live coding music

– robotic pianos –

vitor guerra rolla
postdoctoral fellow

summary

- introduction
- chuck programming language
- MIDI protocol
- pre-loaded code
- robotic pianos => instruments
- live coding

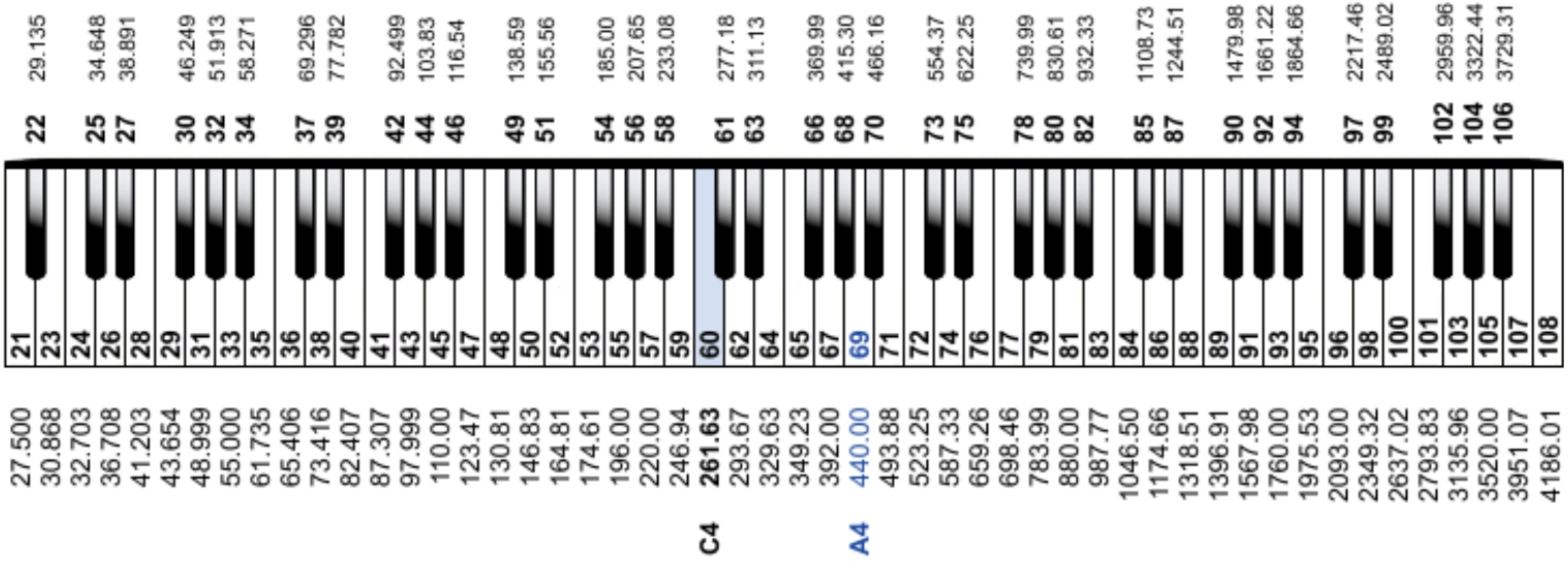
live coding

- performing arts
- on the fly or real-time
- musician(s) + computer(s)
- video animation or image
- pedagogical / learning purpose

chuck

- Ge Wang
- real-time sound synthesis
- time-based programming => now
- operators: => , %
- MIDI compatible

musical instrument digital interface (MIDI)



