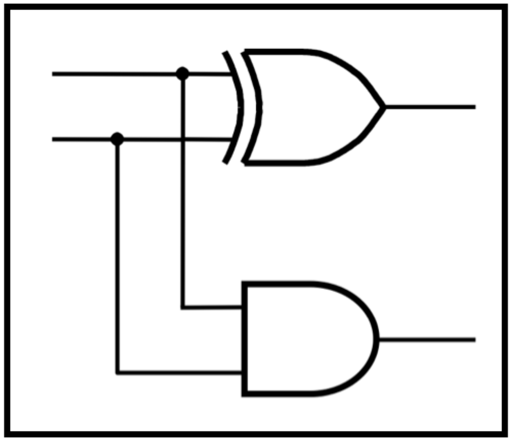


CprE 281: Digital Logic

Instructor: Alexander Stoytchev

http://www.ece.iastate.edu/~alexs/classes/

Karnaugh Maps

CprE 281: Digital Logic
Iowa State University, Ames, IA

Administrative Stuff

- **HW3 is due today**

Administrative Stuff

- **HW4 is out**
- **It is due on Monday Sep 23 @ 10 pm**
- **It is posted on the class web page**
- **I also sent you an e-mail with the link.**

TA Office Hours

- **Hanif Lashari:** **Mondays 2:10 – 3:10pm**
- **Le Zhang:** **Wednesdays 11 am – 1pm**
- **Himani Kohli:** **Thursdays 9 – 10 am**

Go to the Transformative Learning Area (TLA) on the first floor in Coover Hall. Look for a sign that says “CprE 2810 TA.”

Administrative Stuff

- Midterm Exam #1
- When: Friday Sep 27.
- What: Chapter 1 and Chapter 2 plus number systems
- The exam will be closed book but open notes
(you can bring up to 3 pages of handwritten notes).
- Sample exams are posted on the class web page.
- More details to follow.

Quick Review

Do You Still Remember This Boolean Algebra Theorem?

14a. $x \cdot y + x \cdot \overline{y} = x$ Combining

14b. $(x + y) \cdot (x + \overline{y}) = x$

Let's prove 14.a

x	y	$x \cdot y + x \cdot \overline{y} = x$
0	0	
0	1	
1	0	
1	1	

Let’s prove 14.a

x	y	$\mathbf{x \cdot y + x \cdot \overline{y} = x}$
0	0	0
0	1	0
1	0	0
1	1	1

Let’s prove 14.a

x	y	$\mathbf{x \cdot y}$	$+$	$\mathbf{x \cdot \bar{y}}$	$=$	$\mathbf{x}$
0	0	0		0		
0	1	0		0		
1	0	0		1		
1	1	1		0		

Let’s prove 14.a

x	y	$\mathbf{x} \bullet \mathbf{y}$	$+$	$\mathbf{x} \bullet \overline{\mathbf{y}}$	$=$	$\mathbf{x}$
0	0	0	0	0		
0	1	0	0	0		
1	0	0	1	1		
1	1	1	1	0		

Let’s prove 14.a

x	y	$x \cdot y$	$+$	$x \cdot \overline{y}$	$= x$
0	0	0	0	0	0
0	1	0	0	0	0
1	0	0	1	1	1
1	1	1	1	0	1

Let's prove 14.a

x	y	$x \cdot y$	$x \cdot y + x \cdot \overline{y}$	$x \cdot \overline{y}$	$= x$
0	0	0	0	0	0
0	1	0	0	0	0
1	0	0	1	1	1
1	1	1	1	0	1

They are equal.

Motivation

A	B	C	D	F
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	1
1	1	0	0	1
1	1	0	1	0
1	1	1	0	1
1	1	1	1	0

An approach for simplifying logic expressions.

How do we guarantee that we have reached the minimum SOP/POS representation?

This method was described in 1953

M. Karnaugh, "The map method for synthesis of combinational logic circuits"

Transactions of the American Institute of Electrical Engineers, Part I: Communication and Electronics (pages 593 – 599, Volume: 72, Issue: 5, Nov. 1953)

https://ieeexplore.ieee.org/document/6371932

Two-Variable K-Map

Karnaugh Map (K-map)

- **A visual representation of the function**
- **Same information as the truth table**
- **Easier to group minterms**

x_1	x_2	f
0	0	0
0	1	1
1	0	0
1	1	1

(a) Truth table

$x_2 \backslash x_1$	0	1
0	0	0
1	1	1

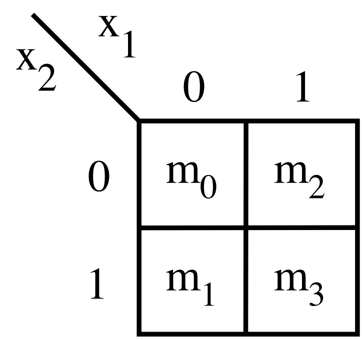
(b) Karnaugh map

Karnaugh Map (K-map)

- **A visual representation of the function**
- **Same information as the truth table**
- **Easier to group minterms**

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

(a) Truth table



(b) Karnaugh map

[Figure 2.49 from the textbook]

Minterms

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

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

Minterm Addition Example

x_1	x_2	f
0	0	0
0	1	1
1	0	0
1	1	1

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

Minterm Addition Example

x_1	x_2	f
0	0	0
0	1	1
1	0	0
1	1	1

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

$$\overline{x}_1 x_2 + x_1 x_2 = x_2$$

Minterm Addition Example

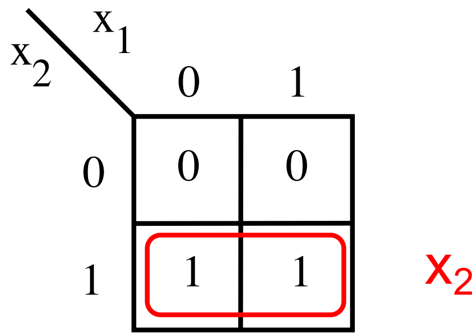
x_1	x_2	f
0	0	0
0	1	1
1	0	0
1	1	1

x_2 \ x_1	0	1
0	0	0
1	1	1

$$\bar{x}_1 x_2 + x_1 x_2 = x_2$$

Minterm Addition Example

x_1	x_2	f
0	0	0
0	1	1
1	0	0
1	1	1



$$\overline{x}_1 x_2 + x_1 x_2 = x_2$$

Another Grouping Example

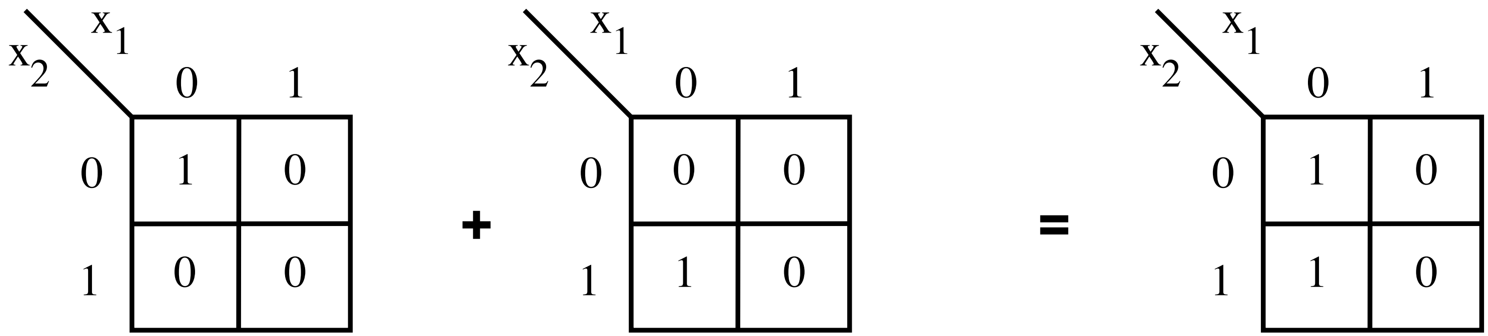
$x_2 \backslash x_1$	0	1
0	1	0
1	0	0

m_0

$x_2 \backslash x_1$	0	1
0	0	0
1	1	0

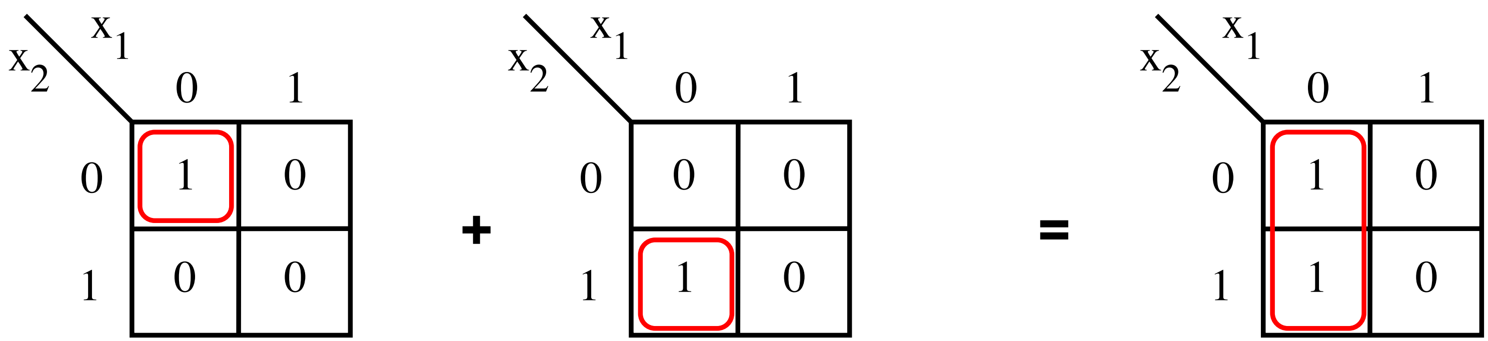
m_1

Another Grouping Example



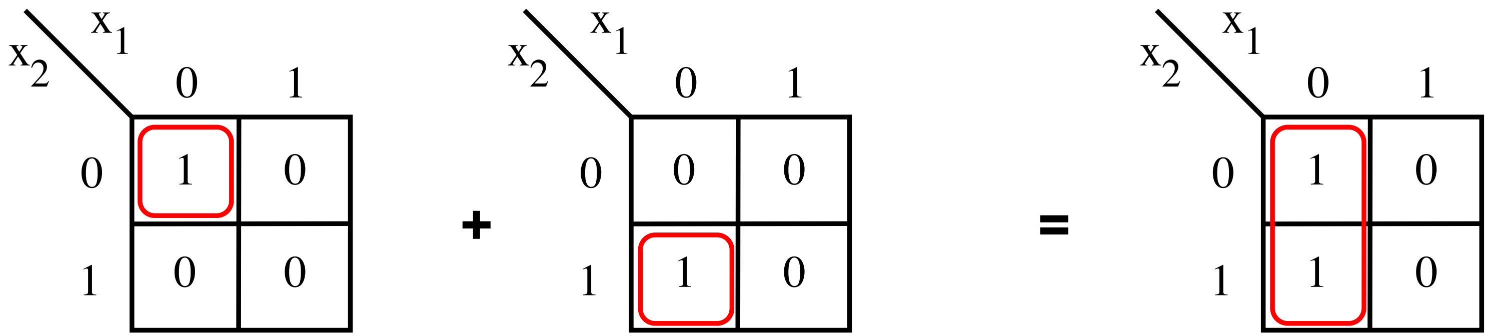
m_0 + m_1 = $m_0 + m_1$

Another Grouping Example



m_0 + m_1 = $m_0 + m_1$

Another Grouping Example



$$m_0 \quad + \quad m_1 \quad = \quad m_0 + m_1$$

$$\bar{x}_1\bar{x}_2 \quad + \quad \bar{x}_1 x_2 \quad = \quad \bar{x}_1$$

Property 14a (Combining)

Grouping Rules

- Group “1”s with rectangles
- Both sides must be a power of 2:
  - 1x1, 1x2, 2x1, 2x2, 1x4, 4x1, 2x4, 4x2, 4x4
- Can use/cover the same minterm more than once
- Can wrap around the edges of the map
- Some rules in selecting groups:
  - Try to use as few groups as possible to cover all “1”s.
  - For each group, try to make it as large as you can (i.e., if you can use a 2x2, don’t use a 2x1 even if that is enough).

Two-Variable K-map

x_1	x_2	f
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

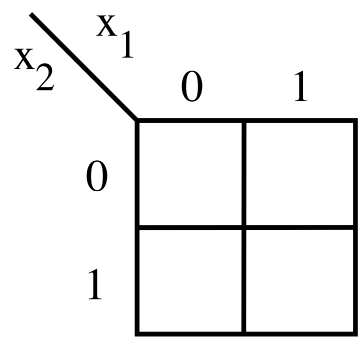
[Figure 2.49 from the textbook]

