Combinational Logic Circuits
Part II -Theoretical Foundations

Overview

- Boolean Algebra
  - Basic Logic Operations
  - Basic Identities
  - Basic Principles, Properties, and Theorems
- Boolean Function and Representations
- Truth Table
- Canonical and Standard Forms
  - Minterms and Maxterms
  - Canonical Sum-Of-Products and Product-Of-Sums forms
  - Standard Sum-Of-Products and Product-Of-Sums forms
  - Conversions
- Karnaugh Map (K-Map)
  - 2, 3, 4, and 5 variable K-maps
- Complement of a Boolean function

Boolean Function Representations

- Truth Table (unique representation)
  - Size of a truth table grows exponentially with the number of variables involved
  - This motivates the use of other representations
- Boolean Equation
  - Canonical Sum-Of-Products (CSOP) form (unique)
  - Canonical Product-Of-Sums (CPOS) form (unique)
  - Standard Forms (NOT unique representations)
- Map (unique representation)
- We can convert one representation of a Boolean function into another in a systematic way

Canonical and Standard Forms

- Canonical and Standard forms of a Boolean function are boolean equation representations
- To introduce them we need the following definitions:
  - Literal: A variable or its complement
  - Product term: literals connected by "•"
  - Sum term: literals connected by "+"
  - Minterm: a product term in which all variables appear exactly once, either complemented or uncomplemented
  - Maxterm: a sum term in which all variables appear exactly once, either complemented or uncomplemented

Minterm: Characteristic Property

- A *minterm* of N variables defines a boolean function that represents exactly one combination (b_j) of the binary variables in the truth table
- The function has value 1 for this combination and value 0 for all others
- There are 2^N distinct *minterms* for N variables
- A *minterm* is denoted by m_j
  - 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 it is uncomplemented

Minterms for Three Variables

- For 3 variables X, Y, Z there are 2^3 minterms (products of 3 literals):

  m_0 = X'•Y'•Z' **m_1 = X'•Y'•Z** **m_2 = X'•Y•Z'** **m_3 = X'•Y•Z**

  m_4 = X•Y'•Z' **m_5 = X•Y'•Z** **m_6 = X•Y•Z'** **m_7 = X•Y•Z**

- <u>Example:</u> consider minterm **m_5:**
  - **m_5** defines a boolean function that represents exactly one combination (b_5=101)
  - the function has value 1 for this combination and value 0 for all others
  - variable Y in **m_5** is complemented because its value in b_5 is 0

	X	Y	Z		m_0	m_1	m_2	m_3	m_4	**m_5**	m_6	m_7
b_0	0	0	0		**1**	0	0	0	0	**0**	0	0
b_1	0	0	1		0	**1**	0	0	0	**0**	0	0
b_2	0	1	0		0	0	**1**	0	0	**0**	0	0
b_3	0	1	1		0	0	0	**1**	0	**0**	0	0
b_4	1	0	0		0	0	0	0	**1**	**0**	0	0
b_5	**1**	**0**	**1**		0	0	0	0	0	**1**	0	0
b_6	1	1	0		0	0	0	0	0	**0**	**1**	0
b_7	1	1	1		0	0	0	0	0	**0**	0	**1**

Maxterm: Characteristic Property

- A *maxterm* of N variables defines a boolean function that represents exactly one combination (b_j) of the binary variables in the truth table
- The function has value 0 for this combination and value 1 for all others
- There are 2^N distinct *maxterms* for N variables
- A *maxterm* is denoted by M_j
  - 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 it is uncomplemented

Maxterms for Three Variables

- For 3 variables X, Y, Z there are 2^3 maxterms (sums of 3 literals):

  $\mathbf{M_0 = X+Y+Z}$ $\mathbf{M_1 = X+Y+Z'}$ $\mathbf{M_2 = X+Y'+Z}$ $\mathbf{M_3 = X+Y'+Z'}$

  $\mathbf{M_4 = X'+Y+Z}$ $\mathbf{M_5 = X'+Y+Z'}$ $\mathbf{M_6 = X'+Y'+Z}$ $\mathbf{M_7 = X'+Y'+Z'}$

