Embedded MATLAB™ Release Notes

How to Contact The MathWorks

`www.mathworks.com`	Web
`comp.soft-sys.matlab`	Newsgroup
`www.mathworks.com/contact_TS.html`	Technical Support

`suggest@mathworks.com`	Product enhancement suggestions
`bugs@mathworks.com`	Bug reports
`doc@mathworks.com`	Documentation error reports
`service@mathworks.com`	Order status, license renewals, passcodes
`info@mathworks.com`	Sales, pricing, and general information

508-647-7000 (Phone)

508-647-7001 (Fax)

The MathWorks, Inc.
3 Apple Hill Drive
Natick, MA 01760-2098

For contact information about worldwide offices, see the MathWorks Web site.

Embedded MATLAB™ Release Notes

Trademarks

MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See `www.mathworks.com/trademarks` for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.

Patents

The MathWorks products are protected by one or more U.S. patents. Please see `www.mathworks.com/patents` for more information.

Contents

Summary by Release

This table provides quick access to what's new in each release. For clarification, see "Using Release Notes" on page 3.

Version (Release)	New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Latest Release R2009b	Yes Details	Yes Summary	Search bug reports for the products that support Embedded MATLAB™: • Fixed-Point Toolbox™ • Real-Time Workshop® • SimEvents® • Simulink® • Stateflow®	Printable Release Notes: PDF Current documentation
R2009a	Yes Details	Yes Summary	Search bug reports for the products that support Embedded MATLAB™: • Fixed-Point Toolbox • Real-Time Workshop • SimEvents • Simulink • Stateflow	No

Version (Release)	New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
R2008b	Yes Details	Yes Summary	Search bug reports for the products that support Embedded MATLAB™: • Fixed-Point Toolbox • Real-Time Workshop • SimEvents • Simulink • Stateflow	No
R2008a	Yes Details	Yes Summary	Search bug reports for the products that support Embedded MATLAB™: • Fixed-Point Toolbox • Real-Time Workshop • SimEvents • Simulink • Stateflow Includes fixes	No

Version (Release)	New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
R2007b	Yes Details	Yes Summary	Search bug reports for the products that support Embedded MATLAB™: • Fixed-Point Toolbox • Real-Time Workshop • SimEvents • Simulink • Stateflow Includes fixes	No

Using Release Notes

Use release notes when upgrading to a newer version to learn about:

- New features
- Changes
- Potential impact on your existing files and practices

Review the release notes for other MathWorks™ products required for this product (for example, MATLAB® or Simulink). Determine if enhancements, bugs, or compatibility considerations in other products impact you.

If you are upgrading from a software version other than the most recent one, review the current release notes and all interim versions. For example, when you upgrade from V1.0 to V1.2, review the release notes for V1.1 and V1.2.

What Is in the Release Notes

New Features and Changes

- New functionality
- Changes to existing functionality

Version Compatibility Considerations

When a new feature or change introduces a reported incompatibility between versions, the **Compatibility Considerations** subsection explains the impact.

Compatibility issues reported after the product release appear under Bug Reports at The MathWorks™ Web site. Bug fixes can sometimes result in incompatibilities, so review the fixed bugs in Bug Reports for any compatibility impact.

Fixed Bugs and Known Problems

The MathWorks offers a user-searchable Bug Reports database so you can view Bug Reports. The development team updates this database at release time and as more information becomes available. Bug Reports include provisions for any known workarounds or file replacements. Information is available for bugs existing in or fixed in Release 14SP2 or later. Information is not available for all bugs in earlier releases.

Access Bug Reports using your MathWorks Account.

