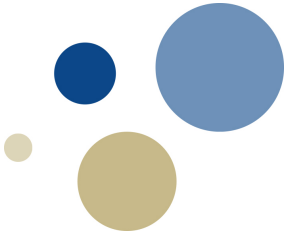
Norwegian University of
Science and Technology



Physical Computing Workshop: Day 2

Embedded Hacking

Anna Xambó
Department of Music, NTNU

16 October 2019

Warm-up Activity
Yesterday’s blogpost and performance

Learning Outcomes

By the end of the session, you will be able to...

- Get familiar with Puredata and Bela and how to work with the two.
- Get a sense of how what is an IDE and how the Bela IDE works.
- Be able to prototype simple “patches” using Puredata, Bela and the breadboard.
- Explore mappings from sensor data to sound.
- Explore adapting some concepts from the “Victorian synthesizer” to the digital domain.
- Demonstrate a custom-made musical instrument in a performance setting.

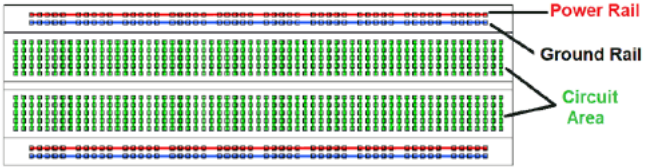
Preparation: Read/skim through the following readings

- Pure Data
  - Puckette, M. (1997): Pure Data: Another Integrated Computer Music Environment
  - Chapter 9: Starting with Pure Data. Farnell, Andy. Designing sound. MIT Press, 2010.
  - Chapter 10: Using Pure Data. Farnell, Andy. Designing sound. MIT Press, 2010.
  - Chapter 11: Pure Data Audio. Farnell, Andy. Designing sound. MIT Press, 2010.
- Bela
  - McPherson, Andrew, and Victor Zappi. "An environment for submillisecond-latency audio and sensor processing on BeagleBone Black." Audio Engineering Society Convention 138. Audio Engineering Society, 2015.

Puredata: Template patterns

— See folder “Puredata” in code day 2: Oscillators, Time, Random.

Breadboard



Source: `https://components101.com/misc/breadboard-connections-uses-guide`

Block I
BELA: Overview

What is Bela?

- An environment for ultra-low-latency processing of audio and sensor data on embedded hardware.
- Based on the low-cost BeagleBone Black single-board computer.
- A custom expansion board features stereo audio and 8 channels each of 16-bit ADC and 16-bit DAC for sensors and actuators.
- It achieves latency as low as 80 microseconds (0.08 milliseconds).
- It combines the best aspects of embedded Linux systems and dedicated microcontrollers for real-time audio.

McPherson, Andrew, and Victor Zappi. "An environment for submillisecond-latency audio and sensor processing on BeagleBone Black." Audio Engineering Society Convention 138. Audio Engineering Society, 2015. [1]

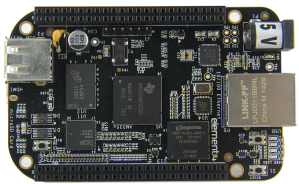
Bela vs Arduino

— In general, microcontroller platforms offer easy connections to hardware sensors and predictable timing, but have limited computing power.

— Embedded computers benefit from the ability to use familiar software tools (Pd, SuperCollider, ChucK, Python) and from the resources of a general-purpose OS, including file I/O and networking. They are optimised to balance many simultaneous processes but no guarantee to prioritize audio performance.

— Arduino and similar microcontrollers are often connected by a serial port to connect to mobile devices / embedded computers. This limits sensor bandwidth causing low bit resolution of sensor data.

BeagleBone Board



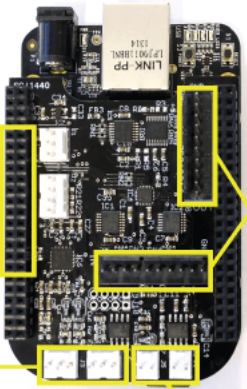
— It is a single-board computer: 1GHz ARM Cortex-A8 processor, 512MB of RAM, 4GB of onboard storage.

Bela

— It is a custom hardware expansion board (“a cape”) which provides stereo audio input and output, plus 8 channels each of 16-bit ADC and 16-bit DAC for sensors and actuators.

— The board also contains onboard stereo 1.1W speaker amplifiers for making self-contained instruments.

Bela Hardware



16 Digital In/Out
Connect buttons, LEDs, motors, and more.

8x Analogue In/Out
Connect devices such as accel-erometers and gyroscopes, and sensors for distance, pressure, stretch, vibration, and more.

Audio In/Out
Connect microphones, pickups or your own audio-rate signal to the audio in. Send the audio out to your head phones, or another device.

Speakers
Connect speakers straight to the board - amplfiers are built in

Bela Documentation

- Bela wiki on GitHub: `https://github.com/BelaPlatform/Bela/wiki`
- Example projects and tutorials: `https://github.com/BelaPlatform/Bela/wiki/Example-projects-and-tutorials`

Block II
BELA IDE

Main functionalities

— IDE stands for “integrated development environment”, which typically includes a source code editor, a debugger and build automation tools.

— Bela IDE documentation:
`https://github.com/BelaPlatform/Bela/wiki/Bela-IDE`

Block III
Pd + BELA

Hello World!

— See folder “Bela” in code day 2.

Block III
Fieldwork

Fieldwork II: Embedded music listening

— Development of a patch that applies music listening concepts.
— Group rehearsal so that each team member has a part or an instance of the music instrument.

Block IV
Rehearsal and Performance

References

[1] Andrew McPherson and Victor Zappi. “An Environment for Submillisecond-Latency Audio and Sensor Processing on BeagleBone Black”. In: *Audio Engineering Society Convention 138*. Audio Engineering Society. 2015.