FEST: Funktionale Verifikation von Systemen

Das zweite Clusterforschungsprojekt nach zwei Jahren Projektlaufzeit

Das Projekt FEST hat sich zum Ziel gesetzt, Lösungen zu erforschen, die eine einheitliche Verifikation von SoCs ermöglichen. Hierzu werden – ausgehend von einer Systembeschreibung – bis hinunter zur elektrischen Ebene Methoden und Verfahren erforscht, die vorhandene Verifikationslücken schließen und dadurch ein hohes Verbesserungspotenzial ermöglichen. Die Projektpartner unterstützen mit dem gewonnenen Forschungs-Know-how die Halbleiter-Industrie in Deutschland mit dem Ziel, die deutsche Kompetenz auf diesem Gebiet auch in Zukunft auf höchstem Niveau zu halten. Eine noch weiter gesteigerte Qualität in der Verifikationsmethodik ermöglicht es der deutschen Industrie, Risiken von Entwurfsfehlern zu verringern und den dazu nötigen Aufwand zu minimieren. Die Verifikation – eine Schlüsselkomponente des SoC-Entwurfs – wird mit diesem Forschungsvorhaben nachhaltig gestärkt und verhilft der Industrie zu einer nachhaltigen Stärkung der Wettbewerbssituation.

Motivation

Durch eine permanente Reduzierung des Stromverbrauchs sowie durch neue technologische Integrationstechniken erleben wir stetig neue Anwendungsmöglichkeiten und eine beispiellose Miniaturisierung von elektronischen Produkten: Bis heute wird von Mikroelektronik und mikroelektronischen Systemen gesprochen, obwohl es sich längst um Nanoelektronik und nanoelektronische Systeme handelt. Mit ihren analogen Ein- und Ausgängen ermöglichen sie die Kommunikation mit dem Menschen, also zu nicht-elektrischen Signalen sowie die effiziente Übertragung durch Funk. Durch ihre digitalen Prozessoren und Datenspeicher verwirklichen sie eine leistungsfähige Datenverarbeitung, z.B. sind neueste Mobilfunkgeräte in der Lage Videos abzuspielen. Allein die Nanoelektronik bietet das Potenzial, um weiterhin den Leistungshunger bei Multimedia- und Spiel-Applikationen zu befriedigen. Diese nanoelektronischen Systeme sind hochintegrierte Bausteine und werden auch als „Systeme auf einem Chip“ (System on a Chip, SoC) bezeichnet. Die Miniaturisierung dieser SoCs bringt zwar viele Vorteile mit sich, aber auch große Herausforderungen:

- Hat man noch die Fähigkeiten Produkte in neuen Technologien fehlerfrei zu entwerfen?
- Ist die Größe der Entwurfsteams für SoC-Entwürfe noch ausreichend?
- Wie lange dauert es, ein System bis zur Marktreife ohne Fehler zu entwerfen?
- Kann die nationale Industrie noch die Produkte entwerfen, die zur Absicherung wichtiger deutscher Märkte notwendig sind?

Der Benutzer erwartet von diesen Produkten, dass alle Funktionen zuverlässig ausgeführt werden. Diese Anforderungen nach Zuverlässigkeit und Sicherheit stehen der wachsenden Komplexität nanoelektronischer Systeme gegenüber. Die Halbleiter-Industrie bewegt sich damit in einem Bereich, der sich durch höchste Anforderungen, Komplexität, Kurzlebigkeit der Produkte und extremen Kostendruck auszeichnet. Um sich hier zu behaupten, ist ein zuverlässiger Entwurfsprozess eine entscheidende Voraussetzung. Dieser zuverlässige Entwurfsprozess kann aber nur zur Verfügung stehen, wenn die Anstrengungen, die zur Qualitätssicherung unternommen werden müssen, in Zukunft noch weiter gesteigert werden.

Viele lukrative Märkte haben sehr hohe Anforderungen an die Zuverlässigkeit

Die deutsche Industrie hat sich in Märkten etabliert, die sehr hohe Anforderungen an die Zuverlässigkeit stellen. Exemplarisch sind hier zwei Märkte beschrieben. In der Automobiltechnik erfolgen die meisten Innovationen schon jetzt nur in Verbindung mit der Nanoelektronik. Hier sind die Bereiche Infotainment, fahrerunterstützende Systeme, Zuverlässigkeit und Sicherheit hervorzuheben. Der Industrie ermöglichen diese Innovationen sich insbesondere im Premiumsegment von anderen Wettbewerbern zu differenzieren. Die Selbstverpflichtung der europäischen Mitgliedsstaaten zur Halbierung der Anzahl der Unfalltoten wird die Anwendungen von SoC-Produkten beschleunigen, um der Anforderung nach mehr Sicherheit im und am Auto nachkommen zu können. So hat laut Statistischen Bundesamt ein falscher Reifendruck im Jahre 2002 rund 25 Prozent der Verkehrsunfälle verursacht. Elektronische Reifendruck-Sensoren können hier zusätzlich Sicherheit schaffen. Ein von einer deutschen Firma beherrschter Markt sind Sicherheits- und Smartcards-ICs. Hier dominiert Infineon den weltweiten Markt, gefolgt von Philips und STM. Das Marktvolumen ist zwar nicht groß, solche ICs sind aber in vielen Massenprodukten zu finden. Beispiele sind intelligente Kreditkarten, Mobilfunktelefon und elektronische Schlüsselsysteme. Durch den Zuwachs an mobilen Anwendungen und der schnellen und sicheren Authentifizierung wird die Bedeutung von Sicherheit-ICs in Zukunft stark wachsen und eine Schlüsselstellung beim Entwurf von komplexen Sicherheitsanwendungen werden.

Förderkennzeichen:
01M3972

Laufzeit:
1.7.2004-30.6.2007

Projektpartner:
- TU Darmstadt
- TU Ilmenau
- TU Kaiserslautern
- Universität Freiburg
- Universität Frankfurt a. M.
- Universität Tübingen

Die Autoren dieses Berichts sind:
- Volker Schöber
- Wolfgang Fengler
- Horst Salzwedel
- Alexander Pacholik
- Thomas Kropf
- Jürgen Ruf
- Stefan Lämmermann
- Martin Schickel
- Volker Nimbler
- Martin Braun
- Hans Eveking
- Bernd Becker
- Christoph Scholl
- Minh Nguyen
- Markus Wedler
- Dominik Stoffel
- Wolfgang Kunz
- Alexander Jesser
- Lars Hedrich

Zuverlässigkeit ist eine Tugend zukünftiger SoCs

Die Anforderungen an Sicherheit und Zuverlässigkeit steigen, wenn nanoelektronische Produkte für immer neue Anwendungen im Auto eingesetzt werden. In diesem Zusammenhang ist eine der großen Schwachstellen beim Schaltungsentwurf die Überprüfung der implementierten Funktionalität der komplexen Systeme. Als prominentes Beispiel einer Verifikationslücke beim Chip-Entwurf kann der als Pentium-Bug bezeichnete Fehler der Firma Intel genannt werden. Die Firma hatte einen massiven Gewinneinbruch in dem Jahr, in dem ein eklatanter Fehler in bereits ausgelieferten Pentium-Prozessoren festgestellt wurde, der auf eine Verifikationslücke zurückzuführen war (siehe Abbildung 1.03). Dieses steht aber nur als Synonym für eine in der Technik bekannte Problematik von unzulänglich implementierten oder unzureichenden Spezifikationen. Sie werden über Errata-Listen den Kunden mitgeteilt, die die Auswirkungen der Fehler umgehen müssen.

