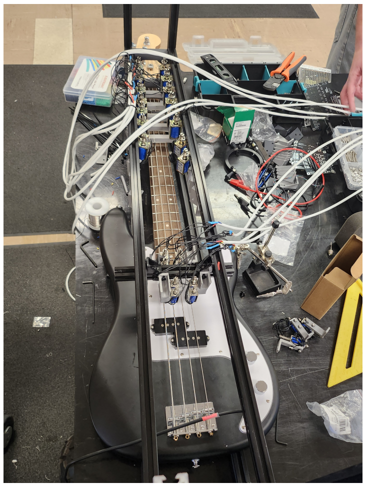
MIDI Bass Guitar Project:

The Continued Development of MIDI Bass Guitar

Idaho State University

College of Technology

6th Semester



Prepared For: Shane Slack

Prepared By: Owen Fujii & Justin Bell

05/07/2025

Table of Contents

Table of Figures

Index of Tables

Abstract

This report details the continuation of the bass guitar project from the work completed in Fall 2024. Development was continued with a focus on the hardware platform and finalized schematics for PCBs. Additionally, a new sub section of the project was developed in order to achieve MIDI over infrared. The project made great progress considering the high demands needed for complete CAD design and assembly. Recommendations were made for future developments to continue the project.

Background Information

This project is a continuation of the revisions that were made to the original bass guitar project. The original MIDI bass guitar project featured bass strings, a fret gantrys, and individual processors for each string. This version the project had limited functionality due to its design that limited its overall performance. Its gantry system caused significant time delay which lead to it lagging behind other instruments in the band or outright skipping notes. Furthermore, a higher level, oversight processor had not been implemented. The result of this was multiple strings would play the same note at the same time due to how the data was processed. Unfortunately, the project would also be put out of commission because an Electrostatic Discharge (ESD) would destroy the processors.

Out of love for the bass guitar, the concept of a MIDI controlled bass guitar was continued during the Fall 2024 semester. Other than sharing the primary concept, the new version of the project is substantially. The new version of the project is designed to be based around a real bass guitar. The idea was that a real bass guitar interfaced with hardware would sound closer to a real bass guitar. Given the massive change in functionality, development was planned to cover multiple semesters. During the fall semester the focus was on developing the appropriate firmware to work alongside the proposed mechanical platform. Key to this development was the master control firmware. This firmware features a robust algorithm for calculating the position of the note that should be played. Additionally the peripheral firmware was mostly developed as one was fully completed and the second partially implemented. Moreover, a basic CAD model of the bass guitar that be integrated was completed in the Fall.

Proposed Problem

The problem that this project seeks to solve is how to create a bass guitar that can be played via the Musical Instrument Digital Interface(MIDI). A previous version was created using a gantry system that moved along a string suspended along the frame. There were several issues with the prior design that limited its maximum performance. This project seeks to improve upon those issues by building a mechanical interface that interacts directly with an already assembled bass guitar. One of the problems to address was how to take the MIDI information and determine what string should be played. This is because stringed instruments have the same notes across multiple strings. However, the issue is that MIDI only provides a notes numerical value. Thus, an algorithm for solving the next string to play had to be solved with a master microprocessor that will make decisions, and then relay its decisions to the other devices. The majority of the firmware was completed in the prior semester.

Another problem that needed to be addressed was what would be the best mechanical interface for integrating the bass guitar. Many observed designs, including the prior project, choose to use a moving gantry that moves along the neck of the instrument. However, accounting the movement of the gantry, which causes delay, into note timing proved to be a difficult endeavour for a project that seeks to not need operators to make special command requests. The current proposed solution is to make use of solenoids to simplify the timing as well as command structure from the microprocessors. For this semester the biggest challenge is having enough time to troubleshoot bringing the entire project together. During this semester the PCBs and the mechanical platform are to be designed. Because of the nature of this project, these designs can only be effectively tested with assembling the entire project. Having enough time to adjust designs with enough lead time to finish by the end of the semester is the biggest concern.

Design Ethics and Environmental Impacts

All engineering/design endeavours have ethical implications that should be considered by designers. Failure to account for societal, economic, and environmental impacts could lead to the development of harmful technologies. For example the Gilbert U-238 Atomic Energy Lab was a “toy” science kit that was marketed to children in the 1950s. However, this product contained real uranium samples, a dangerous radioactive substance, that improper exposure could cause harmful health conditions such as cancer (Marsh). It is reckless products such as this that engineers, designers, and innovators seek to account for the impact of there designs prior to completing their pursuits. This is no different for the development of the MIDI controlled bass guitar project. It seeks to create and ethical and sustainable robotic interface that plays a bass guitar to display technical development.

One of the most obvious factors to account for is the environmental impact of producing the materials to be used. The international semiconductor industry, in 2023, consumed 1.49×10^{11} kWh of energy and produces 7.15×10^{7} tons of carbon dioxide emissions (Wang et al. 47). As a reference during the same year the global carbon dioxide produced was estimated 37.79×10^{9} tons (Ritchie and Max). While the semiconductor industry only represents 0.189% of the total emissions, seeking to lower carbon emissions in industry is important to begin to lower worldwide carbon emission as a whole. More concerning is that the semiconductor industry reported the use 5.51×10^{8} m^{3} , or 1.212 $x10^{11}$ gallons, of Ultrapure water (Wang et al. 47). This is considerable use of water even though there are an estimated 2.2×10^{9} people who lack access to safe drinking water (“WHO/UNICEF Joint”). It is important to note that Ultrapure water is almost 100% pure water that is created using strict refinement processes to remove impurities in the water (Murillo). Thus, the use of Ultrapure water adds more chemical and material demands to the already massive material use of the semiconductor industry. This is all shows that by merely incorporating semiconductor parts, which the project uses for nearly all electronic components, contributes to an industry that consumes massive environmental resources. Ways to mitigate the environmental impacts from using semiconductor components would be to purchase more components that are manufactured in the United States and European Union. These regions have stricter rules on waste and factory emissions leading to less harmful production.

Another factor to consider is that electronic waste, e-waste, that is generated should parts fail. According to UNITAR, in 2024 the global e-waste generated was 62 x 10^9 kilograms of waste of which only 22.3% was documented and disposed of properly ("Global E-Waste"). E-waste has arguable more detrimental effects than the waste produced by the semiconductor industry. This is because there is less accountability and tracking of where these products are being disposed. The project seeks to reduce factors in e-waste by using breakout boards to optically isolate power components from the more fragile integrated circuits. Moreover, in events of catastrophic failure, such as a board catching on fire, the breakout system means that only the affected board needs to be replaced. Should parts fail they should be properly recycled or disposed of to not contribute to e-waste. Improper disposal can lead to the chemicals from the printed circuit boards leaking into the environment and causing damage. Component failure is minimized by the design not causing over stressing of the components. These design decisions were made to account for long term stresses on the components as to extend the life span of all components in use. Thus, minimizing the potential e-waste that could be produced by this project.

Other waste products to consider are the bass guitar, 3d-printed parts, and the aluminum rails. The 3d-printed parts can be handled like similar plastic products. Most areas do not have dedicated 3d-print recycling units, and should therefore be disposed of in proper trash bins. As for the aluminum rails, most recycling plants have aluminum processing making these materials the easiest. While the bass guitar is predominately made of wood, it also has metal components that need to be removed before the wood can be burned or disposed. The majority of the hardware such as the strings, tuners, and electronics can be removed easily be removing the mounting hardware. However, the neck of the bass has metal frets and and truss rod that are glued into the neck that need to be removed. Additionally the body has a polyurethane coating that would need to be removed as to not leach into the environment. While there are additional steps to properly recycle certain components of the project, the majority can be readily handled without causing harmful damage the environment and local ecosystem.

Outside environmental impacts there are also societal impacts that could occur from developing this project. A big worry in the music industry is the current rise in AI generated music. There are a myriad of current AI tools that can be used to completely write and track songs (Anderson). However, the current consensus is that AI will not infringe on working musicians in live settings (Wieduwilt). This is in large part due the disconnect between the audience and tracked instruments. The MIDI bass guitar could potentially bridge the gap and see its use in place of real musicians. One scenario where real players could be displaced is in session work. Session work is where a musician is hired to be recorded playing a track/song that a producer or singer has created. Rather than having to pay to bring in a musician, producers may find it more convenient and cheaper to use the MIDI bass guitar instead. Similarly use cases in live settings may be feasible because of two factors. One being that the MIDI bass guitar would more closely match the sound of a real musician. Additionally, when compared to pre-tracked recordings, the MIDI bass guitar adds a level of novelty that live audiences may enjoy. Genres that are dominated by individual audiences, like pop and rap, have the audience focused on the artist. In these cases that MIDI bass guitar is much cheaper than the rate that a proficient live musician would charge, and audiences would likely take little note that the bass is not being played by a live musician.

Ethically speaking the potential of deriving musicians of opportunities to have an opportunity to take paid work would have two main impacts. First, musicians already struggle to make decent money with the national average salary in the United States is $44,855 (“Musician Salary”). While higher than the median annual income in the United States it does not quantify the real nature of how musicians struggle to make money. This is because salary and wage figures can only be taken from those who have declared that they are professional musicians by trade. In reality the number of local musicians who may take a small gig for little to no money is hard to gather. If this data could be quantified it would likely severely drop the perceived income that musicians. This ties into the second issue that the MIDI bass guitar could present. If the MIDI bass guitar were to be used and reproduced to take the place of musicians, it would then disincentivize working musicians to keep honing their craft as work would dry up. Downstream of this issue is that new players would be less incentivized to pickup and learn the skill of bass guitar to pursue a career in the music industry. Such an outcome would derive many of not only their work but also their passion.

While the prior observations have largely covered the potential concerns regarding the MIDI bass guitar project there are still benefits to this project. Although it is possible that the MIDI bass guitar could musicians to lose work, the reality is that the project still lacks in many areas. It is important to remember that music is a form of art, and with all art there is a certain beauty to be found in the unconventional. Take the example of Vincent Van Gogh’s famous Starry Night painting. The painting does not depict the one to one recreation of a view of the night sky. Instead the painting takes an expressive, vibrant approach that creates a stunning work of art that would not be possible if Gogh had followed realism. This is no different in the world of music. As Wielduwilt points out in his article,

"… While AI can generate music quickly, it can't replace the emotional depth and cultural understanding that humans bring to music." The MIDI bass guitar has the same limitations in the form of the mechanical platform and its control interface. With the mechanical platform the velocity of impact cannot be adjusted with the current revision. The result is that dynamic expression in playing cannot be recreated with the project. Moreover, certain techniques such as chords and slides cannot be replicated with the way the fret modules are setup. Even it the mechanical platform was perfect another major limitation lies with MIDI. Although MIDI is a powerful interface, it is still programmed within the context of a machine. The result is that MIDI still cannot recreate the subtle expressiveness of a human player.

As a whole, the MIDI bass guitar project was designed to simply create a robotic interface to play a bass guitar. The design considers the greater effects, environmental and societal, that could lead to unethical developments and impacts. Waste is minimized by design elements that limit the long term strain on components that could cause failure. Impact on musicians is limited by the inherent limitations of the project. As the platform was never designed nor envisioned to capture the human spirit in music. Instead the project was designed to show the application, and limitations, of robotic systems when applied to the actions that a human would normally conduct.

System Overview

As stated previously the majority of the firmware was developed in the previous semester. However, there was a significant change made to the peripherals that control the solenoids that are used to fret the notes. The original proposed design featured a peripheral for each string. This was to enable individual solenoid control for each string. Once the mock-ups of the frame were completed, it was apparent that such a design was not feasible with the hardware. This being due to the solenoids being too large to be able to place four together in the tight space. Thus it was proposed to change to a full fret bar model instead. There are two main advantages of using this new design. One is that it halves the needed solenoids which significantly reduces the cost of the project. Furthermore, it reduces the peripheral devices streamlining coding as well as potential errors.

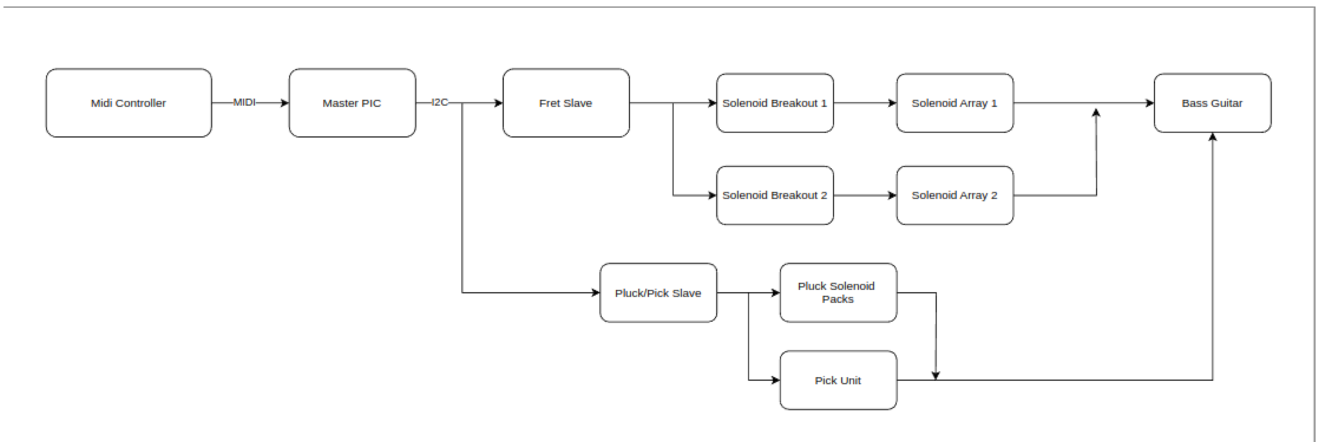


Figure 1: System Block Diagram

The new design is reflected in Figure 1 above. This takes the peripheral devices from five down to two. As previously mentioned the program for the fretting action was not fully completed during the prior semester. This actually proved to be a benefit as the new method changed the function of this program.

Fretting Program

The initial strategy for determining what fret to hold was to individually check each available note and then fire the correct solenoid. However, this strategy was inefficient and cumbersome to code. Thus, a more streamlined evaluation was sought in an effort to make evaluation time more even. The biggest challenge was how to derive a single bit based upon the MIDI note received. It was challenging to determine this when there are three ports that could be enabled. The solution derived relies on a loose successive approximation method. A subroutine was created that converts from hexadecimal to base one. Using this method the MIDI is calculated to be within a certain range based upon the limit of each port. Then the converter takes the hex and makes it a base one number that can be directly written to the port. The converter uses a successive approximation like flow which has a maximum of three evaluations before returning with the converted value. This method significantly reduced the time it took to complete an evaluation and also makes the lowest and highest MIDI note take a similar amount of cycles to complete.

Another aspect of this program that had to be completely rethought was how to determine the fret position now that this program covered all four strings of the bass guitar. The first decision was how to determine which string the command was intended to be executed on. This actually proved to be fairly simple as the string select byte that was created for the plucking peripheral already contained this data. Now the fretting unit features the same two byte structure of the plucking peripheral.

Table 1: Fret Slave I2C Bitmap

Fret Peripheral	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Byte1	M	P	P	P	P	P	P	P
Byte 2	X	X	X	X	G String	D String	A String	E String

The original byte one data is retained with the direct MIDI pitch being directly conveyed to the peripheral with the added change of using the seventh bit as a boolean mute control to convey a 9 or 8 command. With the second byte the string is conveyed with a boolean toggle for strings. Within the code this information is used to decrease the received pitch by string offset. This is because the bass guitar is tuned in fifths which means that each higher string is exactly five pitches above the last. By subtracting the offset of each string relative to the E string, the evaluation section of the code need only evaluate each MIDI as its E string equivalent.

PCB Design

One of the main priorities for this semester of development was to design working PCB’s for the project. All of last semester’s development was completed exclusively on a breadboard. This was because the feasibility of design elements for the project had not been fully determined. Due to the production of the PCBs being completed in China, it was paramount to finish the first revisions early in the semester to allow for multiple revisions to be completed and received during the semester. The first revision made use of the original peripheral scheme of five. Additionally there were two different breakout layouts. The first was for the solenoid array, and the second was exclusively for the pluck peripheral. This was because the pinout of the pluck peripheral was different from the other peripherals.

However, the first designs failed for a number of reasons. One of the most detrimental was that the footprint for the 44-pin PIC16F1789 was about half the size of the actual processor. Without these processors the motherboard was useless. Additionally, the footprint for the DIN-5 connector for the MIDI in was not on the board. This was because the first revision of the motherboard was designed over winter break. As a result remembering what had been done and what needed done was difficult.

The breakout boards also failed due to issues with the components that were used. Initially the solenoid switch was a SMD BJT due to their small package. However, there were issues with getting the BJT to fully saturate due to insufficient current. This resulted in the BJT developing a significant voltage, >5 volts, which caused them to burn out. The solution to this issue was to make the transition over TO-220 package MOSFETs. This provided two main advantages. One was that the MOSFETS only need to see a voltage differential from the source to the gate. As a result the current supplied by the opto-isolators becomes irreverent. Moreover, the use of TO-220 packages means that heatsinks can be mounted on the device. The new PCBs specifically are designed to allow for a mass heatsink to be mounted with ease.

While unfortunate that the first revision did not work for the breakouts, it proved to be a blessing in disguise. This was because there was an overlooked aspect of solenoids that the first revision did not account for. Solenoids function by inducing a field on the coil that then causes the metal pin to fire. As such solenoids, like motors, are an inductive device meaning that there are also field collapse effects. Because of their nature, inductors try to maintain the same current. When a current source is removed from a circuit, inductors attempt to maintain current results in a voltage spike. For high current coils this can result in several kilovolt instantaneous spikes that could potentially damage nearby circuitry. To prevent this a bleeder circuit was added to revision B in order to protect the MOSFETs from field collapse. This proved to be a highly beneficial addition as the new circuitry aided in solenoid performance. The bleeder circuit allows the field to collapse through a load which makes it collapse faster. This resulted in quicker solenoid retraction which leads to better responsiveness.

Regarding the motherboard, the second revision was designed after the changes were made to have only two peripheral devices. The new motherboard was made to reflect this new design which made it significantly more compact than the first board. Furthermore, the new board featured a new pinout for the pluck peripheral. This meant that the same breakout for the fretting array could also be used with the plucking solenoids. The only difference being how many parts needed to be installed. By making these changes the overall project became simpler.

Simplicity was actually a main focus of the project and having the PCBs match this was important. Moreover, the breakout and motherboard layout was designed intentionally to improve reliability and serviceability of the project. The last version of the bass guitar was put out of commission due to an ESD incident that killed the ICs that controlled the project. In observance of this the new design focused on isolation. This was accomplished by opto-isolating all digital outputs from the devices they controlled. Should an ESD event happen to a breakout only that board would be affected. Replacement is also easier and more economical with the multi-board layout. Damage or catastrophic failure is isolated to only the affected board. As a result only one board needs to be replaced rather than a large board that holds all the components. These design considerations reflect the mentality of making this project as reliable as possible.

Mechanical Design

See Appendix D for all mechanical drawings.

Mechanical design was the primary goal of this phase of development. Last semester the mechanical design only encompassed designing the mock model of the bass guitar that was ordered. Thus, the primary focus of this semester was to analyze the best way to integrate the proposed hardware. There were certain design considerations/philosophies that shaped the development of the platform. One was to make the frame as modular as possible to make design changes and further developments simpler. Another was to empathize simplicity over pure design functionality. Due to the massive development work necessary for this semester time is important. The risk of trying to develop complex mechanics to accomplish the task the best way possible was considered too great. Because if a design should fail, it would result in a lose of time that would setback the project significantly. Simplicity also reduces the manufacturing and assembly time needed.

One of the first decisions that had to be made was how to integrate the solenoids that hold the strings down. It had already been found that moderate force, 20 Newton plus, solenoids were needed. Due to the needed force it naturally resulted in larger solenoids. Once a CAD model of the solenoid had been moved into the main bass assembly it was apparent that the initial idea for fretting was not feasible. This was because the solenoids could not be mounted four wide on a fret. The bodies would interfere with each over resulting in no string pressed. An offset system was out of the question because the strings must be clamped as close to the fret as possible or otherwise the force will not be transferred effectively to the string. As a result the string would make a buzzing sound as it scrapes against the fret due to insufficient force holding the string to the fret.

An offset transfer mechanism was theorized to solve this problem. However, the complicity and potential force loss meant this idea was scraped. This is when a common tool for the guitar inspired a revelation. A capo is a simple but important tool used by guitarists to effectively change their guitars tuning without having to retune. It works by clamping across all the strings on a fret causing all the strings to match the tuning of that fret. Because of the muting unit, it was realized that the same concept could be used for the bass guitar project. The idea was to use two solenoids to press down a bar that spans the entire fret. This design has a multitude of benefits that solved the issues at hand. One is that the force is doubled as two solenoids are working together to clamp the string. Moreover, this meant that sufficient space and easier mounting could be achieved. It also significantly reduced the cost of the project because only half of the proposed solenoids were needed. This simple but innovative design would result in massive changes to the overall direction the project.

Upon determining how to fret the notes, the next design decision came down to what frame to use for mounting the solenoids. Because the bars would run along the frets, a frame with rails parallel to the fret board was modeled. The primary frame components were made using 2020 aluminum extrusions. This is because 2020 aluminum extrusions are light, cost effective, and modular making them ideal for a frame. The idea to run the main rails parallel to both sides of the fret board meant that the rails were at a slight angle. This was a result of the bass guitar’s fret board getting wider as it reached the bottom. In order to maintain this angle but also have a study frame, top and bottom mounting brackets were modeled and 3d printed. Thanks to the simplicity of the new fret design, a trapezoidal frame was achieved with stellar rigidity for the application.

