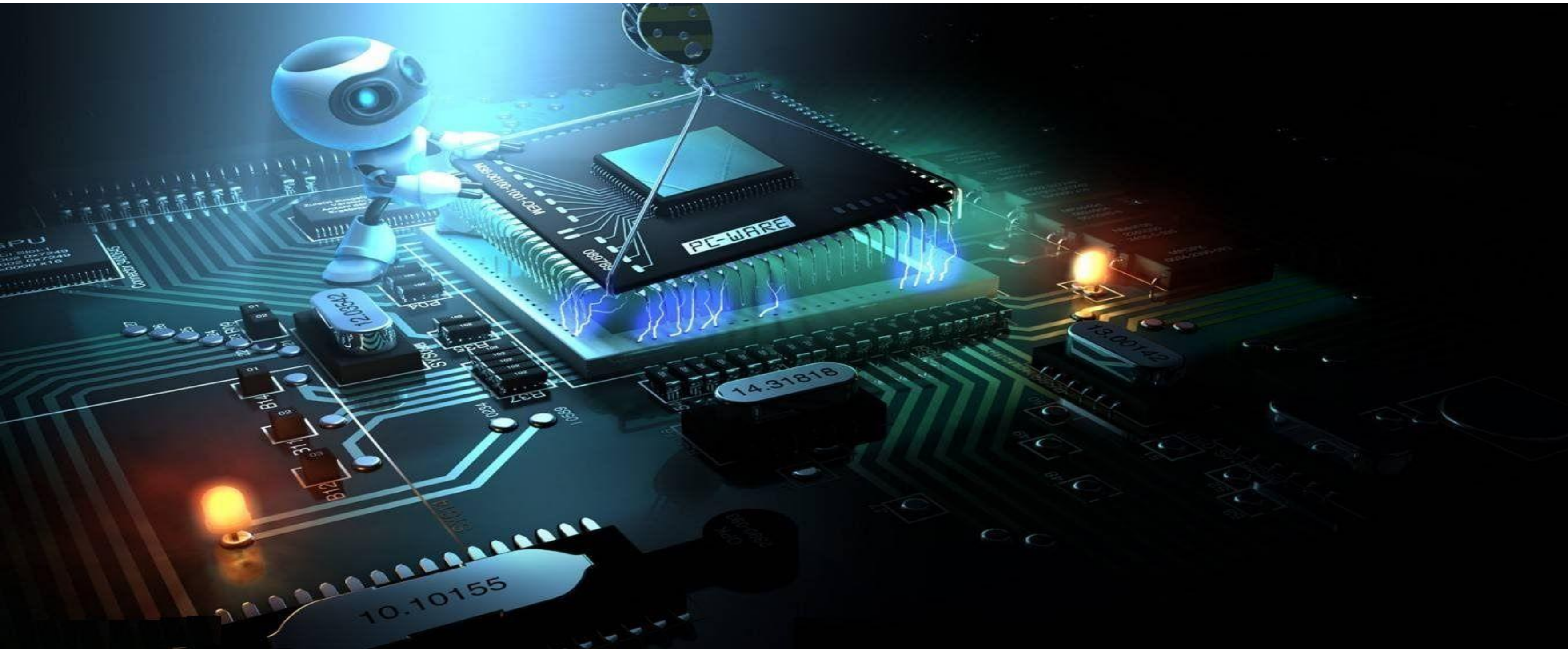
جامعة القاهرة الجديدة
التكنولوجية

Fall 2023

(4)

