

Elektronische Systeme werden in Zukunft autonom auf Störungen und Veränderungen der Umwelt reagieren müssen und dafür den inneren Zustand flexibel anpassen können. Dies erfordert ein neues Denken im Entwurfsprozess von elektronischen Systemen. Daher haben das Bundesministerium für Bildung und Forschung, das Steuerungsgremium des edacentrum und das Leitungsgremium des RSS gemeinsam einen Aufruf für Forschungseinrichtungen gestartet, der zum dritten Clusterforschungsprojekt „Autonome integrierte Systeme“ unter dem Förderkennzeichen 01M3083 führte. Das Projekt ist im Dezember 2006 gestartet und stellt in diesem Kurzbericht die Ziele und Aufgaben der Partner und erste Ergebnisse im Newsletter des edacentrum vor.

Zusammensetzung des Projektkonsortiums:

TU Braunschweig
TU Kaiserslautern
TU München
Uni Tübingen
Uni Paderborn
Uni Erlangen-Nürnberg

Förderkennzeichen

01 M 3083

Laufzeit des Vorhabens:

01.12.2006–30.11.2009

Homepage:

<http://www.edacentrum.de/ais/>

Motivation und Ziel des Projekts

Um einen zuverlässigen Betrieb über die Störung hinaus in elektronischen Systemen zu gewährleisten, ist das Erfassen und Reagieren der Systeme unter gestörten Betriebsbedingungen notwendig. Mit Hilfe von Sensoren, Evaluatoren und Aktoren in MPSoCs können sporadisch auftretende Störungen erfasst, analysiert und geeignete Maßnahmen eingeleitet werden. Um aber auf die erfassten Störgrößen in geeignetem Maße reagieren zu können, müssen elektronische Systeme zusätzlich autonome Eigenschaften besitzen, um selbstheilende Mechanismen einleiten zu können. Hierzu sind neue Entwurfsmethoden für Kommunikationsstrukturen, Daten- und Steuerpfade sowie Codierungsverfahren notwendig, die in den Modulen der Systeme eingesetzt werden.

Neben der Erforschung neuer Hardwareansätze ist auch die Betrachtung der Betriebssystemebene notwendig, um die Prinzipien der Selbstorganisation auch auf höheren Abstraktions- und Applikationsebenen fortzuführen. Das Betriebssystem muss dazu den Einsatz und die Steuerung von autonomen Komponenten unterstützen, damit das System bei einer Betriebsstörung reagieren kann. Hierzu bedarf es entsprechender Vorkehrungen im SoC-Entwurf:

- » kompatible Erweiterungen in der Entwurfsmethodik,
- » EDA-Werkzeuge zur Autonomie- und Zuverlässigkeitsbewertung sowie
- » die Unterstützung von hardwarenaher Software, wie zum Beispiel eines Betriebssystems.

Das AIS-Projekt hat sich zum Ziel gesetzt, eine neue Entwurfsmethodik für autonome integrierte Systeme zu erforschen. Zukünftige elektronische Systeme können dann bei Betriebsstörungen durch autonome Korrekturmaßnahmen eine Selbstheilung einleiten. Die Komponenten erkennen dabei selbständig ihren Zustand und geben diesen an die Betriebssystem-

ebene weiter, wenn die Betriebsstörungen nicht lokal kompensiert werden können. Die Ausbreitung von Fehlern und deren negative Auswirkungen auf die Zuverlässigkeit des MPSoCs werden dadurch an die nächst höhere Ebene weitergeben und nach Möglichkeit selbständig im Betrieb durch Neuorganisation der laufenden Dienste im MPSoC korrigiert.

Mit einer neuen Entwurfsmethodik werden Architekturkomponenten mit autonomen Eigenschaften erweitert und für den Systementwurf zur Verfügung gestellt. Diese Komponenten werden in einem Explorations- und Integrationsprozess auf Systemebene mit einer auf autonomes Verhalten ausgerichteten Betriebssystemumgebung kombiniert. Hierzu werden autonome Elemente im Systementwurf eingeführt, die Kontrollpfade, Datenpfade und Kommunikationsstrukturen der funktionalen Ebene überwachen und auf veränderte Betriebsbedingungen sowie Fehlverhalten reagieren. Dienste werden dazu erforscht, die das Betriebssystem zur internen Selbstoptimierung und Selbstheilung durch Replikation, Überwachung und Migration auf der Basis zuverlässiger Kommunikationsstrukturen einsetzen kann. Die Aufgaben der Forschungspartner sind in Abbildung 1.31 in der edaMatrix dargestellt.