- Example: consider maxterm $\mathbf{M_5}$:
  - $\mathbf{M_5}$ defines a boolean function that represents exactly one combination (b_5=101)
  - the function has value 0 for this combination and value 1 for all others
  - variables X and Z in $\mathbf{M_5}$ are complemented because their values in b_5 are 1

	X	Y	Z		M_0	M_1	M_2	M_3	M_4	$\mathbf{M_5}$	M_6	M_7
b_0	0	0	0		**0**	1	1	1	1	**1**	1	1
b_1	0	0	1		1	**0**	1	1	1	**1**	1	1
b_2	0	1	0		1	1	**0**	1	1	**1**	1	1
b_3	0	1	1		1	1	1	**0**	1	**1**	1	1
b_4	1	0	0		1	1	1	1	**0**	**1**	1	1
$\mathbf{b_5}$	**1**	**0**	**1**		1	1	1	1	1	**0**	1	1
b_6	1	1	0		1	1	1	1	1	**1**	**0**	1
b_7	1	1	1		1	1	1	1	1	**1**	1	**0**

Canonical Forms (Unique)

- Any Boolean function F( ) can be expressed as:
  - a *unique* **sum** of **min**terms
  - a *unique* **product** of **max**terms
- In other words, every function F( ) has two canonical forms:
  - Canonical Sum-Of-Products (CSOP) (sum of minterms)
  - Canonical Product-Of-Sums (CPOS) (product of maxterms)
- The words **product** and **sum** do not imply arithmetic operations in Boolean algebra!
  - they specify the logical operations **AND** and **OR**, respectively

Canonical Sum-Of-Products

- It is a sum of minterms
- The minterms included are those m_j such that F( ) = 1 in row *j* of the truth table for F( )
- Example:
  - Truth table for F(X,Y,Z) at right
  - The canonical sum-of-products form for F is:
    $F(X,Y,Z) = m_1 + m_2 + m_4 + m_6 =$
    $= X'Y'Z + X'YZ' + XY'Z' + XYZ'$

X	Y	Z		F	
0	0	0		0	
0	**0**	**1**		**1**	$m_1 = X'Y'Z$
0	**1**	**0**		**1**	$m_2 = X'YZ'$
0	1	1		0	
1	**0**	**0**		**1**	$m_4 = XY'Z'$
1	0	1		0	
1	**1**	**0**		**1**	$m_6 = XYZ'$
1	1	1		0	

Canonical Product-Of-Sums

- It is a product of maxterms
- The maxterms included are those M_j such that $F(\ ) = 0$ in row j of the truth table for F( )
- Example:
  - Truth table for F(X,Y,Z) at right
  - The canonical product-of-sums form for F is:
    $F(X,Y,Z) = M_0 \bullet M_3 \bullet M_5 \bullet M_7 =$
    $= (X+Y+Z) \bullet (X+Y'+Z') \bullet (X'+Y+Z') \bullet (X'+Y'+Z')$

X	Y	Z		F	
0	**0**	**0**		**0**	$M_0 = X+Y+Z$
0	0	1		1	
0	1	0		1	
0	**1**	**1**		**0**	$M_3 = X+Y'+Z'$
1	0	0		1	
1	**0**	**1**		**0**	$M_5 = X'+Y+Z'$
1	1	0		1	
1	**1**	**1**		**0**	$M_7 = X'+Y'+Z'$

Shorthand: ∑ and ∏

- $F(X,Y,Z) = m_1 + m_2 + m_4 + m_6 =$
  $= X'Y'Z + X'YZ' + XY'Z' + XYZ' =$
  $=$ **$\sum m(1,2,4,6)$**,
  - ∑ indicates that this is a sum-of-products form
  - **m(1,2,4,6)** indicates to included minterms m_1, m_2, m_4, and m_6

- $F(X,Y,Z) = M_0 \cdot M_3 \cdot M_5 \cdot M_7 =$
  $= (X+Y+Z) \cdot (X+Y'+Z') \cdot (X'+Y+Z') \cdot (X'+Y'+Z') =$
  $=$ **$\prod M(0,3,5,7)$**,
  - ∏ indicates that this is a product-of-sums form
  - **M(0,3,5,7)** indicates to included maxterms M_0, M_3, M_5, and M_7

