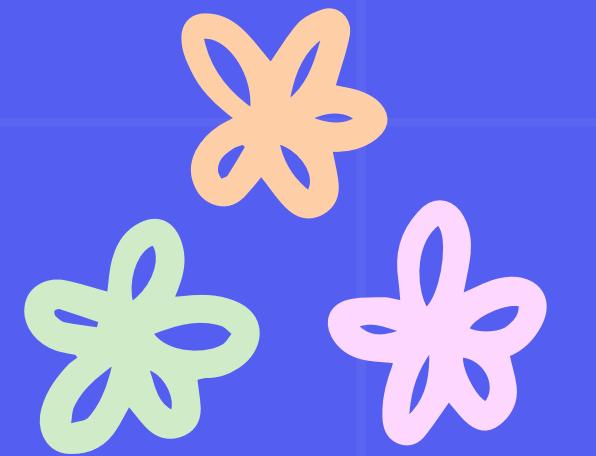


Final Project

Presentation



CHARLIE
SHI



Project Overview

- Real-time music effects controller
- Built with C++ and Python on Raspberry Pi
- Uses physical buttons to control playback and audio effects
- Output shown via LED matrix (MAX7219)





Features

- Start / Pause playback (GPIO 18)
- Stop playback (GPIO 17)
- Pitch shift (+400 cents) toggle (GPIO 22)
- Distortion toggle (GPIO 13)
- Lower pitch toggle (-300 cents, internal logic)
- LED display shows active effects

