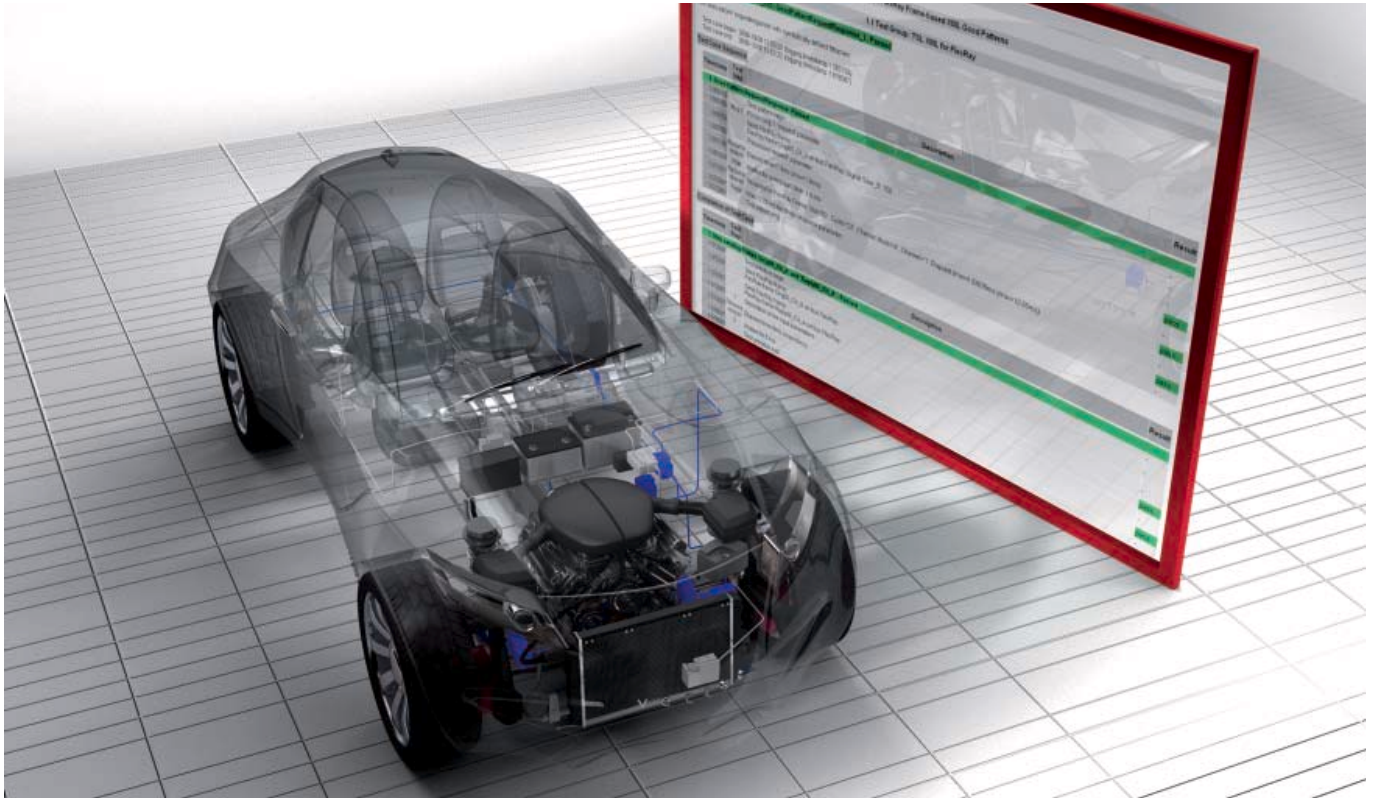


# Über FlexRay hinaus

## Anforderungen an eine moderne Entwicklungsumgebung



**Der Kommunikationsbus FlexRay und das Datenbank-Austauschformat FIBEX repräsentieren die aktuelle Technik im Bereich der Fahrzeugvernetzung. Während des gesamten Entwicklungsprozesses muss der FlexRay-Bus Anforderungen aus den Bereichen der Restbussimulation, Diagnose und weiteren höheren Protokollen, des Tests und der AUTOSAR-Entwicklungsmethodik erfüllen. Diese Anforderungen werden schnell, umfangreich und teilweise komplett neu in den FlexRay-Entwicklungswerkzeugen umgesetzt. In diesem Beitrag werden die Anforderungen an eine effektive Entwicklungsplattform dargestellt. Dabei wird auf die speziellen Anforderungen des FlexRay-Busses eingegangen.**

