Practical Robotics Projects with Arduino

(CSE 4571)

Lab Assignment No – 07

NeoPixel Ring Programming

Submission Date: _____

Branch: CSE		**Section: 2241026**
Name	**Registration No.**	**Signature**
Dinanath Dash	2241004161	

Department of Computer Science and Engineering
Institute of Technical Education and Research (Faculty of Engineering)
Siksha 'O' Anusandhan (Deemed to be University)
Bhubaneswar, Odisha-751030.

Aim:

NeoPixel Ring Programming – To Program a WS2812B NeoPixel LED Strip with Arduino UNO using the Adafruit library for multicolor and brightness effects.

Objectives:

1) **Gain familiarity with the WS2812B NeoPixel LED strip.**
2) **To write an Arduino code that controls individual pixel colours using setPixelColor().**
3) **Write an Arduino sketch to explore brightness control of a WS2812B NeoPixel 16 LEDs in a strip using the setBrightness() function. This function will control the global brightness based on user input from a push button.**
   - 3.1) **Write an Arduino sketch to control the 5 different global brightness levels based on user input from a push button.**
   - 3.2) **Write an Arduino sketch to control the brightness fades UP and DOWN smoothly each time the button is pressed, instead of jumping in steps. (1st press fade UP, 2nd press fade DOWN, 3rd fade UP again and so on).**
4) **To develop an Arduino sketch that generates static multi-colour patterns on the LED strip.**
5) **To develop Arduino code for multi-colour effects such as rainbow, fading, and chasing animations.**

Pre-Lab Questionnaire:

A. Experiment-Specific

1) What is the WS2812B LED commonly called in Adafruit terminology?
2) Which communication protocol is used by WS2812B LEDs?
3) Why is an external resistor (≈330–470 Ω) recommended on the data line of NeoPixels?
4) Why is a capacitor (≈1000 µF) recommended across the LED power supply?
5) What Arduino library is most commonly used to control WS2812B LEDs?
6) Which Arduino UNO pin is commonly used as the data pin for NeoPixels in example codes?
7) How many bytes are required to represent the color of one WS2812B LED?
8) Why must we call pixels.begin() in the setup function?
9) What power supply voltage is typically needed for WS2812B LEDs?
10) What is the typical maximum current consumption of one WS2812B LED at full brightness white?

Answers to Pre-Lab Questions

Components/Equipment Required:

Sl. No.	Name of the Component / Equipment	Specification	Quantity
1)	Arduino UNO R3	16MHz	1
2)	Arduino UNO cable	USB Type A to Micro-B	1
3)	WS2812B NeoPixel 16 LED strip	Any colour of your choice	1
4)	Breadboard	840 Tie points	1
5)	Jumper Wire	----------	As per requirement

Objective 2

To write an Arduino code that controls individual pixel colours using setPixelColor().

