Jiangxi University of Science and Technology

DIGITAL DESIGN

Lecture 11: The Karnaugh Map_2 Minimization

Small Review: Karnaugh Maps (K-Maps)

- A visual way to simplify logic expressions
- It gives the most simplified form of the expression
- Simplification of logic expression using Boolean algebra is awkward because:
  - it lacks specific rules to predict the most suitable next step in the simplification process
  - it is difficult to determine whether the simplest form has been achieved.

Don’t-Care Conditions

- Some logic circuits can be designed so that there are certain input conditions for which there are no specified output levels.
- A circuit designer is free to make the output for any don’t care condition either a 0 or a 1 in order to produce the simplest output expression.

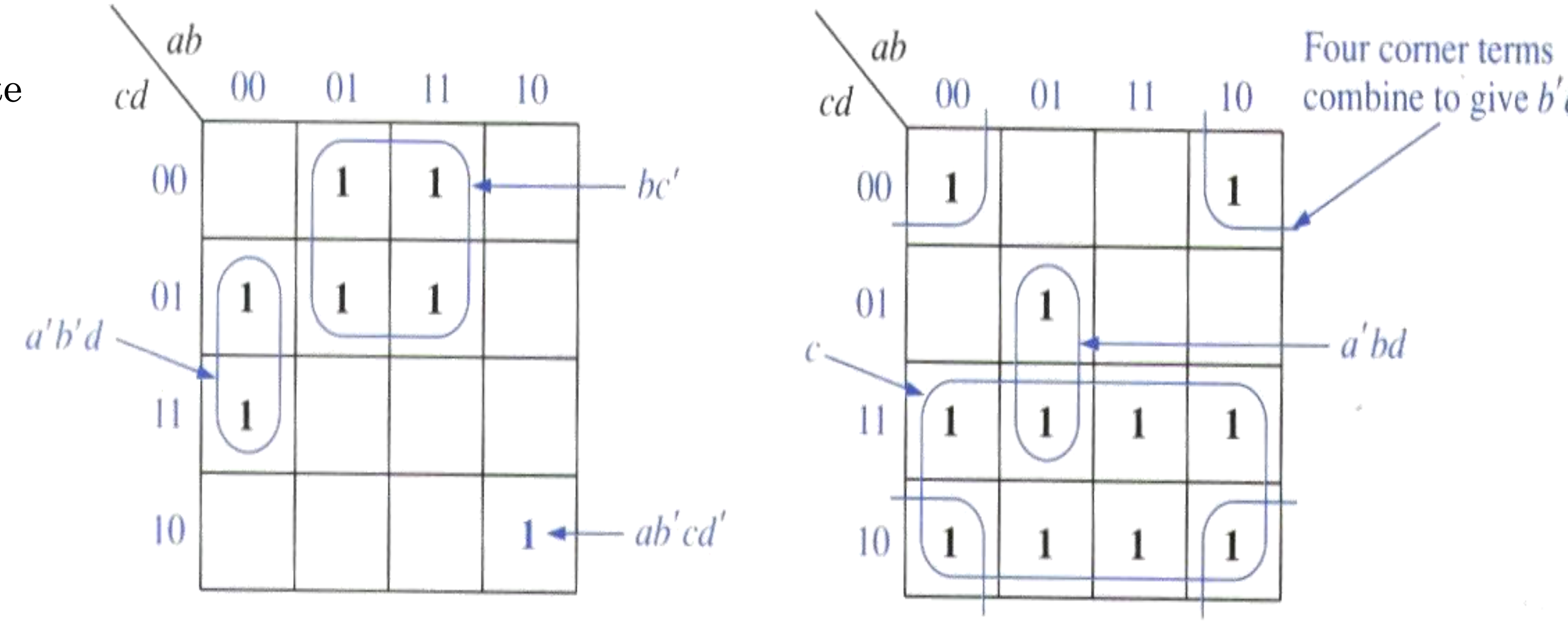
Don’t Care Conditions

- In some circuits we care only about the output for some combinations of input values, because other combinations of input values are not possible or never occur. This gives us freedom in producing a simple circuit with the desired output because the output values for all those combinations that never occur can be arbitrarily chosen.
- The values of the function for these combinations are called don’t care conditions. A d is used in a K-map to mark those combinations of values of the variables for which the function can be arbitrarily assigned.
- In the minimization process we can assign 1s as values to those combinations of the input values that lead to the largest blocks in the K-map.

Simplification Example

Minterms are combined in groups of 2, 4, or 8 to eliminate 1, 2, 3 variables.

Corner terms.



$f_1 = \Sigma m(1, 3, 4, 5, 10, 12, 13)$
$= bc' + a'b'd + ab'cd'$

(a)

$f_2 = \Sigma m(0, 2, 3, 5, 6, 7, 8, 10, 11, 14, 15)$
$= c + b'd' + a'bd$

(b)

Simplification with Don’t Care

- Don’t care “x” is covered if it helps. Otherwise leave it along.

$$f = \Sigma m(1, 3, 5, 7, 9) + \Sigma d(6, 12, 13)$$

cd \ ab	00	01	11	10
00			x	
01	1	1	x	1
11	1	1		
10		x		

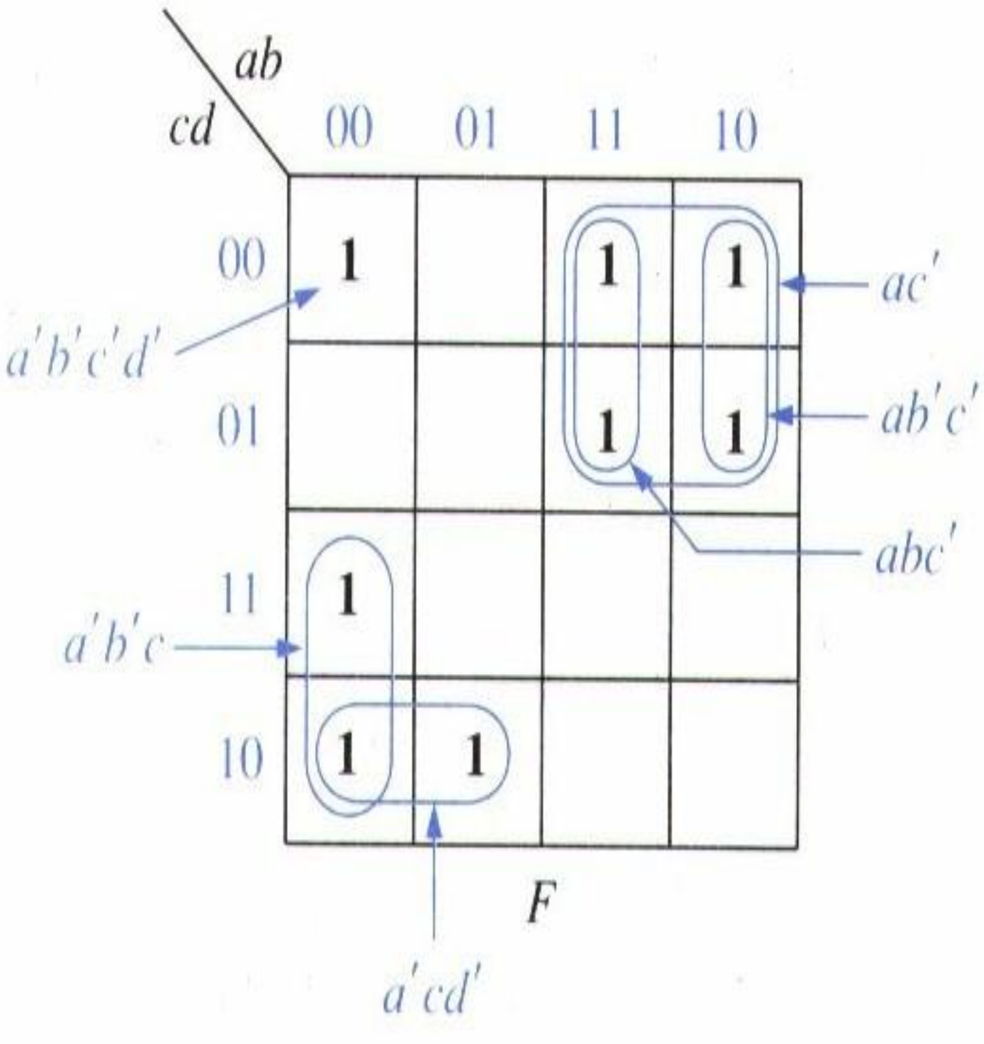
$$f = \Sigma m(1, 3, 5, 7, 9) + \Sigma d(6, 12, 13)$$
$$= a'd + c'd$$

Determination of Minimum Expressions Using Essential Prime Implicates

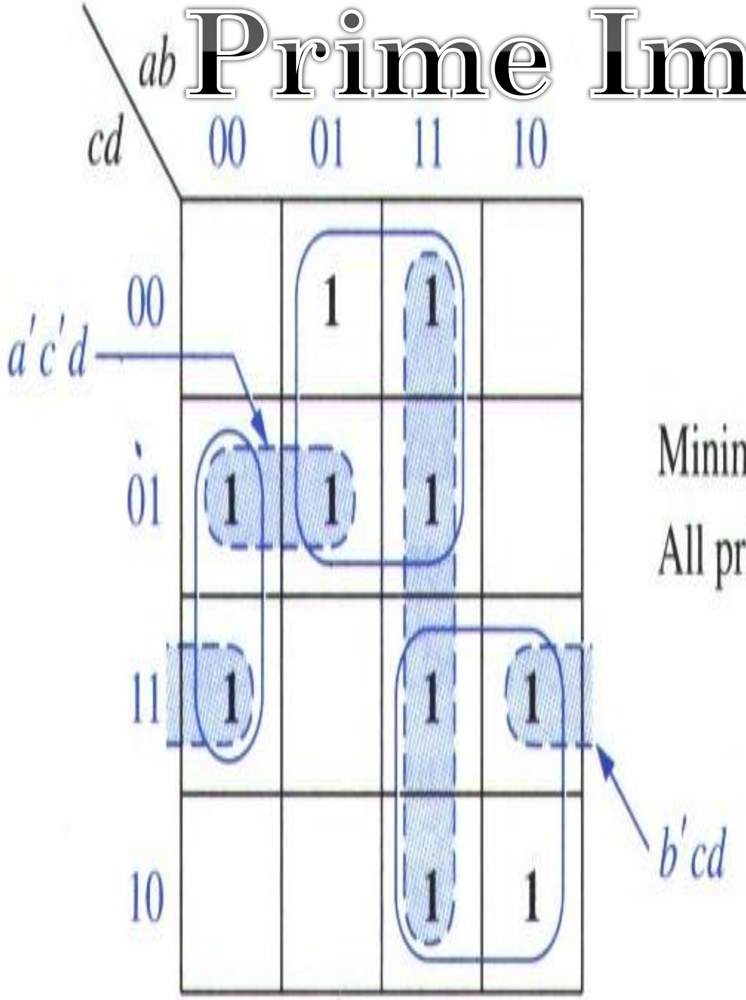
- Definitions:
  - **Implicants**: An implicant of a function F is a single element of the on set (1) or any group of elements that can be combined together in a K-map.
  - **Prime Implicants**: An implicant that cannot be combined with another to eliminate a literal.
  - **Essential Prime Implicants**: If a particular element of the on-set is covered by a single prime implicant. That implicant is called an essential prime implicant.
- All essential primes must be part of the minimized expression.

Implicant of F

- Implicant
  - Any single 1 or any group of 1's
- Prime implicant
  - An implicant that can not be combined with another term to eliminate a variable.
    - a'b'c, a'cd', ac' are prime implicant.
    - a'b'c'd' is not (combined with a'b'cd'). abc' and ab'c' are not.



