

[Accelerate](#) / vDSP

API Collection

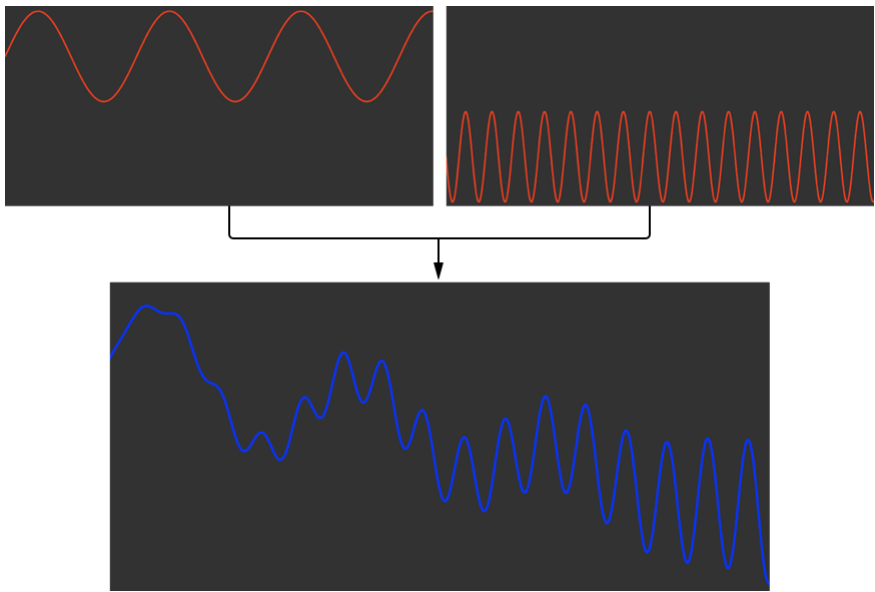
vDSP

Perform basic arithmetic operations and common digital signal processing (DSP) routines on large vectors.

Overview

The vDSP library contains a collection of highly optimized functions for DSP, type conversion, and general purpose arithmetic on large collections. The library includes DSP operations such as convolution and correlation, Fourier transformation, and biquadratic filtering. For arithmetic on large collections, vDSP includes functions such as multiply-add and reduction functions including sum, mean, and maximum.

The following sequence of images illustrates an example of the vDSP library's capabilities. The `vDSP_vtmrg` function combines two waveforms to produce a vector to create a smooth transition between two signals.



Note

Unless otherwise mentioned, vDSP functions with the same input and output sizes (in bytes) work in-place.

The majority of vDSP operations are single-threaded and run on a single core. However, the following functions may be multithreaded depending on the size of the data they're operating on:

- [vDSP_mmul](#)
 - [vDSP_mmulD](#)
 - [vDSP_zmma](#)
 - [vDSP_zmmaD](#)
 - [vDSP_zmms](#)
 - [vDSP_zmmsD](#)
 - [vDSP_zmmul](#)
 - [vDSP_zmmulD](#)
 - [vDSP_zmsm](#)
 - [vDSP_zmsmD](#)
-

Topics

Fundamentals

 Controlling vDSP operations with stride

Operate selectively on the elements of a vector at regular intervals.

 Using vDSP for vector-based arithmetic

Increase the performance of common mathematical tasks with vDSP vector-vector and vector-scalar operations.

Swift overlay

enum vDSP

An enumeration that acts as a namespace for Swift overlays to vDSP.

⌵ vDSP Protocols

Protocols that support Swift implementations of vDSP operations.

Vector generation, filling, and clearing

⌵ Vector generation

Populate vectors with ramps, values from lookup tables, interpolated values, and window functions.

⌵ Vector clear and fill functions

Populate vectors with zeros or a scalar value.

Vector reduction

⌵ Vector extrema calculation

Calculate the minimum and maximum values in a vector.

⌵ Vector average calculation

Calculate the average value in a vector.

⌵ Vector summation

Sum the values in a vector.

Vector geometry functions

⌵ Vector distance and Pythagorean computation

Calculate distance and hypotenuse of vectors.

⌵ Dot product calculation

Calculate the scalar product of two vectors.

Element-wise vector arithmetic

Perform basic arithmetic operations on vectors that contain real and complex values.

⌵ Arithmetic operations

Perform operations on large vectors.

Vector-scalar arithmetic

≡ Vector-scalar real arithmetic functions

Perform element-wise operations on combinations of vectors of real values and scalar values.

Vector-vector arithmetic

≡ Vector-vector real arithmetic functions

Perform element-wise operations on vectors of real values.

≡ Complex basic arithmetic

Perform elementwise operations on vectors of complex values.

≡ Integer arithmetic

Perform elementwise operations on vectors of integer values.

≡ Linear averaging functions

Calculate the element-wise linear average of two vectors.

≡ Polynomial evaluation

Evaluate polynomials using coefficients and independent variables that you supply.

Vector operations

≡ Compression and gathering functions

Compress vectors based on the nonzero elements in a gating vector, or gather vectors based on a separate vector that contains indices.

≡ Copying, element swapping, and merging functions

Copy, swap, and merge the elements of two vectors.

≡ Reversing and sorting functions

Perform in-place reverse and sort operations on a vector.

Vector interpolation

≡ Linear interpolation functions

Compute the linear average between two vectors or between the neighboring elements in one vector.

