Teaching Sound Synthesis in C/C++ on the Raspberry Pi

Henrik von Coler David Runge

Technical University of Berlin - Audio Communication Group

18.05.2017

Course outline

- enable students of different backgrounds to develop standalone real-time sound synthesis projects in a free environment
- algorithms on a sample-wise signal processing level
- usual faculty tools not applicable
- C/C++ reasonably low-level and extensible

Alternatives

- Matlab[28]/ Octave[20] (not real-time)
- JUCE[34] (framework)
- Pd[30], Max/MSP[12], SuperCollider[**?**] (too high-level)
- Pd externals
- FAUST[6] (no functional programming background)

API

- JACK[18]
- JACK clients widely used in research (WONDER[16], SSR[13])
- simple, free, modular, de-facto standard

Hardware

- provide cheap, unified hardware for as many groups as possible
- Bela[4] (too expensive)
- Raspberry Pi 3[10] reasonably cheap and feature-rich
- Per-system price at 70€

OS

- Arch Linux ARM[2]
- Pacman[9]
- ABS[1]
- AUR[3]
- Background in Arch Linux based low latency setups[7]
- Systemd[11] based startup
- Pre-configured image[1]

[1] `https://www2.ak.tu-berlin.de/~drunge/klangsynthese`

Libraries

- jack2[18]
- jackcpp[29]
- sndfile[19]
- rtmidi[33]
- liblo[23]
- yaml[21]
- fftw[22]
- boost[5]

JACK settings

- starting JACK using systemd[31] and loginctl's linger[8]

```
/usr/bin/jackd -R \
  -p 512 \
  -d alsa \
    -d hw:Device \
      -n 2 \
      -p 64 \
      -r 44100
```

Starting projects

- starting projects after starting JACK

```
[Unit]
Description=Example project
After=jack@rpi-usb-44100.service
[Service]
ExecStart=/path/to/executable \
    parameter1 \
    paramter2
Restart=on-failure
[Install]
WantedBy=default.target
```

Examples: DSP

Ready-to-run examples provided in course repository[32].

- Processed Recording (wave-table, granular, vector, concatenative synthesis)
- Oscillators (substractive synthesis, analog modeling), filters (IIR, FIR, bilinear transform) and envelopes
- Additive Synthesis and Spectral Modeling (analysis-resynthesis)
- Physical Modeling (Karpus-Strong[17], bidirectional wave-guides[26])
- FM Synthesis (abstract example: Yamaha DX7)

Examples: Linux, hardware, networking

- Terminal, SSH (with X-forward) with examples for Linux, MacOSX and Windows
- Systemd: (Auto-)starting of JACK and projects
- Networking DHCP-based and creating hotspot
- Using ALSA tools to identify and setup devices
- GCC to compile programs (with rudimentary build scripts)

Vector Synthesizer[25]

- Oscillator -> joystick control in order to morph between four waves
- Moog filter (Oberheim variation): Switch between six different filter types (LP2, LP4, HP2, HP4, BP2, BP4)
- two envelopes (volume, filter)
- two LFOs (volume, filter cutoff)
- 10-12 voices polyphony (due to RPi performance limit)

Digital String Waveguide (Karplus Polyphon)[14]

- physical modeling synthesizer
- polyphonic Karplus-Strong algorithm implementation with fractional allpass delay
- String bending, sustain mode, different excitation signals and burst window functions
- Wah-Wah filter
- FFT convolution (guitar body IRs)

Wave Digital Filter Tonestack[24]

- Wave Digital Filter (WDF) Tonestack
- I2C to use analog potentiometers

Sinusoidal Synthesis with CQT sample analysis[15]

- Analysis: Matlab script to extract CQT spectra and store in a plaintext file
- One table per octave from example instruments
- Synthesis: Reading above tables and generating (interpolated) tables for every MIDI note hit
- Polyphonic, but high number of sines being processed, only a few notes could be played simultaneously

RoMaSynthi[15]

- polyphone software synthesizer
- generates sound in a combination of additive and subtractive synthesis

AKTpi[27]

- Tutorial creating a single-purpose Linux system dedicated to audio projects, using buildroot and targeting the Raspberry Pi boards.

