End-to-End-Kommunikationsparadigmen

Einfluss von End-to-End-Kommunikation auf Safety und Security

Karsten Schmidt, AUDI AG

Viele moderne embedded Systeme sind häufig verteilte Systeme, wobei an diese Systeme zusätzliche Anforderungen bezüglich Safety, Security und Echtzeit gestellt werden. Speziell der aktuelle Umstieg auf Ethernet-basierende Kommunikationssysteme bedingt eine kritische Betrachtung der angewandten Kommunikationsparadigmen, um eine effiziente Kommunikation zu ermöglichen.
Im Rahmen dieser Veröffentlichung werden Entwurfskriterien unter Berücksichtigung einer sogenannten end-to-end Betrachtung diskutiert. Es wird gezeigt, warum eine end-to-end Betrachtung von Kommunikationsbeziehungen unter Berücksichtigung von querschneidenden Aspekten, für eine gute System-architektur wichtig ist. Anhand von Beispielen aus dem automotive Bereich werden end-to-end Eigenschaften diskutiert und es wird untersucht, inwieweit diese Eigenschaften massiven Einfluss auf eine System- und Softwarearchitektur haben. Zusätzlich wird das Thema der dabei notwendigen Softwareabstraktion betrachtet.

Einleitung

Viele moderne embedded Systeme sind häufig verteilte Systeme, wobei an diese Systeme häufig zusätzliche Anforderungen bezüglich Safety, Security und Echtzeit gestellt werden. Ein rapide steigender Vernetzungsgrad von Fahrzeugen und der sich abzeichnende Trend zu hoch automatisierten und autonomen Fahrzeugen zeigen die Notwendigkeit, dass querschneidende Aspekte gemeinsam betrachtet werden müssen. Zusätzlich ist es notwendig, die angewendeten Kommunikationsparadigmen kritisch zu betrachten, um eine effiziente Kommunikation zu ermöglichen.
Im Weiteren erfolgt eine kritische Diskussion und der Vergleich gängiger „end-to-end“ Paradigmen sowie die resultierenden Effekte auf die Softwarearchitektur und deren Einfluss auf die technische Umsetzung. Beim Entwurf von verteilten System geht es letztlich um die Frage, wo welche Teile platziert werden und welche Aufgaben diese Teile im Gesamtverbund eines verteilten embedded Systems haben. In einem solchen Kommunikationssystem trennt man gewöhnlich die Kommunikationsinfrastruktur und die eigentliche Anwendung, welche die Kommunikationsinfrastruktur nutzt. Jedoch besteht bei dieser Herangehensweise die Herausforderung, dass bestimmte Teilaufgaben der Kommunikation auf verschiedene Weise gelöst werden können. Im Folgenden soll dieser Punkt näher untersucht werden.

Problemdefinition

Wir gehen von folgendem Szenario (siehe Bild 1) aus. Es existieren zwei Anwendungen, welche bidirektional Daten austauschen. Dazu benutzen die Anwendungen einen Kommunikations-Stack und übertragen die Anwendungsdaten über ein Netzwerk.

