

# 25 Jahre Entwurf integrierter Schaltungen

Ulrich Golze

Abteilung Entwurf integrierter Schaltungen (E.I.S.)  
Technische Universität Braunschweig  
Mühlenpfordtstr. 23, 38106 Braunschweig  
golze@eis.cs.tu-bs.de

**Abstract:** Diese Arbeit gibt einen Einblick in die Forschung und Lehre zum Chip- und System-Entwurf in der Informatik der TU Braunschweig seit 1983.

## 1 VLSI-Design in der Informatik von 1983 bis 2011

Unterstützt die Technische Universität Braunschweig die Forschung und Lehre des Entwurfs höchstintegrierter digitaler Schaltungen, kurz: Chips? Für ein „ja“ fanden sich 1983 in Braunschweig kaum Anzeichen. Damals war nach dem Vorbild der amerikanischen Mead-Conway-Bewegung [MC80] vom Bundesminister für Forschung und Technologie (BMFT) das universitäre Verbundprojekt E.I.S. zur Förderung eben dieses VLSI-Designs ausgeschrieben (VLSI steht für Very Large Scale Integration).

Niemand in der Braunschweiger Informatik und noch nicht einmal jemand aus den Ingenieurwissenschaften zeigte Interesse an dieser ungewöhnlich großzügigen Förderung. So reiste ich mehr zufällig und aus eigener Initiative als Privatdozent der theoretischen (!) Informatik nach Bonn, um dort neben gestandenen Technik-Bossen wie Robert Piloty, Wolfgang Händler, Detlef Schmidt, Klaus Waldschmidt, Klaus Müller-Glaser, Joachim Mucha, Otto Manck u.a. für die TU Braunschweig das E.I.S.-Projekt einzuwerben, mit einem warmen Segen an Stellen und Sachmitteln.

Nach dem Aufbau eines VLSI-Curriculums aus Vorlesung, Übung und vor allem einem VLSI-Praktikum (das von 1984 bis 2011 mehr als 2500 Studenten der Informatik, Informations-Systemtechnik und Elektrotechnik erlebt und erlitten haben), wurde 1986 die Abteilung Entwurf integrierter Schaltungen (E.I.S.) gegründet, die ich 25 Jahre lang geleitet habe. Sie ging 2011 nahtlos in die heutige Abteilung c3e (Chair for Circuit-Design and Computer Engineering unter Prof. Berékovic) über.

## 2 Ausgewählte Beispiele aus der Geschichte der Abteilung E.I.S.

Einige typische Forschungsprojekte sollen die Geschichte der Abteilung E.I.S. illustrieren.

## 2.1 Ein Kryptografie-Prozessor für das RSA-Verfahren

Verschlüsselungen bieten Sicherheit im Internet und bei der Kommunikation. Für das RSA-Verfahren als eine der sichersten Verschlüsselungen hat die Abteilung E.I.S. einen extrem schnellen Krypto-Prozessor erfunden und gebaut. Er berechnet beispielsweise die RSA-Operation

$$C = M^D \text{ mod } N$$

mit 600-stelligen binären Operanden *10.000 mal schneller* als ein damaliger normaler Rechner. Ermöglicht wurde diese Effizienz auch durch ein raffiniertes reguläres *Layout* der beteiligten 350.000 Transistoren des Chips.

Die weltweit patentierte Schaltung unseres Studenten Holger Sedlak wurde ab 1984 in einer langjährigen Kooperation mit der SIEMENS AG marktreif entwickelt und vom SIEMENS-Vorstand als „brillianter Erfolg“ eingestuft [Se89, Se91].



Bild 1: Der Krypto-Prozessor auf der Chipkarte erlaubt eine abhörsichere Kommunikation und eine fälschungssichere elektronische Unterschrift

## 2.2 Entwicklung eines RISC-Prozessors

In einer mehrjährigen Kooperation mit der Firma LSI LOGIC wurde ab 1990 die Methodik des Entwurfs großer Semi-Custom-Chips erforscht. Als Beispiel wurde der Entwurf des RISC-Prozessors TOOBSIE ausgewählt (TU Braunschweig Integrated Engine).

Der praktische Entwurf samt Entwurfsmethodik umfasste auch die Entwicklung einer Architektur mit Befehlssatz, eine Performance-Analyse, den Einsatz von Hardware-Beschreibungssprachen, eine graphische Verifikation und Animation in frühen Entwurfsphasen mit STATEMATE, eine Optimierung von Kontrolltransfer-Anweisungen, die Entwicklung von Testmustern auf hohen Entwurfsebenen, den Aufbau eines Evaluati-

ons-systems und den Test des RISC-Prozessors mit nichttrivialen Anwendungsprogrammen unter einem eigens entwickelten Betriebssystem.

