Pocket Batería Manual

for Critter & Guitari Organelle

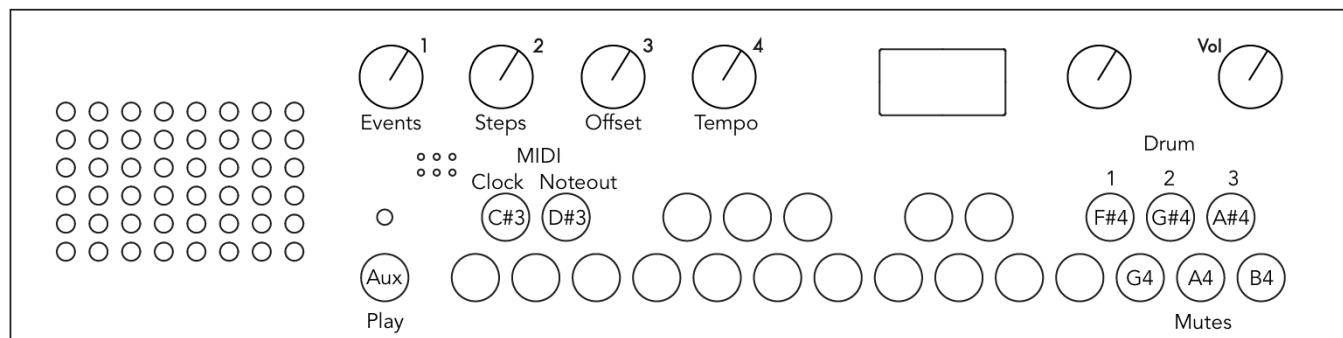Version 0.4, February 2024 by Martin Werner

Overview

Introducing Pocket Batería, a multisample-based Euclidean sequencer that channels the vibrant rhythms of Samba Batucada drum ensembles. However, it's essential to clarify that Pocket Batería is all about capturing the spirit, not cloning or replacing the irreplaceable vibe of a real, human-played Batería.

My main mission with Pocket Batería was to keep things breezy, whether you're laying down beats in the studio or spicing up your live performances. That's why I crammed all the crucial parameters onto just one page—no need to dive into confusing sub-menus. I mean, who likes to get lost in a maze of options when you're in the creative zone, right?

Now, each drum voice in Pocket Batería dances to its own beat, literally. We've got a set of samples that get picked randomly, ensuring that no two hits sound the same in a row. Why? Well, because music should feel as organic and unpredictable as the real deal.

So, in a nutshell, Pocket Batería is your ticket to infusing that Samba magic into your tracks, all wrapped with a UI that's as user-friendly as jamming with your favorite instrument. Get ready to bring the rhythm with a touch of personal flair!

Controls

Aux (or footswitch): Starts playpack and activates MIDI Clock send to external devices (if any are connected)

C#3: Toggles between incoming or outgoing MIDI clock

D#3: Suppresses MIDI note out (does not affect MIDI clock) – by default MIDI note out is turned off and needs to be manually activated if needed.

G4: Mute Drum 1
A4: Mute Drum 2
B4: Mute Drum 3

Muting not only silences audio output but also suppresses MIDI note output for the selected drum voice.

Knob1 + F#4 (hold): Events Drum 1
Knob1 + G#4 (hold): Events Drum 2
Knob1 + A#4 (hold): Events Drum 3
MIDI note out for all 3 drum voices is supported (C3, D3, E3) but suppressed by default (see D#3 above).

Knob2 + F#4 (hold): Steps Drum 1
Knob2 + G#4 (hold): Steps Drum 2
Knob2 + A#4 (hold): Steps Drum 3

Knob3 + F#4 (hold): Offset Drum 1
Knob3 + G#4 (hold): Offset Drum 2
Knob3 + A#4 (hold): Offset Drum 3

Knob4: Tempo from 20-200 BPM (deactivated in MIDI clock receive mode)

