

Y. Collette (ycollette.nospam@free.fr)
<https://audinux.github.io>





Le Live Coding 1/5

SuperCollider : <https://supercollider.github.io/>

```
play{SinOsc.ar(OnePole.ar(Mix( LFSaw.ar([1,0.99],  
[0,0.6],2000,2000).trunc([400,600])*[1,-1] ),0.98)).dup*0.1}
```

<https://www.youtube.com/watch?v=wNWFSladAH8>

CSound : <http://www.csounds.com/>

```
sr = 44100  
ksmps = 32  
nchnls = 2  
0dbfs = 1  
  
instr 1  
  
iflg = p4  
asig oscils .7, 220, 0, iflg  
outs asig, asig
```

QuteCsound

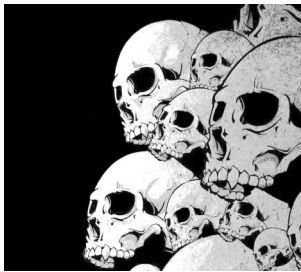
Chuck : <http://chuck.cs.princeton.edu/>

```
// set the global gain  
.1 => dac.gain;
```

```
// connect  
SinOsc a => dac;  
110.0 => a.freq;  
1::second => now;  
SinOsc b => dac;  
220.0 => b.freq;
```

miniAudicle

<https://www.youtube.com/watch?v=BHooZu5xzAs>
<https://www.youtube.com/watch?v=vNrRdyDIniQ>



Le Live Coding 2/5

<http://sonic-pi.net/>

The screenshot shows the Sonic Pi application window. The top toolbar includes buttons for Run, Stop, Rec, Save, Load, Size, Align, Info, Help, and Prefs. The main code editor displays a Ruby script for a live loop. To the right, the 'Préférences' (Preferences) window is open, showing the 'Éditeur' (Editor) tab with options for line numbers, trace, buttons, and tabs, as well as 'Look and Feel' settings like 'Mode sombre' and 'Plein écran'. Below the code editor is a buffer bar with 10 buffers. At the bottom, a 'Tutoriel' (Tutorial) sidebar is visible on the left, and the main display area shows the 'Sonic Pi' logo and version information.

```
1 # Rerezzed
2
3 # Coded by Sam Aaron
4
5 use_debug false
6 use_random_seed 103
7 notes = (scale :e1, :minor_pentatonic, num_octaves: 2).shuffle
8
9 live_loop :rerezzed do
10   tick_reset
11   t = 0.02
12   sleep -t
13   with_fx :bitcrusher do
14     s = synth :mod_dsaw, note: :e2, sustain: 8, note_slide: t, release: 0
15     64.times do
16       sleep 0.125
17       control s, note: notes.tick
18     end
19   end
20 end
```

Buffer 0 Buffer 1 Buffer 2 Buffer 3 Buffer 4 Buffer 5 Buffer 6 Buffer 7 Buffer 8 Buffer 9

==> Welcome to Sonic Pi

1 Bienvenue à Sonic Pi
1.1 Codage en 'live'
1.2 Exploration de l'interface
1.3 Apprendre en jouant
2 Synthés
2.1 Vos premiers Beeps

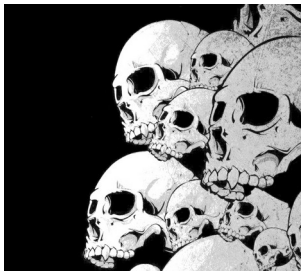
Tutoriel Exemples Synthés Fx

Sonic Pi

music_as :code
code_as :art

v2.10-dev

Sonic Pi v2.10.0-dev-24344 on Linux



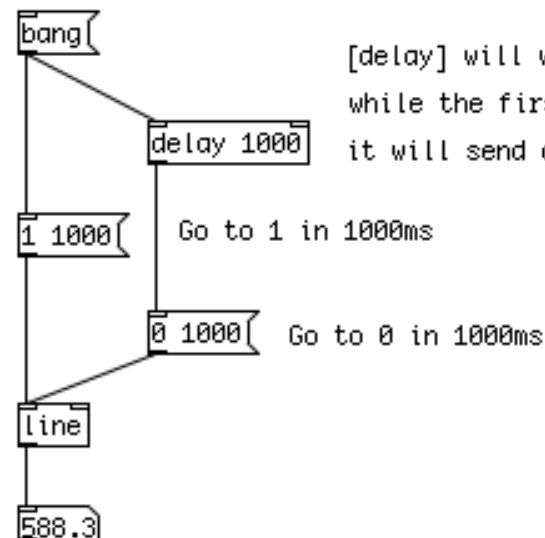
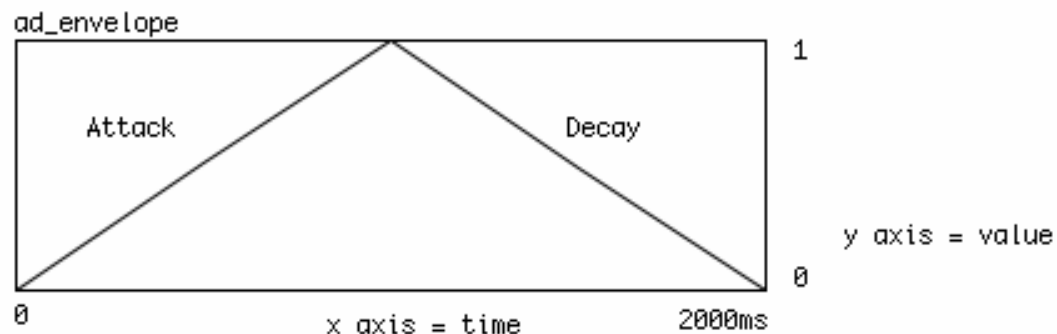
Le Live Coding – 3/5

Pure Data : <http://puredata.info/>

Un outil de programmation
visuel dédié à l'audio et à la
vidéo.

eg2.pd

Graphical representation of a simple up/down, or
Attack/Decay (AD) envelope.



[delay] will wait 1000ms after the input "bang",
while the first ramp is being executed, and then
it will send a "bang" to trigger the second ramp.



