MathWorks
Accelerating the pace of engineering and science

R2015a Service Pack 1 Update 3 Release Notes

Information about bugs fixed and limitations for the update

R2015a Service Pack 1 Update 3 contains bug fixes to R2015a Service Pack 1, including all the fixes from previous updates.

Important Limitations

1. Installation
   - If you add products after installing the Update, you must run the Update installer again to ensure all products are at the same update level.
2. MATLAB Distributed Computing Server, MATLAB Distributed Computing Server for Amazon EC2, and MATLAB Distributed Computing Server - Private Cloud
   - Install the update on both client and worker installations.
3. MATLAB Compiler and MATLAB Compiler SDK
   - After installing the Update, test your deployed applications using the version of the MATLAB Runtime installed on the system of the end-user or MATLAB Production Server.

Bugs Fixed in Update 3

Bug Report	*Summary*
1844615	**Security Issue:** October 2018 security updates for R2013b – R2018a MATLAB, Polyspace, and MATLAB Compiler and Runtime
1843737	**Security Issue:** Running Polyspace or MATLAB with Polyspace installed for R2013b – R2018a may allow remote code execution
1209974	**Incorrect Code Generation:** Incorrect code generated for multiple calls to Rte_Read within a runnable for an explicit read root inport branched to non-reusable subsystem
1214588	**Incorrect Code Generation:** Possible incorrect value for 0th element of static constant boolean array in generated code
1220680	**Incorrect Code Generation:** Possible wrong answer in generated code for Interpolation using Prelookup block related to output data type overflow
1228549	**Incorrect Code Generation:** Side effects of logical expressions might be incorrectly missing in generated code
1282685	**Incorrect Code Generation:** Call to a persistent function that reads a persistent variable without writing it, and has its own code to error if the persistent variable is uninitialized may produce wrong answers
1295281	**Incorrect Code Generation:** `eig` generates incorrect eigenvectors in some cases when the input matrix has rows or columns with at most one nonzero entry
1295557	**Incorrect Code Generation:** Certain operations might be evaluated unexpectedly when used in a logical expression
1297798	**Incorrect Code Generation:** Nested `if` statements with identical conditions containing function calls with side effects may be incorrectly combined into one `if` statement in the generated code

10/23/2018

1330309	**Incorrect Code Generation:** Division operations specified with saturation on integer overflow and nontrivial scaling incorrectly replaced with division implementation that wraps on integer overflow
1363959	Abs block might have incorrect coverage when the input and output of the Abs block have different data types
1404825	**Incorrect Code Generation:** Constant block feeding reusable subsystem may lead to incorrect code generation
1448125	**Incorrect Code Generation:** Possible incorrect code generation for nested handle objects with different lifetimes
1466332	**Incorrect Code Generation:** Incorrect variable assignment to 0.0 or -0.0 in the true and false branches of `if` statement
1470176	**Incorrect Code Generation:** Switch statements using an enumeration with nonunique members may produce a wrong answer in generated code
1477265	**Incorrect Code Generation:** Global storage class with alias on reusable subsystem output signal may cause incorrect code generation
1484948	**Incorrect Code Generation:** Possible wrong answer for simulation and code generation for Sum block whose **Accumulator data type** value is fixed-point with bias
1493338	Incorrect Relational Boundary coverage for saturation block with differing fixed-point input/output datatypes
1533109	**Incorrect Code Generation:** Generated code may be incorrect for a Constant block with a sample time other than `inf` that branches to multiple destinations
1556482	**Incorrect Code Generation:** Incorrect generated code is possible for an inlined For Each subsystem with separate output and update functions
1592176	**Incorrect Code Generation:** Incorrect code may be generated when Discrete Integrators State Port is used and **Single output/update function** parameter is selected
1593304	Generated code for `schur`,`sqrtm`, and `eig` may produce wrong answers
1689177	**Incorrect Code Generation:** Lookup Table (n-D) or Prelookup blocks incorrectly share same prelookup function for some fixed-point inputs and breakpoints
1694869	**Incorrect Code Generation:** Incorrect generated code is possible for multitasking mode
1715853	Proof Objective, Proof Assumption, Test objective and Test condition blocks produce incorrect answers for enumeration values derived from built-in integer types
1734679	Test cases might fail to achieve intended model coverage for objectives impacted by 'mod' operator
1736764	**Incorrect Code Generation:** Incorrect code may be generated for fraction output of Prelookup block in models containing multiple similar Prelookup blocks
1767818	**Incorrect Code Generation:** Incorrect variable reuse when a Model block connects to a Unit Delay block
1771005	Unused local variables in model initialization function
1774427	**Incorrect Code Generation:** Code and Simulation mismatch when Simulink Function is in an algebraic loop
1791690	**Incorrect Code Generation:** The simulation of Direct Lookup Table (n-D) block outputs wrong result for some 1-D tables

