



ECE 477 FINAL REVIEW: TEAM #12

OUTLINE

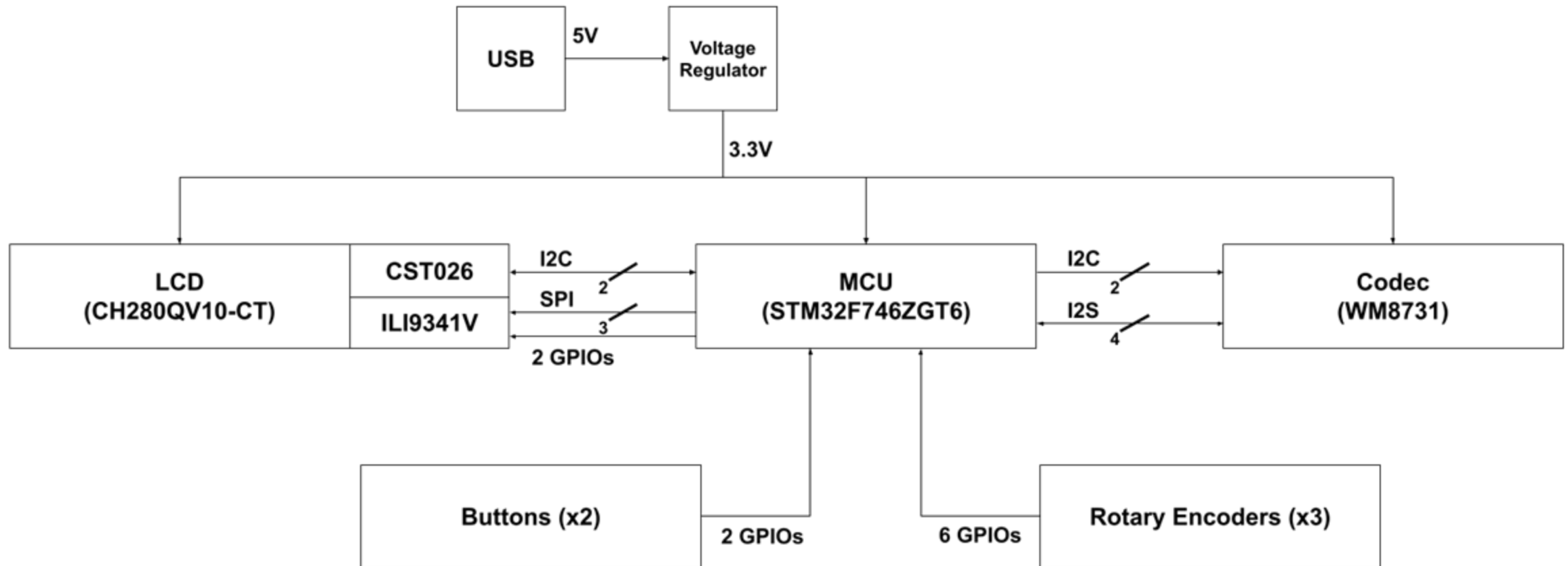
- Project Overview
- Block Diagram
- Design Challenges
- Individual Contributions
- Project Video Demonstration
- Questions

PROJECT OVERVIEW

Our project is an analog input/output audio interface capable of applying equalization, distortion, and delay to the input signal. Its effects can be selected via onboard buttons or touchscreen input, and their parameters can be varied using three rotary encoders.



