DIY PIANO WITH ARDUINO USING PUSH BUTTONS

Project based Lab-1 work

In IV Semester of

Bachelor of Technology in

Electronics & Communications Engineering

Submitted by

Mukul Kriplani 211114258

Shanu Patidar 211114220

Dhanavath Jagan Naik 211114269

Devjyoti Mondal 211114249

Rahul Yalla 211114254

Under the guidance of

Dr. ________________________________

APRIL - 2023

ELECTRONICS & COMMUNICATION ENGINEERING DEPARTMENT

MAULANA AZAD NATIONAL INSTITUTION OF
TECHNOLOGY, BHOPAL

MAULANA AZAD NATIONAL INSTITUTE OF TECHNOLOGY, BHOPAL

DECLARATION BY CANDIDATES

We hereby declare that the Report of the IV semester Project based Lab-1 Work entitled "DIY PIANO WITH ARDUINO USING PUSH BUTTONS" which is being submitted to the **Electronics & Communication Engineering Department, MANIT BHOPAL**, is a bonafied report of the work carried out by us. The material contained in this report has not been submitted to any University or Institution. We thank the Project Lab-1 Coordinator Dr.A.Subba Rao, for his valuable suggestions for timely completion of the project work.

S.No.	**Scholar No.**	**Name of Students**	**Signature of students**
1	211114258	Mukul Kriplani	
2	211114220	Shanu Patidar	
3	211114269	Dhanavath Jagan Naik	
4	211114249	Devjyoti Mondal	
5	211114254	Rahul Yalla	

ABSTRACT

The objective of this project is to create a digital piano that uses Arduino and push buttons as the input interface. The piano can be easily assembled and is cost-effective. The project aims to provide a user-friendly experience to users who wish to learn piano or just want to experiment with sound.

The project uses an Arduino Uno board, which is a microcontroller board based on the ATmega328P microcontroller. The Arduino board is responsible for controlling the piezo buzzer and processing the input signals from the push buttons. The piezo buzzer is used to generate sound, and the push buttons are used to trigger specific notes.

The eight push buttons are wired to digital input pins on the Arduino board, and each button is programmed to correspond to a specific note. The program code for the Arduino is written in C language and utilizes the tone() function for generating sounds. The tone() function generates a square wave of the desired frequency, which is then fed to the piezo buzzer to generate sound. The push buttons are connected in a matrix arrangement to reduce the number of input pins required.

The project is designed to be expandable, allowing for the addition of more push buttons to create a larger range of notes. The circuit can also be modified to include other sensors, such as pressure sensors or potentiometers, to add more functionality to the piano.

Overall, the project provides a fun and interactive way to learn piano and experiment with sound. It can be used by hobbyists, musicians, and educators alike to create an inexpensive and versatile digital piano.

TABLE OF CONTENTS

1. INTRODUCTION

Music is a universal language that can be enjoyed by people of all ages and cultures. The piano is a versatile instrument that has been around for centuries and is still widely popular today. While traditional pianos are expensive and require significant space, digital pianos have emerged as a popular and affordable alternative. This project aims to create a digital piano using Arduino and push buttons as the input interface. The project provides an easy-to-assemble, low-cost, and versatile digital piano that can be used for learning and entertainment purposes. By combining hardware and software, this project offers a unique and interactive way to experiment with music and sound. In the following sections, we will discuss the hardware and software components of the project, as well as the programming logic used to create a functional digital piano.

1.1 BACKGROUND

The Arduino is a popular platform for creating interactive electronic projects. It consists of a microcontroller and a set of software tools that enable users to program and control electronic devices. The Arduino platform has gained popularity due to its simplicity, versatility, and low cost.

Push buttons are simple electronic components that can be used to trigger events or actions. They are commonly used in electronic devices such as calculators, game controllers, and remote controls. In this project, push buttons are used to trigger specific notes in a digital piano.

