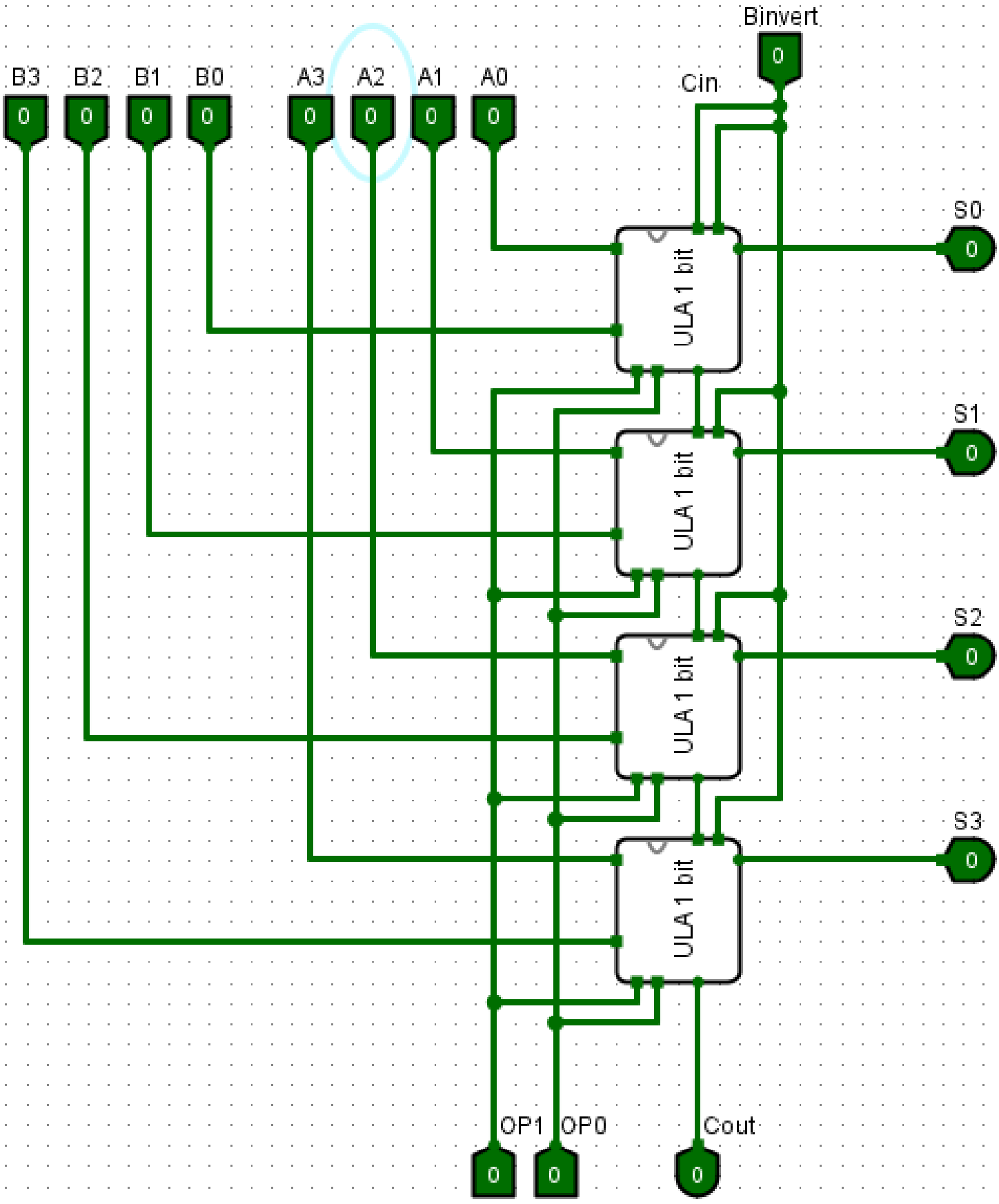
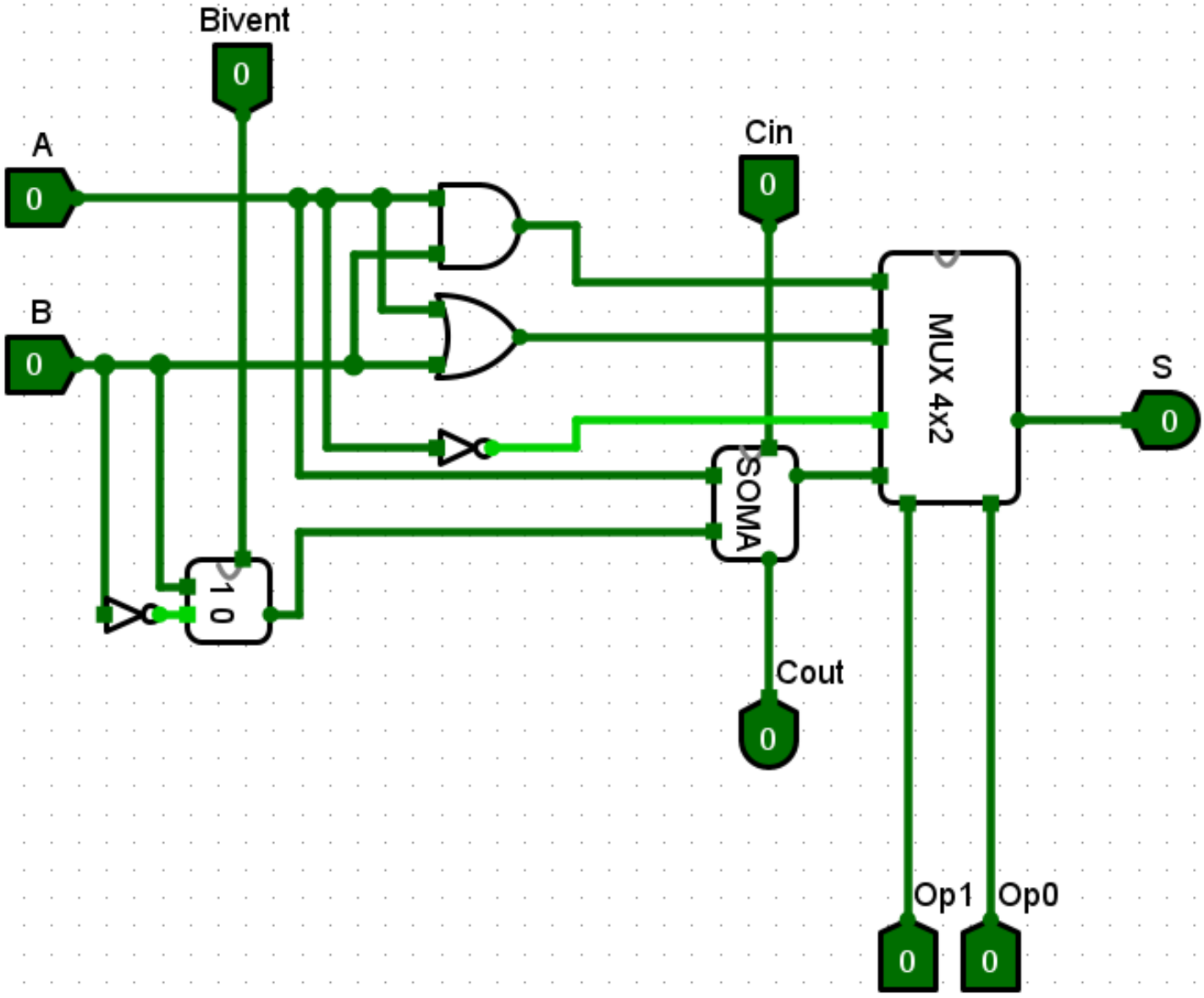
