Karnaugh maps

Jia Chen
jiac@ucr.edu

Part of slides adopted from Hung-Wei Tseng

Karnaugh map

- A systematic and graphical way to reduce the product or sum terms in an expression.

- Key: apply uniting property as judiciously as possible

Uniting Theorem	$a\,b + a\,b' = a$	$(a + b)\cdot(a + b') = a$

Truth table

f = x3'+x1x2'

Row number	x_1	x_2	x_3	f
0	0	0	0	1
1	0	0	1	0
2	0	1	0	1
3	0	1	1	0
4	1	0	0	1
5	1	0	1	1
6	1	1	0	1
7	1	1	1	0

	x_1	x_2	x_3
m_0	0	0	0
m_2	0	1	0
m_4	1	0	0
m_6	1	1	0

	x_1	x_2	x_3
m_4	1	0	0
m_5	1	0	1

- If x3=0, f=1 regardless of the values of x1 and x2
- If x1=1and x2=0, f=1 regardless of the value of x3
- How to easily discover groups of minterms for f=1 that can be combined into single terms?

Karnaugh map

- An alternative to the truth-table form for representing a function

- A map consists of *cells* corresponding to the *rows* of the truth table

Location of two-variable minterms

x_1	x_2	
0	0	m_0
0	1	m_1
1	0	m_2
1	1	m_3

(a) Truth table

$x_2 \backslash x_1$	0	1
0	m_0	m_2
1	m_1	m_3

(b) Karnaugh map

- Advantage: minterms in any two cells that are adjacent, either in the same row or the same column, can be combined.
- Test m2+m3 ?

Steps for SOP simplification

- Create a 2-D truth table with input variables on each dimension, and adjacent column(j)/row(i) only change one bit in the variable.
- Fill each (i , j) with the corresponding result in the truth table
- Combine 2, 4, 8, 16 , …, 2^n Minterms to obtain a SINGLE product term
  - Therefore, in a K map, we can only circle 2, 4, 8, 16 , …, 2^n adjacent cells to obtain a single term!
  - How to get a SINGLE product term (see next slide)
- Find the "minimum cover" that covers all 1s in the graph
- OR the united product terms of all minimum cover

How to get a SINGLE product term ?

- A product terms include only those variables having the same value for all cells in the group represented by this term
- If the variable is 1 in the group, it appears uncomplemented
- If the variable is 0 in the group, it appears complemented

Strategy for SOP simplification

- Intuitive strategy: find as few as possible for the number of groups & as large as possible for the number of cells with 1s for each group

- Each group of 1s has to comprise cells that can be represented by a single product term

- The larger the group of 1s, the fewer the number of variables in the corresponding product term

2-variable K-map example

Input		Output
A	B	
0	0	**1**
0	1	**1**
1	0	**1**
1	1	**0**

		A'	**A**
	B \ A	**0**	**1**
B'	**0**	1	1
B	**1**	1	0

B'

A'

F(A, B) = A' + B'

Practicing 2-variable K-map

- What's the simplified function of the following K-map?
  A. A'
  B. A'B
  C. AB'
  D. B
  E. A

B \ A	0	1
0	0	0
1	1	1

Practicing 2-variable K-map

- What’s the simplified function of the following K-map?
  A. A’
  B. A’B
  C. AB’
  D. B
  E. A

		A’	A
	B \ A	0	1
B’	0	0	0
B	1	1	1

B

3-variable K-map?

- Reduce to 2-variable K-map — 1 dimension will represent two variables
- Adjacent points should differ by only 1 bit
  - So we only change one variable in the neighboring column
  - 00, 01, 11, 10 — such numbering scheme is so-called **Gray–code**

Input			Output
A	B	C	
0	0	0	**1**
0	0	1	**1**
0	1	0	**1**
0	1	1	**1**
1	0	0	**1**
1	0	1	**0**
1	1	0	**1**
1	1	1	**0**

		A'B'	A'B	AB	AB'	
	C \ AB	0,0	0,1	1,1	1,0	
C'	0	1	1	1	1	C'
C	1	1	1	0	0	

A'

F(A, B, C) = A' + C'