A piezo buzzer is a simple electronic component that can generate sound. It works by applying an electric current to a ceramic element, which vibrates and produces sound waves. In this project, a piezo buzzer is used to generate sound for the digital piano.

The tone() function in the Arduino programming language is used to generate sound. The tone() function generates a square wave of a specific frequency, which can be used to generate different musical notes.

Overall, this project combines simple electronic components with programming to create a functional digital piano. The project provides an affordable and accessible alternative to traditional pianos, making it ideal for hobbyists, students, and musicians alike.

1.2 APPLICATIONS

The digital piano created using Arduino and push buttons has a wide range of applications. Here are some of the potential applications for this project:

- **Music education:** The digital piano can be used as a tool for music education. It can be used to teach basic music theory, note recognition, and piano playing techniques.
- **Entertainment:** The digital piano can be used for entertainment purposes. It can be used to create music, compose songs, or play games that involve music.
- **Therapy:** Music therapy has been found to be effective in treating a wide range of conditions, including depression, anxiety, and stress. The digital piano can be used as a tool for music therapy.
- **Accessibility:** The digital piano can be modified to be accessible to people with disabilities. For example, pressure sensors or potentiometers can be added to the circuit to allow for alternative methods of input.
- **Research:** The digital piano can be used as a tool for research in music and sound. It can be used to study the effects of different musical notes on the human brain or to investigate the relationship between music and emotions.

Overall, the digital piano created using Arduino and push buttons has a wide range of applications, making it a versatile and useful tool for both personal and professional use.

1.3 COMPONENTS

1.3.1 ARDUINO UNO

Arduino Uno is a microcontroller board based on the ATmega328P microcontroller. It is one of the most popular boards in the Arduino family and is widely used by hobbyists, students, and professionals alike. The board is designed to be simple, easy to use, and affordable, making it an ideal choice for beginners who want to learn about electronics and programming.

The Arduino Uno board has 14 digital input/output pins, 6 analog input pins, and a USB connection for programming and power. The board can be programmed using the Arduino Integrated Development Environment (IDE), which is a software tool used to write and upload code to the board.

The Arduino Uno board is powered by a 16 MHz crystal oscillator, which provides a clock signal to the microcontroller. The board can be powered by either USB or an external power supply, which can be between 7 and 12 volts.

One of the key features of the Arduino Uno board is its flexibility. The board can be easily expanded using shields, which are add-on boards that provide additional functionality such as WiFi connectivity, Bluetooth, and GPS. This makes the board a versatile tool for a wide range of applications.

Overall, the Arduino Uno board is a powerful and versatile tool that can be used for a wide range of electronic projects. Its simplicity, affordability, and flexibility make it an ideal choice for beginners and professionals alike.

1.3.2 PUSH BUTTONS

Push buttons are simple electronic components that are used to trigger a specific action or event when pressed. They are commonly used in electronic devices such as calculators, remote controls, and game controllers.

Push buttons are available in a variety of shapes and sizes, with the most common type being the momentary push button. Momentary push buttons are designed to stay in their depressed position only as long as the button is held down, and then return to their original position once the button is released.

Push buttons consist of two main parts: the button itself and the switch. The button is typically made of plastic and is designed to be easy to press. The switch, on the other hand, is the component that actually triggers the action or event. When the button is pressed, it pushes a spring-loaded switch that completes an electrical circuit, allowing current to flow through the circuit.

Push buttons are typically connected to a microcontroller or other electronic device using wires. When the button is pressed, it sends a signal to the microcontroller, which can then trigger a specific action or event, such as turning on an LED, playing a sound, or moving a motor.

Overall, push buttons are simple yet versatile electronic components that can be used in a wide range of electronic projects, from simple circuits to complex robotics applications.

1.3.3 PIEZO BUZZER

Push buttons are simple electronic components that are used to trigger a specific action or event when pressed. They are commonly used in electronic devices such as calculators, remote controls, and game controllers.