Step-By-Step Example

x_1	x_2	f
0	0	1
0	1	1
1	0	0
1	1	1

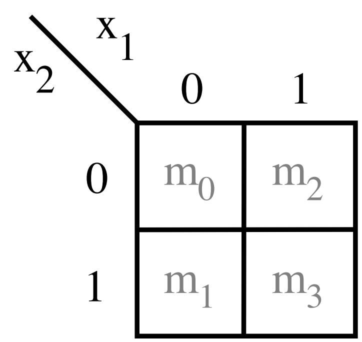
1. Draw The Map

x_1	x_2	f
0	0	1
0	1	1
1	0	0
1	1	1



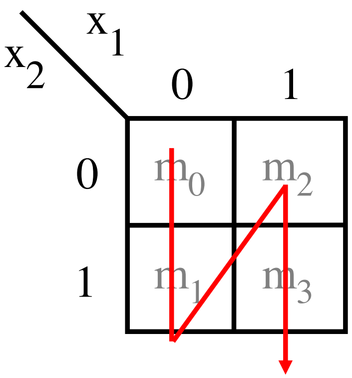
2. Fill The Map

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



2. Fill The Map

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



2. Fill The Map

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

x_2 \ x_1	0	1
0	1	0
1	1	1

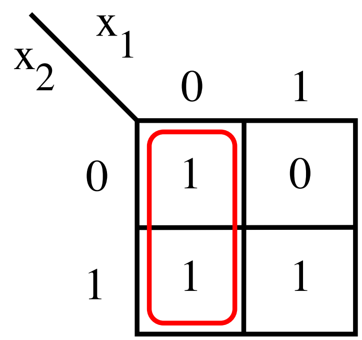
3. Group

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

x_2 \ x_1	0	1
0	1	0
1	1	1

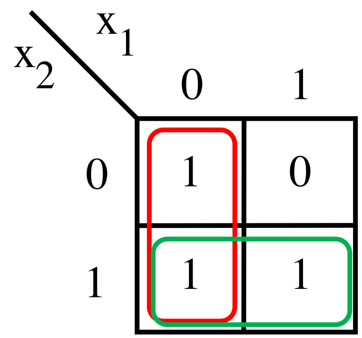
3. Group

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



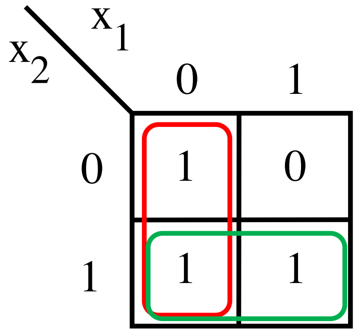
3. Group

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



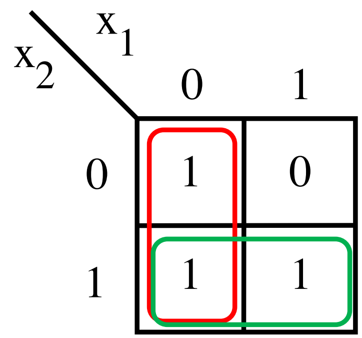
4. Write The Expression

x_1	x_2	f
0	0	1
0	1	1
1	0	0
1	1	1



4. Write The Expression

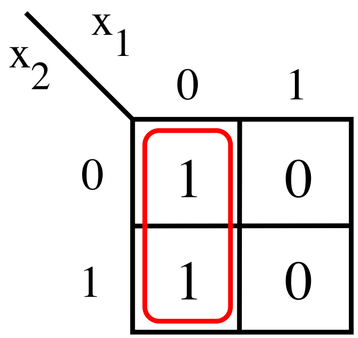
x_1	x_2	f
0	0	1
0	1	1
1	0	0
1	1	1



$\overline{x}_1 + x_2$

Writing The Expression

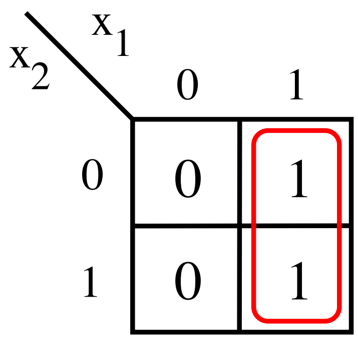
- **Find which variable is constant**



$\overline{x}_1$

Writing The Expression

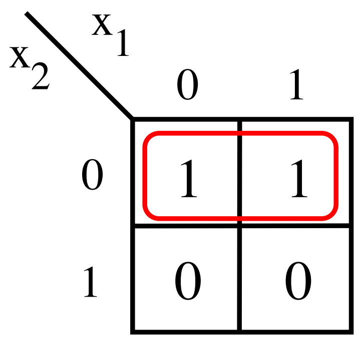
- **Find which variable is constant**



x_1

Writing The Expression

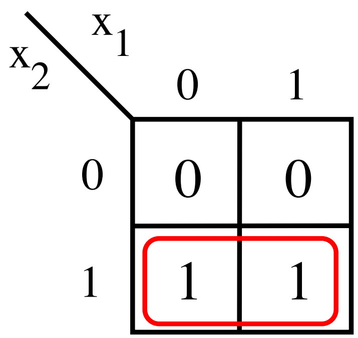
- **Find which variable is constant**



$\overline{x}_2$

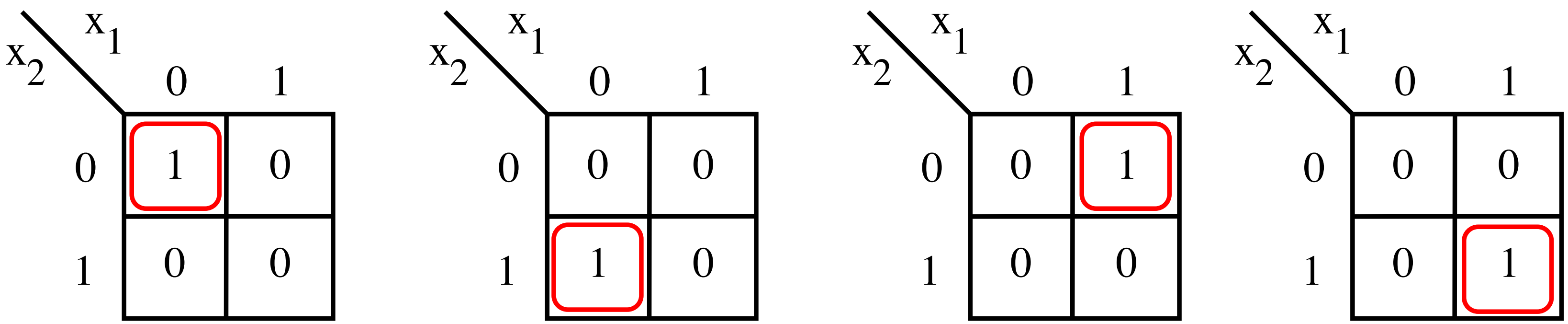
Writing The Expression

- **Find which variable is constant**



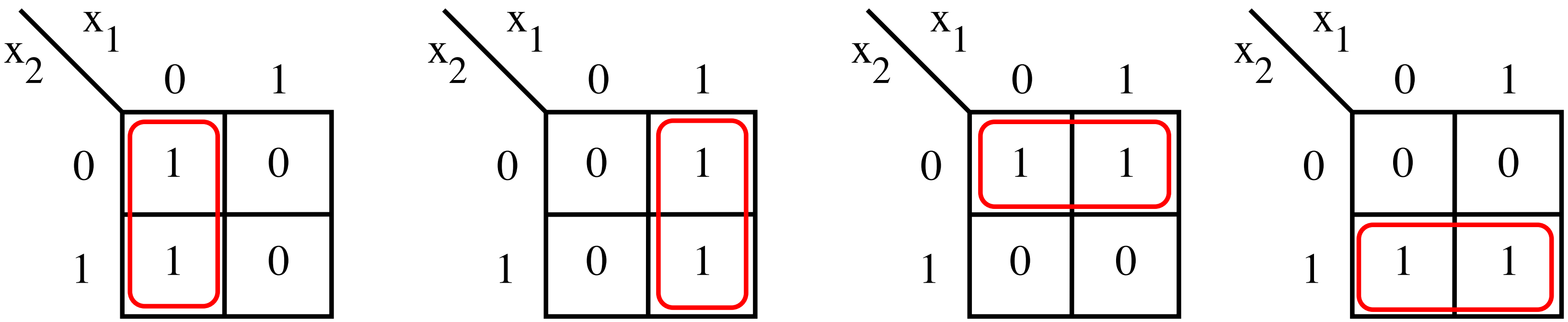
x_2

These are all valid groupings



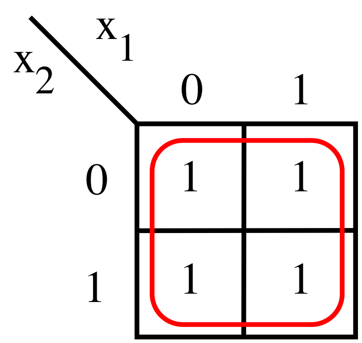
$\bar{x}_1 \bar{x}_2$ $\bar{x}_1 x_2$ $x_1 \bar{x}_2$ $x_1 x_2$

These are all valid groupings



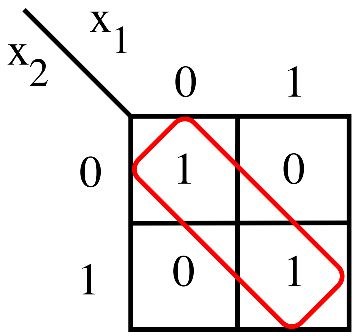
$\overline{x}_1$ x_1 $\overline{x}_2$ x_2

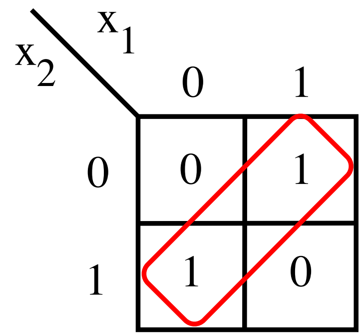
This one is valid too



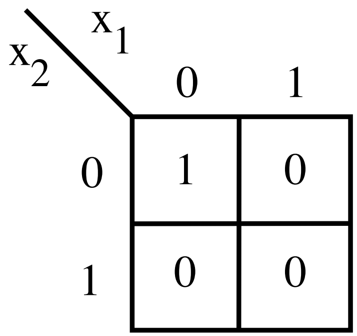
In this case the result is the constant function 1.

Invalid Groupings

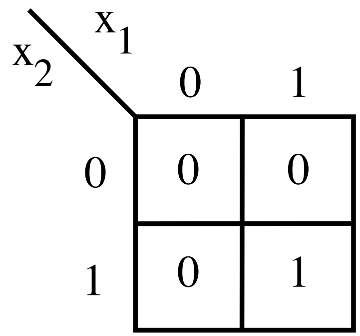




Can’t group diagonally. Why?

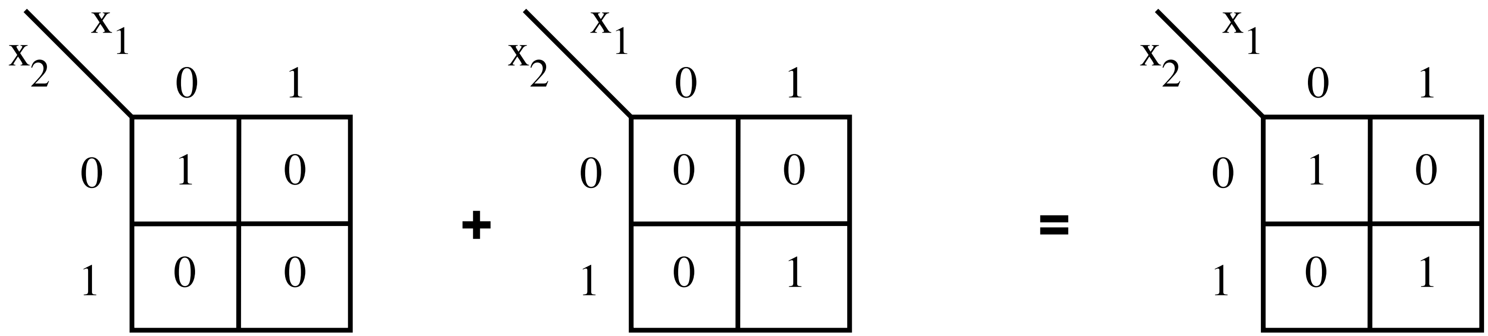


m_0



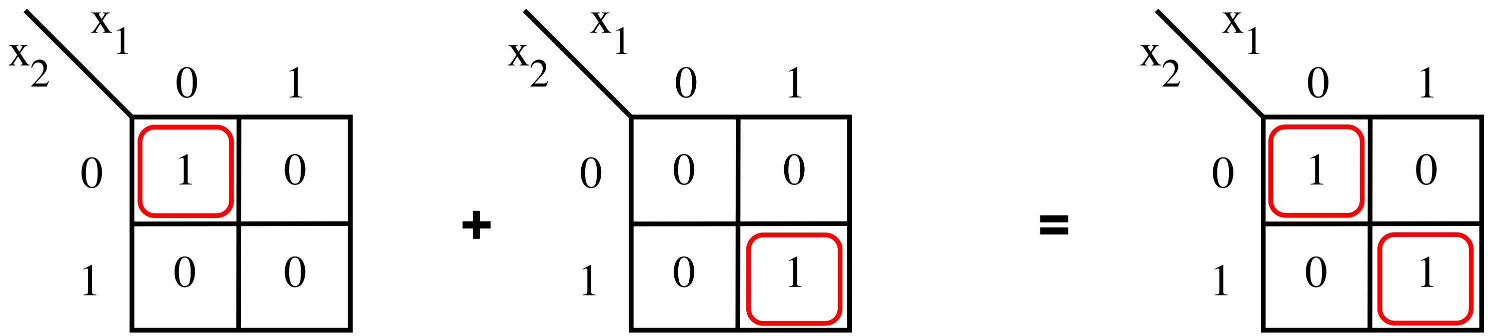
m_3

Can’t group diagonally. Why?



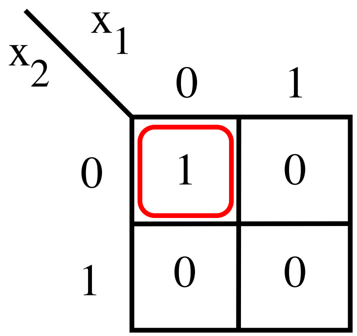
m_0 + m_3 = $m_0 + m_3$

Can’t group diagonally. Why?

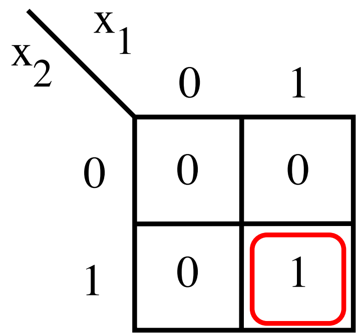


m_0 + m_3 = $m_0 + m_3$

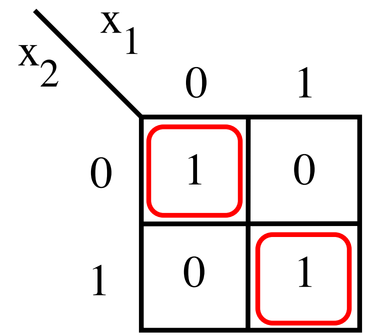
Can't group diagonally. Why?



+



=



$$m_0 + m_3 = m_0 + m_3$$

$$\bar{x}_1\bar{x}_2 + x_1x_2 = \bar{x}_1\bar{x}_2 + x_1x_2$$

We can't use Property 14a here. This can't be simplified.

Three-Variable K-Map

Location of three-variable minterms

x_1	x_2	x_3	
0	0	0	m_0
0	0	1	m_1
0	1	0	m_2
0	1	1	m_3
1	0	0	m_4
1	0	1	m_5
1	1	0	m_6
1	1	1	m_7

(a) Truth table

x_3 \ $x_1 x_2$	00	01	11	10
0	m_0	m_2	m_6	m_4
1	m_1	m_3	m_7	m_5

(b) Karnaugh map

[Figure 2.51 from the textbook]

Location of three-variable minterms

x_1	x_2	x_3	
0	0	0	m_0
0	0	1	m_1
0	1	0	m_2
0	1	1	m_3
1	0	0	m_4
1	0	1	m_5
1	1	0	m_6
1	1	1	m_7

(a) Truth table

x_3 \ $x_1 x_2$	00	01	11	10
0	m_0	m_2	m_6	m_4
1	m_1	m_3	m_7	m_5

(b) Karnaugh map

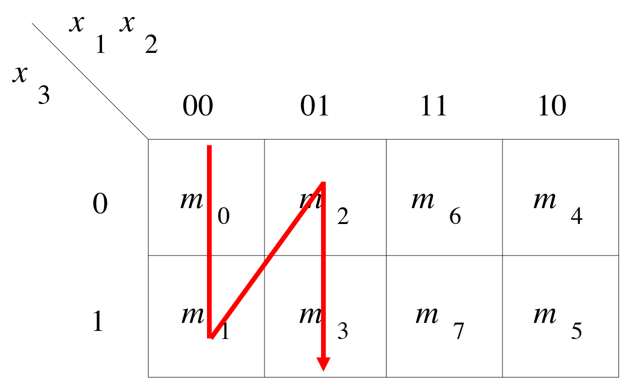
Notice the placement of

- **Variables**
- **Binary pair values**
- **Minterms**

Location of three-variable minterms

x_1	x_2	x_3	
0	0	0	m_0
0	0	1	m_1
0	1	0	m_2
0	1	1	m_3
1	0	0	m_4
1	0	1	m_5
1	1	0	m_6
1	1	1	m_7

(a) Truth table



(b) Karnaugh map

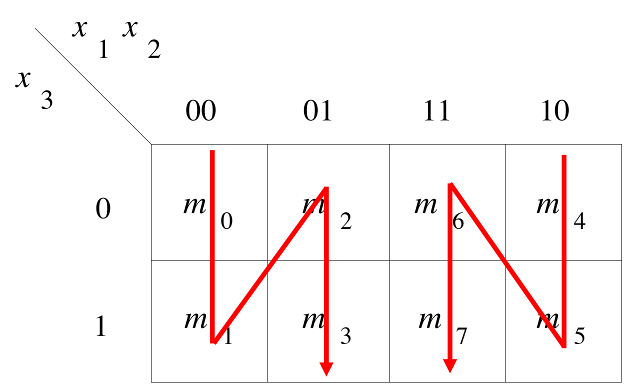
Notice the placement of

- **Variables**
- **Binary pair values**
- **Minterms**

Location of three-variable minterms

x_1	x_2	x_3	
0	0	0	m_0
0	0	1	m_1
0	1	0	m_2
0	1	1	m_3
1	0	0	m_4
1	0	1	m_5
1	1	0	m_6
1	1	1	m_7

(a) Truth table



(b) Karnaugh map

Notice the placement of

- **Variables**
- **Binary pair values**
- **Minterms**

Gray Code

- Sequence of binary codes
- Two neighboring lines vary by only 1 bit

```
00
01
11
10
```

Gray Code

- Sequence of binary codes
- Two neighboring lines vary by only 1 bit

```
000
001
011
010
110
111
101
100
```

Gray Code

- **Sequence of binary codes**
- **Two neighboring lines vary by only 1 bit**

```
000
001
011
010
110
111
101
100
```

Gray Code

- Sequence of binary codes
- Two neighboring lines vary by only 1 bit

```
000
001
011
010
110
111
101
100
```

Gray Code

- Sequence of binary codes
- Two neighboring lines vary by only 1 bit

```
000
001
011
010
110
111
101
100
```

Gray Code

- Sequence of binary codes
- Two neighboring lines vary by only 1 bit

```
000
001
011
010
110
111
101
100
```

Gray Code

- Sequence of binary codes
- Two neighboring lines vary by only 1 bit

```
000
001
011
010
110
111
101
100
```

Gray Code

- Sequence of binary codes
- Two neighboring lines vary by only 1 bit

```
000
001
011
010
110
111
101
100
```

Gray Code

- Sequence of binary codes
- Two neighboring lines vary by only 1 bit

```
000
001
011
010
110
111
101
100
```

Gray Code & K-map

	$s\ x_1\ x_2$
m_0	0 0 0
m_1	0 0 1
m_2	0 1 0
m_3	0 1 1
m_4	1 0 0
m_5	1 0 1
m_6	1 1 0
m_7	1 1 1

x_2 \ $s\ x_1$	00	01	11	10
0	m_0	m_2	m_6	m_4
1	m_1	m_3	m_7	m_5

Gray Code & K-map

	$s\ x_1\ x_2$
m_0	0 0 0
m_1	0 0 1
m_2	0 1 0
m_3	0 1 1
m_4	1 0 0
m_5	1 0 1
m_6	1 1 0
m_7	1 1 1

x_2 \ $s\ x_1$	00	01	11	10
0	000	010	110	100
1	001	011	111	101

Gray Code & K-map

	$s\ x_1\ x_2$
m_0	0 0 0
m_1	0 0 1
m_2	0 1 0
m_3	0 1 1
m_4	1 0 0
m_5	1 0 1
m_6	1 1 0
m_7	1 1 1

x_2 \ $s\ x_1$	00	01	11	10
0	000	010	110	100
1	001	011	111	101

Gray Code & K-map

	$s\ x_1\ x_2$
m_0	0 0 0
m_1	0 0 1
m_2	0 1 0
m_3	0 1 1
m_4	1 0 0
m_5	1 0 1
m_6	1 1 0
m_7	1 1 1

x_2 \ $s\ x_1$	00	01	11	10
0	000	010	110	100
1	001	011	111	101

Gray Code & K-map

	$s\ x_1\ x_2$
m_0	0 0 0
m_1	0 0 1
m_2	0 1 0
m_3	0 1 1
m_4	1 0 0
m_5	1 0 1
m_6	1 1 0
m_7	1 1 1

x_2 \ $s\ x_1$	00	01	11	10
0	000	010	110	100
1	001	011	111	101

Gray Code & K-map

	$s\ x_1\ x_2$
m_0	0 0 0
m_1	0 0 1
m_2	0 1 0
m_3	0 1 1
m_4	1 0 0
m_5	1 0 1
m_6	1 1 0
m_7	1 1 1

x_2 \ $s\ x_1$	00	01	11	10
0	000	010	110	100
1	001	011	111	101

Gray Code & K-map

	$s\ x_1\ x_2$
m_0	0 0 0
m_1	0 0 1
m_2	0 1 0
m_3	0 1 1
m_4	1 0 0
m_5	1 0 1
m_6	1 1 0
m_7	1 1 1

x_2 \ $s\ x_1$	00	01	11	10
0	000	010	110	100
1	001	011	111	101

These two neighbors
differ only in the LAST bit

Gray Code & K-map

	$s\ x_1\ x_2$
m_0	0 0 0
m_1	0 0 1
m_2	0 1 0
m_3	0 1 1
m_4	1 0 0
m_5	1 0 1
m_6	1 1 0
m_7	1 1 1

x_2 \ $s\ x_1$	00	01	11	10
0	000	010	110	100
1	001	011	111	101

These two neighbors
differ only in the LAST bit

Gray Code & K-map

	$s\ x_1\ x_2$
m_0	0 0 0
m_1	0 0 1
m_2	0 1 0
m_3	0 1 1
m_4	1 0 0
m_5	1 0 1
m_6	1 1 0
m_7	1 1 1

x_2 \ $s\ x_1$	00	01	11	10
0	000	010	110	100
1	001	011	111	101

These two neighbors differ only in the FIRST bit

Gray Code & K-map

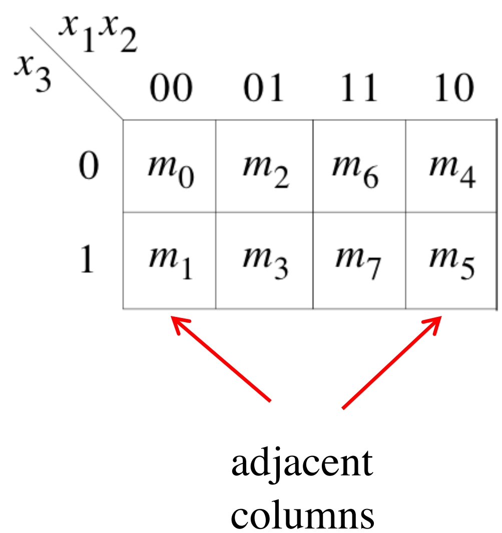
	$s\ x_1\ x_2$
m_0	0 0 0
m_1	0 0 1
m_2	0 1 0
m_3	0 1 1
m_4	1 0 0
m_5	1 0 1
m_6	1 1 0
m_7	1 1 1

