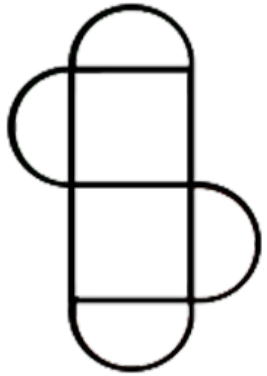
# CMLS - Homework - 2023/24

## ANSAL Three Men Band Project



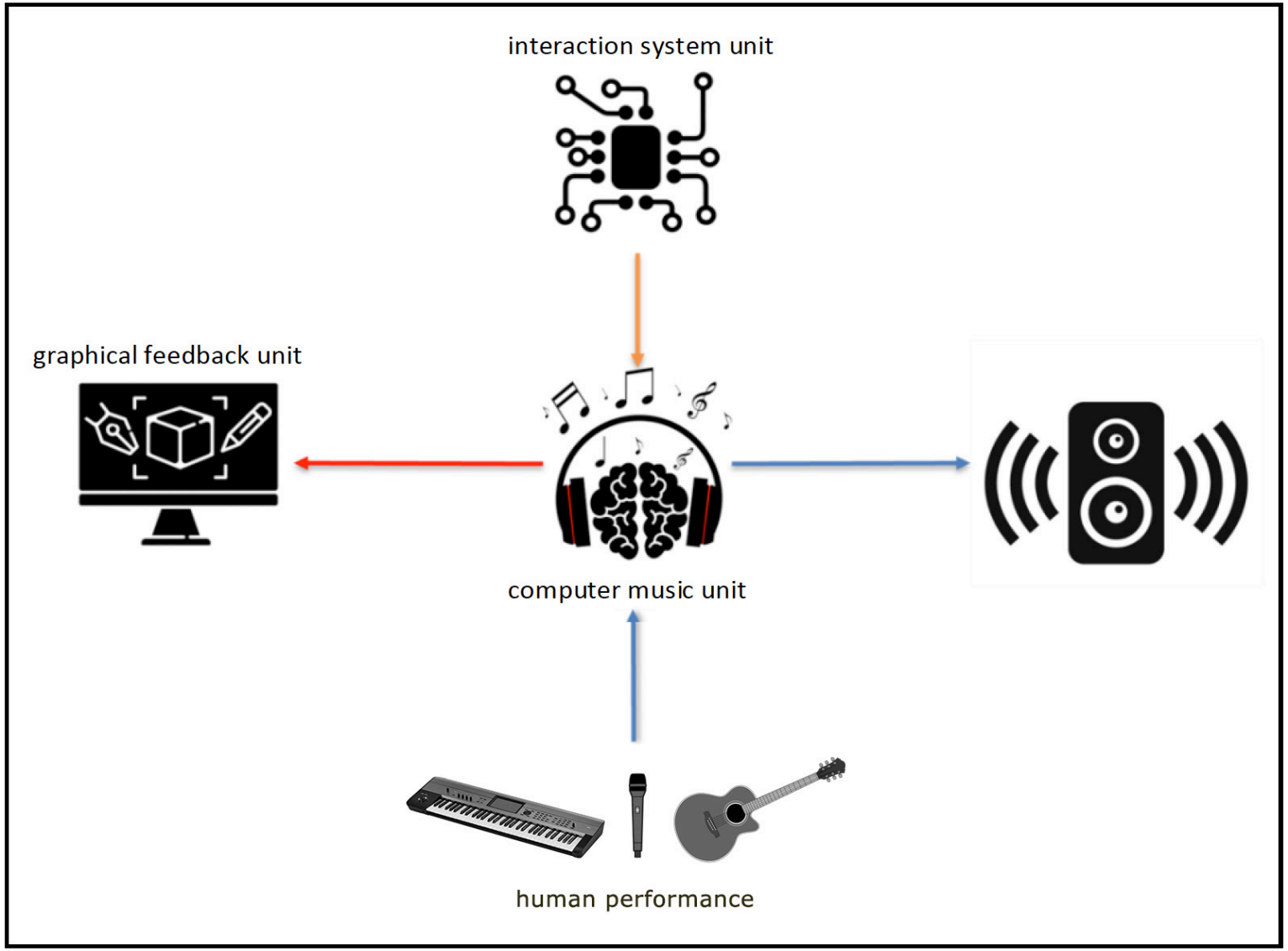
**AN - Angelo Antona**

**SA - Salvatore Benvenuto Zocco**

**AL - Alessandro Manattini**

# 1. The Main Goal

At ANSAL, we are all musicians, singers/songwriter, producers and sound engineers, and it was clear to us, from the very beginning, that we wanted to build a system that could help us play, in real time, live, or in Studio


interaction system unit
graphical feedback unit
computer music unit
human performance

## 1.1 Production Objective and Sound Design

Three modules: one for the voices, a second for keyboards, a third for guitar. Rhythm and drum tracks had to be sequenced. The bass track had to be assigned to keyboard or guitar.

