
MATLAB EXPO 2017

AGCO GmbH

Dipl.-Ing. (FH) Herbert Schindele



Aufbau einer modularen Simulink Real-Time HIL-Lösung

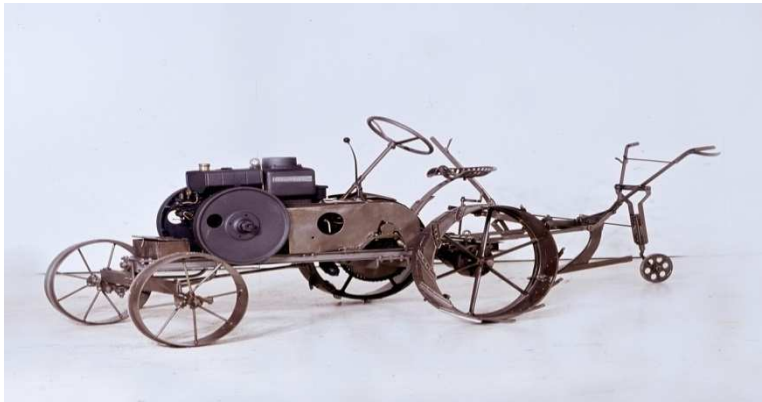
Integration in eine herstellerspezifische Architektur



Agenda

- **Wandel von Traktoren: Vom Dieselross zum Vario**
- **Neue Anforderungen an Softwaretests**
- **Aufbau eines Hardware-in-the-loop Systems**
- **Testlaufautomatisierung**
- **Zusammenfassung**

Wandel von Traktoren

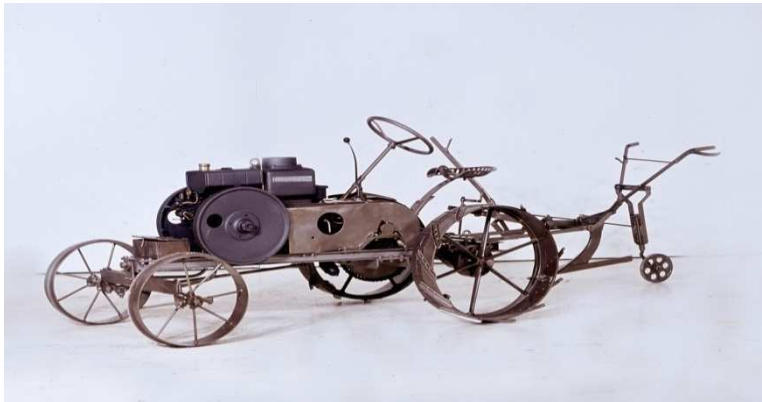


1930: Fendt Dieselross



2017: Fendt Vario 1050

Wandel von Traktoren - Fahrerarbeitsplatz



1930: Fendt Dieselross



2017: Fendt Vario

Challenges

- Legacy Code
- Maintenance von Fahrzeugen im Markt
- Lange Produktlaufzeiten von Elektronik-Komponenten
- Vielzahl von Varianten
- Kurze Time-to-Market
- Begrenzte Ressourcen

Wie kann ein effizienter Test aussehen?

