



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования

**«Самарский государственный технический университет»
(ФГБОУ ВО «СамГТУ»)**

Институт «Автоматики и инженерных технологий»

Основы работы с Arduino

Лабораторная работа №1, отчёт

Выполнил студенты

3 курса, 3-ИАИТ-110 группы

Беляков Даниил Андреевич

Питьев Дмитрий Артёмович

Чалый Антон Викторович

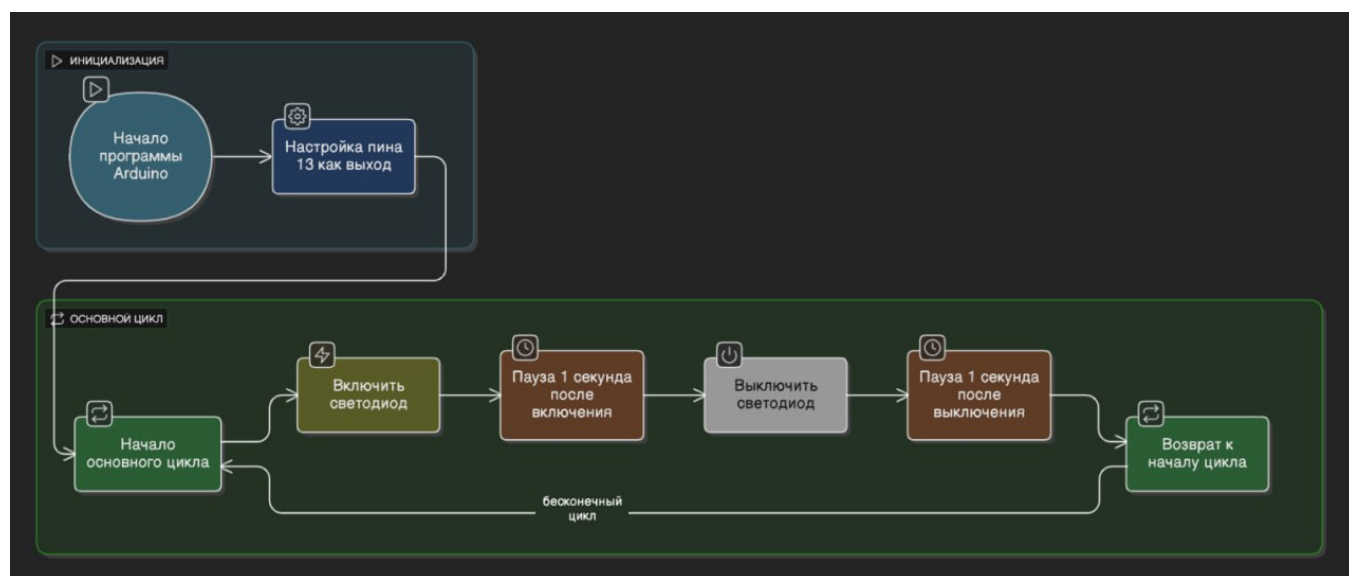
Самара, 2025 г.

Цель работы — изучить основы работы с ПО arduinoIDE, первое подключение arduinoUNO. Изучение основ работы со светодиодами, кнопками и потенциометрами

Задание 1: Подключение светодиода на плате arduinoUNO.

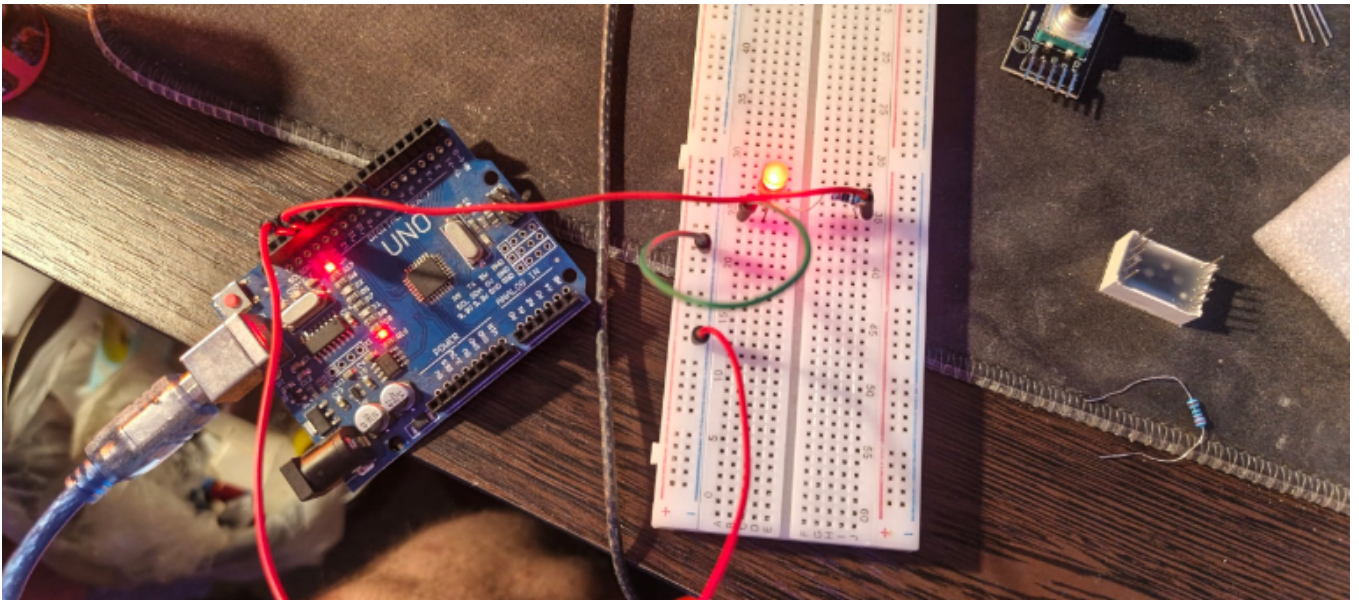
Написать скетч программы для мигания светодиода на плате ардуино, который находится на 13 пине.

```
1 void setup() {  
2     pinMode(13, OUTPUT);  
3 }  
4 void loop() {  
5     digitalWrite(13, HIGH);  
6     delay(1000);  
7     digitalWrite(13, LOW);  
8     delay(1000);  
9 }
```



Задание 2: Подключение красного светодиода к плате arduinoUNO.

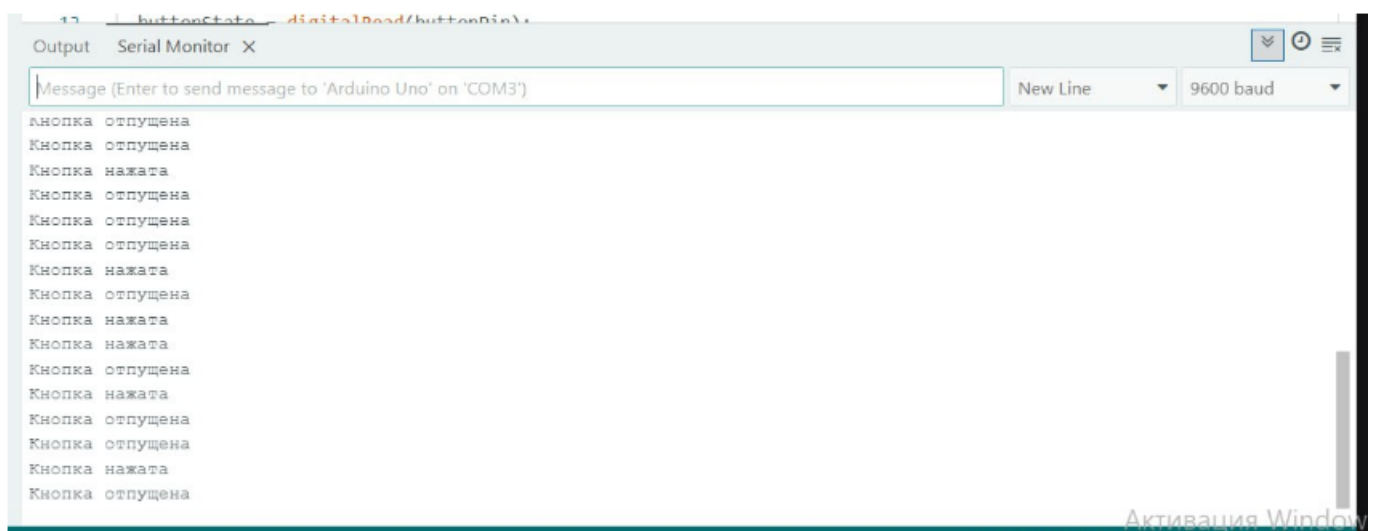
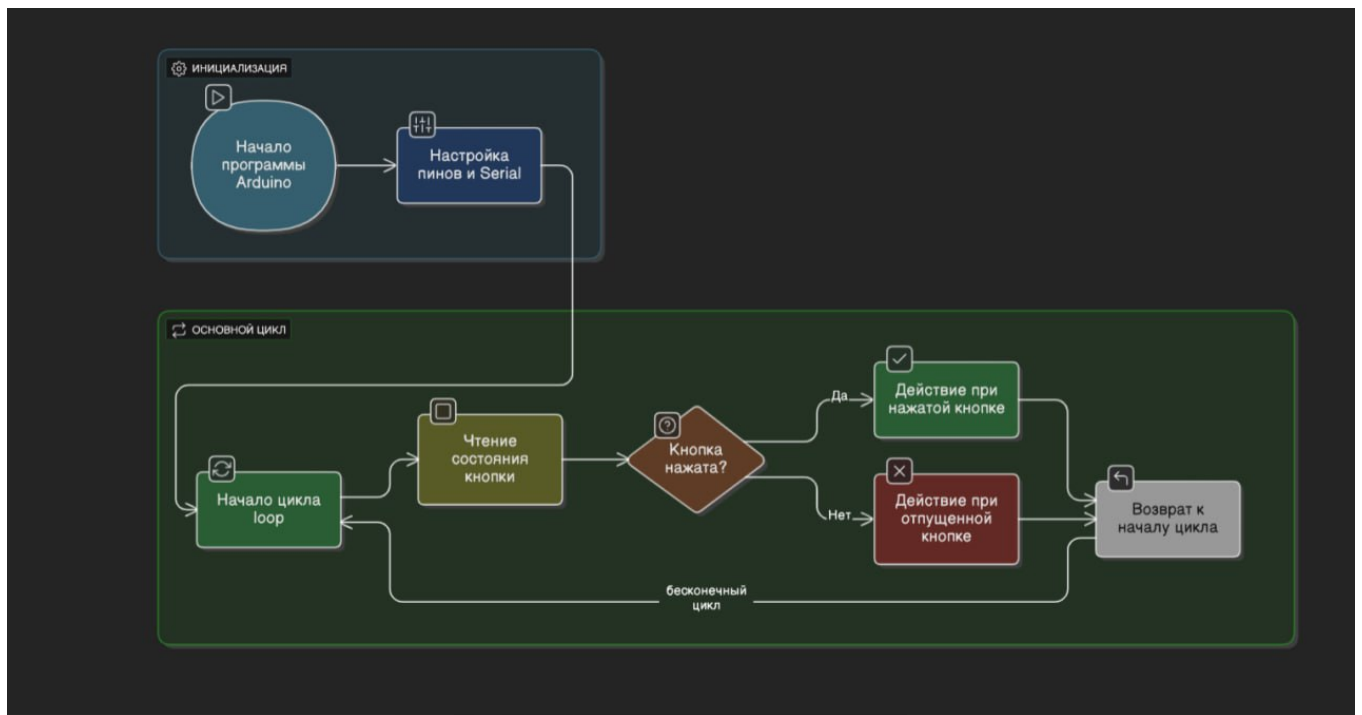
Подключение светодиода через резистор 220 Ом на 13 пин, с использованием скетча программы из 1 пункта.