## 1.2 Module Assignement

We decided to write a **Vocoder**. For such a task, **JUCE** seemed us to be the most logical choice. **SuperCollider** seemed to us more apt for writing synthetic **keyboard sounds**. Regarding **guitar**, pretty soon **SuperCollider** revealed to be geared for the purpose, with a "SoundIn" function that turns the incoming audio signal into a regular Ugen.

# 2. Guitar Synth

We’ve called it “Guitar“ because the way we used it, but it could be used with any audio input signal.

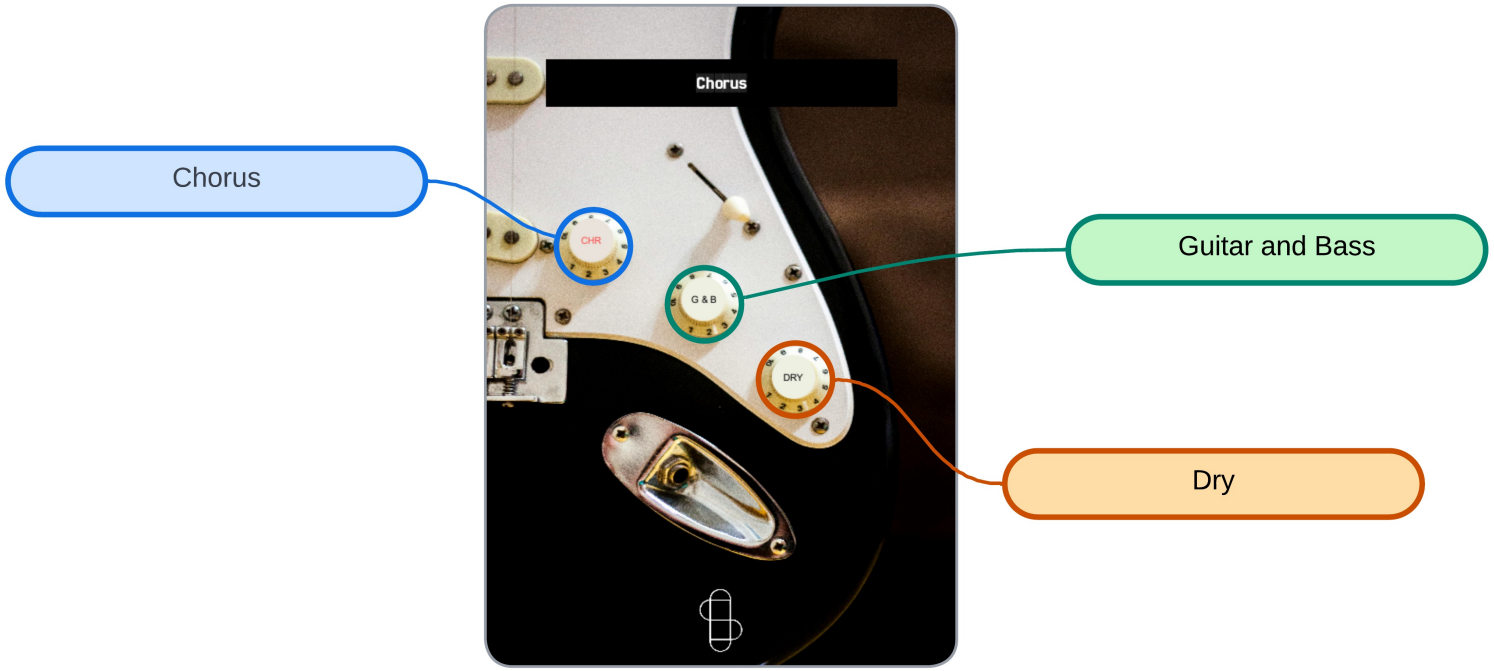
Care has been put in the choice of the interface, the MMA-A, to ensure a stable conversion, and the mic, 4099, a precise supercardioid, both from DPA Microphones.


