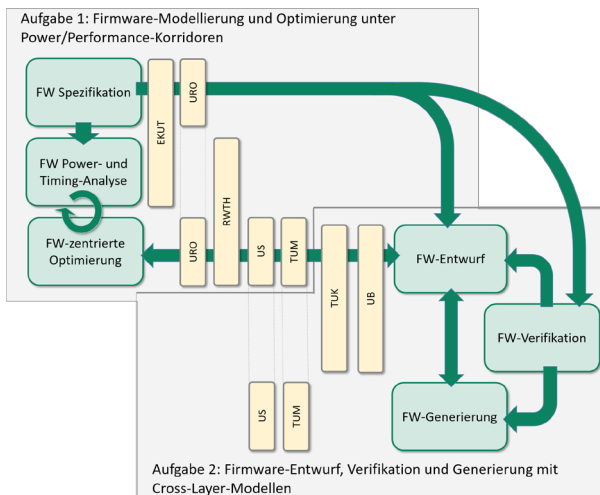


Motivation

Die Durchdringung der Mikroelektronik in eine Vielzahl von Steuerungs- und Sensor-Geräten ist ein anhaltender Trend, der sich in verschiedenen Anwendungsdomänen widerspiegelt, sei es in der Automobiltechnik, in der Energieversorgung bei Power-Control-Geräten oder in der Mobilkommunikation. Diese Anwendungen stellen hohe Anforderungen an die maximal zulässige Leistungsaufnahme sowie den elektrischen Energieverbrauch und erfordern gleichzeitig die Einhaltung strikter applikations-spezifischer Zeitbedingungen. Dazu muss die Firmware stets an sich ändernde Hardware-Parameter sowie Architektur- und Systemkonfiguration angepasst und zusätzlich hinsichtlich gegebener Anwendungs- bzw. Nutzungsszenarien ausgelegt werden.

An dieser Stelle setzt CONFIRM an und erforscht notwendige Schlüsselaspekte zur späteren Realisierung eines durchgängigen, automatisierten Firmware-Entwurfsprozess. Hierzu zählt die modellbasierte Firmware-Spezifikation mit der Fähigkeit zur automatisierten Komposition der Firmware aus Bibliothekselementen inklusive einer optimierten Auslegung im Hinblick auf Echtzeitfähigkeit, Energieverbrauch unter Berücksichtigung der Hardware- und Speicherarchitektur sowie von Anwendungsszenarien. Um die Korrektheit der generierten Firmware unter diesen unterschiedlichen Aspekten nachzuweisen sind ferner Verifikationsansätze erforderlich, um im Modell spezifizierte Garantien korrekt auf die erzeugte Implementierung abzubilden und zu verifizieren.



Arbeitspaketstruktur von CONFIRM

Forschungspartner



Projektkoordination



Kontakt

Dr. Jürgen Haase
(Projektkoordination)
edacentrum GmbH
fon +49 511 762-19698
haase@edacentrum.de

Dr. Dieter Treytnar
(Projektmanagement)
edacentrum GmbH
fon +49 511 762-19687
treytnar@edacentrum.de



Das Projekt CONFIRM wird unter den Förderkennzeichen 16ES0564-70 im Förderprogramm IKT 2020 durch das Bundesministerium für Bildung und Forschung (BMBF) gefördert sowie durch die Industriepartner Infineon Technologies AG, Robert Bosch GmbH, Intel Deutschland AG und Mentor Graphics GmbH unterstützt.



Automatisierter Firmware-Entwurf unter Berücksichtigung von Timing- und Power-Budgets für anwendungsspezifische Elektroniksysteme

