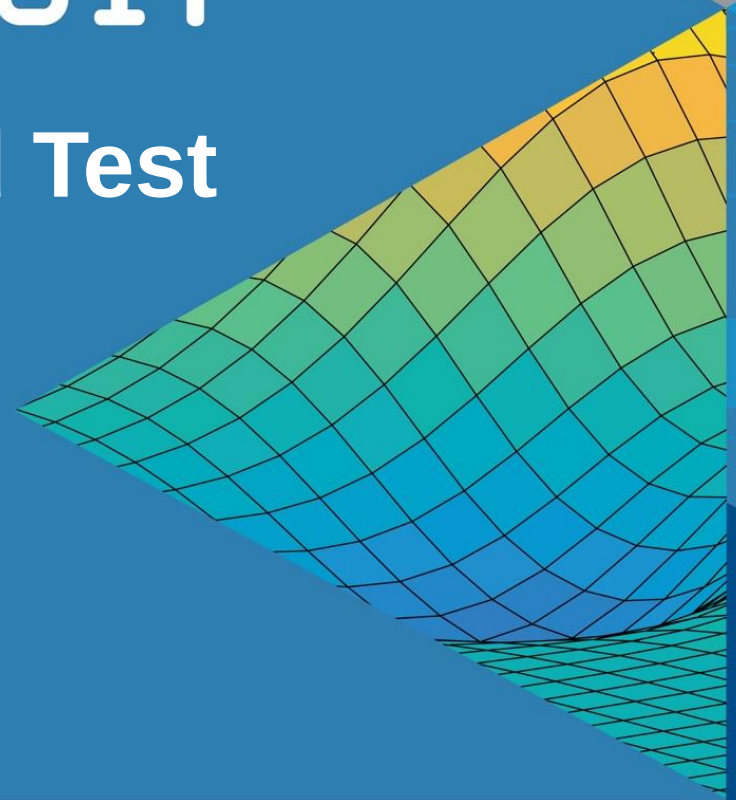
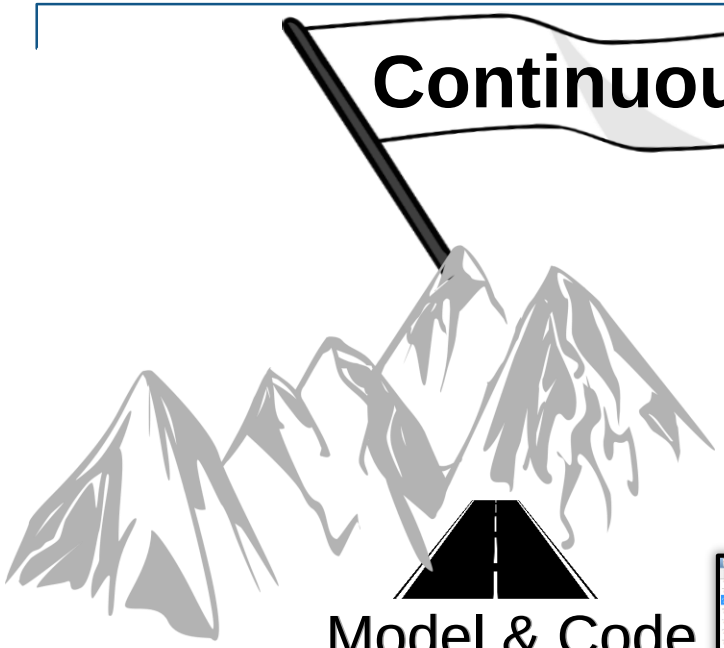
MATLAB EXPO 2017

Verification, Validation and Test in Model Based Design

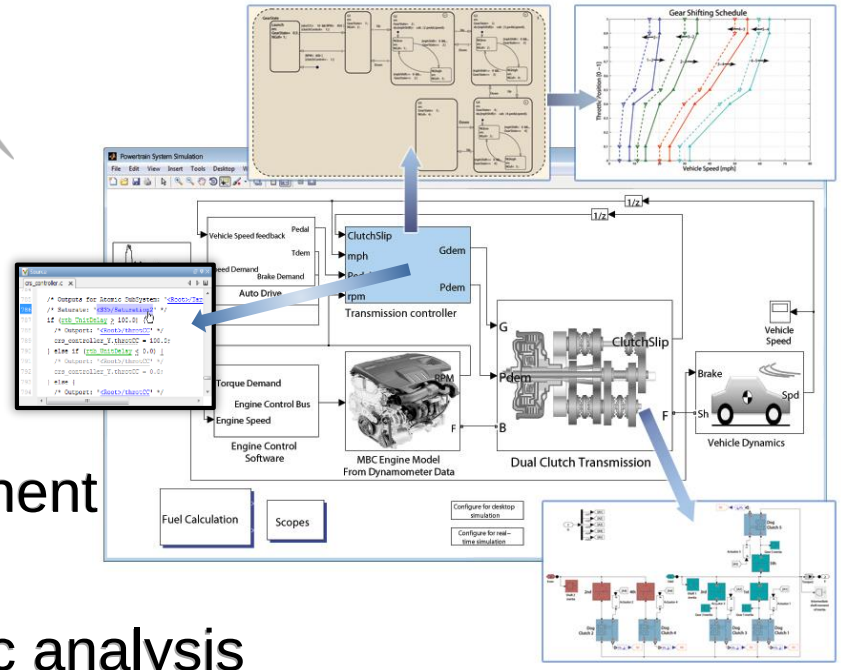
Manohar Reddy



Continuous Test & Verification

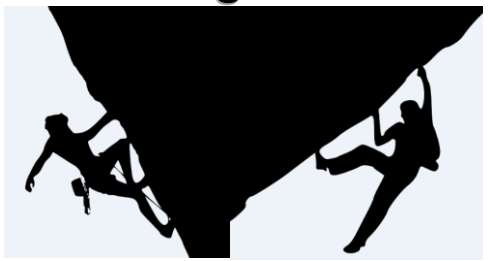


Model & Code



System & Component

Dynamic testing & Static analysis



Productivity

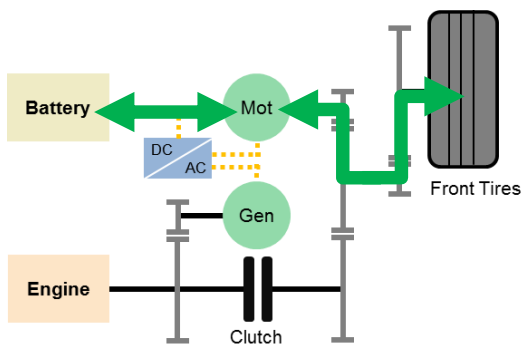
+

Quality

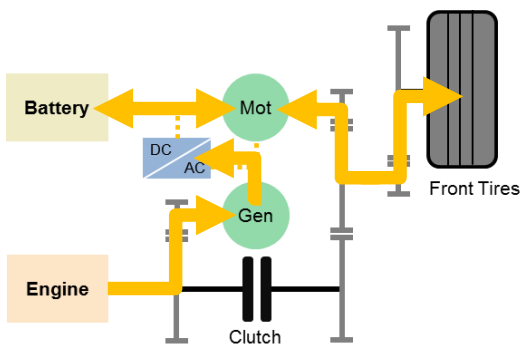
Needs verification!

continuous

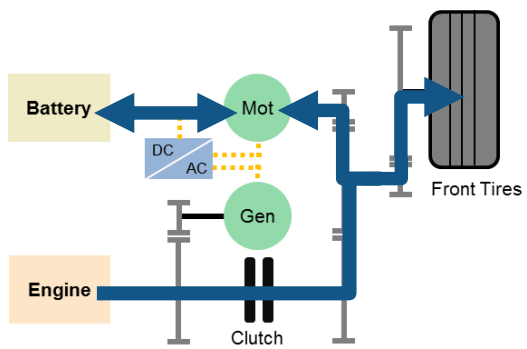
Multi-Mode Hybrid Electric Vehicle with Model Based Design



EV Mode

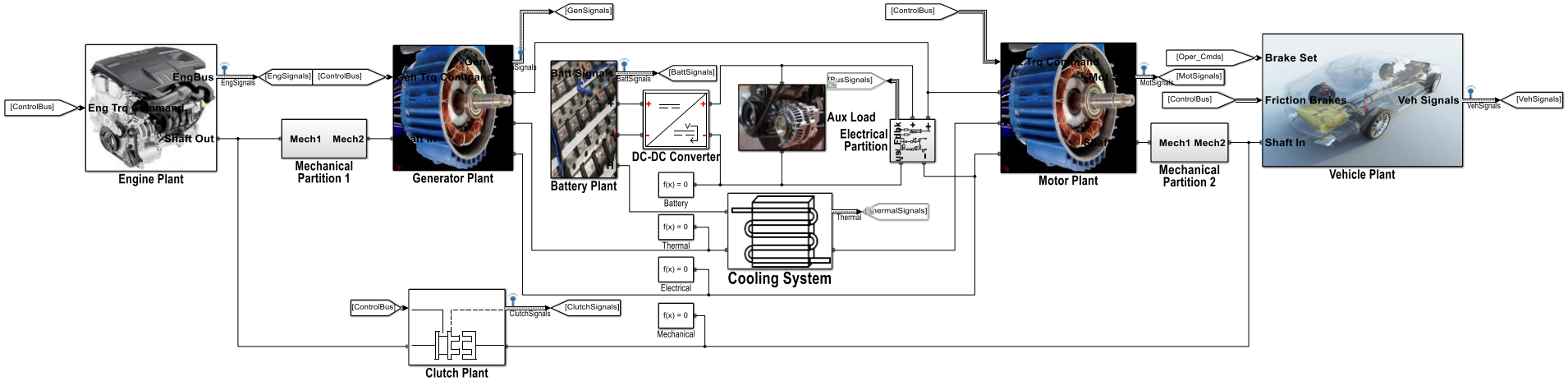


Serial HEV Mode

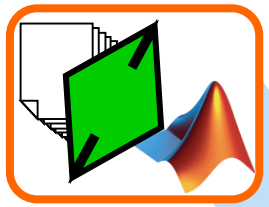


Engine Mode

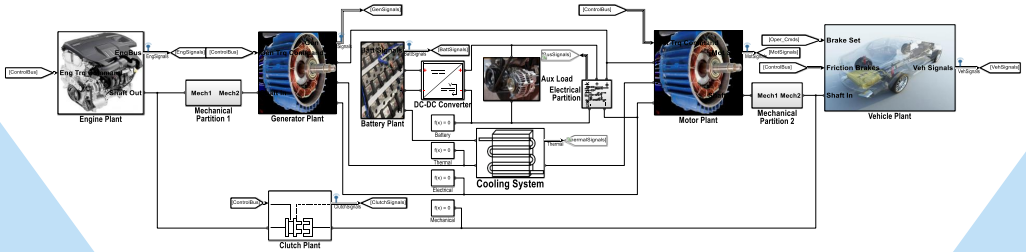
Higuchi, N., Sunaga, Y., Tanaka, M., Shimada, H.: Development of a New Two-Motor Plug-In Hybrid System, SAE 2013-01-1476 (2013)



