

Exercício Prático 03

Disciplina: Arquitetura de Computadores II

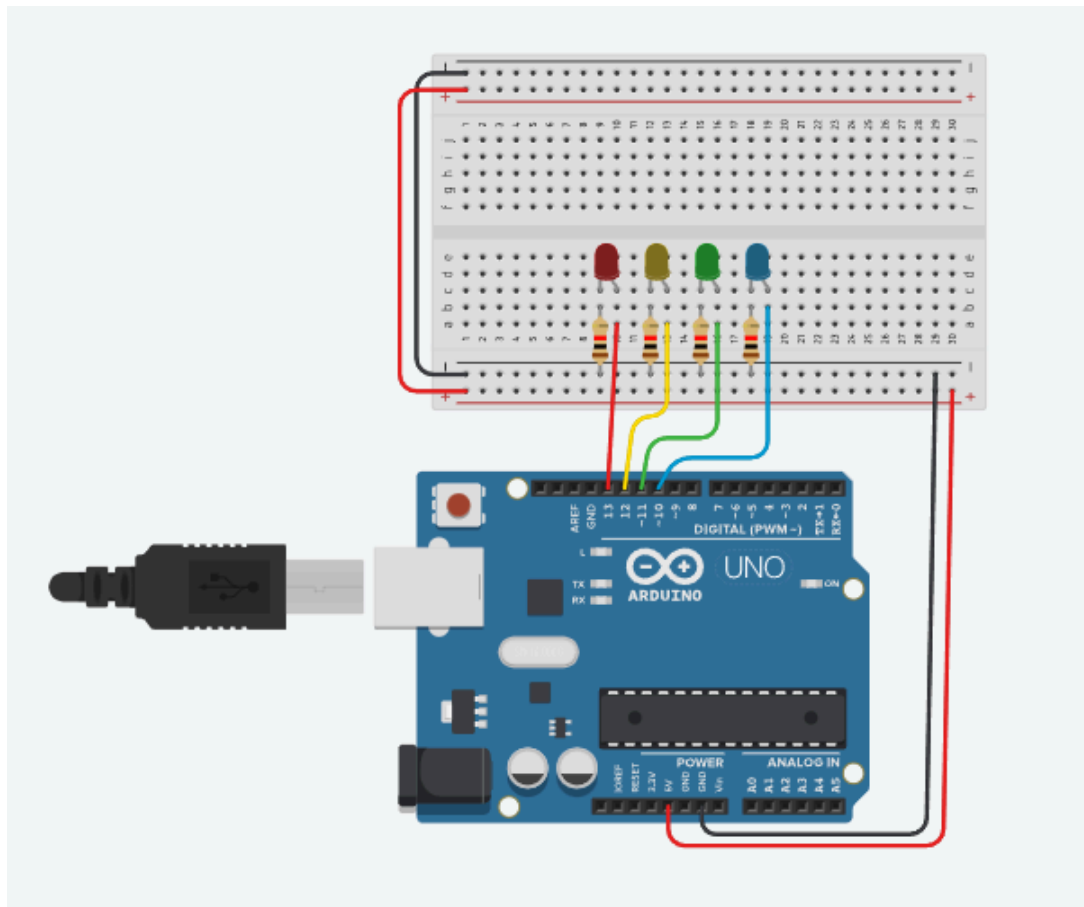
Data: 08/03/2025

Aluna: Alessandra Faria Rodrigues

Matrícula: 828333

PARTE 1

Simulação no Tinkercad de um semáforo temporizado



Projeto:

https://www.tinkercad.com/things/jSYyEFZquXE-semaforo-temporizado?sharecode=Zhh29VRn2VXJ5jmxLLQjsQjP5C5AI2u1Bciy1_JL-SU

