Digital Arithmetic

Computer Systems 2016

Oleks

DIKU

September 14, 2016

0

1

Truth-Tables

NOT

A	$\overline{A}$
0	1
1	0

Truth-Tables

NOT

A	$\overline{A}$
0	1
1	0

AND

A	B	$A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1

OR

A	B	$A + B$
0	0	0
0	1	1
1	0	1
1	1	1

More Truth-Tables

XOR

A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

NAND

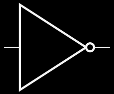
A	B	$A \uparrow B$
0	0	1
0	1	1
1	0	1
1	1	0

NOR

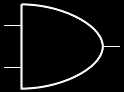
A	B	$A \downarrow B$
0	0	1
0	1	0
1	0	0
1	1	0

Gates

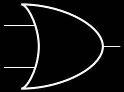
NOT



AND

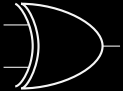


OR

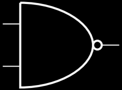


More Gates

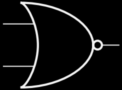
XOR



NAND



NOR



Functional Completeness

Definition

Given a set $\mathbb{B} = \{ 0, 1 \}$, a set of functions $F = \{ f_i : \mathbb{B}^{n_i} \to \mathbb{B} \}$ is *functionally complete* if all functions $f : \mathbb{B}^n \to \mathbb{B}$, for all $n \geq 1$, can be generated by the functions in F.

Examples of fnctionally complete sets:

- $\{$ **NAND** $\}$, $\{$ **NOR** $\}$,
- $\{$ **AND**, **NOT** $\}$, $\{$ **OR**, **NOT** $\}$.

Consequence: Any boolean function can be built by combining functions from a functionally complete set.