An-Najah National University

Faculty of Engineering & Information Technology

Presented in partial fulfillment of the requirements for

Bachelor degree in Computer Engineering

Hardware Graduation Project

ROBHOVEN

FEEL THE MUSIC,
FOLLOW THE LIGHT

Prepared By:

Dana Hamdan

Shahd Salahat

Supervised By:

Dr. Emad Natsheh

June, 2025

Acknowledgment

We would like to express our sincere gratitude to all those who supported and guided us throughout the completion of this project.

First and foremost, we would like to thank our supervisor, Dr.Emad Natsheh for his support, guidance, and invaluable advice.

We also extend our thanks to Eng. Abdullah Hinnawi for his assistance and valuable help during this project.

We are deeply grateful to An-Najah National University for providing the facilities that was crucial for completing this project. The supportive environment and resources greatly contributed to our work.

Finally, we would like to thank our families and friends for their unconditional support and encouragement. Their belief in us kept us motivated, and their help and presence made this journey more enjoyable and memorable.

Disclaimer

This report was written by Dana Hamdan and Shahd Salahat in fulfillment of the requirements for a Bachelor’s degree in Computer Engineering. It has not been altered or corrected, other than editorial corrections, as a result of assessment and it may contain language as well as content errors. The views expressed in it together with any outcomes and recommendations are solely those of us. An-Najah National University accepts no responsibility or liability for the consequences of this report being used for a purpose other than the purpose for which it was commissioned.

Contents

List of Figures

Abstract

This project presents the design and implementation of a robotic system capable of playing a physical piano and guiding users through an interactive and gamified piano learning experience. The idea originated from the need to make music education more engaging and accessible by combining robotics, visual feedback, and game-like elements. Built around an Arduino Mega, the system uses servo motors to press piano keys and an LED strip mapped to individual notes for real-time visual guidance.

The robot operates in multiple modes. In **Performance Mode**, it automatically performs predefined melodies. In **Practice Mode**, the system lights up the keys the user should press and displays the name of the upcoming note on an LCD screen, helping beginners learn note positions and timing. Once familiar, users can enter **Practice Test Mode**, where the LEDs are turned off and the user is tested on their memory and accuracy. During this mode, the **LCD displays what the user presses in real time**, allowing immediate feedback. At the end of the test, the mobile app shows the final score unless the user makes too many mistakes, in which case the session ends immediately with corrective feedback.

An additional feature, **"Guess the Melody" Mode**, challenges users to identify a melody played by the robot. After listening, users select the correct song title from multiple options through a connected web app. Communication between the Arduino Mega and the web interface is handled via an ESP8266 Wi-Fi module, enabling song selection.

This integrated system blends robotics, music, and interactive software to deliver a hands-on, educational experience. It is designed to make piano learning more intuitive, fun, and accessible especially for visual and self-paced learners.

Chapter 1

Introduction

1.1 Background

Learning to play the piano can be difficult and time-consuming, especially for beginners without access to instructors or structured lessons. Traditional methods require reading sheet music, constant practice, and strong self-discipline. While digital tools and apps help to some extent, they lack the physical interaction of a real piano and often don't provide hands-on learning with actual keys. The idea for this project ROBHOVEN emerged from the need to make music education more accessible, fun, and interactive by combining robotics, electronics, and a physical keyboard. The project transforms a standard piano or keyboard into a smart educational tool that not only plays melodies but also helps users practice and learn them in an engaging way.

1.2 Objectives

The main objective of ROBHOVEN is to develop a robotic system that can press piano keys accurately to play melodies and also assist users in learning how to play them. It includes multiple learning modes such as visual guidance through LEDs, note display via an LCD screen, and interactive feedback through a test mode. The system should be able to:

- Automatically perform songs using servo-controlled key presses.
- Guide users to play notes using an LED strip and LCD display.
- Allow practice tests that track user input and display scores.
- Include a guessing game where the robot plays a melody and users identify the song via a connected web interface.

1.3 Significance

This project is significant because it introduces a new approach to piano education, especially for beginners and children who benefit from visual, hands-on learning. Most musical education tools are either software-only or require manual practice with limited feedback. ROBHOVEN fills this gap by offering an interactive and automated physical experience. It also showcases how robotics can be applied in creative and educational ways beyond industrial or domestic automation. With the added web interface and scoring features, the system provides not just instruction, but also motivation and challenge making learning more enjoyable.

1.4 Organization of the Report

This report is structured to clearly present the design, development, and testing of the **ROBHOVEN** system. It consists of the following sections:

- **Introduction** – Provides an overview of the project, highlighting the motivation behind building a piano-playing robot and the limitations of traditional piano learning methods.
- **Literature Review** – Reviews existing musical learning tools and robotic instruments, identifying the lack of physical, interactive learning systems and justifying the need for ROBHOVEN.
- **Methodology** – Explains the system's hardware and software design, covering the components used (Arduino Mega, ESP8266, servos, LED strip, LCD, Switches), and the web-based interface.
- **Results, Analysis, and Discussion** – Evaluates the system's performance across different modes: automatic playback, learning mode, test mode, and the melody guessing game. It discusses accuracy, responsiveness, user engagement, and areas that can be improved.
- **Conclusion and Recommendations** – Summarizes the achievements of the project and suggests future enhancements, such as improved servo coordination, expanded song libraries, and better user interface integration.