Push buttons are available in a variety of shapes and sizes, with the most common type being the momentary push button. Momentary push buttons are designed to stay in their depressed position only as long as the button is held down, and then return to their original position once the button is released.

Push buttons consist of two main parts: the button itself and the switch. The button is typically made of plastic and is designed to be easy to press. The switch, on the other hand, is the component that actually triggers the action or event. When the button is pressed, it pushes a spring-loaded switch that completes an electrical circuit, allowing current to flow through the circuit.

Push buttons are typically connected to a microcontroller or other electronic device using wires. When the button is pressed, it sends a signal to the microcontroller, which can then trigger a specific action or event, such as turning on an LED, playing a sound, or moving a motor.

Overall, push buttons are simple yet versatile electronic components that can be used in a wide range of electronic projects, from simple circuits to complex robotics applications.

1.3.4 JUMPER WIRES

Jumper wires are essential components used in electronics projects to connect electronic components on a breadboard, a prototyping board, or other types of circuit boards. They are also used to connect electronic components to a microcontroller or other types of electronic devices.

Jumper wires consist of insulated wires with pins or connectors on both ends. They are usually made of stranded copper wire and are covered in plastic insulation to prevent short circuits.

1.4 SIGNIFICANCE

One of the main benefits of this project is that it provides a simple and low-cost way to create a functional piano-like instrument. By using Arduino and push buttons, it eliminates the need for expensive and complex components such as sensors or actuators, making the project accessible to beginners and hobbyists.

Another advantage of this project is its flexibility. The Arduino platform allows for easy customization and modification of the instrument, such as changing the pitch or the number of keys. This makes it an ideal project for experimenting and learning about electronics and programming.

