

Umfangreiche Testsysteme aufbauen

SCALEX



SCALEXIO, das neue Hardware-in-the-Loop (HIL)-System von dSPACE, wurde seit seiner Einführung 2011 kontinuierlich erweitert. Die neuste Version unterstützt den Anwender beim Aufbau besonders umfangreicher Testsysteme mit komplexen Umgebungsmodellen sowie der entsprechenden Fehlersimulation.

Größere Systeme durch Multi-core

Durch den steigenden Anteil von Elektronik in Fahrzeugen erhöht sich auch der Funktionsumfang der entsprechenden Steuergeräte. Die daraus resultierenden komplexeren Steuergeräte mit viel I/O und zahlreichen Software-Komponenten erfordern leistungsfähige Testsysteme. Gleichzeitig kommt meist ein sehr umfangreiches Umgebungsmodell zum Einsatz, um beispielsweise das gesamte Fahrzeug abzudecken. Die Steuergeräte werden zusammen mit diesen Umgebungsmodellen in umfangreichen Hardware-in-the-Loop (HIL)-Simulationen getestet. Um jederzeit die Echtzeitsimulation komplexer Modelle garantieren zu können, beinhaltet die SCALEXIO® Processing Unit einen Intel® Core™ i7 mit vier Kernen. Auf einem der Kerne werden die Systemprozesse gerech-

net, die übrigen drei stehen vollständig der Modell- und I/O-Berechnung zur Verfügung. Hierdurch lassen sich mit SCALEXIO sehr umfangreiche und komplexe Modelle simulieren, mit deren Echtzeitsimulation ein einzelner Kern überfordert wäre.

Zwei Varianten für die Modellaufteilung

Um das Umgebungsmodell parallel auf mehreren Kernen berechnen zu können, ist eine möglichst flexible Modellaufteilung hilfreich. Gleichzeitig sind besonders ein einfacher Workflow, eine klare Arbeitsaufteilung für die Modellentwickler sowie kurze Kompilierungszeiten wichtig. Bei der Modellverwaltung muss sowohl die Kommunikation der Teilmodelle untereinander als auch die Verbindung der Modellschnittstellen zu den I/O-Karten übersichtlich gestaltet sein.