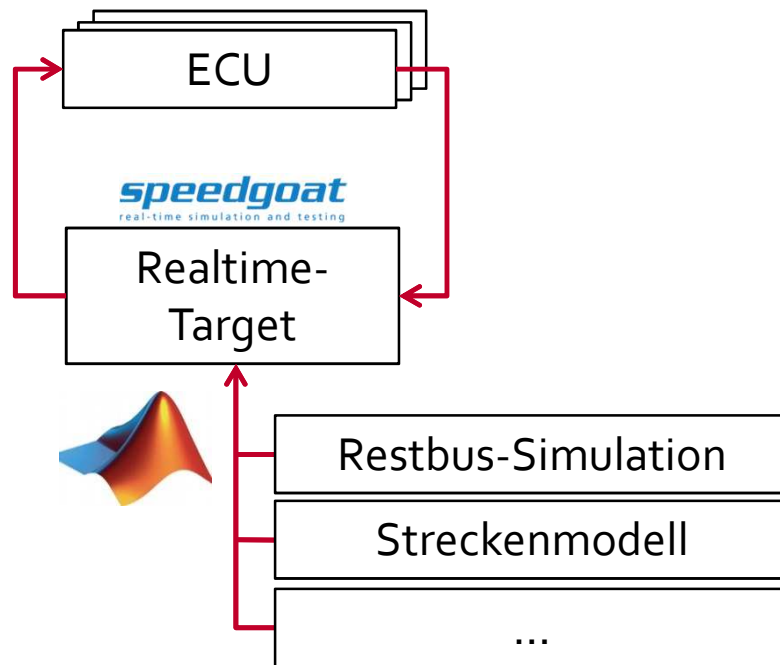


Hardware-in-the-loop



- Speedgoat performance real-time target machine
- Simulink-Modell kompiliert für Simulink-Realtime

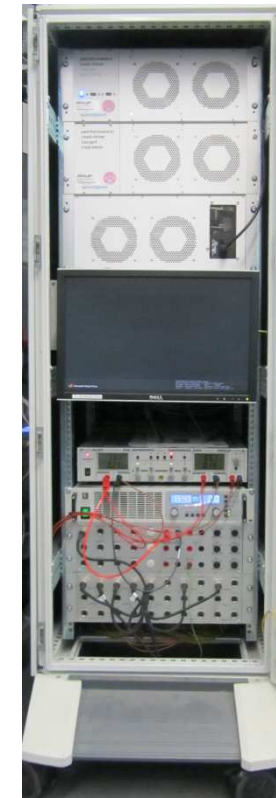
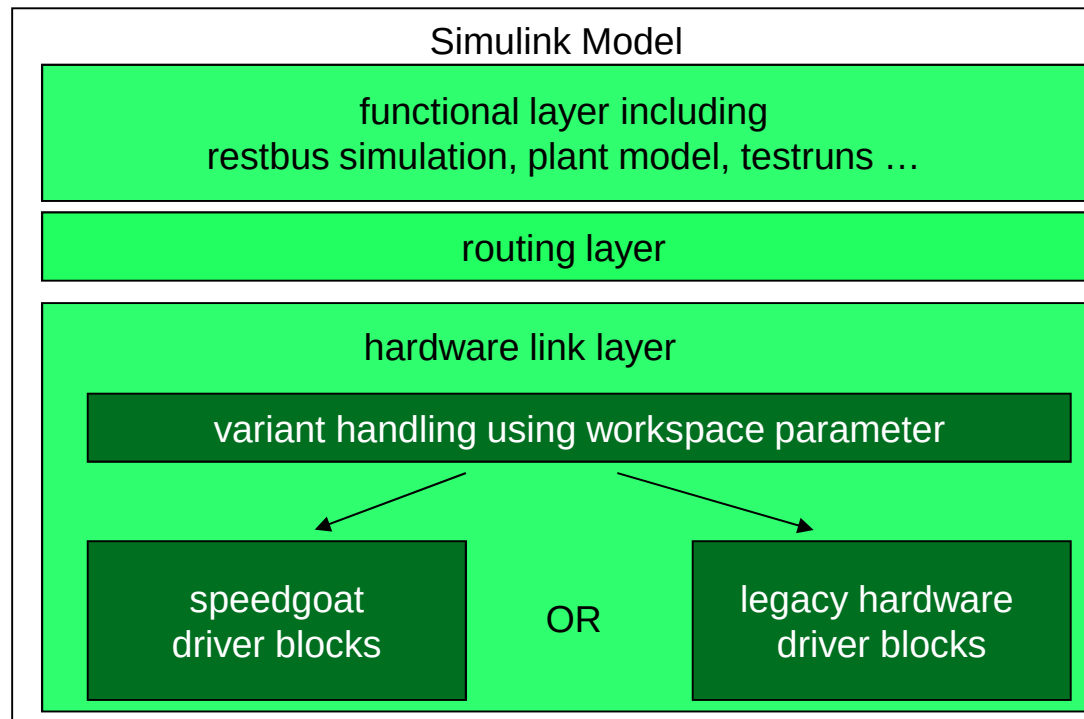
Hardware-in-the-loop