How It Works

GPIO

Buttons use wiringPi for GPIO input

C++

C++ handles playback logic with aplay

Python

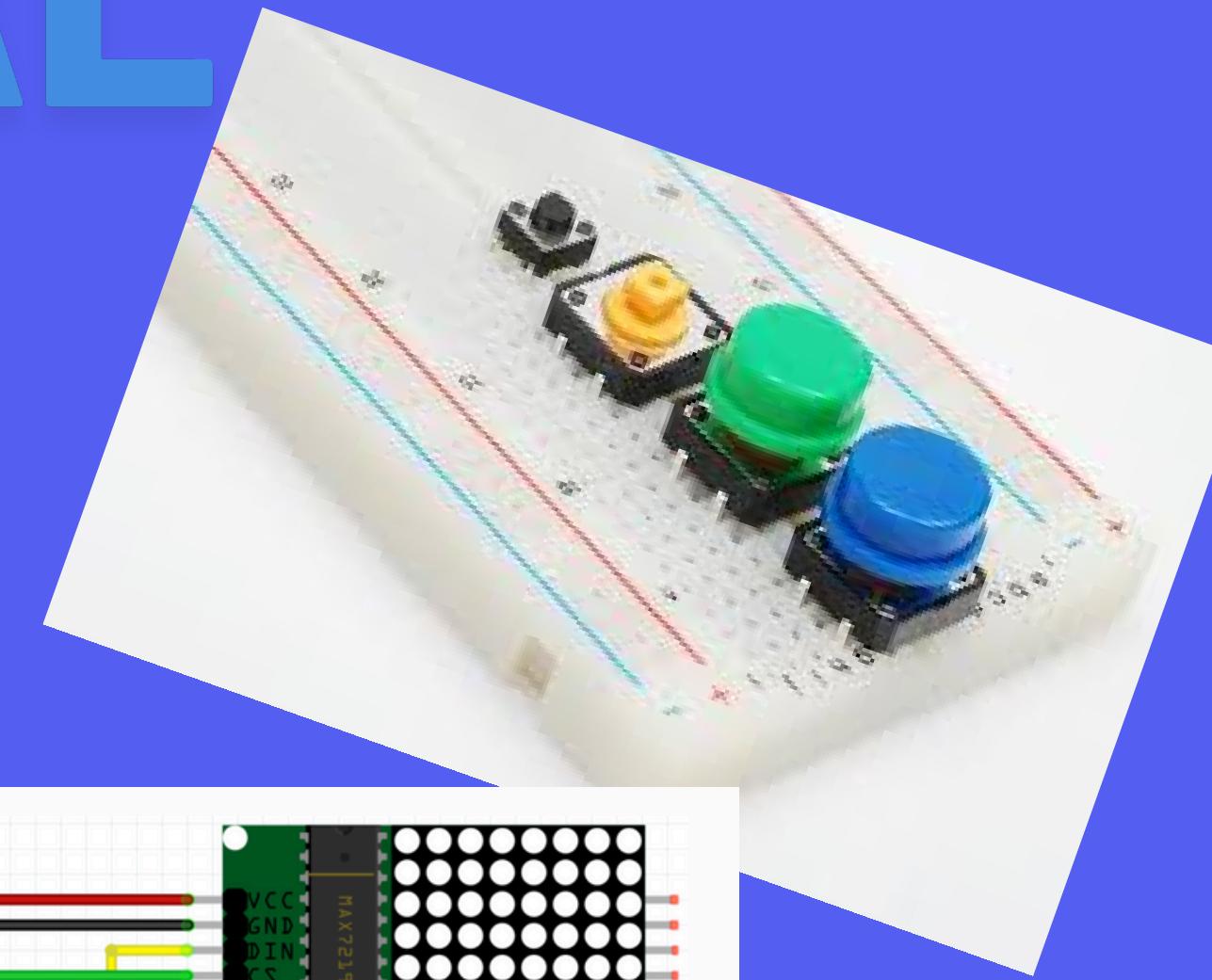
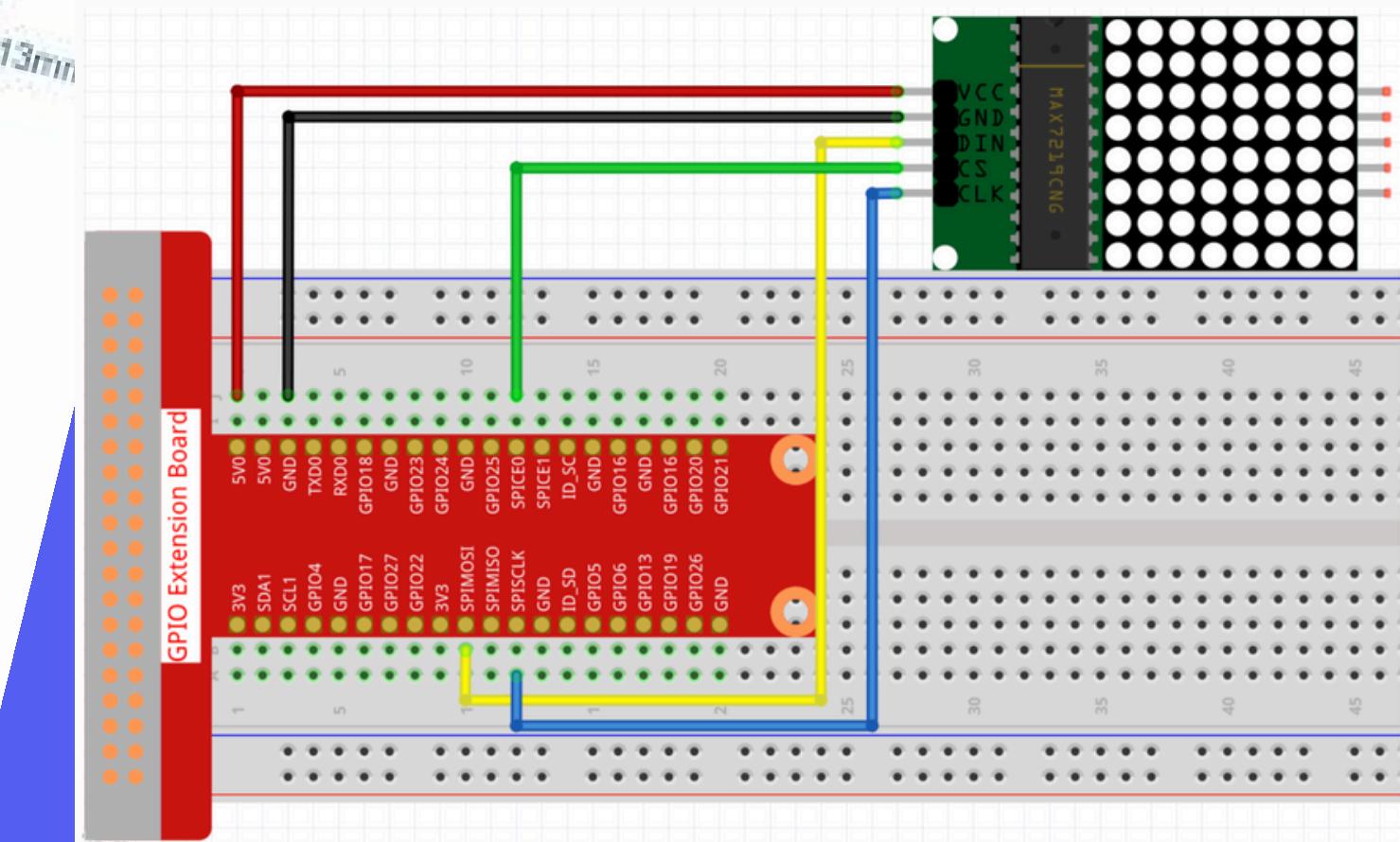
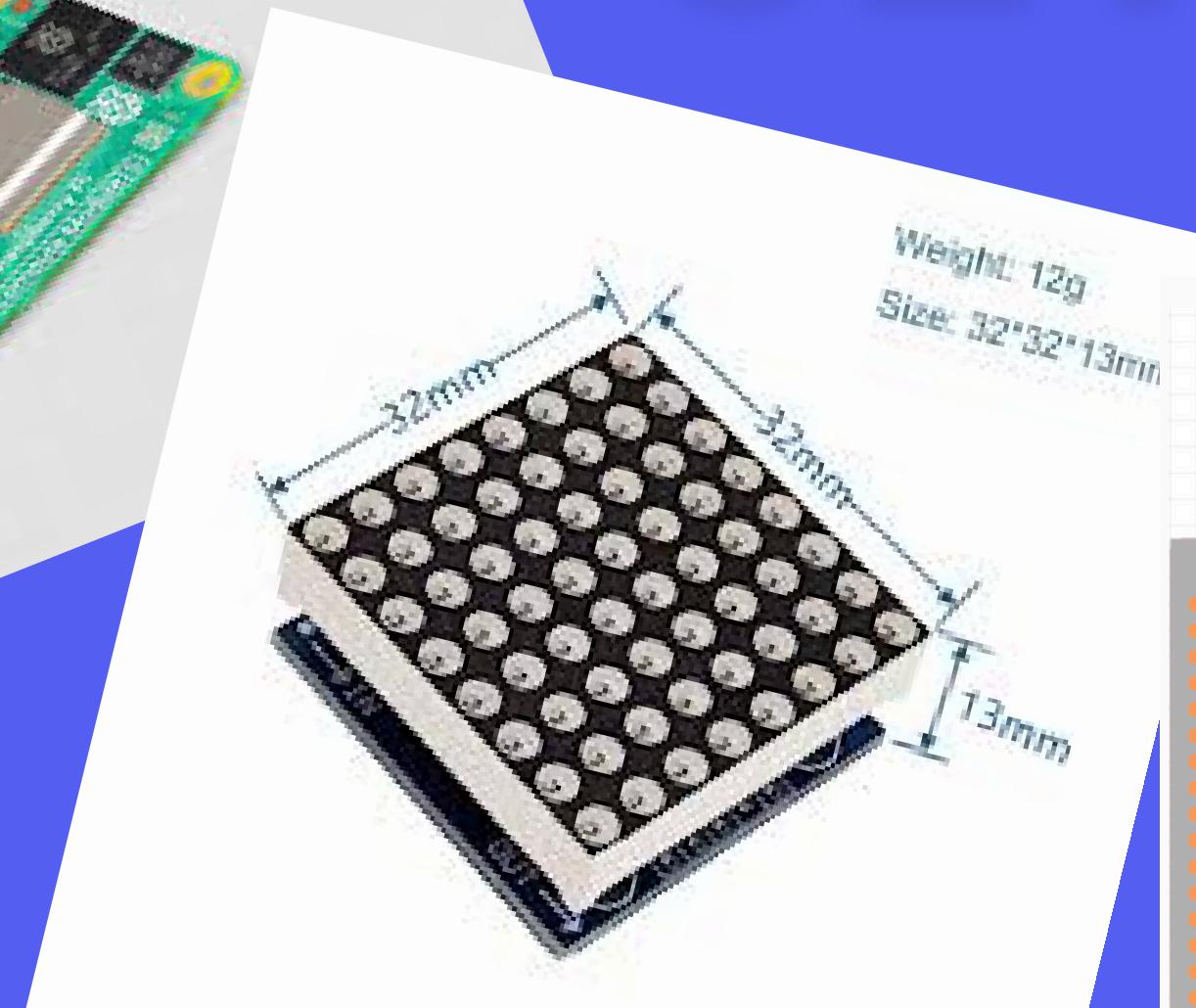
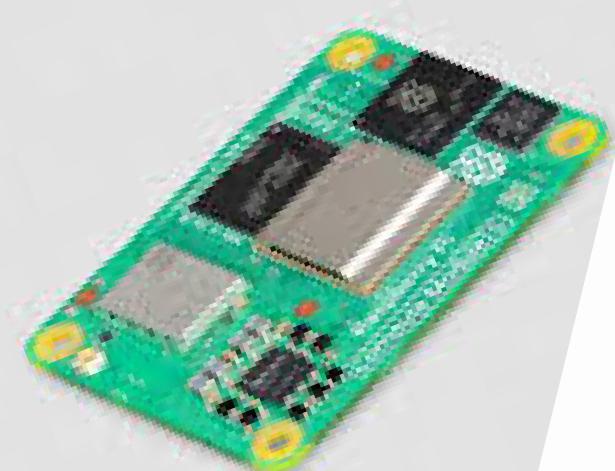
Python script pre-generates .wav files using SoX

