Andrew Thornett

From: Andrew Thornett
Sent: 14 February 2026 20:20
To: Andrew Thornett
Subject: Timing Errors in Spectrum Lab

Hello DJ,

The problem may be related to the new soundcard's (or new audio driver's) sampling rate being too far off the nominal value. In most modes, timestamps in Spectrum Lab are based on the number of audio samples read from the input. When the audio-sample-based timestamp differs a lot from the PC's system time, strange things happen. But without knowing any detail, I cannot tell. Depending on the true reason, switching to a different "time source", or (preferred) calibrating the audio device's sampling rate as explained in the SL manual may fix this.

Cheers,
Wolf .

Machine Config: MCONFIG.INI
This is the default location for all machine-dependent settings for Spectum Lab, for example the soundcard sampling rate calibration table.

Also don't miss the sampling rate calibration if you use the program for the first time, or have installed a new soundcard.
If a soundcard really supports a certain sampling rate (like 48000 Hz / x or 44100 Hz /x) is not always easy to tell, because if the sampling rate 'requested' by the application is not really supported by the hardware, the driver software (or the windows multimedia system) jumps in, and tries to interpolate / extrapolate to realize the sampling rate by software. Unfortunately, Windows does a really bad job on some machines. For example, on one of the author's PCs (a Lenovo Z61m), when trying to run the onboard audio device at 12000 samples/second, the signal sounded 'distorted', and the CPU load caused by the 'SYSTEM' process (indicated in the Windows Task manager) went up to 5 Percent, as long as the audio input was running. Switching back to 11025 samples/second cured this problem, the audio sounded 'clean' again, and the CPU load caused by the 'SYSTEM' process went back to zero.
Again, it's not easy to tell which sampling rates are really supported, so if you find strange effects / poor audio quality / dropouts / unexplainably high CPU load, try a different sampling rate - first try

a fraction or multiple of 48000 Hz (like 12000, 16000, 32000, 480000, 96000). If that doesn't work well enough, try a fraction or multiple of 44100 Hz (like 11024, 22050, or 44100), especially on 'old' machines because sampling rates of 44100/N were the standard in the age of Soundblaster and Co. Last not least, it's possible to compensate the sample rate drift continuously, for example for phase measurements or high-accurary frequency measurements.

Sampling Rate Divisor:
Defines the decimation ratio between the analog/digital conversion rate and the processing rate of all following stages (which you can see in the component window, for example frequency converter, digital filter, but also the Spectrum Analyzer).
Reducing the sample rate at this early stage is especially helpful if you run out of CPU power, because -depending on what you are doing- the real-time sound processing threadwhats_a_thread can 'eat up' a lot of the available CPU time. The internal processing rate is:
<Sampling Rate> divided by <Sample Rate Divisor>, for example:
11025 samples per second, divisor set to "2" will give an internal processing rate of 5512.5 samples per second. This would be enough to process audio signals below 2kHz (theoretically 2756 Hz, but this is impossible because of the anti-alias-filter's roll-off).
Note: After this decimation stage, and before the FFT calculation, there may be more decimation stages. See 'FFT Settings'.