Le Live Coding – 4/5

ProjectM : un diffuseur de vidéo synchronisé à l'audio

<http://projectm.sourceforge.net/>

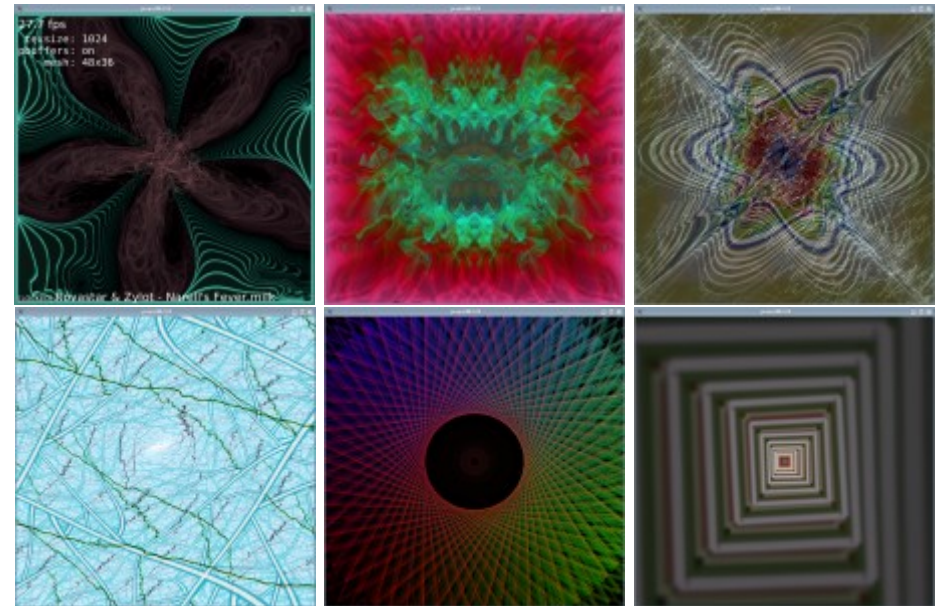
Code issu de WinAmp.