SPI

LED matrix updates using SPI +
max7219_write()

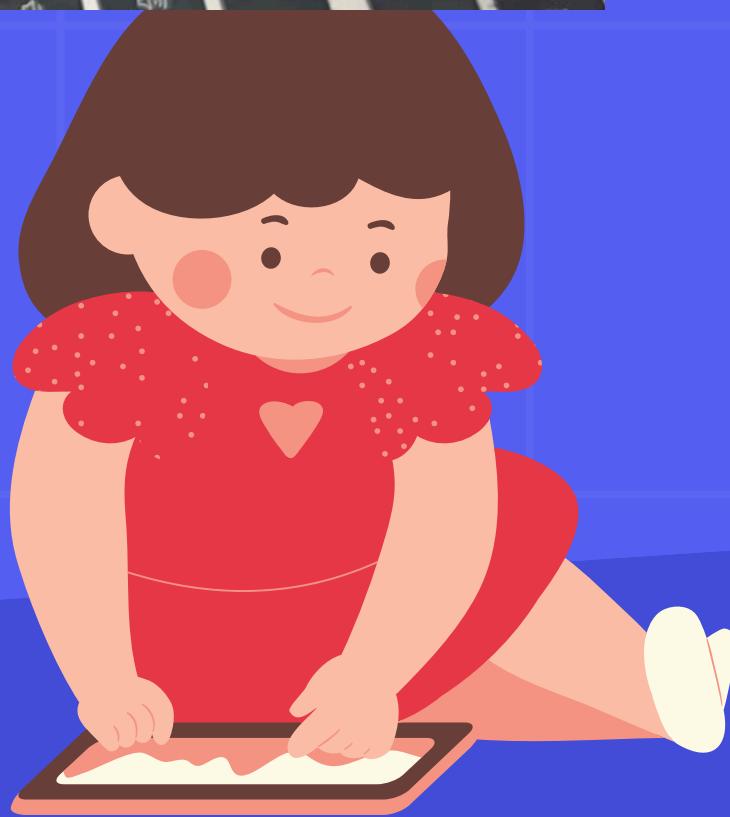
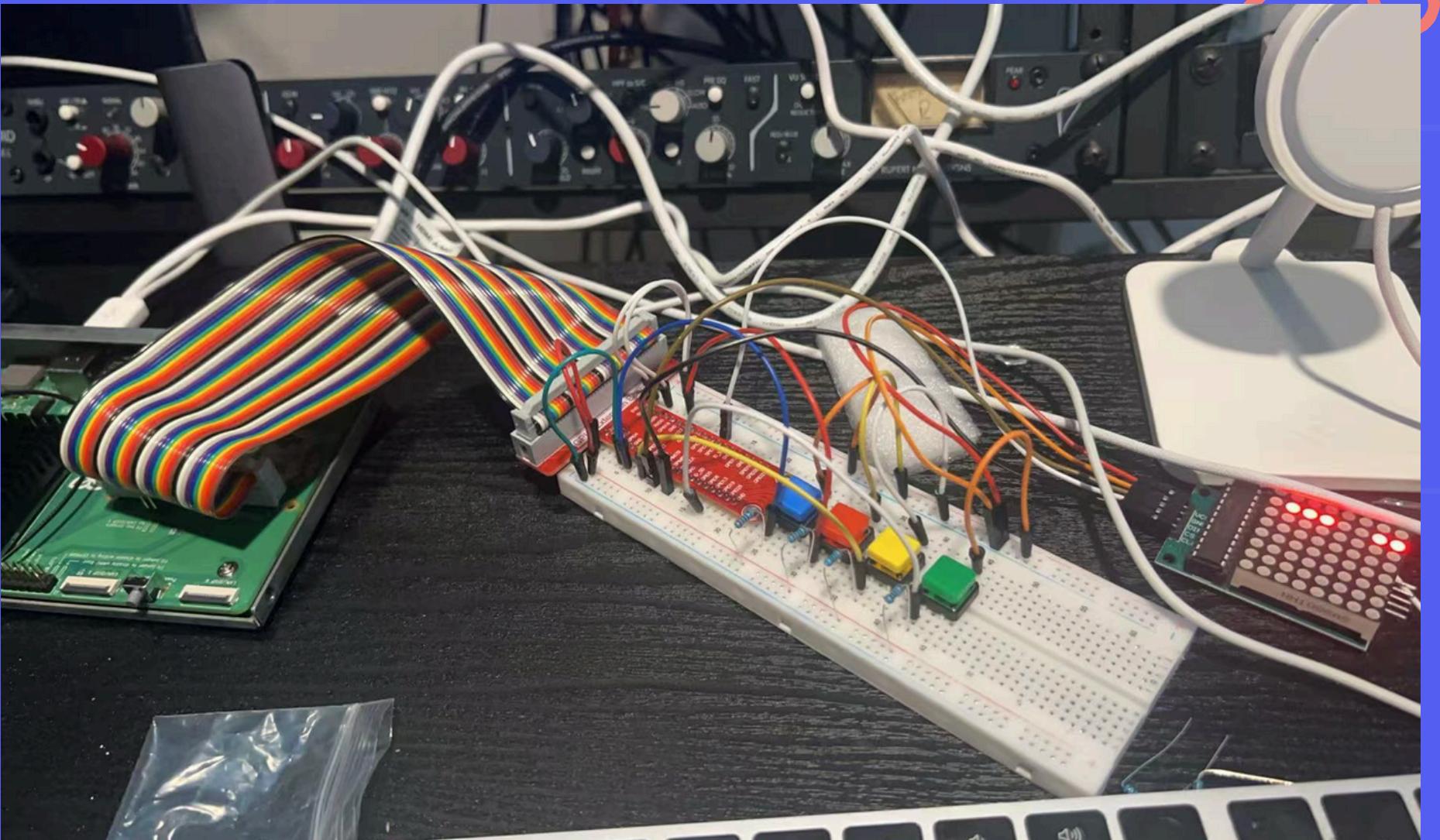
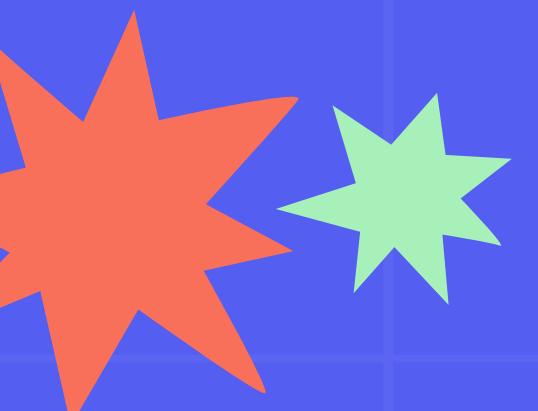