Circuit / Schematic Diagram

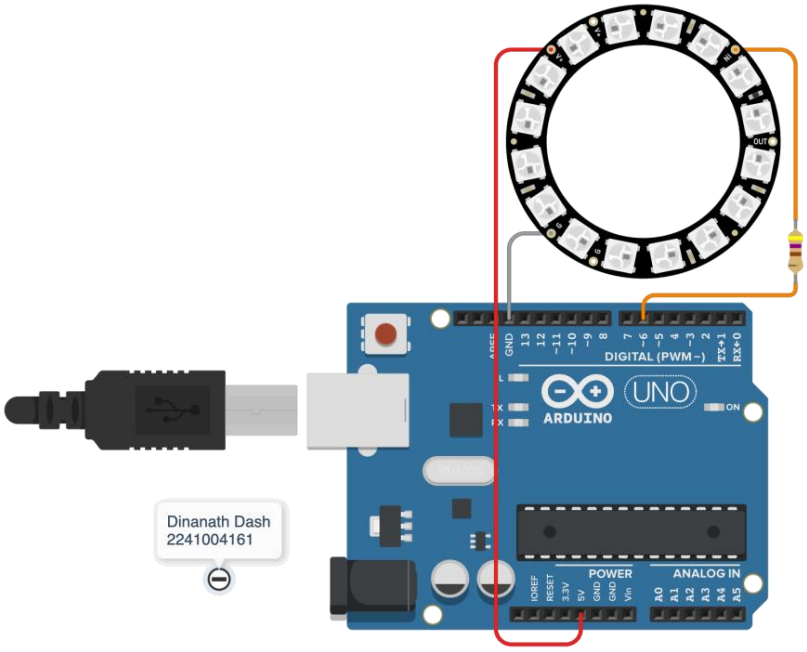


Figure 1: Controls of individual WS2812B NeoPixel colours

Code

Write an Arduino code that controls individual pixel colours using setPixelColor().

```
#include <Adafruit_NeoPixel.h>
#define PIN 6
#define NUMPIXELS 16
Adafruit_NeoPixel strip(NUMPIXELS, PIN, NEO_GRB + NEO_KHZ800);
void setup() {
  strip.begin();
  strip.show();
}
void loop() {
  strip.setPixelColor(0, strip.Color(255, 0, 0));
  strip.setPixelColor(1, strip.Color(0, 255, 0));
  strip.setPixelColor(2, strip.Color(0, 0, 255));
  strip.setPixelColor(3, strip.Color(255, 255, 0));
  strip.setPixelColor(4, strip.Color(0, 255, 255));
  strip.setPixelColor(5, strip.Color(255, 0, 255));
  strip.setPixelColor(6, strip.Color(255, 255, 255));
  strip.show();
  while (1);
}
```

Observation

Figure 2: (Simulation-based controls of individual WS2812B NeoPixel colours)

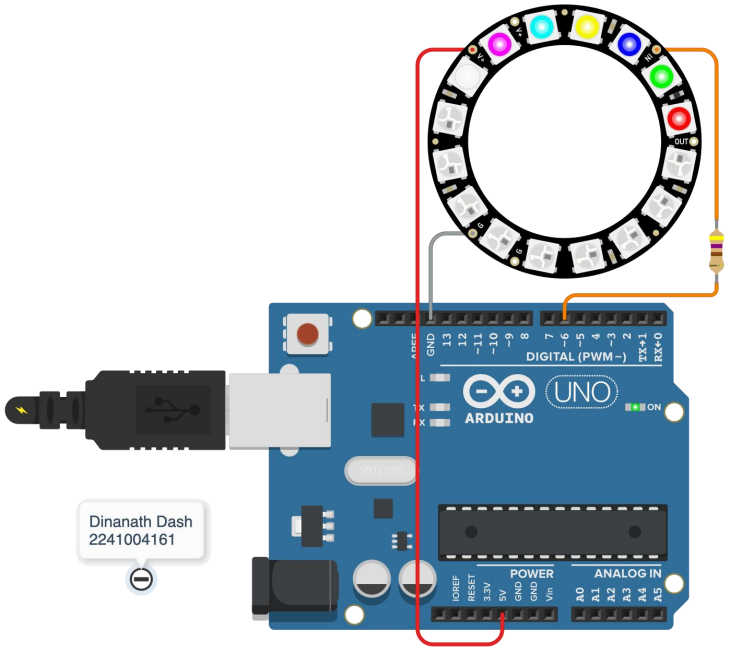
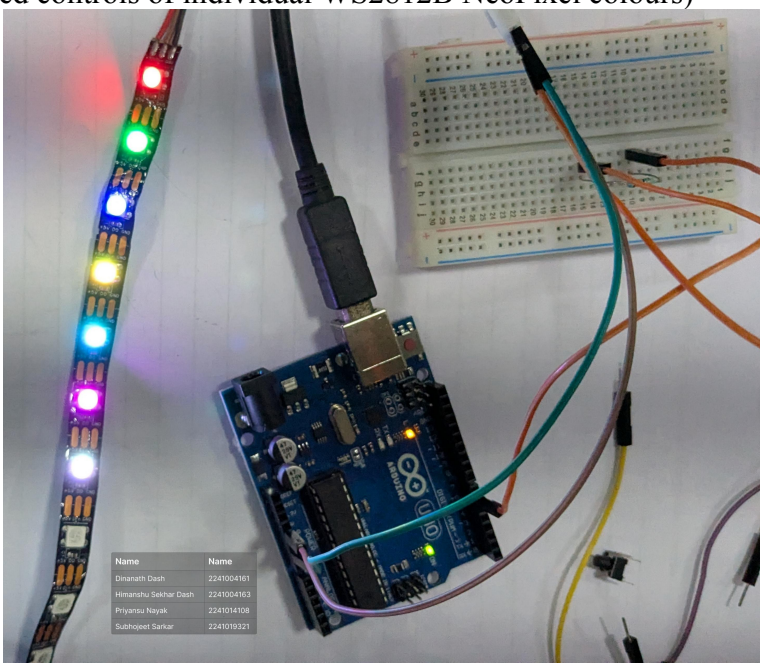


