

Rapid Prototyping und Simulation verteilter Echtzeitsysteme

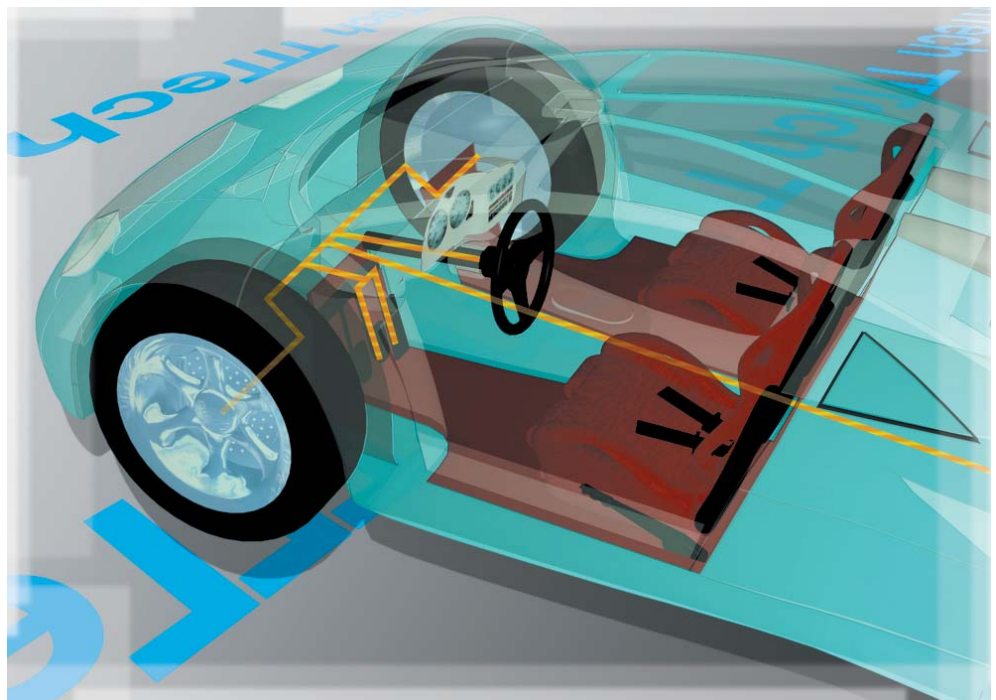
Design und Simulation verteilter Applikationen

Ein neues Blockset für Simulink ermöglicht intuitives Design und eine rasche Systementwicklung durch Integration des TTP-Busverhaltens in das funktionale Modell und automatische Code-Generierung für verteilte Systeme. ENSIO HOKKA, MARKUS PLANKENSTEINER



Dr. Ensio Hokka ist Senior Developer bei der TTTech Computertechnik AG. Er studierte Verfahrenstechnik mit Schwerpunkt Prozessautomatisierung und Datentechnik.

Dr. Markus Plankensteiner ist Marketingleiter bei der TTTech Computertechnik AG. Er studierte Informatik, Mathematik und Wirtschaft und ist seit über 15 Jahren in der IT-Branche tätig.



Die nahtlose Integration elektronischer Subsysteme stellt für Design und Implementierung zuverlässiger Echtzeitsysteme eine schwierige Herausforderung dar. Der Austausch herkömmlicher mechanischer Teilsysteme

durch elektronisch gesteuerte By-Wire-Systeme, die sich aus Komponenten von unterschiedlichen Lieferanten zusammensetzen, verstärkt dieses Problem. Daher drängen Systemintegratoren auf eine zusammensetzbare Netzwerkarchitektur, die eine klare Trennung zwischen

