



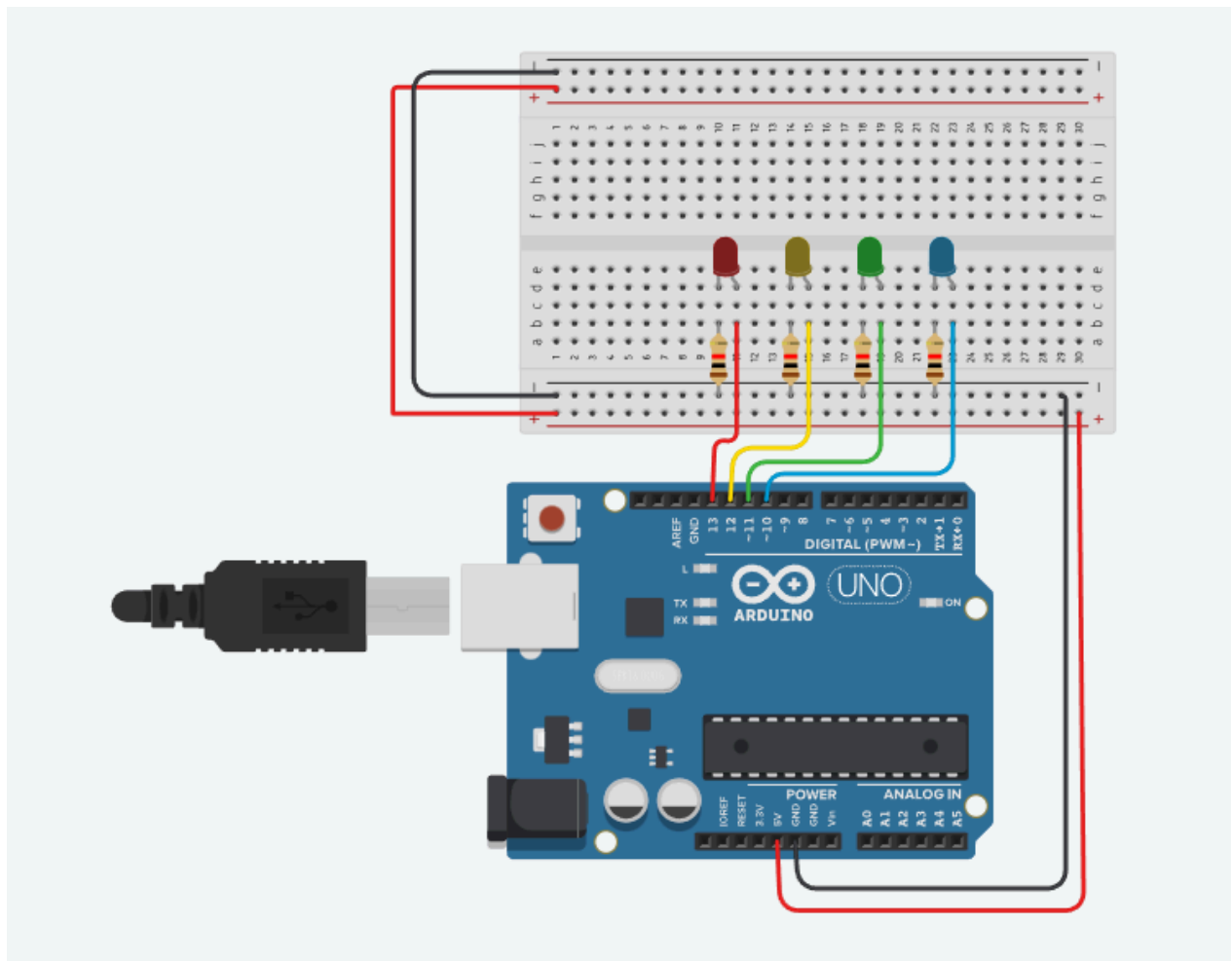
**Pontifícia Universidade Católica de Minas Gerais**  
**Instituto de Ciências Exatas e Informática**  
**Disciplina:** Arquitetura de Computadores II  
**Atividade:** Exercício Prático III

**Prof.:** Romanelli Lodron Zuim

**Nome:** Davi Cândido de Almeida \_857859

## Parte 1 - (O Arduino):

### 1. Montagem do Circuito



Disponível em:

<https://www.tinkercad.com/things/erN3fIZj6BP-semaforoex3part1>

## 2. Programa

```
1  /*
2    Exercicio 01
3    Semafaro Temporizado
4  */
5
6  // Definicao de valores para variáveis
7  int led10_Blue = 10;
8  int led11_Green = 11;
9  int led12_Yellow = 12;
10 int led13_Red = 13;
11
12 bool estadoBlue = false;
13 int contRed=0; // 3 Ciclos
14 int contGreen=0; // 4 Ciclos
15 int contYellow=0; // 2 Ciclos
16
17
18 void setup() {
19     // configura os pinos como saídas DIGITAIS.
20     pinMode(led10_Blue, OUTPUT);
21     pinMode(led11_Green, OUTPUT);
22     pinMode(led12_Yellow, OUTPUT);
23     pinMode(led13_Red, OUTPUT);
24 }
25
26
27
28 void loop() {
29     estadoBlue = !estadoBlue;
30     digitalWrite(led10_Blue, estadoBlue);
31
32     if(estadoBlue){
33
34         if(contRed < 3){
35             digitalWrite(led12_Yellow, 0);
36             digitalWrite(led13_Red, 1);
37             contRed++;
38         }
39
40         if(contRed >= 3 && contGreen < 4){
41             digitalWrite(led13_Red, 0);
42             digitalWrite(led11_Green, 1);
43             contGreen++;
44         }
45
46         if(contRed >= 3 && contGreen >= 4 && contYellow < 2){
47             digitalWrite(led11_Green, 0);
48             digitalWrite(led12_Yellow, 1);
49             contYellow++;
50         }
51
52         if(contRed >= 3 && contGreen >= 4 && contYellow >= 2)
53             contRed = contGreen = contYellow = 0;
54
55     }
56
57     delay(1000);
58 }
```

## Parte 2 - (ULA de 1 Bit):

Obs: Foram feitas duas versões:

- Caso 1: Entradas de input separadas/intercaladas e com números em decimal
- Caso 2: Entrada única e com números em binário

Obs: O circuito se mantém o mesmo nos dois casos

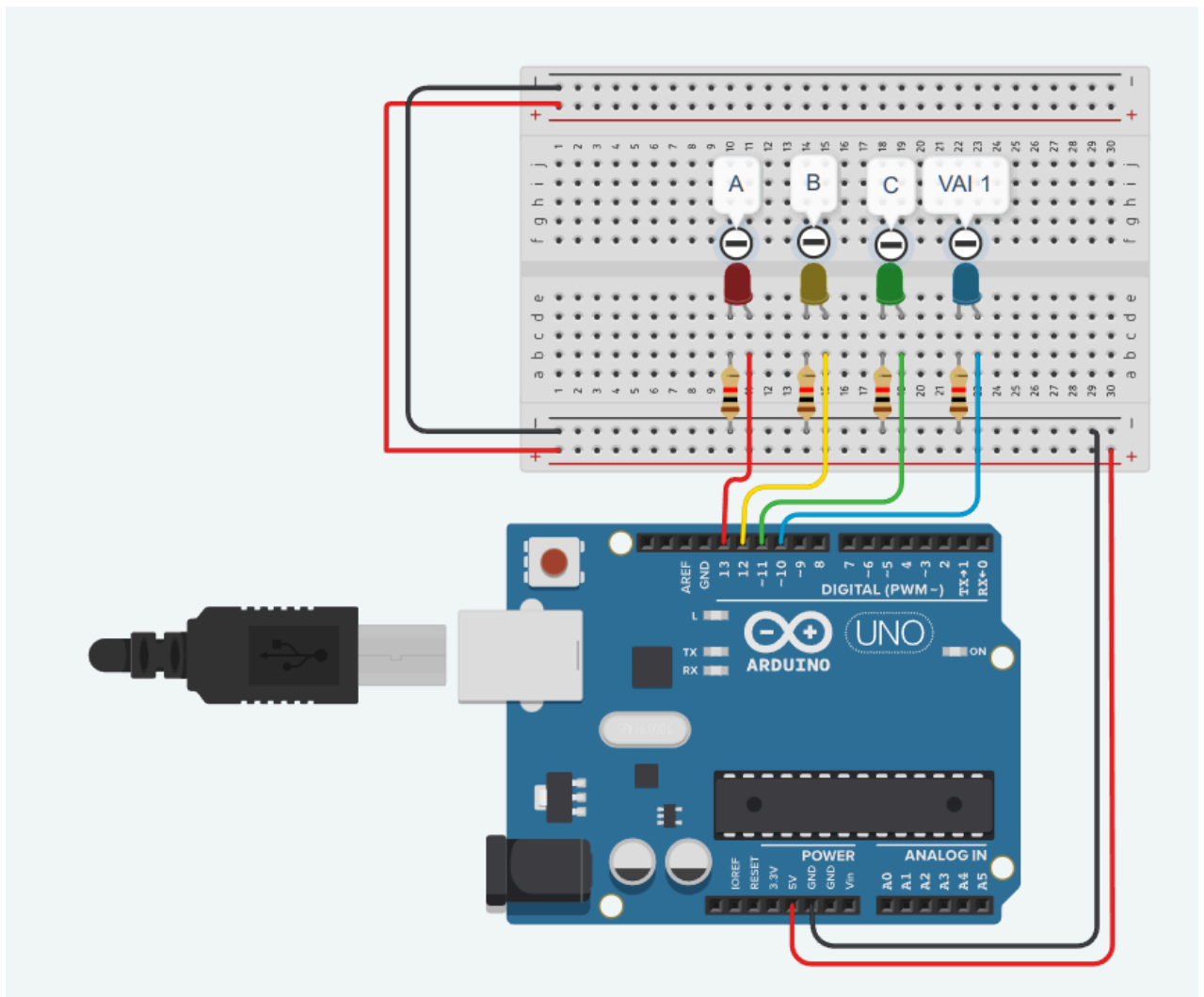
Caso 1 disponível em:

<https://www.tinkercad.com/things/ioOnVOLUvST-semaforoex3part2caso1>

Caso 2 disponível em:

<https://www.tinkercad.com/things/bQuyffP9NIt-semaforoex3part2caso2>

### 3. Montagem do Circuito da ULA



#### 4. Algoritmo caso 1:

```
int led13_A=13;
int led12_B=12;
int led11_C=11;
int led10_Vail=10;
int saida;
int vail;

void setup() {
  Serial.begin(9600);
  pinMode(led13_A,OUTPUT);
  pinMode(led12_B,OUTPUT);
  pinMode(led11_C,OUTPUT);
  pinMode(led10_Vail,OUTPUT);
}

void loop() {
  if (Serial.available() > 0) {
    int a = Serial.parseInt();
    int b = Serial.parseInt();
    int OP = Serial.parseInt();

    Serial.print("a= ");
    Serial.print(a);
    Serial.println();
    Serial.print("b= ");
    Serial.print(b);
    Serial.println();
    Serial.print("OP= ");
    Serial.print(OP);
    Serial.println();

    switch(OP){
      case 0:
        Serial.print("AND(");
        Serial.print(a);
        Serial.print(",");
        Serial.print(b);
        Serial.print(")= ");
        saida = portaAND(a, b);
        Serial.println(saida);
        break;
      case 1:
        Serial.print("OR(");
        Serial.print(a);
        Serial.print(",");
        Serial.print(b);
        Serial.print(")= ");
        saida = portaOR(a, b);
        Serial.println(saida);
        break;
      case 2:
        Serial.print("NOT(");
        Serial.print(a);
        Serial.print(")= ");
        saida = portaNOT(a);
        Serial.println(saida);
        break;
    }
  }
}
```

```
case 3:
  Serial.print("SOMA(");
  Serial.print(a);
  Serial.print(",");
  Serial.print(b);
  Serial.print(")= ");
  saida = portaSOMA(a, b);
  vail = portaAND(a, b);
  Serial.println(saida);

  Serial.print("Vail = ");
  Serial.println(vail);
  break;
}

if(OP > 3 || OP < 0)
  Serial.println("Entrada de OP invalida!");

if(vail)
  mostrar(a,b,saida, vail);
else
  mostrar(a,b,saida);
}

int portaOR(int a, int b){
  return(a|b);
}

int portaAND(int a, int b){
  return(a&b);
}

int portaNOT(int a){
  return(!a);
}

int portaSOMA(int a, int b){
  return(a+b);
}

void mostrar(int a, int b, int saida){
  digitalWrite(led13_A, a > 0);
  digitalWrite(led12_B, b > 0);
  digitalWrite(led11_C, saida > 0);
  digitalWrite(led10_Vail, 0);
}

void mostrar(int a, int b, int saida, int vail){
  digitalWrite(led13_A, a > 0);
  digitalWrite(led12_B, b > 0);
  digitalWrite(led11_C, saida > 0);
  digitalWrite(led10_Vail, vail > 0);
}
```