BMBF-Clusterforschungsprojekt

Laufzeit: 01.01.2017 - 31.12.2019

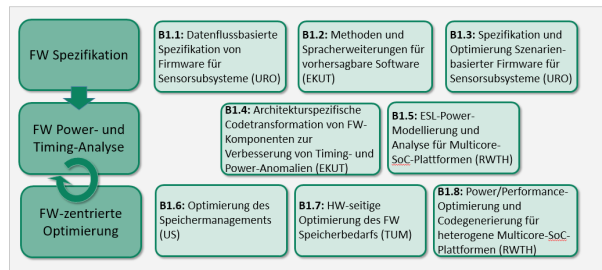
Projektkurzbeschreibung

In CONFIRM erforschen sieben Forschungseinrichtungen Schlüsselaspekte für einen effizienten, automatisierten Entwurf von Firmware, die speziell für die Hardwaregegebenheiten des Systems, aber auch die angestrebten Anwendungsszenarien ausgelegt ist. Hierfür werden gemeinsame Hardware-Firmware-Modelle entwickelt, die eine abstrakte Optimierung bezüglich verschiedener Parameter zulassen. Darauf aufbauend wird eine neue Entwurfsmethode erarbeitet, mit deren Hilfe eine an die Systemkonfiguration und das Nutzungsprofil angepasste Firmware automatisiert erstellt werden kann.



Aufgabe 1: Firmware-Modellierung und Optimierung unter Berücksichtigung von Power/Performance-Korridoren-Maß

In Aufgabe 1 werden neuartige Ansätze zur Spezifikation von Firmware, verbesserte Ansätze zur Analyse von Firmware-Performance und -Power sowie Firmware-spezifische spezifische Verfahren zur Optimierung des Speicherabdrucks und Speicherlayouts sowie zur Performance/Power Optimierung untersucht. Die Beiträge sind dabei auf der Systemebene angesiedelt und überwiegend modellbasiert, weshalb ein Schwerpunkt auf die Modellierung gelegt wird. Die Ergebnisse der Beiträge werden größtenteils auf Modell- oder Quelltextebene repräsentiert. Betrachtete Zielarchitekturen variieren von Single-Mikro-Controller-Architekturen mit sehr beschränkten Ressourcen bis hin zu heterogenen Multi-Core-SoC-Plattformen.



Beiträge zur Aufgabe 1

In einer Top-Down Darstellung beginnt die Aufgabe mit der Spezifikation. Für Sensorsysteme wird eine neuartige modulare Spezifikationsmethodik auf Basis von zeitbehafteten Datenflussmodellen entwickelt. Diese bildet den Ausgangspunkt zur Generierung periodisch geplanter Firmware-Komponenten sowie der Integration geeigneter Re-Sampling Filter (B1.1). Daneben wird eine Makro- und Template-basierte C/C++-Spracherweiterung zur Beschreibung von Timing- und Power-vorhersagbaren Software entwickelt. Dadurch soll generisch implementierte Firmware für gegebene Anwendungsszenarien spezialisiert und somit die Timing- und Power-Vorhersagbarkeit erhöht werden (B1.2). Anwendungsszenarien werden ebenfalls im Kontext der Datenfluss-basierten Spezifikation von Firmware untersucht (B1.3).

Im Bereich der Analyse der Firmware-Performance und -Power werden u.a. Methoden zur Transformation sowohl des Quelltextes als auch des zugehörigen Binärcodes untersucht, um die dynamische und statische Vorhersage von Programmlaufzeiten zu verbessern (B1.4). Auf der Ebene heterogener Multi-Core-SoCs erfolgt eine Entwicklung schneller und genauer White-Box- und Black-Box-Power-Modelle für Multicore-SoC-Komponenten inklusive verschiedener Betriebszustände. Durch Integration der Power-Modelle in eine SystemC-basierte Virtuelle Plattform wird eine schnelle Analyse des zeitlichen Verlaufs der Leistungsaufnahme ermöglicht (B1.5).

Firmware-spezifische Optimierungen werden in drei Beiträgen untersucht. Es werden neuartige Hardware-seitige Verfahren zur Generierung speichereffizienter Firmware entwickelt. Im Gegensatz zu bekannten Verfahren soll hierbei auch die HW/SW-Schnittstelle angepasst werden (B1.7). Daneben wird Optimierung des Speicher-Layouts mit dem Ziel untersucht, den Energiebedarf durch selektiven Betrieb der Speicher zu senken (B1.6). Im Bereich der heterogenen Multi-Core-SoC-Plattformen wird ein Software-Mapping zur gemeinsamen Performance/Power-Optimierung entwickelt. Dabei wird die Einhaltung von Power-Korridoren als Randbedingung berücksichtigt (B1.8).