Chapter 2

Literature Review

Over the years, robotics and music have come together in creative ways—robots that can play instruments, apps that help people learn, and systems that try to make music more interactive. Most of these tools, however, tend to focus on either performance or learning—not both. And while many digital apps have made learning music easier, they often miss the physical, hands-on experience of playing a real instrument.

Some early examples, like Toyota's **Partner Robot**, [1] showed how robots could physically play instruments like the trumpet and violin with impressive accuracy.
Then there's **Teotronico** [2] a robot designed specifically to play the piano and even sing along. While both systems were technically advanced, they were mostly built for show. They didn't try to teach music or help beginners learn .

On the other hand, apps like **Yousician** and **Simply Piano** [3]. Have become popular among learners. These tools use real-time feedback and visual cues to teach songs, but they only work on tablets or phones. You still need to look back and forth between the screen and your instrument, and there's no physical guidance or interaction—just visuals and audio.

A few systems have tried to bridge that gap by adding LEDs above the piano keys to guide beginners on what to press. Some of these use microcontrollers and basic logic to light up the correct keys. But even then, they lack real feedback or testing—you don't know if you've actually learned the song until you try it yourself.

This is where **ROBHOVEN** stands out. It's not just a robot that plays piano—it also *teaches*. It shows you the next note on an LCD, lights up the correct key with an LED, and physically plays the keys so you can watch and learn. Once you've practiced a melody, it lets you test yourself: the LEDs turn off, and you try to play it on your own. If you get it right, the system scores you and displays your performance on the LCD. If that wasn't enough, ROBHOVEN also includes a fun "**Guess the Melody**" game. It plays a tune, and you try to identify it from a list on a connected web app.

Unlike most other systems, ROBHOVEN mixes physical interaction, guidance, testing, and entertainment into one package. It turns learning into an engaging, hands-on experience—and brings a human touch to learning music with a robot.

Chapter 3

Methodology

This section outlines the design, construction, and operational methodology of RobHoven, a robotic system developed to assist users in learning the piano through real-time note guidance, physical finger movements, and interactive feedback across different learning modes..

3.1 System Structure

RobHoven's architecture consists of three primary modules: the mechanical framework, electronic control system, and user interface components. The mechanical structure is designed to control **two robotic hands** with precision, allowing them to traverse the piano keyboard and articulate finger movements accurately. It also supports visual and auditory cues to enhance the learning experience. Figure 3.1 illustrates the overall mechanical design, showing **two robotic hands** mounted on a linear motion system that enables horizontal movement across the keyboard.

Figure 3.1: Robot Body Structure.

3.1.1 Key Components of the Robot

3.1.1.1 Robotic Hand and Fingers:

The robotic system features **two hands**, each equipped with **three independently actuated fingers**, driven by MG90S servo motors. These servo motors offer a rotation range from 0° to 180°, allowing the fingers to replicate natural piano playing motions. **Each of the six fingers** is calibrated to press individual keys with precision, and calibration routines ensure proper alignment, accurate positioning, and consistent force application during operation.

Figure 3.2: Robotic Hand and Fingers.

3.1.1.2 Visual Cue System (LEDs)

- Fourteen LEDs are positioned above the white keys (C3 to B4) to indicate the next note in a melody.
- Due to limited I/O pins, two serial-in parallel-out shift registers are employed to control all LEDs efficiently, turning on/off the appropriate cues dynamically.

The LCD display shows the next note on Practice mode and the current note that the user press it on test.

Figure 3.3:14 LEDs positioned above the keys.

Figure 3.4: Mini switches positioned under the 14 keys.

Figure 3.5: Displaying The Next Note Using the LCD

3.1.2 Linear Motion System

The **hands are** mounted on two parallel 8mm stainless steel rails, guided by linear ball bearings, facilitating smooth horizontal movement across the keyboard..

The movement is driven by NEMA17 stepper motors attached to timing belts controlled via TB6600 3.5A stepper driver, allowing precise positioning.

Limit switches are installed at both ends of the linear axis to define travel limit and prevent mechanical overrun, with a dedicated home switch to reset positions at startup or after sequences.

Figure 3.6: Linear Motion System.

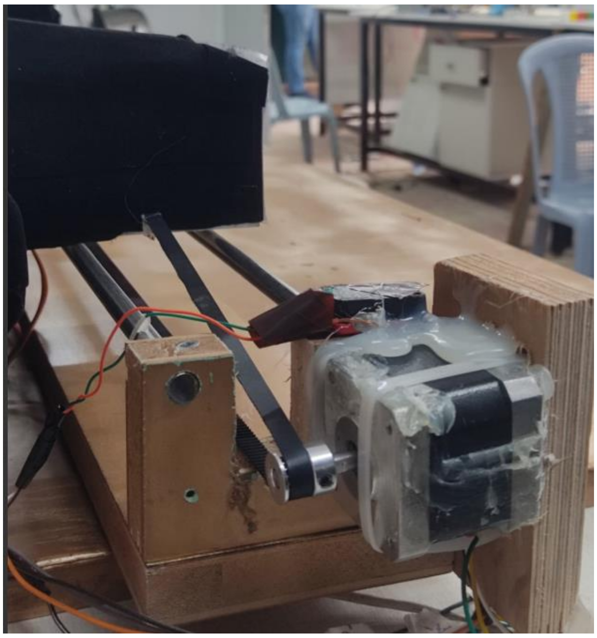

Figure 3.7: Moving Hands Using NEMA 17 Stepper Motor.

