUNKTION	Der Chip enthält Operationsverstärker (OTA), mit der Leerlaufverstärkung $A_{v0} = 100$ dB und der Transitfrequenz $f_i = 1$ kHz bei einer maximalen Lastkapazität $C_{L\text{max}} = 50$ pF. Die aktiven On-Chip-Tiefpassfilter 3. und 8. Ordnung sind auf eine Eckfrequenz von 50 Hz ausgelegt. Die Eckfrequenz wird hierbei durch die Miller-kompensierten Operationsverstärker bestimmt. Das Filter 3. Ordnung enthält 3 Operationsverstärker und hat eine Verstärkung von 30 dB. Das Filter 8. Ordnung enthält 8 Operationsverstärker, hat keine Verstärkung (0 dB) und muss noch extern beschaltet werden.
HERSTELLDATUM	II. Quartal 2014.
KOSTENTRÄGER	MPC-Gruppe Baden-Württemberg.



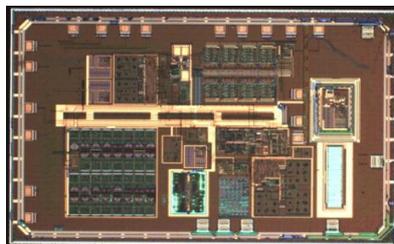
## Galvanisch getrennter High-Side-Gatetreiber



ENTWURF	Achim Seidel, Marco Costa, Joachim Joos, Bernhard Wicht, Hochschule Reutlingen, Alteburgstraße 150, 72762 Reutlingen.
ENTWURFSVERFAHREN	Full Custom Design.
TECHNOLOGIE	180 nm Hoch-Volt BiCMOS.
CHIPFLÄCHE	2 mm x 1,15 mm.
GEHÄUSE	SO 28.
FUNKTIONSBLOCKE	Gatetreiber-Ausgangsstufe mit integrierter Bootstrapschaltung, Gleichrichter, Frequenzdemodulation, Amplitudenmodulation, Bandgap-Referenz, Linearregler, Ladungspumpe, Power-On-Reset.
FUNKTION	Der galvanisch getrennte High-Side-Gatetreiber erhält seine Ansteuersignale mittels Frequenzmodulation über einen diskreten Signalübertrager, worüber auch Signale von der Treiberseite zur Ansteuerelektronik mittels Amplitudenmodulation übertragen werden können. Seine Schaltenergie wird über eine diskrete Bootstrapschaltung bereitgestellt. Ein dauerhaftes Einschalten des Gatetreibers wird durch die Übertragung einer kleinen Energiemenge über den Signalübertrager ermöglicht. Die 15 V-Treiberausgangsstufe enthält eine vollintegrierte Bootstrapschaltung, welche eine Hochvoltkapazität zur Ladungsspeicherung verwendet, wodurch die Schaltung sehr flächeneffizient aufgebaut ist.
HERSTELLDATUM	IV. Quartal 2013.
KOSTENTRÄGER	Industriepartner.
VERÖFFENTLICHUNGEN	<p>A. Seidel, M. Costa, J. Joos, and B. Wicht, "Bootstrap circuit with high-voltage charge storing for area efficient gate drivers in power management systems," in ESSCIRC 2014, <i>Proceedings of the 40th European Solid State Circuits Conference</i>, Sept 2014, pp. 159–162.</p> <p>A. Seidel, M. Costa, J. Joos, and B. Wicht, „Isolated 100% PWM gate driver with auxiliary energy and bidirectional FM/AM signal transmission via single transformer,” in <i>Proc. Applied Power Electronics Conference and Exposition</i>, Charlotte, U.S.A., p. 2581 - 2584, March. 15-19, 2015.</p> <p>A. Seidel, M. Costa, J. Joos, and B. Wicht, "Area Efficient Integrated Gate Drivers Based on High-Voltage Charge Storing", Submitted to IEEE Journal of Solid-State Circuits, July 2015, in press.</p>



