Lecture 1 Outline

Reminders to self:

- ❑ Turn on lecture recording to Cloud
- ❑ Turn on Zoom microphone

- Last Lecture
  - More Boolean Algebra
  - Introduction of Design of Switching Circuits
- Today's Lecture
  - Foundations for K-Maps
    - Minterms
    - Maxterms
  - Start K-Maps

Handouts and Announcements

- Announcements
  - Homework Problems 2-4, 4-1
    - Posted on Carmen this morning
    - Due in Carmen 11:59pm, Thursday 2/2
  - Homework Problems 2-2 and 2-3 reminder
    - HW 2-2 due: 11:25am Monday 1/30
    - HW 2-3 due: 11:59pm Tuesday 1/31
  - Read for Monday: Pages 108-114, 144-149

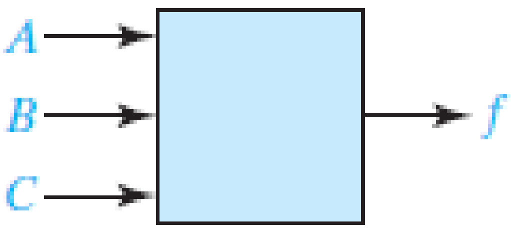
Minterms

- A minterm of n variables is a product of n literals in which each variable
  - appears exactly once
  - in either true or complemented form,
  - but not both

Row No.	A B C	Minterms
0	0 0 0	$A'B'C' = \boxed{m_0}$
1	0 0 1	$A'B'C = m_1$
2	0 1 0	$A'BC' = m_2$
3	0 1 1	$A'BC = m_3$
4	1 0 0	$AB'C' = m_4$
5	1 0 1	$AB'C = m_5$
6	1 1 0	$ABC' = m_6$
7	1 1 1	$ABC = m_7$

- Abbreviations for minterms for $n=3$
- In general, the minterm which corresponds to row i of the truth table is designated m_i
- Index i is usually written in decimal

Minterms



(a)

A	B	C	f	f'
0	0	0	0	1
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	1	0
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

(b)

When a function f is written as a sum of minterms, this is referred to as a

- minterm expansion or
- standard Sum Of Products (SOP)

$$f = A'BC + AB'C' + AB'C + ABC' + ABC$$

$$f(A,B,C) =$$

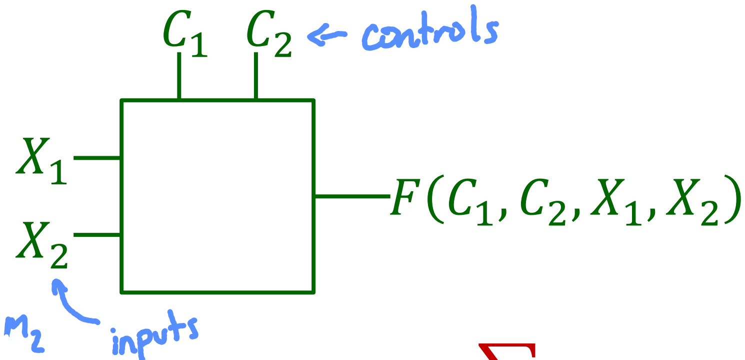
$$f(A,B,C) = \sum m(\qquad)$$

Note that the function is expressed here as a SOP that describes when the function = 1

Minterms

Example:

m_i	C_1	C_2	X_1	X_2	F	
m_0	0	0	0	0	0	
m_1	0	0	0	1	0	
m_2	0	0	1	0	1	m_2
m_3	0	0	1	1	1	m_3
m_4	0	1	0	0	1	m_4
m_5	0	1	0	1	1	m_5
m_6	0	1	1	0	1	m_6
m_7	0	1	1	1	1	m_7
m_8	1	0	0	0	0	
m_9	1	0	0	1	0	
m_{10}	1	0	1	0	0	
m_{11}	1	0	1	1	0	
m_{12}	1	1	0	0	0	
m_{13}	1	1	0	1	1	m_{13}
m_{14}	1	1	1	0	0	
m_{15}	1	1	1	1	1	m_{15}



If $C_1 = 0$ and $C_2 = 0$ $F = X_1$
If $C_1 = 1$ and $C_2 = 1$ $F = X_2$
If $C_1 = 0$ and $C_2 = 1$ $F = 1$
If $C_1 = 1$ and $C_2 = 0$ $F = 0$

$$F(C_1, C_2, X_1, X_2) = \sum m(2, 3, 4, 5, 6, 7, 13, 15)$$

Direct implementation:

- Eight 4-input AND
- One 8-input OR

$$F = \boxed{C_1'C_2'X_1X_2' + C_1'C_2'X_1X_2} + \boxed{C_1'C_2X_1'X_2' + C_1'C_2X_1'X_2 + C_1'C_2X_1X_2' + C_1'C_2X_1X_2} + \boxed{C_1C_2X_1'X_2 + C_1C_2X_1X_2}$$

$C_1'C_2'X_1$

$$C_1'C_2(\overbrace{X_1'X_2' + X_1'X_2 + X_1X_2' + X_1X_2}^{=1}) = C_1'C_2$$

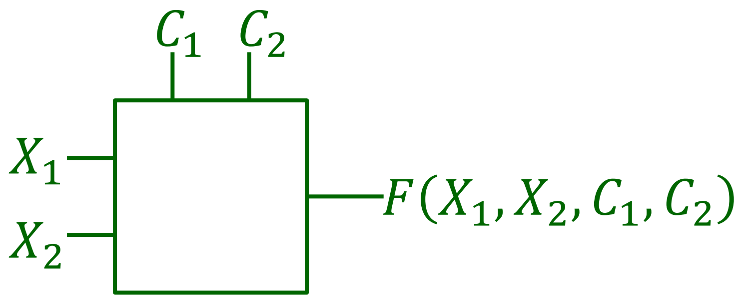
$C_1C_2X_2$

$$F = C_1'C_2'X_1 + C_1'C_2 + C_1C_2X_2$$

Minterms

Example:

m_i	C_1	C_2	X_1	X_2	F	
m_0	0	0	0	0	0	
m_1	0	0	0	1	0	
m_2	0	0	1	0	1	m_2
m_3	0	0	1	1	1	m_3
m_4	0	1	0	0	1	m_4
m_5	0	1	0	1	1	m_5
m_6	0	1	1	0	1	m_6
m_7	0	1	1	1	1	m_7
m_8	1	0	0	0	0	
m_9	1	0	0	1	0	
m_{10}	1	0	1	0	0	
m_{11}	1	0	1	1	0	
m_{12}	1	1	0	0	0	
m_{13}	1	1	0	1	1	m_{13}
m_{14}	1	1	1	0	0	
m_{15}	1	1	1	1	1	m_{15}



If $C_1 = 0$ and $C_2 = 0$ $F = X_1$ ✓
If $C_1 = 1$ and $C_2 = 1$ $F = X_2$ ✓
If $C_1 = 0$ and $C_2 = 1$ $F = 1$ ✓
If $C_1 = 1$ and $C_2 = 0$ $F = 0$ ✓

$$F = C_1'C_2'X_1 + C_1'C_2 + C_1C_2X_2$$

$$C_1'(C_2 + C_2'X_1) = C_1'(C_2 + C_2')(C_2 + X_1)$$

$$= C_1'(C_2C_2 + C_2C_2' + C_2X_1 + C_2'X_1)$$

Elimination theorem (pg 47)

$$= C_1'(C_2 + X_1)$$

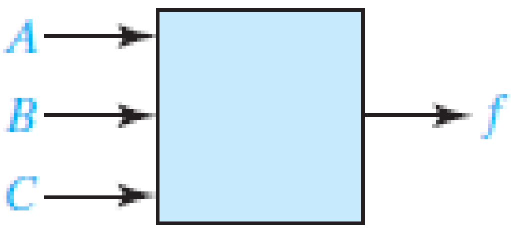
$$F = \boxed{C_1'C_2} + C_1'X_1 + \boxed{C_1C_2X_2}$$

$$= C_2(C_1' + C_1X_2) + C_1'X_1 = C_2(C_1' + X_2) + C_1'X_1$$

$$\boxed{F = C_1'C_2 + C_1'X_1 + C_2X_2}$$

Simplified implementation:
- Three 2-input AND
- One 3-input OR

Maxterms



(a)

A	B	C	f	f′
0	0	0	0	1
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	1	0
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

Could write f based on the 0's of the function

$$f = (A + B + C)(A + B + C')(A + B' + C)$$

While minterms are based on the SOP form, maxterms are based on the Product Of Sums form ( )

$$f(A,B,C) = \qquad\qquad = \prod M(\qquad)$$

Maxterms

- A maxterm of n variables is a of n literals in which each variable
  - appears exactly once
  - in either true or complemented form,
  - but not both
- A maxterm is the of the corresponding minterm (DeMorgan's; e.g. Row #4)

