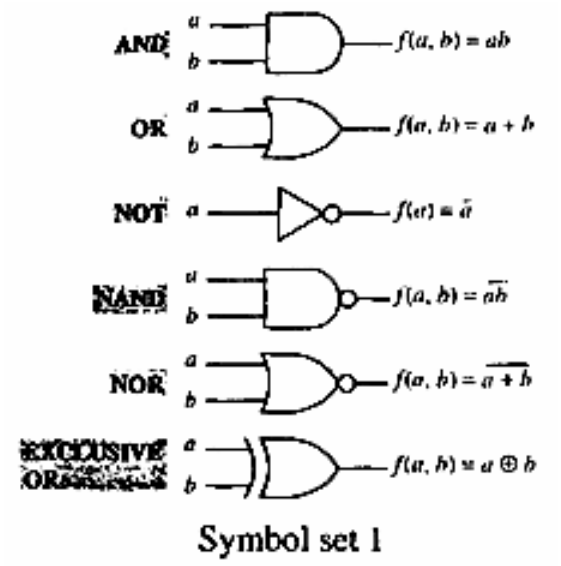
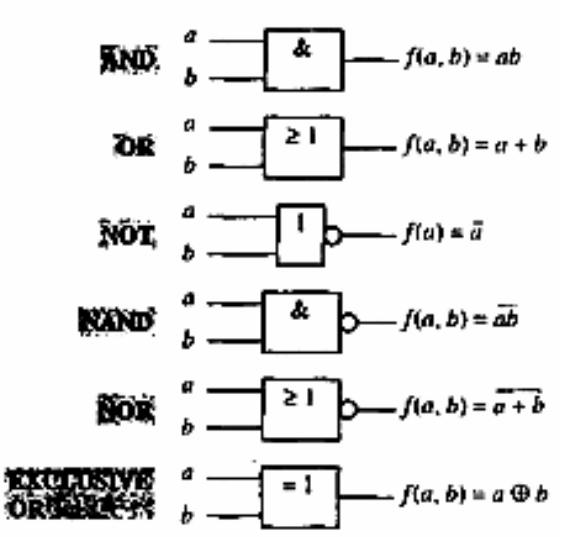
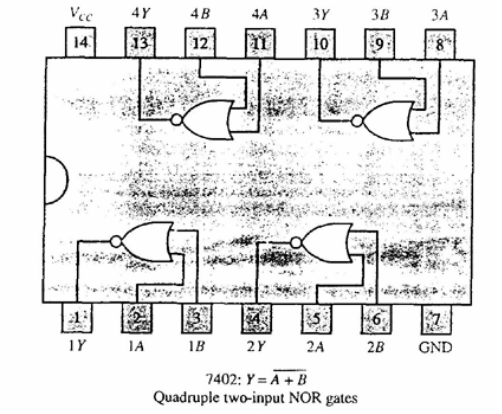
Algebraic and Synthesis (Cont.) Electronic Logic Gates

- Gate Symbols

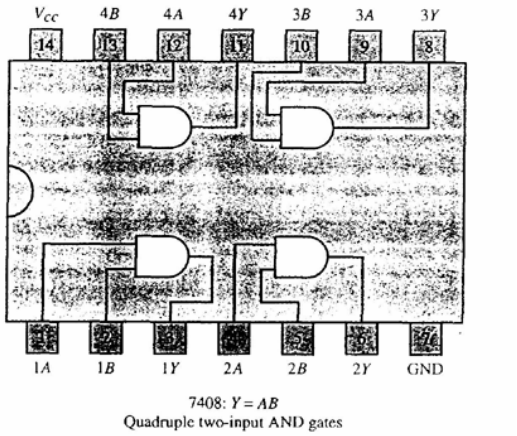




Electronic Logic Gates



7402: $Y = \overline{A + B}$
Quadruple two-input NOR gates



7408: $Y = AB$
Quadruple two-input AND gates

Electronic Logic Gates

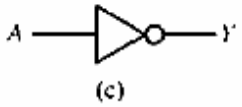
- Basic Function Component

a	$f_{NOT}(a) = \bar{a}$
0	1
1	0

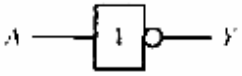
(a)

A	Y
L	H
H	L

(b)



(c)



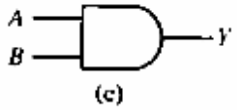
(d)

a	b	$f_{AND}(a, b) = ab$
0	0	0
0	1	0
1	0	0
1	1	1

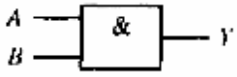
(a)

A	B	Y
L	L	L
L	H	L
H	L	L
H	H	H

(b)



(c)



(d)

Electronic Logic Gates

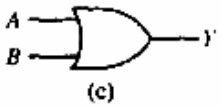
- Basic Function Component

a	b	$f_{OR}(a, b) = a + b$
0	0	0
0	1	1
1	0	1
1	1	1

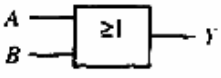
(a)

A	B	Y
L	L	L
L	H	H
H	L	H
H	H	H

(b)



(c)



