

CHAPTER III

BOOLEAN ALGEBRA

- Boolean algebra is a form of algebra that deals with single digit binary values and variables.
- Values and variables can indicate some of the following binary pairs of values:
 - ON / OFF
 - TRUE / FALSE
 - HIGH / LOW
 - CLOSED / OPEN
 - 1 / 0

- Three fundamental operators in Boolean algebra
 - **NOT**: unary operator that complements represented as \bar{A} , A' , or $\sim A$
 - **AND**: binary operator which performs logical multiplication
 - i.e. **A** ANDed with **B** would be represented as **AB** or **A · B**
 - **OR**: binary operator which performs logical addition
 - i.e. **A** ORed with **B** would be represented as **A + B**

NOT

A	\bar{A}
0	1
1	0

AND

A	B	AB
0	0	0
0	1	0
1	0	0
1	1	1

OR

A	B	A + B
0	0	0
0	1	1
1	0	1
1	1	1

BOOL. OPERATIONS

BINARY BOOLEAN OPERATORS

- Below is a table showing all possible Boolean functions F_N given the two-inputs **A** and **B**.

A	B	F_0	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}	F_{15}
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

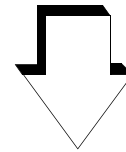
0	AB		A		B	A + B		A + B		$\overline{A + B}$		\overline{B}		\overline{A}		\overline{AB}	1
Null			Inhibition			$A \oplus B$		$A \oplus B$		$\overline{A \oplus B}$		Implication		Implication		Identity	

- Boolean expressions must be evaluated with the following order of operator precedence

- parentheses
- NOT
- AND
- OR

Example:

$$F = \overline{A(\overline{C + \overline{BD}}) + \overline{BC}} \overline{E}$$



$$F = \left(A \left(\overline{C + \overline{BD}} \right) + \overline{BC} \right) \overline{E}$$

The diagram shows the expression with nested brackets and braces to illustrate the order of operations. The innermost expression is \overline{BD} , which is then added to C inside a larger bracket. This result is then negated. This entire term is multiplied by A . Separately, B and C are negated and then added together. These two main terms are then added together, and the final result is negated to produce F .

- Example 1:

Evaluate the following expression when **A = 1, B = 0, C = 1**

$$F = C + \bar{C}B + B\bar{A}$$

- Solution

$$F = 1 + \bar{1} \cdot 0 + 0 \cdot \bar{1} = 1 + 0 + 0 = 1$$

- Example 2:

Evaluate the following expression when **A = 0, B = 0, C = 1, D = 1**

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

- Solution

$$F = 1 \cdot (0 \cdot \bar{1} \cdot 0 + \overline{0 \cdot \bar{0} + 1}) + 1 = 1 \cdot (0 + \bar{1} + 1) = 1 \cdot 1 = 1$$

BOOLEAN ALGEBRA

BASIC IDENTITIES

$$X + 0 = X$$

$$X + 1 = 1$$

$$X + X = X$$

$$X + X' = 1$$

$$(X')' = X$$

$$X + Y = Y + X$$

$$X + (Y + Z) = (X + Y) + Z$$

$$X(Y + Z) = XY + XZ$$

$$X + XY = X$$

$$X + X'Y = X + Y$$

$$(X + Y)' = X'Y'$$

$$XY + X'Z + YZ \\ = XY + X'Z$$

$$X \cdot 1 = X$$

$$X \cdot 0 = 0$$

$$X \cdot X = X$$

$$X \cdot X' = 0$$

$$XY = YX$$

$$X(YZ) = (XY)Z$$

$$X + YZ = (X + Y)(X + Z)$$

$$X(X + Y) = X$$

$$X(X' + Y) = XY$$

$$(XY)' = X' + Y'$$

$$(X + Y)(X' + Z)(Y + Z) \\ = (X + Y)(X' + Z)$$

Identity

Idempotent Law

Complement

Involution Law

Commutativity

Associativity

Distributivity

Absorption Law

Simplification

DeMorgan's Law

Consensus Theorem

- **Duality principle:**

- States that a Boolean equation remains valid if we take the dual of the expressions on both sides of the equals sign.
- The dual can be found by interchanging the **AND** and **OR** operators along with also interchanging the **0**'s and **1**'s.
- This is evident with the duals in the basic identities.
 - For instance: DeMorgan's Law can be expressed in two forms