//Código arduino C++

// Definição dos pinos dos LEDs

int ledVermelho = 13; // LED vermelho (pare)

int ledAmarelo = 12; // LED amarelo (atenção)

int ledVerde = 11; // LED verde (siga)

int ledAzul = 10; // LED azul (piscando durante cada fase)

void setup() {

 // Configuração dos pinos como saída

```

pinMode(ledVermelho, OUTPUT);
pinMode(ledAmarelo, OUTPUT);
pinMode(ledVerde, OUTPUT);
pinMode(ledAzul, OUTPUT);
}

void loop() {
    // Acende o LED vermelho (pare)
    digitalWrite(ledVermelho, HIGH);
    // Pisca o LED azul 3 vezes (indicando tempo de espera no vermelho)
    for(int i = 0; i < 3; i++){
        digitalWrite(ledAzul, HIGH);
        delay(1000);
        digitalWrite(ledAzul, LOW);
        delay(1000);
    }
    // Apaga o LED vermelho
    digitalWrite(ledVermelho, LOW);

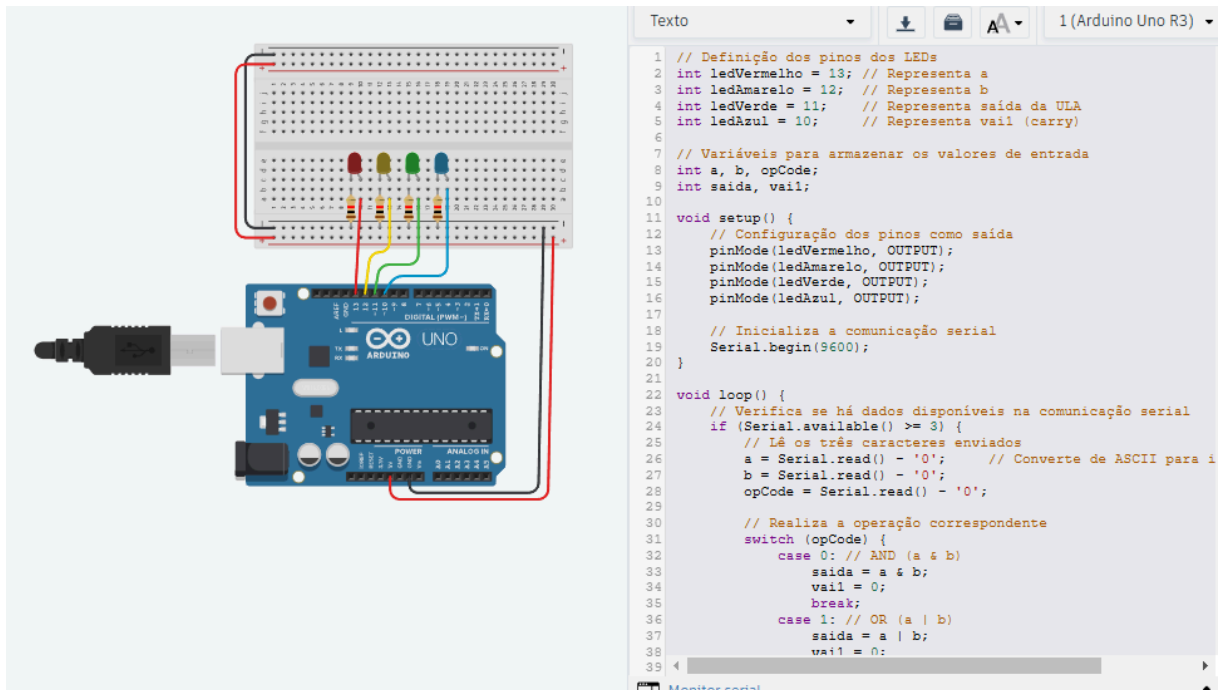
    // Acende o LED verde (siga)
    digitalWrite(ledVerde, HIGH);
    // Pisca o LED azul 4 vezes (indicando tempo de espera no verde)
    for(int i = 0; i < 4; i++){
        digitalWrite(ledAzul, HIGH);
        delay(1000);
        digitalWrite(ledAzul, LOW);
        delay(1000);
    }
    // Apaga o LED verde
    digitalWrite(ledVerde, LOW);

    // Acende o LED amarelo (atenção)
    digitalWrite(ledAmarelo, HIGH);
    // Pisca o LED azul 2 vezes (indicando tempo de espera no amarelo)
    for(int i = 0; i < 2; i++){
        digitalWrite(ledAzul, HIGH);
        delay(1000);
        digitalWrite(ledAzul, LOW);
        delay(1000);
    }
    // Apaga o LED amarelo
    digitalWrite(ledAmarelo, LOW);
}

