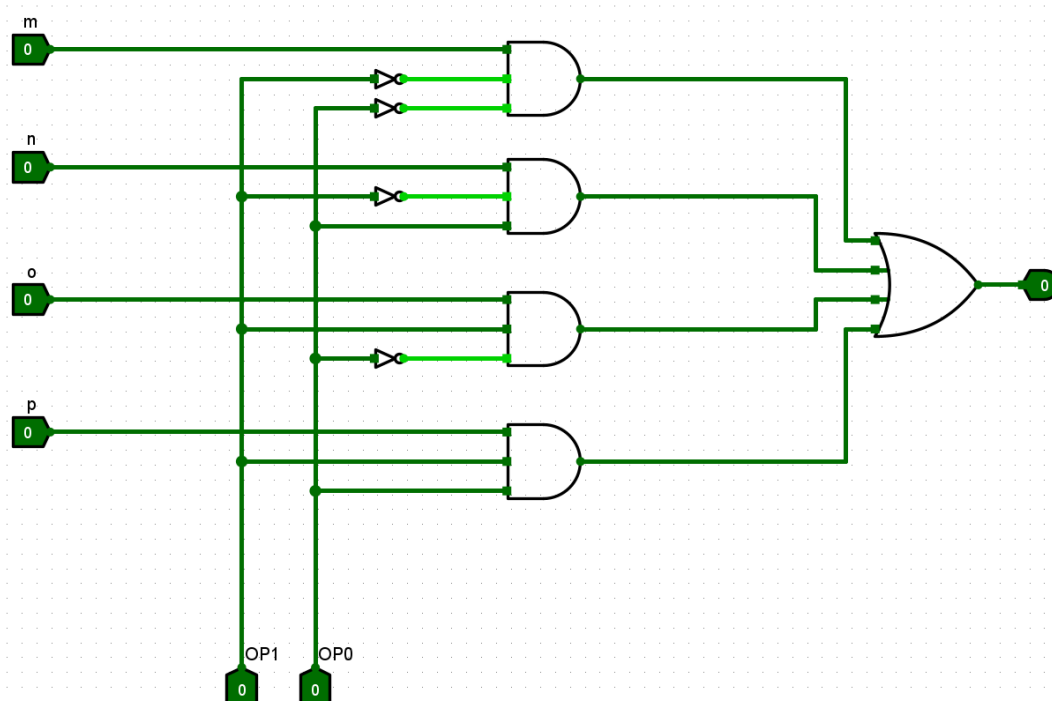
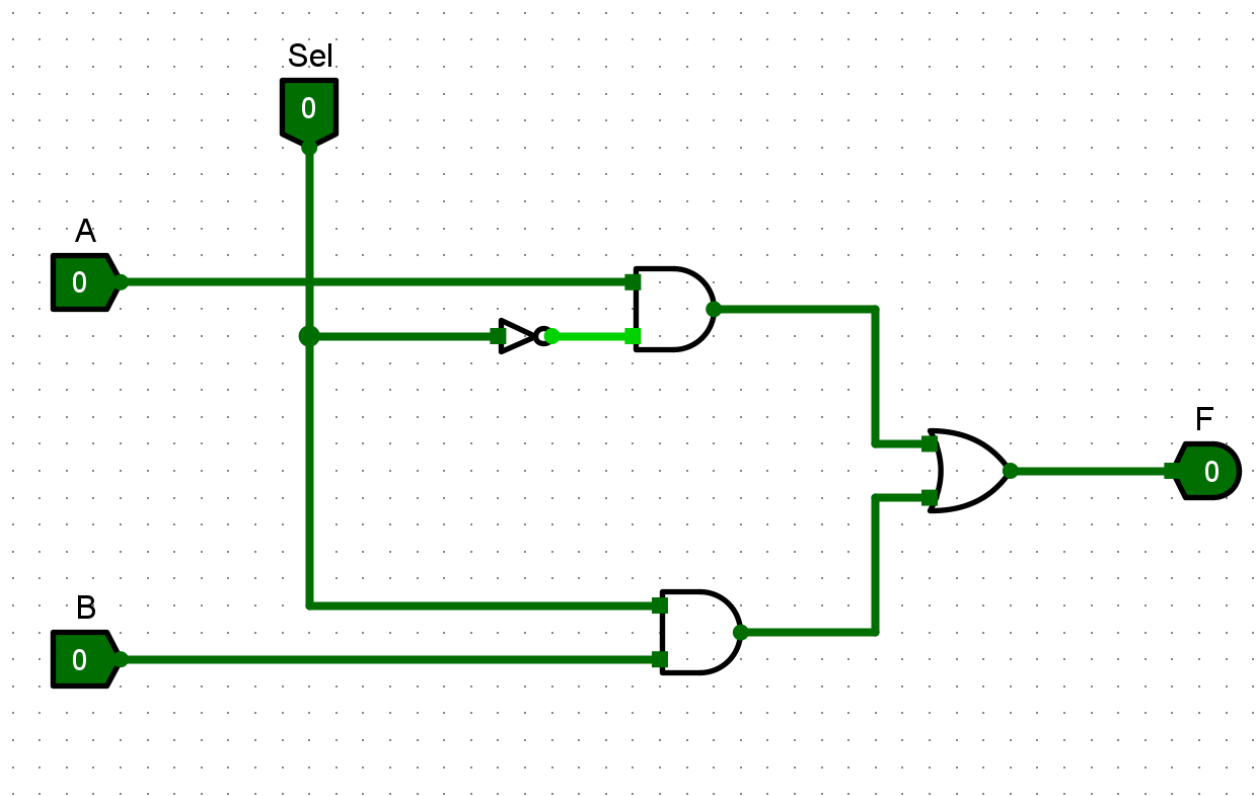