Die wichtigsten Merkmale des Prozessors waren: eine 5-stufige Pipeline, 32 bit breiter Datenpfad, Load-Store-Architektur, 32 Mehrzweckregister, nur ein Zyklus für die meisten Befehle, Drei-Adressformat, Delayed-Branch, Delayed-Load, Forwarding, On-Chip-Instruction-Cache mit RIB-Modus, Branch-Target-Instruction-Cache mit spekulativer Ausführung von Befehlen, durchschnittliche Ausführungszeit von 0,6 Zyklen für einen Branch, Kernel- und User-Modus, drei Interrupt-Ebenen, virtuelle Adressierung, synchrones oder asynchrones Busprotokoll, LSI-Gate-Array der Serie LCA 100k, brutto 106 000 Gatteräquivalente, 0.7 $\mu$ -Sea-of-Gates-Technologie, CPI=0,99.

Die Arbeiten resultierten in den Dissertationen von Michael Schäfers [Sc95], Elmar Cochlovius [Co94], Klaus-Peter Wachsmann [Wa94] und Peter Blinzer [Bl00] sowie in mehreren Lehr- und Arbeitsbüchern [Go95a, Go95b, Go96, Go05].

### 2.3 Statechart-basierter Systementwurf: Komplexe Autoradio-Steuerungen



Bild 2: Das „Herz“ des Autoradios wurde in eine Workstation verpflanzt, wo es bequem erprobt und weiterentwickelt werden kann

Die Benutzeroberfläche eines hochkomplexen Autoradios darf den Autofahrer gleichwohl nicht zu sehr ablenken. In Kooperation mit der Fa. BLAUPUNKT hat die Abteilung E.I.S. ab 1995 ein Verfahren zum schnellen Aufbau von Prototypen entwickelt. Wie in Bild 2 wird ein reales Autoradio mit einem Rechner gekoppelt. Während elementare Funktionen wie der Tuner im Radio verbleiben, werden Kontrolle und Oberfläche in Realzeit im Rechner simuliert, weiterentwickelt und optimiert.

Mit Hilfe paralleler hierarchischer Zustandsdiagramme, den *Statecharts*, können auch beim Autoradio-Design Varianten schnell ausprobiert werden. Ziel ist es, daraus mög-

lichst automatisch effiziente Realisierungen in Hardware und Software zu gewinnen. Man spricht vom *Hardware-Software-Codesign*. Erst nachdem so eine Lösung auf abstrakter Ebene gefunden ist, wird diese beispielsweise in eine Kombination aus Spezialchip und programmiertem Mikrokontroller abgebildet. Die Ergebnisse sind in der ausgezeichneten Dissertation von Jean-Claude Ackad festgehalten [Ac00].

## 2.4 Das intelligente Haus

Auf der EXPO 2000 hat die Abteilung E.I.S. in enger Zusammenarbeit mit der Spin-off-Firma Domologic von Gerrit Telkamp das Intelligente Haus der Öffentlichkeit vorgestellt (Bild 3). Dabei wurden sechs Wohnungen seniorenfreundlich mit eigens entwickelter Home-Automation ausgestattet, die beispielsweise über das normale Stromnetz (Power-Line) vernetzt sind. Ziele waren insbesondere eine Erhöhung der Sicherheit (beispielsweise eine Warnung vor offenen Fenstern oder eingeschaltetem Herd beim Verlassen der Wohnung) und eine ressourcensparende Energieüberwachung.



Bild 3: Idee des intelligenten Hauses und Realisierung als EXPO-Objekt

Dies ist nur ein Beispiel von vielen aus den Aktivitäten zur Home-Automation von 1997 bis 2011 (z.B. Dissertation von Helge Böhme [Bö06], [Te05]).

## 2.5 Ausbildung im Chip- und System-Entwurf und multimediales Lernen

Seit 1984 wurden mehr als 2500 Studenten der Informatik, Informations-Systemtechnik und Elektrotechnik in Entwurf, Simulation, Synthese und praktischem Test von Hardware und Hardware-Software-Systemen ausgebildet. Diese umfasste in der Spätphase der Abteilung u.a. folgende Lehrveranstaltungen.