### Schnelle und verlässliche Simulation von Steuergeräten

Im Entwicklungsprozess eines Steuergeräts ist es essentiell, dieses auch ohne die Integration in das Gesamtsystem in Betrieb nehmen zu können. Die weiteren Steuergeräte des FlexRay-Netzwerks müssen deshalb als Kommunikationspartner simuliert werden. Man spricht dabei von einer Restbussimulation. Die strikten zeitlichen Anforderungen sind bei FlexRay nur schwierig einzuhalten: oft werden Simulationen am FlexRay-Bus nicht-deterministisch ausgeführt. Dies führt durch die so genannten Slot-Misses zu Über- und Unterabtastungen der Daten auf dem Bus. Deshalb wird oft dringend empfohlen, dass diese Simulationen auf einer dedizierten, Jitter-freien und vorhersagbaren (echtzeitfähigen) Plattform ausgeführt werden. Hiefür stehen unter anderem teure HIL-Systeme oder umständlich handhabbare Rapid-Prototyping-Plattformen zur Verfügung. Gerade für den konkreten Anwen-

dungsfall einer Restbussimulation für Tests gibt es eine effektive und kostengünstige Lösung: Die Ausführung der Simulation auf den Hardware-Interfaces der entsprechenden Bussimulations- oder Analyse-Werkzeuge. So erstellt der Anwender mit CANoe RT (Bild 1), z.B. auf dem High-End Interface VN89xx (Q3/2010), eine Echtzeit-Simulation des Gesamtsystems oder er nutzt ein FlexRay-Interface von Vector zum Ausführen der Simulation für ein Steuergerät. Diese Lösungen sind nahtlos in das Simulations- und Testwerkzeug CANoe integriert. Damit kann der Entwickler immer die gleiche Plattform und den gleichen Code verwenden, egal ob er eine reine Simulation oder eine Restbussimulation einsetzt, oder ob er reale Steuergeräte am Bus testet.

Zudem bieten ihm die Werkzeuge und Plattformen zusätzliche Funktionen einer Restbussimulation an. Beispielsweise wird das Sendeverhalten der Steuergeräte abhängig von globalen System-

zuständen (z.B. Zündung ein/aus) automatisch angepasst (Interaction Layer). Das Inkrementieren von Alive-Countern und das Berechnen von zusätzlichen Prüfsummen werden bereits autonom durchgeführt. Der Entwickler kann sich ganz auf die Entwicklung oder den Test der Applikation seines Steuergeräts konzentrieren. Natürlich ist es für Tests auch möglich, gezielt Fehler auf der Kommunikationsebene zu injizieren.

**Protokolle auf höheren Schichten**

Steuergeräte tauschen nicht nur Signale aus, wie z.B. Geschwindigkeit oder Öltemperatur, sondern kommunizieren auch über Protokolle auf höheren Schichten des OSI-Modells der ISO, wie z.B. Transportprotokolle, Netzwerkmanagementprotokolle, beim Flashen oder Kalibrieren. Während des normalen Betriebs des Steuergeräts sind solche Protokolle notwendig, um die Buskommunikation zu koordinieren. In den Service-Phasen werden die Protokolle genutzt, um die Steuergeräte auszuwerten oder zu aktualisieren. Die Steuergeräte eines bestimmten OEMs enthalten jeweils eine konkrete Ausprägung z.B. eines Transport-Protokolls inklusive OEM-spezifischer Anpassungen.

