Pontifícia Universidade Católica de Minas Gerais
Aluna: Rayssa Mell de Souza Silva

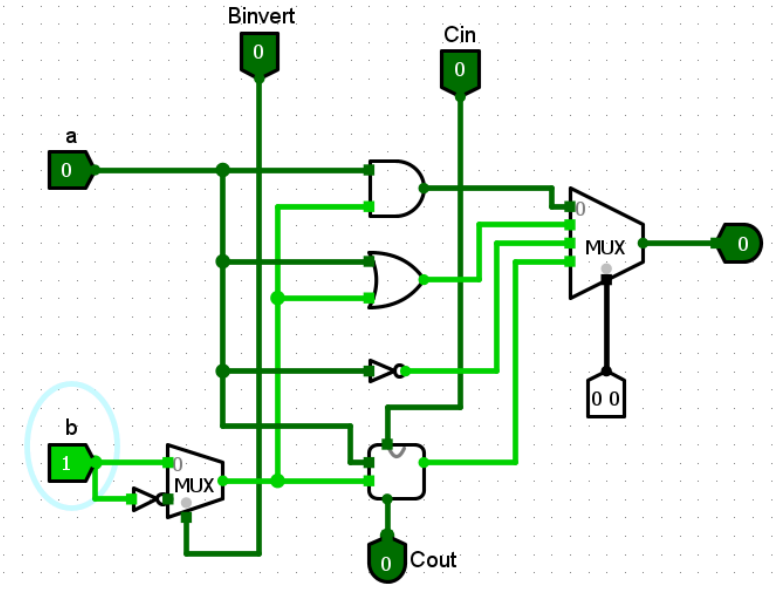
Exercício Prático 02

Parte 1 =>

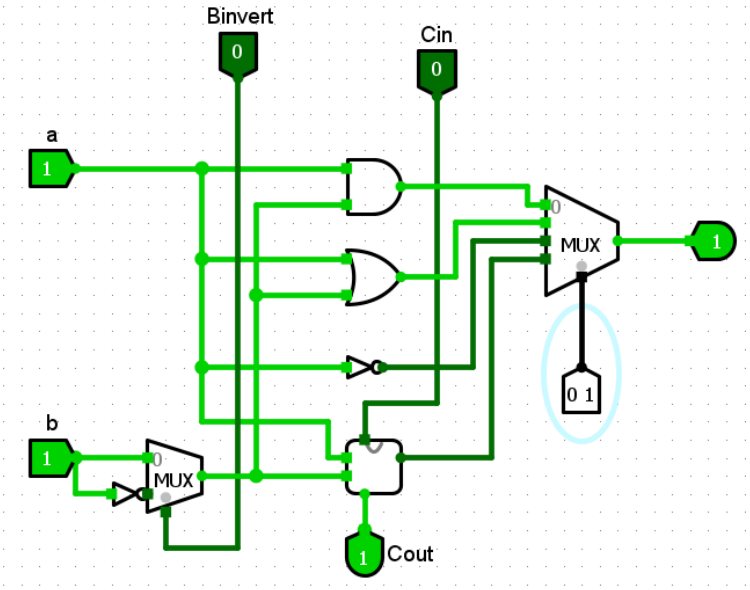
- **ULA de 1 bit:**

Operações:

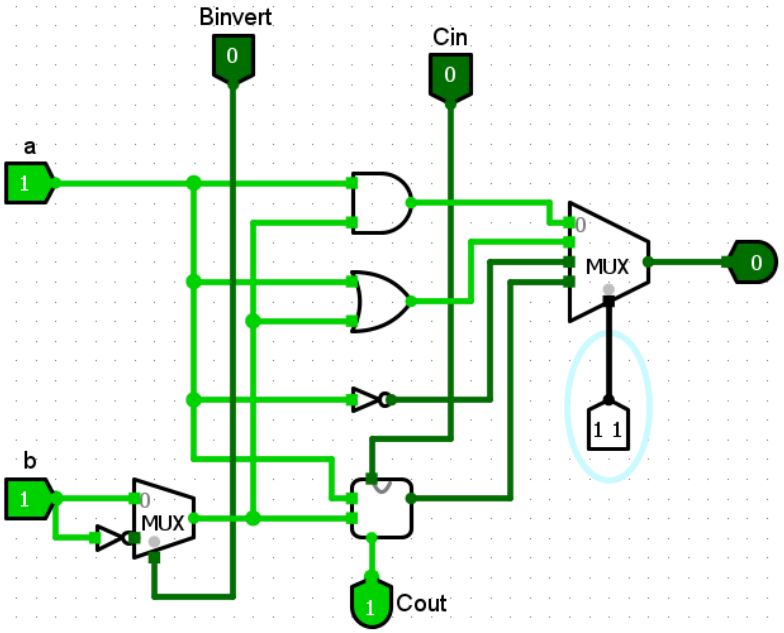
AND(A,B)



