

Annex 1: First project code

```
1 //Pin numbers for LEDs:
2 const int ledBlue = 2;
3 const int ledRed = 3;
4 const int ledGreen = 4;
5 const int ledYellow=5;
6
7 //Pin numbers for PushButton:
8 const int buttonBlue = 13;
9 const int buttonRed = 12;
10 const int buttonGreen = 8;
11
12
13 // Constants used to define duration of game parameters
14 int inGame = 0; //inGame parameter to start running the game
15 const int timeHit = 200; //Max time to hit
16 int timeOutHit = 0; //time out parameter
17
18 int randomLed = 2; //Starting randomLed light to turn on
19 int numberHits=0; //parameter used to count number of hits
20
21
22 int startGame = LOW;
23
24 void setup() {
25     Serial.begin(9600);
26     //initialize the LED pin as OUTPUTS:
27     pinMode(ledBlue, OUTPUT);
28     pinMode(ledRed, OUTPUT);
29     pinMode(ledGreen, OUTPUT);
30
31     //initialize the pushbuttons as INPUTS:
32     pinMode(buttonBlue, INPUT);
33     pinMode(buttonRed, INPUT);
34     pinMode(buttonGreen, INPUT);
35     //pinMode(startButton, INPUT);
36 }
37 void lightSequence() {
38     Serial.println(randomLed);
39     Serial.println(timeOutHit);
40     digitalWrite(randomLed, HIGH); //Turn on random selected light
41
42     int BlueButtonRead = digitalRead(13);
43     int RedButtonRead = digitalRead(12);
44     int GreenButtonRead = digitalRead(8);
45
46     if (timeOutHit <= timeHit) {
47         timeOutHit += 1;
48         if (digitalRead(ledBlue) == HIGH) {
49             if (BlueButtonRead == HIGH) {
50                 Serial.println("Blue hit"); //Print state of BlueButton
51                 delay(200);
52                 digitalWrite(randomLed, LOW); //Turn off the random selected light after pushbutton hit
53                 delay(100);
54                 timeOutHit = 0;
55                 numberHits += 1;
56
57                 randomLed = random(2, 5); // pick random light to turn on
58                 Serial.println(randomLed); // print the pins led randomly chosen light
59                 lightSequence();
60             }
61         }
```

```

62
63   if (digitalRead(ledRed) == HIGH) {
64
65       if (RedButtonRead == HIGH) {
66           Serial.println("Red hit");           //Print state of BlueButton
67           delay(200);
68           digitalWrite(randomLed, LOW);      //Turn off the random selected light after pushbutton hit
69           delay(100);
70           timeOutHit = 0;
71           numberHits += 1;
72
73           randomLed = random(2, 5);          // pick random light to turn on
74           Serial.println(randomLed);         // print the pins led randomly chosen light
75           lightSequence();
76       }
77   }
78
79   if (digitalRead(ledGreen) == HIGH) {
80
81       if (GreenButtonRead == HIGH) {
82           Serial.println("Green hit");        //Print state of BlueButton
83           delay(200);
84           digitalWrite(randomLed, LOW);      //Turn off the random selected light afeter pushbutton hit
85           delay(100);
86           timeOutHit = 0;
87           numberHits += 1;
88
89           randomLed = random(2, 5);          // pick random light to turn on
90           Serial.println(randomLed);         // print the pins led randomly chosen light
91           lightSequence();
92       }
93   }
94   }
95   }
96   if (timeOutHit >= timeHit) {               //If no hit is detected in TimeOutHit seconds another random light is selected
97       digitalWrite(randomLed, LOW);
98       delay(100);
99       timeOutHit = 0;
100      randomLed = random(2, 5);
101      lightSequence();
102   }
103 }
104 }
105 void loop() {
106     int StartButtonRead = digitalRead(6);
107     Serial.println(StartButtonRead);
108     if ( inGame == 0) {
109
110         if (StartButtonRead == HIGH) {       //If StartButton is hit, the Game starts
111             digitalWrite(5, HIGH);
112             Serial.println("Game Started");
113             delay(1000);
114             lightSequence();
115             inGame = 1;
116         }
117     }
118 }
119
120 else {                                       //Else used to count how long the game is running
121     inGame = inGame + 1;
122     lightSequence();
123 }
124
125 if ( inGame >= 5000) {
126     Serial.println("Game Finished");
127     Serial.println("You have hit:");
128     Serial.println(numberHits);
129     inGame = 0;
130     delay(5000);
131 }
132 }

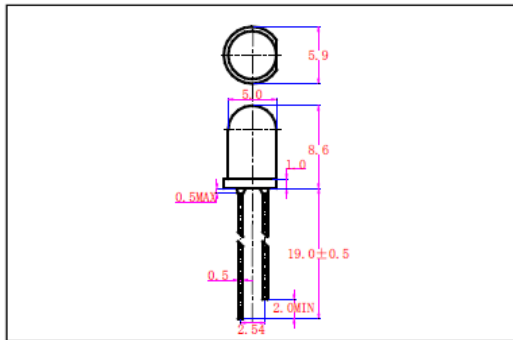
```

