Michael Hemingway
Assignment THEN - Proposal
CART 360, Fall 2018
Nov. 28, 2018

“Pedal | Air Guitar”
github.com/stockhuman/CART-360/ASSIGNMENTS/THEN

Project Description

Having fast run out of time and inspiration, I stumbled on something related to the electric guitar I got my sister for Christmas: an Arduino Pedal! The idea is to create a guitar pedal that interfaces with an online component, to modify performance in real time.

Effect pedals are the most diverse category of electric instrument peripherals, offering unique effects, tonal control, amplification and combinatory powers to the natural sound of the instrument. These can be chained, feeding the audio signal from one device to another and in turn creating exponentially more unique sounds.
The variable effect pedals (one that does more than passively alter the sound going in (ex. *Bazz Fuzz circuit*)) are rather common, yet pedals that dabble in domains beyond audio modification are fairly rare.

The Context

The key interaction here is that of the player and audience collaborating (or disrupting) via the pedal. The wifi interface of the pedal and the controllable effects add a dimension of expressivity where there is usually none; the audience, even without musical mastery can contribute to the sound of a live performance.

The Relationship

The relationship between the pedal, the performer and the user that I wish to create is that of a sense of participatory agency in the creative process, and the dissolution of distance between performer and user. A sense of presence, as if a jam could happen anywhere between strangers or friends (mode dependant), from anywhere in the world.

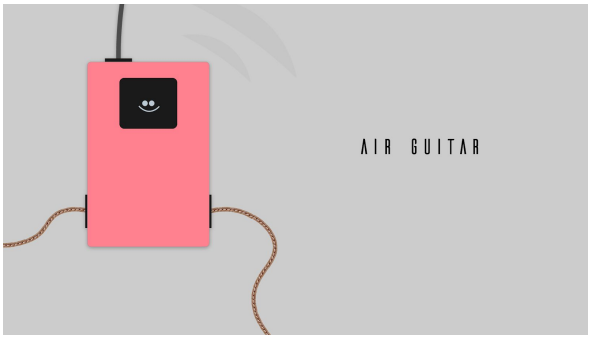
Empowerment

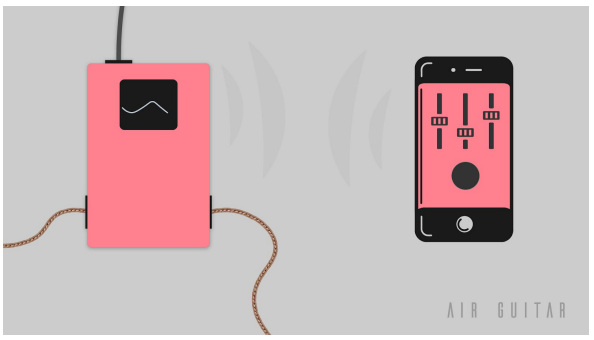
The impetus for this design was, as mentioned above, to be able to play and contribute to my sister's guitar playing, even if I can't play the guitar myself. In that way, I wish for the device to be able to break down the skilled / unskilled barrier, to facilitate collaborative creative expression, even if minimally, between the normally passive listener and the normally in-control musician.

Meaning

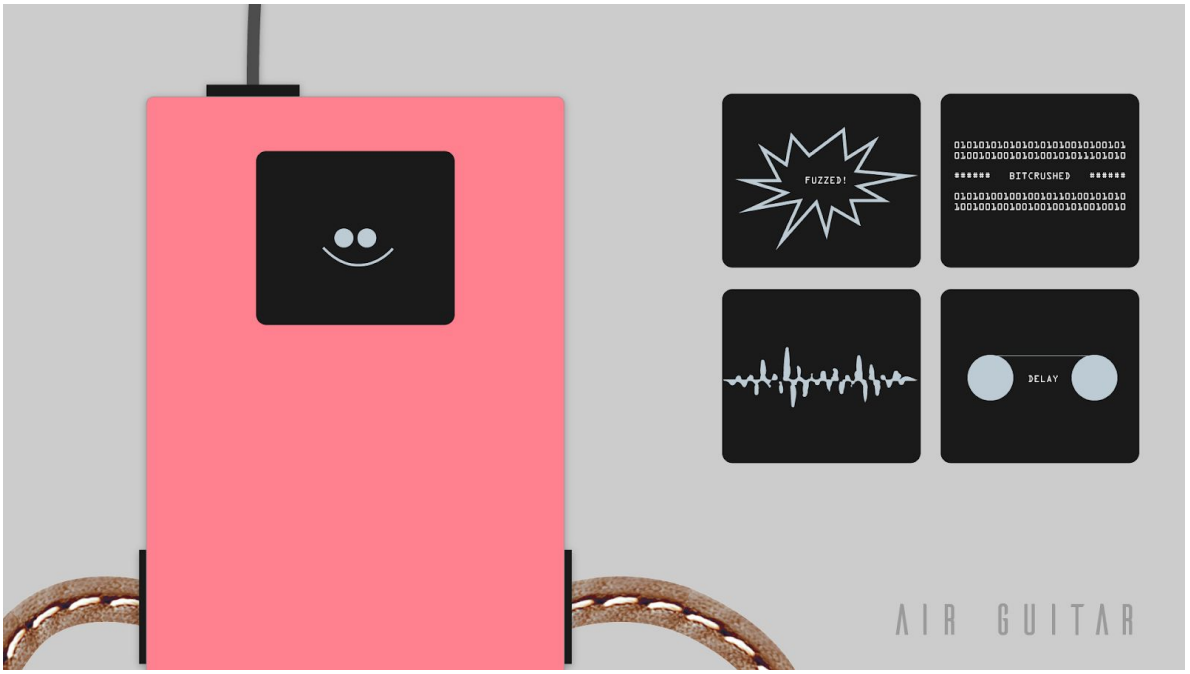
I suppose the meaning one could derive from this machine is that the creative potential of any individual needn't be squandered for lack of technical skill, and that taste - musically or otherwise - is the true barrier to creating great art. In realising this, tool such as this one that bridge the gap between skilled and unskilled level two the persons to an equal playing field of creative expression.