x_2 \ $s\ x_1$	00	01	11	10
0	000	010	110	100
1	001	011	111	101

These four neighbors
differ in the FIRST and LAST bit

They are similar in their MIDDLE bit

Adjacency Rules



Gray Code & K-map

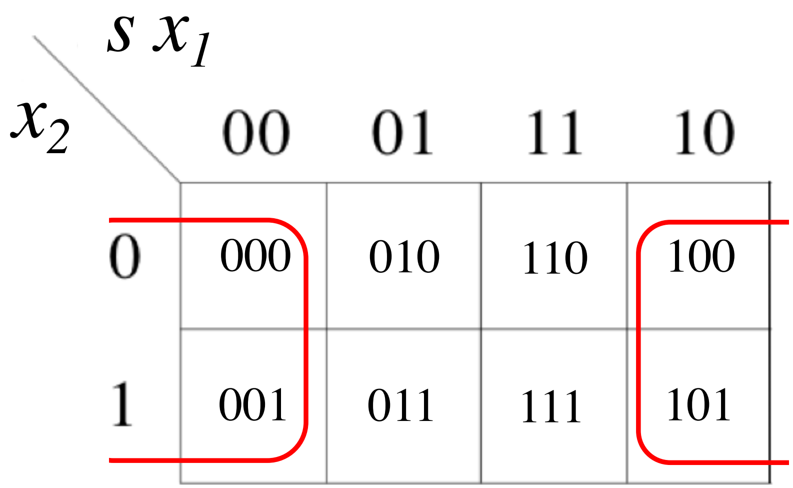
	$s\ x_1\ x_2$
m_0	0 0 0
m_1	0 0 1
m_2	0 1 0
m_3	0 1 1
m_4	1 0 0
m_5	1 0 1
m_6	1 1 0
m_7	1 1 1

x_2 \ $s\ x_1$	00	01	11	10
0	000	010	110	100
1	001	011	111	101

These two neighbors
differ only in the FIRST bit

Gray Code & K-map

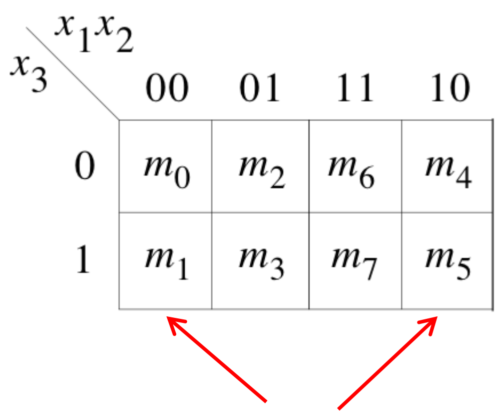
	$s\ x_1\ x_2$
m_0	0 0 0
m_1	0 0 1
m_2	0 1 0
m_3	0 1 1
m_4	1 0 0
m_5	1 0 1
m_6	1 1 0
m_7	1 1 1



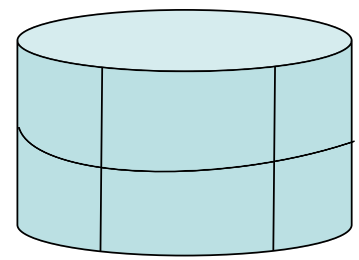
These four neighbors
differ in the FIRST and LAST bit

They are similar in their MIDDLE bit

Adjacency Rules

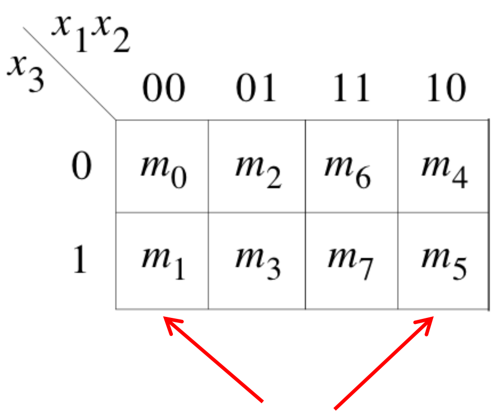


adjacent
columns

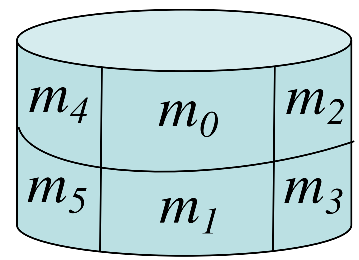


As if the K-map were
drawn on a cylinder

Adjacency Rules

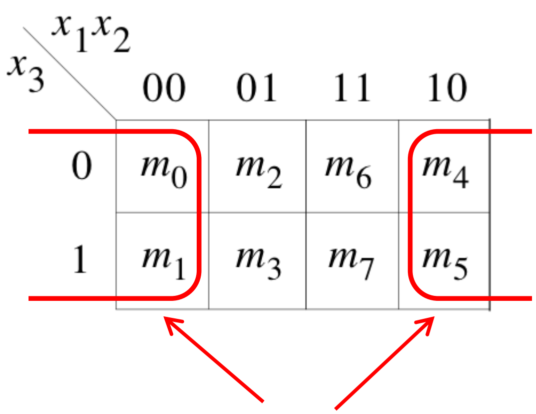


adjacent
columns

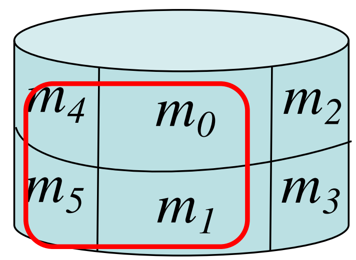


As if the K-map were
drawn on a cylinder

Adjacency Rules

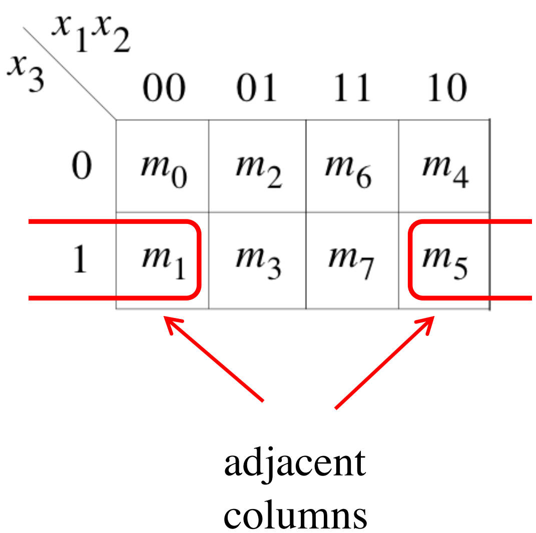


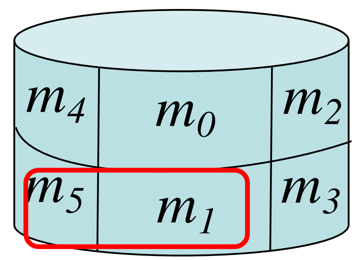
adjacent
columns



As if the K-map were
drawn on a cylinder

Adjacency Rules





As if the K-map were drawn on a cylinder

Examples of Valid Groupings

Groupings and Expressions

x_3 \ x_1x_2	00	01	11	10
0	1	0	0	0
1	0	0	0	0

$\bar{x}_1 \bar{x}_2 \bar{x}_3$

x_3 \ x_1x_2	00	01	11	10
0	0	1	0	0
1	0	0	0	0

$\bar{x}_1 x_2 \bar{x}_3$

x_3 \ x_1x_2	00	01	11	10
0	0	0	1	0
1	0	0	0	0

$x_1 x_2 \bar{x}_3$

x_3 \ x_1x_2	00	01	11	10
0	0	0	0	1
1	0	0	0	0

$x_1 \bar{x}_2 \bar{x}_3$

Groupings and Expressions

x_3 \ x_1x_2	00	01	11	10
0	0	0	0	0
1	1	0	0	0

$\overline{x}_1 \overline{x}_2 x_3$

x_3 \ x_1x_2	00	01	11	10
0	0	0	0	0
1	0	1	0	0

$\overline{x}_1 x_2 x_3$

x_3 \ x_1x_2	00	01	11	10
0	0	0	0	0
1	0	0	1	0

$x_1 x_2 x_3$

x_3 \ x_1x_2	00	01	11	10
0	0	0	0	0
1	0	0	0	1

$x_1 \overline{x}_2 x_3$

Groupings and Expressions

x_3 \ x_1x_2	00	01	11	10
0	1	0	0	0
1	1	0	0	0

$\bar{x}_1 \bar{x}_2$

x_3 \ x_1x_2	00	01	11	10
0	0	1	0	0
1	0	1	0	0

$\bar{x}_1 x_2$

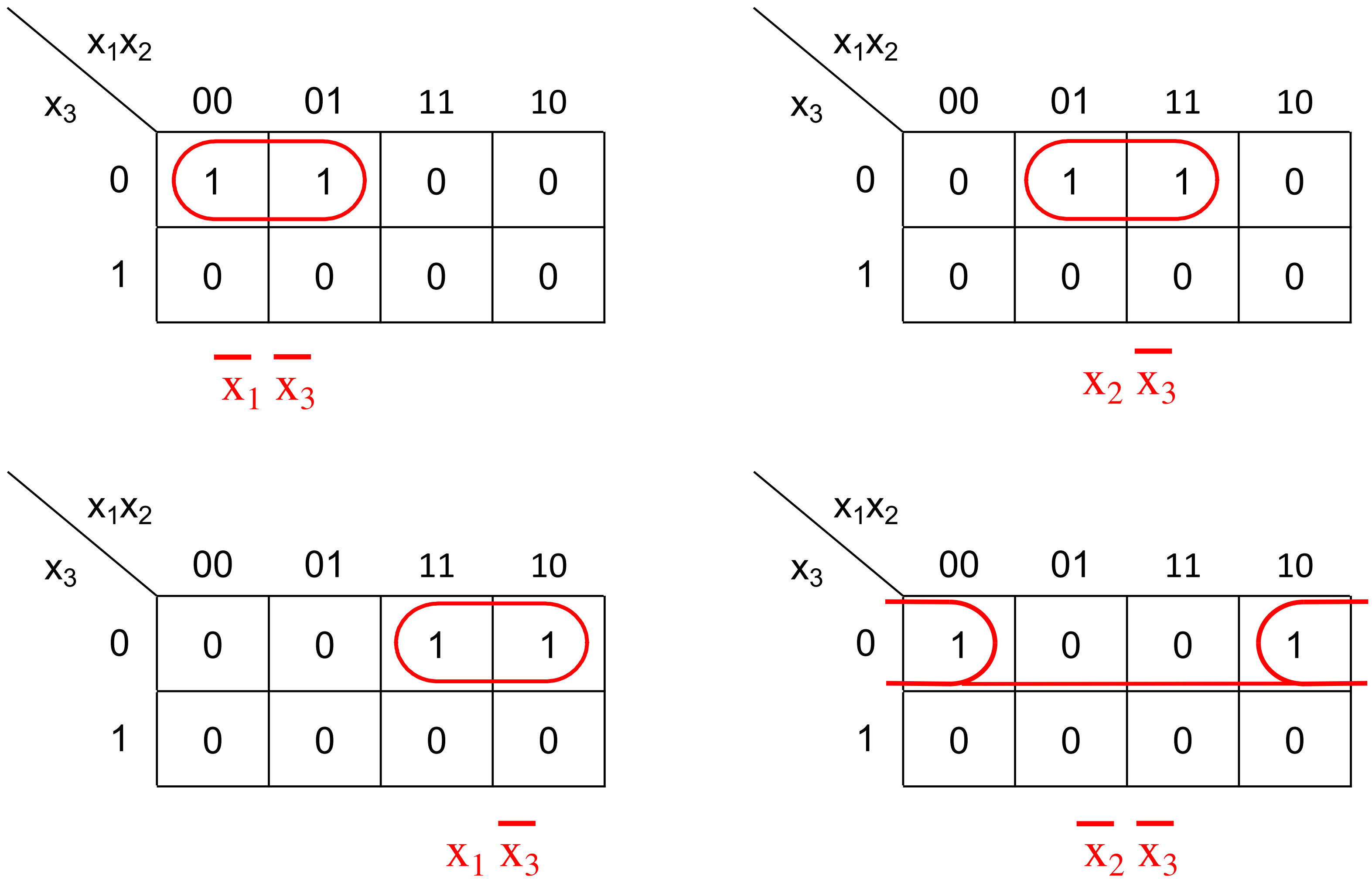
x_3 \ x_1x_2	00	01	11	10
0	0	0	1	0
1	0	0	1	0

$x_1 x_2$

x_3 \ x_1x_2	00	01	11	10
0	0	0	0	1
1	0	0	0	1

$x_1 \bar{x}_2$

Groupings and Expressions



Groupings and Expressions

x_3 \ x_1x_2	00	01	11	10
0	0	0	0	0
1	1	1	0	0

$\bar{x}_1 x_3$

x_3 \ x_1x_2	00	01	11	10
0	0	0	0	0
1	0	1	1	0

$x_2 x_3$

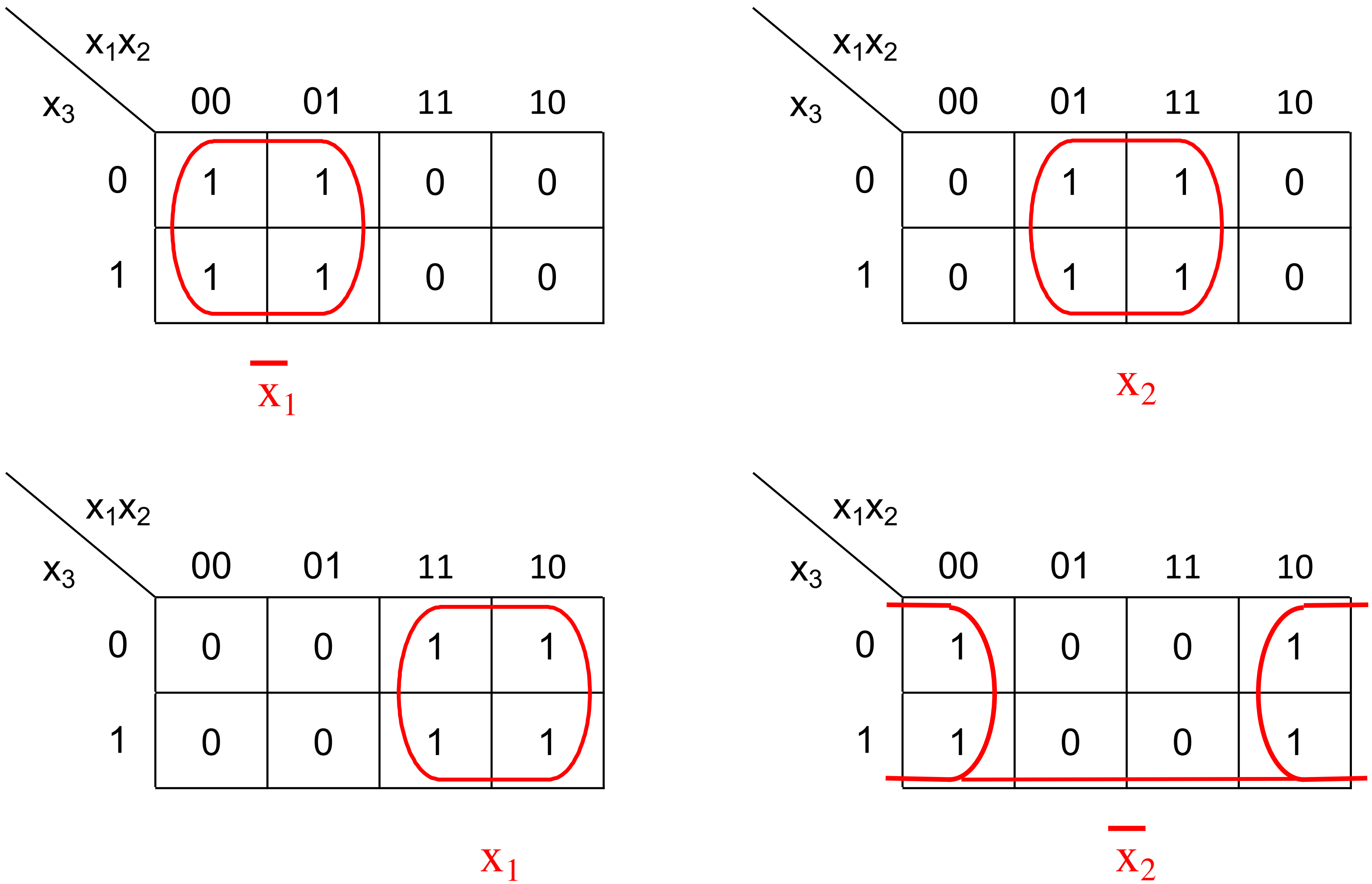
x_3 \ x_1x_2	00	01	11	10
0	0	0	0	0
1	0	0	1	1

$x_1 x_3$

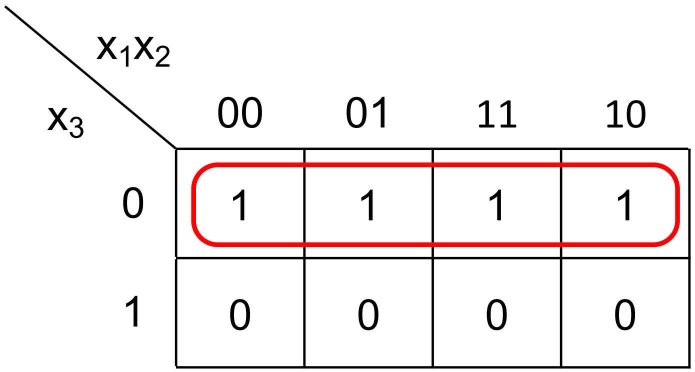
x_3 \ x_1x_2	00	01	11	10
0	0	0	0	0
1	1	0	0	1