Pour lancer la version Jack de ProjectM

```
$ projectM-jack
```

Pour lancer la version PulseAudio de ProjectM

```
$ projectM-pulseaudio
```



F1 : Aide

F2 : Titre de la chanson

F3 : Nom du preset

F4 : Paramétrage du rendu

F5 : FPS

F : Plein écran

L : Verrouiller / Déverrouiller le preset

M : Affiche le menu

R : Preset aléatoire

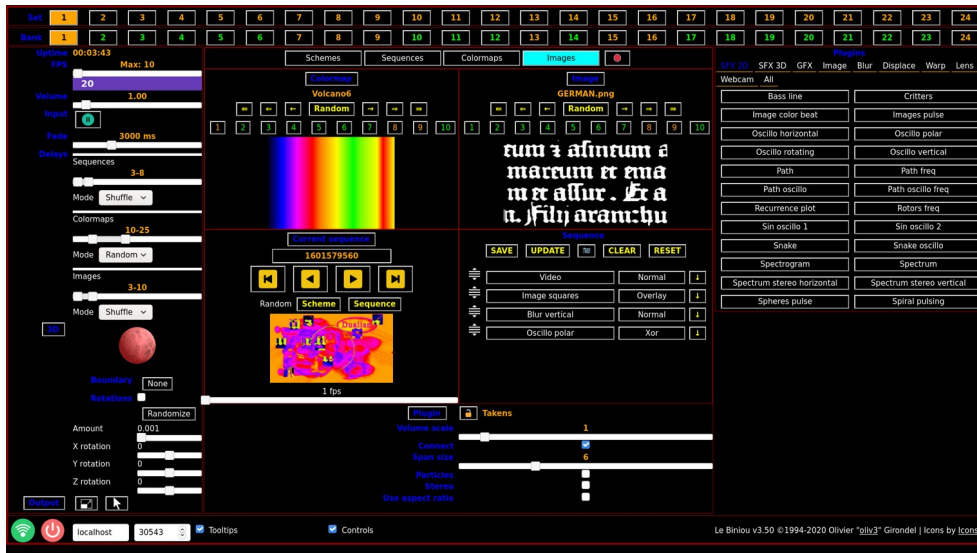
N : Preset suivant

P : Preset précédent



Le Live Coding – 5/5

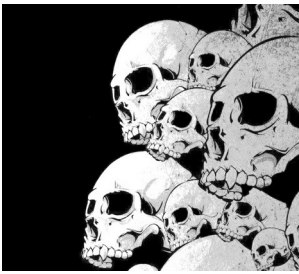
<https://biniou.net/>



Pour démarrer le biniou :
\$ le biniou --input jackaudio

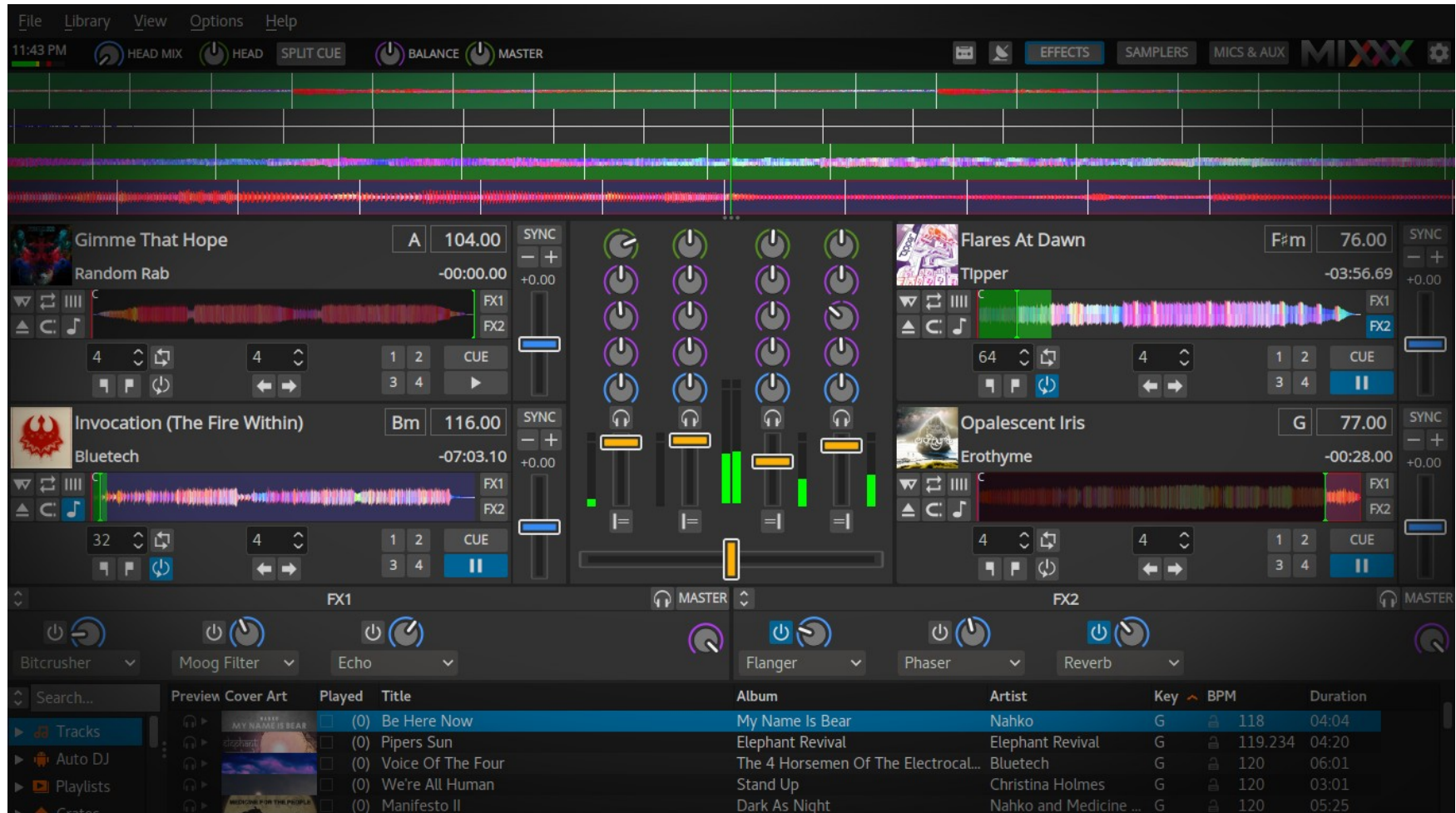
Et en cas de conflit sur l'accès à la webcam :
\$ le biniou --input jackaudio --webcams 0

Le biniou démarre une fenêtre de contrôle (à gauche) et une fenêtre d'animation (à droite).
Il faut ensuite connecter l'entrée de le biniou à une sortie audio.



Mixxx

Pour le DJing



<http://www.mixxx.org>

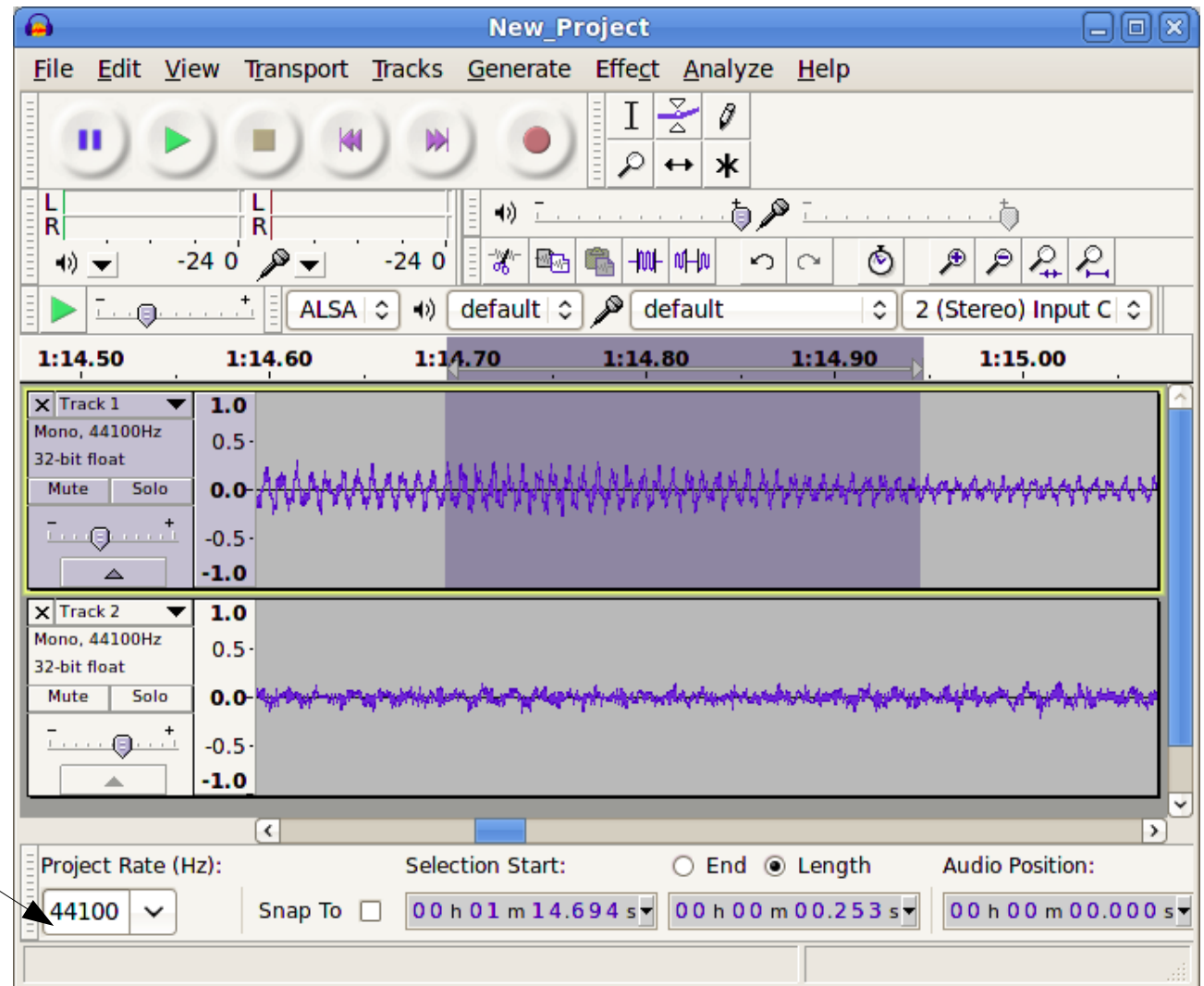


Audacity

L'éditeur audio

Lorsqu'on utilise Audacity avec Jack, il faut bien prendre garde de régler la fréquence d'échantillonnage :
Edition → Préférences
→ Qualité

