

# CUT GLOVE

by Rodrigo Constanzo

<http://www.rodrigoconstanzo.com>

v01



## INSTRUCTION MANUAL

# DESCRIPTION AND OVERVIEW

The Cut Glove is a live sampling and performance instrument built using Max. It was programmed around the Xbox 360 controller. Additionally, the patch has MIDI learn functionality built in, so every parameter can be mapped to any MIDI controller.

# SYSTEM REQUIREMENTS

Requires [Max7](#) or later and the externals listed below. (The patch will work in Max6 but the GUI will not look right)

## EXTERNALS USED

*(all are included in the download)*

Alex Harker externals:

<http://www.alexanderjharker.co.uk/>

Sigmund~ for pitch tracking in “Pitch” module:

[http://www.maxobjects.com/?v=objects&id\\_objet=4713](http://www.maxobjects.com/?v=objects&id_objet=4713)

Individual externals:

karma~ for sampling/playback

framerank~ for mp3ification

morphfilter~ for IR manipulation

## XBOX 360 DRIVER

<https://github.com/d235j/360Controller>

## NEW IN THIS VERSION (v01)

-everything!!

# TABLE OF CONTENTS

Input/Output/Info/Record Section	6
Setup Window	7
Input FX	8
Output FX	10
karma Modules	15
xbox Mapping	21
Troubleshooting	22

# MODULE & GUI BREAKDOWN

The GUI was built to be very minimal and utilitarian. It is not intended to be used while performing, as you don't need to interface with things on screen while you play.



# INPUT/OUTPUT/INFO/RECORD SECTION



This section has three main parts. The bottom left is your input section. The number boxes, and corresponding meters are the first four inputs of your soundcard. If you only have two inputs, only the first two will appear active (as in the screenshot).

The drop-down menu (currently showing “adc”) is where you select whether you want to use live input, or file playback (or test tone/vst). Open/play are for the file playback, although when in vst mode, open opens the current vst instead.

Clicking on the word ‘input’ opens a window that lets you add convolution (equipment emulation) to the different inputs as well as an overall input/output EQ. There is also a chromatic tuner in this section

On the bottom right we have the output section. Here you set the thru level, and the level of all sampler modules.

On the top left you find the master output level/meter and the revamped “Setup” section (explained in detail in the next chapter).

The record button lets you record the summed output of Cut Glove to a file. When you press it you are presented with a file dialog window. Name your file and hit ‘OK’ and you will see the meter going, indicating that you are recording audio.

The ‘info’ button gives a general overview of the patch and mappings.

# SETUP WINDOW

The image shows a 'SETUP' window with a dark theme. At the top left is the title 'SETUP' in large white letters. At the top right is a purple button labeled 'ON'. Below the title bar, there are several rows of settings, each with a label and a value or button. The settings are: DRIVER (Core Audio), INPUT (Built-in Microphone), OUTPUT (Built-in Output, I/O), IO VECTOR (256), SIGNAL VECTOR (16), 34% CPU, SAMPLE RATE (44100), OVERDRIVE (ON), INTERRUPT (OFF), OSC (IP 192.168.2.2), RECEIVE (8000), SEND (9000), and MIDI LEARN (MIDI-learn, clear).

SETUP		ON
DRIVER	Core Audio	
INPUT	Built-in Microphone	OUTPUT Built-in Output I/O
IO VECTOR	256	SIGNAL VECTOR 16 34% CPU
SAMPLE RATE	44100	OVERDRIVE ON INTERRUPT OFF
OSC	IP 192.168.2.2	RECEIVE 8000 SEND 9000
MIDI LEARN	MIDI-learn	clear

Here you have pop-up window for all the DSP, IO settings.

The top section of the window mirrors the settings found in the Max Audio Setup window. You can find more information about the settings in the Max documentation.

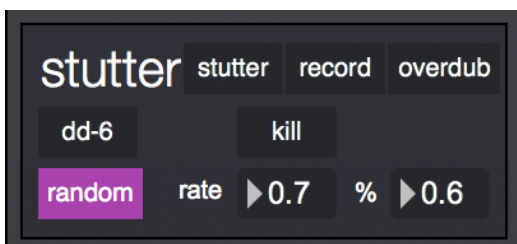
The OSC section lets you send your IP address and receive/send ports. The MIDI learn section lets you map parameters by hovering over them.

# DSP FX

There are a total of four FX slots. The top two are input fx (pre-loop) and the bottom two are output fx (post-loop). Any effect can be loaded into any slot (including multiples of the same effect) by clicking on the name of the effect to bring up a dropdown menu showing the optional fx modules.