BLOCK DIAGRAM



DESIGN CHALLENGES

- *USB Audio Protocol*
 - *Unable to set up the device as a USB audio input using ST's USB library.*
 - *TinyUSB library difficulties with getting a clean signal.*
- *DSP Optimization*
 - *Unable to run all the effects simultaneously.*
 - *Clock speed too low.*
 - *Compiler optimization.*

MicNode_4_Ch (VID=0xCAFE PID=0x4010)

Interface Association Descriptor:

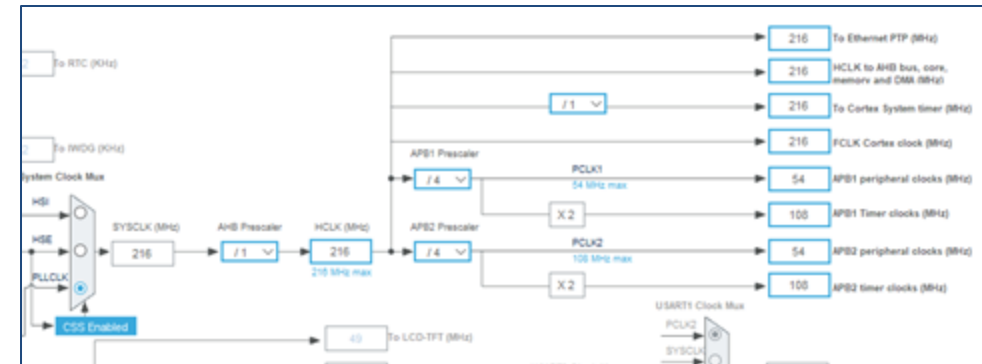
0x08	bLength	
0x08	bDescriptorType	
0x00	bFirstInterface	
0x02	bInterfaceCount	
0x01	bFunctionClass	(Audio Device Class)
0x00	bFunctionSubClass	
0x20	bFunctionProtocol	(Audio Protocol IP version 2.00)
0x00	iFunction	

Interface Descriptor:

0x09	bLength	
0x04	bDescriptorType	
0x00	bInterfaceNumber	
0x00	bAlternateSetting	
0x00	bNumEndpoints	
0x01	bInterfaceClass	(Audio Device Class)
0x01	bInterfaceSubClass	(Audio Control Interface)
0x20	bInterfaceProtocol	(Audio Protocol IP version 2.00)
0x00	iInterface	

AC Interface Header Descriptor:

0x09	bLength	
0x24	bDescriptorType	
0x01	bDescriptorSubtype	
0x0200	bcdADC	
0x03	bCategory	(MICROPHONE)
0x0048	wTotalLength	(72 bytes)
0x00	bmControls	



DESIGN CHALLENGES

- Codec timer bug
 - Audio popping when prototyping codec
 - Occasionally strangely high samples being sent by the codec
 - Turned out to be caused by a timer mismatch between the codec and microcontroller
 - Fixed by putting the microcontroller into slave mode and the codec into master mode

Free protocol modes (7/13) 13

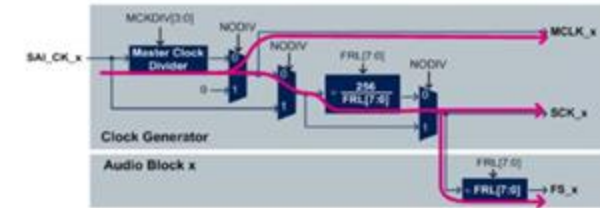
Sampling Rate Adjustment, when MCLK is generated:

$$f_{MCLK} = \frac{f_{SAI_CK}}{2 \times MCKDIV} \quad (1)$$

$$f_{FS} = \frac{f_{MCLK}}{256}$$

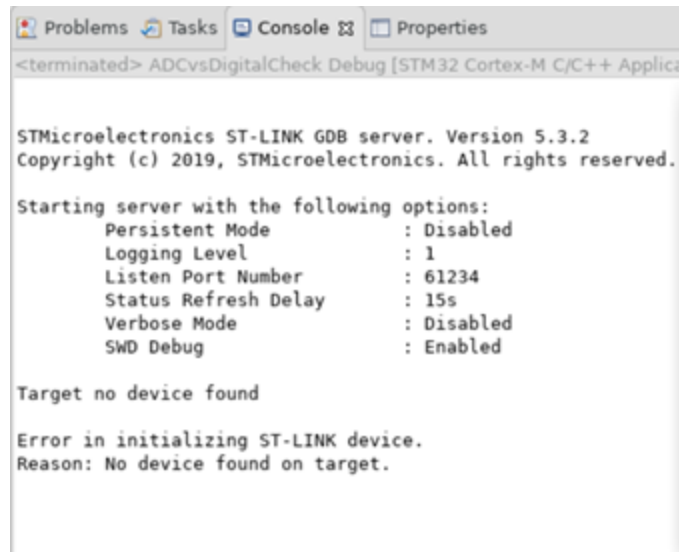
$$f_{SCK} = f_{FS} \times (FRL + 1)$$

$FRL+1 = 8, 16, 32, 64, 128 \text{ or } 256$



DESIGN CHALLENGES

- *Problem: Unable to load code*
- *Debugger (ST-Link Device not found)*
- *ST-Link Utility*
- *Clearing Flash Memory*
- *Error: Initializing incorrect pins*