$\bar{x}_2 x_3$

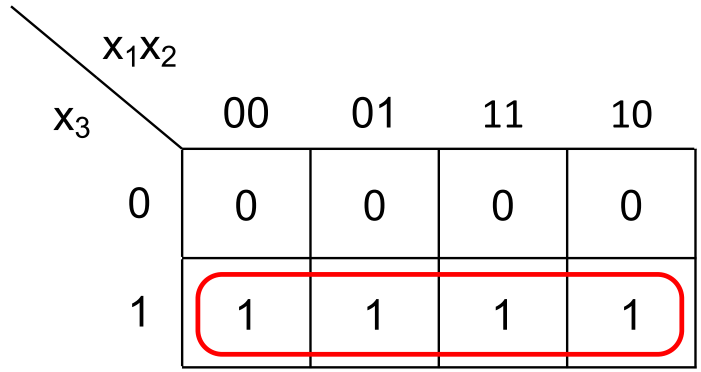
Groupings and Expressions



Groupings and Expressions

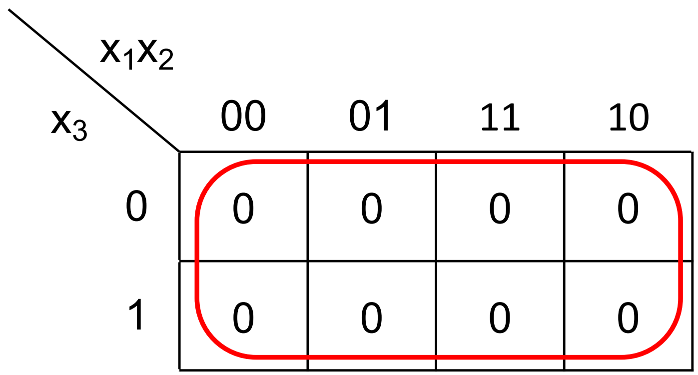


$\overline{x}_3$

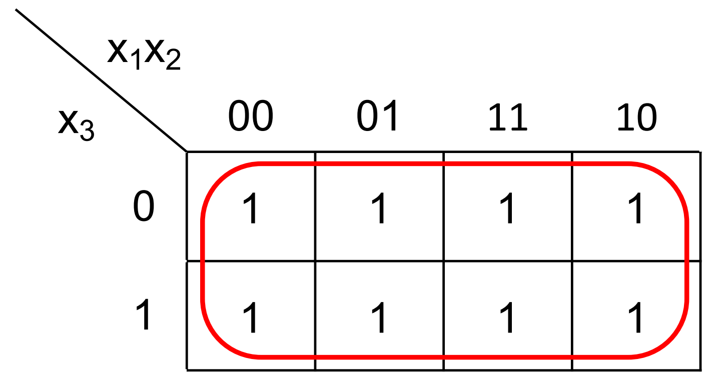


x_3

Groupings and Expressions



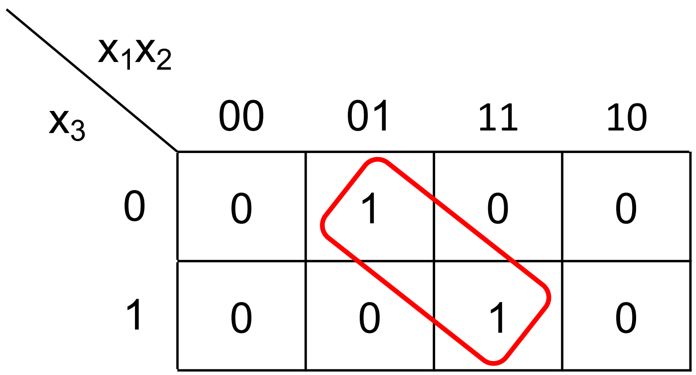
0

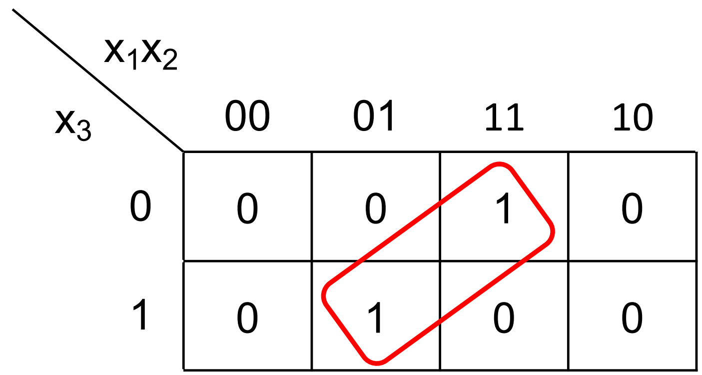


1

Examples of Invalid Groupings

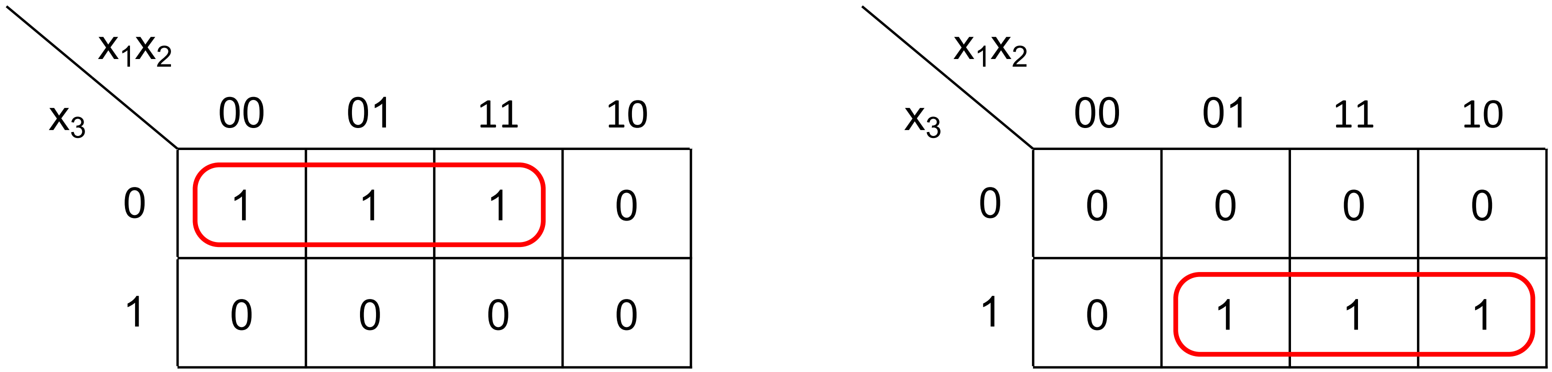
Some Invalid Groupings





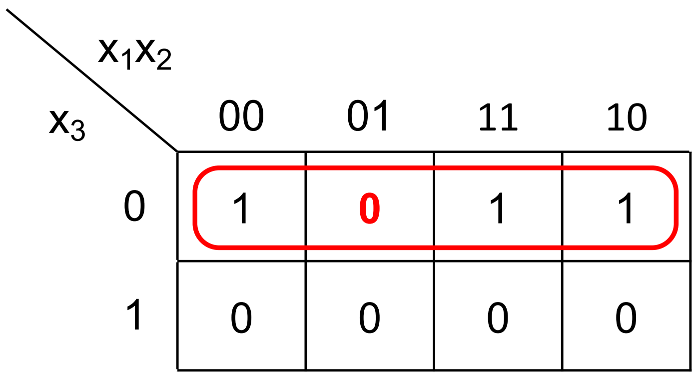
Can’t group diagonally.

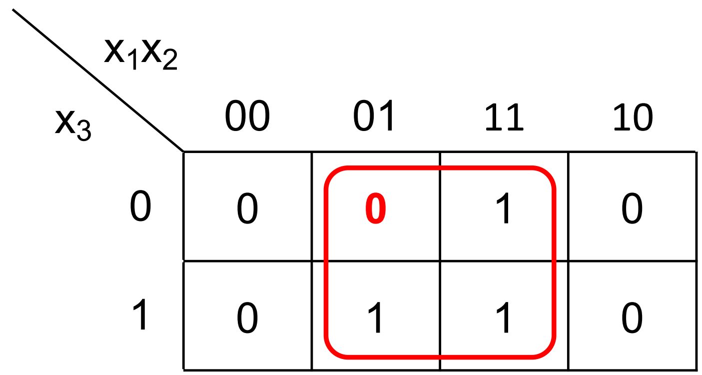
Some Invalid Groupings



Can’t group three in a row.
Each side must be a power of 2.

Some Invalid Groupings

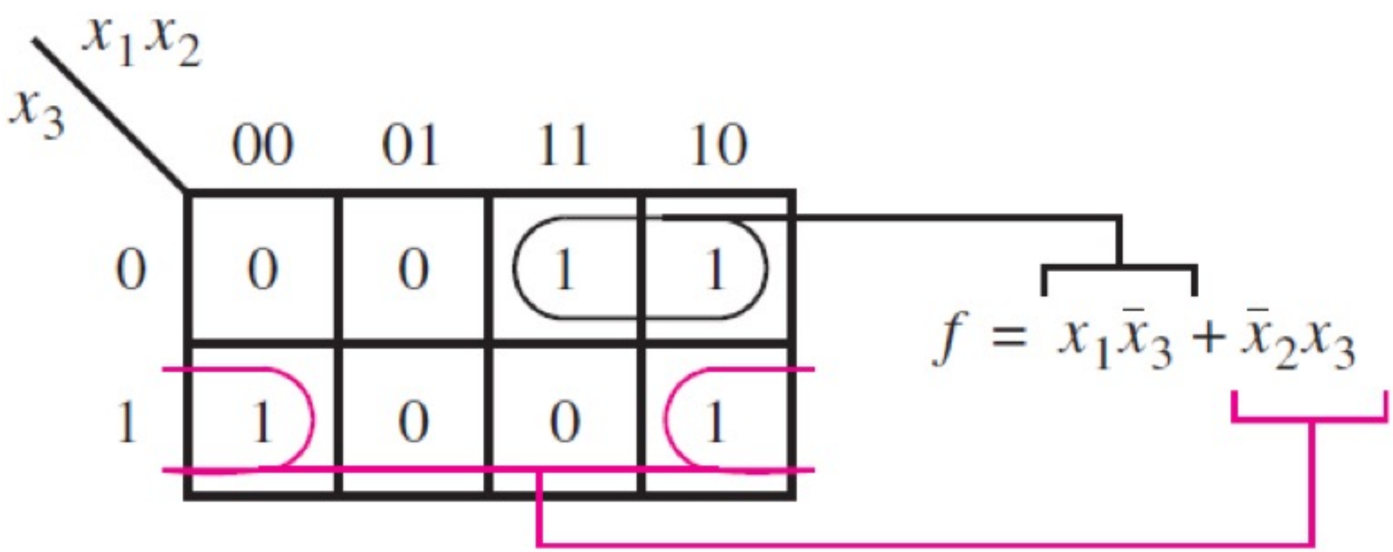




Can’t group zeros and ones together.

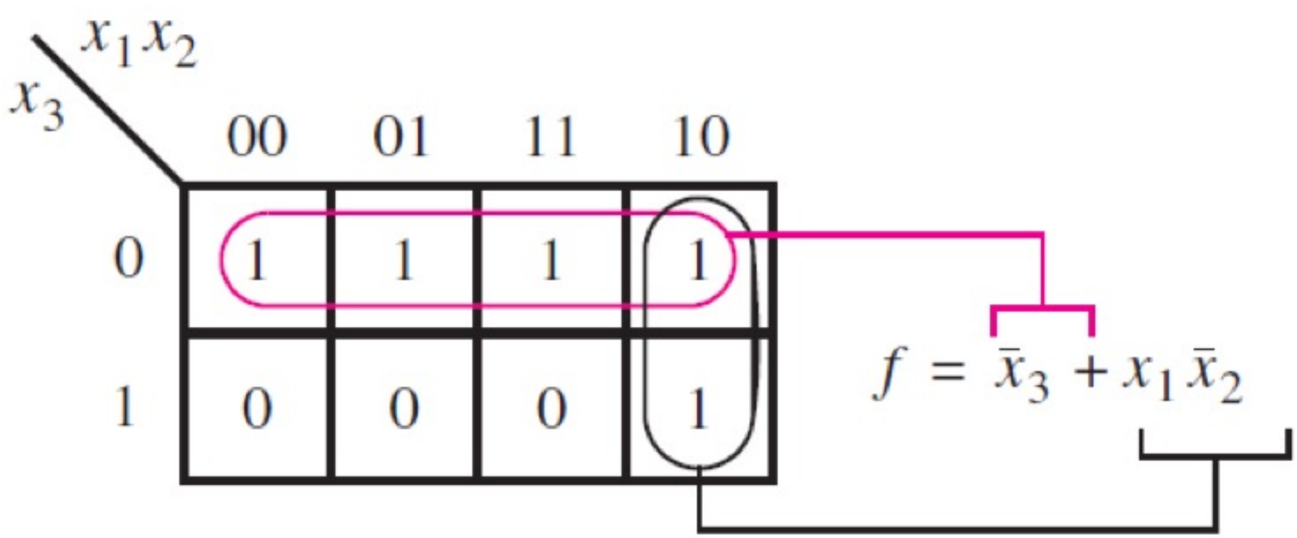
Minimization Examples with 3-variable K-Maps

Examples of three-variable Karnaugh maps



[Figure 2.52a from the textbook]

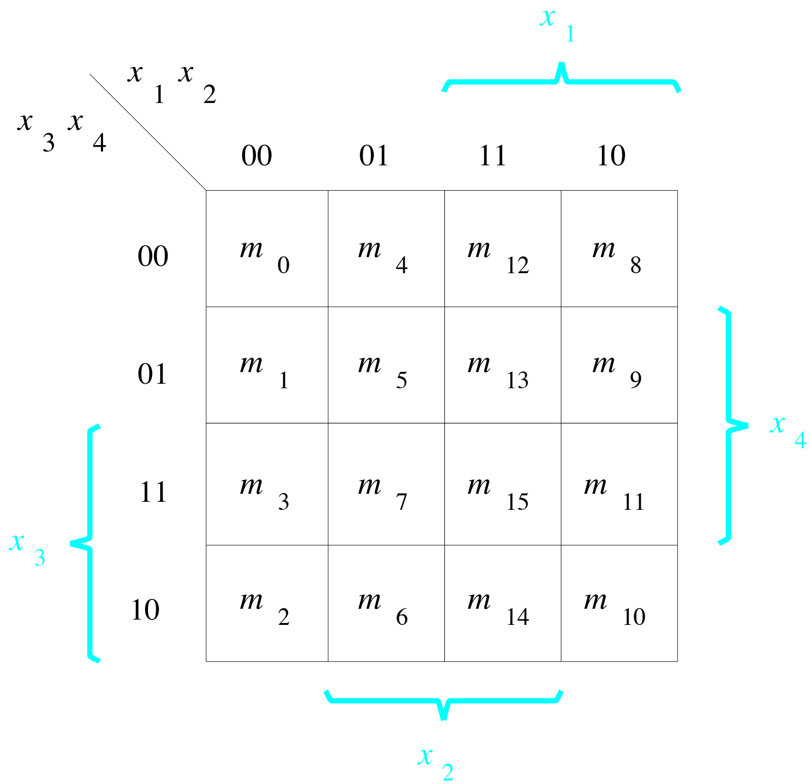
Examples of three-variable Karnaugh maps



[Figure 2.52b from the textbook]

Four-Variable K-Map

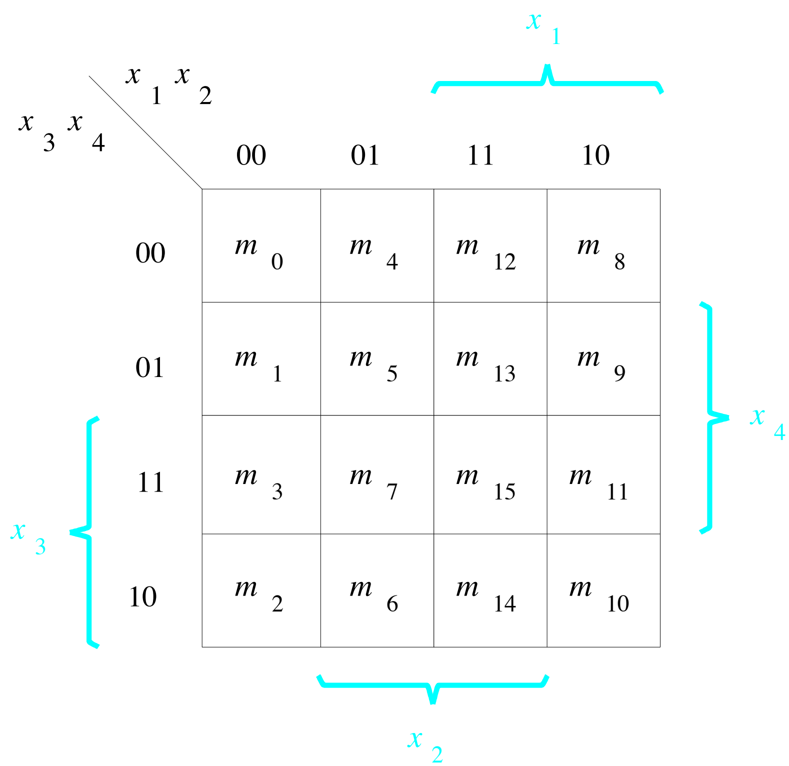
A four-variable Karnaugh map



[Figure 2.53 from the textbook]