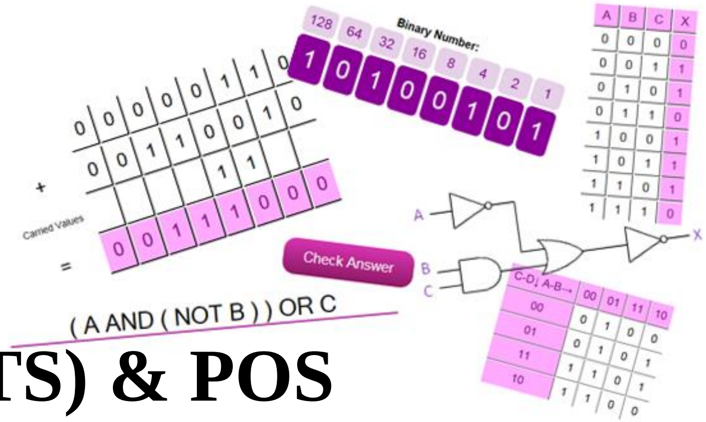


Digital Engineering

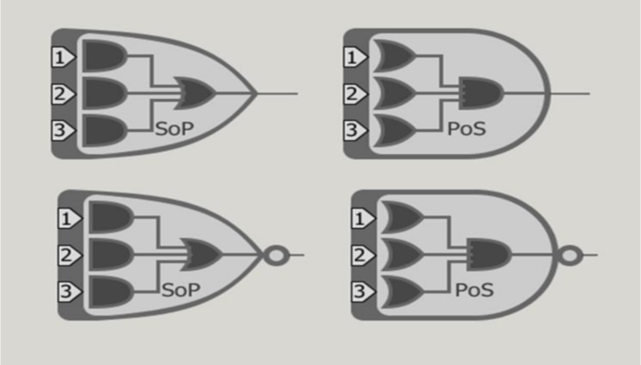
Dr. Hatem Yousry

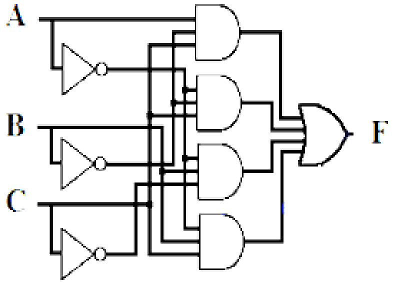
Digital Engineering
Dr. Hatem Yousry

Agenda

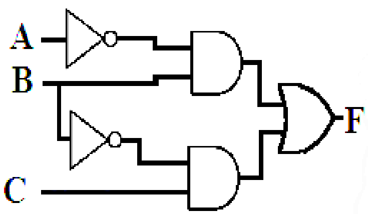


- **SOP (SUM OF PRODUCTS) & POS (PRODUCT OF SUMS).**
- **Truth Table notation for Minterms and Maxterms.**
- **Canonical and Standard Forms.**





Canonical form



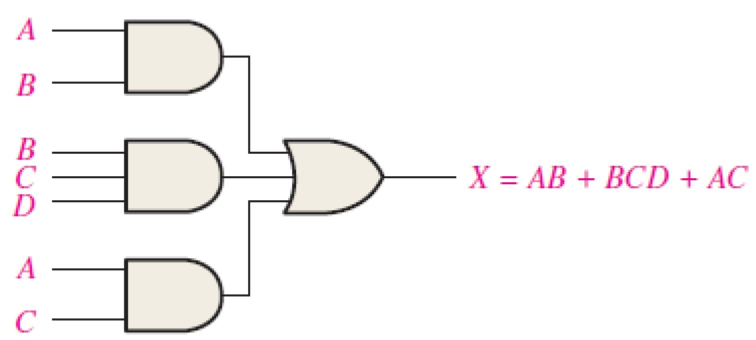
Minimal form

Definitions

- **Literal:** A variable or its complement
- **Product term:** literals connected by •
- **Sum term:** literals connected by +
- **Minterm:** is a Boolean expression resulting in **1 for the output of a single cell, and 0s for all other cells** in a Karnaugh map, or truth table. Minterm for each combination of the variables that produces a **1 in the function** and then taking the **OR** of all those terms.
- **Maxterm:** Maxterm for each combination of the variables that produces a **0 in the function** and then taking the **AND** of all those terms.