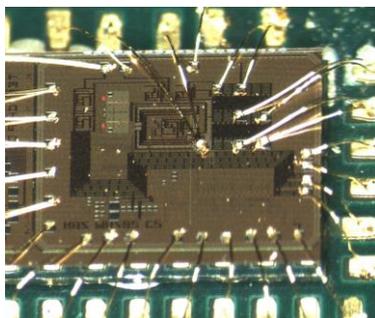
# Hocheffizienter resonanter Spannungswandler bis 30 V mit Maximumsdetektion < 1 ns Auflösung und 50 V High-Speed-Levelshifter



ENTWURF	Tobias Funk, Jürgen Wittmann, Bernhard Wicht, Hochschule Reutlingen, Robert Bosch Zentrum für Leistungselektronik, Alteburgstraße 150, 72762 Reutlingen.
ENTWURFSVERFAHREN	Full Custom Design.
TECHNOLOGIE	180 nm Hoch-Volt BiCMOS.
CHIPFLÄCHE	2.1 mm x 3.6 mm.
GEHÄUSE	CSOIC 28.
FUNKTIONSBLOCKE	Nullspannungsdetektion, Sample and Hold-Schaltung, Komparator, einstellbare Signalverzögerung, Spannungsregler, PWM-Generierung, Levelshifter, Gate- Treiber, Leistungsschalter.
FUNKTION	Um die Größe des passiven Ausgangsfilters zu reduzieren, soll ein Schaltwandler mit Taktfrequenzen > 10 MHz betrieben werden. Die entstehenden Schaltverluste werden durch Zero-Voltage-Switching (ZVS) minimiert. Dazu ist der Leistungs- schalter um einen LC-Resonanzkreis erweitert. Der optimale Einschaltzeitpunkt für den Leistungsschalter ist das Maximum des oszillierenden Spannungsknotens. Hierzu werden zum einen der Hochpunkt des oszillierenden Spannungsknotens detektiert und zum anderen Signallaufzeiten von Levelshifter und Gate-Treiber kompensiert. Um mit dem Schaltwandler einen Eingangsspannungsbereich bis 30 V abdecken zu können, wird der Einschaltzeitpunkt mit einem LSB < 1 ns durch ein prädiktives Regelungsverfahren mit Hilfe einer Mixed-Signal-Auswertung nachgestellt. Dies ermöglicht den effizienten Betrieb von resonanten Spannungsw- andlern mit Schwingfrequenzen bis zu 50 MHz.
HERSTELLDATUM	IV. Quartal 2013.
KOSTENTRÄGER	Industriepartner.
VERÖFFENTLICHUNG	J. Wittmann, Th. Rosahl, B. Wicht: A 50 V High-Speed Level Shifter with High dv/dt Immunity for Multi-MHz DCDC Converters. In ESSCIRC 2014, Proc. 40th European Solid-State Circuits Conference, Venice, Italy, p. 151 - 154, Sept. 22-26, 2014.  T. Funk, J. Wittmann, T. Rosahl, B. Wicht, „A 20 V, 8 MHz Resonant DCDC Converter with Predictive Control for 1 ns Resolution Soft-Switching“, Circuits and Systems (ISCAS), 2015 IEEE International Symposium on, May 2015, in press.