1794275	**Incorrect Code Generation:** Mismatch in behavior with MATLAB when using element-wise multiplication `(.*)` that mixes `double` and `int64` or `uint64` inputs
1175571	Function `pow` may return incorrect analysis results when second argument is a nonpositive integer

Bug Diagnostics Added in Update 3

Update 3 includes additional diagnostics to prevent the following bugs. Error messages displayed will indicate bug ID. Please include bug ID when contacting Technical Support.

Bug Report	*Summary*
1180720	Code generation can fail to detect the use of an unassigned variable
1188011	**Incorrect Code Generation:** Mismatch in behavior with MATLAB when an unassigned structure array is passed to the `size` function
1228373	**Incorrect Code Generation:** Rounding error may occur in generated code that casts a matrix of complex `double` to a matrix of complex `int`
1238796	**Incorrect Code Generation:** Passing a colon as an argument to a function handle generates code instead of throwing an error
1242805	**Incorrect Code Generation:** Literal character escape sequence might be incorrect in generated code
1244362	**Incorrect Code Generation:** Integer overflow may be incorrectly missing from generated code
1245231	Mismatch in behavior with MATLAB when passing an empty character array to the `'%s'` format specifier of `fprintf`
1278094	**Incorrect Code Generation:** Switch statement with a character switch expression and noncharacter case expressions, or vice versa, may result in a wrong answer in generated code
1286919	Saturation port of Discrete-Time Integrator block can give incorrect results
1332690	**Incorrect Code Generation:** Incorrect code generation for MATLAB Function block that uses negative array offsets
1337735	**Incorrect Code Generation:** Generating code for a model with floating point to fixed point conversions may result in incorrect generated code
1415169	**Incorrect Code Generation:** Product, Divide, or Product of Elements blocks with scalar fixed-point output might incorrectly cast intermediate product or division
1423206	**Incorrect Code Generation:** Casting floating-point integral values to fixed-point values might produce a wrong answer in generated code
1433431	**Incorrect Code Generation:** In code generation, using an undefined variable as an index into a matrix or cell array might produce an incorrect answer
1434545	**Incorrect Code Generation:** Code replacement of operations with conceptual arguments of type pointer can generate incorrect code
1435191	**Incorrect Code Generation:** Generated code for Stateflow Chart may be incorrect when a variable is used in multiple parallel states
1435727	**Incorrect Code Generation:** AUTOSAR model with implicit read inside Stateflow function generates incorrect RTE API name