- **$\sum m(1,2,4,6) = \prod M(0,3,5,7) = F(X,Y,Z)$**

Conversion Between Canonical Forms

1. Get the shorthand notation
2. Replace $\sum$ with $\prod$ (or *vice versa*)
3. Replace those ***j***'s that appeared in the original form with those that do not

- Example:

$$
\begin{aligned}
F(X,Y,Z) &= X'Y'Z + X'YZ' + XY'Z' + XYZ' \\
&= m_1 + m_2 + m_4 + m_6 \\
&= \sum m(1,2,4,6) \\
&= \prod M(0,3,5,7) \\
&= (X+Y+Z)\bullet(X+Y'+Z')\bullet(X'+Y+Z')\bullet(X'+Y'+Z')
\end{aligned}
$$

Standard Forms (NOT Unique)

- There are two types of standard forms:
  - Sum-of-Products (SOP) form (NOT unique)
  - Product-of-Sums (POS) form (NOT unique)

- In standard forms, not all variables need to appear in the individual product or sum terms!

- Example1:
  F(X,Y,Z) = X'Y'Z + X'YZ' + XZ'
  F(X,Y,Z) = X'Y'Z + YZ' + XZ'
  are two *standard* sum-of-products forms

Non-canonical terms

- Example2:
  F(X,Y,Z) = (X+Y+Z) • (X+Y'+Z') • (X'+ Z')
  F(X,Y,Z) = (X+Y+Z) • (Y'+Z') • (X'+ Z')
  are two *standard* product-of-sums form

X	Y	Z		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		1
1	1	1		0

Conversion from Standard to Canonical SOP form

1. Expand *non-canonical* product terms by inserting equivalent of 1 for each missing variable **V**:
   (V + V’) = 1
2. Remove duplicate minterms

- Example:
  F(X,Y,Z) = X’Y’Z + YZ’ + XZ’ =
  = X’Y’Z + (X+X’)YZ’ + X(Y+Y’)Z’
  = X’Y’Z + XYZ’ + X’YZ’ + XYZ’ + XY’Z’
  = X’Y’Z + XYZ’ + X’YZ’ + XY’Z’
- Can you do it:
  F(X,Y,Z) = X’Y’Z + X’YZ’ + XZ’

X	Y	Z		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		1
1	1	1		0

Conversion from Standard to Canonical POS form

1. Expand *non-canonical* sum terms by adding 0 for each missing variable **V**:
   $V \cdot V' = 0$
2. Remove duplicate maxterms

- Example:
  $F(X,Y,Z) = (X+Y+Z)\cdot(Y'+Z')\cdot(X'+Z') =$
  $= (X+Y+Z)\cdot(XX'+Y'+Z')\cdot(X'+YY'+Z')$
  $= (X+Y+Z)\cdot(X+Y'+Z')\cdot(X'+Y'+Z')\cdot$
  $(X'+Y+Z')\cdot(X'+Y'+Z')$
  $= (X+Y+Z)\cdot(X+Y'+Z')\cdot(X'+Y'+Z')\cdot$
  $(X'+Y+Z')$
- Can you do it for:
  $F(X,Y,Z) = (X+Y+Z)\cdot(X+Y'+Z')\cdot(X'+ Z')$

X	Y	Z		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		1
1	1	1		0

Karnaugh Maps (Unique)

- A Karnaugh map (K-map) is a unique graphical representation of a Boolean functions
- K-map of a Boolean function of N variables consists of 2^N cells
- One map cell corresponds to a row in the truth table
- Also, one map cell corresponds to a minterm
- Multiple-cell rectangles in the map correspond to standard terms
- The K-map representation is useful for Boolean functions of up to 5 variables. Why?

Two-Variable K-map

	X	Y	F(X,Y)
0	0	0	F(0,0)
1	0	1	F(0,1)
2	1	0	F(1,0)
3	1	1	F(1,1)

X \ Y	0	1
0	**0** F(0,0)	**1** F(0,1)
1	**2** F(1,0)	**3** F(1,1)

X \ Y	0	1
0	**0** m_0=X'Y'	**1** m_1=X'Y
1	**2** m_2=XY'	**3** m_3=XY

- Cell **0** corresponds to row 0 in the truth table and represents minterm X'Y'; Cell **1** corresponds to row 1 and represents X'Y; etc.
- If Boolean function F(X,Y) has value 1 in a row of the truth table, i.e., a minterm is present in the function, then a 1 is placed in the corresponding cell.

