

ENTWURFSASPEKTE FÜR HOCHINTEGRIERTE STEUERGERÄTE MIT UNTERSCHIEDLICHEN ASIL-STUFEN

Die Hochintegration von Fahrzeugfunktionen mit unterschiedlichen ASIL-Stufen bildet die Basis für viele Trends in der Fahrzeugelektronik, führt aber auch zu neuen Wechselwirkungen im Zeitverhalten dieser Systeme. Audi und Syntvision haben deshalb die bestehenden Entwurfsmuster an die geänderten Herausforderungen angepasst.

AUTOREN



DR. KARSTEN SCHMIDT

ist zuständig für die Themen Autosar und Architektur bei der Audi AG in Gaimersheim.



MARKUS BUHLMANN

ist Leiter für die Bereiche Fahrdynamik und Software in der Abteilung Entwicklung Fahrwerk-elektronik bei der Audi AG in Ingolstadt.



CHRISTOPH FICEK

ist Projektmanager Autosar und Safety bei der Syntavision GmbH in Braunschweig.



DR. KAI RICHTER

ist Gründer und Technischer Geschäftsführer (CTO) der Syntavision GmbH in Braunschweig.

MOTIVATION

Automobilhersteller müssen heute neue, innovative Funktionen realisieren, ohne die Anzahl der Steuergeräte weiter zu erhöhen, um unter anderem die Kosten, das Gewicht und den Energieverbrauch zu begrenzen. Dies führt zu hochintegrierten Multi-Funktions-Steuergeräten, die meist auf der Basis von Autosar realisiert werden.

Die kostengünstige und zuverlässige Dimensionierung und Auslegung solcher hochintegrierter Multi-Funktions-Steuergeräte erfordert optimierte Architekturkonzepte, beginnend von der Funktionsarchitektur über die Software-Architektur bis hin zur Steuergerätearchitektur. Die Architekten haben dabei die Aufgabe, geeignete Konzepte auszuwählen, und falls notwendig, Optimierungen und Anpassungen an der Gesamtarchitektur vorzunehmen, und zwar so früh wie möglich im Entwicklungsprozess.

Neu ist dabei die Koexistenz von Funktionen unterschiedlicher ASIL-Stufen nach ISO26262 auf einem Steuergerät. Ganz besonders gilt dies für die Domäne der Fahrwerksregelungen, denn dort existieren viele sicherheitsrelevante Systeme. In ❶ ist schematisch eine Integration von Software mit unterschiedlichen Sicherheitsanforderungen dargestellt. Systeme mit gemischten Kritikalitätsstufen sind neu, da bisher die einzelnen Software-Funktionen auf den jeweiligen Steuergeräten isoliert umgesetzt wurden. In Zukunft werden Systeme mit heterogenen Sicherheitsanforderungen häufiger anzutreffen sein. Für die Integration derartiger Steuer-

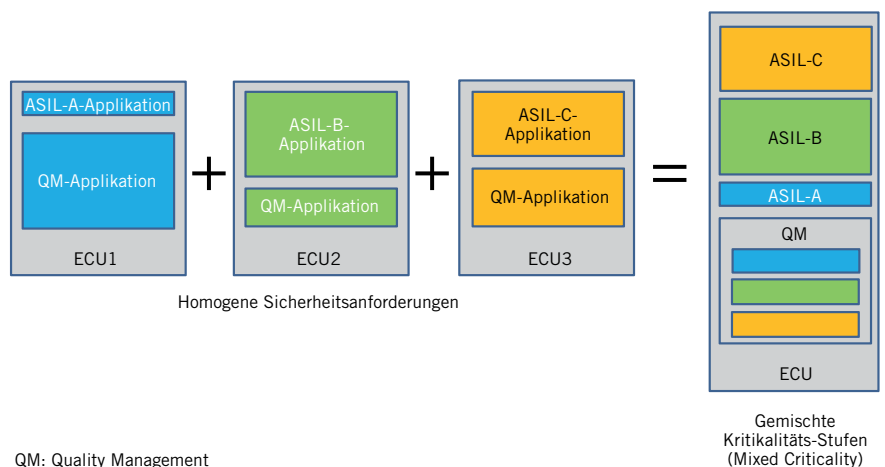
geräte gelten zwei elementare zeitliche Anforderungen:

- : In allen Situationen muss den relevanten Funktionen ausreichend Rechenzeit zur Verfügung stehen, um eine Überlastung des Steuergeräts zu vermeiden und alle Zeitbedingungen sicher einzuhalten.
- : Eine sicherheitsrelevante Funktion darf beispielsweise nicht durch ein unerwartetes Verhalten einer Applikation mit niedrigerer ASIL-Einstufung dauerhaft gestört werden (ISO 26262, Interferenzfreiheit (Freedom from Interference)).

ENTWURFSANSÄTZE IM WIDERSPRUCH

Bei dieser Kombination von Anforderungen ergeben sich aus den heute etablierten Entwurfsansätzen widersprüchliche Vorgaben für die Integration. Dies betrifft insbesondere die Partitionierung der Software und die Planung der Abläufe. Auf reines Echtzeitverhalten optimierte Architekturen halten nur selten alle Sicherheitsanforderungen ein, während die sicheren Architekturen meist eine schlechte Ressourcennutzung aufweisen. Am folgenden Beispiel lässt sich dies eindrucksvoll demonstrieren.

❷ enthält die essenziellen Ressourcen- und Sicherheitsanforderungen der unterschiedlichen Anwendungen sowie die maximalen Laufzeitvorgaben (Deadlines), die hier der Zykluszeit entsprechen. Eine der wichtigsten bei der Integration festzulegenden Eigenschaften ist die Priorität für die Ablaufplanung (Scheduling). Nach der Rate-Monotonic-Scheduling-Methode



❶ Integration von drei Applikationen mit gemischten Kritikalitäts-Stufen

APPLIKATION / TASK	ASIL-STUFE	ZYKLUSZEIT/MAX. LAUFZEIT (DEADLINE) [MS]	LAUFZEIT [MS]	RMS-PRIORITÄT	CAPA-PRIORITÄT
T1	ASIL-C	25	2,5	4	10
T2	ASIL-B	10	1,5	8	8
T3	ASIL-A	1	0,2	10	6
T4	QM	10	4	6	4

2 ASIL-Stufen, Prioritäten und ASIL-Stufen des Beispielsystems

(RMS) [2] wird die Priorität nach Zykluszeit vergeben: je kleiner die Zykluszeit, desto höher die Priorität. Nach der Criticality-Aware-Priority-Assignment-Methode (CAPA) [5] erhält die Task mit der höchsten Sicherheits-Stufe die höchste Priorität. 2 zeigt die Prioritäten verteilt nach den jeweiligen Methoden.

Keine der beiden Methoden liefert ein zufriedenstellendes Ergebnis. Die aus der RMS-Methode resultierenden Prioritäten ergeben einen Ablauf der Tasks ohne Deadline Verletzung, erfüllen jedoch die Forderung nach Interferenzfreiheit nicht. Die ASIL-C-Applikation kann von allen anderen Applikationen unterbrochen werden (Kritikalitätsinversionen). Ein Fehler, beispielsweise in der ASIL-A-Funktion, kann also die ASIL-C-Funktion dauerhaft verdrängen. 3 zeigt die Abläufe anhand eines Gantt-Diagramms. Die RMS-Methode liefert also keine hinreichende Sicherheit. Bei der CAPA-Methode sind derartige Kritikalitätsinversionen ausgeschlossen. Allerdings werden die Ressourcen, insbe-

sondere die CPU-Zeit, nicht effizient durch die Ablaufplanung genutzt. 4 zeigt den Ablauf, in dem als Folge der geänderten Randbedingungen Verletzungen der Deadline von Task T3 auftreten und T4 seine nur knapp einhalten kann.