AND/OR Implementation of an SOP Expression

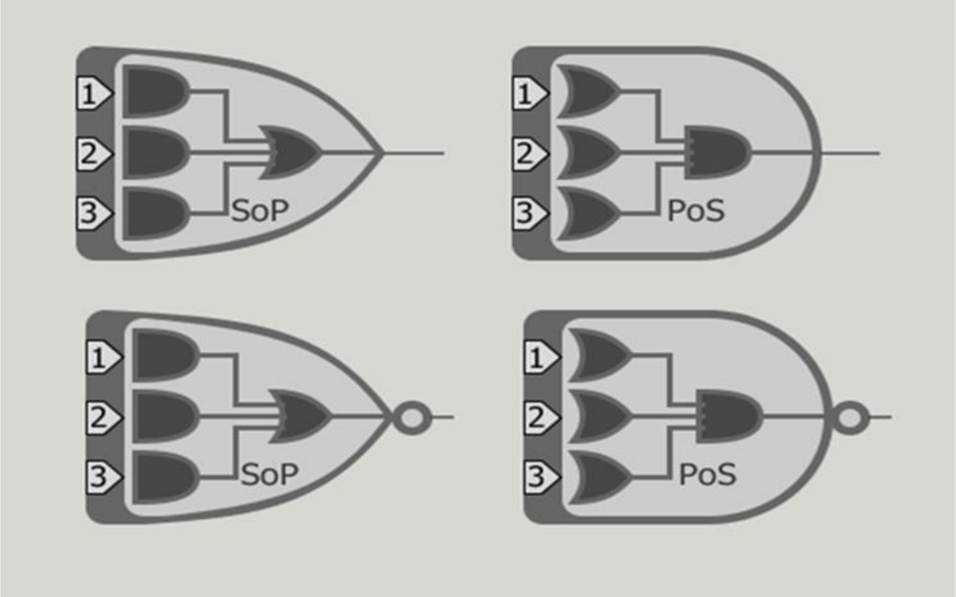
- Implementing an SOP expression simply requires ORing the outputs of two or more AND gates.
- A product term is **produced by an AND operation, and the sum (addition)** of two or more product terms is produced by an **OR operation.**
- Therefore, an SOP expression can be implemented by **AND-OR logic** in which the outputs of a number (equal to the number of product terms in the expression) of AND gates connect to the inputs of an OR gate, for the expression AB + BCD + AC. The output X of the OR gate equals the SOP expression.



The Sum-of-Products (SOP) Form

- When two or more product terms are summed by Boolean addition, the resulting expression is a sum-of-products (SOP).
- Some examples are: ∑ **of** products

$$AB + ABC$$
$$ABC + CDE + \overline{B}C\overline{D}$$
$$\overline{A}B + \overline{A}B\overline{C} + AC$$



Minterm - A Product Term : m_j

- Represents exactly one combination in the truth table.
- Denoted by m_j, where j is the decimal equivalent of the minterm's corresponding binary combination (b_j).
- A variable in m_j is complemented if its value in b_j is 0, otherwise is uncomplemented.
- Example: Assume 3 variables (A,B,C), and $\boldsymbol{j=3}$. Then, $\mathbf{b}_j = \mathbf{011}$ and its corresponding minterm is denoted by $\boldsymbol{m_j} = \mathbf{A'BC}$

Maxterm - A Sum Term : M_j

- Represents exactly one combination in the truth table.
- Denoted by M_j, where j is the decimal equivalent of the maxterm's corresponding binary combination (b_j).
- A variable in M_j is complemented if its value in b_j is 1, otherwise is uncomplemented.
- Example: Assume 3 variables (A,B,C), and $j=3$. Then, $b_j = 011$ and its corresponding maxterm is denoted by $M_j = A+B'+C'$

Truth Table notation for Minterms and Maxterms

- Minterms and Maxterms are easy to denote using a truth table.
- Example: Assume 3 variables x,y,z (order is fixed)

x	y	z		Minterm	Maxterm
0	0	0		$x'y'z' = m_0$	$x+y+z = M_0$
0	0	1		$x'y'z = m_1$	$x+y+z' = M_1$
0	1	0		$x'yz' = m_2$	$x+y'+z = M_2$
0	1	1		$x'yz = m_3$	$x+y'+z' = M_3$
1	0	0		$xy'z' = m_4$	$x'+y+z = M_4$
1	0	1		$xy'z = m_5$	$x'+y+z' = M_5$
1	1	0		$xyz' = m_6$	$x'+y'+z = M_6$
1	1	1		$xyz = m_7$	$x'+y'+z' = M_7$

Canonical Forms (Unique)

- Any Boolean function **F( )** can be expressed as a *unique* **sum** of **min**terms and a unique **product** of **max**terms (under a fixed variable ordering).
- In other words, every function **F()** has two canonical forms:
  - **Canonical Sum-Of-Products**
    - **(sum of minterms)** $\sum$
  - **Canonical Product-Of-Sums** $\prod$
    - **(product of maxterms)**

Canonical Forms

- Canonical Sum-Of-Products:
  The **minterms** included are those m_j such that **F( ) = 1** in row *j* of the truth table for F( ).
- Canonical Product-Of-Sums:
  The **maxterms** included are those M_j such that **F( ) = 0** in row *j* of the truth table for F( ).