## 2.1 The Sound Algorithms

Three families: Guitar and Bass Synth, Chorus/Flager, and a Clean Sound.

## 2.2 Guitar GUI

For our demo, we implemented a control interface were we assigned the three sounds needed.



# 3. Vocoder

Visualization section: output waveform and parameter to change the x and y axys of the visualization.

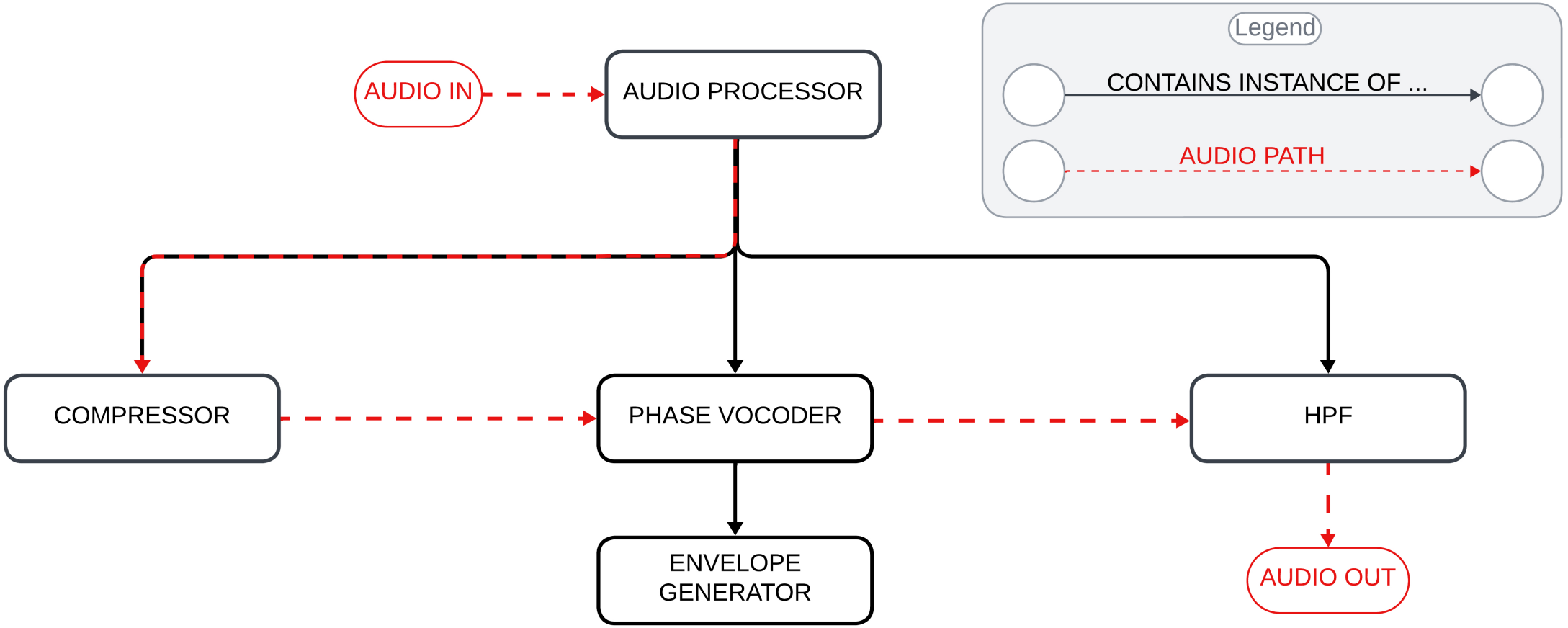
Vocoder envelope parameters.


