Perfect Pitch Instruments
School of Engineering Science
Burnaby, BC · V5A 1S6

ensc340-pp@sfu.ca

November 10, 2000

Dr. Andrew Rawicz
School of Engineering Science
Burnaby, British Columbia
V5A 1S6

Re: ENSC 340 Design Specifications for an Automatic Guitar Tuner

Dear Dr. Rawicz:

Attached you will find Perfect Pitch Instruments' *Design Specifications for an Automatic Guitar Tuner*, a document which outlines the design specifications for our ENSC 340 project.

We are currently in the process of designing and building a device that will automatically tune a guitar as quickly as possible by mechanically tightening or loosening each string. This system requires a controlled feedback mechanism that will bring each string to resonate within a tight range of its ideal frequency. This device features an interactive menu system with a display for quick and easy programming by the user.

The purpose of this design specification is to detail the design and testing that our completed project will fulfill. The document lists the relevant design information needed for successful project completion. Included in this document are design details of the sensor input processing unit, the mechanical actuating unit, and the programmable user interface unit.

Perfect Pitch Instruments is composed of five sharp, creative and talented engineering science students including Gina Millar, Maria Trinh, Aaron Schellenberg, Terrence Yu and Reva Vaze. Should you have any questions or concerns regarding our proposal, please feel free to contact me at (604) 584-8926 or via email.

Sincerely,

Maria Trinh

Maria Trinh
CEO
Perfect Pitch Instruments

Enclosure: Design *Specifications for an Automatic Guitar Tuner*

Perfect Pitch Instruments

Design Specifications for an Automatic Guitar Tuner

Project team: Gina Millar
Aaron Schellenberg
Maria Trinh
Reva Vaze
Terrence Yu

Contact personnel Maria Trinh
ttrinha@sfu.ca

Submitted to: Dr. Andrew Rawicz
Steve Whitmore
School of Engineering Science
Simon Fraser University

Date: November 10, 2000

Executive Summary

With the success of the rock/pop music industry, more and more people becoming inspired to play the guitar. Electronic tuners and pitch pipes exist to help guitar tuning, but they are a hassle, and are sometimes an embarrassment to use when performing in front of others. Automatic tuning guitars and retroactively fitted tuners are available on the market, but often for a price that is greater than the original guitar itself. Our goal is to design a device will be capable of tuning a guitar quickly, cheaply and automatically, and will be installed in a guitar at the time of manufacture.

The Perfect Pitch guitar tuner would be suitable for all skill levels ranging from beginner to professional. It could help to ease the learning curve for a novice guitarist or provide a performer on stage with a quick and effective means for tuning their guitar between songs to whatever settings they desire. Better yet, the price of a guitar with this device installed would only cost a fraction more than a guitar without the tuning mechanism.

The design and construction of the Perfect Pitch Guitar Tuner is dedicated to the development of the following features:

1. A quicker tuning method than any manual means.
2. A tuning accuracy with a steady-state error below the measured Just-Noticeable-Difference (JND).
3. An easy user interface.
4. An affordable cost.

Perfect Pitch Instruments is dedicated to the completion of this project by January 2001.

Table of Contents

List of Figures

List of Tables

Introduction

Very few people have the ability of "perfect pitch," the ability to identify a note on the musical scale without having any reference note for comparison. Although the scientific and psychological aspects of perfect pitch are not well known, one thing is certain: whether perfect pitch is an inborn talent or an acquired trait, it need not be an essential or prerequisite for a musician.

The goal of our project is to design and develop an automatic electric guitar tuner that would eliminate the hassle involved with tuning a guitar manually. Although tuners exist in the marketplace today, many are handheld requiring the user to pluck a string with one hand, hold the tuner with another hand, and adjust the tension in the strings with another. Clearly too many hands are needed, and the tuning process is unnecessarily time consuming. The few existing automatic guitar tuners cost thousands of dollars, and prove unaffordable for the general public.

We envision an automatic guitar tuner that would be installed in a guitar at the time of manufacture. This accessory will be capable of tuning a guitar with one simple strum, and would be suitable for all skill levels ranging from beginner to professional. The tuner can help to ease the learning curve for a novice guitarist or provide a performer on stage with a quick and effective means for tuning their guitar between songs to whatever settings they desire. This implementation will also be affordable to most consumers.

This document outlines the high-level design specifications of this project, which include the design details of the signal input processing unit, the mechanical actuating unit, and the programmable user interface unit.

1. System Overview

The controlled feedback mechanism is diagrammed in Figure 1. Essentially, the user all but has to pluck the strings on the guitar and the tuning is done automatically. The sounding audio frequencies are conditioned and processed electronically, and are subsequently provided as inputs to a control module. After comparing the pitch of the strings to their desired pitch, the control module adjusts the tension in the strings electro-mechanically. The controlled feedback implementation allows the actuators to be adjusted dynamically while tracking the pitch of the string. The process completes once the strings are tuned to their desired frequencies. Then it is time to rock!

