ECE 445

Fall 2024

Project Proposal

Adaptive Response Digital Guitar Pedal

Arya Nagabhyru
Jack Vulich
Will Coombs

Introduction

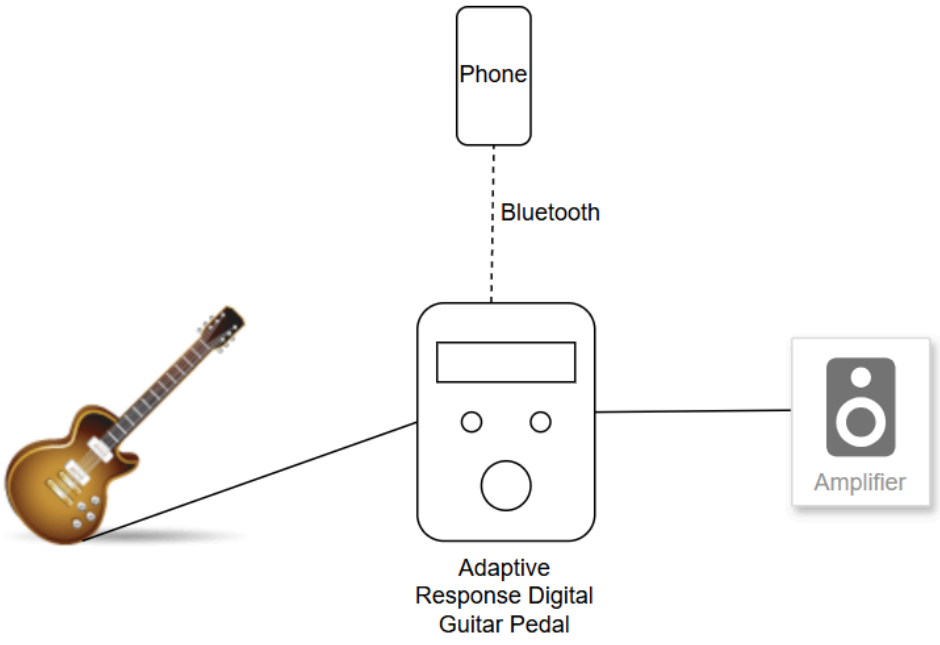
Problem

The original method of guitar amplification used vacuum tubes. While tube amplifiers sound great and are very responsive to the dynamics of the guitar signal (i.e. a quiet note has very little distortion and a loud one is quite distorted), they are heavy and expensive. Modern day solid state amplifiers are not very responsive to the dynamics of the signal, giving the player less flexibility when playing.

Solution

We propose an adaptive response digital guitar pedal. This pedal would respond to the volume of the note played, and adjust the amount of effect accordingly. This would be a cheap, effective way to give the guitar player the responsiveness of a tube amplifier, without the cost of it. This same principle would be applied to other effects as well, allowing the player to adjust the amount of reverb, chorus, delay, or fuzz just by how lightly they pluck a string. This opens up a world of possibilities for the player, allowing them to adjust their sound on the fly. This effects pedal is mainly for the use of the player, to allow them to change their sound without making any adjustments to their amplifier or effects board. It is not something for the listener to notice. There will also be a compression switch, allowing the user to adjust effect amount by playing a note at a different volume, but keeping the output volume at the same level.

