SCHOOL OF ELECTRICAL ENGINEERING AND TELECOMMUNICATIONS

ELEC3117 Final Report
Auto-Tuner

The Automated Guitar Tuner Project

by

Ashutosh Kumar - z5260203

Bronston Ashford - z5146619

Submitted: August 11, 2022

Executive Summary

The Auto-Tuner is a handheld, battery-powered automatic guitar tuner. The aim of Auto-Tuner is to take the hassle out of tuning while providing an array of useful features aimed at beginner musicians. Since a guitar is a wooden-stringed instrument, it requires frequent tuning. This is often a difficult task for beginners and a necessary pain for professionals. Auto-Tuner aims to remove this burden, allowing guitarists to go back to what is really important to them—playing—but also supporting them in improving their skills where possible. Currently, there are no automatic tuners which have features focused on assisting players in their musical development like the Auto-Tuner does.

According to market research, the guitar is a rapidly growing industry, with an unprecedented 15 million sales worldwide in 2019.With rapidly growing sales, there has been an influx of beginner musicians with a trend towards self-guided learning through digital means. Primary market research uncovered that guitarists are highly interested in gadgets and that they showed an interest in an automatic tuner for less than $100. To cater to this self-teaching market, Auto-Tuner has designed its features around building musicianship and ease of use. The features include:

- Built in pulse metronome - for developing musicians timing.
- Tone generator - to assist ear training and manual tuning.
- Self-orientating display - to ensure display is upright for all types of guitar tuning peg positions.
- Less than $100 price point.

An Auto-Tuner prototype has been successfully developed with most of its features working and tested. The automatic tuning is showing promise with its tuning accuracy falling within 6 cents of error and its pulse metronome with less than 0.5% error over its full range. However, more work is needed in bringing the tuning speed to below one minute.

The current automatic guitar tuner market has two main competitors—Roadie and Jowoom. It is estimated that the current units sold of these products are 26,700 units per quarter, of which Auto-tuner expects to take at least a 10% share. The expected cost of manufacturing per device is $51.18, which gives a 40% profit margin when sold for $85. With an initial development cost of $218,000, Auto-Tuner is expected to be profitable within 3.5 years according to NPV calculations with a 9% discount rate. Furthermore, a 5-year business plan is laid out to guide the Auto-Tuner to a 10% market share.

A 5-year development plan has been outlined of how the prototype will be re-designed and improved to enter the market competitively. The plan goes through short-term and long-term visions and how to maintain consistent revision to ensure Auto-Tuner is always ahead of its competitors.

Contents

List of Figures

Introduction

Learning music is a valuable skill, and millions of people around the world attempt to learn it every year. Many musicians begin their musical careers with a guitar, one of the most accessible and popular instruments. However, a guitar requires a lot of regular tuning and upkeep which can be frustrating for the novice and an annoyance for the professional. Furthermore, learning music often requires assistance from other tools such as metronomes and tone generators. Yet, carrying around all these devices brings its own burden.

This is where the Auto-Tuner comes in. A tuning device aimed at guitarists who want the burden of constant tuning lifted and who utilise tools to guide their self learning. How does it work? Once you place the Auto-Tuner on the guitar tuning peg, it will do all the hard work for you. Now the novice no longer has to deal with the frustration of finding the correct pitch, and now the professional can focus on better things while the guitar tunes itself. Once the tuning is done, it's not over. When you need to practise your timing or perfect your pitch, the Auto-Tuner has got you covered with its musician development tools such as the pulse-metronome, so you can feel the rhythm, and its tone generator. Currently, no other device on the market is aimed at providing the array of musician development tools that the Auto-Tuner does.

This report contains in-depth detail about the research that was undertaken on the market and components available, the designing and construction process, and the business/ development plan to send it out into the market successfully.

The aim of this report is to answer the following questions:

- Why is Auto-Tuner a product that will be desirable to consumers?
- How was Auto-Tuner constructed?
- What are the market and financial prospects for Auto-Tuner?
- What are the development plans that are associated with a successful launch?
- What are the risks and issues associated with Auto-Tuner?

2.1 Problem Statement

Guitars require frequent tuning due to the nature of them being wooden stringed instruments. Tuning a guitar is often difficult for beginners or a necessary pain for professionals. Removing this burden will allow guitarists to back to what's important. Playing.

Project Description

3.1 Needs Assessment

The Auto-Tuner features many different attributes which make it desirable in the current market. However, this is a dynamic section that will require constant revision and adapting to suit the market's needs and wants. Doing a thorough needs assessment will allow Auto-Tuner to enter the market competitively and adapt well to the changing market.

3.1.1 Who is the target market?

Research determined that the market is more inclined towards beginners. This is based on secondary market research which uncovered a huge emerging market of amateur/hobbyist guitarists, specifically acoustics, due to their lower entry price. It was also discovered that there is a growing trend among guitarists and self-motivated learners who are turning to online resources to learn from.

All these considerations bring forth a new market that Auto-Tuner can capture, which is the beginner market with a lower budget but a desire to learn music.

3.1.2 Why do they need it?

Novice guitarists have been turning to online resources for self-guided learning. Face-to-face classes have been on a steady decline. This has resulted in large populations of beginner guitarists who require electronic assistance in maintaining their instruments and aiding their practice. There are currently many gadgets that are used to assist in guitar practise and maintenance, such as tuners, metronomes, and string-changing tools. However, the current market offers very few devices that house all of these in one product. The devices that do exist lack features such as a self-orientating-screen that make them uncomfortable to use at times or tone generators, which are important when tuning in a group setting or improving tone recognition.

3.1.3 What do they need?

The Auto-Tuner needs to be simple, easy to use for any person, and be able to accomplish its tasks with a high degree of precision, both efficiently and quickly. The tuner should also account for different tuning peg sizes between different types of guitars. The device will need to be lightweight and portable. Being a complimentary device to a guitar, it should be able to fit inside a guitar case. It should also have a long battery life and be ergonomic—fitting comfortably in most hands and usable by both right-handed and left-handed people.

3.1.4 Where and how will it be used?

As a result of the Auto-Tuners' portability, it is expected that the device can be carried anywhere. However, it is predicted that the primary use will be at home where beginners practice; there are no direct restrictions on where and when it can be used. Since the tuner will use vibration to detect pitch, the Auto-Tuner is not restricted to quiet areas and can be used in any environment without a decrease in accuracy. The user will have the option to select the modes, and once attached to the tuning pegs, the device will utilise its bi-directional motor to rotate the tuning pegs to achieve perfect pitch.

3.1.5 What features do they need in it?

The features that the Auto-Tuner will possess will make the customer's experience in tuning and learning more smoother. They are summarised as below:

- Self-Orienting Screen - A screen that will always be upright regardless of how the device is held.
- Users choice - with the addition of buttons and switches, the user can then have the experience of choosing between the different modes and features that Auto-Tuner possesses.
- Tone generator - a tone at a specific frequency can be produced from a speaker, and this allows the user to practice tone awareness/ recognition as well as tune in a group setting.
- Metronome - a software-developed timing system which will allow the user to improve their timing.
- Vibration system - this will go in conjunction with the device, the user will experience a vibration informing them that a tune or a beat has been completed.
- Multi-purpose peg slots - to allow for different size pegs to fit. This is necessary since not all guitars are the same and each possess it's own unique variation in pegs.
- Long Battery life - It would be inconvenient for constant recharging for tuning, the battery life should be large enough to tune multiple times and last at least a few hours of continuous usage.

3.1.6 Issues and limitations with existing products?

User reviews for the current devices on the market complain of poor readability of the display, either due to the orientation of the screen being upside down in some positions or the user's hand covering it. In addition, users complained of a lack of ingenuity. They desired products that could offer a bit more than simple tuning at that price point, which currently no other automatic tuner offers. This last point is especially true in the modern market, as more customers are using online features and self-paced learning. They require devices to support their learning process alongside all else. Lastly, primary research shows that most people are only willing to pay below $100, whereas the cheapest automatic tuner on the market is $144. Particularly for a novice player, this is a large expense as many beginner guitars are in that price range themselves.

3.2 Unique Selling Points

Auto-Tuner has many unique aspects that the competitors have not incorporated into their products. Using feedback to improve and innovate the software and hardware makes Auto-Tuner superior to what is currently on the market.

- Self-orienting screen - It ensures that regardless of the build of the guitar, the screen is always upright. Some guitars have all the pegs on one side whilst others have it on both, so to ensure that regardless of the guitar's shape, the screen is always upright and comfortable for the user to use.

- Tone generator - A reference pitch allows the device to assist with tuning any instrument or to practice ear training. Often, guitarists are around other musicians and having a tone generator built-in allows users to become more familiar with tone and pitch.

- Metronome - A software-developed timing system will allow the user to improve their timing by having a system that will properly keep time.

- Haptic Systems - Once the tuner is clamped to the head stock of the guitar, a vibration will then be felt directly on the players' fingers. This will alert them and keep them mindful of timing and control when playing their instruments. This is a feature that acts together with the system to inform the user about the completion of tasks or event.

- Lower costs - A major downfall of other products is their high cost, which is not affordable for an average guitarist. To make it beginner-friendly, we will design the device to be under $90 which now brings the price down closer to the higher end of the non-automated tuner price range.

The addition of these extra features it will improve the experience of the customer and differentiate Auto-Tuner from other products that are available on the market.

3.3 Product Concept

Auto-Tuner is a complimentary device to support beginners during their learning experience. In its base state, the Auto-Tuner will be automatically capable of tuning a stringed instrument simply by using vibrations produced from the string. This adds convenience for beginners, who can often find this task arduous and challenging. But beyond this is what makes Auto-Tuner unique, and it's the learning capabilities it provides to the users in the form of a metronome and tone generation. The Auto-Tuner teaches two key skills which are necessary for music: timing and tone or pitch recognition. The difference between Auto-Tuner and other tuners is that Auto-Tuner will provide the user with an educational experience on a well-tuned instrument, whereas others will simply provide the tuned instrument.

3.4 Requirements Analysis

The requirements are a vital aspect of the design; they acknowledge what is required of the product at the bare minimum to make it a desirable and viable product on the market. The requirement analysis requires understanding what the device needs to accomplish by the end of its design.

- The device should be responsive, capable of reacting to the user's instructions in under 1 second.
- The unit should be compact and easily handheld. It should be no bigger than 10 cm x 5 cm x 3 cm.
- The user should be easily able to understand the menu and navigate through it. To be considered easy to use, the user should be able to pick up the device and be able to use it within 5 minutes of first using it.
- The system should be capable of tuning to the correct pitch within 6 cents.
- The system should cost no more than $90.
- Have a self-orienting screen that is always upright.
- The Auto-Tuner should be able to fully tune a guitar in one minute.
- The device should have a metronome range from 20 BPM to 200 BPM with a timing accuracy within 1% error.
- The system should have a tone generator using a speaker or buzzer to generate tone within 6 cents accuracy.
- The device should be able to last at least 2 hours of continuous usage.

3.5 Design Concept

The Auto-Tuner will have many features and different components that each come together to complete tasks and requirements. The technical concept of how components will interact within subsystems and as a holistic system will be explained below.
The key subsystems that makeup Auto-Tuner are:

- Signal Input and amplification circuit.
- Tuning motor .
- User Interface.
- Output tones and haptics.
- Micro-controller.

- Power system.

The signal input to the system will be via vibration sensors, which will produce low millivolt signals at the output. This small signal is then amplified through an amplification circuit, and a DC bias is added to adjust the signal to the correct voltage range for the MCU's ADC (analog-digital converter).

The MCU will use the input signal and run it through an algorithm that will determine the frequency of the signal and produce an output signal that will pass through a digital to analogue converter (DAC) and to a bi-directional motor that will rotate the tuning pegs. This will also be connected in a feedback loop, which will give feedback to the MCU, which will keep running through the algorithm and adjust the fine-tuning, ensuring perfection.

The user interface revolves around the OLED screen and the buttons, which the user can use to interact with the system. The OLED screen will display the menu and information based on what the user is trying to perform. The screen is also synced with a gyroscope, which will give information on the orientation of the system and allow the screen to make adjustments as necessary. Alongside the display, buttons and switches are also present, which will allow the user to shift through the menu and pick the choice of features that are available.

The output tones and haptics revolve around the tone generator and timing feature. The tone generator is created from the MCU and output through a speaker, and the user is capable of changing the pitch. Similarly, the metronome will be a software-developed timekeeper which will produce a vibration using a vibration motor to alert the user that the timing interval event has occurred. This is significant for improving the learning experience and a key skill in music, which is tone recognition and timing.

The microcontroller is the central component of the system. It is the component that analyses the frequency and produces signals to rotate the motor, which will tune the instrument. It also gives instructions to the display based on any instructions given by the user in regards to button presses and switches. It will also produce a tone and timing that the user will be able to control. The incorporation of all of these key learning features into a single device will allow novice players to improve their skills without having to spend large sums of money on additional accessories.

An idealistic concept of what the model would look like to the user is given in 3.1.

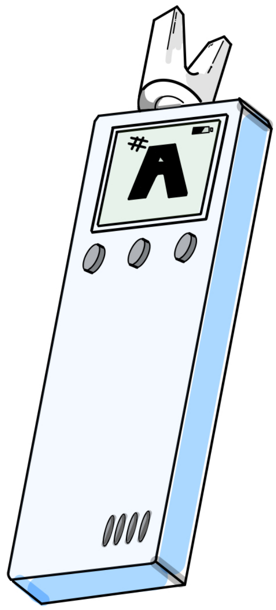


Figure 3.1: *Auto-Tuner conceptualised.*

3.6 High Level Design

The list of requirements and functions can be translated into subsystems of how different components will form a system and how each subsystem will interact with each other.

- Vibration Sensor - Picks up the vibrations of the tuning pegs after a string is strung.
- Amplification - Amplifies the small signal to the range of the MCU ADC.
- MCU - Is the brain of the operation and does the logical operations of all other sub-systems.
- Orientation sensor - Detects the orientation of the device.
- Push buttons - A user-oriented component to allow the user to control their device.
- Screen - Display the menu and important information that users would want to know.
- Tone generation - Plays the selected pitch .
- Metronome - Is a time-keeping function.
- Power source - Most likely batteries which will provide power to the complete system.

- Voltage regulation - Converts the high voltage to more acceptable voltage for the different components.
- Tuning peg control - Component which is connected to the tuning peg and rotates it accordingly.