## STUTTER



“Stutter” is a stutter/glitch module with two modes. The ‘dd-6’ mode, which does time domain stuttering and is modeled after the [Boss DD-6](#) guitar pedal in ‘hold’ mode. The other one is an FFT mode that does spectral freezing. Both modes are always running so you can switch between them even after creating a stutter/glitch. To change modes press the button in the bottom left.

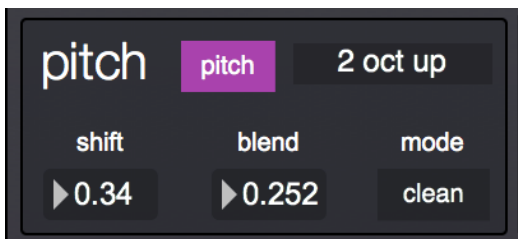


(Stutter cont..)

To activate the Stutter module you must first record a loop. The ‘record’ button records audio for the duration that it is depressed, so it is best to map it to a momentary button. Then the ‘stutter’ button turns on the audio.

“Kill” mode mutes incoming audio when there is a glitch playing. Finally ‘random’ turns on/off the record function at a rate and probability set by the rate and density controls.

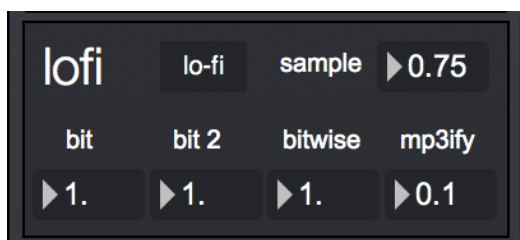
## PITCH



The “Pitch” module uses a delay-based pitch shifting technique to produce somewhat natural, somewhat glitchy sounding pitch shifting. I am aiming for a ‘whammy pedal’ type glitchiness, but it is difficult to emulate.

The preset window (showing “2 down/2 up”) lets you select ranges for the ‘shift’ parameter. These are also inspired by the Whammy pedal. ‘Shift’ controls the pitch shifting in relation to the range of the selected pitch preset. ‘Blend’ controls the dry/wet levels. The ‘dirty/clean’ button affects the pitch tracking. When set to ‘clean’ the window size changes only on new attacks. When set to ‘dirty’ the window changes all the time, so you get lots more artifacts.

# LOFI

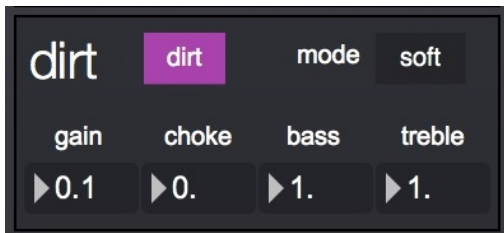


The “Lofi” module is a combination of signal degradation modules. ‘Sample’ controls traditional sample rate reduction, while ‘bit’ & ‘bit 2’ control bit rate reduction using different techniques that allow for fractional bit rate reduction. Both techniques sound slightly different, but allow for a wide range of sounds when combined.

‘Bitwise’ does bit flip/replacement processing and sounds very grungy. Even though this is controlled by a single float value, it is eight discreet steps.

The new ‘mp3ify’ control does some FFT processing on the signal to allow you to dynamically control the psycho-acoustic compression very similarly to how mp3 compression works. When combined with the other signal degradation parameters, this adds a very distinctive and unique twist to the lofi sound world. Note that this parameter behaves ‘inverted’ compared to the other ones in this module. For other parameters lowering the value ‘reduces’ the fidelity; for mp3ify increasing the value turns up the mp3 compression.

# DIRT



The “dirt” module is a distortion module with two modes. The ‘soft’ mode is based on ‘faust’ code by Julius O. Smith and uses cubic nonlinear distortion. It has a milder ‘overdrive’ type sound. The ‘hard’ mode is a variable-hardness clipping function by Laurent de Soras. It is, as the name implies, a harsher distortion.

The ‘gain’ control is the overall level of distortion applied to the signal. ‘choke’ effects the harmonic content and gating of the distortion and varies on the mode (soft/hard) being used. Bass and treble are both cut filters so when set at maximum, the signal remains unaffected.

