Smart Mobility

Schlanker Hypervisor für unterwegs

11.10.2018 Michael Hauser, Dr. James Dickie und Dr. Nigel Tracey

Agile Software-Entwicklung bedingt bei sicherheitskritischen Funktionen eine strikte Trennung der Softwaremodule. Im gegenläufigen Hardwaretrend trägt ein zentrales Steuergerät immer mehr Funktionen. Die Auflösung dieses Widerspruchs mit einem Hypervisor muss spezielle Anforderungen berücksichtigen.

Im Automotive-Bereich sind softwaregesteuerte Features, wie Tempomat oder Parkhilfen, heutzutage Standard. Nachträgliche Funktionsupgrades und zukünftig fortlaufende (Over-the-Air-)Updates einer Funktion dürfen keinesfalls Auswirkungen auf andere Softwaremodule haben. Doch wie lässt sich das gewährleisten, wenn der Trend gleichzeitig zur Konzentration von immer mehr miteinander vernetzten Funktionen auf wenigen, zentralen Steuergeräten geht? Ist es in einem solchen Umfeld überhaupt möglich, vorab mit Tests die funktionale Sicherheit des Gesamtsystems nach den Upgrades und Updates zu validieren und zu verifizieren?

Schon diese beiden Fragestellungen zeigen, dass es gelingen muss, Softwarefunktionen sicher voneinander zu entkoppeln.

Praktikable Partitionierung

Nicht nur unter dem Aspekt der funktionalen Sicherheit ist diese Entkopplung gefragt. Sie vereinfacht auch die Workflows in der Entwicklung, wenn Software unterschiedlicher Hersteller auf einem einzigen Steuergerät betrieben wird. Zudem sorgt die Trennung dafür, dass Steuergeräte in Zeiten zunehmender Cyberkriminalität schwieriger angreifbar sind. Hat sich ein Hacker Zugang zu einer Funktion verschafft, stellt der Hypervisor eine weitere Hürde dar.

Um die Entkopplung zu realisieren, sind verschiedene Ansätze denkbar. So könnte man Softwarefunktionen strikt auf jeweils eigene Steuerungshardware verteilen. Doch sowohl die Hardwarekosten als auch die Systemkomplexität sprechen dagegen.

Handhabbarer ist dagegen eine auf Autosar basierende Architektur mit definierten Partitionierungs- und Separierungskonzepten. Auf dieser Basis ist es möglich, Upgrades einzelner Funktionen zu realisieren und dabei die Beeinträchtigung weiterer Funktionen auszuschließen. So ist gewährleistet, dass die Modifikation einer Funktion keine umfassende Neuvalidierung sämtlicher Software auf dem betreffenden Steuergerät erfordert. Um die Autosar-Konzepte umzusetzen, sind jedoch Erweiterungen notwendig.

Fortsetzung mit dem Hypervisor

Eine gangbare Erweiterung ist der Einsatz eines Hypervisors. Er partitioniert ein einzelnes Steuergerät in diverse virtuelle Maschinen (VM). Obwohl die Funktionen faktisch auf demselben Steuergerät laufen, wähnt sich die jeweilige Software in einem Zustand, in dem es für jede Funktion eine eigene Hardware gibt.

Die Funktionen sind so strikt entkoppelt, dass sie ohne komplette Neuvalidierung einzeln modifiziert werden können – und ihre verschiedenen Hersteller schon in der Entwicklung des Steuergeräts unabhängig von allen anderen arbeiten können.

Softwarefehler oder böswillige Eindringlinge bleiben lokal auf eine einzelne virtuelle Maschine begrenzt. Darüber hinaus kann Software mit verschiedenen ASIL-Sicherheitseinstufungen (Automotive Safety Integrity Level) von der niedrigsten Stufe QM bis zur höchsten Anforderung ASIL D auf einem einzigen Steuergerät laufen. Bei allen Vorteilen kommt es bei einer Hypervisor-Lösung jedoch auf die Umsetzung an. Ohne Anpassung an die spezifische Fahrzeugumgebung droht Ungemach. So benötigt ein Hypervisor üblicherweise ein eigenes Speichermanagement sowie einen sogenannten Hypervisor-Privilege-Modus, der die Zugriffsberechtigungen regelt.

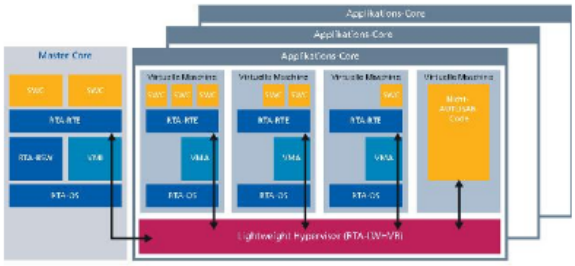
Dieser ist bei klassischen Ausführungen dreistufig: der Hypervisor selbst, die Basissoftware und die Applikationsfunktionen. Doch sowohl das entsprechende Speichermanagement als auch der dreistufige Privilege-Modus werden von generischen Fahrzeug-Mikrocontrollern nicht unterstützt. Dies stand dem breiten Einsatz der Hypervisor-Technologie im Automobil bisher entgegen.

