Model Advisor Report – Quadkopter_Model.slx

Simulink version: 8.8 **Model version:** 1.20

System: Quadkopter_Model/Attitude Controller **Current run:** 04-May-2017 15:14:45

Treat as Referenced Model: off

Run Summary

Pass	Fail	Warning	Not Run	Total
90	0	32	0	122

Simulink Verification and Validation

Modeling Standards 76 0 32 0

DO-178C/DO-331 Checks 18 0 18 0

Check safety-related optimization settings

Check optimization settings in the model configuration that might impact safety.

Warning

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

Status	Parameter	Current Value	Recommended Values
Warning	Block reduction (BlockReduction)	on	off
Warning	Application lifespan (days) (LifeSpan)	1	inf
Warning	Remove root level I/O zero initialization (ZeroExternalMemoryAtStartup) *	off	on

Warning	Remove internal data zero initialization (ZeroInternalMemoryAtStartup) *	off	on
Warning	Remove code that protects against division arithmetic exceptions (NoFixptDivByZeroProtection)	on	off

Recommended Action

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

* The Command-Line values provided in the table are reverse of the settings in the Configuration Parameters Dialog. Therefore, 'on' in the Command-Line corresponds to an "Off" setting in the dialog, and 'off' in the Command-Line corresponds to an "On" setting in the dialog.

 *Check safety-related diagnostic settings for solvers*

Check diagnostic settings in the model configuration that apply to solvers and might impact safety.

Warning

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

Status	Parameter	Current Value	Recommended Values
Warning	Algebraic loop (AlgebraicLoopMsg)	warning	error
Warning	Minimize algebraic loop (ArtificialAlgebraicLoopMsg)	warning	error
Warning	Block priority violation (BlockPriorityViolationMsg)	warning	error
Warning	Unspecified inheritability of sample time (UnknownTsInhSupMsg)	warning	error
Warning	Automatic solver parameter selection (SolverPrmCheckMsg)	warning	error
Warning	State name clash (StateNameClashWarn)	none	warning

Recommended Action

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

Check safety-related diagnostic settings for sample time

Check diagnostic settings in the model configuration that apply to sample time and might impact safety.

Warning

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

Status	Parameter	Current Value	Recommended Values
Warning	Source block specifies -1 sample time (InheritedTsInSrcMsg)	warning	error
Warning	Enforce sample times specified by Signal Specification blocks (SigSpecEnsureSampleTimeMsg)	warning	error

Recommended Action

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

Check safety-related diagnostic settings for signal data

Check diagnostic settings in the model configuration that apply to signal data and might impact safety.

Warning

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

Status	Parameter	Current Value	Recommended Values
Warning	Division by singular matrix (CheckMatrixSingularityMsg)	none	error

Warning	Underspecified data types (UnderSpecifiedDataTypeMsg)	none	error
Warning	Wrap on overflow (IntegerOverflowMsg)	warning	error
Warning	Inf or NaN block output (SignalInfNanChecking)	none	error
Warning	Simulation range checking (SignalRangeChecking)	none	error

Recommended Action

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

Check safety-related diagnostic settings for parameters

Check diagnostic settings in the model configuration that apply to parameters and might impact safety.

Warning

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

Status	Parameter	Current Value	Recommended Values
Warning	Detect underflow (ParameterUnderflowMsg)	none	error
Warning	Detect precision loss (ParameterPrecisionLossMsg)	warning	error

Recommended Action

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

Check safety-related diagnostic settings for data used for debugging

Check diagnostic settings in the model configuration that apply to data used for debugging and might impact safety.

Warning

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

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

Recommended Action

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

Check safety-related diagnostic settings for data store memory

Check diagnostic settings in the model configuration that apply to data store memory and might impact safety.

Warning

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

Status	Parameter	Current Value	Recommended Values
Warning	Detect read before write (ReadBeforeWriteMsg)	UseLocalSettings	EnableAllAsError
Warning	Detect write after read (WriteAfterReadMsg)	UseLocalSettings	EnableAllAsError
Warning	Detect write after write (WriteAfterWriteMsg)	UseLocalSettings	EnableAllAsError

Recommended Action

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

⚠ *Check safety-related diagnostic settings for type conversions*

Check diagnostic settings in the model configuration that apply to type conversions and might impact safety.

Warning

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

Status	Parameter	Current Value	Recommended Values
Warning	Unnecessary type conversions (UnnecessaryDatatypeConvMsg)	none	warning, error
Warning	Vector/matrix block input conversion (VectorMatrixConversionMsg)	none	error

Recommended Action

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

⚠ *Check safety-related diagnostic settings for signal connectivity*

Check diagnostic settings in the model configuration that apply to signal connectivity and might impact safety.

Warning

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

Status	Parameter	Current Value	Recommended Values
Warning	Signal label mismatch (SignalLabelMismatchMsg)	none	error
Warning	Unconnected block input ports (UnconnectedInputMsg)	warning	error
Warning	Unconnected block output ports (UnconnectedOutputMsg)	warning	error
	Unconnected line (UnconnectedLineMsg)	warning	error

Warning			

Recommended Action

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

Check safety-related diagnostic settings for bus connectivity

Check diagnostic settings in the model configuration that apply to bus connectivity and might impact safety.

Warning

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

Status	Parameter	Current Value	Recommended Values
Warning	Unspecified bus object at root Outport block (RootOutportRequireBusObject)	warning	error
Warning	Element name mismatch (BusObjectLabelMismatch)	warning	error
Warning	Mux blocks used to create bus signals (StrictBusMsg)	ErrorLevel1	ErrorOnBusTreatedAsVector

Recommended Action

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

Check safety-related diagnostic settings that apply to function-call connectivity

Check diagnostic settings in the model configuration that apply to function-call connectivity and might impact safety.

Passed

All constraints on model configuration parameters have been met.

	Parameter	Current	Recommended

Status		Value	Values
Pass	Invalid function-call connection (InvalidFcnCallConnMsg)	error	error
Pass	Context-dependent inputs (FcnCallInpInsideContextMsg)	error	error

---

Check safety-related diagnostic settings for compatibility

Check diagnostic settings in the model configuration that affect compatibility and might impact safety

Warning

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

Status	Parameter	Current Value	Recommended Values
Warning	S-function upgrades needed (SFcnCompatibilityMsg)	none	error

Recommended Action

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

Check safety-related diagnostic settings for model initialization

Check whether **All Parameters > Underspecified initialization detection** is set to `Simplified`. If it is set to `Classic` check that the sub-parameters are selected.

Warning

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

Status	Parameter	Current Value	Recommended Values

Warning	Check runtime output of execution context (CheckExecutionContextRuntimeOutputMsg)	off	on

Recommended Action

Set **All Parameters > Underspecified initialization detection** to `Simplified` or select the sub-parameters.

Check safety-related diagnostic settings for model referencing

Check diagnostic settings in the model configuration that apply to model referencing and might impact safety.

Warning

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

Status	Parameter	Current Value	Recommended Values
Warning	Port and parameter mismatch (ModelReferenceIOMismatchMessage)	none	error
Warning	Invalid root Inport/Outport block connection (ModelReferenceIOMsg)	none	error
Warning	Unsupported data logging (ModelReferenceDataLoggingMessage)	warning	error

Recommended Action

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

Check safety-related model referencing settings

Check model referencing settings in the model configuration that might impact safety.

