ECE 445
Spring 2025

Project #60 Proposal:
Digital Pitch Shifter for Guitar

Team Members:
William Chang (wqchang2)
Eric Moreno (emoren40)
Zhengjie Fan (zfan11)

TA: Shengyan Liu
Professor: Michael Oelze

1. Introduction

1.1 Problem

Guitarists without access to a tremolo system face significant limitations in their ability to create expressive vibrato and pitch-bending effects, which are essential for adding emotional depth and dynamic variation to their playing. Without these techniques, the guitar’s sound can feel static or restrained, especially in genres like rock, blues, and jazz, where pitch manipulation is crucial. Traditional tremolo systems, though effective in addressing this issue, require invasive modifications to the guitar body, such as routing or altering the bridge. These changes not only compromise the guitar’s original design but can also affect its sound and value. Additionally, such systems may not be suitable for all playing styles, or for guitarists who prefer a more minimalist approach. As a result, players seeking greater versatility in their instrument face the difficult choice between sacrificing their guitar’s aesthetics or settling for limited expressive capabilities. This is the gap the proposed project aims to fill.

1.2 Solution

The solution to the aforementioned issue is a compact, attachable digital pitch-shifting device that uses a sonic sensor to detect the proximity of the guitarist’s hand to the bridge of the guitar. As the player moves their hand closer or farther from the sensor, the pitch of the guitar signal is dynamically adjusted, allowing for real-time pitch shifts up or down. This enables the guitarist to perform expressive techniques like vibrato and pitch bending, similar to those provided by traditional tremolo systems, but without the need for invasive body modifications. Additionally, the device includes a switch or button that lets the player toggle between upward or downward pitch shifts, offering greater flexibility in controlling the pitch. This lightweight solution enhances the player's creativity while preserving the guitar’s natural design and playability. Furthermore, the additional buttons or switches can enable further effects such as reverb, chorus, or delay, giving the player more creative control over their sound. These augmentations enhance the guitarist’s ability to experiment with a wider range of tones and textures without needing to modify the guitar's body or permanently alter its design.

