

[Accelerate](#) /  / [vImage.PixelBuffer](#) / `multiply(by:divisor:preBias:postBias:destination:)`

Instance Method

`multiply(by:divisor:preBias:postBias:destination:)`

Multiplies each pixel in an 8-bit planar pixel buffer by the specified factor.

iOS 16.0+ | iPadOS 16.0+ | Mac Catalyst | macOS 13.0+ | tvOS 16.0+ | visionOS | watchOS 9.0+

```
func multiply(  
    by factor: Int,  
    divisor: Int,  
    preBias: Int,  
    postBias: Int,  
    destination: vImage.PixelBuffer<vImage.Planar8>  
)
```

Available when Format is `vImage.Planar8`.

Parameters

factor

The multiplication factor.

divisor

A value that the function divides the result by. The function treats 0 as 1.

preBias

A value that the function adds to the source before multiplication.

postBias

A value that the function adds to the result after multiplication.

destination

The destination pixel buffer.

Discussion

This function applies the following operation to each pixel:

```
destination = (((source + preBias) * factor) + postBias) / divisor
```

For example, the following code multiplies each pixel value in an 8-bit planar buffer by 2 and adds 5:

```
let buffer = vImage.PixelBuffer<vImage.Planar8>(
    pixelValues: [10, 20, 30, 40,
                  50, 60, 70, 80],
    size: vImage.Size(width: 4,
                      height: 2))

buffer.multiply(by: 2,
                divisor: 1,
                preBias: 0, postBias: 5,
                destination: buffer)

// Prints
// [ 25, 45, 65, 85,
//   105, 125, 145, 165 ]
print(buffer.array)
```

See Also

Scalar Multiplication

```
func multiply(by: Float, preBias: Float, postBias: Float, destination: vImage.PixelBuffer<vImage.PlanarF>)
```

Multiplies each pixel in a 32-bit planar pixel buffer by the specified factor.