- Eine Drittsemester-Pflichtveranstaltung *Hardware-Software-Systeme* (Klassischer Hardware-Entwurf, Hardware-Software-Codesign, Systementwurf und eingebettete Systeme, programmierbare Logik, Multi-Processor-System-on-Chip (MPSoC),

Register-Transfer-Logik und Logiksynthese, Hardware- und System-Beschreibungssprachen, mit praktischen Übungen am Rechner)

- Eine anschließende Vorlesung mit Übungen *Chip- und System-Entwurf I* (Hardware-Beschreibungssprachen: Modellierungskonzepte, RTL-Entwurf und Logiksynthese; Systementwurf und eingebettete Systeme: Komplexere Beispiele, vollständiger FPGA-Entwurf; System-Entwurfssprache SystemC, Hardware-Software-Codesign, Multi-Processor-System-on-Chip (MPSoC); Test und Testbarkeit; Adaptive Rechner)
- Praktika zum Chip- und System-Entwurf: In mehreren Varianten wurden wechselnde Themen aus aktueller Forschung und Industriekooperation angeboten, beispielsweise Hardware-Software-Codesign mit SystemC (z.B. Echtzeit-Gestenerkennung), Adaptive Rechner (z.B. Bildbearbeitung) oder Home-Automation
- Eine Vertiefungsvorlesung mit Übungen *Chip- und System-Entwurf II* (Transaction-Level-Modellierung (TLM); TLM-Entwurf eingebetteter Systeme: Performance-Analyse, HW-SW-Verifikation; Multi-Processor-System-on-Chip (MPSoC); Kommunikationsmodellierung (Network-on-Chip); Synthese: Layout-Synthese, High-Level-Synthese; Adaptive Compiler

Die klassische Ausbildung in Vorlesung, Übung und Praktikum hat die Abteilung E.I.S. dabei seit 1997 durch multimediale Angebote ergänzt, die Studenten für Studenten in einem Multimedia-Praktikum und anschließenden Arbeiten entwickelt haben (Dissertation von Tamer Çatalakaya [Ça03]).

### 3 Zukunft

Es folgen zwei Beispiele für Arbeiten, die bis zum Ende der Abteilung E.I.S. 2011 durchgeführt wurden und die auch danach weitergeführt werden.

#### 3.1 Adaptive Compiler für Adaptive Computer

Klassische Rechner lassen sich bei Bedarf durch zusätzliche Koprozessoren beschleunigen. In einem Adaptiven Computer erschafft sich ein klassischer Rechner (CPU) für jedes Anwendungsprogramm quasi selbständig einen maßgeschneiderten Koprozessor. Dazu wird eine CPU auf dem gleichen *System-on-Chip* erweitert um einen rekonfigurierbaren Logikbaustein RCU, mit dem grundsätzlich jede logische Schaltung beliebig oft innerhalb von Sekunden geladen werden kann. Hierfür haben wir in mehreren DFG-Projekten zusammen mit der TU Darmstadt einen *Adaptiven Compiler* entwickelt, der ein beliebiges C-Programm automatisch auf die CPU und die RCU aufteilt, indem beispielsweise rechenintensive Schleifen in Hardware ausgelagert und dort hochgradig parallel und schnell und noch dazu viel energiesparender ausgeführt werden (Bild 4).

Die Arbeiten wurden von Andreas Koch initiiert und in seiner ausgezeichneten Dissertation [Ko97] und seiner Habilitationsschrift [Ko04] zusammengefasst. Sie wurden in der Dissertation von Nico Kasprzyk [Ka05] und in der ausgezeichneten Dissertation von Hagen Gädke-Lütjens fortgesetzt [Gäl1] vertieft und werden auch künftig an der TU Darmstadt fortgeführt.

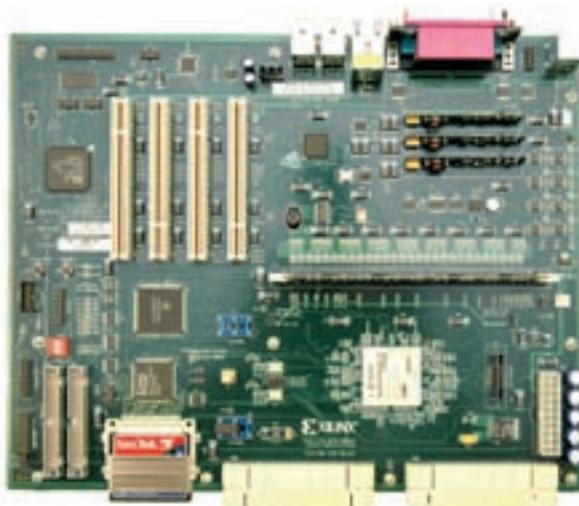


Bild 4: Adaptiver Computer der Abteilung E.I.S.