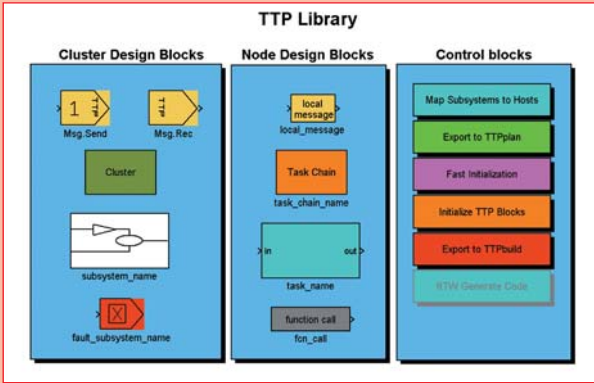


Abb. 1: TTP-Matlink – Design- und Steuerungs-Blockset-Bibliothek

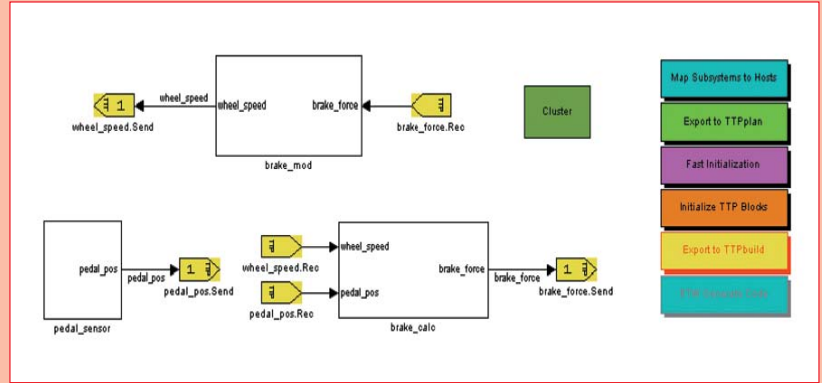


Abb. 4: Simulink-Modell mit TTP-Botschaften und Steuerungsblöcken

Gesamtsystem und Teilsystemen bereitstellt. Erschwerend kommen noch hohe Sicherheits- und Fehlertoleranzanforderungen dazu. Das Wachstum im Bereich elektronischer Bauelemente impliziert einen Bedarf an neuen Entwicklungsmethoden und Verfahren, um die Einführungszeit der Produkte zu beschleunigen. Die Anforderungen an sicherheitskritische Echtzeitsysteme erschweren zusätzlich den Aufbau eines elektronisch gesteuerten Prototypen in kurzer Zeit. Zudem wird hierfür als Basis eine zuverlässige Kommunikationsarchitektur und eine komfortable und zeitsparende Tool-Umgebung benötigt.

Für das Prototyping ist eine klare und intuitive Designstrategie auf hohem Abstraktionsniveau erforderlich. Dabei sollen Designstrategien, Methoden und Verfahren, die für die Implementierung genutzt werden, Ergebnisse liefern, die in den folgenden Stufen der Entwicklung so effizient wie möglich weiterverarbeitet werden können.

Als Kommunikationstechnologie für zuverlässige Echtzeitsysteme ist eine zeitgesteuerte Architektur (Time-Triggered Architecture) Voraussetzung. Für das Design der Regelkreise und die Simulation solcher verteilter Systeme ist es jedoch notwendig, Kommunikationsverzögerungen, die durch den Datenbus und Laufzeiten der Berechnungsprozeduren verursacht werden, zu berücksichtigen. Nur so kann der Designer optimale Algorithmen für die Regelungs- und Steuerungsaufgaben entwickeln.

Matlab, Simulink und Stateflow sind insbesondere in der Automobilindustrie bewährte und weit verbreitete Entwicklungs- und Simulationswerkzeuge. ‚TTP-Matlink‘ ergänzt diese Tools nicht nur durch ein Blockset, das das zeitgesteuerte Kommunikationsverhalten in das Simulationsmodell einbringt, sondern ermöglicht auch die automatische Code-Generierung für die verteilten Applikationen zusammen mit dem Real-Time Workshop Embedded Coder.