OS

- Linux (Arch (1), Ubuntu (2), Debian (1))
- Windows 10 (2)
- MacOSX

IDE

- vim (2)
- Sublime (2)
- Visual Studio
- Atom

Debugging

- Visual Studio (1)
- cout

(Auto-)Starting JACK

- jackd (only testing)
- script (3)
- systemd
- qjackctl/ script

Audio interfaces

- Renkforce
- ESI Dr. DAC nano
- EC Technology USB Sound Card
- Focusrite Scarlett 2i2

Problems

- Unable to use internal audio card reliably
- JACK fails when connecting through WiFi
- File I/O
- Performance limitation
- JACK crashes
- Project optimization
- Hard to debug

Terminal/Visual

terminal	visual
100	0
80	20
90	10
90	10
95	5
100	0

Environment/Project

environment	project
50	50
20	80
30	70
10	90
60	40
30	70

Improvements

- Easier networking setup
- Less low-level audio/MIDI I/O
- More reliable setup
- Real-time kernel (?)
- More relevant C++ examples for audio programming

Hardware

- Setup proved to be pretty versatile
- Test more cheap USB sound cards
- Try Bela[4]

Preparation

- More debugging (gdb, integrated solutions)
- Better pre-configurated audio setup
- Elaborating on C++ programming examples
- Monitoring

Contact

	Henrik von Coler	**David Runge**
Mail:	voncoler@tu-berlin.de	dave@sleepmap.de
XMPP:		dvzrv@sleepmap.de
IRC:		dvzrv@{efnet,freenode,hackint,oftc}


Arch Build System - ArchWiki.
https://wiki.archlinux.org/index.php/Arch_Build_System.


Arch Linux ARM homepage.
https://www.archlinuxarm.org/.


Arch User Repository.
https://aur.archlinux.org/.


Bela homepage.
https://bela.io.


Boost C++ Libraries - Homepage.
http://www.boost.org.


FAUST - Homepage.
http://faust.grame.fr/.


Linux Audio Conference 2015 - Workshop: Arch Linux as a lightweight audio platform - Slides.
http://lac.linuxaudio.org/2015/download/lac2015_arch_slides.pdf.


loginctl man page.
https://www.freedesktop.org/software/systemd/man/loginctl.


Pacman homepage.
https://www.archlinux.org/pacman/.


Raspberry PI Homepage.
https://www.raspberrypi.org/products/raspberry-pi-3-model-b/.


systemd homepage.
https://www.freedesktop.org/wiki/Software/systemd/.

Cycling '74.

Cycling '74.
https://cycling74.com/.

Jens Ahrens, Matthias Geier, and Sascha Spors.

The soundscape renderer: A unified spatial audio reproduction framework for arbitrary rendering methods.
In Audio Engineering Society Convention 124. Audio Engineering Society, 2008.

Moritz Güldenring and.

Digital String Waveguide (Karplus Polyphon).
https://github.com/MoritzGue/Klangsynthese_JaMo.

William Arnold Rodewald and.

Sinusoidal Synthesis with CQT sample analysis.
https://github.com/WRodewald/Klangsynthese.

Marije AJ Baalman.

Updates of the wonder software interface for using wave field synthesis.
LAC2005 Proceedings, page 69, 2005.

Julius O. Smith David A. Jaffe.

Extensions of the Karplus-Strong Plucked-String Algorithm.
Computer Music Journal, 7(2):56–69, 1983.

Paul Davies.

JACK API.
http://www.jackaudio.org/.

Erik de Castro Lopo.

Libsndfile.
http://www.mega-nerd.com/libsndfile/.


John W. Eaton.
GNU Octave.
https://www.gnu.org/software/octave/.


Clark C. Evans.
YAML: YAML Ain't Markup Language.
http://yaml.org/.


Matteo Frigo and Steven G. Johnson.
FFTW Fastest Fourier Transform in the West.
http://www.fftw.org/.


Steve Harris and Stephen Sinclair.
liblo Homepage: Lightweight OSC implementation.
http://liblo.sourceforge.net/.


Chris Hold and Christoph Hohnerlein.
Wave Digital Filter Tonestack.
https://github.com/chohner/rasp_syn.


Corvin Jaedicke and Alberto Monciero.
Vector Synthesizer.
https://gitlab.tubit.tu-berlin.de/c.jaedicke/Klangsynthese_PI.


Matti Karjalainen, Vesa Välimäki, and Tero Tolonen.
Plucked-string models: From the Karplus-Strong algorithm to digital waveguides and beyond.
Computer Music Journal, 22(3):17–32, 1998.


Konrad Krenzlin and Karl Manuel Weber.
AKTpi.
https://krenzlin.github.io/AKTpi.


MathWorks.
Matlab.
https://www.mathworks.com/products/matlab.html.


Alex Norman.
JackCpp.
http://www.x37v.info/projects/jackcpp/.


Miller Puckette.
PureData.
https://puredata.info.


David Runge.
uenv homepage.
https://git.sleepmap.de/software/uenv.git/about/.


David Runge and Henrik von Coler.
AK-Klangsynthese Repository.
https://gitlab.tubit.tu-berlin.de/henrikvoncoler/Klangsynthese_PI.


Gary P. Scavone.
RtMidi.
http://www.music.mcgill.ca/~gary/rtmidi/.


Jules Storer.
JUCE.
https://www.juce.com/.