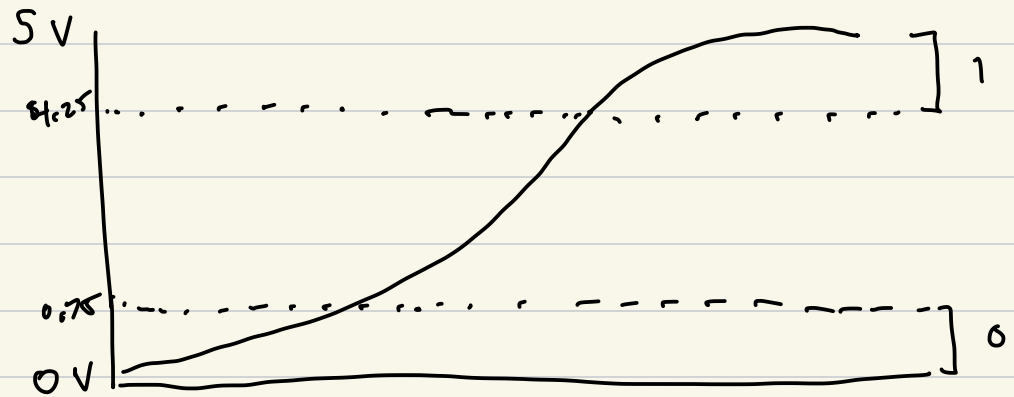
CS315-02 Lab Intro to Digital Design

Digital Design

Analog → Digital



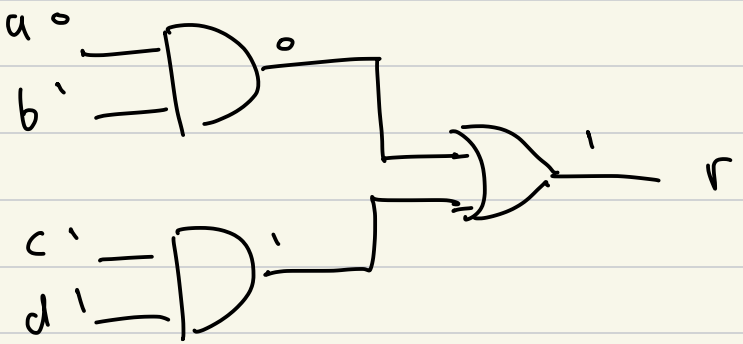
Wires & devices → gates

	AND	OR	NOT
	a, b → r	a, b → r	a → r
C code	r = a & b	r = a | b	r = ~a
Boolean Algebra	r = a · b	r = a + b	r = $\bar{a}$
Logic	r = a ∧ b	r = a ∨ b	r = ¬a

a	b	r
0	0	0
0	1	0
1	0	0
1	1	1

a	b	r
0	0	0
0	1	1
1	0	1
1	1	1

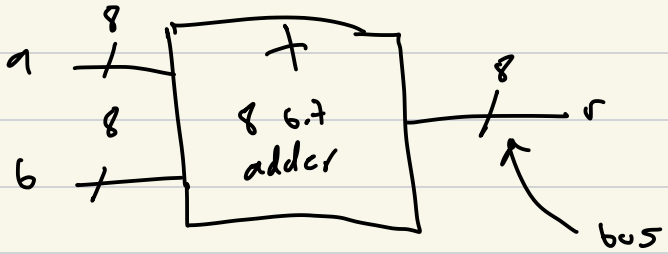
a	r
0	1
1	0



$$r = (a \cdot b) + (c \cdot d)$$
$$= 0 \cdot 1 + 1 \cdot 1$$
$$= 0 + 1$$
$$= 1$$

Abstraction in Digital Design

Goal: build an 8-bit adder



Sum-of-products

Sum of two 1-bit numbers

XOR

a	b	Sum	Cout
0	0	0	
0	1	1	
1	0	1	
1	1	0	

XOR

$sum = a \oplus b$

$sum = (\bar{a} \cdot b) + (a \cdot \bar{b})$

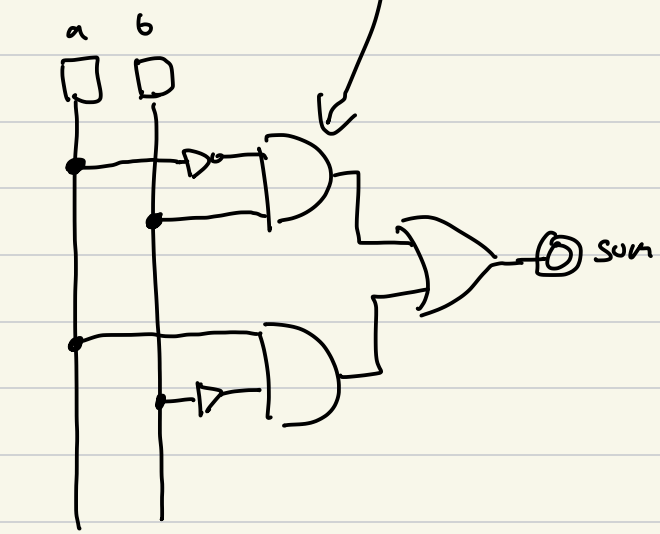
$a = 0 \quad b = 1$

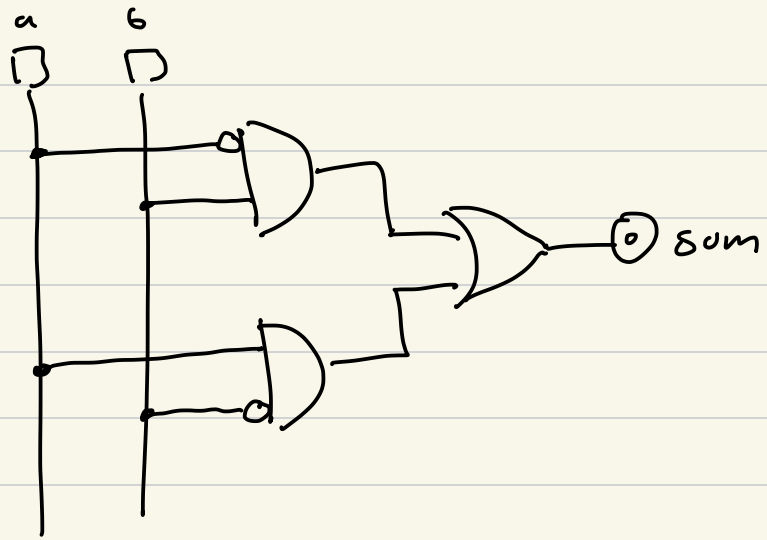
$sum = (\bar{0} \cdot 1) + (0 \cdot \bar{1})$

$= (1 \cdot 1) + (0 \cdot 0)$

$= 1 + 0$

$= 1$





Sum-of-products

1) Build a truth table for your function

For each output

2) Identify rows with output of 1
3) Construct product (·) terms for each row
   a) if input is 1 ⇒ do not invert var
   b) if input is 0 ⇒ invert var
4) Sum (+) all product terms