Vehicle Speedometer Module

Prepared by:

Nishantkumar V Patel

INDEX

1. Objectives

- LCD based TFT cluster has become a norm in today's automotive market. Cluster comes with a 7 inch TFT LCD screen & a 32 bit micro controller. The Cluster could process various information from other ECU's & display it in its large TFT LCD screen.

- To develop a Simulink model as per the requirement and sldd creation, MIL/ SIL simulation and coverage reports, Harness results for the model, design error detection and test generation process using Design verifier, C code generation to be done.
- Also perform the Baseline & Back-to-back tests using Simulink Test Manager.

2. General Overview

- <u>Development of TFT Cluster Speedometer Software Component</u>
- Cluster Instrument receives the signals from other ECU via CAN bus interface. It also receives commands from the driver via steering wheel buttons. The signals are then processed by the Cluster ECU, after which the Cluster ECU may send the vehicle information to the LCD or light the LEDs and so forth, according to the results. Therefore, the Cluster ECU can be viewed as a carrier of the information flow.

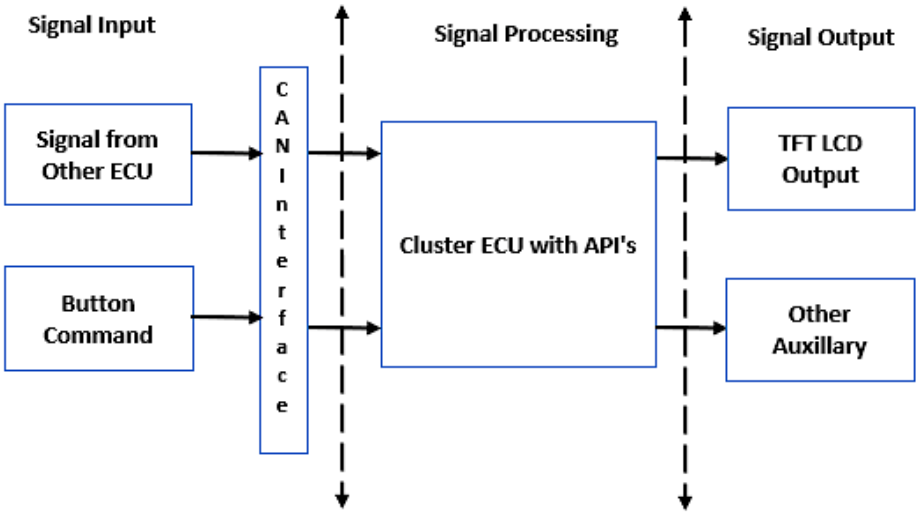
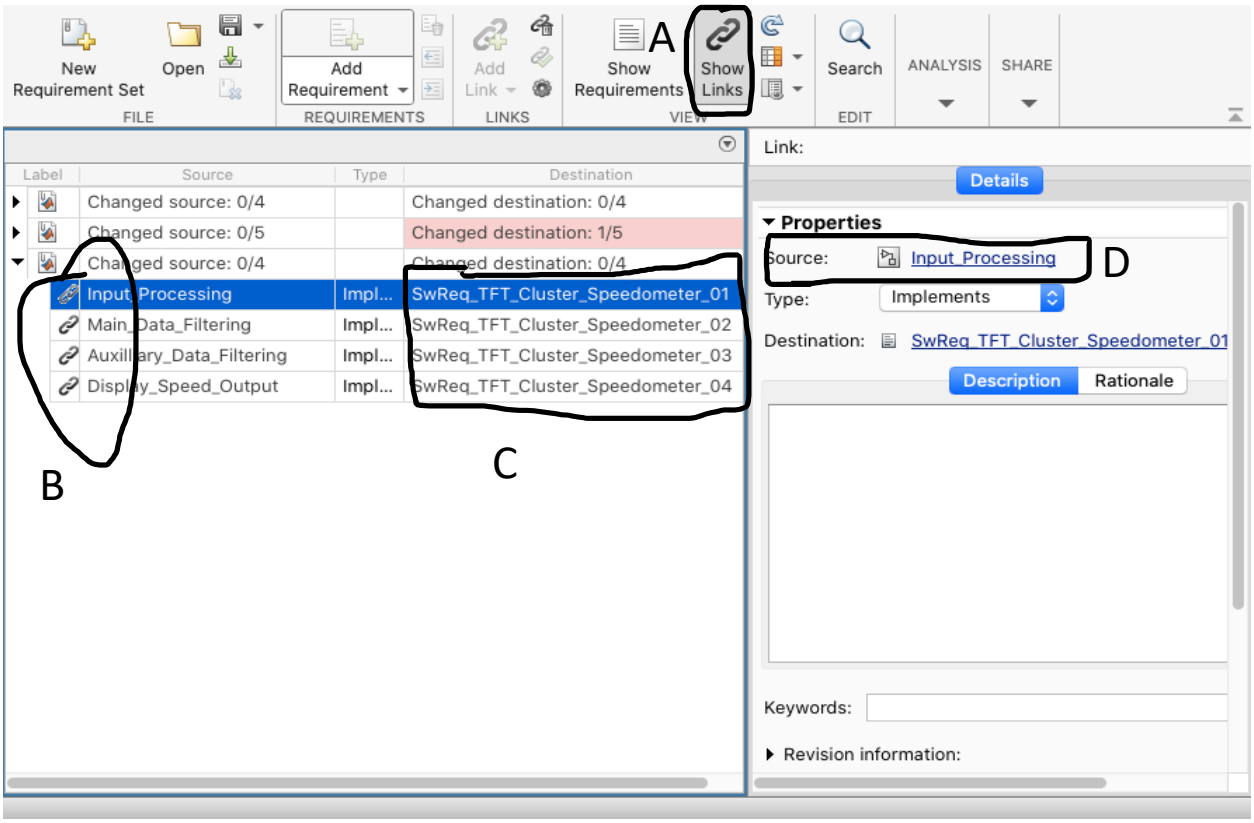


Image Courtesy: Skill Lync Internet

- Cluster Instrument receives the signals from other ECU's via CAN bus interface. It also receives commands from the driver via steering wheel buttons. The signals are then processed by the Cluster ECU, after which the Cluster ECU

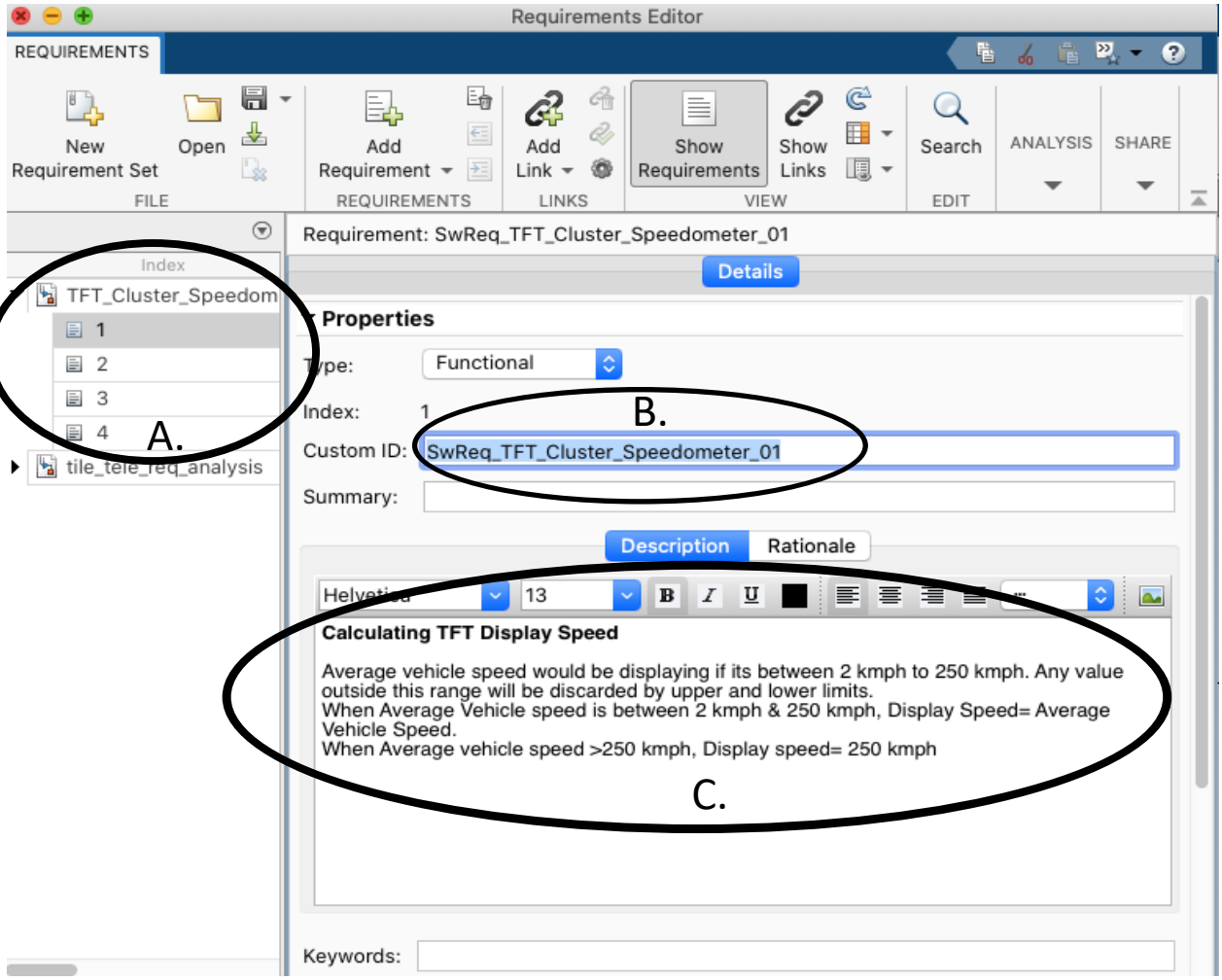
may send the vehicle information to the LCD or light the LEDs and so forth, according to the results. Therefore, the Cluster ECU can be viewed as a carrier of the information flow.

2. Requirement Analysis



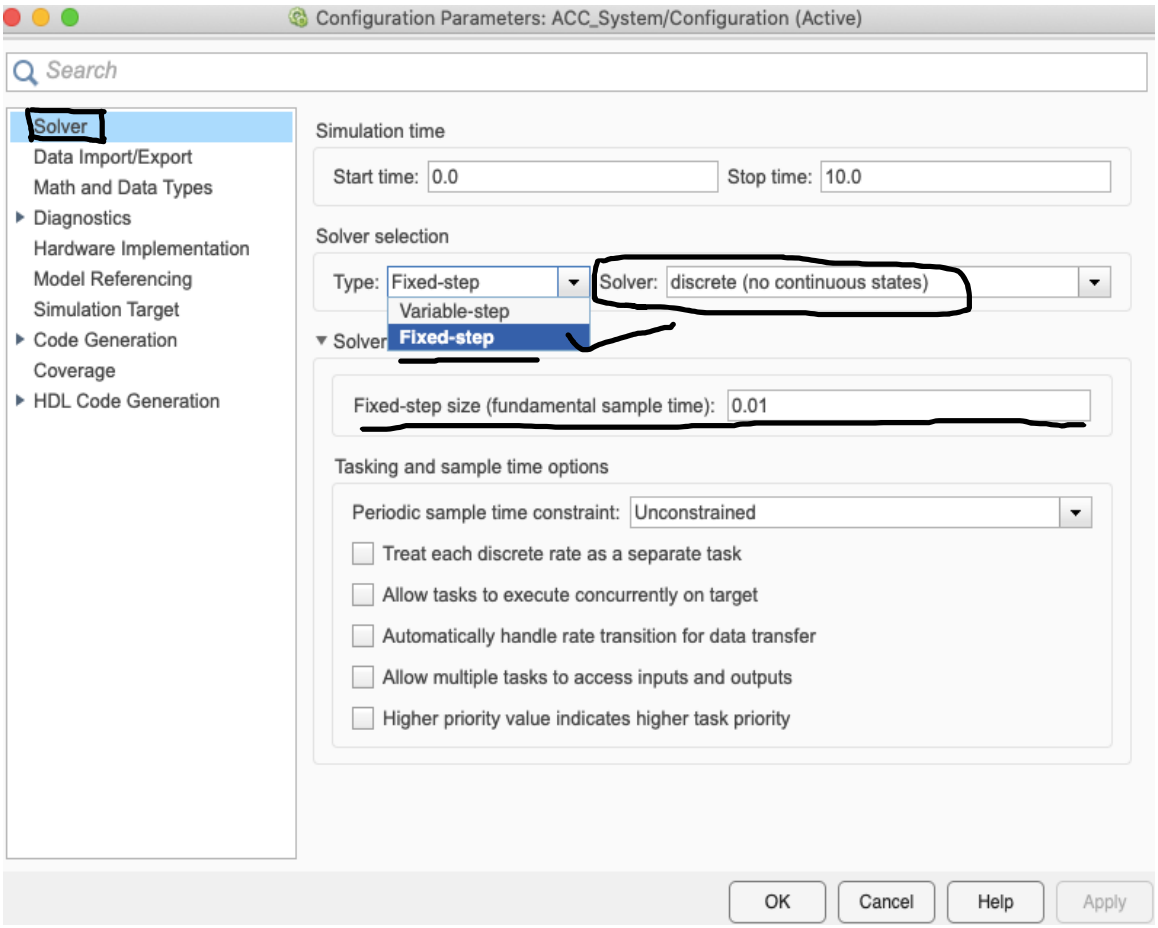
In the Requirement Editor open the Show links to llok out the different requirements which are linked to sifferent subsystem/ blocks.

A. Show Links button in toolbox
B. The number of links attached to their respective subsystem/ blocks.
C. Cutom IDs
D. The Name of the subsystem/ blocks to which that particular requirement is attached.



- A. Circle indicates the number of main requirements for the model. There are 4 different requirements present.
- B. Circle shows the Custom ID or Requirement ID which are generated by the requiments developer at later stage in automotive V cycle.
- C. Circle shows the description of requirements linked to that particular custom ID.

Solver selection:



- The most common practice for model based development, code generation, and MiL-SiL verification purposes; Solver Type should be *Fixed Step* with *fundamental sample time* (as per requirement document) and Solver itself should be either *Auto selection or discrete (no continuous states)*
- Without Fixed step solver selection one is not able to generate the code nor to perform MiL/ SiL verification.

Requirements Report for Vehicle_Speedometer_Module

Table of Contents

List of Tables

Chapter 1. Model Information for "Vehicle_Speedometer_Module"

Table 1.1. Vehicle_Speedometer_Module

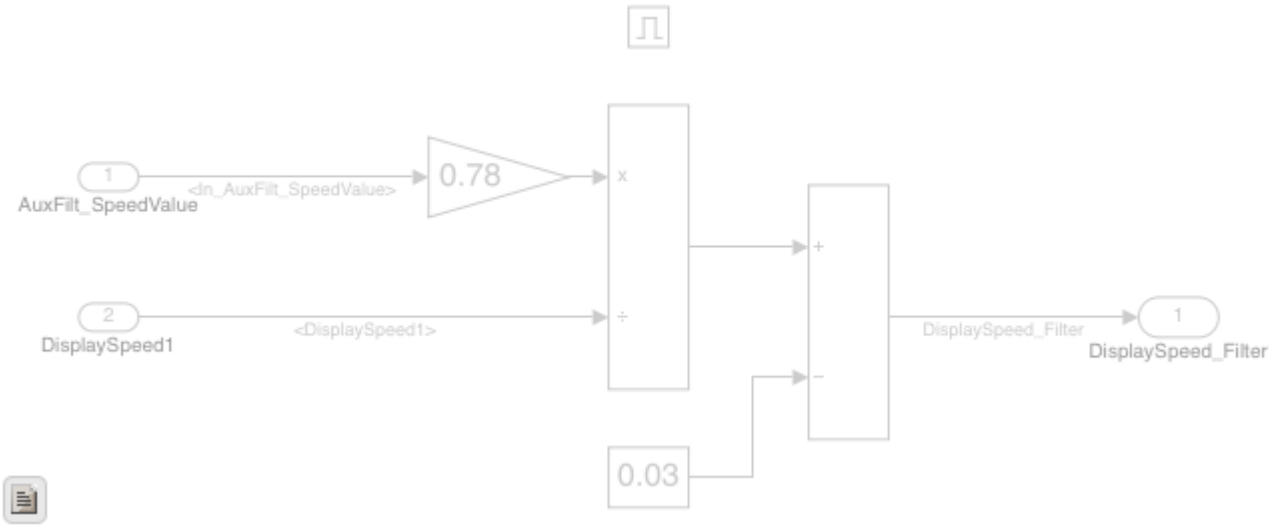
ModelVersion	1.20	*ConfigurationManager*	N/A
Created	Tue Mar 14 11:14:26 2023	*Creator*	jamesbond
LastModifiedDate	Sat Mar 18 17:12:43 2023	*LastModifiedBy*	jamesbond

Chapter 2. Traceability Summary for "Vehicle_Speedometer_Module"

Table 2.1. Artifacts linked in model

ID	Artifact names stored by RMI	Last modified	# links
DOC1	TFT_Cluster_Speedometer_Req.slreqx	Sat Mar 18 17:08:51 2023	4

Chapter 3. System - Auxilliary_Data_Filtering



Show in Simulink

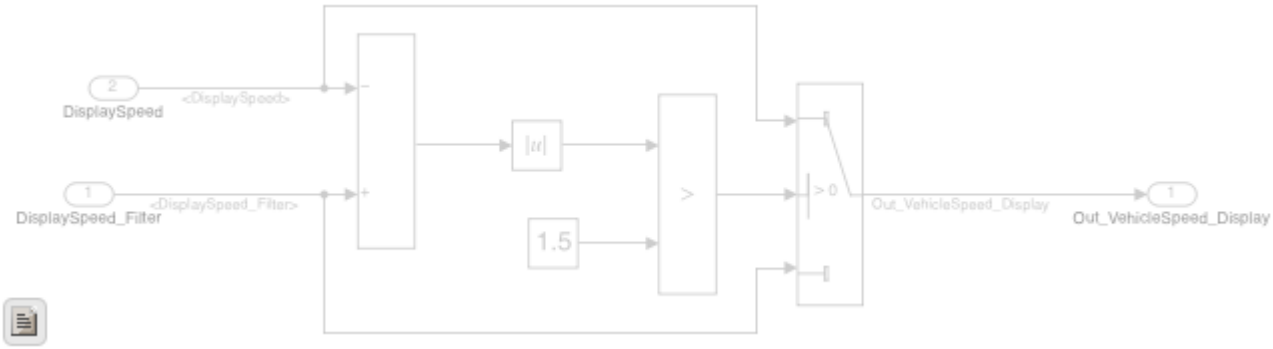
Table 3.1. Vehicle_Speedometer_Module/Speedometer_Module/Auxilliary_Data_Filtering Requirements Data

Link#	Link Description	Link Target (document name and location ID)
1.	"SwReq_TFT_Cluster_Speedometer_03" ------- Details from TFT_Cluster_Speedometer_Req.slreqx: ------ Description: Auxilliary Data Filtering Auxilliary data filtering subsystem will be activated for every 10th milliseconds DisplaySpeed_Filter (t)= ((DisplaySpeed1 (t)*0.78)/ auxilliary filter speed value)-0.03	DOC1, at "3"

Table 3.2. Objects in "Auxilliary_Data_Filtering" that are not linked to requirements

Name	Type
Add	Sum
AuxFilt_SpeedValue	Inport
Constant	Constant
DisplaySpeed1	Inport
DisplaySpeed_Filter	Outport
Divide	Product
Enable	EnablePort
Gain	Gain

Chapter 4. System - Display_Speed_Output



Show in Simulink

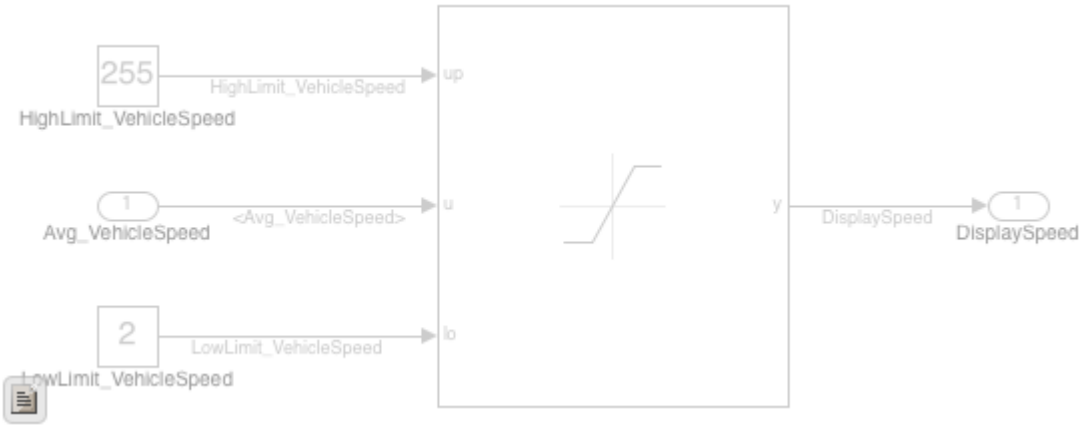
Table 4.1. Vehicle_Speedometer_Module/Speedometer_Module/Display_Speed_Output Requirements Data

Link#	Link Description	Link Target (document name and location ID)
1.	"SwReq_TFT_Cluster_Speedometer_04" ------- Details from TFT_Cluster_Speedometer_Req.slreqx: ---------- Description: TFT Speed Output if (DisplaySpeedFilter - DisplaySpeed)| > 1.5 { Vehicle Speed Display = DisplaySpeed } else { Vehicle Speed Display = DisplaySpeed Filter }	DOC1, at "4"

Table 4.2. Objects in "Display_Speed_Output" that are not linked to requirements

Name	Type
Abs	Abs
Add	Sum
Constant	Constant
DisplaySpeed	Inport
DisplaySpeed_Filter	Inport
Out_VehicleSpeed_Display	Outport
Relational Operator	RelationalOperator
Switch	Switch

Chapter 5. System - Input_Processing



Show in Simulink

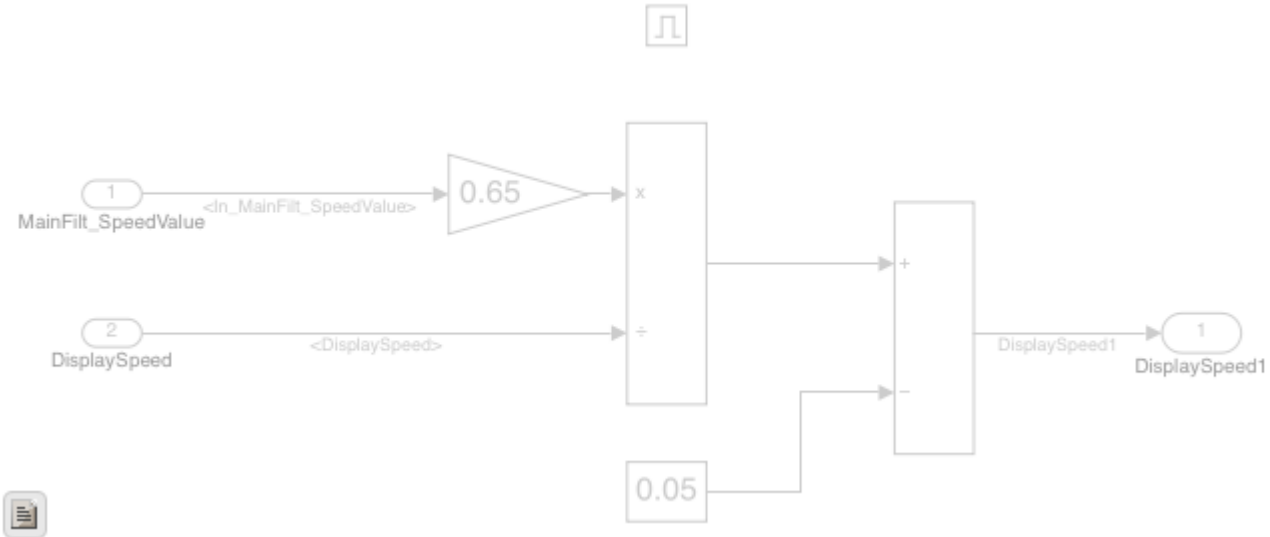
Table 5.1. Vehicle_Speedometer_Module/Speedometer_Module/Input_Processing Requirements Data