Intention & ID/XD

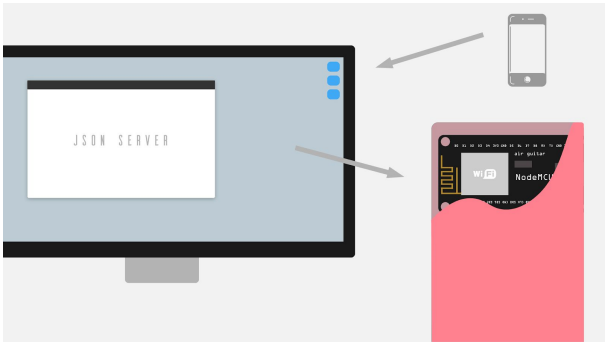




Air Guitar, as I've dubbed it, is a wirelessly controlled effects pedal, for the purpose of creating a listener-and-player connection, be it for fun or musically enriched performances

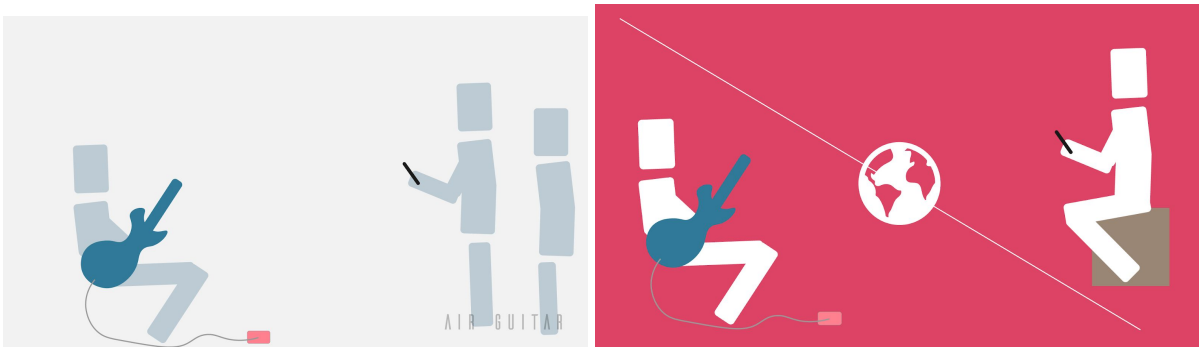


The pedal features a small, 128x128 OLED screen to replay to the player what their audience has elected to do with the raw audio.

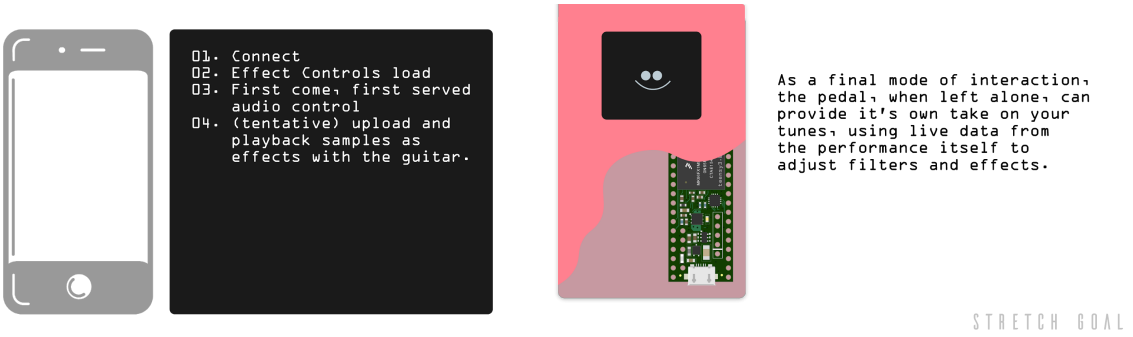


The devices used to interface with the chip navigate to a specified url, and upon connection, the wifi chip within the pedal will relay the user(s)' effects and timing to the MC.