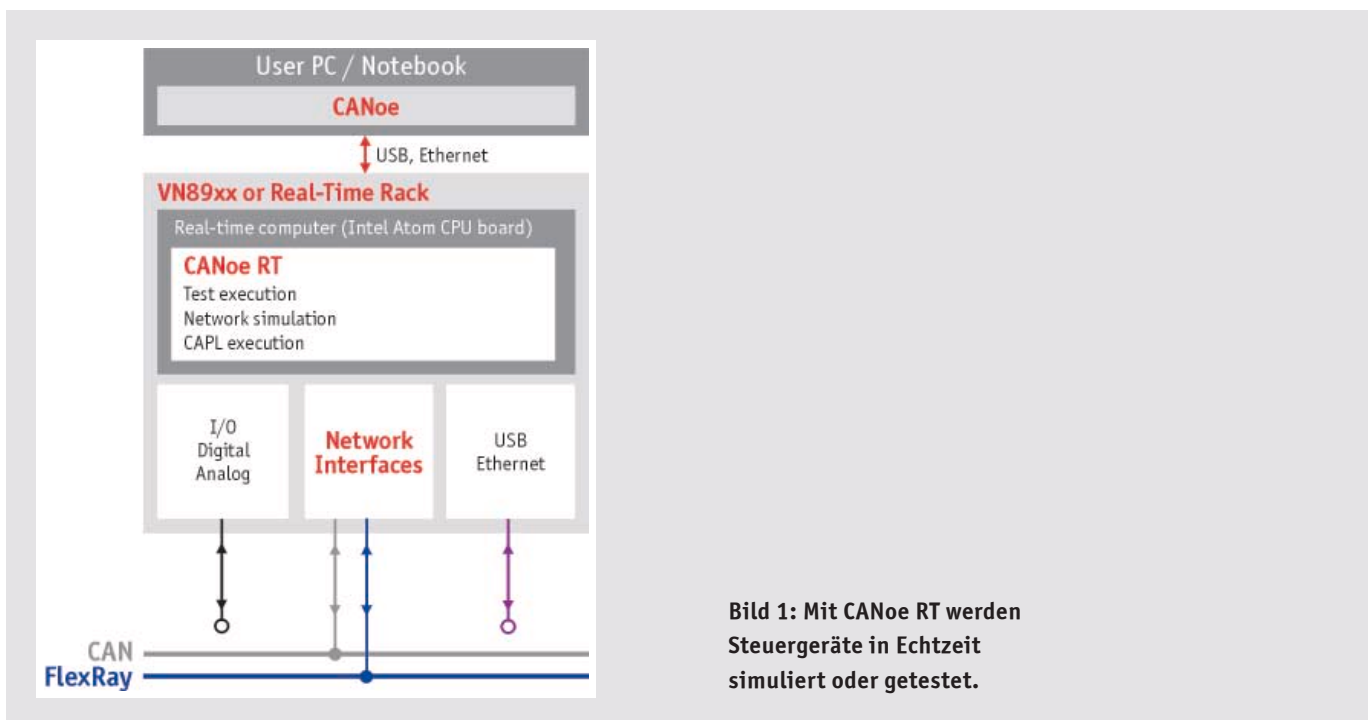
Moderne Entwicklungswerkzeuge müssen in der Lage sein, nicht nur die Basis-Daten auf dem FlexRay-Bus anhand der FIBEX-Beschreibung zu analysieren oder darzustellen, sondern auch die genannten Protokolle auszuwerten und Informationen darüber zu verschicken. Diese Funktionalitäten werden in Form von Add-ons oder Plug-ins bereitgestellt (Bild 2).

Ein modernes Modul-Konzept der Entwicklungswerkzeuge erlaubt das Wiederverwenden der Plug-ins in unterschiedlichen Werkzeugen und das nahtlose Integrieren in das Werkzeug selbst. Im Idealfall integriert sich die Erweiterung so, als wäre sie primärer Bestandteil des Werkzeugs. Die Protokoll-Erweiterungen stellen spezielle Dialoge bereit, um die Parameter der Protokolle einzustellen und abzufragen. Über interne Schnittstellen können die Funktionen der Plug-ins in eigenen Programmen oder Skripten verwendet werden. Damit kann der Entwickler programmgesteuert die Kommunikation beeinflussen oder darauf zugreifen.