Passed

All constraints on model configuration parameters have been met.

Status	Parameter	Current Value	Recommended Values
Pass	Rebuild (UpdateModelReferenceTargets)	IfOutOfDateOrStructuralChange	IfOutOfDateOrStructuralChange
Pass	Pass fixed-size scalar root inputs by value for code generation (ModelReferencePassRootInputsByReference) *	on	on
Pass	Minimize algebraic loop occurrences (ModelReferenceMinAlgLoopOccurrences)	off	off

Recommended Action

* The Command-Line values provided in the table are reverse of the settings in the Configuration Parameters Dialog. Therefore, 'on' in the Command-Line corresponds to an "Off" setting in the dialog, and 'off' in the Command-Line corresponds to an "On" setting in the dialog.

Check safety-related code generation settings

Check optimization settings in the model configuration that might impact safety.

Warning

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

Status	Parameter	Current Value	Recommended Values	Not Recommended Values	Prerequisites
Warning	Requirements in block comments (ReqsInCode)	off	on		SystemTargetFile
Warning	non-finite numbers (SupportNonFinite)	on	off		SystemTargetFile
	absolute time	on	off		

Warning	(SupportAbsoluteTime)				SystemTargetFile
Warning	continuous time (SupportContinuousTime)	on	off		SystemTargetFile
Warning	Terminate function required (IncludeMdlTerminateFcn)	on	off		SystemTargetFile
Warning	Remove error status field in real-time model data structure (SuppressErrorStatus)	off	on		SystemTargetFile
Warning	Parentheses level (ParenthesesLevel)	Nominal	Maximum		SystemTargetFile
Warning	Preserve operand order in expression (PreserveExpressionOrder)	off	on		SystemTargetFile
Warning	Preserve condition expression in if statement (PreserveIfCondition)	off	on		SystemTargetFile
Warning	Minimum mangle length (MangleLength)	1		1, 2, 3	SystemTargetFile

Recommended Action

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

Check safety-related diagnostic settings for saving

Check diagnostic settings in the model configuration that apply to saving model files.

Warning

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

Status	Parameter	Current Value	Recommended Values
	Block diagram contains disabled library links	warning	error

Warning	(SaveWithDisabledLinksMsg)		
Warning	Block diagram contains parameterized library links (SaveWithParameterizedLinksMsg)	warning	error

Recommended Action

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

Check for blocks that do not link to requirements

Check for blocks that do not link to a requirements document

Warning

The following blocks do not link to a requirements document:

- Quadkopter_Model/Attitude Controller
- Quadkopter_Model/Attitude Controller/Attitude Cmd
- Quadkopter_Model/Attitude Controller/State Input
- Quadkopter_Model/Attitude Controller/Altitude (Z) Control
- Quadkopter_Model/Attitude Controller/Bus Selector
- Quadkopter_Model/Attitude Controller/Pitch (Theta) Control
- Quadkopter_Model/Attitude Controller/Roll (Phi) Control
- Quadkopter_Model/Attitude Controller/Yaw (Psi) Control
- Quadkopter_Model/Attitude Controller/Roll (Phi) Correction
- Quadkopter_Model/Attitude Controller/Pitch (Theta) Correction
- Quadkopter_Model/Attitude Controller/Yaw (Psi) Correction
- Quadkopter_Model/Attitude Controller/Alt. (Z) Correction

∧ Less

Recommended Action

For each block in the list, in the Model Editor, right-click the block, select `Requirements Traceability`, and specify a requirement.

Check state machine type of Stateflow charts

Compares the state machine type of all Stateflow charts to the desired type.

Passed

The model does not contain Stateflow charts. Stateflow Charts linked to libraries are not checked.

Input Parameters Selection

Name	Value
State Machine Type	Mealy or Moore

Check Stateflow charts for ordering of states and transitions

Identify Stateflow charts that have **User specified state/transition execution order** cleared.

Passed

No Stateflow charts were found.

Check Stateflow debugging options

Identify whether the following Stateflow debugging options are cleared: **Detect wrap on overflow** , **Detect Cycles**, and **Simulation range checking**

Passed

No Stateflow charts were found.

Check usage of lookup table blocks

Check for Lookup Table blocks, Prelookup blocks and Interpolation blocks that do not generate out-of-range checking code.

Passed

There are no Lookup Table blocks, Prelookup blocks and Interpolation blocks in the model.

Check for MATLAB Function interfaces with inherited properties

Identify MATLAB Functions that have inputs, outputs, or parameters with inherited complexity or data type properties.

Passed

No MATLAB Functions found in the model or subsystem.

Check MATLAB Function metrics

Identify MATLAB Functions that violate complexity limits.

Passed

No MATLAB Functions were found.

Input Parameters Selection

Name	Value
Maximum effective lines of code per function	60
Minimum density of comments	0.2
Maximum cyclomatic complexity per function	15

Check MATLAB Code Analyzer messages

Check MATLAB code used in MATLAB Function blocks

Passed

No MATLAB Function blocks found

Check MATLAB functions defined in Stateflow charts

Passed

No MATLAB functions defined in Stateflow charts found

Check called MATLAB functions

Passed

No external MATLAB functions found

Check MATLAB code for global variables

Check for global variables in MATLAB code used in MATLAB Function blocks

Passed

No MATLAB Function blocks found

Check for global variables in MATLAB functions defined in Stateflow charts

Passed

No MATLAB functions defined in Stateflow charts found

Check for global variables in called MATLAB functions

Passed

No external MATLAB functions found

Check for inconsistent vector indexing methods

Check consistent usage of vector indexing methods across the model or subsystem

Passed

No blocks using vector indexing were found

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

Identify blocks not supported by code generation or not recommended for C/C++ production code deployment.

Warning

The following blocks are not supported or not recommended for C/C++ production code deployment:

Block	Block Type	Code generation support	Recommendation for C/C++ production code deployment
Quadkopter_Model/Attitude Controller/Altitude (Z) Control/Derivative	Derivative	Yes1′2	No

Quadkopter_Model/Attitude Controller/Altitude (Z) Control/Integrator Limited	Integrator	Yes1′2	No
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control/Integrator Limited	Integrator	Yes1′2	No
Quadkopter_Model/Attitude Controller/Roll (Phi) Control /Integrator Limited	Integrator	Yes1′2	No
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control/Integrator Limited	Integrator	Yes1′2	No

Recommended Action

Although Embedded Coder supports these blocks, they are not recommended for C/C++ production code deployment. Review the support notes for these blocks and follow the given advice.

Support notes:

1. Consider using the Model Discretizer to map these continuous blocks into discrete equivalents that support code generation. From a model, select **Analysis > Control Design > Model Discretizer** to access the Model Discretizer.
2. Not recommended for production code.

Check for variant blocks with 'Generate preprocessor conditionals' active

No variant blocks with "Generate preprocessor conditionals" on were found.

Check Stateflow charts for uniquely defined data objects

Identifies local data identifiers that are defined in multiple scopes within a chart.

Passed

No Stateflow charts were found.

Check usage of Math Operations blocks

Identify usage of Math Operation blocks in Simulink that might impact safety

Check usage of Abs blocks

Identify Absolute Value blocks that might produce unreachable code or overflows

Passed

There are no Abs blocks in the subsystem.