Release 2009b Embedded MATLAB

This table summarizes what's new in R2009b:

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Yes Details below	Yes—Details labeled as **Compatibility Considerations**, below. See also Summary	Search bug reports for the products that support Embedded MATLAB™: • Fixed-Point Toolbox • Real-Time Workshop • SimEvents • Simulink • Stateflow Includes fixes	Printable Release Notes: PDF Current documentation

New features and changes introduced in this release are:

Support for Variable-Size Arrays and Matrices

The Embedded MATLAB subset now supports variable-size arrays and matrices with known upper bounds. With this feature, you can define inputs, outputs, and local variables in Embedded MATLAB functions to represent data that varies in size at runtime. You can use variable-size data in:

- Embedded MATLAB compliant M functions, from which you can generate MEX code with `emlmex` and C code with `emlc` (requires Real-Time Workshop software)
- Embedded MATLAB Function blocks in Simulink
- Embedded MATLAB functions in Stateflow charts

For more information, see "Generating Code for Variable-Size Data" in the *Embedded MATLAB™ User's Guide*.

New Options for Controlling Runtime Checks for Faster Performance

`emlmex` generates code that includes runtime checks and external calls. To reduce the size of the generated code, and potentially improve performance, you can use new C-MEX compilation options to control whether or not your generated code performs:

- Integrity checks that detect violations of memory integrity in the generated code
- Responsiveness checks that periodically check for Ctrl+C breaks and refresh graphics
- Extrinsic calls to MATLAB functions

For more information, see "Controlling Runtime Checks" in the *Embedded MATLAB™ User's Guide*.

Ability to Ignore Selected Arguments on Function Calls

The Embedded MATLAB subset supports a new MATLAB operator, the tilde (~) character, which lets you specify unused outputs in a function call, or unused inputs in a function definition. Embedded MATLAB ignores inputs and outputs specified by the tilde (~) placeholder.

This new syntax can help you avoid confusion in your program code and unnecessary clutter in your workspace.

For more information, see "Tilde — ~" in the MATLAB Programming Fundamentals documentation.

Support for Enumerated Data in switch Expressions

You can now use enumerated data as the control variable in `switch` statements in Embedded MATLAB functions.

For more information, see “Control Flow Statements: if, switch, while” in the *Embedded MATLAB™ User's Guide*.

New Runtime Stack for Embedded MATLAB Functions

New runtime stack to help you determine the source of runtime errors in your Embedded MATLAB functions. The runtime stack provides information about the function that generated the error and the sequence of function calls that led up to the execution of this function. For more information, see Debugging Runtime Errors in the Embedded MATLAB User's Guide.

New Embedded MATLAB Runtime Library Functions

The Embedded MATLAB runtime library now provides the following new functions. To view details of the Embedded MATLAB implementation of these functions, see “Embedded MATLAB Function Library — Alphabetical List”.

Data Validation Functions

- `nargchk`
- `nargoutchk`

Fixed-Point Toolbox Functions

- `conv`
- `isfimathlocal`
- `sfi`
- `ufi`

Matrix and Array Functions

- `find`
- `nnz`
- `nonzeros`
- `randperm`

Nonlinear Numerical Methods

- `fzero`
- `quad2d`

Polynomial Functions

- `roots`

Signal Processing Functions

- `barthannwin`
- `bartlett`
- `blackman`
- `blackmanharris`
- `bohmanwin`
- `butter`
- `cfirpm`
- `chebwin`
- `cheby1`
- `cheby2`
- `dpss`
- `ellip`
- `fir1`
- `fir2`
- `fircls`
- `fircls1`
- `firls`
- `firpm`
- `firpmord`

- firrcos
- flattopwin
- gaussfir
- gausswin
- hamming
- hann
- intfilt
- kaiser
- kaiserord
- maxflat
- nuttallwin
- parzenwin
- rectwin
- sgolay
- taylorwin
- triang
- tukeywin
- yulewalk

Special Values

- bitmax

String Functions

- blanks

Structure Functions

- isfield

Change in Case Sensitivity for false, inf, length, nan, and true

In R2009b, `false`, `inf`, `length`, `nan`, and `true` are now case sensitive to match MATLAB.

You must now use:

- `false`
- `inf` or `Inf`
- `length`
- `nan` or `NaN`
- `true`

Compatibility Considerations

Embedded MATLAB functions that compiled successfully in previous releases will fail to compile with R2009b if they do not use the correct case for `false`, `inf`, `length`, `nan`, or `true`.