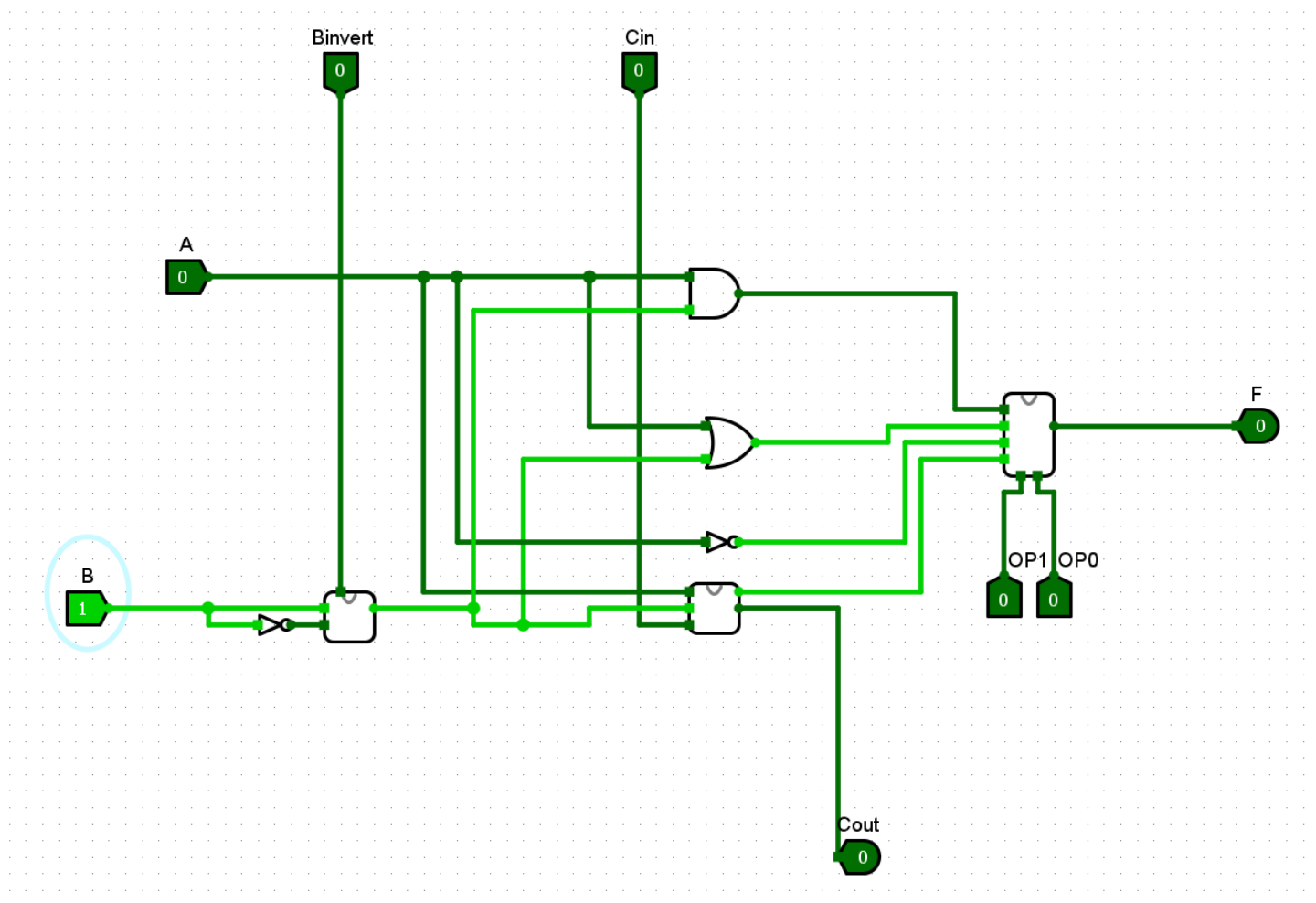
**Matéria:** Exercício Prático 01, Prof. Romanelli – PUC MINAS MG



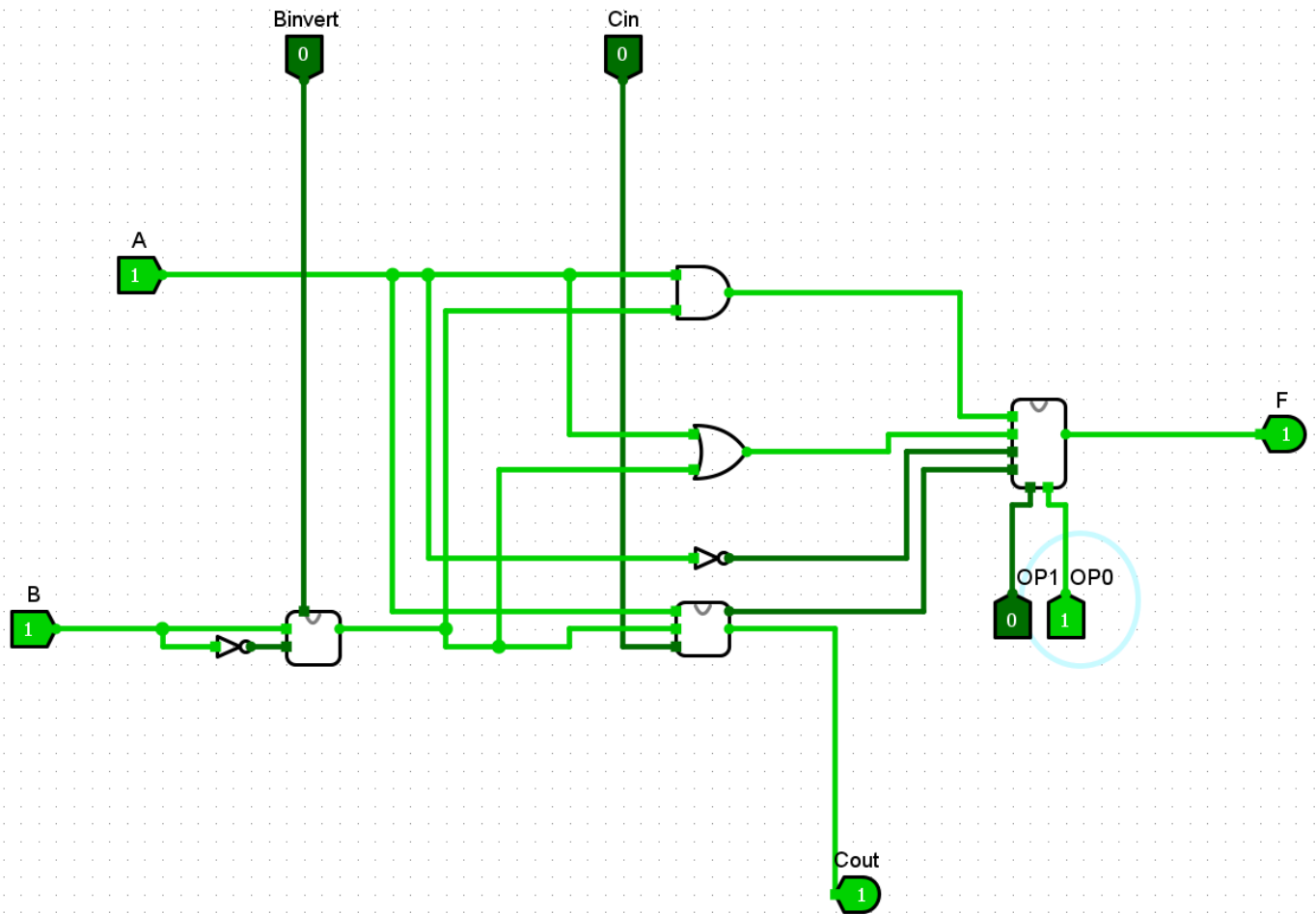


4) Teste a sua ULA de acordo com o seguinte roteiro:

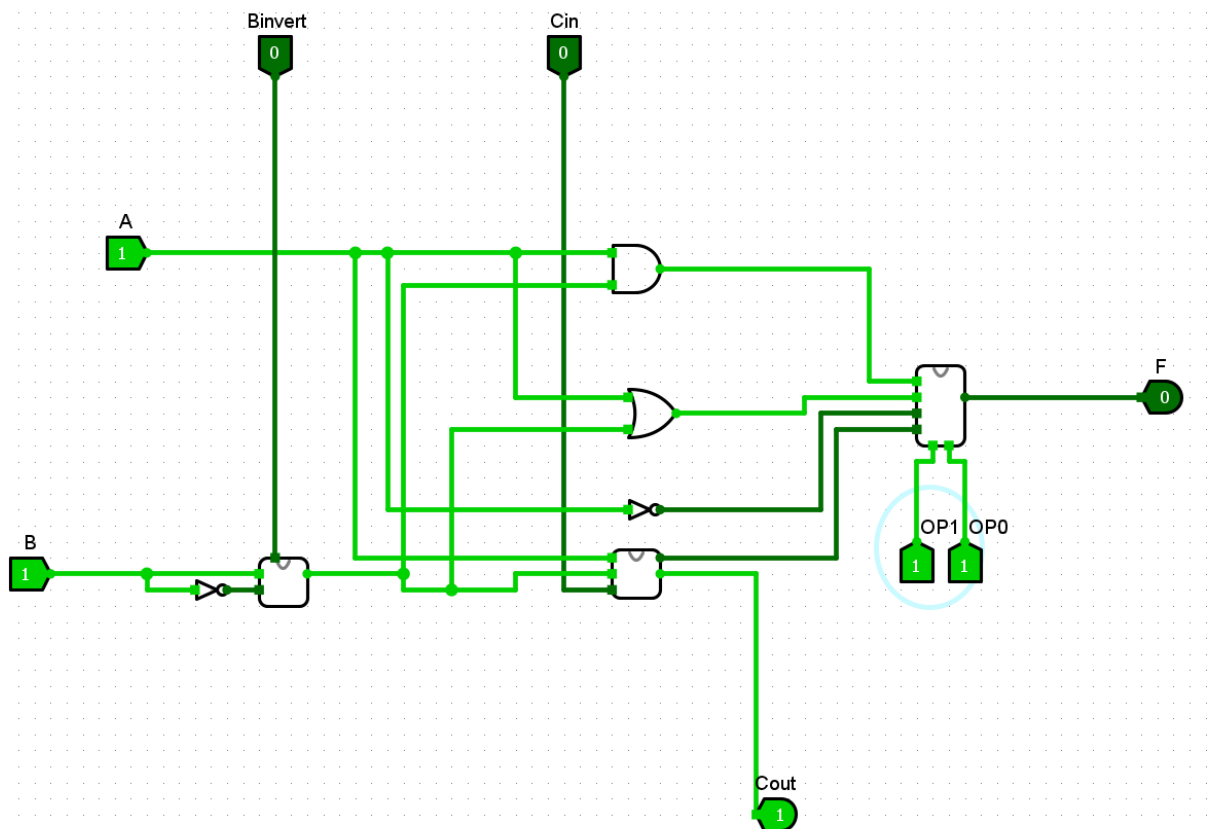
A) A=0; B=1;  
AND(A,B);



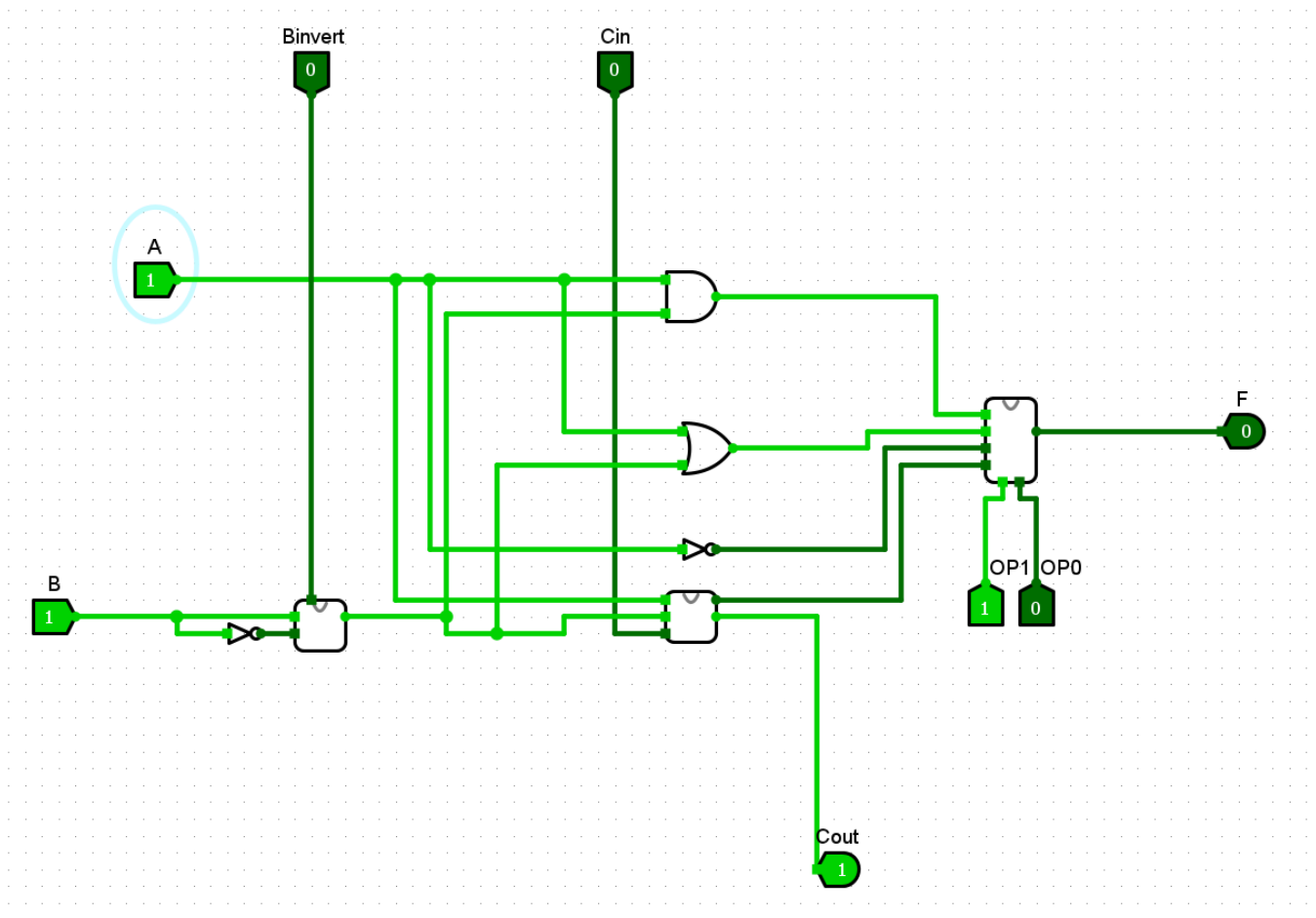
B) A=1; B=1;  
OR(A,B);