```

PARTE 2

ULA



Projeto:

<https://www.tinkercad.com/things/i2yxV0vPpeK-ula?sharecode=WZP8mVlwYlvjNxwaAFKm70BiZodoAmKsE9Z6IMSQPQ>

//Código arduino C++

// Definição dos pinos dos LEDs

int ledVermelho = 13; // Representa a

int ledAmarelo = 12; // Representa b

int ledVerde = 11; // Representa saída da ULA

int ledAzul = 10; // Representa vai1 (carry)

// Variáveis para armazenar os valores de entrada

int a, b, opCode;

int saida, vai1;

void setup() {

// Configuração dos pinos como saída

pinMode(ledVermelho, OUTPUT);

pinMode(ledAmarelo, OUTPUT);

pinMode(ledVerde, OUTPUT);

pinMode(ledAzul, OUTPUT);

// Inicializa a comunicação serial

Serial.begin(9600);

}

```

void loop() {
    // Verifica se há dados disponíveis na comunicação serial
    if (Serial.available() >= 3) {
        // Lê os três caracteres enviados
        a = Serial.read() - '0';    // Converte de ASCII para inteiro
        b = Serial.read() - '0';
        opCode = Serial.read() - '0';

        // Realiza a operação correspondente
        switch (opCode) {
            case 0: // AND (a & b)
                saida = a & b;
                vai1 = 0;
                break;
            case 1: // OR (a | b)
                saida = a | b;
                vai1 = 0;
                break;
            case 2: // NOT (a)
                saida = !a;
                vai1 = 0;
                break;
            case 3: // Soma (a + b) com carry
                saida = (a + b) % 2; // Soma binária (bit menos significativo)
                vai1 = (a + b) / 2; // Carry (vai1)
                break;
            default: // Código inválido
                return;
        }

        // Exibir os valores nos LEDs
        digitalWrite(ledVermelho, a);
        digitalWrite(ledAmarelo, b);
        digitalWrite(ledVerde, saida);
        digitalWrite(ledAzul, vai1);

        // Exibir os resultados no monitor serial
        Serial.print("Entrada: a = ");
        Serial.print(a);
        Serial.print(", b = ");
        Serial.print(b);
        Serial.print(", opCode = ");
        Serial.println(opCode);
    }
}

```

```
        Serial.print("Saída: ");  
        Serial.print(saida);  
        Serial.print(", Vai1: ");  
        Serial.println(vai1);  
    }  
}
```

Teste da ULA:

Início:

A=0; (ou A=0)

B=1; (ou B=1)

AND(A,B); (esta é a operação bit a bit entre A e B)

B=0; (ou B=0)

A=1; (ou A=1)

OR(A,B);

SOMA(A,B); (esta é a operação aritmética da soma entre A e B)

A=0; (ou A=0)

NOT(A);

B=1; (ou B=1)

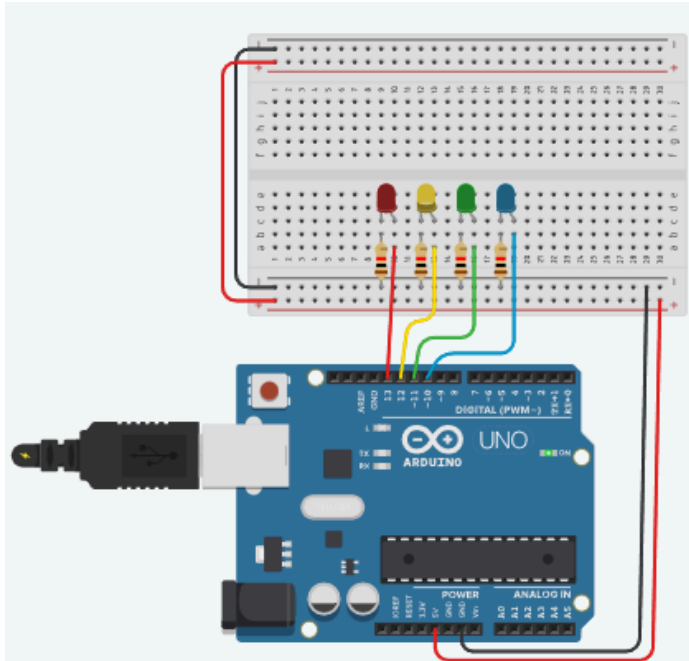
AND(B,A);

Fim.