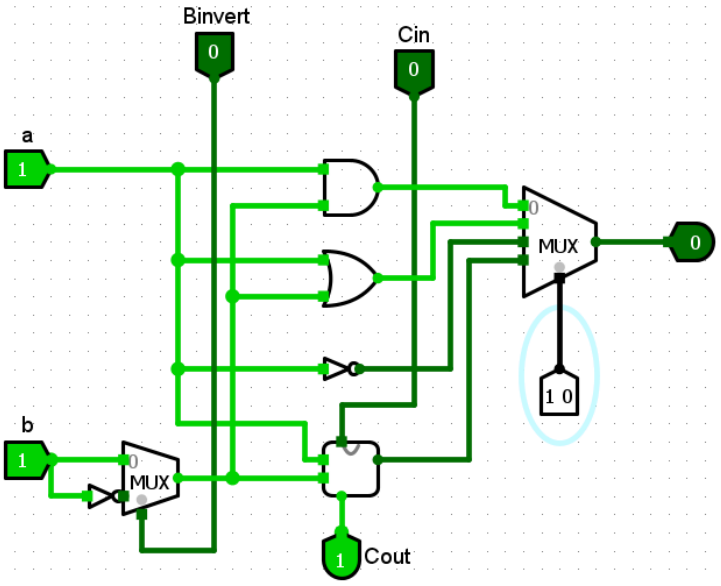
OR(A,B)



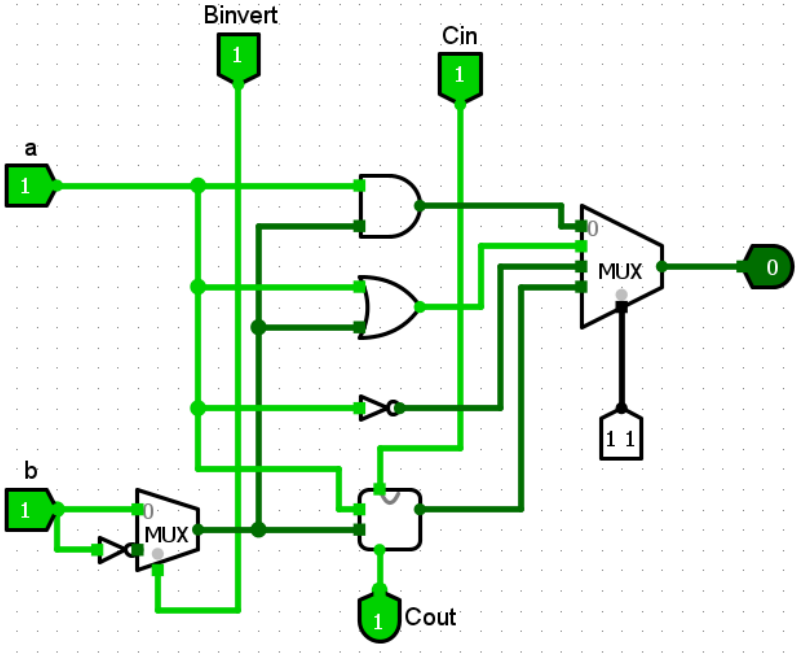
SOMA(A,B)


Binvert
0
Cin
0
a
1
MUX
0
1 1
b
1
MUX
1
Cout

NOT(A)