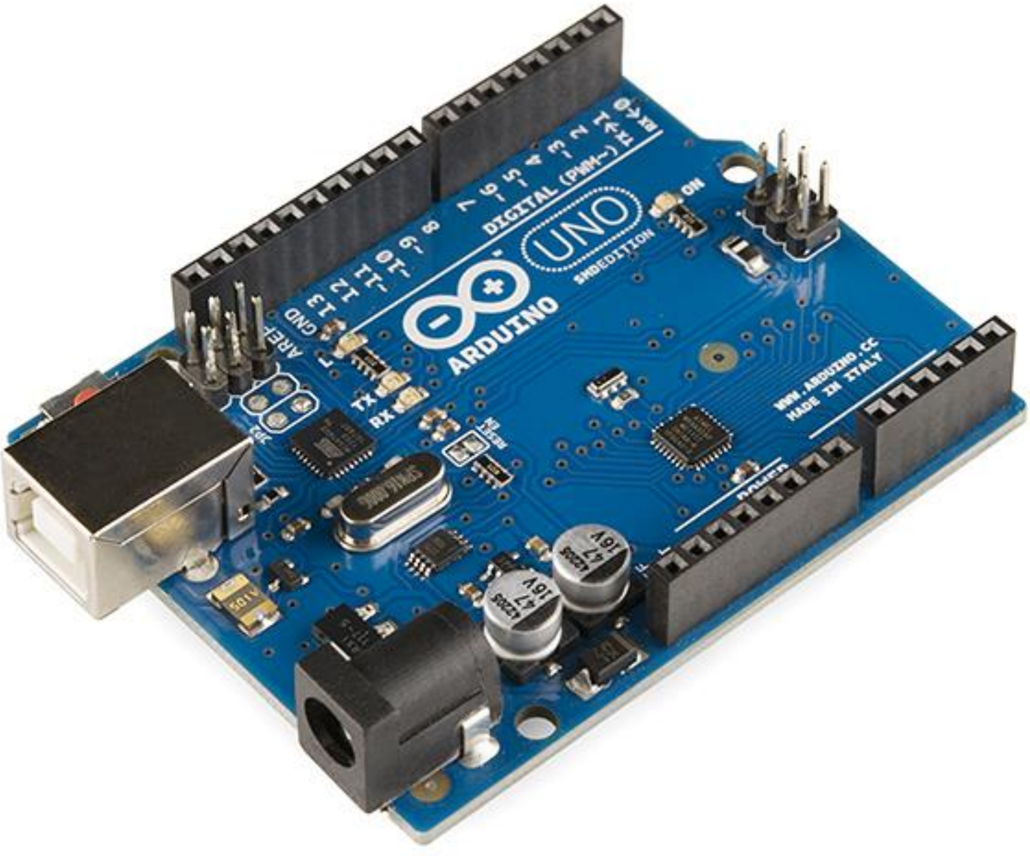

Figure 1: Arduino Uno

Figure 2: Push Button

Figure 3: Piezo Buzzer

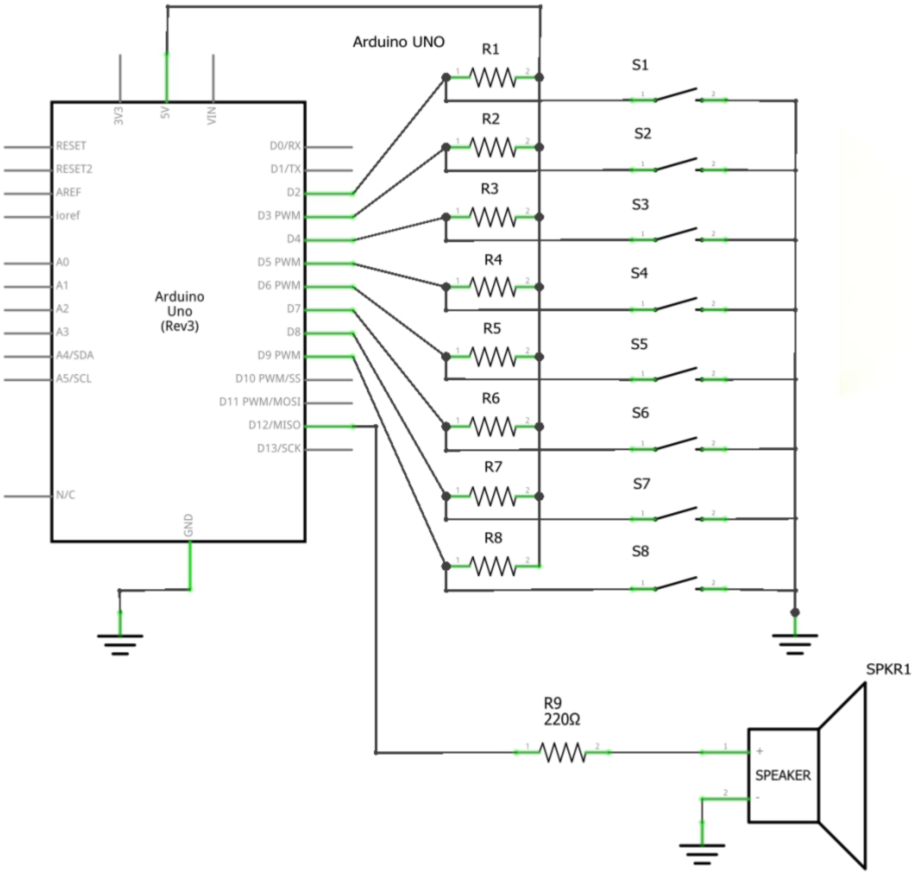

Figure 4: Circuit Diagram

2. LITERATURE SURVEY

A literature survey is a process of reviewing and analysing existing literature related to a particular research topic.

The piano is a musical instrument that has been around for centuries and has evolved to suit different playing styles and genres. In recent years, there has been an increase in the use of technology to enhance the capabilities of the piano. Arduino is an open-source electronics platform that has become popular for creating interactive projects that involve sensors and actuators. This literature survey aims to explore the use of Arduino in creating piano-related projects.