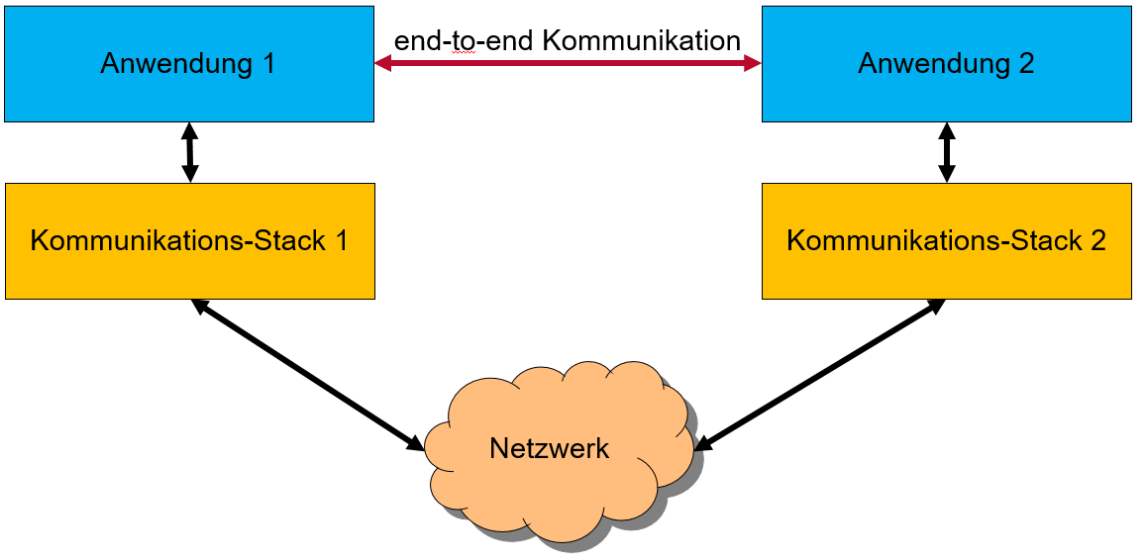

Bild 1 Grundlegende Fragestellung

Die interessante Frage lautet: „Welche Annahmen kann die eigentliche Anwendung über den Kommunikations-Stack und das Netzwerk treffen“. Des Weiteren kann man diskutieren, wie die eigentliche Anwendung Anforderungen an die tieferen Schichten stellen kann. Das betrifft interessanterweise auch Anforderungen an den Kommunikationspartner. Zu den Fragen, die nun für die eigentliche end-to-end Anwendungskommunikation betrachtet werden müssen, gehören unter anderem:

- Sicherstellung eines zeitlichen Determinismus
- Sicherstellung der Datenübertragung unter Safety-Gesichtspunkten
- Sicherstellung der Datenübertragung unter Security-Gesichtspunkten
- Allgemeine QoS-Betrachtungen

Stand der Technik

Diese Betrachtung ist nicht neu. Bereits in [3] wurde dieser Aspekt betrachtet und in [4] anknüpfend diskutiert. Fokus war hier jedoch die klassische IT- Domäne. In [1] und [2] erfolgte eine Anwendung dieser Ideen für die rein zeitliche Betrachtung einer end-to-end Kommunikation innerhalb eines Kommunikationsnetzwerkes eines Fahrzeuges. In [13] erfolgte eine Security-Betrachtung für eine Kommunikationskette Fahrzeug → Backend → Backend. Für PDU-basierte Kommunikation existiert innerhalb des AUTOSAR-Standards [7] die Möglichkeit, die übertragenen Daten gegen zufällige Fehler zu schützen. Neben den rein technischen Themen erfolgte in [10] auch eine Betrachtung der notwendigen Paradigmenwechsel in Bezug auf Timing-Fragestellungen. Auch in den RFC sind Hinweise auf diese Überlegungen zu finden [5] und [6].

Systembetrachtung

Betrachten wir ein typisches vereinfachtes Beispiel aus dem Bereich des hochautomatisierten Fahrens (siehe Bild 2).