Задание 3: Подключение кнопки для управления светодиода.

Подключение кнопки к схеме из 2 задания. Светодиод горит только тогда, когда зажата кнопка, иначе светодиод погашен. Дополнительно выводить в консоль надпись нажата ли кнопка или нет.

```
1  const int buttonPin = 10;
2  const int ledPin = 13;
3  int buttonState = 0;
4  void setup() {
5      pinMode(buttonPin, INPUT_PULLUP);
6      pinMode(ledPin, OUTPUT);
7      digitalWrite(ledPin, LOW);
8      Serial.begin(9600);
9  }
10 void loop() {
11     buttonState = digitalRead(buttonPin);
12     if (buttonState == LOW) {
13         digitalWrite(ledPin, HIGH);
14         Serial.println("Кнопка нажата");
15     } else {
16         digitalWrite(ledPin, LOW);
17         Serial.println("Кнопка отпущена");
18         delay(100);
19     }
20 }
```



Задание 4: Подключение кнопки для управления светодиода.

Подключение кнопки к схеме из 2 задания. Когда кнопку нажали один раз, светодиод загорается и горит до тех пор, пока кнопка не будет нажата вновь. Дополнительно выводить в консоль надпись горит ли светодиод или нет.

```

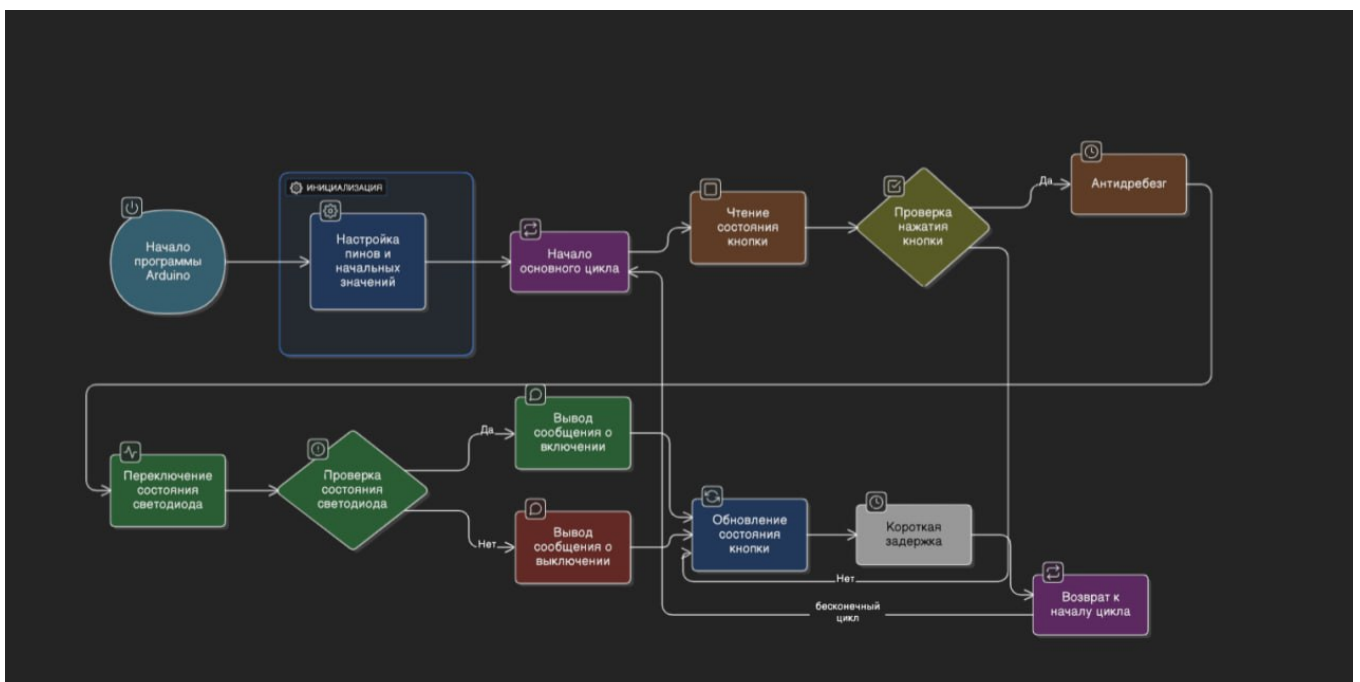
1  const int buttonPin = 10;
2  const int ledPin = 13;
3  bool ledState = false;
4  bool buttonPressed = false;
5  bool lastButtonState = HIGH;
6
7  void setup() {
8      pinMode(buttonPin, INPUT_PULLUP);

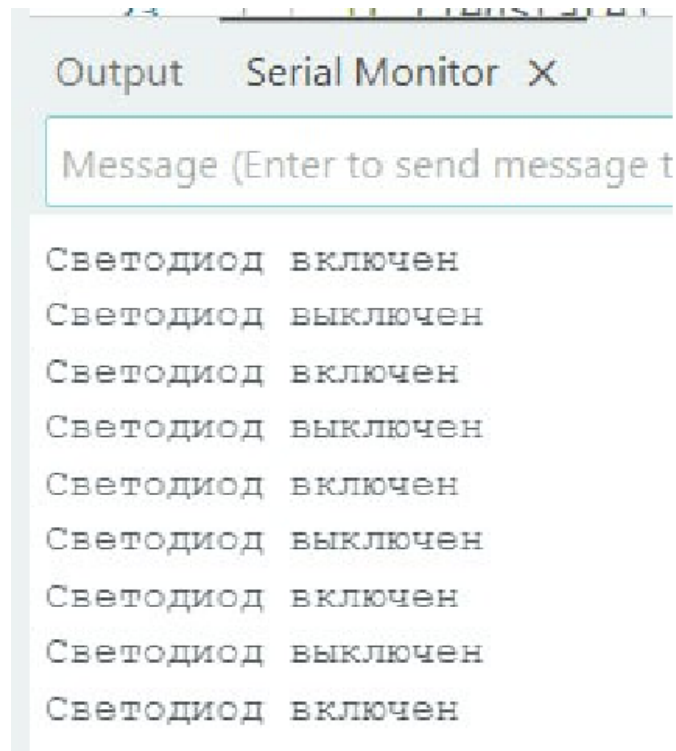
```

```

9     pinMode(ledPin, OUTPUT);
10    digitalWrite(ledPin, LOW);
11    Serial.begin(9600);
12    Serial.println("Светодиод выключен");
13 }
14 void loop()
15     bool currentButtonState = digitalRead(buttonPin);
16
17     if (currentButtonState == LOW && lastButtonState == HIGH) {
18         delay(50);
19         ledstate = !ledstate;
20         digitalWrite(ledPin, ledState ? HIGH : LOW);
21
22         if (ledstate) {
23             Serial.println("Светодиод включен");
24         } else {
25             Serial.println("Светодиод выключен");
26         }
27     }
28     lastButtonState = currentButtonState;
29     delay(10);
30 }
31

```





Задание 5: Подключение потенциометра для управления яркости светодиода.