# SHUFFLER



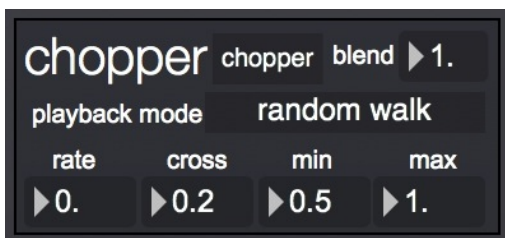
“Shuffler” is a dual mode buffer shuffler. In ‘shuffle’ mode it randomly plays back modulated audio from two prerecorded buffers with the ‘pitch’ controlling how severe the playback speed modulation is.

In ‘spindown’ mode, it takes the last audio fragment being played and repeats it while lowering playback speed, sounding similar to a record being stopped.

(Shuffler cont..)

“Kill” mode mutes incoming audio when there is a glitch playing. Finally, ‘random’ turns on/off the record function at a rate and probability set by the rate and density controls.

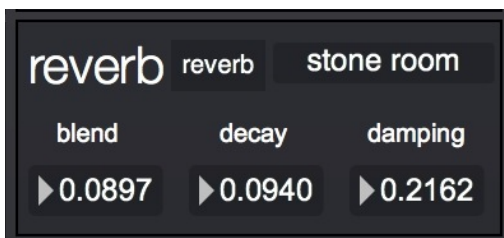
## CHOPPER



The “Chopper” module is based on the example patch of the same name included in the Max6 examples folder. It segments audio based on zero crossings and has options to choose how these are played back using the different playback modes. These define how the tiny fragments are played back with each having a characteristic sound to it. The ‘rate’ controls the rate of audio playback. ‘cross’ controls how many positive zero-crossings are stored per segment and ‘min’/‘max’ are the minimum and maximum length of each segment.

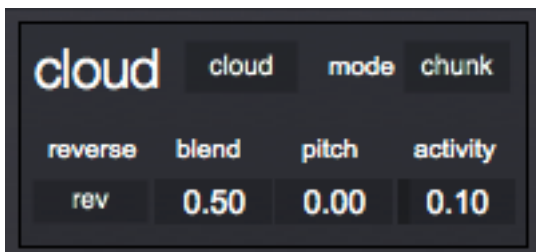
Blend controls the wet/dry level of the effect.

# REVERB



The “Reverb” module is a convolution reverb with 24 built in impulse responses ranging from rooms to halls to noise. ‘blend’ controls the wet/dry level, ‘decay’ processes the impulse response to make the reverb shorter (there’s a small pause in the reverb audio while this is calculated) and ‘damping’ removes some of the high frequencies simulating a more muted acoustic.

# CLOUD



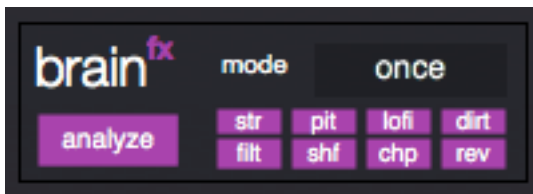
The ‘cloud’ module is an automatic accompaniment module. When the module is engaged, every time there is an onset detected, a short amount of audio is recorded into a buffer AND a different (random) section of the buffer is played back. In ‘chunk’ mode, a single larger section of the buffer is played back each time there is an onset detected. In ‘cloud’ mode, several shorter sections of the buffer are played back creating a pseudo real-time granular synthesis type sound.

The ‘reverse’ parameter determines whether or not some of the fragments that are played back will be reversed.

(Cloud cont..)

The 'blend' parameter sets the level of the dry/wet signals. A setting of 0.5 is both at equal volume. The 'pitch' control sets how varied in pitch (up and down) the playback grains will be. The 'activity' parameter sets how often the onsets can be detected, how short the grains are, and how many grains are played back per onset while in 'cloud' mode.