10/23/2018

1453636	**Incorrect Code Generation:** In MATLAB Function Block, using concatenation to initialize a dynamic matrix generates incorrect answers
1465637	**Incorrect Code Generation:** Unexpected fixed-point data type propagation in MinMax block
1470564	**Incorrect Code Generation:** Possible loss of precision when constant folding large `int64/uint64` inputs to binary `max` and `min` functions in code generation
1492260	**Incorrect Code Generation:** Incorrect generated code is possible for a subsystem with a single output and nonreusable function packaging with arguments
1503581	**Incorrect Code Generation:** `TflCOperationEntryGenerator` and `TflCOperationEntryGenerator_NetSlope CRL` entries can produce incorrect replacements in generated code for add, minus, cast, and shift operations
1511597	**Incorrect Code Generation:** The Pulse Generator block could output incorrect pulses for simulation and generated code
1524916	**Incorrect Code Generation:** Incorrect logic in generated code when atomic subsystem has a single outport fed by a block conditionally writing to its output
1541475	**Incorrect Code Generation:** Breakpoint data type not propagated to input signal for n-D Lookup Table block
1558658	**Incorrect Code Generation:** Casting to logical from a fixed-point type with slope bias scaling that cannot represent zero can produce the wrong answer
1591606	**Incorrect Code Generation:** Mismatch in answer with MATLAB when the input to `isreal` is an object, a structure, or a function handle
1631380	**Incorrect Code Generation:** Reusable functions generated from subsystems connected to root inports with different bus types can get reused incorrectly in generated code
1645303	**Incorrect Code Generation:** Incorrect generated code is possible for temporary variables with constant value indexing inside of a for loop body
1687090	**Incorrect Code Generation:** Incorrect code may be generated for a For Iterator Subsystem containing an Assignment block with the input port `<Y0>` enabled
1737700	**Incorrect Code Generation:** Incorrect code generated for the n-D Lookup table block with **Interpolation method** `Nearest` and **Remove protection against out-of-range input in generated code** check box selected
1772936	**Incorrect Code Generation:** Simulink Function blocks that specify shared input/output argument might reuse wrong input in generated code
1793805	**Incorrect Code Generation:** Write to variable missing when variable is size vector argument of reshape function call within a loop
1804775	**Incorrect Code Generation:** Using Discrete State-Space block with state name defined as xnew can lead to incorrect answer during code generation
1452753	**Incorrect Code Generation:** Incorrect representation of down-counting array subscript with constant unsigned start and stop value

Bugs Fixed in Update 2 (also included in Update 3)

Bug Report	*Summary*
1284809	Embedded Coder may fail to build a model that reuses local block outputs and contains a Unit Delay block

10/23/2018

1286055	**Incorrect Code Generation:** Uninitialized local variable is used for the output buffer of Unit Delay block
1287804	Embedded Coder may fail to build a model when a Unit Delay block directly connects to a reusable subsystem output
1301976	Embedded Coder generates C code that references an undefined variable when a model contains multiple instances of subsystems connected to Unit Delay blocks
1336471	**Incorrect Code Generation:** Possible rounding error or overflow due to incorrect translation of division and multiplication operations
1373141	**Incorrect Code Generation**: Multiple subsystems incorrectly share the same state variable for Unit Delay blocks
1381708	`ModelAdvisor.run()` does not support a large number of models
1495807	**Incorrect Code Generation**: Incorrect generated code is possible for S-function with `ssSetOutputPortOutputExprInRTW` function set to `TRUE`
1637482	**Incorrect Code Generation**: Incorrect generated code is possible for AUTOSAR target when an S-function sets `ssSetOutputPortOutputExprInRTW` to `TRUE`
1649889	**Incorrect Code Generation**: State variable for Unit Delay incorrectly overwritten for subsystems whose function interface includes arguments
1658667	**Incorrect Code Generation**: Uninitialized local variable is used when an Assignment block connects to a Unit Delay block
1668727	**Incorrect Code Generation**: Incorrect code may be generated when an atomic subsystem output connects to a Unit Delay block and the **Optimize global data access** parameter is set to a value other than `none`
1710500	**Incorrect Code Generation**: Simulink model with Variant subsystem might incorrectly generate preprocessor conditionals
1771523	**Incorrect Code Generation**: Incorrect results might occur when the sample times of the input and output of a Unit Delay block are different and **Optimize global data access** parameter is set to a value other than `none`
1773229	**Incorrect Code Generation:** Reusable custom storage class specification at the input of a Unit Delay block can lead to incorrect code
1773232	**Incorrect Code Generation:** Incorrect variable reuse when a Unit Delay block connects to a Constant block inside an enabled, triggered, or function-call subsystem

Bugs Fixed in Update 1 (also included in Update 3)

Bug Report	*Summary*
1189298	Deleting a Stateflow subchart and undoing might crash MATLAB
1193775	Copy and paste of both a subchart and supertransition causes a crash
1284929	Slow performance when exiting MATLAB after using Model Advisor or Upgrade Advisor
1294093	**Incorrect Code Generation:** Incorrect code generation when branched signals are present in Variant choice blocks feeding into an outport
1298221	Parameter tuning in the presence of other errors can fail silently
1309455	S-function parameterized by reference to struct parameter may cause model build to fail

