A new approach to Model-Based Testing in Simulink®

Presented by: Sean Wyatt
## Content

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Need and Challenges for Function Developers</td>
</tr>
<tr>
<td>2</td>
<td>What is RT2?</td>
</tr>
<tr>
<td>3</td>
<td>Test Case Modeling&lt;br&gt;Test Case Execution&lt;br&gt;Test Case Assessment and Analysis&lt;br&gt;Test Report Generation</td>
</tr>
<tr>
<td>4</td>
<td>Benefits of RT2</td>
</tr>
</tbody>
</table>
## Content

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Need and Challenges for Function Developers</td>
</tr>
<tr>
<td>2</td>
<td>What is RT2?</td>
</tr>
</tbody>
</table>
| 3 | Test Case Modeling  
    Test Case Execution  
    Test Case Assessment and Analysis  
    Test Report Generation          |
| 4 | Benefits of RT2                                                |
A new approach to Model-Based Testing in Simulink®

The Need

- Software defects originate from all stages of the development process
  - (Model-based) function development
  - Software code generation
  - Software build and integration
  - Hardware development
  - ...

- Errors are often found far too late
  - High pressure and risk at the end of the engineering cycle

..... resulting in high cost of fixing errors


Barry Boehm, Software Engineering Economics
A new approach to Model-Based Testing in Simulink®
The Challenges for Function Developers
Challenges Summarized

- Tracking requirements coverage and changes to their dispositions
- Deciding how to stimulate the function
- Creating test variants of the stimulations in a systematic and repeatable manner
- Assessing the results
- Creating reports with rich details that support ISO 26262:6 and 8

ETAS’ solution is RT2
## Content

<table>
<thead>
<tr>
<th></th>
<th>Need and Challenges for Function Developers</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>What is RT2?</td>
</tr>
</tbody>
</table>
| 3 | Test Case Modeling                          
|   | Test Case Execution                         
|   | Test Case Assessment and Analysis           |
|   | Test Report Generation                      |
| 4 | Benefits of RT2                             |
ETAS RT2 is a tool for **designing, executing and assessing systematic tests** for:

- **Functional models** (Simulink or ASCET)

- **Software-in-the-loop** platforms (e.g. C-code .exe, ISOLAR-EVE)

- **Back-to-back** testing: automated comparison of test results between model and software test
A new approach to Model-Based Testing in Simulink®

RT2 Testing Approach: Model-based Testing

- Normally, test case design is done using scripts

- What’s the challenge of scripting?
  - Manage huge amounts of variants
  - Hard to overview the testing strategy and coverage
  - Programming skills required
  - High maintenance effort

- RT2 takes a different approach: test cases are described using models
  - Intuitive representation of complexity
  - Efficient management of variants
  - A “language” function developers understand
## Content

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Need and Challenges for Function Developers</td>
</tr>
<tr>
<td>2</td>
<td>What is RT2?</td>
</tr>
</tbody>
</table>
| 3 | Test Case Modeling  
Test Case Execution  
Test Case Assessment and Analysis  
Test Report Generation |
| 4 | Benefits of RT2 |
A new approach to Model-Based Testing in Simulink®
RT2 Testing Process

1. Test Case Design Model
2. Execution
3. Automated Test Assessment
4. Test Report

Simulink® Model

Instrumented Simulink® Model

Interface
Parameters

RT2 Testing Process
A new approach to Model-Based Testing in Simulink®

RT2 Testing Process

1. Test Modeling
2. Test Execution
3. Test Assessment
4. Test Reporting
Test case scenarios often consist of sequences of logical phases.

1. Start Vehicle
2. Accelerate until 50 km/h
3. Emergency brake with steering wheel to the left
4. Repeat steps 2 and 3 a total of 10 times
5. Stop the car
6. Engine off

State machine with textual annotations are descriptive and easy to understand
A new approach to Model-Based Testing in Simulink®

Test Modeling: Test State Definitions

- **Direct definition**

- **Test step list:** definition of a step sequence supported by a graphical editor

- **Time partitioned tests:** definition through a hierarchical state machine

Formulas:

\[ MIN_{\text{LIGHT\_OFF}} + 20 \times \sin(t/10) + 7.5 \times \text{random}() \]
Hierarchical and parallel states