**Testen mittels Diagnose**

Der Zugriff auf die Diagnose-Funktionalität eines Steuergeräts ist gerade in den frühen Phasen des Entwicklungsprozesses, z.B. zum Auslesen des Fehlerspeichers oder Flashen, sehr wichtig. Da die Diagnose normalerweise über einen CAN-Bus auf ein Steuergerät zugreift, wird für ein FlexRay-Steuergerät ein CAN-FlexRay-Gateway benötigt. Gerade in den frühen Phasen ist dieses aber meistens noch nicht verfügbar. Deshalb unterstützt das „Diagnostic Feature Set“ von CANoe und das Mess- und Kalibrierwerkzeug CANape den direkten Zugriff auf die Diagnosedaten über den FlexRay-Bus (Bild 3). Dazu werden die speziellen Transportprotokolle für FlexRay unterstützt (AUTOSAR, ISO und OEM-spezifische Varianten).

Nach dem Laden der Diagnose-Beschreibungsdatenbanken im CANdela oder ODX-Format stehen in CANoe und CANape Dialoge und Funktionen für die Diagnose bereit. Damit wird z.B. der Fehlerspeicher inklusive der symbolischen Interpretation seiner Fehler-



**Bild 1: Mit CANoe RT werden Steuergeräte in Echtzeit simuliert oder getestet.**

Codes ausgelesen und dargestellt. Zusätzlich werden alle Diagnose-Anfragen inklusive ihrer Antworten und Parameter mit Hilfe der Diagnose-Konsole symbolisch verschickt und ausgewertet. Die Diagnose-Kommunikation wird in ihrer domänen-spezifischen Notation in den Traces dargestellt. Ebenso steht ein eigenes Diagnose-API zur Verfügung. Damit ist ein einfacher Zugriff bei der Programmierung, dem Test und der Werkzeug-Steuerung möglich.

### Steuergerätestests einfach erstellen

Häufig eingesetzte Standardtests verlaufen oftmals nach dem gleichen Schema. Deshalb sollte ein Testwerkzeug oft wiederkehrende Testmuster unterstützen. Für CANoe werden sequentielle Tests und Testschritte mit Hilfe des „Test Automation Editors“ sehr einfach erstellt. Dazu werden vorgefertigte oder benutzer-spezifische Bibliotheken eingebunden. Diese erlauben dem Testingenieur, die Testfälle einfach auszuwählen und zu parametrieren. Optional können komplizierte oder aufwendige Tests algorithmisch spezifiziert werden. Damit ist es auch möglich, Schleifen oder Verzweigungen anhand von Ergebnissen der vorhergehenden Tests durchzuführen. Die statischen und algorithmischen Tests sind kombinierbar.

Zusätzlich zu den stimulierenden Tests kann gleichzeitig die Buskommunikation anhand des FlexRay-Protokolls überwacht werden. Dazu zählen u.a. das Detektieren von Null-Frames und Erroneous-Frames sowie das Überwachen der Frame-Periode, des Antwortzeitverhaltens, u.s.w. Die Überwachungen erfolgen mit so genannten observierenden Checks, die bereits in CANoe enthalten sind und nur noch aktiviert und parametriert werden müssen.

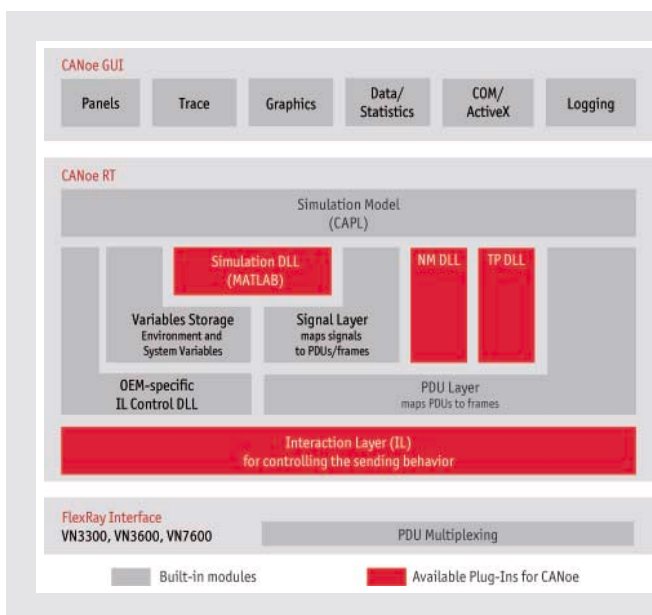
Für Hardware-In-the-Loop- oder Umgebungssimulationen ist häufig das Anschließen des Steuergeräts an seine reale Sensorik und Aktorik nötig. Die digitalen und analogen Ein- und Aus-

