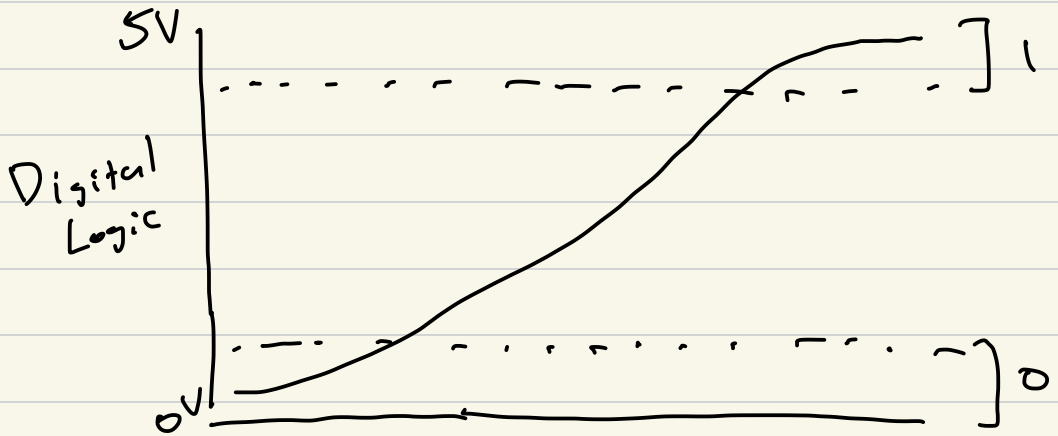
CS315-01 Lab Intro to Digital Design

Digital Design

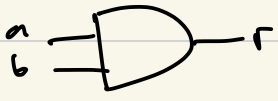
Analog → Digital



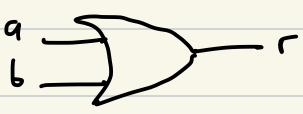
wires

devices → gate

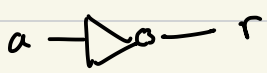
AND



OR



NOT

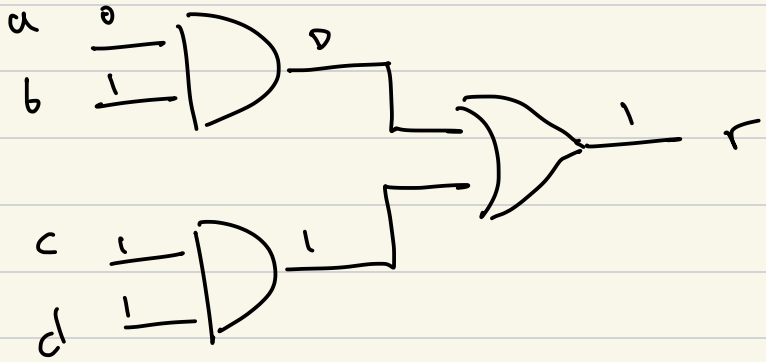


	AND	OR	NOT
C code	r = a && b	r = a | b	r = !a
Boolean Algebra	r = a · b	r = a + b	r = $\bar{a}$
Logic	r = a ∧ b	r = a ∨ b	r = ¬a

