Piano Visualizer Design Document

By

Jay Park

Nuwan Singhal

Sarayu Suresh

Design Document for ECE 445, Senior Design, Spring 2026

TA: Wenjing Song

27th February 2026

Project No. 98

Contents

1. Introduction.. 3

1.1 Problem.. 3

1.2 Solution.. 3

1.3 Visual Aid:..4

1.4 High-level Requirements.. 4

2 Design.. 5

2.1 Block Diagram..5

2.2 Subsystem Overview.. 6

2.1.1 RGB LED Matrix Subsystem.. 6

2.2.2 SD Card Storage Subsystem..7

2.1.3 MIDI Input Subsystem..9

2.1.4 User Interface Subsystem..10

2.1.5 Power Management Subsystem..12

2.1.6 Control Subsystem.. 14

2.1.7 Microphone Subsystem..16

2.3 Tolerance Analysis.. 17

3. Cost and Schedule.. 18

3.1 Cost Analysis..18

3.2 Schedule.. 21

4. Ethics Safety and Societal Impact..22

References.. 23

1. Introduction

1.1 Problem

Learning piano is hard for beginners because the core feedback loop is slow. A new player must translate sheet music into correct keys, timing, and duration, then self diagnose mistakes while staying on tempo. In practice, many learners either do not read music yet or cannot tell whether they pressed the correct note at the correct time, especially when multiple notes occur quickly or simultaneously. This leads to repeated errors, wasted practice time, and frustration that causes people to stop practicing altogether.

Existing solutions partially address this but introduce new barriers. App based tools often depend on a phone, internet access, and a camera or microphone that can be unreliable in noisy rooms. Other options like commercial light up keyboards and proprietary learning systems are expensive, locked into specific hardware, or do not provide precise, immediate feedback for both timing and correctness. Our project targets this gap by providing a standalone, low latency visual guidance and feedback system that works directly with a keyboard input and optional audio input, so a learner can practice any song stored locally and immediately see what to play and whether they played it correctly.

1.2 Solution

We propose a real time piano input visualizer that reads MIDI files from an SD card and provides visual guidance using a 64×32 RGB LED matrix. The system parses note, velocity, and timing data from the selected song and displays which keys should be pressed and for how long.

The device accepts direct MIDI input from a digital piano and compares incoming notes to the expected notes defined in the MIDI file. It provides immediate visual feedback indicating correctness and timing accuracy. Optional microphone based input enables compatibility with acoustic instruments. Additional features such as adjustable playback speed, single hand practice mode, and wait mode allow users to tailor the learning experience to their skill level.

1.3 Visual Aid:

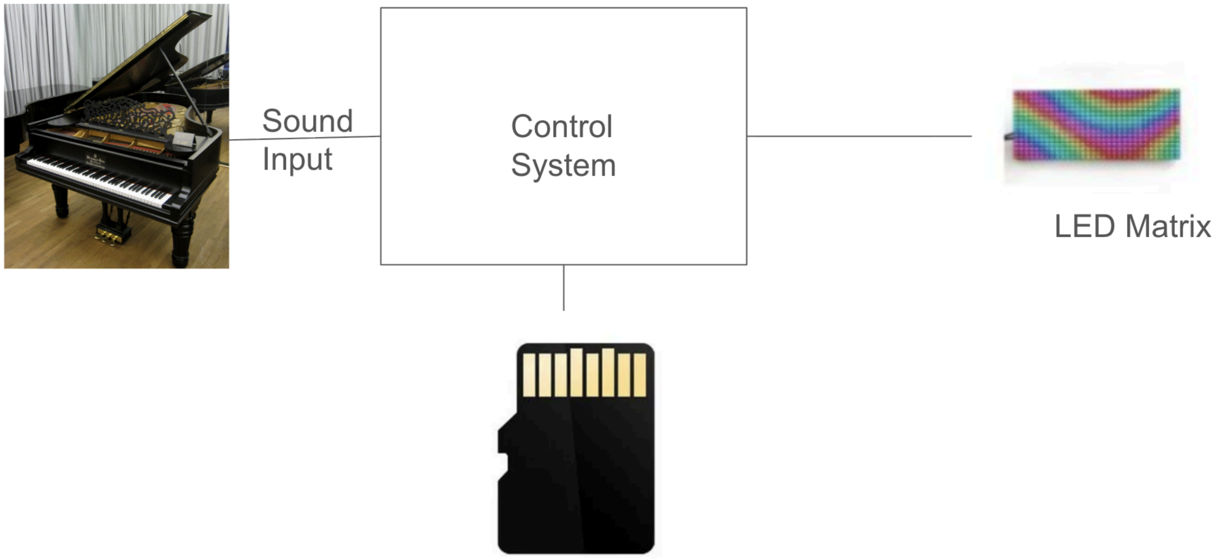

Figure 1 Visual Aid for our project

The SD card contains the song that the user wants to learn, and it goes into the control system. Then the control system takes MIDI and sound input from the piano and checks if the correct piano input is being played. The LED matrix shows feedback of correctness, timing accuracy, and note duration.

1.4 High-level Requirements

1. The system shall read and parse Standard MIDI files from a microSD card and display the correct note positions and durations on the 64×32 RGB LED matrix, while maintaining a display refresh rate of 60 Hz or higher.
2. When receiving 5 pin DIN MIDI input from a keyboard, the system shall detect Note On/Note Off events and update the LED matrix to indicate correctness with an end to end latency of 30 ms or less, while supporting up to 10 simultaneous notes.
3. When using microphone based input, the system shall classify a single played note and update the LED matrix with an end to end latency of 200 ms or less, with at least 90% note classification accuracy in a controlled environment.