Link#	Link Description	Link Target (document name and location ID)
1.	"SwReq_TFT_Cluster_Speedometer_01" ------- Details from TFT_Cluster_Speedometer_Req.slreqx: ---------- Description: Calculating TFT Display Speed Average vehicle speed would be displaying if its between 2 kmph to 250 kmph. Any value outside this range will be discarded by upper and lower limits. When Average Vehicle speed is between 2 kmph & 250 kmph, Display Speed= Average Vehicle Speed. When Average vehicle speed >250 kmph, Display speed= 250 kmph	DOC1, at "1"

Table 5.2. Objects in "Input_Processing" that are not linked to requirements

Name	Type
Avg_VehicleSpeed	Inport
DisplaySpeed	Outport
HighLimit_VehicleSpeed	Constant
LowLimit_VehicleSpeed	Constant

Chapter 6. System - Main_Data_Filtering



Show in Simulink

Table 6.1. Vehicle_Speedometer_Module/Speedometer_Module/Main_Data_Filtering Requirements Data

Link#	Link Description	Link Target (document name and location ID)

Link#	Link Description	Link Target (document name and location ID)
1.	"SwReq_TFT_Cluster_Speedometer_02" ------- Details from TFT_Cluster_Speedometer_Req.slreqx: ------ Description: Main Data Filtering Main data filtering subsystem will be activated for every 10th milliseconds DisplaySpeed1 (t)= ((DisplaySpeed (t)*0.65)/ Main Filter Speed value)- 0.05	DOC1, at "2"

Table 6.2. Objects in "Main_Data_Filtering" that are not linked to requirements

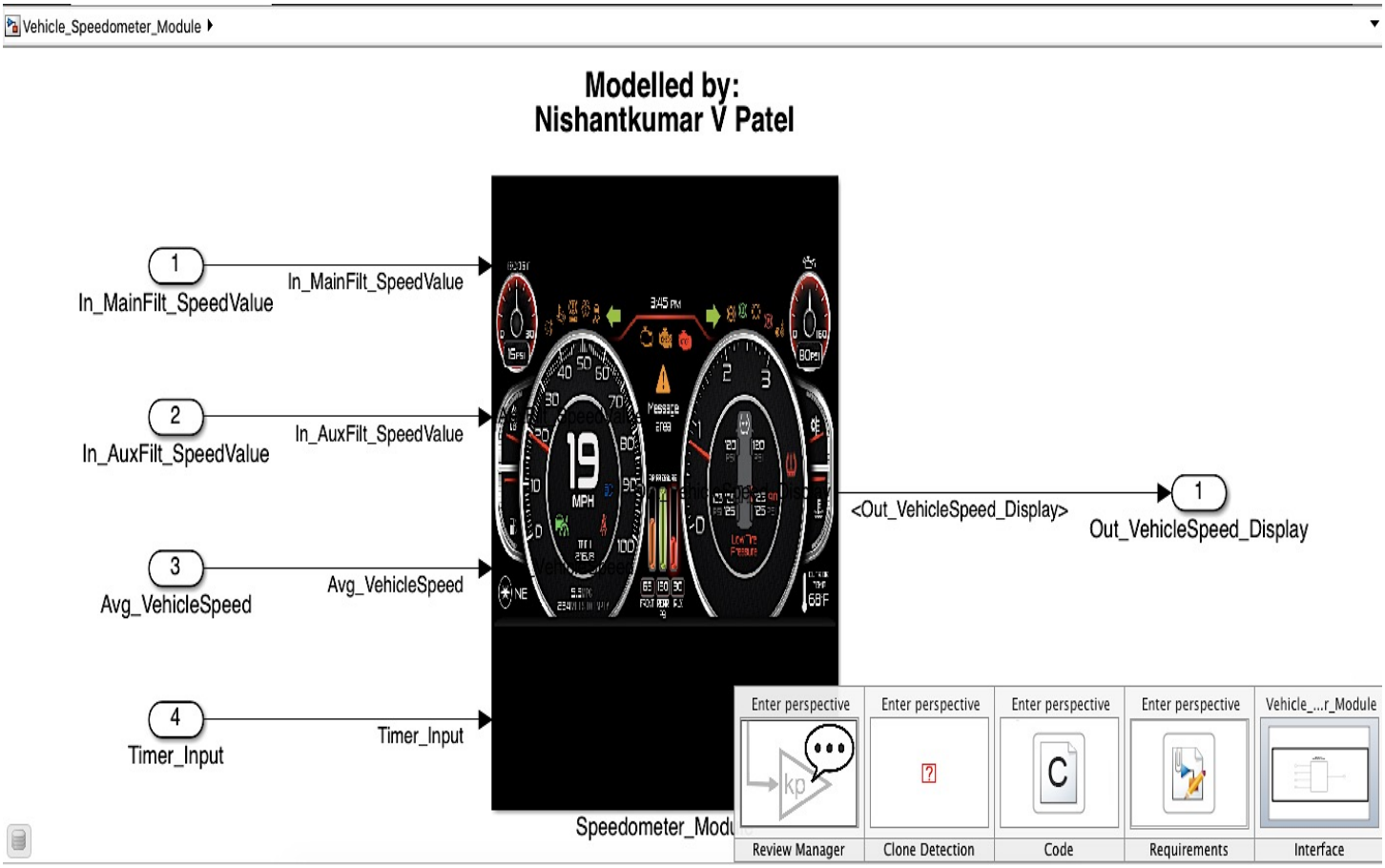
Name	Type
Add	Sum
Constant	Constant
DisplaySpeed	Inport
DisplaySpeed1	Outport
Divide	Product
Enable	EnablePort
Gain	Gain
MainFilt_SpeedValue	Inport

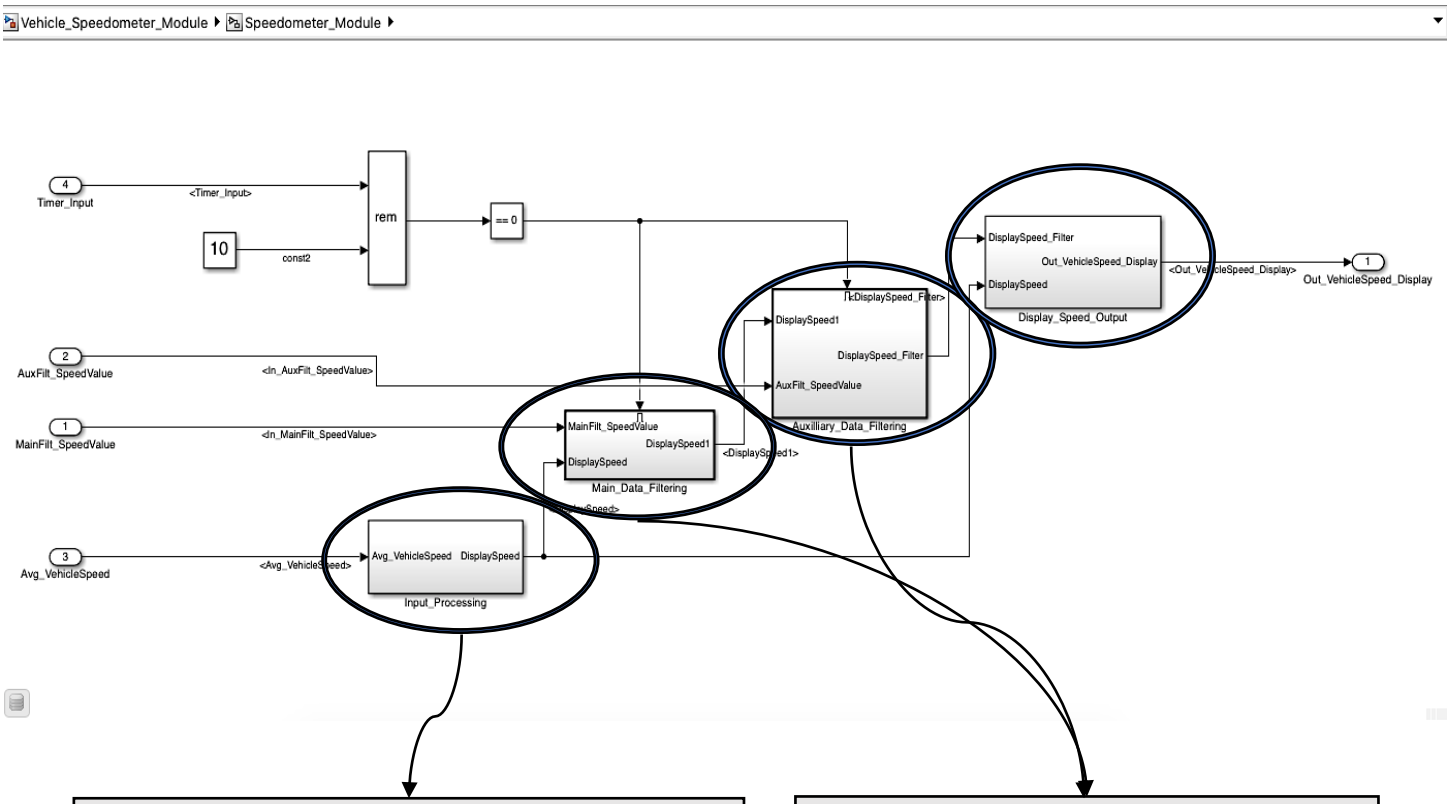
Chapter 7. Systems in "Vehicle_Speedometer_Module" that have no links to requirements

Table 7.1. Systems and subsystem blocks in "Vehicle_Speedometer_Module" that have no links to requirements

Model or subsystem block	Children with links
Vehicle_Speedometer_Module	None
Vehicle_Speedometer_Module/Speedometer_Module	4 out of 12
Vehicle_Speedometer_Module/Speedometer_Module/Compare To Zero	None
Vehicle_Speedometer_Module/Speedometer_Module/Input_Processing/Saturation Dynamic	None

3. Model Development in Simulink





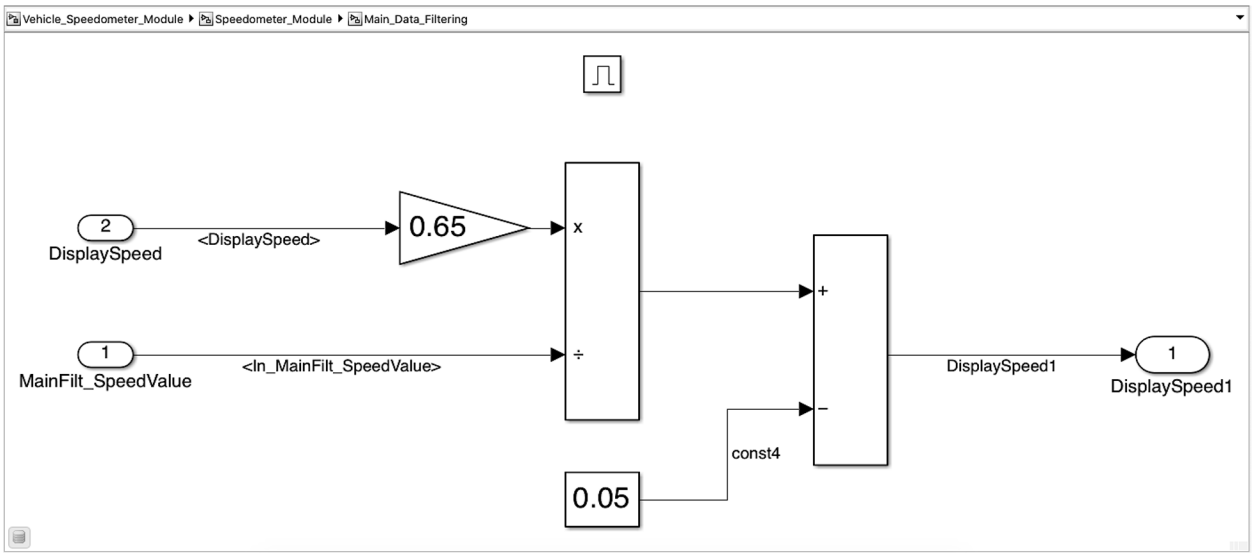
Average Vehicle Speed would be displayed only if its between 2 kmph & 250 kmph. Any value outside this range will be discarded by its lower limit & upper limit

When Average Vehicle Speed is < 2 kmph, DisplaySpeed = 2 kmph

When Average Vehicle Speed is in between 2 kmph & 250 kmph, DisplaySpeed = Average Vehicle Speed

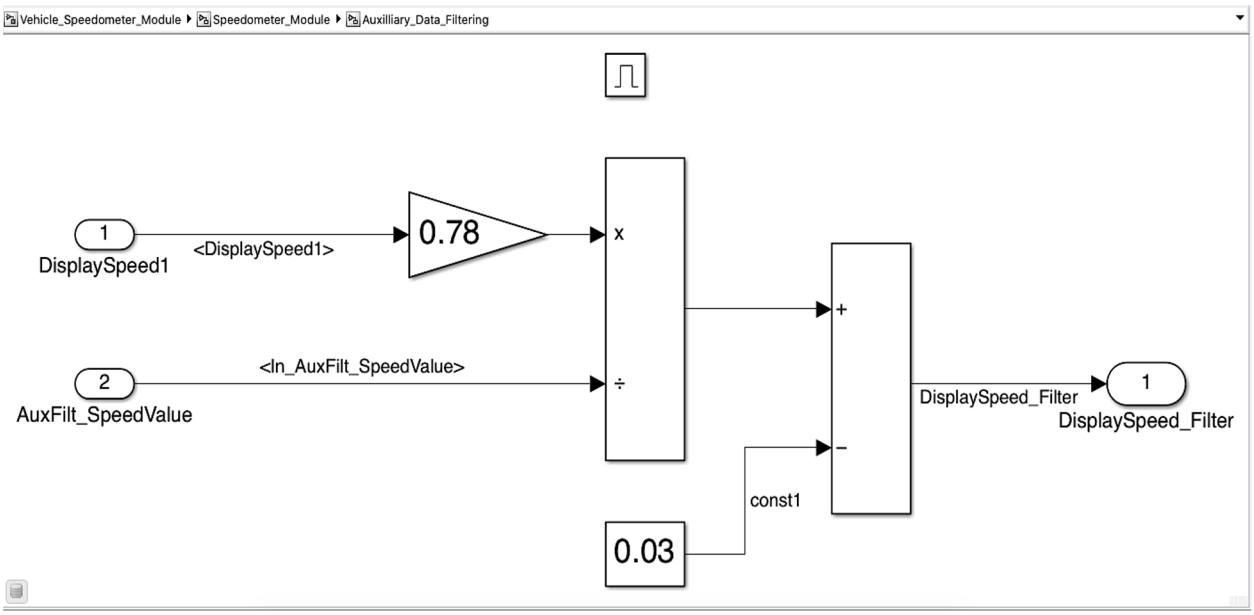
When Average Vehicle Speed is > 250 kmph, DisplaySpeed = 250 kmph

This two main and auxilliary data filtering subsystem can only be enabled every 10th milliseconds of simulation time.



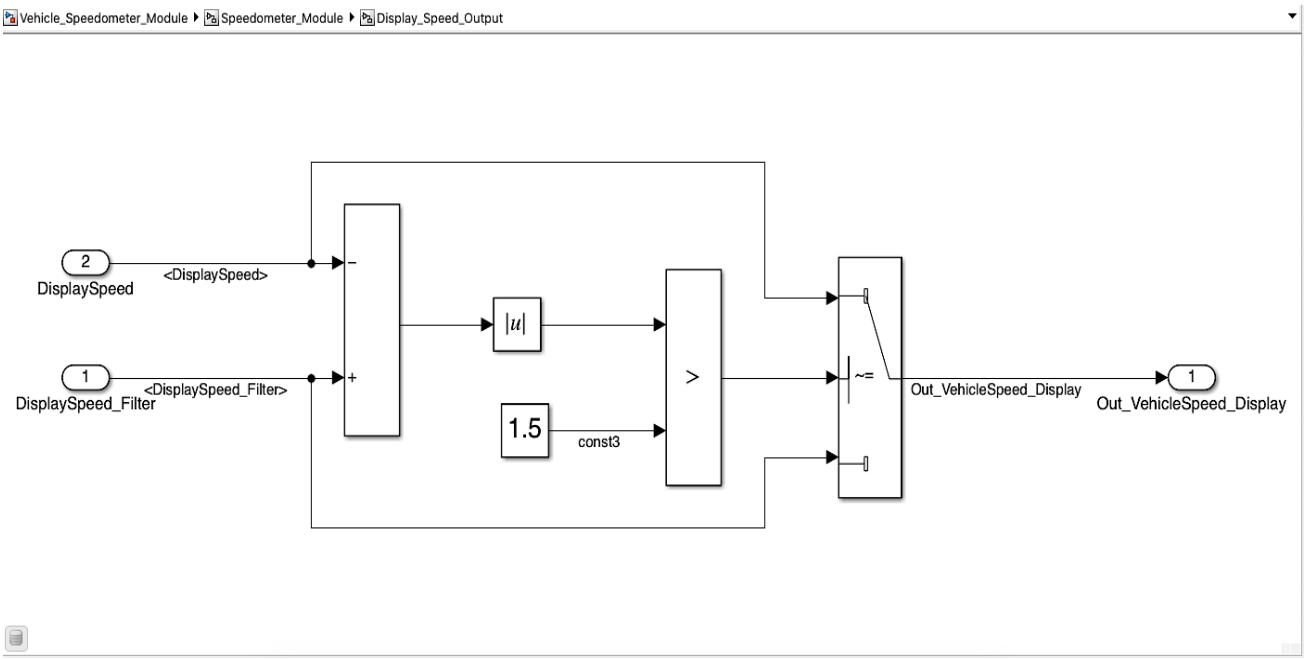
Main Data Filtering subsystem will be activated for every 10th millisecond

DisplaySpeed1 (t) = ((DisplaySpeed (t) x 0.65)/ Main Filter Speed Value) 0.05



Auxiliary Data Filtering subsystem will be activated for every 10th millisecond

DisplaySpeed_Filter (t) = ((DisplaySpeed1 (t) x 0.78)/ Auxillary Filter Speed Value) - 0.03



If |(DisplaySpeedFilter - DisplaySpeed)| > 1.5

{

Vehicle Speed Display = DisplaySpeed

}

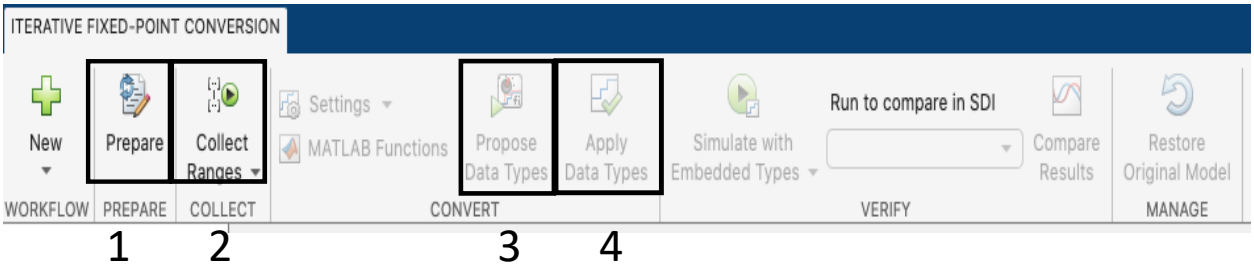
else

{

Vehicle Speed Display = DisplaySpeed Filter

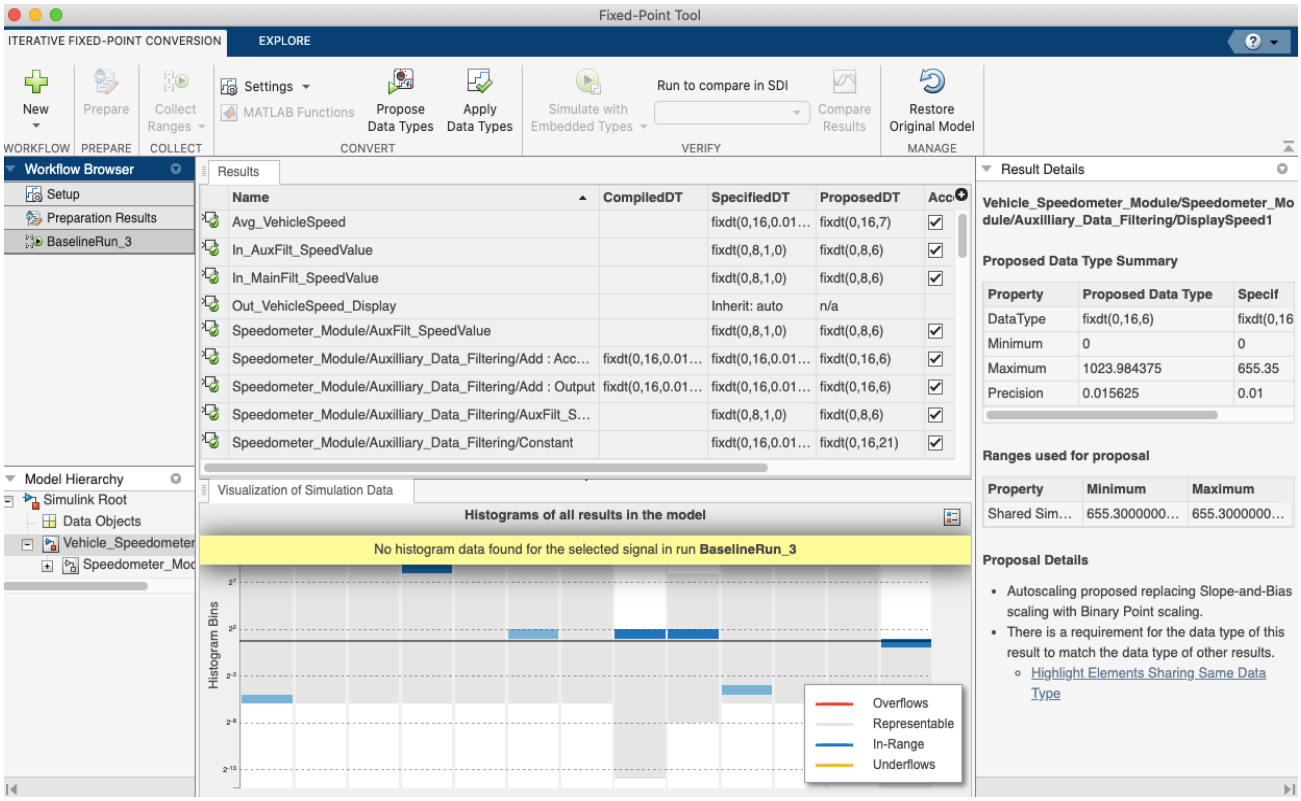
}

Fixed Point Tool:



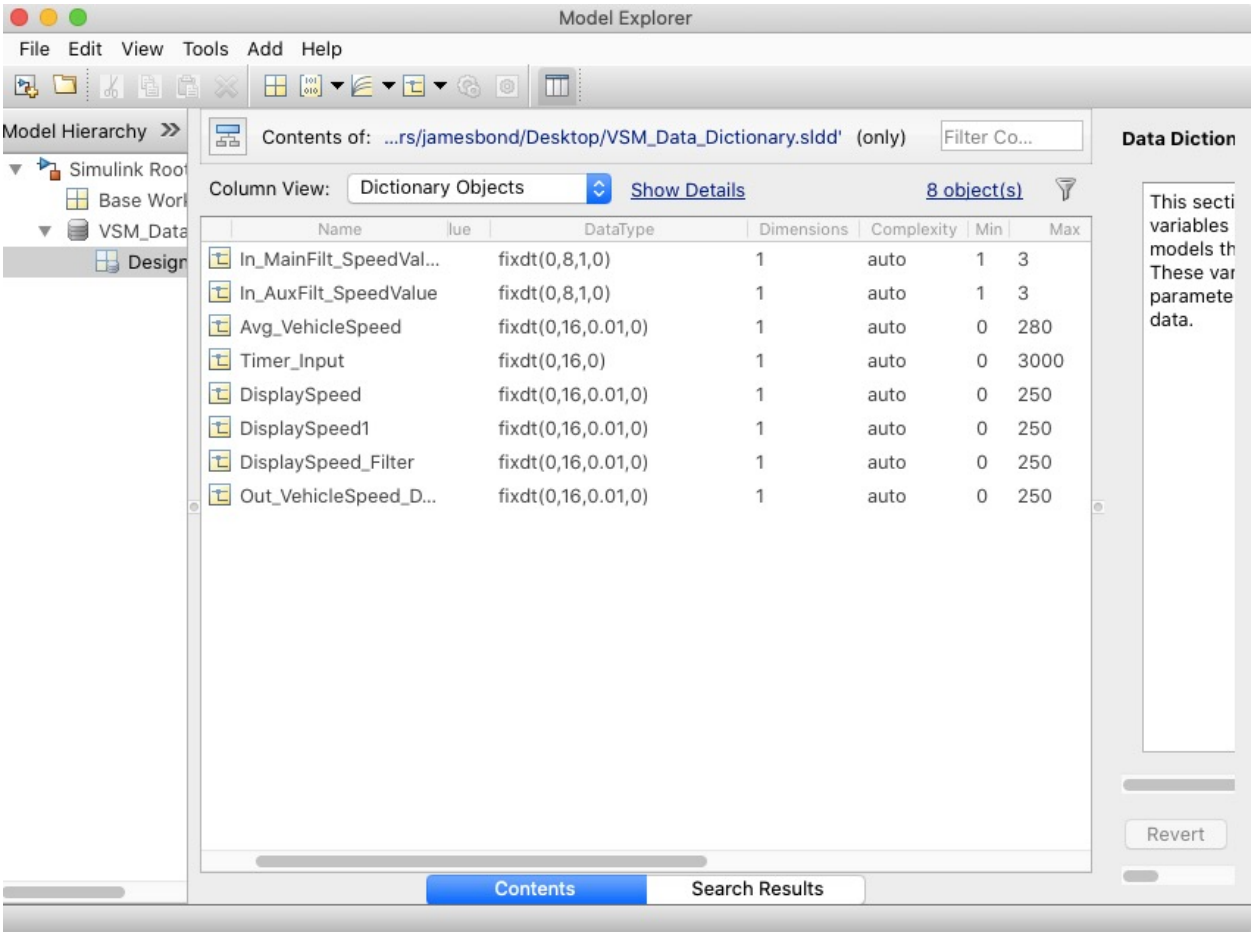
1. Marked box is showing the *Prepare* button to initialize the fixed point tool over the present simulink model.
2. Marked square box is showing the *Collect Ranges* button to collect the existing paramets information, data types of sifferent blocks, value, etc.
3. Setps is the *Proposed Data Types* which is used to proposed the new fixed point data type based upon the existing datatypes, usage, ranges of different blocks.
4. Final step is the *Apply Data Types*, which ever data type the Fixed point tool proposed will be applied to each and every blocks.