(d)

a b	sum(a, b)	sum$(a, b) \geq 1$?	$f_{OR}(a, b) = a + b$
0 0	0	False	0
0 1	1	True	1
1 0	1	True	1
1 1	2	True	1

Electronic Logic Gates

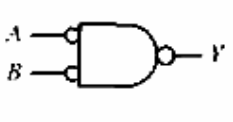
- Positive VS. Negative Logic

$$
\begin{aligned}
y &= a \cdot b \\
&= \overline{\overline{a \cdot b}} \\
&= \overline{\bar{a} + \bar{b}} \\
&= \bar{f}_{OR}(\bar{a}, \bar{b})
\end{aligned}
$$

Electronic Logic Gates

- Positive VS. Negative Logic

$$
\begin{aligned}
y &= a + b \\
&= \overline{\overline{a + b}} \\
&= \overline{\bar{a} \cdot \bar{b}} \\
&= \bar{f}_{AND}(\bar{a}, \bar{b})
\end{aligned}
$$



(b)

Electronic Logic Gates

- NAND Gates

a	b	$f_{\text{NAND}}(a, b) = \overline{ab}$
0	0	1
0	1	1
1	0	1
1	1	0

(a)

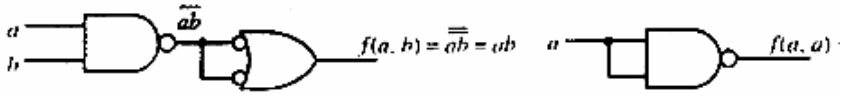
A	B	Y
L	L	H
L	H	H
H	L	H
H	H	L

(b)

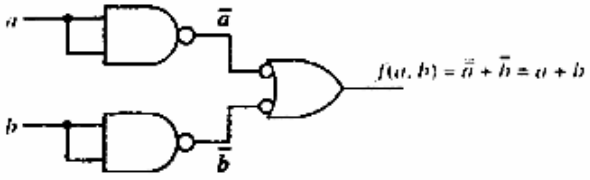
(c) (d) (e)

Electronic Logic Gates

$$f_{\text{NAND}}(a, a) = \overline{a \cdot a} = \bar{a} \quad = f_{\text{NOT}}(a)$$
$$f_{\text{NAND}}(a, b) = \overline{\overline{a \cdot b}} = a \cdot b \quad = f_{\text{AND}}(a, b)$$
$$f_{\text{NAND}}(\bar{a}, \bar{b}) = \overline{\bar{a} \cdot \bar{b}} = a + b = f_{\text{OR}}(a, b)$$



AND gate NOT gate



OR gate

Electronic Logic Gates

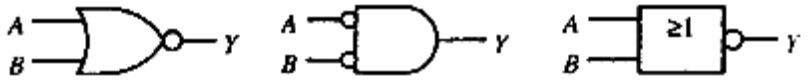
- NOR Gates

a	b	$f_{NOR}(a,b) = \overline{a+b}$
0	0	1
0	1	0
1	0	0
1	1	0

(a)

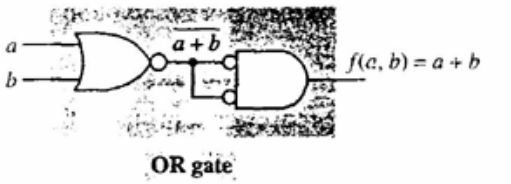
A	B	Y
L	L	H
L	H	L
H	L	L
H	H	L

(b)

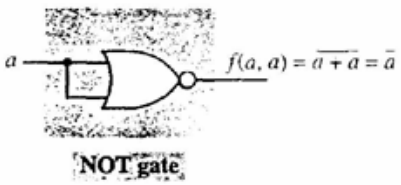


Electronic Logic Gates

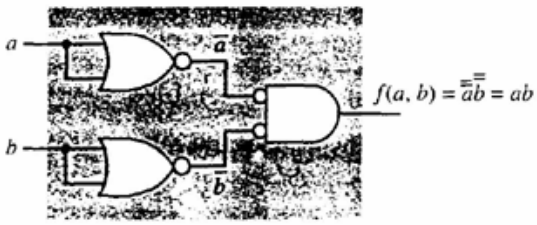
$$f_{NOR}(a,a) = \overline{a+a} = \bar{a} \quad = f_{NOT}(a)$$
$$f_{NOR}(a,b) = \overline{\overline{a+b}} = a+b = f_{OR}(a,b)$$
$$f_{NOR}(\bar{a},\bar{b}) = \overline{\bar{a}+\bar{b}} = a \cdot b \quad = f_{AND}(a,b)$$



OR gate



NOT gate



AND gate

Electronic Logic Gates

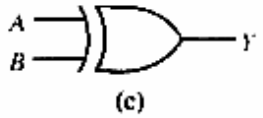
- Exclusive OR Gates (XOR Gates)

a	b	$f_{XOR}(a, b) = a \oplus b$
0	0	0
0	1	1
1	0	1
1	1	0

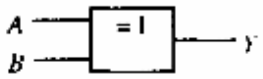
(a)

A	B	Y
L	L	L
L	H	H
H	L	H
H	H	L

(b)



(c)



