Model Advisor Report – dc_motor.slx

Simulink version: 10.5 Model version: 1.15 Current run: 25-Mar-2022 15:01:49 System: dc motor Treat as Referenced Model: off **Run Summary** Incomplete Justified **Passed** Failed Warning Not Run Total 0 <u> 1</u>0 **O 6**5 **118** 193 Model Advisor Identify lookup table blocks that generate expensive out-of-range checking code Not Run Electric Check configuration parameters for generation of inefficient saturation code Not Run ☐ Check for blocks not recommended for C/C++ production code deployment Not Run ☐ Check output types of logic blocks Not Run ☐ Check the hardware implementation Not Run

■ Identify questionable software environment specifications Not Run
☐ Identify questionable code instrumentation (data I/O) Not Run
Identify blocks generating inefficient algorithms Not Run
☐ Check configuration parameters for MISRA C:2012 Not Run
Check for blocks not recommended for MISRA C:2012 Not Run
☐ Check for unsupported block names Not Run
Check usage of Assignment blocks Not Run
Check for switch case expressions without a default case Not Run
☐ Check for missing error ports in AUTOSAR receiver interfaces Not Run
☐ Check configuration parameters for secure coding standards Not Run
☐ Check for blocks not recommended for secure coding standards Not Run

ldentify questionable subsystem settings Not Run
Check for blocks not supported for row-major code generation Not Run
Identify TLC S-Functions with unset array layout Not Run
Identify blocks that generate expensive fixed-point and saturation code Not Run
Check for missing const qualifiers in model functions Not Run
Check bus object names that are used as bus element names Not Run
ldentify questionable fixed-point operations Not Run
Identify blocks that generate expensive rounding code Not Run
Check for bitwise operations on signed integers Not Run
Check for recursive function calls Not Run

Check for equality and inequality operations on floating-point values Not Run
Check integer word lengths Not Run
Simulink
☐ Check optimization settings Not Run
Identify unconnected lines, input ports, and output ports Not Run
☐ Check root model Inport block specifications Not Run
El Check diagnostic settings ignored during accelerated model reference simulation Not Run
Check for parameter tunability information ignored for referenced models Not Run
☐ Check for implicit signal resolution Not Run
☐ Check for optimal bus virtuality Not Run
☐ Check for calls to slDataTypeAndScale() Not Run

☐ Check for Discrete-Time Integrator blocks with initial condition uncertainty Not Run
Identify disabled library links Not Run
Identify parameterized library links Not Run
Identify unresolved library links Not Run
Identify configurable subsystem blocks in the model for converting to variant subsystem blocks. Not Run
Check usage of function-call connections Not Run
El Check and update mask image display commands with unnecessary imread() function calls Not Run
Check and update mask to affirm icon drawing commands dependency on mask workspace Not Run
Identify Environment Controller blocks to be replaced with Variant Source blocks Not Run
Runtime diagnostics for S-functions Not Run
Check if Read/Write diagnostics are enabled for Data Store blocks Not Run

Check Data Store Memory blocks for multitasking, strong typing, and shadowing issues Not Run
Check Model History properties Not Run
E Check S-functions in the model Not Run
Open the Upgrade Advisor Not Run
Check structure parameter usage with bus signals Not Run
Check for large number of function arguments from virtual bus across model reference boundary Not Run
Check Delay, Unit Delay and Zero-Order Hold blocks for rate transition Not Run
Check bus signals treated as vectors Not Run
Check for potentially delayed function-call block return values Not Run
Identify block output signals with continuous sample time and non-floating point data type Not Run

Check usage of Merge blocks Not Run
Check usage of Outport blocks Not Run
Check usage of Discrete-Time Integrator blocks Not Run
Check model settings for migration to simplified initialization mode Not Run
Check for non-continuous signals driving derivative ports Not Run
Check data store block sample times for modeling errors Not Run
Check for potential ordering issues involving data store access Not Run
Identify unit mismatches in the model Not Run
Identify automatic unit conversions in the model Not Run
Identify disallowed unit systems in the model Not Run
Identify undefined units in the model Not Run

