CS47 - Lecture 14

Kaushik Patra
(kaushik.patra@sjsu.edu)

1

- Standard Forms
- Karnaugh Map

[Chapter 2 (2-3, 2-4) of Logic & Computer Design Fundamentals, 4th Edition, M. Morris Mano, Charles R. Kime]

[Appendix B of 'Computer Organization & Design by Hennessy, Patterson]

Standard Forms ...

SOP & POS

- There are two types of standard forms
  - Sum of Product (**<u>SOP</u>**) terms
  - Product of Sum (**<u>POS</u>**) terms
- Purpose of Standard forms
  - Compact forms for larger expression
  - Facilitate simplification process

3

- Standard form contains product terms and sum terms.
  - Product terms contain only AND operations. Example of product terms will be XYZ'
  - Sum terms contain only OR operations. Example of sum terms will be X+Y'+Z.
- Sum of Product (SOP) is OR operations between product terms. An example of SOP expression is ABC + B'C + A'B.
- Product of Sum (POS) is AND operations between sum terms. An example of POS expression is (A+B+C)(B'+C)(A'+B).

Minterms & Maxterms

- Extension of two types of term in SOP, POS
  - '**Minterms**' are the ***product*** terms containing ***all the variables*** of the functions appearing ***exactly once*** in their complemented or un-complemented form.
  - '**Maxterms**' are the ***sum*** terms containing ***all the variables*** of the functions appearing ***exactly once*** in their complemented or un-complemented form.

4

- There are total 2^n minterms or maxterms for a n-variable Boolean functions.
- Both minterms and maxterms property is that they represent exactly one combination of binary variable values in the truth table.
- Example of a minterm for a function F(X,Y,Z) will be X'YZ'. There will be total $2^3 = 8$ minterms for this function.
- Example of a minterm for a function F(X,Y,Z) will be (X' + Y + Z'). There will be total $2^3 = 8$ maxterms for this function.

SOP of minterms

X	Y	Z	Product Term	Symbol	F(X,Y,Z)
0	0	0	X'Y'Z'	m_0	1
0	0	1	X'Y'Z	m_1	0
0	1	0	X'YZ'	m_2	1
0	1	1	X'YZ	m_3	0
1	0	0	XY'Z'	m_4	0
1	0	1	XY'Z	m_5	1
1	1	0	XYZ'	m_6	0
1	1	1	XYZ	m_7	1

$F(X,Y,Z) = X'Y'Z' + X'YZ' + XY'Z + XYZ = m_0 + m_2 + m_5 + m_7 = \Sigma m(0,2,5,7)$

5

- Each row in the truth table represents one unique minterm.

- In a minterm variable is complemented for a specific row in the truth table if its corresponding value in the truth table for that term is 0. If the value is 1, non-complemented form of the variable is taken for constructing the minterm. For example for the 3rd row of the above truth table has value entry of X=0, Y=1 and Z=0. Hence the corresponding minterm is X'YZ'.

- If a function contains high number of variables (e.g. 100 variable boolean function) it is not very efficient to write all the variables in the minterm. That is why, minterms are alternatively represented as m_x, where x is the decimal value corresponding the binary value represented by the variables in that row. For example X'YZ' is represented a m_2.

- The function is expressed in SOP as logical sum of all the minterms which results the function value to be 1 in the truth table. Instead of writing explicitly, the sum is expressed as sum of the corresponding minterm symbol. To further reduce the expression notation classical algebraic summation symbol is used with list of indices for the minterm. For example 'F(X,Y,Z) = Σm(0,2,5,7)' means 'F(X,Y,Z) = X'Y'Z' + X'YZ' + XY'Z + XYZ'.

POS of maxterms

X	Y	Z	Sum Term	Symbol	F(X,Y,Z)
0	0	0	X+Y+Z	M_0	1
0	0	1	X+Y+Z'	M_1	0
0	1	0	X+Y'+Z	M_2	1
0	1	1	X+Y'+Z'	M_3	0
1	0	0	X'+Y+Z	M_4	0
1	0	1	X'+Y+Z'	M_5	1
1	1	0	X'+Y'+Z	M_6	0
1	1	1	X'+Y'+Z'	M_7	1

$F(X,Y,Z) = (X+Y+Z')(X+Y'+Z')(X'+Y+Z)(X'+Y'+Z) = M_1M_3M_4M_6 = \Pi M(1,3,4,6)$

6

- Each row in the truth table represents one unique maxterm.

- In a maxterm variable is complemented for a specific row in the truth table if its corresponding value in the truth table for that term is 1. If the value is 0, non-complemented form of the variable is taken for constructing the maxterm. For example for the 3rd row of the above truth table has value entry of X=0, Y=1 and Z=0. Hence the corresponding maxterm is (X+Y'+Z).

