Basic gates
1. AND
2. OR
3. NOT

Other gates
1. NAND → Not AND $f(A,B) = \overline{A \cdot B}$
2. NOR → Not OR $f(A,B) = \overline{A+B}$
3. XOR → Exclusive OR
4. XNOR → Not Exclusive OR

XOR

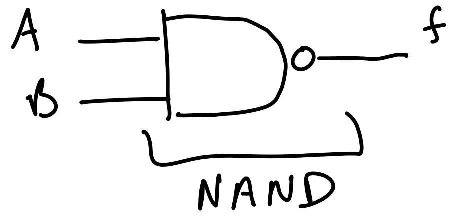
A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

XNOR

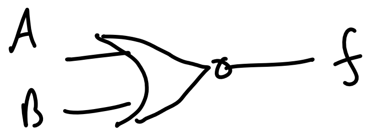
A	B	$\overline{A \oplus B}$
0	0	1
0	1	0
1	0	0
1	1	1

ANSI network symbols

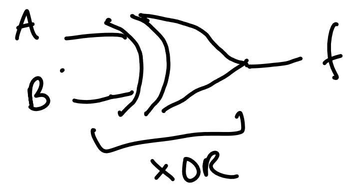
NAND ≡ Not of AND



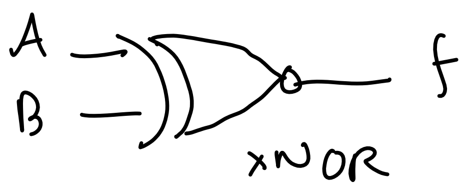
NOR ≡ Not of OR



XOR ≡



XNOR ≡



$$XOR = A \oplus B$$
$$= m_1 + m_2$$
$$= \bar{A}B + A\bar{B}$$

B \ A	0	1
0	0	1
1	1	0

$$XNOR = \overline{A \oplus B}$$
$$= m_0 + m_3$$
$$= \bar{A}\bar{B} + AB$$

B \ A	0	1
0	1	0
1	0	1

NAND/NOR gates + Petricks

Vikas Dhiman for ECE275

September 25, 2023

1 Circuit design using NAND/NOR gates

Example 1. *Implement the function* $f(x_1, x_2, x_3) = \sum m(2, 3, 4, 6, 7)$ *using (1) NAND gates only and (2) NOR gates only.*

NAND gates only $\equiv$ Sum of products (SOP)

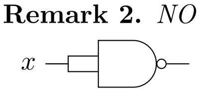
$$f = A\bar{B} + \bar{A}B = \overline{\underset{x}{\boxed{\overline{A\bar{B}}}}} + \overline{\underset{y}{\boxed{\overline{\bar{A}B}}}} \qquad x = \overline{A\bar{B}} \quad y = \overline{\bar{A}B}$$

$$= \bar{x} + \bar{y}$$

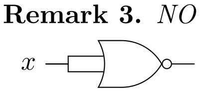
$$= \overline{x \cdot y} \qquad | \text{ By Demorgan's Theorem}$$

Remark 1. *NAND-NAND logic is generated from SOP form. NOR-NOR logic is generated from POS form.*

Remark 2. *NOT gate can also be created from a NAND gate* $\bar{x} = \overline{x \cdot x}$.