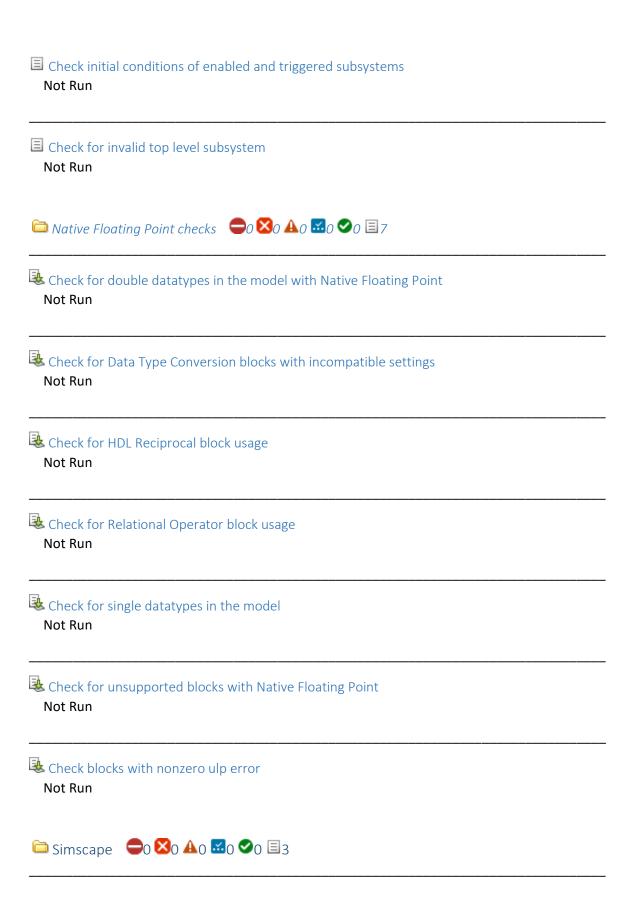
Identify ambiguous units in the model Not Run
Identify questionable operations for strict single-precision design Not Run
i Simulink Coder
Identify blocks using one-based indexing Not Run
☐ Check solver for code generation Not Run
☐ Check for blocks not supported by code generation Not Run
☐ Check for model reference configuration mismatch Not Run
Check code generation identifier formats used for model reference Not Run
Check for relative execution order change for Data Store Read and Data Store Write blocks Not Run
Check reuse of subsystem code Not Run
Check sample times and tasking mode

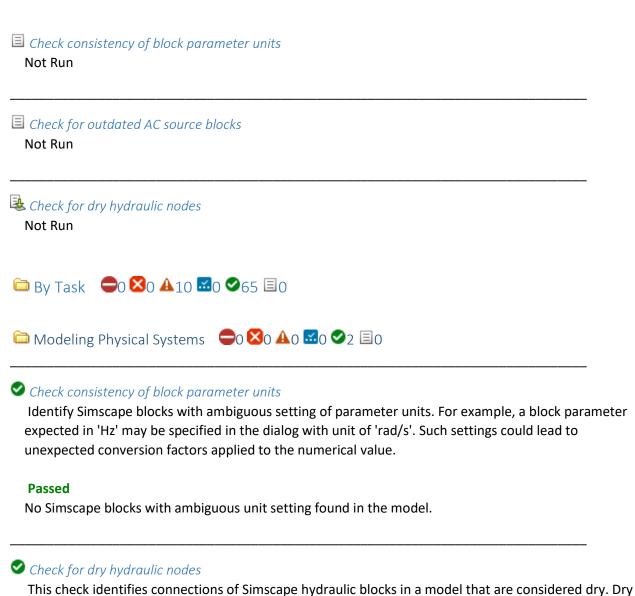
Not Run

6	□ HDL Coder
6	lacktriangle Checks for blocks and block settings $lacktriangle$ 0 l
	Check for unsupported blocks Not Run
	Check for HDL Reciprocal block usage Not Run
	Check for MATLAB Function block settings Not Run
	Check for obsolete Unit Delay Enabled/Resettable blocks Not Run
	Check for infinite and continuous sample time sources Not Run
	Check for unsupported storage class for signal objects Not Run
	Check for Stateflow chart settings Not Run