$$(X + Y)' = X'Y'$$



as well as

$$(XY)' = X' + Y'$$



- Example: Simplify the following expression

$$F = BC + B\bar{C} + BA$$

- Simplification

$$F = B(C + \bar{C}) + BA$$

$$F = B \cdot 1 + BA$$

$$F = B(1 + A)$$

$$F = B$$

- Example: Simplify the following expression

$$F = A + \bar{A}B + \bar{A}\bar{B}C + \bar{A}\bar{B}\bar{C}D + \bar{A}\bar{B}\bar{C}\bar{D}E$$

- Simplification

$$F = A + \bar{A}(B + \bar{B}C + \bar{B}\bar{C}D + \bar{B}\bar{C}\bar{D}E)$$

$$F = A + B + \bar{B}C + \bar{B}\bar{C}D + \bar{B}\bar{C}\bar{D}E$$

$$F = A + B + \bar{B}(C + \bar{C}D + \bar{C}\bar{D}E)$$

$$F = A + B + C + \bar{C}D + \bar{C}\bar{D}E$$

$$F = A + B + C + \bar{C}(D + \bar{D}E)$$

$$F = A + B + C + D + \bar{D}E$$

$$F = A + B + C + D + E$$

- Example: Show that the following equality holds

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

- Simplification

$$\begin{aligned}\overline{A(\overline{B}\overline{C} + BC)} &= \overline{A} + \overline{(\overline{B}\overline{C} + BC)} \\ &= \overline{A} + (\overline{\overline{B}\overline{C}})(\overline{BC}) \\ &= \overline{A} + (B + C)(\overline{B} + \overline{C})\end{aligned}$$

STANDARD FORMS

SOP AND POS

- Boolean expressions can be manipulated into many forms.
- Some standardized forms are required for Boolean expressions to simplify communication of the expressions.

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

- Example:

$$F(A, B, C, D) = AB + \bar{B}C\bar{D} + AD$$

- **Products-of-sums (POS)**

- Example:

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

STANDARD FORMS

MINTERMS

- The following table gives the minterms for a **three-input** system

			m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7
A	B	C	$\bar{A}\bar{B}\bar{C}$	$\bar{A}\bar{B}C$	$\bar{A}B\bar{C}$	$\bar{A}BC$	$A\bar{B}\bar{C}$	$A\bar{B}C$	$AB\bar{C}$	ABC
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

STANDARD FORMS

SUM OF MINTERMS

- **Sum-of-minterms** standard form expresses the Boolean or switching expression in the form of a **sum of products** using **minterms**.
- For instance, the following Boolean expression using minterms

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

could instead be expressed as

$$F(A, B, C) = m_0 + m_1 + m_4 + m_5$$

or more compactly

$$F(A, B, C) = \sum m(0, 1, 4, 5) = \text{one-set}(0, 1, 4, 5)$$

Minterms are products, so
this is called a
"sum of products" (SOP)

STANDARD FORMS

MAXTERMS

- The following table gives the maxterms for a **three-input** system

			M_0	M_1	M_2	M_3	M_4	M_5	M_6	M_7
			$A + B + C$	$A + \bar{B} + C$	$\bar{A} + B + C$	$\bar{A} + \bar{B} + C$				
			$A + B + \bar{C}$	$A + \bar{B} + \bar{C}$	$\bar{A} + B + \bar{C}$	$\bar{A} + \bar{B} + \bar{C}$				
A	B	C								
0	0	0	0	1	1	1	1	1	1	1
0	0	1	1	0	1	1	1	1	1	1
0	1	0	1	1	0	1	1	1	1	1
0	1	1	1	1	1	0	1	1	1	1
1	0	0	1	1	1	1	0	1	1	1
1	0	1	1	1	1	1	1	0	1	1
1	1	0	1	1	1	1	1	1	0	1
1	1	1	1	1	1	1	1	1	1	0

STANDARD FORMS

PRODUCT OF MAXTERMS

- **Product-of-maxterms** standard form expresses the Boolean or switching expression in the form of **product of sums** using **maxterms**.
- For instance, the following Boolean expression using maxterms

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

could instead be expressed as

$$F(A, B, C) = M_1 \cdot M_4 \cdot M_7$$

or more compactly as

$$F(A, B, C) = \prod M(1, 4, 7) = \text{zero-set}(1, 4, 7)$$

STANDARD FORMS

MINTERM AND MAXTERM EXP.

