

What's new in MATLAB and Simulink for Model-Based Design

Magnus Jung
Application Engineer

What's New?

☰ CONTENTS
Close

< All Products

< Simulink i

Release Notes for Simulink

ON THIS PAGE

- R2016a
- R2015aSP1
- R2015b
- R2015a
- R2014b
- R2014a
- R2013b
- Compatibility Summary

Simulink Release Notes

Bug Reports
Bug Fixes

▼ R2016a

New Features, Bug Fixes, Compatibility Considerations

Simulation Analysis and Performance

- › Automatic Solver Option: Set up and simulate your model more quickly with automatically selected solver settings
- › One-Click Display: Click a signal line within the simulation to zoom in on the connection
- › Simulation Metadata Diagnostics: Under the hood, you can now see the simulation metadata
- › Multi-Input Root Inport Mapping: Connect multiple inports to a single root inport
- › Simulation for Mixed Targets: Simulate your model on different hardware targets
- › Time Out feature for Performance Advisor: Set a time limit for the Performance Advisor analysis
- › Solver Profiler to speed up simulation performance
- › Diagnostic Viewer performance improvements

Component-Based Modeling

- › Variant Source and Sink Blocks with Component-Based Modeling
- › Scoping Simulink Functions: Call Simulink functions from within a scope
- › Simulink Units: Specify, visualize, and check consistency of units on chart interfaces
- › Mask Dialogs: Create masks with flexible dialog boxes
- › Mask Images: Quickly add images to mask dialog boxes
- › Tracing Simulink Functions: Display call stacks for Simulink functions

☰ CONTENTS
Close

< All Products

< Stateflow i

Release Notes for Stateflow

ON THIS PAGE

- R2016a
- R2015aSP1
- R2015b
- R2015a
- R2014b
- R2014a
- R2013b
- R2013a
- R2012b
- R2012a
- R2011b
- R2011a

Stateflow Release Notes

Bug Reports
Bug Fixes

▼ R2016a

New Features, Bug Fixes, Compatibility Considerations

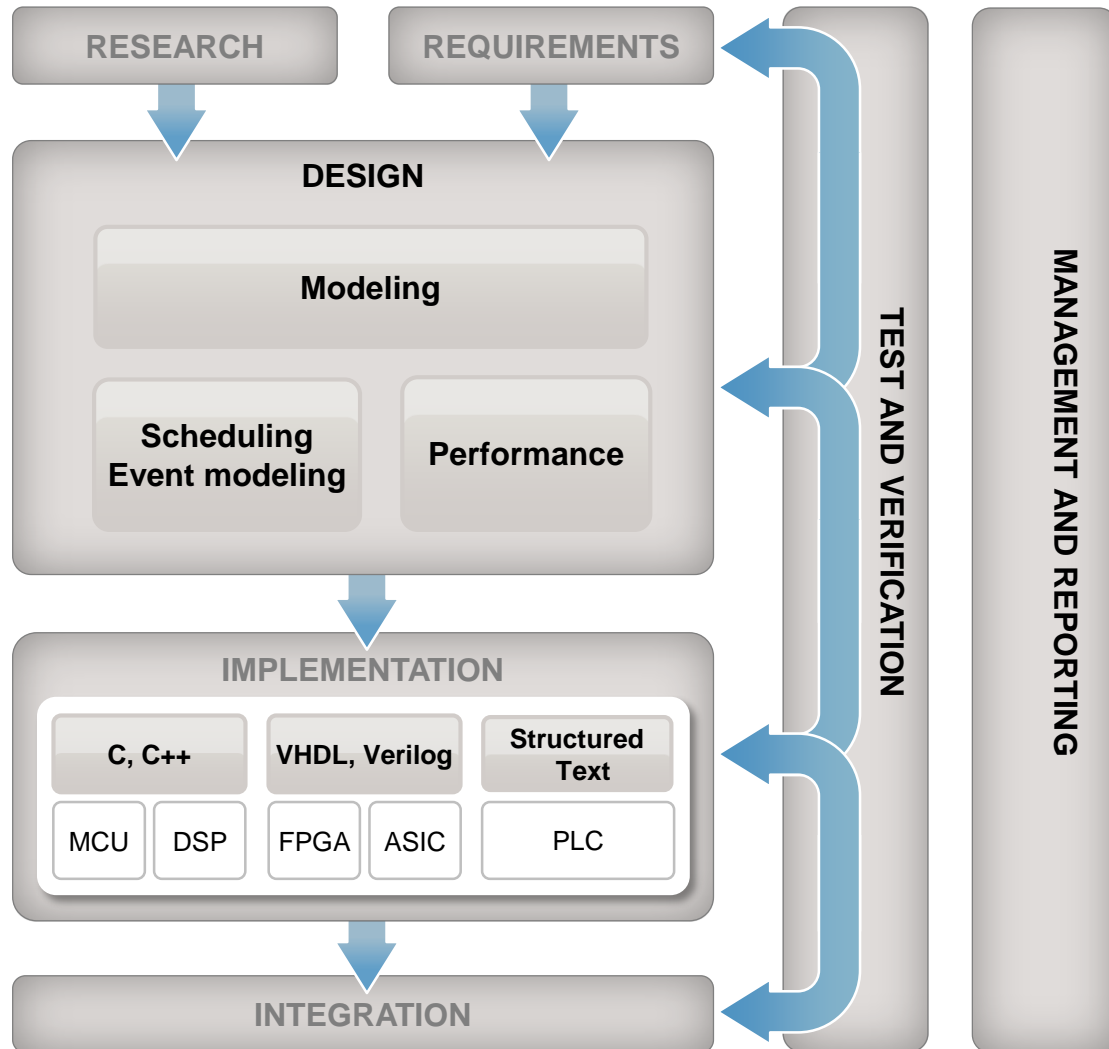
- › Smart Editing Cues: Accelerate common editing tasks with just-in-time contextual prompts
- › Intelligent Chart Completion: Build charts faster with automatic addition of default transitions and creation of complementary state names
- › Simulink Units: Specify, visualize, and check consistency of units on chart interfaces
- › Output Logging: Log output signals for charts
- › JIT for Messages: Reduce model update time for messages with JIT compilation technology
- › API changes for commented objects ⚠
- › Stateflow model templates for common design patterns
- › UserData parameter available for storing values

R2015aSP1

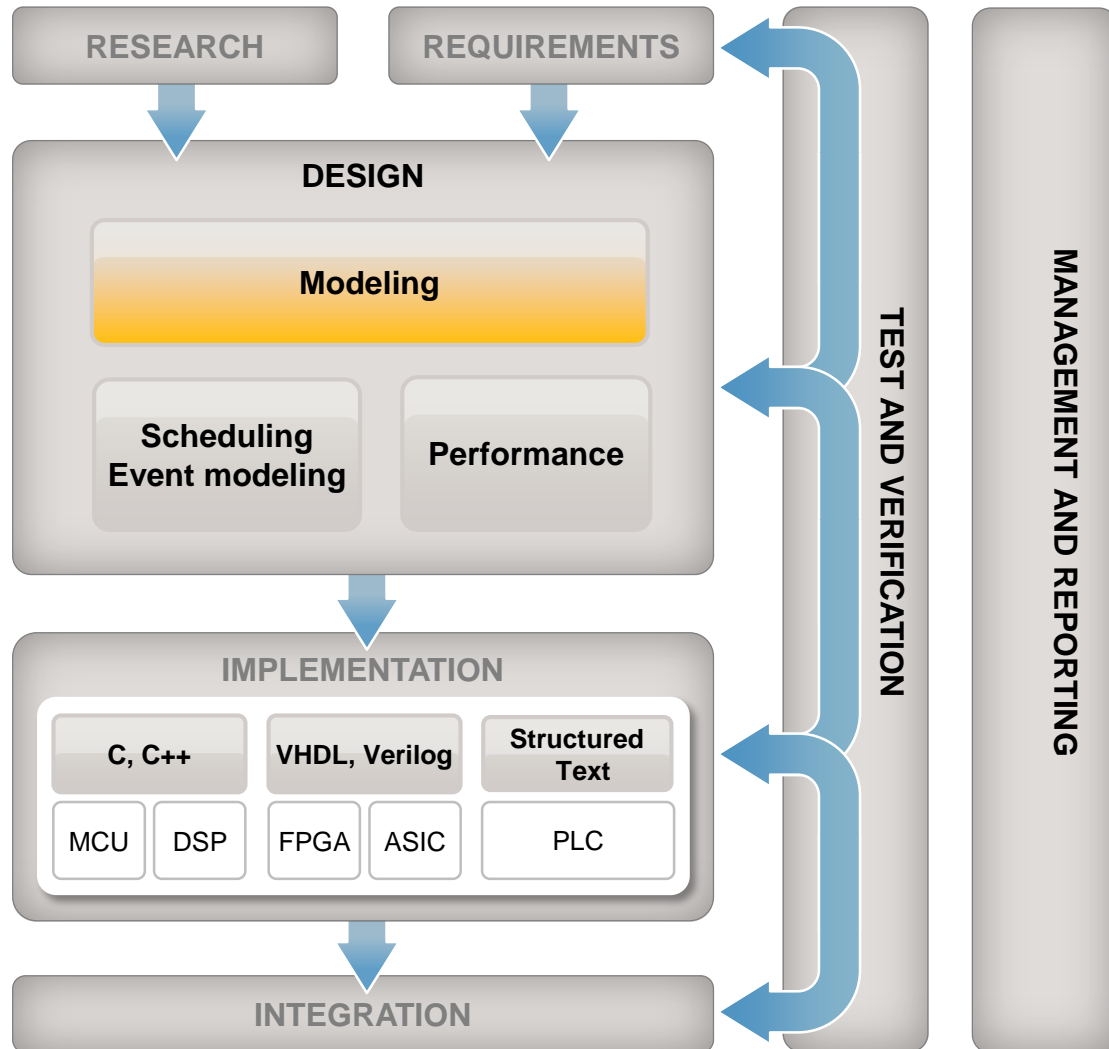
Bug Fixes

› R2015b

Model-Based Design Workflow



Model-Based Design Workflow



HOME PLOTS APPS SHORTCUTS

Expo L M C cl P Search Documentation

New Script New Open Compare Import Data Save Workspace New Variable Open Variable Clear Workspace Analyze Code Run and Time Clear Commands Simulink Layout Parallel Preferences Set Path Add-Ons Help Request Support Community

FILE VARIABLE CODE SIMULINK ENVIRONMENT RESOURCES

C:\Projects\Internal\EXPO2016\WhatsNewInMLandSLforMBD\MATLAB

Current Folder

Name ^

- Folder
 - drafts
 - present
 - slprj
- Script
 - openSE_resource.m
- Simulink Model
 - modelingExample.slx
- WinZip File
 - modelingExample_20160418.zip

Command Window

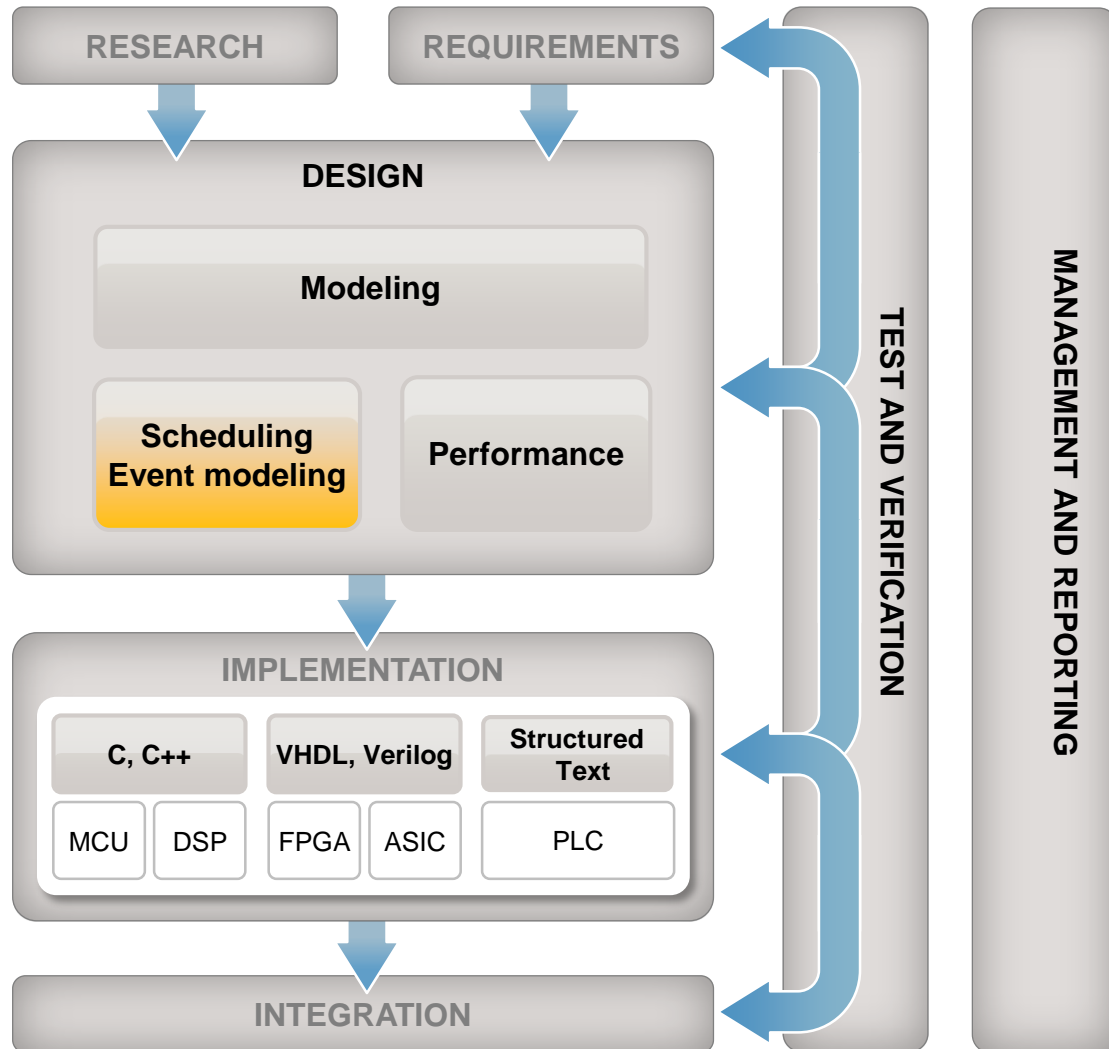
```
f >>
```

modelingExample.slx (Simulink Model)

