

Piano Using Arduino

1. Introduction

This project demonstrates how to create a simple electronic piano using Arduino. By pressing different buttons, various musical notes are generated through a buzzer.

2. Key Components

- Arduino Uno Board
 - 8 Push Buttons
 - 1 Buzzer
 - Jumper Wires
 - Breadboard
 - Resistors (10k ohm recommended)
-

3. Working Principle

1. **Button Setup:** Eight push buttons are connected to the digital input pins of the Arduino.
 2. **Input Reading:** Each button is continuously monitored for a HIGH signal (pressed state).
 3. **Tone Generation:** When a button is pressed, the Arduino outputs a specific frequency tone through the buzzer corresponding to that button.
 4. **Sound Output:** The buzzer produces different musical notes based on which button is pressed.
 5. **Delay for Stability:** A short delay is added in the loop to ensure smooth operation and avoid multiple detections from a single press.
-

4. Circuit Overview

- **Push Buttons:** Connected between Arduino digital pins (2 to 9) and ground, with pull-up resistors if necessary.
 - **Buzzer:** Positive terminal connected to digital pin 13 and the negative terminal connected to GND.
 - **Arduino Pins:** Configured as INPUT for buttons and OUTPUT for the buzzer.
-

5. Code

//Piano Using Arduino

//Code written by-

//Fuad Hasan

//BME, KUET

int but1 = 2;

int but2 = 3;

int but3 = 4;

int but4 = 5;

int but5 = 6;

int but6 = 7;

int but7 = 8;

int but8 = 9;

int buzzer = 13;

void setup()

{

pinMode(but1, INPUT);

pinMode(but2, INPUT);

```
pinMode(but3, INPUT);  
pinMode(but4, INPUT);  
pinMode(but5, INPUT);  
pinMode(but6, INPUT);  
pinMode(but7, INPUT);  
pinMode(but8, INPUT);
```

```
pinMode(buzzer, OUTPUT);  
}
```

```
void loop()
```

```
{
```

```
int b1 = digitalRead(but1);  
int b2 = digitalRead(but2);  
int b3 = digitalRead(but3);  
int b4 = digitalRead(but4);  
int b5 = digitalRead(but5);  
int b6 = digitalRead(but6);  
int b7 = digitalRead(but7);  
int b8 = digitalRead(but8);
```

```
if(b1==1){  
    tone(buzzer, 300,100);  
}
```

```
if(b2==1){
    tone(buzzer, 400,100);
}
if(b3==1){
    tone(buzzer, 500,100);
}
if(b4==1){
    tone(buzzer, 600,100);
}
if(b5==1){
    tone(buzzer, 700,100);
}
if(b6==1){
    tone(buzzer, 800,100);
}
if(b7==1){
    tone(buzzer, 900,100);
}
if(b8==1){
    tone(buzzer, 1000,100);
}

delay(10);
}
```

6. Code Explanation

- **Variable Declaration:** Defines 8 button pins and 1 buzzer pin.
- **Setup Function:**
 - Sets button pins as INPUT to detect presses.
 - Sets the buzzer pin as OUTPUT to generate sound.
- **Loop Function:**
 - Reads the state of each button.
 - Checks if any button is pressed (state HIGH).
 - If pressed, generates a specific tone on the buzzer with a set frequency (in Hz) and a short duration (100 ms).
 - A short delay (10 ms) ensures smooth reading without bouncing effects.

7. Conclusion

This simple Arduino-based piano project shows how electronic components can be used creatively to simulate musical instruments. It is an excellent beginner project to learn about buttons, buzzers, and basic coding structures in Arduino.