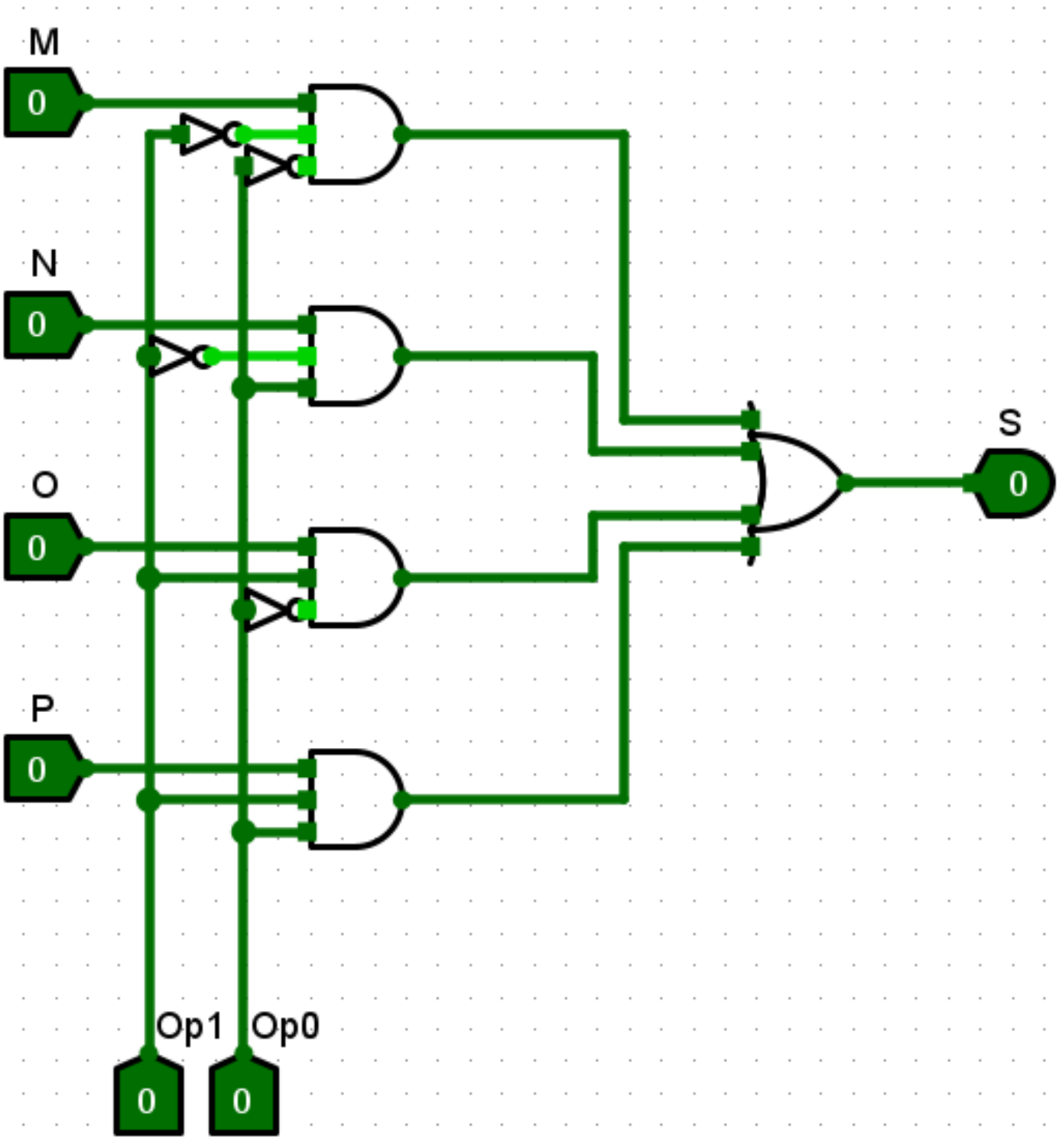
# ULA 4Bits