### 3.2 Entwurfsmethoden für eingebettete Systeme: Interoperabilitäts-Standards für SystemC

Eingebettete Systeme gehören zu den wichtigsten und spannendsten Anwendungen der modernen Informatik. Dabei geht es nicht nur um so bekannte Beispiele wie Handy oder Digitalkamera, sondern im modernen PKW verstecken sich beispielsweise zig vernetzte Computer, und fast jeder technische Gegenstand in Haushalt, Medizin, Produktion, Militär, Transport, Sport und Unterhaltung ist ein Kandidat für einen eingebetteten Chip.

Hierfür sind Systems-on-Chip (SoC) die grundlegende Technologie, bei der Software auf Standard-Computern und Spezial-Hardware wie im vorigen Abschnitt auf einem einzigen Chip untergebracht sind. Mit Milliarden möglicher Transistoren auf einem Chip dürfen die Entwickler jedoch schon längst nicht mehr in Logikgattern denken, sondern müssen auf abstrakter Systemebene, dem *System-Level*, möglichst viele vorgefertigte Hardware- und Software-Bausteine (*Intellectual Property*, IP) einkaufen und diese mit System-Beschreibungssprachen zu einem System-on-Chip zusammensetzen.

Dabei ist es der Traum jedes Designers, die verschiedenen IP-Bausteine wie bei einem LEGO-Baukasten mühelos und rasch zusammensetzen und miteinander kommunizieren zu lassen. Diese *Interoperabilität* genannte wechselseitige Kommunikationsfähigkeit

ist bisher jedoch leider nur ein Traum, die Wirklichkeit deutet Bild 5 an: es gibt fast so viele „Standards“ für Interfaces wie Firmen, die sie anbieten.



Bild 5: Mangelhafte Interoperabilität [Kl08]

Eine bessere Interoperabilität war daher unser Forschungsthema. Kommunikation zwischen IP-Bausteinen in Hardware und Software lässt sich besonders wirkungsvoll mit einer System-Beschreibungssprache modellieren. SystemC ist derzeit die prominenteste solche Sprache. Sie ist eine Erweiterung der objektorientierten Programmiersprache C++ und ermöglicht das Hardware-Software-Codesign kompletter System-on-Chips.

Im Rahmen der GreenSocs-Initiative erarbeiteten wir einen offenen Standard für die Kommunikationsmodellierung mit SystemC. Ziel war eine industrieweit akzeptierte SystemC-Schnittstelle, mit der IP-Cores verschiedener Hersteller unabhängig von deren Kommunikationsprotokollen zu einem System-on-Chip zusammengesetzt werden können.

Die Entwicklung erfolgte in enger Abstimmung mit namhaften Partnern aus Industrie und Forschung. Erste Ergebnisse wurden der Open-SystemC-Initiative (OSCI) Ende 2005 vorgestellt. Im November 2007 wurde der TLM-2.0-Standard für SystemC verabschiedet, der auf unseren Basiskonzepten aufbaute. Darüber hinaus flossen sie in die OCP-Implementierung für SystemC ein.

2008 wurde unsere GreenBus-Open-Source-Bibliothek so modifiziert, dass sie vollständig den Standard TLM-2.0 unterstützt. 2009 wurde das TLM-2.0-Language-Reference-Manual (LRM) als Vorstufe zur IEEE-Standardisierung veröffentlicht, an dessen Entwicklung die Abteilung E.I.S. aktiv mitwirkte.

Unsere Arbeiten zur Interoperabilität wurden von Wolfgang Klingauf initiiert und resultierten in seiner ausgezeichneten Dissertation [Kl08] und einem Best Paper Award [GPK07]. Sie wurden von Robert Günzel erfolgreich fortgeführt [Gü11].

Noch einen Schritt weiter geht Christian Schröder in seiner Dissertation [Sc11]: er untersucht nicht nur die Interoperabilität zwischen technischen Bausteinen, sondern auch zwischen den Entwicklungswerkzeugen verschiedener Hersteller für solche Bausteine.

