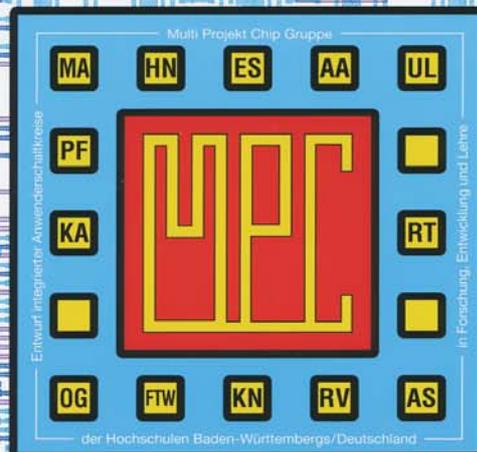


MULTIPROJEKTCHIP GRUPPE

BADEN-WÜRTTEMBERG

MPC-Workshop Februar 2007

Pforzheim



MULTIPROJEKTCHIP GRUPPE

BADEN-WÜRTTEMBERG

MPC-Workshop Februar 2007

Pforzheim

Cooperating Organization
Solid-State Circuits Society Chapter
IEEE Germany Section



ISSN 1862-7102

Herausgeber: Hochschule Ulm

**Alle bisherigen MPC-Workshopbände finden Sie im Internet unter:
<http://www.belwue.de>**

© 2007 Hochschule Ulm

Das Werk und seine Teile sind urheberrechtlich geschützt. Jede Verwertung in anderen als den gesetzlich zugelassenen Fällen bedarf deshalb der vorherigen schriftlichen Einwilligung des Herausgebers Prof. A. Forster, Hochschule Ulm, Prittwitzstraße 10, 89075 Ulm.

Adressen der

MULTIPROJEKT-CHIP-GRUPPE (MPC-Gruppe)

BADEN - WÜRTTEMBERG

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

Hochschule Aalen

Prof. Dr. Bartel, Postfach 1728, 73428 Aalen
Tel.: 07361/576-107, E-Mail: manfred.bartel@htw-aalen.de

Hochschule Albstadt-Sigmaringen

Prof. Dr. Rieger, Johannesstr. 3, 72458 Albstadt-Ebingen
Tel.: 07431/579-124, E-Mail: rieger@hs-albsig.de

Hochschule Esslingen

Prof. Dr. Lindermeir, Flandernstr. 101, 73732 Esslingen
Tel.: 0711/397-4230, E-Mail: walter.lindermeir@hs-esslingen.de

Hochschule Furtwangen

Prof. Dr. Rülling, Postfach 28, 78113 Furtwangen
Tel.: 07723/920-503, E-Mail: rue@hs-furtwangen.de

Hochschule Heilbronn

Prof. Dr. Schröder, Max-Planck-Str. 39, 74081 Heilbronn
Tel.: 07131/504-639, E-Mail: jschroeder@hs-heilbronn.de

Hochschule Karlsruhe

Prof. Dr. Koblitz, Postfach 2440, 76012 Karlsruhe
Tel.: 0721/925-2238, E-Mail: rudolf.koblitz@hs-karlsruhe.de

Hochschule Konstanz

Prof. Dr. Voland, Brauneggerstraße 55, 78462 Konstanz
Tel.: 07531/206-644, E-Mail: voland@htwg-konstanz.de

Hochschule Mannheim

Prof. Dr. Paul, Speyerer Str. 4, 68136 Mannheim
Tel.: 0621/292-6351, E-Mail: g.paul@hs-mannheim.de

Hochschule Offenburg

Prof. Dr. Jansen, Badstr. 24, 77652 Offenburg
Tel.: 0781/205-267, E-Mail: d.jansen@fh-offenburg.de

Hochschule Pforzheim

Prof. Dr. Kesel, Tiefenbronner Str. 65, 75175 Pforzheim
Tel.: 07321/28-6567, E-Mail: frank.kesel@fh-pforzheim.de

Hochschule Ravensburg-Weingarten

Prof. Dr. Ludescher, Postfach 1261, 88241 Weingarten
Tel.: 0751/501-9685, E-Mail: ludescher@hs-weingarten.de

Hochschule Reutlingen

Prof. Dr. Kreutzer, Alteburgstraße 150, 72762 Reutlingen
Tel.: 07121/271-7059/-7060, E-Mail: hans.kreutzer@reutlingen-university.de

Hochschule Ulm

Prof. Forster, Postfach 3860, 89028 Ulm
Tel.: 0731/50-28180, E-Mail: forster@hs-ulm.de

Inhaltsverzeichnis

	Seite
1. Algorithmen für die sichere Netzanbindung von Embedded Systemen A. Sikora, BA Lörach	5
2. GMSK Demodulation - Realisierung auf FPGA Basis M. Hummel, HS Heilbronn	17
3. Bit-Error Tests an einem FPGA Design mit schneller serieller Übertragungsstrecke T. Helgeth, EADS Deutschland, HS Ulm	23
4. VHDL Synthese & FPGA Emulation eines Microcontrollersystems für eine elektronische Pille T. Benneckenstein, N. Fawaz, D. Jansen, HS Offenburg	35
5. Assoziativer Koprozessor für schnelle Textsuche in umfangreichen Datenbanken C. Layer, Uni Ulm	41
6. Design of a Low Power PTAT Bandgap Cell Ji Li, D. Jansen, HS Offenburg	49
7. Lohnt sich der Einsatz verlustleistungsoptimierender Buskodierverfahren? F. Kesel, M. Gaiser, HS Pforzheim	55
8. Synchronization of a Local XTAL-Oscillator to an External High-Precision Frequency Standard W. San-Um, Uni, Pathum-Thani, Thailand W. Ludescher, HS Weingarten	61
9. Simulation Rechenintensiver A/MS-Systeme in der Automobilindustrie R. Dölling, H. Mielenz, Robert Bosch GmbH	69
Gefertigte Bausteine	
10. Testchip für einen 10-Bit-ADC M. Diebold, G. Forster, HS Ulm	90
11. Wearlog D. Bau, D. Jansen, HS Offenburg	91

Algorithmen für die sichere Netzanbindung von Embedded Systemen

Prof. Dr.-Ing. Axel Sikora, Dipl.-Ing. Dipl. Wirt.-Ing.

Berufsakademie Lörrach, Hangstraße 46-50, 79539 Lörrach

Tel. 07621-2071-441, Telefax 07621-2071-495, E-Mail sikora@ba-loerrach.de

Zusammenfassung

Der Grad der Vernetzung von mikroelektronischen Systemen nimmt immer mehr zu. Dies gilt nicht nur für PC- und Mainframe-Rechner, sondern auch für nicht so leistungsfähige Embedded Systeme. Mit dem Vordringen der so genannten Embedded Connectivity gewinnen in diesem Bereich insbesondere auch die Anforderungen an die Sicherheit mit ihren wichtigsten Parametern Vertraulichkeit, Authentifizierung und Integrität an Bedeutung.

Da es in den meisten Fällen nicht möglich ist, die gleichen Technologien und Protokolle in Embedded Systemen einzusetzen wie in PC- oder Mainframe-basierten Rechnern, müssen spezifische Lösungen identifiziert werden, die zum einen im Rahmen der Leistungsfähigkeit des Mikrocomputers liegt, zum Zweiten auch die Sicherheitsanforderungen sinnvoll erfüllt, und zum Dritten auch die nahtlose Vernetzung mit der Office-Welt ermöglicht.

Der Beitrag untersucht die grundlegende Problematik, diskutiert die Eignung bestehender und neu entwickelter Algorithmen und Protokolle für die Entwicklung sicherer Embedded Systeme. Hierbei wird neben dem Vergleich zwischen Software- und Hardware-basierten Lösungen auch auf architektonische Aspekte der Platzierung von kryptographischen Beschleunigern, sowie auf den momentan laufenden europäischen eSTREAM-Wettbewerb eingegangen, der einen für Embedded Systeme effizienten Stromchiffre identifizieren soll.

1. Embedded Networking

Ob Fernzugriff auf die Niveaumessung des Öltanks, Anpassung der aktiven Heizungskennlinie oder Ableitung von Verbrauchswerten: Embedded Webserver in den Endgeräten und das Internet als Netzwerk bilden eine ideale und zukunftsweisende Plattform, um solche und ähnliche Aufgaben effizient zu lösen. Hierbei werden mit Embedded Systemen und Internet-basierter Kommunikation zwei günstige und grundle-

gende Technologien vorteilhaft verbunden [1]. Insbesondere entfallen die komplexe Einrichtung und Verwaltung sowie die u.U. hohen Kosten von Modemstrecken. Bei web-basierten Lösungen entfällt zusätzlich die problematische Pflege von Auswerteprogrammen.

Von besonderer Attraktivität erscheint hierbei, dass das Internet-Protokoll auch unmittelbar auf die kostengünstigen und außerordentlich flexiblen drahtlosen Kurzstreckenfunksysteme (Short Range Wireless Networks – SRWN [2]) erweitert werden können [3] [4].

2. Parameter der Sicherheit

2.1. Definition

Sicherheit wird auch für Embedded Systeme immer wichtiger. Dies gilt insbesondere dann, wenn sie über Netzwerkfunktionalität verfügen. Besondere Risiken ergeben sich dann, wenn Internet-Protokolle verwendet und die Systeme auch an das Internet angebunden werden, da sie dann auch Ziel aller dort verbreiteten systematischen Angriffe werden können [5]. Ein weiteres wichtiges Szenario besteht in der drahtlosen Anbindung von Embedded Systemen.

Sicherheit kann als Zustand des Nichtvorhandenseins von oder Schutz vor Gefahren und Risiken beschrieben werden. Sicherheit ist damit aber eine subjektiv wahrnehmbare Größe, die weder direkt sichtbar noch messbar ist [7]. Dabei schließt der deutsche Begriff "Sicherheit" zwei Bestandteile ein:

- Safety bezieht sich auf die Zuverlässigkeit eines Systems, z.B. in Bezug auf dessen Ablauf- und Ausfallsicherheit.
- Security bezieht sich auf den Schutz eines Systems vor beabsichtigten Angriffen.

2.2. Aspekte

Eine sichere Kommunikation von Geräten über Netzwerke umfasst mindestens die folgenden Aspekte [7]:

- **Vertraulichkeit (Confidentiality, Privacy):** Hierunter versteht man die Sicherheit gegen unerlaubten Zugriff auf Informationen, das zu einem Informationsgewinn beim Abhörenden führt.
- **Integrität (Integrity):** Dies sind Angriffe zum Zwecke einer meist partiellen Veränderung von Informationen. In Bezug auf die Funktion eines Netzwerks kann sich die Integrität sowohl auf die Inhalte von Datenpaketen als auch auf deren Steuerinformationen und hierbei insbesondere auf die Adressierung beziehen. In diesem Sinne ist Integrität mit der Authentifizierung eng verbunden.
- **Authentifizierung (Authentication):** ist die Überprüfung, ob ein Sender wirklich derjenige ist, der dieser vorgibt zu sein. Dabei kann unterschieden werden zwischen der Authentizität des Nutzers (User), des Geräts (Device) oder der Nachricht (Message).
- **Autorisierung (Authorisation):** beschreibt die Zugangssteuerung, dass bestimmte Informationen oder Dienste nur einem beschränkten Kreis von Nutzern zur Verfügung stehen.
- **Verbindlichkeit oder Unabstreitbarkeit (Non-Repudiation):** gewährleistet die Möglichkeit, dass ein Subjekt die durchgeführten Aktionen nicht abstreiten kann.
- **Verfügbarkeit (Availability) und Zugang (Access):** Informationen sind nur dort und dann zugänglich, wo und wann sie von Berechtigten gebraucht werden. Unberechtigte haben weder einen Zugang zu den Informationen, noch sind sie in der Lage, die Verfügbarkeit der Informationen zu stören.

2.3. Maßnahmen für Security

Sicherheit kann nur erreicht werden, wenn die gesamte Architektur und ihre Umsetzung den Erfordernissen entspricht. Ein wesentlicher Baustein hierzu stellen kryptographische Verfahren dar [9], wie dies in Tabelle 1 in Bezug auf die Schutzparameter dargestellt ist, welche der Aspekte durch kryptographische Verfahren geschützt werden können.

Schutzparameter	mögliche Realisierungen	
	andere	für Netzwerk- anwendungen
Vertraulichkeit	mechanischer Schutz	Verschlüsselung
Integrität	mechanischer Schutz, Administration	Hash-Funktionen
Authentifizierung	Passwort, Biometrie	Signaturen
Autorisierung	Administration	Signaturen
Verbindlichkeit / Unabstreitbarkeit	Logging	Signaturen
Verfügbarkeit / Zugang	mechanischer Schutz	verschiedenste

Tabelle 1: Maßnahmen zum Schutz für die unterschiedlichen Parameter der Security [8]

Darüber hinaus müssen aber auch gesamthafte Betrachtungen angestellt werden, die z.B. die Schlüsselgenerierung und -verteilung, sowie die mechanische und die elektrische Sicherheit betreffen [10].

2.4. Anforderungen für Ad-Hoc Wireless Networking

Im Fall von drahtlosen Ad-Hoc-Netzwerken ist eine Reihe von zusätzlichen Anforderungen zu berücksichtigen [31]:

- Das drahtlose Medium stellt ein besonderes Sicherheitsrisiko dar, weil ein mechanischer Schutz der Kommunikation praktisch unmöglich ist.
- Auf Grund der gewünschten – und wegen der fehlenden Benutzerschnittstellen oft auch notwendigen - Ad-Hoc-Verwaltung ist ein minimaler manueller Aufwand anzustreben. Insbesondere ist eine manuelle Eingabe des Schlüsselmaterials zu vermeiden.
- Diesbezüglich ist aber zu berücksichtigen, dass damit der Austausch des Schlüsselmaterials auf die Kommunikation über das unsichere drahtlose Medium beschränkt ist.
- Die Leistungsfähigkeit der Rechnerknoten ist oft noch geringer als in vielen anderen Netzwerkknoten, so dass viele kryptographische Verfahren nicht eingesetzt werden können. Dies gilt insbesondere für die Verfahren, die mit 32 Bit langen Basisworten arbeiten.
- Viele drahtlose Knoten verfügen nur über beschränkte Energieressourcen. Gilt dies bereits für batteriegespeiste Knoten, so gewinnt die Beschränkung der Energiemenge für energieautarke Funkknoten besondere Relevanz [32] [33].
- Die Länge der Funktelegramme ist oft nur so gering, dass gängige blockbasierte Verfahren einen zu großen Overhead mit sich bringen würden. Eine

Verlängerung der Funktelegramme ist sowohl aus Gründen der Energieaufnahme als auch wegen der ansteigenden Paketfehlerrate bei gegebener hoher Bitfehlerrate zu vermeiden.

- Auf Grund der hohen geforderten Kosteneffizienz kommen oft hardware-beschleunigte Verfahren nicht in Frage [6]. Außerdem hat sich für die gegenwärtig sehr verbreiteten Zwei-Chip-Lösungen herausgestellt, dass eine Integration der Hardware-Unterstützung in den Transceiver kaum Leistungsvorteile bringt, da ein erhöhter Verkehr über die serielle SPI-Schnittstelle zwischen Mikrocontroller und Transceiver die Einsparungen wieder zunichte macht. Aus diesem Grund konzentriert sich diese Untersuchung auf software-basierte Verfahren.
- Beim Einsatz von Ad-Hoc-Management-Protokollen ist auch dieser Management-Verkehr in besonderer Weise zu schützen.

3. Untersuchung bestehender kryptographischer Algorithmen

3.1. Überblick

Von der Vielzahl der in Abb. 2 gezeigten grundsätzlichen Verfahren können für die Anwendung in drahtlosen Ad-Hoc-Netzwerken unmittelbar eine Reihe von Verfahren ausgeschlossen werden:

- Die asymmetrischen Verfahren weisen nicht nur in der Schlüsselgenerierung, die ja auch auf Hostrechnern durchgeführt werden können, sondern auch in den Ver- und Entschlüsselungsschritten eine recht hohe Komplexität auf. Deswegen werden sie in der ersten Version dieser Untersuchung nicht näher untersucht, da sie beim Einsatz von unidirektionalen Funkknoten prinzipiell identische Probleme aufweisen.
- Hinzu kommt, dass asymmetrische Verfahren sich für solche Architekturen in besonderer Weise eignen, bei denen sich die Leistungsfähigkeit der Geräte asymmetrisch verhält. Dann kann das leistungsfähigere Gerät auch die komplexere Funktion durchführen [12].

Eine solche Asymmetrie gibt es bei vielen Funksystemen nicht, da alle Knoten über den gleichen Mikrocontrollerkern verfügen.

Damit sind gegenwärtig auch die Verfahren, die auf Elliptischen Kurven basieren, ausgeschlossen.

- Verschiedenste Untersuchungen zeigen, dass die Hash-Verfahren einen Aufwand benötigen, der den von Verschlüsselungsverfahren erreicht oder sogar übersteigt (vgl. z.B. Tabelle 2). Vor dem Hin-

tergrund in Tabelle 1 gezeigten Hintergrund, dass Hash-Verfahren lediglich die Authentizität schützen, aber keine Vertraulichkeit ermöglichen, erscheinen diese wenig attraktiv. Hinzu kommt, dass die Länge des hinzuzufügenden Hash-Codes in der Regel signifikant ist.

- Bei den symmetrischen Verfahren steht eine Vielzahl von Algorithmen zur Verfügung. Diese teilen sich in strom- und blockbasierte Verfahren auf, die in den Abschnitten 3.3 und 3.4 kurz beschrieben und auf ihre Eignung für den Einsatz in Funksystemen überprüft werden.

3.2. Komplexitätsabschätzung

Bei den im Folgenden zusammengestellten Beschreibungen der wichtigsten kryptographischen Algorithmen und dem dabei erwähnten Rechenaufwand ist unbedingt zu berücksichtigen, dass der Rechenaufwand im Wesentlichen von drei Aspekten abhängt.

- Die Komplexität des Algorithmus an sich. Diese lässt sich unterteilen in
 - Anzahl und Komplexität der auszuführenden Operationen,
 - Anzahl und Komplexität der Befehlswoorte (Footprint des Programmspeichers)
 - Aufwand beim Datenspeicher (Footprint des Datenspeichers)
- Von der Qualität der Implementierung. Insbesondere die manuelle Optimierung von häufig durchlaufenen Programmteilen auf Assembler-Ebene kann wesentliche Verbesserungen mit sich bringen.
- Die Prozessorarchitektur. Hierbei ist nicht nur die Wortbreite zu berücksichtigen – zumal viele der neueren Verschlüsselungsverfahren auf 32 Bit-Architekturen ausgelegt sind, sondern auch die Speicherarchitektur. Insbesondere bei der Verwendung einfacher Mikrocontrollerkerne (wie z.B. der weit verbreiteten 8051-Architektur) macht sich der limitierte Speicheraufbau leistungsbegrenzend bemerkbar. Von besonderer Bedeutung sind hierbei:
 - die Größe des internen Speichers, der innerhalb eines Prozessorzyklus geschrieben oder gelesen werden kann, sowie
 - die Flexibilität der Akkumulatorstruktur. Wenn – wie bei vielen kleinen Mikrocontrollerkernen – nur ein Akkumulatorregister zur Verfügung steht, müssen zusätzliche Prozessorzyklen für die Ausführung von Kopierbefehle eingesetzt werden.

Diese Aspekte werden in [13] beispielhaft dargestellt.

Es ist dem Autor bewusst, dass in der folgenden Aufstellung diese Aspekte nicht unabhängig betrachtet werden. Dies ist lediglich mit der verfügbaren Quellenlage zu begründen, führt aber im Detail zu Inkonsistenzen bei der Untersuchung.

Darüber hinaus sei darauf hingewiesen, dass die meisten der Untersuchungen, die sich dem Thema Verschlüsselung auf Embedded Systemen widmen, immer noch deutlich größere Systeme im Blick haben, als sie bei den vielen Funksystemen zur Verfügung, vgl. z.B. [14].

3.3. Strombasierte Verfahren

3.3.1 Vorbemerkung

Strombasierte Algorithmen basieren auf der fortlaufenden reversiblen Verknüpfung mit einem endlos langen Schlüssel, der mit einem Schlüsselgenerator erzeugt wird [15].

In Bezug auf den Einsatz in Funksystemen weisen sie den Vorteil auf, dass u.U. auch nur geringe Datenmengen verschlüsselt werden können.

In der Regel gelten Stromchiffren als effizienter umzusetzen, aber auch mit einem geringeren Sicherheitsniveau verbunden. Allerdings ist hierbei zu berücksichtigen, dass bei einigen Stromchiffren u.U. recht starke Abhängigkeiten zwischen dem verwendeten Schlüssel und den verschlüsselten Daten auftreten können. Diese Problematik ist insbesondere beim RC4-Algorithmus in der Umsetzung in dem Wired Equivalent Privacy-(WEP)-Protokoll des IEEE802.11 bekannt [16] [17]. Sie lässt sich allerdings durch eine geeignete Wahl der Initialisierungsvektoren (IV) umgehen.

Zusätzlich ist zu beachten, dass die meisten Implementierungen auch von Stromverschlüsselungsverfahren im Embedded Bereich in Hardware erfolgen, da die Schiebeoperationen sehr effizient in Hardware umgesetzt werden können. In Software ist ihre Implementierung häufig weniger effizient.

Tabelle 2 zeigt einen Vergleich über die Effizienz von verschiedenen Verfahren, wobei sowohl strom- als auch blockbasierte, sowie Hash-Verfahren zum Einsatz kommen. Als Plattform für den Technologievergleich wurde mit einem 90 MHz Pentium Prozessor ein zum Zeitpunkt der Veröffentlichung leistungsfähiger PC-Prozessor verwendet. Die Algorithmen wurden in optimiertem Assembler implementiert [20].

Algorithm	Mbit/s	Relative speed
SEAL	198	1.0
RC4	110	1.8
RC5-32/12	38.4	5.2
DES	16.9	11.7
MD5	133.1	1.5

Tabelle 2: Rechenaufwand für die Embedded Implementierung verschiedener kryptographischer Algorithmen [20]

In [18] werden die gängigen Stromverschlüsselungsverfahren A5, ARC4, SEAL, SCOP und PIKE auf ihre Effizienz und Eignung in Embedded Systemen hin untersucht. Hierbei treten vor allem SCOP [19] und ARC4 in Bezug auf ihre Energieeffizienz heraus. Während SCOP eine gute Energieeffizienz bei längeren Datenströmen aufweist, wird der RC4 hervorgehoben, da er auch mit kurzen Datenströmen betrieben werden kann.

3.3.2 RC4

RC-4 wurde von Ron Rivest entwickelt. RC steht dabei für Ron's Code oder Rivest Cipher. Es handelt sich um ein Stromverschlüsselungsverfahren, bei dem mit einem sehr einfachen Verfahren eine lange pseudozufällige Folge von Zeichen erzeugt wird, die als Schlüssel mit dem Klartext XOR-verknüpft wird. Die Schlüssellänge kann von einem bis zu 256 Oktetten reichen. RC-4 speichert 258 verschiedene Zustandsinformationen. 256 dieser Oktette sind Permutationen von 0, 1 bis 255 (und somit gleichverteilt). Der Key-Scheduling-Algorithmus (KSA) erzeugt eine initiale Permutation. Der Pseudo-Random-Generation-Algorithmus (PRGA) generiert hieraus den Bitstrom, mit dem der Klartext verschlüsselt wird.

Mit einer C-basierten Implementierung wurde die in Tabelle 3 gezeigten Werte für die Prozessorzyklen auf einem 8051-kompatiblen Prozessor gemessen. Es sei darauf hingewiesen, dass unter den Prozessorzyklen hierbei nicht die Oszillatorzyklen, sondern beim gegebenen Derivat die durch 12 geteilten Oszillatorzyklen zu verstehen sind. Aus Tabelle 3 lassen sich folgende Aussagen ableiten:

- Der größte Aufwand wird für die Initialisierung verbraucht.
- Der Aufwand für diese Initialisierung ist praktisch unabhängig von der Länge des Schlüssels. Er sinkt sogar leicht mit steigender Schlüssellänge (vgl. Abb. 3).
- Die gesamte Komplexität für eine Initialisierung erreicht gut 100 ms im Falle eines 12 MHz-8051-Mikrocontrollers.

- Der Aufwand für die Verschlüsselung selbst (XOR-Verknüpfung) ist praktisch linear abhängig von der Länge des Datenstroms (vgl. Abb. 4).
- Der Aufwand für die Verschlüsselung selbst (XOR-Verknüpfung) ist unabhängig von der Länge des verwendeten Schlüssels.

In [13] wird eine eigene Implementierung vorgestellt, die einen Durchsatz von 98.765 Bytes / s erreicht. Diese basiert allerdings auf einem ATMega32 Prozessor, bei dem 258 Bytes RAM unmittelbar angesprochen werden.

3.3.3 eSTREAM

Unter eSTREAM [22] versteht man das ECRYPT Stream Cipher Project. Hierbei ist ECRYPT das "European Network of Excellence for Cryptology", das im Rahmen eines vierjährigen Programms im Rahmen des Information Societies Technology (IST) Programm des sechsten Rahmenprogramms (FP6) der europäischen Kommission gefördert wird [21].

In diesem Projekt werden verschiedene neue Stromchiffren im Rahmen eines Wettbewerbs evaluiert und verglichen. Ein Hauptaugenmerk liegt dabei auch auf der effizienten Implementierbarkeit auch auf kleineren Mikroprozessoren und -controllern.

key-length [Bytes]	text-length [Bytes]	rc4_setup [ProcCycles]	rc4_encrypt [ProcCycles]
8	2	103055	1037
8	4	103055	1806
8	8	103055	3344
8	16	103055	6420
8	32	103055	12572
8	64	103055	24876
8	128	103055	49483
16	2	102912	1037
16	4	102912	1806
16	8	102912	3344
16	16	102912	6420
16	32	102912	12572
16	64	102912	24876
16	128	102912	49483
32	2	102840	1037
32	4	102840	1806
32	8	102840	3344
32	16	102840	6420
32	32	102840	12572
32	64	102840	24876
32	128	102840	49483
64	2	102804	1037
64	4	102804	1806
64	8	102804	3344
64	16	102804	6420
64	32	102804	12572
64	64	102804	24876
64	128	102804	49483
128	2	102786	1037
128	4	102786	1806
128	8	102786	3344
128	16	102786	6420
128	32	102786	12572
128	64	102786	24876
128	128	102786	49483

Tabelle 3: Rechenaufwand für eine C-basierte RC4-Implementierung

In diesem Projekt werden verschiedene neue Stromchiffren im Rahmen eines Wettbewerbs evaluiert und verglichen. Ein Hauptaugenmerk liegt dabei nach Auskunft der Veranstalter auch auf der effizienten Implementierbarkeit auf kleineren Mikroprozessoren und -controllern.

In [37] findet sich eine Untersuchung von verschiedenen sESTREAM-Kandidaten in Bezug auf die Implementierung auf 8-Bit-Mikrocontrollern. Für die Evaluation wurde eine AVR-Architektur gewählt, die für ihre gute Effizienz und Speicherverwaltung bekannt ist.

Tabelle 4 und Abbildung 5 zeigen die wichtigsten Ergebnisse dieser Untersuchung.

Cipher	block size [byte]	Encryption [cycles]	Encryption [cycles/byte]	Throughput [bytes/sec]
AES	16	12574	785.88	10179
Dragon	128	24227	189.27	42267
HC-128	64	10804	168.81	47390
LEX	40	8061	201.53	39696
Salsa20	64	90802	1418.78	5638
Salsa20 improved	64	48942	764.72	10461
SOSEMANUK	80	14134	176.68	45279

Tabelle 4: Rechenaufwand für die Verschlüsselung auf einem 8 Bit-AVR_Mikrocontroller [37]

Diese lassen sich wie folgt zusammenfassen:

- LEX stellt sich in Bezug auf die untersuchten Implementierungen als der effizienteste Algorithmus heraus.
- Herkömmliche und bewährte blockbasierte Verfahren, wie z.B. AES (vgl. Abschnitt 3.4.3) benötigen jedoch nur 50 % mehr Taktzyklen.
- Andere Algorithmen, wie z.B. der Salsa20, der auf 32 Bit-Architekturen über die beste Leistungsfähigkeit verfügt, benötigen hingegen auch auf der gewählten flexiblen 8 Bit-Architektur mehr als 5-mal so viele Taktzyklen.

3.3.4 Salsa20

Bei dem Algorithmus Salsa20 [35] [36] handelt es sich um einen ursprünglichen Hash-Algorithmus, dessen Ausgabe mit dem Datenstrom XOR-verknüpft wird.

[34] untersucht die Möglichkeiten, den Salsa 20-Algorithmus auf einem 8051 zu implementieren. Hierbei muss man allerdings feststellen, dass die extensive Nutzung von 32 Bit langen Wörtern und die limitierten Speicher- und Akkumulator-Ressourcen auf dem 8051 zu einem übermäßigen Anstieg der Kopierbefehle führen. Wie in Abb. 1 dargestellt, gilt dies auch dann, wenn folgende weitere Optimierungen vorgenommen werden:

- Bei der „pipeline friendly“-Realisierung handelt es sich um eine Implementierung, die für den Einsatz auf Mikroprozessoren mit Pipelines optimiert wurde. Auch wenn der verwendete 8051 nicht mit einer Pipeline ausgestattet ist, macht sich eine leichte Verbesserung der Performance bemerkbar.
- Die Variante „fast rolls and shift“ nutzt optimierte Schiebepfehle. Insbesondere werden hierbei Schiebepfehle um mehr als 8 Bit auf Kopiervorgänge von Bytes zurückgeführt.
- In einem dritten Schritt werden nun noch möglichst viele der Datenworte in den internen indirekt adressierbaren Speicherbereich gelegt.

Mit allen diesen Maßnahmen kann eine Durchlaufzeit von 144 ms erreicht werden.

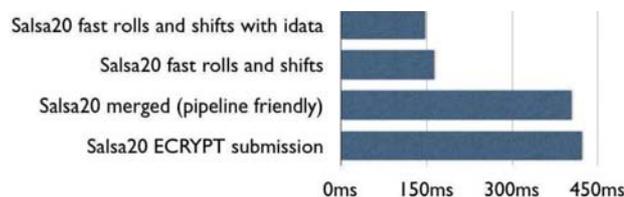


Abbildung 1: Zeitaufwand des Salsa20-Algorithmus auf einem 8051-Mikrocontroller mit 8 MHz [34]

3.4. Blockbasierte Verfahren

3.4.1 Vorbemerkung

Blockbasierte Algorithmen arbeiten mit Blöcken fester Länge. In Bezug auf den Einsatz in Funksystemen sind die Verfahren vorzuziehen, die einerseits möglichst wenig Daten verschlüsseln, hierfür andererseits aber einen möglichst langen Schlüssel verwenden.

3.4.2 DES/3DES

Der immer noch weit verbreitete Data Encryption Standard in der einfachen Durchführung (DES) oder der dreifachen Wiederholung (Triple DES, 3DES) verwendet Blocklängen von 64 Bit. Die Schlüssellänge pro Durchführung beträgt 56 Bit. Bei 3DES kann mit drei Schlüsseln (K1, K2, K3) oder auch mit zwei Schlüsseln mit K1 = K3 betrieben werden, wodurch sich resultierende Schlüssellängen von 56, 112 und 168 Bit ergeben.

Trotz der großen Verbreitung von DES und 3DES in vielen SmartCard- und Netzwerksystemen, wie IPsec oder SSH/SSH2, sind viele Angriffe auf diese Algorithmen bekannt [23], die nahe legen, dass kein signifikanter Schutz gegeben ist. Darüber hinaus zeigt Tabelle 6 den höheren Rechenaufwand von DES im Vergleich zu AES.

3.4.3 AES

In vielen drahtlosen Systemen findet heutzutage der Nachfolger von DES-Einsatz.

Dieser Advanced Encryption Standard (AES) [24] wurde 1998 in einem Wettbewerb ausgewählt, der auch zahlreiche weitere renommierte Algorithmen hervorgebracht hat. Aus diesen wurden 1998 fünf Finalisten ausgewählt. Hierzu zählten Twofish, Serpent, Rijndael, RC6 und MARS.

Als Gewinner des Finales ging der blockorientierte Algorithmus Rijndael von Joan Daemen und Vincent

Rijmen hervor, der heutzutage meist nur AES, manchmal auch als AES-Rijndael bezeichnet wird.

AES verarbeitet Datenblöcke einer Länge von 128 Bit unter Einsatz von 128, 192 oder 256 Bit langen Schlüsseln.

Auf Grund der immensen Verbreitung des AES-Algorithmus liegen verschiedenste Implementierungen in Hardware und Software vor [30]. AES wurde seinerzeit auch ausgewählt, weil seine Implementierung im Vergleich zu anderen blockbasierten Algorithmen auch auf kleinen Mikroprozessoren und –controllern möglich ist. Dies ist insbesondere darauf zurück zu führen, dass [13]:

- Erstens verwendet er als grundlegende Datenstruktur ein Byte, im Gegensatz zu vielen anderen Algorithmen die 32-BitWorte verwenden. Dadurch entfallen die in vielen anderen Algorithmen erforderlichen Überträge bei Additionen, Multiplikationen und Rotationen.
- Zweitens werden, falls die S-Box vorab berechnet wurde, überwiegend XOR und Additionen als Operationen verwendet. Die Schiebevorgänge der Shiftrows-Transformationen sind immer Verschiebungen um ein Vielfaches von 8 Bit, so dass sie sich durch Kopiervorgänge, beziehungsweise durch einen Offset bei der Adressierung ersetzen lassen.
- Drittens kann der Rundenschlüssel während einer Runde aus dem Schlüssel der vorherigen Runde berechnet werden, so dass eine Berechnung des Rundenschlüssels während des Ver-, beziehungsweise Entschlüsselungsvorganges durchgeführt werden kann. Dies ermöglicht die Implementierung auf Mikrocontrollern mit sehr kleinem Hauptspeicher, der nicht alle Rundenschlüssel gleichzeitig aufnehmen kann.

Den Vorteil von AES im Vergleich mit anderen blockbasierten Algorithmen zeigt Tabelle 6 [25].

(Key, block length)	Code size	Number of cycles	Execution time
	(bytes)		(msec)
	768	4065	4.1
(128,128)	826	3744	3.7
	1016	3168	3.2
(192,128)	1125	4512	4.5
(256,128)	1041	5221	5.2

Tabelle 5: Rechenaufwand für Implementierungen von AES auf einem 8051-Mikrocontroller [26]

Zwei weitere Implementierungen mit noch günstigerer Performance sind unter [27] und [26] erwähnt. Die

beiden Implementierungen stellen ein typisches Trade-Off zwischen Codegröße und Leistungsfähigkeit dar.

Hierbei ist hervorzuheben, dass eine Implementierung mit 3744 Zyklen, wie sie als Open-Source-Software vorliegt 16Byte Text (State) mit einem 16 Byte Key (Round) verschlüsselt, auf einem 8051-Mikrocontroller bei einer Taktfrequenz von 12 MHz einen Zeitaufwand von etwa 3,7 ms beansprucht. Dies erscheint als eine akzeptable Größe, die den „Austausch“ eines Datenprogramms auf dem Funkkanal gegen die Verschlüsselung der Daten erlaubt.

Dies gilt auch dann, wenn außerdem noch die Prozessorzyklen berücksichtigt werden, um die Daten in den internen Speicher zu kopieren. Einschränkend muss weiterhin berücksichtigt werden, dass eine unmittelbare Implementierung auf Grund der großen Blocklänge nicht sinnvoll erscheint. Allerdings erscheint eine erweiterte Implementierung im Sinne einer strombasierten Verwendung attraktiv (vgl. Kap. 3.5).

Zu erwähnen ist auch, dass - ungeachtet der Tatsache, dass es sich um ein symmetrisches Verschlüsselungsverfahren handelt – die Entschlüsselung einen etwas höheren Rechenaufwand mit sich bringt [28]. Dies ist im Wesentlichen darauf zurück zu führen, dass beim inversen Verfahren zwar die identische Grundstruktur, aber ein anderes Polynom verwendet wird. Der zusätzliche Rechenaufwand wird in [28] mit etwa 30 % angegeben.

Ebenfalls zu erwähnen ist die Tatsache, dass auch beim AES-Algorithmus die Möglichkeit besteht, die Anzahl der Runden zu reduzieren, um eine geringere Laufzeit zu erzielen. Dies geschieht aber im Trade-Off mit der Sicherheit. So ist z.B. ein Angriff auf einen AES-Algorithmus mit sieben Runden bekannt [29].

3.5. Strombasierte Verfahren unter Nutzung blockbasierter Algorithmen

Der Nachteil der relativ langen benötigten Wortlängen beim Einsatz blockbasierter Verfahren lässt sich dadurch ausgleichen, dass der aus einem blockbasierten Algorithmus erhaltene Datenblock mit dem eigentlichen Datenstrom XOR-verknüpft wird. Damit wird der aus dem blockbasierten Verfahren abgeleitete Datenblock praktisch als Input für ein Stromverschlüsselungsverfahren verwendet. Hierzu muss das blockbasierte Verfahren ebenfalls mit einem Datenblock gefüllt werden. Im einfachsten Fall ist dieser Datenblock fest im Bauelement vorgegeben.

Um sich zusätzlich gegen Replay-Angriffe schützen zu können, erscheint es sinnvoll, einen Initialisierungsvektor vorzusehen, dessen Durchlauf z.B. mit

Hilfe eines Rolling-Code-Verfahrens pseudo-zufällig erzeugt wird. Dies ist in Abb. 6 dargestellt.

4. Ausblick

Auf der Grundlage der beschriebenen Untersuchungen werden folgende Aussagen, bzw. Empfehlungen getroffen:

- Bei der Auswahl einer Software-basierten Lösung sollte darauf geachtet werden, dass eine Hardware-Implementierung in einer zukünftigen ASIC-Implementierung möglich wäre.
- Gängige Hash-Verfahren leider darunter, dass die hinzuzufügenden Blocklängen im Vergleich zu den Funktelegrammen viel zu groß sind.
- Gleiches gilt auch für die gängigen blockbasierten Verschlüsselungsalgorithmen.
- Stromverschlüsselungsverfahren weisen dieses Problem nicht auf. Die untersuchten Algorithmen bringen jedoch eine deutlich zu hohe Komplexität mit sich.
- Aus diesem Grund wird ein blockbasiertes Verschlüsselungsverfahren empfohlen, dessen Ausgabe für eine strombasierte Verschlüsselung genutzt wird. Unter Verwendung eines Initialisierungsvektors lässt sich somit eine sinnvolle Verschlüsselung mit einem effizienten Schutz gegen Replay Angriffe verbinden.
- Der benötigte Rechenaufwand auf der Verschlüsselungsseite liegt im Bereich von 5 ms, so dass ein Datengramm auf dem Kanal gegen die Sicherheit auf der Übertragungsstrecke „eingetauscht“ werden kann.

5. Quellenangaben

- [1] Sikora, A., Brügger, P., "Virtual Private Infrastructure - An Industry Consortium for Unified and Secure Web Control with Embedded Devices", 9th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2003), Lisbon, Portugal.
- [2] A. Sikora, "Short Range Wireless Networks – spoiled for choice", Embedded World Conference, 13.-15.2.2007, Nuremberg.
- [3] D. Rahusen, A. Sikora, "IPv6 over IEEE802.15.4 – Status and Trends", Embedded World Conference, 13.-15.2.2007, Nuremberg.
- [4] A. Klapproth, T. Bürkli. „TCP/IP über IEEE 802.15.4“, Design & Elektronik Entwicklerforum, 4.4.2006, München.
- [5] N. Braun, D. Lill, A. Sikora, "Webanschluss inklusive - Neue Gerätekategorie "Appliances" steht zwischen Embedded-Geräten und PC - neue Chancen, neue Anwendungen", Elektronik 4/2002, S., 86-92.
- [6] Sikora, A., "Hard- und Softwarelösungen für sichere Embedded Systeme", Design&Elektronik-Entwicklerforum "Drahtlose & drahtgebundene Netzwerke für Industrie & Automotive", München, 7. Juli 2004.
- [7] Sikora, A., "Technische Grundlagen der Rechnerkommunikation: Internet-Protokolle und Anwendungen", Fachbuchverlag Leipzig im Carl Hanser Verlag, 2003, ISBN 3-446-22455-6.
- [8] Straub, T., Sikora, A., "Cryptography on Embedded Systems", embedded world 2005 Conference, Nürnberg, S. 517-536.
- [9] Kaufman, C., Perlman, R., Speciner, M., "Network Security", Prentice-Hall, 2002, ISBN 0-13-046019-2.
- [10] H. Adamczyk, "Sichere drahtlose Kommunikation im industriellen Umfeld mit ZigBee", at Automatisierungstechnik, erscheint im Mai 2007.
- [11] Braun, N., Gillen, T., Sikora, A., "Gerätekommunikation via Virtual Private Networks (VPN)", Elektronik 8/2006, S. 58-65.
- [12] H. Gaubatz, J.-P. Kaps, B. Sunar, „Public Key Cryptography in Sensor Networks - Revisited“, in: C. Castellucia et.al. (eds.), „Security in Ad-hoc and Sensor Networks“, ESAS, LNCS3313, pp. 2-18, 2005.
- [13] U. Wurst, "Einsatz kryptographischer Verfahren auf stark ressourcenbeschränkten Geräten“, Diplomarbeit Nr. 2106, Universität Stuttgart, 2003.
- [14] S. Seys, "Cryptographic Algorithms and Protocols for Security and Privacy in Wireless Ad Hoc Networks“, Dissertation, May 2006, KU Leuven.
- [15] A. Menezes, P. van Oorschot, S. Vanstone, "Handbook of Applied Cryptography“, CRC Press, 1996, <http://www.cacr.math.uwaterloo.ca/hac>.
- [16] S. Fluhrer, I. Mantin, A. Shamir, "Weaknesses in the key scheduling algorithm of RC4“, English Annual Workshop on Selected Areas in Cryptography (08/2001).
- [17] Stubblefield, Ionnidis, Rubin, "Using the Fluhrer, Mantin and Shamir Attack to Break WEP“, AT&T Labs Technical Report TD-4ZCPZZ (08/2001), http://www.wisdom.weizmann.ac.il/~itsik/RC4/Papers/wep_attack.ps.

- [18] O.T. Bethge, "Untersuchung der Effizienz von Stromchiffren auf eingebettete Prozessoren", Diplomarbeit Ruhr-Universität Bochum, 2003: http://www.crypto.ruhr-uni-bochum.de/imperia/md/content/texte/thesen/da_bethge.pdf.
- [19] S.V. Maltchev, P.T. Antonov, "The SCOP Stream Cipher", December, 1997 <http://www.geocities.com/smaltchev/scop.html>
- [20] Phillip Rogaway, D. Coppersmith, "A Software Optimized Encryption Algorithm", Journal of Cryptology, 1998, <http://seclab.cs.ucdavis.edu/papers/Rogaway/seal.pdf>.
- [21] "Network of Excellence in Cryptology", IST-2002-507932", <http://www.ecrypt.eu.org/>
- [22] "eSTREAM PHASE 2", <http://www.ecrypt.eu.org/stream/>
- [23] Grabbe, J.O., "The DES Algorithm Illustrated", <http://orlingrabbe.com/des.htm>.
- [24] NIST FIPS Pub 197: Advanced Encryption Standard (AES), Federal Information Processing Standards Publication 197, U.S. Department of Commerce/N.I.S.T., November 2001, <http://www.ntis.org/>.
- [25] Vincent Journot, "Evaluation of Serpent, one of the AES-finalists, on 8-bit microcontrollers", May 11, 2000, <http://www.cl.cam.ac.uk/~mgk25/download/serpent-8051.tar.gz>
- [26] http://www.usenix.org/publications/library/proceedings/smartcard99/full_papers/daemen/daemen_html/
- [27] "AES: Timings of the best known implementations", <http://www.di.ens.fr/~granboul/recherche/AES/timings.html#Rij%208051%20Caesar>
- [28] J. Daemen, V. Rijmen, "AES Proposal: Rijndael", verfügbar unter (25.1.07): <http://www.iaik.tugraz.at/research/krypto/AES/old/~rijmen/rijndael/rijndaeldocV2.zip>
- [29] N. Ferguson et.al., "Improved cryptanalysis of Rijndael", In: "Fast Software Encryption", 7th International Workshop, FSE 2000, New York, NY, USA, volume 1978 of Lecture Notes in Computer Science, pages 213-230, Springer, 2001.
- [30] <http://www.iaik.tugraz.at/research/krypto/AES/old/~rijmen/rijndael/>
- [31] Y. Xiao, J. Li, Y. Pan (Hrsg.), "Security And Routing in Wireless Networks, Nova Biomedical, 2005.
- [32] F. Schmidt, W. Heller, „Energieautarke Funksensoren: von verrückten Ideen zu Massenmarkt-Produkten“, in: Sensoren und Messsysteme, 13. ITG/GMA-Fachtagung, März 2006, Freiburg.
- [33] R. Thomasius et.al. „Entwurf kleinster energieautarker Funksensoren“, in: Sensoren und Messsysteme, 13. ITG/GMA-Fachtagung, März 2006, Freiburg.
- [34] M. Bölter, D. Schauenberg, "Power efficiency measurement of cryptographic algorithms for embedded networking applications", Studienarbeit im Studiengang Informationstechnik der Berufsakademie Lörrach, 29. Januar 2007.
- [35] "Salsa20 Stream Cipher", <http://cr.yp.to/snuffle.html>
- [36] "Salsa20 Code – ECRYPT Submission", http://www.ecrypt.eu.org/stream/p2ciphers/salsa20/salsa20_p2source.zip
- [37] G. Meiser, T. Eisenbarth, K. Lemke-Rust, C. Paar, „Software Implementations of eSTREAM Profile I Ciphers on 8-bit AVR Microcontrollers“, The State of the Art of Stream Ciphers, SASC 2007, Bochum.

