

# Reactis V2013

Released August 14, 2013



# New Simulink Support

- ▶ Initial support for R2013a
- ▶ Support for .slx Format
  - ▶ Introduced in R2012a to replace .mdl format
  - ▶ XML-based format
- ▶ Support Simulink *configuration set reference* concept

# Multiple Condition Coverage (MCC)

MCC metric tracks if all combinations of condition outcomes for a decision have been exercised.

# Multiple Condition Coverage (MCC)

MCC metric tracks if all combinations of condition outcomes for a decision have been exercised. For decision  $A \ \&\& \ B \ \&\& \ C$ , MCC targets are:

# Multiple Condition Coverage (MCC)

MCC metric tracks if all combinations of condition outcomes for a decision have been exercised. For decision  $A \ \&\& \ B \ \&\& \ C$ , MCC targets are:

## No Short-Circuiting

A	B	C	Decision
F	F	F	F
F	F	T	F
F	T	F	F
F	T	T	F
T	F	F	F
T	F	T	F
T	T	F	F
T	T	T	T

# Multiple Condition Coverage (MCC)

MCC metric tracks if all combinations of condition outcomes for a decision have been exercised. For decision  $A \ \&\& \ B \ \&\& \ C$ , MCC targets are:

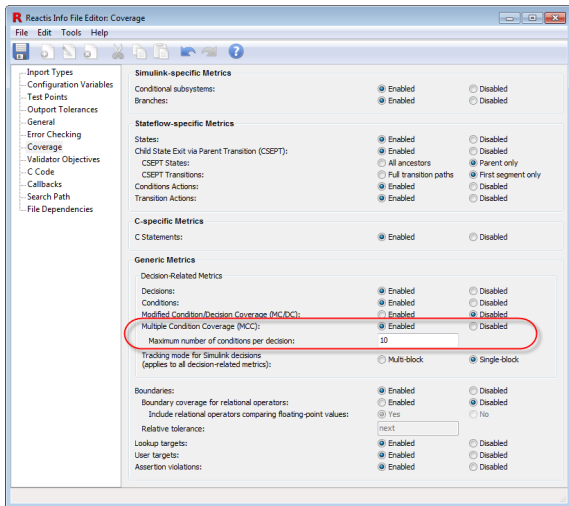
## No Short-Circuiting

A	B	C	Decision
F	F	F	F
F	F	T	F
F	T	F	F
F	T	T	F
T	F	F	F
T	F	T	F
T	T	F	F
T	T	T	T

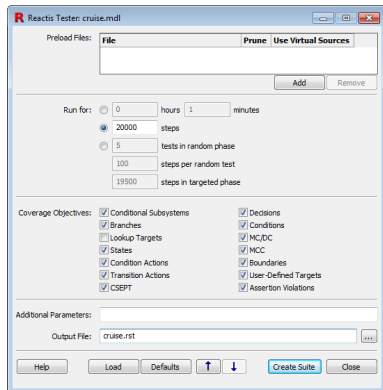
## Short-Circuiting

A	B	C	Decision
F	x	x	F
T	F	x	F
T	T	F	F
T	T	T	T

# MCC Now Supported by Reactis

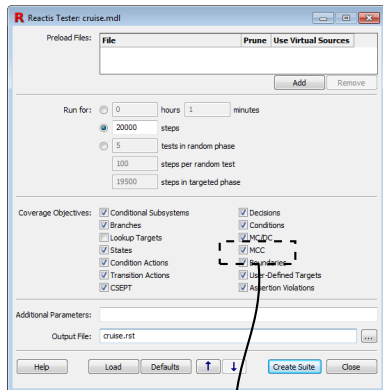


# MCC Support in Reactis Tester



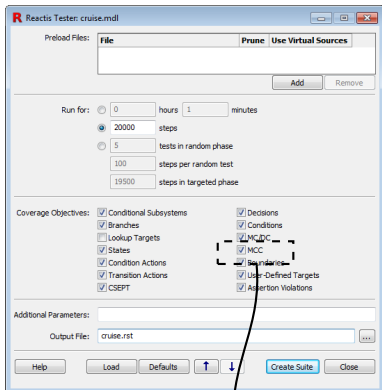


# MCC Support in Reactis Tester

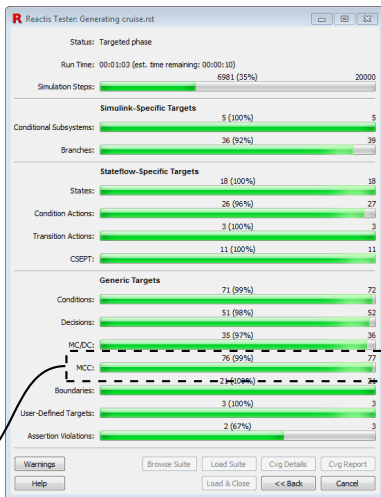


Tester attempts to cover MCC targets

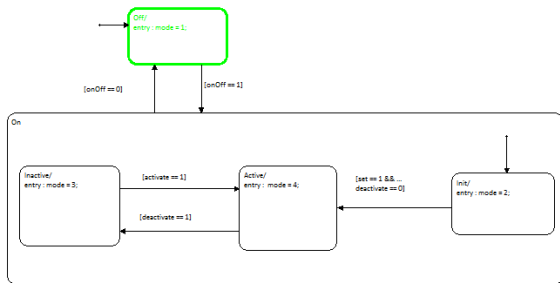
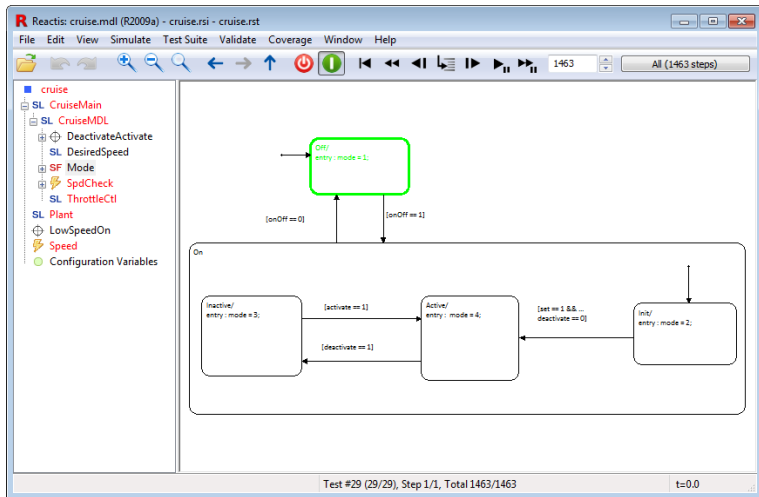
