Unit 7

Multi-level Gate Circuits
NAND and NOR Gates

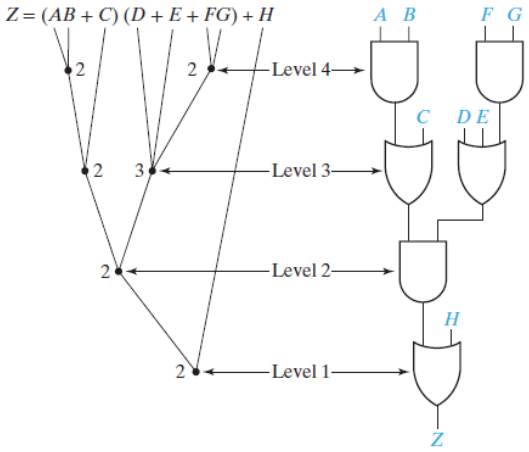
Logic Circuits (Spring 2022)

Multi-level Circuits

- Number of *levels* of gates
  - The maximum number of gates cascaded in series between a circuit input and the output
  - Inverters are normally NOT counted ⇐ most inputs are driven from flip-flop outputs, and the complements are provided with its original value
- AND-OR circuits: a two-level circuit
  - A level of AND gates
  - Followed by an OR gate at the output
- OR-AND circuit: a two-level circuit
- OR-AND-OR circuit: a three-level circuit
  - A level of OR gates
  - Followed by a level of AND gates
  - Followed by an OR gate at the output

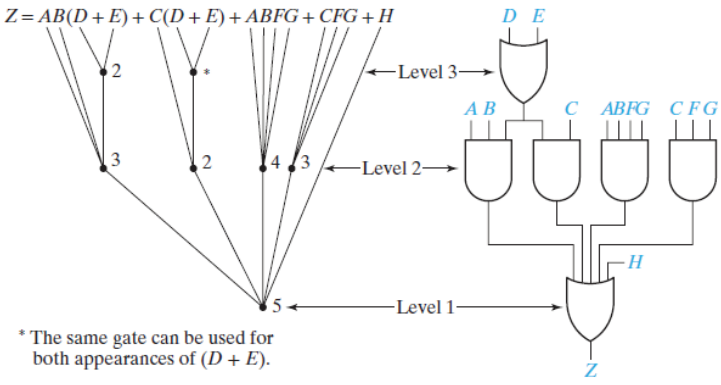
Example 1: Four-level Design

$$Z = (AB + C)(D + E + FG) + H$$



Example 1: Three-level Design

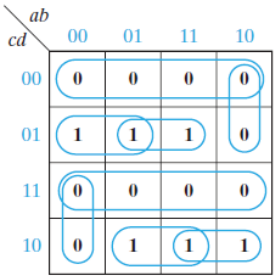
$$Z = AB(D + E) + C(D + E) + ABFG + CFG + H$$

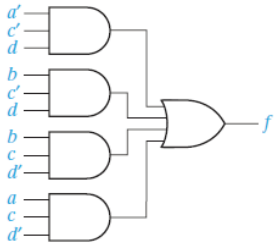


Example 2: AND-OR Circuit

$f(a, b, c, d) = \Sigma\, m(1, 5, 6, 10, 13, 14)$

$f = a'c'd + bc'd + bcd' + acd'$





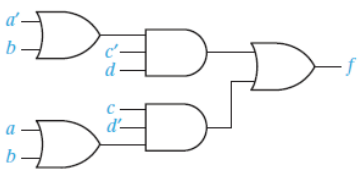
# of gate inputs =

Example 2: OR-AND-OR Circuit

$f(a, b, c, d) = \Sigma\, m(1, 5, 6, 10, 13, 14)$

$f = a'c'd + bc'd + bcd' + acd'$

$f = c'd(a' + b) + cd'(a + b)$



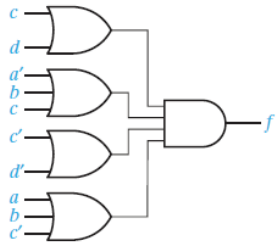
# of gate inputs =

Example 2: OR-AND Circuit

$f(a, b, c, d) = \Sigma\, m(1, 5, 6, 10, 13, 14)$

$f' = c'd' + ab'c' + cd + a'b'c$

$f = (c + d)(a' + b + c)(c' + d')(a + b + c')$



# of gate inputs =

Example 2: AND-OR-AND Circuit

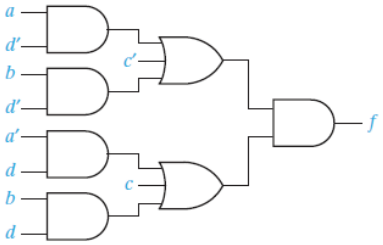
$f(a, b, c, d) = \Sigma\, m(1, 5, 6, 10, 13, 14)$