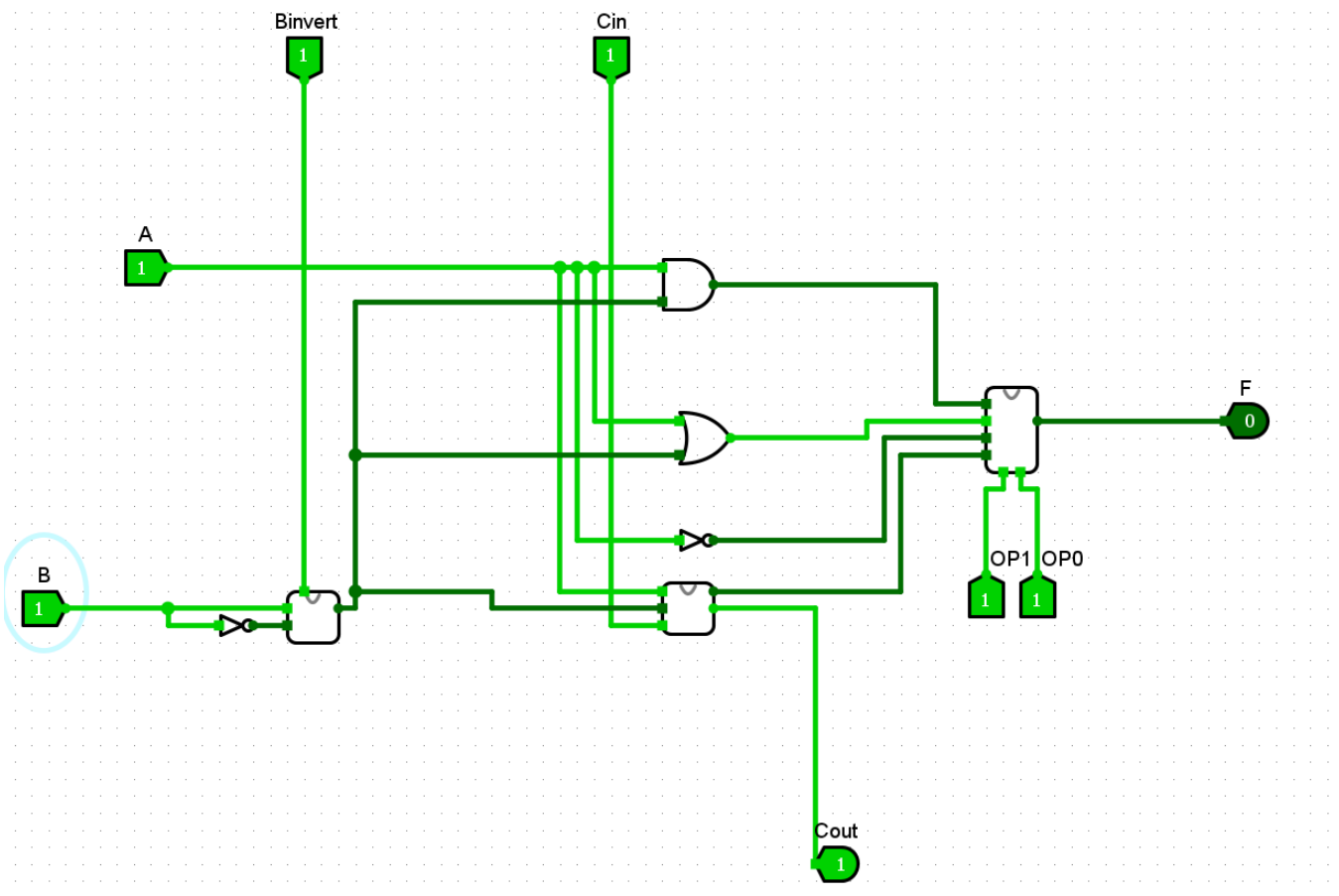
C) SOMA(A,B);