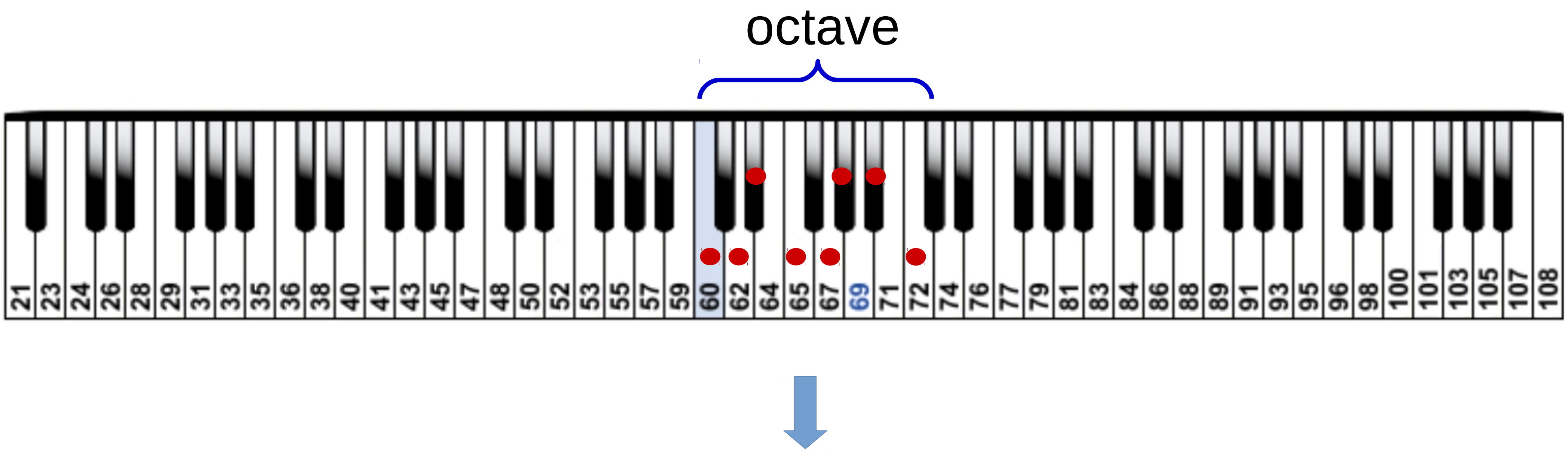
Std.mtof()

pre-loaded code

- root key $\longrightarrow$ (key.ck)
  changeRootNote();
- musical scales $\longrightarrow$ (scales_lib.ck)
  quantize();
- monitoring $\longrightarrow$ (monitor.ck)
  monitor();

pre-loaded code

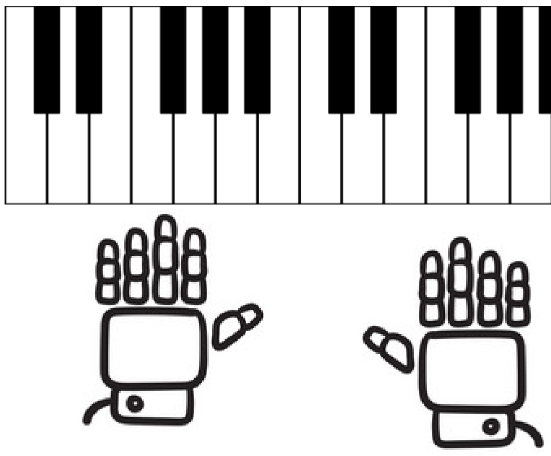
- (scales_lib.ck) => function quantize(C, aeolian)