The pedal, being a pedal, can interface with the rest of the performer's toolchain, colouring the sound further.



Player and user(s) enter a collaborative space, regardless of user skill. As the unit pings a specified server and the handheld controlling phones do as well, this collaborative space needn't exist within the confines of a room nor continent.



Finally, the pedal can riff off of the player's input on it's own, using performance data past and present to colour filters and delays automatically.

Similar Work

1. pedalSHIELD DUE Arduino Guitar Pedal
   I came across the pedalshield as the first thing that resembled what I was after. It's a programmable, arduino-based effect pedal for bass and guitar. This is the most technically similar project I've come across, in that it fits my intended use case and form factor, yet instead boasts more of a 'generic pedal that you can program' idea. The pedal is a shield attachment to various Arduino models (not just the DUE), providing ADC, DAC, Audio I/O and control to the otherwise underpowered arduino. This project includes a fairly robust audio library of computationally inexpensive digital effects.
2. DiscoJar Sound Reactive lamp
   Another technically similar project, this time involving the sound analysis object in the Teensy audio library to adjust the colors of an RGB LED lamp. This lamp interfaces with the internet to change modes, and uses the same ESP8266 at the core of it's wireless stack. Beyond that simple audio-in, lightshow-out, there is not much else to be described.
3. Absolut Choir and Absolut Quartet

The most conceptually similar work I've found; these projects essentially aim to transform remote user input (over the internet, and in the case of the choir, non-musical data) into commands for the sound synthesis of these machines. Developed by a team that later started Teenage Engineering, both installations were featured in galleries in Stockholm on semi-permanent exhibit. Unfortunately the online interface has been retired, and the projects are no longer accessible to the public.

Absolut Choir (the more recent of the two, circa 2007) consists of an array of brightly colored speakers shaped in the form of faces, harmonizing together in what sounds like a IBM Daisy Bell rendition of incoherent yet consonant classical singing.

The other, Absolute Quartet, demoed in 2002, consists of a massive array of ping pong balls and drums, each tuned to a different note. Little levers actuate the launch of the balls so as to strike the drum, in tune with a keyboard performance by someone on the internet accessing the installation's portal. The key technical difference here is that of the artificial intelligence employed in Absolut Choir to synthesize sound from the phrases submitted by participants.

On top of being aesthetically (and acoustically) inventive, these two projects

Differentiation

The main difference from the projects listed above is the networked connectivity. The role of the programmable hardware in their use cases is a static, 1-user-1-sound setup where effects are loaded in, modified and reprogrammed between performances. With regards to the Choir projects, the pedal is an intermediary, and not the end result: these machines generated sound via their input, much like a remotely controlled instrument. The pedal I intend to build instead colors the performance, and may optionally do so on its own, but always with the guitar or bass input as a base. The userspace I intend to create, the emotional placement of the user and guitarist is far more personal in this way, where one isn't simply ordering a machine to play sounds, but instead collaborating with another human being on hopefully great music.

Feasibility

I've written this section with the unfortunate understanding that I'm already behind, and that I've missed prototype one. I've spent the last two weeks building the platform to have this system functional, and while some time was wasted with chips that ultimately were unsuitable (or broken, in the case of the Teensy Audio shield), I've built a system out of a core Teensy 3.6, an ESP8266-01 NodeMCU, SSD1351 display and a PCM5102 I^2S board. Prototyped and functional is Wireless transmission of commands via the ESP, audio passthrough and playback, SD card read and write and display output.

The challenges ahead

1. Reducing background noise.
2. Building the web UI
3. Additional effects
4. WiFi Latency (Input via directly served html or json communication)

References

https://tttapa.github.io/ESP8266/Chap14%20-%20WebSocket.html - ESP8266 Websockets
https://www.youtube.com/watch?v=Q0eUoFIXrWE - ESP8266 WiFi Direct
https://www.pjrc.com/teensy/gui/index.html Teensy Audio Library GUI
https://github.com/kirberich/teensy_ssd1351 Teensy SSD1351 (Display) library
https://www.youtube.com/watch?v=EnfjYwe2A0w Creating an ADC / ADC basics