A High Quality Open Source DSP Implementation of a Analog Synthesizer Emulation

In this time of software sound synthesizers the quality of the simulation of analog synthesizers is not regarded a very fundamental issue. The working prototype system discussed in this article is a fast responding, straight DSP algorithm, Open Source Midi module, where the traditional analog synthesis building blocks are implemented by DSP algorithms written in the C programming language. The DSP code is a straight implementation of its analog counterpart without FIR-ification or time reordering and with attention for the Nyquist criterion without resampling/filtering. The generators are additive synthesis sawtooth waves with limited spectrum. The filter is a 32 bit based time varying resonant 4 pole IIR implementation.

1. Introduction

Because of the warmness and smoothness and expressiveness of the sound, analog synthesizers and their sounds are still popular in various musical scenes. There are more than a few simulations with digital signal processing, but many of these will not last a second in a one to one comparison. There are a number of reasons for this, sampling issues being among the most important ones. Signal accuracy in the algorithms used for the synthesizer building blocks is another, including the properties of digital circuit noise, which is not averaging like analog circuits are. Finally, for well known and popular types of analog synthesizer blocks the equivalent simulation of the actual electronics, including crosstalk and all the complexities of transistors is not easily done and must be seen in the context of sampling issues, too.

The synthesizer I prototyped doesn't do anything else than a straight but sampling-correct oscillator, envelope generator, "voltage" controlled amplifier and filter, and low frequency oscillators and a multiple varying delay path chorus unit.

No attempt is made to mimic a specific type of classical analog machine, but attention is paid to making the ?platonic? building blocks work straight only. The unit is a physical unit, not computer or hardware simulation experiment, it is housed as prototype in a 19 inch rack unit which has been demonstrated on various occasions.

2. The "Platonic" synthesizer building blocks

Platonic indicates a "pure" and clear concept, like the concept of a equally sided triangle, here an oscillator is an idealized block which generates a certain waveform of exact nature at a frequency which is a function of its input.

Such a digitally controlled oscillator is not the same as its analog equivalent: luckily it doesn't drift, it generates probably less noise, the input can determine the frequency over a mathematically perfect function, and depending a bit on the implementation it is probably cheaper and easier to build.

One problem for normally (linear, equidistant, impulse-) sampled signal paths is the strict observance of the Nyquist rate, where in this case the highest signal frequency component must lie under 48KHz/2. The reconstruction assumption normally is that the perfect impulses are convolved with a good approximation of a sync function in the DA converters reconstruction aka anti-aliasing filter. This condition is almost never really met to my knowledge in audio equipment, and also the good electronicists' practice of therefore even more observing fitting distance of the highest digital signal path frequency to the Nyquist frequency is easily violated.

At least phase freedom of the the higher frequencies is impacted, resulting in a dull and annoying sound when reproduced with good audio amplification.

In this design, the sampling rate problem is attacked from the start: the oscillator, which is supposed to be a sawtooth wave, works by additive synthesis of a number of the first harmonics of the (theoretical) Fourier analysis which are stored in a very oversampled sample memory, and which gets read by the DSP algorithm.

This way, at the start of the signal path, the spectrum is perfectly limited to the Nyquist rate, so that no anti-aliasing filtering is needed to ensure frequency limitation of the 'hard' edges of the waveform.

The filter is a digital equivalent of a 4th order analog filter with (overall) feedback control for making it resonate, and the filter, when the samples are interpreted as impulse samples, adds theoretically no harmonics, so here too, the sampling criterion will be honored.

Of course, the envelope generator with controlled amplifier at fast rates (like the attack section) may well add very noticeable harmonics to the signal, their combination is not filtered or limited in frequency by design.

3. The DSP implementations

Using block representations, the main synth is a midi module with effect unit input and 2 stereo outputs, an a number of controls.

The DSP is a blackfin at 500 MHz bus-coupled with a big CPLD which communicates with a display, an entry knob, control knobs and the MIDI input opto-coupler, all devices are memory mapped.

All the software and the hardware programming is Free and Open Source, and can be freely used and downloaded for non-commercial purposes from

http://www.theover.org/Synth

4. The Resulting System and listening results

Given the limitations, the sound of the system is good, clean, fat, accurate, and the response is very direct. Many traditional synthesizer sounds and new ones can be made.