

RANGKAIAN LOGIKA

TE 2214

Materi Kuliah ke-4

LOGIC GATE

Overview

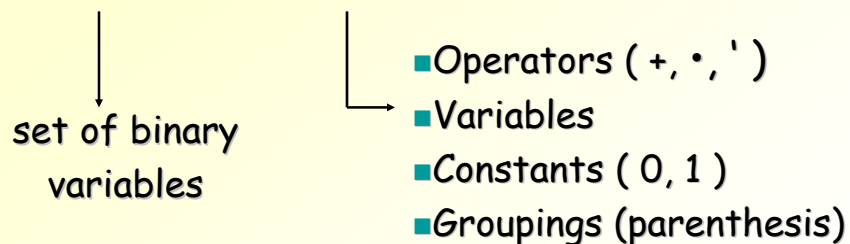
- **Binary logic and Gates**
- **Boolean Algebra**
 - Basic Properties
 - Algebraic Manipulation
- **Standard and Canonical Forms**
 - Minterms and Maxterms (Canonical forms)
 - SOP and POS (Standard forms)
- **Karnaugh Maps (K-Maps)**
 - 2, 3, 4, and 5 variable maps
 - Simplification using K-Maps
- **K-Map Manipulation**
 - Implicants: Prime, Essential
 - Don't Cares

Binary Logic

- Deals with binary variables that take 2 discrete values (0 and 1), and with logic operations
- Three basic logic operations:
 - AND, OR, NOT
- Binary/logic variables are typically represented as letters: A,B,C,...,X,Y,Z

Binary Logic Function

$F(\text{vars}) = \text{expression}$



Example: $F(a,b) = a' \cdot b + b'$

$G(x,y,z) = x \cdot (y+z')$

Basic Logic Operators

- 1-bit logic AND resembles binary multiplication:

$$0 \cdot 0 = 0, \quad 0 \cdot 1 = 0,$$

$$1 \cdot 0 = 0, \quad 1 \cdot 1 = 1$$

- 1-bit logic OR resembles binary addition, except for one operation:

$$0 + 0 = 0, \quad 0 + 1 = 1,$$

$$1 + 0 = 1, \quad 1 + 1 = 1 (\neq 10_2)$$

Truth Tables for logic operators

Truth table tabular form that uniquely represents the relationship between the input variables of a function and its output

2-Input **AND**

A	B	F=A·B
0	0	0
0	1	0
1	0	0
1	1	1

2-Input **OR**

A	B	F=A+B
0	0	0
0	1	1
1	0	1
1	1	1

NOT

A	F=A'
0	1
1	0

Truth Tables (cont.)

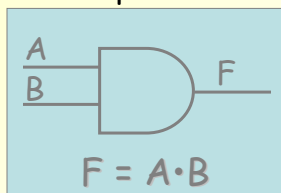
- Q: Let a function $F()$ depend on n variables. How many rows are there in the truth table of $F()$?

A: 2^n rows, since there are 2^n possible binary patterns/combinations for the n variables

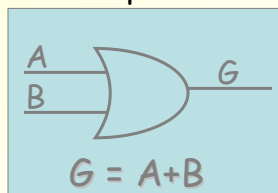
Logic Gates

- Logic gates are abstractions of electronic circuit components that operate on one or more input signals to produce an output signal.