- Maxterms are alternatively represented as M_x, where x is the decimal value corresponding the binary value represented by the variables in that row. For example (X+Y'+Z) is represented a M_2.

- The function is expressed in POS as logical product of all the maxterms which results the function value to be 0 in the truth table. Instead of writing explicitly, the product is expressed as product of the corresponding maxterm symbol. To further reduce the expression notation classical algebraic multiplication symbol is used with list of indices for the maxterm. For example 'F(X,Y,Z) = ΠM(1,3,4,6)' means 'F(X,Y,Z) = (X+Y+Z')(X+Y'+Z')(X'+Y+Z)(X'+Y'+Z)'.

Relation between SOP & POS

- Maxterm and minterm with same index holds complement relation.
  - e.g. $M_3 = m_3'$
  - $M_3 = X + Y' + Z' = (X'YZ)' = m_3'$
- If $F(X,Y,Z) = \Sigma m(0,2,5,7)$, then $F'(X,Y,Z) = \Sigma m(1,3,4,6)$
  - i.e. $F(X,Y,Z) = (m_1 + m_3 + m_4 + m_6)' = m_1'.m_3'.m_4'.m_6'$
  - Hence $F(X,Y,Z) = M_1.M_3.M_4.M_6 = \Pi M(1,3,4,6)$
  - SOP expression is equivalent to its POS expression with exact set of disjoint indices to its SOP equivalent.

7

- To calculate SOP expression of complement of a function in SOP form, we take those entries in the truth table which results 0. In the complement function those rows with 0 as result in original function will become 1 and rows with 1 as the result in the original function will become zero. The indices of the minterm set in the original function is disjoint to the indices of the minterm set in the complimented function.

- Fun facts
  - Maxterms are seldom used directly when dealing with Boolean functions, since we can always replace them with minterm expression.
  - A function that includes all the minterms is equal to logic 1.
  - Any Boolean function can be expressed as logical sum of minterms.

Karnaugh Map ...

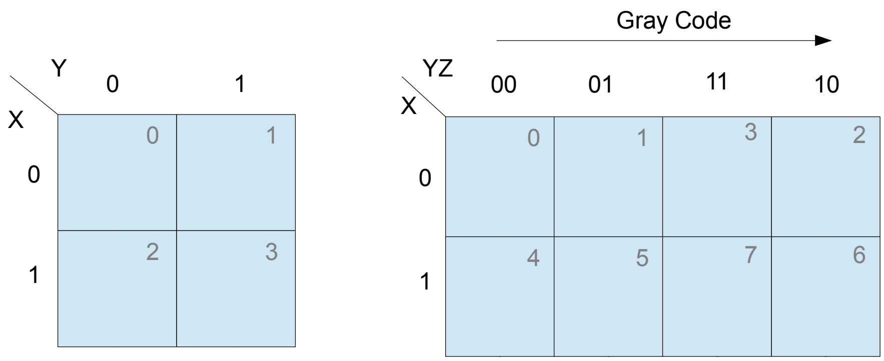
8

Karnaugh Map

- It is a visual technique to reduce Boolean expression in standard form (SOP or POS).
  - It takes advantage of human pattern recognition ability.
- First introduced by Maurice Karnaugh in 1953.
  - It was a refinement of Veitch diagram technique developed in 1952 by Edward Veitch.
- It is usually called as K-map technique.

9

K-Map Table Structure



Two variable K-Map

Three variable K-Map

10

- The number of rows and columns are exact power of 2. For example for a 3 variable K-Map, number of rows is 2 and columns is 4.

- Corresponding variables are written on the top-left corner of the table.

- Table row and column indices are written in Gray Code.
  - Gray code encoding of number ensures that two adjacent binary numbers differs only by one bit.
  - Each column are adjacent to each other.
  - The first and last columns are adjacent to each other.
  - Each rows are adjacent to each other.
  - The first and last row are adjacent to each other.

- Each cell / box of the table is marked with corresponding minterm number.