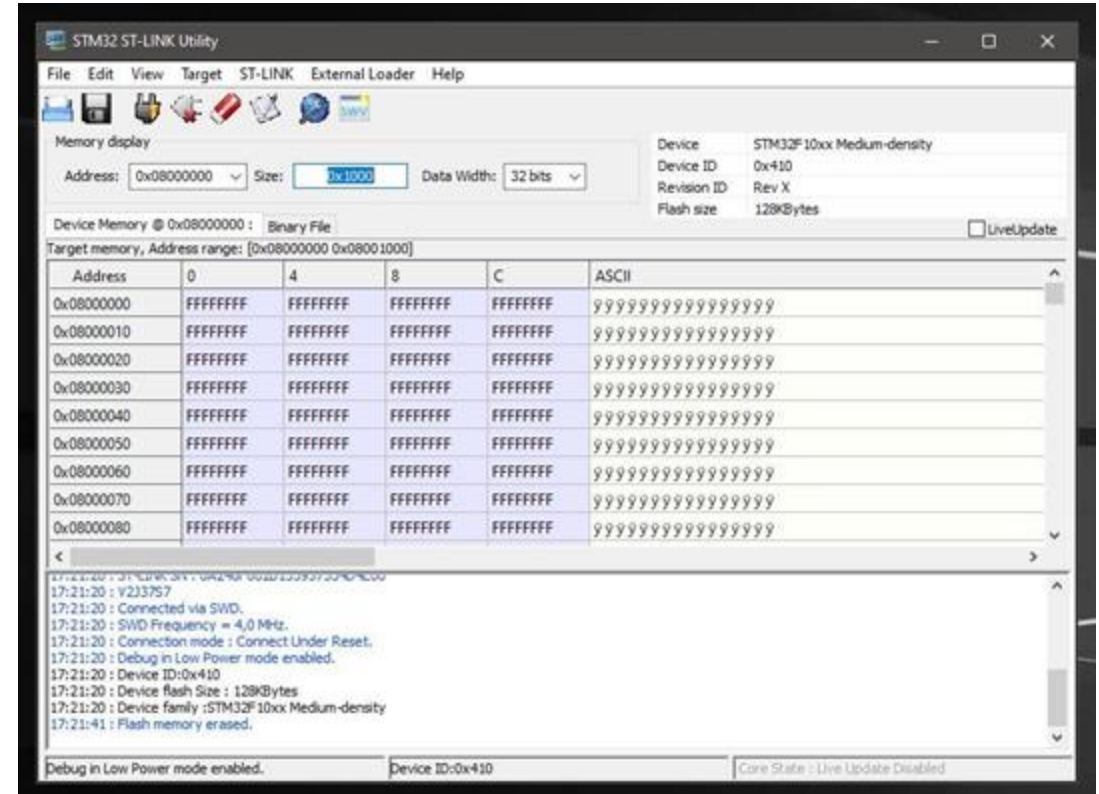
```
<terminated> ADCvsDigitalCheck Debug [STM32 Cortex-M C/C++ Applica

STMicroelectronics ST-LINK GDB server. Version 5.3.2
Copyright (c) 2019, STMicroelectronics. All rights reserved.

Starting server with the following options:
  Persistent Mode      : Disabled
  Logging Level       : 1
  Listen Port Number  : 61234
  Status Refresh Delay: 15s
  Verbose Mode        : Disabled
  SWD Debug           : Enabled

Target no device found

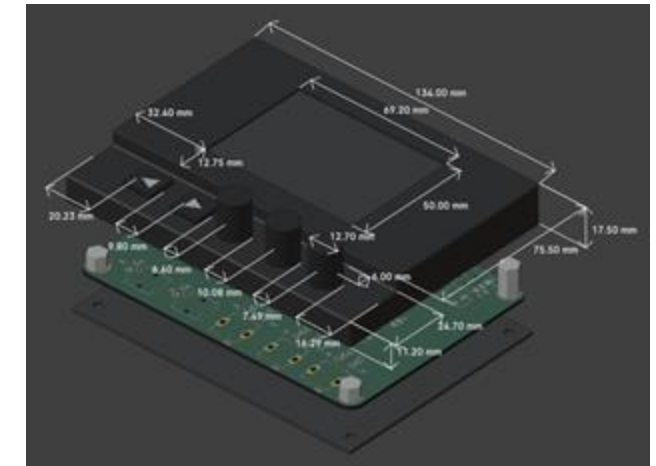
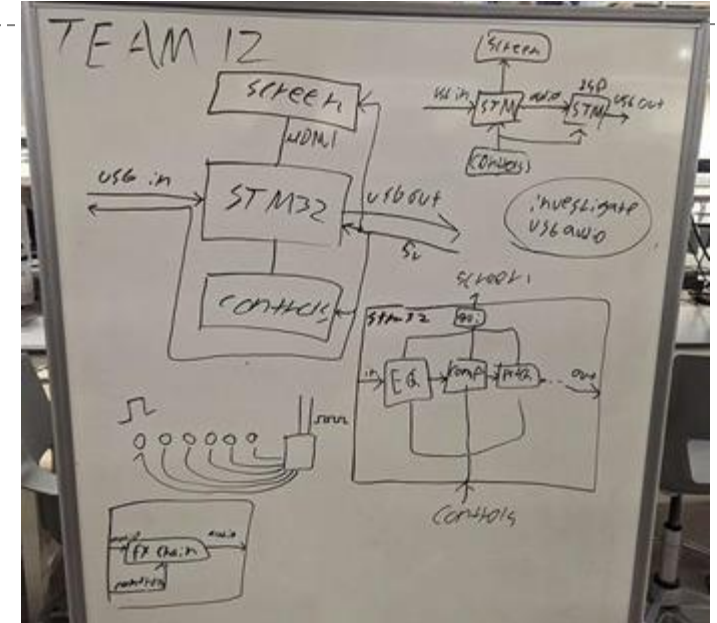
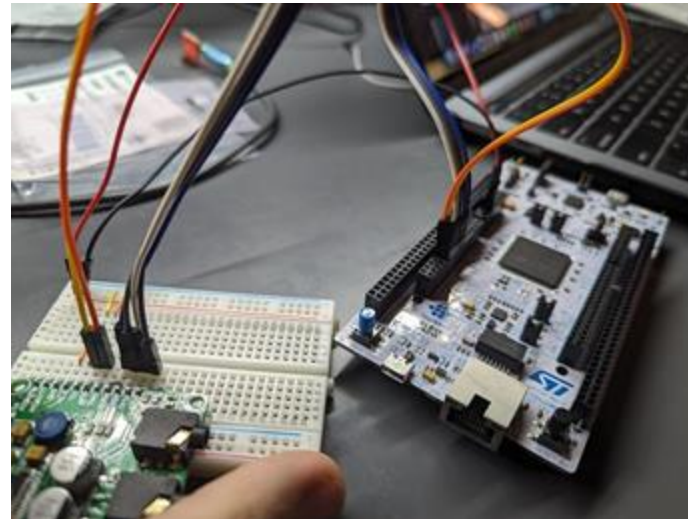
Error in initializing ST-LINK device.
Reason: No device found on target.
```