Binvert
0
B3
B2
B1
B0
A3
A2
A1
A0
0
0
0
0
0
0
0
0
Cin
S0
S1
S2
S3
0
0
0
0
ULA 1 bit
ULA 1 bit
ULA 1 bit
ULA 1 bit
OP1
OP0
Cout
0
0
0

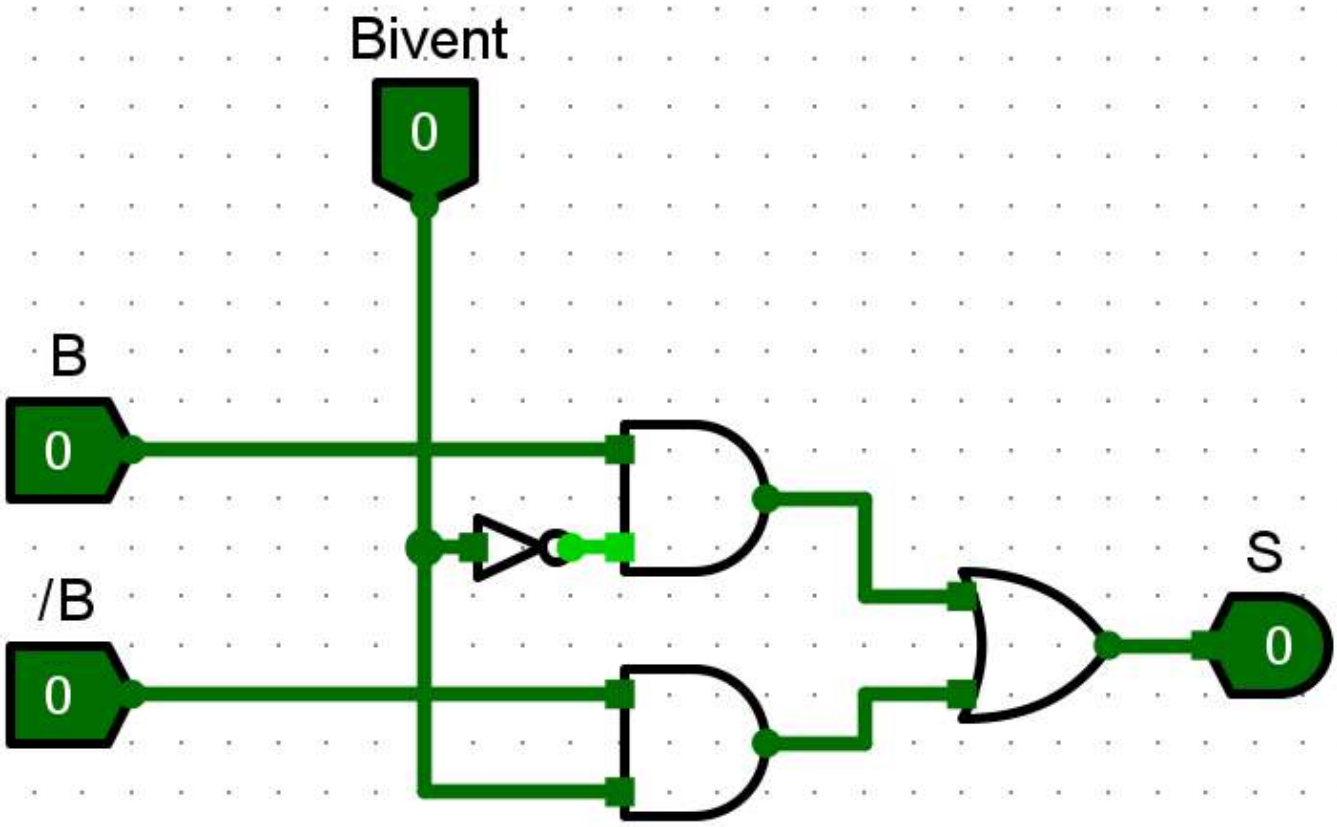
ULA 1Bit



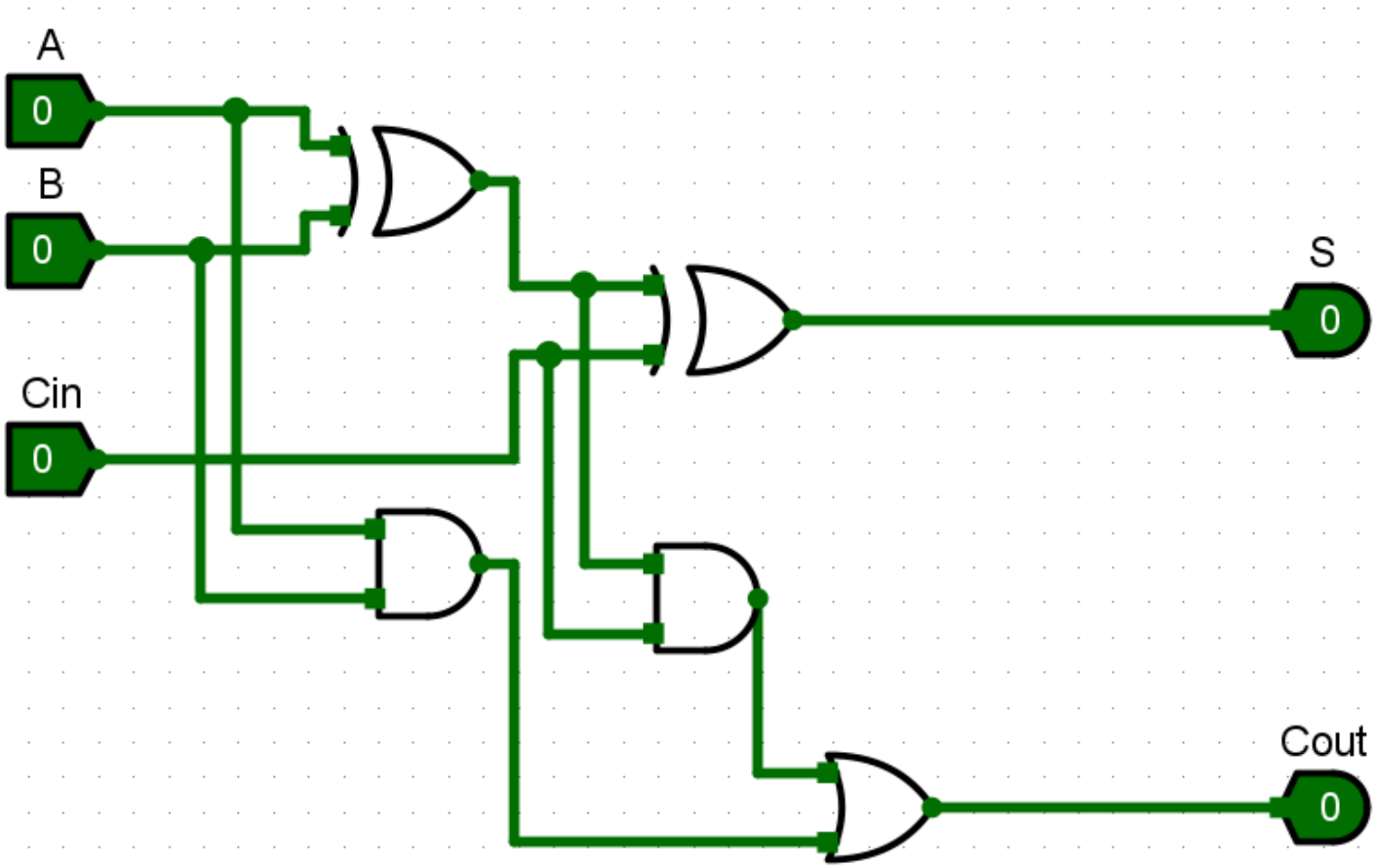
MUX 4x1



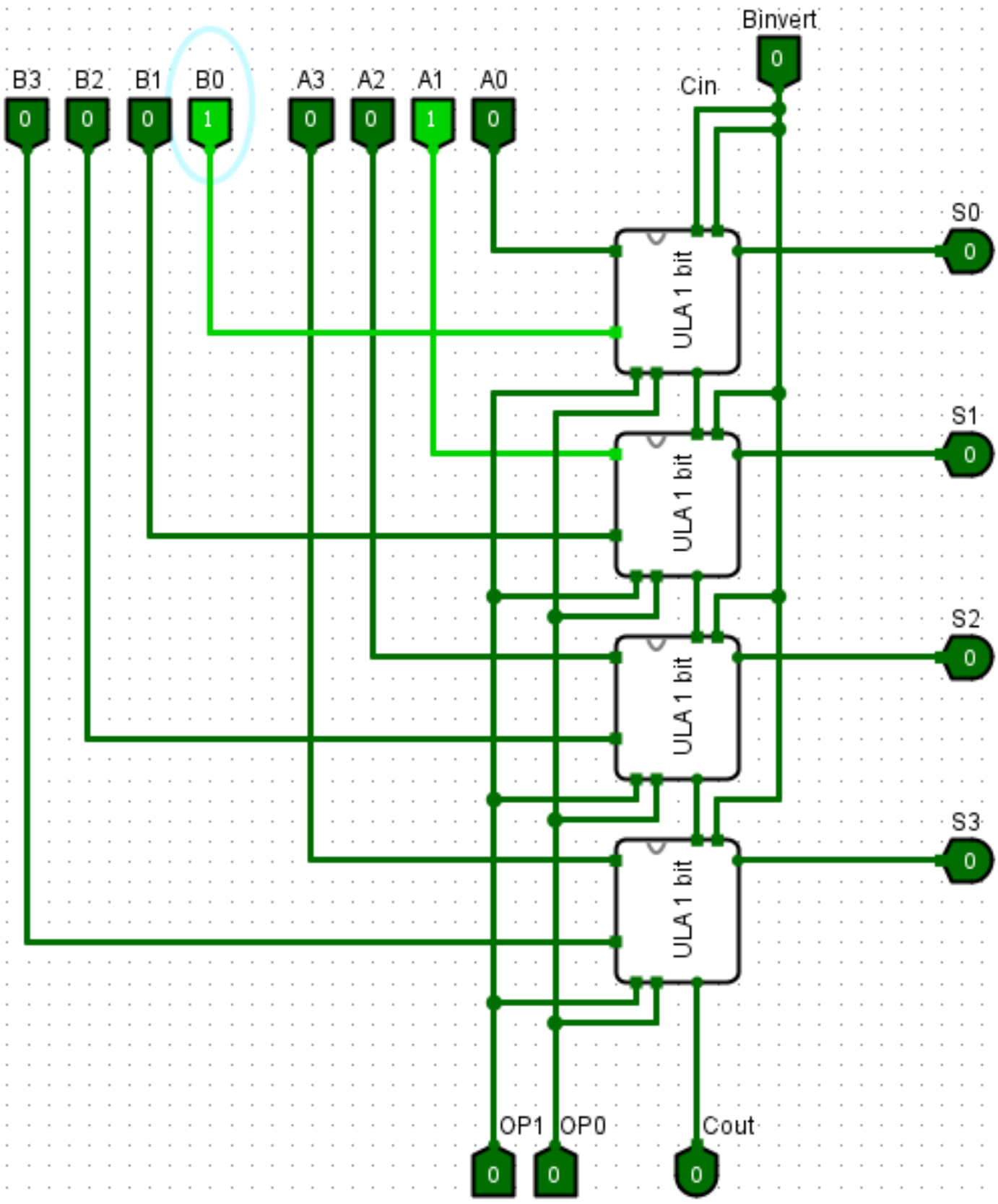
