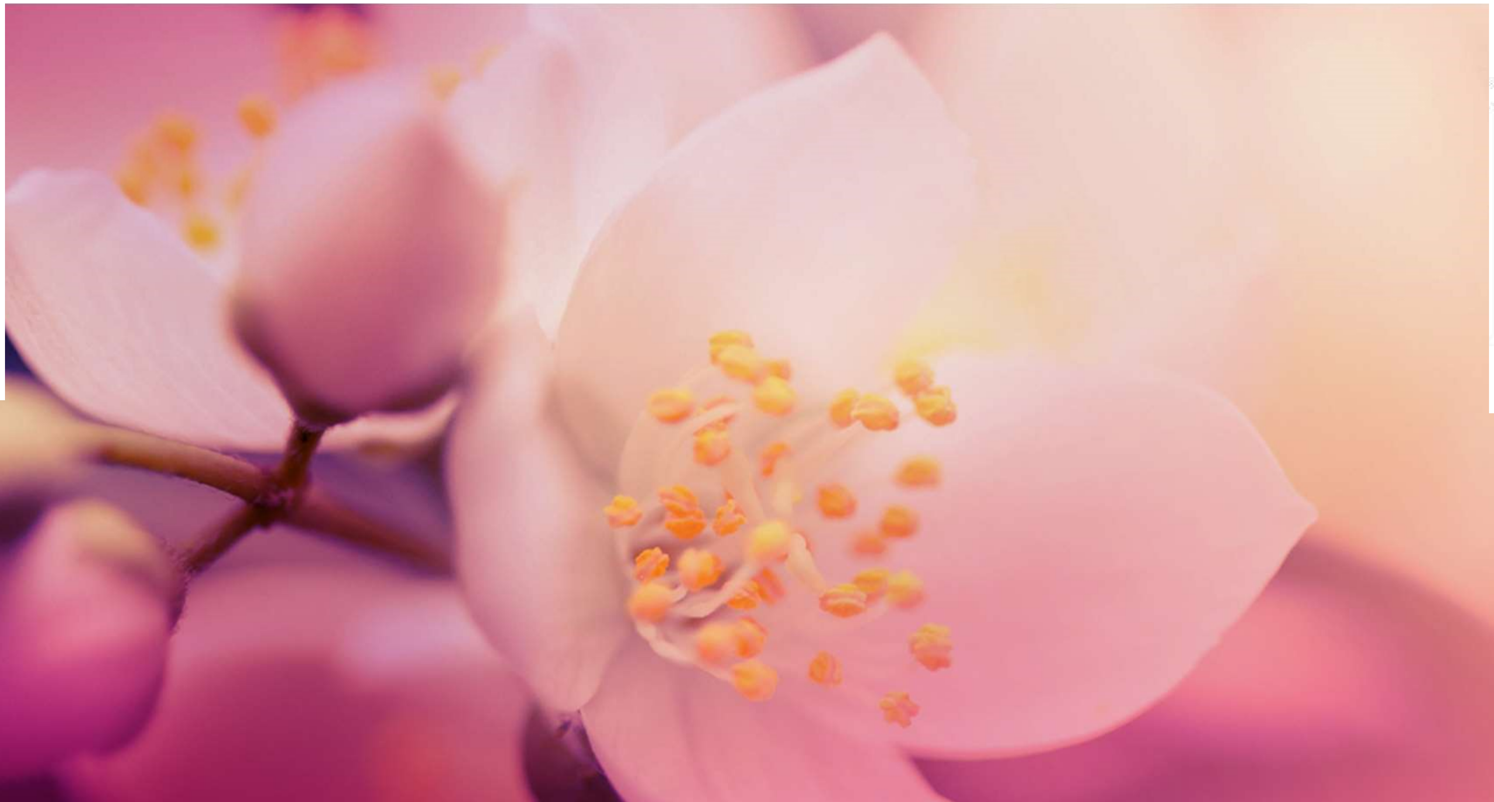


Modern Digital System Design

ECE 2372 / Fall 2018 / Lecture 03

Texas Tech University
Dr. Tooraj Nikoubin

Applications of Boolean Algebra, Minterm and Maxterm Expansions

Boolean Functions : Terminology

F(a,b,c) = a'bc + abc' + ab + c

- **Variable**

– Represents a value (0 or 1), Three variables: **a**, **b**, and **c**

- **Literal**

– Appearance of a variable, in true or complemented form

– Nine literals: **a'**, **b, c, a, b, c', a, b,** and **c**

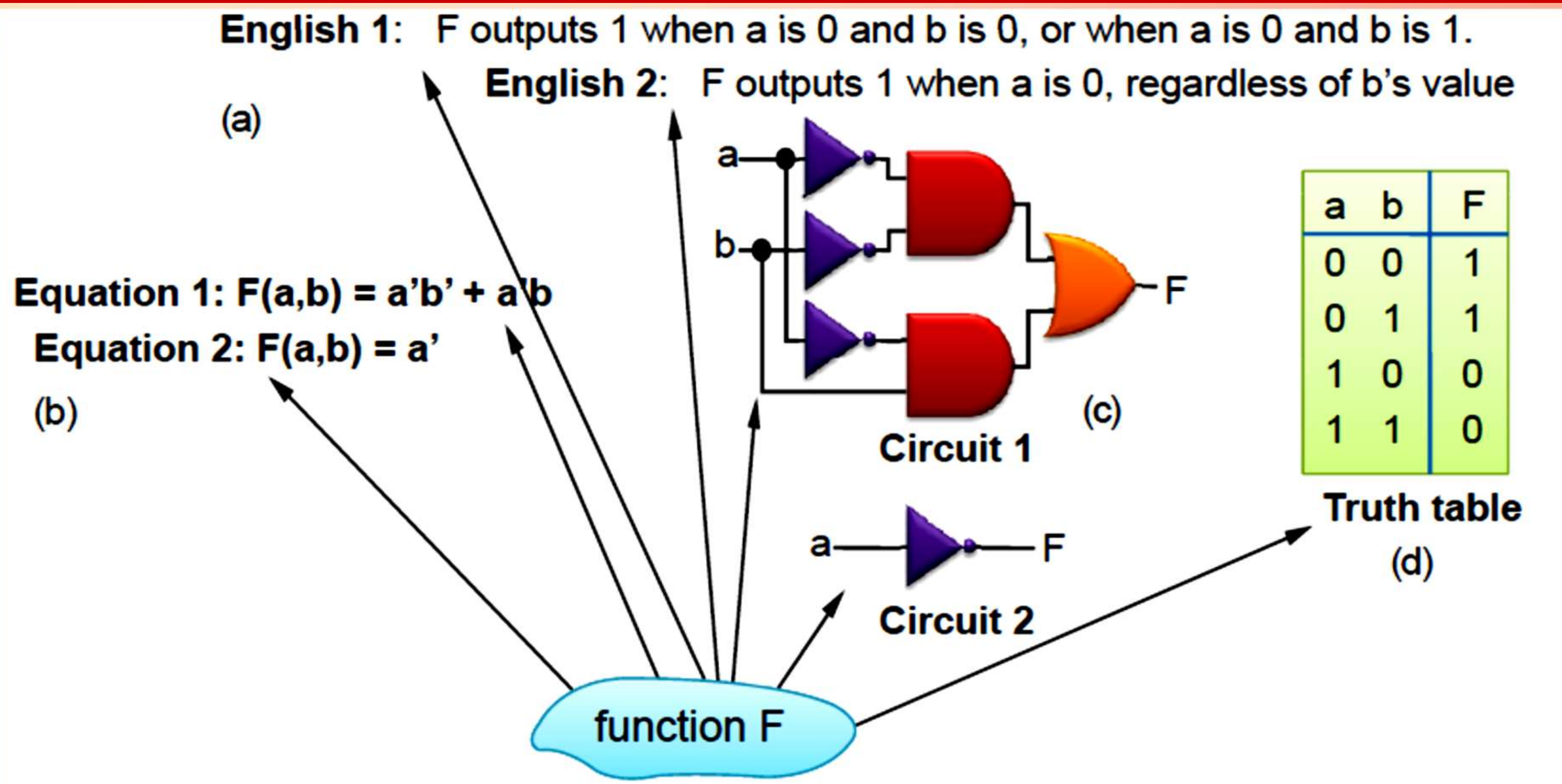
- **Product term**

– Product of literals, Four product terms: **a'bc, abc', ab, c**

- **Sum-of-products (SOP)**

– Above equation is in sum-of-products form.

Representations of Boolean Functions



- A function can be represented in different ways
  - Above shows seven representations of the same functions F(a,b), using four different methods: English, Equation, Circuit, and Truth Table

Truth Table Representation of Functions

- Define value of F for each possible combination of input values
  - 2-input function: 4 rows
  - 3-input function: 8 rows
  - 4-input function: 16 rows
- Q: Use truth table to define function F(a,b,c) that is 1 when abc is 5 or greater in binary

a	b	F
0	0	
0	1	
1	0	
1	1	

(a)

a	b	c	F
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

(b)

a	b	c	d	F
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

(c)

a	b	c	F
0	0	0	**0**
0	0	1	**0**
0	1	0	**0**
0	1	1	**0**
1	0	0	**0**
1	0	1	**1**
1	1	0	**1**
1	1	1	**1**

Converting among Representations

- Can convert from any representation to any other
- Common conversions
  - Equation to circuit
  - Truth table to equation
  - Equation to truth table
    - Easy -- just evaluate equation for each input combination (row)
    - Creating intermediate columns helps