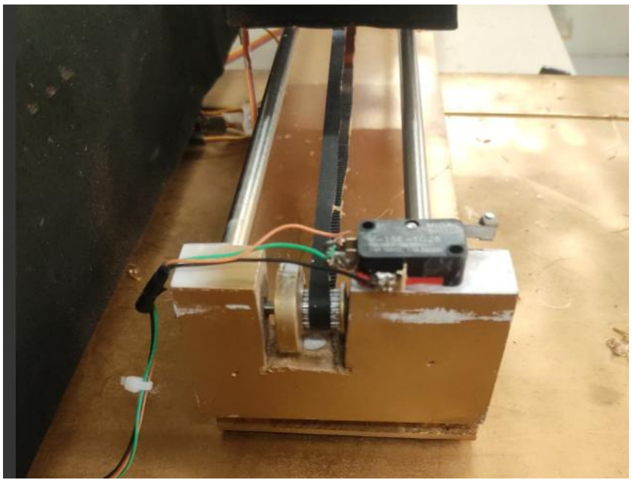

Figure 3.8: Timing belt attached to Motors.

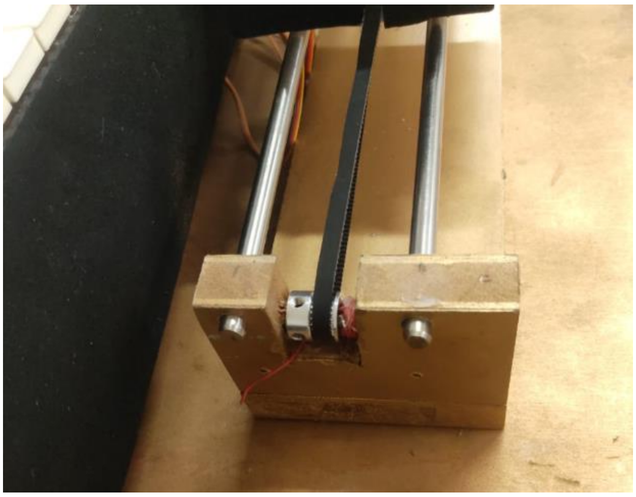

Figure 3.9: Two parallel 8mm stainless steel rails.

3.1.2.1 Hand Positioning and Homing

- The system performs homing routines at startup, moving the **hands to predefined initial positions** using limit switches. During operation, the Arduino and ESP32 controllers manage **hands movement commands**, ensuring accurate navigation to target keys for both performance and game modes.mode

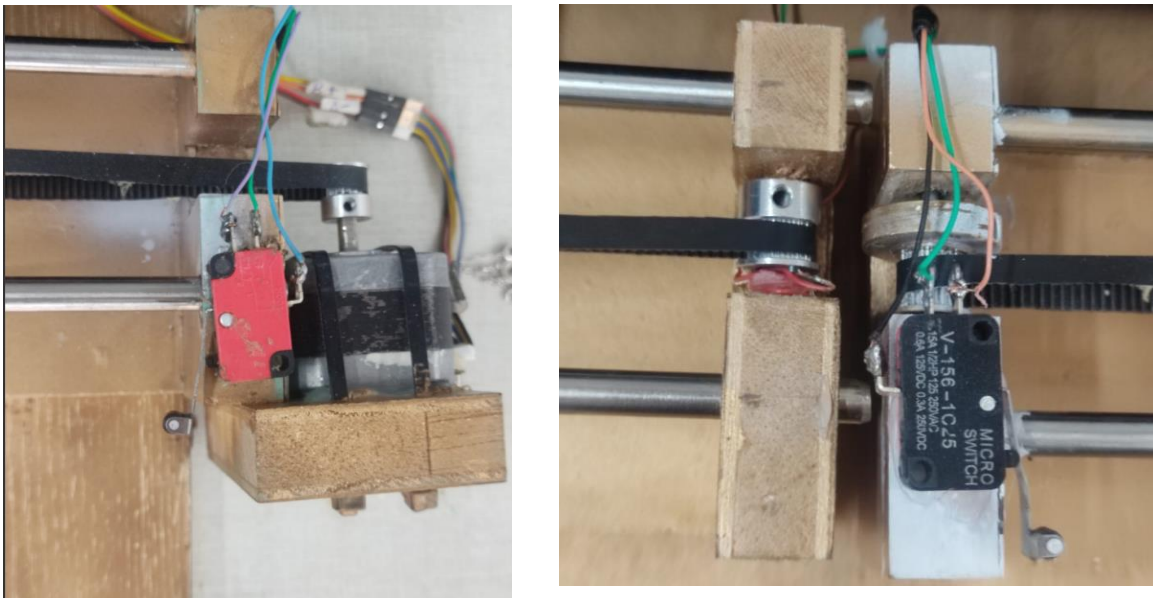

Figure 3.10: Limit Switches for homing.