MUX 2x1



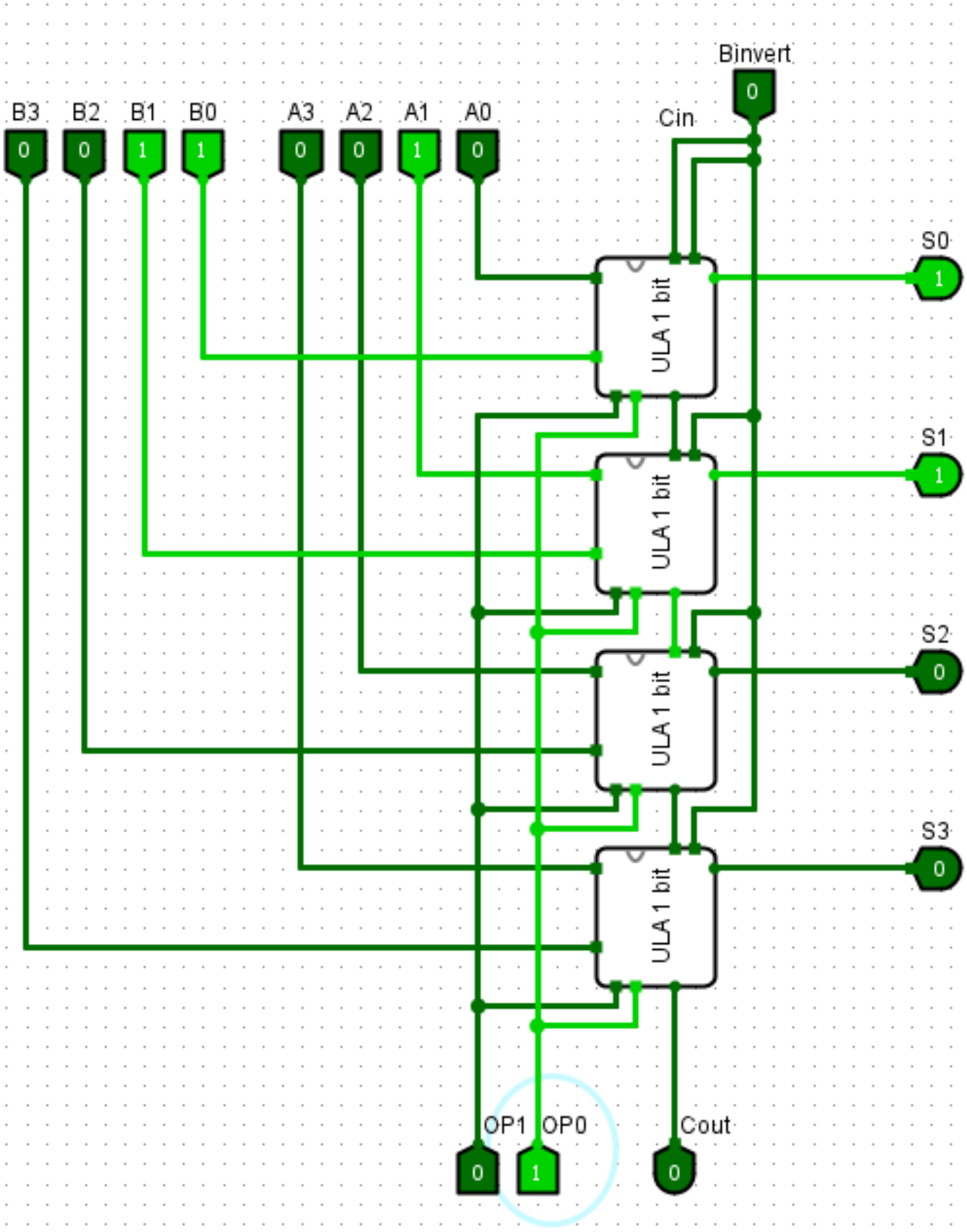
Soma



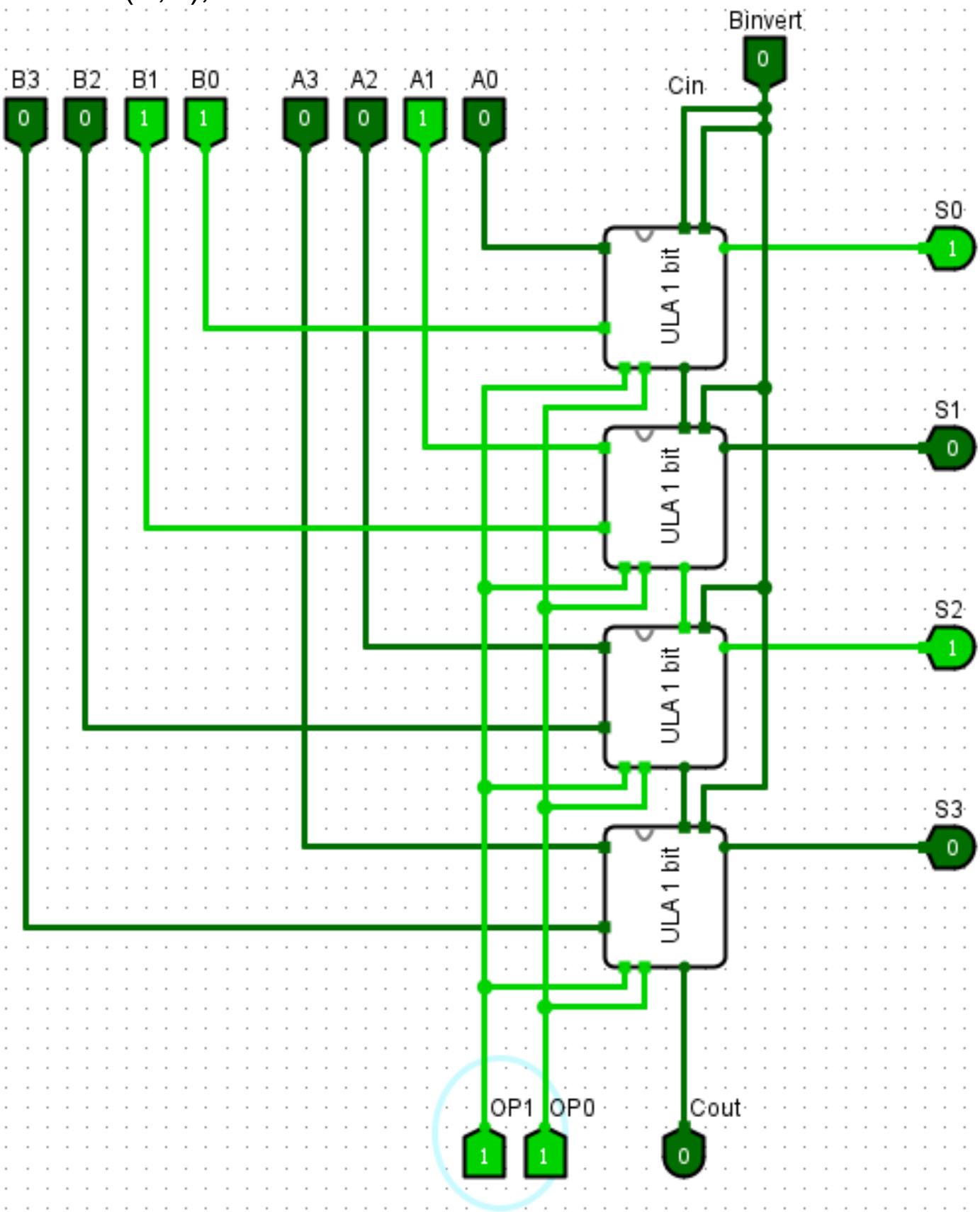
AND(A,B);


B3
B2
B1
B0
A3
A2