Binvert
0
Cin
0
a
1
MUX
0
1 0
b
1
MUX
1
Cout

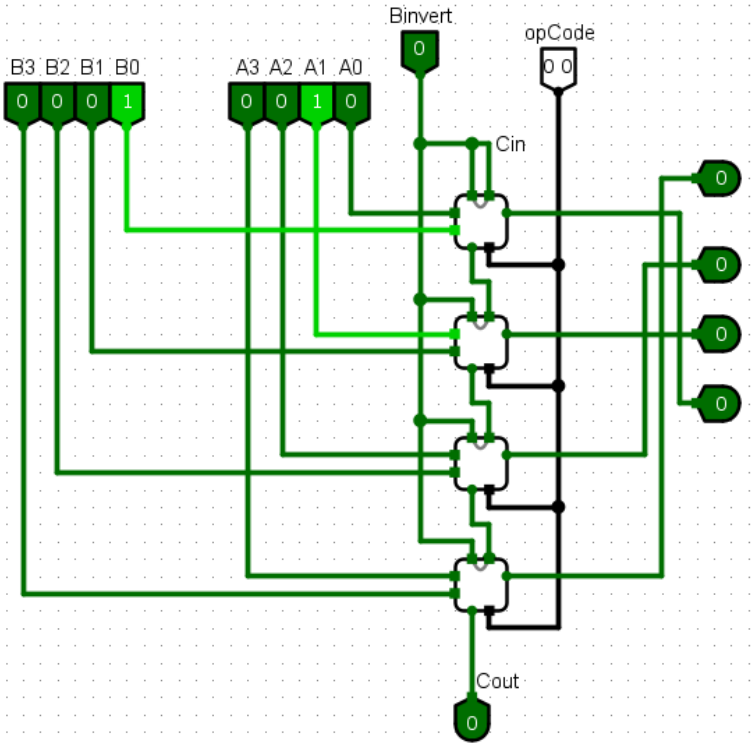
SOMA(A, -B)


Binvert
1
Cin
1
a
1
MUX
0
1 1
b
1
MUX
1
Cout

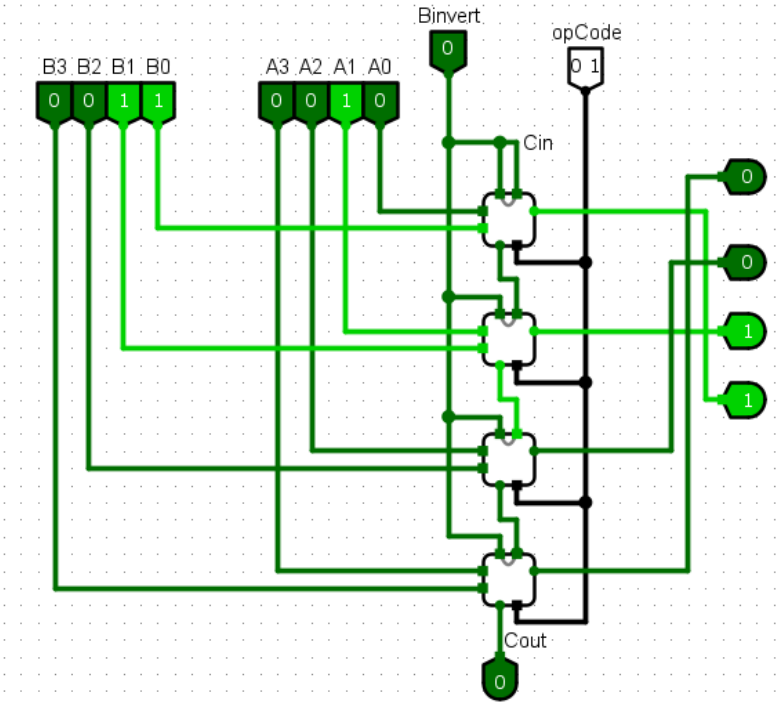
- **ULA para 4 bits:**

Operações:

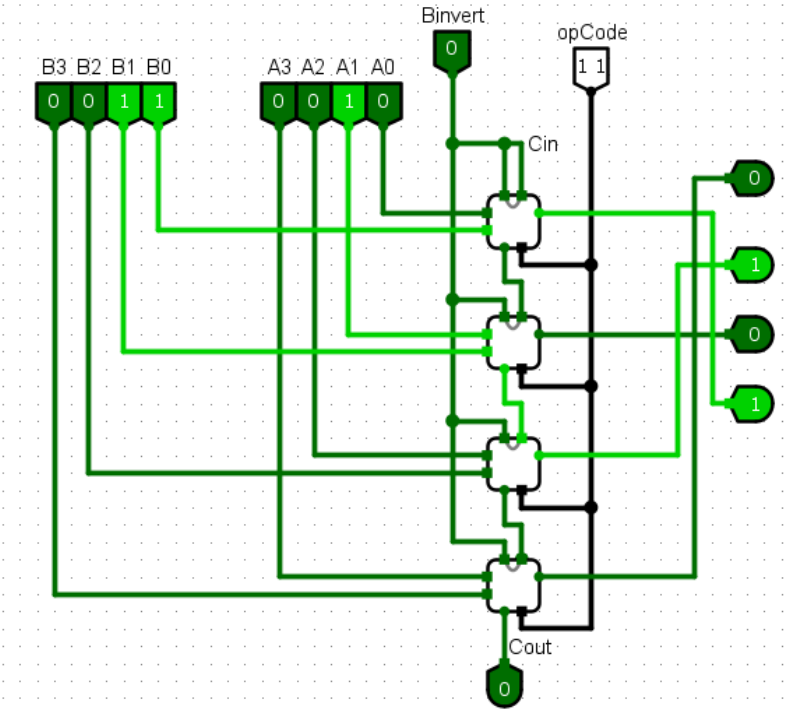
AND(A,B)



