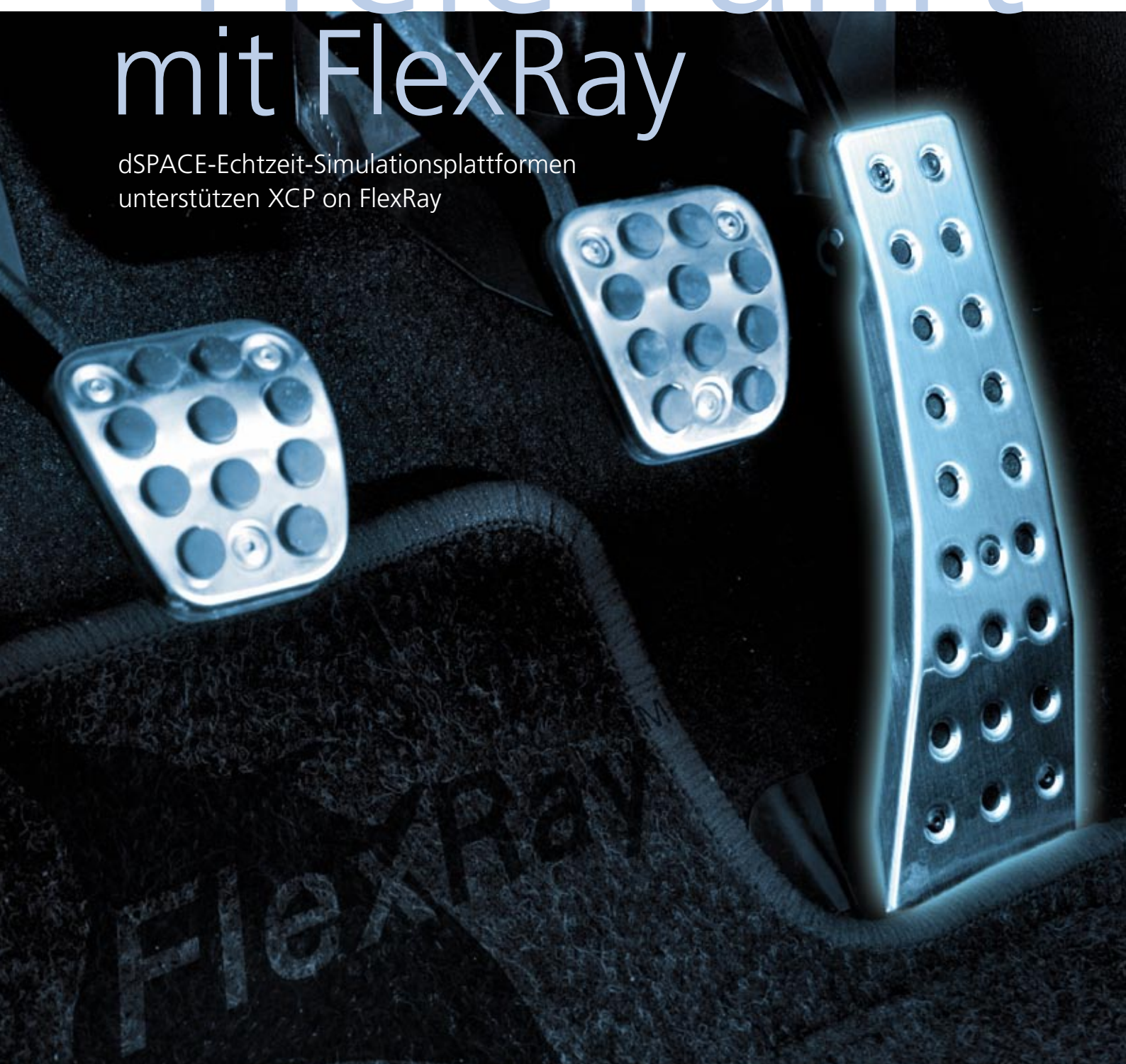
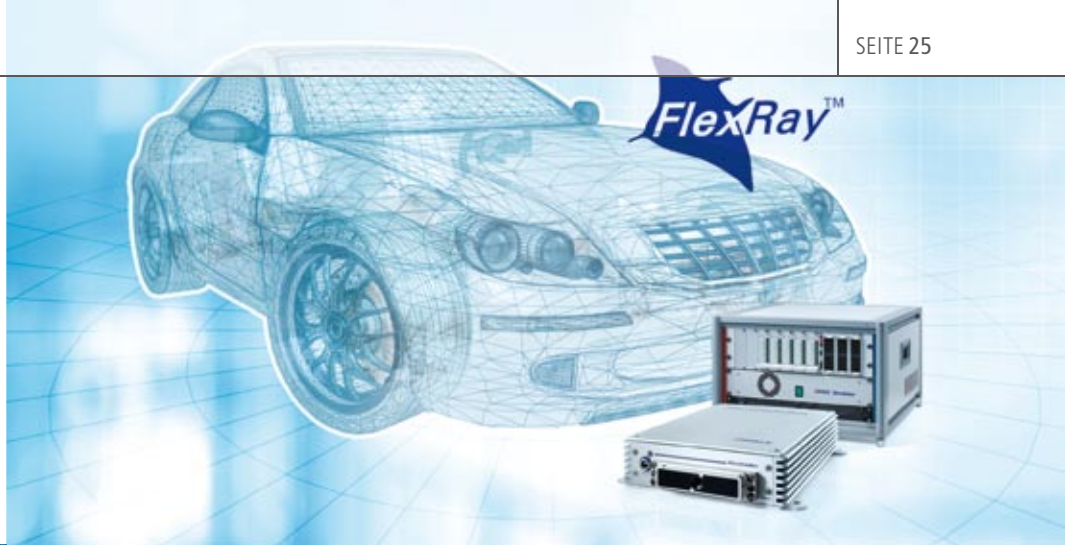