Check usage of Gain blocks

Identify Gain blocks whose value equals 1

Passed

No Gain blocks with value equal to 1 in the model or subsystem

---

Check usage of Math Function - Natural logarithm blocks

Identify Math Function - Natural logarithm blocks that might result in non-finite output signals.

Passed

No Math Function - Natural logarithm blocks found.

Check usage of Math Function - Common (base 10) logarithm blocks

Identify Math Function - Common (base 10) logarithm blocks that might result in non-finite output signals.

Passed

No Math Function - Common (base 10) logarithm blocks found.

Check usage of Math Function - Remainder after division blocks

Identify Math Function - Remainder after division blocks that might result in non-finite output signals.

Passed

No Math Function - Remainder after division blocks found.

Check usage of Math Function - Reciprocal blocks

Identify Math Function - Reciprocal blocks that might result into non-finite signals at their outputs.

Passed

No Math Function - Reciprocal blocks found.

Check Usage of Assignment blocks

Identify Assignment blocks with possibly incomplete array initialization that do not have the simulation run-time diagnostic **Action if any output element is not assigned** set to:

- Warning, if Assignment block is in an iterator subsystem
- Error, if Assignment block is not in an iterator subsystem

Passed

All Assignment blocks are configured with block parameter **Action if any output element is not assigned** set to Warning or Error.

Check usage of Signal Routing blocks

Check Switch blocks

Identify Switch blocks that might generate code with inequality operations (~=) in expressions where at least one side of the expression is a floating-point variable or constant.

Passed

There are no Switch blocks in the model or subsystem.

Check usage of Logic and Bit Operations blocks

Identify usage of Logic and Bit Operations blocks in Simulink that might impact safety.

Check blocks that compute relational operators

Identify blocks computing a relational operator that is operating on different data types or equating floating-point types.

Passed

There are no blocks that compute relational operators in the subsystem.

Check Logic blocks

Identify Logical Operator blocks that have non-boolean inputs or outputs.

Passed

There are no Logic blocks in this model or subsystem.

Check usage of Ports and Subsystems blocks

Identify usage of Ports and Subsystems blocks in Simulink that might impact safety.

Check While Iterator blocks

Identify While Iterator blocks that might cause infinite loops.

Passed

There are no While Iterator blocks in the subsystem.

Check For Iterator blocks

Identify For Iterator blocks that cause variable loops.

Passed

There are no For Iterator blocks in this subsystem.

Check If blocks

Identify If blocks with If expressions or Elseif expressions that might cause floating point equality or inequality comparisons in generated code.
Identify If blocks using Elseif expressions without having an Else condition.
Identify If blocks with output ports that are not connected to If Action Subsystem blocks.

Passed

There are no If blocks in this model or subsystem.

Check Switch Case blocks

Identify Switch Case blocks that do not have a default case.
Identify Switch Case blocks with output ports that are not connected to Switch Case Action Subsystem blocks.

Passed

No Switch Case blocks were found in the model or subsystem.

Check sample time-dependent blocks

Identify sample time-dependent blocks in While and For Iterator subsystems.

Passed

No For or While Iterator subsystems found.

Check model object names

Identify invalid names of following model objects (first invalid name fragment is highlighted):

- Blocks
- Signals
- Parameters
- Busses
- Stateflow elements

Warning

The following model objects have invalid names:

Block	Name
Quadkopter_Model/Attitude Controller	AttitudeController
Quadkopter_Model/Attitude Controller/Attitude Cmd	AttitudeCmd
Quadkopter_Model/Attitude Controller/State Input	StateInput
Quadkopter_Model/Attitude Controller/Altitude (Z) Control	Altitude(Z) Control
Quadkopter_Model/Attitude Controller/Altitude (Z) Control/Desired Z	DesiredZ
Quadkopter_Model/Attitude Controller/Altitude (Z) Control/Z State	ZState
Quadkopter_Model/Attitude Controller/Altitude (Z) Control/Bus Selector	BusSelector
Quadkopter_Model/Attitude Controller/Altitude (Z) Control/Gravity Offset (Level Flight)	GravityOffset (Level Flight)
Quadkopter_Model/Attitude Controller/Altitude (Z) Control/Integrator Limited	IntegratorLimited
Quadkopter_Model/Attitude Controller/Altitude (Z) Control/Alt. (Z) Correction	Alt (Z) Correction
Quadkopter_Model/Attitude Controller/Bus Selector	BusSelector
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control	Pitch(Theta) Control
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control/Attitude Command	AttitudeCommand

Quadkopter_Model/Attitude Controller/Pitch (Theta) Control/Theta State	ThetaState
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control/Q State	QState
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control/Bus Selector	BusSelector
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control/Integrator Limited	IntegratorLimited
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control/Theta Correction	ThetaCorrection
Quadkopter_Model/Attitude Controller/Roll (Phi) Control	Roll(Phi) Control
Quadkopter_Model/Attitude Controller/Roll (Phi) Control /Attitude Command	AttitudeCommand
Quadkopter_Model/Attitude Controller/Roll (Phi) Control /Phi State	PhiState
Quadkopter_Model/Attitude Controller/Roll (Phi) Control /Bus Selector	BusSelector
Quadkopter_Model/Attitude Controller/Roll (Phi) Control /Integrator Limited	IntegratorLimited
Quadkopter_Model/Attitude Controller/Roll (Phi) Control /Phi Correction	PhiCorrection
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control	Yaw(Psi) Control
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control/Attitude Command	AttitudeCommand
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control/Psi State	PsiState
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control/R State	RState
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control/Bus Selector	BusSelector
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control/Integrator Limited	IntegratorLimited
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control/Psi Correction	PsiCorrection
Quadkopter_Model/Attitude Controller/Roll (Phi) Correction	Roll(Phi) Correction
Quadkopter_Model/Attitude Controller/Pitch (Theta) Correction	Pitch(Theta) Correction
Quadkopter_Model/Attitude Controller/Yaw (Psi) Correction	Yaw(Psi) Correction

Quadkopter_Model/Attitude Controller/Alt. (Z) Correction	Alt (Z) Correction

∧ Less

Signal	**Name**
Quadkopter_Model/Attitude Controller/Altitude (Z) Control	DCorr
Quadkopter_Model/Attitude Controller/Altitude (Z) Control	ICorr
Quadkopter_Model/Attitude Controller/Altitude (Z) Control	PCorr
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control	DCorr
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control	ICorr
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control	PCorr
Quadkopter_Model/Attitude Controller/Roll (Phi) Control	DCorr
Quadkopter_Model/Attitude Controller/Roll (Phi) Control	ICorr
Quadkopter_Model/Attitude Controller/Roll (Phi) Control	PCorr
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control	DCorr
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control	ICorr
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control	PCorr

∧ Less

Recommended Action

Change flagged names of model objects

Display model version information

Display model configuration and checksum information.

NOTE: These checks were performed on a subsystem, but are based on root-level settings.

Model configuration and checksum information

Attribute	Value
Model Version	1.20
Author	magogs00
Date	Mon Apr 10 14:40:33 2017
Model Checksum	1615430392 1435335912 2888209078 286490704

IEC 61508, IEC 62304, ISO 26262, and EN 50128 Checks 14 0 6 0

Check state machine type of Stateflow charts

Compares the state machine type of all Stateflow charts to the desired type.

Passed

The model does not contain Stateflow charts. Stateflow Charts linked to libraries are not checked.