(d)

$$
\begin{aligned}
a \oplus b &= \bar{a}b + a\bar{b} \\
&= \bar{a}a + \bar{a}b + a\bar{b} + b\bar{b} && \text{[P2(a), P6(b)]} \\
&= \bar{a}(a + b) + \bar{b}(a + b) && \text{[P5(b)]} \\
&= (\bar{a} + \bar{b})(a + b) && \text{[P5(b)]}
\end{aligned}
$$

$$
\begin{aligned}
a \oplus a &= 0 \\
a \oplus \bar{a} &= 1 \\
a \oplus 0 &= a \\
a \oplus 1 &= \bar{a}
\end{aligned}
$$

Electronic Logic Gates

$$
\begin{aligned}
a \oplus a &= 0 \\
a \oplus \bar{a} &= 1 \\
a \oplus 0 &= a \\
a \oplus 1 &= \bar{a}
\end{aligned}
$$

a b	sum(a, b)	sum(a, b) = 1?	$f(a, b) = a \oplus b$
0 0	0	False	0
0 1	1	True	1
1 0	1	True	1
1 1	2	False	0

Electronic Logic Gates

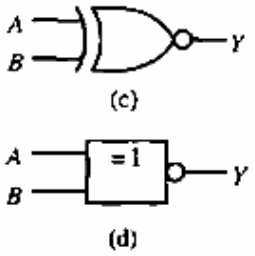
- Exclusive NOR Gates (XNOR Gates)

a	b	$f_{XNOR}(a,b) = a \odot b$
0	0	1
0	1	0
1	0	0
1	1	1

(a)

A	B	Y
L	L	H
L	H	L
H	L	L
H	H	H

(b)



$$a \odot b = \overline{a \oplus b}$$
$$= \overline{\bar{a}b + a\bar{b}}$$
$$= \overline{\bar{a}b} \cdot \overline{a\bar{b}}$$
$$= (a + \bar{b})(\bar{a} + b)$$
$$= a\bar{a} + ab + \bar{a}\bar{b} + \bar{b}b$$
$$= ab + \bar{a}\bar{b}$$

It can also be easily verified that

$$a \oplus \bar{b} = a \odot b$$

Analysis of Combinational Circuits

- Algebraic Methods

$$P_1 = \overline{ab}$$
$$P_2 = \overline{\bar{a} + c}$$

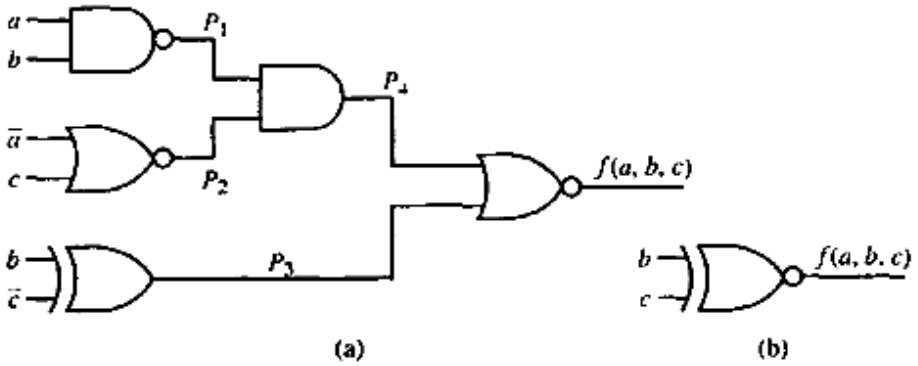
$$P_3 = b \oplus \bar{c}$$
$$P_4 = P_1 \cdot P_2 = \overline{ab}\ \overline{(\bar{a} + c)}$$

$$f(a,b,c) = \overline{P_3 + P_4}$$
$$= \overline{(b \oplus \bar{c}) + \overline{ab}\ \overline{(\bar{a} + c)}}$$

$$\bar{f}(a,b,c) = (b \oplus \bar{c}) + \overline{ab}\ \overline{\bar{a} + c}$$
$$= bc + \bar{b}\bar{c} + \overline{ab}\ \overline{\bar{a} + c}$$
$$= bc + \bar{b}\bar{c} + (\bar{a} + \bar{b})a\bar{c}$$
$$= bc + \bar{b}\bar{c} + a\bar{b}\bar{c}$$
$$= bc + \bar{b}\bar{c}$$
$$\bar{f}(a,b,c) = b \odot c$$

