Tool Chain from Physical Modeling to ECU Software

Outline

- Relevance of ECU Software for Bosch
- Motivation of Physical Models on Embedded Targets (ECU)
- Challenges of ECU Software Development
- Solution Approach: Assisted Transformation of Physical Models to ECU Software
- Tool Chain Prototype
- Application Example
- Summary
- Outlook
Tool Chain from Physical Modeling to ECU Software

Relevance of ECU Software for Bosch

- Modern cars have more than 100 ECUs: Electronic Stability Program, Electronic Engine Control,…

- Complexity of software increases, with advanced control functions.

- Specialized hardware: µController with specialized core.
Introduction: AUTOSAR

- Layer Concept
  - HW Abstraction (BSW)
  - Runtime Environment (RTE)
  - Application Layer (ASW)

- Benefits
  - Standardized interfaces
  - Standardized structure of software modules.
  - Standardized hardware independent sharing of application software.

MOOPROD Workshop Feb. 7./8. 2017, Linköping, SE | 1/4/2017
© Robert Bosch GmbH 2016. All rights reserved, also regarding any disposal, exploitation, reproduction, editing, distribution, as well as in the event of applications for industrial property rights.
Tool Chain from Physical Modeling to ECU Software

Motivation of Physical Models on Embedded Targets (ECU)

- Online Physics-based Models on ECU, a key technology for:
  - Virtual Sensors, Observers,
  - Model-Based Diagnosis,
  - Feed Forward controllers,
  - Model Predictive Control.

Virtual Sensor (Boost Pressure)

Tool Chain from Physical Modeling to ECU Software
Motivation of Physical Models on Embedded Targets (ECU)

- Online Physics-based Models on ECU, a key technology for:
  - Virtual Sensors, Observers,
  - **Model-Based Diagnosis,**
  - Feed Forward controllers,
  - Model Predictive Control.

```
Motivation of Physical Models on Embedded Targets (ECU)

- Online Physics-based Models on ECU, a key technology for:
  - Virtual Sensors, Observers,
  - **Model-Based Diagnosis,**
  - Feed Forward controllers,
  - Model Predictive Control.
```
Tool Chain from Physical Modeling to ECU Software

Motivation of Physical Models on Embedded Targets (ECU)

- Online Physics-based Models on ECU, a key technology for:
  - Virtual Sensors, Observers,
  - Model-Based Diagnosis,
  - Feed Forward controllers,
  - Model Predictive Control.
Tool Chain from Physical Modeling to ECU Software
Motivation of Physical Models on Embedded Targets (ECU)

- Online Physics-based Models on ECU, a key technology for:
  - Virtual Sensors, Observers,
  - Model-Based Diagnosis,
  - Feed Forward controllers,
  - Model Predictive Control.
Tool Chain from Physical Modeling to ECU Software

Challenges of ECU Software Development

Simulation Engineer
- System Requirements
- Physical Modeling, Model Validation, Design Optimization
- Object oriented Physical Model Library
- Validated Process Model

Control Engineer
- Control Design, Discretization, Verification & Validation
- Simplified Plant Model
- Embedded Subset
- Controller Model

Function Developer
- ECU Compliant Software Implementation
- ECU Code
- Maps
- SW Architecture Base Software

Calibration Engineer
- Test Cases
- Measurements
- Calibration, Parameter Identification, Test
- ECU Software
- ECU Software

© Robert Bosch GmbH 2016. All rights reserved, also regarding any disposal, exploitation, reproduction, editing, distribution, as well as in the event of applications for industrial property rights.
Tool Chain from Physical Modeling to ECU Software
Assisted Transformation of Physical Models to ECU SW

Simulation Engineer
- System Requirements
- Physical Model Library
- Object oriented
- Validated Process Model
- Physical Modeling, Model Validation, Design Optimization

Control Engineer
- Simplified Plant Model
- Control Design, Discretization, Validation&Verification
- Embedded Subset
- Block Library
- Controller Model

Function Developer
- Generated ECU Code
- SW Architecture Base Software
- ECU Compliant Software Implementation
- Maps
- Calibration, Parameter Identification, Test