A=0; (ou A=0)

B=1; (ou B=1)

AND(A,B);



```
1 // Definição dos pinos dos LEDs
2 int ledVermelho = 13; // Representa A
3 int ledAmarelo = 12; // Representa B
4 int ledVerde = 11; // Representa resultado
5 int ledAzul = 10; // Representa resultado
6
7 // Variáveis para armazenar os valores de entrada
8 int a, b, opCode;
9 int saida, vail;
10
11 void setup() {
12   // Configuração dos pinos como saída
13   pinMode(ledVermelho, OUTPUT);
14   pinMode(ledAmarelo, OUTPUT);
15   pinMode(ledVerde, OUTPUT);
16   pinMode(ledAzul, OUTPUT);
17
18   // Inicializa a comunicação serial
19   Serial.begin(9600);
20 }
21
22 void loop() {
23   // Verifica se há dados disponíveis
24   if (Serial.available() >= 3) {
25     // Lê os três caracteres
26     a = Serial.read() - '0';
27     b = Serial.read() - '0';
28     opCode = Serial.read() - '0';
29   }
30 }
```

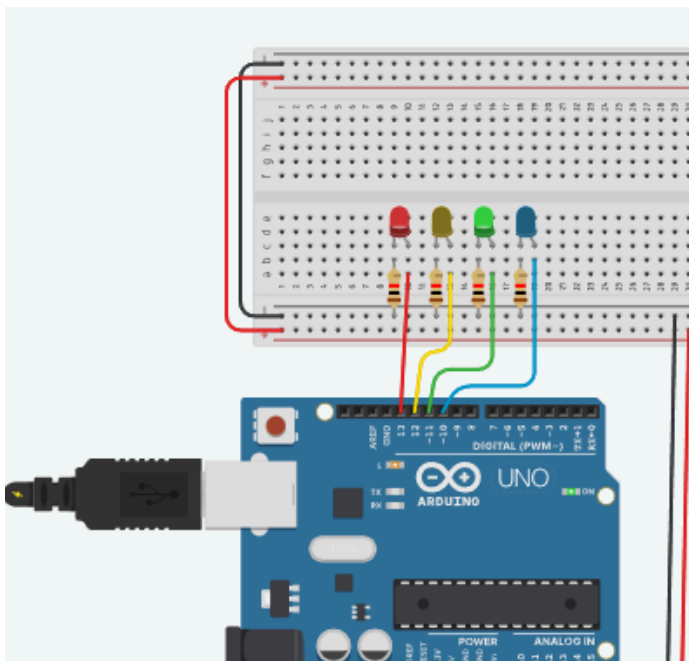
Monitor serial

Entrada: a = 0, b = 1, opCode = 0
Saída: 0, Vail: 0

B=0; (ou B=0)

A=1; (ou A=1)

OR(A,B);