2 Design

2.1 Block Diagram

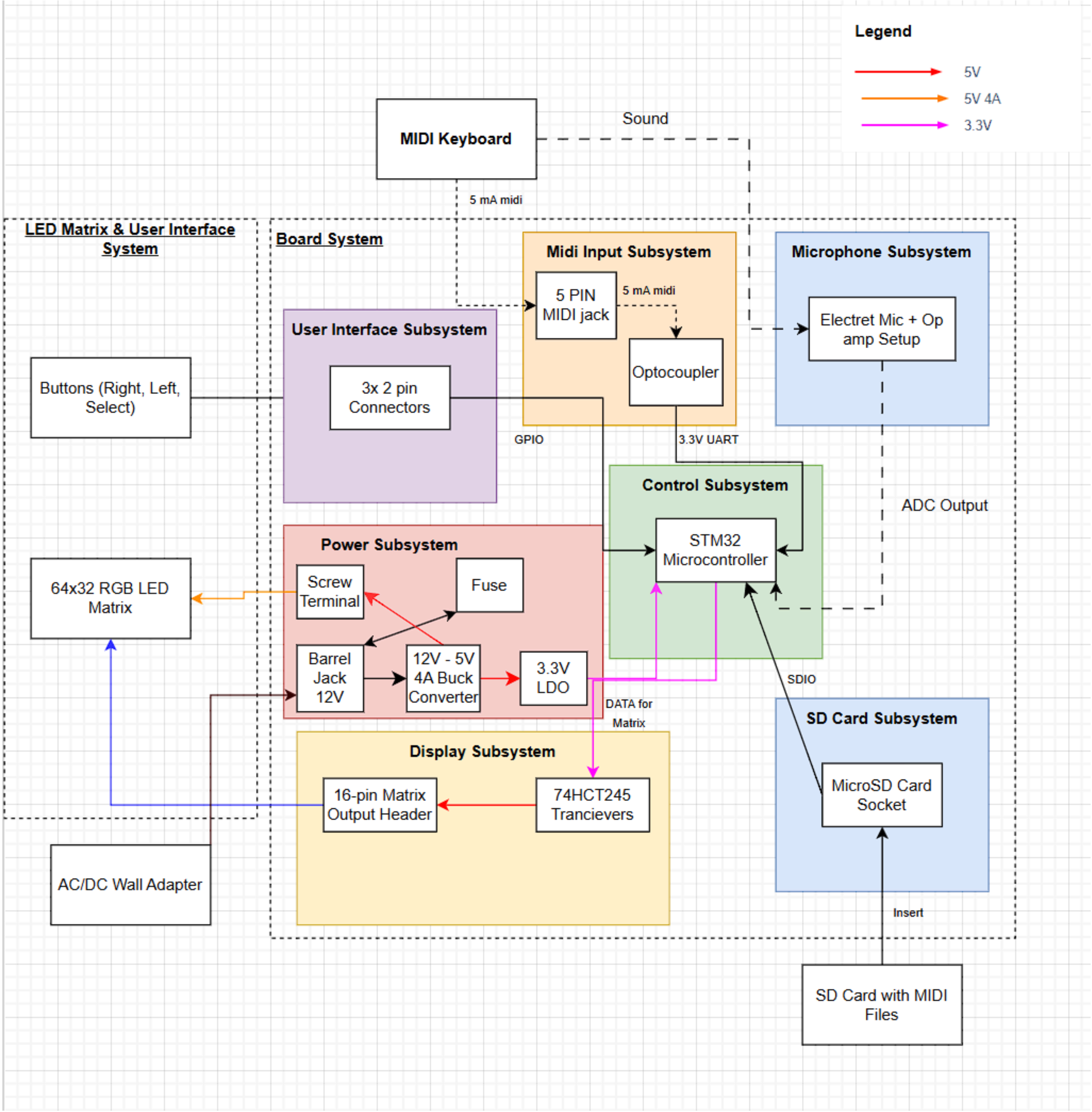

Figure 2 Block Diagram of the entire system.

2.2 Subsystem Overview

2.1.1 RGB LED Matrix Subsystem

The RGB LED Matrix Subsystem is responsible for providing all real time visual output to the user and serves as the primary feedback interface of the system. It displays the piano keys that should be pressed according to the selected MIDI file and visually indicates note duration and correctness in response to user input. This subsystem directly fulfills the first high level requirement by ensuring that note position and timing information are presented clearly and without perceptible flicker.

The selected display is a 64×32 HUB75 RGB LED matrix operating at 5 volts with a maximum current draw of approximately 4 amperes under full brightness conditions. The panel uses a 1/16 scan multiplexing scheme, requiring continuous row updates to maintain a stable image. To achieve a minimum refresh rate of 60 hertz, the STM32F446RET6 microcontroller updates all 16 scan rows at a frequency of at least 960 hertz. Direct Memory Access is used to stream pixel data to the display to reduce processor overhead and ensure deterministic timing behavior.

Because the microcontroller operates at 3.3 volt logic levels and the LED matrix requires 5 volt logic inputs, a 74HCT245 transceiver is used for level shifting to ensure reliable signal integrity. The matrix is sized to represent at least four octaves of piano keys, covering the majority of beginner and intermediate repertoire while allowing future expansion.

This subsystem interfaces electrically with the Power Management Subsystem for high current 5 volt supply and logically with the MIDI Input and SD Card subsystems, which provide the note and timing data required for display generation. Verification of this subsystem will include measurement of refresh rate using an oscilloscope and confirmation that the display operates at or above 60 hertz without visible flicker.

Table 1 R&V Table for RGB LED Matrix Subsystem

Requirements	Verification
• The RGB LED Matrix Subsystem shall operate at 5 V and maintain a minimum refresh rate of 60 Hz to prevent visible flicker.	• Power the LED matrix using the 5 V supply. • Use an oscilloscope to measure the row scan frequency. • Confirm that the effective display refresh rate is at least 60 Hz. • Visually inspect the display to confirm absence of perceptible flicker.
• The subsystem shall correctly display at least four octaves of piano keys and represent note duration in real time.	• Load a test MIDI file covering four octaves. • Confirm that each MIDI note maps to the correct LED position. • Verify that note duration matches expected timing from the MIDI file.
• The subsystem shall safely interface 3.3 V MCU logic to 5 V display inputs using level shifting.	• Measure MCU output voltage levels. • Measure matrix input voltage levels after the 74HCT245 transceiver. • Confirm proper 5 V logic levels are achieved without signal distortion.

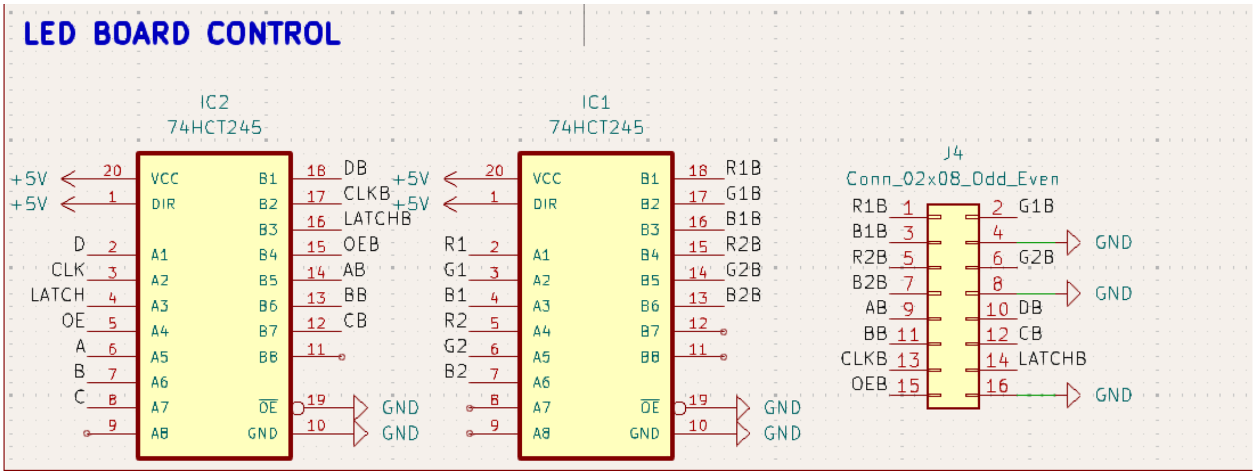

Figure 3 RGB LED Matrix Subsystem Schematic

2.2.2 SD Card Storage Subsystem

The SD Card Storage Subsystem is responsible for storing and providing access to Standard MIDI files used by the system for visual guidance and correctness evaluation. This subsystem directly supports the first high level requirement by ensuring that note number, velocity, and timing information can be reliably retrieved and processed in real time.