- Iterative Conversion approach is used.



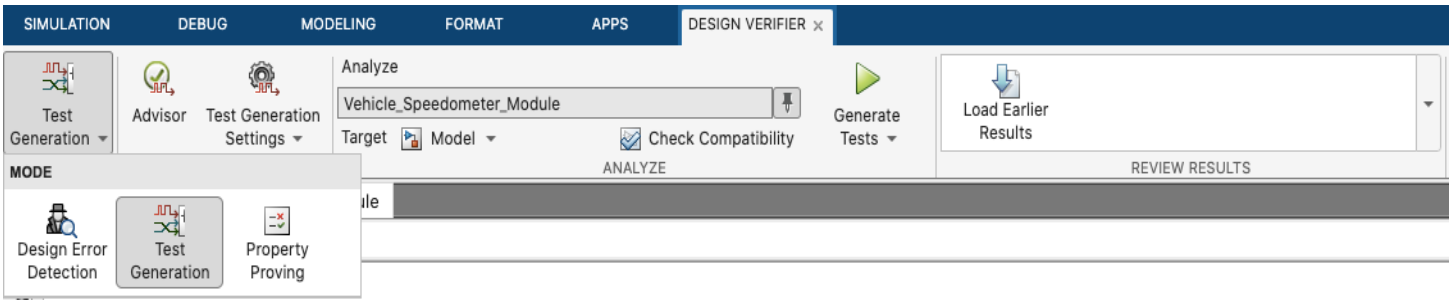
- Fixed point toolbox is used to prevent the data and memory loss occurred in the model and this data and memory loss will be reflect into C-code which is ultimately tedious for ECU.
- So, for that reasons fixed point concept hase to be implemented for each and every signals throughout the model in order to have neat and better code in hand.
- Fixed point tool box is used to predict the fixed point data type's value. What has to be the value for every single signals.

4. Simulink Data Dictionary



- Data dictionary is used for creating the attributes and variables used at model level in different signals and block parameter values.
- As you can see the input signals have the storage classs of *Imported Extern* while oputput signals have storage class of *Exported Global*.
- The other signals within the model are like to be more local so those were defined as *Localizeable*.

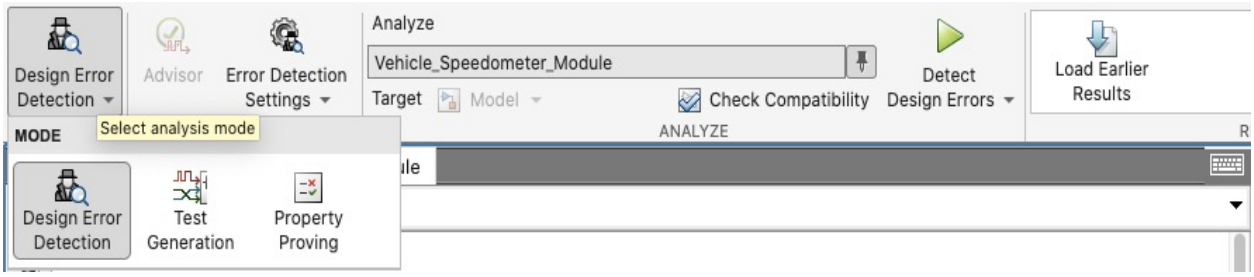
5. Simulink Design Verifier



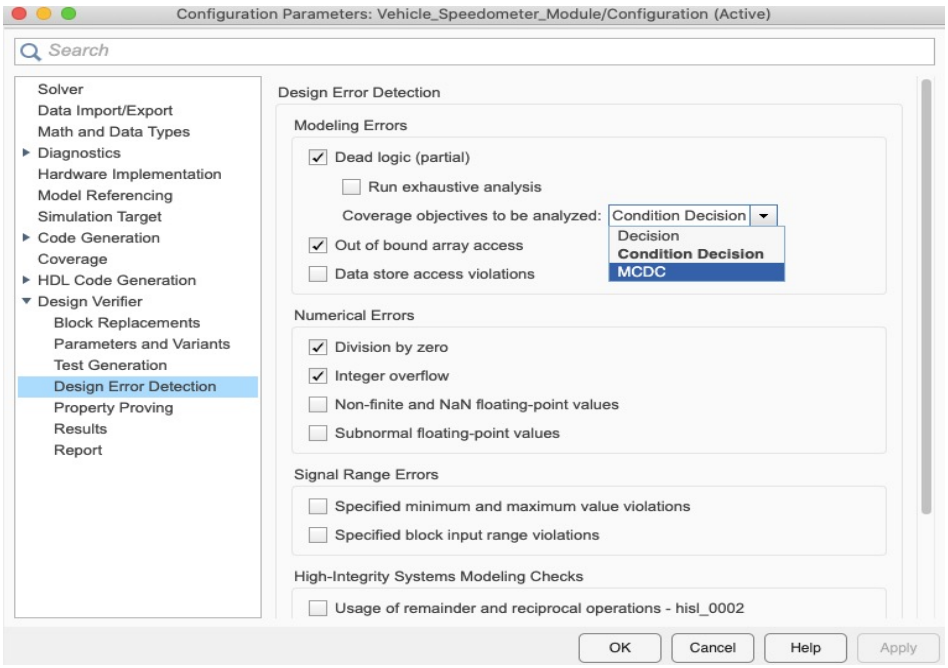
- The above picture is the design verifier toolbox. The main usages of design verifier is for *Test generation* and *Design error detection.*

- It is standard to firstly go for Design error detection then for Test generation mode. Design error detection usually done for errors like *division by zero, integer overflow, and dead logics.*
- Then after performing the design error detection operation, Test generation can be conducted where the simulink compiler automatically generates the test cases for model testing and then generate the report for coverage analysis.

Design Error Detection



open Apps>Design Verifier> Toolbox> Design Error Detection



- Open the Model configuration parameters>Design Verifier>Design Error Detection.
- Design error detection mode includes the different type of errors presented in simulink model such as Division by *Zero, Integer Overflow, Dead Logics* are the most common selected for detection process.
- Coverage analysis for dead logic can be choose as per the needs *MCDC, Condition Decision or Decision*.

Simulink Design Verifier Report

Vehicle_Speedometer_Module

jamesbond

19-Mar-2023 02:44:45

Table of Contents

Chapter 1. Summary

Analysis Information

Model:	Vehicle_Speedometer_Module
Release:	R2022a Update 2
Checksum:	379700363 2376830566 2080933560 886107326
Mode:	Design error detection
Model Representation:	Built on 19-Mar-2023 02:43:15
Status:	Completed normally
PreProcessing Time:	16s
Analysis Time:	19s

Objectives Status

Number of Objectives:	**26**	
Objectives Valid:	10	(38%)
Dead Logic:	0	(0%)

Chapter 2. Analysis Information

Table of Contents

2.1. Model Information

File:	Vehicle_Speedometer_Module
Version:	1.29
Time Stamp:	Sun Mar 19 02:09:56 2023
Author:	jamesbond

2.2. Analysis Options

Mode:	DesignErrorDetection
Rebuild Model Representation:	IfChangeIsDetected
Detect dead logic (partial):	on
Run exhaustive analysis for dead logic:	off
Coverage objectives analyzed for dead logic:	ConditionDecision
Detect integer overflow:	on
Detect division by zero:	on
Detect specified minimum and maximum value violations:	off
Detect out of bound array access:	on
Detect non-finite and NaN floating-point values:	off
Detect subnormal floating-point values:	off
Detect data store access violations:	off
Detect specified block input range violations:	off
Detect usage of remainder and reciprocal operations (hisl_0002):	off
Detect usage of square root operations (hisl_0003):	off
Detect usage of log and log10 operations (hisl_0004):	off
Detect usage of Reciprocal Square Root blocks (hisl_0028):	off
Maximum Analysis Time:	300s
Block Replacement:	off
Parameters Analysis:	off
Include expected output values:	off
Randomize data that do not affect the outcome:	off
Additional analysis to reduce instances of rational approximation:	on

Save Data:	on
Save Harness:	off
Save Report:	off

2.3. Constraints

Table of Contents

2.3.1. Design Min Max Constraints

Name	Design Min Max Constraint
Avg_VehicleSpeed	[0..280]
In_MainFilt_SpeedValue	[1..3]
In_AuxFilt_SpeedValue	[1..3]

Chapter 3. Dead Logic

Simulink Design Verifier proved these objectives to be unreachable or dead logic. This can be a side effect of parameter configurations or minimum and maximum constraints specified on inputs. Simulink Design Verifier ran a partial check for dead logic. Consider enabling the 'Dead logic > Run exhaustive analysis' configuration option in order to perform an exhaustive analysis.

Chapter 4. Design Error Detection Objectives Status

Table of Contents

4.1. Objectives Valid

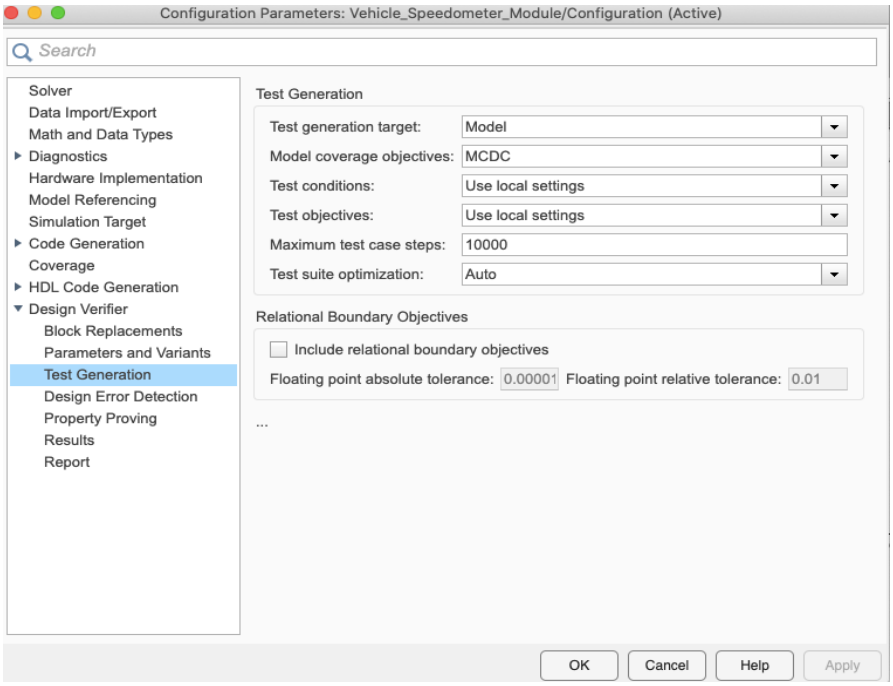
#	Type	Model Item	Description	Analysis Time (sec)
17	Integer overflow	Speedometer_Module/Main_Data_Filtering/Gain	Overflow	12
22	Division by zero	Speedometer_Module/Main_Data_Filtering/Divide	Division by zero	12
23	Integer overflow	Speedometer_Module/Main_Data_Filtering/Divide	Overflow	12
26	Integer overflow	Speedometer_Module/Auxilliary_Data_Filtering/Gain	Overflow	12
28	Integer overflow	Speedometer_Module/Main_Data_Filtering/Add	Overflow	12
30	Integer overflow	Speedometer_Module/Display_Speed_Output/Add	Overflow	12
33	Division by zero	Speedometer_Module/Auxilliary_Data_Filtering/Divide	Division by zero	12
34	Integer overflow	Speedometer_Module/Auxilliary_Data_Filtering/Divide	Overflow	12
38	Integer overflow	Speedometer_Module/Display_Speed_Output/Abs	Overflow	12
40	Integer overflow	Speedometer_Module/Auxilliary_Data_Filtering/Add	Overflow	16

Chapter 5. Derived Ranges

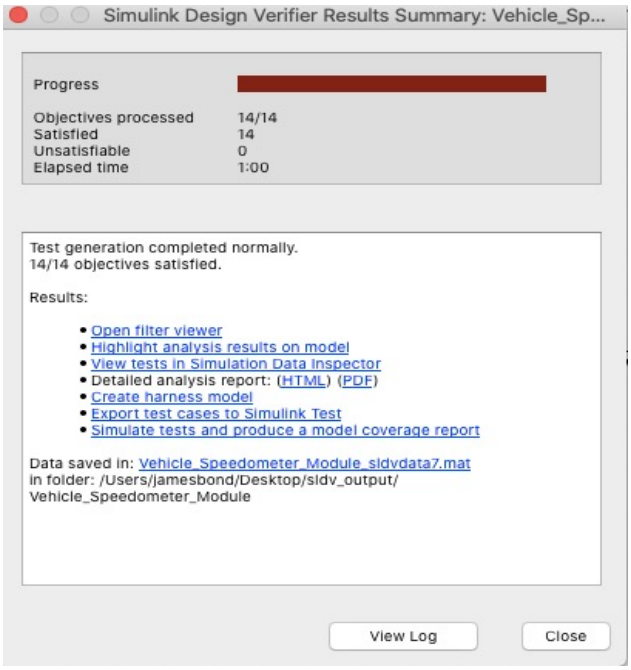
Signal	Derived Ranges
Speedometer_Module/Compare To Zero/Constant- Outport 1	0
Speedometer_Module/Constant- Outport 1	10
Speedometer_Module/Display_Speed_Output/Constant- Outport 1	1.5
Avg_VehicleSpeed- Outport 1	[0..280]
Speedometer_Module/Input_Processing/Saturation- Outport 1	[2..250]
Speedometer_Module/Rem- Outport 1	[0..9]
Speedometer_Module/Compare To Zero/Compare- Outport 1	[F..T]
Speedometer_Module/Main_Data_Filtering/Constant- Outport 1	0.05000000000000000027755575615628913510590791702270507812 5
Speedometer_Module/Main_Data_Filtering/Gain- Outport 1	[1.29000000000000003552713678800500929355621337890625..162.490000000000009094947017729 28
In_MainFilt_SpeedValue- Outport 1	[1..3]
Speedometer_Module/Main_Data_Filtering/Divide- Outport 1	[0.42999999999999999333866185224906075745820999145507812 5..162.49000000000000909494701 77
Speedometer_Module/Auxilliary_Data_Filtering/Constant- Outport 1	0.02999999999999999888977697537484345957636833190917968 75
Speedometer_Module/Auxilliary_Data_Filtering/Gain- Outport 1	[0.28999999999999998001598555674718227237462997436523437 5..126.68999999999999772626324 55
Speedometer_Module/Main_Data_Filtering/Add- Outport 1	[0.38000000000000000444089209850062616169452667236328125..162.43999999999999772626324 556
Speedometer_Module/Display_Speed_Output/Add- Outport 1	[-250..124.65999999999999658939486835151910781860351562 5]
In_AuxFilt_SpeedValue- Outport 1	[1..3]

Signal	Derived Ranges
Speedometer_Module/Auxilliary_Data_Filtering/Divide-Outport 1	[0.089999999999999996669330926124530378729104995727539062 5..126.68999999999999772626324 5
Speedometer_Module/Display_Speed_Output/Abs-Outport 1	[0..250]
Speedometer_Module/Auxilliary_Data_Filtering/Add-Outport 1	[0.059999999999999997779553950749686919152736663818359375..126.65999999999999658939486 83
Speedometer_Module/Display_Speed_Output/Relational Operator- Outport 1	[F..T]
Speedometer_Module/Display_Speed_Output/Switch-Outport 1	[0.5..250]
Out_VehicleSpeed_Display- Outport 1	[0.5..250]

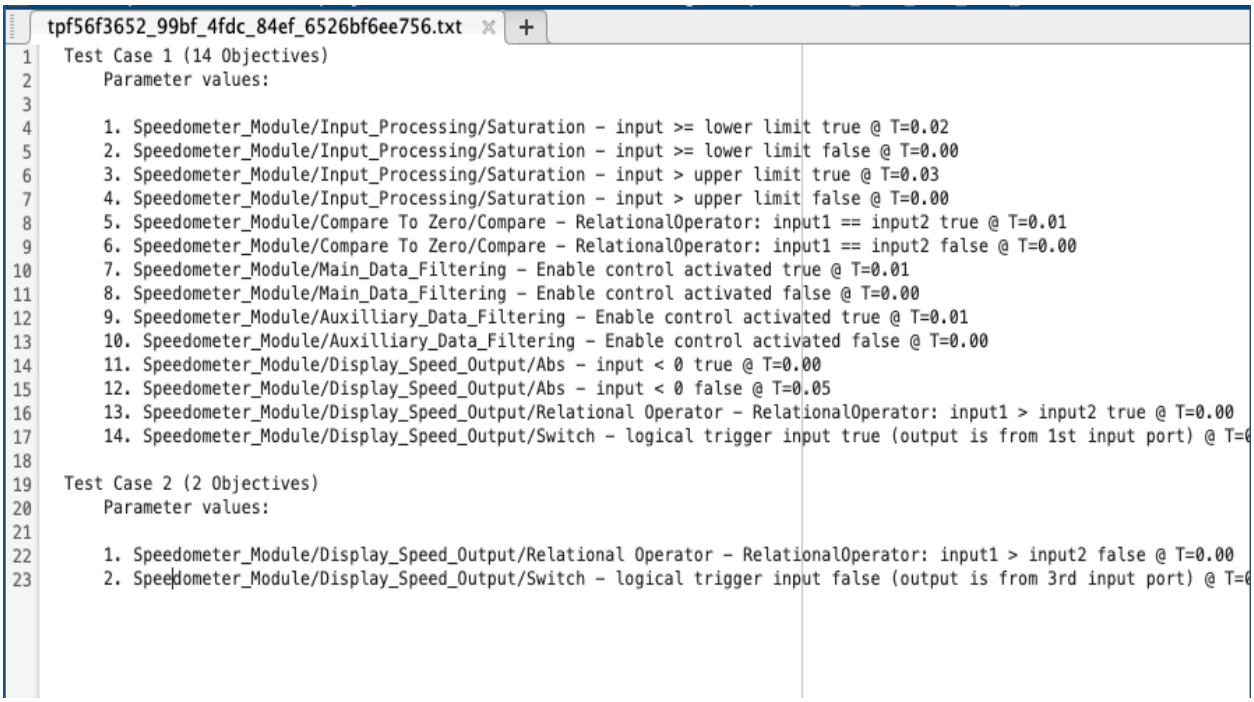
Test Generation



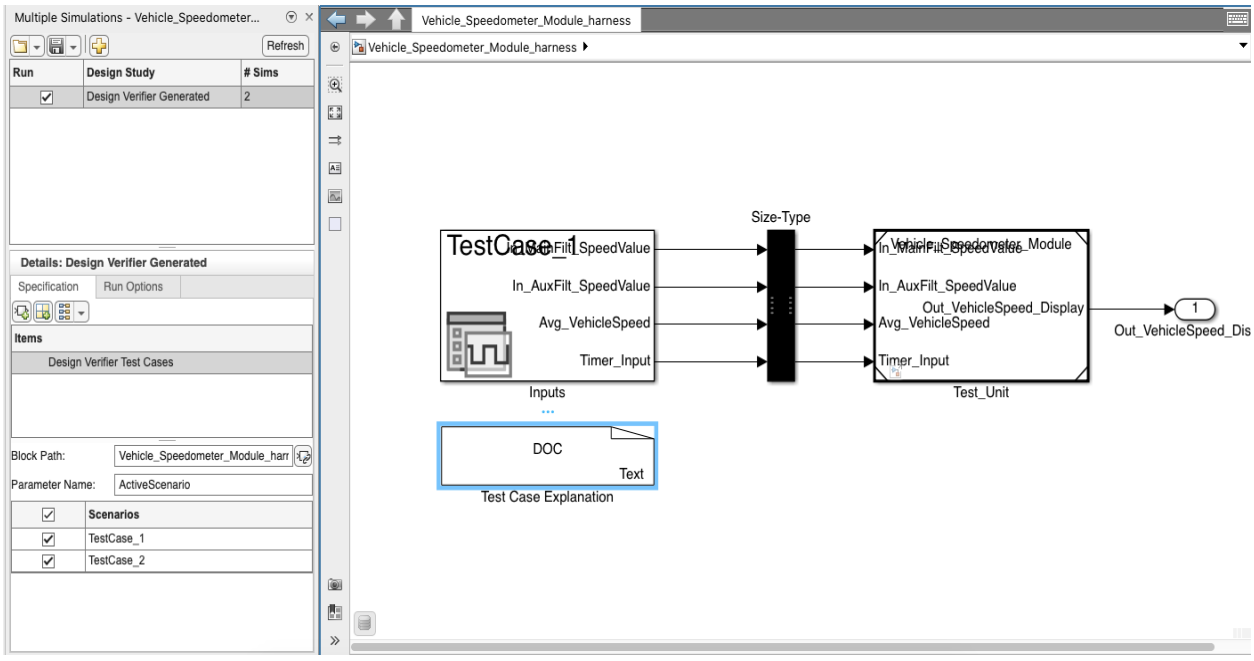
- open Model configuration settings> Design Verifier>Test Generation
- Test conditions, test objectives, and maximum test case steps can be modified. Test suite optimization is usually kept as Auto.