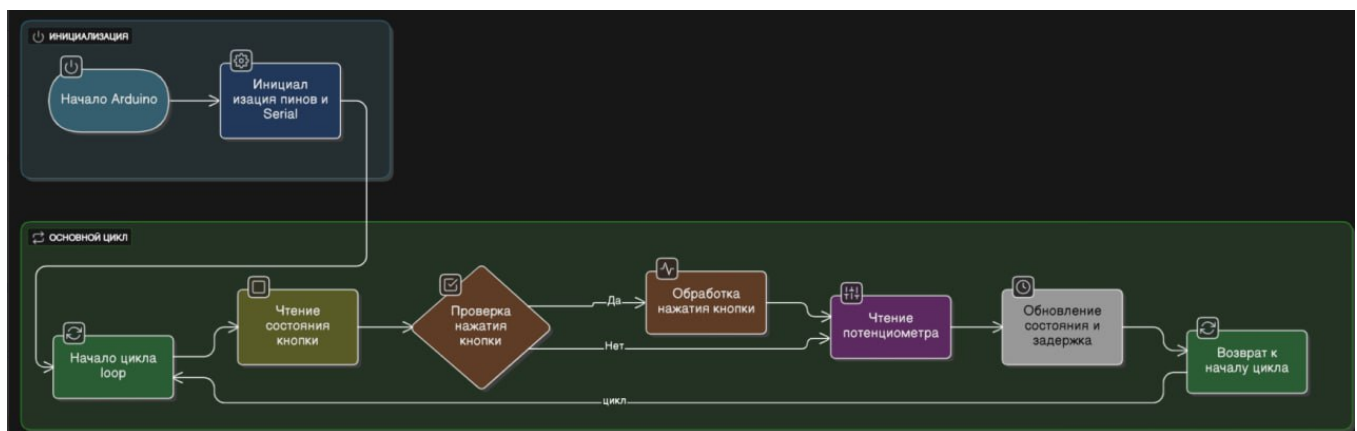
Подключение потенциометра к схеме из 2 задания. Дополнительно вывод значения с потенциометра в консоль.

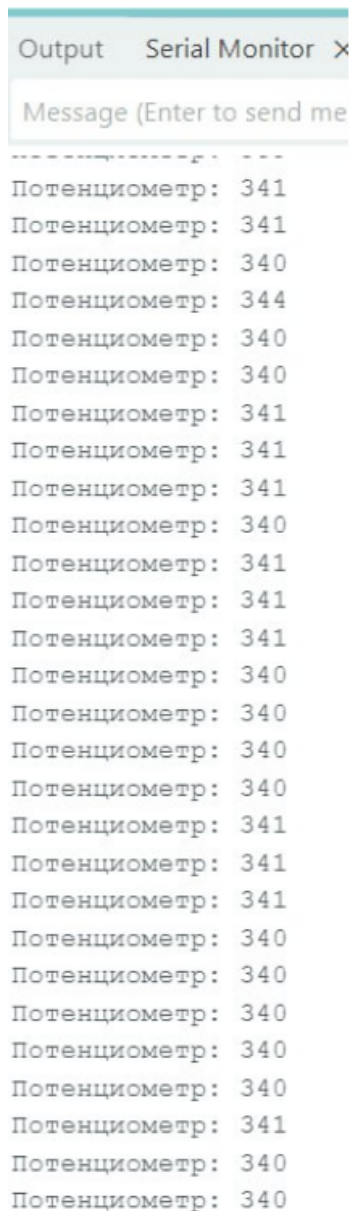
```
1  const int buttonPin = 10;
2  const int ledPin = 13;
3  const int potPin = A0;
4  bool ledState = false;
5  bool lastButtonState = HIGH;
6
7  void setup() {
8      pinMode(buttonPin, INPUT_PULLUP);
9      pinMode(ledPin, OUTPUT);
10     digitalWrite(ledPin, LOW);
11     Serial.begin(9600);
12     Serial.println("Светодиод выключен");
13 }
14
15 void loop() {
16     bool currentButtonState = digitalRead(buttonPin);
17
18     if (currentButtonState == LOW && lastButtonState == HIGH) {
19         delay(50);
```

```

20     ledState = !ledState;
21     digitalWrite(ledPin, ledState ? HIGH : LOW);
22
23     if (ledState) {
24         Serial.println("Светодиод включен");
25     } else {
26         Serial.println("Светодиод выключен");
27     }
28 }
29
30 lastButtonState = currentButtonState;
31
32 int potValue = analogRead(potPin);
33 Serial.print("Потенциометр: ");
34 Serial.println(potValue);
35
36 delay(100);
37 }
38

```





Задание 6: Подключение кнопки к заданию 5.

Задание заключается в изменение яркости светодиода, но при нажатии и удержании кнопки светодиод должен гореть с уровнем яркости, которая была до момента нажатия кнопки, и при удержании кнопки изменить яркость потенциометром было невозможно. Дополнительно вывод значения с потенциометра в консоль, если зажата кнопка, выводить надпись: Изменение яркости невозможно!

```
1 const int buttonPin = 10;
2 const int ledPin = 11;
3 const int potPin = A0;
4
5 int ledBrightness = 0;
6 int savedBrightness = 0;
7 bool buttonPressed = false;
```



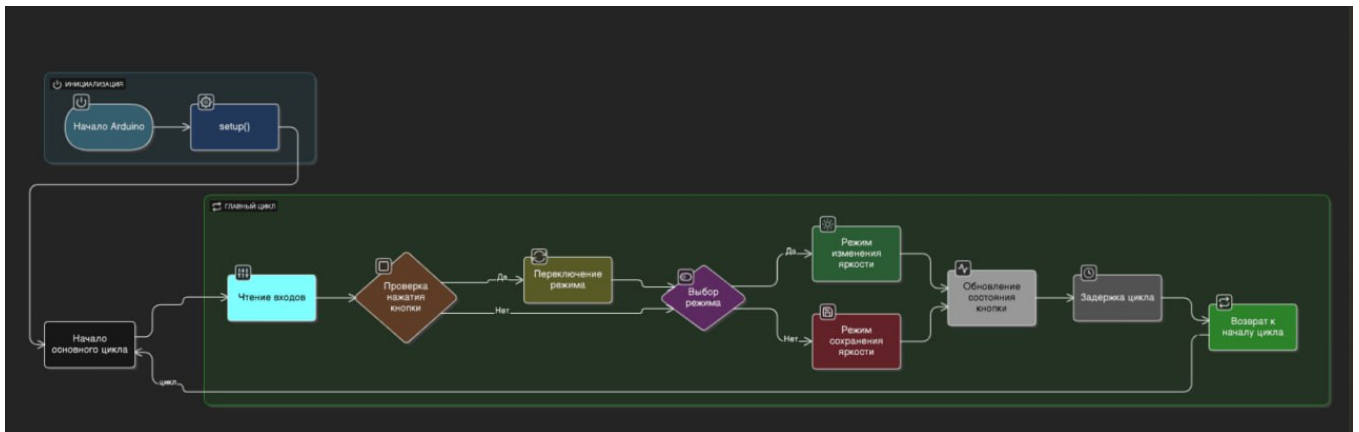
```

8  bool lastButtonState = HIGH;
9
10 void setup() {
11     pinMode(buttonPin, INPUT_PULLUP);
12     pinMode(ledPin, OUTPUT);
13     Serial.begin(9600);
14     analogWrite(ledPin, 0);
15     Serial.println("Светодиод выключен");
16 }
17
18 void loop() {
19     bool currentButtonState = digitalRead(buttonPin);
20     int potValue = analogRead(potPin);
21
22     if (currentButtonState == LOW && lastButtonState == HIGH) {
23         delay(50);
24         buttonPressed = !buttonPressed;
25
26         if (buttonPressed) {
27             savedBrightness = ledBrightness;
28             Serial.println("Изменение яркости невозможно!");
29         } else {
30             Serial.println("Изменение яркости разрешено");
31         }
32     }
33
34     if (!buttonPressed) {
35         ledBrightness = map(potValue, 0, 1023, 0, 255);
36         analogWrite(ledPin, ledBrightness);
37         Serial.print("Яркость: ");
38         Serial.println(ledBrightness);
39     } else {
40         analogWrite(ledPin, savedBrightness);
41     }
42
43     lastButtonState = currentButtonState;
44     delay(100);
45 }

```

Message (Enter to :

яркость: 9
Яркость: 9
Яркость: 10
Яркость: 11
Яркость: 9
Яркость: 9
Яркость: 8
Яркость: 7
Яркость: 8
Яркость: 9
Яркость: 9
Яркость: 9
Яркость: 10
Яркость: 11
Яркость: 10
Яркость: 10
Яркость: 10
Яркость: 11
Яркость: 15
Яркость: 16
Яркость: 16
Яркость: 16
Яркость: 17
Яркость: 16
Яркость: 16
Яркость: 16
Яркость: 16
Яркость: 16
Яркость: 16
Яркость: 16
Яркость: 16
Яркость: 18
Яркость: 19



Задание 7: Светофор на 3 светодиодах различных цветов.