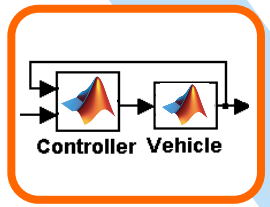
Multi-Mode Hybrid Electric Vehicle



Define Requirements

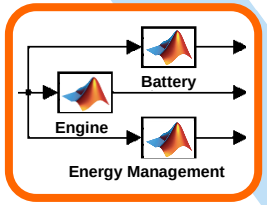


Complete Integration &Test



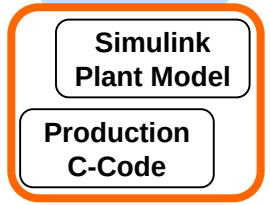
System-Level Specification

System-Level Integration & Test



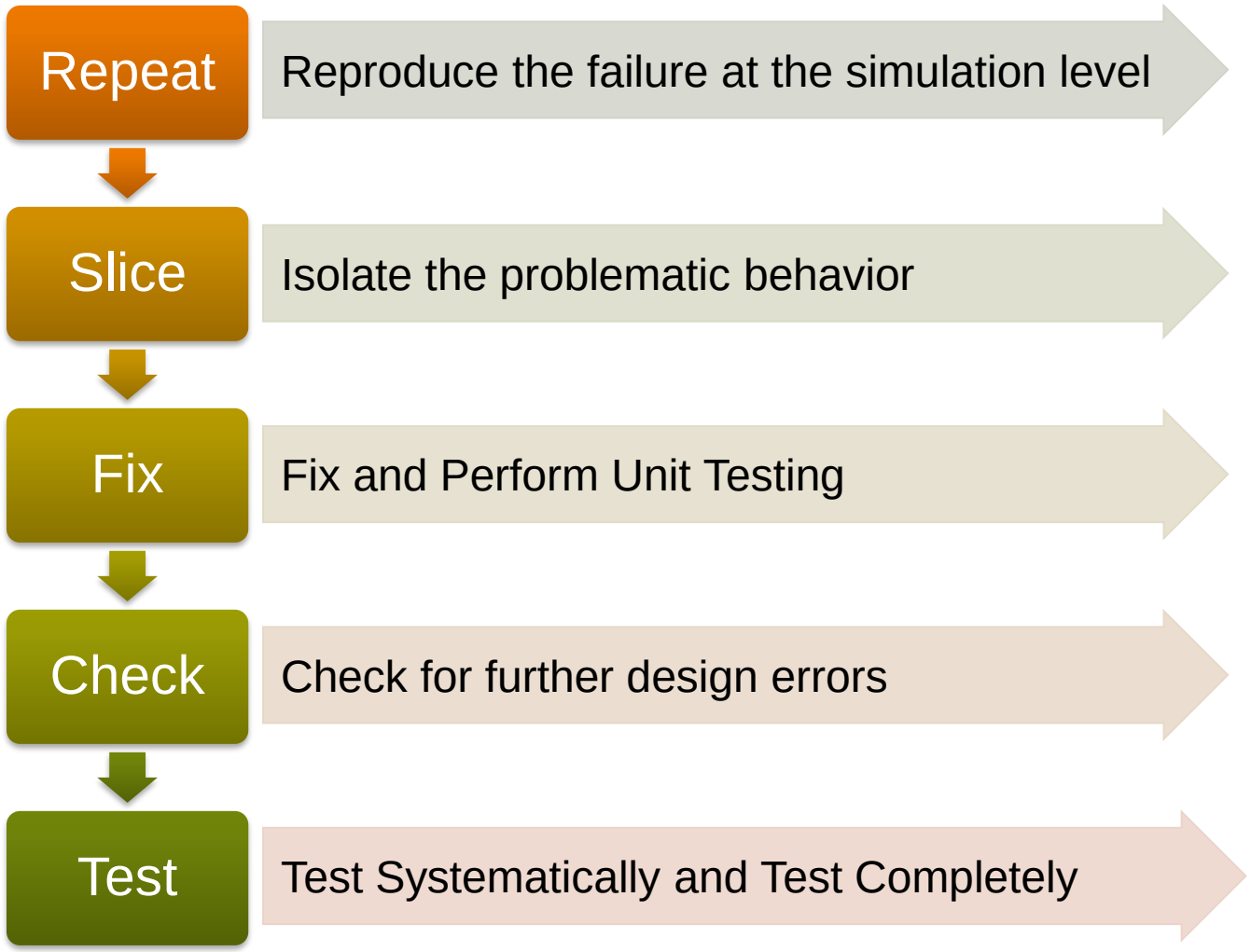
Subsystem Design

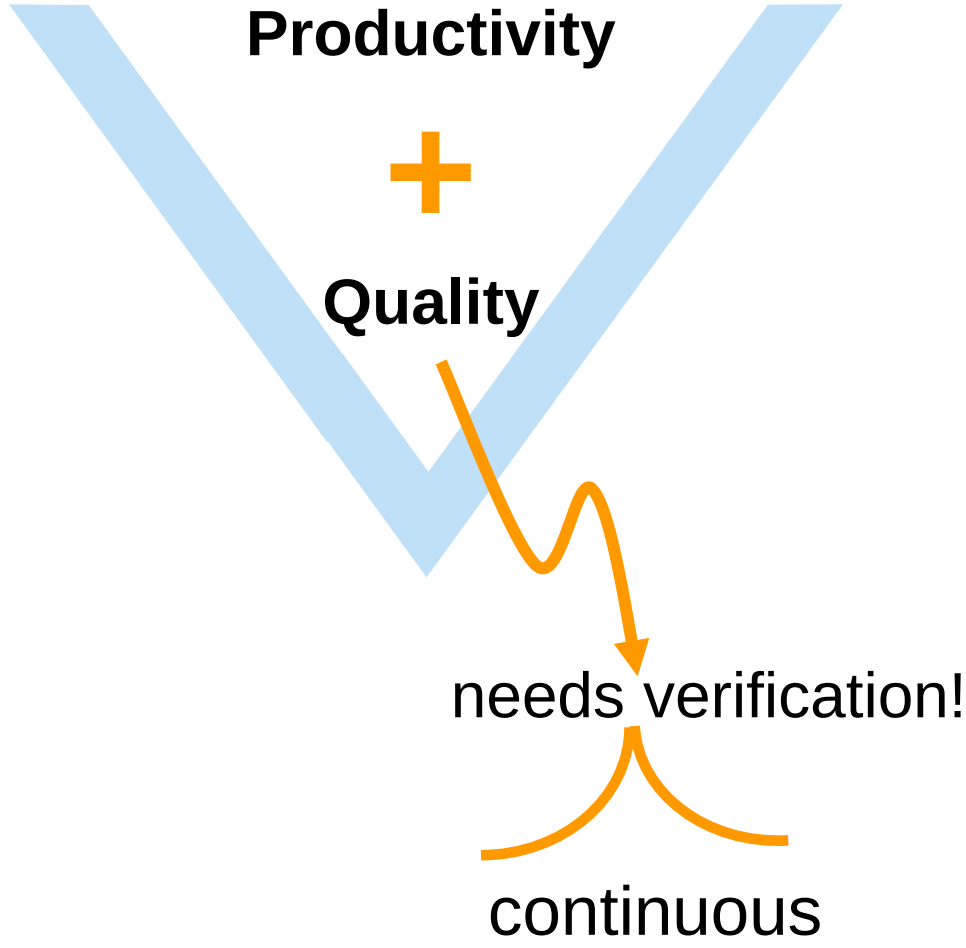
Subsystem Integration & Test



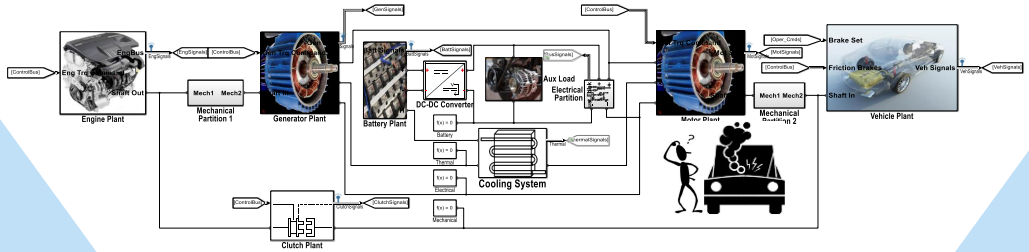
Subsystem Implementation

Continuous Test and Verification Framework

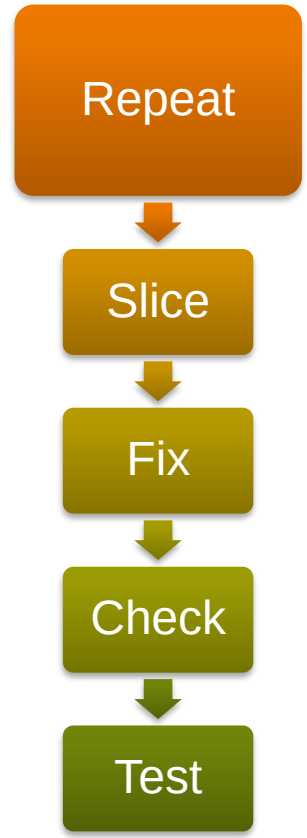




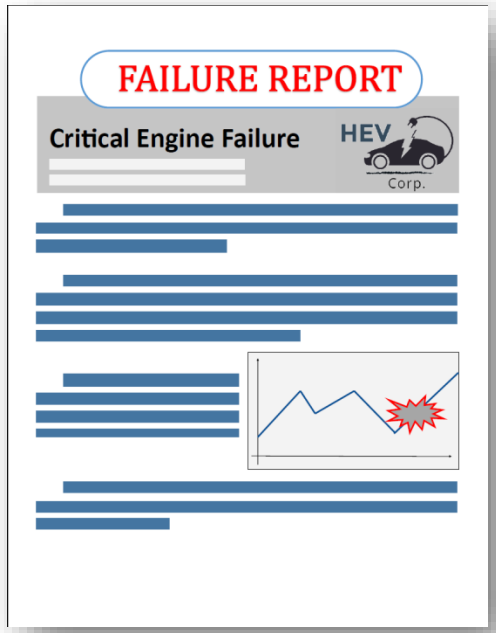
Reproduce the Failure in Simulation

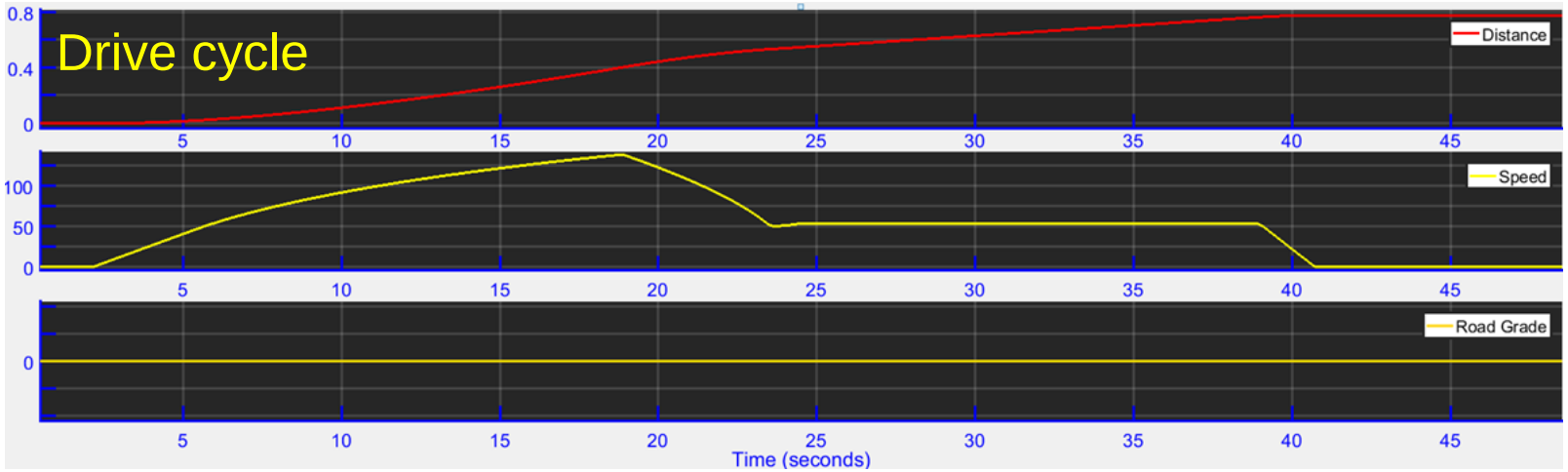


Confirm the problem In the Lab/Desktop Simulation



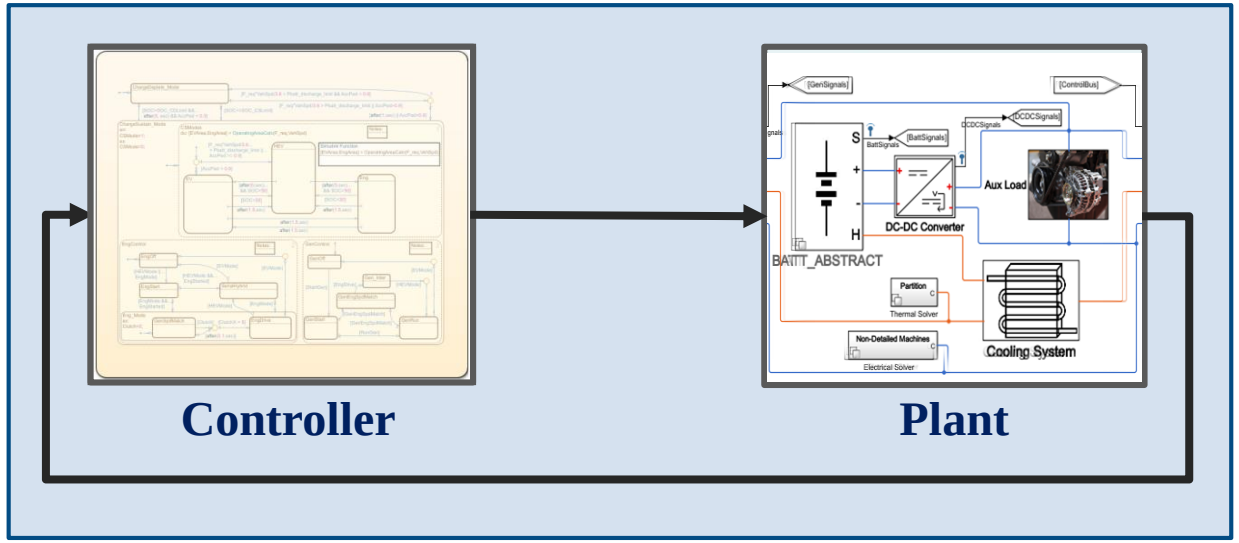
Failure Report



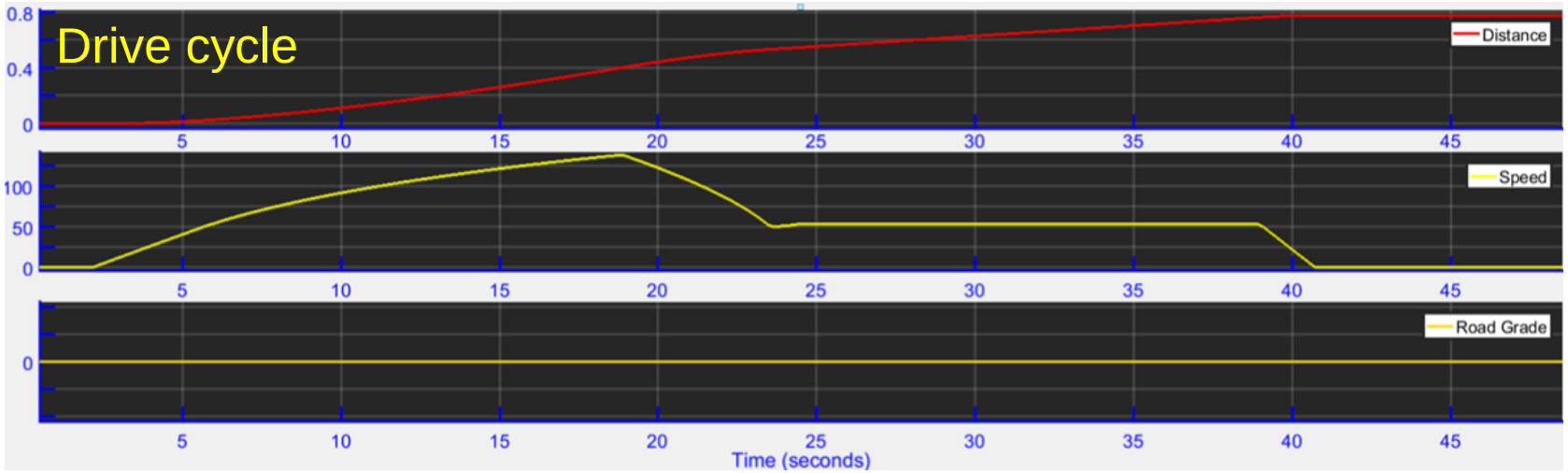


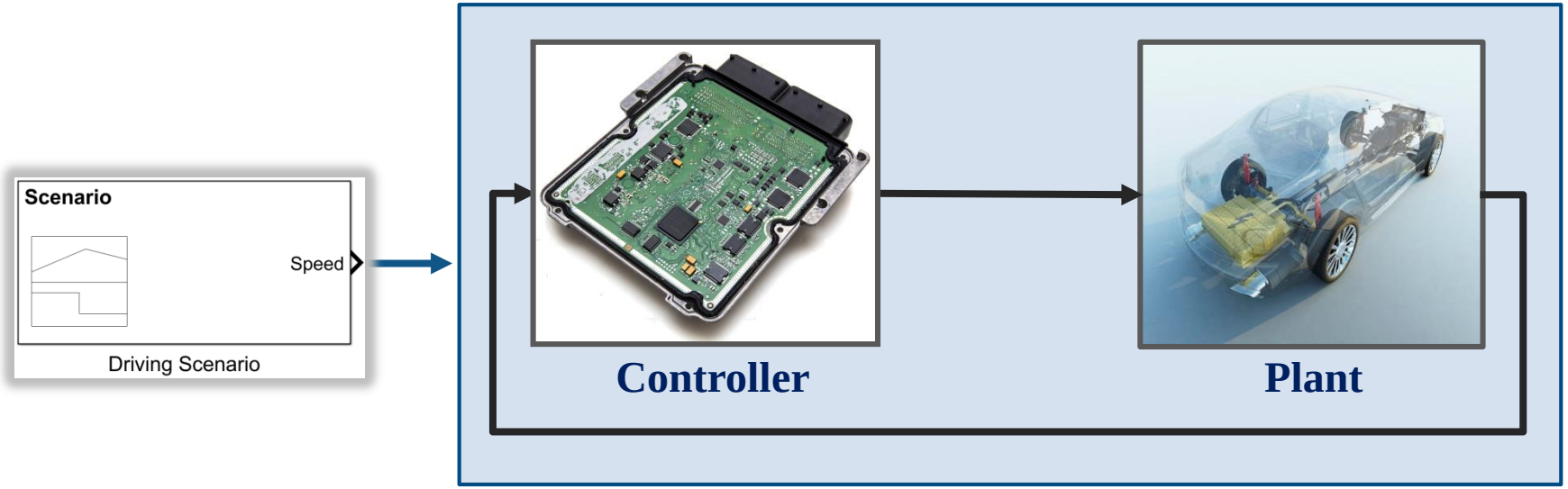
Reproduce Failure

Simulation Environment



Modeling the Test

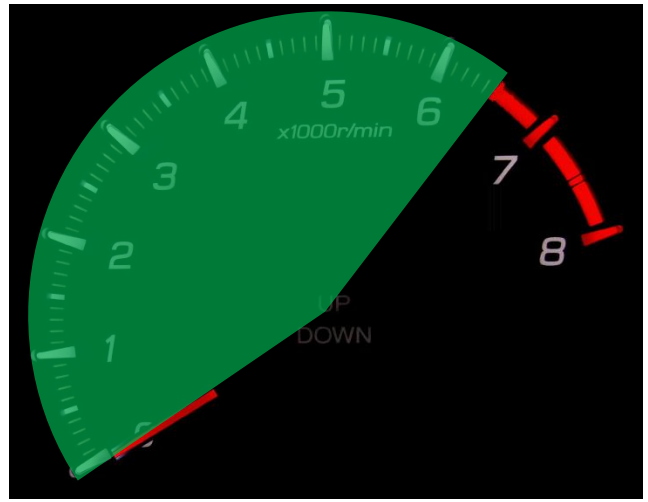




Modeling the Test

Safety Property

Engine RPM must remain within operating bounds limits



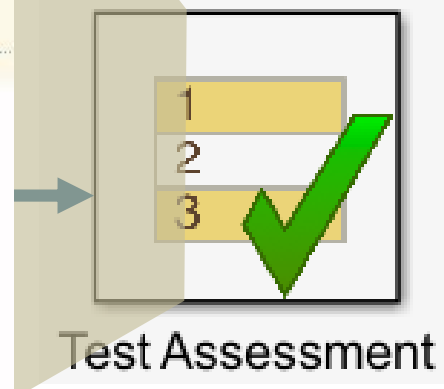
Symbols

Input

1. EngSignals
2. BattSignals
3. GenSignals
4. VehSignals
5. ClutchSignals
6. MotSignals

Step

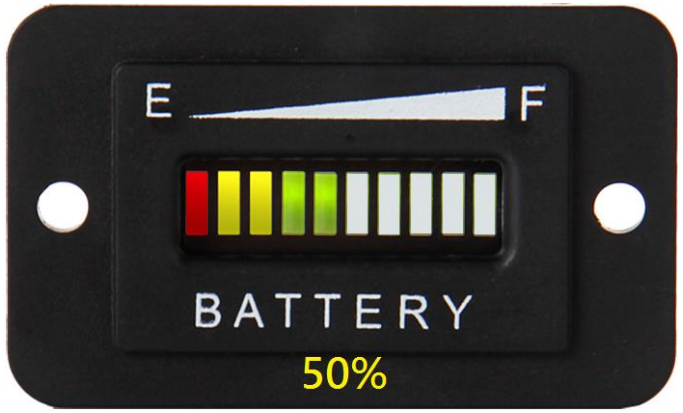
Assessment

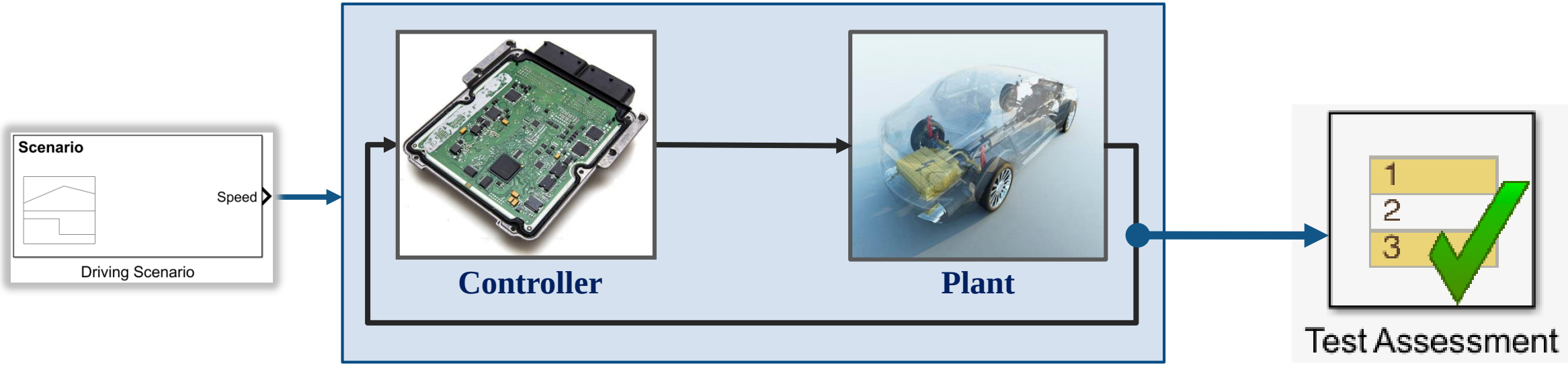


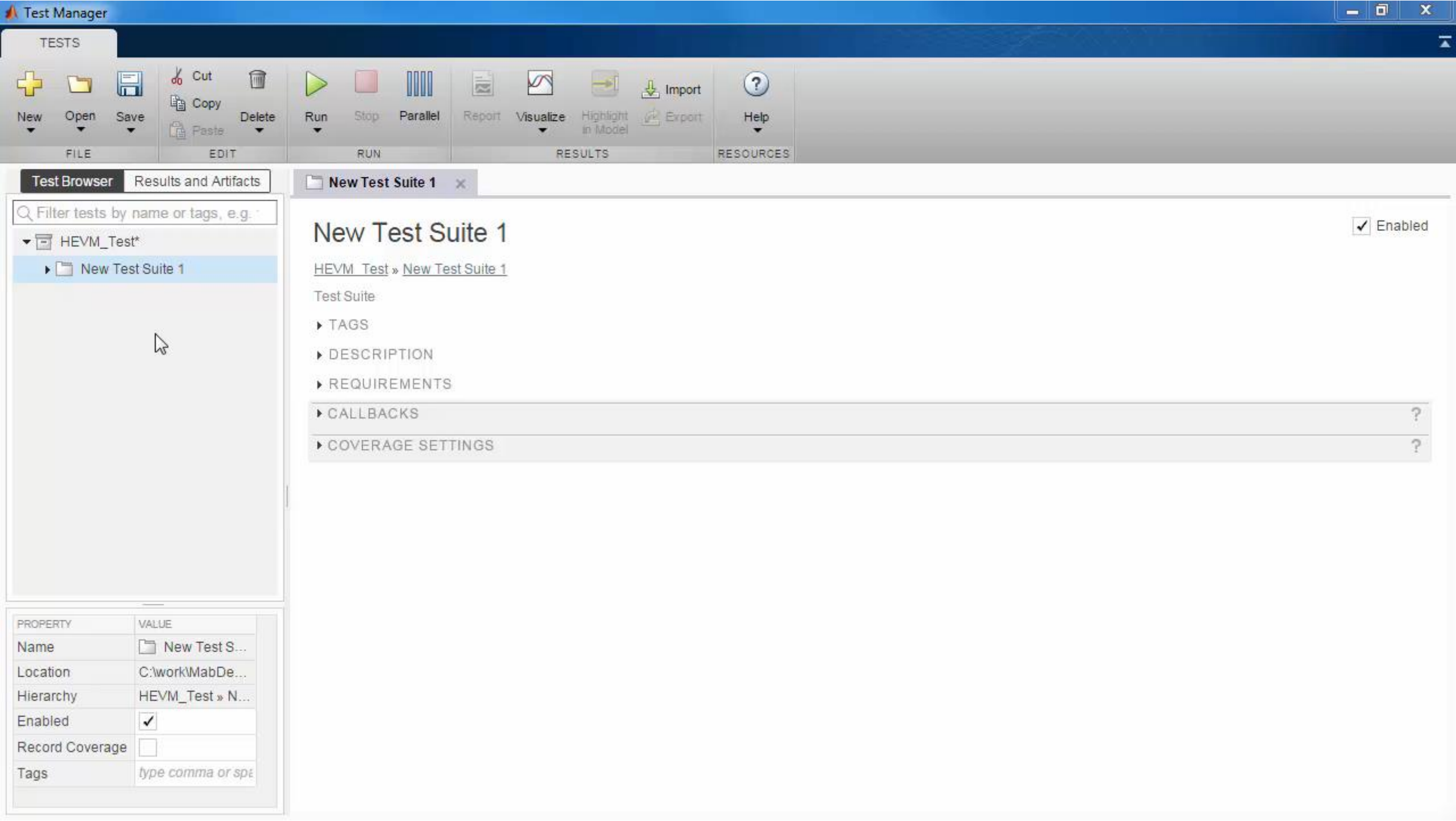
Battery State of Charge

Initial state of charge?