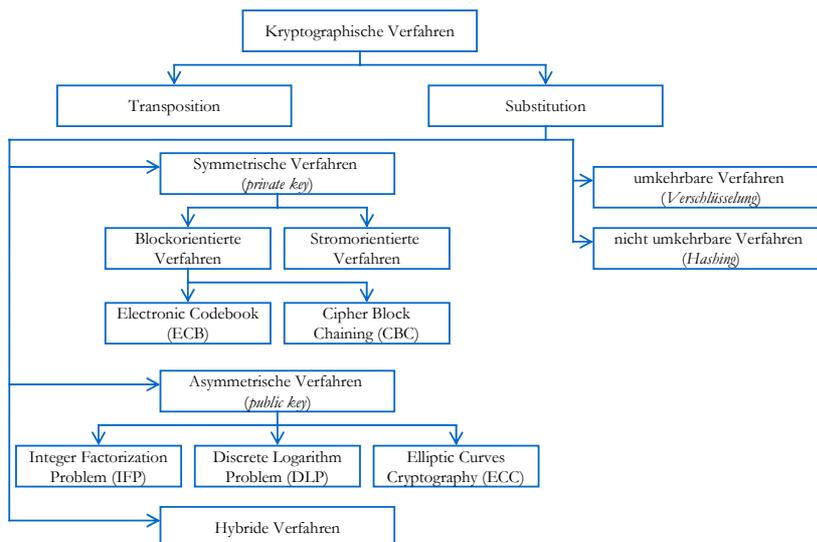


Abbildung 2: Übersicht über kryptographische Verfahren, nach [11]

Candidat	RAM used (bytes)	code size (bytes)	Number of machine cycles
MARS seems to be hardly implementable on 8-bit microcontrollers			
RC6	77	706	58056
RIJNDAEL	54	826	3977
SERPENT	56	2021 (7130)	70339 (60300)
TWOFISH	52	2063	19819
DES			6000
TRIPLE DES			18000

Tabelle 6: Rechenaufwand für Implementierungen der AES-Kandidaten und DES, bzw. 3DES auf einem 8-Bit Mikrocontroller [25]

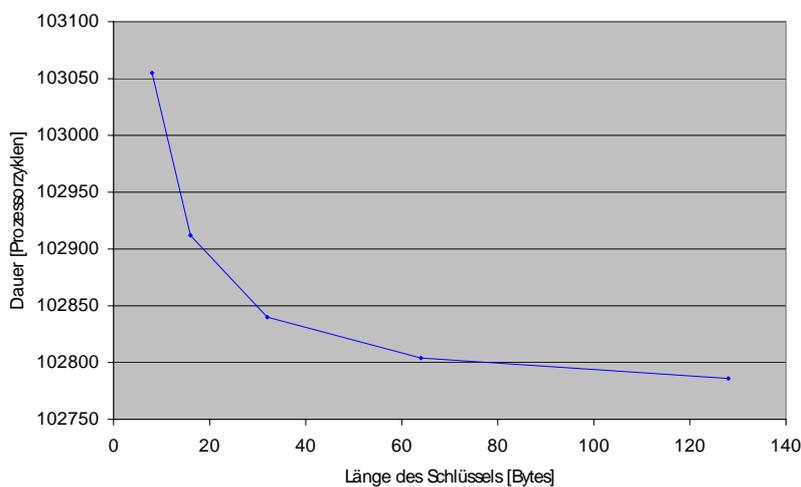


Abbildung 3: Dauer der RC4-Setup-Phase in Abhängigkeit von der Schlüssellänge

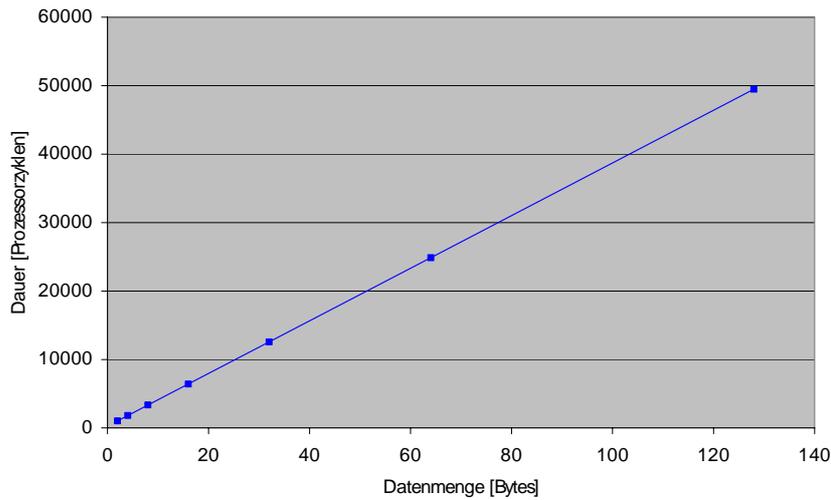


Abbildung 4: Dauer der RC4-Verschlüsselung in Abhängigkeit von der Datenmenge

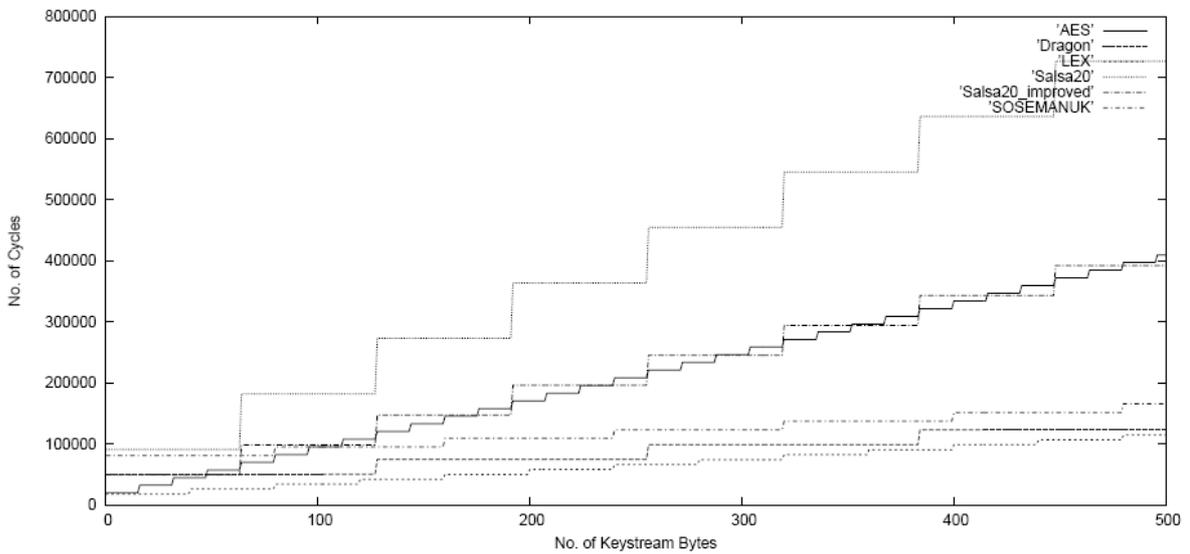


Abbildung 5: Rechenaufwand für die Verschlüsselung auf einem 8 Bit-AVR_Mikrocontroller [37]

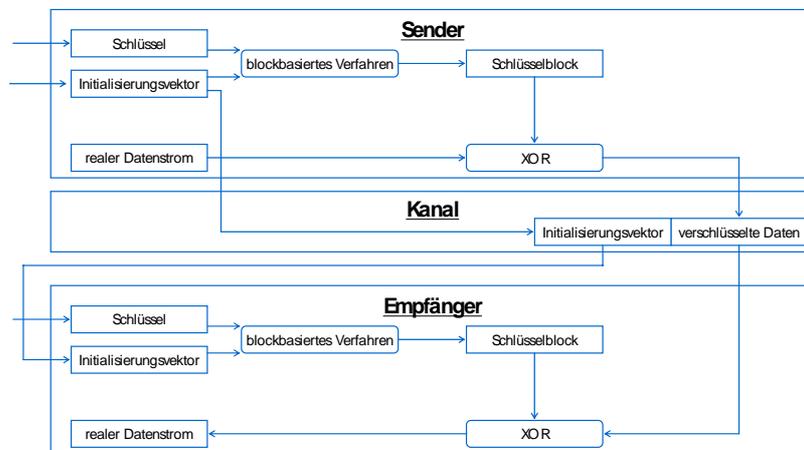


Abbildung 6: Verwendung eines blockbasierten Verschlüsselungsverfahrens für eine strombasierte Verschlüsselung unter Verwendung eines Initialisierungsvektors

GMSK Demodulation - Realisierung auf FPGA Basis

Markus Hummel

Hochschule Heilbronn, Max-Planck-Str. 39, 74081 Heilbronn

Im Bereich der Telekommunikation sind FPGA in digitalen Endgeräten wie Mobiltelefonen oder auch in den Vermittlungsstellen der Telekommunikationsunternehmen zu finden. Im Automobilbereich kommen eingebettete Systeme bei der Motorregelung, der Bremsensteuerung oder der Fahrzeugnavigation zum Einsatz.

Schnelle Kommunikation ist in der heutigen Zeit sehr wichtig geworden. Seit der ersten Übertragung von Informationen mit Hilfe elektrischer Signale von Samuel Finley Morse im Jahre 1837 hat sich in der Telekommunikationsbranche einiges verändert. Von der Entwicklung des Telefons und dem Aufbau eines Telefonnetzes, über die Möglichkeit Signale über sehr weite Distanzen zu übermitteln bis hin zur Satellitenkommunikation war es ein weiter Weg.

Heute ist es Stand der Technik, Signale mit Hilfe digitaler Modulationsverfahren zu übertragen. Digitale Übertragungsverfahren haben heute großartige Möglichkeiten in Bezug auf die Übertragungseffizienz, der Bandbreite Nutzung und die Qualitätssicherheit der Übertragung.

Gegenstand dieser Veröffentlichung ist die Datenrückgewinnung aus einem GMSK Signal. Die Hauptaufgabe ist die Implementierung von Embedded VHDL Software auf einem geeigneten FPGA. Bei dem FPGA Board handelt es sich um das Evaluationsboard Spartan 3 von Xilinx.

FPGAs eignen sich ideal für die Implementierung der Verbindungslogik, z.B. für durchzuführende Formatkonvertierungen oder auch zur Verknüpfung unterschiedlichster Schnittstellen, die benötigt werden, um Prozessoren mit Echtzeit-Peripherie zu verbinden, z. B. Modems, A/D-Umsetzer und digitale Empfänger- / Sendebau- steine. Mit den Möglichkeiten der Realisierung von signalverarbeitenden Algorithmen wird der Einsatz von FPGAs um die Auslagerung von rechenzeitaufwendigen Funktionen der digitalen Signalverarbeitung erweitert. Begünstigt wird dies durch eine hochgradig parallele Implementierung eines Algorithmus im FPGA und durch eine stetig

wachsende Anzahl von Logikgattern in neuen FPGA-Generationen.

1. Die Übertragungsmethode GMSK

1.1. Einführung

GMSK ist die Abkürzung für Gaussian Minimum Shift Keying. Dieses Verfahren wird beim Mobilfunkstandard GSM (Groupe Spéciale Mobile) eingesetzt. Heute lautet die Bezeichnung Global System for Mobile Communications. Es handelt sich um ein Standard für voll-digitale Mobilfunknetze und ist der weltweit am meisten verbreitete Mobilfunk-Standard.

GMSK kann als kontinuierliches Phasenmodulationsverfahren beziehungsweise als digitales Frequenzmodulationsverfahren betrachtet werden. In der Phasenänderung des Signals ist die Information, ob eine „1“ oder eine „0“ gesendet wurde, enthalten.

Die Phase ist zu keinem Zeitpunkt konstant sondern ändert sich kontinuierlich.

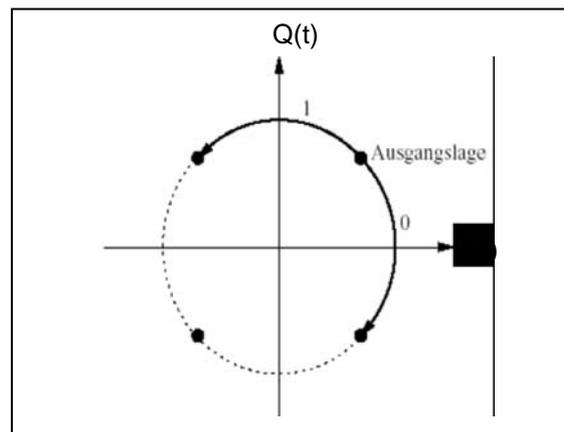


Abbildung 1: Demodulationsregel

Bei der Übertragung spielt es keine Rolle, ob es sich bei der Übertragungsstrecke um eine drahtgebundene oder um eine Funkstrecke handelt. Die Schwierigkeit

liegt nicht in dem Medium, das die Daten überträgt, sondern im Aufwand, der betrieben werden muss, um die Daten effizient zu übertragen.

1.2. GMSK Impulsformung

Im GMSK Verfahren wird aus einem rechteckigen Impuls eines Bits ein abgerundeter Impuls geformt. Dies geschieht mit Hilfe eines digitalen Filters mit Gaußglockenkurvencharakteristik. Der Verlauf im Zeitbereich stellt sich wie folgt dar:

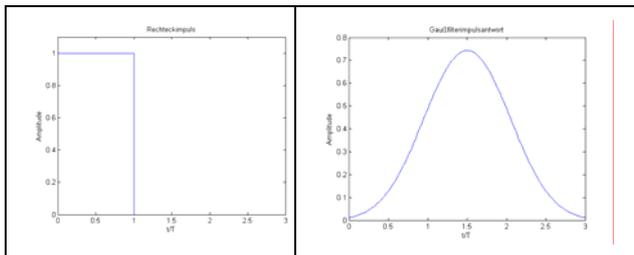


Abbildung 2: Impulsformung

Aus dem rechteckigen Impuls

$$\gamma_{rect}(t) = \text{rect}\left(\frac{t}{T}\right)$$

ist durch den Gauß-Tiefpass ein Gaußimpuls

$$\gamma_{GMSK}(t) = \frac{1}{2} \left[\text{erf}\left(\sqrt{\frac{2}{\ln 2}} \cdot \pi \cdot f_{3dB} \cdot T \cdot \left(\frac{t}{T} + \frac{1}{2}\right)\right) - \text{erf}\left(\sqrt{\frac{2}{\ln 2}} \cdot \pi \cdot f_{3dB} \cdot T \cdot \left(\frac{t}{T} - \frac{1}{2}\right)\right) \right]$$

geworden. Der Impuls erstreckt sich über ungefähr drei Symbolauern.

1.3. Die Spektrum - Auswirkungen

Je größer die Symbolinterferenz ist, desto sanfter bzw. runder sind die Übergänge zwischen den einzelnen Symbolen und desto niedriger sind die benötigten Frequenzen. Kann ein Teil der hohen Frequenzen eingespart werden, benötigt man weniger Bandbreite. Grundsätzlich sind alle Frequenzen bis zu einer höchsten Frequenz in einem Signal enthalten. Je tiefer die höchste Frequenz angesetzt werden kann, umso geringer wird die benötigte Bandbreite.

Die im Signal enthaltenen Frequenzen beeinflussen die Anstiegszeit der Signalamplitude. Die Amplitude bei einem binären Signal ist entweder „0“ oder „1“. Wenn die Anstiegszeit durch die maximal zulässige Höchsfrequenz begrenzt wird, führt das nicht dazu, dass die Amplitude kleiner ausfällt. In Wirklichkeit fängt das Signal früher an zu steigen, damit es den

nominellen Signalpegel erreicht. Daraus ergeben sich Symbolinterferenzen. Je kleiner die höchste Frequenz, desto größer die Symbolinterferenz.

In einer Grafik soll der Unterschied zwischen binären Daten, dem MSK (MSK (= Minimum Shift Keying) gleiches Phasenmodulationsverfahren ohne Gaußfilterung, das heißt ohne Symbolinterferenzen) und dem GMSK Verfahren im Spektrum aufgezeigt werden. Auf den folgenden Abbildungen sind qualitativ die Spektren der verschiedenen Signale dargestellt.

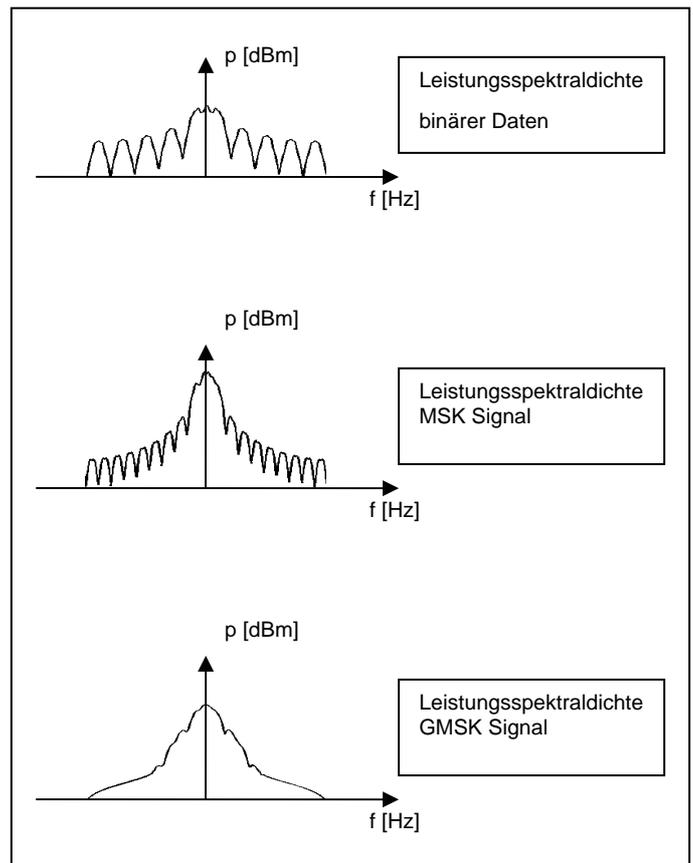


Abbildung 3: Vergleich von Leistungsspektraldichten

1.4. Das Basisbandsignal

Das Basisbandsignal wird mathematisch in folgender Form ausgedrückt:

Allgemein gilt für das Basisbandsignal

$$s(t) = a(t) \cos(2\pi f_0 t + \varphi(t))$$

Bei einem frequenzmodulierten Signal wird die momentane Frequenz durch



$$f(t) = f_0 + k_f m(t)$$

gegeben. Es gilt: $a(t) = A$ ist eine positive Konstante. Sie zeigt, dass die Frequenz in der Umgebung von f_0 mit Hilfe des Modulationssignals $m(t)$ geändert wird. Die Gewichtung k_f ergibt die Stärke oder Tiefe der Modulation. Durch Integration der momentanen Frequenz erhält man die momentane Zeitfunktion der Phase $\varphi(t)$

$$\varphi(t) = 2\pi \int_0^t (f_0 + k_f m(\tau)) d\tau + \varphi_0 = 2\pi f_0 t + 2\pi k_f \int_0^t m(\tau) d\tau + \varphi_0$$

die dann das frequenzmodulierte Signal $s(t)$ definiert:

$$s(t) = A \cos(\varphi(t)) = A \cos(2\pi f_0 t + 2\pi k_f \int_0^t m(\tau) d\tau + \varphi_0)$$

$$= A \cos(2\pi f_0 t + \varphi(t))$$

Die allgemeine Form eines Basisbandsignals kann durch Erweiterung zu einer zweiten Darstellungsform gebracht werden:

$$s(t) = a(t) \cos(\varphi(t)) \cos(2\pi f_0 t) - a(t) \sin(\varphi(t)) \sin(2\pi f_0 t)$$

$$= i(t) \cos(2\pi f_0 t) - q(t) \sin(2\pi f_0 t)$$

Die Signale $i(t)$ der Inphase Komponente und $q(t)$ der Quadratur Komponente, gegeben durch

$$i(t) = m(t) \cos(\varphi)$$

$$q(t) = m(t) \sin(\varphi)$$

bilden die so genannten Quadratur-Komponenten des Basisbandsignals $s(t)$.

2. Umsetzung auf der Hardware

Nach der Aufbereitung und Abtastung des analogen Basisbandsignals, werden die Abtastwerte parallel in den FPGA übertragen. Nachfolgend werden die drei wesentlichen Funktionen der Software eingehend vorgestellt.

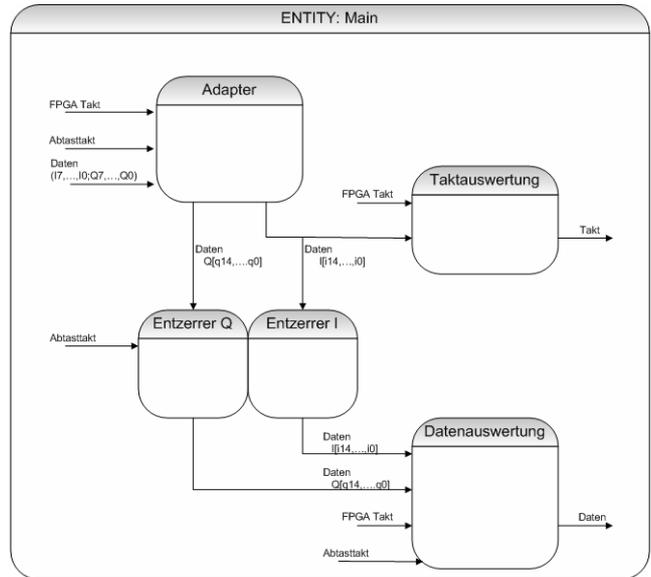


Abbildung 4: Blockschaltbild der Software

2.1. Der Entzerrer

2.1.1 Funktionsbeschreibung Entzerrer

Bedingt durch die Symbolinterferenz beeinflusst das nachfolgende Bit den Verlauf der Phase. Bei einem Übergang des Datenvektors von einer gesendeten „1“ zu einer gesendeten „0“ kehrt die Phase bereits vor ihrem eigentlichen Extrempunkt um. Dadurch entsteht ein Signal, das weniger Bandbreite benötigt. Allerdings wirkt der Verlauf der Phase verschwommen. Mit einem digitalen Filter kann man aber gezielt bestimmte Frequenzen verstärken. Die Funktion „Entzerrer“ macht die Symbolinterferenz rückgängig. Bei dem sendeseitigen Gaußfilter wurden die hohen Frequenzen unterdrückt. Auf der Empfängerseite verstärkt der Entzerrer genau diese Frequenzen. Nach dem Entzerrer sollte der Phasenverlauf des Signals dem Phasenverlauf eines MSK Modulierten Signals entsprechen.

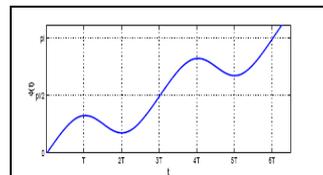


Abbildung 5: GMSK Phasenverlauf

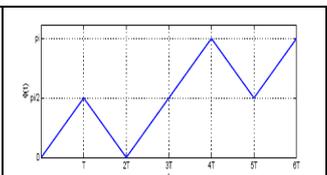


Abbildung 6: MSK Phasenverlauf

2.1.2 Die Eigenschaften des Entzerrers

Es handelt sich um ein digitales Filter 2. Ordnung. Mit der Parametrisierung werden die charakteristischen Eigenschaften festgelegt. Die Invertierung des sendeseitigen Gaußfilters führt zu einem empfängerseitigen IIR-Filter (= Infinite Impuls Response-Filter).

Die Übertragungsfunktion des Filters

$$Y(z) = \frac{0,1096 - 0,3593 \cdot z^{-1} + 1,1781 \cdot z^{-2}}{1 + 0,3050 \cdot z^{-1}} \cdot X(z)$$

entspricht folgender Differenzgleichung:

$$Y(n) = b_0 \cdot x_n + b_1 \cdot x_{n-1} + b_2 \cdot x_{n-2} - a_1 \cdot y_{n-1}$$

- mit:
- $a_1 = 0,3050$
 - $b_0 = 0,1096$
 - $b_1 = -0,3593$
 - $b_2 = 1,1781$

Diese Gleichung kann in einen FPGA programmiert werden. Die einzige Bedingung ist, dass man die Parameter für die Programmierung auf Integerwerte transformieren muss. Das führte innerhalb der Entzerrer Entity zu erheblichen Bitarraylängen.

Um die Parameterwerte mit einer Genauigkeit von vier Stellen nach dem Komma annähern zu können, wurden im ersten Schritt die Parameter mit einem Faktor von 2^{20} multipliziert. Im Dualsystem bedeutet eine derartige Transformation eine Stellenverschiebung um 20 Stellen. In der folgenden Tabelle sind die Resultate aus der Umrechnung dargestellt:

Parameter	Wert	errechnete Anpassung	eingesetzte Anpassung	Abweichung
b_0	0,1096	114923,9296	114924	6,71E-08
b_1	-0,3593	-376753,3568	-376753	3,40E-07
b_2	1,1781	1235327,386	1235327	- 3,68E-07
a_0	1	1048576	1048576	0,00E+00
a_1	0,305	319815,68	319816	3,05E-07

Es genügt jedoch nicht nur die Parameter um 20 Stellen zu verschieben. Es müssen auch die Daten, die

von der Komponente Adapter übergeben werden, mit diesem Faktor angepasst werden.

2.2. Die Taktrückgewinnung

2.2.1 Funktionsweise der Taktrückgewinnung

Der Takt wird aus der I Komponente des Basisbandsignals zurück gewonnen. Die Abtastwerte werden kontinuierlich durch ein Schieberegister geschoben. Mit einem Algorithmus werden die Registerinhalte analysiert. Dies führt zu einem spezifischen Punkt auf dem Basisbandsignal. Von diesem Punkt wird mit Hilfe eines Delays der Zeitpunkt für die steigende Taktflanke ermittelt. Die Bedingungen für diesen Triggerpunkt greifen allerdings nur, wenn das Basisbandsignal einen seiner Extrempunkte überschreitet und zwei gleiche Bits aufeinander folgend gesendet wurden. Diese Bedingungen haben sich auf das Ergebnis der Taktrückgewinnung sehr positiv ausgewirkt.

Sobald ein Triggerpunkt gefunden wurde, steuert die Funktion Taktauswertung den Takt für die Datenrückgewinnung. Die Analyse des Basisbands wird weiterhin fortgesetzt, um erneut einen Triggerpunkt zu finden. Damit wird der Takt für die Datenrückgewinnung ständig neu synchronisiert.

2.2.2 Ergebnis der Taktauswertung

Die daraus gewonnene Taktübereinstimmung ist in folgender Abbildung dargestellt:

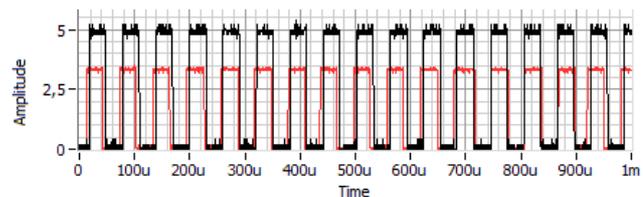


Abbildung 7: Ergebnis Taktsynchronisierung

2.3. Die Demodulation

In der Funktion „Demodulator“ werden die Daten aus dem zuvor bearbeitete Basisbandsignal zurück gewonnen.

Die Phasenänderung beim GMSK Signal beträgt $\frac{\pi}{2}$ pro Bit. Innerhalb einer Bitdauer wird das Basisbandsignal zehnfach abgetastet. Das Basisbandsignal folgt



einem sinusförmigen Verlauf, betrachtet man diesen innerhalb eines Zeitintervalls von n bis $n + \frac{\pi}{2}$, so erkennt man einen monotonen Verlauf (für $n = k \cdot \frac{\pi}{2}; k \in \mathbb{N}$). Für die Demodulation kann man sich diese Eigenschaft zu Nutze machen.

Die Abtastwerte werden durch ein zehn Speicher tiefes Register geschoben. Die steigende Taktflanke signalisiert den Beginn eines neuen Bits. Sobald in die Speicher die neuen Werte übernommen worden sind, erfolgt die Auswertung der Speicherinhalte. Es wird geprüft ob sich die Speicherwerte kontinuierlich größer oder kleiner geworden sind. Durch die Analyse der Speicherinhalte bekommt man innerhalb eines Taktes eine exakte Tendenz, wie sich das abgetastete Signal verhält.

Die Information über die Tendenz wird in 2 Variablen zu jeweils 2 Bit für die Weiterverarbeitung gespeichert. Die Tendenzauswertung kann zu drei verschiedenen Ergebnissen gekommen sein:

1. Basisbandsignal steigt monoton an
2. Basisbandsignal fällt monoton ab
3. Keine Tendenz erkennbar

Als Hintergrund zu 3.: Das GMSK Signal führt bei Änderung des Bitwerts seine vollständige Phasenänderung nicht aus; in diesem Fall beträgt die Änderung der Phase weniger als $\frac{\pi}{2}$. Durch den Entzerrer wurde dieser Effekt zwar minimiert, aber nicht vollständig rückgängig gemacht. Daher muss damit gerechnet werden, dass der Fall auftritt, bei dem keine klare Tendenz erkennbar ist.

Das Ergebnis aus der Tendenzanalyse wird mit dem gespeicherten Ergebnis aus dem vorangegangenen Takt verglichen und ausgewertet. In einer Tabelle sind alle denkbaren Kombinationen der Ergebnisse aus dem aktuellen und des vorangegangenen Taktes mit dem dazugehörigen Ausgabewert hinterlegt.

Die auftretenden Kombinationen wurden experimentell an der Hardware erarbeitet indem man bekannte Bitkombinationen demoduliert und ausgewertet hat.

Bit-Error Tests an einem FPGA Design mit schneller serieller Übertragungsstrecke

Thomas Helgeth

Hochschule Ulm, Prittwitzstraße 56, 89077 Ulm

EADS Deutschland GmbH, Wörthstraße 58, 89077 Ulm

thomashelgeth@hotmail.com

In vielen Anwendungen werden Daten seriell von einer Einsteckkarte über die "Backplane" zu einer anderen Leiterkarte übertragen. Dabei müssen Leitungen über Strecken von bis zu 50 cm und mindestens 2 Steckverbindungen geführt werden. Vor allem ist bei, sehr hohen Übertragungsraten die Abschwächung bzw. Verfälschung des Signals nicht unerheblich.

Die FPGAs der Firma XILINX[®] enthalten spezielle I/O-Ports (RocketIO[™]) mit deren Hilfe sich Übertragungsraten von bis zu 6,5 Gb/s realisieren lassen.

In einen Virtex-4[™] FPGA wurde ein Bit-Error-Test Design implementiert, das deterministisch vorhersagbare Bitfolgen sendet, wieder empfängt und mit den gesendeten Prüfbitfolgen vergleicht. Die Tests werden von einem PowerPC[™] Prozessorsystem im FPGA gesteuert und ausgewertet.

Die Bitfehlerrate, mit deren Hilfe eine Aussage über die Qualität einer Übertragungsstrecke gemacht werden kann, wird über RS232 an einen PC ausgegeben.

Mit dem Testboard können verschiedene Übertragungsmedien und OnBoard-Loops getestet werden.

1. Signalintegrität

"Unter Signalintegrität ist die Qualität des zu übertragenden digitalen Signals vom Treiber (Sender) zur Last (Empfänger) zu verstehen." [4]

Um eine "gute" Signalintegrität zu gewährleisten sollte eine Impedanzanpassung der Übertragungsstrecke

durchgeführt werden. Außerdem sollten Übertragungsverluste und Übersprechen möglichst klein gehalten werden.

1.1. Impedanzanpassung

Bei einem Rechtecksignal entstehen Oberwellen, die einem Vielfachen der Grundfrequenz entsprechen. Bei der verwendeten Taktrate von 250 MHz entstehen im Spektrum Oberwellen im Gigahertz Bereich.

Die Leiterbahn auf dem Testboard wirkt dann wie ein Dipol und die Energie wird abgestrahlt. Deshalb muss die Leitungsführung impedanzkontrolliert sein, d.h. über die ganze Leitungslänge sollte die Leitung die gleiche Impedanz haben.

Wird eine Leitung nicht impedanzkontrolliert ausgeführt, kommt es zu Reflektionen auf der Leitung. Die Welle des Signals läuft dabei vom Ausgang zum Eingang eines anderen Bausteins und wird dort reflektiert. Von dort kann die Welle zu beliebigen dritten Bausteinen reflektiert werden. Das hat zur Folge, dass der Eingang des dritten Bausteins zum falschen Zeitpunkt dieses reflektierte Signal als neues Taktsignal erkennt und den Schaltkreis entsprechend falsch taktet.

Eine impedanzkontrollierte Leiterplatte kann nur mit Wissen über die Herstellverfahren und in enger Zusammenarbeit mit dem Layouter geroutet und gefertigt werden. Programme wie "Polar" von Polar Instruments[®] unterstützen die Planung von High Speed Signalleitungen.

Bei der Impedanzanpassung der differentiellen Leitungen, wie sie für serielle High-Speed Übertragungen genutzt werden, sind folgende Punkte zu beachten:

- differentielle Leitungspaare sollten die gleiche Länge haben. Über FR4 Platinenmaterial wandert eine Welle mit $1,6 \cdot 10^8$ m/s. Daher müssen Leiterbahnen für 3 Gb/s Übertragungen bis auf 8,5 mm gleich lang sein. Ist die Übertragung noch schneller (etwa 6 Gb/s) ist eine Längenanpassung bis auf 1,7 mm zwischen N- und P-Signal nötig.
- die Leitungspaare (strip line) müssen in einer definierten geometrischen Anordnung zwischen zwei Referenz-Platinenlagen verlaufen. Eine Empfehlung von XILINX™ [1] ist in *Abbildung 1-1* dargestellt.

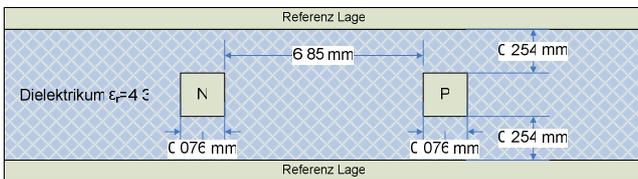


Abbildung 1-1: geometrische Lage einer "strip line" zwischen zwei Lagen der Platine

- die differentiellen Leitungspaare müssen mit einem Terminierungswiderstand von 100 Ohm abgeschlossen sein.
- Steckverbindungen schwächen das Signal ab und stören die Impedanz der Übertragungsstrecke. Spezielle High-Speed Stecker versuchen die negativen Effekte durch spezielle Signalführung, interne Schirmung und Bezugspotentiale zwischen den Pins schon im Stecker zu verringern.
- Vias die das High-Speed Signal von einer Leiterkartenebene auf eine andere führen, sollen soweit möglich, vermieden werden. An verschiedene Stellen am Via herrschen unterschiedliche Feldstärken. Diese Unterschiede haben negativen Einfluss auf die Impedanz der Übertragungsstrecke. *Abbildung 1-2* zeigt die von ALTERA® durchgeführte Simulation der Feldstärke an einem Via durch 2 Ebenen einer FR4 Leiterkarte. Ideal wäre eine gleichmäßige dunkle Schraffur.

Auch der Teil des Vias, der nicht die beiden Platinenlagen miteinander verbindet (stub), verursacht zusätzliche Reflexionen. Daher wird in einem Verfahren, das "backdrilling" genannt wird, die leitende Hülle des Vias von beiden Seiten bis zur Ziellage aufgebohrt.

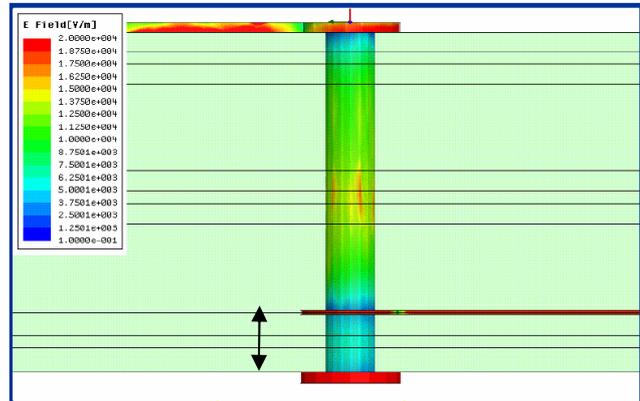


Abbildung 1-2: Simulation der Feldstärkenunterschiede an einem Via

1.2. Übersprechen

Zwei differentielle Leitungen stellen einen elektrischen Schwingkreis dar und können daher sowohl als Sender als auch als Empfänger von elektrischen Feldern dienen. Wird ein Signal übertragen, das gemeinsam neben anderen Adernpaaren geführt ist, so wird dieses Signal eingekoppelt. Die Einkopplung erfolgt induktiv, kapazitiv oder galvanisch.

Um Übersprechen zu vermeiden, sollten Tx- und Rx-Leitungen über größere Strecken möglichst nicht parallel und nahe nebeneinander verlaufen. Eine Faustregel besagt, der Abstand zwischen 2 Leitungen sollte um den Faktor 3 größer sein als ihre Breite.

1.3. Übertragungsverluste

In Realität ist das Dielektrikum kein idealer Isolator, d. h. eine gewisse Anzahl von Elektronen durchdringt es. Durch diese Vorgänge, sowie durch Umpolarisationen im Material geht ein Teil der elektrischen Energie als Verlustwärme verloren.

Skineffekt,

dieser Effekt verändert den Widerstand der innerhalb der Leiterbahn und sorgt dadurch für Verluste. Wenn ein Leiter von Signalen mit wechselnder Polarität durchflossen wird, ist die Stromdichte im Inneren der Leitung niedriger als an der Oberfläche.

Grund dafür ist, dass sich bei wechselnder Polarität des Stromflusses auch das Magnetfeld verändert. Dieses erzeugt nicht nur außen, sondern auch im Inneren des Leiters Wirbelströme. Diese sind dem Erzeugerstrom entgegengerichtet und schwächen diesen ab.

Dielektrische Verluste,

sie entstehen infolge der durch Wechselfelder bedingten Umpolarisation der Elementarteilchen des Dielektrikums. Sie werden in Schwingungen versetzt und erzeugen dabei Reibungswärme. Auch hier geht ein Teil der elektrischen Energie als Wärme verloren.

$$P_{\text{verlust}} = U_{\text{rms}}^2 * 2 * \pi * f * C * \tan \delta \quad (1-1)$$

Der $\tan \delta$ ist frequenzabhängig und ein Maß für dielektrische Verluste. Wie groß die dielektrischen und ohmschen Verluste sind hängt unter anderem vom Material aus dem die Leiterkarten gefertigt sind ab. Der Verlustfaktor $\tan \delta$ sollte möglichst klein sein. Allerdings sind Materialien mit kleinem $\tan \delta$ aufwendiger herzustellen und zu bearbeiten. Deshalb sind sie teurer als Standard Leiterkarten aus FR4 Material.

Material	Verlustfaktor $\tan \delta$ bei 1 GHz	Kostenfaktor zu FR4
FR4	0,02...0,03	1
GETEK™	0,012	1,5
Polyimide	0,007...0,01	3
Speedboard™	0,004	5
ROGERS (3003)	0,0013	6,5

Tabelle 1-1: Verlustfaktoren verschiedener Leiterkartenmaterialien

Durch die beschriebenen physikalischen Effekte werden Signale abgeschwächt bzw. verfälscht. Es kann also vorkommen, dass der Entscheider im Receiver einen zu hohen oder niedrigen Spannungspegel detektiert und als "falschen" HIGH bzw. LOW Zustand interpretiert.

Zwar können durch geeignetes Leiterkartenmaterial und bei Beachtung einiger Regeln beim Platinenlayout die Signalverluste und -verfälschungen minimiert werden. Will man eine Aussage wie viele Bits dennoch "falsch" empfangen werden und damit eine Aussage über die Qualität des gesamten Übertragungssystems machen, kann ein "Bit Error Test" durchgeführt werden.

2. Theoretische Informationen zu Bitfehlern

Im Folgenden soll geklärt werden, wie die verschiedenen Bitfehlerarten definiert und zu berechnen sind. Auch wird darauf eingegangen, was aufgrund der Gaußschen Fehlerverteilung, der theoretisch minimale Bitfehler ist.

2.1. Fehlerarten

Rauschförmige Störsignale mit Gaußscher Amplitudenverteilung bewirken zeitlich zufällig verteilte Fehler (random error). Diese entstehen bei einem ungenügenden Abstand des Nutzsignals vom Störsignal. Als Modell für diese Art von Fehlern dient der symmetrisch gestörte Binärkanal, bei dem die Verfälschung einer Eins in eine Null und umgekehrt gleichwahrscheinlich ist.

Rauschförmige Störungen führen zu einem schlagartigen Auftreten von Bitfehlern, sobald die Spitzen des Störsignals die Entscheidungsschwelle erreichen. So kann eine Verminderung des Signal-Rauschabstands um ein dB die Bitfehlerhäufigkeit um eine Zehnerpotenz vergrößern.

Impulsförmige Störsignale treten im Allgemeinen spontan auf und haben bündelförmige Bitfehler (burst errors) zur Folge. Pro Störimpuls ist mit mehreren Bitfehlern zu rechnen. Hier dient der asymmetrisch gestörte Binärkanal als Modell, bei dem die Verfälschung der Bits nicht gleichwahrscheinlich ist.

Ursache können z.B. Störimpulse sein, die aus Starkstromeinrichtungen stammen und über verschiedene Wege in das Übertragungssystem eingekoppelt werden. Auch atmosphärische Entladevorgänge kommen als Ursache in Frage.

2.2. Berechnung

Die Bitfehlerrate (bit error ratio) ist der Quotient, aus der Anzahl der in einem beliebigen Zeitintervall fehlerhaft empfangenen Bits (Bitfehler) und der Anzahl im gleichen Zeitintervall insgesamt empfangenen Bits.

$$\text{Bitfehlerrate} = \frac{\text{Bitfehler}}{\text{übertragene Bits}} \quad (2-1)$$

Beispielsweise bedeutet eine Bitfehlerhäufigkeit von $3 \cdot 10^{-6}$, dass von 1 Million übertragener durchschnittlich 3 Bits falsch sind. Die Bitfehlerrate oder auch Bitfehlerverhältnis kann aus Messergebnissen berechnet werden.

Die Bitfehlerhäufigkeit ist ebenfalls ein Maß für die Qualität der Übertragungsstrecke. Im Englischen wird die Bitfehlerhäufigkeit als "bit error rate" (BER) bezeichnet, mit der Dimension "Anzahl Fehler pro Zeiteinheit".

$$\text{Bitfehlerhäufigkeit} = \frac{\text{Bitfehler}}{\text{Zeit}} \quad (2-2)$$

Die Bitfehlerwahrscheinlichkeit (bit error probability, BEP) ist die theoretisch Angabe der Bitfehlerhäufigkeit bei unendlich langer Messzeit. Für ein gleichverteiltes, binäres Signal, einem idealen Entscheider und normalverteiltem (weißen) Rauschen kann die Bitfehlerwahrscheinlichkeit berechnet werden.

Die Wahrscheinlichkeitsdichtefunktion gibt an mit welcher Wahrscheinlichkeit ein Bit das mit einem bestimmten Spannungspegel empfangen wird als Eins $e_1(t)$ bzw. Null $e_0(t)$ bewertet wird. Beispielsweise werden logische Einsen bei bipolarer LVDS Übertragung mit + 400 mV dargestellt. Durch Rauschen das als Gauss verteilt angenommen werden kann und die Charakteristik des Übertragungskanal (Tiefpassverhalten) wird dieser Pegel verändert.