3.2 Electrical and Feedback Systems

3.2.1 Servo and Finger Control

- The six servos controlling the robotic fingers receive PWM signals from the microcontrollers. Each servo is carefully calibrated to deliver accurate pressing force and precise finger placement on the piano keys.

- The robot uses optimized algorithms to determine the most suitable finger paths—not necessarily the shortest—ensuring smooth, natural-sounding melodies while maintaining correct timing and eliminating unnecessary delays.

Figure 3.11: Finger controlled using MG90S Servos.

3.2.2 Visual Cue System (LEDs)

- A total of 14 LEDs are positioned above 14 physical piano keys (white keys from C3 to B4). These LEDs serve as visual cues to indicate which key the user should press next. Due to I/O limitations, the LEDs are controlled through two serial-in parallel-out shift registers (each with 8 outputs from QA to QH0), allowing efficient management of all 14 LEDs using minimal pins.
- The LCD works in tandem with the LED cues where it displays critical information such as the upcoming note on practice mode , current mode on test mode, and sensor inputs (limit switches) to create an integrated feedback system, guiding the user visually and aurally for an effective learning experience.

3.2.3 Audio Feedback

- The system incorporates a DFPlayer Mini module connected to speakers, providing clear and engaging audio feedback that enhances user interaction. Upon startup, the system plays a welcoming message to greet the user. For each mode Practice Mode, Performance Mode, and Game Mode the audio system delivers spoken instructions, guiding the user through the operation and expectations of that mode.

- During gameplay or testing sessions, the system also plays distinct sounds based on user performance. A positive audio cue is triggered when the user successfully completes a task or wins a round, while a negative feedback sound is played in the event of an incorrect input or loss. This auditory layer reinforces learning, maintains engagement, and helps user's quickly associate sounds with performance outcomes.

3.3 User Interface and Mode Management

The system interfaces with a mobile, allowing users to select modes: **Practice**, **Performance**, or **Guess the Melody**.
In practice mode, users input melody names, and the system retrieves note sequences via an Audio-to-Note Extraction System.
The LCD display shows the current note, and LEDs light up to guide the user.
During practice, the system waits for user key presses, detected via limit switches, to verify correctness, update cues, and provide feedback.
The app manages mode-specific workflows, including scoring, retry options, and game-over conditions.

3.4 Operational Workflow

3.4.1 Initialization

Power-on initializes all components, performs homing routines for hands and linear axes.
The system calibrates servos and ensures all sensors and feedback systems are operational.

3.4.2 Practice Mode

1. Audio instructions are delivered through speakers to guide the user.
2. The user enters a melody name via the app.
3. The system fetches the note sequence from an Audio-to-Note Extraction System.
4. The first note is displayed on the LCD, and the corresponding key’s LED lights up.
5. The user should follow the instruction and press the right key.

Upon success:

1. The LED turns off and the next LED turns on.
2. The LCD updates with the next note.
3. The process repeats until the melody completes.

After finishing:

1. Audio instructions are delivered through speakers to guide the user.
2. The system prompts for a replay or transition to a test.
3. For testing, LEDs turn off, and the LCD displays the pressed note, with correctness verified via limit switches.
4. The system tracks errors, enforces a maximum number of wrong attempts, and displays scores accordingly.

3.4.3 Performance Mode

1. The user selects performance mode via the app.
2. The user enters a melody name via the app.
3. The system retrieves the melody’s notes via Audio-to-Note Extraction System.
4. The robot moves from the home position to the starting note using the optimal finger path, ensuring accurate timing and sound quality.
5. The robot plays the melody with precise timing, mimicking human-like finger movements to produce accurate sound output.

3.4.4 Guess the Melody Mode

1. The robot plays a random melody.
2. The app displays three options for the user to guess.
3. The system evaluates the user's choice and provides feedback, including correctness

3.5 Implementation Details

3.5.1 Audio-to-Note Extraction:

Convert any commercial song from YouTube into a clean, machine-readable stream of note names that can drive the RobHoven piano robot or any other downstream MIDI-style consumer.

- Accept a song title from the user
- Search YouTube, pull down best-quality audio, transcode to MP3
- Split audio into two stems (vocals & accompaniment)
- Run frame-level pitch estimation on the vocal stem
- Filter out micro-events (< 30 ms) and convert MIDI numbers → note names.

•

3.5.2 Synchronization Between two Hands:

To achieve synchronized parallel performance between the two hands, RobHoven employs an ESP32 microcontroller to split execution logic. The first robotic hand is controlled by the Arduino Mega, while the second hand is operated independently via the ESP32. Melody data is divided and sent to each controller, allowing both hands to press keys simultaneously without delay, ensuring precise coordination during complex performances.

3.6 Hardware components

3.6.1 Microcontrollers

Arduino Mega 2560

The Arduino Mega 2560 is the main controller of the project. It receives signals from sensors, processes the data, and controls motors, Servos, switches, LEDS, and LCD. It ensures the robot moves correctly , making real-time decisions for smooth operation.