Using External MIDI Controllers

If you want to use an external MIDI controller just open the **midi_ctrl_mapping** sub-patch and adapt the controls to your specific needs. By default it has been set up to match a Faderfox MX12 New Module.
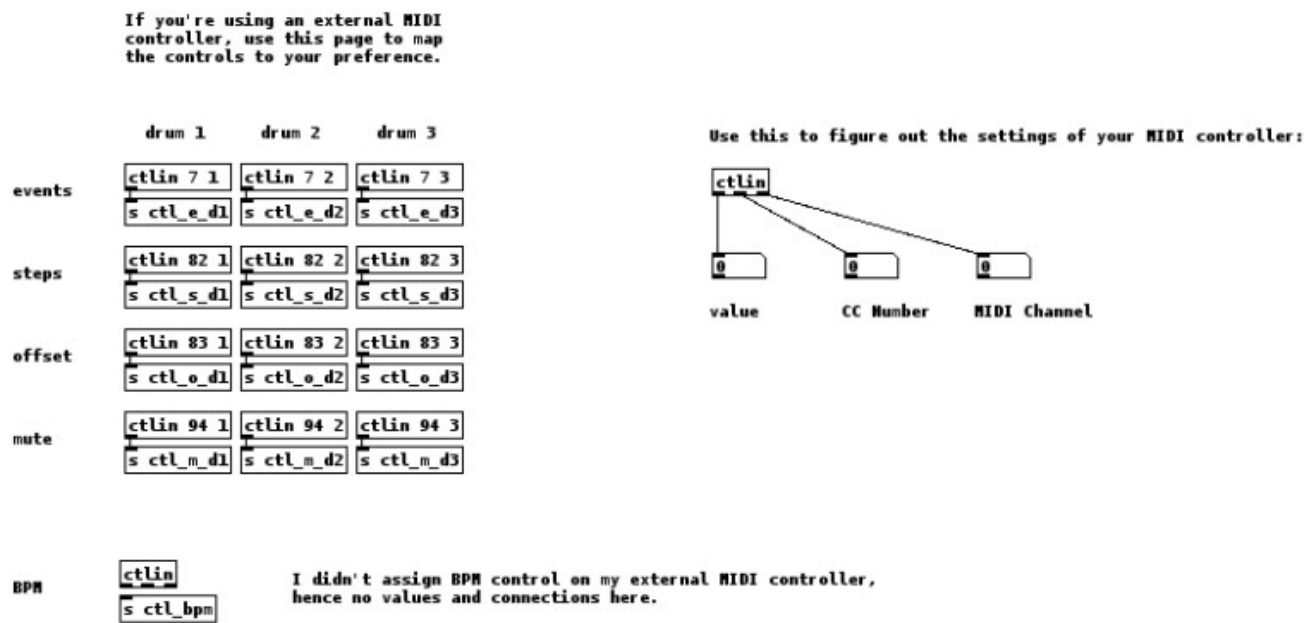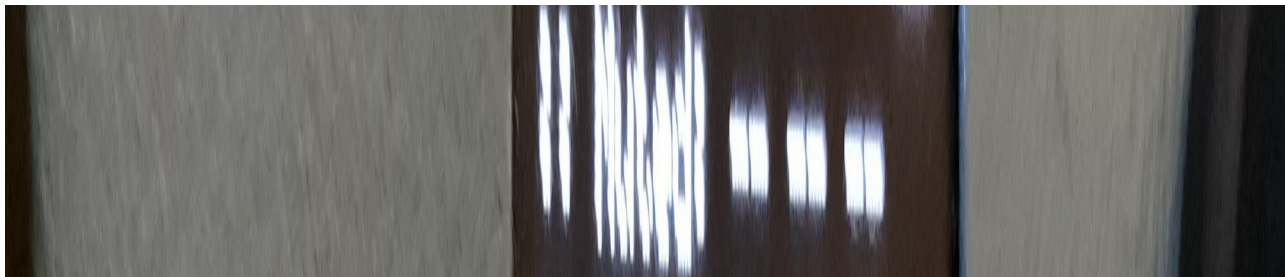
If you're using an external MIDI
controller, use this page to map
the controls to your preference.

	drum 1	drum 2	drum 3
events	ctlin 7 1	ctlin 7 2	ctlin 7 3
	s ctl_e_d1	s ctl_e_d2	s ctl_e_d3
steps	ctlin 82 1	ctlin 82 2	ctlin 82 3
	s ctl_s_d1	s ctl_s_d2	s ctl_s_d3
offset	ctlin 83 1	ctlin 83 2	ctlin 83 3
	s ctl_o_d1	s ctl_o_d2	s ctl_o_d3
mute	ctlin 94 1	ctlin 94 2	ctlin 94 3
	s ctl_m_d1	s ctl_m_d2	s ctl_m_d3

Use this to figure out the settings of your MIDI controller:

ctlin

| value | CC Number | MIDI Channel |

BPM
ctlin
s ctl_bpm

I didn't assign BPM control on my external MIDI controller,
hence no values and connections here.