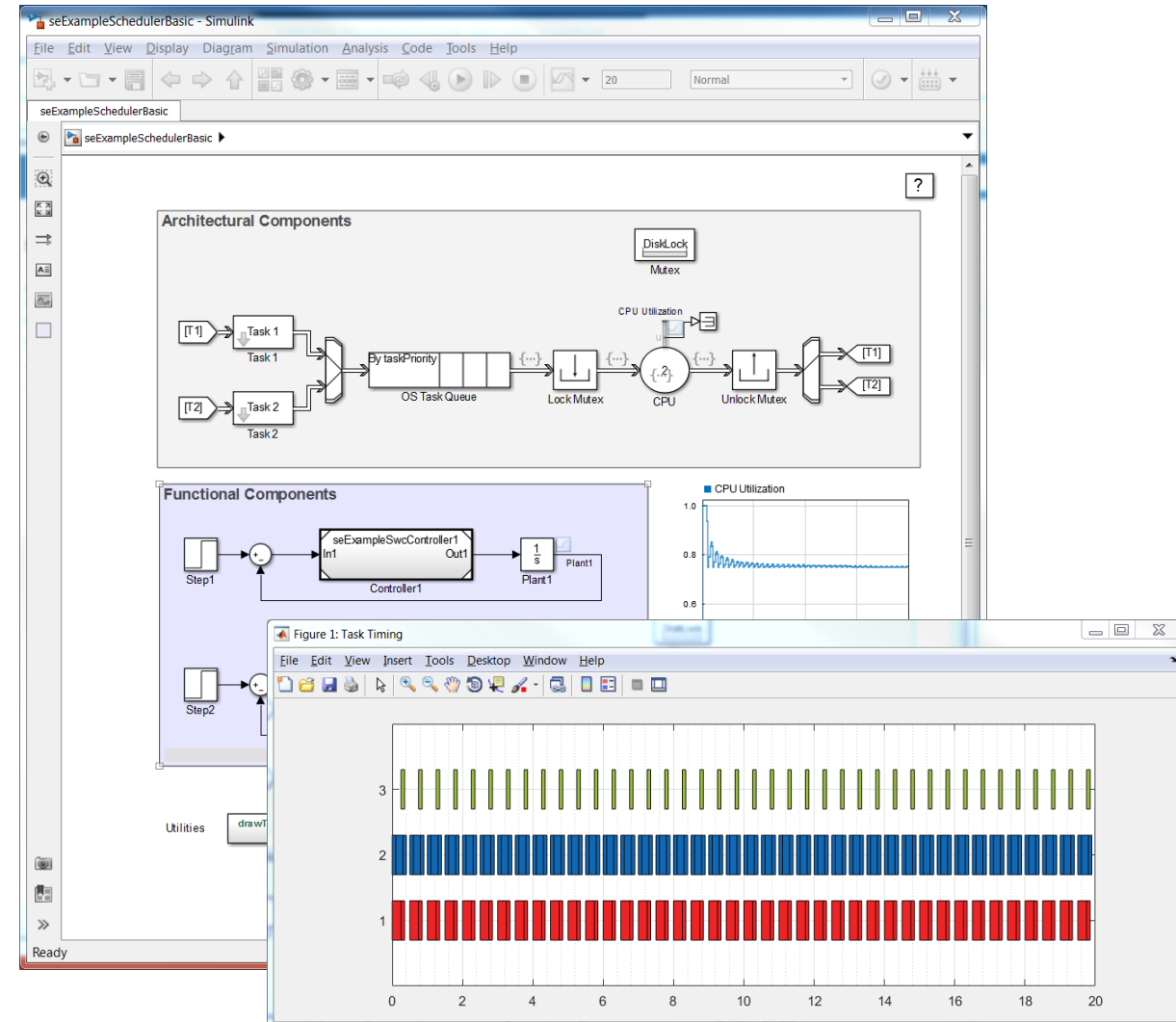
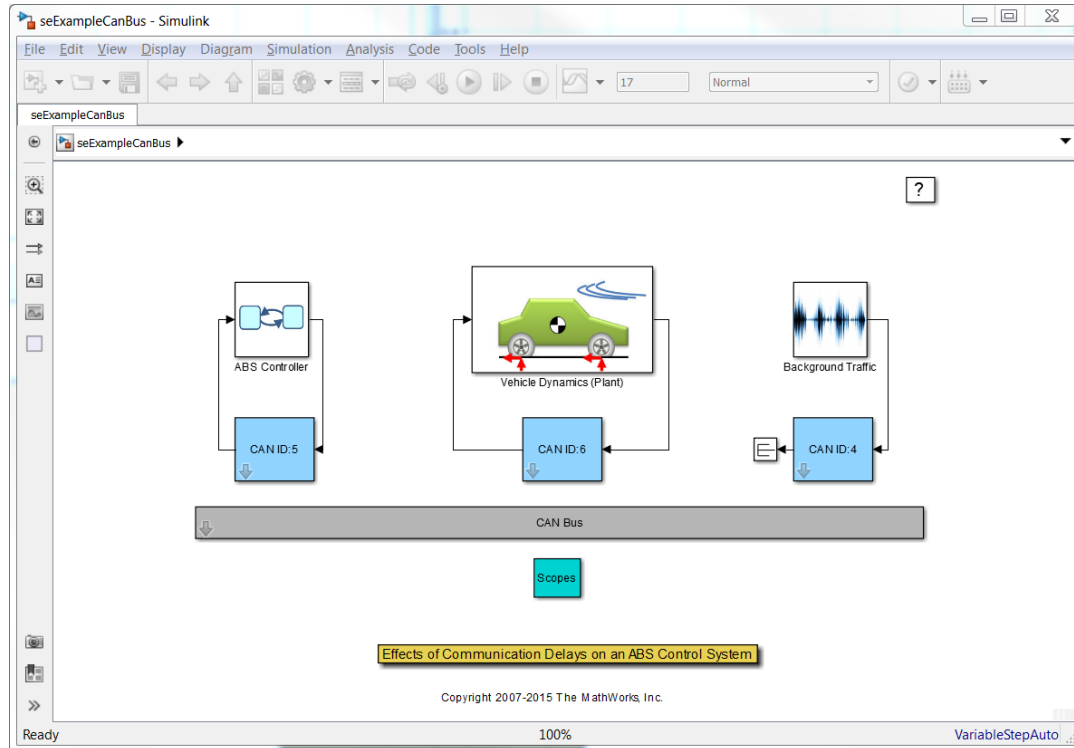
Workspace

Name ^	Value
--------	-------

Model-Based Design Workflow

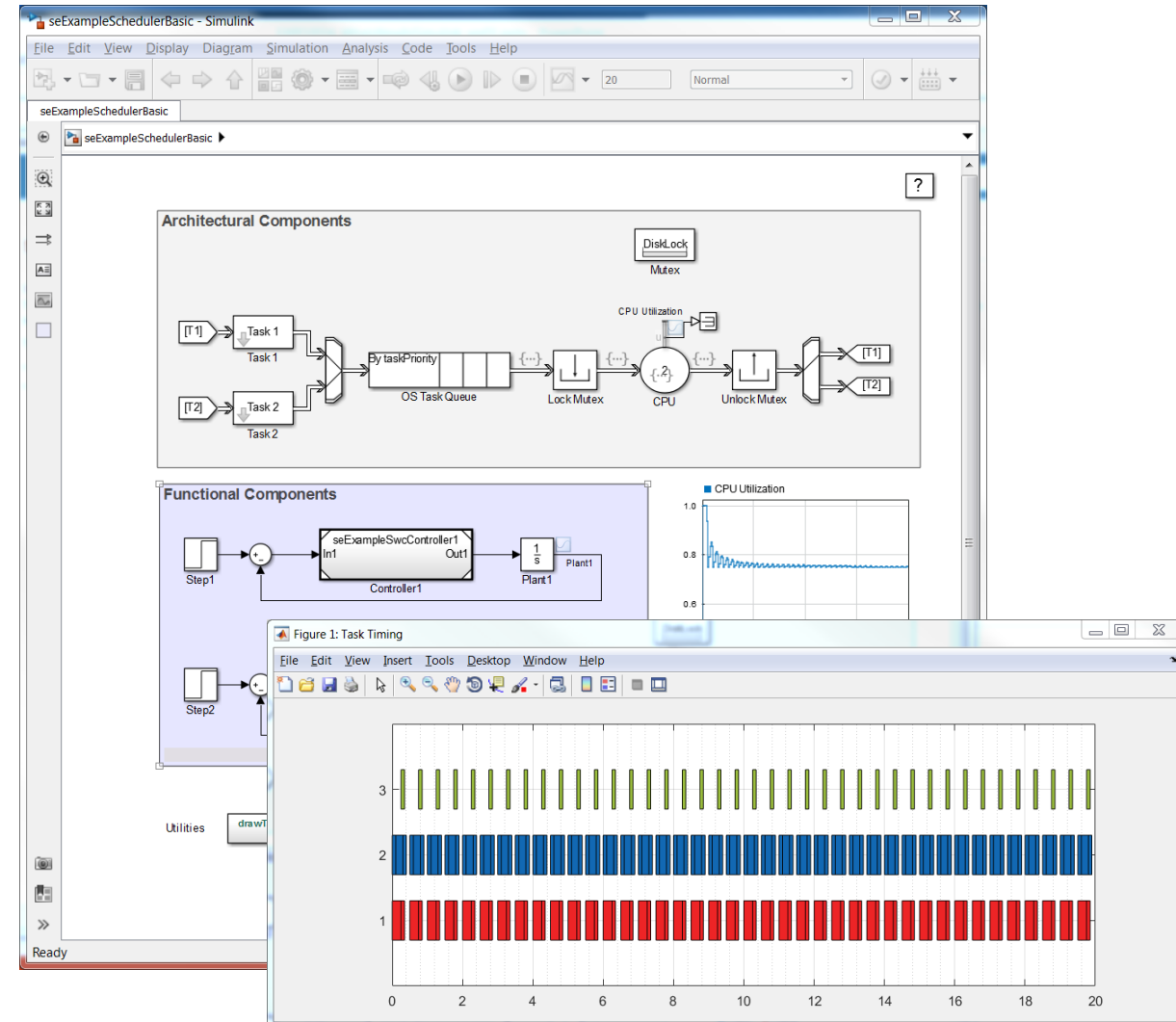
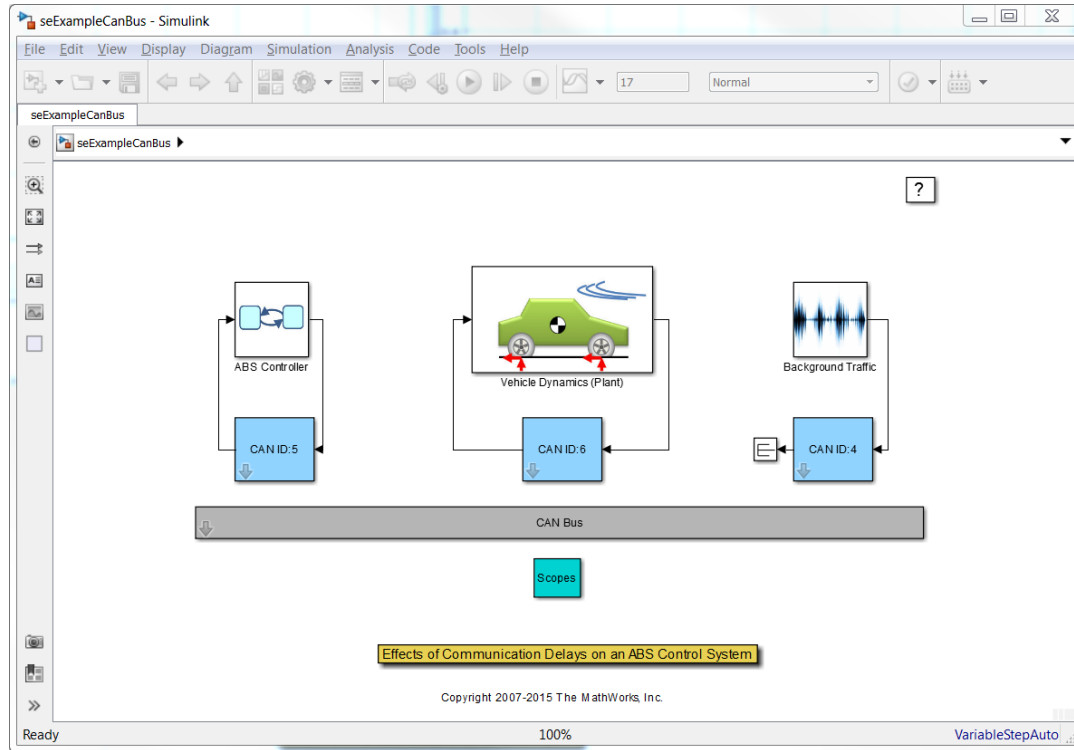