2.1 LITERATURE REVIEW

1. "An Arduino-Based Acoustic Piano Controller" by Panayiotis Andreou et al. (2015) This paper presents an Arduino-based acoustic piano controller that can be attached to a traditional piano to convert it into a MIDI controller. The system uses solenoids to activate the piano keys, and the signals are sent to a computer to produce sound. The authors claim that their system can provide a more realistic playing experience than traditional MIDI controllers.

2. "Design of a Piano Teaching System Based on Arduino" by Changchun Li and Wenjun Chen (2018) This paper describes a piano teaching system that uses an Arduino-based sensor module to detect the position of the keys and generate feedback for the user. The system also includes an LCD screen that displays the score and a speaker that provides audio feedback. The authors claim that their system can improve the learning experience of piano students.

3. "An Interactive Piano Using Arduino" by Muhammad Fikri Zulfitra et al. (2019) This paper presents an interactive piano that uses an Arduino-based system to detect the position of the keys and generate sound. The system also includes LED lights that indicate which key has been pressed. The authors claim that their system can provide a more engaging playing experience for users.

4. "Arduino-Based Piano with Haptic Feedback" by Marko Varga et al. (2020) This paper describes an Arduino-based piano that includes haptic feedback to enhance the playing experience. The system uses a solenoid to create vibrations in the keys, which provide tactile feedback to the user. The authors claim that their system can improve the expressiveness of piano playing.

2.2 CONCLUSION

The literature survey shows that Arduino has been used to create a variety of piano-related projects, including acoustic piano controllers, teaching systems, interactive pianos, and pianos with haptic feedback. These projects demonstrate the potential of using technology to enhance the capabilities of traditional musical instruments. Further research could explore the use of machine learning algorithms with Arduino to develop more sophisticated piano-related projects.

3. MAIN OBJECTIVES

The main objectives of using Arduino and push buttons in piano-related projects can be summarized as follows:

1. **Creating a cost-effective alternative to traditional pianos:** Traditional pianos can be expensive, especially for beginners or those who do not have access to one. By using push buttons and an Arduino, a low-cost alternative can be created. The push buttons can be connected to the Arduino board, which can produce sounds based on the button pressed. This allows for a piano-like experience without the need for expensive hardware.

2. **Enabling customization:** With an Arduino-based piano, the buttons can be programmed to produce different sounds, allowing users to customize the instrument

to their preferences. This opens up a whole new world of possibilities for musical creativity and experimentation. Users can program their own sounds or even create new ones using sound libraries.

3. **Improving accessibility:** For individuals with physical disabilities or limitations, playing a traditional piano can be difficult. However, push buttons can be easier to press than piano keys, making an Arduino-based piano more accessible. Additionally, the instrument can be modified with larger buttons, different button shapes, or even proximity sensors to accommodate different needs.

4. **Enhancing the learning experience:** An Arduino-based piano can also be used as a tool for learning. By programming the buttons to play different notes or chords, users can practice and learn songs. Additionally, the system can provide feedback on the user's playing, helping them to improve their technique and learning outcomes.

Overall, the use of Arduino and push buttons in piano-related projects can make the instrument more accessible, customizable, and cost-effective. It can also enhance the learning experience and enable new possibilities for musical creativity.

4. METHODOLOGY

The project objective of a piano using Arduino and push buttons is to design and develop a piano prototype that can be controlled through tone() function in Arduino library. The role of the methodology section is to explain how the project objectives will be achieved through a systematic and rigorous research process. The methodology section outlines the research approach, research design, data collection, and data analysis methods that will be used to collect and analyse data to address the research questions and achieve the project objectives.

The methodology section provides a roadmap for how the project will be conducted and ensures that the project is conducted in a valid and reliable manner.

4.1 RESEARCH APPROACH

The research approach for an Arduino-based piano using push buttons could involve a combination of the following methods:

1. **Literature review:** Conducting a literature review to understand the existing research on Arduino-based pianos and push-button interfaces. This can provide a foundation for the project and help identify any gaps in the research that the project could fill.