## BRAIN

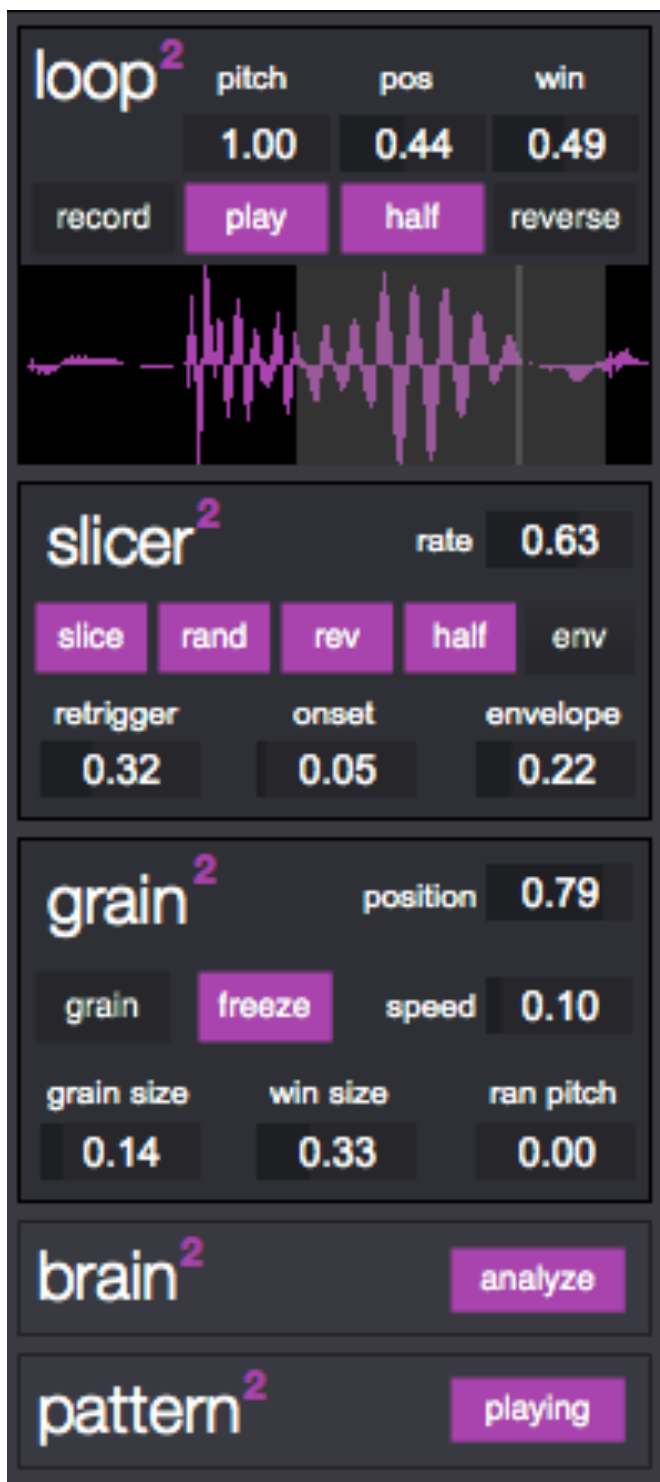


The 'brain' module controls both the input and output effects. It works by analyzing incoming audio and creating dynamic presets for each of the effect parameters.

There are multiple modes for the 'brain' module. 'once' means the preset will be generated only once, until the patch is restarted. 'every' changes the preset every time the effect is turned on. The 'live' modes control the parameters in realtime by controlling the individual parameters, the effect on/off state, or both.

You can individual turn on/off the eight main effects, though when loading new fx modules, each one's Brain setting is set to 'on' by default. Any new fx modules that are added (ie 'Cloud') will not have discreet controls for turning them on/off.

# THE KARMA MODULES



There are four identical ‘karma’ modules which are looper/ samplers with different buffer playback/processing options.

# LOOP



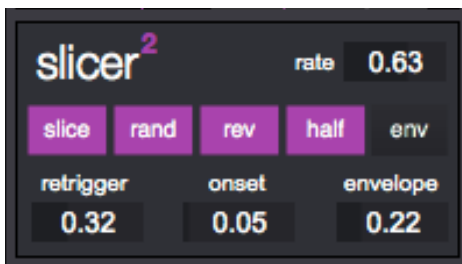
The “loop” module is a dynamically sized (as in you start end the loop whenever you want) varispeed looper at it’s core. It is built around karma~, a custom sampling/playback looper external designed by myself, and coded by raja. It is modelled after the sampler/looper in [The Party Van](#), which is modelled after the “[Where’s The Party At](#)” 8-bit sampler/looper, which is somewhat modelled on the [Line 6 DL-4](#) looper.

The pitch control works almost as a ‘clock’ control. Any changes in pitch made while recording are heard when playing back. While recording you can also engage half-speed and reverse, including reversing ‘past zero’ which just makes the loop jump to the highest point you’ve written in the buffer up to that point.

The ‘position’/‘window’ controls control the start/end points of the loop.



# SLICER



Like all the ‘karma’ modules, the “Slicer” module is linked in with other modules. You can start creating a loop with the ‘slicer’ engaged and it will jump around the buffer writing tiny slices into it, creating a pointillistic/glitchy loop.