# MCC Support in Reactis Tester



Tester attempts to cover MCC targets



# Tracking MCC in Simulator



Test #29 (29/29), Step 1/1, Total 1463/1463

t=0.0

# Tracking MCC in Simulator

The screenshot displays the Reactis simulator window titled "Reactis: cruise.mdl (R2009a) - cruise.rsi - cruise.rst". The interface includes a menu bar (File, Edit, View, Simulate, Test Suite, Validate, Coverage, Window, Help), a toolbar with navigation icons, and a progress indicator showing "1463" steps out of "All (1463 steps)".

On the left, a project tree lists components: **cruise** (blue square), **SL CruiseMain** (red square), **SL CruiseMDL** (red square), **DeactivateActivate** (blue circle), **SL DesiredSpeed** (red square), **SF Mode** (red square), **SpdCheck** (yellow lightning bolt), **SL ThrottleCtl** (red square), **SL Plant** (red square), **LowSpeedOn** (blue circle), **Speed** (red square), and **Configuration Variables** (green circle).

The main workspace shows a state machine diagram. A state box at the top is highlighted with a green border and contains the text "Off/ entry : mode = 1;". It is connected to a larger state box below labeled "On" with the text "Decision with MCC targets".

The "On" state contains a sub-state box with "init/ entry : mode = 2;". A dashed box indicates a transition from the "init" state to the "Decision with MCC targets" state, labeled "[set = 1 S.S. ... deactivate = 0]".

Transitions from the "Decision with MCC targets" state are labeled "[onOff = 0]" leading to the "Off" state and "[onOff = 1]" leading back to the "On" state.

The bottom status bar shows "Test #29 (29/29), Step 1/1, Total 1463/1463" and "t=0.0".