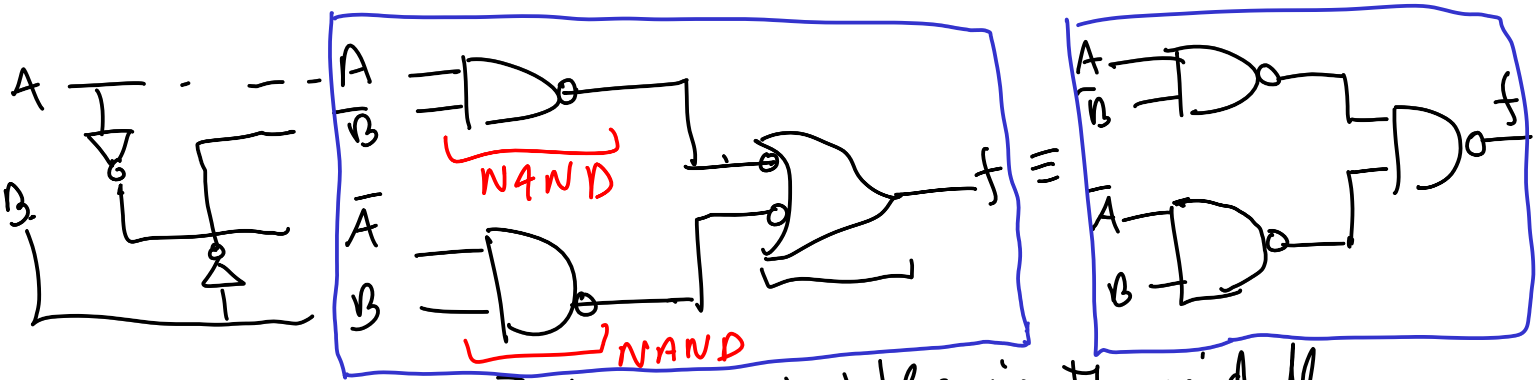
Remark 3. *NOT gate can also be created from a NOR gate* $\bar{x} = \overline{x + x}$.



Problem 1. *Design the simplest circuit that implements the function* $f(x_1, x_2, x_3) = \sum m(3, 4, 6, 7)$ *using (1) NAND gates only (2) NOR gates only.*

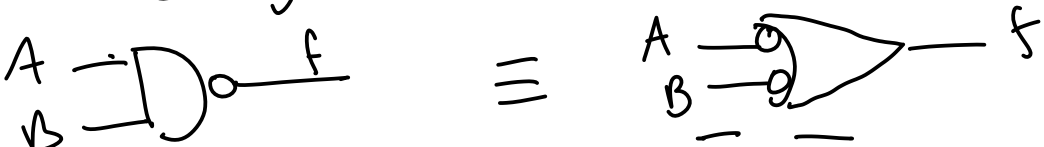
$f = A\bar{B} + \bar{A}B$

Bubble pushing

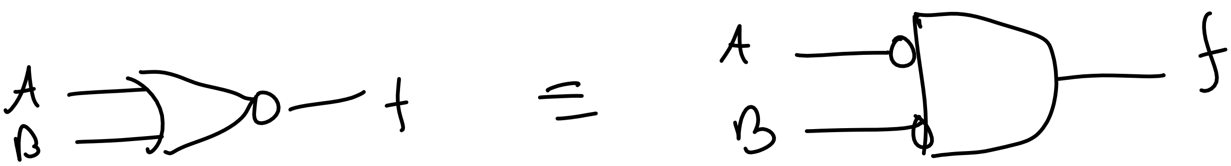


(1) Introduce bubbles in the middle

DeMorgan's theorem



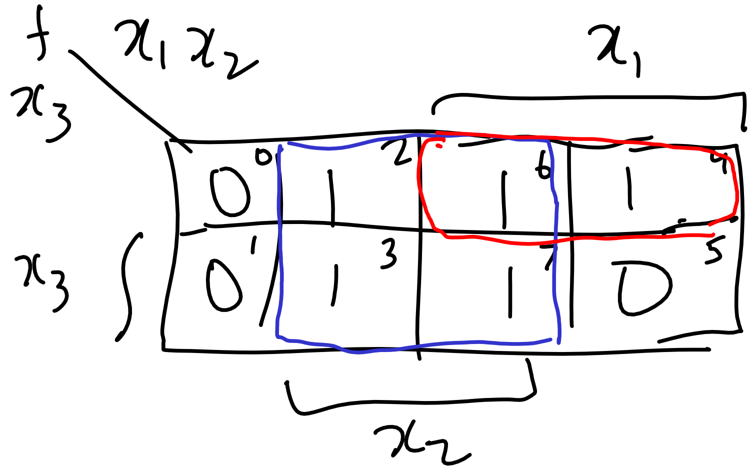
$f = \overline{A.B} = \bar{A} + \bar{B}$