Canonical and Standard Forms

Minterms and Maxterms

- **A minterm** (standard product): an **AND** term consists of all literals in their normal form or in their complement form.
  - For example, two binary variables *x* and *y*,
    - *xy, xy', x'y, x'y'*
  - It is also called a **standard product.**
  - *n* variables con be combined to form 2^n minterms.
- **A maxterm** (standard sums): an **OR** term
  - It is also call a **standard sum.**
  - 2^n maxterms.

Minterms and Maxterms

- Each *maxterm* is the complement of its corresponding *minterm*, and vice versa.

Minterms and Maxterms for Three Binary Variables

			Minterms		Maxterms	
x	***y***	***z***	**Term**	**Designation**	**Term**	**Designation**
0	0	0	$x'y'z'$	m_0	$x + y + z$	M_0
0	0	1	$x'y'z$	m_1	$x + y + z'$	M_1
0	1	0	$x'yz'$	m_2	$x + y' + z$	M_2
0	1	1	$x'yz$	m_3	$x + y' + z'$	M_3
1	0	0	$xy'z'$	m_4	$x' + y + z$	M_4
1	0	1	$xy'z$	m_5	$x' + y + z'$	M_5
1	1	0	xyz'	m_6	$x' + y' + z$	M_6
1	1	1	xyz	m_7	$x' + y' + z'$	M_7

Minterms and Maxterms

- The complement of a Boolean function
  - The **minterm**s that produce a **maxterms**
  - $f_1' = m_0 + m_2 + m_3 + m_5 + m_6 = x'y'z' + x'yz' + x'yz + xy'z + xyz'$
  - $f_1 = (f_1')'$
    $= (x+y+z)(x+y'+z)(x+y'+z')(x'+y+z')(x'+y'+z) = M_0 M_2 M_3 M_5 M_6$
  - $f_2 = (x+y+z)(x+y+z')(x+y'+z)(x'+y+z) = M_0 M_1 M_2 M_4$
- Any Boolean function can be expressed as
  - **A sum of minterms ("sum" meaning the ORing of terms).**
  - **A product of maxterms ("product" meaning the ANDing of terms).**
  - **Both boolean functions are said to be in Canonical form.**

Sum of Minterms

- Sum of minterms: there are 2^n minterms and 2^{2^n} combinations of function with *n* Boolean variables.
- Example : express $F = A+BC'$ as a sum of minterms.
  - $F = A+B'C = A\,(B+B') + B'C = AB + AB' + B'C = AB(C+C') + AB'(C+C') + (A+A')B'C = ABC+ABC'+AB'C+AB'C'+A'B'C$
  - $F = A'B'C + AB'C' + AB'C + ABC' + ABC = m_1 + m_4 + m_5 + m_6 + m_7$
  - **$\mathbf{F}(A, B, C) = \Sigma(1, 4, 5, 6, 7)$**
  - or, built the truth table first

Truth Table for $F = A + B'C$

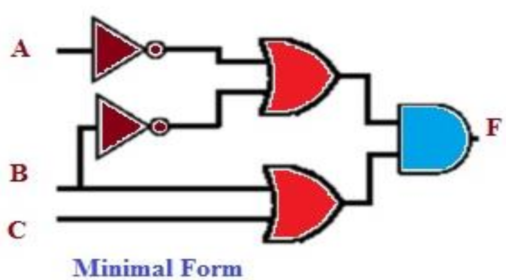
A	B	C	F
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

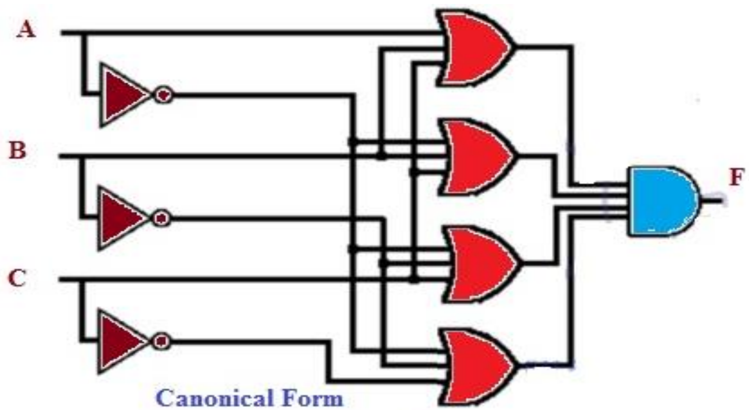
Product of Maxterms

- Product of maxterms: using distributive law to expand.
  - $x + yz = (x + y)(x + z) = (x+y+zz')(x+z+yy') = (x+y+z)(x+y+z')(x+y'+z)$
- Example: express $F = xy + x'z$ as a product of maxterms.
  - $F = xy + x'z = (xy + x')(xy + z) = (x+x')(y+x')(x+z)(y+z) = (x'+y)(x+z)(y+z)$
  - $x'+y = x' + y + zz' = (x'+y+z)(x'+y+z')$
  - $F = (x+y+z)(x+y'+z)(x'+y+z)(x'+y+z') = M_0M_2M_4M_5$
  - $\boldsymbol{F(x, y, z) = \Pi(0, 2, 4, 5)}$