- This is the result pop up window open after running the Test generation mode. It shows all the 14 out of 14 are satisfiable objectives.



- Generated 2 test cases containing different parameters are written in command window.



- This is the harness model of test generation creating after running the test generation using Design Verifier app. As one can see total 2 test cases are created automatically by Design Verifier.

Simulink Design Verifier Report

Vehicle_Speedometer_Module

jamesbond

19-Mar-2023 02:54:03

Table of Contents

Chapter 1. Summary

Analysis Information

Model:	Vehicle_Speedometer_Module
Release:	R2022a Update 2
Checksum:	4081228814 4271405805 3120410839 383593881
Mode:	Test generation
Model Representation:	Built on 19-Mar-2023 02:52:11
Test Generation Target:	Model
Status:	Completed normally
PreProcessing Time:	19s
Analysis Time:	60s

Objectives Status

Number of Objectives:	**14**	
Objectives Satisfied:	14	(100%)

Chapter 2. Analysis Information

Table of Contents

2.1. Model Information

File:	Vehicle_Speedometer_Module
Version:	1.29
Time Stamp:	Sun Mar 19 02:09:56 2023
Author:	jamesbond

2.2. Analysis Options

Mode:	TestGeneration
Rebuild Model Representation:	IfChangeIsDetected
Test Generation Target:	Model
Test Suite Optimization:	Auto
Maximum Testcase Steps:	10000time steps
Test Conditions:	UseLocalSettings
Test Objectives:	UseLocalSettings
Model Coverage Objectives:	MCDC
Add tests for the missing coverage:	off
Include Relational Boundary Objectives:	off
Maximum Analysis Time:	300s
Block Replacement:	off
Parameters Analysis:	off
Include expected output values:	off
Randomize data that do not affect the outcome:	off
Additional analysis to reduce instances of rational approximation:	on
Save Data:	on
Save Harness:	off
Save Report:	off

2.3. User Artifacts

Coverage Data: n/a
Test Data: n/a

2.4. Constraints

Table of Contents

2.4.1. Design Min Max Constraints

Name	Design Min Max Constraint
Avg_VehicleSpeed	[0..280]
In_MainFilt_SpeedValue	[1..3]
In_AuxFilt_SpeedValue	[1..3]

Chapter 3. Test Objectives Status

Table of Contents

3.1. Objectives Satisfied

Simulink Design Verifier generated test cases that exercise these test objectives.

#	Type	Model Item	Description	Analysis Time (sec)	Test Case
1	Decision	Speedometer_Module/Input_Processing/Saturation	input >= lower limit **true**	28	1
2	Decision	Speedometer_Module/Input_Processing/Saturation	input >= lower limit **false**	28	1
3	Decision	Speedometer_Module/Input_Processing/Saturation	input > upper limit **true**	28	1
4	Decision	Speedometer_Module/Input_Processing/Saturation	input > upper limit **false**	28	1
5	Condition	Speedometer_Module/Compare To Zero/Compare	RelationalOperator: input1 == input2 **true**	28	1
6	Condition	Speedometer_Module/Compare To Zero/Compare	RelationalOperator: input1 == input2 **false**	28	1
7	Decision	Speedometer_Module/Main_Data_Filtering	Enable control activated **true**	28	1
8	Decision	Speedometer_Module/Main_Data_Filtering	Enable control activated **false**	28	1
9	Decision	Speedometer_Module/Auxilliary_Data_Filtering	Enable control activated **true**	28	1
10	Decision	Speedometer_Module/Auxilliary_Data_Filtering	Enable control activated **false**	28	1
11	Condition	Speedometer_Module/Display_Speed_Output/Relational Operator	RelationalOperator: input1 > input2 **true**	28	1
12	Condition	Speedometer_Module/Display_Speed_Output/Relational Operator	RelationalOperator: input1 > input2 **false**	53	2
13	Decision	Speedometer_Module/Display_Speed_Output/Switch	logical trigger input **false (output is from 3rd input port)**	53	2
14	Decision	Speedometer_Module/Display_Speed_Output/Switch	logical trigger input **true (output is from 1st input port)**	28	1

Chapter 4. Model Items

Table of Contents

This section presents, for each object in the model defining coverage objectives, the list of objectives and their individual status at the end of the analysis. It should match the coverage report obtained from running the generated test suite on the model, either from the harness model or by using the sldvruntest command.

4.1. Speedometer_Module/Input_Processing/Saturation

View

#:	Type	Description	Status	Test Case
1	Decision	input >= lower limit true	Satisfied	1
2	Decision	input >= lower limit false	Satisfied	1
3	Decision	input > upper limit true	Satisfied	1
4	Decision	input > upper limit false	Satisfied	1

4.2. Speedometer_Module/Compare To Zero/Compare

View

#:	Type	Description	Status	Test Case
5	Condition	RelationalOperator: input1 == input2 true	Satisfied	1
6	Condition	RelationalOperator: input1 == input2 false	Satisfied	1

4.3. Speedometer_Module/Main_Data_Filtering

View

#:	Type	Description	Status	Test Case
7	Decision	Enable control activated true	Satisfied	1
8	Decision	Enable control activated false	Satisfied	1

4.4. Speedometer_Module/Auxilliary_Data_Filtering

View

#:	Type	Description	Status	Test Case
9	Decision	Enable control activated true	Satisfied	1
10	Decision	Enable control activated false	Satisfied	1

4.5. Speedometer_Module/Display_Speed_Output/Relational Operator

View

#:	Type	Description	Status	Test Case
11	Condition	RelationalOperator: input1 > input2 true	Satisfied	1
12	Condition	RelationalOperator: input1 > input2 false	Satisfied	2

4.6. Speedometer_Module/Display_Speed_Output/Switch

View

#:	Type	Description	Status	Test Case
13	Decision	logical trigger input false (output is from 3rd input port)	Satisfied	2
14	Decision	logical trigger input true (output is from 1st input port)	Satisfied	1

Chapter 5. Test Cases

Table of Contents

This section contains detailed information about each generated test case.

5.1. Test Case 1

Summary

Length: 0.03 second (4 sample periods)
Objectives Satisfied: 12

Objectives

Step	Time	Model Item	Objectives
1	0	Speedometer_Module/Input_Processing/Saturation Speedometer_Module/Input_Processing/Saturation Speedometer_Module/Compare To Zero/Compare Speedometer_Module/Main_Data_Filtering Speedometer_Module/Auxilliary_Data_Filtering Speedometer_Module/Display_Speed_Output/Relational Operator Speedometer_Module/Display_Speed_Output/Switch	2. input >= lower limit false 4. input > upper limit false 6. RelationalOperator: input1 == input2 false 8. Enable control activated false 10. Enable control activated false 11. RelationalOperator: input1 > input2 true 14. logical trigger input true (output is from 1st input port)
2	0.01	Speedometer_Module/Compare To Zero/Compare Speedometer_Module/Main_Data_Filtering Speedometer_Module/Auxilliary_Data_Filtering	5. RelationalOperator: input1 == input2 true 7. Enable control activated true 9. Enable control activated true
3	0.02	Speedometer_Module/Input_Processing/Saturation	1. input >= lower limit true
4	0.03	Speedometer_Module/Input_Processing/Saturation	3. input > upper limit true

Generated Input Data

Time	0	0.01	0.02
Step	1	2	3
In_MainFilt_SpeedValue	1	3	3

Time	0	0.01	0.02
Step	**1**	**2**	**3**
In_AuxFilt_SpeedValue	3	3	3
Avg_VehicleSpeed	0.979999999999999982236431605997495353221893310546875	1.689999999999999946709294817992486059665679931640625	5.639999
Timer_Input	69	0	0

5.2. Test Case 2

Summary

Length: 0.01 second (2 sample periods)
Objectives Satisfied: 2

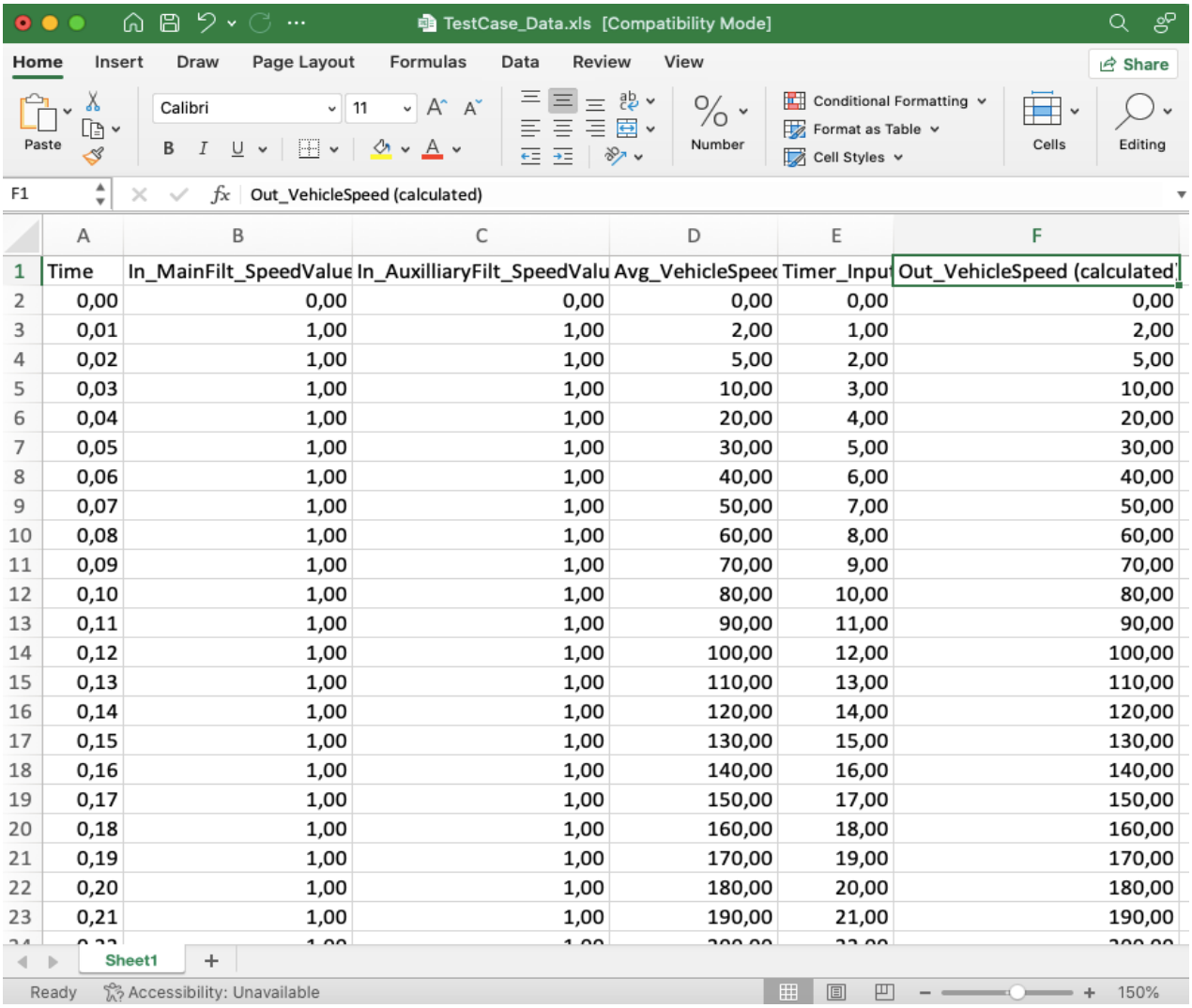
Objectives

Step	Time	Model Item	Objectives
2	0.01	Speedometer_Module/Display_Speed_Output/Relational Operator Speedometer_Module/Display_Speed_Output/Switch	12. RelationalOperator: input1 > input2 false 13. logical trigger input false (output is from 3rd input port)

Generated Input Data

Time	0	0.01
Step	**1**	**2**
In_MainFilt_SpeedValue	1	2
In_AuxFilt_SpeedValue	3	3
Avg_VehicleSpeed	159.180000000000006821210263296961784362792968750	25.420000000000001705302565824240446090698242187500
Timer_Input	310	311

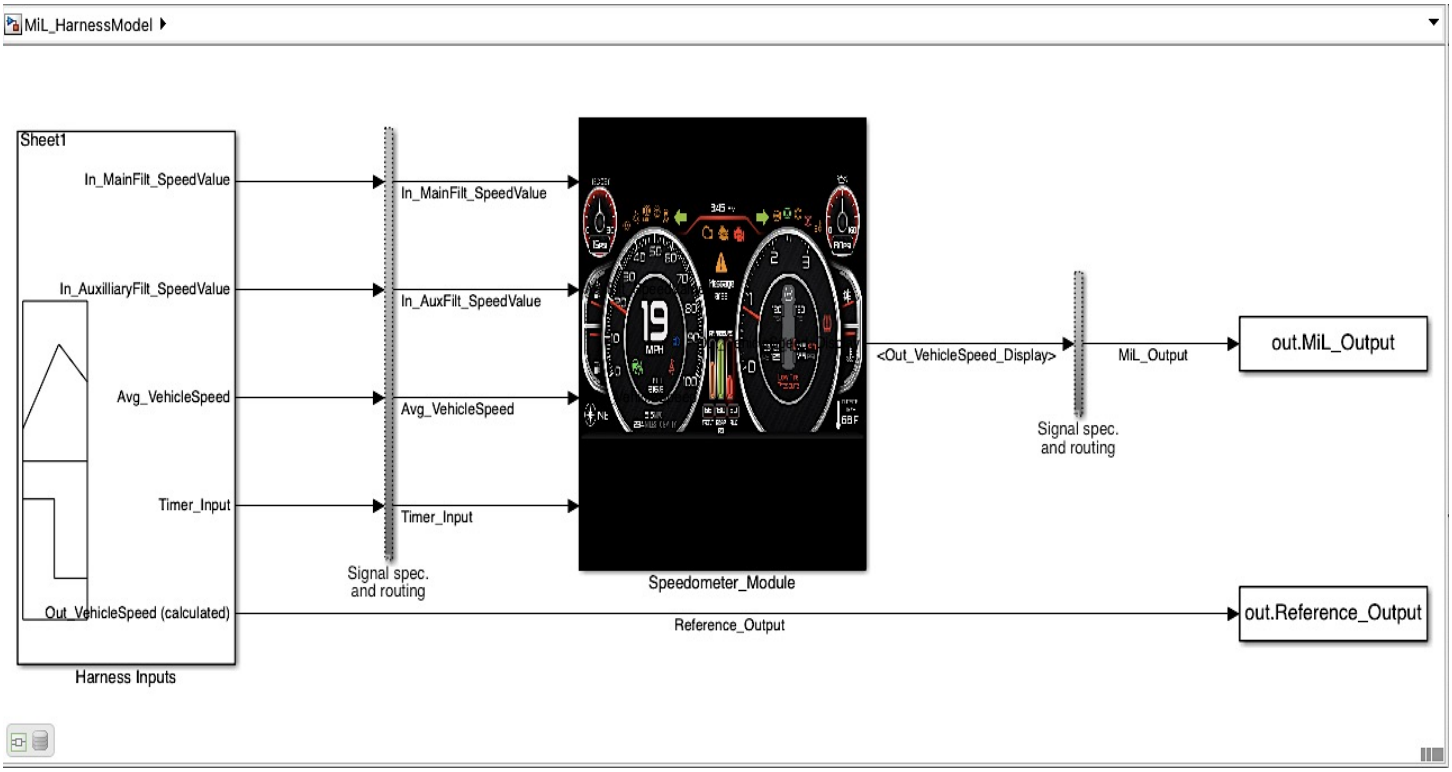
6. Test Cases



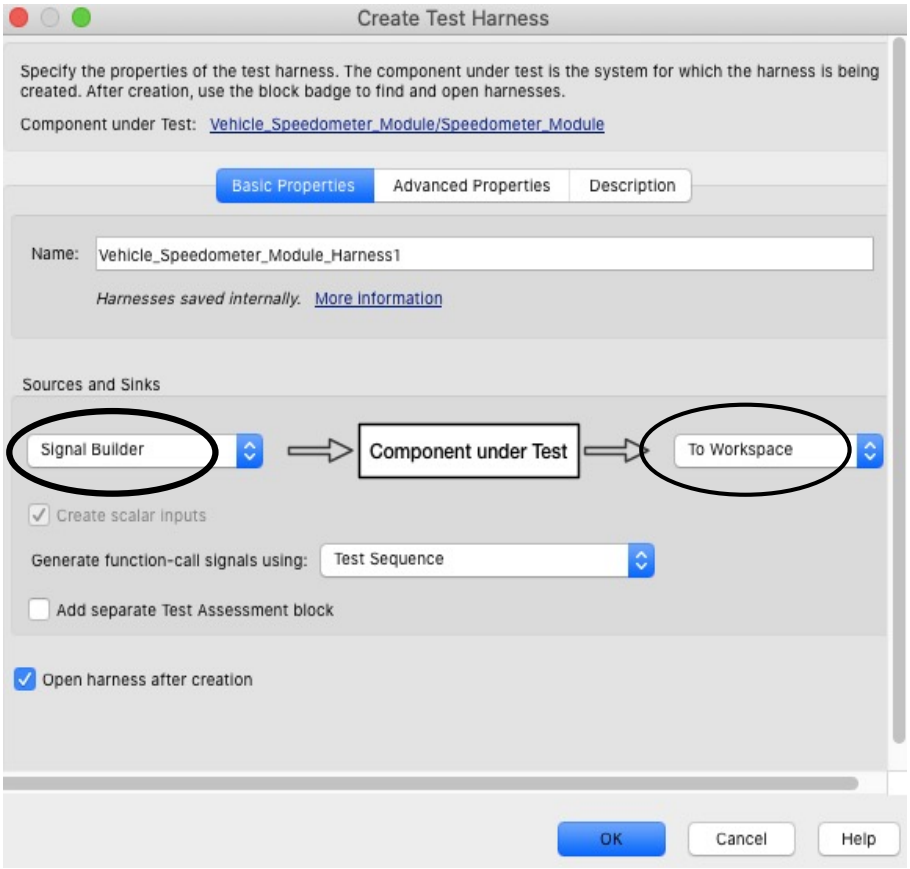
	A	B	C	D	E	F
1	Time	In_MainFilt_SpeedValue	In_AuxilliaryFilt_SpeedValu	Avg_VehicleSpeed	Timer_Input	Out_VehicleSpeed (calculated)
2	0,00	0,00	0,00	0,00	0,00	0,00
3	0,01	1,00	1,00	2,00	1,00	2,00
4	0,02	1,00	1,00	5,00	2,00	5,00
5	0,03	1,00	1,00	10,00	3,00	10,00
6	0,04	1,00	1,00	20,00	4,00	20,00
7	0,05	1,00	1,00	30,00	5,00	30,00
8	0,06	1,00	1,00	40,00	6,00	40,00
9	0,07	1,00	1,00	50,00	7,00	50,00
10	0,08	1,00	1,00	60,00	8,00	60,00
11	0,09	1,00	1,00	70,00	9,00	70,00
12	0,10	1,00	1,00	80,00	10,00	80,00
13	0,11	1,00	1,00	90,00	11,00	90,00
14	0,12	1,00	1,00	100,00	12,00	100,00
15	0,13	1,00	1,00	110,00	13,00	110,00
16	0,14	1,00	1,00	120,00	14,00	120,00
17	0,15	1,00	1,00	130,00	15,00	130,00
18	0,16	1,00	1,00	140,00	16,00	140,00
19	0,17	1,00	1,00	150,00	17,00	150,00
20	0,18	1,00	1,00	160,00	18,00	160,00
21	0,19	1,00	1,00	170,00	19,00	170,00
22	0,20	1,00	1,00	180,00	20,00	180,00
23	0,21	1,00	1,00	190,00	21,00	190,00

- The test case data has written in microsoft excel file. Its very common to wrote in excel file then imported into model with the help of signal builder. This is the standard approach used for MiL validation.
- Approximately over 100 tests are written in above screenshoted excel file.

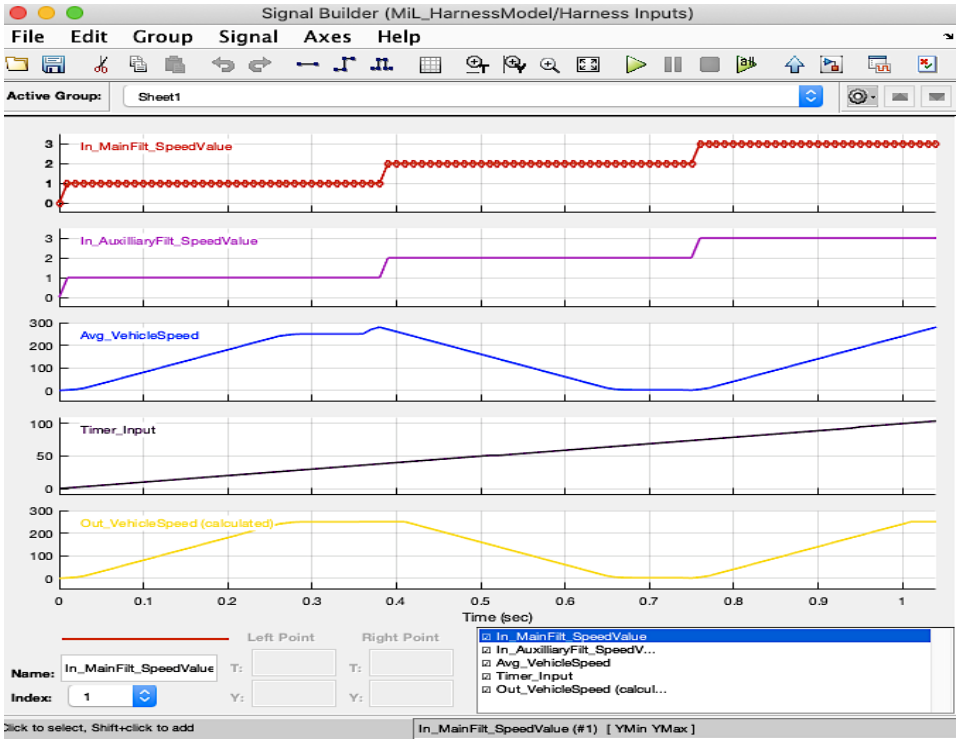
7. Model in Loop Simulation & Testing



- This simulink environment is for model in loop testing and simulation purpose.
- Model in Loop environment can be created by selecting the test harness option for respective model/ subsystem.
- The *excel file testcase* data has been imported to Signal builder block. Those are considered to be Harness Inputs.
- *To workspace* block are used for storing the output data/ results which are extracted from model (MiL data) and other is from signal builder block (reference data).

