Toolchain Definition and Integration for ISO 26262-Compliant Development

July 1

MathWorks AUTOMOTIVE CONFERENCE 2020
Introduction

- MathWorks tools like Simulink and Stateflow are suitable for generating code for ISO 26262 QM to ASIL-D applications

- MATLAB has emerged for AD/ADAS algorithm prototyping
  - A natural language for matrices, image processing, deep learning
  - MATLAB source (text) is also seamless to integrate with Agile workflow tools

- Can we generate certifiable code from MATLAB?
Yes! Through Simulink Integration

- Called by the MATLAB Function block and/or Stateflow
  - Inlined MATLAB operators
  - External functions
  - Long list of language features that support code generation
  - And functions, including toolboxes like Sensor Fusion, Stats and Machine Learning, Automated Driving, Deep Learning

- MATLAB code generation is supported by our IEC Certification Kit and reference workflow
Best practice

- We can combine these and have the best of both worlds
  - Richness of the MATLAB language
  - Rigor of the Simulink family of verification tools

- “I’m a MATLAB user, is Simulink for me?”
  - If you need to provide **evidence of conformance**
  - To define **architecture** around MATLAB algorithms
Verification workflow

- Trace requirements ⇔ design ⇔ implementation ⇔ validation
- Meet design & implementation standards
- Show intended and no unintended functionality
  - Coverage is key evidence
MATLAB + Simulink ISO 26262 Workflow

- Our ISO verification activities now support this combined language
  - Requirements traceability
  - Design standards
  - Prove correct functionality
  - Prove absence of unintended functionality

This Photo by Unknown Author is licensed under CC BY-SA
Simulink Requirements supports authoring, importing/exporting, and linking requirements to **model elements**, test cases (Simulink Test)

+ Blocks, Charts, lines of MATLAB code
+ Requirements Traceability report for evidence
+ MATLAB source and user comments can be included as generated comments
Requirements
Traceability sample
Design and Code Standards

- Simulink Check has checks for good MATLAB style and improving code compliance
  - Enforcement of low complexity
  - Enforcement of comment density
  - Strong data typing between MATLAB and Simulink
  - Find logical operators with mixed data types
- Some MATLAB & Embedded Coder settings for MISRA-C
- MATLAB style guides are limited in scope (MAAB, NASA)
Demonstrate correct functionality

- Requirements-based test authoring, execution via Simulink Test
- Simulink Design Verifier (SLDV) property proving
- SLDV design error detection
- Back to back testing for model to code for Software-in-the-Loop (SIL), Processor-in-the-Loop (PIL)
Demonstrate no unintended functionality

+ Simulink Coverage to show completeness of test cases
  + Model coverage
  + Code coverage for SIL/PIL

+ SLDV can generate missing tests
Summary so far

- Customers are successfully using MATLAB in ISO 26262-compliant products today

- Our verification workflow and tools support MATLAB called by Simulink

- But… there are some gaps remaining
  - Potential issues with MISRA-C compliance of code generated from MATLAB
  - Achieving MATLAB vs C code coverage
  - Simplifying Simulink model reviews
Code standards compliance

- Practice is to
  - run model checks
  - generate code
  - analyze compliance

- Issues discovered?
  - document and proceed
  - rework the algorithm
  - rewrite a compliant function (toolboxes)

- Result is an allowed function list (language subset)
- Process gets more efficient over time

**Tools:**
- Simulink Check
- Polyspace Bug Finder
Code coverage

- MATLAB functions can be complex in C/C++

- One test case gets coverage in MATLAB, but more required to show no unintended functionality in the generated C

- Strategies include
  - Develop unit tests for feature/function
  - Implement a simpler replacement

```matlab
% 5. Compute Kalman Gain: 
W = P*W' + inv(K*P*W') * R;
```
Reviewing Simulink models

- Are you reviewing Simulink models?
  - 1-1 or 1-many at desk or in conference rooms?
  - Screen sharing apps?

- Modern workforces are often distributed and busy, making this a challenge

- Tools to manage the review process, such as Gerrit Code Review, are becoming a popular approach
Text-based differences + review comments
Gerrit Code Review

Gerrit implements a web-based review and approval workflow for git patch revisions.

Review comments are shared in the context of the source.

But, binary formats not supported (.slx)
Model reviews with built-in features

- Configure SCM with external diff tool for MATLAB files
  - E.g., "C:\Program Files\MATLAB\R2019a\bin\win64\mlDiff.exe" %LOCAL %PWD %REMOTE
  - Note this will reuse a running MATLAB not start a new instance
- Publish model comparison to MS Word format
- Annotate and share Word document with comments/replies
Extending this concept *into* Simulink

- Custom add-on to Simulink context menu
- Block badge indicates comment attached
- Publish to Gerrit when ready to share
Customers are successfully using Simulink **AND MATLAB** in ISO 26262-compliant products today

- There are some gaps remaining
  - Potential issues with MISRA-C compliance of code generated from MATLAB
  - Achieving MATLAB to C code coverage
  - Simplifying Simulink model reviews

**Best Practices Technical paper**
Do you use MATLAB code in your ISO 26262 components?

- Yes
- No
- No, but planning to

Please contact us with questions

lrosqvis@mathworks.com