Input Parameters Selection

Name	Value
State Machine Type	Mealy or Moore

Check for inconsistent vector indexing methods

Check consistent usage of vector indexing methods across the model or subsystem

Passed

No blocks using vector indexing were found

Check for MATLAB Function interfaces with inherited properties

Identify MATLAB Functions that have inputs, outputs, or parameters with inherited complexity or data type properties.

Passed
No MATLAB Function found in the model or subsystem.

Check MATLAB Function metrics

Identify MATLAB Functions that violate complexity limits.

Passed
No MATLAB Functions were found.

Input Parameters Selection

Name	Value
Maximum effective lines of code per function	60
Minimum density of comments	0.2
Maximum cyclomatic complexity per function	15

Check MATLAB Code Analyzer messages

Check MATLAB code used in MATLAB Function blocks

Passed
No MATLAB Function blocks found

Check MATLAB functions defined in Stateflow charts

Passed
No MATLAB functions defined in Stateflow charts found

Check called MATLAB functions

Passed
No external MATLAB functions found

Check MATLAB code for global variables

Check for global variables in MATLAB code used in MATLAB Function blocks

Passed
No MATLAB Function blocks found

Check for global variables in MATLAB functions defined in Stateflow charts

Passed
No MATLAB functions defined in Stateflow charts found

Check for global variables in called MATLAB functions

Passed
No external MATLAB functions found

Display model metrics and complexity report

Display number of elements and name, level, and depth of subsystems for the model or subsystem

Model metrics information
Display number of elements for Simulink blocks and Stateflow constructs

Summary

Element Type	Count
Inport	13
Outport	8
SubSystem	4

Simulink

Block Type	Count
Inport	13
Gain	12
Sum	8
Outport	8
BusSelector	5
SubSystem	4
Integrator	4
Derivative	1
Constant	1

Model complexity information

Display name, level, and depth of subsystems

Maximum Subsystem Depth: 2

Subsystem Depth

Subsystem Name	Level	Depth
Quadkopter_Model/Attitude Controller/Altitude (Z) Control	1	1
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control	1	1
Quadkopter_Model/Attitude Controller/Roll (Phi) Control	1	1
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control	1	1

Check for unconnected objects

Identify unconnected lines, input ports, and output ports in the subsystem

Passed

There are no unconnected lines, input ports, and output ports in this subsystem.

Check for root Inports with missing properties

Identify Inport blocks in the top-level of the model with missing or inherited sample times, data types, or port dimensions. Inport block properties are specified with block parameters or Simulink signal data objects that explicitly resolve to the connected signal lines.

Warning

This check is only supported at the model level.

Recommended Action

To run this analysis, please open the model advisor from the top level of the model instead of the subsystem level and start the analysis.

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

Identify blocks not supported by code generation or not recommended for C/C++ production code deployment.

Warning

The following blocks are not supported or not recommended for C/C++ production code deployment:

Block	**Block Type**	**Code generation support**	**Recommendation for C/C++ production code deployment**
Quadkopter_Model/Attitude Controller/Altitude (Z) Control/Derivative	Derivative	Yes1′2	No
Quadkopter_Model/Attitude Controller/Altitude (Z) Control/Integrator Limited	Integrator	Yes1′2	No
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control/Integrator Limited	Integrator	Yes1′2	No
Quadkopter_Model/Attitude Controller/Roll (Phi) Control /Integrator Limited	Integrator	Yes1′2	No

Quadkopter_Model/Attitude Controller/Yaw (Psi) Control/Integrator Limited	Integrator	Yes1′2	No

Recommended Action

Although Embedded Coder supports these blocks, they are not recommended for C/C++ production code deployment. Review the support notes for these blocks and follow the given advice.

Support notes:

1. Consider using the Model Discretizer to map these continuous blocks into discrete equivalents that support code generation. From a model, select **Analysis > Control Design > Model Discretizer** to access the Model Discretizer.
2. Not recommended for production code.

Check for model objects that do not link to requirements

Check Simulink blocks and Stateflow objects that do not link to a requirements document

Warning

The following blocks do not link to a requirement document:

- Quadkopter_Model/Attitude Controller
- Quadkopter_Model/Attitude Controller/Attitude Cmd
- Quadkopter_Model/Attitude Controller/State Input
- Quadkopter_Model/Attitude Controller/Altitude (Z) Control
- Quadkopter_Model/Attitude Controller/Bus Selector
- Quadkopter_Model/Attitude Controller/Pitch (Theta) Control
- Quadkopter_Model/Attitude Controller/Roll (Phi) Control
- Quadkopter_Model/Attitude Controller/Yaw (Psi) Control
- Quadkopter_Model/Attitude Controller/Roll (Phi) Correction
- Quadkopter_Model/Attitude Controller/Pitch (Theta) Correction
- Quadkopter_Model/Attitude Controller/Yaw (Psi) Correction

- Quadkopter_Model/Attitude Controller/Alt. (Z) Correction

Λ Less

Recommended Action

For each object in the list, in the Model Editor, right-click the block, select `Requirements Traceability`, and specify a requirement.

Check usage of Math Operations blocks

Identify usage of Math Operation blocks in Simulink that might impact safety

Check usage of Abs blocks

Identify Absolute Value blocks that might produce unreachable code or overflows

Passed

There are no Abs blocks in the subsystem.

Check usage of Gain blocks

Identify Gain blocks whose value equals 1

Passed

No Gain blocks with value equal to 1 in the model or subsystem

---

Check Usage of Assignment blocks

Identify Assignment blocks with possibly incomplete array initialization that do not have the simulation run-time diagnostic **Action if any output element is not assigned** set to:

- Warning, if Assignment block is in an iterator subsystem
- Error, if Assignment block is not in an iterator subsystem

Passed

All Assignment blocks are configured with block parameter **Action if any output element is not assigned** set to Warning or Error.

Check usage of Signal Routing blocks

Check Switch blocks

Identify Switch blocks that might generate code with inequality operations (~=) in expressions where at least one side of the expression is a floating-point variable or constant.

Passed

There are no Switch blocks in the model or subsystem.

Check usage of Logic and Bit Operations blocks

Identify usage of Logic and Bit Operations blocks in Simulink that might impact safety.

Check blocks that compute relational operators

Identify blocks computing a relational operator that is operating on different data types or equating floating-point types

Passed

There are no blocks that compute relational operators in this subsystem

---

Check Logic blocks

Identify Logical Operator blocks that have non-boolean inputs or outputs.

Passed

There are no Logic blocks in this model or subsystem.

Check usage of Ports and Subsystems blocks

Identify usage of Ports and Subsystems blocks in Simulink that might impact safety.

Check While Iterator blocks

Identify While Iterator blocks that might cause infinite loops

Passed

There are no While Iterator blocks in this subsystem

Check For Iterator blocks

Identify For Iterator blocks that cause variable loops

Passed

There are no For Iterator blocks in this subsystem

Check If blocks

Identify If blocks with If expressions or Elseif expressions that might cause floating point equality or inequality comparisons in generated code.
Identify If blocks using Elseif expressions without having an Else condition.
Identify If blocks with output ports that are not connected to If Action Subsystem blocks.

Passed

There are no If blocks in this model or subsystem.

Check Switch Case blocks

Identify Switch Case blocks that do not have a default case.
Identify Switch Case blocks with output ports that are not connected to Switch Case Action Subsystem blocks.

Passed

No Switch Case blocks were found in the model or subsystem.

