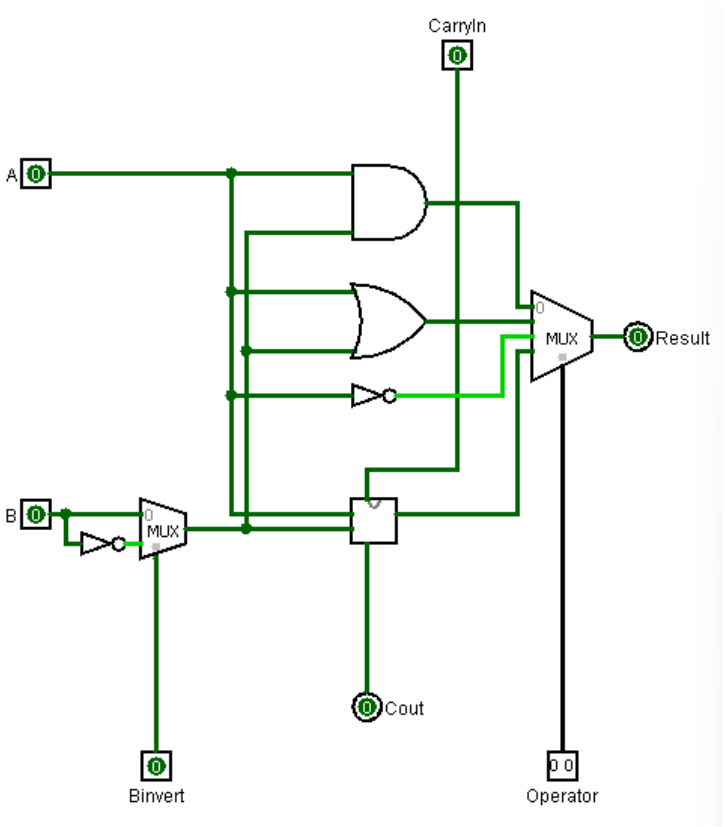
EXERCÍCIO PRÁTICO 2

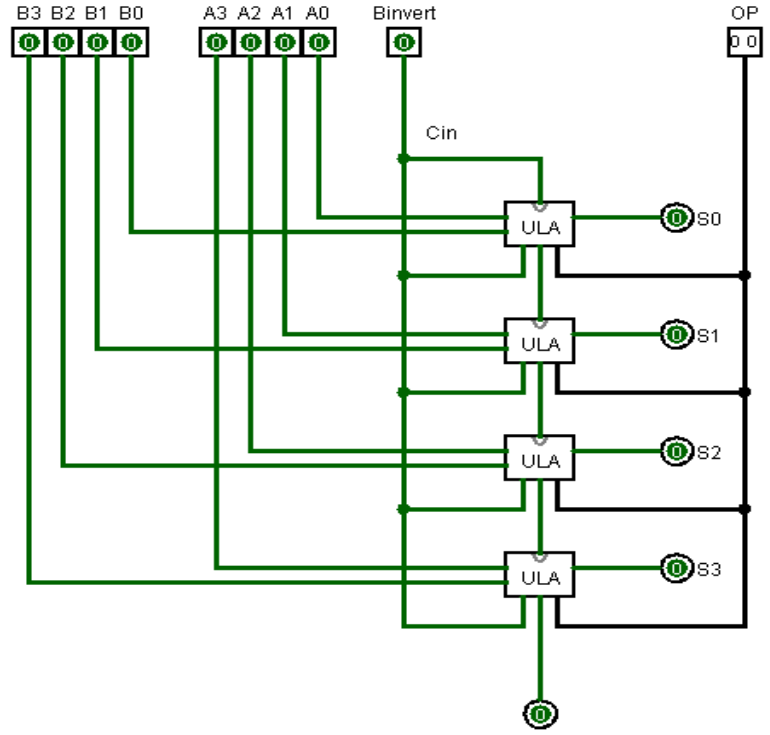
ALUNO: GUSTAVO LORENZO CAMPOS

PARTE 1)