Row No.	$A\ B\ C$	Minterms	Maxterms
0	0 0 0	$A'B'C' = m_0$	$A + B + C = M_0$
1	0 0 1	$A'B'C = m_1$	$A + B + C' = M_1$
2	0 1 0	$A'BC' = m_2$	$A + B' + C = M_2$
3	0 1 1	$A'BC = m_3$	$A + B' + C' = M_3$
4	1 0 0	$AB'C' = m_4$	$A' + B + C = M_4$
5	1 0 1	$AB'C = m_5$	$A' + B + C' = M_5$
6	1 1 0	$ABC' = m_6$	$A' + B' + C = M_6$
7	1 1 1	$ABC = m_7$	$A' + B' + C' = M_7$

- Abbreviations for maxterms for
- In general, the maxterm which corresponds to row i of the truth table is designated M_i
- Index i is usually written in

General Minterems & Maxterms

DESIRED FORM

GIVEN FORM	Minterm Expansion of F	Maxterm Expansion of F	Minterm Expansion of F'	Maxterm Expansion of F'
Minterm Expansion of F	———	maxterm nos. are those nos. not on the minterm list for F	list minterms not present in F	maxterm nos. are the same as minterm nos. of F
Maxterm Expansion of F	minterm nos. are those nos. not on the maxterm list for F	———	minterm nos. are the same as maxterm nos. of F	list maxterms not present in F

General Minterems & Maxterms

For our previous function

DESIRED FORM

GIVEN FORM	Minterm Expansion of f	Maxterm Expansion of f	Minterm Expansion of f'	Maxterm Expansion of f'
$f = \Sigma\, m(3, 4, 5, 6, 7)$	__________	$\Pi\, M(0, 1, 2)$	$\Sigma\, m(0, 1, 2)$	$\Pi\, M(3, 4, 5, 6, 7)$
$f = \Pi\, M(0, 1, 2)$	$\Sigma\, m(3, 4, 5, 6, 7)$	__________	$\Sigma\, m(0, 1, 2)$	$\Pi\, M(3, 4, 5, 6, 7)$

Order of Variables

The order of the variables in a function name matters

- $F(A,B,C) = \sum m(2,4) = A'BC' + AB'C'$
- $G(\quad\quad) = \sum m(2,4) =$

m_2 is symmetric for the orders of variables in F and G

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

m_i		B		G
m_0	0	0	0	0
m_1	0	0	1	0
m_2	0	1	0	
m_3	0	1	1	0
m_4	1	0	0	
m_5	1	0	1	0
m_6	1	1	0	0
m_7	1	1	1	0

Incompletely Specified Functions

- In some systems, certain combinations of inputs will never occur
- Those combinations of
  - might not be allowed, or
  - might never occur,
  - or the output might be used in such a way that we don't care what it is for some combinations of inputs
- For these combinations, we "don't care" what the value of F is
- The function F is then considered incompletely specified
- Truth table with don't-cares:
- In minterm and maxterm expansions respectively d, and D specify "don't cares"

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

$$F = \sum m(\qquad) + \sum d(\qquad)$$

$$F = \prod M(\qquad) \cdot \prod D(\qquad)$$

Incompletely Specified Functions

- Don’t cares may allow you to simplify a logic function
  - Can count it as a 1 in the truth table ( ), if that will help reduce the expression
  - Or can ignore it (including is optional)
- We will see more about how to do this in K-maps

Summary of where we are

1. Build a truth table from a “Design Statement”
   - Such as example from last lecture
   - List all possible input combinations and their outputs
2. Minterm expansion to write SOP logic expression for the system
3. SOP generates a maximum of two stages of logic
   - First stage: AND ( )
   - Second stage: OR ( )
   - Assumes all variables and their complements are available
   - Might need a “zero” stage of inverters to generate complements if they are not already available