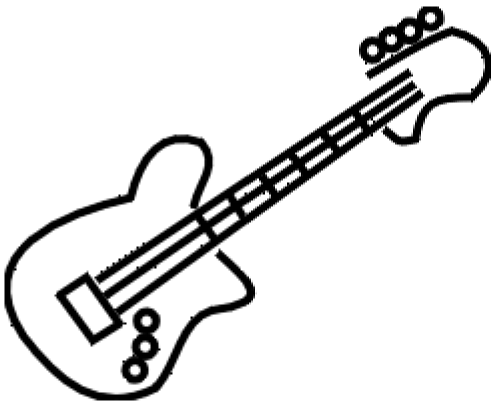


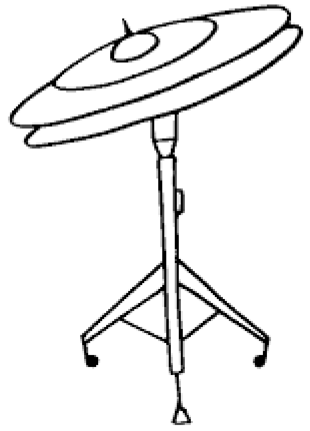
C aeolian scale

robotic pianos

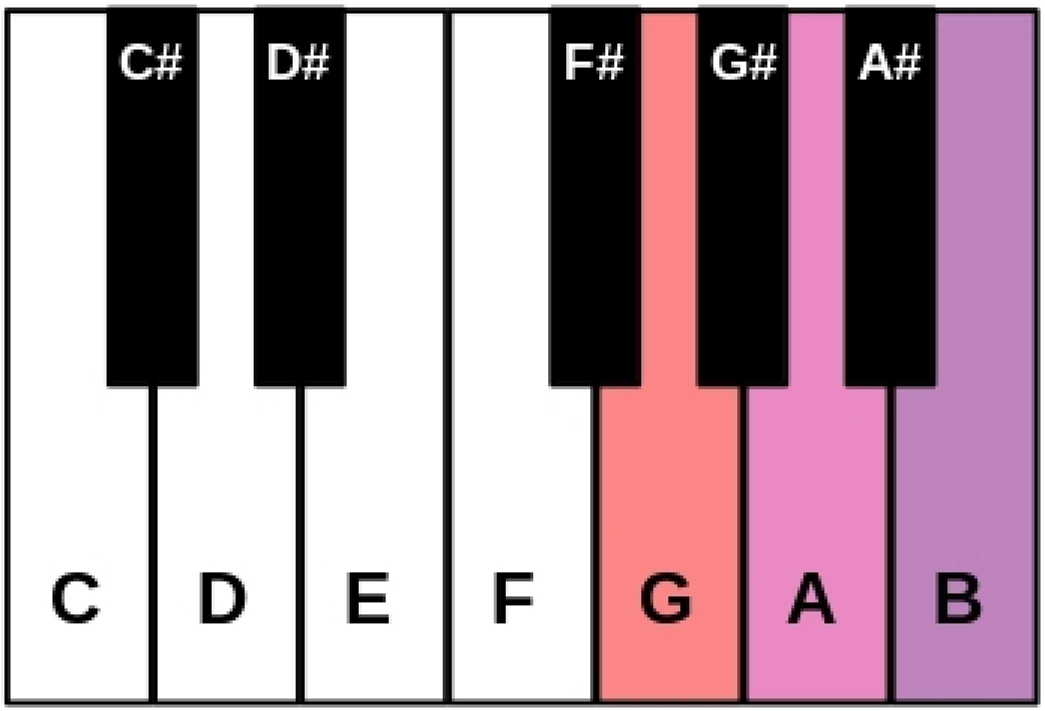
- composed by Andrew Sorensen
- impromptu language =>
- two piano hands + guitar + hi-hat







lefthand – step 1



immutable !

55 57 59

2 * G + A + B

(4 beats)