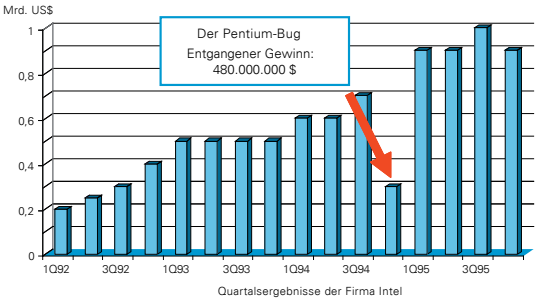

Abbildung 1.03: Ein Verifikationsproblem bei der Entwicklung des Pentiumprozessors verursachte einen entgangenen Gewinn von ca. 500 Millionen Dollar (Quelle: Eveking, TU-Darmstadt)

Neue Anforderungen an den Entwurfsprozess

Simulationstechniken reichen schon lange nicht mehr aus, um Entwurfsfehler wie den Pentium-Bug aufzudecken. Eine Simulation prüft exemplarisch mit ausgewählten Daten, ob beispielsweise eine arithmetische Operation ausgeführt werden kann. Alle möglichen arithmetischen Operationen in modernen Schaltungen auszuführen hieße, dass die Simulationen Monate oder gar Jahre benötigen, was inakzeptabel ist. Eine Verifikation hingegen kann prüfen, ob in allen möglichen Fällen die arithmetische Operation richtige Ergebnisse liefert. Sie kann darüber hinaus Gegenbeispiele liefern, an welcher Stelle der Schaltungsentwurf versagt. Solche Berechnungen können schon nach wenigen Minuten oder Stunden abgeschlossen sein. Dies verdeutlicht, dass die Verifikation eine entscheidende Komponente beim Schaltungsentwurf ist, deren Auswirkungen über Verlust oder Gewinn einer Firma entscheiden. Im Projekt VALSE-XT zeigte ein Vergleich mit einem simulationsbasierten Ansatz die Überlegenheit der formalen Methodik, da die Steigerung der Entwurfsqualität mit einem geringeren und besser vorhersagbaren Verifikationsaufwand einhergeht (siehe Abbildung 1.04).

Die Komplexität des Entwurfsprozesses – bedingt durch die Miniaturisierung der nanoelektronischen

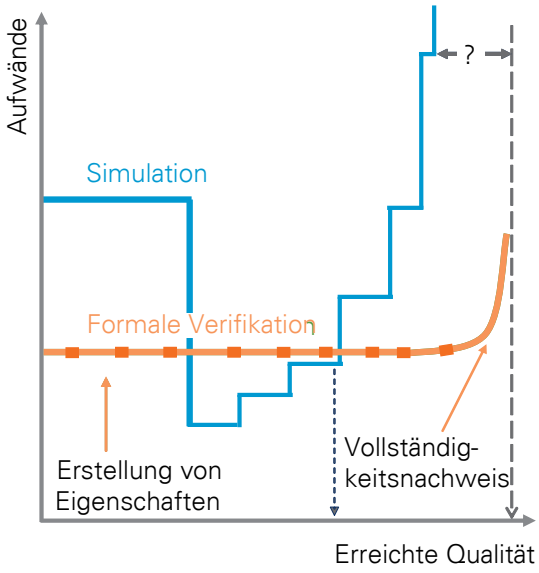

Abbildung 1.04: Komplexitätsproblem der simulationsbasierten Verifikation im Vergleich zur formalen Verifikation (Quelle: VALSE-XT)ew

Strukturen – wächst stark an. Neben der oft fehlenden Kompetenz, komplexe Systeme überhaupt entwerfen zu können, werden die steigenden Kosten für den SoC-Entwurf zu einem weiteren großen Problem. Hinzu kommt, dass die Einmal-Kosten zur Produktionsvorbereitung bei neuen Technologien einen großen Kostenanteil bei der Produktentwicklung ausmachen. Sie wachsen mit dem Einsatz neuester Lithographieverfahren so stark an, dass ein Re-Design des Produkts zusammen mit der zusätzlichen Lieferverzögerung ein großes finanzielles Risiko bedeutet. Es ist daher bereits beim ersten Entwurf von entscheidender Bedeutung, jeden Fehler zu vermeiden, der ein Re-Design erfordern würde (First-Time-Right). Notwendig dazu sind neue und umfassendere Lösungen in allen Bereichen der Verifikation. Diese Problematik wird noch dadurch verschärft, dass in den oben beschriebenen Märkten höchste Anforderungen an die Entwurfsqualität der Nanoelektronik gestellt werden: Die Verifikation entwickelt sich zur Achillesverse des SoC-Entwurfs. Gesteigert wird dieses Problem dadurch, dass die Software zu einem unzuverlässigen Verhalten bei Funktionen beiträgt. Die Erforschung einer gemeinsamen Verifikationsmethodik für Hardware und Software stellt die größte Herausforderung in der Verifikation in den nächsten Jahren dar.

