Computer Architecture

Keivan Navi

[Cal Poly Pomona University]

[knavi@cpp.edu]

Office hours: Tu/Th 5:25 Pm to 6:55 Pm

Office: 8-49

Computer Architecture

Karnaugh Map

K-map

Karnaugh Map (K-map)

- A Karnaugh map (K-map) is a visual diagram that simplifies Boolean algebra expressions. It's a variation of a truth table that's useful for understanding logic circuits.
- Maurice Karnaugh introduced the K-map in 1953 as an improvement on Edward W. Veitch's 1952 Veitch chart. K-maps are best for functions with two to four variables, but can be used for functions with five or six variables. Using K-maps for functions with seven or more variables is very difficult or impossible.

K- map (continued)

- To create a K-map, you group adjacent 1's in rows and columns. The rules for K-map groups are:
- Groups must contain only 1's
- Groups must be rectangles
- The sides of the rectangle must be a power of two: 1, 2, or 4
- Every 1 is covered by at least one group
- There are as few groups as possible
- Each group is as large as possible

Examples of K-map (2 input gates)

- NAND gate: (ab)'= $\overline{ab}$

	a'	a
b'	1	1
b	1	0

- NOR gate : (a+b)' =$\overline{a + b}$

	a'	a
b'	1	0
b	0	0

- b:

	a'	a
b'	0	0
b	1	1

- a XNOR b: (a xor b)'

	a'	a
b'	1	0
b	0	1

Examples of K-map (3 input gates)

- 1- c’

	a'b'	a'b	ab	ab'
c'	1	1	1	1
c	0	0	0	0

(1)

- 3- (a xor b xor c)’=
- XNOR (a,b,c)

	a'b'	a'b	ab	ab'
c'	1	0	1	0
c	0	1	0	1

(3)

2- a’b+bc’

	a'b'	a'b	ab	ab'
c'	0	1	1	0
c	0	1	0	0

(2)

4- a’b’+ a’c’+ b’c’=

Minority (a,b,c)

	a'b'	a'b	ab	ab'
c'	1	1	0	1
c	1	0	0	0

(4)

Example of 3 input K-map (continued)

- 5- b':

	a'b'	a'b	ab	ab'
c'	1	0	0	1
c	1	0	0	1

(5)

6- bc'+ab

	a'b'	a'b	ab	ab'
c'	0	1	1	0
c	0	0	1	0

(6)

- 7- ab+ac'+bc' = Majority (a, b, c')

	a'b'	a'b	ab	ab'
c'	0	1	1	1
c	0	0	1	0

(7)

Example of 4 input K-map

- d’:

	a'b'	a'b	ab	ab'
c'd'	1	1	1	1
c'd	0	0	0	0
cd	0	0	0	0
cd'	1	1	1	1

c’d’:

	a'b'	a'b	ab	ab'
c'd'	1	1	1	1
c'd	0	0	0	0
cd	0	0	0	0
cd'	0	0	0	0

- (a XNOR b XNOR c XNOR d)’

	a'b'	a'b	ab	ab'
c'd'	1	0	1	0
c'd	0	1	0	1
cd	1	0	1	0
cd'	0	1	0	1

a’b’ + ab =a XNOR b

	a'b'	a'b	ab	ab'
c'd'	1	0	1	0
c'd	1	0	1	0
cd	1	0	1	0
cd'	1	0	1	0

Example of 4 input K-map (continued)

- a’b’ + ab + c’d

	a'b'	a'b	ab	ab'
c'd'	1	0	1	0
c'd	1	1	1	1
cd	1	0	1	0
cd'	1	0	1	0

a’b’c’+abc= c(a’b’+ab)=
=c (a XNOR b)

	a'b'	a'b	ab	ab'
c'd'	1	0	1	0
c'd	1	0	1	0
cd	0	0	0	0
cd'	0	0	0	0

- (a XOR b XOR c XOR d)+ bd

	a'b'	a'b	ab	ab'
c'd'	1	0	1	0
c'd	0	1	1	1
cd	1	1	1	0
cd'	0	1	0	1

(a XOR b XOR c XOR d)’(a+b+c+d)

	a'b'	a'b	ab	ab'
c'd'	0	0	1	0
c'd	0	1	0	1
cd	1	0	1	0
cd'	0	1	0	1

Sum of Product/ Product of Sum

- **Minterms:** Minterms are the fundamental part of Boolean algebra. Minterm is **the product of N literals where each literal occurs exactly once**. Minterm is represented by m. The output for the minterm functions is 1.
- **Maxterms:** Maxterm is a fundamental part of Boolean algebra. Maxterms are **the sum of various distinct literals in which each literal occurs exactly once**. Maxter is represented M. The output result of the maxterm function is 0.
- **SOP uses minterms to create a Boolean expression which is product of Boolean variables whereas POS uses maxterms to create a Boolean expression which is sum of Boolean variables.**