Inputs		Outputs	Term
a	b	F	F = sum of
0	0	1	a'b'
0	1	1	a'b
1	0	0	
1	1	0	

F = a'b' + a'b

Q: Convert to truth table: F = a'b' + a'b

Inputs				Output
a	b	a' b'	a' b	F
0	0	1	0	**1**
0	1	0	1	**1**
1	0	0	0	**0**
1	1	0	0	**0**

Q: Convert to equation

a	b	c	F	
0	0	0	0	
0	0	1	0	
0	1	0	0	
0	1	1	0	
1	0	0	0	
1	0	1	1	ab'c
1	1	0	1	abc'
1	1	1	1	abc

F = ab'c + abc' + abc

Standard Representation: Truth Table

- How to determine two functions are the same?
  - Use algebraic methods
  - But if we failed, does that prove *not* equal? No.
- Solution: Convert to truth tables
  - Only ONE truth table representation of given same functions: Standard representation

F = ab + a'

a	b	F
0	0	1
0	1	1
1	0	0
1	1	1

Same

F = a'b' +a'b + ab

a	b	F
0	0	1
0	1	1
1	0	0
1	1	1

Canonical Form -- Sum of Minterms

- Truth tables too big for numerous inputs
- Use standard form of equation instead

Boolean algebra: create sum of minterms

- ***Minterm***: product term with every function literal appearing exactly once, in true or complemented form
- Just multiply-out equation until sum of product terms
- Then expand each term until all terms are minterms

Determine if $F(a,b) = ab+a'$ is same function as $F(a,b) = a'b'+a'b+ab$ by to canonical form.

$F = ab+ a'$ (already sum of products)
$F = ab + a'(b+b')$ (expanding term)
$F = ab + a'b + a'b'$ (it is canonical form)

Canonical form or Standard Form

- Canonical forms

–Sum of minterms (SOM)

–Product of maxterms (POM)

- Standard forms (may use less gates)

–Sum of products (SOP)

–Product of sums (POS)

F = ab+a' (already sum of products:SOP)

F = ab + a'(b+b') (expanding term)

F = ab + a'b + a'b' (it is canonical form:SOM)

Canonical Forms

- **It is useful to specify Boolean functions in a form that:**
  - **Allows comparison for equality.**
  - **Has a correspondence to the truth tables**

- **Canonical Forms in common usage:**
  - **Sum of Minterms (SOM)**
  - **Product of Maxterms (POM)**

Minterms

- **product term** is a term where literals are ANDed.
- Example: x'y', xz, xyz, …
- **minterm** : A product term in which all variables appear exactly once, in normal or complemented form
- Example: F(x,y,z) has 8 minterms
x'y'z', x'y'z, x'yz', …
- Function with n variables has $\mathbf{2^n}$ minterms
- A minterm equals 1 at exactly one input combination and is equal to 0 otherwise
- Example: x'y'z' = 1 only when x=0, y=0, z=0
- A minterm is denoted as mi where i corresponds the input combination at which this minterm is equal to 1

2 Variable Minterms

- **Two variables (X and Y) produce 2x2=4 combinations**

XY (both normal)
XY' (X normal, Y complemented)
X'Y (X complemented, Y normal)
X'Y' (both complemented)

Maxterms

- **Maxterms are OR terms with every variable in true or complemented form.**

X+Y (both normal)
X+Y' (x normal, y complemented)
X'+Y (x complemented, y normal)
X'+Y' (both complemented)

Maxterms and Minterms

Two variable minterms and maxterms.

Index	Minterm	Maxterm
0	$x' y'$	$x + y$
1	$x 'y$	$x + y'$
2	$x y'$	$x' + y$
3	$x y$	$x' + y'$

The index above is important for describing which variables in the terms are true and which are complemented.

Minterms

Minterms for Three Variables

X	Y	Z	Product Term	Symbol	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7
0	0	0	$\overline{X}\,\overline{Y}\,\overline{Z}$	m_0	1	0	0	0	0	0	0	0
0	0	1	$\overline{X}\,\overline{Y}Z$	m_1	0	1	0	0	0	0	0	0
0	1	0	$\overline{X}Y\overline{Z}$	m_2	0	0	1	0	0	0	0	0
0	1	1	$\overline{X}YZ$	m_3	0	0	0	1	0	0	0	0
1	0	0	$X\overline{Y}\,\overline{Z}$	m_4	0	0	0	0	1	0	0	0
1	0	1	$X\overline{Y}Z$	m_5	0	0	0	0	0	1	0	0
1	1	0	$XY\overline{Z}$	m_6	0	0	0	0	0	0	1	0
1	1	1	XYZ	m_7	0	0	0	0	0	0	0	1

Variable complemented if 0
Variable uncomplemented if 1

m_i indicated the i[th] minterm
i indicates the binary combination
m_i is equal to 1 for ONLY THAT combination