Check for large matrix operations Not Run
Check for blocks that have nonzero output latency Not Run
☐ Check architecture name Not Run
Check clock settings Not Run
Check clock, reset, and enable signals Not Run
Check file extension Not Run
Check generics Not Run
Check naming conventions Not Run
□ Check package file names Not Run
☐ Check signal and port names Not Run

☐ Check entity and architecture Not Run
☐ Check module/entity names Not Run
☐ Check top-level subsystem/port names Not Run
\bigcirc Model configuration checks \bigcirc 0 \bigcirc 0 \bigcirc 0 \bigcirc 0 \bigcirc 6
☐ Check delay balancing setting Not Run
☐ Check for global reset setting for Xilinx and Altera devices Not Run
☐ Check inline configurations setting Not Run
☐ Check for model parameters suited for the HDL code generation Not Run
☐ Check for visualization settings Not Run
Check algebraic loops Not Run
$\stackrel{ ext{lin}}{=}$ Checks for ports and subsystems $\begin{array}{c} ext{lin} & ext{lin}$





This check identifies connections of Simscape hydraulic blocks in a model that are considered dry. Dry nodes physically represent a hydraulic segment modeled as an incompressible fluid.

Passed

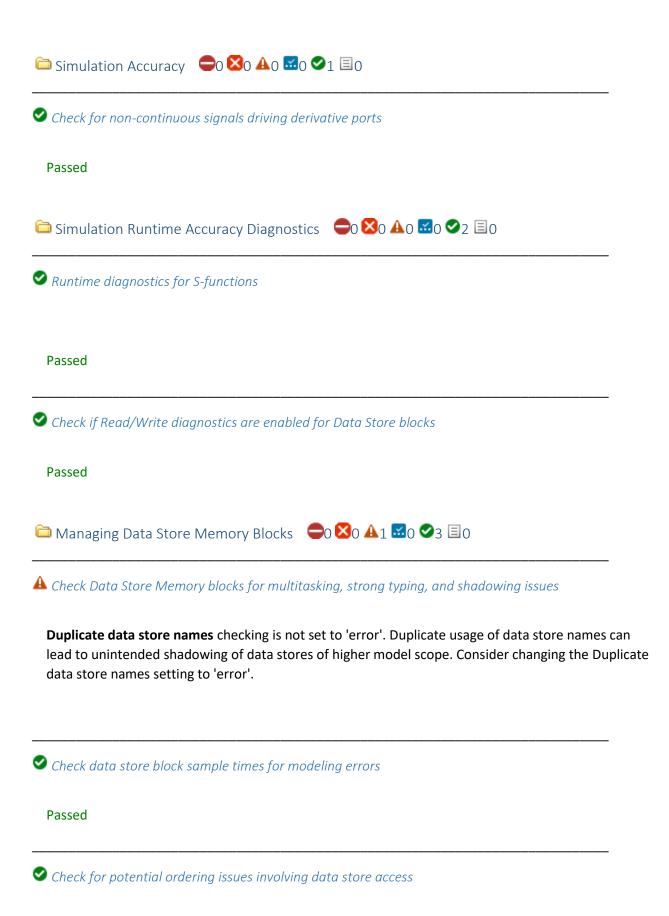
Check has passed. No dry hydraulic nodes found.



☑ Identify Environment Controller blocks to be replaced with Variant Source blocks

Passed

The model does not contain any Environment Controller blocks.



Check for relative execution order change for Data Store Read and Data Store Write blocks

The system does not have any Data Store Read or Data Store Write blocks.

⊘ Check Model History properties

Check models for edited Model History property values

Check that parameters in the Model Properties dialog History pane use the default tags. In the MDL file format you can configure some model properties to make use of source control tool keyword substitution. If you save your model in SLX format, source control tools cannot perform keyword substitution. Any information in the model file from such keyword substitution is cached when you first save the MDL file as SLX, and is never updated again. The Model Properties History pane and any Model Info blocks in your model show stale information from then on.