Подключение 3 светодиодов различных цветов. Сделать светофор с переключением цветов, закономерность такая: - Зеленый светодиод горит n-ое количество секунд; - За 3-4 секунды до переключения зеленый светодиод начинает моргать; - Зеленый гаснет, загорается желтый светодиод на 1 секунду; - Красный светодиод горит n-ое количество секунд; - Последнюю секунду красного света одновременно горят красный и желтый светодиод; - Оба светодиода гаснут и загорается зеленый, далее цикл повторяется. Реализовать два варианта работы светофора:

А) выполнить светофор с помощью задержек.

```

1  const int GREEN_PIN = 9;
2  const int YELLOW_PIN = 10;
3  const int RED_PIN = 11;
4
5  const int GREEN_TIME = 10;
6  const int RED_TIME = 8;
7
8  void setup() {
9      pinMode(GREEN_PIN, OUTPUT);
10     pinMode(YELLOW_PIN, OUTPUT);
11     pinMode(RED_PIN, OUTPUT);
12
13     digitalWrite(GREEN_PIN, LOW);
14     digitalWrite(YELLOW_PIN, LOW);
15     digitalWrite(RED_PIN, LOW);
16 }
17
18 void loop() {
19     // Зеленый горит
20     digitalWrite(GREEN_PIN, HIGH);

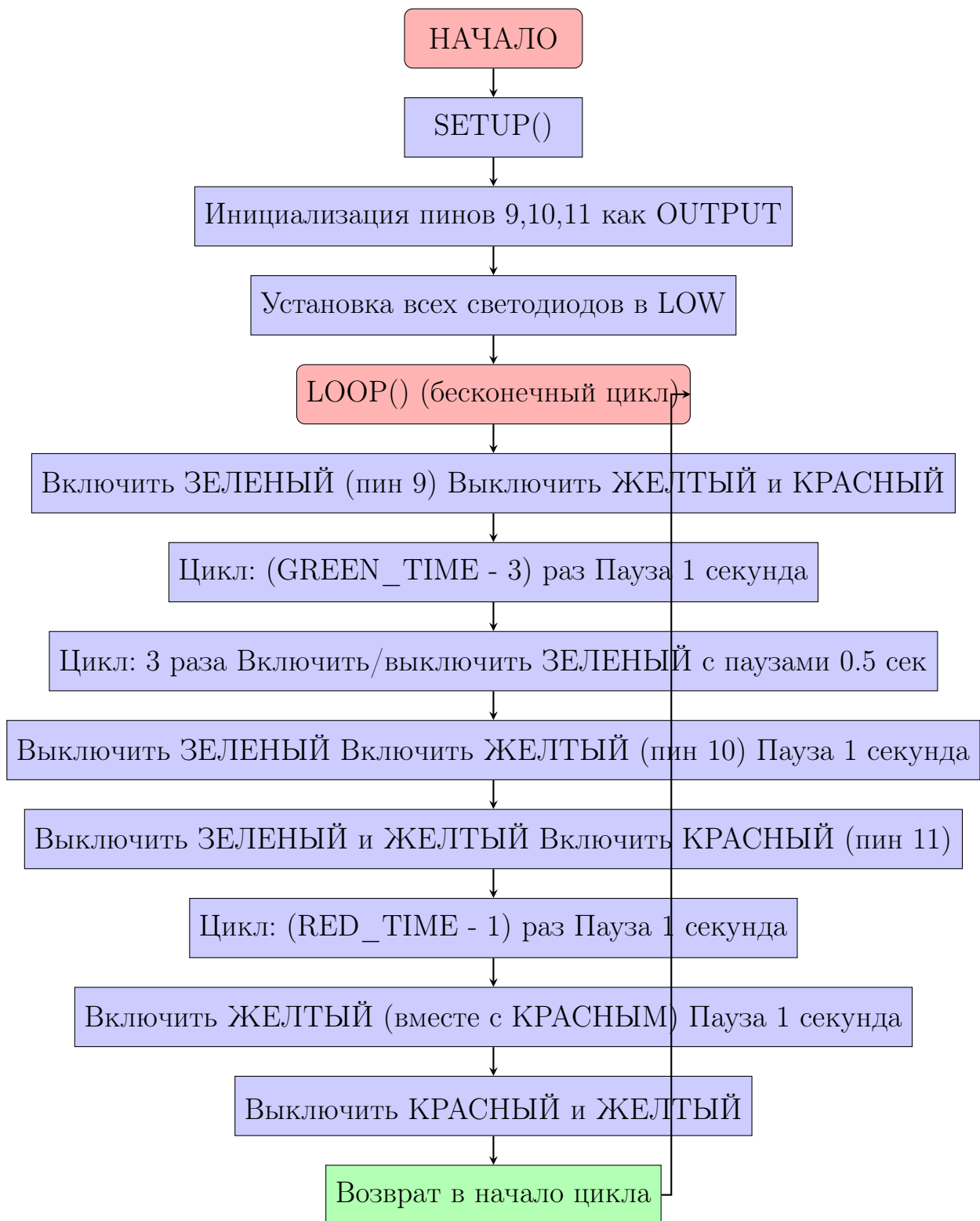
```

```

21 digitalWrite(YELLOW_PIN, LOW);
22 digitalWrite(REDD_PIN, LOW);
23
24 for(int i = GREEN_TIME - 3; i > 0; i--) {
25     delay(1000);
26 }
27
28 // Зеленый мигает
29 for(int i = 3; i > 0; i--) {
30     digitalWrite(GREEN_PIN, HIGH);
31     delay(500);
32     digitalWrite(GREEN_PIN, LOW);
33     delay(500);
34 }
35
36 // Желтый
37 digitalWrite(GREEN_PIN, LOW);
38 digitalWrite(YELLOW_PIN, HIGH);
39 digitalWrite(REDD_PIN, LOW);
40 delay(1000);
41
42 // Красный горит
43 digitalWrite(GREEN_PIN, LOW);
44 digitalWrite(YELLOW_PIN, LOW);
45 digitalWrite(REDD_PIN, HIGH);
46
47 for(int i = RED_TIME - 1; i > 0; i--) {
48     delay(1000);
49 }
50
51 // Красный + Желтый
52 digitalWrite(YELLOW_PIN, HIGH);
53 delay(1000);
54
55 // Переход к зеленому
56 digitalWrite(REDD_PIN, LOW);
57 digitalWrite(YELLOW_PIN, LOW);
58 }
59
60 }

```

Блок-схема работы светофора



Б) выполнить светофор с помощью циклов `for` с выводом оставшихся секунд зеленого и красного света в консоль.

```
1 const int greenLed = 11;  
2 const int yellowLed = 12;
```

```

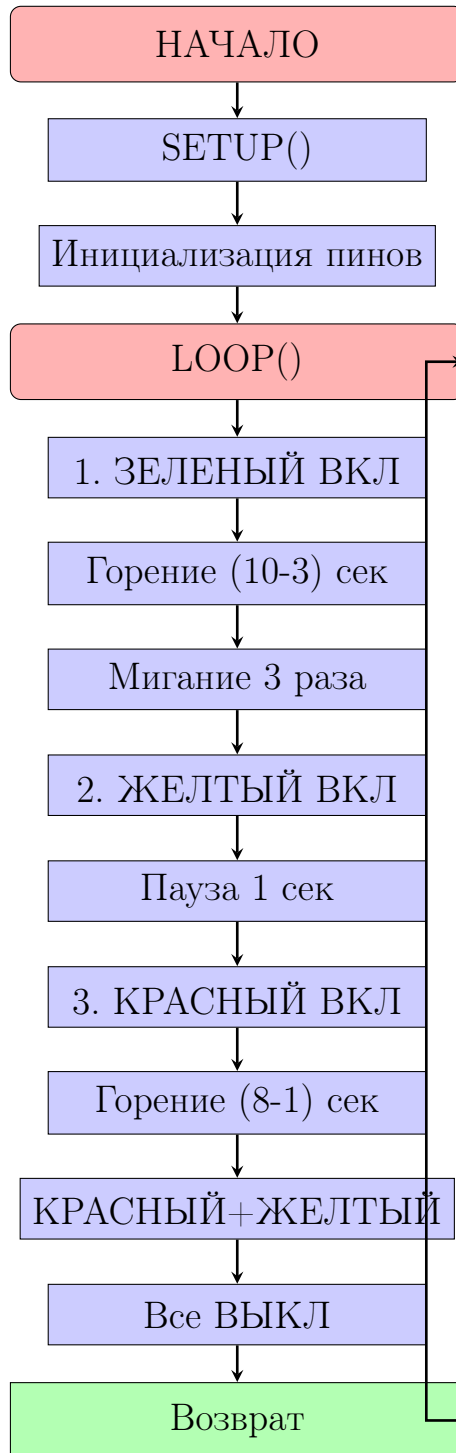
3  const int redLed = 13;
4  const int greenTime = 10;
5  const int redTime = 8;
6
7  void setup() {
8      pinMode(greenLed, OUTPUT);
9      pinMode(yellowLed, OUTPUT);
10     pinMode(redLed, OUTPUT);
11     Serial.begin(9600);
12     digitalWrite(greenLed, LOW);
13     digitalWrite(yellowLed, LOW);
14     digitalWrite(redLed, LOW);
15 }
16
17 void loop() {
18     Serial.println("=== ЗЕЛЕНый ===");
19     digitalWrite(greenLed, HIGH);
20     digitalWrite(yellowLed, LOW);
21     digitalWrite(redLed, LOW);
22
23     for(int i = greenTime; i > 0; i--) {
24         Serial.print("Зеленый: ");
25         Serial.print(i);
26         Serial.println(" сек");
27
28         if(i <= 4) {
29             for(int j = 0; j < 10; j++) {
30                 digitalWrite(greenLed, !digitalRead(greenLed));
31                 delay(100);
32             }
33         } else {
34             delay(1000);
35         }
36     }
37
38     Serial.println("=== ЖЕЛТый ===");
39     digitalWrite(greenLed, LOW);
40     digitalWrite(yellowLed, HIGH);
41
42     for(int i = 1; i > 0; i--) {