Once the primary frame was modeled, work could begin on designing parts that would later be 3d printed to mount the remaining hardware. Special L brackets were designed to have mounting holes for the solenoids and the 2020 rails. The fret bar uses a matched radius to the fret board and was designed to allow for a foam insert. The foam, weather stripping, is necessary to let the string vibrate while also filling in the gaps to create a tighter seal on the string. Without this key detail the strings vibrate poorly and risk sliding under the bar. For ease of mounting, threaded inserts allow the solenoids to screw directly into the bar without additional hardware. As previously mentioned the rails expand as they progress, This necessitated a slightly modified bar length for each fret. Which did increase the time it took to model the project. Additionally, the frets become closer in spacing as pitch increases. Because of this, there were special brackets designed to create a staggered offset to fit all the solenoids. The inset design does limit overall force transfer. However, the majority of MIDI fed to the bass guitar will not make use of these higher frets. As such performance losses are seen as negligible.

The greater challenge was mounting the solenoids that pluck the strings. Initially the idea was to offset the solenoid at an angle similar to how a real player plucks the string. However, the issue was that a static setup would cause the solenoid to strike the string again which is undesirable. Ideally the same angled concept would be used to implement a cam geometry or other mechanism that caused the solenoid to slightly deflect on retraction. However, it was believed that this would take too long to develop. Instead the new solution used a direct mounted solenoid that lightly strikes the string. The solenoid is tuned to only fire just long enough to energize the coil to move the firing pin but not enough for it to stay extended. This results in the solenoid only having enough energy to hit the string to cause it to vibrate before bouncing off the string…. Since there are two solenoids for plucking it meant that offset mounts were required to keep the solenoids inline with the strings. If each solenoid was individually mounted it would have been simple to mount the solenoid directly over the string. However, for space reasons the pair of solenoids needed to be mounted as closely together as possible. As a result mounts that accounted for the 1.22 degree offset had to be created to accommodate each of the strings. This ensures proper alignment with the strings.

Comparatively, designing the mounts for the muting unit was straight forward and simple. The muting unit only requires one solenoid per string and gives greater flexibility on where to mount the unit. Furthermore, the alignment of the solenoid is not critical so long as a pad is large enough to make contact with the string. Because of this simplicity, the muting solenoids feature the same L mounting bracket and pad mounts. Although simple, the muting unit is critical for making the bass sound its best. The bar style fretting action means that the strings want to freely ring out when clamped quickly. With the muting units resolve this by only allowing the selected string to vibrate. This design feature was initially created to prevent the other strings from resonating with the note that was being played. However, it proved to be critical to when the design was changed.

MIDI Over Infrared

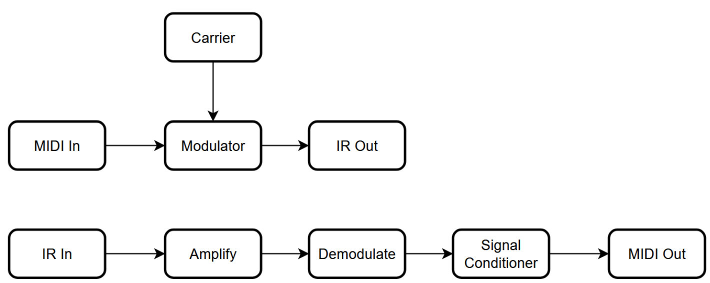


Figure 2: IR Block diagram

A secondary objective of this project was to create a system that sends midi optically. The purpose for this is so the MIDI Tesla coil can be used as previously when it was hardwired in the coil would send noise feedback through the line corrupting the MIDI signal. By optically isolating the MIDI signal the coil will not interfere with the incoming MIDI. Transmitting the MIDI over RF was considered but the coil has the potential to destroy that signal. The transmitter requires five volts, ground and a MIDI input to output modulated MIDI over infrared. The receiver requires five volts, ground, a modulated MIDI signal over infrared and outputs MIDI. By using infrared a MIDI instrument can be placed anywhere as long as the receiver is pointing towards the transmitter. The implementation of this system will be a single transmitter at the MIDI that uses a string of infrared LEDs along a wall/ceiling or as a sphere. Each MIDI device receiving via the infrared would then have to point the receiver at the source. On the transmitter the MIDI is being modulated with 1MHz before transmitting. The receiver will pick this signal up, amplify it and feed the signal into a PLL. The PLL ensures only data at 1MHz will make it to the MIDI output. This handles not sending signals from things like TV remotes to be passed through to the instrument. After the PLL the signal goes through an edge triggered inverter to produce a clean and usable signal to the MIDI out plug.

Infrared Schematics

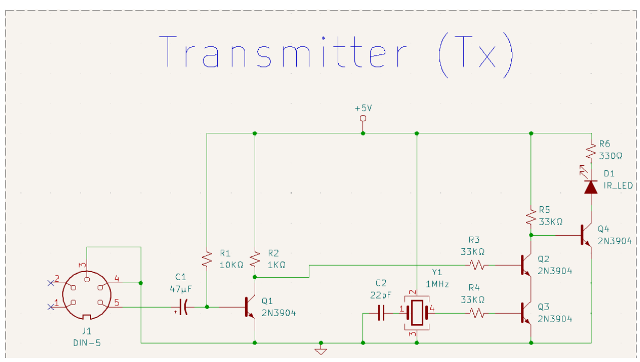


Figure 3: Tx Schematic

It is important to note R6 will have to be physically bypassed or removed on a new version when the infrared LED strip is being implemented as the strip has its own resistors on it.

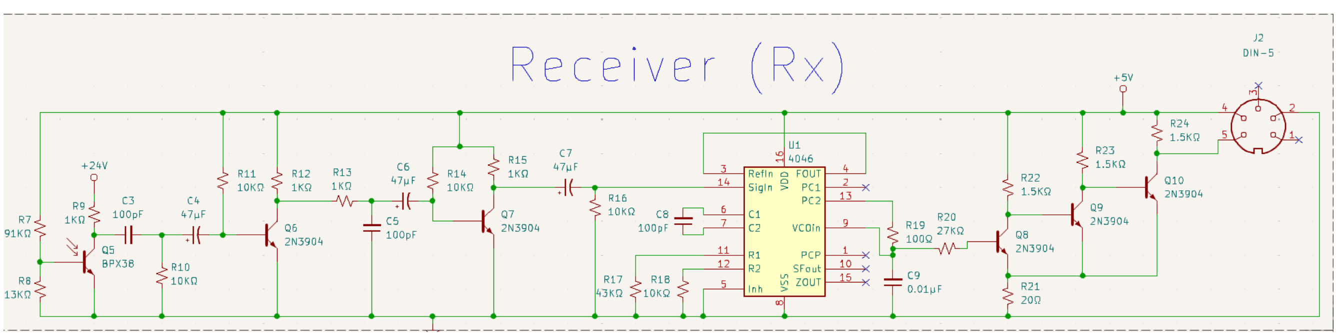


Figure 4: Rx Schematic

Infrared MIDI Measurements

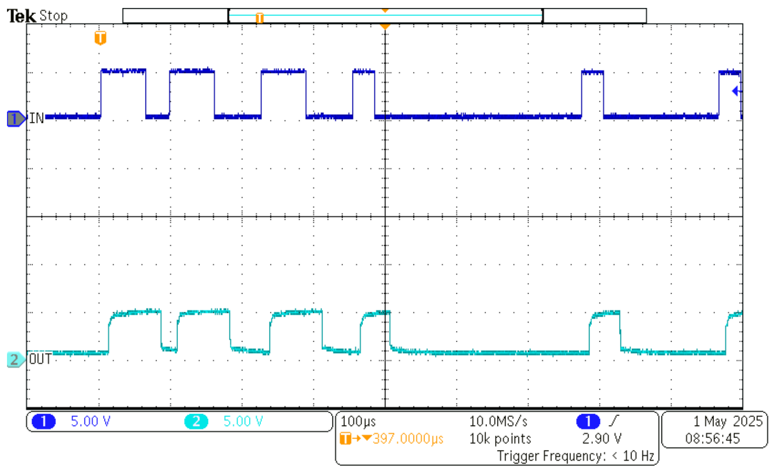


Figure 5: Demodulated MIDI to Original

This figure shows the MIDI going into the transmitter and the MIDI coming out of the receiver to be used by an instrument. There is a measured 15 microsecond delay which is about half a bit in relation to MIDI.

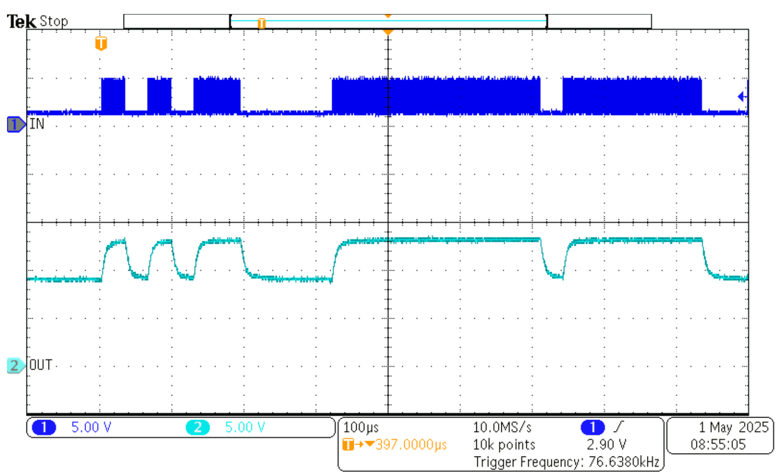


Figure 6: Modulated Signal and Demod

Observe how on the Rx signal the carrier is not seen. The carrier is being received but at a low level. This is why the signal must be amplified to be properly read by the PLL.

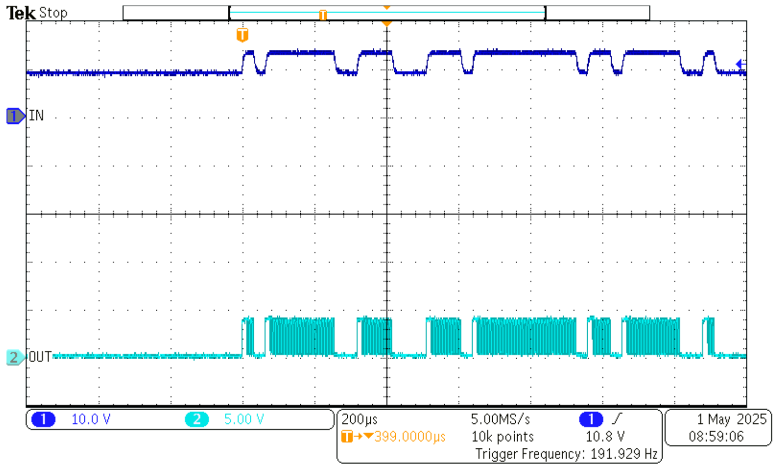


Figure 7: Post Amplification Capture

Shown here is the signal being received and the signal after being amplified. Post amplification the carrier signal is now readable for the PLL

Calibration

In order to have the desired operation of this project, there are required calibration steps to complete. The first and most important calibration is to tune the bass. Instruments, especially guitars and basses, can loose their tuning due to ambient changes and shifts in tension. Proper note pitch can only be assured if the bass is tuned before use. There are no tuning setting has been directly integrated into the firmware. As a result all tuning must be done manually and with system off. If the boards stay on, then the muting units will prevent the strings from ringing out properly which will reduce tuning accuracy. For tuning, download a free tuning app from your preferred app store. Then open the app and place the phone near the amplifier’s speaker. Ensure that the room is quiet and free from excess ambient noise as this will reduce accuracy. Then begin manually pluck the string and observe the note that appears on the tuner. Standard tuning for a bass guitar is, from lowest to highest pitch, E, A,D,G. Note that the diameter follows pitch meaning the thickest string is the E string whereas the thinnest string is the G string. The pitch observed on the tuner should generally be close to the desired pitch unless the tuner has been substantially moved or the strings have been changed. If the pitch is lower, referred to as flat, then turn the tuning peg for the associated string in a clockwise action to raise the pitch. Conversely turn the dial the opposite direction, counterclockwise, to lower the pitch if it is too high, which is referred to as sharp. Once all the strings have been tuned the system can be turned on. It is important to note that drop or irregular tunings are not recommended because it may result in inconsistent notes because the firmware recognizes the fifths relationship of standard bass tuning. It is possible to use lower tunings if all the strings are dropped to an equal pitch. Do note that the firmware features a lockout for notes outside normal tuning range. Thus, MIDI must still remain in the normal limits of a 4 string bass guitar in standard tuning.

For Firmware calibration see Bass Guitar Report page 22 Fall 2024 by Owen Fujii

Trouble Shooting

For Firmware troubleshooting see Bass Guitar Report pages 23-31 Fall 2024 by Owen Fujii.

Outside of firmware faults or issues, hardware alignment and mechanical issues can also cause issues with system operation. The frame and solenoids can have issues that can cause undesirable operation. Refer to figure 39 for troubleshooting flowchart to follow along with this section. In general, issues with the system will be related to frame alignment. This is because the solenoids, particularly those that fret the strings, have a great deal of force that push against the frame. Overtime this could lead to issues with the frame moving away from its initial position. Should the frame move away from the fret board issues with contact may occur. For the fret bars, this will either cause no contact with the fret leading to a muted sound without a change in pitch, or the more likely being that a buzzing sound from the string occurs. Fret buzz as it is called can happen for a number of reasons. In the case of a moved frame, it is due to insufficient pressure on the string causing it not to fully be fretted. If this is the case, then the buzzing will occur at the fret that the bar is at. However, if the fret buzz occurs at a later fret, then this is the result of the bass itself being out of alignment. The neck of the bass guitar is under high tension, over 140 lbs, which can cause its shape to change as the wood of the neck shift. This is why occasional maintenance on the bass is necessary. Because of how many steps are in this process, it is recommended to find a tutorial video on how to do bass guitar setup.

The converse issue to the frame being pushed away would be frame sag. If the frame begins to sag it can cause the fret bars to always make contact with the strings even when they are retracted. Should this occur, the open strings will not ring out; instead causing a muted sound. Similarly, the alignment of the fret bars in relation to its solenoids is important for proper retraction. If the bar is misaligned, then one or both of the solenoids may begin to bind causing insufficient retraction. As a result, the fret bar may sit on the strings causing the same muted problem.

For the plucking units, the main issues that can occur are misalignment. This is because the positioning of the solenoids are important to proper contact. Due to spacing the brackets are only held in place by one screw. This causes a pivot point where the force from the solenoids wants to pivot the bracket due to the offset design. Over time the solenoids may deflect from the string enough to not make contact. Moreover, if just enough overlap is made, the tip of the solenoid will follow through the string, but will get caught on retraction. This is why the brackets feature a slotted design as this gives the most adjustment.

While the other solenoids are only fired for a small portion of time, The muting unit, currently, is activated the entire time the power is applied. This is because the muting unit should be active when the others are not. However, this leads to them getting significantly hotter than the other solenoids. When the coils get hot the field is less efficient which causes the solenoids to fire slower. If this starts to happen, turn off the power immediately. Then wait for the solenoids to cool before beginning again. For this reason it is not recommended to leave the project on when it is not actively playing music.

Should the solenoids be allowed to continue to heat, it could result in coil failure. In a worst case scenario, the coil could begin to heat to the point of melting which is a fire risk with a wood body directly below. Future revisions of the code should implement a time out feature on the muting unit to prevent heat saturation. With the current revision, these solenoids are the most likely to fail due to stress overtime.

As a whole, this project was generally designed to need as little maintenance as possible. The firmware was designed to give the operator feedback on the current status on the processes and statuses of the microprocessors. Moreover, the mechanical platform was designed to be as ridged as possible with the hardware available. However, with sizing constraints, certain designs may be less robust. Thus, occasional maintenance and troubleshooting will be needed to maintain the full functionality of the project.

Cost Analysis

Refer to Figure 40 for a complete breakdown of the costs of the project.

This version of the project for all materials was estimated to cost $1,233.84. Compared to last semester, this cost is less than the estimate of $1,530.19 for the electronics alone. The decrease in cost is attributed to the large reduction in solenoids. Initially, the proposed solenoids count needed would total 92 solenoids. Changing to the fret bar model resulted in only 52 solenoids needed for the new system. This almost halved the total cost of the electronics needed to complete the project.

Regarding the hardware, the prolific usage of 3d-printed parts significantly reduced the overall cost of the hardware. All the parts needed for the project can be achieved with 1kg of PETG. The low cost, $13.99 for a spool, is significantly cheaper than predicted. While numerous, the plastic needed to produce each part was actually very small. This goes against the prediction of having an increased mechanical cost due to printed parts. Moreover, the use of 2020 aluminum extrusion meant that the frame cost was manageable at $105.75. Still the cost of this project was significantly greater than the other projects. However, this cost is justified as stated in the last report, “Despite this cost, the end product will present a valuable example of how electronics can be integrated to interface with systems that were designed around humans” (Fujii, 2024).

Engineering Log

Table 2: Owen's Time Distribution

Hours Distribution	
PCB Design	15:00:00
CAD Design	46:31:00
Assembly	38:35:00
Coding	17:50:00.00
Report	19:00:00.00
Total	136:56:00.00

The table above shows the time distribution for the general categories of work. The majority of time this semester was focused on designing the necessary parts in CAD. A close second is naturally the associated assembly of the project which took another large chunk of time. Interesting the time it took to design the PCBs was lower than expected. This is likely because the boards themselves were relatively simple with enough space for adequate trace spacing. However, it is still very fascinating as two revisions of PCBs where completed this semester. The coding and report times were within the expected range.

Overall I believe that time was well spent this semester. I was surprised with how time was spent on this project especially given that I was gone for an entire week. Trying to complete the project within the time allotted was a monumental task given how much mechanical work was needed. It did pay off to focus on the PCBs at the beginning of the semester to allow for two design revisions to be made during the semester. The manufacturing time for all the 3d printed parts was longer than expected. This was a combination of both machine limits and the sheer amount of parts that needed to be printed. Furthermore, full troubleshooting was limited because development reached a point, where every part of the project was dependent on the completion of another step. This did slow the overall process because work could not always be completed simultaneous. Despite these setbacks I was pleased to have completed a working demo by the end of the semester.

Justin’s Log

Justin Week One Hours				
Date	Clock-In	Clock-Out	Time(hrs)	Description
4/7/2025	10:00	11:00	1	breadboarding
4/7/2025	13:15	14:30	1.25	git branch organization
4/8/2025	13:00	14:30	1.5	finished breadboarding
4/9/2025	10:00	12:00	2	figuring out midi
4/9/2025	13:15	15:00	1.75	I2C troubleshooting
4/10/2025	10:00	11:00	1	I2C troubleshooting
4/10/2025	13:15	14:30	1.25	I2C troubleshooting
4/11/2025	13:00	14:30	1.5	I2C troubleshooting
4/11/2025	10:00	12:00	2	I2C troubleshooting
4/12/2025	13:15	15:00	1.75	I2C troubleshooting
4/12/2025	10:00	11:00	1	I2C troubleshooting
4/13/2025	13:15	14:30	1.25	I2C troubleshooting
4/13/2025	13:00	14:30	1.5	I2C troubleshooting
		Total:	18.75	

Table 3: Justin's W1 Hours

For the first week I bread-boarded up the master and fret slave circuit to begin finishing the fret program. Upon programming I ran into I2C issues and spent the rest of the week learning I2C to figure out what needed to be fixed.

Justin Week Two Hours				
Date	Clock-In	Clock-Out	Time(hrs)	Description
4/14/2025	9:00	11:40	2.667	I2C troubleshooting
4/14/2025	12:40	14:30	1.833	I2C troubleshooting
4/15/2025	9:00	11:45	2.75	I2C troubleshooting
4/15/2025	13:15	14:40	1.417	I2C troubleshooting
4/16/2025	9:00	11:40	2.667	Finalizing Fret Slave
4/16/2025	13:00	16:00	3	Finalizing Fret Slave
4/17/2025	9:15	11:40	2.417	Investigationg MIDI over IR
4/17/2025	13:00	15:30	2.5	Investigationg MIDI over IR
4/18/2025	9:00	11:40	2.667	Investigationg MIDI over IR
4/18/2025	13:00	15:00	2	Investigationg MIDI over IR
		Total:	23.917	

Table 4: Justin's W2 Hours

For the first part of this week I worked on I2C, my problem with the fret program was regarding the clock stretching. After completing the fret slave code I began designing a system for transmitting MIDI over infrared.

Justin Week Three Hours				
Date	Clock-In	Clock-Out	Time(hrs)	Description
4/21/2025	9:00	11:40	2.667	Investigationg MIDI over IR
4/21/2025	12:40	14:30	1.833	Investigationg MIDI over IR
4/22/2025	9:00	11:45	2.75	Investigationg MIDI over IR
4/22/2025	13:15	14:40	1.417	Investigationg MIDI over IR
4/23/2025	9:00	11:40	2.667	Investigationg MIDI over IR
4/23/2025	13:00	16:00	3	Investigationg MIDI over IR
4/24/2025	9:15	11:40	2.417	Investigationg MIDI over IR
4/24/2025	13:00	15:30	2.5	Soldering Breakout Boards
4/25/2025	9:00	11:40	2.667	Soldering Breakout Boards
4/25/2025	13:00	15:00	2	Soldering Breakout Boards
		Total:	23.917	

Table 5: Justin's W3 Hours

I spent the better part of this week working on the IR system. The last part of the week I spent soldering up the solenoid breakout boards.