Messages, Functions and Scheduling



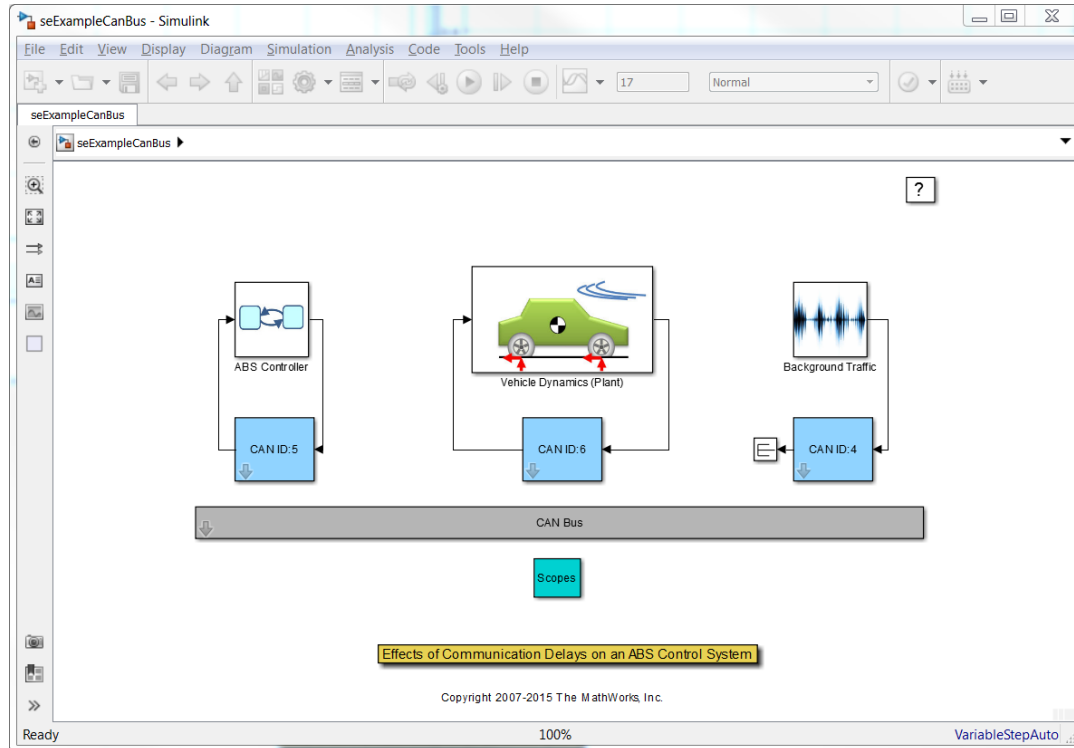
New SimEvents

Discrete-event simulation engine for multidomain system models



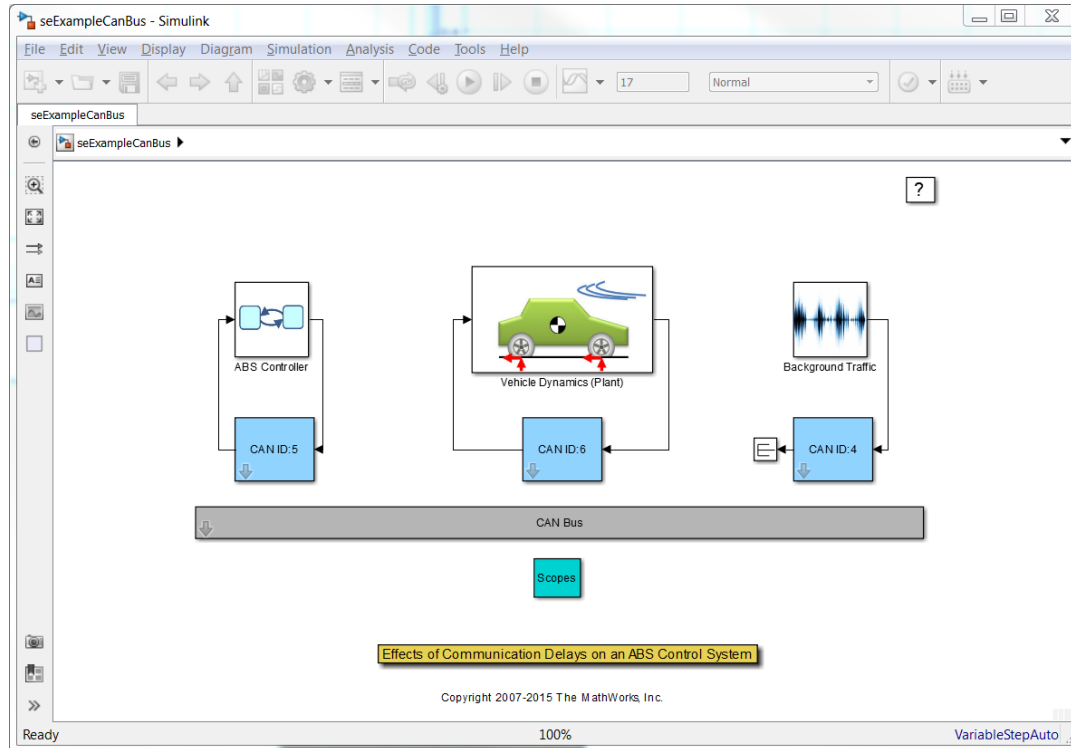
New SimEvents

Discrete-event simulation engine for multidomain system models



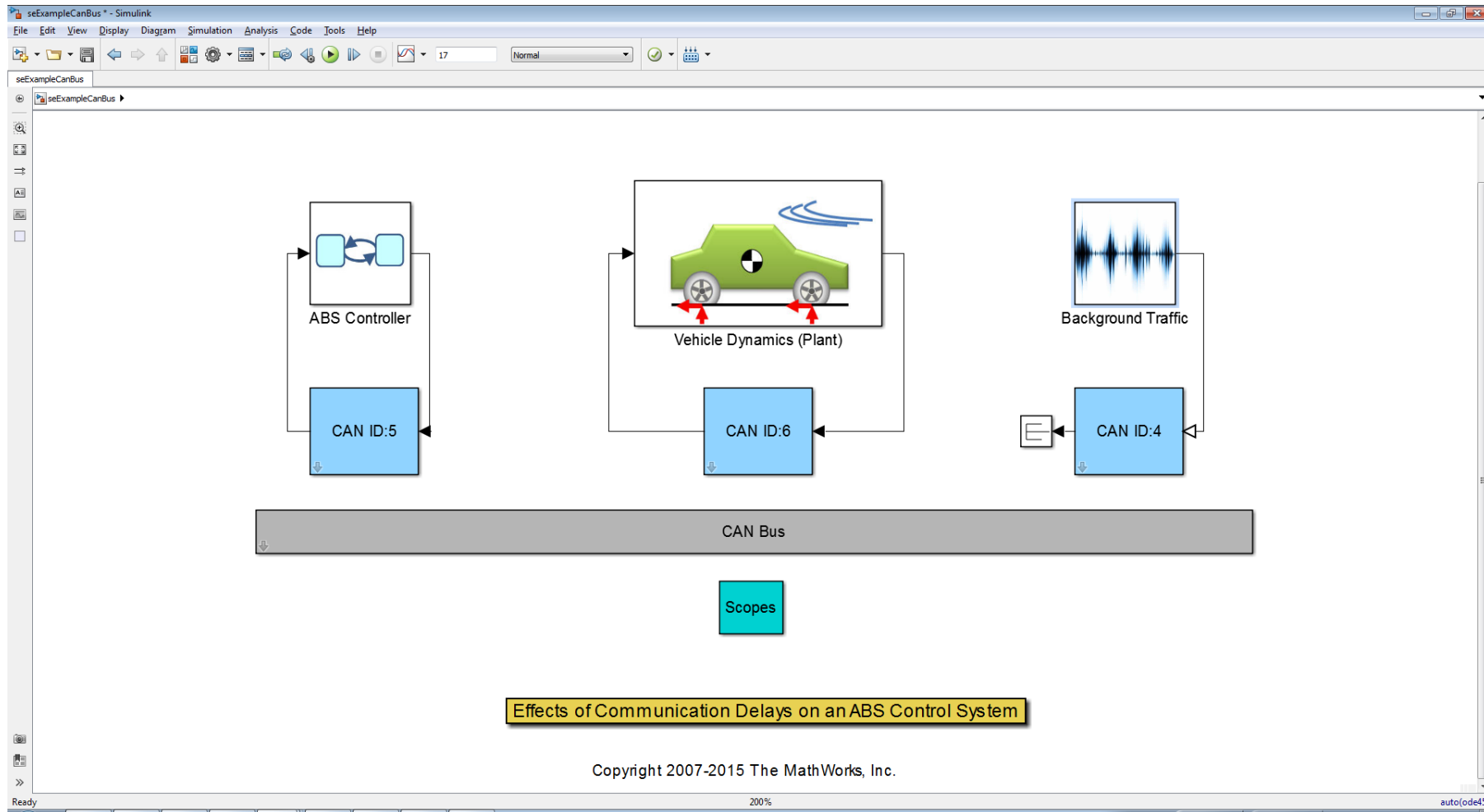
New SimEvents

Discrete-event simulation engine for multidomain system models



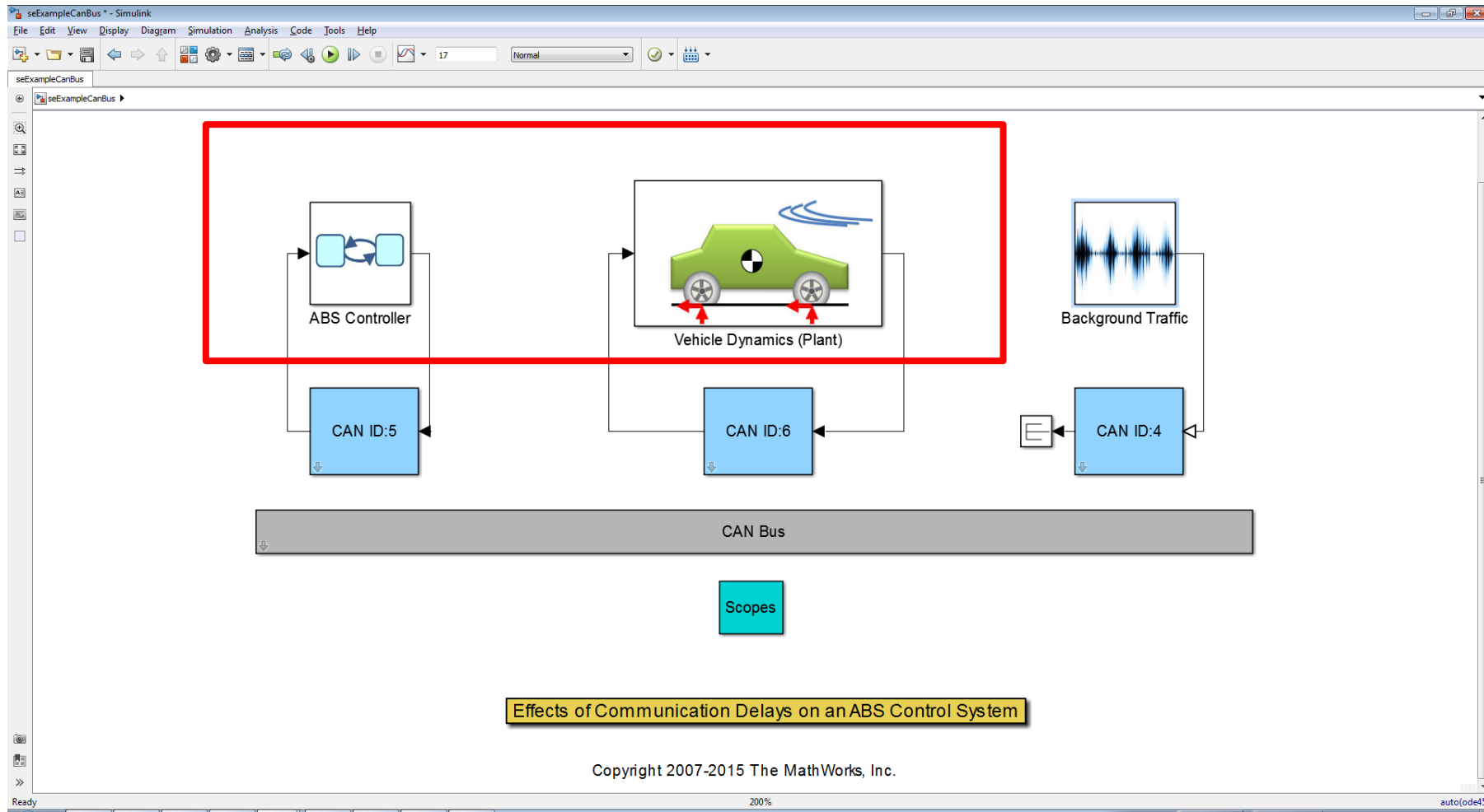
How does communication delays effect your system performance?

CAN simulation with Simulink and SimEvents

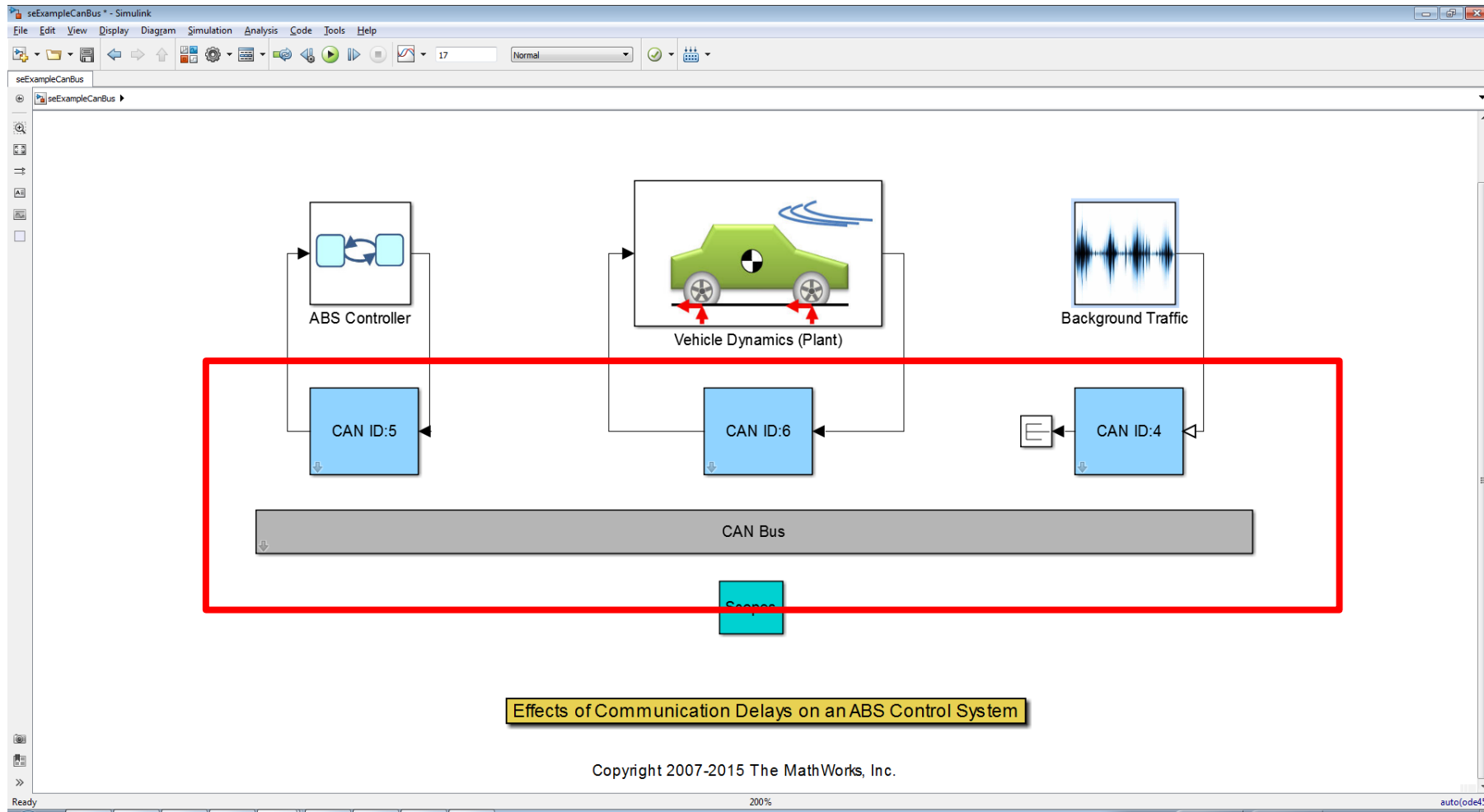


Copyright 2007-2015 The MathWorks, Inc.

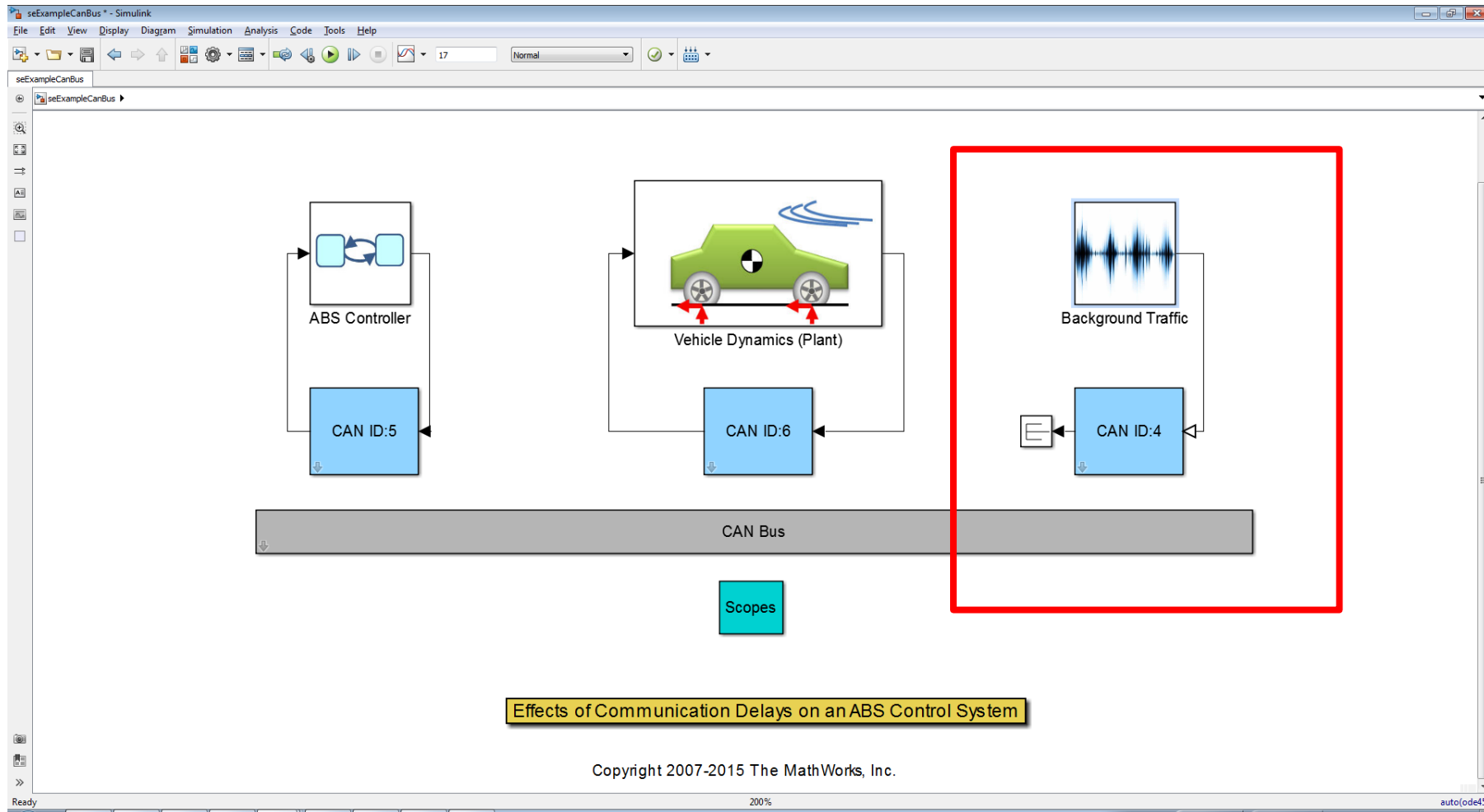
CAN simulation with Simulink and SimEvents