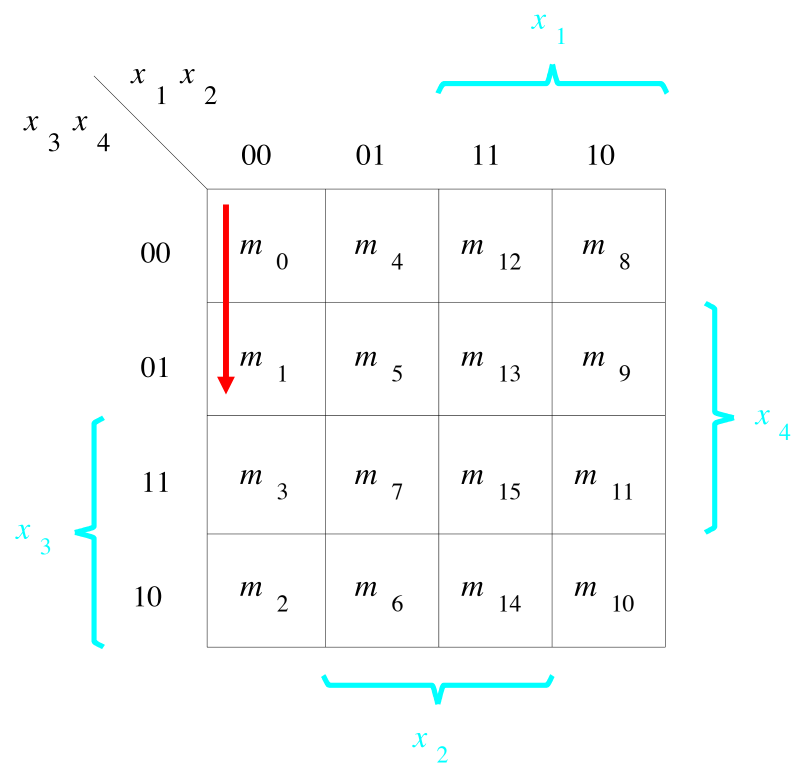
A four-variable Karnaugh map

x1	x2	x3	x4	
0	0	0	0	m0
0	0	0	1	m1
0	0	1	0	m2
0	0	1	1	m3
0	1	0	0	m4
0	1	0	1	m5
0	1	1	0	m6
0	1	1	1	m7
1	0	0	0	m8
1	0	0	1	m9
1	0	1	0	m10
1	0	1	1	m11
1	1	0	0	m12
1	1	0	1	m13
1	1	1	0	m14
1	1	1	1	m15



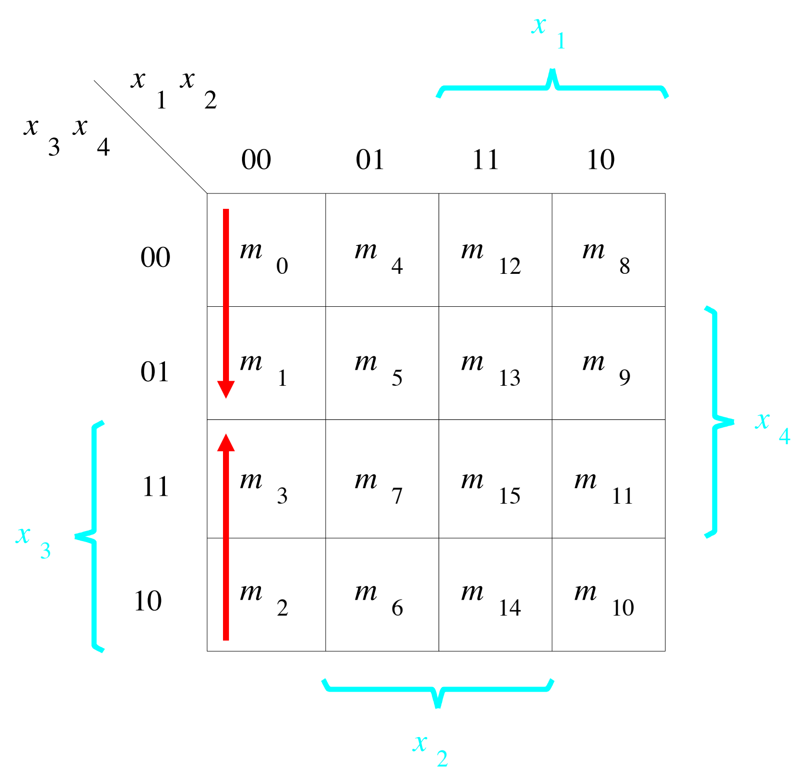
A four-variable Karnaugh map

x1	x2	x3	x4	
0	0	0	0	m0
0	0	0	1	m1
0	0	1	0	m2
0	0	1	1	m3
0	1	0	0	m4
0	1	0	1	m5
0	1	1	0	m6
0	1	1	1	m7
1	0	0	0	m8
1	0	0	1	m9
1	0	1	0	m10
1	0	1	1	m11
1	1	0	0	m12
1	1	0	1	m13
1	1	1	0	m14
1	1	1	1	m15



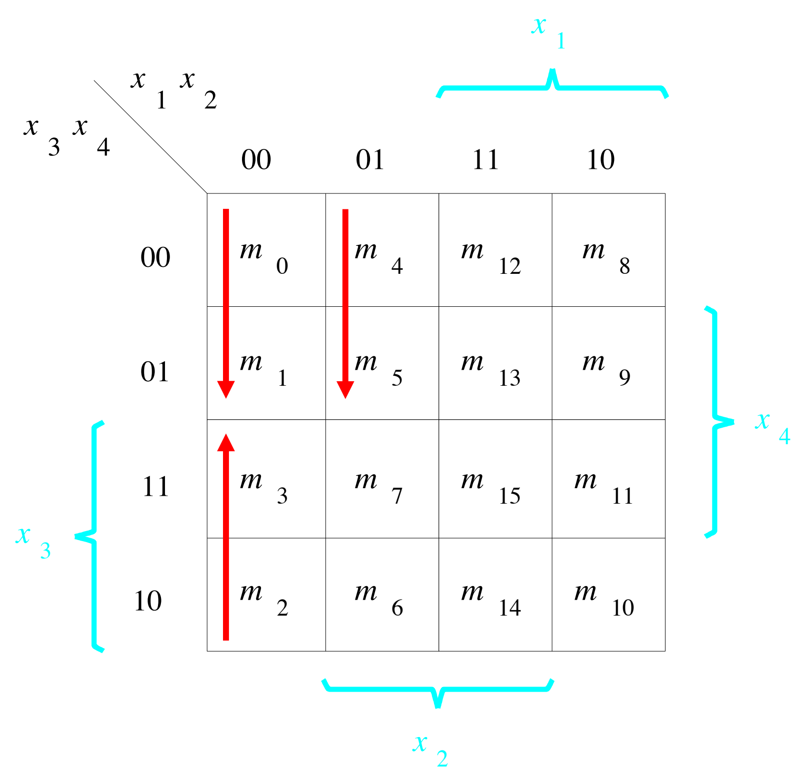
A four-variable Karnaugh map

x1	x2	x3	x4	
0	0	0	0	m0
0	0	0	1	m1
0	0	1	0	m2
0	0	1	1	m3
0	1	0	0	m4
0	1	0	1	m5
0	1	1	0	m6
0	1	1	1	m7
1	0	0	0	m8
1	0	0	1	m9
1	0	1	0	m10
1	0	1	1	m11
1	1	0	0	m12
1	1	0	1	m13
1	1	1	0	m14
1	1	1	1	m15



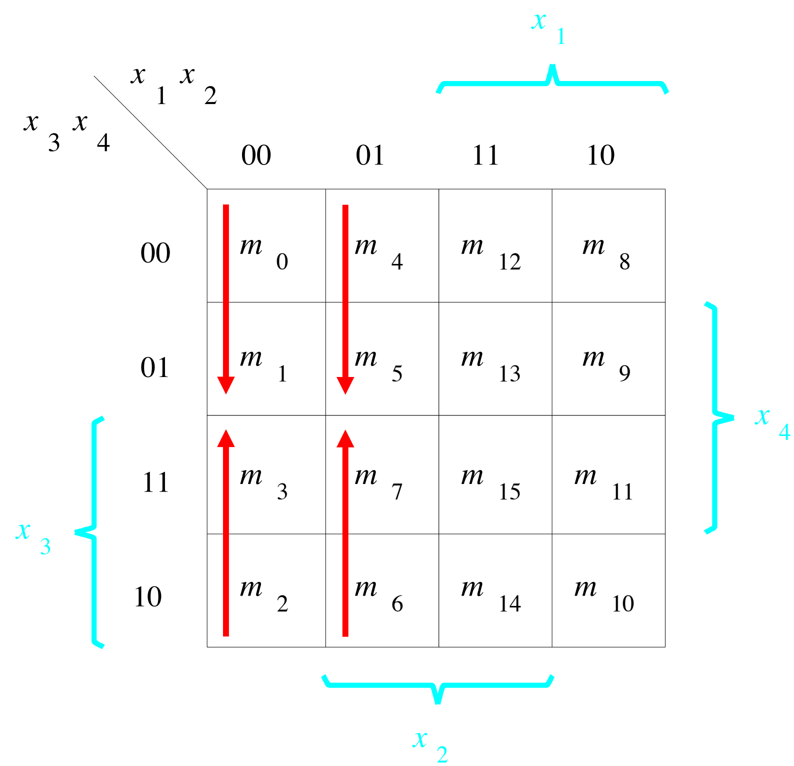
A four-variable Karnaugh map

x1	x2	x3	x4	
0	0	0	0	m0
0	0	0	1	m1
0	0	1	0	m2
0	0	1	1	m3
0	1	0	0	m4
0	1	0	1	m5
0	1	1	0	m6
0	1	1	1	m7
1	0	0	0	m8
1	0	0	1	m9
1	0	1	0	m10
1	0	1	1	m11
1	1	0	0	m12
1	1	0	1	m13
1	1	1	0	m14
1	1	1	1	m15



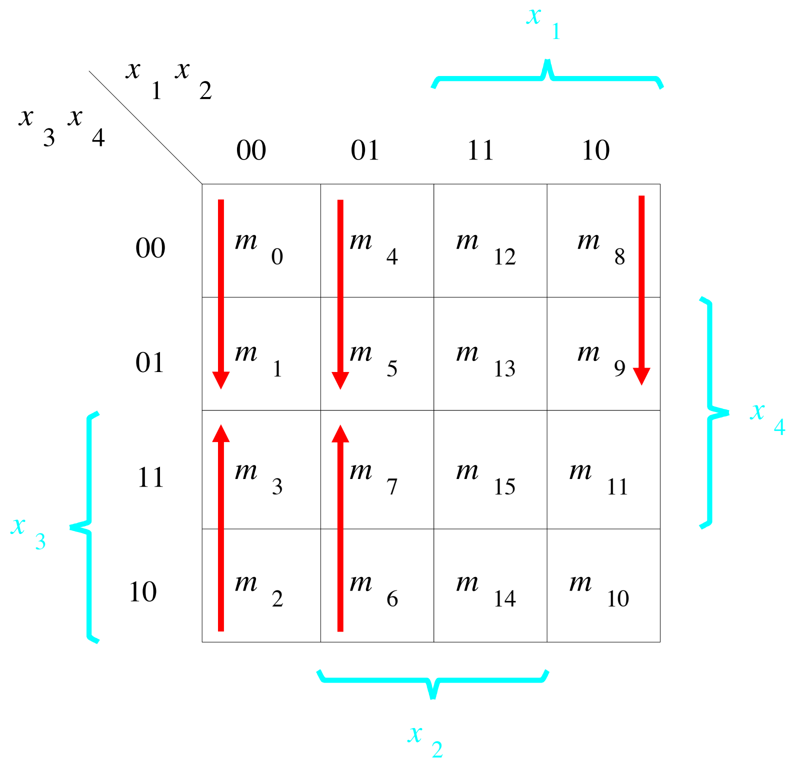
A four-variable Karnaugh map

x1	x2	x3	x4	
0	0	0	0	m0
0	0	0	1	m1
0	0	1	0	m2
0	0	1	1	m3
0	1	0	0	m4
0	1	0	1	m5
0	1	1	0	m6
0	1	1	1	m7
1	0	0	0	m8
1	0	0	1	m9
1	0	1	0	m10
1	0	1	1	m11
1	1	0	0	m12
1	1	0	1	m13
1	1	1	0	m14
1	1	1	1	m15



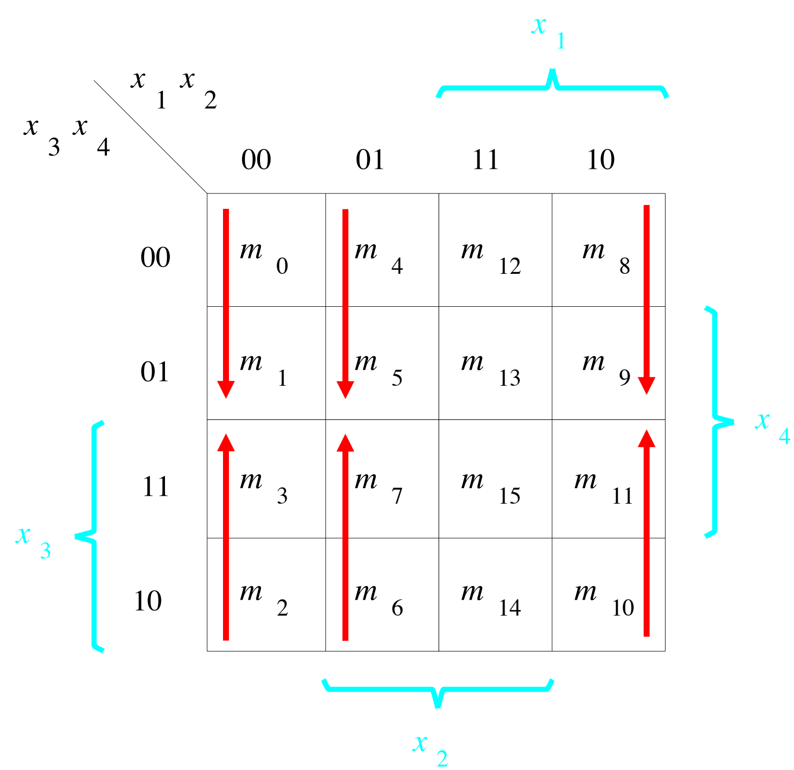
A four-variable Karnaugh map

x1	x2	x3	x4	
0	0	0	0	m0
0	0	0	1	m1
0	0	1	0	m2
0	0	1	1	m3
0	1	0	0	m4
0	1	0	1	m5
0	1	1	0	m6
0	1	1	1	m7
1	0	0	0	m8
1	0	0	1	m9
1	0	1	0	m10
1	0	1	1	m11
1	1	0	0	m12
1	1	0	1	m13
1	1	1	0	m14
1	1	1	1	m15



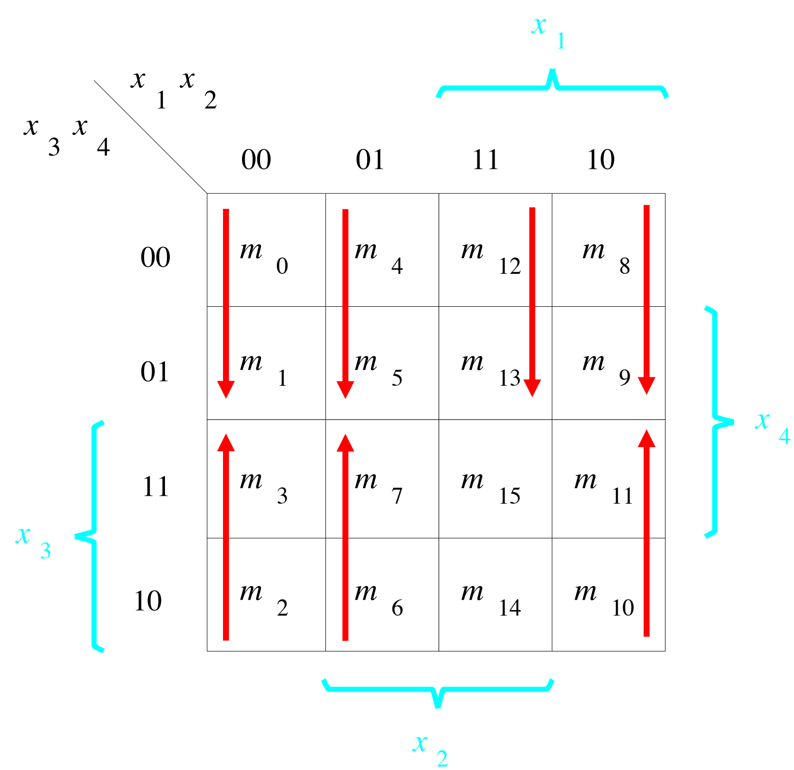
A four-variable Karnaugh map

x1	x2	x3	x4	
0	0	0	0	m0
0	0	0	1	m1
0	0	1	0	m2
0	0	1	1	m3
0	1	0	0	m4
0	1	0	1	m5
0	1	1	0	m6
0	1	1	1	m7
1	0	0	0	m8
1	0	0	1	m9
1	0	1	0	m10
1	0	1	1	m11
1	1	0	0	m12
1	1	0	1	m13
1	1	1	0	m14
1	1	1	1	m15



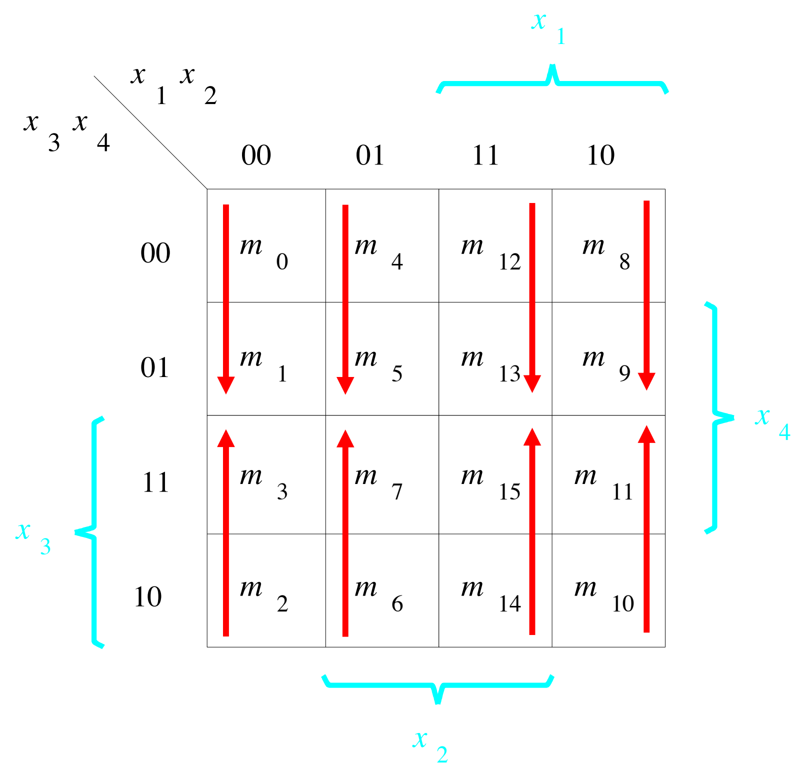
A four-variable Karnaugh map

x1	x2	x3	x4	
0	0	0	0	m0
0	0	0	1	m1
0	0	1	0	m2
0	0	1	1	m3
0	1	0	0	m4
0	1	0	1	m5
0	1	1	0	m6
0	1	1	1	m7
1	0	0	0	m8
1	0	0	1	m9
1	0	1	0	m10
1	0	1	1	m11
1	1	0	0	m12
1	1	0	1	m13
1	1	1	0	m14
1	1	1	1	m15