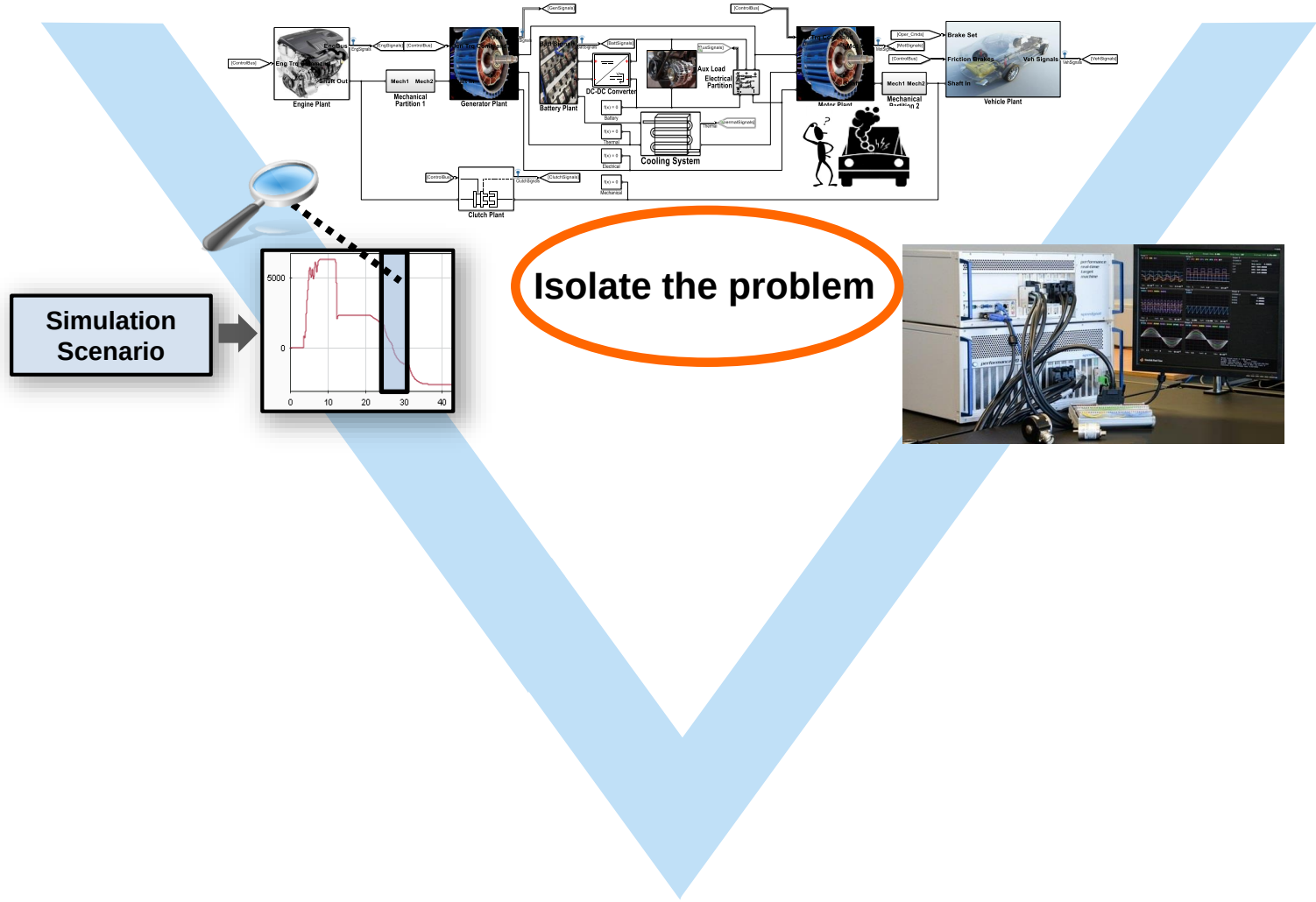
Sweep from 100% to 50%





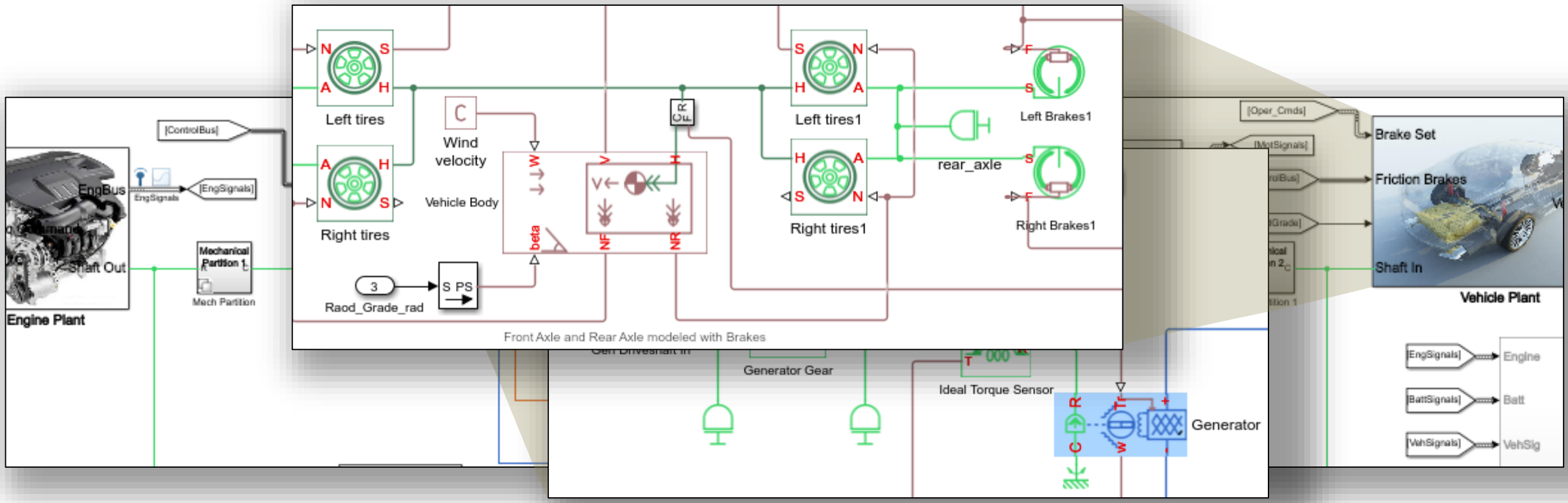

Test Manager
TESTS
New
Open
Save
Cut
Copy
Paste
Delete
Run
Stop
Parallel
Report
Visualize
Highlight in Model
Import
Export
Help
FILE
EDIT
RUN
RESULTS
RESOURCES
Test Browser
Results and Artifacts
Filter tests by name or tags, e.g.
HEVM_Test*
New Test Suite 1
PROPERTY
VALUE
Name
New Test S...
Location
C:\work\MabDe...
Hierarchy
HEVM_Test » N...
Enabled
Record Coverage
Tags
type comma or spa
New Test Suite 1
New Test Suite 1
Enabled
HEVM_Test » New Test Suite 1
Test Suite
TAGS
DESCRIPTION
REQUIREMENTS
CALLBACKS
COVERAGE SETTINGS

Model Slicing: Isolate the problematic behavior



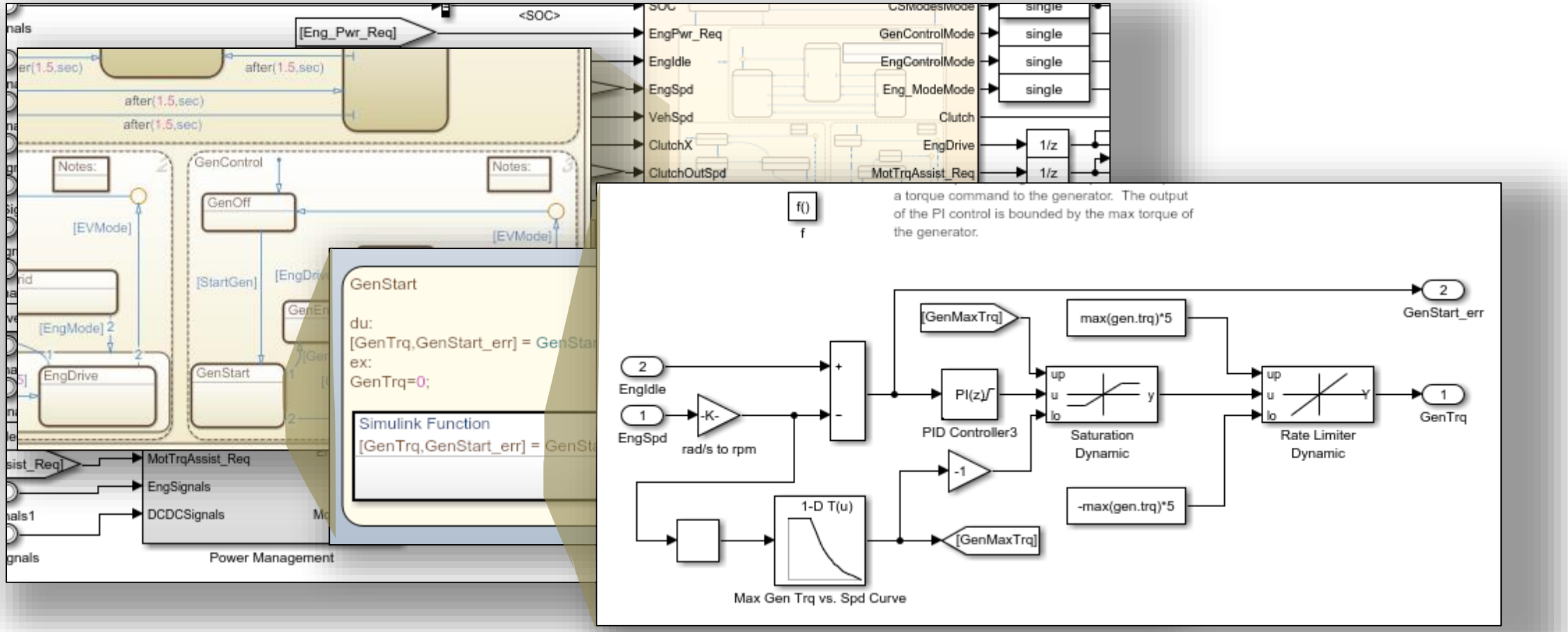
Challenge of Understanding Behavior

- Complexity of Plant



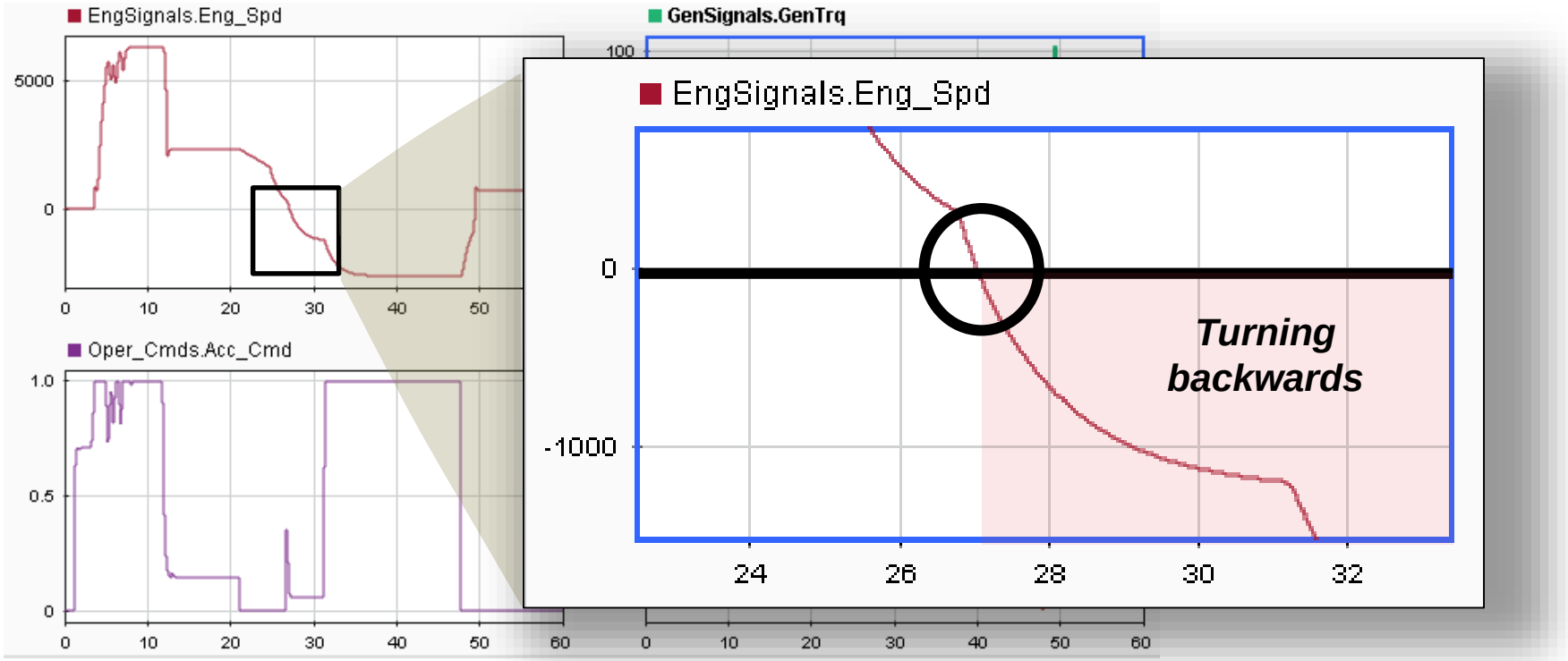
Challenge of Understanding Behavior

- Complexity of Plant
- Complexity of Controller

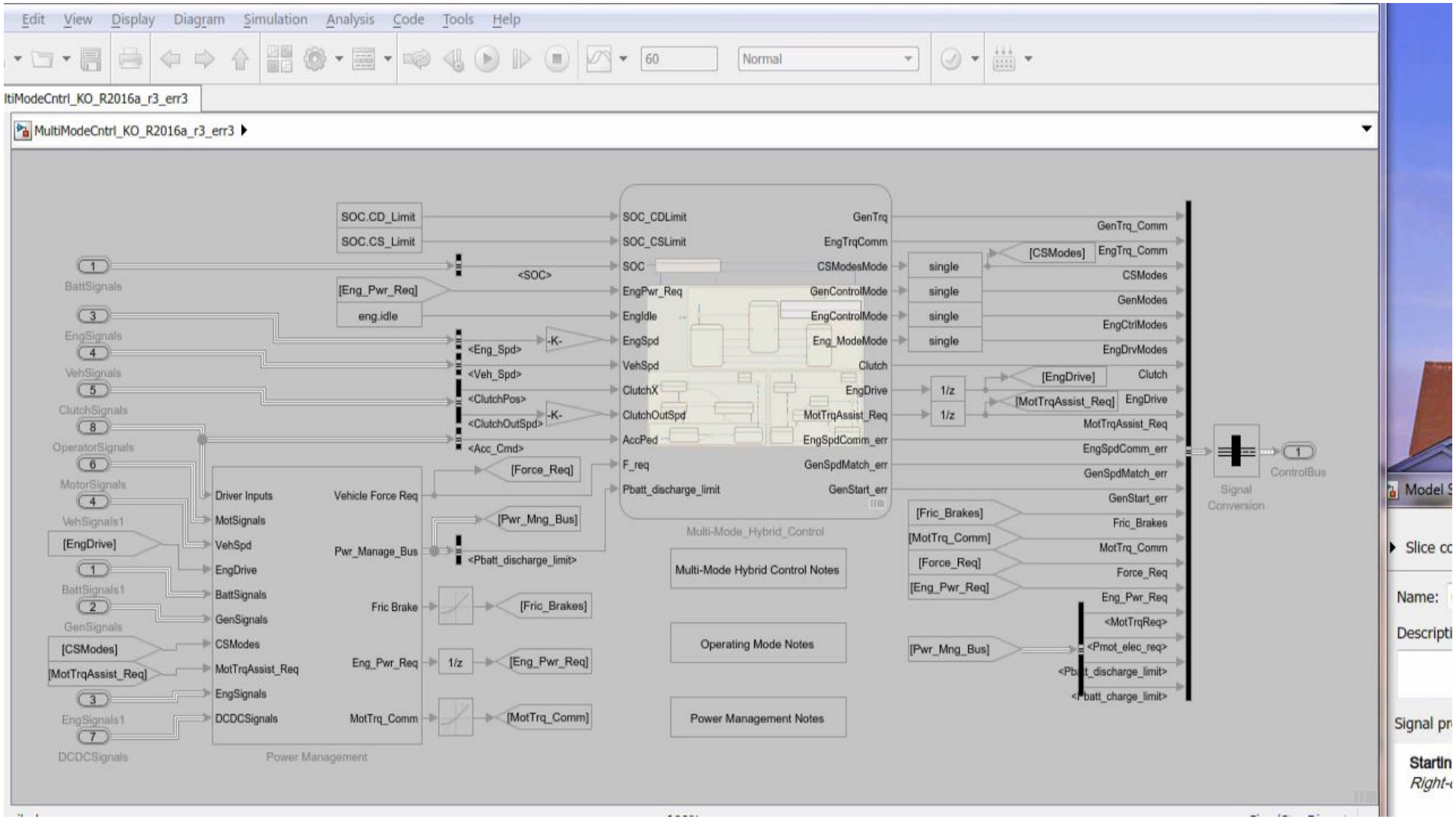


Challenge of Understanding Behavior

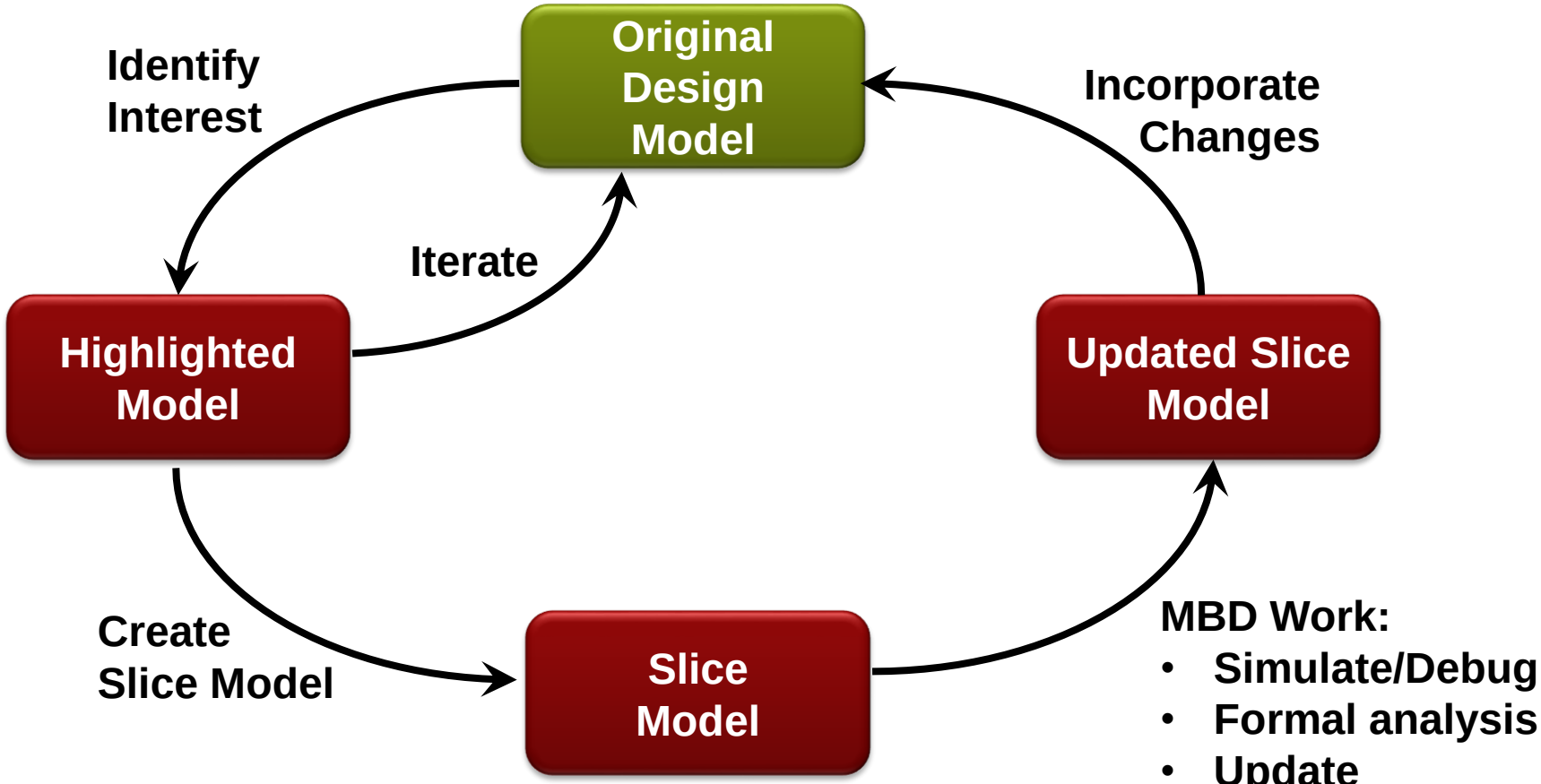
- Complexity of Plant
- Complexity of Controller
- Complexity of Dynamics