# Freie Fahrt mit FlexRay

dSPACE-Echtzeit-Simulationsplattformen  
unterstützen XCP on FlexRay





Die Steuergerätekommunikation über FlexRay gewinnt an Bedeutung. Für die Entwicklung und die Absicherung automotiver Regelungssysteme ist der Zugriff auf steuergeräteinterne Größen über standardisierte Protokolle notwendig. Die dSPACE-Echtzeit Simulationsplattformen unterstützen das leistungsfähige XCP-on-FlexRay-Protokoll. Typische Anwendungsfälle sind das Funktionsbypassing und der Steuergeräte-Test mittels Hardware-in-the-Loop-Simulation.

#### Anforderungen an moderne Bussysteme

Als CAN-basierte Busarchitekturen für den Einsatz im Auto entwickelt wurden, betrug der Datenverkehr auf automotiven Bussystemen nur einen geringen Bruchteil im Vergleich zu heute. LIN ist als besonders kostengünstiges Bussystem entwickelt worden und eignet sich aufgrund der geringen Datenübertragungsraten für Einsatzzwecke mit niedrigen Datenmengen und vergleichsweise langen Übertragungszeiten. Die Bussysteme CAN und LIN stoßen in modernen Fahrzeugen zunehmend an ihre Grenzen. Als ein Ausweg bietet sich der Übertragungsstandard FlexRay an, denn FlexRay ist schneller (heute 10 Mbit/s; zum Vergleich: CAN bietet maximal 1 Mbit/s, LIN maximal 20 Kbit/s) und erlaubt zugleich einen deterministischen Datenverkehr. Deterministisch bedeutet, dass Zeitfenster (Slots) zyklisch in exakt definierter Reihen-

folge abgearbeitet werden, in denen jeweils nur ein bestimmter Teilnehmer senden darf. Hierdurch ist sichergestellt, dass wichtige Nachrichten (z.B. an Lenkung oder Bremsen) innerhalb definierter Zeiträume abgearbeitet werden. Außerdem lassen sich auch Teilsysteme durch die exakte zeitliche Planung des Datenverkehrs einfach zusammensetzen.

