logic/boolean variables $a, b, c, \dots \in \{0, 1\}$

0 = F = False, 1 = T = True

truth table = all possib

$a = F$ OR $b = T$

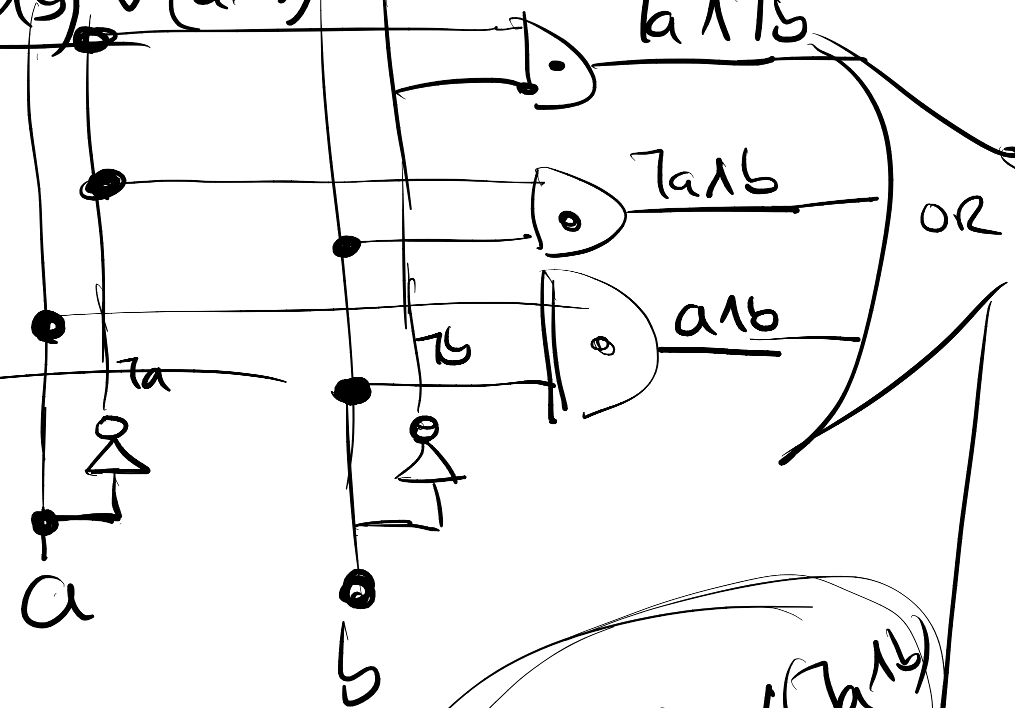
$\neg a$	a	b	c	$a \vee b$ (OR)	$a \vee b \vee c$	$a \wedge b$	$a \wedge b \wedge c$	$\neg a \vee b$
1	0 F	0 F	0 F	0	0	0	0	1
1	0 F	0 F	1 T	0	1	0	0	1
1	0 F	1 T	0 F	1	1	0	0	1
1	0 F	1 T	1 T	1	1	0	0	1
0	1 T	0 F	0 F	1	1	0	0	0
0	1 T	0 F	1 T	1	1	0	0	0
0	1 T	1 T	0 F	1	1	1	0	1
0	1 T	1 T	1 T	1	1	1	1	1

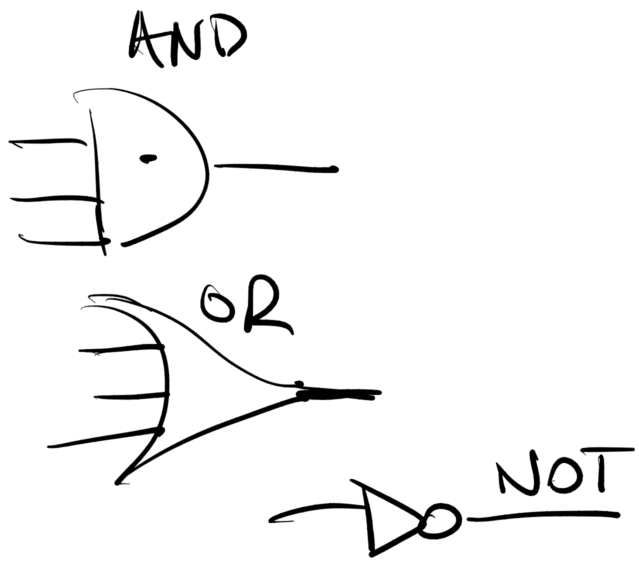
$\neg a \vee b = (\neg a) \vee b$ ✓

$\neq \neg(a \vee b)$

$\neg\neg a = a$ ✓

a	b	$\neg a \vee b$	$(\neg a \wedge \neg b) \vee (\neg a \wedge b) \vee (a \wedge b)$
0	0	1	
0	1	1	
1	0	0	
1	1	1	





$\neg a \vee b$

$(\neg a \wedge \neg b) \vee (\neg a \wedge b) \vee (a \wedge b)$

Gates AND OR NOT XOR

a	b	$a \land b$	$a \lor b$	$\neg a$	XOR	NAND $= \neg(\text{AND})$	XNOR $= \neg(\text{XOR})$	NOR $= \neg(\text{OR})$
0	0	0	0	1	0	1	1	1
0	1	0	1	1	1	1	0	0
1	0	0	1	0	1	1	0	0
1	1	1	1	0	0	0	1	0

$\neg(a \land b)$

$\neg a \lor \neg b$

exclusive OR

exactly 1 = one

$\underbrace{(a \lor b)}_{\text{one}=T} \land \underbrace{\neg(a \land b)}_{\text{not both}=T}$