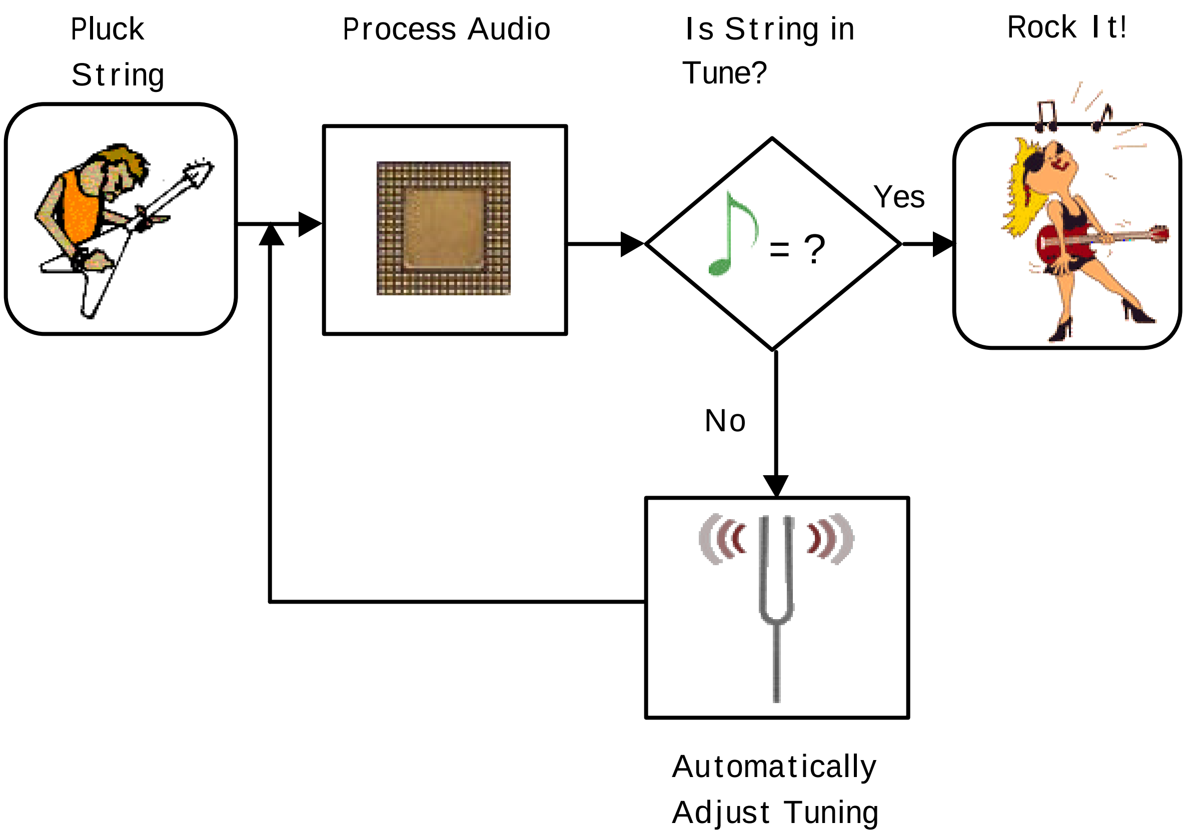


Figure 1: General Feedback Mechanism

Essentially, once the strings are installed and manually tuned at the head of the guitar within a tolerance of +/- 1 semitone of the desired pitch, an automatic fine-tuning system will take over (at the base of the guitar) and precisely adjust each string within a very small tolerance (below the Just-Noticeable-Difference).

To implement this controlled feedback mechanism, a high-level design is proposed. Its overview is shown in
Figure 2.

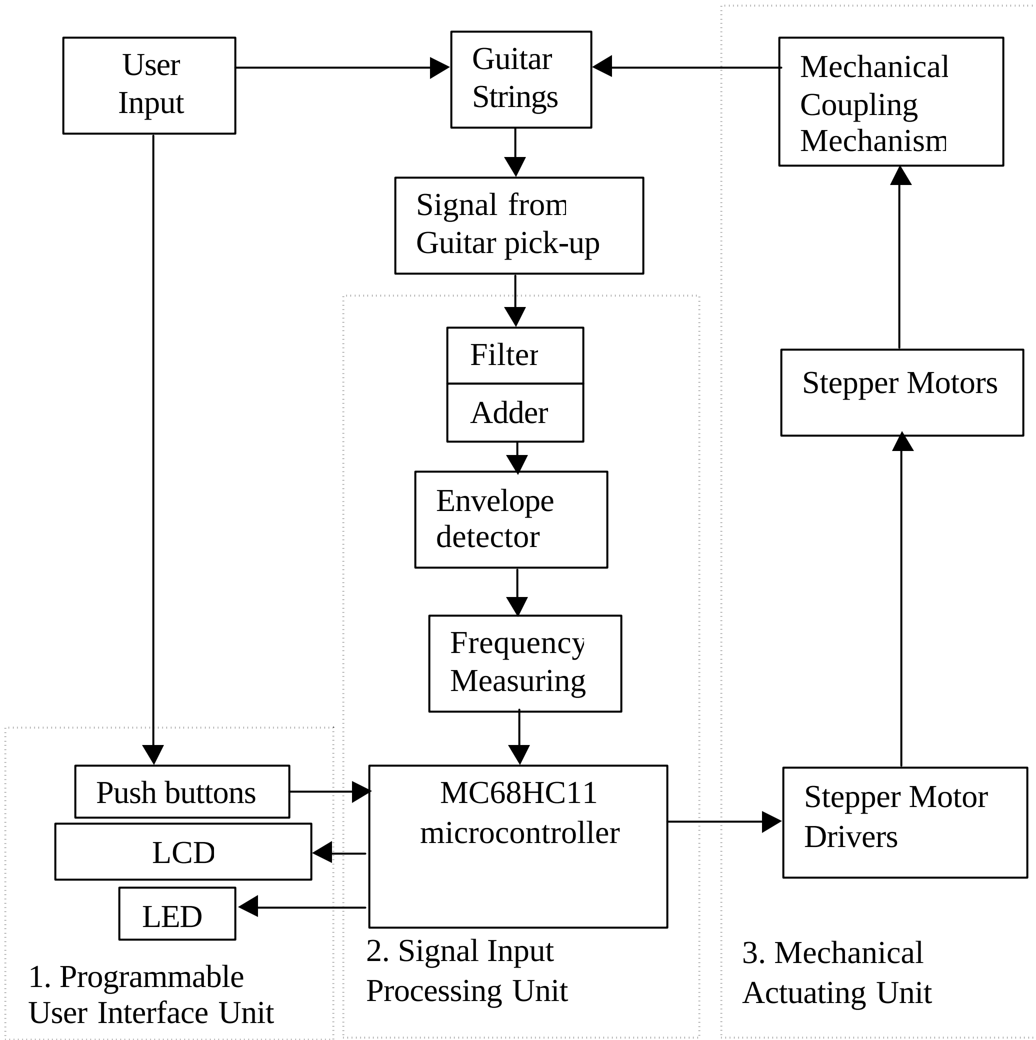


Figure 2. Design System Overview

The entire system is composed of three main modules:

1. the programmable user interface unit
2. the signal input processing unit, and
3. the mechanical actuating unit

The interactions between each of these modules work to achieve the controlled feedback mechanism as shown in Figure 1. The user programs the string settings using the Programmable User Interface Unit (1) and then plucks the guitar strings. The signal from the plucked string exits the guitar via the guitar pick-ups. The Signal Input Processing Unit (2) then processes this signal. The deviation of this signal from the ideal signal will determine the actuating message sent by the MC68HC11 microcontroller to the Mechanical Actuating Unit (3). The motors will either tighten or loosen the guitar strings (depending on the message from the microcontroller) via the mechanical coupling mechanism. The LCD and LED are also updated by the microcontroller for monitoring by the user.

Each of these system modules will be further detailed in the following pages.

2. Programmable User Interface

2.1 User Interface Overview

Due to the nature of our project the user interface is made as limited as possible so as not to clutter up the body of the guitar. A visual representation of the user interface is shown in Figure 3.