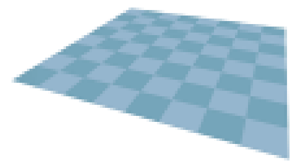
K-Map Table Structure

Gray Code

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

Gray Code

Four variable K-Map



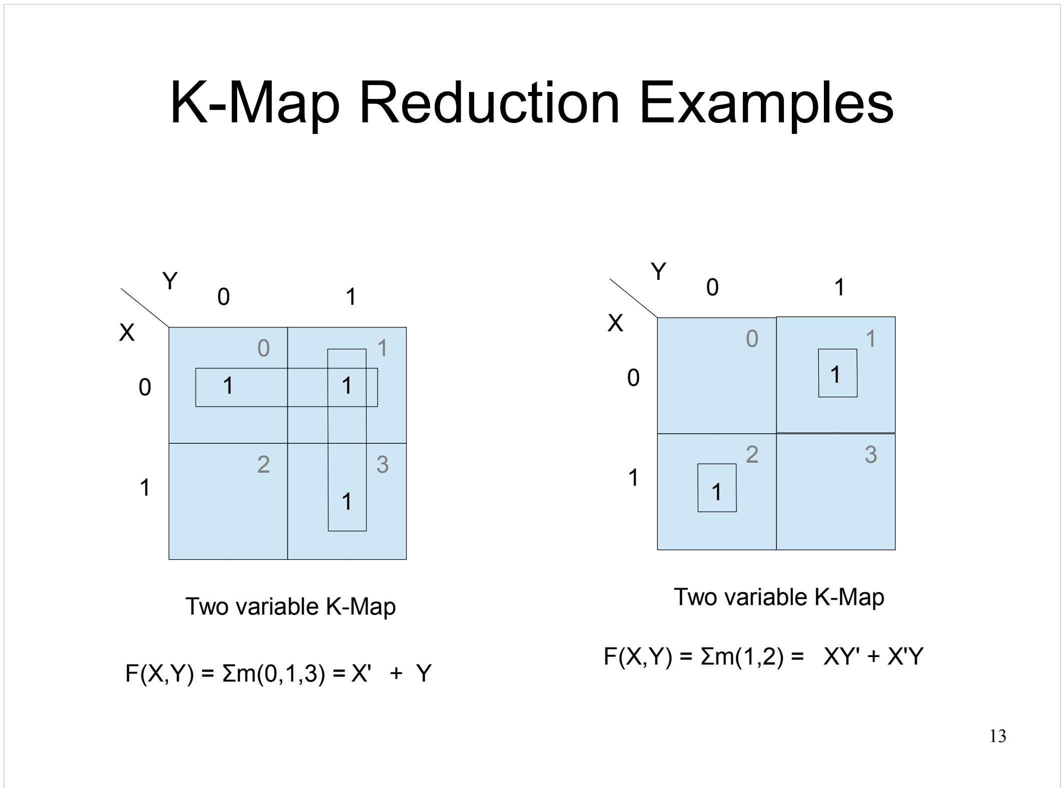
11

K-Map Reduction Process

- For a SOP expression put 1 for each minterm in the expression in the corresponding box in the table.
- Group adjacent 1s in maximum way in power of two – 1, 2, 4, 8, etc.
- Write one product term for each such group of one.