```

```

43     Serial.print("Желтый: ");
44     Serial.print(i);
45     Serial.println(" сек");
46     delay(1000);
47 }
48
49 Serial.println("=== КРАСНЫЙ ===");
50 digitalWrite(yellowLed, LOW);
51 digitalWrite(redLed, HIGH);
52
53 for(int i = redTime; i > 0; i--) {
54     Serial.print("Красный: ");
55     Serial.print(i);
56     Serial.println(" сек");
57
58     if(i == 1) {
59         digitalWrite(yellowLed, HIGH);
60         Serial.println("Красный + Желтый");
61     }
62
63     delay(1000);
64 }
65
66 Serial.println("=== ПЕРЕКЛЮЧЕНИЕ ===");
67 digitalWrite(redLed, LOW);
68 digitalWrite(yellowLed, LOW);
69 delay(1000);
70 }
71

```

```

-
Красный: 6 сек
Красный: 5 сек
Красный: 4 сек
Красный: 3 сек
Красный: 2 сек
Красный: 1 сек
Красный + Желтый
=== ПЕРЕКЛЮЧЕНИЕ ===
=== ЗЕЛЕНый ===
Зеленый: 10 сек
Зеленый: 9 сек
Зеленый: 8 сек
Зеленый: 7 сек
Зеленый: 6 сек
Зеленый: 5 сек
Зеленый: 4 сек
Зеленый: 3 сек
Зеленый: 2 сек
Зеленый: 1 сек
=== ЖЕЛТЫЙ ===
Желтый: 1 сек
=== КРАСНЫЙ ===
Красный: 8 сек
Красный: 7 сек
Красный: 6 сек
Красный: 5 сек
Красный: 4 сек
Красный: 3 сек

```

```

.....
Ожидание следующего нажатия кнопки...
Кнопка нажата! Начинаем отсчет 5 секунд..
До начала переключения: 5 сек
До начала переключения: 4 сек
До начала переключения: 3 сек
До начала переключения: 2 сек
До начала переключения: 1 сек
=== НАЧАЛО ЦИКЛА СВЕТОФОРА ===
Зеленый: 7 сек
Зеленый мигает: 3 сек
Зеленый мигает: 2 сек
Зеленый мигает: 1 сек
Желтый: 1 сек
Красный: 5 сек
Красный + Желтый: 1 сек
=== КОНЕЦ ЦИКЛА СВЕТОФОРА ===
Ожидание следующего нажатия кнопки...

```

Задание 8: Сделать светофор по кнопке.

Всегда горит зеленый свет светофора, после нажатия кнопки через 5 секунд светофор начинает переключаться, после переключения обратно на зеленый свет ожидание следующего нажатия кнопки.

```

1 const int buttonPin = 10;
2 const int greenLed = 11;
3 const int yellowLed = 12;
4 const int redLed = 13;
5

```

```

6  bool buttonPressed = false;
7  bool lastButtonState = HIGH;
8  bool trafficLightActive = false;
9
10 void setup() {
11     pinMode(buttonPin, INPUT_PULLUP);
12     pinMode(greenLed, OUTPUT);
13     pinMode(yellowLed, OUTPUT);
14     pinMode(redLed, OUTPUT);
15     Serial.begin(9600);
16
17     digitalWrite(greenLed, HIGH);
18     digitalWrite(yellowLed, LOW);
19     digitalWrite(redLed, LOW);
20     Serial.println("Ожидание нажатия кнопки...");
21 }
22
23 void loop() {
24     bool currentButtonState = digitalRead(buttonPin);
25
26     if (currentButtonState == LOW && lastButtonState == HIGH &&
27         ↪ !trafficLightActive) {
28         delay(50);
29         buttonPressed = true;
30         Serial.println("Кнопка нажата! Начинаем отсчет 5 секунд...");
31     }
32
33     if (buttonPressed && !trafficLightActive) {
34         for(int i = 5; i > 0; i--) {
35             Serial.print("До начала переключения: ");
36             Serial.print(i);
37             Serial.println(" сек");
38             delay(1000);
39         }
40
41         trafficLightActive = true;
42         buttonPressed = false;
43         runTrafficLightCycle();
44     }

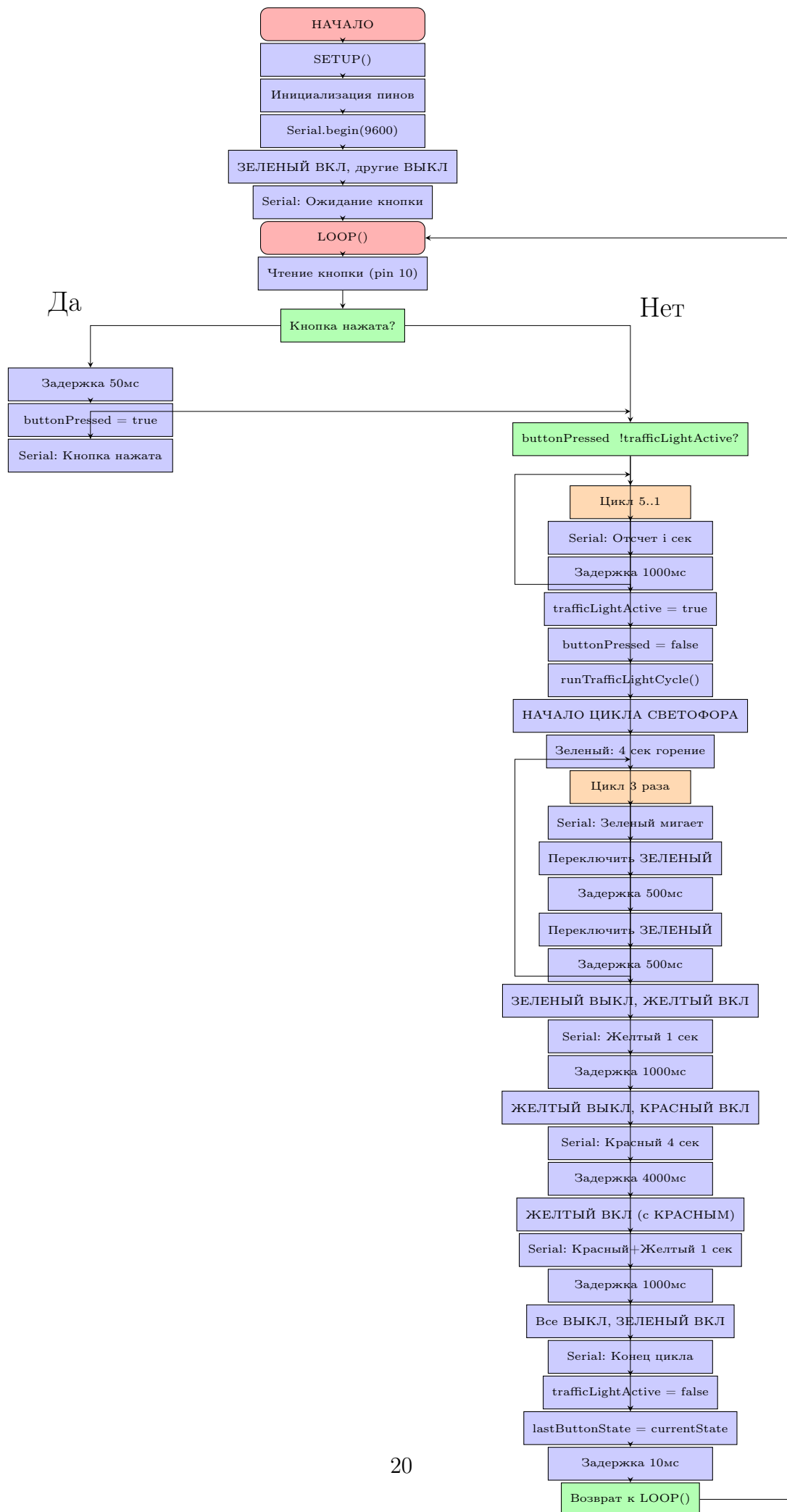
```

```

45     lastButtonState = currentButtonState;
46     delay(10);
47 }
48
49 void runTrafficLightCycle() {
50     Serial.println("=== НАЧАЛО ЦИКЛА СВЕТОФОРА ===");
51
52     // Зеленый мигает последние 3 секунды
53     Serial.println("Зеленый: 7 сек");
54     delay(4000);
55
56     for(int i = 3; i > 0; i--) {
57         Serial.print("Зеленый мигает: ");
58         Serial.print(i);
59         Serial.println(" сек");
60         digitalWrite(greenLed, !digitalRead(greenLed));
61         delay(500);
62         digitalWrite(greenLed, !digitalRead(greenLed));
63         delay(500);
64     }
65
66     // Выключаем зеленый, включаем желтый
67     digitalWrite(greenLed, LOW);
68     digitalWrite(yellowLed, HIGH);
69     Serial.println("Желтый: 1 сек");
70     delay(1000);
71
72     // Выключаем желтый, включаем красный
73     digitalWrite(yellowLed, LOW);
74     digitalWrite(redLed, HIGH);
75     Serial.println("Красный: 5 сек");
76     delay(4000);
77
78     // Последняя секунда красного + желтый
79     digitalWrite(yellowLed, HIGH);
80     Serial.println("Красный + Желтый: 1 сек");
81     delay(1000);
82
83     // Выключаем все, включаем зеленый
84     digitalWrite(redLed, LOW);

```

```
85 digitalWrite(yellowLed, LOW);
86 digitalWrite(greenLed, HIGH);
87
88 Serial.println("=== КОНЕЦ ЦИКЛА СВЕТОФОРА ===");
89 Serial.println("Ожидание следующего нажатия кнопки...");
90
91 trafficLightActive = false;
92 }
93
```



Задание 9: Подключение RGB светодиода.