# Tracking MCC in Simulator

The screenshot displays the Reactis simulator window for a model named 'cruise.mdl (R2009a)'. The interface includes a menu bar (File, Edit, View, Simulate, Test Suite, Validate, Coverage, Window, Help), a toolbar with navigation and simulation controls, and a progress indicator showing '1463' steps out of 'All (1463 steps)'. On the left, a project tree lists components: 'cruise', 'SL CruiseMain', 'SL CruiseMDL', 'DeactivateActivate', 'SL DesiredSpeed', 'SF Mode', 'SpdCheck', 'SL ThrottleCtl', 'SL Plant', 'LowSpeedOn', 'Speed', and 'Configuration Variables'. The main workspace shows a state machine diagram with a central state box labeled 'Decision with MCC targets'. This state has an 'On' label and is connected to other states: 'inactive/entry: mod' (left), 'mode = 1;' (top), and 'init/entry: mode = 2;' (right). Transitions are labeled with conditions: '[onOff == 0]' from 'mode = 1;' to 'inactive/entry: mod', and '[onOff == 1]' from 'inactive/entry: mod' to 'mode = 1;'. A dashed box indicates a transition from 'mode = 1;' to 'init/entry: mode = 2;' with the condition '[set == 1 && ... deactivate == 0]'. A 'Toggle Breakpoint' button is visible, with a 'View Coverage Details' button below it. The status bar at the bottom shows 'Test #29 (29/29), Step 1/1, Total 1463/1463' and 't=0.0'.

# Tracking MCC in Simulator

The screenshot displays the Reactis simulator interface. The main window shows a state machine diagram with a highlighted state: "On" with a green box containing "On/entry : mode = 1;". A transition labeled "[onOff == 1]" leads to another state. A "Toggle Breakpoint" button is visible, with a "View Coverage Details" button below it. A "Decision with MCC targets" box is overlaid on the diagram, with an arrow pointing to the "View Coverage Details" button.

The "Coverage Details" window is open, showing a table with MCC targets:

Decision		Condition	Condition		MC/DC		MC/DC	
True	False		True	False	True	False	True	False
1/30	1/4	set==1.0 deactivate==0.0	1/20	1/4	TT: 1/30	Fx: 1/4	TT: 1/30	TF: 1/20

The simulator window title is "Reactis: cruise.mdl (R2009a) - cruise.rsi - cruise.rst". The status bar at the bottom indicates "Test #29 (29/29), Step 1/1, Total 1463/1463" and "t=0.0".

# Tracking MCC in Simulator

The screenshot displays the Reactive Systems simulator interface. At the top, the window title is "ReactiveCruise.mdl (R2009a) - cruise.rsi - cruise.rst".

Two "Coverage Details" windows are open. The left window shows a table with columns for Decision (True/False), Condition, and MC/DC (TT, FF, TF, FT). The right window shows a table with columns for Decision and Covered, with a filter for "set==1.0 deactivate==0.0".

The main diagram shows a state machine with a state "On" and a "Decision with MCC targets" block. A "Toggle Breakpoint" button is visible, with a "View Coverage Details" button below it. A green box highlights the "On" state with the text "On/entry: mode = 1;".

Decision	Decision	Condition	Condition	Condition	MC/DC	MC/DC
True	False		True	False	True	False
1/30	1/4	set==1.0	1/20	1/4	TT: 1/30	FF: 1/4
		deactivate==0.0	1/30	1/20	TT: 1/30	TF: 1/20

set==1.0	deactivate==0.0	Decision	Covered
False	x	False	1/4
True	False	False	1/20
True	True	True	1/30

# Tracking MCC in Simulator

Row for each MCC target

**Coverage Details (Left Window)**

Decision		Condition		MC/DC	
True	False	True	False	True	False
1/30	1/4	set==1.0	1/20	1/4	TT: 1/30
		deactivate==0.0	1/30	1/20	Fx: 1/4
					TF: 1/20

**Coverage Details (Right Window)**

Click on column headers to enable filters

set==1.0	deactivate==0.0	Decision	Covered
False	x	False	1/4
True	False	False	1/20
True	True	True	1/30

**Decision with MCC targets**

Toggle Breakpoint  
View Coverage Details

Test #29 (29/29), Step 1/1, Total 1463/1463 t=0.0



# MCC Filtering

Decision MCC

Click on column headers to enable filters Clear Filter

set==1.0	deactivate==0.0	Decision	Covered
False	x	False	1/4
True	False	False	1/20
True	True	True	1/30

Help Close

# MCC Filtering

*Click column header*

The image shows two side-by-side screenshots of a software window titled "Coverage Details". The window has a tab labeled "Decision" and "MCC". Below the tab, there is a text prompt "Click on column headers to enable filters" and a "Clear Filter" button. The main content is a table with columns: "set==1.0", "deactivate==0.0", "Decision", and "Covered".