Prime Implicant



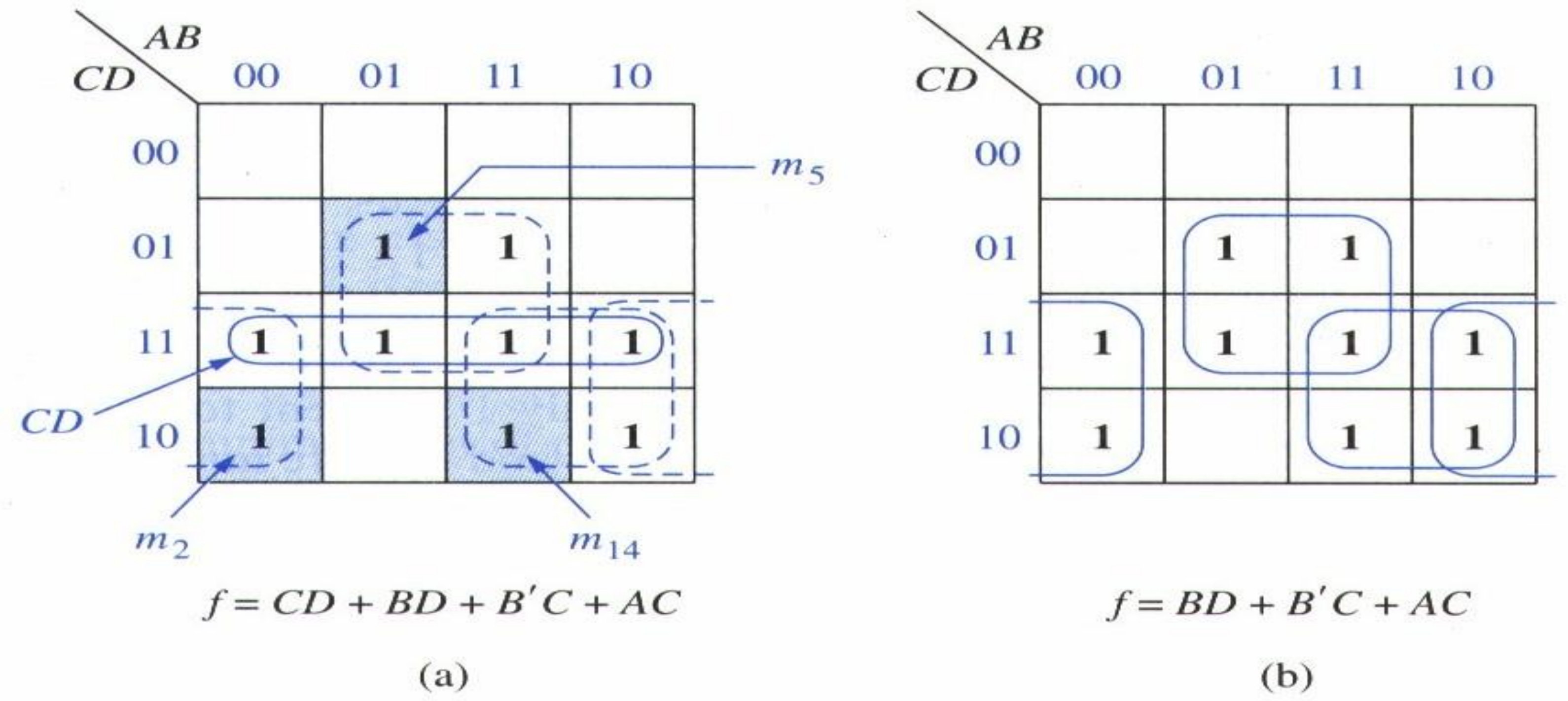
Minimum solution: $F = a'b'd + bc' + ac$

All prime implicants: $a'b'd$, bc', ac, $a'c'd$, ab, $b'cd$

- A single 1 which is not adjacent to any other 1's.
- Two adjacent 1's which are not contained in a group of four 1's. And so on.
  - Shaded loops are also prime implicants, but not part of the minimum solution.
  - a'c'd and b'cd are already covered by other group.

  So we do not need them.

Essential Prime Implicants for Minimum SOP

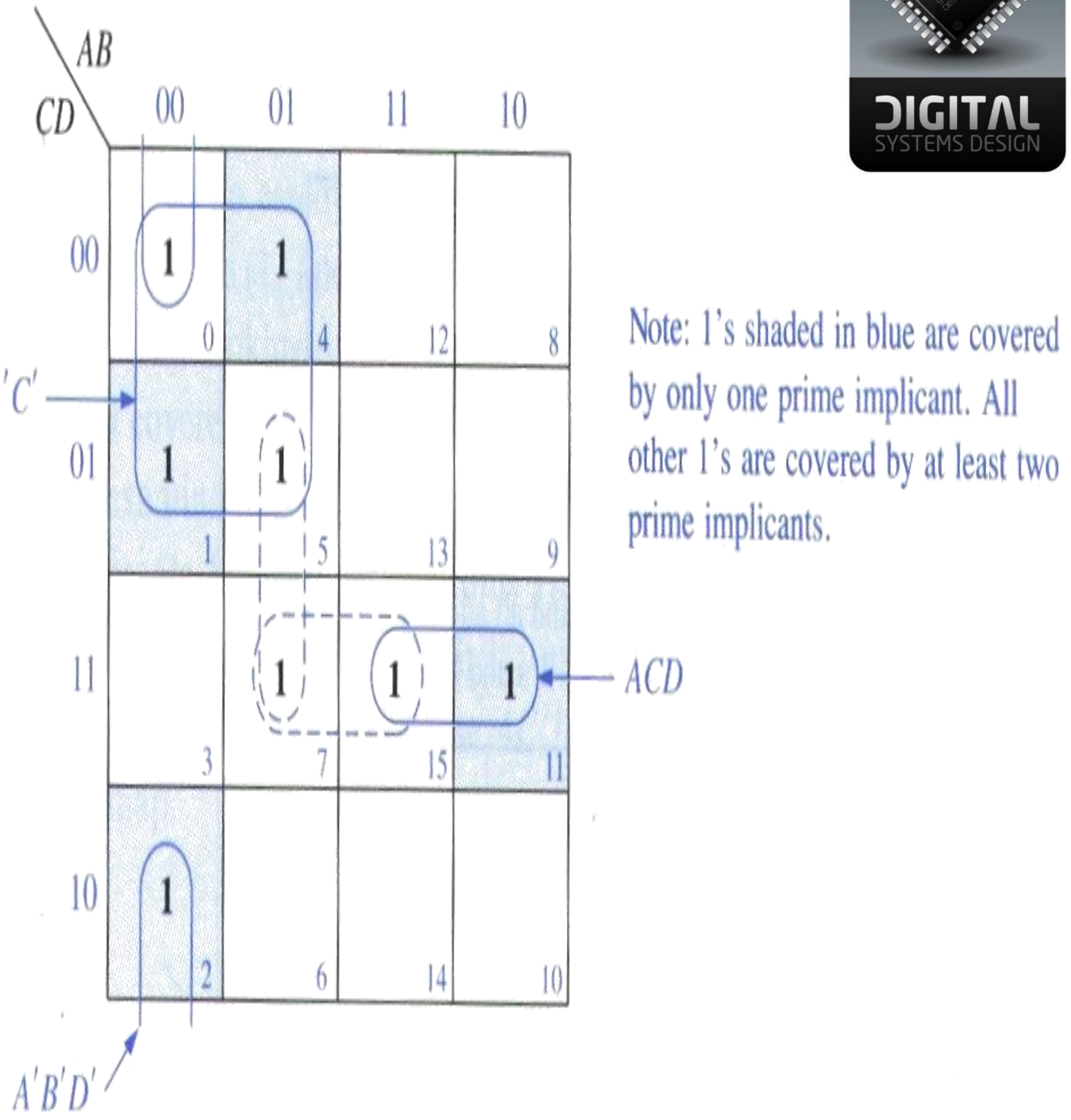


- If CD is chosen first, then f has 4 terms. We don't need CD since it is covered by other group. CD is not a essential prime implicant.
- m2 is essential prime implicant since it is covered only be one prime implicant. So does m5, and m14. We need them in the answer.

Rule of Thumb

•Look at all squares adjacent to a minterm.

- If the given minterm and all of the 1's adjacent to it are covered by a single term, then that term is an essential prime implicant
- But if it is covered by more than two prime implicant, we can not tell whether this term is essential or not.
- The solution is
  A'C' + A'B'D' +ACD + (A'BD or BCD).



Four-Variable K-Maps

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

$$f = \sum(0,8) = \overline{B} \bullet \overline{C} \bullet \overline{D}$$

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

$$f = \sum(5,13) = B \bullet \overline{C} \bullet D$$

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

$$f = \sum(13,15) = A \bullet B \bullet D$$

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

$$f = \sum(4,6) = \overline{A} \bullet B \bullet \overline{D}$$

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

$$f = \sum(2,3,6,7) = \overline{A} \bullet C$$

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

$$f = \sum(4,6,12,14) = B \bullet \overline{D}$$

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

$$f = \sum(2,3,10,11) = \overline{B} \bullet C$$

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

$$f = \sum(0,2,8,10) = \overline{B} \bullet \overline{D}$$

Four-Variable K-Maps

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

$f = \sum(4,5,6,7) = \overline{A} \bullet B$

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

$f = \sum(3,7,11,15) = C \bullet D$

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

$f = \sum(0,3,5,6,9,10,12,15)$

$f = A \otimes B \otimes C \otimes D$

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

$f = \sum(1,2,4,7,8,11,13,14)$

$f = A \oplus B \oplus C \oplus D$

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

$f = \sum(1,3,5,7,9,11,13,15)$

$f = D$

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

$f = \sum(0,2,4,6,8,10,12,14)$

$f = \overline{D}$

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

$f = \sum(4,5,6,7,12,13,14,15)$

$f = B$

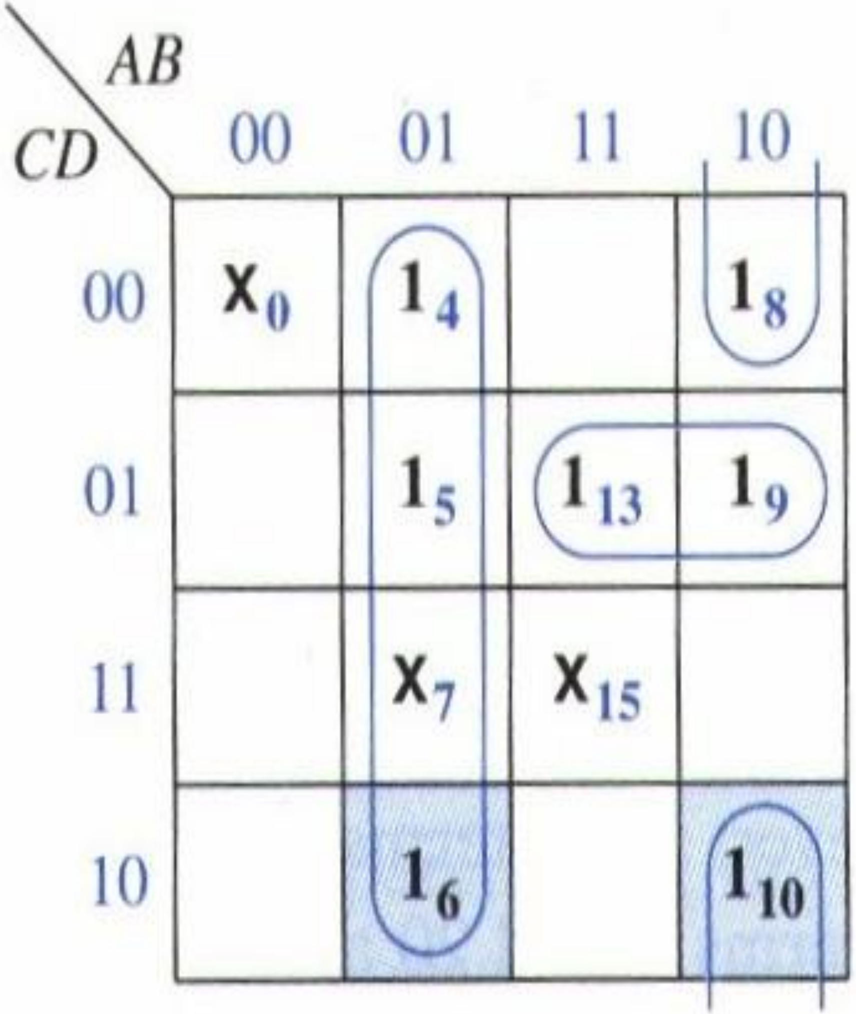
AB \ CD	00	01	11	10
00	1	1	1	1
01	0	0	0	0
11	0	0	0	0
10	1	1	1	1