$f' = c'd' + ab'c' + cd + a'b'c$

$f = (c + d)(a' + b + c)(c' + d')(a + b + c')$

$f = [c + d(a' + b)][c' + d'(a + b)]$

$f = (c + a'd + bd)(c' + ad' + bd')$



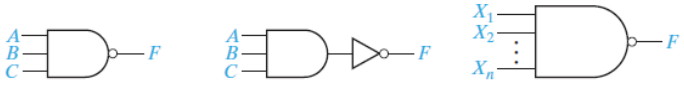
# of gate inputs =

NAND Gates

- NAND Gate
  - Equivalent to an AND gate followed by an inverter
  - AND-NOT gate
- *n*-input NAND gate's output

$$F = (X_1 X_2 \ldots X_n)' = X_1' + X_2' + \cdots + X_n'$$

- Symbols



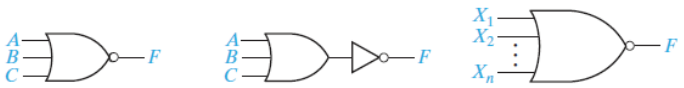
(a) Three-input NAND gate (b) NAND gate equivalent (c) *n*-input NAND gate

NOR Gates

- NOR Gate
  - Equivalent to an OR gate followed by an inverter
  - OR-NOT gate
- *n*-input NOR gate's output

$$F = (X_1 + X_2 + \cdots + X_n)' = X_1' X_2' \ldots X_n'$$

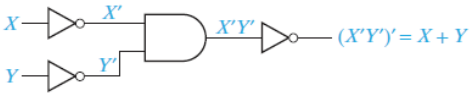
- Symbols



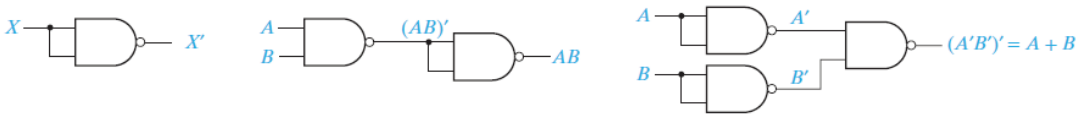
(a) Three-input NOR gate (b) NOR gate equivalent (c) *n*-input NOR gate

Functionally Complete Operations & Gates

- Functionally complete set of operations
  - A set of logic operations is said to be *functionally complete* if any Boolean function can be expressed in terms of this set of operations
  - A simple example set: { AND, OR, NOT }
  - Another example set: { AND, NOT }

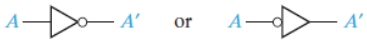


- Functionally complete gate
  - If a single gate forms a functionally complete set by itself, then any switching function can be realized using only gates of that type
  - Example: NAND