Сделать светофор на нем, определить какие цвета нужно смешать, чтобы получить желтый сигнал светофора, так как на RGB три цвета: Red Green Blue. Не получится реализовать одновременное горение красного с желтым.

```
1  const int buttonPin = 2;
2  const int redPin = 9;
3  const int greenPin = 10;
4  const int bluePin = 11;
5
6  bool buttonPressed = false;
7  bool lastButtonState = HIGH;
8  bool trafficLightActive = false;
9
10 void setup() {
11     pinMode(buttonPin, INPUT_PULLUP);
12     pinMode(redPin, OUTPUT);
13     pinMode(greenPin, OUTPUT);
14     pinMode(bluePin, OUTPUT);
15     Serial.begin(9600);
16
17     setGreen();
18     Serial.println("Ожидание нажатия кнопки...");
19 }
20
21 void loop() {
22     bool currentButtonState = digitalRead(buttonPin);
23
24     if (currentButtonState == LOW && lastButtonState == HIGH &&
25         ↪ !trafficLightActive) {
26         delay(50);
27         buttonPressed = true;
28         Serial.println("Кнопка нажата! Начинаем отсчет 5 секунд...");
29     }
30
31     if (buttonPressed && !trafficLightActive) {
32         for(int i = 5; i > 0; i--) {
33             Serial.print("До начала переключения: ");
34             Serial.print(i);
35             Serial.println(" сек");
36             delay(1000);
37         }
38     }
39 }
```



```

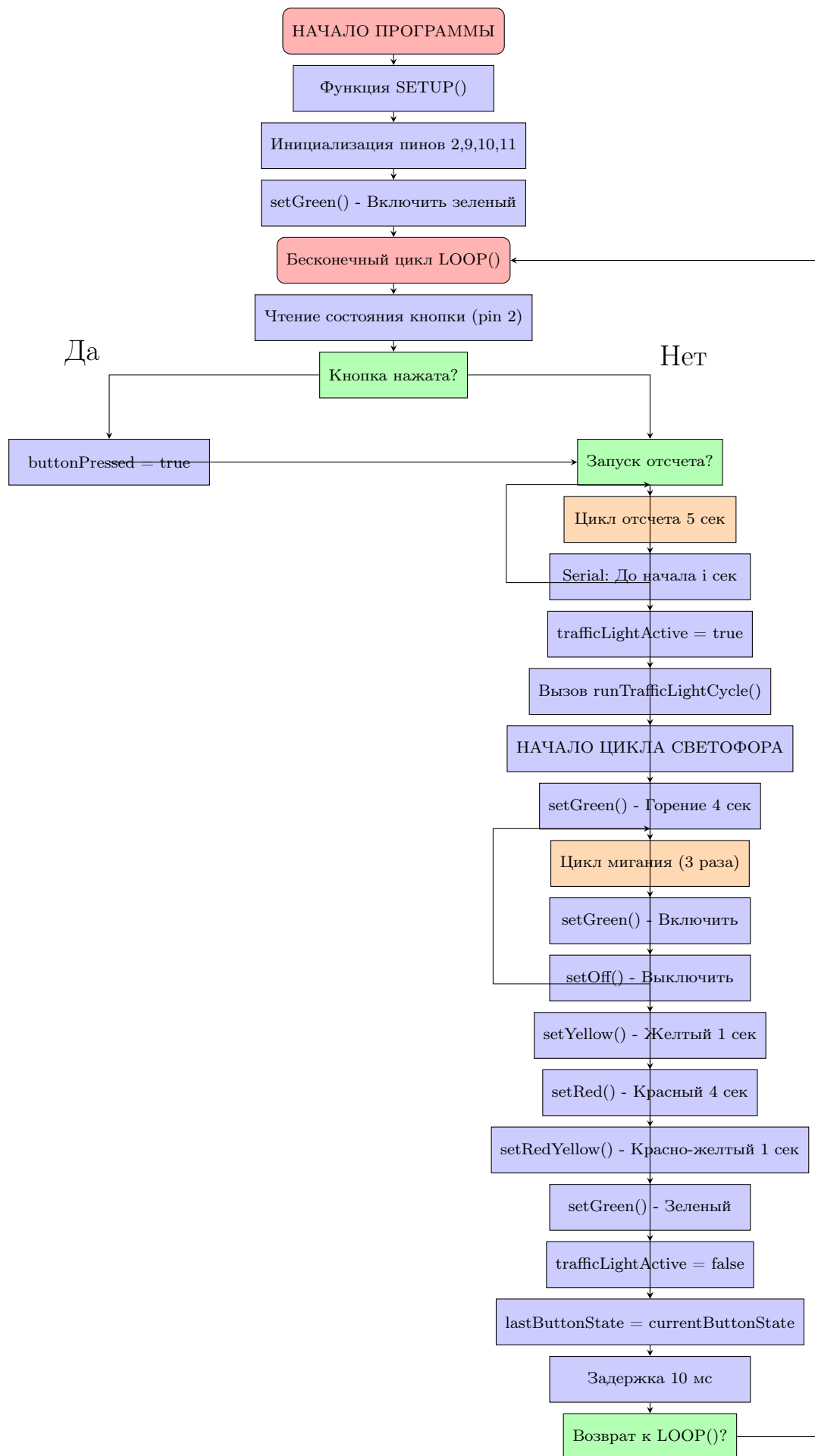
36     }
37
38     trafficLightActive = true;
39     buttonPressed = false;
40     runTrafficLightCycle();
41 }
42
43 lastButtonState = currentButtonState;
44 delay(10);
45 }
46
47 void setGreen() {
48     analogWrite(redPin, 0);
49     analogWrite(greenPin, 255);
50     analogWrite(bluePin, 0);
51 }
52
53 void setYellow() {
54     // Желтый = Красный (255) + Зеленый (150)
55     analogWrite(redPin, 255);
56     analogWrite(greenPin, 150);
57     analogWrite(bluePin, 0);
58 }
59
60 void setRed() {
61     analogWrite(redPin, 255);
62     analogWrite(greenPin, 0);
63     analogWrite(bluePin, 0);
64 }
65
66 void setRedYellow() {
67     // Красный + Желтый = Красный (255) + Зеленый (75) - более
68     ↪ оранжевый оттенок
69     analogWrite(redPin, 255);
70     analogWrite(greenPin, 75);
71     analogWrite(bluePin, 0);
72 }
73
74 void setOff() {
75     analogWrite(redPin, 0);

```

```

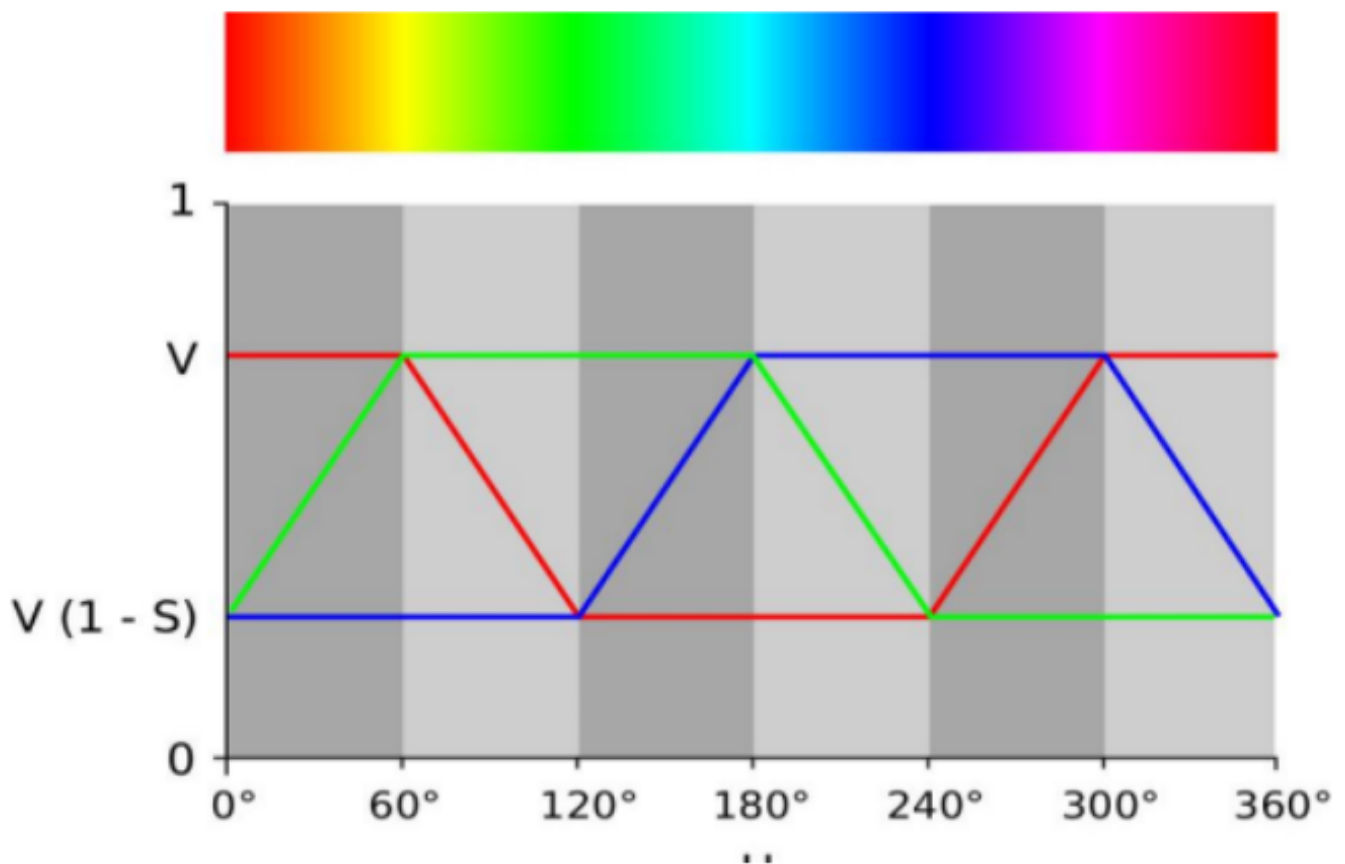
75   analogWrite(greenPin, 0);
76   analogWrite(bluePin, 0);
77 }
78
79 void runTrafficLightCycle() {
80     Serial.println("=== НАЧАЛО ЦИКЛА СВЕТОФОРА ===");
81
82     // Зеленый горит 7 секунд
83     setGreen();
84     Serial.println("Зеленый: 7 сек");
85     delay(4000);
86
87     // Зеленый мигает 3 секунды
88     for(int i = 3; i > 0; i--) {
89         Serial.print("Зеленый мигает: ");
90         Serial.print(i);
91         Serial.println(" сек");
92         setGreen();
93         delay(500);
94         setOff();
95         delay(500);
96     }
97
98     setYellow();
99     Serial.println("Желтый: 1 сек");
100    delay(1000);
101
102    setRed();
103    Serial.println("Красный: 5 сек");
104    delay(4000);
105
106    setRedYellow();
107    Serial.println("Красный + Желтый: 1 сек");
108    delay(1000);
109
110    setGreen();
111    Serial.println("=== КОНЕЦ ЦИКЛА СВЕТОФОРА ===");
112    Serial.println("Ожидание следующего нажатия кнопки...");
113
114    trafficLightActive = false;}

