



Die Zukunft simulieren

Simulationsframework für die Mobilität von morgen

In nicht allzu ferner Zukunft werden wir uns in fahrerlosen Fahrzeugen sicher und effizient über die Straßen bewegen. Bereits heute arbeiten zahlreiche Ingenieure daran, dass diese Vision Wirklichkeit wird. Ein wichtiger Erfolgsfaktor auf dem Weg dorthin ist die Simulation im virtuellen Umfeld. Nur durch frühzeitiges Testen und Validieren lässt sich die Komplexität beherrschen. Beim globalen Simulationsframework IDCSim, das Bosch für Daimler entwickelt, spielt ISOLAR-EVE von ETAS eine wichtige Rolle.

Unter uns: Die Zukunft können wir natürlich nicht vorhersagen. Doch mit unserem Simulationsframework arbeiten wir an der Zukunft des vollautomatisierten Fahrens. Wir, das sind weltweit über hundert Entwicklungsingenieure bei Daimler, das Plattform-Entwicklungsteam bei Bosch und die Entwicklungsmannschaft von ETAS. Unser Ziel: Mit einem durchgängigen Simulationsframework entwickeln wir Steuergerätesoftware zukünftiger Systeme für automatisiertes Fahren. Doch der Reihe nach.

Durch die Komplexität der Aufgabe sind wir gezwungen, schon an der Seriensoftware zu arbeiten, während noch über die Hardware nachgedacht wird. Anders kann das weltweite Entwicklungsteam den ambitionierten Terminrahmen nicht einhalten.

Kontinuierlich arbeiten wir an unseren Konzepten und Funktionen, um diese in das virtuelle Simulationsframework zu integrieren. Im Gegensatz zu den relativ statischen Hardware-in-the-Loop-(HiL-)Lösungen verändert sich beim hier angewandten Software-in-the-Loop-(SiL-)Ansatz das System laufend.

Doch auch beim SiL-Ansatz darf das zukünftige Seriensteuergerät nicht außer Acht gelassen werden. Es basiert auf der Plattform Bosch DASy (Driver Assistance System Domain Controller), die den AUTOSAR-Standard verwendet. Schon heute implementiert der Mikrocontroller die AUTOSAR-Classic-Plattform. Künftig wird der Mikroprozessor auf der AUTOSAR-Adaptive-Plattform aufbauen.

Die Herausforderung

Die funktionale Sicherheit muss in unzähligen Fahrmanövern nachgewiesen werden, von denen viele in realen Testfahrten nur schwer reproduzierbar oder sogar gefährlich sind. Virtuelle Testfahrten sind dabei eine Schlüsseltechnologie für hochautomatisiertes Fahren. Nur so ist es möglich, die erforderliche Testabdeckung für die Absicherung der Software effizient zu erreichen. Testfahrten im Fahrzeug werden nur noch an ausgewählten Fahrmanövern durchgeführt; hauptsächlich, um die nötigen Testdaten zu gewinnen und die Gültigkeit der Simulation immer wieder zu überprüfen.

Bei der Entwicklung des Simulationsframeworks betreten wir in vielen Bereichen Neuland. Treffen doch die Welten der Embedded-Steuergeräte und der IT aufeinander. Während bei den einen das Echtzeitverhalten und die funktionale Sicherheit nach ISO 26262 eine wichtige Rolle spielen, sind bei der IT hohe Datenraten, Connectivity und Cyber Security von hoher Bedeutung. All dies muss eingehalten werden.

Mit unserem Simulationsframework für komplexe Software zukünftiger Fahrzeugsysteme können Entwickler weltweit schnell und sicher auf umfangreiche Daten zugreifen und ihre Funktionen validieren.

Ziel der Entwicklung ist es auch, das Framework möglichst universell einzusetzen und die Skalierung vom einzelnen Entwicklungsarbeitsplatz bis zum massiv parallelen Test auf Serverclustern oder in der Cloud sicherzustellen. Darüber hinaus soll es sich ebenso gut für den Test einer einzelnen neuen Softwarefunktion eignen wie für die Simulation eines komplexen Steuergerätenetzwerks, das aus Sensoren und Domänen-ECUs besteht. Daher wurde von Anfang an auf eine flexible und skalierbare Architektur des Frameworks geachtet.