A1
A0
Binvert
Cin
S0
S1
S2
S3
ULA 1 bit
ULA 1 bit
ULA 1 bit
ULA 1 bit
OP1
OP0
Cout

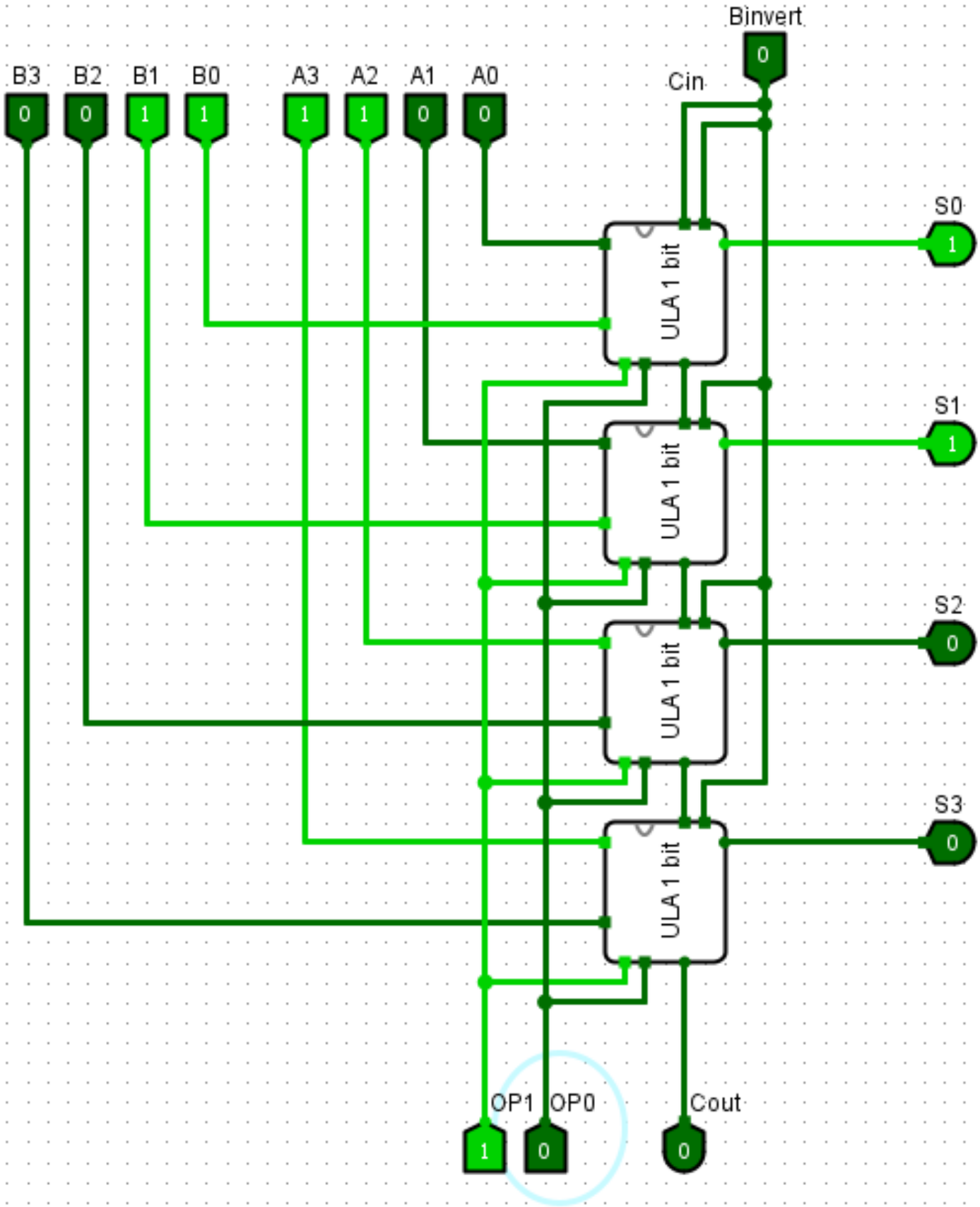
OR(A,B);