Annex 2: LEDs Datasheets

• LED Red:

产品型号 (Part number system for led lamp) UR502DC (Φ 5MM 短脚超亮红四元)

外形图 (Package Dimensions) 单位: (Unit): mm



晶片 (CHIP)	
材质 (Material)	InGaAlP
颜色 (Color)	红色
胶体 (Colloid)	
材质 (Material)	环氧树脂
颜色 (Color)	红色扩散

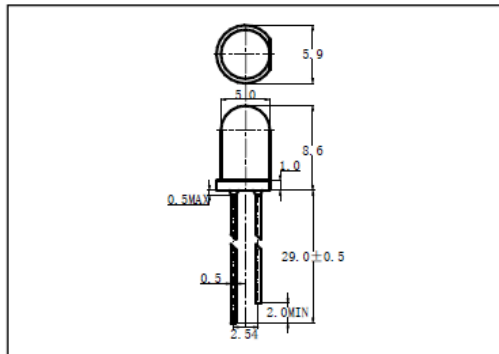
极限参数 (Absolute Maximum Ratings) (Ta=25℃)

项目参数 (Parameter)	符号 (Symbol)	数值	单位 (Unit)
最大功耗 (Max Power Dissipation)	P _M	80	mW
最大正向电流 (Max Continuous Forward Current)	I _{FM}	30	mA
最大反向电压 (Max Reverse Voltage)	V _{RM}	5	V
最大脉冲峰值电流 (Peak Forward Current)	I _{FP}	75	mA
焊接温度/时间 (Lead Soldering Temperature/Time)	T _{SOL}	240/≤3S	℃/S
工作环境 (Operating Temperature Range)	T _{OPR}	-25~+85	℃
储存温度 (Storage Temperature Range)	T _{STR}	-30~+100	℃

• LED Blue:

产品型号 (Part number system for led lamp) (Φ 5MM 长脚蓝发蓝)

外形图 (Package Dimensions) 单位: (Unit): mm



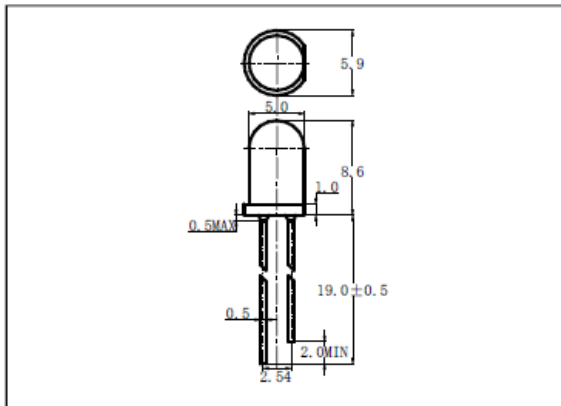
晶片 (CHIP)	
材质 (Material)	GaP
颜色 (Color)	蓝色
胶体 (Colloid)	
材质 (Material)	环氧树脂
颜色 (Color)	蓝色

极限参数 (Absolute Maximum Ratings) (Ta=25℃)