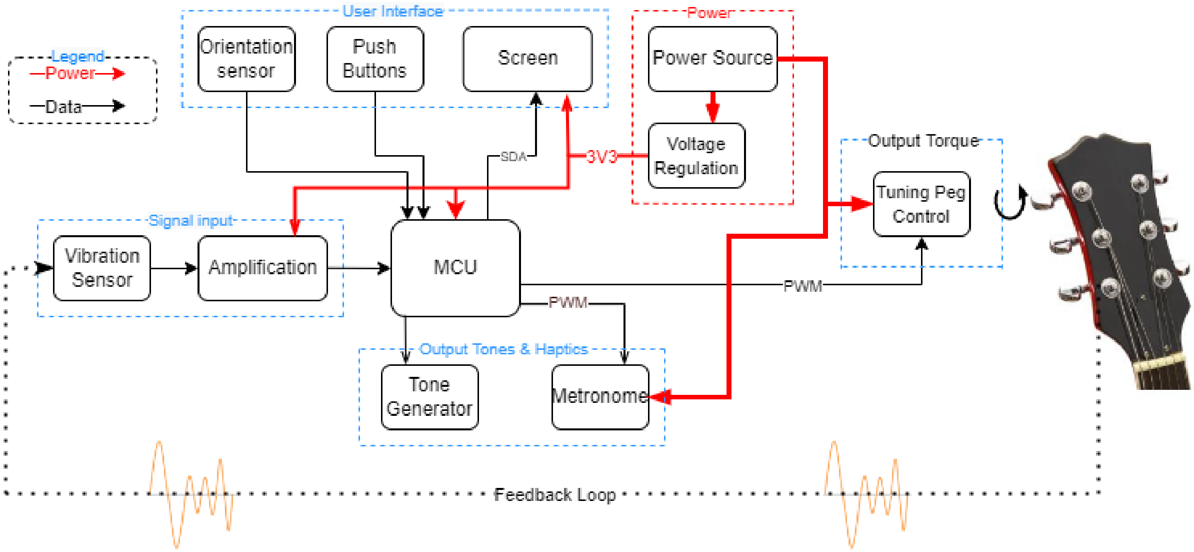


Figure 3.2: *High Block Level Diagram.*

3.7 Technical Components

There were a few changes that were made to our materials from when the design proposal was proposed. Most components were kept consistent as the majority of the design was already completed by then. However, issues that arose when trying to interface and recognise the limitations of some components caused us to change to better materials.

3.7.1 Components

STM32L432CBTx

The core of the Auto-Tuner is from the STM32L4 series MCU. The MCU will be responsible for controlling the sub-systems, performing digital signal processing (DSP) algorithms on the vibration signal, and controlling the state machine. The STM32 was specifically chosen primarily due to its low power consumption and ARM Cortex-M4-based design, as this would then translate to increased battery life. Furthermore, the M4 processor comes with a built-in DSP instruction set, making it capable of processing audio signals in real-time. The MCU also has the added benefit of having a small footprint with only 48 pins. This allows for a compact design.

Piezo Transducer

The vibration sensor used will be a piezo transducer, which converts the vibrations into small electric signals in the range of 10 mV to 100 mV. A piezo is ideal because it is a passive element that does not draw power, which helps to reduce battery consumption. To add to that, they are also very cheap whilst still being quite accurate.

Rail-to-Rail Low Voltage Op-amp (LMC864)

The op-amp is required to amplify the low-voltage vibration signal of the piezo transducer. The LMC864 was chosen for its low voltage operation and rail-to-rail capabilities. The output of the op-amp needs to be between 0 and 3.3V in order to interface with the ADC of the MCU. The rail-to-rail feature allows the full use of this voltage range. Furthermore, the LMC864 is low-cost, which helps to achieve the desired price point for the Auto-Tuner.

Tuning motor (SG5010)

The servo motor is what provides the torque for turning the guitar's tuning pegs until the desired pitch is achieved. The SG5010 was chosen as it is an affordable continuous servo with sufficient torque. Being a continuous servo greatly simplifies the design due to H-bridge circuitry being provided within the servo. Furthermore, the SG5010 is the most stocked component in its category, meaning fewer chances for delays in delivery.

OLED Screen (SSD-1306)

The SSD-1306 is a common and inexpensive display. It has built-in screen mirroring capabilities, which is critical for our self-orientating display feature. Furthermore, it is low cost, has a comprehensive data sheet, and is easy to interface with.

Gyroscope (MPU-6050)

The gyroscope is used to detect the orientation of the Auto-Tuner relative to the users' viewing position. The MCU will use the detected orientation to flip the OLED display such that the screen is always viewable. The MPU-6050 was chosen as it is well documented with 6-axis measurements. Another key factor that made this more appealing was its availability on the market. There were only a few locally sourced gyroscopes available in Australia, which were still low cost and had fast delivery compared to similar products available from China.

Vibrating mini motor disc (10B27.3018)

The vibration motor is used for the pulse metronome. The metronome's output will be pulses of vibration at the desired beats per minute. The user should feel the vibration while the Auto-Tuner is clamped to the guitar's headstock. The 10B27.3018 is used due to its small size yet powerful vibration. It produces a sharp vibration with low latency, which is desirable for a time-keeping device. The benefit of its small size is its low current requirements, which aid in improving battery life. Moreover, it is widely available and inexpensive.

Push buttons and slide switches

Because of their simplicity and ubiquity, low-cost push buttons and slide switches are used for the user interface with the Auto-Tuner. Furthermore, there are many online resources available that assist in implementation.

3.7.2 Updated Components

Low Drop Out Linear Regulator - AP1117

The reason this was changed was that the linear regulator has a far smaller footprint compared to the buck converter, which meant that space could be further saved and it was capable of converting the 6V input from the batteries into the 3.3V required by the other components to operate. This had a significant impact in that it saved a large chunk of space in the overall build; it eliminated an extra point of connection when all the components were being soldered onto a single PCB; and it made the device easier to fit into the case.

AA Battery (Instead of Li-Ion Batter cells)

This choice was made at the last minute to replace the Li-Ion batteries with AA batteries. This was as AA batteries are readily available and cheaper. Housing is also made and sold at local retailers such as Jaycar, which saves money by not creating a new one. The major disadvantage here was that Li-ion batteries are far smaller, and by switching to AA batteries we significantly increased the size. However, the key reason for doing this was that AA batteries are capable of providing far more amperage per hour compared to a Li-ion at that price point. This then allowed the device to be operational well beyond the requirements that were set in regards to the battery life.

Speaker Module - XC3744

The speaker was used instead of a simple buzzer as it offers a higher quality sound and is multi-toned. This was especially important when compared to the initial buzzer that was going to be used. It was very monotone and would not produce a good sound. Furthermore, the module also offered an on-board potentiometer that allowed for volume control, which simplified the building process for the prototype.

Updates to Material from Development Proposal

Table 4.1: *Updates in Materials from Development proposal.*

Section Updated	**How and Why**
Problem Statement	This section required an update based on our previous statement, which pointed the problem towards external issues as opposed to problems that guitarists had with tuning itself. This was updated to be more guitarist oriented and how Auto-Tuner solves it.
Market Research	This was updates to compass a better comparison of the competitors. Previously, the market analysis was weak and didn't provide a thorough investigation of what the competitors offered and how Auto-Tuner could differentiate itself from them.
Technical Approach	A significant change that was made was the use of a state machine to determine the tuning motor control instead of a PID controller. This was done in an on-off sequence with the detection and motor running in intervals one at a time. This was necessary since the motor added additional vibration, which distorted the input signal, and the PID controller wasn't capable of taking care of mitigating this.
Components	Some changes to components were made, such as the voltage regulator, speaker, and power source were made. This was done for reasons such as smaller footprint, better quality, and availability. This is explained in more detail in section 3.7.2.
Requirements analysis	This was revised to be more reflective of what was noted in the updated market research. It also used the feedback given and added additional key requirements which were missing, such as user experience.

Market Analysis

5.1 Market Research

5.1.1 Primary Market Research

Primary market research was done through an online survey asking nine questions aimed at determining the level of interest from potential customers (see appendix B). The survey collected 30 responses from guitarists ranging from beginners to professionals.
The key findings are as follows:

- The results correlated with secondary research that found the largest group among beginners, intermediates, and professionals were beginners, making up 46.7%.
- Tuning a guitar was an annoyance for 26.7% of guitarists while 56.7% felt neutral towards it.
- More than 76% of guitarists buy guitar-related accessories.
- Eighty percent of guitarists use a metronome to assist learning.
- Over 50% of all guitarists were interested in an automatic tuner. Where 64% of beginners were interested in an automatic tuner.
- The majority of guitarists are willing to pay between $40 and $107.

Refer to Appendix 10.2 for the complete survey results.

5.1.2 Secondary Market Research

The automatic guitar tuner is a niche market, as currently only two products exist which are Roadie and Jawoon. This makes it difficult to analyse the automatic tuner market directly in great depth. For this reason, this analysis will include research on the guitar market in general to help estimate the size of the automatic tuner market.

Guitar Market Valuation

Guitar sales have been on a steady increase since the Great Financial Crisis (GFC) in 2009, going from a global valuation of $821 million to $8 billion in 2019. Furthermore, likely as a response to COVID-19 lock-downs, 2020 guitar sales peaked at $9.2 billion, a record 15% increase in one year. In terms of actual guitar sales, this is about 15 million guitars sold globally [1].

A breakdown of the market in terms of acoustic and electric guitar sales is demonstrated in figure 5.1. Acoustic sales have outpaced electric sales since the GFC in 2009 (In the US). Likely due to the lower entry cost as amplification isn't needed. However, the difference has been reducing since 2019 [2]. Furthermore, the global electric guitar market is expected to reach 6.4 billion by 2028 and so it is still a significant market to consider when designing a guitar related device.

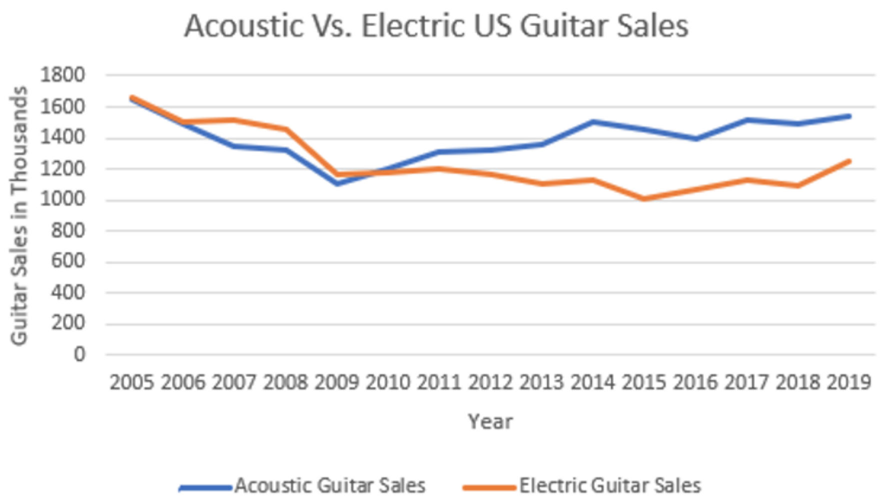


Figure 5.1: *Acoustic vs Electric sales in the US.*

Automatic Tuner Market

There are currently two main products in the automated guitar tuner market. The Roadie series by Band Industries and the Smart Winder by JOWOOM. The Roadie 3 (latest in the series) retails on amazon for $179 and the Smart Winder for $139. For comparison, non-automatic tuners on amazon mostly fall in the $25 to $50 range.

On Amazon.com the Roadie 3 and Roadie 2 hold a total of 1,500 customer reviews. The highest reviewed guitar tuner is a $30 non-automated tuner and has 18,000 reviews for comparison. This makes the Roadie 8.3% as popular as the most popular tuner while also being in a much more premium category [3]. The JOWOOM smart winder is much less well known with less than 300 reviews and is a Chinese-produced product found on Ali-Express. This suggests that Roadie holds 80% of the automated tuner market share.

Customer Profile

The following research was done to determine who the main type of customer is. It is found that in the last couple of years, unprecedented numbers of people have turned to self-motivated learning through online resources. In 2021, Fender Musical Instruments conducted a study which found that, in the last two years, a whopping 7% of the US population (16 million people in the age range between

13 and 64) had started learning guitar [4]. Furthermore, in 2020 Fender released an online app for learning guitar which became highly successful. When they offered a 3-month free trial to the first 100,000 people, it was filled within 24 hours. In response, they increased the number of spots and they topped out with 930,000 users two months later. Other online platforms, such as Gibson App and Yousician, reported similar boosts in subscribers [1]. This demonstrates a large demographic of beginner guitarists who are turning to tools to help teach themselves to play guitar. Therefore, there is a large market for devices with features aimed at self learning.

Competitor Analysis

The two competing products on the market are the Roadie 3 by Band Industries and the JOWBOOM Smart Winder, featured in the image below. Both devices operate similarly. They are small, battery operated, handheld devices which tune one string at a time. The Roadie 3 has a more premium design with more features where the Smart Winder has a lower price point with less features. A side by side comparison of their features in listed in table 5.1.

Figure 5.2: *Left: Smart Winder. Right: Roadie 3.*

Table 5.1: *Comparison of competitors features*

Feature	**Roadie 3**	**SmartWinder**
Price	$179	$139
Automatic Tuning	✓	✓
Acoustic & Electric compatible	✓	✓
Metronome	✓	✗
Tone Generator	✗	✗
USB Rechargeable	✓	✓
Hours of battery life	✓	✓
Customisable Tuning	✓	✗

5.2 Projected Sales

An estimate of expected sales is done using the break down approach. On Amazon, other devices in the automated tuner category have on average a total of 204 reviews per quarter (see appendix A). The average Amazon buyer review rate ranges from 2-10%. Using the conservative figure of 10%, this means approximately 2,040 units are sold on Amazon per quarter. In 2021, Amazon accounted for 40% of all ecommerce sales, and ecommerce accounted for 19.1% of all sales [5][6]. Factoring these in, the total units sold per quarter world wide can be found as:

$$\frac{2040}{(0.191)(0.4)} = 26,700\,units.$$

Since this market has only two competing products, we can assume a third product would be able to penetrate at least 10%. This gives a conservative estimate of 2,700 sales per quarter at maturity.

5.3 Possible Selling Price

When selecting the price point of the Auto-Tuner three factors were considered. The price of the competitors, the price of components, and the price customers were willing to pay.

1. The cheapest automatic tuner on the market is the Smart Winder at $139.
2. In section 7.2.1, the manufacturing costs of the Auto-Tuner are estimated in the range of $50.
3. Primary market research found that most customers wanted to pay less than $107.

With these considerations, the selling price of $85 AUD was chosen as the best balance between profit margin and remaining significantly under the competitors' cheapest price while staying within the customers' range.

Detailed Design

6.1 Principle of Operation

The Auto-Tuner is a handheld device for automatically tuning guitar strings one at a time. The working prototype is featured in the image below. The device has two main modes of operation: tuning mode and feature mode. After turning on the device, the user can use a switch to select the desired mode. The principle of operation for each mode is as follows.

Figure 6.1: *Auto-Tuner prototype.*

Tuning Mode

Firstly, the user toggles the selection switch on the case of the Auto-Tuner to tuning mode. The Auto-Tuner displays that it is in tuning mode and then proceeds to display the tuning screen. The user will then hold their guitar as they normally would when they play. With one hand, the user will place the Auto-Tuner's motor-peg attachment over the tuning peg of the low E string of the guitar (see figure 6.2).