#### FlexRay in der Serienentwicklung

Der FlexRay-Bus findet zunehmend Einsatz in Serienprojekten. Die Ausführung einzelner Tasks in den Steuergeräten wird dabei über FlexRay synchronisiert. Durch die deterministische Datenübertragung in Form von dedizierten FlexRay-Slots können die bei verschiedenen Zulieferern entwickelten und getesteten Steuergeräte ohne gegenseitige Beeinflussung zum Gesamtsystem zusammengesetzt werden (Abbildung 1). Für die Applikation und den Test dieser



Steuergeräte über eine einheitliche Schnittstelle kann das von ASAM e.V. standardisierte, universelle Mess- und Applikationsprotokoll XCP zum Einsatz kommen. XCP ist nicht nur unabhängig von der physikalischen Datenübertragungsschicht, sondern berücksichtigt sowohl heutige als auch zukünftige Kommunikationsschnittstellen. Mit der Einführung des FlexRay-Busses im Fahrzeug gewinnt XCP on FlexRay immer mehr an Bedeutung.

### RTI Bypass Blockset: Jetzt mit XCP on FlexRay

Um Steuergeräte-Entwickler noch besser für die Anforderungen an zukünftige Fahrzeug-Bussysteme zu rüsten, ermöglicht dSPACE nun die Kommunikation zwischen dSPACE-Echtzeit-Plattformen und Steuergeräten über XCP on FlexRay. Dabei übernimmt das dSPACE-System die Rolle des XCP-Masters, während die Steuergeräte XCP-Slaves darstellen. Möglich wird dies zum Beispiel mit der neuen XCP-on-FlexRay-Schnittstelle des RTI Bypass Blocksets von dSPACE, einem Simulink®-Blockset zur dialogbasierten Konfiguration

von Bypass-Schnittstellen und für den Steuergeräte-Echtzeitzugriff. Die FlexRay-Schnittstelle des RTI Bypass Blocksets setzt bei der Buskonfiguration auf dem dSPACE FlexRay Configuration Tool auf. Im dSPACE FlexRay Configuration Tool wird die FIBEX-Netzwerkbeschreibungdatei geladen und es werden neben der übrigen FlexRay-Kommunikation auch die XCP-on-FlexRay-Anteile konfiguriert. Das Ergebnis ist eine RTI FlexRay Blockset Library, deren Blöcke dem Simulationsmodell hinzugefügt werden können. Die Konfiguration der Mess- und Stimulidaten, die für den Austausch mit dem betreffenden Steuergerät vorgesehen sind, erfolgt in einem zweiten Schritt mit Hilfe des RTI Bypass Blocksets. Dazu liest man zunächst die ASAP2-Datei ein, die die verfügbaren Steuergerätevariablen enthält. Anschließend definiert man über RTI-Blöcke die Mess- und Stimulizugriffe auf das Steuergerät (Abbildung 2). Mit dem RTI Bypass Blockset lassen sich über denselben FlexRay-Bus bis zu vier Steuergeräte parallel ansprechen.

### Flexible Konfiguration im RTI Bypass Blockset

Für die Definition der XCP-Slots pro Steuergerät bietet das RTI Bypass Blockset spezielle Konfigurationsdialoge, die den Abgleich der FlexRay-Kommunikationsoptionen (FlexRay-Puffer) des einzelnen Steuergeräts mit den tatsächlich zu verwendenden FlexRay-Slots erlauben. Der Anwender kann so die Übertragungsbandbreite für die Kommunikation mit dem individuellen Steuergerät und die Zeitpunkte der Datenübertragung im FlexRay-Zyklus frei definieren (Abbildung 3, Abbildung 4). Optional lassen sich die für die jeweilige Datenübertragung zu verwendenden XCP-Slots entweder manuell vorgeben oder automatisch durch das RTI Bypass Blockset zuweisen. Ist die genaue Lage der XCP-Slots im Kommunikationszyklus in Relation zu den Freischnitten innerhalb der Steuergeräte-Tasks bekannt, kann der Anwender somit die Zeitpunkte der Datenerfassung und Stimulation optimal vorgeben. Ist die Lage der XCP-Slots bezogen auf die Freischnitte nicht bekannt, bietet sich eine automatische Zuweisung an. In diesem Fall werden anfallende Daten-