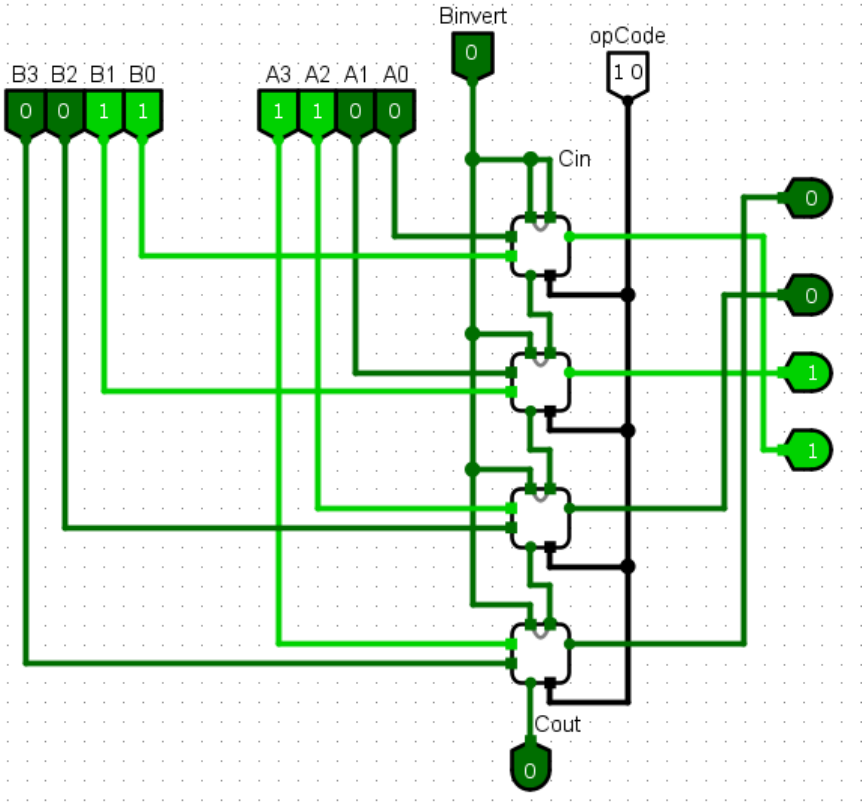
OR(A,B)



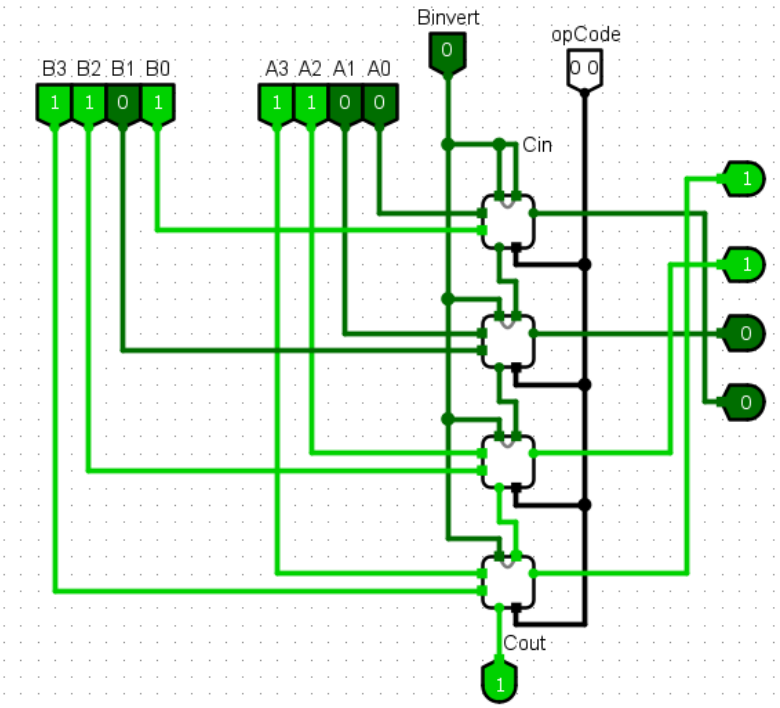
SOMA(A,B)