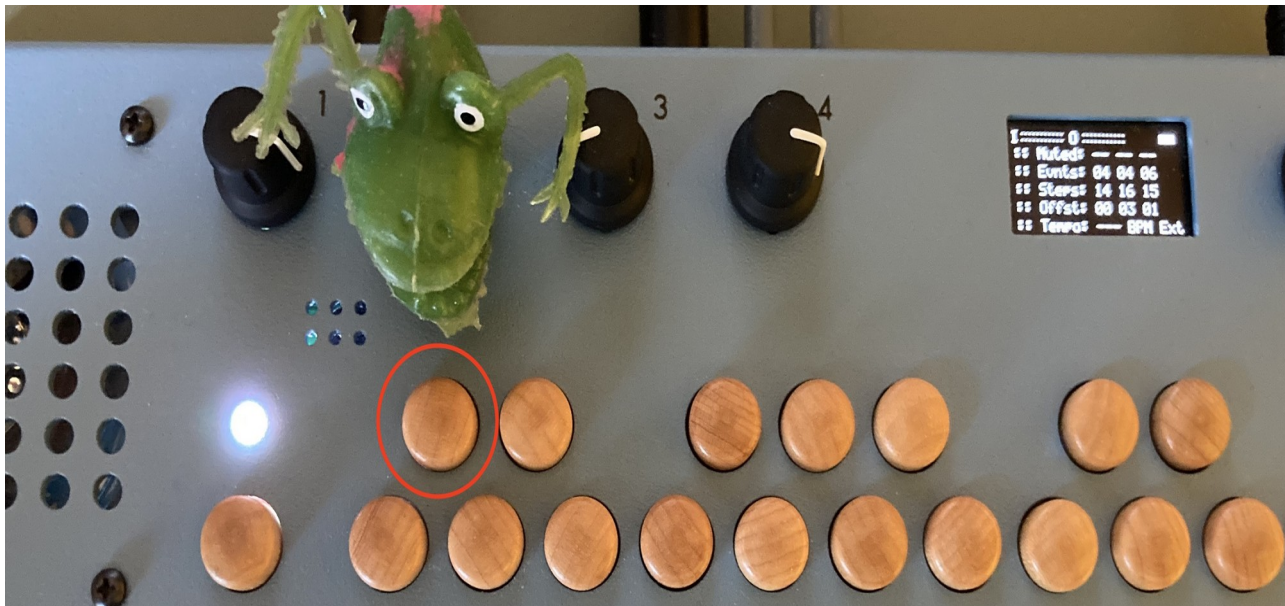*Figure 1: midi_ctrl_mapping sub patch*

MIDI Clock

The Aux button activates playback and concurrently sends MIDI Clock to external devices. The LED turns green, serving as an indicator that MIDI Clock is actively being sent out and not received.

Take note of line 5 on the display: **4: Tempo: 180 BPM** – an additional cue that signifies you are currently in MIDI Clock send mode. While the playback is in progress (indicated by the green LED), toggling for an external MIDI clock is temporarily disabled to avert any potential syncing complications. To ensure a seamless experience and prevent syncing nightmares, it is advisable to halt playback (LED not illuminated) before attempting any clock adjustments.
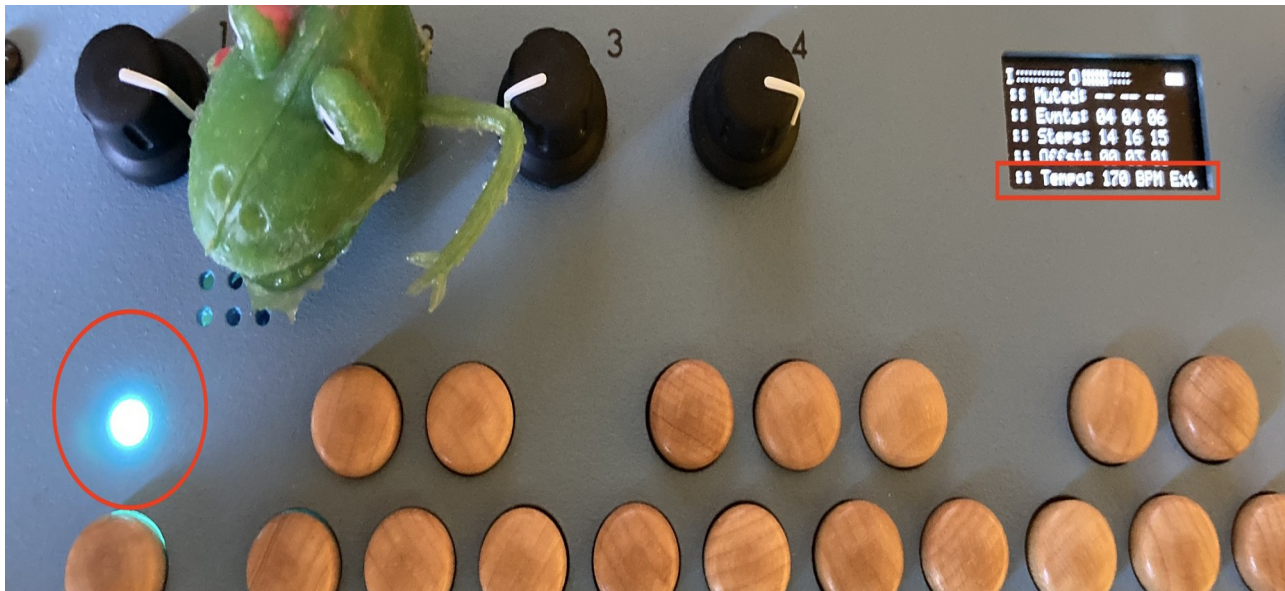
Receive MIDI Clock from external MIDI device (External Device = Host)

