

Adaptive Software-Architekturen für automatisierte Systeme

Fail-Operational: Wie hochautomatisierte Funktionen trotz Fehler funktionieren

Gereon Weiß, Philipp Schleiß
Fraunhofer ESK, Hansastr. 32, 80686 München

Die zunehmende Automatisierung von Systemen erfordert neue Ansätze zur Steigerung deren Verlässlichkeit und Flexibilität. In zukünftig hochautomatisierten Fahrzeugen kann der Fahrer die Kontrolle über das Fahrzeug vollständig abgeben und muss erst nach 10 Sekunden wieder übernehmen können. Hierfür müssen die hochautomatisierten Fahrfunktionen auch im Fehlerfall weiter funktionieren, d.h. fail-operational sein. Der Beitrag stellt ein neues Konzept und Lösung für zukünftige adaptive Fahrzeugsoftware-Architekturen vor. Dies ermöglicht kosteneffizient, die Ausfallsicherheit in eingebetteten, sicherheitskritischen Systemen zu realisieren. Es werden die grundsätzlichen Herausforderungen, neuen Mechanismen und die Integration in die heutige Entwicklung (u.a. mit AUTOSAR) dargestellt. Das Konzept wurde unter anderem in einem E-Fahrzeug implementiert und evaluiert.

Automatisierung – von fail-silent zu fail-operational

Die fortschreitende Automatisierung hin zum Autonomen Fahren erfolgt in aufeinander aufbauenden Stufen mit zunehmendem Automatisierungsgrad. Der nächste Schritt von einem teilautomatisierten zu einem hochautomatisierten System (VDA Stufe 3 [1]) erfordert jedoch eine erhöhte Form der Flexibilität (wie sie auch für neue Mobilitätskonzepte notwendig wird [2]) und Ausfallsicherheit. In diesem Schritt muss der Fahrer das Fahrzeug und das Verkehrsgeschehen nicht ständig selber überwachen, sondern übernimmt wieder die Kontrolle in einem definierten Zeitraum. Wenn eine Situation auftritt, die das automatisierte System nicht selbständig handhaben kann, sollte der Fahrer innerhalb 10 Sekunden wieder die Fahrzeugkontrolle übernehmen. In dem Fall muss jedoch auch garantiert werden, dass das Fahrzeugsystem für einen solchen Zeitraum selbständig funktional bleibt. Fällt beispielsweise ein Steuergerät aus, muss das System dies eigenständig kompensieren können. Ist beispielsweise die Lenkfunktion durch den Ausfall betroffen, so muss diese schnellstmöglich auf andere Weise wiederhergestellt werden, ohne dass das Fahrzeug in einen unsicheren Zustand gerät. Hierfür ist aber ein für den Automotive-Bereich neue Art der Sicherheit notwendig, denn bisher konnten fehlerhafte Systeme häufig dank mechanischer Backup-Lösungen einfach abgeschaltet werden. Nun ist ein Wechsel von diesem sogenannten *fail-silent* zu einem *fail-operational* Verhalten notwendig. Letzteres besagt, dass auch im Fehlerfall die sicherheitskritische Funktionalität aufrecht erhalten wird. Dieses Sicherheitskonzept wird bereits im Avionikbereich mit mehrfacher Redundanz

realisiert [3]. Dies kann jedoch unter anderem aufgrund der hohen Kosten nicht einfach für den Automobilbereich übernommen werden. Aktuelle Ansätze verwenden typischerweise für jede Funktionalität dedizierte Hardware in mehrfach-redundanter Ausführung. Dieser Lösungsansatz mag für eine kleine Anzahl an Fahrfunktionen noch sinnvoll nutzbar sein. Jedoch erscheint er hinsichtlich der vielen ausfallsicheren Fahrfunktionen in zukünftigen Fahrzeugen als zu aufwändig und kostspielig.

Software-basierte Redundanz durch Adaptivität

Um eine effizientere Lösung für fail-operational Verhalten zu erhalten, können aktuelle Möglichkeiten zur Integration mehrerer unabhängiger Funktionen auf einem Steuergerät genutzt werden. Bei einem solchen Konzept sind die hochintegrierten Funktionen nicht mehr an ein dezidiertes Steuergerät gebunden, sondern sie teilen sich Rechenressourcen mit anderen Funktionen auf einer hoch performanten Plattform.