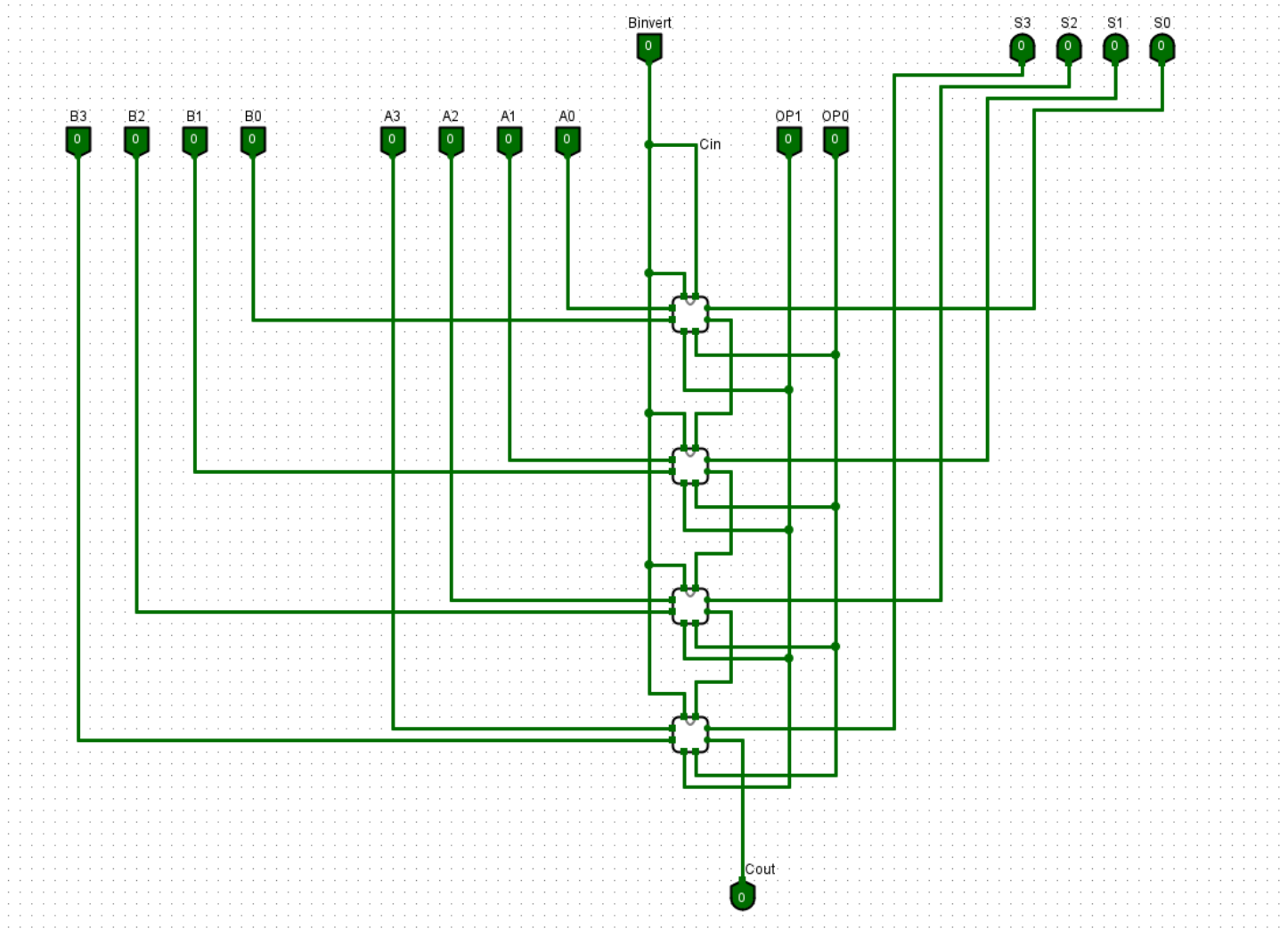
D) NOT(A);



E) SOMA (A,-B);



5) Usando essa ULA de 1 bit, construa essa ULA para 4 bits no Logisim e verifique o seu funcionamento. Veja como funciona o barramento de instruções (operation) e o barramento de dados (a e b). Observe a ligação do Binvert ao Carry\_in da primeira ULA.

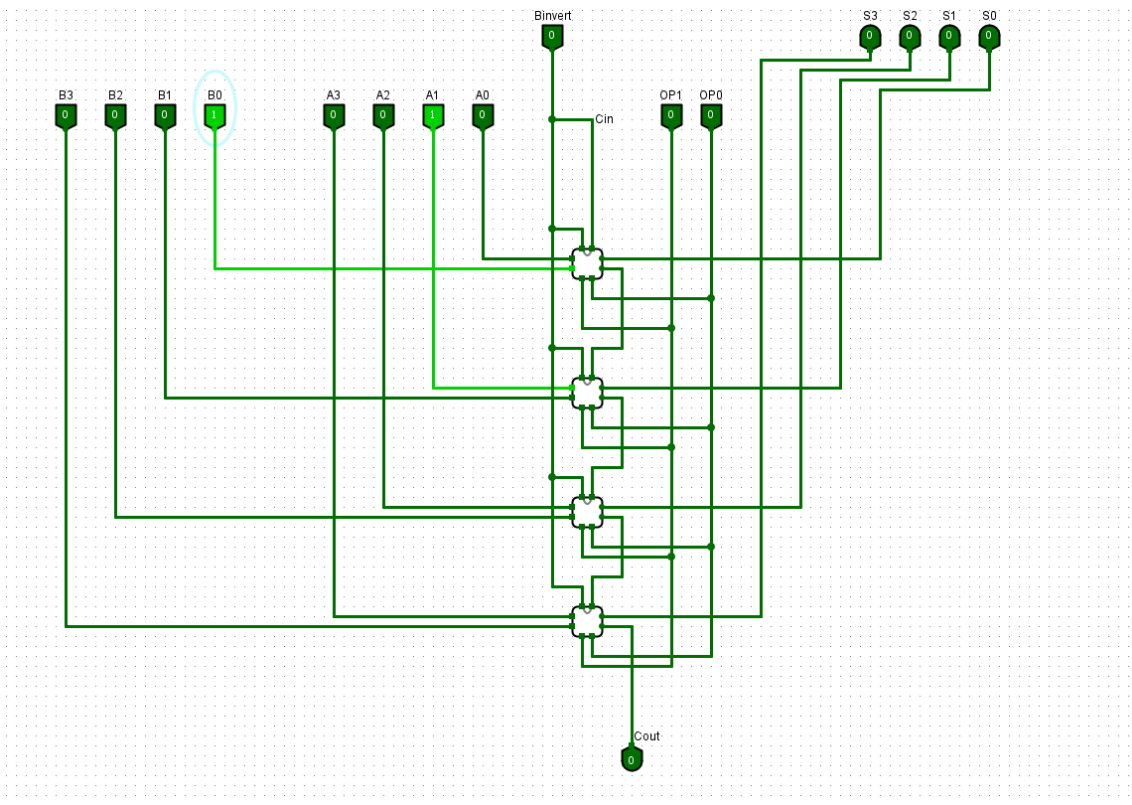


6) Teste a sua ULA de acordo com o seguinte roteiro (considerando os números de 4 bits):