Figure 3: (Hardware-based controls of individual WS2812B NeoPixel colours)



Objective 3

Write an Arduino sketch to explore brightness control of a WS2812B NeoPixel 16 LEDs in a strip using the setBrightness() function. This function will control the global brightness based on user input from a push button.

Code

3.1) Write an Arduino sketch to control the 5 different global brightness levels based on user input from a push button.

```
#include <Adafruit_NeoPixel.h>
#define PIN 6
#define NUMPIXELS 16
#define BUTTON 2
Adafruit_NeoPixel strip(NUMPIXELS, PIN, NEO_GRB + NEO_KHZ800);
int brightnessLevels[] = {20, 60, 120, 180, 255};
int levelIndex = 0;
void setup() {
  pinMode(BUTTON, INPUT_PULLUP);
  strip.begin();
  strip.show();
  strip.setBrightness(brightnessLevels[levelIndex]);
}
```

```
void loop() {
  if (digitalRead(BUTTON) == LOW) {
    levelIndex = (levelIndex + 1) % 5;
    strip.setBrightness(brightnessLevels[levelIndex]);
    colorWipe(strip.Color(0, 0, 255), 10);
    delay(300);
  }
}
```

```
void colorWipe(uint32_t color, int wait) {
  for (int i = 0; i < strip.numPixels(); i++) {
    strip.setPixelColor(i, color);
    strip.show();
    delay(wait);
  }
}
```

3.2) Write an Arduino sketch to control the brightness fades UP and DOWN smoothly each time the button is pressed, instead of jumping in steps. (1st press fade UP, 2nd press fade DOWN, 3rd fade UP again and so on).

```
#include <Adafruit_NeoPixel.h>
#define PIN 6
#define NUMPIXELS 16
#define BUTTON 2
Adafruit_NeoPixel strip(NUMPIXELS, PIN, NEO_GRB + NEO_KHZ800);
bool fadeUp = true;
bool lastButton = HIGH;
void setup() {
  pinMode(BUTTON, INPUT_PULLUP);
  strip.begin();
  strip.show();
}
void loop() {
  bool currentButton = digitalRead(BUTTON);
  if (lastButton == HIGH && currentButton == LOW) {
    fadeUp = !fadeUp;
    if (fadeUp) fadeBrightness(0, 255);
    else fadeBrightness(255, 0);
  }
  lastButton = currentButton;
}
void fadeBrightness(int start, int end) {
  int step = (start < end) ? 1 : -1;
  for (int b = start; b != end; b += step) {
    strip.setBrightness(b);
    colorShow(strip.Color(255, 100, 0));
    delay(5);
  }
}
void colorShow(uint32_t color) {
  for (int i = 0; i < strip.numPixels(); i++)
    strip.setPixelColor(i, color);
  strip.show();
}
```