Das Projekt FEST im Überblick

Eine systematische und methodische Vorgehensweise zur Verifikation von der System-Ebene bis zur elektrischen Ebene fehlt bis heute. Das Projekt FEST will die Lücke schließen und neue Ansätze zur Verifikation von Systemen erforschen und die Integration in einem Gesamtsystem erproben. Es arbeiten dabei 6 Universitäten im Projekt zusammen, um über die eigenen Verifikationskompetenzen hinaus eine Vernetzung der unterschiedlichen Forschungsergebnisse zu erreichen. Hierzu werden neue Verifikationsansätze auf ihre Wirksamkeit erforscht: ausgehend von Beschreibungen der Systemebene über Modelle der Architektur- und

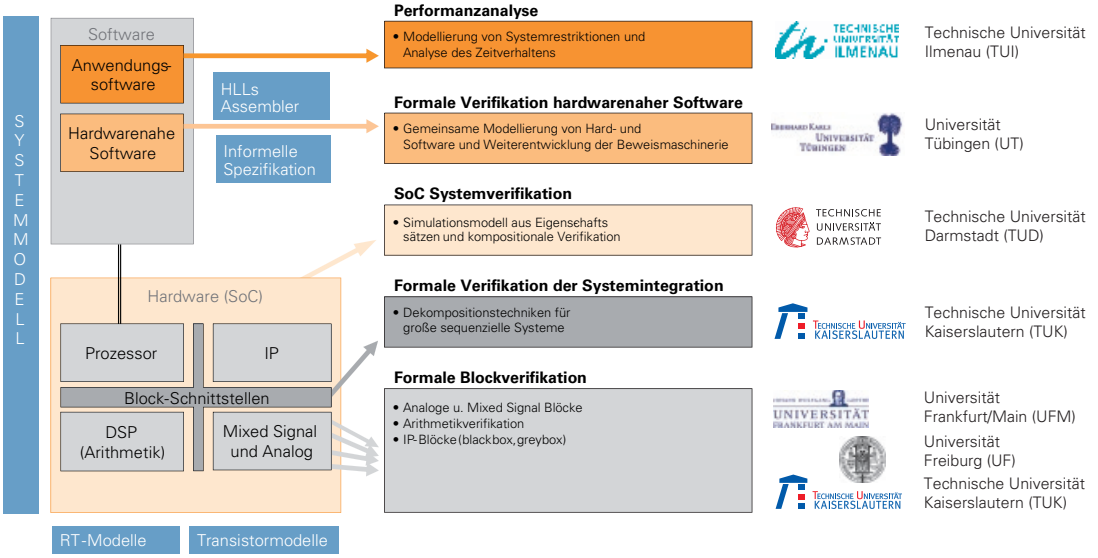

Abbildung 1.05: Einordnung der Verifikationsschritte und Verifikationsmodelle in die Abstraktionsebenen

Register-Transfer-Ebene bis hin zur elektrischen Schaltungsebene, wobei Komplexitätsgrenzen realer Schaltungsgrößen berücksichtigt werden. Abbildung 1.05 zeigt schematisch auf welche Abstraktionsebenen sich die oben beschriebenen Untersuchungen konzentrieren. Hier wird deutlich, dass auf allen Ebenen eine Modellierung notwendig ist, um Algorithmen und Verfahren zur Verifikation anzuwenden. Diese beiden Schwerpunkte werden in diesem Projekt eine herausragende Rolle spielen, um existierende Ansätze zu vernetzen.

Auf Systemebene werden von der TU Ilmenau Methoden zur Modellierung von Zeiteigenschaften – auch Zeitverifikation genannt – erforscht, die das Verhalten des gesamten Systems überprüfen können, bevor eine Implementierung erfolgt. Ein Ziel dieses Ansatzes ist, die Anzahl der Backtracks im Entwurfsprozess in der Implementierung der Halbleiterbausteine zu halbieren. Ein weiteres Ziel sowohl auf Systemebene als auch auf Architekturebene wird von der Uni Tübingen erforscht. Hier wird ein einheitlicher Ansatz für HW-SW-Verifikation von Systemen zu erforscht, der die gemeinsame Behandlung von Soft- und Hardwarekomponenten erlaubt und die bis heute dominierende getrennte Behandlung ablösen wird. Die TU Darmstadt erforscht die Verifikation auf Architekturebene mit einem Verfahren, welches die Systemeigenschaften in einem kompositionalen Verifikationsprozess mit Hilfe individueller Blockeigenschaften prüft. Die Uni Freiburg erforscht Eigenschaftsprüfungen zur Erweiterung der Blockverifikation, so dass eine Technik noch in Szenarien anwendbar ist, wo die einfache Blockspezifikation noch unvollständig ist (Black- und Grey-Boxes). Die gewonnenen Techniken werden genutzt, um Gegenbeispiele zu generieren und Fehler bei der Eigenschaftsprüfung zu lokalisieren. Zur Verbesserung erforscht die Uni Kaiserslautern die Modellgenerierung im so genannten Front-End, um dadurch die Leistungsfähigkeit der Verifikationsmethodik für digitale Blöcke deutlich zu steigern. Bei der Lösung von pathologischen Fällen der Verifikation sequentieller Schaltungen und bei Arithmetikblöcken, wird eine Effizienzsteigerung um eine Größenordnung angestrebt. Auch werden Verifikationen erstmals möglich, bei denen heutige EDA-Werkzeuge und -Methoden noch scheitern. Für gemischt analog-digitale Schaltungen erforscht die Uni Frankfurt eine Methodik zum Mixed-Signal Model-Checking. Diese wird in der Lage sein, Toleranzen der Parameter des Analogteils der Schaltung zu berücksichtigen und in einer digitalen Verifikationsumgebung einzusetzen, damit SoCs mit digitalen und analogen Blöcken verifizierbar sind.

Im Folgenden werden die Ergebnisse der Forschungspartner und deren Einbindung ins Gesamtprojekt dargestellt.

Zeitverifikation auf Systemebene

Um die gesteigerte Komplexität elektronischer Systeme bewältigen zu können und Redesign Zyklen zu minimieren, werden Verfahren zur Evaluierung auf Systemebene erforscht. Diese Verfahren beinhalten die effektive funktionale Simulation auf abstraktem Niveau sowie die Modellierung und Simulation von Performance und Zuverlässigkeit, siehe Abb.4. Beim funktionalen Entwurf (Functional Level) existieren verschiedene Modellierungsdomänen, genannt seien hier Discrete Event, Finite State Machine und Synchronous Dataflow, die innerhalb eines Systemmodells vorkommen und verschiedene Systemaspekte adressieren. Im Bereich der Validierung auf Missionsebene und Electronic System Level (ESL) durch Simulation existieren zahlreiche Forschungen seitens der TU Ilmenau [1] [2]. Zu erwähnen sei hierbei vor allem das Werkzeug MLDesigner. Als weiteres Forschungsfeld ist die Petri-Netz-basierte Modellierungsanalyse von eingebetteten Systemen zu erwähnen, die eine Prüfung von Zeiteigenschaften erlaubt [3] [4].