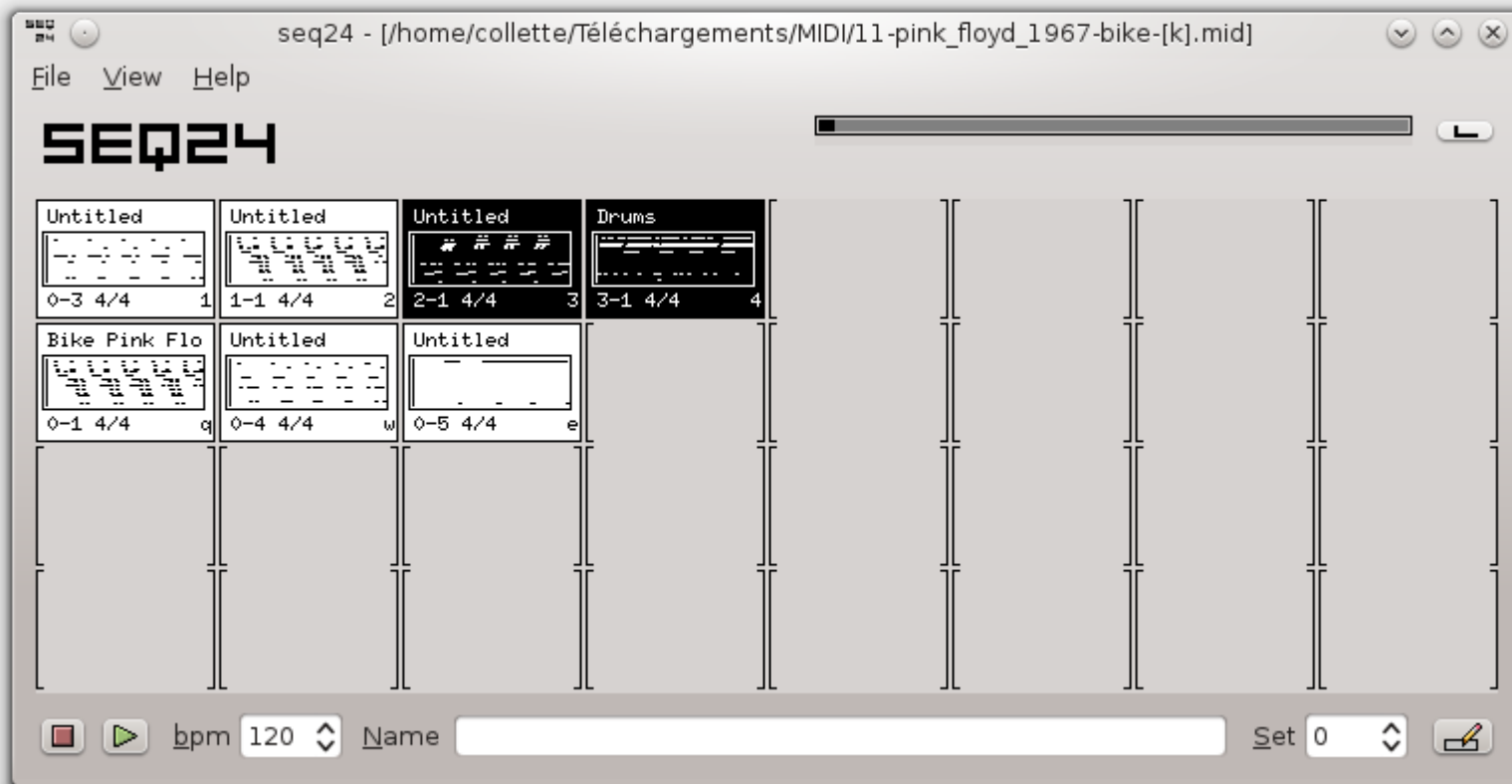
Il faudra faire correspondre cette fréquence d'échantillonnage avec celle de Jack.