Zur Entschärfung der genannten Problemfelder hat Bosch Automotive Electronics – Body Electronics (AE-BE) in einem OEM-Projekt den Etas Lightweight Hypervisor (RTA-LWHVR) verwendet. Der optimierte Automotive-Hypervisor hat nur noch einen Speicherbedarf von 5 Kilobyte (kB) und die Zugriffszeiten konnten in unterschiedlichen Anwendungsfällen um Faktor vier bis fünf verbessert werden. Bei diesem Hypervisor ist eine Beeinflussungen zwischen den virtuellen Maschinen ausgeschlossen. Im konkreten Projekt war ein zentrales Body-Steuergerät in elf virtuelle Maschinen partitioniert, die jeweils für Software von unterschiedlichen Zulieferern reserviert waren. Die ASIL-Einstufung reichte dabei von QM bis B.

Über Autosar hinaus

Trotz der Menge und Heterogenität der Softwarefunktionen verlief die gekapselte Ausführung der Softwarefunktionen auf dem Lightweight Hypervisor ohne Komplikation. Dies gelang, weil die virtuellen Maschinen zwar auf den gemeinsamen Speicher zugreifen, die Zugriffe und die Laufzeiten auf dem Kern aber klar geregelt sind. Die hohe Performanz resultiert aus dem Aufteilen der Rechnerkerne in einen Master-Core und die Applikations-Cores.

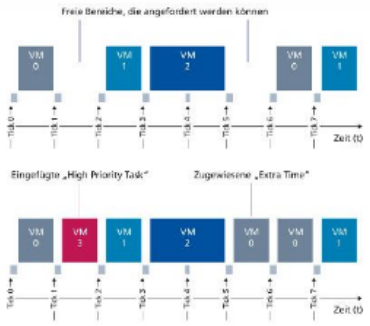
Während dem Master-Core das Hardwaremanagement sowie der Betrieb der zentralisierten Basissoftware und einiger Software-Applikationen zukommt, beherbergen die Applikations-Cores die strikt getrennten virtuellen Maschinen (Bild 1) – wahlweise mit aufgeteilten Laufzeitumgebungen (RTE) gemäß Autosar oder auch mit nicht-Autosar-konformer Software. Bei diesem Ansatz ist die entsprechende Inter-Core-Kommunikation (ICC) wichtig. Sie wurde von Bosch AE-BE entwickelt. Funktionen können zusätzlich zur ohnehin garantierten Ausführungszeit weitere Zeitbudgets anfordern, ohne dass es zu Abstrichen bei der Ausführung anderer Funktionen kommt (Bild 2).



© Etas

Bild 1: Schematischer Aufbau eines Steuergeräts mit dem Lightweight Hypervisor RTA-LWHVR.

[Glossar: RTA-BSW – AUTOSAR-Basissoftware; RTA-OS – Betriebssystem; RTA-RTE – Laufzeitumgebung; SWC – Softwarekomponente in gesicherter Anwendung; VMA – Adapter zur virtuellen Maschine; VM – Virtuelle Maschine, VMI – Virtuelle Maschinenschnittstelle; ICC – Inter-Core-Kommunikation.]



© Etas

Bild 2: Zeitmanagement in einem Applikations-Core.

Die Echtzeitanforderungen sind also jederzeit gewährleistet. Bei hohem Laufzeitbedarf, welcher die eigene Kapazität zu überlasten droht, können die virtuellen Maschinen beim Hypervisor anfragen, ob sie vorübergehend auf zusätzliche vorgehaltene Laufzeitfenster Zugriff erhalten. In diesem Fall merkt der Hypervisor die virtuelle Maschine in einer Warteschleife vor. Ist ein Laufzeitfenster frei, erlaubt er der ersten virtuellen Maschine in der Schleife, dieses zu nutzen. Mit diesem Vorgehen werden die Störungen durch eine hohe Systemlast minimiert. Jede virtuelle Maschine kann jedoch nur die Reservierung eines Fensters anfordern. So wird verhindert, dass eine fehlerhafte oder gehackte virtuelle Maschine die Kontrolle über das System erlangt.

Features

Mit gegenüber klassischen Hypervisors drastisch reduzierten Overheads mit 5 kB Speicherbedarf und einem auf 5 Prozent der verfügbaren Kernkapazität reduzierten Leistungsbedarf, adressiert der RTA-LWHVR die spezifischen Randbedingungen für Automotive Embedded Systeme. Der Hypervisor kann an die jeweiligen Mikrocontroller angepasst werden. Die ersten Projekte erfolgten mit Mikrocontrollern von STMicroelectronics, Infineon TriCore oder NXP Semiconductors (Freescale).

Dabei gewährleistet er eine ebenso zuverlässige wie leistungsfähige Partitionierung von Steuergeräten, auf denen damit künftig Software unterschiedlicher Hersteller und Sicherheitsklassen betrieben werden können. Mit selbstadaptierender Inter-Core-Kommunikation und strikter Kapselung können softwaregesteuerte Funktionen unabhängig voneinander entwickelt und – auch bei Fahrzeugen in Kundenhand – jederzeit ohne Neuvalidierung des Gesamtsystems modifiziert werden.

Damit legt der Lightweight Hypervisor eine abgesicherte Basis für agile Software- und Funktionsentwicklung in der Automobilindustrie, die auch dynamische Security-Systeme mit regelmäßigen bedarfsgerechten Sicherheitsupdates ermöglicht.

Autoren

Michael Hauser, Teamleiter Software-Entwicklung bei Bosch Automotive Electronics
Dr. James Dickie, Produktmanager RTA-Solutions bei Etas
Dr. Nigel Tracey, General Manager bei Etas

Quelle: https://www.elektroniknet.de/design-elektronik/embedded/schlanker-hypervisor-fuer-unterwegs-157681.html