Im Idealfall entscheidet der Empfänger bei Übertragung, dass eine negative Spannung einer logischen Null entspricht, umgekehrt entspricht eine positive Spannung einer logischen Eins. Die Wahrscheinlichkeitsdichtefunktionen wird wie folgt dargestellt:

$$p_{y(0)}(x | e_0(t)) = \left(\frac{1}{\sqrt{2\pi\sigma_{n_e}^2}} \right) \frac{e^{-\frac{(x-y_0)^2}{2\sigma_{n_e}^2}}}{2\sigma_{n_e}^2} \quad (2-3)$$

$$p_{y(0)}(x | e_1(t)) = \left(\frac{1}{\sqrt{2\pi\sigma_{n_e}^2}} \right) \frac{e^{-\frac{(x-y_1)^2}{2\sigma_{n_e}^2}}}{2\sigma_{n_e}^2} \quad (2-4)$$

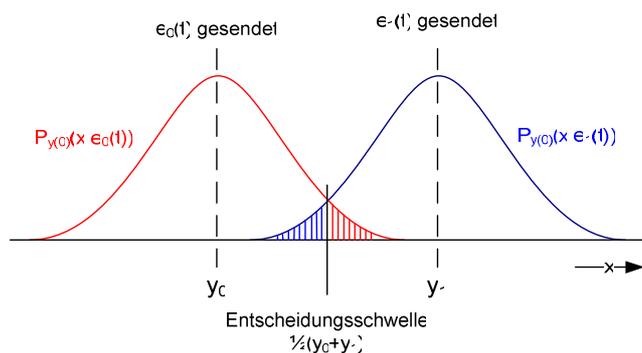


Abbildung 2-1: Wahrscheinlichkeitsdichtefunktion bipolare Übertragung

In Abbildung 2-1 sind die beiden Funktionen 2-7 und 2-8 grafisch dargestellt. Die jeweils schraffierten Flächen geben die Fehlerwahrscheinlichkeit bei entweder einer gesendeten Eins ($e_1(t)$) oder Null

($e_1(t)$) wieder. Da beide Flächen gleich sind kann die Fehlerwahrscheinlichkeit mit

$$P_{e_1} = \int_{-\infty}^c p_{y(0)}(x | e_1(t)) dx \quad \text{mit} \quad c = \frac{1}{2}(y_0 + y_1) \quad (2-5)$$

berechnet werden. Die Berechnung des Integrals (2-9) erfolgt mit dem gaußschen Fehlerintegral $\Phi(x)$ oder der komplementären Fehlerfunktion $erfc(x)$.

$$P_{e_1} = P_{y(0)}(c) = \frac{1}{2} erfc \left(\frac{y_1 - c}{\sqrt{2\sigma_{n_e}^2}} \right) \quad (2-6)$$

Lösungen zur komplementären Fehlerfunktion können in Tabellen nachgeschlagen oder durch

$$erfc(x) = \left(\frac{1}{x\sqrt{\pi}} \right)^{(-x^2)} \quad (2-7)$$

angenähert werden, der relative Fehler dabei ist kleiner 2%. In Anwendungen ist es üblich nicht wie in (2-6) c und σ^2 anzugeben, sondern das Signal-Rauschverhältnis SNR (signal noise ratio).

$$SNR = \frac{\frac{1}{2}(y_0^2 + y_1^2)}{\sigma_{n_e}^2} \quad (2-8)$$

Daraus ergibt sich für die Bitfehlerwahrscheinlichkeit P

$$P = \frac{1}{2} erfc \left(\sqrt{\frac{1}{2} SNR} \right) \quad (2-9)$$

Die Wahrscheinlichkeit, dass ein Bit verfälscht wird, ist umso kleiner je besser das Signal-Rausch-Verhältnis ist.

2.3. Einordnung von Bitfehlerraten

Um die Angabe einer Bitfehlerrate einordnen und bewerten zu können werden hier die spezifizierten, bzw. typischen Bitfehlerraten einiger anderer Übertragungsstandards angegeben.

Art	Bandbreite	Bitfehlerrate [1/Bit]
Video on Demand (VoD)	0,5 bis 15 MB/s	10^{-5}
Internettelefonie (Voice over IP VoIP)	16 kBit/s	10^{-5}

Art	Bandbreite	Bitfehlerrate [1/Bit]
Internetverbindung über http	128 kBit/s	10^{-8}
File Transfer Protocol (ftp)	128 kBit/s	10^{-8}
PCI Express	2,5 Gb/s	10^{-12}

Tabelle 2-1: Vergleich von Bitfehlerraten verschiedener Übertragungsstandards

3. Augendiagramme zur Beurteilung der Signalintegrität

Ein Augendiagramm ist ein Hilfsmittel um die Signalqualität einer digitalen Datenübertragung zu beurteilen. Dazu wird der Datenstrom in gleich lange Abschnitte unterteilt. Diese werden mit Hilfe eines Hochgeschwindigkeits- Speicher- Oszilloskops übereinander geschrieben. Auf dem Schirm entsteht ein Bild mit der Form eines Auges.

Um die hohen Datenraten abtasten zu können wurde ein Oszilloskop von AGILENT® mit der Bezeichnung "infiniium 54855 DSO" verwendet. Das Oszilloskop hat eine Bandbreite von 6 GHz und eine Auflösung von 20Gs/s.

Die vertikale Augenöffnung zeigt, ob eine Übertragung überhaupt möglich ist und wie empfindlich die Übertragung gegenüber Störungen ist. Ist das Auge geschlossen, so ist keine Übertragung möglich, da HIGH und LOW Zustände nicht mehr unterschieden werden können.

Charakteristische Größen in einem Augendiagramm sind die horizontale und vertikale Augenöffnung. Beide sind im Idealfall so groß wie möglich.

Als "Interface Jitter" wird die Phasenmodulation der Nulldurchgänge in einem digitalen Datenstrom bezeichnet. Die Nulldurchgänge des Signals sollten idealerweise immer zur gleichen Zeit stattfinden. Die Zeitspanne vom frühesten bis zum spätesten Nulldurchgang mehrerer Messungen wird als Jitter bezeichnet, dieser sollte möglichst klein sein.

Die Folgenden Abbildungen zeigen Augendiagramme des RocketIO™ Loops MGT 105A auf Lage 4 des Testboards bei verschiedenen Geschwindigkeiten Das Signal wurde mit einem differentiellen Tastkopf direkt an den Vias des FPGAs gemessen. Sie unterscheiden sich durch die Übertragungsrate 2,5 Gb/s bzw. 5 Gb/s und die verwendete Übertragungsstrecke

Übertragung des "2⁷-1 Testmusters" mit 2,5 Gb/s an MGT 105 A (OnBoard Loop)

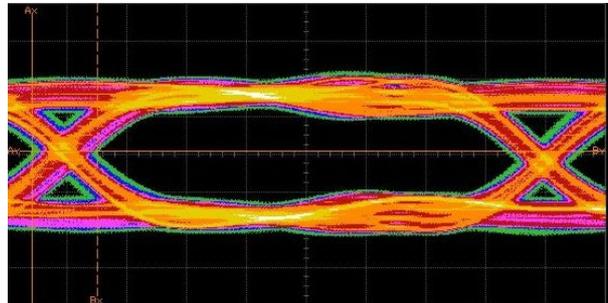


Abbildung 3-1: TX Signal an MGT 105A

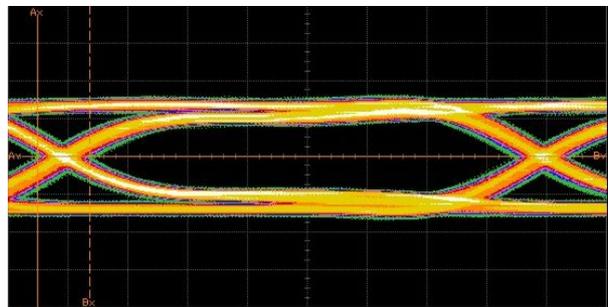


Abbildung 3-2: RX Signal an MGT 105A

Durch Leitungsverluste ist die Augenhöhe am Empfänger kleiner als am Sender

Augenhöhe	RX: 663 mV	TX: 1033 mV
Augenbreite	RX: 359 ps	TX: 364 ps
Jitter (ps)	RX: 43,7	TX: 55,7 ps
Horizontal: 50 ps/Div	Vertikal: 500 mv/Div.	Auflösung: 20GS/s

Übertragung des "2⁷-1 Testmusters" mit 5 Gb/s an MGT 105A (OnBoard Loop)

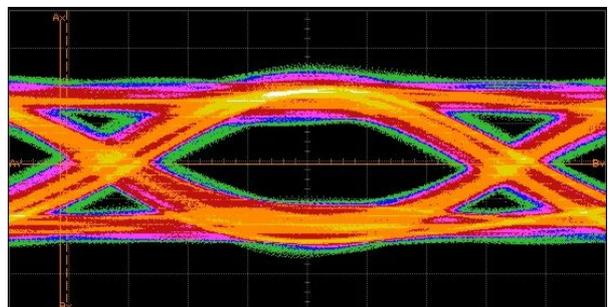


Abbildung 3-3 TX Signal an MGT 105A

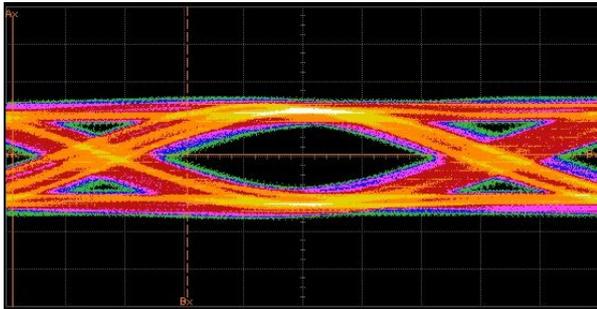


Abbildung 3-4: RX Signal an MGT 105A

Sichtbar größerer Jitter und abgeschwächtes Signal. im Vergleich zum TX-Signal

Augenhöhe	RX: 748 mV	TX: 938 mV
Augenbreite	RX: 109 ps	TX: 103 ps
Jitter (ps)	RX: 87,5 ps	TX: 47,7 ps
Horizontal: 30 ps/Div	Vertikal: 500 mv/Div.	Auflösung: 20GS/s

Die oben dargestellten Augendiagramme zeigen Signale mit ausreichend großen Augenöffnungen und relativ wenig Jitter. Mit diesen Signalen sind Bitfehlerraten von besser 10^{-12} zu erreichen.

Übertragung des "2⁹-1 Testmusters" mit 2,5 Gb/s an MGT 110A (ungeschirmte Kupferleitung)

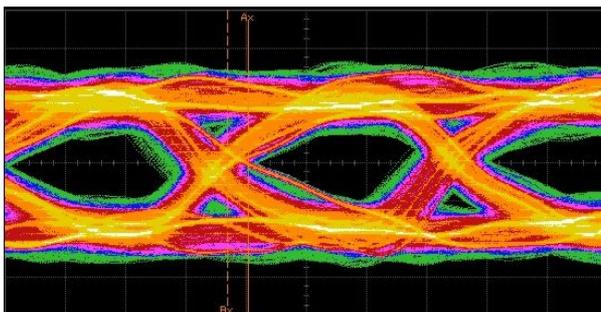


Abbildung 3-5 TX Signal an MGT 105A

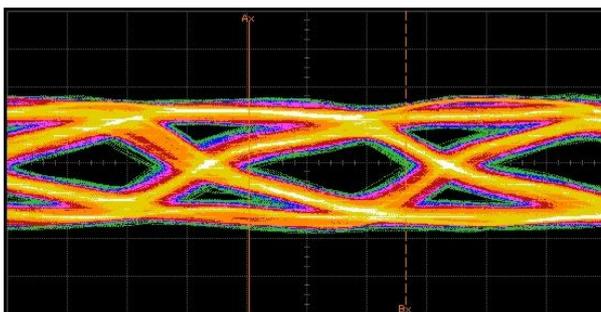


Abbildung 3-6: RX Signal an MGT 105A

Zum Vergleich die Augendiagramme *Abbildung 3-5* und *Abbildung 3-6*. Als Übertragungsstrecke wurde eine 15 cm lange, verdrehte, ungeschirmte Kupferleitung gewählt. Die hier erreichte Bitfehlerrate liegt bei 10^{-6} .

Augenhöhe	RX: 318 mV	TX: 637 mV
Augenbreite	RX: 260 ps	TX: 265 ps
Jitter (ps)	RX: 25,5 ps	TX: 22,7 ps
Horizontal: 100 ps/Div	Vertikal: 500 mv/Div.	Auflösung: 20GS/s

4. Architektur des Bit Error Test (XBERT)

Bei einem Bit-Fehler-Test (Bit-Error-Test, BERT) werden deterministisch vorhersagbare Bitfolgen auf einer Übertragungsstrecke gesendet, wieder empfangen und mit den gesendeten Prüfbitfolgen verglichen.

4.1. Überblick

Die Firma XILINX® stellt unter dem Namen RocketIO™ XBERT ein Bit-Error-Test-Referenzdesign zur Verfügung, dieses soll an eine bestehende, von der EADS entwickelte Hardware angepasst werden.

Ein RocketIO™ ist ein Multi-Gigabit-Transceiver, dieser MGT ist eine spezielle IO Zelle im FPGA die es ermöglicht differentielle Signale bis 6 Gb/s zu senden und empfangen.

Das XBERT Referenzdesign bietet folgende Möglichkeiten:

- auf bis zu 12 RocketIO™Channels können Tests durchgeführt werden
- auf jedem Channel können verschiedene, auch selbst definierte, Testmuster uncodiert oder in 8B/10B Codierung übertragen werden
- Bit Error Tests können synchron oder asynchron zwischen zwei MGT des gleichen FPGA oder zwischen 2 verschiedenen FPGA durchgeführt werden
- Über das "Dynamic Reconfiguration Interface" können die MGT Eigenschaften auch während des Tests verändert werden
- über eine RS232 UART Schnittstelle können Testparameter verändert und Testergebnisse ausgegeben werden

Das RocketIO™ XBERT Design besteht aus verschiedenen Modulen, wie sie in *Abbildung 4-1* dargestellt sind.

Das Multi-Channel XBERT Modul generiert für jeden Channel einstellbar eines der folgenden Datenmuster,

- eine Pseudo-Zufalls-Bitfolge (Pseudo Random Bit Sequence, PRBS), oder
- einen Takt in verschiedenen Verhältnissen zum REFCLK, oder
- ein vom Benutzer definiertes Testmuster

Ebenfalls im Multi-Channel XBERT Modul werden die Testmuster wieder empfangen und auf Übertragungsfehler überprüft.

Genauer besteht ein XBERT Multi-Channel aus 2 MGT (MGTA und MGTB) die physikalisch in einer Zelle des Virtex-4™ zusammengefasst sind. MGTA und MGTB haben jeweils einen eigenen Daten-Verifikations-Block, verwenden aber denselben Testmuster Generator. In jedem XBERT Channel können somit unabhängige Testmuster übertragen werden.

Der in dieser Arbeit verwendete Virtex-4™ beinhaltet 8 MGT somit sind 4 Channels realisierbar. Die Übertragungsrate der MGT ist bis zu 8 Gb/s einstellbar und hängt von der Taktrate des REFCLK (250 MHz) und den Teilern im MGT ab.

Ein weiteres Modul ist der embedded IBM® PowerPC™405 Prozessor (PCC405), der über die DCRtoOPB Bridge an den On-chip Peripheral Bus (OPB) angebunden ist. Dieser Prozessorkern ist ein Hardmacro und somit fest im FPGA eingebunden.

Der On-chip Peripheral Bus ist ein in PowerPC™ Prozessorsystemen verwendeter synchroner Multimaster Bus mit 32 Bit Daten- und 32 Bit Adressbits.

Der Prozessor kann über je einen Memory Controller (DS-BRAM-CNTLR und IS-BRAM-CNTLR) auf 2 getrennte Block-RAM (BRAM) Speicher zugreifen. Es steht ein 32 kB Datenspeicher (DS-BRAM) und ein 64 kB Befehlsspeicher (IS-BRAM) zur Verfügung.

Ebenfalls an den OPB ist ein UART Lite Core angebunden. Über ihn ist es möglich, interaktiv, während eines Bit-Error-Tests Parameter vom PC aus zu verändern. Beispielsweise können die Equalizer in Sender und Empfänger angepasst oder die Teiler des REFCLK verändert werden. Außerdem können die Ergebnisse der Bit-Error-Tests an den angeschlossenen PC übertragen werden.

Über eine weitere Bridge ist auch das Multi Channel XBERT Modul an den OPB angeschlossen. Die Bridge übersetzt vom OPB in das General Purpose Input/Output Format (GPIO) über das das Multi Channel XBERT Modul abgesprochen wird.

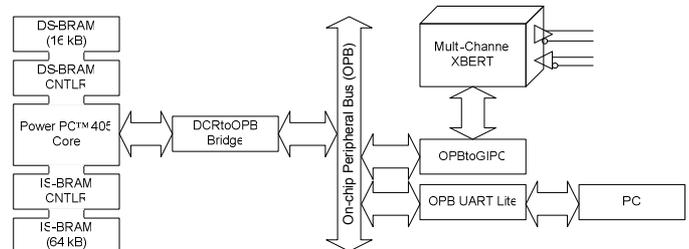


Abbildung 4-1: Systemarchitektur des XBERT FPGA Designs

4.2. Pseudo-Zufalls-Bitfolgen

Als Testmuster, die vom Multi Channel XBERT Modul erzeugt werden, eignen sich am Besten Pseudo-Zufalls-Bitfolgen. Da diese Sender und Empfänger gleichmäßig belasten ohne sich dabei zu oft zu wiederholen.

Die Pseudo-Zufalls-Bitfolgen werden im Multi Channel XBERT Modul von einem linear rückgekoppelten Fibonacci Schieberegister erzeugt. (Linear Feedback Shift Register, LFSR)

Mathematisch lassen sich LFSR mit Polynomen darstellen, ein LFSR, das zwei Rückführungen an den Stellen 6 und 8 besitzt lässt sich daher mit $1+x^6+x^8$ beschreiben.

Die ITU (International Telecommunication Union) ist eine Unterorganisation der UNO und beschäftigt sich mit den technischen Aspekten der Telekommunikation. Diese Organisation hat in ihrer Empfehlung O.150 spezifiziert, welche Pseudo-Zufalls-Bitfolgen bei Bit-Error-Test verwendet werden sollten. Auch XILINX® verweist in Dokumentationen ihrer Produkte auf die O.150 Spezifikation [2].

Folgende Testmuster können vom Multi Channel XBERT Modul erzeugt werden:

- 12 verschiedene von der ITU spezifizierte Pseudo Zufalls Bitfolgen
- Taktsignal mit der 1/2, 1/8, 1/10, 1/16 oder 1/20 Frequenz des Referenztaktes
- linear ansteigende Werte (Counter) bis zu einer Wortbreite von 40 Bit
- 8 vom Benutzer definierte sich wiederholende 16 bzw. 20 Bit Worte

4.3. Implementierung des Bit Error Test Designs

Um das XBERT Design auf der Testleiterkarte in Betrieb nehmen zu können muss der so genannte Design Flow durchlaufen werden. In *Abbildung 4-2* ist dieser dargestellt. Zuvor müssen in einigen Projektdateien des Xilinx™ Platform Studios (XPS) Anpassungen an die Testhardware der EADS vorgenommen werden, diese werden im Folgenden erläutert.

User Constraints File

Im UCF (User Constraint File) werden Angaben darüber gemacht, wie das Design in die Hardware eingebunden werden soll. Beispielsweise werden interne FPGA Signale einzelnen I/O-Pins zugeordnet und die Platzierung der MGTs im FPGA wird festgelegt. Auch können die Anstiegszeit (slew rate) einzelner Ausgänge und einige andere Parameter konfiguriert werden. Zu beachten ist dass Constraints, die im UCF angegeben werden, eventuell gleiche Attribute des VHDL Codes überschreiben.

C-Sourcecode

In den C-Dateien befindet sich der Quellcode der Software die vom PowerPC™ Prozessor ausgeführt werden soll. Die Software kann in C/C++ Entwicklungsumgebungen wie beispielsweise Eclipse SDK oder Borland® C++ Builder erstellt werden. Die Software lässt sich in 5 Bereiche unterteilen:

- XBERT_ENTRY (xbert.h): Hier werden allgemeine Konfigurationsdaten gespeichert, beispielsweise die Anzahl der Kanäle, der Prozessortyp, die Versionsnummer usw. Dieser Teil der Software wird nur ein Mal instanziiert.
- CHN_ENTRY (xbert.h und xbert.c): Hier werden Informationen abhängig vom jeweiligen MGT Channel gespeichert, wie zum Beispiel der Fehlerzähler, Statusbits, Status des Loopback. PMA und Takt Attribute werden ebenfalls für jeden Channel einzeln gespeichert. Die entsprechende Address Map befindet sich in der Virtex-4™ RocketIO™ Bit-Error Tester Application Note [3].
- GIPO_DRIVER (gpio.c und gpio.h): Der Treiber besteht aus 3 Ebenen. Ebene 0 stellt Funktionen zur Verfügung um von Prozessor-Kontroll-Registern (Device Control Register, DCR) auf den 32 Bit GPIO (General Purpose In-Output) Bus schreiben und lesen zu können. Ebene 1 stellt die Verbindung zwischen dem GPIO Bus und den Datenstrukturen von XBERT_ENTRY und CHN_ENTRY her. Ebene 3 stellt Funktionen zur

Verfügung, um an einem Channel alle Parameter konfigurieren und auch überwachen zu können.

- UART_DRIVER (uart.h und uart.c): Auch dieser Treiber unterteilt sich in 2 Ebenen. Ebene 0 initialisiert den Treiber und stellt Routinen zum Lesen und Schreiben der Prozessor-Kontroll-Registern (DCR) zur Verfügung. Ebene 1 übernimmt die Formatierung der Ein- und Ausgabe und beinhaltet Routinen, um die ANSI/VT100 Standard Terminal Funktionen zu unterstützen.
- MAIN_APP: Hier sind das Hauptmenü der XBERT Software. Sowie die Funktionen für die Interaktion mit dem Benutzer, der über ein Terminal Programm die XBERT Software steuern kann, hinterlegt. Eine Anleitung für den Gebrauch der Software befindet sich in der Xilinx™ XBERT Application Note [3].

Das Xilinx™ Platform Studio nutzt den "GNU Compiler Collection" (gcc) um Maschinencode zu generieren. Ein Linker-Skript enthält die Informationen in welcher Weise die einzelnen Quelldateien verknüpft werden müssen. Die bmm-Datei, die dabei erzeugt wird, wird bei der Generierung der bit-Datei eingebunden.

Xilinx Microprocessor Project (xmp-Datei)

In dieser Datei sind die Projekt Einstellungen des Xilinx Platform Studios gespeichert, also die Pfade zu den Dateien die Software und Hardware beschreiben. Außerdem wird in der xmp-Datei angegeben welche Tools zur Synthese und "Place and Route" verwendet werden sollen. Auch werden hier Angaben über den verwendeten Prozessor (PPC405) und FPGA (V4FX20) gespeichert.

Microprocessor Hardware Specification (mhs-Datei)

In dieser Datei wird die Hardware des verwendeten Microprocessor Systems beschrieben. Größe des Speichers, Busbreite, Übertragungsrate usw. der Komponenten wie Speicher, Bus, Interfaces und Bridges werden hier definiert.

Während der Bit-Datei-Generierung stellt sich heraus, dass der kompilierte Quellcode (bmm-Datei) größer ist als Datenspeicher im FPGA zu Verfügung gestellt wurde. Im Hardware Specifaicaion File kann der zu implementierende Speicher durch Angabe des Adressraums definiert werden. Der Speicher wurde daher von 16k auf 32k vergrößert.

Microprocessor Software Specification (mss-Datei)

Da die verwendeten Komponenten (Speicher, Busse, Interfaces) in verschiedenen Hardwareversionen vorliegen, wird in der mss-Datei festgelegt in welcher

Version die Komponenten implementiert werden sollen.

Die folgende Grafik zeigt die eingesetzten Programme und die dabei erzeugten Dateien.

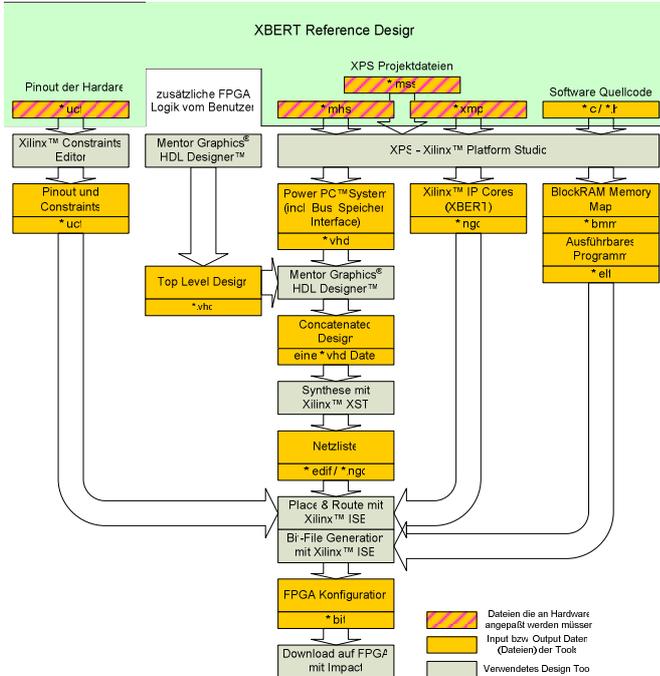


Abbildung 4-2: Designflow des XBERT Designs

Die folgende Auflistung zeigt die verwendeten FPGA Ressourcen und damit den Umfang des erstellten FPGA Designs.

Logic Utilization:

No. of Slice Flip Flops:	9,606 out of 17,088	56%
No. of 4 input LUTs:	11,064 out of 17,088	64%
No. of occupied Slices:	8,292 out of 8,544	97%

LUTs:

Number 4 input LUTs:	12,071 out of 17,088	70%
Number 4 input LUTs used as logic:	11,064	
Number used for Dual Port RAMs:	672	
Number used as Shift registers:	332	

Number of bonded IPADs:	24 out of 24	100%
Number of bonded OPADs:	16 out of 16	100%
Number of bonded IOBs:	74 out of 320	23%
Number of BUFG/BUFGCTRLs:	13 out of 32	40%
Number of FIFO16/RAMB16s:	40 out of 68	58%
Number of DSP48s:	16 out of 32	50%

5. Durchführung der Bit Error Tests

5.1. Testaufbau

Das Testboard besteht aus 16-lagigem FR4 Platinenmaterial. Die RocketIO™ Schnittstellen des Virtex-4™ sind zu verschiedenen Steckverbindungen bzw. OnBoard-Loops geroutet.

Vier unabhängige EPSON® Oszillatoren versorgen jeden der 4 Channels mit einem 250 MHz Referenztakt. Ein Channel enthält zwei Multi Gigabit Transceiver (MGT); somit sind 8 verschiedene Loops auf dem Testboard möglich.

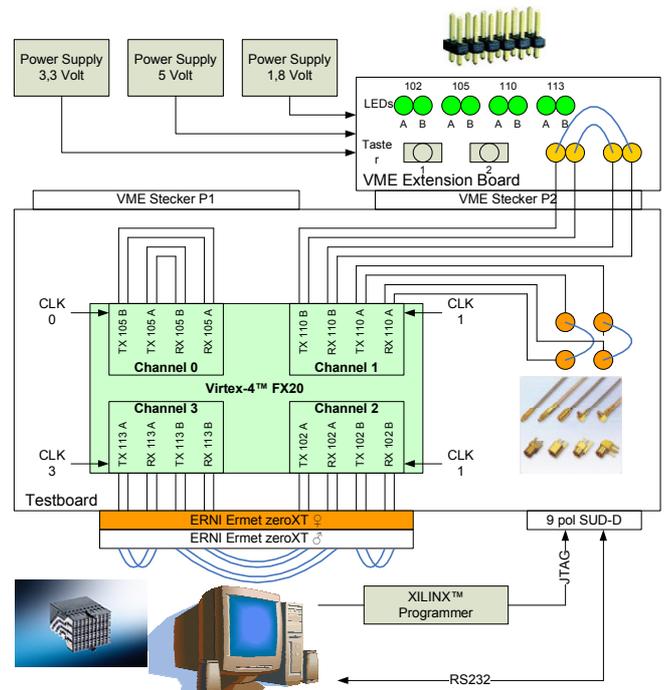


Abbildung 5-1: Schematische Darstellung des Testboards und des XBERT Testaufbaus

Die LEDs auf dem VME Extension Board leuchten wenn ein Testpattern über den jeweiligen MGT gesendet wird. Der Taster 1 löst ein Reset des PowerPC™ aus, während Taster 2 ohne Funktion bleibt.

Die verschieden geführten Loops auf dem Testboard beeinflussen die Signalintegrität, dadurch können Aussagen gemacht werden, wie beispielsweise Vias, die das Signal auf verschiedene Leiterplattebenen führen, die Bitfehlerrate beeinflussen. Auch der Einfluss der verwendeten Steckverbindungen und Leitungslängen auf die Übertragungsqualität kann so ermittelt werden.

Die folgenden Abbildungen zeigen jeweils einen Schnitt durch das Testboard und verdeutlichen wie

das differenzielle Signal über verschieden Leiterplattebenen und Steckverbindungen geführt wird.

Channel 0

- MGT 105A: Micro strip Line im Loop auf Lage 4 des Testboards
- MGT 105B: Micro strip Line im Loop auf Lage 13 des Testboards

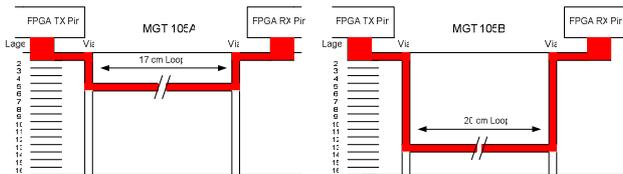


Abbildung 5-2: Schnitt durch die Leiterbahn des Testboards, Übertragungsstrecke der MGT 105A und 105B

Channel 1

- MGT 110A: das Signal wird auf Lage 9 zum VME Stecker geführt von dort auf das Extension Board zum Pfostenstecker.

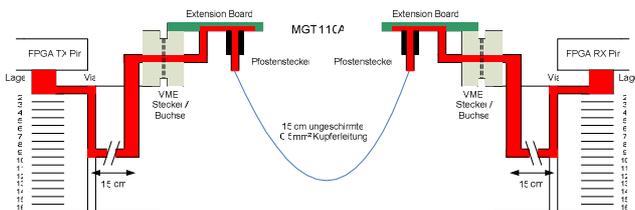


Abbildung 5-3: Schnitt durch die Übertragungsstrecke des MGT 110A

- MGT 110B: Micro strip Line zu den MMCX-Buchsen, die Buchsen sind mit 30 cm EF176 Koaxialkabel verbunden.

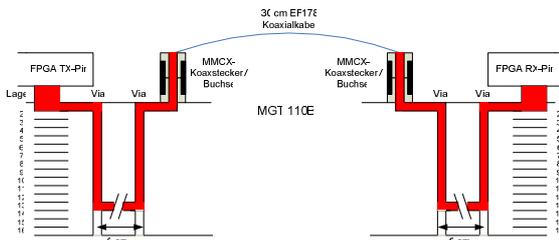


Abbildung 5-4: Schnitt durch die Übertragungsstrecke des MGT 110B

6. Ergebnisse der Bit-Error-Tests

Ziel der Messungen war es, eine Aussage über die Bitfehlerrate der auf dem Testboard eingesetzten MGTs auf verschiedenen Übertragungsstrecken bei verschiedenen Geschwindigkeiten machen zu können.

6.1. Testbedingungen

Die Art des verwendeten Testpatterns hatte keine nennenswerte Auswirkung auf die Bitfehlerrate. Es wurde daher für alle hier dokumentierten Versuche das Testmuster mit dem Polynom "27-1" verwendet.

Alle Tests wurden über einen Zeitraum von länger als 15 Stunden durchgeführt, so dass ein ausreichend großer Vertrauensbereich gegeben ist und die Angaben statistisch sinnvoll sind.

Für alle Tests wurde die Software Version 1.2 des XBERT Designs verwendet.

Für jeden durchgeführten Test wurde ein Testprotokoll erstellt.

7. Ergebnisse der Bit Error Tests

7.1. Verschiedene Spannungsversorgungen

Ziel der Messungen war es eine Aussage über die Bitfehlerrate der auf dem Testboard eingesetzten MGTs auf verschiedenen Übertragungsstrecken bei verschiedenen Geschwindigkeiten machen zu können.

Untersucht werden sollte, wie die Art der Spannungsversorgung, entweder über Längs- oder Schaltregler, die Bitfehlerrate beeinflusst.

In der ersten Konfiguration wurde sowohl der MGT als auch das Core des FPGAs aus demselben Texas Instruments® TPS54610 Schaltregler gespeist.

2,5 Gb/s			
Channel 0 (MGT105)		Channel 1 (MGT 110)	
onboard Oberfläche	onboard Lage 9	Koaxkabel	ungesch. Kupferdraht
BER: $1,25 \cdot 10^{-12}$	BER: $1,25 \cdot 10^{-12}$	BER: $4,17 \cdot 10^{-11}$	BER: $2,23 \cdot 10^{-6}$

Tabelle 7-1: 1,2 Volt aus FPGA-Core Schaltregler, 2,5Gb/s Übertragung

In einem weiteren Test wurden die MGTs von einem separaten Schaltregler von Texas Instruments® TPS54610 versorgt.

2,5 Gb/s			
Channel 0 (MGT105)		Channel 1 (MGT 110)	
onboard Oberfläche	onboard Lage 9	Koaxkabel	ungesch. Kupferdraht
BER: $7,92 \cdot 10^{-15}$			
Errors: 0	Errors: 0	Errors: 0	Errors: 0
5 Gb/s			
BER: $3,92 \cdot 10^{-15}$	BER: $3,92 \cdot 10^{-15}$	BER: $3,92 \cdot 10^{-15}$	keine Verbindung möglich
Errors: 0	Errors: 0	Errors: 0	

Tabelle 7-2: Spannung aus Schaltregler, 5 Gb/s Übertragung

In dieser Konfiguration wird die Spannungsversorgung der MGTs durch den Längsregler LT1963 von LINEAR[®] realisiert. Dies ist auch der Vorschlag den XILINX[®] im "RocketIO™ User Guide" [12] macht.

2,5 Gb/s			
Channel 0 (MGT105)		Channel 1 (MGT 110)	
onboard Oberfläche	onboard Lage 9	Koaxkabel	ungesch. Kupferdraht
BER: $8,27 \cdot 10^{-15}$			
Errors: 0	Errors: 0	Errors: 0	Errors: 0
5 Gb/s			
BER: $3,62 \cdot 10^{-15}$	BER: $3,62 \cdot 10^{-15}$	BER: $3,62 \cdot 10^{-15}$	keine Verbindung möglich
Errors: 0	Errors: 0	Errors: 0	

Tabelle 7-3: Spannung aus separatem Längsregler, 5 Gb/s Übertragung

Die Bitfehlerrate in dieser Konfiguration ist gering. Allerdings ist der maximale Ausgangsstrom des auf dem Testboard verwendeten Längsreglers (LT1963) zu gering um alle 4 Channels zu betreiben.

Die für die spätere Anwendung wahrscheinlichste Konfiguration ist eine Serienschaltung von Schalt- und Linearregler. Auch in dieser Konfiguration ist die Bitfehlerrate gering.

2,5 Gb/s			
Channel 0 (MGT105)		Channel 1 (MGT 110)	
onboard	onboard	Koaxkabel	ungesch.

Oberfläche	Lage 9		Kupferdraht
BER: $5,62 \cdot 10^{-15}$			
Errors: 0	Errors: 0	Errors: 0	Errors: 0

Tabelle 7-4: Serienschaltung von Schalt- und Linearregler, 2,5Gb/s

Die in Konfiguration 1.) im Vergleich zu allen andern Tests relativ schlechte Bitfehlerrate, wurde auf die 450 kHz Störung in der Spannungsversorgung zurückgeführt. Durch große zusätzliche Abblockkondensatoren ($680\mu\text{F}$) am Eingang des MGT konnten die Störeffekte abgeschwächt werden.

Die Erwartung, dass sich die Schaltfrequenz des Reglers in Konfiguration 2.), in der Versorgungsspannung als Störung wiederfindet wurde nicht bestätigt. Zwar empfiehlt XILINX[®] die Verwendung rauscharmer Längsregler [2], allerdings wurde auch mit dem Schaltregler "TPS54610" von Texas Instruments[™] ein gleich gutes Ergebnis erreicht.

7.2. Verschiedene Übertragungsstrecken

Ein zweiter Aspekt der untersucht werden sollte ist die Abhängigkeit der Bitfehlerrate von der Übertragungsstrecke.

Hier sind nur kleine Unterschiede in den Augendiagrammen erkennbar. Vias mit kurzen Stichleitungen (stubs) erzeugen weniger Reflexionen und sorgen damit für eine bessere Signalintegrität. Der Loop von mit Vias von der Oberfläche bis auf Lage 13 weist daher weniger Jitter auf. Auf die Bitfehlerrate hatte ein "routing" über verschiedenen Platinenlagen keine Auswirkung.

Auch eine Verbindung über die MMCX-Buchsen und zwei Koaxleitungen erzeugt bis zu den getesteten 5 Gb/s keine Bitfehler.

7.3. Ausblick

Je nachdem wie die RocketIO™ Verbindungen im späteren System eingesetzt werden, sollte die Auswirkung längerer Koaxleitungen auf die Bitfehlerrate überprüft werden.

Ein im Hinblick auf den späteren Einsatz der RocketIO™ Verbindungen wichtiger Test ist der Betrieb unter verschiedenen externen Einflüssen.

Die Bestimmung der Bitfehlerrate sollte in einem Temperaturschrank sowohl bei hoher, wie auch niedriger Temperatur durchgeführt werden. Systeme,

die in der Luftfahrt eingesetzt werden, müssen in einem breiten Temperaturbereich funktionieren.

Auch elektromagnetische Felder, die auf die Übertragungsstecke einwirken, sind Störfaktoren. Der Einfluss dieser Störquellen sollte ebenfalls in Tests ermittelt werden.

8. Quellverzeichnis

[1] XILINX®, Virtex-4 RocketIO Multi-Gigabit Transceiver, User Guide, UG076, v3.2, 29 September 2006

[2] XILINX® XAPP713 (v1.0), Virtex-4™ RocketIO™ Bit-Error-Tester, Vinod Kumar Venkatavaraden, 22 Juni 2006

[3] XILINX® UG242 (v1.0), BER Tester User Guide, 22 Juni 2006

[4] Gommlich, Schuon, Schiele&Schön Verlag Berlin 2005, "Datenübertragung und Messtechnik"

[5] Thomas Helgeth, Diplomarbeit, "Entwicklung und Inbetriebnahme eines High Speed Serial Interfaces für XILINX® Virtex-4™ FPGA", Januar 2007

„VHDL Synthese & FPGA Emulation eines Microcontroller- systems für eine elektronische Pille“

Tim Benneckenstein, Nidal Fawaz, Dirk Jansen

Hochschule Offenburg, Badstr. 24, 77652 Offenburg

0781/205-128

Das hier vorliegende Schriftstück entstand während meiner Diplomarbeit im ASIC Design Center der Hochschule Offenburg im Zeitraum von November 2006 bis März 2007. Die Aufgabenstellung der Arbeit, welche unter dem Titel „VHDL Synthese und FPGA Emulation eines Microcontrollersystems für eine elektronische Pille“ geführt wurde, war dabei sehr vielseitig und untergliederte sich in die typischen Problemstellungen eines Hard- & Software Co-Design Prozesses.

1. Einleitung

Träger und Auftraggeber dieses Werkes war das Institut für angewandte Forschung, kurz IAF, der Hochschule Offenburg. Die im Jahre 1986 gegründete Organisation wird seit 1995 von Herrn Prof. Jansen geleitet und ist eine hochschulinterne Institution, welche mehrere Kooperationen und Forschungsbeziehungen mit der Industrie unterhält, als auch zahlreiche kreative Projekte aus internen Quellen vorantreibt. Die einzelnen Tätigkeits- und Forschungsgebiete gründen sich dabei auf Professoren, Mitarbeiter und Labore aller fünf Fachbereiche der Hochschule.

Eine der aktivsten Komponenten des IAF stellt das ASIC Design Center der Hochschule Offenburg dar, in welchem bereits in der Vergangenheit zahlreiche innovative Projekte, wie etwa die Entwicklung eines EKG Recorders oder das Design eines ASICs zur Verschleißmessung an Arbeitsmaschinen entstanden.

Ein aktuelles Projekt des ASIC Design Centers ist das Projekt „ePille“, welches gleichzeitig Bestandteil der Doktorarbeit meines Zweitbetreuers Herrn Fawaz ist. Als interdisziplinäres Co-Projekt aus dem Bereich der Medizintechnik, besteht seine Zielsetzung darin, einen Chip zur gezielten Medikamentenabgabe und Erfassung von Messwerten im menschlichen Körper zu entwickeln. Eine solche elektronisch gesteuerte Pille ist vor allem für jene Anwendungsszenarien interessant, in welchen die Abgabe des Wirkstoffes genau zu einem bestimmten Zeitpunkt erfolgen muss, etwa der Verabreichung von Insulin bei Diabetikern.

Hierzu verfügt die ePille über eine Kommunikationsverbindung im Zusammenspiel mit einer externen Kontrolleinheit, über welche das integrierte Microcontrollersystem gesteuert werden kann. Ein, hinsichtlich der realen Pillendimension etwas überzeichnetes Kommunikationsmodell, ist in Abb. 1 zu sehen.

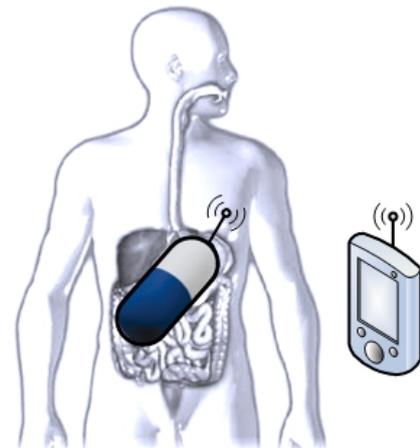


Abbildung 1: Kommunikationsmodell ePille

Im Großen und Ganzen verbirgt sich hinter der eventuell etwas futuristisch anmutenden Bezeichnung ePille also nicht viel mehr als ein miniaturisiertes Microcontrollersystem mit integrierter Sensor-Aktor-Phalanx sowie drahtloser Kommunikationsschnittstelle.

Da ein solch komplexes Projekt unmöglich von einer Person allein realisiert werden kann, wurden einige Teilbereiche dieses Unterfangens aus diesem Grunde als eigenständige Diplomarbeiten ausgeschrieben. Das Design des Mikrocontrollersystems war dabei Bestandteil dieser Arbeit, während die Entwicklung der Kontrolleinheit in der Abschlussarbeit meines Teamkollegen Herrn Stefan Müller realisiert wurde.

2. Aufgabenbeschreibung

Die sehr vielseitige Aufgabenstellung des Projektes gliederte sich in die klassischen Arbeitsschritte eines Hard & SW Co-Design Prozesses, genauer gesagt:

- der Synthese des Microcontrollersystems aus größtenteils vorhandenen Komponenten, welche zwecks ihrer Verwendung in der ePille angepasst und geringfügig modifiziert werden mussten.
- der FPGA Emulation der digitalen Schaltung zwecks Test und Verifikation in Hardware.
- der Entwicklung einer Evaluationsplatine zur Kontrolle der Hardwarefunktionalität sowie Durchführung von Tests.
- der Programmierung von Hardwaretreibern für die einzelnen Peripheriekomponenten des Systems.
- der Implementierung einer Testbench zwecks Endabnahme und Analyse des reibungslosen Zusammenspiels von Hard- und SW-Komponenten.

Sinn und Zweck dieser Diplomarbeit besteht somit im eigentlichen Sinne nicht in der Realisierung des fertigen Produktes, sondern vielmehr in der Schaffung einer soliden Basis für die spätere ASIC Synthese. Durch die diversen Einzelschritte sollen dabei die prinzipielle Funktionalität des Systems sichergestellt, und gleichzeitig via Test und Verifikationen möglichst alle Fehlerquellen ausgeräumt werden.

3. Design Flow

Die Abfolge der Teilarbeiten erfolgt dabei gemäß des in Abb. 2 gezeigten Flussdiagramms.

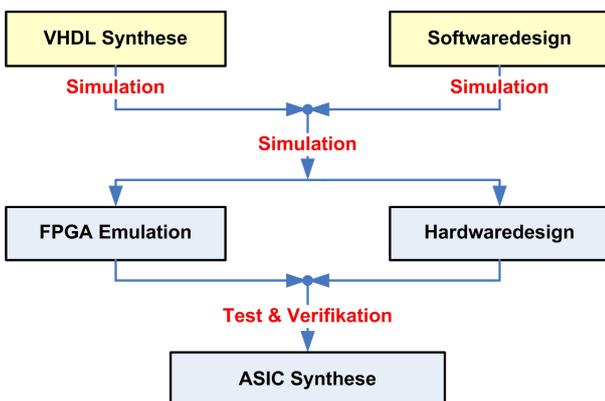


Abbildung 2: Design Flow

Begonnen wird mit der VHDL Synthese des digitalen Systems mittels der Software „HDL Designer“ sowie der anschließenden Simulation desselbigen per „do-Files“ unter Verwendung des Programms „ModelSim“. Mittels des im ASIC Design Center entwickeltem Cross Compiler „Sirius IDE“ für den in der ePille verwendeten gleichnamigen 16 Bit Microprozessor, werden während der Designphase zeitgleich die benötigten Hardwaretreiber realisiert. Auch diese werden zu-

nächst separat innerhalb der IDE auf ihre Funktionalität überprüft, ehe die Simulation & Verifikation des Zusammenspiels von Hard- und Software erfolgt.

In dieser frühen Phase gründen sich die Erfolgsergebnisse auf reine Softwaresimulation, weshalb im Folgeschritt der Übergang des Systems auf die Hardwareplattform erfolgt. Mittels des FPGA Synthese Tools „Precision Synthesis“ wird das ePillen Design auf einem „Altera Cyclone II“ FPGA Baustein emuliert und parallel mit der Entwicklung des Evaluationboards begonnen, welches diverse externe Peripheriekomponenten sowie Testmöglichkeiten bereitstellt.

Zur endgültigen Abnahme und Verifikation des reibungslosen Zusammenspiels zwischen Hard- und Softwarekomponenten soll letztendlich ein Client-Server basierendes Testprogramm implementiert werden, über welches die Gesamtfunktionalität des ePillen Systems verifiziert werden kann.