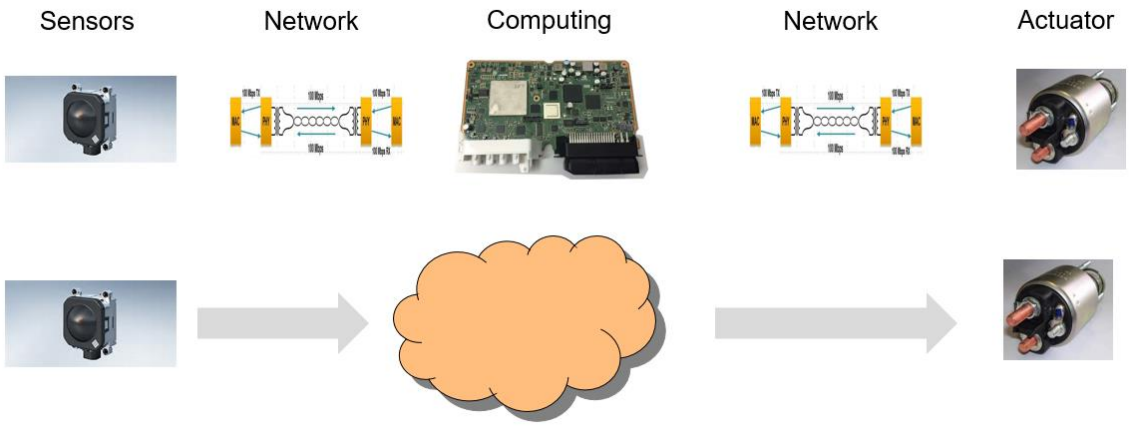

Bild 2 Beispiel aus dem Bereich des hochautomatisierten Fahrens

Für eine Funktion des hochautomatisierten Fahrens wird folgendes Szenario betrachtet. Startend von Sensoren über die in den Steuergeräten zu verarbeitenden Sensordaten bis zu den Aktuatoren werden Daten fusioniert und aggregiert, um komplexe Entscheidungen zu treffen, die letztlich das Fahrverhalten aktiv beeinflussen. Diese Prozesse erzeugen besondere Herausforderungen an eine holistische Systemmodellierung.

Betrachtet man dieses Szenario, ergeben sich einige interessante Fragestellungen. Zunächst ist zu klären, wo die eigentlichen Endpunkte der Kommunikation zu finden sind. Im Sensor scheint es noch relativ einfach zu sein. Aber auch hier existieren unterschiedlichste technische Umsetzungen, die durch die Unterschiede bei der gewählten Kommunikationsanbindung die Systemkomplexität erhöhen.

Endpunkt dieser Kommunikation ist nun das zentrale Steuergerät, welches wiederum Teil einer weiteren Kommunikationsbeziehung ist. Aus reiner Anwendungssicht existiert zusätzlich die logische bzw. funktionale Beziehung Sensor zu Aktuator. So muss z.B. beim Erkennen eines Hindernisses der Aktuator eine entsprechende Reaktion auslösen.

Typische Herausforderungen
Ausgehend von der Systembetrachtung ergeben sich nun einige interessante Frage-stellungen, für die im automotive Kontext Lösungen existieren, die jedoch für andere Industrien in ähnlicher Ausprägung existieren.

Sicherstellung der Datenübertragung unter Safety-Gesichtspunkten
Hier geht es um den Aspekt, dass sich Sender und Empfänger sicher sein können, dass die Daten während der Übertragung nicht zufällig, also durch Störungen, verändert wurden. In der Automobilindustrie gibt es etablierte Standards, um dies sicherzustellen (siehe [7]). Wie bereits in der originalen Veröffentlichung [3] diskutiert, muss beim gesamten Systementwurf und bei der Softwareentwicklung diskutiert werden, wo die

notwendigen Komponenten für die ASIL-D-Funktionalitäten umgesetzt werden. In [11] ist dazu eine Diskussion für ASIL-Steuergeräte zu finden.