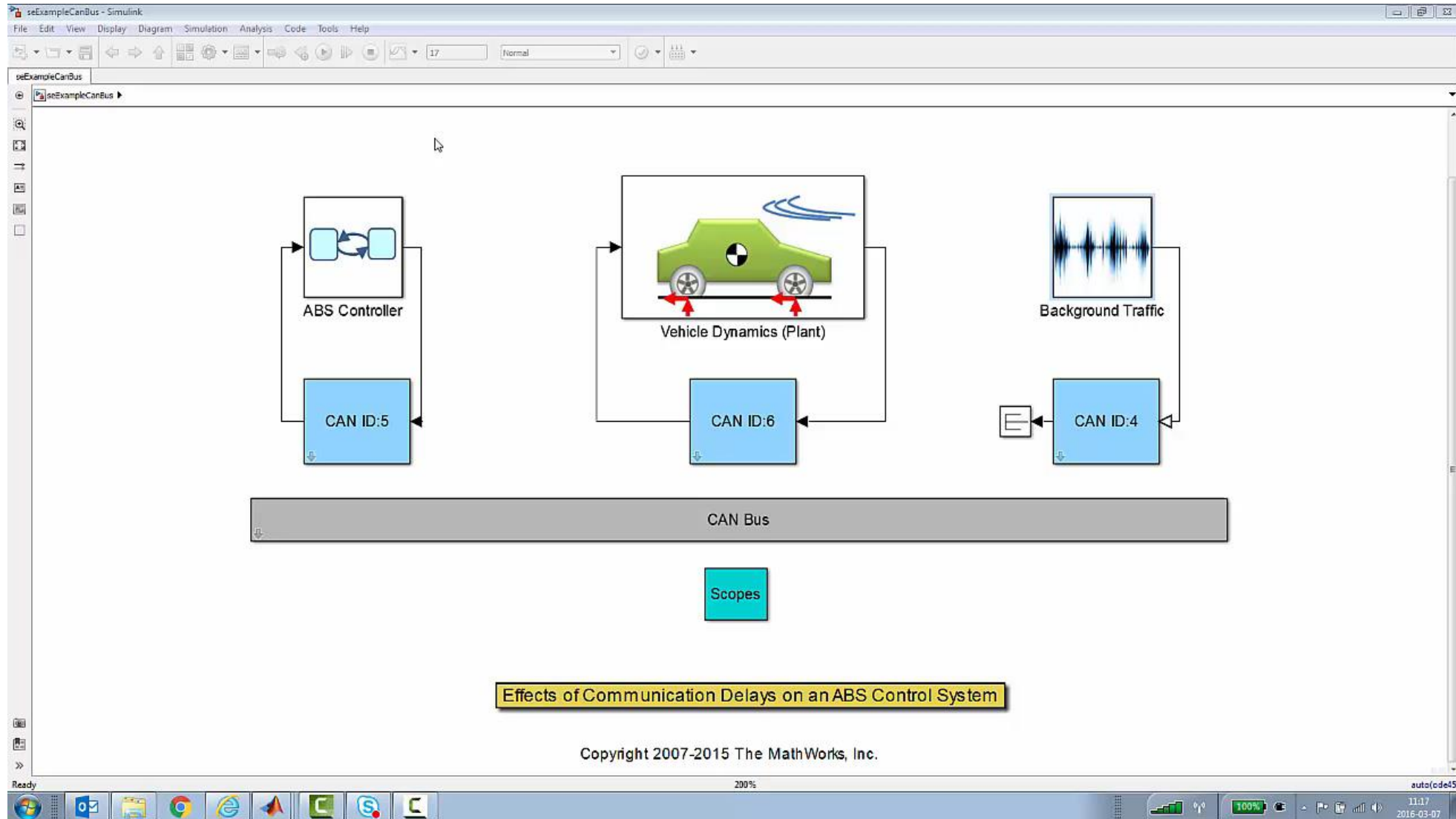
CAN simulation with Simulink and SimEvents



CAN simulation with Simulink and SimEvents

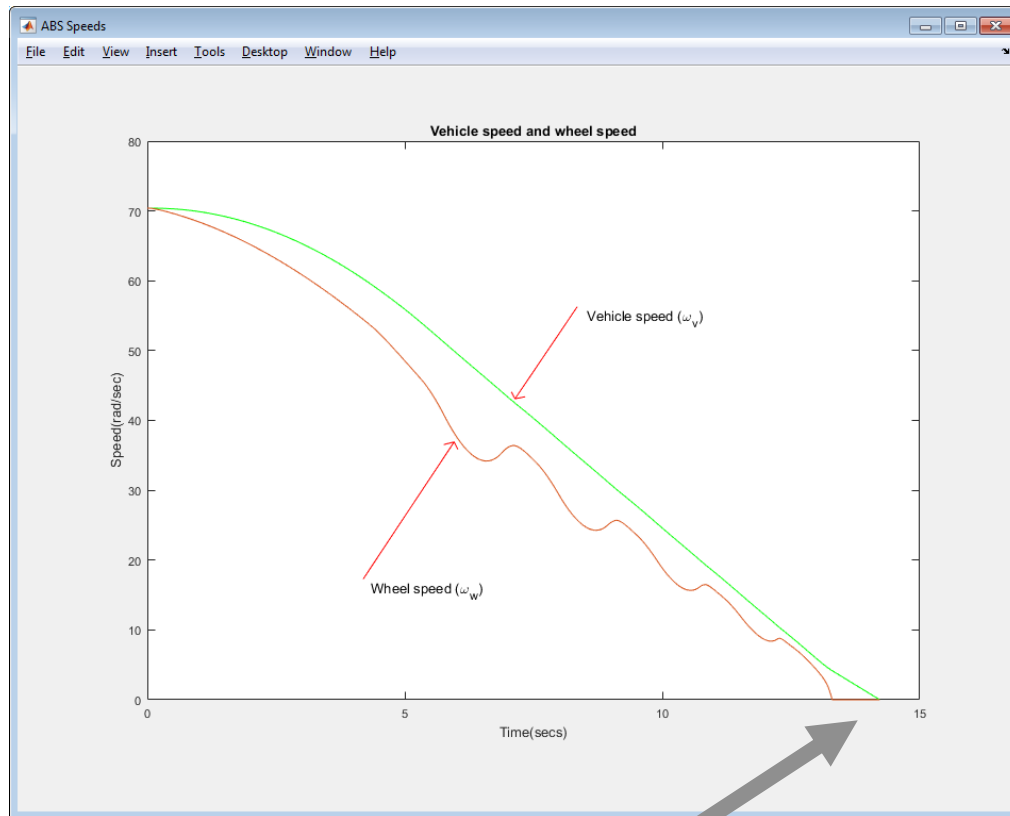


CAN simulation with Simulink and SimEvents



CAN simulation with Simulink and SimEvents

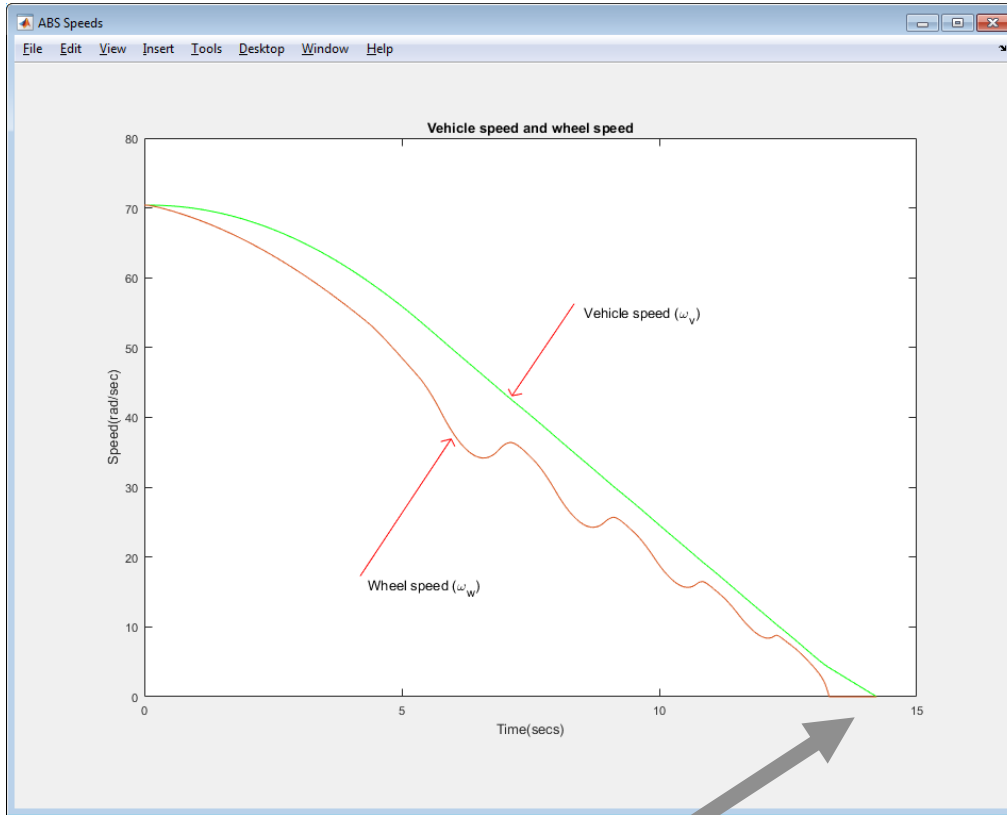
Without background noise



Vehicle stops after 14.2 seconds

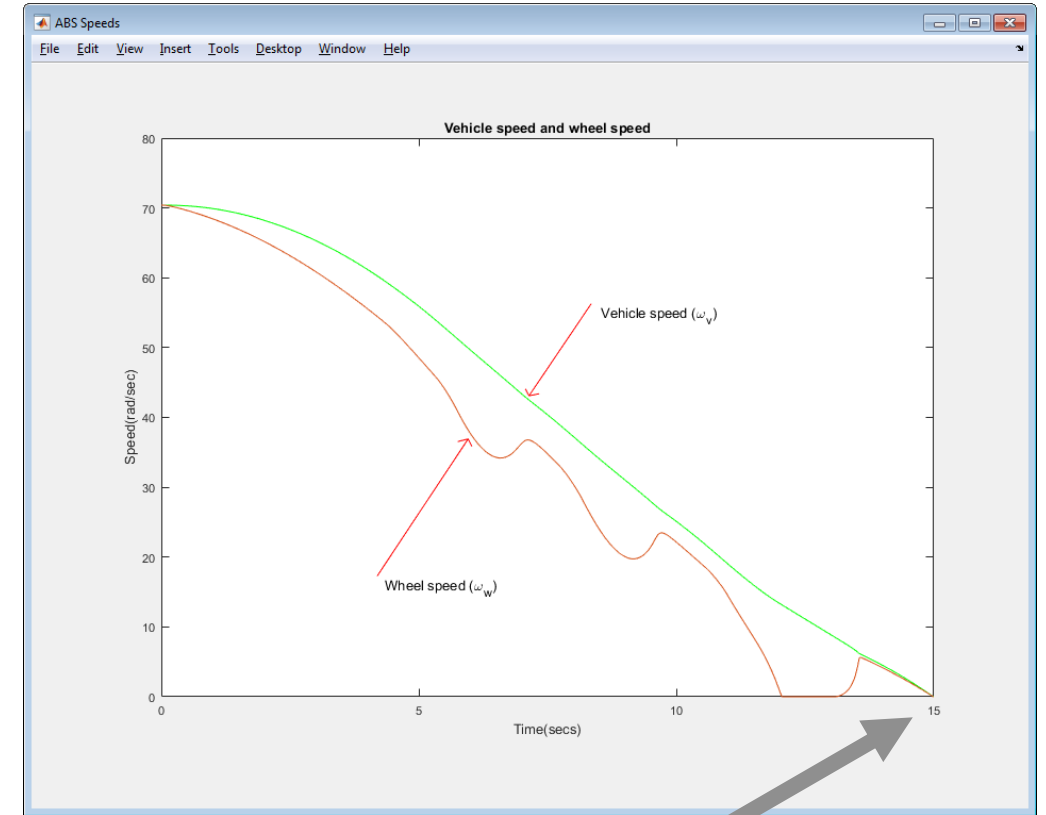
CAN simulation with Simulink and SimEvents

