



# Indikatorenbasierte Bewertung von Entwurfsentscheidungen auf Systemebene

M. Freier<sup>1</sup>, A. Wenzler<sup>1</sup>, T. Mayer<sup>1</sup>, J. Gerlach<sup>2</sup>, and W. Rosenstiel<sup>3</sup>

<sup>1</sup>Robert Bosch GmbH, Germany

<sup>2</sup>Hochschule Albstadt-Sigmaringen, Germany

<sup>3</sup>Eberhard Karls Universität Tübingen, Germany

Correspondence to: M. Freier (marvin.freier@de.bosch.com)

**Zusammenfassung.** Bei dem Entwurf von anwendungsspezifischen integrierten Schaltungen muss ein Entwickler zusätzliche Funktionen integrieren und die zunehmende Komplexität bewältigen. Für die Reduzierung der Kosten bleibt weiterhin die Verkürzung der benötigten Entwicklungszeit ein Ziel. Ein entscheidender Faktor dabei ist die zuverlässige Absicherung von Entwurfsentscheidungen in einer frühen Entwurfsphase. In der vorliegenden Arbeit wird eine Methode vorgestellt, die eine Bewertung von Entwurfsentscheidungen bei Systemmodellen mittels automatisch ermittelter Indikatoren ermöglichen soll. Die Systemmodelle können mit der Entwurfsumgebung MATLAB/Simulink oder in SystemC beschrieben sein.

## 1 Einführung

Eine bewährte Herangehensweise bei der ASIC-Entwicklung ist der Top-Down-Entwurf. Dabei wird zu Beginn des Entwurfs ein abstraktes Modell auf Systemebene entworfen. Mit Hilfe von einem Systemmodell kann ein Entwickler das grundsätzliche Gesamtverhalten des Systems betrachten und die Auswirkung der Integration zusätzlicher Funktionen untersuchen. Mit einem Top-Down-Ansatz lässt sich die steigende Komplexität eines Systems bewältigen und der Entwickler wird bei der Verkürzung der Entwicklungszeit unterstützt.

Ein Systemmodell wird auf Basis der Spezifikation des ASICs modelliert. In dieser frühen Entwurfsphase sind einige Entwurfsentscheidungen, wie die HW/SW-Partitionierung oder das Kommunikationskonzept, noch nicht getroffen. Grundlagen für die Beantwortung von Entwurfsfragen sind neben der Spezifikation, Modelleigenschaften und die Erfahrung eines Entwicklers. Im weiteren Entwurfsablauf wird unter Einhaltung der Spezifikation

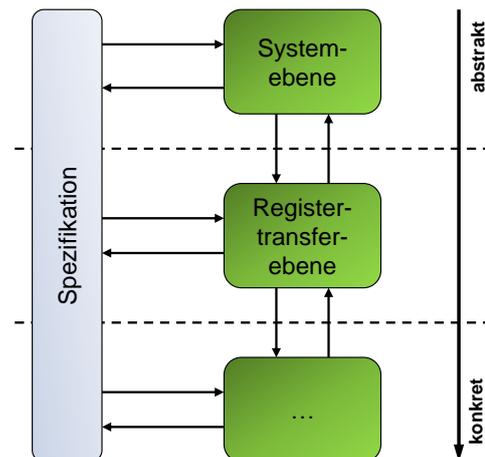


Abb. 1. Exemplarischer Entwurfsablauf

das Modell schrittweise bis zur Implementierung verfeinert. Die endgültige Implementierung wird in einer Hardwarebeschreibungssprache, wie VHDL oder Verilog, beschrieben (Mauderer et al., 2011). In Abbildung 1 wird dieses Vorgehen exemplarisch dargestellt.