Das Beispiel zeigt die Wechselwirkung unterschiedlicher Entwurfsansätze und Sicherheitsanforderungen auf die Einhaltung der Echtzeiteigenschaften. Mit den bestehenden Methoden lassen sich jedoch nicht beide Anforderungen gleichzeitig erfüllen. Also müssen neue Entwurfparadigmen gefunden oder die bestehenden an die neuen Bedingungen angepasst werden.

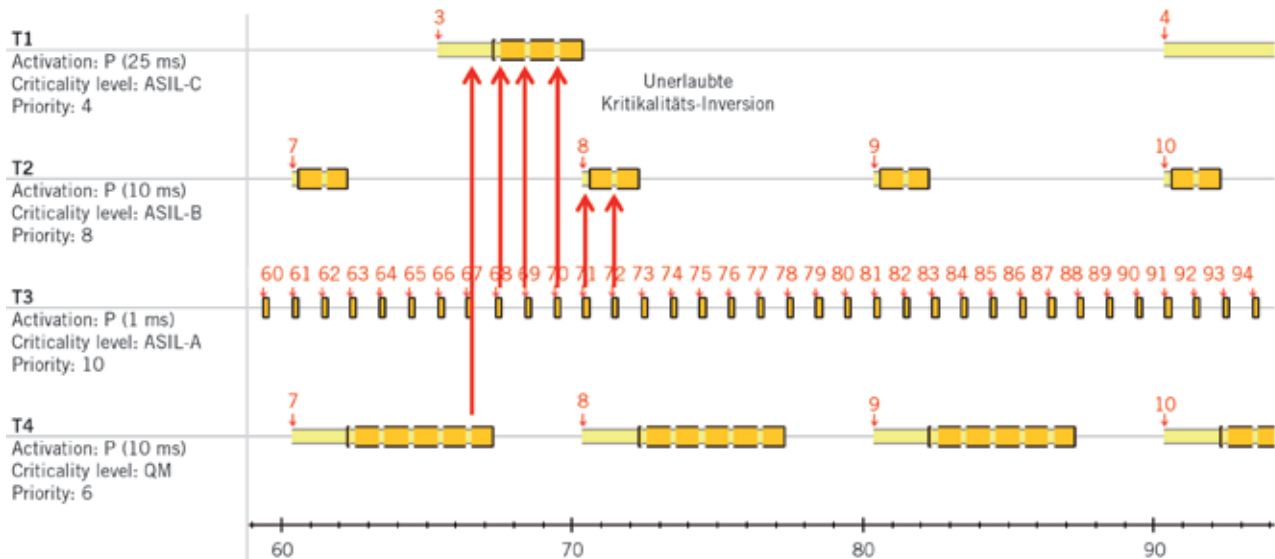
MASSNAHMEN

Da beide Entwurfsansätze bezüglich der Priorität auf unterschiedlichen Zielsetzungen beruhen, müssen andere Maßnahmen ergriffen werden:
 : Umgehen des Problems durch Anheben der ASIL-Stufen aller Applikationen