Kont@kt:
Alexander Pacholik
Technische Universität Ilmenau,
Fachgebiet Rechnerarchitektur
Postfach 100565,
D-98684 Ilmenau
fon: 03677 69-1221,
alexander.pacholik@tu-ilmenau.de

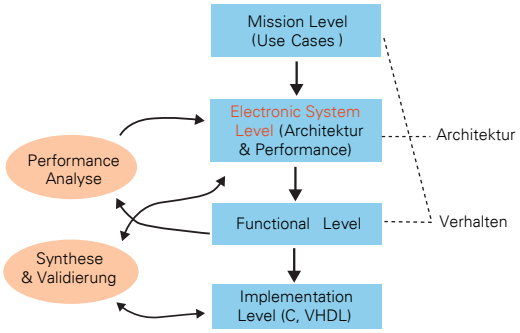

Abbildung 1.06: Mission Level Designmethodik

Bei abstrakten funktionalen Multidomänenmodellen besteht die Problematik der Integration von Zeiteigenschaften, da einige Domänen nicht zeitbehaftet sind oder spezielle Konstrukte zu deren Integration benutzt werden [5]. Im Projekt FEST werden die einheitliche Notation von Zeiteigenschaften in Multidomänenmodellen sowie Methoden der Extraktion und Prüfung des Zeiteigenschaftsmodells untersucht. Zusätzlich werden Aspekte der Dekomposition von Modell und Zeiteigenschaften betrachtet. Für die verschiedenen Modellierungsdomänen wurden Anforderungen an die Constraint-Beschreibung definiert und in Prototypen umgesetzt. Die Forschung ist hier allerdings noch nicht abgeschlossen. Zur einheitlichen Beschreibung wurden Transformationsmöglichkeiten untersucht und für eine softwaretechnische Umsetzung aufbereitet. Zur Verfeinerungen von Constraints wurden Regeln definiert und Prüfmöglichkeiten betrachtet.

Kooperation bestehen im Projekt mit der Uni Tübingen, deren Forschungsschwerpunkt in der Erforschung von Verfahren und Tools zum Model Checking besteht. Ziel der Kooperation ist der Austausch von Modellen und Methoden (Tools). Eine weitere Kooperation zum Austausch von Modellen insbesondere unter Aspekten der Zeitbeschreibung und -verifikation besteht mit der Uni Frankfurt.

Nach Abschluss des Projekts wird die entwickelte Methodik (siehe Abbildung 1.06) geeignet sein, ein formales Zeitmodell aus einem Constraint-annotierten funktionalen Multidomänenmodell zu extrahieren, wobei für das funktionale Modell bestimmte Beschränkungen in Bezug auf Domänen und Elementen einzuhalten sind. In dem formalen Modell sind vorgegebene Zeitbeschränkungen auf Systemebene prüfbar. Weiterhin wird eine Konsistenzprüfung der Constraints zwischen bestimmten Hierarchieebenen ermöglicht.

Methoden und Tools zur Hardware/ Software Co-Verifikation

Die Verifikation hardwarenaher eingebetteter Softwarekomponenten soll durch Anpassung und Weiterentwicklung existierender Verifikationstechniken aus dem Bereich der Hardwareverifikation ermöglicht werden (siehe Abbildung 1.07). Zudem werden verschiedene formale und semiformale Techniken kombiniert und Teile der Algorithmen parallelisiert, um Systeme mit größeren Zustandsräumen analysieren zu können. Ein weiterer Schwerpunkt liegt auf der automatischen Extraktion von Verifikationseigenschaften aus Modellen auf hohen Abstraktionsebenen. Diese aus UML/SysML gewonnenen Systemeigenschaften werden dann in der HW-SW-Verifikation auf niedrigeren Abstraktionsniveaus wiederverwendet. Außerdem finden sie ihre Anwendung bei der Bestimmung von Überdeckungsmaßen bei semiformalen Techniken.

Die zentrale Idee bei der Parallelisierung der Zustandsraum-Traversierung besteht in der Verteilung der Berechnung von Folgezuständen (Image Computation) auf mehrere Knoten eines Clusterrechners. Ein Einsatz in Grid-Umgebungen soll mit der Firma Bosch evaluiert werden. Die Knoten kommunizieren untereinander mittels des Message Passing Interface (MPI). Die Überlappungen der verteilten Zustandsmengen müssen minimiert werden, um die Verfahren effizient einsetzen zu können. Folgearbeiten konzentrierten sich auf diesen Punkt, um diese Überlappung statisch und dynamisch zu reduzieren [6] [7] [8]. Eine neue Guiding Technik wurde implementiert, welche automatisch die Menge der „interessanten" Variablen im Design mit Hilfe der spezifizierten Eigenschaften findet. Diese Variablen dienen zur Steuerung der Zustandsraumexplosion. Das Verfahren beschleunigt den gesamten Verifikationsprozess [8].

Als Basistechnologie für die Extraktion von Eigenschaften wird der von der Uni Tübingen entwickelte SystemC Checker überarbeitet, der es erlaubt, temporallogische Formeln in PSL (Property Specification Language) gegen simulierte SystemC-Modelle zu prüfen [9]. Für die Eigenschaftsextraktion wurde der mögliche Einsatz des FZI SystemC-Parser untersucht, welcher ein SystemC-Modell in eine XML-Repräsentation umwandelt. Diese Darstellung wurde für eine Analyse des Systems verwendet, um Eigenschaften zu generieren.

Die Methode der Eigenschaftsextraktion wurde dahin gehend überarbeitet, dass nicht mehr ein SystemC-Modell als Grundlage verwendet wird, sondern Modelle der Unified Modeling Language (UML) bzw. Systems Modeling Language (SysML) [10]. Durch den Einsatz dieser Modelle werden die Eigenschaften aus einer hohen Abstraktionsebene extrahiert, die sich sehr nah an der Spezifikation befindet. Im Rahmen dieser Arbeiten fand eine Kooperation mit der Technischen Universität Ilmenau statt. Es besteht die Möglichkeit, die Eigenschaftsextraktion und die Methoden der Zeitverifikation in einem gemeinsamen Verifikationsframework zu verbinden. Diese Methode wird zurzeit an den beiden Universitäten untersucht. Es wurden weiterhin C-Software-Verifikationswerkzeuge untersucht, um einen Überblick über den Stand der Technik zu erstellen, aktuelle Werkzeuge zu vergleichen und die Grenzen der aktuellen Ansätze herauszuarbeiten. Die Arbeit

Eberhard Karls Universität Tübingen

Kont@kt:
Dr. habil. Thomas Kropf,
Dr. Jürgen Ruf,
Stefan Lämmermann
Universität Tübingen
Wilhelm-Schickard-Institut
für Informatik, Lehrstuhl Technische Informatik
Sand 13
72076 Tübingen
fon: 07071 29-75458
{kropf, ruf, laemmerm}@
informatik.uni-tuebingen.de

wurde im Rahmen einer laufenden Kooperation mit der Firma Bosch durchgeführt. Die Ergebnisse dieser Aktivität werden in die Entwicklung unserer eigenen Verifikationswerkzeuge zur hardwarenahen Softwareverifikation fließen.

