# Fixed-Point Toolbox Release Notes

The Fixed-Point Toolbox 1.3 Release Notes describe the changes introduced in the latest version of the Fixed-Point Toolbox. The following topics are discussed in these Release Notes:

- "New Features" on page 1-2
- "Major Bug Fixes" on page 1-5
- "Known Software Problems" on page 1-6

The Fixed-Point Toolbox Release Notes also provide information about the earlier versions of the product, in case you are upgrading from a version that was released prior to Release 14 with Service Pack 2.

- Chapter 2, "Fixed-Point Toolbox 1.2 Release Notes"
- Chapter 3, "Fixed-Point Toolbox 1.1 Release Notes"
- Chapter 4, "Fixed-Point Toolbox 1.0 Release Notes"

# Contents

## Fixed-Point Toolbox 1.1 Release Notes

**3**

## Fixed-Point Toolbox 1.0 Release Notes

**4**

# Fixed-Point Toolbox 1.3 Release Notes

# New Features

This section summarizes the new features and enhancements introduced in the Fixed-Point Toolbox 1.3:

- "Fixed-Point Toolbox Function Support Added to Embedded MATLAB" on page 1-2

- "Double, Single, and Boolean Data Type Support Added to the fi Object" on page 1-3

- "Helper Functions Added for Accessing Logged Information" on page 1-3

- "New Fixed-Point Doubles Override, Min/Max Logging, and Scaling Demo" on page 1-3

- "RoundMode Property Value 'round' Now Called 'nearest'" on page 1-4

If you are upgrading from a release earlier than Release 14SP2, then you should also see New Features in the Fixed-Point Toolbox 1.2 Release Notes.

## Fixed-Point Toolbox Function Support Added to Embedded MATLAB

The Simulink Embedded MATLAB Function block lets you compose a MATLAB language function in a Simulink model that generates embeddable code. When you simulate the model or generate code for a target environment, a function in an Embedded MATLAB Function block generates efficient C code. This code meets the strict memory and data type requirements of embedded target environments. In this way, Embedded MATLAB Function blocks bring the power of MATLAB for the embedded environment into Simulink.

For more information on using Embedded MATLAB, refer to the following sections in the Simulink documentation:

- Embedded MATLAB Function block reference page

- "Using the Embedded MATLAB Function Block"

- "Embedded MATLAB Function Block Reference"

You can now use a significant number of Fixed-Point Toolbox functions with Embedded MATLAB. Refer to "Using the Fixed-Point Toolbox with Embedded MATLAB" in the Using Fixed-Point Toolbox documentation.

---

**Note** To simulate models using fixed-point data types in Simulink, including when using the Embedded MATLAB Function block, you must have a Simulink Fixed Point license.

---

## Double, Single, and Boolean Data Type Support Added to the fi Object

The `fi` object now supports `double`, `single`, and `boolean` data types. The values `double`, `single`, and `boolean` have been added to the `DataType` and `DataTypeMode` properties of the `numerictype` object. Math operations are supported for `fi` objects with data type `single` or `double`, but not `boolean`.

## New Fixed-Point Doubles Override, Min/Max Logging, and Scaling Demo

Since floating-point data types are now supported in the Fixed-Point Toolbox, it is possible to use doubles override and min/max scaling to help you choose the appropriate scalings for fixed-point variables in your algorithms. This is especially helpful when converting a floating-point algorithm to fixed point. A new demo "Fixed-Point Doubles Override, Min/Max Logging, and Scaling" leads you through an example of this process. You can access this demo from the **Demos** pane of the Help browser under Toolboxes > Fixed-Point.

## Helper Functions Added for Accessing Logged Information

In the previous release it became possible to log overflows and underflows as warnings for all assignment, plus, minus, and multiplication operations when the `fipref` `LoggingMode` property is set to `on`. Now when `LoggingMode` is on, you can also use the following helper functions to return logged information to you at the MATLAB command line:

- `maxlog` — Returns the maximum real-world value

- `minlog` — Returns the minimum real-world value

- `noperations` — Returns the number of quantized operations

- `noverflows` — Returns the number of overflows

- `nunderflows` — Returns the number of underflows

To clear the log, use the function `resetlog`.