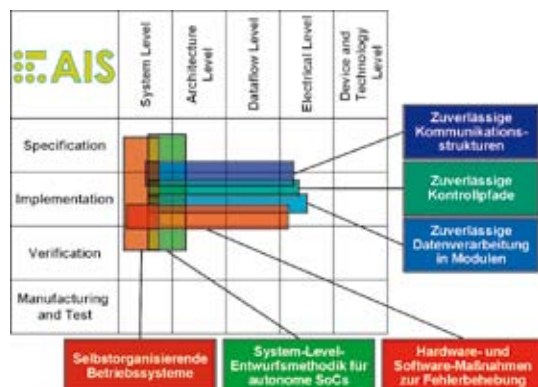


Abbildung 1.31: Die Aufgaben der Projektpartner in der edaMatrix

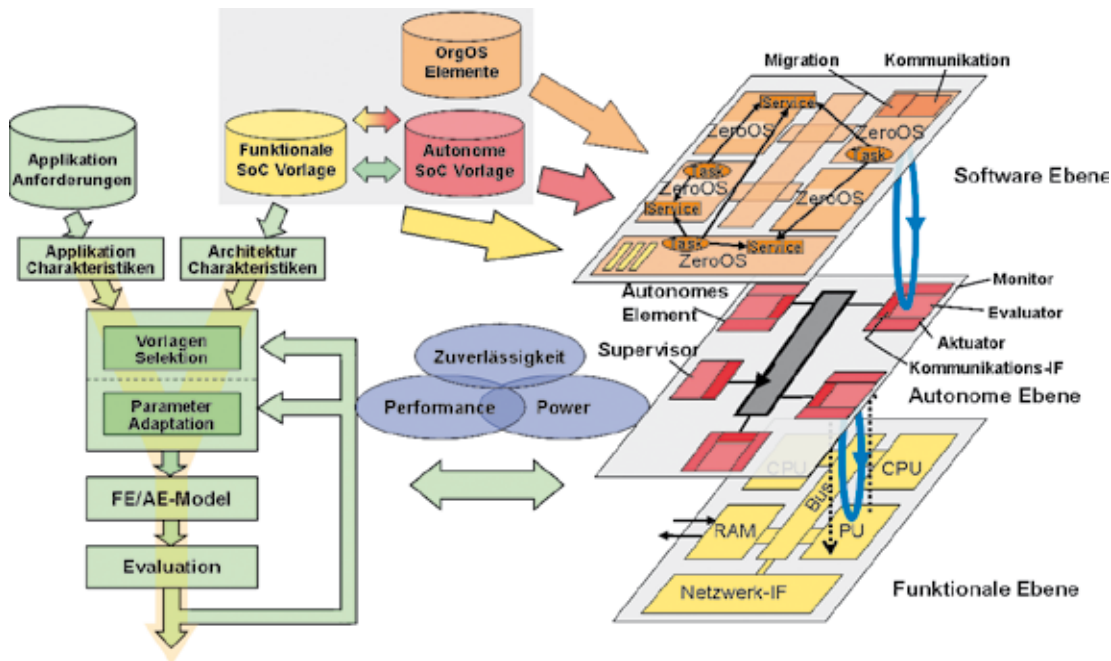


Abbildung 1.32: Drei Modellebenen unterstützen den Entwurf autonomer Systeme, wobei für jede Komponente ein Modell dimensioniert und für eine Systemexploration zur Verfügung gestellt wird.