## 5. Algoritmo caso 2:

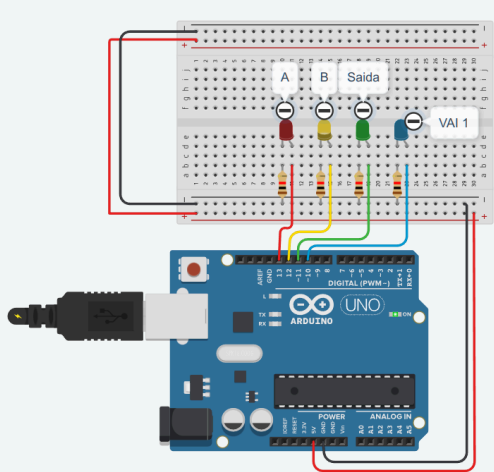
```
1 int led13_A=13;
2 int led12_B=12;
3 int led11_C=11;
4 int led10_Vail=10;
5 int saida;
6 int vail;
7
8 int vetor[4];
9
10 void setup() {
11     Serial.begin(9600);
12     pinMode(led13_A,OUTPUT);
13     pinMode(led12_B,OUTPUT);
14     pinMode(led11_C,OUTPUT);
15     pinMode(led10_Vail,OUTPUT);
16 }
17
18 void loop() {
19     if (Serial.available() > 0) {
20
21         int entrada = Serial.parseInt();
22
23         separarDigitos(entrada);
24
25         bool a = vetor[0];
26         bool b = vetor[1];
27         bool OP1 = vetor[2];
28         bool OP2 = vetor[3];
29
30
31         Serial.print("a= "); Serial.println(a);
32         Serial.print("b= "); Serial.println(b);
33         Serial.print("OP= ");
34         Serial.print(OP1); Serial.println(OP2);
35
36         if(!OP1 && !OP2){
37             Serial.print("AND(");
38             Serial.print(a);
39             Serial.print(",");
40             Serial.print(b);
41             Serial.print(")= ");
42             saida = portaAND(a, b);
43             Serial.println(saida);
44         }else if (!OP1 && OP2){
45             Serial.print("OR(");
46             Serial.print(a);
47             Serial.print(",");
48             Serial.print(b);
49             Serial.print(")= ");
50             saida = portaOR(a, b);
51             Serial.println(saida);
52         }else if (OP1 && !OP2){
53             Serial.print("NOT(");
54             Serial.print(a);
55             Serial.print(")= ");
56             saida = portaNOT(a);
57             Serial.println(saida);
58         }else if (OP1 && OP2){
59             Serial.print("SOMA(");
60             Serial.print(a);
61             Serial.print(",");
62             Serial.print(b);
63             Serial.print(")= ");
64             saida = portaSOMA(a, b);
65             vail = portaAND(a, b);
66             Serial.println(saida);
67
68             Serial.print("Vail = ");
69             Serial.println(vail);
70         }else{
71             Serial.println("Entrada de OP invalida!");
72         }
73
74         if(vail)
75             mostrar(a,b,saida, vail);
76         else
77             mostrar(a,b,saida);
78     }
79 }
80
81 int portaOR(int a, int b){
82     return(a|b);
83 }
84
85 int portaAND(int a, int b){
86     return(a&b);
87 }
88
89 int portaNOT(int a){
90     return(!a);
91 }
92
93 int portaSOMA(int a, int b){
94     return(a^b);
95 }
96
97 void mostrar(int a, int b, int saida){
98     digitalWrite(led13_A, a > 0);
99     digitalWrite(led12_B, b > 0);
100     digitalWrite(led11_C, saida > 0);
101     digitalWrite(led10_Vail, 0);
102 }
103
104 void mostrar(int a, int b, int saida, int vail){
105     digitalWrite(led13_A, a > 0);
106     digitalWrite(led12_B, b > 0);
107     digitalWrite(led11_C, saida > 0);
108     digitalWrite(led10_Vail, vail > 0);
109 }
110
111
112 void separarDigitos(int entrada){
113     int i = 3;
114
115     while(entrada > 0 && i >= 0){
116         vetor[i--] = entrada % 10;
117         entrada /= 10;
118     }
119
120     // Se a entrada for menor que 3 digitos, completa com 0
121     while (i >= 0) {
122         vetor[i--] = 0;
123     }
124 }
125
126
127
128
```