Summary of where we are

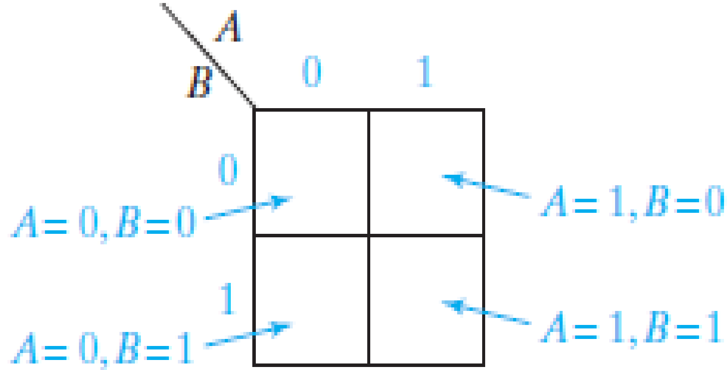
4. "Don't Care" inputs are combinations that are not allowed or are impossible
   - Treat as "Wild Cards"
   - ***Can*** be used to simplify expression
5. Maxterm expansion POS also yields two stages of logic
   - First stage: OR ( )
   - Second stage: AND ( )
   - Assumes all variables and their complements are available
   - Might need a "zero" stage of inverters to generate complements if they are not already available

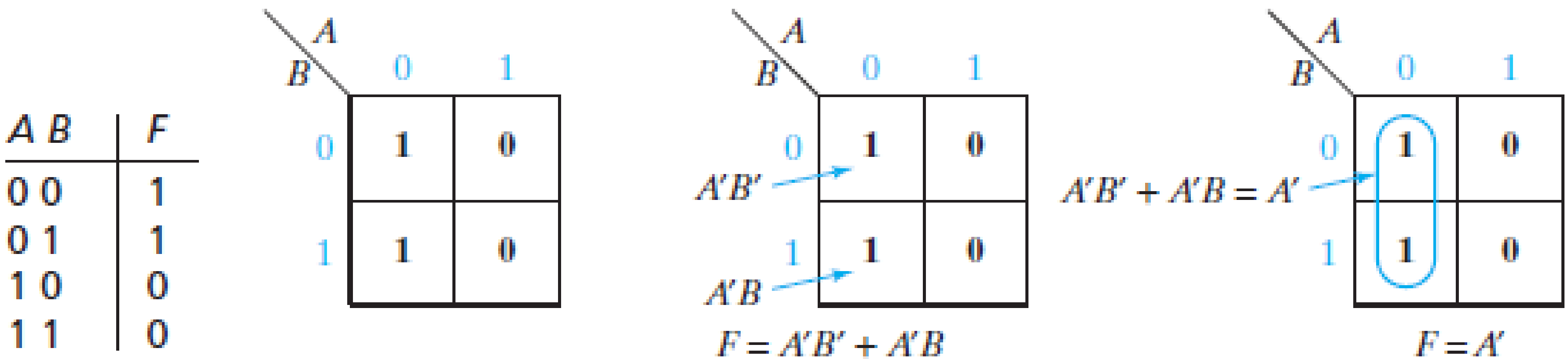
Karnaugh Maps

- Karnaugh Maps ( ) provide a visual way to reduce logic ( )
- Plot minterms next to each other so that each term differs by only 1 variable from the minterms around it
- Example: $AB'C + ABC = AC(\qquad) = AC(\quad) = AC$
- K-map will guarantee (2-stage logic):
  - Fewest # of gates
  - Fewest # variables per gate
- K-map does not guarantee a unique solution

2-Variable Karnaugh Maps

- Two-variable K-Map
- Fill the map with 1s and 0s consistent with the truth table





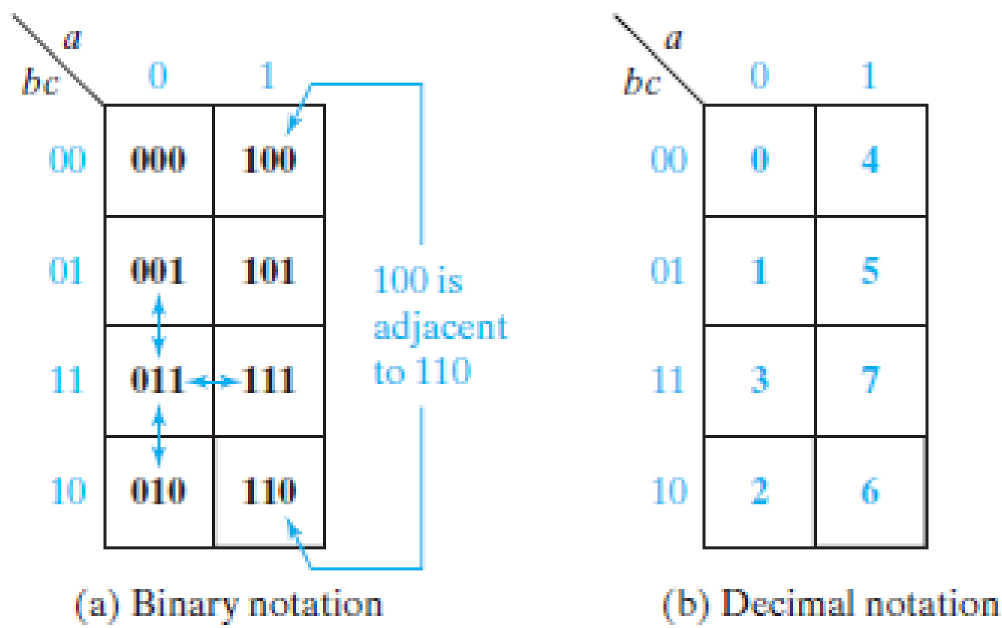
(a) (b) (c) (d)