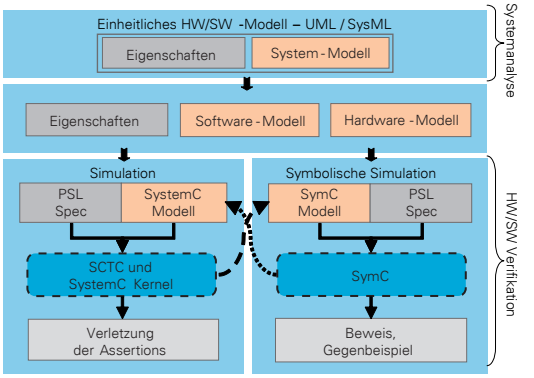

Abbildung 1.07: Verifikationsablauf und Werkzeuge für die Hardware/Software Co-Verifikation

Kompositionale Verifikation auf Systemebene

Der gegenwärtige Stand der Technik erlaubt die formale Verifikation von Blöcken, aber nicht von kompletten Systemen. Sind die Blöcke aber formal verifiziert, dann sind die zur Verifikation benutzten Eigenschaften auch eine abstrakte Repräsentation der Funktionalität der Blöcke. Beispielsweise kann in Eigenschaften auf Signalwerte zu mehreren Zeitpunkten referenziert werden und nicht nur, wie z.B. bei Zustandsdiagrammen, zum gegenwärtigen und nächsten Zeitpunkt. Die Abstraktheit der Blockeigenschaften kann daher ausgenutzt werden, um Blöcke durch abstrakte Modelle zu ersetzen, die aus den Blockeigenschaften generiert werden. Die aus den Eigenschaften erzeugten abstrakten Modelle werden „Cando-Objekte" genannt, weil sie jedes bis auf das durch die Eigenschaften explizit verbotene Verhalten nachbilden [11]. Für Cando-Objekte gelten eine Reihe von Kompositionalitäts-Eigenschaften [12]. Systemeigenschaften können daher nicht mehr nur auf dem ursprünglichen Modell, sondern auch auf einem Modell verifiziert werden, bei dem die Blöcke durch die Cando-Objekte ersetzt sind (siehe Abbildung 1.08).

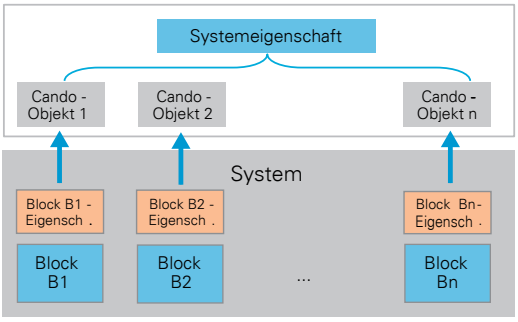

Abbildung 1.08: Kompositionale Verifikation mit Cando-Objekten

An der TU Darmstadt werden Verfahren zur effizienten Generierung von Cando-Objekten aus Blockeigenschaften entwickelt. Die Eigenschaften können z.B. in PSL spezifiziert werden. Cando-Objekte werden als VHDL-Beschreibungen erzeugt und können für die formale Verifikation, aber auch für die Simulation weiterbenutzt werden. Mit diesen Verfahren ist es an der TU Darmstadt zum ersten Mal gelungen, aus Eigenschaftssätzen signifikanter Komplexität (Bus-Protokolle wie z.B. AMBA-AHB oder PCI) ausführbare abstrakte Modelle zu generieren. Darüber hinaus können die Verfahren zur Lösung einer Reihe weiterer Probleme wie z.B. Konsistenz und Vollständigkeit von Eigenschaftssätzen benutzt werden.

Black-Box-Techniken bei der Eigenschaftsprüfung

Black-Box-Techniken werden eingesetzt, um Fehlererkennung und -lokalisierung schon in einem frühen Stadium des Entwurfs zu ermöglichen. Die Anwendung automatisierter, formaler Verifikationsmethoden auf SoC-Entwürfe und ihre Integration mit Black-Box-Techniken zieht eine wesentlich exaktere Fehlererkennung und Fehlerlokalisierung nach sich, als dies bei simulationsbasierten Ansätzen der Fall wäre (siehe Abbildung 1.09). In unserem Ansatz gehen wir von einer Spezifikation durch eine Menge temporaler Eigenschaften aus, die durch eine gegebene sequentielle Implementierung erfüllt werden sollen. Erfüllt die Implementierung die spezifizierten Eigenschaften nicht, dann soll der Fehler dem Designer durch die Berechnung „guter Gegenbeispiele" erklärt werden. Diese Gegenbeispiele stellen Abläufe des Systems dar, die zum einen den Fehlereffekt sichtbar machen, zum anderen für die praktische Anwendung so kurz wie möglich sein sollten und so wenige Systemkomponenten wie möglich benutzen sollten. Neben der optimierten Berechnung von Gegenbeispielen stellen wir Methoden zur automatischen Lokalisierung von Designfehlern zur Verfügung. Beide Ziele werden unter Anwendung so genannter Black-Box-Techniken erreicht.

Sowohl bei der Berechnung guter Gegenbeispiele als auch bei der Fehlerlokalisierung sind folgende grundlegenden Fragestellungen von Interesse. Auf Basis einer unvollständigen Schaltung, d.h. einer Schaltung, die sogenannte Black-Boxes enthält, ist einerseits die Frage zu beantworten, ob es möglich ist, die Black Boxes durch Implementierungen zu ersetzen, so dass die spezifizierte Eigenschaft erfüllt ist („Realisierbarkeit"). Andererseits wird die Frage gestellt, ob die Eigenschaft für jede mögliche Ersetzung der Black-Box erfüllt ist („Validität"). Hierzu wurde an der Uni Freiburg ein Modellprüfer entwickelt, mit dem es unter Einsatz von AIGs (And-Inverter-Graphen) als Mittel zur Repräsentation großer Zustandsräume gelingt, die angesprochenen Aufgaben zu lösen.