Correlation between past samples and the new one.

Set the level of "normalization" of the input signal in order to make it more uniform in amplitude.

Vocoder envelope parameters: set the HPF cut frequency improoving speech intellegibility.

# 3. Vocoder


Legend
CONTAINS INSTANCE OF ...
AUDIO PATH
AUDIO IN
AUDIO PROCESSOR
COMPRESSOR
PHASE VOCODER
HPF
ENVELOPE GENERATOR
AUDIO OUT

# 4. Synth and System Integration

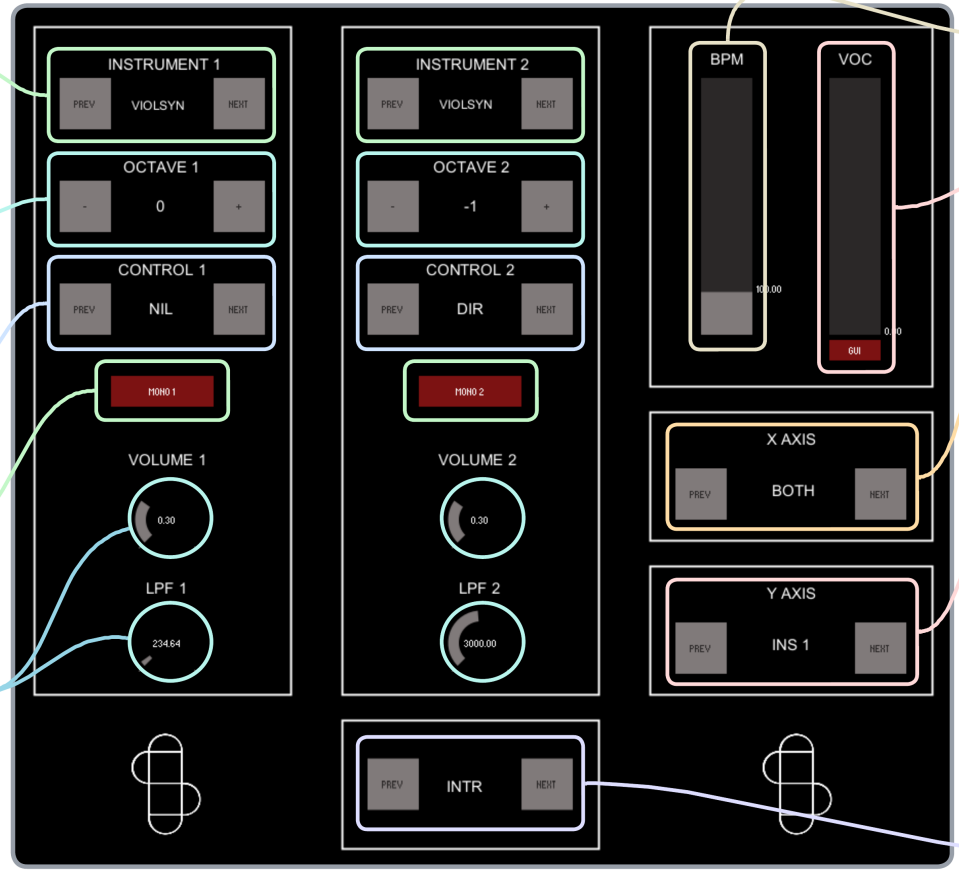
It allows to chose the sound to play for each instrument.

It allows to shift the octave of each instrument independently

It allows the control pedal to be connected to the LPF cutoff for each instrument. The mapping can be either direct or inverse.

It allows you to set the instruments to monophonic mode.

It allows you to manually set and display the levels of volume and low-pass filter frequency.




It allows you to set the BPM at which the sequences play.