Passed

This model uses the default value for property ModifiedByFormat.

Passed

This model uses the default value for property ModifiedDateFormat.

Passed

This model uses the default value for property ModelVersionFormat.





There are no user-defined S-functions in the model.



Identify unit mismatches in the model

Check for unit mismatches in the model.

Passed

No unit mismatches found.

☑ Identify automatic unit conversions in the model Check for automatic unit conversions. **Passed** No automatic unit conversions found. ☑ Identify disallowed unit systems in the model Check for disallowed unit systems. **Passed** No disallowed unit systems were found. Identify undefined units in the model Check for undefined units. **Passed** No undefined units were found. ☑ Identify ambiguous units in the model Check for ambiguous units. **Passed** No ambiguous units were found. Modeling Signals and Parameters using Buses
□ ■0 ■0 ■1 ■0 ●2 ■0 **⊘** Check for optimal bus virtuality Passed **⊘** Check structure parameter usage with bus signals This test is skipped because it requires an activated Simulink Coder product **A** Check bus signals treated as vectors

Bus signal treated as vector

Identify bus signals in the model that are treated as vectors by the Simulink software.

Warning

The model uses bus signals properly. However, the model is not configured to detect future changes that might result in improper bus signal usage.

Recommended Action

To detect these changes, in the Configuration Parameters dialog box, set the Bus signal treated as vector diagnostic to error.

Buses - Bus signal treated as vector





A Check optimization settings

Check optimization settings

Warning

The model configuration parameters are not set to the recommended values specified in the data file.

Status	Parameter	Current Value	Recommended Values
Warning	StateBitsets	off	on
Warning	DataBitsets	off	on

Recommended Action

Follow the links in the result table to modify the model configuration parameters.

☑ Identify blocks using one-based indexing

Check the model for blocks configured for one-based indexing

Passed All blocks in the model use zero-based indexing.	_
Identify questionable software environment specifications Passed	
Identify lookup table blocks that generate expensive out-of-range checking code Passed	
Identify questionable code instrumentation (data I/O) Passed	
 Check output types of logic blocks Identify logic blocks that are outputting non-Boolean data types. Passed There are no logic blocks in the model or subsystem. 	
Check configuration parameters for generation of inefficient saturation code Passed	
✓ Identify blocks that generate expensive rounding code Check for expensive rounding operations in multiplication and division Passed	
Check Optimization and Hardware Implementation settings (Lookup Blocks)	-
Passed	
Check for expensive rounding in a data type conversion	-
Passed	

	Check for expensive rounding modes in the model		
	Passed		
?	Identify questionable fixed-point operations Check for multiword operations		
	Passed		
	Check for expensive multiplication code		
	Passed		
	Check for expensive division code		
	Passed		
	Identify lookup blocks with uneven breakpoint spacing		
	Passed		
	Check for expensive pre-lookup division		
	Passed		
	Check for expensive data type conversions		
	Passed		
	Check for fixed-point comparisons with predetermined results		
	Passed		
	Check for expensive binary comparison operations		
	Passed		

Check for expensive fixed-point types			
hat generate expensive fixed-point and saturation code ocks for questionable fixed-point operations			
nal Operation blocks for questionable fixed-point operations			
pe Conversion Inherited blocks for questionable fixed-point operations			
blocks for questionable fixed-point operations			
ocks for questionable fixed-point operations			
e blocks for questionable fixed-point operations			

Identify Min Max blocks for questionable fixed-point operations				
Passed				
Identify Discrete Integrator blocks for questionable fixed-point operations				
Passed				
Identify Compare To Constant blocks for questionable fixed point exertions				
Identify Compare To Constant blocks for questionable fixed-point operations				
Passed				
Identify Lookup Table blocks for questionable fixed-point operations				
Passed				
Identify blocks that will invoke net slope computation				
Passed				
Identify Product blocks that are less efficient				
Passed				
Check for expensive saturation code				
Passed				