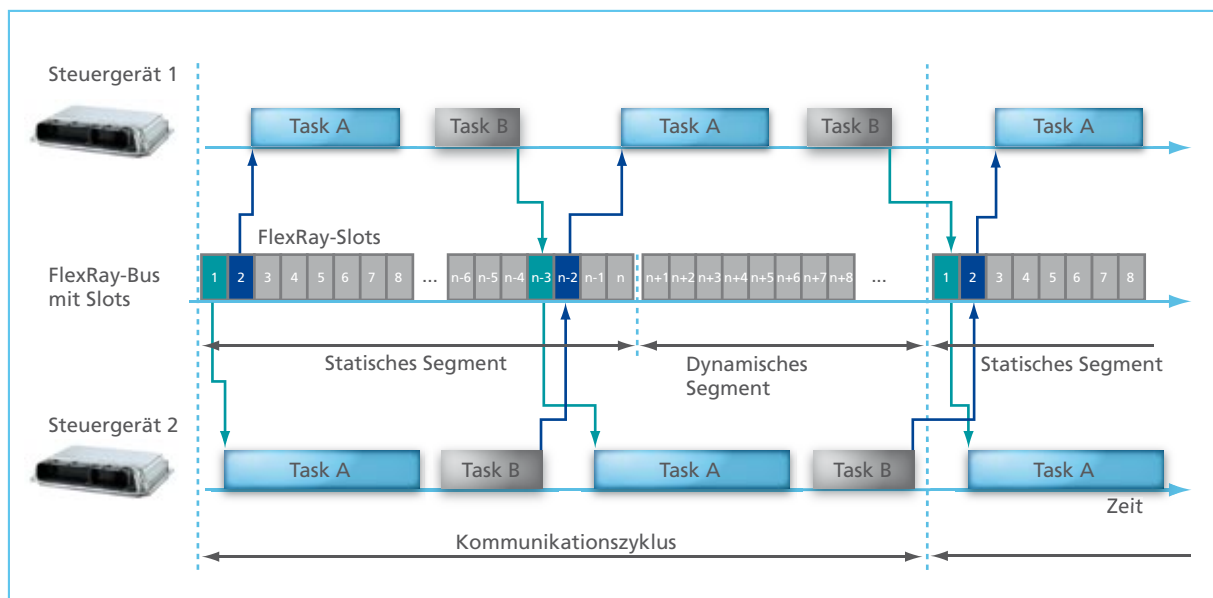


Abbildung 1: Synchronisierung von Steuergeräte-Tasks über zugewiesene FlexRay-Slots. Das Gesamtsystem kann ohne gegenseitige Beeinflussung der einzelnen Steuergeräte zusammengesetzt werden.