0,5 sec → time interval between notes

lefthand – step 2

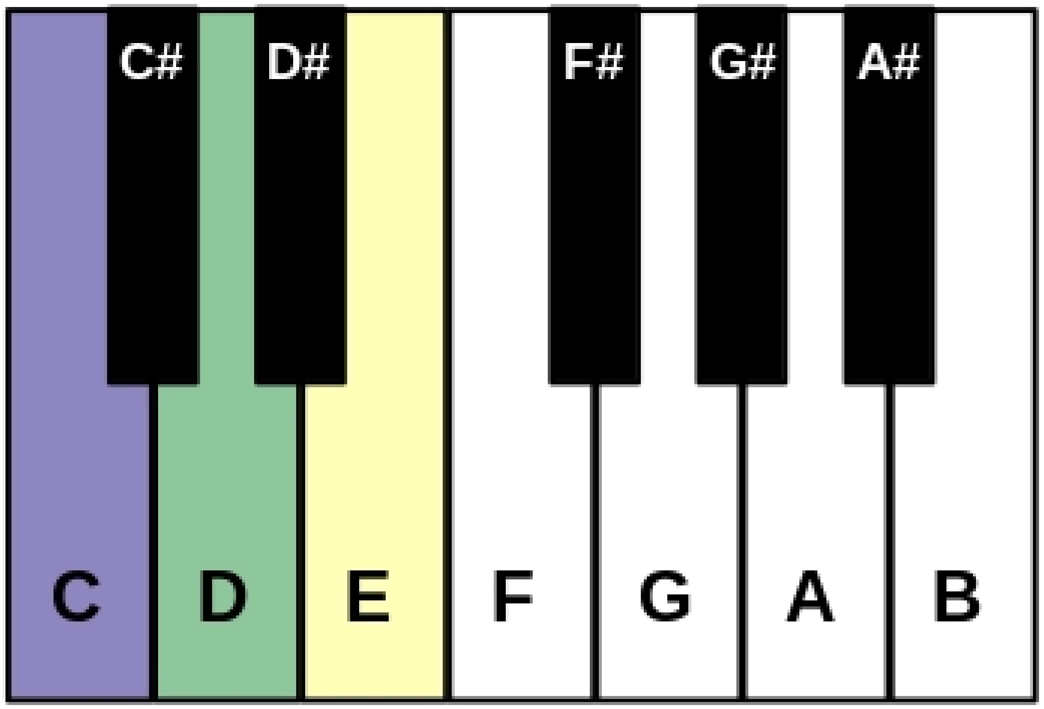
<u>mutable</u> !

↓

key
.changeRootNote()

↓

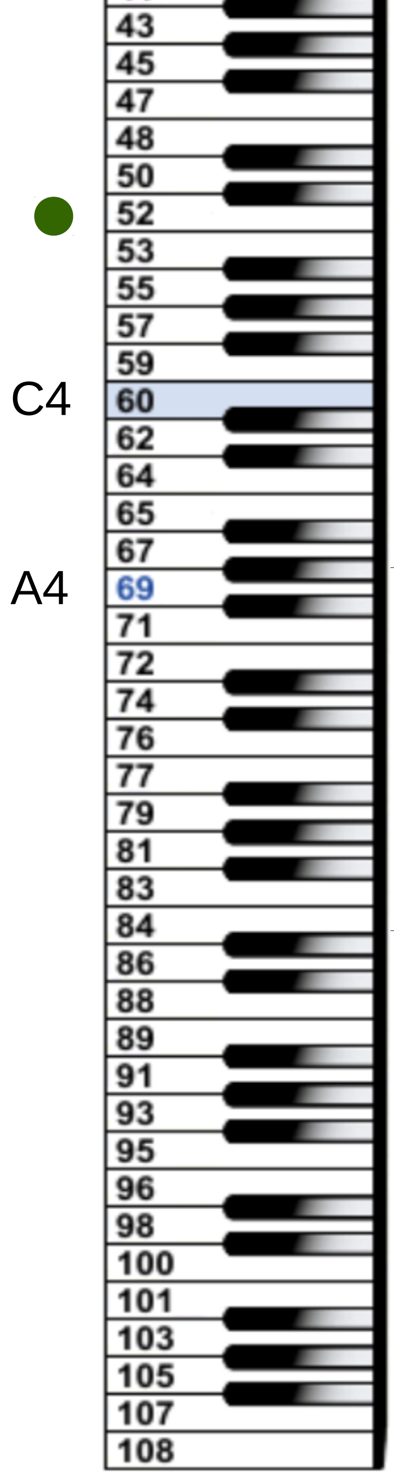
harmonic
progression



48 50 52

One note per measure
(8 beats)