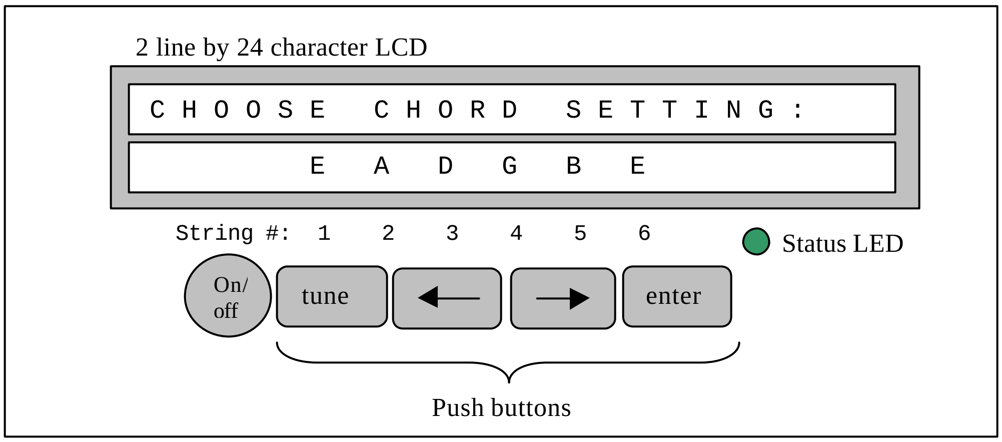


Figure 3. Front-End of User Interface

The controls that the user has at her disposal are the following:

- 1 On/Off switch - System power button and Emergency Stop button
- 1 "Tune" button - Toggles between Tune/Not Tune modes
- 3 programming push buttons – Used to program system settings
- 1 - 2 Line by 24 character LCD display – displays feedback for the user
- 1 Bi-colour LED - Indicates tuning status:
  - Green – string is tuned within specification,
  - Red – String is not within specification and needs to be re-plucked.

The user must be able to interface with the automatic tuning system to:

1. Program the chord settings to which the tuner will set each string
2. Tell the system when to start tuning
3. If necessary, abort the tuning process

The initial LCD screen shows the standard strings settings for the guitar in the "Choose Chord Setting" window. The user can change each string from the standard tuning by moving the cursor (using the ← and → buttons) to highlight the desired string, and then pressing by "enter". Once the user has done so, a list of possible pitch settings for that particular string is made available, from which the user can again select, using the ← and → and "enter" buttons.

Once the user has chosen the chord settings, she must press the "tune" button to ready the tuning system. Once a string has been plucked, the mechanism will be activated. The user can pluck any string, and the cursor on the LCD will highlight the string being tuned. Once each string has been tuned successfully, the LED will turn from red to green.

For any reason, the user can abort the tuning process by pressing the On/Off button at any time.

Table 1 shows the possible status messages displayed by the LCD once the user has pressed the "tune" button.

Table 1. LCD status displays

Status Message	System Status
"Tuning in progress"	The system is in tuning mode; no new data entry will be accepted
"String #X successfully tuned"	The system has exited tuning mode, with the string in tune.
"Pluck String Again"	The system is still in tuning mode and requires the user to pluck the string again.
"Error: string too tight"	The system detect that a string may snap if tightened any further. The user either has chosen a poor string setting, or the string is old and must be changed.
"Error: motor jam"	The motor is not rotating when it should.
"Error: tuning failed, system failure"	The system has experienced an electrical or mechanical failure.
"Error: too many strings"	The system detects that more than one string was plucked at the same time, and not action is taken.
"Error: tuning failed, check string"	After 5 attempts, the system aborts the tuning process and the user must check the string manually.

The user has the option of resetting the system at any time by pressing the On/Off button twice.

2.2 Hardware Implementation of User Interface

Figure 4 shows how the LCD, the push buttons and the bi-colored LED are all connected to the MC68HC11 microprocessor.

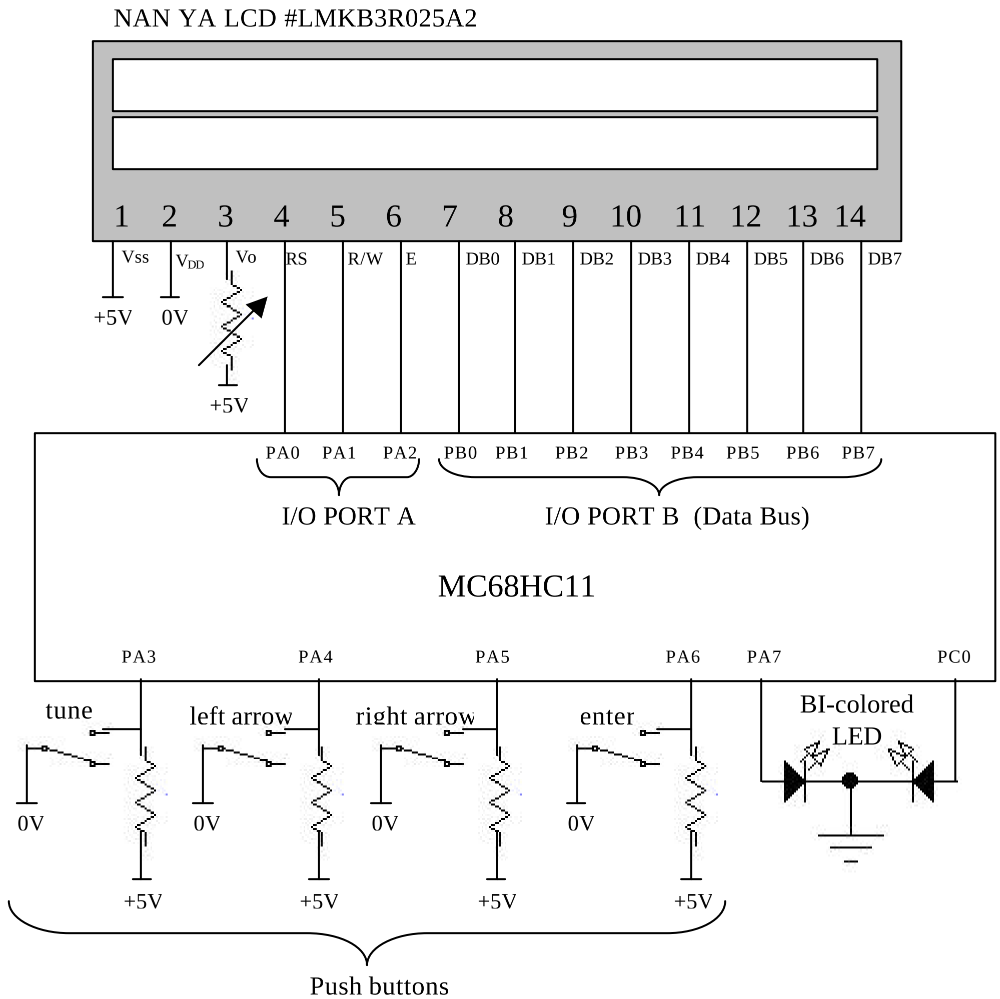


Figure 4. Hardware Implementation of the User Interface

The LCD requires all of the microprocessor's Port B I/O pins to deliver and received 8-bit words (DB0-DB7), plus three more pins from Port A for its enable, read/write and RS lines.