-PRINT DA ULA DE 1 BIT:



-PRINT DA ULA DE 4 BITS:

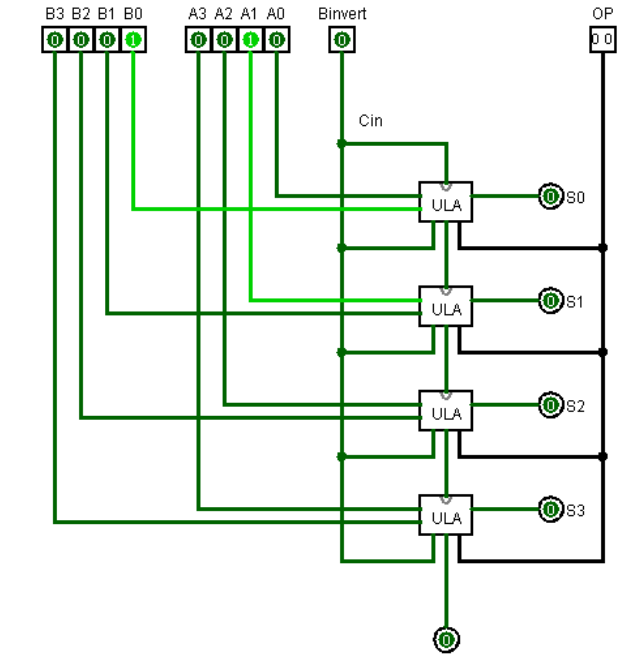


-TABELA DE RESULTADOS PARA A UNIDADE DE 4 BITS:

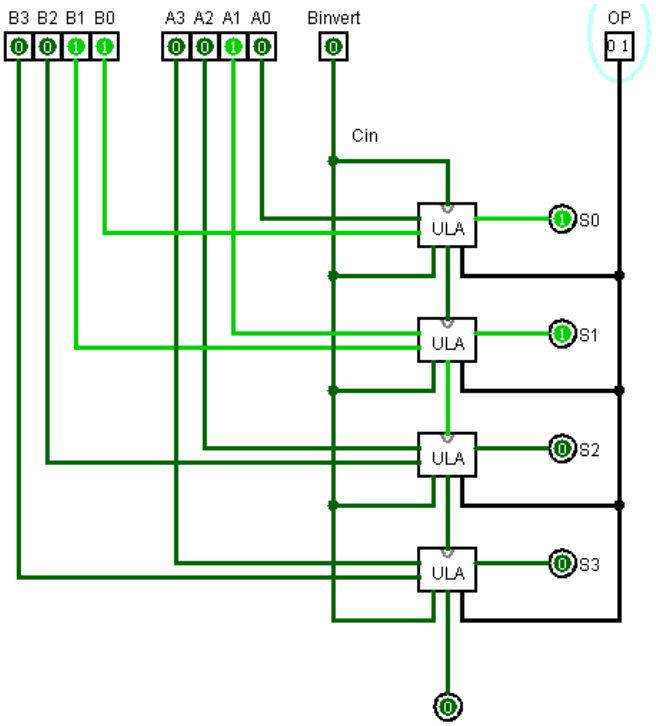
Instrução Realizada	**Binário (A,B,Op.code)**	**Valor em Hexa**	**Resultado em Binário**
AND(A,B)	0010 0001 00	(0000 1000 0100)0x084	0000
OR(A,B)	0010 0011 01	(0000 1000 1101)0x08D	0011
SOMA(A,B)	0010 0011 11	(0000 1000 1111)0x08F	0101
NOT(A)	1100 0011 10	(0011 0000 1110)0x30E	0011
AND(B,A)	1100 1101 00	(0011 0011 0100)0x334	1100

-UM PRINT PARA CADA UM DOS TESTES REALIZADOS:

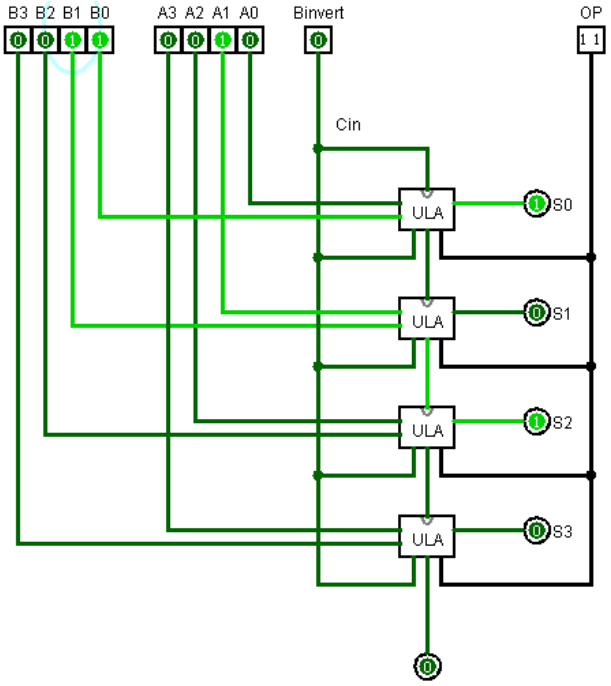
AND(A,B):


B3 B2 B1 B0
A3 A2 A1 A0
Binvert
OP
0 0
Cin
ULA
S0
ULA
S1
ULA
S2
ULA
S3

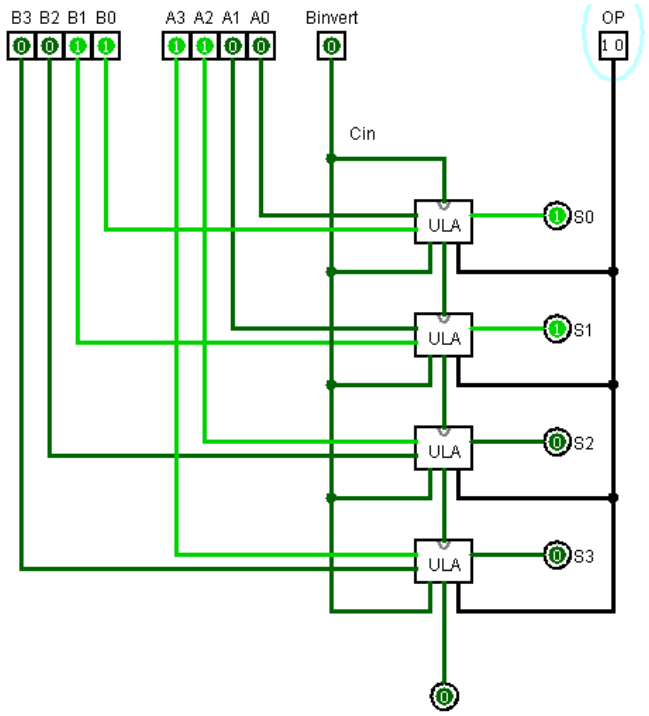
OR(A,B):