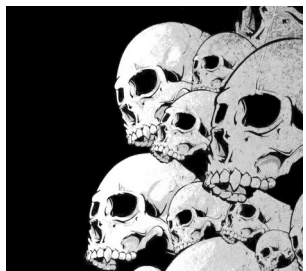


Seq24

Un séquenceur matriciel



<https://launchpad.net/seq24>
<https://github.com/ahlstromcj/sequencer64>



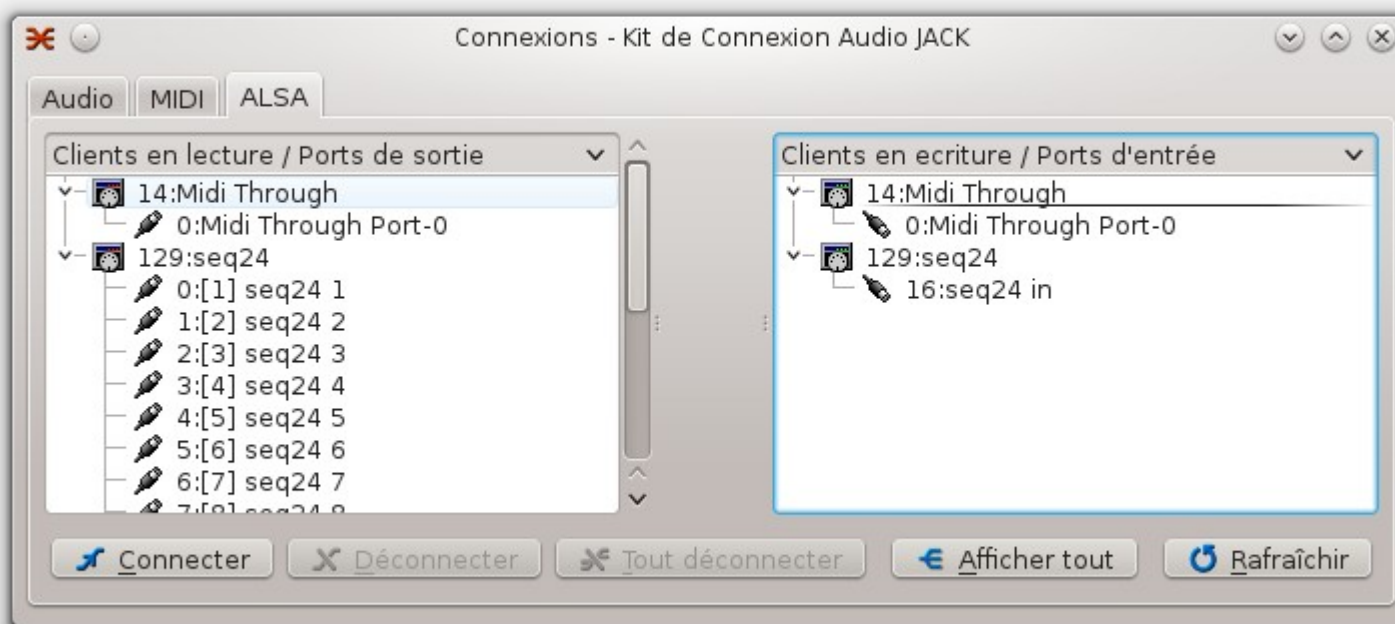
Seq24

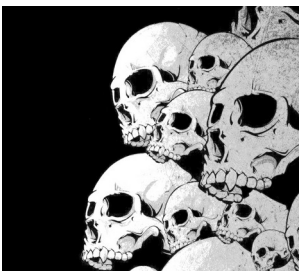
Coté Jack

Démarrage en ligne de commande recommandé :

```
$ seq24 -m
```

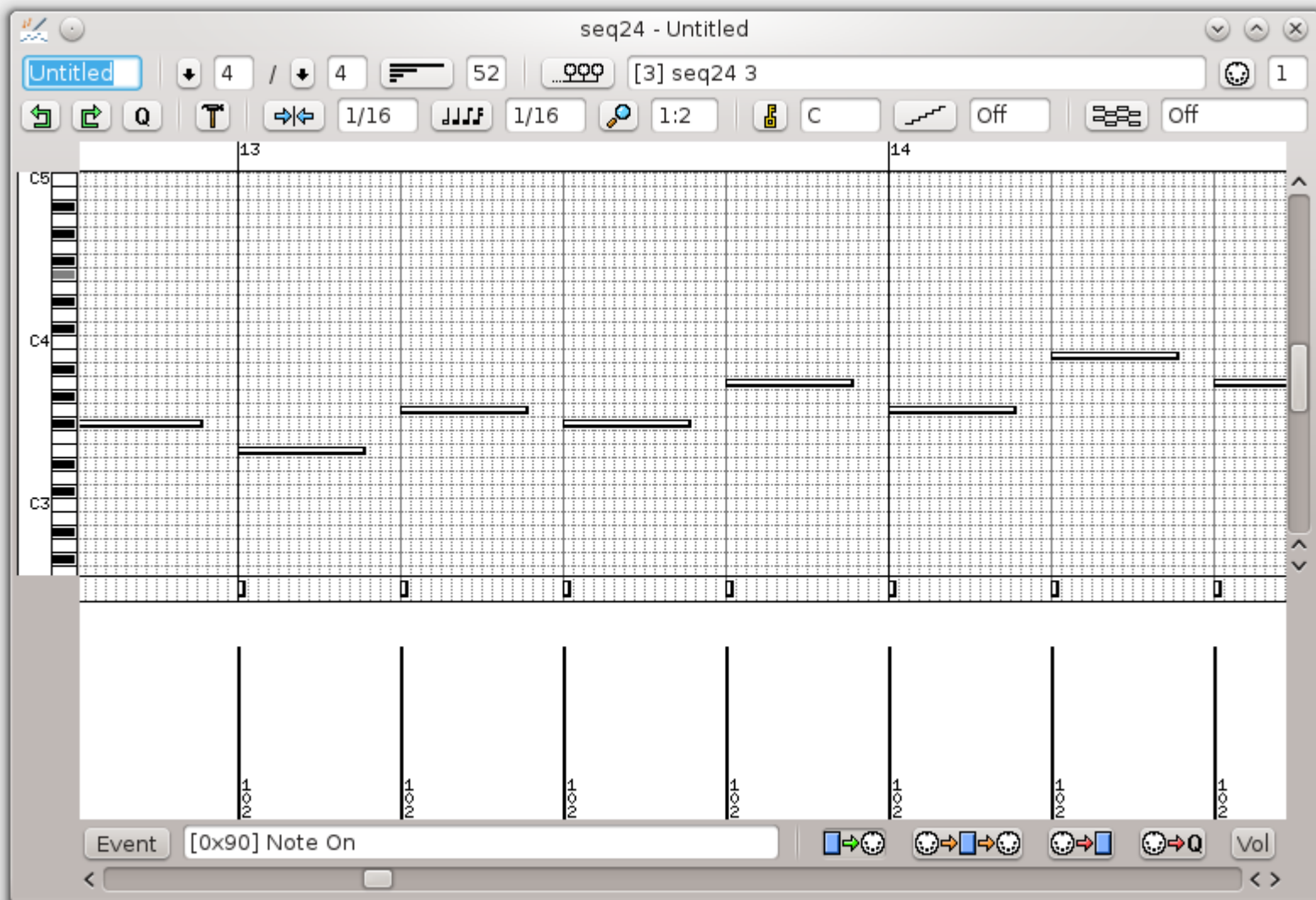
-m, --manual_alsa_ports: seq24 ne réquisitionnera pas de ports ALSA