- Given an arbitrary Boolean function, such as

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

how do we form the canonical form for:

- **sum-of-minterms**

- Expand the Boolean function into a sum of products. Then take each term with a missing variable **X** and **AND** it with **X + \bar{X}** .

- **product-of-maxterms**

- Expand the Boolean function into a product of sums. Then take each factor with a missing variable **X** and **OR** it with **$X\bar{X}$** .

STANDARD FORMS

FORMING SUM OF MINTERMS

- Example

$$\begin{aligned}
 F(A, B, C) &= AB + \bar{B}(\bar{A} + \bar{C}) = AB + \bar{A}\bar{B} + \bar{B}\bar{C} \\
 &= AB(C + \bar{C}) + \bar{A}\bar{B}(C + \bar{C}) + (A + \bar{A})\bar{B}\bar{C} \\
 &= \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + A\bar{B}\bar{C} + AB\bar{C} + ABC \\
 &= \sum m(0, 1, 4, 6, 7)
 \end{aligned}$$

A	B	C	F
0	0	0	1 ← 0
0	0	1	1 ← 1
0	1	0	0
0	1	1	0
1	0	0	1 ← 4
1	0	1	0
1	1	0	1 ← 6
1	1	1	1 ← 7

Minterms listed as
1s in Truth Table

STANDARD FORMS

FORMING PROD OF MAXTERMS

- STANDARD FORMS
- PRODUCT OF MAXTERMS
- MINTERM & MAXTERM
- FORM SUM OF MINTERMS

- Example

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

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

(using distributivity)

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

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

$$= \prod M(2, 3, 5)$$

Maxterms are sums, so
this is called a
"product of sums" (POS)

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

← 2

← 3

← 5

Maxterms listed as
0s in Truth Table

STANDARD FORMS

CONVERTING MIN AND MAX

- Converting between sum-of-minterms and product-of-maxterms
- The two are complementary, as seen by the truth tables.
- To convert interchange the \sum and \prod , then use missing terms.
 - Example: The example from the previous slides

$$F(A, B, C) = \sum m(0, 1, 4, 6, 7)$$

$$F'(A,B,C) = \text{SUM } m(2,3,5)$$

is re-expressed as

$$F(A, B, C) = \prod M(2, 3, 5)$$

$$F'(A,B,C) = \text{PI } M(0,1,4,6,7)$$

where the numbers 2, 3, and 5 were missing from the minterm representation.

SIMPLIFICATION

KARNAUGH MAPS

- Often it is desired to simplify a Boolean function. A quick graphical approach is to use Karnaugh maps.