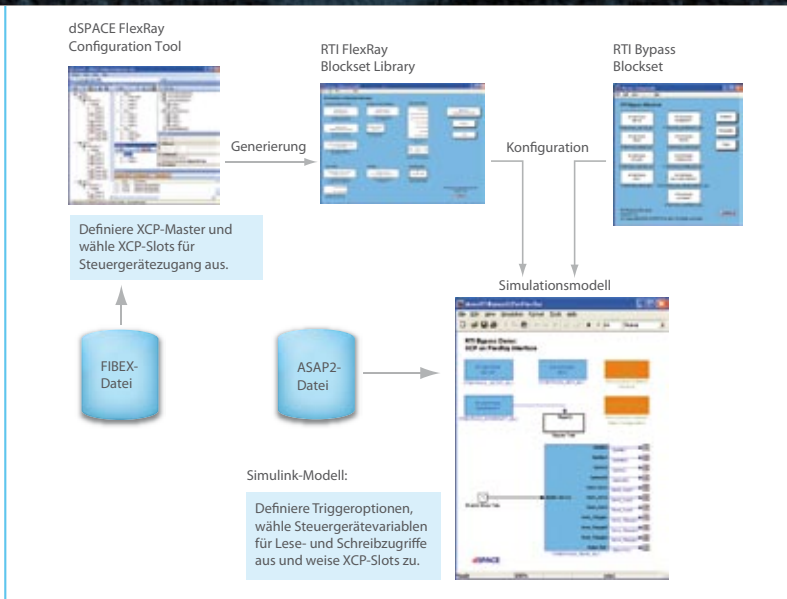


Abbildung 2: Konfigurationsprozess mit dem dSPACE FlexRay Configuration Tool und dem RTI Bypass Blockset.

pakete mit dem jeweils nächsten verfügbaren XCP-Slot übertragen. Unabhängig vom gewählten Mechanismus kann zudem im Falle einer Mehrfachverwendung von XCP-Slots (z.B. XCP-Slot kommuniziert abwechselnd mit Steuergerät 1 und 2) die Reihenfolge der vom Simulationssystem zu übertragenden Daten priorisiert werden. Zur Synchronisation der Echtzeit-Plattformen mit dem Steuergerät stehen

unterschiedliche Mechanismen zur Verfügung. Zum einen lassen sich Berechnungen auf der Simulationsplattform synchron zu bestimmten Zeitpunkten innerhalb des FlexRay-Zyklus starten; die Ausführung erfolgt hierbei zeitgesteuert. Zum anderen ist der Start von Berechnungen auch unmittelbar nach Abschluss der zugehörigen Datenübertragung möglich.

### Anwendungsfall 1: Bypassing über XCP on FlexRay

Im Rahmen der Funktionsentwicklung ermöglicht das RTI Bypass Blockset das Entwickeln und Erproben neuer Steuergeräte-Funktionen auf einer Echtzeit-Simulationsplattform, wobei der Datenaustausch und die Synchronisation mit dem Steuergerät über die existierende FlexRay-Schnittstelle erfolgen. Um den Datenaustausch zu ermöglichen, werden durch XCP-Serviceaufrufe oder Freischnitte an definierten Stellen im Steuergerätecode alle notwendigen Daten an das Entwicklungssystem übertragen. Dieses berechnet dann die neue Funktion (Bypass-Task) und überträgt die Ausgangsgrößen zurück ans Steuergerät, wo sie mit einem XCP-Freischnitt in den Speicher des Steuergeräts übernommen werden. Diese Sequenz kann dabei innerhalb desselben FlexRay-Zyklus durchgeführt werden (Funktions-Bypass ohne Datenversatz). Wegen des engen zeitlichen Zusammenhangs zwischen den Task-Ausführungen und der FlexRay-Kommunikation findet diese typischerweise im

Abbildung 3: Das RTI Bypass Blockset bietet spezielle Dialoge zur manuellen Konfiguration der für XCP zu verwendenden FlexRay-Slots. Alternativ kann die Zuordnung der Slots automatisch erfolgen.