The LCD I/O pins are described in the table below:

Table 2. LCD pin-outs

LCD Pin Number	Symbol	Level	Function	Pin connection on the MC68HC11
1	Vss	0V	Power Supply	-
2	V_{DD}	+5V		-
3	Vo	0-5V	Variable contrast	-
4	RS	H/L	Instruction/data input	PA0
5	R/W	H, H→L	Data read/write	PA1
6	E	H/L	Enable signal	PA2
7	DB0	H/L	Data bus line	PB0
8	DB1	H/L	Data bus line	PB1
9	DB2	H/L	Data bus line	PB2
10	DB3	H/L	Data bus line	PB3
11	DB4	H/L	Data bus line	PB4
12	DB5	H/L	Data bus line	PB5
13	DB6	H/L	Data bus line	PB6
14	DB7	H/L	Data bus line	PB7

Each push button (“tune,” “left arrow,” “right arrow,” and “enter”) requires another Port A pin.

Table 3. Push button pin connections

Push button	Function	MC68HC11 pin connection
Tune	Sets the system to “tune” mode	PA3
←	Moves the LCD cursor left	PA4
→	Moves the LCD cursor right	PA5
Enter	Enters the user’s choice	PA6

The bi-colored LED requires two more pins (PA7 for red and PC0 for green).

The user interface requires a total of 17 I/O pins from the microprocessor.

2.3 User Interface Firmware

As soon as the power is turned on, the user interface firmware waits for the user to press any button, and then reacts to each button press. Turning the power off will cause the firmware to shut down. Figure 5 shows a high-level flowchart of the user interface algorithm.

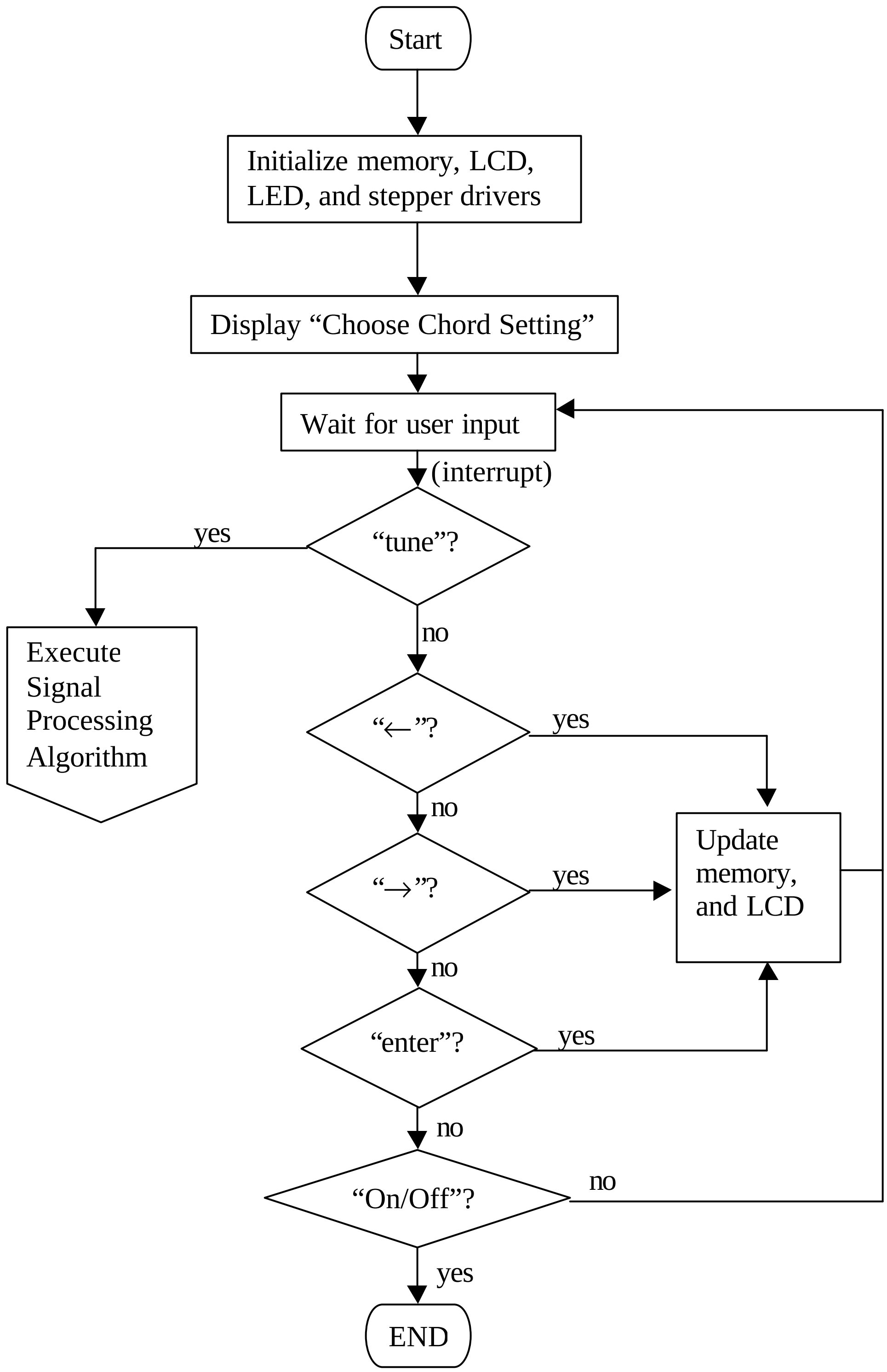


Figure 5. High-level Block Diagram of User Interface Algorithm

3. The Signal Input Processing Unit

A block diagram implementation of the Signal Processing Unit is given in Appendix A. The unit processes the input signal from the guitar pickups in two parallel operations. The entire process relies on "beat detection", and consists of frequency resolution and frequency to voltage conversion. The need for two parallel units will soon become apparent.

