MATLAB EXPO 2019

Toolchain Definition and Integration for ISO 26262-Compliant Development

Dave Hoadley
Introduction

- We’ve seen how 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

- How to generate certifiable code from MATLAB?
Yes! MATLAB and 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
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 $\leftrightarrow$ design $\leftrightarrow$ implementation $\leftrightarrow$ 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
Traceability

- 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 comparison reviews
Simulink core blocks and toolboxes

- Well-understood
- Up-front tools to help
- Few surprises
MATLAB and toolboxes

- Emerging usage
- Less up-front advice
- Iterative process today
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
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 EXPO 2019
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)
Extending this concept to Simulink

- Custom add-on to Simulink context menu
- Block badge indicates comment attached
- Publish to Gerrit when ready to share
Review a revised model

- Compare patch revisions in Simulink
- Attach review comments to either revision
Summary redux

▪ Customers are successfully using Simulink **AND MATLAB** in ISO 26262-compliant products today

▪ Our verification workflow and tools support MATLAB called by Simulink

▪ 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

▪ Contact me [dhoadley@mathworks.com](mailto:dhoadley@mathworks.com) and at the ISO 26262 table