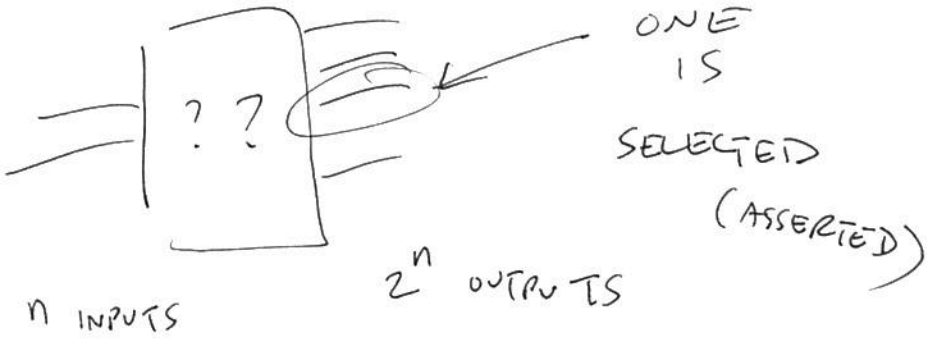
THE DECODER

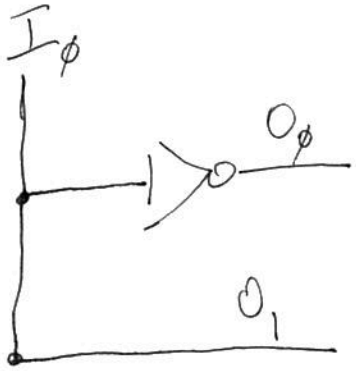


How?

1-2 LINE DEC.

I_0	O_0	O_1
0	1	0
1	0	1

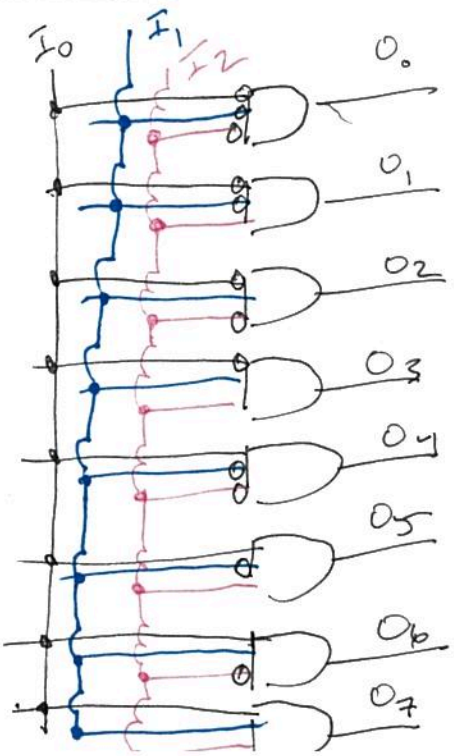
⇒



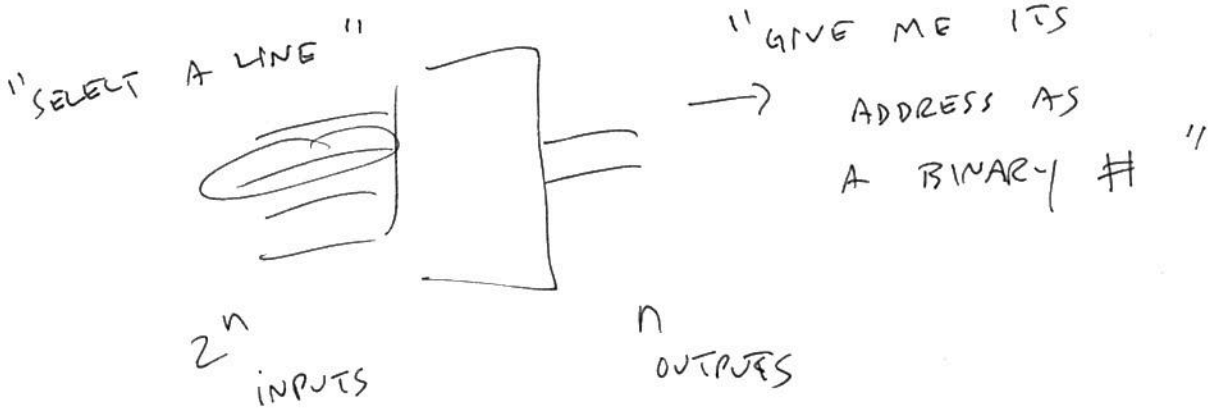
3-8 DECODER

I_0	I_1	I_2	O_0	O_1	O_2	O_3	O_4	O_5	O_6	O_7
0	0	0	1							
0	0	1		1						
0	1	0			1					
0	1	1				1				
1	0	0					1			
1	0	1						1		
1	1	0							1	
1	1	1								1