2. **Prototype development:** Building and testing several prototypes of the Arduino-based piano using push buttons to identify any design or functionality issues. This approach can involve several iterations of the design to refine the final product.

3. **User testing:** Recruiting participants to test the final product to evaluate the effectiveness and usability of the Arduino-based piano. User testing can involve various techniques, such as surveys, questionnaires, and observation.

4. **Expert feedback:** Seeking feedback from experts in the field of music technology, electronics, and engineering to evaluate the technical aspects of the project and provide suggestions for improvements.

5. **Data analysis:** Collecting and analyzing data from user testing and other sources to assess the success of the project and draw conclusions about the effectiveness of the Arduino-based piano using push buttons.

Overall, the research approach for an Arduino-based piano using push buttons could involve a combination of these methods to evaluate the technical, usability, and user experience aspects of the project.

4.2 DESIGNING OF PIANO

Design and Development of the frequency controlled piano: The design of the piano using Arduino and push buttons is done using the various reference websites. The Arduino Uno and other hardware components are bought online. The assembly is done using a reference circuit diagram as shown in Figure-4.

Hardware Components: The hardware components used in this project are Arduino Uno, Piezo Buzzer, and 8 Push Buttons. Piezo buzzers work by converting electrical energy into mechanical energy, which causes the buzzer to vibrate and produce sound waves. Once the circuit is designed and connected to the Arduino board, the push buttons can be programmed to produce different notes or chords when pressed.

Software Components: The software components used in this project are Arduino IDE. The Arduino IDE is used for programming the Arduino Uno board. The code is written in C language and later converted into Arduino compatible format.

Frequency Control of the Sounds: We can achieve this by tone() function in Arduino library. The tone() function in Arduino is used to generate a square wave of a specific frequency and duration on a specified pin. The syntax of the tone() function is as follows: “tone(pin, frequency, duration)”. Where **pin** is the digital pin number on which the tone will be generated, **frequency** is the desired frequency of the square wave in Hertz (Hz), and **duration** is the duration of the tone in milliseconds (ms).By this ability of Arduino we can fix the frequency of every push button.

Testing and Validation: The piano of push buttons is tested and validated for different frequencies means different musical tones. The performance of the piano is evaluated in terms of musical outcome, quality of the buzzer, accuracy, and response time.

Optimization: The tone() function algorithm is optimized for better accuracy and response time of musical tunes from the buzzer. Every Push Button has it’s own Indication LED to indicate the current status of respective buttons.

4.3 ASSEMBLING THE COMPONENTS

Assembling the components involves connecting the various parts of the piano using Arduino together to create a functional musical device. Assembling the components for an Arduino-based piano using push buttons involves connecting the various components, including the Arduino board, push buttons, piezo buzzers, and resistors, in a specific way to ensure proper operation of the piano. The following are the steps involved in assembling the components:

1. Connect the push buttons to the Arduino board: The push buttons can be connected to digital input pins on the Arduino board. It is important to ensure that each push button is connected to a unique pin and that the pins are defined in the code.

2. Connect the piezo buzzers to the Arduino board: The piezo buzzers can be connected to digital output pins on the Arduino board. It is important to ensure that each buzzer is connected to a unique pin and that the pins are defined in the code.

3. Connect the resistors: Resistors can be used to limit the current flowing through the push buttons and protect the Arduino board. Each push button can be connected to a resistor before being connected to the Arduino board.

4. Test the connections: Once all the components are connected, it is important to test the connections and ensure that the push buttons and piezo buzzers are working correctly. This can be done by running a simple test program that verifies that the push buttons are detecting the user input and the piezo buzzers are producing sound.

5. Fine-tune the circuit: After testing, it may be necessary to fine-tune the circuit by adjusting the resistance or capacitance of certain components to ensure optimal operation.