Check sample time-dependent blocks

Identify sample time-dependent blocks in While and For Iterator subsystems.

Passed

No For or While Iterator subsystems found.

Check for root Inports with missing range definitions

Identify root-level Inport blocks with missing or erroneous minimum or maximum values. Inport block minimum and maximum values are specified with block parameters or Simulink signal objects that explicitly resolve to the connected signal lines.

Warning

This check is only supported at the model level.

Recommended Action

To run this analysis, please open the model advisor from the top level of the model instead of the subsystem level and start the analysis.

Check for root Outports with missing range definitions

Identify root-level Outport blocks with missing or erroneous minimum or maximum values. Outport block minimum and maximum values are specified with block parameters or Simulink signal objects that explicitly resolve to the connected signal lines.

Warning

This check is only supported at the model level.

Recommended Action

To run this analysis, please open the model advisor from the top level of the model instead of the subsystem level and start the analysis.

Check model object names

Identify invalid names of following model objects (first invalid name fragment is highlighted):

- Blocks
- Signals
- Parameters
- Busses
- Stateflow elements

Warning

The following model objects have invalid names:

Block	Name
Quadkopter_Model/Attitude Controller	AttitudeController
Quadkopter_Model/Attitude Controller/Attitude Cmd	AttitudeCmd
Quadkopter_Model/Attitude Controller/State Input	StateInput
Quadkopter_Model/Attitude Controller/Altitude (Z) Control	Altitude(Z) Control
Quadkopter_Model/Attitude Controller/Altitude (Z) Control/Desired Z	DesiredZ
Quadkopter_Model/Attitude Controller/Altitude (Z) Control/Z State	ZState
Quadkopter_Model/Attitude Controller/Altitude (Z) Control/Bus Selector	BusSelector
Quadkopter_Model/Attitude Controller/Altitude (Z) Control/Gravity Offset (Level Flight)	GravityOffset (Level Flight)
Quadkopter_Model/Attitude Controller/Altitude (Z) Control/Integrator Limited	IntegratorLimited
Quadkopter_Model/Attitude Controller/Altitude (Z) Control/Alt. (Z) Correction	Alt (Z) Correction
Quadkopter_Model/Attitude Controller/Bus Selector	BusSelector

Quadkopter_Model/Attitude Controller/Pitch (Theta) Control	Pitch(Theta) Control
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control/Attitude Command	AttitudeCommand
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control/Theta State	ThetaState
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control/Q State	QState
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control/Bus Selector	BusSelector
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control/Integrator Limited	IntegratorLimited
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control/Theta Correction	ThetaCorrection
Quadkopter_Model/Attitude Controller/Roll (Phi) Control	Roll(Phi) Control
Quadkopter_Model/Attitude Controller/Roll (Phi) Control /Attitude Command	AttitudeCommand
Quadkopter_Model/Attitude Controller/Roll (Phi) Control /Phi State	PhiState
Quadkopter_Model/Attitude Controller/Roll (Phi) Control /Bus Selector	BusSelector
Quadkopter_Model/Attitude Controller/Roll (Phi) Control /Integrator Limited	IntegratorLimited
Quadkopter_Model/Attitude Controller/Roll (Phi) Control /Phi Correction	PhiCorrection
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control	Yaw(Psi) Control
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control/Attitude Command	AttitudeCommand
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control/Psi State	PsiState
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control/R State	RState
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control/Bus Selector	BusSelector
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control/Integrator Limited	IntegratorLimited
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control/Psi Correction	PsiCorrection

Quadkopter_Model/Attitude Controller/Roll (Phi) Correction	Roll(Phi) Correction
Quadkopter_Model/Attitude Controller/Pitch (Theta) Correction	Pitch(Theta) Correction
Quadkopter_Model/Attitude Controller/Yaw (Psi) Correction	Yaw(Psi) Correction
Quadkopter_Model/Attitude Controller/Alt. (Z) Correction	Alt (Z) Correction

Λ Less

Signal	**Name**
Quadkopter_Model/Attitude Controller/Altitude (Z) Control	DCorr
Quadkopter_Model/Attitude Controller/Altitude (Z) Control	ICorr
Quadkopter_Model/Attitude Controller/Altitude (Z) Control	PCorr
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control	DCorr
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control	ICorr
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control	PCorr
Quadkopter_Model/Attitude Controller/Roll (Phi) Control	DCorr
Quadkopter_Model/Attitude Controller/Roll (Phi) Control	ICorr
Quadkopter_Model/Attitude Controller/Roll (Phi) Control	PCorr
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control	DCorr
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control	ICorr
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control	PCorr

Λ Less

Recommended Action

Change flagged names of model objects

Display configuration management data

Display model configuration and checksum information

NOTE: These checks were performed on a sub-system, but are based on root-level settings.

Model configuration and checksum information

Attribute	Value
Model Version	1.20
Author	magogs00
Date	Mon Apr 10 14:40:33 2017
Model Checksum	1615430392 1435335912 2888209078 286490704

Check usage of Stateflow constructs

Identify usage of Stateflow constructs that might impact safety

Check for Strong Data Typing with Simulink I/O

Verify configuration settings for strong data typing on the boundaries between Simulink and Stateflow constructs

Passed

No Stateflow charts were found.

Check Stateflow port naming

Identify mismatches between names of Stateflow ports and associated signals

Passed

No Stateflow charts were found.

NOTE: The following checks were performed on a subsystem, but are based on root-level settings.

Check Stateflow data object scoping

Identify Stateflow data objects with local scope that are not scoped at the chart level or below

Passed

No Stateflow charts were found.

Check formatting of Stateflow state actions

Identify missing line breaks between entry action (en), during action (du), and exit action (ex) entries in states. Identify missing line breaks after semicolons (;) in statements.

Passed

No Stateflow charts were found.

Check Stateflow charts for ordering of states and transitions

Identify Stateflow charts that have **User specified state/transition execution order** cleared.

Passed

No Stateflow charts were found.

Check Stateflow debugging options

Identify whether the following Stateflow debugging options are cleared: **Detect wrap on overflow** , **Detect Cycles**, and **Simulation range checking**

Passed

No Stateflow charts were found.

Check Stateflow charts for uniquely defined data objects

Identifies local data identifiers that are defined in multiple scopes within a chart.

Passed

No Stateflow charts were found.

MathWorks Automotive Advisory Board Checks 44 0 8 0

Check for indexing in blocks

Identify blocks with inconsistent indexing. To increase code efficiency, use zero-based indexing.

Passed

All blocks in the system use a consistent indexing method.

Check for prohibited blocks in discrete controllers

Identify blocks that are not allowed in discrete controllers. Prohibited blocks include all continuous blocks and some source and sink blocks.

Warning

The following discrete controllers contain prohibited blocks:

- Quadkopter_Model/Attitude Controller/Altitude (Z) Control/Derivative
- Quadkopter_Model/Attitude Controller/Altitude (Z) Control/Integrator Limited
- Quadkopter_Model/Attitude Controller/Pitch (Theta) Control/Integrator Limited
- Quadkopter_Model/Attitude Controller/Roll (Phi) Control /Integrator Limited
- Quadkopter_Model/Attitude Controller/Yaw (Psi) Control/Integrator Limited

Recommended Action

Replace the above blocks with equivalent discrete blocks.

Check for prohibited sink blocks

Identify sink blocks that must be removed prior to code generation.