The basic functionality of the “Slicer” is to jump around the buffer (either written, or while recording) at regular or random intervals (controlled by the ‘random’ button). ‘Rate’ controls the overall speed of this process. The ‘reverse’ toggle allows for slices to be played back in reverse as well as forward (this is independent of the global ‘reverse’ controlled in the “Loop” module). The ‘half’ reverse does the same, but for half-speed playback. The ‘envelope’ control adds a linear fade out to each slice playback which adds an almost percussive sound to each slice. The ‘envelope’ parameter controls how long the duration of that envelope is.

‘Retrigger’ controls how likely the slicer is going to play back the same slice again. This can add a ‘retriggery’ kind of sound to the slicer module. And finally ‘onset’ determines how likely the slicer module is to jump to an onset point in the sample. This only works with samples that have at least 10 onsets defined.

# GRAIN



Like it's name would imply, “Grain” is a granular synthesis module. It is based on [sugarSynth](#) by Nobuyasa Sakonda.

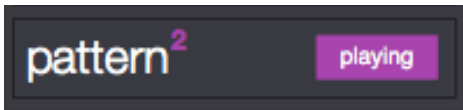
The ‘position’ parameter controls the position of the grain playback engine in the sample buffer. This is tied to the playhead of the “Loop” module in such a way that you can turn on the “Grain” module at any point and it will start granulizing the sound present at playhead position.

‘Freeze’ & ‘speed’ control how “Grain” moves through a buffer. If ‘freeze’ mode is engaged “Grain” will just stay put and continue to spit out grains. If ‘freeze’ is turned off, the granular synth will move through the buffer as controlled by the ‘speed’ control. A ‘speed’ setting of 1 would be 100% playback speed.

‘Grain size’, ‘window size’, and ‘random pitch’ control the granular synth engine and do exactly what their name implies. ‘Grain size’ controls the size of each grain of audio. ‘Window size’ controls the size of the buffer window that grains are played back from and ‘random pitch’ randomizes the playback pitch of each grain.

“Grain” and “Slice” can be used in combination to jump around the buffer randomly while doing granular synthesis on that portion of the buffer.

# PATTERN

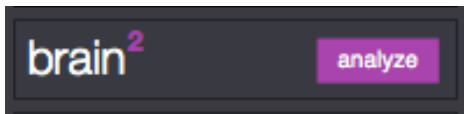


The “Pattern” module lets you record, and loop, any parameter changes in the ‘karma’ section. This effects xbox controller button presses, and analog controller parameter controls. It is coded in a way that allows you to make a pattern out of absolutely anything in this section, including creating loops.

The “Pattern” mode will allow you to start/stop a pattern for as long as you want but once you press ‘record’ it creates a new one. Note that patterns are not saved with the patch, so when you load the patch there is no pattern loaded.

The xbox mapping for this control creates a new pattern when engaged. It is not possible to stop, and restart the same pattern from the controller. This is intentional, although sometimes not ideal.