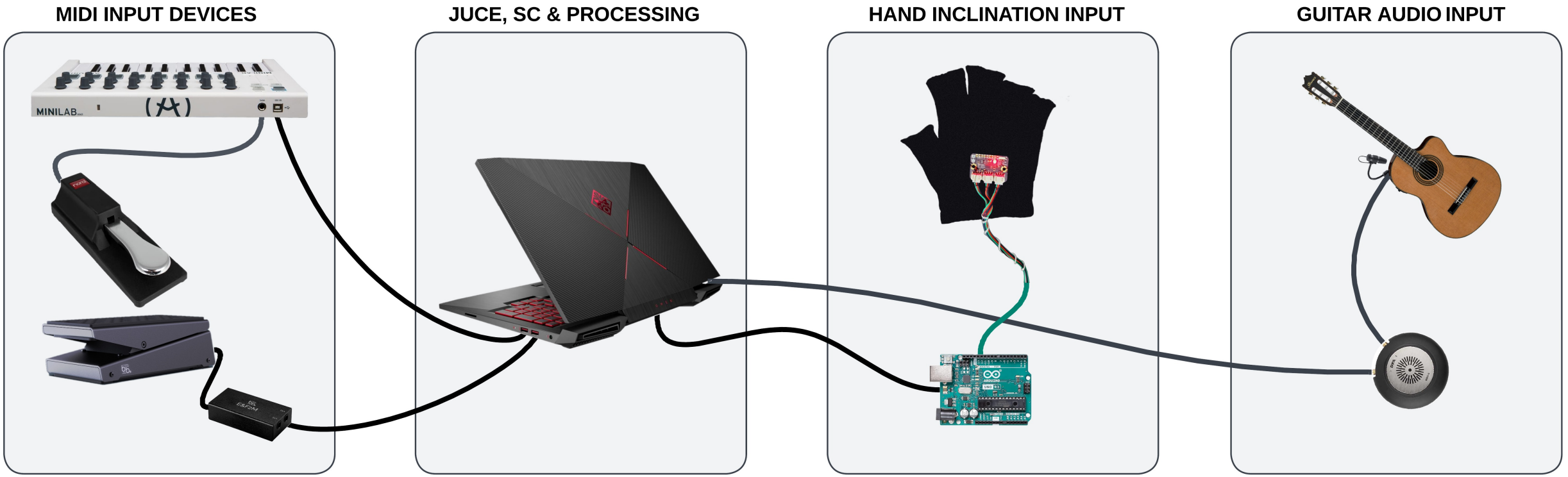
It allows you to view and set the input gain of the signal going to the vocoder.

It is the mapping of the glove's tilt along the x-axis (commonly referred to as roll). It allows you to vary the pitch of one or both available instruments.

It is the mapping of the glove's tilt along the y-axis (commonly referred to as pitch). It allows you to vary the cutoff frequency of one or both available instruments.

It allows you to select one of the preset settings.

