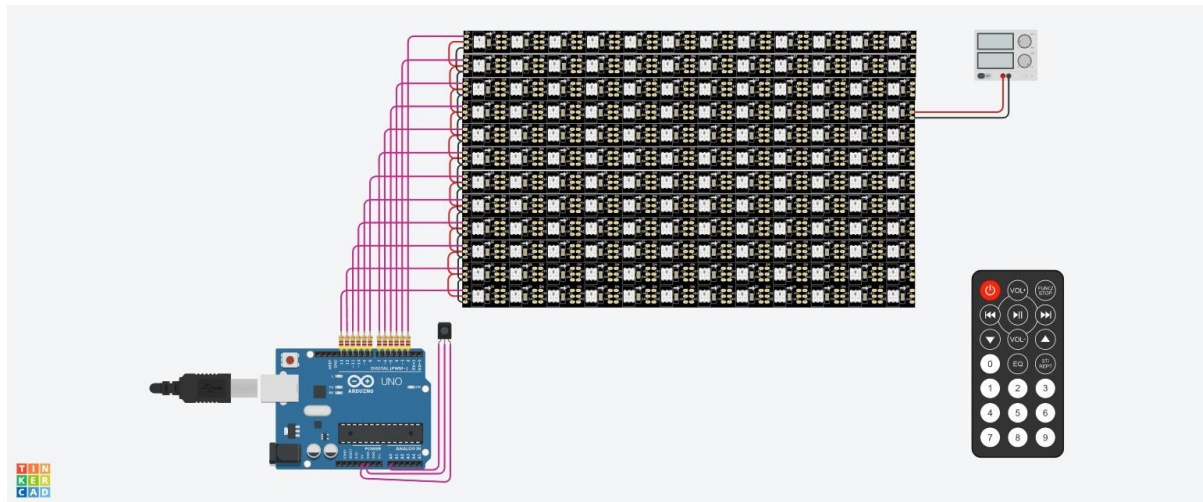


1. Use IR remote to switch between presenting the text "IITG" and displaying a star.

<https://www.tinkercad.com/things/7CjZRmpAvaA-cool-kup-lahdi/editel?sharecode=qp3Sv6YDBJYMdvEy-rVV-FmD7o3jjLNd8UvmiWDzanM>



