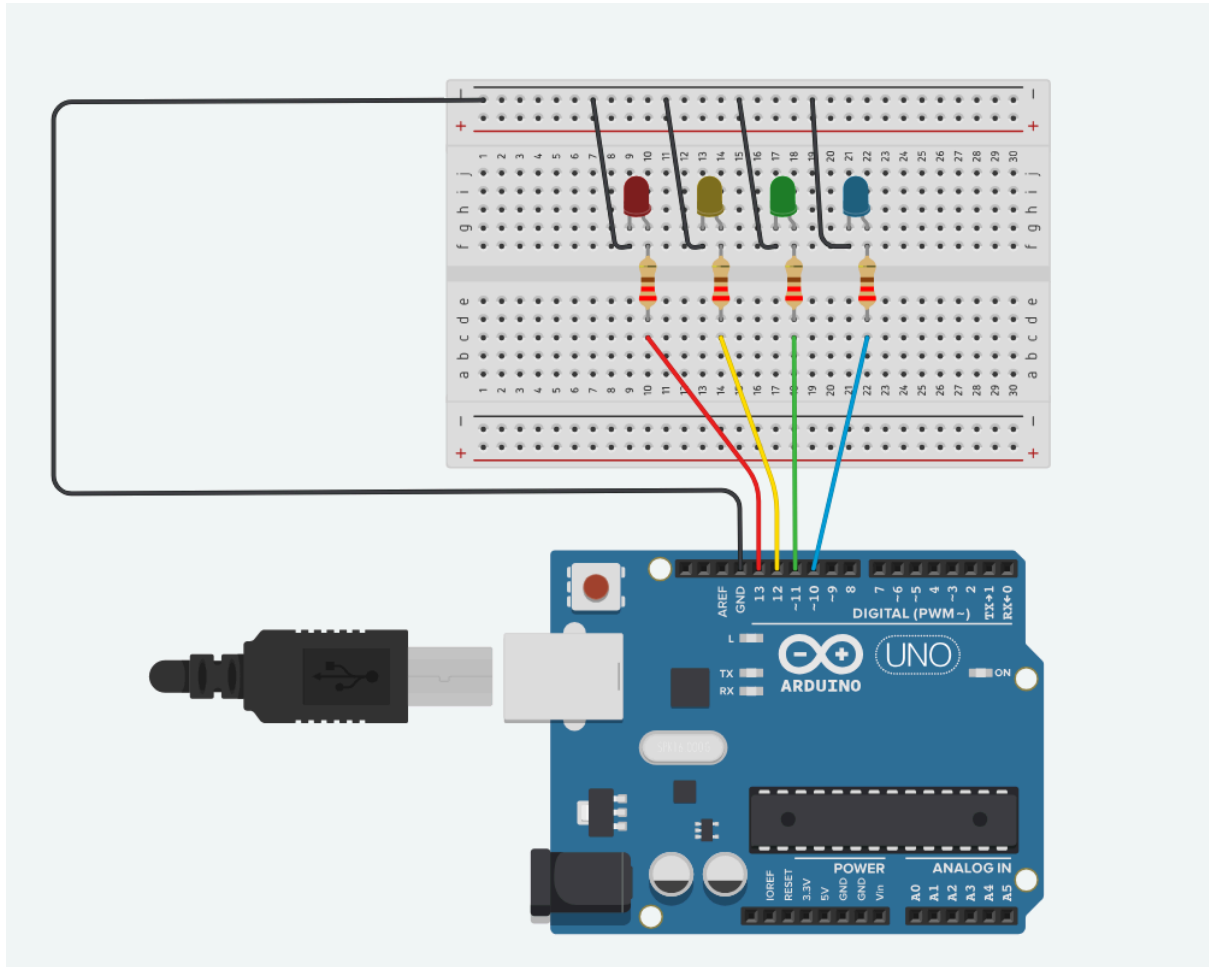


## Exercício Prático 3 - AC-II

### 1. Semáforo temporizado:



```

1  const int pinoVermelho = 13;
2  const int pinoAmarelo = 12;
3  const int pinoVerde = 11;
4  const int pinoAzul = 10;
5
6  void setup() {
7      pinMode(pinoVermelho, OUTPUT);
8      pinMode(pinoAmarelo, OUTPUT);
9      pinMode(pinoVerde, OUTPUT);
10     pinMode(pinoAzul, OUTPUT);
11 }
12
13 void loop() {
14     for (int ciclo = 0; ciclo < 3; ciclo++) {
15         digitalWrite(pinoVermelho, HIGH); // Acende o LED Vermelho
16         delay(1000); // Espera 1 segundo
17         digitalWrite(pinoVermelho, LOW); // Apaga o LED Vermelho
18         delay(1000); // Espera 1 segundo
19
20         digitalWrite(pinoAzul, HIGH); // Acende o LED Azul
21         delay(1000); // Espera 1 segundo
22         digitalWrite(pinoAzul, LOW); // Apaga o LED Azul
23         delay(1000); // Espera 1 segundo
24     }
25
26     for (int ciclo = 0; ciclo < 4; ciclo++) {
27         digitalWrite(pinoVerde, HIGH); // Acende o LED Verde
28         delay(1000); // Espera 1 segundo
29         digitalWrite(pinoVerde, LOW); // Apaga o LED Verde
30         delay(1000); // Espera 1 segundo
31
32         digitalWrite(pinoAzul, HIGH); // Acende o LED Azul
33         delay(1000); // Espera 1 segundo
34         digitalWrite(pinoAzul, LOW); // Apaga o LED Azul
35         delay(1000); // Espera 1 segundo
36     }
37
38     for (int ciclo = 0; ciclo < 2; ciclo++) {
39         digitalWrite(pinoAmarelo, HIGH); // Acende o LED Amarelo
40         delay(1000); // Espera 1 segundo
41         digitalWrite(pinoAmarelo, LOW); // Apaga o LED Amarelo
42         delay(1000); // Espera 1 segundo
43
44         digitalWrite(pinoAzul, HIGH); // Acende o LED Azul
45         delay(1000); // Espera 1 segundo
46         digitalWrite(pinoAzul, LOW); // Apaga o LED Azul
47         delay(1000); // Espera 1 segundo
48     }
49 }
50

