Lecture 4, Jan. 15, 2007

Gate level minimization : Karnaugh map

We learned how to represent a logic circuit using a truth table and how to translate a truth table into an algebraic expression. However, the expression found is not necessary the optimal expression for implementing the digital circuit that was represented by the truth table. The fact is that while each digital system has only one truth table, the algebraic expression representing a system is not unique. That is we can have several expression representing the same digital system. The task of gate level minimization consists in finding the optimal expression, usually, in the sense of minimizing the number of gates used in the expression.

One tool used in gate level minimization is a Karnaugh Map or a K-map. A K-map for a digital system with n variables consists of 2^n squares. Each square represents one minterm.

<u>Two variable K-map</u>

Two variables x and y take 4 vales 00, 01, 10, and 11. These correspond to minterms m_0, m_1, m_2, m_3.

A 2X2 array of squares can house these minterms

$x \backslash y$	0	1
0	$x'y'$	$x'y$
1	xy'	xy

If we put a 1 in those squares for whom the value of the function is 1, we get a useful means for representing the digital system.

For example, the AND function has the following K-map:

$x \backslash y$		
		1

Since only the square corresponding to $m_3 = xy$ is marked, The function is xy.

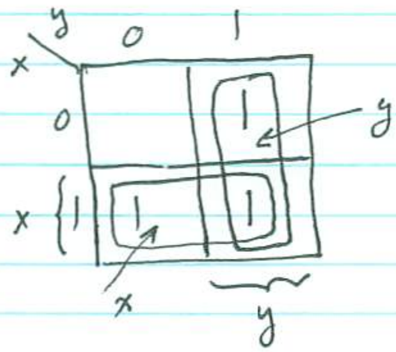
For OR, we have the following K-map

$x \backslash y$	0	1
0		1
1	1	1

The function is:

$$m_1 + m_2 + m_3 = x'y + xy' + xy = x + y$$

Another way to do this is to combine the marked squares into groups of two for which either x ~~[illegible]~~ or y takes the box uncomplemented and complemented values.



Note that the horizontal group (the second row) has y and y' so, it represents the ~~[illegible]~~ variable x, i.e.,

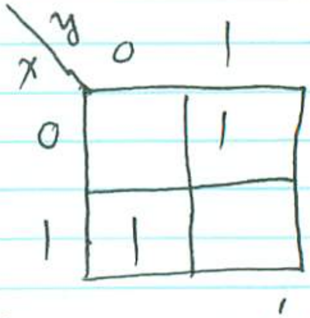
$$xy' + xy = x(y + y') = x$$

Similarly, the second column is $x'y + xy = y$

So, the logic is $x + y$.

Example: Find the optimal gate level expression for

x	y	F
0	0	0
0	1	1
1	0	1
1	1	0



$$F = m_1 + m_2 = x'y + xy'$$

Three varile K-map

For three variables x, y, z, we need $2^3 = 8$ squares.

The K-map is:

$x \backslash yz$	00	01	11	10
0	$x'y'z'$ m_0	$x'y'z$ m_1	$x'yz$ m_3	$x'yz'$ m_2
1	$xy'z'$ m_4	$xy'z$ m_5	xyz m_7	xyz' m_6

y

z

Note that the first row corresponds to x', since x is equal to zero for all squares on this row while y and z take all values 00, 01, 11, 10.

Similarly the second row simplifies to x.

For squares on the right correspond to y and for middle squares represent z.

Note that yz values are not written is normal binary order. They are rather written as Gray codes so that any two adjacent square are different in only one bit.

Example: Simplify the function:

$$F(x,y,z) = \sum(2,3,4,5)$$

$x'y$

$x \backslash yz$	00	01	11	10
0			1	1
1	1	1		

xy'

$$F(x,y,z) = xy' + x'y$$

Note that, we need to consider a K-map as a group of squares drawn on a ~~torod~~ toroid (a Do-nut) rather than on a flat surface as such the square at the beginning and the end of each row are adjacent. The same is true for those squares located at the ~~beginning~~ beginning and the end of each column.

Example: Simplify $F(x,y,z) = \sum(3,4,6,7)$

yz

$x \backslash yz$	00	01	11	10
0			1	
1	1		1	1

$$F(x,y,z) = yz + xz'$$

Example: Simplify the function

$$F(x, y, z) = \sum(0, 2, 4, 5, 6)$$

$x \backslash yz$	00	01	11	10
0	1			1
1	1	1		1

The four squares at the beginning and the end of the rows have all variants of xy an have 0 for z. So, they represent z'. The two squares on the second row correspond to xy'. So:

$$F(x, y, z) = z' + xy'.$$

Four-varible K-map

$wx \backslash yz$	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}

In a 4-variable map:

One square represents one minterm with 4 literals
Two adjacent squares represent a term with three literals
4 adjacent squares represent a term with 2 variables
8 adjacent squares represent a term with one literal.

Example: Simplify the function

$$F(w,x,y,z) = \Sigma(0,1,2,4,5,6,8,9,12,13,14)$$

wx \ yz	00	01	11	10
00	1	1		1
01	1	1		1
11	1	1		1
10	1	1		

$w'z'$

xz'

y'

$$F(w,x,y,z) = y' + w'z' + xz'$$

Prime Implicants:

A prime implicant is a product term formed by combining the maximum possible number of squares in a K-map. If a single square cannot be combined with any other square it forms a prime implicant. similarly, any two adjacent squares that cannot be part of a group of 4 adjacent cells form a prime implicant. Also 4 adjacent squares that cannot be part of a group of 8 adjacent cells form a prime implicant.

If there is a square that can only be part of one prime implicant then that prime implicant is called an <u>essential</u> <u>prime</u> <u>implicant</u>.

The procedure for minimizing the expression for a logical circuit represented by a K-map starts by finding all <u>essential</u> <u>prime</u> <u>implicants</u>.

The simplified expression is then formed by logical sum (OR) of those terms and the minterms remaining (those not included in the essential prime implicants).

Example: Find the minimal expression for:

$$F(A,B,C,D) = \sum(0,2,3,5,7,8,9,10,11,13,15)$$

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

There are two essential prime implicants one representing BD and another representing $B'D'$.

The three remaining minterms may be grouped with different minterms. One grouping is to combine m_3 and m_{11} with m_7 and m_{15} to form CD and combining m_9 and m_{11} with m_8 and m_{10} to form AB'. Then

$$F = BD + B'D' + CD + AB'$$

Another way is to combine m_3, m_7, m_{11}, m_{15} to get CD and to combine m_9 and m_{11} with m_{13} and m_{15} to get AD.

Then

$$F = BD + B'D' + CD + AD$$

By combining m_3 and m_{11} with m_2 and m_{10} we get $B'C$ and we have two other minimal expressions:

$$F = BD + B'D' + B'C + AB'$$

and

$$F = BD + B'D' + B'C + AD$$