Two-Variable K-map -- Examples

Truth Table | **K - map** | **Canonical and Standard SOP**

X	Y	F1
0	0	0
0	1	0
1	0	0
1	1	1

X\Y	0	1
0		
1		1

$F1 = m_3 = XY$ (canonical)

X	Y	F2
0	0	0
0	1	0
1	0	1
1	1	1

X\Y	0	1
0		
1	1	1

$F2 = m_2 + m_3$

$= XY' + XY$ (canonical)

$= X$ (standard)

X	Y	F3
0	0	0
0	1	1
1	0	1
1	1	0

X\Y	0	1
0		1
1	1	

$F3 = m_1 + m_2$

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

X	Y	F4
0	0	1
0	1	1
1	0	0
1	1	1

X\Y	0	1
0	1	1
1		1

$F4 = m_0 + m_1 + m_3$

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

$= X' + Y$ (standard)

Two-Variable K-map (cont.)

- Any two adjacent cells in the map differ by ONLY one variable
  - appears complemented in one cell and uncomplemented in the other
  - Example:
    m_0 (=X'Y') is adjacent to m_1 (=X'Y) and m_2 (=XY') but NOT m_3 (=XY)

X \ Y	0	1
0	m_0=X'Y'	m_1=X'Y
1	m_2=XY'	m_3=XY

- Multiple-cell rectangles in the map correspond to standard terms
  - Examples:
    - 2-cell rectangle $\boxed{m_2 | m_3}$ corresponds to term X:
      $m_2 + m_3 = XY' + XY = X \cdot (Y' + Y) = X$
    - 4-cell rect. $\begin{array}{|c|c|}\hline m_0 & m_1 \\ \hline m_2 & m_3 \\ \hline\end{array}$ corresponds to constant 1:
      $m_0 + m_1 + m_2 + m_3 = X'Y' + X'Y + XY' + XY =$
      $= X' \cdot (Y' + Y) + X \cdot (Y' + Y) = X + X' = 1$

X \ Y	0	1
0	m_0=X'Y'	m_1=X'Y
1	m_2=XY'	m_3=XY

Three-Variable K-map

	X	Y	Z	F(X,Y,Z)
0	0	0	0	F(0,0,0)
1	0	0	1	F(0,0,1)
2	0	1	0	F(0,1,0)
3	0	1	1	F(0,1,1)
4	1	0	0	F(1,0,0)
5	1	0	1	F(1,0,1)
6	1	1	0	F(1,1,0)
7	1	1	1	F(1,1,1)

Y

X \ YZ	00	01	11	10
0	**0** F(0,0,0)	**1** F(0,0,1)	**3** F(0,1,1)	**2** F(0,1,0)
X 1	**4** F(1,0,0)	**5** F(1,0,1)	**7** F(1,1,1)	**6** F(1,1,0)

Z

Y

X \ YZ	00	01	11	10
0	**0** m_0=X'Y'Z'	**1** m_1=X'Y'Z	**3** m_3=X'YZ	**2** m_2=X'YZ'
X 1	**4** m_4=XY'Z'	**5** m_5=XY'Z	**7** m_7=XYZ	**6** m_6=XYZ'

Z

- Cell **0** corresponds to row 0 in the truth table and represents minterm X'Y'Z'; Cell **1** corresponds to row 1 and represents X'Y'Z; etc.
- If F(X,Y,Z) has value 1 in a row of the truth table, i.e., a minterm is present in the function, then a 1 is placed in the corresponding cell.

