Musical Display Board

Practical 2

Student Number: 170005748

Student Name: Calum Davison Bowden

Abstract (maximum 100 words)

This project aims to create a device which is interactable with minimum physical input as a result of COVID-safe interactions. Photoresistors are used to detect input from the user and create sound, similar to a Theremin. The sounds made can then be mixed using a simple mixing display and played back to the user. The intention was to create something that multiple people could contribute to, or just have a play with something a bit different.

Interaction Overview

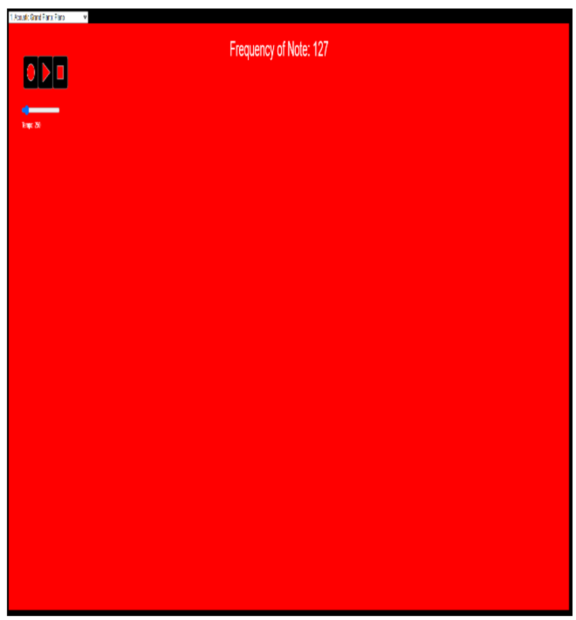

Figure 1

Figure 1 shows an image of the device setup and display. The user interacts by moving their hand in and out of the boxes, blocking or unblocking the light which is detected by the photoresistor on the top of the box. The screen display allows the user to select various instruments, play, record, and delete different tracks.

Parts List

Hardware

- Arduino Uno
- 2 Bright White LEDs (in this case one is substituted)
- 2 Photoresistors
- Piezo Speaker
- Potentiometer
- 2 220Ohm Resistors
- 1 4kOhm Resistor
- Wires
- Screen
- Tin Foil (or other reflective material, optional)

Other

- Arduino IDE

Process

As shown in Figure 1, the device is put together using 2 LEDs which shine directly onto a photoresistor in an enclosed box to avoid light outside of the device affecting readings (i.e. Sunlight). The boxes were sized to allow a hand to move easily in and out without touching the box. Tin foil is used to provide a reflective surface for the inside of the box, and so that even when covering the LED, some light manages to reach the photoresistor. These photoresistors are connected to the analog in pins so the Arduino can take readings of the light values. Only one bright white LED was included in the pack, so green was used for the "volume" box. This was chosen because sunlight tends towards the green/yellow end of the spectrum and, from research [4], photoresistors tend to be more sensitive to light of this wavelength. The light is placed on the bottom because it feels more intuitive to cover up a light and get a response, than to cover the photoresistor, as it is easier to see that there is less light bouncing around the box.

A piezo speaker is attached to provide real-time output of the readings and give a sense of what the raw input sounds like, as this is converted on the display end. The "volume" photoresistor is connected to serve as resistance for the speaker and thus controls the volume of the speaker directly. Additionally, a potentiometer is wired up as the resistor to ground for the "frequency" photoresistor to allow the sensitivity to be altered to find the best range. During testing this was found to be 10kOhms, or the maximum of the potentiometer. Figure 2 shows a wiring diagram for the current set up:

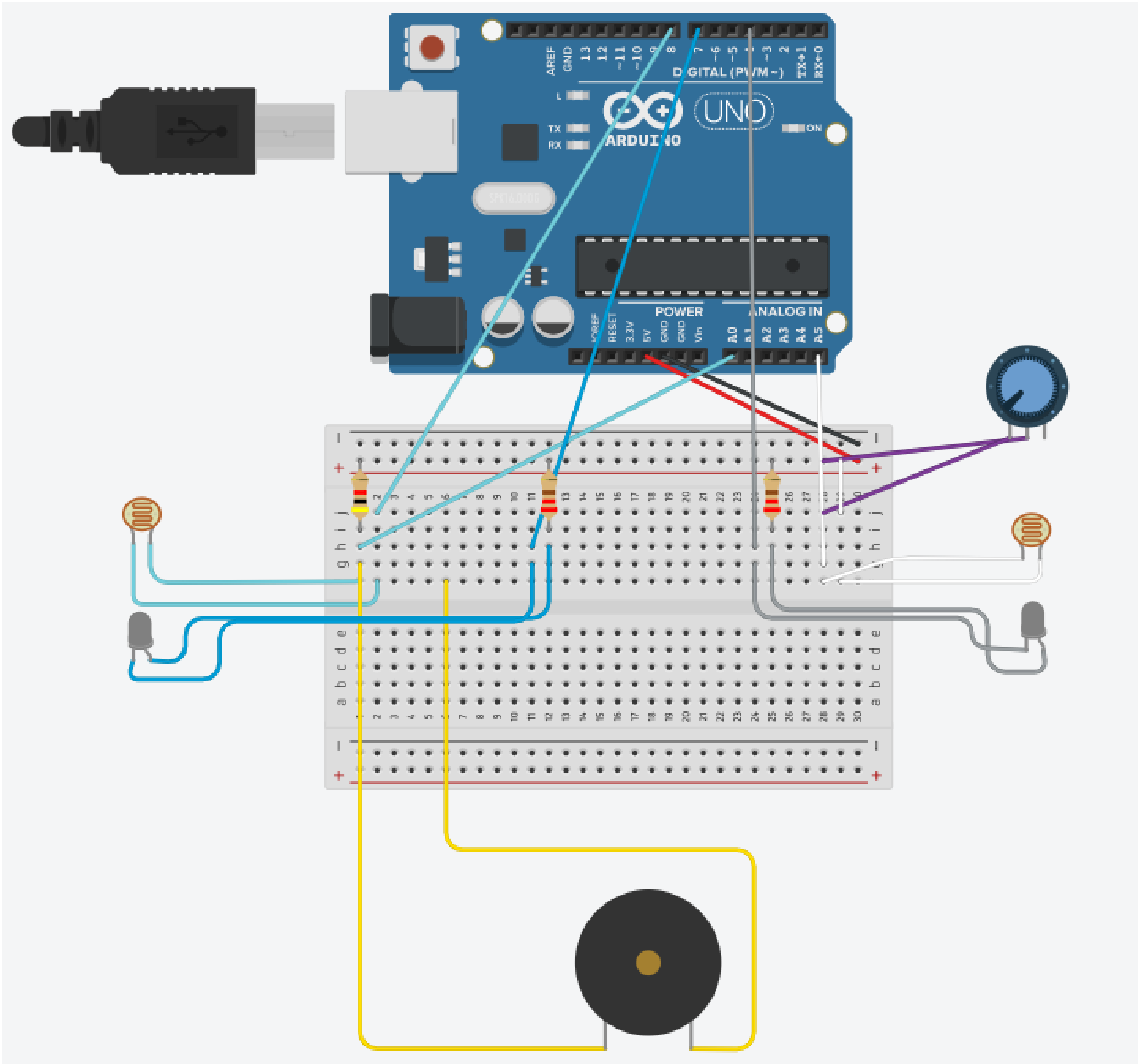

Figure 2 Wiring Diagram (made using tinkercad [3])

The Arduino has a setup method on first start to calibrate the photoresistors – find the minimum and maximum light values that both receive as each photoresistor can be slightly different. Every loop it also updates in case the lighting has changed so that readings are correct. The only caveat is that it will only widen the range of values from min to max. If the lighting were to get significantly dimmer, a reset would be required.

The display board uses a library and API called WebAudioFont [1] which supplies just about every MIDI instrument and handles playing the instruments through the WebAudio API. The user can select any instrument from the 1000+ and the frequencies recorded through the physical device will be converted to notes of that instrument. These sequences are then saved and can be played back. Multiple sequences can be made, of multiple instruments, and played back together. The update frequency of the Arduino can also be set from 100ms to 2000ms allowing for different “tempos”. The colour of the background also dynamically changes based on the frequency of note played using a colour mapping algorithm from [2].

Challenges & Key Points

Getting the lighting correct for the photoresistors was the most challenging part. First sunlight kept altering the readings, then in a box, it would only read max or min, until resorting to tin foil to provide a more reflective surface. It was a goal to allow exporting to MIDI, however, due to the differing tempos, this would have been a challenging and time-consuming task, but given more time would definitely like to explore this. Another challenge was making the output audio sound like a theremin with the linear change between frequencies. This was attempted using MIDI pitch bend events but didn’t quite work as intended and was difficult to do in real-time.

References

[1] https://github.com/surikov/webaudiofont

[2] https://www.particleincell.com/2014/colormap/

[3] https://www.tinkercad.com/

[4] https://electronics.stackexchange.com/questions/211894/what-colors-or-wavelengths-are-photoresistors-sensitive-to

Appendix

This will not be included in the page count of your report.