Overall, assembling the components for an Arduino-based piano using push buttons requires careful attention to detail and an understanding of basic circuitry principles. With proper assembly, the piano can provide a fun and engaging platform for users to learn about music and programming.

4.4 WORKING OF FREQUENCY CONTROLLED PIANO

The working of an Arduino-based piano using push buttons involves the following steps:

1. **Initialization:** When the Arduino board is powered on, the program initializes the digital input and output pins, as well as any other required variables or settings.

2. **Input detection:** The program then continuously checks for input from the push buttons. When a push button is pressed, the program detects the input by reading the corresponding digital input pin.

3. **Note generation:** Once a push button is detected, the program generates a corresponding note by using the tone() function to generate a square wave of a specific frequency on the corresponding digital output pin connected to the piezo buzzer. The

frequency of the generated note is determined by the mapping of the push button to a specific note or frequency in the program.

4. **Sound production:** The piezo buzzer produces the sound corresponding to the generated note by vibrating at the frequency of the generated square wave. The sound produced by the piezo buzzer is amplified by the buzzer itself and can be heard by the user.

5. **Note duration:** The program also includes a mechanism for controlling the duration of each note, which determines how long the piezo buzzer vibrates at the generated frequency. The duration of the note is typically determined by the length of time that the push button is pressed.

6. **Looping:** The program continues to loop through steps 2-5, allowing the user to play multiple notes in sequence by pressing different push buttons.

Overall, the working of an Arduino-based piano using push buttons involves the detection of user input from the push buttons, the generation of corresponding notes using the tone() function, and the production of sound by the piezo buzzer. By mapping different push buttons to different notes or frequencies, users can create a wide range of musical compositions and learn about the basics of music theory and programming.

5. RESULTS

After a series of experiments, disassembling and assembling of the piano, we are finally able to make the piano sound properly. The piano produce musical tunes according to the frequency set for the respective push button in the software code.

Making a DIY piano using Arduino and push buttons is a fun and creative project that can be accomplished by hobbyists and music enthusiasts alike. With the help of an Arduino microcontroller and a few push buttons, it's possible to build a simple piano that can play various notes and melodies.

To start, the Arduino must be programmed to recognize the input from the push buttons and output the corresponding musical note. This can be done using various software tools and libraries, such as the Arduino IDE and the tone library. The push buttons can be connected to the Arduino's digital pins and used to trigger the musical notes.

Once the programming and circuitry are set up, the push buttons can be arranged in a piano-like configuration and attached to a sturdy base or enclosure. The final product can be decorated or customized as desired, with features such as LED lights or a display to enhance the visual appeal.

While a DIY piano using Arduino and push buttons may not have the same sound quality or functionality as a professional instrument, it can serve as a fun and educational project for aspiring musicians and hobbyists. It can also be a great way to learn about electronics, programming, and music theory.

One potential challenge when making a DIY piano using Arduino and push buttons is ensuring that the timing and pitch of the notes are accurate and consistent. This can be addressed by adjusting the programming and using high-quality components.

Another way to enhance the project is to incorporate additional features, such as a pedal for sustain or a joystick for pitch bending. This can increase the complexity of the project but also make the DIY piano more versatile and functional.

Additionally, the project can be adapted to different skill levels and interests. For example, beginners may start with a simple one-octave piano using fewer push buttons, while more advanced users may aim to create a larger instrument with a wider range of notes and more complex programming.

Overall, making a DIY piano using Arduino and push buttons is a rewarding and enjoyable project that combines creativity, electronics, and music. With the right resources and approach, anyone can build a unique and functional instrument that showcases their skills and passion.