- Minterms in adjacent squares can be combined since they differ by only one variable

3-Variable Karnaugh Maps

- Three-variable K-Map plotted in a similar manner
- The value of one variable, *a*, is listed on the top and the values of the other two, *b* and *c*, are listed on the side
  - Order of variables in the minterms shown here is *abc*
  - Note: alternative layout with one variable on the side and two on the top is equally valid

FIGURE 5-3 Location of Minterms on a Three-Variable Karnaugh Map



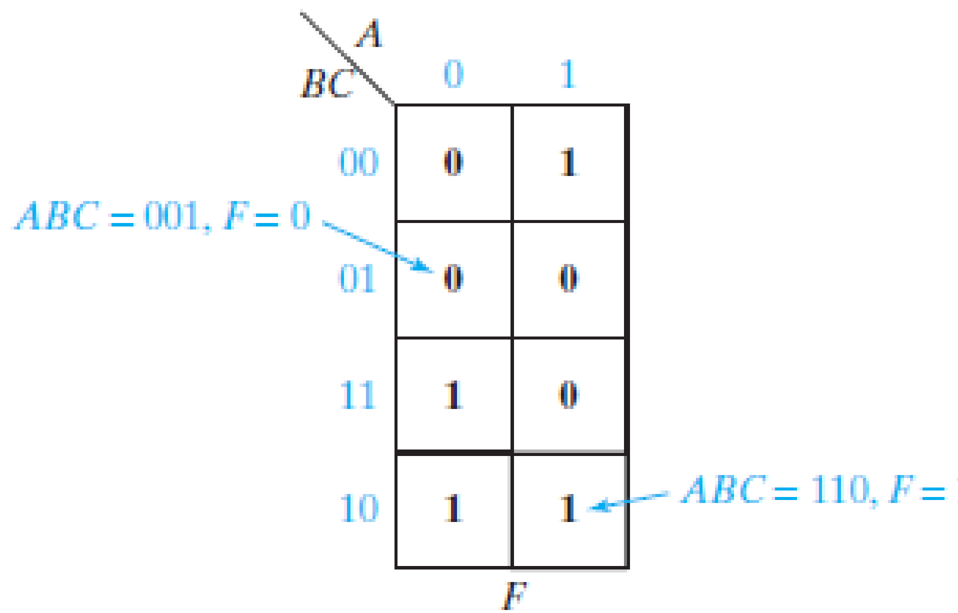
- Note "Gray-code-like" order
- Only one variable changes from row-to-row

3-Variable Karnaugh Maps

- Minterms in adjacent squares of map differ in only one variable and therefore can be combined using uniting theorem $XY + XY' = X$
- Do this as an example. Formal algorithm next lecture

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

(a)



(b)

4-Variable Karnaugh Maps

A	B	C	D	F
0	0	0	0	m_0
0	0	0	1	m_1
0	0	1	0	m_2
0	0	1	1	m_3
0	1	0	0	m_4
0	1	0	1	m_5
0	1	1	0	m_6
0	1	1	1	m_7
1	0	0	0	m_8
1	0	0	1	m_9
1	0	1	0	m_{10}
1	0	1	1	m_{11}
1	1	0	0	m_{12}
1	1	0	1	m_{13}
1	1	1	0	m_{14}
1	1	1	1	m_{15}

Map with AB on and CD on

Remember that the map wraps

- Left edge is adjacent to right edge (   )
- Top edge is adjacent to bottom edge (   )