INDIVIDUAL CONTRIBUTIONS

Liam Roach

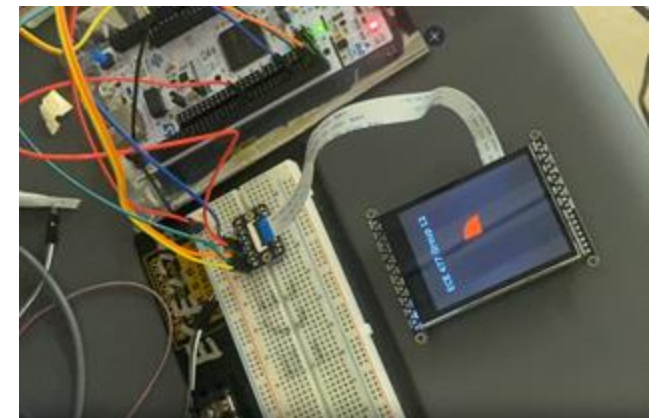
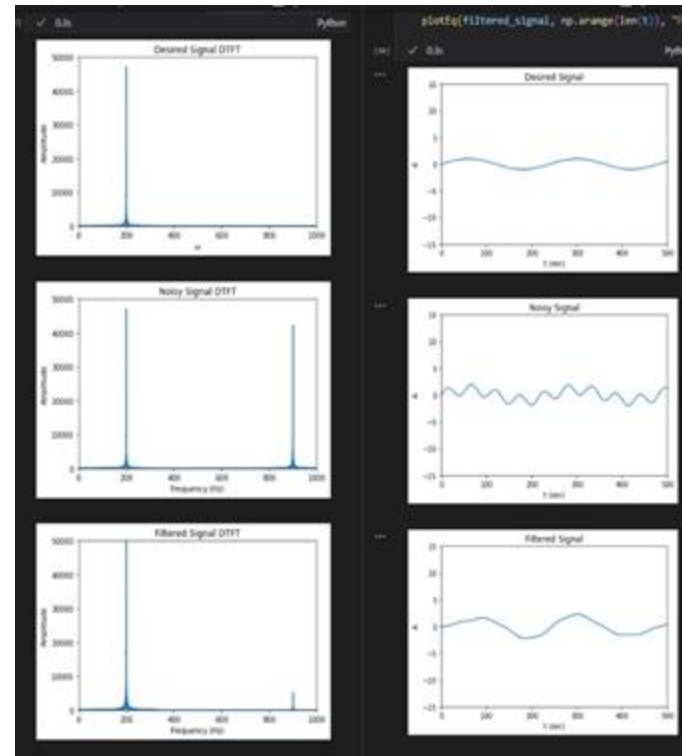
- Guided initial project design
- Codec interfacing and debugging
- Part selection and procurement
- Assisted with PCB design revisions
- Packaging design and assembly
- PCB soldering