$$f(a,b,c) = \overline{b \odot c} = b \oplus c$$

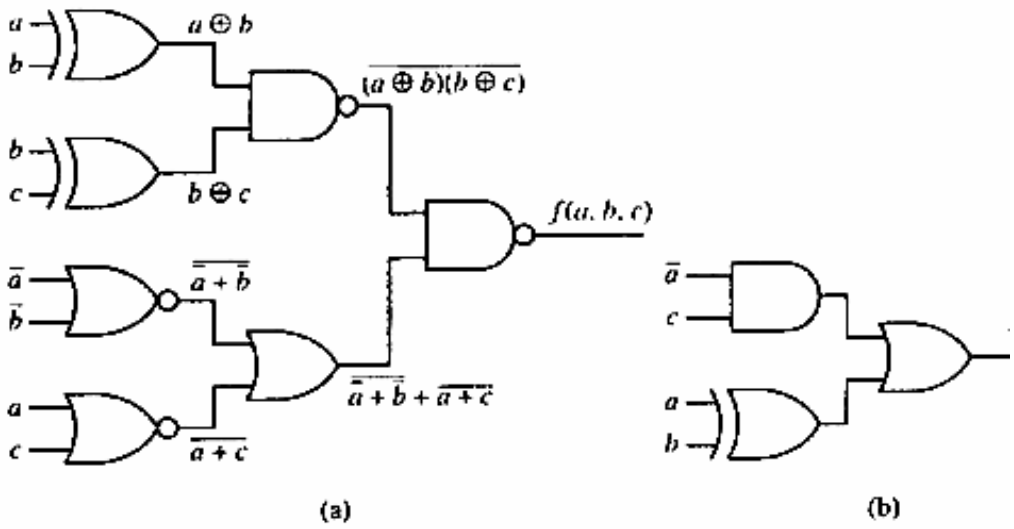
Analysis of Combinational Circuits



Analysis of Combinational Circuits

$$
\begin{aligned}
&f(a,b,c)\\
&= \overline{\overline{(a \oplus b)(b \oplus c)} \cdot (\bar{a} + \bar{b} + \overline{a + c})}\\
&= \overline{\overline{(a \oplus b)(b \oplus c)}} + \overline{\bar{a} + \bar{b}} + \overline{\overline{a + c}}\\
&= (a \oplus b)(b \oplus c) + (\bar{a} + \bar{b})(a + c)\\
&= (a\bar{b} + \bar{a}b)(b\bar{c} + \bar{b}c) + (\bar{a} + \bar{b})(a + c)\\
&= a\bar{b}b\bar{c} + a\bar{b}\bar{b}c + \bar{a}bb\bar{c} + \bar{a}b\bar{b}c + \bar{a}a + \bar{a}c + a\bar{b} + \bar{b}c\\
&= a\bar{b}c + \bar{a}b\bar{c} + \bar{a}c + a\bar{b} + \bar{b}c\\
&= \bar{a}b\bar{c} + \bar{a}c + a\bar{b} + \bar{b}c\\
&= \bar{a}b\bar{c} + \bar{a}c + a\bar{b}\\
&= \bar{a}b + \bar{a}c + a\bar{b}\\
&= \bar{a}c + a \oplus b
\end{aligned}
$$

Analysis of Combinational Circuits



Analysis of Combinational Circuits

- Truth Table Methods

abc	$\bar{a}c$	$a \oplus b$	$f(a, b, c)$
0 0 0	0	0	0
0 0 1	1	0	1
0 1 0	0	1	1
0 1 1	1	1	1
1 0 0	0	1	1
1 0 1	0	1	1
1 1 0	0	0	0
1 1 1	0	0	0

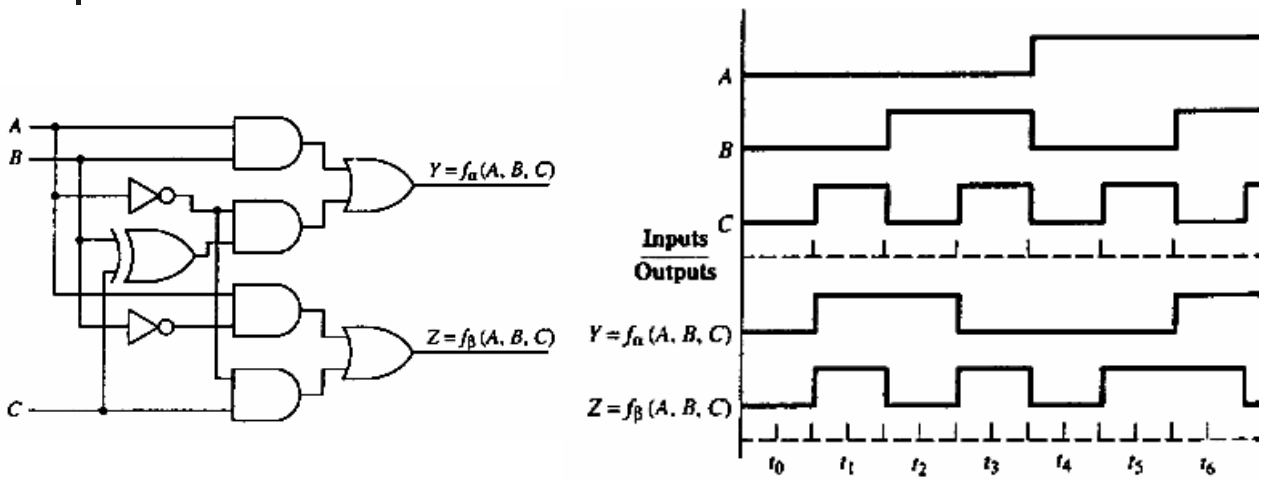
Analysis of Combinational Circuits

- Timing Diagram
  - : is a graphic representation of input and output signal relationship in switching network.