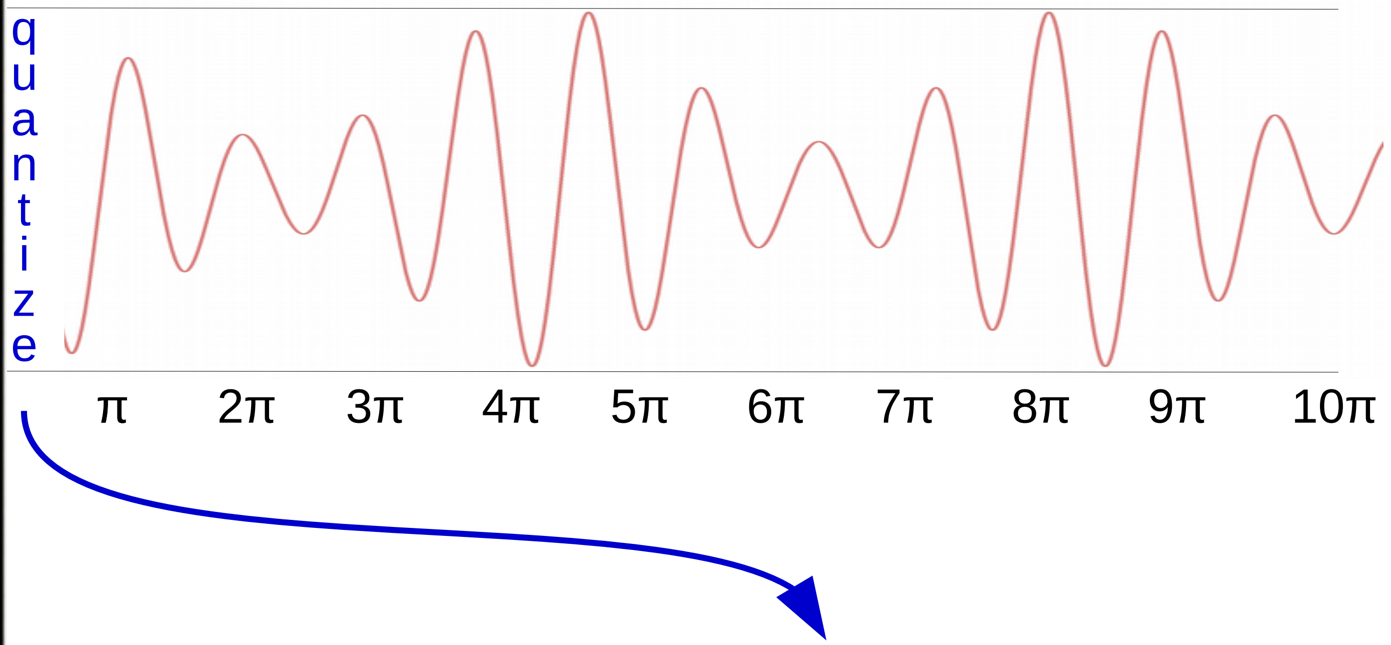
0,5 sec → time interval
+
0,25 sec offset → step 1



righthand

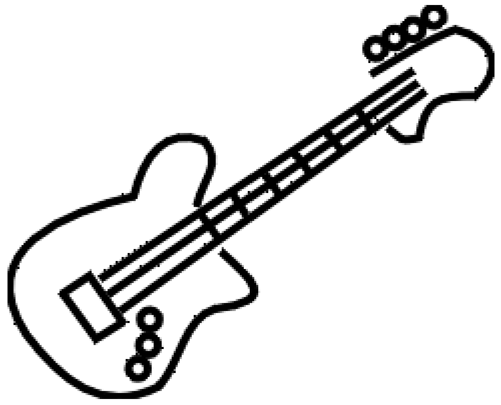
(3 * cos (x / 2) + 5) * (cos ((7 * x)/3) + key.root + 24)

x = 90°

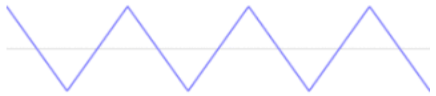


quantize(wave, scale, key.root);