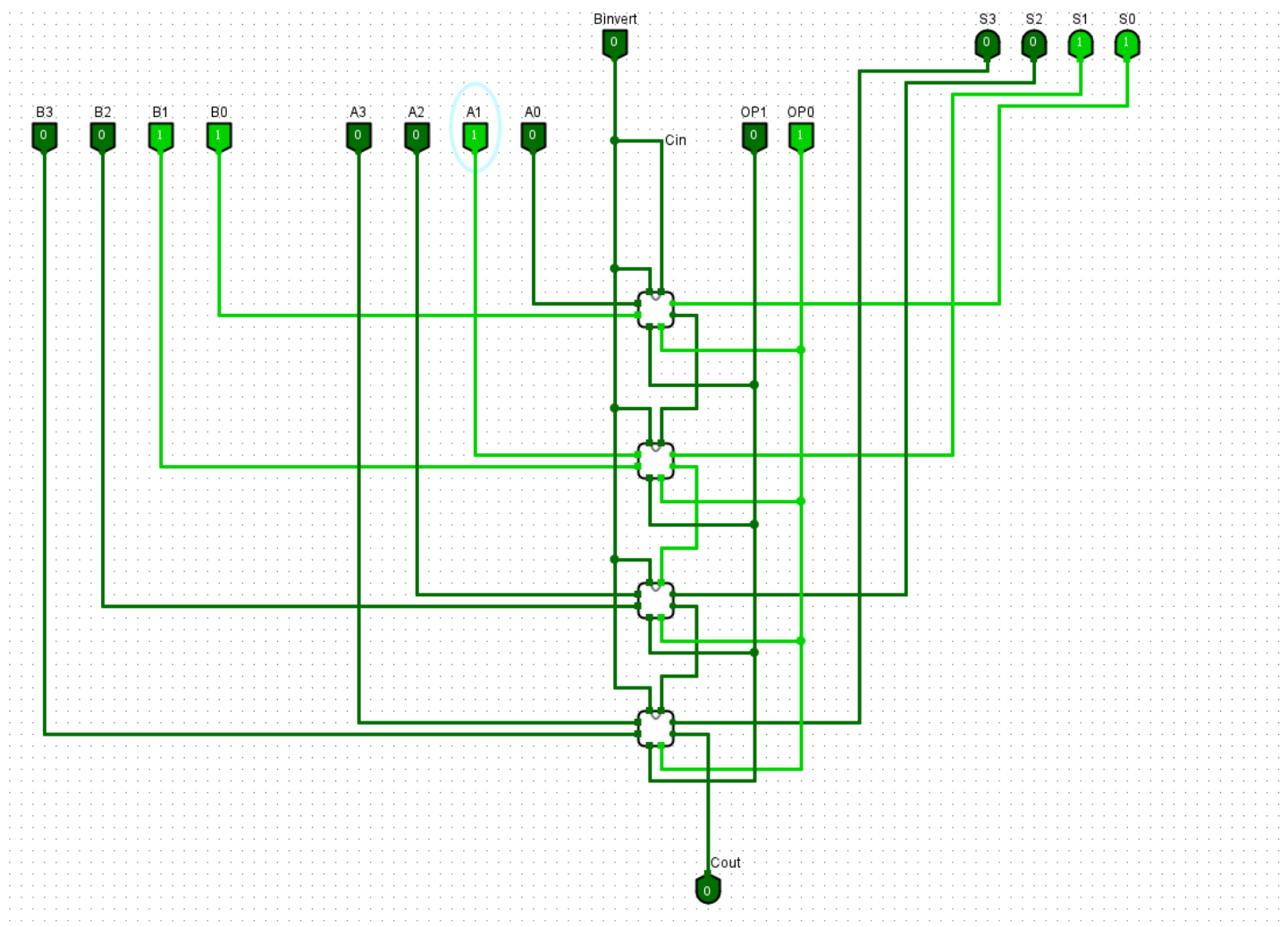
a) A=2; (ou A=0010)

B=1; ( ou B=0001)

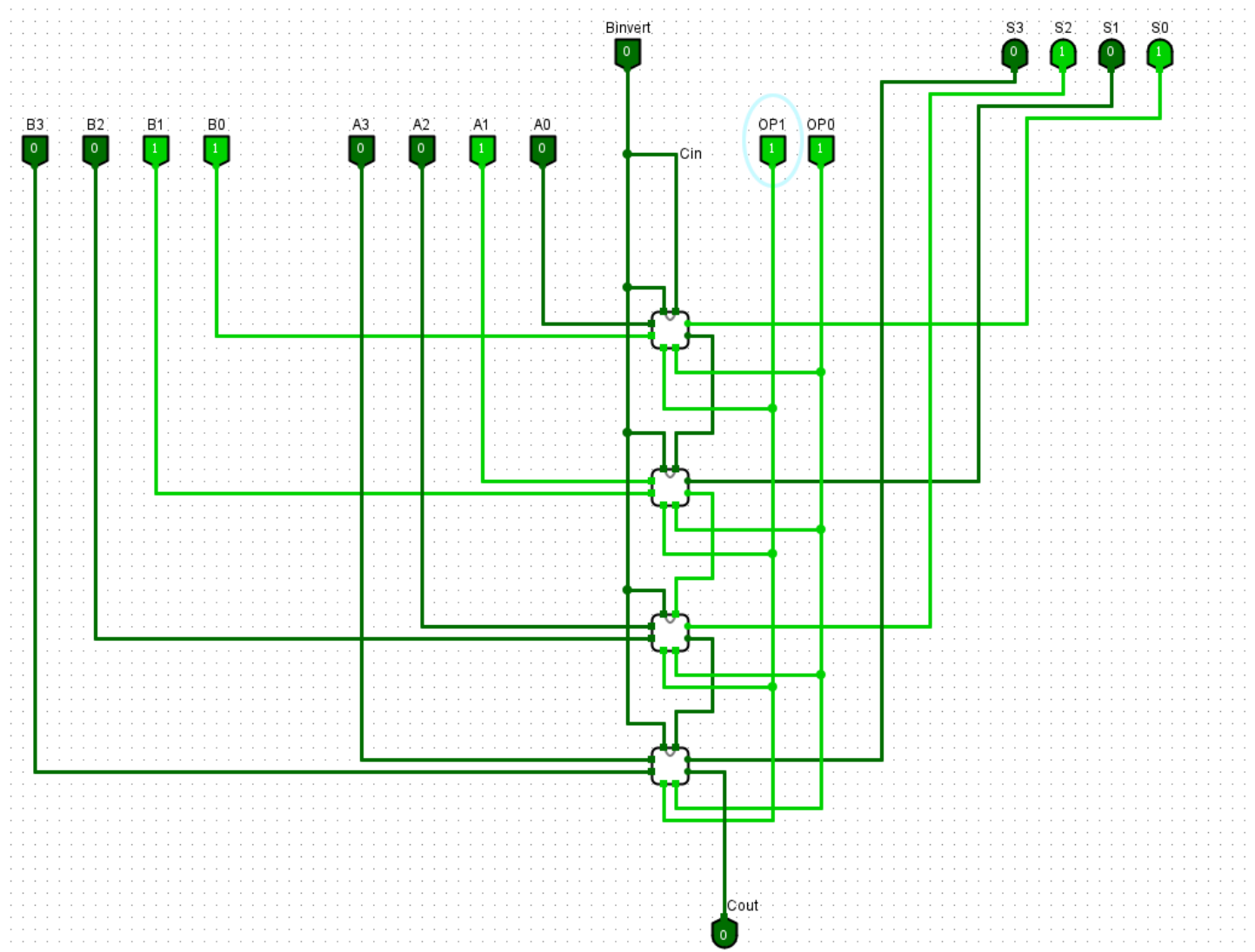
AND(A,B);