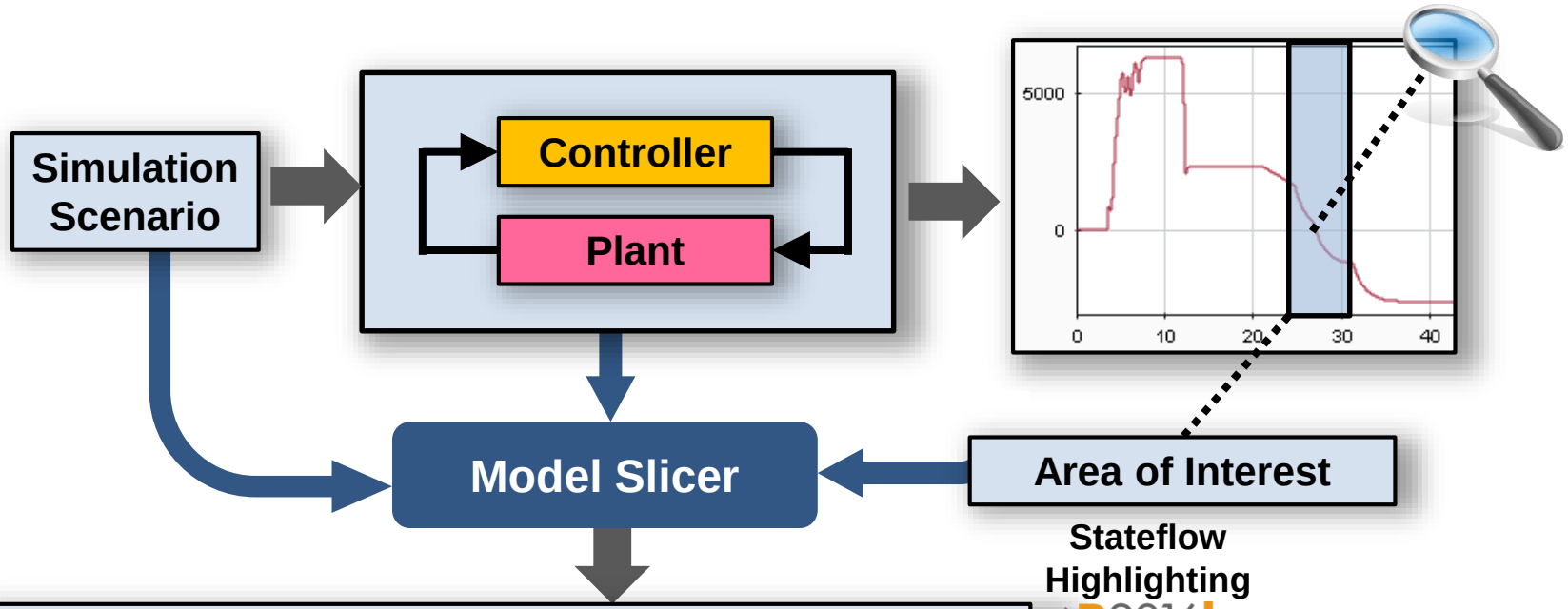
Model Slicing

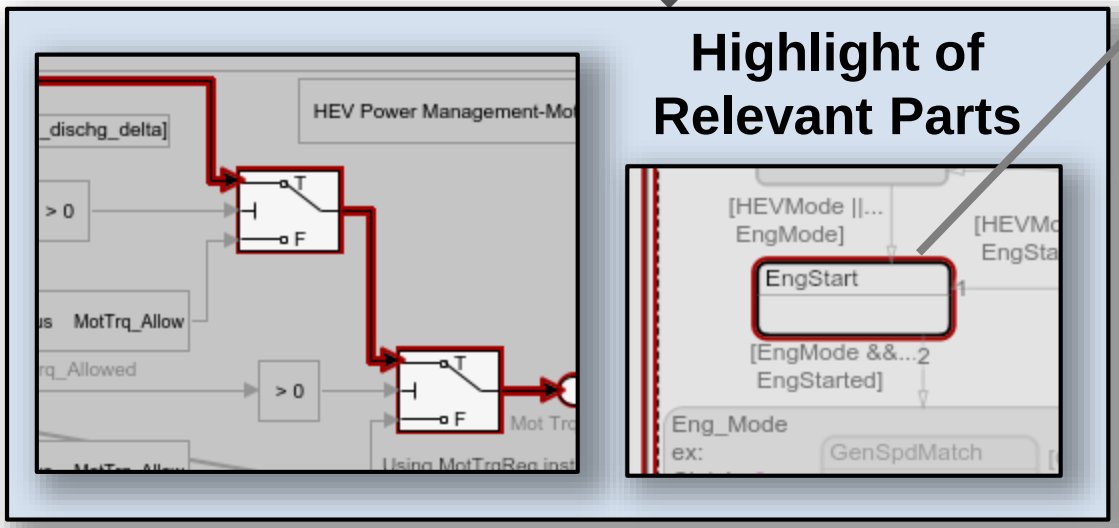


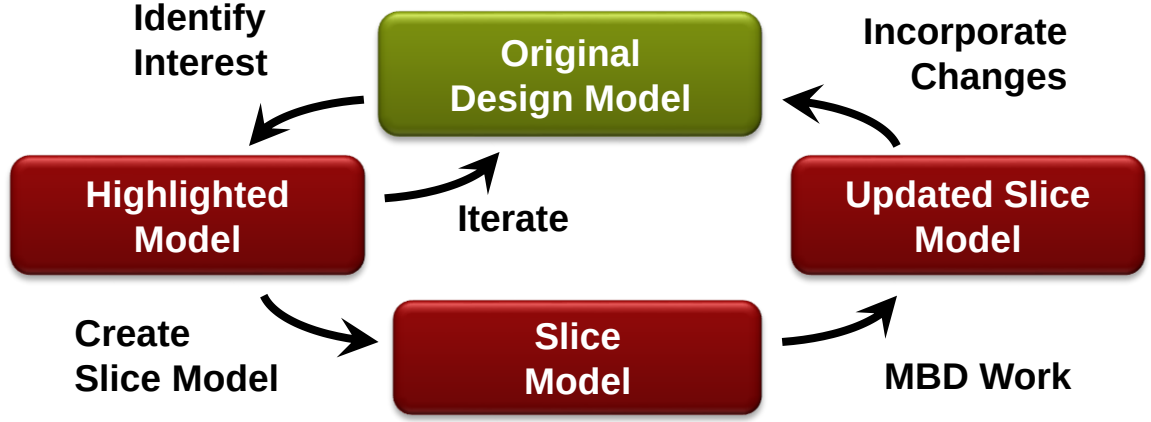
Complete Model Slicer Workflow



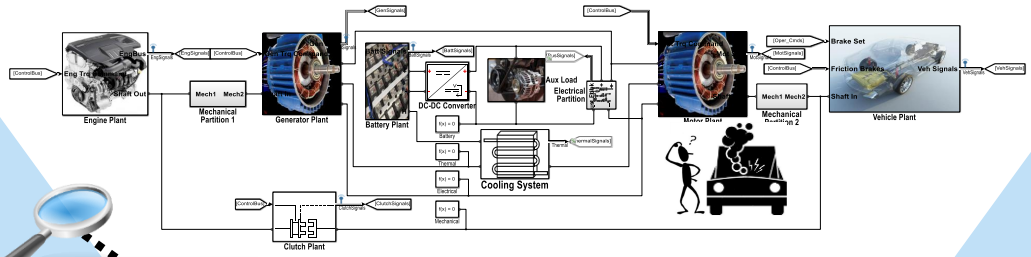
Isolating Troubling Behavior with Model Slicer