 Identify blocks generating inefficient algorithms Passed No inefficient algorithms found in the model. 							
Modeling	Single-Precision Systems 😑 😆 0 🗸	∆ 1 ≅ 0 ⊘ 0 ≣ 0					
	stionable operations for strict single-pre	•		_			
This check ve design.	erifies the status of model settings that	will help you ach	ieve a strict single-precis	ion			
Warning							
The following	g model settings are non-optimal to a s	ingle-precision de	esign:				
Model Name	Configuration Parameter	Current Value	Recommended Value]			
dc_motor	Default for underspecified data type	double	single	_			
				Check			
for double p	recision operations						
	entifies blocks that introduce double-p ake sure that its port data types and in	•		ne check			
Warning							

The following blocks use double-precision floating-point operations:

- dc_motor/Subsystem/to rpm
- dc_motor/Subsystem/multiplier
- dc_motor/Scope1
- dc_motor/Ka*phi
- dc_motor/Subsystem/1//La
- dc_motor/Subsystem/Ka*phi*la
- dc_motor/Subsystem/Scope2
- dc_motor/Subsystem/Product
- dc_motor/Subsystem/tl
- dc_motor/Subsystem/Constant
- dc_motor/Subsystem/Product1
- dc_motor/Subsystem/Add
- dc_motor/Subsystem/B
- dc_motor/Subsystem/Add3
- dc_motor/Subsystem/1//J
- dc_motor/Va
- dc_motor/Subsystem/Ra
- dc_motor/Subsystem/Ka*phi*w
- dc_motor/Subsystem/Add2
- dc_motor/Subsystem/Integrator3
- dc_motor/Subsystem/Integrator2

Λ Less

☐ Migrating to Simplified Initialization mode ☐0 ☎0 ☎0 ☎0 ☎4 亘0	
Check usage of Merge blocks Check usage of Merge blocks	
This check finds and reports issues related to merge blocks for migrating to simplified initialize mode.	zatio
See Also	
Check usage of Merge blocks	
Underspecified initialization detection	
Passed	
Check usage of Outport blocks Check usage of Outport blocks	
This check finds and reports issues related to Outport blocks and Conditional Subsystems for migrating to simplified initialization mode.	
See Also	
Check usage of Outport blocks	
Underspecified initialization detection	
Passed	

⊘ Check usage of Discrete-Time Integrator blocks

Check usage of Discrete-Time Integrator blocks

This check finds and reports issues related to Discrete-Time Integrator blocks for migrating to simplified initialization mode

See Also

- Check usage of Discrete-Time Integrator blocks
- Underspecified initialization detection

Passed

Check model settings for migration to simplified initialization mode

Check for model level messages

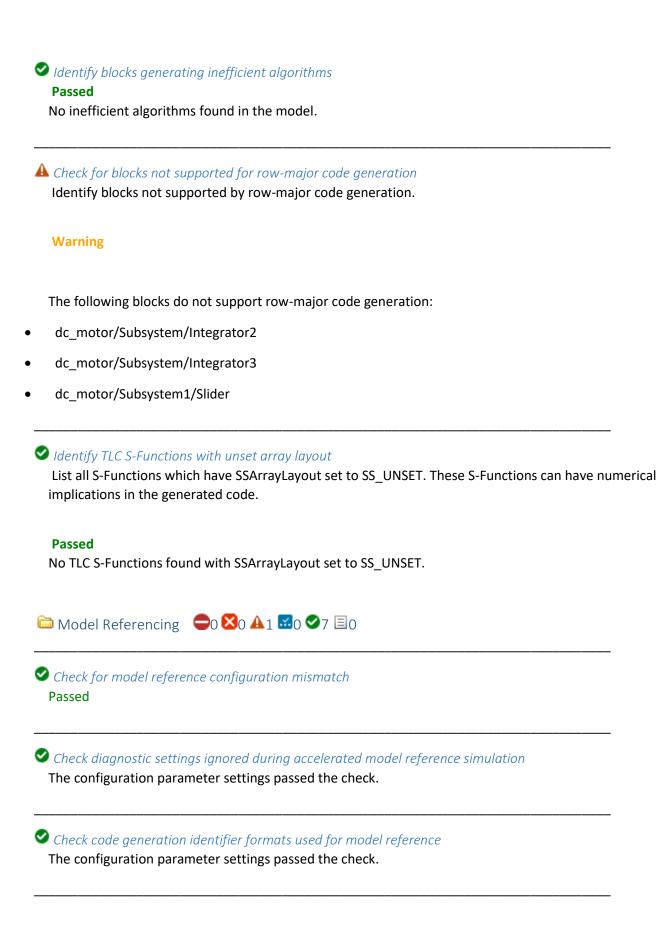
This check finds and reports model level messages for migrating to simplified initialization mode.