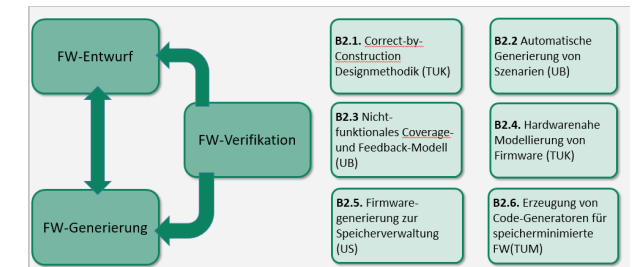
Aufgabe 2: Firmware-Entwurf, Verifikation und Generierung mit Cross-Layer-Modellen

Die aus den Systemmodellen (Aufgabe 1) zu entwerfende Implementierung des Hardware/Firmware-Systems soll sich in Bezug auf nicht-funktionale Parameter wie Leistungsaufnahme, Performanz und Temperatur vorhersagbar verhalten, indem die Hardware- und Softwarekomponenten dynamisch zur Laufzeit nach Maßgabe vorgegebener Parameterbedingungen (Constraints) konfiguriert werden.

Die Abstraktionsebene der Implementierung stellen RTL-Beschreibungen für die Hardware und Binärkode (Assembler- oder Maschinenprogramme) für die Software dar. Auf dieser Ebene lassen sich für einzelne Systemkomponenten wegen der feingranularen Modellierung genaue funktionale und nicht-funktionale Analysen durchführen. Allerdings sind Simulationen des Gesamtsystems aus Komplexitätsgründen auf dieser Ebene nicht möglich. Dazu können stattdessen die abstrakten Modelle der Systemebene dienen, sofern sie eine hohe Genauigkeit in Bezug auf die tiefer liegenden Abstraktionsschichten aufweisen und möglichst in einem wohldefinierten Zusammenhang zu der konkreten Implementierung stehen.

Es soll eine Entwurfsmethodik für Firmware erforscht und entwickelt werden, die ausgehend von abstrakten Modellen systematische Entwurfsverfeinerungen unterstützt, um die oben genannten Entwurfsziele zu erreichen (B.2.1).

Für bestimmte Funktionalitäten des Systems (Speicherverwaltung) soll die Firmware dabei automatisch generiert werden (B.2.5). Da Systeme zur Entwurfszeit in Hinblick auf schwer vorhersehbare Kundenanwendungen und zum Zweck der Wiederverwendbarkeit von Komponenten über zahlreiche Konfigurationsmöglichkeiten verfügen, soll eine automatische Anpassung der Firmware auf den speziellen Anwendungsfall durch Firmwaregeneratoren unterstützt werden (B2.6).



Beiträge zur Aufgabe 2

Auf Systemebene werden zur Verifikation der Einhaltung von Power-Constraints und zur Festlegung der Powerstrategien neue Verfahren der Constrained-Random-Verifikation erforscht und entwickelt (B.2.3, B.2.2). Die für die korrekte Implementierung der Powerstrategien notwendigen funktionalen Voraussetzungen sollen ebenenübergreifend durch Garantien erfasst und nachgewiesen werden (B.2.1). Dazu sollen Berechnungsmodelle für die Anwendung formaler Beweistechniken entwickelt werden (B.2.4).

Die Arbeiten in CONFRM sollen im Sinne der Empfehlungen einschlägiger Sicherheitsstandards wie der Norm ISO-26262 im Automobilbereich durchgeführt werden. Dies beinhaltet die Beachtung von Empfehlungen dieser Norm bei der Entwicklung von Codegeneratoren (B.2.5, B.2.6). Weitere Beiträge im Sinne von ISO-26262 sind die Ausdehnung des Anwendungsbereichs formaler Technik von der Hardware auf die Firmware (B.2.4), die Durchgängigkeit des Verifikationsablaufs unter Einbindung abstrakter Systemmodelle (B.2.1) sowie die Quantifizierung der auf Systemebene durch die Verifikationstechniken erzielte Abdeckung (B.2.3).