

Setting Fixed-Point Math Attributes

You control how the fixed-point toolbox implements fixed-point arithmetic by setting the fimath properties. By setting these properties appropriately, you can control the fixed point toolbox to reflect how the operations are being performed in hardware. This can be set in Matlab using the fimath() function:

F = fimath(...'PropertyName',PropertyValue...)

You can add these properties to your fixed point data type by including it as an additional argument to fi():

a = fi(v,s,w,f,F) returns a fixed-point object with value v, signed property value s, wordlength w, fraction-length f and embedded.fimath F.

Matlab Documentation can be found at:

Matlab -> Help -> Documentation -> Fixed-Point Toolbox -> Fixed-Point Basics -> Fixed-Point Data Representation and Arithmetic -> Set Fixed-Point Math Attributes

Matlab fixed point documentation (pdf's) can also be found on the D2L site

Properties that can be set:

The properties that can be set are (**properties in bold are useful for hardware**):

- 1) **CastBeforeSum** — Whether both operands are cast to the sum data type before addition. Possible values of this property are
 - a) 1 (cast before sum)
 - b) 0 (do not cast before sum)
 - c) The MATLAB factory default value of this property is 1 (true).
 - d) This property is hidden when the SumMode is set to FullPrecision.

- 2) **OverflowAction** — Action to take on overflow. The value of the OverflowAction property can be one of the following strings:
 - a) Saturate — Saturate to maximum or minimum value of the fixed-point range on overflow.
 - b) Wrap — Wrap on overflow. This mode is also known as two's complement overflow.
 - c) The MATLAB factory default value of this property is Saturate.
- 3) **ProductBias** — Bias of the product data type
- 4) **ProductFixedExponent** — Fixed exponent of the product data type
- 5) **ProductFractionLength** — Fraction length, in bits, of the product data type. This value can be any positive or negative integer. The product data type defines the data type of the result of a multiplication of two fi objects. The MATLAB factory default value of this property is 30.
- 6) **ProductMode** — Defines how the product data type is determined. In the following descriptions, let A and B be real operands, with [word length, fraction length] pairs $[W_a \ F_a]$ and $[W_b \ F_b]$, respectively. W_p is the product data type word length and F_p is the product data type fraction length.
 - a) FullPrecision — The full precision of the result is kept. An error is generated if the calculated word length is greater than MaxProductWordLength.

$$W_p = W_a + W_b$$

$$F_p = F_a + F_b$$

- b) KeepLSB — Keep least significant bits. You specify the product data type word length, while the fraction length is set to maintain the least significant bits of the product. In this mode, full precision is kept, but overflow is possible. This behavior models the C language integer operations.

$$W_p = \text{specified in the ProductWordLength property}$$

$$F_p = F_a + F_b$$

- c) **KeepMSB** — Keep most significant bits. You specify the product data type word length, while the fraction length is set to maintain the most significant bits of the product. In this mode, overflow is prevented, but precision may be lost.

$$W_p = \text{specified in the ProductWordLength property}$$

$$F_p = W_p - \text{integer length}$$

where

$$\text{integer length} = (W_a + W_b) - (F_a + F_b)$$

- d) **SpecifyPrecision** — You specify both the word length and fraction length of the product data type.

$$W_p = \text{specified in the ProductWordLength property}$$

$$F_p = \text{specified in the ProductFractionLength property}$$

- e) The MATLAB factory default value of this property is FullPrecision.

- 7) **ProductSlope** — Slope of the product data type
- 8) **ProductSlopeAdjustmentFactor** — Slope adjustment factor of the product data type
- 9) **ProductWordLength** — Word length, in bits, of the product data type. This value must be a positive integer. The product data type defines the data type of the result of a multiplication of two fi objects. The MATLAB factory default value of this property is 32.
- 10) **RoundingMethod** — Rounding method.
- a) **Ceiling** — Round toward positive infinity.
- b) **Convergent** — Round toward nearest. Ties round to the nearest even stored integer. This is the least biased rounding method provided by Fixed-Point Toolbox software.
- c) **Zero** — Round toward zero.
- d) **Floor** — Round toward negative infinity.
- e) **Nearest** — Round toward nearest. Ties round toward positive infinity.

f) Round — Round toward nearest. Ties round toward negative infinity for negative numbers, and toward positive infinity for positive numbers.

g) The MATLAB factory default value of this property is Nearest.

11) **SumBias** — Bias of the sum data type

12) **SumFixedExponent** — Fixed exponent of the sum data type

13) **SumFractionLength** — Fraction length, in bits, of the sum data type

14) **SumMode** — Defines how the sum data type is determined In the following descriptions, let A and B be real operands, with [word length, fraction length] pairs $[W_a \ F_a]$ and $[W_b \ F_b]$, respectively. W_s is the sum data type word length and F_s is the sum data type fraction length.

Note: In the case where there are two operands, as in $A + B$, *NumberOfSummands* is 2, and $\text{ceil}(\log_2(\text{NumberOfSummands})) = 1$. In $\text{sum}(A)$ where A is a matrix, the *NumberOfSummands* is $\text{size}(A,1)$. In $\text{sum}(A)$ where A is a vector, the *NumberOfSummands* is $\text{length}(A)$.

a) FullPrecision — The full precision of the result is kept. An error is generated if the calculated word length is greater than **MaxSumWordLength**.

$$W_s = \text{integer length} + F_s$$

where

$$\text{integer length} = \max(W_a - F_a, W_b - F_b) + \text{ceil}(\log_2(\text{NumberOfSummands}))$$

$$F_s = \max(F_a, F_b)$$

b) KeepLSB — Keep least significant bits. You specify the sum data type word length, while the fraction length is set to maintain the least significant bits of the sum. In this mode, full precision is kept, but overflow is possible. This behavior models the C language integer operations.

$$W_s = \text{specified in the } \text{SumWordLength} \text{ property}$$
$$F_s = \max(F_a, F_b)$$

- c) KeepMSB — Keep most significant bits. You specify the sum data type word length, while the fraction length is set to maintain the most significant bits of the sum and no more fractional bits than necessary. In this mode, overflow is prevented, but precision may be lost.

$$W_s = \text{specified in the SumWordLength property}$$

$$F_s = W_s - \text{integer length}$$

where

$$\text{integer length} = \max(W_a - F_a, W_b - F_b) + \text{ceil}(\log_2(\text{NumberOfSummands}))$$

- d) SpecifyPrecision — You specify both the word length and fraction length of the sum data type.

$$W_s = \text{specified in the SumWordLength property}$$

$$F_s = \text{specified in the SumFractionLength property}$$

- e) The MATLAB factory default value of this property is FullPrecision.

15) SumSlope — Slope of the sum data type

16) SumSlopeAdjustmentFactor — Slope adjustment factor of the sum data type

17) **SumWordLength** — Word length, in bits, of the sum data type. This value must be a positive integer. The sum data type defines the data type of the result of a sum of two fi objects. The MATLAB factory default value of this property is 32.