EEL 4914 Electrical Engineering Design
(Senior Design)

Project Preliminary Design Report

28 January 2008

Project Title: Self-Tuning Guitar

Team Name: The Toms

Team Members:

Thomas Negley
tombo438@ufl.edu
(863) 207-5871

Thomas Johnsen
colttom@ufl.edu
(954) 464-928

Project Abstract:

Our project consists of an electric guitar equipped with the ability to detect the frequency at which an individual string is vibrating and adjusting the tension on the string until it vibrates at a predetermined desired frequency. The string frequency will be captured using the guitar's standard output jack, and then a microcontroller attached to the back of the guitar will interpret the frequency and send commands to a collection of motors (most likely stepper motors) on the head in order to adjust the string that is playing. The unit's controls will consist of a six-position switch used to select the string and a set of LEDs that indicate the tuning status. We ultimately intend to power the unit from an attached rechargeable battery.

Table of Contents

Features

The self-tuning guitar is just that, self tuning. All a guitarist will need to do is turn on the tuning system and strum the strings until tuned. The fiddling with the tension in the strings is left to the guitar and more specifically the motors on the guitar. This takes most of the work from the guitarist and places it on the tuning system on the guitar. People who regularly play guitar or who are just starting out will find this system very beneficial to use just before playing.

The self tuning guitar will be extremely useful for the musician who wants to effortlessly tune their guitar. All of the processing is done on the guitar, so there is no external device that needs to be lugged around and attached to the guitar. This automated tuner will be easy to use, with a simple set of switches and LEDs to control the device. Once the user indicates they are ready to tune, a simple strum of the strings will get the tuning process started—and finished.

Objectives

There are three main objectives of the Self-Tuning Guitar. The first objective is the detection and determination of the frequency of the string being tuned. The second objective is to determine if the string being tuned needs to be tightened or loosened to attain the desired tuning. If the frequency of the string is too low, the string needs to be tighten and conversely if the frequency is too high, the string should be loosened to attain the proper tuning. This leads to the third objective which is to control stepper motors to tighten or loosen the strings. Once it is determined whether the frequency is too low or too high, the motor will have to tighten or loosen the string, respectively.

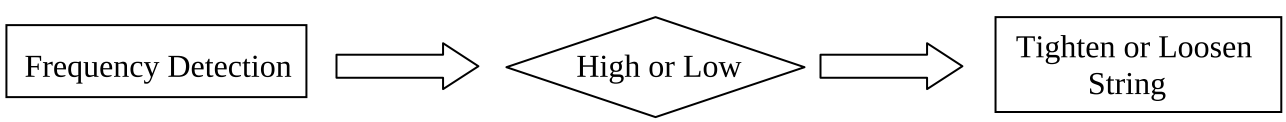

Figure 1: Process Diagram of the three high level objectives

Technical Concepts

The technical implementation of the three main objectives can be accomplished in the five steps illustrated below.