$B = 1$

$CD \backslash AB$	00	01	11	10
00				
01				
11				
10				

$D = 1$

$C = 1$

$A = 1$

$AB \backslash CD$	00	01	11	10
00	m_0	m_1	m_3	m_2
01	m_4	m_5	m_7	m_6
11	m_{12}	m_{13}	m_{15}	m_{14}
10	m_8	m_9	m_{11}	m_{10}

Map with AB on and CD on

K-Map Minimal SOP Algorithm

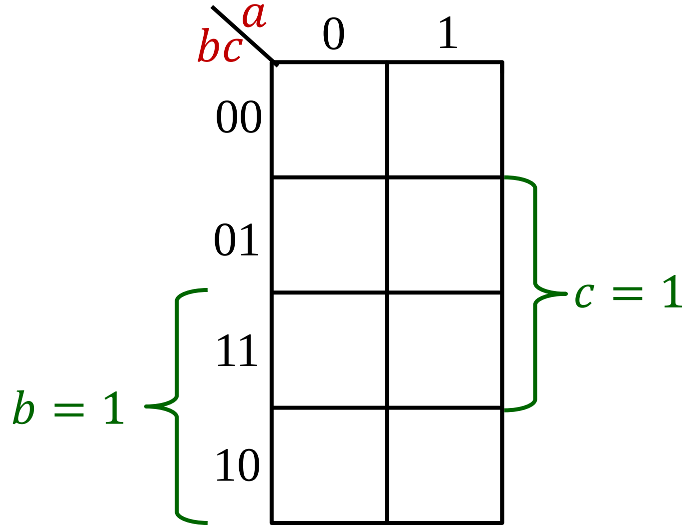
1. Draw the largest rectangular box (squares are a special case of rectangular) that:
   A. Does not include any 0s
   B. Has height ( ) that is a power of 2 ( )
   C. Has width ( ) that is a power of 2
2. Make sure that every 1 on the map is in the **largest possible box**
3. Reduce the products by looking at boxes
   A. The reduced expression will be SOP
   B. The result may not be unique, but will be maximally reduced

K-Map Example

a	b	c	f
0	0	0	**0**
0	0	1	**1**
0	1	0	**0**
0	1	1	**1**
1	0	0	**1**
1	0	1	**1**
1	1	0	**0**
1	1	1	**1**

$$f(a,b,c) = \sum m(1,3,4,5,7)$$

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

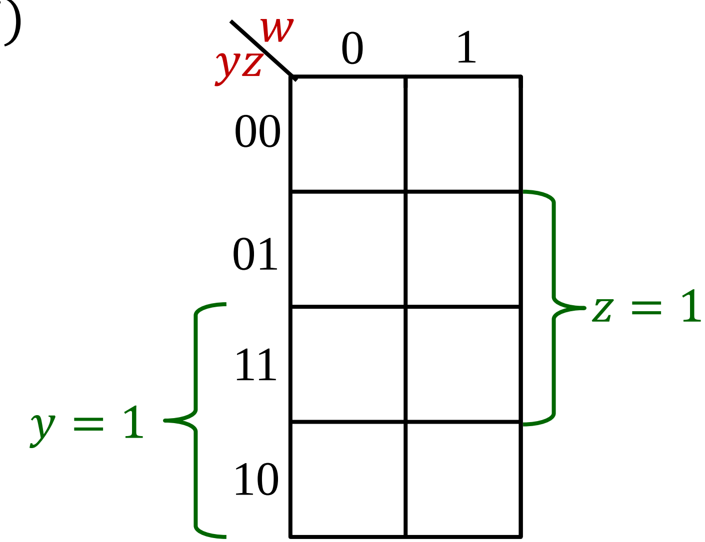


K-Map Example

w	y	z	f
0	0	0	**0**
0	0	1	**1**
0	1	0	**0**
0	1	1	**1**
1	0	0	**1**
1	0	1	**1**
1	1	0	**X**
1	1	1	**1**

$$f(w,y,z) = \sum m(1,3,4,5,7)$$

$$+\sum d(6)$$



Don't Cares in K-maps

- All 1s must be covered
- Xs are used only if they will simplify the resulting expression

Note: Still SOP form

K-Map Amount of Reduction

K-map Boxes that are Groups of:

- 2 → reduce minterm by   variable
- 4 → reduce minterm by   variables
- 8 → reduce minterm by   variables
- 1 → reduce minterm by   variables (full-length minterm)