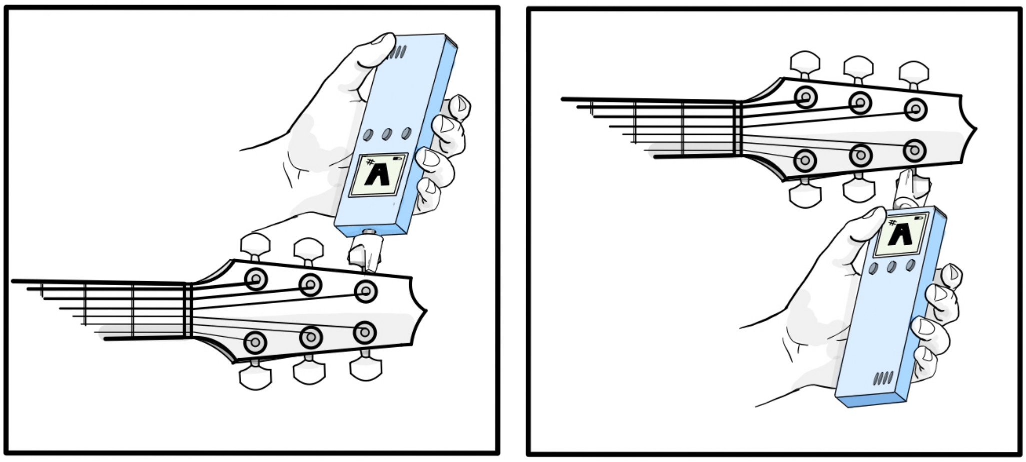


Figure 6.2: *The Auto-Tuner in tuning mode.*

The guitar's low E string is then strummed with the free hand. Once the Auto-Tuner detects the string is being played, it will begin the automatic tuning process. As the string is strummed, the Auto-Tuner will feel the vibrations of the string through the contact it makes with the guitar's tuning peg. As the Auto-Tuner adjusts the guitar strings, an indicator on the display will be positioned relative to the centre of the screen to indicate how far out of tune the current pitch is (see figure 6.3).

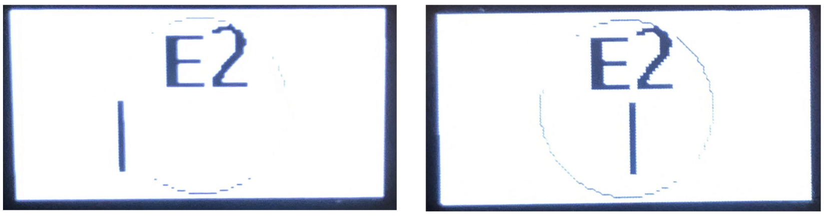


Figure 6.3: *Tuning screen for low E string. Indicator showing current state of strings pitch.*

The string will need to periodically strum until the Auto-Tuner vibrates, indicating that the string is in tune. The next string, A, is then loaded into the Auto-Tuners pitch detection algorithm automatically and the process is repeated. The Auto-Tuner tunes the strings in order: E,A,D,G,B, E, and once completed, starts the cycle over again until the user powers off or selects a different mode. In addition, if the users' tuning pegs are placed on the underside of the guitar, the screen of the display will self-orientate so that it is always visible from any tuning position.

Feature Mode

The feature mode has two options: metronome and tone generator. When the switch is positioned in feature mode, the display provides a menu with two options (see figure 6.4). When the user selects *metronome* they will have the option to select the desired beats per minute (BPM). Once selected, the Auto-Tuner will vibrate in short pulses according to the selected BPM. The user then places the device on their lap or in their pocket and practises the guitar while they feel the pulse of the metronome. If the user chooses the *tone* option, they can then select a desired pitch from the menu. The pitch is then produced through a small speaker inside the Auto-Tuner. The user and other nearby musicians can then use the tone as a reference pitch to tune in group settings or improve tone recognition.

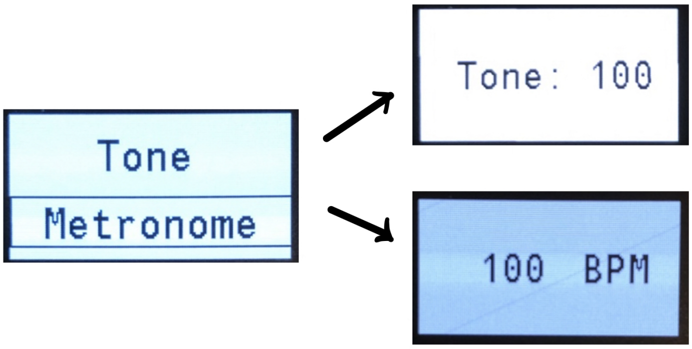


Figure 6.4: *Feature mode menu.*

6.2 Sub-System Design

Auto-Tuner contains a total of six sub-systems:

- User-Interface.
- Vibration Sensing and Amplification.
- Tuning Motor.
- Output Tones & haptic.
- Control System - Hardware and Software.
- Power.

The different sub-systems can also be seen highlighted in the high block level diagram 3.2.

6.2.1 User Interface

The user-interface sub-system consists of the components which are directly interacting with the user. Included in this sub-system are the OLED screen, push buttons, switches, and the gyroscope. Each one of these components interacts with the user, takes the instructions input by the user and informs the MCU to undertake that action.

OLED Screen

The OLED screen used is the SSD1306 series. A detailed reason for this choice is given in section 3.7. In short, it was due to its popularity, comprehensive data sheet, low cost, and built-in screen flipping capability. The benefits of using an OLED instead of an LCD were obvious since the OLED has far greater control over the display with the use of bitmaps. LCD offers a set character library which is limited and not ideal for this project.

The OLED screen displays the screen that the user is currently on and the options that the user can select. It is operated using an I2C pin which takes in inputs from the SDA and SCL pins and operates using a 3.3V power supply. The screen displays a variety of different screens based on the mode and selection choices depending on the user. At start-up, the screen displays the name of the product ”Auto-tuner”, and based on the orientation of the mode selection switch - goes into either tuning mode or feature mode.

When in tuning mode, the screen displays the name of the string that is being tuned and this ranges between E,A,D,G,B,E and below it displays a bar that moves closer to the center as the tuning reaches completion.

When in feature mode, it displays a menu between tone or timing. If the tone is selected, this gives the user a screen where they can select the tone that they want to play. Similarly, if the timing screen is selected, then the screen will display another screen which will show the BPM. The screen is a critical component that connects the user and the system and gives the customer control over their device. The process of selection can be seen in figures 6.3 and 6.4.

Push Buttons

To navigate between the different options available to the user, they use push buttons. These work by having two connections, one to VCC and one to ground, whilst the third is connected to the MCU. Once the button is pressed, the signal output is high as the path to VCC is closed, and this is detected by the MCU, which lets it know that a specific button has been pressed. It is a simple component that is critical in allowing users controllability over their devices.

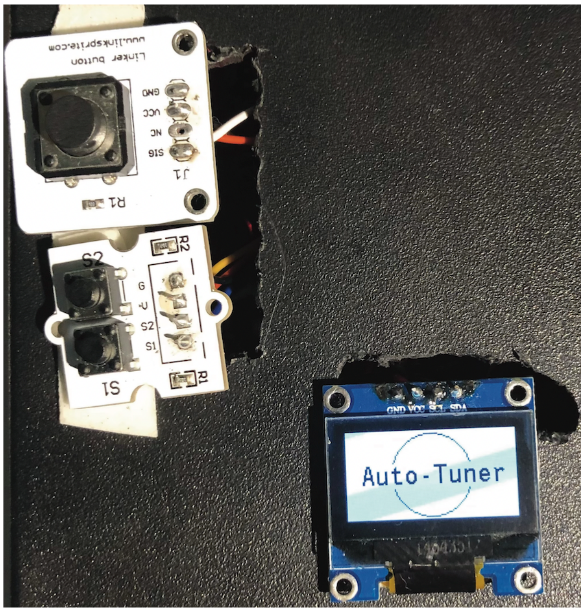


Figure 6.5: *Buttons and OLED screen.*

Switches

Included in the build are also two switches that are used for mode selection and turning on/off the device. The switches are shown in figure 6.6. The bottom switch is the power switch. It is connected between the power and the MCU and, once switched on, allows power to pass through to the MCU and turn on the device. The top switch is a feature mode select switch. This switch allows the user to select between the different modes that Auto-Tuner offers, which are the ”tuning” mode and the ”feature” mode. This switch works by using an interrupt which is triggered at both the rising edge and the falling edge. Based on the direction that the switch edge is, the MCU decodes it and switches to the right mode as soon as the switch is flicked.

Figure 6.6: *Switches (Top - selection, Bottom - Power.)*

Gyroscope

The final component, which indirectly interacts with the user and is also a key feature of Auto-Tuner is the gyroscope. The component used was the MPU6050. A detailed description is provided in the section 3.7. Its main purpose is to determine the orientation of the device and inform the MCU, which then determines whether the screen needs to be flipped or not. The gyroscope is connected to the MCU through an I2C pin, utilising the SDA and SCL pins, and is powered by a 3.3V source.

The gyroscope outputs two sets of data. Acceleration data, measured through the change in velocity over time ($\frac{m}{s^2}$), and change in angle measured over time ($\frac{\circ}{s}$). To determine these, the gyroscope uses row, pitch, and yaw, which are the measures of the rotations along the x, y, and z axes [7]. Using the relationship between row, pitch, and yaw and the angles, a relationship is derived which is known as "*Euler's Angles*". By determining the change in acceleration and finding the angles between the changes, the rotation along an axis can be found and this is simplified in the formula's below [8].

$$AngleX = tan^{-1}\left(\frac{Acceleration - Y}{Acceleration - Z}\right) * \frac{180}{\pi}$$

$$AngleY = tan^{-1}\left(\frac{Acceleration - X}{\sqrt{(Acceleration - Y)^2 + (Acceleration - Z)^2}}\right) * \frac{180}{\pi}$$

.

This equation was tested thoroughly in different scenarios, such as starting up at random orientation, so a reference cannot be set as well as under random motion. The results were promising and the device was capable of determining the degree of rotation and, once the angle reached greater than 90° the screen would flip.

It works through the use of a timing interrupt. This was a key challenge as the gyroscope could not be consistently polled as its sequence to read data is fairly long and disturbed other programs at the same time. Hence, in order to ensure that the gyro could be constantly communicating whilst not being computationally heavy on the MCU, a timing interrupt was used to overcome this challenge. The timing interrupt works by using a timer. Once the counter set by the timer is reached, the program starts an interrupt event for the gyro in which it reads the data from the gyroscope and if the orientation has changed, the screen is flipped; otherwise, it does nothing and the program is returned to where it left off.

The interrupt solution solved several issues, such as making sure the orientation data was consistently being read and used, making the code more compact, and decreasing the computation time on the MCU without slowing down the performance.

6.2.2 Vibration Sensing and Amplification

The vibration detection is done by a piezo transducer fixed to the casing of the Auto-Tuner. As the guitar's string is played, the piezo will turn the vibrations into a small electrical signal. This signal is in the order of millivolts and needs to be magnified to approximately 3.3V, the maximum of the ADC's voltage range. Furthermore, since the ADC samples voltages between 0-3.3V a DC bias of 1.65V is required.

To achieve a large gain, a cascaded dual op-amp circuit is used (see figure 6.7). Each op-amp has a gain of 30, which becomes a gain of 900 when cascaded. The cascaded configuration was used to limit the amplification of each op-amp's input offset voltage while still providing high gain. The DC bias is derived by using a voltage divider on the ADC's reference voltage VDDA and applying it to the op-amp's non-inverting input. Using the same reference voltage as the ADC ensures that the amplification circuit output is always compatible.with the ADC. Furthermore, to keep the op-amp output bound to 0–3.3V, a single supply low voltage rail-to-rail op-amp is ideal. The op-amp TLV8541DBVR was chosen as it fits the criteria necessary for this operation and is available at a low cost.

Finally, at the output of the amplification circuit is a low-pass filter (LPF) with a cut-off frequency of 500 Hz. The cut-off frequency was chosen to be as low as possible while still capturing the full frequency range of the guitar. This is to limit the noise introduced by handling the device.

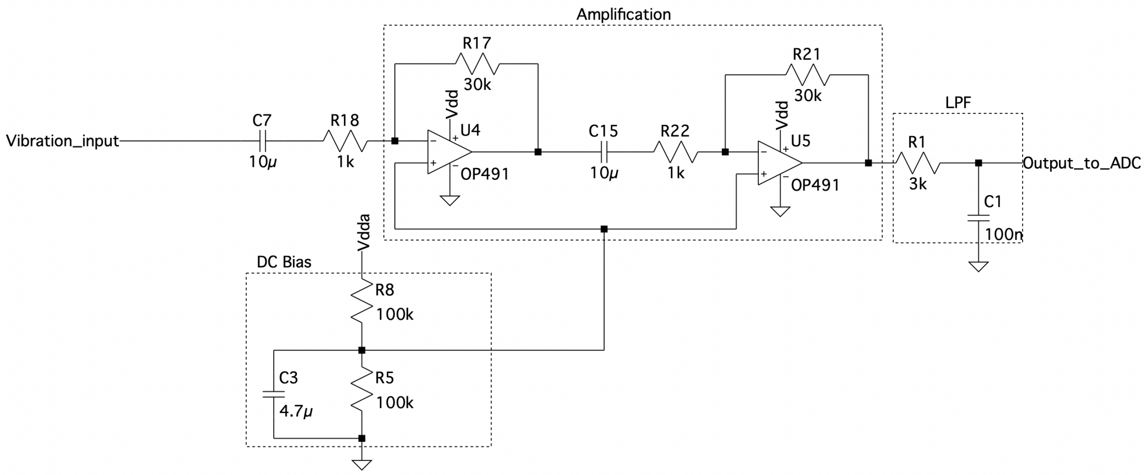


Figure 6.7: *Vibration amplification circuit.*

6.2.3 Tuning Motor

To control the tuning pegs of the guitar, a SG5010 continuous servo motor was chosen. The continuous servo offers simplicity as it only requires a control signal and power. Furthermore, the SG5010 was chosen for its low price while also delivering adequate torque.

The servo is connected to 6V, produced from the battery pack and ground, while its control signal is from the output of a PWM enabled pin of the MCU. In addition, to control the tuning pegs of the guitar, a fork-like attachment is fitted to the motor output so as to be able to grip the tuning pegs. For the prototype, 4 bolts were used with a plastic cover to give a tighter fit.

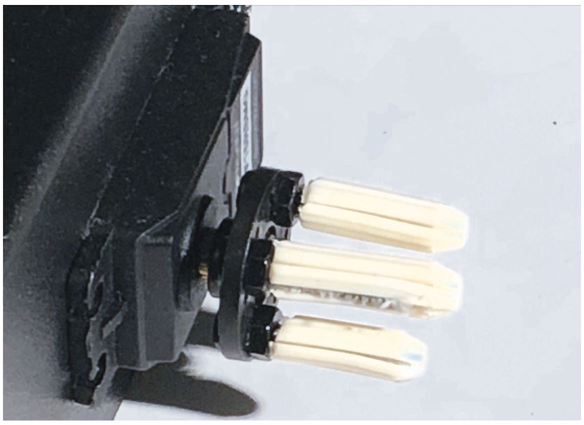


Figure 6.8: *Guitar tuning peg motor interface.*

6.2.4 Output Tones & Haptic Feedback

Tone Generation

The tone generator is used in conjunction with a speaker. The purpose of this feature is to imitate the sounds produced by a guitar and output them from the device. The aim here was to help beginners improve their tone recognition abilities as well as use it to potentially tune other instruments in a group setting. The output tones can be selected anywhere between 80 Hz and 330 Hz, which corresponds to the guitar frequency range.