Maxterms

- **Sum term** : A term where literals are ORed.
- Example: x'+y', x+z, x+y+z, …
- **Maxterm** : a sum term in which all variables appear exactly once, in normal or complemented form
- Example: F(x,y,z) has 8 maxterms (x+y+z), (x+y+z'), (x+y'+z), ...
- Function with n variables has $\mathbf{2^n}$ maxterms
- A maxterm equals 0 at exactly one input combination and is equal to 1 otherwize
- Example: (x+y+z) = 0 only when x=0, y=0, z=0
- A maxterm is denoted as Mi where i corresponds the input combination at which this maxterm is equal to 0

Maxterms

Maxterms for Three Variables

X	Y	Z	Sum Term	Symbol	M_0	M_1	M_2	M_3	M_4	M_5	M_6	M_7
0	0	0	$X+Y+Z$	M_0	0	1	1	1	1	1	1	1
0	0	1	$X+Y+\overline{Z}$	M_1	1	0	1	1	1	1	1	1
0	1	0	$X+\overline{Y}+Z$	M_2	1	1	0	1	1	1	1	1
0	1	1	$X+\overline{Y}+\overline{Z}$	M_3	1	1	1	0	1	1	1	1
1	0	0	$\overline{X}+Y+Z$	M_4	1	1	1	1	0	1	1	1
1	0	1	$\overline{X}+Y+\overline{Z}$	M_5	1	1	1	1	1	0	1	1
1	1	0	$\overline{X}+\overline{Y}+Z$	M_6	1	1	1	1	1	1	0	1
1	1	1	$\overline{X}+\overline{Y}+\overline{Z}$	M_7	1	1	1	1	1	1	1	0

Variable complemented if 1
Variable not complemented if 0

M_i indicated the i^{th} maxterm i indicates the binary combination M_i is equal to 0 for ONLY THAT combination

Expressing Functions with Minterms

- **Boolean function can be expressed algebraically from a give truth table**
- **Forming sum of ALL the minterms that produce 1 in the function**

Example : Consider the function truth table defined by the

F(X,Y,Z)= X’Y’Z’ + X’YZ’ + XY’Z + XYZ =

Σ (0 2 5 7) = m0 + m2 + m5 + m7 =

X	Y	Z	m	F
0	**0**	**0**	m_0	**1**
0	0	1	m_1	0
0	**1**	**0**	m_2	**1**
0	1	1	m_3	0
1	0	0	m_4	0
1	**0**	**1**	m_5	**1**
1	1	0	m_6	0
1	**1**	**1**	m_7	**1**

Expressing Functions with Maxterms

- **Boolean function : Expressed algebraically from a give truth table**
- **By forming logical product (AND) of ALL the maxterms that produce 0 in the function**

Example:
Consider the function defined by the truth table

F(X,Y,Z) = Π M(1,3,4,6)
Applying DeMorgan

F' = $m_1 + m_3 + m_4 + m_6$ = Σm(1 3 4 6)
F = F'' = $[m_1 + m_3 + m_4 + m_6]$'
= m1'.m3'.m4'.m6'
Note the indices in this list are those that
= $M_1 \,.\, M_3 \,.\, M_4 \,.\, M_6$ = Π M(1,3,4,6)

Note the indices in this list are those that are missing from the previous list in Σm(0,2,5,7)

X	Y	Z	M	F	F'
0	0	0	M_0	1	0
0	**0**	**1**	$\mathbf{M_1}$	**0**	**1**
0	1	0	M_2	1	0
0	**1**	**1**	$\mathbf{M_3}$	**0**	**1**
1	**0**	**0**	$\mathbf{M_4}$	**0**	**1**
1	0	1	M_5	1	0
1	**1**	**0**	$\mathbf{M_6}$	**0**	**1**
1	1	1	M_7	1	0

Sum of Minterms vs Product of Maxterms

- A function can be expressed algebraically as:
  - The sum of minterms
  - The product of maxterms
- Given the truth table, writing F as
  - Σmi – for all minterms that produce 1 in the table, or
  - ΠMi – for all maxterms that produce 0 in the table
- Minterms and Maxterms are complement of each other.

Example: minterm & maxterm

- **Write E = Y' + X'Z' in the form of Σmi and ΠMi?**
- **Method1** construct the Truth Table as shown

E= $\Sigma m(0,1,2,4,5)$, and $E = \Pi M(3,6,7)$

X	Y	Z	m	M	E
0	0	0	m_0	M_0	**1**
0	0	1	m_1	M_1	**1**
0	1	0	m_2	M_2	**1**
0	1	1	m_3	M_3	0
1	0	0	m_4	M_4	**1**
1	0	1	m_5	M_5	**1**
1	1	0	m_6	M_6	0
1	1	1	m_7	M_7	0

Example (Cont.)