Figure 3.12: Arduino Mega2560

3.6.2 Motors and Drivers

3.6.2.1 Stepper Motors

Two NEMA 17 Stepper Motor: The motors drive the two parallel 8mm stainless steel rails, guided by linear ball bearings, facilitating smooth horizontal movement across the keyboard.

Figure 3.13: NEMA17 Stepper Motor.

3.6.2.2 Two Stepper Motor Drivers (TB6600): Provide the necessary control and power to the stepper motor.

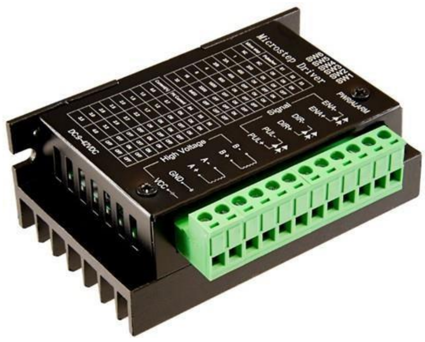

Figure 3.14: TB6600 Stepper Motor Driver

3.6.2.3 MG90S Servos Six MG90S Servos: Each finger is actuated by an MG90S servo motor, capable of rotating from 0° to 180° (or reversed for opposing hand orientation)

Figure 3.15: MG90S Servos

3.6.3 Sensors:

3.1.1.1 Micro limit switches: used 14 under the 2 octaves (C3 to B4) to detect what the user press on the keyboard

Figure 3.16: Micro Limit Switches

3.1.1.2 Limit Switches: used two for both ends of linear rod to define the travel boundaries and prevent mechanical overrun

Figure 3.17: Limit Switch

3.1.2 Other Components

3.1.2.1 Shift Registers (8-bit) 74HC165: Due to I/O limitations, the LEDs are controlled through two serial-in parallel-out shift registers (each with 8 outputs from QA to QH), allowing efficient management of all 14 LEDs using minimal pins.

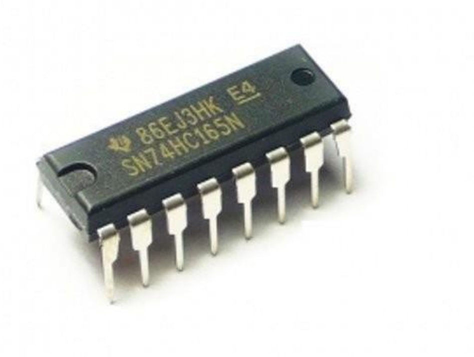

Figure 3.18: Shift Registers (8-bit) 74HC165

3.6.3.1 LCD Display with I2C Module:

An LCD module is connected to the Arduino to display the next note in the melody for Practice mode and the current note on the test mode.

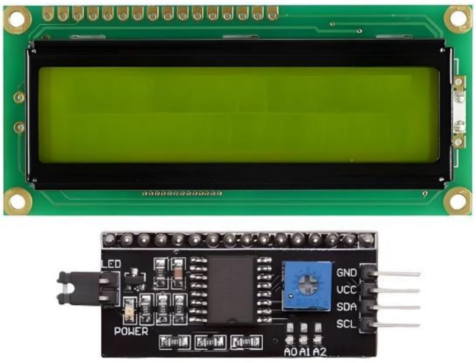

Figure 3.19: LCD Display with I2C Module

3.6.3.2 Stripe of white LEDs:

divide it to control each led alone positioned above 14 physical piano keys (white keys from C3 to B4) connected to shift registers due to I/O limitations These LEDs serve as visual cues to indicate which key the user should press next

Figure 3.20: White LEDs Stripe.

3.6.3.3 8mm Stainless Steel Rod: These rods serve as the **linear guide rails** for the robotic hands. They provide a stable and low-friction path for the sliding carriage, allowing the hands to move horizontally across the keyboard. Paired with linear motion bearings, they ensure smooth and precise traversal during performance and practice.

Figure 3.21: 8mm stainless steel rod.

3.6.3.4 Linear Motion Bearings: Mounted on the sliding carriage of each robotic hand, these bearings enable **smooth movement** along the stainless steel rods. They reduce mechanical friction and prevent vibration or wobble during hand repositioning, ensuring accurate alignment over the piano keys.

Figure 3.22: linear motion bearings.

3.6.3.5 Timing Belt: The timing belt connects the stepper motor to the moving platform that carries the robotic hand. It allows for **precise, controlled movement** across the keyboard by translating rotational motion from the stepper into linear motion. This component is essential for syncing note positions with hand location.

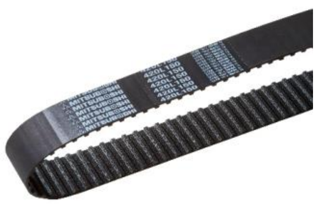

Figure 3.23: Timing Belt.

3.6.3.6 ESP8266: Used as a **Wi-Fi interface module**, the ESP8266 allows wireless communication between the robot and the web/mobile application. It handles mode selection, melody input, and result display from the browser, enabling seamless interaction without onboard buttons.