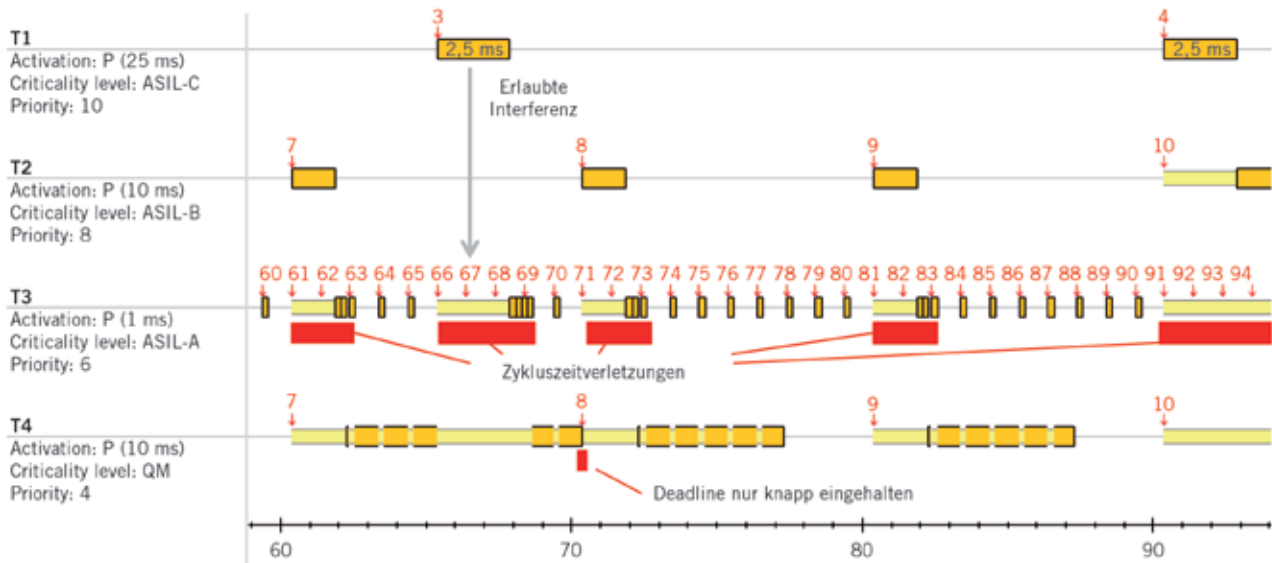
- : Steigerung der zur Verfügung stehenden Rechenleistung durch Einsatz leistungsfähigerer Hardware
 - : Vermeidung des Widerspruchs durch Anpassen der Funktions- und Software-Architektur
 - : Erweiterung des Entwurfsraums für den Schedule (Timing Protection)
 - : Kombinationen aus den oben genannten Maßnahmen.
- Die ersten zwei hier gelisteten Maßnahmen sind sehr kostenintensiv und stellen daher kaum gangbare Alternativen dar.

ANPASSUNGEN DER TIMING-ARCHITEKTUR

Die vielversprechendste Lösung besteht in der Anpassung der Funktions- und Software-Architektur derart, dass insbesondere die größten Effizienzprobleme eines CAPA-basierten Ablaufplans von vornherein vermieden werden. Was also macht CAPA-Abläufe so ineffizient? Es sind die seltenen und dann sehr lang laufenden Applikatio-



3 Zeitlicher Software-Ablauf nach RMS (Rate Monotonic Scheduling)



④ Zeitlicher Ablauf nach CAPA-Maßnahmen (Criticality Aware Priority Assignment)

nen, also diejenigen Applikationen mit großen Zykluszeiten, relativ großen Ausführungszeiten und einer gleichzeitig hohen ASIL-Stufe. Dies betrifft Task T1 in unserem Beispiel und ist in ④ sehr gut erkennbar. Die Tasklaufzeit überschreitet die Zykluszeiten der niederprioritären Tasks und verdrängt diese daher über deren Zykluszeit hinaus.

Wenn wir nun T1 auf mehrere, dafür kürzer laufende, Tasks verteilen, erhalten wir deutlich verbesserte Zeitabläufe. Die Blockierungszeiten werden kürzer. Es bleibt mehr Rechenzeit für niedrigere Prioritäten „übrig“, und in der Folge gibt es weniger Zykluszeitverletzungen. Im vorliegenden Fall können wir die Zykluszeit von T1 (Zykluszeit 25 ms, 10 Runnables) auf 5 ms reduzieren und jeweils nur zwei der Runnables ausführen. Dabei bleibt der

Gesamtzyklus unverändert: alle zehn Runnables werden innerhalb des Zyklus von 25 ms genau einmal ausgeführt, nur nicht als Einheit. Diese Anpassung wird auch als Zykluszeitanpassung (Period Transformation [4]) bezeichnet.