Seq24

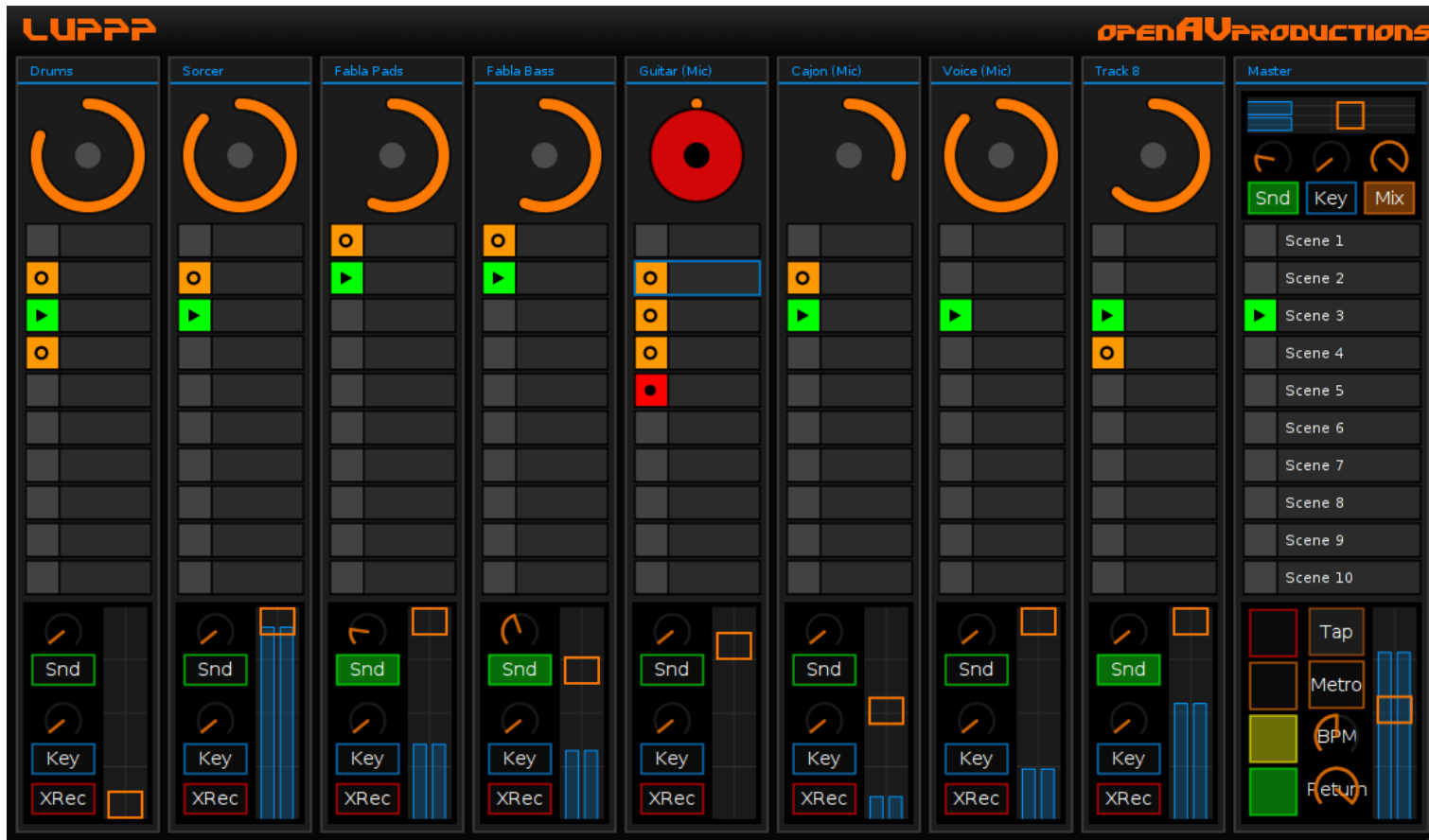
L'éditeur MIDI



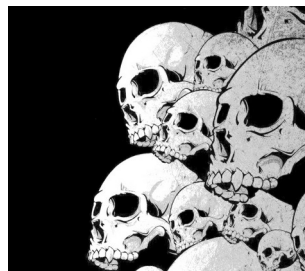


OpenAV / Luppp

Un séquenceur matriciel



<http://openavproductions.com/luppp/>



Impro-visor

Pour le Jazz

The screenshot displays the Impro-visor software window titled "Impro-visor: 12-Bar Blues". The interface includes a menu bar (File, Edit, Transpose, View, Play, Utilities, Window, Grammar: My, Preferences, Help) and a toolbar with various icons. Below the toolbar, there are controls for Playback Location (0:00 to 3:12), Looping (Loop, 2), Volume (Mute), Tempo (180.0), Transpose (0), Bars per Chorus (12), Tracker Delay (0), and Parallax (0). A "Textual Entry" field is also present. The main display area shows a 12-Bar Blues progression for Clifford Brown, generated from grammars learned from solos of different players. The progression is displayed on a musical staff with various chords and notes. The chords are: F13, Bb13, Bo7, F13, Cm9, F13b9, Bb13, Bo7, F13, D7#5#9, Gm9, C13b9, F13, D7#5#9, Gm9, C13b9. The style is set to "swing".

<http://www.cs.hmc.edu/~keller/jazz/improvisor/>



Impro-visor

Pour le Jazz

Pour connecter Impro-visor à QSynth, il faut lancer l'interface virtuelle MIDI de ALSA :

```
$ sudo modprobe snd-virmidi
```

On obtient 4 Virtual Raw MIDI comme le montre l'image suivante:
Enalsa Out, on a :

- 14: MIDI Through
- 20: Virtual Raw MIDI 1-0
- 21: Virtual Raw MIDI 1-1
- 22: Virtual Raw MIDI 1-2
- 23: Virtual Raw MIDI 1-3

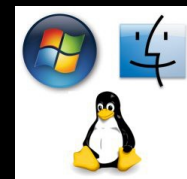
Après cela, il suffit de connecter Impro-Visor à une entrée Virtual RawMIDI et Qsynth à une sortie Virtual RawMIDI.