⇒

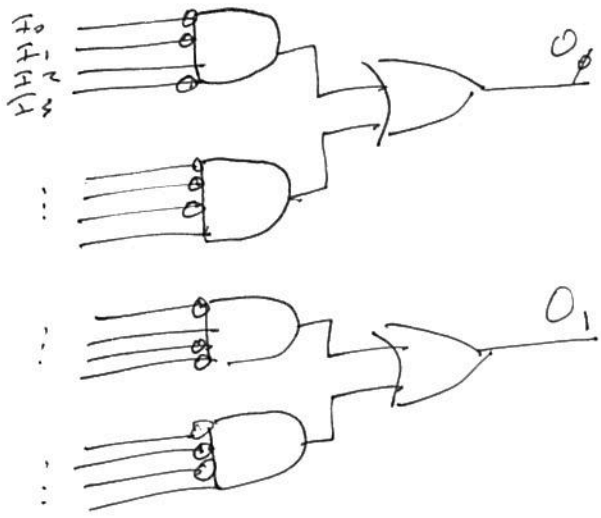


THE ENCODER

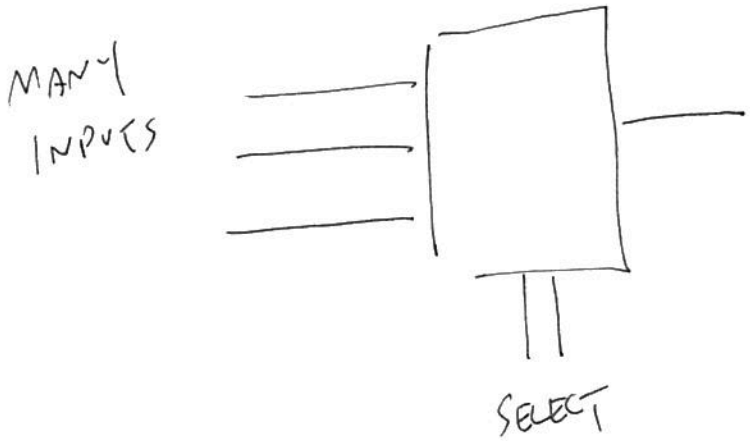


How? 4-2 ENCODER

I_0	I_1	I_2	I_3	O_0	O_1
1	0	0	0	0	0
0	1	0	0	0	1
0	0	1	0	1	0
0	0	0	1	1	1
A	B	C	D		

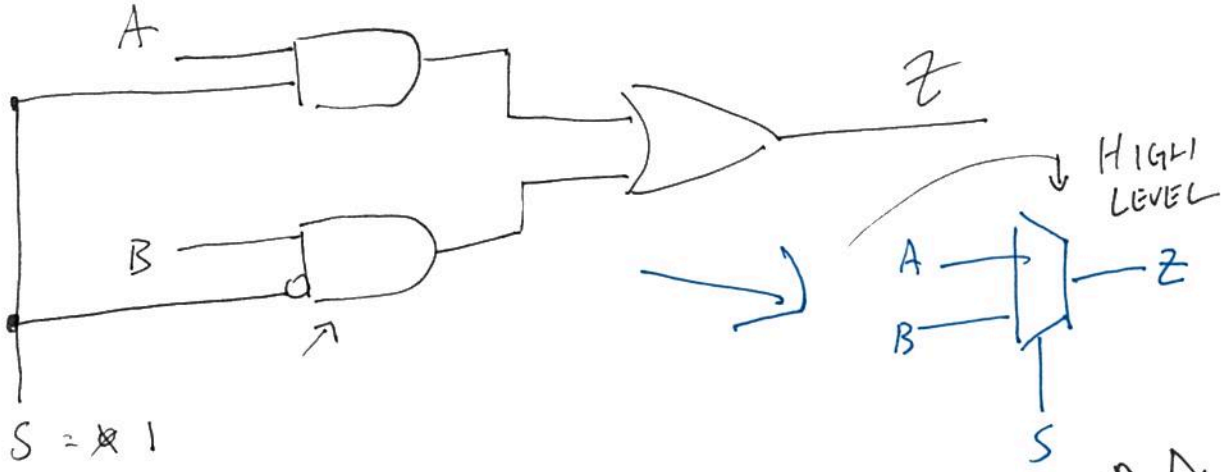


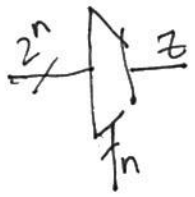