ENTWURFSRAUMERWEITERUNG DURCH AUTOSAR TIMING PROTECTION

Timing Protection ist ein Systemservice und überwacht zur Laufzeit auf dem Steuergerät die Ausführungszeiten der Tasks. Überschreiten diese die eingestellten Höchstgrenzen, wird dies als Fehler interpretiert und eine konfigurierbare Fehlerbehandlung ausgeführt. Durch die Timing Protection kann im Falle einer Kritikalitätsinversion die Interferenz begrenzt werden und genügt den Sicherheitsanforderungen nach ISO 26262.

Ist Timing Protection im System verfügbar, wird der Entwurfsraum für den Schedule erweitert, denn Kritikalitätsinversionen sind nicht mehr ausdrücklich verboten, sondern können gezielt abgesichert und ausgenutzt werden. Einschränkungen ergeben sich aus der Zertifizierung der Timing Protection selbst. Prinzipiell sollte die Timing Protection nach dem gleichen oder höheren ASIL-Level entwickelt sein wie die Anwendung mit der höchsten Einstufung. Der Einsatz einer Timing Protection macht es zudem erforderlich, die kritischen Grenzbereiche des Schedules zuverlässig vorherzusagen.

⑤ zeigt, dass nun alle Anforderungen eingehalten sind. Es gibt keine unerlaubten Interferenzen, und alle Tasks halten ihre Zykluszeiten zuverlässig ein.

TOSHIBA
Leading Innovation >>>

> UNSERE KFZ-ELEKTRONIK: DEFINITIV AUF DER ÜBERHOLS PUR

Bei der Entwicklung moderner neuer Halbleiterlösungen für Anwendungen in der Kfz-Technik gibt Toshiba weiterhin Gas.

Unsere aktuellen integrierten und skalierbaren SOC-Lösungen aus der Capricorn-Reihe sind für Anwendungen mit hohen Qualitätsanforderungen wie z.B. Kombiinstrumente oder Head-up-Displays konzipiert. Der Capricorn-F, ein Mitglied unserer Capricorn Familie, bietet Unterstützung für 3D-Grafik und sorgt dafür, dass grafisch animierte Szenen, wie z.B. Zeigerrotationen für Drehzahl und Geschwindigkeit, hochwertig anmutend auf dem Display abgebildet werden.