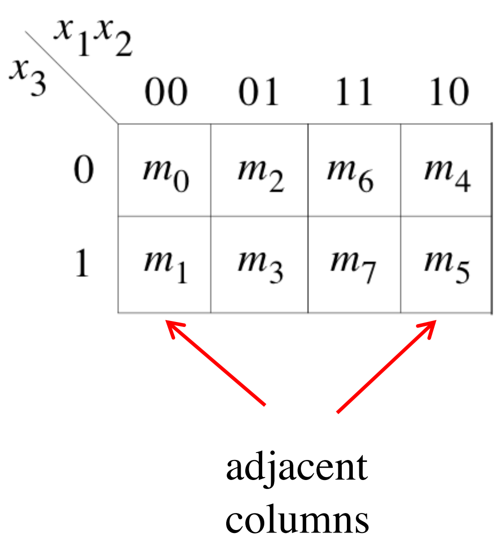


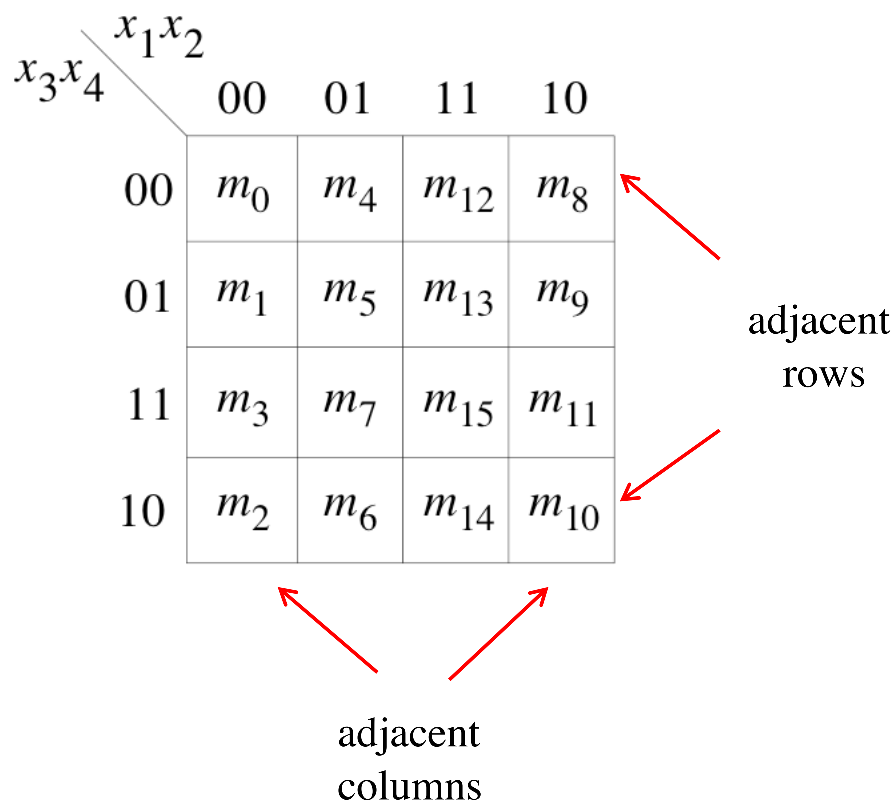
A four-variable Karnaugh map

x1	x2	x3	x4	
0	0	0	0	m0
0	0	0	1	m1
0	0	1	0	m2
0	0	1	1	m3
0	1	0	0	m4
0	1	0	1	m5
0	1	1	0	m6
0	1	1	1	m7
1	0	0	0	m8
1	0	0	1	m9
1	0	1	0	m10
1	0	1	1	m11
1	1	0	0	m12
1	1	0	1	m13
1	1	1	0	m14
1	1	1	1	m15

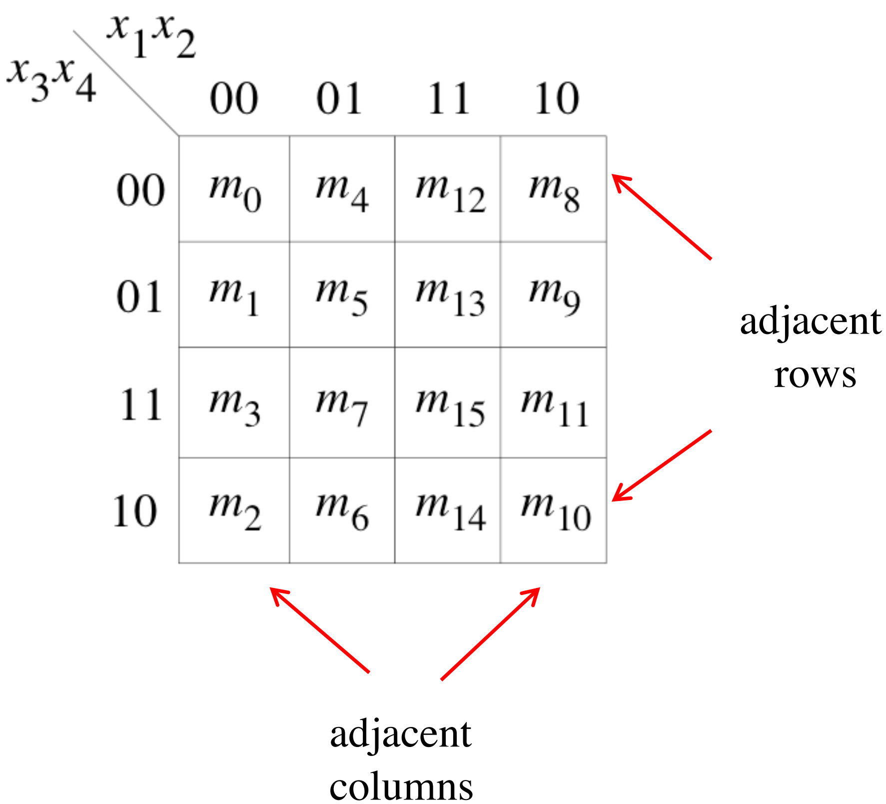


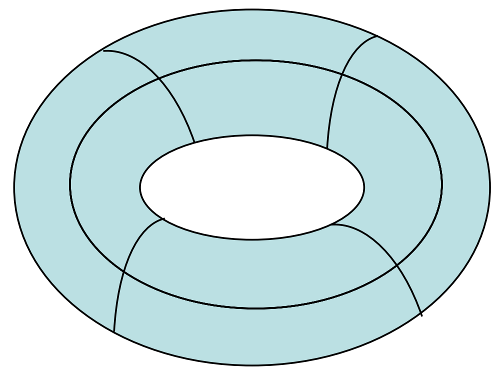
Adjacency Rules





Adjacency Rules





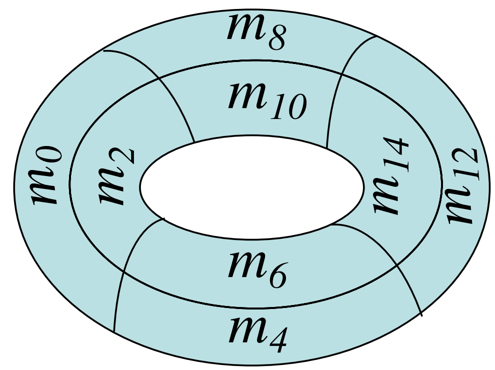
As if the K-map were drawn on a torus

Adjacency Rules

x_3x_4 \ x_1x_2	00	01	11	10
00	m_0	m_4	m_{12}	m_8
01	m_1	m_5	m_{13}	m_9
11	m_3	m_7	m_{15}	m_{11}
10	m_2	m_6	m_{14}	m_{10}