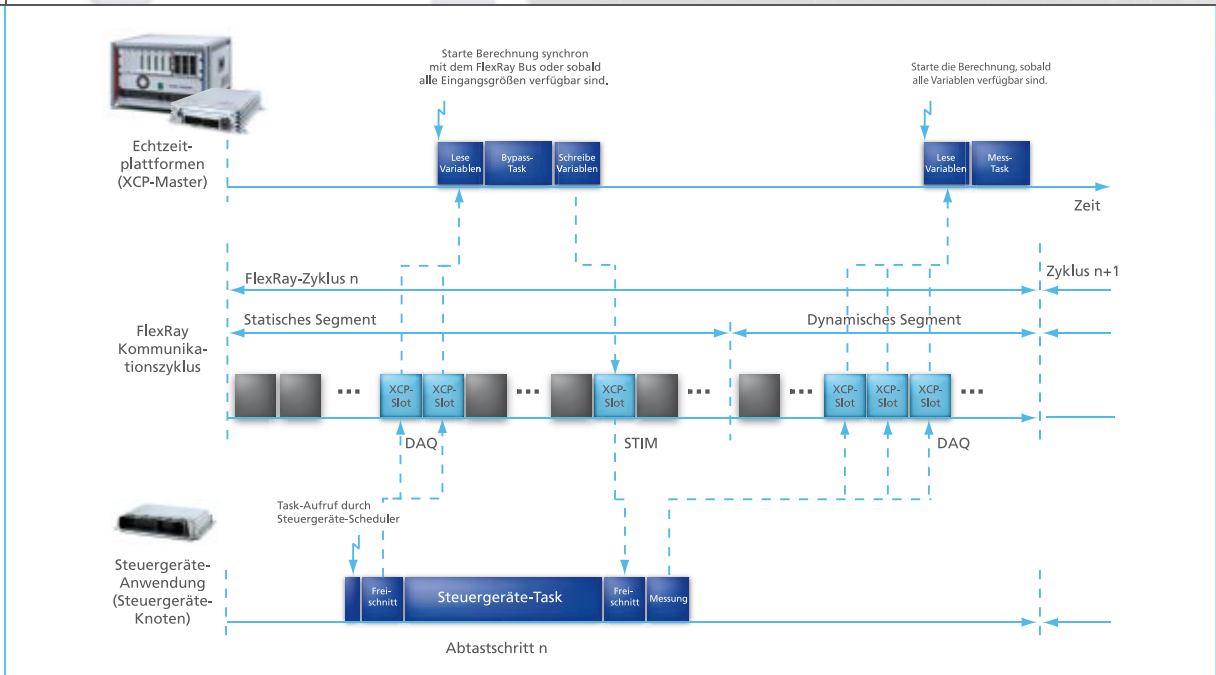


Abbildung 4: FlexRay-Kommunikationssequenz für einen Funktionsbypass ohne Datenversatz sowie für eine Messdatenerfassung.

statischen Teil des FlexRay-Zyklus statt (Abbildung 4). Bei Anwendungen mit Datenversatz rechnet das Steuergerät mit Funktionsergebnissen des letzten Abtastschrittes. In solchen Szenarien erfolgt die FlexRay-Kommunikation zum Beispiel im dynamischen Segment des FlexRay-Zyklus.

**Anwendungsfall 2: Datenerfassung im HIL-Szenario**  
 Eine weitere typische XCP-on-FlexRay-Anwendung des RTI Bypass Blocksets ist das Überwachen von steuergeräteinternen Größen in einem Hardware-in-the-Loop (HIL)-Szenario. Dazu nutzt man in der Regel im Steuergerät vorhandene

XCP-Serviceaufrufe am Ende der Steuergeräte-Tasks. Die XCP-Kommunikation erfolgt meist über das dynamische Segment des FlexRay-Zyklus (Abbildung 4). Simulations- und steuergeräteinterne Daten können dabei exakt zeitlich zueinander korreliert aufgezeichnet werden. ■