## 6. Preenchimento da tabela de instruções

Instrução Realizada	Binário (A, B , Op.code)	Valor em Hexa (Ox ...)	Resultado em binário
AND(A,B)	0 1 00	0x4	0
OR(A,B)	1 0 01	0x9	1
SOMA(A,B)	1 0 11	0x11	1
NOT(A)	0 0 10	0x2	1
AND(B,A)	0 1 00	0x4	0

Print Instruções:

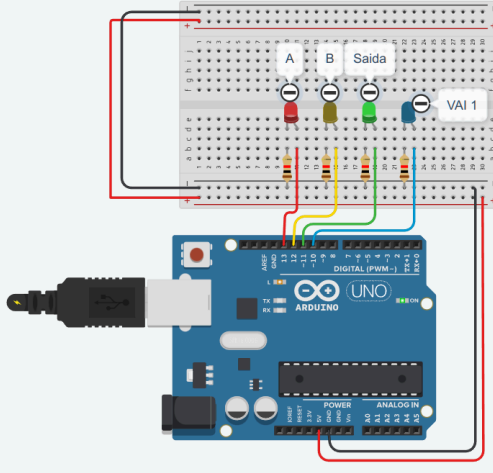
AND(A,B)	0 1 00	0x4	0
----------	--------	-----	---



```

88 }
89
90 int portaAND(int a, int b){
91     return(a&b);
92 }
93
94 int portaNOT(int a){
95     return(!a);
96 }
97
98 int portaSOMA(int a, int b){
99     return(a+b);
100 }
101
102 void mostrar(int a, int b, int saida){
103     digitalWrite(led13_A, a > 0);
104     digitalWrite(led12_B, b > 0);
105     digitalWrite(led11_C, saida > 0);
106     digitalWrite(led10_Vail, 0);
107 }
108
109 void mostrar(int a, int b, int saida, int vail){
110     digitalWrite(led13_A, a > 0);
111     digitalWrite(led12_B, b > 0);
112     digitalWrite(led11_C, saida > 0);
113     digitalWrite(led10_Vail, vail > 0);
114 }
115
116 Monitor serial
a= 0
b= 1
OP= 0
AND(0,1)= 0
  
```

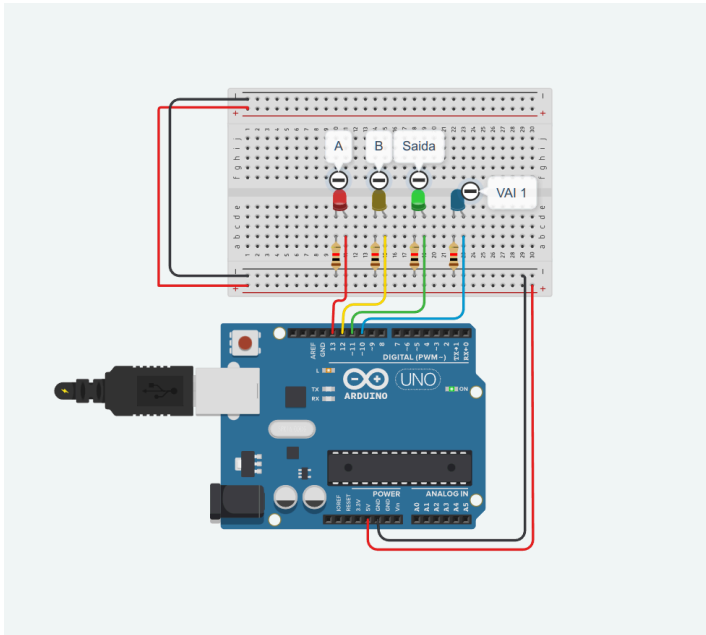
OR(A,B)	1 0 01	0x9	1
---------	--------	-----	---



```

88 }
89
90 int portaAND(int a, int b){
91     return(a&b);
92 }
93
94 int portaNOT(int a){
95     return(!a);
96 }
97
98 int portaSOMA(int a, int b){
99     return(a+b);
100 }
101
102 void mostrar(int a, int b, int saida){
103     digitalWrite(led13_A, a > 0);
104     digitalWrite(led12_B, b > 0);
105     digitalWrite(led11_C, saida > 0);
106     digitalWrite(led10_Vail, 0);
107 }
108
109 void mostrar(int a, int b, int saida, int vail){
110     digitalWrite(led13_A, a > 0);
111     digitalWrite(led12_B, b > 0);
112     digitalWrite(led11_C, saida > 0);
113     digitalWrite(led10_Vail, vail > 0);
114 }
115
116 Monitor serial
a= 1
b= 0
OP= 1
OR(1,0)= 1
  