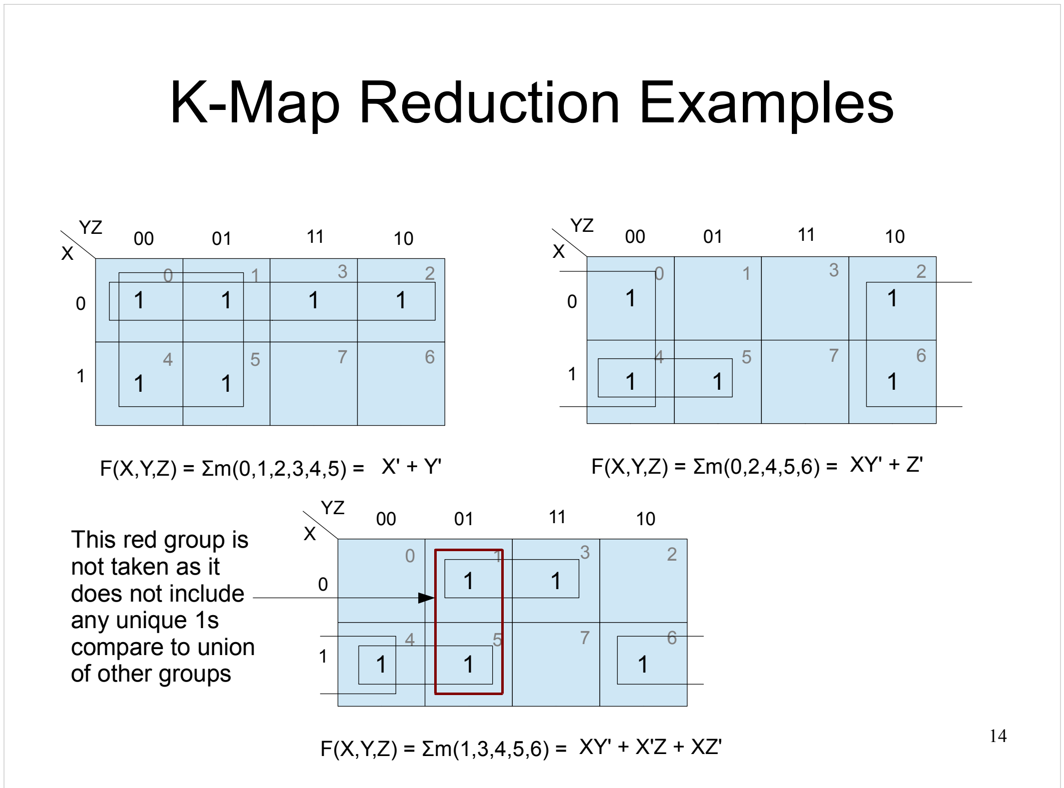
12

- For a Boolean expression $F(X,Y,Z) = \Sigma m(2,3,5,6)$, 1 will be placed in the box with minterm indices 2,3,5, and 6.
- Different group of one can overlap each other. However, they must contain some unique 1s, i.e. one group can not be a subset of the union of the other group.
- Product terms will contain only the variable (or its complementary form) if and only if the variable does not change its value in the group.
- All the identified product terms are then summed to have a reduced SOP form of the expression.



- For Boolean expression F(X,Y) = Σm(0,1,3) there are two groups chosen.
  - The boxes in the table for the minterm 0 and 1 are horizontally adjacent hence they are grouped together. For this group, only variable X does not vary and stays at value 0. This gives the term X' in the reduced expression.
  - The boxes in the table for the minterm 1 and 3 are vertically adjacent, hence they also group together. For this group variable Y does not vary, but X does. The variable Y stays at 1. This gives the term Y in the reduced expression.
  - Combining two groups, we get X'+Y as the reduced expression for the given Boolean expression.

- For the Boolean expression F(X,Y) = Σm(1,2), there are also two groups can be chosen. However these two groups are singular group (meaning that it contains only one 1 in the group).
  - The box for minterm 2 gives the term XY' in the reduced expression sine both X and Y remains constant at 1 and 0 respectively.
  - The box for minterm 1 gives the term X'Y in the reduced expression sine both X and Y remains constant at 0 and 1 respectively.
  - Combining two groups we get the reduced (nothing reduced here in this case) expression as X'Y + XY'

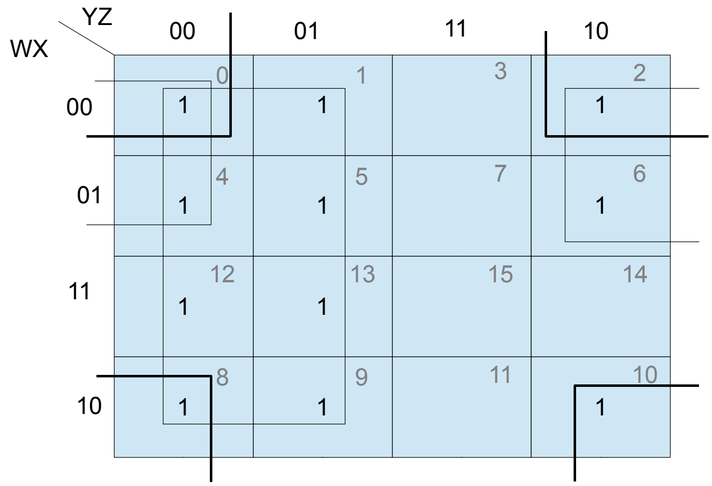


- For the expression F(X,Y,Z) = Σm(0,1,2,3,4,5) the following groups are chosen.
  - All the boxes for minterms 0,1,3,2 are adjacent and X remains constant at 0. Hence the term X'.
  - All the boxes for minterms 0,1,4,5 are adjacent and Y remain constant at 0. Hence the term Y'.
  - Combining both the term, we get reduced expression as X' + Y'

- For the expression F(X,Y,Z) = Σm(0,2,4,5,6) the following groups are chosen.
  - All the boxes for minterms 0,2,4,6 are adjacent (outer two columns are adjacent to each other) and Z remains constant at 0. Hence the term Z'.
  - All the boxes for minterms 4,5 are adjacent and both X and Y remain constant at 1 and 0 respectively. Hence the term XY'.
  - Combining both the term, we get reduced expression as XY' + Z'

- For the expression F(X,Y,Z) = Σm(1,3,4,5,6) the following groups are chosen.
  - All the boxes for minterms 1,3 are adjacent. In this group X and Z remains constant at 0 and 1 respectively. Hence the term X'Z.
  - All the boxes for minterms 4,5 are adjacent. In this group X and Y remain constant at 1 and 0 respectively. Hence the term XY'.
  - All the boxes for minterms 4,6 are adjacent (outer two columns are adjacent to each other. In this group X and Z are constant at 1 and 0. Hence the term XZ'
  - Combining both the term, we get reduced expression as XY'+X'Z+XZ'
  - The minterms 1,5 also could be grouped, but they are not including any unique minterms that is not included in other group. Hence this group was not considered. If it had been considered, then we could drop the group 4.5 and the reduced expression would be X'Z + XY' + Y'Z.