Zur Berechnung minimaler Gegenbeispiele, die auf möglichst wenigen Komponenten basieren, werden möglichst große Teile des Designs ausgeblendet, aus denen „Black-Boxes" gebildet werden. Wenn die folgende Eigenschaftsüberprüfung für das „Black-Box-Design" fehlschlägt, d.h. wenn wir für das Black-Box-Design die Nicht-Realisierbarkeit zeigen

Kont@kt:
Prof. Dr. Bernd Becker
Prof. Dr. Christoph Scholl
Universität Freiburg
Institut für Informatik
Georges-Koehler-Allee 51
79110 Freiburg
fon: 0761 203 {-8140, -8152},
{becker, scholl}@
informatik.uni-freiburg.de

TECHNISCHE UNIVERSITÄT DARMSTADT

Kont@kt:
Prof. Dr. Hans Eveking
Technische Universität Darmstadt,
Fachgebiet Rechnersysteme
Merckstr. 25
64283 Darmstadt
fon: 06151 162076,
eveking@rs.tu-darmstadt.de

können, wird dieses reduzierte Design zur Berechnung eines kompakten Gegenbeispiels verwendet. Dieses Gegenbeispiel ist dann in der Lage, den Fehlereffekt unabhängig von den ausgeblendeten Teilen zu erklären. Im Gegensatz dazu wird für die Fehlerlokalisierung nach kleinen Teilbereichen des Designs gesucht, so dass bei Ausblenden dieser Teile die Realisierbarkeit der gewünschten Systemeigenschaft gezeigt werden kann. Dies bedeutet, dass der vorliegende Fehler durch Abändern der ausgeblendeten Teile des Systems korrigierbar ist, so dass der mögliche Fehlerort eingegrenzt werden kann.

In Zusammenarbeit mit der Uni Darmstadt werden Black-Box-Techniken zu Grey-Box-Verifikationsansätzen verallgemeinert. Dabei wird das Innenleben der „Grey-Boxes“ durch einige wenige kritische Eigenschaften beschrieben. Die Anwendbarkeit der Konzepte auf analoge und Mixed-Signal-Schaltungen wird zusammen mit der Universität Frankfurt untersucht.

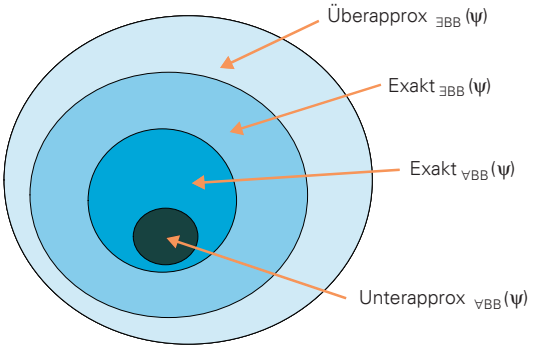

Abbildung 1.09: Darstellung von unter- und überapproximierten Zustandsmengen bei der Eigenschaftsüberprüfung für Black-Box-Designs

Kont@kt:
Prof. Dr. Wolfgang Kunz
TU Kaiserslautern,
AG Entwurf Informationstechnischer Systeme,
Postfach 3049
67653 Kaiserslautern
fon: 0631 205-3066
kunz@eit.uni-kl.de

Frontend-Modellgenerierung

Durch gezielte Maßnahmen im Frontend eines Property Checkers kann die Performanz bei der Lösung komplexer Verifikationsaufgaben drastisch gesteigert werden. Dies wird anhand pathologischer Fälle der formalen Blockverifikation demonstriert. Ein besonders interessantes Anschauungsbeispiel ergab sich durch die Möglichkeit, in Kooperation mit der Firma OneSpin Solutions GmbH an der formalen Verifikation des bei Infineon in der Entwicklung befindlichen Tricore 2 Prozessors mitzuwirken. Die TU Kaiserslautern beschäftigte sich dabei mit der MAC Unit, die zahlreiche DSP-Instruktionen mit Multiply-Accumulate-Operationen implementiert. Der formale Nachweis der Korrektheit des arithmetischen Ergebnisses auf voller Bitbreite war mit herkömmlichen Techniken bis jetzt nicht möglich. Durch eine gezielte Modellierung der Prozessorarithmetik im Frontend des Property Checkers und eine entsprechende Anpassung des Flows (Abbildung 1.10) konnten hier entscheidende Fortschritte erzielt werden. An der TU Kaiserslautern wurde das Modellierungskonzept des „arithmetic bit-level (ABL)“ und ein dazu passendes Normalisierungsverfahren erforscht [16]. Mit dessen Hilfe war es erstmals möglich die arithmetische Korrektheit aller MAC-Instruktionen des Tricore 2 auf voller Bitbreite nachzuweisen. Das internationale Interesse and dieser Lösung ist sehr groß, wie zahlreiche Anfragen und eine Vortragseinladung an die University of California at Berkeley belegen. Die Fortschritte bei der Arithmetikverifikation können auch in Hinblick auf die Eigenschaftsprüfung von Mixed-Signal-Schaltungen sehr nützlich sein. Dazu werden gerade in Kooperation mit der Uni Frankfurt Untersuchungen angestellt.

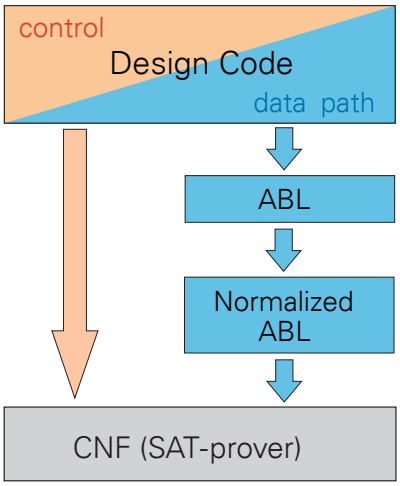

Abbildung 1.10: Verifikationsflow für Kontrolllogik und Arithmetik

Als zweites Thema werden Maßnahmen der Frontendmodellierung im Zusammenhang mit der Verifikation sequenzieller Systeme untersucht. Es konnte gezeigt werden, dass durch eine bestimmte Zustandskodierung die Generierung von Invarianten bei der temporalen Induktion effektiv unterstützt werden kann. Darüber hinaus wurde ein Verfahren zur Verifikation von Hardwareprotokollimplementierungen entwickelt, das auf einer geeigneten Dekomposition des Zustandsraumes beruht. Erste Untersuchungen an industriellen Entwürfen (Infineon) belegen, dass damit der manuelle Verifikationsaufwand drastisch reduziert werden kann [17]. Auch die Frontend-Maßnahmen zur Verifikation sequenzieller Systeme sind in Fachkreisen auf großes Interesse gestoßen, wie ein eingeladener Vortrag zu diesem Thema auf der ASICON-05 belegt. Die an der TU Kaiserslautern entwickelten Methoden werden z.Zt. mit dem an der TU Darmstadt entwickelten Ansatz zur kompositionalen Verifikation kombiniert. Erste erfolgversprechende Ergebnisse liegen bereits vor. Ein besonderes Potential der erzielten Forschungsergebnisse in Hinblick auf zukünftige Arbeiten liegt insbesondere im Bereich der Verifikation hardwarenaher Software. Untersuchungen dazu werden bereits gemeinsam mit der Uni Tübingen durchgeführt.