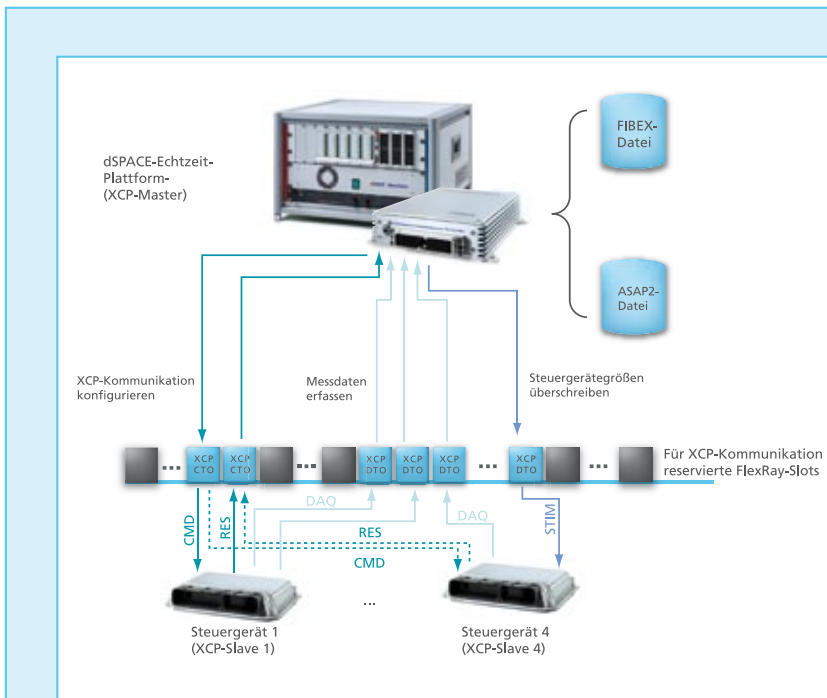


Abbildung 5: XCP-on-FlexRay-Kommunikation: Command Transfer Objects (CTO) werden für Kommando-Antwort-Sequenzen (CMD, RES) verwendet, Data Transfer Objects (DTO) für Mess- und Stimuldaten (DAQ, STIM). dSPACE-Simulationsplattformen unterstützen pro FlexRay-Bus einen parallelen Zugriff auf bis zu vier XCP-Slaves.

## XCP-on-FlexRay-Technologie

### XCP: Das universelle Mess- und Applikationsprotokoll

Zur Entwicklung und für die Absicherung von Steuergeräten ist es notwendig, parallel zur statisch vordefinierten Buskommunikation zusätzliche Daten dynamisch mit dem Steuergerät auszutauschen. Typische Anwendungen hierfür sind das Verstellen von Kennwerten (Applizieren), das steuergerätesynchrone Erfassen von Messdaten (DAQ) und das zyklische Überschreiben von Variablen im Steuergeräte-RAM (STIM). Für diese Anwendungen ist das XCP-Protokoll maßgeschneidert. Es basiert auf einem Master-Slave-Konzept, bei dem der PC oder die Echtzeit-Simulationsplattform die Rolle des XCP-Masters, und die

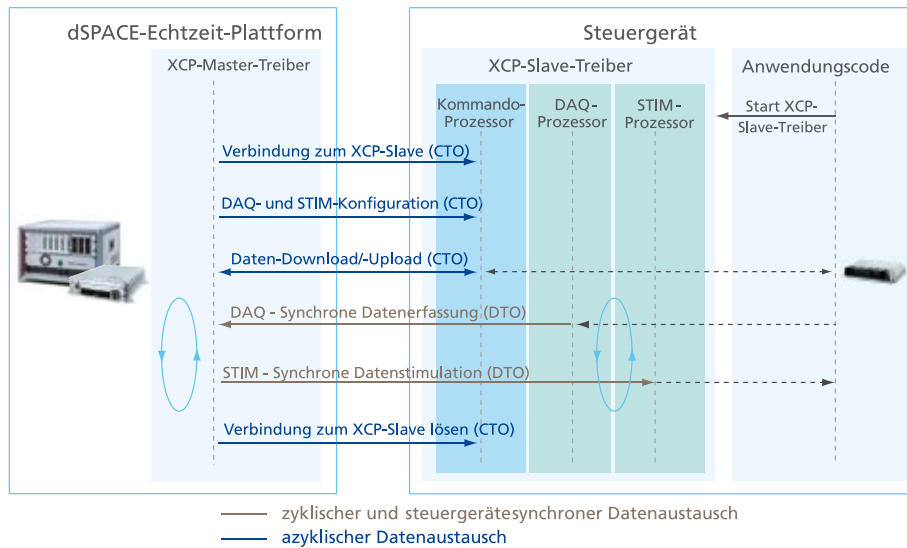


Abbildung 6: XCP-Kommunikationssequenzen. Der XCP-Slave-Treiber besteht aus den drei Komponenten Kommando-, DAQ- und STIM-Prozessor.