B3 B2 B1 B0
A3 A2 A1 A0
Binvert
OP
0 1
Cin
ULA
S0
ULA
S1
ULA
S2
ULA
S3

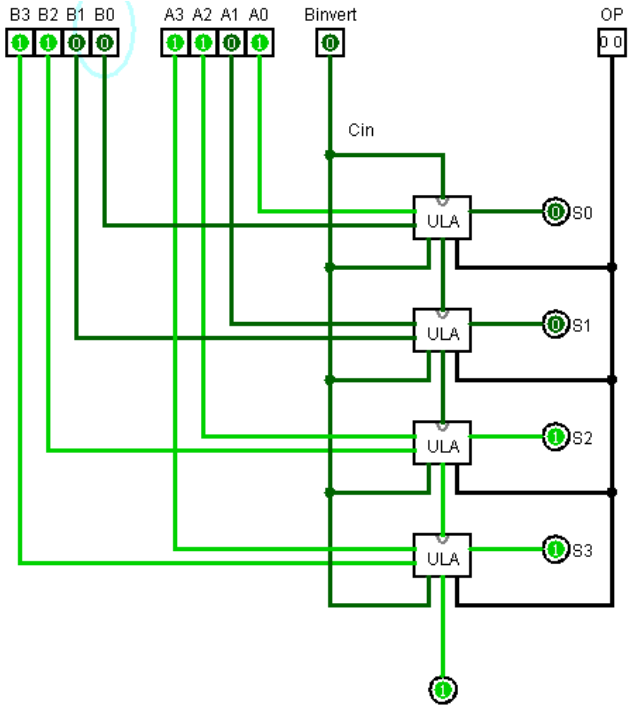
SOMA(A,B):


B3 B2 B1 B0
A3 A2 A1 A0
Binvert
OP
1 1
Cin
ULA
S0
ULA
S1
ULA
S2
ULA
S3

NOT(A):

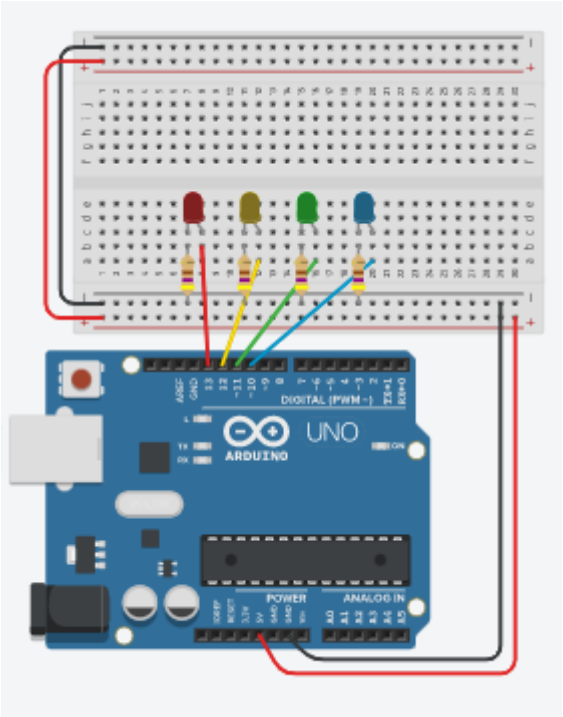

B3 B2 B1 B0
A3 A2 A1 A0
Binvert
OP
1 0
Cin
ULA
S0
ULA
S1
ULA
S2
ULA
S3

AND(B,A):



PARTE 2)

-PRINT DA TELA DO TINKERCAD MOSTRANDO A MONTAGEM E O PROGRAMA:
EXERCÍCIO 1:



```
int led12 = 12;
int led13 = 13;
int x = 0;


// Rotina executada 1 vez e que em geral configura entradas e sa
void setup() {
  // configura os pinos como saídas DIGITAIS.
  pinMode(led10, OUTPUT);
  pinMode(led11, OUTPUT);
  pinMode(led12, OUTPUT);
  pinMode(led13, OUTPUT);

}

// the loop routine runs over and over again forever:
void loop() {
  if(x==0){
  digitalWrite(led12, LOW);   // Faz a saída do respectivo Led s
  digitalWrite(led13, HIGH);   // Faz a saída do respectivo Led
  }else if(x==3){
  digitalWrite(led13, LOW);   // Faz a saída do respectivo Led s
  digitalWrite(led11, HIGH);   // Faz a saída do respectivo Led
  }else if(x==7){
  digitalWrite(led11, LOW);   // Faz a saída do respectivo Led s
  digitalWrite(led12, HIGH);   // Faz a saída do respectivo Led
  }else if(x==8){
  x=-1;
  }
  digitalWrite(led10, HIGH);   // Faz a saída do respectivo Led
  delay(1000);                // espera por 100 ms
  digitalWrite(led10, LOW);   // Faz a saída do respectivo Led s
  delay(100);                // espera por 100 ms
  x++;
}
```