Justin Week Four Hours				
Date	Clock-In	Clock-Out	Time(hrs)	Description
4/28/2025	9:30	11:40	2.167	Investigationg MIDI over IR
4/28/2025	13:00	14:40	1.667	Investigationg MIDI over IR
4/29/2025	9:00	11:40	2.667	Investigationg MIDI over IR
4/29/2025	12:45	14:40	1.917	Investigationg MIDI over IR
4/30/2025	9:15	11:40	2.417	Investigationg MIDI over IR
4/30/2025	12:30	16:00	3.5	Write up for MIDI over IR
5/1/2025	9:15	11:20	2.083	Write up for MIDI over IR
5/1/2025	13:00	15:00	2	Write up for fret slave
5/2/2025	9:10	11:40	2.5	Write up for MIDI over IR
5/2/2025	13:00	15:00	2	Write up for fret slave
		Total:	22.917	

Table 6: Justin's W4 Hours

This week i finished up the MIDI over IR system and began write up for the report

Justin Week Five Hours				
Date	Clock-In	Clock-Out	Time(hrs)	Description
5/5/2025	9:30	11:40	2.167	Write up for MIDI over IR
5/5/2025	13:00	14:40	1.667	Soldering Amplifier Board
5/6/2025	9:00	11:40	2.667	Writing Presentation
5/6/2025	12:45	14:50	2.083	Writing Presentation
5/7/2025	0:00	0:00	0.000	
5/7/2025	13:00	14:50	1.833333333	Engineering Log Write Up
5/8/2025			0.000	
5/8/2025			0	
		Total:	10.417	

Table 7: Justin's W5 Hours

This week I worked on writing and the presentation.

Overall I think I managed my time well, I think I could have completed the MIDI over IR a few days quicker if I had been more streamlined in my troubleshooting problems with the design. In particular I spent a while on how to achieve a high frequency carrier to be picked up on the receiver. My original problem with this is that I was using a phototransistor with no base pin so it could not be properly biased. After obtaining a three pin phototransistor I was able to bais it to pick up the signal better. At this point though even with amplification the receiver struggled to routinely pick up the carrier. The next thing I did was put more voltage on the collector to allow more current and this is ultimately what was needed to pick up the carrier. Considering Owen was gone the entire first week I started working I feel like I made a good pace. Had Owen been present I probably would have figured out the I2C quicker but the problem was still solved.

Conclusion

The continued development of this project has resulted in a sufficient proof of concept for the proposed design. From its inception, this project has tried to be ambitious in its desire to closely replicate the sound of a human bass player. In terms of ground covered, this semester achieved more development than the prior semester. Last semester was primarily focused on developing the firmware for the master processor. Because of the algorithm that was needed, the time it took to develop the program, and troubleshoot the serial communications took the majority of the semester. Which is why the last semester only completed two programs, and the equivalent model of the bass guitar.

Contrast that development with this semester. This semester achieved the completion of two revisions of PCBs, a near complete CAD model for the hardware, partial assembly of the hardware, and a proof of concept for MIDI over IR. Of course the increased development was aided by the increased time available this semester; as well as having two people work on the project. However, it was the focus on trying to deliver a working project by the end of the semester that kept the project moving. Having a timeline and focused development meant that designs were completed faster and more thoroughly. For instance, fewer design revisions to the 3d printed parts were needed because the CAD representations of the hardware were highly accurate. Needing fewer revisions was highly important to saving print time that would otherwise have led to the project being significantly delayed.

That said, the project did not fully achieve the full goals that were desired. The primary goal was to have a fully complete and assembled project by the end of the semester. However, in reality only part of the project could be completed. A major contributor to this was underestimating assembly, design, and manufacturing time necessary to complete the project. For example, print times for batches of parts could take up to 3-4 hours. This bottle neck resulted in the choice to bring a personal 3d printer designed to print faster to achieve completing the necessary parts by the end of the semester, yet even with this printer it was not possible to have all the parts printed by the end of this semester. There was simply too many parts that needed to be designed, printed, and test fitted before the semester ended. So while significant progress was made, it was not enough to overcome all the needs to achieve the primary goal of the semester.

All in all, this project had progressed to a satisfactory state by the end of the semester. While not complete, the work completed got the project very close to completion. Moreover, even in its unfinished state, the demo completed by the project showed that the concept of the project was successful. The timing of the system worked superbly in being highly responsive to the incoming data transmission. Achieving this result was a testament to the careful design decisions that were made all throughout the project. Once this project is fully completed, it shall fully demonstrate the wide skill diversity learned throughout the program. A demonstrate that is the proud achievement of Justin and myself.

Recommendations

Unfortunately, the full assemble to the project was not completed before the end of the semester. This was largely due to constraints of waiting for parts to be printed or arrive in shipping. As such it is necessary to complete final assembly of the project. Included in this is the need to have a CAD models made for higher fret bar mounts, and the full electronics cases for DIN rail mounting. Once the project has been mounted to a fixed chassis, final troubleshooting of the firmware will need to be completed. This is because the fret program was not fully debugged, because it required the full platform to be built. Completing the above recommendations will result in the project having been completed to a workable state.

Outside of finishing the current revision of the project, there are also some improvements to the project that would improve functionality and reliability. One change would be to add a crystal to the PCB of the motherboard's master processor. The master PIC16F1788 currently runs on the internal oscillator at 32Mhz. While the power and ground are isolated, power bumps or noise can cause the processor to crash which can lead to lost MIDI data. This is why it is suggested to use a crystal to improve reliability of the master. Another important change is to implement a time out feature for the pluck slave to release its muting units after no data has been received. Making this change will increase the reliability of the project and reduce stress on the solenoids.

A more significant recommended development is to revise the current plucking unit. The current configuration was designed to be simple and effective to strike the strings. However, the tonality of that this configuration is subpar. It produces a more shrill, abrasive sound than desirable. The new design should emphasize replicating the soft “bum-bum” sound that the bass guitar produces with finger style playing. This will require a more robust mounting unit, but should decrease maintenance needs. Creating this improved unit will bring the project into the intended design features of the project.

Appendix A: Citations

Anderson, Jon. "Will Artificial Intelligence Wreck the Music Industry?" *Two Story Melody*, 23 Jan. 2023, https://twostorymelody.com/artificial-intelligence-in-the-music-industry/. Accessed 28 Mar. 2025.

"Global E-Waste Monitor 2024: Electronic Waste Rising Five Times Faster Than Documented E-Waste Recycling." *UNITAR*, 2024, https://unitar.org/about/news-stories/press/global-e-waste-monitor-2024-electronic-waste-rising-five-times-faster-documented-e-waste-recycling. Accessed 28 Mar. 2025.

Fujii, Owen. "Bass Guitar Project: A MIDI Controlled Robotic Bass Guitar". *Idaho State University,* 09 Dec. 2024.

Marsh, Allison. "Fun-and Uranium-for the Whole Family in This 1950s Science Kit." *IEEE*, 31 Jan. 2020, https://spectrum.ieee.org/fun-and-uranium-for-the-whole-family-in-this-1950s-science-kit. Accessed 28 Mar. 2025.

Murillo, Homar. "What is Ultrapure Water?" *chemicals.co.uk*, 27 July 2022, https://www.chemicals.co.uk/blog/what-is-ultrapure-water. Accessed 28 Mar. 2025.

Richie, Hannah and Roser, Max. "CO_2 Emissions." *Our World in Data*, Jan. 2024, https://ourworldindata.org/co2-emissions. Accessed 28 Mar. 2025.

"WHO/UNICEF Joint Monitoring Program for Water Supply, Sanitation and Hygiene (JMP) – Progress on Household Drinking Water, Sanitation and Hygiene 2000-2022: Special Focus on Gender." *United Nations*, 02 July 2023, https://www.unwater.org/publications/who/unicef-joint-monitoring-program-update-report-2023. Accessed 28 Mar. 2025.

Wang, Qi, et al. "Environmental Data and Facts in the Semiconductor Manufacturing Industry: An Unexpected High Water and Energy Consumption Situation." Water Cycle, vol. 4, 2023, pp. 47–54, www.sciencedirect.com/science/article/pii/S2666445323000041, https://doi.org/10.1016/j.watcyc.

Wieduwilt, Christopher. "Will AI Replace Musicians? a Look into the Future." *The AI Musicperneur,* 03 Mar. 2025, https://www.aimusicpreneur.com/ai-music-faq/will-ai-replace-musicians/. Accessed 28 Mar. 2025.