≡ Quadratic interpolation functions

Compute the quadratic interpolation between the neighboring elements in a vector.

Vector filtering

≡ Biquadratic IIR filters

Apply biquadratic filters to single-channel and multichannel data.

≡ Single-channel biquadratic filters

Filter a single-channel signal with a cascade of biquadratic sections.

≡ Multichannel biquadratic filters

Filter a multichannel signal with a cascade of biquadratic sections.

≡ Finite impulse response filters

Perform finite impulse response filtering with decimation and antialiasing on vectors of real or complex values.

≡ Recursive filters

Perform two-pole two-zero recursive filtering on a vector.

Vector conversion functions

≡ Conversion to decibel equivalents

Convert vectors that contain power or amplitude data to decibels.

≡ Type conversion

Perform element-wise floating-point to integer and integer to floating-point conversion.

≡ Complex vector conversion

Perform element-wise split-complex to interleaved and interleaved to split-complex conversion.

≡ Polar-rectangular conversion

Convert each element of a vector between radius-angle and Cartesian pairs.

Single-vector arithmetic functions

≡ Absolute and negation functions

Compute the absolute or negated value of each element in a vector.

≡ Integration functions

Compute the running sum, Simpson, or trapezoidal integration of a vector.

- ⌵ Clipping, limit, and threshold operations
Apply clipping, limit, or threshold rules to the elements in a vector.
- ⌵ Normalization functions
Compute the mean and standard deviation of a vector and calculate new elements to have a zero mean and a unit standard deviation.
- ⌵ Phase computation functions
Calculate the element-wise phase values, in radians, of a complex vector.
- ⌵ Complex conjugation functions
Calculate the complex conjugate of the elements in a vector.
- ⌵ Vector squaring functions
Compute the square, signed square, or squared magnitude of the elements in a vector.
- ⌵ Fractional part extraction
Truncate the elements of a vector to a fraction.
- ⌵ Zero crossing search
Count and find the zero crossings in a vector.

vDSP_zvsma

vDSP_zvsmaD

vDSP_zvzsm1

vDSP_zvzsm1D

Single-vector sliding-window operations

- ⌵ Sliding-window reduction functions
Calculate maximum values and sums of values in a sliding window.

Vector-to-vector spectra and coherence computation

- ⌵ Autospectrum computation
Compute the element-wise sum of the squares of the real and imaginary parts of a complex vector.
- ⌵ Cross-spectrum computation
Compute the element-wise product of a vector and the conjugate of a second vector.

- ⌵ Coherence function computation
Compute the coherence of two vectors.

Vector-to-vector extrema functions

- ⌵ Vector-to-vector minima and maxima
Compute the element-wise minimum or maximum values or magnitudes in a vector.
- ⌵ Extrema finding functions
Extract the values from a vector that fall outside a range.

Matrix operations

- ⌵ Multiplication
Multiply vectors that contain real or complex values.
- ⌵ Transposition
Transpose vectors that contain real values.
- ⌵ Matrix and submatrix copying functions
Copy the contents of a submatrix to another submatrix.

Vector and matrix correlation and convolution

- ⌵ 1D correlation and convolution
Use correlation to compare and convolution to combine vectors of real or complex data.
- ⌵ 2D convolution
Perform convolution operations on matrices of real data.

Vector and matrix Fourier transforms

- ⌵ Fast Fourier transforms
Transform vectors and matrices of temporal and spatial domain complex values to the frequency domain, and vice versa.
- ⌵ Discrete Fourier transforms
Transform vectors of temporal and spatial domain complex values to the frequency domain, and vice versa.

☰ Discrete Cosine transforms

Transform vectors of temporal and spatial domain real values to the frequency domain, and vice versa.

Data types

`typedef` `vDSP_Length`

An unsigned-integer value that represents the size of vectors and the indices of elements in vectors.

`typedef` `vDSP_Stride`

An integer value that represents the differences between indices of elements, including the lengths of strides.

`struct` `DSPComplex`

A structure that represents a single-precision complex value.

`typedef` `COMPLEX_SPLIT`

`struct` `DSPDoubleComplex`

A structure that represents a double-precision complex value.

`typedef` `DOUBLE_COMPLEX_SPLIT`

`struct` `DSPSplitComplex`

A structure that represents a single-precision complex vector with the real and imaginary parts stored in separate arrays.

`struct` `DSPDoubleSplitComplex`

A structure that represents a double-precision complex vector with the real and imaginary parts stored in separate arrays.

`struct` `VectorizableDouble`

A structure that represents a double-precision real value for biquadratic filtering and discrete Fourier transforms.

`struct` `VectorizableFloat`

A structure that represents a single-precision real value for biquadratic filtering and discrete Fourier transforms.

Constants

The version of vDSP at compile time.

Macros

`vDSP_DeprecateTranslations`

`vDSP_ENUM`

See Also

Signal Processing Essentials

📄 Controlling vDSP operations with stride

Operate selectively on the elements of a vector at regular intervals.

📄 Using linear interpolation to construct new data points

Fill the gaps in arrays of numerical data using linear interpolation.

📄 Using vDSP for vector-based arithmetic

Increase the performance of common mathematical tasks with vDSP vector-vector and vector-scalar operations.

📄 Resampling a signal with decimation

Reduce the sample rate of a signal by specifying a decimation factor and applying a custom antialiasing filter.