Canonical and Standard Forms

- We need to consider formal techniques for the simplification of Boolean functions.
- Identical functions will have exactly the same canonical form.
- **Minterms and Maxterms**
- Sum-of-Minterms and Product-of- Maxterms
- Product and Sum terms
- **Sum-of-Products (SOP) and Product-of-Sums (POS).**





Digital Engineering
Dr. Hatem Yousry

Example

- Truth table for **f_1(a,b,c)** at right
- The canonical sum-of-products form for f_1 is
  $f_1(a,b,c) = m_1 + m_2 + m_4 + m_6$
  $= a'b'c + a'bc' + ab'c' + abc'$
- The canonical product-of-sums form for f_1 is
  $f_1(a,b,c) = M_0 \cdot M_3 \cdot M_5 \cdot M_7$
  $= (a+b+c)\cdot(a+b'+c')\cdot(a'+b+c')\cdot(a'+b'+c')$.
- **Observe that: $m_j = M_j'$**

a	b	c		f_1
0	0	0		0
0	0	1		1
0	1	0		1
0	1	1		0
1	0	0		1
1	0	1		0
1	1	0		1
1	1	1		0

Shorthand: ∑ and ∏

- $f_1(a,b,c) = \sum$ **m(1,2,4,6),** where **∑ indicates that this is a sum-of-products form**, and m(1,2,4,6) indicates that the minterms to be included are m_1, m_2, m_4, and m_6.
- $f_1(a,b,c) = \prod$ **M(0,3,5,7),** where **∏ indicates that this is a product-of-sums form**, and M(0,3,5,7) indicates that the maxterms to be included are M_0, M_3, M_5, and M_7.
- **Since $m_j = M_j'$ for any *j*,**
  $$\sum m(1,2,4,6) = \prod M(0,3,5,7) = f_1(a,b,c)$$

Digital Engineering
Dr. Hatem Yousry

Conversion of a General Expression to SOP Form

- Any logic expression can be changed into SOP form by applying Boolean algebra techniques.
- For example, the expression **A(B + CD)** can be converted to SOP form by applying the **distributive law:**

$$A(B + CD) = AB + ACD$$

Convert each of the following Boolean expressions to SOP form:

(a) $AB + B(CD + EF)$ **(b)** $(A + B)(B + C + D)$ **(c)** $\overline{(\overline{A + B}) + C}$

Solution

(a) $AB + B(CD + EF) = AB + BCD + BEF$

(b) $(A + B)(B + C + D) = AB + AC + AD + BB + BC + BD$

(c) $\overline{(\overline{A + B}) + C} = (\overline{\overline{A + B}})\overline{C} = (A + B)\overline{C} = A\overline{C} + B\overline{C}$

Conversion Between Canonical Forms

- Replace **$\sum$ with $\prod$** (or *vice versa*) and replace those *j'*s that appeared in the original form with those that do not.
- Example:

$$\mathbf{f_1(a,b,c) = a'b'c + a'bc' + ab'c' + abc'}$$
$$\mathbf{= m_1 + m_2 + m_4 + m_6}$$
$$\mathbf{= \sum(1,2,4,6)}$$
$$\mathbf{= \prod(0,3,5,7)}$$
$$\mathbf{= (a+b+c)\bullet(a+b'+c')\bullet(a'+b+c')\bullet(a'+b'+c')}$$

Standard Forms (NOT Unique)

- Standard forms are "*like*" canonical forms, except that **not all variables need appear** in the individual product (SOP) or sum (POS) terms.
- **Example:**
  $f_1(a,b,c) = a'b'c + bc' + ac'$
  is a *standard* sum-of-products form
- $f_1(a,b,c) = (a+b+c)\bullet(b'+c')\bullet(a'+c')$
  is a *standard* product-of-sums form.