Binvert
opCode
B3 B2 B1 B0
0 0 1 1
A3 A2 A1 A0
0 0 1 0
1 1
0
Cin
0
1
0
1
Cout
0

NOT(A)


Binvert
opCode
B3 B2 B1 B0
0 0 1 1
A3 A2 A1 A0
1 1 0 0
1 0
0
Cin
0
0
1
1
Cout
0

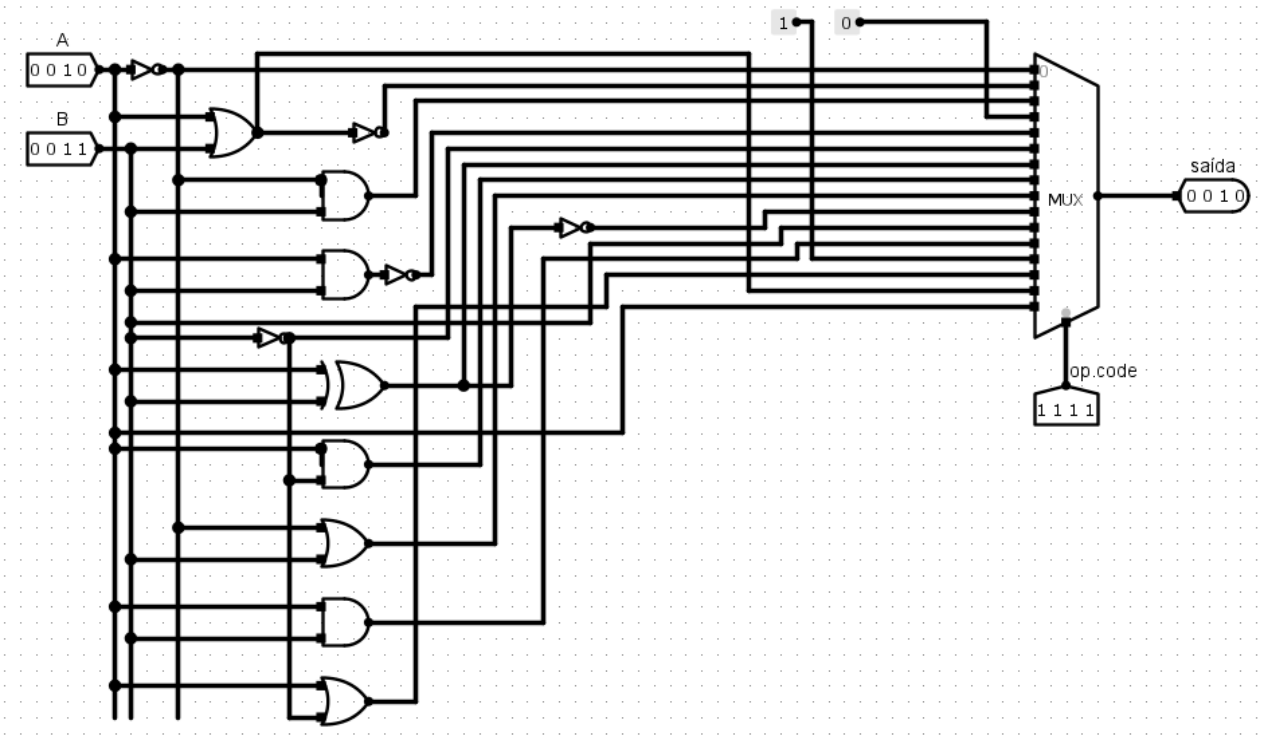
AND(A,B)



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 0111) = 0x337	1100

Parte 2 =>

ULA
Instrução: 4CB



Instrução	Binário	Resultado da Operação
450	010001010000	B
CB1	110010110001	0
A32	101000110010	1
C43	110101000011	0
124	000100100100	F
785	011110000101	7
9B6	100110110110	2
CD7	110011010111	0
FE8	111111101000	E
649	011001001001	D
D9A	110110011010	9

FCB	111111001011	C
63C	011000111100	1
98D	100110001101	1
76E	011101101110	F
23F	001000111111	2

- Se o objetivo fosse realmente testar esta ULA, quantas linhas a nossa tabela verdade deveria ter, ou seja na verdade a tabela que você preencheu deveria ter quantas linhas?

R: A tabela verdade desta ULA teria 256 linhas, pois 2^4 = 256.