Appendix B: Mechanical Drawing

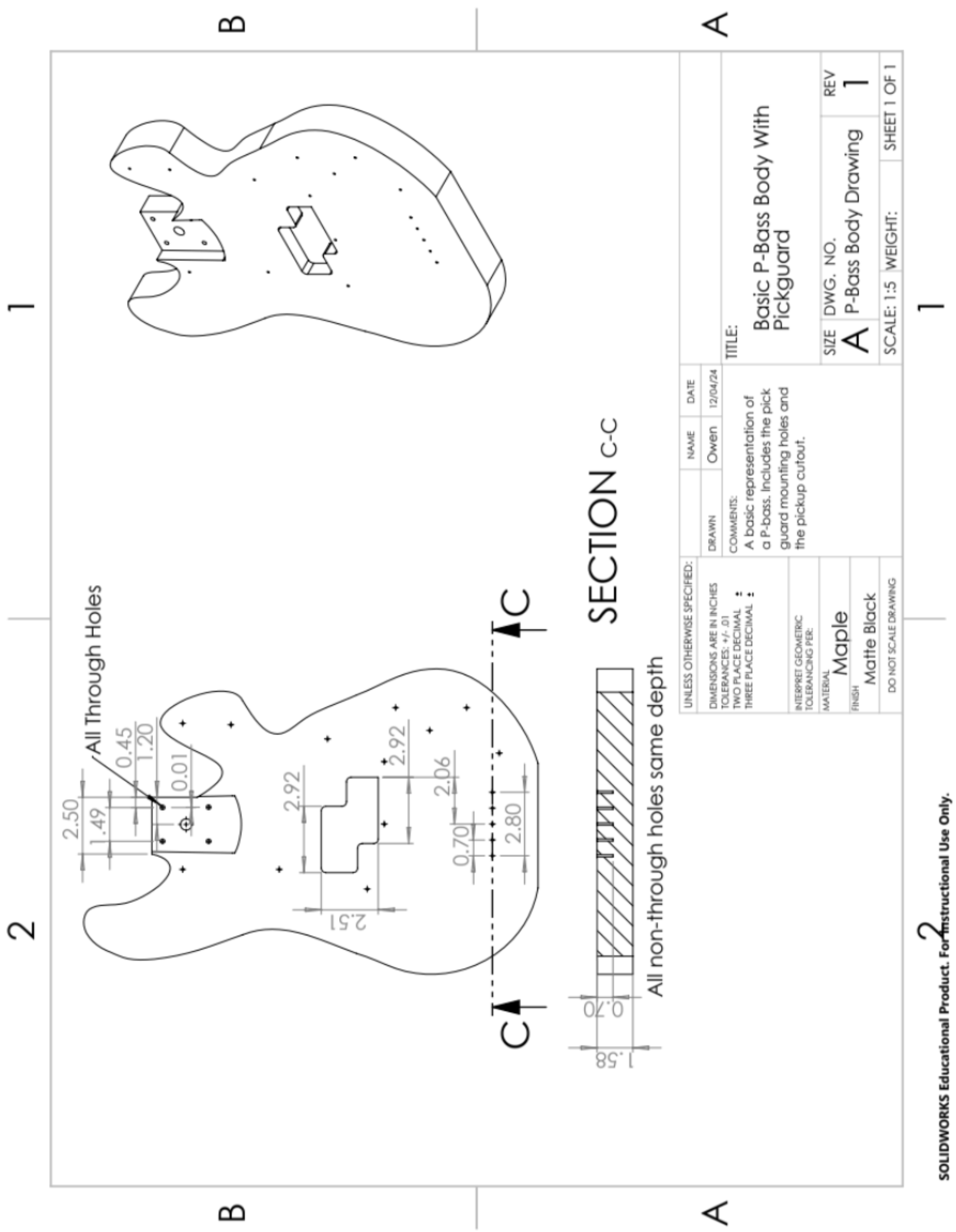


Figure 8: P-Bass Model

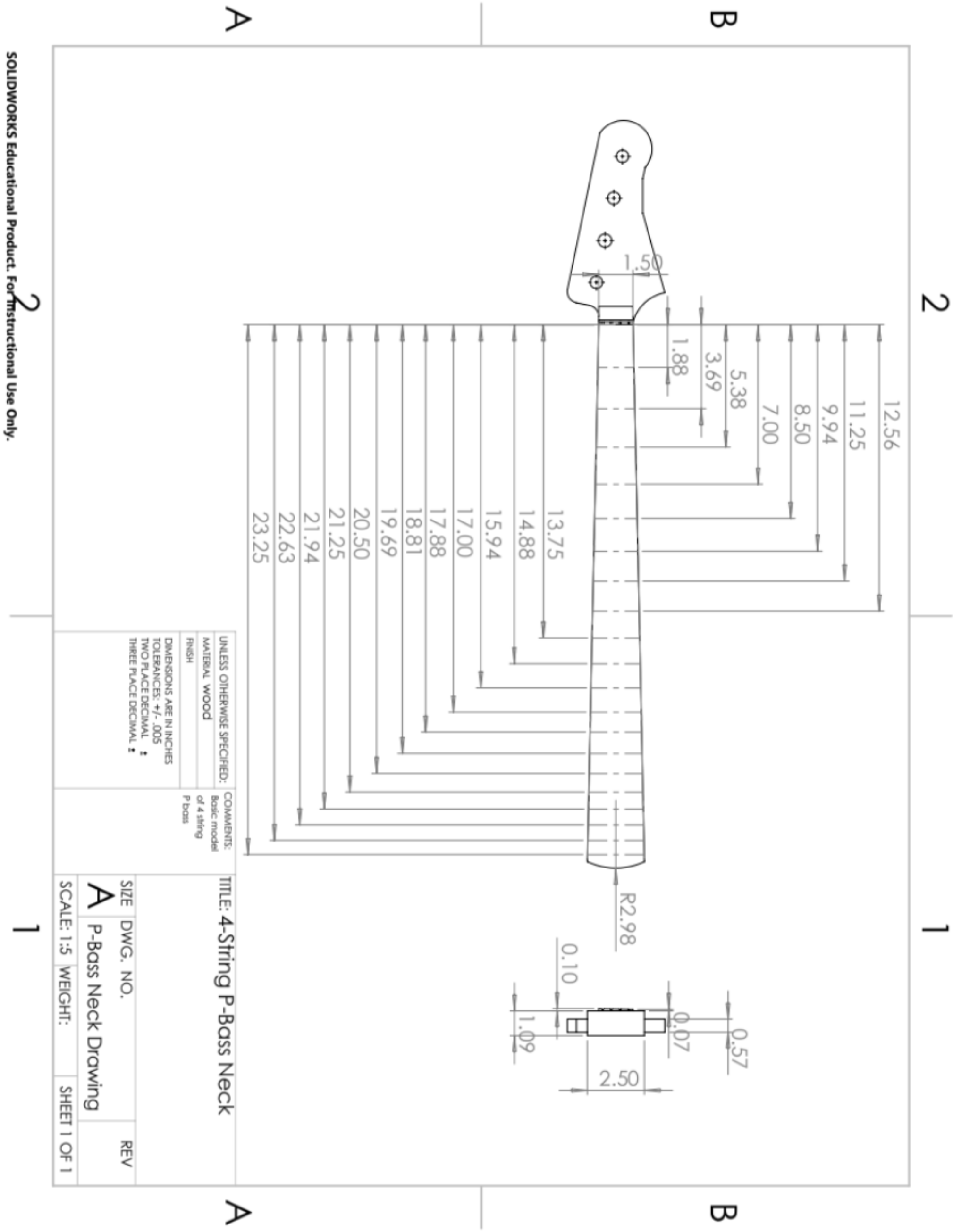


Figure 9: P-Bass Neck Model

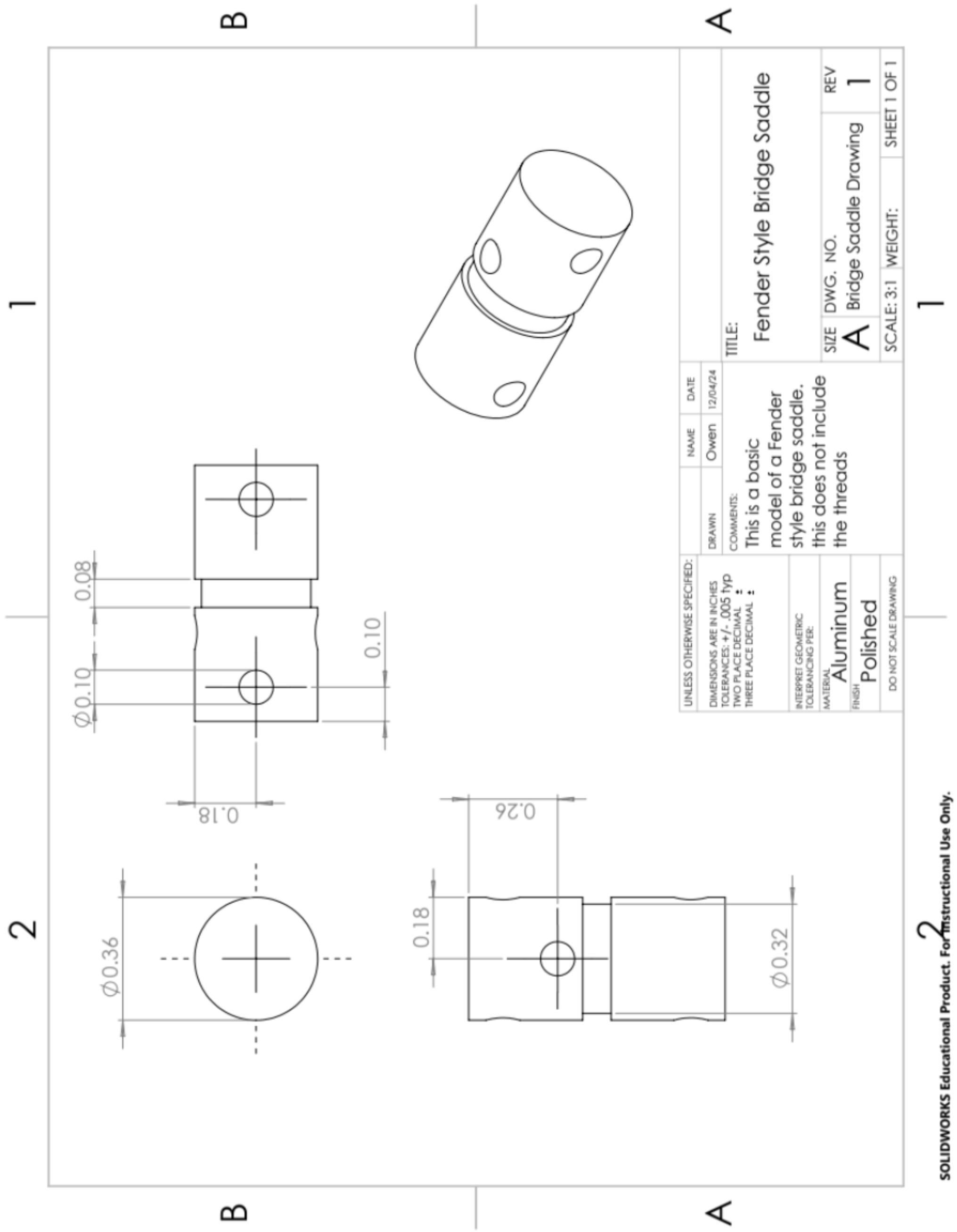


Figure 10: Bridge Saddle Model

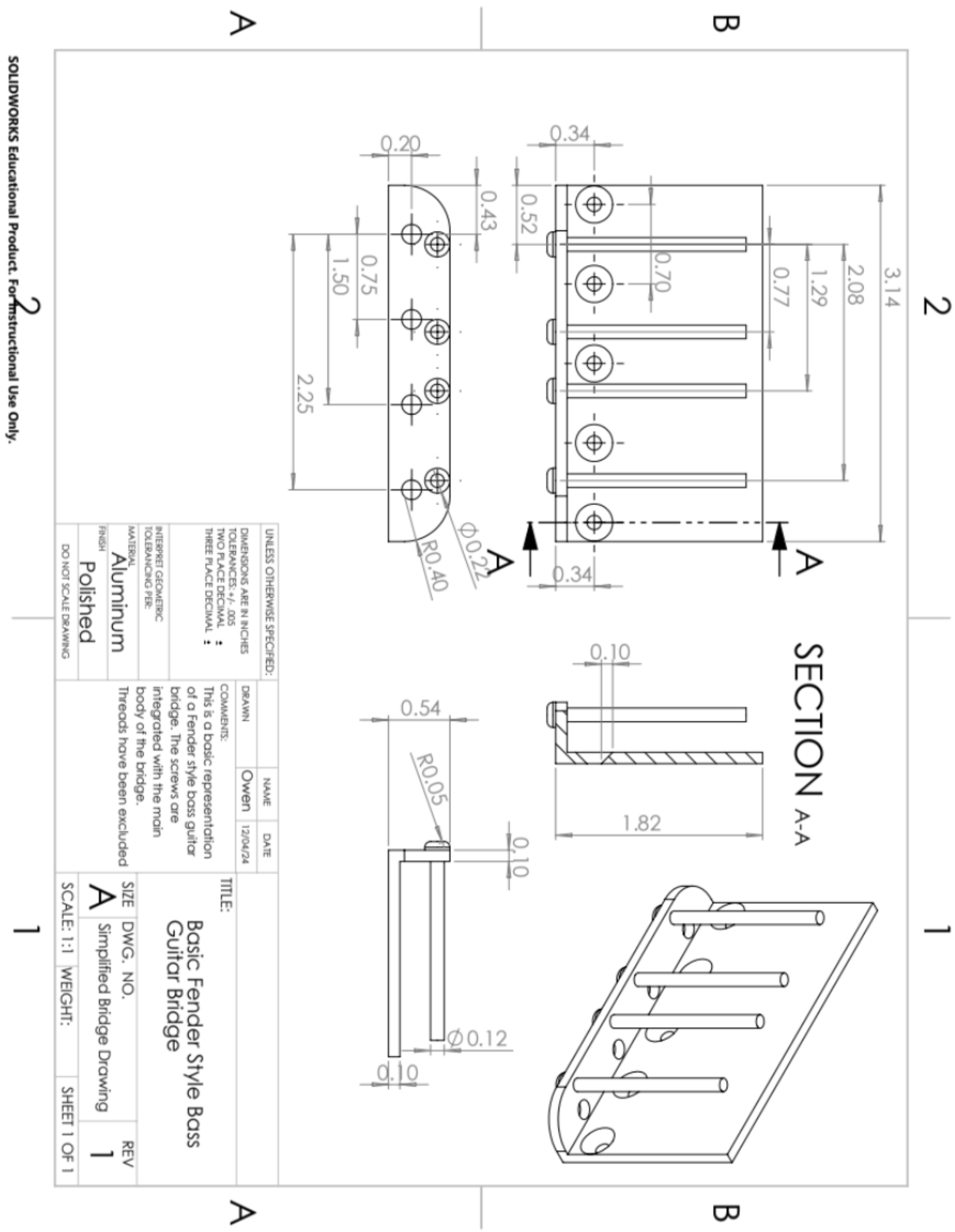


Figure 11: Simplified Bridge Model

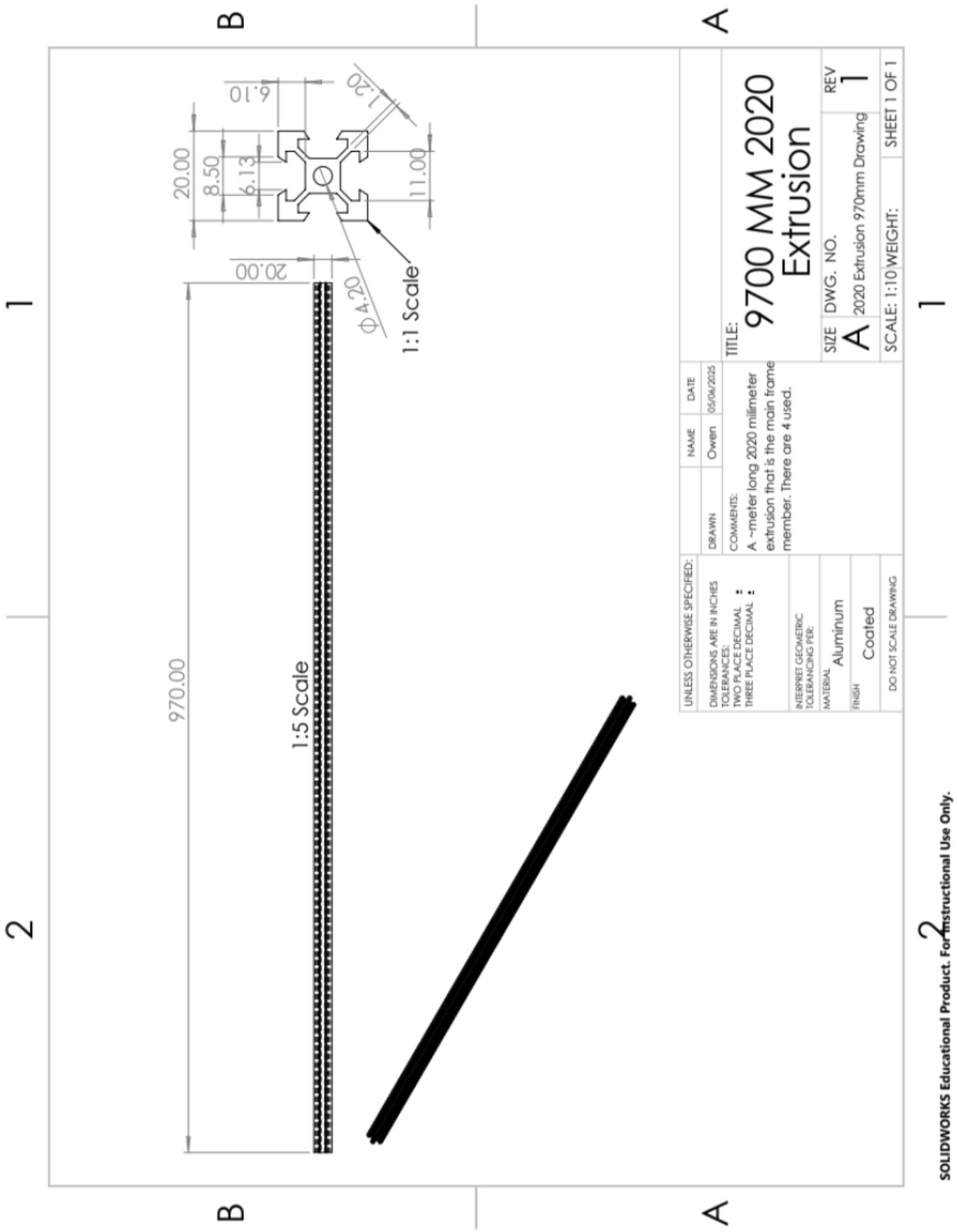


Figure 12: 2020 Long Aluminum Extrusion

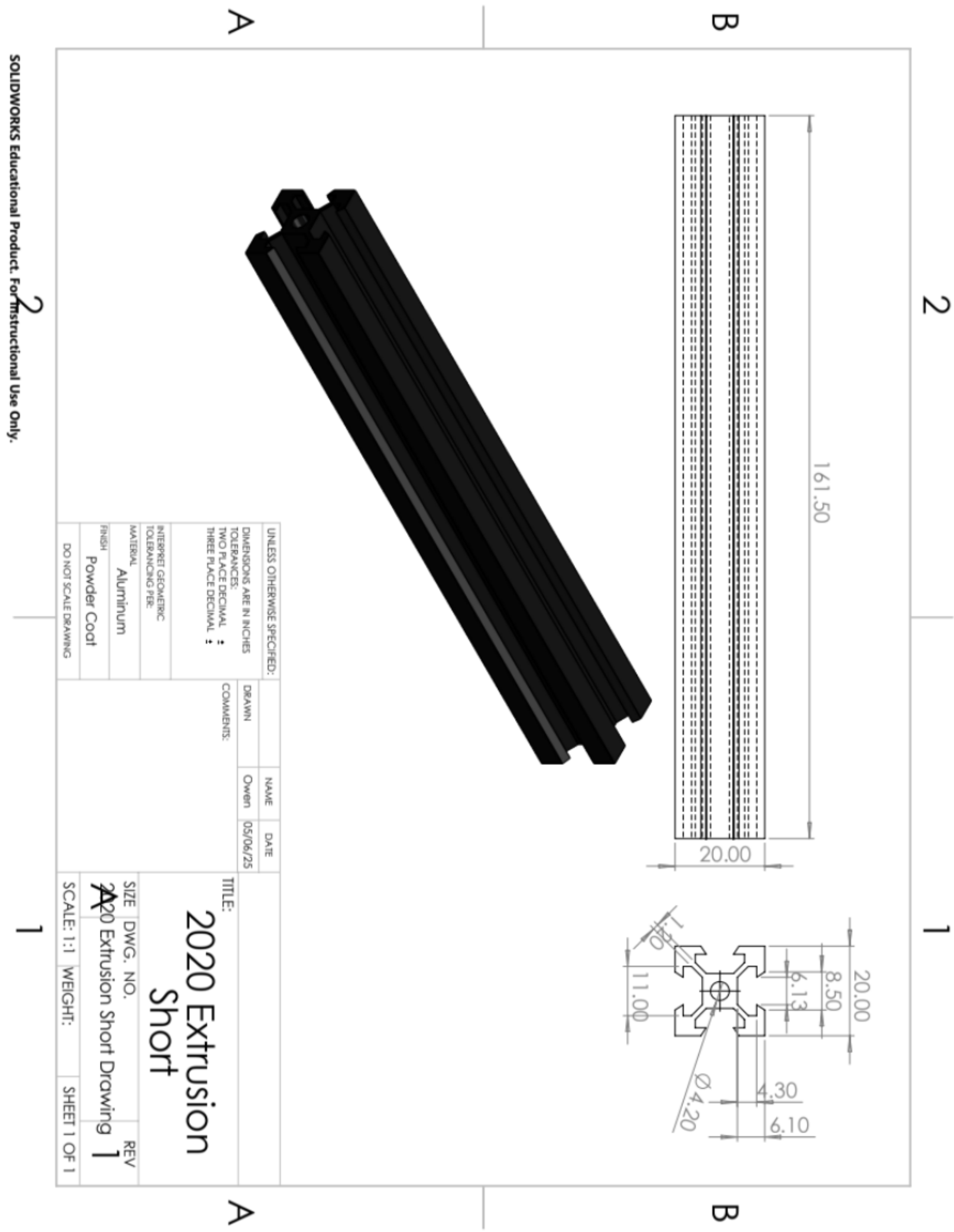


Figure 13: 2020 Short Aluminum Extrusion

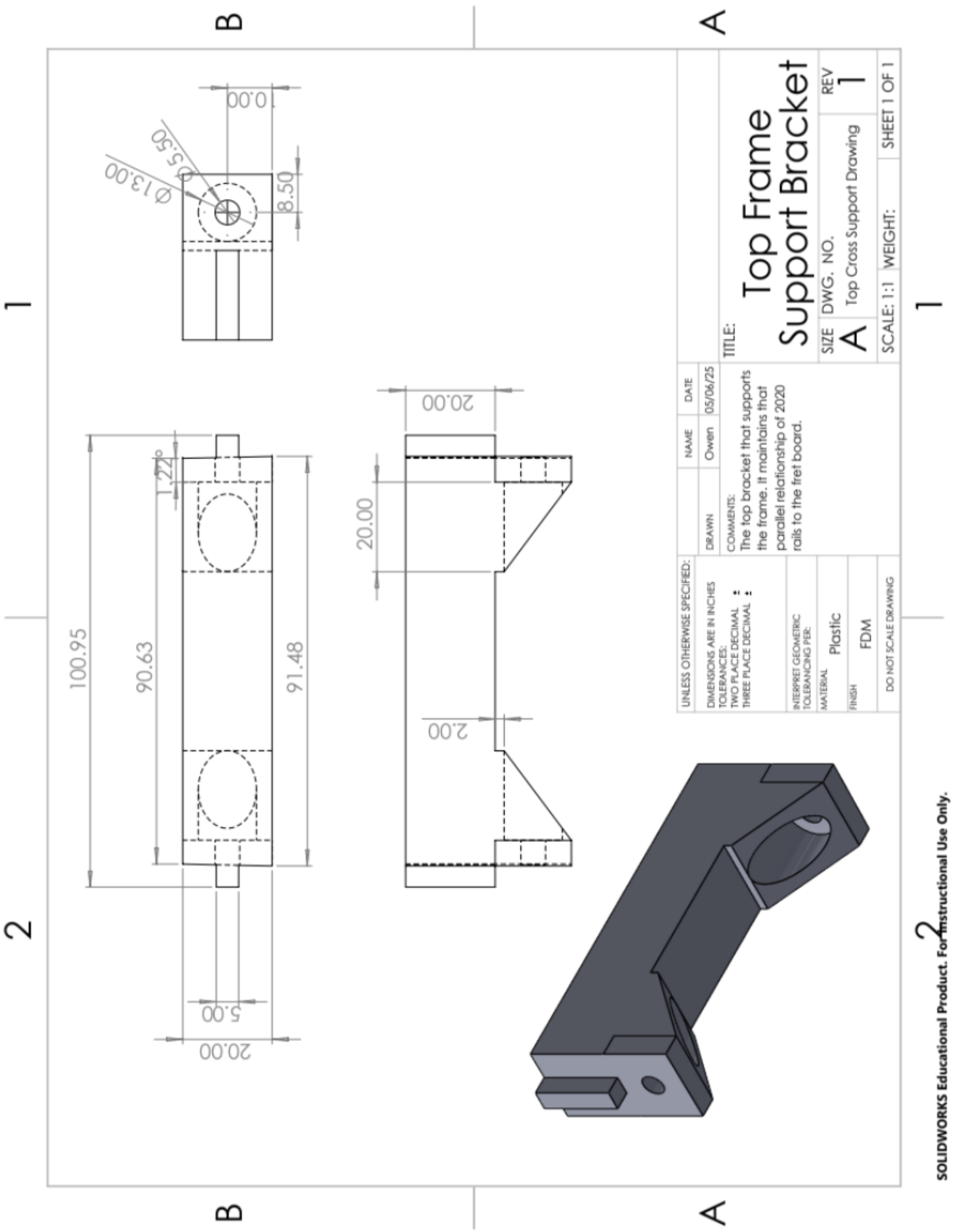