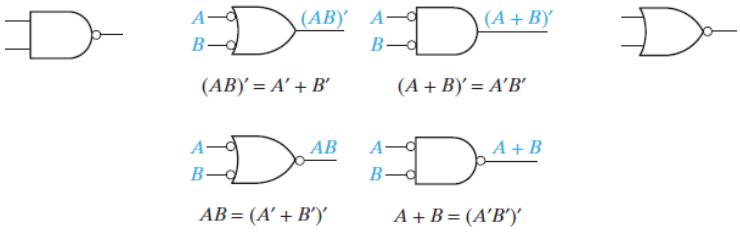


Alternative Gate Symbols

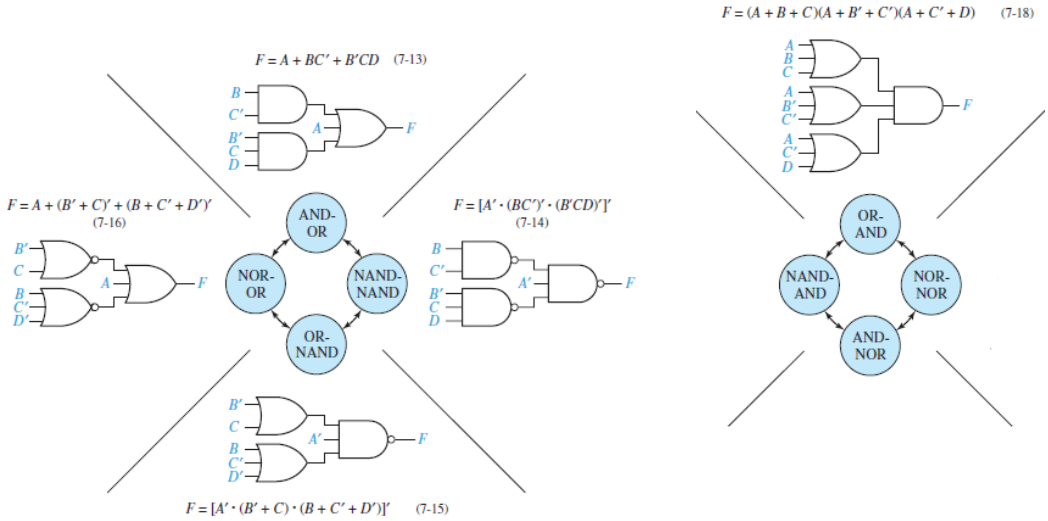
- Inverters



- AND, OR, NAND, NOR

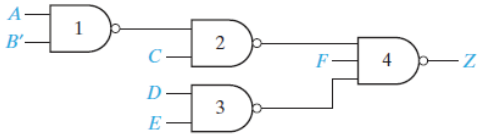


Eight Basic Forms for Two-Level Circuits

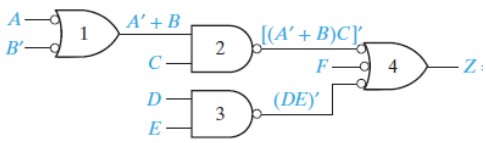


NAND Gate Circuit Conversion

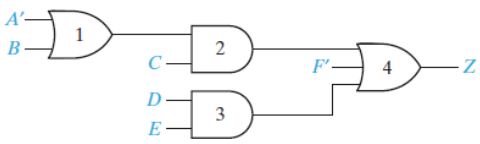
- NAND gates ⇨ AND-OR network



(a) NAND gate network



(b) Alternate form for NAND gate network

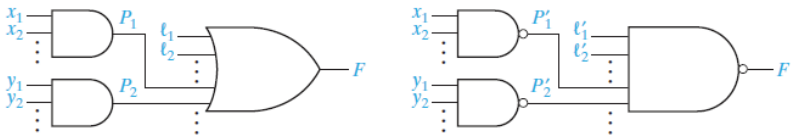


$Z = (A' + B)C + F' + DE$

(c) Equivalent AND-OR network

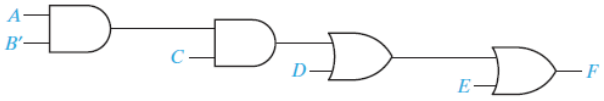
NAND Gate Circuit Conversion

- Two-level AND-OR network ⇨ NAND gates
  - Find a minimum sum-of-products expression
  - Draw the corresponding two-level AND-OR circuit
  - Replace all gates with NAND gates leaving the gate interconnection unchanged
  - If the output gate has any single literals as inputs, complement these literals

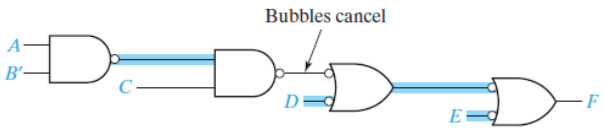


NAND Gate Circuit Conversion

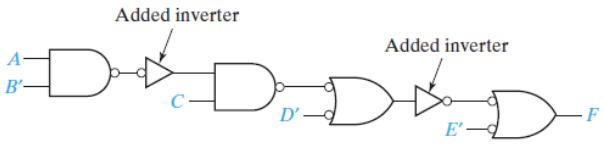
- AND-OR network ⇨ NAND gates



(a) AND-OR network



(b) First step in NAND conversion



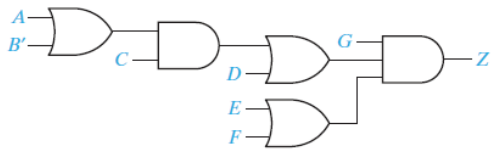
(c) Completed conversion

NOR Gate Circuit Conversion

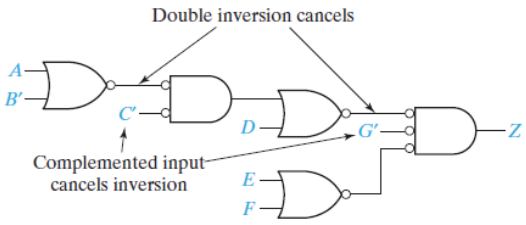
- Two-level OR-AND network ⇨ NOR gates
  - Find a minimum product-of-sums expression
  - Draw the corresponding two-level OR-AND circuit
  - Replace all gates with NOR gates leaving the gate interconnection unchanged
  - If the output gate has any single literals as inputs, complement these literals

NOR Gate Circuit Conversion

- OR-AND network ⇨ NOR gates



(a) Circuit with OR and AND gates



(b) Equivalent circuit with NOR gates

Multi-level Design Using NAND & NOR Gates

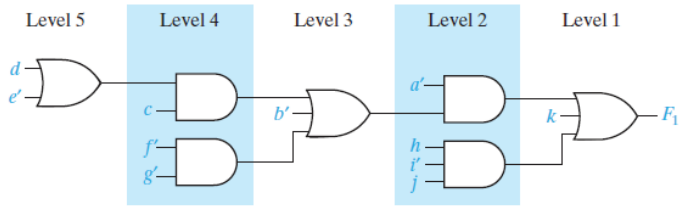
- Procedure to design a multi-level NAND gate circuits
  - Simplify the switching function to be realized
  - Design a multi-level circuit of AND & OR gates. The output gate must be OR. AND gate outputs cannot be used as AND gate inputs. OR gate outputs cannot be used as OR gate inputs
  - Number the levels starting with the output gate as level 1. Replace all gates with NAND gates, leaving all interconnection between gates unchanged.
  - Leave the inputs to level 2, 4, 6, … unchanged. Invert any literals which appear as inputs to levels 1, 3, 5, …

···· – AND – OR – AND – OR

4 3 2 1

Multi-level Design Using NAND & NOR Gates

$$F_1 = a'[b' + c(d + e') + f'g'] + hi'j + k$$



(a) AND-OR network