Sichere Kommunikation für sensible Daten zwischen ECUs
Security ist ein abstrakter Begriff, der die Widerstandsfähigkeit eines Systems gegenüber vorsätzlichen Angriffen beschreibt. Dementsprechend ist der wichtigste Unterschied zum vorherigen Abschnitt der, dass über Effekte durch böswilligen Eingriff auf die Datenkommunikation gesprochen wird. Dazu gehören:

- Injektion von böswilligen Steuerbefehlen,
- Einfügen, Löschen, Manipulation, Wiederholung und Verzögerung von Nachrichten
- Abhören von sensitiven Informationen

Für Security-Betrachtungen richtet sich ein Angriff auf einen bestimmten Teil des Systems, wie z.B. Schnittstellen, Anwendungen oder Kommunikationskanäle. Typische Angriffsziele im Automobilbereich sind Steuergeräte (Electronic Control Units (ECUs)), Diebstahlschutzsysteme, Vernetzungs- oder Zahlungssysteme. Typische Angriffe versuchen die Entfernung von Sicherheitsabfragen, das Abhören bzw. Modifizieren von Kommunikationsdaten, oder Änderungen an der originalen Firmware zu erreichen [13]. Während des Systementwurfs werden, basierend auf einer Risikoanalyse, die notwendigen Systemkomponenten identifiziert, die notwendig sind, um entsprechende Gegenmaßnahmen umzusetzen. Aber auch hier spielt die Frage der Verortung der notwendigen Softwareteile eine wichtige Rolle.

Quality of Service
Quality of Service [12] beschreibt die Güte eines Dienstes in einem Kommunikations-netzwerk. Innerhalb eines Netzwerkes finden unterschiedliche Kommunikationsabläufe mit unterschiedlichsten Protokollen statt. Für jeden dieser Kommunikationsabläufe müssen verschiedenste Aspekte betrachtet werden. Beispielsweise das Zeitverhalten, das sich in der Latenz, also der Laufzeit der Nachricht, sowie dem Jitter, also der maximalen Änderung dieser Laufzeit, ausdrückt. An den Netzwerkknoten sind Puffergrößen sowie die Priorisierung der Nachricht bzw. des Datenpakets relevant. Innerhalb eines Steuergerätes ergibt sich die notwendige Abschätzung der dazu notwendigen Ressourcen. Dazu gehören die notwendigen Speicher für erforderlichen Buffer für unterschiedlichste Nutzdaten und die sich ergebene Rechenzeit für die Verarbeitung dieser Daten. Erschwerend kommt hinzu, dass die Last, die sich für ein einzelnes Steuergerät ergibt, auch davon abhängt, wie die Gesamtkommunikation innerhalb des Fahrzeuges organisiert ist. Es ist also wichtig, die komplette Kommunikationskette zu untersuchen, da sich z.B. die Latenz über alle Knotenpunkte betrachtet aufsummiert (vergleiche Bild 2).

Softwareentwicklung
Ein weiterer interessanter Aspekt innerhalb dieser Betrachtungen nimmt die Software ein. Das betrifft sowohl die Anwendungssoftware als auch die Software für den Kommunikations-Stack. In Zukunft gewinnt die Fragestellung, wo wird die notwendige

Funktion platziert, die die Security, Safety oder die Quality of Service sicherstellt, immer mehr an Bedeutung.

Die Nutzung von Standardsoftware erscheint in vielen Fällen attraktiv, da man sich mit den oben angesprochenen Fragestellungen nicht auseinandersetzen muss („die Basissoftware wird's schon richten"). Jedoch deckt dieser Ansatz typischerweise nur einen Teil der Systemanforderungen ab [14]. Durch eine geschickte Auswahl der notwendigen Software und deren sinnvolle Aufteilung in den Kommunikations-Stack oder die Anwendungssoftware können effizient Systeme entwickelt werden, die zusätzlich ein hohes Maß an Zuverlässigkeit besitzen.

Erschwert wird die Systementwicklung, Funktionsentwicklung und die eigentliche Softwareentwicklung durch die Aufgabenteilung innerhalb der Organisation. Zusätzlich existiert die Notwendigkeit des notwendigen Know-How-Schutzes bei der Zusammenarbeit unterschiedlicher Firmen. In [10] ist dieser Fakt ausführlich beschrieben. Verstärkt wird dieser Effekt durch die Integration unterschiedlicher Softwareanteile in ein Steuergerät. Bedingt durch die Zusammenarbeit über Firmengrenzen hinweg und der Integration von Software unterschiedlichster Parteien ist es für den eigentlichen Integrator schwer, bei Problemen mit einem Gesamtsystemverständnis an die Fehlersuche zu gehen. Bei Fragestellungen die Security oder Safety-Themen betreffen wird dies noch anspruchsvoller.