A microSD card socket is mounted directly on the PCB and interfaced to the STM32F446RET6 microcontroller using the SPI communication protocol. The SPI interface operates at a clock frequency selected to balance reliability and throughput, allowing the system to read MIDI files without introducing perceptible startup delay. Upon song selection, the microcontroller initializes the FAT file system, opens the selected MIDI file, and parses the file header and track data to extract note events, timing deltas, and velocity information.

The extracted MIDI event data is stored in memory buffers and used by the LED Matrix Subsystem to determine which keys to display and for how long. The parsed data also serves as the reference sequence for comparison against incoming MIDI or microphone input during practice mode.

The subsystem is designed to support multiple MIDI files stored on the SD card, allowing users to add or remove songs without modifying firmware. Verification of this subsystem will include confirming successful SPI communication, correct parsing of MIDI event data, and consistent reproduction of note timing and pitch across repeated file loads.

Table 2 R&V Table for SD Card Storage Subsystem

Requirements	**Verification**
● The subsystem shall successfully read and parse Standard MIDI files from a microSD card using SPI communication.	● Insert a formatted microSD card containing valid MIDI files. ● Power the system and select a song. ● Confirm that note number, velocity, and timing data are correctly extracted.
● The subsystem shall allow users to select among multiple stored MIDI files.	● Store at least three distinct MIDI files on the SD card. ● Navigate through the user interface and confirm correct file selection and playback.
● The subsystem shall operate without data corruption during repeated read operations.	● Repeatedly load and parse the same MIDI file at least 20 times. ● Confirm consistent and identical note output across trials.

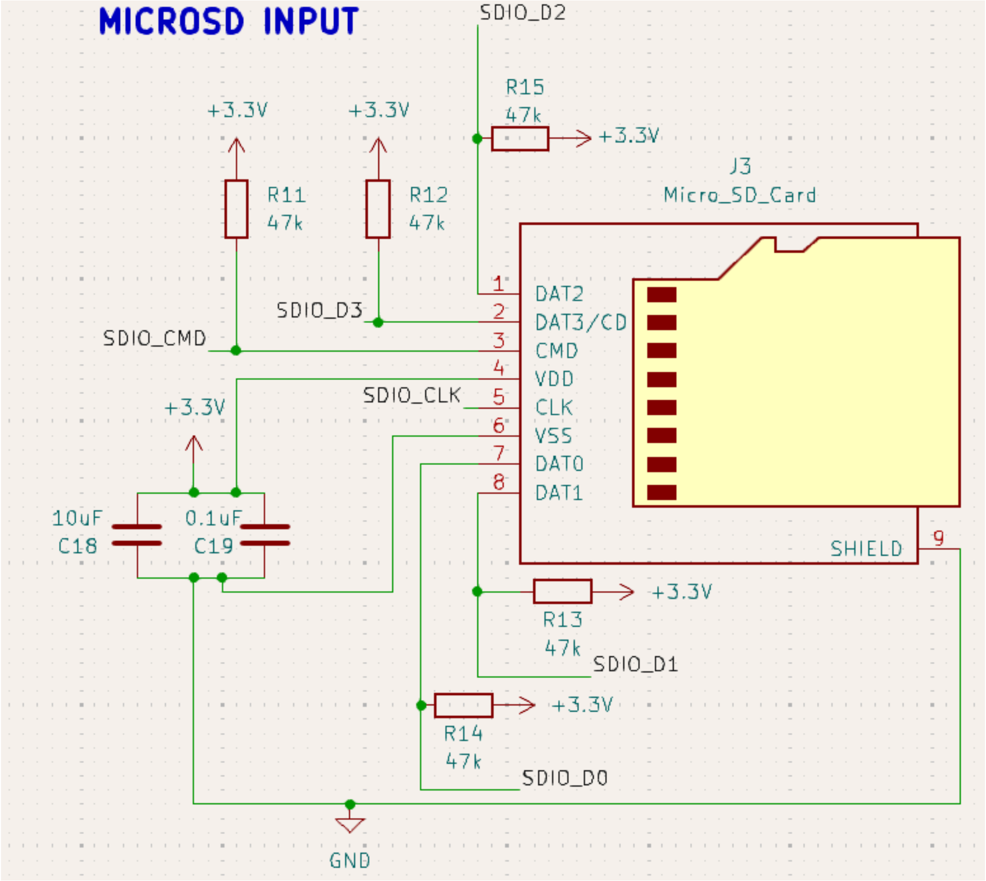

Figure 4 SD Card Storage Subsystem Schematic

2.1.3 MIDI Input Subsystem

The MIDI Input Subsystem is responsible for receiving real time note data from an external digital piano and providing low latency input to the system for correctness evaluation. This subsystem directly satisfies the second high level requirement by ensuring that Note On and Note Off events are detected and processed within the required 30 millisecond latency window.

A standard 5 pin DIN MIDI connector is used to receive input from the keyboard. MIDI communication follows the MIDI 1.0 electrical specification and operates as a 5 mA current loop at a baud rate of 31,250 bits per second using asynchronous serial transmission. To ensure electrical isolation between the external instrument and the embedded system, a 6N138 optocoupler is used in accordance with the MIDI standard. This prevents ground loops and protects the microcontroller from external voltage transients.

The output of the optocoupler is conditioned to produce a clean 3.3 volt logic signal compatible with the STM32F446RET6 UART peripheral. The microcontroller UART is configured to operate at 31,250 baud with 8 data bits, no parity, and one stop bit. Incoming MIDI bytes are buffered and parsed in firmware to detect Note On and Note Off messages, extract velocity values, and compute note duration based on timestamp differences.

The parsed MIDI events are compared against the expected note sequence extracted from the SD Card Subsystem. Correctness and timing accuracy are then passed to the RGB LED Matrix Subsystem for immediate visual feedback. Verification of this subsystem will include confirming correct baud rate operation using a logic analyzer, validating electrical isolation through continuity testing, and measuring end to end input latency to ensure it does not exceed 30 milliseconds.