Solution: **Method2_a**	Solution: **Method2_b**
$E = Y' + X'Z'$	$E = Y' + X'Z'$
$= Y'(X+X')(Z+Z') + X'Z'(Y+Y')$	$E' = Y(X+Z)$
$= (XY'+X'Y')(Z+Z') + X'YZ' + X'Z'Y'$	$= YX + YZ$
$= Y'Z+X'Y'Z+XY'Z'+X'Y'Z'+X'YZ'+X'Z'Y'$	$= YX(Z+Z') + YZ(X+X')$
$= m_5 + m_1 + m_4 + m_0 + m_2 + m_0$	$= XYZ+XYZ'+X'YZ$
$= m_0 + m_1 + m_2 + m_4 + m_5$	$E = (X'+Y'+Z')(X'+Y'+Z)(X+Y'+Z')$
$= \sum m(0,1,2,4,5)$	$= M_7 . M_6 . M_3$
	$= \Pi M(3,6,7)$
To find the form ΠMi, consider the remaining indices	To find the form Sm_i, consider the remaining indices
$E = \Pi M(3,6,7)$	$E = \sum m(0,1,2,4,5)$

Canonical Forms

- **The sum of minterms and the product of maxterms forms are known as the canonical forms of a function.**

Standard Forms

- Sum of Products (SOP) and Product of Sums (POS) are also standard forms

- $AB+CD = (A+C)(B+C)(A+D)(B+D)$

- The sum of minterms is a special case of the SOP form, where all product terms are minterms

- The product of maxterms is a special case of the POS form, where all sum terms are maxterms

SOP and POS Conversion

SOP → POS

$F = AB + CD$

$= (AB+C)(AB+D)$

$= (A+C)(B+C)(AB+D)$

$= (A+C)(B+C)(A+D)(B+D)$

POS → SOP

$F = (A'+B)(A'+C)(C+D)$

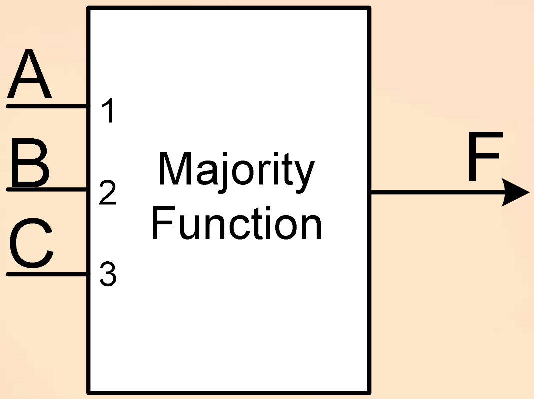
$= (A'+BC)(C+D)$

$= A'C+A'D+BCC+BCD$

$= A'C+A'D+BC+BCD$

$= A'C+A'D+BC$

Example : Three input Majority Function



- True Table
- Minterms
- Maxterms
- Canonical : Sum of Minterms (SOM)
- Standard: Sum of Producs (SOP)
- Canonical : Product of Maxterms (POM)
- Standard: Product of sum (POS)
- Design with Only NAND Gate
- Design with Only NOR Gate

• Example:

- ∗ Majority function
  - » Output is one whenever majority of inputs is 1
  - » We use 3-input majority function

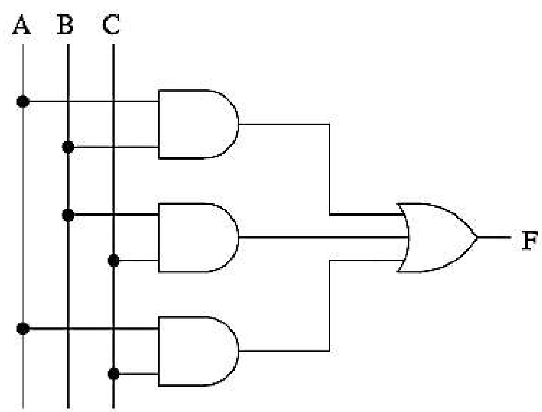
Logic Functions

3-input majority function

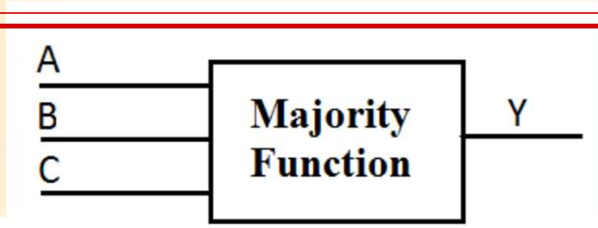
A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

• Logical expression form

$$F = A\,B + B\,C + A\,C$$



Majority Function



	[illegible]	2	1	
	2^2	2^1	2^0	
	A	B	C	
m_0	0	0	0	0
m_1	0	0	1	0
m_2	0	1	0	0
m_3	0	1	1	1
m_4	1	0	0	0
m_5	1	0	1	1
m_6	1	1	0	1
m_7	1	1	1	1

Maxterms

Minterms

- 3-input majority function

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

- SOM logical expression
- Four product terms
  - Because there are 4 rows with a 1 output

$$F = \overline{A}\,B\,C + A\,\overline{B}\,C + A\,B\,\overline{C} + A\,B\,C$$

- Sigma notation

$$\Sigma(3, 5, 6, 7)$$

- 3-input majority function

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

- POM logical expression
- Four sum terms
  - * Because there are 4 rows with a 0 output