$f = \sum(0,1,2,3,8,9,10,11)$

$f = \overline{B}$

Example

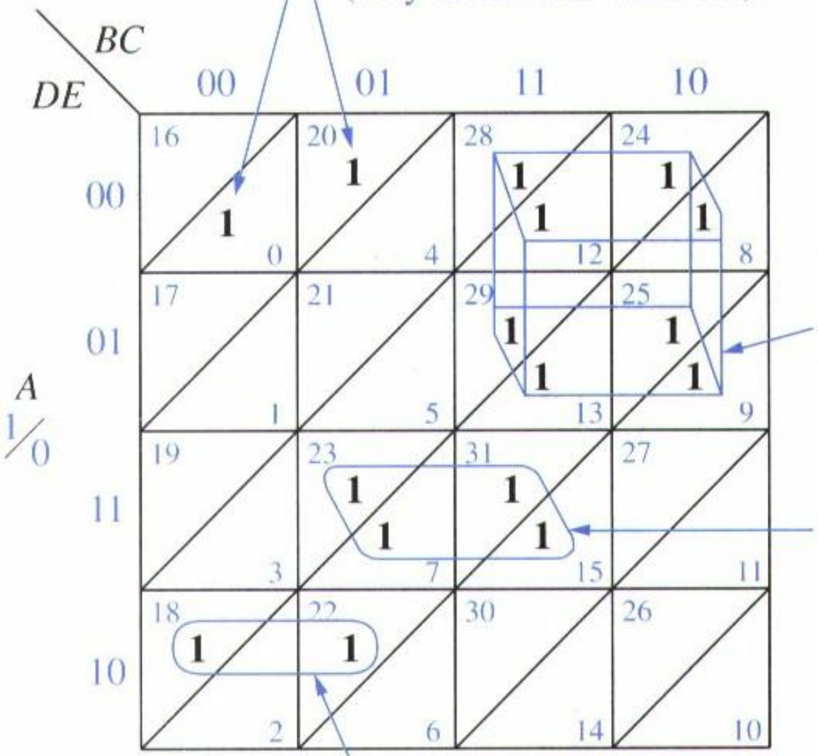
- Find the 1 that is covered by only one term first (Do not share with other circle).



Shaded 1's are covered by only one prime implicant

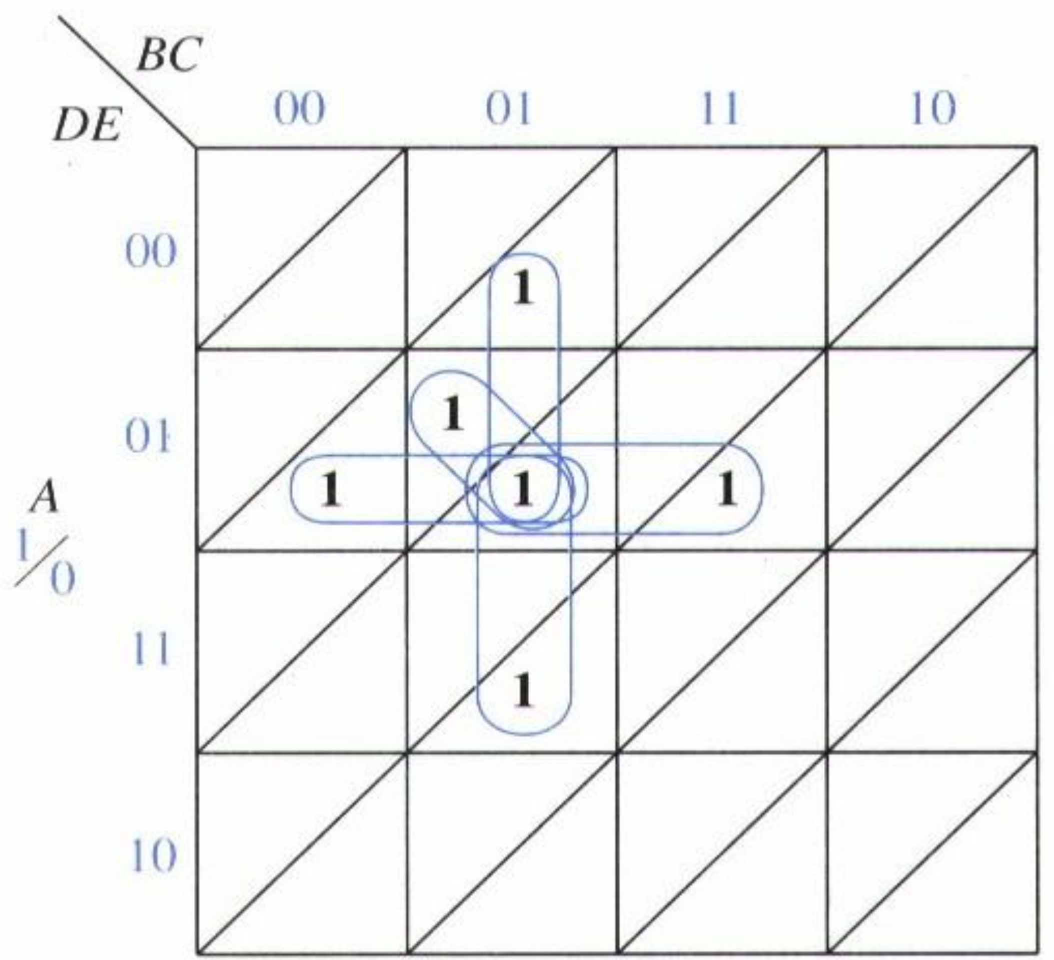
5-Variable K Map

- Use two 4-variable map to form a 5-variable K map (16 + 16 = 32) (A,B,C,D,E)
    - A’ in the bottom layer
    - A in the top layer.

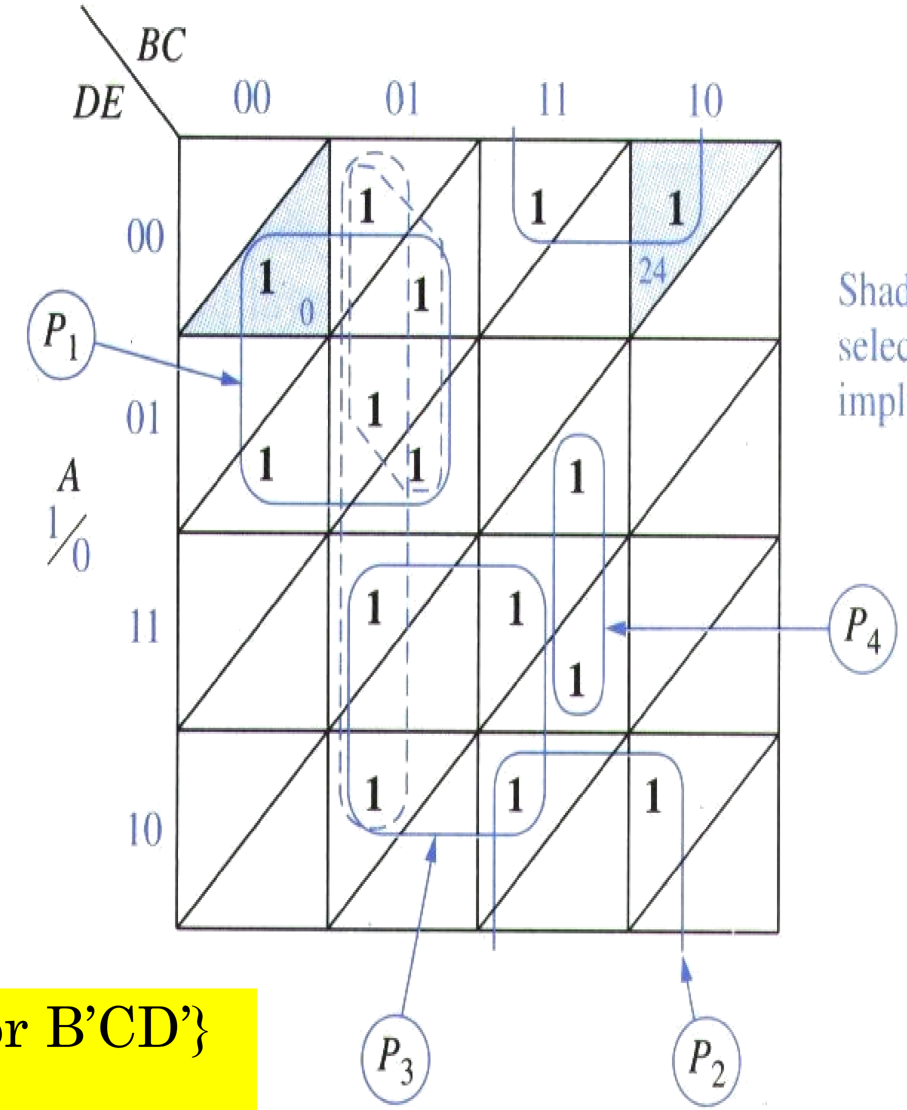


5 Neighbors

- Same plane and above or under



Example: 5-variables



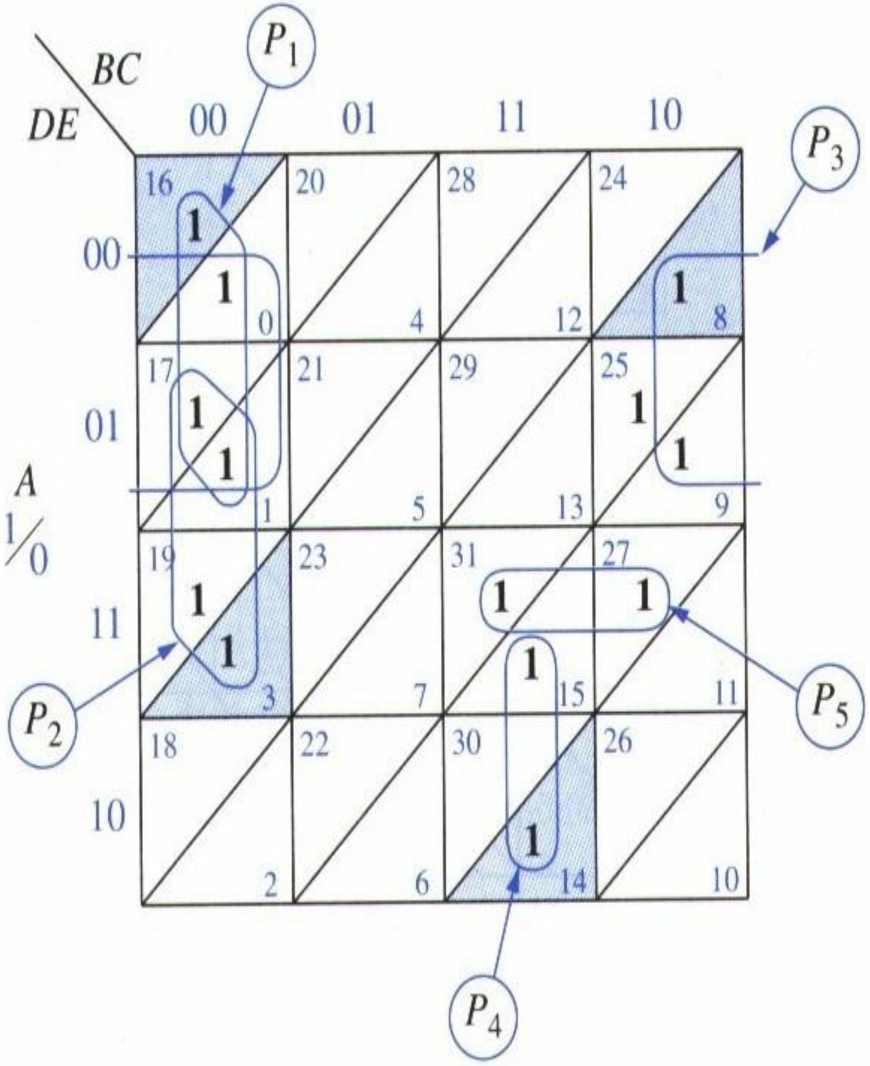
Shaded 1's are used to select essential prime implicants.