THE MULTIPLEXER



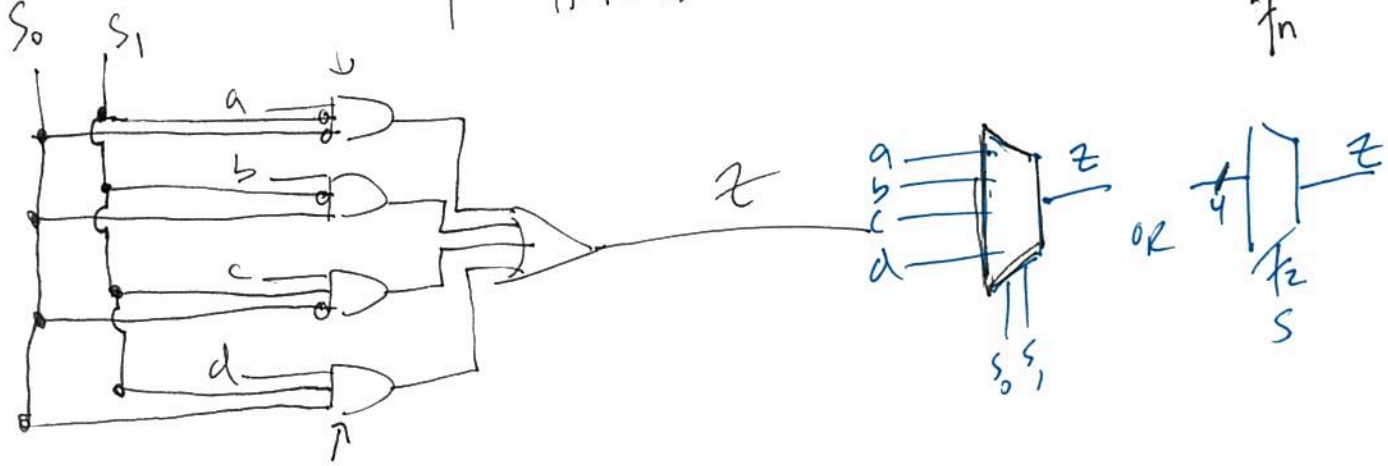
"n to 1 SWITCH"

2 INPUTS





4 INPUTS

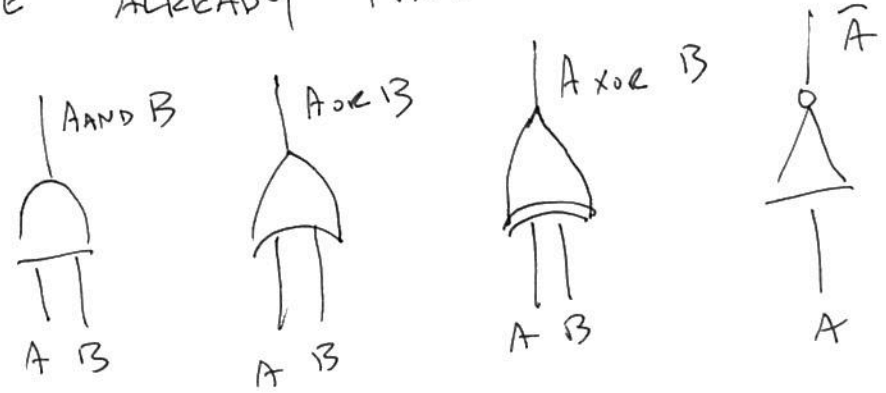


CALCULATOR

4 FUNCTIONS

- BITWISE AND (A, B)
- BITWISE OR (A, B)
- BITWISE XOR (A, B)
- BITWISE NOT (A)

WE ALREADY HAVE COMPONENTS !