Passed

There are no prohibited blocks in the subsystem.

Check whether block names appear below blocks

Identify blocks where the name is not displayed below the block.

Passed

All blocks have names displayed below the block.

Check for mixing basic blocks and subsystems

Identify levels in the model that include basic blocks and subsystems. Each level of a model must be designed with blocks of the same level (for example, only subsystems or only basic blocks).

Passed

The model does not mix basic blocks and subsystems at the same level.

Check display for port blocks

Identify Inport and Outport blocks that do not specify `Port number` for the **Icon display** block parameter.

Passed

All port blocks display the port number.

Check subsystem names

Identify subsystem names that use characters that are not correct in C code.

Passed

All subsystem names use correct characters.

Check port block names

Identify names of Inport or Outport blocks that use characters that are not correct in C code.

Warning

The following Inport and Outport block names contain incorrect characters:

Error	Incorrect port block
Name contains incorrect characters.	Quadkopter_Model/Attitude Controller/Attitude Cmd
Name contains incorrect characters.	Quadkopter_Model/Attitude Controller/State Input
Name contains incorrect characters.	Quadkopter_Model/Attitude Controller/Roll (Phi) Correction
Name contains incorrect characters.	Quadkopter_Model/Attitude Controller/Pitch (Theta) Correction
Name contains incorrect characters.	Quadkopter_Model/Attitude Controller/Yaw (Psi) Correction
Name contains incorrect characters.	Quadkopter_Model/Attitude Controller/Alt. (Z) Correction

Recommended Action

Rename the Inport or Outport blocks to use correct characters.

Check character usage in block names

Identify block names that use characters that are not correct in C code.

Passed

All blocks use correct characters.

Check orientation of Subsystem blocks

Check for subsystems that do not have the correct orientation. Blocks with the correct orientation have inputs on the left and outputs on the right.

Passed

All subsystem blocks use the correct orientation.

Check font formatting

Identify inconsistent formatting of text.

Warning

Font formatting is not consistent.

The following font characteristics are used in the model/subsystem. Font characteristics are sorted by number of occurrences. The most common characteristics are bold.

Font Name	Font Size	Font Style
Helvetica	**10** 9	**normal**

Recommended Action

To have consistent font formatting, click **Modify All Fonts** to apply the font formatting selected in the input parameters above to all objects.

Input Parameters Selection

Name	Value
Font Name	Common
Font Size	Common
Font Style	Common

Check transition orientations in flow charts

Identify transitions in Stateflow flow charts that are drawn incorrectly.

Check for conditions drawn vertically

Condition expressions should be drawn on the horizontal segments of flow charts.

Passed

All condition expressions were drawn horizontally.

Check for action transitions drawn vertically

Transition actions should be drawn on the vertical segments of flow charts.

Passed

All transition actions were drawn vertically.

Check for junctions for default transitions

All Junctions in a flow chart should have a default exit transition.

Passed

All Junctions have a default exit transition.

Check for transitions that combine condition and action

Flow charts should not combine condition evaluations and action expressions in a single transition.

Passed

No combined expressions were found in the chart.

Check for nondefault block attributes

Identify blocks that use and fail to display nondefault values.

Passed

Model displays all block parameter values that are not default values.

Check signal line labels

Identify blocks that require labeled signals. A subset of source and destination blocks require labeled signals.

Check source block labels

The following source blocks require labeled signals; Inport, From, Data Store Read, Constant, Bus Selector, Demux, Selector. If the signal name is visible on the block, this rule is considered met.

Warning

The following signals have no label:

- Quadkopter_Model/Attitude Controller
- Quadkopter_Model/Attitude Controller
- Quadkopter_Model
- Quadkopter_Model
- Quadkopter_Model
- Quadkopter_Model
- Quadkopter_Model/Attitude Controller
- Quadkopter_Model/Attitude Controller
- Quadkopter_Model/Attitude Controller
- Quadkopter_Model/Attitude Controller

Recommended Action

Add a new or propagated label to the signal line.

Check destination block labels

The following destination blocks require labeled signals; Outport, Goto, Data Store Write, Bus Creator, Mux, Subsystem, Chart. If the signal name is visible on the source block, this rule is considered met.

Warning

The following signals have no label:

- Quadkopter_Model/Attitude Controller/Roll (Phi) Correction/
- Quadkopter_Model/Attitude Controller/Pitch (Theta) Correction/
- Quadkopter_Model/Attitude Controller/Yaw (Psi) Correction/
- Quadkopter_Model/Attitude Controller/Alt. (Z) Correction/
- Quadkopter_Model/Attitude Controller/
- Quadkopter_Model/Attitude Controller/
- Quadkopter_Model/Attitude Controller/Altitude (Z) Control/

- Quadkopter_Model/Attitude Controller/Pitch (Theta) Control/
- Quadkopter_Model/Attitude Controller/Roll (Phi) Control /
- Quadkopter_Model/Attitude Controller/Yaw (Psi) Control/

Recommended Action

Add a new or propagated label to the signal line.

Check for propagated signal labels

Identify propagated labels on signal lines.

Check subsystem input labels

Identify subsystem inputs that are labeled and display propagated signals.

Passed

All inputs to the subsystem have labels and display propagated signals.

Check subsystem output labels

Identify outputs from subsystems that are labeled and display signal propagation.

Passed

All outputs from the subsystem have labels and display propagated signals.

Signal propagation for nonsubsystem blocks

Identify the signal propagation status for both transformative and nontransformative blocks.

Passed

All outputs from non subsystem blocks correctly use labels and display propagated signals.

Check default transition placement in Stateflow charts

Identify all groupings of states that do not have a default transition or do not have the default state as the top-most state.

Passed

No Stateflow charts were found.

Check return value assignments of graphical functions in Stateflow charts

Identify graphical functions with multiple assignments of return values in Stateflow charts.

Passed

No Stateflow charts were found.

Check entry formatting in State blocks in Stateflow charts

Identify missing line breaks between entry action (en), during action (du), and exit action (ex) entries in states. Identify missing line breaks after semicolons (;) in statements.

Passed

No Stateflow charts were found.

Check usage of return values from a graphical function in Stateflow charts

Identify calls to graphical functions that are used in conditional expressions.

Passed

No conditional expressions containing calls to graphical functions were found.

Check for pointers in Stateflow charts

Identify pointer operations on custom code variables.

Note: This check applies only to Stateflow charts that use C as the action language.

Passed

No pointer operations were found.

Check for event broadcasts in Stateflow charts

Identify undirected event broadcasts that might cause recursion during simulation and generate inefficient code.

Passed

No instances of undirected event broadcast were found.

Check transition actions in Stateflow charts

Identify missing line breaks between transition actions.

Passed

All transition actions are formatted correctly.

Check for MATLAB expressions in Stateflow charts

Identify Stateflow objects that use MATLAB expressions that are not suitable for code generation.

Note: This check applies only to Stateflow charts that use C as the action language.

Passed

No Stateflow objects that use MATLAB expressions were found.

Check for Simulink diagrams using nonstandard display attributes

Identify nonstandard display attributes in Simulink diagrams.

Check format settings

Identify incorrect model-level format options.

Warning

NOTE: The following subchecks were performed on a subsystem, but are based on root-level settings.The following format display options are incorrect.

Display Attribute	Recommended Value	Actual Value
Display > Signals & Ports > Wide Nonscalar Lines	on	off
View > Model Browser > Show Model Browser	off	on
Display > Library Links	none	all

Recommended Action

Set the format options to the recommended value.