INDIVIDUAL CONTRIBUTIONS

Jakub Kowalski

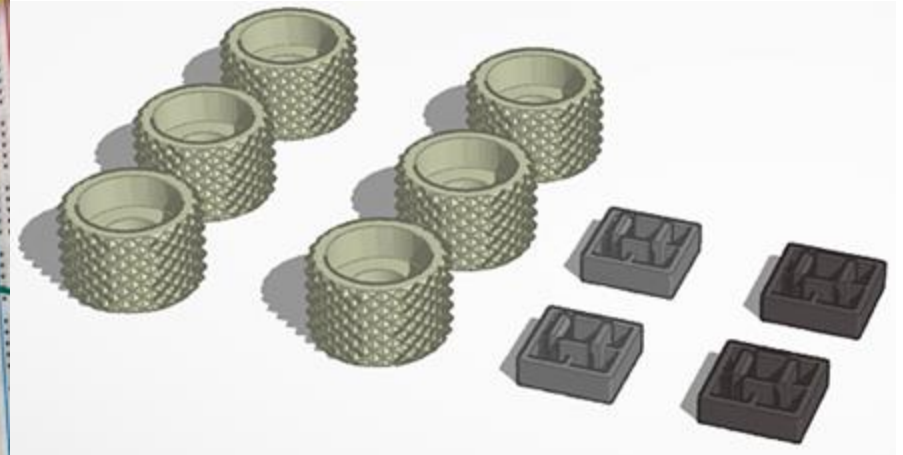
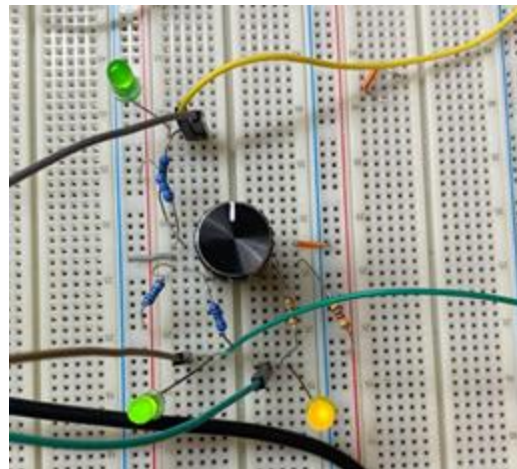
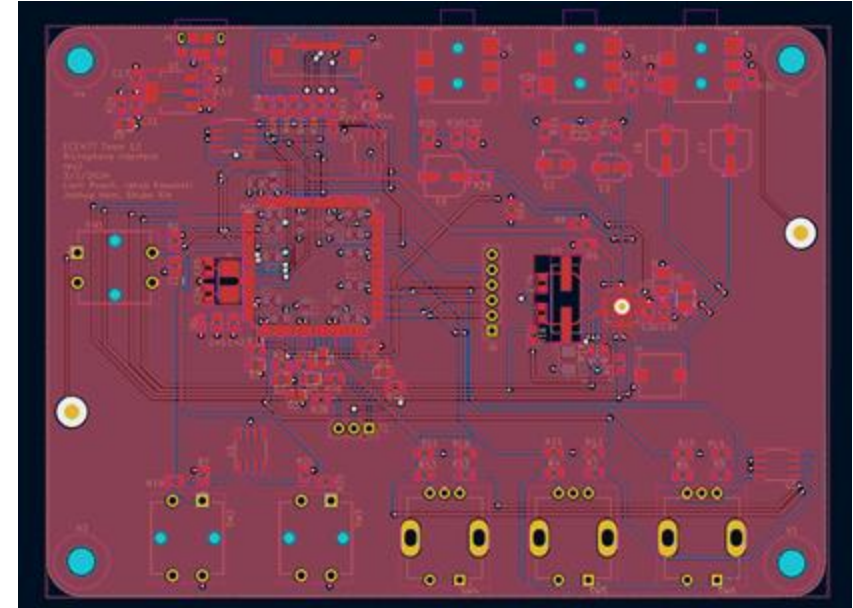
- Initial Concept
- USB Protocol Prototyping
- LCD Touchscreen Driver Implementation
- TouchGFX Library Implementation / UI Design
- Component Integration
- DSP Implementation & Optimization
- User Testing