Aufgaben im Projekt

Im Projekt werden Prinzipien der Selbstorganisation, dem Selbstschutz und der Selbstheilung für den Entwurfsprozess erforscht. In AIS konzentrieren sich die Forschungsaufgaben auf innovative Lösungen, die zu einer messbaren Verbesserung der Systemzuverlässigkeit beitragen, gleichzeitig aber Funktionsinvarianz bei geringem zusätzlichem Ressourcenbedarf sicherstellen. Zur Modellierung der zukünftig notwendigen Systemeigenschaften werden drei Verhaltensebenen genutzt: die funktionale, die autonome und die Betriebssystemebene, wie in Abbildung 1.32 zu sehen ist. Die autonome Verhaltensebene bildet dabei einen engen Verbund mit dem dazugehörigen Betriebssystem. Aus dem Systemmodell werden für die einzelnen Komponenten die relevanten Parameter ermittelt und ein zugehöriges Komponentenmodell evaluiert und optimiert. Diese Modellierungen einzelner Komponenten werden bei einem Systementwurf in den einzelnen Entwurfsschritten von der Spezifikation bis zur Implementierung berücksichtigt. Die folgenden Abschnitte beschreiben die individuellen Aufgaben und die ersten Ergebnisse der Forschungspartner.

System-Level-Entwurfsmethodik für autonome SoCs

Es wird eine Entwurfsmethodik für autonome integrierte Systeme erforscht, die bereits auf Systemebene eine Bewertung des Einsatzes von Verfahren zur Selbstorganisation und deren Auswirkungen auf die Zuverlässigkeit des Gesamtsystems zulässt. Dazu sind Prinzipien der Selbstorganisation auf Systemebene zu modellieren und im Spannungsfeld der Entwurfsparameter Zuverlässigkeit, Verlustleistung und Performanz in einem ganzheitlichen Ansatz zu analysieren. Unter Verwendung des Systemmodells sind dann autonome

Elemente so auszuwählen und den Komponenten der funktionalen Ebene zuzuordnen, dass gegebene Zuverlässigkeitsanforderungen per Konstruktion erfüllt werden.

Im ersten Jahr wurde ein Ansatz entwickelt, mit dem erstmalig temperaturabhängige Zuverlässigkeitseffekte in einer frühen Entwurfsphase analysiert werden können. Ausgehend von einer ausführbaren Spezifikation sowie unter Verwendung von Power-State-Machines für alle Systemkomponenten werden in einem ersten Schritt Aktivitätsverteilungen und daraus resultierend zeitliche und räumliche Leistungsverteilungen aller Systemkomponenten bestimmt. Die Leistungsverteilungen werden zusammen einem gegebenen Floorplan und einem Reliability-Block-Diagramm zu einer Zuverlässigkeitsbewertung herangezogen, die Alterungsfehler, welche durch einen Arrheniusansatz beschrieben werden, sowie durch Temperaturzyklen verursachte Fehler nach Coffin-Manson, unter zusätzlicher Berücksichtigung von strukturellen Redundanzen berechnet. Die entwickelte Methode wurde auf ein MPSoC angewendet. Es wurde dargestellt, wie sich die Temperaturen der Mikroarchitektureinheiten eines Prozessors während des Betriebs unterscheiden und sich auf die Systemzuverlässigkeit auswirken. Das Verfahren zeigt bereits jetzt ein hohes Potential zur frühzeitigen Zuverlässigkeitsbewertung von MPSoC-Architekturalternativen unter Berücksichtigung temperaturabhängiger Zuverlässigkeitseinflüsse.

Hardware- und Software-Maßnahmen zur Fehlerbehebung

Es werden Hardware- und Software-Maßnahmen zur Fehlerbehebung erforscht, um Änderungen der Kommunikationswege, der Lastverteilung, der Protokolle

oder der Taktrate im Fehlerfall zu ermöglichen. Diese verringern im Allgemeinen die Leistungsfähigkeit eines autonomen Systems. Vorhandene Echtzeitgarantien, die im fehlerfreien Betrieb gegeben werden können, sind in derartigen Situationen unter Umständen nicht mehr aufrecht zu halten. Es werden daher Verfahren erforscht, die es erlauben, die Sensitivität eines autonomen Systems gegenüber fehlerbedingten Änderungen seines Verhaltens zu analysieren und zu optimieren. Dabei werden sowohl transiente Fehler, die einmalige Korrekturmaßnahmen von begrenzter Dauer erfordern, als auch permanente Fehler, welche eine dauerhafte Rekonfiguration des Systems zur Folge haben, untersucht.

