

Short course Sonic Pi

Dipl.-Inf. Markus Sing
markus.sing@gmx.ch

homemade July 2026

Special thanks to Felix Bánteli and Simon Rupf

Sonic Pi

- open source music program
- from simple to professional :-)
- also: live coding

download from <https://sonic-pi.net/>

versions for Windoofs, macOS, Raspberry Pi OS

⚠ other Linux stuff not really supported

Ubuntu: – older version of Sonic Pi (v3.2.2)

– adjust output audio stack

<https://linuxmusicians.com/viewtopic.php?p=176499> ↩
&hilit=Problems+and+fixes+with+jack+in+ubuntu#p176499

Sonic Pi – important adjustment!

⚠️ prefs: Audio: Synths and FX: Save Mode

Get a Tone

- synthetic

```
play :C5    # or: play 72 (MIDI note number for C5)
sleep 1     # 1 -> 1 beat
            # 2 -> 2 beats
            # 0.5 -> 1/2 beat
            # etc.
```

- play sample

```
sample :guit_e_slide
```

Get a Tone: all at the same time versus one after another

- ```
play :C5 # all at the
 play :E5 # same time
 play :G5 # -> chord
 sleep 0.5
```

- ```
play :C5    # one  
sleep 0.5    #  
play :E5    # after  
sleep 0.5    #  
play :G5    # another  
sleep 0.5    #
```

try: play all tones of one octave one after another

bpm – one after another, good practice :-)

```
use_bpm 104    # number of quarters per minute
```

```
full = 4
```

```
half = 2
```

```
quarter = 1
```

```
eighth = 0.5
```

```
play :C5    # 1/2
```

```
sleep half
```

```
play :E5    # 1/4
```

```
sleep quarter
```

```
play :G5    # 1/8
```

```
sleep eighth
```

```
:
```

Improve Sound

```
use_synth :tb303    # -> Acid Bass
use_bpm 104         # number of quarters per minute

full = 4
half = 2
quarter = 1
eighth = 0.5

with_fx :reverb do  # -> echo (Kim Wild, "Kids in America")
  play :C5    # 1/2
  sleep half
  play :E5    # 1/4
  sleep quarter
  play :G5    # 1/8
  sleep eighth
  :
end
```

Run Stuff parallel — and Rerun always again

```
live_loop :LoopName1 do  
  :  
end
```

```
live_loop :LoopName2 do  
  :  
end
```


Random Stuff

```
live_loop :LoopName1 do
  play rrand(72, 86)    # notes from 72 (:C5) to 86 (:D6)
                        # !!! also intermediate values
  sleep rrand(0.5, 1)   # duration from eighth to quarter
                        # !!! also intermediate values
end
```

Music Programming, to be continued ...

For much more music commands — ask Felix ...

There is just one more thing :-)

Live Coding with Smartphone

smartphone app: Sensors2OSC (source: fdroid)

OSC: Open Sound Control

connection: via common network / router

(this is much easier then via creating a hotspot)

also via bluetooth possible ...

Via common Network — required Settings

- get the internet address of your network (e.g. 10.128.0.71)
- smartphone, Sensors2OSC:
settings (: right above) / Communication / Host 10.128.0.71
/ Port 4560
- SonicPi:
prefs: I/O: Network-Based OSC: activate OSC-Server: ON
Send/Receive remote OSC: ON
⚠ just don't care about network
details here :-)
prefs: Editor: show Cue-Log: ON to see incoming OSC packets

SonicPi: Code to get Data in

inside a live_loop:

```
x, y, z = get "/osc:*/accelerometer"
```

use x, y, z in commands to change your sound

phone flat on table, screen facing up:

- x: moving left / right
- y: moving forward / backward
- z: moving up / down

Sensors2OSC: Settings to send Data

/accelerometer ON

then send data ON

then move smartphone :-)