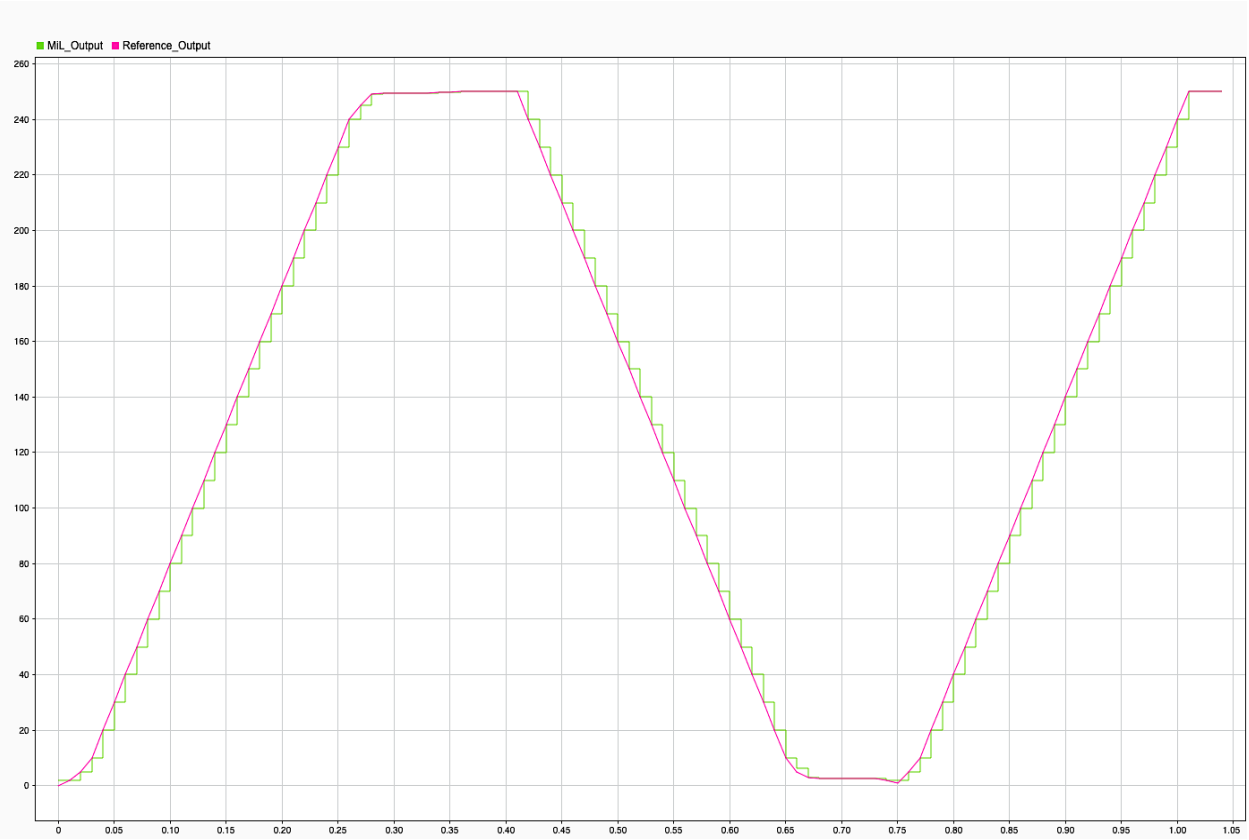


- The source block is signal builder and sink block is To workspace.



- Total 5 signals are imported including calculated/ reference data in Signal Builder block.

MiL Result:



- MiL result is used to compare and tested with the help of *Data Inspector* app in Simulink.
- The green line graph indicates the Model in Loop or one can say the model produced results, while pink color lines are for reference data which is imported through excel test case data file.
- MiL is well enough stays in with the reference data throughout the simulation. So one cay say that the developed model can be further sent for thr software in Loop validation and verification stage.

Coverage Report for MiL_HarnessModel

Table of Contents

Analysis Information

Coverage Data Information

Collected in version	(R2022a)

Model Information

Model version	1.29
Author	jamesbond
Last saved	Sun Mar 19 02:10:59 2023

Harness information

Harness model(s)	MiL_HarnessModel
Harness model owner	MiLTesting_HarnessModel

Simulation Optimization Options

Default parameter behavior	inlined
Block reduction	forced off
Conditional branch optimization	on

Coverage Options

Analyzed model	MiL_HarnessModel/Speedometer_Module
Logic block short circuiting	off

Blocks Eliminated from Coverage Analysis

#	Model Object	Rationale
	MiL_HarnessModel/Speedometer_Module/Display_Speed_Output/Abs	Simulink optimization for unsigned value

Tests

Test	Started execution	Ended execution
Run 1	19-Mar-2023 02:11:25	19-Mar-2023 02:11:26

Summary

Model Hierarchy/Complexity		Run 1	
		Decision	Execution
1. Speedometer_Module	8	100%	100%
2. . . . Auxilliary_Data_Filtering	2	100%	100%
3. . . . Compare To Zero		NA	100%
4. . . . Display_Speed_Output	1	100%	100%
5. . . . Input_Processing	2	100%	100%
6. . . . Main_Data_Filtering	2	100%	100%

Details

1. SubSystem block "Speedometer_Module"

Child Systems: Auxilliary_Data_Filtering, Compare To Zero, Display_Speed_Output, Input_Processing, Main_Data_Filtering

Metric	Coverage (this object)	Coverage (inc. descendants)
Cyclomatic Complexity	1	8
Decision	NA	100% (10/10) decision outcomes
Execution	NA	100% (16/16) objective outcomes

Full Coverage

Model Object	Metric
Math block "Rem"	Execution
Constant block "Constant"	Execution

2. SubSystem block "Auxilliary_Data_Filtering"

Justify or Exclude

Parent: MiL_HarnessModel/Speedometer_Module

Metric	Coverage (this object)	Coverage (inc. descendants)
Cyclomatic Complexity	2	2

Decision	100% (2/2) decision outcomes	100% (2/2) decision outcomes
Execution	NA	100% (4/4) objective outcomes

Decisions analyzed

Enable control activated	100%
false	94/105
true	11/105

Full Coverage

Model Object	Metric
Gain block "Gain"	Execution
Product block "Divide"	Execution
Sum block "Add"	Execution
Constant block "Constant"	Execution

3. SubSystem block "Compare To Zero"

Justify or Exclude

Parent: MiL_HarnessModel/Speedometer_Module

Metric	Coverage (this object)	Coverage (inc. descendants)
Cyclomatic Complexity	0	0
Execution	NA	100% (1/1) objective outcomes

Full Coverage

Model Object	Metric
RelationalOperator block "Compare"	Execution

4. SubSystem block "Display_Speed_Output"

Justify or Exclude

Parent: MiL_HarnessModel/Speedometer_Module

Metric	Coverage (this object)	Coverage (inc. descendants)
Cyclomatic Complexity	0	1
Decision	NA	100% (2/2) decision outcomes
Execution	NA	100% (4/4) objective outcomes

Full Coverage

Model Object	Metric
Switch block "Switch"	Decision, Execution
Sum block "Add"	Execution
RelationalOperator block "Relational Operator"	Execution
Constant block "Constant"	Execution

5. SubSystem block "Input_Processing"

Justify or Exclude

Parent: MiL_HarnessModel/Speedometer_Module

Metric	Coverage (this object)	Coverage (inc. descendants)
Cyclomatic Complexity	0	2
Decision	NA	100% (4/4) decision outcomes
Execution	NA	100% (1/1) objective outcomes

Full Coverage

Model Object	Metric
Saturate block "Saturation"	Decision, Execution

6. SubSystem block "Main_Data_Filtering"

Justify or Exclude

Parent: MiL_HarnessModel/Speedometer_Module

Metric	Coverage (this object)	Coverage (inc. descendants)
Cyclomatic Complexity	2	2
Decision	100% (2/2) decision outcomes	100% (2/2) decision outcomes
Execution	NA	100% (4/4) objective outcomes

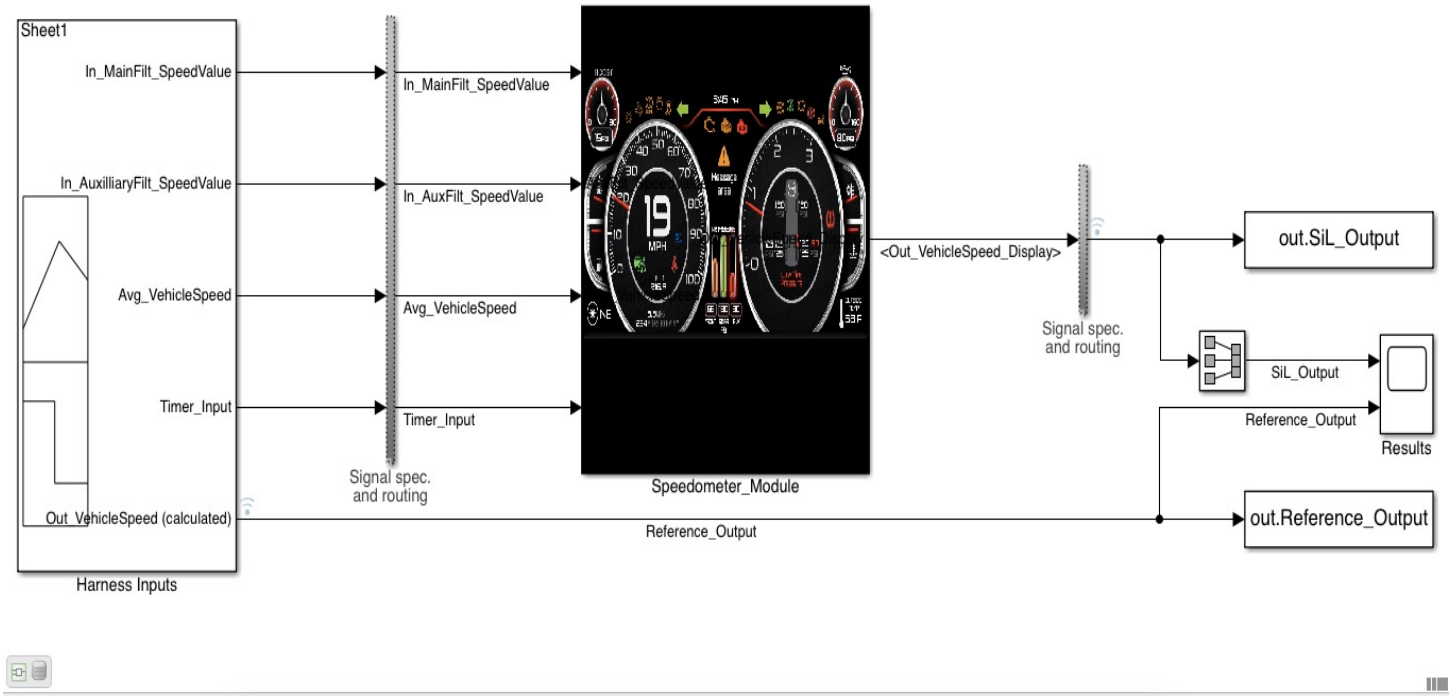
Decisions analyzed

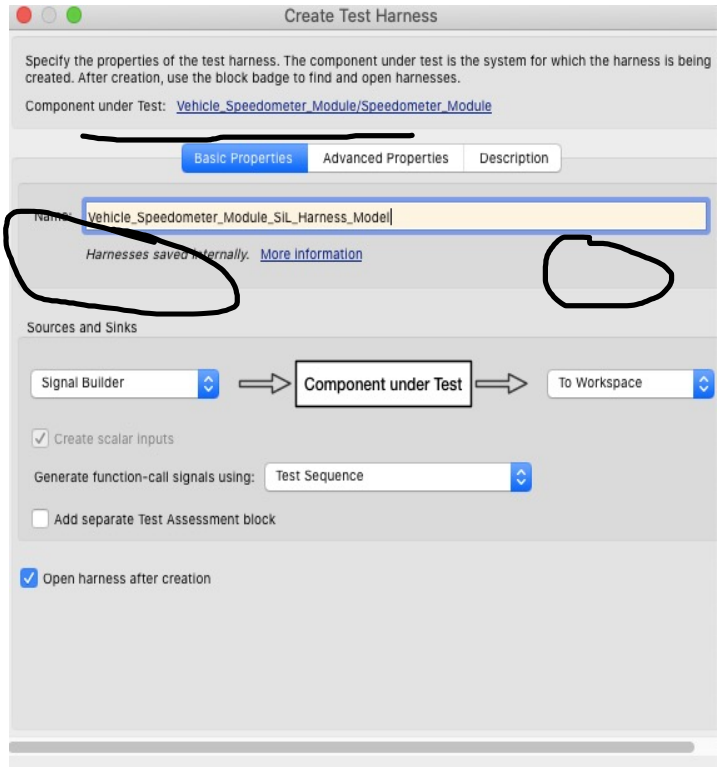
Enable control activated	100%
false	94/105
true	11/105

Full Coverage

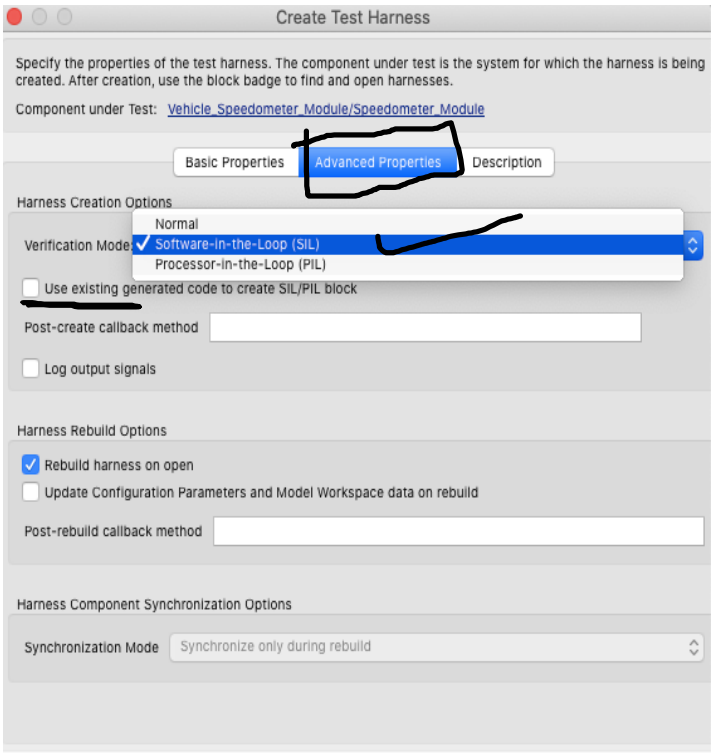
Model Object	Metric
Gain block "Gain"	Execution
Product block "Divide"	Execution
Sum block "Add"	Execution
Constant block "Constant"	Execution

8. Software in Loop Simulation & Testing



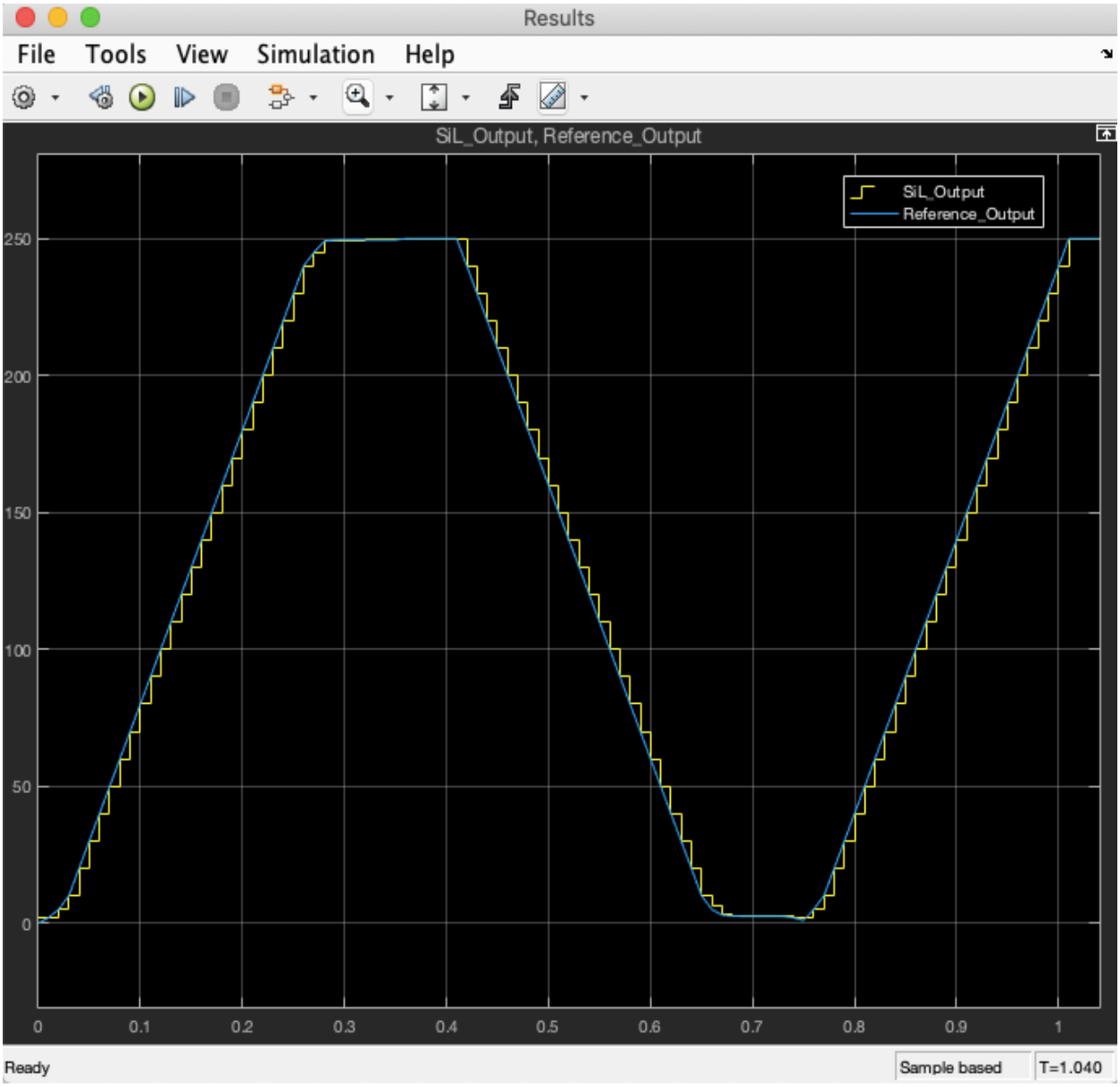


- Open the Create Test Harness pop up window by right click on main model subysystem. Then Select the Source and Sink blocks you want to kept as in harness model. Generally the most common is Signal Builder (Source block) and To Workspace (Sink block)



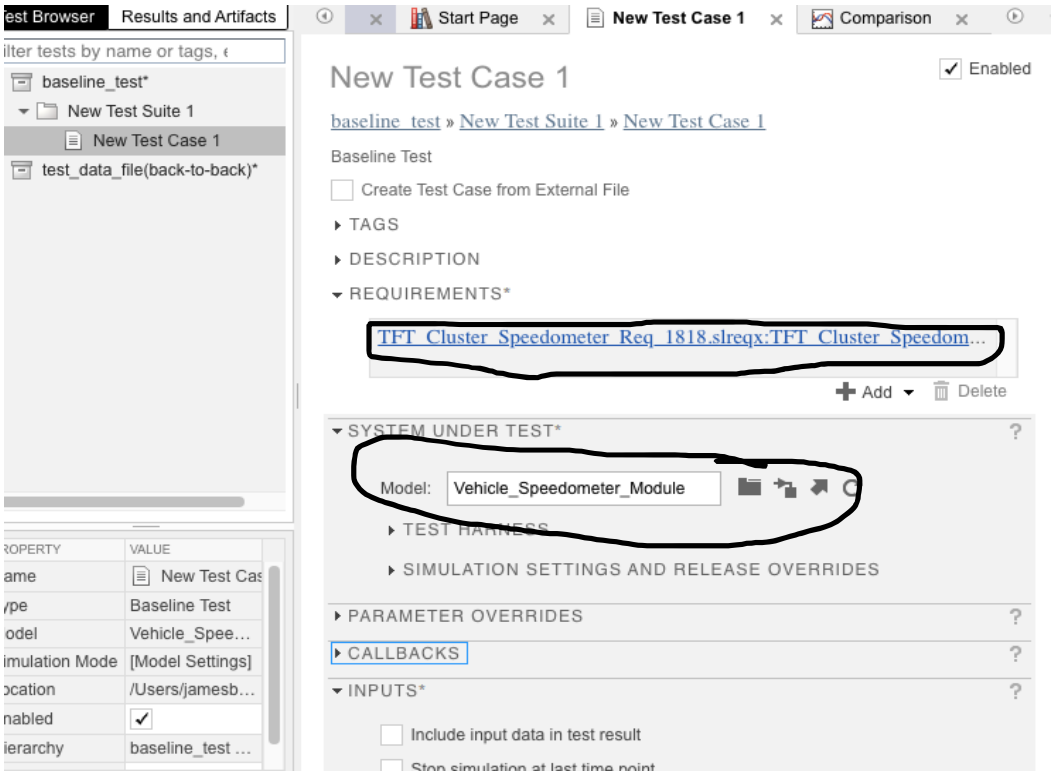
- Go to Advanced Properties to change the verififcation mode from Normal (which is previously Model in Loop) to Software-in-Loop (SiL).
- It can also be done by selecting the main model subsystem and Generate S-fucntion to create the SiL harness model for Code verification.

SiL Result:

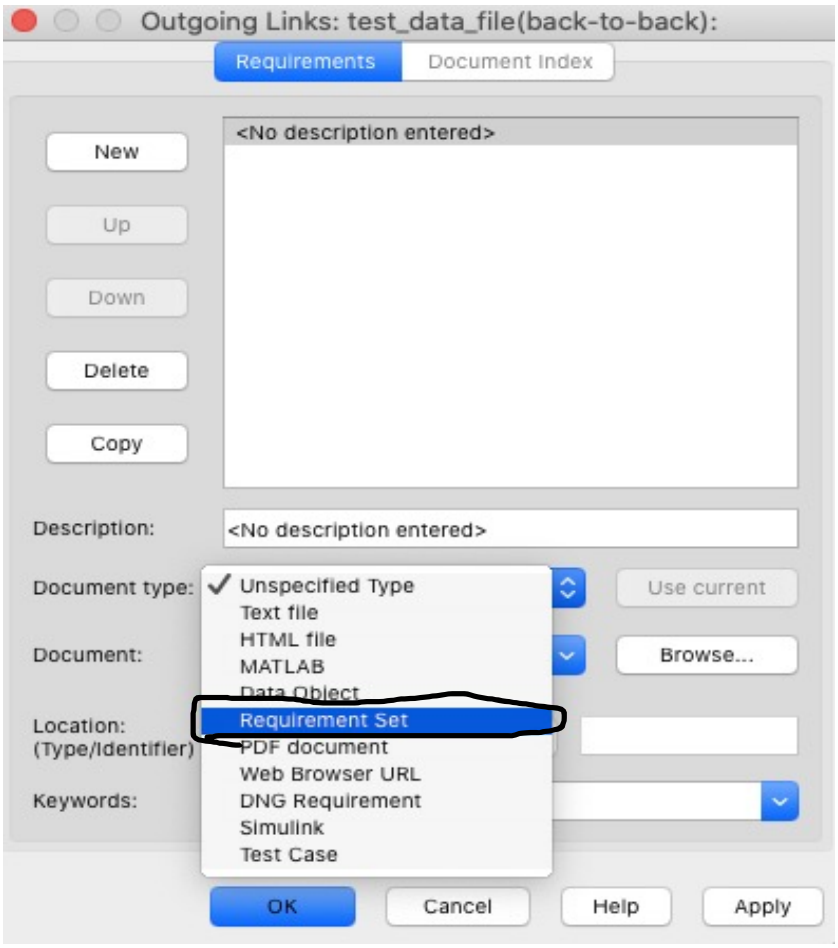


- As it can be observed that, SiL output and reference output do corrlate very well.
- Which means, the generated code for the specific model is sustainable and perform good enough for MiL and SiL testing stages. So, the code can be send forward for further Hardware-in-Loop testing in auto V cycle.