1.3 Visual Aid

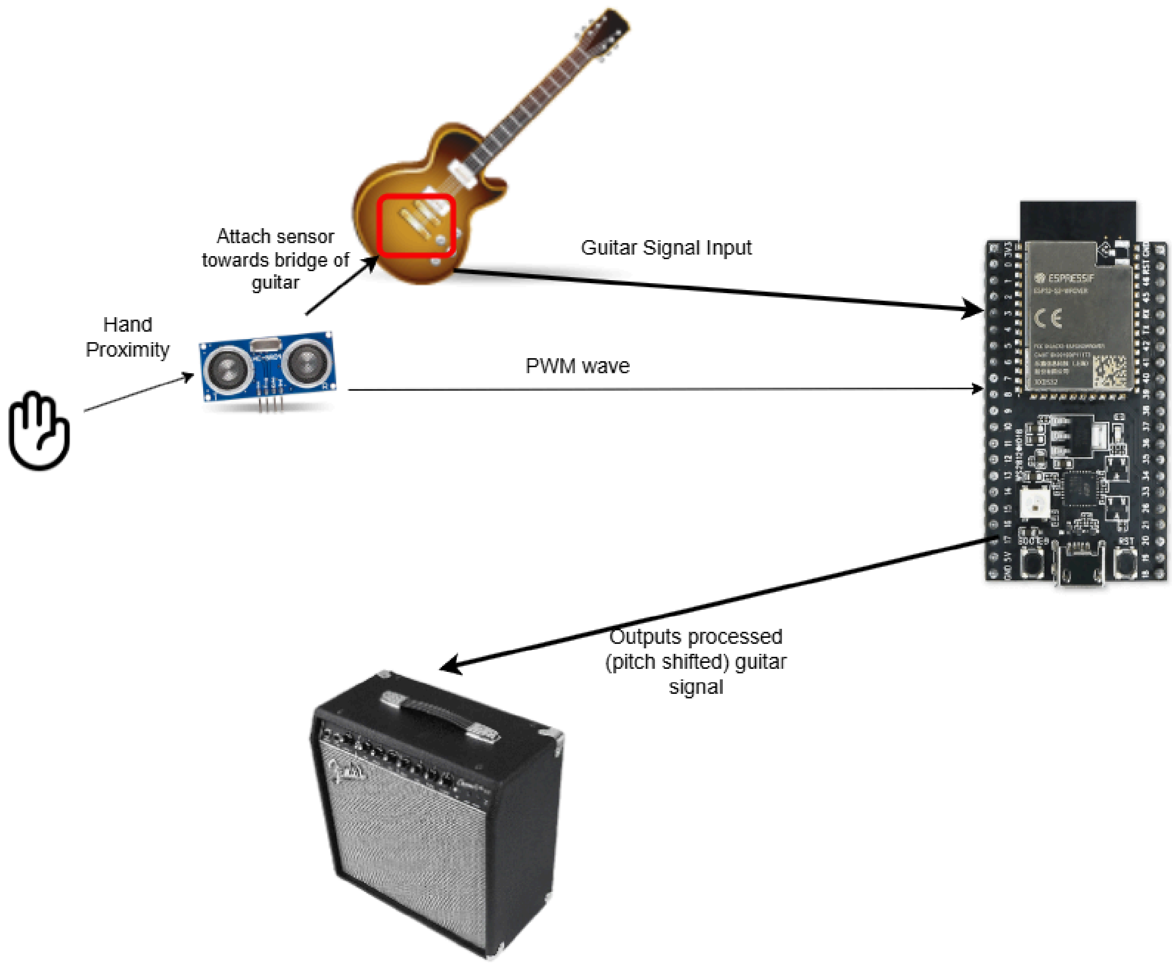

Figure 1: Visual Aid

1.4 High-Level Requirements List

Real-Time Pitch Shifting with Low Latency

- The system must process incoming guitar signals and shift their pitch without perceptible delay (<10ms total latency) to ensure natural playability.

High-Fidelity Audio Processing

- The pitch-shifted output must maintain at least 48 kHz sampling rate and 12-bit depth for ADC and 8-bit depth for DAC. (ESP32 microcontroller)

User-Controlled Pitch Adjustment

- The device must allow adjustable pitch shifting from -12 to +12 semitones.

