# Model Advisor Report – AMBD_GlucoMeter_Yash.slx

**Simulink version:** 10.5 **Model version:** 1.4

**System:** AMBD_GlucoMeter_Yash **Current run:** 26-Mar-2022 09:28:29

**Treat as Referenced Model:** off

**Run Summary**

| Incomplete | Failed | Warning | Justified | Passed | Not Run | Total |
|---|---|---|---|---|---|---|
| 0 | 0 | 9 | 0 | 64 | 115 | 188 |

## Model Advisor

### By Product 0 0 0 0 0 115

### Embedded Coder 0 0 0 0 0 28

---

*Identify lookup table blocks that generate expensive out-of-range checking code*

Not Run

---

*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

---

*Identify 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

---

*Identify 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 0 0 0 0 0 42

---

*Check optimization settings*

Not Run

---

*Identify unconnected lines, input ports, and output ports*

Not Run

---

*Check root model Inport block specifications*

Not Run

---

*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

---

*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

---

*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

Simulink Coder 0 0 0 0 0 9

---

*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

---

*Check for blocks that have constraints on tunable parameters*

Not Run

---

HDL Coder 0 0 0 0 0 36

---

*Checks for blocks and block settings* *0 0 0 0 0 10*

---

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 Trigonometric Function block for LUT-based approximation method

Not Run

---

Check for large matrix operations

Not Run

---

Check for blocks that have nonzero output latency

Not Run

*Industry standard checks* *0* *0* *0* *0* *0* *11*

---

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

*Model configuration checks* 0 0 0 0 0 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

*Checks for ports and subsystems* 0 0 0 0 0 2

---

Check initial conditions of enabled and triggered subsystems

Not Run

---

Check for invalid top level subsystem

Not Run

*Native Floating Point checks* 0 0 0 0 0 7

---

Check for double datatypes in the model with Native Floating Point

Not Run

---

Check for Data Type Conversion blocks with incompatible settings

Not Run

---

Check for HDL Reciprocal block usage

Not Run

---

Check for Relational Operator block usage

Not Run

---

Check for single datatypes in the model

Not Run

---

Check for unsupported blocks with Native Floating Point

Not Run

---

Check blocks with nonzero ulp error

Not Run

By Task 0 0 9 0 64 0

Replacing Blocks That Will Be Removed 0 0 0 0 1 0

---

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

**Passed**

The model does not contain any Environment Controller blocks.

Simulation Accuracy 0 0 0 0 1 0

---

*Check for non-continuous signals driving derivative ports*

Passed

Simulation Runtime Accuracy Diagnostics 0 0 0 0 2 0

---

*Runtime diagnostics for S-functions*

Passed

---

*Check if Read/Write diagnostics are enabled for Data Store blocks*

Passed

Managing Data Store Memory Blocks 0 0 1 0 3 0

---

*Check Data Store Memory blocks for multitasking, strong typing, and shadowing issues*

**Duplicate data store names** checking is not set to 'error'. Duplicate usage of data store names can lead to unintended shadowing of data stores of higher model scope. Consider changing the Duplicate data store names setting to 'error'.

---

*Check data store block sample times for modeling errors*

Passed

---

*Check for potential ordering issues involving data store access*

Passed

---

*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.

## Simulink Model File Integrity 0 0 0 0 1 0

---

*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.

## S-function Checks 0 0 0 0 1 0

---

*Check S-functions in the model*

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

## Units Inconsistencies 0 0 0 0 5 0

---

*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

---

*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

## Code Generation Efficiency 0 0 1 0 10 0

---

*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**

All logic blocks are being used appropriately.