Figure 14: Top Support Bracket Model

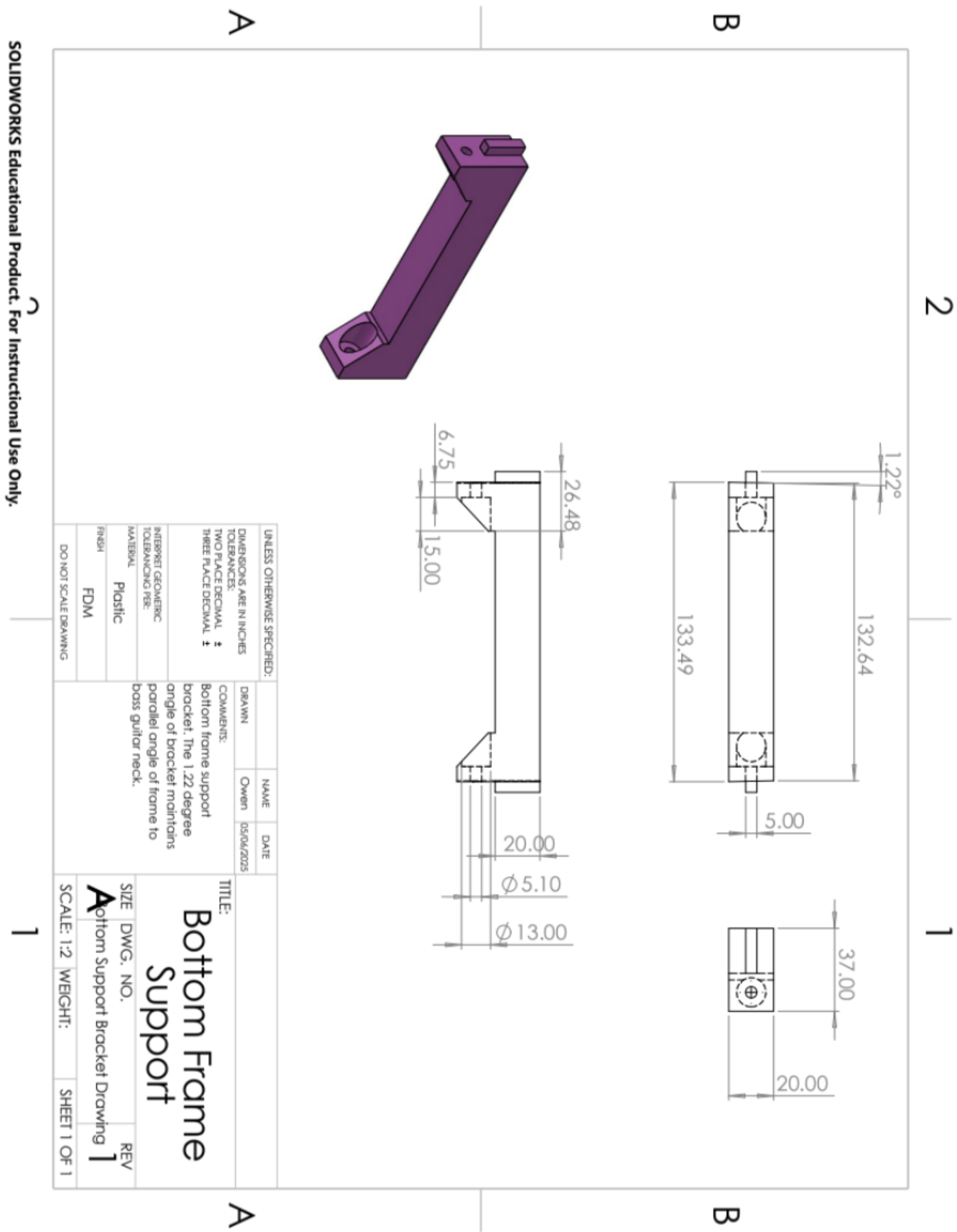


Figure 15: Bottom Support Bracket Model

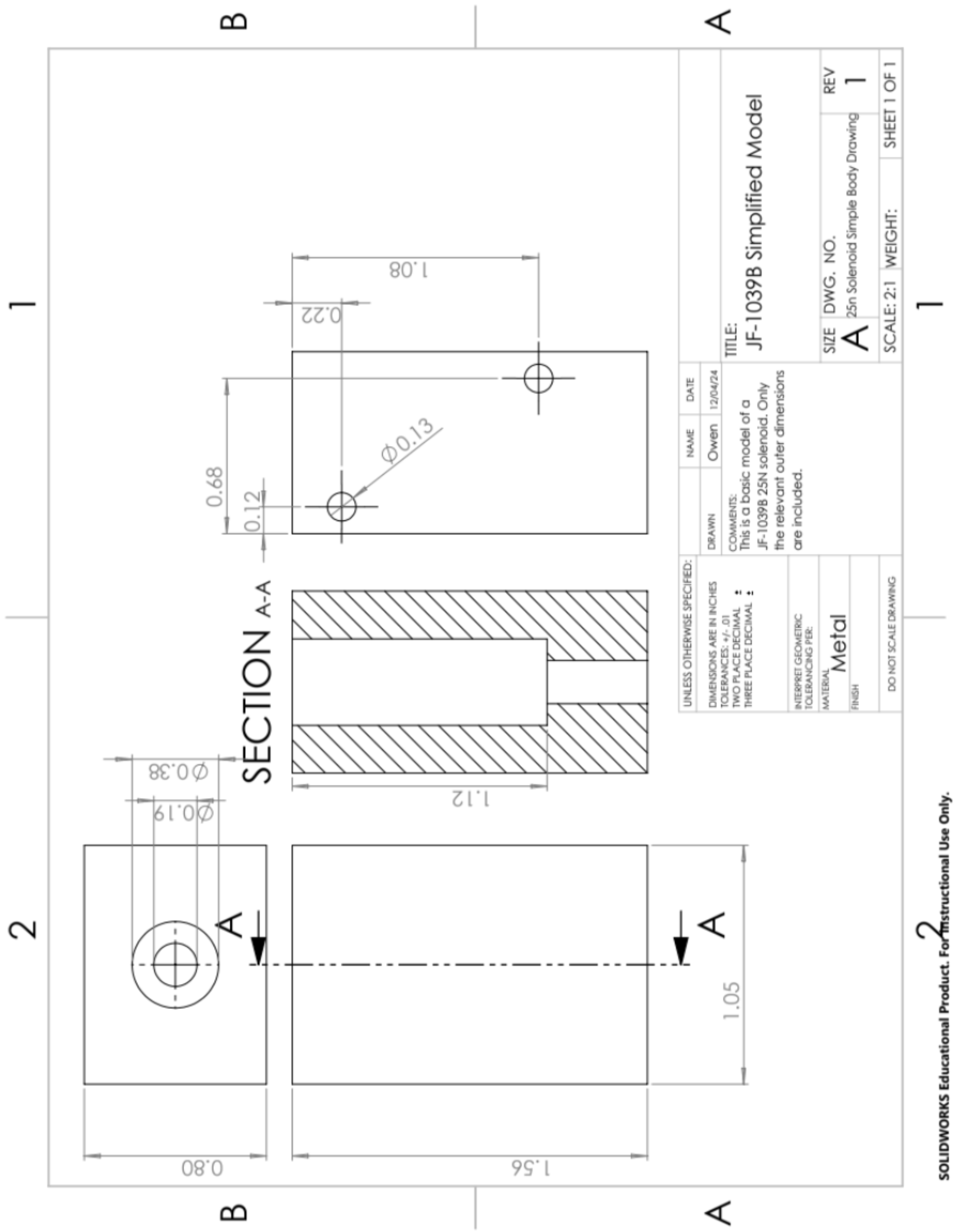


Figure 16: 25N Solenoid Basic Model

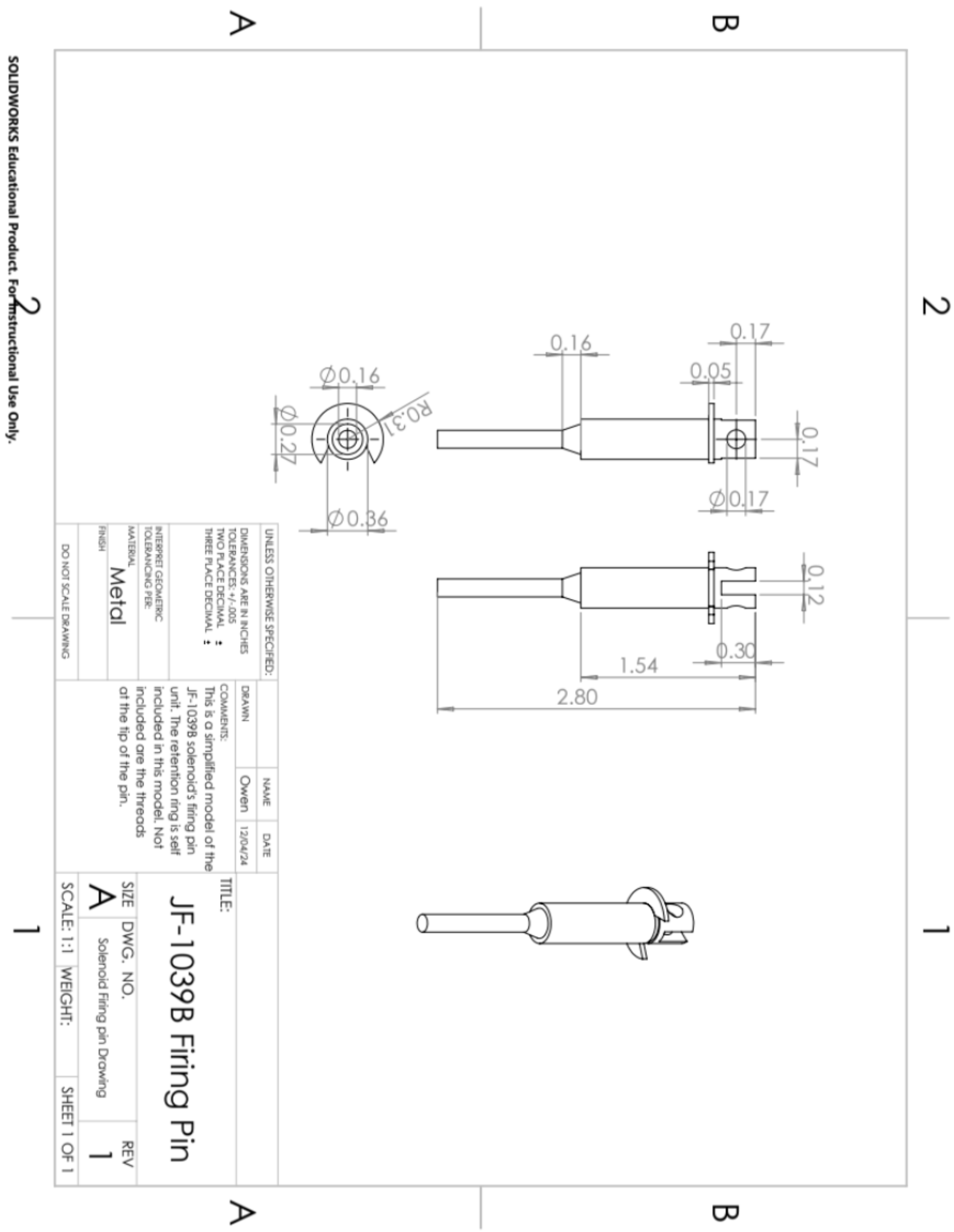


Figure 17: 25N Solenoid Firing Pin Model

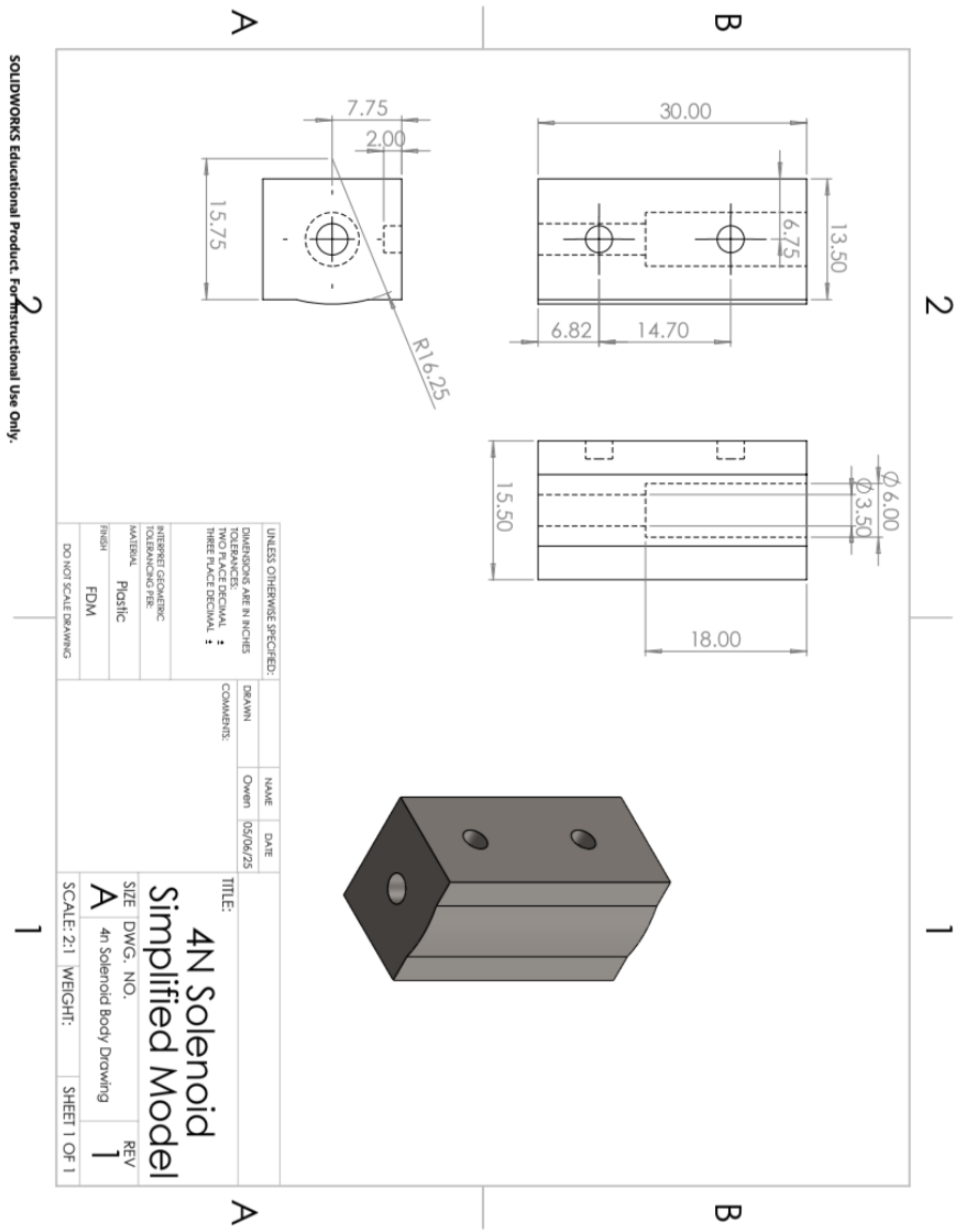


Figure 18: 4N Solenoid Basic Model

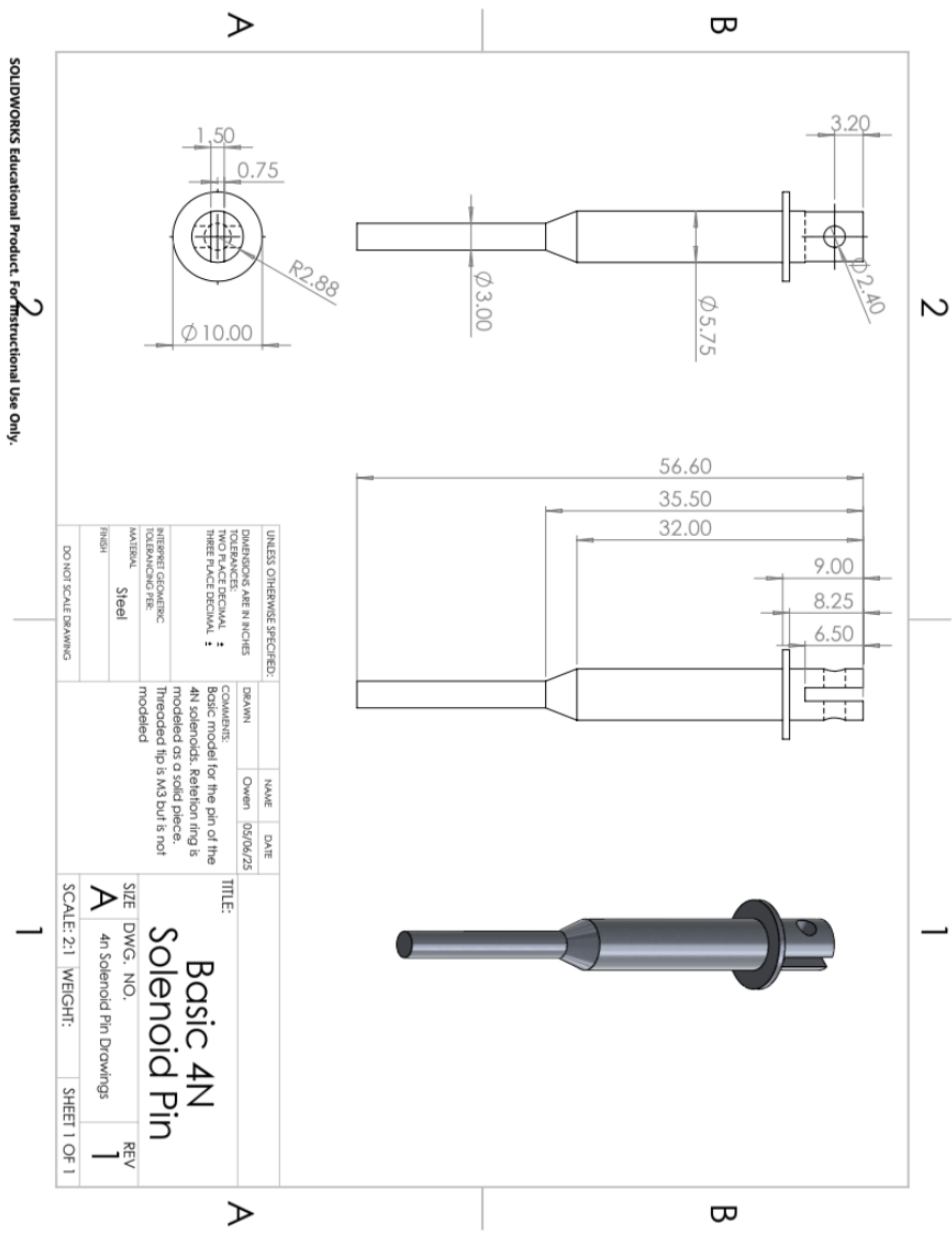


Figure 19: 4N Solenoid Firing Pin Model

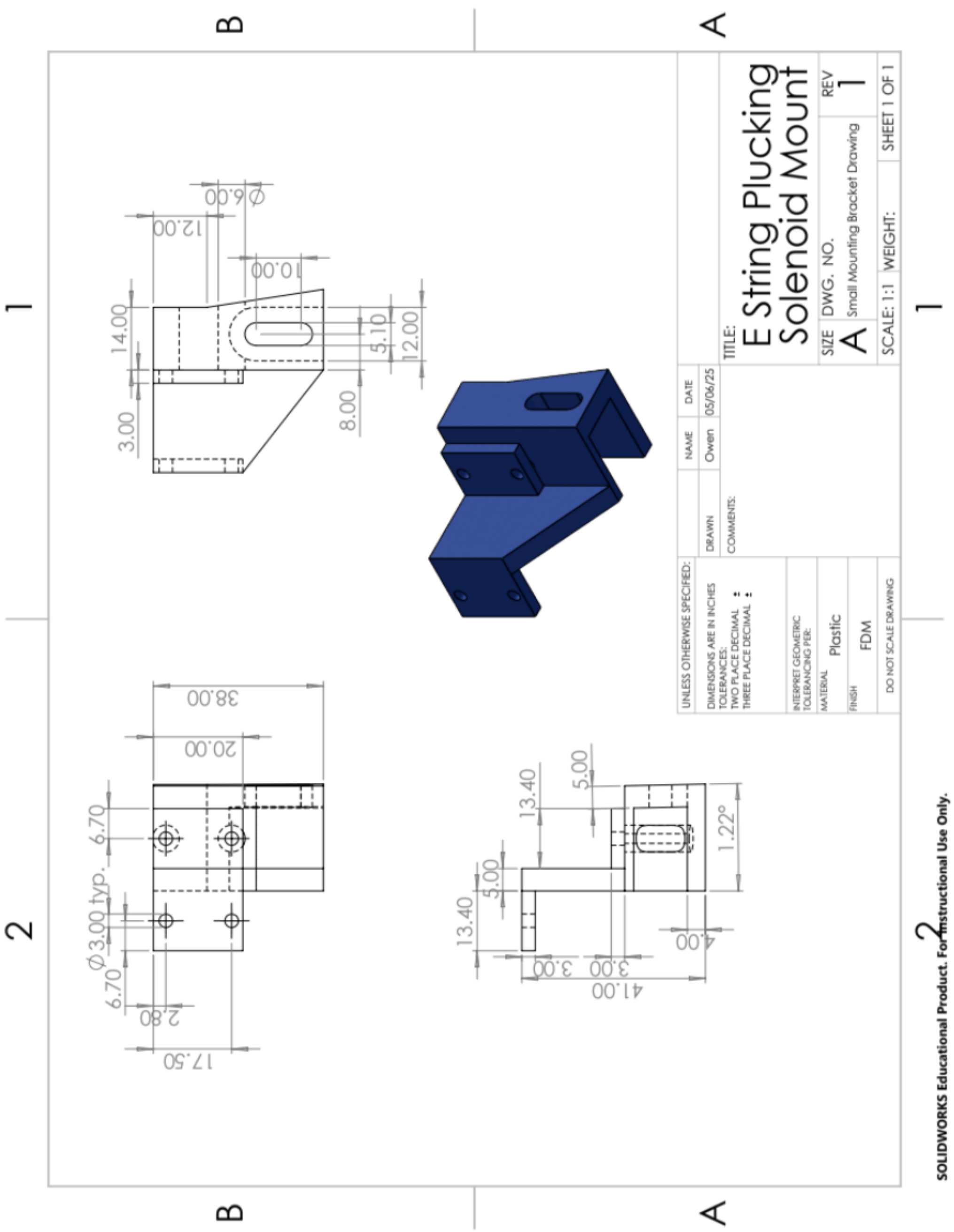


Figure 20: Plucking Pack Mount