Circuit / Schematic Diagram

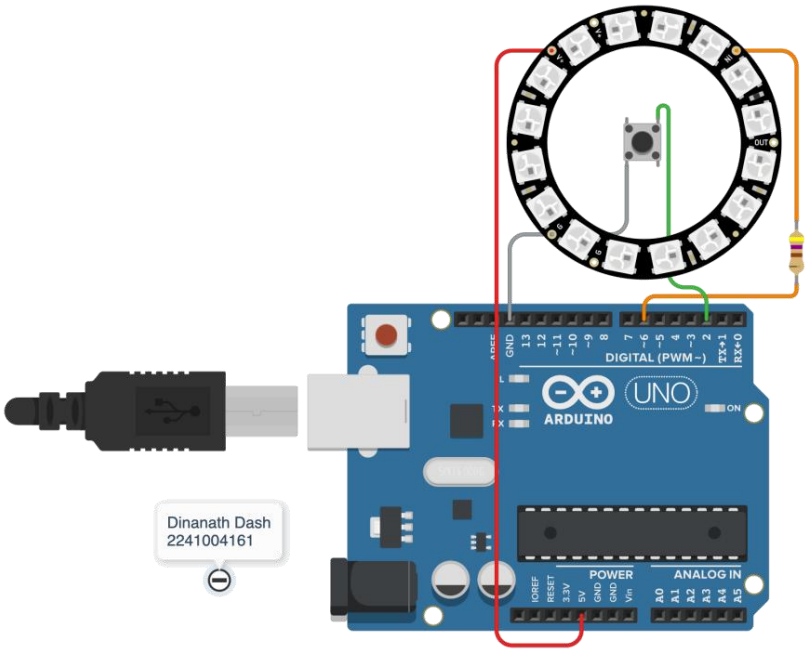


Figure 4: Brightness control of a WS2812B NeoPixel 16 LEDs in a strip

Observation

Figure 5: (Simulation-based brightness control of a WS2812B NeoPixel 16 LEDs in a strip)

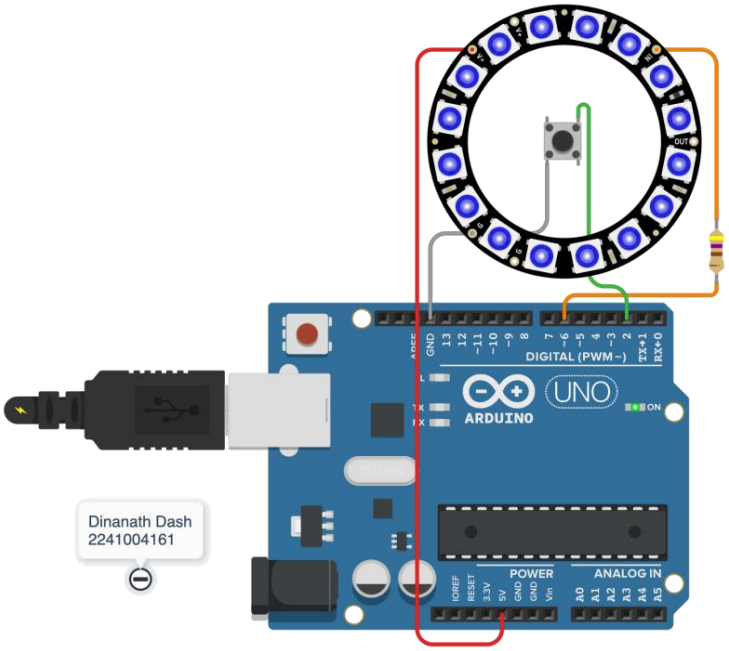
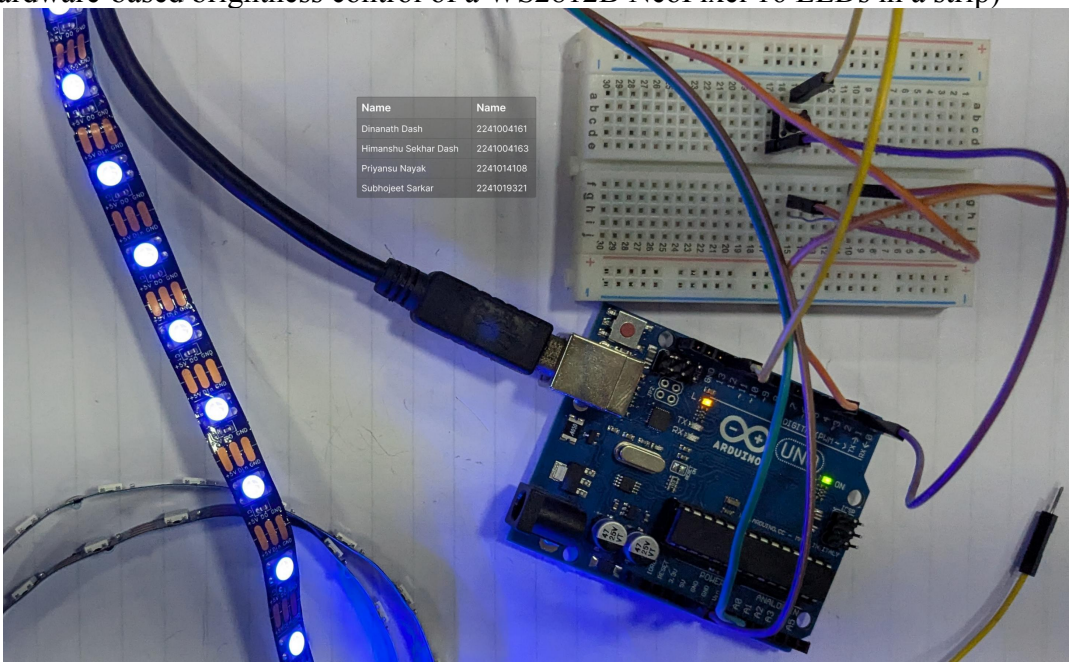