Im Bereich der permanenten Fehler wurde eine Migrationsstrategie erforscht, welche in kritischen Situationen Umverteilungen der Last in Hinblick auf die daraus resultierende Echtzeitfähigkeit evaluieren. Auf diese Art wird autonomen Diensten zur Durchführung von Prozessmigrationen ein Bewertungsmaß bezüglich des Zeitverhaltens zur Verfügung gestellt. Transiente Fehler, die aufgrund der notwendigen Korrekturen einen zeitlich begrenzten Overhead erzeugen, wirken sich direkt auf die Eigenschaften der aktuellen Konfiguration aus. In diesem Zusammenhang befindet sich derzeit ein Modell in der Untersuchung, welches durch eine Kopplung des Verhaltens von Komponenten im Fehlerfall mit probabilistischen Fehlermodellen die Zuverlässigkeit des Gesamtsystems bewertet. Dazu werden zunächst die genauen Konsequenzen von Fehlern bzw. des durch sie entstandenen Korrektur-Overheads in Hinblick auf die Performance untersucht. Darauf aufbauend können Randfälle, welche die Einhaltung vorgegebener Zeitbedingungen gerade noch gewährleisten, abgeleitet werden. In Kombination mit den gegebenen probabilistischen Fehlermodellen ist es anschließend möglich, Metriken zur Charakterisierung der Systemzuverlässigkeit anzugeben.

Zuverlässige Datenpfade

Neben der Systemebene und der Architekturplattform werden Strategien, Konzepte und spezifische Architekturweiterungen zum Zwecke der Zuverlässigkeitsoptimierung von MPSoC-Komponenten erforscht. Das Ziel ist die autonome Erkennung, Bewertung und Korrektur transienter und/oder permanenter Fehler in den Datenpfaden von eingebetteten RISC-Prozessoren. Transiente Logikfehler entstehen entweder durch ionisierende Strahlung (Soft Errors) oder durch Variationen aller Art (Temperatur, IR-Drop, Alterung). Besonderes Augenmerk liegt auf der Beherrschung von Mehrfachfehlern, die laut neuerer Veröffentlichungen an Relevanz zunehmen.

Im ersten Jahr wurden die zu betrachtenden Fehlermodelle konkretisiert, sowie eine Hintergrundforschung zu existierenden Fehlererkennungsmethoden durchgeführt. Als greifbares Ergebnis dieser ersten Projektphase liegt nun eine kompakte und präzise

Beschreibung des Technikstandes hinsichtlich Methoden und Techniken zur Verbesserung der Zuverlässigkeit von SoC-Komponenten vor. Des Weiteren wurde die Implementierung einer fehlertoleranten RISC-CPU-Pipeline basierend auf dem Leon-3-Prozessorkern von Gaisler begonnen.

Zuverlässige Kontrollpfade

Komplementär und ergänzend zur autonomen Fehlerbehandlung in Datenpfaden ist es notwendig, die Zuverlässigkeit von Kontrollpfaden in MPSoC-Komponenten mit geeigneten Hardware- und Software-Maßnahmen zu gewährleisten bzw. zu erhöhen. Speziell wird dies durch autonome Überwachungseinheiten erreicht, die Fehler in der Programmausführung von Prozessoren sowie der Kontrolllogik von Hardware-Modulen erkennen, bewerten und korrigieren. Ein besonderes Augenmerk wird auf auftretende Speicherfehler wie „Soft Errors“, aber auch auf gezielte lokale Attacken auf Kontrolllogikeinheiten gelegt. Dies beinhaltet sowohl die Kontrolle der korrekten Zustandsüberführung als auch die der korrekten Zustandsspeicherung. Es werden Methoden erforscht, die den Kontrollfluss überwachen sowie reversible und nichtreversible Maßnahmen im Fehlerfall einleiten, wie zum Beispiel die Terminierung durch erzwungene Überführung in „sichere“ Zustände oder automatische Ablaufkorrekturen.