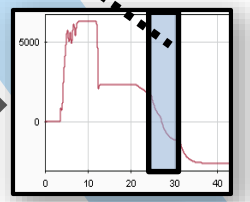






Fixing the failure and Unit Testing



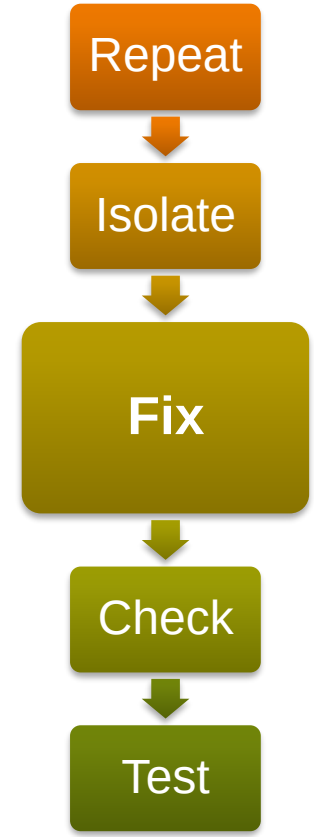


Simulation Scenario

Repeat & isolate the problem

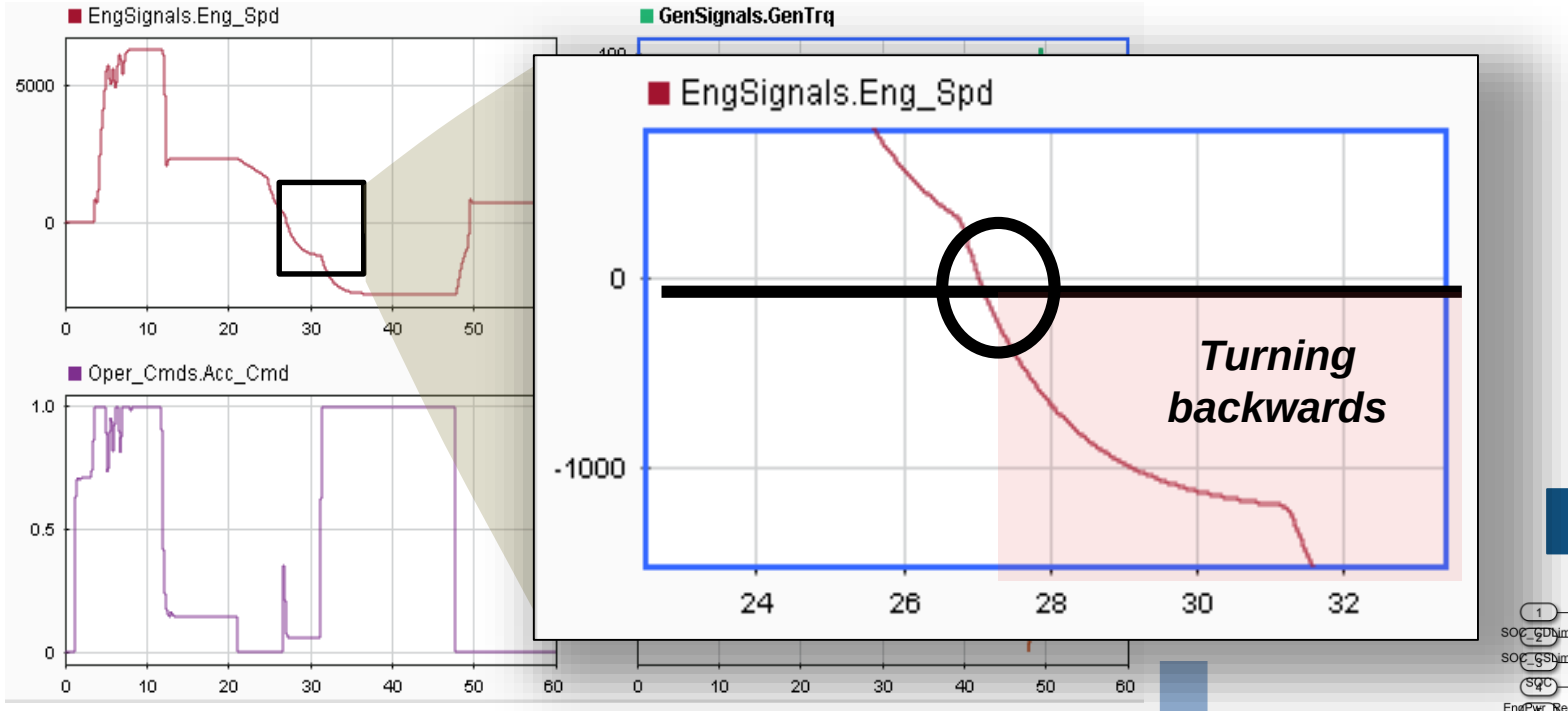
…

Fix faulty components

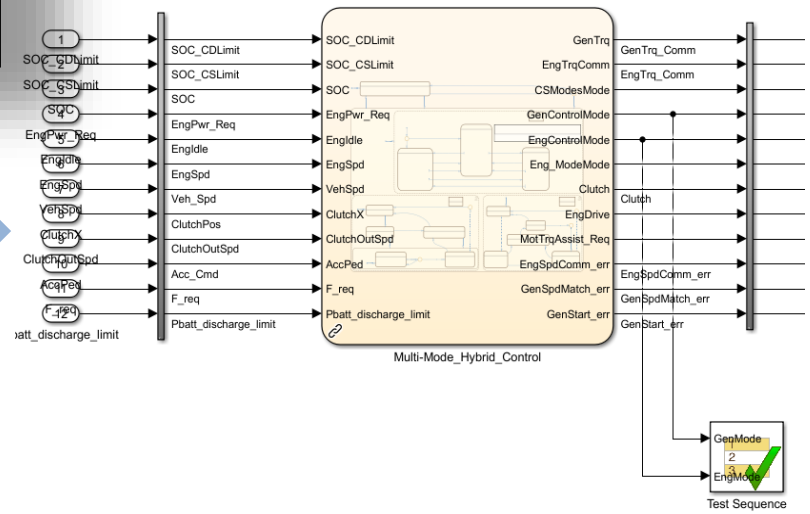


State Synchronization Error

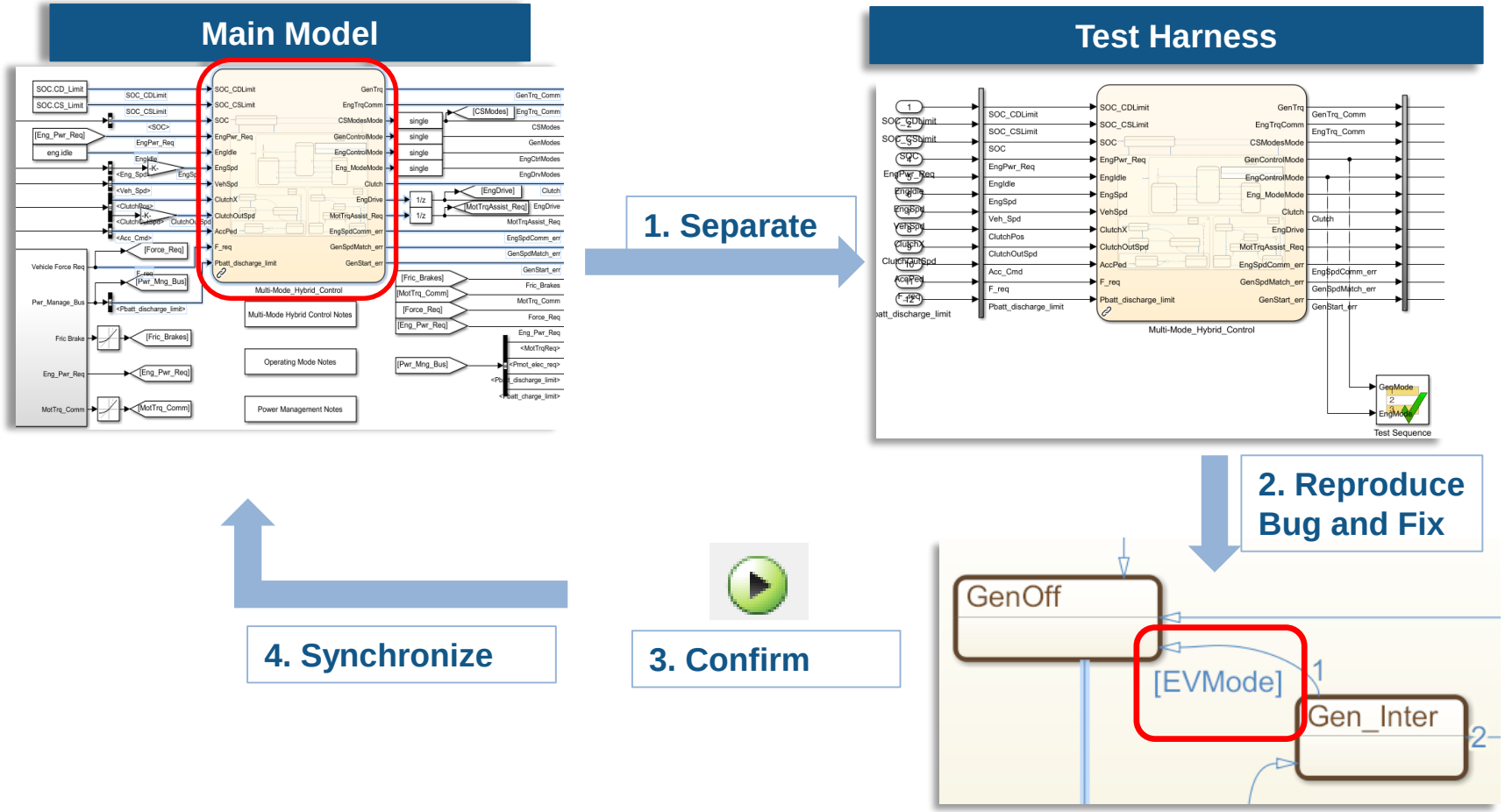
Engine Turning Backwards

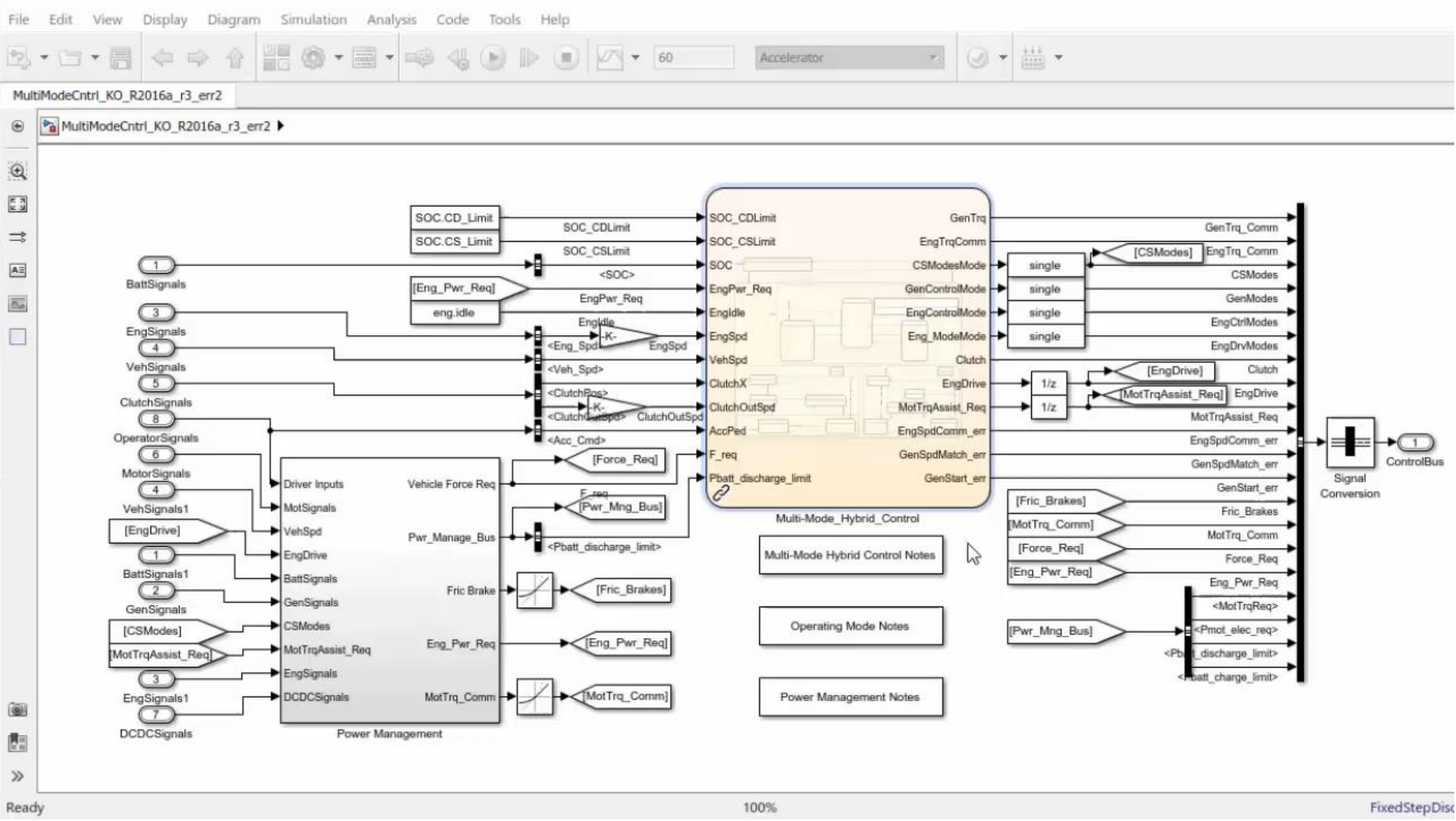


Isolate and Fix



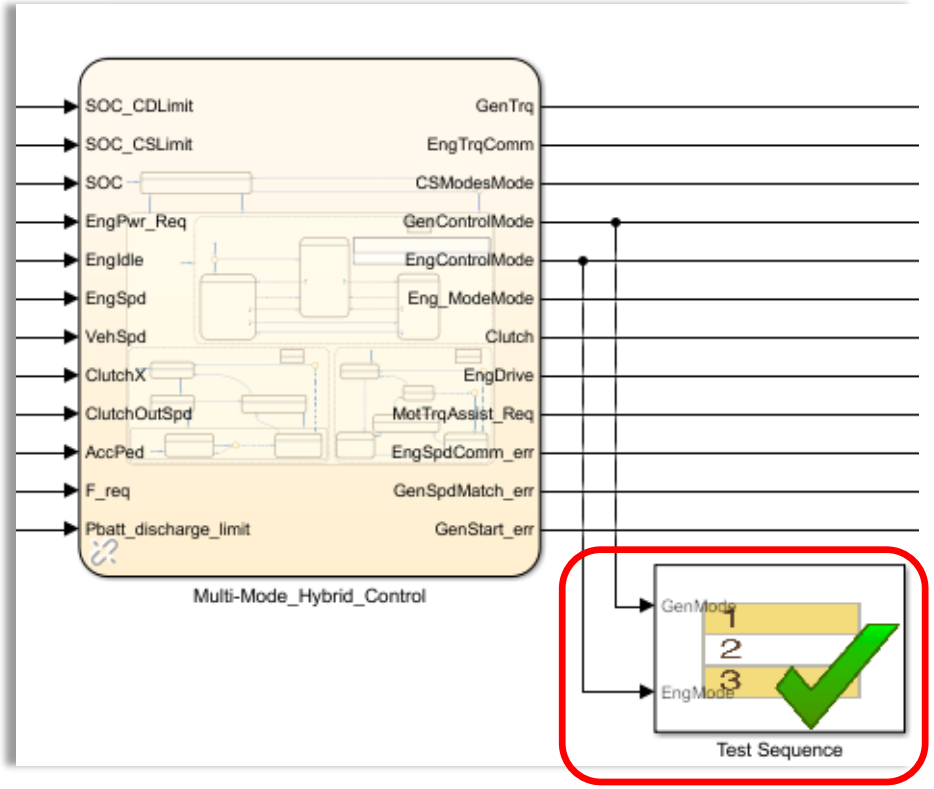
Unit Testing Workflow



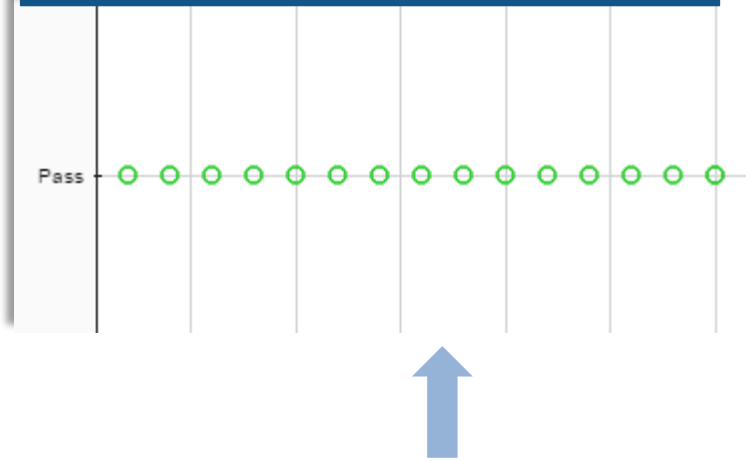

File Edit View Display Diagram Simulation Analysis Code Tools Help
MultiModeCntrl_KO_R2016a_r3_err2
MultiModeCntrl_KO_R2016a_r3_err2
SOC.CD_Limit
SOC.CS_Limit
SOC_CDLimit
SOC_CSLimit
<SOC>
[Eng_Pwr_Req]
eng.idle
EngPwr_Req
EngIdle
<Eng_Spd>
-K-
EngSpd
<Veh_Spd>
<ClutchPos>
<ClutchOutSpd>
ClutchOutSpd
<Acc_Cmd>
1 BattSignals
3 EngSignals
4 VehSignals
5 ClutchSignals
8 OperatorSignals
6 MotorSignals
4 VehSignals1
[EngDrive]
1 BattSignals1
2 GenSignals
[CSModes]
[MotTrqAssist_Req]
3 EngSignals1
7 DCDCSignals
Driver Inputs
MotSignals
VehSpd
EngDrive
BattSignals
GenSignals
CSModes
MotTrqAssist_Req
EngSignals
DCDCSignals
Vehicle Force Req
Pwr_Manage_Bus
Fric Brake
Eng_Pwr_Req
MotTrq_Comm
Power Management
[Force_Req]
F_req
[Pwr_Mng_Bus]
<Pbatt_discharge_limit>
[Fric_Brakes]
[Eng_Pwr_Req]
[MotTrq_Comm]
SOC_CDLimit
SOC_CSLimit
SOC
EngPwr_Req
EngIdle
EngSpd
VehSpd
ClutchX
ClutchOutSpd
AccPed
F_req
Pbatt_discharge_limit
GenTrq
EngTrqComm
CSModesMode
GenControlMode
EngControlMode
Eng_ModeMode
Clutch
EngDrive
MotTrqAssist_Req
EngSpdComm_err
GenSpdMatch_err
GenStart_err
Multi-Mode_Hybrid_Control
Multi-Mode Hybrid Control Notes
Operating Mode Notes
Power Management Notes
single
single
single
single
[CSModes]
1/z
1/z
[EngDrive]
[MotTrqAssist_Req]
[Fric_Brakes]
[MotTrq_Comm]
[Force_Req]
[Eng_Pwr_Req]
[Pwr_Mng_Bus]
GenTrq_Comm
EngTrq_Comm
CSModes
GenModes
EngCtrlModes
EngDrvModes
Clutch
EngDrive
MotTrqAssist_Req
EngSpdComm_err
GenSpdMatch_err
GenStart_err
Fric_Brakes
MotTrq_Comm
Force_Req
Eng_Pwr_Req
<MotTrqReq>
<Pmot_elec_req>
<Pbatt_discharge_limit>
<Pbatt_charge_limit>
Signal Conversion
1 ControlBus
Ready
100%
FixedStepDisc

In-model Verification

Isolate Component in Test Harness



Verification Result streamed to Data Inspector

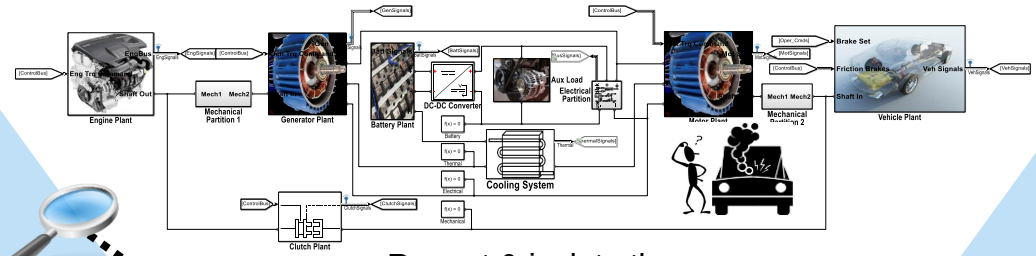


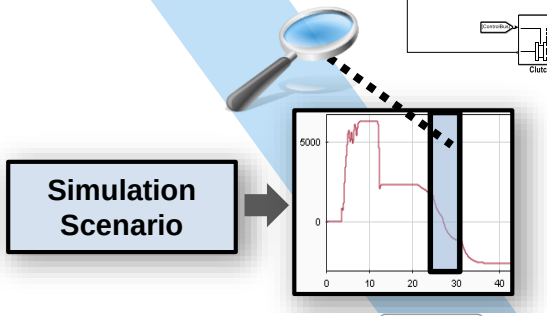
In-model Verification with *verify* keyword

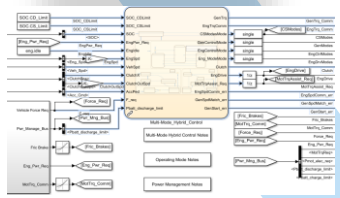
```
CheckOperatingModes

if EngMode == Start
   verify(GenMode ~= Run);
end
```

Check for further design errors







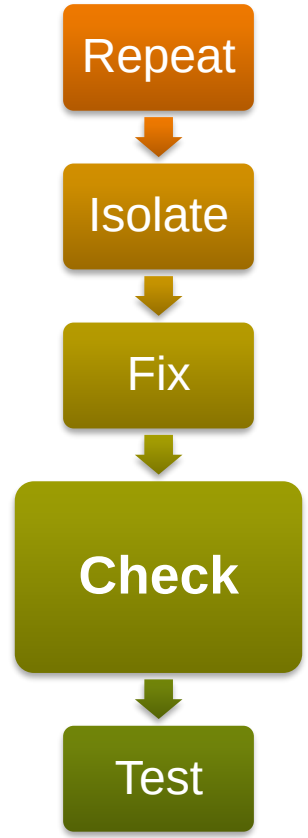
Repeat & isolate the problem

…

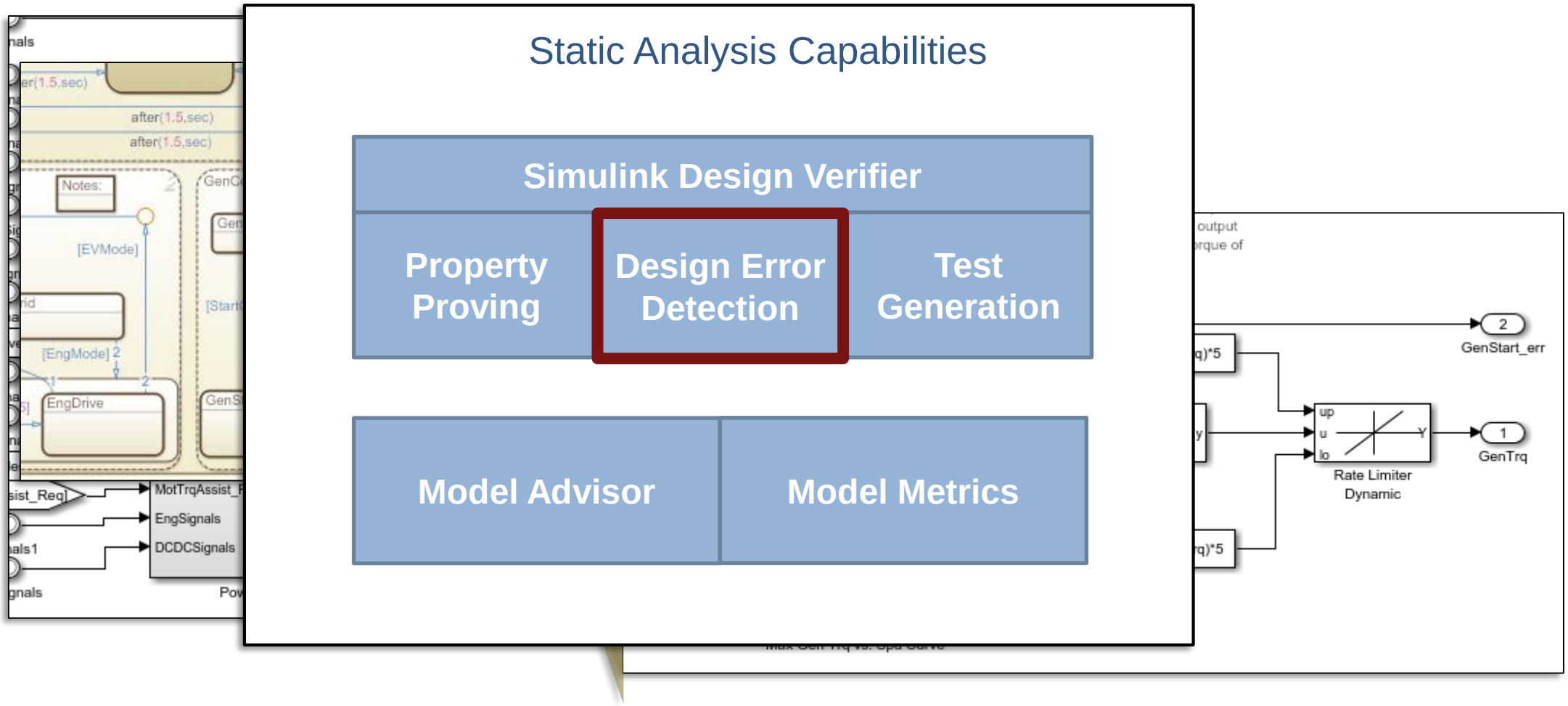
Fix faulty components

…

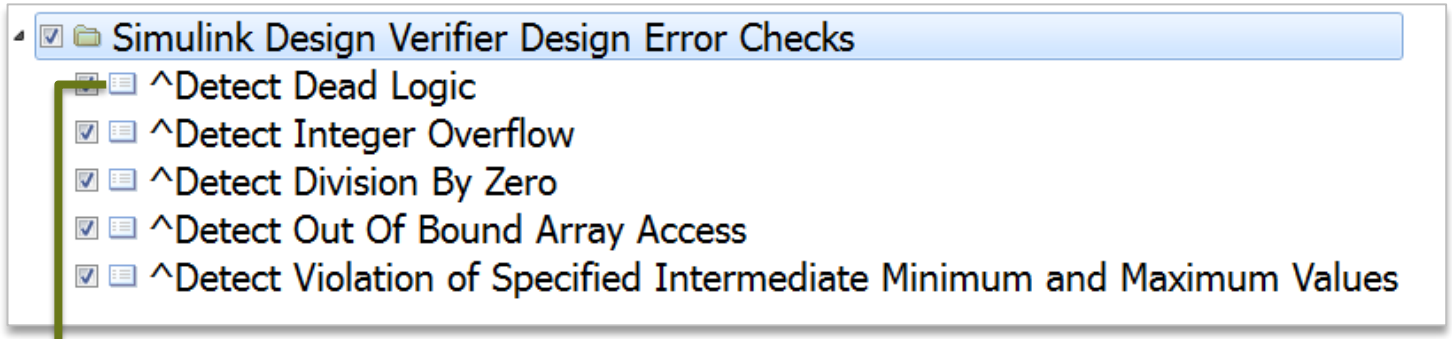
Check everything



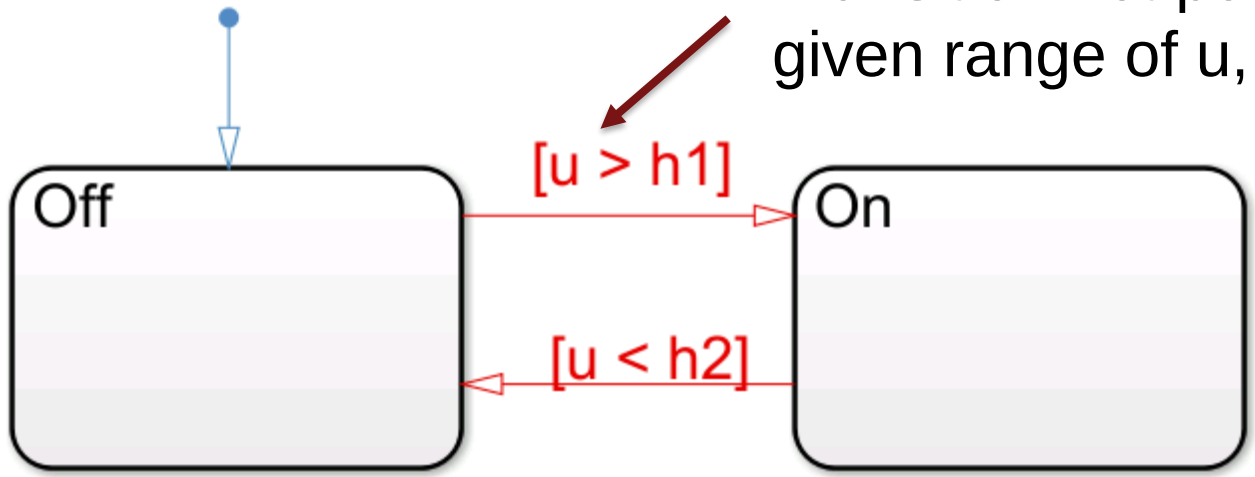
Develop a Robust Design with
Static Checking