## 4 Literaturverzeichnis

- [Ac00] Ackad, J.-C.: Optimierte automatische Statechart-Implementierungen im Software- und Hardware-Entwurf eingebetteter Systeme, Dissertation, Abteilung E.I.S., TU Braunschweig 2000 (mit Auszeichnung).
- [Bl00] Blinzer, P.: Möglichkeiten und Methoden der Schaltungssynthese, Dissertation, Abteilung E.I.S., TU Braunschweig 2000.
- [Bö06] Böhme, H.: Virtuelle Java-Maschinen für kleine eingebettete Systeme, Dissertation, Abteilung E.I.S., TU Braunschweig 2006.
- [Ça03] Çatalkaya, T.: Kostengünstige multimediale Lernprogramme zum Chip-Entwurf, Dissertation, Abteilung E.I.S., TU Braunschweig 2003.
- [Co94] Cochlovius, E.: Spezifikation, Analyse und Simulation großer VLSI-Entwürfe mit Statecharts und Activitycharts, Dissertation, Abteilung E.I.S., TU Braunschweig 1994.
- [Go95a] Golze, U.: VLSI-Entwurf eines RISC-Prozessors: Eine Einführung in das Design großer Chips und die Hardware-Beschreibungssprache VERILOG HDL, Vieweg 1995.
- [Go95b] Golze, U. unter Mitarbeit von Blinzer, P., Cochlovius, E., Schäfers, M. und Wachsmann, K.-P.: Der RISC-Prozessor TOOBIE: Ein Hintergrundband für den Entwurfsspezialisten, Vieweg 1995.
- [Go96] Golze, U.: VLSI Chip Design with the Hardware Description Language VERILOG. An Introduction Based on a Large RISC Processor Design, Springer 1996.
- [Go05] Golze, U.: 大型RISC处理器设计:用描述语言Verilog设计VLSI芯片, Beihang University Press, Beijing, China 2005.
- [GPK07] Grosse, D.; Peraza, H.; Klingauf, W.; Drechsler, R.: Measuring the Quality of a SystemC Testbench by using Code Coverage Techniques, Forum on Specification & Design Languages (FDL), Barcelona 2007 (Best paper award).
- [Gä11] Gädke-Lütjens, H.: Dynamic Scheduling in High-Level Compilation for Adaptive Computers, Dissertation, Abteilung E.I.S., TU Braunschweig 2011 (mit Auszeichnung).
- [Gü11] Günzel, R.: Taktgenaue Bus-Simulation mit der Transaction-Level-Modellierung, Dissertation, Abteilung E.I.S., TU Braunschweig 2011.
- [Ka05] Kasprzyk, N.: COMRADE — Ein Hochsprachen-Compiler für Adaptive Computersysteme, Dissertation, Abteilung E.I.S., TU Braunschweig 2005.
- [Kl08] Klingauf, W.: Systematic Transaction-Level Communication Modeling with SystemC, Dissertation, Abteilung E.I.S., TU Braunschweig 2008 (mit Auszeichnung).
- [Ko97] Koch, A.: Regular Datapaths on Field-Programmable Gate Arrays, Dissertation, Abteilung E.I.S., TU Braunschweig 1997 (mit Auszeichnung).
- [Ko04] Koch, A.: Advances in Adaptive Computer Technology, Habilitationsschrift, Abteilung E.I.S., TU Braunschweig 2004.
- [MC80] Mead, C.; Conway L.: Introduction to VLSI Systems, Addison-Wesley 1980.
- [Sc95] Schäfers, M.: Effizienter Entwurf großer RISC-Prozessoren, Dissertation, Abteilung E.I.S., TU Braunschweig 1995.
- [Sc11] Schröder, C.: Konfigurations-Interoperabilität von Hardware-Software-Modellen in SystemC, Dissertation, Abteilung E.I.S., TU Braunschweig 2011.
- [Se89] Sedlak, H.: Der Public-Key-Code Kryptographie-Prozessor - Technische Anforderungen und erkennbare Grenzen von Entwurfssystemen, 4. E.I.S.-Workshop, Bonn 1989.
- [SG91] Sedlak, H.; Golze, U.: Full-Custom-Layout eines Chipkartenprozessors für das RSA-Verfahren der Kryptographie, 5. E.I.S.-Workshop, Dresden 1991.
- [Te05] Telkamp, G.: Home Automation for the Elderly and Disabled, Net-atHome Conference, Nizza 2005.
- [Wa94] Wachsmann, K.-P.: Fehlermodelle für höhere Hardware-Beschreibungen beim Entwurf großer VLSI-Chips, Dissertation, Abteilung E.I.S., TU Braunschweig 1994.