- Ans: F = A'B'D' + ABE' + ACD + A'BCE + {AB'C or B'CD'}
  - P1 + P2 + P3 + P4 + AB'C or B'CD'

One More

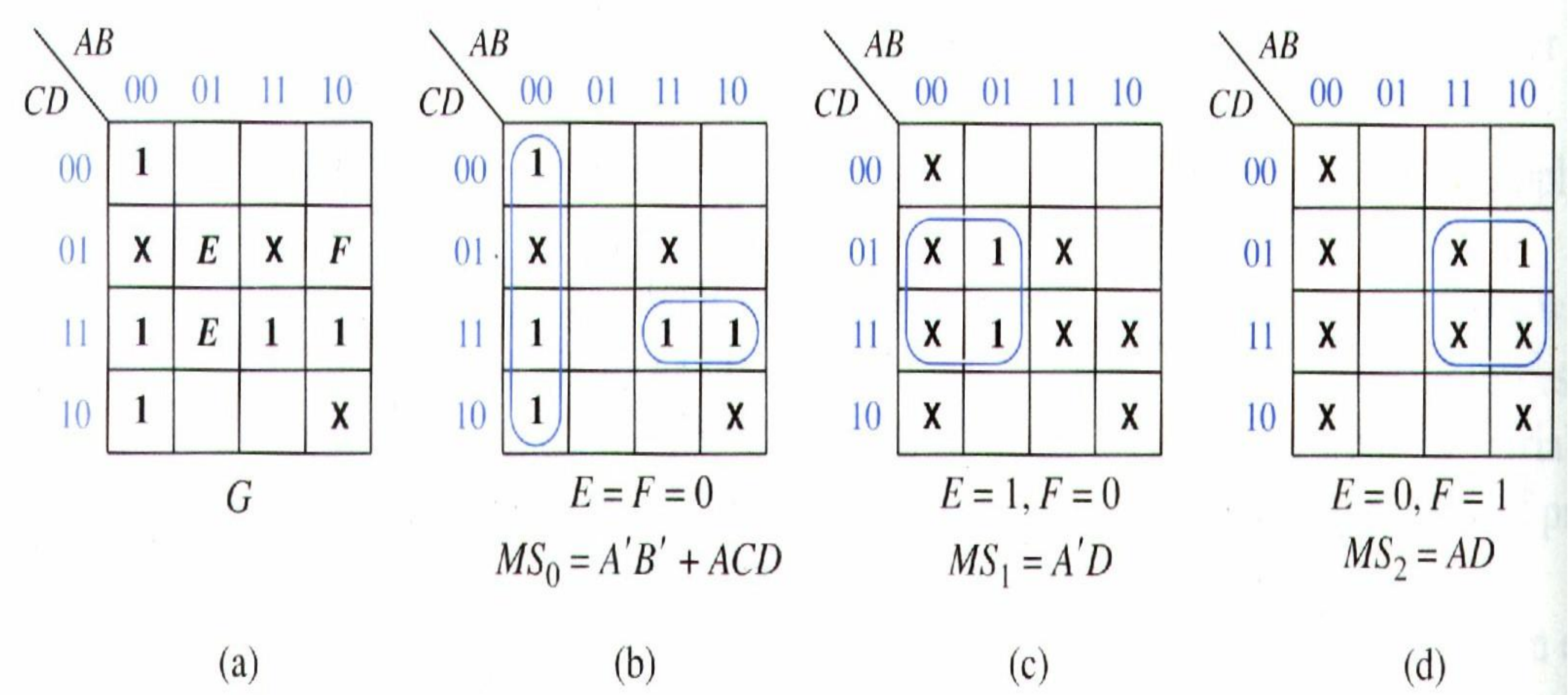
$F = B'C'D' + B'C'E + A'C'D' + A'BCD + ABDE + \{C'D'E \text{ or } AC'E\}$

$(17,19,25,27 = AC'E), (1,9,17,25 = C'D'E)$



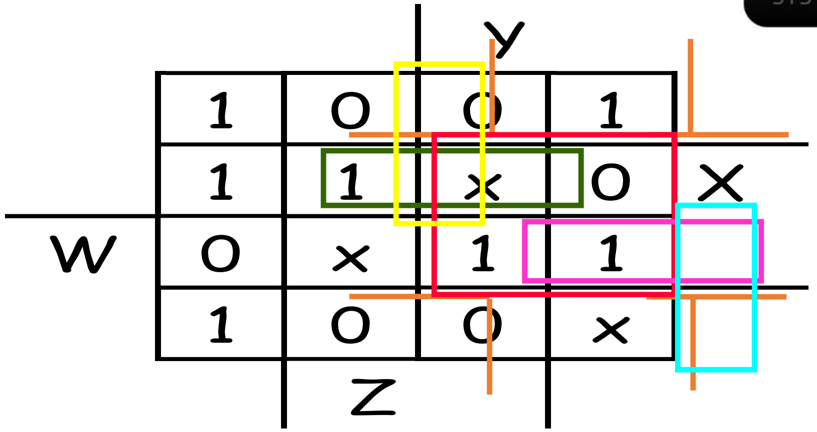
Simplification Using Map-Entered Variables

- Extend K-map for more variables.
  - When E appears in a square, if E = 1, then the corresponding minterm is present in the function G.
  - G (A,B,C,D,E,F) = $m_0 + m_2 + m_3 + Em_5 + Em_7 + Fm_9 + m_{11} + m_{15}$ + (don't care terms)



Solutions for practice K-map 3

- Find a MSP for: f(w,x,y,z) = Σm(0,2,4,5,8,14,15), d(w,x,y,z) = Σm(7,10,13)



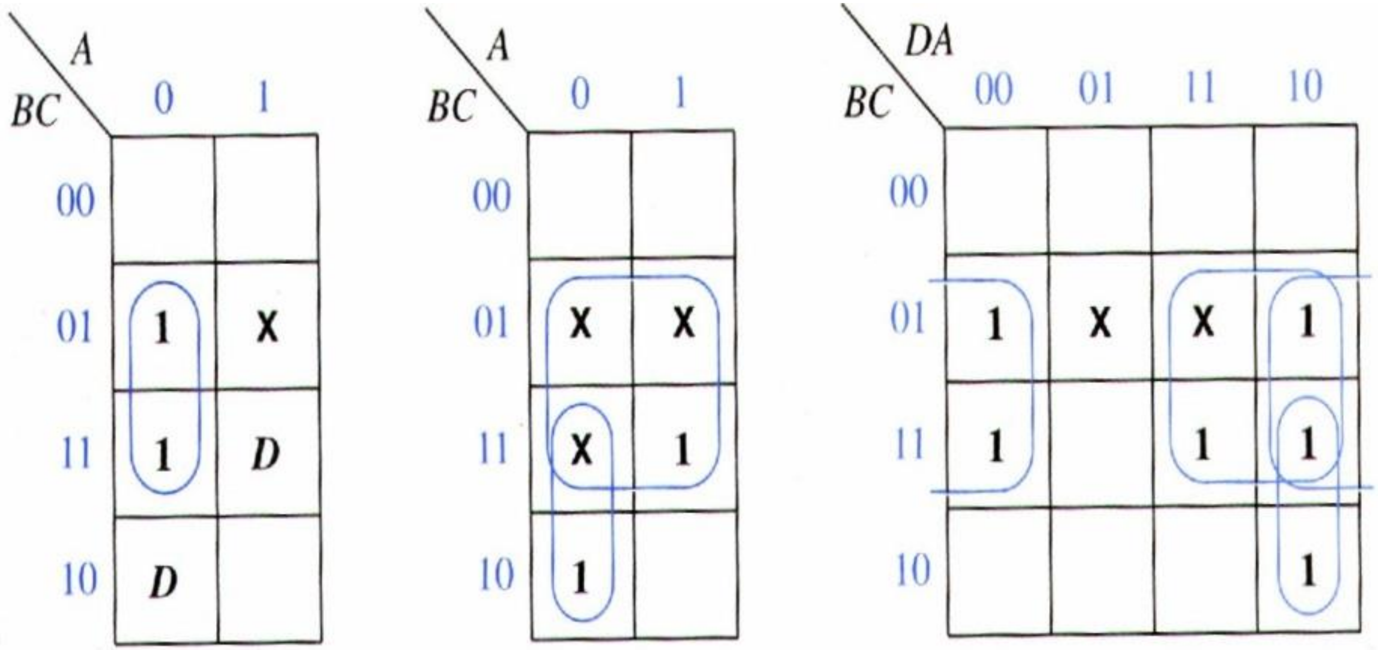
All prime implicants are circled. We can treat X's as 1s if we want, so the red group includes two X's, and the light blue group includes one X.

The *only* essential prime implicant is x'z'. The red group is not essential because the minterms in it also appear in other groups.

The MSP is x'z' + wxy + w'xy'. It turns out the red group is redundant; we can cover all of the minterms in the map without it.

Map-Entered Variable

- F(A,B,C,D) = A'B'C + A'BC + A'BC'D + ABCD + (AB'C), (don't care)
  - Choose D as a map-entered variable.
  - When D = 0, F = A'C (Fig. a)
  - When D = 1, F = C + A'B (Fig. b)
    - two 1's are changed to x's since they are covered in Fig. a.
- F = A'C + D(C+A'B) = A'C + CD + A'BD



(a) (b) (c)

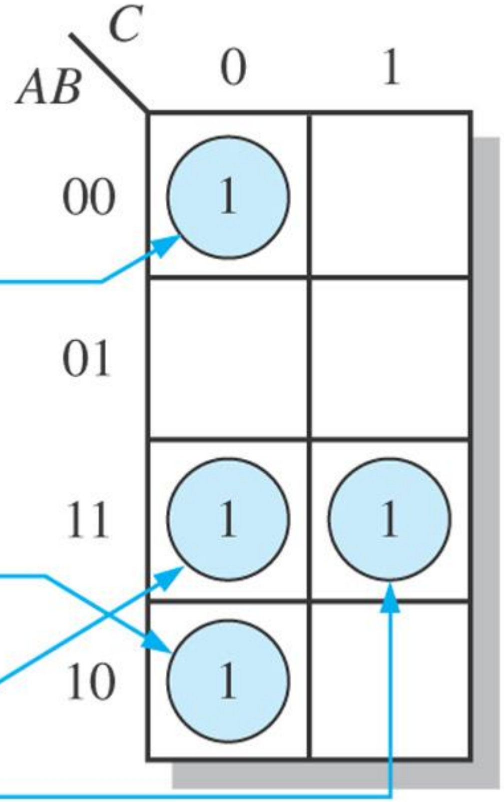
General View for Map-Entered Variable Method

- Given a map with variables P1, P2 etc, entered into some of the squares, the minimum SOP form of F is as follows:
- F = MS0 + P1 MS1 + P2MS2 + … where
  - MS0 is minimum sum obtained by setting P1 = P2 .. =0
  - MS1 is minimum sum obtained by setting P1 = 1, Pj = 0 ($j \neq 1$), and replacing all 1's on the map with don't cares.

  - Previously, G = A'B' + ACD + EA'D + FAD.

$$X = \overline{A}\,\overline{B}\,\overline{C} + A\overline{B}\,\overline{C} + AB\overline{C} + ABC$$

Inputs A	Inputs B	Inputs C	Output X
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1



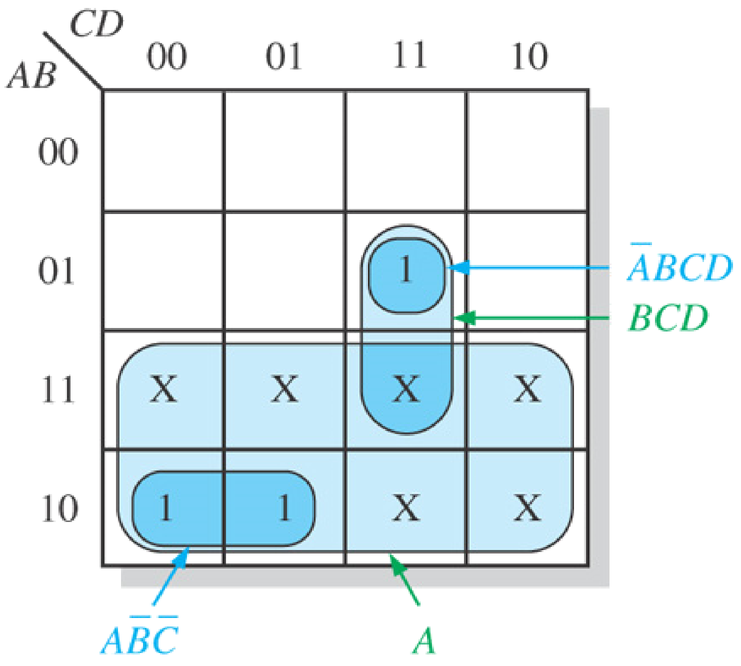
- Figure 4–35
  Example of mapping directly from a truth table to a Karnaugh map.