**2-variable
Karnaugh map**

		B	
		0	1
A	0	0	0
	1	0	1

$$F = AB$$

**3-variable
Karnaugh map**

		BC			
		00	01	11	10
A	0	0	1	1	0
	1	0	1	1	1

$$F = AB + C$$

**4-variable
Karnaugh map**

		CD			
		00	01	11	10
AB	00	0	1	0	0
	01	0	1	0	0
	11	1	1	1	1
	10	0	1	0	0

$$F = AB + \bar{C}D$$

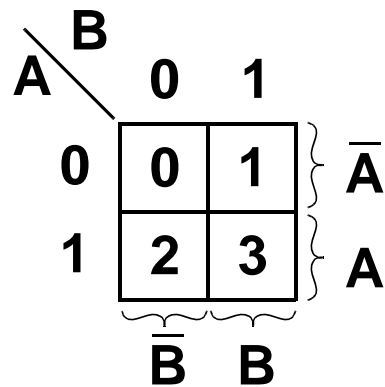
SIMPLIFICATION

KARNAUGH MAP ORDERING

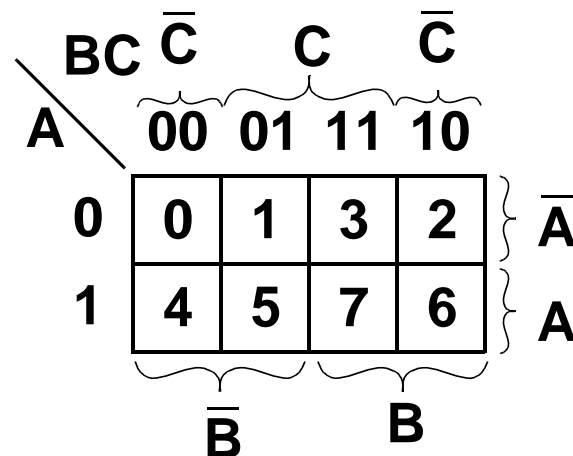
- STANDARD FORMS
- SIMPLIFICATION
-KARNAUGH MAPS

- Notice that the ordering of cells in the map are such that moving from one cell to an adjacent cell only changes one variable.

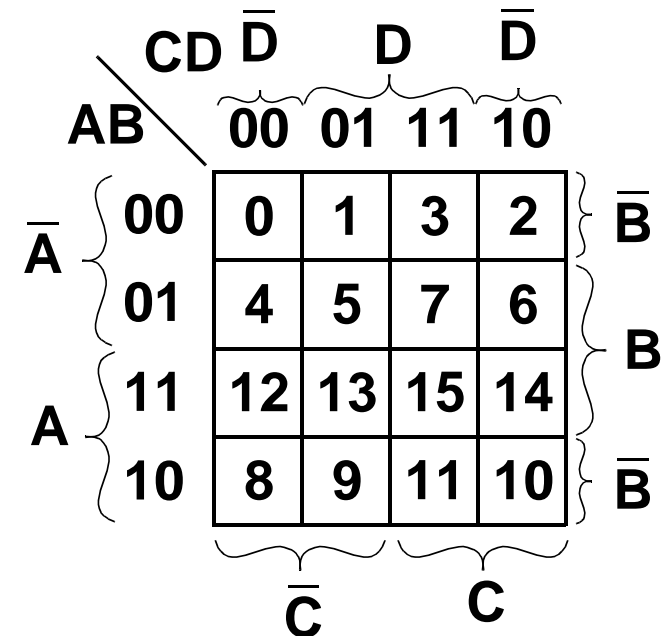
**2-variable
Karnaugh map**



**3-variable
Karnaugh map**



**4-variable
Karnaugh map**



- This ordering allows for grouping of minterms/maxterms for simplification.

SIMPLIFICATION

IMPLICANTS

- **Implicant**
 - Bubble covering only 1s (size of bubble must be a power of 2).
- **Prime implicant**
 - Bubble that is expanded as big as possible (but increases in size by powers of 2).
- **Essential prime implicant**
 - Bubble that contains a 1 covered only by itself and no other prime implicant bubble.
- **Non-essential prime implicant**
 - A 1 that can be bubbled by more than one prime implicant bubble.

		CD			
		00	01	11	10
AB	00	1	1	0	0
	01	0	0	1	0
	11	0	1	1	1
	10	1	1	0	0

- **Procedure for finding the SOP from a Karnaugh map**
 - Step 1: Form the 2-, 3-, or 4-variable Karnaugh map as appropriate for the Boolean function.
 - Step 2: Identify all essential prime implicants for 1s in the Karnaugh map
 - Step 3: Identify non-essential prime implicants for 1s in the Karnaugh map.
 - Step 4: For each essential and one selected non-essential prime implicant from each set, determine the corresponding product term.
 - Step 5: Form a sum-of-products with all product terms from previous step.

SIMPLIFICATION

EXAMPLE FOR SOP (1)

SOP: SUM OF PRODUCTS

- SIMPLIFICATION
- KARNAUGH MAP ORDER
- IMPLICANTS
- PROCEDURE FOR SOP

- Simplify the following Boolean function

$$F(A, B, C) = \sum m(0, 1, 4, 5) = \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + A\bar{B}\bar{C} + A\bar{B}C$$

- Solution:

		BC			
		00	01	11	10
A	0	1	1	0	0
	1	1	1	0	0

zero-set(2, 3, 6, 7)

one-set(0, 1, 4, 5)

- The essential prime implicants are \bar{B} .
- There are no non-essential prime implicants.
- The sum-of-products solution is $F = \bar{B}$.