Enalsa In, on a :

- 14: MIDI Through
- 20: Virtual Raw MIDI 1-0
- 21: Virtual Raw MIDI 1-1
- 22: Virtual Raw MIDI 1-2
- 23: Virtual Raw MIDI 1-3
- 128:Timidity



Milkytracker



<http://www.milkytracker.org/>



Historique

Soundtracker – 1987 (Amiga)
 Protracker – 1990 (Amiga)
 Octamed – 1991 (Amiga)
 Scream Tracker 3 – 1993 (PC)
 Fast Tracker 2 – 1995 (PC)
 Impulse Tracker 2 – 1996 (PC)
 Renoise – 2000 (PC & Mac)
 Skalettracker – 2003 (PC)

Type de fichiers

XM – MOD – IT – S3M

Voir l'article de [wikipedia](#)

Exemple YouTube

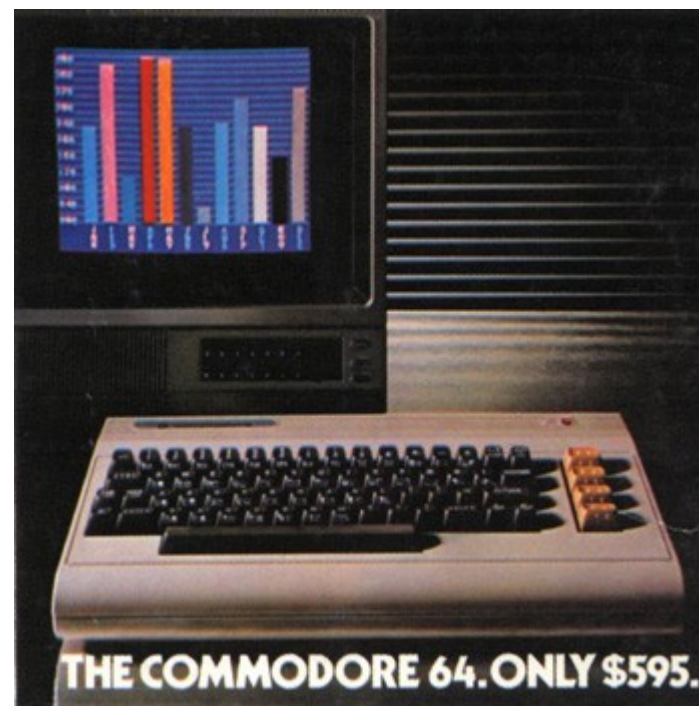


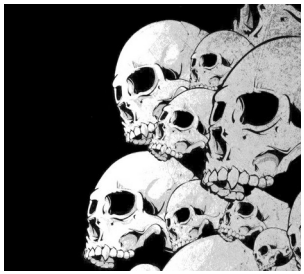
Milkytracker

Amiga - 1987



Commodore - 1982



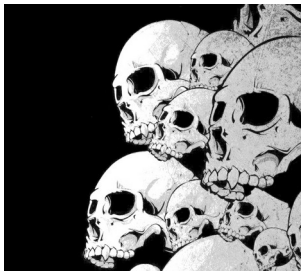


Klystrack



Exemple YouTube

<https://kometbomb.github.io/klystrack/>



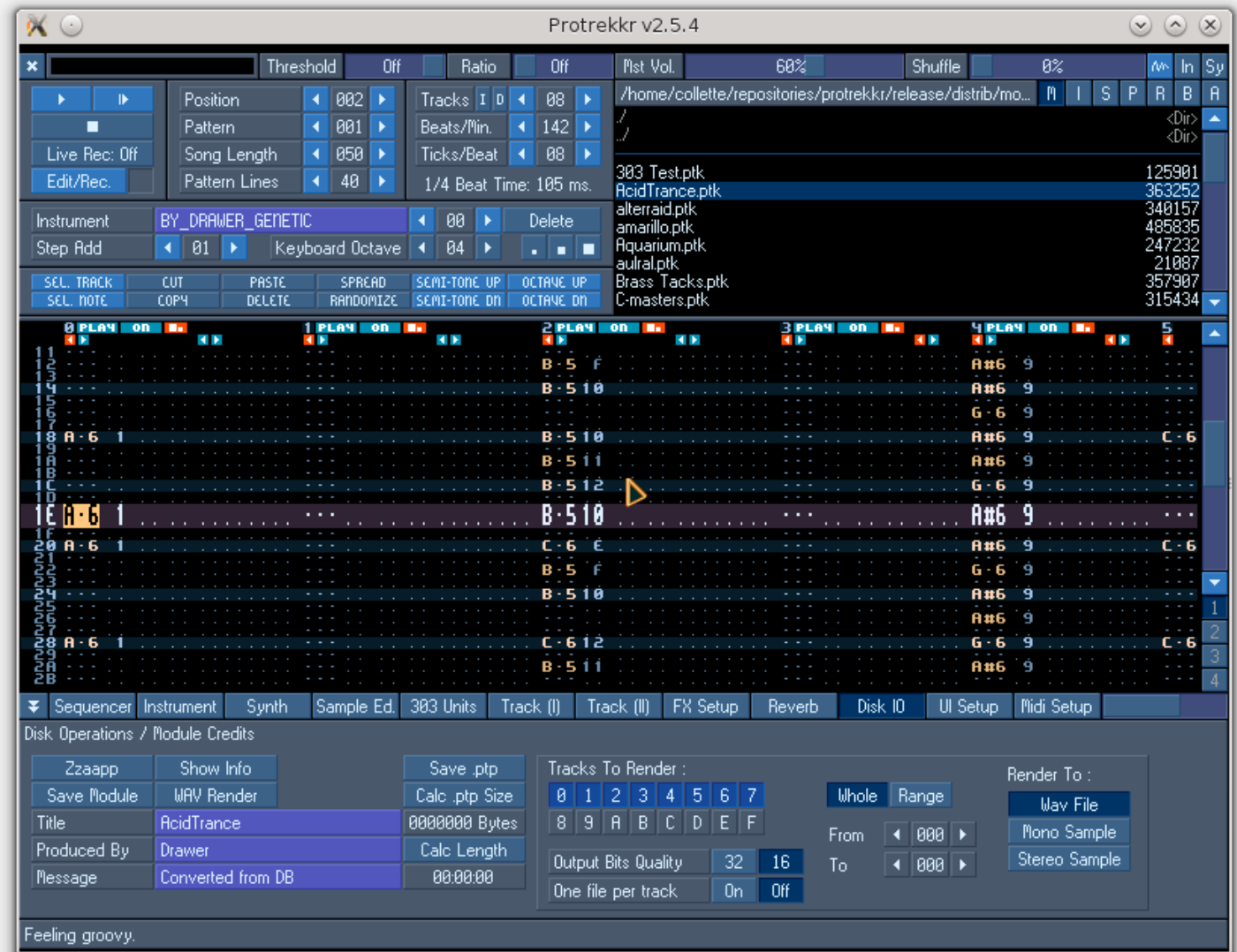
Protrekkr

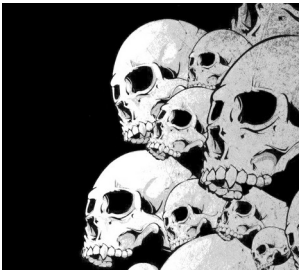
<https://github.com/falkTX/protrekkr>

Deux versions de ProTrekkr existent :
- une version OSS
- une version Jack
La version hébergée sur GitHub est compatible Jack.

Exemple YouTube