Die nachfolgende Dokumentation der Einzelschritte orientiert sich der Abfolge nach am eben beschriebenen Design Flow, und beginnt umseitig mit der Beschreibung der VHDL Synthese.

4. VHDL Synthese

Der Löwenanteil dieses Arbeitsschrittes bestand in der Auswahl und Verschaltung der einzelnen Unterkomponenten des Microcontrollersystems, wobei ein Großteil der verwendeten Funktionsblöcke bereits in fertiger Form aus der eigens für den Schaltungsentwurf realisierten Bibliothek des ASIC Design Centers übernommen wurde.

Im Gegensatz zu ihrer Verwendung in vergangenen Projekten basiert das ePillen System erstmalig auf einer Multiplexer Busstruktur mit separaten I/O Registern anstelle der zuvor verwendeten Tristate Logik mit bidirektionalem Dateninterface. Aus diesem Grund mussten Peripherieeinheiten sowie Buscontroller an diese neuen Gegebenheiten angepasst und bezüglich ihrer Busanbindung leicht modifiziert werden. Der Controller selbst wurde dabei im Zuge des Redesigns vollständig überarbeitet und an die Adress- und Ansteuerlogik des ePillen Systems angepasst. Die Taktversorgung wurde komplett neu konzipiert und verfügt über die Möglichkeit zur Auswahl mehrerer interner als auch externer Quellen, um gegen jedwede Synthese- und Fabrikationsfehler bei der späteren Realisierung als ASIC gewappnet zu sein. Im Zusammenhang hierzu wurde die Energiesparvorrichtung mittels eines so genannten „Wakeup Managers“, welcher den Systemtakt nur bei Eintreten bestimmter Events auf das System schaltet, ebenfalls modifiziert.

Zwei zusätzliche Funktionsblöcke ermöglichen zum einen das gezielte Aufwecken per externem Steuersignal bzw. Signalaktivität an der induktiven Empfangseinheit, als auch die komplette Überbrückung des WUM bei Fehlfunktion.

4.1. Architektur des Gesamtsystems

Herzstück, des in Abb. 3 gezeigten Gesamtsystems ist der eigens im ASIC Design Center entwickelte 16 Bit μ C „Sirius“ im Zusammenspiel mit 128 Byte kombinatorischem ROM, welches den Bootloadercode für das Hochfahren des Systems enthält, sowie 4 kB RAM für BIOS und Anwendungsprogramme.

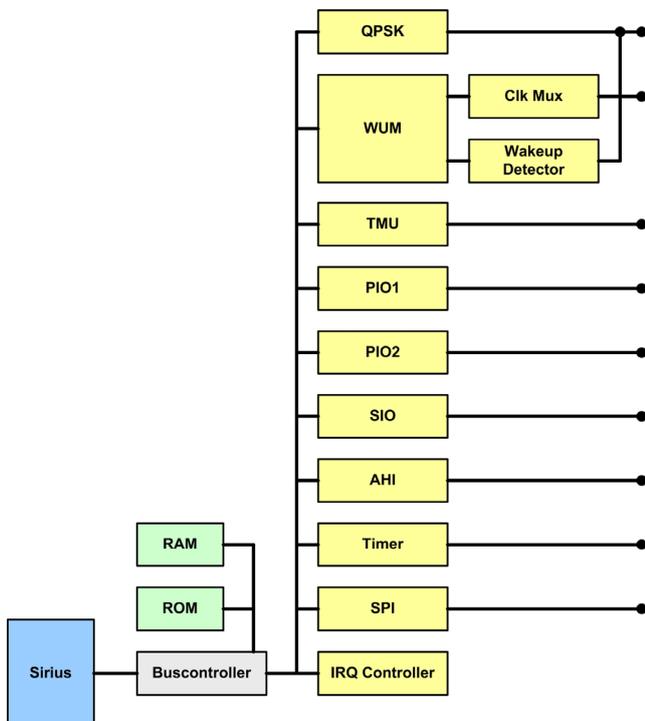


Abbildung 3: Systemarchitektur

Der zentrale Buscontroller regelt Ansteuerung sowie Kommunikationsfluss der Peripheriekomponenten, im Einzelnen bestehend aus:

- Einem QPSK Modem zur Ansteuerung des induktiven Sende-/Empfangsschwingkreises bei Datenübertragung mit der externen Kontrolleinheit.
- Einem so genannten Wakeup Manager, zwecks Energieeinsparung des Gesamtsystems, welcher den Systemtakt eventgesteuert ein- und ausschaltet und die ePille somit gezielt aufwecken bzw. schlafen legen kann.
- Dem Digitalfrontend zur Ansteuerung eines Temperatursensor ICs zur Erfassung von Messwerten.

- Zwei parallelen Schnittstellen (PIO = Parallel I/O), zur Ansteuerung externer Sensoren und Aktoren.
- Einer RS232 Schnittstelle zur Kommunikation mit seriellen Komponenten sowie für den Testbetrieb.
- Einem Frequenzgenerator (AHI = Audio Human Interface) zur Ansteuerung eines externen Piezo Summers zu Testzwecken.
- Einer Timerkomponente zur zeitlichen Steuerung bestimmter Aufgaben und Ereignisse.
- Einem SPI Controller (SPI = Serial Peripheral Interface) zur Ansteuerung eines externen Flash Speichers, aus welchem beim Bootvorgang des Systems BIOS und Anwendungsprogramm in den Arbeitsspeicher geladen werden.
- Einem Interruptcontroller zur Reaktion auf bestimmte Ereignisse, wie z.B. eintreffende Datenpakete über das QPSK Modem oder Timerinterrupts.

5. Entwicklung des Evalboards

Zwecks Test & Verifikation des ePillen Systems in Hardware wurde eine Evaluationsplatine benötigt, welche diverse Peripheriekomponenten enthalten, sowie die Möglichkeit bieten sollte Testsignale abzugreifen und Steuerkommandos einzuspeisen. Die Anforderungen an dieses für die Funktionskontrolle unverzichtbare Bauteil können dabei wie folgt formuliert werden:

- Die Platine sollte den externen Flash Baustein, aus welchem beim Bootvorgang BIOS und Anwendungsprogramm geladen werden enthalten.
- Das Board sollte dabei die Möglichkeit bieten diesen festverlöteten Speicher sowohl über einen externen Programmierer mit Daten zu beschreiben, als auch über Umschalten eines Multiplexers direkt über das System im FPGA auslesen zu können („IN-System-Programmable“). Durch dieses Feature sollte das bisher gängige zeitaufwendige externe Programmieren des SPI Flashs vermieden werden, was ein ständiges Aus- und Einbauen des Bausteins erforderte, und gerade bei der Softwareentwicklung sehr lästig war.
- Eine weitere Anforderung bestand im Design der Treiberkomponenten sowie dem analogen Aufbau des induktiven Sende-/Empfangsschwingkreises.
- Die Schaltung sollte weiterhin die Ansteuerung des bereits existierenden Thermozellen ICs realisieren, welcher jedoch im Gegensatz zu den übrigen Bauteilen mit 5V anstelle von 3.3V versor-

gungsspannung betrieben wird. Diese unglückliche Tatsache erforderte das Design einer bidirektionalen Pegelwandlung bei der Kommunikation mit dem Baustein.

- Neben den unentbehrlichen Mess- und Kommunikationskomponenten sollte das Evalboard zusätzlich eine Anzahl verschiedener virtueller Sensoren & Aktoren enthalten, welche im Nachhinein durch reale Komponenten erweitert werden können. Zur Testzwecken wurden die parallelen Schnittstellen deshalb mit Leuchtdioden bzw. DIP Schaltern bestückt und ein externer Piezo Summer an den Frequenzgenerator angeschlossen.
- Um gegenüber Fehlfunktionen der von der PLL generierten Betriebsfrequenz bei der ASIC Synthese gewappnet zu sein wurde eine alternative Taktversorgung mittels eines auf der Platine enthaltenen 11 MHz Quarzoszillators realisiert.

Die, als Aufsteckplatine für das im ASIC Design Center verwendete Altera Cyclone II FPGA Kit gedachte Testboard ist in den Abbildungen 4 und 5 zu sehen:

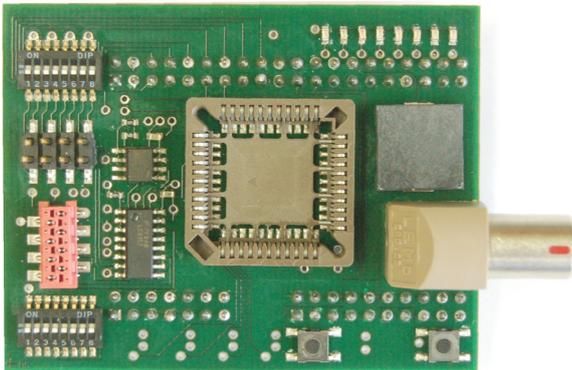


Abbildung 4: Vorderseite Evalboard

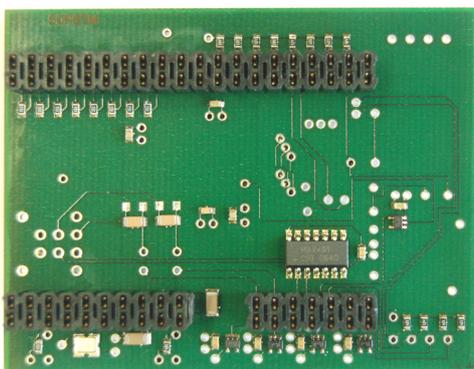


Abbildung 5: Rückseite Evalboard

6. Hardwaretreiber

Eine unabdingbare Voraussetzung für die Inbetriebnahme und Verifikation des Systems besteht in der Implementierung von Hardwaretreibern für die einzelnen Peripheriekomponenten. Zu dieser Aufgabe gehörte neben der Treibererstellung insbesondere auch die Programmierung von Interrupt Service Routinen für einige Funktionsblöcke, um auf bestimmte Ereignisse, wie etwa Timerevents oder das Eintreffen von Datenpaketen über die QPSK reagieren zu können.

Eine der Hauptanforderungen an das Softwaredesign besteht darin die, in der Maschinensprache des Sirius Microprozessors realisierten Treiberrountinen kompatibel zur Programmiersprache C zu implementieren um auf der so geschaffenen Basis spätere Anwendungsprogramme, wie etwa auch die Testbench, in einer Hochsprache erstellen zu können.

Bereits existierende Elemente wie Treiber und ISR der SIO, sowie Steuerfunktionen des SPI Controller wurden 1 zu 1 in das ePillen BIOS übernommen. Darüber hinaus wurden die folgenden zusätzlichen Softwareblöcke realisiert:

Interrupt Service Routinen	
QPSK	
Name	Beschreibung
int_tx_qpsk	Sendeinterrupt der QPSK
int_rx_qpsk	Empfangsinterrupt der
Timer	
isr_timer0	Timerinterrupt
Hardwaretreiber (1)	
PIO Treiber	
C Syntax	Beschreibung
void writepio1(char va-	Schreibt PIO1 Datenregis-
void setpio2(char mask)	Konfiguriert PIO1 I/O
char readpio1()	Liest PIO1 Datenregister
void writepio2(char va-	Schreibt PIO2 Datenregis-
void setpio2(char mask)	Konfiguriert PIO2 I/O
char readpio2()	Liest PIO2 Datenregister
AHI Treiber	
void sound(char f, char	Erzeugt ein Tonsignal.
WUM Treiber	
void setmask(char msk)	Setzt WUM Maskenregister
char readwummaske()	Liest WUM Maskenregister
void writewume-	Setzt WUM Eventregister
char readwumevent()	Liest WUM Eventregister
void sleep()	Legt das System schlafen

Hardwaretreiber (2)	
Timer Treiber	
C Syntax	Beschreibung
<code>void startTimer (int time)</code>	Setzt den Timer.
TMU Treiber	
<code>measure(char resolution)</code>	Misst Temperaturwert
QPSK Treiber	
<code>void sendMsg(char* msg)</code>	Sendet Daten via QPSK
<code>char* readMsg(char str)</code>	Empfängt Nachricht
<code>char readByte()</code>	Empfängt einzelnes
Utilities	
<code>void wait(char time)</code>	Warteschleife
<code>void halt()</code>	Stoppt den Prozessor
<code>char getport(char nr)</code>	Liest Peripherieport
<code>void putport(char nr,char</code>	Schreibt Peripherieport

- Kontrolle der QPSK mitsamt Treibern und Interrupt Service Routinen.
- Prüfung der Funktionalität des Wakeup Managers, seiner Treiberrountinen, sowie der neu entwickelten Eventsteuerung.
- Funktionsanalyse der Timerkomponente samt Treiber und ISR.
- Leistungskontrolle der TMU sowie der zugehörigen Softwareroutine.
- Verifikation der parallelen Schnittstellen und ihrer Softwarekomponenten.
- Überprüfung des AHI Interfaces sowie seines Treibers.
- Funktionstest der zum Debuggen sehr nützlichen Utility Functions, wie dem Auslesen und Beschreiben von Port- und Speicheradressen.

7. Testbench

Von der Architektur her orientiert sich die ePille Testbench am Client-Server Modell. Eine in Java programmierte Terminalsoftware realisiert die Kommunikationsverbindung mit der ePille über die serielle Schnittstelle des PCs und steuert den Microcontroller via Kommandomacros. Der in „C“ implementierte Testbench Counterpart im System wertet diese Befehle aus, ruft die entsprechenden Treiberrountinen auf und sendet eine Antwort über den COM Port zurück an den Client.

Die Struktur des Testprogramms wurde absichtlich auf eine Client- sowie eine Serverapplikation verteilt, um einen möglichst großen Teil der Datenverarbeitung, wie z.B. Überprüfung fehlerhafter Eingaben, Auswertung von Ergebnissen, sowie das Exception Handling aus der ePille auszugliedern. Durch diese Vereinfachung der Serverseite wird derjenige Part des Testprogramms, welcher innerhalb des Microcontroller-systems ausgeführt wird, zum einen erheblich kleiner aber vor allem wesentlich unanfälliger gegen Fehler, da hier ausschließlich Softwarefunktionen ablaufen welche zum unmittelbaren Betrieb des Systems notwendig sind und so gut wie keinen Overhead.

7.1. Funktionalitäten

Zu den Funktionalitäten der Testbench gehören, Kontrolle aller Treiber, ISR und HW-Komponenten. Die Testfunktionen umfassen im Einzelnen:

- Test der SIO sowie der COM Port Treiber und Interruptrountinen.

Nachfolgende Abbildung zeigt anhand der Aufnahme sowie graphischen Darstellung mehrerer Temperaturwerte, exemplarisch eine der zahlreichen Funktionalitäten des Client Terminalprogramms.

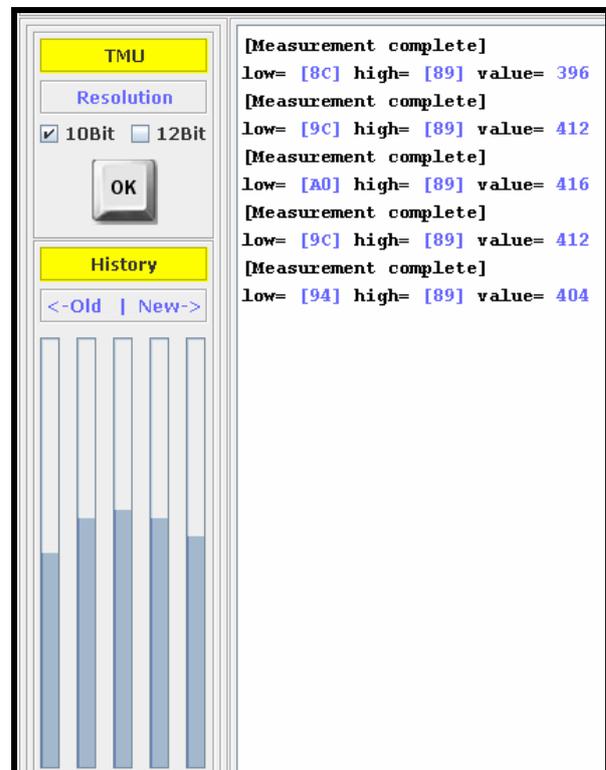


Abbildung 6: Verifikation der Temperaturmessung

8. Zusammenfassung

Die Ergebnisse dieser Arbeit können wie folgt zusammengefasst werden:

Es wurde ein komplettes Microcontrollersystem samt Peripheriekomponenten in VHDL zusammengestellt, simuliert und auf einen Altera Cyclone II FPGA synthetisiert. Die einzelnen Komponenten wurde dabei größtenteils aus der bestehenden Bibliothek des ASIC Design Centers übernommen, an die ePillen Systemarchitektur angepasst und modifiziert, teilweise jedoch komplett neu entwickelt.

Um das Gesamtsystem auf einer Hardwareplattform testen und verifizieren zu können, wurde ein Evaluationsboard entwickelt, welches benötigte Peripheriekomponenten wie etwa den externen SPI Flash oder den Thermozellen IC enthält, sowie zahlreiche Test- und Messmöglichkeiten bietet.

Um das System letztendlich in Betrieb nehmen zu können wurde ein auf das ePillen System angepasstes BIOS zusammengestellt, sowie Treiber und Interrupt Service Routinen für die einzelnen Hardwarekomponenten implementiert.

Die abschließende Verifikation erfolgte mittels einer eigens entwickelten Testbenchapplikation anhand derer die volle Funktionsfähigkeit sämtlicher Hard- und Softwareblöcke sowie ihr reibungsloses Zusammenspiel nachgewiesen wurde.

Bei Gegenüberstellung der Aufgabenbeschreibung dieser Diplomarbeit mit den erreichten Ergebnissen, konnten sämtliche Anforderungen sowie Spezifikationen erfüllt und durch Tests verifiziert werden.

- Die VHDL Synthese des ePillen Systems mitsamt der Anpassung bereits vorhandener Komponenten sowie Hinzufügen zusätzlicher Features bietet eine solide Basis für ein voll funktionsfähiges Microcontrollersystem.
- Mit der Evaluationsplatine konnte eine komfortable Testplattform zur Verifikation und Test aller Funktionalitäten zur Verfügung gestellt werden. Layout und Schaltplan sind fehlerfrei und bedürfen keinerlei Verbesserungen oder Modifikationen, so dass der bestehende Entwurf bedenkenlos zur Fertigung zusätzlicher Boards verwendet werden kann.
- Den zukünftigen Softwareentwicklern des ePillen System wurde durch voll funktionsfähige Hardwaretreiber und Interrupt Service Routinen eine solide Hardwareabstraktionsebene zur Verfügung gestellt, so dass spätere Applikationen bequem in der Hochsprache „C“ implementiert werden können.

nen. Die Treiberrouinen erfüllen alle denkbaren Anwendungsfälle und bilden an und für sich bereits eine komplette API für komplexe Anwendungsprogramme oder sogar ein eigenes Betriebssystem.

- Mit der verteilten Testbench Applikation wurde die Funktionsfähigkeit sämtlicher Hard- und Softwarekomponenten erfolgreich verifiziert. Durch das grafische User Interface, sowie die Debuggingfunktionalitäten (Port I/O und Memory I/O) ist eine schnelle und bequeme Funktionsüberprüfung der ePille möglich. Die modulare Softwarearchitektur von Client- und Serverapplikation, sowie des Kommunikationsprotokolls kann leicht erweitert werden und ermöglicht die nachträgliche Realisierung von Zusatzfeatures.

9. Referenzen

- [1] Daniel Bau, IAF Report 07/2006 Projekt Wearlog Technischer Bericht, Hochschule Offenburg, 2006
- [2] Dirk Jansen, Softwaredokumentation „Sirius IDE Hochschule Offenburg, 2006
- [3] Markus Mogel, „Entwicklung eines synchronen Transceivers mit DQPSK Modulation und Soft Shift Keying für induktive Datenübertragung mit Erprobung in einem FPGA “
Diplomarbeit, Hochschule Offenburg, 2006
- [4] Nidal Fawaz, „Development of CP_DQPSK Modulator und Demodulator using VHDL for inductive data transmission“
Master Thesis, Hochschule Offenburg, 2002
- [5] Stefan Mescheder, „Erweiterung, Optimierung, Verifikation und Layout einer integrierten Schaltung als aktives Sensorsystem“
Diplomarbeit, Hochschule Offenburg, 2002
- [6] Mark Durrenberger, „Überarbeitung eines 32 Bit Prozessorkerns mit Optimierung der ALU für die Synthese in eine 0.35um CMOS Bibliothek und Erprobung in einem FPGA“
Diplomarbeit, Hochschule Offenburg, 2006
- [7] Mark Durrenberger, Daniel Bau, „Komponentenbibliothek für die ASIC Synthese“
Technischer Bericht, Hochschule Offenburg, 2006

Assoziativer Koprozessor für schnelle Textsuche in umfangreichen Datenbanken

Christophe Layer

Universität Ulm, Institut für Mikroelektronik

Albert-Einstein-Allee 43, D-89081 Ulm

christophe.layer@uni-ulm.de

Durch die fortschreitende Entwicklung von großen Festplatten und anderen Speichermedien hoher Kapazität bilden sich riesige Entropiereservoirs in Form von alten, aber nicht verworfenen Informationen. Daher müssen IT-Dienste effektive und intelligente Funktionen zum Auffinden von Daten zur Verfügung stellen. Ziel ist es, mittels einer speziellen Hardware-Architektur den Hauptprozessor eines Rechners zu entlasten, die spezialisierten Berechnungen auszulagern und dadurch Zeit zu gewinnen. Der VLSI-Schaltungsentwurf auf FPGA-Basis zeigt, dass eine standardmäßige FPGA-Entwicklungsplattform trotz einer viel kleineren Taktung im Vergleich zu den neuesten CPUs mehrere Größenordnungen schneller ist als eine Software-Implementierung auf PC-Basis.

1 Einleitung

Obwohl General-Purpose-Prozessoren rekordverdächtige Geschwindigkeiten erreichen, müssen immer noch Hunderte von Gigabytes zwischen Speicher und Prozessor über Bus-Systeme eingeschränkter Größe hin- und hertransportiert werden, um eine Handvoll relevanter Kilobytes aus einem umfangreichen digitalen Speicher zu entnehmen. Die einzige langfristige Lösung könnte sein, diese Aufgabe in eine bestimmte Hardware-Plattform in der Nähe des Speichermediums auszulagern.

Um dieses Problem zu lösen, entwickeln wir einen Koprozessor für die Textsuche, der wie ein assoziatives Cache-System arbeitet und eine größere Bandbreite für den Zugriff auf den Speicherbereich der Datenbank bietet. Der Suchprozess, der einen kompletten Durchgang aller Einträge nötig macht, wird in zwei Phasen aufgeteilt. In der Filterungsphase werden alle für die Suchanfrage relevanten Dokumente ausgewählt, während in der Verifikationsphase ein besseres Matching auf einer höheren Ebene ermöglicht wird. Da diese Auswertung eine komplexe Aufgabe mit wenigen spezialisierten Funktionen ist, deren tatsächliche Implementierung sich öfters ändert, verbleibt sie in der Software-Ebene. Die Filterung ist andererseits ein sich wiederholender Vorgang, der eine bestimmte Distanz innerhalb einer binären Eigenschaftsmatrix berech-

net, wofür sich eine Hardwareplattform anbietet. Aus diesem Grund haben wir am Institut ein System untersucht, das letztendlich auf einem FPGA-Prototypingboard implementiert wurde. Dieser Beitrag soll sowohl die Entwicklung einer skalierbaren und hochgradig parallelisierbaren Architektur zeigen, als auch die Untersuchung der Sortieralgorithmen und deren Implementierung unter strengen Timing- und Flächenbeschränkungen darstellen.

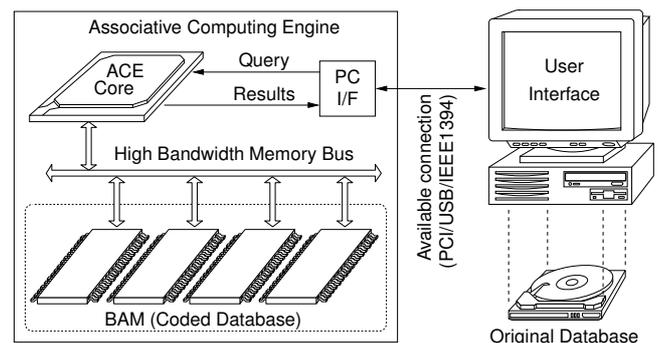


Abbildung 1: Hardware-Lösungsvorschlag für die beschleunigte Bearbeitung von Suchanfragen, bei dem die Datenverarbeitung ausgelagert wird.

Wie in Bild 1 dargestellt, ist unser System eine Hochleistungs-Suchmaschine für Datenbanken, die den Hauptprozessor eines Computers von komplizierten Suchanfragen entlastet. Der hier präsentierte ACE (Assoziative Computing Engine) basiert auf einem einheitlichen Ansatz, der auf Datenbanken verschiedenster Struktur anwendbar ist, von Multimedia-Applikationen bis hin zu reinen Textdaten. Im Anschluss an diese Einleitung gibt Abschnitt 2 einen Überblick über die Suchprozedur und erklärt die assoziative Berechnungsmethode. Abschnitt 3 präsentiert die für die hohe Geschwindigkeit verantwortliche interne Architektur des ACE, insbesondere die Art, in der Speicherzugriffe erfolgen müssen. Der letzte Abschnitt diskutiert die Leistung unseres Hardware-Prototyps im Vergleich zu verschiedenen Software-Lösungen und zeigt außerdem ein Simulationsmodell auf, das auf neueren FPGA- und DRAM-Technologien basiert.

2 Fehlertolerante Textsuche

In den vergangenen Jahrzehnten wurde eine Vielzahl von Suchalgorithmen entwickelt, untersucht und veröffentlicht, z. B. in [1–3, 9, 17, 18]. Auch heute sind Texte immer noch die bedeutendste Form des Informationsaustausches, obwohl die Daten in vielen erweiterten Formaten gespeichert werden.

2.1 Distanz zweier Sequenzen

Ein wichtiges Maß für die Unterschiedlichkeit ist die Edit-Distanz, die einige gängige Distanzen wie Levenshtein oder Hamming umfasst [16]. In der Informationstheorie ist die Hamming-Distanz zwischen zwei Zeichenfolgen die Anzahl der Stellen, an denen das entsprechende Symbol unterschiedlich ist, mit der Einschränkung, dass die Folgen dieselbe Länge haben. Eine Erweiterung dieser Maßzahl ist die Edit-Distanz, bestimmt durch die Mindestzahl an notwendigen Operationen zur Transformation einer Zeichenfolge in eine andere, wobei eine solche Operation aus Einfügen, Löschen oder Austausch eines einzelnen Symbols besteht. Eine Kostenfunktion, die die Edit-Distanz zwischen zwei Zeichenfolgen $x_{1\dots i}$ und $y_{1\dots j}$ berechnet, kann beispielsweise so definiert werden:

$$c_{i,j} = \min(c_{i-1,j-1} + d_{i,j}, c_{i-1,j} + 1, c_{i,j-1} + 1) \quad (1)$$

mit $c_{0,0} = 0$, wobei die Hamming-Distanz $d_{i,j} = 0$ wenn $x_i = y_j$ und sonst 1 beträgt, gemäß [17]. Eine Parametrisierung von (1) ist möglich, indem gewichtete Kostenfaktoren für die Penalty-Werte bei Austausch, Einfügen oder Löschen verwendet werden, wie es als Beispiel im Beitrag von [6, 7] diente. Allerdings ist aus Symmetriegründen das Einfügen eines Symbols in einer Zeichenfolge äquivalent zum Löschen in der anderen Zeichenfolge.

2.2 Hash-Funktionen

Hashing wird als Tabellen-Suchmethode benutzt, wobei die Position eines Schlüssels in der Datenstruktur mit Hilfe einer Hash-Funktion direkt aus dem Schlüsselwert berechnet wird. Hierbei kann auf Daten, die mit dem Schlüssel verknüpft sind, direkt und mit einer konstanten durchschnittlichen Suchzeit $O(1)$ zugegriffen werden. Somit wird die Zeitkomplexität $O(\log n)$ als Resultat einer Suche in binären Bäumen oder mit anderen sequentiellen Suchmethoden überwunden. Die Abfragemethode wendet zum Finden eines Schlüsseleintrags in Form einer Zeichenkette $x_{0\dots m-1}$, eine Funktion h an, definiert als

$$h(x) = \sum_{i=0}^{m-1} x_i \cdot B^i \bmod W \quad (2)$$

wobei m die Schlüssellänge, B eine Zahl als Polynombasis und W die Breite der Hash-Tabelle bedeutet. Üblicherweise ist W eine beliebig große Primzahl, deren

Ziel es ist, eine möglichst gleichförmige Verteilung von $h(x)$ über W für jedes x zu erreichen. Die Berechnung von $h(x)$, die die Position des Schlüsseleintrags x in der Tabelle mit $0 \leq h(x) < W$ liefert, kann in der Zeitkomplexität $O(1)$ durchgeführt werden [17].

Die typische Suchoperation kann durch bitweises Vergleichen von Signaturen s_Q und s_T einer Anfrage Q und eines Textes T durchgeführt werden. Die Hamming-Distanz $d(Q, T)$ zwischen der Maske $(s_Q \& s_T)$ und der Anfrage-signatur s_Q entspricht dem Übereinstimmungsgrad von Q zu T und kann mit den einfachen logischen Verknüpfungen

$$d(Q, T) = \text{Zahl_der_Einsen}\{(s_Q \& s_T) \oplus s_Q\} \quad (3)$$

abgeschätzt werden, wobei die Symbole $\&$ und \oplus den bitweisen UND- bzw. XOR-Operationen entsprechen. Selbst wenn eine hiermit errechnete Null-Distanz auf Grund der vorhandenen Kollisionswahrscheinlichkeit keine exakte Übereinstimmung garantiert, ist diese Methode äußerst schnell.

2.3 Assoziatives Suchverfahren

Die Suchmethode von Lapir [5, 10] basiert auf Hash-Funktionen. Anfragen werden dabei lokal in einer codierten Datenbank durchgeführt, d. h. einem Index, der als BAM (Bit Attribute Matrix) bezeichnet wird und ein Abbild des ursprünglichen Textcontainers ist. Wie in Bild 2 dargestellt, beinhaltet der Suchalgorithmus zwei Phasen. Zuerst liefert die Filterungsphase mit approximativem Abgleich eine Liste relevanter Dokumente aus der BAM zurück, und dann berechnet die Evaluationsphase über

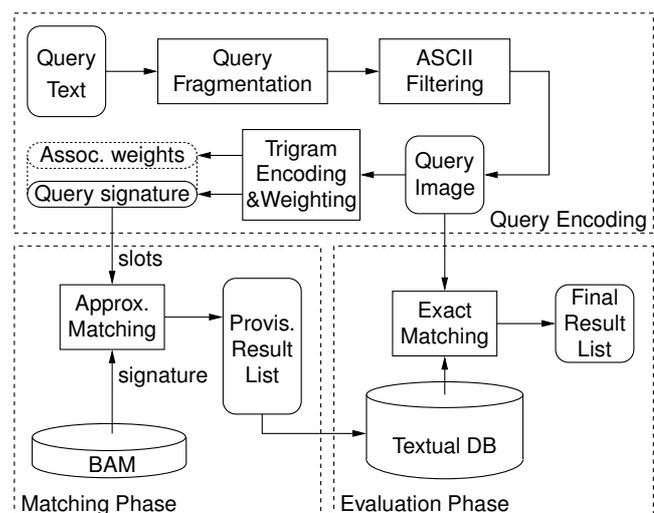


Abbildung 2: Die Filterungs- und Evaluationsphase der AAM entsprechen einem approximativem und exaktem Abgleich. Sie erfolgen nach Signatur-Encodierung der Anfrage u. A. mittels Trigramm basierte Hash-Funktionen.

einen präziseren Abgleich den Relevanzfaktor der Dokumente unter Berücksichtigung der tatsächlichen Anfrage. Diesen beiden Phasen geht eine Vorverarbeitung der Anfrage Q voraus, die ASCII-Filterung, Fragmentierung von langen Anfragen, Trigramm-Encodierung in Anfrage-Slots sowie Gewichtung der Wörter entsprechend ihrer semantischen Bedeutung umfasst. Die erste Phase wird in Verbindung mit der BAM verwendet, um eine große Zahl von nicht relevanten Dokumenten von der Suche auszuschließen. Danach werden die verbleibenden Kandidaten in absteigender Reihenfolge nach Relevanz sortiert und an die nächste Phase übergeben. Um dieses vorläufige Suchergebnis zu verfeinern, ist die zweite Phase weitaus komplexer. Sie kann aber auch sehr kurz sein, sofern die erste Phase bereits genügend selektiv war.

2.4 Bewertung der Arithmetik

Als erstes muss eine gestellte Anfrage fragmentiert werden. Basierend auf den Informationen in den Signaturen der BAM, z. B. das Vorkommen eines bestimmten Trigramms in der Seite, werden schlecht passende Seiten mehr oder weniger stark mit Penalty belegt, entsprechend der Wertigkeit von verschiedenen Wörtern, aus denen ein Anfragefragment besteht. Eine "Mismatch Penalty" $\psi(n)$ wird zugewiesen, falls eine Seite das Trigramm n aus der Anfrage nicht enthält. Eine "Switch Penalty" $\sigma(n)$ wird zugewiesen, wenn ein Trigramm n in der Abfolge der Suchprozedur nicht seinem Vorgänger $n-1$ folgt. Der Grund hierfür ist, dass wenn aufeinanderfolgende Merkmale in der Seitensignatur gefunden werden, es sehr wahrscheinlich ist, dass sie zum selben Satz im Originaldokument gehören; und umgekehrt. Daher gilt, je kleiner der Penalty-Wert ist, desto größer die Ähnlichkeit zu einem in der BAM-Seite p gefundenen Suchfragment. Bei der Verarbeitung eines Fragments f liefert der gesamte Penalty-Wert

$$\mathcal{P}_p(f) = \sum_n (\sigma(n) + \psi(n)) \times \omega_n \quad (4)$$

wobei ω_n die mit dem Anfrage-Slot s verknüpfte Wertigkeit bedeutet. Danach werden die für eine BAM-Seite ermittelten Penalty-Werte miteinander kombiniert, um eine globale Bewertung zu ermitteln. Das Auswertungsergebnis S_p wird nach jedem Anfragefragment f angepasst, indem der Logarithmus des Penalty-Wertes $\mathcal{P}_p(f)$ mit einer Konstante \mathcal{K} ergänzt wird:

$$S_p = \sum_f (\mathcal{K} - \log_2 \mathcal{P}_p(f)). \quad (5)$$

Der Zweck des Logarithmus ist, dass große Werte nur für sehr kleine Fragment-Penalty-Werte $\mathcal{P}_p(f)$ aufsummiert werden. Im Falle einer linearen Funktion würden große Penalty-Werte von schlecht übereinstimmenden Fragmenten das Ergebnis beeinflussen und die kleinen

Penalty-Werte von gut übereinstimmenden Fragmenten unterdrücken [10]. Die verbleibenden Verarbeitungsschritte in der ersten Phase sind Sortieren aller Seiten entsprechend ihres Ergebnisses S_p und die Auswahl der Besten.

3 System-Design

Da die erste Phase am umfangreichsten ist, vor allem wenn man die enorme Zahl von Seiteneinträgen in der BAM berücksichtigt, gibt es dort einen besonderen Optimierungsbedarf bei Speicherzugriffen sowie der Prozessmethode. Daher haben wir uns entschieden, einen Hardware-Beschleuniger zu implementieren, der die Berechnungen zum approximativen Matching durchführt.

3.1 Speicherzugriffe

Für einen schnellen Direktzugriff auf die Daten, was wir für die Speicherung der BAM zu erreichen versuchen, gibt es nur zwei sinnvolle Alternativen. Bei der ersten handelt es sich um die schnelle aber verhältnismäßig teure SRAM (Static Random Access Memory) und bei der zweiten um die sehr dichte aber recht komplexe SDRAM (Synchronous Dynamic Random Access Memory) Technologie. Als Ausgangspunkt für den Datenfluss muss die BAM-Schnittstelle einen flexiblen und schnellen Zugriff auf die BAM-Daten bieten. Die Besonderheit unseres Systems liegt darin, dass bitweise vertikal auf die BAM zugegriffen wird, d. h. per Attribut anstatt horizontal per Signatur. Dies ermöglicht das Auslesen eines N bit-Vektors, wie in Abb. 3 dargestellt, und daher die parallele Verarbeitung von N Seiten durch die Recheneinheiten.

Die Aufgabe des BAM-Interface ist die Transformation einer Liste von Anfragen-Slots in physikalische Adressen des Speichermediums in Bezug auf die Abszisse, d. h. die tatsächliche Slot-Position in der virtuell erstellten Anfragenignatur, und in Bezug auf die Ordinaten, d. h. die Adresse der Seite $P+1$. Dementsprechend können sich

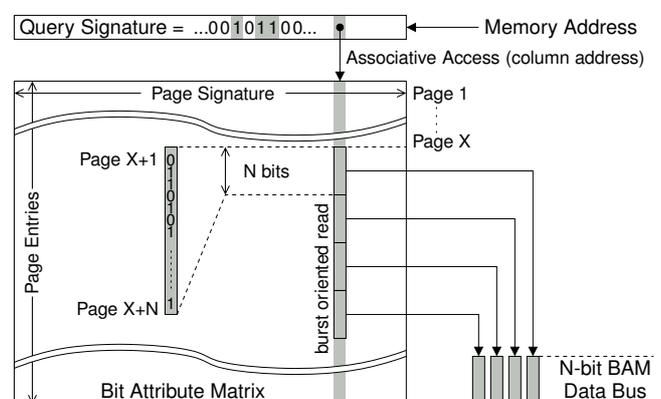


Abbildung 3: Vektorisierte Burst-Datenzugriffe auf die Bit Attribut Matrix, gesteuert vom BAM-Interface.

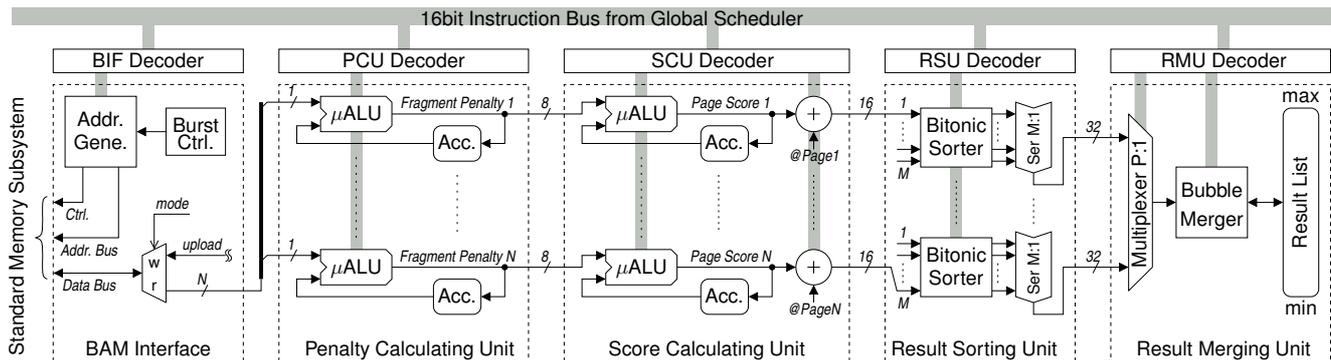


Abbildung 4: Interne Hardware-Architektur des ACE mit den Modulen PCU, SCU, RSU und RMU, die die erste Phase des Suchalgorithmus ausführen sowie ein BAM-Interface repräsentieren, welches unabhängig vom verwendeten Speichertyp einen effizienten Datenaustausch ermöglicht.

die nächsten aus der BAM gelesenen N bit-Datenworte entweder an der nächsten angezeigten Spalte in der Liste von Anfragen-Slots der Seiten $P+1$ bis $P+N$ befinden, etwa eine horizontale Bewegung in Bild 3, oder auf den folgenden Seiten $P+N+1$ bis $P+2N$ mit demselben Attribut, etwa eine vertikale Bewegung innerhalb derselben Spalte. Diese Entscheidung liegt beim globalen Controller und hängt von der Charakteristik der verwendeten Speicherelemente ab. In unserem Design kommt eine Mischung aus beiden Richtungen zum Einsatz, was sich in Form eines Zickzack-Musters für die optimale Nutzung des Speicher-Interface äußert.

3.2 Modulare Architektur

Unser Ziel ist die Entwicklung einer modularen Architektur, die in der Lage ist, N Seiten der BAM mit einem sehr hohen Datendurchsatz parallel zu verarbeiten, ausgehend vom Speicherzugriff bis hin zur sortierten Ergebnisliste. Wie in [13] eingeführt und in [14] weiter ausgearbeitet wurde, ist unser Modell ein assoziativer Prozessor vom Typ SIMD (Single Instruction Multiple Data) mit mehreren Ebenen von spezialisierten Prozessorelementen (PEs), angeordnet in einem irregulären systolischen Array. Den PEs werden spezielle Aufgaben und verschiedene Wortbreiten zugeordnet, damit sie optimal im Suchalgorithmus platziert werden können. Die Gesamtarchitektur wird von einem globalen Controller gesteuert, der die Hash-Anfragen interpretiert und in einfache Instruktionen für die einzelnen Einheiten übersetzt. An dieser Stelle opfern wir etwas Flexibilität zugunsten einer höheren Leistung, indem wir einen spezialisierten assoziativen Prozessor realisieren. Wie in Bild 4 veranschaulicht, besteht das System aus fünf Prozessormodulen:

- Das BIF (BAM Interface) ist der erste Prozessblock, dessen Aufgabe es ist, einen schnellen Zugriff auf die vom nächsten Modul benötigten BAM-Informationen durchzuführen. Er muss die Hash-Werte der Anfrage in die von der Hardware-Zielplattform abhängenden

physikalischen Speicheradressen umsetzen.

- Die PCU (Penalty Calculating Unit) ist verantwortlich für die Berechnung der Fragment-Penalty gemäß (4). Sie muss die von der BAM kommenden Daten und die Anfrage-Informationen vom globalen Controller, z. B. die Wertigkeit jedes einzelnen Anfrageworts, mit hoher Geschwindigkeit verarbeiten.
- Die SCU (Score Calculating Unit) aktualisiert den Ergebniswert der Seite inkrementell nach (5), mit dem Penalty-Wert eines Fragments aus der PCU. Die Hardware-Implementierung der Logarithmus-Berechnung nach [15] wird in [11] detailliert geschildert, mit Berücksichtigung einer eventuellen Skalierbarkeit.
- Die RSU (Result Sorting Unit) hat die Funktion, die Seiten in der Reihenfolge ihrer Relevanz zur Anfrage in Abhängigkeit von ihrem Ergebnis zu sortieren. Dies beinhaltet Vergleichs- und Austauschoperationen und benötigt ein internes Sortiernetzwerk gemäß [4, 9, 19]. Das Bitonic Sortiernetz ist ein Sortierverfahren, das sich durch seine reguläre Struktur auszeichnet. Dessen Hardware-Implementierung wird in [12] genau beschrieben.
- Die RMU (Result Merging Unit) als letzter Prozessblock ist der RSU sehr ähnlich und bezweckt die Verbindung der neuesten sortierten Resultate in eine endgültige Ergebnisliste der Länge R . Bei dem hier verwendeten Sortierverfahren handelt es sich um das Odd-Even-Merge Sort gemäß [4, 9].

In unserer modularen Architektur (vgl. Bild 4) umfasst der Datenpfad zwei wesentliche Teile, die sich grundsätzlich in ihrer Art der Datenverarbeitung unterscheiden. Der erste Teil besteht aus der PCU und der SCU und die Daten werden voll parallel verarbeitet. Der zweite Teil des Datenpfads setzt sich aus RSU und RMU zusammen und widmet sich einem speziellen Problem, bei dem nicht nur der Durchsatz, sondern auch die Skalierbarkeit eine große Rolle spielen.

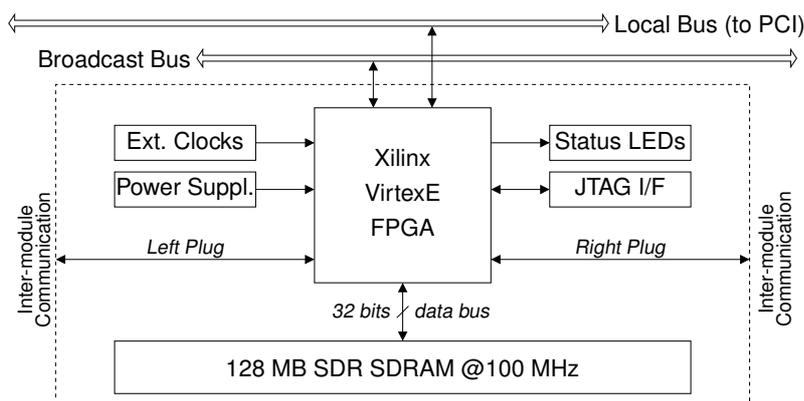


Abbildung 5: Der Hardware-Prototyp umfasst einen Xilinx FPGA der VirtexE-Serie und 4 SDRAM-Speicherchips, die insgesamt 128 MB über einen 32 bit-Datenbus bei 100 MHz Taktfrequenz erreichbar machen.