To achieve this, an audio module was used, this module uses a 3.3V supply and the signal input is provided by the MCU, which generates the sine waves. The module offers a potentiometer that allows for volume control and uses an op-amp to boost the voltage going into the speaker to maximise the quality of the sound. The sine waves are generated using a timer that corresponds with the fundamental frequency of a tone and are then mathematically adjusted such that they also include a few of the key harmonics to make the output sound as realistic and close to a guitar sound as possible. This is then output from the speaker, which was chosen since it produces a cleaner and multitone sound compared to a buzzer, which has a monotone ”buzz” sound at different speeds. The tone generation screen menu is seen in 6.4. The module used is shown in the figure below.

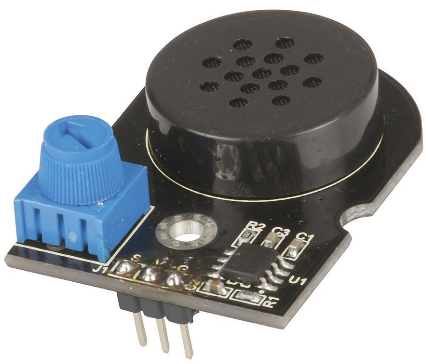


Figure 6.9: *Speaker Module.*

Metronome

The metronome is a timing device that uses a vibration motor, a timer, and an interrupt to indicate the end of a beat. The metronome is an additional feature that was added to Auto-Tuner to improve the timing of its users. It operates by breaking down a minute into beats with the help of a timer, and the end of the timer is felt by the user through a vibration of the box. The timing can be selected anywhere between 20 BPM and 200 BPM, which is the standard beat that music is played at.

The metronome operates using a timer, once the menu is selected the BPM that the OLED shows is converted into counters using the following formula:

$$Prescalar = Prescalar \times \frac{60}{BPM}$$

.

The timer uses the prescaler to call an interrupt similar to the one used by the gyroscope, and once the timer is triggered, it toggles the vibration motor for 200 ms. The vibration motor is attached to the case such that once it is on, the vibration can be easily felt by the user. This was also one of the main reasons why we chose a vibration motor as opposed to using sound to indicate the end of a beat, as it can then be used everywhere without the effect of external noise affecting the user experience. The motor also uses a boost circuit to directly source current from the power supply rather than the MCU and does this using a BJT, which turns on and off the motor by diverting the flow of current.

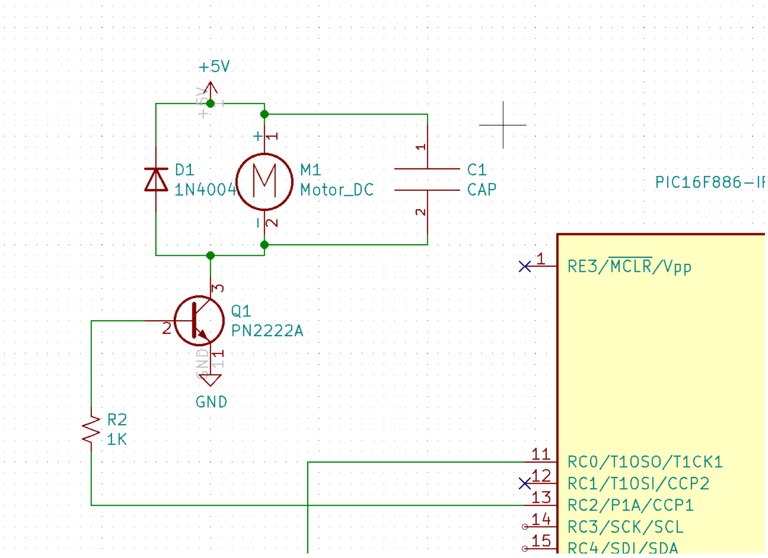


Figure 6.10: *Motor Amplifier circuit.*

6.2.5 Control System

Hardware

The Auto-Tuner's subsystems are all interfaced with and controlled by an MCU. The MCU needs to be powerful enough to process real-time vibration signals, and have enough pins to interface with all the subsystems, have an onboard ADC while being low-powered enough to operate on batteries. For this reason, an STM32L431CBT6 was chosen. It contains a cortex M4 CPU which has a DSP instruction set and floating point unit (FPU). Being an L4 series MCU also means it consumes low power and is ideal for battery-powered devices.

The MCU interfaces with the subsystems with the following ports (see figure 6.11).

- Vibration sensing circuit output input into ADC pin.
- OLED Display control through I2C1 pins.
- PWM output GPIO pin to tuning motor.
- Output tones signal generated at DAC pin.
- Button 1, dual button, controlled through GPIO pins and uses a GPIO pin set high for VCC.
- Button 2, single button, controlled through GPIO pin and uses another GPIO pin set high for VCC.
- Gyroscope output to I2C2 pins.
- Mode selection switch controlled with a GPIO pin and uses another GPIO pin set high for VCC.
- Debugging connection for code uploading through NRST, SWO, SWDIO, SWCLK.

In addition, to improve performance, the MCU was designed with an external crystal oscillator circuit and power isolation for the ADC reference voltage. The external crystal oscillator gives higher accuracy with the ADC sampling, which helps increase the accuracy of the tuning capabilities. A 24MHz crystal XRCGB24M000F0L00R0 was chosen for this task due to it being the fastest crystal compatible with the MCU and its low price.

For better ADC performance the ADC reference voltage is separated from the MCU's power source by a ferrite bead and coupling capacitors (see figure 6.12). The values for the components were chosen as they were recommended by the vendor.

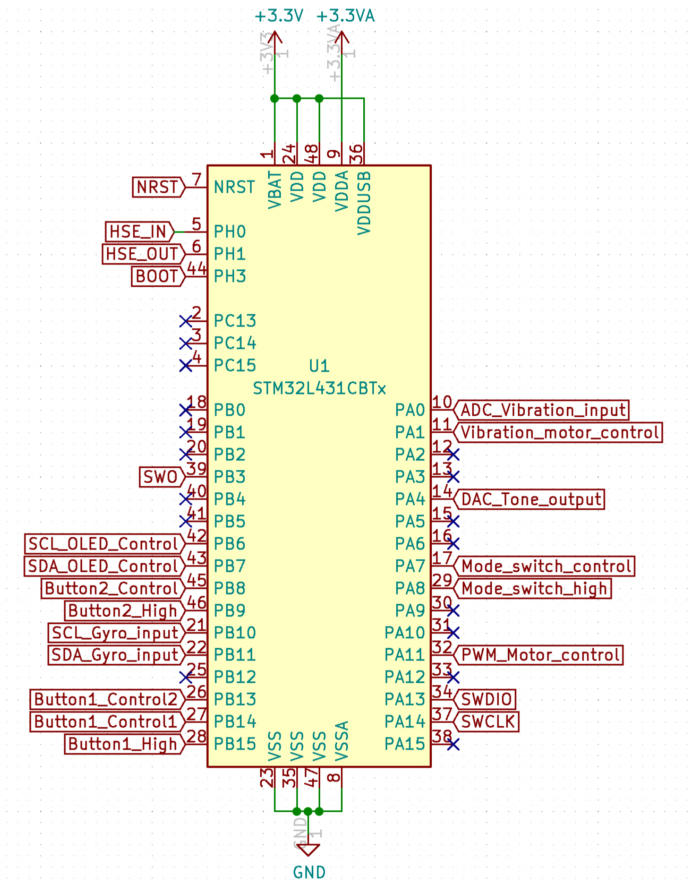


Figure 6.11: *MCU Connections.*

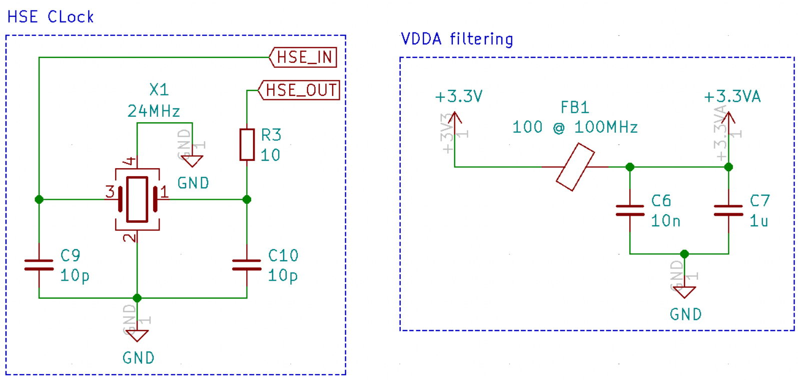


Figure 6.12: *Left: crystal oscillator circuit. Right: ADC reference voltage filtering.*

Software

Pitch Detection

To be able to automatically tune a guitar the MCU needs to be able to detect pitch from the vibration signal, calculate the error signal and then send the appropriate control signal to the tuning motor.

The pitch detection algorithm used is called McLeod's Pitch Method (MPM) from the paper *A smarter way to find pitch* [9]. The MPM has three phases: frequency detection, peak picking, and parabolic interpolation.

Firstly, the frequency detection done by using the square difference function below. Expanding the function shows that this is a summation of an auto correlation function and a sum of squares.

$$d(\tau) = \sum_{j=t}^{t+W-\tau} (x_j - x_{j+\tau})^2 = \sum_{j=t}^{t+W-\tau} x_j^2 + x_{j+\tau}^2 - 2x_j x_{j+\tau}$$

.

Dividing the equation by the sum of squares results in a normalised square difference function (NSDF).

$$n(\tau) = \sum_{j=t}^{t+W-\tau} \frac{2x_j x_{j+\tau}}{x_j^2 + x_{j+\tau}^2}$$

.

The output of the NSDF is similar to an autocorrelation function except the values are bound by [-1,1]. The largest peak in the NSDF output corresponds to a lag τ where the detected frequency is the sample rate divided by the lag. The benefit of normalisation is that a threshold can be very easily set to detect this peak. From experimentation, choosing the first major peak to pass a threshold of 0.9 gave the most consistent results.

The accuracy of this pitch detection is limited by the discrete lag values. To further increase the accuracy, parabolic interpolation is used to estimate where a peak may lie in between lags. When the lag associated with the peak is found, its value and its nearest neighbours values are used to form a parabola. The position of the maximum of this parabola can give a closer estimate to where the true peak lies.

The MPM was chosen for its robustness and suitability for instruments. Since the fundamental frequency is often missing from the spectrum of a guitars signal an autocorrelation based method works well. For the full code of the MPM implementation see appendix A.

Motor Control

A simple finite state machine is employed for the control system. The state machine, shown below, calculates the error of the current string pitch to the desired pitch. The state machine will decide the direction and speed of the motor based on the size and polarity of the error. Once the pitch error fits within ± 5 cents of the desired pitch for 5 samples in a row, the next string pitch is loaded into the state machine and the process starts again.

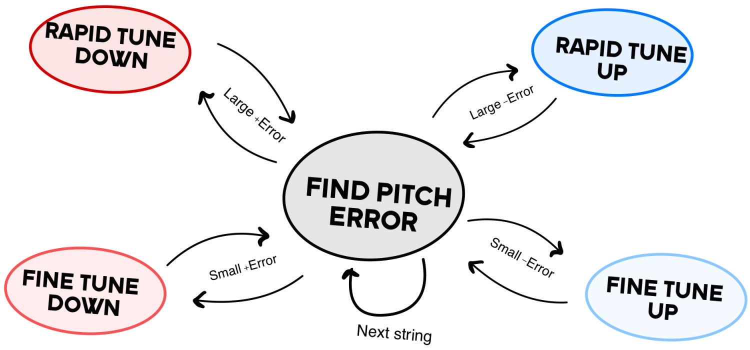


Figure 6.13: *Finite state machine.*

6.2.6 Power

The Auto-Tuner requires two voltages. A 6V to operate the vibration and tuning motor and 3.3V for everything else. To achieve 6V four AA batteries are run in series. At 1.5V each, four AA batteries result in 6V with between 1.5-3 amp-hours of charge. At full load, the motor can reach currents of up to 1A and so these batteries would provide up to 3 hours of continual use. A normal tuning routine lasts up to a minute and a guitar can be tuned several times a day. If it is assumed a guitar is tuned 5 times a day then a full battery pack could last up to $\frac{3\times60}{5} = 36$ days with normal usage. This adequate battery life and and the simplicity of AA batteries is why we chose to use this as the power source.

To achieve the 3.3V a AP1117 low drop-out voltage regulator is used. It was chosen primarily for its simplicity and small footprint.

6.3 Key Design Decisions

Vibration Sensing

The key design decision of the Auto-Tuner was to target the vibrations of the guitar as its input signal. There are two other possibilities: using an audio signal or getting the electrical signal directly from the guitar pickups through its input jack. Each method has distinct advantages and drawbacks. The audio signal can be used with acoustic and electric guitars but has to deal with environmental noise. The direct electrical signal has no environmental noise but is limited to only those guitars that have pickups. Furthermore, it adds the burden of having to bring out cables every time you want to tune. The vibration approach takes the best of both worlds. Vibrations are agnostic towards acoustic or electric and are much less prone to environmental noises than audio. For example, vibration sensing will have some noise from being bumped, but this noise does not tend to lie in the guitar's frequency range and so can easily be filtered out. Unlike audio, which may have environmental noise such as human voice, which does lie in the guitar's frequency range.

The drawbacks of the vibration approach are that vibrations are weak signals and require significant amplification. Furthermore, it has the potential to be interfered with by the tuning motor vibrations. However, these flaws can be overcome with careful design. Therefore, vibration input is the best choice for a robust to noise device with a simpler user experience.

Finite State Machine

Operation of the tuning motor introduces vibrations which interfere with the pitch detection. For this reason, a state machine was chosen as the method of control. The state machine allows for separating tuning motor operation and pitch detection into separate states. Separating these into different states means that the motor will never be running while the pitch is being detected. Furthermore, this method is simple to implement and performs well.

Pitch Detection Algorithm

6.4 Specifications

The specifications for all the features of the Auto-Tuner are as follows.

Automatic Tuning

- Tune all 6 strings in under one minute.
- Tune with an accuracy of ± 5 cents.

Metronome

- Implement timings from 20 BPM to 200 BPM.
- Have timing accurate to within 3%.

Tone Generator

- Output frequency from 80 Hz to 330Hz (guitar frequency range).

- Output pitch accurate to within 0.5 cents.

Power

- Last two hours of continuous motor use.

