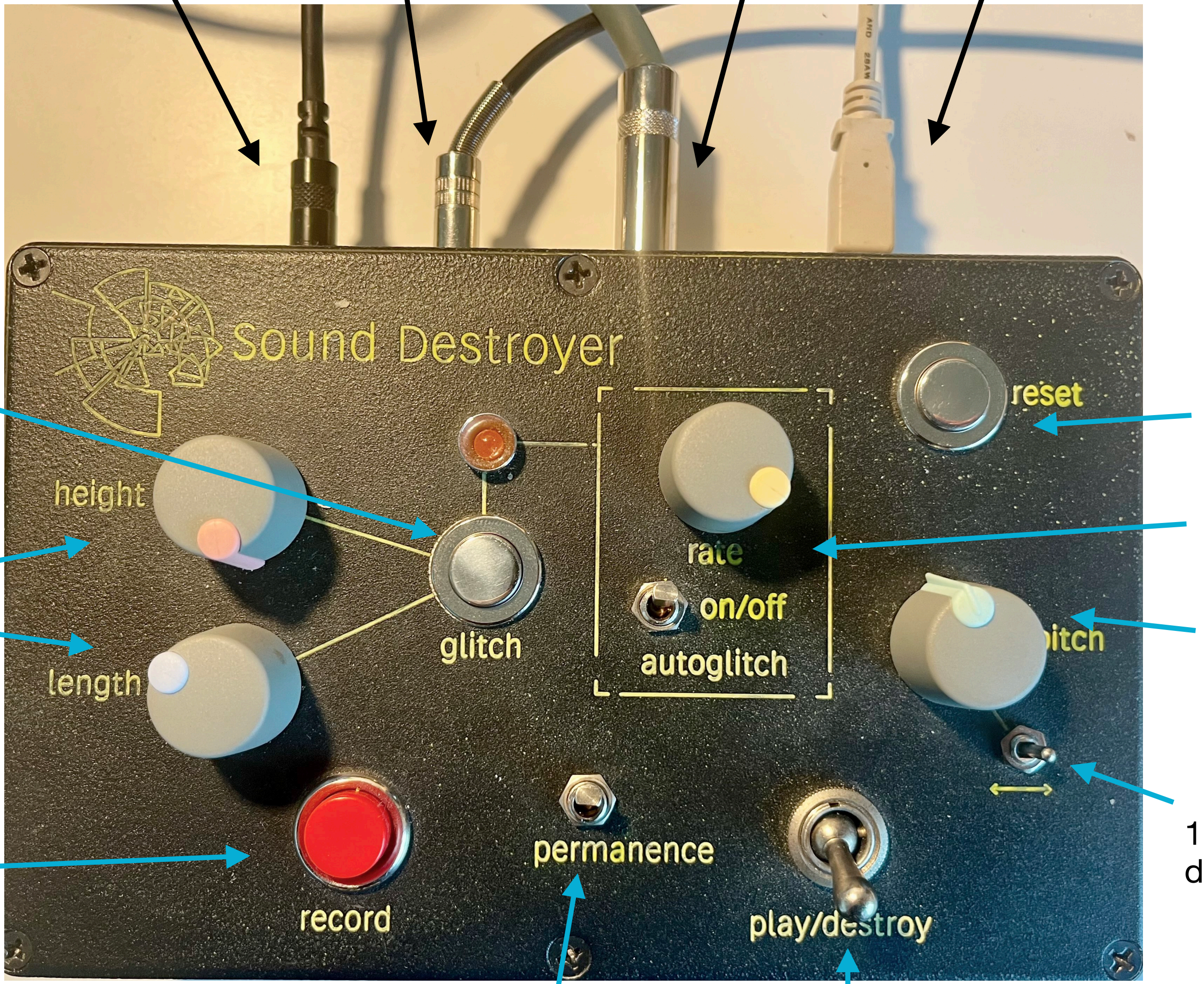
Sound Destroyer: the manual



1. Audio Output

2. Audio Input

3. Footswitch Input

4. USB Power

5. Glitch Trigger

6. Glitch Settings

7. Audio Sample Record Button

8. Permanence Switch

9. Play/Destroy Switch

10. Playback direction

11. Pitch

12. AutoGlitch system

13. Reset wavetable

Yann Seznec / November 2024

Welcome to the Sound Destroyer!