- Test der Software bevor ein Fahrzeug zur Verfügung steht
- Vielzahl von Varianten
→ über Variantensteuerung möglich



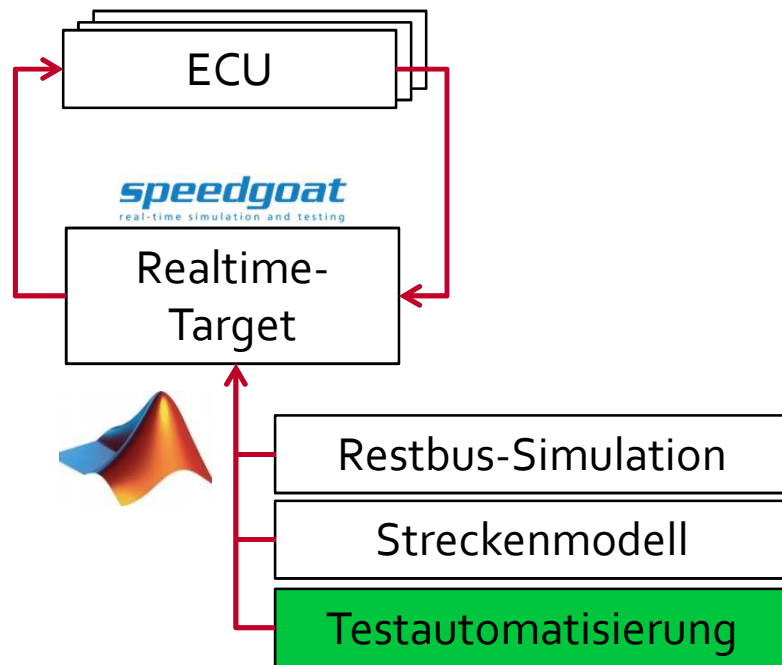
Echtzeitmodell - Struktur



Echtzeitmodell - Struktur

- Modell bildet verschiedene Typen/Varianten über Parametersätze ab
- Hardwareunabhängige Modellierung der funktionalen Modellebenen
- Restbusmodell, auch für proprietäre Protokolle
- Nutzung aller Rechnerkerne des Realtime-Targets

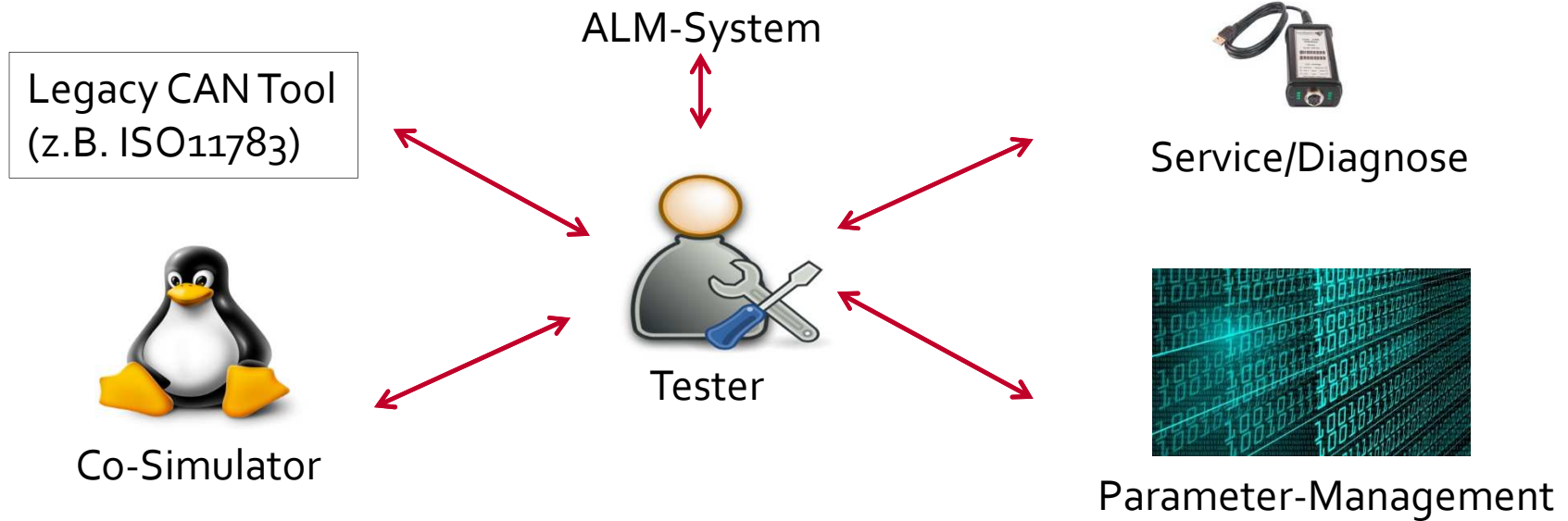
Hardware-in-the-loop und Testautomatisierung