项目参数 (Parameter)	符号 (Symbol)	数值	单位 (Unit)
最大功耗 (Max Power Dissipation)	P _M	80	mW
最大正向电流 (Max Continuous Forward Current)	I _{FM}	20	mA
最大反向电压 (Max Reverse Voltage)	V _{RM}	5	V
最大脉冲峰值电流 (Peak Forward Current)	I _{FP}	75	mA
焊接温度/时间 (Lead Soldering Temperature/Time)	T _{SOL}	240/≤3S	℃/S
工作环境 (Operating Temperature Range)	T _{OPR}	-25~+85	℃
储存温度 (Storage Temperature Range)	T _{STR}	-30~+100	℃

- LED Green:

外形图 (Package Dimensions) 单位: (Unit): mm



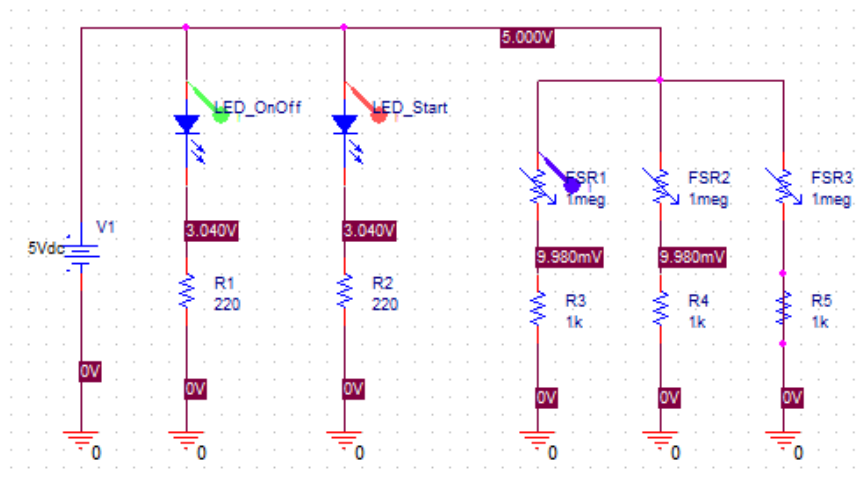
晶片 (CHIP)	
材质 (Material)	GaP
颜色 (Color)	绿色
胶体 (Colloid)	
材质 (Material)	环氧树脂
颜色 (Color)	绿色扩散

极限参数 (Absolute Maximum Ratings) (Ta=25°C)

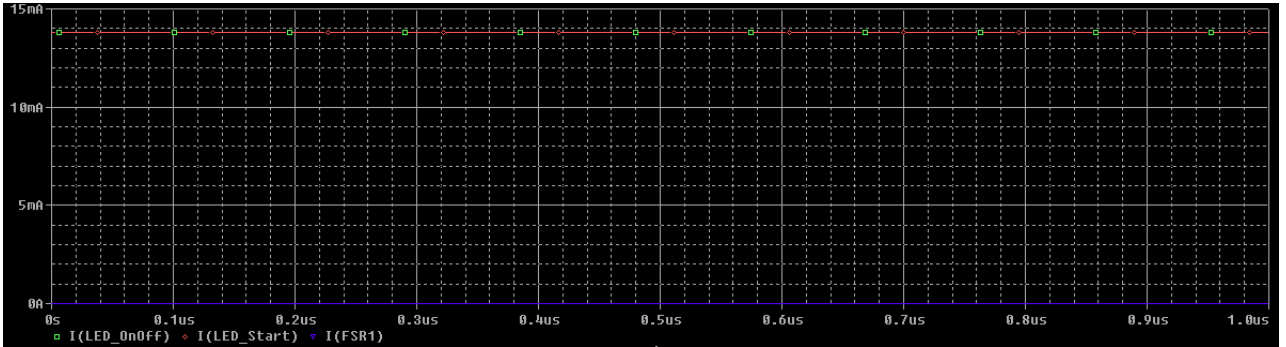
项目参数 (Parameter)	符号 (Symbol)	数值	单位 (Unit)
最大功耗 (Max Power Dissipation)	P _M	80	mW
最大正向电流 (Max Continuous Forward Current)	I _{FM}	20	mA
最大反向电压 (Max Reverse Voltage)	V _{RM}	5	V
最大脉冲峰值电流 (Peak Forward Current)	I _{FP}	75	mA
焊接温度/时间 (Lead Soldering Temperature/Time)	T _{SOL}	240/≤3S	°C/S
工作环境 (Operating Temperature Range)	T _{OPR}	-25~+85	°C
储存温度 (Storage Temperature Range)	T _{STR}	-30~+100	°C