4 Ergebnisse

Das ACE-System wurde als Prototyp auf einer FPGA-Plattform implementiert, die eine praktische Verifizierung der Effizienz unseres Ansatzes ermöglicht.

4.1 Hardware-Implementierung

Durch den Entwurf einer skalierbaren und flexiblen Architektur konnte eine Lösung bereitgestellt werden, die sich für die meisten aktuell auf dem Markt erhältlichen Hardware-Plattformen eignet, ohne eine Eigenentwicklung auf Basis von FPGA- oder ASIC-Technologie auszuschließen. Darüber hinaus legten wir uns beim Systementwurf aus oben genannten Gründen auf Standard-DRAMs fest, die momentan die beliebtesten Bausteine für besonders speicherintensive Anwendungen sind.

Das an der Universität Paderborn entwickelte modulare Prototyping-System für integrierte Schaltkreise namens Raptor 2000 [8] integriert alle wichtigen Komponenten, um Schaltungen und System-Designs mit einer Komplexität von bis zu 60 Millionen Transistoren zu realisieren. Die wichtigsten Eigenschaften des Boards (Abb. 5) sind die Fähigkeit, das System über ein PCI-Interface mit einem PC zu verbinden, die Möglichkeit zum Ergänzen vieler Erweiterungsmodule auf demselben Motherboard und das Vorhandensein eines zusätzlichen Broadcast-Bus, der es wiederum dem lokalen Multi-Master-Bus ermöglicht, die PCI-Anbindung zu übernehmen.

Heute kann diese Plattform mit Recht als veraltet bezeichnet werden, da die Technologie in der Zwischenzeit große Fortschritte gemacht hat, nicht nur in Bezug auf die Transistorgröße, sondern auch in Bezug auf Standards. Andererseits zeigen die Ergebnisse der Implementierung nicht nur, dass die Architektur eine sehr kleine Fläche aufweist, sondern dass das System für Hochgeschwindigkeits-Speicherbausteine wie die vorhandenen 100 MHz SDRAM-Chips geeignet ist. Auf einem Xilinx-FPGA vom Typ XCV2000E werden weniger

als 7000 Slices (<40%) für das vollständige Design verwendet, wobei die Länge der Ergebnisliste willkürlich auf einen recht hohen Wert von $R=100$ Stellen in der RMU gesetzt wurde. Die Größe des SRAM-Speichers auf dem Chip liegt auf Grund des vorhandenen Instruction-Speichers, der die übersetzte Anfrage vorhält, bei 2 KB. Allerdings kann festgestellt werden, dass unser Ziel erreicht wurde, da die endgültige Synthese des ACE eine Frequenz von über 100 MHz ermöglicht. Die Bezeichnung des derart realisierten Modells lautet "Hardware 1" bzw. ACE_{100MHz} .

4.2 Skalierbarkeit

In unserer Arbeit wurde ein besonderes Augenmerk auf den Entwurf eines flexiblen parametrisierbaren Systems gelegt, das nicht nur eine Lösung für verschiedene Hardware-Plattformen darstellt, sondern dessen Skalierbarkeit es auch erlaubt, höheren Leistungsanforderungen wie auch größeren bzw. kleineren Busbreiten gerecht zu werden. Durch die Untersuchung von Sortiernetzwerken und verschiedenen Implementierungsmethoden, die die Variation von Bitbreiten und der Zahl der zu sortierenden Einträge zulassen, konnten wir eine erweiterte Architektur ausarbeiten, die eine Adaptierung von Sortier-Subsystemen an jede Art von Hardware-Plattform sowie einen Kompromiss zwischen Flächenbedarf und benötigtem Datendurchsatz erlaubt. Eine Synthese des RTL-Codes in VHDL, gefolgt von Platzierung und Verdrahtung auf der nächsten FPGA-Generation, z. B. der Xilinx Virtex-II Familie, lieferte dieselben Flächenanforderungen bei einer Verdopplung der internen Taktfrequenz auf über 200 MHz. In einem solchen Fall muss das Speicher-Subsystem ebenfalls hochskaliert werden, wodurch DDR (Double Data Rate) SDRAM-Bausteine mit doppeltem Datendurchsatz zum Einsatz kommen müssen. Das Schlüsselement des neuen Designs ist der breite Datenbus zwischen den Prozessoreinheiten und der BAM, auf dem bei jeder Flanke eines ty-

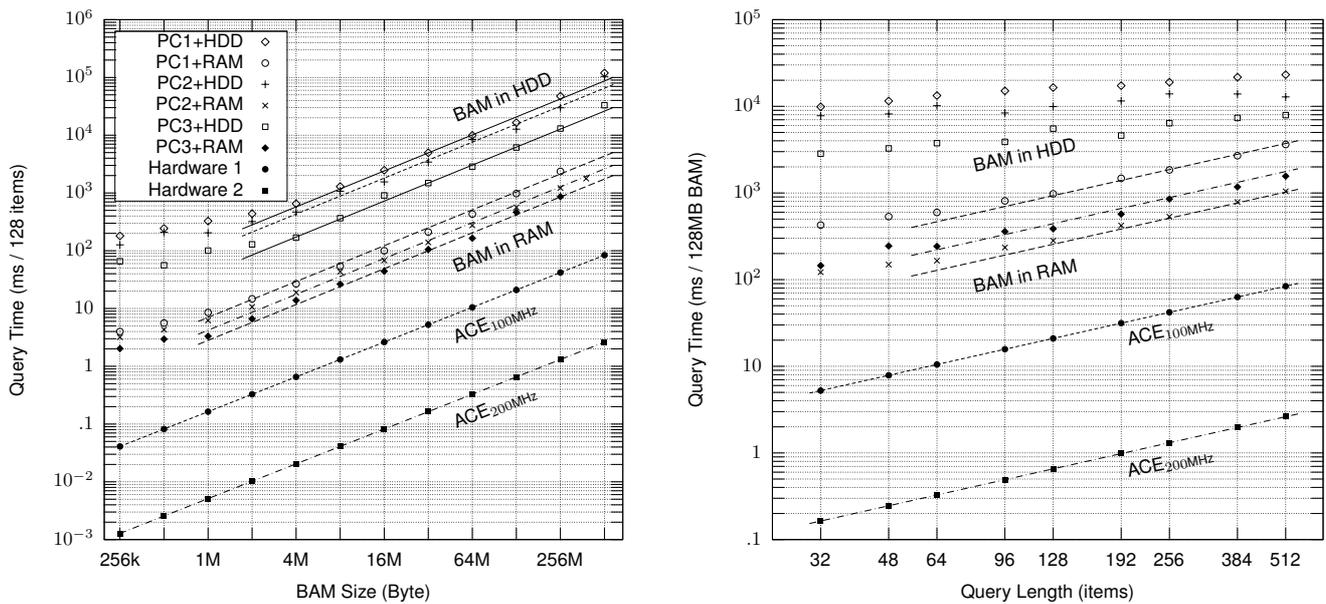


Abbildung 6: Veranschaulichung der Anfragezeit der ACE-Modelle und verschiedener Rechnerarchitekturen über der BAM-Größe für eine Anfrage von 128 Slots (links) und über die Anfragelänge für eine BAM von 128 MB (rechts).

pischerweise auf 200 MHz spezifizierten Takts 256 Bits übertragen werden können, was einem immensen Datendurchsatz von 12,8 GB/s entspricht. Als direkte Konsequenz der Busverbreiterung und des DDR-Zugriffs macht der interne Parallelisierungsgrad $N=2 \times 256$ in den Prozessoreinheiten PCU und SCU es notwendig, dass die nachfolgenden RSU und RMU eine größere Zahl an Seitenergebnissen in einer Zeit sortieren müssen, die nach wie vor von der theoretisch minimalen Anfragenlänge festgelegt ist. Dieses Simulationsmodell wird als "Hardware 2" bzw. ACE_{200MHz} bezeichnet.

4.3 Performance-Messungen

Wie in Abb. 6 und Tabelle 1 gezeigt, spielt die Prozessorgeschwindigkeit eines PC nur eine kleine Rolle in der Verarbeitungsdauer einer Anfrage. Bei Verwendung einer dreimal so schnellen CPU (z. B. PC3 im Vergleich zu PC1), reduziert sich die Suchzeit anscheinend nur auf die Hälfte. Fortschritte in der Festplattentechnologie in den vergangenen 5 Jahren erlauben die Reduzierung der Suchzeit auf ein Fünftel des ursprünglichen Werts. Darüber hinaus kann eine Anfrage um den Faktor 20 beschleunigt werden, wenn die BAM im Hauptspeicher des Systems abgelegt ist, also im RAM anstatt auf der Festplatte. Bemerkenswert in Abb. 6 ist die Tatsache, dass bei langen Suchanfragen (rechts abgebildet) der Hauptspeicher des Rechners als Cache-Speicher der Festplatte fungiert. Der Grund hierfür ist, dass in diesem Fall oft bereits von der Festplatte gelesene Teile der BAM wieder untersucht werden, die dann nicht erneut vom Datenträger geladen werden müssen. Der technologische Fortschritt bei Speicherbausteinen in

Bezug auf Zugriffsfrequenz und Datenrate macht den größten Unterschied in unserer Bewertungsumgebung aus. Wir können nicht nur den Zeitvorteil zwischen den verschiedenen Software-Lösungen erklären, sondern auch zwischen den beiden Hardware-Modellen. Dies beweist, dass die Kombination aus schnellem Speicher und schnellem Prozessor noch kein hochleistungsfähiges System ergibt. Wie in diesem Beitrag gezeigt, kommt es auf die vorteilhafte Konnektivität an, die ein 100-MHz-FPGA mit Standard-SDRAM-Technologie (Hardware 1) mehr als zwanzig Mal schneller als einer der leistungsfähigsten derzeit erhältlichen PCs macht.

Als Zusammenfassung zeigt Tab. 1 die durchschnittliche Beschleunigung der simulierten ACE-Architektur "Hardware 2" gegenüber den gemessenen Testplattformen, von verschiedenen Software-Lösungen, bei denen die BAM im RAM oder auf der Festplatte gespeichert ist, bis hin zur Prototyping-Plattform, die standardmäßige Speicherbausteine verwendet. Bei Verwertung aktueller FPGAs mit

Anw.	Prozessoreinheit	BAM-Speicherort	Beschleunigung
SW 1	PIII 1 GHz	ATA 100 Festplatte	~ 25000
SW 1'	PIII 1 GHz	SDR SDRAM 133	~ 1500
SW 2	Athlon 2 GHz	ATA 100 Festplatte	~ 18000
SW 2'	Athlon 2 GHz	DDR SDRAM 166	~ 800
SW 3	PIV 3 GHz	SATA 150 Festplatte	~ 5000
SW 3'	PIV 3 GHz	DDR SDRAM 200	~ 700
HW 1	XCVE 100 MHz	SDR SDRAM 100	32
HW 2	XC4V 200 MHz	DDR SDRAM 200	Referenz

Tabelle 1: Durchschnittliche Beschleunigung, ermittelt durch Simulation der ACE-Architektur "Hardware 2" als Referenz für verschiedene Testplattformen.

erreichbaren Taktfrequenzen von über 500 MHz sind noch deutlicheren Ergebnisse zu Gunsten der Hardwarelösung zu erwarten. Beispielsweise würde eine Anfrage mit einer Länge von 20 Worten in einer Textdatenbank von 20 GB etwa 20 ms auf einer Plattform vom Typ "Hardware 2" benötigen, während theoretisch dieselbe Anfrage auf einem aktuellen PC in etwa einer Minute bearbeitet werden kann.

5 Zusammenfassung

Unsere Arbeit hat aufgezeigt, wie rechenintensive Datenverarbeitungsalgorithmen unter drastischen Geschwindigkeits- und Flächenbeschränkungen effizient auf elementare Hardware-Strukturen abgebildet werden können. Als die zwei Hauptaspekte zeigen sich die Gesamtskalierbarkeit des Designs, die verschiedene Bitbreiten und Modultiefen entlang des Datenpfades unterstützt, sowie das Erreichen von minimaler Hardware-Fläche bei maximalem Datendurchsatz der globalen Architektur. Dank der Skalierbarkeit des Systems und der modularen Implementierungsmethode zeigten die Simulationsergebnisse einen enormen Geschwindigkeitsgewinn von mehr als vier Größenordnungen für die beschleunigte ACE-basierte Plattform im Vergleich zu einem PC, wo eine recht große BAM nirgends anders als auf der Festplatte gespeichert werden würde. Vom Standpunkt der Hardware aus betrachtet, kann man die hier präsentierte VLSI-Struktur als sehr gut geeignet für die ASIC-Implementierung bezeichnen, unter Berücksichtigung der parallelen Recheneinheit aus PCU und SCU, sowie auch der parallelen Sortiereinheit aus RSU und RMU.

Danksagung

Der Autor möchte sich an dieser Stelle bei Prof. Dr.-Ing. Hans-Jörg Pfeleiderer für seine Kritik und Anregungen, Prof. Pál Ruján und Dr. Gennady Lapir für die fachliche Unterstützung und Diskussionen, und Dipl.-Ing. Oliver Pfänder für die Hilfe bei der Übersetzung bedanken.

Literatur

- [1] R. Baeza-Yates, B. Ribeiro-Neto, "Modern Information Retrieval", Addison Wesley, 1999.
- [2] Z. Baker, V. Prasanna, "Efficient Hardware Data Mining with the Apriori Algorithm on FPGAs", *Proc. IEEE Symp. Field Programmable Custom Computing Machines*, pp. 3-12, 2005.
- [3] Z. Baker, V. Prasanna, "An Architecture for Efficient Hardware Data Mining using Reconfigurable Computing System", *Proc. IEEE Symp. Field Programmable Custom Computing Machines*, pp. 67-75, 2006.
- [4] K. E. Batcher, "Sorting Networks and Their Applications", *Proc. AFIPS Spring Joint Computer Conf.*, Vol. 32, pp. 307-314, 1968.
- [5] S. Berkovich, E. El-Qawasmeh, G. Lapir, "Organization of Near Matching in Bit Attribute Matrix Applied to Associative Access Methods in Information Retrieval", *Proc. IASTED Int. Conf. Applied Informatics*, pp. 62-65, Feb. 1998.
- [6] H.-M. Blüthgen, P. Osterloh, H. Blume, T. G. Noll, "A Hardware Implementation for Approximate Text Search in Multimedia Applications", *Proc. IEEE Int. Conf. Multimedia and Expo*, pp. 1425-1428, 2000.
- [7] H. Blume, H.-M. Blüthgen, T. G. Noll, "Integration von hochperformanten ASICs in reconfigurierbare Systeme zur Bereitstellung zusätzlicher Multimedia-Funktionalitäten", *Proc. 25. MPC Workshop*, pp. 19-28, Feb. 2001.
- [8] H. Kalte, M. Porrmann, U. Rückert, "A Prototyping Platform for Dynamically Reconfigurable System on Chip Designs", *Proc. IEEE Workshop on Heterogeneous Reconfigurable Systems on Chip*, 2002.
- [9] D. E. Knuth, "The Art of Computer Programming, Volume 3: Sorting and Searching", Second Edition, Addison Wesley, 1997.
- [10] G. M. Lapir, "Use of Associative Access Method Information Retrieval Systems", *Proc. Pittsburgh Conf. on Modeling and Simulation*, Vol. 23, No. 2, pp. 951-958, 1992.
- [11] C. Layer, H.-J. Pfeleiderer, C. Heer, "A Scalable Compact Architecture for the Computation of Integer Binary Logarithms Through Linear Approximation", *Proc. IEEE Int. Symp. Circuits and Systems*, Vol. 2, pp. 421-424, 2004.
- [12] C. Layer, H.-J. Pfeleiderer, "A Reconfigurable Recurrent Bitonic Sorting Network for Concurrently Accessible Data", *Proc. Int. Field Programmable Logic Conf.*, LNCS 3203, Springer-Verlag, pp. 648-657, 2004.
- [13] C. Layer, H.-J. Pfeleiderer, "Efficient Hardware Search Engine for Associative Content Retrieval of Long Queries in Huge Multimedia Databases", *Proc. IEEE Int. Conf. Multimedia and Expo*, pp. 1034-1037, 2005.
- [14] C. Layer, H.-J. Pfeleiderer, "Hardware Implementation of an Approximate String Matching Algorithm Using Bit Parallel Processing for Text Information Retrieval Systems", *Proc. IEEE Int. Conf. Signal Processing Systems*, pp. 193-198, 2005.
- [15] J. N. Mitchell, "Computer Multiplication and Division Using Binary Logarithm", *IRE Trans. Electronic Computers*, Vol. EC-11, pp. 512-517, 1962.
- [16] G. Navarro, "A Guided Tour to Approximate String Matching", *ACM Comp. Surveys*, Vol. 33, pp. 31-88, 2001.
- [17] G. Navarro, M. Raffinot, "Flexible Pattern Matching in Strings: Practical On-Line Search Algorithms for Texts and Biological Sequences", Cambridge Univ. Press, 2002.
- [18] R. Sedgewick, "Algorithms", Sec. Edition, Addison Wesley Longman, 1988.
- [19] H. S. Stone, "Parallel Processing with the Perfect Shuffle", *IEEE Trans. Computers*, Vol. C-20, No. 2, pp. 153-161, 1971.

Design of a Low Power PTAT Bandgap Cell

Ji Li, D. Jansen, ASIC Design Center Offenburg

Hochschule Offenburg, Badstraße 24, D-77652 Offenburg

Telefon, 0781/205-274

Abstract: Bandgap principle is introduced in the analog circuits design as reference source on which all relevant currents and voltages dependent. This article describes a bandgap circuit with transistor re-dimension, in 0.35µm MOS technique. The 1.2-v reference voltage, with 50mv variable, achieved for the input voltage between 1.8v and 3.5v, and constant temperature performance is accomplished with 1.46mv variable over the temperature range from -40°C to 100°C. The total current flow is 9µA in the room temperature and with total power dissipation low down to 30 µW. The second part introduces the basic knowledge of bandgap circuits, and in the third part, the low power PTAT bandgap cell design has been presented.

Index Terms: PTAT, CTAT

1. Introduction

Reference circuits are basic elements in many applications ranging from purely analog, mixed-signal to digital design. The demand for low voltage references is especially apparent in mobile battery operated products such as cellular phones, pagers, camera recorders, and laptops. Consequently, low voltage and low quiescent current flow are intrinsic and required characteristics conducive toward increased battery efficiency and longevity. Low voltage operation is also a consequence of process technology. This is because isolation barriers decrease as the component densities per unit area increase, thereby exhibiting lower breakdown voltage. As a result, maintaining high dynamic range while operating at reduced power supply voltages demands that the reference circuitry be more accurate [4].

Bandgap reference circuit, based on bipolar integrated circuits is one of the traditional voltage reference circuits. Others like, circuits based on threshold voltage (V_{th}) differences in MOS transistors, MOS transistor circuits in which threshold voltage and mobility temperature dependencies are compensated for, current mode as well as beta multiplier, are well implemented in many applications. Recent state of the art system LSIs, however, demand the use of voltage

references which operate at low voltages, are robust to process fluctuations, exhibit low electrical current consumption, can be integrated with standard CMOS technologies, and allow flexible voltage adjustment [5]. Here, low power PTAT bandgap cell, based on bipolar integrated circuits, has been studied.

2. Theory of Bandgap Reference

The principle of the bandgap voltage reference is to balance the negative temperature coefficient of a pn junction with the positive temperature coefficient of the thermal voltage, $V_t = kT/q$

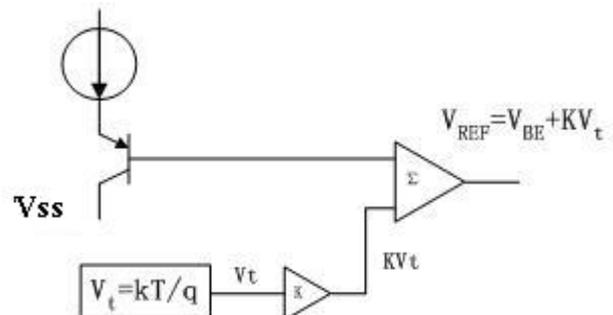


Fig. 1. General principle of the bandgap reference

A voltage V_{BE} is generated from a pn-junction diode having a temperature coefficient of approximately $-2.2\text{mV}/^\circ\text{C}$ at room temperature. A thermal voltage $V_t (=kT/q)$, which is proportional to absolute temperature (PTAT), is generated. The circuit also has a temperature coefficient of $+0.085\text{mV}/^\circ\text{C}$ at room temperature. If the V_t voltage is multiplied by a constant K and summed with the V_{BE} voltage, then the output voltage is given as [2]

$$V_{REF} = V_{BE} + KV_t \tag{1}$$

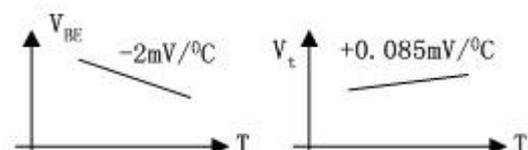


Fig. 2. Temperature coefficient of base-emitter voltage and thermal voltage

Fig. 2. shows very simple generation of reference voltages (Operating point adjustment with resistor).

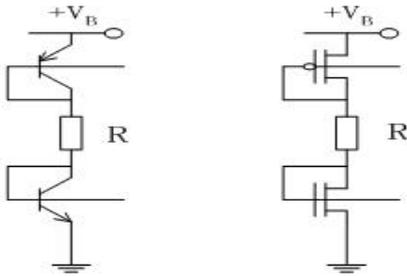


Fig. 3. Simple generation of reference voltages

The circuit produces near the lower and upper supply voltage lines a voltage drop across an 'active resistance', which also can be converted via a current mirror into a current. Fig. 3. is the simple generation of reference voltages and in Fig. 4. it shows a traditional bandgap reference circuit. The reference voltage depends on temperature, resistance, and operating voltage. For the realization of low currents one needs large resistance values and thus much chip area [1].

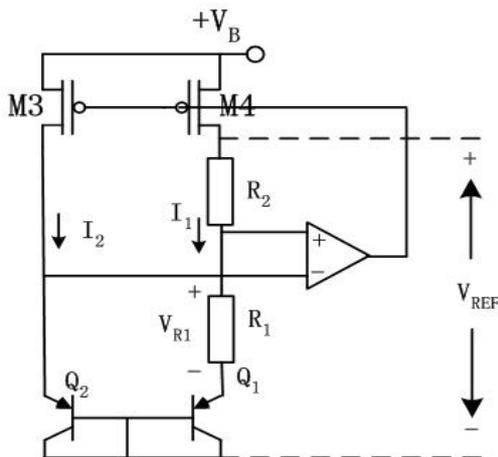


Fig. 4. A conventional bandgap reference

$$V_{REF} = V_{gap} + \frac{kT}{q}(\gamma - 1) \left[1 - \ln \left(\frac{T}{T_0} \right) \right] \quad (2)$$

V_{gap} is the extrapolated bandgap voltage of a semiconductor, γ is the mobility temperature exponent of the charge carriers in a bipolar transistor and T_0 is the reference temperature. The term can be realized by chosen proper value of the resistor ratio R_1/R_2 , satisfying the equation (3)

$$\frac{R_1}{R_2} = \frac{1}{\ln n} \left[(\gamma - 1) - \ln \left(\frac{U_T \ln \frac{A_{E1}}{A_{E2}}}{C_1 \cdot R_1 \cdot T \cdot T_0^{(\gamma-1)}} \right) \right] \quad (3)$$

A_{E1}/A_{E2} is the ratio of the emitter areas of bipolar transistors Q_1 and Q_2 , U_T is the thermal voltage kT/q and C_1 is a process constant.

The output voltage, at the reference temperature T_0 , depends on the process spread, the offset voltage of the op-amp and matching of the resistors [2].

Sensitivity and fractional temperature coefficient are two basic characteristics of the dependence of a reference on power supply and temperature. The sensitivity of V_{REF} to changes in a power supply V_S is given as

$$S_{V_S}^{V_{REF}} = \lim_{\Delta V_{xx} \rightarrow 0} \frac{\Delta V_{REF} / V_{REF}}{\Delta V_S / V_S} = \frac{V_S}{V_{REF}} \left(\frac{\Delta V_{REF}}{\Delta V_S} \right) \quad (4)$$

The above equation show how the sensitivity can be calculated. Furthermore, once the sensitivity is known, it can be applied using the following form.

$$\frac{\Delta V_{REF}}{V_{REF}} \approx (S_{V_S}^{V_{REF}}) \left(\frac{\Delta V_S}{V_S} \right) \quad (5)$$

This formulation depicts: if the sensitivity of V_{REF} with respect to V_S is unity then a 10% a change in V_S will cause a 10% change in V_{REF} .

The temperature coefficients, which are generally expressed in terms of ppm/ $^{\circ}C$, are defined by

$$TC = \left(\frac{1}{x} \right) \left(\frac{dx}{dT} \right) 10^6 \text{ ppm}/^{\circ}C \quad (6)$$

The value of TC, which can be either positive or negative, is determined by the material properties and is often quite small. The absolute values of TC are often less than 1000 ppm/ $^{\circ}C$. Unfortunately, since integrated circuits are often expected to operate over a relatively wide temperature range (0 to 70 $^{\circ}C$ commercial or -55 to 125 $^{\circ}C$ military) the effects of the TC can be significant.

The $S_T^{V_{REF}}$ could be used as a measure of the dependence of the reference on temperature by replacing V_S with T. The fractional temperature coefficient (TC_F) of V_{REF} is defined as

$$TC_F(V_{REF}) = \frac{1}{V_{REF}} \frac{\partial V_{REF}}{\partial T} = \frac{1}{T} S_T^{V_{REF}} \quad (7)$$

The units of TC_F are expressed in terms of parts per million per $^{\circ}C$ (ppm/ $^{\circ}C$). References with a TC_F of less than 50 ppm/ $^{\circ}C$ are considered to be stable with respect to temperature [3].

3. Circuit Realization

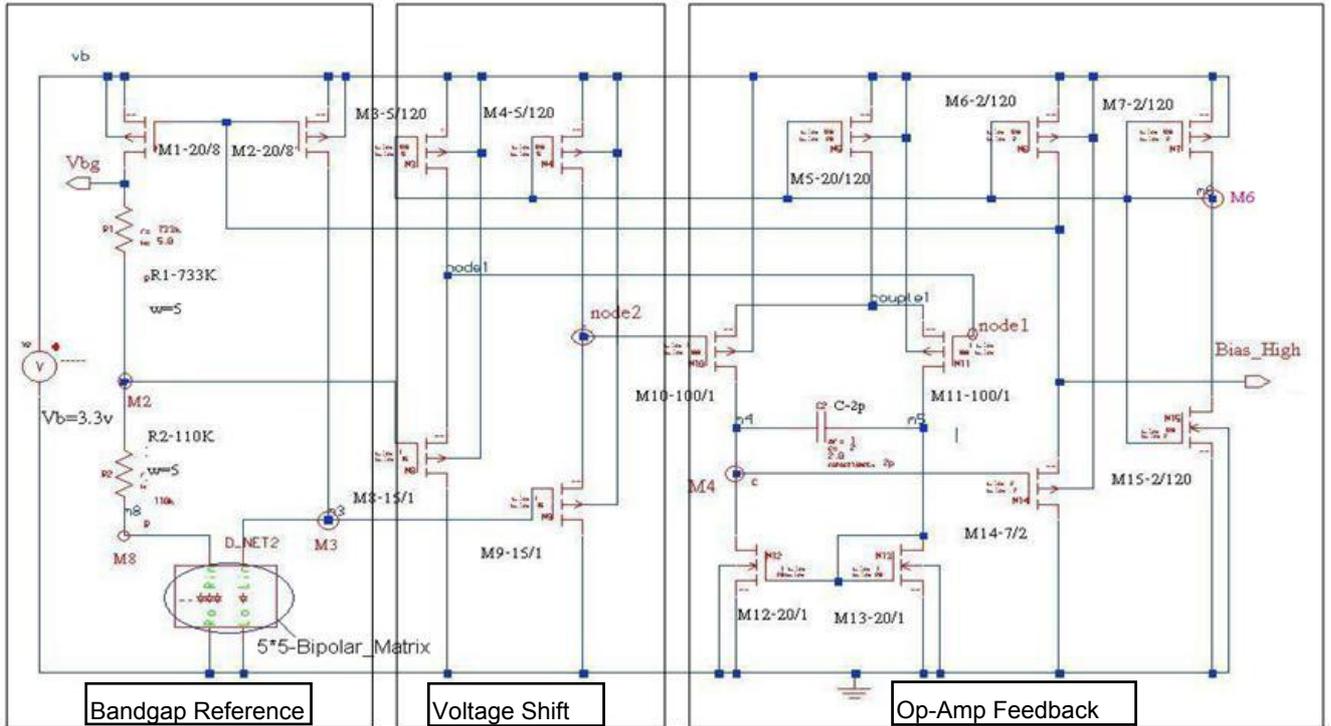


Fig. 5. Low Power PTAT Bandgap Cell Realization

The bandgap reference cell, described in this article, shows a comparatively low total power consumption with 3.3v supply voltage. The general principle relies on the traditional bandgap reference, refer to Fig. 4. With a suitable resistor ratio, R_1/R_2 , a PTAT bandgap cell, in fine condition, is achieved, and the result will be discussed later. Fig. 5. presents the overall bandgap cell.

The drain current, I_D , in the saturation region, is defined as follow

$$I_D = K \cdot (V_{GS} - V_{TH})^2 \cdot (1 + \lambda \cdot V_{DS}) \quad (8)$$

Where $K = C_{ox} \cdot \mu(T) \cdot W / L$ is the transconductance parameter (in A/V^2), λ is the channel length modulation parameter ($1/V$), and $C_{ox} = \epsilon / t_{ox}$ is the specific gate capacitance [1]. So, by utilizing a small channel width or big channel length, the current flow through each branch can be comparatively low. In this case, the drain current of M1 is around 0.8uA. The bandgap cell is composed of three parts: Bandgap reference, Voltage shift and operational amplifier-feedback part. See Fig. 5.

The bandgap cell generates the necessary PTAT reference voltage by employing the bipolar network, which is made up of bipolar 5×5 matrix, connecting

24 in parallel vs. one in solitude, Fig. 6. The current flow difference between two branches is shown in the simulation, see Fig. 7.

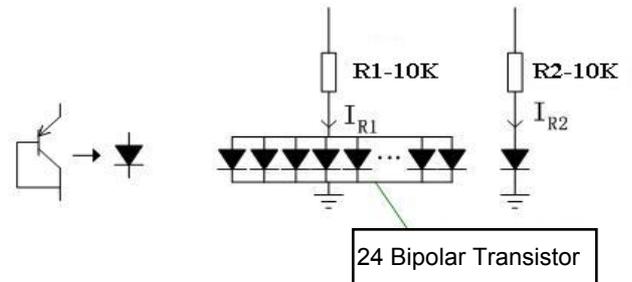


Fig. 6. Bipolar network

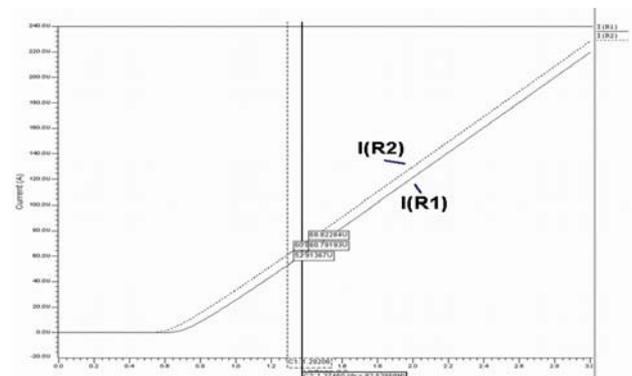


Fig. 7. Characteristics of Bipolar network

The drain currents of M1 and M2 have a negative temperature coefficient, and in the next stage, voltage shift, the p-MOS transistors, M8 and M9, perform a positive temperature characteristic, which compensates the PTAT reference voltage by choosing suitable resistor ratio of R_1/R_2 , and all together contribute to a stable output with little dependent on the temperature. The compensation attribute is depicted in Fig. 8.

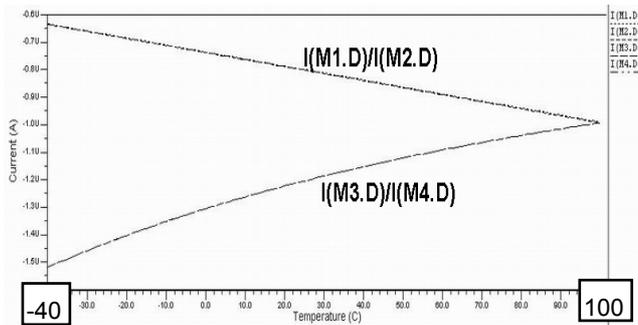


Fig. 8. Current compensation over temperature(-40°C to 100°C)

The operational amplifier used in this circuit is similar like the traditional op-amp, instead, by using a capacitor connected between positive and negative output, for the stability of the output, considering the low frequency, around 100kHz.

The feedback, leading from drain of M6, controls the current flow of M1, and compensates the conversely proportional to absolute temperature (CTAT) voltages. M6 and M14 form the output stage, and generate the bias-high output voltage. M7 and M15, current mirror, provides bias voltage for p-MOS transistor, M3 to M6. Considering the voltage dependency, the drain current of M15, n-MOS transistor, which is controlled by the supply voltage, obviously goes up along with the supply voltage increasing, while the current, flowing through M6 and M14 and controlled by the output of the operational amplifier, goes down in the mean while, which balances the reference voltage over the supply voltage. Due to the high impedance, to use this bandgap reference cell, an external op-amp, connected as voltage follower, is needed to improve the driving capability of the bandgap reference. See Fig.9.

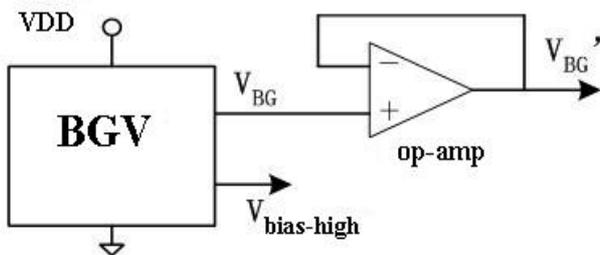


Fig. 9. Application of BG Cell

The bias_high, Vbias_high, derived from the output stage, M6 and M14, of BG cell, can be applied as bias_high input for the further applications. The performance of the bias_high is shown in the next part. More detail simulation results of transient analysis and DC performance are as well as displayed.

4. Simulation Results

Transient analysis:

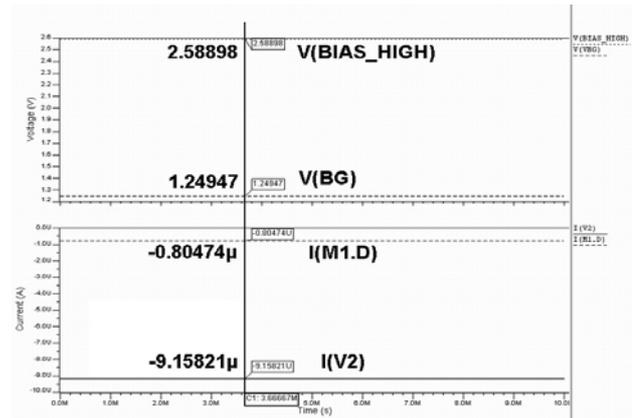


Fig. 10. Transient analysis (Vbg, Vbias_high, Iv2)

The transient analysis shows the bandgap voltage, $V_{bg}=1.249V$, the bias high, $V_{bias_high}=2.589V$, and the total current flow, $I_{v2}=9.158\mu A$ at the room temperature (300K). The supply voltage is 3.3v, and the total power consumption is calculated as

$$I_{V2} \times V_b = 9.158\mu A \times 3.3V = 30.22\mu W$$

Temperature dependency:

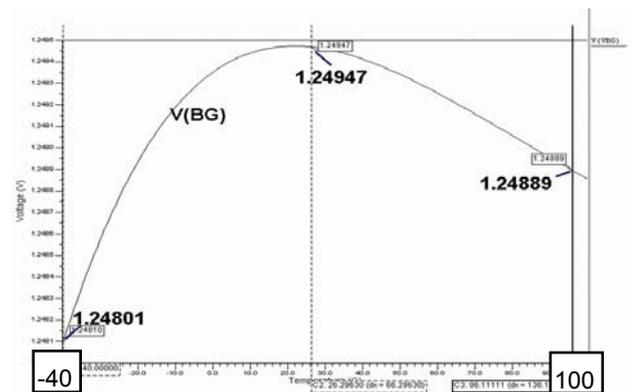


Fig. 11. Temperature performance of bandgap ref

In Fig. 11. the output of bandgap reference voltage shows little dependent on temperature, with a variable of 1.46mV over temperature range from -40°C to 100°C, as well, Fig. 12. is the temperature performance of bias_high output, which also shows a good temperature independent performance with tiny difference over the same temperature range.

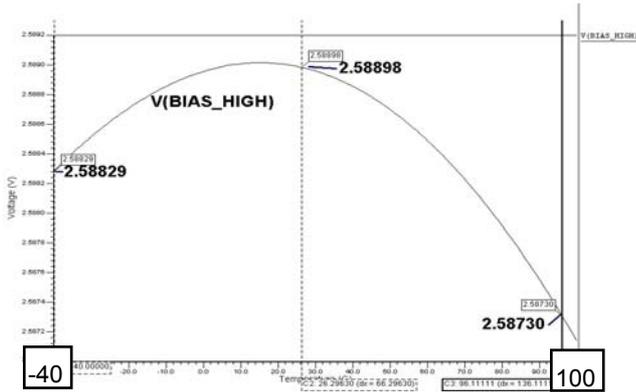


Fig. 12. Temperature performance of Vbias_high

Voltage dependency:

Fig. 13. is the sweep analysis, with supply voltage, Vb, altering from 0-5V. The bandgap voltage shows 60mV variable over the supply voltage from 1.8V to 3.5V. And the low part is the currents, flowing through M1, and M2, which also perform a stable characteristic over supply voltage.

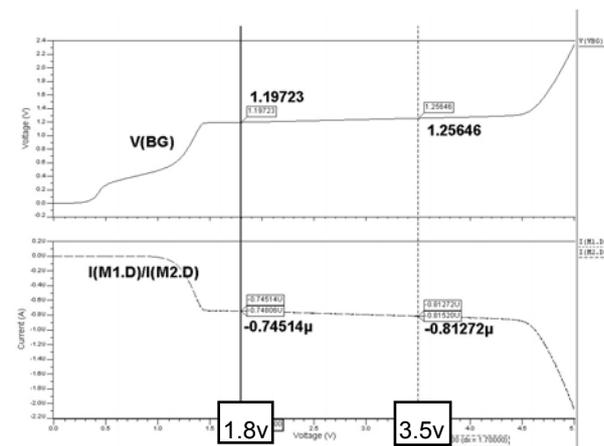


Fig. 13. Sweep analysis(Vb,supply voltage,0-5v)

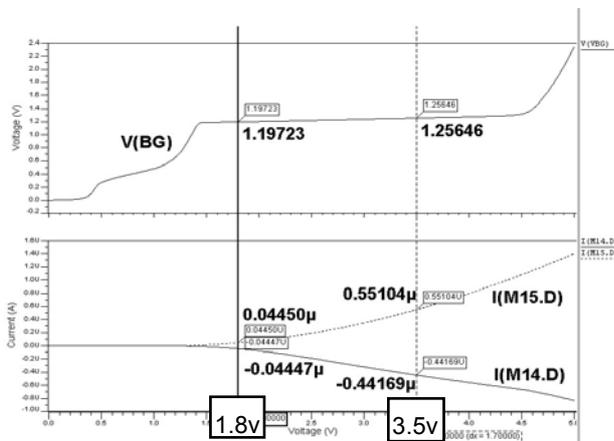


Fig. 14. Sweep analysis (Current compensation)

The lower part of Fig. 14. is the drain current of M14 and M15, which gives an intuition depiction of how the

circuit compensating over the variable supply voltage. The current, flowing through M15, n-MOS transistor, will go up with the supply voltage increasing, at the same time, because of the self-compensation of the differential amplifier, the current flow through M14, p-MOS transistor, will decrease. Due to this behaviour, the bandgap reference gets a stable output over the supply voltage.

Fig. 15. is the Layout of the overall PTAT Bandgap Cell and the bipolar transistor matrix.

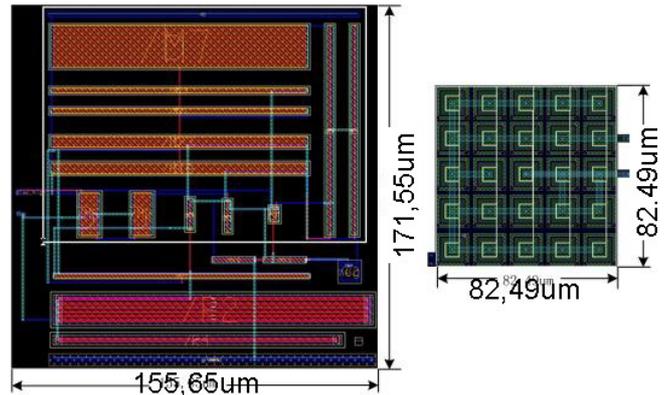


Fig. 15. Layout of PTAT Bandgap Cell

5. Conclusion

A PTAT Bandgap Cell of curvature-corrected and low power consumption was presented in this article. By re-dimension of the MOS transistor, the total current flow goes down to 10μA and the total power consumption is around 30uW. The low cost of power well fulfils requirements of nowadays large variety applications, from mobile communication, medicine to education and military, etc. The circuit possesses a favourable temperature characteristic, with a variable of 1.46mV over a wide temperature range from -40°C to 100°C.

Table 1. concludes the performance of the bandgap reference cell and the comparison with some of other previous designs.

Table 1. Performance of the Bandgap Reference Cell

	This work	Ref.[4]	Ref.[8]
Technology(μm)	0.35-(CMOS)	2-(MOSIS)	0.6-(CMOS)
Sup. Current(μA)	9	15	23
Ref. Voltage(v)	1.24947	0.595	1.14205
Chip Area(mm ²)	0.0335	0.223	0.057
TC (ppm/°C)	8.3	6.5	6.1

6. Reference

- [1] D. Jansen, The Electronic Design Automation Handbook, Kluwer Academic Publishers, ISBN: 978-1-4020-7502-5
- [2] Phillip E. Allen, Douglas R. Holberg, CMOS Analog Circuit Design, OXFORD University Press, ISBN: 0-19-511644-5
- [3] Randall L. Geiger, Phillip E. Allen, Noel R. Strader, VLSI Design Techniques for Analog and Digital Circuits, McGraw-Hill Publishing Company, ISBN 0-07-100728-8
- [4] Gabriel A. Rincon-Mora and Phillip E. Allen, "A 1.1-V Current-Mode and Piecewise-Linear Curvature-Current Bandgap reference", IEEE Journal of Solid-State Circuits, Vol. 33, No. 10, October 1998
- [5] H Watanabe, S Ando, H Aota, M Dainin, Y Chun, K Taniguchi, "CMOS Voltage Reference Based on Gate Work Function Differences in Poly-Si Controlled by Conductivity Type and Impurity Concentration", IEEE Journal of Solid-State Circuits, Vol. 38, No. 6, June 2003
- [6] S.S. Taylor, "A GaAs MESFET Schottky diode barrier height reference circuit," in *ISSCC Dig. Tech. Papers*, Feb. 1997, pp. 360-361.
- [7] R. A. Blauschild, P. Tucci, R. S. Muller, and R. G. Meyer, "A new NMOS temperature-stable voltage reference," IEEE J. Solid State Circuits, vol. SC-13, pp. 767-773, Dec. 1978.
- [8] Ka Nang Leung, Philip K.T.Mok, Chi Yat Leung, "A 2-V 23uA 5.3-ppm/OC Curvature-Compensated CMOS Bandgap Voltage Reference", IEEE Journal of SS Circuits, Vol. 38. No. 3, March 2003

Lohnt sich der Einsatz verlustleistungsoptimierender Buskodierverfahren?

Prof. Dr.-Ing. Frank Kesel

Dipl.-Ing. (FH) Manuel Gaiser

Hochschule Pforzheim, Tiefenbronner Str. 65, 75175 Pforzheim

Tel.: 07231/28-6065

Fax: 07231/28-6060

E-Mail: {frank.kesel,manuel.gaiser}@hs-pforzheim.de

In den letzten Jahren wurde eine Vielzahl von Buskodierverfahren veröffentlicht, mit deren Hilfe ein Beitrag zur Minimierung der gesamten Verlustleistungsaufnahme elektronischer Systeme geleistet werden soll. Ursprünglich für Off-Chip-Busse bestimmt, sollen sich neuere Verfahren beispielsweise durch Berücksichtigung von Koppeleffekten auch für On-Chip-Bussysteme eignen.

Durch die Buskodierung wird die hohe Schaltaktivität von den Busleitungen hinein in die Buskodierschaltungen mit vergleichsweise niedrigen internen Verdrahtungskapazitäten verlagert. Dem Gewinn in Form einer Verringerung der durch Lastkapazitäten von Busleitungen verursachten Verlustleistungsaufnahme steht also ein Verlust in Form eines höheren Stromverbrauchs durch die hinzugefügten Kodier- und Dekodierschaltungen gegenüber.