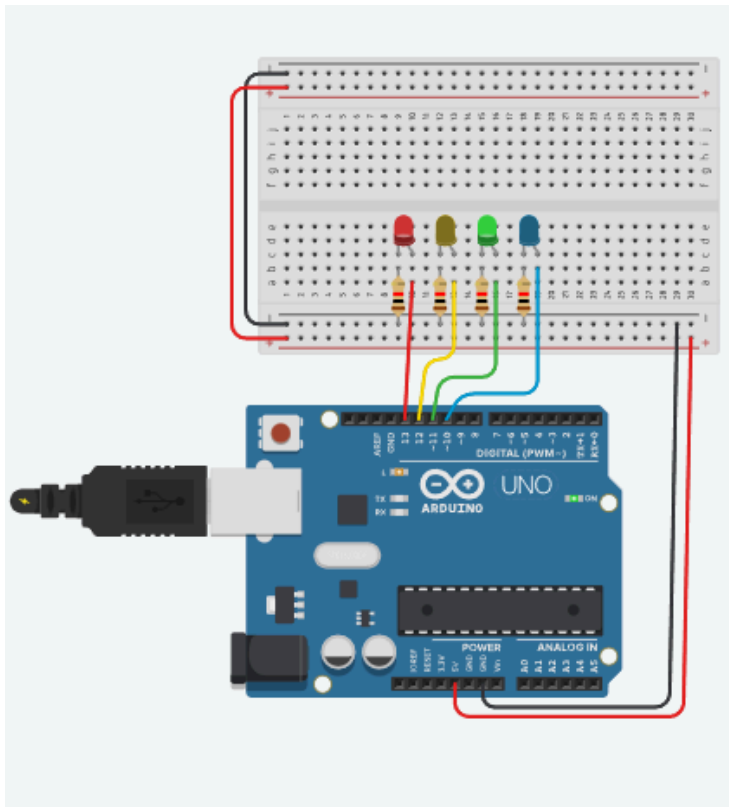


```
1 // Definição dos pinos dos LEDs
2 int ledVermelho = 13; // Representa A
3 int ledAmarelo = 12; // Representa B
4 int ledVerde = 11; // Representa resultado
5 int ledAzul = 10; // Representa resultado
6
7 // Variáveis para armazenar os valores de entrada
8 int a, b, opCode;
9 int saida, vail;
10
11 void setup() {
12   // Configuração dos pinos como saída
13   pinMode(ledVermelho, OUTPUT);
14   pinMode(ledAmarelo, OUTPUT);
15   pinMode(ledVerde, OUTPUT);
16   pinMode(ledAzul, OUTPUT);
17
18   // Inicializa a comunicação serial
19   Serial.begin(9600);
20 }
21
22 void loop() {
23   // Verifica se há dados disponíveis
24   if (Serial.available() >= 3) {
25     // Lê os três caracteres
26     a = Serial.read() - '0';
27     b = Serial.read() - '0';
28     opCode = Serial.read() - '0';
29   }
30 }
```