Conversion of SOP from standard to canonical form

- Expand ***non-canonical*** terms by **inserting equivalent of 1** in each missing variable x:
  $(x + x') = 1$
- Remove duplicate minterms
- **$f_1(a,b,c) = a'b'c + bc' + ac'$**

$$= a'b'c + (a+a')bc' + a(b+b')c'$$

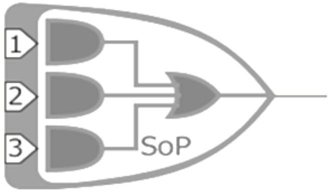
$$= a'b'c + abc' + a'bc' + abc' + ab'c'$$

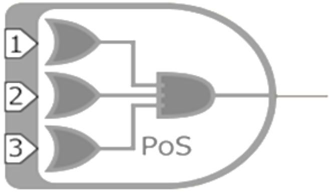
$$= a'b'c + abc' + a'bc + ab'c'$$

Conversion of POS from standard to canonical form

- Expand noncanonical terms by **adding 0** in terms of missing variables (*e.g.*, **xx' = 0**) and using the **distributive law**
- Remove duplicate maxterms
- $\mathbf{f_1(a,b,c) = (a+b+c)\bullet(b'+c')\bullet(a'+c')}$
  $\mathbf{= (a+b+c)\bullet(aa'+b'+c')\bullet(a'+bb'+c')}$
- $\mathbf{= (a+b+c)\bullet(a+b'+c')\bullet(a'+b'+c')\bullet(a'+b+c')\bullet(a'+b'+c')}$
  $\mathbf{= (a+b+c)\bullet(a+b'+c')\bullet(a'+b'+c')\bullet(a'+b+c')}$

SOP $\sum$ and POS $\prod$





- SOP for **Minterm (m) in the form of**
- **And + And + And + ……**
- $\sum$ *(m1, m2 , …….m_j) = sum of minterms*
- **As** the **minterms** included are those m_j such that **F( ) = 1** in row *j* of the truth table for F( ).
- POS for **Maxterm (M) in the form of**
- **OR . OR . OR . ……….**
- $\prod$ ***(M1, M2, ……….M_j) = product of maxterms***
- **As** the **maxterms** included are those M_j such that **F( ) = 0** in row *j* of the truth table for F( ).
- **Observe that:** $m_j = M_j'$

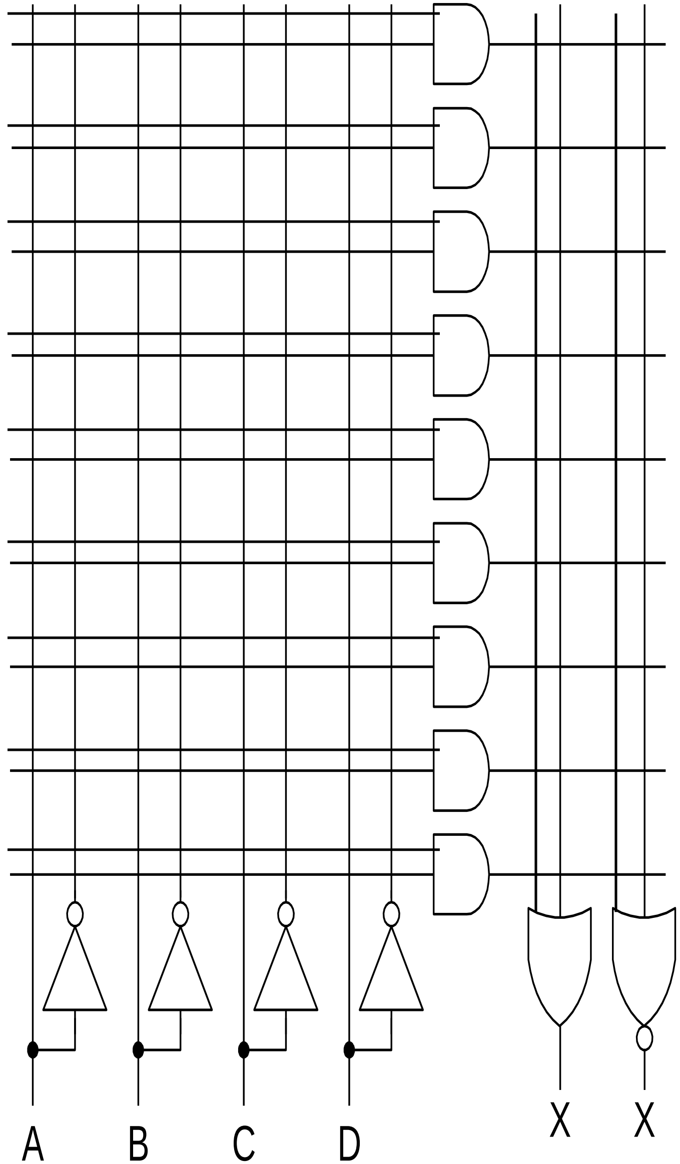
Name:

	Inputs				Output		
0	0	0	0	0			
1	0	0	0	1			
2	0	0	1	0			
3	0	0	1	1			
4	0	1	0	0			
5	0	1	0	1			
6	0	1	1	0			
7	0	1	1	1			
8	1	0	0	0			
9	1	0	0	1			
10	1	0	1	0			
11	1	0	1	1			
12	1	1	0	0			
13	1	1	0	1			
14	1	1	1	0			
15	1	1	1	1			

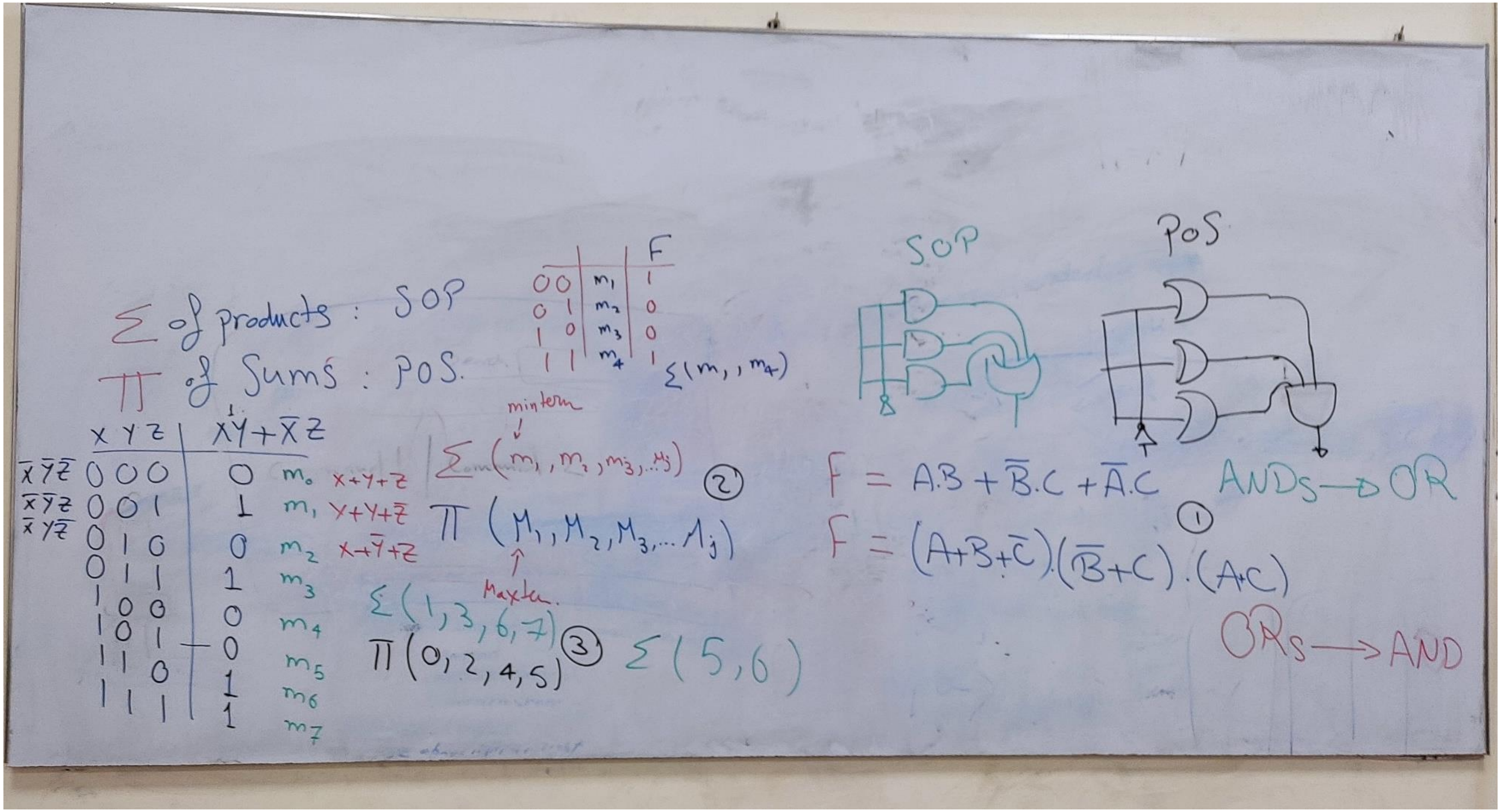
X =

K- Map

X =

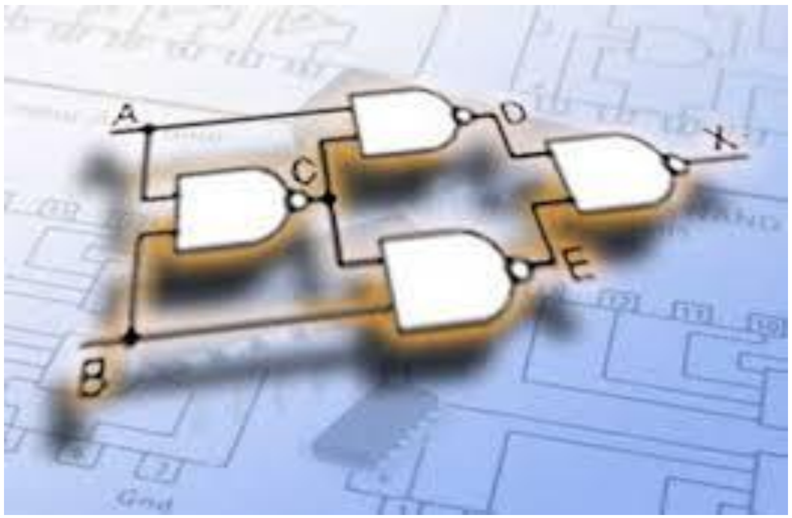