```



Задание 10: Подключение к 9 заданию потенциометра.

Реализовать плавную смену цветов по диаграмме представленной на рисунке.



```
1  const int R_PIN = 9;
2  const int G_PIN = 10;
3  const int B_PIN = 11;
4  const int POT_PIN = A0;
5
6  int hue = 0;  // 0..359
7
8  void setRGB(int r, int g, int b) {
9      analogWrite(R_PIN, r);
10     analogWrite(G_PIN, g);
11     analogWrite(B_PIN, b);
12 }
13
14 void hsvToRgb(int hue, int &red, int &green, int &blue) {
15     float huePosition = hue / 60.0;
16     int sector = (int)floor(huePosition) % 6;
17     float fraction = huePosition - sector;
18
19     int value = 255;  // Фиксированная максимальная яркость
```

```

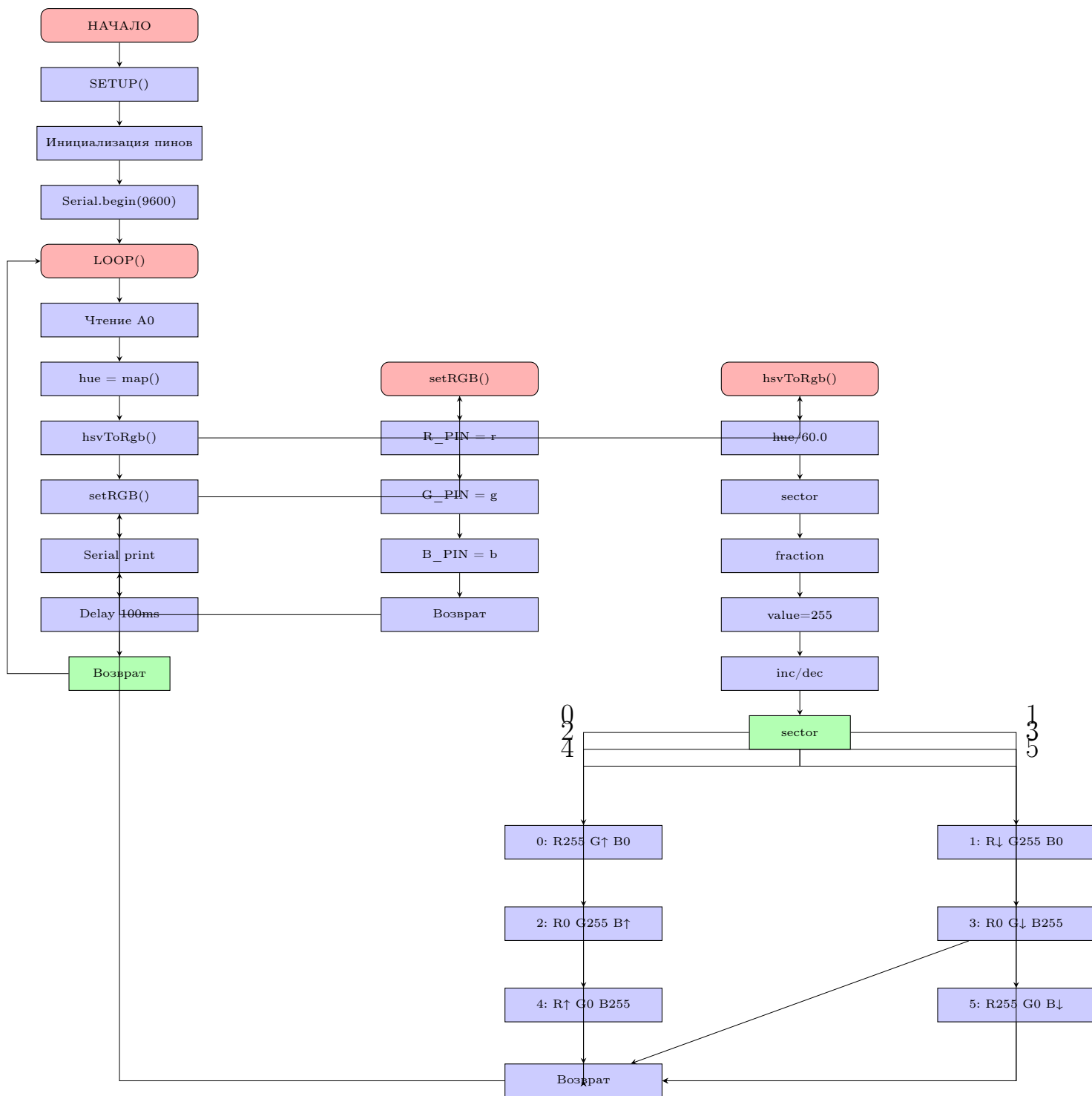
20  int decreasing = (int)(value * (1 - fraction));
21  int increasing = (int)(value * fraction);
22
23  switch (sector) {
24      case 0: red = value;          green = increasing;  blue = 0;
25              ↪ break;
26      case 1: red = decreasing; green = value;          blue = 0;
27              ↪ break;
28      case 2: red = 0;              green = value;          blue = increasing;
29              ↪ break;
30      case 3: red = 0;              green = decreasing; blue = value;
31              ↪ break;
32      case 4: red = increasing; green = 0;              blue = value;
33              ↪ break;
34      case 5: red = value;          green = 0;              blue = decreasing;
35              ↪ break;
36  }
37
38  }
39
40  void setup() {
41      pinMode(R_PIN, OUTPUT);
42      pinMode(G_PIN, OUTPUT);
43      pinMode(B_PIN, OUTPUT);
44      Serial.begin(9600);
45  }
46
47  void loop() {
48      int potValue = analogRead(POT_PIN);
49      hue = map(potValue, 0, 1023, 0, 359);
50
51      int r, g, b;
52      hsvToRgb(hue, r, g, b);
53      setRGB(r, g, b);
54
55      Serial.print("Hue=");
56      Serial.print(hue);
57      Serial.print(" | RGB=");
58      Serial.print(r);
59      Serial.print(",");
60      Serial.print(g);
61      Serial.print(",");
62      Serial.print(b);
63      Serial.print("\n");
64  }

```

```

55 Serial.print(b);
56 Serial.println(" ");
57
58 delay(100);
59 }
60

```



```

Hue=359 | RGB=(255,0,4)
Hue=359 | RGB=(255,0,4)
Hue=359 | RGB=(255,0,4)
Hue=359 | RGB=(255,0,4)
Hue=359 | RGB=(255,0,4)
Hue=359 | RGB=(255,0,4)
Hue=359 | RGB=(255,0,4)
Hue=318 | RGB=(255,0,178)
Hue=262 | RGB=(93,0,255)
Hue=212 | RGB=(0,119,255)
Hue=161 | RGB=(0,255,174)
Hue=120 | RGB=(0,255,0)
Hue=80 | RGB=(169,255,0)
Hue=46 | RGB=(255,195,0)
Hue=16 | RGB=(255,68,0)
Hue=3 | RGB=(255,12,0)
Hue=3 | RGB=(255,12,0)
Hue=3 | RGB=(255,12,0)

```

Задание 11: Подключение RCB светодиода и энкодера.