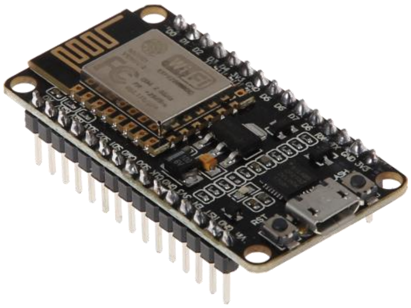

Figure 3.24: ESP8266

3.6.3.7 ESP32: The ESP32 is used to control the **second robotic hand**, operating in parallel with the Arduino Mega that controls the first hand. This dual-controller setup enables **synchronized, two-hand performance**, where each microcontroller handles its own servos, minimizing execution delays and expanding the playable range.

Figure 3.25: ESP32

3.6.3.8 DF MINI PLAYER: This module provides **audio output**, playing pre-recorded MP3 instructions, feedback sounds, and system alerts. It improves user engagement and accessibility by offering voice-guided interaction and feedback during all modes (Practice, Game, and Performance).

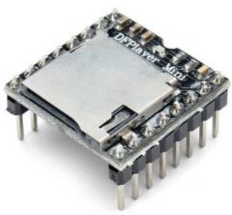

Figure 3.26: DF Mini Player.

3.7 Constraints

3.7.1 Timing and Synchronization Challenges

• **Note Accuracy in Performance Mode**: Keeping consistent timing between servo presses and stepper travel required fine-tuned delays and testing to ensure musical accuracy.

3.7.2 Dependency on Audio-to-Note Extraction System

- **Fast-note dropout**: grace notes under ~30 ms often vanish; lower the duration filter or run a second pass with a short-window spectrogram if you need every trill.
- **Polyphonic bleed:** multiple instruments confuse pitch trackers; isolate target line first (e.g., vocal stem) or switch to a multi-fundamental model like CREPE-full.
- **External correction service**: optionally feed raw MIDI to Melodyne-API-like tool or use Magenta's "NoteSequence cleaning" to auto-fix rests, merges, and sustain lengths.

3.7.3 Inability to Play Black Keys

• **Mechanical Design Limitation**: The robot hand and finger system was designed to align precisely with the white piano keys. Due to the vertical height and recessed position of black keys, the current finger structure and actuation mechanism cannot physically reach or press them without risk of collision or misalignment.

3.7.4 Fixed Gap Between Hands

- **Limited Note Range Coverage**: The fixed horizontal gap between Robhoven's two hands restricts the ability to play widely spaced left- and right-hand parts simultaneously.
- **Collision Risk**: When the hands need to play adjacent or overlapping notes, the rigid spacing can lead to physical interference or require timing adjustments to avoid clashes.
- **Reduced Flexibility**: The system lacks dynamic hand repositioning, limiting its ability to adapt to pieces with shifting hand positions or large two-hand chords.
- **Artificial Timing Offsets:** To accommodate the fixed spacing, the software sometimes staggers note playback unnaturally, which affects musical expression and timing fidelity.

3.8 Mobile Application:

The Robhoven interface is a responsive web/mobile application that enables users to interact with the robotic piano system in real time. It serves as the primary control hub, offering access to three distinct operational modes:

- **Performance Mode**
  This mode allows Robhoven to autonomously perform pre-programmed melodies. Users simply select a song, and the robotic hand executes it with precise servo control and timing synchronization. It's ideal for demos and showcasing mechanical accuracy.
- **Practice Mode**
  Designed as an educational tool, this mode guides users through melodies by lighting up corresponding keys and waiting for the correct key press. Feedback is given instantly, helping learners build timing and muscle memory. The robot only proceeds when the correct key is pressed, reinforcing accuracy.
- **Game Mode (Guess the Melody)**
  In this gamified mode, Robhoven plays a random melody segment. The user must identify the melody from multiple choices. The interface gives real-time feedback on whether the guess was correct or not and can communicate pass/fail signals to the robot to trigger celebratory or corrective responses (e.g.,

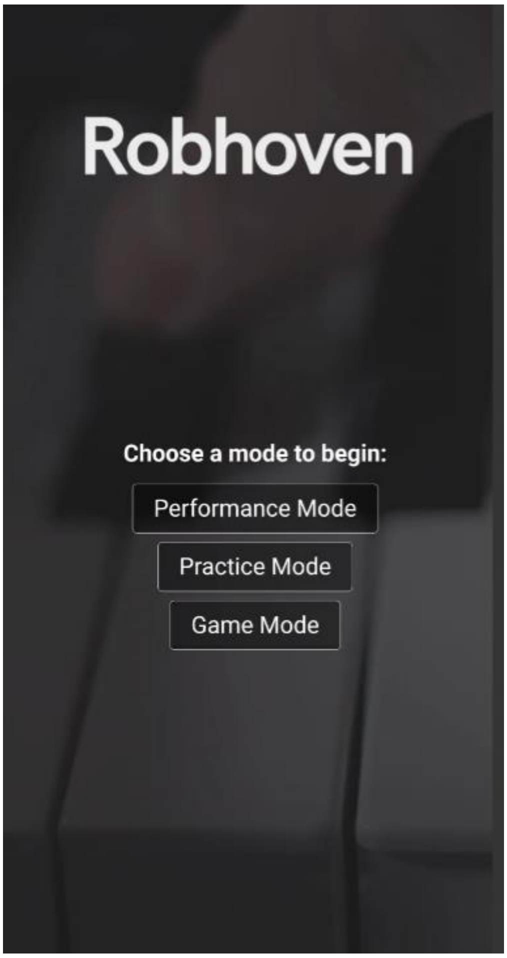

Figure3.27 Mobile Application

The app uses clean UI elements, with clearly labeled buttons and modes, and communicates with the ESP8266 module over HTTP to send commands and receive robot state updates. It’s designed to run smoothly on both desktop and mobile browsers, requiring no installation.