Code:

```
#include <Adafruit_NeoPixel.h>

#ifdef __AVR__
#include <avr/power.h>
#endif

#include <IRremote.h>

#define NP1 2
#define NP2 3
#define NP3 4
#define NP4 5
#define NP5 6
```

```
#define NP6 7

#define NP7 8

#define NP8 9

#define NP9 10

#define NP10 11

#define NP11 12

#define NP12 13

#define NUM_PIXELS 12

Adafruit_NeoPixel Neo1(NUM_PIXELS, NP1, NEO_GRB + NEO_KHZ800);
Adafruit_NeoPixel Neo2(NUM_PIXELS, NP2, NEO_GRB + NEO_KHZ800);
Adafruit_NeoPixel Neo3(NUM_PIXELS, NP3, NEO_GRB + NEO_KHZ800);
Adafruit_NeoPixel Neo4(NUM_PIXELS, NP4, NEO_GRB + NEO_KHZ800);
Adafruit_NeoPixel Neo5(NUM_PIXELS, NP5, NEO_GRB + NEO_KHZ800);
Adafruit_NeoPixel Neo6(NUM_PIXELS, NP6, NEO_GRB + NEO_KHZ800);
Adafruit_NeoPixel Neo7(NUM_PIXELS, NP7, NEO_GRB + NEO_KHZ800);
Adafruit_NeoPixel Neo8(NUM_PIXELS, NP8, NEO_GRB + NEO_KHZ800);
Adafruit_NeoPixel Neo9(NUM_PIXELS, NP9, NEO_GRB + NEO_KHZ800);
Adafruit_NeoPixel Neo10(NUM_PIXELS, NP10, NEO_GRB + NEO_KHZ800);
Adafruit_NeoPixel Neo11(NUM_PIXELS, NP11, NEO_GRB + NEO_KHZ800);
Adafruit_NeoPixel Neo12(NUM_PIXELS, NP12, NEO_GRB + NEO_KHZ800);

void setup() {
  Neo1.begin();
  Neo2.begin();
  Neo3.begin();
  Neo4.begin();
  Neo5.begin();
  Neo6.begin();
  Neo7.begin();
  Neo8.begin();
```

```
Neo9.begin();
Neo10.begin();
Neo11.begin();
Neo12.begin();
IrReceiver.begin(A0);
Serial.begin(9600);
}

void straightLine(int i){
    uint32_t blueColor = Neo1.Color(0, 0, 255);
    Neo1.setPixelColor(i, blueColor);
    Neo2.setPixelColor(i, blueColor);
    Neo3.setPixelColor(i, blueColor);
    Neo4.setPixelColor(i, blueColor);
    Neo5.setPixelColor(i, blueColor);
    Neo6.setPixelColor(i, blueColor);
    Neo7.setPixelColor(i, blueColor);
    Neo8.setPixelColor(i, blueColor);
    Neo9.setPixelColor(i, blueColor);
    Neo10.setPixelColor(i, blueColor);
    Neo11.setPixelColor(i, blueColor);
    Neo12.setPixelColor(i, blueColor);
    Neo1.show();
    Neo2.show();
    Neo3.show();
    Neo4.show();
    Neo5.show();
    Neo6.show();
    Neo7.show();
    Neo8.show();
```

```

Neo9.show();

Neo10.show();

Neo11.show();

Neo12.show();


}

void showLetters() {
    uint32_t blueColor = Neo1.Color(0, 0, 255);

    // Printing I
    straightLine(0);
    //Printing I
    straightLine(2);
    //Printing T
    int i;

        i = 3;

        Neo1.setPixelColor(i, blueColor);

        Neo1.show();


        i = 4;

        Neo1.setPixelColor(i, blueColor);

        Neo1.show();

    straightLine(5);

        i = 6;

        Neo1.setPixelColor(i, blueColor);

        Neo1.show();

        i = 7;

        Neo1.setPixelColor(i, blueColor);

Neo1.show();