This device records audio input and gives you a set of controls for destroying the sound you have recorded. It does this using a looping wavetable playback system - essentially, the playback of the sound is controlled through a separate graph. That graph can then be "glitched". The resulting audio output can then be overwritten to the original, making any changes permanent.

Inside the box is a Bela board, running a Pure Data patch. If you'd like to see how this works on your computer, you can download the code here.

Some notes on this as a prototype object:

This current version runs on a Bela board and is using wires and panel mount components. It is therefore very much a prototype and thus has a two main quirks to be aware of.

Quirk #1: Startup time

This device takes about 30 second to start working. To start, plug in the power using the USB socket and be patient. The light may flash when ready, or maybe not. The only way to know if it's really working is to try it out.

Quirk #2: Switch/knob states on startup

Usually, when you plug in the instrument the settings should be working the way they look. However I'm not certain if the switches in particular are always set properly on startup. That means, for example, that the "autoglitch" switch may be in the "on" position, but not actually be *on*. It is worth flipping the switches back and forth once on startup just to make sure, in case anything doesn't seem to be working as expected. This is possible with the knobs as well, though much less likely.

1. Audio Output

3.5mm stereo output. Suitable for headphones or out to a mixer or recorder or interface. There is no volume control on the device, if you want volume control you'll need to do that externally.

2. Audio Input

3.5mm stereo input. You can also use a mono jack input. This is like a high gain line level input, so you can use a dynamic microphone directly in, or send signal from a mixer. When a sample is recorded it is automatically normalised, so weaker signals will still be audible but the quieter the signal the higher the resulting noise. There is no gain control on the device.

3. Footswitch Input

This is a normal instrument cable input (like a guitar cable). Use the provided cable and footswitch in this socket if desired. When plugged in, the footswitch acts the same as the record button (7). Audio is recorded while the switch is held down, and recording stops when the switch is released.

4. USB Power

Old-style USB port for power. Most phone-charger style USB power supplies should work. It can also be connected to a power bank for battery power, or connected to a laptop. In case of code updates, this can be connected to a computer. There is no power switch. Plugging in the power is enough to turn it on (see *Quirk #1*).

5. Glitch Trigger button

Pressing this button introduces a single “glitch” into the playback wavetable. The light should flash to indicate a new glitch has been written. The glitch is generated in accordance with the current glitch settings (6).

6. Glitch Settings controls - “height” and “length”

These knobs control the range of potential values that will be used to generate a glitch. “Length” is the potential length of the glitch, and “Height” is the potential strength of the glitch. These knobs are not setting the exact values, but rather they are setting the maximum range of the random values that might be generated. Thus even if the knobs are never moved, pressing the Glitch button will still generate different glitches each time. For details on this, see “*How the glitches work*”.

7. Audio Sample Record Button

Pressing and holding this button will record the current audio coming into the input (2). Releasing the button will stop recording. The longest potential sample is 100 seconds. This is recorded in stereo at 44.1khz. The length of the playback loop will be the length of time this button was held down.

8. Permanence switch

This switch selects whether the audio that is being played is also simultaneously overwriting into the buffer. With this switch activated, any glitches that are introduced into the playback are made permanent in real time and the original sound can not be reinstated. Generally speaking, if this switch is activated then the sample will eventually disappear, though this can take many forms depending on other settings.

9. Play/Destroy

This switch plays and stops the looping sample. If the “permanence” switch (8) is activated, any destructive changes you hear will be permanently overwritten. The playback will loop the current sample length.

10. Playback Direction

This plays the sound sample forwards or backwards. If you record a sample while it is set to “backwards”, then it will be reversed (I think?).

11. Pitch

This sets the playback pitch of the wavetable (and thus, the resulting sound). There is no “reset” to this, so if you want the original pitch you just have to tweak it a little bit at a time until it gets there.