Chapter 4

Results and Discussion

4.1 Data Collected

RobHoven was tested under different conditions to evaluate its mechanical performance, note-playing accuracy, feedback responsiveness, and usability in various modes (practice, performance, and test). Results were analyzed to highlight key strengths and limitations.

4.1.1 Note Accuracy and Timing in Performance Mode

The robot was tested with several pre-programmed melodies. The performance was evaluated based on note precision and rhythm consistency.

• RobHoven was able to strike the correct white-key notes with acceptable timing accuracy.

• Performance consistency was high for slow-to-medium tempo songs. Fast melodies showed minor delays due to servo speed and stepper transition time.

4.1.2 Practice Mode Responsiveness

Practice mode was tested to assess real-time guidance, feedback response, and system synchronization.

• LEDs correctly highlighted the note to be pressed, and the LCD displayed the corresponding note using custom glyphs.

• When the correct key was pressed (detected via **mini switches** under the piano keys), the system advanced smoothly to the next note.

• Contact issues with mini switches occasionally caused missed or false activations, affecting flow.

4.1.3 Test Mode Evaluation

Test mode was assessed by tracking correct vs. incorrect inputs and verifying feedback accuracy.

- The system correctly evaluated user responses using data from **mini switches** mounted under the piano.
- The LCD showed real-time note names corresponding to user input.
- Wrong key presses triggered visual using the App.
- Physical alignment and press sensitivity of the mini switches influenced accuracy, especially if keys were not fully depressed.

4.1.4 Mechanical Movement and Finger Actuation

The robot's hand positioning and key pressing mechanism were analyzed for speed, reliability, and stress.
• The stepper motors reliably moved the hands across the 36 white keys.
• Servo fingers consistently returned to the correct position after striking a key.

4.1.5 Error Estimation, Challenges, and Solutions

The objective of this project was to develop a stable and precise robotic system capable of playing piano melodies accurately through mechanical actuation and intelligent control. Robhoven successfully achieved this goal, demonstrating the ability to press keys, interpret notes, and handle multiple modes of operation. However, throughout the development and testing phases, several technical challenges emerged that required iterative debugging, fine-tuning, and hardware adjustments to ensure reliable performance.

1. **No Support for Black Keys**

   • Due to the mechanical layout, fingers cannot reach recessed black keys without redesign.
   • **Solution**: System limited to white-key melodies only. A future redesign may include vertical articulation or angled fingers.

2. **Audio-to-Note Extraction Accuracy**

   • The note extraction from uploaded audio using Audio-to-Note Extraction System was sometimes inaccurate, especially for fast or polyphonic music.

   • **Solution:** Manual correction of extracted notes was used as a temporary fix. A local MIDI mapping tool is being considered.

3. **Interference Wiring Congestion and Maintenance Difficulty**

   • The robot’s compact layout resulted in a dense wiring system prone to disconnections.
   • **Solution**: Improved cable routing and use of labeled connectors to reduce debugging time.

4. **Mini Switch Press Debouncing**
   • False positives or missed detections occurred due to mechanical bounce or partial presses of the **mini switches** under keys.
   • **Solution**: Software debounce was implemented using timing thresholds to ignore rapid changes in state. The system waits for stable input before registering a valid press.

5. **Performance Unreachable Middle Keys Due to Hand Gap**
   • During testing, it was found that the physical gap between the two robotic hands created a dead zone a range of white keys that neither hand could reach. This limited RobHoven's ability to play certain melodies that require consecutive notes spanning across both hands.
   • **Solution**: Due to mechanical constraints and wiring limitations, no direct fix was applied in the current version. Instead, melodies were selected or transposed to avoid placing notes in the unreachable region. A future enhancement could involve overlapping hand ranges or a third finger mechanism to bridge the gap.

Key Contributions

1. **Multimodal Piano Learning Assistance**
   RobHoven integrates visual (LEDs, LCD), auditory (DFPlayer Mini with instructional MP3s), and physical (servo-actuated key presses) guidance into a single system. This multimodal approach enhances learning for users with different learning styles, improving engagement and retention.
2. **Three Distinct Interactive Modes**
   The system offers Practice Mode (note-by-note guidance with feedback), Performance Mode (autonomous melody playback), Test Mode (scored user input), and Guess The Melody Mode. This makes RobHoven more than a player it's an interactive tutor and evaluator.
3. **Real-Time Feedback System Using Mini Switches**
   Each 14 white piano key includes an embedded mini switch that detects user interaction in real time. This allows accurate tracking of user responses and supports precise progression logic in practice and test modes.
4. **Speech Feedback Integration with MP3 Audio Playback**
   Using the DFPlayer Mini, RobHoven plays voice instructions and responses from a preloaded SD card, giving it a human-like instructional quality. It can encourage, warn, or prompt users based on performance, which enhances user motivation.
5. **Custom LCD Display with Note Glyphs**
   The 16x2 I2C LCD shows big stylized note names (e.g., DO, RI, MI) using custom glyphs to improve visual clarity. The centered display of note names helps users focus on the current task without confusion or small-font reading errors.
6. **Precise Servo-Based Key Actuation**
   RobHoven uses MG90S servo motors for finger actuation, allowing controlled and repeatable key

presses. These servos were calibrated for speed and force to simulate human-like play, especially during performance mode.