Table 3 R&V Table for MIDI Input Subsystem

Requirements	**Verification**
● The subsystem shall accept standard 5 pin DIN MIDI input at 31250 baud.	● Connect a MIDI keyboard transmitting at 31250 baud. ● Use a logic analyzer to confirm correct reception of serial data.
● The subsystem shall detect Note On and Note Off events with latency not exceeding 30 ms.	● Trigger a note on the keyboard. ● Measure the time between MIDI signal transition and LED update using an oscilloscope. ● Confirm latency is ≤ 30 ms.
● The subsystem shall support detection of up to 10 simultaneous notes.	● Play 10 simultaneous keys on the keyboard. ● Confirm all notes are detected and displayed without dropped events.

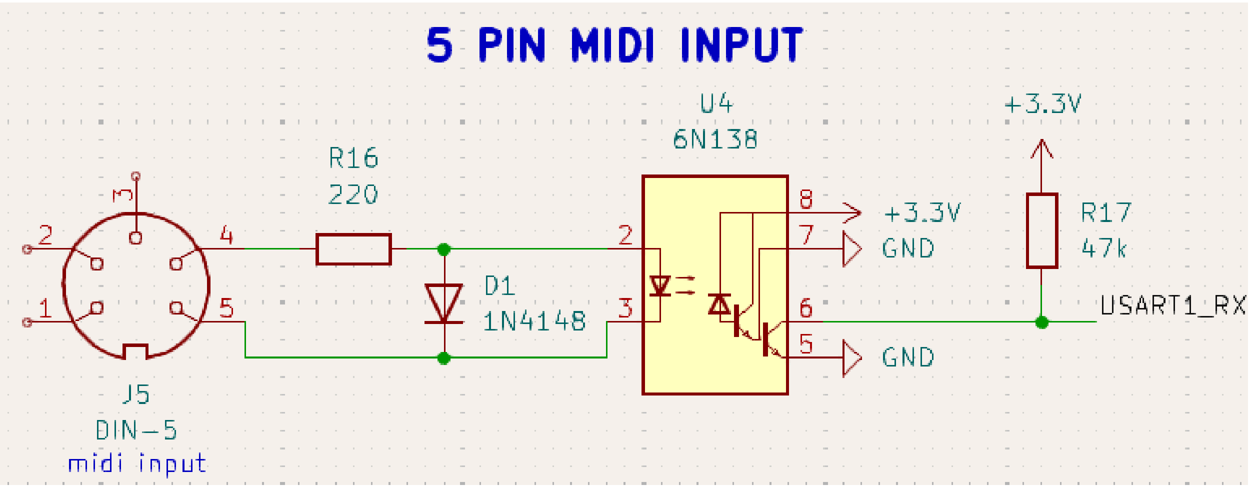

Figure 5   MIDI Input Subsystem Schematic

2.1.4 User Interface Subsystem

The User Interface Subsystem is responsible for allowing the user to select songs, adjust playback parameters, and control system operation. This subsystem enables interaction with the SD Card Storage Subsystem for song selection and configures operational modes that affect the MIDI Input and RGB LED Matrix subsystems.

User input is provided through physical push buttons connected to general purpose input pins on the STM32F446RET6 microcontroller. These buttons allow navigation through stored MIDI files, selection of playback modes such as normal, wait mode, and single hand practice, adjustment of playback speed, and control of start and stop functionality. Each button input is configured with appropriate pull-up or pull-down resistors and is debounced in firmware to prevent false triggering due to mechanical contact bounce.

System feedback related to menu navigation and operational state is displayed directly on the RGB LED Matrix. Simple text indicators and status symbols are rendered using predefined font patterns stored in memory. This approach eliminates the need for a separate LCD display while maintaining clear user feedback.

The subsystem interfaces electrically with the microcontroller GPIO pins and logically with the SD Card and MIDI subsystems by modifying configuration variables that affect playback behavior. Verification of this subsystem will include confirming correct detection of button presses under repeated operation, validating proper debounce behavior, and ensuring that menu selections result in the intended system configuration changes.

Table 4 R&V Table for User Interface Subsystem

Requirements	Verification
● The User Interface Subsystem shall detect physical button presses and update system state accordingly without false triggering.	● Press each button at least 20 times in succession. ● Confirm that each press results in exactly one detected input event. ● Confirm that no additional unintended state transitions occur due to mechanical bounce. ● Observe system logs or debug output to verify correct detection.
● The subsystem shall allow the user to select among multiple MIDI files stored on the SD card.	● Store at least three MIDI files on the microSD card. ● Navigate through the user interface using the push buttons. ● Confirm that each selected file loads correctly and displays the appropriate note data.
● The subsystem shall allow the user to adjust playback speed and operational modes including normal mode, wait mode, and single hand practice mode.	● Select each playback mode individually. ● Confirm that system behavior changes according to mode definition. ● Adjust playback speed and measure note timing to verify speed scaling is applied correctly.
● The subsystem shall provide visual feedback of menu navigation and system status on the RGB LED matrix.	● Navigate through menu options. ● Confirm that displayed text or status indicators change appropriately. ● Confirm that operational state changes are visibly reflected on the LED matrix.

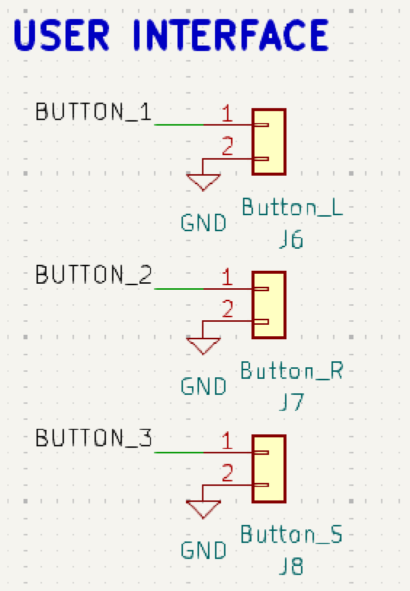

Figure 6 User Interface Subsystem Schematic

2.1.5 Power Management Subsystem

The Power Management Subsystem is responsible for providing stable and protected power to all system components. This subsystem ensures reliable operation of the RGB LED Matrix, microcontroller, and peripheral circuitry while preventing damage due to overcurrent conditions.

The system is powered by an external regulated 5 volt DC adapter rated for up to 4 amperes. The 64×32 RGB LED matrix operates directly from the 5 volt rail and can draw up to approximately 4 amperes under worst case full white brightness conditions. To provide sufficient margin and prevent voltage droop under peak load, a 5 V 4 A supply is selected.

The STM32F446RET6 microcontroller and supporting logic operate at 3.3 volts. A linear AMS1117-3.3 low dropout regulator is used to step down the 5 volt rail to 3.3 volts for logic level circuitry. Proper input and output decoupling capacitors are included in accordance with the regulator datasheet to ensure voltage stability and reduce ripple.

Overcurrent protection is implemented using a fuse placed in series with the 5 volt input to protect against excessive current draw or short circuit conditions, particularly from the LED matrix. Screw terminal blocks are used to provide mechanically secure power connections and reduce the risk of intermittent contact under load.