Three-Variable K-map -- Examples

Truth Table | **K - map** | **Canonical and Standard SOP**

X	Y	Z	F1
0	0	0	1
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	0

$\longleftrightarrow$

X \ YZ	00	01	11	10
0	1		1	
1		1		1

(Y over columns 11, 10; Z under columns 01, 11; X beside row 1)

$\longleftrightarrow$

$F1 = m_0 + m_3 + m_5 + m_6 =$

$= X'Y'Z' + X'YZ +$

$XY'Z + XYZ'$

(canonical)

X	Y	Z	F2
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

$\longleftrightarrow$

X \ YZ	00	01	11	10
0	1		1	1
1			1	1

(Y over columns 11, 10; Z under columns 01, 11; X beside row 1)

$\longleftrightarrow$

$F2 = m_0 + m_2 + m_3 + m_6 + m_7 =$

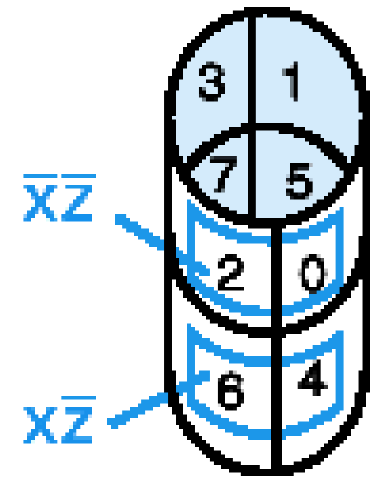
$= X'Y'Z' + X'YZ' + X'YZ +$

$XYZ' + XYZ$ (canonical)

$= \mathbf{X'Z'} + \mathbf{Y}$ (standard)

Three-Variable K-map (cont.)

X \ YZ	00	01	11 (Y)	10 (Y)
0	**0** $m_0=X'Y'Z'$	**1** $m_1=X'Y'Z$	**3** $m_3=X'YZ$	**2** $m_2=X'YZ'$
X 1	**4** $m_4=XY'Z'$	**5** $m_5=XY'Z$	**7** $m_7=XYZ$	**6** $m_6=XYZ'$
		Z	Z	



- NOTE: variable ordering is important – assume function F(X,Y,Z) then X specifies the rows in the map and YZ the columns
- Each cell is adjacent to three other cells (left, right, up or down).
  - Left-edge cells are adjacent to right-edge cells!
- One cell represents a minterm of 3 literals
- A rectangle of 2 adjacent cells represents a product term of 2 literals
- A rectangle of 4 cells represents a product term of 1 literal
- A rectangle of 8 cells encompasses the entire map and produces a function that is equal to logic 1

Four-Variable K-map

	W	X	Y	Z	F(W,X,Y,Z)
0	0	0	0	0	F(0,0,0,0)
1	0	0	0	1	F(0,0,0,1)
2	0	0	1	0	F(0,0,1,0)
3	0	0	1	1	F(0,0,1,1)
4	0	1	0	0	F(0,1,0,0)
5	0	1	0	1	F(0,1,0,1)
6	0	1	1	0	F(0,1,1,0)
7	0	1	1	1	F(0,1,1,1)
8	1	0	0	0	F(1,0,0,0)
9	1	0	0	1	F(1,0,0,1)
10	1	0	1	0	F(1,0,1,0)
11	1	0	1	1	F(1,0,1,1)
12	1	1	0	0	F(1,1,0,0)
13	1	1	0	1	F(1,1,0,1)
14	1	1	1	0	F(1,1,1,0)
15	1	1	1	1	F(1,1,1,1)

Y

WX \ YZ	00	01	11	10
00	**0** m_0=W'X'Y'Z'	**1** m_1=W'X'Y'Z	**3** m_3=W'X'YZ	**2** m_2=W'X'YZ'
01	**4** m_4=W'XY'Z'	**5** m_5=W'XY'Z	**7** m_7=W'XYZ	**6** m_6=W'XYZ'
11	**12** m_{12}=WXY'Z'	**13** m_{13}=WXY'Z	**15** m_{15}=WXYZ	**14** m_{14}=WXYZ'
10	**8** m_8=WX'Y'Z'	**9** m_9=WX'Y'Z	**11** m_{11}=WX'YZ	**10** m_{10}=WX'YZ'

X

W

Z

- Cell **0** corresponds to row 0 in the truth table and represents minterm W'X'Y'Z'; Cell **1** corresponds to row 1 and represents W'X'Y'Z; etc.
- If F(W,X,Y,Z) has value 1 in a row of the truth table, i.e., a minterm is present in the function, then a 1 is placed in the corresponding cell.