Neben einer Literaturrecherche zu den veröffentlichten Buskodierverfahren bildet die Untersuchung dieser zuletzt genannten Gewinn- und Verlustrechnung sowie deren Abhängigkeiten von bestimmten Anwendungsgebieten und Technologieparametern den Schwerpunkt dieser Arbeit. Praktische Anwendung finden die gewonnenen Erkenntnisse in einer MATLAB-Toolbox zur Unterstützung der Entscheidungsfindung bei der Auswahl von Buskodierverfahren.

1. Einleitung

Wesentlichen Anteil an der Verlustleistungsaufnahme integrierter CMOS-Schaltungen hat der dynamische Stromverbrauch durch Auf- und Entladevorgänge. Hier liegt der Ansatz für die Entwicklung einer Vielzahl von Buskodierverfahren, mit deren Hilfe durch schaltungstechnische Maßnahmen auf Architekturebene ein Beitrag zur Minimierung der gesamten Verlustleistungsaufnahme elektronischer Systeme geleistet werden soll.

Für die durch das Auf- und Entladen von Lastkapazitäten verursachte dynamische Verlustleistung gilt folgender Zusammenhang:

$$P_{dyn,Load} = \frac{1}{2} C_L \cdot U_{dd}^2 \cdot f \cdot a \quad (1-1)$$

Neben der Lastkapazität C_L einer Verbindungsleitung ist die Versorgungsspannung U_{dd} ein an die eingesetzte Technologie gekoppelter Parameter und damit auf Architekturebene nicht beeinflussbar. Die Taktfrequenz f wird im Wesentlichen von der erforderlichen Datenrate bestimmt. Lediglich die Aktivitätsrate a kann mit geeigneten Kodierverfahren reduziert werden.

Der Einbau des Buskodierers erfolgt direkt in, bzw. unmittelbar hinter der Datenquelle, der Einbau des Dekodierers in räumlicher Nähe zur Datensenke.

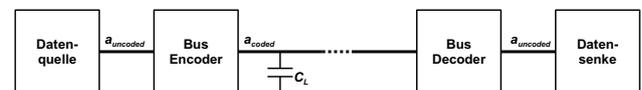


Abbildung 1.1: Bus mit Kodier- und Dekodierschaltungen

2. Buskodierverfahren

2.1. Kennzahlen

Die Kodiereffizienz Eff_C eines Buskodierverfahrens ist gegeben als

$$Eff_C = 1 - \frac{a_{coded}}{a_{uncoded}} = 1 - \frac{m_{coded}}{m_{uncoded}} \quad (2-1)$$

Eff_C ist also gleich Null, wenn die Aktivitätsraten des kodierten (a_{coded}) und des unkodierten ($a_{uncoded}$) Busses identisch sind. Ist der Quotient kleiner als 1, also $a_{coded} < a_{uncoded}$, so ist Eff_C positiv. Derselbe Zusammenhang besteht auch bezüglich der Anzahl der Signaländerungen m_{coded} , bzw. $m_{uncoded}$.

Für die Energieeffizienz Eff_E gilt entsprechend

$$Eff_E = 1 - \frac{E_{coded}}{E_{uncoded}} \quad (2-2)$$

Die Energieeffizienz nimmt also eine Betrachtung der Energiebilanz des gesamten kodierten Bussystems (E_{coded}) in Relation zum Energieverbrauch des unkodierten ($E_{uncoded}$) Bussystems vor. Auch hier gilt, dass Eff_E dann positiv wird, wenn $E_{coded} < E_{uncoded}$.

Durch den Einsatz eines Buskodierverfahrens soll die Anzahl der Signaländerungen auf Busleitungen reduziert werden. Wegen des linearen Zusammenhangs zwischen der Leitungskapazität und der Länge einer Leitung wird also umso mehr Energie eingespart, je länger der Bus ist. Dem gegenüber steht der Energieverbrauch der Kodierschaltungen. Ist die eingesparte Energie E_S gleich dem Energieverbrauch der Kodierschaltungen E_{Codec} , so gilt:

$$E_{Codec} = E_S = E_{Bus,uncoded} - E_{Bus,coded} \quad (2-3)$$

Setzt man für E_{Bus} die entsprechenden formelmäßigen Zusammenhänge ein und vernachlässigt den Einfluss von Repeaterschaltungen und Koppelkapazitäten, so ergibt sich

$$E_{Codec} = 0.5 U^2 L C_s m_{uncoded} Eff_C \quad (2-4)$$

mit U = Versorgungsspannung, L = Länge der Busleitung, C_s = spezifische Platten- und Randkapazität einer Verbindungsleitung, $m_{uncoded}$ = Anzahl Signaländerungen des unkodierten Busses und Eff_C = Kodiereffizienz der Buskodierschaltung.

Damit gilt für die Buslänge, ab der sich der Einsatz eines Buskodierverfahrens lohnt (*effektive Buslänge* L_{eff}):

$$L_{eff} = \frac{E_{Codec}}{0.5 U^2 C_s m_{uncoded} Eff_C} \quad (2-5)$$

2.2. Klassifikation

Die wesentlichen Merkmale zur Klassifizierung von Buskodierverfahren sind:

- Die *Methodik* (z.B. Prädiktions-, Referenz-, Multiplex- oder tabellenbasierte Verfahren)
- Das *Anwendungsgebiet* (Datenbusse, Adressbusse)
- Die *Technologie* (Off-Chip / On-Chip)
- Die Unterteilung in *statische* und *adaptive Verfahren*
- Das benötigte Vorwissen über die zu kodierenden Daten zum Entwurf der

Kodierschaltung (*Daten a priori bekannt*, bzw. *nicht bekannt*).

Für weitere Untersuchungen wurden aus der Vielzahl der Publikationen zwei repräsentative Verfahren ausgewählt:

- Das *Businvert-Verfahren* (BI) [8] invertiert die Daten auf dem Bus, wenn die Hammingdistanz zwischen dem aktuellen Zustand des Busses und dem zu kodierenden Datenwort größer als die Hälfte der Busbreite ist. Bei BI handelt es sich um ein statisches Multiplex-Verfahren. Für die Signalisierung des aktuellen Schaltungszustandes (invertiert, bzw. nicht invertiert) wird eine zusätzliche Signalleitung benötigt. Das BI-Verfahren eignet sich für die Kodierung unkorrelierter Datenströme, also z.B. für Datenbusse.
- Das *Zero-Transition-Verfahren* (T0) [9] lässt den Zustand des Busses unverändert, wenn zwei aufeinander folgende Datenwörter eine konstante Differenz aufweisen, andernfalls wird unkodiert übertragen. T0 ist ein statisches Prädiktionsverfahren, das wie BI eine zusätzliche Signalleitung benötigt. Es eignet sich für die Kodierung von Adressbussen.

3. Modellierung des Energieverbrauchs

3.1. Überblick

Die Modellierung des Energieverbrauchs auf Systemebene setzt zunächst eine Betrachtung der relevanten Effekte auf Schaltkreisebene voraus. Zum Energieverbrauch in integrierten CMOS-Schaltungen tragen drei Komponenten bei:

- Das Aufladen von Lastkapazitäten, im Einzelnen:
 - Platten- und Randkapazität der Signalleitungen gegen das Substrat
 - Koppelkapazitäten zwischen benachbarten Signalleitungen
 - Gate-, Source- und Drainkapazitäten der MOS-Transistoren
- Querströme im Umschaltzeitpunkt, im Wesentlichen abhängig von:
 - der Anstiegs- bzw. Abfallzeit der Eingangsflanke,
 - der Lastkapazität,
 - dem Verstärkungsfaktor und der Schwellspannung der MOS-Transistoren
 - sowie der Betriebsspannung.

- Statische Leckströme, verursacht durch:
 - Subthreshold-Ströme im Kanal,
 - Leckströme zwischen Diffusionsgebieten und Substrat,
 - Gate-Tunnelströme

3.2. Abstraktionsebenen und Abgleich

Die in diesem Kapitel beschriebenen Modelle ermöglichen die Untersuchung des *Energieverbrauchs* einzelner Datentransfers, fokussieren also auf die Energieentnahme aus der Spannungsquelle. Modelliert wird auf *Systemebene* mit Hilfe von *MATLAB*-Skripten, d.h. auf der Ebene von Bussen und Kodierschaltungen. Als Grundlage für die Modelle dienen Simulationsergebnisse von Synopsys *PrimePower*, eines Powersimulators auf *Gatterebene*. Die Verifikation der Simulationsergebnisse erfolgte mit *ELDO* anhand kleinerer Kodierschaltungen auf *Transistorebene*.

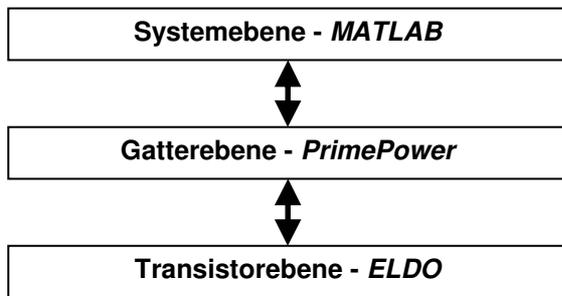


Abbildung 3.1: Abstraktionsebenen

Zwischen den Simulationsergebnissen der einzelnen Abstraktionsebenen zeigt sich für die untersuchten Technologien (CMOS 0.5 µm / 0.35 µm) und Kodierschaltungen eine gute Übereinstimmung:

Zelle / Schaltung	CMOS 0.5 µm	CMOS 0.35 µm
BI-kodiertes Bussystem (4 Bit)	6.3 %	8.8 %

Tabelle 3.1: Abweichungen PrimePower–ELDO

Zelle / Schaltung	CMOS 0.5 µm	CMOS 0.35 µm
BI-kodiertes Bussystem (16 Bit)	-4.0 %	5.0 %
T0-kodiertes Bussystem (32 Bit)	-5.7 %	-20.1 %

Tabelle 3.2: Abweichungen MATLAB–PrimePower

Die vergleichsweise hohe Abweichung des T0-Systemmodells von den Primepower-Ergebnissen

wird durch das im ersten Ansatz noch relativ einfache Verfahren bei der Erstellung des T0-Systemmodells verursacht.

3.3. Systemkomponenten

Zur Modellierung des Energieverbrauchs von *Repeatern* wurde der Energieverbrauch pro Signaländerung unter standardisierten Bedingungen ermittelt.

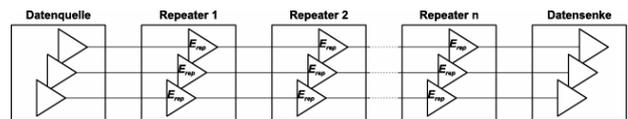


Abbildung 3.2: Modell der Repeater auf dem Bus

Ist E_{rep} der Energieverbrauch eines Repeaters für einen Umschaltvorgang, so summiert sich der gesamte interne Energieverbrauch $E_{rep,ges}$ aller Repeater für m_{rep} Umschaltvorgänge zu:

$$E_{rep,ges} = m_{rep} \cdot n \cdot E_{rep} \quad (3-1)$$

Bei *Kodier- und Dekodierschaltungen* ist der Energieverbrauch im Gegensatz zu den Repeatern abhängig u.a. vom aktuellen internen Schaltungszustand und dem Zustand der Busleitungen. Da die Berücksichtigung aller möglichen Schaltungszustände bei der Modellbildung zu aufwändig wäre, wurde eine Vorgehensweise gewählt, bei der zunächst der Energieverbrauch pro Transfer für ein breites Spektrum von Schaltungszuständen bestimmt wird. Anschließend erfolgt eine Reduktion der so gewonnenen Daten auf einen relevanten Satz von Eingangsparametern. Beim *BI-Verfahren* sind dies die Hamming-Distanz zwischen Ein- und Ausgang, sowie der Zustand der INV-Leitung.

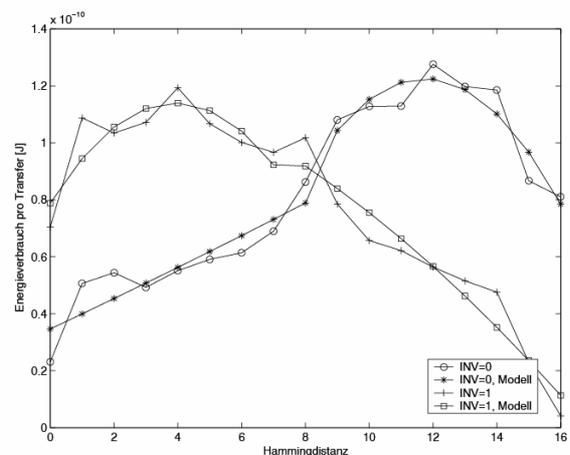


Abbildung 3.3: Energieverbrauch pro Transfer (BI)

Da der Energieverbrauch des *T0-Kodierers* für die meisten Stimuli innerhalb eines begrenzten Bereichs

liegt, wurde in einem ersten Ansatz nur zwischen der Übertragung sequentieller und nicht sequentieller Daten unterschieden und für jeden dieser beiden Fälle eine konstanter Energieverbrauch pro Transfer angenommen.

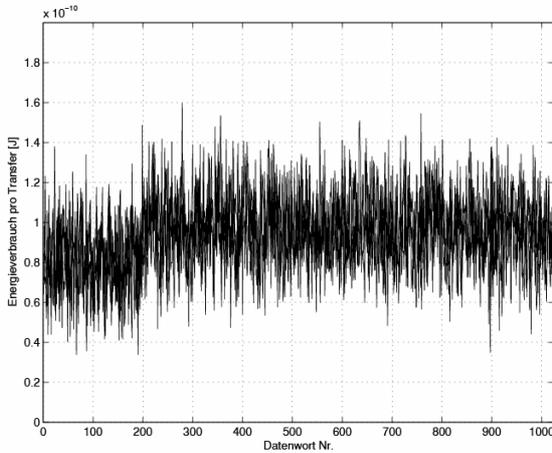


Abbildung 3.4: Energieverbrauch pro Transfer (T0)

Bei der Modellierung von Busleitungen ist lediglich das Aufladen von Lastkapazitäten und das Umladen von Koppelkapazitäten zwischen benachbarten Signalleitungen zu berücksichtigen. Allerdings beschreibt das entwickelte Modell nur Koppelwirkungen zwischen direkt benachbarten Busleitungen. Zur Bestimmung der Parameter für die Modellbildung wurden die erforderlichen Kapazitätsbeläge aus den Design Rule Manuals der verfügbaren ASIC-Technologien entnommen. Entsprechendes gilt für die Eingangskapazitäten der Repeater und Datensenken. Der Einfachheit halber wurden nur typische Werte verwendet, bzw. bei den Koppelkapazitäten die durchschnittlichen Werte der Kapazitätsbeläge über alle Kombinationen von Leiterbahnen auf allen Verdrahtungsebenen berechnet.

4. Anwendung

Für die verfügbaren Technologien (CMOS 0,5 µm / 0,35 µm) wurde anhand der vorgestellten Modelle untersucht, ob sich der Einsatz verlustleistungsoptimierender Buskodierverfahren lohnt.

Das nachfolgende Beispiel eines 16 Bit breiten BI-kodierten Busses macht zunächst deutlich, wie wichtig die Betrachtung der Energieeffizienz bei der Auswahl der optimalen Partitionierung (d.h. der Unterteilung des Busses in kleinere, unabhängig voneinander kodierte Teilbusse) ist. Als Stimulus wurde dabei ein synthetisch erzeugter Datensatz (*effcmax*) verwendet, der die Eigenschaften des BI-Verfahrens optimal ausnutzt, auch im Hinblick auf die Reduzierung von Koppelwirkungen.

Betrachtet man ausschließlich die Kodiereffizienz, so ist ein BI-Kodierer mit 16 Bit Breite optimal:

Partitionierung	Kodiereffizienz
1 x 16	0.938
2 x 8	0.875
4 x 4	0.750
8 x 2	0.500

Tabelle 4.1: Kodiereffizienz für Datensatz *effcmax*

Dagegen macht der Verlauf der Energieeffizienz über der Buslänge deutlich, dass für kleine Buslängen die 8x2-Partitionierung optimal wäre. Die dargestellten Ergebnisse beziehen sich auf die 0,35 µm-Technologie, der Anteil parallel verlaufender Busleitungen wurde mit 10 % angenommen.

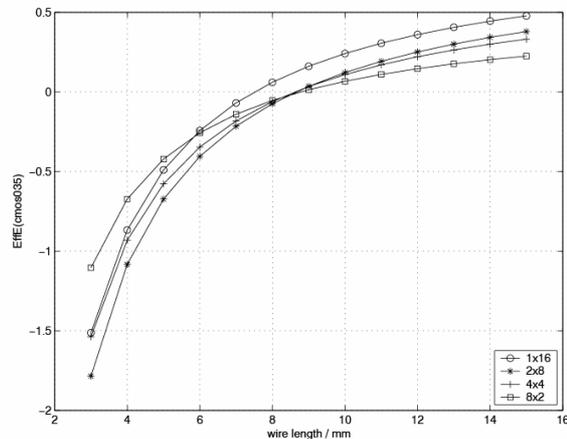


Abbildung 4.1: Energieeffizienz (*effcmax*)

Offensichtlich ist der interne Energieverbrauch der 8x2-Partitionierung geringer, als derjenige eines 16 Bit breiten Kodierers. Für größere Buslängen kommt dagegen die fast doppelt so große Kodiereffizienz des 16-Bit-Kodierers zur Auswirkung.

Die kürzeste effektive Buslänge liegt im dargestellten Beispiel bei ca. 7,5 mm für die 1x16-Partitionierung (Energieeffizienz = 0, d.h. aufgewendete Energie = eingesparte Energie). Dieses Bild verändert sich auch nicht wesentlich, wenn der Anteil parallel verlaufender Busleitungen auf 50 % gesetzt und der Abstand der Repeater verkürzt wird. Die kürzeste effektive Buslänge liegt dann für die 0,5 µm-Technologie bei ca. 4 mm, für die 0,35 µm-Technologie bei ca. 4,5 mm. Diese Längen liegen nur bedingt im Bereich realistischer Werte und werfen zudem die Frage auf, ob ein positiver Effekt durch die Technologiekalibrierung zu erwarten ist.

In [1] werden zwei Skalierungsfaktoren beschrieben:

- Technologie-Skalierungsfaktor S (bezieht sich auf die minimale Strukturgröße, -13% p.a.)
- Skalierungsfaktor der globalen Verbindungsleitungen S_L (proportional zum Skalierungsfaktor der Chipgrößen, +6% p.a.)

Für die einzelnen Komponenten des Gesamtenergieverbrauchs ergeben sich somit folgende Zusammenhänge:

Verdrahtungskapazitäten (global)

Für die Buskodierung sind hochkapazitive, also globale Verbindungsleitungen von Interesse. Die Leitungslänge L skaliert mit S_L , die Leitungsbreite W und die Dicke t des Dielektrikums mit S . Für die Platten- und Randkapazitäten $C_{a,f}$ besteht der Zusammenhang:

$$C_{a,f} \sim \frac{WL}{t} \rightarrow \text{Zunahme proportional zu } S_L$$

Verdrahtungskapazitäten (lokal)

Für die Kapazitäten lokaler und globaler Verdrahtungsleitungen gilt der gleiche Zusammenhang wie bei den globalen Verbindungsleitungen, jedoch skalieren die Längen der lokalen Verdrahtung mit S .
 → Abnahme proportional zu S

Koppelkapazitäten

Nach [4] ist für kleinere Technologien ein starker Anstieg der Koppelkapazitäten zu erwarten. Abbildung 4.2 wurde der genannten Veröffentlichung entnommen und stellt die Entwicklung des Quotienten aus Koppel- und Verdrahtungskapazitäten über der Technologieentwicklung dar.
 → exponentieller Anstieg mit S

Gate-Kapazitäten

Gatekapazitäten sind proportional zu Transistorlänge- und Weite und zur spezifischen Oxidkapazität. Damit ergibt sich:

$$C_{gate} \sim C_{ox}WL \sim \frac{WL}{t_{ox}}$$

→ Abnahme proportional zu S

Leckströme

Leckströme sind relevant für Technologien ≤ 250 nm
 → starke Zunahme erwartet

Querströme

Querströme sind proportional zum Sättigungsstrom, für den gilt:

$$I_{sat} \sim C_{ox}WU \sim \frac{WU}{t_{ox}}$$

→ Skalierung proportional zur Versorgungsspannung U

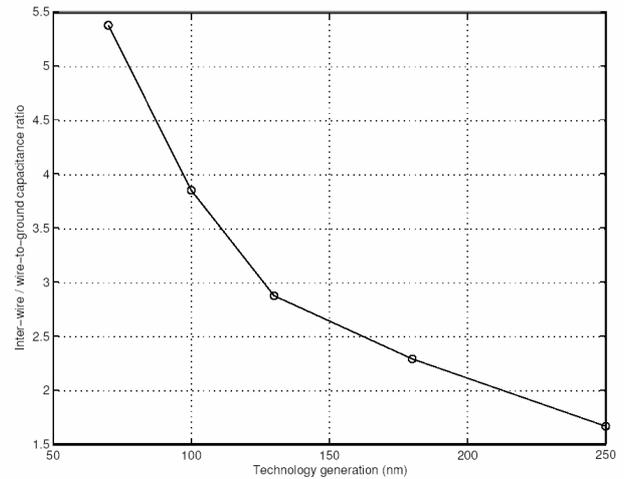


Abbildung 4.2: Verhältnis von Koppel- zu Platten- und Randkapazitäten über den Technologiegenerationen (Quelle: [4])

Die dargestellten Zusammenhänge zeigen eine Abnahme des durch Logikschaltkreise und Repeater verursachten Energieverbrauchs (Gatekapazitäten, Querströme) und eine Zunahme des Anteils der Verbindungsleitungen (globale Verdrahtungs- und Koppelkapazitäten). Dies weist für Strukturbreiten oberhalb 250 nm auf eine Zunahme der Energieeffizienz beim Übergang auf kleinere Technologien hin. Mit den untersuchten Technologien nicht quantifizierbar sind die Auswirkungen des entgegengesetzten Effekts durch Leckströme bei Technologien unterhalb 250 nm.

Wie oben bereits erwähnt, lässt sich der von der Technologieskalierung erwartete positive Effekt für die untersuchten Technologien nicht bestätigen. Die Ursachen hierfür liegen in Unterschieden zwischen theoretisch erwarteten und simulierten Skalierungsfaktoren, wie nachfolgend dargestellt:

Die verwendeten Simulationswerkzeuge gestatten die Untergliederung des gesamten Energieverbrauchs in die Einzelkomponenten

- E_{Int} : Interner Energieverbrauch eines Gatters, verursacht durch interne Verdrahtungskapazitäten, Leck- und Querströme
- E_{Cin} : Durch das Aufladen der Eingangskapazitäten von Gattern verursachter Energieverbrauch
- $E_{Wire,Cl}$: Energieverbrauch verursacht durch das Aufladen der Platten- und Randkapazitäten von Signalleitungen
- $E_{Wire,Cc}$: Energieverbrauch verursacht durch das Aufladen der Koppelkapazitäten von Signalleitungen

Tabelle 4.2 stellt dar, wie die genannten Komponenten beim Übergang von der 0.5 μm -Technologie zur 0.35 μm -Technologie skalieren:

Komponente	Skalierungsfaktor
E_{Cin}	0.49
E_{Int}	0.67
$E_{Wire,Cc}$	1.04
$E_{Wire,Cl}$	0.59

Tabelle 4.2: Simulierte Skalierungsfaktoren

Die auf Lastkapazitäten zurückzuführenden Anteile skalieren stärker, als der interne Energieverbrauch der Kodierschaltungen. Somit verschieben sich die Verhältnisse zugunsten der Busse und die Kodiereffizienz sinkt beim Technologieschritt.

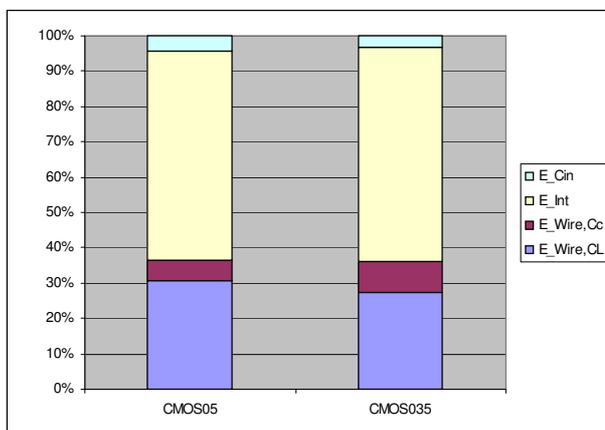


Abbildung 4.3: Simulierte Anteile der Komponenten am gesamten Energieverbrauch

5. Zusammenfassung

Für die untersuchten Technologien ist aufgrund der Simulationsergebnisse nicht zu erwarten, dass die effektiven Leitungslängen beim Einsatz der Verfahren BI und $T0$ im Bereich realistischer Werte liegen werden.

Außerdem zeigen die Simulationen des Energieverbrauchs im Gegensatz zu den vorgenommenen Abschätzungen, dass beim Übergang von 0.5 μm hin zur 0.35 μm -Technologie kein klarer Trend in Richtung höherer Energieeffizienz erkennbar ist. Die offensichtlichen Abweichungen zwischen Theorie und Praxis legen die Notwendigkeit weiterer Nachforschungen nahe. Hilfreich wäre hierbei sicher, auch den Übergang zu Technologien unter 0.35 μm betrachten zu können.

Die Parameter des Busmodells wurden bisher aufgrund von Informationen aus Datenblättern abgeschätzt und bedürfen der Verifikation anhand realistischer Layouts. Zu nennen wären hier neben den spezifischen Kapazitätswerten auch die Anzahl der Repeater, Werte für typische Buslängen und der Anteil parallel verlaufender Verbindungsleitungen.

6. Literatur

- [1] J. M. Rabaey, A. Chandrakasan, B. Nikolić, "Digital Integrated Circuits", *Prentice Hall*, 2003
- [2] CMOS05 Design Rule Manual
- [3] CMOS035 Design Rule Manual
- [4] Y. Zhang, J. Lach, K. Skadron und M.R. Stan: "Odd/Even Bus Invert with Two-Phase Transfer for Buses with Coupling", *Proceedings of the International Symposium on Low-Power Electronics and Design - ISLPED*, 2002
- [5] T. Lindkvist, J. Löfvenberg, O. Gustafsson, "Deep Sub-Micron Bus Invert Coding", *Proceedings of the 6th Nordic Signal Processing Symposium - NORSIG*, 2004
- [6] C. Kretschmar, A.K. Nieuwland, D. Müller, "Why Transition Coding for Power Minimization of on-Chip Buses does not work", *Proceedings of the Design, Automation and Test in Europe Conference and Exhibition - DATE*, 2004
- [7] D. Helms, E. Schmidt, W. Nebel, "Leakage in CMOS Circuits - an Introduction", *Integrated Circuit and System Design, Power and Timing Modeling, Optimization and Simulation - PATMOS*, 2004
- [8] M.R. Stan und W. Burleson, "Two-Dimensional Codes for Low Power", *Proceedings of the International Symposium on Low Power Electronics and Design - ISLPED*, 1996
- [9] L. Benini, G. Micheli, E. Macii, D. Sciuto und C. Silvano, "Asymptotic Zero-Transition Activity Encoding for Address Buses in Low-Power Microprocessor-Based Systems", *Proceedings of the Seventh Great Lakes Symposium on VLSI - GLSVLSI*, 1997

Synchronization of a Local XTAL-Oscillator to an External High-Precision Frequency Standard

Wimol San-Um* and Walter Ludescher**

*School of Communications, Instrumentation and Control
Sirindhorn International Institute of Technology, Bangkadi Campus
Thammasat University, Pathum-Thani, Thailand
Tel: + 662-501-3505 (until 20) ext.1102, E-Mail: wimol_siit@yahoo.com

**Department of Electronics and Computer Science
Ravensburg-Weingarten University of Applied Science
Postfach 1261, D-88241 Weingarten, Germany
Tel: 0751-501-9685, E-Mail: ludescher@hs-weingarten.de

Abstract

Synchronization of a local XTAL-oscillator to an external high-precision frequency standard is presented. The signal of an external frequency standard is received and used to monitor the frequency of a local oven-controlled XTAL-oscillator (OCXO). A frequency-error of the local OCXO causes an increase in phase difference, which is used to correct and lock the local OCXO to the reference signal. Possible references are e.g. DCF77 or TV synchronization patterns. As a design example, a 10-MHz OCXO has been implemented to be synchronized to DCF77 and a commercially available 10-MHz rubidium-disciplined crystal oscillator. In particular, a digital frequency divider, which divides the frequencies of the OCXO and the reference frequencies down to 0.01 Hz for use in synchronization process, has also been implemented in FPGA Xilinx Spartan 2S50 chip. The circuit utilizes approximately 65% of the overall area of the FPGA chip and operates at the maximum allowable clock frequency of 17.5 MHz.

1. Introduction

An Oven-Controlled XTAL Oscillators (OCXO) generally provides a high degree of frequency stability for broadcast and measurement applications where precise frequency of oscillation is critical. Typical OCXO is enclosed in a thermally insulated container along with a heating element and a temperature sensor in order to avoid temperature changes that affect the resonance frequency of a piezoelectric crystal [1]. Practically, the frequency stability of the OCXO, however, has influenced from various internal and external destabilizing factors such as a change in ambient temperature or short-term instability and long-term aging of a crystal itself [2]. These problems consequently cause a change in oscillation frequency and therefore additional techniques are necessarily

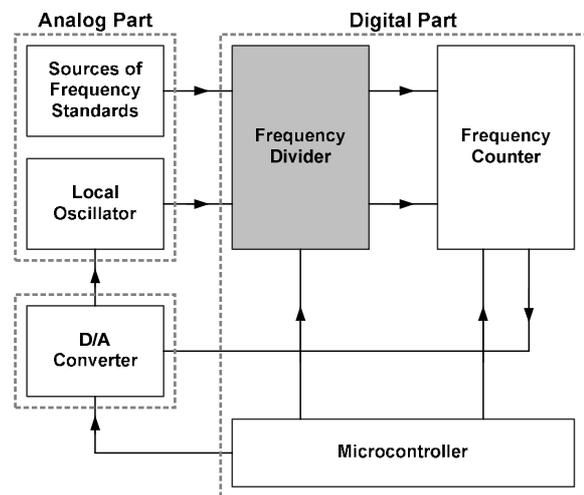


Figure 1: A block diagram of major components for synchronizing a local oscillator to a reference frequency.

required for sustaining a precise and accurate oscillation frequency [3]. Although dual-mode and modulational methods [4-5] have recently been proposed for corrections in oscillation frequency of the OCXO, a large modification of OCXO circuits is required and results in relatively complex circuit implementations. Besides, GPS-controlled methods that compare a phase difference between multiplied local oscillator frequency and successive GPS pulses have also been proposed [6-7]. Such techniques, however, measure the phase difference at high frequency, i.e. 100MHz, and therefore synchronizing process may not as accurate as low frequency operation. In addition, external circuits such as phase-locked loops and filters are also necessarily required.

Therefore, an alternative technique for a correction in the oscillation frequency of the local OCXO using time-based synchronization is proposed. Figure 1 shows the block diagram of major components for

synchronizing a local oscillator to a reference frequency. As shown in Figure 1, the circuit consists mainly of analog and digital parts. The analog part includes a local oscillator and sources frequency standards. On the other hand, the digital part includes a frequency divider and a frequency counter. The frequency divider divides the frequency of high-frequency signals from the local OCXO and the reference frequency down to, for example, 1 Hz for precise phase comparisons. These low-frequency signals will be sent to the frequency counter, which compares a phase difference and provides a frequency-error. The resulting frequency-error will subsequently be sent back to the D/A converter, interfacing between analog and digital parts, and used to adjust the oscillation frequency of the local OCXO. Note that the digital part is controlled by a microcontroller so that the circuit can operate automatically.

Based on the concept described in Figure 1, a 10-MHz OCXO has been implemented as a local oscillator. An international call sign named DCF77 with a carrier of 77.5 kHz has been chosen as a major reference frequency. Additionally, the commercially available PRS10 named RUBY, which is an ultra low phase noise 10-MHz rubidium-disciplined crystal oscillator, has also been considered as an alternative reference frequency.

It is obvious in Figure 1 that the frequency divider plays an important role in this synchronization process and this paper therefore presents the implementation of digital frequency dividers in the Field Programmable Gate Array (FPGA) using Very High-Speed Description Language (VHDL). Additional functions, i.e. a bit-error monitoring, a phase synchronizer and a microcontroller interface, are also included. This paper is organized as follows: Section 2 describes detailed designs of frequency divider blocks, Section 3 then gives details of additional functions, Section 4 deals with the architecture of the top level design and pin descriptions in the FPGA board. Section 5 shows synthesized and experimental results. Finally, recommendations for future works and conclusions are provided in Sections 6 and 7, respectively.

2. Design of Frequency Dividers

Designs of digital frequency dividers in VHDL can be based on many techniques such as a cascade connection of divide-by-2 dividers using flip-flops [8] or a dual-modulus divider [9]. However, such techniques offer a relatively small value of divisor and may not be synthesizable for a relatively complex design with a large value of divisor. Therefore, this work alternatively develops the frequency dividers using a Finite State Machine (FSM) technique, which offers not only a simple design process but also a flexibility to include some additional functions such as set, clear and scan data operations. The FSM technique is primarily used to design seven different versions of basic frequency divider blocks which will

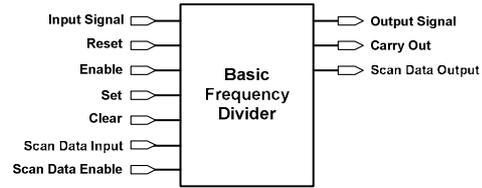


Figure 2: A block diagram of a basic frequency divider demonstrating all possible input and output terminals.

Table 1: Summary of basic frequency divider blocks; (a) Designed entities, (b) Available terminals; \checkmark = Yes, \times = No.

(a) Entities		(b) Available Terminals									
Entity Names	Values of Divisors	Inputs						Outputs			
		Input Signal	Global Reset	Enable	Set	Clear	Scan Data Input	Scan Data Output	Output Signal	Carry Out	Scan Data Output
Dlf10x	$\div 10$	\checkmark	\checkmark	\times	\times	\times	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark
Dlf10e	$\div 10$	\checkmark	\checkmark	\checkmark	\times	\times	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark
Dls10x	$\div 10$	\checkmark	\checkmark	\times	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark
Dls10e	$\div 10$	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark
Drs31x	$\div 31$	\checkmark	\checkmark	\times	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark
Drs5e	$\div 5$	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark
Dl4x	$\div 4$	\checkmark	\checkmark	\times	\times	\times	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark

be connected in cascade in order to provide a variety of divided frequency and a large value of divisor.

2.1 Basic Frequency Divider Blocks

Figure 2 shows the general block diagram of the frequency divider which demonstrates all possible input and output terminals. Typically, two major input terminals are an input clock and a global reset whilst two major output terminals are a divided output frequency and a carry out. Additionally, scan data operations, which are used to monitor a proper operation of internal components especially for ASIC design, can be available through scan data enable (SDE), scan data input (SDI) and scan data output (SDO) terminals.

As will be seen later, some frequency divider blocks in the top level design, however, may not include enable, set and clear operations. Therefore, seven different versions of frequency divider blocks have been designed in order to meet a requirement of the top level design. Table 1 summarizes the designed entities and corresponding terminals of the basic frequency divider blocks where the major difference is an inclusion of enable, set and clear functions. It can be seen from Table 1 that there are four versions of divide-by-10 dividers, i.e. *Dlf10x*, *Dlf10e*, *Dls10x* and *Dls10e*. In addition, divide-by-5 and divide-by-31 dividers, which are specifically used for dividing the frequency of DCF77, are denoted as *Drs5e* and *Drs31x*, respectively. Finally, a divide-by-4 divider denoted as *Dl4x* for use in the bit-error monitoring circuit.

2.2 A Design Example of the *Dls10e*

With reference to Table 1, detailed design and analysis of the divide-by-10 frequency divider called *Dls10e* with enable, set and reset functions are demonstrated as a particular example. Typically, a state diagram of the FSM is primarily designed before coding in VHDL. Figure 3 shows the state diagram of the *dls10e*. As shown in Figure 3, the circuit consists of 10 states denoted from S0 to S9. Each state has 4 bits in which the output is the least significant bit and the carry out is a combination of these bits. When both global reset and enable are set to High, the circuit counts the ascending slopes of the input clock and changes to the corresponding state.

Additionally, set and clear functions are used to set the current operating state to states S9 and S0, respectively. Figure 4 shows the timing diagram of the *dls10e* under a normal operation. It can be seen from Figure 4 that ten periods of the input clock are reduced to only one period of the output. In other words, the frequency of the clock is divided by ten with a duty cycle of 50%. In addition, the carry out, which will be used as an enable signal for the following stage, is provided at the last clock period.

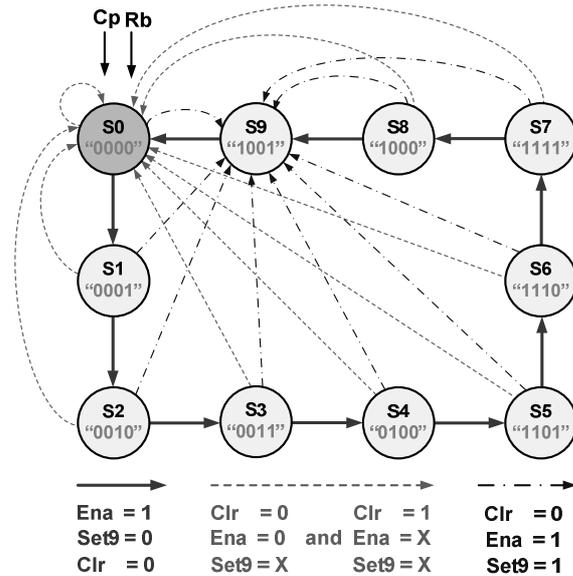


Figure 3: A state diagram of a divide-by-10 frequency divider with enable, set and clear functions.

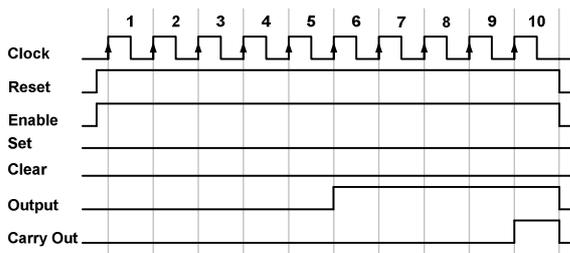


Figure 4: Timing diagrams of a divide-by-10 frequency divider with enable, set and clear functions.

Table 2: Summary of divider blocks; (a) Designed entities, (b) Available output frequencies; \checkmark = Yes, \times = No.

(a) Entities			(b) Available Output Frequencies										
Names of Entities	Input Sources	Set and Clear	1 MHz	100kHz	10 kHz	2.5kHz	5kHz	1kHz	100Hz	10Hz	1Hz	0.1Hz	0.001Hz
			DIV1	OCXO	\times	\checkmark	\checkmark	\checkmark	\times	\times	\checkmark	\checkmark	\checkmark
DIV2	OCXO	\checkmark	\checkmark	\checkmark	\checkmark	\times	\times	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark
DIV3	RUBY	\checkmark	\times	\times	\times	\checkmark							
DIV4	DCF77	\checkmark	\checkmark	\checkmark	\checkmark	\times	\times	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark

2.3 Cascade Connections of Basic Frequency Divider Blocks

As mentioned earlier, two reference frequencies, i.e. 10 MHz and 77.5 kHz, will be divided down to very frequencies of 1 Hz or less for synchronization. Therefore, a cascade connection of basic frequency divider blocks summarized in Table 1 can offer a large value of divisor, for example, a divisor of 10,000,000 is required for dividing the frequency of 10MHz down to 1 Hz. Figure 5 shows the cascade connection of divide-by-10 divider blocks for dividing a frequency of 10 MHz down to 0.01 Hz. As shown in Figure 5, the input clock frequency is 10 MHz and consequently nine of divide-by-10 divider blocks are connected in cascade. It is also seen from Figure 5 that nine of divided output frequencies are available, ranging from 1 MHz to 0.01 Hz. Note that the first divide-by-10 block does not require an enable signal whilst other remaining blocks use the carry out from the previous stage as an enable signal. This configuration can also be used as a clock for other systems. In addition, Figure 6 shows the cascade connection of frequency divider blocks for dividing a frequency of 77.5 kHz down to 0.01 Hz. Note that number 77500 cannot be divided using only divide-by-10 blocks and therefore the divide-by-31 and divide-by-5 divider blocks are initially used.

With reference to Figures 5 and 6, four particular frequency dividers, i.e. *DIV1*, *DIV2*, *DIV3* and *DIV3*, have been designed for the top level. Table 2 summarizes the designed entities of four frequency dividers for the local OCXO and the two reference frequencies. It can be seen from Table 2 that blocks *DIV1*, *DIV2* and *DIV4* are used to divide the 10-MHz signals from OCXO and RUBY. The major difference between *DIV1* and *DIV2* is an inclusion of set and clear operations. Additionally, the block *DIV3* is particularly used for dividing the 77.5 kHz signal from DCF77.

3. Design of Additional Functions

Three additional circuits included in the top level design of the FPGA chip are (1) the bit-error motoring circuit, (2) the phase synchronization circuit and (3) the microcontroller interface circuit. Table 3

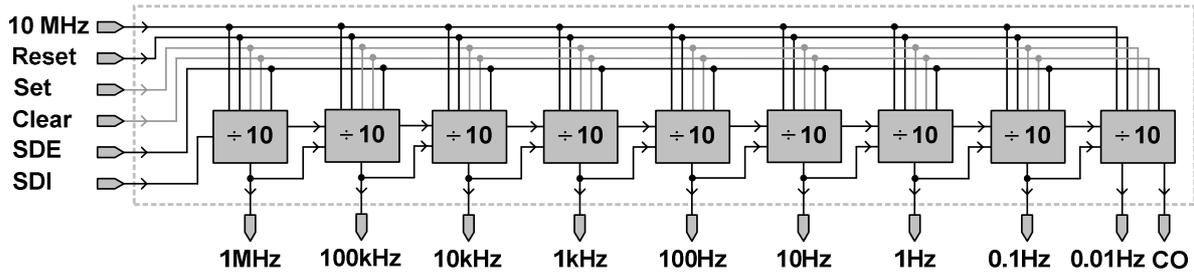


Figure 5: A cascade connection of frequency divider blocks for dividing the frequency of 10 MHz down to 0.01 Hz.

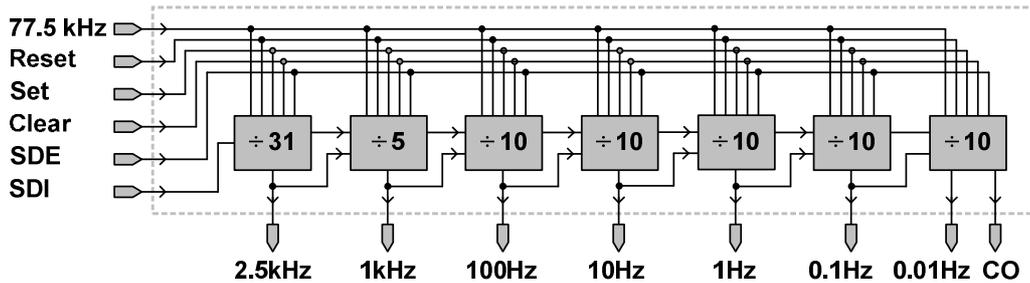


Figure 6: A cascade connection of frequency divider blocks for dividing the frequency of 77.5 kHz down to 0.01 Hz.

summarizes the designed entities for these three circuits. Firstly, the bit-error monitoring circuit is denoted in the designed entity as *BMC* and used to monitor receiving error bits, which may possibly be a result of either unclear atmosphere, especially during that day time, or a fault of the receiver circuit. The monitoring process is possible through a comparison of phases between two investigating signals. For instance, the error bits of DCF77 can be monitored by comparing the phases between the available frequency of 2.5 kHz from the block *DIV4* and the frequency of 2.5 kHz divided from the OCXO using the divide-by-4 divider (*DIV4x*) described in Table 1. The resulting phase-difference can be sent to a microcontroller for calculating an amount of bit errors. This phase-difference can also be used in digital phase-locked loops for further improvement.

Secondly, the phase synchronization circuit is denoted in the designed entity as *PHC* and used to synchronize the phases of the OCXO and other two reference frequencies. In other words, this circuit internally provides a “Set” signal to start the ascending slopes of the frequency dividers *DIV2*, *DIV3* and *DIV4* at the same time. Consequently, the frequency counter can start counting the frequency-error at the same ascending slopes of all frequencies resulting in a precise counting of time shift. Note that the set operation is controlled by the microcontroller and therefore no external terminal is available for the Set input signal in the top level design.

Finally, the microcontroller interface circuit is denoted in the designed entity as *uCIF* and designed for purposes of an automatic control process and a reduction of pins in the FPGA board. The design of the microcontroller interface circuit is based on a 4-bit shift register and a decoder. This circuit has exploited

only five external input pins of the FPGA board, including a single address bit and 4-bit control signals, and these pins are directly connected to the microcontroller. The resulting 16 sets of 4-bit control signals are internally provided as useful outputs for controlling other circuits such as multiplexers. It should be noted that this circuit requires two steps of operation, i.e. the set of control signals must initially be selected by setting the address to High and the control signals can subsequently be loaded into the selected set by setting the address to Low.