In einem solchen System können mögliche Ausfälle einzelner Steuergeräte global berücksichtigt und kompensiert werden. Hierfür muss das System adaptive auf solche Fehlerereignisse reagieren können. So können beispielsweise verschiedene Fehler mit dem gleichen Backup-Steuergerät im Fehlerfall gehandhabt werden. Nach den bisherigen Ansätzen mit einzelner Hardware-Redundanz müsste jeder Steuergeräteausfall durch mindestens ein spezielles anderes Steuergerät abgesichert werden. Da nicht alle Funktionen sicherheitskritisch sind, das heißt, sie sind zur Einhaltung des Sicherheitsziels nicht notwendig, kann die Menge an Hardware-Redundanz weiter reduziert werden. In den betrachteten Fehlerfällen werden dann nur alle notwendigen Funktionen weiter ausgeführt, indem unkritische Funktionen verworfen werden oder ressourcenärmere Varianten ausgeführt werden. Dieses Konzept wird auch als *Graceful Degradation* bezeichnet. Durch dieses können trotz reduzierten Systemkapazitäten nach dem Ausfall von Komponenten noch sichere Konfigurationen erreicht werden.

Zukünftige E/E-Architekturen automatisierter Fahrzeuge werden zahlreiche hochverfügbare Fahrfunktionen aufweisen und eine Vielzahl von potenziellen Fehlerfällen berücksichtigen müssen. Eine manuelle Konfiguration jedes Steuergeräts für jedes Ausfallszenario ist dabei nicht durchführbar. Unter anderem müssen dabei nämlich diverse Randbedingungen eingehalten werden. Zum Beispiel muss sichergestellt werden, dass jede ausfallsichere Funktion innerhalb eines funktionspezifischen Zeitraums wieder aktiviert wird. Dies liegt typischerweise im Bereich weniger Millisekunden. Analog muss auch gewährleistet sein, dass Funktionen deterministisch aktiviert werden und es nicht zu Doppelaktivierungen einzelner Funktionen nach einer Rekonfiguration kommt. Damit ein System mit solcher Komplexität fehlerfrei entwickelt werden kann, sind Verfahren zur automatisierten Generierung ausfallsicherer Systeme entscheidend. Mit diesen kann einerseits das komplexe Problem der Verteilung und Allokation von

Softwarekomponenten auf Steuergeräte und CPU-Cores für alle Fehlersituationen gelöst werden. Darüber hinaus können gültige Konfiguration für alle an der Rekonfiguration beteiligten Module wie für Scheduling oder Kommunikationsbeziehungen automatisiert erzeugt werden.

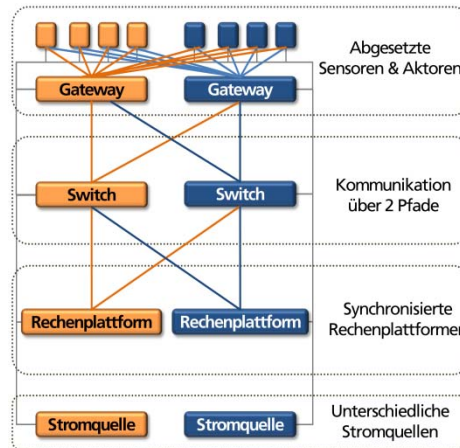


Abb. 1: Anforderungen an die sicherheitskritischen Teile der Hardware-Architektur

Um den skizzierten systemweiten Ansatz zur erhöhten Ausfallsicherheit durch Adaptivität zu ermöglichen [4], bedarf es allgemeinen Anforderungen an die Hardware-Architektur (s. Abb. 1) und einer speziellen Softwarekomponente. Teile der Hardware-Architektur, die für Funktionen mit fail-operational Verhalten genutzt werden soll, muss gegen bestimmte Fehler robust sein. So sind neben abgesetzten sowie redundanten Sensoren und Aktuatoren auch zwei Kommunikationswege zu diesen notwendig. Nur so kann sichergestellt werden, dass ein Linkbruch nicht zum vollständigen Ausfall der Funktionalität führt, da beispielsweise ein Aktuator nicht mehr erreichbar ist. Darüber hinaus ist eine Synchronisierung der beteiligten Steuergeräte unabdingbar, damit die Adaption konsistent und zu gleichen Zeitpunkten erfolgen kann. Jedes beteiligte Steuergerät muss zudem in der Lage sein, eigene Fehler zuverlässig zu erkennen, um sich entweder selbst zu deaktivieren oder alternativ die Auswirkungen des Fehlers auf die jeweiligen Funktionen abzuleiten. Da eine einzelne Stromquelle einen *Single-Point-of-Failure* darstellt, ist es wichtig, diese gesondert abzusichern oder redundant auszulegen. Die spezielle Softwarekomponente baut auf dieser Hardware-Architektur auf [5]. Sie nimmt in verlässlicher Weise und verteilt die Rekonfiguration der einzelnen Steuergeräte vor, indem die jeweils benötigten Funktionen adaptiv auf den vorhandenen Steuergeräten aktiviert werden.

