Faculty of Environment and Technology,
UWE Department of Computer Science and Creative Technology
Audio Process Design and Implementation
UFCFE4-30-2

Assignment 2: Synth Plugin – Guide

Christoph Schick (14010758)
12 April 2016

Summary

It's not as simple as it seems.
In theory many things seem rather simple, until they are to be put into practice.
It's no difference when it comes to developing a synthesiser.
This guide shows how quickly it can become very complex.

Contents

Page no.

1. INTRODUCTION

Programming a simple FM / AM synthesiser is not as easy as it seams. Although both frequency and amplitude modulation are basic modulation techniques, their application in synthesiser development naturally comes with difficulties. Some of which are discussed in this brief guide. This guide further explains the way the synthesiser works and how it is designed.

2. THE SYNTHESIS PROCESS

For the synthesiser at hand the fundamental sound is created using frequency modulation. For which the user can choose between two modulator waveforms. This sound basis is mixed with a lower (half the initial frequency) sine signal and panned from almost wet only to almost dry only over the duration of a second. The amplitude modulation operator then processes the resulting signal. Following the signal flows through the ADSR envelope, a LFO that adds a tremolo and finally a low pass filter. The whole synthesiser runs then through the final high-pass filter that eliminates unwanted rumble.

The synthesiser is designed to deliver a wide range of electronic sounds and timbres. It employs a total of six operators, which are fixed in their position in the process chain. There are two frequency modulator operators, two carrier operators and two amplitude modulator operators.

The user can switch between the two frequency modulators, and the carriers are cross fading from one prominent to another. For the amplitude operators, there is one AM operator, which operates dependent on the fundamental frequency. And one LFO, for the final tremolo effect.

The user has control over several parameters within the process chain. This is explained in detail later.

3. KEY FEATURES AND AUDIO PROCESS

The key features of this synthesiser are for one that the user can choose the waveform of the frequency modulation and to which degree the “wet” and the “dry” signal will be mixed after the first second of the tone.
Then the synthesiser also enables the user to play “lead” notes when notes are played at maximum velocity.
These functions offer a rich spectrum of sounds from dull and deep to screechy and bell-like.
A couple of filters are applied at the end of the processing chain to make the sound over all more pleasant. The raw frequency modulation sounds are fairly rich in high frequency content. And also mirror back from the

low end, resulting in an overloaded low-end area. Hence the global high-pass filter is set fix at 50 Hz whereas the low-pass filter is free for the user to set according to their musical preferences.

4. USE OF PARAMETERS

As mentioned earlier, the user has a wide selection of parameters to control.
In the first row of controls there are the general user controls. Sorted there for the sake of this assignment as the second row of controls is controlled by the modulation data. The bottom row contains controllers over amplitude envelope and the "dry/wet" mix.
The user only has control over selected parameters in the processing chain. Beginning with the scaling of the frequency of the FM modulators ("Tuning" on the UI). Then the user controls the depth of the modulation with the "Richness" control. Obviously there is the "Mod Type" toggle, which switches back and forth between sine and triangle wave for the FM modulator.
The scaling of the AM modulator is also in the hands of the user. Controlled with the "AM Mod" knob. The next relevant component of the process chain is the ADSR envelope. The user has power over all of the stages. Although their ranges are limited to one second, it still provides a flexible tool to alter the timbre of the instrument. The respective controls on the interface are named accordingly.
Two dials control the LFO's rate and depth (named accordingly), and finally there are controls for the low-pass filter's cut off frequency and the output gain.

5. USE OF MODULATION DATA

The modulation data provided in the LogicX project has been used in following ways.
The first mod has proven to be useful as a tremolo control. As such it is mapped to the LFO depth control. This leaves the user merely with setting the rate of preference.
The second mod has not been employed as it wasn't considered useful by the developer.
The third set of modulation data, which – more or less – gradually rises from 0.0 to 1.0 over the duration of the piece, could be used to create a gradual evolution of timbre over the duration of the piece. To do so it is mapped to the synthesiser's "richness" control to gradually raise the richness of the sound produced.
The fourth set of data is mapped to the frequency control for the AM modulator. That is because this creates a clearly audible lead sound. It also introduces a curious effect when the value is changed over the duration of a note.

Finally there is the fifth set of modulation data. It is mapped to the synthesiser's release parameter for the amplitude envelope. This way the automation creates a change in timbre from verse to chorus. When less notes are being played the release is longer, when more notes are played, it is shorter. Thus thinning out the sound in order to not overwhelm the listener when heavy chords are being played.

REFERENCES

Nash, C. (2016) sawWave [C++ class definition]. Available from: https://blackboard.uwe.ac.uk/webapps/blackboard/content/listContent.jsp?course_id=_253378_1&content_id=_4488227_1 [Accessed 10 April 2016].