2. Design

2.1 Block Diagram

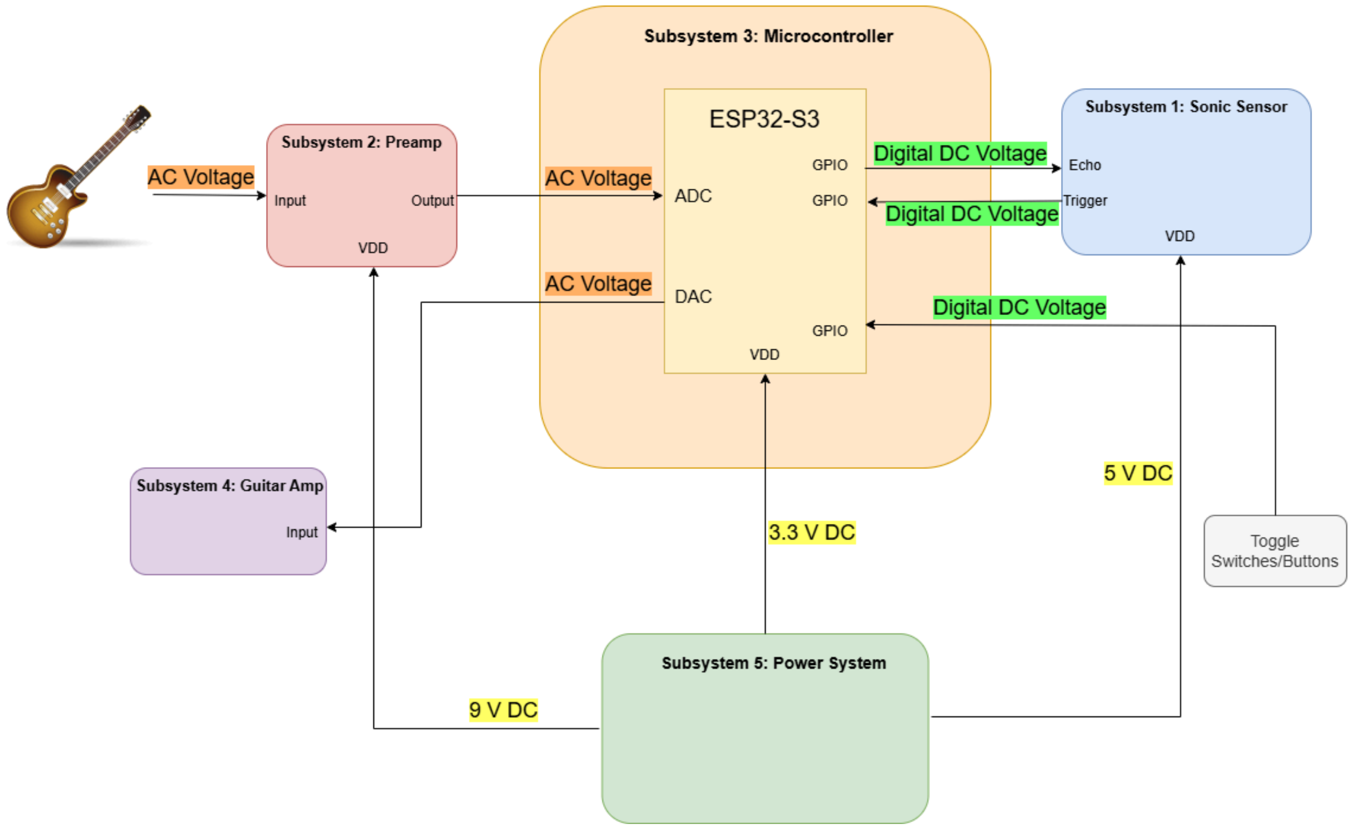

Figure 2: Block Diagram

2.2 Subsystem Overview

2.2.1 Subsystem 1: Sonic Sensor

The HC-SR04 ultrasonic sensor will play a crucial role in detecting the proximity of the guitarist's hand to the sensor, which will then be used to adjust the pitch of the guitar signal. The sensor operates using two primary pins: the Trigger pin and the Echo pin. The Trigger pin receives a pulse signal from the ESP32 to initiate the emission of an ultrasonic wave, while the Echo pin sends back a signal to the ESP32 that is used to calculate the distance based on the time it takes for the wave to return. This distance will dynamically influence the intensity of the pitch-shifting effect.

2.2.2 Subsystem 2: Guitar Preamp

A guitar preamp pedal will be placed between the guitar and the microcontroller to boost the guitar's signal, which typically ranges in the hundreds of millivolts. The preamp will increase the signal to a level suitable for the ESP32's ADC, ensuring that the microcontroller can properly process the audio input.

2.2.3 Subsystem 3: Microcontroller (Audio I/O, Signal Processing, Sensor I/O)

The ESP32 microcontroller will serve as the central unit responsible for managing both the input and output of signals, as well as performing real-time signal processing for the project. One of its primary roles will be handling audio input and output through its ADC (Analog-to-Digital Converter) and DAC (Digital-to-Analog Converter) pins. The ESP32 will convert the guitar signal from analog to digital using the ADC, process it with pitch-shifting algorithms, and then convert it back to analog using the DAC for output to a guitar amplifier. In addition to audio processing, the microcontroller will interact with the HC-SR04 ultrasonic sensor by sending a trigger pulse through its GPIO pin to the TRIG pin in order to initiate a reading. It will then read the output of the Echo pin to calculate the distance between the sensor and the player's hand, which will influence the pitch-shifting parameters. Furthermore, the microcontroller will manage user interactions such as toggling effects or adjusting parameters using additional GPIO pins connected to buttons or switches.

2.2.4 Subsystem 4: Guitar Amplifier

A 7-watt combo amp will be used to amplify and output the pitch-shifted guitar signal from the ESP32 to an audible level. After the microcontroller processes the audio and applies the pitch shift, the combo amp will boost the signal, making it loud enough for the guitar speaker to produce sound.

2.2.5 Subsystem 5: Power System

The power management system will use a 5V power supply to ensure stable operation of both the ESP32 microcontroller and the HC-SR04 ultrasonic sensor. Since the ESP32 requires 3.3V, a voltage regulator will step down the 5V supply to provide a stable 3.3V output for the microcontroller. The HC-SR04 sensor, which operates at 5V, will be powered directly from the same 5V supply to ensure proper functionality. A common ground will be shared between all components to maintain reliable communication. Additionally, since the HC-SR04's Echo pin outputs 5V, a voltage divider can be used to step down the signal to a safe 3.3V for the ESP32's GPIO.

2.3 Subsystem Requirements

2.3.1 Subsystem 1: Sonic Sensor

1. The system must accurately measure distance with a resolution of at least 1 cm.
2. It must provide stable and noise-free data to the microcontroller for real-time processing.
3. The sensor must operate within a range of 2 cm to 15 cm.

2.3.2 Subsystem 2: Guitar Preamp

1. The preamp must amplify the guitar's AC signal to a level suitable for ADC conversion.
2. It must provide low-noise operation with minimal distortion.

2.3.3 Subsystem 3: Microcontroller (Audio I/O, Signal Processing, Sensor I/O)

1. The system must sample the guitar signal at a minimum of 44.1 kHz with at least 12-bit resolution.
2. It must implement real-time pitch-shifting with a maximum processing delay of 10 ms.
3. The microcontroller must read sensor data and adjust pitch parameters dynamically.

2.3.4 Subsystem 4: Guitar Amplifier

1. The amplifier must provide a clean output signal with minimal harmonic distortion.

2.3.5 Subsystem 5: Power System

1. The system must provide stable 3.3V ± 0.1V and 5V ± 0.1V power sources for all components.
2. Overvoltage, undervoltage, and short-circuit protection must be implemented.