Mixed-Signal Model-Checking

Ein weiteres Ziel des Projekts ist die Verifikation von gemischt analogen und digitalen Schaltungen (siehe Abbildung 1.11). Eine Herausforderung besteht immer noch in der Modellierung von integrierten Schaltungs-

klassen, die einen unterschiedlichen Werte- und Zeitcharakter besitzen. Es existieren bereits Modellierungsmöglichkeiten (Timed Automata, Hybride Automaten etc.), die den kontinuierlichen Werte- und Zeitcharakter analoger Schaltungen wiedergeben können, jedoch bisher nicht in der Lage sind, vollständige Eigenschaftsprüfungen durchzuführen. Zudem unterliegen diese Modelle zumeist Restriktionen, welche die zu behandelnden Schaltungsklassen stark einschränken. Zwingend notwendig ist hier die Einführung von Zeitbedingungen, die die temporalen Abläufe der Schaltungen wiedergeben und gleichzeitig die Signalabhängigkeiten innerhalb beider Schaltungsklassen berücksichtigen.

Ein neuer Ansatz besteht darin, das analoge Teilsystem so zu diskretisieren, dass eine Zusammenführung mit dem Modell des digitalen Teilsystems möglich ist. Ein besonderes Merkmal des Modells ist, dass die Zeit in beiden Teilsystemen unterschiedlich zu behandeln ist und modelliert werden muss. Ausgehend von einem Algebro-Differentialgleichungssystem wird der kontinuierliche Zustandsraum der analogen Teilschaltung aufbaut und in geeigneter Weise diskretisiert. Hierdurch entsteht ein zeitbehaftetes Transitionssystem mit vielen unterschiedlichen diskreten Zeiten. Dieses wird zur symbolischen Weiterverarbeitung in ein Multi Terminal Binary Decision Diagram (MTBDD) überführt. Parallel werden MTBDDs erzeugt, welche die digitalen Schaltungszustände und deren Transitionen wiedergeben. Die so erzeugten MTBDDs werden anschließend gemeinsam mit einer zu verifizierenden Spezifikation einem Model-Checking Algorithmus zugeführt. Die verwendetet Spezifikationssprache ist CTL-AT, mit der es möglich ist, neben dem Standard CTL-Sprachumfang Zeitintervalle und analoge Zustandsgebiete anzugeben.

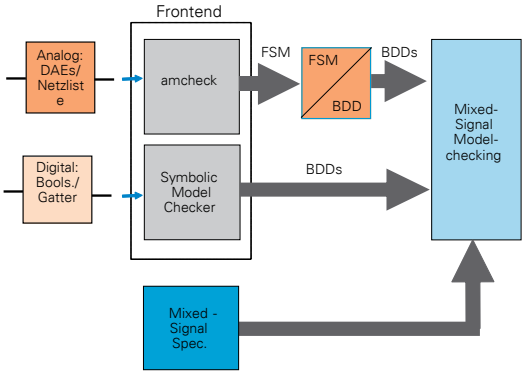

Abbildung 1.11: Mixed-Signal Model-Checking

Resümee

Das Clusterforschungsprojekt hat sich zum Ziel gesetzt, Verifikationslücken zu schließen, um somit dem langfristigen Ziel eines geschlossenen Verifikationsflows von der Systemebene bis zur Transistorebene näher zu kommen. Dieser Bericht gibt dabei einen Überblick über die schon nach zwei Jahren Projektlaufzeit erreichten Ergebnisse. Die umfangreiche Literatur, die während der Projektlaufzeit veröffentlicht wurde, gibt einen detaillierten Einblick in die Forschungsergebnisse. Nähere

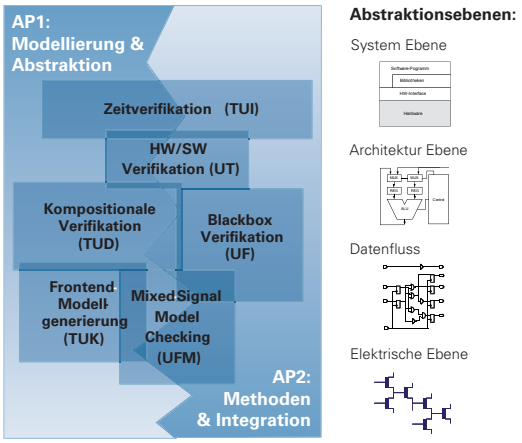

Abbildung 1.12: Themenschwerpunkte und Arbeitspakete im FEST-Projekt

Informationen können auch über die angegebenen Kontaktadressen in Erfahrung gebracht werden.

In den sechs Themenschwerpunkten des Projekts wurden große Fortschritte erzielt und neue Ansätze zur Verifikation erforscht und Methoden ausgetauscht. Die Modellierungsverfahren und Methoden können bestehende Verfahren verbessern oder ergänzen, so dass Verifikationslücken geschlossen werden können. Abbildung 1.12 stellt die Themenschwerpunke des Projekts dar, die in zwei Arbeitspaketen bearbeitet werden.

Schon nach zwei Jahren Projektlaufzeit wurden wichtige Ergebnisse zur Erprobung der neuen Verfahren erzielt. Eine Überführung der erforschten Methoden in die industrielle Praxis ist der nächste Schritt, um die Modelle und Methoden weiter für eine industrielle Anwendung zu verbessern. Erste Ansätze zum Einsatz im industriellen Umfeld sind erfolgt. Im letzten Jahr stehen der intensive Austausch mit den Industriepartnern, sowie die Verfeinerung der Demonstratoren, die im Rahmen der Forschungsaktivitäten erstellt wurden, im Vordergrund der Projektarbeit. Um die Ergebnisse für die Industrie in EDA-Werkzeugen und Flows nutzbar zu machen, sind weitere Forschungs- und Entwicklungsanstrengungen zusammen mit der Anwender- und EDA-Industrie notwendig.