Ein entscheidender Punkt bei der ASIC-Entwicklung ist die zuverlässige Absicherung von Entwurfsentscheidungen in einer frühen Entwurfsphase. Mit Hilfe einer zuverlässigen Absicherung können zusätzliche Kosten durch mögliche Fehlentscheidungen vermieden werden. Die potentiellen Kosten und die benötigte Entwicklungszeit steigen mit dem fortschreitenden Entwurf. Eine Herausforderung bei der Absicherung von Entwurfsentscheidungen auf Systemebene ist das Fehlen von relevanten Informationen. Einerseits sind durch die abstrakte Beschreibung nicht alle Informationen verfügbar, andererseits werden Details zur Beschleunigung

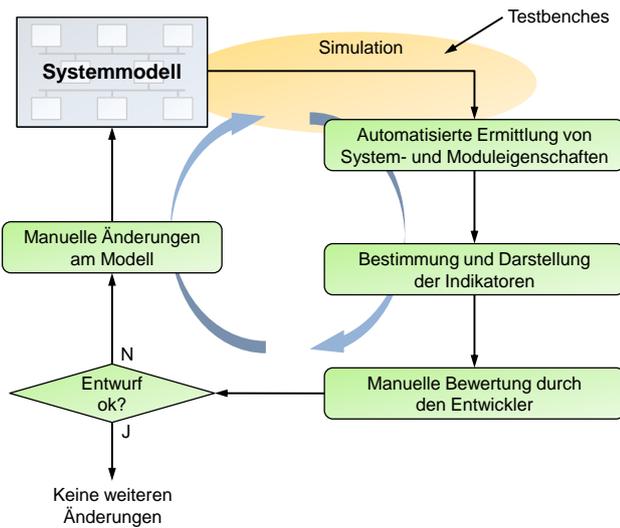


Abb. 2. Methode für die Architekturexploration

der Simulation gezielt ausgeblendet. In der vorliegenden Arbeit wird eine Methode vorgestellt, die eine zuverlässige Absicherung von Entwurfsentscheidungen bei abstrakten Systemmodellen durch automatisiert ermittelte Indikatoren ermöglichen soll.

Im nachfolgenden Abschnitt wird die Methode und die einzelnen Schritte vorgestellt. Der Abschnitt drei beschreibt die Anforderungen an die Entwurfsfragen und beinhaltet Beispiele. Die Systemmodelle werden in Abschnitt vier behandelt. Außerdem wird SystemC für die Beschreibung von Systemmodellen kurz vorgestellt. Im fünften Abschnitt wird die Beispielschaltung mit den enthaltenen Komponenten beschrieben. Der abschließende Abschnitt sechs fasst die Arbeit zusammen.

## 2 Methode

Die Methode soll in einer frühen Entwurfsphase eingesetzt werden und den Entwickler bei der Absicherung von Entwurfsentscheidungen unterstützen. Im Fokus der Methode steht ein abstraktes Systemmodell eines ASICs. Dieses Modell bildet entsprechend der Spezifikation die grundsätzliche Funktionalität der zu entwerfenden integrierten Schaltung ab. Abbildung 2 veranschaulicht die einzelnen Schritte der Methode.

In einem ersten Schritt sollen System- und Moduleigenschaften ermittelt werden. Diese Eigenschaften bilden den Ausgangspunkt für die weiteren Betrachtungen. Die Ermittlung der Eigenschaften soll während einer Simulation erfolgen. Hierfür sollen in der Simulation unterschiedliche Sichtweisen abgedeckt werden. Zu den Sichtweisen zählt die Sicht des Kunden, der ASIC-Entwickler, der Modulentwickler und der Softwareentwickler. Da die Testumgebung und die Test-szenarien eine deutliche Auswirkung auf die Systemeigen-

schaften besitzen, müssen die zu untersuchende Testszenarien sorgfältig ausgewählt werden. Beispiele für System- und Moduleigenschaften sind die Taktrate, die Latenz oder die Übertragungsraten.