K-Map Reduction Examples

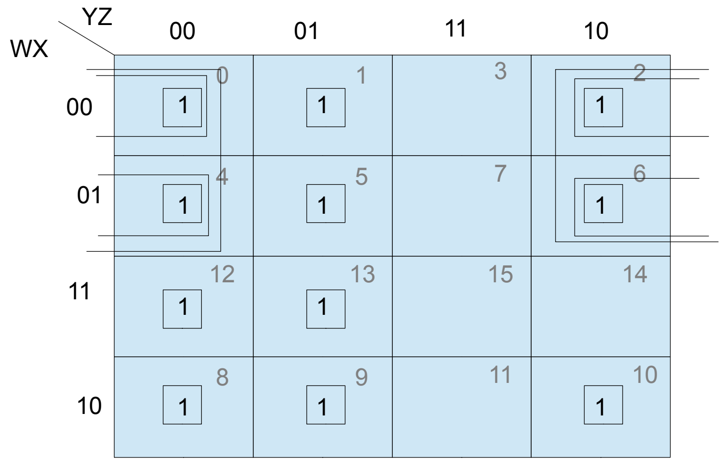


F(W,X,Y,Z) = Σm(0,1,2,4,5,6,8,9,10,12,13) = Y' + W'Z' + X'Z'

15

- In the Boolean expression F(W,X,Y,Z) = Σm(0,1,2,4,5,6,8,9,10,12,13) the groups are chosen as following.
  - The minterms 0,1,4,5,12,13,8,9 are adjacent to each other. For this group only Y remains constant at 0 giving the term Y'.
  - The minterms 0,2,4,6 are adjacent (since the boundary columns are adjacent). For this group both W and Z are constant at 0 giving the term W'Z'.
  - The minterms 0,2,8,10 are adjacent since the boundary rows and columns are adjacent to each other. For this group both X and Z are constant at 0 giving the term X'Z'.
  - Combining three individual product terms we get he SOP expression as Y'+W'Z'+X'Z'.

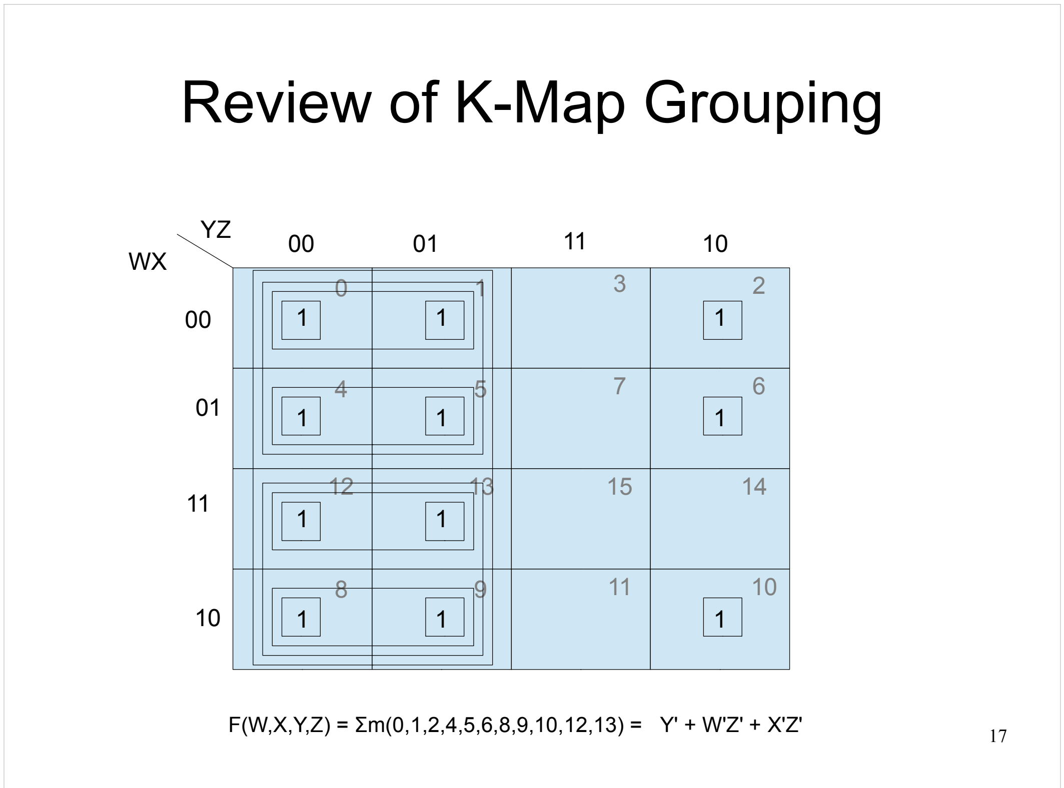
Review of K-Map Grouping



F(W,X,Y,Z) = Σm(0,1,2,4,5,6,8,9,10,12,13) = Y' + W'Z' + X'Z'