Case

- Have a footprint to fit comfortably in a hand: 10cm $\times$ 5cm $\times$ 3cm

Gyroscope

- The screen should always be orientated upright to the users point of view.

User Experience

- The Auto-Tuner's screen should respond to button presses in less than one second.
- New users should be able to learn to control and navigate all features in under 5 minutes.

6.5 Bill of Materials

Table 6.1: *Component price list - Shipping included into price.*

Item	Component	PPU AUD	QTY	TOTAL	SOURCE
MCU	STM32L431CBT6	6.22	1	6.22	Mouser
Crystal OSC 24MHz	XRCGB24M000F0L00R0	0.221	1	0.221	Mouser
Op-amp	TLV8541DBVR	0.331	1	0.331	Mouser
Ferrite Bead	BLM21AG601SN1D	0.048	1	0.048	Mouser
Resistors 0603	CRCW06031K00FKEAHP	0.034	15	0.51	Mouser
Capacitors 0603	C0603C105K8PAC7411	0.013	20	0.26	Mouser
LED 0603	SML-LXIL0603UPGCTR	0.258	1	0.258	Mouser
Piezo Transducer	ABT-441-RC	0.791	2	1.582	Element-14
OLED Display	SSD-1306	1.92	1	1.92	Alibaba
3-Axis Gyroscope	MPU-6050	4.76	1	4.76	Alibaba
Push Button	TACT SWITCH 1825910-6	0.09	3	0.27	Element-14
Slide Switch	EG1206A	0.653	1	0.653	Element-14
Vibration Motor	ROB-08449	3.15	1	3.15	Mouser
Continuous Servo	SG5010	6.6	1	6.6	Alibaba
Linear Regulator	AP1117	0.167	1	0.167	Mouser
Speaker Module	XC3744	4.75	1	4.75	Alibaba
			TOTAL	31.69	

6.6 Performance Analysis

The specifications able to be tested were: tuning speed, tuning accuracy, metronome range and metronome accuracy.

Tuning speed

The guitar was made out of tune and placed on a table with the Auto-Tuner laid flat with its motor attached to the low E tuning peg (see figure 6.14). A timer was then started when the tuning sequence was initiated.

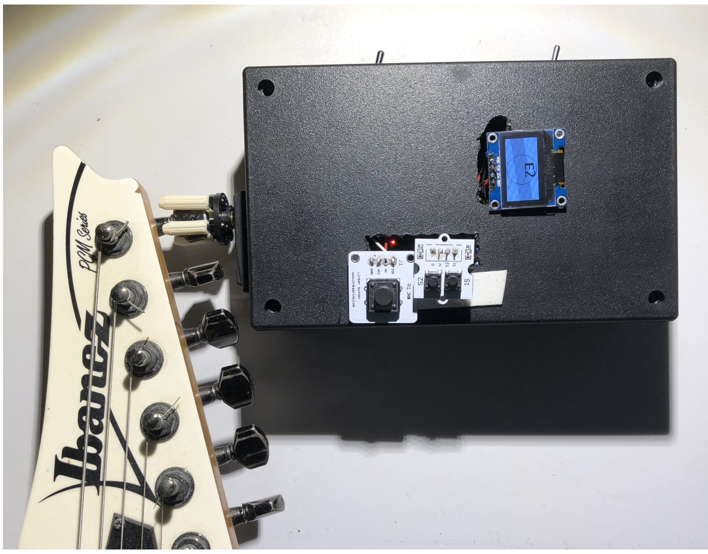


Figure 6.14: *Tuning speed experiment setup.*

Results and Comments

The resulting time was 1 minute and 34 seconds. The reason for the slower than specified time of 1 minute is that the D string was detecting some false pitches which made the tuning over shoot several times. The pitch detection needs some fine tuning for this string.

Tuning Accuracy

After the tuning speed test the guitars tuning was analysed with an accurate guitar tuning app. The results for each string are shown in the figure below.

Results and Comments

The goal was to be within 6 cents of the desired pitch. According to the guitar tuner app each string had an error of 2-3 cents except the G3 string which had an error of approximately 6 cents. The Auto-Tuner fulfills this specification.

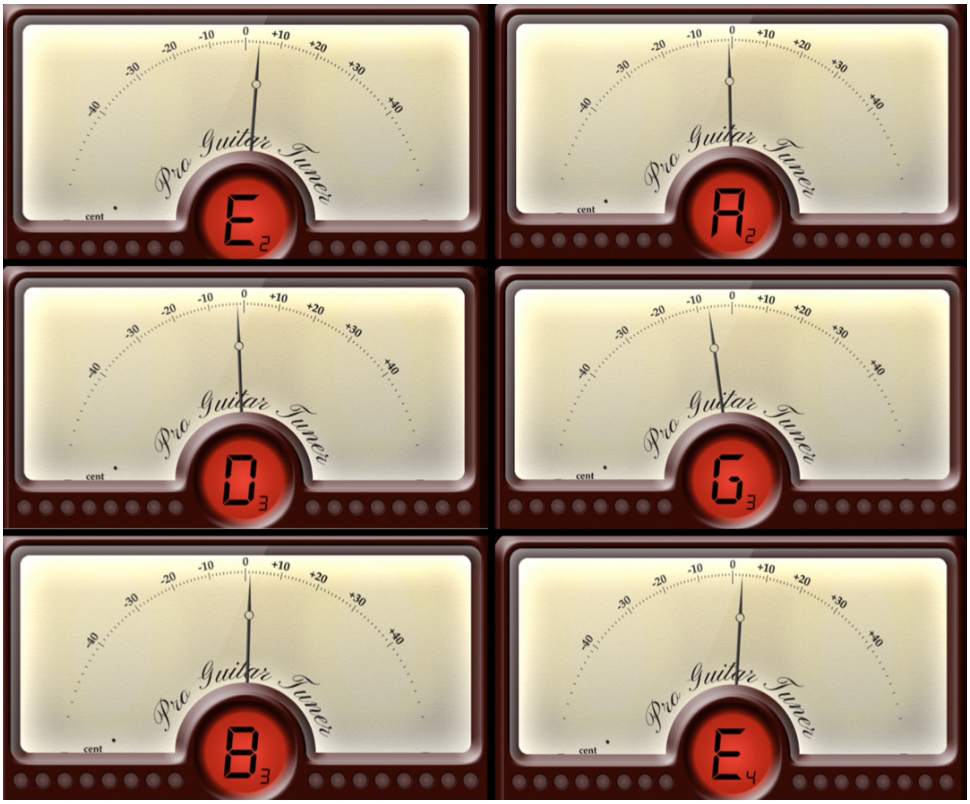


Figure 6.15: *Guitar strings accuracy in cents.*

Metronome Range and Accuracy

To test the pulse-metronome's range and accuracy a piezo sensor was attached to the surface and the output connected to an audio input jack which was then sampled in Matlab for 10 seconds (see figure 6.16 and figure 6.18).

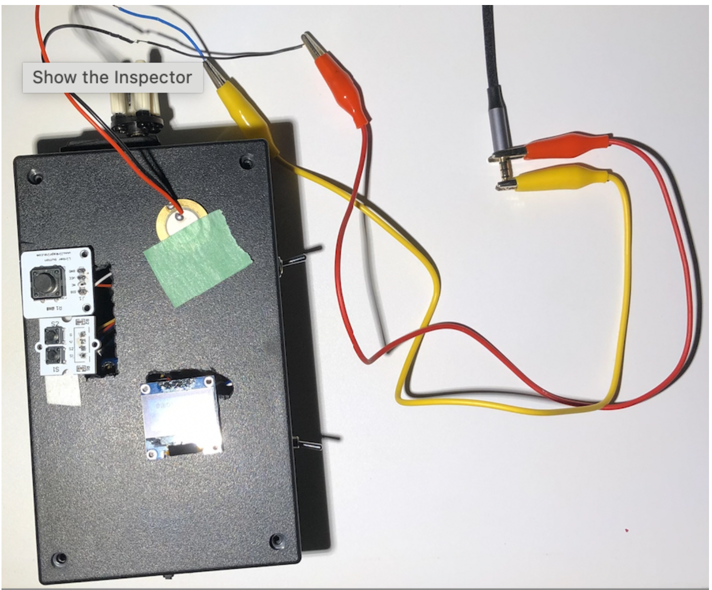


Figure 6.16: *Metronome experimental setup.*

Results and Comments

For the 20bpm test the distance between pulses is $4.13651 - 1.13668 = 2.99983$ between pulses. This equates to $\frac{60}{2.99983} = 20.011$ bpm. This has an error of $\frac{20.011-20}{20} \times \%100 = 0.055\%$.

The 200bpm test has a distance of $2.91993 - 2.62088 = 0.29905$ which is $\frac{60}{0.29905} = 200.635$ bpm. This is an error of $\frac{200.635-200}{20} \times \%100 = 0.32\%$.

The Auto-Tuner performed 20bpm to 200bpm will under the desire 3% error margin.

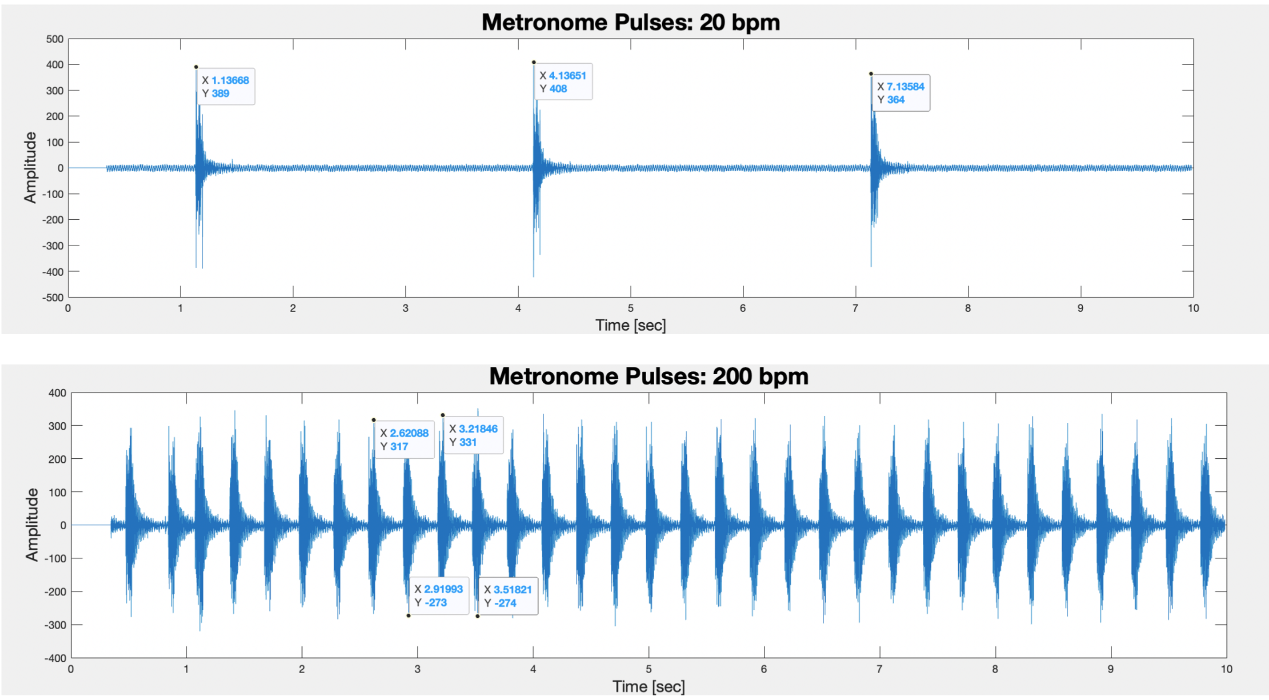


Figure 6.17: *Metronome pulses sampled in Matlab.*

Self Orientating Screen

The test was done by simply placing the device in an upright position and then flipping it 180 °and recording the results. Both orientations were captured and displayed in the figure below.

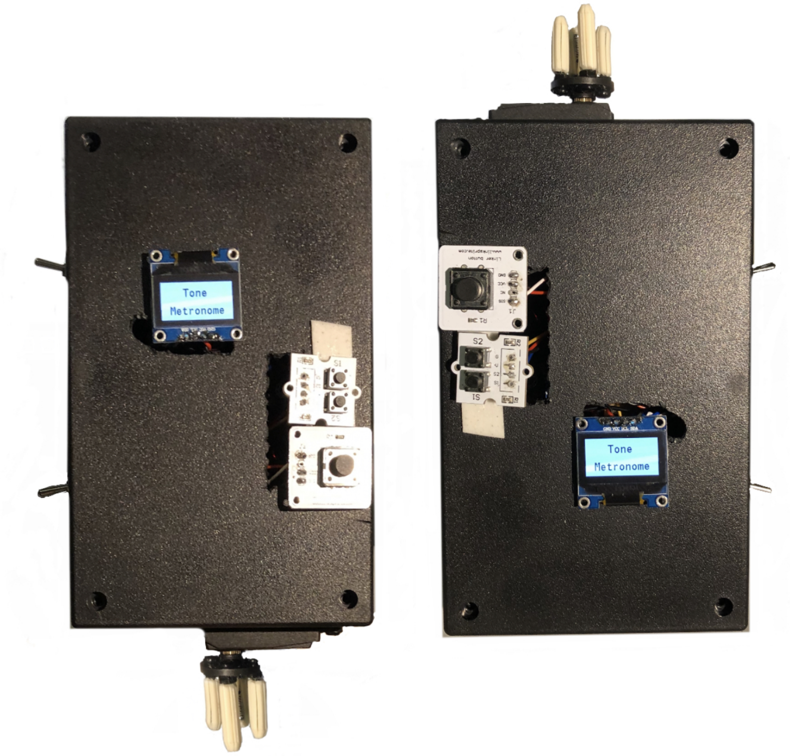


Figure 6.18: *Self orientating screen test.*

Results and Comments

From the image, it can be seen that the screen remains upright despite the body of the device being flipped. This fulfils the self oreintating specification.

Business Plan

7.1 Five Year Business Plan

The goal of Auto-Tuner in the first five years is to secure a minimum 10% of the entire automatic tuner market (based on projected sales in section 5.2). To help guide towards this goal a five year plan has been laid out.

Year 1

Research suggests that electronic devices of similar size can take one to two years of development [10]. Since the Auto-Tuner is aimed at a drastically lower price point than the competition its features are much more limited and so its development time is estimated at being towards the one year mark. Hence, the first year of the plan is the development phase.

In addition, during the first year a kick starter will be initiated to achieve three things:

- Initiate advertisement of the product before development has been completed.
- Gauge level of interest before product release.
- Help fund the cost of development.

Year 2

In the second year the product will be released and marketing campaign begin. Resources suggest that businesses 2-5% of operational costs should go towards advertising [11]. Auto-Tuner will go with the 5% to help ensure the 10% market share goals.

Year 3

After the product has been on the market for a year it is likely the device will require some updates based on user experience or flaws discovered. To account for this 1-months of development costs are reserved to make minor revisions.

Year 4

Updated product released.