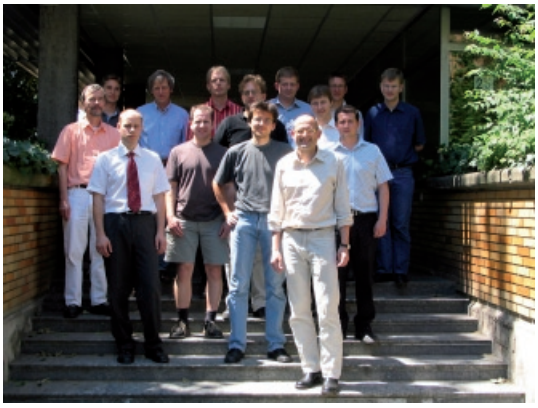

Abbildung 1.13: Das Forscherteam bei einem Projekttreffen in Darmstadt

JOHANN WOLFGANG GOETHE
UNIVERSITÄT
FRANKFURT AM MAIN

Kont@kt:
Prof. Dr.-Ing. Lars Hedrich
Johann Wolfgang Goethe-Universität Frankfurt am Main,
Institut für Informatik,
Professur für Entwurfsmethodik
Postfach 11 19 32
60054 Frankfurt
fon: 069 798-22297
hedrich@em.informatik.uni-frankfurt.de

Kont@kt (FEST):
Dr.-Ing. Volker Schöber,
Projektkoordination edacentrum,
Schneiderberg 32
30167 Hannover
fon: 0511 762-19688
fax: 0511 762-19695
schoeber@edacentrum.de

Weitere Informationen im Internet: http://www.edacentrum.de/projekte/edaclusterforschung/fest.html

Schlüsselwörter:
Mission Level Design «
ESL-Simulation «
ESL-Integration «
Validierung «
Zeitanalyse «
Systemverifikation «
Systemverifikation «
Hardwarenahe
SW-Verifikation «
Semiformale Verifikation «
Eigenschaftsextraktion «
HW-SW-Co-Verifikation «
Kompositionale Verifikation «
Synthese aus Eigenschaften «
Cando-Objekt «
Frontend-Modellgenerierung «
Arithmetic Bit-Level «
Tricore «
And-Inverter-Graphen «
Black-Boxes «
Grey-Boxes «
Protokollverifikation «
Mixed-Signal Modellierung «
Mixed-Signal Verifikation «
Modelchecking «
Formale Verifikation «
Multi Terminal Binary
Decision Diagram «
MTBDD «
CTL-AT «

Wichtige Veröffentlichungen des Projekts

[1] H. Rath, G. Schorcht, H. Salzwedel: Simulationsumgebung für Bordnetze – Bordnetz-Spezifikationen modellieren und validieren, Hanser automotive, S. 34–38, Carl Hanser Verlag, 5–6 2006.

[2] H. Salzwedel: Mission Level Design of Avionics, AIAA-IEEE DASC – The 23rd Digital Avionics Systems Conference 2004, Salt Lake City, Utah, USA, 24.–28.10.2004.

[3] T. Hummel, W. Fengler: Design of Embedded Control Systems Using Hybrid Petri Nets and Time Interval Petri Nets, in: M. A. Adamski, A. Karatkevich, M. Wegrzyn (Eds.): Design of Embedded Control Systems, ISBN 0-387-23630-9, Springer, S. 141–154, 2005.

[4] V. Duridanova, T. Hummel, O. Fengler, W. Fengler: Verifikation von Spezifikationsmodellen mit Intervall-Petri-Netzen, in: D. Stoffel, W. Kunz (Eds.): Methoden und Beschreibungssprachen zur Modellierung und Verifikation von Schaltungen und Systemen, 7. GI/ITG/GMM-Workshop zu Modellierung und Verifikation, ISBN 3-8322-2486-6, Shaker Verlag, S. 184–193, 2004.

[5] A. Pacholik, W. Fengler, H. Salzwedel, O. Vinogradov: Real Time Contraints in System Level Specifications Improving the Verification Flow of Complex Systems, Net.ObjectDays 2005 – OORE, 05, Erfurt, Proceedings S. 283–294, ISBN 3-9808628-4-4t, 19.–22.9.2005.

[6] Pradeep K. Nalla, Roland J. Weiss, Prakash M. Peranandam, Jürgen Ruf, Thomas Kropf, Wolfgang Rosenstiel: Distributed Symbolic Bounded Property Checking, 4th International Workshop on Parallel and Distributed Methods in Verification (PDMC), Lisboa, Portugal, 10.7.2005.

[7] Prakash M. Peranandam, Pradeep K. Nalla, Roland J. Weiss, Jürgen Ruf, Thomas Kropf, Wolfgang Rosenstiel: Overlap Reduction in Symbolic System Traversal, IEEE International High Level Design Validation and Test Workshop 2005 (HLDVT), Napa Valley, California, USA, 30.11.–2.12.2005.

[8] Prakash M. Peranandam, Pradeep K. Nalla, Roland J. Weiss, Jürgen Ruf, Thomas Kropf, Wolfgang Rosenstiel: Fast Falsification Based on Symbolic Bounded Property Checking, 43th Design Automation Conference (DAC), San Francisco California, USA, 24–28.7.2006.

[9] Roland J. Weiss, Jürgen Ruf, Thomas Kropf, Wolfgang Rosenstiel: Efficient and Customizable Integration of Temporal Properties into SystemC, Forum on specification and Design Languages (FDL), Lausanne, Switzerland, 27–30.9.2005.

[10] Stefan Lämmermann, Roland J. Weiss, Jürgen Ruf, Thomas Kropf, Wolfgang Rosenstiel: Automatic Generation of Verification Properties for SoC Design from SysML-Diagrams, 3rd International UML for SoC Design Workshop at DAC'06, San Francisco California, USA, 23.7.2006.

[11] M.Schickel, V. Nimbler, M. Braun, H. Eveking: On Consistency and Completeness of Property-Sets: Exploiting the Property-Based Design-Process, Proc. FDL, 2006

[12] M.Schickel, V. Nimbler, M. Braun, H. Eveking: Property-Based Model Generation for Compositional Verification, to be published.

[13] C. Scholl and B. Becker: Checking equivalence for partial implementations, Design Automation Conference, pages 238–243, 2001.

[14] T. Nopper and C. Scholl: Approximate symbolic model checking for incomplete design, Formal Methods in Computer-Aided Design, pages 290–305, November 2004.

[15] F. Pigorsch, C. Scholl, and S. Disch: Advanced unbounded model checking based on AIGs, BDD sweeping, and quantier scheduling, Formal Methods in Computer-Aided Design, November 2006.

[16] M. Wedler, D. Stoffel, W. Kunz: Normalization on the Arithmetic Bit Level, ACM/IEEE Design Automation Conference (DAC), Anaheim, Juni 2005.

[17] M. Nguyen, D. Stoffel, M. Wedler, W. Kunz: Transition-by-Transition FSM Traversal for Reachability Analysis in Bounded Model Checking, Proc. IEEE/ACM Int. Conf. on Computer-Aided Design (ICCAD), San Jose, CA, November 2005.

[18] W. Hartong, L. Hedrich, E. Barke: Model Checking Algorithms for Analog Verification, DAC, Juli 2002.

[19] D. Grabowski, D. Platte, L. Hedrich, E. Barke: Time Constrained Verification of Analog Circuits using Model-Checking Algorithms, ENTCS, Januar 2005.