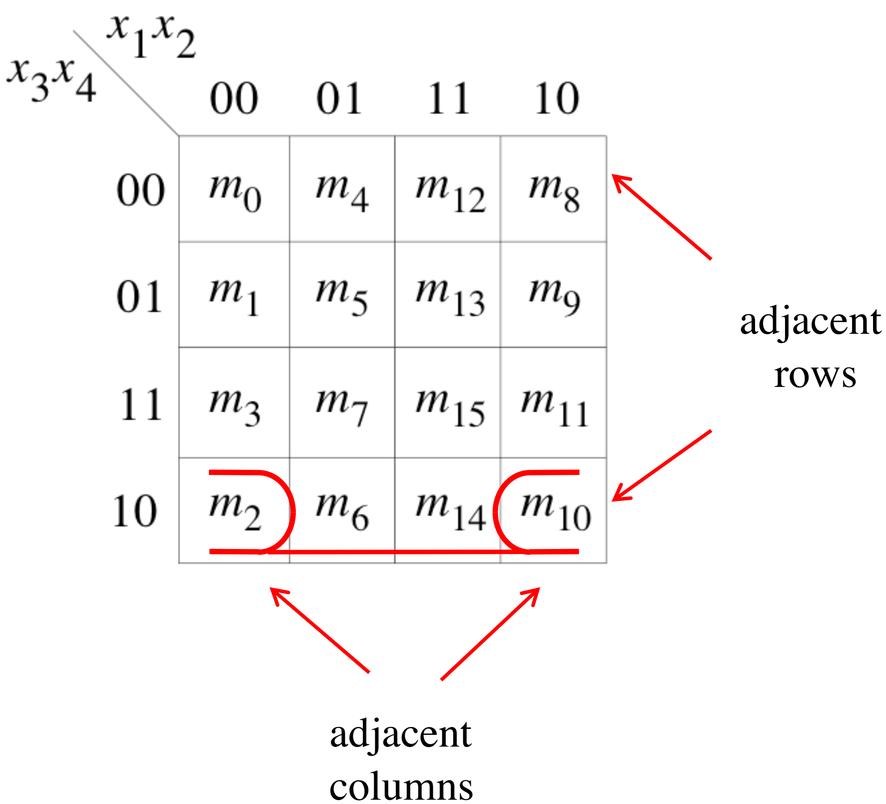
adjacent rows

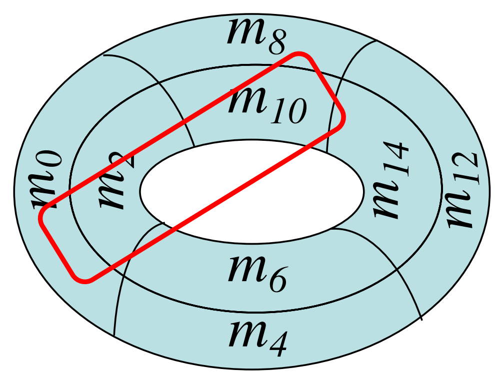
adjacent columns



As if the K-map were drawn on a torus

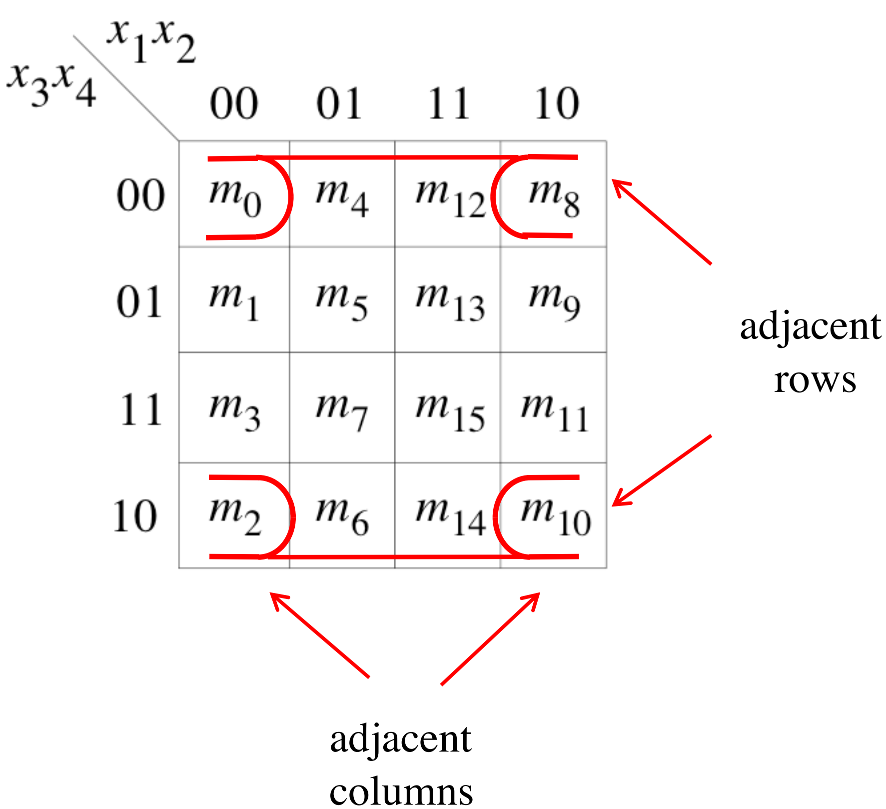
Adjacency Rules

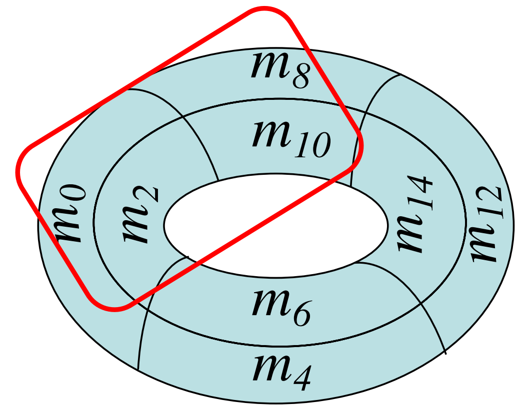




As if the K-map were drawn on a torus

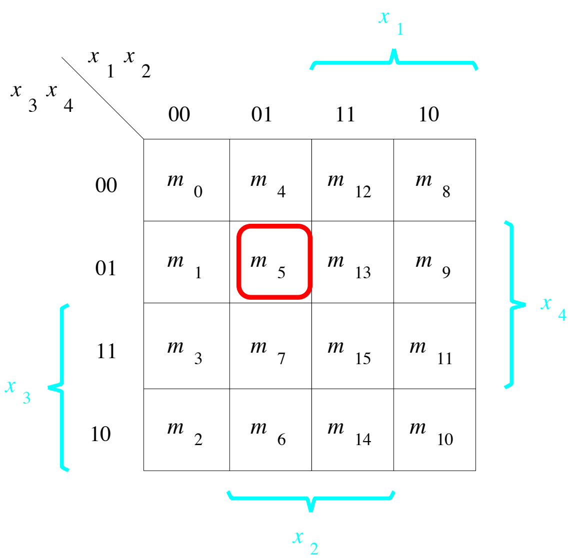
Adjacency Rules

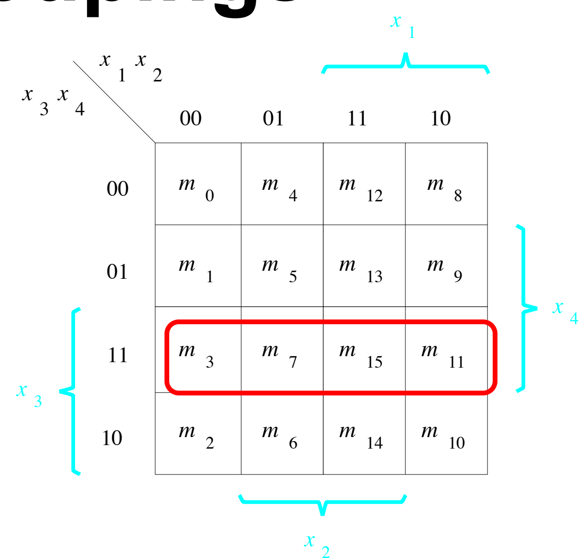


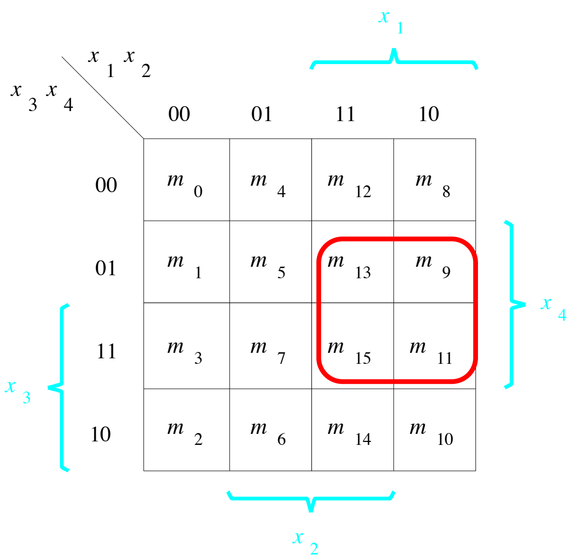


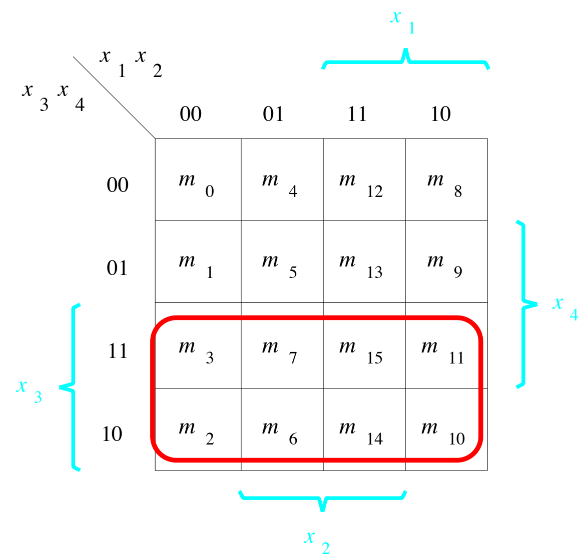
As if the K-map were drawn on a torus

Some Valid Groupings

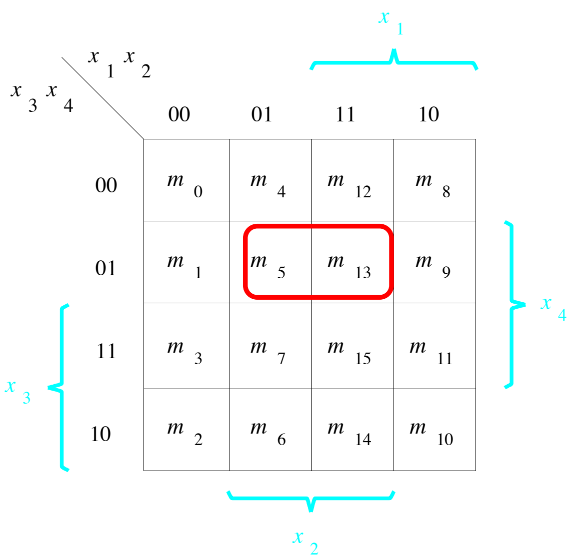


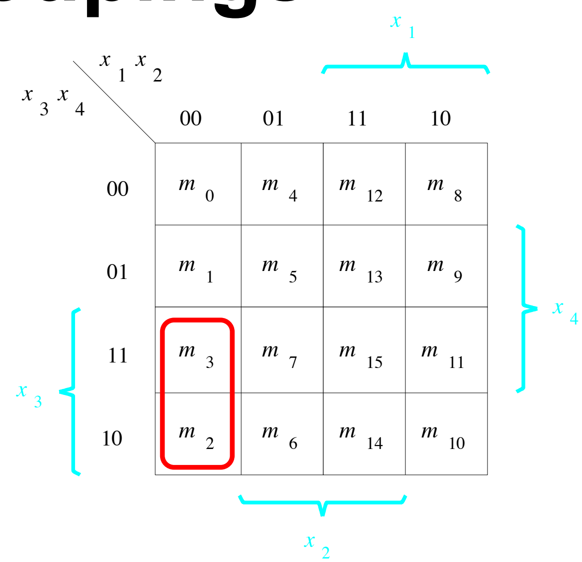


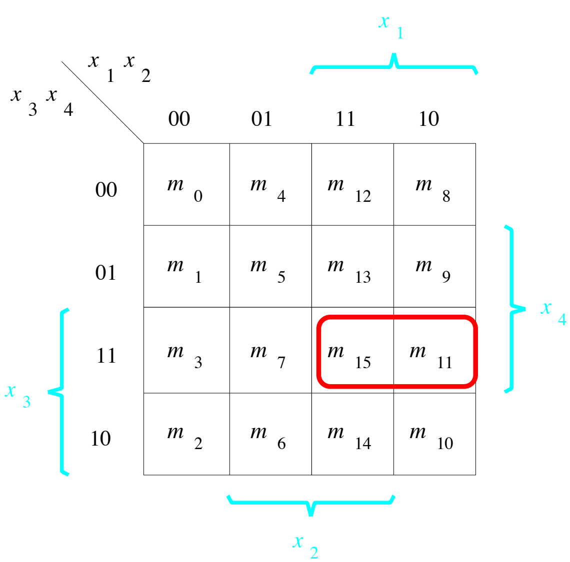


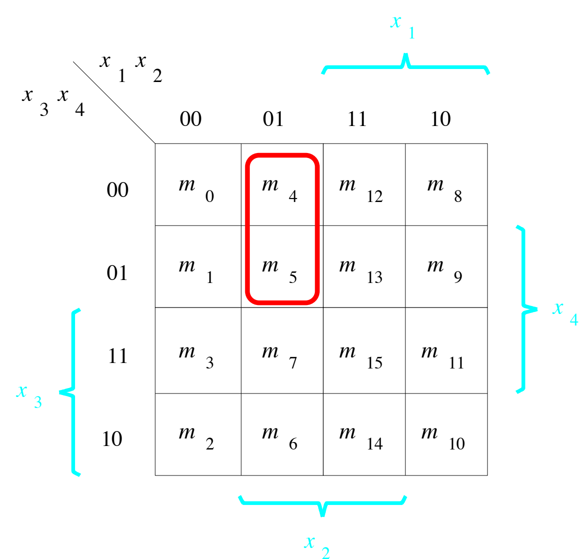


Some Valid Groupings

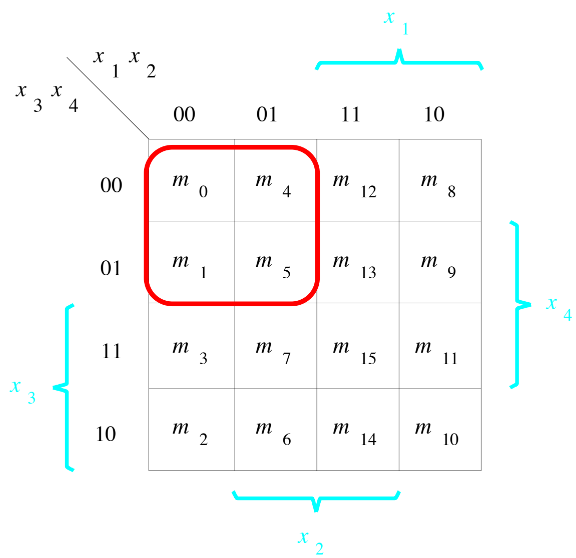


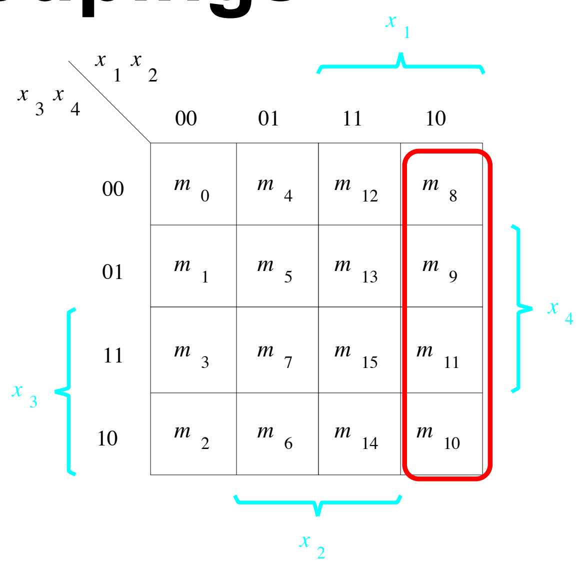


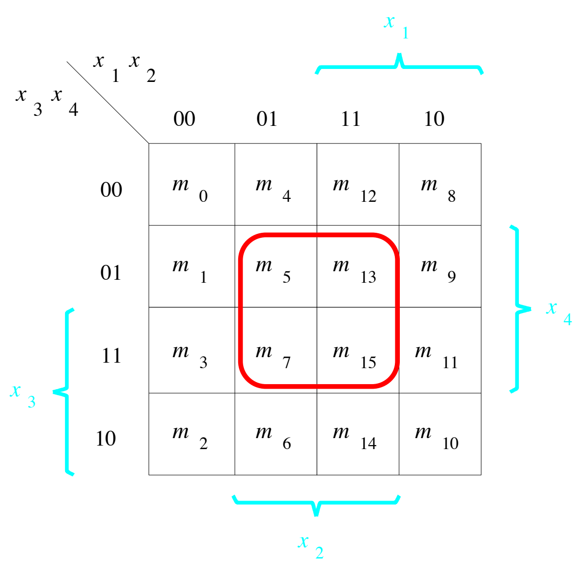


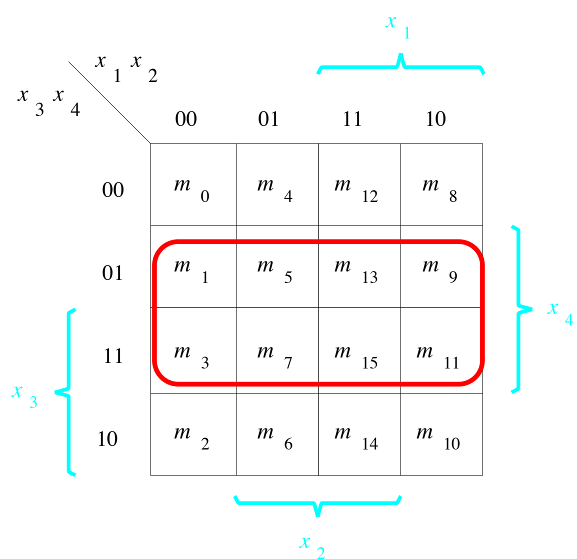


Some Valid Groupings

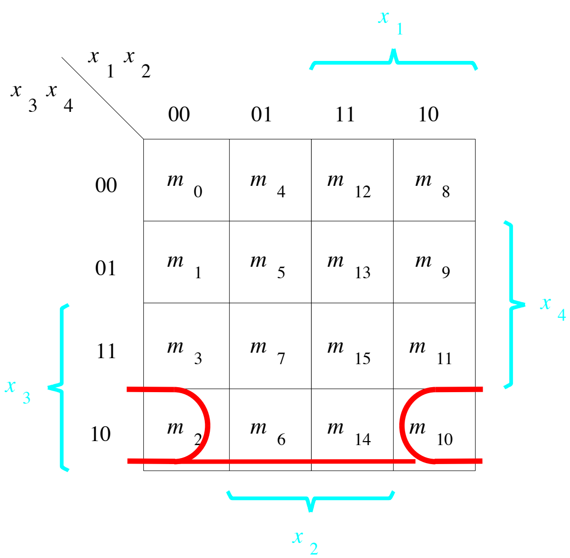


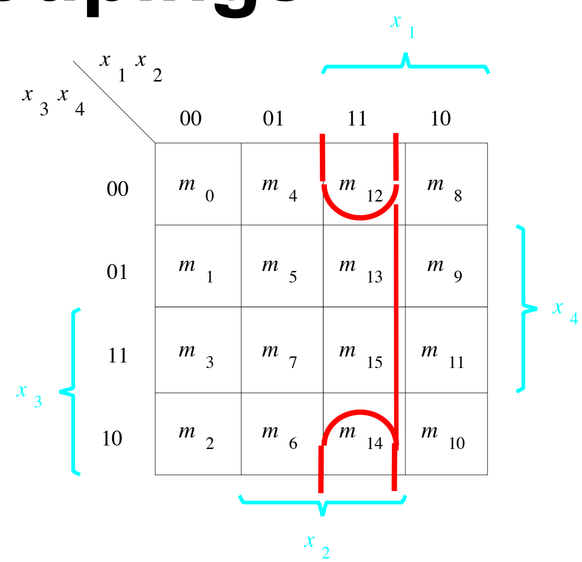


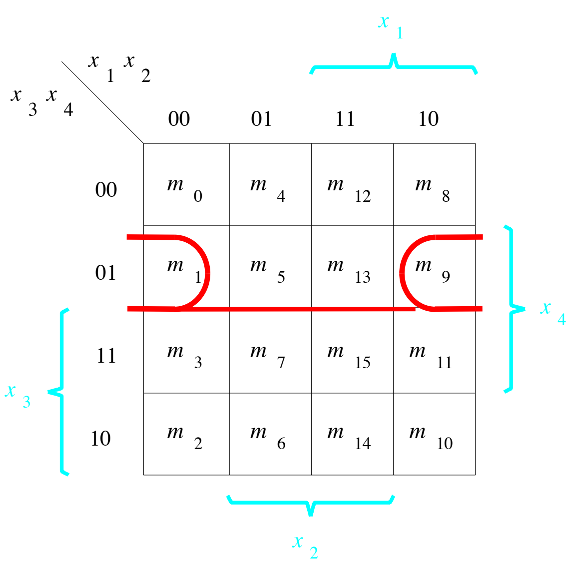


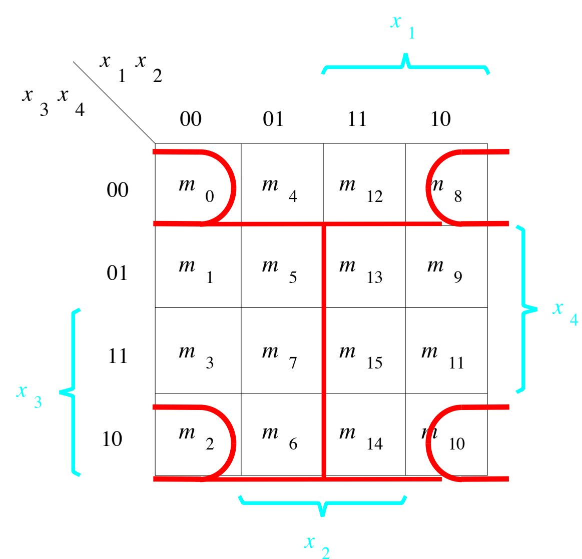


Some Valid Groupings

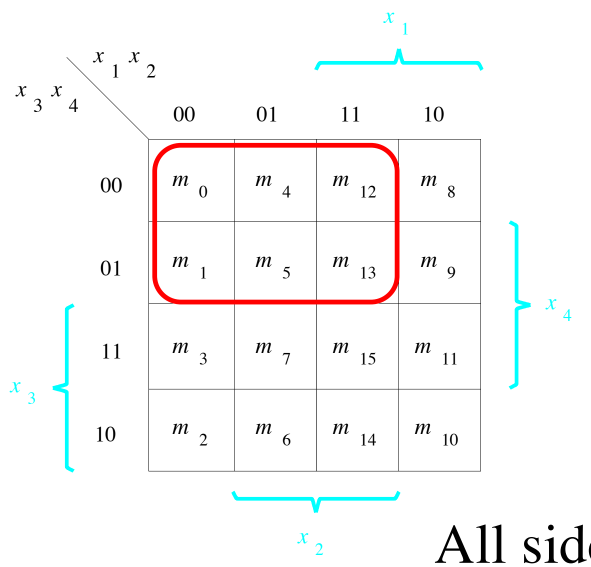


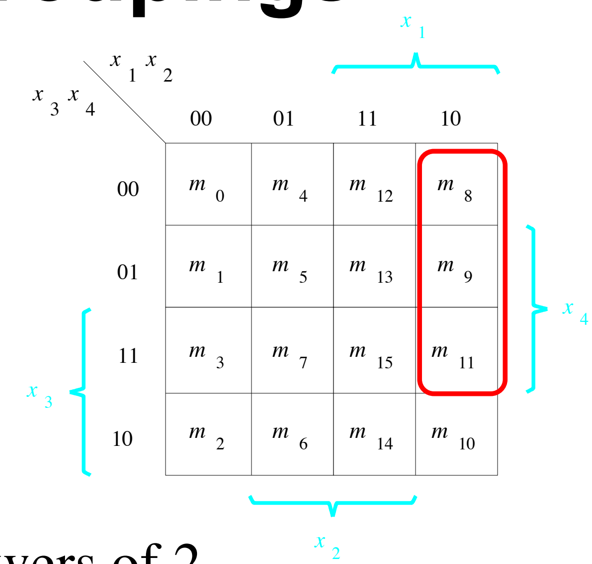




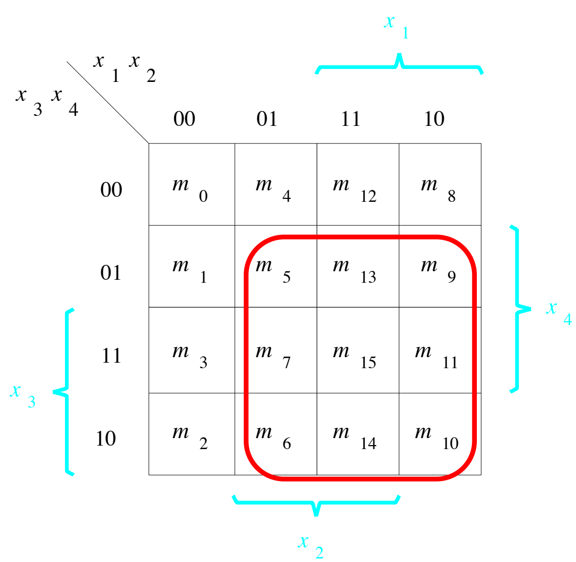


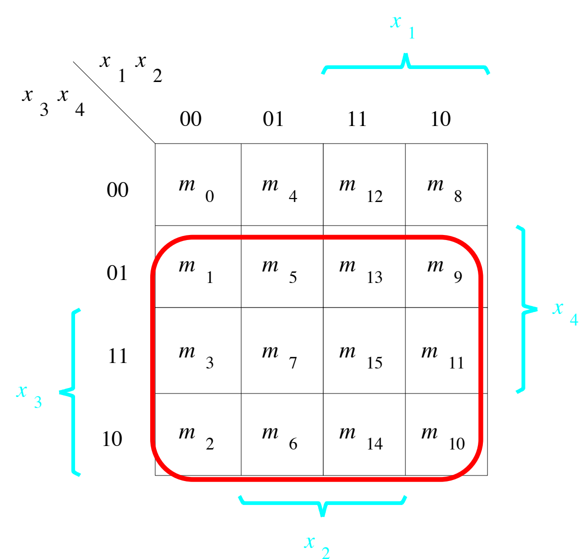
Some Invalid Groupings



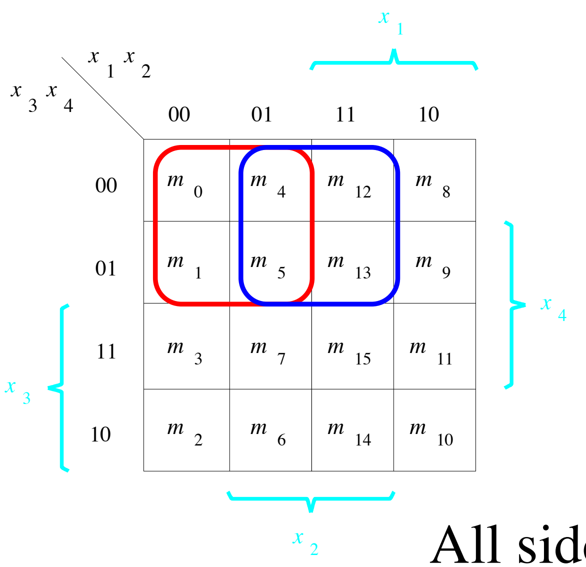


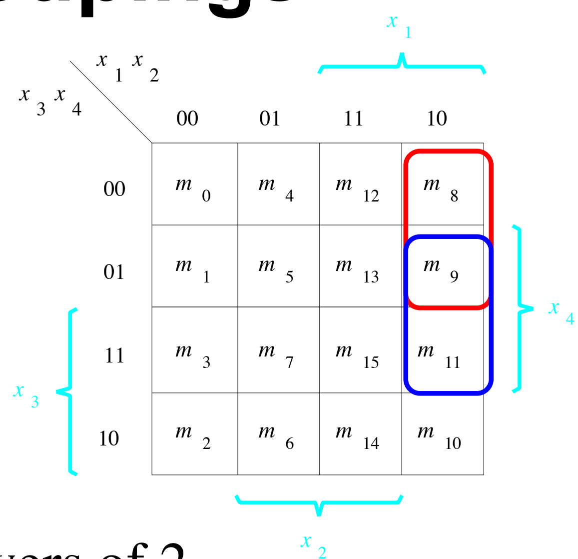
All sides must be powers of 2.



