Experiment No: 04

Experiment Title: Playing Melody with Buzzer using Arduino

Course Code: CSE 460

Course Title: IoT Laboratory

Date of Submission: 20 December, 2023

Submitted To-

Dr. Md. Ezharul Islam
Professor
Department of CSE
Jahangirnagar University,
Savar, Dhaka, Bangladesh

Samsun Nahar Khandakar
Lecturer
Department of CSE
Jahangirnagar University,
Savar, Dhaka, Bangladesh

Submitted By-
Sumaita Binte Shorif
ID: 357

Group: 06
Group Members:

ID	Name
353	Khandoker Nosiba Arifin
356	Mangsura Kabir Oni
357	**Sumaita Binte Shorif**

Experiment 04: Playing Melody with Buzzer using Arduino

Objectives:

The main objectives of this experiment are:

- To gain a fundamental understanding of using an Arduino to generate musical tones, focusing on concepts such as frequency and period.
- To program an Arduino to play a simple melody using a buzzer.
- To observe the relationship between code and sound output in the context of Arduino programming.

Apparatus:

- Arduino Uno Board
- USB Cable
- Breadboard
- LED
- Jumper Wires
- Resistors
- Internet-connected device, etc.

Introduction:

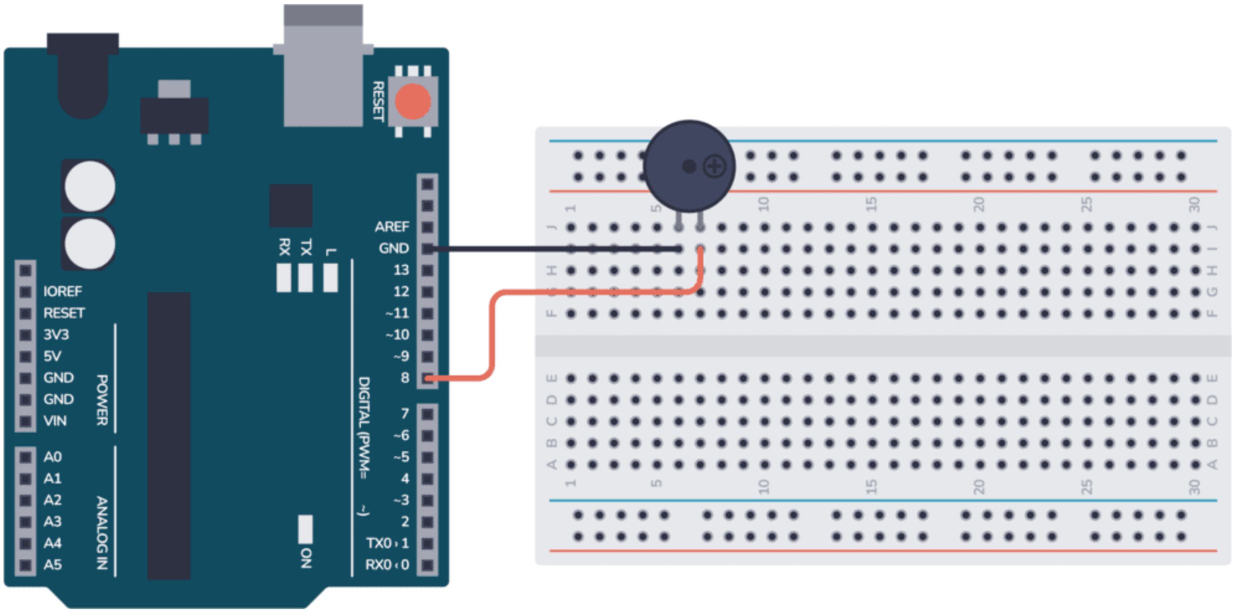
In this experiment, we delve into the diverse capabilities of the Arduino microcontroller, a versatile tool widely utilized in electronic projects. Specifically, we focus on its application in generating musical tones through a buzzer. Through the strategic control of the buzzer's frequency, we can produce a range of distinct notes, providing the foundation for creating intricate melodies. The experiment centers on the exploration of techniques for generating musical tones and playing melodies, leveraging the functionalities of an Arduino Uno board in conjunction with a buzzer.

Methodology:

Experimental Setup:

- Connect the positive pin of the buzzer to digital pin 8 on the Arduino.
- Connect the negative pin of the buzzer to the ground (GND) pin on the Arduino.
- Ensure the Arduino is connected to the computer via the USB cable.