Monitor serial

Entrada: a = 1, b = 0, opCode = 1
Saída: 1, Vail: 0

SOMA(A,B);



```

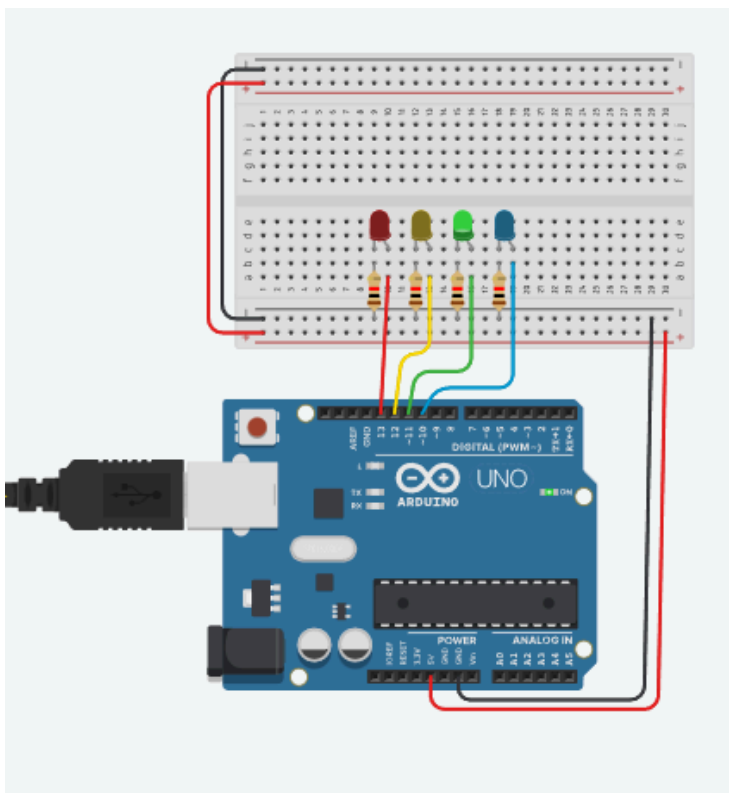
1 // Definição dos pinos dos LEDs
2 int ledVermelho = 13; // Representa o pino 13
3 int ledAmarelo = 12; // Representa o pino 12
4 int ledVerde = 11; // Representa o pino 11
5 int ledAzul = 10; // Representa o pino 10
6
7 // Variáveis para armazenar os dados de entrada
8 int a, b, opCode;
9 int saida, vail;
10
11 void setup() {
12     // Configuração dos pinos para saída
13     pinMode(ledVermelho, OUTPUT);
14     pinMode(ledAmarelo, OUTPUT);
15     pinMode(ledVerde, OUTPUT);
16     pinMode(ledAzul, OUTPUT);
17
18     // Inicializa a comunicação serial
19     Serial.begin(9600);
20 }
21
22 void loop() {
23     // Verifica se há dados de entrada
24     if (Serial.available() >= 3) {
25         // Lê os três caracteres de entrada
26         a = Serial.read() - '0';
27         b = Serial.read() - '0';
28         opCode = Serial.read() - '0';
29     }
30
31     // Realiza a soma
32     saida = a + b;
33     vail = 1;
34
35     // Acende o LED correspondente ao resultado
36     if (saida == 1) {
37         digitalWrite(ledVermelho, HIGH);
38     } else {
39         digitalWrite(ledVermelho, LOW);
40     }
41
42     // Acende o LED correspondente ao resultado
43     if (saida == 2) {
44         digitalWrite(ledAmarelo, HIGH);
45     } else {
46         digitalWrite(ledAmarelo, LOW);
47     }
48
49     // Acende o LED correspondente ao resultado
50     if (saida == 3) {
51         digitalWrite(ledVerde, HIGH);
52     } else {
53         digitalWrite(ledVerde, LOW);
54     }
55
56     // Acende o LED correspondente ao resultado
57     if (saida == 4) {
58         digitalWrite(ledAzul, HIGH);
59     } else {
60         digitalWrite(ledAzul, LOW);
61     }
62
63     // Espera 2 segundos
64     delay(2000);
65 }