12. AutoGlitch system

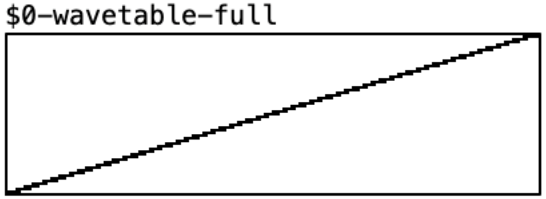
The on/off switch here activates AutoGlitch, in which a glitch is automatically introduced into the wavetable at regular increments. The rate of glitching is set by the knob. The glitch light will flash to indicate each time a glitch has been introduced. This is the same as if you were to press the “glitch” button regularly. Each time a glitch is introduced it is set according to the glitch settings (6). The autoglitch rate is relative to the length of the sample, from 1-10x per loop.

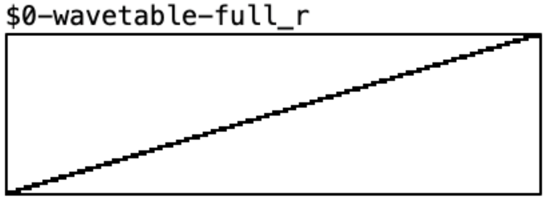
13. Reset

This button resets the wavetable to a straight line and thus a “normal” playback. If the permanence switch has *not* been activated, pressing this button will allow you to hear your original sample (perhaps at a different pitch, according to how you have set that knob). If the permanence switch *has* been activated, the reset button will still reset the wavetable, but it will be playing back a loop that has been permanently overwritten so it will likely still sound glitchy.

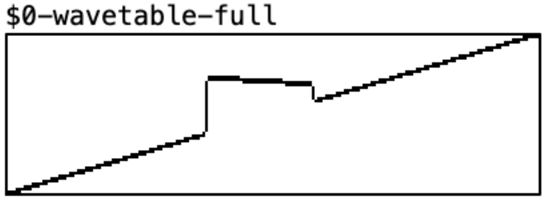
How the Glitches Work

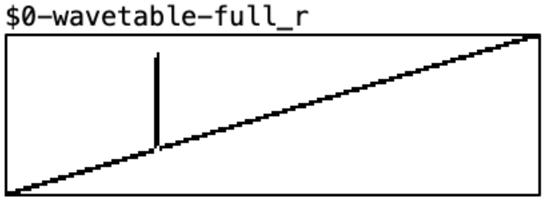
The looping of the audio is achieved using *wavetable playback*. This means that there is a graph that sets how the sound file is played back. With a stereo file, there are two of these. For a standard, straightforward, glitch-free playback, we use a diagonal line from lower left to upper right. The *x* axis here represents the time of playback (the length of the recorded sample) and the *y* axis represents the location of the sample being played back.



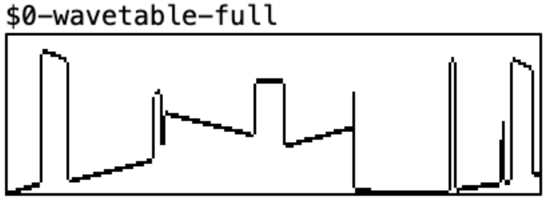


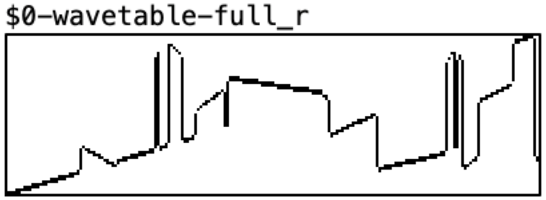
When a glitch is introduced, either with the button or the autoglitch system, a single glitch is introduced into each of these graphs, one for left and one for right. The shape of these glitches is determined by a set of random numbers, the range of which are set by the "length" and "height" knobs.





Pressing the button multiple times, or using the autoglitch system, will result in wavetables with quite a variety of glitches. Different settings on the knobs will result in different types of glitch sounds. Great "height" values tend to result in more drastic sounding glitches, while longer "lengths" can often create glitches that slow the pitch down considerably. Combining the two in different ways generates often surprising results, but you can learn to guess what the outcomes might be like.





Pressing the "reset" button sets the wavetable back to the unglitched diagonal line you see at the top.

Yann Seznec / November 2024