## RoundMode Property Value 'round' Now Called 'nearest'

The `RoundMode` property value `round` is now `nearest`. This is a reflection of the fact that this rounding mode is identical to the Simulink rounding mode `round toward nearest`, and different from the behavior of the MATLAB `round` function.

# Major Bug Fixes

To view major bug fixes made in R14SP3 for the Fixed-Point Toolbox, use the Bug Reports interface on the MathWorks Web site.

Note that if you are not already logged in to Access Login, when you link to the Bug Reports interface (see below), you will be prompted to log in or create an Access Login account.

After you are logged in, use this Bug Fixes link. You will see the bug report for the Fixed-Point Toolbox. The report is sorted with fixed bugs listed first, and then open bugs.

If you are viewing these release notes in PDF form on the MathWorks Web site, you can refer to the HTML form of the release notes on the MathWorks Web site and use the link provided.

For bug fixes added prior to R14SP2, see "Major Bug Fixes" on page 3-2 in the Fixed-Point Toolbox Version 1.1 release notes.

# Known Software Problems

To view important open bugs in R14SP3 for the Fixed-Point Toolbox, use the Bug Reports interface on the MathWorks Web site.

Note that if you are not already logged in to Access Login, when you link to the Bug Reports interface (see below), you will be prompted to log in or create an Access Login account.

After you are logged in, use this Open Bugs link. You will see the bug report for the Fixed-Point Toolbox. The report is sorted with fixed bugs listed first, and then open bugs. You can select the Status column to list the open bugs first.

If you are viewing these release notes in PDF form on the MathWorks Web site, you can refer to the HTML form of the release notes on the MathWorks Web site and use the link provided.

# Fixed-Point Toolbox 1.2 Release Notes

# New Features

This section introduces the new features and enhancements added to the Fixed-Point Toolbox 1.2 (Release 14 with Service Pack 2) since Fixed-Point Toolbox 1.1 (Release 14 with Service Pack 1):

- "Overflow and Underflow Logging" on page 2-2
- "New Functions" on page 2-2

## Overflow and Underflow Logging

The Fixed-Point Toolbox now allows you to log overflows and underflows as warnings for all assignment, plus, minus, and multiplication operations. Refer to "Using fipref Objects to Set Logging Preferences" in the Fixed-Point Toolbox documentation for more information.

## New Functions

The following functions are new in the Fixed-Point Toolbox 1.2:

| | | | | |
|---|---|---|---|---|
| abs | all | and | any | area |
| bar | barh | buffer | clabel | comet |
| comet3 | compass | coneplot | contour | contour3 |
| contourc | contourf | diag | end | errorbar |
| etreeplot | ezcontour | ezcontourf | ezmesh | ezplot |
| ezplot3 | ezpolar | ezsurf | ezsurfc | feather |
| fplot | gplot | hankel | hist | histc |
| inspect | intmin | ipermute | isnumeric | isobject |
| line | logical | lowerbound | mesh | meshc |
| meshz | not | numberofelements | or | patch |
| pcolor | permute | plot3 | plotmatrix | plotyy |
| polar | pow2 | quiver | quiver3 | rgbplot |
| ribbon | rose | scatter | scatter3 | sdec |
| sign | slice | spy | stairs | stem |

| stem3 | streamribbon | streamslice | streamtube | sum |
|---|---|---|---|---|
| surf | surfc | surfl | surfnorm | text |
| toeplitz | treeplot | tril | trimesh | triplot |
| trisurf | triu | uplus | upperbound | voronoi |
| voronoin | waterfall | xlim | ylim | zlim |

# Major Bug Fixes

The Fixed-Point Toolbox 1.2 includes several bug fixes made since Version 1.1. This section describes the particularly important Version 1.2 bug fixes. If you are upgrading from a version earlier than Version 1.1, you should also see the Version 1.1 Major Bug Fixes.