This subsystem interfaces electrically with all other subsystems by providing regulated supply rails. Verification of this subsystem will include measuring output voltage under maximum LED load conditions, confirming that the 5 volt rail remains within ±5 percent tolerance, verifying 3.3 volt regulation stability, and confirming fuse operation under simulated fault conditions.

We made a buck converter to ensure that the voltage is kept at 5 volt for the LED matrix.

Table 5 R&V Table for Power Management Subsystem

Requirements	Verification
● The subsystem shall provide a regulated 5 V supply capable of delivering up to 4 A to the LED matrix.	● Apply maximum LED load. ● Measure supply voltage using a multimeter. ● Confirm voltage remains within ±5% of 5 V.
● The subsystem shall accept GPIO from the user interface subsystem.	● Measure regulator output under normal operation. ● Confirm output remains within ±5% of 3.3 V.
● The subsystem shall include overcurrent protection.	● Simulate overcurrent condition. ● Confirm fuse interrupts excessive current flow.

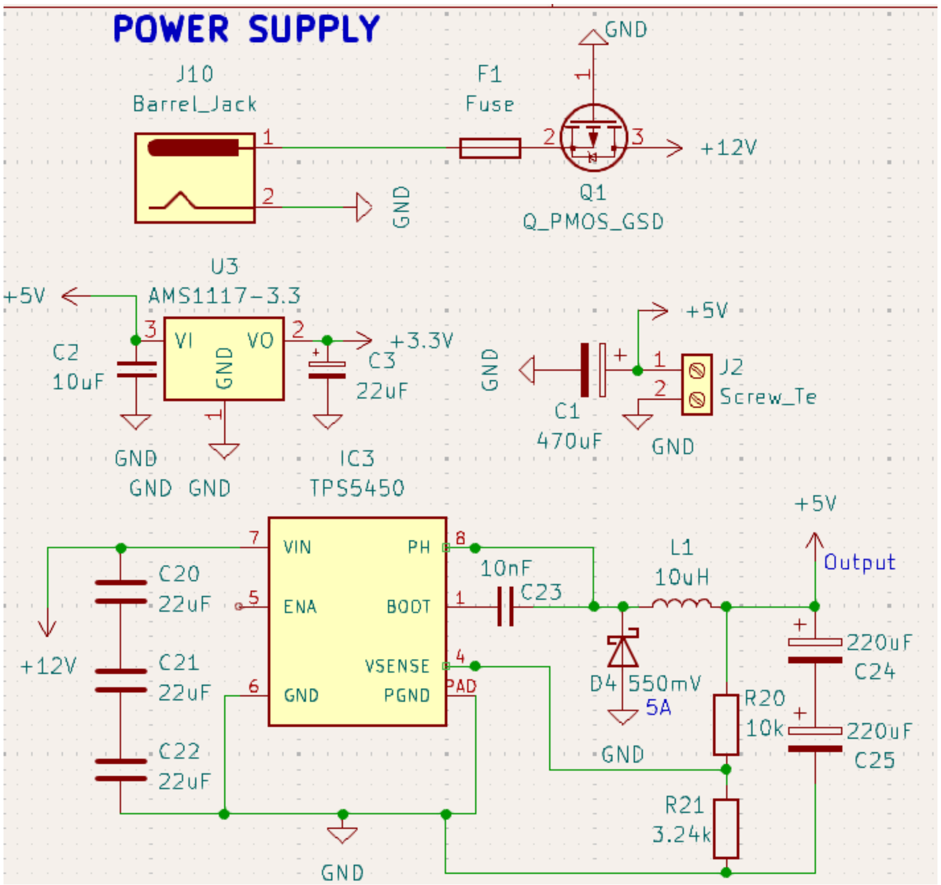

Figure 7 Power Management Subsystem Schematic

2.1.6 Control Subsystem

The Control Subsystem is centered around the STM32F446RET6 microcontroller operating at 3.3 volts and serves as the central processing and coordination unit of the entire system. This subsystem manages communication between all peripheral modules, performs real time data processing, and determines the visual output displayed on the RGB LED matrix.

The microcontroller interfaces with the SD Card Storage Subsystem via SPI to retrieve and parse MIDI file data, including note number, velocity, and timing information. Parsed data is stored in memory buffers and used as the reference sequence for correctness evaluation. The microcontroller also interfaces with the MIDI Input Subsystem through a UART peripheral configured at 31,250 baud to receive Note On and Note Off messages. For acoustic input, the microcontroller samples audio from the microphone subsystem using its ADC and performs FFT based signal processing to classify detected notes.

Incoming user input from the User Interface Subsystem is read through GPIO pins and processed to update system state, including song selection, playback mode, and speed adjustment. Based on the current mode and input data, the microcontroller compares expected note events with received MIDI or classified audio events and determines timing accuracy and correctness.

The processed results are transmitted to the RGB LED Matrix Subsystem using timer driven updates and Direct Memory Access to maintain deterministic refresh timing and meet the required 60 Hz display rate. The Control Subsystem therefore acts as the logical integration layer that coordinates sensor input, file data, user commands, and display output in real time.

This subsystem receives regulated 3.3 volt power from the Power Management Subsystem and is responsible for ensuring that all timing and latency requirements defined in the high level system requirements are satisfied.

Table 6   R&V Table for Control Subsystem

Requirements	**Verification**
● The subsystem is able to send correct DATA for the matrix.	● Check if the correct song is displayed on the matrix. ● Confirm if the musical input is compared to the actual notes and produce appropriate assessment. ● Confirm that note number, velocity, and timing data are correctly extracted.
● The subsystem is able to receive and read MIDI files from the microSD card	● Compare the data contained in the SD card files with those received from the SDIO pins of the STM32 to ensure the data received is correct
● The subsystem shall correctly receive analog inputs from the microphone and check which note they correspond to	● Test the input of the ADC pin of the STM32 for various input sounds and see if the system reads them and is able to use FFT to find their frequency