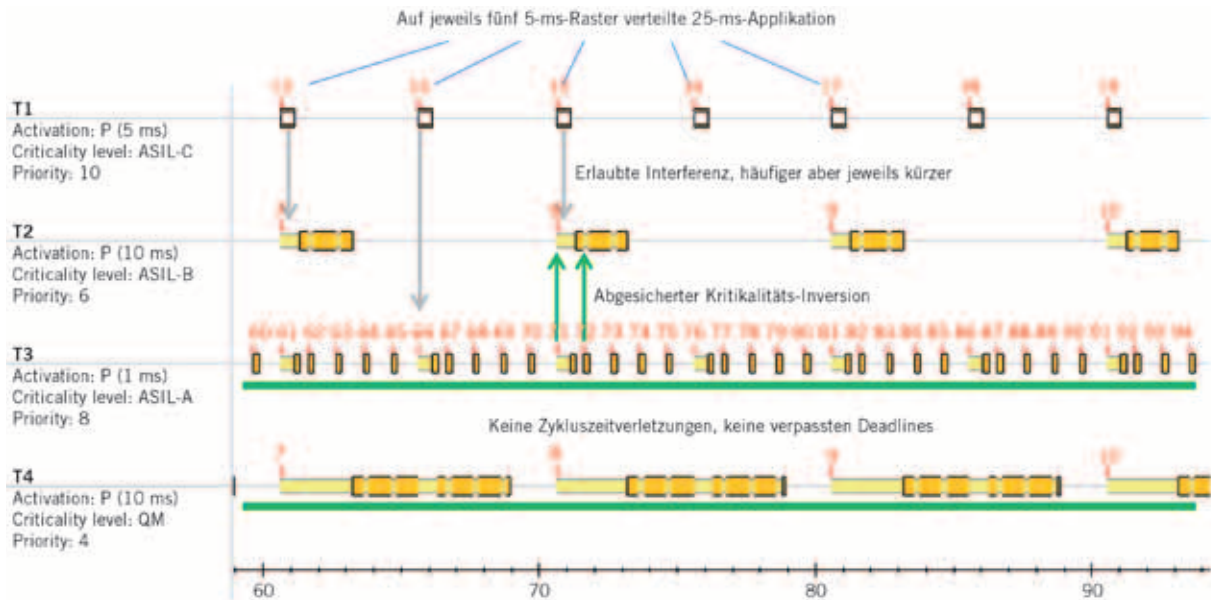
Auch für die Sicherheit in den Fahrzeugen der Zukunft ist gesorgt: Die ARM Cortex™ M3-Mikrocontroller von Toshiba adressieren die Anforderungen der ISO26262 zur funktionalen Sicherheit von Straßenfahrzeugen und reduzieren Hardware- und Software-Overhead.

Und dies sind nur einige wenige Beispiele für die zahlreichen Innovationen im Toshiba-Produktangebot für den Automobilbau, die dazu beitragen, dass wir bei der Entwicklung neuer Lösungen eine führende Rolle spielen.

Besuchen Sie uns noch heute auf www.toshiba-components.com/automotive



SCAN ME



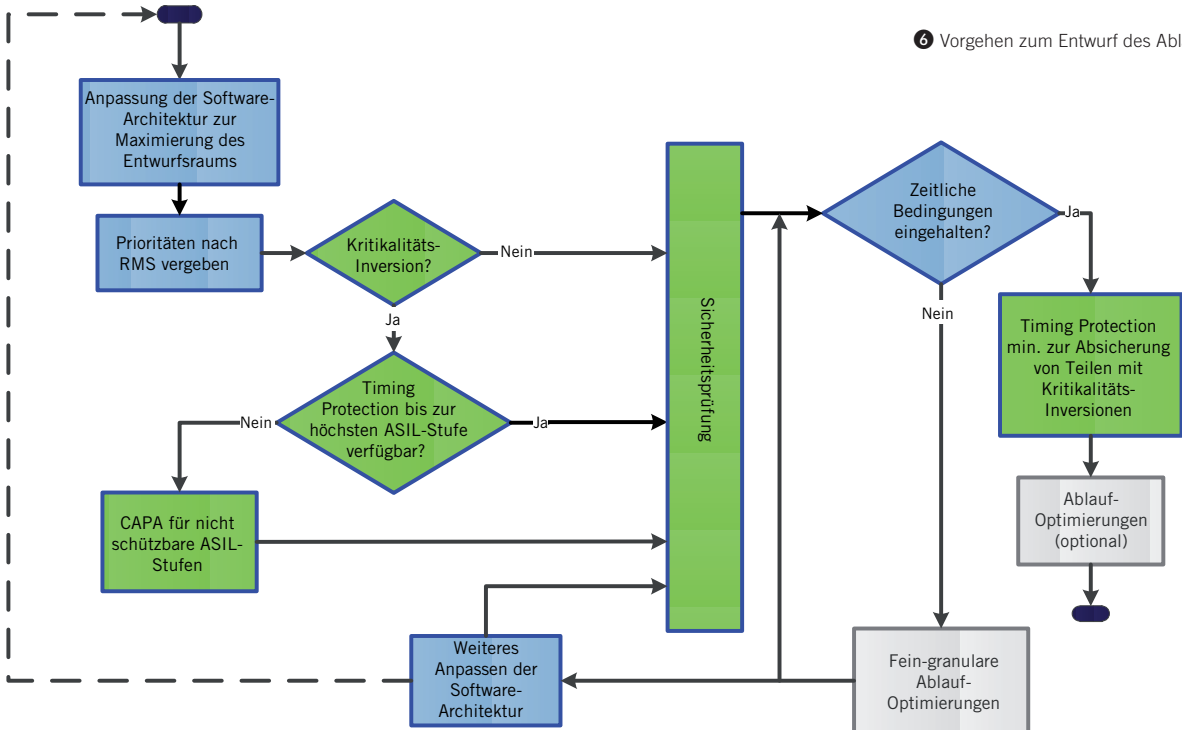
5 CAPA mit Raster-Transformation (zusätzlich Timing Protection)