Ex.1) Find the min cost NAND implementation

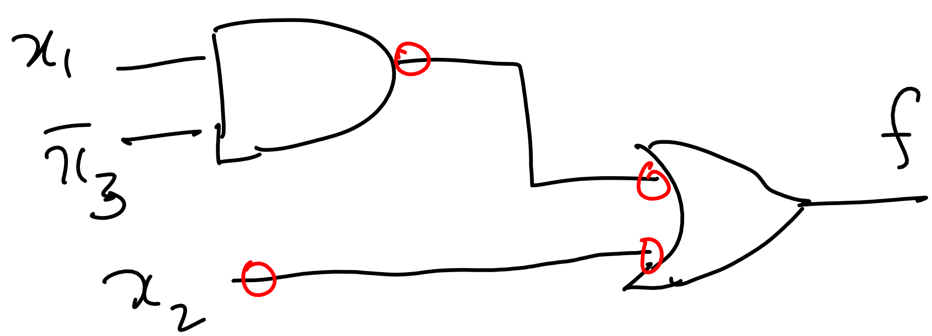
$f(x_1, x_2, x_3) = \sum m(2, 3, 4, 6, 7)$

for NAND, we design SOP implementation

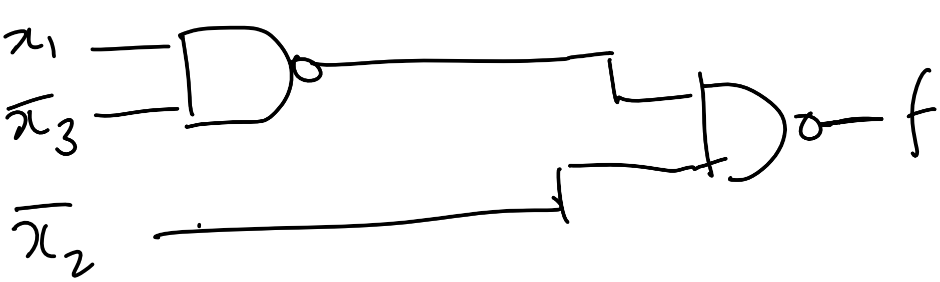


$PI = \{x_2, \ x_1 \bar{x}_3\} = EPI$

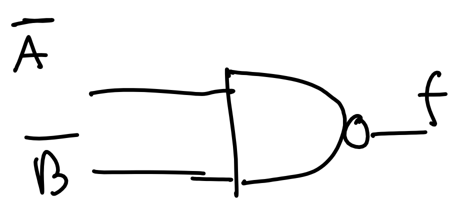
$f = x_2 + x_1 \bar{x}_3$



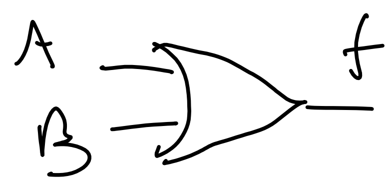
|||



Q



|||



$f = \overline{\bar{A} \cdot \bar{B}}$
$= A + B$

$f(x_1, x_2, x_3) = \sum m(2, 3, 4, 6, 7)$

for NOR implementation find POS

$\overline{f}$

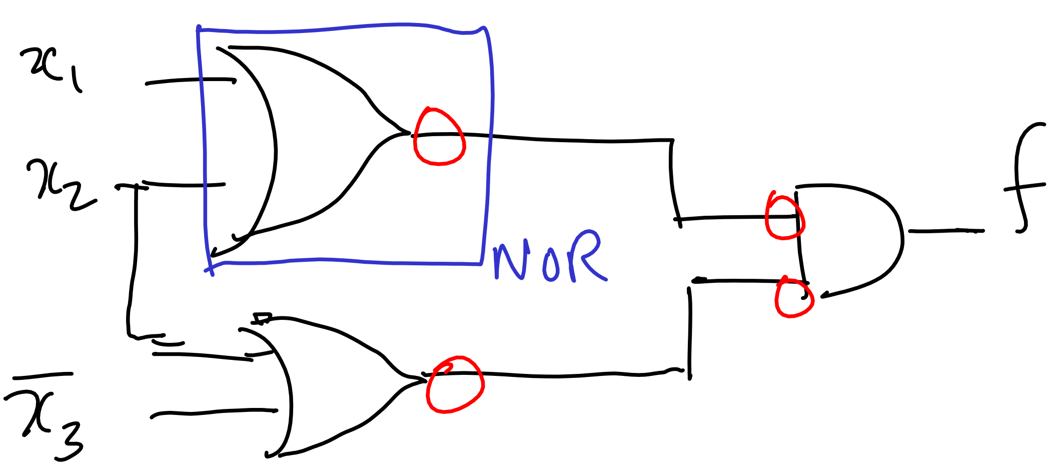
$x_3 \backslash x_1x_2$	00	01	11	10
0	1 (0)	0 (2)	0 (6)	0 (4)
1	1 (1)	0 (3)	0 (7)	1 (5)

