Real-Time Audio Effects Processor

Team 6: Antonio Hernandez Olivares, Cesar Chich-Saquic, Fedya Henrichs-Tarasenkov, Rafael Cervantes

Background

- **Problem:** Most audio effects require expensive hardware or complex software
- **Current Solutions:**
  - Professional rack units ($500-$2000),
  - Software plugins (need computer)
  - Overall limited consumer options with poor quality
- **Gap:** No current affordable, stand-alone, user-friendly solution
- **Motivation:**
  - Make audio processing more available for hobbyists
  - Educational tool for DSP concepts
  - Enable personalized music experience

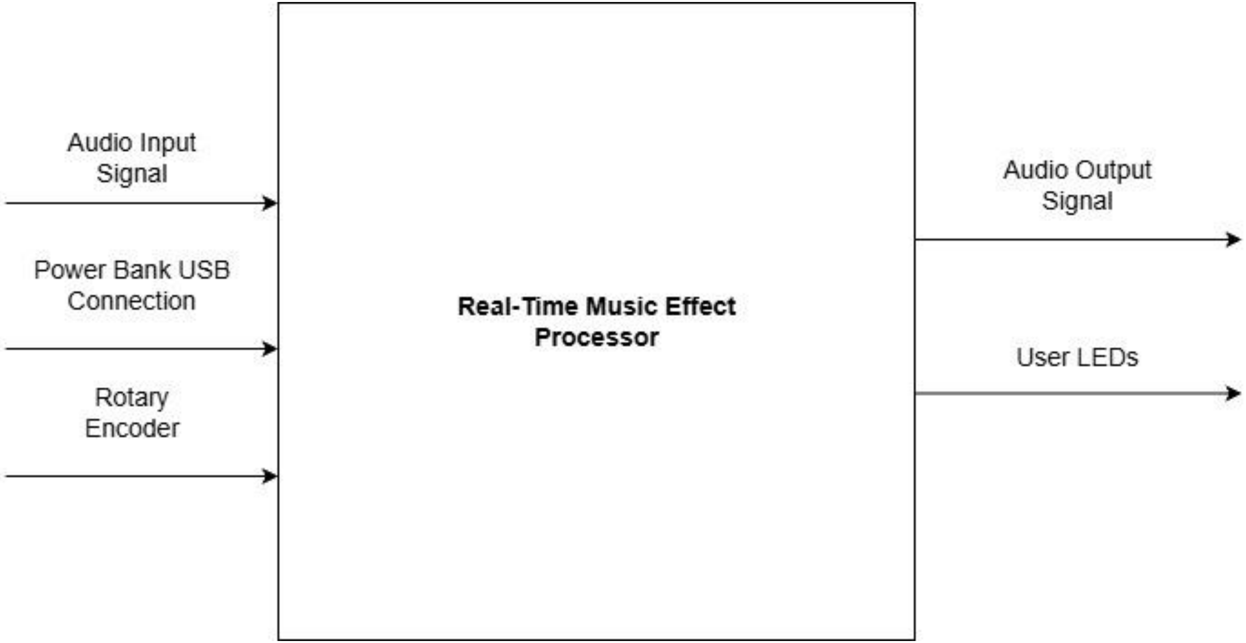
Concept of Operations

- **Setup:** Connect an audio source via 3.5 mm audio jack to the input of the effects box and connect the output of the box to headphones or a speaker via 3.5 mm audio jack (AUX cord)
- **Operation:**
  - Click encoder buttons to toggle chosen effects
  - Rotate encoders to adjust strength
  - LEDs indicate active effects
- **Use Cases:**
  - Personal music enhancement
  - Live instrument processing

Objectives & Requirements

- **Objectives:**
  - Design and build a system that will be able to convert an audio signal by adding various audio effects of our choosing
- **Requirements:**
  - Must: be portable, pass through audio signal, have a clean and intuitive user interface, have at least 1 audio effect, have a case
  - Should: have up to 3 audio effects
  - May: have a screen displaying audio audio spectrum