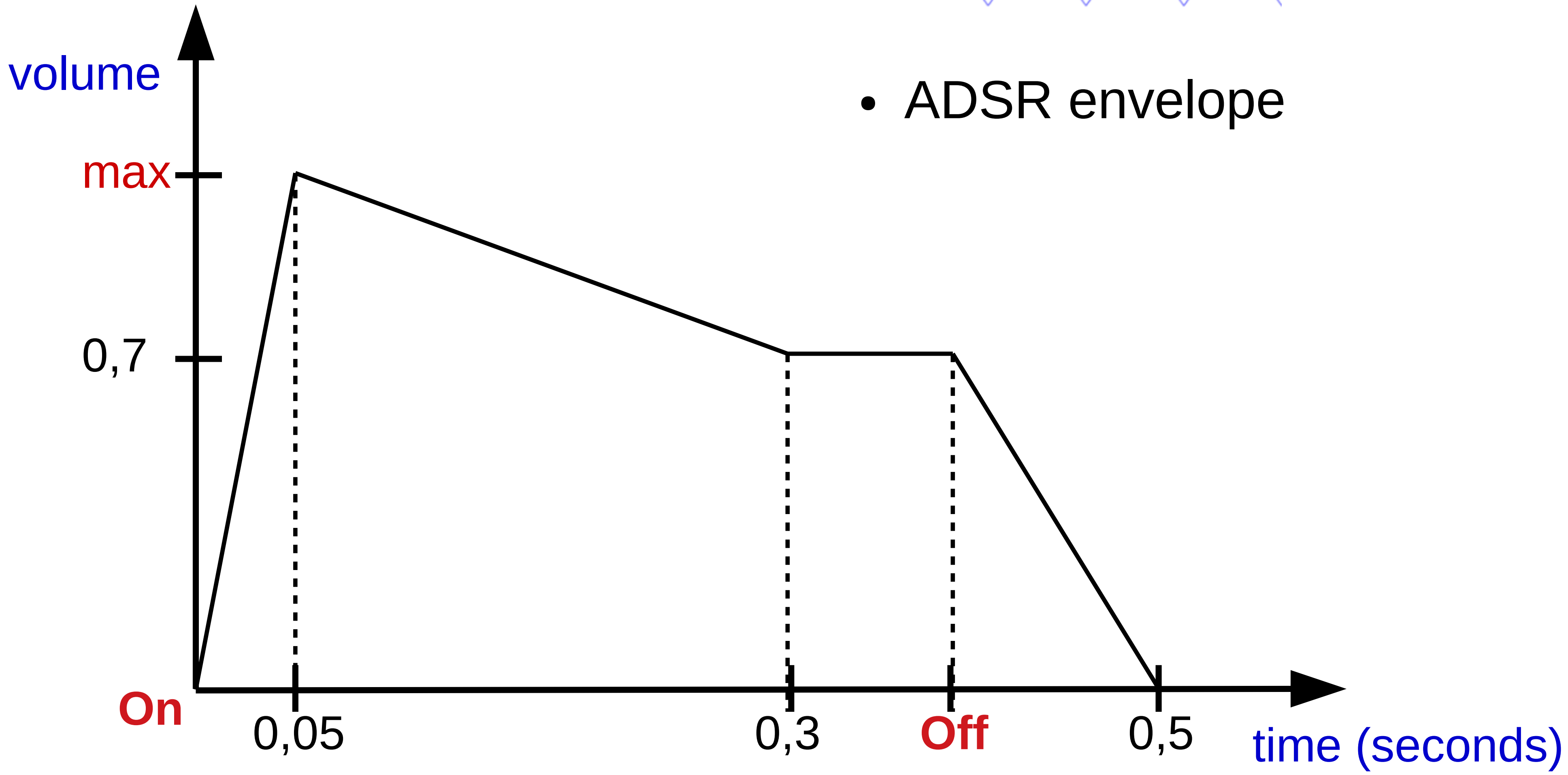
guitar



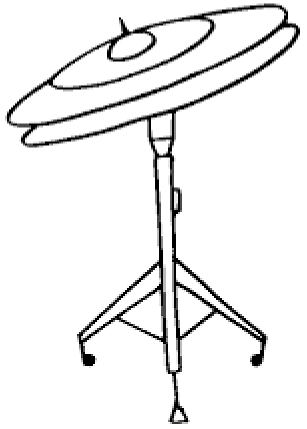
- triangle wave form



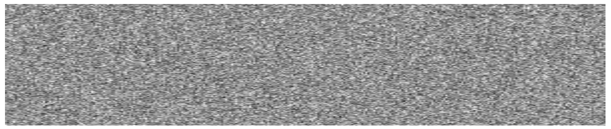
- ADSR envelope