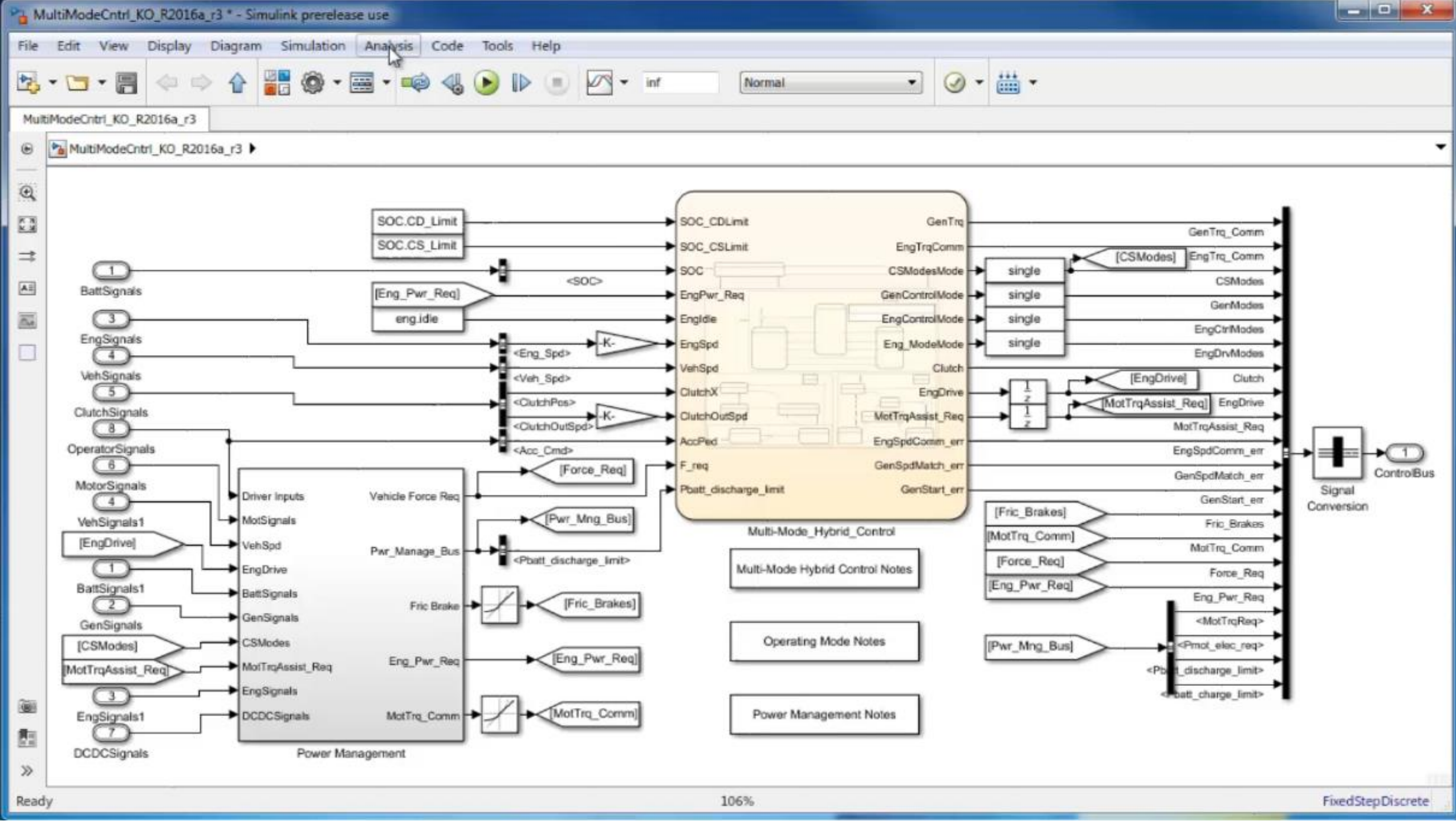
Find Hidden Issues with
Design Error Detection

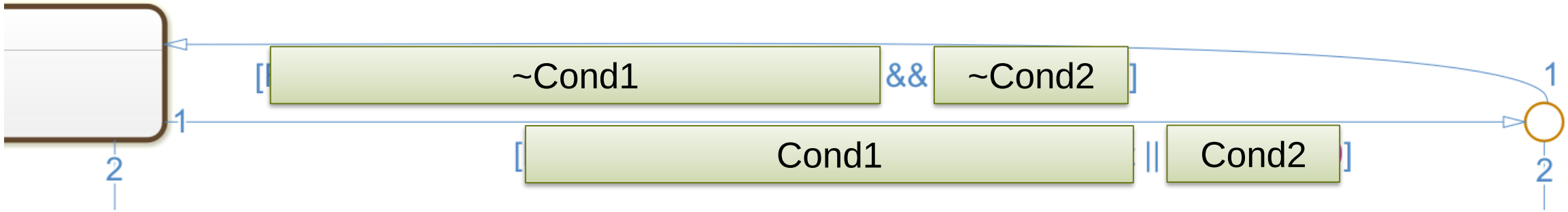


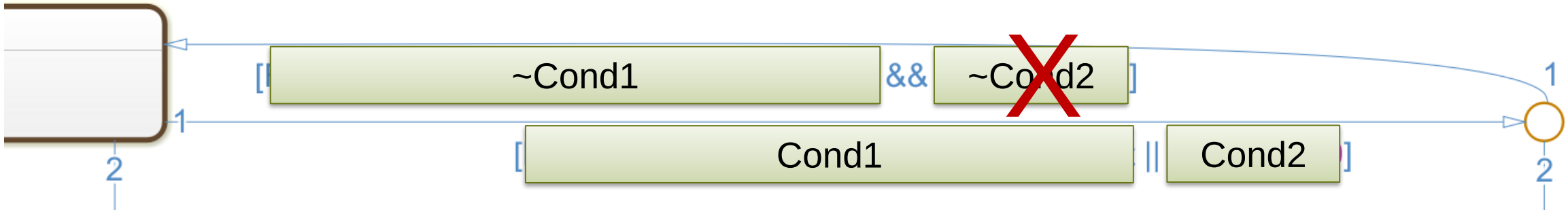
Transition not possible
given range of u, h1, h2

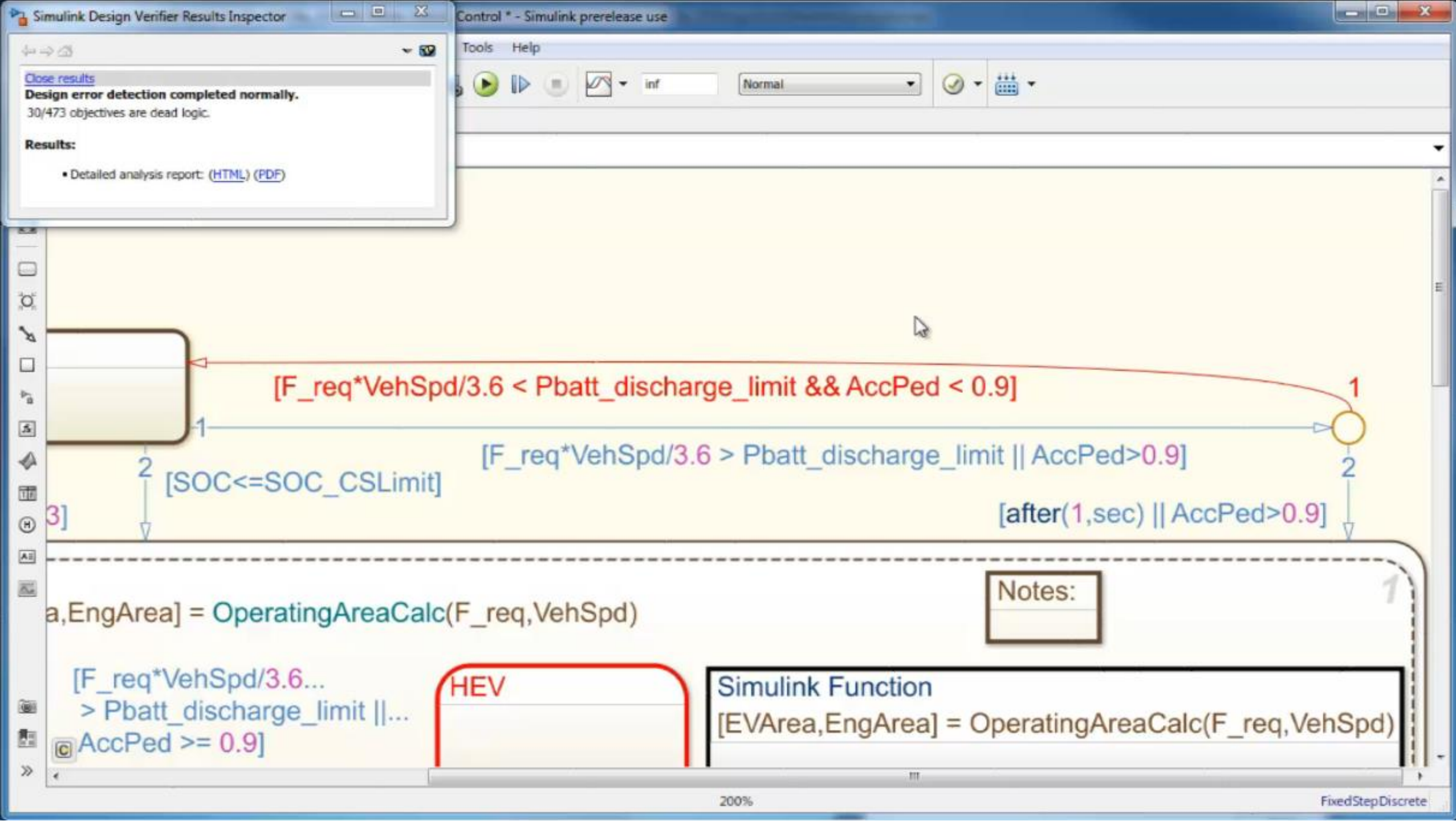


U: [0,10]
h1 = 15
h2 = 5


MultiModeCntrl_KO_R2016a_r3 * - Simulink prerelease use
File Edit View Display Diagram Simulation Analysis Code Tools Help
inf
Normal
MultiModeCntrl_KO_R2016a_r3
MultiModeCntrl_KO_R2016a_r3
SOC.CD_Limit
SOC.CS_Limit
1 BattSignals
3 EngSignals
4 VehSignals
5 ClutchSignals
8 OperatorSignals
6 MotorSignals
4 VehSignals1
[EngDrive]
1 BattSignals1
2 GenSignals
[CSModes]
[MotTrqAssist_Req]
3 EngSignals1
7 DCDCSignals
[Eng_Pwr_Req]
eng.idle
<SOC>
<Eng_Spd>
<Veh_Spd>
<ClutchPos>
<ClutchOutSpd>
<Acc_Cmd>
K-
K-
Driver Inputs
MotSignals
VehSpd
EngDrive
BattSignals
GenSignals
CSModes
MotTrqAssist_Req
EngSignals
DCDCSignals
Vehicle Force Req
Pwr_Manage_Bus
Fric Brake
Eng_Pwr_Req
MotTrq_Comm
Power Management
[Force_Req]
[Pwr_Mng_Bus]
<Pbatt_discharge_limit>
[Fric_Brakes]
[Eng_Pwr_Req]
[MotTrq_Comm]
SOC_CDLimit
SOC_CSLimit
SOC
EngPwr_Req
EngIdle
EngSpd
VehSpd
ClutchX
ClutchOutSpd
AccPed
F_req
Pbatt_discharge_limit
GenTrq
EngTrqComm
CSModesMode
GenControlMode
EngControlMode
Eng_ModeMode
Clutch
EngDrive
MotTrqAssist_Req
EngSpdComm_err
GenSpdMatch_err
GenStart_err
Multi-Mode_Hybrid_Control
Multi-Mode Hybrid Control Notes
Operating Mode Notes
Power Management Notes
single
single
single
single
[CSModes]
[EngDrive]
[MotTrqAssist_Req]
[Fric_Brakes]
[MotTrq_Comm]
[Force_Req]
[Eng_Pwr_Req]
[Pwr_Mng_Bus]
GenTrq_Comm
EngTrq_Comm
CSModes
GenModes
EngCtrlModes
EngDrvModes
Clutch
EngDrive
MotTrqAssist_Req
EngSpdComm_err
GenSpdMatch_err
GenStart_err
Fric_Brakes
MotTrq_Comm
Force_Req
Eng_Pwr_Req
<MotTrqReq>
<Pmot_elec_req>
<Pbatt_discharge_limit>
<Pbatt_charge_limit>
Signal Conversion
1 ControlBus
Ready
106%
FixedStepDiscrete


~Cond1
&&
~Cond2
1
Cond1
||
Cond2
1
2
2

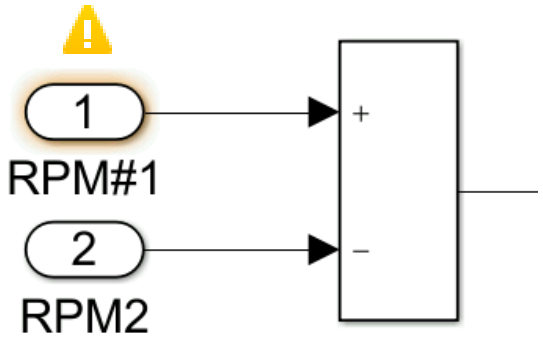

~Cond1
&&
~Cond2
1
1
Cond1
||
Cond2
2
2


Simulink Design Verifier Results Inspector
Close results
Design error detection completed normally.
30/473 objectives are dead logic.
Results:
• Detailed analysis report: (HTML) (PDF)
Control * - Simulink prerelease use
Tools
Help
inf
Normal
[F_req*VehSpd/3.6 < Pbatt_discharge_limit && AccPed < 0.9]
1
1
[F_req*VehSpd/3.6 > Pbatt_discharge_limit || AccPed>0.9]
2
2
[SOC<=SOC_CSLimit]
3]
[after(1,sec) || AccPed>0.9]
Notes:
a,EngArea] = OperatingAreaCalc(F_req,VehSpd)
[F_req*VehSpd/3.6...
> Pbatt_discharge_limit ||...
AccPed >= 0.9]
HEV
Simulink Function
[EVArea,EngArea] = OperatingAreaCalc(F_req,VehSpd)
200%
FixedStepDiscrete

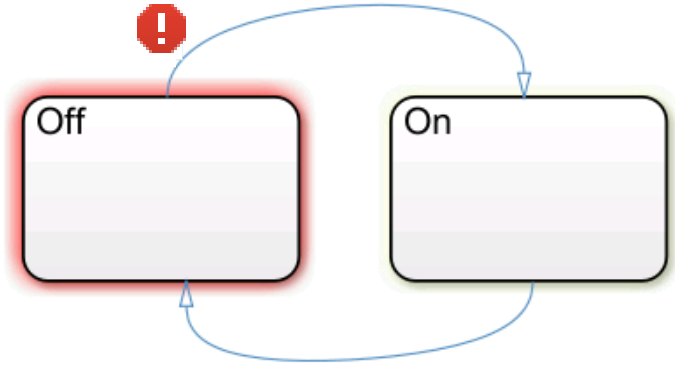
Prevent errors by
Fixing-as-you go

R2016b

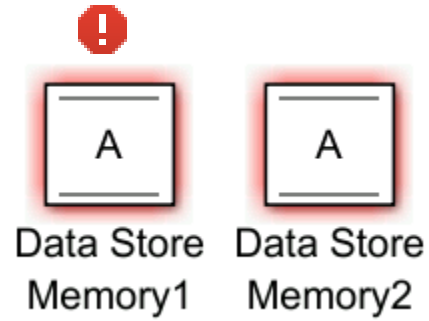
- Edit-time checking
  - Simulink
  - Stateflow
  - Modeling Standards
    - Prohibited blocks violations
    - Block and port name violations
- Customize rules to corporate standards