Figure 6: (Hardware-based brightness control of a WS2812B NeoPixel 16 LEDs in a strip)



Objective 4

To develop an Arduino sketch that generates static multi-colour patterns on the LED strip.

Code

Write an Arduino sketch that generates static multi-colour patterns on the LED strip

```
#include <Adafruit_NeoPixel.h>
#define PIN 6
#define NUMPIXELS 16
Adafruit_NeoPixel strip(NUMPIXELS, PIN, NEO_GRB + NEO_KHZ800);
void setup() {
  strip.begin();
  strip.show();
}
void loop() {
  for (int i = 0; i < strip.numPixels(); i++) {
    if (i % 3 == 0) strip.setPixelColor(i, strip.Color(255, 0, 0)); // Red
    else if (i % 3 == 1) strip.setPixelColor(i, strip.Color(0, 255, 0)); // Green
    else strip.setPixelColor(i, strip.Color(0, 0, 255));
  }
  strip.show();
```

while (1); *}*

Circuit / Schematic Diagram

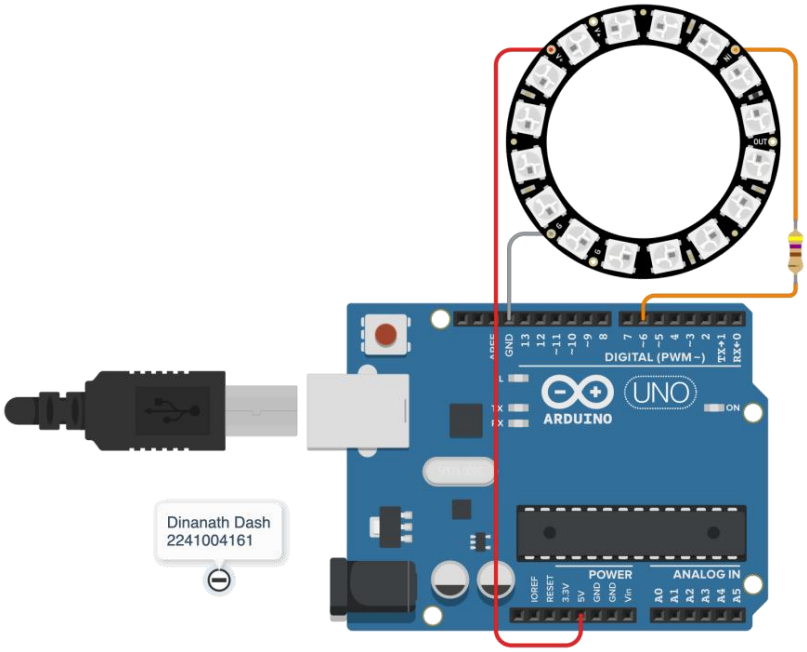


Figure 7: Static multi-colour patterns on the LED strip

Observation

Figure 8: (Simulation-based Static multi-colour patterns on the LED strip)