16

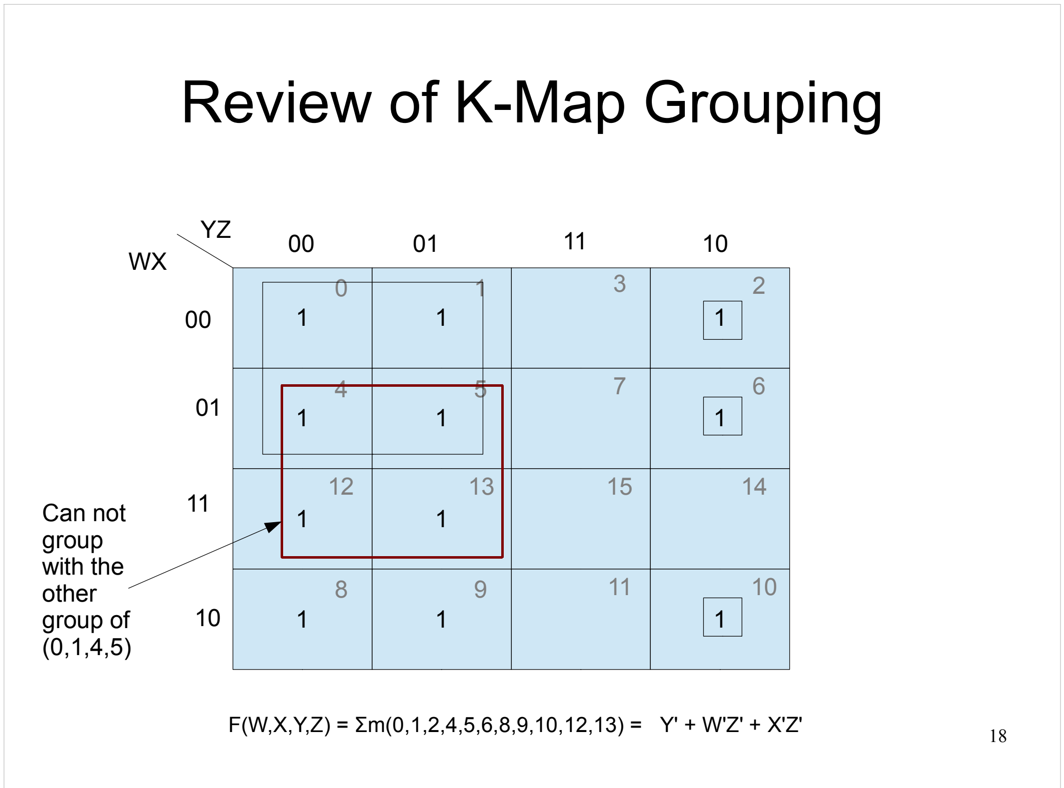
- We group adjacent to reduce the product term. Grouping two adjacent term means that they have only one variable changing and hence can be reduced to a new product term by eliminating the variable literal. For example grouping 0 and 2 minterms means (W'X'Y'Z' + W'X'YZ') = W'X'Z'(Y'+Y) = W'X'Z'.1 = W'X'Z'. Similarly we can reduce 4 and 6 as W'XZ'. Now since 0 and 4 are adjacent and 2,6 are adjacent, then we can group (0,4) and (2.6) into (0,4,2,6) which will have exactly one variable literal. In this case it is X. So we can express the term as (W'X'Z' + W'XZ') = W'Z'(X+X') = W'Z'.1 = W'Z'.



- We can group two mutually exclusive adjacent groups with same number of minterms. This means, we group two adjacent groups with one minterm, which will give a group of 2 minterms. Then we can group two adjacent groups with 2 minterms each to for a group of 4 minterms and so on. For example, initial group can be (0,1) (4,5) (12,13) (8.9). Then we can group ((0,1) (4,5)) and ((12, 13) (8,9)) because
  - Boundary element in the individual group has exact one adjacent to another boundary element in other group.
    - For example of ((0,1) (4,5)), 0 is adjacent to 4 and 1 is adjacent to 5 hence these two groups can be merged into one group of (0,1,4,5).
  - Each group has exact same number of minterms.
  - Both groups are mutually exclusive.