Without background noise



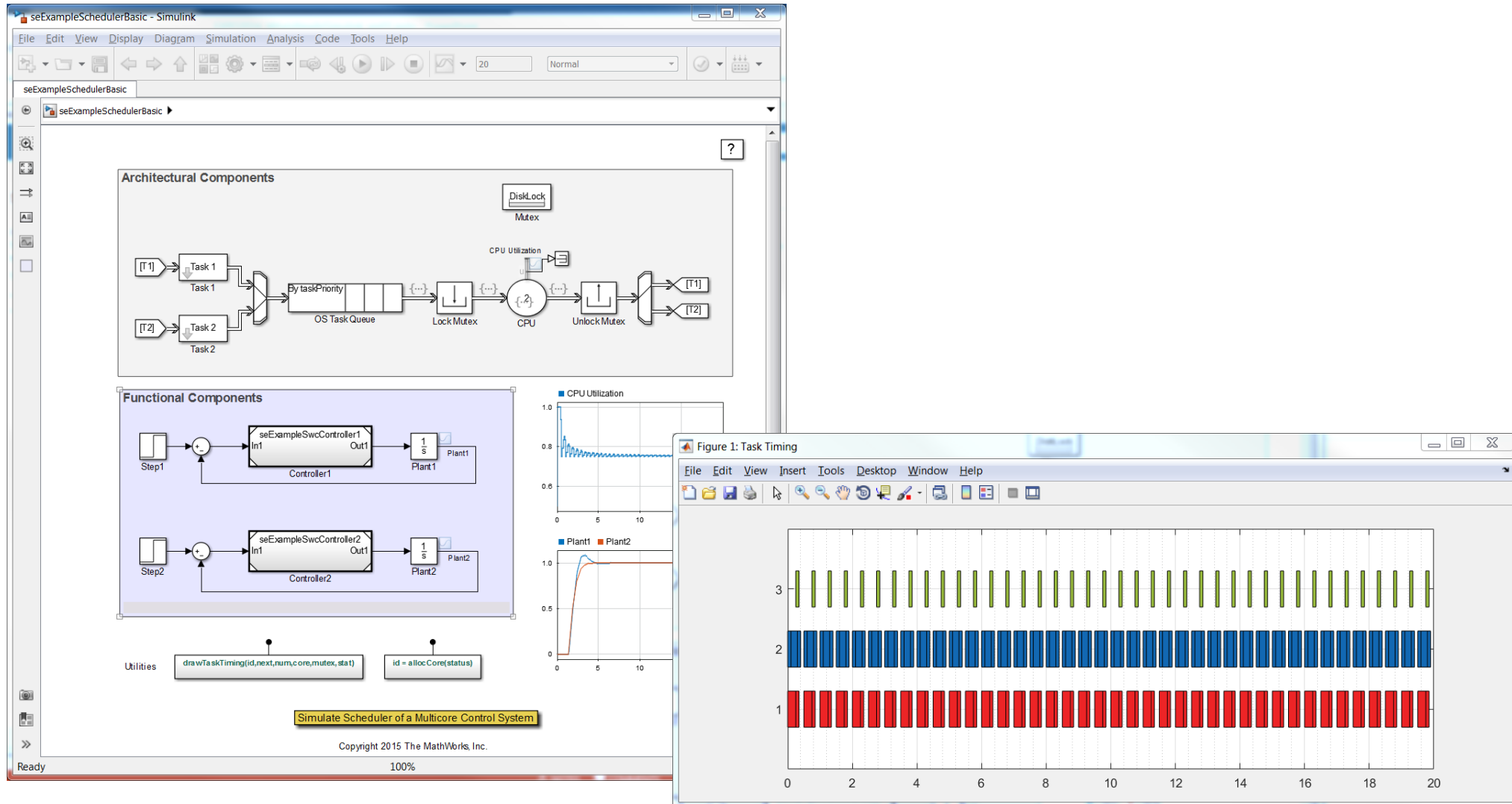
Vehicle stops after 14.2 seconds

With background noise

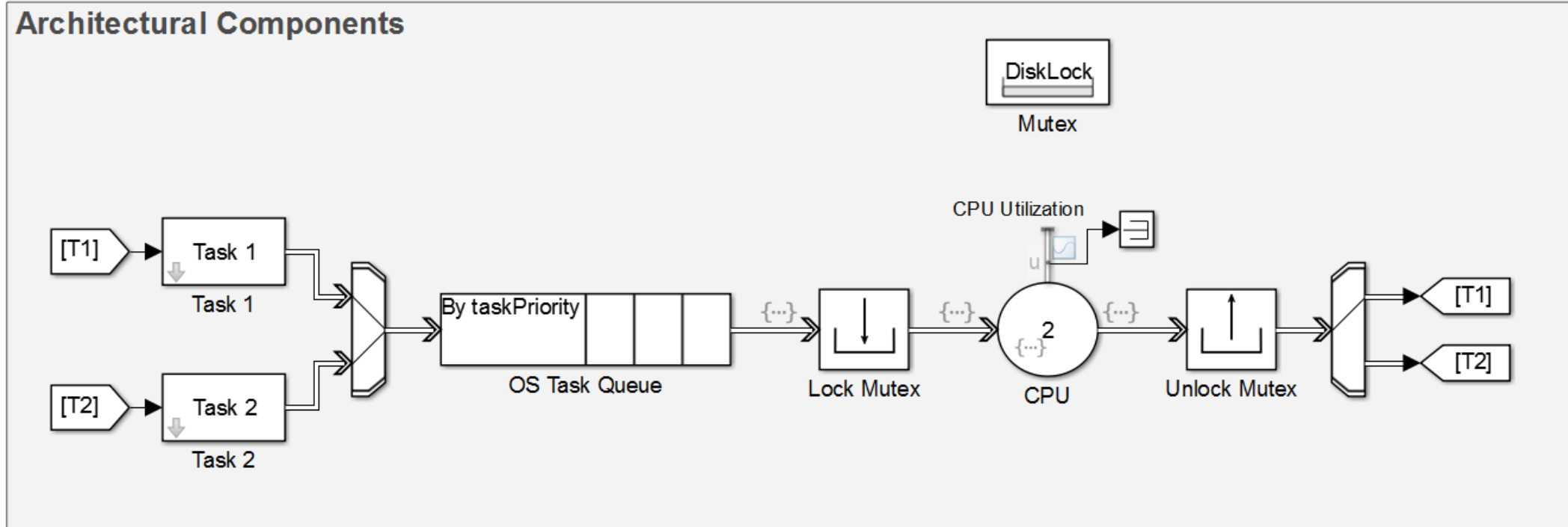


Vehicle stops after 15 seconds

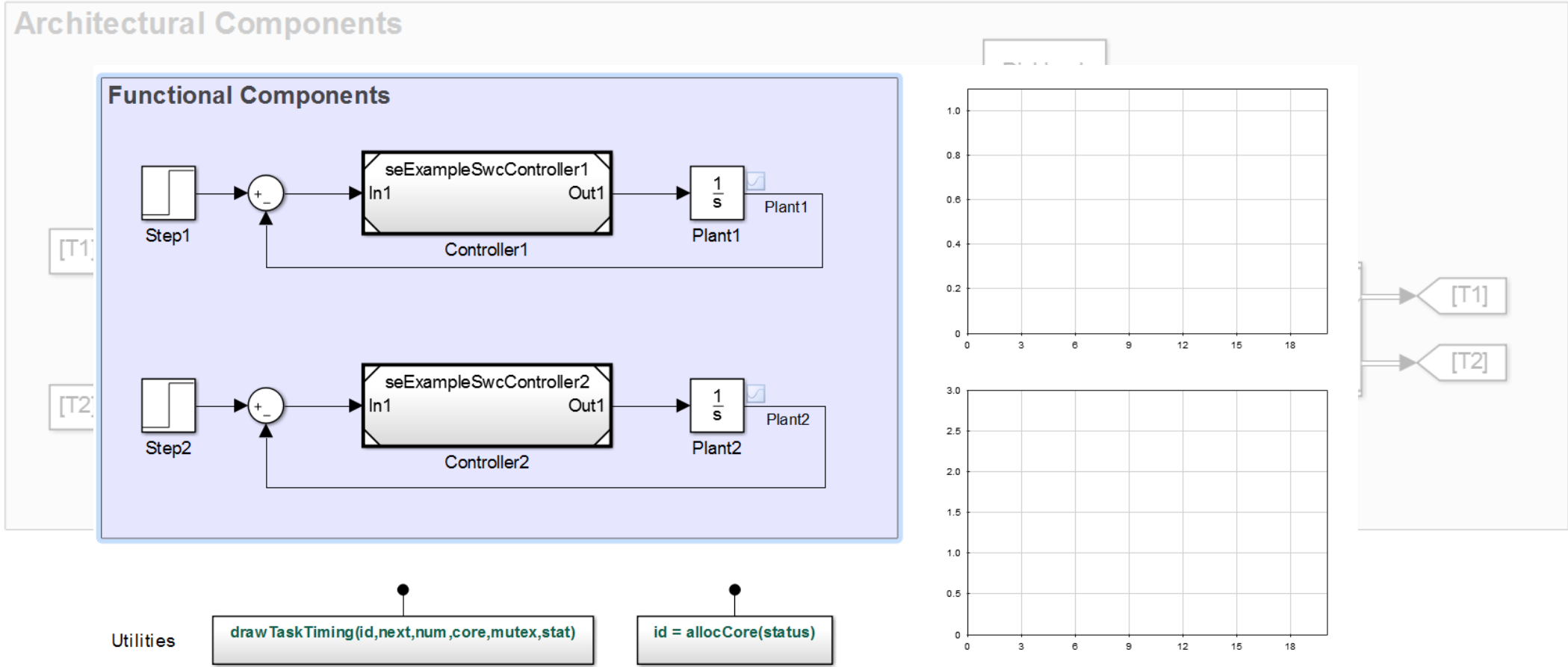
Scheduler Example



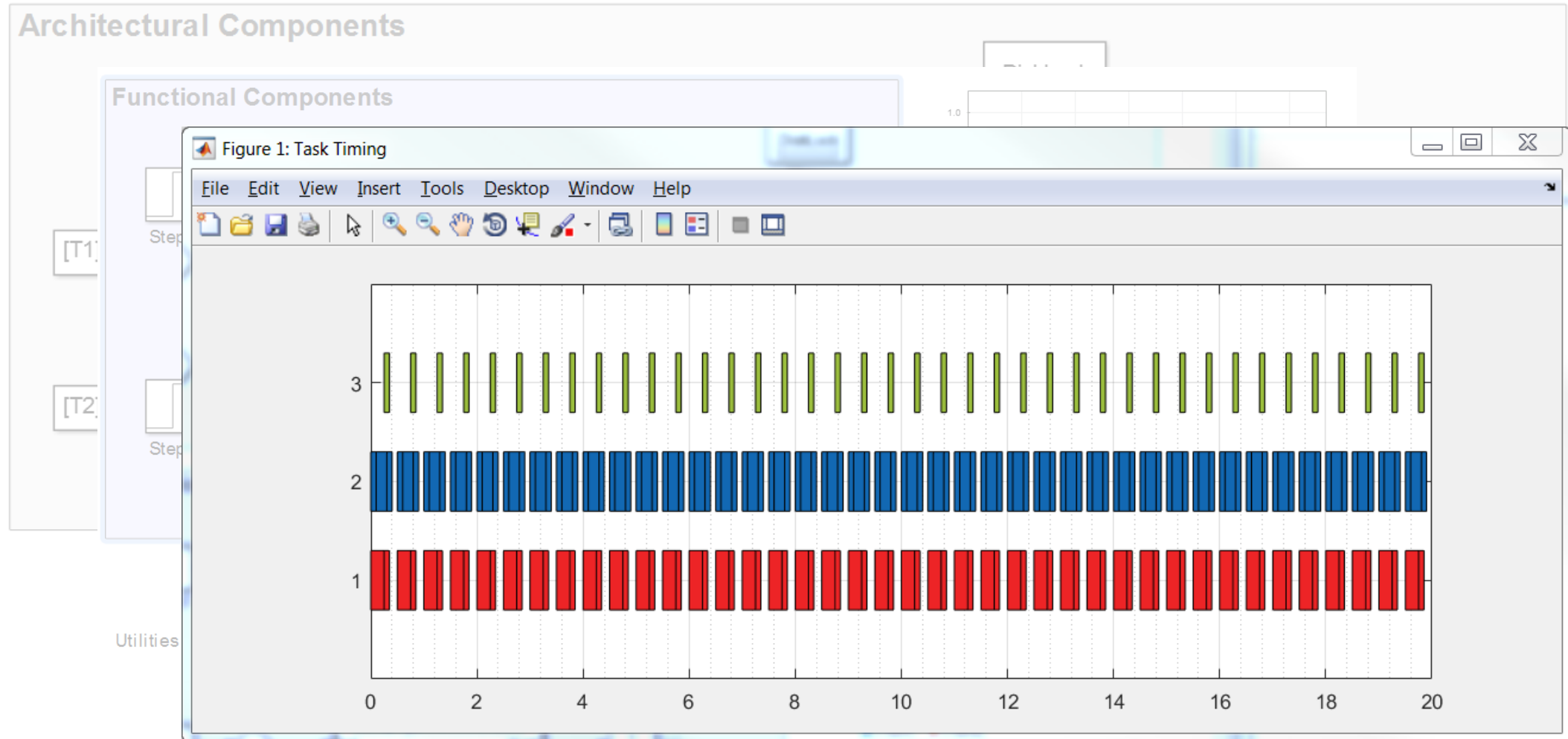
Scheduler Example



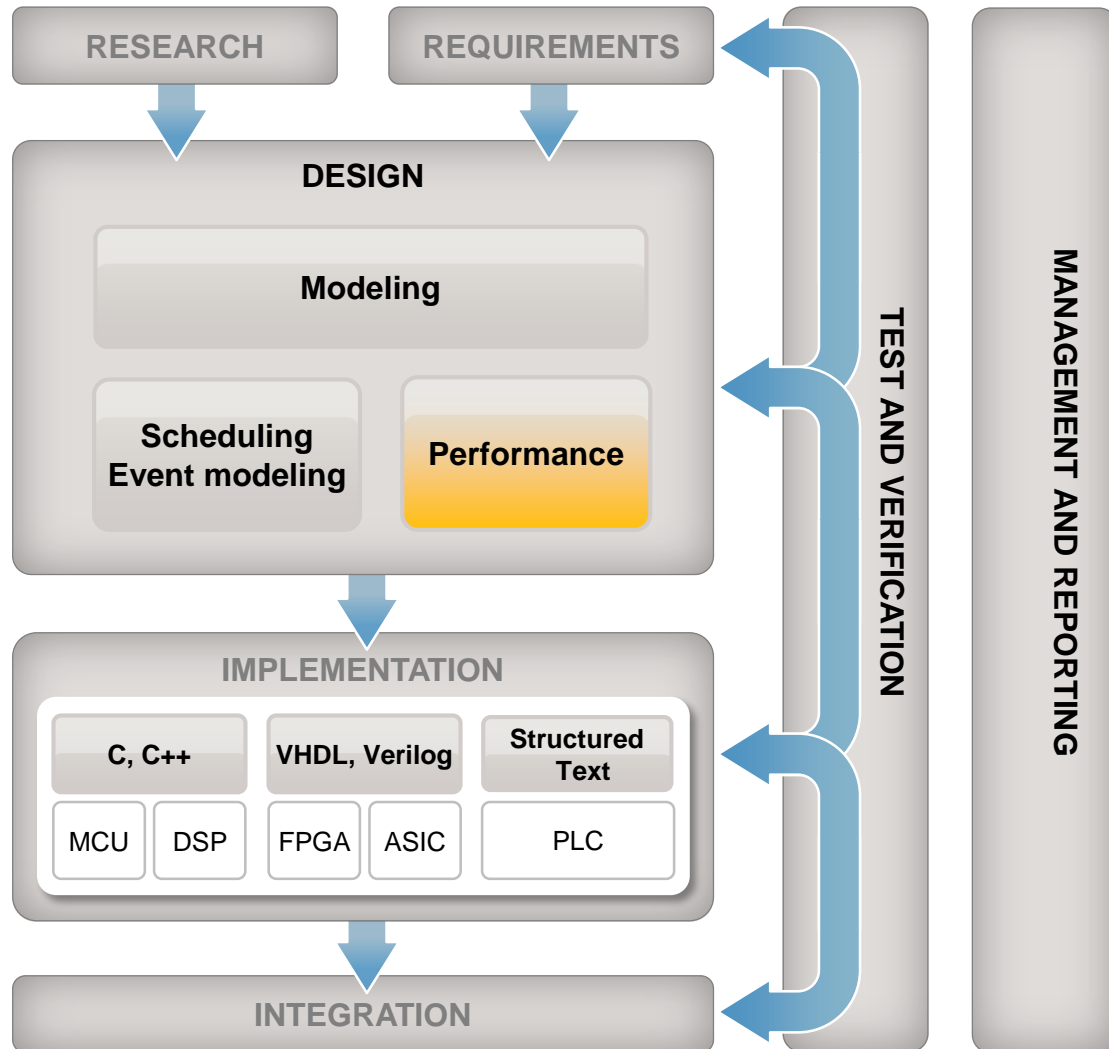
Scheduler Example



Scheduler Example



Model-Based Design Workflow



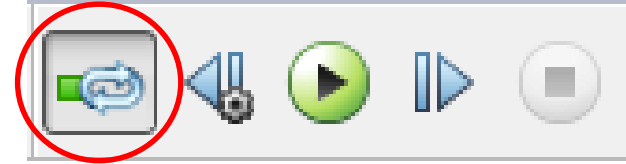
Fast Restart

Run consecutive simulations more quickly



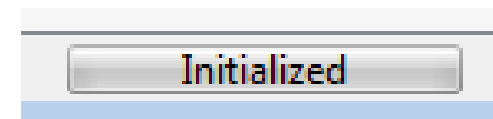
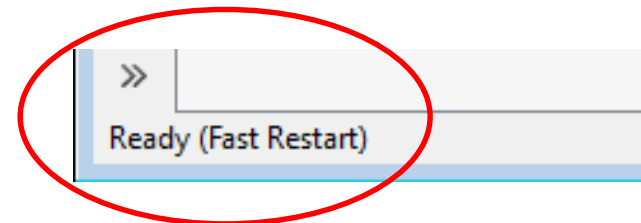
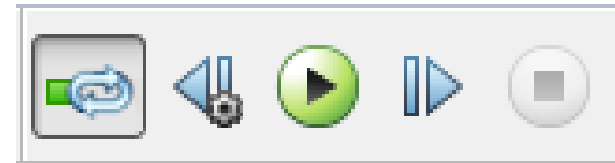
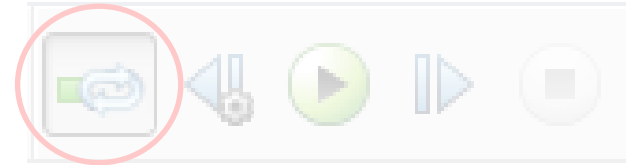
Fast Restart