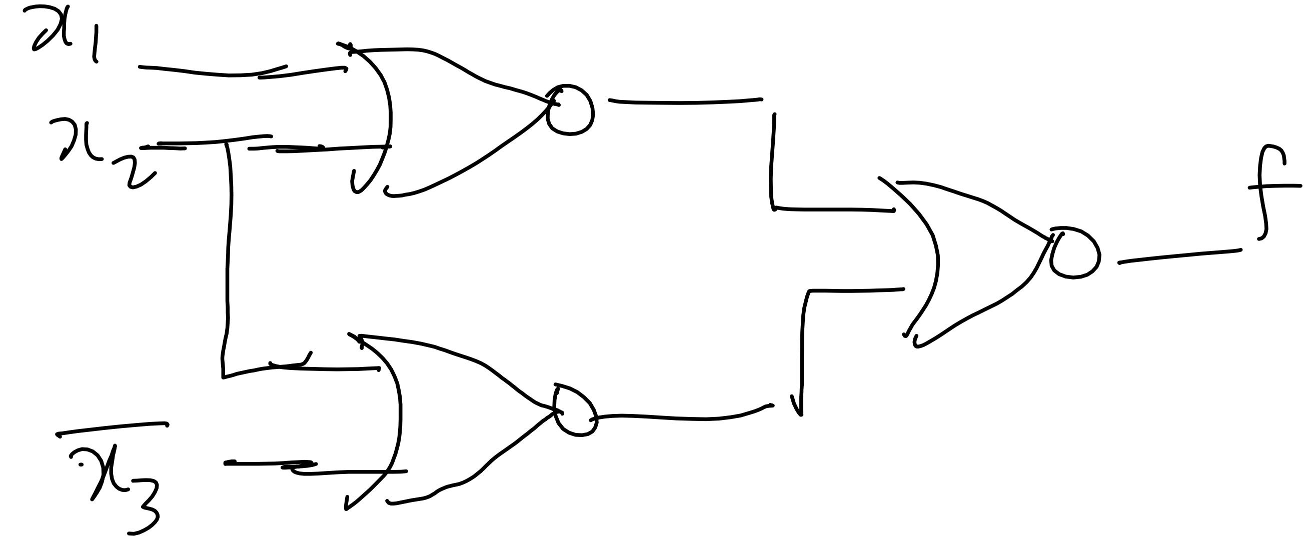
$PI = \{\overline{x}_1\overline{x}_2, \overline{x}_2 x_3\} = EPI$

$\overline{f} = \overline{x}_1\overline{x}_2 + \overline{x}_2 x_3$

$f = \overline{\overline{x}_1\overline{x}_2 + \overline{x}_2 x_3}$

$= (x_1 + x_2)\cdot(x_2 + \overline{x}_3)$





Prob 1 $\quad f(x_1, x_2, x_3) = \sum m(3, 4, 6, 7)$

for NOR find POS

$\bar{f}$

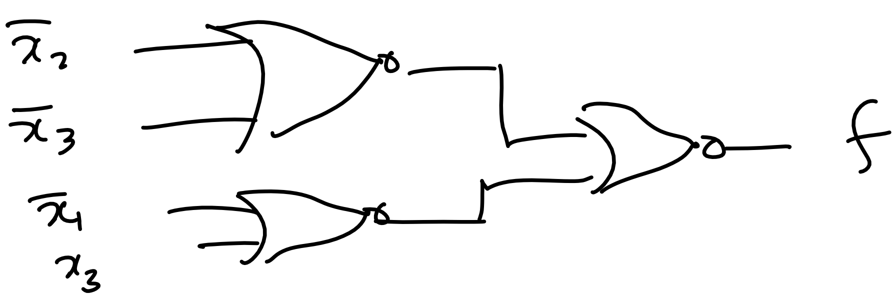
x_3 \			x_1	x_1
	1 (0)	1 (2)	0 (6)	0 (4)
x_3	1 (1)	0 (3)	0 (7)	1 (5)
		x_2	x_2	

$PI = \{ x_2 x_3,\ x_1 x_2,\ x_1 \bar{x}_3 \}$

$EPI = \{ x_2 x_3,\ x_1 \bar{x}_3 \}$

$\bar{f} = x_2 x_3 + x_1 \bar{x}_3$

$f = (\bar{x}_2 + \bar{x}_3)(\bar{x}_1 + x_3)$



2 PI Table reduction and Petrick's method

This is not in the text-book. For additional reading, please refer to the linked resources on the website.