10/23/2018

1321874	Comments in Truth Tables using non-ASCII characters are corrupted when saved to MDL format Truth Table
1336928	Build processes that use the LCC-win64 compiler do not complete parallel build of a model that contains referenced models
1339267	**Incorrect Code Generation:** Model blocks with multiple function-call input ports driven by Function Call Split blocks using Simulink Accelerator execute the wrong function calls during simulation
1341923	Code generated for variant subsystem does not compile
1355378	**Incorrect Code Generation:** Generated code for MATLAB function block or Stateflow Chart containing variable-sized arrays may be incorrect
1355727	Running a model from a working directory with an existing slprj folder can cause MATLAB to crash
1362661	Uncompilable code may be generated for custom S-function that has expression folding enabled
1364309	**Incorrect Code Generation:** Generated code for a Variant Subsystem that is nested in control flow may be missing a required curly brace
1365276	Error message occurs on Windows stating that the build has failed
1371315	**Incorrect Code Generation:** Uninitialized size vector in the generated code
1378420	**Incorrect Code Generation:** Generated C/C++ code for a recursive MATLAB function inside Stateflow Chart may be incorrect if the function contains a large variable
1384324	**Incorrect Code Generation:** Incorrect generated code is possible when an Assignment block and a Unit Delay block are in the same For iterator subsystem
1386965	ASAP2 code generation error “Unable to find BlockFcns within the scope” for reference models using export-functions modeling style
1388596	**Incorrect Code Generation:** Generated code may be incorrect for a model that contains chained Merge blocks that span reusable subsystem boundaries
1404201	Variant subsystem fail to build when using bitsets to store state configuration/boolean data
1416249	A PostLoadFcn callback within the library having commented blocks causes its instance in the model to uncomment
1421253	With model in accelerator mode, Root Inport Mapping tool with compile option generates error
1430224	`bitshift(A,K)` returns the wrong answer in generated code and MATLAB Function blocks when `K` is a fi object and `K==0`
1435708	AUTOSAR 4.0 code replacement library might incorrectly replace a fixed-point operation
1436779	The Model Advisor check Check for propagated signal labels can report an out of scope violation when running on a Simulink subsystem
1444200	SIL and PIL simulations with LDRA code coverage can produce an error
	Performance Improvement - Optimize Stateflow transition routing in large models
1453109	**Incorrect Code Generation:** Embedded Coder can produce unexpected code interface for a Simulink Function block connected to a root Outport block
1455531	Stateflow transitions are drawn on the wrong subviewer
1458078	Protected model creation fails to locate configuration sets located in data dictionaries when the original model is using configset reference
1463792	The memory section associated with a subfunction is unpredictable

1466966	**Incorrect Code Generation:** Generated code for S-function with non-scalar signal input can produce incorrect numerical results
1468884	**Incorrect Code Generation:** Incorrect generated code is possible for atomic subsystem with no outports and with Minimize algebraic loop occurrences parameter set to on
1473861	Subsystem output is incorrectly initialized in certain cases when subsystem connects to a Merge block
1475400	**Incorrect Code Generation:** Incorrect generated code may occur when virtual blocks reorder the input data to reduced block
1475873	**Incorrect Code Generation:** AUTOSAR 4.0 code replacement library `Mfx_MulShRight_*` entries can provide incorrect calculation
1480737	MATLAB crashes when simulating the Simulink Math Function block with Function set to `mod` or `rem`
1489596	**Incorrect Code Generation:** Incorrect `Mfx_Div*` replacements can occur in the generated code when using the AUTOSAR 4.0 CRL Library
1493674	Models with Rate Limiter block and model reference can crash while doing simulation
1505057	MATLAB throws an error when trying to generate code for a model with reusable subsystem connected to a block using TLC

Update 1 also includes performance improvements related to common Simulink and Stateflow actions including: line routing, dragging/moving blocks in a large models, copy/paste in the model, `open_system`, model navigation, create subsystem, and transition routing in large models.