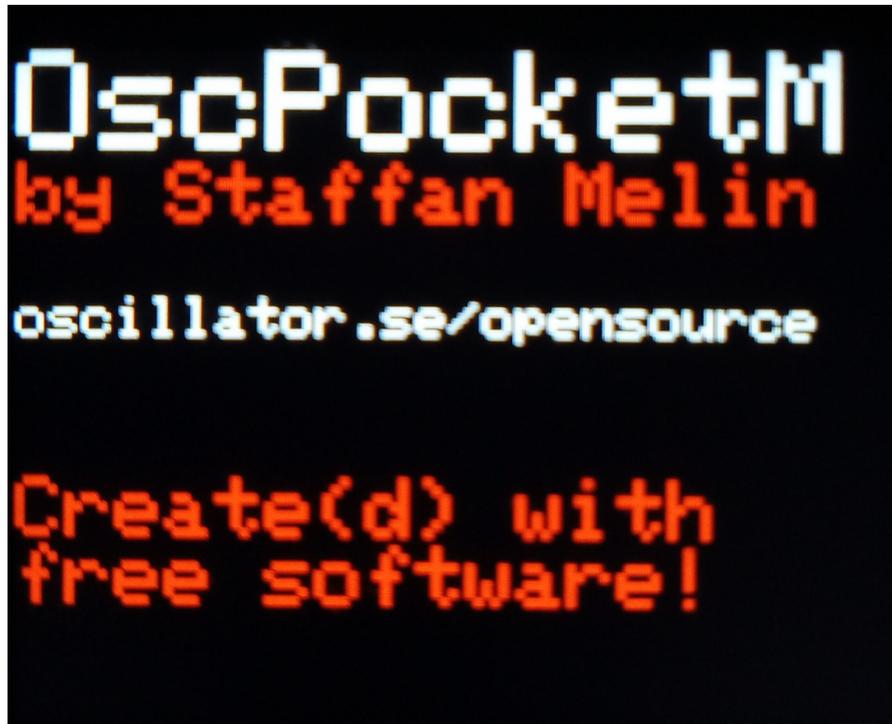


# OSCPOCKETM

by Staffan Melin



```
Project: OscPocketM  
Author: Staffan Melin, staffan.melin@oscillator.se  
License: GNU General Public License v3.0  
Version: 20210909  
Project site: https://oscillator.se/opensource
```

Thanks to Leif-Jöran Olsson for the M5Stack Core2!

# Introduction

Welcome to the OscPocketM (OPM) - the portable beatmaker for the M5Stack Core 2!

In your hands you will have 3 synthesizers and 1 drum machine that all can be sequenced and used to produce songs.

The synthesizers have selectable waveform, a low pass filter that can be modulated, an adjustable envelope and an optional second detunable oscillator.

The drum machine has seven sounds: kick, snare, open and closed hihat, clap, tom lo and tom hi. All sounds can be tweaked.

All sounds can be balanced using the built in mixer.

The synths have a piano roll step sequencer and the drum machine has dedicated 5 track step sequencer.

Sequences can be chained using the song editor.

There are also utility functions for saving and loading setups and sequences, as well as copying and clearing sequences and settings gate times.

All software is running on the Core2 including sound generation using the excellent Mozzi library.

# Table of Contents

Introduction.....	2
Specifications.....	3
How to use it.....	3
Screen: Main.....	4
Screen: Edit synthesizer.....	6
Screen: Edit drums.....	7
Screen: Edit synth sequence.....	8
Screen: Edit drum sequence.....	9
Screen: Edit song.....	10
Screen: Mixer.....	11
Screen: Utility.....	12
How to install it.....	13
Install and adjust Mozzi.....	13
Install OPM.....	13
Output to a speaker.....	14

## Specifications

- 3 x synths with 4 waveforms, LPF, ENV, 2nd oscillator
- 1 x drum machine with 7 sounds
- sequence: 32 x 1/16 note = 2 bars
- 6 sequences per synthesizer and drum
- song: up to 150 sequences
- save slots: 0-9

## How to use it

Most screens are divided into a grid of 4 x 4 areas, buttons. If they contain items in white color they can be clicked. If they contain numerical values you can decrease and increase them by touching the left- and righthand side of the button.

The Back button always returns you to the Main screen.

Note that touch on the Core2 works best if you hold it in your hand or have it connected to a computer.

