Test Automation and ISO-26262 Preparation for New Bus Product Line with Fully Electrified Powertrain
MathWorks Automotive Conference 2019
Scope
Continental Products used by VDL

Next generation of public transportation busses
  - Fully electrified powertrain
  - gVCU implements high-level driveline controller functions

Off-The-Shelf HW products
  - generic Vehicle Control Unit
  - Body Controller
  - Multiplexer Nodes
  - Instrument Cluster

Toolchain Model Based Development System
  - Based on MATLAB Simulink
  - MBDS IDE and modeling framework
  - Compliant to ISO-26262
  - CI/CT support
gVCU - generic Vehicle Control Unit

System Overview

Features

› Central Driveline Controller
  › Torque Control, Brake Control, Cruise Control, …
› High performance µC for complex control algorithms
› HW and basic SW is ASIL-B ready
› Application SW is programmed by the customer using MBDS

8 April 2019
Dr. Sven Semmelrodt, CVAM © Continental AG
MATLAB/Simulink versus MBDS

Non-Automotive SW Design

Automotive SW Design

✓ Single button-click solution
✓ No MATLAB expert required
MBDS Environment
Composition & Deployment

Specific Extension(s):

Modular Base Resource Type

Generic Extension:

MBDS Core, Library and Help

Base Environment:

MATLAB and Simulink

› Modeling: Simulink and Stateflow
› Code generation: Embedded Coder
› MBDS graphical user interface used as IDE
› Quality assurance measures, ACG, build & download
› Block library for design and integration
› Target HW and customer specific extension, e.g. BSW, compiler, …
ISO-26262 Conformance
Tool Evaluation

› Hardware & Basic Software are ASIL-B ready

› **MBDS**: Evaluation of a software tool by analysis (ISO-26262, Part 8, Chapter 11.4.5)
MBDS Test Automation
3-Pillars of the Test Approach

› Configuration, execution & evaluation: Test Suite view in the MBDS IDE
› Implementation of test stimuli & expected values: Signal Definition Table
› Test frame: MBDS Model Template

MBDS Signal Definition Table

MBDS Test Suite View

MBDS Model Template
MBDS Test Automation

Test Implementation, Execution & Evaluation

Load test data

Test Data (*.xlsx)

Log stimuli, expected values & output signals

Test Outputs (*.mat)

Evaluate outputs

Test Management

Test Suite (*.xml)

Test Configuration

› One or multiple test levels
› One or multiple test cases
› With or without model coverage
› Manual inspection using Simulation Data Inspector
› Stimuli, expected values, outputs and configuration can be archived
MBDS Test Automation
Test Implementation using Signal Definition Table

- Test data editor for Stimuli & Expected Values: MS Excel
  - Our customers are MS Excel but no MATLAB experts
- Each test case is specified by a separate sheet in the SDT
  - Using defaults, a test case may contain signal subsets only
  - Simple test data specification using transition based time/value pairs
- Evaluation parameters (Shift & Tolerance) per expected value
- Test case & test step description
- Requirements for traceability
- Implementation of SDT by Signal Builder blocks (Q&D debugging)
- SDT can be automatically derived from existing models
MBDS Test Automation
Automated Test Evaluation & Test Report

- Test report is generated based on the test levels & test cases selected for a test session
- Logged test outputs are compared to the expected values (considering Tolerance & Shift)
- Test information overview with test result summary
- Test description & requirements
- Signal test results & deviation statistics are displayed
- Test results are tagged in HTML report for simple CT result extraction jobs
- Fully automated via API
MBDS Test Automation
Automated Test Evaluation & Test Report

› Test report is generated based on the test levels & test cases selected for a test session
› Logged test outputs are compared to the expected values (considering Tolerance & Shift)
› Test information overview with test result summary
› Test description & requirements
› Signal test results & deviation statistics are displayed
› Test results are tagged in HTML report for simple CT result extraction jobs
› Fully automated via API
› Signal plots for fast visual inspection
MBDS Test Automation
Automated Test Evaluation & Test Report

› Test report is generated based on the test levels & test cases selected for a test session
› Logged test outputs are compared to the expected values (considering Tolerance & Shift)
› Test information overview with test result summary
› Test description & requirements
› Signal test results & deviation statistics are displayed
› Test results are tagged in HTML report for simple CT result extraction jobs
› Fully automated via API
› Signal plots for fast visual inspection
› Requirements to test results matrix
Continuous Integration / Continuous Testing

Why?

› Continuous Integration & Testing
  › Improvement of the SW quality by continuously integration of the SW (agile development process)
  › Automated Activities
    › Static model analysis using automated model rule checks
    › Calculate model metrics (e.g. subsystem depth, cyclomatic complexity, …)
    › Automatically execute test sessions and summarize test results
    › Determine model test coverage
    › Automatically generate code, model documentation & run the build process
  › Running above listed activities of a complex SW project is a time consuming activity
    › A fully automated ‘nightly’ build & test feature is required
Continuous Integration / Continuous Testing

MBDS API

› MBDS API
  › All major MBDS features are available via simple API scripts
  › Applicable for single or all project items
  › All arguments are strongly checked on plausibility

› MBDS Remote Tool
  › Call MBDS API functions to automate MBDS features from batch jobs or OS command line.
  › MATLAB is started via COM-Interface with MBDS in silent mode

Features
- Project Item Handling
- Code & Document Generation
- Model Rule Checking
- Execute Test Suite
- Build & Download

COM Interface
Conclusions

› With the KIBES system platform Continental provides
  › Vehicle Control Hardware & Development Environment
    › ASIL B ready according ISO-26262
    › Fully automated quality assurance measures
    › Integration into CI/CT activities is supported

› VDL Activities
  › Currently quality assurance measures are applied on models implementing QM and ASIL rated features
  › Continuous integration & testing will be applied using the MBDS API and MBDS Remote
  › Statement VDL: We are working together on making the VDL software architecture future-proof with Continentals Model Based Development System
Thank you for your attention!