Four-Variable K-map -- Examples

W	X	Y	Z	F
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

Truth Table

K - map

WX \ YZ	00	01	11	10
00	1		1	1
01				
11	1	1	1	1
10	1	1	1	1

Y X W Z

K - map

Canonical and Standard SOP

$F = m_0 + m_2 + m_3 +$
$m_8 + m_9 + m_{10} + m_{11} + m_{12} + m_{13} + m_{14} + m_{15} =$

= W’X’Y’Z’ + W’X’YZ’ + W’X’YZ +
WX’Y’Z’ + WX’Y’Z + WX’YZ’ + WX’YZ +
WXY’Z’ + WXY’Z + WXYZ’ + WXYZ (canonical form)

= **W’X’Z’** + W’X’YZ’ + W’X’YZ +
WX’Y’Z’ + WX’Y’Z + WX’YZ’ + WX’YZ +
WXY’Z’ + WXY’Z + WXYZ’ + WXYZ (standard form)

= **W’X’Z’** + **X’Y** +
WX’Y’Z’ + WX’Y’Z + WX’YZ’ + WX’YZ +
WXY’Z’ + WXY’Z + WXYZ’ + WXYZ (standard form)

= **W’X’Z’** + **X’Y** + **W** → (standard form)

Four-Variable K-map (cont.)

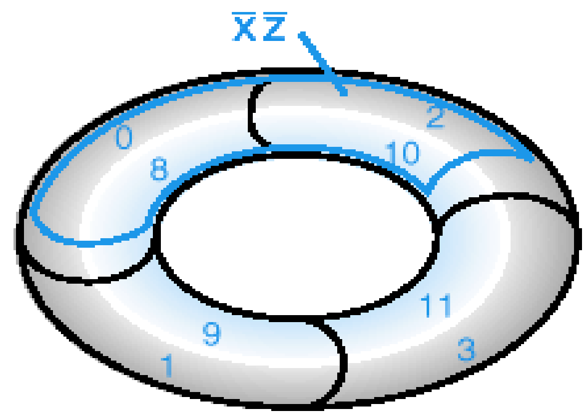
Y

WX \ YZ	00	01	11	10
00	0 m_0 W'X'Y'Z'	1 m_1 W'X'Y'Z	3 m_3 W'X'YZ	2 m_2 W'X'YZ'
01	4 m_4 W'XY'Z'	5 m_5 W'XY'Z	7 m_7 W'XYZ	6 m_6 W'XYZ'
11	12 m_{12} WXY'Z'	13 m_{13} WXY'Z	15 m_{15} WXYZ	14 m_{14} WXYZ'
10	8 m_8 WX'Y'Z'	9 m_9 WX'Y'Z	11 m_{11} WX'YZ	10 m_{10} WX'YZ'