Circuit Diagram:



Source code:

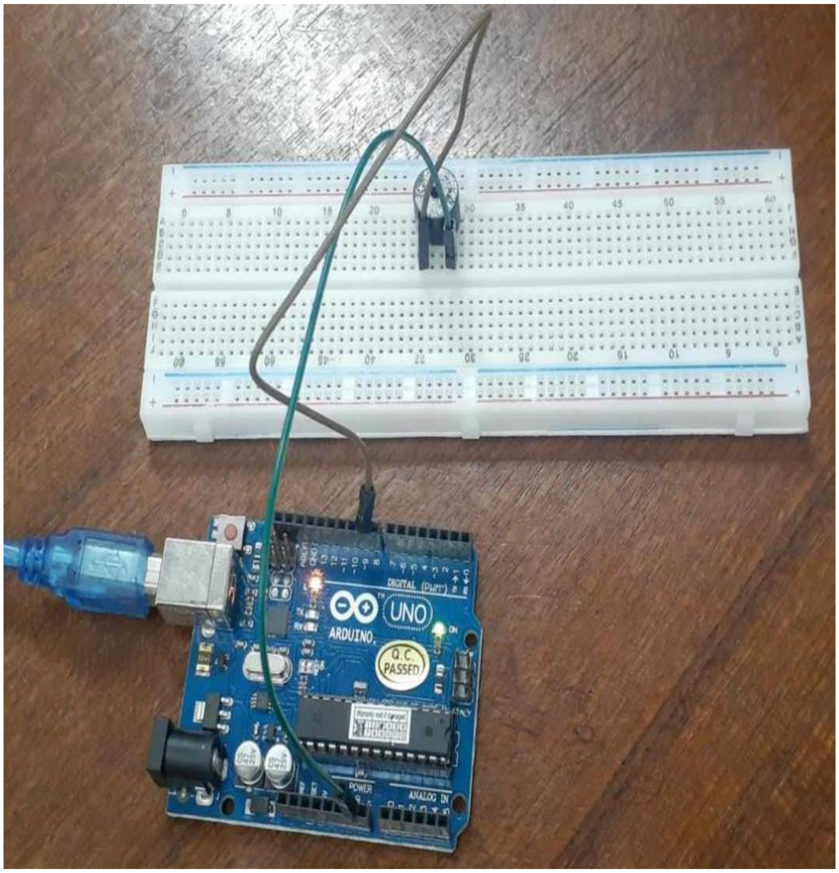
```
int buzzerpin = 9;
int length = 15;

char notes[]="ccggaagffeeddc ";

int beats[] = {1,1,1,1,1,1,2,1,1,1,1,1,1,2,4};

int tempo = 300;


void playTone(int tone,int duration) {

  for (long i=0;i<duration*1000L; i+= tone*2) {
    digitalWrite(buzzerpin,HIGH);
    delayMicroseconds(tone);
    digitalWrite(buzzerpin,LOW);
    delayMicroseconds(tone);

  }

}
```

```
void playNote(char note,int duration) {

char names[] = {'c','d','e','f','g','a','b','c'};

int tones[] = {1915,1700,1519,1432,1275,1136,1014,956};


   for(int i=0;i<8;i++) {
    if(names[i] == note) {
    playTone(tones[i],duration);

    }

   }

  }


  void setup() {

   // put your setup code here, to run once:
   pinMode(buzzerpin, OUTPUT);


  }


  void loop() {

   for(int i=0;i<length;i++) {
    if(notes[i]==' ') {
    delay(beats[i]*tempo);

    }

    else {
    playNote(notes[i],beats[i]*tempo);

    }

   delay(tempo/2);

   }   // put your main code here, to run repeatedly:

  }
```

Upload & Execution:

- Click on the "Upload" button to transfer the code to the Arduino board.
- Power up the Arduino board by connecting it to the computer.
- Observe and listen to the generated melody from the buzzer.

Results and Discussions:



Upon executing the code, the buzzer produced the programmed melody. By altering the frequencies assigned to each note in the code, different melodies can be generated. The **playTone()** function in Arduino enables the creation of various musical tones by controlling the frequency and duration of the notes played.

The relationship between the code and the sound output was observed to be direct. Changes in the code, such as modifying note frequencies or adjusting the duration of each note, directly affected the melody produced by the buzzer.

Conclusion:

After performing this experiment, we will be able to successfully demonstrate the process of generating musical tones and playing melodies using an Arduino and a buzzer, this lab underscored the inherent relationship between code implementation and sound output. The project illuminated the versatility of Arduino in generating musical outputs, showcasing its adaptability in creating diverse melodies. The understanding gained in manipulating frequencies and durations within the code not only facilitated the creation of varied musical compositions but also laid the foundational knowledge for engaging in more intricate and complex musical projects using Arduino technology.