```

## 2. ULA de 1 bit:

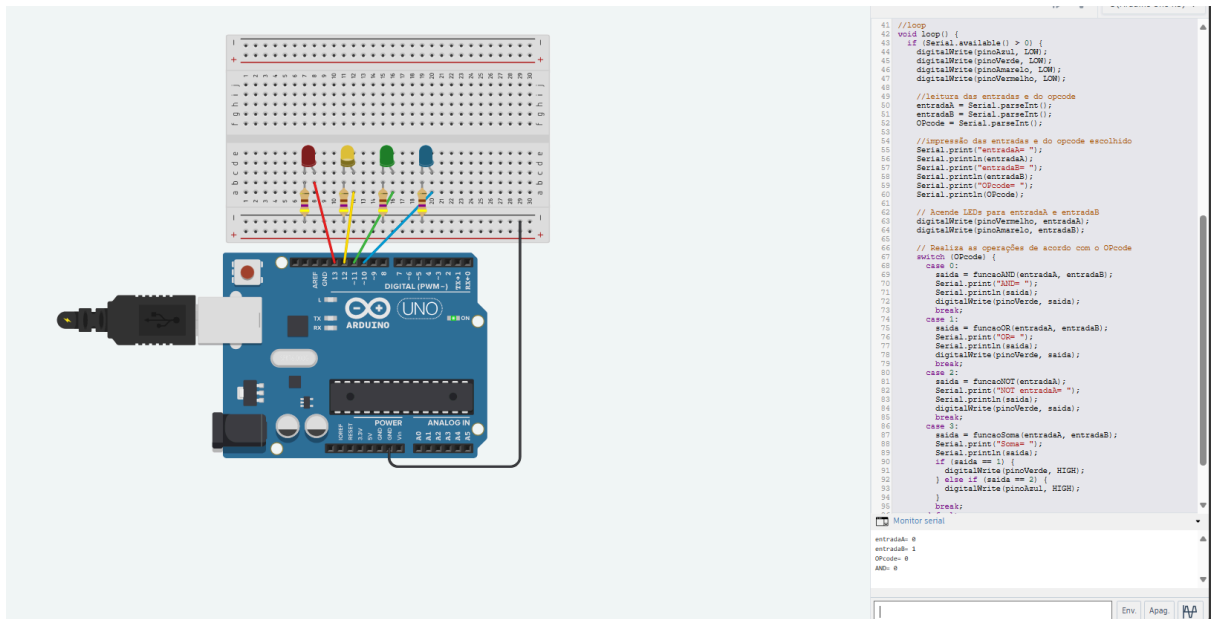
```

41 //loop
42 void loop() {
43     if (Serial.available() > 0) {
44         digitalWrite(pinoAzul, LOW);
45         digitalWrite(pinoVerde, LOW);
46         digitalWrite(pinoAmarelo, LOW);
47         digitalWrite(pinoVermelho, LOW);
48
49         //leitura das entradas e do opcode
50         entradaA = Serial.parseInt();
51         entradaB = Serial.parseInt();
52         OPcode = Serial.parseInt();
53
54         //impressão das entradas e do opcode escolhido
55         Serial.print("entradaA= ");
56         Serial.println(entradaA);
57         Serial.print("entradaB= ");
58         Serial.println(entradaB);
59         Serial.print("OPcode= ");
60         Serial.println(OPcode);
61
62         // Acende LEDs para entradaA e entradaB
63         digitalWrite(pinoVermelho, entradaA);
64         digitalWrite(pinoAmarelo, entradaB);
65
66         // Realiza as operações de acordo com o OPcode
67         switch (OPcode) {
68             case 0:
69                 saida = funcaoAND(entradaA, entradaB);