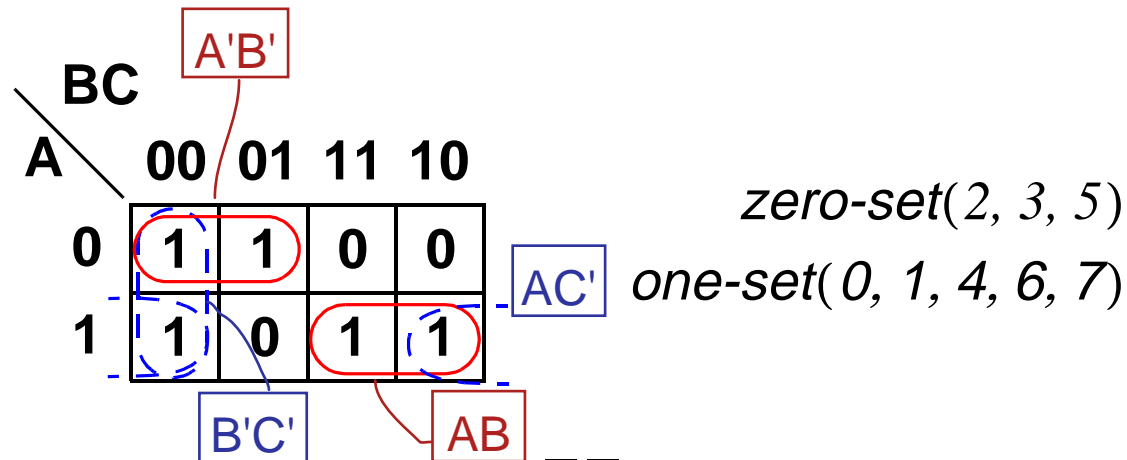
SIMPLIFICATION

EXAMPLE FOR SOP (2) SOP: SUM OF PRODUCTS

- Simplify the following Boolean function

$$F(A, B, C) = \sum m(0, 1, 4, 6, 7) = \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + A\bar{B}\bar{C} + AB\bar{C} + ABC$$

- Solution:



- The essential prime implicants are $\bar{A}\bar{B}$ and AB .
- The non-essential prime implicants are $\bar{B}\bar{C}$ or $A\bar{C}$.
- The sum-of-products solution is

$$F = AB + \bar{A}\bar{B} + \bar{B}\bar{C} \text{ or } F = AB + \bar{A}\bar{B} + A\bar{C}.$$

SIMPLIFICATION

PROCEDURE FOR POS POS: PRODUCT OF SUMS

- **Procedure for finding the SOP from a Karnaugh map**
 - Step 1: Form the 2-, 3-, or 4-variable Karnaugh map as appropriate for the Boolean function.
 - Step 2: Identify all essential prime implicants for **0s** in the Karnaugh map
 - Step 3: Identify non-essential prime implicants for **0s** in the Karnaugh map.
 - Step 4: For each essential and one selected non-essential prime implicant from each set, determine the corresponding sum term.
 - Step 5: Form a product-of-sums with all sum terms from previous step.

SIMPLIFICATION

EXAMPLE FOR POS (1)

POS: PRODUCT OF SUMS

- SIMPLIFICATION
- PROCEDURE FOR SOP
- EXAMPLE FOR SOP
- PROCEDURE FOR POS

- Simplify the following Boolean function

$$F(A, B, C) = \prod M(2, 3, 5) = (A + \bar{B} + C)(A + \bar{B} + \bar{C})(\bar{A} + B + \bar{C})$$

- Solution:

		BC			
		00	01	11	10
A	0	1	1	0	0
	1	1	0	1	1

zero-set(2, 3, 5)

one-set(0, 1, 4, 6, 7)

- The essential prime implicants are $\bar{A} + B + \bar{C}$ and $A + \bar{B}$.
- There are no non-essential prime implicants.
- The product-of-sums solution is $F = (A + \bar{B})(\bar{A} + B + \bar{C})$.