Ausgehend von den System- und Moduleigenschaften sollen im nachfolgenden Schritt Indikatoren automatisiert abgeleitet werden. Die Indikatoren bilden die Grundlage für die Absicherung von Entwurfsentscheidungen. Wie in Formel 1 beschrieben, lässt sich ein Indikator  $Ind_k$  aus der Summe der Eigenschaften  $x$  und der dazugehörigen Gewichtungen  $g$  bilden. Die Gewichtung einer Eigenschaft soll mit einer Gewichtungsfunktion berechnet werden. Ein weiterer Aspekt bei der Bestimmung der Indikatoren ist die Reduzierung der benötigten Eigenschaften. Dadurch können die Testszenarien angepasst und die Simulationszeit, die zur Ermittlung der System- und Moduleigenschaften benötigt wird, reduziert werden. Jedoch muss dabei beachtet werden, dass die Aussagekraft eines Indikators nicht negativ beeinflusst wird.

$$f(Ind_k) = \sum_{i=0}^n g_i \cdot x_i \quad (1)$$

Sobald die Indikatoren berechnet wurden, sollen diese dem Entwickler für die Überprüfung der Entwurfsentscheidung ausgegeben werden.

Auf Basis der Indikatoren soll ein Entwickler eine realisierte Komponente hinsichtlich einer konkreten Entwurfsfrage bewerten. Dieser Schritt erfolgt bewusst nicht automatisiert und die ermittelten Indikatoren sollen für den Entwickler nachvollziehbar sein. Andererseits soll ein Entwickler an dieser Stelle die Möglichkeit besitzen, sein Expertenwissen einfließen zu lassen und sich gegebenenfalls auch entgegen der Bewertung zu entscheiden. Gründe könnten beispielsweise nichtfunktionale Anforderungen wie Erhöhung der Flexibilität trotz eines geringeren Flächenverbrauchs sein.

Für den Fall, dass ein Entwickler mit den getroffenen Entwurfsentscheidungen unzufrieden ist, kann der Entwickler eigenhändig Änderungen an der Entwurfsvariante vornehmen und die Methode nochmals durchlaufen. Durch die wiederholte Anwendung kann die Auswirkung der Änderung auf das Gesamtsystem untersucht werden. Außerdem stehen dem Entwickler zusätzlich zu der aktuellen Bewertung die bereits betrachteten Entwurfsvarianten mit den entsprechenden Indikatoren zur Verfügung. Der erstrebte Gewinn liegt darin, dass die Explorationsschleife dadurch verkleinert wird. Dieses Vorgehen kann beliebig oft wiederholt werden. Die Methode endet sobald eine Entwurfsvariante den Anforderungen eines Entwicklers entspricht.

## 3 Entwurfsfragen

In einer frühen Entwurfsphase werden die Anforderungen an die zu entwickelnde integrierte Schaltung weitestgehend

vom Kunden definiert. Beispiele für solche Anforderungen sind unter anderem der maximal erlaubte Strom- und Flächenverbrauch oder die zu realisierenden Schnittstellen. Ausgehend von solchen Anforderungen werden konkrete Entwurfsfragen für die Entwicklung eines ASICs abgeleitet. Werden die Entwurfsfragen fehlerbehaftet beantwortet, können abhängig von der Entwurfsphase erhebliche Kosten entstehen.

Die im vorherigen Kapitel vorgestellte Methode soll bei Entwurfsfragen eingesetzt werden, die typischerweise in einer frühen Entwurfsphase auftreten und den Entwurf wesentlich beeinflussen. Für die Betrachtung der Entwurfsfragen müssen die nachfolgenden Anforderungen erfüllt werden:

- *Simulation.* Die System- und Moduleigenschaften, von denen in einem nachfolgenden Schritt der Methode die Indikatoren abgeleitet werden, können während einer Simulation ermittelt werden. Bei der Beschreibung der Testszenarien muss darauf geachtet werden, dass die Testfälle für die Ermittlung der benötigten Eigenschaften anwendbar sind und die relevanten Fälle abdecken.
- *Verfügbarkeit.* Die benötigten Module zur Betrachtung der Entwurfsfrage können auf Systemebene beschrieben und in das Systemmodell integriert werden. Trotz der abstrakten Beschreibung eines Moduls müssen die Eigenschaften für eine zuverlässige Absicherung geeignet sein.
- *Zuverlässigkeit.* Der Entwickler bewertet auf Basis automatisch ermittelter Indikatoren eine Entwurfsfrage. Aus diesem Grund müssen die Simulationsergebnisse und die festgestellten System- und Moduleigenschaften für den Entwickler nachvollziehbar sein. Ein weiterer wichtiger Aspekt ist die Reproduzierbarkeit der erzielten Ergebnisse.
- *Aufwand.* Bei der Betrachtung mancher Entwurfsfragen muss das Systemmodell verfeinert werden. Jedoch soll das Hinzufügen zusätzlicher Information überschaubar bleiben und nicht zu viel Entwicklungszeit in Anspruch nehmen. Außerdem sollen die Simulationsergebnisse in einer relativ kurzen Zeit ausgewertet werden können.
- *Nutzen.* Der Einsatz der vorgestellten Methode zur Untersuchung einer Entwurfsfrage muss gegenüber dem bisherigen Vorgehen überprüft werden. Bei dem bisherigen Vorgehen werden die Entwurfsentscheidungen größtenteils auf Basis von Tabellenkalkulationen getroffen. Jedoch können diese Kalkulationen das dynamische Verhalten des Gesamtsystems nicht oder bestenfalls ansatzweise abbilden.

Bei der Verletzung einer dieser Anforderungen kann eine zuverlässige oder ressourcenschonende Absicherung einer Entwurfsentscheidung nicht garantiert werden.

Die Auslegung des Frontends, die Parametrisierung einer Schnittstelle oder die Hardware/Software-Partitionierung (im Folgenden: HW/SW-Partitionierung) sind Themen, die im Idealfall in einer frühen Entwurfsphase entschieden werden. Durch die fortschreitende Digitalisierung eines ASICs nimmt die Bedeutung einer geeigneten HW/SW-Partitionierung zunehmend zu. Bei der Digitalisierung werden bisher analoge Komponenten als digitale Elemente umgesetzt. Der Grund hierfür liegt hauptsächlich im besseren Shrink-Faktor digitaler Elemente bei neueren Technologien. Der Shrink-Faktor steht bei einem Technologiewechsel für die prozentuale Verringerung der benötigten Fläche einer Komponente. Ein weiterer Aspekt bei der HW/SW-Partitionierung ist die Realisierung einer Komponente in Software. Eine softwaretechnische Implementierung bietet gegenüber einer in Silizium gegossenen Lösung eine verbesserte Flexibilität bei einem höheren Flächenverbrauch. Solch ein Tradeoff muss ein Entwickler bei der Absicherung einer Entwurfsentscheidung betrachten.

#### 4 Systemmodelle

Für die Beschreibung eines Systems auf Systemebene können verschiedene Beschreibungssprachen eingesetzt werden (Mauderer et al., 2011). Dabei bilden Systemmodelle die grundsätzliche Funktionalität auf einer abstrakten Ebene ab und können sich neben der Beschreibungssprache in der Abstrahierung, der Sichtweise und dem Umfang voneinander unterscheiden. Innerhalb dieser Forschungsarbeit werden Systemmodelle betrachtet, die in SystemC (Accellera Systems Initiative, 2012) oder mittels MATLAB/Simulink (MathWorks, Inc., 2012) beschrieben sind.

Die Systembeschreibungssprache SystemC ist eine C++-Klassenbibliothek und erweitert die objektorientierte Programmiersprache für die Beschreibung typischer Hardwareeigenschaften wie Synchronisation, Parallelität oder Interprozesskommunikation. SystemC wird für die Beschreibung komplexer elektronischer Systeme auf Systemebene oder auf der Registertransferebene verwendet. Im Gegensatz zu anderen Beschreibungssprachen wird SystemC überwiegend für die Modellierung und Simulation eines Gesamtsystems eingesetzt (Maillet-Contoz, 2010). Dieses Gesamtsystem kann neben Hardwarekomponenten auch in Software realisierte Komponenten enthalten. Die Software kann über Host-Code Execution oder einem Instruction Set Simulator eingebunden werden (Cockx, 2000). Bei der Verwendung des TLM-Ansatzes (Transaction Level Modeling) liegt der Fokus des Datentransfers auf die Funktionalität und nicht auf der tatsächlichen Realisierung. Durch die Kapselung und der abstrakteren Beschreibung der Kommunikation können Module ohne aufwändige Änderungen am Modell ausgetauscht werden (Ghenassia, 2005).