HARDWARE SETUP



Demo

Prototyping looks like this ----->

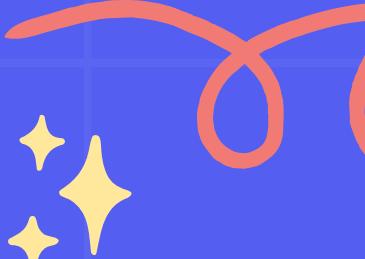


Audio Processing



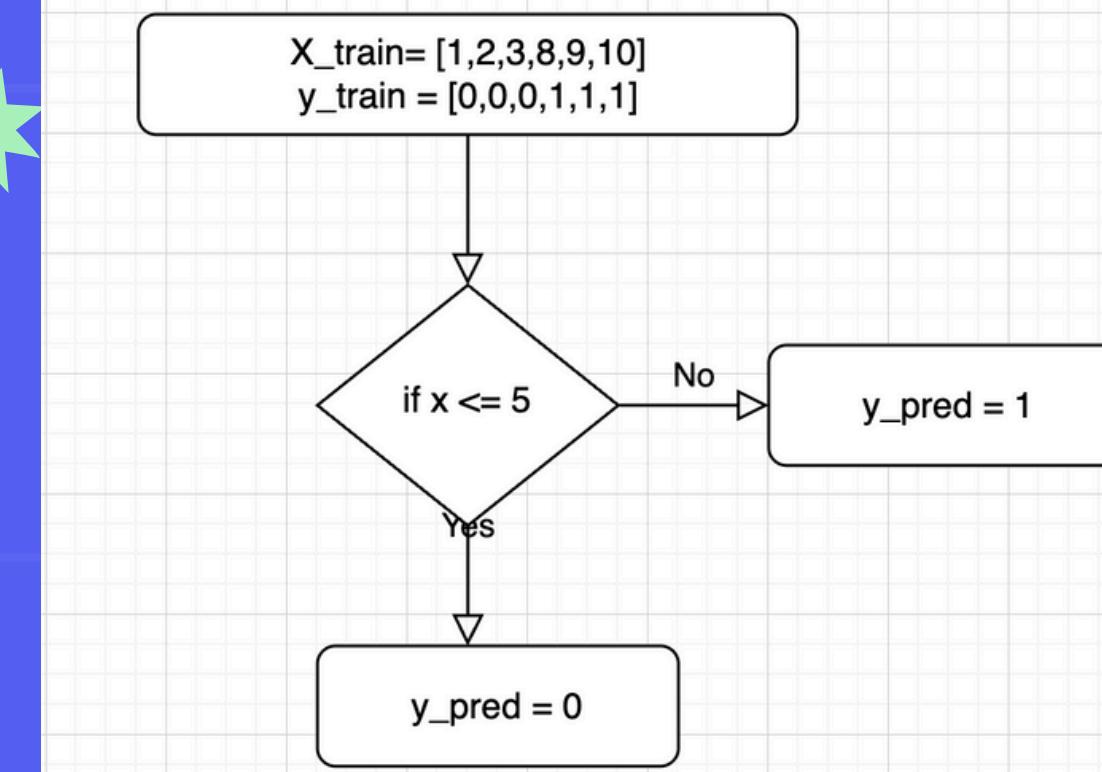
SoX

- Uses SoX for pitch and distortion
- Pre-renders 8 possible effect combinations:
 - Original, Lower, Pitch, Distortion, etc.



LOGIC FLOW

Selected using internal
'if' logic in C++



Why This Design?

Sometime when we lay on the bed or don't want to move, having a controller that can control the speaker remotely and have DSP function in it, wasn't a cool thing to show off with your friends?





SELF AWARENESS

Add real-time DSP

Add volume knob

Add spectrum display to LED matrix

Add web or mobile interface

End of Presentation

Thank you for
your time

Charlie SHI