Year 5

After 4 years of the Auto-Tuner being on the market research will begin on user experience. This will be in the form of surveys and reviews. This will be the foundation of the development phase for the Auto-Tuner 2 in the following years.

7.2 Five Year Development Costs Estimate

7.2.1 Manufacturing Cost

PCB Manufacturing and Assembly
The cost of PCBs and component assembly was estimated through PCB manufacturer PCBWay's online quoting tool [12]. Using a pad size estimation of 50mm x 50mm, 4 layers, one sided, and one design per unit, a price is quoted as $0.61 per pad, for orders of 10,000.

PCBWay also offers assembly. For a device with 17 unique components, 44 SMD components, and a 1 LQFP component (MCU) the cost of assembly comes to $0.82 per unit for orders of 10,000.

Total cost per unit: $1.43

Injection Moulded Electronics Casing
The Auto-Tuner will require a plastic casing to house the PCB and other components. The casing price is estimated by PCBWay's online quoting tool for injection molded orders [13]. CAD files for a similar-sized guitar tuning device were downloaded from a free CAD library and used as a proxy for the Auto-Tuner design. The files were uploaded to PCBWay's quotation tool where they quoted $17,592 AUD for 5000 units. This comes to $3.52 per unit.

Total cost per unit: $3.52

Component Handling and Insertion Costs
The final product will require further assembly. The motors, sensors, and OLED will require assem-

bling into the plastic casing. The price is estimated using the recommended component handling and insertion costs guide in the lecture notes as \$0.25 per component [14]. The remaining components are: 2 x motors, OLED display, 2 x battery cells, 2 x battery management module, 2 x peizo transducer, gyroscope, and buck converter for a total of 10 items.

Total cost per unit: \$2.50

Packaging and Shipping

The packaging and shipping is estimated with the provided formula below, where k is the weight of the Auto-Tuner [14].

$$k+1 = cost$$

. Based on the weight of the heaviest components, and the weight of similar sized devices, an estimate of $k = 0.4$ kg is used.

Total cost per unit: \$1.40

Overhead Margin

A flat 20% is added to the cost per device to accommodate all other indirect costs.

Total Manufacturing Costs

The total cost for manufacturing, including overhead, comes to **\$51.2** AUD per unit.

7.2.2 Development Costs

Initial Development

The Auto-Tuner will require the expertise of engineers to develop the final product over a period of one year. Assuming that an average of 1 engineer will be working full-time for 52 weeks at \$120 an hour, this comes to a total of $120 \times 35 \times 52 = \$218,400$.

Product Update

In the third year of the business plan it is expected the Auto-Tuner will need some revision to fix any bugs or defects. One month of an engineers wage will be used to estimate this cost. Using one engineer for 40 hours per week at \$120 per hour, this comes to $4 \times 40 \times \$120 = \$19,200$

User Experience Research
In year 5 of the 5 year business plan research will begin on user experience to inform development of the potential Auto-Tuner 2. An estimate of this cost is done through Crux Collaborative's online price estimate [15]. Crux Collaborative offer user experience research for \$20k-40k AUD. Therefore, \$20,000 will of development cost will be reserved for the fifth year.

7.2.3 Operational Costs

The ongoing costs will be adverting of the product. The amount to spend on advertising is recommended between 2-5% according to online resources [11]. The Auto-Tuner will use 5% to push the sales of the product to maturity sooner.

7.3 Estimate of Potential Profitability

To estimate the profitability a net present value (NPV) is calculated for each quarter over the span of 5 years. The NPV estimates the profitability of an investment with equation below, where r is the discount rate and y_n is the net revenue for the nth quarter.

$$NPV = \sum \frac{y_n}{(1+r)^n}.$$

The discount rate averages between 7.5% to 9%, to be conservative 9% was chosen for this analysis [16].

The sales per quarter are estimated by the formula:

$$S(n) = S_0(1 - e^{-an}).$$

For a given nth quarter the total sales expected is an asymptotic curve towards the mature sales volume S_0(calculated in section).Where the time constant a models the steepness of the curve. An $a = 0.35$ is used due to the investment into advertisement and $S_0 = 2,700$ was calculated in section 5.2.

The below tables contains the NPV per quarter for the first 5 years.

Table 7.1: *Year 1 NPV per quarter.*

Year 1	**Q1**	**Q2**	**Q3**	**Q4**
Development Costs	-218,400	0	0	0
Sales (units)	0	797	1,359	1,755
Sales Revenue	0	67,745	115,515	149,175
Manufacturing Costs	0	-40,790	-69,554	-89,821
Business Tax (25%)	0	-6,738	-11,490	-14,838
Operational Costs	0	-3,387	-5,765	-7,459
Net Revenue	0	16,828	28,695	37,057
NPV	-218,400	-202,960	-178,808	-150,193

Table 7.2: *Year 2 NPV per quarter.*

Year 2	**Q1**	**Q2**	**Q3**	**Q4**
Development Costs	0	0	0	0
Sales (units)	2231	2369	2467	2536
Sales Revenue	172,890	189,635	201,365	209,695
Manufacturing Costs	-104,100	-114,182	-121,245	-126,261
Business Tax (25%)	-17,197	-18,863	- 20,030	-20,858
Operational Costs	-8,645	-9,482	-10,068	-10,485
Net Revenue	42,948	47,108	50,021	52,091
NPV	-119,769	-89,152	-59,326	-30,830

Table 7.3: *Year 3 NPV per quarter.*

Year 3	**Q1**	**Q2**	**Q3**	**Q4**
Development Costs	-19,200	0	0	0
Sales (units)	2584	2618	2643	2660
Sales Revenue	215,560	219,640	222,530	224,655
Manufacturing Costs	-129,792	-132,249	-133,989	-135,268
Business Tax (25%)	-21,442	-21,848	-22,135	-22,347
Operational Costs	-10,778	-10,982	-11,127	-11,233
Net Revenue	34,348	54,561	55,279	55,807
NPV	-13,592	11,529	34,879	56,506

Table 7.4: *Year 4 NPV per quarter.*

Year 4	**Q1**	**Q2**	**Q3**	**Q4**
Development Costs	0	0	0	0
Sales (units)	2671	2680	2686	2690
Sales Revenue	226,100	227,035	227,800	228,310
Manufacturing Costs	136,138	136,701	137,162	137,469
Business Tax (25%)	-22,490	- 22,583	-22,659	-22,710
Operational Costs	-11,305	-11,352	-11,390	-11,416
Net Revenue	56,166	56,398	56,588	56,715
NPV	76,475	94,871	111,805	127,375

Table 7.5: *Year 5 NPV per quarter.*

Year 5	**Q1**	**Q2**	**Q3**	**Q4**
Development Costs	-20,000	0	0	0
Sales (units)	2693	2695	2697	2698
Sales Revenue	228,650	228,905	229,075	229,245
Manufacturing Costs	137,674	137,827	137,930	138,032
Business Tax (25%)	-22,744	- 22,769	-22,786	-22,803
Operational Costs	-11,433	-11,445	-11,454	-11,462
Net Revenue	56,799	56,863	56,905	56,947
NPV	136,644	149,784	161,848	172,924

In five years the Auto-Tuner shows a net present value of $173,000, where year 3 quarter 2 is when it starts to turn a profit.

Development Plan

8.1 Key Future Tasks

With the initial prototype completed, there is still a long process that needs to be undertaken and satisfied before Auto-Tuner is fit for consumers and commercial sales. The prototype was created to show that it can be done and now there is a lot of key tasks which will need to be done before it is completed. In 7, the business plan for the next five years was discussed. It is evident that the first year is the toughest and one that will require the most considerations before the product is rolled out. Provided in this section is a breakdown of the first year and a deeper look into how to ensure that Auto-Tuner is released under perfect conditions.

8.1.1 Re-design & Improvements

This is the first task that will need to be undertaken and is the stage where the feedback from the presentation of the prototype is taken and reworked to be better. A similar breakdown will happen here as in the previous parts, since each member presented their individual sub-systems, and use that specific feedback to improve their own sections. The major feedback that was given are summarised below.

- Move all stationary and passive components into a single PCB, preferably surface mounted. This will significantly save space and ensure that there are no wire connections that can become frayed and deteriorate over time.
- 3D print the case exactly for the product. This will ensure that each component have their own secure location in the case and are tightly fitted. Furthermore, ensure that the case is ergonomic and fits well into a person's hand.
- Improve the quality of the sound coming out of the speakers. A guitar does not produce a single sine wave frequency, rather it is built up of many harmonics and this should be decoded and used for tone generation to get as close a sound as possible.
- Optimise the menu screen to be more responsive to the user's input and perhaps get a slightly larger screen if costs permit.

8.1.2 Verification & Field testing

This is another key part that will be necessary to get additional feedback from the actual users and make modifications before the product is rolled out. This phase will consist of a controlled distribution of the enhanced prototype which will be closely monitored and at the end a survey taken about their experience whilst using this device. The feedback will be invaluable as it would be coming directly from the users themselves. In this section the most important things would be:

- Production of a small group of prototypes.
- Controlled distribution to a small number of musicians and potential users.
- Gathering data at the end of the trial and analysing the survey data for potential fixes and modifications.
- Making the modifications before rolling out the final product.

8.1.3 Production & Sales

This will be the final phase in the first year where the product should be optimised for the market before being rolled out. However the work does not stop here, as even if the product is now complete, it need to be distributed and consumers need to know of its existence. This is why this phase is one of if not the most important. In this phase the following should be completed:

- Find a trust worthy manufacturer to mass produce Auto-Tuner.
- Find storage space and make valuable connection with retailers to distribute the product to.
- Run advertising campaigns so that the target market knows that this product exists and is available.

By now, Auto-Tuner should be perfected and ready to hit the market. Here is where the marketing team would take over and create advertisements and marketing campaigns over the next few years. However constant revision will still be necessary as stated before, the market is dynamic and the needs of consumers are always changing. Updates and improvements will consistently need to be made in order to maintain competitiveness and advantage over competitors.

8.2 Work Breakdown Structure

With the initial work that was necessary to build the product, additional work is required to bring it to completion. Included in this work breakdown structure is both the structure pre-prototype and post-prototype and how each task will be broken down into manageable sections.

1. **Research & Development (Completed - 19 hrs)**
   This was a major sector of our design process and involved doing primary and secondary research, thinking about our target market and our competitors to understand how we could differentiate ourselves from others already on the market.

   A) Brainstorm Ideas - Think about complete project, how it should interface and what key features that will need to be implemented (**5 hrs**).

   B) Use a survey to collect data from target musicians to understand the behaviour of the market and its needs (**1 hr**).

   C) Look at the competitors and how their products behave for the user as well as on the market. Check the reviews they receive and innovate any improvements that could be introduced (**7 hrs**).

   D) Determine costs & profits - including manufacturing, servicing and production costs, NPV and future value of our products (**6 hrs**).

2. **Product planning & Design (Completed - 11 hrs)**
   This part was a pivotal section where the components and products were chosen and concept designed. This involved making final decisions on the technical components and how the subsystems will be designed and the complete product interface with each subsystem. This also required us to think from the users perspective and how they will interact with Auto-Tuner.

   A) Using the requirements and needs analysis to decide on the features (**6 hrs**).

   B) View reviews on current products and determine what the market believes needs to be changed to competitor products (**4 hrs**).

   C) Buy components that have already been chosen (**1 hr**).

3. **Component Assembly (Completed - 41 hrs)**
   This is the largest section where the assembling of the complete project onto a breadboard and interfaced to the MCU took place. This section includes many sub tasks which will be delegated to each team member generously.

   A) Interfacing OLED screen with MCU (**2 hrs**).

   B) Interfacing gyroscope and connecting it to OLED screen (**4 hrs**).

   C) Connecting buttons and switches to MCU and interfacing them to activate a response within the system (**1 hr**).

D) Figure out how to use timing interrupts and hardware interrupts (**4 hrs**).

E) Create an algorithm to determine tone from vibration signal and supply a response signal (**10 hrs**).

F) Connect vibration sensor and amplification circuit together to MCU (**3 hrs**).

G) Output a tone from the buzzer (**4 hrs**).

H) Connect the vibration motor (**2 hrs**).

I) Interface voltage regulator to the power circuit as well as recharging circuit to allow for recharging (**5 hrs**).

J) Create a feedback loop and connect to bi-direction motor to rotate the tuning pegs correctly in both directions (**5 hrs**).

K) Create Menu interface for user experience on the screen (**3 hrs**).

L) Timing software system to act as a metronome to keep timing (**2 hrs**).

4. **PCB Design (Completed - 16 hrs)**
This section involved moving all our parts onto a single PCB. This will help save space and ensure that there are no exposed components. A singular PCB will also simplify the product and save space which is important for a size constrained device.

A) Plan the components placement on PCB (**5 hrs**).

B) Connect and solder all components onto PCB (**10 hrs**).

C) Ensure that there are spaces for extra addition if necessary (**1 hr**).

5. **Structural Design (Completed - 5 hrs)**
This section revolves around creating an encasing for the products and only letting the user view the key features rather than the electronics behind the system. The user should only view the display, buttons and switches.

A) Find suitable encasing that is appropriate for this device on the internet if it exists (**1 hr**).

B) Drill holes and cut holes for switches, buttons and screen (**2 hr**).

C) Fit the product with proper placement so that components are set inside the case (**2 hrs**).

6. **Final testing & Product (Completed - 16 hrs)**
This is the final section where all components and features came together into one simple device which should be handheld and compact. It should features display and buttons for user interaction and involves going through the following to ensure this task is completed.

A) Testing (**8 hrs**).

B) Final Presentation (**4 hr**).

C) Final report (**4 hrs**).

7. **Re-design & Improve (To be completed - 20 hrs - 3 weeks)**
   This section involves taking in the feedback from the prototype and applying that to redesign and improve the product. Also improve any other systems which were not optimised and efficient as possible. Some of the feedback and systems which will need improvement are

   A) Placing all components in a single PCB (**5 hrs**).

   B) 3D printing a case and making it more comfortable in the hand (**7 hrs**).

   C) Improve the sound quality of the tone generator and make it sound more realistic (**4 hrs**).

   D) Improve the menu screen (**4 hrs**) .

8. **Verification & field testing (To be completed - 60 hrs - 6 months)**
   This section is for after the designing has been successful and all investors/ shareholders are happy with our design then this step is about testing it in the field, taking surveys before and after using this product letting free samples be used by users in a real-scenario setting.

   A) Production of prototypes (**5 hrs**).

   B) Controlled distribution to users along with survey (**35 hrs**).

   C) Gathering and analysing survey data (**5 hrs**).

   D) Making necessary revisions based on feedback (**15 hrs**) .

9. **Production & sales (To be completed - 50 hrs - years)**.
   This section goes for once the prototype has been completely verified and tested on the field and assuming positive feedback is received. This section involves mass production and selling to the consumers at a global scale.

   A) Find a trusty manufacturer who is capable of mass producing Auto-Tuner (**15 hrs**).

   B) Find storage place and shops where this product can be distributed and held at (**10 hrs**).