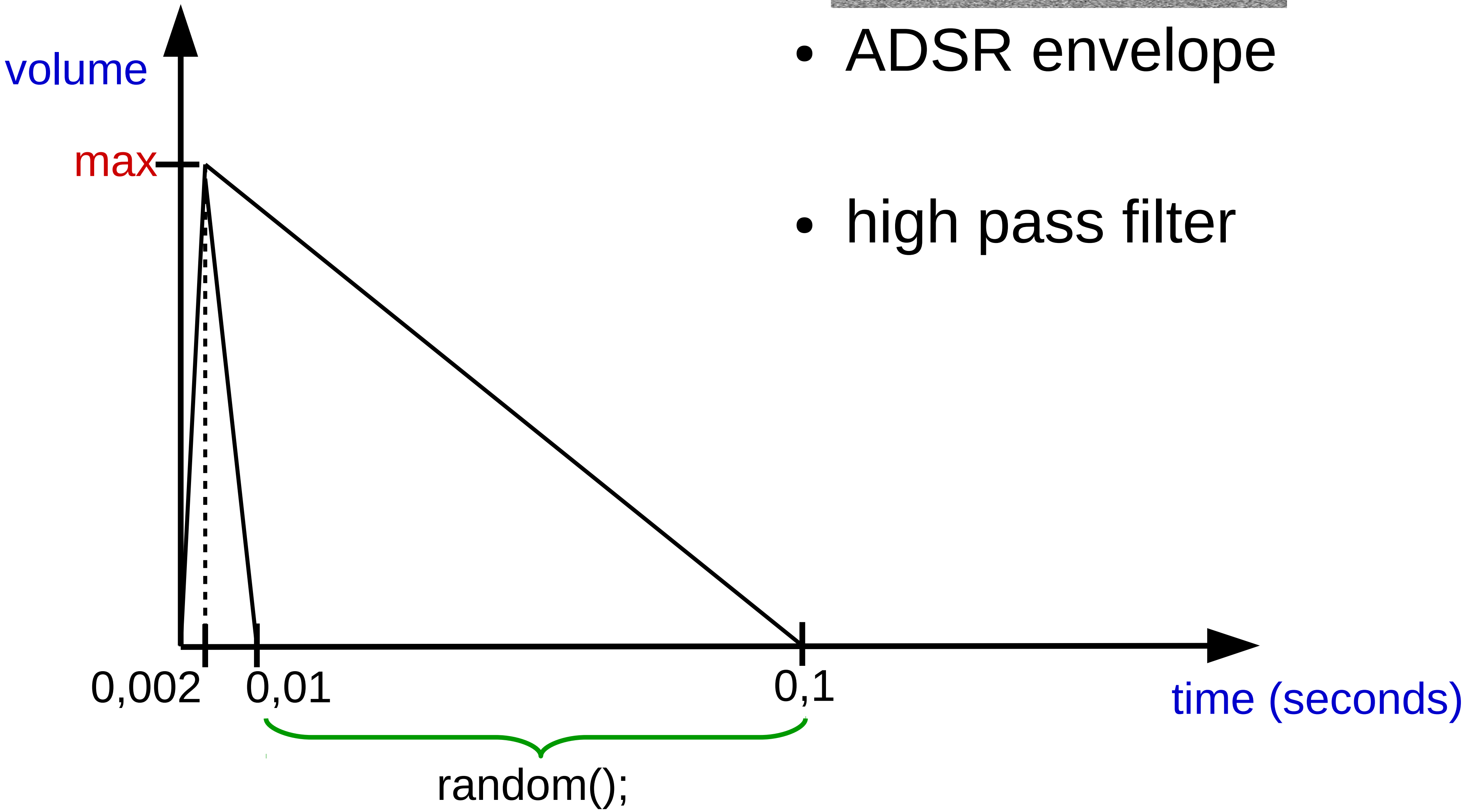
hi-hat



- noise



- ADSR envelope
- high pass filter



live coding – robotic pianos

impa

TOPLAP

VisgrafLab

thank you

impa

TOPLAP

VisgrafLab