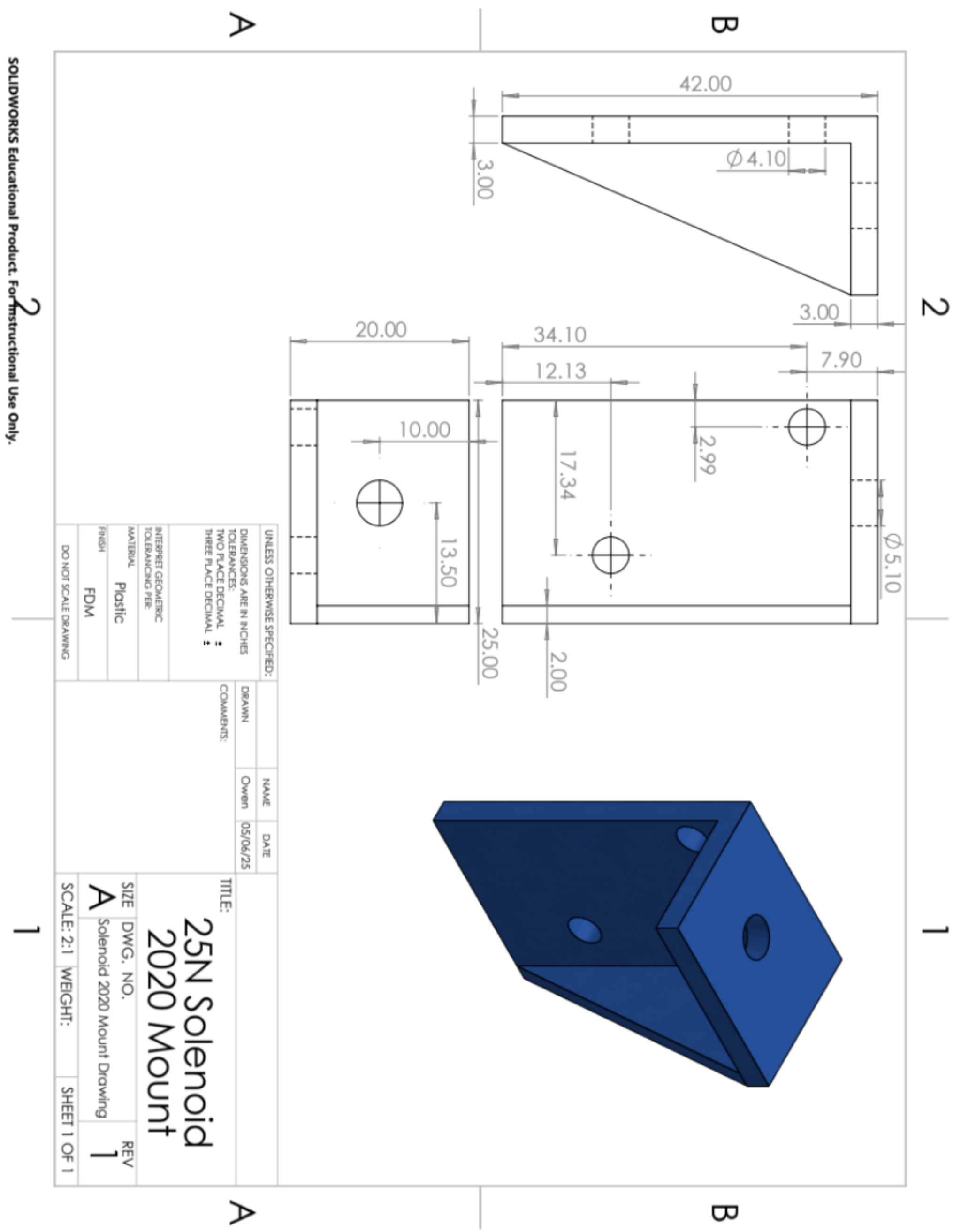


Figure 21: 25N Solenoid Frame Mount

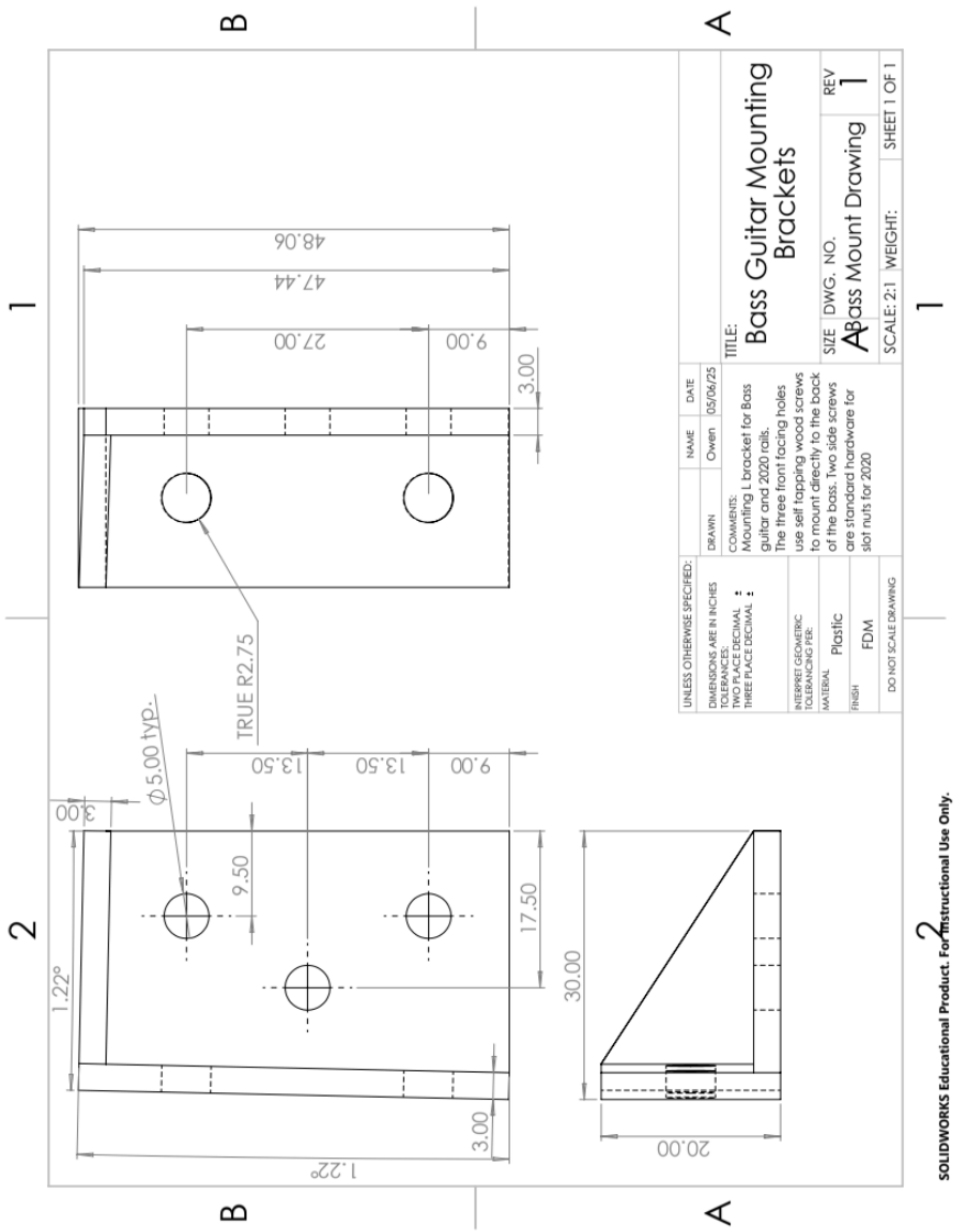


Figure 22: Bass Guitar Mounting Bracket

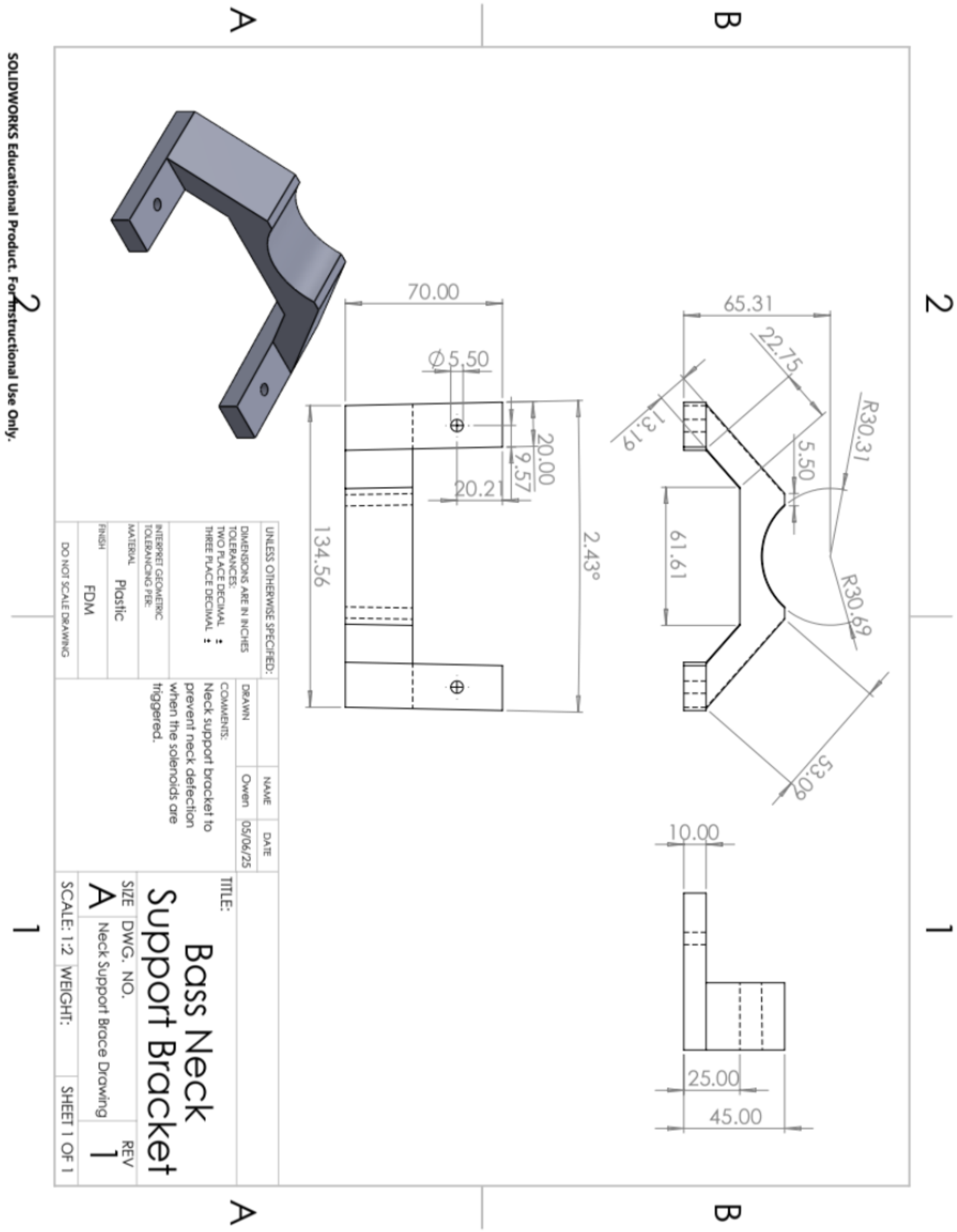


Figure 23: Bass Guitar Neck Support Bracket

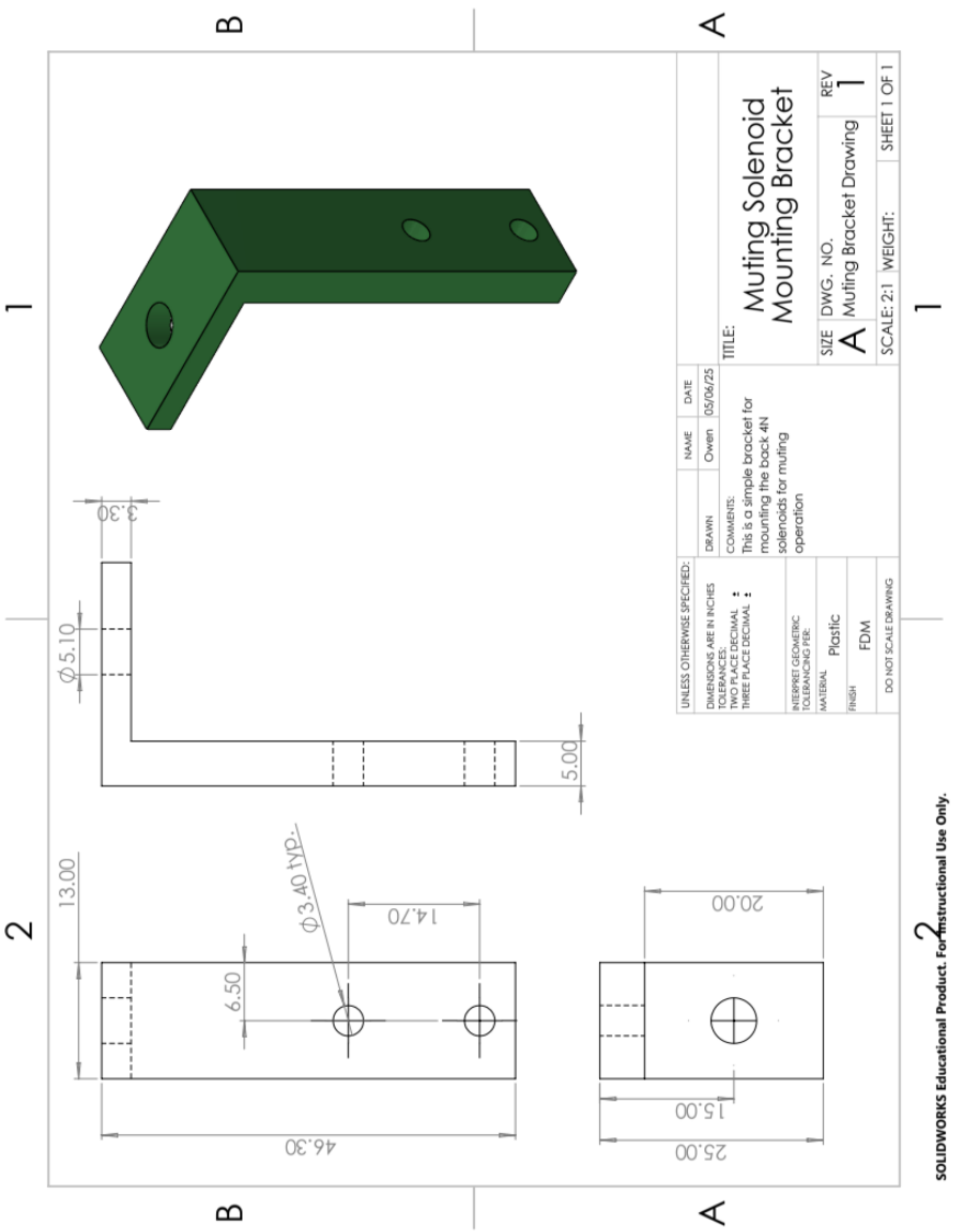


Figure 24: Muting Solenoid Mounting Bracket

Appendix C: Schematics

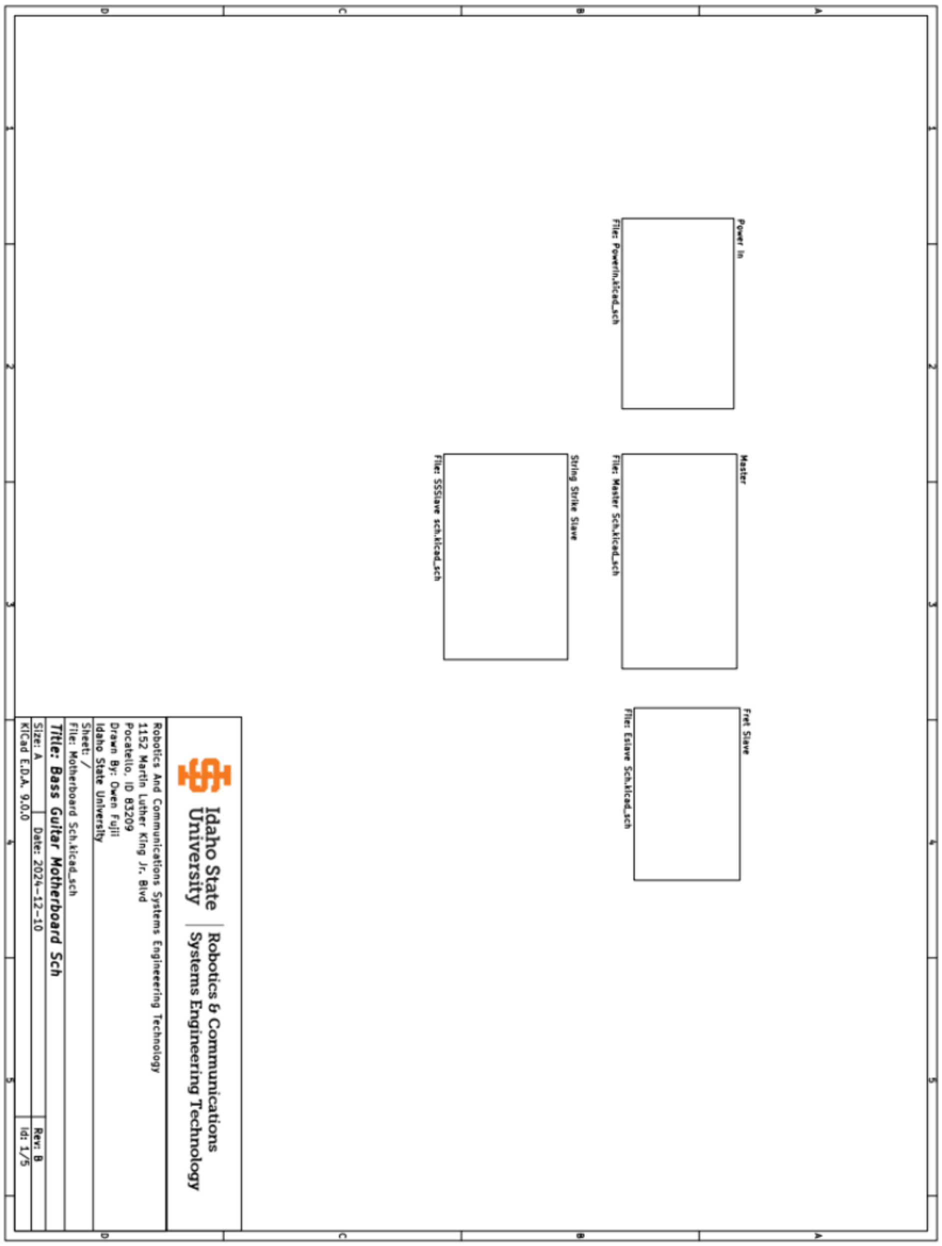


Figure 25: Motherboard Cover Page

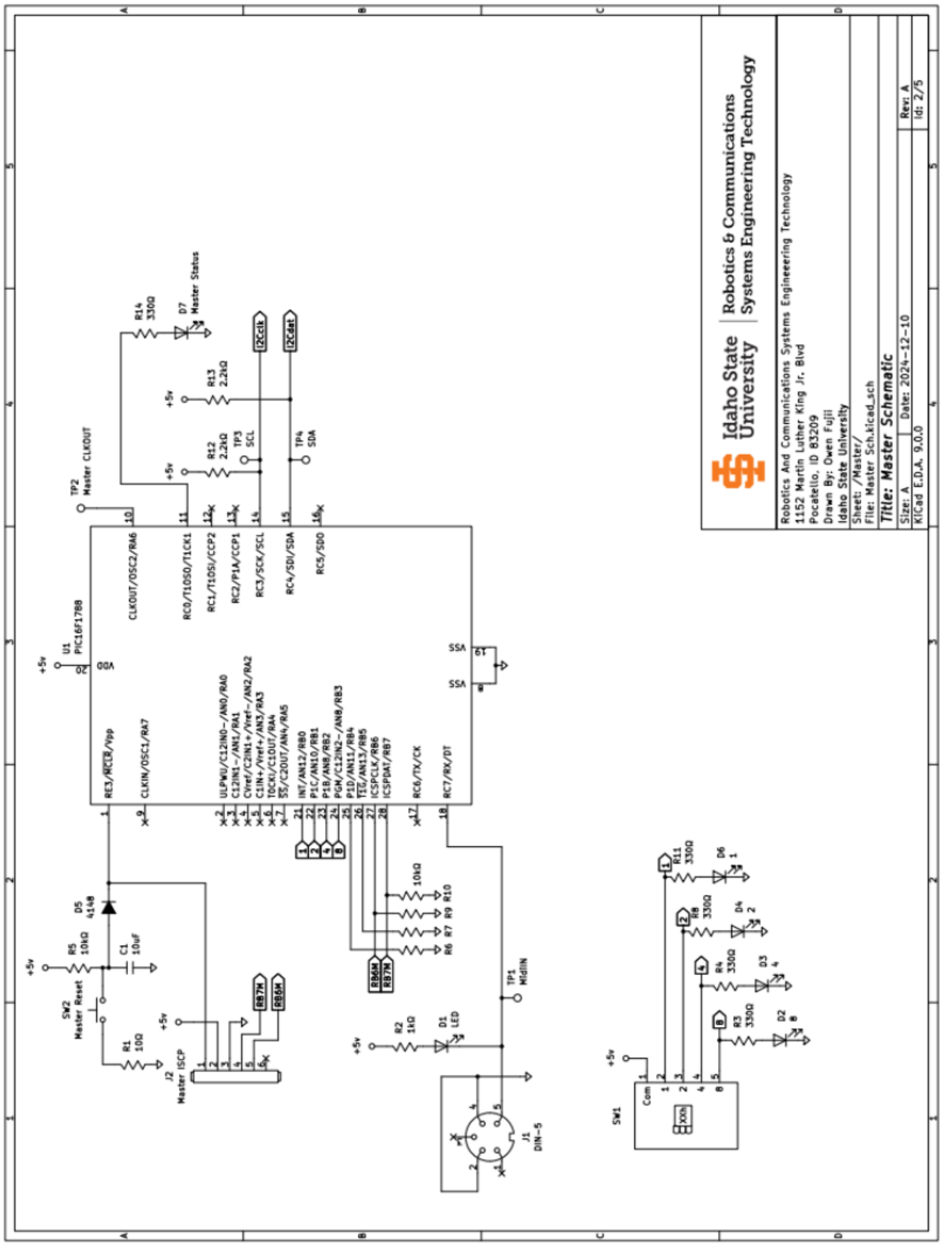


Figure 26: Motherboard Master Schematic

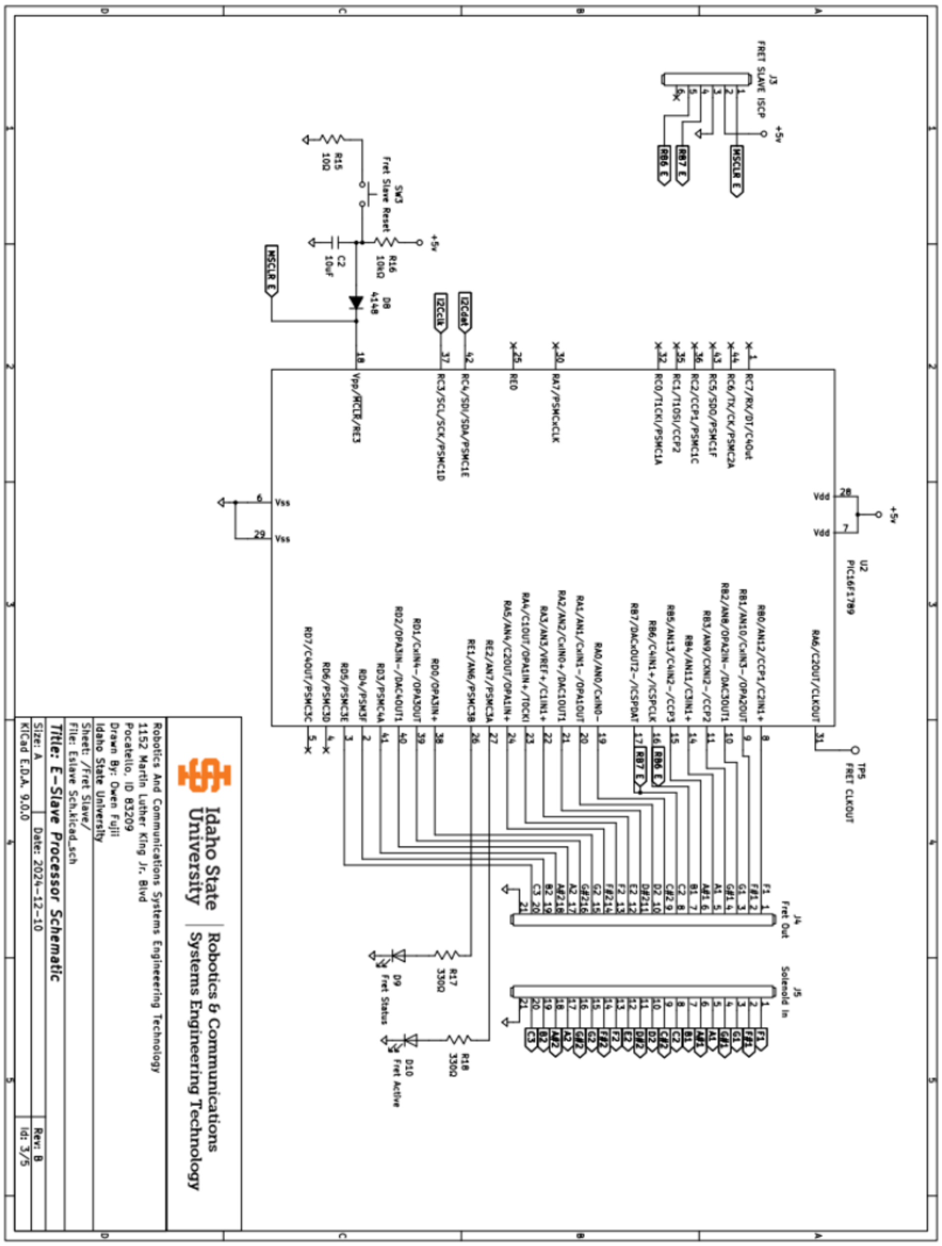