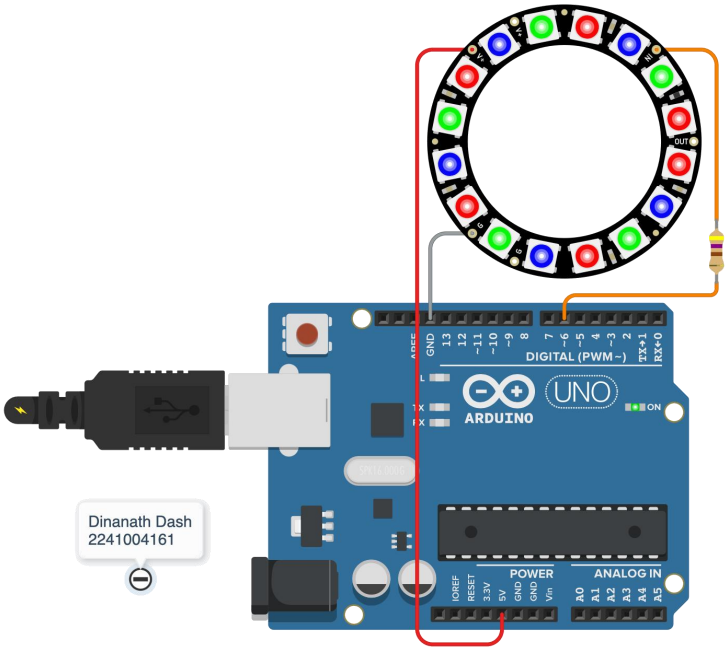
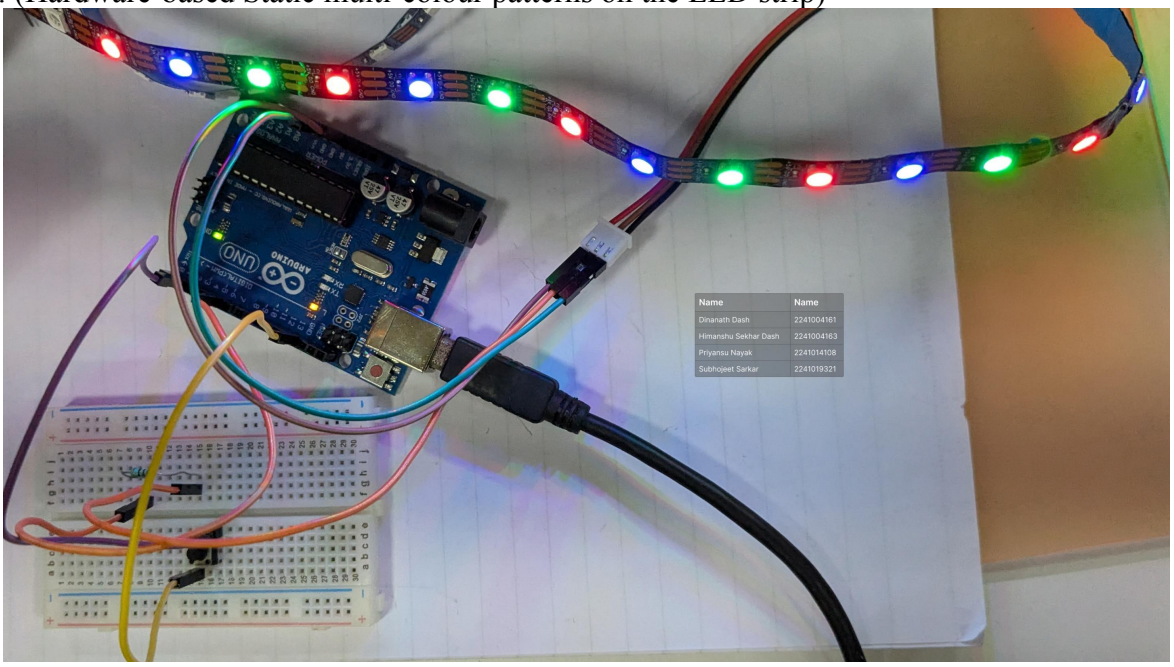


Figure 9: (Hardware-based Static multi-colour patterns on the LED strip)



Objective 5

To develop Arduino code for multi-colour effects, such as rainbow, fading, and chasing animations.