WE NEED TO SELECT ONE BASED ON A NUMBER "OPCODE"

00 MEANS "AND"

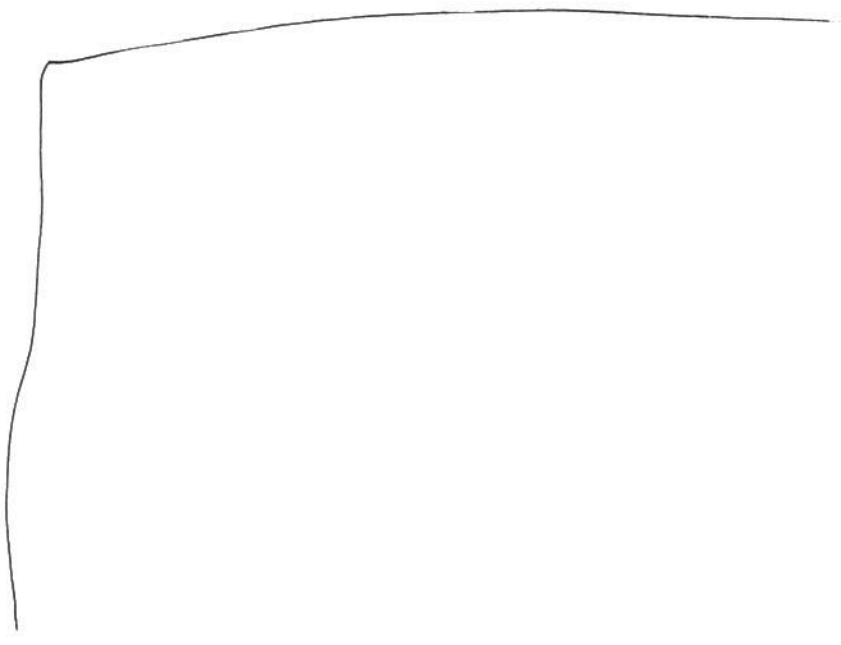
01 MEANS "OR"