- Figure 4–36
  Example of the use of "don't care" conditions to simplify an expression.

Inputs $ABCD$	Output Y
0 0 0 0	0
0 0 0 1	0
0 0 1 0	0
0 0 1 1	0
0 1 0 0	0
0 1 0 1	0
0 1 1 0	0
0 1 1 1	1
1 0 0 0	1
1 0 0 1	1
1 0 1 0	X
1 0 1 1	X
1 1 0 0	X
1 1 0 1	X
1 1 1 0	X
1 1 1 1	X

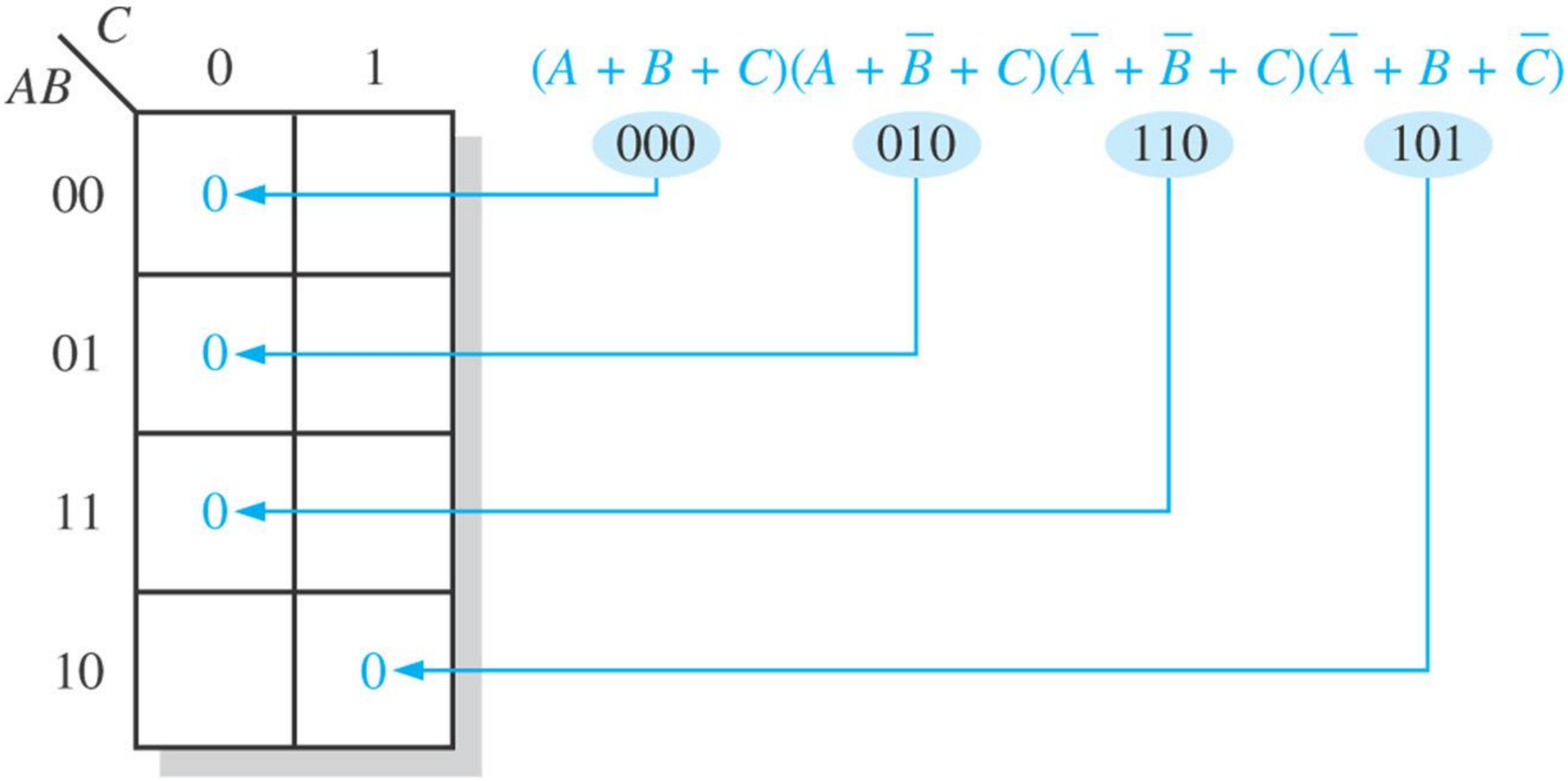
Don't cares

(a) Truth table



(b) Without "don't cares" $Y = A\overline{B}\overline{C} + \overline{A}BCD$
With "don't cares" $Y = A + BCD$

- Figure 4–37
  Example of mapping a standard POS expression.

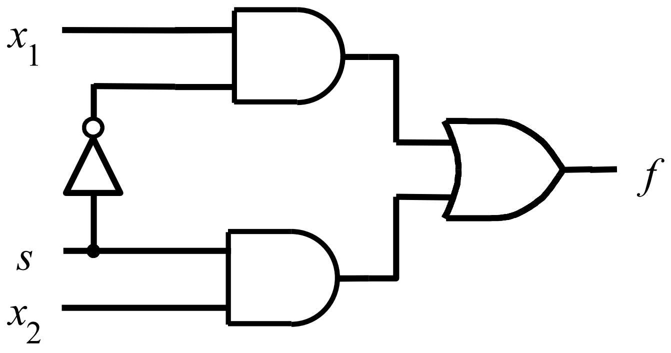


Jiangxi University of Science and Technology

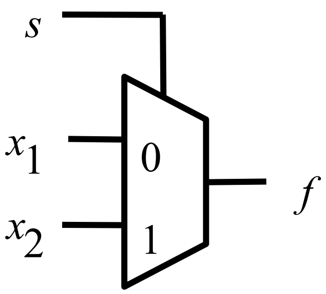
Minimization

Example: 2-1 Multiplexer (Definition)

- Has two inputs: x_1 and x_2
- Also has another input line s
- If s=0, then the output is equal to x_1
- If s=1, then the output is equal to x_2



(b) Circuit



(c) Graphical symbol

Truth Table for a 2-1 Multiplexer

Let's Draw the K-map

$s\ x_1\ x_2$	$f(s, x_1, x_2)$
0 0 0	0
0 0 1	0
0 1 0	1
0 1 1	1
1 0 0	0
1 0 1	1
1 1 0	0
1 1 1	1

	$s\ x_1\ x_2$	$f(s, x_1, x_2)$
m_0	0 0 0	0
m_1	0 0 1	0
m_2	0 1 0	1
m_3	0 1 1	1
m_4	1 0 0	0
m_5	1 0 1	1
m_6	1 1 0	0
m_7	1 1 1	1

$s \backslash x_1 x_2$	00	01	11	10
0	m_0	m_2	m_6	m_4
1	m_1	m_3	m_7	m_5

Let's Draw the K-map

	$s\ x_1\ x_2$	$f(s, x_1, x_2)$
m_0	0 0 0	0
m_1	0 0 1	0
m_2	0 1 0	1
m_3	0 1 1	1
m_4	1 0 0	0
m_5	1 0 1	1
m_6	1 1 0	0
m_7	1 1 1	1

$s \backslash x_1x_2$	00	01	11	10
0	0	1	0	0
1	0	1	1	1

Let's Draw the K-map

	$s\,x_1\,x_2$	$f(s, x_1, x_2)$
m_0	0 0 0	0
m_1	0 0 1	0
m_2	0 1 0	1
m_3	0 1 1	1
m_4	1 0 0	0
m_5	1 0 1	1
m_6	1 1 0	0
m_7	1 1 1	1

$s \backslash x_1x_2$	00	01	11	10
0	0	1	0	0
1	0	1	1	1

$$f(s, x_1, x_2) = \bar{x}_1 x_2 + s\,x_1$$

Something is wrong!

Jiangxi University of Science and Technology

Compare this with the SOP derivation

Let's Derive the SOP form

$s\ x_1\ x_2$	$f(s, x_1, x_2)$
0 0 0	0
0 0 1	0
0 1 0	$\boxed{1}$
0 1 1	$\boxed{1}$
1 0 0	0
1 0 1	$\boxed{1}$
1 1 0	0
1 1 1	$\boxed{1}$

Where should we put the negation signs?

$\overline{s}\, x_1\, \overline{x}_2$

$\overline{s}\, x_1\, x_2$

$s\, \overline{x}_1\, x_2$

$s\, x_1\, x_2$

$$f(s, x_1, x_2) = \overline{s}\, x_1\, \overline{x}_2 + \overline{s}\, x_1\, x_2 + s\, \overline{x}_1\, x_2 + s\, x_1\, x_2$$

Let's simplify this expression

$$f(s, x_1, x_2) = \bar{s}\, x_1\, \bar{x}_2 + \bar{s}\, x_1\, x_2 + s\, \bar{x}_1\, x_2 + s\, x_1\, x_2$$

$$f(s, x_1, x_2) = \bar{s}\, x_1\, (\bar{x}_2 + x_2) + s\, (\bar{x}_1 + x_1)\, x_2$$

$$f(s, x_1, x_2) = \bar{s}\, x_1 + s\, x_2$$

Let's Draw the K-map again

	$s\ x_1\ x_2$	$f(s, x_1, x_2)$
m_0	0 0 0	0
m_1	0 0 1	0
m_2	0 1 0	1
m_3	0 1 1	1
m_4	1 0 0	0
m_5	1 0 1	1
m_6	1 1 0	0
m_7	1 1 1	1

$x_2 \backslash s\ x_1$	00	01	11	10
0	m_0	m_2	m_6	m_4
1	m_1	m_3	m_7	m_5

The order of the labeling matters.

Let's Draw the K-map again

	$s\,x_1\,x_2$	$f(s, x_1, x_2)$
m_0	0 0 0	0
m_1	0 0 1	0
m_2	0 1 0	1
m_3	0 1 1	1
m_4	1 0 0	0
m_5	1 0 1	1
m_6	1 1 0	0
m_7	1 1 1	1

x_2 \ $s\,x_1$	00	01	11	10
0	0	1	0	0
1	0	1	1	1

$$f(s, x_1, x_2) = \overline{s}\,x_1 + s\,x_2$$

This is correct!