Управление цветами как в 10 задании. При нажатии и удержании кнопки энкодера, при дальнейшем вращении изменять яркость цвета который был до нажатия на кнопку

```

1  const int R_PIN = 9;
2  const int G_PIN = 10;
3  const int B_PIN = 11;
4
5  const int ENCODER_PIN_A = 2;
6  const int ENCODER_PIN_B = 3;
7  const int BUTTON_PIN = 4;
8
9  int hue = 0;           // 0..359
10 int brightness = 255;  // 0..255 (Value)
11
12 // Для энкодера
13 int lastA = HIGH;
14 int lastB = HIGH;

```



```

15
16 void setRGB(int r, int g, int b) {
17     analogWrite(R_PIN, r);
18     analogWrite(G_PIN, g);
19     analogWrite(B_PIN, b);
20 }
21
22 void hsvToRgb(int hue, int brightness, int &red, int &green, int
    ↪ &blue) {
23     // Делим оттенок на 60°, чтобы понять сектор (0..5)
24     float huePosition = hue / 60.0;
25     int sector = (int)floor(huePosition) % 6;
26     float fraction = huePosition - sector; // дробная часть для
    ↪ плавного перехода
27
28     // Промежуточные значения для плавного перехода между секторами
29     int value = brightness; // максимальное значение (V)
30     int decreasing = (int)(value * (1 - fraction)); // спадающий
    ↪ канал
31     int increasing = (int)(value * fraction); // возрастающий
    ↪ канал
32
33     switch (sector) {
34         case 0: red = value;          green = increasing; blue = 0;
    ↪ break;
35         case 1: red = decreasing; green = value;          blue = 0;
    ↪ break;
36         case 2: red = 0;              green = value;      blue = increasing;
    ↪ break;
37         case 3: red = 0;              green = decreasing; blue = value;
    ↪ break;
38         case 4: red = increasing; green = 0;              blue = value;
    ↪ break;
39         case 5: red = value;          green = 0;          blue = decreasing;
    ↪ break;
40     }
41 }
42
43
44
45 void setup() {

```

```

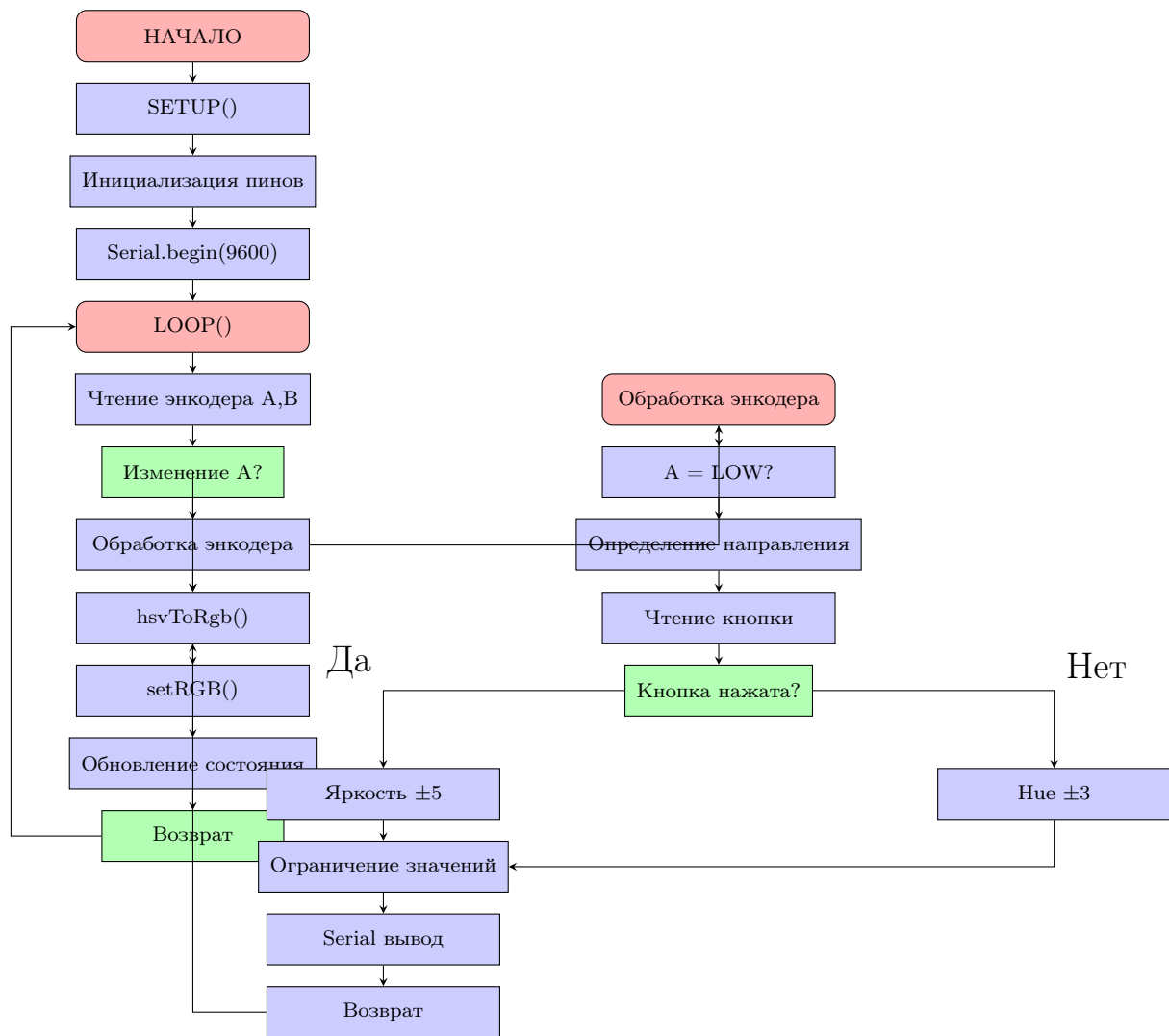
46 pinMode(R_PIN, OUTPUT);
47 pinMode(G_PIN, OUTPUT);
48 pinMode(B_PIN, OUTPUT);
49
50 pinMode(ENCODER_PIN_A, INPUT_PULLUP);
51 pinMode(ENCODER_PIN_B, INPUT_PULLUP);
52 pinMode(BUTTON_PIN, INPUT_PULLUP);
53
54 lastA = digitalRead(ENCODER_PIN_A);
55 lastB = digitalRead(ENCODER_PIN_B);
56
57 Serial.begin(9600);
58 }
59
60 void loop() {
61     int A = digitalRead(ENCODER_PIN_A);
62     int B = digitalRead(ENCODER_PIN_B);
63     if (A != lastA) { // Изменился канал A → шаг энкодера
64         if (A == LOW) {
65             // Определяем направление по каналу B
66             if (B == HIGH) {
67                 encoderTurn(+1); // вправо
68             } else {
69                 encoderTurn(-1); // влево
70             }
71         }
72     }
73
74     lastA = A;
75     lastB = B;
76
77     int r, g, b;
78     hsvToRgb(hue, brightness, r, g, b);
79     setRGB(r, g, b);
80 }
81
82 // Обработка шага энкодера
83 void encoderTurn(int dir) {
84     int buttonState = digitalRead(BUTTON_PIN);
85

```

```

86  if (buttonState == LOW) {
87      // Кнопка нажата → меняем яркость
88      brightness += dir * 5;
89      if (brightness < 0) brightness = 0;
90      if (brightness > 255) brightness = 255;
91
92      Serial.print("Поворот: ");
93      Serial.print(dir > 0 ? "вправо" : "влево");
94      Serial.print(" | Режим: ЯРКОСТЬ | Яркость=");
95      Serial.println(brightness);
96
97  } else {
98      // Кнопка не нажата → меняем цвет (Hue)
99      hue += dir * 3;
100     if (hue < 0) hue += 360;
101     if (hue >= 360) hue -= 360;
102
103     Serial.print("Поворот: ");
104     Serial.print(dir > 0 ? "вправо" : "влево");
105     Serial.print(" | Режим: ЦВЕТ | Hue=");
106     Serial.println(hue);
107 }
108 }

```



```

Поворот: вправо | Режим: ЯРКОСТЬ | Яркость=255
Поворот: вправо | Режим: ЯРКОСТЬ | Яркость=255
Поворот: вправо | Режим: ЯРКОСТЬ | Яркость=255
Поворот: влево | Режим: ЯРКОСТЬ | Яркость=250
Поворот: влево | Режим: ЯРКОСТЬ | Яркость=245
Поворот: влево | Режим: ЯРКОСТЬ | Яркость=240
Поворот: влево | Режим: ЯРКОСТЬ | Яркость=235
Поворот: влево | Режим: ЯРКОСТЬ | Яркость=230
Поворот: влево | Режим: ЯРКОСТЬ | Яркость=225
Поворот: влево | Режим: ЯРКОСТЬ | Яркость=220
Поворот: влево | Режим: ЯРКОСТЬ | Яркость=215
Поворот: влево | Режим: ЯРКОСТЬ | Яркость=210
Поворот: влево | Режим: ЦВЕТ | Hue=327
Поворот: влево | Режим: ЯРКОСТЬ | Яркость=205
Поворот: влево | Режим: ЦВЕТ | Hue=324
Поворот: влево | Режим: ЦВЕТ | Hue=321
Поворот: влево | Режим: ЦВЕТ | Hue=318
Поворот: влево | Режим: ЦВЕТ | Hue=315
Поворот: влево | Режим: ЦВЕТ | Hue=312
  
```