Figure 27: Motherboard Fret Peripheral Schematic

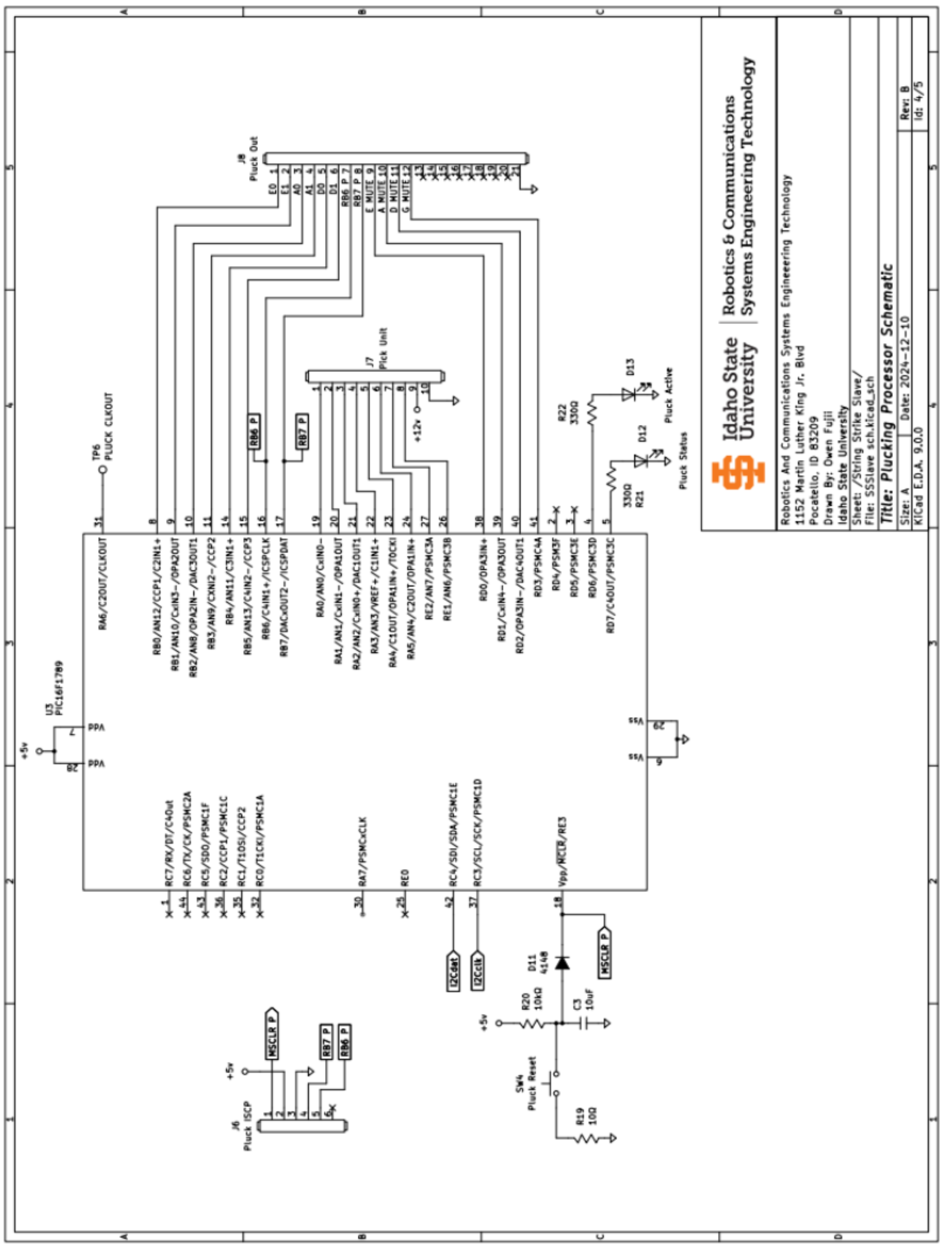


Figure 28: Motherboard Pluck Peripheral Schematic

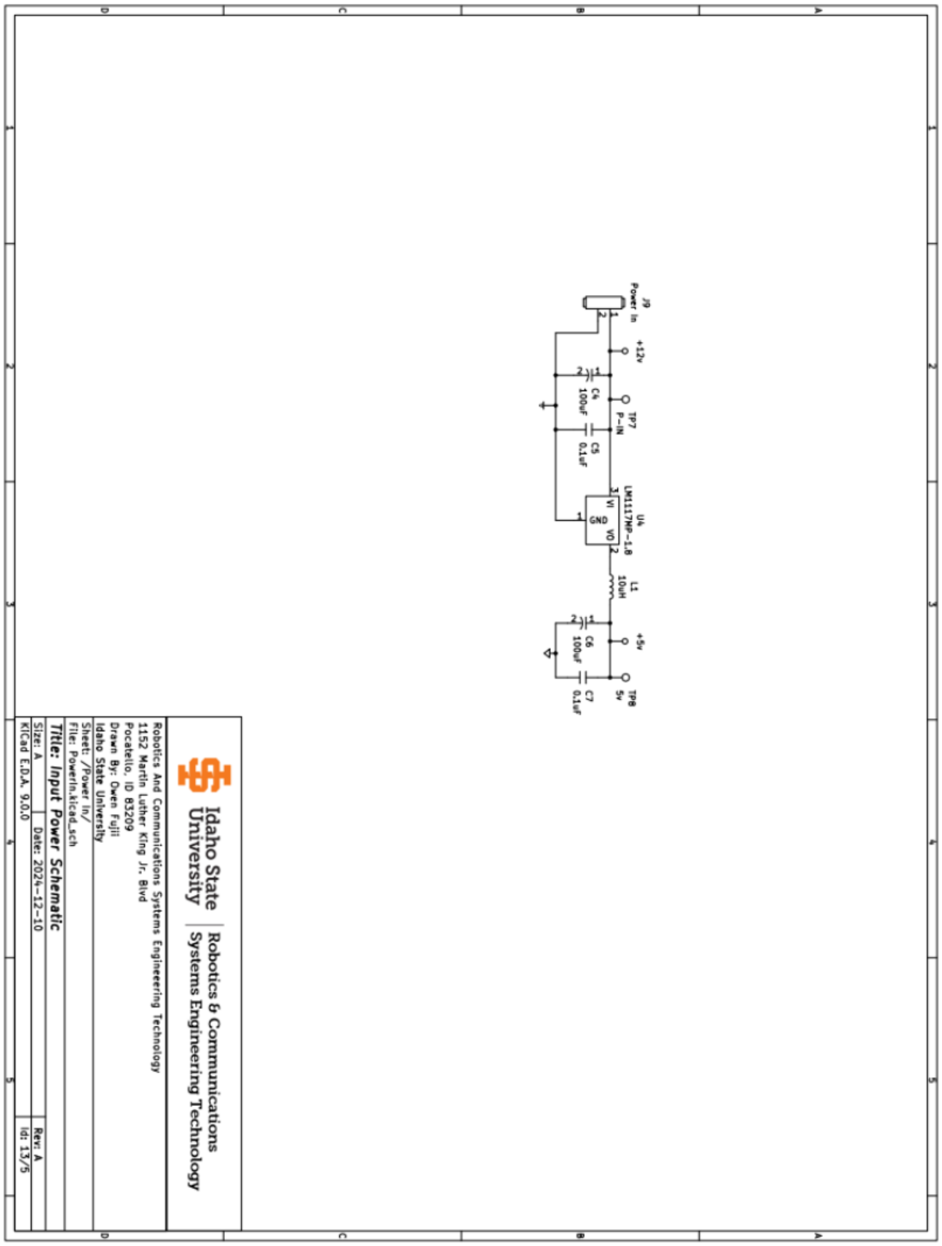


Figure 29: Motherboard Power Schematic

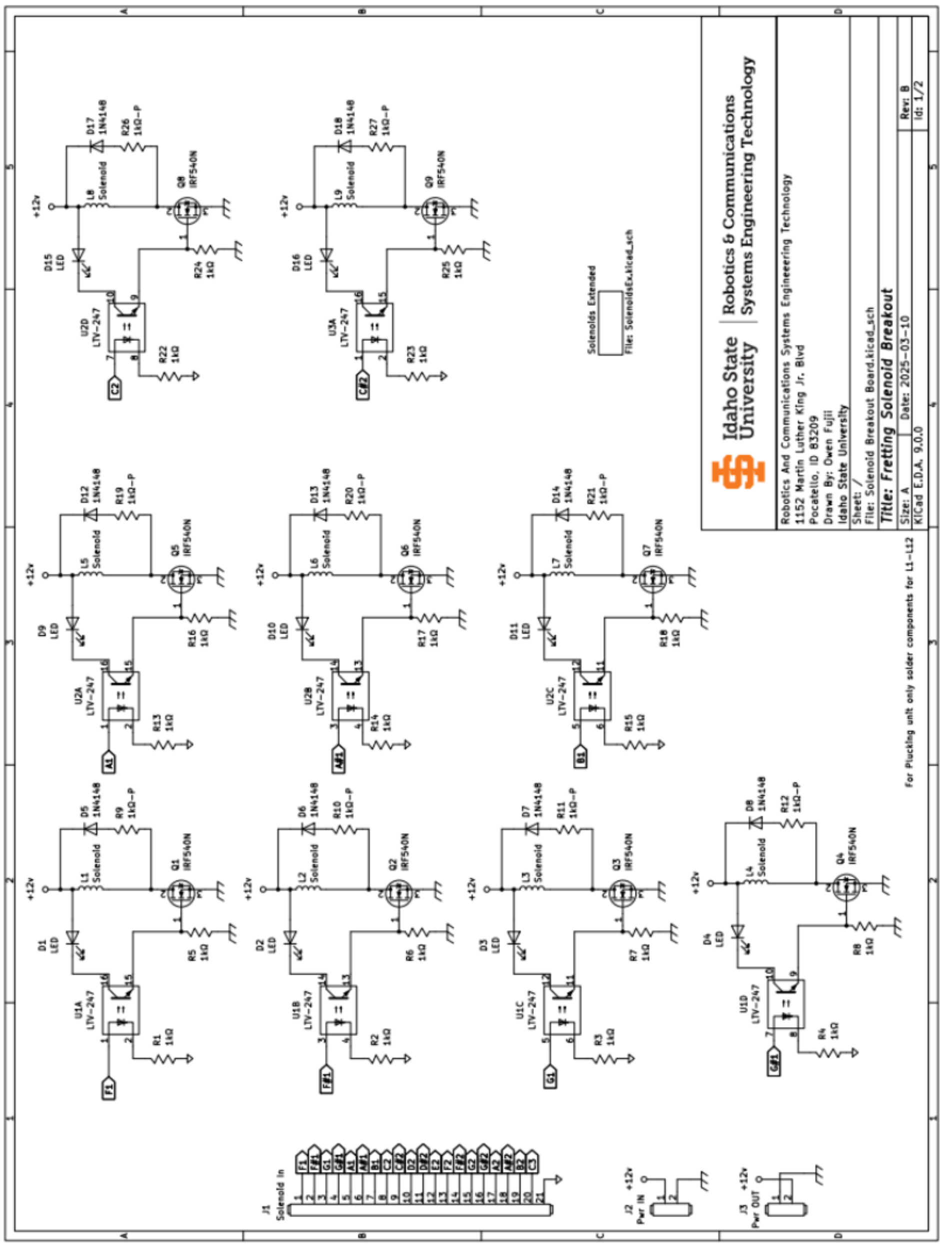
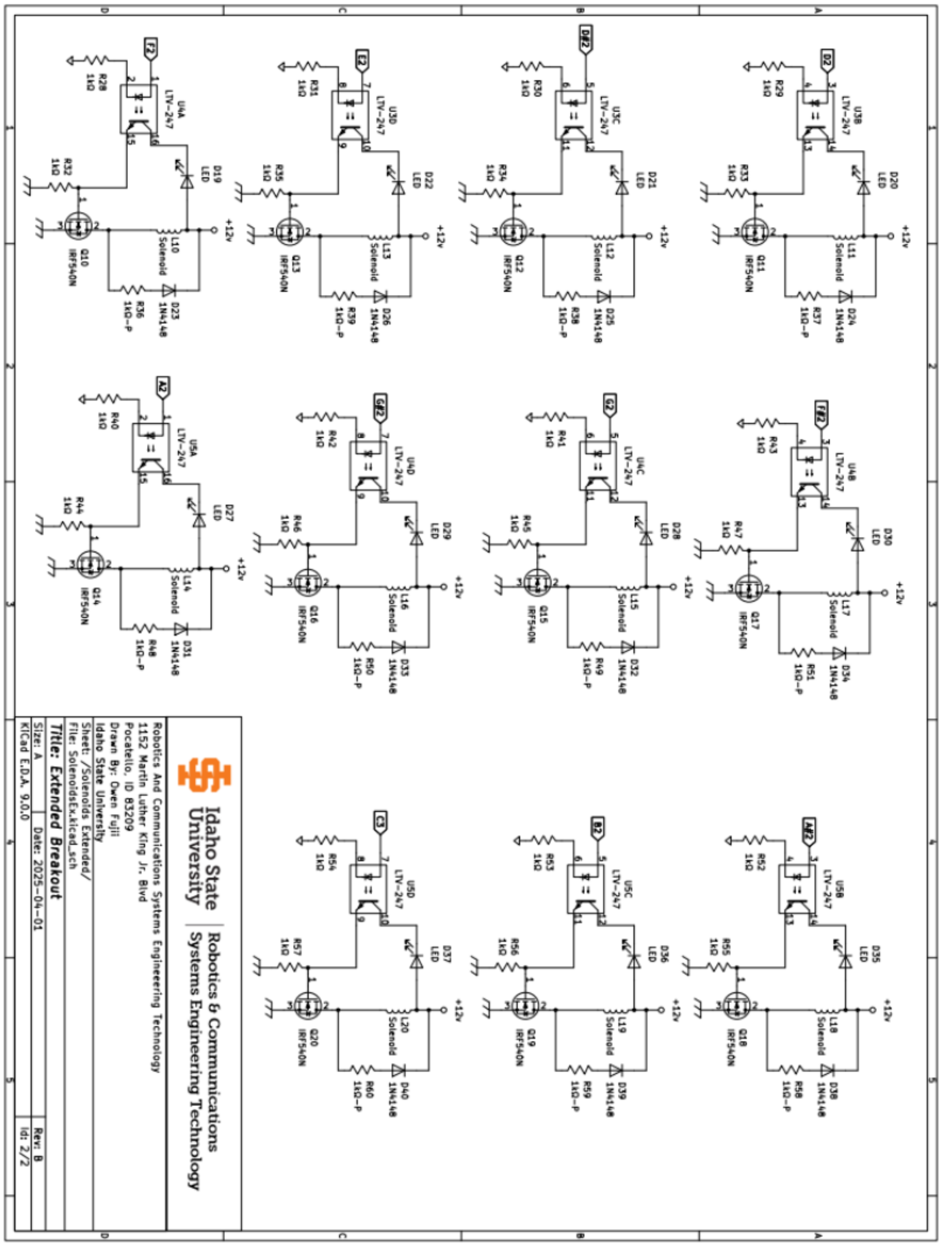


Figure 30: Breakout Board Schematic P1


Idaho State University
Robotics & Communications Systems Engineering Technology
Title: Extended Breakout
Size: A
Date: 2025-04-01
Rev: B
KiCad E.D.A. 9.0.0
Id: 2/2