Zusammenfassung

Die Komplexität im Entwicklungsprozess hochintegrierter Steuergeräte erfordert eine Berücksichtigung von querschneidenden Entwurfsaspekten. Dazu gehören neben Security und Safety auch Fragen der Echtzeitfähigkeit und des Ressourcenverbrauches. Es ist wichtig zu verstehen, dass dies keine leichte Aufgabe ist, sondern dass dazu vielmehr ein Paradigmenwechsel notwendig ist. Die AUDI AG arbeitet aktuell an der Etablierung eines Security-Prozesses. Dabei werden Erfahrungen aus den Safety-Prozessen entsprechend übernommen und adaptiert.

Einfache Architektur-Standardlösungen existieren nur für sehr wenige Spezialfälle. Wichtig ist die Fähigkeit, komplexe Architekturen zu modellieren und zu analysieren sowie die Kompetenz von Systemarchitekten auch in den neuen Fragestellungen auszubauen.

Literaturverzeichnis

[1] K. Reif, K. Schmidt, F. Gesele, S. Reichelt, M. Saeger, N. Seidler, „Networked control systems in motor vehicles" in ATZelektronik worldwide, 04/2008 Pages 18-23, Springer Fachmedien Wiesbaden GmbH (2008)

[2] K. Schmidt, M. Buhlmann, C. Ficek, K. Richter, „Design Patterns for Highly Integrated ECUs with various ASIL Level", ATZ elektronik worldwide Edition, 2012-01.

[3] J.H. Saltzer, D.P. Reed and D.D. Clark, „END-TO-END ARGUMENTS IN SYSTEM DESIGN" M.I.T. Laboratory for Computer Science

[4] T. Moors, „A critical review of „End-to-end arguments in system design"

[5] RFC 3117, „On the Design of Application Protocols“, https://tools.ietf.org/html/rfc3117
[6] RFC 3439, „Some Internet Architectural Guidelines and Philosophy“, https://tools.ietf.org/html/rfc3439
[7] AUTOSAR 4.3 „Specification of SW-C End-to-End Communication Protection Library“
[8] AUTOSAR 4.3 „Specification of Module Secure Onboard Communication“
[9] K. Schmidt, „Ethernet und IP-Netzwerkstacks im Auto“ ESE-Kongress 2016
[10] K. Schmidt, D. Marx, K. Richter, K. Reif, A. Schulze, T. Flämig, „On Timing Requirements and a Critical Gap between Function Development and ECU Integration“, SAE World Congress, April 2015, Detroit, USA
[11] J. Wolf, P. Müller. „Sicherheit und Leistung durch ASIL-D-AUTOSAR-Basis-software“, Hanser automotive 7-8/2016
[12] F. Netter, F. Reimann, „Quality of Service (QoS) in geswitchten Fahrzeugnetz-werken“, International Congress Electronics in Vehicles, 2015, VDI-Berichte Band 2249 (2015) Seite 561-572
[13] A. Weimerskirch, „Do Vehicles Need Data Security?“, SAE International, December 2011.
[14] C. Jakobs, P. Tröger, „Quo vadis, AUTOSAR?“, INFORMATIK 2017,

Autor

Dr. Karsten Schmidt studierte Elektrotechnik an der TU Dresden und arbeitet als Entwicklungsingenieur bei der AUDI AG. Er ist verantwortlich für neue Security-Architekturkonzepte und den Einfluss der Ethernet-Vernetzung auf zukünftige Steuergerätegenerationen.

Kontakt

E-Mail: karsten.schmidt@audi.de