# Hardware Configuration



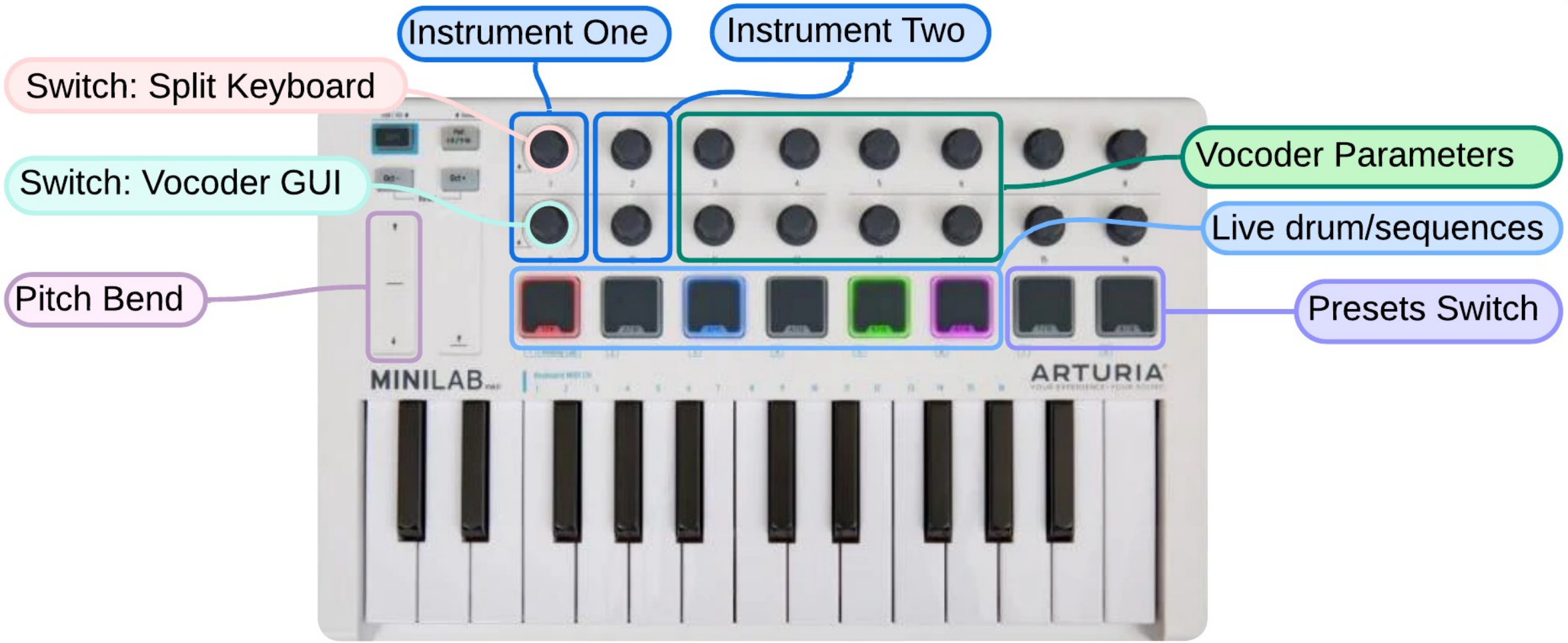
### 4.1.1 Hardware Configuration

**MIDI Input Devices**: MIDI keyboard, Sustain Pedal, Volume control pedal (used for MIDI CC control) with Analog to MIDI signal adapter

### 4.1.2 Hand Inclination Input

The system takes input from the hand movements of the keyboardist. Hardware devices are: **Accelerometer**, Mounted on a glove, and **Arduino Uno**, Required to derive angle data from the accelerometer coordinates and to interface the accelerometer with the system.

# 4.2 Synthesizer Features



## 4.2.1 Sound Patches Details

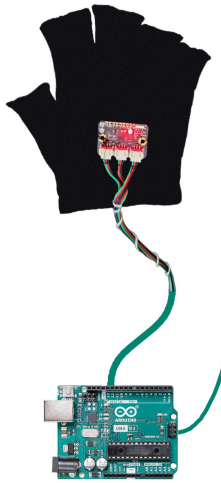
**Rhodes**: RHODES1 and RHODES2. **Bass Synths**: BASSYN1, BASSYN2, BASSYN3, and BASSIMP. **Lead and Wave Synths**: LEADSCR, TRIWAVE, and SAWWAVE.