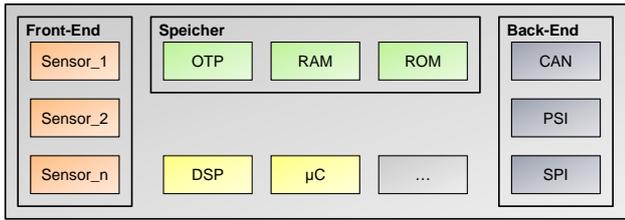


Abb. 3. Komponenten der Beispielschaltung

## 5 Beispielschaltung

Eine erste Anwendung der Methode findet bei dem Entwurf einer Sensorauswerteschaltung statt. Dieser ASIC soll einen Drehratensensor und mehrere Beschleunigungssensoren auswerten und die aufgearbeitete Sensordaten bereitstellen können. Der ASIC soll in automotiven Steuergeräten (engl. Electronic Control Units, ECUs), wie beispielsweise dem Elektronischen Stabilitätsprogramm (ESP<sup>®</sup>) oder einem Airbag-Steuergerät, eingesetzt werden. Als Beispiel wird der Vorgänger des zu entwerfenden ASICs verwendet. Von diesem ASIC liegt die Spezifikation inklusive den Anforderungen, den Blockschaltbildern und der Testumgebung mit Testfällen vor. In Abbildung 3 werden einige charakterisierende Komponenten der Beispielschaltung dargestellt.

Im Frontend des ASICs werden die Signalläufe der bereits angesprochenen Drehraten- und Beschleunigungssensoren mittels analoger Komponenten, wie zum Beispiel Filterstrukturen oder Analog-Digital-Wandler, für die weitere Signalverarbeitung im ASIC aufbereitet. Die Signalverarbeitung findet in einem Digitalen Signalprozessor (engl. Digital Signal Processor, DSP) und teilweise in Hardware statt. Der ASIC besitzt als Speicherelemente einen nicht wiederbeschreibbaren ROM (Read-Only Memory), einen einmalig überschreibbaren OTP (One-Time Programmable) und einen mehrfach beschreibbaren RAM (Random-Access Memory). Das ROM-Modul enthält die Firmware, die im Einsatz der Schaltung auf dem Mikrocontroller ausgeführt wird. Mit Hilfe des applikationsspezifischen Codes aus dem OTP werden die Sensoren an spezielle Anwendungen angepasst. Neben weiteren Modulkonfigurationen bietet der OTP eine Möglichkeit für die Bereitstellung von potentiellen Software-Patches. Während der Laufzeit werden Daten von den einzelnen Modulen im RAM gespeichert. Diese Daten können über verschiedene Schnittstellen ausgelesen werden. Zu den Schnittstellen gehören unter anderem CAN (Controller Area Network), SPI (Serial Peripheral Interface) und PSI (Peripheral Sensor Interface). Die Schnittstellen CAN und PSI sind im integrierten Mikrocontroller implementiert. Außerdem stellt der Mikrocontroller noch Safety-Funktionen zur Verfügung. Das System besitzt von den Schnittstellen PSI und SPI verschiedene Ausprägungen. Bei der SPI unterscheiden sich diese in der Größe der Frames und dem Versenden der Antwort auf eine Anfrage. Eine Antwort kann da-

bei innerhalb des gleichen Frames oder im anschließenden Frame erfolgen. In Abbildung 3 werden beispielsweise die eingesetzten Busse für die interne Kommunikation oder der Safety-Controller nicht dargestellt.