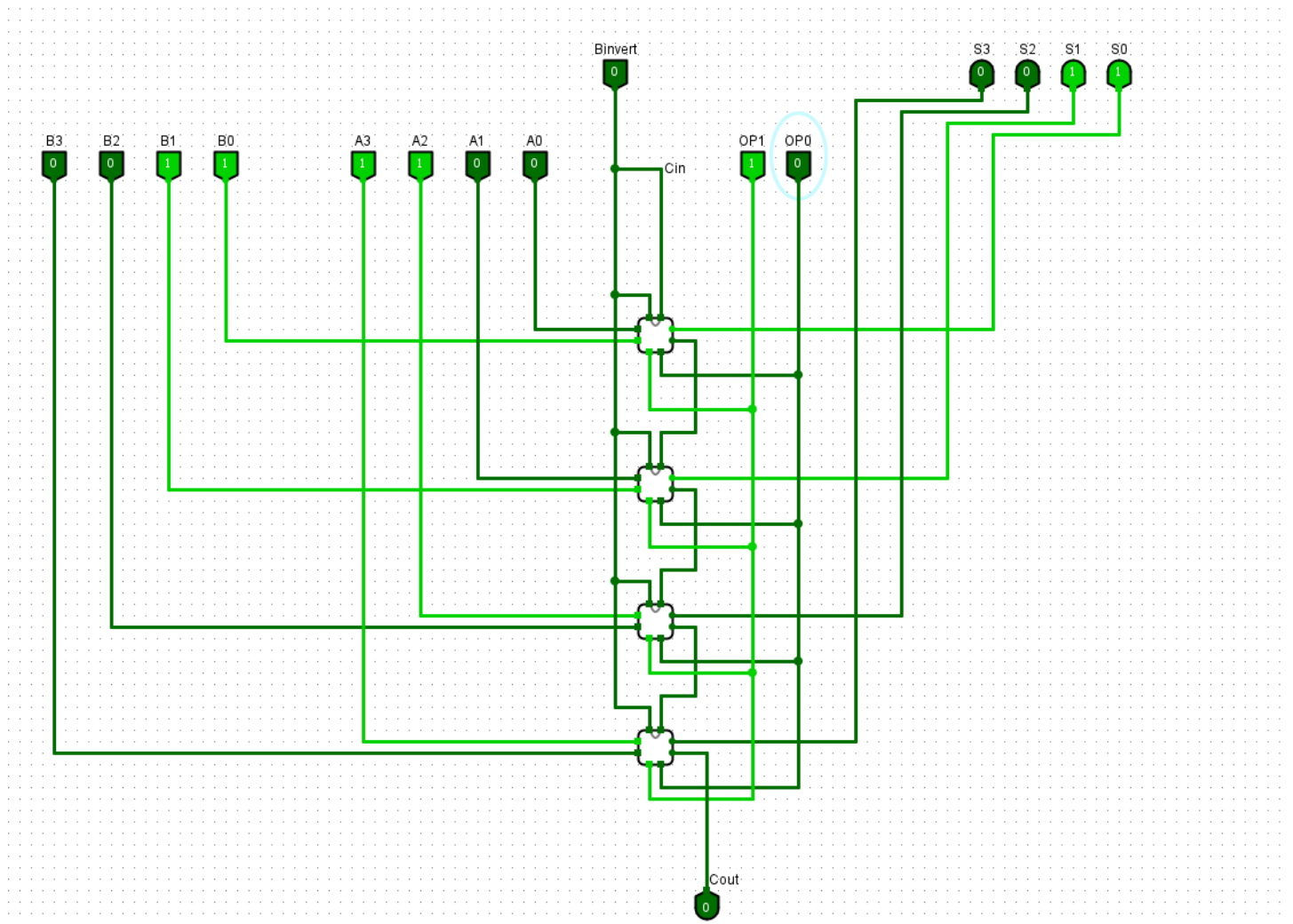
**b) B=3; (ou B=0011)**  
 OR(A,B);



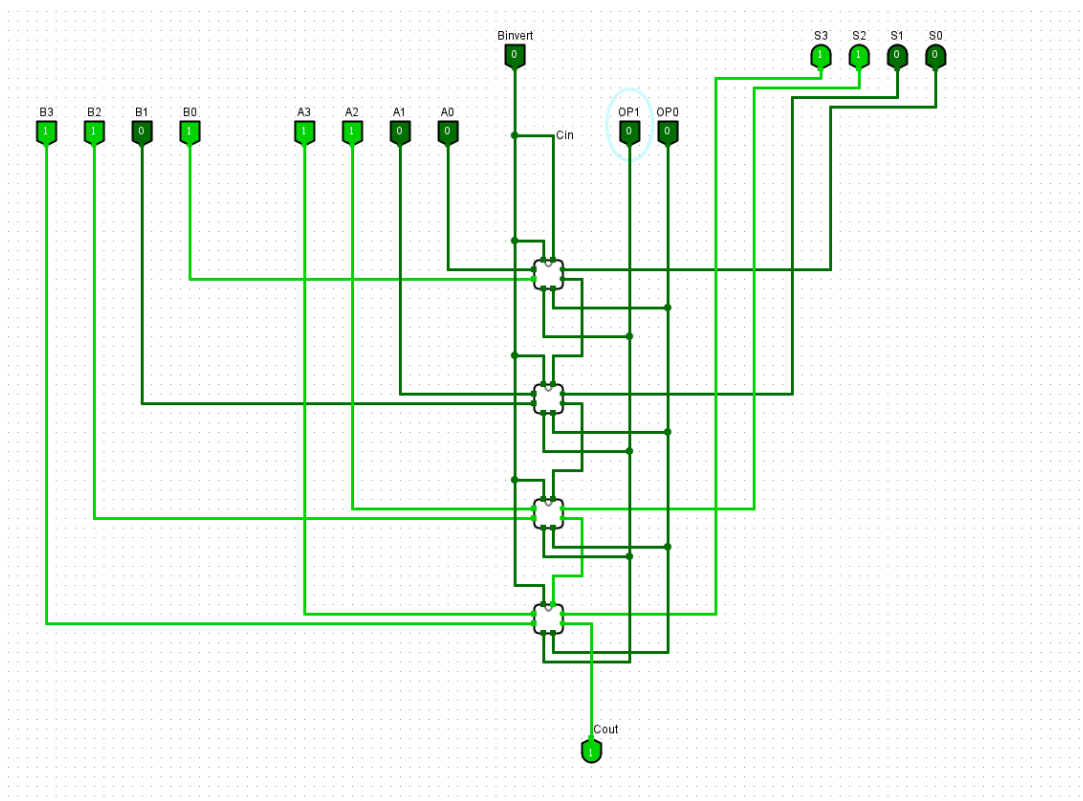
c) SOMA(A,B);