---

*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**

**Passed**

---

*Identify blocks that generate expensive fixed-point and saturation code*

**Identify Sum blocks for questionable fixed-point operations**

**Passed**

---

**Identify Relational Operation blocks for questionable fixed-point operations**

**Passed**

---

**Identify Data Type Conversion Inherited blocks for questionable fixed-point operations**

**Passed**

---

**Identify Switch blocks for questionable fixed-point operations**

**Passed**

---

**Identify Logic blocks for questionable fixed-point operations**

**Passed**

---

**Identify Saturate blocks for questionable fixed-point operations**

**Passed**

---

**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 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 0 1 0 0 0

---

*Identify questionable operations for strict single-precision design*

**Check model settings related to single-precision design**

This check verifies the status of model settings that will help you achieve a strict single-precision design.

**Warning**

The following model settings are non-optimal to a single-precision design:

| Model Name | Configuration Parameter | Current Value | Recommended Value |
|---|---|---|---|
| AMBD_GlucoMeter_Yash | Default for underspecified data type | double | single |

**Check for double precision operations**

This check identifies blocks that introduce double-precision operations. For each block that the check identifies, make sure that its port data types and intermediate settings are set correctly.

**Warning**

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

- AMBD_GlucoMeter_Yash/Subsystem/Switch

  AMBD_GlucoMeter_Yash/Subsystem/Switch2

  AMBD_GlucoMeter_Yash/Subsystem/Switch1
- AMBD_GlucoMeter_Yash/Subsystem/Constant4
- AMBD_GlucoMeter_Yash/Subsystem/Switch3

  AMBD_GlucoMeter_Yash/Subsystem/Switch2

  AMBD_GlucoMeter_Yash/Subsystem/Switch1

  AMBD_GlucoMeter_Yash/Subsystem/Switch
- AMBD_GlucoMeter_Yash/Subsystem/Constant3
- AMBD_GlucoMeter_Yash/Subsystem/Switch2

  AMBD_GlucoMeter_Yash/Subsystem/Switch1

AMBD_GlucoMeter_Yash/Subsystem/Switch

- AMBD_GlucoMeter_Yash/Subsystem/Constant2
- AMBD_GlucoMeter_Yash/Subsystem/Switch1

AMBD_GlucoMeter_Yash/Subsystem/Switch

- AMBD_GlucoMeter_Yash/Subsystem/Constant1
- AMBD_GlucoMeter_Yash/Subsystem/Switch
- AMBD_GlucoMeter_Yash/Subsystem/Constant
- AMBD_GlucoMeter_Yash/Scope
- AMBD_GlucoMeter_Yash/Subsystem/Compare To Constant/Compare
- AMBD_GlucoMeter_Yash/Subsystem/Compare To Constant/Compare

AMBD_GlucoMeter_Yash/Subsystem/Compare To Constant/Constant

- AMBD_GlucoMeter_Yash/Subsystem/Compare To Constant1/Compare
- AMBD_GlucoMeter_Yash/Subsystem/Compare To Constant1/Compare

AMBD_GlucoMeter_Yash/Subsystem/Compare To Constant1/Constant

- AMBD_GlucoMeter_Yash/Subsystem/Compare To Constant2/Compare
- AMBD_GlucoMeter_Yash/Subsystem/Compare To Constant2/Compare

AMBD_GlucoMeter_Yash/Subsystem/Compare To Constant2/Constant

- AMBD_GlucoMeter_Yash/Subsystem/Compare To Constant3/Compare
- AMBD_GlucoMeter_Yash/Subsystem/Compare To Constant3/Compare

AMBD_GlucoMeter_Yash/Subsystem/Compare To Constant3/Constant

- AMBD_GlucoMeter_Yash/Signal Builder/FromWs

∧ Less

## Migrating to Simplified Initialization mode 0 0 0 0 4 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 initialization mode.

**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**

---

## Row-Major Code Generation 0 0 0 0 3 0

---

*Identify blocks generating inefficient algorithms*

**Passed**

No inefficient algorithms found in the model.

---

*Check for blocks not supported for row-major code generation*

Identify blocks not supported by row-major code generation.

**Passed**

No unsupported blocks found for row-major code generation

---

*Identify TLC S-Functions with unset array layout*

List all S-Functions which have SSArrayLayout set to SS_UNSET. These S-Functions can have numerical implications in the generated code.