TTA und TTP

Die zeitgesteuerte Kommunikationsarchitektur TTA (Time-Triggered Architecture) wurde speziell für verteilte Echtzeitsysteme mit hohen Sicherheitsanforderungen entwickelt. Das Herzstück von TTA ist das zeitgesteuerte Kommunikationsprotokoll TTP (Time-Triggered Protocol).

In TTP werden alle Aktivitäten des Kommunikationssystems gemäß einem vordefinierten, systemweit bekannten Ablaufplan durchgeführt. Die dafür notwendige globale Zeitbasis wird durch die von den TTP-Controllern autonom durchgeführte, fehlertolerante Uhrensynchronisation definiert. Der Buszugriff erfolgt mit dem Zeitschlitzverfahren TDMA (Time Division Multiple Access).

Zwischen einem Kommunikationscontroller und dem Host-Prozessor befindet sich eine als CNI (Communication Network Interface)

bezeichnete Datenschnittstelle, über die ausschließlich Dateninformation und keine Steuersignale ausgetauscht werden. Somit kann der Hostprozessor auch im Fehlerfall keinen Einfluss auf das Kommunikationsverhalten des TTP-Controllers nehmen. Dieser Schutzmechanismus, bezeichnet als Temporal-Firewall, verhindert die Fortpflanzung von Fehlern und bedingt, dass für alle Echtzeitnachrichten im System der Update-Zeitpunkt von vornherein spezifiziert und allen Busteilnehmern bekannt ist. Aus dem zeitgesteuerten Prinzip ergibt sich eine konstante Buslast, die nur durch den deterministischen Ablaufplan bestimmt wird. Der vom Systemintegrator festzulegende zeitliche Ablauf führt zu genau spezifizierten Schnittstellen im Zeit- und Wertebereich. Das Kommunikationssystem garantiert autonom, dass das zeitliche Verhalten dem Design entspricht, unabhängig davon, ob nur ein Teil des Systems oder alle Knoten aktiv sind. Diese Eigenschaft zielt direkt auf die Komplexitätsproblematik verteilter Systeme ab und garantiert eine problemlose Zusammensetzbarkeit (composability) der Komponenten, da sich das Kommunikationsverhalten bei einer Systemintegration nicht verändern kann. Dieser Umstand ist für die Zertifizierung von sicherheitskritischen Anwendungen von großem Vorteil, da Bereiche verschiedener Sicherheitsniveaus definiert und unabhängig zertifiziert werden können. Ein entsprechend konfiguriertes TTP-System erkennt und toleriert jeden Einzelfehler,