Run consecutive simulations more quickly



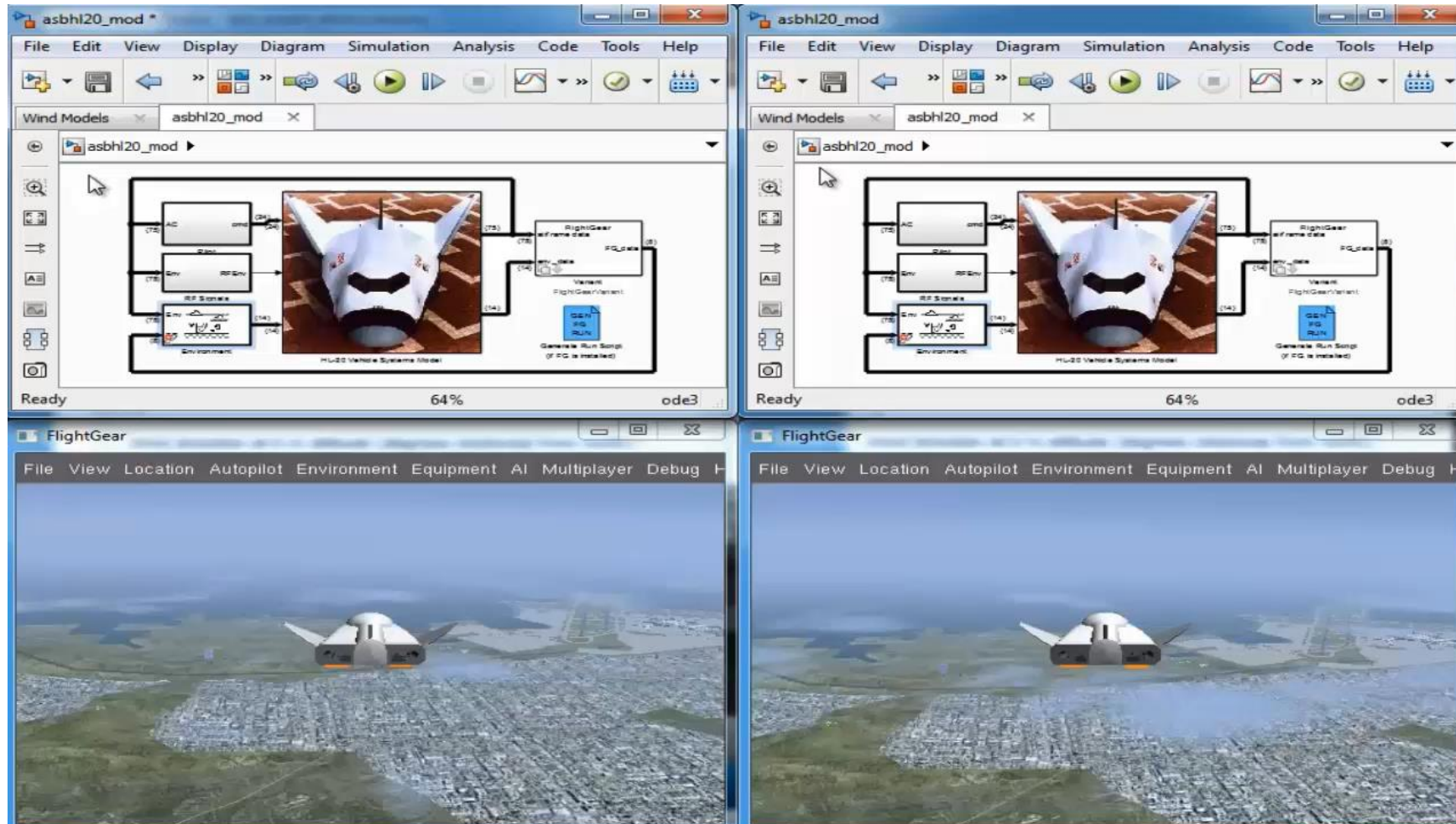
Fast Restart

Run consecutive simulations more quickly

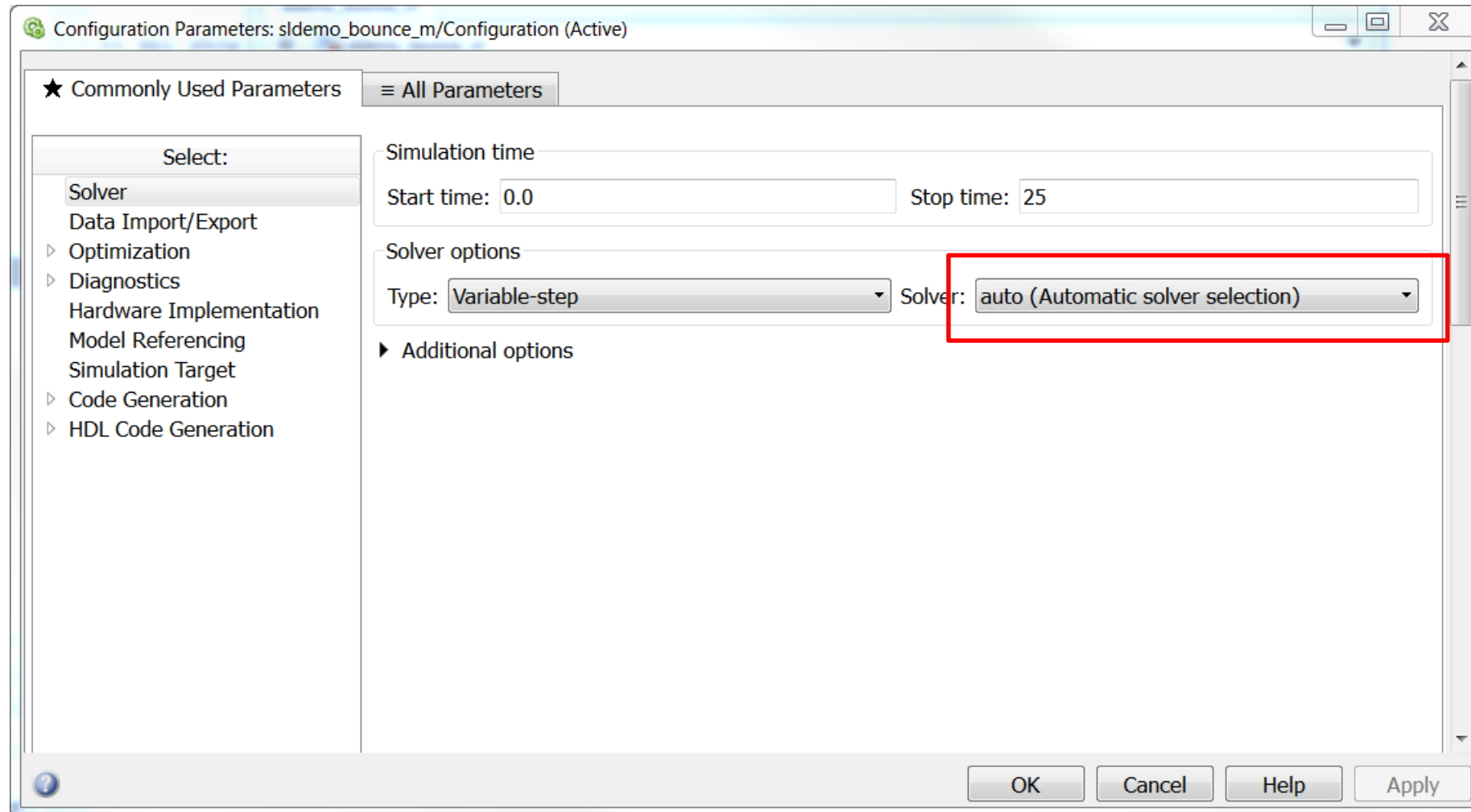


Simulink - Faster consecutive simulations

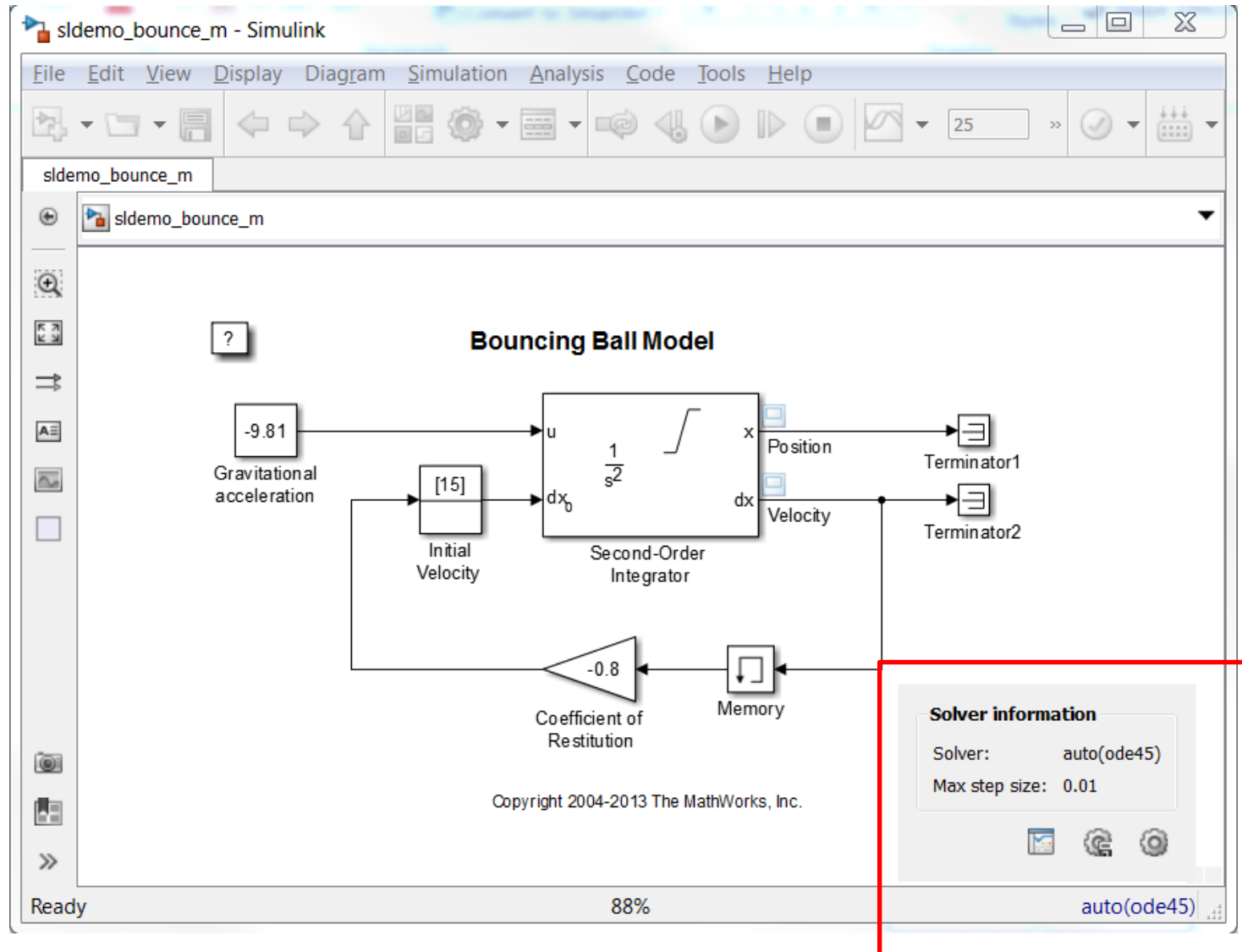
Fast Restart



Automatic Solver Selection



Understanding the selected solver



Understanding the selected solver

The screenshot displays the Simulink environment for a model named "sldemo_bounce_m". The main workspace shows a "Bouncing Ball Model" diagram. The diagram consists of the following blocks and connections:

- A constant block with the value -9.81 labeled "Gravitational acceleration" is connected to the input u of the "Second-Order Integrator" block.
- A constant block with the value $[15]$ labeled "Initial Velocity" is connected to the input dx_0 of the "Second-Order Integrator" block.
- The "Second-Order Integrator" block has a transfer function of $\frac{1}{s^2}$ and outputs "Position" (x) and "Velocity" (dx).
- The "Velocity" output is connected to a "Memory" block.
- The "Memory" block is connected to a gain block with the value -0.8 labeled "Coefficient of Restitution".
- The output of the gain block is connected back to the input dx_0 of the "Second-Order Integrator" block.
- The "Position" output is connected to a "Terminator1" block.
- The "Velocity" output is also connected to a "Terminator2" block.

At the bottom right of the workspace, a "Solver information" panel is highlighted with a red box. It displays the following information:

- Solver: auto(ode45)
- Max step size: 0.01

The status bar at the bottom of the window shows "Ready", "88%", and the solver name "auto(ode45)".