EXERCÍCIO 2:

```
int a; //a
int b; //b
int op; //operacao
char inputBuffer[3];
int blueLed = 10;
int redLed = 13;
int greenLed = 11;
int yellowLed = 12;

int saida;

void setup() {
  Serial.begin(9600);
  pinMode(redLed, OUTPUT);
  pinMode(blueLed, OUTPUT);
  pinMode(greenLed, OUTPUT);
  pinMode(yellowLed, OUTPUT);
}

void loop() {

  if (Serial.available() > 0) {
    String inputString = "";
    while (inputString.length() < 3) {
      if (Serial.available()) {
        char c = Serial.read();
        if (c >= '0' && c <= '9') {
          inputString += c;
        }
      }
    }

    a = inputString.charAt(0) - '0';
    b = inputString.charAt(1) - '0';
    op = inputString.charAt(2) - '0';
```

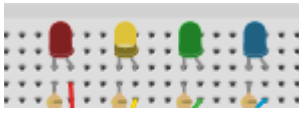
```
36
37      //Serial.print("a: ");
38      //Serial.println(a);
39      //Serial.print("b: ");
40      //Serial.println(b);
41      //Serial.print("op: ");
42      //Serial.println(op);
43      digitalWrite(blueLed, 0);
44      digitalWrite(redLed, a);
45      digitalWrite(yellowLed, b);
46      // Executa a operação correspondente
47      switch (op) {
48      case 0: // Operação AND
49        saida = portaand(a, b);
50        //Serial.print("and= ");
51        //Serial.print(saida);
52        //Serial.println();
53        break;
54      case 1: // Operação OR
55        saida = portaor(a, b);
56        //Serial.print("or= ");
57        //Serial.print(saida);
58        //Serial.println();
59        break;
60      case 2: // Operação NOT
61        saida = portanot(a);
62        //Serial.print("not entrada1= ");
63        //Serial.print(saida);
64        //Serial.println();
65        break;
66      case 3:
67        saida = portaxor(a, b);
68        //Serial.print("xor = ");
69        //Serial.print(saida);
70        //Serial.println();
71        break;
72      default:
73        Serial.println("Operação inválida!"); // Mensagem de erro
74      }
75      mostra(saida);
76    }
77
78  }
79
80  int portaxor(int a, int b) {
81    if (a == 1 && b == 1) {
82      digitalWrite(blueLed, 1);
83    }
84    return (a ^ b);
85  }
86
87  int portaor(int a, int b) {
88    return (a | b);
89  }
90
91  int portaand(int a, int b) {
92    return (a & b);
93  }
94
95  int portanot(int a) {
96    if (a == 1){
97      return 0;
98    }
99    else{
100     return 1;
101   }
102 }
103
104 void mostra(int a) {
105   if (a == 1) {
```

```
106       digitalWrite(greenLed, 1);
107     } else {
108       digitalWrite(greenLed, 0);
109     }
110     return;
111   }
```

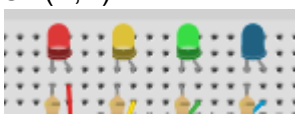
-RESULTADO DAS INSTRUÇÕES EXECUTADAS:

Instrução Realizada	Binário (A,B,Op.code)	Resultado	Vai 1
AND(A,B)	0 1 0	0	0
OR(A,B)	1 1 1	1	0
SOMA(A,B)	1 1 3	0	1
NOT(A)	1 1 2	0	0
AND(A,-B)	1 -1 3	0	0

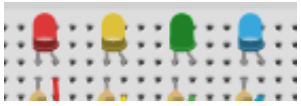
AND(A,B):



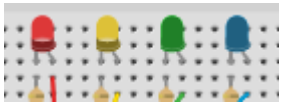
OR(A,B):



SOMA(A,B):



NOT(A):



SOMA(A,-B):