70                 Serial.print("AND= ");
71                 Serial.println(saida);
72                 digitalWrite(pinoVerde, saida);
73                 break;
74             case 1:
75                 saida = funcaoOR(entradaA, entradaB);
76                 Serial.print("OR= ");
77                 Serial.println(saida);
78                 digitalWrite(pinoVerde, saida);
79                 break;
80             case 2:
81                 saida = funcaoNOT(entradaA);
82                 Serial.print("NOT entradaA= ");
83                 Serial.println(saida);
84                 digitalWrite(pinoVerde, saida);
85                 break;
86             case 3:
87                 saida = funcaoSoma(entradaA, entradaB);
88                 Serial.print("Soma= ");
89                 Serial.println(saida);
90                 if (saida == 1) {
91                     digitalWrite(pinoVerde, HIGH);
92                 } else if (saida == 2) {
93                     digitalWrite(pinoAzul, HIGH);
94                 }
95                 break;
96         }
97     }
98 }

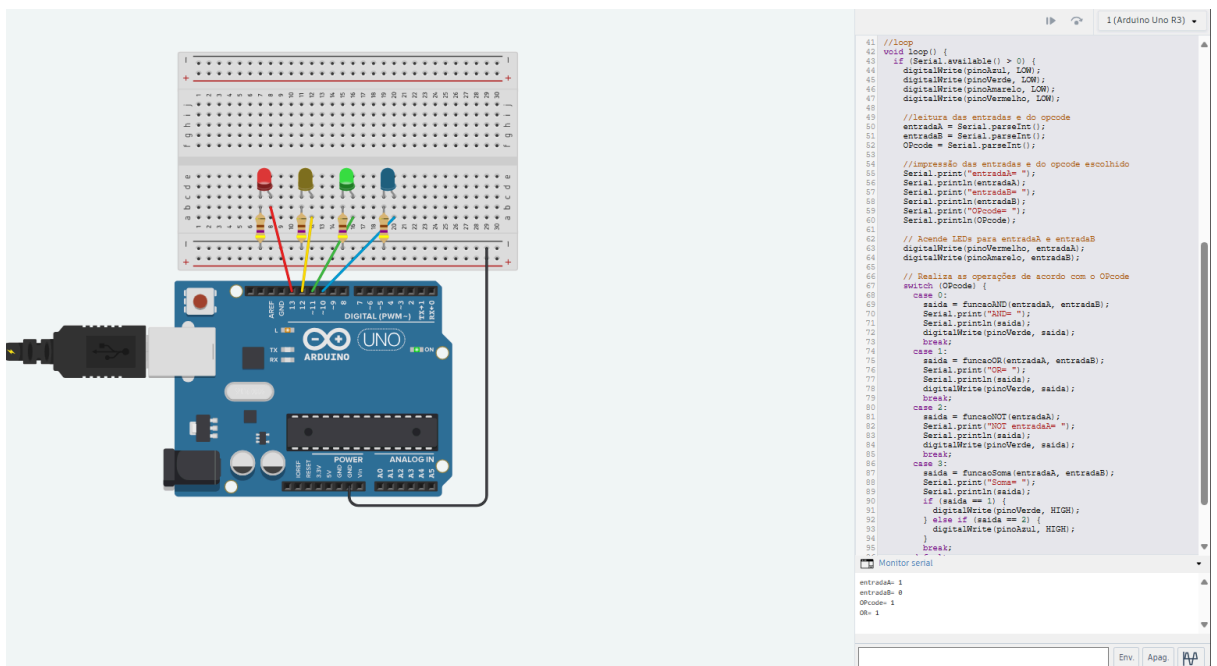
```