Table 3: Summary of designed entities for bit-error monitoring, phase synchronization and microcontroller interface circuits ;(a) Designed entities,(b) Descriptions.

(a) Design Entities		(b) Descriptions
Names	Operating Clock Sources	
<i>BMC</i>	10-MHz OCXO	Bit-error monitoring
<i>PSC</i>	10-MHz OCXO	Phase synchronization
<i>uCIF</i>	Internally generated clock in FPGA	Microcontroller interface

4. Architecture of Top Level Design

Figure 7 shows the block diagram of the top level design of the digital frequency divider for synchronization of the 10-MHz local OCXO to two reference frequencies of 77.5 kHz from DCF77 and 10 MHz from RUBY. As shown in Figure 7, the top level design employs all designed entities previously described in Tables 2 and 3. In addition, one of 4-to-1 multiplexer and four of 16-to-1 multiplexers are

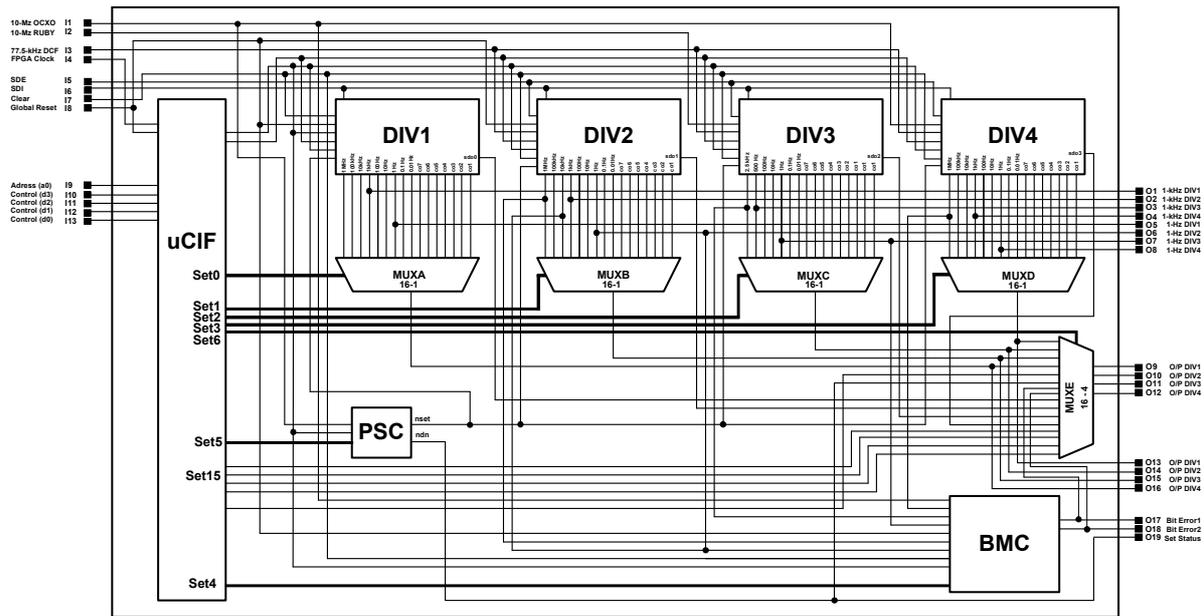


Figure 7: A block diagram of the top level design of the digital frequency divider for synchronization of the 10-MHz local OCXO to standard frequencies of 77.5 kHz from DCF77 and 10 MHz from RUBY.

employed for selecting a frequency of interest. As the 16-to-1 multiplexers require 16 input signals, the carry outs from each frequency dividers blocks are therefore included with those divided frequencies in order to fulfill a requirement of 16 inputs of the multiplexers. With reference to Figure 7, this design has exploited totally 32 pins of the FPGA board.

On the one hand, the circuit uses 13 pins of the FPGA board for input signals. Three external input clock signals from the local OCXO, RUBY and DCF77 are connected to the available FPGA clock pins at pins 11, 12 and 13, respectively. The internally generated clock in the FPGA board, which is available at pin 14, is inherently used as an operating clock for the microcontroller interface, respectively. The SDE and SDI are available at pins 15 and 16, respectively. Pins 17 and 18 are available for global reset and clear signals, respectively. Note that the reset, clear and scan data pins are connected to the line printers so that the values can be programmed by a computer. In addition, the output pins from the microcontroller that provides sets of control signals can be interfaced to pins 19, 110, 111, 112 and 113.

On the other hand, the circuit uses 19 pins of the FPGA board to provide the output signals. It is seen from Figure 7 that there are four sets of divided output frequencies. Firstly, the set of 1-kHz frequencies which is provided by those four dividers DIV1, DIV2, DIV3 and DIV4 at pins O1, O2, O3 and O4, respectively. Secondly, the set of 1-Hz frequencies which is also provided by those four dividers and directly connected to the light-emitting diodes (LEDs) at pins O5, O6, O7 and O8. These two sets have been designed for investigating a proper operation of the circuit and therefore the expected 1-

Hz and 1-Hz output signals must automatically be visible after connecting to the power supply.

Thirdly, the set of selectable divided output frequencies through the output multiplexer are provided at pins O9, O10, O11 and O12. The output signals of this set are directly connected to the microcontroller for a consequent control process. Finally, the same set of the selectable divided output frequencies mentioned above. Instead of connecting to the microcontroller, these signals, however, are provided at external pins O13, O14, O15 and O16. This output signals will be used for synchronization process through the frequency counter. In addition, pins O17 and O18 provide the monitored bit errors of the investigating input clocks. The set status which is connected to pin O19 can also be observed.

5. Experimental Results

The designed VHDL codes of the final top level design using frequency divider blocks and additional circuits described in Tables 2 and 3 were initially simulated through ModelSim in SunSolaris Microsystems environments. Such VHDL codes were consequently synthesized through Leonardo for routing the internal gates and generating a bit stream to be loaded into the FPGA chip. The target technology of this work is the FPGA Xilinx Spartan 2TQ144S50. Table 4 summarizes the synthesized results through Leonardo of the top level design shown in Figure 7. It can be seen from Table 4 that the circuit utilizes approximately 65% of the FPGA chip. The maximum allowable of the clock frequencies are more than 10 MHz and sufficient for this work where the maximum operating frequency is 10 MHz.

Table 4: Synthesized results of the final top design from Leonardo ;(a) Device utilization,(b) Maximum allowable clock frequencies.

	Resources	Used	Available	Percents
(a) Device Utilization	IOs	28	92	30.43%
	Function Generators	987	1536	64.26%
	CLB Slices	494	768	64.32%
	DFFs of Latches	209	2082	10.04%
(b) Clock Frequencies	OCXO	17.5 MHz		
	Internal Clock	50.1 MHz		
	77.5-kHz DCF77	22.6 MHz		
	10-MHz RUBY	17.5 MHz		

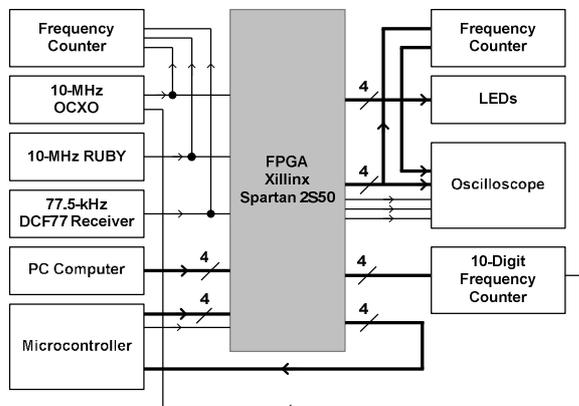


Figure 8: A block diagram the experimental set up of the synchronization process.

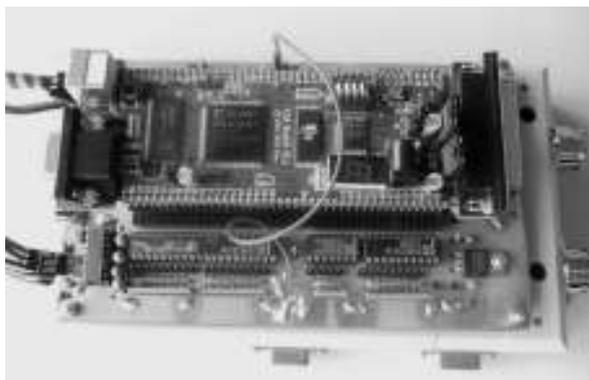


Figure 9: A photograph of the implemented circuit on board in this experiment.

Figure 8 shows the block diagram of the experiment setup. As shown in Figure8, the input signals were from the 10-MHz local OCXO and the two reference frequencies. The PC computer was used to set the values of reset, clear and scan data operations. In addition, the frequency counter was connected to the input frequency sources for investigating the input frequency. The set of low frequency signals of 1 Hz was connected to the LEDs for investigating

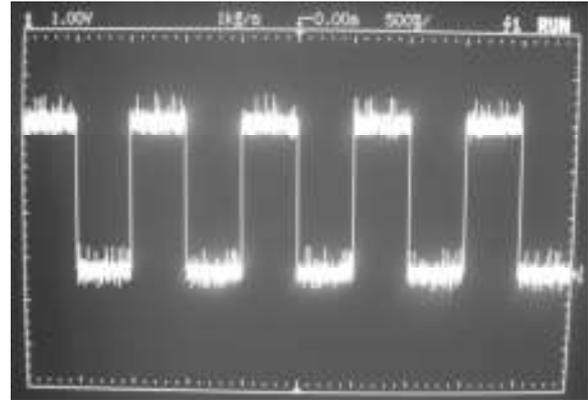


Figure 10: A photograph of waveforms of the 1-Hz divided frequency.

operations of the circuit. Moreover, both sets of 1-Hz and 1-kHz signals were also connected to the oscilloscope in order to investigate the waveforms.

In particular, the useful divided frequency of 1 Hz was directly connected to the frequency counter in order to count the time shift and sent back to the local OCXO. Figure 9 shows the implemented board of the FPGA chip and the microcontrollers. As a simple example, the waveform of the output signal with the frequency of 1 Hz from the OCXO is demonstrated in Figure 10.

6. Recommendation for Future Works

As the objectives of this work are the implementations of frequency dividers in the FPGA chip, the results has shown that high-frequency input signals from both local oscillator and reference frequencies can be divided down to 0.01Hz for use in time base synchronization. This primarily designed frequency dividers in FPGA chip can be further improved through ASIC design for synchronizing to other very high frequency standard such as TV synchronization patterns. Based on the top level design shown in Figure 7, three recommendations are as follows; Firstly, the number input pins for scan data functions as well as the clear operation can be reduced by placing into the control address interfacing with the microcontroller. Secondly, the scan data function can be separately connected for each block for a faster operation, especially for an implementation in AISIC. Finally, the final bit stream can be statically stored in the memory so that the circuit can automatically operate after connecting the power supply.

7. Conclusions

Synchronization of a local XTAL-oscillator to an external high-precision frequency standard has been shown. The applied technique is relatively simple through the use of an external frequency standard to monitor the frequency of a local oven-controlled XTAL-oscillator. A frequency-error of the local OCXO causes an increase in phase difference which is used

to correct and lock the local OCXO to the reference signal. As a particular example, the digital frequency divider, which divides the frequencies of the 10-MHz OCXO and the reference frequencies, i.e. DCF77 and RUBY, down to 0.01 Hz for use in synchronization process, has also been implemented in FPGA Xilinx Spartan 2S50 chip. The circuit utilizes approximately 65% of the overall area of the FPGA chip and operates at the maximum allowable clock frequency of 17.5 MHz. The implemented circuit and low-frequency output waveforms have also been demonstrated. This work therefore offers a cost-effective alternative to adjust and monitor high-precision OCXO in real world applications

8. Acknowledgements

The authors are grateful to Baden-Württemberg Scholarship for financial supports to conduct a 6-month research project in electronic design at Ravensburg-Weingarten: University of Applied Science. Additionally, the authors would like to thank Mr. Weber for his useful helps in computer systems.

9. References

- [1] K.Hiroshi, I.Manabu and U.Takeshi, " Ovencontrolled crystal oscillator for high stability", www.freepatentsonline.com.
- [2] www.radio-electronics.com.
- [3] L. Vorokhovsky, V. Anastastev and A. Volkov, "The significant improvement of Basic OCXO for time and frequency standards", In the Proceedings of the IEEE International Frequency Control Symposium and PDA Exhibition Jointly with the 17th European Frequency and Time Forum, 2003.
- [4] Y. Watanabe, K. Ozaki, S. Goka and H. Sekimoto, " Ultra-Stable OCXO using Dual-mode Crystal oscillator", In the proceeding of IEEEIEIA International Frequency Control Symposium and Exhibition, 2000.
- [5] S. Shmaliy, "The Modulational Method of Quartz Crystal Oscillator Frequency Stabilization", In the publication of IEEE transactions on ultrasonics, ferroelectrics, and frequency control, vol. 45, no. 6, November 1998.
- [6] A. Lisowis, A. Czarnecki and Z. Rau, "A method for quartz oscillator synchronization by GPS signal" ,In the Proceeding of European Frequency Time Forum, 5-7 March 1996.
- [7] C. Cheng, F. Chang, and K. Tu, "Highly Accurate Real-Time GPS Carrier Phase-Disciplined Oscillator", In the publication of IEEE Transactions of instrument and measurement, Vol.54.No.2, April 2005.
- [8] www.hyperphysics.phy-astr.gsu.edu.
- [9] B. Boorman, "Design of dual-modulus dividers in an FPGA" EDA Magazine, September 28, 2006.

Simulation Rechenintensiver A/MS-Systeme in der Automobilindustrie

Dr.-Ing. Rolando Dölling, Robert Bosch GmbH
Dipl. Bio-Informatik Holger Mielenz, Robert Bosch GmbH

Automotive Electronics

1

AE/EIM3 - Dr.-Ing. R. Dölling | 02.02.2007 | © Alle Rechte bei Robert Bosch GmbH, auch für den Fall von Schutzrechtsanmeldungen. Jede Verfügungsbefugnis, wie Kopier- und Weitergaberecht, bei uns.



BOSCH

Abstract

Die steigenden Anforderungen an den Entwurfsprozess in der Automobilelektronik lassen die Entwicklung neuer Methoden notwendig werden, die zu einer Verbesserung der Entwurfs-effizienz beitragen können. Die funktionale Verifikation per Simulation stellt einen wichtigen Teilschritt in der Überprüfung der Spezifikationskonsistenz entworfenen Schaltungskomponenten dar. Auf Transistorebene ist dies meist mit sehr langen Simulationszeiten verbunden, weshalb in größeren Designs zeitkritische Komponenten durch manuell erzeugte Verhaltensmodelle ersetzt und Mixed-Level Simulationen durchgeführt werden.

In dieser Arbeit wird zunächst eine datenbasierte Methode vorgestellt, die ausgehend von vorab generierten Simulationsdaten, Verhaltensmodelle für Analog-Digital Mixed-Signal (A/MS) Schaltungen automatisch erstellt. In der Mixed-Level Simulation besitzen derartige Modelle deutlich reduzierte Simulationszeiten bei hoher Genauigkeit. Als formelbasierte Modelle sind sie zudem nicht von bestimmten Simulatoren bzw. Hardwarebeschreibungssprachen abhängig, sondern können über einfache Schnittstellen in Simulink-, MAST- oder VHDL-AMS basierte Designs eingebunden werden.

Weiterhin wird das Zusammenspiel zwischen den verschiedenen Simulatoren, die rechenintensive Modelle beinhalten, mit Hilfe einer hausinternen Schnittstelle namens InSiMS (Integrated Co-Simulation for Mixed Signal Systems), vorgestellt. InSiMS übernimmt die Kommunikation zwischen den verschiedenen Simulatoren und hilft auch bei der Generierung von Simulationsdaten für die Erstellung von datenbasierten Modelle. Dadurch sind Co-Simulationen bei unterschiedlichsten Abstraktionsebenen eines Designs möglich. Typische Anwendungsbereiche für die Gesamtsimulation mit InSiMS bei Bosch sind Sensoren und deren komplette Auswerteelektronik sowie Endstufen, Magnetkreise und Mechanik und Hydraulik von Einspritzsystemen. Einige dieser Beispiele werden im Rahmen des Vortrags kurz vorgestellt.

Automotive Electronics

2

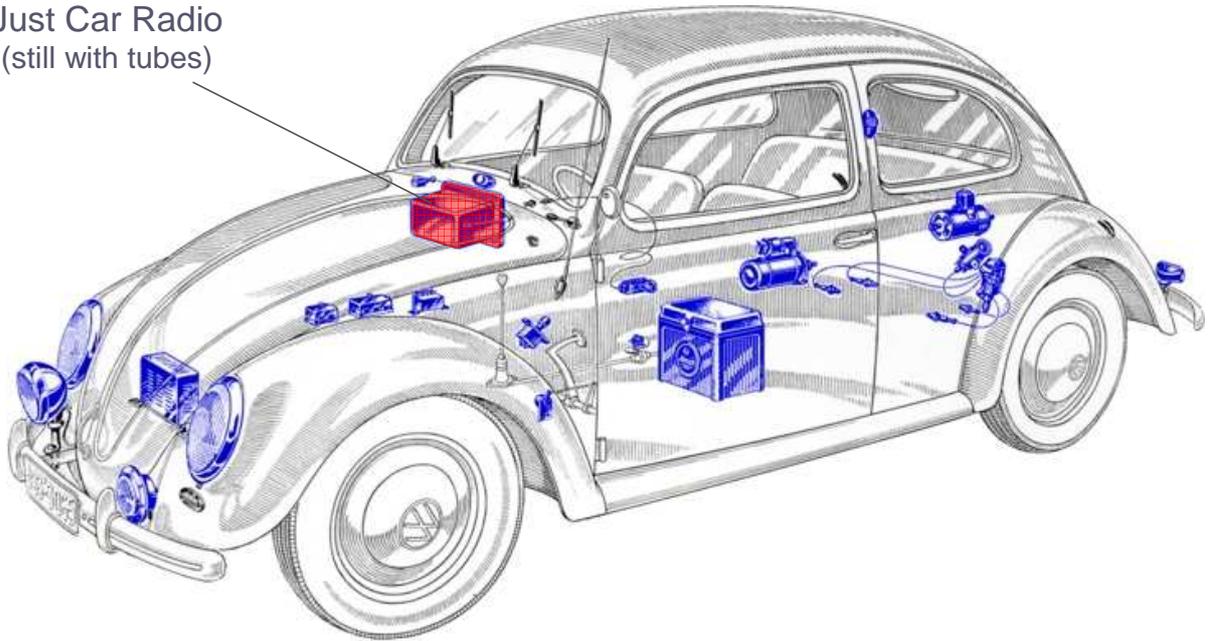
AE/EIM3 - Dr.-Ing. R. Dölling | 02.02.2007 | © Alle Rechte bei Robert Bosch GmbH, auch für den Fall von Schutzrechtsanmeldungen. Jede Verfügungsbefugnis, wie Kopier- und Weitergaberecht, bei uns.



BOSCH

Automotive Electronic Yesterday ...

Just Car Radio
(still with tubes)



Source: Bosch

Automotive Electronics

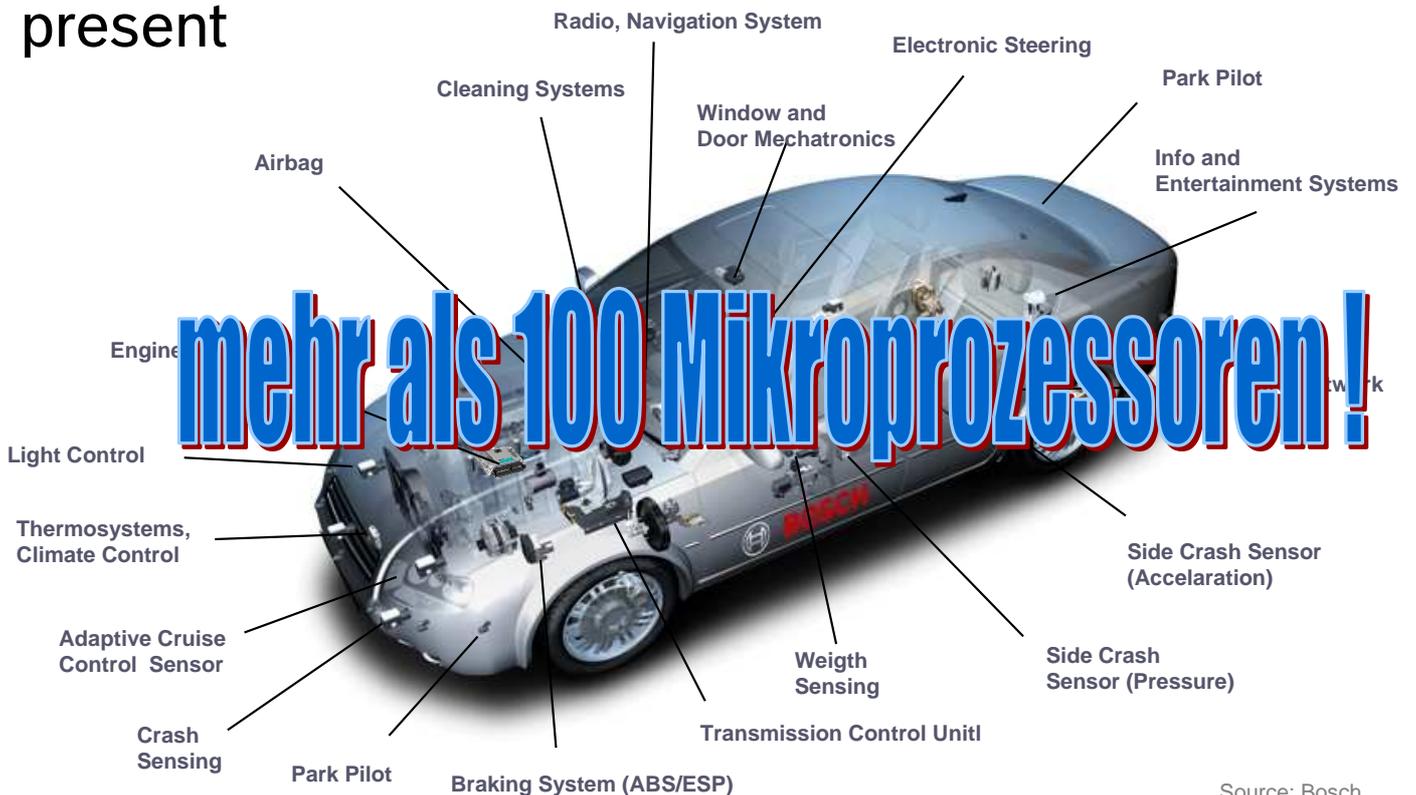


BOSCH

3

AE/EIM3 - Dr.-Ing. R. Dölling | 02.02.2007 | © Alle Rechte bei Robert Bosch GmbH, auch für den Fall von Schutzrechtsanmeldungen. Jede Verfügungsbefugnis, wie Kopier- und Weitergaberecht, bei uns.

present



Source: Bosch

Automotive Electronics



BOSCH

4

AE/EIM3 - Dr.-Ing. R. Dölling | 02.02.2007 | © Alle Rechte bei Robert Bosch GmbH, auch für den Fall von Schutzrechtsanmeldungen. Jede Verfügungsbefugnis, wie Kopier- und Weitergaberecht, bei uns.

Inhalt

- Motivation
- Idee zur Simulationsperformanz
- Darstellung der Methodik
 - Konzeption - Identifikation
 - Gewinnung der Datenbasis
 - Identifikation und Modellbildung
 - Neuronale Netze
 - Fuzzy
 - Neuro-Fuzzy
 - Support Vektor Regression
- Anwendungsbeispiel: Charge Pump
- Ergebnisse
- Weitere Anwendungen

- Zusammenfassung

Automotive Electronics

5

AE/EIM3 - Dr.-Ing. R. Dölling | 02.02.2007 | © Alle Rechte bei Robert Bosch GmbH, auch für den Fall von Schutzrechtsanmeldungen. Jede Verfügungsbefugnis, wie Kopier- und Weitergaberecht, bei uns.

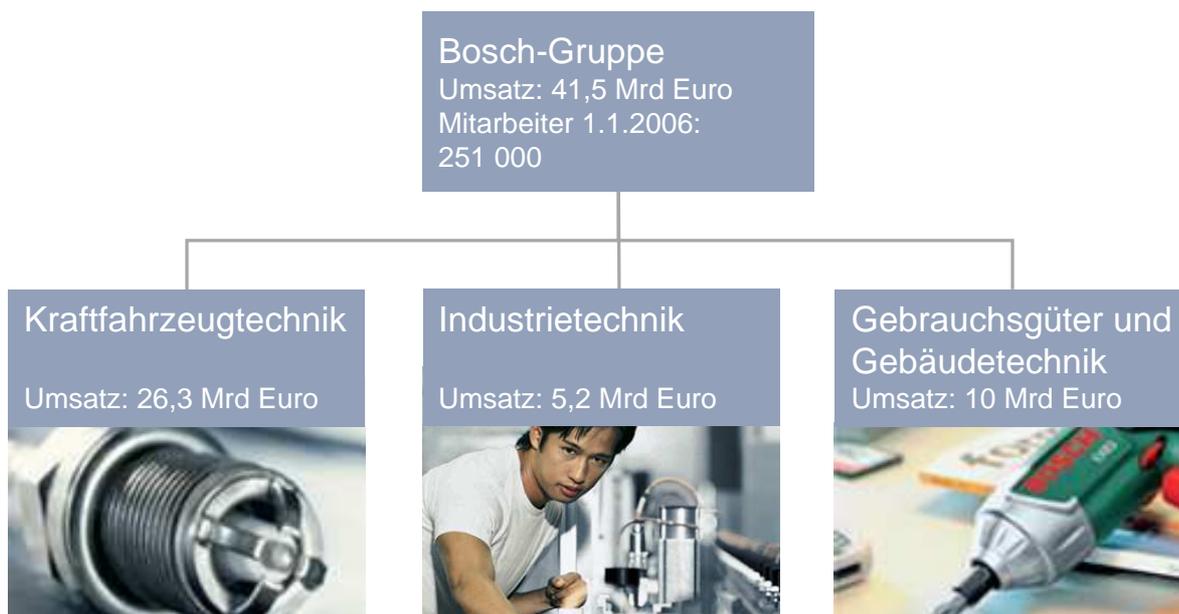


BOSCH

Bosch-Gruppe – Fakten und Zahlen

Die Struktur der Bosch-Gruppe

Unternehmensbereiche



6



BOSCH

Unternehmensbereich Kraftfahrzeugtechnik

Geschäftsbereiche



Bosch Engineering GmbH³⁾

¹⁾ Blaupunkt GmbH (100 % Bosch)

²⁾ ZF Lenksysteme GmbH (50 % Bosch)

³⁾ Entwicklungsdienstleistungen auf dem Gebiet elektronischer Fahrzeugsysteme



Simulation Rechenintensiver A/MS-Systeme

MPC-Workshop, Pforzheim

Sensorsystems for automotive applications

Inertial sensors

Gyroscope
(Airbag, Navigation)



Accelerometers
(Airbag, ESP)

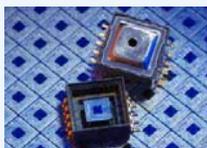


Yaw rate sensors
(ESP)



Pressure sensors

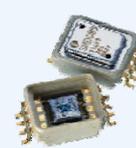
Manifold air pressure
(Motronic)



High pressure
(Motronic)



Barometric pressure
(Motronic, Airbag)



Media and weight sensors

i-Bolt
(Seat occupancy)



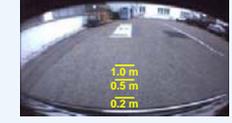
CO₂, NO,
temp., H₂O



Oil condition



Driver Assistance Systems: Comfort and Safety

<p>Ultra-Sonic</p>		<p>Standard Parking</p>		<p>Parking Space Measurement</p>		<p>Semi-autonomous Parking Assistant</p>	
<p>Long Range Radar</p>		<p>ACC > 30km/h</p>		<p>ACC plus 0 to ~200km/h</p>		<p>Predictive Safety Systems</p>	
<p>Integrated Vision System</p>		<p>Night Vision Support</p>		<p>Lane Departure Warning</p>		<p>Video Park Pilot</p>	

Automotive Electronics

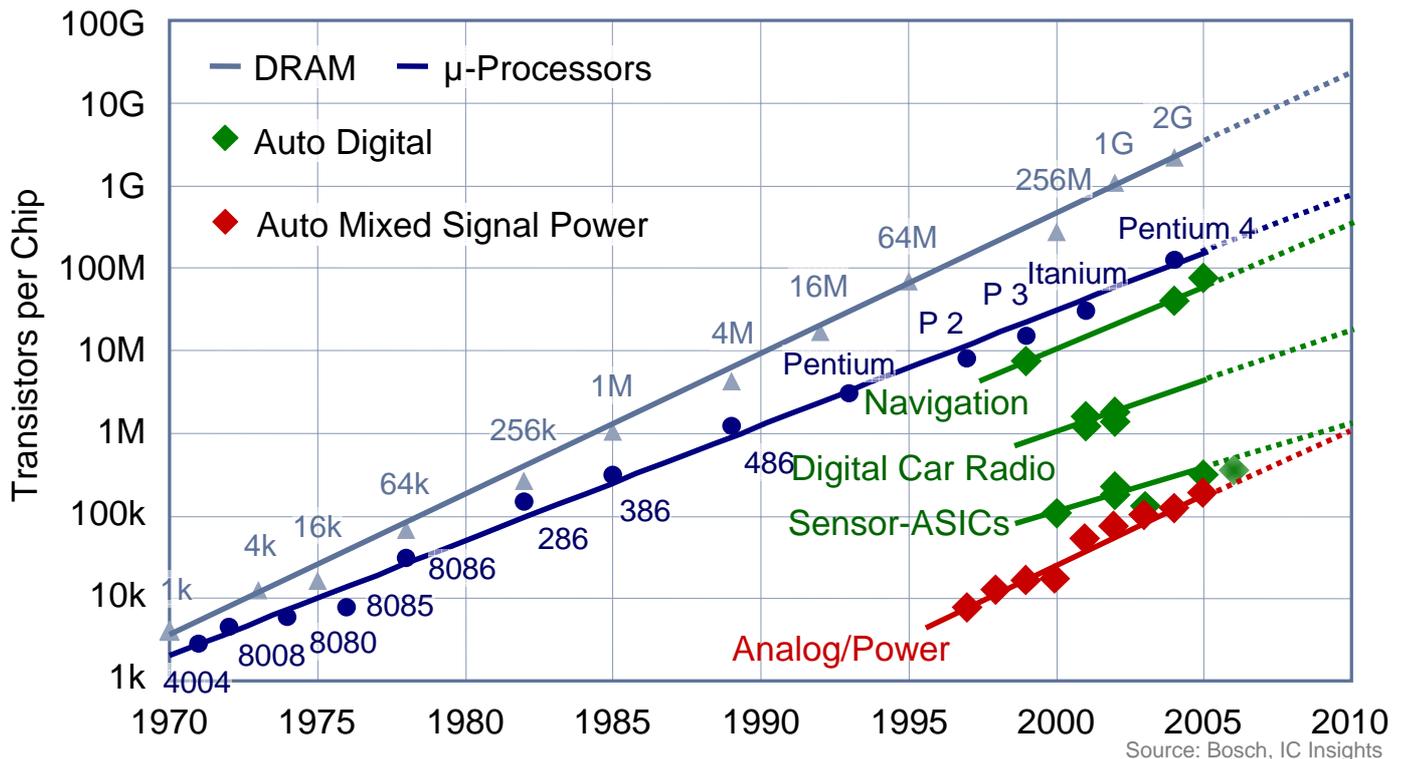


BOSCH

9

AE/EIM3 - Dr.-Ing. R. Dölling | 02.02.2007 | © Alle Rechte bei Robert Bosch GmbH, auch für den Fall von Schutzrechtsanmeldungen. Jede Verfügungsbefugnis, wie Kopier- und Weitergaberecht, bei uns.

Automotive Semiconductors follow the Trend



Automotive Electronics



BOSCH

10

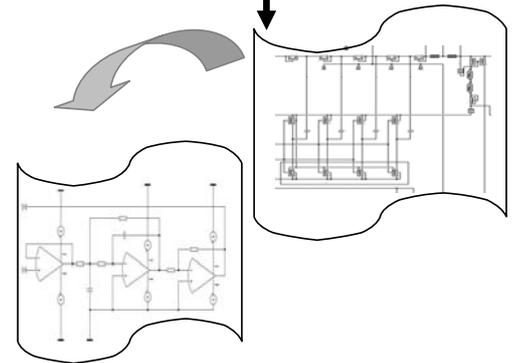
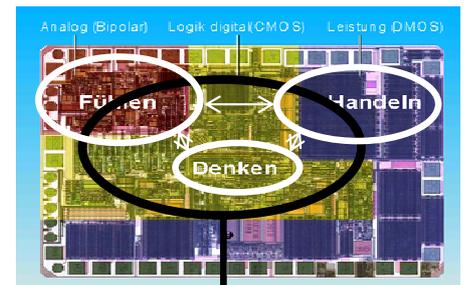
AE/EIM3 - Dr.-Ing. R. Dölling | 02.02.2007 | © Alle Rechte bei Robert Bosch GmbH, auch für den Fall von Schutzrechtsanmeldungen. Jede Verfügungsbefugnis, wie Kopier- und Weitergaberecht, bei uns.

Motivation – Mixed-Level Simulation

Funktionale Verifikation per Simulation ist ein wichtiger Schritt im Entwicklungsprozess

Simulation ist auf Transistorebene meist mit **sehr langen Rechenzeiten** verbunden

Lösung:
Ersetzung rechenzeitkritischer Komponenten durch funktionale schnellere Modelle
"Pin Kompatibel" (Transistorebene)



Reduktion der Simulationszeiten !

Automotive Electronics

11 AE/EIM3 - Dr.-Ing. R. Dölling | 02.02.2007 | © Alle Rechte bei Robert Bosch GmbH, auch für den Fall von Schutzrechtsanmeldungen. Jede Verfügungsbefugnis, wie Kopier- und Weitergaberecht, bei uns.



BOSCH

Motivation – Einsatz und Ziele

Einsatz:

- Modellierung analoger und gemischt analog/digitaler Schaltungen (" Pin - Kompatibel ")
- Einsatz zur funktionalen Verifikation in der Mixed-Level Simulation (Spezifikationskonsistenz)

Ziele:

- Verkürzung der Simulationszeiten komplexer Systeme (x100)
- Simulation Gesamtsysteme (Co-Simulation)
- Fehler bei Modellgenerierung < 2%
- Weitestgehend automatisiertes Verfahren

+/-

- „Modellverschlüsselung“ (*black-box* Ansatz)
Kein Rückschluss auf Technologie-Parameter möglich
Keine Vorkenntnisse über die Schaltung notwendig
- Verhaltensmodell durch Modellierung auf Systemebene

Automotive Electronics

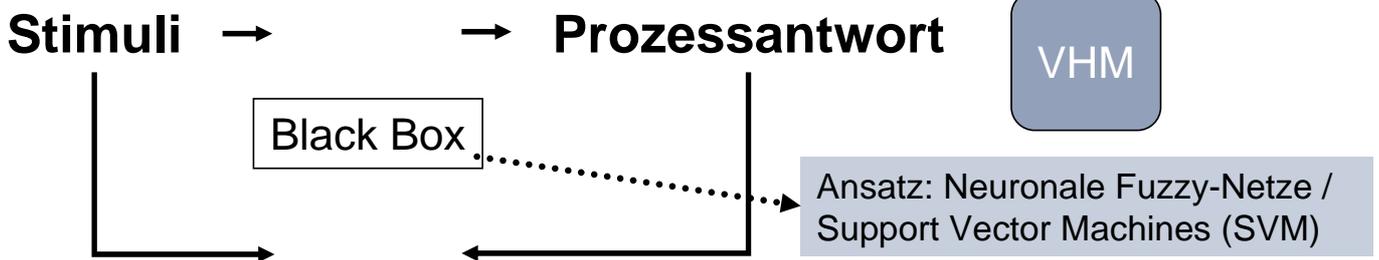
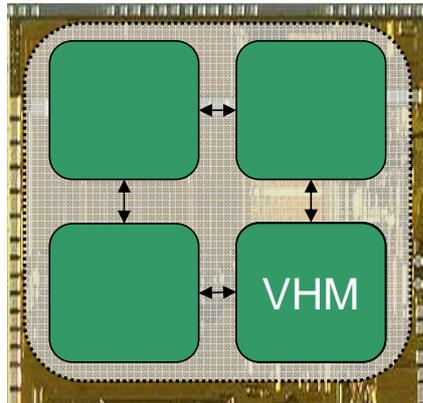
12 AE/EIM3 - Dr.-Ing. R. Dölling | 02.02.2007 | © Alle Rechte bei Robert Bosch GmbH, auch für den Fall von Schutzrechtsanmeldungen. Jede Verfügungsbefugnis, wie Kopier- und Weitergaberecht, bei uns.



BOSCH



ASIC



Automotive Electronics

13

AE/EIM3 - Dr.-Ing. R. Dölling | 02.02.2007 | © Alle Rechte bei Robert Bosch GmbH, auch für den Fall von Schutzrechtsanmeldungen. Jede Verfügungsbefugnis, wie Kopier- und Weitergaberecht, bei uns.



BOSCH

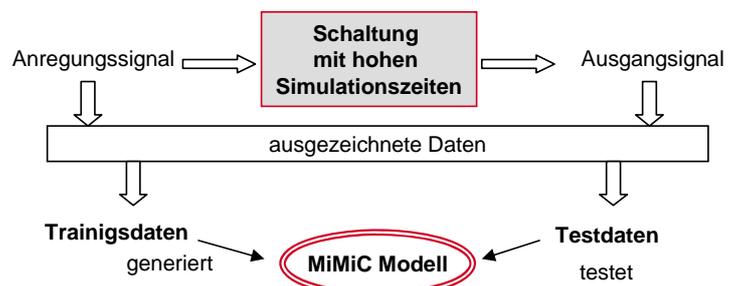
→ Modellierung A/MS Systeme mittels Identifikationsverfahren auf Basis einer hybriden Neurofuzzy- und Support Vector Machine-Methode (NFI+SFM)

Vollautomatisierter Black-Box Ansatz mit folgenden Schritten:

- Selektion der Anregungssignale
- Optimierung der Modellgenerierung durch Verwendung von statistischen unabhängigen Testdaten
- Auswahl der Identifikationsstruktur
- Modellgenerierung und -parametrisierung aus Trainingsdaten

Öffentlich geförderte Projekte :

Anastasia+	2002-2005
Verona	2006-2009



Automotive Electronics

14

AE/EIM3 - Dr.-Ing. R. Dölling | 02.02.2007 | © Alle Rechte bei Robert Bosch GmbH, auch für den Fall von Schutzrechtsanmeldungen. Jede Verfügungsbefugnis, wie Kopier- und Weitergaberecht, bei uns.

