Harriet Phillips [redacted]

9th April 2020

COMP2300 Synth Part-2 – The Additive Synthesis of a Triangle and a Sawtooth Wave

Overview

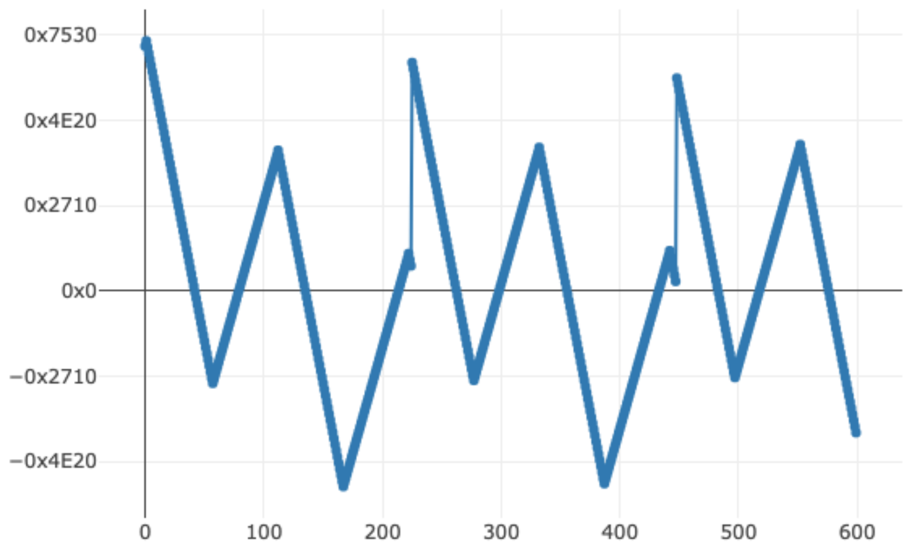
Part-2 of Synth is an implementation of the additive synthesis of a triangle wave and a sawtooth wave. The triangle wave has a frequency of 440Hz and a peak-peak amplitude of 0x4000 to -0x4000. The Sawtooth wave has a frequency of 220Hz and an amplitude of 0x3000 to -0x3000. I chose these particular frequencies as they are harmonics, and chose the amplitudes because they would not add to over 0x7FFF or under 0x8000, which would cause them to leave the signal range.

Implementation

The implementation begins with initialization of the variables needed to calculate the waves.
The number of outputs needed per period is calculated for each wave by dividing the sample rate by the desired frequency. This is divided by half for the triangle wave, as it needs to go from peak to trough and back. The scalar used to calculate the next number in the sequence is calculated by dividing the full range of numbers by the number of outputs required.
The calculations for each wave are broken into two parts. The subtraction loop and the addition loop. Inside the subtraction loop, the wave's scaler is subtracted from the wave's counter to get the next number in the sequence, until the counter reaches the lower limit of the amplitude.
The addition loop for triangle is the reverse of its subtraction loop – it adds the scaler to the counter until the counter reaches the upper limit of the amplitude.
The addition loop for the sawtooth wave resets the counter to the upper limit of the amplitude.
When the triangle's counter is decreasing – i.e. it has not yet reached the lower limit for this cycle, it is in state 0. If it is increasing it is in state 1. This is so that when the sawtooth wave's value has been calculated it can return to the correct loop for the next value to be calculated for the triangle. This is not required for the sawtooth wave, as the triangle only needs to reference it's decreasing loop.

Output

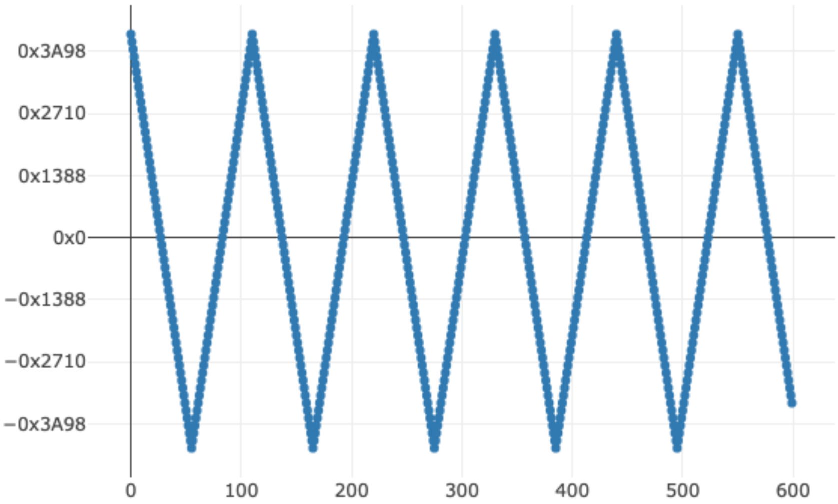
The graphical output of the additive synthesis performed of the wave over 600 samples is:



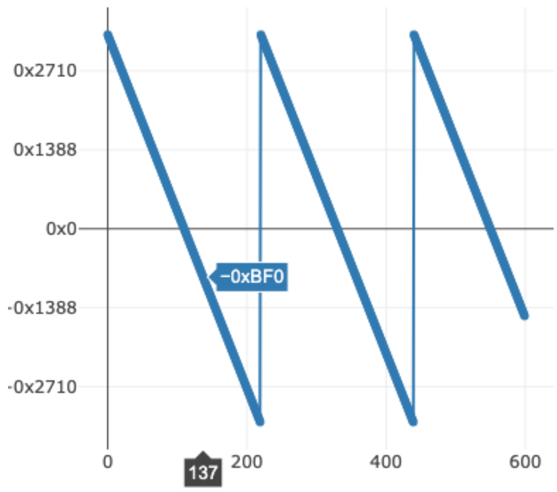
From this graph it is possible to figure out where in the sequence each of the wave's is. For example, you can tell that the sawtooth wave has reached its lower amplitude as the wave's jumps amplitude jumps up at ~220 samples.

Harriet Phillips, [redacted]
9th April 2020

The separate graphs of the two waves are:
Triangle wave:



Sawtooth wave:

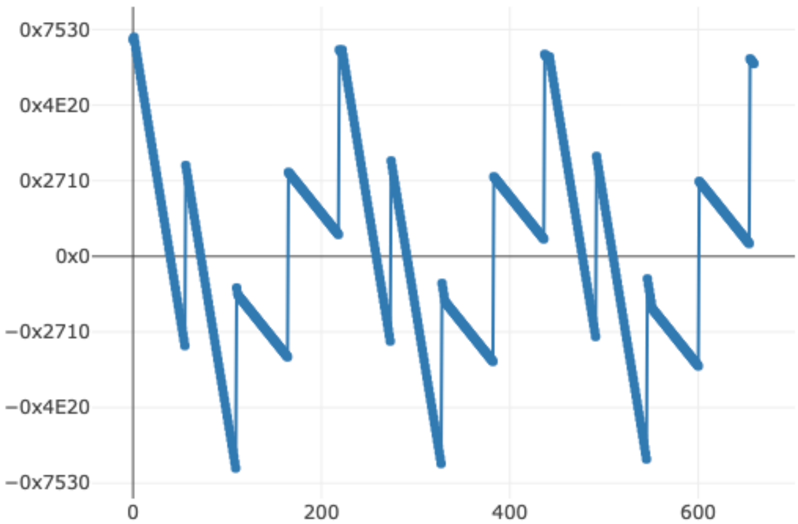


Reflection:

Initially, I started investigating the additive synthesis of two triangle waves. The main problem I came across when doing this was making sure the triangles would go back to the right spot in the other triangle's algorithm after they finished executing their calculation. This is why I implemented triangle states. Originally, there was a state for each triangle, however it was not necessary to have a state for the sawtooth wave. The rest of the implementation wasn't particularly difficult, as I had already programmed a triangle wave for part-1.

Once I had this working, I decided it might be interesting to see what the additive synthesis of a triangle and sawtooth wave would look like. I found programming the sawtooth wave was actually easier than the triangle wave, as there was no need for states, and the addition "loop" was very simple.

I also experimented with a few different frequencies for each, which output some interesting graphs. For example, this graph results when the sawtooth has a 880Hz frequency and the triangle wave has a 220Hz frequency.



In the future, I would be interested in adding more waveforms to the additive synthesis wave. As I haven't had much use of using the discoboard's memory, I ran out of registers to use and couldn't store the variables required for more waveforms. If I had more practice with using memory I could store the values required for different waves and load them each successively to create an additive synthesis wave form from them.