Testes feitos:

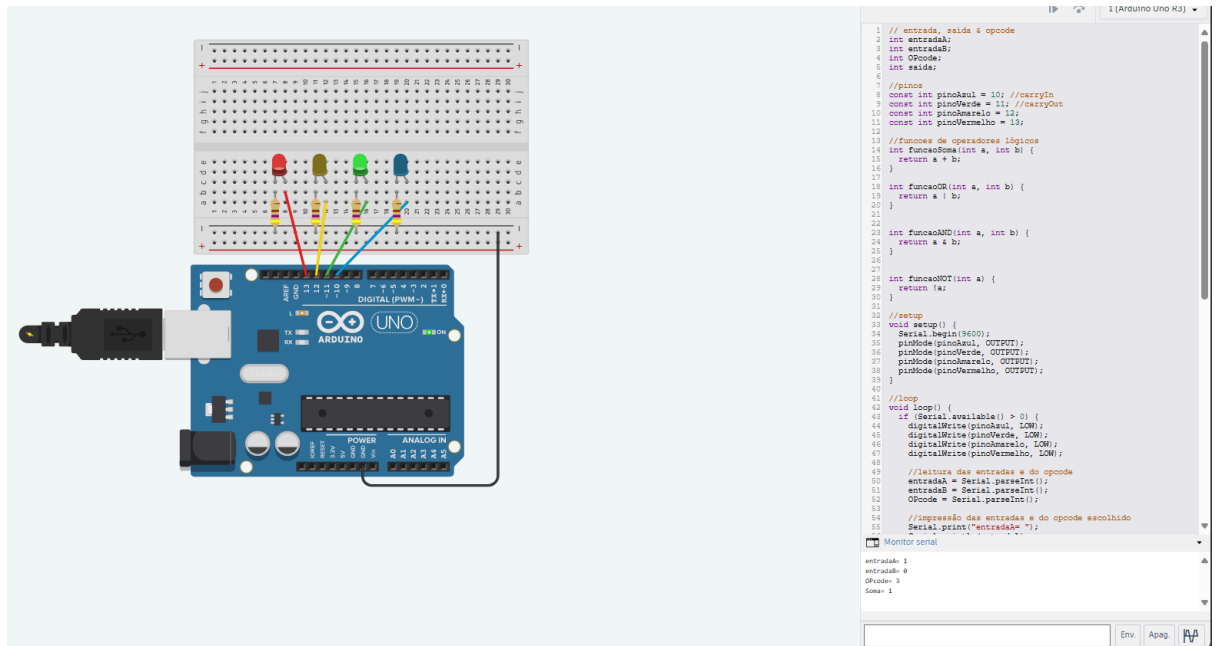
- 0 x 1 x 0



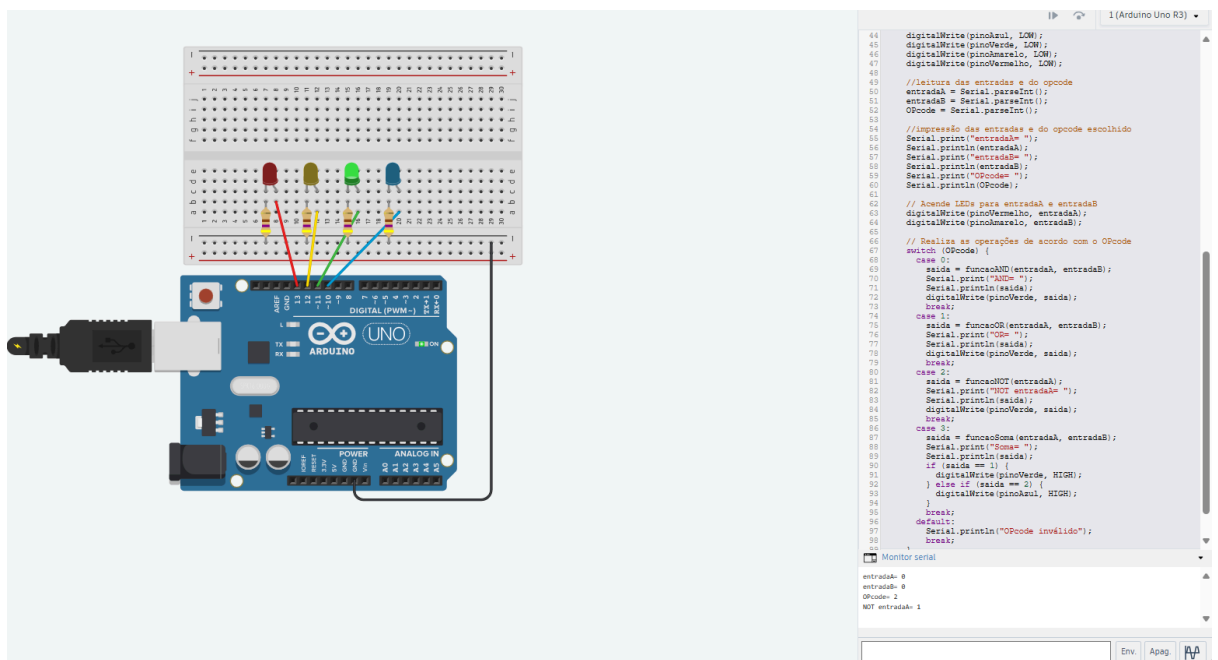
- 1 x 0 x 1



- 1 x 0 x 3



- 0 x 0 x 2



- 0 x 1 x 0

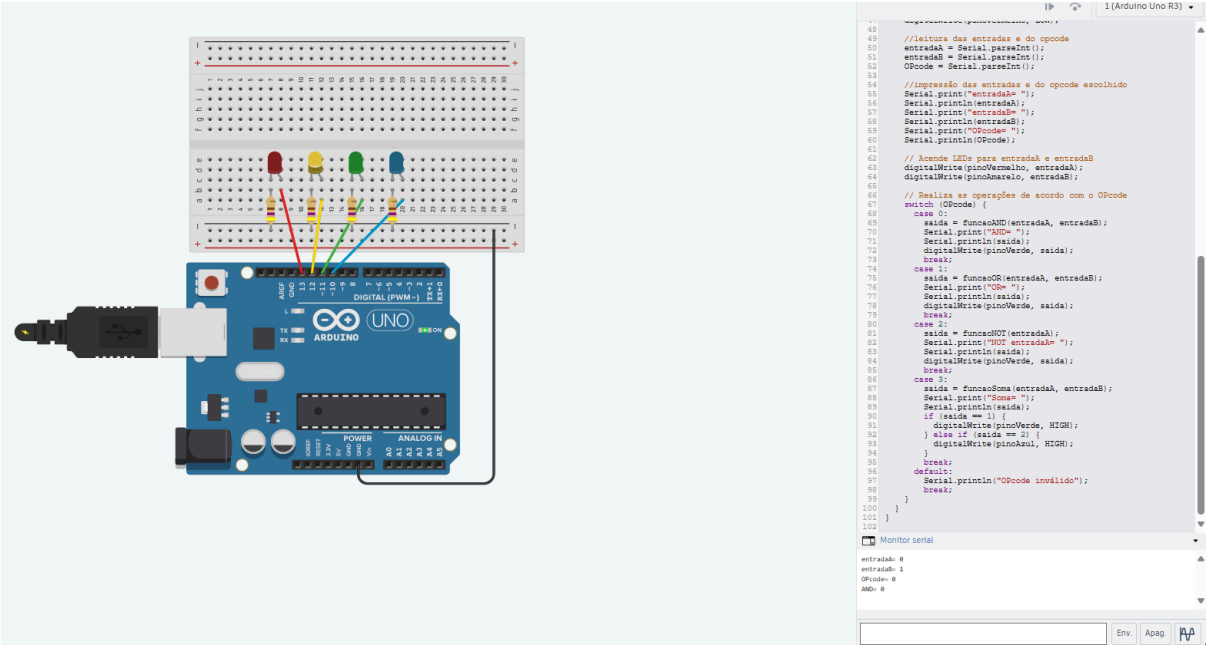


Tabela preenchida:

Instrução realizada	Binário (A,B,Op.code)	Valor em Hexa (0x ...)	Resultado em binário
AND(A,B)	0 1 00	0x4	0
OR(A,B)	10 01	0x9	1
SOMA(A,B)	1 0 11	0Xb	1
NOT(A)	0 0 10	0X2	1
AND(B,A)	0 1 00	0X4	0