SIMPLIFICATION

EXAMPLE FOR POS (2)

- Simplify the following Boolean function

$$F(\mathbf{A}, \mathbf{B}, \mathbf{C}) = \prod M(0, 1, 5, 7, 8, 9, 15)$$

- Solution:

- The essential prime implicants

are $\mathbf{B} + \mathbf{C}$ and $\bar{\mathbf{B}} + \bar{\mathbf{C}} + \bar{\mathbf{D}}$.

zero-set(0, 1, 5, 7, 8, 9, 15)

one-set(2, 3, 4, 6, 10, 11, 12, 13, 14)

- The non-essential prime implicants

can be $\mathbf{A} + \bar{\mathbf{B}} + \bar{\mathbf{D}}$ or $\mathbf{A} + \mathbf{C} + \bar{\mathbf{D}}$.

- The product-of-sums solution can be either

$$F = (\mathbf{B} + \mathbf{C})(\bar{\mathbf{B}} + \bar{\mathbf{C}} + \bar{\mathbf{D}})(\mathbf{A} + \bar{\mathbf{B}} + \bar{\mathbf{D}})$$

or

$$F = (\mathbf{B} + \mathbf{C})(\bar{\mathbf{B}} + \bar{\mathbf{C}} + \bar{\mathbf{D}})(\mathbf{A} + \mathbf{C} + \bar{\mathbf{D}})$$