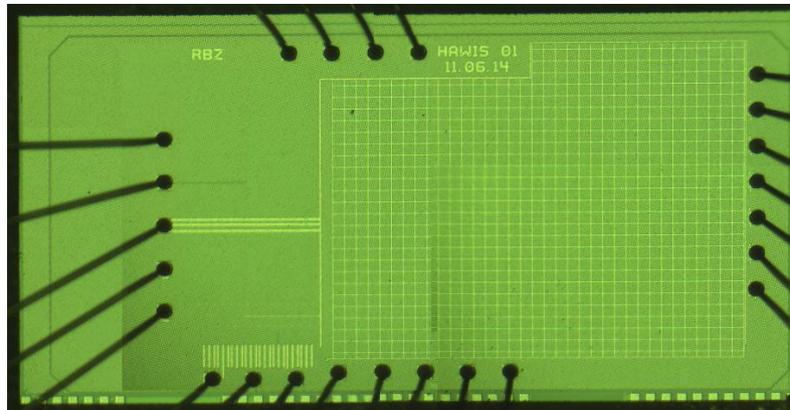
## Synchroner 12 V 10 MHz Abwärtswandler mit Totzeitregelung in 125 ps Auflösung



ENTWURF	Jürgen Wittmann, Bernhard Wicht, Hochschule Reutlingen, Alteburgstraße 150, 72762 Reutlingen.
ENTWURFSVERFAHREN	Full Custom Design.
TECHNOLOGIE	180 nm Hoch-Volt BiCMOS.
CHIPFLÄCHE	2.1 mm x 3.0 mm.
GEHÄUSE	Bare Die.
FUNKTIONSBLOCKE	Halbbrücken-Leistungsstufe, frequenzkompensierte Sample & Hold-Schaltungen, Komparatoren, 8-Bit-Digitalzähler, digitale differentielle 8-Bit-Verzögerungskette.
FUNKTION	Der Abwärtswandler ist für 12 V Eingangsspannung und ein sehr schnelle Taktfrequenz von 10 MHz ausgelegt. Zur Minimierung der dynamischen Verlustleistung dient eine neuartige Regelung der Totzeit an der Halbbrücken-Leistungsstufe. Gegenüber herkömmlichen Lösungen kann auf die Detektion des Einschaltens der Body-Diode verzichtet werden, wodurch zusätzliche Verluste und Störungen entfallen. Ein Prädiktiver Regelansatz erreicht eine genaue Regelung der Totzeit mit einer sehr kleinen zeitlichen Auflösung von 125 ps. Diese Auflösung wird mittels einer digitalen differentiellen Verzögerungskette erreicht, die sehr geringe Herstellungs- und Temperaturschwankungen aufweist. Messungen belegen eine Verringerung der Verlustleistung um 30 %, was einer Erhöhung des Wirkungsgrades um etwa 7 % entspricht.
HERSTELLDATUM	II. Quartal 2014.
KOSTENTRÄGER	Industriepartner.
VERÖFFENTLICHUNGEN	J. Wittmann, A. Barner, Th. Rosahl, and B. Wicht, "A 12V 10MHz Buck Converter with Dead-Time Control Based on a 125ps Differential Delay Line," submitted to ESSCIRC 2015, <i>41st European Solid State Circuits Conference</i> , Sept 2015.