Verteilte Adaption zur Laufzeit

Um nach einem Ausfall alle kritischen Funktionalitäten eines Fahrzeugs betriebsfähig zu halten, ist eine deterministische Adaption aller beteiligten Steuergeräte notwendig. Hierzu wurde ein neues Basissoftwaremodul namens *Safe Adaptation Platform Core* (SAPC) [6] entwickelt, das die Verfügbarkeit aller Softwarekomponenten (SWC) zur Laufzeit verwaltet. Um Informationen über den Systemzustand dezentral zu erfassen, versendet jede SAPC-Instanz synchrone Status-Nachrichten (sogenannte *Health-Vectors*) mit dem Zustand aller verwalteten SWC-Instanzen. Auf dieser Grundlage kann jede SAPC-Instanz unabhängig analysieren, ob sie eine Rekonfiguration durchführen muss. Da die Adaptionen durch eine zentrale Planung im Entwurf definiert werden, sind die Entscheidungen der verteilten Instanzen für sicherheitskritische Funktionen im Verbund konsistent. Dieses Verfahren zum Austausch des Systemzustands ist in Bild 1 abgebildet.

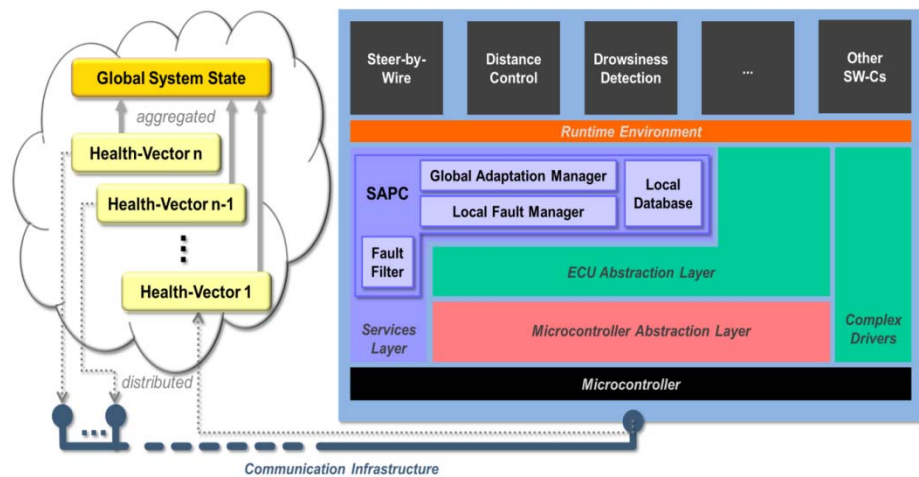


Abb. 2: ECU-übergreifende Überwachung und Rekonfiguration

Werkzeug zur automatisierten Systemsynthese

Um automatisch zu einer fehlertoleranten Systemkonfiguration zu gelangen, wurde aufbauend auf dem AUTOSAR-Austauschformats [6] ein neues Modellierungskonzept entworfen, um auch die Anforderungen bzgl. der Verfügbarkeit einzelner SWC-Instanzen in unterschiedlichen Fehlermodi exakt zu spezifizieren. Basierend auf diesen Informationen analysiert nun das entwickelte Tooling den Daten- und Kontrollfluss zwischen den einzelnen Runnables im Systemmodell und bestimmt einen Schedule für jede ECU und jeden Fehlermodus. Analog dazu werden auch die Übertragungszeitpunkte und die Zusammensetzung von PDUs unter Berücksichtigung der busspezifischen Eigenschaften bestimmt. Der gesamte automatisierte Prozess ist schematisch in Abb. 3 dargestellt.

