Faster development of automotive control software with Modelica and FMI

Emmanuel Chrisofakis
Daimler AG

Andreas Junghanns
QTronic GmbH

September 19, 2013
Design of Complex Dynamic Systems: Modelica & FMI
Dassault Systemes Campus, Paris
Faster development of automotive control software with Modelica and FMI

History of Modelica at Daimler transmission development

Motivation: Simulation for controls development

Application examples
- controls development: 7G-Tronic, 7G-DCT, ...
- measurement and calibration
- testing TCUs on PCs
- closed-loop prediction of aging and wearing

Conclusion
History of Modelica at Daimler transmissions

1998  Start to use SiL for transmission controls development
- Software in the Loop: inhouse SiL tool 'backbone'
- run fixed-point control software of the TCU on PC
- hand coded vehicle and transmission models (C code)
- application: 7G-Tronic transmission

2002  Successful evaluation of Dymola/Modelica
- port all transmission models to Modelica
- applications: 7G-Tronic family, 7G-DCT

2006  Introduction of commercial SiL tool Silver (QTronic)
- about 120 installations at Daimler today
- supports model exchange with suppliers

2011  Introduction of FMI to import Modelica models into SiL
- today: all Modelica models distributed via FMI
Motivation

- Size of control software doubles every few years
- Budget for development and test does not

Need to work in parallel threads across teams and organizations
- move controls development and test to PC
- based on simulation of vehicle and control software
Development of Control Software

Development with
- C/C++, Simulink, ...

Simulation tools
Modelica, FMU, Simulink, ...

HiL Test
Road Test

SiL Test

Virtual Integration

Development with
- C/C++, Simulink, ...

Simulation tools
Modelica, FMU, Simulink, ...

HiL Test
Road Test

SiL Test

Virtual Integration

Development with
- C/C++, Simulink, ...

Simulation tools
Modelica, FMU, Simulink, ...

HiL Test
Road Test

SiL Test

Virtual Integration

Development with
- C/C++, Simulink, ...

Simulation tools
Modelica, FMU, Simulink, ...

HiL Test
Road Test

SiL Test

Virtual Integration
Dymola plant model

**top level model of car and transmission**

- submodel of torque converter and lock-up clutch
- submodel of gearbox mechanics
- submodel of gearbox hydraulics

all models developed with own Modelica libraries on the basis of standard Modelica!
AMG SPEEDSHIFT MCT 7-speed sports transmission
7G-DCT Dual Clutch Transmission of Mercedes-Benz
7G-DCT Dual Clutch Transmission for AMG SLS
- vehicle and transmission model: Dymola/Modelica
- virtual TCU with Silver, test with TestWeaver
- every main software release: virtual endurance testing
- every sw release: 24h test run in parallel on several PCs
- thousands of gearshifts generated and analyzed

publications:
ATZelektronik, 6/2009
Automated test of the AMG DCT Speedshift control software, 9th International CTI Symposium Innovative Automotive Transmissions, Berlin, 2010
Virtual endurance testing

**Simulation of Mercedes test courses with following goals:**

- safeguarding of drivetrain components
- calculation of load collectives for gearbox and drivetrain
- monitoring of functional code changes (impact on gearbox and drivetrain)

**shifting analysis**

<table>
<thead>
<tr>
<th>SDL</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>IST</td>
<td>100</td>
<td>200</td>
<td>300</td>
<td>400</td>
<td>500</td>
<td>600</td>
<td>700</td>
<td>800</td>
<td>900</td>
</tr>
</tbody>
</table>

**thermal analysis of friction elements**

**gear dependent load collectives**
Test of control software

oscillation of target gear
- found by TestWeaver
- replay in Silver

improved control software
- run regression test
- problem solved

source: Schaich, Breitinger, Tatar,
Automated test of the AMG DCT
Speedshift control software.
9th International CTI Symposium
Innovative Automotive Transmissions,
Conclusion

Benefits of SiL for automotive software development

- **Fast feedback**
  - on the laptop of the developer

- Comfortable functions for **debugging and test**
  - fault simulation, pre-calibration, adaptation, etc.

- **Concurrent engineering**
  - developers, teams, OEM and suppliers

- **Sharing of results without sharing IP**
  - exchange of compiled models via FMI

- Executing others contributions without their tools
  - no extra complexity
  - no extra tool licenses