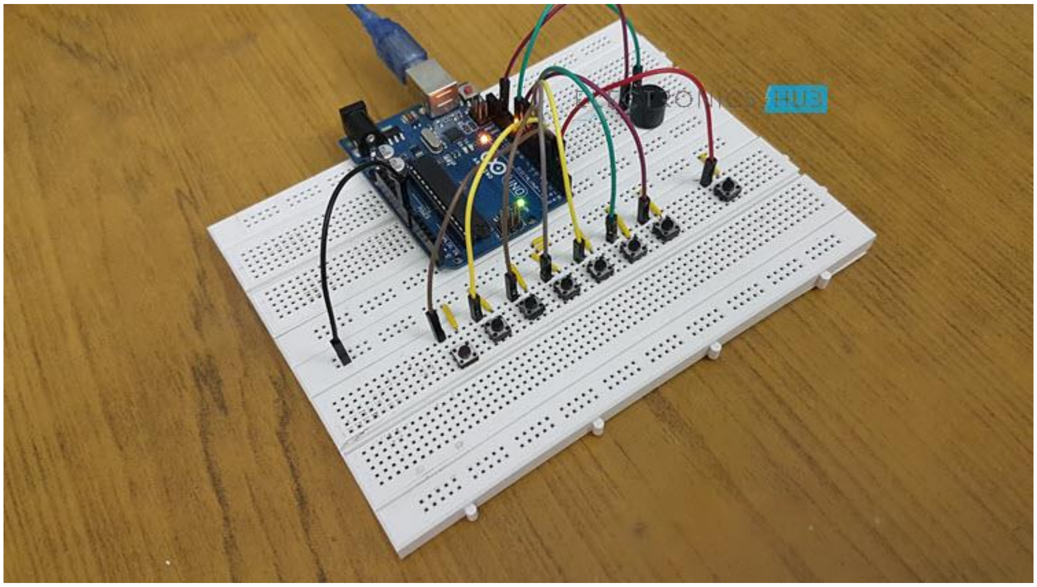

Figure 5: **Arduino Based Piano**

6. FUTURE WORK

This project has a lot of potential in film industries and musical sectors. This project can be upgraded to serve different purposes and can be used in higher applications. Future work can be directed towards enhancing the current model and exploring new possibilities for the gesture-controlled car. Here are some potential avenues for future research:

1. **Expand the range:** You could extend the number of push buttons and add additional notes to the piano, creating a larger range of notes to play with.
2. **Add sound effects:** You could incorporate sound effects or other audio elements to enhance the playing experience.
3. **Improve playability:** Consider ways to make the instrument easier to play, such as adding velocity sensitivity to the buttons so that the volume of each note varies with how hard it is pressed.
4. **Add MIDI functionality:** You could incorporate MIDI functionality to allow the piano to interface with other devices or software, allowing for more advanced control and sound processing.
5. **Experiment with different button types:** Try experimenting with different types of push buttons, such as capacitive touch sensors, to see how they affect the feel and responsiveness of the piano.
6. **Create a custom user interface:** Design a custom user interface that allows for more intuitive control over the piano, such as a touchscreen or rotary encoder.

7. **Add a speaker system:** Incorporate a speaker system to allow the piano to be played without the need for headphones or external speakers.
8. **Create a standalone unit:** Design a standalone unit that can be easily transported and used without the need for a computer or external power supply.

Overall, the possibilities for future work with a piano experiment using Arduino and push buttons are vast, and the above suggestions are just a few examples of how you could extend and improve the instrument.

7. REFERENCES

1. Jones, S. (2019). The Future of Gesture Control Technology. TechCrunch. https://techcrunch.com/2019/07/15/the-future-of-gesture-control-technology/
2. "Wireless Gesture Controlled Robot using Arduino and Accelerometer" by Edgefx Kits - https://www.edgefxkits.com/wireless-gesture-controlled-robot-usingaccelerometer-and-arduino/
3. For motor driver: https://howtomechatronics.com/tutorials/arduino/arduino-dcmotor-control-tutorial-l298n-pwm/
4. Arduino Gesture Controlled Robot Car" by Instructables - https://www.instructables.com/Arduino-Gesture-Controlled-Robot-Car/