Neben den hohen technischen Herausforderungen muss das Simulationsframework außerdem sehr robust gegen Fehlfunktionen sein. Schließlich arbeiten weltweit über hundert Entwickler damit. Da darf es nicht zu Ausfällen kommen.

Arbeitsweise

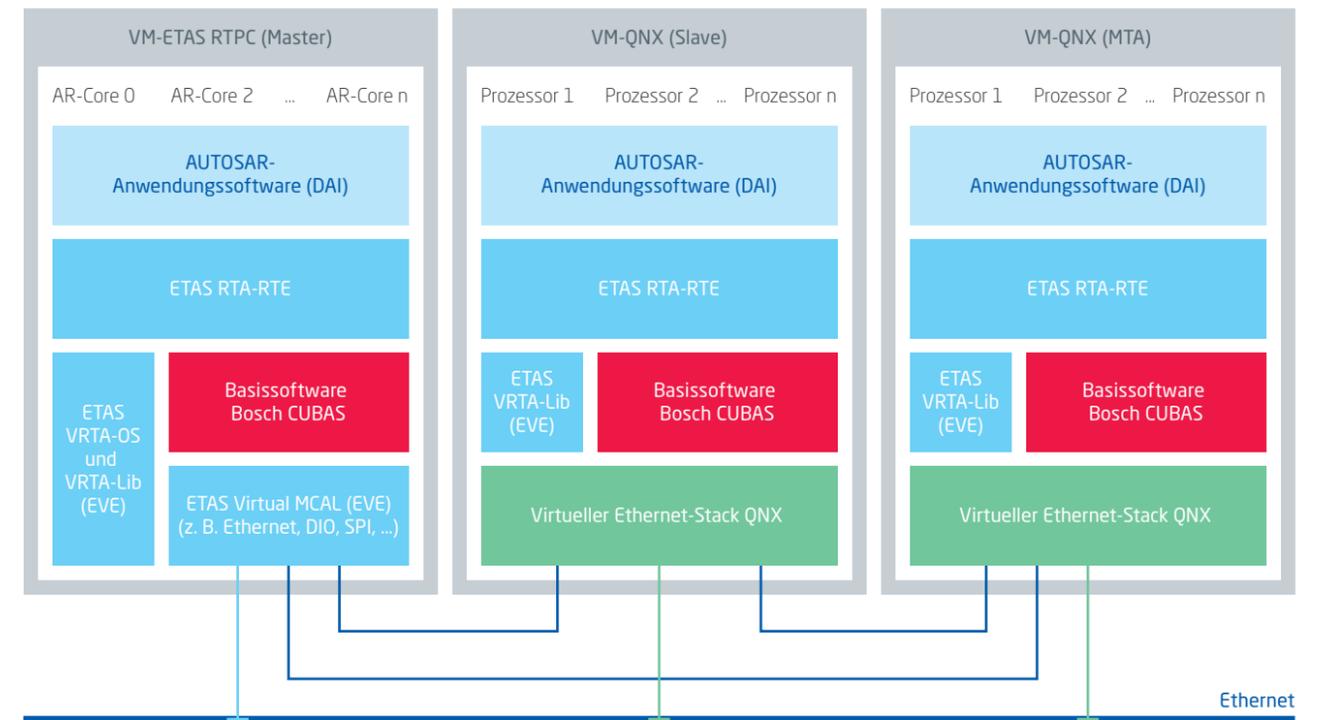
Das Simulationsframework virtualisiert die Fahrzeugumgebung und simuliert das ADAS-Steuergerät sowie die Sensoren, wie Ultraschall, Radar, Kamera und viele andere. Es berechnet die Entscheidungen des Systems und die entsprechende Reaktion der Aktoren. So kann die Anwendungssoftware mit dem Streckenmodell simuliert werden – und umgekehrt. Mit der Benutzerschnittstelle steuert der Anwender die Simulation. Sie wurde als Engineering-Dienstleistung von ETAS entwickelt.

Durch die Virtualisierung steht jedem Entwickler direkt am Arbeitsplatz eine Validierungsumgebung zur Verfügung. Sie basiert immer auf den jeweils aktuellen Versionen aller Softwarekomponenten und Umgebungsmodelle. Eine schnelle und einfache „up-to-date“-Funktionsvalidierung einschließlich Debugging, Code-Coverage-Analyse und der Erzeugung von Logdateien ist so möglich.