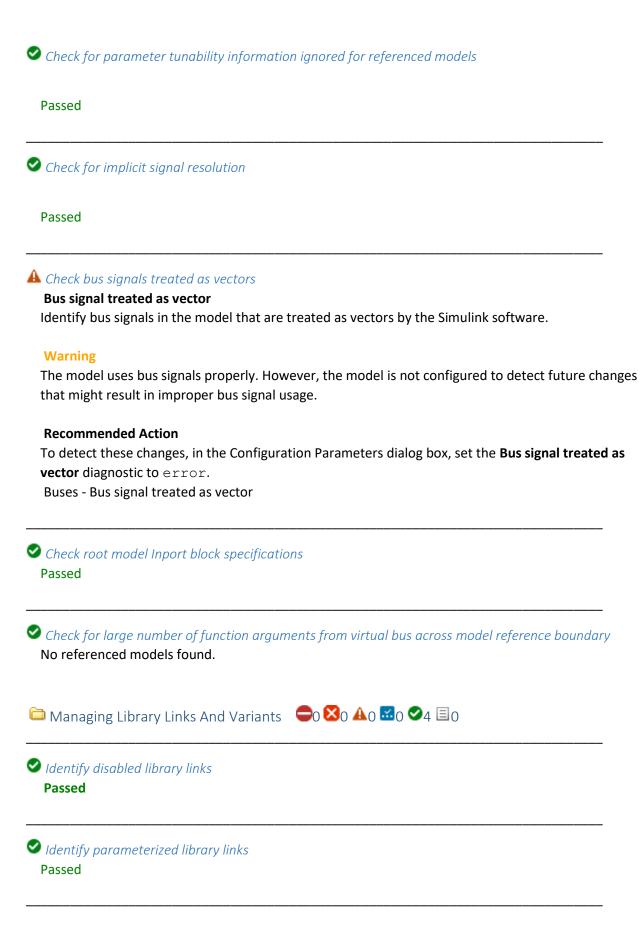
See Also

- Check model settings for migration to simplified initialization mode
- Underspecified initialization detection

Passed

 $\stackrel{ ext{lin}}{=}$ Row-Major Code Generation $\stackrel{ ext{lin}}{=}$ 0 $\stackrel{ ext{lin}}{≤}$ 0 $\stackrel{ ext{lin}}{≤}$ 2 $\stackrel{ ext{lin}}{≡}$ 0





Identify unresolved library links
Passed

Identify configurable subsystem blocks in the model for converting to variant subsystem blocks. Identify and upgrade Configurable Subsystem blocks in the model or subsystem level.

Passed

No configurable subsystem blocks found.

- Data Transfer Efficiency →0 🔀0 🗚0 🚾0 💇1 🗏 0
- **⊘** Check Delay, Unit Delay and Zero-Order Hold blocks for rate transition

Passed

The model does not contain Delay, Unit Delay or Zero-Order Hold blocks that perform rate transition.

- _____
- ⚠ Check configuration parameters for MISRA C:2012

Identify configuration parameters that might impact MISRA C:2012 compliant code generation.

Warning

The model configuration parameters are not set to the recommended values specified in the data file.