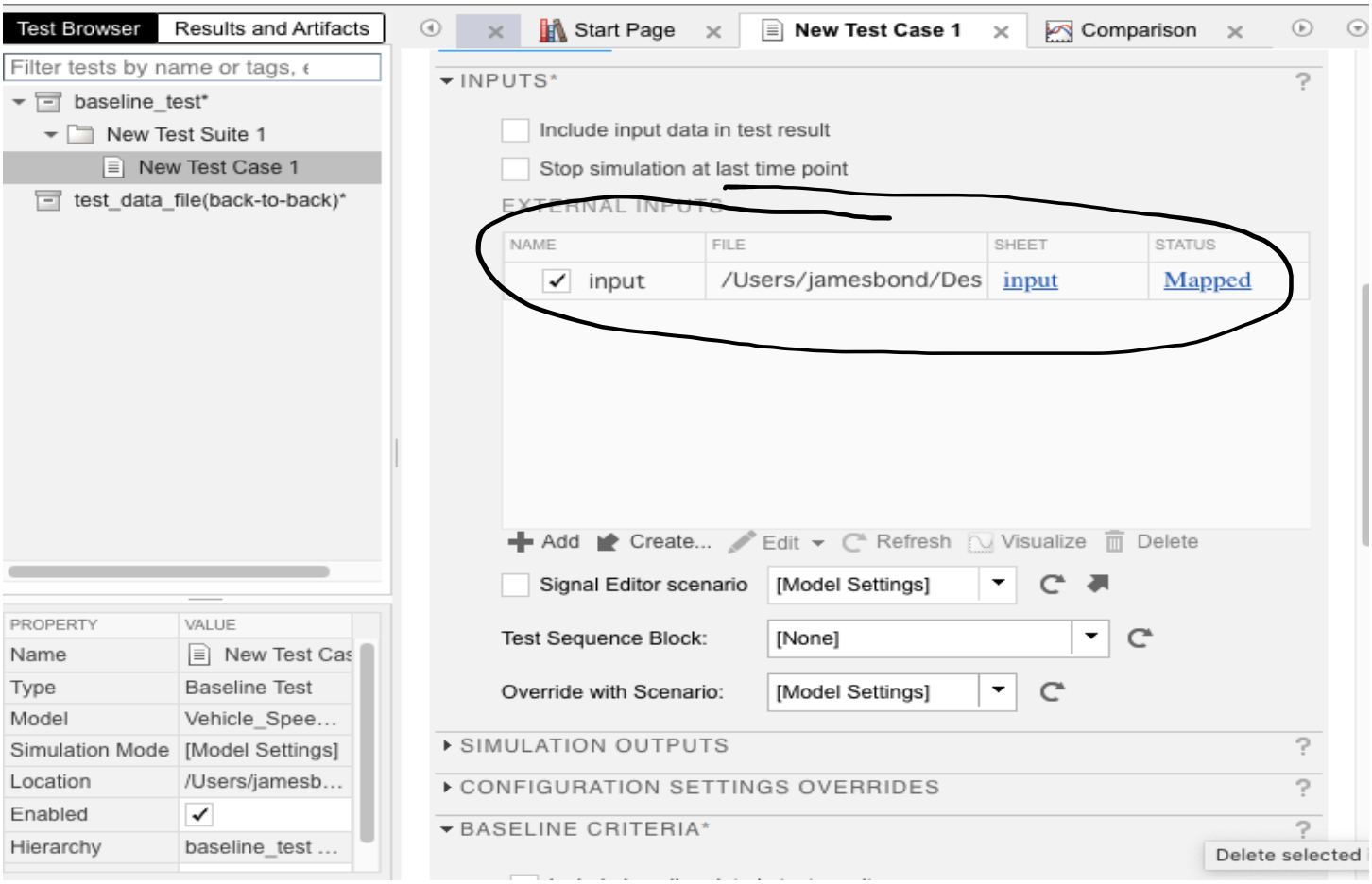
9. Baseline Test
(Simulink Test Manager)

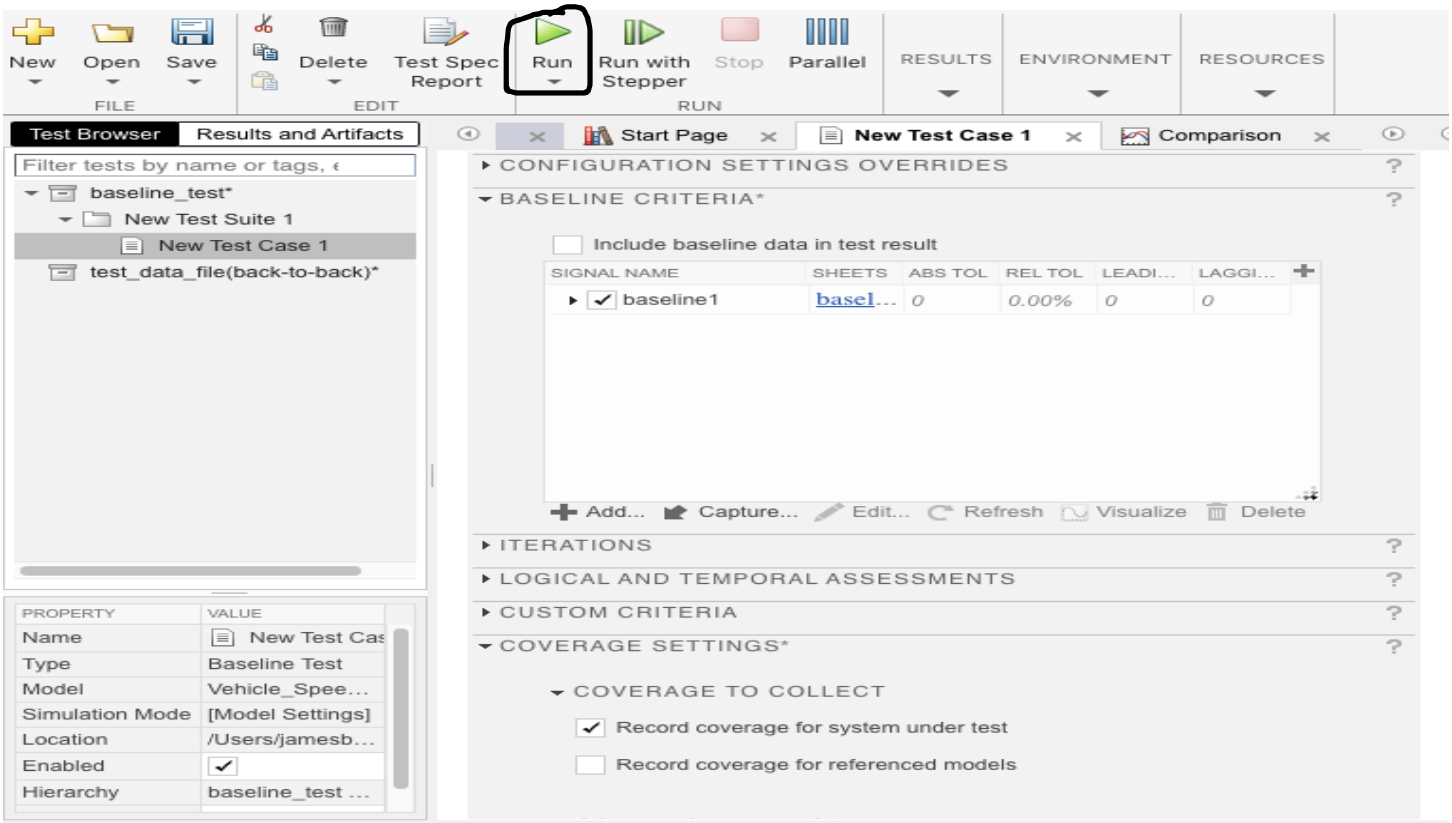


- Open the Simulink Test Manager and Cretae a new blank file to initiate the Baseline Test procedure.
- Add the Requirement *.slreqx* extension file in the Reuirement tab.
- The requirement document type should be in that case is *Reuirement Set.*
- Next step is to create input data file by creating the new excel file. The test case data should be add to this newly created excel file.
- Then createmapped the data.
- After that, cretae Baseline for input.
- After creating the baseline test setup, run baseline test to compare the results, tolerances.

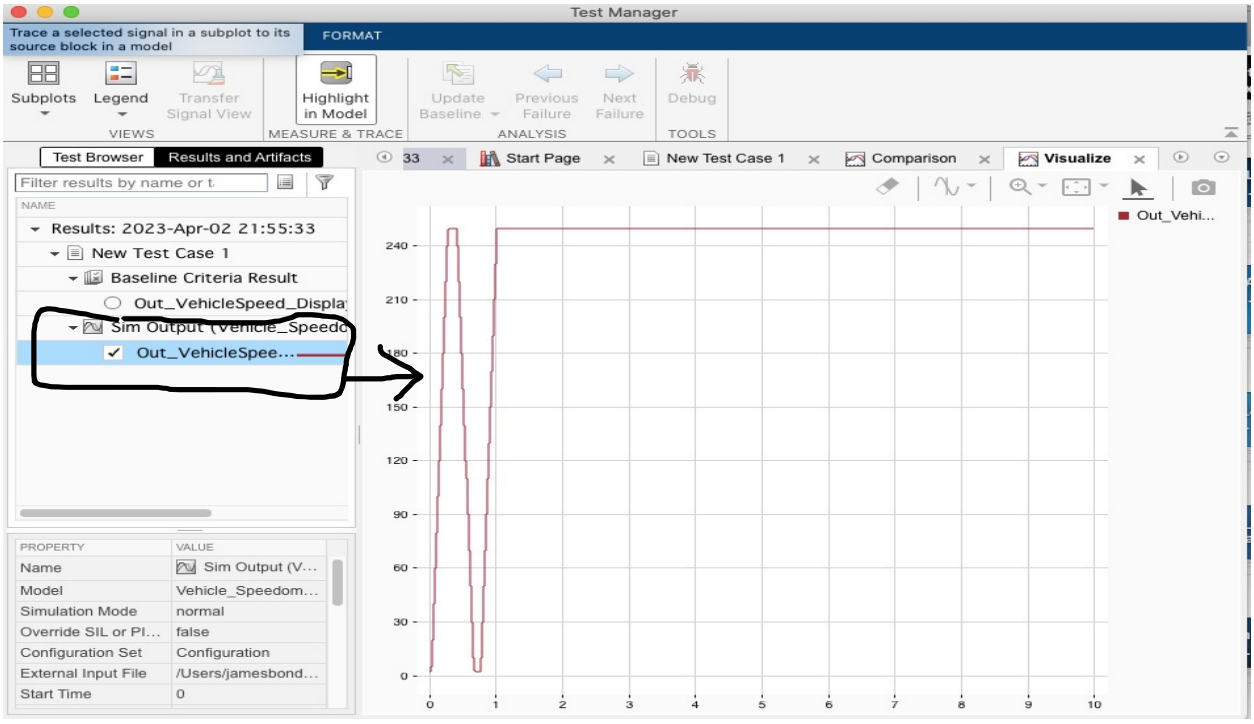


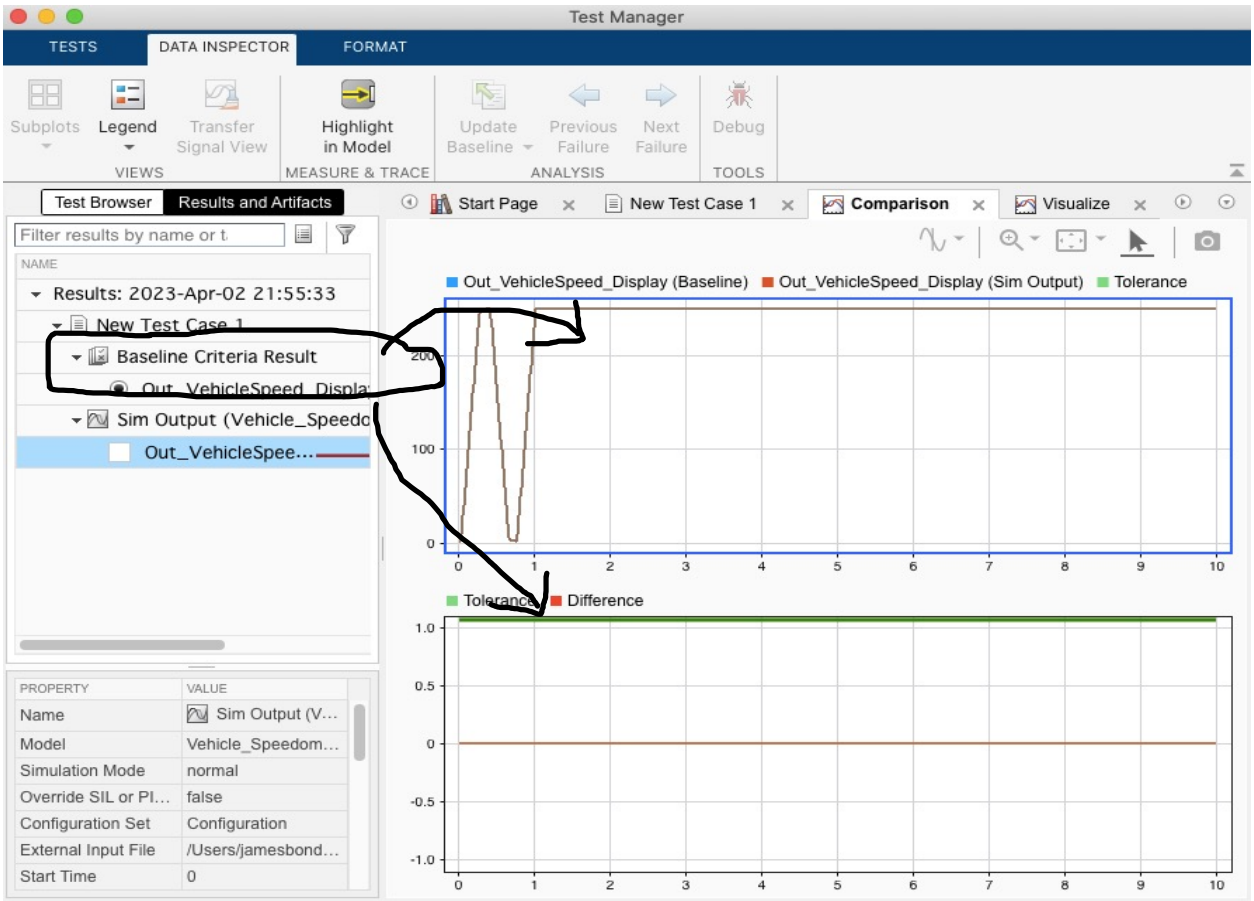
	A	B	C	D	E	F
	time	In_MainFilt_SpeedValue	In_AuxFilt_SpeedValue	Avg_VehicleSpeed	Timer_Input	
		Type: uint8	Type: uint8	Fixdt: numerictype(0,16,0.01,0)	Type: uint16	
		Source: Input				
2	0	0	0	0	0	
3	0,01	1,00	1,00	2,00	1,00	
4	0,02	1,00	1,00	5,00	2,00	
5	0,03	1,00	1,00	10,00	3,00	
6	0,04	1,00	1,00	20,00	4,00	
7	0,05	1,00	1,00	30,00	5,00	
8	0,06	1,00	1,00	40,00	6,00	
9	0,07	1,00	1,00	50,00	7,00	
10	0,08	1,00	1,00	60,00	8,00	
11	0,09	1,00	1,00	70,00	9,00	
12	0,10	1,00	1,00	80,00	10,00	
13	0,11	1,00	1,00	90,00	11,00	
14	0,12	1,00	1,00	100,00	12,00	


Test Browser
Results and Artifacts
Filter tests by name or tags, e
baseline_test*
New Test Suite 1
New Test Case 1
test_data_file(back-to-back)*
Start Page
New Test Case 1
Comparison
INPUTS*
Include input data in test result
Stop simulation at last time point
EXTERNAL INPUTS
NAME
FILE
SHEET
STATUS
input
/Users/jamesbond/Des
input
Mapped
Add
Create...
Edit
Refresh
Visualize
Delete
Signal Editor scenario
[Model Settings]
Test Sequence Block:
[None]
Override with Scenario:
[Model Settings]
SIMULATION OUTPUTS
CONFIGURATION SETTINGS OVERRIDES
BASELINE CRITERIA*
Delete selected
PROPERTY
VALUE
Name
New Test Cas
Type
Baseline Test
Model
Vehicle_Spee...
Simulation Mode
[Model Settings]
Location
/Users/jamesb...
Enabled
Hierarchy
baseline_test ...


New
Open
Save
FILE
Delete
Test Spec Report
EDIT
Run
Run with Stepper
Stop
Parallel
RUN
RESULTS
ENVIRONMENT
RESOURCES
Test Browser
Results and Artifacts
Filter tests by name or tags, e
baseline_test*
New Test Suite 1
New Test Case 1
test_data_file(back-to-back)*
Start Page
New Test Case 1
Comparison
CONFIGURATION SETTINGS OVERRIDES
BASELINE CRITERIA*
Include baseline data in test result
SIGNAL NAME
SHEETS
ABS TOL
REL TOL
LEADI...
LAGGI...
baseline1
basel...
0
0.00%
0
0
Add...
Capture...
Edit...
Refresh
Visualize
Delete
ITERATIONS
LOGICAL AND TEMPORAL ASSESSMENTS
CUSTOM CRITERIA
COVERAGE SETTINGS*
COVERAGE TO COLLECT
Record coverage for system under test
Record coverage for referenced models
PROPERTY
VALUE
Name
New Test Cas
Type
Baseline Test
Model
Vehicle_Spee...
Simulation Mode
[Model Settings]
Location
/Users/jamesb...
Enabled
Hierarchy
baseline_test ...

Baseline Test Results:





Report Generated by Test Manager

Title: **Baseline_Test_report**

Author:

Date: **02-Apr-2023 22:11:23**

Test Environment

Platform: MACI64
MATLAB: (R2022a)

Summary

Name	Outcome	Duration (Seconds)
Results: 2023-Apr-02 21:55:33	✓	31.941
New Test Case 1	✓	31.898

Results: 2023-Apr-02 21:55:33

Result Type:	Result Set
Parent:	None
Start Time:	02-Apr-2023 21:55:38
End Time:	02-Apr-2023 21:56:10
Outcome:	Total: 1, Passed: 1

Aggregated Coverage Results

Analyzed Model	Sim Mode	Complexity	Decision	Execution
Vehicle_Speedometer_Module	Normal	10	100%	100%

Back to Report Summary

New Test Case 1

Test Result Information

Result Type:	Test Case Result
Parent:	Results: 2023-Apr-02 21:55:33
Start Time:	02-Apr-2023 21:55:38
End Time:	02-Apr-2023 21:56:10
Outcome:	Passed

Test Case Information

Name:	New Test Case 1
Type:	Baseline Test
Baseline Name:	baseline1
Baseline File:	/Users/jamesbond/Desktop.xlsx

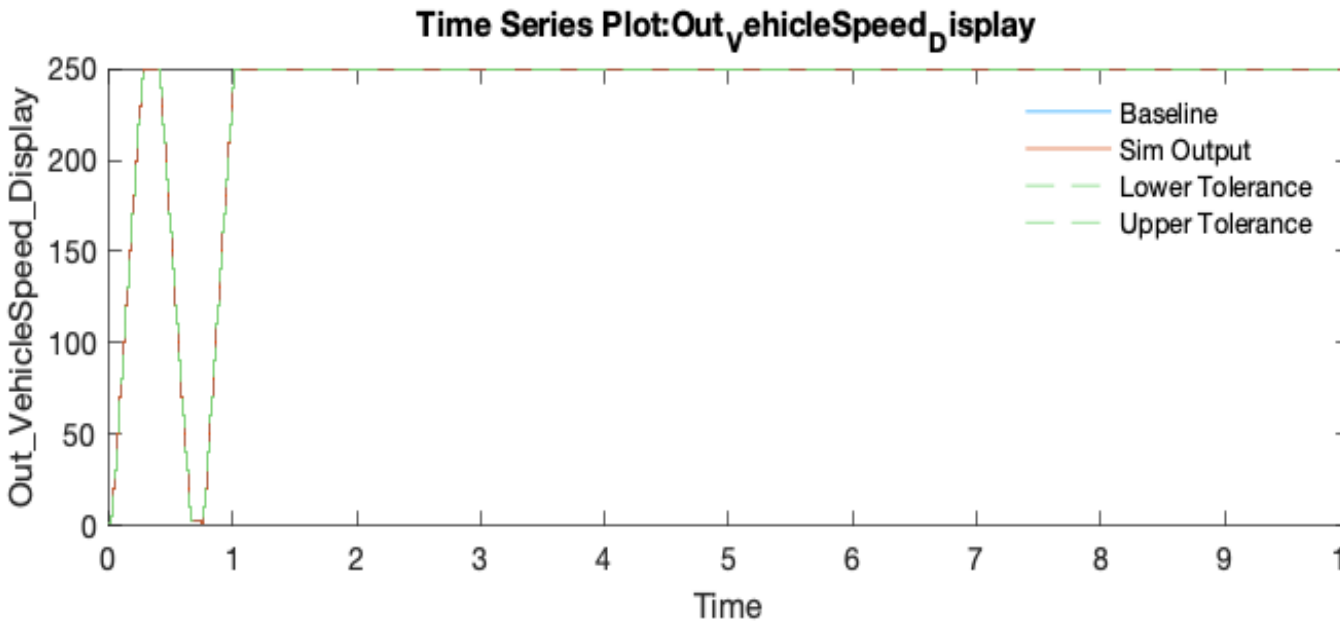
Test Case Requirements

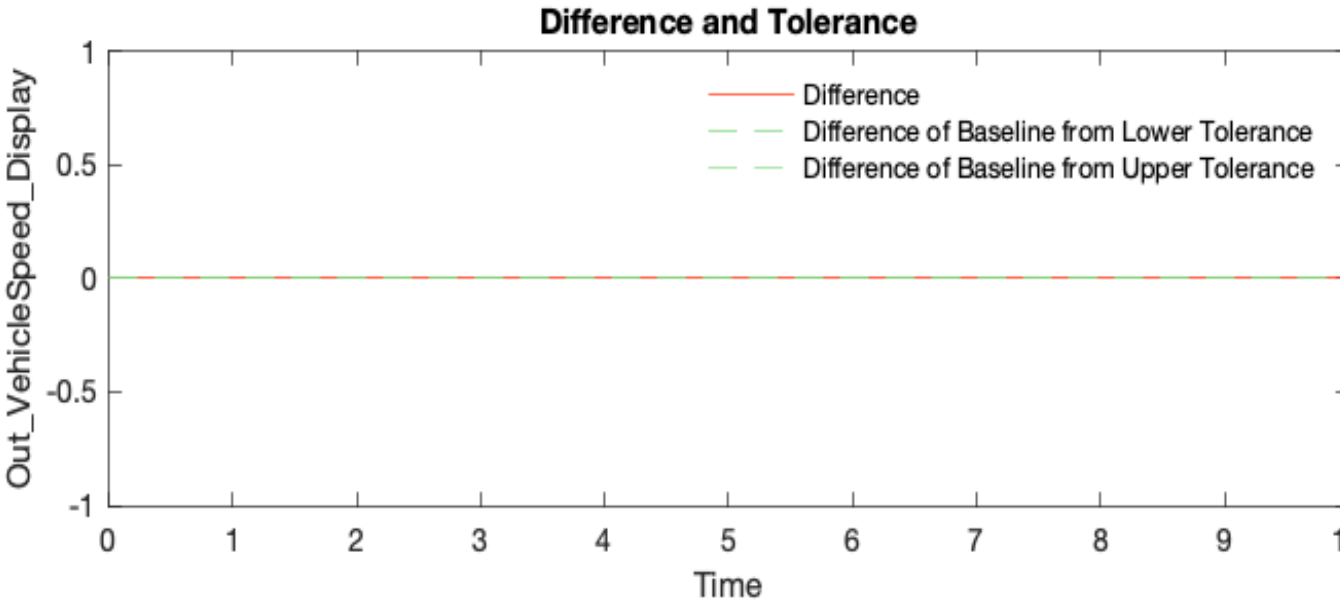
Description:	TFT_Cluster_Speedometer_Req_1818.slreqx:TFT_Cluster_Speedometer_Req_1818
Document:	../../../Desktop/TFT_Cluster_Speedometer_Req_1818.slreqx

Baseline Comparison

Name	Abs Tol	Rel Tol	Lead Tol	Lag Tol	Max Diff	Data Type 1	Units 1	Sample Time 1	Data Type 2	Units 2	Sample Time 2	Interp	Sync	Link to Plot
Out_VehicleSpeed_Display	0	0	0	0	0	fxp_slope_bias_scaling			ufix16_Sp01		0.01	zoh	union	Link

Name	Abs Tol	Rel Tol	Lead Tol	Lag Tol	Max Diff	Data Type 1	Units 1	Sample Time 1	Data Type 2	Units 2	Sample Time 2	Interp	Sync
Out_VehicleSpeed_Display	0	0	0	0	0	fxp_slope_bias_scaling			ufix16_Sp01		0.01	zoh	union