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

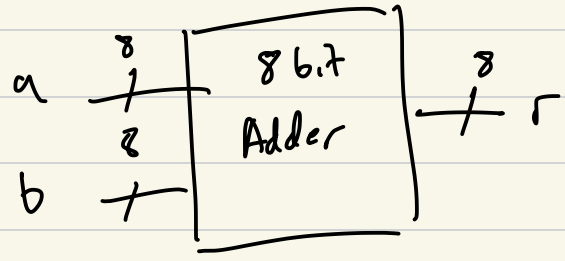
a	r
0	1
1	0



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

Abstraction

Goal



Sum-of-products

sum of two 1-bit numbers

a	b	sum
0	0	0
0	1	1
1	0	1
1	1	0

XOR sum = $a \oplus b$

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

product term

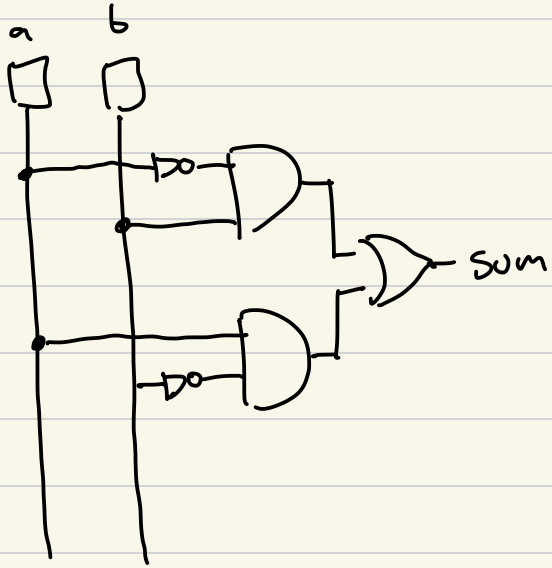
sum

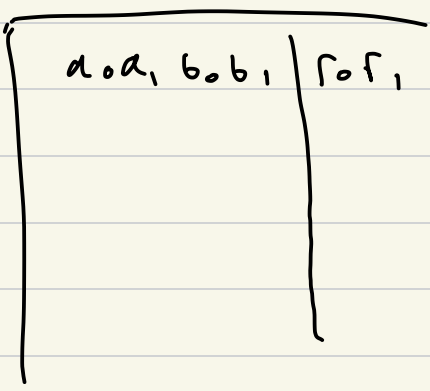
$a = 0 \quad b = 1$

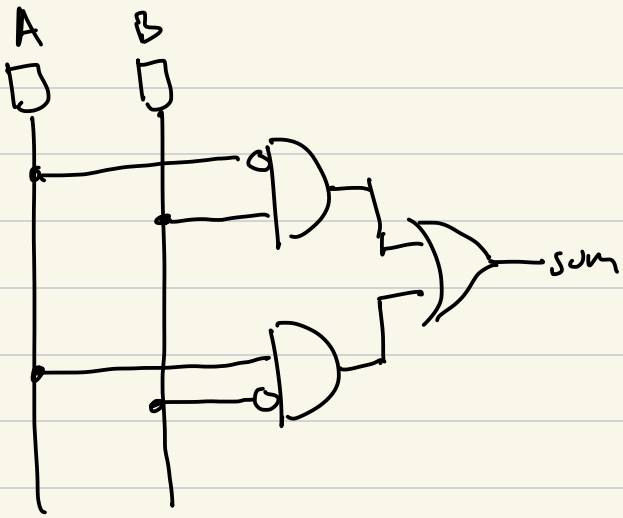
$$sum = (\bar{0} \cdot 1) + (0 \cdot \bar{1})$$
$$= (1 \cdot 1) + (0)$$
$$= \boxed{1}$$

sum = 0;

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







a_1 a_0	b_1 b_0	r_1 r_0
0 0	0 1	0 1
1 1	1 0	1 1

Sum-of-products

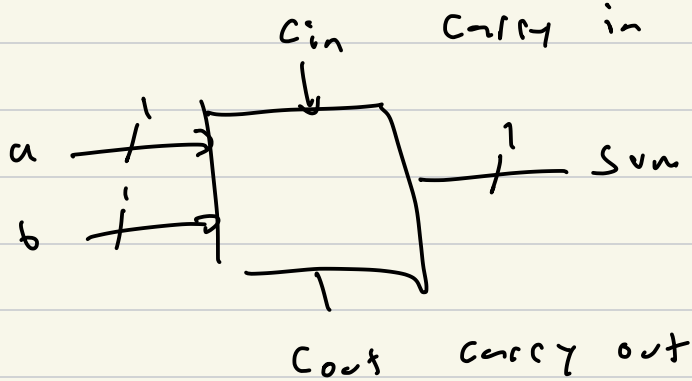
1) build truth table

2) Identify row with output 1

3) Construct product ($\cdot$) terms for each row
   a) don't invert if input is 1
   b) invert if input is 0

4) Sum (+) all product terms

1 bit full adder



a	b	Cin	sum	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

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