X

W

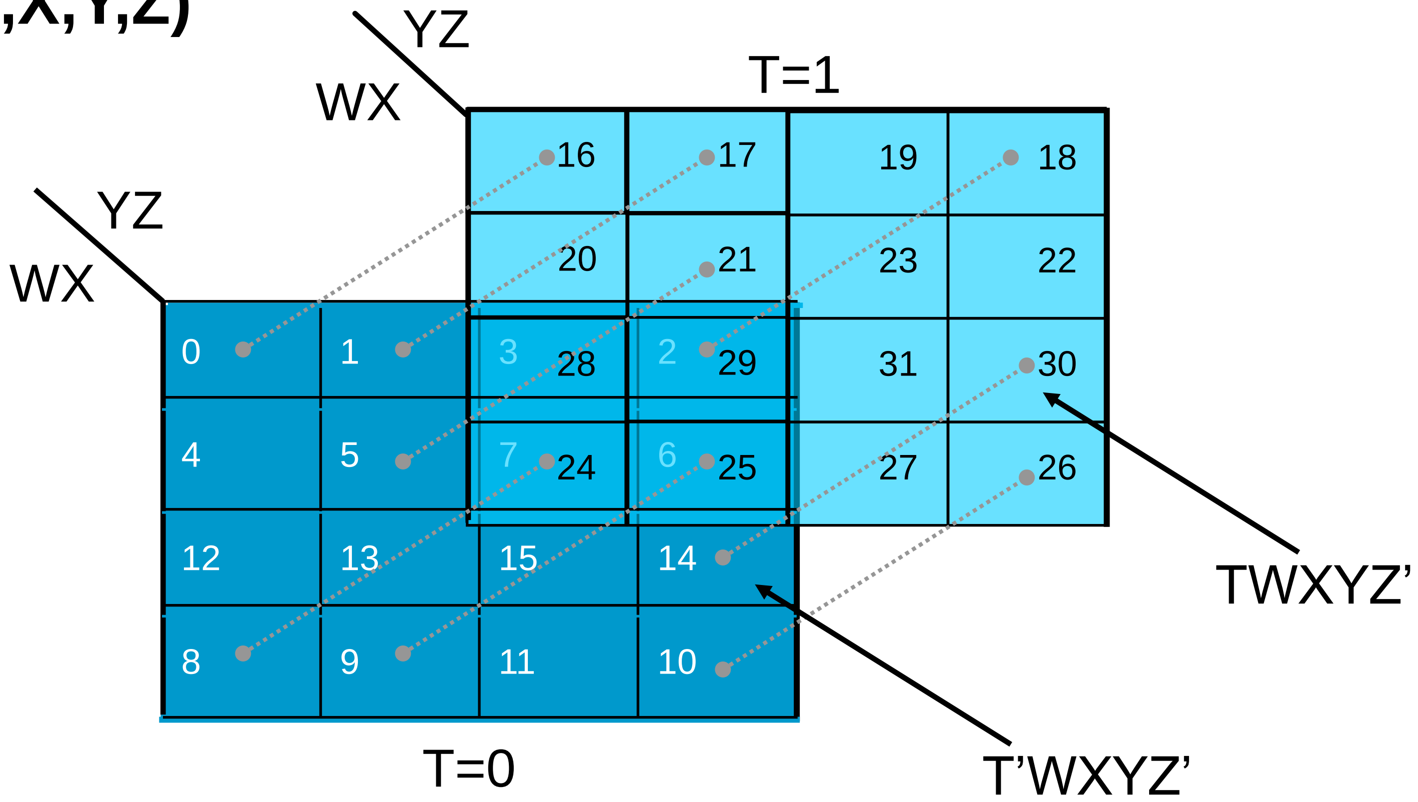
Z



- NOTE: variable ordering is important – assume function F(W,X,Y,Z) then WX specifies the rows in the map and YZ the columns
- Each cell is adjacent to four cells (left, right, up, down)
  - Top cells are adjacent to bottom cells; Left-edge cells are adjacent to right-edge cells
- One cell represents a minterm of 4 literals
- A rectangle of 2 adjacent cells represents a product term of 3 literals
- A rectangle of 4 cells represents a product term of 2 literals
- A rectangle of 8 cells represents a product term of 1 literal
- A rectangle of 16 cells produces a function that is equal to logic 1

Five-Variable K-map

F(T,W,X,Y,Z)



- Can you draw six-variable K-map ?

Complement of a Boolean Function

- The complement representation of function F is denoted as **F'**
- **F'** can be obtained by interchanging 1's to 0's and 0's to 1's in the column showing F of the truth table
- **F'** can be derived by applying DeMorgan's theorem on F
- **F'** can be derived by
  1. taking the dual of F, i.e., interchanging "•" with "+", and "1" with "0" in F and
  2. complementing each literal
- The *complement* of a function IS NOT THE SAME as the *dual* of the function

Complementation: Example

Consider function $F(X,Y,Z) = X'YZ' + XY'Z'$

- Table method

X	Y	Z	F	**F'**
0	0	0	0	**1**
0	0	1	0	**1**
0	1	0	1	**0**
0	1	1	0	**1**
1	0	0	1	**0**
1	0	1	0	**1**
1	1	0	0	**1**
1	1	1	0	**1**

- DeMorgan method:

F' $= (X'YZ' + XY'Z')'$ -- apply DeMorgan

$= (X'YZ')' \bullet (XY'Z')'$ -- DeMorgan again

$= \mathbf{(X+Y'+Z) \bullet (X'+Y+Z)}$

- Dual method:

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

-- interchange "•" with "+" to find the dual of F

$G = (X'+Y+Z')\bullet(X+Y'+Z')$ G is the dual of F

-- complement each literal to find F'

$\mathbf{F' = (X+Y'+Z) \bullet (X'+Y+Z)}$