```

SOMA(A,B)	1 0 11	0x11	1
-----------	--------	------	---

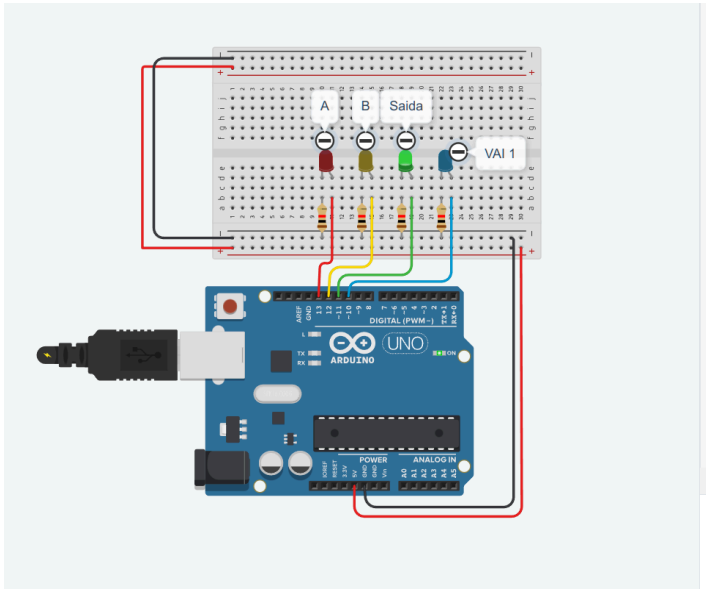


```

88 }
89
90 int portaAND(int a, int b){
91     return(a&b);
92 }
93
94 int portaNOT(int a){
95     return(!a);
96 }
97
98 int portaSOMA(int a, int b){
99     return(a+b);
100 }
101
102 void mostrar(int a, int b, int saida){
103     digitalWrite(led13_A, a > 0);
104     digitalWrite(led12_B, b > 0);
105     digitalWrite(led11_C, saida > 0);
106     digitalWrite(led10_Vail, 0);
107 }
108
109 void mostrar(int a, int b, int saida, int vail){
110     digitalWrite(led13_A, a > 0);
111     digitalWrite(led12_B, b > 0);
112     digitalWrite(led11_C, saida > 0);
113     digitalWrite(led10_Vail, vail > 0);
114 }
115
116 Monitor serial
a= 1
b= 0
OP= 3
SOMA(1,0)= 1
Vail = 0