Die technische Umsetzung

Ein wichtiges Element zur Beherrschung der hohen Systemkomplexität war die Vorgehensweise nach dem Baukasten-Prinzip, um einzelne Bereiche zu kapseln. Dabei war der Weg nicht immer gleich von vornherein klar. Unser Ziel war es, mit frühen Prototypen zu lernen und, wenn nötig, früh zu korrigieren, um dann den richtigen Weg zu finden. Inzwischen haben wir einen hohen Reifegrad erreicht und befinden uns in der Optimierungsphase.

Die bei Bosch entwickelte Simulationsumgebung basiert unter anderem auf ETAS ISOLAR-EVE, mit dem die virtuellen Steuergeräte erzeugt werden, und der ETAS-RTPC-Software, die durch ihre Linux-Basis als effiziente Ausführungsplattform für die virtuellen Steuergeräte dient. Schlüssel zum Erfolg ist ein hohes Verständnis der realen Steuergeräte. Nur so ist eine realitätsnahe Simulation möglich.



Aufbau der Simulationsumgebung für Mikrocontroller und -prozessoren

Die ISOLAR-EVE-Technologie bildet eine wichtige Grundlage für das virtuelle Steuergerät, indem sie für die Virtualisierung des Mikrocontrollers das Betriebssystem (VRTA-OS für Windows und Linux), den Mikrocontroller Abstraction Layer (MCAL) sowie Teile des Build-Prozesses bereitstellt. Sie sorgt auch für die nahtlose Integration der AUTOSAR Runtime Environment (RTA-RTE). Für die Virtualisierung der Mikroprozessoren wird ebenfalls das VRTA-OS verwendet, gemeinsam mit VMware und QNX.

Die Vorteile

Jeder Entwickler hat nun eine leistungsfähige Simulationsplattform an seinem Arbeitsplatz. Die Verfügbarkeit von Hardware ist somit kein Engpass mehr. Die Testiterationen werden kürzer, die Testabdeckung wird höher. So erreicht die Software schneller einen hohen Qualitätsstand und spätere Tests am HiL und im Fahrzeug werden reduziert. Auch kritische Fahrsituationen können sicher am Schreibtisch betrachtet, detailliert analysiert und beliebig oft reproduziert werden.

Doch damit nicht genug. Die Verwendung virtueller Steuergeräte eröffnet auch Möglichkeiten, die eine hardwarebasierte Validierungsplattform nicht bieten kann: Künstliche Schnittstellen, Playback von kritischen Situationen, Zeitraffer oder gedehnte Zeit machen es den Ingenieuren leichter, Fehler aufzuspüren oder das Gesamtsystem zu verstehen.

Fazit

Mit unserem Simulationsframework für komplexe Software zukünftiger Fahrzeugsysteme mit Mikrocontrollern und -prozessoren können Entwickler schnell und sicher auf umfangreiche Daten zugreifen und ihre Funktionen validieren. Derart umfassende Projekte erfordern es, dass alle Experten eng als Partner zusammenarbeiten – sei es bei der Spezifikation, Umsetzung oder Tool-Qualifizierung. Diese Zusammenarbeit ist uns gelungen – und gerade auch wegen unserer Unterschiedlichkeit. Dabei sind wir noch lange nicht am Ziel, denn mit dem Kenntnisstand wachsen die Anforderungen – auch in Richtung Künstliche Intelligenz. Das spornt uns an. Wir vom Entwicklungsteam tragen stolz dazu bei, dass wir in Zukunft sicher automatisiert vorankommen.

Autoren

Christoph Baumann ist Projektleiter bei der Daimler AG in Stuttgart. **Johannes Dinner** ist Kunden-Projektleiter bei der Robert Bosch GmbH in Abstatt. **Dr. Roland Samlaus** ist Teilprojektleiter für die Plattformentwicklung Simulationsframework bei der Robert Bosch GmbH in Stuttgart. **Ricardo Alberti de Souza** ist Consultant für ECU-Virtualisierung, Adaptive AUTOSAR und SW-Architektur bei der ETAS GmbH.