## Hoch-Volt-Interface für 230 V-Netzbetrieb von ICs

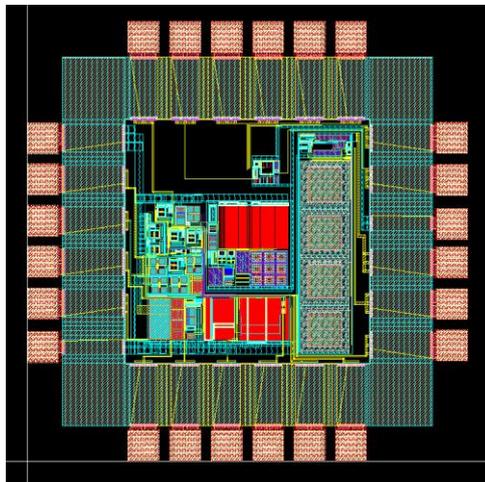


ENTWURF	Daniel Lutz, Bernhard Wicht, Robert Bosch Zentrum für Leistungselektronik Alteburgstraße 150, 72762 Reutlingen.
ENTWURFSVERFAHREN	Full Custom Design.
TECHNOLOGIE	180 nm Hoch-Volt BiCMOS.
CHIPFLÄCHE	3,88 mm x 2,19mm.
GEHÄUSE	CDIP 48.
FUNKTIONSBLOCKE	Hoch-Volt-Interface für 230V Netzspannung, Gleichrichter, Zwischenkreisstabilisierung.
FUNKTION	Bei diesem Chip handelt es sich um einen Testchip, welcher im vom BMBF geförderten Projekt „Hochintegrierte <u>A</u> CDC- <u>W</u> andler als Kleinstnetzteil für direkten 230 V Netzbetrieb von <u>I</u> ntegrierten <u>S</u> chaltungen“ (HAWIS) entwickelt wurde. Dieser Chip enthält Testschaltungen für ein Interface zur 230 V-Netzspannung. Die Netzspannung wird auf eine im Chip handhabbare Spannung gewandelt. Diese Spannung wird gleichgerichtet und mittels einer Zwischenkreisstabilisierung auf eine einstellbare Spannung (z.B. 40 V) stabilisiert. Diese stabilisierte Spannung steht auf dem Chip weiteren Funktionsblöcken zur Verfügung.
HERSTELLDATUM	II. Quartal 2014.
KOSTENTRÄGER	Industriepartner.





## Ultra-Low-Power-Verstärker-ASIC mit Energy Harvesting



ENTWURF	Benjamin Steiner, Gerhard Forster, Hochschule Ulm, Institut für Kommunikationstechnik, Prittwitzstraße 10, 89075 Ulm.
ENTWURFSVERFAHREN	Full Custom Design.
TECHNOLOGIE	UMC L180 Mixed-Mode/RF 180 nm CMOS 6M/1P 1.8V/3.3V.
CHIPFLÄCHE	0,98 mm x 0,98 mm.
GEHÄUSE	QFN 24.
FUNKTIONSBLOCKE	Sperrschwinger, Aktivgleichrichter, Bandgap-Referenz, Schaltregler, Linearregler, Logik, Spannungsverdoppler, Ultra-Low-Power-Verstärker.
FUNKTION	Der Chip enthält den IP-Core einer Versorgungseinheit, die eine Versorgungsspannung von wahlweise 1,8 V oder 3,3 V zum Betrieb eines ASICs zur Verfügung stellt. Die Versorgungseinheit kommt ihrerseits mit einer Speisespannung von weniger als 100 mV aus. Als Nutzlast befindet sich auf dem Chip ein Operationsverstärker mit zwei Differenzeingängen (Differential Difference Amplifier, DDA) zur Verstärkung von hochohmigen Differenzsignalen. Mit einer Stromaufnahme von 1 $\mu$ A erreicht er eine Verstärkung von 70 dB und eine Transitfrequenz von 1 MHz.
HERSTELLDATUM	II. Quartal 2014.
KOSTENTRÄGER	MPC-Gruppe Baden-Württemberg.
VERÖFFENTLICHUNG	J. Rechtsteiner, G. Forster, „Ein Energy Harvesting IP für den Einsatz in einem 0,18 $\mu$ m CMOS ASIC“, <i>Workshop der Multiprojekt-Chip-Gruppe Baden-Württemberg</i> , Offenburg, Tagungsband ISSN 1868-9221, S. 1 – 12, Ausgabe 47, Februar 2012.

## 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. Schick, (07531) 206-657  
cschick@htwg-konstanz.de

### Hochschule Mannheim

Prof. Dr. Giehl, (0621) 292-6860  
j.giehl@hs-mannheim.de

### Hochschule Offenburg

Prof. Dr. Sikora, (0781) 205-416  
axel.sikora@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. Wicht, (7121) 271-7090  
bernhard.wicht@reutlingen-university.de

### Hochschule Ulm

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

[www.mpc.belwue.de](http://www.mpc.belwue.de)