$$f_\alpha(A, B, C) = \sum m(1, 2, 6, 7)$$
$$= \bar{A}\bar{B}C + \bar{A}B\bar{C} + AB\bar{C} + ABC$$
$$= \bar{A}\bar{B}C + B\bar{C} + AB$$

$$f_\beta(A, B, C) = \sum m(1, 3, 5, 6)$$
$$= \bar{A}\bar{B}C + \bar{A}BC + A\bar{B}C + AB\bar{C}$$
$$= \bar{A}C + A\bar{B}C + AB\bar{C}$$

Analysis of Combinational Circuits



Analysis of Combinational Circuits

Time	Inputs ABC	Outputs $f_\alpha(A, B, C)$	$f_\beta(A, B, C)$
t_0	000	0	0
t_1	001	1	1
t_2	010	1	0
t_3	011	0	1
t_4	100	0	0
t_5	101	0	1
t_6	110	1	1
t_7	111	1	0

Analysis of Combinational Circuits

- Propagation Delays
  - The Delay between time of a input change and the corresponding output change.

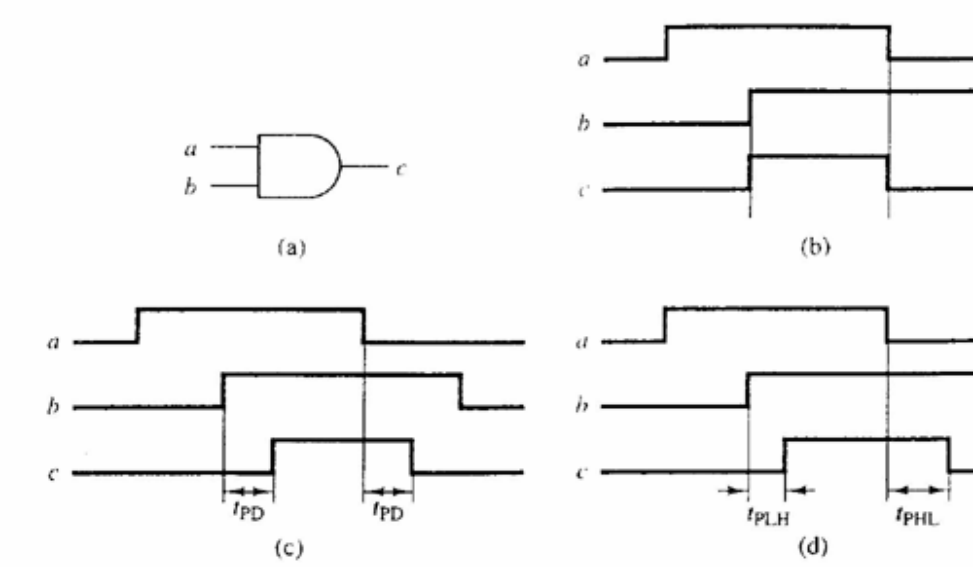
t_{PLH} = propagation delay time, low- to high-level output

t_{PHL} = propagation delay time, high- to low-level output

  - Single propagation delay parameter , t_{PD}

$$t_{PD} = \frac{t_{PLH} + t_{PHL}}{2}$$

Analysis of Combinational Circuits

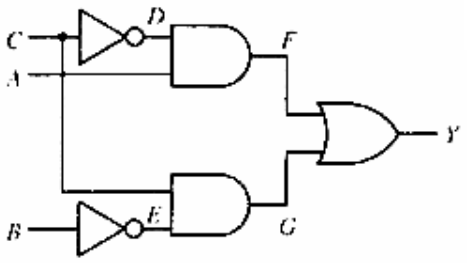


Analysis of Combinational Circuits

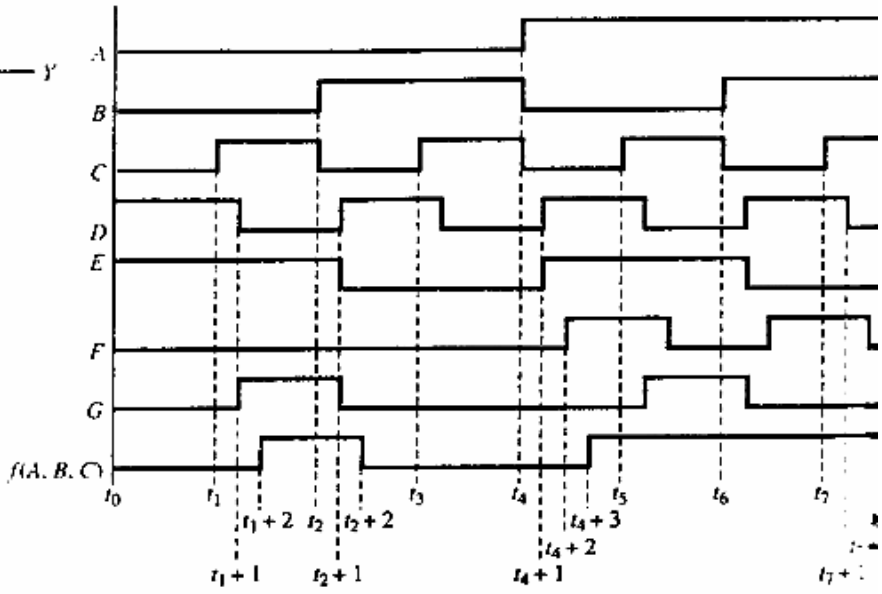
Logic Family	Propagation Delay t_{PD} (ns)	Power Dissipation Per Gate (mW)
7400	10	10
74H00	6	22
74L00	33	1
74LS00	9.5	2
74S00	3	19
74ALS00	3.5	1.3
74AS00	3	8
74HC00	8	0.17