- Now we can also merge group (0,1,4,5) and (12,13,8,9) because boundary elements (4,5) is adjacent to boundary elements (12,13) in other group. Also each group has exactly same number of elements and they are mutually exclusive. Hence they can be merged into (0,1,4,5,12,13,8,9).

- This is why we group minterms into power of 2, like 1, 2, 4, 8, etc. number of minterms in a group.



- We need to merge mutually exclusive group because otherwise it will not result in any further reduction. For example, (0,1,4,5) can not be merged with (4.5.12.13) and give a group (0,1,4,5,12,13). First of all this will result in a group cardinal of 6 which is not a power of 2.
  - Group (0.1.4.5) gives a reduced term as W'Z' and (4,5,12,13) gives XZ'. Grouping them together will give a term W'Z' + XZ' which can not be reduced further. This happened because we did not group the minterms which are differing from each other with respect to change in only one variable. In this case, Z' remains same, but W' suddenly disappeared in other term and a new literal of X came in. With respect to row indices, grouping (0,1,4,5) and (4,5,12,13) means both W and X are varying in this grouping, which contradicts the grouping rule where only one variable can change its value.

Map Manipulation

- How do we choose the groups for reduction?

- Let's define the following terminology.
  - An **implicant** is any product term of a SOP form for expression.
  - A ***prime implicant*** is an product term which can not be covered by any other more general (reduced term – terms with fewer literals).
  - An ***essential prime implicant*** is an prime implicant that covers an output of the function in such a way the no other combination of prime implicant can cover.

19

- For a boolean function F, if a product term P evaluated to be 1 will also make F as 1 them P is an implicant of F. In this case, we say P implies F.
  - For example, F(X,Y,Z,W) = XY + YZ + W has may implicant, such as XY, YZ, XYZ. XYZW, and many mores.

- If we removed any term from a prime implicant, it can no longer stays as prime implicant. For the above example, ZY, YZ and W are the prime implicants. XYZ and XYZW are not prime implicant since can be further generalized to XY, YZ, and W.

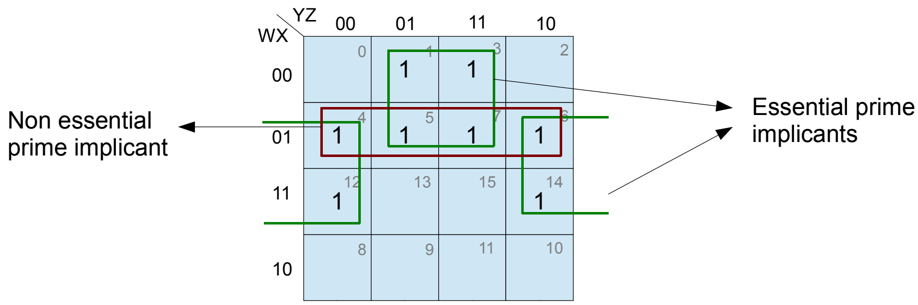
- If a prime implicant contains a minterm that no other prime implicant has, the prime implicant is called essential prime implicant. In this example XY and YZ and W are the essential prime implicants.

- An essential prime implicant contains a minterm that no other prime implicant has.

- The sum of all prime implicant is called complete sum or minimal covering sum.

Map Manipulation

- Identifying essential and non-essential prime implicants in the K-map.



F(W,X,Y,Z) = Σm(1,3,4,5,6,7,12,14) = W'Z + XZ' + W'X = W'Z + XZ'

20

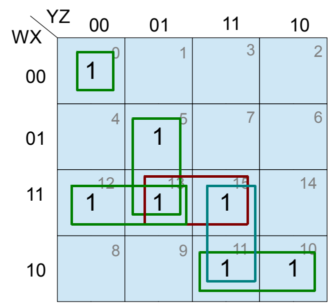
- The expression F(W,X,Y,Z) = Σm(1,3,4,5,6,7,12,14) contains prime implicants of W'Z, XZ' and W'X. Among these W'Z is one essential prime implicant since it contains minterms 1,3 which are not included in any other prime implicants. Similarly XZ' is also an essential prime implicant since it contains the minterms 12, 14 which are not included in any other prime implicants. However the term W'X is not an essential prime implicant since all the minterms it covers (4,5,6,7) are included into other prime implicants.

- The essential prime implicant should contain at least one minterm which is not included into any other prime implicants. If it is not the case, then we call the prime implicant as non-essential prime implicant.