## Screen: Main

Seq	Song	0	120
Syn0	Syn1	Syn2	Drums
Syn0	Syn1	Syn2	Drums
SongEd	Mixer		Util

### Function:

- Seq: Start/stop sequence mode
- Song: Start/stop song mode
- Current sequence
- BPM
- Yellow Syn0-2 + Drums: edit sounds
- Orange: Syn0-2 + Drums: edit sequence
- SongEd: Edit song
- Mixer: Adjust individual levels
- Util: Utilities

Main sceeen, sequence mode:

Seq	Song	0	120
		28	
Syn0	Syn1	Syn2	Drums
Syn0	Syn1	Syn2	Drums
SongEd	Mixer		Util

Main screen, song mode:

Seq	Song	3	120
		5	
Syn0	Syn1	Syn2	Drums
Syn0	Syn1	Syn2	Drums
SongEd	Mixer		Util

## Screen: Edit synthesizer

Wave:	FMode:		
Saw	Fixed		
FCut:	FRes:		
50	100		
A:	R:		
0	60		
2Wave:	2Detun:		Back
Saw	4		

### Function:

- Wave: Waveform of 1st oscillator: Sine, Triangle, Sawtooth, Square.
- Filtermode: Modulation of filter cutoff; fixed, random, fast, slow.
- FCut: Filter cutoff frequency
- FRes: Filter resonance
- A: Attack time
- R: Release time
- 2Wave: Waveform of 2nd oscillator.
- 2Detun: Detune of 2nd oscillator.

## Screen: Edit drums

KiFreq	KiRel :		ClRel :
60	120		200
SnFreq	SnRel :		
300	160		
HHOFq :	HHORel :	HHCFq :	HHCRel :
6000	60	4000	40
TomFqL	TomFqH	TomRel	Back
100	200	160	

### Function:

- KiFreq: Kick osc frequency
- KiRel: Kick release time
- ClRel: Clap release time
- SnFreq: Snare osc frequency
- SnRel: Snare release time
- HHOFq: Hihat Open high pass filter frequency
- HHORel: Hihat Open release time
- HHCFq: Hihat Closed high pass filter frequency
- HHCRel: Hihat Closed release time
- TomFqL: Tom Lo osc frequency
- TomFqH: Tom Hi osc frequency
- TomRel: Tom Lo/Hi release time

## Screen: Edit synth sequence

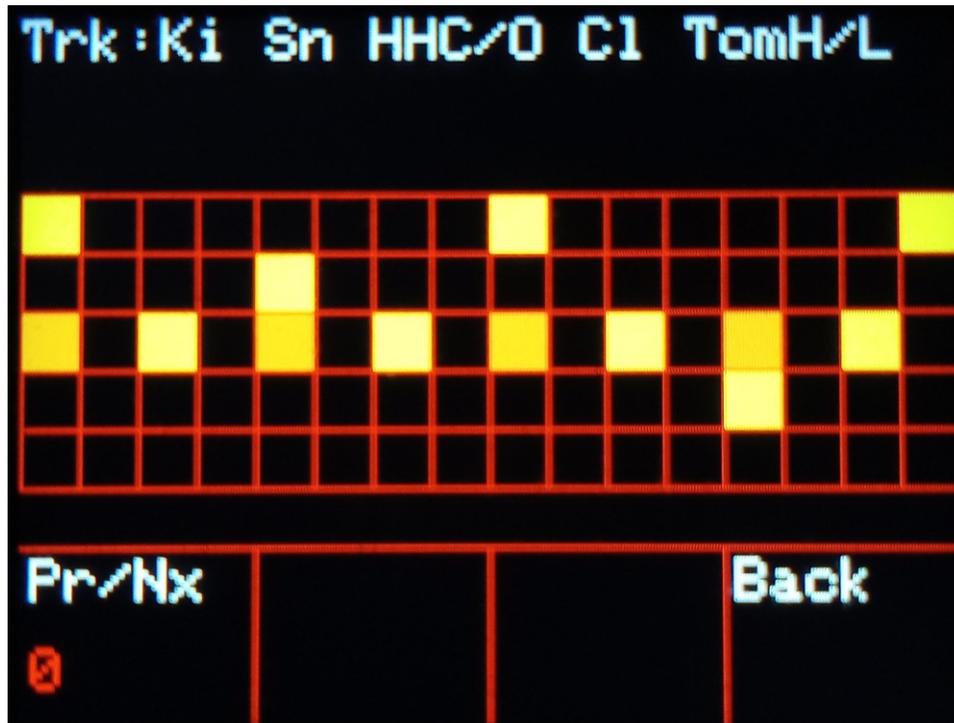


Function:

- Dn/Up: Scroll the grid down and up. The number indicates the MIDI note number of the lowest row.
- Pr/Nx: Scroll the grid left and right. The number indicates the step number of the leftmost column.
- Trnsp: Transpose the notes down and up.