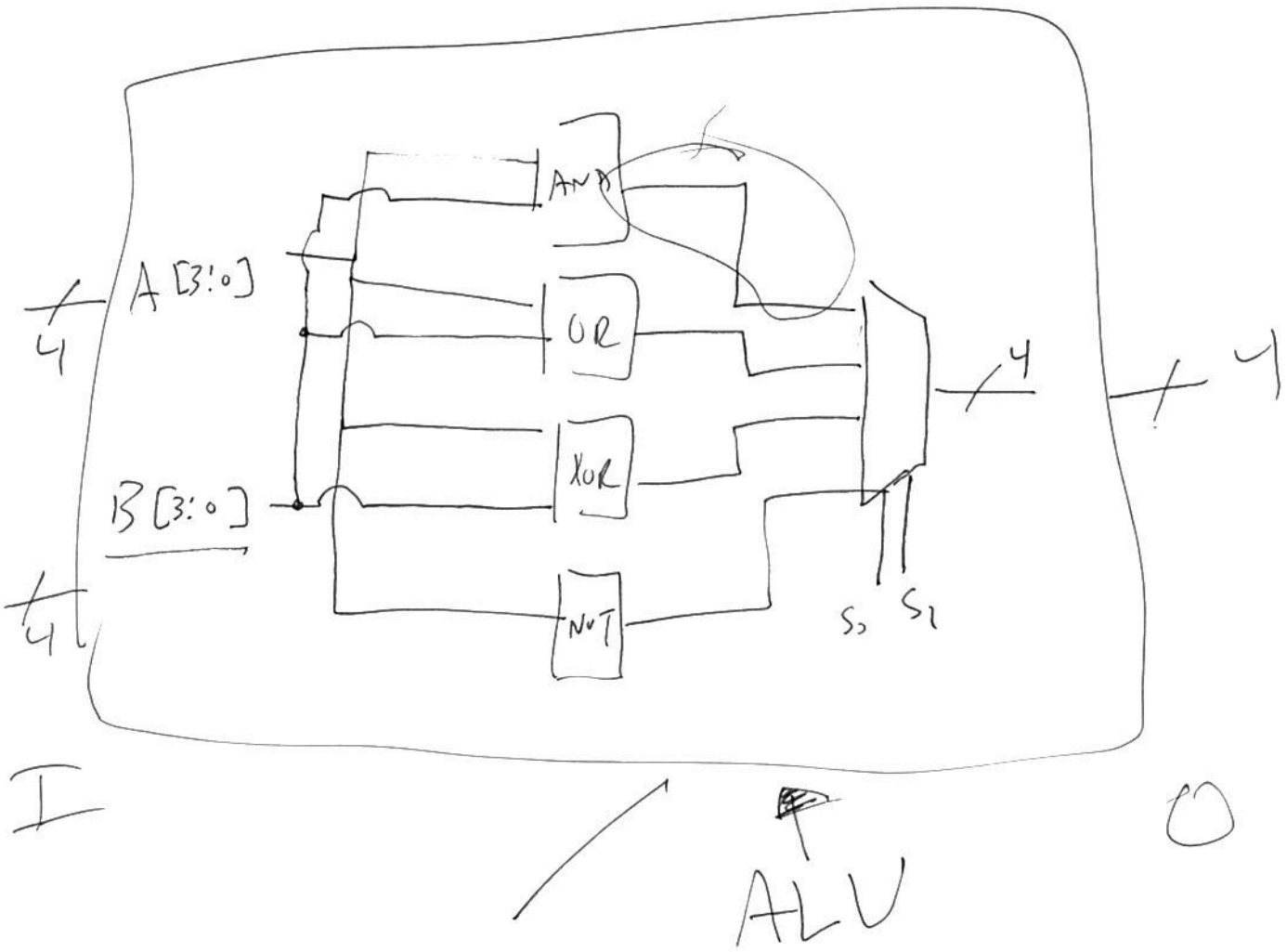
10 MEANS "XOR"

11 MEANS "NOT"

CALL IT S[1:0]

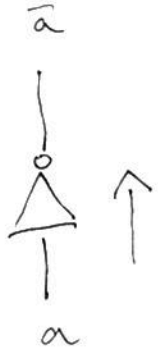
???