## Segmentation Violation Due to Multiple Loop Operations Fixed

In Version 1.1 of the Fixed-Point Toolbox in the last release, it was possible to obtain a segmentation violation when performing multiple operations on a `fi` object within a loop. This has been fixed for this release.

# Known Software Problems

This section lists known software problems in Version 1.2 of the Fixed-Point Toolbox.

## isequal Always Returns true for fimath Objects

In Release 14 with Service Pack 2, the Fixed-Point Toolbox isequal function always returns true for fimath objects. For example, the following returns 1 when it should return 0:

```
a = fimath('roundmode','floor');
b = fimath('roundmode','ceil');
isequal(a,b)

ans =

     1
```

To work around this problem, use the following syntax instead:

```
isequal(get(a),get(b))

ans =

     0
```

# Fixed-Point Toolbox 1.1 Release Notes

# Major Bug Fixes

The Fixed-Point Toolbox 1.1 includes several bug fixes made since Version 1.0. This section describes the particularly important Version 1.1 bug fixes.

### Bitwise Operators Return Correct Answers for [Slope Bias] Signals

In the previous release, bitwise functions such as `bitshift` might have given wrong answers for [Slope Bias] fixed-point signals. This has been corrected in this release.

### fi Object Operations with an Empty Array Work Properly

In the previous release, a segmentation violation occurred for any operation with the format

```
a op e
```

where `a` is a `fi` object, `e` is an empty array, and `op` is any operator such as +,-,*,.*,<,>, etc. This has been corrected in this release.

### ispropequal Returns Correct Answers for fimath Objects

The `ispropequal` function has been updated to work properly in this release.

# Fixed-Point Toolbox 1.0 Release Notes

# Introduction to the Fixed-Point Toolbox

The Fixed-Point Toolbox provides fixed-point data types in MATLAB and enables algorithm development by providing fixed-point arithmetic. The Fixed-Point Toolbox enables you to create the following types of objects:

- `fi` — Defines a fixed-point numeric object in the MATLAB workspace. Each `fi` object is composed of value data, a `fimath` object, and a `numerictype` object

- `fimath` — Governs how overloaded arithmetic operators work with `fi` objects

- `fipref` — Defines the display attributes for `fi` objects

- `numerictype` — Defines the data type and scaling attributes of `fi` objects

- `quantizer` — Quantizes data sets

## Features

The Fixed-Point Toolbox provides you with

- The ability to define fixed-point data types, scaling, and rounding and overflow methods in the MATLAB workspace

- Bit-true real and complex simulation

- Basic fixed-point arithmetic with binary point-only signals

    - Arithmetic operators +, -, *, .*

    - Division using the `divide` function

- Arbitrary word length up to `intmax('uint16')`

- Relational, logical, and bitwise operators

- Data visualization via the `plot` function

- Statistics functions such as `abs`, `max`, and `min`

- Conversions between binary, hex, double, and built-in integers

- Interoperability with Simulink, Signal Processing Blockset, and Filter Design Toolbox

- Compatibility with the Simulink To Workspace and From Workspace blocks

# Getting Help

This section tells you how to get help for the Fixed-Point Toolbox in this document and at the MATLAB command line.

## Getting Help in the Fixed-Point Toolbox User's Guide

The objects of the Fixed-Point Toolbox are discussed in the following chapters:

- "Working with fi Objects"
- "Working with fimath Objects"
- "Working with fipref Objects"
- "Working with numerictype Objects"
- "Working with quantizer Objects"

To get in-depth information about the properties of these objects, refer to "Property Reference".

To get in-depth information about the functions of these objects, refer to "Functions — Alphabetical List".

## Getting Help at the MATLAB Command Line

To get command-line help for Fixed-Point Toolbox objects, type

```
help objectname
```

For example:

```
help fi
help fimath
help fipref
help numerictype
help quantizer
```

To invoke Help Browser documentation for Fixed-Point Toolbox functions from the MATLAB command line, type

```
doc fixedpoint/functionname
```

For example:

```
doc fixedpoint/int
doc fixedpoint/add
doc fixedpoint/savefipref
doc fixedpoint/quantize
```