VORGEHEN IN DER ENTWICKLUNGSPRAXIS

Aus den beschriebenen Maßnahmen ergibt sich das in 6 gezeigte Vorgehensmodell. Zunächst sollte ein Ablaufplan nach RMS entworfen werden, dessen Prüfung auf Kritikalitätsinversionen vergleichsweise einfach ist. Im negativen

Fall wird der Entwurfsraum durch die Timing Protection erweitert. Idealerweise ist diese für die gesamte Software-Architektur verfügbar, ansonsten bis zum verfügbaren ASIL-Level, wobei dann für nicht schützbares ASIL-Stufen CAPA zum Einsatz kommt. Die Sicherheitsprüfung kontrolliert, ob der Schedule per Design zertifizierbar ist. Anschließend wird der

Schedule auf die Einhaltung der zeitlichen Vorgaben untersucht. Sind diese erfüllt, wird die Timing Protection für den Schedule konfiguriert und optionale Optimierungen für weitere Effizienzverbesserungen können folgen. Bei Verletzung der zeitlichen Parameter sind Maßnahmen erforderlich wie zum Beispiel die Anpassung der Software-Architektur.



6 Vorgehen zum Entwurf des Ablaufplans

Um die beschriebenen Optimierungen bereits in einer frühen Phase der Entwicklung durchführen zu können, benötigen wir neben den typischerweise vorhandenen Konfigurationsinformationen (Tasks, Zykluszeiten, ASIL-Stufen) auch Informationen zur Ausführungszeit der Tasks beziehungsweise Runnables. Für das Treffen der grundlegenden Entscheidungen sind auch grobe Abschätzungen zielführend, eine hohe Genauigkeit wird nicht benötigt.

ARCHITEKTUR UND AUTOSAR

Bei der Entwicklung automobiler Steuergeräte sind sehr unterschiedliche Architekturaspekte zu berücksichtigen. Nach [3] existieren für eingebettete Systeme fünf primäre Architektursichten:

- : Sicht auf Funktionsarchitektur und
- : Vernetzungsarchitektur
- : Sicherheit und Zuverlässigkeitssicht
- : Timing und Ressourcensicht
- : Integrationsicht.

Typischerweise erfolgt der Entwurf der Funktionsarchitektur losgelöst von Integrationsaspekten. Jedoch beeinflusst ein Funktionsentwickler durch seine Entwurfsentscheidungen stark die spätere Integration [1]. Dazu zählt zum Beispiel die Möglichkeit, eine Zykluszeitanpassung durchführen zu können. Es bietet sich an, Funktionen in eine entsprechende Anzahl von Runnables zu teilen. Dadurch ergeben sich bei der Integration mehr Freiheitsgrade, um die zeitlichen Anforderungen erfüllen zu können.