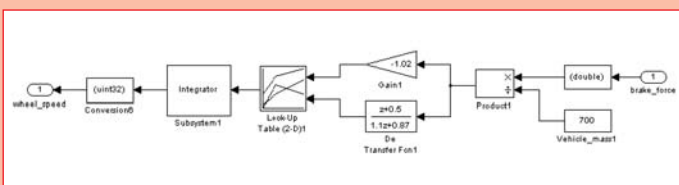


Abb. 2: Beispiel Modell eines Bremssystems

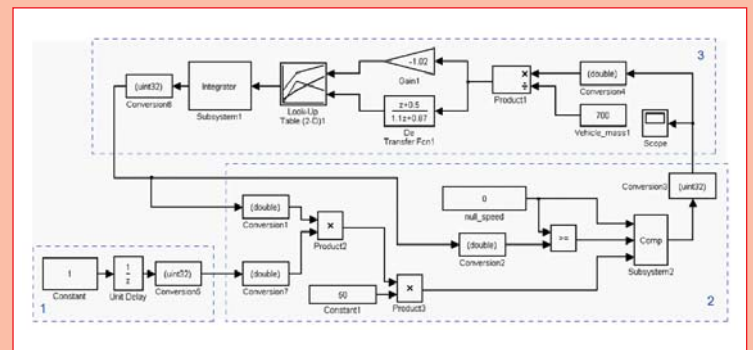


Abb. 3: Gruppierung des Modells in Subsysteme

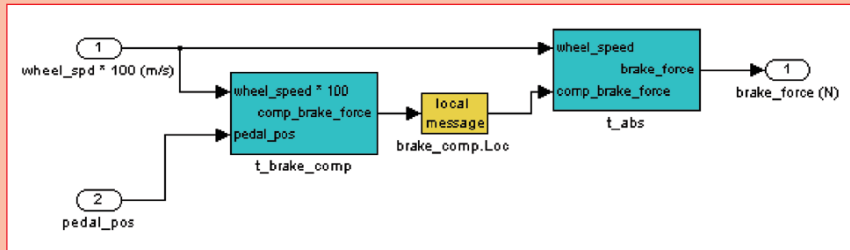


Abb. 5: Subsystem mit lokaler Botschaft

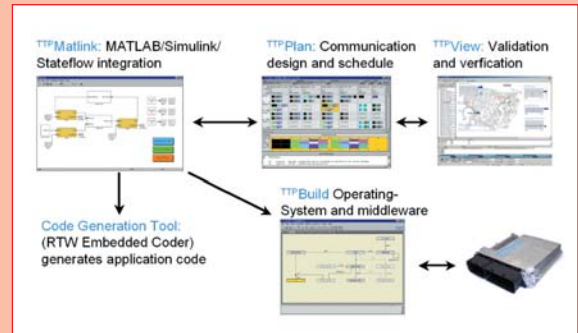


Abb. 6: Die Verknüpfung von TTP-Matlink mit den TTP-Tools und RTW Embedded Coder

dessen Folgefehler und darüber hinaus eine Vielzahl an Mehrfachfehlern. Durch die TTP-immanente Stationszustandsinformation (membership service), den Bus-Wächter (bus guardian), die Fehlertoleranzschicht (FT-COM Layer) sowie einem doppelt ausgeführten Übertragungskanal und die Unterstützung von aktiver Redundanz durch Replica-Determinismus wird ein zuverlässiger Kommunikationsablauf sichergestellt, eine rasche Fehlererkennung gewährleistet und somit die Applikationssoftware deutlich entlastet.

Die Funktionalität von TTP-Matlink

Die Verteilung von Algorithmen auf mehrere Steuereinheiten und die damit verbundene Kommunikation über einen Datenbus mit dem TDMA-Kommunikationsverfahren führen zu einer Veränderung des Zeit- und Systemverhaltens eines Regelungs- und Steuerungsdesigns. TTP-Matlink (Abb. 1) bietet die gesamte Funktionalität, um diese Einflüsse im funktionalen Modell zu berücksichtigen. Die Blockset-Bibliothek integriert Matlab und Simulink mit der TTP-Entwicklungsumgebung ‚TTP-Tools‘ und erlaubt es dem Anwender, während des gesamten Entwicklungsprozesses mit einer einzigen Werkzeugkette auf hoher Abstraktions-ebene zu arbeiten.

Das funktionale Modell – die Designschritte

Der erste Schritt beim Design ist der Aufbau des funktionalen Modells mit Matlab/Simulink bzw. Stateflow. Dieser Schritt umfasst die traditionelle Entwicklungsarbeit beim Entwurf von Regelungs- und Steuerungslösungen (Abb. 2). Im Anschluss wird das komplexe Modell in logische Einheiten (Subsysteme) gegliedert (Abb. 3). Die Gruppierung berücksichtigt, dass einzelne Subsysteme letztendlich auf verschiedenen Host-CPU's laufen können. Dabei müssen replizierte Komponenten nicht in alle Instanzen einbezogen werden, wodurch das Systemdesign wesentlich beschleunigt und Änderungen erleichtert werden. Als nächster Schritt werden die TTP-Nachrichten definiert, die über den

Datenbus ausgetauscht werden. Dabei sollte jede Kommunikation zwischen Subsystemen über TTP-Nachrichten erfolgen. Dies gilt auch für den Fall, dass Subsysteme möglicherweise auf der selben Host-CPU laufen, um wohldefinierte Module und die sich daraus ergebende Flexibilität sicherzustellen.