```

NOT(A)	0 0 10	0x2	1
--------	--------	-----	---

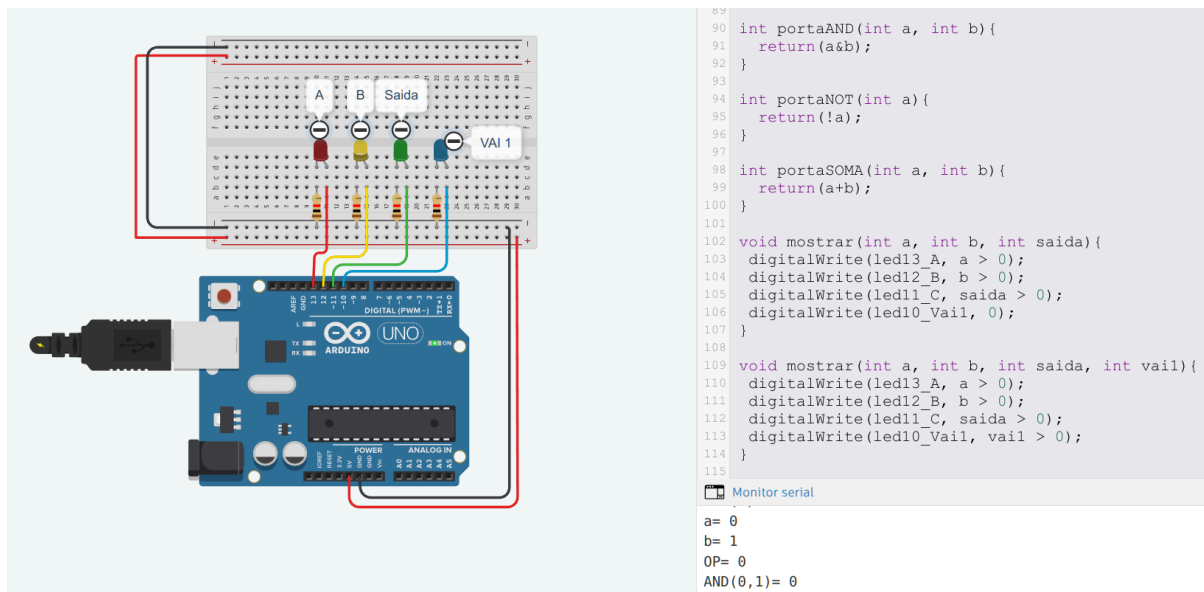


```

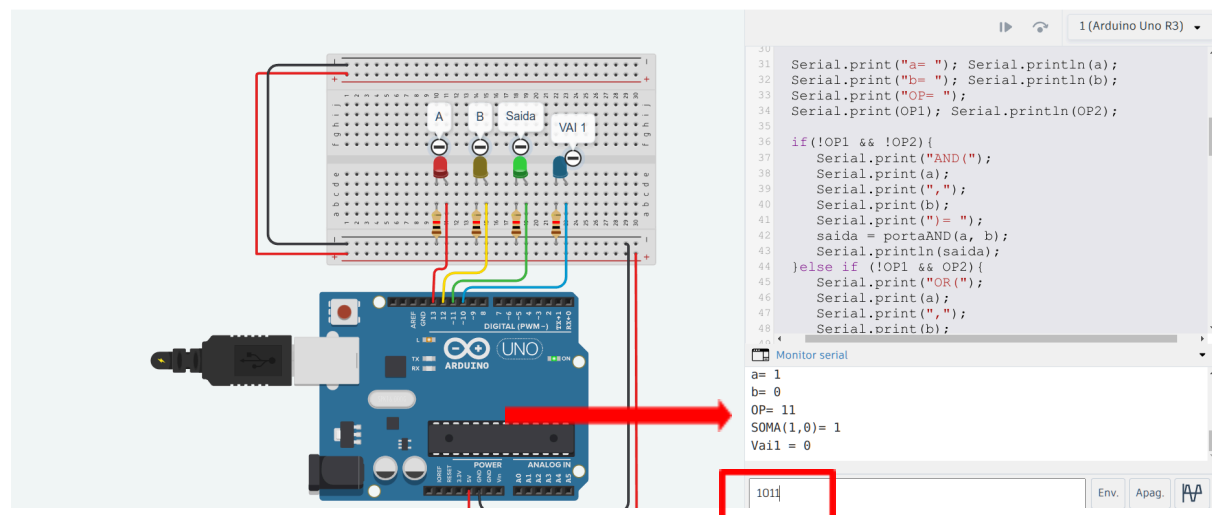
88 }
89
90 int portaAND(int a, int b){
91     return(a&b);
92 }
93
94 int portaNOT(int a){
95     return(!a);
96 }
97
98 int portaSOMA(int a, int b){
99     return(a+b);
100 }
101
102 void mostrar(int a, int b, int saida){
103     digitalWrite(led13_A, a > 0);
104     digitalWrite(led12_B, b > 0);
105     digitalWrite(led11_C, saida > 0);
106     digitalWrite(led10_Vail, 0);
107 }
108
109 void mostrar(int a, int b, int saida, int vail){
110     digitalWrite(led13_A, a > 0);
111     digitalWrite(led12_B, b > 0);
112     digitalWrite(led11_C, saida > 0);
113     digitalWrite(led10_Vail, vail > 0);
114 }
115
116 Monitor serial
a= 0
b= 0
OP= 2
NOT(0)= 1

```

AND(B,A)	0 1 00	0x4	0
----------	--------	-----	---



Exemplo de como é no caso 2 (Entrada única e somente em binário):



Veja em: <https://www.tinkercad.com/things/bQuyffP9Nlt-semaforoex3part2caso2>

GitHub com os códigos:

<https://github.com/DaviKandido/Arquitetura-de-Computadores/tree/main/Arq.%20de%20Comp.%20II/Trabalhos%20Praticos/Exercicio%20Pr%C3%A1tico%203>