Calibration Engineer
- Test Cases
- Measurements
- ECU Software
- ECU Software with physics-based Functions

© Robert Bosch GmbH 2016. All rights reserved, also regarding any disposal, exploitation, reproduction, editing, distribution, as well as in the event of applications for industrial property rights.
Tool Chain from Physical Modeling to ECU Software

Solution Approach

- Challenge
  - Usage of object-oriented physical modeling
  - Leverage rich multi-domain libraries
  - Make physical models accessible to SW developers
  - Standardized model exchange
  - Generate embedded compliant code
  - Integrate physical models in ECU software architecture
  - Execute physical models on realtime targets
  - Enable cost effective realization.

- Solution Approach
  - Modelica™ language
  - MSL and other commercial libraries
  - Constraint Graph Analysis
  - FMI for model exchange
  - Code generation, MISRA compliance check
  - FMI AUTOSAR wrapper
  - AUTOSAR SW integration and deployment
  - Integrated tool chain from physical models to ECU software.
Application Example

- Object-oriented physical modeling in Modelica™
  - Definition of partial models for the building blocks the control circuit.
  - Definition of a generic control circuit structure using replaceable partial models.
  - Implementation of a variety of concrete models by extending the partial models.
    - PI Feed Back Controller
    - Feed Forward Controller
Tool Chain from Physical Modeling to ECU Software

Application Example

- Object-oriented physical modeling in Modelica™
  - Definition of the physical behavior using acausal physical ports.
  - Reuse of standard components from Modelica libraries.
  - Parameterization using physical units
Tool Chain from Physical Modeling to ECU Software
Application Example

- Simulation of the dynamic behavior
  - Straight forward realization of feed forward controller by inverting the physical plant model.
  - Comparison of simulation shows much improved following behavior of FF-Controller vs. PI-Controller.
Tool Chain from Physical Modeling to ECU Software

Application Example

- Simulation of the dynamic behavior
  - But, after applying a Step Command instead of Sine the index reduction fails.
  - The system can neither be compiled nor simulated.
Tool Chain from Physical Modeling to ECU Software

Application Example

- Transfer of DAE from Modelica™ to ASCET-CONGRA™
  - Dump of the symbolic equations (DAE) of the inverse plant model as txt file from OpenModelica using compiler flag +d=dumdaelow, optdaedump or dumpindxdae.
Tool Chain from Physical Modeling to ECU Software
Application Example

- Constraint Graph Analysis in ETAS ASCET-CONGRA™
  - Parse and convert the DAE dump txt file and import into ETAS ASCET-CONGRA.
  - Visualization of the symbolic equations and variables as constraint graph.
Application Example

- Constraint Graph Analysis in ETAS ASCET-CONGRA™
  - Hiding some of the parameters makes the relevant structure more visible.
  - By deriving a directed graph “Flow” The over-constraint (red) and under-constraint (blue) substructures become obvious.
  - The higher index problem is apparent due to the fact that the red substructure is determined by the Index0 variables I1_phi, I2_phi, I3_phi, while the blue substructure depends on the index2 variables (I1_a, I2_a, I3_a).
Application Example

- Constraint Graph Analysis in ETAS ASCET-Congra™
  - The index is reduced by adding the time derivatives of the over-constraint relations R09 and R04.
Application Example

- Constraint Graph Analysis in ETAS ASCET-CONGRA™
  - After defining the known inputs and outputs, states can be selected explicitly.
  - The directed graph displays the reached and not reached branches.
  - States can be added until the required output can be calculated.
Application Example

- **Constraint Graph Analysis in ETAS ASCET-CONGRA™**
  - Finally it can be shown that with:
    - $y$ and $u_{Sensor\_dot}$ as input and
    - only two states $I3\_w$, $SD\_phi\_rel$ the
    - required output $u$ can be calculated.