Two Different Ways to Draw the K-map

x_1	x_2	x_3	
0	0	0	m_0
0	0	1	m_1
0	1	0	m_2
0	1	1	m_3
1	0	0	m_4
1	0	1	m_5
1	1	0	m_6
1	1	1	m_7

(a) Truth table

x_3 \ x_1x_2	00	01	11	10
0	m_0	m_2	m_6	m_4
1	m_1	m_3	m_7	m_5

(b) Karnaugh map

x_1 \ x_2x_3	00	01	11	10
0	m_0	m_1	m_3	m_2
1	m_4	m_5	m_7	m_6

Another Way to Draw 3-variable K-map

x_1	x_2	x_3	
0	0	0	m_0
0	0	1	m_1
0	1	0	m_2
0	1	1	m_3
1	0	0	m_4
1	0	1	m_5
1	1	0	m_6
1	1	1	m_7

(a) Truth table

x_3 \ x_1x_2	00	01	11	10
0	m_0	m_2	m_6	m_4
1	m_1	m_3	m_7	m_5

(b) Karnaugh map

x_2x_3 \ x_1	0	1
00	m_0	m_4
01	m_1	m_5
11	m_3	m_7
10	m_2	m_6

Gray Code

- Sequence of binary codes
- Neighboring lines vary by only 1 bit

00	000
01	001
11	011
10	010
	110
	111
	101
	100

Gray Code & K-map

	$s\ x_1\ x_2$
m_0	0 0 0
m_1	0 0 1
m_2	0 1 0
m_3	0 1 1
m_4	1 0 0
m_5	1 0 1
m_6	1 1 0
m_7	1 1 1

x_2 \ $s\ x_1$	00	01	11	10
0	m_0	m_2	m_6	m_4
1	m_1	m_3	m_7	m_5

x_2 \ $s\ x_1$	00	01	11	10
0	000	010	110	100
1	001	011	111	101

Gray Code & K-map

	$s\,x_1\,x_2$
m_0	0 0 0
m_1	0 0 1
m_2	0 1 0
m_3	0 1 1
m_4	1 0 0
m_5	1 0 1
m_6	1 1 0
m_7	1 1 1

x_2 \ $s\,x_1$	00	01	11	10
0	000	010	110	100
1	001	011	111	101

These two neighbors
differ only in the LAST bit

Gray Code & K-map

	$s\ x_1\ x_2$
m_0	0 0 0
m_1	0 0 1
m_2	0 1 0
m_3	0 1 1
m_4	1 0 0
m_5	1 0 1
m_6	1 1 0
m_7	1 1 1

x_2 \ $s\ x_1$	00	01	11	10
0	000	010	110	100
1	001	011	111	101

These two neighbors
differ only in the LAST bit

Gray Code & K-map

	$s\ x_1\ x_2$
m_0	0 0 0
m_1	0 0 1
m_2	0 1 0
m_3	0 1 1
m_4	1 0 0
m_5	1 0 1
m_6	1 1 0
m_7	1 1 1

x_2 \ $s\ x_1$	00	01	11	10
0	000	010	110	100
1	001	011	111	101

These two neighbors
differ only in the FIRST bit

Gray Code & K-map

	$s\ x_1\ x_2$
m_0	0 0 0
m_1	0 0 1
m_2	0 1 0
m_3	0 1 1
m_4	1 0 0
m_5	1 0 1
m_6	1 1 0
m_7	1 1 1

x_2 \ $s\ x_1$	00	01	11	10
0	000	010	110	100
1	001	011	111	101

These two neighbors
differ only in the FIRST bit

Adjacency Rules

x_2 \ $s\,x_1$	00	01	11	10
0	000	010	110	100
1	001	011	111	101

adjacent
columns

Gray Code & K-map

	$s\ x_1\ x_2$
m_0	0 0 0
m_1	0 0 1
m_2	0 1 0
m_3	0 1 1
m_4	1 0 0
m_5	1 0 1
m_6	1 1 0
m_7	1 1 1

x_2 \ $s\ x_1$	00	01	11	10
0	000	010	110	100
1	001	011	111	101

These four neighbors
differ in the FIRST and LAST bit

They are similar in their MIDDLE bit

Gray Code & K-map

	$s\ x_1\ x_2$
m_0	0 0 0
m_1	0 0 1
m_2	0 1 0
m_3	0 1 1
m_4	1 0 0
m_5	1 0 1
m_6	1 1 0
m_7	1 1 1

x_2 \ $s\ x_1$	00	01	11	10
0	000	010	110	100
1	001	011	111	101

These four neighbors
differ in the FIRST and LAST bit

They are similar in their MIDDLE bit

A four-variable Karnaugh map

x_3x_4 \ x_1x_2	00	01	11	10
00	m_0	m_4	m_{12}	m_8
01	m_1	m_5	m_{13}	m_9
11	m_3	m_7	m_{15}	m_{11}
10	m_2	m_6	m_{14}	m_{10}

x_1

x_2

x_3

x_4

[Figure 2.53 from the textbook]

A four-variable Karnaugh map

x1	x2	x3	x4	
0	0	0	0	m0
0	0	0	1	m1
0	0	1	0	m2
0	0	1	1	m3
0	1	0	0	m4
0	1	0	1	m5
0	1	1	0	m6
0	1	1	1	m7
1	0	0	0	m8
1	0	0	1	m9
1	0	1	0	m10
1	0	1	1	m11
1	1	0	0	m12
1	1	0	1	m13
1	1	1	0	m14
1	1	1	1	m15

x_3x_4 \ x_1x_2	00	01	11	10
00	m_0	m_4	m_{12}	m_8
01	m_1	m_5	m_{13}	m_9
11	m_3	m_7	m_{15}	m_{11}
10	m_2	m_6	m_{14}	m_{10}

x_1 x_2 x_3 x_4

Adjacency Rules

x_3 \ x_1x_2	00	01	11	10
0	m_0	m_2	m_6	m_4
1	m_1	m_3	m_7	m_5

adjacent columns

x_3x_4 \ x_1x_2	00	01	11	10
00	m_0	m_4	m_{12}	m_8
01	m_1	m_5	m_{13}	m_9
11	m_3	m_7	m_{15}	m_{11}
10	m_2	m_6	m_{14}	m_{10}

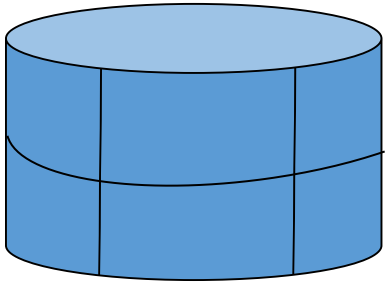
adjacent rows

adjacent columns

Adjacency Rules

$x_3 \backslash x_1x_2$	00	01	11	10
0	m_0	m_2	m_6	m_4
1	m_1	m_3	m_7	m_5

adjacent columns

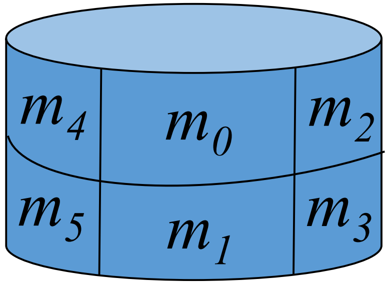


As if the K-map were drawn on a cylinder

Adjacency Rules

$x_3 \backslash x_1x_2$	00	01	11	10
0	m_0	m_2	m_6	m_4
1	m_1	m_3	m_7	m_5

adjacent
columns



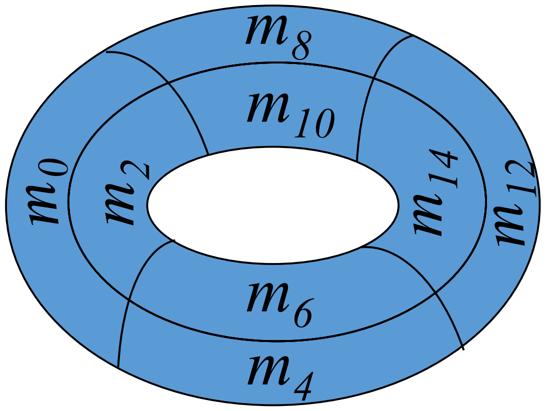
As if the K-map were
drawn on a cylinder

Adjacency Rules

x_3x_4 \ x_1x_2	00	01	11	10
00	m_0	m_4	m_{12}	m_8
01	m_1	m_5	m_{13}	m_9
11	m_3	m_7	m_{15}	m_{11}
10	m_2	m_6	m_{14}	m_{10}

adjacent rows

adjacent columns



As if the K-map were drawn on a torus

Gray Code & K-map

x1	x2	x3	x4	
0	0	0	0	m0
0	0	0	1	m1
0	0	1	0	m2
0	0	1	1	m3
0	1	0	0	m4
0	1	0	1	m5
0	1	1	0	m6
0	1	1	1	m7
1	0	0	0	m8
1	0	0	1	m9
1	0	1	0	m10
1	0	1	1	m11
1	1	0	0	m12
1	1	0	1	m13
1	1	1	0	m14
1	1	1	1	m15

x_3x_4 \ x_1x_2	00	01	11	10
00	m_0	m_4	m_{12}	m_8
01	m_1	m_5	m_{13}	m_9
11	m_3	m_7	m_{15}	m_{11}
10	m_2	m_6	m_{14}	m_{10}

x_1 x_2 x_3 x_4

Gray Code & K-map

x1	x2	x3	x4	
0	0	0	0	m0
0	0	0	1	m1
0	0	1	0	m2
0	0	1	1	m3
0	1	0	0	m4
0	1	0	1	m5
0	1	1	0	m6
0	1	1	1	m7
1	0	0	0	m8
1	0	0	1	m9
1	0	1	0	m10
1	0	1	1	m11
1	1	0	0	m12
1	1	0	1	m13
1	1	1	0	m14
1	1	1	1	m15

x_1

x_3x_4 \ x_1x_2		00	01	11	10
0000	0100	1100	1000	m_{12}	m_8
0001	0101	1101	1001	m_{13}	m_9
0011	0111	1111	1011	m_{15}	m_{11}
0010	0110	1110	1010	m_{14}	m_{10}

x_3

x_4

x_2

Example of a four-variable Karnaugh map

$x_3x_4 \backslash x_1x_2$	00	01	11	10
00	0	0	0	0
01	0	0	1	1
11	1	0	0	1
10	1	0	0	1

$$f_1 = \bar{x}_2 x_3 + x_1 \bar{x}_3 x_4$$

$x_3x_4 \backslash x_1x_2$	00	01	11	10
00	0	0	0	0
01	0	0	1	1
11	1	1	1	1
10	1	1	1	1

$$f_2 = x_3 + x_1 x_4$$

Jiangxi University of Science and Technology

Strategy For Minimization

Grouping Rules

- Group "1"s with rectangles
- Both sides a power of 2:
  - 1x1, 1x2, 2x1, 2x2, 1x4, 4x1, 2x4, 4x2, 4x4
- Can use the same minterm more than once
- Can wrap around the edges of the map
- Some rules in selecting groups:
  - Try to use as few groups as possible to cover all "1"s.
  - For each group, try to make it as large as you can (i.e., if you can use a 2x2, don't use a 2x1 even if that is enough).

Terminology

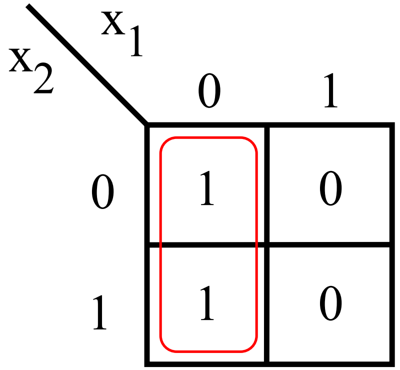
Literal: a variable, complemented or uncomplemented

Some Examples:

- X_1
- X_2

Terminology

- Implicant: product term that indicates the input combinations for which the function output is 1
- Example



- $\bar{x}_1$ - indicates that $\bar{x}_1\bar{x}_2$ and $\bar{x}_1x_2$ yield output of 1

Terminology

- Prime Implicant
  - Implicant that cannot be combined into another implicant with fewer literals

  - Some Examples

$x_3 \backslash x_1x_2$	00	01	11	10
0	0	1	1	1
1	1	1	1	0

Not prime

$x_3 \backslash x_1x_2$	00	01	11	10
0	0	1	1	1
1	1	1	1	0

Prime

Terminology

- Essential Prime Implicant
  - Prime implicant that includes a minterm not covered by any other prime implicant

  - Some Examples

$x_3 \backslash x_1x_2$	00	01	11	10
0	0	1	1	1
1	1	1	0	0

Terminology

- Cover
  - Collection of implicants that account for all possible input valuations where output is 1

- Ex. $x_1' x_2 x_3 + x_1 x_2 x_3' + x_1 x_2' x_3'$
- Ex. $x_1' x_2 x_3 + x_1 x_3'$