Example of minterm

- abc+ a’bc’+ de:
- In this example abc, “a’bc’ and de are mintems
- ABCD+ FGH +HYT’
- In this example ABCD, FGH and HYT’ are minterms.
- (a+b)(e+f):
- In this example a+b and e+f are Maxterms
- (yazd+abc’+de)(a+b’)(g’+h’):
- In this example yazd, a+b’ and g’+h’ are Maxterms

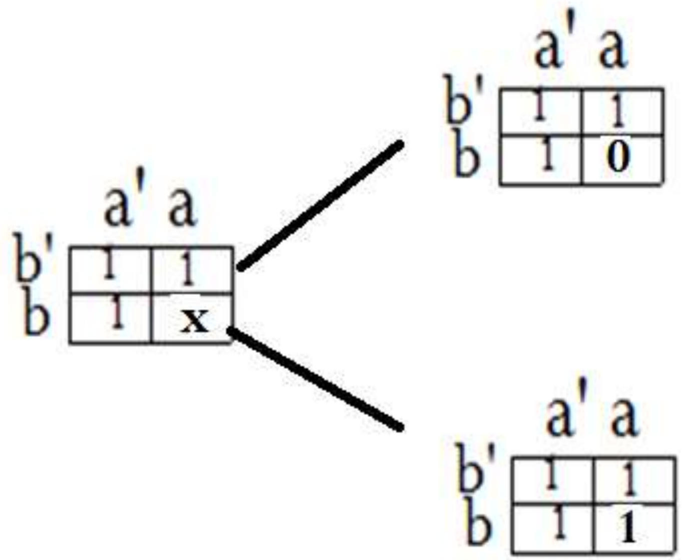
Don’t Care (x)

- “Don't care” may also refer to **an unknown value in a multi-valued logic system**, in which case it may also be called an X value or don't know.
- The "Don't care" condition says that **we can use the blank cells of a K-map to make a group of the variables**. To make a group of cells, we can use the "don't care" cells as either 0 or 1, and if required, we can also ignore that cell. We mainly use the "don't care" cell to make a large group of cells.
- **Don't Care cell can be represented by a cross(X) or minus(-) or phi(Φ) in K-Maps representing.**

Examples of Don’t Cares

- We have two choices

	a'	a
b'	1	1
b	1	x



- Which one is a better choice?
- Why?

Examples of Don’t Care (continued)

- We have two choices:

	a'b'	a'b	ab	ab'
c'	1	x	1	0
c	0	1	0	1

- Which one do you chose?

	a'b'	a'b	ab	ab'
c'	1	x	1	0
c	0	1	0	1

	a'b'	a'b	ab	ab'
c'	1	0	1	0
c	0	1	0	1

	a'b'	a'b	ab	ab'
c'	1	1	1	0
c	0	1	0	1

Examples of Don't Care (continued)

- What is the better choice?

	a'b'	a'b	ab	ab'
c'd'	1	0	1	0
c'd	1	x	1	1
cd	1	0	1	0
cd'	1	0	1	0

	a'b'	a'b	ab	ab'
c'd'	1	0	1	0
c'd	1	0	1	1
cd	1	0	1	0
cd'	1	0	1	0

	a'b'	a'b	ab	ab'
c'd'	1	0	1	0
c'd	1	1	1	1
cd	1	0	1	0
cd'	1	0	1	0

Examples of Don’t Care (continued)

	a'b'	a'b	ab	ab'
c'd'	1	0	1	0
c'd	1	x	1	x
cd	1	0	1	0
cd'	1	0	1	0

	a'b'	a'b	ab	ab'
c'd'	1	0	1	0
c'd	1	0	1	1
cd	1	0	1	0
cd'	1	0	1	0

	a'b'	a'b	ab	ab'
c'd'	1	0	1	0
c'd	1	0	1	0
cd	1	0	1	0
cd'	1	0	1	0

	a'b'	a'b	ab	ab'
c'd'	1	0	1	0
c'd	1	1	1	0
cd	1	0	1	0
cd'	1	0	1	0

	a'b'	a'b	ab	ab'
c'd'	1	0	1	0
c'd	1	1	1	1
cd	1	0	1	0
cd'	1	0	1	0