gangsgrößen müssen dabei vorgegeben oder eingelesen werden. Hierfür und für „Open-Loop“-Simulationen und -Tests wurde das „VT System“ von Vector entwickelt. Es stellt automobil-taugliche I/O-Ports zur Verfügung, die auf einfachste Weise in den Test eingebunden werden. Das System ist dabei gegenüber allgemeinen digitalen oder analogen I/O-Karten in der Lage, wahlweise die Original>Last oder den originalen Sensor oder eine entsprechende Simulation des Ein- oder Ausgangs bereitzustellen. Gleichzeitig können über entsprechende Signalkonditionierungen große Spannungsbereiche und Ströme abgedeckt werden. Somit kann auch ein Fehlverhalten eines Sensors oder Aktors oder ein Kurzschluss simuliert werden.

### Testen von AUTOSAR-Funktionalität in FlexRay-Systemen

In der Design-Methodik von AUTOSAR ist die Kommunikation als Basisdienst für die Applikation transparent. Im Rahmen des Gesamtsystems ist die Kommunikation in der „AUTOSAR System Description“ definiert. Soll ein AUTOSAR-Steuergerät einfach und schnell getestet werden, muss dem Testwerkzeug die Kommunikationsbeschreibung des Steuergeräts bekannt sein. Es ist benutzerfreundlicher, direkt die „AUTOSAR System Description“ anstatt FIBEX-Datenbanken einzulesen, wie es CANoe.FlexRay unterstützen wird.

Im Kontext von AUTOSAR sind „Software Components“ (SWCs), „Runtime Environment“ (RTE) und „Virtual Function Bus“ (VFB) zentrale Konzepte. Auch wenn das Steuergerät noch nicht physikalisch zur Verfügung steht, muss das Testen der „Software Components“ möglich sein. Dazu erlaubt der „DaVinci Component Tester“ die Einbindung des realen Steuergeräte-Codes einer SWC in die Testumgebung. Dabei übernimmt CANoe das Bereitstellen der RTE, des VFB und der Basissoftware. Der Test erfolgt mit den bekannten Funktionen aus CANoe, inklusive dem automatischen Erstellen eines Testreports.



**Bild 2:** Für CANoe.FlexRay steht ein leistungsfähiges Modul-Konzept zur Verfügung, dass das Einbinden von zusätzlichen Protokollen oder einer anwendungsspezifischen und kommunikativen Verhaltensteuerung der Simulation erlaubt.

**Ausblick**

In Zukunft wird die Summe der AUTOSAR-Steuergeräte und die Anzahl der darin enthaltenen Softwarekomponenten rapide wachsen. Für diese komplexen Systeme besteht Bedarf an einer leistungsfähigen Analysemöglichkeit der internen Kommunikation zwischen den SWCs. Dafür muss das Testwerkzeug auf die Kommunikation des VFB zugreifen können. Dies geschieht üblicherweise über XCP-on-FlexRay. Gerade bei umfangreichen Tests ist dies aus Performance-Gründen nicht ausreichend. Deshalb erfolgt die Ankopplung des Steuergerätes zusätzlich mittels einer JTAG- oder Nexus-Debug-Schnittstelle an die Testumgebung CANoe. Nur so können dann sehr schnell Parameter des VFB, der RTE oder der Basis-Software (BSW) eines realen AUTOSAR-Steuergerätes eingelesen und verstellt werden. Der VFB des realen Steuergeräts wird somit an die Testumgebung in CANoe angeschlossen. Ähnlich wie heute an den externen Steuergeräte-Anschlüssen kann damit künftig an den steuergeräte-internen Schnittstellen der SWCs analysiert und getestet werden.

Werden die beschriebenen Anforderungen technisch ausgefeilt in den Entwicklungswerkzeugen und bei der Implementierung von FlexRay-Systemen umgesetzt, so können FlexRay-Netzwerke wesentlich effizienter als CAN-Netzwerke entwickelt werden.