Annex 3: OrCAD Simulations:

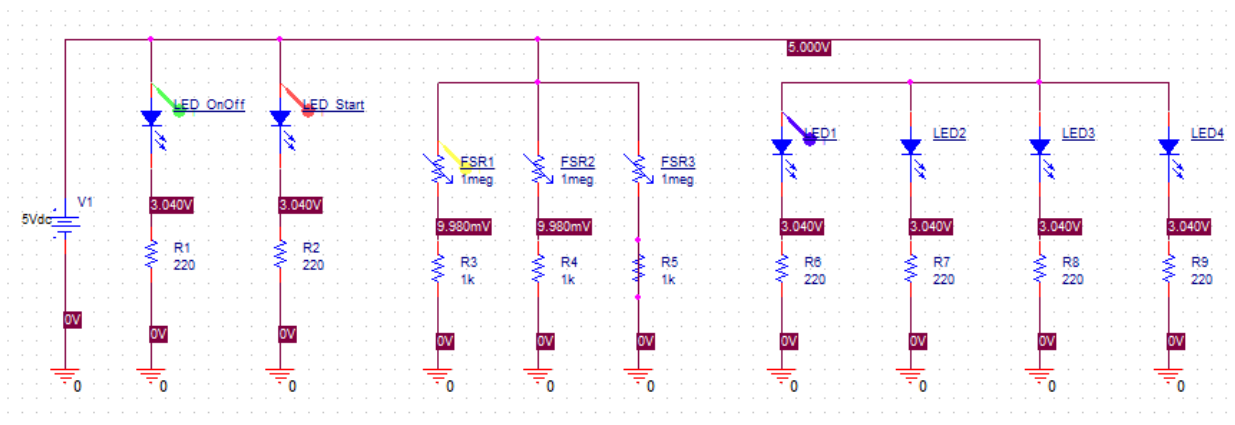
- First configuration schematic:



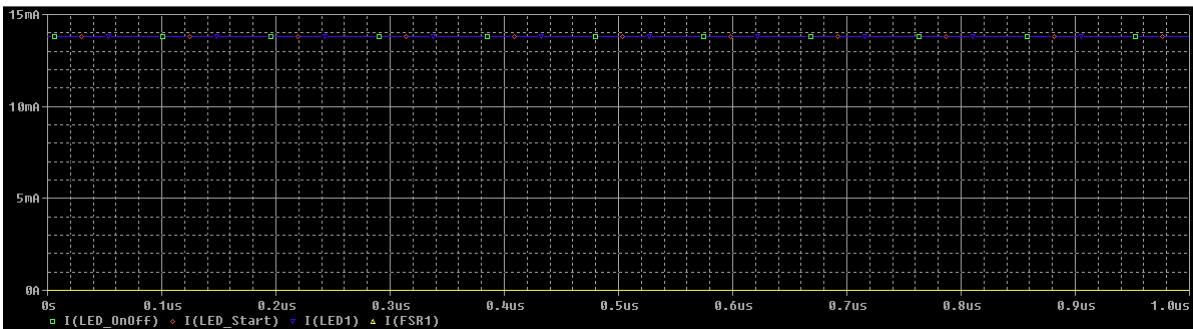
- First configuration graphical results:



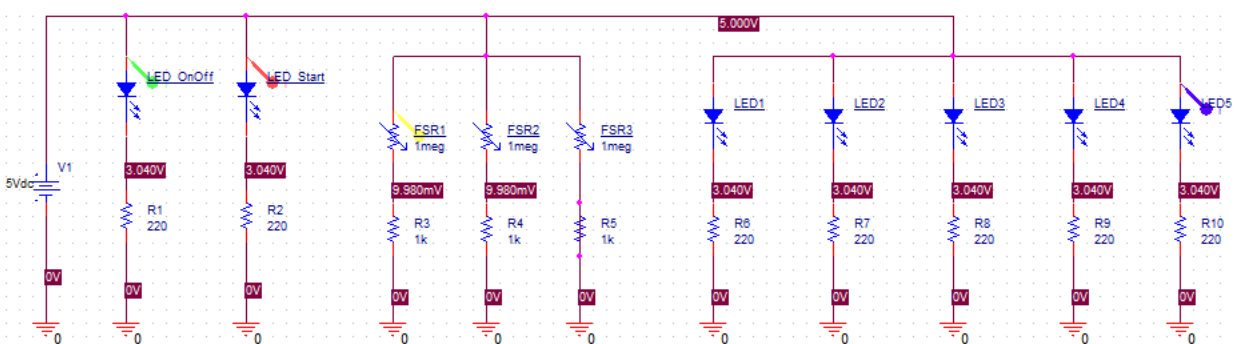
- Second configuration schematic:



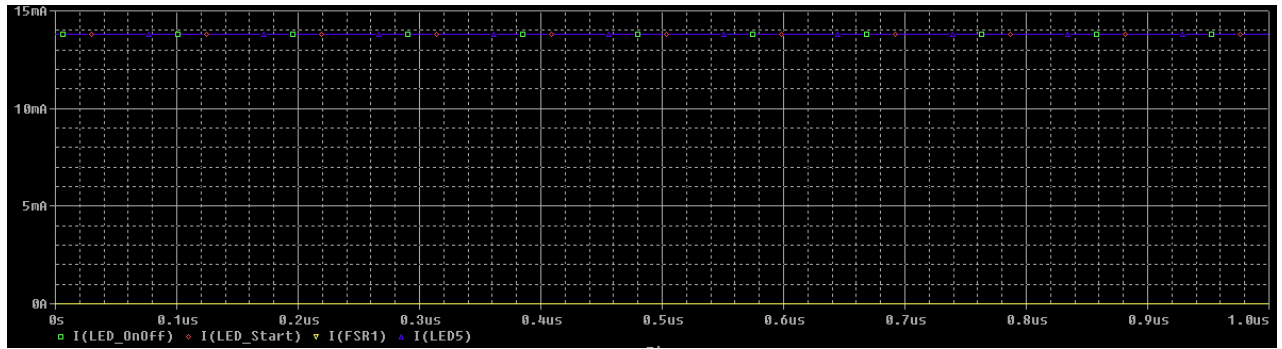
- Second configuration graphical results:



- Third configuration schematic:



- Third configuration graphical results:



Annex 4: Final code

```
1 //Pin numbers for LEDs:
2 const int ledBlue = 2;
3 const int ledRed = 3;
4 const int ledGreen = 4;
5 const int ledYellow = 6;
6 const int ledRedStart = 5;
7
8 //Pin numbers for Sensors
9 const int sensorBlue = A0;
10 const int sensorRed = A3;
11 const int sensorGreen = A5;
12
13 // Constants used to define duration of game parameters
14 int inGame = 0; //inGame parameter to start running the game
15 const int timeHit = 3000; //Max time to hit
16 int timeOutHit = 0; //time out parameter
17
18 int randomLed = 2; //Starting randomLed light to turn on
19 int numberHits = 0; //parameter used to count number of hits
20
21
22 int startGame = LOW; // Start game variable
23 int OnOffSwitch = LOW; // On Off switch variable
24
25
26 void setup() {
27   Serial.begin(9600);
28   //inicialize the LED pin as an OUTPUTS:
29   pinMode(ledBlue, OUTPUT);
30   pinMode(ledRed, OUTPUT);
31   pinMode(ledGreen, OUTPUT);
32   pinMode(ledRedStart, OUTPUT);
33
34   //inicialize the pushbuttons as INPUTS:
35   pinMode(sensorBlue, INPUT);
36   pinMode(sensorRed, INPUT);
37   pinMode(sensorGreen, INPUT);
38
39 }
40 void lightSequence() {
41
42   digitalWrite(randomLed, HIGH); //Turn on random selected light
43
44   int BlueSensorRead = analogRead(sensorBlue);
45   int RedSensorRead = analogRead(sensorRed);
46   int GreenSensorRead = analogRead(sensorGreen);
47
48   if (timeOutHit <= timeHit) {
49     timeOutHit += 1;
50     if (digitalRead(ledBlue) == HIGH) {
51       if (BlueSensorRead > 800) {
52         Serial.println("Blue hit"); //Print state of BlueButton
53         Serial.println(BlueSensorRead);
54         delay(200);
55         digitalWrite(randomLed, LOW); //Turn off the random selected light after pushbutton hit
56         delay(300);
57         timeOutHit = 0;
58         numberHits += 1;
59
60         randomLed = random(2, 5); // pick random light to turn on
61         Serial.println(randomLed); // print the pins led randomly chosen light
62         lightSequence();
```

```

63     }
64 }
65
66 if (digitalRead(ledRed) == HIGH) {
67
68     if (RedSensorRead > 800) {
69         Serial.println("Red hit");           //Print state of BlueButton
70         Serial.println(RedSensorRead);
71         delay(200);
72         digitalWrite(randomLed, LOW);      //Turn off the random selected light after pushbutton hit
73         delay(300);
74         timeOutHit = 0;
75         numberHits += 1;
76
77         randomLed = random(2, 5);          // pick random light to turn on
78         Serial.println(randomLed);         // print the pins led randomly chosen light
79         lightSequence();
80     }
81 }
82
83 if (digitalRead(ledGreen) == HIGH) {
84
85     if (GreenSensorRead > 800) {
86         Serial.println("Green hit");       //Print state of BlueButton
87         Serial.println(GreenSensorRead);
88         delay(200);
89         digitalWrite(randomLed, LOW);      //Turn off the random selected light afeter pushbutton hit
90         delay(300);
91         timeOutHit = 0;
92         numberHits += 1;
93
94         randomLed = random(2, 5);          // pick random light to turn on
95         Serial.println(randomLed);         // print the pins led randomly chosen light
96         lightSequence();
97     }
98 }
99
100 }
101 if (timeOutHit >= timeHit) {               //If no hit is detected in TimeOutHit seconds another random light is selected
102     digitalWrite(randomLed, LOW);
103     delay(100);
104     timeOutHit = 0;
105     randomLed = random(2, 5);
106     Serial.println(randomLed);
107     lightSequence();
108 }
109 }
110 }
111
112 void loop() {
113     // Code that will run repeatedly:
114
115     int OnOffSwitch = digitalRead(6);
116     int StartButtonRead = digitalRead(7);
117     if (OnOffSwitch == HIGH) {             // On Off switch controlling light ilumination
118         if ( inGame == 0) {
119             digitalWrite(ledRedStart, HIGH);
120             delay(300);
121             digitalWrite(ledRedStart, LOW);
122             delay(300);
123             digitalWrite(ledRedStart, HIGH);
124             //Serial.println("LedRedStart ON");

```



```

125
126
127     if (StartButtonRead == HIGH) {           //If StartButton is hit, the Game starts
128         Serial.println("Game Started");
129         delay(1000);
130         lightSequence();
131         inGame = 1;
132     }
133 }
134
135
136 }
137
138
139 else {                                       //"Else" used to count how long the game is running
140     inGame = inGame + 1;
141     Serial.println(inGame);
142     lightSequence();
143 }
144
145 if ( inGame >= 10000) {
146     digitalWrite(randomLed, LOW);
147     Serial.println("Game Finished");
148     Serial.println("You have hit:");
149     Serial.println(numberHits);
150     inGame = 0;
151     delay(5000);
152 }
153 }
154 else{                                       //If switch closed, all off.
155     digitalWrite(ledRedStart, LOW);
156     digitalWrite(OnOffSwitch, LOW);
157     digitalWrite(ledBlue, LOW);
158     digitalWrite(ledRed, LOW);
159     digitalWrite(ledGreen, LOW);
160     inGame = 0;
161 }
162 }

```