Zur Vervollständigung des Cluster-Designs definiert der Anwender die verwendete Architektur und konfiguriert das Kommunikationssystem (z.B.: TDMA-Rundenzeit, Übertragungsrate, Art des Kommunikationscontrollers). Ebenso werden die einzelnen Subsysteme den Host-CPU's zugeordnet und bestimmte, sicherheitsrelevante Teilsysteme repliziert.

Per Knopfdruck exportiert TTP-Matlink die Designdaten nach ‚TTP-Plan‘, dem Cluster Design Tool für TTP-basierte Systeme. TTP-Plan errechnet den TDMA-Kommunikationszeitplan und speichert ihn in einer MEDL (Message Descriptor List) ab. Letztere wird bei der Implementierung in die Kommunikationscontroller geladen und beinhaltet die komplette Konfiguration des Kommunikationsablaufs. Beim anschließenden Knotendesign werden die Anwendungsalgorithmen der Subsysteme (Abb. 5) in einzelne Tasks gegliedert und diese spezifiziert. Lokale Botschaften dienen dem Informationsaustausch zwischen Tasks desselben Subsystems. Die Task-Designdaten können per Knopfdruck mit ‚TTP-Build‘, dem TTP-Knoten Design Tool, ausgetauscht werden. TTP-Build errechnet das Timing der Tasks für die jeweiligen Knoten und generiert die Fehlertoleranz-Schicht (FT-COM Layer).

Nachdem die kalkulierten Daten von TTP-Plan und TTP-Build in das Simulink-Modell importiert wurden, ist das Design des verteilten Systems abgeschlossen.

Simulation

Mit TTP-Matlink kann das verteilte System zusammen mit dem vorher konzipierten Kommunikationsverhalten simuliert werden. Dabei werden insbesondere das zeitliche Verhalten der Kommunikation (TDMA) und der Tasks berücksichtigt.

Durch die Simulation kann sowohl der Einfluss der Kommunikation auf das Steuer- und Regel-

verhalten überprüft und optimiert, als auch Parameter des Kommunikationsprotokolls (TDMA-Rundenzeit, Periode von Botschaften, Bandbreite) abgestimmt werden. Die Simulation einzelner Tasks erfolgt generationentreu, wodurch eine exakte zeitliche Beziehung zwischen Berechnung und Bereitstellung von Busnachrichten erreicht wird.

Der Anwender kann den Entwurf seiner Sicherheitsstrategie überprüfen, indem er einzelne Nachrichten ausfallen lässt und so den Einfluss auf das gesamte Systemverhalten beobachtet. Das Ergebnis kann in eine geänderte Redundanzstrategie zurückfließen. Veränderungen im Kommunikationsverhalten durch verschiedene Redundanz-Varianten können ebenso simuliert werden.

Die Simulation des Gesamtsystems bietet nicht nur unter dem Gesichtspunkt der Integration ein wichtiges Werkzeug zur Verifikation der Systemanforderungen sondern ermöglicht auch dem Zulieferer seine entwickelten Komponenten optimal in das Gesamtgefüge einzupassen. Hierbei kommt die Eigenschaft der Zusammensetzbarkeit der zeitgesteuerten Architektur zum Tragen.

Für die Simulation müssen algorithmische Details nicht weitergegeben werden, sondern können durch die ‚Wrapping‘-Mechanismen mit Hilfe von generierten S-Funktionen verborgen bleiben.

Automatische Code-Generierung

Nach erfolgreichem Systemdesign kann der Anwender mittels TTP-Matlink und dem ‚Real-Time Workshop‘ Embedded Coder den entsprechenden Code generieren, der auf ‚TTP-OS‘, das TTP-Echtzeitbetriebssystem, und die Fehlertoleranz-Schicht (FT-COM) abgestimmt ist. Der Code wird für die jeweilige Zielplattform aufbereitet und fertig zum kompilieren bzw. linken abgelegt.