Minimum number of SOP terms

- Minimum number of SOP terms to cover the following function?
  A. 1
  B. 2
  C. 3
  D. 4
  E. 5

Input			Output
A	B	C	
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	**0**
1	1	1	**1**

Minimum number of SOP terms

- Minimum number of SOP terms to cover the following function?
  A. 1
  $\boxed{\text{B. 2}}$
  C. 3
  D. 4
  E. 5

Input			Output
A	B	C	
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	**0**
1	1	1	**1**

F(A, B, C) = A'C' + BC'

		A'B'	**A'B**	**AB**	**AB'**
		0,0	0,1	1,1	1,0
C'	0	1	1	0	0
C	1	0	1	1	0

A'C'

A'B

BC

We don't need A'B to cover all 1s

x

Minimum number of SOP terms

- Minimum number of SOP terms to cover the following function?
  A. 1
  B. 2
  C. 3
  D. 4
  E. 5

Input			Output
A	B	C	
0	0	0	**1**
0	0	1	**1**
0	1	0	**0**
0	1	1	**0**
1	0	0	**1**
1	0	1	**1**
1	1	0	**0**
1	1	1	**0**

Minimum number of SOP terms

- Minimum number of SOP terms to cover the following function?

  A. 1

  B. 2

  C. 3

  D. 4

  E. 5

Input			Output
A	B	C	
0	0	0	**1**
0	0	1	**1**
0	1	0	**0**
0	1	1	**0**
1	0	0	**1**
1	0	1	**1**
1	1	0	**0**
1	1	1	**0**

	C (A,	A'B' 0,0	A'B 0,1	AB 1,1	AB' 1,0	A'B' 0,0
C'	0	1	0	0	1	1
C	1	1	0	0	1	1

B'

x

Minimum SOP terms

- What's the minimum sum-of-products expression of the given truth table?
  A. A'B'C' + A'BC'+ A'BC + AB'C'
  B. A'B'C + AB + AC
  C. AB'C' + B'C'
  D. A'B + B'C'
  E. A'C' + A'B + AB'C'

Input			Output
A	B	C	
0	0	0	**1**
0	0	1	**0**
0	1	0	**1**
0	1	1	**1**
1	0	0	**1**
1	0	1	**0**
1	1	0	**0**
1	1	1	**0**

Minimum SOP terms

- What's the minimum sum-of-products expression of the given truth table?
  A. A'B'C' + A'BC'+ A'BC + AB'C'
  B. A'B'C + AB + AC
  C. AB'C' + B'C'
  D. A'B + B'C'
  E. A'C' + A'B + AB'C'

	A'B' 0,0	A'B 0,1	AB 1,1	AB' 1,0
C' 0	1	1	0	1
C 1	0	1	0	0

B'C'

A'B

Input			Output
A	B	C	
0	0	0	**1**
0	0	1	**0**
0	1	0	**1**
0	1	1	**1**
1	0	0	**1**
1	0	1	**0**
1	1	0	**0**
1	1	1	**0**

4-variable K-map

- Reduce to 2-variable K-map — both dimensions will represent two variables
- Adjacent points should differ by only 1 bit
  - So we only change one variable in the neighboring column
  - Use Gray-coding — 00, 01, 11, 10

		A'B'	A'B	AB	AB'
		00	01	11	10
C'D'	00	1	0	0	0
C'D	01	1	0	0	0
CD	11	0	0	0	0
CD'	10	1	0	0	1

A'B'C'

B'CD'

F(A, B, C) = A'B'C'+B'CD'

4-variable K-map

- What's the minimum sum-of-products expression of the given K-map?
  A. B'C' + A'B'
  B. B'C'D' + A'B' + B'C'D'
  C. A'B'CD' + B'C'
  D. AB' + A'B' + A'B'D'
  E. B'C' + A'C'D'

		A'B'	A'B	AB	AB'
		00	01	11	10
C'D'	00	1	0	0	1
C'D	01	1	0	0	1
CD	11	0	0	0	0
CD'	10	1	1	0	0

4-variable K-map

- What's the minimum sum-of-products expression of the given K-map?
  - A. B'C' + A'B'
  - B. B'C'D' + A'B' + B'C'D'
  - C. A'B'CD' + B'C'
  - D. AB' + A'B' + A'B'D'
  - E. B'C' + A'C'D'

