University of Science and Technology Chittagong (USTC)
Faculty of Science, Engineering & Technology
Department of Computer Science & Engineering

Project report on
8x8x8 LED CUBE that turns on by clapping, dances with music, reacts with sound and bits.

Course code : CSE 316

Course Title : Microprocessor and Interfacing Lab

Team members :

Abhilash Chowdhury (GL)	ID: 0022210005101010
Pritom Gupta	ID: 0022210005101034
Tanmay Das	ID: 0022210005101005
Tazrian Binte Mahfuz	ID: 0022210005101016
Nusrat Chowdhury	ID: 0022210005101019
Nushrat Jahan Meem(Nishu)	ID: 0022210005101027

Batch : 38th
Semester: 5th
Department : CSE, FSET

Teacher in charge:

Prianka Das
[Lecturer]
USTC

TABLE OF CONTENTS

	TITLE	Page No.
	ABSTRACT	**1**
	KEYWORDS	**1**
1.0	**INTRODUCTION**	**2**
2.0	**BACKGROUND**	**2-3**
	2.1 Project	
	2.2 Simulator	
3.0	**PROJECT EVALUATION**	**3**
4.0	**CRITICAL EVALUATION**	**4**
5.0	**CONCLUSION**	**5**
6.0	**ACKNOWLEDGEMENT**	**5**
7.0	**REFERENCES**	**5**
8.0	**APPENDIX**	**6-8**
	8.1 Code implementation of project	

ABSTRACT

This project showcases the creation of an 8x8x8 LED cube that harnesses multiple sensory inputs, including clapping, dances with music, reacts with sound and bits, to craft an immersive and interactive experience. The system, powered by an Arduino microcontroller, seamlessly integrates various sensory modalities to generate a dynamic and engaging visual display. The system's key functionalities include:

1. **Clapping Control**: The LED cube illuminates in response to a detected clap, with the intensity and duration of the clap influencing the light patterns.
2. **Music Control**: The LED cube adapts to music by altering patterns and colors based on the tempo, rhythm, and melody.
3. **Sound Control**: The LED cube reacts to diverse sounds by modifying patterns and colors according to the sound's frequency, amplitude, and duration.

The 8x8x8 LED cube project exemplifies the integration of sensory inputs and outputs to create an immersive and interactive lighting system. By responding to clapping, music, sound, and bit control, the system offers a unique and captivating experience. This project highlights the potential of interactive lighting systems in various applications, including entertainment, education, and art.

KEYWORD

- 8x8x8 LED cube
- 3D LED matrix
- Sensory inputs
- Multiplexing
- Clap detection
- Music control
- Bit control
- Arduino microcontroller
- Anode/cathode control
- Diffused LEDs
- Soldering

INTRODUCTION

The 8x8x8 LED cube project is an interactive lighting system that responds to a wide range of sensory inputs, including clapping, music, and sound bits. By leveraging the capabilities of an Arduino microcontroller, this project seamlessly integrates sensory data processing and LED cube control to create a dynamic and captivating visual display.

BACKGROUND

❖ Project:

⌗ **Materials and Components**

- **LEDs and Construction**: 512pcs 5mm 2pin GreenLED, soldering iron, soldering lead, bredboard, connecting wires, jumper wires(male to male, female to male), wire cutters.
- **Control and Sound Processing**: Arduino Uno, clapping sensor module, sound sensor module, current-limiting resistors, power supply.

⌗ **Building the LED Cube**

1. **Planning**: Draw an 8x8 grid for LED placement, connecting each LED's cathode to a common layer and each anode to a column(of length-width roughly 2cm to separate the layers).
2. **Assembling Layers**: Solder LEDs into 8x8 grids to form layers. Test each layer before stacking.
3. **Connecting Layers**: Stack layers and connect cathodes vertically to form columns. Use bare copper wire for stability.

⌗ **Hardware Setup**

1. **Microcontroller**: Use an Arduino Uno, planning pin assignments for 64 columns and 8 layers.
2. **Driver Circuit**: Implement jumper wire(male to male)to control current to each layer. Connect columns to bredbroad using jumper wire(female to male) via resistors.
3. **Sound Sensor**: Connect the clapping sensor module and sound sensor module to an analog input on the Arduino. Adjust sensitivity with a potentiometer.

⌗ **Software Development**

Clapping Detection: Write code to sample clapping sensor data, detect claps, and alternate LEDs with stabilize.

Sound Processing and Music Synchronization: Use sound sensor module to process sound, extract frequency components, and create LED patterns based on beats.

⌗ **Integration and Testing**

1. **Connecting Components**: Ensure all hardware components are connected correctly.
2. **Uploading Software**: Upload the software to the Arduino and test the clapping detection feature.
3. **Music Synchronization**: Play music to verify LED patterns and adjust parameters for optimal performance.

⌗ **Challenges and Solutions**

1. **Noise and Interference**: Use software filters and sound sensor module to reduce noise.
2. **Power Management**: Ensure stable power supply and proper grounding.