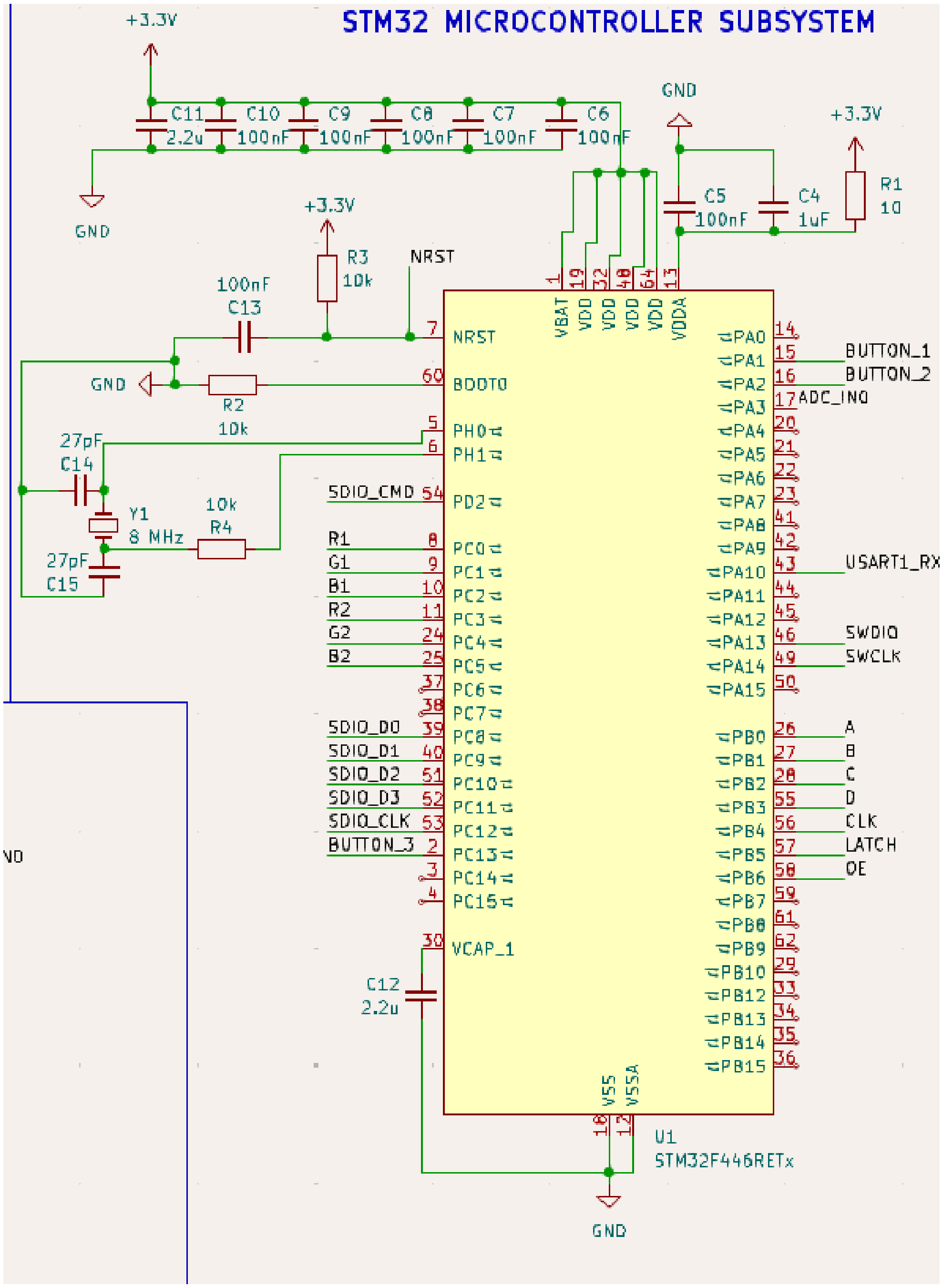

Figure 8 Control Subsystem Schematic

2.1.7 Microphone Subsystem

Table 7   R&V Table for Microphone Subsystem

Requirements	Verification
- Make sure that the single correct note is recognized	- Play a single note on the keyboard. - Confirm that the note is detected and displayed without dropped events.
- The subsystem shall detect Note On and Note Off events with latency not exceeding 50 ms.	- Play a note on the piano. - Measure the time between MIDI signal transition and LED update using an oscilloscope. - Confirm latency is ≤ 50 ms.
- The system shall take input from the mic that is connected to the system.	- Play a note through the mic. - Use a logic analyzer to confirm correct reception of serial data.

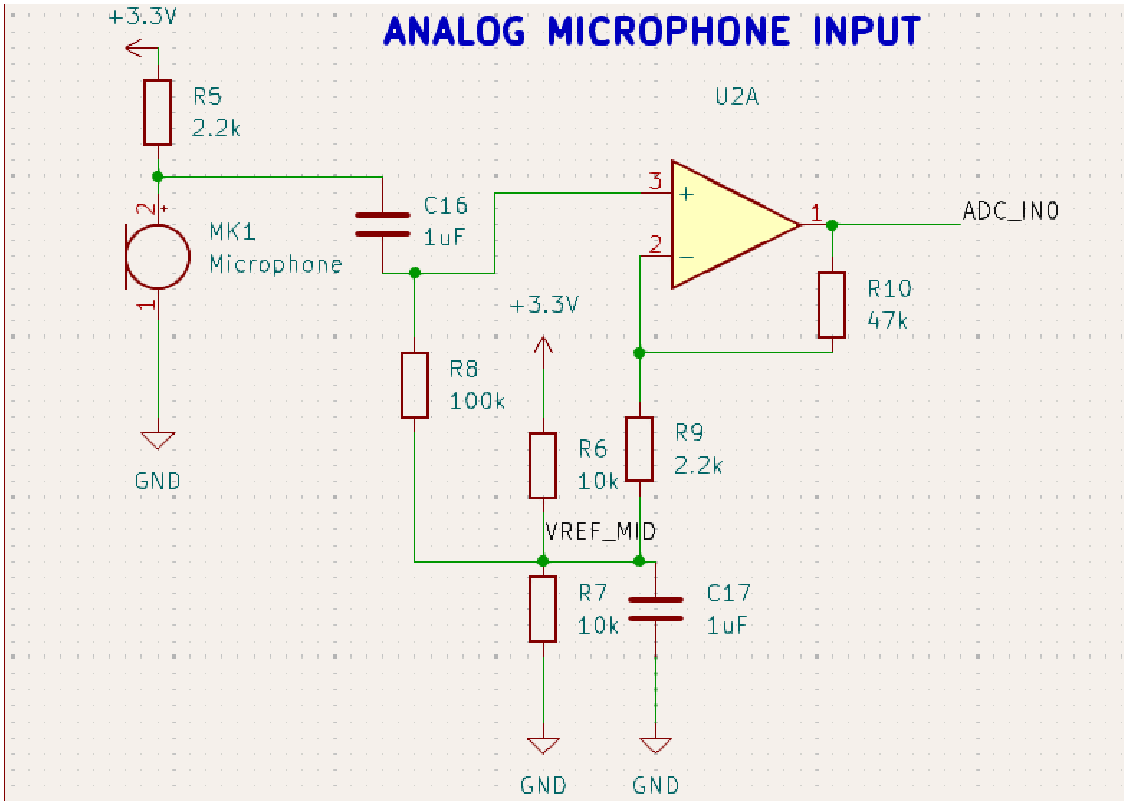

Figure 9   Microphone Subsystem Schematic

The Microphone subsystem receives input from the sound of the note through the electret microphone and sends the amplified output to the stm32 microcontroller. The analog microphone design also includes a coupling capacitor and bias resistor to remove the DC offset as well as a reference bias to reduce the noise from the ground. The R5 shown in Figure 9 allows the electret microphone to receive bias current in order to function. C16 and R8 provide a high pass filter. The op amp acts as a non amplifying inverter with a gain of 22, to allow the signal to be easily detected by the ADC pin of the STM32.