Release 2009a Embedded MATLAB

This table summarizes what's new in R2009a:

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Yes Details below	Yes—Details labeled as **Compatibility Considerations**, below. See also Summary	Search bug reports for the products that support Embedded MATLAB: • Fixed-Point Toolbox • Real-Time Workshop • SimEvents • Simulink • Stateflow Includes fixes	Printable Release Notes: PDF Current documentation

New features and changes introduced in this release are:

Support for Enumerated Types

The Embedded MATLAB subset now supports enumerated types, which allow you to use meaningful names for defining data that takes a limited set of predefined values, such as system states or modes. Enumerations offer advantages over strings in efficiency (comparisons are faster) and error checking of code generated by `emlmex`. For more information, see "Working with Enumerated Data" in the Embedded MATLAB documentation.

New Compilation Report for emlmex

The new compilation report provides compile-time type information for the variables and expressions in your M-code. This information helps you

find the sources of error messages and understand type propagation rules, particularly for fixed-point data types. For more information, see "Working with Compilation Reports" in the Embedded MATLAB documentation.

Compatibility Considerations

The following internal and external browsers do not support the new compilation report:

- MATLAB internal browser (on 64-bit UNIX® platforms only)
- MACI internal browser
- Microsoft Internet Explorer® 6

To view the compilation report, you must have your MATLAB Web preferences configured to use an external browser, for example, Mozilla® Firefox®. To learn how to configure your MATLAB Web preferences, see "Specifying the System Browser for UNIX Platforms" in the MATLAB documentation.

New -report Compilation Option for emlmex

The new `-report` compilation option for `emlmex` provides the ability to request a compilation report. If this option is not specified, `emlmex` generates a report only if there are error or warning messages. For more information, see `emlmex`.

Use of Basic Linear Algebra Subprograms (BLAS) Libraries for Speed

The Embedded MATLAB subset now uses BLAS libraries to speed up low-level matrix operations during simulation. For more information, see "Speeding Up Simulation with the Basic Linear Algebra Subprograms (BLAS) Library".

New Embedded MATLAB Runtime Library Functions

The Embedded MATLAB runtime library now provides the following new functions:

Fixed-Point Toolbox Functions

- `add`
- `mpy`
- `mrdivide`
- `rdivide`
- `sub`

Signal Processing Functions

- `fft2`
- `ifft2`

Histogram Functions

- Non-graphical `hist`

Release 2008b Embedded MATLAB

This table summarizes what's new in R2008b:

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Yes Details below	Yes—Details labeled as **Compatibility Considerations**, below. See also Summary	Search bug reports for the products that support Embedded MATLAB: • Fixed-Point Toolbox • Real-Time Workshop • SimEvents • Simulink • Stateflow Includes fixes	Printable Release Notes: PDF Current documentation

New features and changes introduced in this release are:

Support for Fixed-Point Word Lengths Up to 128 Bits

The Embedded MATLAB subset now supports up to 128 bits of precision for accelerating fixed-point algorithm execution with the `emlmex` function and generating efficient code for targets with nonstandard word sizes.

New eml.nullcopy Function for Declaring Uninitialized Variables

The function `eml.nullcopy` allows you to declare the size, type, and complexity of a data variable without forcing unnecessary initialization of memory.

See `eml.nullcopy` in the Embedded MATLAB Function Reference and "Declaring Uninitialized Variables" in the Embedded MATLAB User's Guide.

New Embedded MATLAB Getting Started Tutorial

This tutorial provides new users with hands-on experience to get them started quickly with the Embedded MATLAB subset and achieve the goal of generating efficient, embeddable C code from existing M-code. See "Introduction to the Embedded MATLAB Subset" in the Embedded MATLAB Getting Started documentation.

Empty Fields in Structures No Longer Produce Errors

You can now define structures that have empty fields.

emlmex Uses Same Trigonometric Functions as MATLAB

`emlmex` now uses the same library as MATLAB for implementing trigonometric operations during simulation. In previous releases, `emlmex` used the C compiler implementation of trigonometric functions, sometimes producing different results from MATLAB. Now, results are consistent with MATLAB and independent of compiler choice.