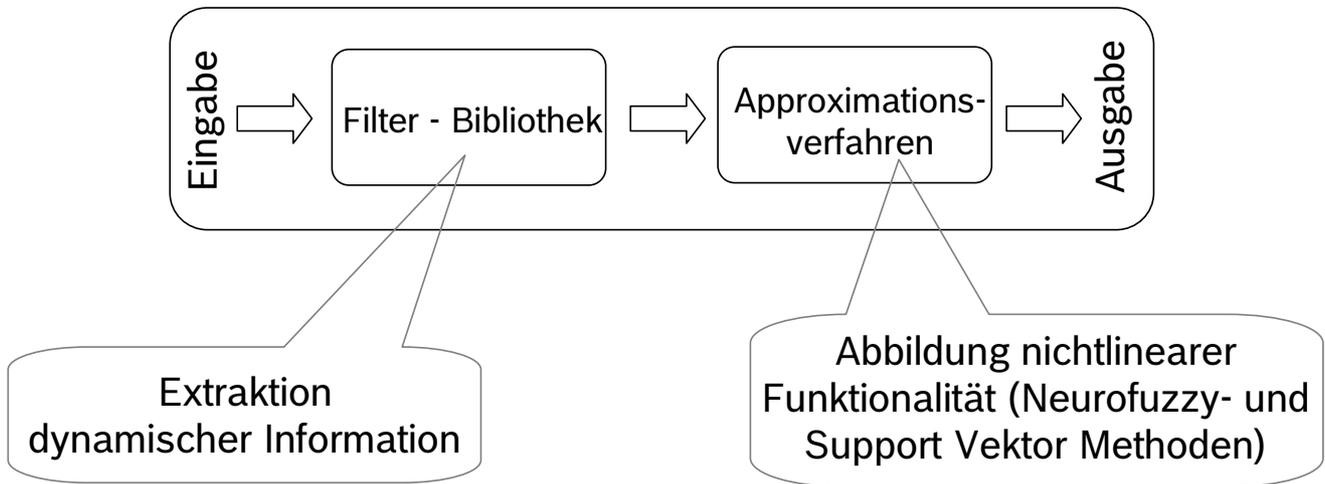


BOSCH

Methodik – Modellarchitektur

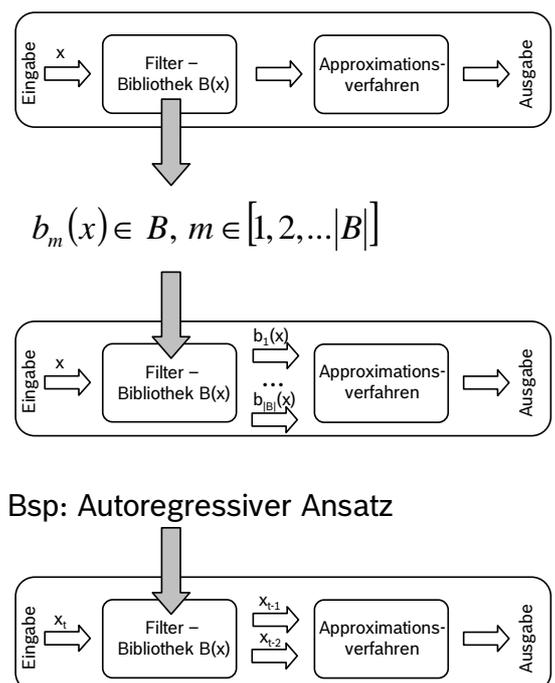
Schaltungen sind dynamische Systeme und besitzen zeitvariante nichtlineare Eigenschaften

⇒ „Externe Dynamik“



Umsetzung – Darstellung dynamischer Information

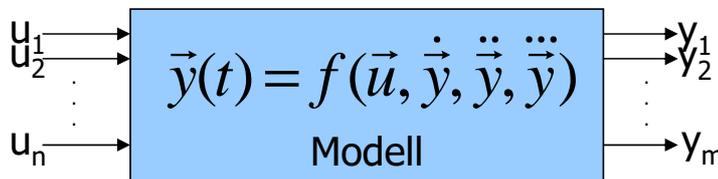
- Kontinuierliche Signale:
 - Klassische Filter
(Laguerre-, Kautz- ...)
 - Frequenzinformation
 - Phaseninformation
 - Ereignisdiskrete Signale:
 - Shannon-Entropie
(variabler Fensterbereich)
- Tapped-Delay Informationen



Bsp: Autoregressiver Ansatz

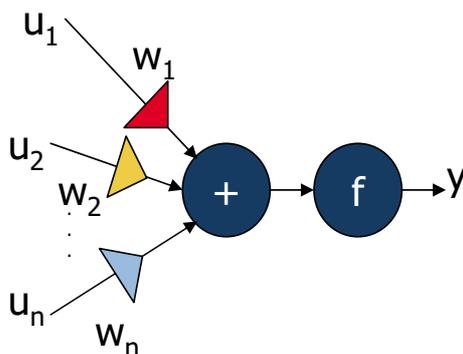


- Identifikation:
 - Extraktion der Eigenschaften einer Schaltung
 - Modellierung der Eigenschaften
- Modelleigenschaften:
 - Anzahl Ein- und Ausgänge
(Multiple Input Multiple Output MIMO)
 - Linearität bzw. Nichtlinearität
 - Statisches oder dynamisches Verhalten unbekannter Ordnung



Neuronale Netze

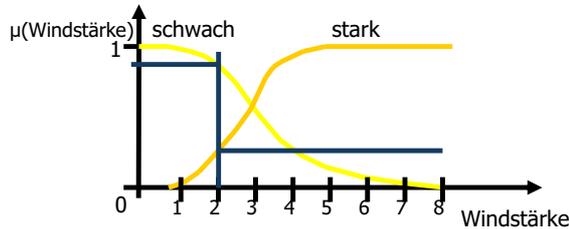
- Neuronale Netze sind gekoppelte künstliche Neuronen
- Künstliches Neuron:
 - Eingänge u_i durch w_i gewichtet
 - Akkumulierung der gewichteten Eingänge
 - Verhaltensbestimmung durch Ausgangsfunktion f (linear, nichtlinear)



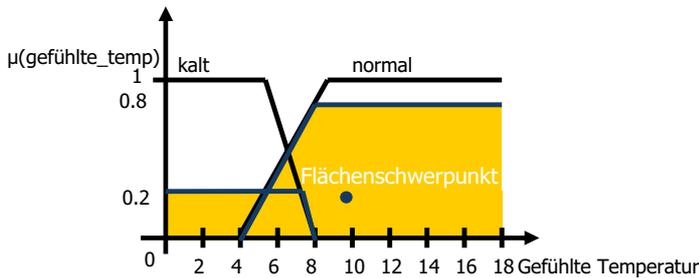
$$y = f \left(\sum_{i=1}^n w_i \cdot u_i \right)$$

Fuzzy-Logic

Windstärke = 2



R2: $\mu(\text{schwach}) = 0.8$
R1: $\mu(\text{stark}) = 0.2$

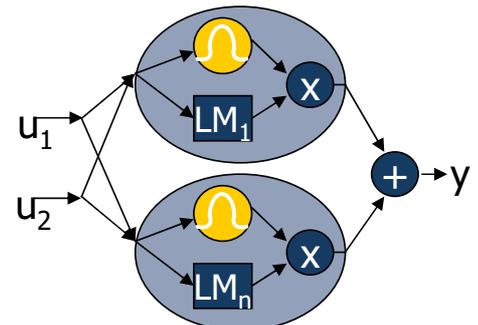


➔ Gefühlte Temperatur = 9.8



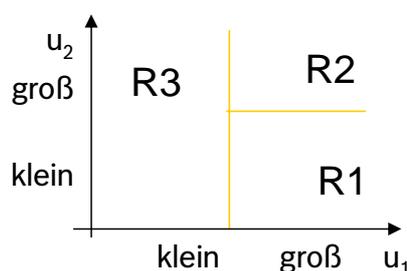
Neuro-Fuzzy Modelle

- ➔ Neuron besteht aus lokalem Modell und Gültigkeitsfunktion
- ➔ Lokales Modell (LM):
 - Lineare Funktionen approximieren Ausgangswerte
- ➔ Gültigkeitsfunktion
 - Eingangssignale werden durch Fuzzy-Logic in Arbeitsbereiche aufgeteilt



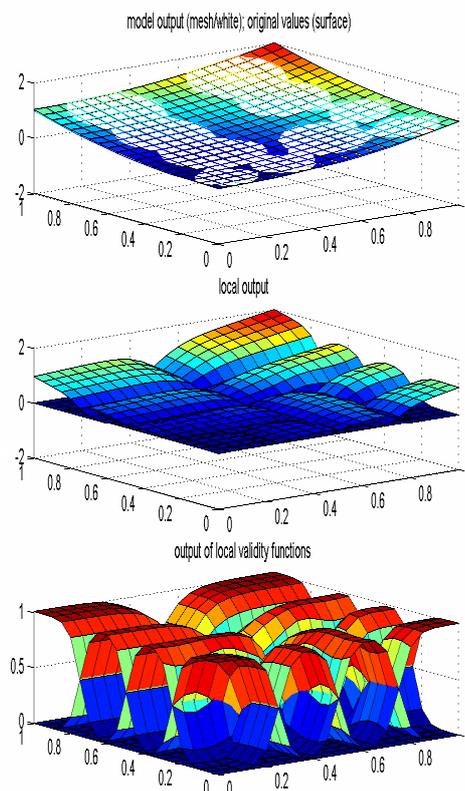
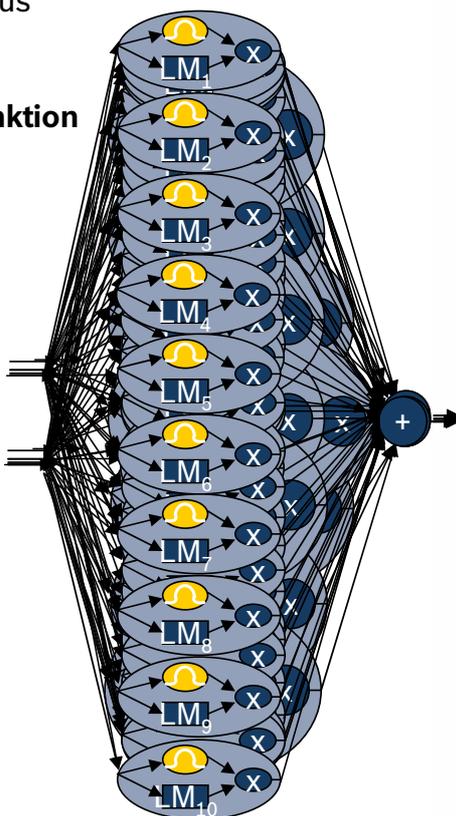
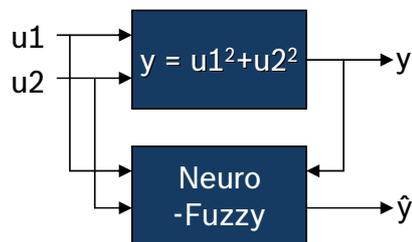
➔ Analogie zur Fuzzy-Logic:

R1: IF $u_1 = \text{groß}$ AND $u_2 = \text{klein}$ THEN $y_1 = b_1 + u_1 * w_{11} + u_2 * w_{12}$
 R2: IF $u_1 = \text{groß}$ AND $u_2 = \text{groß}$ THEN $y_2 = b_2 + u_1 * w_{21} + u_2 * w_{22}$
 R3: IF $u_1 = \text{klein}$ AND $u_2 = \text{d.c.}$ THEN $y_3 = b_3 + u_1 * w_{31} + u_2 * w_{32}$



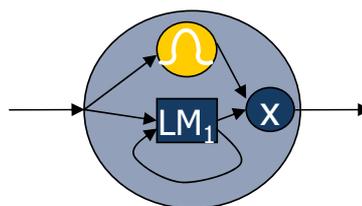
Funktionsweise des Lernalgorithmus

- Beispiel
Approximation der Funktion
 $y = u_1^2 + u_2^2$

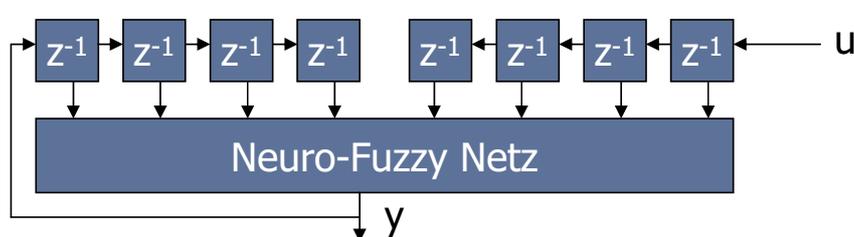


Dynamische Modelle

- Interne Dynamik
Lokale Rückkopplung



- Externe Dynamik
Tapped-Delay-Lines mit Rückkopplung des Ausgangs



Rückkopplungen bringen Instabilität in ein System

➔ Alternativ: Filter als Dynamikelemente

Filter als Dynamikelemente

→ Lineare Filter:

$$y = a_0 F_0(u(k)) + a_1 F_1(F_0(u(k))) + \dots + a_n F_n(\dots)$$

- Filterkoeffizienten a_0 bis a_n durch Least-Squares Algorithmus bestimmbar
- Filterelemente $F(x)$

→ FIR-Filter:

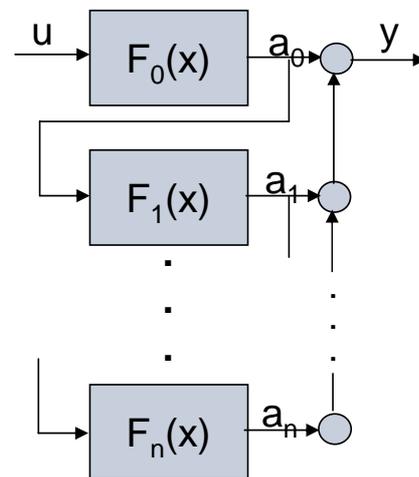
- Filter repräsentiert System mit Pol im Ursprung

→ Laguerre-Filter:

- Filter repräsentiert gedämpftes System
- Reelle Polstelle

→ Kautz-Filter:

- Filter repräsentiert schwingungsfähiges System
- Konjugiert komplexe Polstelle



Umsetzung – Nichtlineare Approximation

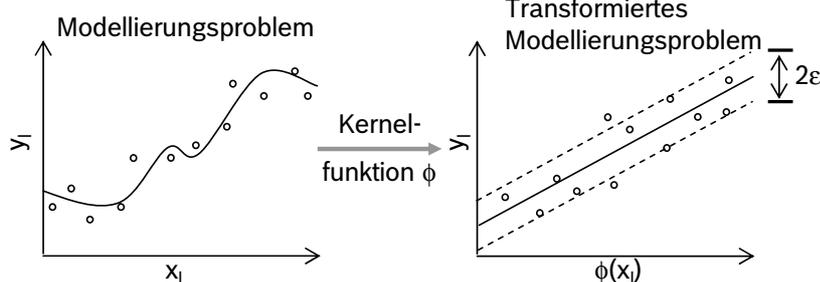
Support Vektor Regression

• l Beobachtungen

$$(x_l, y_l), x_l \in \mathbb{R}^n, y_l \in \mathbb{R}$$

• Lineare Regression

$$f(x) = \langle w, x \rangle + b = y$$



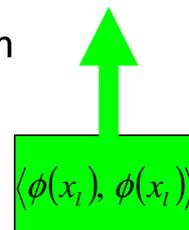
• Regressionsschätzung über konvexes Optimierungsproblem

$$\text{Primal: minimiere } \frac{1}{2} \|w\|^2 \text{ unter } \begin{cases} y_l - \langle w, x_l \rangle - b \leq \epsilon \\ \langle w, x_l \rangle + b - y_l \leq \epsilon \end{cases}$$

• Darstellung in Dualer Form mittels Langrange-Multipliiertor (α)

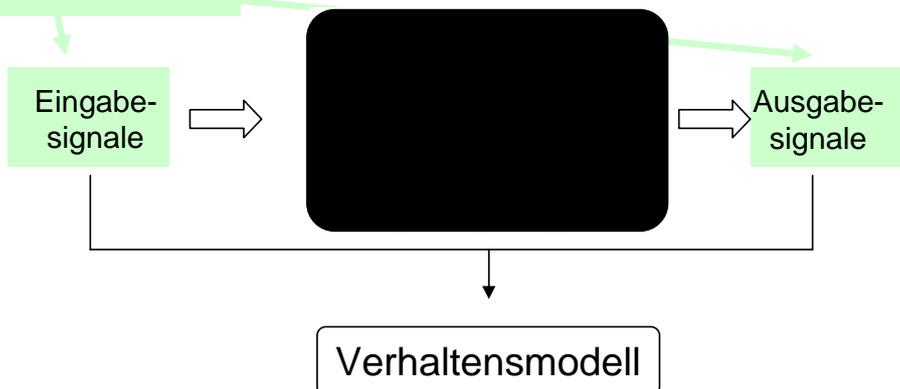
$$w = \sum_l (\alpha_l - \alpha_l^*) \cdot x_l$$

$$f(x) = \sum_l (\alpha_l - \alpha_l^*) \langle x_l, x \rangle + b$$



- Datenbasierte Generierung von Verhaltensmodellen

- Vorgehen:



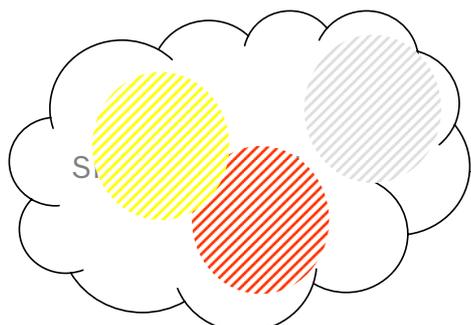
- Abstrakte Modellierung abseits physikalischer Grundlagen

Weitestgehend automatische Modellgenerierung

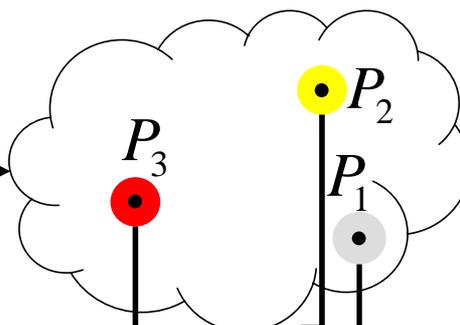
- Verhaltensmodellierung mit ‚einfachen‘ mathematischen Beschreibungen

Beschleunigung benötigter Simulationszeiten

Arbeitsraum der Schaltung



Parameterraum der Signalquellen(n)



Ziel: Abdeckung des Arbeitsraumes für stabile Verhaltensmodellierung

Ansatz: Abdeckung des Arbeitsraumes durch Abdeckung des korrespondierenden Parameterraumes

Stimulidaten	Outputdaten
$sG(P_1)$	out_1
$sG(P_2)$	out_2
$sG(P_3)$	out_3
...	...

$$out_1 = \text{prozess}(sG(P_1))$$

$$out_2 = \text{prozess}(sG(P_2))$$

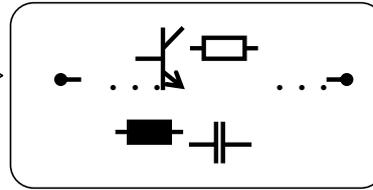
$$out_3 = \text{prozess}(sG(P_3))$$

Spezifikation der Eingangssignale

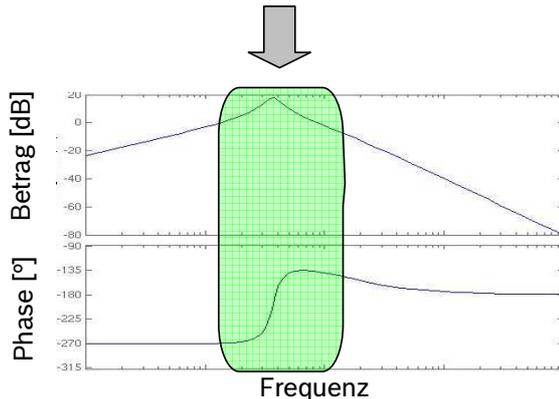
->

Arbeitsraum der Schaltung

Bsp: - sinusoides Signal
 - Parameterintervalle:
 $[f_{\min} \quad f_{\max}]$
 $[A_{\min} \quad A_{\max}]$
 Rauschterm



Begrenzter
Bildbereich



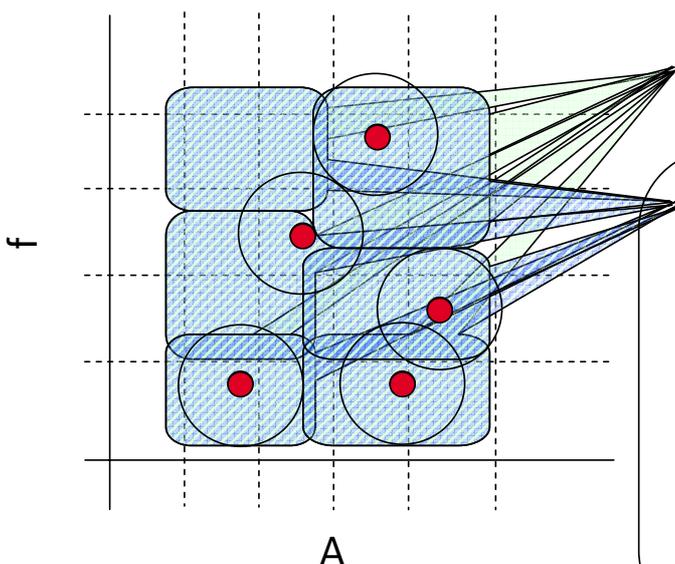
- Transiente Signalverläufe
- Hohe Relevanz bzgl. Modellierungsproblem
- Detektion ‚nichtlinearer‘ Bereiche
- Iterativer Ansatz

Automotive Electronics

27 AE/EIM3 - Dr.-Ing. R. Dölling | 02.02.2007 | © Alle Rechte bei Robert Bosch GmbH, auch für den Fall von Schutzrechtsanmeldungen. Jede Verfügungsbefugnis, wie Kopier- und Weitergaberecht, bei uns.



BOSCH



Initialisierung:

Experimentelles Design für initiale transiente Simulationen

Modellierung:

Ableitung eines Modells (Trainingsdaten)
 Bewertung der Modellgüte (Testdaten)

Detektion kritischer Bereiche:

Verfeinerung des Informationsgehalts
 in Bereichen mit geringer Generalisierung

Detaillierte Simulation:

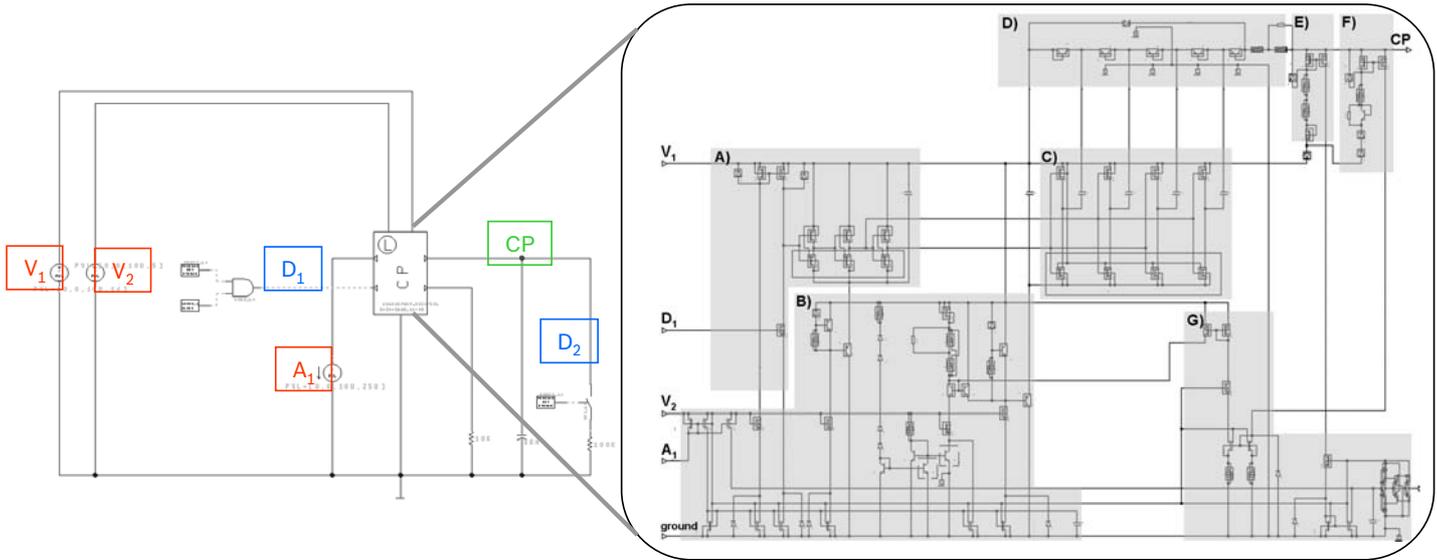
Wiederholen bis Optimum erreicht ist

Automotive Electronics

28 AE/EIM3 - Dr.-Ing. R. Dölling | 02.02.2007 | © Alle Rechte bei Robert Bosch GmbH, auch für den Fall von Schutzrechtsanmeldungen. Jede Verfügungsbefugnis, wie Kopier- und Weitergaberecht, bei uns.



BOSCH



Spezifikation:

Eingänge:

2 x digital (D_1 {0, 3 MHz}, D_2 [0, 3 kHz])

3 x analog (V_1 [0, 12 V], V_2 [0, 6 V], A_1 [0, 10 μ A])

Ausgänge:

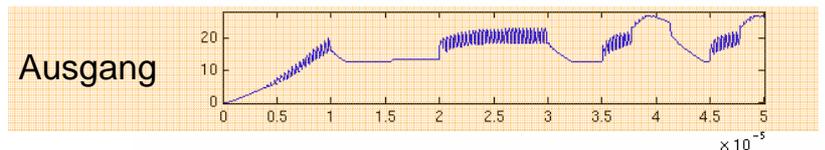
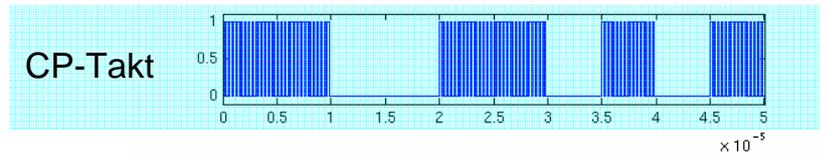
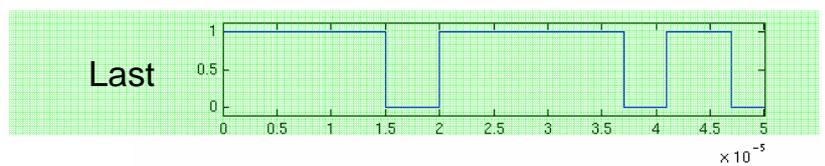
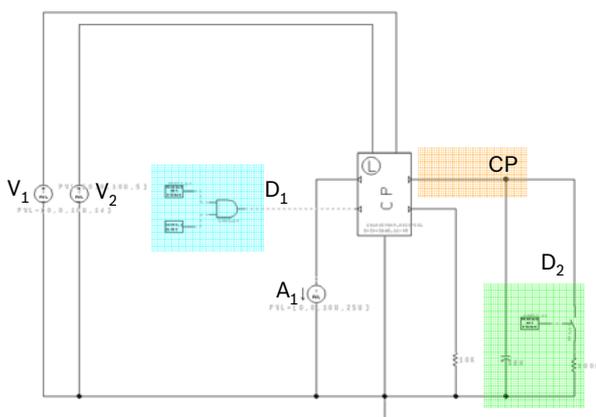
1 x analog CP

Automotive Electronics

29 AE/EIM3 - Dr.-Ing. R. Dölling | 02.02.2007 | © Alle Rechte bei Robert Bosch GmbH, auch für den Fall von Schutzrechtsanmeldungen. Jede Verfügungsbefugnis, wie Kopier- und Weitergaberecht, bei uns.



BOSCH



Ergebnisse:

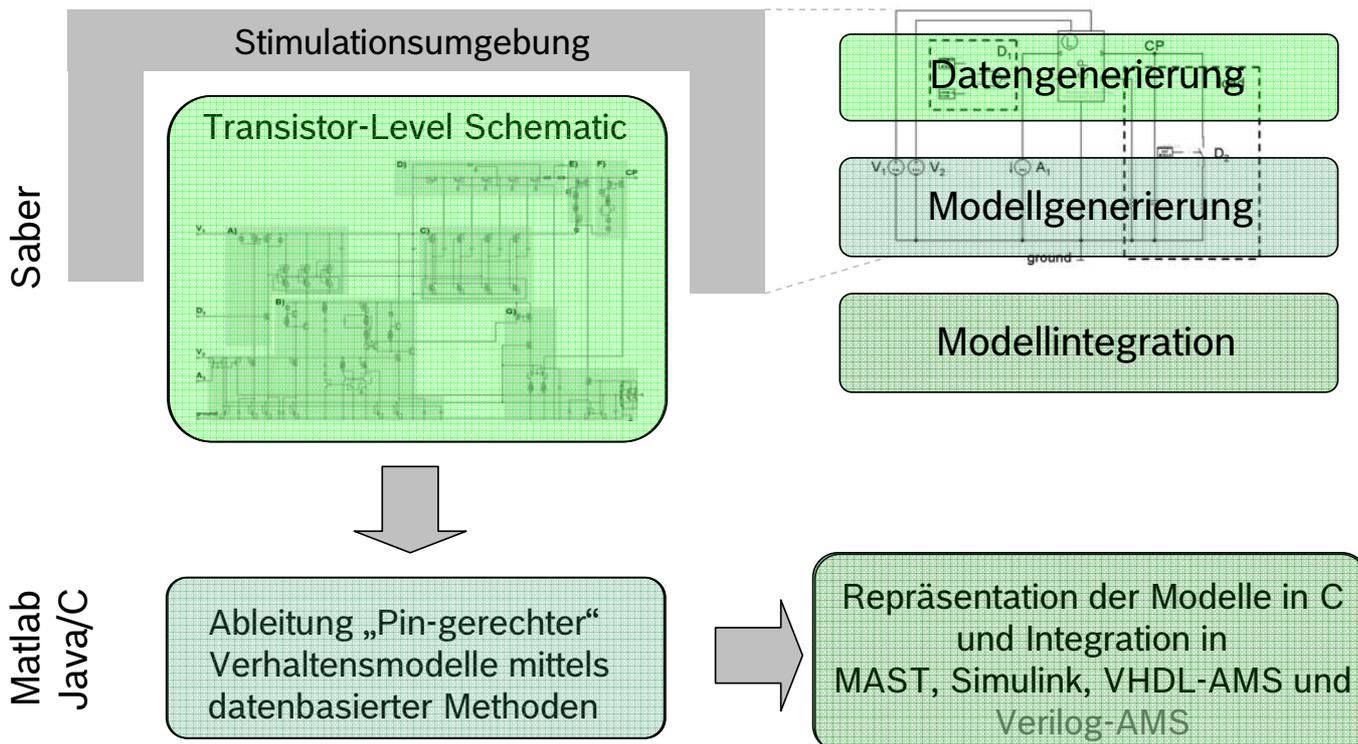
- Modellfehler (mittel) ~ 5%
- Konvergenz nach 30 Iterationen
- *Speed-up* ~ 100
- Integration der Modelle in Simulink, MAST, VHDL-AMS

Automotive Electronics

30 AE/EIM3 - Dr.-Ing. R. Dölling | 02.02.2007 | © Alle Rechte bei Robert Bosch GmbH, auch für den Fall von Schutzrechtsanmeldungen. Jede Verfügungsbefugnis, wie Kopier- und Weitergaberecht, bei uns.



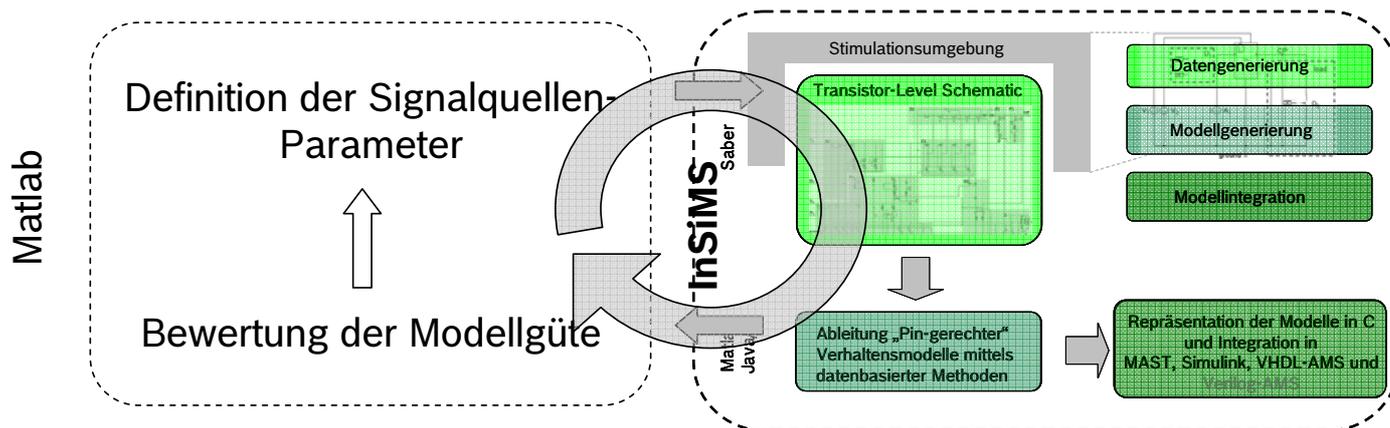
BOSCH



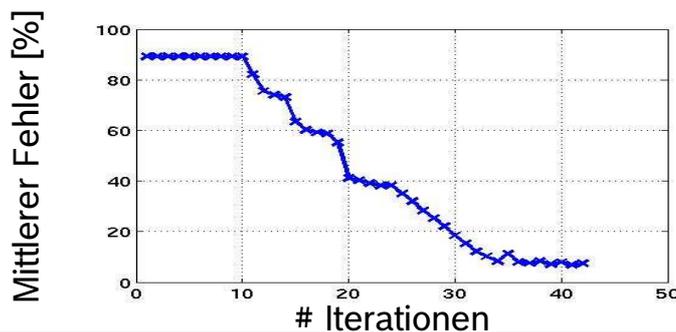
Automotive Electronics



BOSCH



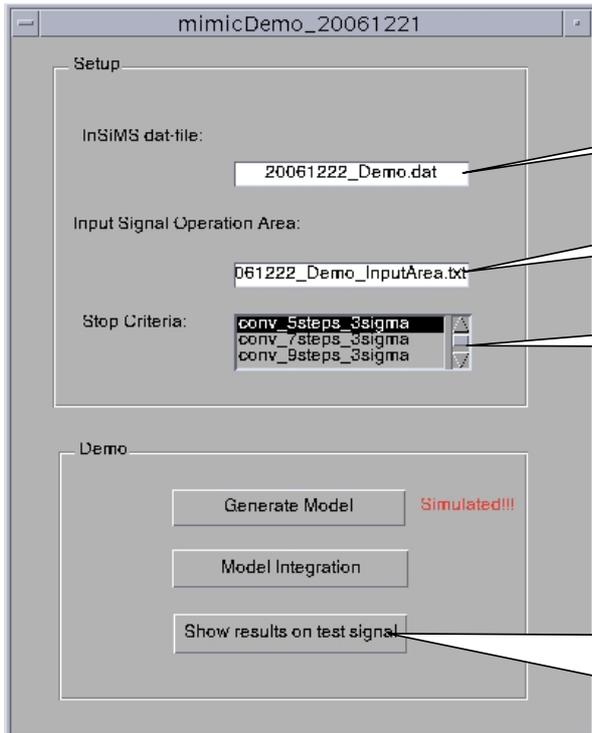
Beispiel



Automotive Electronics



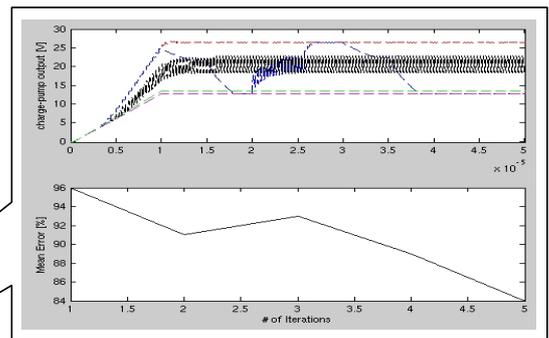
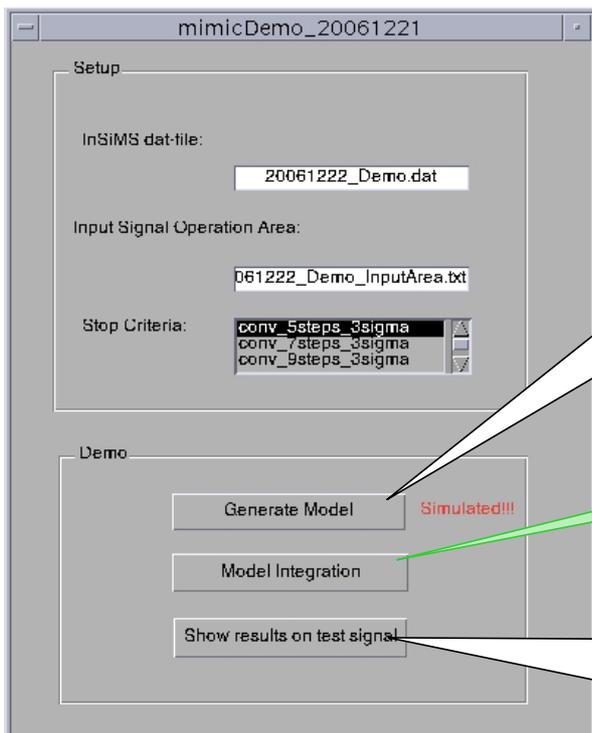
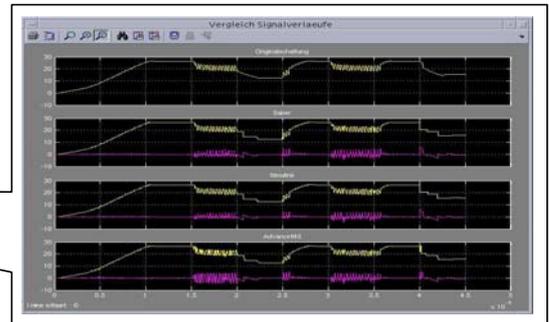
BOSCH



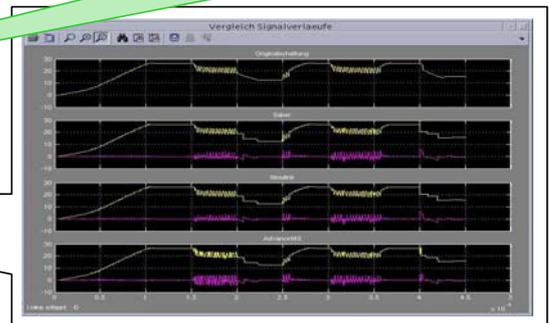
Bereitstellung der Simulationsumgebung

Arbeitsbereich und Typ der Eingangssignale (aus Spec)

Wahl eines Abbruchkriteriums

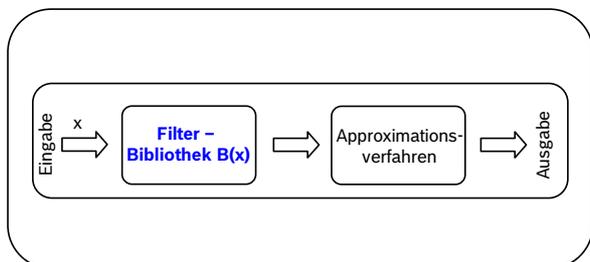


- Integration des C-Modells in Saber, Simulink und AdvanceMS
- Start der Simulationen



Bewertung der Modellstruktur

Filterbibliothek



Modellarchitektur (MiMic)

Ordnung	speed-up	Modellfehler [%]
global	182	16.3
2	164	14.3
4	140	5.3
8	121	2.4

- ➔ Filter werden automatisch bestimmt
- ➔ Modellfehler/speed-up problemspezifisch steuerbar
- ➔ Fehler konvergiert für Modelle höherer Ordnung



Bewertung Simulationszeitgewinn

	Rechenzeit [s]	Modellfehler [%]	Zeitbedarf (automatischer Ablauf)
Originalschaltung	~ 850	-	~ 425 min (30 Simulationen) ~ 190 min Iterative Modell- generierung
Modell Saber	~ 8	~ 5	
Modell Simulink	~ 8	~ 4	
Modell AdvanceMS	~ 6	~ 6	

Speed-up ~ 100x

Fehler ~ 5 %

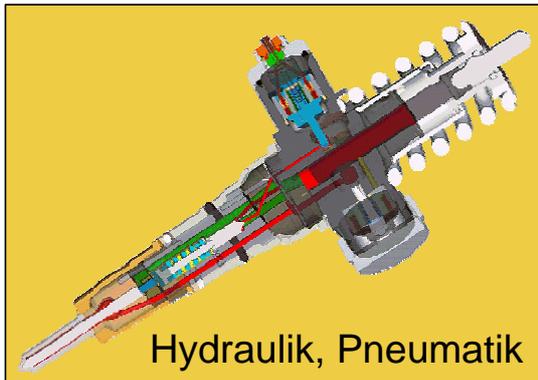
~ 10.2 h



Sensor



digitale elektronik



Hydraulik, Pneumatik

analoge/digitale Elektronik



Automotive Electronics

37 AE/EIM3 - Dr.-Ing. R. Dölling | 02.02.2007 | © Alle Rechte bei Robert Bosch GmbH, auch für den Fall von Schutzrechtsanmeldungen. Jede Verfügungsbefugnis, wie Kopier- und Weitergaberecht, bei uns.

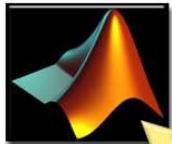


BOSCH

Sensor & Umgebung

digitale Elektronik in VHDL

Matlab/Simulink



ModelSim



InSiMS



AMESim

Sockets



Saber

Hydraulik, Pneumatik

analoge Elektronik in VHDL-AMS / MAST

Nutzen

Entwickler:

- ✓ Zusätzlicher Modellierungsaufwand entfällt
- ✓ Höhere Genauigkeit durch Gesamtsimulation mit exakteren Modelle

Kunde:

- ✓ Höhere Qualität
- ✓ Entwicklungszeitreduktion
 - ✓ geringere Produktkosten
 - ✓ schnellere Reaktionszeiten vom Kundenauftrag bis zur Lieferung

Automotive Electronics

38 AE/EIM3 - Dr.-Ing. R. Dölling | 02.02.2007 | © Alle Rechte bei Robert Bosch GmbH, auch für den Fall von Schutzrechtsanmeldungen. Jede Verfügungsbefugnis, wie Kopier- und Weitergaberecht, bei uns.



BOSCH

□ Motivation / Problem

- Simulation komplexere A/MS-Systeme

□ Problem

- A/MS – Systeme benötigen auf Transistorebene lange Simulationszeiten

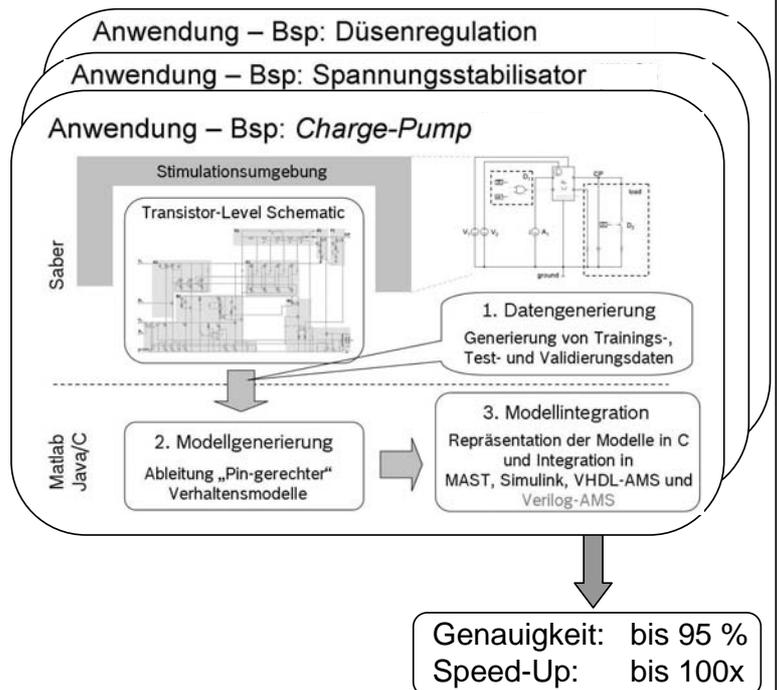
□ Weg

- Simulationszeitreduktion mittels Modellierung auf höherer Abstraktionsebene und Integration der "Pin gerechten" Modelle in der Gesamtsimulation
- Datenbasierte Modellgenerierung auf Basis von Neurofuzzy-und Support Vektor Machine-Methoden (MIMIC⁺-Modelle)

□ Heutiger Stand

- Erste Modellierungsansätze für analoge und gemischt analog-digitale Schaltungen
- Erste Anwendung auf heterogene Systeme (Elektromechanik - Hydraulik- Co-Simulation)
- Flexible Integration der Modelle in Simulink, MAST, VHDL-AMS, Verilog-AMS
- Co-Simulation
- Erreichbare Genauigkeit bis 95%, Simulationszeitgewinn ~ 100x
- Automatische Modellgenerierung mit geringer Nutzerinteraktion

+) Computation of Behav*iour*al* Models for Mixed-Signal Components*

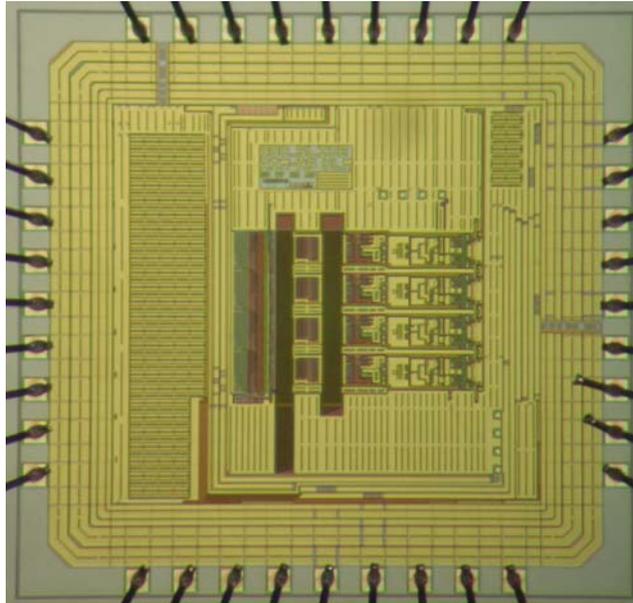


Automotive Electronics



BOSCH

Testchip für einen 10-Bit-ADC



Entwurf:	Hochschule Ulm, Institut für Kommunikationstechnik Bearbeiter: Michael Diebold, Betreuer: Prof. Dipl.-Phys. Gerhard Forster
Layouterstellung:	Hochschule Ulm, Labor Mikroelektronik (Mixed Signal-Entwurf) Analogteil: Full Custom Design Digitalteil: Standardzellen-Entwurf
Technologie:	C35B4C3 0,35 μm CMOS 4 Metal / 2 Poly / HR
Chipfertigung:	Fa. AMS, Österreich, über Europractice
Herstelldatum:	II. Quartal 2006
Kostenträger:	MPC-Gruppe Baden-Württemberg
Chipdaten:	Chipfläche: 2,24 x 2,33 mm^2 Gehäuse: CLCC 44 Funktionsblöcke: Analogteil: Referenzspannungsteiler S/H, Vorverstärker, getakteter Komparator, Autozero Digitalteil: Taktgenerator, SAR
Funktion:	Mit dem Testchip sollten kritische Komponenten eines 10-Bit-ADCs nach dem Sukzessiv-Approximationsverfahren untersucht werden. Der spätere Chip soll einmal aus 16 ADC-Slices bestehen, die im Pipeline-Betrieb angesteuert werden, um einerseits eine Abtastrate von 20 MSa/s und andererseits mehrere Kalibrierzyklen zu erlauben. Der Testchip enthält 4 Slices mit zusätzlichen Diagnoseanschlüssen. Er konnte bereits erfolgreich getestet werden.

Wearlog

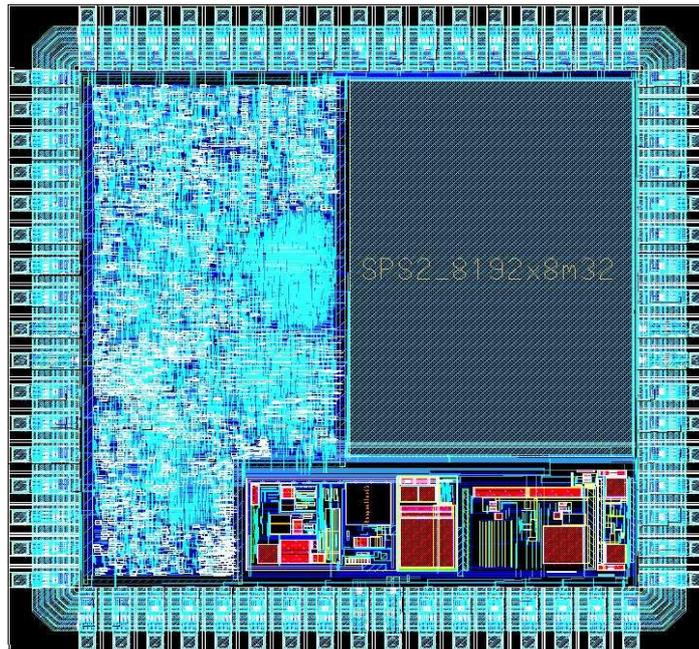


Abbildung 1: Layout auf IC-Station

- Entwurf: Hochschule Offenburg
Bearbeiter: Daniel Bau
Betreuer: Prof. Dr.-Ing. Dirk Jansen
- Layouterstellung: Hochschule Offenburg
(Standardzellenentwurf & analog design)
- Technologie: AMIS 0.35 μm CMOS M-A
- Chipfertigung: Europractice, MPW Run 1449
- Herstelldatum: September 2006
- Kostenträger: MPC-Mittel FH-Verbund Baden-Württemberg
- Chipdaten: Chipgröße: 3.5mm x 3.5mm
Gehäuse: JLCC 68
- Funktion: Wearlog steht für die Messbarkeit von Verschleiß (Wear) und das Aufzeichnen(Log) der gemessenen Daten. Funktional sind Temperaturen, Erschütterungen und Drehzahlen programm- oder zeitgesteuert zu messen und in einem Speicher abrufbar zu hinterlegen. Die gemessenen Daten sind induktiv über eine RFID-Schnittstelle oder kabelgebunden über eine serielle Schnittstelle zugänglich bzw. jederzeit auslesbar.
- Testergebnisse: Das Design wurde zunächst als VHDL-Entwurf konzipiert, auf einem FPGA emuliert, die integrierten analogen Schaltungen simuliert und als ASIC-Design mit Hilfe der Cadence- und Mentor-Software geroutet, über EURO PRACTICE gefertigt. Bei der Inbetriebnahme zeigte sich, dass der integrierte PLL aufgrund eines Layout-Fehlers ein Fehlverhalten aufweist. Die RFID-Einheit und die restlichen integrierten Schaltungen sind so weit wie bisher untersucht funktionsfähig.