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

- Pi notation

$$\Pi\,(0, 1, 2, 4)$$

Majority Function

$Y = m_3 + m_5 + m_6 + m_7 =$

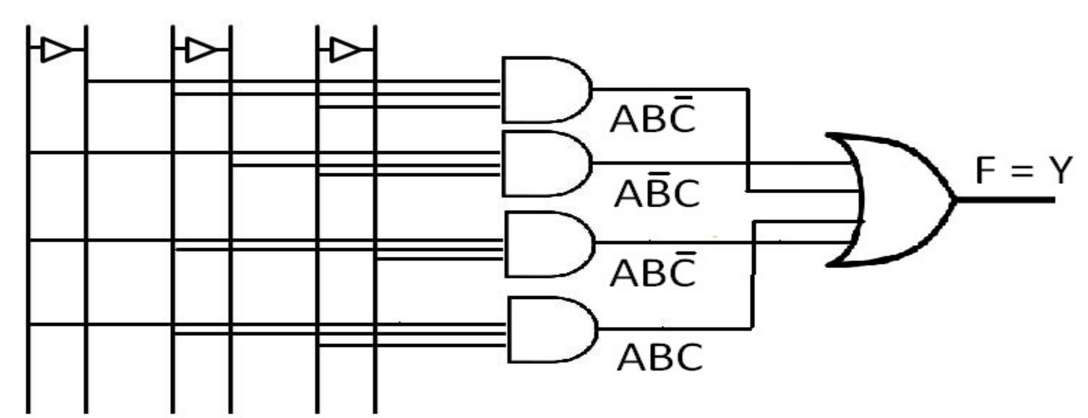
$Y = \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC$

	A	B	C	Y
m_0	0	0	0	0
m_1	0	0	1	0
m_2	0	1	0	0
m_3	0	1	1	1
m_4	1	0	0	0
m_5	1	0	1	1
m_6	1	1	0	1
m_7	1	1	1	1

(Some Of Minterms) SOM

$4AND_3$, $1OR_4$, 3NOT

Majority Function



$$Y = \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC$$

$$Y = C(A \oplus B) + AB(\overset{1}{\overleftrightarrow{C + C'}})$$

$$Y = C(A \oplus B) + AB$$

$Y = BC + AC + AB$ SOP(min)

$3AND_2$ $1OR_3$

Majority Function

$$M_0 = \overline{m_0} = \overline{\bar{A}\bar{B}\bar{C}} = A + B + C$$

$$M4 = \overline{A\bar{B}\bar{C}} = \overline{m4}$$

$$Y = M_0 \,.\, M_1 \,.\, M_2 \,.\, M_4$$

$$Y = (A + B + C)(A + B + \bar{C})(A + \bar{B} + C)(\bar{A} + B + C)$$

$4OR_3$, $1AND_4$, $3NOT$

$$Y = [A + B + \underbrace{(C\bar{C})}_{0}]\ [A + C + \underbrace{(B\bar{B})}_{0}]\ [B + C + \underbrace{(A\bar{A})}_{0}]$$