New Embedded MATLAB Runtime Library Functions

The Embedded MATLAB runtime library now provides the following new functions:

Fixed-Point Toolbox Functions

- `reinterpretcast`
- `sort`

Matrix and Array Functions

- `circshift`

- `isequalwithequalnans`

String Functions

- `bin2dec`
- `hex2dec`

Improperly-Scaled Fixed-Point Relational Operators Now Match MATLAB Results

When evaluating relational operators, the Embedded MATLAB subset computes a common type that encompasses both input operands. In previous releases, if the common type required more than 32 bits, Embedded MATLAB functions and code generated by `emlmex` may have given different answers from MATLAB. Now, results are consistent.

Compatibility Consideration

Some relational operators generate multi-word code even if one of the fixed-point operands is not a multi-word value. To work around this issue, cast both operands to the same fixed-point type (using the same scaling method and properties).

emlmex Now Prevents Default fimath Mismatches for MEX Functions

MEX functions generated with `emlmex` use the default `fimath` value in effect at compile time. If you do not specify a default `fimath` value explicitly using the `-F` option, `emlmex` uses the MATLAB default `fimath` value. If your MEX function uses the MATLAB default `fimath`, `emlmex` generates an error if the compile-time value does not match the runtime value. See “Specifying Default fimath Values for MEX Functions” in the Embedded MATLAB documentation.

Release 2008a Embedded MATLAB

This table summarizes what's new in R2008a:

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Yes Details below	Yes—Details labeled as **Compatibility Considerations**, below. See also Summary	Search bug reports for the products that support Embedded MATLAB: • Fixed-Point Toolbox • Real-Time Workshop • SimEvents • Simulink • Stateflow Includes fixes	Printable Release Notes: PDF Current documentation

New features and changes introduced in this release are:

Support for varargin and varargout

You can now use `varargin` and `varargout` for specifying variable numbers of arguments in Embedded MATLAB functions. The Embedded MATLAB subset supports use of `varargin` and `varargout` for specifying property-value pairs, in for-loops, and for passing arguments from one function to another. See "Specifying Variable Numbers of Arguments in Embedded MATLAB Functions" in the Embedded MATLAB User's Guide.

eml.inline Function for Controlling Inlining in Generated Code

The new `eml.inline` function allows you to control inlining of Embedded MATLAB functions in generated code. See "Inlining Functions" in the Embedded MATLAB User's Guide.

eml.unroll Function for Loop Unrolling

The new `eml.unroll` function allows you to unroll for-loops in Embedded MATLAB functions to improve performance in generated code. See "Unrolling for-Loops" in the Embedded MATLAB User's Guide.

eml.allowpcode Function for Compiling P-Code

The new `eml.allowpcode` function lets you compile P-code that is compliant with the Embedded MATLAB subset. With this capability, you can distribute algorithms that use Embedded MATLAB optimizations and also protect your source code. See `eml.allowpcode` in the Embedded MATLAB Function Reference.

emlcoder.Example and emlcoder.egc Directives for Specifying Constant Inputs

When you know primary inputs will not change at runtime, you can specify them as constant values using `emlcoder.Example` and `emlcoder.egc` to eliminate overhead in generated code. See "Specifying Constant Inputs" in the Embedded MATLAB User's Guide.

New Embedded MATLAB Runtime Library Functions

The Embedded MATLAB runtime library now provides fifteen new functions in the following categories:

Aerospace Toolbox Quaternion Functions

- `quatconj`
- `quatdivide`
- `quatinv`
- `quatmod`
- `quatmultiply`
- `quatnorm`
- `quatnormalize`

Fixed-Point Toolbox Functions

- `bitreplicate`
- `ceil`
- `convergent`
- `fix`
- `floor`
- `nearest`
- `round`

Matrix and Array Functions

- `ndgrid`

emlmex Searches Directories in Different Order

When you use the `-I` option to add directories to the Embedded MATLAB path, `emlmex` now searches directories from left to right. This behavior matches C compilers such as `gcc` and the MATLAB compiler `mcc`.