Die Methode soll unter anderem bei der Betrachtung einer Entwurfsfrage aus dem Bereich der HW/SW-Partitionierung eingesetzt werden. Durch eine zuverlässige Absicherung der HW/SW-Partitionierung auf Systemebene kann eine erste Abschätzung des Flächenverbrauchs und des Speicherbedarfs vorgenommen werden. Diese Abschätzungen können bei der Festlegung der Technologie als zusätzlicher Faktor eingesetzt werden. Andererseits kann zu einem frühen Zeitpunkt mit der Softwareentwicklung begonnen werden. Für die Absicherung der Entwurfsfrage soll neben der Samplerate auch ein Indikator ermittelt werden, der den Flächenverbrauch einer Komponente bei einer Realisierung in Hardware darstellt. Entsprechend der Formel 1 lässt sich der Indikator durch die Summe gewichteter Eigenschaften beschreiben. Zu den Eigenschaften zählen etwa die Wortbreiten der Eingangs- und Ausgangssignale, die Anzahl der Stufen eines Filters und die Anzahl elementarer Elemente, wie Schieberegister, Addierer oder Multiplizierer. Dabei sollen die initialen Gewichtungsfaktoren durch den Entwickler konfigurierbar sein. Gegenüber dem herkömmlichen Ansatz findet die Betrachtung durch den Entwickler direkt in der Entwicklungsumgebung am Modell statt und bezieht das dynamische Verhalten ein. Die Anwendung der Methode ist Bestandteil momentaner Arbeiten.

## 6 Zusammenfassung

In der vorliegenden Arbeit wurde eine Methode für die Entwicklung von ASICs vorgestellt um eine zuverlässige Absicherung von Entwurfsentscheidungen in einer frühen Entwurfsphase zu ermöglichen. Die Absicherung erfolgt auf Basis automatisch ermittelter Indikatoren. Bei der Methode werden während einer Simulation definierte System- und Moduleigenschaften eines Systemmodells, welches mittels MATLAB/Simulink oder in SystemC beschrieben ist, festgestellt. Anhand dieser Eigenschaften werden Indikatoren bestimmt und dem Entwickler grafisch ausgegeben. Auf Basis der Indikatoren bewertet ein Entwickler die getroffenen Entwurfsentscheidungen und nimmt gegebenenfalls Änderungen am Modell vor. Bei Änderungen am Systemmodell kann das methodische Vorgehen wiederholt werden.

Für die Untersuchung der Entwurfsfragen müssen diese bestimmten Anforderungen entsprechen. Beispielsweise müssen die für die Indikatoren benötigten System- und Moduleigenschaften während einer Simulation ermittelt werden können.

Die Methode soll bei dem Entwurf einer Sensorauswerteschaltung eingesetzt werden. Hierbei sollen mehrere Entwurfsfragen untersucht und zuverlässig abgesichert werden. Als Beispielschaltung wird der Vorgänger des zu entwickelnden Systems verwendet.

*Danksagung.* Die Autoren danken dem Bundesministerium für Bildung, Wissenschaft, Forschung und Technologie (BMBF) für die finanzielle Unterstützung im Rahmen des Projektes SANITAS (Förderkennzeichen 01M3088E).

## Literatur

- Accellera Systems Initiative: Homepage Accellera Systems Initiative, <http://www.accellera.org/home/>, 2012.
- Cockx, J.: Efficient Modeling of Preemption in a Virtual Prototype, in: Rapid System Prototyping, 2000, RSP 2000, Proceedings, 11th International Workshop on, 14–19, IEEE, 2000.
- Ghenassia, F.: Transaction Level Modeling with SystemC, Springer Verlag, 2005.
- Maillet-Contoz, L.: Standards for System Level Design, in: Computer-Aided Design (ICCAD), 2012 IEEE/ACM International Conference on, 332–335, Syst. Platforms Group, STMicroelectronics, Grenoble, France, 2010.
- MathWorks, Inc.: Homepage MathWorks - MATLAB and Simulink for Technical Computing, <http://www.mathworks.com>, 2012.
- Mauderer, A., Oetjens, J., and Rosenstiel, W.: System-Level Design for Automotive Mixed-Signal-ASICs: An Industrial Point of View, in: Proceedings of MBMV, Vol. 2011, 2011.