The lower C# key functions as a toggle for the external MIDI clock. Upon activation, the LED illuminates in white, signifying the enabling of MIDI Clock reception from external devices.

Observe the alteration in line 5 to :: Tempo: --- BPM Ext – indicating the shift into MIDI Clock receive mode. During this phase, the adjustment of tempo using knob 4 on the Organelle is restricted, as denoted by the locked state. Tempo modifications are exclusively executable on the external MIDI host.

When you hit Play on the external device, watch the LED turn cyan – a cool way of saying, "Hey, I'm syncing up with that external MIDI clock!" Your BPMs from the host device pop up on line 5 as

:: Tempo 170 BPM Ext.

Just a heads-up, if you're tweaking the tempo and it's less than 10 BPM, it might take a beat to show up correctly. We're working on making that smoother in future updates.

Once you stop the groove on the external device, the LED stays cyan, giving you a visual nod that you're still in the external MIDI clock mode. Now, here's the lowdown: You can't hit the aux button to start the internal clock while the external one is partying away – safety first, you know? And the reverse is true too – no toggling the external MIDI clock while the internal one is doing its thing.

Tap the lower C# key, and bam, external MIDI clock takes a bow. Now, you're back in charge, free to switch up the tempo with knob 4. It's like a dance – one step at a time!

Adding your own samples

In case you want to use your own samples – which I highly encourage you to do – there are a few simple rules to follow:
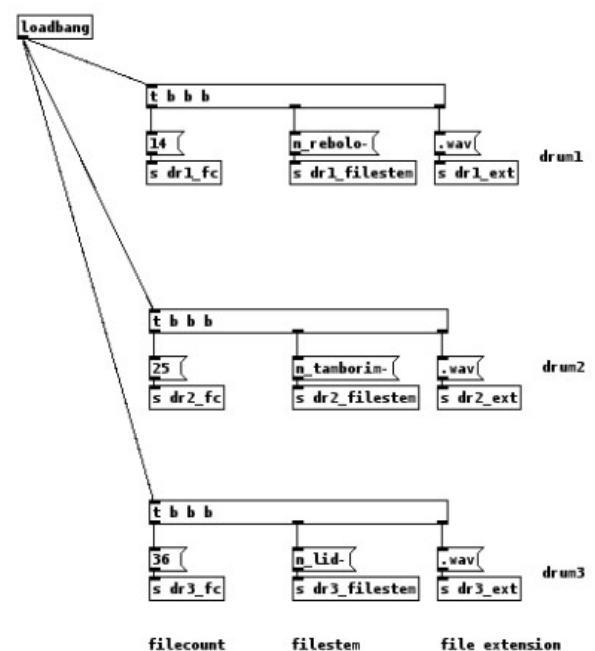
- Samples must be placed in **samples** subfolder.

Best practice in Filenaming

- All samples are required to adhere to specific file naming conventions: Uniformity is key, with each file sharing an identical name within a set. The sole distinguishing characteristic should be a sequential number commencing from 1, ensuring a systematic arrangement.

- Do not use spaces in file names and stick to one file extension per set.

- Avoid leading zeros in the sequence number.

- Don't mix up file extensions – use wav files only!

- Use at least a sample set of two samples per drum voice. The random file selection will not work with a single sample per drum

voice as it prevents the same sample from playing twice in a row.

In the soundfile_mapping sub-patch you have to store your filestem, extension and the count of your sample sets. This is necessary for the random sample selection process to work. Check the contents of the samples subfolder and comments in the sub-patch for examples and best practice.

The soundfile_mapping sub-patch