		A'B'	A'B	AB	AB'
		00	01	11	10
C'D'	00	1	0	0	1
C'D	01	1	0	0	1
CD	11	0	0	0	0
CD'	10	1	1	0	0

B'C'

A'CD'

Steps for POS simplification

- Create a 2-D truth table with input variables on each dimension, and adjacent column(j)/row(i) only change one bit in the variable.
- Fill each (i , j) with the corresponding result in the truth table
- Combine 2, 4, 8, 16 , …, 2^n Maxterms to obtain a SINGLE sum term
    - ➢ Therefore, in a K map, we can only circle 2, 4, 8, 16 , …, 2^n adjacent cells to obtain a single term!
    - ➢ How to get a SINGLE sum term (see next slide)
- Find the “minimum cover” that covers all 0 s in the graph
- AND the united sum terms of all minimum cover

How to get a SINGLE sum term ?

- A sum terms include only those variables having the same value for all cells in the group represented by this term

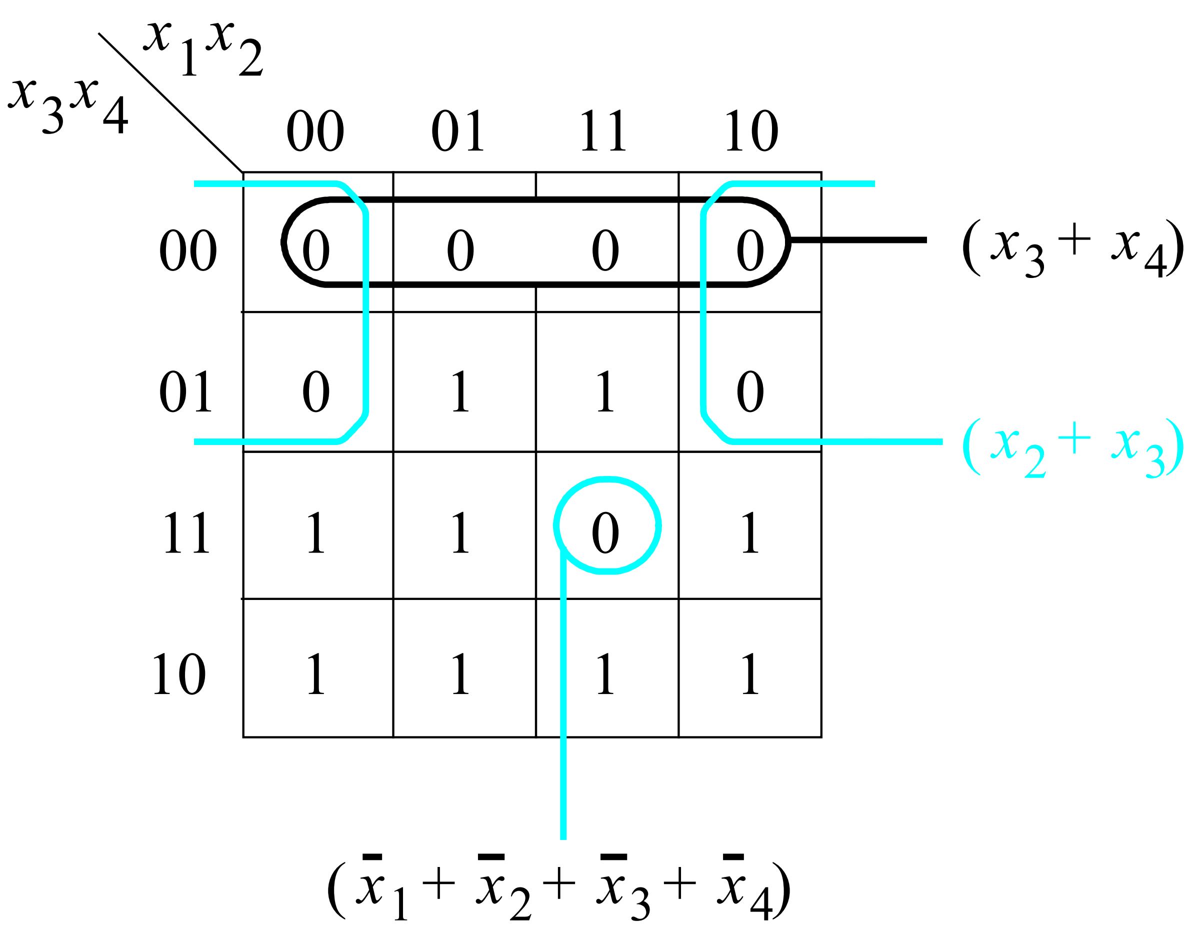
- If the variable is 1 in the group, it appears complemented