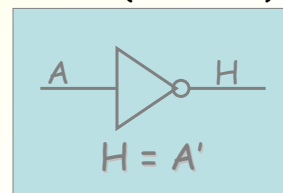
2-Input AND



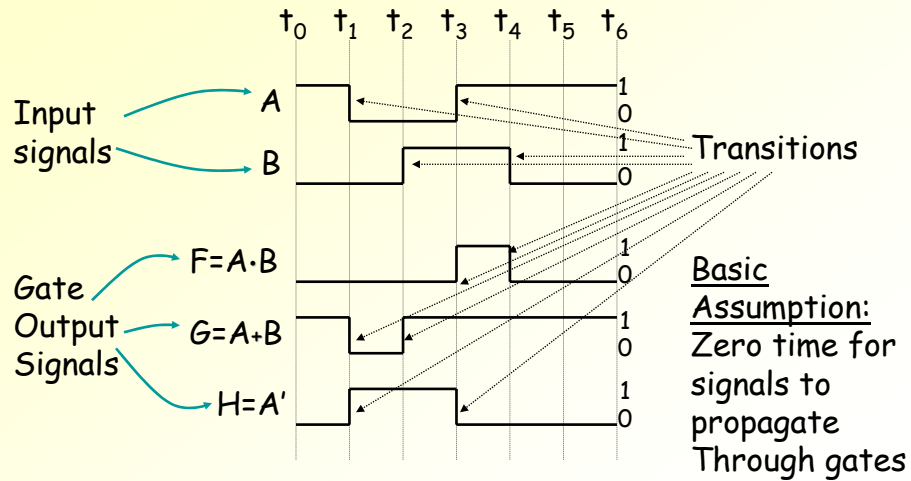
2-Input OR



NOT (Inverter)

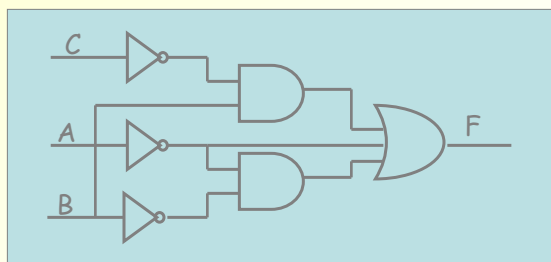


Timing Diagram



Combinational Logic Circuit from Logic Function

- Consider function $F = A' + B \cdot C' + A' \cdot B'$
- A combinational logic circuit can be constructed to implement F, by appropriately connecting input signals and logic gates:
 - Circuit input signals \rightarrow from function variables (A, B, C)
 - Circuit output signal \rightarrow function output (F)
 - Logic gates \rightarrow from logic operations

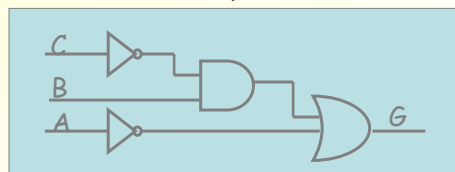
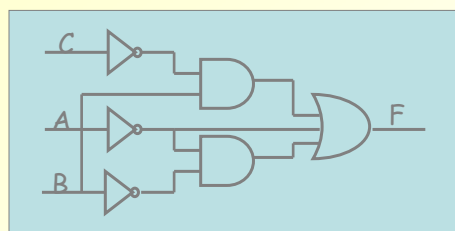


Combinational Logic Circuit from Logic Function (cont.)

- In order to design a cost-effective and efficient circuit, we must minimize the circuit's size (area) and propagation delay (time required for an input signal change to be observed at the output line)
- Observe the truth table of $F=A' + B \cdot C' + A' \cdot B'$ and $G=A' + B \cdot C'$
- Truth tables for F and G are identical \rightarrow same function
- Use G to implement the logic circuit (less components)

A	B	C	F	G
0	0	0	1	1
0	0	1	1	1
0	1	0	1	1
0	1	1	1	1
1	0	0	0	0
1	0	1	0	0
1	1	0	1	1
1	1	1	0	0

Combinational Logic Circuit from Logic Function (cont.)



TUGAS – 4

***CARI DATA SHEET DARI IC CMOS & TTL
SERIES DI INTERNET :***

**74_{LS}00, 74_{LS}02, 74_{LS}04, 74_{LS}08,
74_{LS}10, 74_{LS}11, 74_{LS}20, 74_{LS}21,
74_{LS}27, 74_{LS}30, 74_{LS}32, 74_{LS}86**