		CD			
		00	01	11	10
AB	00	0	0	1	1
	01	1	0	0	1
	11	1	1	0	1
	10	0	0	1	1

SIMPLIFICATION

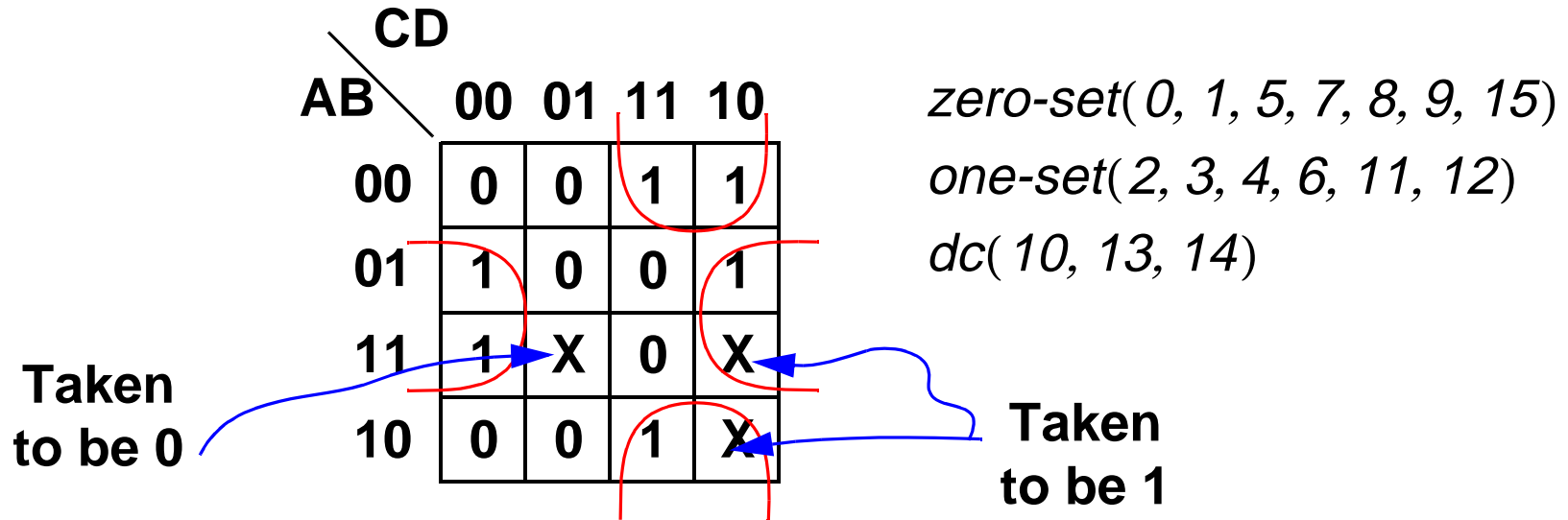
DON'T-CARE CONDITION

- Switching expressions are sometimes given as **incomplete**, or with **don't-care conditions**.
 - Having don't-care conditions can simplify Boolean expressions and hence simplify the circuit implementation.
 - Along with the *zero-set*() and *one-set*(), we will also have *dc*().
 - Don't-cares conditions in Karnaugh maps
 - Don't-cares will be expressed as an "X" or "-" in Karnaugh maps.
 - Don't-cares can be bubbled along with the 1s or 0s depending on what is more convenient and help simplify the resulting expressions.

SIMPLIFICATION

DON'T-CARE EXAMPLE (1)

- Find the SOP simplification for the following Karnaugh map

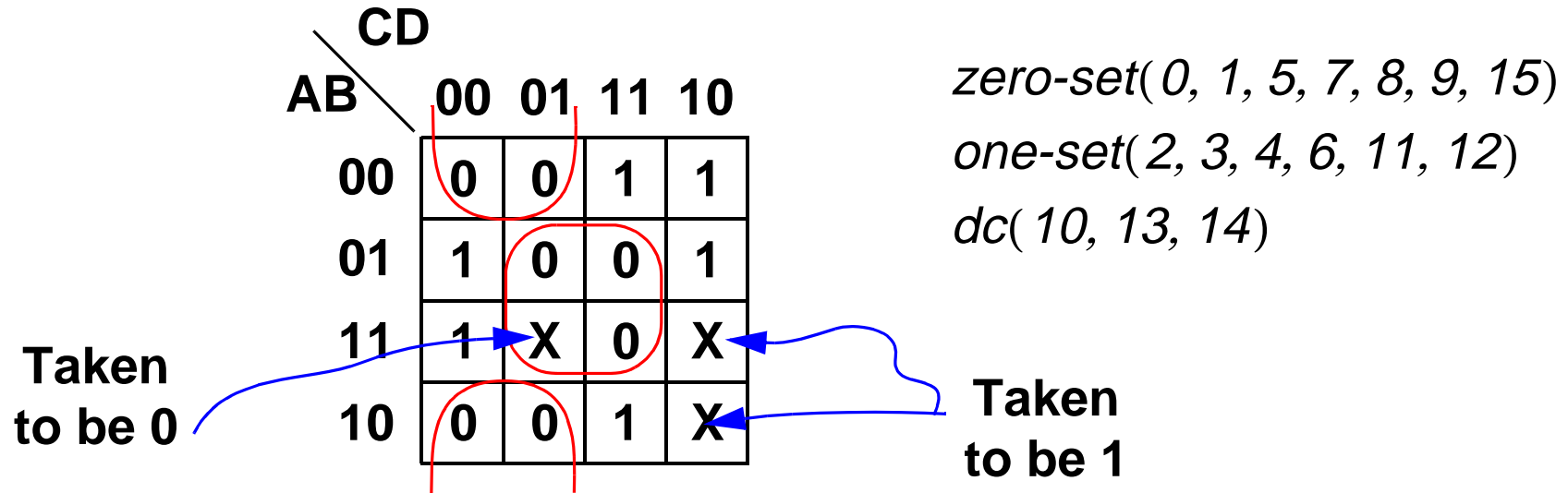


- Solution:
 - The essential prime implicants are $\mathbf{B\bar{D}}$ and $\mathbf{\bar{B}C}$.
 - There are no non-essential prime implicants.
 - The sum-of-products solution is $\mathbf{F = \bar{B}C + B\bar{D}}$.

SIMPLIFICATION

DON'T-CARE EXAMPLE (2)

- Find the POS simplification for the following Karnaugh map



- Solution:

- The essential prime implicants are $\mathbf{B + C}$ and $\mathbf{\bar{B} + \bar{D}}$.
- There are no non-essential prime implicants.
- The product-of-sums solution is $\mathbf{F = (B + C)(\bar{B} + \bar{D})}$.