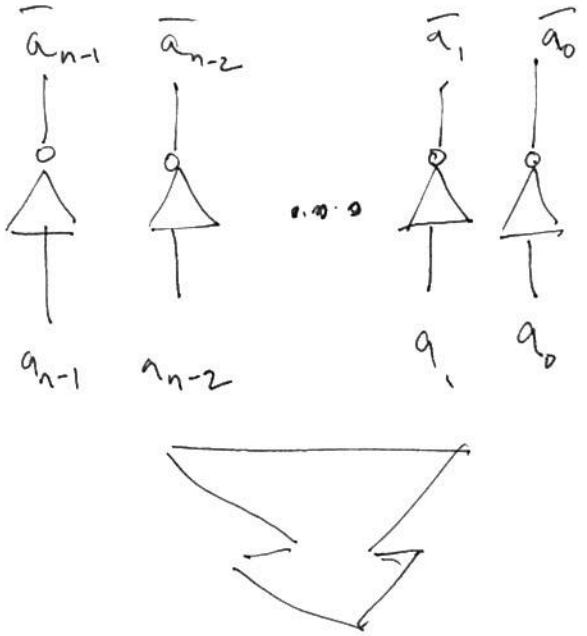


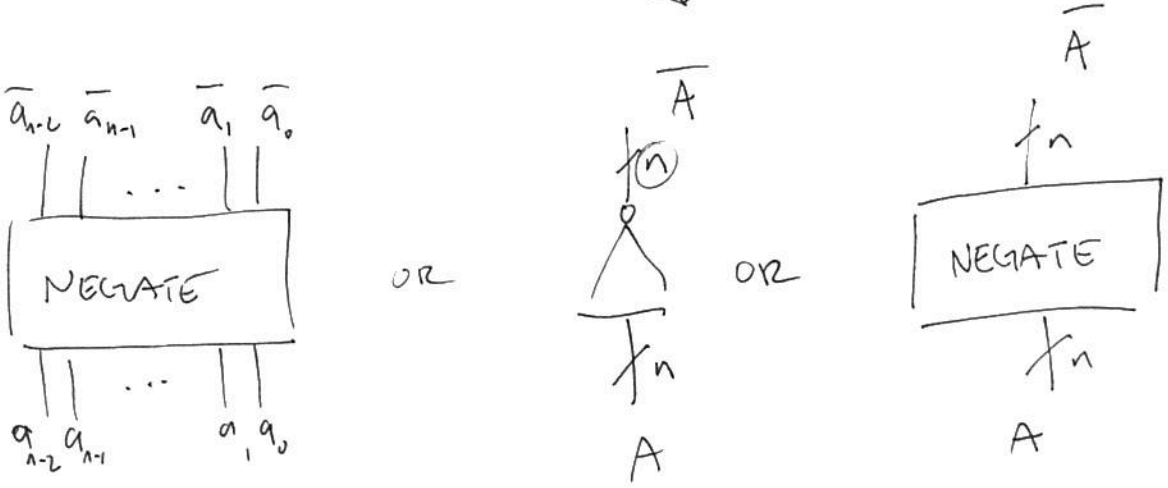
S_0	S_1	
0	0	AND
0	1	OR
1	0	XOR
1	1	NOT

FLIPPING BITS

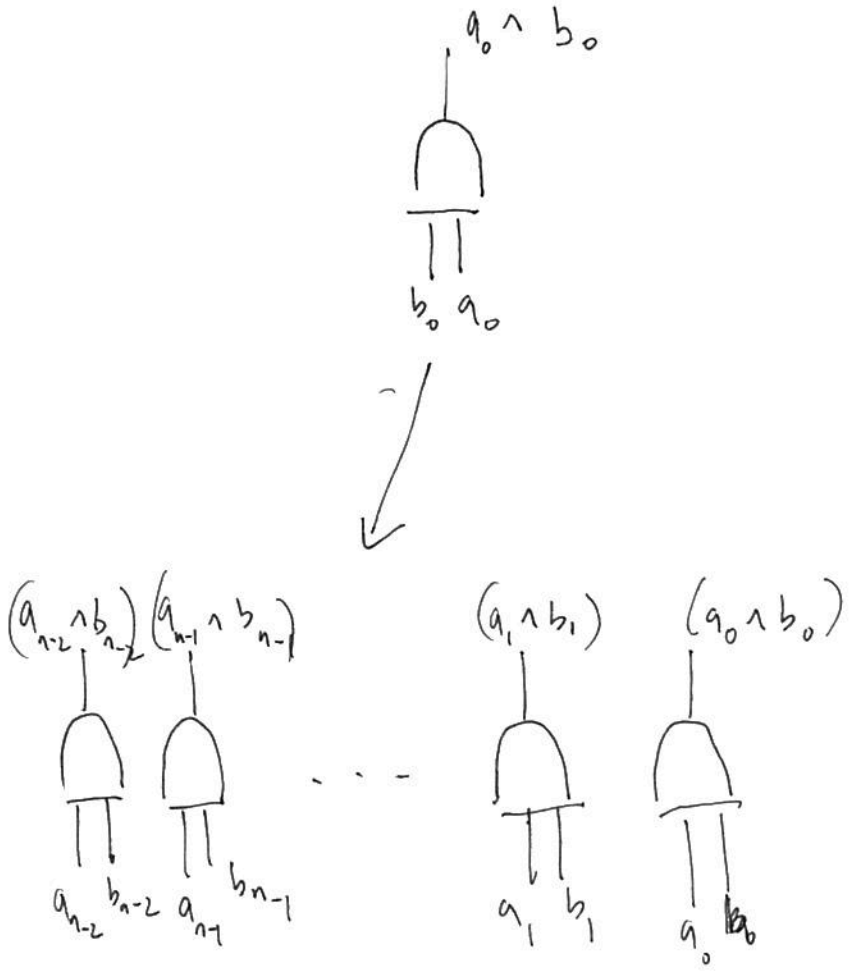


$$A = \{a_{n-1}, a_{n-2}, \cdots, a_1, a_0\}$$





BITWISE OPS



SAME FOR

OR

XOR

NAND

NOR

XNOR

THE FULL ADDER

HOW TO ADD 2 BINARY #s?