Definition 1 (Implicant)**.** *Given a function f of n variables, a product term P is an implicant of f if and only if for every combination of values of the n variables for which $P = 1$, f is also equal to 1.*

Definition 2 (Prime Implicant)**.** *A prime implicant of a function f is an implicant which is no longer an implicant if any literal is removed from it.*

There are 4 main steps in the Quine-McCluskey algorithm/PI Table reduction and Petrick's method:

1. Generate Prime Implicants
2. Construct Prime Implicant Table. PIs as columns, and minterms as rows (don't cares are excluded).
3. Reduce Prime Implicant Table by repeating following steps until they it cannot be reduced further
   (a) Remove Essential Prime Implicants
   (b) Row Dominance: Remove *dominating* rows. (i.e. unnecessary minterms)
   (c) Column Dominance: Remove *dominated* columns. (i.e. remove unnecessary PIs)
4. Solve Prime Implicant Table by Petrick's method

2.1 Generate Prime Implicants

Example 2. *Generate prime implicants of the function $F(A, B, C, D) = \sum m(0, 2, 5, 6, 7, 8, 10, 12, 13, 14, 15)$ using Quine-McCluskey method*

Steps:

1. Start with writing minterms in binary format (include don't cares as minterms).

2. Create potential groups of minterms that can be combined (merged). The only minterms that can be combined differ only be single 1. Create a new list of combined minterms as n-1 literal implicants.

3. Check off the minterms that could be combined. Unchecked minterms are prime implicants (PIs).

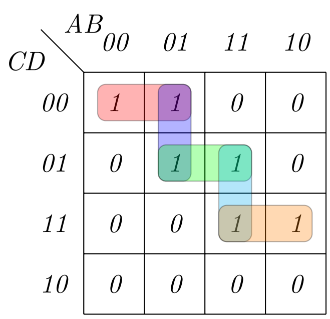
4. Repeat the grouping process with n-1 literal implicants.

Problem 2. *Generate PIs for the function* $F(A, B, C, D) = \sum m(0, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13)$.

2.2 Prime Implicants table and reduction

Example 3. *Reduce the prime implicants* $\{\bar{B}\bar{D}, C\bar{D}, BD, BC, A\bar{D}, AB\}$ *using prime implicants table.*

Example 4.



Example 5.

$CD \backslash AB$	00	01	11	10
00	d	0	0	0
01	1	1	d	d
11	1	1	0	0
10	1	d	0	0

Example 6. *Reduce the following PI table*

	$\bar{A}\bar{D}$	$\bar{B}\bar{D}$	$\bar{C}\bar{D}$	$\bar{A}C$	$\bar{B}C$	$\bar{A}B$	$B\bar{C}$	$A\bar{B}$	$A\bar{C}$
0	X	X	X						
2	X	X		X	X				
3				X	X				
4	X		X			X	X		
5						X	X		
6	X			X		X			
7				X		X			
8		X	X					X	X
9								X	X
10		X			X			X	
11					X			X	
12			X		X		X		X
13							X		X

2.3 Petrick's method

Example 7. *Solve the Prime Implicant table using Petrick's method*

	$p_1 = \bar{A}C$	$p_2 = \bar{B}C$	$p_3 = \bar{A}B$	$p_4 = B\bar{C}$	$p_5 = A\bar{B}$	$p_6 = A\bar{C}$
3	X	X				
5			X	X		
7	X		X			
9					X	X
11		X			X	
13				X		X

Example 8. *Find the minimum SOP expression for the function* $F(A, B, C, D) = \sum m(2, 3, 7, 9, 11, 13) + \sum d(1, 10, 15)$ *using Quine-McCluskey method.*