d) A=12; (ou A=1100)  
NOT(A);



e) B=13; ( ou B=1101)  
AND(B,A);





7) Para o programa de teste acima, preencher a tabela a seguir considerando que cada linha corresponderá à execução de uma instrução (a primeira linha já foi realizada, observe que a palavra deverá conter 10bits, para escrevermos em hexa completamos os dois bits à esquerda com zero):

INSTRUÇÃO REALIZADA	BINÁRIO (A, B, Op.Code)	VALOR EM HEXA (Ox ...)	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 0000 10	(0011 0000 0010) = 0x302	0011
AND (B,A)	1100 1101 00	(0011 0011 0100) = 0x334	1100

## Parte 2

Nesta experiência você irá projetar no logisim o circuito 74181, que foi inicialmente utilizado para a construção de computadores de 8 e 16 bits (conforme as figuras abaixo). Posteriormente iremos implementar uma ULA semelhante dentro do Arduino, por isso é importante conhecê-la.

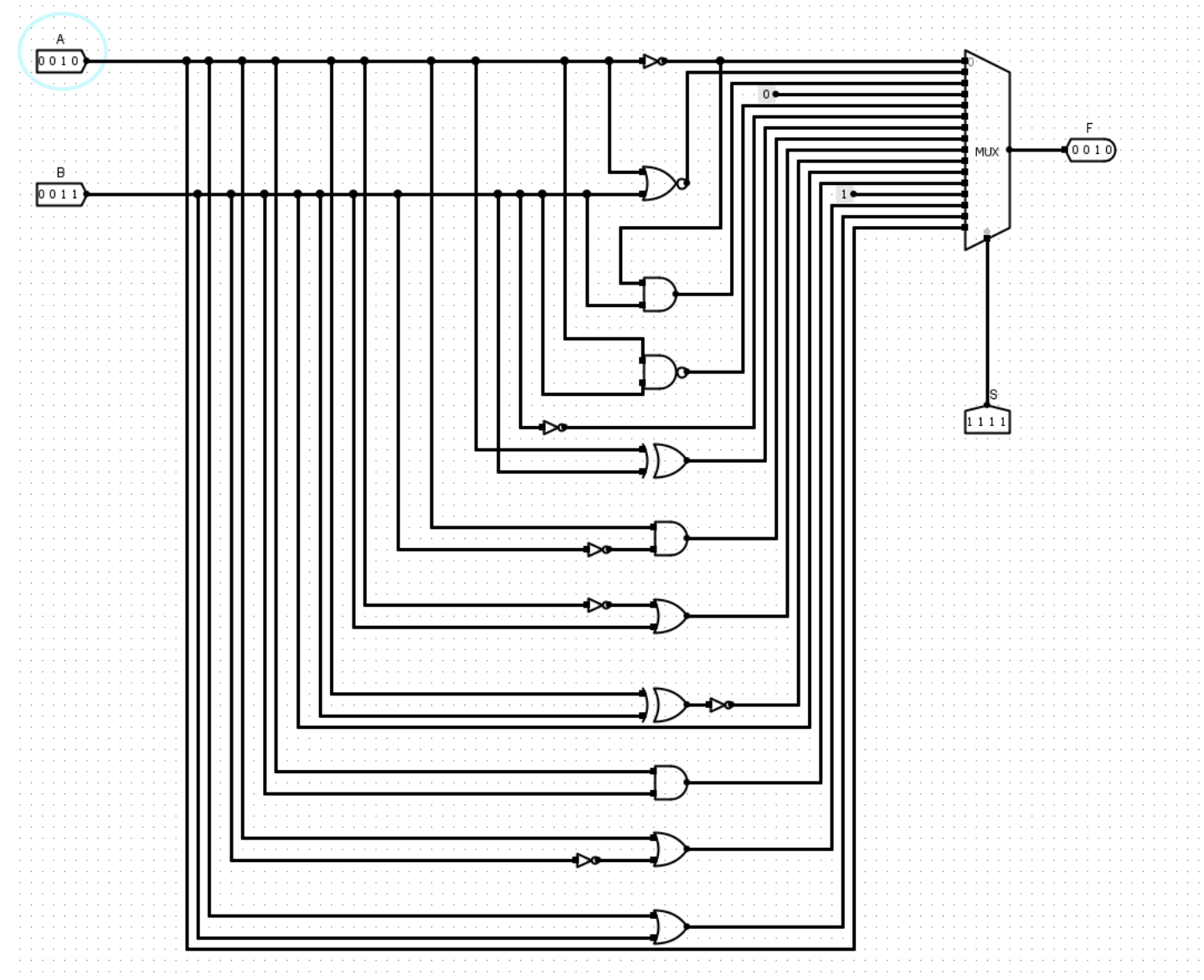
1) A tabela de teste com as funções da ULA completamente preenchida:

Instruções	Binário	Resultado da operação
450	0100 0101 0000	1011
CB1	1100 1011 0001	0010
A32	1010 0011 0010	0001
C43	1100 0100 0011	0000
124	0001 0010 0100	0100
785	0111 1000 0101	0111
9B6	1001 1011 0110	0010
CD7	1100 1101 0111	0000
FE8	1111 1110 1000	1110
649	0110 0100 1001	1101
D9A	1101 1001 1010	1001
FCB	1111 1100 1011	1100
63C	0110 0011 1100	0001

98D	1001 1000 1101	1111
76E	0111 0110 1110	0111
23F	0010 0011 1111	0010

2)

O projeto da ULA no Logisim com um printscreen de alguma instrução da tabela sendo executada.



**Responder:** 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?

→ Essa tabela na verdade deveria ter  $2^{12}$  linhas, ou seja, 4096.