Als erstes Ergebnis wurde ein Konzept erarbeitet, das die automatische Generierung von Überwachungsgraphen eines gegebenen Programms und Prozessors, ermöglicht. Weiterhin wurde eine Methode erforscht, wie basierend auf dieser formalen Programmdarstellung eine Überwachung der korrekten Kontrollflussbefehlsausführung durch einen Automaten durchgeführt werden kann. Um diesen Automaten nicht für jedes individuelle Programm neu synthetisieren zu müssen, wurden Konzepte einer generischen, mikroprogrammierbaren Implementierung des Überwachungsautomaten untersucht.

Zuverlässige Kommunikationsstrukturen

Um zuverlässige MPSoCs zu entwerfen, ist die zuverlässige Kommunikation zwischen den Modulen in autonomen integrierten Systemen ebenso wichtig wie die Erhöhung der Zuverlässigkeit der einzelnen Module. Ein weiteres Ziel ist deshalb die Untersuchung und Erforschung von selbstkalibrierenden Techniken und Codierungsverfahren, die die Zuverlässigkeit der Kommunikation steigern. Diese Techniken sollen die Anforderungen an die Zuverlässigkeit dynamisch zur Laufzeit an den aktuellen Systemkontext anpassen. Somit erlauben sie für vorgegebene QoS-Anforderungen (Quality of Service, z. B. Fehlerrate, Durchsatz, Latenz) hinsichtlich der Kommunikation eine optimale Abstimmung (Trade-off) zwischen Zuverlässigkeit, Energie und Performanz zu identifizieren. Diese Arbeiten erfolgen in enger Abstimmung mit den Partnern, die an der erhöhten Zuverlässigkeit in den Modulen forschen, um

Die Autoren des Artikels sind:

V. Schöber, A. Herkersdorf,
W. Stechele, J. Zeppenfeld,
A. Bouajila, R. Ernst,
M. Sebastian, O. Bringmann,
B. Sander, J. Teich, D. Ziener,
N. Wehn, M. May,
F. Rammig, K. Stahl.

Die Homepage von AIS ist über www.edacentrum.de/ais zu erreichen. Dort erhalten Sie weitere Informationen zum Projekt. Forschungs- und Industriepartner können über einen geschützten Bereich Projektberichte und weitere Publikationen herunterladen. Fragen hierzu beantwortet gerne Volker Schöber.

die Gesamtzuverlässigkeit des Systems zu maximieren. Die aus den Trade-offs abgeleiteten Kostenmodelle fließen in die System-Level-Entwurfsmethodik ein. Des Weiteren werden die erforschten Techniken als Basistechnologie für die Taskmigration im selbstorganisierenden Betriebssystem eingesetzt.

Die Berücksichtigung der Anwendungsebene beim Systementwurf wurde anhand einer Fallstudie quantitativ evaluiert. Für die Demonstration wurde ein WiMAX LDPC-Decoder gewählt. Er arbeitet nach einem probabilistischen Belief-Propagation-Algorithmus. Die Architektur wird von Speichermodulen und Verbindungen (Interconnects) dominiert und ist daher repräsentativ für heutige Systems-on-Chip-Lösungen. Die Implementierung wurde auf allen Entwurfsebenen von der Anwendungs- bis zur unteren Block-Ebene untersucht. Das Ziel ist, die Zuverlässigkeit des Gesamtsystems mit einem minimalen Mehraufwand an Hardware zu erhöhen. Durch Ausnutzung der inhärenten Fehlertoleranz des Algorithmus wird mit einem zusätzlichen Flächenaufwand von etwa 20 % die Dekodiergüte bis zu sehr hohen Raten von injizierten transienten Fehlern praktisch erhalten.