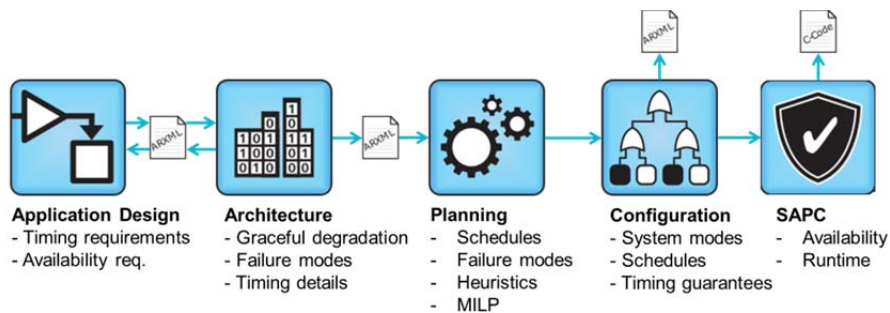


Abb. 3: Workflow zur automatisierten AUTOSAR-Systemsynthese

Zusammenfassung & Ausblick

Um den Anforderungen an die Ausfallsicherheit zukünftiger E/E-Architekturen in einer kosteneffizienten Art gerecht zu werden, sind neue Architektur- und Automatisierungs-Ansätze unerlässlich. Der vorgestellte Ansatz erlaubt es, mit einer sparsamen Hardwarearchitektur und Software-basierter Adaption die hohen Sicherheitsanforderungen automatisierter Fahrzeugsysteme bzgl. der Verfügbarkeit zu erreichen. Damit dies auch in aufkommenden Systemen effizient realisiert werden kann, ist eine Standardisierung der fail-operational Konzepte notwendig. Hierdurch ist es möglich, eine kostengünstige und herstellerübergreifende Interoperabilität der nicht wettbewerbsdifferenzierenden Mechanismen zu etablieren.

Literatur- und Quellenverzeichnis

- [1] Verband der Automobilindustrie (VDA). (letzter Zugriff: 14.10.2016) Automatisiertes Fahren. [Online]. <https://www.vda.de/de/themen/innovation-und-technik/automatisiertes-fahren/automatisiertes-fahren.html>
- [2] Projekt: Adaptive City Mobility 2. (letzter Zugriff: 14.10.2016) [Online]. <http://www.adaptive-city-mobility.de/>
- [3] P. Bieber, E. Noulard, C. Pagetti, T. Planche, and F. Vialard, "Design of Future Reconfigurable IMA Platforms," *Special Issue on the 2nd International Workshop on Adaptive and Reconfigurable Embedded Systems (APRES'09)*, 2009.
- [4] SafeAdapt. (letzter Zugriff: 14.10.2016) Safe Adaptive Software for Fully Electric Vehicles. [Online]. <http://www.safeadapt.eu>
- [5] A. Ruiz, G. Juez, P. Schleiss, and G. Weiss, "A safe generic adaptation mechanism for smart cars," *IEEE 26th International Symposium on Software Reliability Engineering (ISSRE 2015)*, 2015.
- [6] SafeAdapt, "D3.1 Concept for Enforcing Safe Adaptation during Runtime," Project Deliverable, 2015.

[7] AUTOSAR. (letzter Zugriff: 14.10.2016) AUTomotive Open System Architecture. [Online].
<http://www.autosar.org>

Diese Arbeit wurde in Teilen durch das Bundesministerium für Wirtschaft und Energie und der Europäischen Union (Framework Programme 7 – Grant Agreement No.: 608945) Projekt SafeAdapt gefördert.

Autoren



Dr. Gereon Weiss is Group Manager and Deputy Manager of the business unit Automotive at the Fraunhofer Institute for Embedded Systems and Communication Technologies ESK. He is also responsible for the core competencies Dependable Software and Adaptive Systems. Furthermore, Gereon Weiss has been in charge of several national and international research projects as well as industrial projects in the area of networked embedded software systems and is author of numerous publications.



Philipp Schleiß, M.Sc. ist wissenschaftlicher Mitarbeiter bei Fraunhofer ESK in München und dort im Geschäftsfeld Automotive tätig. Sein Themenschwerpunkt liegt in der Entwicklung von effizienten Architekturen für sicherheitskritische Systeme. 2012 schloss er ein Masterstudium der Wirtschaftsinformatik an der Technischen Universität München ab und strebt derzeit eine Promotion im Bereich der Synthese von ausfallsicheren Echtzeitsystemen an der Universität Augsburg an.

Kontakt

Internet: www.safeadapt.eu, www.esk.fraunhofer.de

E-Mail: gereon.weiss@esk.fraunhofer.de

philipp.schleiss@esk.fraunhofer.de