Analysis of Combinational Circuits

$$f(A, B, C) = \sum m(1, 4, 5, 6)$$
$$= \bar{A}\bar{B}C + A\bar{B}\bar{C} + A\bar{B}C + AB\bar{C}$$
$$= A\bar{C} + \bar{B}C$$



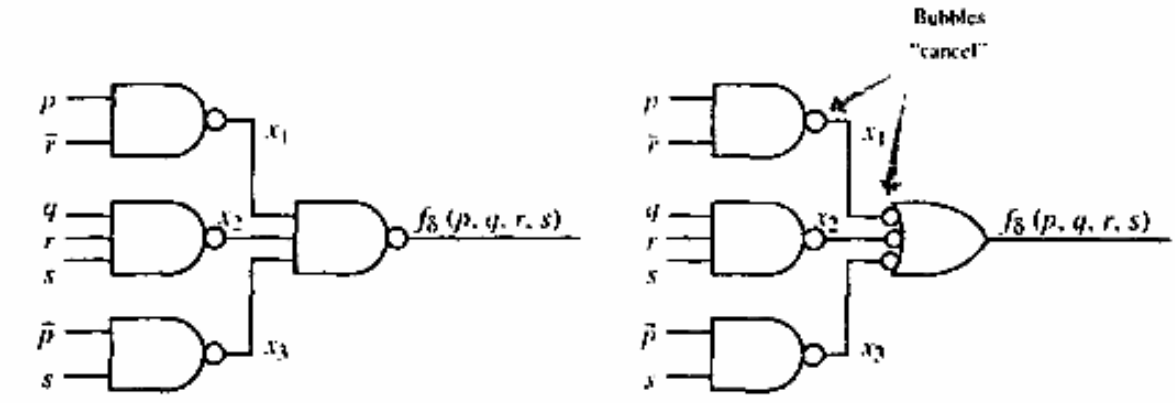
A B C	f(A, B, C)
0 0 0	0
0 0 1	1
0 1 0	0
0 1 1	0
1 0 0	1
1 0 1	1
1 1 0	1
1 1 1	0



Synthesis of Combinational Logic Circuit

- AND-OR and NAND Networks

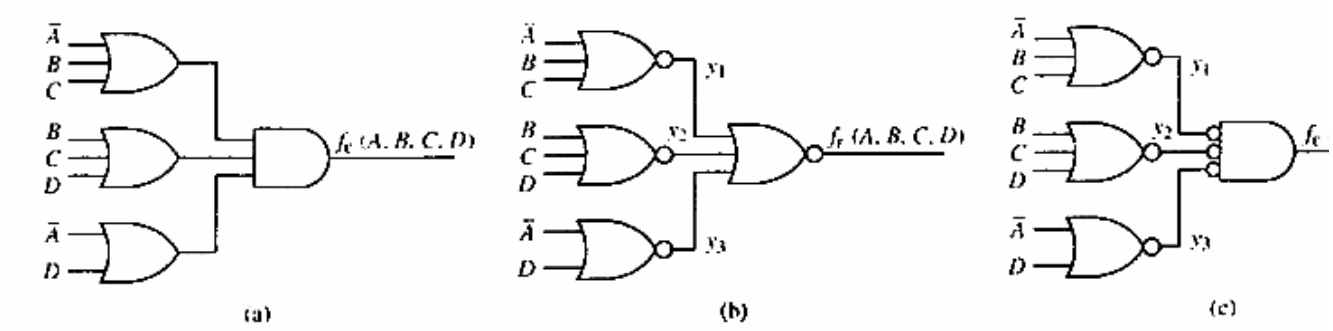
$$f_8(p, q, r, s) = p\bar{r} + qrs + \bar{p}s$$