$\underbrace{(a \land \neg b)}_{a=T,\ b=F} \lor \underbrace{(b \land \neg a)}_{b=T,\ a=F}$

$\neg[(a \lor b) \land \neg(a \land b)]$

$(a \land b) \lor (\neg a \land \neg b)$

universal gate : everything can be made with just that gate

NAND , NOR

Formulas vs Circuits vs Truth Table

truth table → formula → circuit.

a	b	output
0	0	1
0	1	1
1	0	0
1	1	1

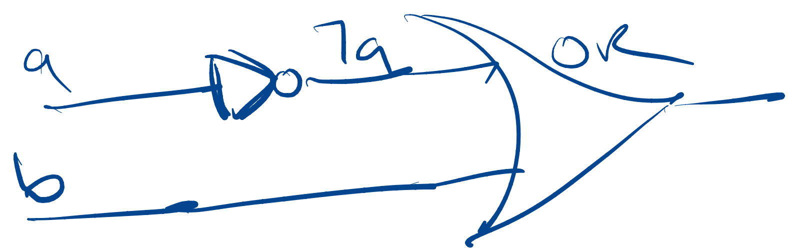
DNF: look for OUT=1 cases and "OR" them.

$$\underbrace{(\boxed{\neg a \wedge \neg b})}_{\text{clause 1}} \vee \underbrace{(\neg a \wedge b)}_{\text{clause 2}} \vee \underbrace{(a \wedge b)}_{\text{clause 3}}$$

CNF: look for OUT=0 cases
avoid ALL of them } ¬ Negate them, AND in between

$$\underbrace{\neg(a \wedge \neg b)}_{\text{clause 1}} \wedge \_\ \_\ \_$$

$$\neg a \vee b$$



a	b	$\neg a$	$\neg b$	$\boxed{\neg a \vee b}$	$\neg a \wedge \neg b$	$\neg a \wedge b$	$a \wedge b$	$(\neg a \wedge \neg b) \vee (\neg a \wedge b) \vee (a \wedge b)$
0	0	1	1	1	1	0	0	1
0	1	1	0	1	0	1	0	1
1	0	0	1	0	0	0	0	0
1	1	0	0	1	0	0	1	1

claim $\boxed{\neg a \vee b} \equiv (\neg a \wedge \neg b) \vee (\neg a \wedge b) \vee (a \wedge b)$

$$(\neg a \wedge \neg b) \vee (\neg a \wedge b) \vee (a \wedge b)$$

$$(\neg a \wedge \neg b) \vee \left[((\neg a \wedge b) \vee a) \wedge ((\neg a \wedge b) \vee b) \right]$$

$$(\neg a \wedge \neg b) \vee \left[\underset{T}{(\neg a \vee a)} \wedge (b \vee a) \wedge (\neg a \vee b) \wedge \underset{b}{(b \vee b)} \right]$$

$$(\neg a \wedge \neg b) \vee \left[(b \vee \underset{F}{(a \wedge \neg a)}) \wedge b \right]$$

$$(\neg a \wedge \neg b) \vee b = \boxed{(\neg a \vee b)} \wedge \overset{T}{(b \vee \neg b)}$$

$$\neg(a \wedge b) = \neg a \vee \neg b$$

$$\boxed{\neg(a \vee b) = \neg a \wedge \neg b}$$

$$\neg\neg a = \neg(\neg a) = a$$

$$a \wedge b = b \wedge a$$

$$a \vee b = b \vee a$$

$$a \wedge T = a$$

$$a \wedge F = F$$

$$a \vee T = T$$

$$a \vee F = a$$

$$a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$$

$a \cdot (b + c) = ab + ac$ (numbers)

$$a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$$

a	b	$a \vee b$	$\neg(a \vee b)$	$\neg a \wedge \neg b$
0	0	0	1	1
0	1	1	0	0
1	0	1	0	0
1	1	1	0	0

a	b	c	$a \vee (b \wedge c)$	$(a \vee b) \wedge (a \vee c)$
0	0	0	0	0
0	0	1	0	0
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

Carry (Full adder) on bits

CNF(S): look for 0-case negate, AND

a	b	c = c_{in}	S	Cout
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

$\neg(\neg a \wedge \neg b \wedge \neg c) \wedge \neg(\neg a \wedge b \wedge c)$
$\wedge \neg(a \wedge \neg b \wedge c) \wedge \neg(a \wedge b \wedge \neg c)$

$(a \vee b \vee c) \wedge (a \vee \neg b \vee \neg c)$
$\wedge (\neg a \vee b \vee \neg c) \wedge (\neg a \vee \neg b \vee c)$

S-DNF: look for 1-cases, OR them
$(\neg a \wedge \neg b \wedge c) \vee (\neg a \wedge b \wedge \neg c) \vee (a \wedge \neg b \wedge \neg c) \vee (a \wedge b \wedge c)$

Cout-DNF $= (\neg a \wedge b \wedge c) \vee (a \wedge \neg b \wedge c) \vee (a \wedge b \wedge \neg c) \vee (a \wedge b \wedge c)$

S-DNF: look for 1-cases, OR them

$(\neg a \land \neg b \land c) \lor (\neg a \land b \land \neg c) \lor (a \land \neg b \land \neg c) \lor (a \land b \land c)$

Cout-DNF $= (\neg a \land b \land c) \lor (a \land \neg b \land c) \lor (a \land b \land \neg c) \lor (a \land b \land c)$