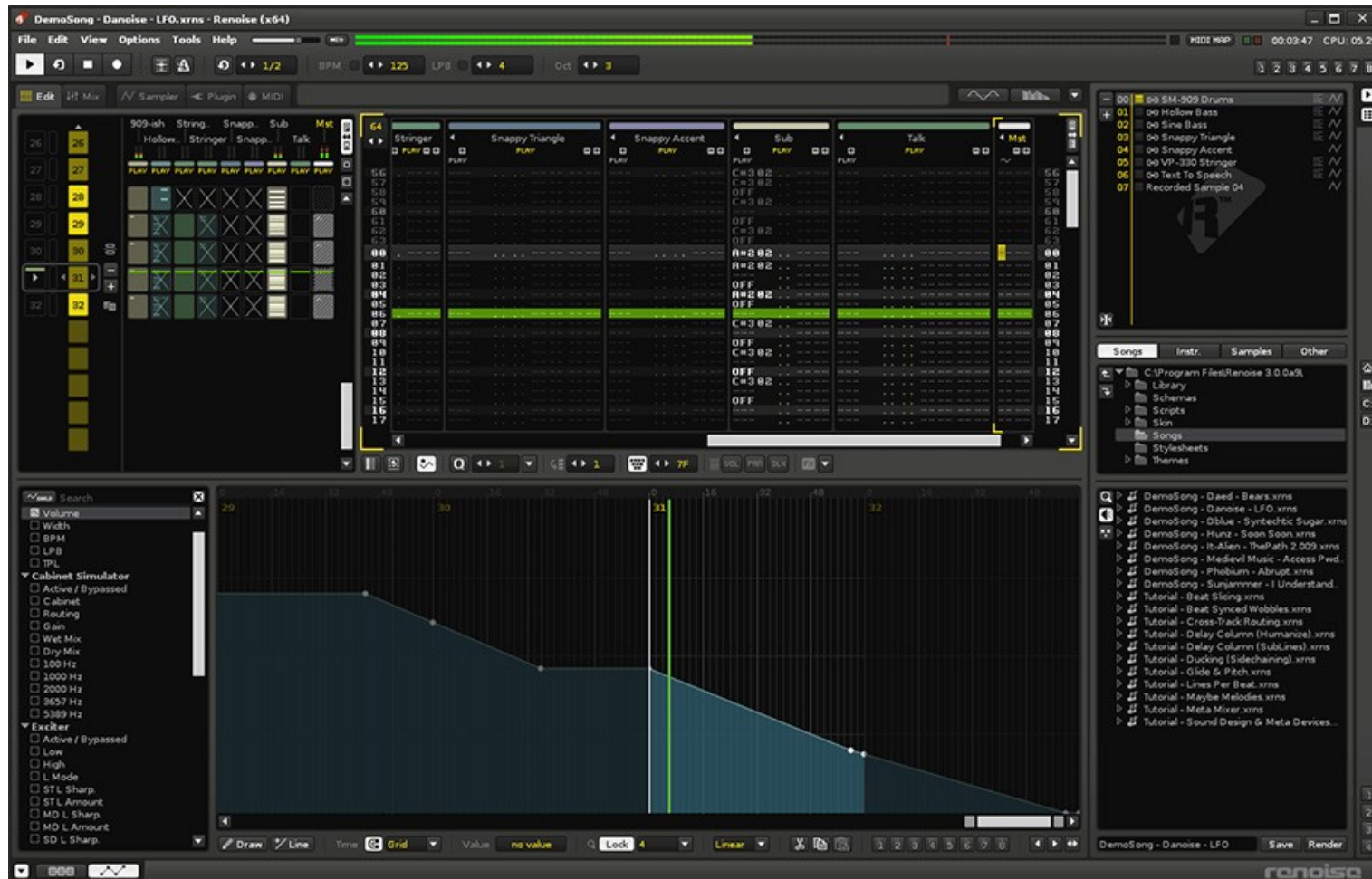
24/08/2013





Renoise

<https://www.renoise.com>





Divers

Des fichiers pour Protrekkr et MilkyTracker :

<http://modarchive.org/>

Rivendell – La radio Open Source

<http://www.rivendellaudio.org/>

Jack Net / Jamulus / Ninjam

La musique via internet

<http://lcon.sourceforge.net/>

<http://www.cockos.com/ninjam/>



Webographie

Presets en tout genre pour les outils Linux : Fichiers pour le mixage :

<https://musical-artifacts.com/>

Sources de samples :

<http://freesound.org/>

<https://archive.org/>

http://wiki.laptop.org/go/Sound_samples

Documentations de divers outils :

<https://en.flossmanuals.net/>

Site communautaire :

<http://linuxmao.org/Accueil>

<http://libremusicproduction.com/>

<http://www.linuxaudio.org/>

<http://linuxmusicians.com/>

Chansons de Nine Inch Nails :

<http://www.ninremixes.com/multitracks.php>

Différentes chansons :

<http://www.cambridge-mt.com/ms-mtk.htm>

Des ressources en live coding :

<http://sccode.org>

<http://users.sussex.ac.uk/~nc81/modules/cm1/workshop.html>