INDIVIDUAL CONTRIBUTIONS

Joshua Hom

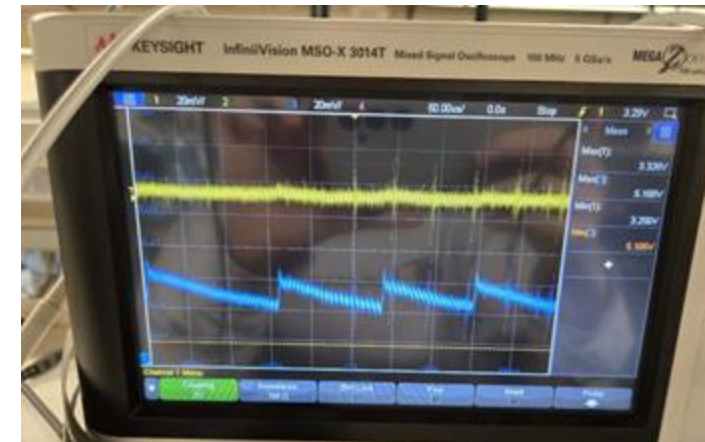
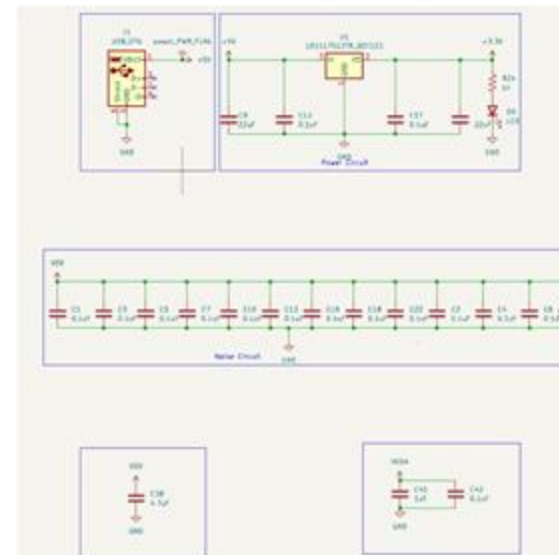
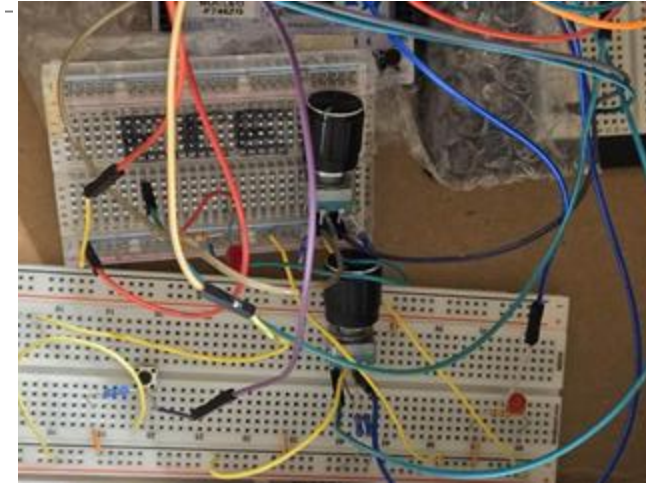
- Prototyped Rotary Encoders
- Schematic / Integration
- PCB Layout
- Rotary Encoder and Button Caps
- PCB Assembly Assist
- DSP implementation research
- Software Review
- User Testing



INDIVIDUAL CONTRIBUTIONS

Shubo Xie

- Power Electronics
- Schematic Diagram
- PCB Design Rule Fixing
- LCD Library Implementation
- Control System Integration
- Microphone Implementation
- USB Interface



PROJECT DEMONSTRATION

- PSDR #1 (Hardware): An ability to send and receive an audio signal to and from a codec and a microcontroller via I2S.
- PSDR #2 (Software): An ability to apply five band EQ, distortion, and delay effects via DSP on an input audio.
- PSDR #3 (Hardware): An ability to control DSP parameters using input to a microcontroller from a set of rotary encoders and buttons.
- PSDR #4 (Hardware): An ability for the microcontroller to interface with an LCD display via SPI.
- PSDR #5 (Software): An ability to provide a GUI to display DSP parameters and corresponding audio effects.

Stretch PSDR:

- PSDR #6 (hardware): An ability to control parameters and interact with the user interface via capacitive touch screen.

PROJECT DEMONSTRATION VIDEO

<https://youtu.be/lcTVyilBquE>

Questions?