3.1 Frequency Resolution of Input Signal

To resolve the frequency of the input signal, we need to perform the following

1) Modulation
2) Filtering
3) Envelope Detection
4) Limiter

Modulation involves adding a *Reference Square Wave* to the *Input Signal*. The modulated signal then passes through the clock tunable low pass filter (having a cuttoff frequency close to the desired string frequency) to remove harmonics from the input signal, as well as from the additive square wave. The output of the filter is a modulated signal with a sinusiodal envelope having a frequency equal to the difference between the *Reference Square Wave* and the *Input Signal.* To detect this envelope, we have implemented and envelope detector consisting of a simple diode, and RC circuit. The DC component of the signal if removed through a coupling capacitor. The output of the envelope detector is then fed into a limiter circuit that squares up the input signal as long as its amplitude is above a predetermined threshold level. Thus the limiter output is a square wave with constant amplitude, and frequency equal to the difference between the *Reference Square Wave* and the *Input Signal.*

3.2 Frequency to Voltage Conversion

Since we want to determine if the string should be tuned up or tuned down, we need to have two parallel units as is shown in the block diagram. The frequency of Reference Square Wave #1 is set slightly lower than the desired frequency and the frequency of Reference Square Wave #2 set slightly above the desired frequency.

The limiter outputs are connected to the input compare channel of the MC68HC11 microcontroller. The microcontroller is configured such that it generates an interrupt and every time a zero crossing on the input signal is detected. The interrupt service routine latches the local timer value into memory, and thus the frequency of the limiter output signal can be measured. If the frequency of the signal of Limiter #1 is greater than the frequency of the signal of Limiter #2, we conclude that the string needs to be tuned down and vise versa. The microcontroller now knows which direction the motors should turn in, and by what amount.

3.3 Microprocessor Selection

Since we decided not to process our signals digitally early on, we could rule out a digital signal processor fairly soon in the design process. It was decided that the Perfect Pitch Automatic Guitar Tuner called for a microprocessor that would be able to process an incoming signal, control a user interface on an LCD, and actuate the motors that would ultimately tune the guitar strings. Various microcontrollers were considered, but the one chosen was an E series Motorola 68HC11. Our rationale was based upon availability (we were given the chip for free), familiarity (all our group members have had past experience with the chip), and capability to meet all our processing needs. The list below highlights features that are important and beneficial to our system.

- M68HC11 CPU
- 512 Bytes of On-Chip RAM and 2048 bytes of EEPROM
- 8-Channel 8-Bit Analog-to-Digital (A/D) Converter
- 16-Bit Timer System
    - Three Input Capture (IC) Channels
    - Four Output Compare (OC) Channels
- 8-Bit Pulse Accumulator
- Real-Time Interrupt Circuit
- 38 General-Purpose Input/Output (I/O) Pins
    - 16 Bi-directional I/O Pins
    - 11 Input-Only Pins
    - 11 Output-Only Pins

A diagram of the 68HC11 architecture is included in Appendix B.

3.4 Signal Input Processing Firmware

Figure 6 shows a high-level block diagram of the signal input processing algorithm as described in Sections 3.1 to 3.2.
NOTE: `STRING`, `NUM_PLUCKED`, `ZEROS1`, `ZEROS2`, `F1` and `F2` are variables.