Understanding the selected solver - Solver Profiler

Solver Profiler: sldemo_bounce_m - Statistics

SOLVER PROFILER

From: 0 To: 25 Buffer: 50000

Open Save Run Stop

Zoom In Zoom Out Pan

Zero Crossing Solver Exception Solver Reset

Highlight Block Trace to Source Trace to Destination Remove Trace

STATES SHARE

FILE LOG PROFILE VIEW FILTER TRACE

Statistics

MODEL INFORMATION

Blocks with states	2
States	2
Start time	0
Stop time	25
Absolute tolerance	1.00e-06
Relative tolerance	1.00e-03

STEP INFORMATION

Max step size	1.00e-02
Min step size	3.55e-15
Average step size	9.01e-03
Max step size usage(%)	89.37
Total steps	2775

EVENT INFORMATION

Zero crossing source	3
Zero crossing source triggered	1
Total zero crossing	130
Total solver reset	131
Total solver exception	0
Error control exception	0
Newton iteration exception	0

Step Size

Suggestion Zero Crossing Solver Exception

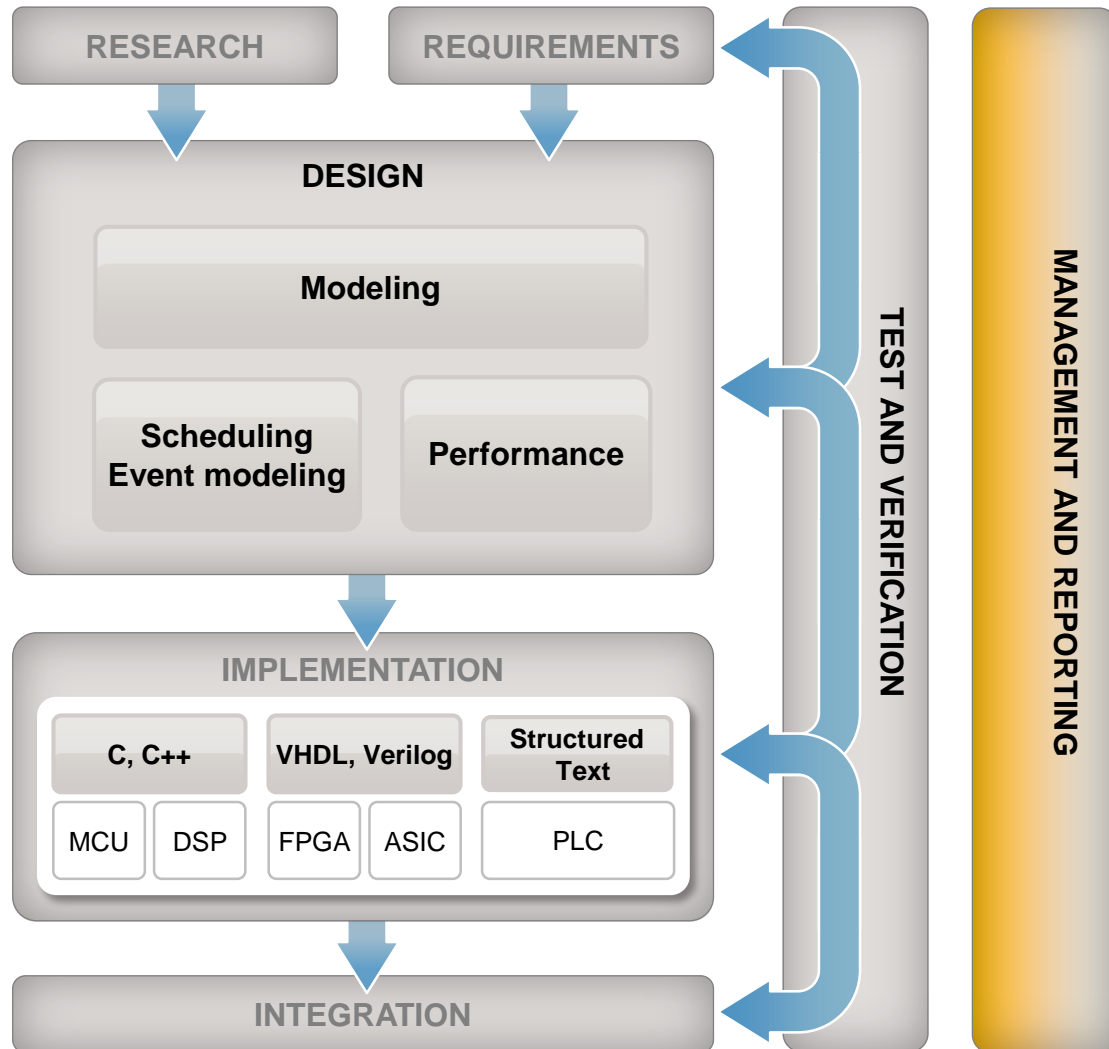
Solver profiler detected dense groups of zero crossings in these regions:
From: 19.36 To: 21.33

1. Identify states that contributed to the majority of zero crossings in these regions.
2. Highlight each block in the model. Explore the zero crossing setting for the block.
3. Examine upstream blocks to identify potential modeling improvements.

During simulation, the solver used the maximum step size 89.37% of total simulation time. You can improve the simulation speed. Increase the maximum step size in the Solver pane of model configuration parameters.

Click and drag to move Zero Crossing or its tab...

Model-Based Design Workflow



Three-Way Model Merge

The screenshot displays the 'Three-Way Merge' interface for a project named 'mine_slproject_f14.slx'. The interface is divided into several sections:

- Navigation and Controls:** Includes 'Next' and 'Previous' buttons, a 'Linked scrolling' checkbox, and buttons for 'Top Model', 'Bottom Model', 'Highlight Now', and 'Always highlight in models'. There are also 'Filter' and 'Accept & Close' options.
- Model Comparison:** Three columns represent different model versions:
 - Theirs:** ID: 34a6598997685c06df4571edf5a31bff2e6c7f
 - Base:** ID: 5268240c99c699aff23ab83fcf4ab5fa4a9eafd4
 - Mine:** (Current model)
- Model Hierarchy:** Each column shows a tree view of model components including 'Model Configuration Sets', 'Configuration', 'Diagnostics', 'Solver', 'Model Hierarchy', 'Simulink', 'Pilot', 'Pilot response amplitude', 'StickCommand_rad', 'Pilot:1 -> Bus Creator:1', 'Pilot:1 -> Branch', and 'EditorSettings'. Red highlights indicate differences between the models.
- Target View:** Located at the bottom, it shows the merged model's hierarchy with radio buttons for selecting the source of each element. A red exclamation mark icon indicates unresolved conflicts.
- Conflict Resolution Panel:** A pop-up window titled 'Resolve remaining 5 conflicts' is visible on the right. It shows a 'Filtered View (3)' of conflicts, including 'Conflict', 'Conflicted manual merge', and 'Manual merge'. A 'Total' row is also present.

Report Generation

Bookmarks

- Table of Contents
- Chapter 1. Model Version
- Chapter 2. Root System
- Chapter 3. Subsystems
- Chapter 4. System Design Variables
- Chapter 5. Requirements Traceability
- Chapter 6. System Model Configuration
- Chapter 7. Glossary
- Chapter 8. About this Report

Chapter 2. Root System

Table of Contents

Description

Blocks

Parameters

Block Execution Order

Figure 2.1. CruiseControl_harness

Description

The cruise controller was designed with Stateflow. To test the controller, we use a harness setup test vectors imported from Excel, and the outputs compared to expected results.

Report Generation

Bookmarks

- Table of Contents
- Chapter 1. Model Version
- Chapter 2. Root System**
- Chapter 3. Subsystem s
- Chapter 4. System Design Variables
- Chapter 5. Requirements Traceability
- Chapter 6. System Model Configuration
- Chapter 7. Glossary
- Chapter 8. About this Report

Chapter 2. Root System

Table of Contents

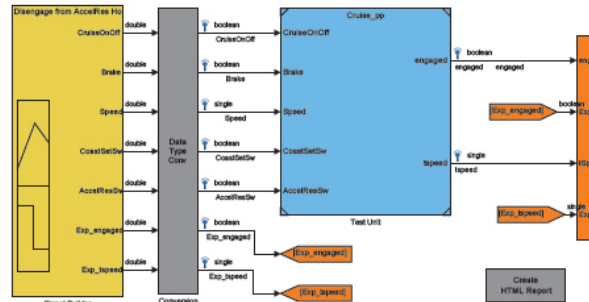
Description

Blocks

Parameters

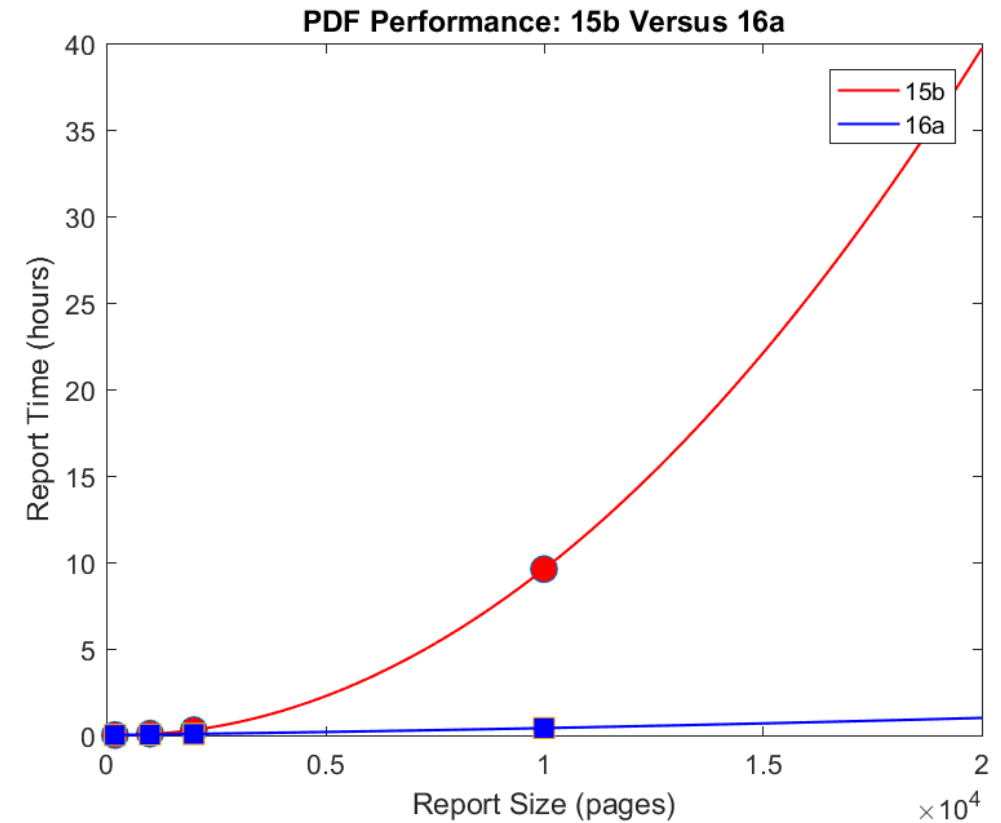
Block Execution Order

Figure 2.1. CruiseControl_harness

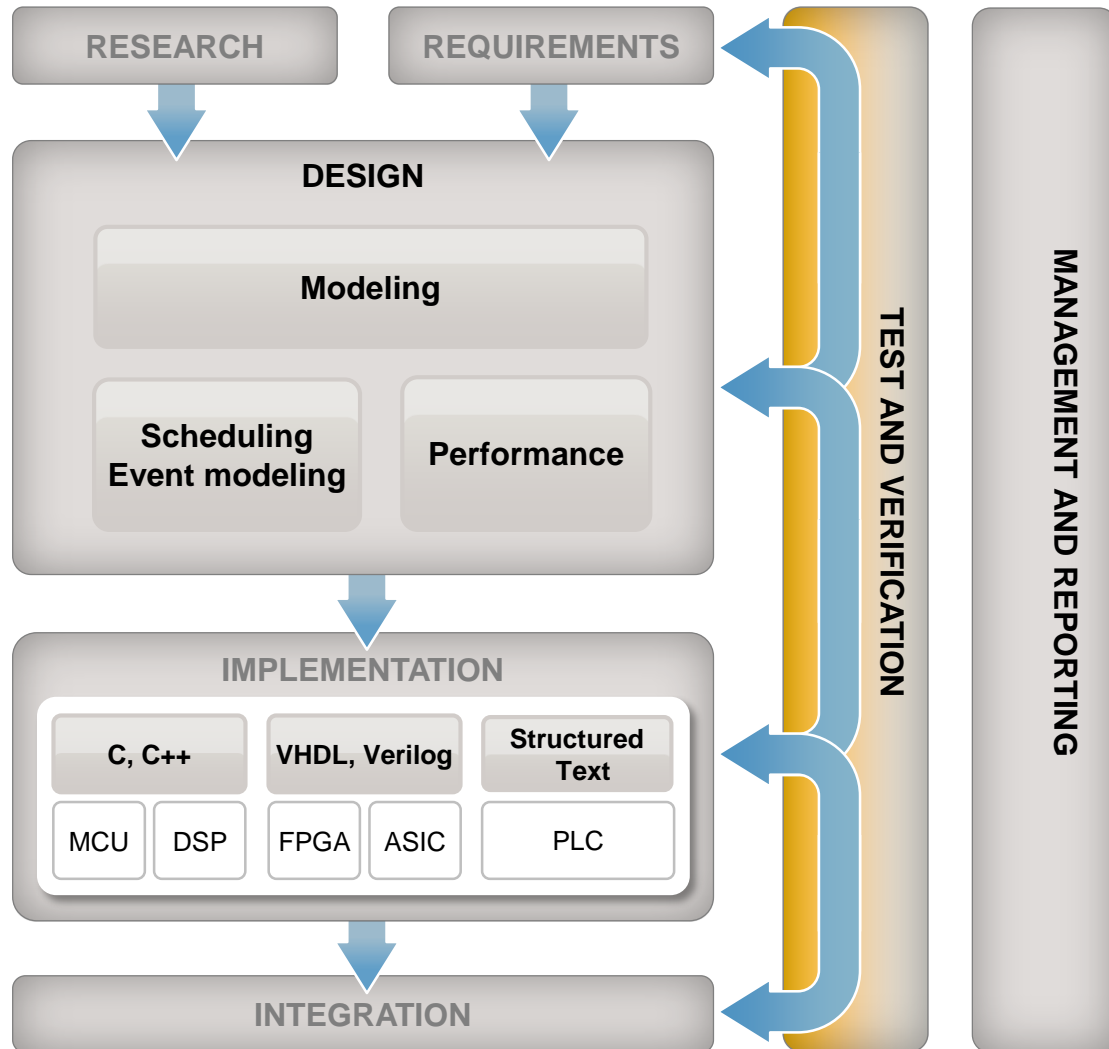


Description

The cruise controller was designed with Stateflow. To test the controller, we use a harness setu test vectors imported from Excel, and the outputs compared to expected results.