This subsystem directly satisfies the third high level requirement by ensuring that Note On and Note Off events are detected and processed within the required 50 millisecond latency window.

The input that was received from the microphone is compared against the expected note of the song that the user decided to play from the SD card. The timing and evaluation of the note is compared and the appropriate output is shown in the RGB LED matrix. We can verify this system by testing whether the note is recognized, detect note on and off within a latency of 50 ms, and test whether the note was received through the microphone. We plan on using a FFT function on the STM32 through the ARM CMSIS-DSP Library as our main goal is to efficiently and quickly detect the frequency of incoming notes rather than develop our own FFT algorithm. The plan for this part of the project is for it to take one note at a time and act as an alternative to MIDI cable input. Robust polyphonic detection seems unrealistic and unnecessary for a project of this scale.

2.3 Tolerance Analysis

The primary design aspect that poses risk to a successful completion of our project and meeting all of our high level goals is the microphone subsystem. The main challenge here is going to be to find a balance between accuracy and speed, as matching audio input from a piano to predefined notes would require digital signal processing techniques such as FFT analysis. We will have to select an appropriate point size for the FFT and ensure that the LED display stays flicker-free and the response time for the input is still reasonable.

The MCU we plan to use (STM32F446RE) has a maximum internal clock speed of 180MHz. Performing an FFT using the CMSIS-DSP library takes varying amounts of time depending on the number of points in the FFT, 512 points is under 0.1 millisecond, 1024 points is under 0.2 milliseconds, 2048 points is up to 0.3 milliseconds and 4096 points is around 3 milliseconds. From this, the 2048 appears to be the highest possible accuracy without an unreasonable waiting time for the user.

To maintain 60hz on a 64x32 matrix (HUB75) with a 1/16 scan rate, we essentially need the MCU to give 16 orders, 60 times a second which leads to 960hz to maintain the display. 960 hertz means we need to send new data to the display every 1.04 milliseconds, since the 2048-point FFT takes 0.3ms, we have around 0.7ms to match the FFT to the note and tell the MCU which LEDs to turn on which gives us enough time to do so, meaning it is possible.

3. Cost and Schedule

3.1 Cost Analysis

Table 8 Cost Analysis Table

Part Name	Manufacturer	Quantity	Cost (USD)	Link
STM32F446RET6 microcontroller	STMicroelectronics	1	8.15	Here
64x32 RGB LED Matrix (HUB75)	Adafruit	1	39.95	Here
74HCT245 transceiver	ON Semiconductor	3	2.07	Here
Micro SD Card Socket	Adafruit	1	1.95	Here
Micro SD Card	Solid Run STD	1	12	Here
Barrel Jack 5v	adafruit	1	0.95	Here
Screw Terminal	Amphenol Anytek	1	0.54	Here
AMS1117-3.3 low dropout regulator	UMW	1	0.29	Here
2 Pin Connector	TF Connectivity AMP Connectors	3	2.76	Here
DIN-5, 5 Pin Midi Jack	Adafruit	1	1.75	Here
6N138 Optocoupler	Lite-ON Inc	1	0.82	Here
Fuse	Bourns Inc.	1	0.64	Here
Electret Mic	Same Sky	1	0.79	Here

Amplifier	Microchip Technology	1	0.48	Here
5 pin connector	Tinkerspace	1	1.99	Here
02x08 connector	Sparkfun	1	0.95	Here
1k Ohms	Yageo	2	0.2	Here
220 Ohms	Yageo	1	0.1	Here
1N4148 Diode	Diodes Incorporates	1	0.1	Here
47k Ohms	Yageo	7	0.7	Here
Green LED	Rohm Semiconductors	1	0.51	Here
Red LED	Rohm Semiconductors	1	0.53	Here
0.1uF Capacitor	Yageo	8	0.8	Here
10uF Capacitor	Yageo	2	1.3	Here
10k ohms	Yageo	6	2.4	Here
10 ohms	Yageo	1	0.1	Here
Ceramic Resonator 8MHZ	YIC	1	0.14	Here
2.2u Capacitor	Yageo	2	0.22	Here
27pF Capacitor	Murata Electronics	2	0.2	Here

1uF Capacitor	TDK Corporation	3	0.36	Here
2.2k Ohms	Yageo	1	0.1	Here
100k Ohms	Yageo	1	0.11	Here
PMOS	Adafruit	1	2.95	Here
22uF capacitor	Murata Electronics	4	3.48	Here
470uF capacitor	Nichicon	1	0.8	Here
10nF capacitor	KEMET	1	0.13	Here
10uH Inductor	Abracon LLC	1	0.1	Here
550mV Schottky Diode	Diodes Incorporated	1	0.15	Here
200uF Capacitor	Chemi-Con	2	4.28	Here
3.24k Ohms	Stackpole Electronics Inc	1	0.1	Here

Total: 95.7 dollars in raw cost.

Labor cost per person should be 30 dollars/hour (USD)*2.5*80 hours = 6,000 dollars per person for this project. In total, the labor cost of the whole project would be 18,000 dollars.

3.2 Schedule

Table 9 Planned Schedule Table

Time	Work	Person
02/23~03/01	Finalize design document and PCB layout. Submit first PCB order. Prepare for Design Review.	Everyone Nuwan (Finalize PCB layout)
03/02~03/08	Complete Design Review. Assemble breadboard prototype. Prepare for Breadboard Demo.	Everyone
03/09~3/15	Breadboard Demo. Begin PCB assembly once boards arrive.	Everyone(Demo)
03/16~03/22	Spring Break	Everyone

03/23~03/29	Complete PCB soldering. Implement SD card interface and MIDI parsing firmware. Begin subsystem testing.	Nuwan, Jay (soldering)
		Sarayu (Testing subsystem)
		Jay(SD card)
		Nuwan(MIDI parsing firmware)

03/30~04/05	Integrate LED matrix driver with parsed MIDI data. Begin full system integration.	Nuwan(Integrate LED matrix) Sarayu, Jay(Full system integration)

04/06~04/12	Progress Demo. Demonstrate integrated LED + MIDI functionality. Implement microphone FFT subsystem.	Everyone(Demo)
		Jay(Implement microphone FFT subsystem)

04/13~04/19	Complete system integration. Debug timing, accuracy, and stability issues. Conduct full verification testing.	Sarayu, Jay (Debug) Nuwan, Sarayu (full verification testing)
04/20 ~ 04/26	Mock Demo and presentation practice. Finalize enclosure and polish user interface behavior.	Everyone
04/27 ~ 05/03	Final Demo and Final Presentation.	Everyone

05/04	Final paper submission, lab checkout, documentation completion.	Everyone

4. Ethics Safety and Societal Impact

This project follows the professional responsibilities outlined in the IEEE Code of Ethics. We ensure that all reported performance metrics, including timing accuracy and response latency, are based on measured data rather than ideal estimates. All design decisions, testing procedures, and failures are documented to support transparency and reproducibility.

