

## Додаток Г

### Програма RGB1

Програма демонструє роботу з RGB світлодіодом. Алгоритм програми простий – по черзі включаємо червоний, зелений та синій колір RGB світлодіода з паузою в 0,5 секунди.

#### *Лістинг програми RGB1*

```
const byte rPin = 11;
const byte gPin = 10;
const byte bPin = 9;

void setup() {
    pinMode( rPin, OUTPUT );
    pinMode( gPin, OUTPUT );
    pinMode( bPin, OUTPUT );
}

void loop() {
    digitalWrite( bPin, LOW );//Blue-off
    digitalWrite( rPin, HIGH );//Red-on
    delay( 500 );

    digitalWrite( rPin, LOW ); //Red-off
    digitalWrite( gPin, HIGH ); //Green-on
    delay( 500 );

    digitalWrite( gPin, LOW ); //Green-off
    digitalWrite( bPin, HIGH );//Blue-on
    delay( 500 );
}
```

### Програма RGB2

Програма демонструє роботу з RGB світлодіодом. Алгоритм програми простий – по черзі включаємо червоний, зелений та синій колір RGB світлодіода з паузою в 0,5 секунди з використанням масиву

#### *Лістинг програми RGB2*

```
const byte rgbPins[3] = {11,10,9};

void setup() {
    for( byte i=0; i<3; i++ )
        pinMode( rgbPins[i], OUTPUT );
}

void loop() {
    digitalWrite( rgbPins[2], LOW );
    digitalWrite( rgbPins[0], HIGH );
    delay( 500 );
}
```

```

    digitalWrite( rgbPins[0], LOW );
    digitalWrite( rgbPins[1], HIGH );
    delay( 500 );
    digitalWrite( rgbPins[1], LOW );
    digitalWrite( rgbPins[2], HIGH );
    delay( 500 );
}

```

### Програма RGB3

Програма демонструє роботу з RGB світлодіодом. У програмі використовується масив для вибору R, G або B світлодіода та двовірний масив для вибору кольору світіння RGB світлодіода. Алгоритм програми простий – по черзі включається червоний, жовтий, зелений, блакитний, синій, фіолетовий колір RGB світлодіода з паузою в 1 секунду.

#### *Лістинг програми RGB3*

```

const byte rgbPins[3] = {11,10,9};
const byte rainbow[6][3] = {
    {1,0,0}, // red
    {1,1,0}, // yellow
    {0,1,0}, // green
    {0,1,1}, // light blue
    {0,0,1}, // blue
    {1,0,1}, // purple
};
void setup() {
    for( byte i=0; i<3; i++ )
        pinMode( rgbPins[i], OUTPUT );
}

void loop() {
    for( int i=0; i<6; i++ ){
        for( int k=0; k<3; k++ ){
            digitalWrite(rgbPins[k], rainbow[i][k]);
        }
        delay( 1000 );
    }
}

```

### Програма RGB4

Програма демонструє роботу з RGB світлодіодом. У програмі використовується ШІМ для повільної зміни кольору світіння RGB світлодіода. Алгоритм програми простий – повільно виключаємо один колір та одночасно повільно включаємо інший колір.

#### *Лістинг програми RGB4*

```

const byte rgbPins[3] = {11,10,9};
int dim = 1;

```

```

void setup() {
    for(byte i=0; i<3; i++){
        pinMode( rgbPins[i], OUTPUT );
    }
    // початковий стан (Red - on)
    analogWrite(rgbPins[0], 255);
    analogWrite(rgbPins[1], 0);
    analogWrite(rgbPins[2], 0);
}

void loop() {
    for(int i=255; i>=0; i--){
        analogWrite( rgbPins[0], i/dim );//Red-off
        analogWrite( rgbPins[1], (255-i)/dim );//Green-on
        delay(10);
    }

    for(int i=255; i>=0; i--){
        analogWrite( rgbPins[1], i/dim );//Green-off
        analogWrite( rgbPins[2], (255-i)/dim );//Blue-on
        delay(10);
    }

    for(int i=255; i>=0; i--){
        analogWrite( rgbPins[2], i/dim );//Blue-off
        analogWrite( rgbPins[0], (255-i)/dim );//Red-on
        delay(10);
    }
}

```

## Програма RGB5

Програма демонструє роботу з RGB світлодіодом. У програмі використовується лічильник натиснення тактової кнопки для зміни 7 кольорів світіння RGB світлодіода з використанням функції «антидребезгу». Алгоритм програми простий – у початковому стані RGB світлодіод виключений, з кожним натисненням кнопки змінюється 1 з 7 кольорів світіння світлодіода.

### *Лістинг програми RGB5*

```

#define BLED 9 //D9 - Blue
#define GLED 10 //D10 - Green
#define RLED 11 //D11 - Red
#define BUTTON 2 //D2 - button

boolean lastButton = LOW; //попередній стан кнопки
boolean currentButton = LOW; //поточний стан кнопки
int ledMode = 0; //режим роботи

boolean debounce(boolean last);
void setMode(int mode);

void setup()

```

```

{
    pinMode (BLED, OUTPUT);
    pinMode (GLED, OUTPUT);
    pinMode (RLED, OUTPUT);
    pinMode (BUTTON, INPUT_PULLUP);
}

void loop()
{

    currentButton = debounce(lastButton); //читаємо стан кнопки
    if (lastButton == LOW && currentButton == HIGH) {
        ledMode++; }
    lastButton = currentButton;
    if (ledMode == 8) ledMode = 0;
    setMode(ledMode); //зміна режиму
}

//Функція антидребезгу
boolean debounce(boolean last)
{
    boolean current = digitalRead(BUTTON);
    if (last != current){
        delay(5);
        current = digitalRead(BUTTON);
    }
    return current;
}

//Вибір режиму роботи світлодіоду
void setMode(int mode)
{
    if (mode == 1)//Red
    {
        digitalWrite(RLED, HIGH);
        digitalWrite(GLED, LOW);
        digitalWrite(BLED, LOW);
    }
    else if (mode == 2)//Green
    {
        digitalWrite(RLED, LOW);
        digitalWrite(GLED, HIGH);
        digitalWrite(BLED, LOW);
    }
    else if (mode == 3)//Blue
    {
        digitalWrite(RLED, LOW);
        digitalWrite(GLED, LOW);
        digitalWrite(BLED, HIGH);
    }
    else if (mode == 4) //purple (Red + Blue)
    {
        analogWrite(RLED, 127);
        analogWrite(GLED, 0);
        analogWrite(BLED, 127);
    }
}

```

```

}
else if (mode == 5) // turquoise (Blue + Green)
{
    analogWrite(RLED, 0);
    analogWrite(GLLED, 127);
    analogWrite(BLED, 127);
}
else if (mode == 6) //orange(Green + Red)
{
    analogWrite(RLED, 127);
    analogWrite(GLLED, 127);
    analogWrite(BLED, 0);
}
else if (mode == 7) //white(Red + Green + Blue)
{
    analogWrite(RLED, 85);
    analogWrite(GLLED, 85);
    analogWrite(BLED, 85);
}
else// off
{
    digitalWrite(RLED, LOW);
    digitalWrite(GLLED, LOW);
    digitalWrite(BLED, LOW);
}
}

```