MAKE A TRUTH TABLE!

INPUTS: A B ... C_{IN}

OUTPUTS: S ... C_{OUT}

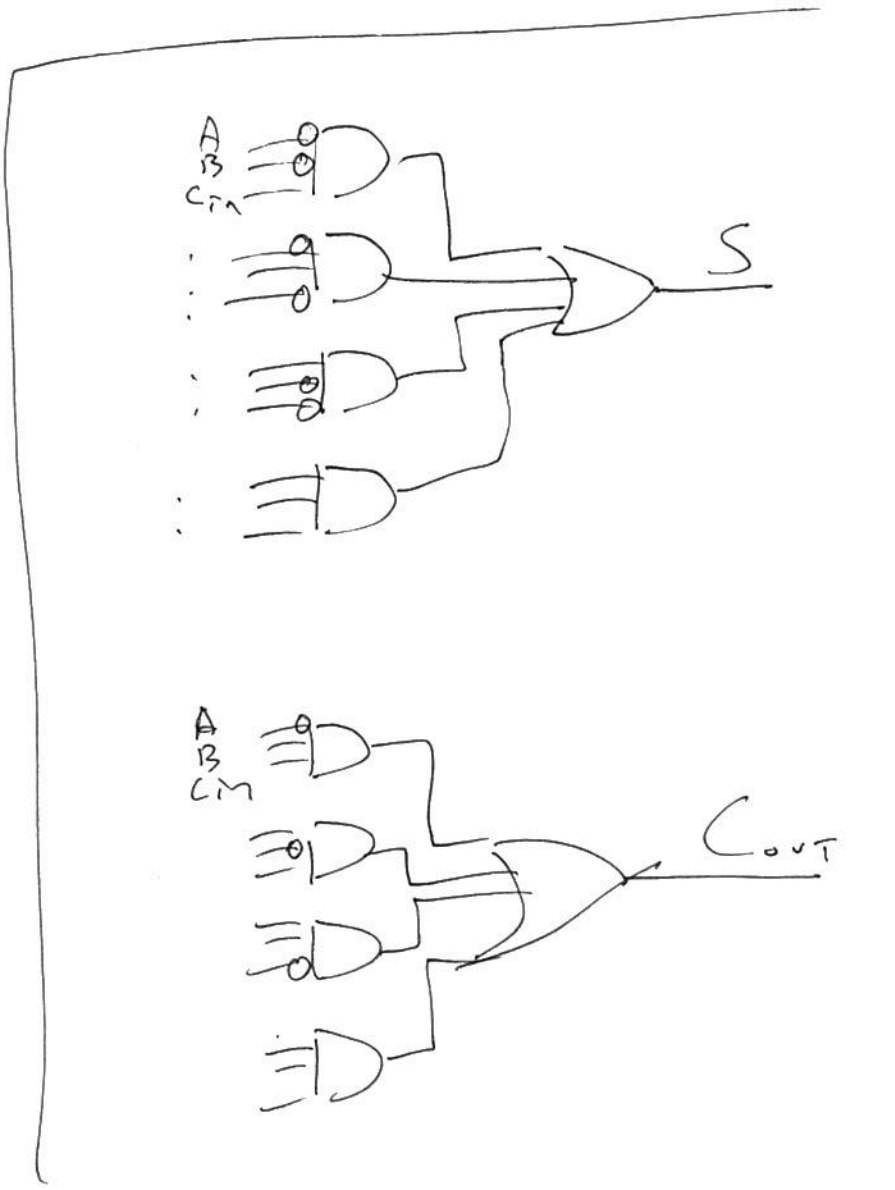
Cin Cout

```
 01
+01
---
 0
```

A	B	C_{IN}	S	C_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$$S = \bar{A}\bar{B}C_{IN} + \bar{A}B\bar{C}_{IN} + A\bar{B}\bar{C}_{IN} + ABC_{IN}$$

$$C_{OUT} = \bar{A}BC_{IN} + A\bar{B}C_{IN} + AB\bar{C}_{IN} + ABC_{IN}$$



1-bit ADDER

"FULL ADDER"