Code

Write an Arduino code for multi-colour effects, such as rainbow, fading, and chasing animations.

```
#include <Adafruit_NeoPixel.h>
#define PIN 6
#define NUMPIXELS 16
Adafruit_NeoPixel strip(NUMPIXELS, PIN, NEO_GRB + NEO_KHZ800);
void setup() {
  strip.begin();
  strip.show();
}
void loop() {
  rainbow(10);     // Rainbow animation
  colorFade(10);   // Smooth fade
  chase(strip.Color(255, 0, 0), 50);  // Red chasing
}
void rainbow(int wait) {
  for (long firstPixelHue = 0; firstPixelHue < 5 * 65536; firstPixelHue += 256) {
    for (int i = 0; i < strip.numPixels(); i++) {
      int pixelHue = firstPixelHue + (i * 65536L / strip.numPixels());
      strip.setPixelColor(i, strip.gamma32(strip.ColorHSV(pixelHue)));
    }
    strip.show();
    delay(wait);
  }
}
void colorFade(int wait) {
  for (int r = 0; r < 256; r++) {
    fillColor(r, 0, 255 - r);
    delay(wait);
  }
  for (int r = 255; r >= 0; r--) {
    fillColor(r, 0, 255 - r);
    delay(wait);
  }
}
void fillColor(int r, int g, int b) {
  for (int i = 0; i < strip.numPixels(); i++)
    strip.setPixelColor(i, r, g, b);
  strip.show();
}
void chase(uint32_t color, int wait) {
  for (int i = 0; i < strip.numPixels(); i++) {
    strip.clear();
    strip.setPixelColor(i, color);
    strip.show();
    delay(wait);
  }

}
```

Circuit / Schematic Diagram

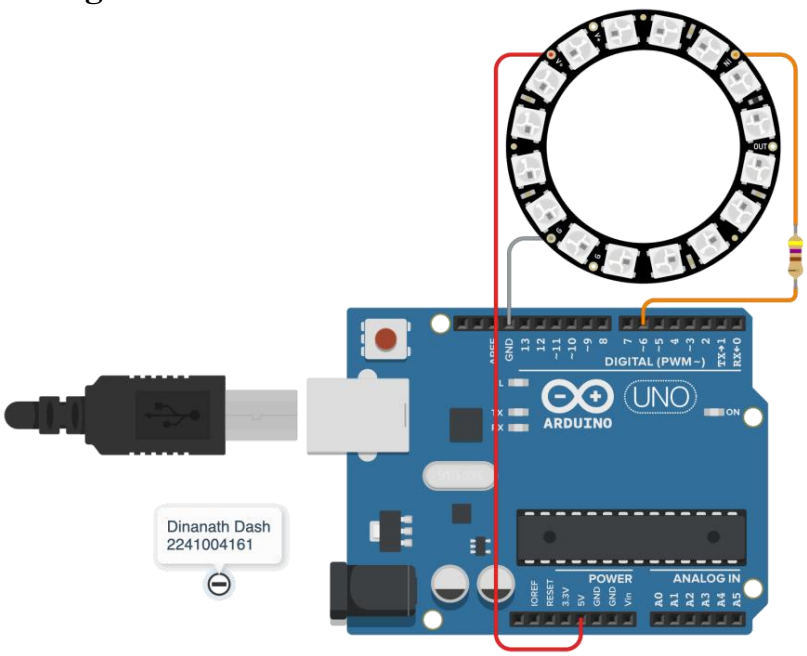


Figure 10: Multi-colour effects on the LED strip animation

Observation

Figure 11: (Simulation-based multi-colour effects on the LED strip animation)