[Back to Report Summary](#) [Back to Criteria Results](#)

Simulation

System Under Test Information

Model:	Vehicle_Speedometer_Module
Release:	Current

Simulation Mode:	normal
Override SIL or PIL Mode:	0
Configuration Set:	Configuration
External Input Name:	input
External Input File:	/Users/jamesbond/Desktop/nv.xls
Start Time:	0
Stop Time:	10
Checksum:	1959822732 1146748841 1462711833 2177022575
Simulink Version:	10.5
Model Version:	1.43
Model Author:	jamesbond
Date:	Sun Apr 02 13:49:32 2023
User ID:	jamesbond
Model Path:	/Users/jamesbond/Desktop/TFT Speedometer Module/Vehicle_Speedometer_Module.slx
Solver Name:	FixedStepDiscrete
Solver Type:	Fixed-Step
Fixed Step Size:	0.01
Simulation Start Time:	2023-04-02 21:55:39
Simulation Stop Time:	2023-04-02 21:55:53
Platform:	MACI64

Simulation Logs:

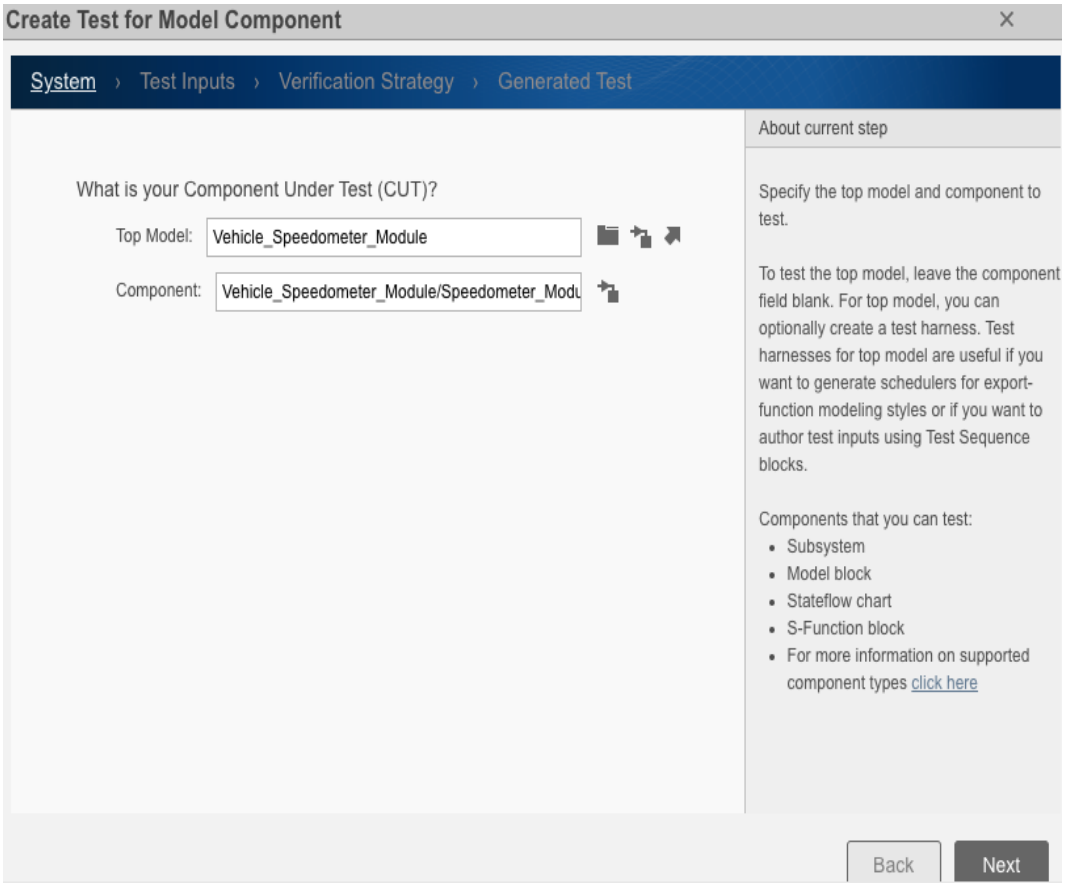
Inconsistent numeric values for port 1 of 'Vehicle_Speedometer_Module/In_AuxFilt_SpeedValue': Output value (uint8(0)) at major time step 0 is less than minimum (uint8(1)) from 'Vehicle_Speedometer_Module/In_AuxFilt_SpeedValue'

Inconsistent numeric values for port 1 of 'Vehicle_Speedometer_Module/Avg_VehicleSpeed': Output value (fi([],0,16,0.01,0,'int',uint16(29000))) at major time step 1.05 is greater than maximum (fi([],0,16,0.01,0,'int',uint16(28000))) from 'Vehicle_Speedometer_Module/Avg_VehicleSpeed'

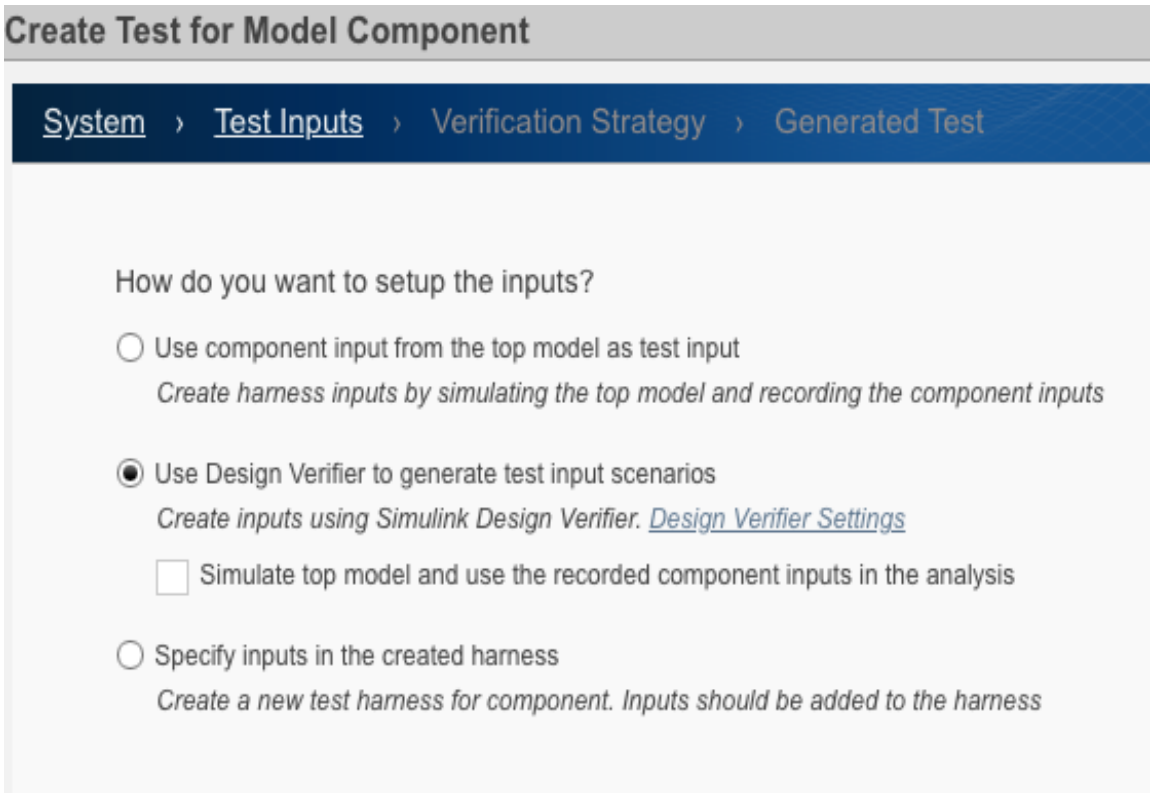
Back to Report Summary

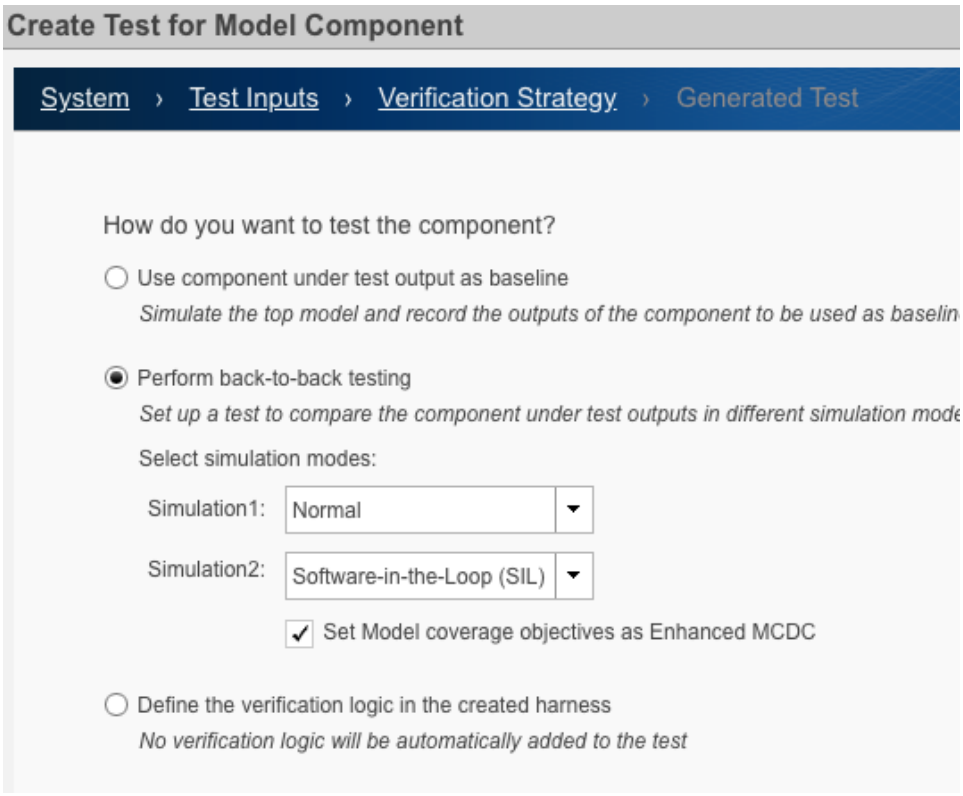
Back to Back Test
(Simulink Test Manager)

- Simulink Test Manager Toolbox > New > Auto Create > Test for Model Component.
- Select the top/ main subsystem for model selection of TFT Speedometer Module.

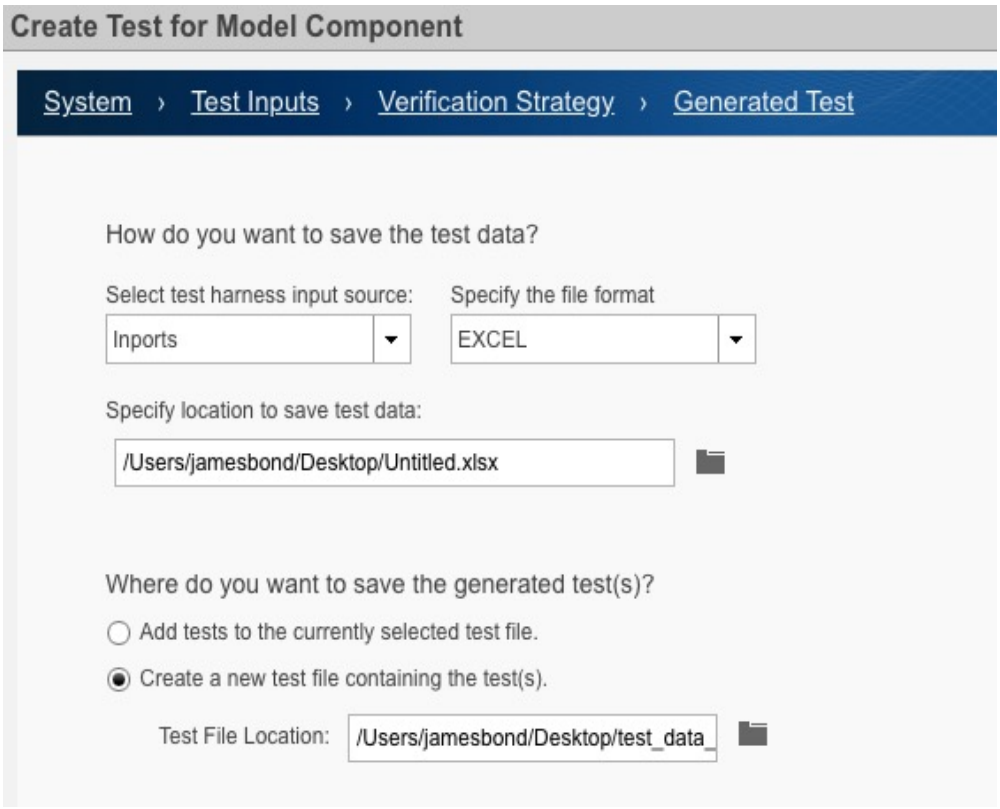


- Second step is *Test Inputs*, where usually the standard approach is to select the *Use Design Verifier to generate test input scenarios.*

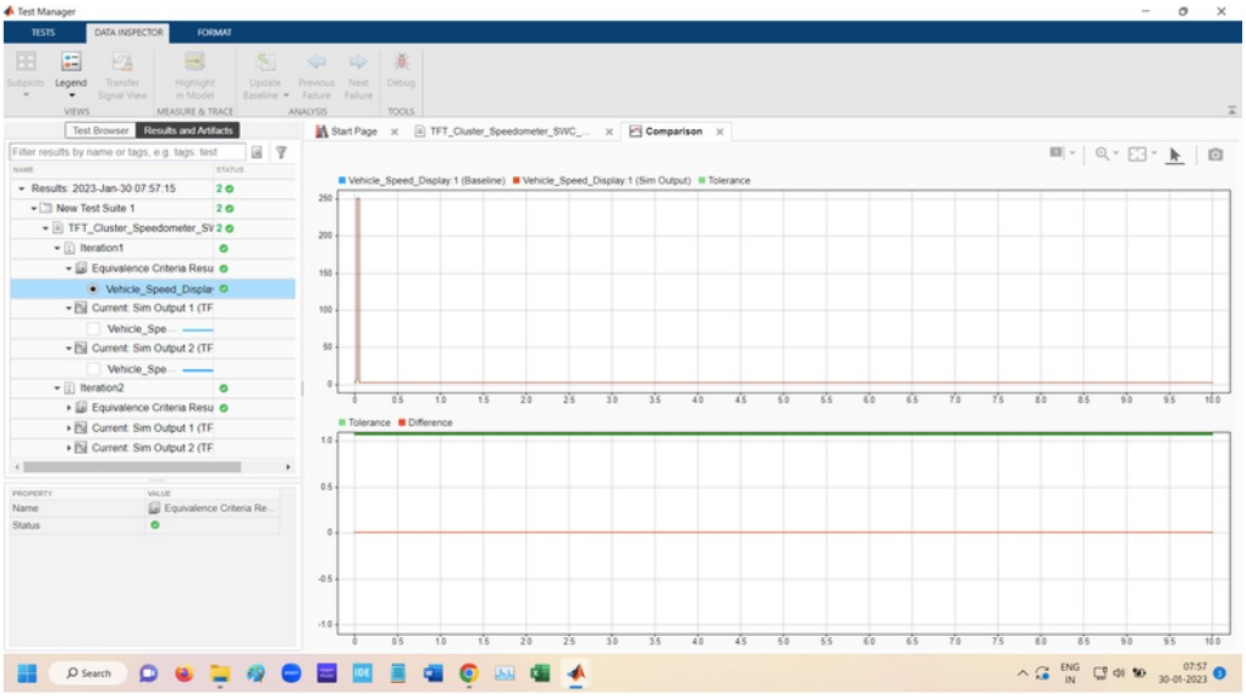


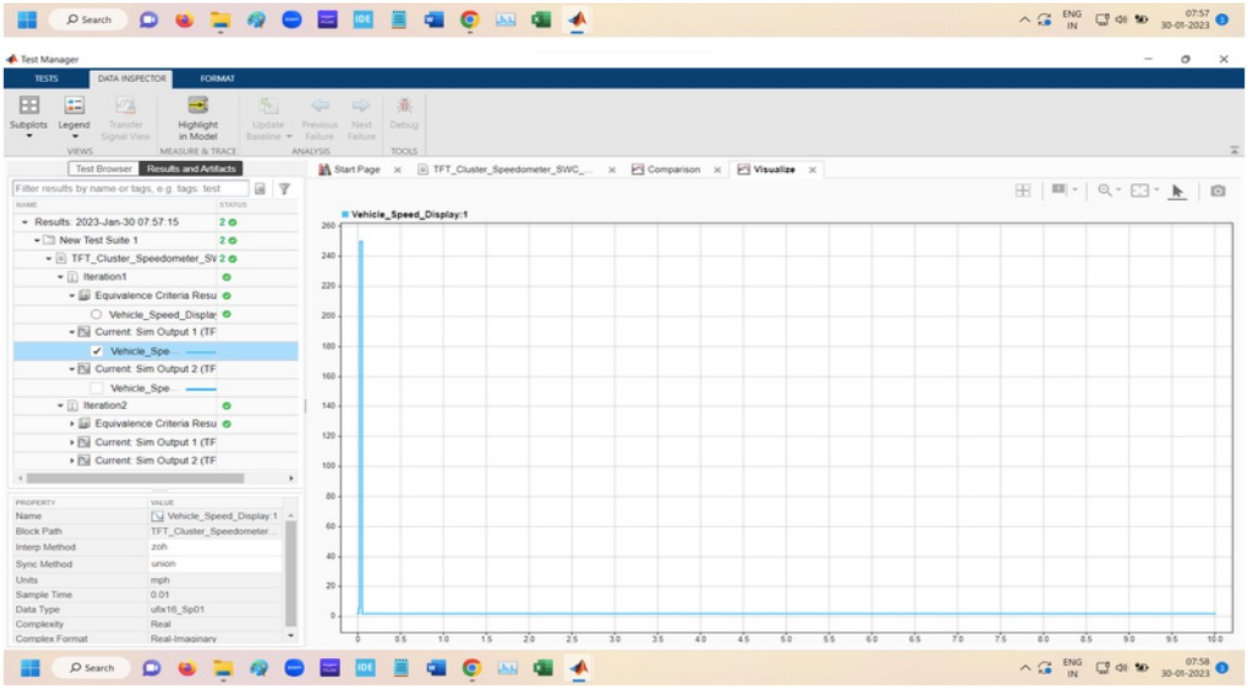


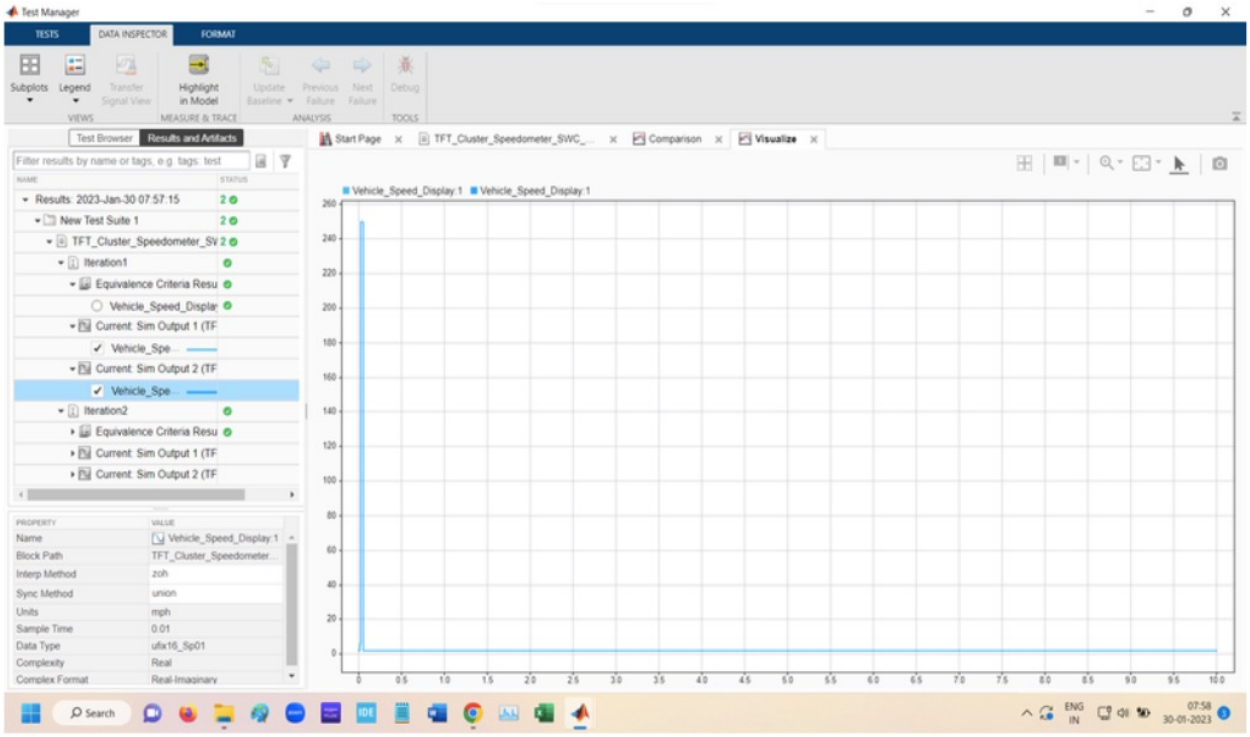
- Third Step is *Verification Strategy*, as I wanted to perform therefore I selected option Perform back-to-back testing. Simulation1 is Normal means Model-in-Loop simulation. Simulation2 is SiL.



- For generated test again the default options are good to go with, Excel file for stroing the tests, test harness input source is Inports. Select the location and name of generated *.mldatx* extension file.
- After completing these 4-steps, the same procedure like Baseline test will be applied to Back-to-back testing.