Check block colors

Identify blocks using nonstandard colors.

Warning

The following blocks use nonstandard colors:

- Quadkopter_Model/Attitude Controller
- Quadkopter_Model/Attitude Controller/Altitude (Z) Control
- Quadkopter_Model/Attitude Controller/Pitch (Theta) Control
- Quadkopter_Model/Attitude Controller/Roll (Phi) Control
- Quadkopter_Model/Attitude Controller/Yaw (Psi) Control

Recommended Action
Set the block foreground color to black and the background color to white.

Check canvas colors
Identify canvases that are not white.

Passed
All diagrams use a white canvas.

Check diagram zoom
Identify diagrams that do not have zoom factor set to 100 %.
Note: Zoom factors can differ for each instance of a model diagram opened in Simulink Editor

Warning
The following diagrams do not have zoom factor set to 100 percent:

- Quadkopter_Model/Attitude Controller
- Quadkopter_Model/Attitude Controller/Altitude (Z) Control
- Quadkopter_Model/Attitude Controller/Pitch (Theta) Control
- Quadkopter_Model/Attitude Controller/Roll (Phi) Control
- Quadkopter_Model/Attitude Controller/Yaw (Psi) Control

Recommended Action
For each listed diagram, select **View > Zoom > Normal View (100%)**.

Check input and output settings of MATLAB Functions

Identify MATLAB Functions that have inputs, outputs, or parameters with inherited complexity, data type, or size properties.

Passed
No MATLAB Functions found in the model or subsystem.

Check MATLAB Function metrics

Identify MATLAB Functions that violate complexity limits.

Passed

No MATLAB Functions were found.

Input Parameters Selection

Name	Value
Maximum effective lines of code per function	60
Minimum density of comments	0.2
Maximum cyclomatic complexity per function	15

Check MATLAB code for global variables

Check for global variables in MATLAB code

Check for global variables in MATLAB code used in MATLAB Function blocks

Passed

No MATLAB Function blocks found

Check for global variables in MATLAB functions defined in Stateflow charts

Passed

No MATLAB functions defined in Stateflow charts found

Check for global variables in called MATLAB functions

Passed

No external MATLAB functions found

Check file names

Identify file names with incorrect characters or formatting.

Warning

The following files have incorrect file names:

File Name	Incorrect Character or Format
GPSModel.mdl.r2016b	File name contains more than one dot (.).
Quadkopter_Model.slx.r2013b	File name contains more than one dot (.).
matlab(RPM_Throttle_Secs).mat	File name contains incorrect characters. Correct characters are a-z, A-Z, 0-9, and underscore (_).
quadModel_+.mat	File name contains incorrect characters. Correct characters are a-z, A-Z, 0-9, and underscore (_).

Recommended Action

Rename the above files to remove incorrect characters and formatting.

Check folder names

Identify folders using incorrect characters and formatting.

Passed

All folder names have correct formatting.

Check positioning and configuration of ports

Identify input and output ports with incorrect positioning and configurations.

Check Inport blocks position

Identify Inport blocks that result in left-flowing signals.

Passed

There are no Inport blocks in the model that result in left-flowing signals.

Check Outport block position

Identify Outport blocks that result in left-flowing signals.

Passed

There are no Outport blocks in the model that result in left-flowing signals.

Check port orientation

Identify port blocks with nondefault orientation.

Passed

All ports use the default orientation.

Check for duplicate Inports blocks

Identify duplicate Inport blocks.

Passed

All Inport blocks in the model are used once.

Check for matching port and signal names

Identify mismatches between names of ports and corresponding signals.

Passed

All signal and port names match.

Check for unconnected ports and signal lines

Identify unconnected block input ports, output ports, and signal lines.

Passed

All lines and ports in the model are connected.

Check position of Trigger and Enable blocks

Identify Trigger and Enable blocks that are not located at the top of the subsystem diagram.

Passed

The model does not contain Trigger and Enable blocks, or the blocks are located at the top of the subsystem diagram.

Check usage of tunable parameters in blocks

Identify tunable parameters used to specify expressions, data type conversions, or indexing operations.

Passed

Tunable parameters are not used in the model.

Check Stateflow data objects with local scope

Identify Stateflow data objects with local scope that are not scoped at the chart level or below.

Passed

No Stateflow charts were found.

Check for Strong Data Typing with Simulink I/O

Identify whether Stateflow charts have **Use Strong Data Typing with Simulink I/O** cleared.

Passed

No Stateflow charts were found.

Check usage of exclusive and default states in state machines

Identify Stateflow charts and substates that incorrectly use or define exclusive and default states.

Check Stateflow charts for exclusive states

Identify Stateflow charts that have singular exclusive (OR) states.

Passed

The Stateflow charts do not have singular exclusive (OR) states.

Check Stateflow charts for undefined default states

Identify Stateflow charts that do not define default states.

Passed

Each Stateflow chart defines a default state.

Check for multiple states assigned as the default state
At the root level in the Stateflow hierarchy only one state should be assigned as the default.

Passed
The root level of the chart has only one default state assigned.

Check for substates with singular OR states
States configured as OR should always be part of a group of states.

Passed
No singular OR states were detected.

Check for substates without default states defined
At every level in the Stateflow hierarchy a default state should be assigned.

Passed
All substates have default states assigned.

Check for substates with multiple default states defined
At every level in the Stateflow hierarchy only one state should be assigned as the default.

Passed
All levels of the chart have only one default state assigned.

Check Implement logic signals as Boolean data (vs. double)
Identify whether **Implement logic signals as Boolean data (vs. double)** is selected.

Passed
Implement logic signals as Boolean data (vs. double) is selected.

Check model diagnostic parameters
Identify diagnostic parameters that are set to `none`.

Warning
The following diagnostics parameters are set to `none`:

- Inf or NaN block output
- Duplicate data store names

Recommended Action

In the Configuration Parameters dialog box, set the above diagnostic parameters to `warning` or `error`.

Check the display attributes of block names

Identify whether to display block names.

Check for blocks with hidden names and obvious function

Identify block names that are displayed but can be hidden due to obvious behavior.

Passed

All blocks with obvious behavior have hidden names.

Check for non-descriptive displayed block names

Identify block names that are displayed but should be hidden due to a lack of a descriptive name.

Passed

All displayed names provide descriptive information.

Check for missing block names

Identify block names that are hidden but should be displayed to show a descriptive name.

Passed

All displayed names provide descriptive information.

Check character usage in signal labels

Identify signal labels that are not correct for C variable names.

Passed

All signal labels use correct characters.

Check Trigger and Enable block names

Match the name of the Trigger and Enable blocks to the name of the signal that triggers the subsystem. If the signal is not named, use any name for the block.

Check Trigger block names
Identify Trigger blocks that do not match the names of the signals to which they are connected.

Passed
All Trigger blocks in the system are correctly named.

Check for root level Trigger block
Identify Trigger blocks placed at the root level of the model.

Passed
No Trigger blocks where found at the root level.

Check Enable block names
Identify Enable blocks that do not match the names of the signals to which they are connected.

Passed
All Enable blocks in the system are correctly named.

Check for root level Enable block
Identify Enable blocks placed at the root level of the model.

Passed
No Enable blocks where found at the root level.