In the left screenshot, the table contains three rows:

set==1.0	deactivate==0.0	Decision	Covered
False	x	False	1/4
True	False	False	1/20
True	True	True	1/30

In the right screenshot, the "Decision" column header is highlighted with a mouse cursor. The table now only displays the row where "Decision" is "True":

set==1.0	deactivate==0.0	T: Decision	Covered
True	True	True	1/30

A yellow callout box with blue text says "Only display rows with *True* in column". An arrow points from this box to the "T: Decision" header in the right screenshot. Another arrow points from the "Decision" header in the left screenshot to the callout box.

# MCC Filtering

*Click column header*

R Coverage Details

Decision MCC

Click on column headers to enable filters

Clear Filter

set==1.0	deactivate==0.0	Decision	Covered
False	x	False	1/4
True	False	False	1/20
True	True	True	1/30

Help

Close

Only display rows with *True* in column

*Click column header again*

R Coverage Details

Decision MCC

Click on column headers to enable filters

Clear Filter

set==1.0	deactivate==0.0	F: Decision	Covered
False	x	False	1/4
True	False	False	1/20

Help

Only display rows with *False* in column

# MCC Considerations

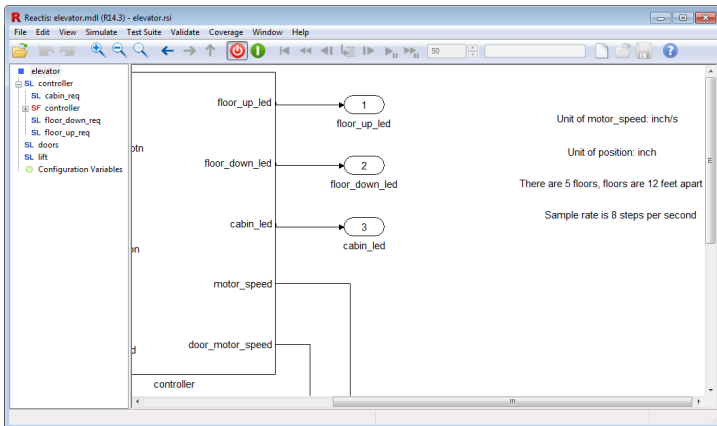
- ▶ Decision with  $n$  conditions has  $2^n$  MCC targets.
- ▶ When short-circuiting is enabled, many fewer MCC targets. Number is between MC/DC ( $n+1$ ) and  $2^n$ .
- ▶ Even if 100% MCC is not goal, MCC coverage details can be helpful in obtaining MC/DC.
- ▶ In model-specific settings (*Edit* → *Coverage...*), can set upper-bound on number of conditions in a decision; if exceeded MCC will not be tracked.