Die Berücksichtigung der unterschiedlichen Architekturaspekte erfordert eine Stärkung der Architekturtätigkeiten. Die konsequente Umsetzung von Architekturprinzipien hilft bei späteren Optimierungen, beginnend von der Funktionsarchitektur bis zur Steuergerätearchitektur. Mit Autosar existiert schon heute eine Basis für die Integration unterschiedlicher Funktionen, die alle genannten Architekturaspekte bei konsequenter Nutzung unterstützt.

FAZIT

In gemeinsamen Projekten haben Audi und Syntavision die vorgestellte Entwurfsmethodik erfolgreich eingesetzt. So konnten Architekturvarianten für zukünftige hochintegrierte Steuergeräte untersucht werden. Damit ist gezeigt, dass der grundlegende Ansatz schon heute und auch in

DANKE

Zu diesen in diesem Artikel beschriebenen Ergebnissen haben eine Vielzahl an Kollegen der Audi AG, der AEV GmbH und der EFS GmbH sowie der Firma Syntavision GmbH beigetragen. Besonderer Dank gilt dabei Dr. Bernd Weber, Frank Schimmack, Georg Hofstetter, Eugen Schill, Ingo Houben, Andreas Baudisch, Denny Marx und Nils Seidler.

frühen Entwicklungsphasen neuer Steuergeräte funktioniert. Gleichzeitig offenbart sich, dass die genannten Randbedingungen, insb. die Zykluszeitanpassung, bereits in der Funktionsentwicklung beziehungsweise beim Übergang in die Software-Entwicklung berücksichtigt werden müssen und können. Timing und Safety repräsentieren sogenannte querschnittende Architekturaspekte, die sowohl beim Entwurf als auch im Prozess Berücksichtigung finden müssen und ein Refactoring erfordern. Es ergibt sich ein Wechselspiel der verschiedenen Entwicklungsphasen, die nicht mehr unabhängig abgearbeitet werden können, sondern ein Bewusstsein für das Gesamtsystem erfordern, um die komplexen Anforderungen für die Entwicklung hochintegrierter Steuergeräte meistern zu können.

LITERATURHINWEISE

- [1] Reif, K. et al.: Vernetzte Regelsysteme im Kraftfahrzeug. In ATZ 04/2008
- [2] Liu, C. et al.: Scheduling algorithms for multiprogramming in hard real-time environment. In Journal of ACM (73), Nr. 20, S. 46-61X
- [3] Douglass, B.: Real-Time Agility: The Harmony/ESW Method for Real-Time and Embedded Systems Development. Addison-Wesley Professional, 1 Edition, 2009
- [4] Sha, L.; Lehoczky, J.; Rajkumar, R.: Solutions for Some Practical Problems in Prioritized Preemptive Scheduling. IEEE Real-Time Systems Symposium, 1986
- [5] De Niz, D.; Lakshmanan, K.; Rajkumar, R.: On the Scheduling of Mixed-Criticality Real-Time Task Sets. RTSS 2009. 30th IEEE F T



DOWNLOAD DES BEITRAGS
www.ATZonline.de



READ THE ENGLISH E-MAGAZINE
order your test issue now:
SAM-service@springer.com



Universal-Genie

Ein Werkzeug, das Standards setzt. Die PROtronic TopLINE bietet neue Perspektiven im seriennahen Rapid Control Prototyping:

- Dual Prozessor Architektur mit 1 GHz PowerPC
- Flexible FPGA-Technologie
- Integrierter Datenlogger
- Integrierte und konfigurierbare Signalkonditionierung und Leistungsendstufen
- Durchgehende Toolkette – vom Modell bis hin zur Serie
- Für Motor, Elektro-KFZ, Hybrid, Fahrwerk, Komfortelektronik, Getriebe, Assistenten...

Die Revolution für seriennahes Rapid Control Prototyping



embedded world 2012
Stand 119, Halle 4

AFT Atlas Fahrzeugtechnik GmbH
Gewerbestraße 14 · D-58791 Werdohl
Tel. 0 2392-809 0

www.protronic-info.de