$$Y = (A + B)\ (A + C)\ (B + C) \qquad (POS)_{min}$$

$$Y = [A + (BC)]\ [B + C]$$

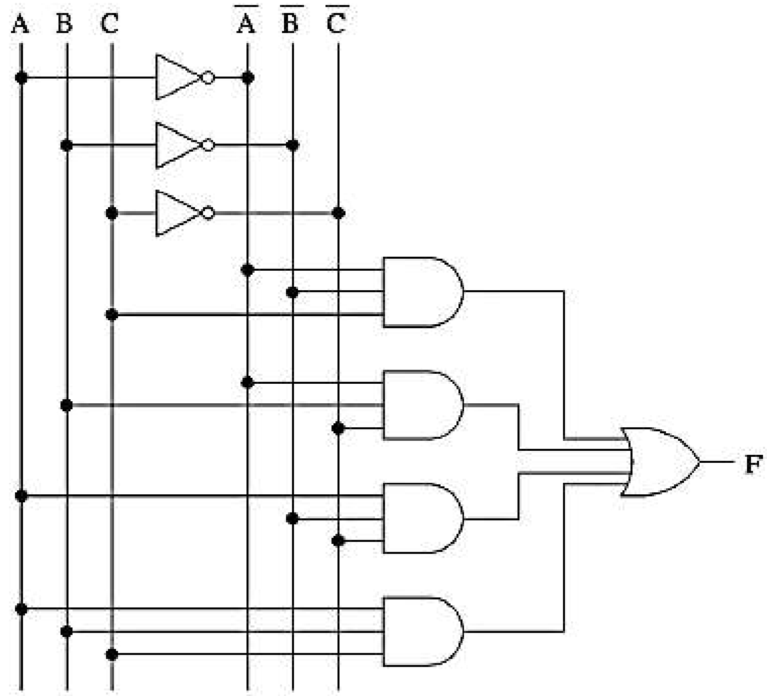
$$Y = AB + AC + BC \qquad (SOP)_{min}$$

$$A + \bar{A}B = \underbrace{(A + \bar{A})}_{1}(A + B) = A + B$$

3-input even-parity function

A	B	C	F
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	0
1	1	1	1

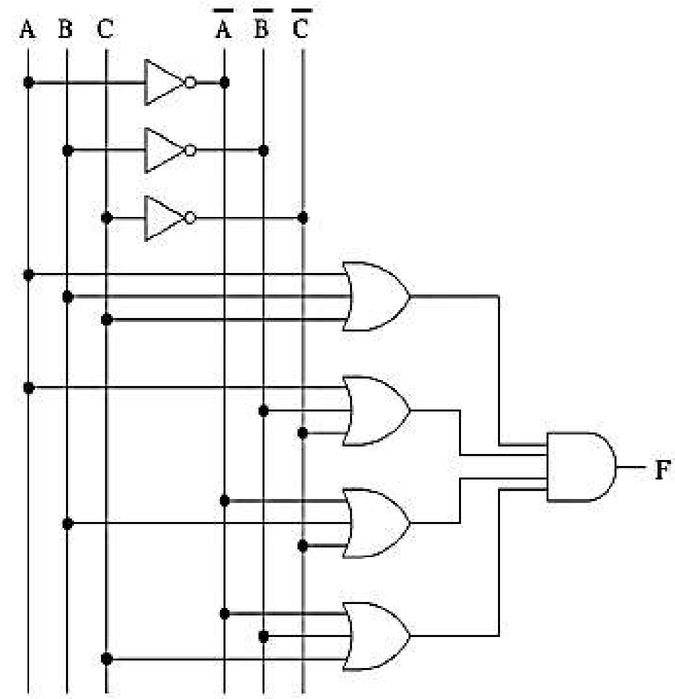
- SOM implementation



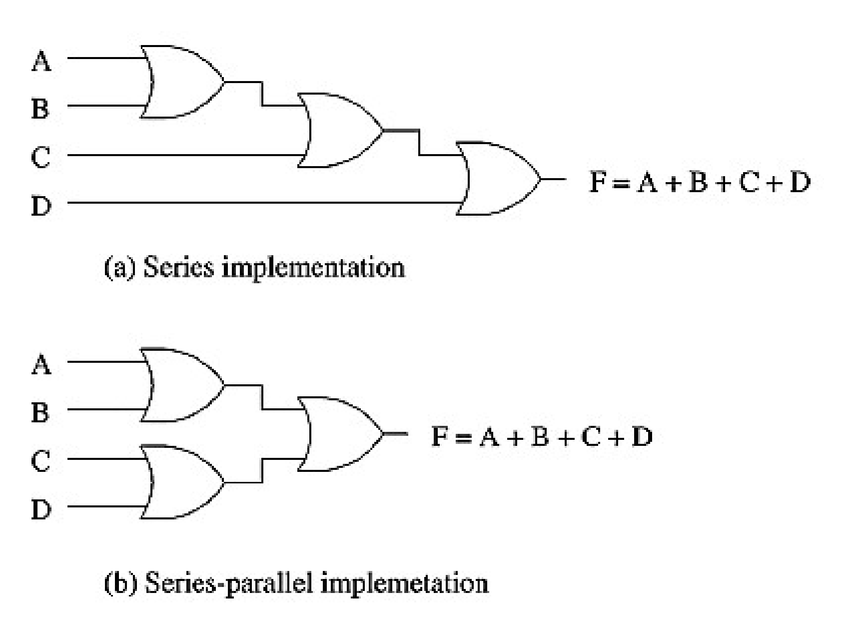
3-input even-parity function

A	B	C	F
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	0
1	1	1	1

- POM implementation



- Various ways to build higher-input gates
  - Series
  - Series-parallel
- Propagation delay depends on the implementation
- Series implementation
  - » 3-gate delay
- Series-parallel implementation
  - » 2-gate delay