2.4 Tolerance Analysis

The sonic sensor plays a crucial role in controlling the pitch shift by detecting the distance between the guitarist's hand and the sensor. Accurate and timely readings are essential for real-time pitch-shifting control. The sonic sensor's performance is highly dependent on distance measurement accuracy, which is usually ±1 cm under ideal conditions. A deviation larger than this could lead to inconsistent pitch shifts or delays in detecting changes in the guitar's position, making it difficult to achieve smooth pitch modulation. Furthermore, the response time of the sensor is critical for real-time interaction with the system. A delay in sensor updates could introduce latency in the pitch-shifting feedback loop, causing noticeable lag between the guitar player's action and the system's response.

To quantify the impact of measurement accuracy and timing drift, we'll use the sensor's distance measurement model. The sensor typically uses an ultrasonic wave to measure distance, where the time is given by:

$$d = \frac{v*t}{2}$$

Where:

- d is the distance to the object,
- v is the speed of sound (approximately 343 m/s),
- t is the time of travel.

If the sensor has an accuracy of ±1 cm, this means that the error in distance measurement is approximately ±0.01 m.

$$\Delta t = \frac{2*\Delta d}{v}$$

$$\Delta t = \frac{2*(0.01)}{(343)} \approx 58.3\ \mu s$$

This time error could affect the precision of pitch-shifting calculations if it accumulates over time, especially in events of constant pitch shifting sequences where real-time updates are necessary. Even small errors in time measurement could result in audio discrepancies between sensor updates and the microcontroller. To ensure minimal impact, the system should handle this error through calibration of the sensor or microcontroller's sensitivity. By adjusting the threshold of distance and pitch shifting, we can increase the tolerance of the sensor readings so even small changes made in distance will be ignored and avoid triggering any unwanted changes.

3. Ethics and Safety

Ethical Considerations

Our digital pitch-shifting guitar project must follow ethical guidelines set by the IEEE and ACM Codes of Ethics to ensure fairness, responsibility, and safety in both its development and use.

1. Intellectual Property and Fair Use
    - Since our project may involve existing signal processing techniques or open-source software, we must ensure that we properly cite and follow all licensing agreements.
    - The IEEE Code of Ethics emphasizes honesty in authorship and respect for intellectual property. To follow this, we will credit all sources, avoid plagiarism, and ensure that any external components we use are legally allowed.
2. User Safety and Responsible Design
    - High audio output levels could damage hearing if not properly managed. To prevent this, we will limit volume levels, implement safety features, and test the device under different conditions.
    - The ACM Code of Ethics encourages designing systems that improve quality of life while minimizing harm. Our design will ensure that users are protected from sudden loud noises or unintended hardware malfunctions.
3. Accessibility and Inclusivity
    - Our device should be usable by all musicians, regardless of experience level. We will provide clear documentation, an easy-to-use interface, and setup guides to help users understand how to operate it safely and effectively.
    - Ensuring inclusivity aligns with the ACM Code of Ethics, which states that computing professionals should make technology accessible to a wide audience.

Safety Considerations

Our project includes electronic and software components, each of which presents potential safety risks. To ensure safe operation, we will follow industry standards and regulations while implementing features to minimize hazards.

1. Electrical Safety
    - Our device operates on low-voltage DC power, but it still requires protection against short circuits, overvoltage, and overheating to prevent damage or injury.
    - To mitigate these risks, we can use fuses, voltage regulators, and a physical containment box to protect both the device and the user.
2. Hearing and Acoustic Safety
    - Prolonged exposure to loud audio can cause permanent hearing damage. According to OSHA Noise Expectation Standards, sounds above 85 dB can be dangerous over extended periods.
    - Our design will include built-in volume limits with possibly gain control features to prevent excessive sound levels from being output.

- Additionally, we will provide safety guidelines in the user manual to help musicians understand the risks of prolonged exposure to high sound levels.

References

1. SparkFun Electronics, "Ultrasonic Distance Sensor - HC-SR04," [Online]. Available: https://www.sparkfun.com/ultrasonic-distance-sensor-hc-sr04.html. [Accessed: 13-Feb-2025].
2. Espressif Systems, *ESP32-S3 Datasheet*, [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32-s3_datasheet_en.pdf. [Accessed: 13-Feb-2025].
3. Espressif Systems, *ESP-IDF I2S API Reference (ESP32-S3)*, [Online]. Available: https://docs.espressif.com/projects/esp-idf/en/stable/esp32s3/api-reference/peripherals/i2s.html. [Accessed: 13-Feb-2025].
4. Digi-Key Electronics, *What Is the I2S Communication Protocol?*, [Online]. Available: https://www.digikey.com/en/maker/tutorials/2023/what-is-the-i2s-communication-protocol. [Accessed: 13-Feb-2025].