⌗ **Conclusion and Future Work**

This project demonstrates creating an 8x8x8 LED cube that turns on by clapping, dances with music, and reacts to sound. Future improvements could include more complex LED patterns, better sound processing algorithms, and merger with smart home devices.

❖ **Simulator:**

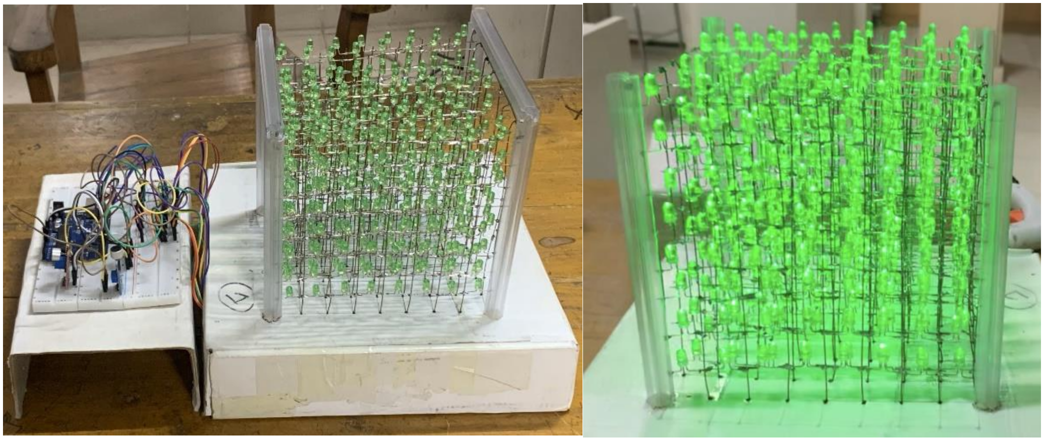

Fig: 8x8x8 3D LED cube full structure(left) and LED cube running(right)

PROJECT EVALUATION

The culmination of our project is a testament to the collective efforts and dedication of our entire team. Below is a breakdown of the contributions made by each team member:

❑ **Shopping:**
Abhilash and Pritom were responsible for procuring all necessary products.

❑ **Soldering:**
Abhilash and Tazrian soldered and prepared the first four LED walls, while Pritom and Tanmay handled the remaining four.

❑ **Structure Building and Wire Connection:**
Nishu and Nusrat were responsible for constructing the final structure and connecting the wires.

❑ **Coding:**
Abhilash developed the code for the Arduino.

❑ **Circuit Diagram:**
Pritom created the circuit diagram.

❑ **Lab Report:**
Tanmay authored the lab report.

❑ **Presentation:**
Tazrian, Nusrat, and Nishu collaborated on the presentation.

This comprehensive evaluation highlights the significant contributions of each team member, demonstrating the collaborative spirit and diverse skillset that led to the successful completion of our project.

CRITICAL EVALUATION

Technical Aspects:

- **Complexity:** The project involves complex electronics and programming, which can be challenging for those without extensive experience.
- **Noise Interference:** Noise interference from other electronic devices can affect the cube's performance.
- **Power Management:** The cube requires a stable power supply to ensure reliable performance.

Aesthetic and Functional Aspects:

- **Visual Appeal:** The cube's design is unique and visually captivating, showcasing the team's creativity and technical skill.
- **Interactivity:** The cube's ability to respond to sound and music through the use of a sound sensor and sound processing algorithms is a testament to the team's problem-solving abilities.

Future Enhancements and Applications:

- **Enhanced Animations:** Future enhancements could include more advanced LED animations, improving the cube's visual appeal and interactivity.
- **Improved Sound Analysis:** Improved sound analysis algorithms could enhance the cube's ability to react to sound and music, creating a more immersive experience.
- **Integration with Smart Home Technologies:** The cube could be integrated with other smart home technologies to expand its functionality and user experience.

Threats and Challenges:

- **Noise Interference:** Noise interference from other electronic devices can affect the cube's performance.
- **Power Management:** The cube requires a stable power supply to ensure reliable performance.
- **Limited Resources:** Limited resources, such as budget constraints or limited access to specific components, can hinder the project's progress.

CONCLUSION

The 8x8x8 LED cube project, which turns on by clapping, dances with music, and reacts to sound and beats, showcases a fusion of electronics, programming, and creative design. This interactive and visually captivating project highlights the versatility and power of microcontrollers like the Arduino Uno.

Throughout the project, we explored the meticulous process of constructing an 3D LED cube, involving careful planning, soldering, and layering of 512 LEDs into a precise 8x8x8 grid. The integration of a sound sensor and Sound processing algorithms enabled the cube to dynamically react to sounds in its environment, creating engaging light patterns synchronized with music and reacting to environmental sounds.

Challenges such as noise interference and power management were addressed with practical solutions, ensuring reliable performance and stable power supply. Incorporating clapping detection introduced a user-friendly interaction feature, enhancing the overall functionality and user experience.