IO

ausgebaut

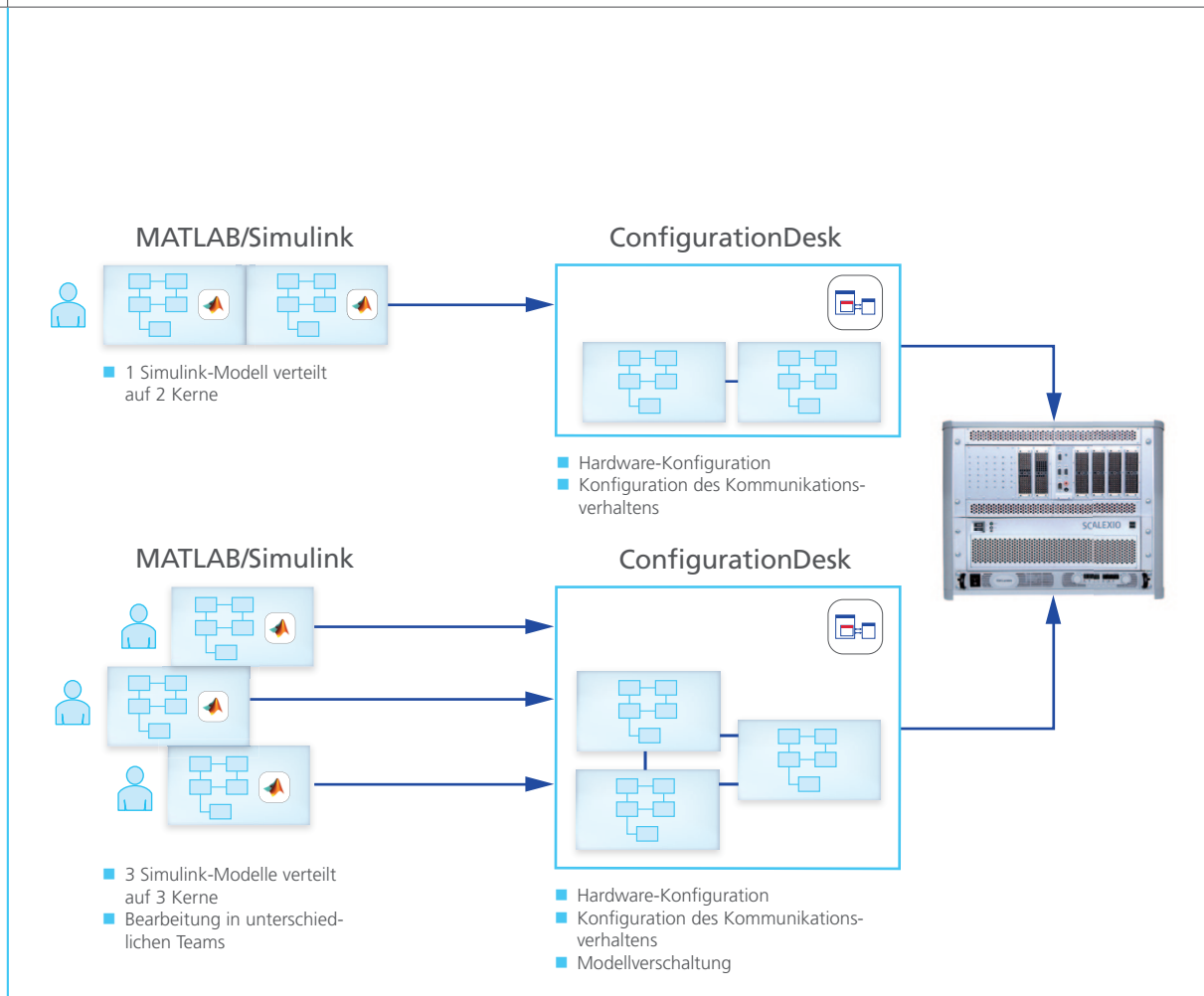


Abbildung 1: Alternative Workflows für die Modellaufteilung.

Für die Modellerstellung und die Aufteilung auf verschiedene Prozessorkerne gibt es zwei unterschiedliche Ansätze (Abbildung 1):

1. Das Modell wird als Gesamtsystem entwickelt und anschließend auf die Kerne aufgeteilt. Hierzu muss das Gesamtsystem derart gestaltet sein, dass es mehrere „schneidbare“ Modellteile enthält. In jedem dieser Modellteile kann eine beliebige Anzahl von Subsystemen realisiert werden, in denen referenzierte Modelle möglich sind. Durch einen speziellen von dSPACE entwickelten MATLAB®/Simulink®-Block werden Modellteile direkt einem Prozessorkern zugewiesen, es sind also keine IPC (Inter Processor Communication)-Blöcke notwendig. Hierdurch ergibt sich eine leichtere Verwaltung des Modells und bei Änderungen in der Zuweisung müssen nur die betroffenen Modellteile erneut kompiliert werden. Da Änderungen direkt im Gesamtmodell erfolgen, ist eine Offline-Simulation in Simulink möglich. Je nach Größe des Modells können längere Lade- und Kompilierzeiten erforderlich sein.
2. Anstelle eines großen Gesamtsystems wird das Simulink-Modell von Beginn an in individuellen Teilmodellen entwickelt. In einem Teilmodell werden Vorgänge zusammengefasst, die eng miteinander gekoppelt sind, zwingend nacheinander berechnet und somit auf einem Prozessorkern untergebracht werden müssen. Jedes Modell wird dann genau einem Kern zugewiesen. Änderungen werden nur in den einzelnen Teilmodellen durchgeführt, wodurch sich kurze Lade- und Kompilierzeiten ergeben. Die Nut-

zung von Teilmodellen hat den Vorteil, dass verschiedene Teams zeitgleich arbeiten können.

Grafische Konfiguration mit ConfigurationDesk

Bei der Erstellung des Modells als Gesamtsystem erfolgt die Konfiguration der Modellkommunikation und die der I/O-Karten mit dSPACE ConfigurationDesk®. Gleiches gilt bei der Erstellung von Teilmodellen, wobei noch die Modellverschaltung hinzukommt. Aus Simulink heraus werden sowohl die Modellkomponenten als auch die entsprechenden Kommunikationsdateien importiert und grafisch in ConfigurationDesk dargestellt. Diese Vorgehensweise ist bei beiden Modellaufteilungsvarianten identisch. Durch die Kommunikationsdateien wird die Weitergabe von Informationen erleichtert, nachträgliche Änderungen können einfach umgesetzt