Variants of a state

Variants of combinations of states

Test case 1

Test case 2

Accelerate: with half throttle

until speed: 120 km/h

Emergency brake: steering wheel to the right

Test Modeling: State Arrangements and Variations

A new approach to Model-Based Testing in Simulink®

1

Public | ETAS/COM-Ana | 2013-10-19 | © ETAS GmbH 2013. All rights reserved, also regarding any disposal, exploitation, reproduction, editing, distribution, as well as in the event of applications for industrial property rights.
A new approach to Model-Based Testing in Simulink®
RT2 Testing Process

1. Test Modeling
2. Test Execution
3. Test Assessment
4. Test Reporting
Test cases stimulate the test object by continuously affecting system quantities (inputs)

Test cases can react to system behavior by observing system quantities (outputs)

Test Object = Simulink, ASCET, C-code etc.
A new approach to Model-Based Testing in Simulink®

Test Execution (2/2)

Simulink® Model

Instrumented Simulink® Model

Test Case Design Model

Execution

Interface
Parameters
A new approach to Model-Based Testing in Simulink®
RT2 Testing Process

1. Test Modeling
2. Test Execution
3. Test Assessment
4. Test Reporting
A new approach to Model-Based Testing in Simulink®
Test Assessment – Time Interval Analysis (1/2)

- Assessments conducted after test data is collected

- When specific state is active: Analyses focused on a time interval when a specific state is active

Steering wheel to the left

- Duration of the emergency brake may not exceed 10 sec.
- Deceleration must be less than 1.5 m/s².
- Revs per minute must be less than 6000 min⁻¹.

Diagram:

1. Start engine
2. Accelerate with full throttle until speed 30 km/h
3. Emergency brake steering wheel to the left
4. Stop the car
5. Engine off
A new approach to Model-Based Testing in Simulink®
Test Assessment – Time Interval Analysis (1/2)

- **Absolute time**: A time interval specified by absolute time
  - \([t \geq 50s]\)

- **Time patterns**: Explicit time intervals specified
  - \([v_{\text{vehicle}}(t) \geq 100 \text{ kph}]\) time intervals with speed \(\geq 100\)
  - \([\text{foo}(t) = 1] [\text{foo}(t) = 2]\) time intervals with foo=1 followed by foo=2

- **Temporal regular expressions**: Time patterns as special cases
  - Assessments can be analyzed in multiple time intervals:
    e.g. *time intervals where vehicle speed drops below 50kph for less than 10s*
A new approach to Model-Based Testing in Simulink®
RT2 Testing Process

1. Test Modeling

2. Test Execution

3. Test Assessment

4. Test Reporting
A new approach to Model-Based Testing in Simulink®

Test Summary and Requirements Tracking

Detail-rich reports

Requirements Management

- Import/Synchronization of requirements
- Import/Export/Synchronization of test cases
- Import/Export/Synchronization of links
- Impact analysis when requirement changes
- DOORS® Integration
## Content

<table>
<thead>
<tr>
<th>1</th>
<th>Need and Challenges for Function Developers</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>What is RT2?</td>
</tr>
<tr>
<td>3</td>
<td>Test Case Modeling</td>
</tr>
<tr>
<td></td>
<td>Test Case Execution</td>
</tr>
<tr>
<td></td>
<td>Test Case Assessment and Analysis</td>
</tr>
<tr>
<td></td>
<td>Test Report Generation</td>
</tr>
<tr>
<td>4</td>
<td>Benefits of RT2</td>
</tr>
</tbody>
</table>
A new approach to Model-Based Testing in Simulink®
Key Benefits of RT2

Comprehensive tools suite for test design, automation, assessment and documentation for Simulink® models

− Intuitive test case creation - Enables users to quickly develop test cases with a short learning curve

− Test variant management - Enables users to explore functional behavior in multifaceted test cases, in large test campaigns

− Powerful assessment capabilities - Allows users to make decisions from test results without additional tools

− Traceability between requirements and models - Improves quality and ensures that deliverables meet the needs and expectations of stakeholders

− Test reuse across development phases (MiL/SiL)

− Supports ISO26262 compliant development processes
Thank you

Merci

Vielen Dank

감사합니다.

谢谢

Vielen Dank