# BRAIN

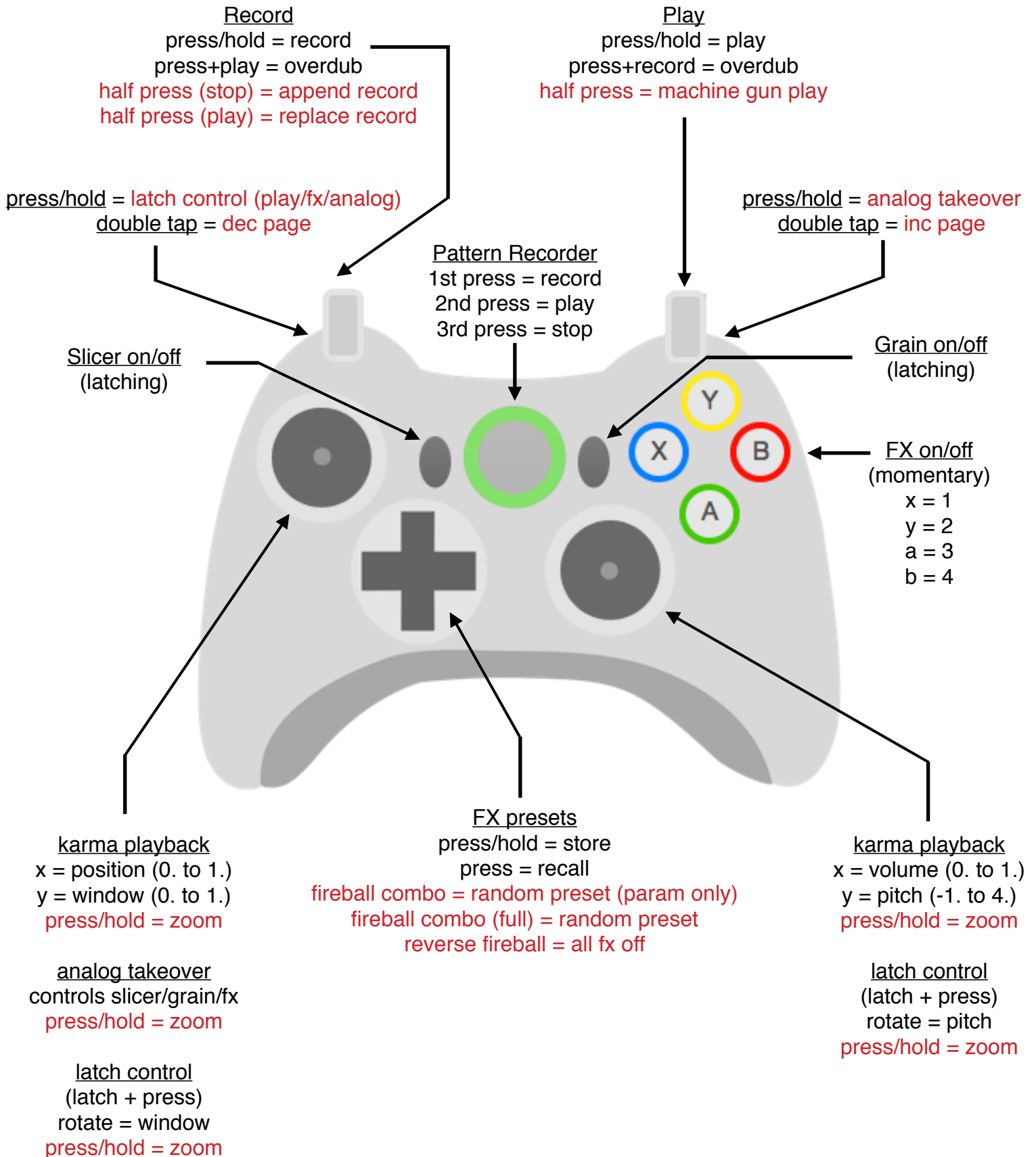


The “Brain” module is the ‘backseat driver’ of the whole ‘karma’ section of the patch. It uses complex\* buffer analysis to change all the parameters of the “Slice”, & “Grain” modules. When the ‘analyze’ button is engaged, every time you create a new loop, the “Brain” module will change the above mentioned parameters to something musically/sonically relevant. You can still manually control things, but this is meant as an intelligent preset generator so that when I turn on “Grain” it doesn’t always sound the same, but it isn’t random or unrelated to the contents of the buffer.

I spent a tremendous amount of time coding, tweaking, testing, and updating the analysis algorithms to something I find musically useful.

\*Fifteen different analysis parameters are used and weighed against each other different for each control parameter. Each parameter is also weighted against what it was set to previously.

# XBOX MAPPING



# TROUBLESHOOTING GUIDE

## **“I’ve opened the patch but nothing is happening”**

*Make sure the DSP engine is turned ON and that you have your soundcard and I/O selected properly in the “SETUP” window.*

*Also make sure you have installed the externals in the appropriate folder (Max6/Cycling '74/msp-externals/).*

## **“I can sample and playback but I can’t hear my dry audio”**

*Turn up the “thru” volume in the top right of the main window, but be careful if you are using a microphone as this can cause a feedback loop.*

## **“I’ve connected my xbox controller but it’s not showing up”**

*Make sure you’ve download and installed the xbox 360 driver from <https://github.com/d235j/360Controller>*

## **“My CPU usage is through the roof and causing dropouts”**

*Every module is coded to be as efficient as possible and most use no CPU at all when not in use. This means that you can really push the CPU by having many things going at once. Additionally the pattern recorders are based around a standard msp object which appears to have a memory leak, so starting to record a pattern and not ending it can cause audio dropouts after a while.*

# STUFF

For any questions, comments, bugs, or feature requests contact me at [rodrigo.constanzo@gmail.com](mailto:rodrigo.constanzo@gmail.com)

Special thanks to raja, Alex Harker, Pierre Alexandre Tremblay, Sam Andreae, Barbara Cassatopo, and countless people on the cycling74 and monome forums.

Patch can be downloaded at my webpage:  
<http://www.rodrigoconstanzo.com>