Test Manager
TESTS
DATA INSPECTOR
FORMAT
Subplots
Legend
Transfer Signal View
Highlight in Model
Update Baseline
Previous Failure
Next Failure
Debug
VIEWS
MEASURE & TRACE
ANALYSIS
TOOLS
Test Browser
Results and Artifacts
Filter results by name or tags, e.g. tags: test
NAME
STATUS
Results: 2023-Jan-30 07:57:15
New Test Suite 1
TFT_Cluster_Speedometer_SV
Iteration1
Equivalence Criteria Resu
Vehicle_Speed_Display
Current: Sim Output 1 (TF
Vehicle_Spe...
Current: Sim Output 2 (TF
Iteration2
PROPERTY
VALUE
Name
Vehicle_Speed_Display:1
Block Path
TFT_Cluster_Speedometer...
Interp Method
zoh
Sync Method
union
Units
mph
Sample Time
0.01
Data Type
ufix16_Sp01
Complexity
Real
Complex Format
Real-Imaginary
Start Page
TFT_Cluster_Speedometer_SWC_...
Comparison
Visualize
Vehicle_Speed_Display:1

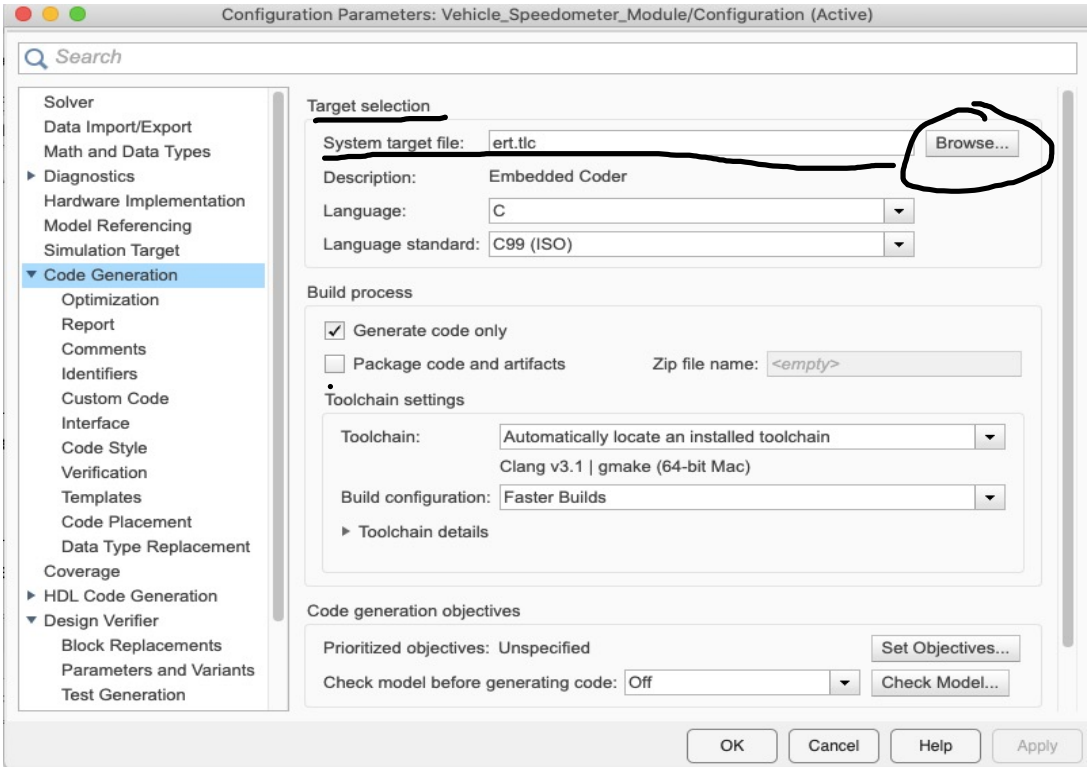

Test Manager
TESTS
DATA INSPECTOR
FORMAT
Subplots
Legend
Transfer Signal View
Highlight in Model
Update Baseline
Previous Failure
Next Failure
Debug
VIEWS
MEASURE & TRACE
ANALYSIS
TOOLS
Test Browser
Results and Artifacts
Filter results by name or tags, e.g. tags: test
NAME
STATUS
Results: 2023-Jan-30 07:57:15
New Test Suite 1
TFT_Cluster_Speedometer_SV
Iteration1
Equivalence Criteria Resu
Vehicle_Speed_Display
Current: Sim Output 1 (TF
Vehicle_Spe...
Current: Sim Output 2 (TF
Vehicle_Spe...
Iteration2
PROPERTY
VALUE
Name
Vehicle_Speed_Display:1
Block Path
TFT_Cluster_Speedometer...
Interp Method
zoh
Sync Method
union
Units
mph
Sample Time
0.01
Data Type
ufix16_Sp01
Complexity
Real
Complex Format
Real-Imaginary
Start Page
TFT_Cluster_Speedometer_SWC_...
Comparison
Visualize
Vehicle_Speed_Display:1
Vehicle_Speed_Display:1

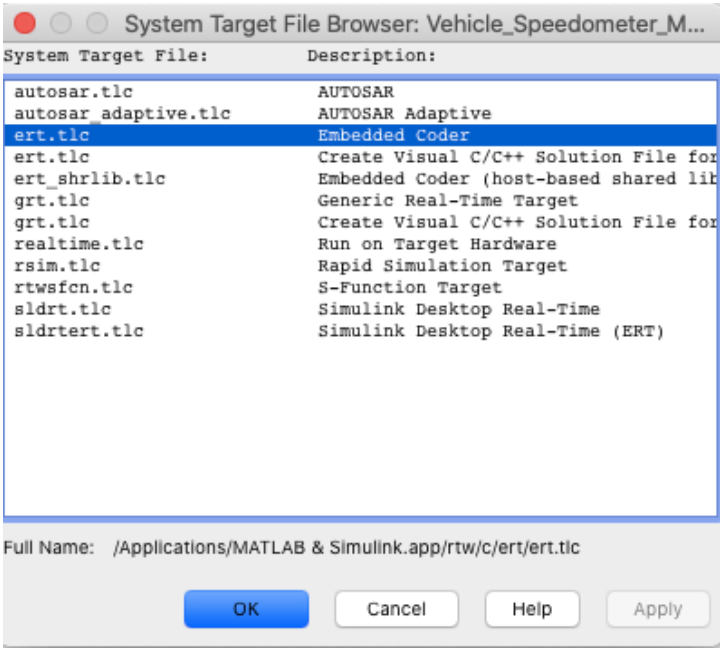
10. Code Generation

- This Code Generation Report is a C-language codings for given model with header and c-files.
- The code generation report includes number of different files.
- This code is generated based on Embedded Coder
- Here I only show main file (ert.tlc) which is simulink target file, model files (.c and .h files) so in total 3 files.
- ert.tlc (simulink target file/ main file)

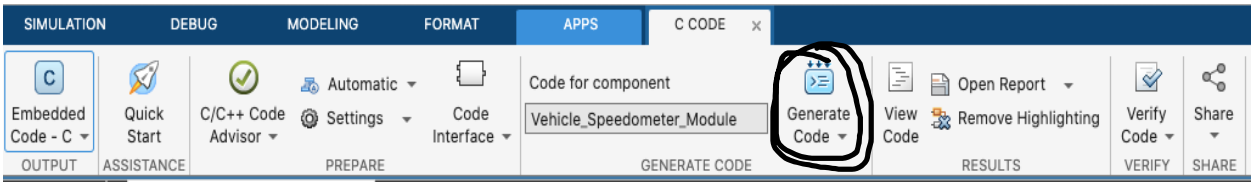
  ACC_System.c (c file/ model file)

  ACC_System.h (header file/ model file)





- open Model configuration settings>Code Generation> Target Selection tab click on the Browse button to access the above pop up window in order to select the System Target File.



Code Generation Report for 'Vehicle_Speedometer_Module'

Model Information

Author	jamesbond
Last Modified By	jamesbond
Model Version	1.35
Tasking Mode	SingleTasking

Configuration settings at time of code generation

Code Information

System Target File	ert.tlc
Hardware Device Type	Intel->x86-64 (Mac OS X)
Simulink Coder Version	9.7 (R2022a) 13-Nov-2021
Timestamp of Generated Source Code	Sun Mar 19 04:03:56 2023
Location of Generated Source Code	/Users/jamesbond/Desktop/Vehicle_Speedometer_Module_ert_rtw
Type of Build	Model
Memory Information	Global Memory: 18(bytes) Maximum Stack: 16(bytes)
Objectives Specified	**Unspecified**

Additional Information

Code Generation Advisor	Not run

Vehicle_Speedometer_Module.h

```
/*
 * Academic License - for use in teaching, academic research, and
meeting
 * course requirements at degree granting institutions only.  Not
for
 * government, commercial, or other organizational use.
 *
 * File: Vehicle_Speedometer_Module.h
 *
 * Code generated for Simulink model 'Vehicle_Speedometer_Module'.
 *
 * Model version                  : 1.35
 * Simulink Coder version         : 9.7 (R2022a) 13-Nov-2021
 * C/C++ source code generated on : Sun Mar 19 04:03:56 2023
 *
 * Target selection: ert.tlc
 * Embedded hardware selection: Intel->x86-64 (Mac OS X)
 * Code generation objectives: Unspecified
 * Validation result: Not run
 */

#ifndef RTW_HEADER_Vehicle_Speedometer_Module_h_
#define RTW_HEADER_Vehicle_Speedometer_Module_h_
#ifndef Vehicle_Speedometer_Module_COMMON_INCLUDES_
#define Vehicle_Speedometer_Module_COMMON_INCLUDES_
#include "rtwtypes.h"
```

```c
#endif                         /*
Vehicle_Speedometer_Module_COMMON_INCLUDES_ */


#include "Vehicle_Speedometer_Module_types.h"


/* Macros for accessing real-time model data structure */

#ifndef rtmGetErrorStatus

#define rtmGetErrorStatus(rtm)         ((rtm)->errorStatus)

#endif


#ifndef rtmSetErrorStatus

#define rtmSetErrorStatus(rtm, val)    ((rtm)->errorStatus = (val))

#endif


/* Block signals (default storage) */

typedef struct {

  uint16_T DisplaySpeed_Filter;        /* '<S2>/Add' */

} B_Vehicle_Speedometer_Module_T;


/* External inputs (root inport signals with default storage) */

typedef struct {

  uint8_T In_MainFilt_SpeedValue;      /* '<Root>/
In_MainFilt_SpeedValue' */

  uint8_T In_AuxFilt_SpeedValue;       /* '<Root>/
In_AuxFilt_SpeedValue' */

  uint16_T Avg_VehicleSpeed;           /* '<Root>/Avg_VehicleSpeed'
*/

  uint16_T Timer_Input;                /* '<Root>/Timer_Input' */

} ExtU_Vehicle_Speedometer_Modu_T;
```

```c
/* External outputs (root outports fed by signals with default
storage) */

typedef struct {

  uint16_T Out_VehicleSpeed_Display;   /* '<Root>/
Out_VehicleSpeed_Display' */

} ExtY_Vehicle_Speedometer_Modu_T;


/* Real-time Model Data Structure */

struct tag_RTM_Vehicle_Speedometer_M_T {

  const char_T * volatile errorStatus;

};


/* Block signals (default storage) */

extern B_Vehicle_Speedometer_Module_T Vehicle_Speedometer_Module_B;


/* External inputs (root inport signals with default storage) */

extern ExtU_Vehicle_Speedometer_Modu_T Vehicle_Speedometer_Module_U;


/* External outputs (root outports fed by signals with default
storage) */

extern ExtY_Vehicle_Speedometer_Modu_T Vehicle_Speedometer_Module_Y;


/* Model entry point functions */

extern void Vehicle_Speedometer_Module_initialize(void);

extern void Vehicle_Speedometer_Module_step(void);

extern void Vehicle_Speedometer_Module_terminate(void);
```

```
/* Real-time Model object */
```

```c
/*
 * Academic License - for use in teaching, academic research, and
meeting
 * course requirements at degree granting institutions only.  Not
for
 * government, commercial, or other organizational use.
 *
 * File: Vehicle_Speedometer_Module.c
 *
 * Code generated for Simulink model 'Vehicle_Speedometer_Module'.
 *
 * Model version                  : 1.35
 * Simulink Coder version         : 9.7 (R2022a) 13-Nov-2021
 * C/C++ source code generated on : Sun Mar 19 04:03:56 2023
 *
 * Target selection: ert.tlc
 * Embedded hardware selection: Intel->x86-64 (Mac OS X)
 * Code generation objectives: Unspecified
 * Validation result: Not run
 */

#include "Vehicle_Speedometer_Module.h"
#include "rtwtypes.h"
#include "Vehicle_Speedometer_Module_private.h"

/* Block signals (default storage) */
B_Vehicle_Speedometer_Module_T Vehicle_Speedometer_Module_B;
```

```
/* External inputs (root inport signals with default storage) */

ExtU_Vehicle_Speedometer_Modu_T Vehicle_Speedometer_Module_U;


/* External outputs (root outports fed by signals with default
storage) */

ExtY_Vehicle_Speedometer_Modu_T Vehicle_Speedometer_Module_Y;


/* Real-time model */

static RT_MODEL_Vehicle_Speedometer__T
Vehicle_Speedometer_Module_M_;

RT_MODEL_Vehicle_Speedometer__T *const Vehicle_Speedometer_Module_M
=

  &Vehicle_Speedometer_Module_M_;


/* Model step function */

void Vehicle_Speedometer_Module_step(void)

{

  int16_T rtb_Add;

  uint16_T rtb_Add_0;

  uint16_T tmp;


  /* Outputs for Atomic SubSystem: '<Root>/Speedometer_Module' */

  /* Saturate: '<S5>/Saturation' incorporates:

   *  Inport: '<Root>/Avg_VehicleSpeed'

   */

  if (Vehicle_Speedometer_Module_U.Avg_VehicleSpeed > 25000) {


    /* Saturate: '<S5>/Saturation' */


    Vehicle_Speedometer_Module_Y.Out_VehicleSpeed_Display = 25000U;
```

```
  } else if (Vehicle_Speedometer_Module_U.Avg_VehicleSpeed < 200) {

    /* Saturate: '<S5>/Saturation' */

    Vehicle_Speedometer_Module_Y.Out_VehicleSpeed_Display = 200U;
  } else {

    /* Saturate: '<S5>/Saturation' */

    Vehicle_Speedometer_Module_Y.Out_VehicleSpeed_Display =

      Vehicle_Speedometer_Module_U.Avg_VehicleSpeed;
  }


  /* End of Saturate: '<S5>/Saturation' */


  /* Outputs for Enabled SubSystem: '<S1>/Auxilliary_Data_Filtering'
incorporates:

   *  EnablePort: '<S2>/Enable'

   */

  /* Outputs for Enabled SubSystem: '<S1>/Main_Data_Filtering'
incorporates:

   *  EnablePort: '<S6>/Enable'

   */

  /* RelationalOperator: '<S3>/Compare' incorporates:

   *  Constant: '<S3>/Constant'

   *  Inport: '<Root>/Timer_Input'

   *  Math: '<S1>/Rem'

   */

  if ((uint16_T)(Vehicle_Speedometer_Module_U.Timer_Input % 10) ==
0) {
```

```
    /* Product: '<S6>/Divide' incorporates:
     *  Gain: '<S6>/Gain'
     *  Inport: '<Root>/In_MainFilt_SpeedValue'
     *  Saturate: '<S5>/Saturation'
     */
    if (Vehicle_Speedometer_Module_U.In_MainFilt_SpeedValue == 0U) {
      rtb_Add_0 = MAX_uint16_T;

      /* Divide by zero handler */
    } else {
      rtb_Add_0 = (uint16_T)((uint32_T)(uint16_T)((((65U *
        Vehicle_Speedometer_Module_Y.Out_VehicleSpeed_Display) >> 7)
* 2684355UL)
        >> 21) /
Vehicle_Speedometer_Module_U.In_MainFilt_SpeedValue);
    }
```

```
    /* Product: '<S2>/Divide' incorporates:

     *  Constant: '<S6>/Constant'

     *  Gain: '<S2>/Gain'

     *  Inport: '<Root>/In_AuxFilt_SpeedValue'

     *  Product: '<S6>/Divide'

     *  Sum: '<S6>/Add'

     */

    if (Vehicle_Speedometer_Module_U.In_AuxFilt_SpeedValue == 0U) {

      rtb_Add_0 = MAX_uint16_T;



      /* Divide by zero handler */

    } else {

      rtb_Add_0 = (uint16_T)((uint32_T)(uint16_T)(((((uint16_T)
(rtb_Add_0 - 5U) *

        39U) >> 6) * 2684355UL) >> 21) /

        Vehicle_Speedometer_Module_U.In_AuxFilt_SpeedValue);

    }
```

```
    /* Sum: '<S2>/Add' incorporates:
     *  Constant: '<S2>/Constant'
     *  Product: '<S2>/Divide'
     */
    Vehicle_Speedometer_Module_B.DisplaySpeed_Filter = (uint16_T)
(rtb_Add_0 - 3U);
  }

  /* End of RelationalOperator: '<S3>/Compare' */
  /* End of Outputs for SubSystem: '<S1>/Main_Data_Filtering' */
  /* End of Outputs for SubSystem: '<S1>/Auxilliary_Data_Filtering'
*/

  /* Sum: '<S4>/Add' incorporates:
   *  Saturate: '<S5>/Saturation'
   *  Sum: '<S2>/Add'
   */
  rtb_Add_0 = Vehicle_Speedometer_Module_B.DisplaySpeed_Filter;
  if (Vehicle_Speedometer_Module_B.DisplaySpeed_Filter > 32767) {
    rtb_Add_0 = 32767U;
  }

  tmp = Vehicle_Speedometer_Module_Y.Out_VehicleSpeed_Display;
```

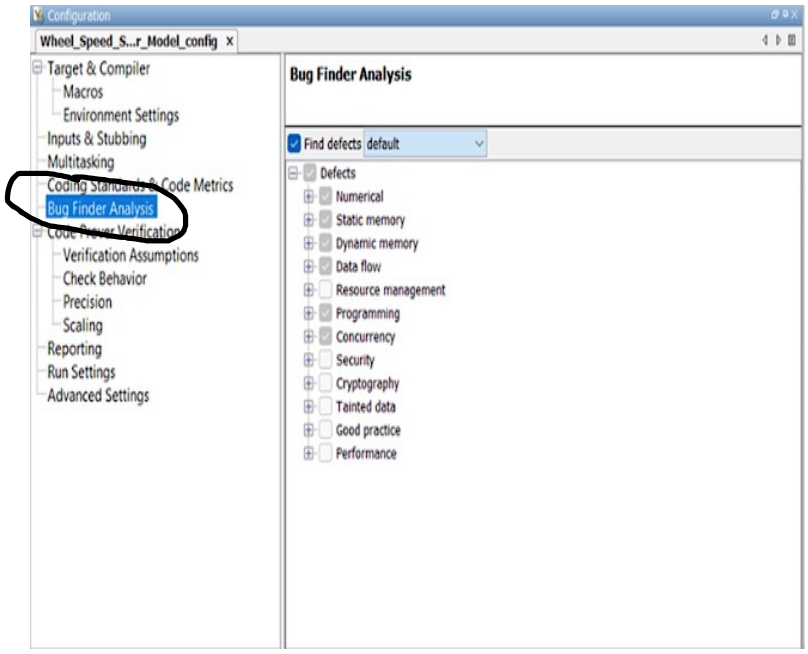
```
  if (Vehicle_Speedometer_Module_Y.Out_VehicleSpeed_Display > 32767)
{

    tmp = 32767U;

  }


  rtb_Add = (int16_T)(rtb_Add_0 - tmp);


  /* End of Sum: '<S4>/Add' */


  /* Abs: '<S4>/Abs' incorporates:

   *  Sum: '<S4>/Add'

   */

  if (rtb_Add < 0) {

    rtb_Add_0 = (uint16_T)-rtb_Add;

  } else {

    rtb_Add_0 = (uint16_T)rtb_Add;

  }


  /* End of Abs: '<S4>/Abs' */


  /* Switch: '<S4>/Switch' incorporates:

   *  RelationalOperator: '<S4>/Relational Operator'

   */

  if (rtb_Add_0 <= 150) {

    /* Saturate: '<S5>/Saturation' incorporates:
```

```
     *  Outport: '<Root>/Out_VehicleSpeed_Display'
     *  Sum: '<S2>/Add'
     */
    Vehicle_Speedometer_Module_Y.Out_VehicleSpeed_Display =
      Vehicle_Speedometer_Module_B.DisplaySpeed_Filter;
  }

  /* End of Switch: '<S4>/Switch' */
  /* End of Outputs for SubSystem: '<Root>/Speedometer_Module' */
}

/* Model initialize function */
void Vehicle_Speedometer_Module_initialize(void)
{
  /* (no initialization code required) */
}

/* Model terminate function */
void Vehicle_Speedometer_Module_terminate(void)
{
  /* (no terminate code required) */
}

/*
 * File trailer for generated code.
```

```
*
* [EOF]
*/
```

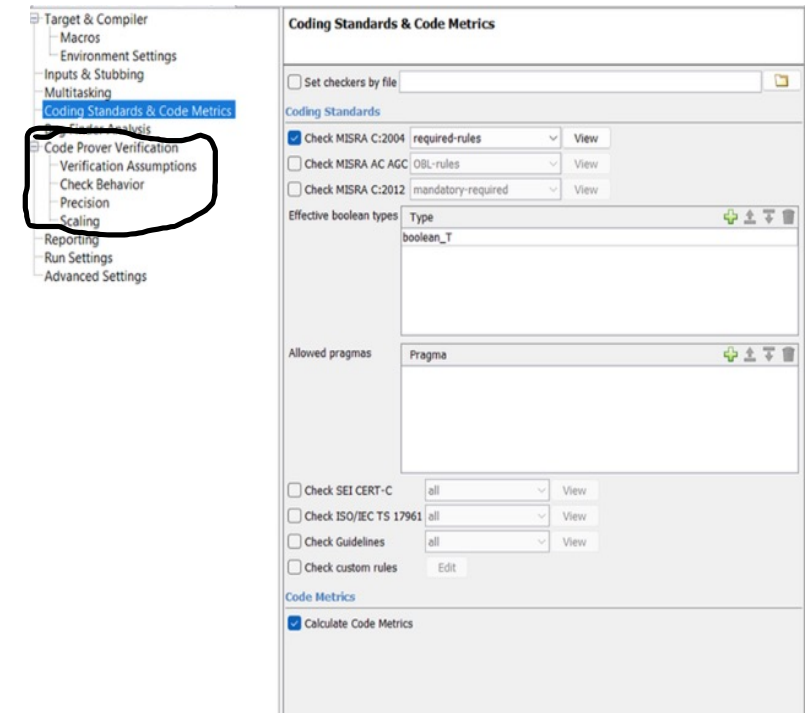
11. Static Code Analsysis (Polyspace)

Bug Finder:

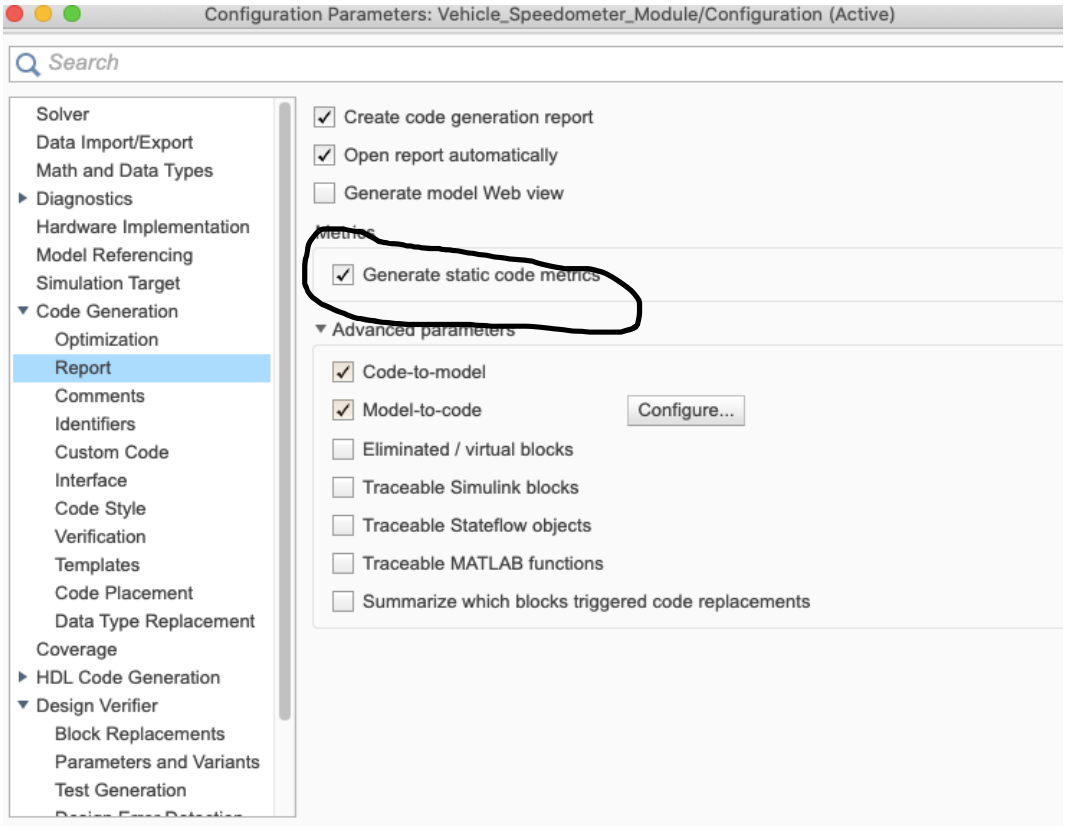


- In the configuration settings option called Target & compiler select the source code language as c, change the target processor as X86_64.

Code Prover:



- In Configuration>Code prover verification tab
- In Precision Select the Precision level -2 and Verification Level is Software Safety Analysis Level-2. then After Click on Save Button the Configuration setting will be save then After one can to go for the Code Prover running.



- For static code ananlysis, one has to check up the *Generate static code metrics* during Code Generation stage.