Binvert
0
B3
0
B2
0
B1
1
B0
1
A3
0
A2
0
A1
1
A0
0
Cin
S0
1
S1
1
S2
0
S3
0
ULA 1 bit
ULA 1 bit
ULA 1 bit
ULA 1 bit
OP1
0
OP0
1
Cout
0

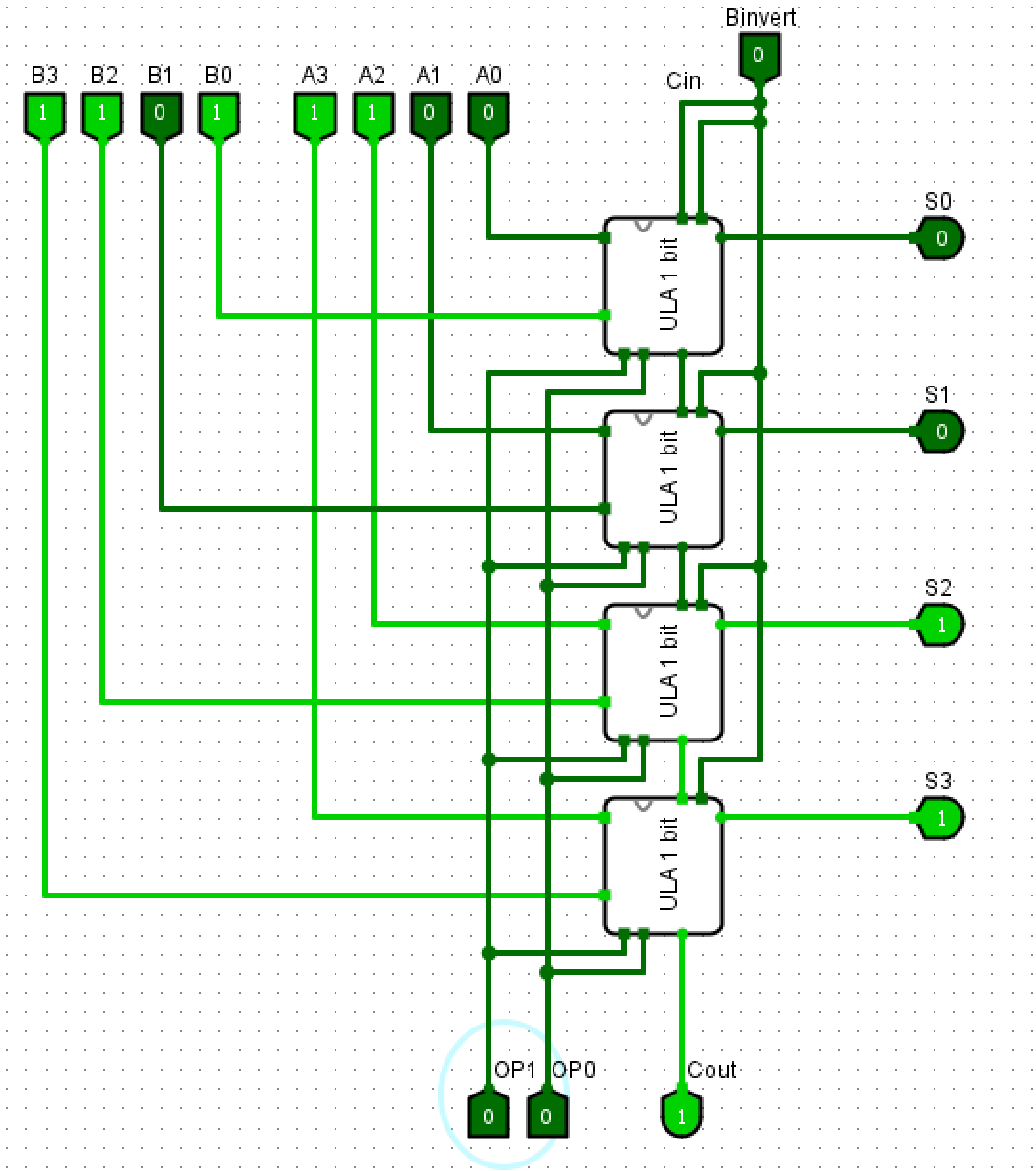
SOMA(A,B);


Binvert
0
B3
B2
B1
B0
A3
A2
A1
A0
0
0
1
1
0
0
1
0
Cin
S0
1
ULA 1 bit
S1
0
ULA 1 bit
S2
1
ULA 1 bit
S3
0
ULA 1 bit
OP1
OP0
Cout
1
1
0

NOT(A);