## 4.2.2 Preset Details

- **Mono Bass, Drum, and Synth**
- **Theremin-like Setup**
- **Synth and Drum**

**Note**: The ***Keyboard Split*** can be activated/deactivated with a simple, dedicated procedure

# 4.3 Implementation Details





ArduinoAdapter

OSCCommunication

GUI

Main

DrumMidi

Knobs

PresetManaging

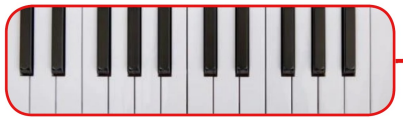
Inst_Midi

DrumSequencies

DrumSynthDef

Inst_SynthDef

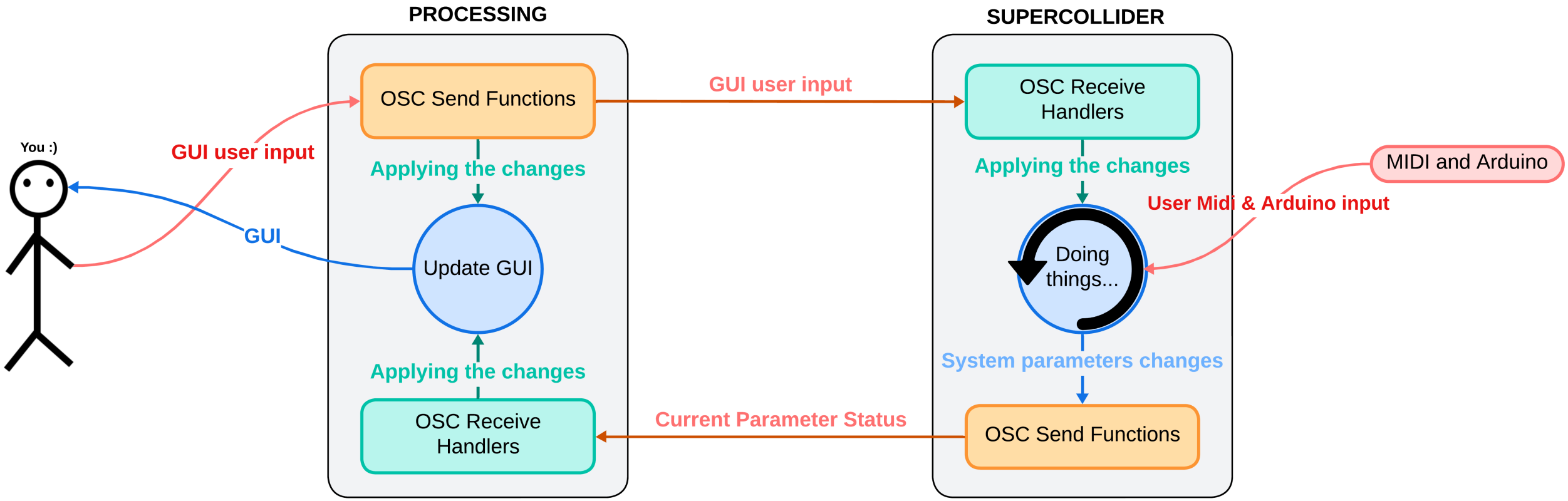
VST3Adapter



# 4.4 Interaction with Accelerometer and Arduino

Regarding the implementation of the glove, the input management is almost entirely handled in SuperCollider. However, the data received in SuperCollider is not the raw 3-axis accelerometer data. The accelerometer detects acceleration along the three Cartesian axes, whereas SuperCollider receives acceleration relative to the hand's orientation. To perform this conversion, we used formulas typically employed for managing drone orientation in flight.

# 4.5 Communication between SuperCollider and Processing



## 4.5.1 Processing

Processing generates the GUI, including buttons, sliders, and knobs for controlling parameters such as volume, low-pass filters (LPF), instrument selection, octaves, control pedals, and presets. Excluding initialization and support functions, the code can be grouped into the following main sections:

- **Communication Management with SuperCollider**
  - Sending OSC Messages
  - Receiving OSC Messages
- **User Input Management**
- **GUI Update**

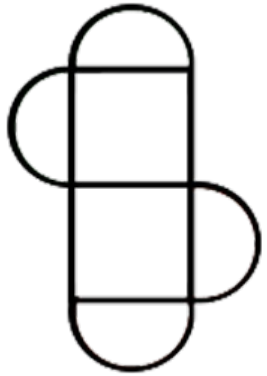
## 4.5.2 Supercollider

In SuperCollider, we can also divide communication management into main sections:

- **Receiving OSC Messages**
- **Sending OSC Messages**
- **GUI Update Routine**

# CMLS - Homework - 2023/24

## ANSAL Three Men Band Project



**AN - Angelo Antona**

**SA - Salvatore Benvenuto Zocco**

**AL - Alessandro Manattini**