L0 Block Diagram



Audio Input Signal

Power Bank USB Connection

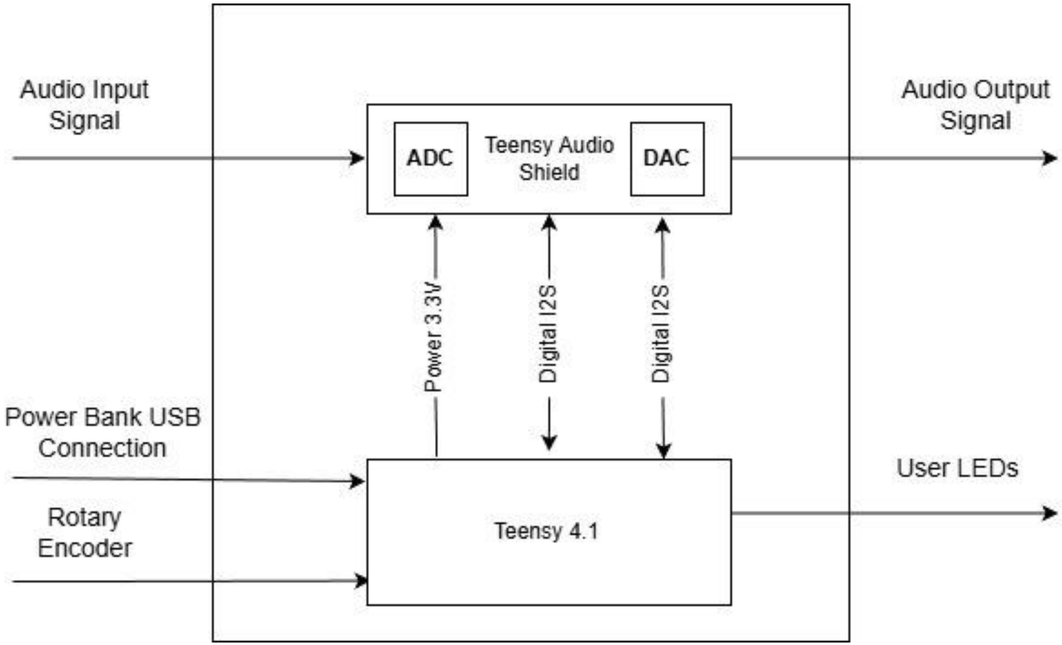
Rotary Encoder

Real-Time Music Effect Processor

Audio Output Signal

User LEDs

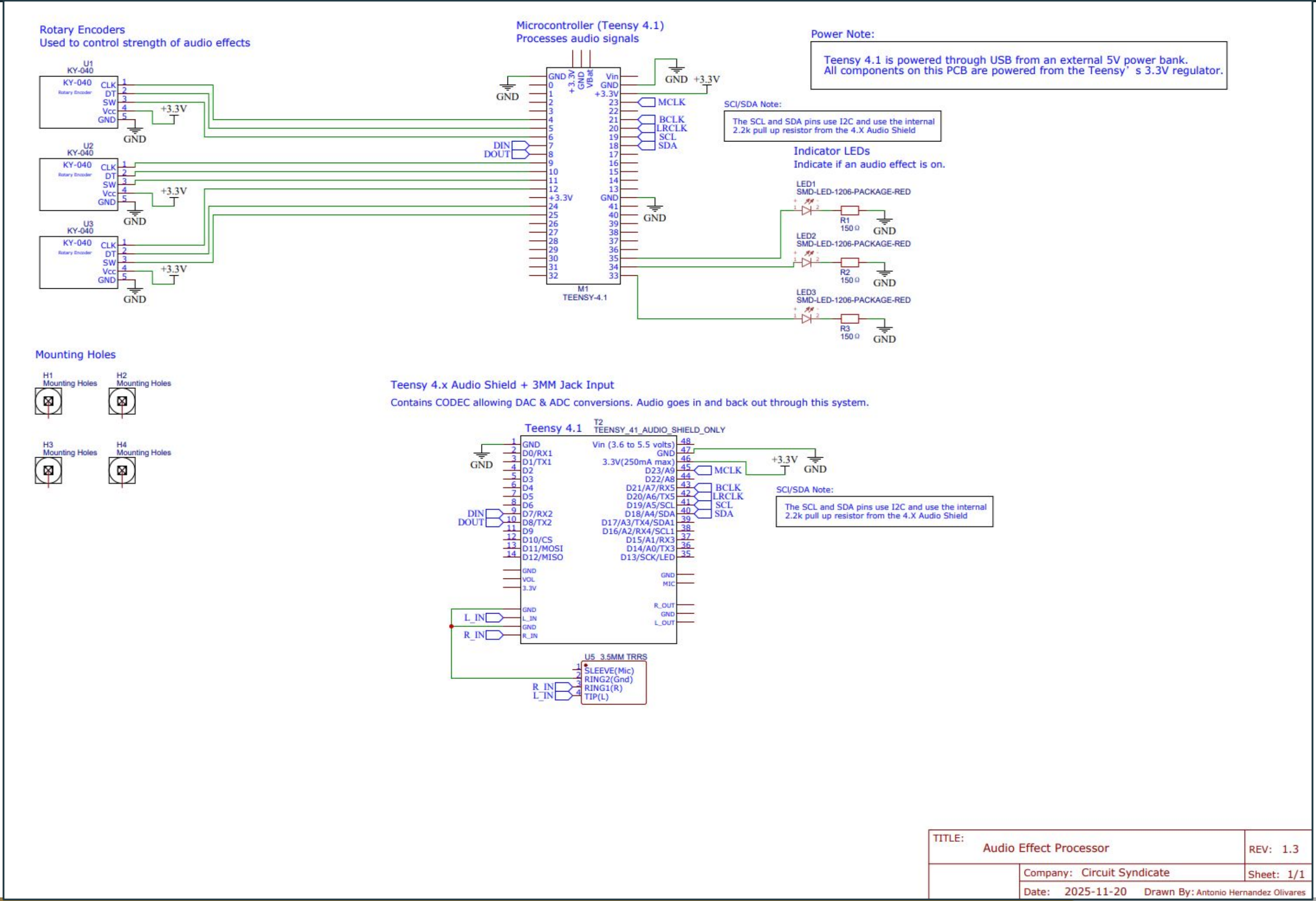
L1 Block Diagram



Hardware Implementation

Schematics, board layout, tools used

- Power distribution
  - We have a battery pack to provide 5V via USB
- Grounding planes
  - Flooded ground planes on both sides of PCB
- Audio Shield wiring, I2S and I2C
  - Audio shield and Teensy 4.1 wired together like they were stacked on top of each other. They need I2S for audio signal and I2C for digital communication between the audio shield and Teensy
  - The Audio shield uses I2S with the Teensy 4.1 to convert incoming audio with the build in ADC, then converts the output with its DAC.
- Encoders and LEDs for user menu
  - Encoders function as both our switch and knob. LEDs wired to signal which effects are on
- EasyEDA instead of KiCAD
  - EasyEDA allows for contribution on the website instead of having to share a file like on KiCAD



Rotary Encoders
Used to control strength of audio effects

Microcontroller (Teensy 4.1)
Processes audio signals

Power Note:

Teensy 4.1 is powered through USB from an external 5V power bank.
All components on this PCB are powered from the Teensy's 3.3V regulator.

SCI/SDA Note:

The SCL and SDA pins use I2C and use the internal 2.2k pull up resistor from the 4.X Audio Shield

Indicator LEDs
Indicate if an audio effect is on.