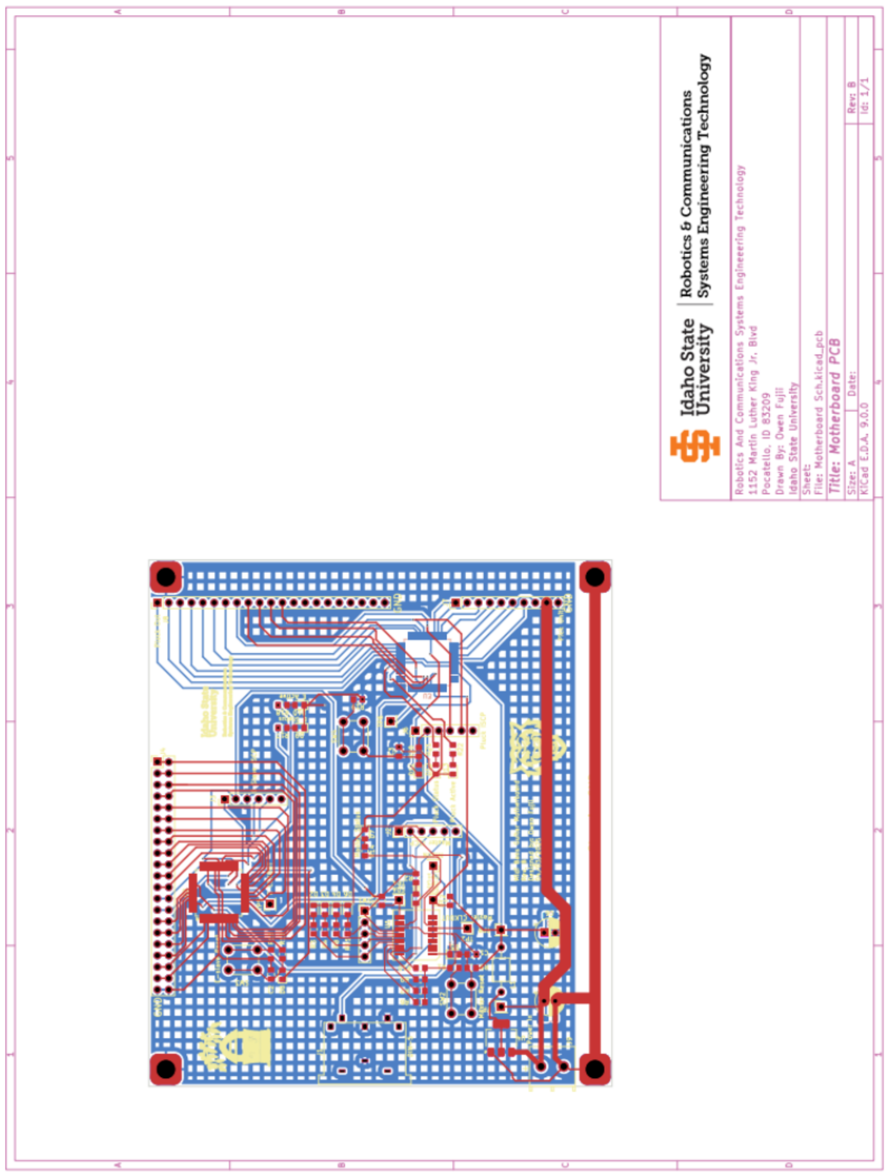


Figure 31: Motherboard PCB

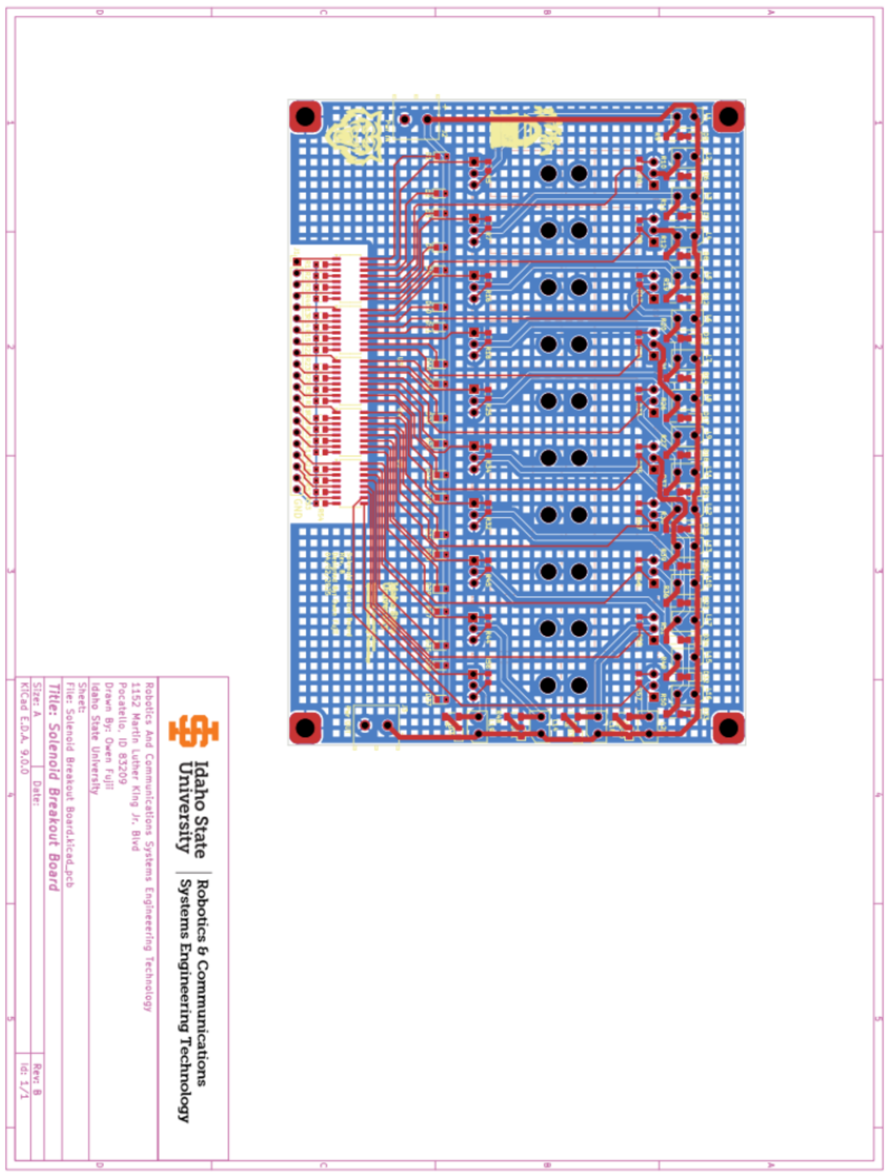


Figure 32: Solenoid Breakout PCB

Appendix D: Flowcharts

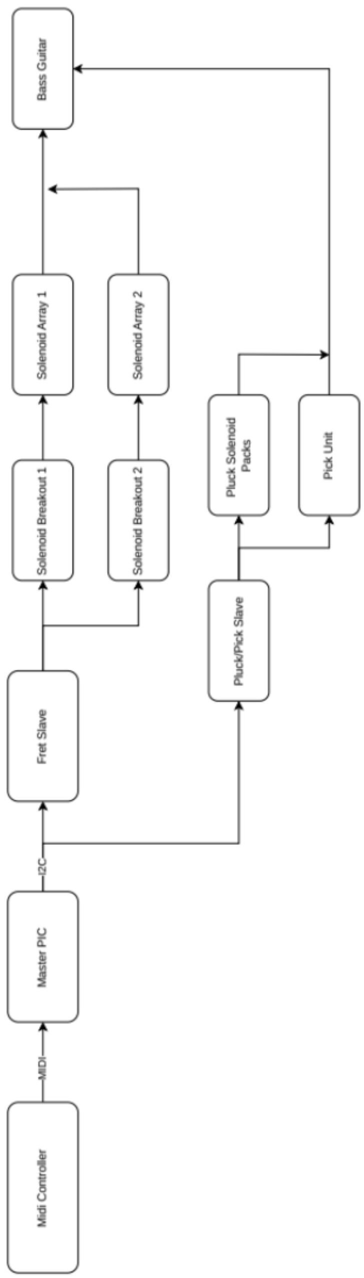


Figure 33: Functional Block Diagram

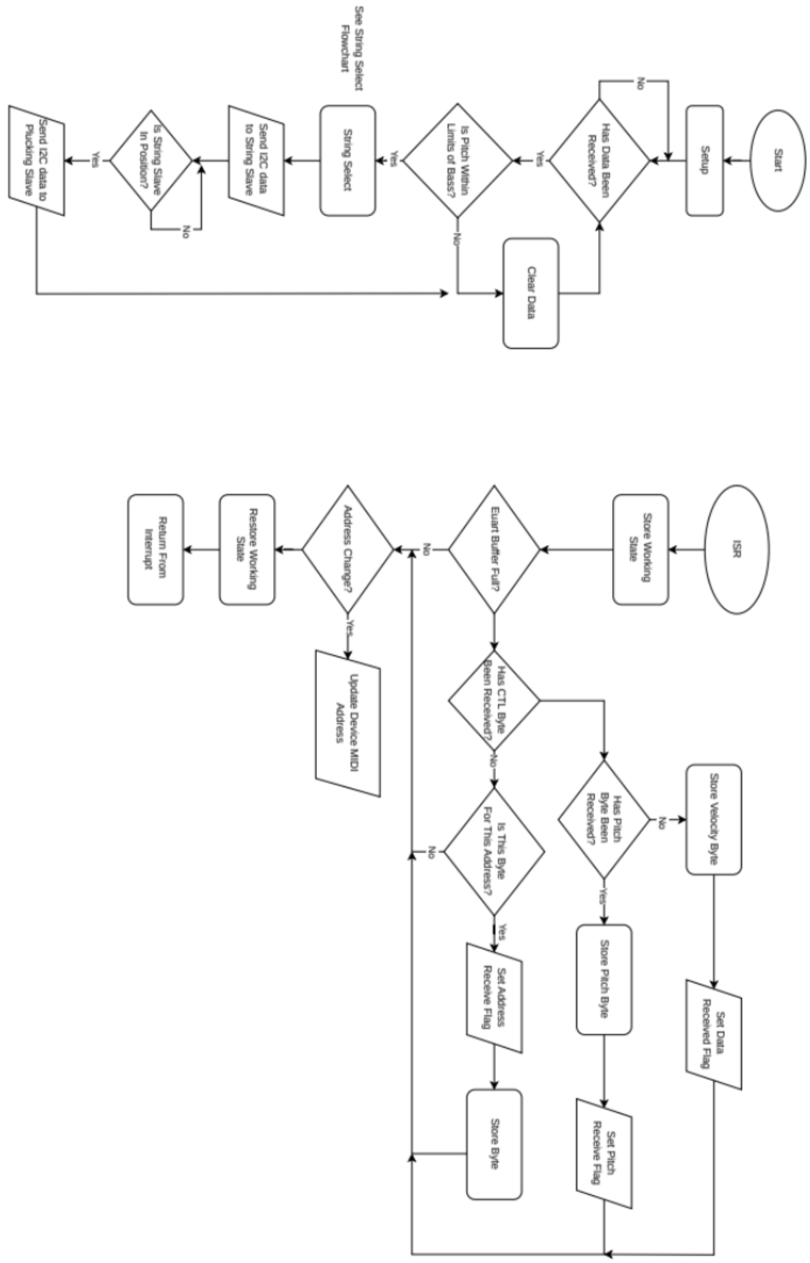


Figure 34: Master Flowchart

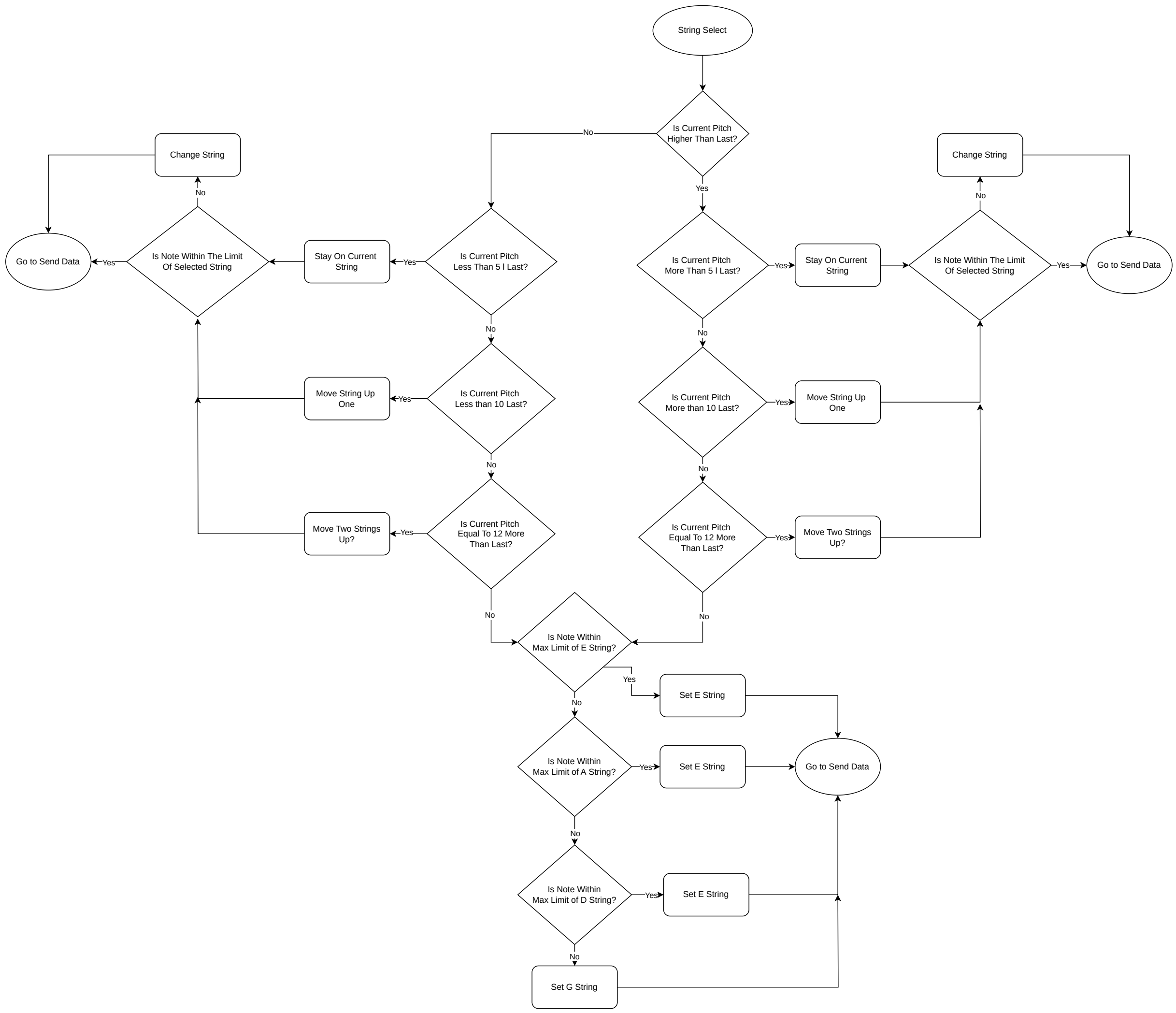


Figure 35: String Select Algorithm Flowchart

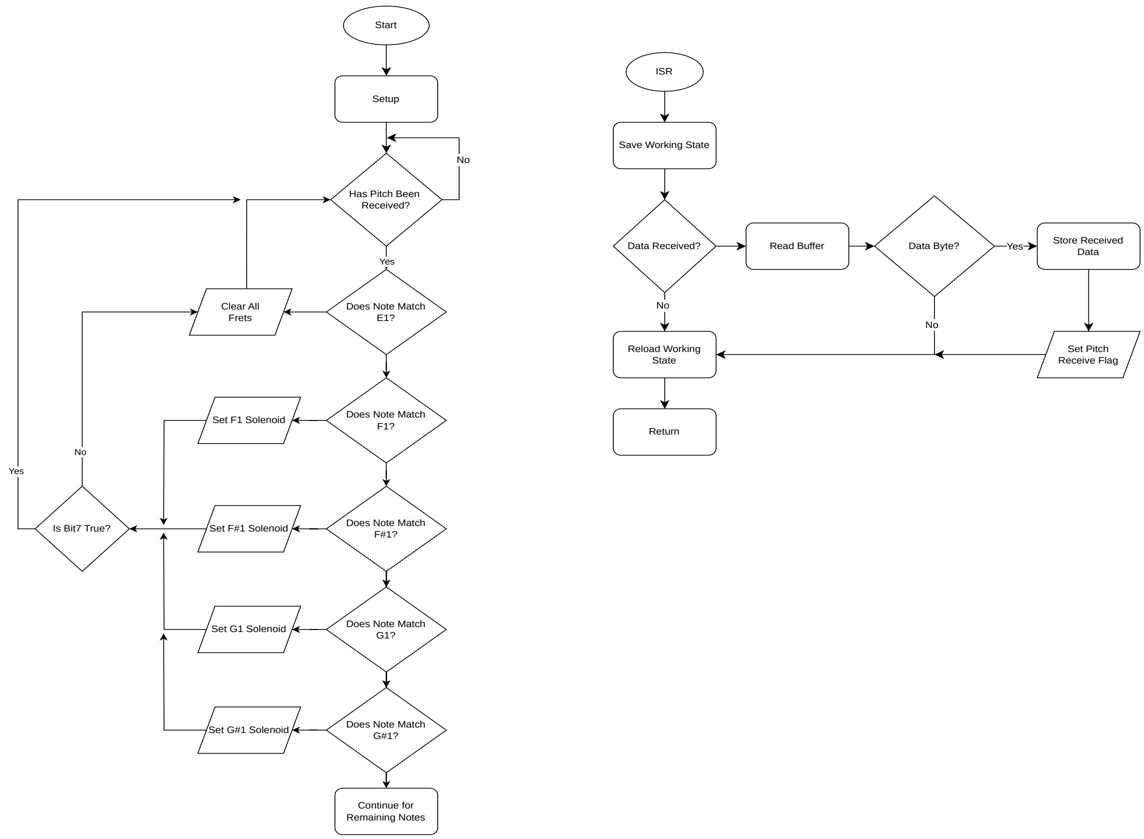


Figure 36: Fretting Program Flowchart

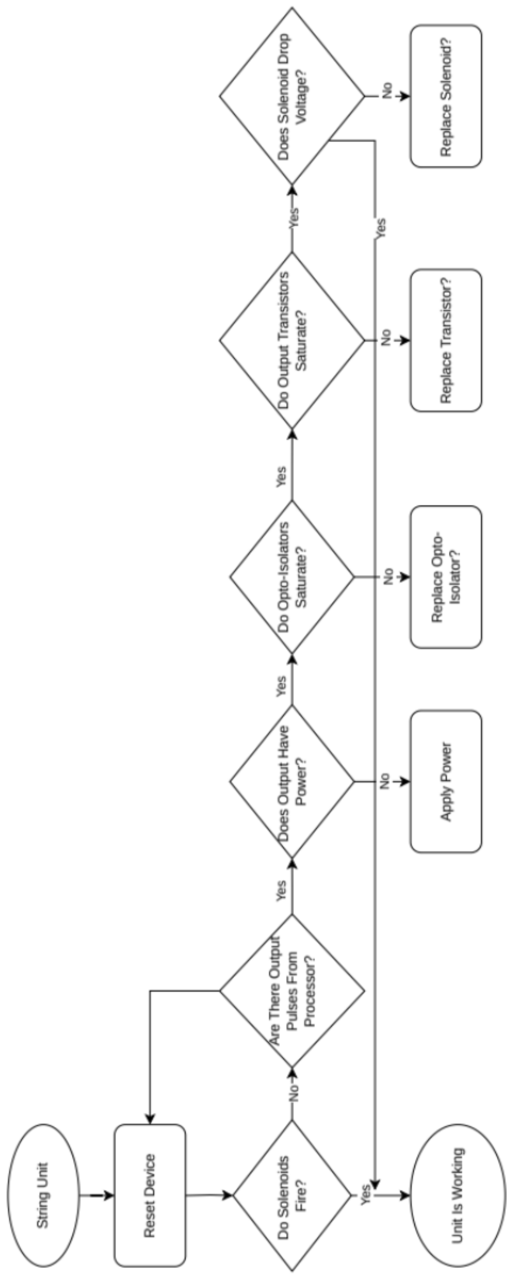


Figure 37: Fretting Unit Troubleshooting Flowchart

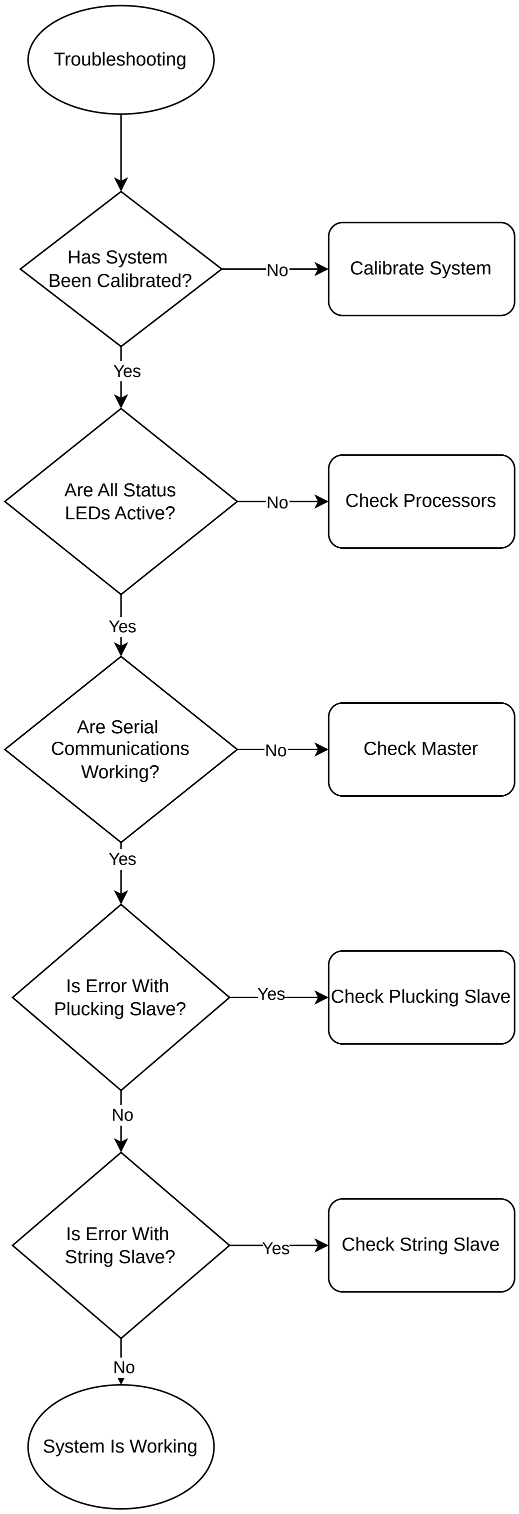


Figure 38: Firmware Troubleshooting Flowchart

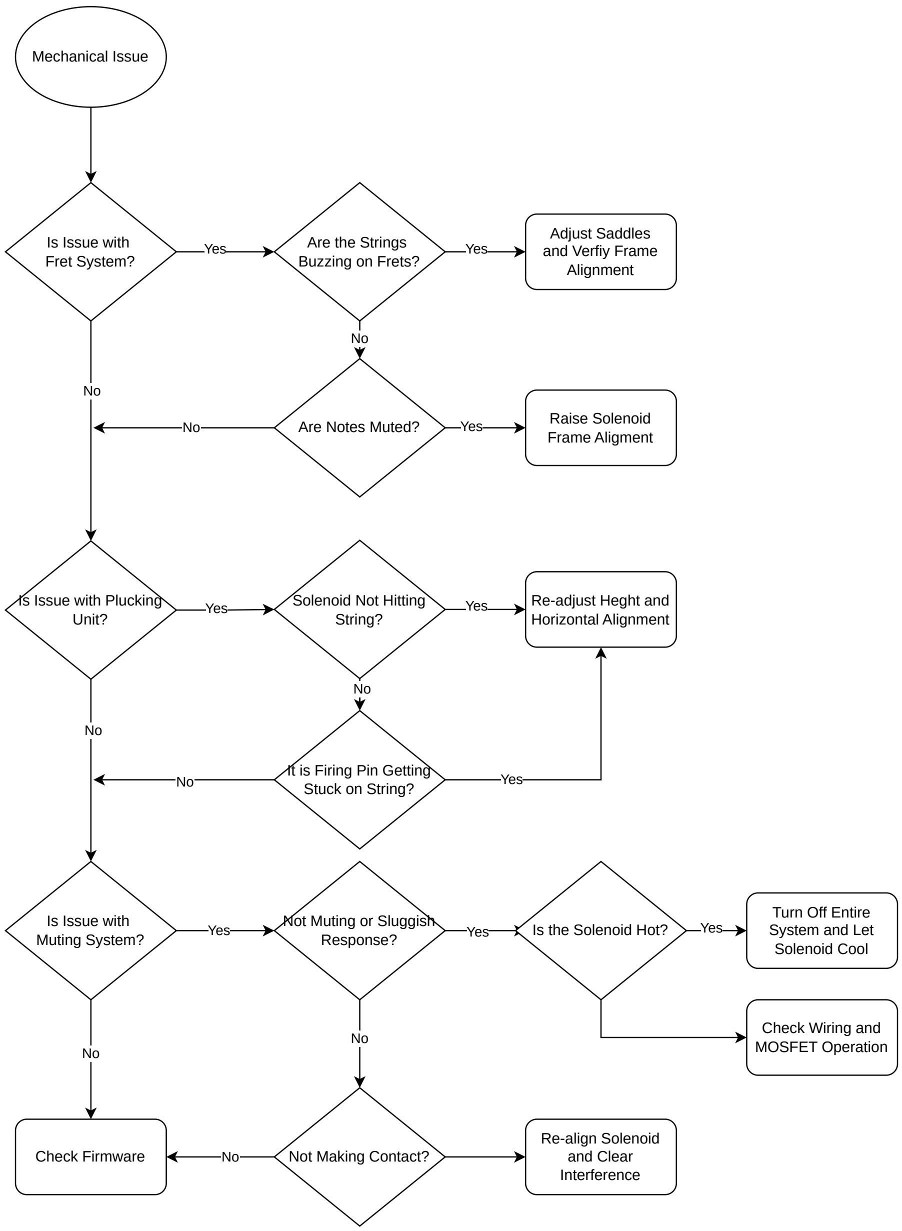


Figure 39: Mechanical Troubleshooting Flowchart

Appendix E: Tables

MIDI Controlled Bass Guitar Projected BOM										
Item Numbe	Quantity	Component Type	Reference	Value	Footprint	Supplier	Part Number	Cost (per 100)	Cost Per 100 Project	Cost/Project
1	3	Capacitor	C1-C3	10uF	0805	Digikey	490-6473-2-ND	$0.0360	$10.80	$0.11
2	2	Capacitor	C5,C7	0.1uF	0805	Digikey	311-1140-2-ND	$0.0200	$4.0000	$0.04
3	2	Capacitor	C4,C6	100uF	6.3x11mm	Digikey	1189-1856-ND	$0.1137	$22.74	$0.23
4	3	Resistor	R1,R15,R19	10	0805	Digikey	RMCF0805JG10R0TR-ND	$0.0085	2.55	$0.03
5	9	Resistor	~	330	0402	Digikey	P330LTR-ND	$0.0125	$11.25	$0.11
6	21	Resistor	~	1k	0402	Digikey	P1.00KLTR-ND	$0.0126	$26.46	$0.26
7	52	Power Resistor	~	1K		Digikey	RA73F2A10KBTD	$1.0600	$5,512.00	$55.12
8	2	Resistor	R12,R13	2.2k	0402	Digikey	P2.20KLTR-ND	$0.0125	$2.50	$0.03
9	55	Diode	~	4148	0805	Digikey	1801-TS4148TBGTR-ND	$0.0458	$251.90	$2.52
10	60	LED	~	LED	0805	Digikey	732-4982-2-ND	$0.1359	$815.40	$8.15
11	1	Microcontroller	U1	PIC16F1788	SSOP-28	Digikey	PIC16F1788-I/SS-ND	$2.1400	$214.00	$2.14
12	2	Microcontroller	U2,U3,U8,U9,10	PIC16F1789	TQFP-44	Digikey	PIC16F1789-I/PR-ND	$2.5000	$500.00	$5.00
13	13	Opto-Isolator	U4-U7,U11-U30	LTV-247	SOIC-16	Digikey	160-2255-2-ND	$0.6456	$839.28	$8.39
14	1	Linear Regulator	U31	LM1117MP	SOT-223	Digikey	488-LM1117MPX 33NOPBCT-N	$0.4478	$44.78	$0.45
15	3	Switch	SW2-SW4	Push Switch	6x6x5mm	Digikey	450-1650-ND	$0.1074	$32.22	$0.32
16	1	Switch	SW1	Hex Thumbswitch	2.54mm x 5	Digikey	A7BS-254-1-ND	$13.1209	$1,312.09	$13.12
17	1	Inductor	L93	10uH	3.96mm x 2pin	Digikey	M8278-ND	$0.8359	$83.59	$0.84
18	40	Solenoid	L1-L20 (2x)	25N	2.54mm Pin Header	Digikey	IF-1039B	$13.9900	$55,960.00	$559.60
19	12	Solenoid	L1-L12	4N	3.96mm x 2pin	Amazon	IF-0530B	$12.9900	$15,588.00	$155.88
20	1	Bass	Bass	P-Bass	N/A	Amazon	Squier Precision Bass	$129.9900	$12,999.00	$129.99
21	52	Transistor	Q1-Q20 (3x)	IRF540N	TO-220	Mouser	942-IRF540NPBF	$0.6380	$3,317.60	$33.18
22	52	Header		3.96mm Header	3.96mm x 2pin	Mouser	5716404452	$0.0950	$494.00	$4.94
23	52	Connector	~	3.96mm Con.	~	Mouser	571-3-640429-2	$0.2540	$1,320.80	$13.21
24	2	Header	J2-J13	2.54mm Headers	2.54mm x varies	Amazon	MCIGICM 40pin	$4.9900	$998.00	$9.98
25	1	Connector	J1	DIN-5	~	Mouser	571-5212044-1	$3.8900	$389.00	$3.89
26	1	PETG	~	1Kg	~	Amazon	1.75mm Matte Grey PETG	$13.9900	$1,399.00	$13.99
27	3	Aluminuim	~	3m	~	Amazon	1220mm 2020 Aluminuim Extr	$35.2500	$10,575.00	$105.75
28	1	Weather Stripping	~	2ft	~	Amazon	Weather Strip Silicone D Shap	$0.9900	$99.00	$0.99
29	3	Breakout PCB	~	~	~	PCBWAY	Solenoid-Breakout (1)	$5.0000	$500.00	$15.00
30	1	Motherboard PCB	~	~	~	PCBWAY	Midi_Motherboard_PCB	$4.6000	$460.00	$4.60
31	1	Wire	~	~	~	Amazon	200ft CL3P	$85.9900	$8,599.00	$85.99
									Total Cost Per 100	$122,383.96
		Member	Hours	Wage Per Hour	Labor Cost					
	Labor	Owen	132.167	$50.00	$6,608.35				Total Cost Per Unit	$1,233.84
		Justin	96	$50.00	$4,800.00					
				Labor Total	$11,408.35				Project Cost	$12,642.19

Figure 40: Bill of Materials

Project Code and Repo

Use the QR to access the repository that has all the code as other associated files for the project. Pull the “stable” branch for the latest stable release of the project files.