This project not only provided a hands-on learning experience in electronics and coding but also resulted in a stunning piece of interactive art. Future enhancements could include more advanced LED animations, improved sound analysis algorithms, and integration with other smart home technologies for expanded functionality.

In summary, the 8x8x8 LED cube is a demonstration to the creativity and technical skills required to bring together various components into a consistent and impressive final product. It serves as a rewarding project for supporters looking to explore deeper into the domains of dynamic electronics and sound-reactive lighting systems.

ACKNOWLEDGEMENT

We would like to express our gratitude to the project guide, our course teacher **Mrs. Prianka Das,** who gave us the opportunity to do this wonderful project on the topic "**8x8x8 LED CUBE that turns on by clapping, dances with music, reacts with sound and bits**", which also helped us in doing a lot of research and we came to know about so many new things.

REFERENCES

Websites:

□ **https://www.jameco.com/Jameco/workshop/JamecoFavorites/8x8x8-led-cube.html**

Videos:

□ **https://youtu.be/Pdinux1xKcA?si=2PEPZG1CU3lDUCYp**

□ **https://youtu.be/uneaM07mwtg?si=1vCD0quCcwcmlrUc**

APPENDIX

⌖Implementation:

```
#define Sensor 8
#define RELAY_PIN 12
#define soundPin A0
#define LED1 2
#define LED2 3
#define LED3 4
#define LED4 5
#define LED5 6
#define LED6 7
#define LED7 9
#define LED8 10

int clap = 0;
long detection_range_start = 0;
long detection_range = 0;
long lastSoundDetected = 0;
boolean status_lights = false;

void setup() {
  pinMode(Sensor, INPUT);
  pinMode(13, OUTPUT);
  pinMode(RELAY_PIN, OUTPUT);
  digitalWrite(RELAY_PIN, HIGH);

  Serial.begin(9600);
  pinMode(LED1, OUTPUT);
  pinMode(LED2, OUTPUT);
  pinMode(LED3, OUTPUT);
  pinMode(LED4, OUTPUT);
  pinMode(LED5, OUTPUT);
  pinMode(LED6, OUTPUT);
  pinMode(LED7, OUTPUT);
  pinMode(LED8, OUTPUT);
}

void loop() {
  int status_sensor = digitalRead(Sensor);

  // Clap detection
  if (status_sensor == 0) {
    if (clap == 0) {
      detection_range_start = detection_range = millis();
    clap++;
  } else if (clap > 0 && millis() - detection_range >= 50) {
    detection_range = millis();
    clap++;
  }
}

// Check for double clap
if (millis() - detection_range_start >= 400) {
  if (clap == 2) {
    if (!status_lights) {
      status_lights = true;
      digitalWrite(13, HIGH);
      digitalWrite(RELAY_PIN, LOW);
      lastSoundDetected = millis(); // Reset sound detection timer
    } else if (status_lights) {
      status_lights = false;
      digitalWrite(13, LOW);
      digitalWrite(RELAY_PIN, HIGH);
      turnOffLEDs();
    }
  }
  clap = 0;
}

// If the system is active, run the music reactive LED code
if (status_lights) {
  long sum = 0;
  for (int i = 0; i < 100; i++) {
    sum += analogRead(soundPin);
  }
  sum = sum / 100; // average the sample of sound

  if (sum >= 450) digitalWrite(LED1, HIGH); else digitalWrite(LED1, LOW);
  if (sum >= 350) digitalWrite(LED2, HIGH); else digitalWrite(LED2, LOW);
  if (sum >= 250) digitalWrite(LED3, HIGH); else digitalWrite(LED3, LOW);
  if (sum >= 100) digitalWrite(LED4, HIGH); else digitalWrite(LED4, LOW);
  if (sum >= 100) digitalWrite(LED5, HIGH); else digitalWrite(LED5, LOW);
  if (sum >= 250) digitalWrite(LED6, HIGH); else digitalWrite(LED6, LOW);
  if (sum >= 350) digitalWrite(LED7, HIGH); else digitalWrite(LED7, LOW);
  if (sum >= 450) digitalWrite(LED8, HIGH); else digitalWrite(LED8, LOW);

  delay(10);
  Serial.println(sum);

  // Update last sound detected time if there is sound
```

```
    if (sum >= 50) {
      lastSoundDetected = millis();
    }

    // Check if 10 seconds have passed with no sound
    if (millis() - lastSoundDetected >= 10000) {
      status_lights = false;
      digitalWrite(13, LOW);
      digitalWrite(RELAY_PIN, HIGH);
      turnOffLEDs();
    }
  }
}

// Function to turn off all LEDs
void turnOffLEDs() {
  digitalWrite(LED1, LOW);
  digitalWrite(LED2, LOW);
  digitalWrite(LED3, LOW);
  digitalWrite(LED4, LOW);
  digitalWrite(LED5, LOW);
  digitalWrite(LED6, LOW);
  digitalWrite(LED7, LOW);
  digitalWrite(LED8, LOW);
}
```