

Cyren

A Music Effects Device

sddec19-15 email: jgshaver@iastate.edu

Project Overview

- ❑ Device is designed to replace or supplement a guitar pedal setup with similar user interface (stomp switches, floor pedal, etc)
- ❑ Take input audio from guitar/bass
- ❑ Features four programmable and chainable “effect pedals” that will be simulated over software.
- ❑ User can interact with knobs and push buttons to select or control effect levels and parameters
- ❑ Output manipulated audio signal to connected speakers/amp

Design Pivot

Microcontroller: RockPro64 -> Raspberry Pi

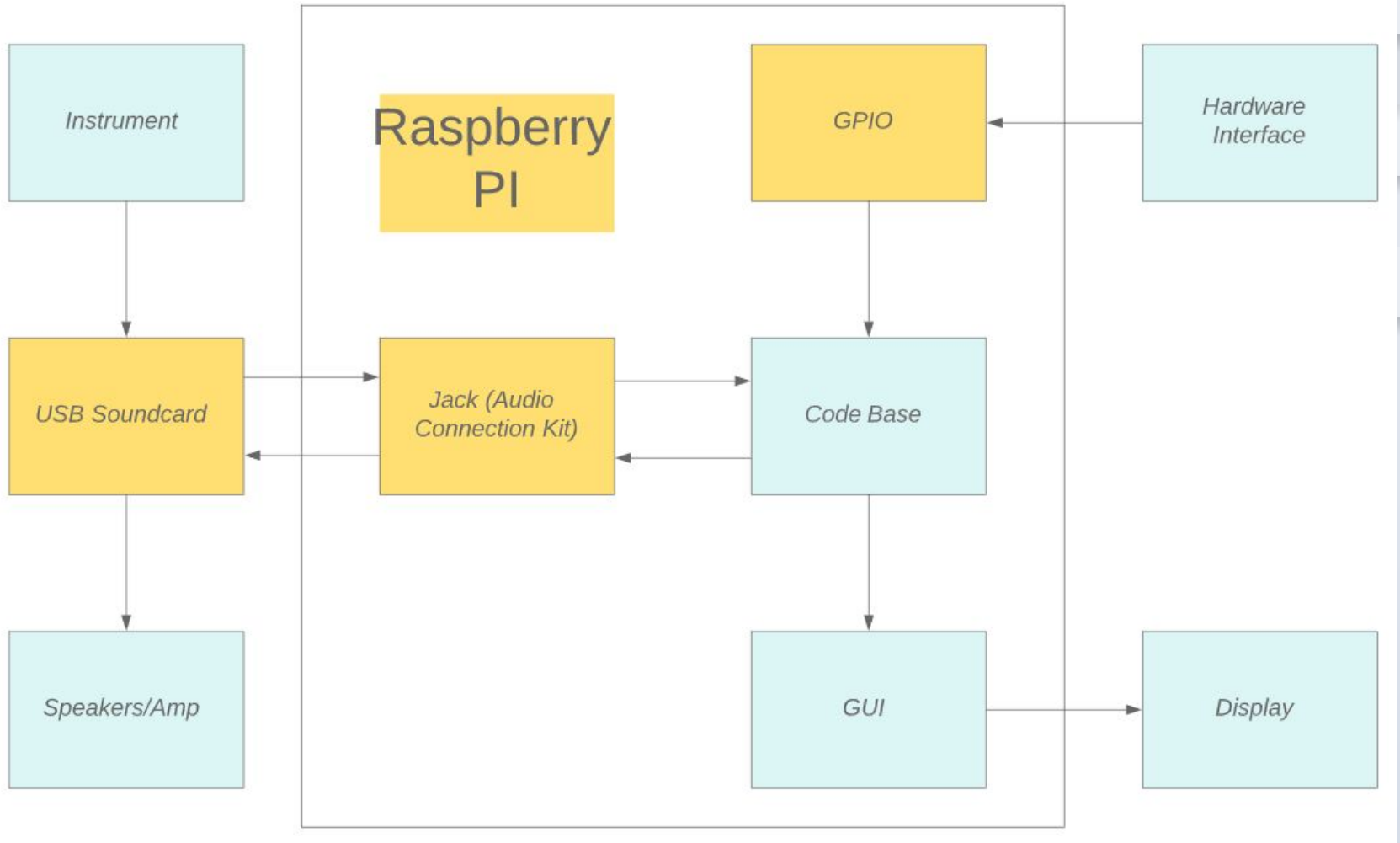
Why?

- ❑ Poor documentation
- ❑ Incapable audio capabilities
- ❑ Less support

This affected:

- ❑ GPIO interface
- ❑ Audio signal transfer - JACK
- ❑ Hardware change - USB sound card





Hardware: Rotary Encoders - Debouncing



General Description

- Push button signal flips between on/off state when pressed.

Possible Solutions

- Hardware implementation with 555 timer one off circuit.
- Software implementation with state machine.

Current Progress

- Decided on software implementation with 2-bit state machine - still having debouncing - will attempt with increased bit depth (4-bit).
- Ordered 555 timers for backup - would require rebuilding the rotary encoder circuits.

Software: Audio Routing Utility (JACK)

General Description

- ❑ JACK is the audio routing utility we are using to collect input from audio ports. It allows us to route the audio through our client application and out to our speakers.
- ❑ We are experiencing issues when routing the audio through our filters.

Possible Solutions

- ❑ Currently we are copying a set of frames at a time from the buffer, altering them using our filters, and then copying them to the output port. We believe implementing a circular buffer will fix the issue we are having.

Current Progress

- ❑ As of now you can hear the filters being applied, but it sounds like they are being applied to a “background” like noise, rather than the actual audio input.

Software: Audio Effects

General Description

- ❑ Audio effects for our device are digital and are written in C -algorithm source: *Audio Effects: Implementation, Theory and Application* - Joshua Reiss, Andrew McPherson
- ❑ We have not had satisfactory results transitioning these effects from our test environment(Visual Studio) to our client application.

Possible Solutions

- ❑ More accessibility for testing
- ❑ Continued work on transitioning code to client application

Current Progress

- ❑ Some of the remaining effects (Time-Delay) are best implemented with a circular buffer and this has not been completed with our JACK client code yet.

Ongoing Challenge - GUI Development

Switching GUI libraries

- ❑ Original library was GTK - intended for desktop environments and is structured so the GUI acts as the top-level “runner” for the rest of the code.
- ❑ Switched to LittlevGL - designed for embedded software. LVGL can run on a separate thread from the main program, essentially running “on top” of the rest of the code.

Integrating GUI into existing code base

- ❑ GUI is built and has been tested in separate environment.
- ❑ Hard to test GUI functionality when core functionality is still in development.

Upcoming Challenge - Enclosure

- ❑ Final design will be fabricated this week
- ❑ Material will be wood
- ❑ Components will be mounted in set holes
- ❑ LCD display will set in the cutout
- ❑ The Pi GPIO will be placed flat on the base, evenly accessible to each component