Übersetzung der englischen Veröffentlichung im FlexRay-Sonderheft Hanser Automotive, 11/2009

**Alle Bilder:**

Vector Informatik GmbH

**Links:**

Hompag Vector: [www.vector.com](http://www.vector.com)

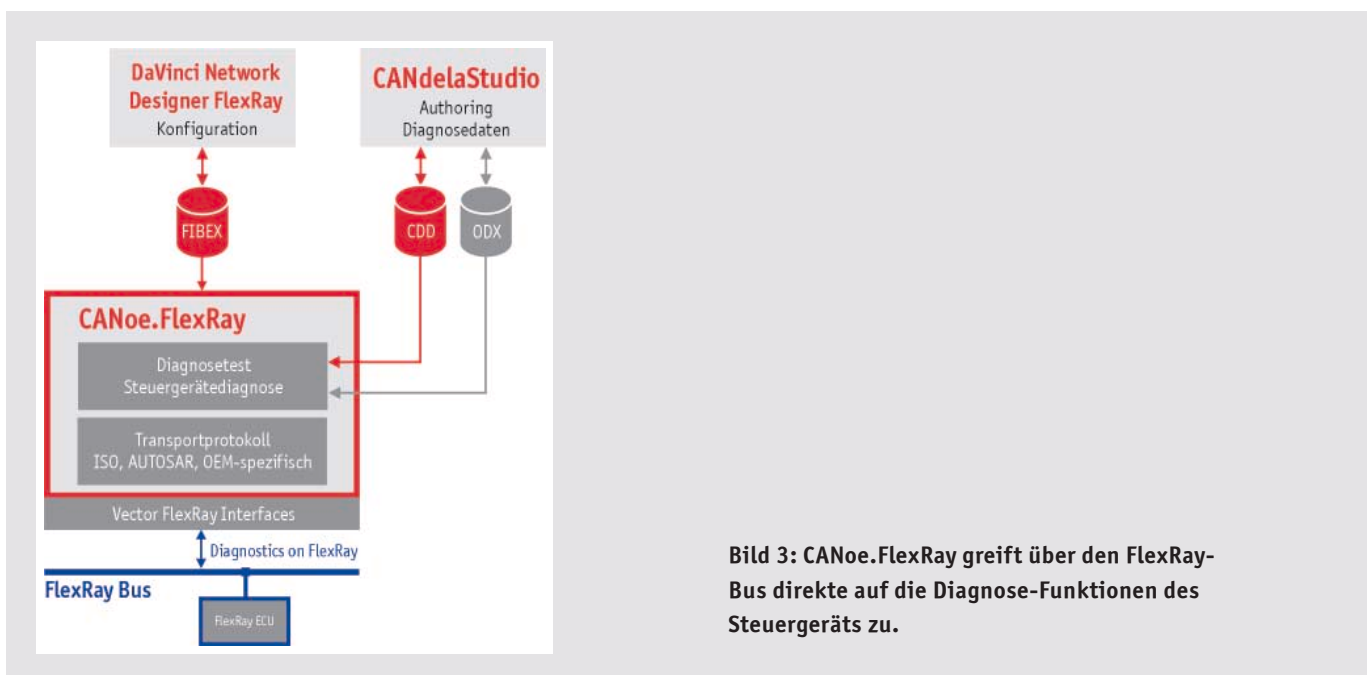
Product Information CANoe: [www.vector.com/canoe](http://www.vector.com/canoe)



**Carsten Böke (Dr. rer. nat.)**

Dr. rer. nat. Carsten Böke ist Senior Software Development Engineer in der Produktlinie "Tools for Networks and Distributed Systems"

Vector Informatik GmbH  
70499 Stuttgart  
Germany  
Tel. +49 (0)711/80670-4923,  
Fax +49 (0)711/80670-111,  
E-mail: [carsten.boeke@vector.com](mailto:carsten.boeke@vector.com)



**Bild 3: CANoe.FlexRay greift über den FlexRay-Bus direkte auf die Diagnose-Funktionen des Steuergeräts zu.**