- It remains an algebraic loop (orange), which can be passed to the symbolic engine to see whether this loop can be analytically solved or will have to be broken at one or multiple edges to be solved iteratively or by using a delay.
Tool Chain from Physical Modeling to ECU Software

Application Example

- Constraint Graph Analysis in ETAS ASCET-CONGRA™
  - In this case the algebraic loop can be solved analytically.
  - The computation sequence is determined.
  - The corresponding FMU is generated in the background.
Tool Chain from Physical Modeling to ECU Software

Application Example

- MiL Simulation of FF-Controller
  - Import of the FMU into the system simulation.
  - Replacing the original controller block with the FMU.
  - Verification of the results.
**Tool Chain from Physical Modeling to ECU Software**

**Application Example**

- **FMU2eFMU Semi-automated Prototype**
  - Use only ECU compliant types.
  - Replace ECU non-compliant code.

- **eFMU2AUTOSAR Prototype**
  - Generate AUTOSAR model description (ARXML).
  - Generate AUTOSAR function calls.

- Integrate FMU-SWC in AUTOSAR ASW
Tool Chain from Physical Modeling to ECU Software
Integration of FMU in AUTOSAR

- Architecture
  - FMUs as AUTOSAR SW-Cs in ASW.
  - Interface and service communication over Ports.
  - Solver library in BSW.
  - Computation algorithm executes FMUs using the events provided by Real-Time OS.

Jonathan Neudorfer, Siva Sankar Armugham, Mathews Peter, Naresh Mandipalli, Isidro Corral, Christian Bertsch, Karthikeyan Ramachandran
Tool Chain from Physical Modeling to ECU Software
Prototype Integration of FMU in AUTOSAR

- Interface mapping of FMI to AUTOSAR.
- FMI interface functions are called via wrapper function using AUTOSAR client server communication.
- Data Exchange over RTE. AUTOSAR RTE read/write routines with FMI get/setReal functions.
Tool Chain from Physical Modeling to ECU Software

Summary

State of the Art

- Plant modeling and control design is powerful in object oriented physical M&S tools for offline simulation.
- No code generators for ECU → Code of offline simulation cannot easily be used on ECU.

Solution Approach

- Enable function developers to analyze physical models in software related fashion.
- Use FMI standard of physical model on all targets.
- Use FMI standard with AUTOSAR for embedded targets

Benefits

- Reduce complexity of ECU software using physical models.
- Reduce development effort for advanced ECU functions.
Tool Chain from Physical Modeling to ECU Software
Outlook: Future Tool Chain based on eFMI → PFP EMPHYSIS

Simulation Engineer  Control Engineer  Function Developer  Calibration Engineer

Physical Modeling Tools

OMEdit

OpenModelica

Simulation

Dymola

AMESim

MapleSim

…

efmu XML

AST eFMU

CONGRA

FMU to SWC

AUTOSAR Wrapper

FMU SWC

AUTOSAR

SWCs

ASCET

Bosch ECU

efmu C

Source Code eFMU

efmu C

Object Code eFMU

efmu BIN

Object Code FMU

RTPC, e.g.

ETAS RTPC

© Robert Bosch GmbH 2016. All rights reserved, also regarding any disposal, exploitation, reproduction, editing, distribution, as well as in the event of applications for industrial property rights.
Thanks to all contributors.

Jonathan Neudorfer¹, Siva Sankar Armugham¹
Mathews Peter¹, Naresh Mandipalli¹
Oliver Lenord², Christian Bertsch²
Markus Behle³, Arndt-Michael Meyer³

Thank you for your attention.

¹) Robert Bosch Engineering and Business Solutions Private Limited, 560 095 Bangalore, India
²) Robert Bosch GmbH, Corporate Sector Research and Advance Engineering, 71272 Renningen, Germany
³) ETAS GmbH, Borsigstraße 14, 70469 Stuttgart, Germany
Alternative Workflow

Simulation Engineer → Control Engineer → Function Developer → Calibration Engineer

Physical Modeling Tools
- OMEdit
- OpenModelica
- SimulationX
- Dymola
- AMESim
- MapleSim

Source Code FMU → FMU to eFMU → FMU to SWC → FMU SWC

Computer

Bosch ECU

physical modeling tools

Fmu to swc

Computer

Dymola

Computer

Dymola