Model Based ECU Development
An Integrated MiL, SiL, HiL Approach
Agenda

- Model-Based ECU Development – Today
  - Model-in-the-Loop (MiL) Development
  - Software-in-the-Loop (SiL) Development
  - Hardware-in-the-Loop (HiL) Development

- Challenges of the Traditional Process

- Requirements of an Integrated MiL/SiL/HiL Approach

- An ETAS Solutions

- Conclusions
Model-Based ECU Development - Today
Process Steps

- **Goal:**
  - Rapid development, test and validation of new control strategies

- **Methodologies:**
  - Modeling
  - Model-in-the-loop
  - Software-in-the-loop
  - Hardware-in-the-loop
Overall System Architecture

Driver

Subsystem (e.g. Engine)
Sensor
Actuator

Subsystem (e.g. Transmission)
Sensor
Actuator

Subsystem (e.g. Chassis)
Sensor
Actuator

Vehicle Network (e.g. CAN)

Environment (Road, Wind, etc.)
Migration to a Virtual Environment – 1/3

- **Environment** (Road, Wind, etc.)
- **Vehicle**
- **Driver**
- **Subsystem** (e.g. Engine)
- **Subsystem** (e.g. Transmission)
- **Subsystem** (e.g. Chassis)
- **Virtual Sensor/Actuator Signal Bus**
- **Vehicle Network (e.g. CAN)**
Migration to a Virtual Environment – 2/3
Migration to a Virtual Environment – 3/3
Objectives:
- Functional validation and calibration of ECU sub-system models
- Evaluate interactions between ECU sub-systems
- Refinement of plant models

Pre-requisites:
- Plant models with adequate fidelity
- All signal interface definitions (e.g. sensors)
- Test scenarios for validation (e.g. test vectors, test stimuli)
Software-in-the-Loop (SiL) Overview

- Objectives:
  - Functional validation of ECU software architecture
    → e.g. CAN bus configuration and load
  - Verification of ECU software implementation against model

- Pre-requisites:
  - ECU software modules (C-code)
  - ECU communication architecture (e.g. CAN network)
  - Test scenarios from MiL (e.g. test vectors, test stimuli)
**Objectives:**
- Validation of ECU software in real-time with actual sensor and actuator signals
- Better calibration values

**Prerequisites:**
- Plant model should run in real-time
- Emulation of all electrical interfaces of the ECU
- All control models converted to ECU code
Challenges of the Traditional Process - 1/3

• Increasing complexity of plant models
  • Controls development needs domain specific models
    → slow to execute on the PC (e.g. transmission hydraulics, fuel-cell stacks)

• Function oriented ECU development
  • Models are built in different environments or different versions of the same environment
  • Model data and interfaces are difficult to merge

• Distribution of electronic features over multiple ECUs
  • Higher system integration and validation complexity
Challenges of the Traditional Process - 2/3

• System and sub-system testing
  • Test cases for MiL cannot be reused in HiL
    ➔ Due to tool and configuration inconsistencies
  • Tests scripts have to be adapted when HiL hardware is switched

• Handling of C-code for SiL
  • Large number of legacy C-code modules are needed
    ➔ Current solutions are not modular and scalable
  • Limited access to C-code parameters for measurement and calibration
  • Production standards (ASAP, MSR, OSEK etc.) are not compatible
Challenges of the Traditional Process - 3/3

- Control over model execution in MiL and SiL
  - Execution may not adapt to PC compute power
  - Speed-up or slow-down not possible

- Relative timing behavior of control modules in MiL and SiL
  - No provision to schedule control modules to run according to production OS requirements (e.g. OSEK)

- User Interfaces and Data Exchange between MiL/SiL and HiL
  - Different tools require creation and maintenance of separate calibration and measurement profiles
  - Often, datasets cannot be easily exchanged
An Integrated MiL/SiL/HiL Approach
Requirements - 1/4

Plant Models:

- Flexibility in coupling with plant models of different origins, platforms and interfaces
- Support for both homogeneous (single executable) and heterogeneous (co-simulation) modes on the PC
- Ability to control the run-time behavior on the PC
  - speed-up, slow-down as needed (available compute power)
- Re-use of the same plant model across MiL, SiL and HiL
- Re-use of stimuli sets (e.g. driver inputs, closed-loop drive profiles etc.) from MiL to SiL to HiL
Control System Models:

- Maintain the modularity of function oriented ECU development  
  → i.e. integrate models from different modeling tools, versions and organizations into one system
- Ease of connecting control model signals with plant model signals  
  → i.e. creating the Virtual Signal Bus
- Support for upcoming standards  
  → e.g. AUTOSAR interface description files
- Ability to schedule various sub-systems to run under timer or event driven tasks on the PC
- Fast turnaround times for MiL and SiL  
  → i.e. incrementally build and compile model changes
C-code integration:

- Ability to integrate C-code with models at different levels
  - e.g. source code, object code, .dll files, .lib files
- Measure and calibrate C-code parameters in SiL
  - e.g. re-use available ASAM-MCD-2MC description files
- Validate models and C-code while utilizing “platform services” on the desktop PC
  - e.g., RTOS settings, timer and event based tasks etc.
- Maintain modular C-code architecture of the ECU in SiL
An Integrated MiL/SiL/HiL Approach
Requirements - 4/4

Process:

• Identical user interface for control of plant and ECU software models
  → especially important as one moves from PC (MiL/SiL) to HiL

• Ability to re-use MiL/SiL artifacts in HiL
  • Test-scripts
  • ECU model calibrations
  • Plant models
  • Data sets (e.g. stimuli, test data)
An Integrated MiL/SiL/HiL Approach
ETAS Solution

INTECRIO
- A comprehensive PC based integration platform for MiL and SiL

LABCAR
- A HiL system with scalable hardware and open software architecture

INCA
- ECU measurement and calibration software
An Integrated MiL/SiL/HiL Approach

ETAS Solution

INTECRIO as a homogeneous MiL/SiL integration platform
An Integrated MiL/SiL/HiL Approach
ETAS Solution

INTECRIO as a heterogeneous MiL/SiL integration platform
An Integrated MiL/SiL/HiL Approach
ETAS Solution

Integrating C-code in **INTECRIO** via INCODIO for MiL and SiL
An Integrated MiL/SiL/HiL Approach
ETAS Solution

Using INCA for Calibration and Measurement in MiL, SiL and HiL
An Integrated MiL/SiL/HiL Approach
ETAS Solution

Using **LABCAR-AUTOMATION** for Testing in MiL, SiL and HiL
An Integrated MiL/SiL/HiL Approach

Customer Success Stories

• A major US OEM
  • Major advance in calibration development:
    → 75% in MiL, 100% in HiL
  • Most software bugs removed before going into vehicle
  • 3x improvement in turnaround times over standard process
    → i.e., controls and plant model changes in MiL
  • 7x improvement in model execution times on the PC

• A major European OEM
  • 5x improvement in model execution times on the PC
  • Sharing experiments, data, stimuli between users
An Integrated MiL/SiL/HiL Approach

Conclusions

INTECRIO, LABCAR, and INCA
• A integrated tool suite for MiL, SiL and HiL development

Key Advantages:
• Reduce non value-added steps (e.g. data conversion, adaptation of test sequences, GUIs etc.)
• Re-use test scripts, stimuli, plant models and ECU data
• Reduce costs associated with the ECU development process
  → do more in the virtual environment
  → reduce dependence on fleet and dyno testing
Model Based ECU Development
An Integrated MiL, SiL, HiL Approach

Thank you for your attention!
Your questions are welcome.