- This example also demonstrate the K-map based explanation of the Consensus theorem. In the consensus theorem, the non-essential prime implicant gets dropped from the expression.

Map Manipulation

- There can be a situation where selecting one essential prime implicant will make another prime implicant non-essential.



F(W,X,Y,Z) = Σm(0,5,10, 11, 12,13,15) = W'X'Y'Z' + XY'Z + WX'Y + **WXZ**

OR

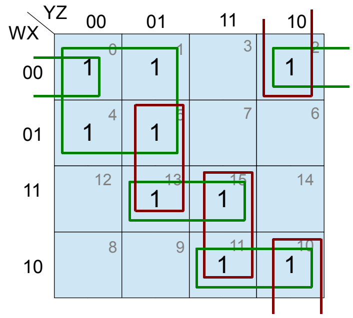
W'X'Y'Z' + XY'Z + WX'Y + **WYZ**

21

- In this example of F(W,X,Y,Z) = Σm(0,5,12,13,15,11,10) we have three distinct essential prime implicants – W'X'Y'Z' , XY'Z , WX'Y. Now to include the minterm 15 exclusively it can be grouped into either WXZ or WYZ. In this case, if we use WXZ in the expression then WYZ does not remain as an essential prime implicant. Similarly if we use WYZ in the expression then WXZ is no longer an essential prime implicant.

Map Manipulation

- ***Selection rule*** states that select non-overlapping essential prime implicants as much as possible.



F(W,X,Y,Z) = Σm(0,1,2,4,5,10,11,13,15) = W'Y' + WXZ + WX'Y + W'X'Z'

or

W'Y' + XY'Z + WYZ + X'YZ'

but not

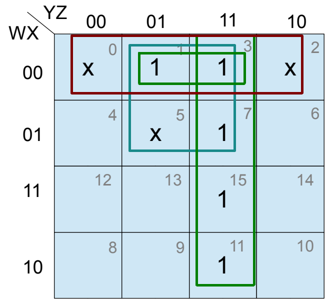
W'Y' + XY'Z + WYZ + WX'Y + W'X'Z'

22

- In this example of F(W,X,Y,Z) = Σm(0,1,2,4,5,10,11,13,15) \
  - We can group the minterms as (**0**,1,4,5),(13,16),(10,11),(**0**,2) with one overlap of minterm 0 (marked in bold in group). This gives 4 product terms in the SOP form of this expression.
  - We also can group minterms as (0,1,4,**5**),(**5**,13),(11,15),(2,12) with one overlap of minterm 5 (marked in bold in group). This also gives 4 product terms in the SOP form of this expression.
  - We can also group (**0**,1,4,**5**),(**5**,12),(15,**11**),(10,**11**),(**0**,2). Even though all of them are essential prime implicants, it gives 3 overlaps and produce 5 product terms in SOP form of this expression which is clearly not most reduced form that can be obtained for this expression.

Use of don't care term

- Treat don't care term (x) as 1 if required to minimize the expression better.



F(W,X,Y,Z) = Σm(1,3,7,11,15) + d(0,2,5) = YZ + W'X'Z

or

YZ + W'X'

or

YZ + W'Z

23

- Don't care terms are the minterms which does not dictate overall function value. Even if the minterm becomes 1, it does not make the function 1. Even if the minterm becomes 0, it does not make the function 0.

- Don't care terms in an expression are denoted as d(list of minterms). For example, d(0,2,5) means the minterms 0,2, and 5 are don't care terms for the expression F(W,X,Y,Z) = Σm(1,3,7,11,15) + d(0,2,5). In this case, we can choose not to involve don't care terms and have a reduced from of YZ+W'X'Z).

- Don't care terms can be involved in K-map based minimization because we can include them in the reduced form since they do not affect the final value of the function. In the above example, if we include minterms 0 and 2 then the reduced form will be (YZ+W'X'). If we include minterm 5, the the reduced form will be (YZ+W'Z).

- Inclusion of don't care term reduce the product terms. In the above example, before inclusion of the don't care terms, the minimization process gave two product term with 2 and 3 variable involved. By including the don't care terms, we were able to express the function as two product terms with 2 variables each.

- Though represented by 'x', "don't care" and "unknown" terminology are different from each other. They convey totally different ideas. Don't care term is applicable in context of minterm, where as the unknown value is applicable to denote a value (not known) for a Boolean variable.

CS47 - Lecture 14

Kaushik Patra
(kaushik.patra@sjsu.edu)

24

- Standard Forms
- Karnaugh Map

[Chapter 2 (2-3, 2-4) of Logic & Computer Design Fundamentals, 4th Edition, M. Morris Mano, Charles R. Kime]

[Appendix B of 'Computer Organization & Design by Hennessy, Patterson]