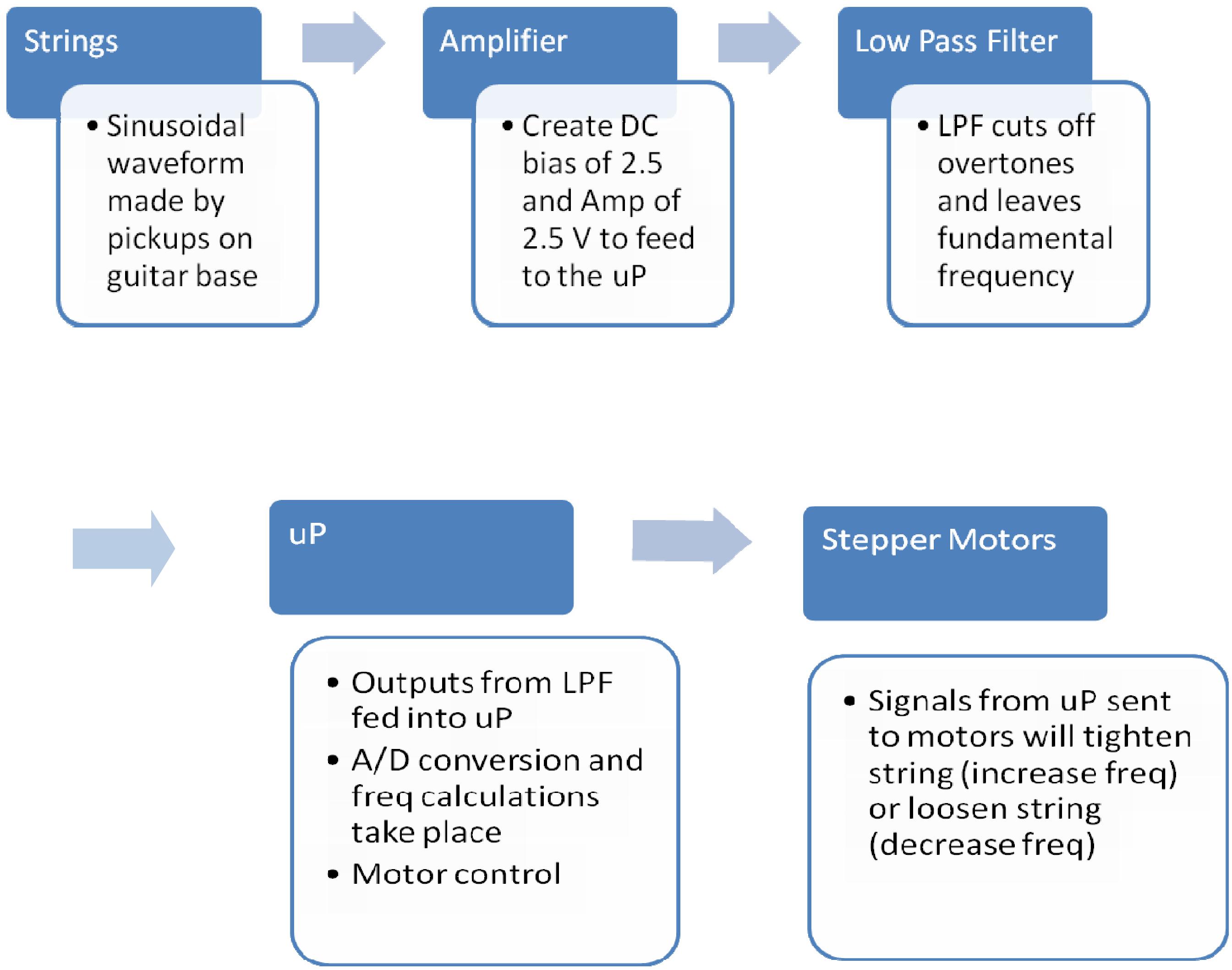

Figure 2: Illustration of the technical concepts to be used

This can be done using the A/D converter on a microprocessor such as the ATmega128. The amount of time between samples of the same magnitude can be used to calculate the frequency of the sine wave resonating off of the guitar string. Using a standard tuning based on the A note set at 440Hz, each open string has a specific frequency when tuned. The low E, or the open sixth string has a frequency of 82.4Hz; the fifth string 110.0Hz; the fourth string 146.8Hz; the third string 196.0Hz; the second string 246.9Hz; the high E, or the first string tunes to 329.6Hz. These values can be stored in memory and be compared to by the frequency detected by the processor.

The output of the LPF will be an analog signal. In order to do calculations within the uP we will need to convert to digital values. In order to do this, the A/D Converter will sample the incoming data and record the values in the A/D registers. A “zero” value of 2.5V will be used in our calculations so attenuation in the signal can be ignored. In order to calculate the frequency, the time elapsed in between the “zero” value points in the waveform will be measured with the uP timer.

One major design choice that has been made concerns the type of motor to use. Simple DC motors were immediately ruled out because of the need for precise control and the need to use both forward and reverse motion. The choice was then made between servos and stepper motors. Servos were ruled out because even though they provide the necessary precision, they normally do not allow the motor to turn more than one full revolution. Stepper motors were ultimately chosen because they are capable of being precisely controlled and they allow more than one full revolution in both forward and reverse directions.

A simple way to control the stepper motors without worrying about overshooting too much is to have varying degrees of rotation hard coded into the uP. For example, if the frequency is off by 10% the motor should only turn a small amount and for a 30% error the motor should turn more. The exact amounts of motor control will be determined after some experimentation with the motors