Compatibility Considerations

In previous releases, the search order was from right to left. The change may produce unexpected results if you use `emlmex -I` with overloaded functions. For example, suppose you specify two folders on the `-I` argument list, where each folder contains a file named `myFcn.m`, as follows:

```
emlmex -I 'dir1;dir2' myFcn.m
```

In this release, `emlmex` adds `myFcn.m` from `dir1` to the Embedded MATLAB path. In previous releases, `dir2` took precedence.

Release 2007b Embedded MATLAB

This table summarizes what's new in R2007b:

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Yes Details below	Yes—Details labeled as **Compatibility Considerations**, below. See also Summary.	Search bug reports for the products that support Embedded MATLAB: • Fixed-Point Toolbox • Real-Time Workshop • SimEvents • Simulink • Stateflow Includes fixes	Printable Release Notes: PDF Current product documentation

New features and changes introduced in this release are:

Support for Algorithms That Span Multiple M-Files

You can now generate embeddable code for external M-functions from Embedded MATLAB. This feature allows you to call external functions from

multiple locations in an M-file or model and include these functions in the generated code. See “Writing Reusable Code” in the Embedded MATLAB documentation.

Compatibility Consideration

In previous releases, Embedded MATLAB did not compile external M-functions, but instead dispatched them to MATLAB for execution (after warning). Now, the default behavior is to compile and generate code for external M-functions called from Embedded MATLAB. If you do not want Embedded MATLAB to compile external M-functions, you must explicitly declare them to be extrinsic, as described in “Calling MATLAB Functions” in the Embedded MATLAB documentation.

Calling C Functions from Embedded MATLAB Now Supported

Previously, you could not call existing C code from Embedded MATLAB code. In R2007b, a new `eml` interface is provided to help you integrate existing C code with Embedded MATLAB code. For details, see “Calling C Functions from the Embedded MATLAB Subset” in the Embedded MATLAB documentation.

Support for Data Type Override

The Embedded MATLAB subset now supports data type override. This feature facilitates fixed-point design and enables a single source for fixed- and floating-point code generation. Requires Fixed-Point Toolbox.

Return Types of bitget and sign Functions Now Match MATLAB

The Embedded MATLAB subset has changed the return data types of the Fixed-Point Toolbox functions `bitget` and `sign` to match MATLAB, as follows:

Fixed-Point Toolbox Function	New Return Type	Old Return Type
`bitget`	**`fi` with word length = 1**	`uint8`
`sign`	**`int8`**	`fi` with word length = 8 and fraction length = 0

Compatibility Consideration

Embedded MATLAB functions that call `bitget` and `sign` may now produce different results than expected because of the new return types.

New Embedded MATLAB Runtime Library Functions

The Embedded MATLAB runtime library now provides thirteen new functions in the following categories:

- "Array Operations" on page 23
- "Fixed-Point Toolbox Bit-Wise Functions" on page 23

Array Operations

- `bsxfun`

Fixed-Point Toolbox Bit-Wise Functions

- `bitandreduce`
- `bitconcat`
- `bitorreduce`
- `bitrol`
- `bitror`
- `bitsliceget`
- `bitsll`

- `bitsra`
- `bitsrl`
- `bitxorreduce`
- `getlsb`
- `getmsb`

Compatibility Summary for Embedded MATLAB

This table summarizes new features and changes that might cause incompatibilities when you upgrade from an earlier version, or when you use files on multiple versions. Details are provided in the description of the new feature or change.

Release	New Features and Changes with Version Compatibility Impact
Latest Version R2009b	See the **Compatibility Considerations** subheading for each of these new features or
R2009a	See the **Compatibility Considerations** subheading for each of these new features or changes: • "New Compilation Report for emlmex" on page 11
R2008b	See the **Compatibility Considerations** subheading for each of these new features or changes: • "Improperly-Scaled Fixed-Point Relational Operators Now Match MATLAB Results" on page 16

Release	New Features and Changes with Version Compatibility Impact
R2008a	See the **Compatibility Considerations** subheading for each of these new features or changes: • "emlmex Searches Directories in Different Order" on page 20
R2007b	See the **Compatibility Considerations** subheading for each of these new features or changes: • "Support for Algorithms That Span Multiple M-Files" on page 21 • "Return Types of bitget and sign Functions Now Match MATLAB" on page 22