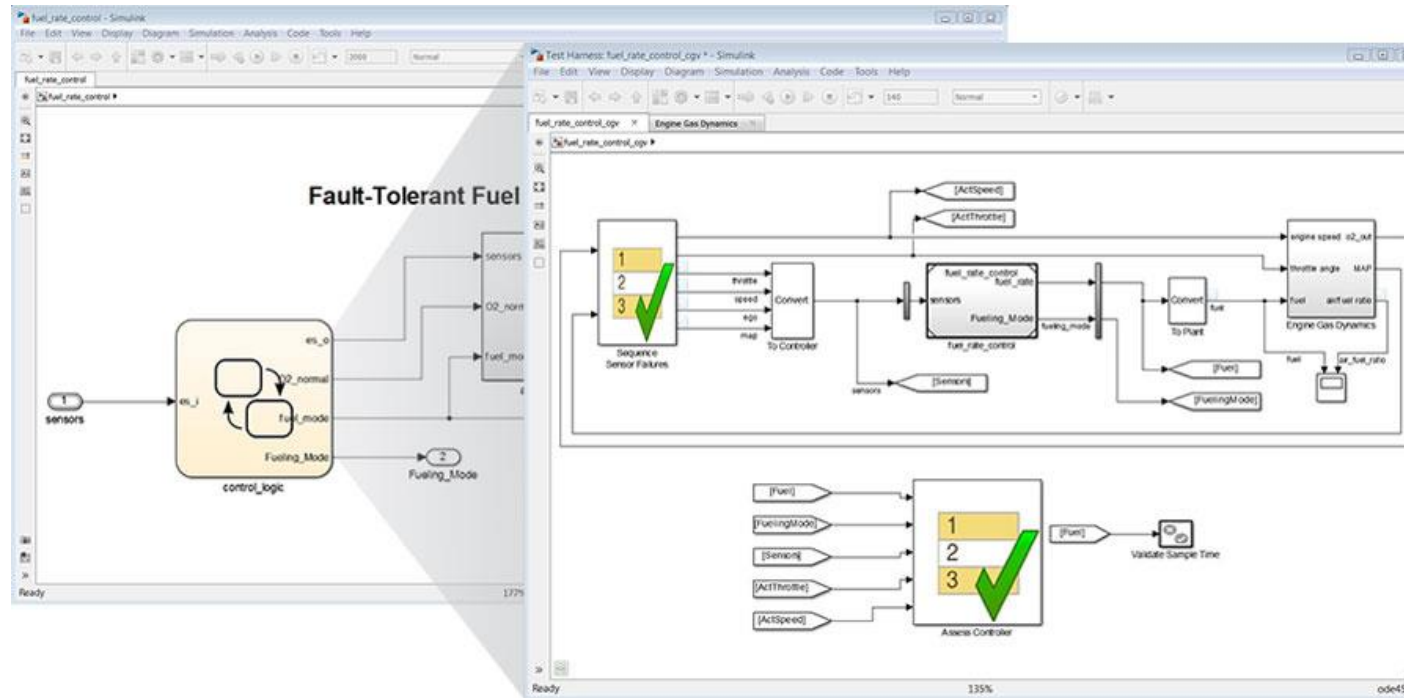


Model-Based Design Workflow



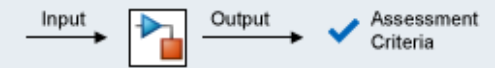
New Product! Simulink Test

Develop, Manage, and execute simulation-based tests

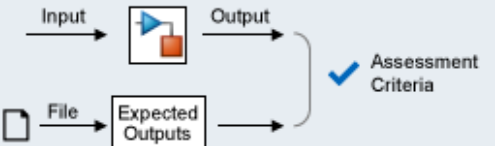


Test Case Templates

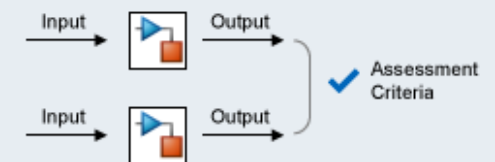
Simulation Test



Baseline Test

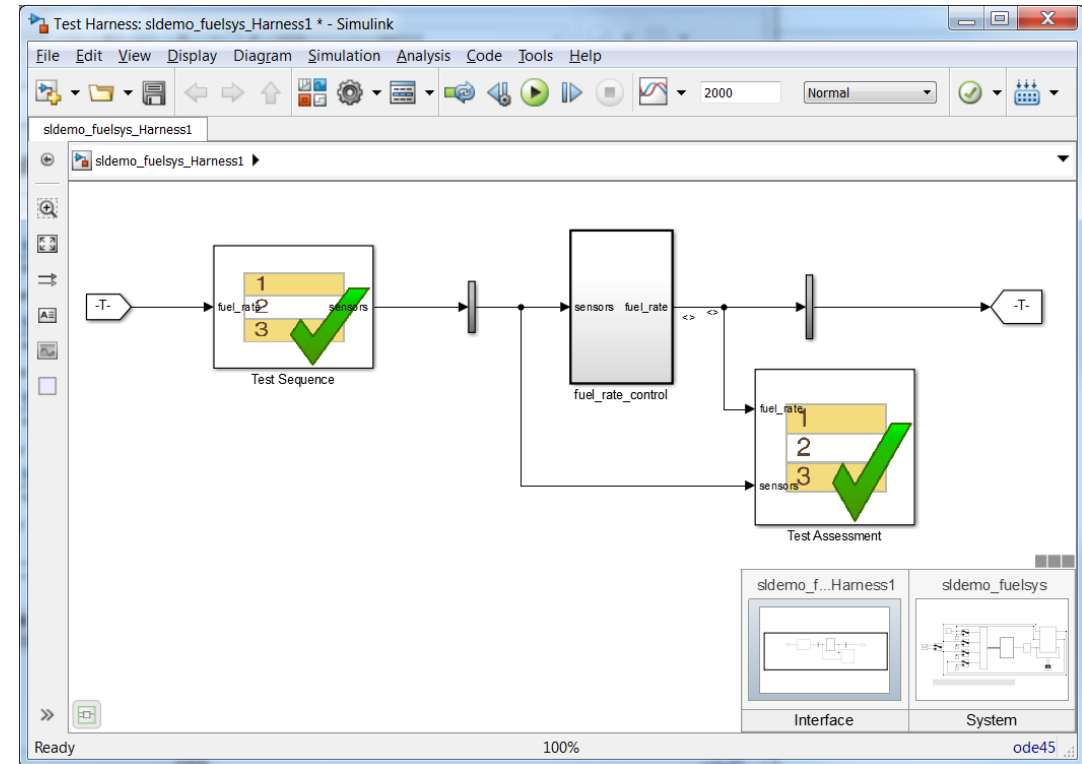


Equivalence Test



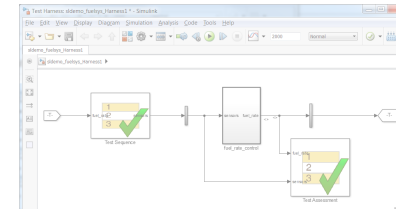
New Product! Simulink Test

- Automatically generate Test Harness

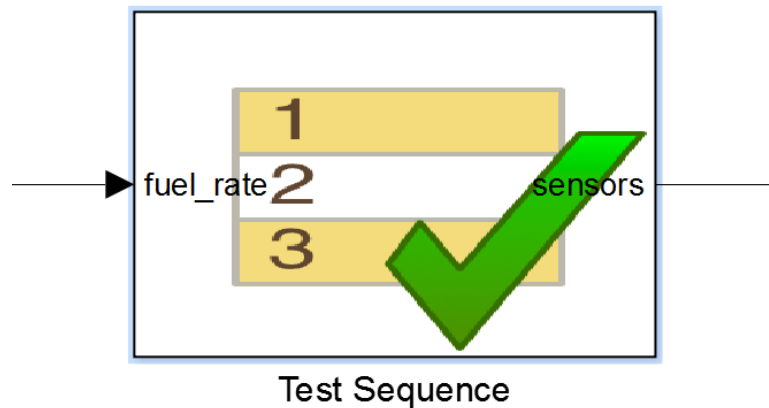


New Product! Simulink Test

- Automatically generate Test Harness



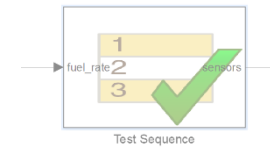
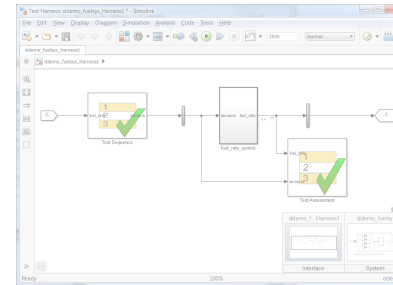
- Test Sequence block



Step	Transition	Next Step
<pre>initialize on_off = false; Tproj = single(0);</pre>	1. true	Normal_on_off ▼
<pre>Normal_on_off end_test = 0;</pre>		
<pre>On on_off = true;</pre>	1. FanOn == true	Wait ▼
<pre>Wait on_off = false; verify(FanOn == true,... 'Simulink:verify_scenario1',... 'Fan should be active');</pre>	1. after(20,sec)	Off ▼
<pre>Off on_off = true;</pre>	1. FanOn == false	End ▼
<pre>End on_off = false; end_test = 1;</pre>		

New Product! Simulink Test

- Automatically generate Test Harness
- Create Test Sequences
- Manage and Reporting

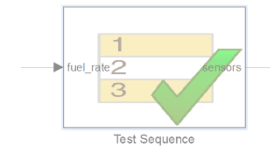
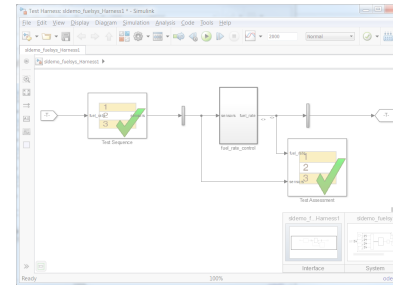


Step	Transition	Next Step
Initialize on_off = false; Tproj = single(0);	1. true	Normal_on_off
Normal_on_off end_test = 0;	1. FanOn == true	Wait
Wait on_off = false; verify(FanOn == true,... "Simulink.verify_scenario1"... "Fan should be active");	1. after(20_sec)	Off
Off on_off = true;	1. FanOn == false	End
End on_off = false; end_test = 1;		

NAME	STATUS
Results : 2014-Dec-10 10:41:08	1 ✓ 1 ✗
Test File	1 ✓ 1 ✗
Test Suite	1 ✓ 1 ✗
Baseline Test Case	✗
Baseline Criteria Result	✗

New Product! Simulink Test

- Automatically generate Test Harness
- Create Test Sequences
- Manage and Reporting



Step	Transition	Next Step
Initialize on_off = false; Tproj = single(0);	1. true	Normal_on_off
Normal_on_off end_test = 0;		
On on_off = true;	1. FanOn == true	Wait
Wait on_off = false; verify(FanOn == true, ... "Simulink.verify_scenario1", ... "Fan should be active");	1. after(20_sec)	Off
Off on_off = true;	1. FanOn == false	End
End on_off = false; end_test = 1;		

NAME	STATUS
Results : 2014-Dec-10 10:41:08	1 1
Test File	1 1
Test Suite	1 1
Baseline Test Case	
Baseline Criteria Result	

Real-Time testing with Simulink Real-Time



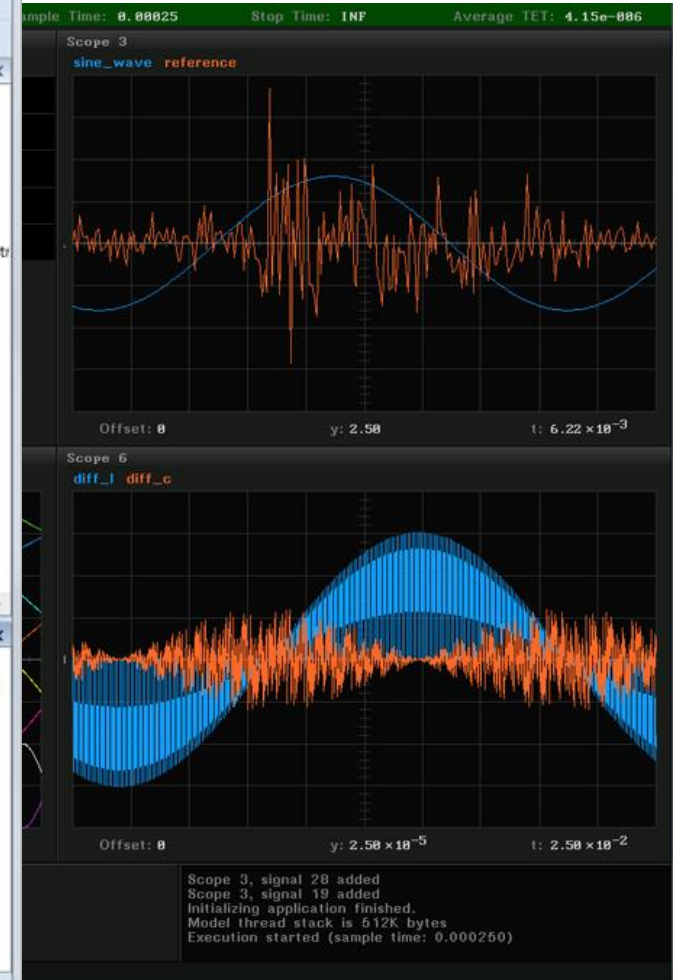
Real-Time testing with Simulink Real-Time

The Simulink Real-Time Explorer window displays the following components:

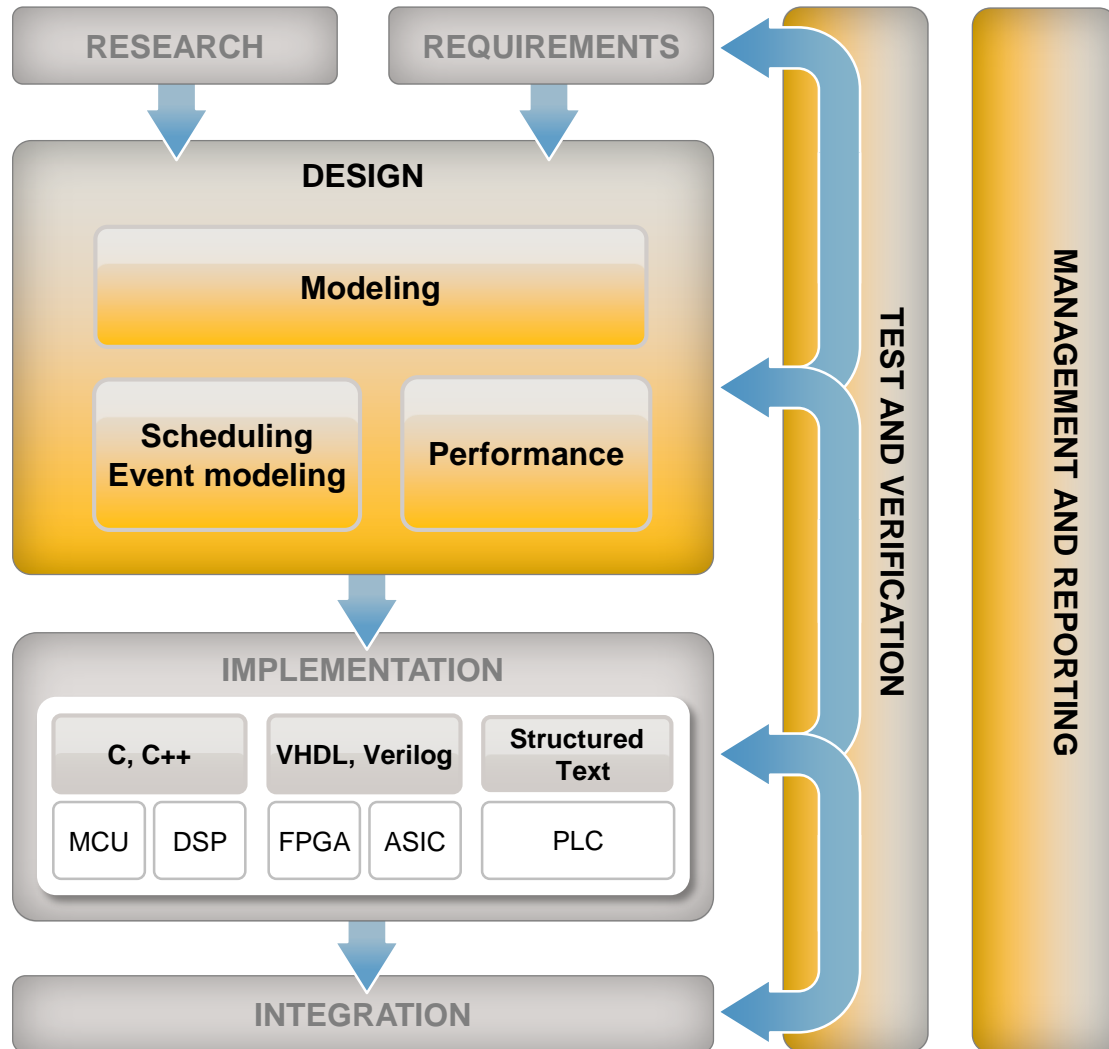
- Control Panel (Control_Panelslrtp):** A gauge labeled 'Noise' with a scale from 20 to 32. The needle points to approximately 26. A status indicator shows 'ON' with a green bar and the value '517.9'.
- TargetPC1/osc_ip:** Shows 'Mode: Real-Time Multi Tasking', 'Execution Time: 3405.83', and 'Task Execution Time: Average 7.039E-07, Maximum'. Below it is a 'Gain' block with a value of 1.
- Monitoring Table (osc_ip/Aux):**

Monitor	Signal Name	Monitoring Value	Actions	Index
<input checked="" type="checkbox"/>	Bias	490.99054	[Stop] [Refresh]	
<input checked="" type="checkbox"/>	Gain	-3903286.72489	[Stop] [Refresh]	
<input type="checkbox"/>	Gain2		[Stop] [Refresh]	
- Block List Table (osc_ip):**

Name	Is Edited	Block Name	Actions	Dimensions
Bias	False	Reference	[Stop] [Refresh]	[1, 1]
Amplitude	False	Source	[Stop] [Refresh]	[1, 1]
Bias	False	Source	[Stop] [Refresh]	[1, 1]
Frequency	False	Source	[Stop] [Refresh]	[1, 1]



Model-Based Design Workflow



Questions!

Thanks!