Implementation Using Other Gates

- Using NAND gates
  - * Get an equivalent expression

$$A\,B + C\,D = \overline{\overline{A\,B + C\,D}}$$

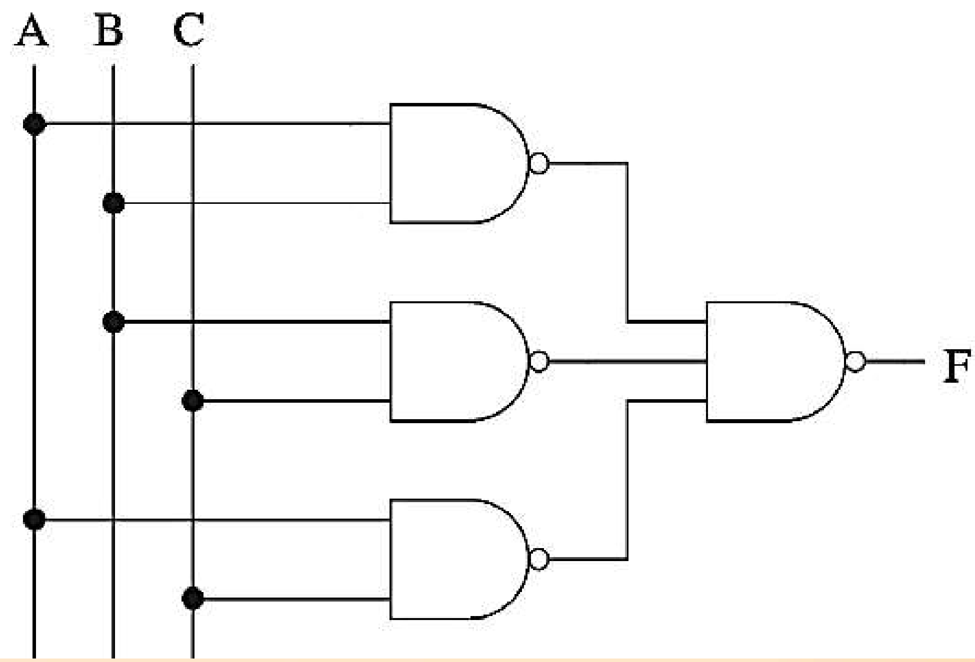
  - * Using de Morgan's law

$$A\,B + C\,D = \overline{\overline{A\,B} \cdot \overline{C\,D}}$$

  - * Can be generalized
    - » Majority function

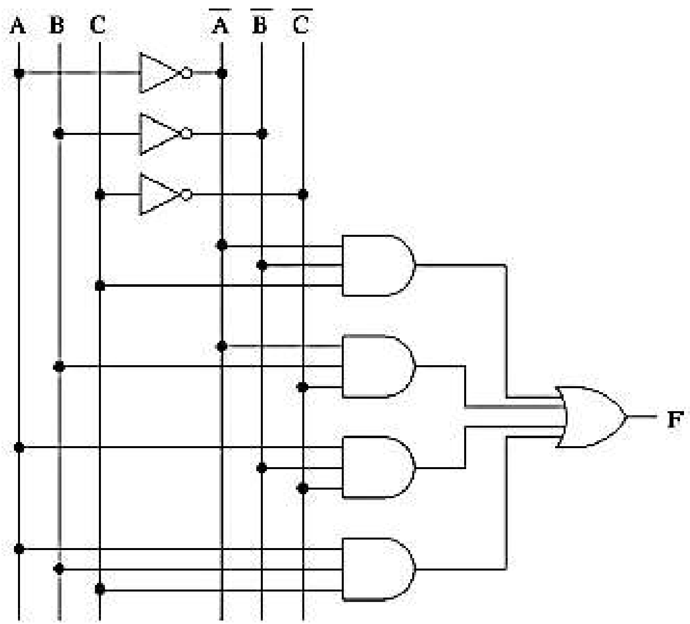
$$A\,B + B\,C + AC = \overline{\overline{A\,B} \cdot \overline{BC} \cdot \overline{AC}}$$

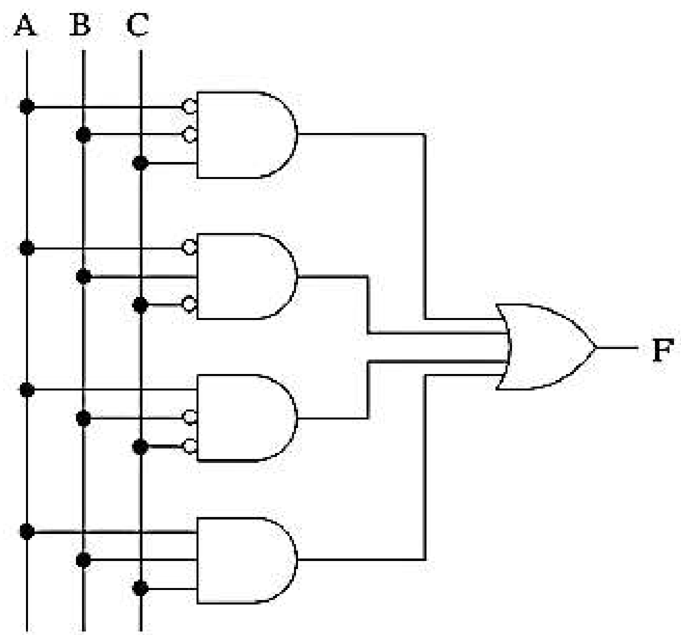
- Majority function



Implementation Using Other Gates

Bubble Notation





THANK YOU