Click to add a note. Click a note to erase it.

## Screen: Edit drum sequence



### Function:

- Track order: Kick, Snare, Hihat, Clap, Tom.
- Pr/Nx: Scroll the grid left and right. The number indicates the step number of the leftmost column.
- Click to add a note. Click a note to erase it.
- Hihat row: Click once to add a Closed Hihat. Click once more to convert it to an Open Hihat (orange color).
- Tom row: Click once to add a Hi Tom. Click once more to convert it to an Lo Tom (orange color).

## Screen: Edit song

4	4		Pr/Nx
5	5		Ins
6	4		Del
7	5	End	Back

### Function:

- Column 1: Step number.
- Column 2: Sequence number
- Column 3: Click to set last step of Song.
- Pr/Nx: Scroll the song up and down.
- Ins: Insert a step in the same row as the Ins button
- Del: Delete the step in the same row as the Del button.

## Screen: Mixer

Main:			
1			
Syn0:	Syn1:	Syn2:	SMain:
2	2	2	1
Kick:	Snare:	Hihat:	Clap:
1	1	3	2
Tom:			Back
1			

### Function:

- Note: 0 is highest volume and higher values = lower volume.
- Main: Main output volume.
- Syn0-2: Volume of synthesizers.
- SMain: Volume of summed synthesizers.
- Kick: Volume of Kick.
- Snare: Volume of Snare.
- Hihat: Volume of Open and Closed hihat.
- Clap: Volume of Clap.
- Tom: Volume of Tom Lo and Tom Hi.

## Screen: Utility

Save	Dest: 0	Load	Src: 0
Copy	Clear	Screen 3	Batt%: 84
FrSrc: Syn0	FrSeq: 0	ToSrc: Syn0	ToSeq: 0
Gate0: 50	Gate1: 50	Gate2: 50	Back

Function:

Data can be saved in 10 slots numbered 0-9. Saved data: Synth settings, Drum settings, Sequences, Song, Gate times, Tempo.

- Save: Save data in slot Dest.
- Load: Load data from slot Src.

Sequences can be copied between synths, and from one drum sequence to another.

- FrSrc: source to copy from.
- FrSeq: sequence to copy from.
- ToSrc: source to copy from.
- ToSeq: sequence to copy from.

Sequences can be emptied (cleared).

- FrSrc: source to clear.
- FrSeq: sequence to clear.

Gate time controls how long a sequenced synthesizer note is on (ie before it enters the release stage).

- Gate0-2: Gate in percent of 1/16 for synthesizer 0-2.

## How to install it

### Install and adjust Mozzi

Download and install the Mozzi library:

<https://sensorium.github.io/Mozzi/download/>

In AudioConfigESP32.h change these lines to:

```
// Set output mode
#define ESP32_AUDIO_OUT_MODE PT8211_DAC
```

and

```
// For external I2S output, only: I2S_PINS
#define ESP32_I2S_BCK_PIN 12
#define ESP32_I2S_WS_PIN 0
#define ESP32_I2S_DATA_PIN 2
```

The AudioConfigESP32.h file can be found in the directory where Arduino libraries are installed. On my GNU/Linux machine it is:

```
<home>/Arduino/libraries/Mozzi-master/
```

This works even though the Mozzi uses a NS4168 I2S amp.

### Install OPM

Before uploading, set Tools > PSRAM: Disabled in the Arduino IDE.

Compile and upload the OPM code to your M5Stack Core2.

Touch the screen to make the splash screen go away.

The OPM comes with 6 demo sequences and a demo song:

1. Touch Seq and the first sequence plays (sequence 0).
2. Touch Seq to stop.
3. Set the sequence number to 1 by touching the righthand side of the sequence number (third button on the topmost row).
4. Touch Seq to play it.
5. Touch Seq to stop.
6. Touch Song to listen to the song which uses sequences 2-5.

The touch screen is not that precise. I prefer to use a "pointing stick".

## Output to a speaker



You can easily output the sounds to an active speaker (or another kind of input).

In AudioConfigESP32.h change this line to:

```
// Set output mode
#define ESP32_AUDIO_OUT_MODE INTERNAL_DAC
```

With your Core2 powered off, remove the back plate and connect a patch cable to GND and another to DAC G26.

The DAC should be connected to the tip of an audio cable and the GND to the sleeve.

Power up the Core2 and recompile and upload the program.