Steuergeräte die der XCP-Slaves spielen. Die XCP-Treibersoftware bleibt in der Regel als Bestandteil der Seriensoftware im Steuergerät. Ausgehend vom XCP-Master wird während der Laufzeit des Systems die XCP-Kommunikation über Kommando-Antwort-Sequenzen mittels XCP Command Transfer Objects (CTO) konfiguriert. Der nicht zyklische Datenaustausch (Download/Upload), zum Beispiel für das Applizieren, erfolgt ebenfalls über XCP-Kommandopakete. Nach Abschluss der Konfiguration findet die steuergerätesynchrone Übertragung der Mess- und Stimulidaten zyklisch über spezielle XCP Data Transfer Objects (DTO) statt, die vom Empfänger nicht beantwortet werden müssen und somit eine höhere Bandbreite bieten (Abbildung 6).

### Schnell und flexibel: XCP on FlexRay

Bei XCP on FlexRay besteht eine Herausforderung darin, die XCP-typische ereignisorientierte Kommunikation, wie beispielsweise über CAN, in die deterministische Kommunikation des FlexRay-Busses zu integrieren. Dazu sind im FlexRay-Kommunikationszyklus

exklusiv Slots für die XCP-Kommunikation reserviert und in der FIBEX-Netzwerkbeschreibungsdatei als XCP-Slots gekennzeichnet. Die XCP-Kommunikation über FlexRay mit allen Steuergeräten erfordert wegen der in den CTO-Paketen enthaltenen Slave-bezogenen Knotenadresse prinzipiell nur zwei einzelne XCP-Slots für den Austausch von Kommando-Antwort-Sequenzen (CMD, RES). Je nach benötigter Bandbreite, insbesondere im Hinblick auf die zyklische Übertragung von Mess- und Stimulidaten (DAQ, STIM), lässt sich die Kommunikation um weitere XCP-Slots ergänzen. Generell können XCP-Slots sowohl im statischen als auch im dynamischen Segment des FlexRay-Kommunikationszyklus liegen. Die Mehrzahl der XCP-Slots findet man in der Regel im dynamischen Segment, um für Messaufgaben Bandbreite nicht statisch allozieren zu müssen. Das statische Segment ist in der Regel für die deterministische Kommunikation vorgesehen. Die Zuteilung der XCP-Slots im FlexRay-Zyklus erfordert schon während der Busplanung eine Berücksichtigung sowohl der insgesamt für XCP zur

Verfügung gestellten Bandbreite als auch der Zeitpunkte der XCP-Kommunikation. Bei einigen Anwendungen werden bis etwa 30% der möglichen Bandbreite des FlexRay-Busses für die XCP-Kommunikation bereitgestellt.

### Paralleler Zugriff auf mehrere Steuergeräte

Für den parallelen Zugriff auf mehrere Steuergeräte in einem FlexRay-Netzwerk lassen sich beliebige Teilmengen der XCP-Slots für die jeweilige Kommunikation nutzen. Dies führt zu einer dynamischen Aufteilung der insgesamt verfügbaren Bandbreite. In der ASAP2-Beschreibungsdatei ist die jeweilige XCP-Slave-Konfiguration des einzelnen Steuergeräts abgelegt. Dazu zählen Informationen zu steuergeräteinternen FlexRay-Puffern, die für einen statisch vordefinierten oder dynamisch zur Laufzeit konfigurierbaren Austausch von CTO- und DTO-Paketen genutzt werden können (Abbildung 5).