Visual Aid



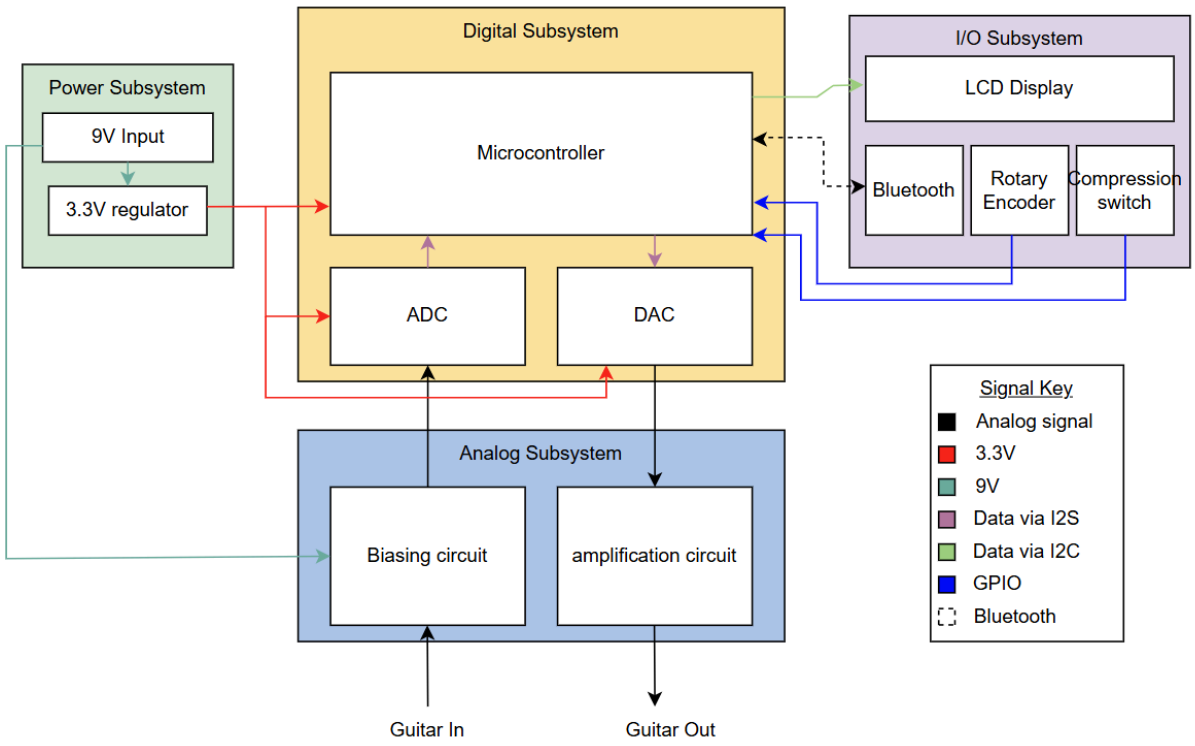
High Level Requirements

Responsive: our unit will need to respond to the volume of the incoming signal to adjust the amount of effect being used

Power: The pedal must be designed to run from a 9V wall plug (max 2A).
Low Latency: our design must process the guitar signal fast enough so that the player does not notice any delay in the signal. Ideally, the delay will be less than 12ms.

Design

Block Diagram



Subsystem Overview

Analog Subsytem

The input guitar signal will need to adjusted to the proper voltage level before going into the ADC. It will also need to be amplified after the DAC and before it is sent out of the pedal. This will be done via analog amplification circuits.

Microcontroller Subsystem

The microcontroller is what will perform the filtering and processing of the guitar signal. It will also manage the I/O of the pedal, including the LCD display, rotary encoder, and compression switch. The microcontroller being used is the ESP32-S3.

Power Subsystem

The power of this board will come from a 9V wall outlet. We will have a regulator regulate the voltage down to 3.3V for the microcontroller, DAC, and ADC, and likely use the 9V for the digital subsystem.

ADC/DAC Subsystem

The analog signal coming in must be converted to digital so that the microcontroller can process it. It also must be converted back to analog after the processing is done. We plan to use 32 bit hardware controlled ADC/DAC to remove some strain from the microcontroller.

Subsystem Requirements

Analog Subsytem

The analog subsystem must amplify the incoming signal and bias it to the voltage that the ADC requires. This will allow us to get the most accurate readings. We also must amplify the outgoing signal so that it is strong enough for the guitar amplifier.

Microcontroller Subsystem

The microcontroller must be able to run off of 500mA of current. It also must have enough pins to interface with the DAC, ADC, LCD display, rotary encoder, and compression switch.

Power subsystem

The power subsystem must have separate regulators for the microcontroller, DAC, and ADC to minimize noise. The regulators for the ADC and DAC must be able to supply at least 50mA, and the regulator for the microcontroller must be able to supply at least 500mA.

ADC/DAC subsystem

The ADC and DAC must both be 32 bit resolution and must both be able to sample at 48kHz. Ideally, they should communicate with the microcontroller via I2S.

Tolerance Analysis

We concluded that the ESP32-S3, with its 240 MHz clock speed and dual-core architecture, offers enough processing power for real-time digital signal processing. Given that the pedal needs to process audio at a 48 kHz sampling rate with 32-bit resolution, each sample will require:

$$48{,}000 \text{ samples/sec} \times 32 \text{ bits/sample} = 1.536 \text{ Mbits/sec}$$

$$1.536 \text{ Mbits/sec} \div 8 = 192 \text{ KB/sec}$$

This is well within the I/O capabilities of the ESP32-S3, which supports communication protocols like I2S for efficient handling of the ADC and DAC. Additionally, the ESP32-S3 can handle over 5000 clock cycles per sample at a 48 kHz sampling rate, which provides sufficient headroom for realtime filtering, volume detection, and dynamic adjustment of effects, as required by our design.

Ethics and Safety

Our design involves digital signal processing and unique control mechanisms based on the dynamic response of the guitar input. To ensure compliance with ethical standards regarding intellectual property, we will conduct thorough research to confirm that our design does not infringe on existing patents or proprietary technologies.

Regarding privacy, while our pedal does not handle personal user data, we still take privacy concerns seriously. For example, if the pedal were to be expanded to include features like Bluetooth connectivity or smartphone app integration in the future, we would ensure that any data collected would be managed with strict privacy protocols. In this way, we would guarantee that no unauthorized data collection or sharing would take place, ensuring the user's privacy is protected.

Safety is another key consideration in our design. While the pedal operates at relatively low voltage (9V from a standard wall plug), it is still important to protect users from potential hazards. To ensure electrical safety, we will incorporate protective features such as proper insulation and overcurrent protection in the power subsystem. These features will mitigate the risk of electrical faults or short circuits that could harm the user or damage the pedal.