Synthesis of Combinational Logic Circuit

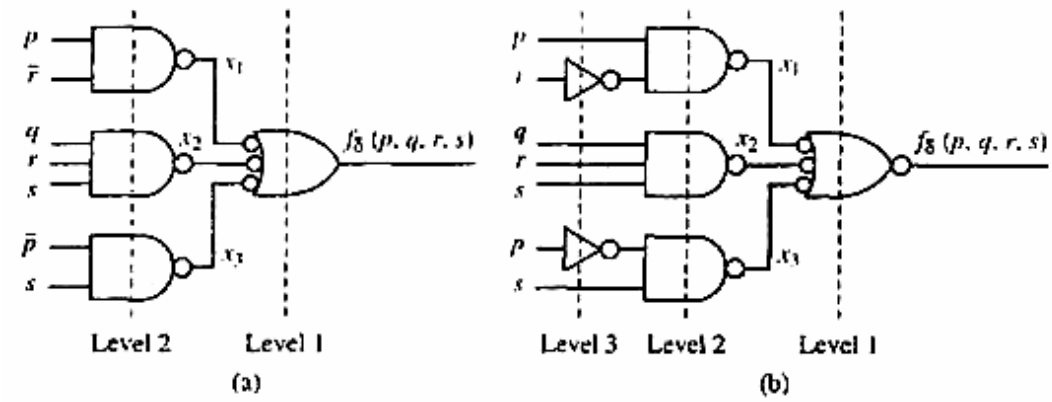
- OR-AND and NOR Networks

$$f_e(A, B, C, D) = (\bar{A} + B + C)(B + C + D)(\bar{A} + D)$$



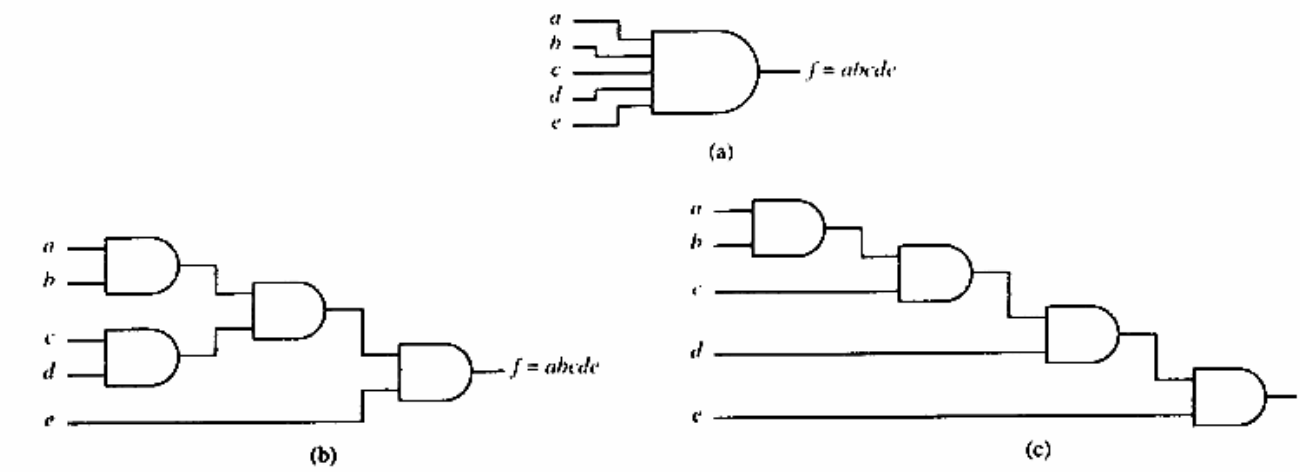
Synthesis of Combinational Logic Circuit

- Two Level Network



Synthesis of Combinational Logic Circuit

$$f(a,b,c,d,e) = abcde$$



Synthesis of Combinational Logic Circuit

- Implement in NAND or NOR Logic
  - STEP 1: Express the function in minterm (maxterm) list form.
  - STEP 2: Write out the minterm (maxterm) in algebraic form.
  - STEP 3: Simplify the function in SOP (POS) from.
  - STEP 4: Use Theorem 8a(8b) and T3 to transform the expression into the NAND (NOR) formuation.
  - STEP 5: Draw the NAND (NOR) logic diagram.

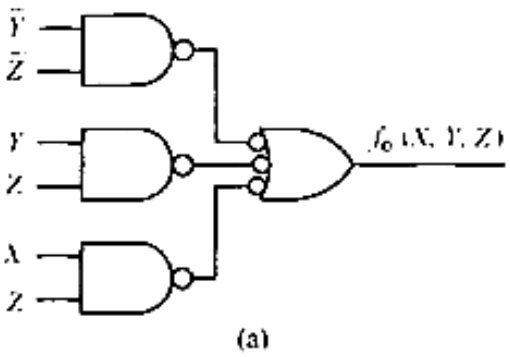
Synthesis of Combinational Logic Circuit

Implement $f_\phi(X,Y,Z) = \sum m(0,3,4,5,7)$ in NAND logic.

1. $f_\phi(X,Y,Z) = \sum m(0,3,4,5,7)$
2. $f_\phi(X,Y,Z) = m_0 + m_3 + m_4 + m_5 + m_7$
   $= \bar{X}\bar{Y}\bar{Z} + \bar{X}YZ + X\bar{Y}\bar{Z} + X\bar{Y}Z + XYZ$
3. $f_\phi(X,Y,Z) = \bar{Y}\bar{Z} + YZ + XZ$ [T6(a)]
4a. $f_\phi(X,Y,Z) = \overline{\overline{\bar{Y}\bar{Z}}} + \overline{\overline{YZ}} + \overline{\overline{XZ}}$ [T3]

or

4b. $f_\phi(X,Y,Z) = \overline{\overline{\bar{Y}\bar{Z} + YZ + XZ}}$ [T3]
$= \overline{\overline{\bar{Y}\bar{Z}} \cdot \overline{YZ} \cdot \overline{XZ}}$ [T8(a)]



(a)

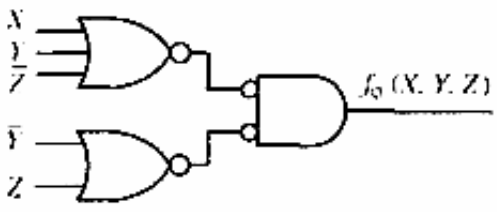
Synthesis of Combinational Logic Circuit

Implement $f_\phi(X,Y,Z) = \sum m(0,3,4,5,7)$ in NOR logic.