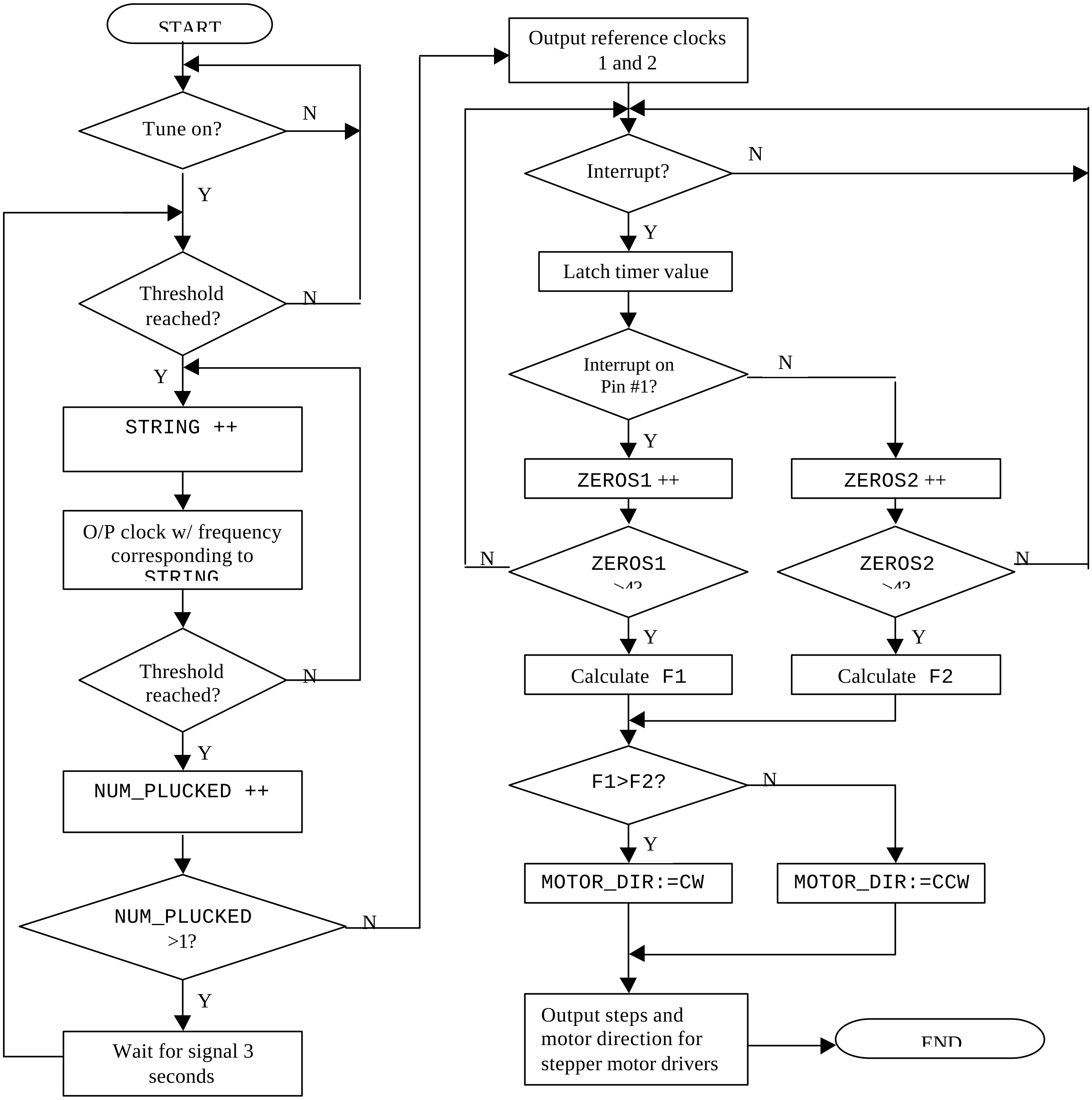


Figure 6. High Level Block Diagram of the Signal Input Processing Algorithm

4. Mechanical Actuating Unit

4.1 Guitar and Bridge

The electric guitar we selected is a 3 pick-up guitar (2 single coil, 1 Humbucker) complete with a Kahler floating bridge (similar to a Floyd Rose). After some preliminary design work, it was decided that the most suitable way to interface motors to the guitar strings for our purposes is to use a pre-manufactured floating bridge.

The most common embodiment of the floating bridge is the patented Floyd Rose Floating Bridge, in which the entire bridge pivots about two vertical posts aligned across the body of the guitar (i.e. perpendicular to the strings). The term “floating” refers to the fact that the bridge is held in place by the counter-balancing forces of the six tensioned strings, which travel along the top of the body, and 2-3 tensioned springs traveling underneath the body Figure 7. The two pulling forces cause the bridge to balance in a somewhat horizontal position. A lever attached to the bridge (often called a tremolo bar or whami bar), permits the guitarist to pivot the bridge around the vertical posts by exerting torque on the lever. This causes a change in string pitch, as the tension in all strings is either increased or decreased.

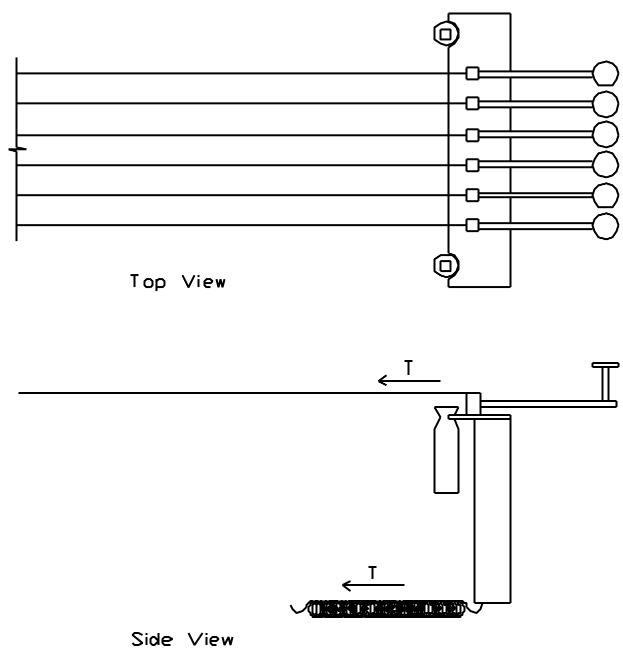


Figure 7. Floating Bridge

The guitar strings tend to go out of tune when repeatedly subjected to strains caused by the yanking of the tremolo bar. In order to reduce the susceptibility of the strings to detuning, a locking nut, consisting of a simple clamping mechanism is installed at the top of the guitar neck. This clamps the strings in place so that the tuning pegs in the head of the guitar do not unwind. When the nut is locked, however, the guitarist is unable to tune her guitar using the conventional tuning pegs, therefore, a secondary set of tuning pegs, called "fine tuners" are installed on the floating bridge itself, allowing the guitarist to tune her guitar after the nut has been locked.

We shall interface our motors to these fine tuners. Some of the advantages that these tuners provide are: (1) a mechanical advantage, provided by levers and screw drives, that significantly reduces the torque requirements of our motors (see Figure 8), (2) no holding torque is required to keep the string at the desired tension, and (3) they eliminate the complicated task of designing our own string tensioning mechanism. One disadvantage to this solution, however, is that we are limited to the tuning range of the fine tuners, which varies from 5 semitones for the low E string to 2.5 semitones for the high E string.

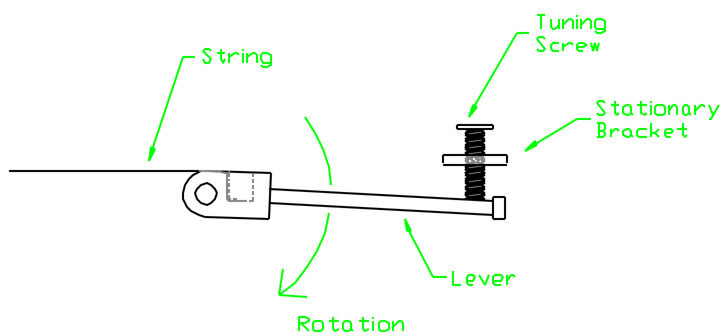


Figure 8. Fine Tuner (Side View)

4.2 Coupling Device

This mechanical device couples the motor power to the floating bridge via the fine tuners. For each string, we shall replace the fine tuning peg shown in Fig 2 with our own shaft coupling device Figure 9

The new threaded shaft will be coupled to the motor shaft by way of a simple belt and pulley system Figure 9. By adjusting the ratios of the pulley radii, we can add further mechanical advantage to the system by making the radius of the secondary pulley larger than the primary (motor) pulley.

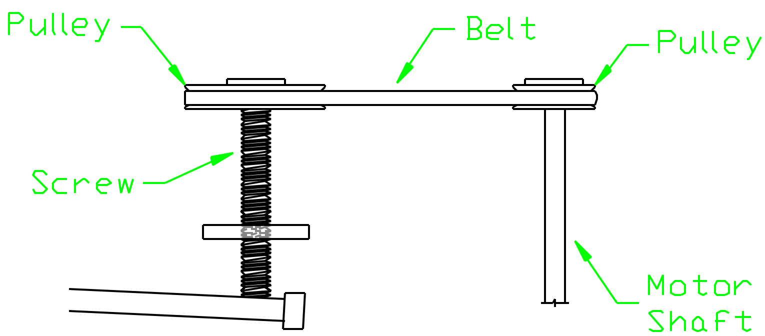


Figure 9. Coupling Device

4.3 Motors

Up to six motors shall be installed in the body of the guitar. A cavity in the back of the guitar shall be routed out to install the motors, with the drive shafts poking out of the top of the guitar body near the floating bridge. The guitar body is approximately 1 ¾" thick and 3 ½" from the back of the bridge to the end of the guitar. We realize that due to these physical constraints, it may not be possible to install all six motors in the selected guitar. Every attempt will be made to install as many motors as possible, however, regardless of these constraints, the electrical system is designed to accommodate up to six motors.