Invalid port name

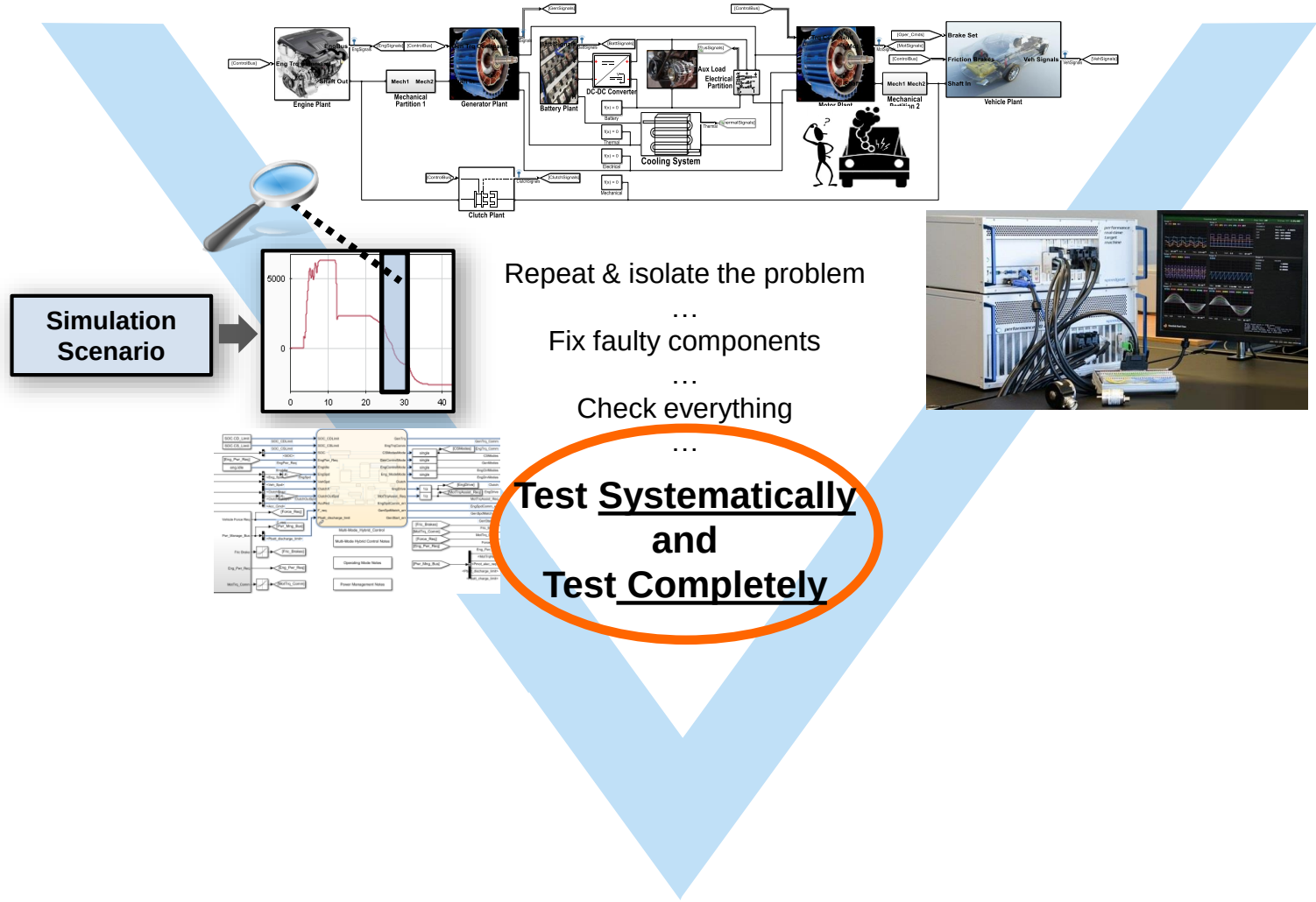


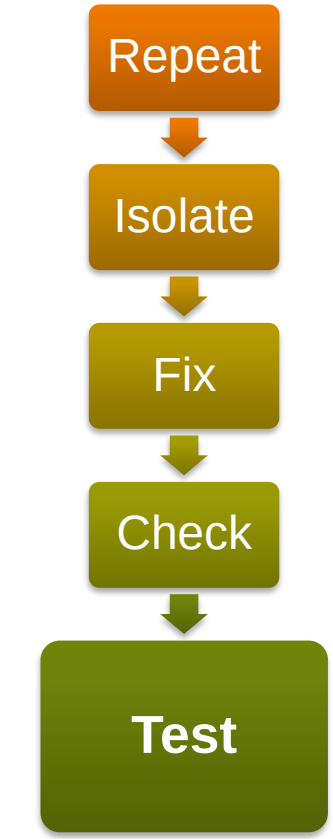
Missing Default Transition

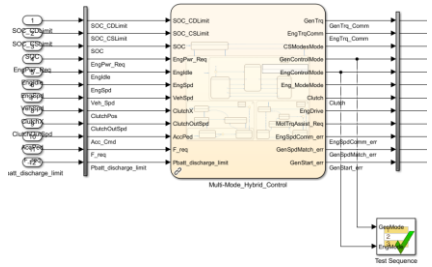


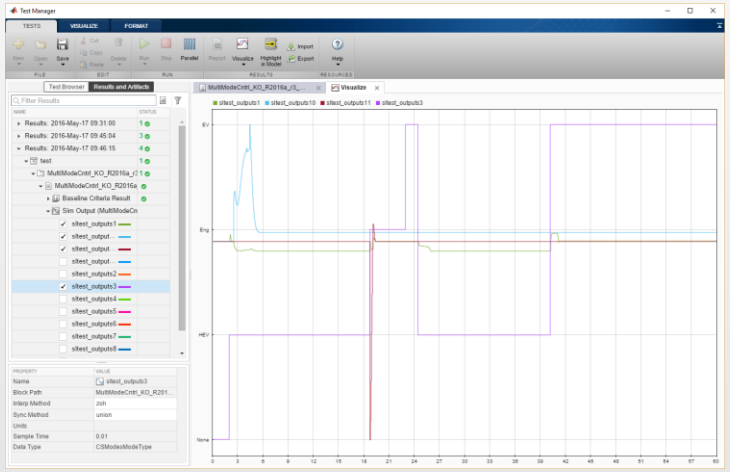
Duplicate Data Store

Test : Systematically and Completely





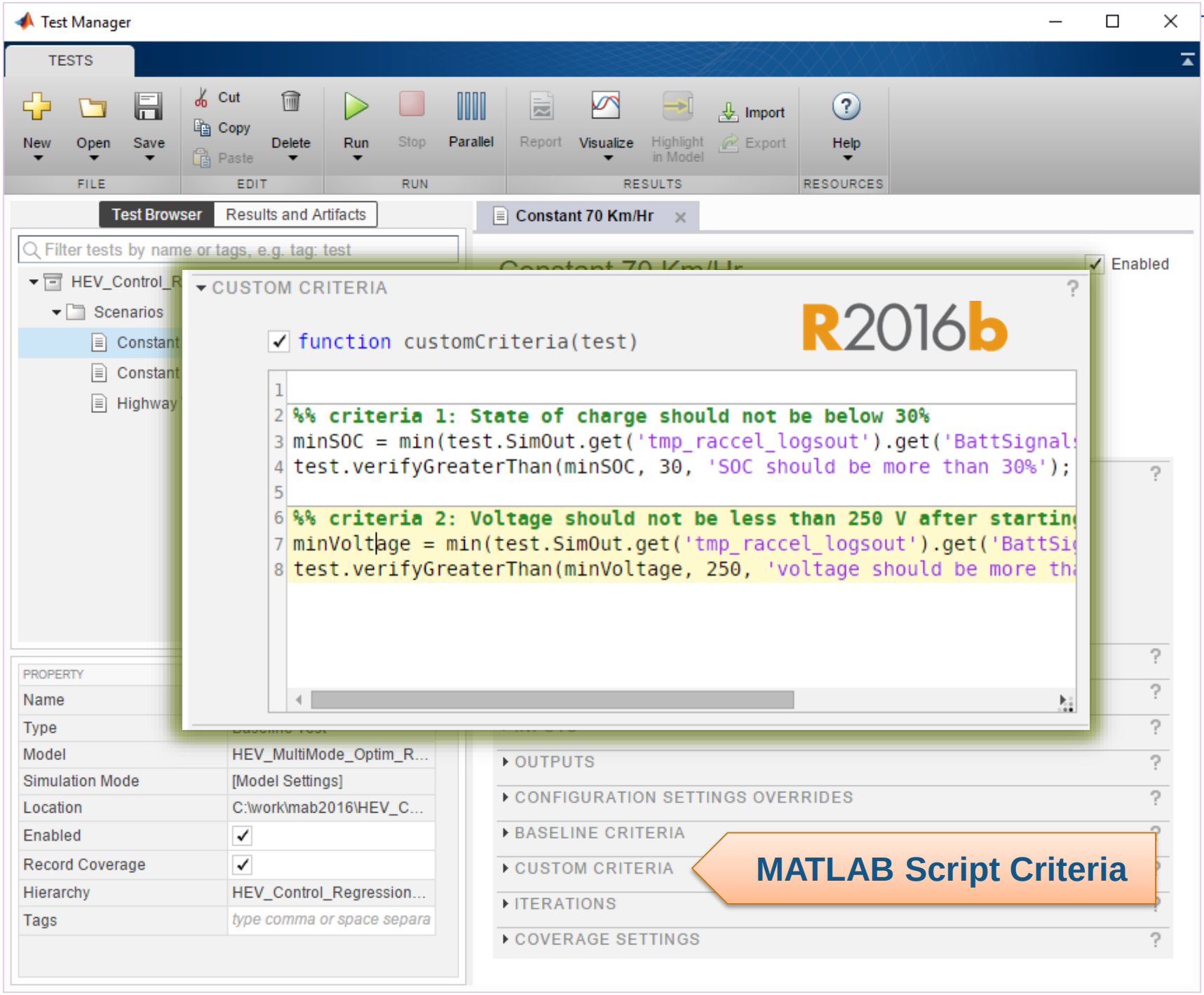


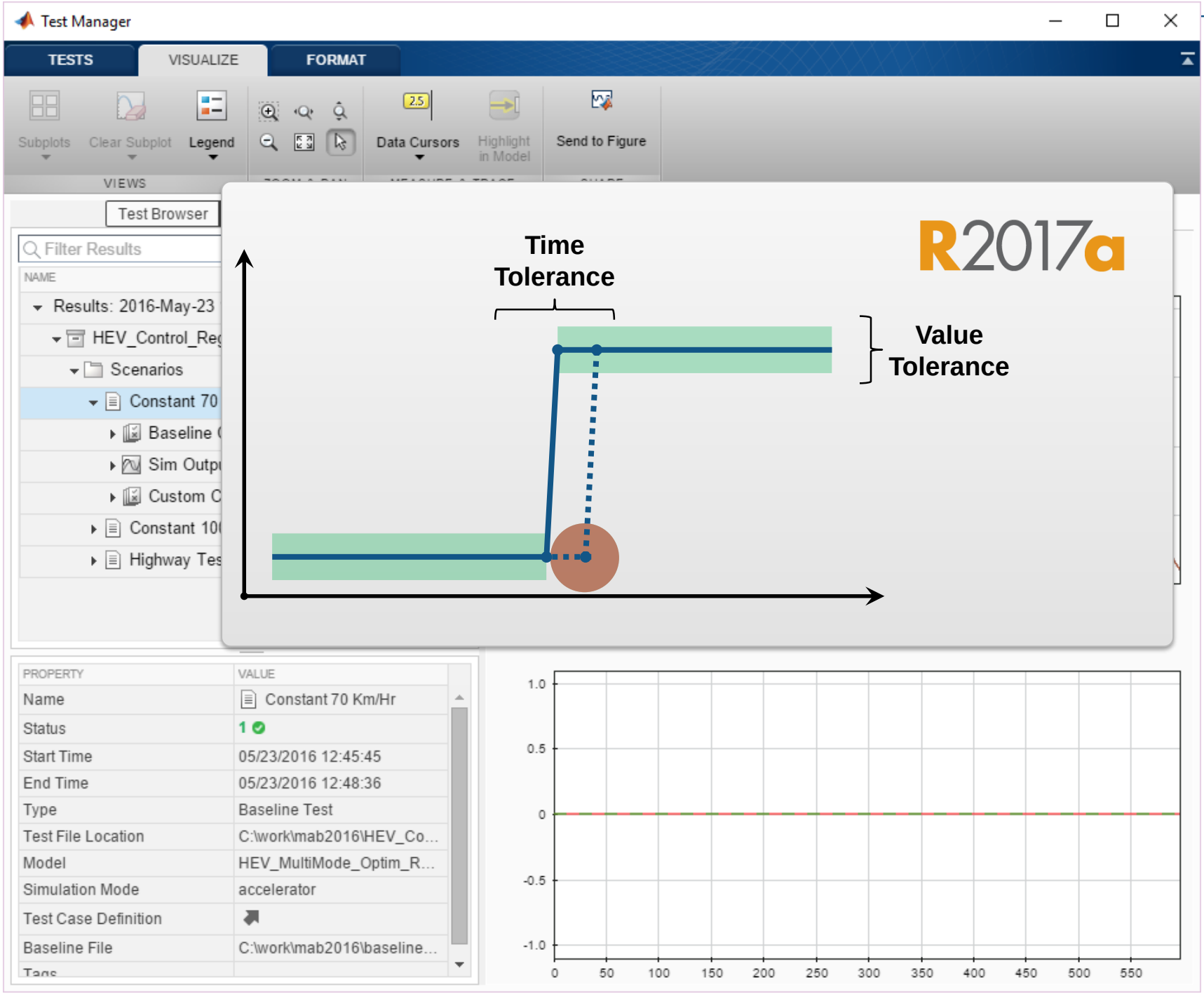


Test Manager Platform

Systematic authoring, management, execution, and reporting of test cases

- Unites together a broad set of capabilities
- Simulink Test
- R2015a

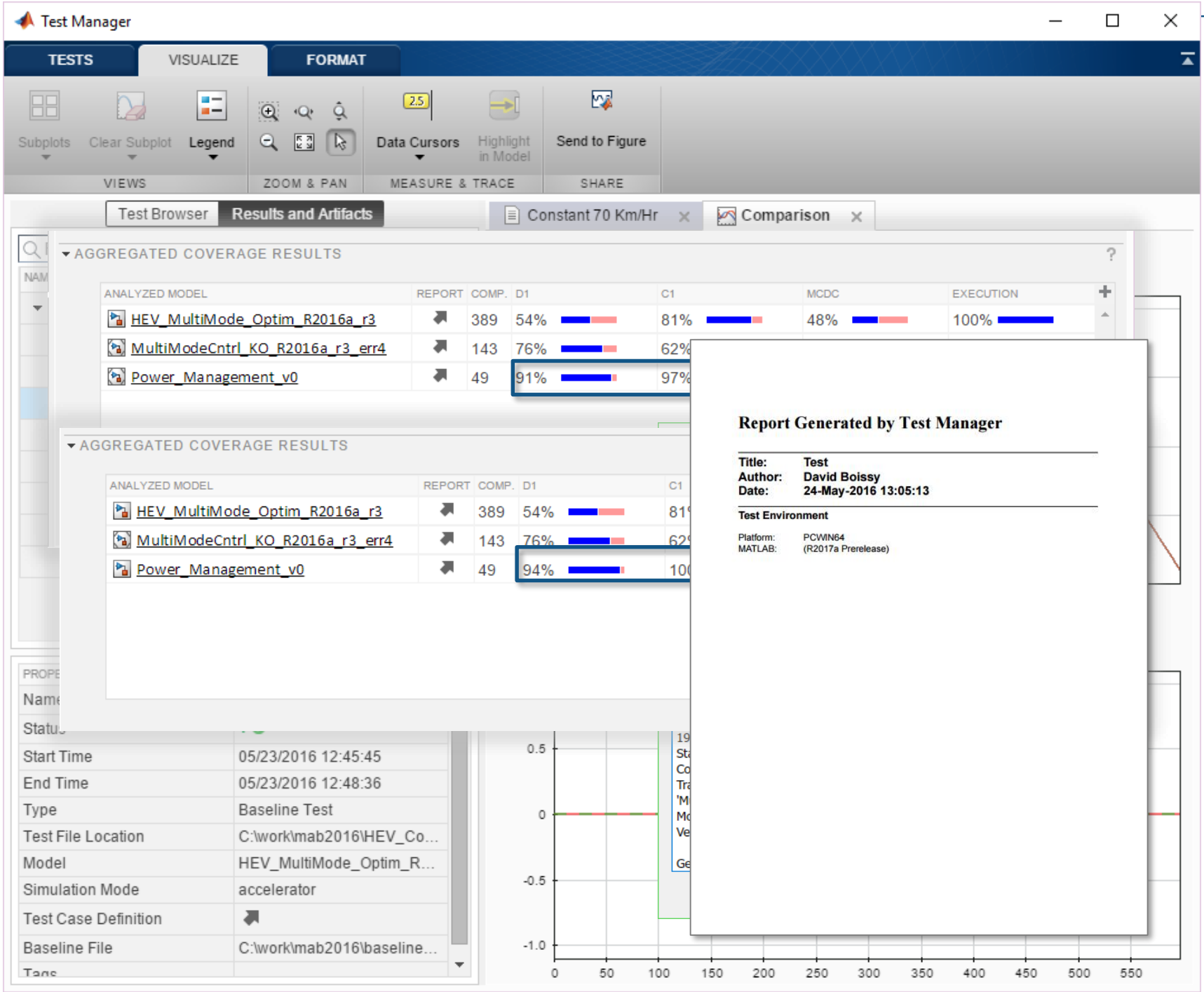

Test Manager
TESTS
New
Open
Save
Cut
Copy
Paste
Delete
Run
Stop
Parallel
Report
Visualize
Highlight in Model
Import
Export
Help
FILE
EDIT
RUN
RESULTS
RESOURCES
Test Browser
Results and Artifacts
Filter tests by name or tags, e.g. tag: test
HEV_Control_R
Scenarios
Constant
Constant
Highway
Constant 70 Km/Hr
Enabled
CUSTOM CRITERIA
function customCriteria(test)
R2016b
1
2 %% criteria 1: State of charge should not be below 30%
3 minSOC = min(test.SimOut.get('tmp_raccel_logsout').get('BattSignal
4 test.verifyGreaterThan(minSOC, 30, 'SOC should be more than 30%');
5
6 %% criteria 2: Voltage should not be less than 250 V after startin
7 minVoltage = min(test.SimOut.get('tmp_raccel_logsout').get('BattSi
8 test.verifyGreaterThan(minVoltage, 250, 'voltage should be more th
PROPERTY
Name
Type
Model
HEV_MultiMode_Optim_R...
Simulation Mode
[Model Settings]
Location
C:\work\mab2016\HEV_C...
Enabled
Record Coverage
Hierarchy
HEV_Control_Regression...
Tags
type comma or space separa
OUTPUTS
CONFIGURATION SETTINGS OVERRIDES
BASELINE CRITERIA
CUSTOM CRITERIA
ITERATIONS
COVERAGE SETTINGS
MATLAB Script Criteria


Test Manager
TESTS
VISUALIZE
FORMAT
Subplots
Clear Subplot
Legend
Data Cursors
Highlight in Model
Send to Figure
VIEWS
Test Browser
Filter Results
NAME
Results: 2016-May-23
HEV_Control_Reg
Scenarios
Constant 70
Baseline
Sim Outpu
Custom C
Constant 10
Highway Tes
Time Tolerance
Value Tolerance
R2017a
PROPERTY
VALUE
Name
Constant 70 Km/Hr
Status
1
Start Time
05/23/2016 12:45:45
End Time
05/23/2016 12:48:36
Type
Baseline Test
Test File Location
C:\work\mab2016\HEV_Co...
Model
HEV_MultiMode_Optim_R...
Simulation Mode
accelerator
Test Case Definition
Baseline File
C:\work\mab2016\baseline...
Tags
1.0
0.5
0
-0.5
-1.0
0
50
100
150
200
250
300
350
400
450
500
550

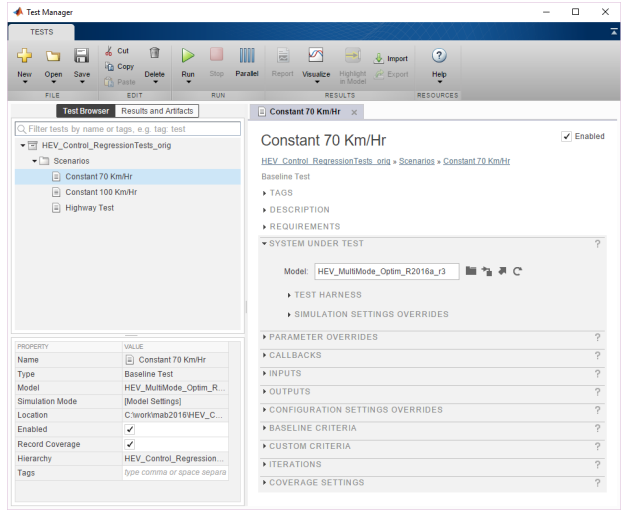
Top-It-Off Workflow

R2017a

1. Run Existing Tests
   - Aggregate Coverage
2. Simulink Design Verifier
   - Generate Test Cases
3. Run New Tests
   - Aggregate Coverage


Test Manager
TESTS
VISUALIZE
FORMAT
Subplots
Clear Subplot
Legend
VIEWS
Data Cursors
Highlight in Model
ZOOM & PAN
MEASURE & TRACE
Send to Figure
SHARE
Test Browser
Results and Artifacts
Constant 70 Km/Hr
Comparison
AGGREGATED COVERAGE RESULTS
ANALYZED MODEL
REPORT
COMP.
D1
C1
MCDC
EXECUTION
HEV_MultiMode_Optim_R2016a_r3
389
54%
81%
48%
100%
MultiModeCntrl_KO_R2016a_r3_err4
143
76%
62%
Power_Management_v0
49
91%
97%
AGGREGATED COVERAGE RESULTS
ANALYZED MODEL
REPORT
COMP.
D1
HEV_MultiMode_Optim_R2016a_r3
389
54%
MultiModeCntrl_KO_R2016a_r3_err4
143
76%
Power_Management_v0
49
94%
Report Generated by Test Manager
Title: Test
Author: David Boissy
Date: 24-May-2016 13:05:13
Test Environment
Platform: PCWIN64
MATLAB: (R2017a Prerelease)
Start Time 05/23/2016 12:45:45
End Time 05/23/2016 12:48:36
Type Baseline Test
Test File Location C:\work\mab2016\HEV_Co...
Model HEV_MultiMode_Optim_R...
Simulation Mode accelerator
Test Case Definition
Baseline File C:\work\mab2016\baseline...
0.5
0
-0.5
-1.0
0
50
100
150
200
250
300
350
400
450
500
550