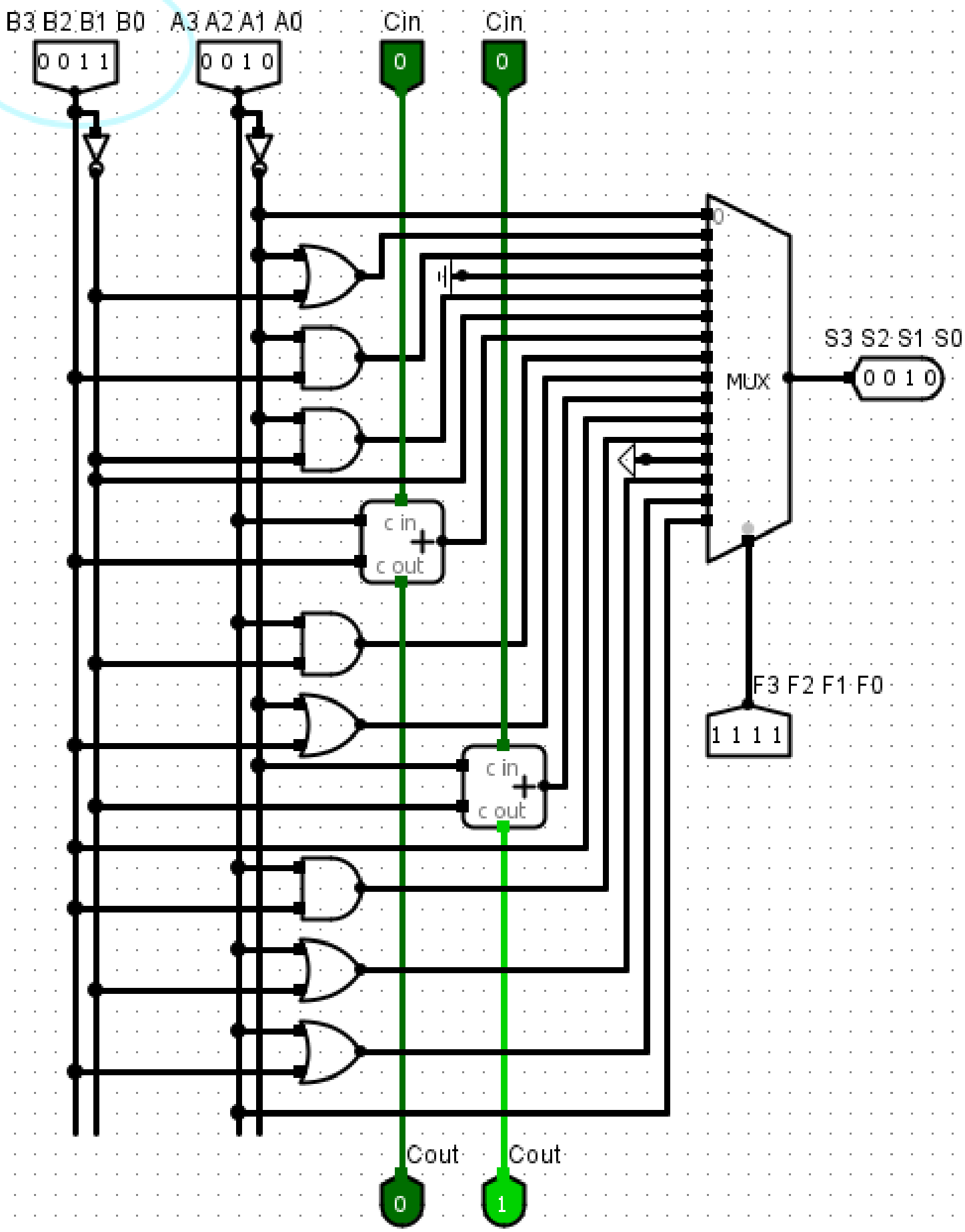

Binvert
0
B3
B2
B1
B0
A3
A2
A1
A0
0
0
1
1
1
1
0
0
Cin
S0
1
S1
1
S2
0
S3
0
ULA 1 bit
ULA 1 bit
ULA 1 bit
ULA 1 bit
OP1
OP0
Cout
1
0
0

AND(B,A);


Binvert
0
B3
1
B2
1
B1
0
B0
1
A3
1
A2
1
A1
0
A0
0
Cin
S0
0
ULA 1 bit
S1
0
ULA 1 bit
S2
1
ULA 1 bit
S3
1
ULA 1 bit
OP1
0
OP0
0
Cout
1

| Instrução Realizada | Binário (A,B,Op.code) | Valor em Hexa (0x ...) | 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 |

ULA 74ls181


B3 B2 B1 B0
0 0 1 1
A3 A2 A1 A0
0 0 1 0
Cin
0
Cin
0
MUX
S3 S2 S1 S0
0 0 1 0
c in
c out
c in
c out
F3 F2 F1 F0
1 1 1 1
Cout
0
Cout
1

| Instruções | Binário | Resultado da Operação |
|---|---|---|
| 450 | 0100 0101 0000 | 1011 = 11 |
| CB1 | 1100 1011 0001 | 0111 = 7 |
| A32 | 1010 0011 0010 | 0001 = 1 |
| C43 | 1100 0100 0011 | 0000 = 0 |
| 124 | 0001 0010 0100 | 1100 = 12 |
| 785 | 0111 1000 0101 | 0111 = 7 |
| 9B6 | 1001 1011 0110 | 0100 = 4 |
| CD7 | 1100 1101 0111 | 0000 = 0 |
| FE8 | 1111 1110 1000 | 1110 = 14 |
| 649 | 0110 0100 1001 | 0100 = 4 |
| D9A | 1101 1001 1010 | 1001 = 9 |
| FCB | 1111 1100 1010 | 1100 = 12 |
| 63C | 0110 0011 1100 | 1111 = 15 |
| 98D | 1001 1000 1101 | 1111 = 15 |
| 76E | 0111 0110 1110 | 0111 = 7 |
| 23F | 0010 0011 1111 | 0010 = 2 |

Resposta:

2^12 = 4096 colunas na tabela verdade indo de 0(0000 0000 0000) a 4095(1111 1111 1111).