Mounting Holes

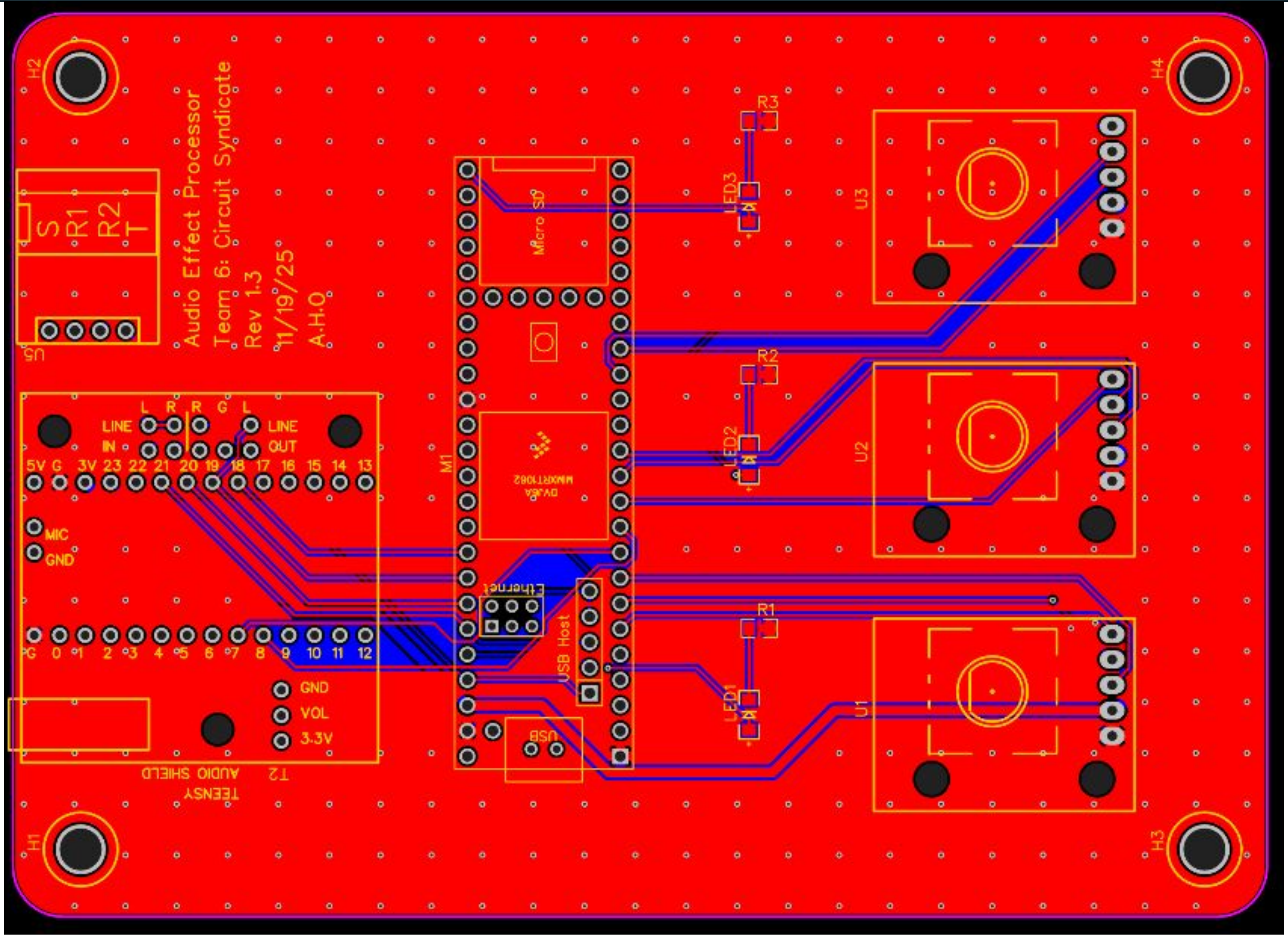
Teensy 4.x Audio Shield + 3MM Jack Input

Contains CODEC allowing DAC & ADC conversions. Audio goes in and back out through this system.

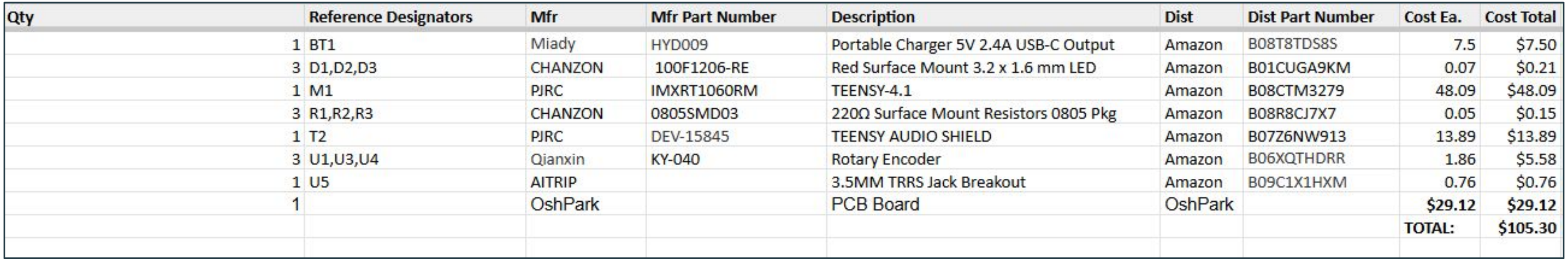
SCI/SDA Note:

The SCL and SDA pins use I2C and use the internal 2.2k pull up resistor from the 4.X Audio Shield

TITLE: Audio Effect Processor
REV: 1.3
Company: Circuit Syndicate
Sheet: 1/1
Date: 2025-11-20
Drawn By: Antonio Hernandez Olivares


Audio Effect Processor
Team 6: Circuit Syndicate
Rev 1.3
11/19/25
A.H.O
H1
H2
H3
H4
U5
S
R1
R2
T
M1
Micro SD
MIMXRT1062
DVJ6A
Ethernet
USB Host
USB
R1
R2
R3
LED1
LED2
LED3
U1
U2
U3
L R R G L
LINE IN
LINE OUT
5V G 3V 23 22 21 20 19 18 17 16 15 14 13
MIC
GND
G 0 1 2 3 4 5 6 7 8 9 10 11 12
GND
VOL
3.3V
T2
TEENSY AUDIO SHIELD

Bill of Materials

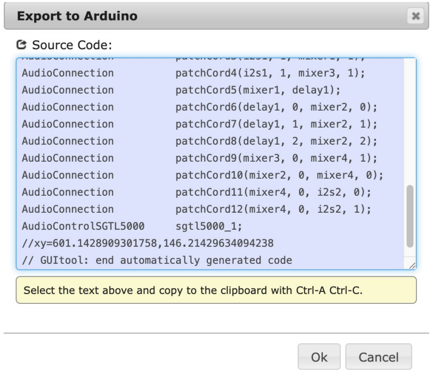


Qty	Reference Designators	Mfr	Mfr Part Number	Description	Dist	Dist Part Number	Cost Ea.	Cost Total
1	BT1	Miady	HYD009	Portable Charger 5V 2.4A USB-C Output	Amazon	B08T8TDS8S	7.5	$7.50
3	D1,D2,D3	CHANZON	100F1206-RE	Red Surface Mount 3.2 x 1.6 mm LED	Amazon	B01CUGA9KM	0.07	$0.21
1	M1	PJRC	IMXRT1060RM	TEENSY-4.1	Amazon	B08CTM3279	48.09	$48.09
3	R1,R2,R3	CHANZON	0805SMD03	220Ω Surface Mount Resistors 0805 Pkg	Amazon	B08R8CJ7X7	0.05	$0.15
1	T2	PJRC	DEV-15845	TEENSY AUDIO SHIELD	Amazon	B07Z6NW913	13.89	$13.89
3	U1,U3,U4	Qianxin	KY-040	Rotary Encoder	Amazon	B06XQTHDRR	1.86	$5.58
1	U5	AITRIP		3.5MM TRRS Jack Breakout	Amazon	B09C1X1HXM	0.76	$0.76
1		OshPark		PCB Board	OshPark		**$29.12**	**$29.12**
							TOTAL:	**$105.30**

Software Implementation

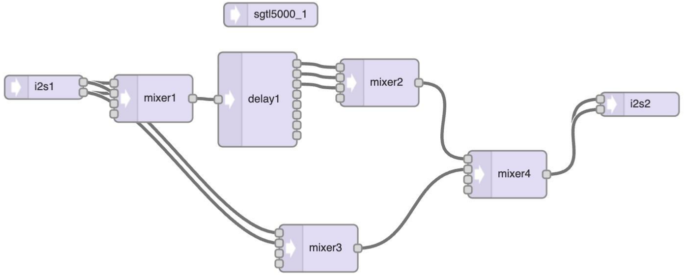
Key Libraries Used:

- Teensy Audio Library
  - Pre-built DSP components (no DSP algorithms written from scratch)
  - Visual design tool for drag-and-drop signal routing
  - Built in effects like delay, reverb, and bass boost
  - Handles 16-bit, 44.1 kHz streaming audio
- KY-040 Encoder Library
  - Interrupt-driven for responsive control and preventing false readings



```
AudioConnection          patchCord3(i2s1, 1, mixer1, 1);
AudioConnection          patchCord4(i2s1, 1, mixer3, 1);
AudioConnection          patchCord5(mixer1, delay1);
AudioConnection          patchCord6(delay1, 0, mixer2, 0);
AudioConnection          patchCord7(delay1, 1, mixer2, 1);
AudioConnection          patchCord8(delay1, 2, mixer2, 2);
AudioConnection          patchCord9(mixer3, 0, mixer4, 1);
AudioConnection          patchCord10(mixer2, 0, mixer4, 0);
AudioConnection          patchCord11(mixer4, 0, i2s2, 0);
AudioConnection          patchCord12(mixer4, 0, i2s2, 1);
AudioControlSGTL5000     sgtl5000_1;
//xy=601.1428909301758,146.21429634094238
// GUItool: end automatically generated code
```

Select the text above and copy to the clipboard with Ctrl-A Ctrl-C.



Software Implementation Cont.

Main Loop Structure:

- For each of the three encoders, the program:
    - Checks if encoder value changed (parameter adjustment)
    - Updates effect if it’s currently active
    - Monitors button press (enable/disable effect)
    - Updates LED status

Effect Control:

Effect	Parameter	Encoder Range
Bass Boost	Level	0.85-1.0
Reverb	Room Size	0.8-1.0
Delay	Time (base)	300-800ms

Software Implementation Cont.

Key Features:

- Interrupt-driven encoders: Responsive, no missed turns or false readings
- Button debouncing: 200ms lockout prevents accidental toggles
- Parameter reset: Effects start at default values when enabled
- Surround sound: Always-on 3D audio enhancement

IP and Prior Work

- **GNU General Public License v3.0**
  - Ensures our work remains open source
  - Included with all source code, schematics and documentation
- **Third Party IP used:**

1. Teensy Audio Library (MIT License): Audio processing framework
2. Arduino/Teensyduino (LGPL): Development Platform
3. Teensyduino Example Code (LGPL): Used pre-made example code as our base for hardware testing and learning how to use different functions from the library
4. KY-040 Library: Used for interrupt control

Testing

- **Methodology:**
  - Unit Testing: Individual effect algorithms
  - Integration Testing: Full audio pipeline
  - User Testing: Interface usability (encoders, LEDs)
- **Results:**
  - Each effect worked as expected, integration through changing audio connection setup within code, encoders worked with both click and rotate functionality

Results

- **What Worked:**
  - Real-time performance: no audio dropouts
  - Audio quality
- **Challenges:**
  - Audio Noise: Proper grounding and power filtering caps
  - Dynamic audio routing: Mixer-based transitions
  - Cost about $100

Contributions

Antonio Hernandez Olivares

- Circuit Schematic Design
- PCB Layout Design
- Software Development

Cesar Chich-Saquic

- Documentation
- PCB Testing
- PCB Assembly

Fedya Henrichs-Tarasenkov

- Software Testing
- Breadboard Testing
- PCB Testing
- Documentation

Rafael Cervantes

- Documentation
- PCB Layout Design
- Circuit Schematic Design

Lessons Learned

Team Lessons Learned

- PCB Design
- PCB Testing
- Team Collaboration
- How to use github
- Practice schematic design

Individual Lessons Learned

- Fedya:
  - Github repo construction/organization
  - Testing- unit, bottom-up
  - Teensy audio library
- Rafael:
  - EasyEDA
- Cesar:
  - PCB testing and assembly
  - Team Collaboration and organization
- Antonio:
  - Buy Chinese. No OSH Park.
  - Ground planes with via stitching

Demo and Questions

https://github.com/FedyaHenrichs-Tarasenkov/music-effects-practicum/tree/main