# Text Search Enhancement

Text search now searches Simulink annotations

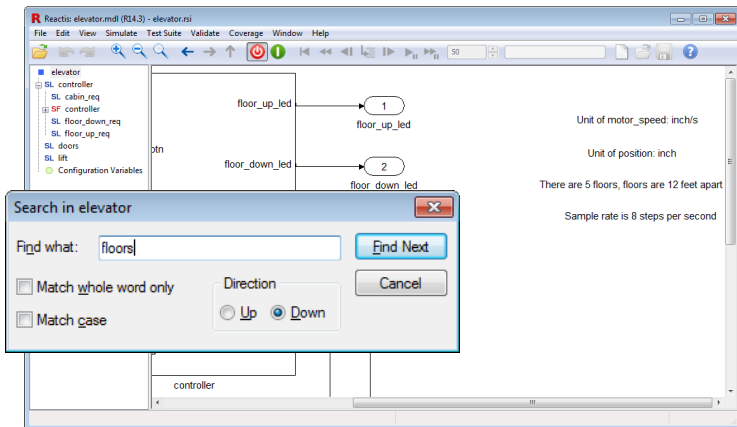
# Text Search Enhancement

Text search now searches Simulink annotations



# Text Search Enhancement

Text search now searches Simulink annotations



# Text Search Enhancement

Text search now searches Simulink annotations

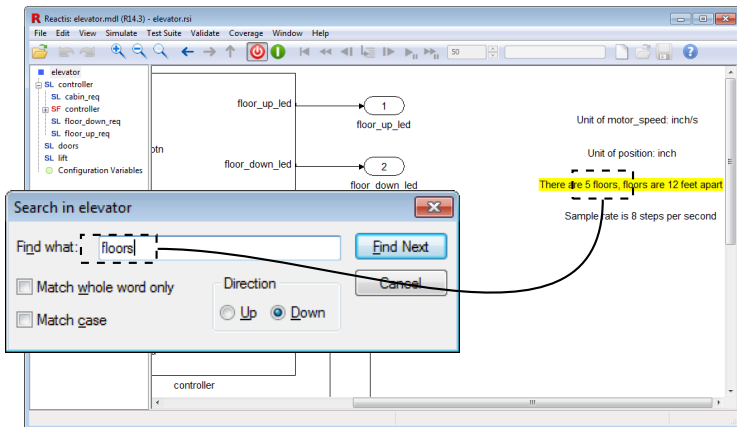
The screenshot displays the MATLAB/Simulink environment for a project named 'elevator.mdl (R14.3) - elevator.sci'. The main workspace shows a Simulink block diagram with several blocks: 'SL controller', 'SL cabin\_req', 'SF controller', 'SL floor\_down\_req', 'SL floor\_up\_req', 'SL doors', and 'SL lift'. A search dialog box is open in the foreground, titled 'Search in elevator'. The search criteria are set to 'floors', with 'Find Next' selected. The dialog also includes options for 'Match whole word only', 'Match case', and 'Direction' (set to 'Down').

In the background, the Simulink model is annotated with text. The text 'There are 5 floors, floors are 12 feet apart' is highlighted in yellow. Other annotations include 'Unit of motor\_speed: inch/s', 'Unit of position: inch', and 'Sample rate is 8 steps per second'. The diagram also shows signal lines for 'floor\_up\_led' and 'floor\_down\_led' connected to blocks labeled '1' and '2' respectively.

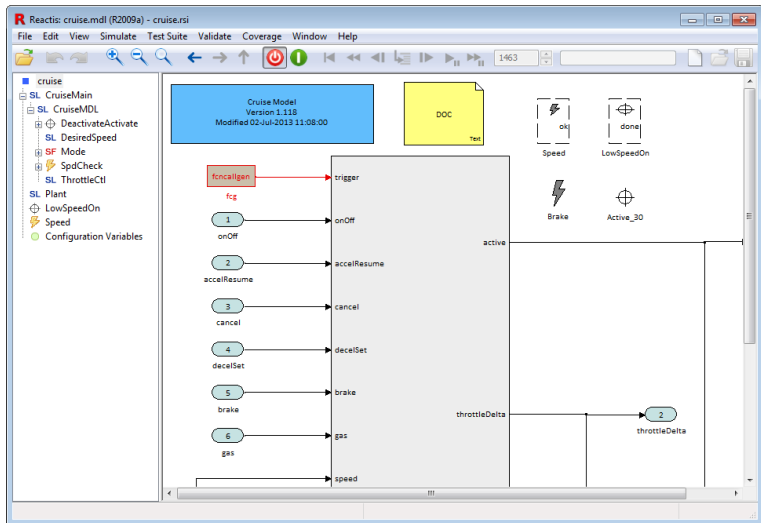


# Text Search Enhancement

Text search now searches Simulink annotations



# Display Simulink Block Colors



# Display Model Info and Doc Block Content

## Display contents of Model Info and Doc blocks

The screenshot displays the Reactis software interface for a model named 'cruise.mdl (R2009a) - cruise.rsi'. The main workspace shows a block diagram with a 'Cruise Model' block (Version 1.113, Modified 10-Apr-2013 15:34:52) and a 'DOC' block (Text). Below these, a signal flow diagram shows 'fncallgen' connected to 'trigger', with intermediate blocks 'fcg', 'onOff', 'accelResume', 'cancel', and 'decelSet'. A 'DOC block' dialog box is open, displaying the following text:

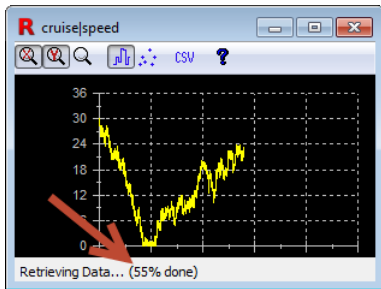
**R DOC block**

This model implements a cruise control to demonstrate the different capabilities offered by Reactis to test and validate models. Chapter 3 of the Reactis User's Guide describes how to use Reactis with this model.