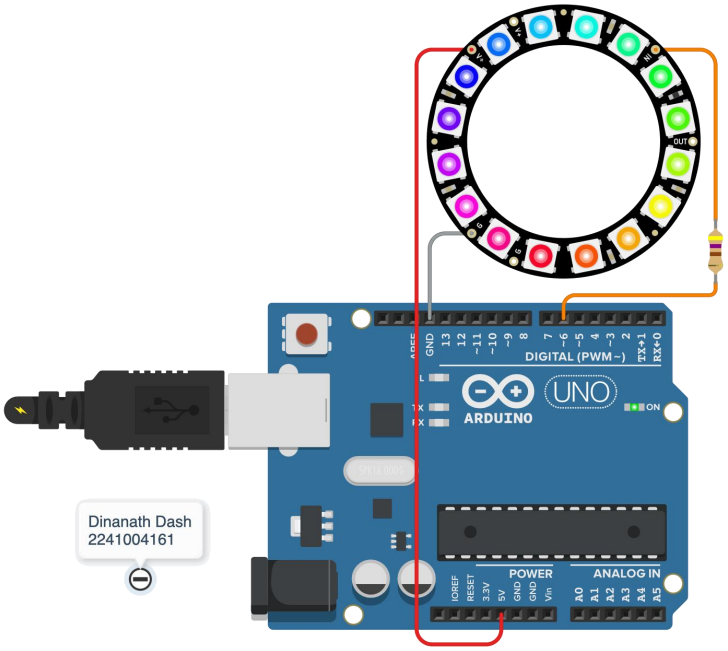
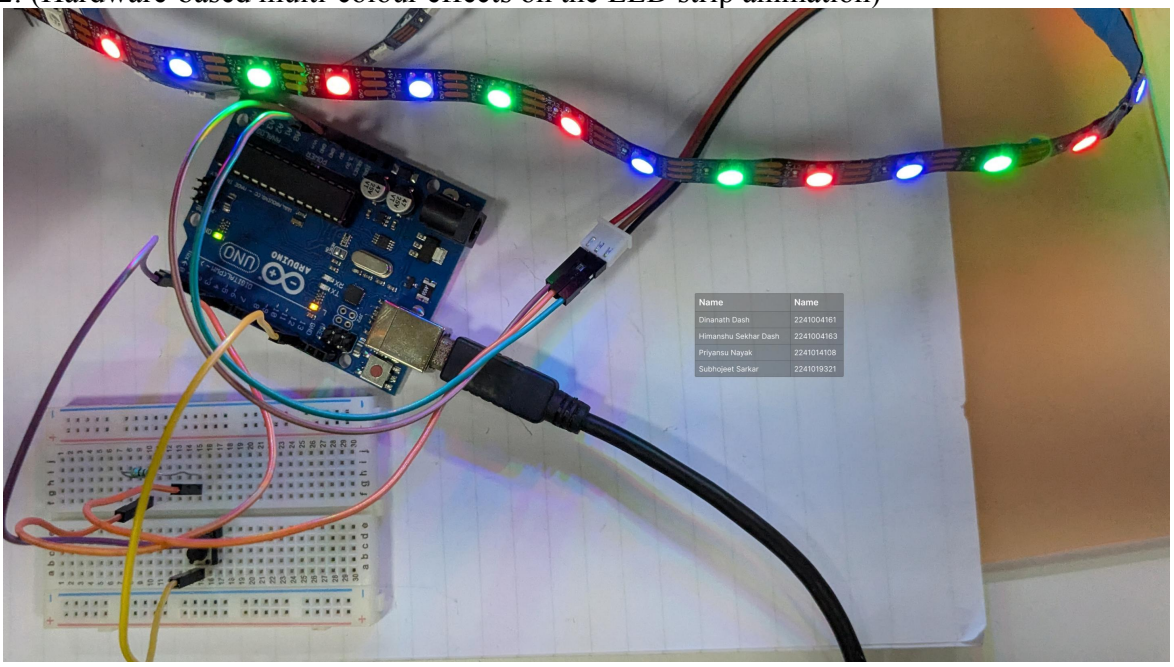


Figure 12: (Hardware-based multi-colour effects on the LED strip animation)



Conclusion

Precautions

Post Experiment Questionnaire:

1) What function is used to actually display the updated color values on the NeoPixel LEDs?
2) How do you set the brightness of all NeoPixels in the strip?
3) Which function is used to assign an RGB color value to a particular pixel?
4) What happens if you set pixel colors but forget to call pixels.show();?
5) What was observed when brightness was set to maximum for multiple LEDs?
6) How can you create a color-changing rainbow animation on NeoPixels?
7) What color order (RGB/GRB) did your NeoPixel strip use in the experiment?
8) What practical limitation did you observe when using Arduino UNO for NeoPixel control?

Answers to Post-Lab Questions

(Signature of the Faculty)	**(Signature of the Student)**
	Name:
Date:	**Registration No.:**
	Branch:
	Section