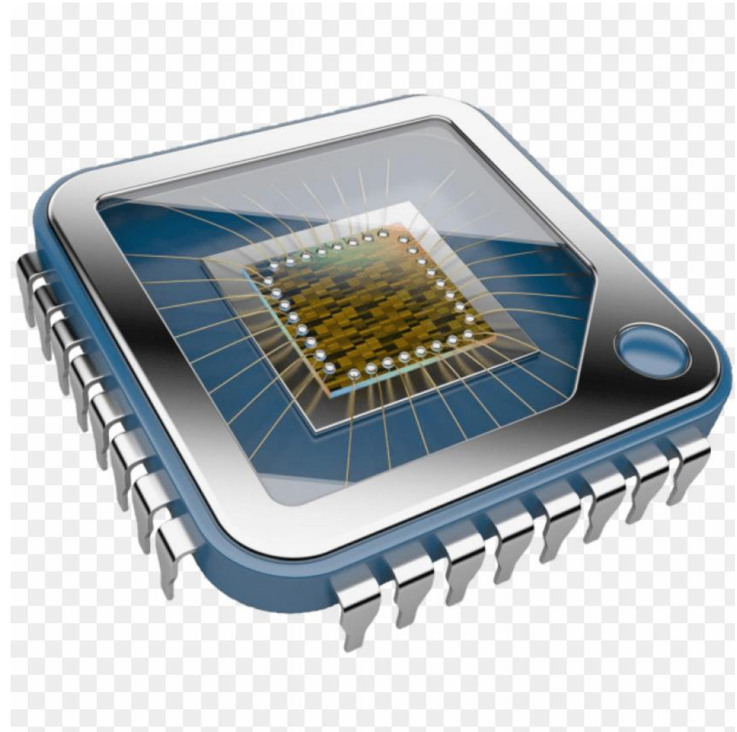
SOP ∑ and POS ∏



Σ of products : SOP
Π of Sums : POS
F
minterm
Σ(m₁, m₄)
X Y Z | XY + X̄Z
Σm(m₁, m₂, m₃, ... mⱼ)
Π(M₁, M₂, M₃, ... Mⱼ)
Maxterm
Σ(1, 3, 6, 7)
Π(0, 2, 4, 5)
Σ(5, 6)
SOP
POS
F = A.B + B̄.C + Ā.C
F = (A+B+C̄).(B̄+C).(A+C)
ANDs → OR
ORs → AND

Thank You





Dr. Hatem Yousry
E-mail: Hyousry@nctu.edu.eg