The interface also includes a left-hand tree view with the following items:

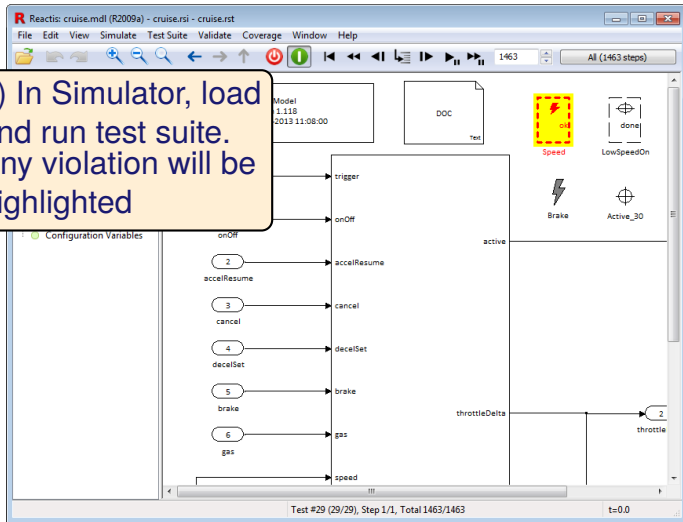
- cruise
  - SL CruiseMain
  - SL Plant
  - LowSpeedOn
  - Speed
  - Configuration Variables

# Show Progress when Opening Scope



# Run-to-Violation

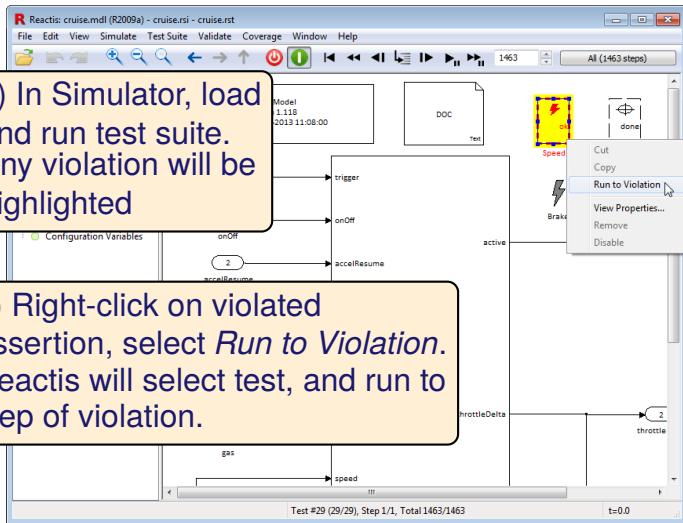
1) In Simulator, load and run test suite. Any violation will be highlighted



# Run-to-Violation

1) In Simulator, load and run test suite. Any violation will be highlighted

2) Right-click on violated assertion, select *Run to Violation*. Reactis will select test, and run to step of violation.



# Lookup Table Enhancements

- ▶ Enhancements to N-d lookup tables, Pre Lookup, Interpolation
  - ▶ Track coverage
  - ▶ Support more type combinations
- ▶ In coverage reporting, show breakpoint values

	$[-\text{inf}, c1]$	$[c1, c2]$	$[c2, c3]$	$[c3, +\text{inf}]$
$[-\text{inf}, r1]$	1/4	2/3	5/1	1/6
$[r1, r2]$	1/5	2/2	3/11	2/1
$[r2, +\text{inf}]$	1/1	1/3	1/2	1/7

Old

	$u2 < 1.0$	$u2 \geq 1.0$	$u2 \geq 10.0$	$u2 \geq 100.0$
$u1 < 1.0$	1/4	2/3	5/1	1/6
$u1 \geq 1.0$	1/5	2/2	3/11	2/1
$u1 \geq 10.0$	1/1	1/3	1/2	1/7

New

- ▶ Tester gets better coverage for lookup tables

# API Functions to Export Coverage Reports

- ▶ *rsTesterWithReport* produces a coverage report after a Tester run
- ▶ *rsSimRunSuiteWithReport* produces a coverage report after running a test suite in Simulator