$x_3 \backslash x_1x_2$	00	01	11	10
0	0	0	1	1
1	0	1	0	0

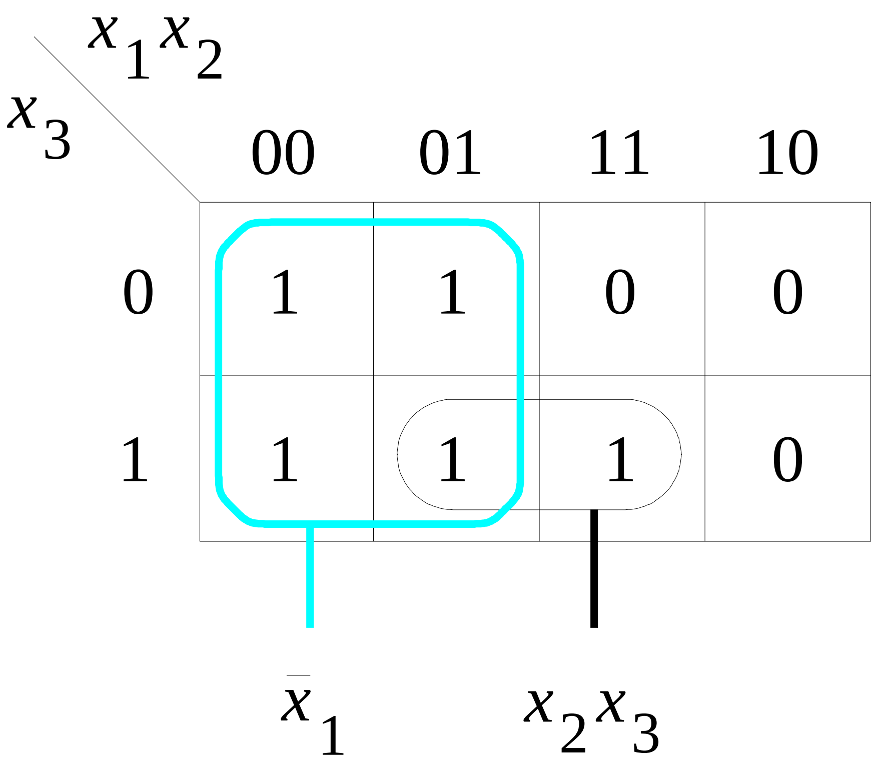
Example

- Give the Number of
  - Implicants?
  - Prime Implicants?
  - Essential Prime Implicants?

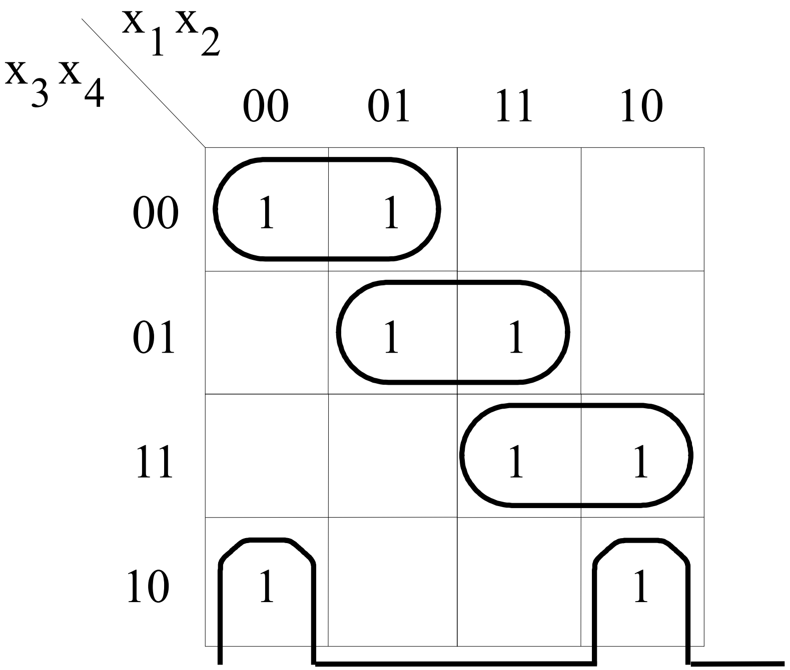
$x_3 \backslash x_1x_2$	00	01	11	10
0	1	1	0	0
1	1	1	1	0

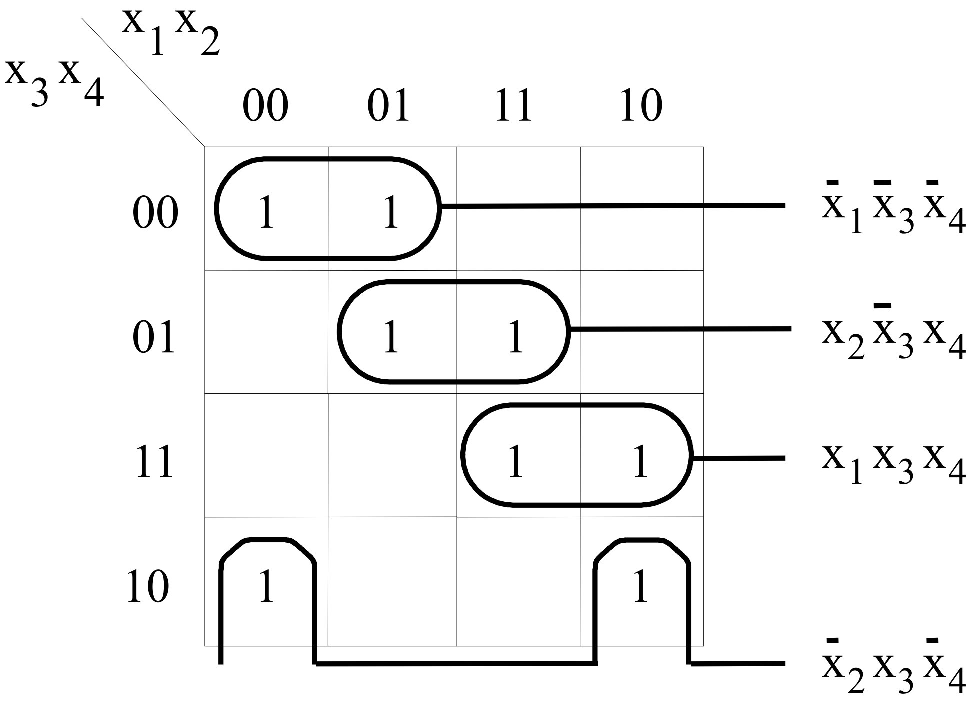
Example

Three-variable function $f(x_1, x_2, x_3) = \sum m(0, 1, 2, 3, 7)$



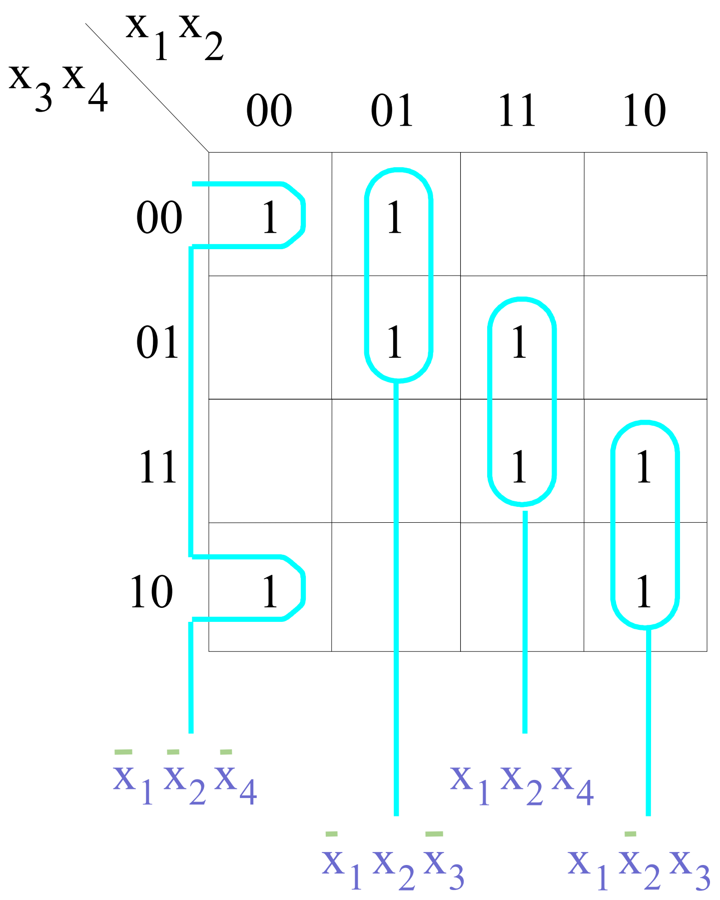
Example





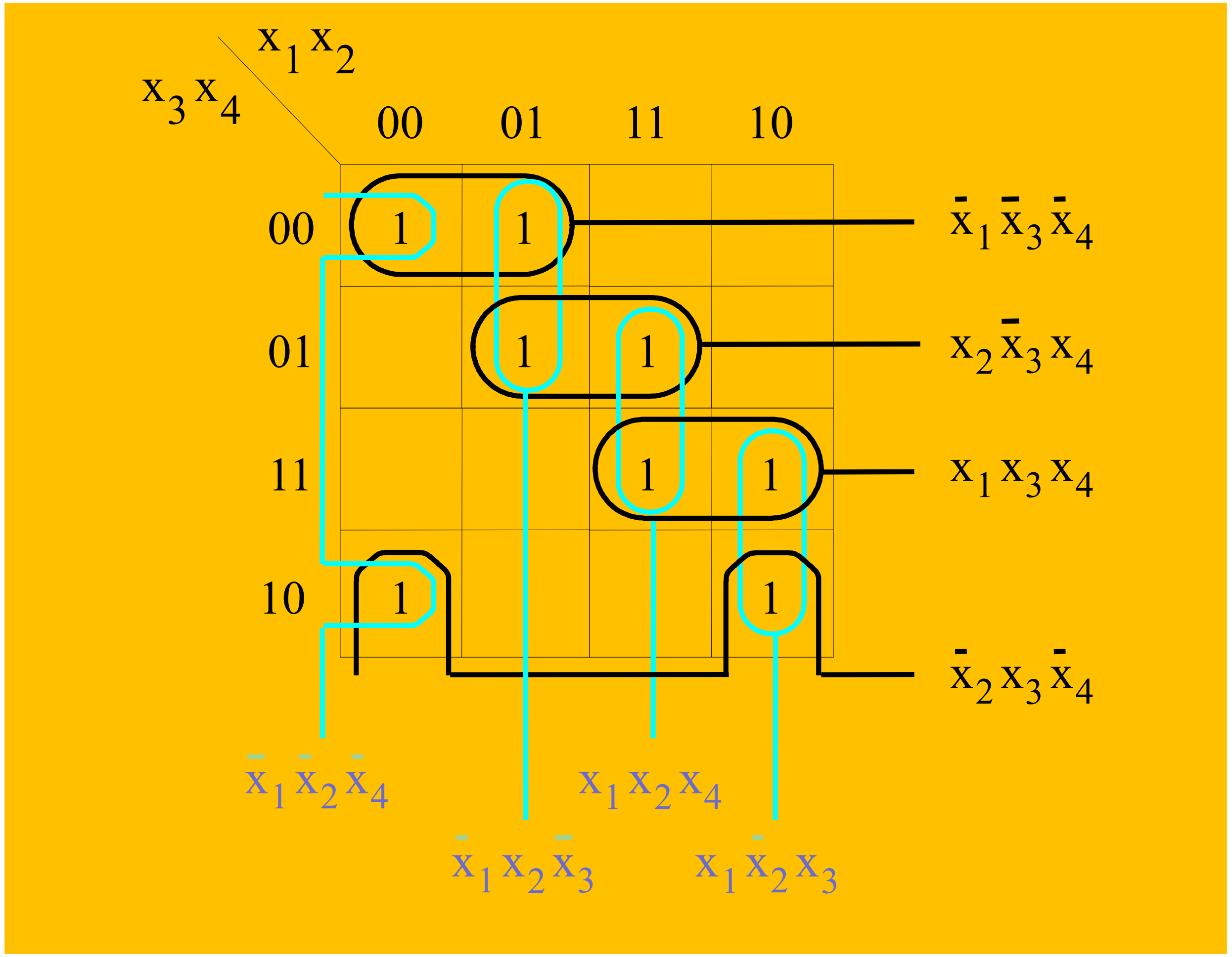
$$f = \bar{x}_1\bar{x}_3\bar{x}_4 + x_2\bar{x}_3x_4 + x_1x_3x_4 + \bar{x}_2x_3\bar{x}_4$$