The fine tuners on the Kahler floating bridge are not perpendicular to the horizontal plane of the guitar body; rather they are tilted 5° - 10° from the perpendicular axis. This poses an additional complexity when mounting the motors, however, this can be overcome by using angles spacers between the motor and the mounting surface. This way the motor shafts and tuning shafts are parallel, as shown in Figure 9

Each string shall be independently controlled by a miniature stepper motor. When comparing stepper motors to DC servomotors, we found that steppers tended to be shorter and wider than DC servomotors, which better suits our needs, given the physical restriction imposed by the shape of the guitar body. Stepper motors are readily controlled by a microcontroller and provide accurate discrete angular positioning in an open-loop control configuration.

We have yet to determine the exact torque required by our tuning mechanism, however, in preliminary tests, the estimated required torque was comparable to the output of a NEMA 17 disk drive stepper motor at low to medium rpm. Our selected motors shall likely be one of: Mitsumi model# M25SP-5, M35SP-8, M35SP-9, or equivalent. These are all high performance motors, with high torque outputs per physical size. They have angular resolutions of 24 or 48 steps/rev, which is plenty for our task, however, the resolution can be increased by adjusting the ratio of pulley radii described above.

4.4 Driver Circuitry

As mentioned earlier, our design calls for six stepper motors to be controlled independently. In order do this effectively with our HC11, we shall use two 2-8 decoders to route the output pulse control signals from the HC11 to one of the six stepper motors. Additional I/O from the HC11 will be required to control the decoders.

The six pairs of outputs from the decoders ICs will be amplified by power transistors to drive the motors. These shall likely be in a Darlington configuration.

4.5 Power Supply

The automatic guitar tuner will run off of a standard wall socket power supply using a 120VAC to 24 VDC adaptor. The required voltages for different components (i.e. 24VDC for motors, 5VDC for LCD, microprocessor, etc.) can be obtained by stepping down the 24 VDC using voltage dividers and regulators.

5. Testing

5.1 Programmable User Interface

The user interface will be tested using the following procedure. The signal processing procedures in the HC11 will be preprogrammed to return the appropriate values to the UI.

1. Turn the system on using the power switch.
2. Using the buttons and LCD, program the tuner to standard tuning and select a string.
3. Set the “Tune” switch.
4. Trigger the threshold externally. Verify that the LED illuminates red, then green.
5. Turn tune mode off.
6. Program a different tuning, and repeat steps 3 through 5.
7. Turn the system off.

5.2 Signal Input Processing Unit

The input processing unit will be tested using the following procedure. Ensure that the tuner has been set to standard tuning before starting this test procedure.

1. Pluck the low E string of the guitar. View output of filter on an oscilloscope. The signal should be a clean sine wave with a frequency of approximately the standard tuning (86Hz).

2. Connect the oscilloscope to the signal conditioning stage output. Pluck the string again and observe the output signal. Verify that the signal is a low frequency square wave.
3. Tune the low E string to a frequency slightly lower than the standard tuning. Connect the oscilloscope to the output of the microprocessor (output to the mechanical unit). Pluck the string and view the output. Verify that the output signal is the one required to tighten the string.
4. Tune the string slightly higher than the standard tuning and repeat step 3. Verify that the output signal will loosen the string.
5. Repeat steps 1 through 4 for each string.

5.3 Mechanical Actuating Unit

The mechanical system will be tested using the following procedure. Set the fine tuners to their centered position before starting.

1. Generate a signal that will turn the motors appropriately.
2. Apply the signal to the mechanical system input.
3. Observe the mechanical system. Motors should stop when fine tuners reach their limits.
4. Repeat 1 to 3 for the opposite direction.
5. Repeat steps 1 through 4 for each motor.

5.6 Overall System

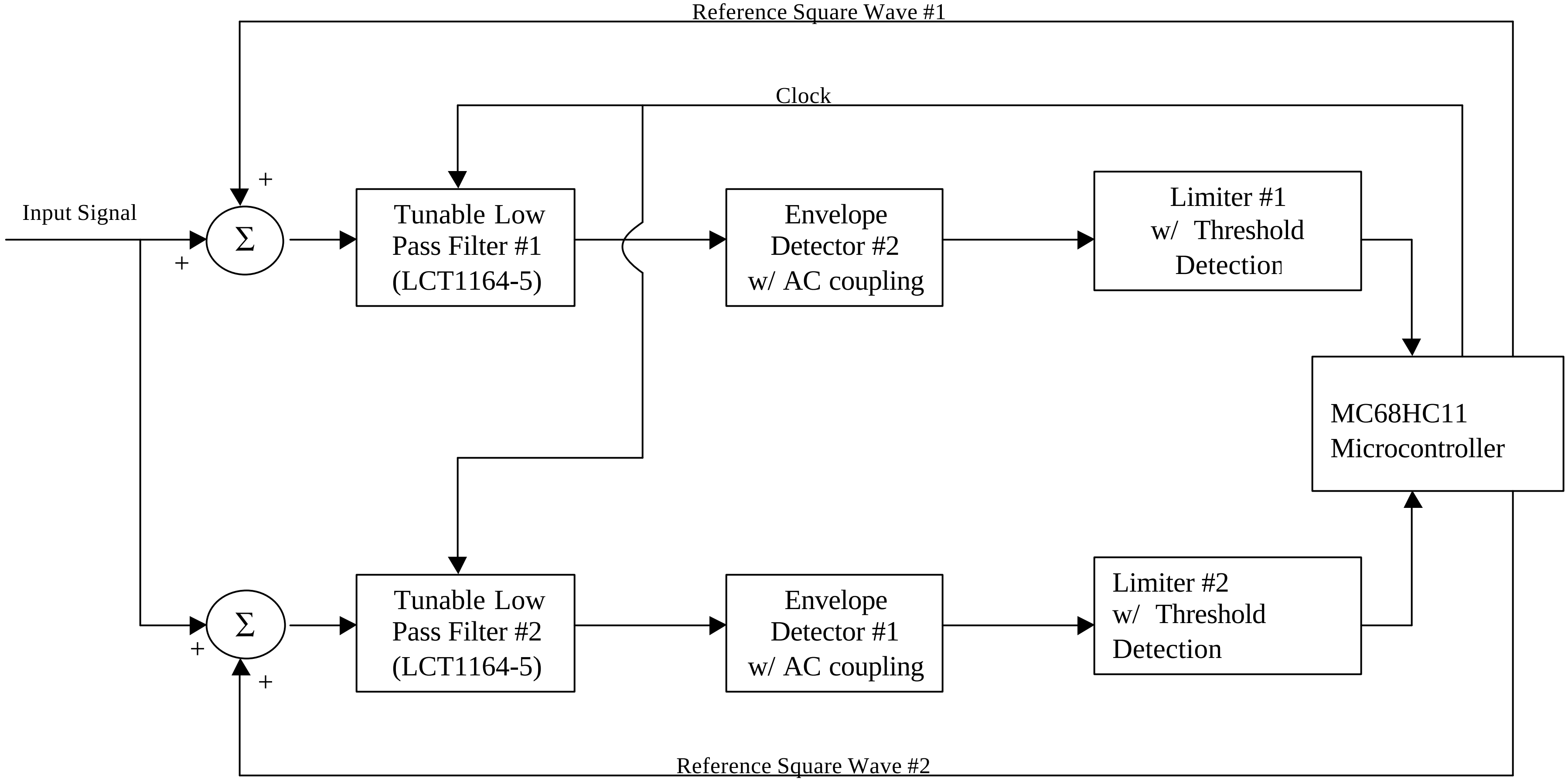
The entire guitar tuning system will be tested as follows:

1. Verify that the fine tuners are in centered position.
2. Tune the low E string to standard tuning minus one semi-tone.
3. Turn on the tuning system using power button.
4. Program the user interface to standard tuning.
5. Set the device to “Tune” mode.
6. Pluck the low E string. Verify that the LED illuminates red.
7. Wait until the tuner has finished. The LED should now be green.
8. Turn the tuning mode off.
9. Verify that the string frequency is within the specification limits.
10. Repeat steps 1 through 9 for the standard tuning plus one semi-tone.
11. Repeat steps 1 though 10 for each string.

This page is left intentionally blank.

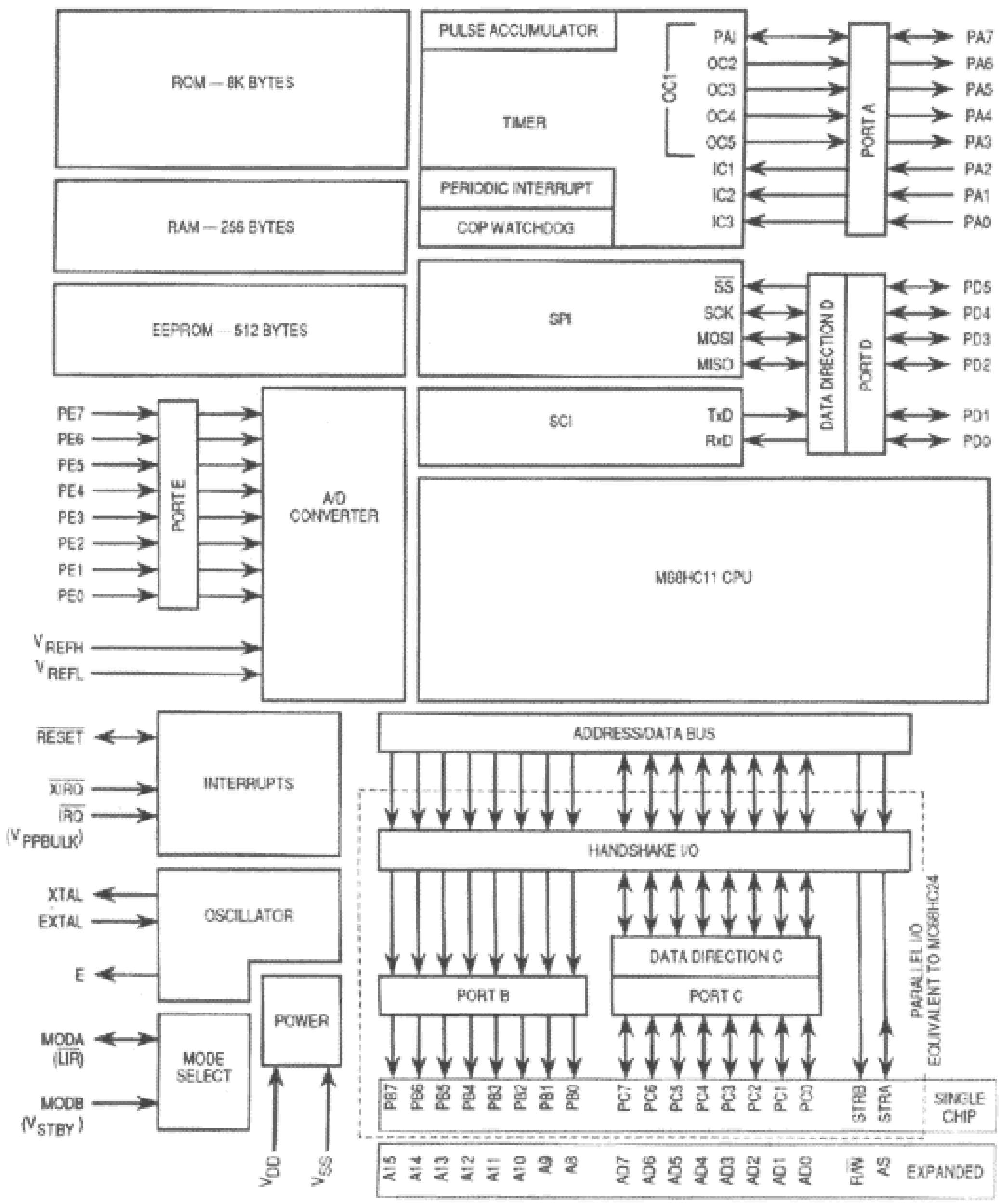
Appendix A

Signal Processing Unit



Appendix B

MC68HC11 Achitecture



References

[1] Mark Products Corp, *LMKB3R025A2 Liquid Crystal Display Data Sheet*
[2] Motorola Corp, *MC68HC11 Handbook*