```

```
//Printing G  
straightLine(8);  
  
i = 9;  
Neo1.setPixelColor(i, blueColor);  
Neo12.setPixelColor(i, blueColor);  
Neo1.show();  
Neo12.show();  
  
i = 10;  
Neo1.setPixelColor(i, blueColor);  
Neo12.setPixelColor(i, blueColor);  
Neo8.setPixelColor(i, blueColor);  
Neo1.show();  
Neo8.show();  
Neo12.show();  
  
i = 11;  
Neo1.setPixelColor(i, blueColor);  
Neo2.setPixelColor(i, blueColor);  
Neo3.setPixelColor(i, blueColor);  
Neo8.setPixelColor(i, blueColor);  
Neo9.setPixelColor(i, blueColor);  
Neo10.setPixelColor(i, blueColor);  
Neo11.setPixelColor(i, blueColor);  
Neo12.setPixelColor(i, blueColor);  
Neo1.show();  
Neo2.show();  
Neo3.show();  
Neo8.show();  
Neo9.show();  
Neo10.show();
```

```
Neo11.show();
Neo12.show();
}
void showStar(){
    uint32_t nColor = Neo1.Color(255,255,0);
    Neo4.setPixelColor(0, nColor);
    Neo4.show();
    Neo4.setPixelColor(1, nColor);
    Neo4.show();
    Neo4.setPixelColor(2, nColor);
    Neo4.show();
    Neo3.setPixelColor(3, nColor);
    Neo3.show();
    Neo2.setPixelColor(4, nColor);
    Neo2.show();
    Neo1.setPixelColor(5, nColor);
    Neo1.show();
    Neo2.setPixelColor(6, nColor);
    Neo2.show();
    Neo3.setPixelColor(7, nColor);
    Neo3.show();
    Neo4.setPixelColor(8, nColor);
    Neo4.show();
    Neo4.setPixelColor(9, nColor);
    Neo4.show();
    Neo4.setPixelColor(10, nColor);
    Neo4.show();
    Neo5.setPixelColor(9, nColor);
    Neo5.show();
```

```
Neo6.setPixelColor(8, nColor);  
Neo6.show();  
Neo7.setPixelColor(7, nColor);  
Neo7.show();  
Neo8.setPixelColor(8, nColor);  
Neo8.show();  
Neo9.setPixelColor(8, nColor);  
Neo9.show();  
Neo10.setPixelColor(9, nColor);  
Neo10.show();  
Neo11.setPixelColor(9, nColor);  
Neo11.show();  
Neo10.setPixelColor(8, nColor);  
Neo10.show();  
Neo9.setPixelColor(7, nColor);  
Neo9.show();  
Neo8.setPixelColor(6, nColor);  
Neo8.show();  
Neo7.setPixelColor(5, nColor);  
Neo7.show();  
Neo5.setPixelColor(1, nColor);  
Neo5.show();  
Neo6.setPixelColor(2, nColor);  
Neo6.show();  
Neo7.setPixelColor(3, nColor);  
Neo7.show();  
Neo8.setPixelColor(2, nColor);  
Neo8.show();  
Neo9.setPixelColor(1, nColor);
```

```

Neo9.show();
Neo10.setPixelColor(0, nColor);
Neo10.show();
Neo11.setPixelColor(0, nColor);
Neo11.show();
Neo10.setPixelColor(1, nColor);
Neo10.show();
Neo9.setPixelColor(2, nColor);
Neo9.show();
Neo8.setPixelColor(3, nColor);
Neo8.show();
Neo7.setPixelColor(4, nColor);
Neo7.show();
}
void clearAll(){
    for(int pixel=0;pixel<12;pixel++){
        Neo1.setPixelColor(pixel, Neo1.Color(0, 0, 0));
        Neo2.setPixelColor(pixel, Neo2.Color(0, 0, 0));
        Neo3.setPixelColor(pixel, Neo3.Color(0, 0, 0));
        Neo4.setPixelColor(pixel, Neo4.Color(0, 0, 0));
        Neo5.setPixelColor(pixel, Neo4.Color(0, 0, 0));
        Neo6.setPixelColor(pixel, Neo4.Color(0, 0, 0));
        Neo7.setPixelColor(pixel, Neo4.Color(0, 0, 0));
        Neo8.setPixelColor(pixel, Neo4.Color(0, 0, 0));
        Neo9.setPixelColor(pixel, Neo4.Color(0, 0, 0));
        Neo10.setPixelColor(pixel, Neo4.Color(0, 0, 0));
        Neo11.setPixelColor(pixel, Neo4.Color(0, 0, 0));
        Neo12.setPixelColor(pixel, Neo4.Color(0, 0, 0));
    }
}

```



```

    Neo1.show();
    Neo2.show();
    Neo3.show();
    Neo4.show();
    Neo5.show();
    Neo6.show();
    Neo7.show();
    Neo8.show();
    Neo9.show();
    Neo10.show();
    Neo11.show();
    Neo12.show();
}

int mapCodeToButton(unsigned long code) {
    if ((code & 0x0000FFFF) == 0x0000BF00) {
        code >>= 16;
        if (((code >> 8) ^ (code & 0x00FF)) == 0x00FF) {
            return code & 0xFF;
        }
    }
    return -1;
}

int readInfrared() {
    int result = -1;
    if (IrReceiver.decode()) {
        unsigned long code = IrReceiver.decodedIRData.decodedRawData;
        result = mapCodeToButton(code);
        IrReceiver.resume();
    }
}

```

```

    return result;
}

int button=-1;

int flags=0;

void loop(){

    int reading = readInfrared();

    if (reading == 16) {

        if (flags==0) {

            clearAll();

            showLetters();

            flags = 1;

        }

        else{

            clearAll();

            showStar();

            flags = 0;

        }

    }

}

```

Approach:

Firstly, I used Neopixel LED Strips to construct a 12x12 LED matrix. NeoPixels are LEDs that can be controlled individually. Each NeoPixel LED contains a tiny microcontroller, which allows us to set its color independently. This means we can control the color and brightness of each LED in a chain or matrix. To make this task easier, I utilized a helpful toolkit called the "Adafruit NeoPixel library.". Next, I created two patterns "IITG" and star. Then, I used IR remote to switch between presenting the text "IITG" and displaying a star.