Selbstorganisierende Betriebssysteme

Das Ziel ist die Erforschung eines autonomen Betriebssystems für MPSoCs. Das System soll die autonome und funktionale Ebene unterstützen, aber auch gleichzeitig ein API (Application Program Interface) für die Anwendungen bereitstellen. Die Verarbeitungselemente, aus denen ein SoC besteht, haben zu beschränkte Ressourcen, um ein vollständiges Betriebssystem mit komplexer Funktionalität auszuführen. Aus dieser Motivation leitet sich das Grundkonzept des Betriebssystems ab: die von den Anwendungen angeforderten Dienste laufen verteilt unter verschiedenen Verarbeitungselementen. Des Weiteren wird die Verteilung genutzt, um die Fehlertoleranz zu erhöhen. Folglich können die autonomen Fähigkeiten des Betriebssystems nur mit Hilfe von Redundanz und Selbstoptimierung gewährleistet werden. Da die Ressourcen beschränkt sind, ist ihre effiziente Nutzung notwendig. Aus diesem Grund wird lastabhängig und dynamisch entschieden, wie die Dienste auf dem Prozessornetz verteilt werden. Einige Dienste, die so genannten fundamentalen Dienste, müssen lokal in jedem Knoten ausgeführt werden. Zum Beispiel sind die Kommunikationsdienste und Migrationdienste in allen Knoten zu finden. Bei der Verteilung der Dienste wird kontinuierlich eine Zielfunktion optimiert, die sich aus Parametern wie Prozessorauslastung, Speicherbelegung, Kommunikationskosten und Verlustleistung berechnet.

In den letzten Monaten wurden Maßnahmen für autonome Betriebssysteme evaluiert, die durch Reflektionsdienste das System in seiner Performanz beobachten und bewerten, sowie entsprechende Rekonfigurationsdienste einleiten kann. Arbeiten wurden gestartet, die autonomen Dienste in einer separaten autonomen

Betriebssystemschiicht zu verankern. Diese bilden die Schnittstelle zwischen der Hardwareabstraktion und der Schicht der Anwendungsdienste, damit eine Analogie zwischen der Betriebssystemarchitektur und der Gesamtsystemarchitektur hergestellt wird.

Resümee

In den ersten Monaten konnte das Konzept zum Entwurf autonomer integrierter Systeme weiter verfeinert und validiert werden. Die Ergebnisse im Projekt werden in Form von prototypischen Implementierungen von allen Projektpartnern geprüft. Die Interoperabilität wird demonstriert. Hierzu steht eine FPGA-basierte Rapid-Prototyping-Umgebung der Firma Gaisler Research jedem Forschungspartner als Demonstrator zur Verfügung. Dies ermöglicht, dass jeder Partner die Verfahren und Interoperabilität des gesamten Projekts nutzen kann.

Aktuelle Publikationsliste des Projekts

- [1] W. Stechele, O. Bringmann, R. Ernst, A. Herkersdorf, K. Hojenski, P. Janacik, F. Rammig, J. Teich, N. Wehn, J. Zeppenfeld, D. Ziener: „Concepts for Autonomic Integrated Systems“, *edaWorkshop 07, Hannover, Juni 2007.*
- [2] W. Stechele, O. Bringmann, R. Ernst, A. Herkersdorf, K. Hojenski, P. Janacik, F. Rammig, J. Teich, N. Wehn, J. Zeppenfeld, D. Ziener: „Autonomic MPSoCs for Reliable Systems“, *Zuverlässigkeit und Entwurf, München, März 2007.*
- [3] B. Sander, O. Bringmann, W. Rosenstiel: *Applikationsspezifische Zuverlässigkeitsbewertung von MPSoCs, edaWorkshop, Hannover, Juni 2007.*
- [4] B. Sander, O. Bringmann, W. Rosenstiel: *Temperaturabhängige Zuverlässigkeitsbetrachtungen auf Systemebene, Vortrag, Fach- und Kooperationsworkshop "Technologie und deren Auswirkung", Dresden, Mai 2007.*
- [5] B. Sander, O. Bringmann, T. Schönwald, J. Schnerr, W. Rosenstiel: *Applikationsspezifische Zuverlässigkeitsbewertung von Multiprocessor Systems-on-Chip auf Systemebene, Zuverlässigkeit und Entwurf, GMM-Fachbericht, VDE Verlag, Berlin, März 2007.*
- [6] M. Alles, T. Brack, N. Wehn, "A Reliability-Aware LDPC Code Decoding Algorithm," in *Proc. 56th Vehicular Technology Conference (VTC Spring '07), Dublin, Ireland, Apr. 2007.*
- [7] M. May, N. Wehn: "A Case Study in Reliability-Aware Design: A Resilient LDPC Code Decoder," *Accepted for publication, Design, Automation and Test in Europe (DATE '08), Munich, Germany, Mar. 2008.*

Kont@kt
 Dr. Volker Schöber
 fon: (05 11) 7 62 – 1 96 88
 schoeber@edacentrum.de