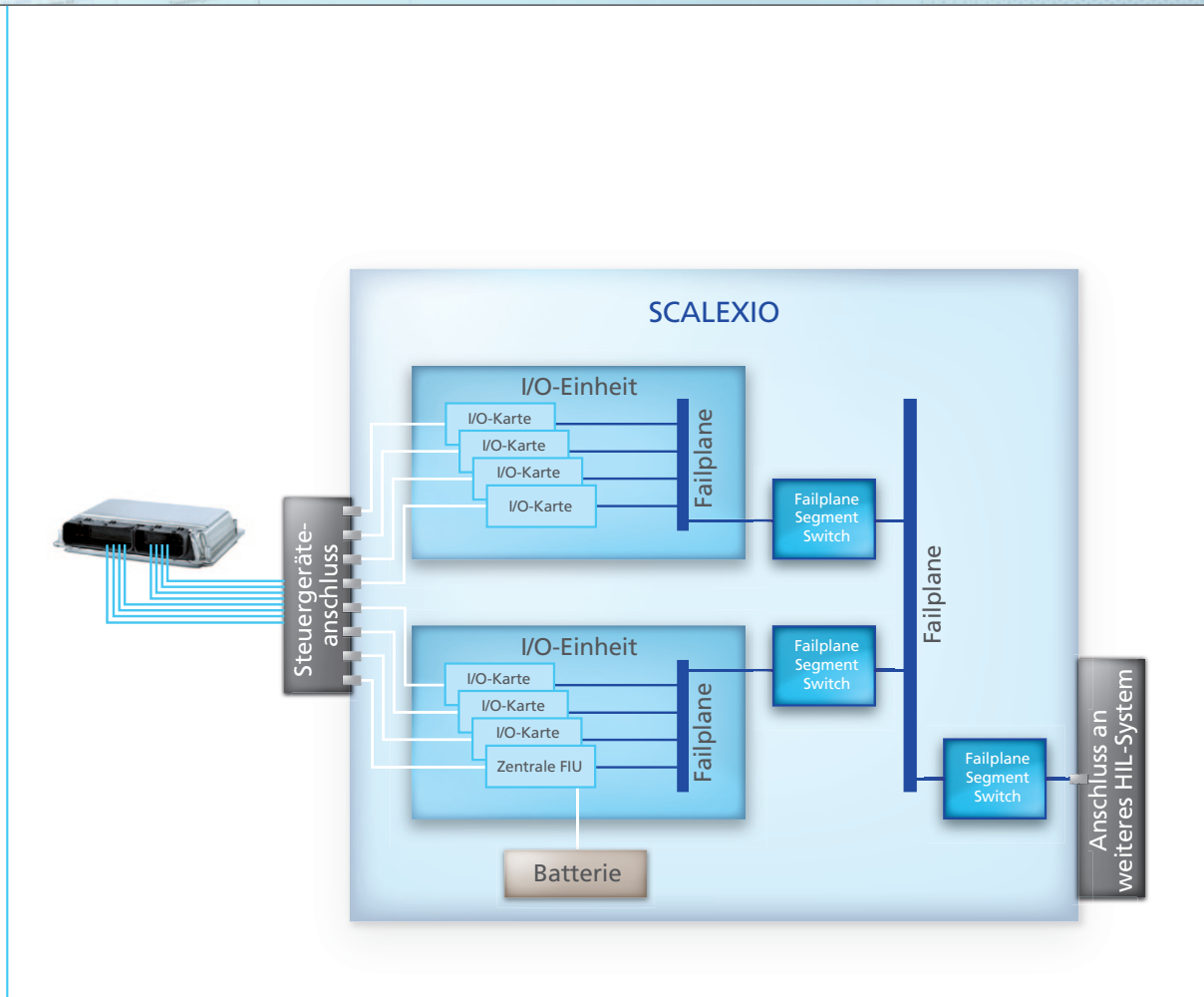


Abbildung 2: Infrastruktur der FIU bei größeren Systemen.

werden. In ConfigurationDesk lassen sich zudem Konfigurationssets erstellen. Damit lassen sich pro Teilmodell verschiedene Build-Versionen generieren, die für bestimmte Compiler optimiert sind. Trotz unterschiedlicher Rahmenbedingungen, beispielsweise unterschiedlicher Speicherplatzgrößen, wird somit möglichst effizienter Code erzeugt.

Bessere Signalqualität während der Fehlersimulation

Große SCALEXIO-Systeme mit hoher I/O-Kanalanzahl erfordern auch bei der Fehlersimulation angepasste Lösungen (Abbildung 2). Mit einer Failplane Insertion Unit (FIU) werden elektrische Fehler an den I/O-Pins des Steuergeräts aufgeprägt. Um diese für Fehler bei allen Steuergeräte-Signalen zu nutzen, können alle Signale über sogenannte Failplanes zu den zentralen FIU-Kompo-

nenten geleitet werden. Dabei stört es nicht, wenn sich dies bei größeren Aufbauten über mehrere Schränke erstreckt, da die Failplanes der einzelnen Schränke miteinander verbunden werden können und sich bei steigender Anzahl an I/O-Kanälen erweitern lassen. Die Failplanes sind in Segmente unterteilt, die einzeln zugeschaltet werden können. Hierdurch werden die Leitungskapazitäten durch lange Leitungen, die eine Verfälschung der Signale verursachen können, minimiert und eine hohe Signalqualität gewährleistet. Wenn ein Fehler aufgeprägt werden soll, werden nur die für die Signalführung benötigten Segmente durch die Failplane Segment Switches zugeschaltet. Die Segmentierung erfolgt hinter jeder I/O Unit und für jeden Schrank. Systemerweiterungen sind somit problemlos und ohne Qualitätseinbußen möglich. Die Ansteuerung

des Failplane Segment Switch erfolgt direkt durch SCALEXIO, ohne dass der Anwender sich damit beschäftigen muss.

Frei Wahl bei Netzteilen

Mit einem SCALEXIO-System kann die Batteriesimulation direkt aus dem Simulationsmodell heraus angesprochen werden. Für die Steuerung des Netzteils wird der DS2907 Battery Simulation Controller eingesetzt, der Strom- und Spannungswerte vorgibt. In einem SCALEXIO-System können pro DS2907 bis zu zwei verschiedene Netzteile verwendet werden. Damit können Bordnetze mit unterschiedlichen Spannungen realisiert werden, um das korrekte Verhalten des Steuergeräts zu testen. ■