7. **Stepper-Controlled Horizontal Hand Movement**
   A NEMA17 stepper motor moves each hand horizontally along the piano using linear rails. This setup ensures precise alignment with the white keys and makes it possible to dynamically shift hand position based on note location.
8. **Web-Controlled User Interface with ESP8266**
   the robot integrates an ESP8266 module that provides a wireless interface for users to select modes, input melodies, and monitor system status. This eliminates the need for onboard buttons and simplifies user interaction.
9. **Support for External Note Extraction**
   The system can convert real-world melodies into note arrays using external audio-to-Note extraction System. This allows users to input custom songs beyond the hardcoded melodies, significantly expanding the robot's use case.
10. **Educational Focus with Scoring and Correction Feedback**
    In test mode, RobHoven evaluates user performance and provides feedback, including score calculation and correction tracking. This gamified interaction turns passive practice into an engaging and measurable learning experience.

Chapter 5

Conclusion and Future work

5.1 Conclusion

This project successfully developed RobHoven, an interactive piano-learning robotic assistant capable of guiding users through playing melodies using real-time visual, auditory, and physical feedback. Unlike static learning tools, RobHoven brings music education to life by physically pressing piano keys, reacting to user input, and adapting based on performance across three modes: Practice, Test, and Performance.

RobHoven integrates multiple hardware and software components, including servo-actuated robotic fingers, a linear stepper-driven movement system, an LCD screen with custom musical note display, a DFPlayer Mini for instructional audio playback, and an ESP8266 module for wireless control. Mini switches under each white key detect user interaction, enabling the robot to track learning progress and offer accurate feedback.

Throughout development and testing, RobHoven demonstrated its ability to:

- Perform full melodies with controlled timing and accurate note execution.
- Guide users note-by-note using LEDs, LCD, and voice cues.
- Evaluate and score user responses in real time.

Despite its success, the system faced several challenges — mechanical constraints prevented reaching black keys and a central dead zone between the two robotic hands limited coverage of some notes. External note extraction APIs occasionally produced errors, requiring manual correction. Still, the project met its core objectives and delivered a working prototype suitable for educational use.

This project provided a valuable hands-on learning experience in embedded systems, robotics, hardware-software integration, and real-world debugging. It required applying concepts from control systems, signal processing, mechanical design, and user-centered development to solve practical problems in a creative, scalable way.

5.2 Recommendations for Improvement

While RobHoven achieved its primary goals, the following improvements are recommended to enhance its capability, flexibility, and user experience:

1. **Expanded Key Coverage**

• **Bridging the Hand Gap**: The current design leaves a range of keys in the middle unreachable by either hand. Future designs should include overlapping hand movement ranges or an additional finger module to eliminate this dead zone.

• **Support for Black Keys**: A new mechanical structure is needed to reach elevated and recessed black keys, possibly through articulated or angled fingers with vertical reach control.

2. More Accurate and Flexible Note Extraction

• **Reliable AI-Based Transcription**: The current system depends on external audio-to-Note Syste, which can be inaccurate or inconsistent. A local model trained on piano-specific datasets or a MIDI file import tool would improve reliability.

• **Dynamic Rhythm Handling**: Enhancing timing flexibility in performance mode will allow for more expressive and realistic playback of extracted songs.

3. **Mechanical Optimization and Modularity**

• **Improved Finger Design**: Redesigning the finger brackets using stronger, lightweight materials would reduce jitter and mechanical stress over time.

• **Modular Components**: Designing plug-and-play style finger or switch modules would simplify maintenance, part replacement, and upgrades.

4. **Enhanced Audio Synchronization**

• **BUSY Pin Integration**: Monitoring the DFPlayer Mini's BUSY pin will ensure that audio playback is fully synchronized with visual displays and LED cues.

• **Expanded Voice Instruction Set**: Including more MP3 prompts (e.g., encouragements, tips) would enhance interaction and motivation.

5. **Smarter Feedback and User Interaction**

• **Adaptive Difficulty**: Adding logic that increases or decreases difficulty based on user performance would personalize the learning experience.

References

[1] Doi, M., & Nakajima, Y. (2007). *Toyota Partner Robots*. Wired. https://www.wired.com/2007/12/toyota-shows-vi/

[2] Prosseda, R. (2012). *Teotronico: The Pianist Robot*. Teotronica. http://www.teotronico.com

[3] Yousician & Simply Piano. Mobile app-based music learning platforms. https://www.yousician.com, https://www.joytunes.com/simply-pianoWired. (Dec 10, 2007). *Toyota Shows Violinist Robot*—overview of the multi-jointed Partner Robot playing classical music. emerald.com+6wired.com+6en.wikipedia.org+6