**Passed**

No TLC S-Functions found with SSArrayLayout set to SS_UNSET.

## Model Referencing 0 0 1 0 7 0

---

*Check for model reference configuration mismatch*

Passed

---

*Check diagnostic settings ignored during accelerated model reference simulation*

The configuration parameter settings passed the check.

---

*Check code generation identifier formats used for model reference*

The configuration parameter settings passed the check.

---

*Check for parameter tunability information ignored for referenced models*

Passed

---

*Check for implicit signal resolution*

Passed

---

*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

---

*Check root model Inport block specifications*

Passed

---

*Check for large number of function arguments from virtual bus across model reference boundary*

No referenced models found.

## Managing Library Links And Variants 0 0 0 0 4 0

---

*Identify disabled library links*

**Passed**

---

*Identify parameterized library links*

Passed

---

*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.

## Modeling Standards for MISRA C:2012 0 0 1 0 12 0

*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.

| Status | Parameter | Current Value | Recommended Values | Prerequisites |
|---|---|---|---|---|
| Warning | Model Verification block enabling (AssertControl) | UseLocalSettings | DisableAll | |

| | | | | |
|---|---|---|---|---|
| D - Warning | UtilityFuncGeneration | Auto | Shared location | |
| Warning | GenerateSharedConstants | *Prerequisite constraint not met.* | off | UtilityFuncGeneration |
| D - Warning | SystemTargetFile | Non-ERT based target | ERT based target | |
| Warning | SupportContinuousTime | *Prerequisite constraint not met.* | off | SystemTargetFile |
| Warning | SupportNonInlinedSFcns | *Prerequisite constraint not met.* | off | SystemTargetFile |
| Warning | MatFileLogging | on | off | |
| Warning | ParenthesesLevel | *Prerequisite constraint not met.* | Standards, Maximum | SystemTargetFile |
| Warning | CastingMode | *Prerequisite constraint not met.* | Standards | SystemTargetFile |
| Warning | InternalIdentifier | *Prerequisite constraint not met.* | Shortened | SystemTargetFile |
| Warning | Use division for fixed-point net slope computation (UseDivisionForNetSlopeComputation) | off | on, UseDivisionForReciprocalsOfIntegersOnly | |
| Warning | EnableSignedLeftShifts | *Prerequisite constraint not met.* | off | SystemTargetFile |
| Warning | EnableSignedRightShifts | *Prerequisite constraint not met.* | off | SystemTargetFile |

| | | | | |
|---|---|---|---|---|
| Warning | Inf or NaN block output (SignalInfNanChecking) | none | warning | |
| Warning | Dynamic memory allocation in MATLAB functions (MATLABDynamicMemAlloc) | on | off | |
| Warning | Undirected event broadcasts (SFUndirectedBroadcastEventsDiag) | warning | error | |
| Warning | Compile-time recursion limit for MATLAB functions (CompileTimeRecursionLimit) | 50 | 0 | |
| Warning | Enable run-time recursion for MATLAB functions (EnableRuntimeRecursion) | on | off | |
| Warning | MATLABFcnDesc | *Prerequisite constraint not met.* | on | GenerateComments, SystemTargetFile |
| Warning | InstructionSetExtensions | SSE2 | None | |

∧ 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*

Identify switch case expressions that do not have a default case.

**Passed**

All switch case expressions have default cases.

---

*Check for missing error ports in AUTOSAR receiver interfaces*

Identify 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*

Identify bitwise operations on signed integers.

**Passed**

No bitwise operations on signed integers found.

---

*Check for recursive function calls*

Identify function calls that are recursive.

**Passed**

No recursive function calls 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.

---

*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.