   C) Run advertising campaigns to show the consumers this product exists (**25 hrs**).

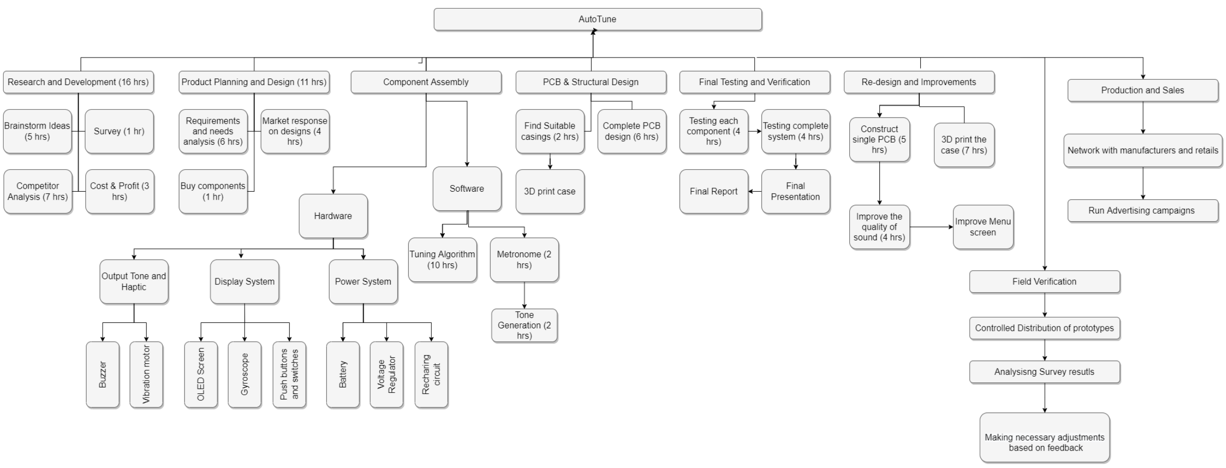


Figure 8.1: *Work Breakdown Structure.*

8.3 Gantt Chart

The gantt chart is a summary of the work structure breakdown and how the tasks are divided among members and the dependencies that each specific task has on other tasks as well as the timeline for each section. It includes the tasks that have already been completed and the tasks that are still leftover.

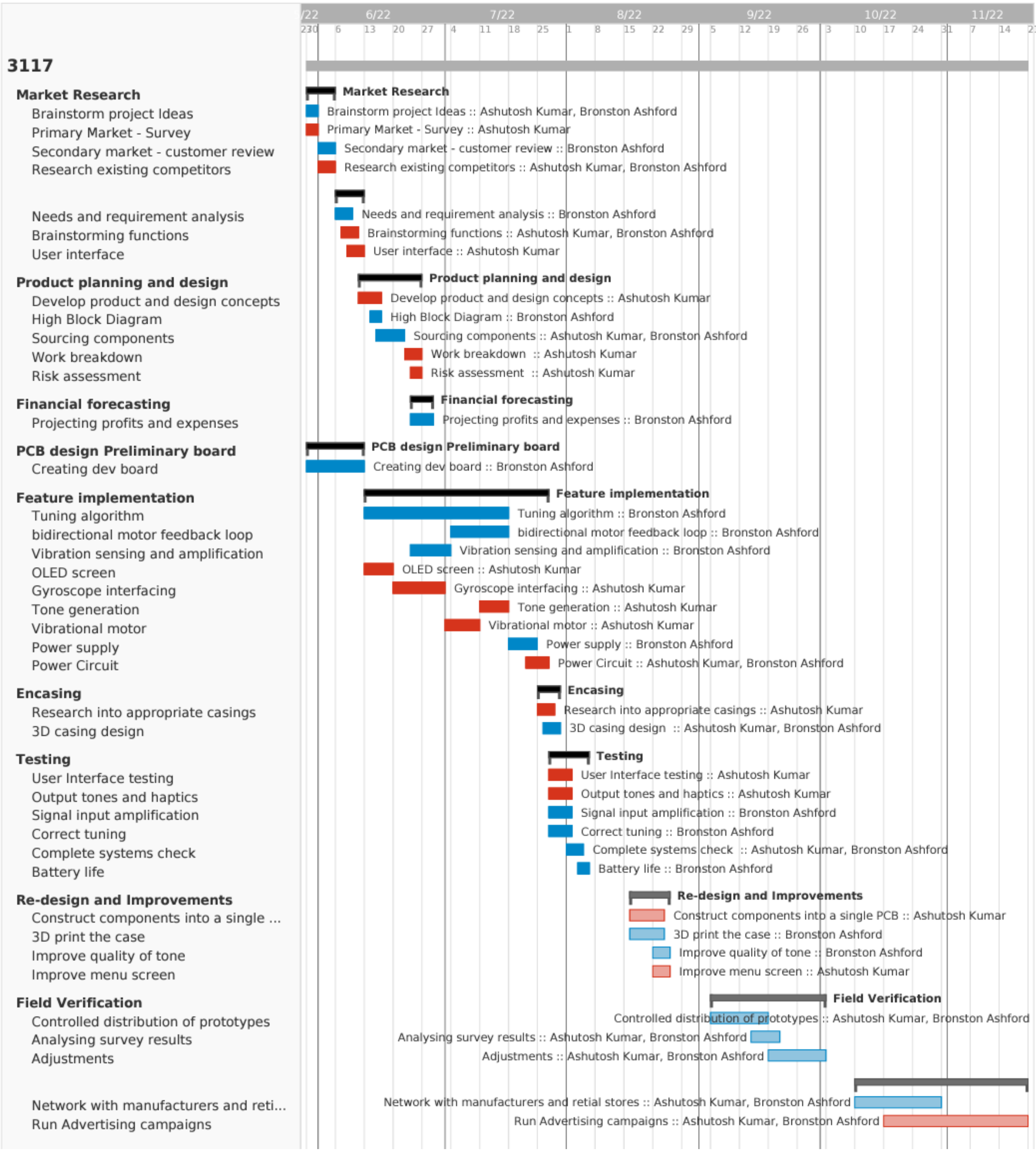


Figure 8.2: *Gantt Chart of Project.*

8.4 Additional Resources

There are some additional resources which will be necessary for Auto-Tuner to reach completion.

8.4.1 3D Printer

This is a key component which will be necessary for the designing of the case to house the components. Having a 3D printer would allow for freedom to design the case in the perfect shape to house all the components neatly and securely. This is especially important for components that cannot be built onto the PCB such as the vibration motor which will need to be housed securely to the box and away from the other components so that all its vibration is transferred to the case and not the other components.

8.4.2 PCB Milling Machine

The amplifier circuit, MCU, gyroscope can all be built within a single PCB. This will significantly reduce space and eliminate wires which can be frayed during transport or rough use. Creating a singular PCB will made the system more slick, secure and robust. To fabricate the PCB boards a PCB milling machine is required.

8.4.3 Engineers

Marketing is a critical component for any new products success and failure. If the customer basis aren't aware of this product - they will not purchase or search for it. Marketing is a vital tool and it is important that all beginner guitarist know that a device which can support them in their learning exists.

8.4.4 Marketing team

To further develop this tasks and to make the improvements that have been suggested in the key future tasks section, engineers will need to dedicate hours to solving the problems. These can be finished using the same engineers who designed the system and giving them feedback on what can be improve will carry it to completion.

8.4.5 Funding

As all start-ups require, Auto-tuner will also require funding and investments for it to be mass produced, transported, advertised and stored until profits are made.

8.5 Risk Management

The risk assessment goes through the potential different risks that may be encountered whilst developing Auto-Tuner. These risks range from direct risks associated with the production to the risks post production when selling to the market. Included here are the technical, business and project management risks.

8.5.1 Technical Risks

Table 8.1: *Technical Risks.*

Risks	**Probability**	**Severity**	**Mitigation Strategy**
Mechanical failure of components	Low	High	Ensure extra components are available Check each component works Ensure each sub-system works individually
Short circuit of MCU or other components due to improper connection or soldering	Low	High	Read the data sheets of each component carefully Know the current and voltage limits of each component before supplying power
Failure of tuner working with different guitars	Medium	High	Intensive testing on the field using different guitars and if possible different stringed instruments
insufficient noise filtering resulting in incorrect pitch detection	Medium	High	Ensure that algorithm is thoroughly tested with both simulated noise and real noise Filter out noise as much as possible both digitally as well as through analogue means.
Vibration motor causing problems in operations of device due to feedback	Low	medium	Separate the vibration motor from the circuit and ensure it has its own separate section in the encasing.
Incorrect PCB design	Low	Medium	Members should be familiar with the software used for PCB design and double check all connections before sending it for construction.

8.5.2 Project Management

Table 8.2: *Project Management Risk.s*

Risks	Probability	Severity	Mitigation Strategy
Illness or Covid cases	Medium	Medium	• Set guidelines on how employees should act if they display any symptoms of illness or Covid. • Ensure that workspace is regularly cleaned and care is taken to limit spreading by regularly washing hands and wearing masks.
Overlap of tasks and repetition between members due to poor communication	Low	Low	Have a clear line of discussion and communication between team members so that everyone is aware of exactly which tasks they are responsible for.
Components becoming harder to source	Medium	High	• Ordering components earlier than necessary to account for delays. • Having contingencies in place such as extra components that could replace others in case of shortage. • Order from Australian based manufacturers to decrease shipping delays.
Delays in PCB manufacturing	Medium	Medium	• Ordering earlier than necessary to account for delays. • Using Australian (local) manufacturers to account for shipping delays.
Inaccurate deadlines and budgeting constraints	Low	High	Set clear deadlines and budget which are strict and communicated well with the team members.

8.5.3 Business Risks

Table 8.3: *Business Risks.*

Risks	**Probability**	**Severity**	**Mitigation Strategy**
Decreasing Demand of such products	Low	High	• Ongoing Market research • Continuously adapting and changing product to meet market demands • Researching for consistent improvement of device and addition of extra features
Failure of incorrectly forecast earnings and losses	Low	High	• Continuously revisiting and revising budget • Making adjustments to ensure that the budget and forecast are on track as much as possible

8.6 Contribution Breakdown

Table 8.4: *Task Allocation.*

Task	**Ashutosh**	**Bronston**
Primary Market Research	✓	
Secondary Market Research		✓
Needs Assessment	✓	
Requirement Analysis	✓	
Problem Statement		✓
Product & design concept		✓
High Block Level Diagram		✓
Technical component	✓	
Present and future costs & profitability		✓
Work structure breakdown & gantt chart	✓	
Risk Assessment	✓	
Tuning Algorithm		✓
Signal Input subsystem		✓
Tuning control and feedback		✓
Display subsystem	✓	
Output Tones and haptics	✓	
Power (50 %)	✓	✓
Encasing		✓
Testing (50 %)	✓	✓
Redesign	✓	
Improvements		✓
Verification		✓
Field Testing	✓	
Production (50 %)	✓	✓
Sales (50 %)	✓	✓
Post production (50 %)	✓	✓

Conclusion

Tuning is a very important step for musical instruments to ensure that they produce the right pitch, this is specifically true for stringed instruments which need constant tuning. Currently on the market there are a great deal of tuners and accessories which help guitarist ensure that their instrument is in pristine condition. This is especially necessary now due to the growing influx of beginners entering the guitar market. Whilst beginners usually require a tuning device, they need further support to truly learn the guitar.

Market analysis shows that majority guitarist are taking up self-motivated learning which is growing at an exponential rate. This produced a great opportunity for a product such as Auto-Tuner to take the market with the additional feature that it offers. With a estimated $85 price point, it is a low priced accessory which almost all guitarist should have in their arsenal, it being capable of automatically tuning a guitar from its pitch to producing a tone. The user is bound to improve and appreciate these features as they will see an rapid incline of their skills. With a small competitor market, Auto-Tuner is a unique product with a lot to offer.

There are still a lot of revision and developments that will need to be made on the device before it is mass producible. The feedback from the prototype assignments to users feedback during field testing will make sure that consistent revision and improvements are made to get the perfect product which can then enter the market competitively. A business plan has been developed for the next 5 years and net present value calculations which show that Auto-Tuner will start making a profit in 3.5 years and has a present value of $ 173,000.

Auto-Tuner goes beyond being a simple tuner, it is a teacher and musical friend which only cares about making sure everyone can learn an instrument and have some musical talent. It is a device which will most definitely be used in most musical homes as an accessory to build ones skills either individually or with friends and family.

Bibliography

[1] Music Strive. *Guitar Sales Statistics 1*, 2022. https://musicstrive.com/guitar-sales-statistics/.

[2] Student of guitar. *Guitar Sales Statistics 2*, 2022. https://studentofguitar.com/guitar-industry-statistics/.

[3] Amazon. *Guitar Tuner Search Results*, 202. https://www.amazon.com.au/s?k=guitar+tuner&i=mi&ref=nb_sb_noss.

[4] Fenders Newsroom. *Fenders Newsroom*, 2021. https://spotlight.fender.com/newsroom/news/844.

[5] LandingCube. *HOW TO GET REVIEWS ON AMAZON IN 2021 (PROVEN & SAFE METHODS)*, 2021. https://landingcube.com/how-to-get-reviews-on-amazon/.

[6] Digital Commerce 360. *US ecommerce growth tapers off in recent quarters*, 2022. https://www.digitalcommerce360.com/article/us-ecommerce-sales/.

[7] Mecharithm. *Explicit Representation for the orientation in robotics: Roll-Pitch-Yaw Angles*, 2022. https://www.mecharithm.com/explicit-representations-orientation-robotics-roll-pitch-yaw-angles///.

[8] Stack Exchange. *How to calculate Euler Angles from gyroscope output*, 2015. https://robotics.stackexchange.com/questions/6953/how-to-calculate-euler-angles-from-gyroscope-output//.

[9] G.Wyvill P.McLeod. A smarter way to find pitch. 2005.

[10] Predictable Designs. *How long does it take to develop a new product and get it to market?*, 2022. https://predictabledesigns.com/how-long-to-develop-a-new-hardware-product-and-get-it-to-market/.

[11] The Media Captain. *How Much Money Should You Spend on Advertising and Marketing?*, 2022. https://www.themediacaptain.com/how-much-money-should-you-spend-on-advertising-and-marketing/.

[12] Price Quote. *PCB Specification Selection*, 2022. https://www.pcbway.com/orderonline.aspx?x=50&y=50&num=-1&Layersquote=4&Thicknessquote=1.6.

[13] Price Quote. *Injection Molding*, 2022. https://www.pcbway.com/rapid-prototyping/manufacture/?type=4&reffercode=TOP.

[14] D.Taubman. Chapter 6: Economics and costing. 2016. D.B.Loeffler,"Costing your ELEC3117 Design Project".

[15] Crux Collaborative. *Effective Ways to Plan and Budget for User Research*, 2018. https://cruxcollaborative.com/insights/effective-ways-plan-and-budget-user-research//.

[16] Globe St. *The Cap Rate and Discount Rate*, 2019. https://www.globest.com/2019/08/08/the-cap-rate-and-discount-rate//.

