Digital Logic Design

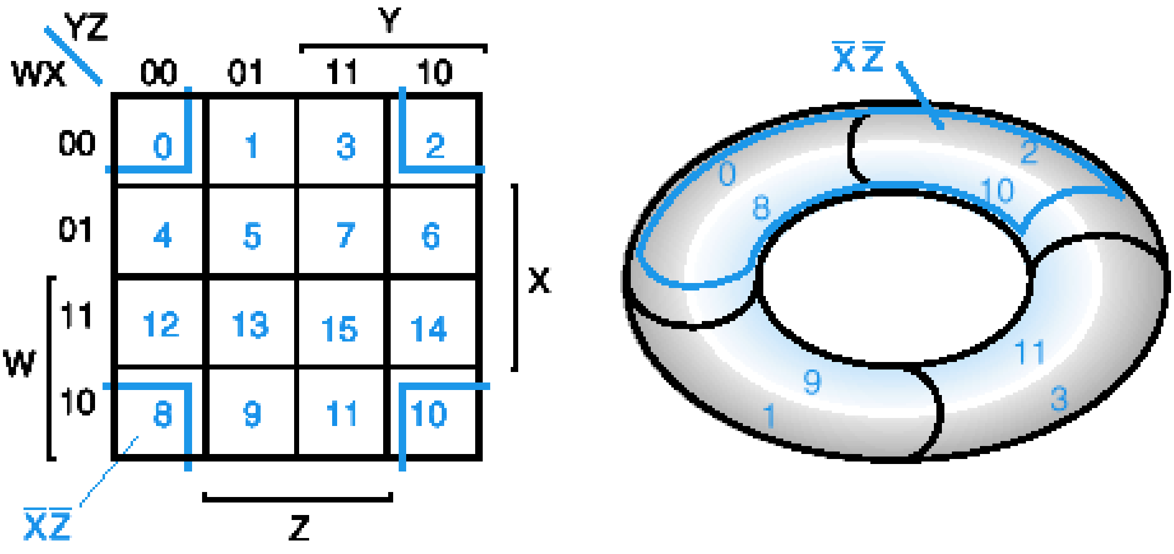
Lecture of Week 11-16 March 2011

K-MAP

Four Variable

The University of Gujrat

Four-Variable Map: Flat and on a Torus to Show Adjacencies



(a) (b)

Four-variable K-Maps

WX \ YZ	00	01	11	10
00				
01				
11				
10				

Four-variable K-Maps

WX \ YZ	00	01	11	10
00	0	1	3	2
01	4	5	7	6
11	12	13	15	14
10	8	9	11	10

F(W,X,Y,Z) = Σ(2,4,5,6,7,9,13,14,15)

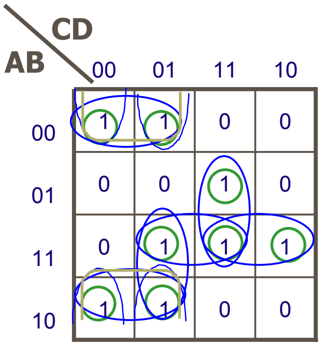
Four-variable K-Maps

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

```
F = !W . X
  + X . Y
  + !W . Y . !Z
  + W . !Y . Z
```

Implicant

- Definition
  - A product term is an **Implicant** of a Boolean function if the function has an output 1 for all minterms of the product term.
- In K-map, an **Implicant** is
  - bubble covers only 1 (bubble size must be a power of 2)



Prime Implicants

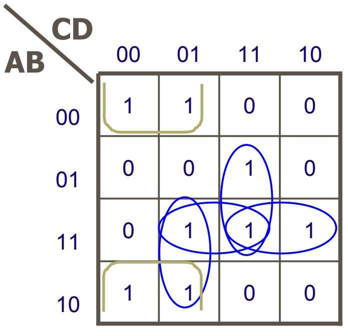
```
F = X . !Y . Z
  + !X . !Z
  + !X . Y
```

Each product term is an **implicant**

A product term that cannot have any of its variables removed and still imply the logic function is called a **prime implicant**.