```

Monitor serial

Entrada: a = 1, b = 0, opCode = 3
Saída: 1, Vail: 0

A=0; (ou A0)
NOT(A);



```

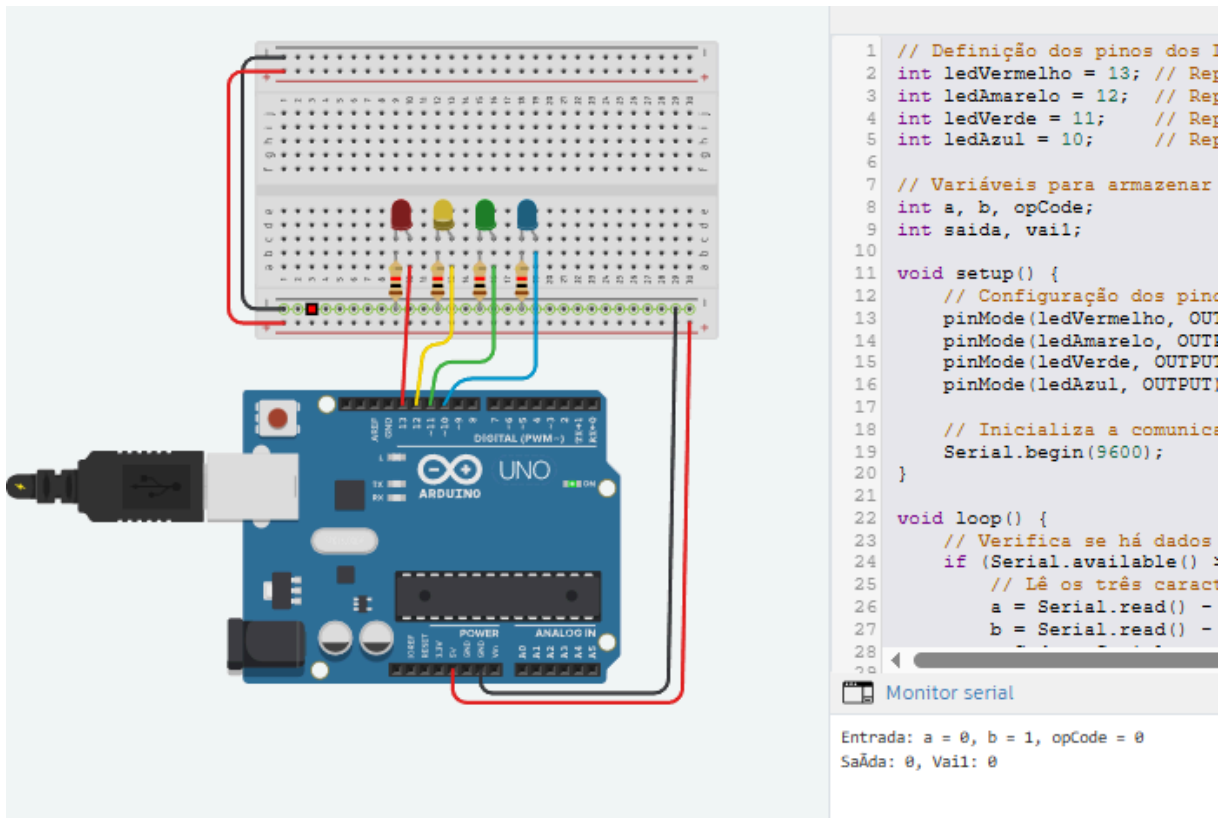
1 // Definição dos pinos dos LEDs
2 int ledVermelho = 13; // Representa o pino 13
3 int ledAmarelo = 12; // Representa o pino 12
4 int ledVerde = 11; // Representa o pino 11
5 int ledAzul = 10; // Representa o pino 10
6
7 // Variáveis para armazenar os dados de entrada
8 int a, b, opCode;
9 int saida, vail;
10
11 void setup() {
12     // Configuração dos pinos para saída
13     pinMode(ledVermelho, OUTPUT);
14     pinMode(ledAmarelo, OUTPUT);
15     pinMode(ledVerde, OUTPUT);
16     pinMode(ledAzul, OUTPUT);
17
18     // Inicializa a comunicação serial
19     Serial.begin(9600);
20 }
21
22 void loop() {
23     // Verifica se há dados de entrada
24     if (Serial.available() >= 3) {
25         // Lê os três caracteres de entrada
26         a = Serial.read() - '0';
27         b = Serial.read() - '0';
28         opCode = Serial.read() - '0';
29     }
30
31     // Realiza a operação NOT
32     saida = 1 - a;
33     vail = 1;
34
35     // Acende o LED correspondente ao resultado
36     if (saida == 1) {
37         digitalWrite(ledVermelho, HIGH);
38     } else {
39         digitalWrite(ledVermelho, LOW);
40     }
41
42     // Acende o LED correspondente ao resultado
43     if (saida == 2) {
44         digitalWrite(ledAmarelo, HIGH);
45     } else {
46         digitalWrite(ledAmarelo, LOW);
47     }
48
49     // Acende o LED correspondente ao resultado
50     if (saida == 3) {
51         digitalWrite(ledVerde, HIGH);
52     } else {
53         digitalWrite(ledVerde, LOW);
54     }
55
56     // Acende o LED correspondente ao resultado
57     if (saida == 4) {
58         digitalWrite(ledAzul, HIGH);
59     } else {
60         digitalWrite(ledAzul, LOW);
61     }
62
63     // Espera 2 segundos
64     delay(2000);
65 }

```

Monitor serial

Entrada: a = 0, b = 0, opCode = 2
Saída: 1, Vail: 0

B=1; (ou B=1)
AND(B,A);



Instrução realizada	Binário (A,B,Op.code)	Valor em Hexa (0x ...)	Resultado em binário
AND(A,B)	0 1 00	4	0
OR(A,B)	1 0 01	9	1
SOMA(A,B)	1 0 11	B	1
NOT(A)	0 0 10	2	1
AND(B,A)	1 0 00	8	0