Statu s	Parameter	Current Value	Recommended Values	Prerequisites
Warni ng	Model Verification block enabling (AssertControl)	UseLocalSet tings	DisableAll	

D - Warni ng	UtilityFuncGeneration	Auto	Shared location	
Warni ng	GenerateSharedConstants	Prerequisite constraint not met.	off	UtilityFuncGene ration
D - Warni ng	SystemTargetFile	Non-ERT based target	ERT based target	
Warni ng	SupportContinuousTime	Prerequisite constraint not met.	off	SystemTargetFil e
Warni ng	SupportNonInlinedSFcns	Prerequisite constraint not met.	off	SystemTargetFil e
Warni ng	MatFileLogging	on	off	
Warni ng	ParenthesesLevel	Prerequisite constraint not met.	Standards, Maximum	SystemTargetFil e
Warni ng	CastingMode	Prerequisite constraint not met.	Standards	SystemTargetFil e
Warni ng	InternalIdentifier	Prerequisite constraint not met.	Shortened	SystemTargetFil e
Warni ng	Use division for fixed-point net slope computation (UseDivisionForNetSlopeComputation)	off	on, UseDivisionForReciprocalsOfIn tegersOnly	
Warni ng	EnableSignedLeftShifts	Prerequisite constraint not met.	off	SystemTargetFil e
Warni ng	EnableSignedRightShifts	Prerequisite constraint not met.	off	SystemTargetFil e

Warni ng	Inf or NaN block output (SignalInfNanChecking)	none	warning	
Warni ng	Dynamic memory allocation in MATLAB functions (MATLABDynamicMemAlloc)	on	off	
Warni ng	Undirected event broadcasts (SFUndirectedBroadcastEve ntsDiag)	warning	error	
Warni ng	Compile-time recursion limit for MATLAB functions (CompileTimeRecursionLimit)	50	0	
Warni ng	Enable run-time recursion for MATLAB functions (EnableRuntimeRecursion)	on	off	
Warni ng	MATLABFcnDesc	Prerequisite constraint not met.	on	GenerateComm ents, SystemTargetFil e

Λ Less

Recommended Action

Modify the configuration parameters listed above to the recommended values.

✓ Check for blocks not recommended for C/C++ production code deployment
 Passed

Check for blocks not recommended for MISRA C:2012 Passed
Check for unsupported block names Passed
Check usage of Assignment blocks Passed
Check for switch case expressions without a default case dentify switch case expressions that do not have a default case.
Passed Il switch case expressions have default cases.
Check for missing error ports in AUTOSAR receiver interfaces dentify AUTOSAR receiver interface ports that do not have a matching error port. Passed Model is not configured as an AUTOSAR target.
Check for bitwise operations on signed integers dentify bitwise operations on signed integers.
Passed lo bitwise operations on signed integers found.
Check for recursive function calls dentify function calls that are recursive.
Passed To recursive function calls found.



No equality or inequality operations on floating-point values found.

⊘ Check for missing const qualifiers in model functions

Identify missing const qualifiers in model functions.

Passed

Model does not use customized model functions.

Check integer word lengths

Identify integer word length that are not compliant with hardware implementation settings.

Passed

All used integer word length are compliant with hardware implementation settings.

Check bus object names that are used as bus element names

Identify bus object names that are used as bus element names.

Passed

No bus object names are used as bus element names.





A Check configuration parameters for secure coding standards

Identify configuration parameters that might impact secure coding standards compliant code generation.

Warning

The model configuration parameters are not set to the recommended values specified in the data file.

Status	Parameter	Current Value	Recommende d Values	Prerequisites
Warnin	Model Verification block enabling (AssertControl)	UseLocalSetting s	DisableAll	
D - Warnin g	SystemTargetFile	Non-ERT based target	ERT based target	
Warnin	SupportContinuousTime	Prerequisite constraint not met.	off	SystemTargetFile
Warnin	SupportNonInlinedSFcns	Prerequisite constraint not met.	off	SystemTargetFile
Warnin	MatFileLogging	on	off	
Warnin	EnableSignedLeftShifts	Prerequisite constraint not met.	off	SystemTargetFile
Warnin	EnableSignedRightShifts	Prerequisite constraint not met.	off	SystemTargetFile
Warnin	Inf or NaN block output (SignalInfNanChecking)	none	warning	
Warnin	Dynamic memory allocation in MATLAB functions (MATLABDynamicMemAlloc)	on	off	
Warnin	Undirected event broadcasts (SFUndirectedBroadcastEventsDia g)	warning	error	
Warnin	Compile-time recursion limit for MATLAB functions (CompileTimeRecursionLimit)	50	0	

