



**SINEVIBES**

DSP Algorithm Collection  
**Reference Manual**

# Matrix Reverb

Reverb effect based on feedback delay network. Includes two versions: 4 delay lines with a 4x4 feedback matrix, and 8 delay lines with a 8x8 feedback matrix.

`MatrixReverb8x8.h` (`MatrixReverb4x4.h`) + `DampingFilter.h` + `InlineMath.h` + `LowCutFilter.h` + `SimpleDelay.h` + `SimpleLF0.h` + `SinCosTable.h`

---

## MAIN CALLS

`void` `setMaxBlockSize( int inBlockSize )`

Defines the maximum possible block size the algorithm will be able to process. By default, it's set to 16384 sample frames. Must be called before `setSampleRate()`.

- `inBlockSize` block size, frames.

`void` `setSampleRate( float inSampleRate )`

Initialize the class.

- `inSampleRate` sample rate, Hz.

`void` `processBlock( float *inBlockL, float *inBlockR, float *inPreDelayTime, float *inSize, float *inDecay, float *inModFrequency, float *inModDepth, float *inLowDamping, float *inHighDamping, int inInterpolation, float *outBlockL, float *outBlockR, int blockSize )`

Process one mono/stereo sample block.

- `*inBlockL` left channel input
- `*inBlockR` right channel input - or `NULL` for mono processing
- `*inPreDelayTime` pre-delay time, seconds (0.0..1.0)
- `*inSize` reverb space size (0.0..1.0)
- `*inDecay` reverb tail decay time (0.0..1.0)
- `*inModFrequency` modulation frequency, Hz (0.05..5.0)
- `*inModDepth` modulation depth (0.0..1.0)
- `*inLowDamping` low-pass damping filter frequency, Hz (20..20.000 Hz)
- `*inHighDamping` high-pass damping filter frequency, Hz (20..2.000 Hz)
- `inInterpolation` interpolation type:
  - linear (0)
  - spline (1)
- `*outBlockL` left channel output
- `*outBlockR` right channel output - or `NULL` for mono processing
- `blockSize` number of frames to process (must be the same for all input/output blocks).

## OPTIONAL CALLS

`void` `reset()`

Zero any internal buffers or state variables.

## USAGE EXAMPLE

```
#include "MatrixReverb8x8.h"

// create a class instance for the 8x8 version
MatrixReverb8x8 mMatrixReverb;

// initialize with host's maximum block size and current sample rate
mMatrixReverb.setMaxBlockSize( hostMaxBlockSize );
mMatrixReverb.setSampleRate( sampleRate );

// process one stereo sample block, use spline interpolation
mMatrixReverb.processBlock( inputL, inputR, preDelay, size, decay, modFrequency,
modDepth, lowDamping, highDamping, 1, outputL, outputR, blockSize );
```