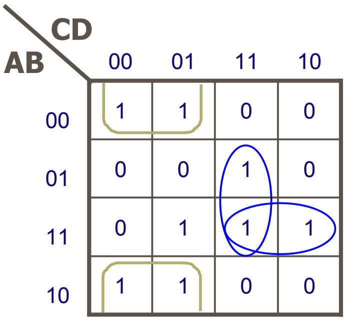
Prime Implicant

- Definition
  - If the removal of any literal from an implicant **I** results in a product term that is not an implicant of the Boolean function, then **I** is an **Prime Implicant**.
  - Examples
    - **BCD** is an implicant, but **CD** or **BD** or **BC** do not imply a 1 in this function; **BCD** is a **PI**
- In K-map, a **Prime Implicant (PI)** is
  - bubble that is expanded as big as possible (bubble size must be a power of 2)



Essential Prime Implicant

- Definition
  - If a minterm of a Boolean function is included in only one PI, then this PI is an **Essential Prime Implicant**.
- In K-map, an **Essential Prime Implicant** is
  - Bubble that contains a 1 covered only by itself and no other PI bubbles



Non-Essential Prime Implicant

- Definition
  - A **Non-Essential Prime Implicant** is a PI that is not an Essential PI.
- In K-map, an **Non-Essential Prime Implicant** is
  - A 1 covered by more than one PI bubble

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

Simplification for SOP

- Form K-Map for the given Boolean function
- Identify all Essential Prime Implicants for 1's in the K-map
- Identify non-Essential Prime Implicants in the K-map for the 1's which are not covered by the Essential Prime Implicants
- Form a sum-of-products (SOP) with all Essential Prime Implicants and the necessary non-Essential Prime Implicants to cover all 1's

Example for SOP

- Identify all the essential PIs for 1's
- Identify the non-essential PIs to cover 1's
- Form an SOP based on the selected PIs

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

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

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

or

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

Example for SOP

- Identify all the essential PIs for 1's
- Identify the non-essential PIs to cover 1's
- Form an SOP based on the selected PIs

$$F = \sum m(0,1,7,8,9,13,14,15)$$

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

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

or

$$F = \bar{B}\bar{C} + ABC + BCD + ABD$$

Example for SOP

- Identify all the essential PIs for 1's
- Identify the non-essential PIs to cover 1's
- Form an SOP based on the selected PIs

$$F = \prod M(1,3,4,6,11,12)$$

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

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

or

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

Prime Implicants

- All the prior definitions apply to '0' (or maxterm) as well
- Consider these implicants imply a '0' output

Simplification for POS

- Form K-Map for the given Boolean function
- Identify all Essential Prime Implicants for 0's in the K-map
- Identify non-Essential Prime Implicants in the K-map for the 0's which are not covered by the Essential Prime Implicants
- Form a product-of-sums (POS) with all Essential Prime Implicants and the necessary non-Essential Prime Implicants to cover all 0's

Example for POS

- Identify all the essential PIs for 0's
- Identify the non-essential PIs to cover 0's
- Form an POS based on the selected PIs

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

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

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

Example for POS

- Identify all the essential PIs for 0's
- Identify the non-essential PIs to cover 0's
- Form an POS based on the selected PIs

$$F = \prod M(1, 3, 4, 6, 11, 12)$$

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

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

Don't Care Condition — X

- Don't care (X)
  - Those input combinations which are irrelevant to the target function (i.e. If the input combination signals can be guaranteed never occur)
  - Can be used to simplify Boolean equations, thus simply logic design
- In K-map
  - Use X to express Don't Care in the map
  - Don't care can be bubbled as 1 or 0 depending on SOP or POS simplification to result into bigger bubble

Another Example of Don’t Care (SOP)

$$F(A,B,C,D)=\sum m(2,3,4,6,11,12)+d(10,13,14)$$

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

$$F = B\overline{D} + \overline{B}C$$

Another Example of Don’t Care (POS)

$$F(A,B,C,D)=\sum m(2,3,4,6,11,12)+d(10,13,14)$$

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

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