Example: Another Solution

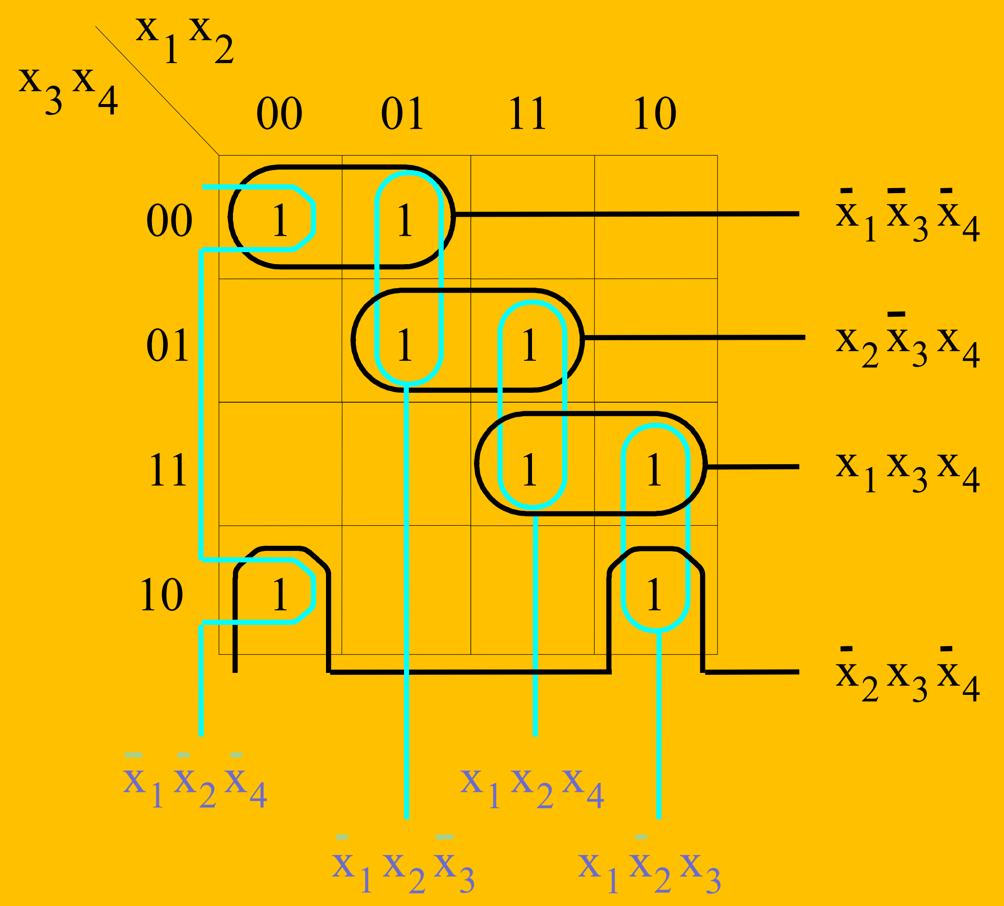


$$f = \bar{x}_1 \bar{x}_2 \bar{x}_4 + \bar{x}_1 x_2 \bar{x}_3 + x_1 x_2 x_4 + x_1 \bar{x}_2 x_3$$

Example: Both Are Valid Solutions



Example: Both Are Valid Solutions



$$f = \bar{x}_1 \bar{x}_3 \bar{x}_4 + x_2 \bar{x}_3 x_4 + x_1 x_3 x_4 + \bar{x}_2 x_3 \bar{x}_4$$

$$f = \bar{x}_1 \bar{x}_2 \bar{x}_4 + \bar{x}_1 x_2 \bar{x}_3 + x_1 x_2 x_4 + x_1 \bar{x}_2 x_3$$

Minimization of Product-of-Sums Forms

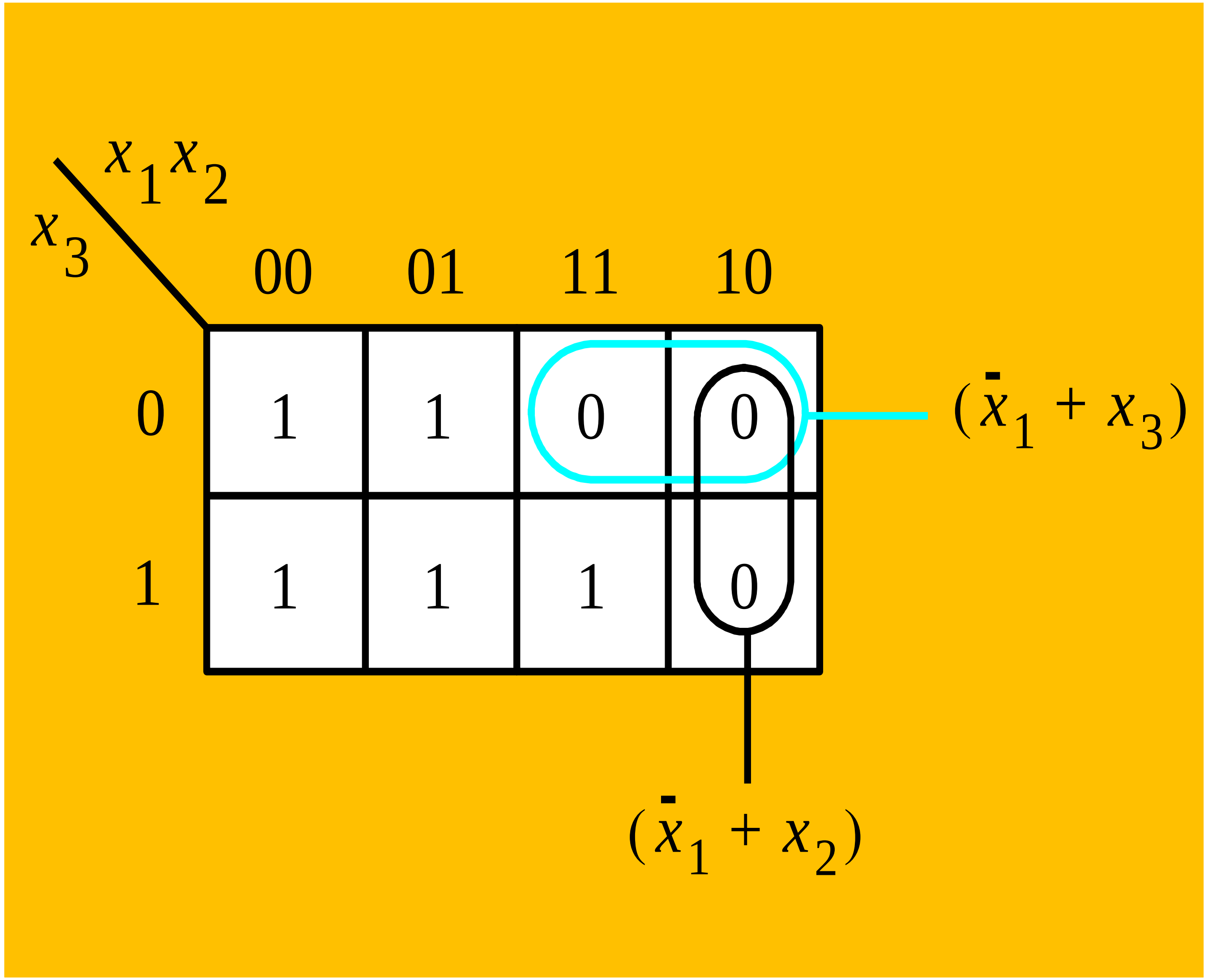
Do You Still Remember This Boolean Algebra Theorem?

14a. $x \cdot y + x \cdot \overline{y} = x$ **Combining**

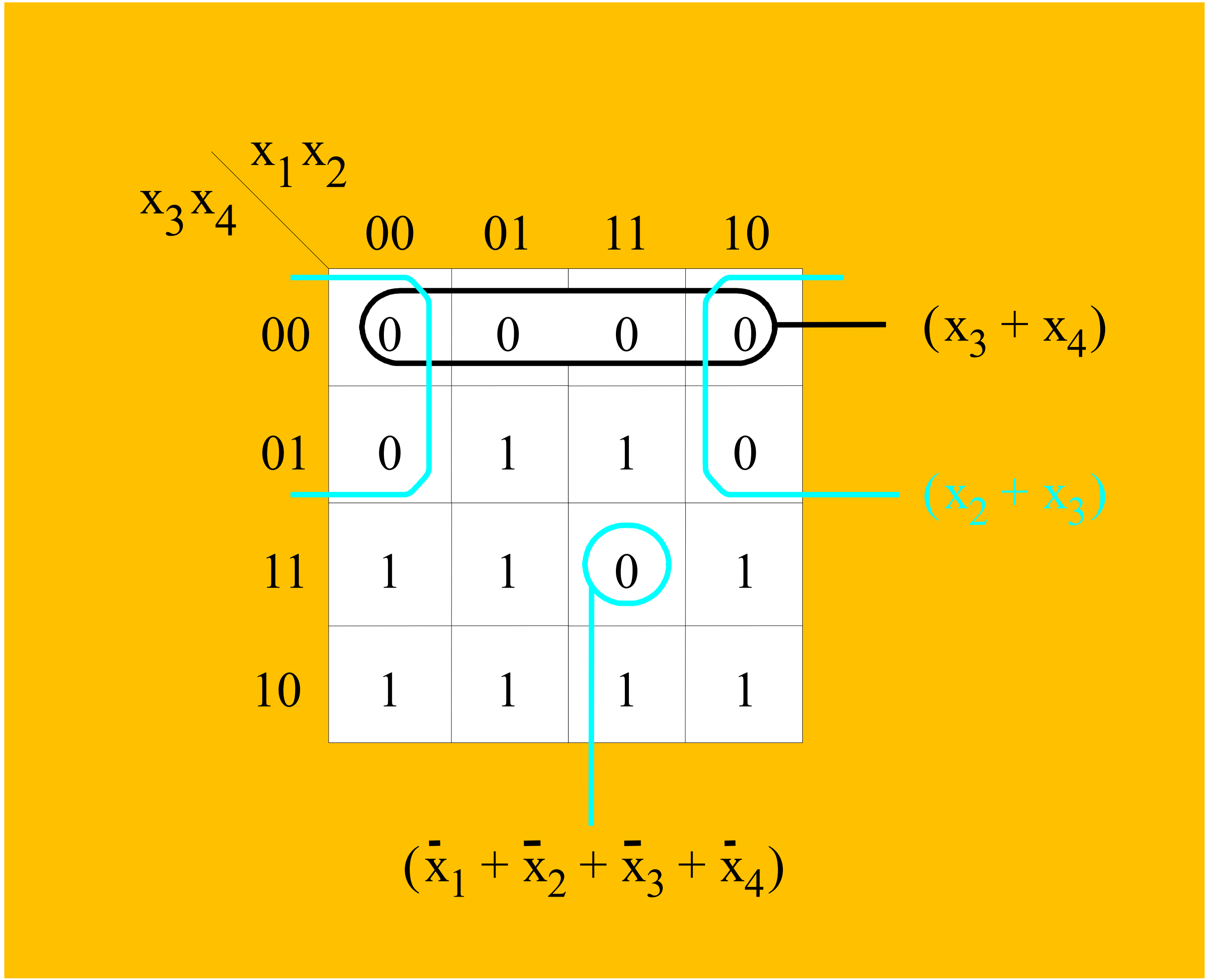
14b. $(x + y)\cdot(x + \overline{y}) = x$

江西理工大学 JIANGXI UNIVERSITY OF SCIENCE AND TECHNOLOGY

POS minimization of f (x1, x2, x3) = $\prod$ M(4, 5, 6)



POS minimization of f (x1,…, x4) = ∏ M(0, 1, 4, 8, 9, 12, 15)



Reference

- http://www.ee.surrey.ac.uk/Projects/CAL/digital-logic/gatesfunc/
- *CprE 281: Digital Logic,Iowa State University, Ames, IA,Copyright © Alexander Stoytchev*
- **Digital Systems: Principles and Applications, 11/e**
- *Ronald J. Tocci, Neal S. Widmer, Gregory L. Moss*