- If the variable is 0 in the group, it appears uncomplemented

Strategy for POS simplification

- Intuitive strategy: find as few as possible for the number of groups & as large as possible for the number of cells with 0s for each group
- Each group of 0 s has to comprise cells that can be represented by a single sum term
- The larger the group of 0 s, the fewer the number of variables in the corresponding sum term

Simplification of PoS forms



F = (x3+x4)(x2+x3)(x1’+x2’+x3’+x4’)

Incompletely Specified Functions

- Situations where the output of a function can be either 0 or 1 for a particular combination of inputs
- This is specified by a don’t care in the truth table
- This happens when
  - The input does not occur. e.g. Decimal numbers 0… 9 use 4 bits, so (1,1,1,1) does not occur.
  - The input may happen but we don’t care about the output. E.g. The output driving a seven segment display – we don’t care about illegal inputs (greater than 9)

K-Map with “Don’t Care”s

You can treat “X” as either 0 or 1
— depending on which is more advantageous

		A’B’	A’B	AB	AB’
		0,0	0,1	1,1	1,0
C’	0	1	0 X 1	1	1
C	1	1	1	0	0

K-Map with “Don’t Care”s

		A’B’	A’B	AB	AB’
		0,0	0,1	1,1	1,0
C’	0	1	0 X	1	1
C	1	1	1	0	0

AC’

A’B’ A’C

If we treat the “X” as 0?

F(A,B,C)=A’B’+A’C+AC’

K-Map with “Don’t Care”s

	A’B’	A’B	AB	AB’
	0,0	0,1	1,1	1,0
C’ 0	1	X 1	1	1
C 1	1	1	0	0

C’

A’

If we treat the “X” as 1?

F(A,B,C) = C’ + A’

K-Map with "Don't Care"s

You can treat "X" as either 0 or 1
— depending on which is more advantageous

		A'B'	A'B	AB	AB'
		0,0	0,1	1,1	1,0
C'	0	1	0 X 1	1	1
C	1	1	1	0	0

AC'
C'

A'B' A' A'C

If we treat the "X" as 0?

F(A,B,C)=A'B'+A'C+AC'

If we treat the "X" as 1?

F(A,B,C) = C' + A'

4-input K-Maps with Don't Cares

- How many of the following could be a valid for the given K-map?
  1. C'D + A'CD + A'BC
  2. C'D + A'D + A'BC
  3. C'D + A'CD
  4. A'D + B'C'D + A'BCD'

  A. 0
  B. 1
  C. 2
  D. 3
  E. 4

		A'B'	A'B	AB	AB'
		00	01	11	10
C'D'	00	0	0	0	0
C'D	01	1	1	x	1
CD	11	1	x	0	0
CD'	10	0	1	0	0

4-input K-Maps with Don’t Cares

- How many of the following could be a valid for the given K-map?

  1. C’D + A’CD + A’BC
  2. C’D + A’D + A’BC
  3. ~~3~~ C’D + A’CD
  4. A’D + B’C’D + A’BCD’

  A. 0
  B. 1
  C. 2
  D. 3
  E. 4

		A’B’	A’B	AB	AB’
		00	01	11	10
C’D’	00	0	0	0	0
C’D	01	1	1	x	1
CD	11	1	x	0	0
CD’	10	0	1	0	0

A’D

B’C’D

C’D

A’CD

A’BC

A’BCD’