Check visibility of block port names
Identify port block names that are not uniformly displayed. The block names must all be displayed or none displayed. Library blocks are an exception to this rule. This check ignores masked and subsystem blocks.

Check for incorrect port name display
Identify ports that are incorrectly displaying names.

Passed
Subsystem blocks are correctly displayed.

Check for incorrect subsystem port name display
Identify subsystems that are incorrectly displaying names.

Passed

Subsystem blocks are correctly displayed.

Input Parameters Selection

Name	Value
Display all port names (Diagram > Format > Show Block Name).	true

---

Check usage of Relational Operator blocks

Identify Relational Operator blocks that connect to constants with the first (upper) input value.

Passed

This model does not contain Relational Operator blocks.

Check usage of Switch blocks

Identify Switch blocks that do not use Boolean inputs for the switch condition (input 2), and do not use `u2 ~= 0` for the **Criteria for passing first input** block parameter.

Check Switch block parameters

Identify Switch blocks with the parameter **Criteria for passing first input** not set to `u2 ~= 0`.

Passed

The block parameter **Criteria for passing first input** is correctly configured.

Check for Boolean switch condition

Identify blocks that do not use Boolean signal switch conditions (input 2).

Passed

The switch condition is a Boolean signal.

Check usage of buses and Mux blocks

Check usage of buses and Mux blocks

Mux blocks used to create bus signals

Identify Mux blocks in the model that are creating bus signals. For guidelines about changes that you might need to make to models, callbacks, scripts, or tests after running the check, see Compatibility

Issues.

Passed

The model uses Mux blocks properly. Model is configured to detect future changes that might result in improper Mux block usage.

Bus signal treated as vector

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

Warning

The 'Bus Usage' check works only from top-level models.

Recommended Action

Run the Model Advisor from the top-level model to perform this check.

Check for bitwise operations in Stateflow charts

Identify bitwise operators (&, |, and ^) in Stateflow charts. If **Enable C-bit Operations** is selected for a chart, only bitwise operators in expressions containing Boolean data types are reported. Otherwise, all bitwise operators are reported for the chart.

Note: This check applies only to Stateflow charts that use C as the action language.

Identify bitwise operators in charts with 'Enable C-bit Operations' selected.

Display Stateflow charts with **Enable C-bit Operations** selected. Identify bitwise operators (&, |, and ^) in expressions containing Boolean data types.

Passed

The following charts have **Enable C-bit Operations** selected:

N/A

No bitwise operators in expressions containing Boolean data types were found in the above charts.

Identify bitwise operators in charts with 'Enable C-bit Operations' cleared

Display Stateflow charts with **Enable C-bit Operations** cleared. Identify bitwise operators (&, |, and ^) in expressions.

Passed

The following charts have **Enable C-bit Operations** cleared:

N/A

No bitwise operators in expressions with Boolean data types were found in the above charts.

Check for comparison operations in Stateflow charts

Identify comparison operations with different data types in Stateflow objects.

Passed

No comparison operations with different data types were found.

Check for unary minus operations on unsigned integers in Stateflow charts

Identify unary minus operations applied to unsigned integers in Stateflow objects.

Passed

No unary minus operations applied to unsigned integers in Stateflow objects were found.

Check for equality operations between floating-point expressions in Stateflow charts

Identify equal to operations (==) in expressions where at least one side of the expression is a floating-point variable or constant.

Passed

No equal to operations in expressions where at least one side of the expression is a floating-point variable or constant were found.

Check for mismatches between names of Stateflow ports and associated signals

Identify mismatches between names of Stateflow ports and the associated signals.

Passed

No Stateflow charts were found.

Check scope of From and Goto blocks

Identify incorrect scoping of From and Goto blocks. For signal flows, From and Goto blocks must use local scope. Control flow can use global scope.

Passed

All From and Goto blocks are used correctly.

Model Metrics 10 0 0 0

Simulink block metric

Display number of blocks in the model or subsystem.

Total Blocks: 56

Passed

Component	Blocks
Quadkopter_Model/Attitude Controller/Altitude (Z) Control	12
Quadkopter_Model/Attitude Controller	11
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control	11
Quadkopter_Model/Attitude Controller/Roll (Phi) Control	11
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control	11

Subsystem metric

Display number of Subsystems in the model or subsystem.

Total Subsystems: 4

Passed

Component	Subsystems
Quadkopter_Model/Attitude Controller	4

Quadkopter_Model/Attitude Controller/Altitude (Z) Control	0
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control	0
Quadkopter_Model/Attitude Controller/Roll (Phi) Control	0
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control	0

Library link metric

Display number of library links in the model or subsystem.

Total Library Links: 0

Passed

Component	Library Links
Quadkopter_Model/Attitude Controller	0
Quadkopter_Model/Attitude Controller/Altitude (Z) Control	0
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control	0
Quadkopter_Model/Attitude Controller/Roll (Phi) Control	0
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control	0

Effective lines of MATLAB code metric

Display number of effective lines of MATLAB code.

No metric data available. The component scope of the metric algorithm is not applicable.

Passed

Stateflow chart objects metric

Display number of Stateflow objects in each chart.

No metric data available. The component scope of the metric algorithm is not applicable.

Passed

Lines of code for Stateflow blocks metric

Display number of code lines for Stateflow blocks.

No metric data available. The component scope of the metric algorithm is not applicable.

Passed

Nondescriptive block name metric

Display non-descriptive names of Inport, Outport and Subsystem blocks.

Total Nondescriptive Names: 0

Passed

Component	Nondescriptive Names
Quadkopter_Model/Attitude Controller	0
Quadkopter_Model/Attitude Controller/Altitude (Z) Control	0
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control	0
Quadkopter_Model/Attitude Controller/Roll (Phi) Control	0
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control	0

Data and structure layer separation metric

Display data and structure layer separation, defined by MAAB modeling guideline db_0143.

Total Non-conforming Blocks: 0

Passed

Component	Non-conforming Blocks
Quadkopter_Model/Attitude Controller	0
Quadkopter_Model/Attitude Controller/Altitude (Z) Control	0
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control	0
Quadkopter_Model/Attitude Controller/Roll (Phi) Control	0
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control	0

Subsystem depth metric

Display depth of subsystems in the model or subsystem.

Passed

Component	Subsystem Level	Subsystem Depth
Quadkopter_Model/Attitude Controller/Altitude (Z) Control	1	1
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control	1	1
Quadkopter_Model/Attitude Controller/Roll (Phi) Control	1	1
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control	1	1
Quadkopter_Model/Attitude Controller	2	0

Cyclomatic complexity metric

Display cyclomatic complexity for Simulink and Stateflow objects.

Passed

Component	Local Complexity (Component level)	Aggregated Complexity (Component level and descendants)
Quadkopter_Model/Attitude Controller	**0**	0
Quadkopter_Model/Attitude Controller/Altitude (Z) Control	**0**	0
Quadkopter_Model/Attitude Controller/Pitch (Theta) Control	**0**	0
Quadkopter_Model/Attitude Controller/Roll (Phi) Control	**0**	0
Quadkopter_Model/Attitude Controller/Yaw (Psi) Control	**0**	0

Requirements Consistency 4 0 0 0

Identify requirement links with missing documents

Passed

Identify requirement links that specify invalid locations within documents

Passed

Identify selection-based links having description fields that do not match their requirements document text

Passed

Identify requirement links with path type inconsistent with preferences

Passed