## Modeling Standards for Secure Coding (CERT C, CWE, ISO/IEC TS 17961) 0 0 2 0 7 0

---

*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 | Recommended Values | Prerequisites |
|---|---|---|---|---|
| Warning | Model Verification block enabling (AssertControl) | UseLocalSettings | DisableAll | |
| D - Warning | SystemTargetFile | Non-ERT based target | ERT based target | |
| Warning | SupportContinuousTime | *Prerequisite constraint not met.* | off | SystemTargetFile |
| Warning | SupportNonInlinedSFcns | *Prerequisite constraint not met.* | off | SystemTargetFile |
| Warning | MatFileLogging | on | off | |
| Warning | EnableSignedLeftShifts | *Prerequisite constraint not met.* | off | SystemTargetFile |
| Warning | EnableSignedRightShifts | *Prerequisite constraint not met.* | off | SystemTargetFile |
| Warning | Inf or NaN block output (SignalInfNanChecking) | none | warning | |
| Warning | Dynamic memory allocation in MATLAB functions (MATLABDynamicMemAlloc) | on | off | |
| Warning | Undirected event broadcasts (SFUndirectedBroadcastEventsDiag) | warning | error | |

| | | | | |
|---|---|---|---|---|
| Warning | Compile-time recursion limit for MATLAB functions (CompileTimeRecursionLimit) | 50 | 0 | |
| Warning | Enable run-time recursion for MATLAB functions (EnableRuntimeRecursion) | on | off | |
| Warning | 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.

---

*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.

*High-Integrity Systems* 0 0 1 0 0 0

Code 0 0 1 0 0 0

---

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.

| Status | Parameter | Current Value | Recommended Values | Prerequisites |
|---|---|---|---|---|

| | | | | |
|---|---|---|---|---|
| Warning | Model Verification block enabling (AssertControl) | UseLocalSettings | DisableAll | |
| D - Warning | UtilityFuncGeneration | Auto | Shared location | |
| Warning | GenerateSharedConstants | *Prerequisite constraint not met.* | off | UtilityFuncGeneration |
| D - Warning | SystemTargetFile | Non-ERT based target | ERT based target | |
| Warning | SupportContinuousTime | *Prerequisite constraint not met.* | off | SystemTargetFile |
| Warning | SupportNonInlinedSFcns | *Prerequisite constraint not met.* | off | SystemTargetFile |
| Warning | MatFileLogging | on | off | |
| Warning | ParenthesesLevel | *Prerequisite constraint not met.* | Standards, Maximum | SystemTargetFile |
| Warning | CastingMode | *Prerequisite constraint not met.* | Standards | SystemTargetFile |
| Warning | InternalIdentifier | *Prerequisite constraint not met.* | Shortened | SystemTargetFile |
| Warning | Use division for fixed-point net slope computation (UseDivisionForNetSlopeComputation) | off | on, UseDivisionForReciprocalsOfIntegersOnly | |
| Warning | EnableSignedLeftShifts | *Prerequisite constraint not met.* | off | SystemTargetFile |

| | | | | |
|---|---|---|---|---|
| Warning | EnableSignedRightShifts | *Prerequisite constraint not met.* | off | SystemTargetFile |
| Warning | Inf or NaN block output (SignalInfNanChecking) | none | warning | |
| Warning | Dynamic memory allocation in MATLAB functions (MATLABDynamicMemAlloc) | on | off | |
| Warning | Undirected event broadcasts (SFUndirectedBroadcastEventsDiag) | warning | error | |
| Warning | Compile-time recursion limit for MATLAB functions (CompileTimeRecursionLimit) | 50 | 0 | |
| Warning | Enable run-time recursion for MATLAB functions (EnableRuntimeRecursion) | on | off | |
| Warning | MATLABFcnDesc | *Prerequisite constraint not met.* | on | GenerateComments, SystemTargetFile |
| Warning | InstructionSetExtensions | SSE2 | None | |

Λ Less

**Recommended Action**

Modify the configuration parameters listed above to the recommended values.

## Upgrading to the Current Simulink Version 0 0 1 0 0 0

---

### *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 AMBD_GlucoMeter_Yash.

Open the Upgrade Advisor