Since our proposal, the main changes we have made is adhering to the MIDI standards by using an optocoupler as well as the inclusion of testing LEDs for both of our PCB voltage values to provide a clear indication of if the power is on. We also considered the risk of buttons being directly on the board to users, and thus switched to 2 pin connectors, with which we will use external buttons outside of the pcb casings.

Protecting the end users is of utmost importance and is primarily done through adhering to safety standards by using a fuse to prevent fire hazards during short circuits, limiting user interaction directly with the pcb itself and the use of an optocoupler to prevent electrical spikes. The visual human interaction with the system is also adhering to safety levels by ensuring that the brightness of the LED board is suitable for closer range use, preventing eye strain. The frequency of the LED board is also kept at 60hz, to prevent excessive flickering which can cause motion sickness.

From a safety perspective, the system uses a confirmed DC Buck converter design from Texas Instruments, and a standardised 12V wall adapter. A fuse protects against excessive current draw from the LED matrix, and a voltage regulator provides stable 3.3 volt power to logic components. The MIDI interface includes an optocoupler to electrically isolate the keyboard from the microcontroller, preventing ground loops and protecting connected equipment. The environmental damage is minimized as it uses a max of 12 volts of external supply.

Societally, the device promotes accessible music education by providing a standalone learning tool that does not require internet access or expensive proprietary hardware. It supports independent learning while remaining affordable and safe, aligning engineering practice with positive community impact. From an economic perspective, the device offers a cheaper alternative to current solutions which require either smart devices and internet connectivity or access to music teachers which can heavily vary by region and requires high amounts of money compared to our solution. On a global and cultural level, access to learning about music and instruments is a prominent part of many cultures, but access to such tools can heavily depend on one's socioeconomic status as well as the technology available in their region. One of our devices aims to allow an alternative approach to learning the piano, which does not require as many resources or costs. One particular change we made was switching from a specialised 5V 4A wall adapter to a more standard 12V one and including a DC buck converter on the pcb as standard equipment is much more widely available and cheaper to replace. The materials used in this product are all safe in regards to home usage, and the planned casing will allow the product to be accessible to almost all age groups.

According to the I. of IEEE Code of Ethics[3], we need to make a design that holds safety of the public, and our product is created to fit the ethical design that is purposed for creating something that is beneficial for the public, and it complies with safety standards as we operate under 5 volts. We disclosed all the possible factors of the product that might pose a threat in the second paragraph.

We are adhering to MIDI 1.0 Specification standards[6] by using an optocoupler which provides galvanic isolation, preventing high voltage surges which can damage other electronic parts as well as create shocks.

Additionally, we take criticism of our product from peers, faculties, professors, and users, which is part of I. in IEEE code of ethics[3]. Our members have gone through basic safety training and technological experiences for creating the product and will give each member credit for the contribution to the product.

Our product meets Illinois Information Technology Accessibility Act Standards 1.0 as our product utilizes “input method that complies with Telecommunication Products”[2]. In addition, our product does not cause “screen flicker greater than 2Hz and lower than 55Hz.”

While in the process of making the product, we will follow the safety rules of ECEB, and we will assemble and solder the product in the lab areas, using safe practices. Some of these safe practices include using proper ventilation and eye protection when soldering to prevent the inhalation of harmful fumes and injuries. Due to the use of high currents (up to 4A) for the LED board, we used very thick traces on our pcb. During the initial testing phase of the PCB, we plan to begin testing with only the power supply subsystem with the fuse in place to ensure that any issues are promptly seen in the event of a failure. The inclusion of 2 LEDs on the PCB serves to protect developers during the testing phase as it can indicate to us the level of current through the system depending on the brightness and functionality of the LEDs, rather than requiring manual testing, reducing the risk of human contact with high current traces.

References

[1] Adafruit Industries. (2020). 64x32 RGB LED Matrix - 6mm pitch. Adafruit.com. https://www.adafruit.com/product/2276?gad_source=1&gad_campaignid=21079227318&gbraid=0AAAADx9JvRWYU5tJsMEXvWKXJLSAmWvC&gclid=CjwKCAiAtLvMBhB_EiwA1u6_Pt3TwoBxsc8LnE150b79cn8LxB16iv_X7rbdVaUUnf4V9SRT6fXxnBoCLwgQAvD_BwE

[2] Getting started with STM32F4xxxx MCU hardware development. (2018). https://www.st.com/resource/en/application_note/an4488-getting-started-with-stm32f4xxxx-mcu-hardware-development-stmicroelectronics.pdf

[3] Illinois Information Technology Accessibility Act Standards 1.0. (2026). State.il.us. https://www.dhs.state.il.us/IITAA/IITAAStandards.html

[4] Institute of Electrical and Electronics Engineers. (2020). IEEE Code of Ethics. https://www.ieee.org/content/dam/ieee-org/ieee/web/org/about/corporate/ieee-code-of-ethics.pdf

[5] Labs, L. (2024, March 30). Tang Nano 9K: HUB75 LED Panels. Lushay Labs. https://learn.lushaylabs.com/led-panel-hub75/

[6] MIDI 1.0 Detailed Specification. (2024, January 19). MIDI.org. https://midi.org/midi-1-0-detailed-specification

[7] Power Designer. (2026). Ti.com. https://webench.ti.com/power-designer/switching-regulator/select

[8] SD-50BV Symbol, Footprint & 3D Model by Same Sky | SnapMagic Search (formerly SnapEDA). (2026). Snapeda.com. https://www.snapeda.com/parts/SD-50BV/Same%20Sky/view-part/

[9] STM32F446RET6 | DigiKey Electronics. (2026). DigiKey Electronics. https://www.digikey.com/en/products/detail/stmicroelectronics/STM32F446RET6/5175962

[10] STM32 PCB Design (2022). Phil's Lab. https://www.youtube.com/watch?v=PMEpQZ90f34&list=WL&index=2&t=1789s

[11] "Piano." Wikipedia, February 23, 2026. https://en.wikipedia.org/wiki/Piano.

[12] riktw. (2020, April 27). Driving a 64*64 RGB LED panel with an FPGA. - jaeblog. Jaeblog. https://justanotherelectronicsblog.com/?p=636

[13] Texas Instruments 74HC245: Symbol, Footprint, 3D STEP Model | Ultra Librarian. (2024). Ultralibrarian.com. https://app.ultralibrarian.com/details/01cf4bd4-1075-11e9-ab3a-0a3560a4cccc/Texas-Instruments/74HC245?exports=42&open=exports

[14] Wong, Amanda. “Learn More about SD Card, Mini SD Card & Micro SD Card.” iBoysoft, February 26, 2026. https://iboysoft.com/wiki/sd-card.html.