

Spl-Lab Measurement Library (M-Lib)

Measurement Library (M-Lib) is a specialized software library designed for performing measurements and calculations using Spl-Lab equipment. It provides an API interface for integrating measurement functions into custom applications.

General Information

- The library must be loaded and remain in memory throughout the session.
 - If the library is unloaded, initialization and device setup functions must be called again after reloading.
 - All core functions require prior execution of Init and SetDevice.
-

Core Functions

◆ Init — Library Initialization

Initializes the library and activates the license. Must be called **before starting any operations**.

function Init(Secure: String; Owner: String): Integer; external 'MLib.dll' name 'Init';

- Secure — license key string
- Owner — organization name

Returns:

Integer — error code (0 if successful)

◆ GetDeviceList — Retrieve Device List

Detects audio input/output devices and connected Spl-Lab hardware. Should be called **after initialization** to obtain the list of available devices.

function GetDeviceList(var devPlayList, devRecordList, devUsbList: TStringList): Integer; external 'MLib.dll' name 'GetDeviceList';

- devPlayList — list of playback devices
- devRecordList — list of recording devices
- devUsbList — list of connected Spl-Lab devices

Returns:

Integer — error code (0 if successful)

◆ SetDevice — Set Active Devices

Specifies the playback, recording, and Spl-Lab devices to be used. Must be called **before executing measurement functions**.

function SetDevice(playDev, recDev, usbDev: String): Integer; external 'MLib.dll' name 'SetDevice';

- playDev — selected playback device name
- recDev — selected recording device name
- usbDev — selected Spl-Lab device name

Returns:

Integer — error code (0 if successful)

Measurement Functions

◆ DelayFinder — Delay Measurement

Measures time shift between audio signals at different frequencies. The result can be returned as time, distance, or phase angle.

function DelayFinder(qChannels, mesUnit, timeShift: Integer; freqMask: array of Byte; var resLeft, resRight: array of Double): Integer; external 'MLib.dll' name 'DelayFinder';

- qChannels — number of channels (1 — left only, 2 — left and right)
- mesUnit — measurement unit:
1 — time (ms), 2 — distance (cm), 3 — angle (degrees)
- timeShift — initial delay in milliseconds (used to compensate for signal path latency)
- freqMask — frequency mask array (16384 values: 1 — include, 0 — skip). Index corresponds to frequency. Allows flexible selection of frequency range.
- resLeft, resRight — result arrays for each channel. If only one channel is used, the second array returns zero.

Returns:

Integer — error code (0 if successful)

◆ RtaMeasure — Frequency Response Measurement

Performs frequency response measurement by playing a test signal and analyzing the result. Returns octave spectrum and array of center frequencies.

function RtaMeasure(Octave: Integer; var octaveRes, freqRes: array of Double): Integer; external 'MLib.dll' name 'RtaMeasure';

- Octave — octave resolution:
0 — 1/1 octave, 1 — 1/2, ... 5 — 1/24 octave
- octaveRes — measured frequency response (dB per octave)
- freqRes — array of center frequencies for each octave band

Returns:

Integer — error code (0 if successful)