- Test der Software bevor ein Fahrzeug zur Verfügung steht
- Vielzahl von Varianten
→ über Variantensteuerung möglich
- Begrenzte Ressourcen
- Kurze Time-to-Market

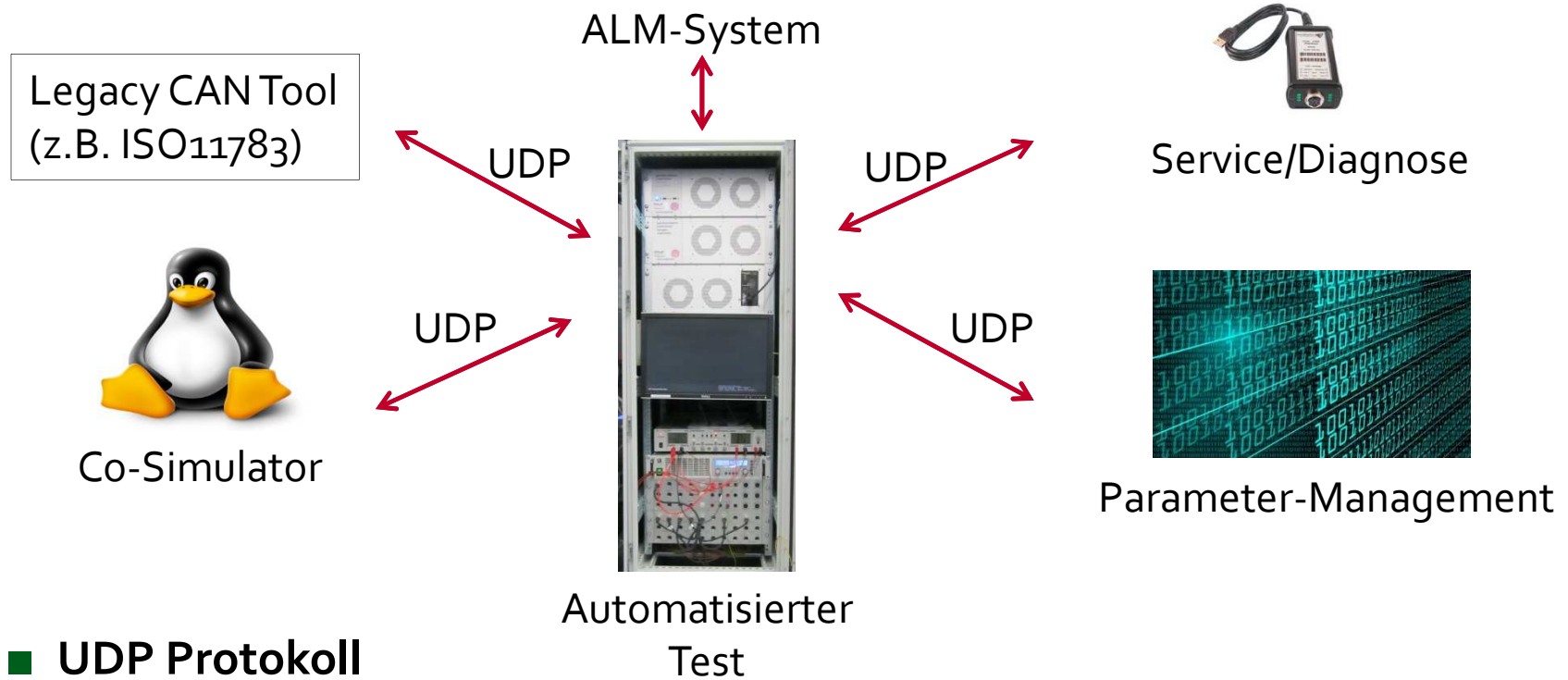


Schnittstellen



- Realtimefähige Schnittstelle
- Plattformunabhängig
- Performante, kostengünstige Hardware

Schnittstellen



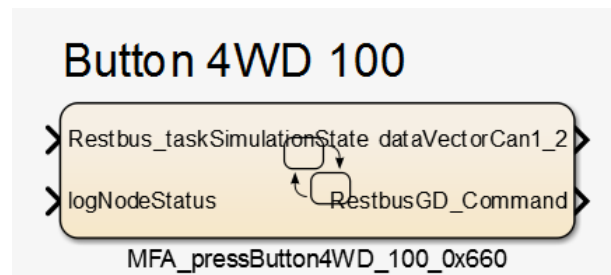
- **UDP Protokoll**
- **Eigene Protokolldefinition**

Testlaufentwicklung

- **Testsequenzen in Stateflow**
 - Einfache, übersichtliche Implementierung
 - Volle Integration in Simulink
- **Testlauf ist Teil des Modelles und wird in Echtzeit ausgeführt**
- **Datalogging erfolgt auf dem Target**

Library-Blocks

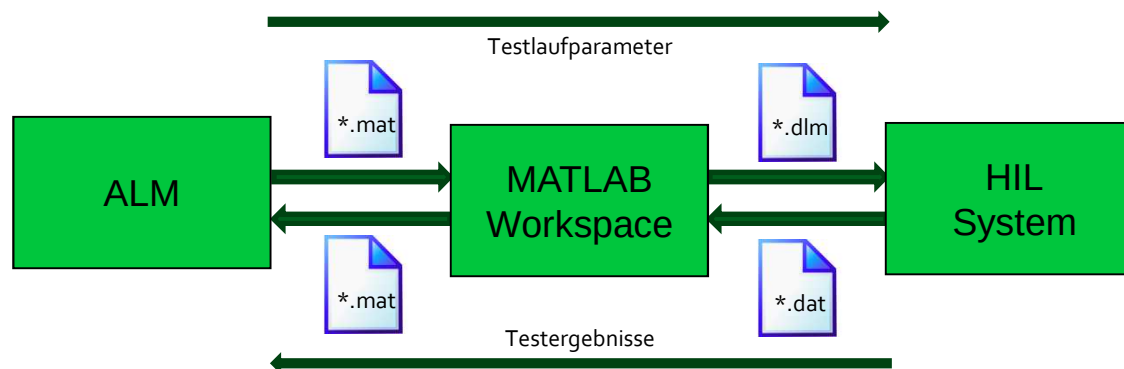
- Einfaches Handling von Standardaktionen
- Wiederverwendbare Funktionsmodule
- Atomic Subcharts



- Einfache Integration in Testlauf
- Abstraktion und individuelle Parametrierung möglich

Parametrierbare Testsequenzen

- Testsequenzen werden im ALM definiert
- Einlesen und Konvertierung in Workspace-Parameter
- Test von einfachen Testsequenzen, inklusive Checks und Auswertungen sind effizient möglich
- Rückportierung der Testergebnisse ins ALM-System



Zusammenfassung

- **Frühzeitiger Softwaretest**
- **Effektive Regressionstests**
- **Automatisierte Tests 24/7**
- **Tests von Varianten mit geringem Aufwand darstellbar**
- **Reduzierung von kostenintensiven
Prototypenfahrzeugen**