[17] Bloomberg press. *Guitar Market size*, 2021. https://www.bloomberg.com/press-releases/2021-12-06/electric-guitars-market-size-worth-6-4-billion-by-2028-ca

[18] CB insight. *Band Industries*, 2022. https://www.cbinsights.com/company/roadie-tuner.

[19] Wikipedia. *Roadie Tuner*, 2022. https://en.wikipedia.org/wiki/Roadie_Tuner.

[20] D.B.Loeffler. Instrument timbres and pitch estimation. 2006. D.B.Loeffler,"Instrument Timbres and Pitch Estimation in Polyphonic Music Archived 2007-12-18 at the Wayback Machine". Master's Thesis, Department of Electrical and Computer Engineering, Georgia Tech. April(2006).

Appendix

10.1 Appendix A - McLeod's Pitch Method C Code

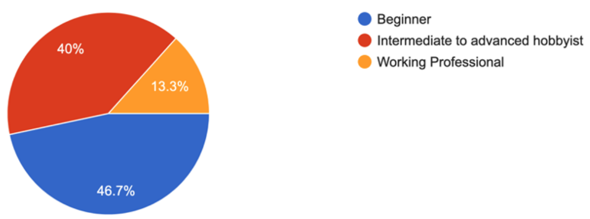
```c
#include <stdio.h>
#include <string.h>
#include <stdint.h>
#include <math.h>
#include "mpm.h"

#define PEAK_THRESHOLD 0.9
float32_t mpm_reserved_memory[2 * BLOCK_SIZE] = {0};

void mpm_sum_f32(float32_t *pSrc, uint16_t scrLen, float32_t *pRes)
{
   *pRes = 0;
   for (uint16_t i = 0; i < scrLen; i++)
   {
      *pRes += *pSrc;
      pSrc++;
   }
}

void mpm_find_peak_f32(float32_t *pSrc, uint16_t *tau)
{
   uint16_t flag = 0;
   uint16_t valid_peak_flag = 0;
   float32_t peak_value = 0;

   for (uint16_t i = 0; i < BLOCK_SIZE; i++)
    {
      if (flag == 0 && *pSrc < 0)
      {
         flag = 1;
      }
      if (flag == 1)
```

```
        {
            if (*pSrc > peak_value && *pSrc > PEAK_THRESHOLD)
            {
            peak_value = *pSrc;
               *tau = i;
                valid_peak_flag = 1;
            } else if (valid_peak_flag == 1)
            {
               return;
            }
        }
        pSrc++;
    }
}
void mpm_NSDF_f32(float32_t *pSrc, float32_t **pDst)
{
   float32_t *xcorr = &mpm_reserved_memory[1];
   arm_correlate_f32(&pSrc[0], BLOCK_SIZE , &pSrc[0], BLOCK_SIZE, xcorr);
   float32_t *r = &xcorr[BLOCK_SIZE - 1];
   *pDst = r;

   float32_t *xs = &mpm_reserved_memory[0];
   float32_t *p_xs1 = &xs[0];
   float32_t *p_xs2 = &xs[BLOCK_SIZE - 1];
   float32_t xs1, xs2;
   arm_mult_f32(&pSrc[0], &pSrc[0], &xs[0], BLOCK_SIZE);
   mpm_sum_f32(&xs[0], BLOCK_SIZE, &xs1);
   xs2 = xs1;

   for (uint16_t tau = 0; tau < BLOCK_SIZE ; tau++)
   {
      *r = 2 * (*r) / (xs1 + xs2);
      xs1 = xs1 - (*p_xs1);
      xs2 = xs2 - (*p_xs2);
      r++;
      p_xs1++;
      p_xs2--;
   }
}

void mpm_parabolic_interpolation_f32(uint16_t x_pos, float32_t a, float32_t b,
    float32_t c, float32_t *delta_tau)
{
```

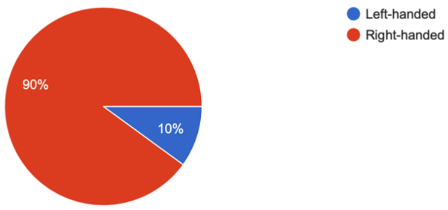
```
    a = 20*log10(a);
    b = 20*log10(b);
    c = 20*log10(c);
    float32_t delta_pos = 0.5 * (a - c) / (1 - 2.0*b + c);
    *delta_tau = x_pos + delta_pos;
}

void mpm_mcleod_pitch_method_f32(float32_t *pData, float32_t *pitch_estimate)
{
    float32_t *p_ncorr;
    mpm_NSDF_f32(pData, &p_ncorr);
    uint16_t tau = 1;
    mpm_find_peak_f32(p_ncorr, &tau);

    if (tau > BLOCK_SIZE - 2)
    {
        tau = BLOCK_SIZE - 2;
    }
    uint16_t xp = tau;
    float32_t a = p_ncorr[tau - 1];
    float32_t b = p_ncorr[tau];
    float32_t c = p_ncorr[tau + 1];

    float32_t delta_tau = 0;
    mpm_parabolic_interpolation_f32(xp, a, b, c, &delta_tau);
    *pitch_estimate = FS / delta_tau;
```

10.2 Appendix B - Primary Market Research Survey

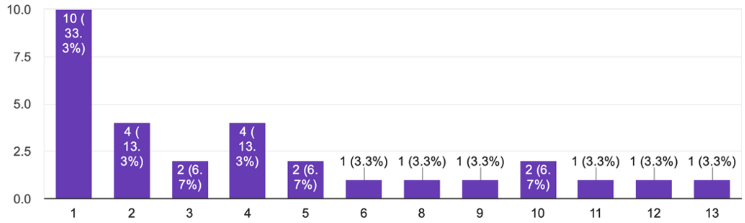
Q1: Do you consider yourself a beginner, intermediate hobbyist or professional?



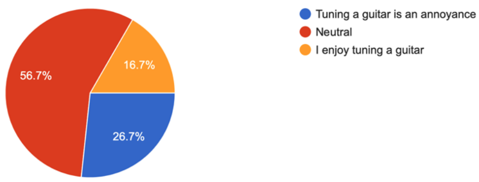
Q2: Are you a left-handed guitarist or right-handed?



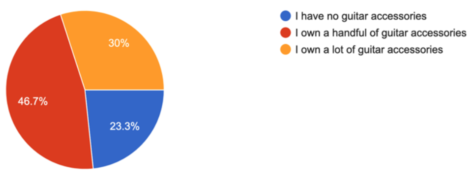
Q3: How many guitars do you own?



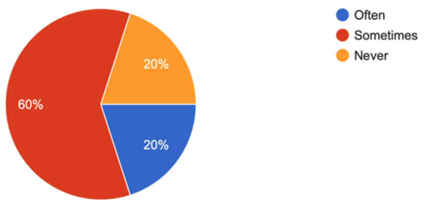
Q4: How do you feel about tuning a guitar?



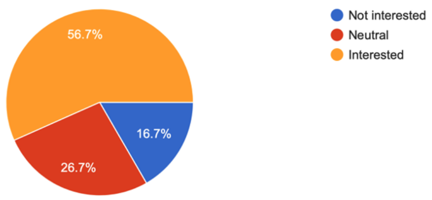
Q5: Do you own any guitar accessories?



Q6: Do you use a metronome when you learn guitar?



Q7: Would you be interested in a device which can automatically tune a guitar?

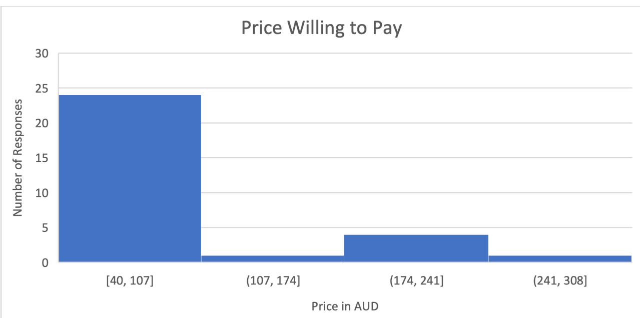


Where 64% of beginners selected 'interested'

Q8: What additional features (if any) would you like to see on an automated guitar tuner?

Metronome, good accuracy, plectrum holder, multiple tuning settings.

Q9: What price would you pay for an automatic guitar tuner?



10.3 Appendix C - Sample Codes

Sample code for Gyroscope timer interrupt

```
// Timer interrupt code
void HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef*htim16)
{
    MPU6050_Get_Accel_Scale(&myAccelScaled);
    MPU6050_Get_Gyro_Scale(&myGyroScaled);
    if (myAccelScaled.x > 0 && screenflip == 0) {
        ssd1306_Init1();
        screenflip = 1;
    } else if (myAccelScaled.x < 0 && screenflip == 1){
        ssd1306_Init2();
        screenflip = 0;
    }
    HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_15);
}
```

Sample code for button operations

```
while (1)
{


    Button1_val = HAL_GPIO_ReadPin(Button1_GPIO_Port, Button1_Pin);
    Button2_val = HAL_GPIO_ReadPin(Button2_GPIO_Port, Button2_Pin);
    Button3_val = HAL_GPIO_ReadPin(Button3_GPIO_Port, Button3_Pin);

        if (Button2_val == 1 || Button3_val == 1){
          if (counter == 0 ) {
              oled_clear_screen();
              ssd1306_DrawRectangle(0, 36, 128,  58, 0);
              ssd1306_UpdateScreen();
              counter = 1;
              HAL_Delay(200);
          } else {
              oled_clear_screen();
              ssd1306_DrawRectangle(0, 8, 128,  30, 0);
              ssd1306_UpdateScreen();
              counter = 0;
              HAL_Delay(200);
          }}

          if (Button1_val == 1) {
              if (counter == 0) {
              oled_timing_screen(timing);
              ssd1306_UpdateScreen();
          } else {
              oled_tone_screen(tone);
              ssd1306_UpdateScreen();
          }}
```

Oled Print functions

```c
void oled_init1(void)
{
    ssd1306_Init1();
    ssd1306_Fill(White);
    ssd1306_DrawCircle(SCREEN_CENTRE_X, SCREEN_CENTRE_Y, RADIUS, Black);

    char home_screen[11] = "Auto-Tuner";
    for(int x = WIDTH; x > MID_X_POS_MED; x--)
    {
        ssd1306_SetCursor(x, MID_Y_POS_MED);
        ssd1306_WriteString(home_screen, FONT_MED, Black);
        ssd1306_UpdateScreen();
    }

    HAL_Delay(3000);

    ssd1306_Fill(White);
    ssd1306_DrawCircle(SCREEN_CENTRE_X, SCREEN_CENTRE_Y, RADIUS, Black);
    ssd1306_UpdateScreen();
}

void oled_init2(void)
{
    ssd1306_Init2();
    ssd1306_Fill(White);
    ssd1306_DrawCircle(SCREEN_CENTRE_X, SCREEN_CENTRE_Y, RADIUS, Black);

    char home_screen[11] = "Auto-Tuner";
    for(int x = WIDTH; x > MID_X_POS_MED; x--)
    {
        ssd1306_SetCursor(x, MID_Y_POS_MED);
        ssd1306_WriteString(home_screen, FONT_MED, Black);
        ssd1306_UpdateScreen();
    }

    HAL_Delay(3000);

    ssd1306_Fill(White);
    ssd1306_DrawCircle(SCREEN_CENTRE_X, SCREEN_CENTRE_Y, RADIUS, Black);
    ssd1306_UpdateScreen();
}

void oled_print_string(char *string)
{
     ssd1306_SetCursor(10, 24);
     ssd1306_WriteString(string, FONT_MED, Black);
     ssd1306_UpdateScreen();

}
```

```c
void oled_print_f32(float *var)
{


    char var_string[10] = {0};
    sprintf(var_string,"%.2f", *var);

    ssd1306_WriteString(var_string, FONT_LARGE, Black);
    ssd1306_SetCursor(25, 24);
    ssd1306_UpdateScreen();
}

void oled_print_no_detection(void)
{
    char var_string = '-';
    ssd1306_WriteString(&var_string, FONT_LARGE, Black);
    ssd1306_SetCursor(SCREEN_CENTRE_X, SCREEN_CENTRE_Y);
    ssd1306_UpdateScreen();
}

void oled_clear_screen(void)
{
    ssd1306_Fill(White);
    ssd1306_DrawCircle(SCREEN_CENTRE_X, SCREEN_CENTRE_Y, RADIUS, Black);
    ssd1306_UpdateScreen();

}

void oled_selection_screen(void)
{
    ssd1306_Fill(White);

    char selection_screen[5] = "Tone";
    char select_screen[7] = "Timing";
    for(int x = 98; x > MID_X_POS_MED; x--)
    {
        ssd1306_SetCursor(40, 10);
        ssd1306_WriteString(selection_screen, FONT_MED, Black);
        ssd1306_SetCursor(30, 40);
        ssd1306_WriteString(select_screen, FONT_MED, Black);
        ssd1306_UpdateScreen();

    }
    HAL_Delay(3000);
}
```

```c
void oled_tone_screen(int tone)
{

    ssd1306_Fill(White);

    char tone_screen[5] = "Tone";
    char tone_scc[7] = "Screen";
    for(int x = WIDTH; x > MID_X_POS_MED; x--)
    {
        ssd1306_SetCursor(x, 10);
        ssd1306_WriteString(tone_screen, FONT_MED, Black);
        ssd1306_SetCursor(x, 40);
        ssd1306_WriteString(tone_scc, FONT_MED, Black);
        ssd1306_UpdateScreen();
    }

    HAL_Delay(3000);

    ssd1306_Fill(White);
    ssd1306_SetCursor(84, MID_Y_POS_MED);
    char tone_string[10];
    sprintf(tone_string,"%d", tone);
    ssd1306_WriteString(tone_string, FONT_MED, Black);
    ssd1306_SetCursor(18, MID_Y_POS_MED);
    ssd1306_WriteString("Tone: ", FONT_MED, Black);
    ssd1306_UpdateScreen();

    HAL_Delay(3000);
}



void oled_timing_screen(int timing)
{

    ssd1306_Fill(White);

    char timing_screen[7] = "Timing";
    char timing_scc[7] = "Screen";
    for(int x = WIDTH; x > MID_X_POS_MED; x--)
    {
        ssd1306_SetCursor(x, 10);
        ssd1306_WriteString(timing_screen, FONT_MED, Black);
        ssd1306_SetCursor(x, 40);
        ssd1306_WriteString(timing_scc, FONT_MED, Black);
        ssd1306_UpdateScreen();
    }

    HAL_Delay(3000);

    ssd1306_Fill(White);
    ssd1306_SetCursor(90, MID_Y_POS_MED);
    char timing_string[10];
    sprintf(timing_string,"%d", timing);
    ssd1306_WriteString(timing_string, FONT_MED, Black);
    ssd1306_SetCursor(5, MID_Y_POS_MED);
    ssd1306_WriteString("Timing: ", FONT_MED, Black);
    ssd1306_UpdateScreen();

    HAL_Delay(3000);
}
```