1. $f_\phi(X,Y,Z) = \prod M(1,2,6)$
2. $f_\phi(X,Y,Z) = M_1 \cdot M_2 \cdot M_6$
   $= (X + Y + \bar{Z})(X + \bar{Y} + Z)(\bar{X} + \bar{Y} + Z$
3. $f_\phi(X,Y,Z) = (X + Y + \bar{Z})(\bar{Y} + Z)$ [T6(b)]
4a. $f_\phi(X,Y,Z) = \overline{\overline{(X + Y + \bar{Z})}} \cdot \overline{\overline{(\bar{Y} + Z)}}$ [T3]

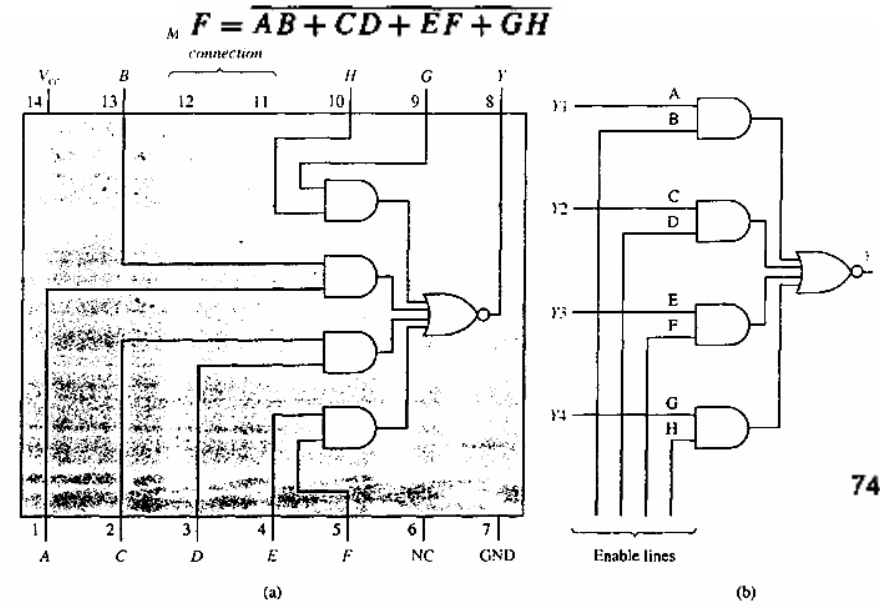
or

4b. $f_\phi(X,Y,Z) = \overline{\overline{(X + Y + \bar{Z})(\bar{Y} + Z)}}$ [T3]
$= \overline{\overline{(X + Y + \bar{Z})} + \overline{(\bar{Y} + Z)}}$ [T8(b)]



Synthesis of Combinational Logic Circuit

- AOI Circuit (AND-OR-INVERT)



7454 2-2-2-2 AND-OR-invert circuit.

Synthesis of Combinational Logic Circuit

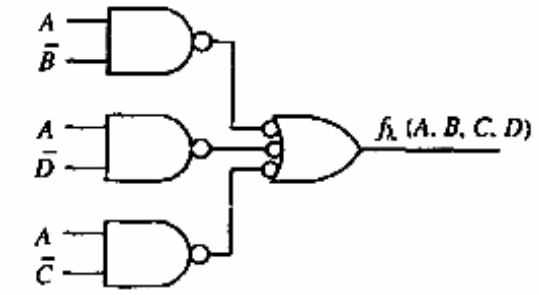
- Factoring
  - Factoring is a technique for obtaining higher-level form of switching function.
  - For require less hardware.
  - For more economical to implement.
  - For limited fan-in must be used.

Synthesis of Combinational Logic Circuit

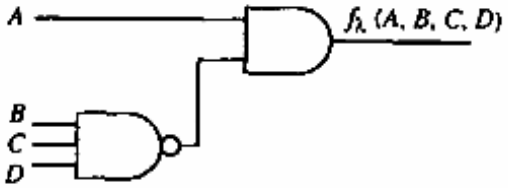
Suppose we are given the following four-variable function:

$$f_2(A, B, C, D) = A\bar{B} + A\bar{D} + A\bar{C}$$

$$\begin{aligned} f_2(A, B, C, D) &= A\bar{B} + A\bar{D} + A\bar{C} \\ &= A(\bar{B} + \bar{D} + \bar{C}) \\ &= A(\overline{BCD}) \end{aligned}$$



(a)



(b)

Synthesis of Combinational Logic Circuit

Realize the function $f(a,b,c,d) = \sum m(8,13)$ using only two-input AND and OR gates.

We begin by writing the canonical SOP form:

$$\begin{aligned} f(a, b, c, d) &= \sum m(8, 13) \\ &= a\bar{b}\bar{c}\bar{d} + ab\bar{c}d \end{aligned}$$

$$\begin{aligned} f(a, b, c, d) &= a\bar{b}\bar{c}\bar{d} + ab\bar{c}d \\ &= (a\bar{c})(bd + \bar{b}\bar{d}) \end{aligned}$$