Integration with Test Automation Servers

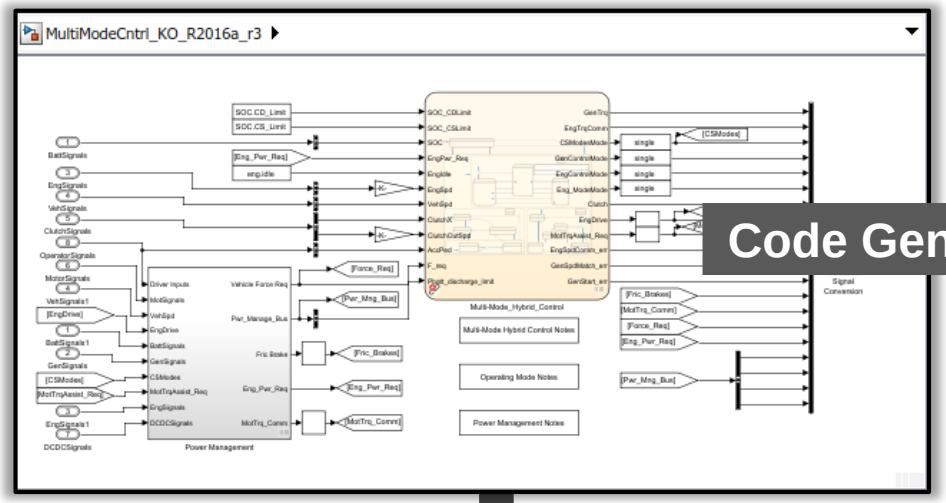


Jenkins

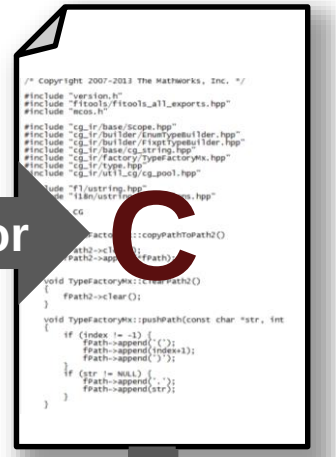
Any continuous integration system that supports Test Anything Protocol (TAP)

R2016b

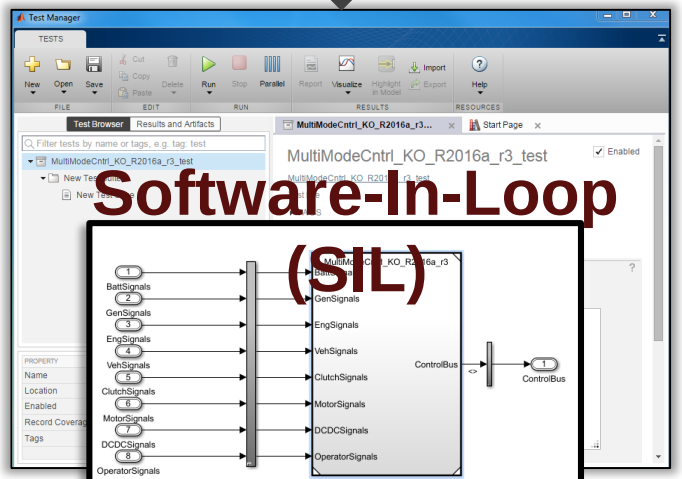
Code-to-Model Verification



Code Generator



Test Manager

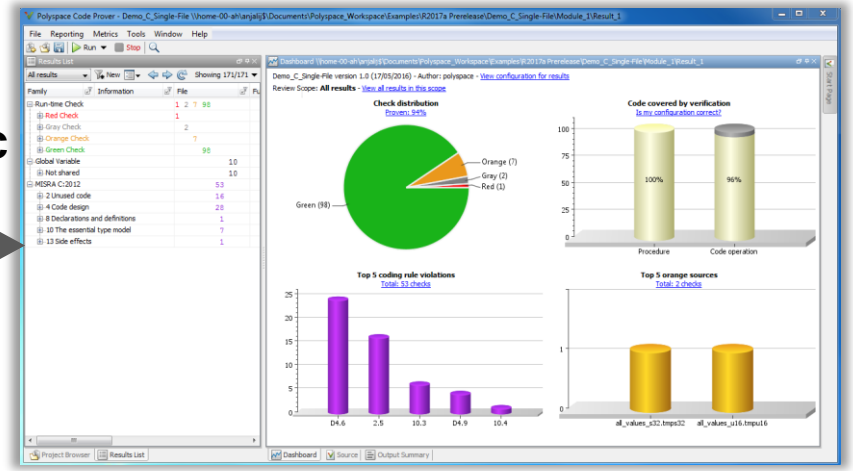


Software-In-Loop (SIL)

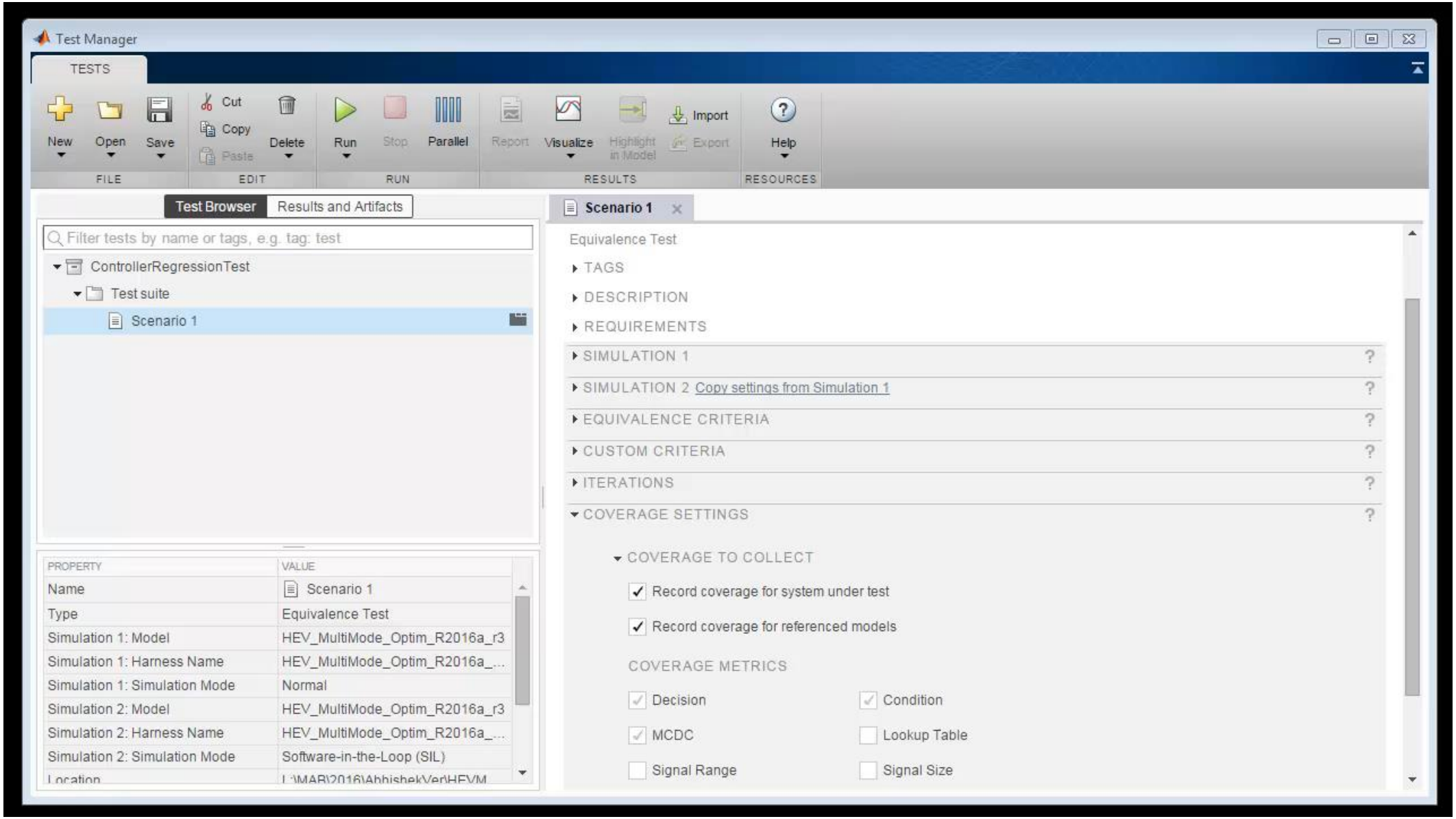
Dynamic

Static

Polyspace

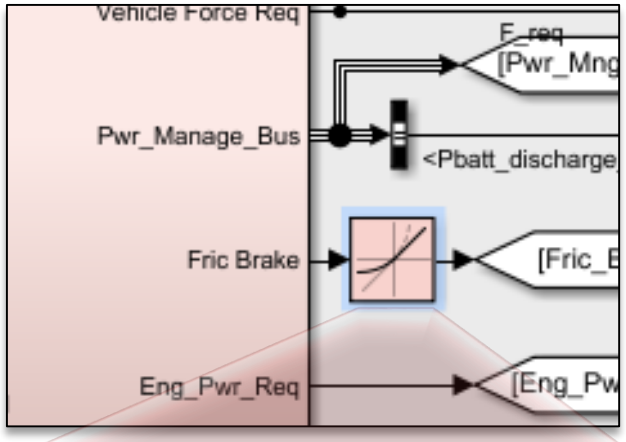


Equivalence Checking and Code Coverage (Software-In-Loop)



Justification for Code Coverage

R2016b



10.28. Decision `ClutchOutSpd > 250.0F` **(line 2769)**

Justify or Exclude

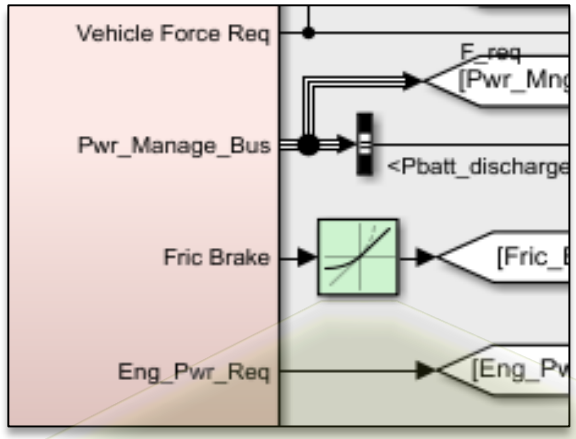
Function: MultiModeCntrl_KO_R2016a_r3_err2_sil_step
Model Object: Rate Limiter3
Uncovered Links:

Metric	Coverage
Decision (D1)	50% (1/2) decision outcomes

Decisions analyzed:

`ClutchOutSpd > 250.0F`	50%
false	101/101
true	0/101

Justify



10.28. Decision `ClutchOutSpd > 250.0F` **(line 2769)**

Justified (Remove this rule)

Function: MultiModeCntrl_KO_R2016a_r3_err2_sil_step
Model Object: Rate Limiter3

Metric	Coverage
Decision (D1)	100% ((1+1)/2) decision outcomes

Decisions analyzed:

`ClutchOutSpd > 250.0F`	50%
false	101/101
true	0/101

Continuous Test and Verification Framework helps to…

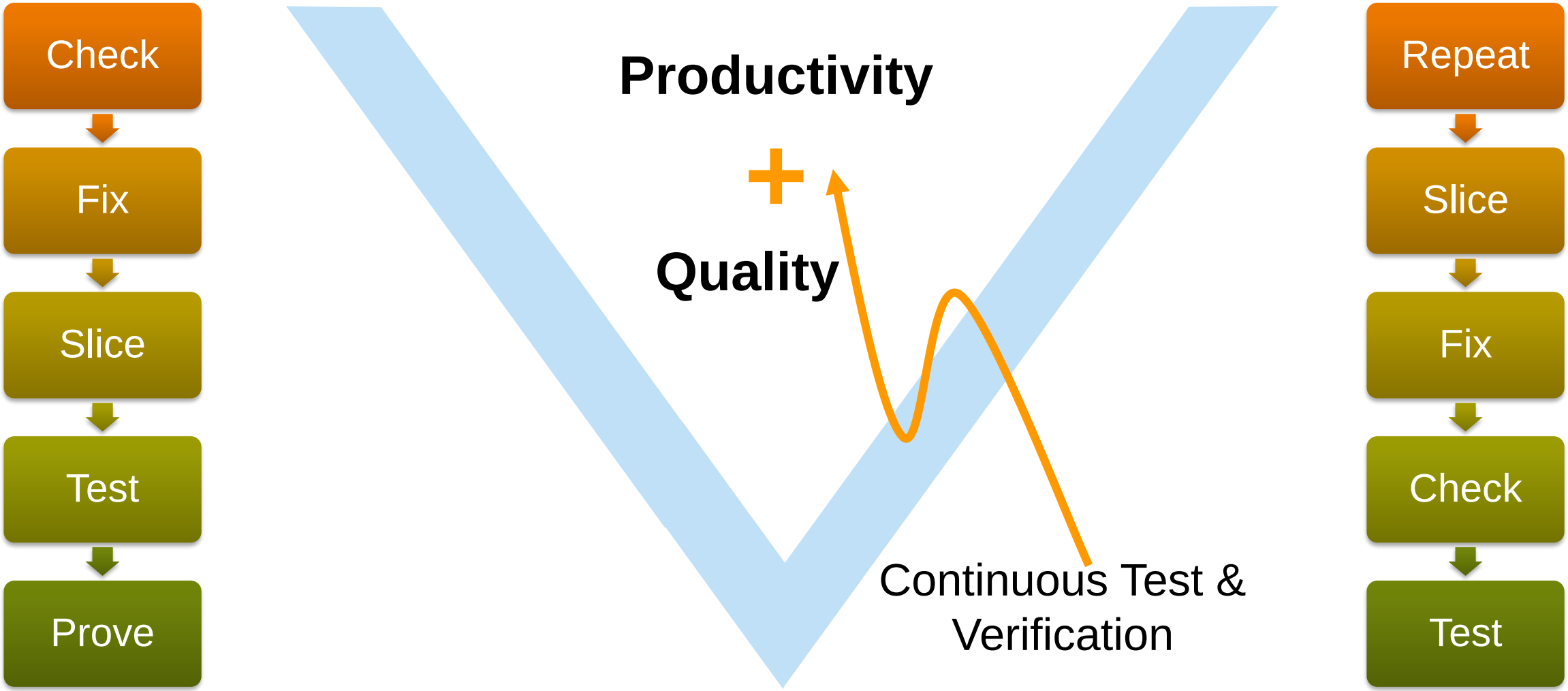
"Reactively" Reproduce the Field Issues in Simulation

- **Repeat** — Reproduce the failure in simulation
- **Slice** — Isolate the problematic behavior
- **Fix** — Fix and Perform Unit Testing
- **Check** — Check for further design errors
- **Test** — Test Systematically and Test Completely

"Proactively" Prove that Implementation satisfies Requirements

- **Check** — Check for design errors early
- **Fix** — Fix and Perform Unit Testing
- **Slice** — Isolate the problematic behavior to simplify debugging
- **Test** — Test Systematically and Test Completely
- **Prove** — Prove Safety Properties/Requirements

Model Based Design helps to…



MathWorks Training Offerings

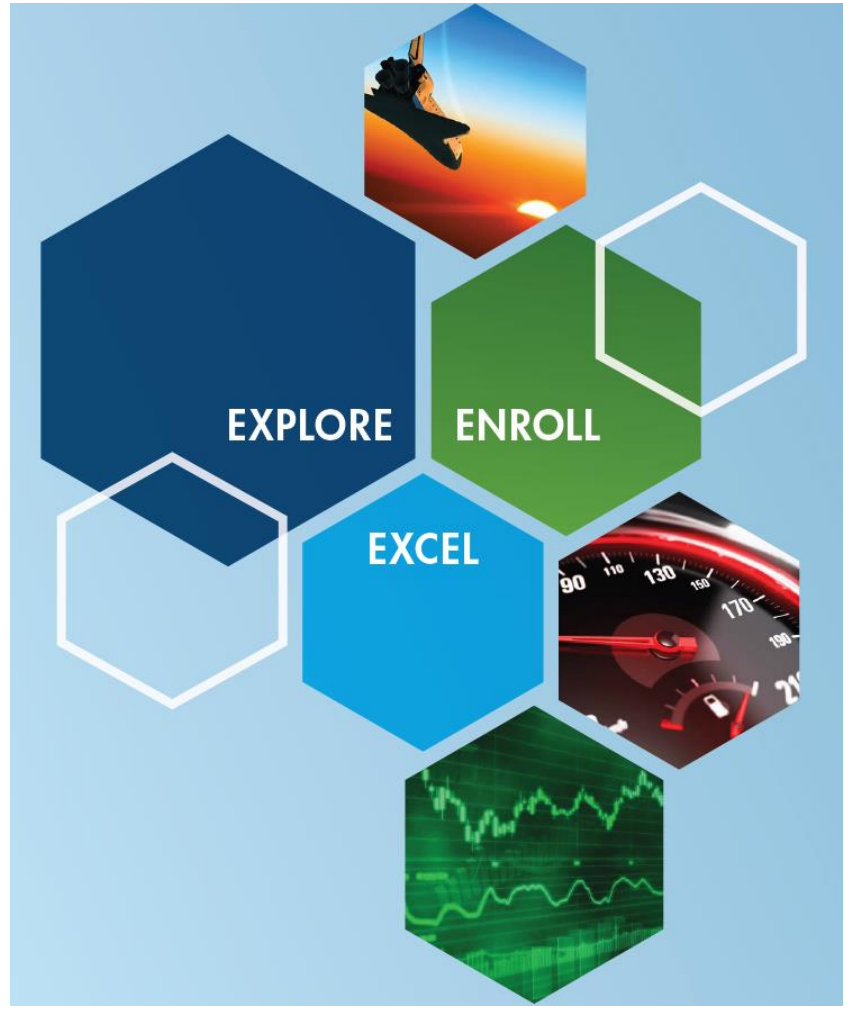
Verification and Validation of Simulink Models

ADVANCED

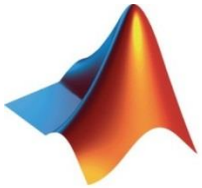
This one-day course describes techniques for testing Simulink model behavior against system requirements. Topics include:

- Identifying the role of verification and validation in Model-Based Design
- Creating test cases for Simulink models
- Analyzing simulation results to verify model behavior
- Automating testing activities and managing results
- Formally verifying model behavior
- Automatically generating artifacts to communicate results

Prerequisites: *MATLAB Fundamentals* and *Simulink for System and Algorithm Modeling*. This course is intended for intermediate or advanced Simulink users.



http://www.mathworks.com/services/training/



MathWorks®

Accelerating the pace of engineering and science

Speaker Details

Manohar Reddy

Email: Manohar.Reddy@mathworks.in

Contact MathWorks India

Call: 080-6632-6000

Email: info@mathworks.in