

Synthesis Recipe

Electric Guitar

The waveform of an electric guitar string shortly after being plucked, is similar to a waveform somewhere between a triangle and a sawtooth wave (figure 1). The specifics depend on where we pluck the string. We can generate this waveform using a pulse-width shifted triangle wave, for which we can use the opcode **vco2**. Refer to the lecture notes from Week 5 for how to create a (PWM) triangle wave. The suggested pulse width shift is 0.1.



Figure 1: Waveform of an electric guitar.

The sound will begin decaying, both in amplitude and brightness almost immediately after being plucked. This can be implemented using an amplitude envelope and a lowpass filter envelope (try **butlp** for the lowpass filter).

We can add a fixed bandpass filter to brighten the sound and give it character (try **butbp**). This filter may reduce the overall level, so you may need to upscale the overall amplitude somewhere in your code after this filter (just multiply it by a number greater than 1).

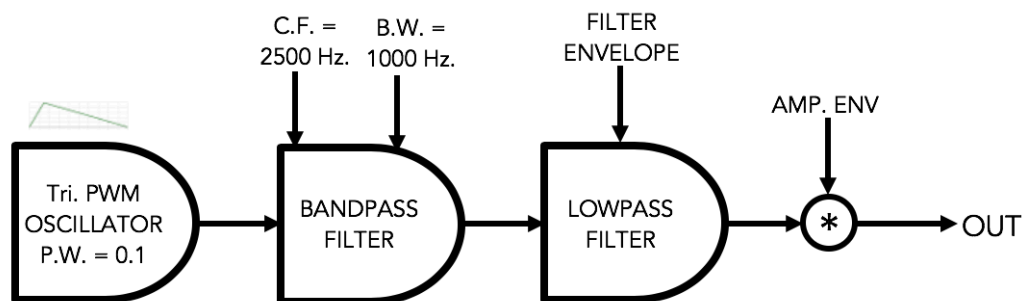


Figure 2: Schematic for an electric guitar string

Envelopes

The amplitude envelope (AMP.ENV) can be a simple single segment exponential decay (figure 3).

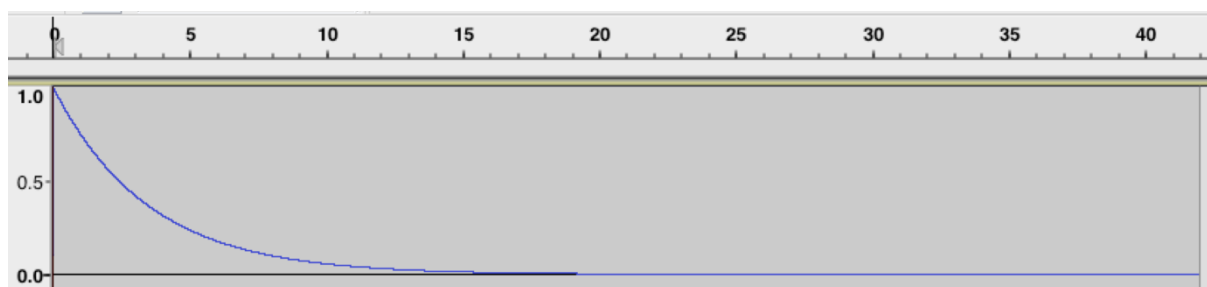


Figure 3: Amplitude envelope (start value 1, end value 0.0001)

A note duration of 40 seconds is reasonable for a low E (note number 40). Use this as the duration of your entire envelope.

The filter envelope (FILTER ENVELOPE) can be a slightly more elaborate multi-stage envelope with exponential segments (use a-rate). A short fast decay stage (figure 4) gives the sound more of a pluck-like attack.

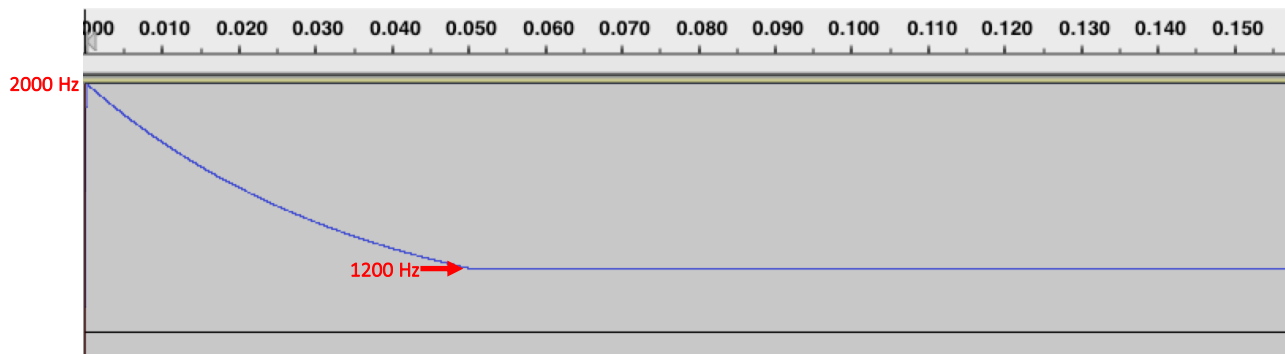


Figure 4: Filter envelope (attack stage detail)

The remainder of the filter envelope is a longer, gradual decay. The ending value is 2 Hz.

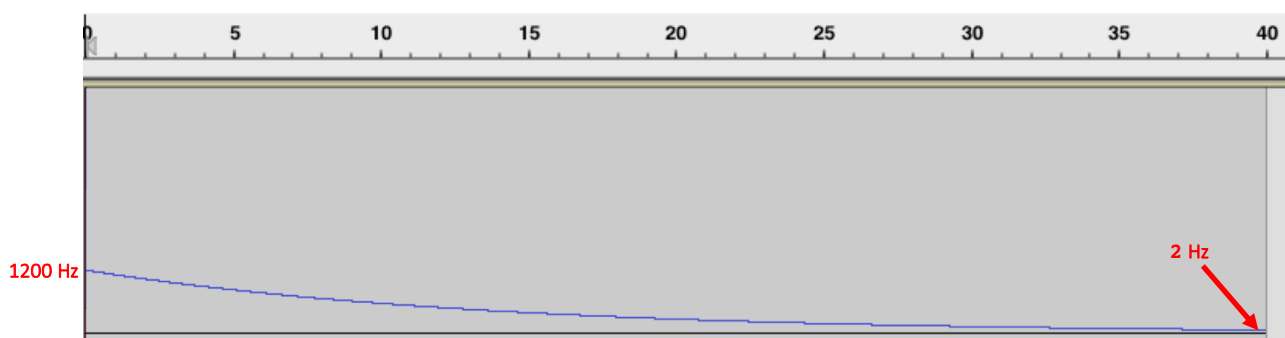


Figure 5: Filter envelope (complete). The attack stage shown in figure 4 is still there, it's just not visible

Going Further

- Typically, higher notes will decay quicker, and lower notes will decay more slowly. You can implement this by making your envelope decay times inversely proportional to the frequency of the note. You could make note 40 the pivot point about which envelopes will either be shortened or lengthened according to the note played. Example: *Week3 -> 05_Dish02.csd* does something like this with the variable it calls **iMult**.
- Demonstrate your instrument with some notes, melodies or chords from the Csound score, or make it a live instrument, playable from a MIDI keyboard.
- See if you can add a separate sound element that imitates the sound of the plectrum striking the string.
- Add some reverb, delay or distortion (**clip** opcode).