Warnin	Enable run-time recursion for MATLAB functions (EnableRuntimeRecursion)	on	off	
Warnin	MATLABFcnDesc	Prerequisite constraint not met.	on	GenerateComments , SystemTargetFile

Λ Less

Recommended Action

Modify the configuration parameters listed above to the recommended values.

⊘ Check for blocks not recommended for C/C++ production code deployment

Passed

Check for blocks not recommended for secure coding standards

Passed

Check usage of Assignment blocks

Passed

Check for switch case expressions without a default case Identify switch case expressions that do not have a default case.

Passed

All switch case expressions have default cases.

Check for bitwise operations on signed integers Identify bitwise operations on signed integers.

Passed

No bitwise operations on signed integers found.

Check for equality and inequality operations on floating-point values

Identify equality and inequality operations on floating-point values.

Passed

No equality or inequality operations on floating-point values found.



Output Check integer word lengths

Identify integer word length that are not compliant with hardware implementation settings.

Passed

All used integer word length are compliant with hardware implementation settings.













⚠ Check configuration parameters for MISRA C:2012

Identify configuration parameters that might impact MISRA C:2012 compliant code generation.

Warning

The model configuration parameters are not set to the recommended values specified in the data file.

Statu s	Parameter	Current Value	Recommended Values	Prerequisites
Warni ng	Model Verification block enabling (AssertControl)	UseLocalSet tings	DisableAll	
D - Warni	UtilityFuncGeneration	Auto	Shared location	

ng				
Warni ng	GenerateSharedConstants	Prerequisite constraint not met.	off	UtilityFuncGene ration
D - Warni ng	SystemTargetFile	Non-ERT based target	ERT based target	
Warni ng	SupportContinuousTime	Prerequisite constraint not met.	off	SystemTargetFil e
Warni ng	SupportNonInlinedSFcns	Prerequisite constraint not met.	off	SystemTargetFil e
Warni ng	MatFileLogging	on	off	
Warni ng	ParenthesesLevel	Prerequisite constraint not met.	Standards, Maximum	SystemTargetFil e
Warni ng	CastingMode	Prerequisite constraint not met.	Standards	SystemTargetFil e
Warni ng	InternalIdentifier	Prerequisite constraint not met.	Shortened	SystemTargetFil e
Warni ng	Use division for fixed-point net slope computation (UseDivisionForNetSlopeComputation)	off	on, UseDivisionForReciprocalsOfIn tegersOnly	
Warni ng	EnableSignedLeftShifts	Prerequisite constraint not met.	off	SystemTargetFil e
Warni ng	EnableSignedRightShifts	Prerequisite constraint not met.	off	SystemTargetFil e
Warni	Inf or NaN block output	none	warning	

ng	(SignalInfNanChecking)			
Warni ng	Dynamic memory allocation in MATLAB functions (MATLABDynamicMemAlloc)	on	off	
Warni ng	Undirected event broadcasts (SFUndirectedBroadcastEve ntsDiag)	warning	error	
Warni ng	Compile-time recursion limit for MATLAB functions (CompileTimeRecursionLimit)	50	0	
Warni ng	Enable run-time recursion for MATLAB functions (EnableRuntimeRecursion)	on	off	
Warni ng	MATLABFcnDesc	Prerequisite constraint not met.	on	GenerateComm ents, SystemTargetFil e

Λ Less

Recommended Action

Modify the configuration parameters listed above to the recommended values.





▲ Open the Upgrade Advisor

Warning

To check for upgrade issues, open the Upgrade Advisor.

Recommended Action

Click the link below to close the Model Advisor and open the Upgrade Advisor for dc_motor. Open the Upgrade Advisor