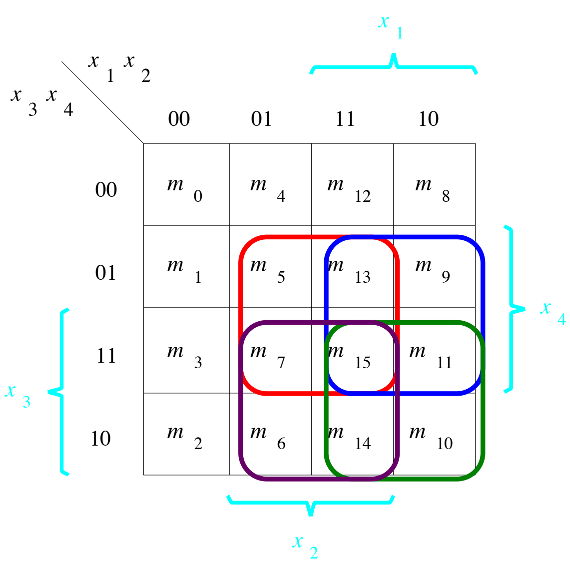


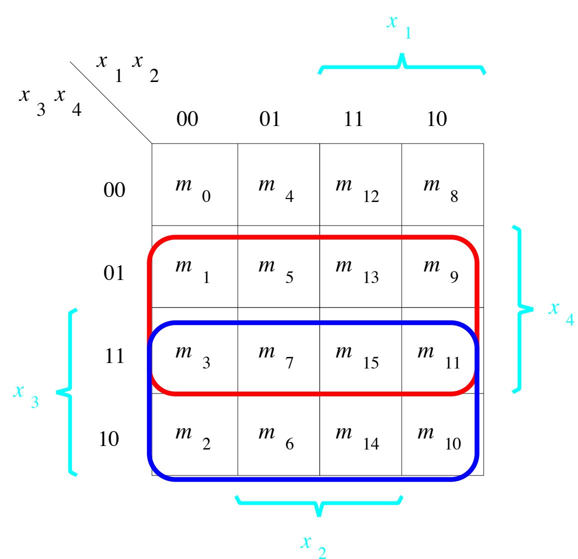
Some valid Groupings





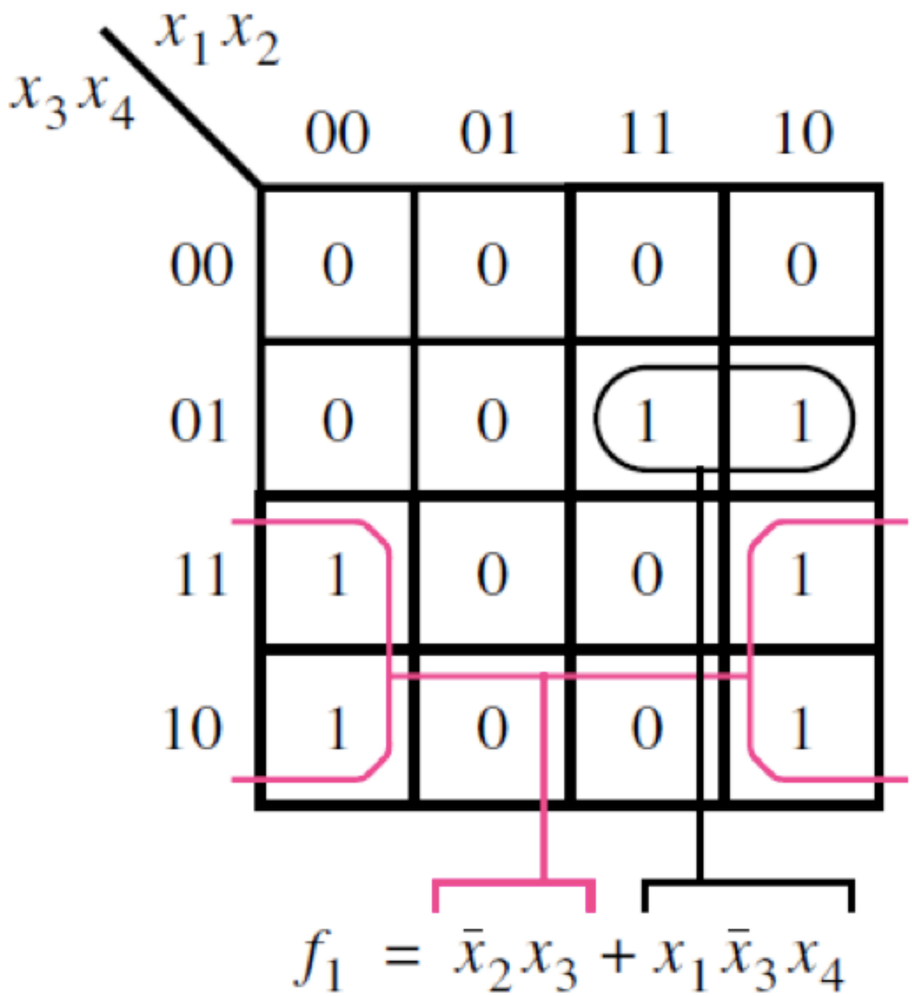
All sides must be powers of 2.





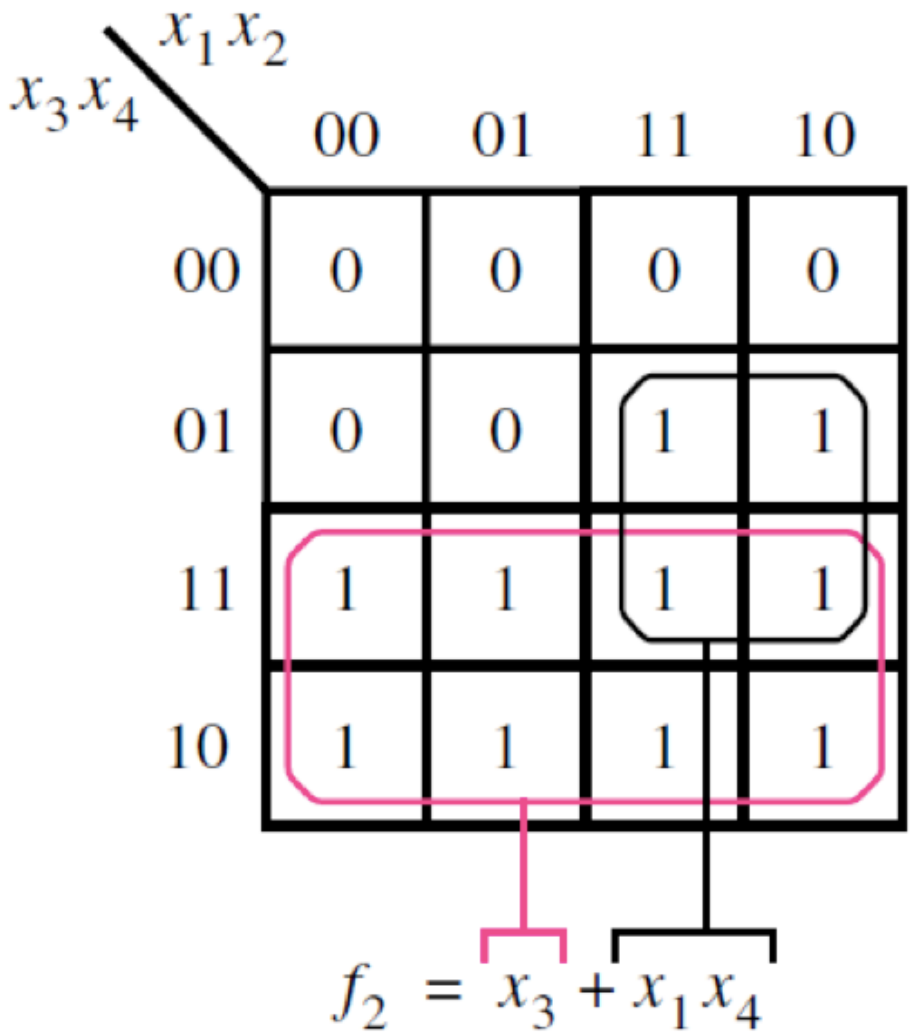
Minimization Examples with 4-variable K-Maps

Example of a four-variable Karnaugh map



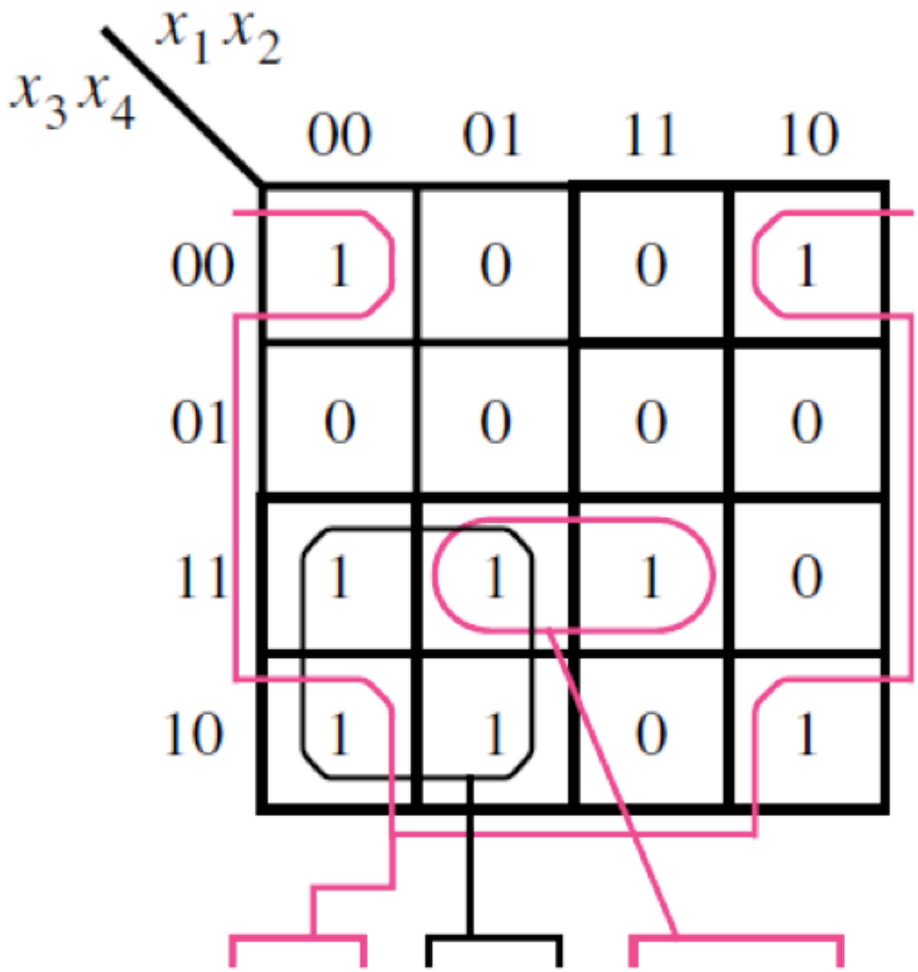
[Figure 2.54 from the textbook]

Example of a four-variable Karnaugh map



[Figure 2.54 from the textbook]

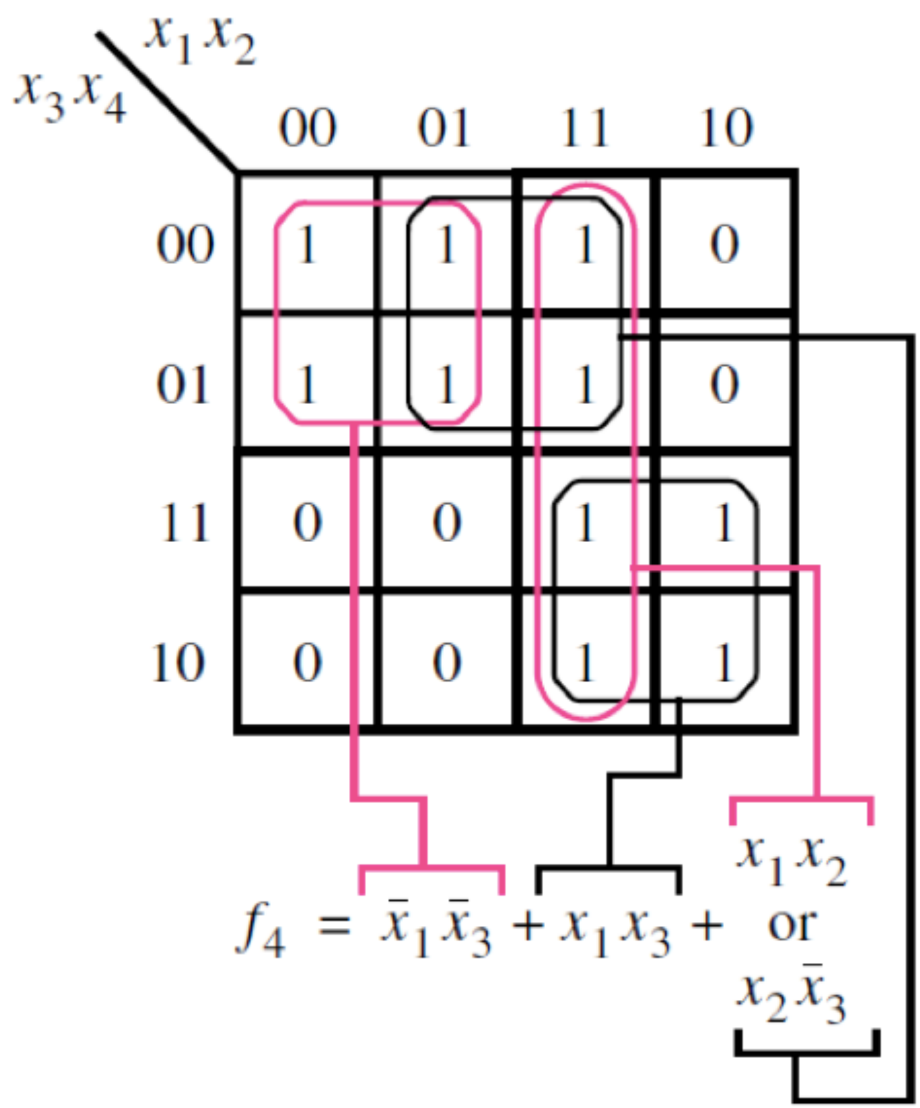
Example of a four-variable Karnaugh map



$$f_3 = \bar{x}_2 \bar{x}_4 + \bar{x}_1 x_3 + x_2 x_3 x_4$$

[Figure 2.54 from the textbook]

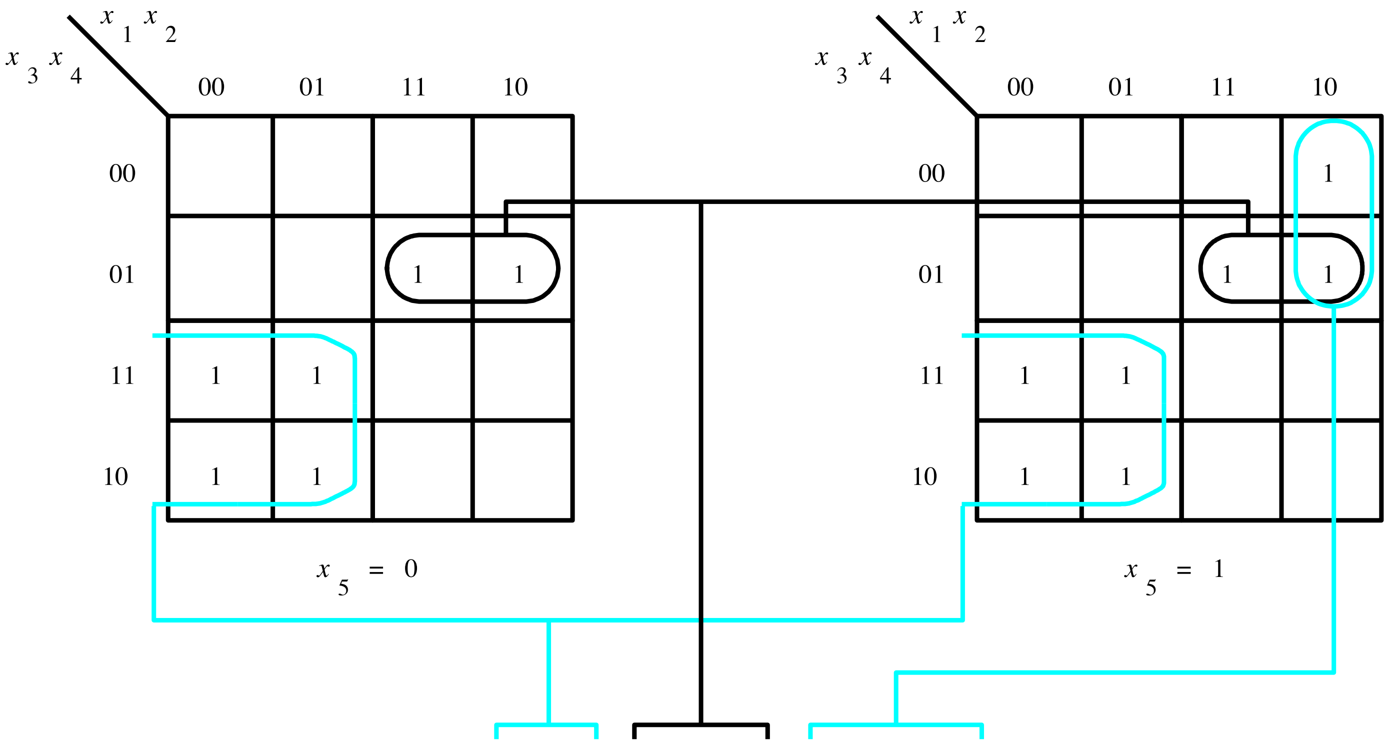
Example of a four-variable Karnaugh map



[Figure 2.54 from the textbook]

Five-Variable K-Map

A five-variable Karnaugh map

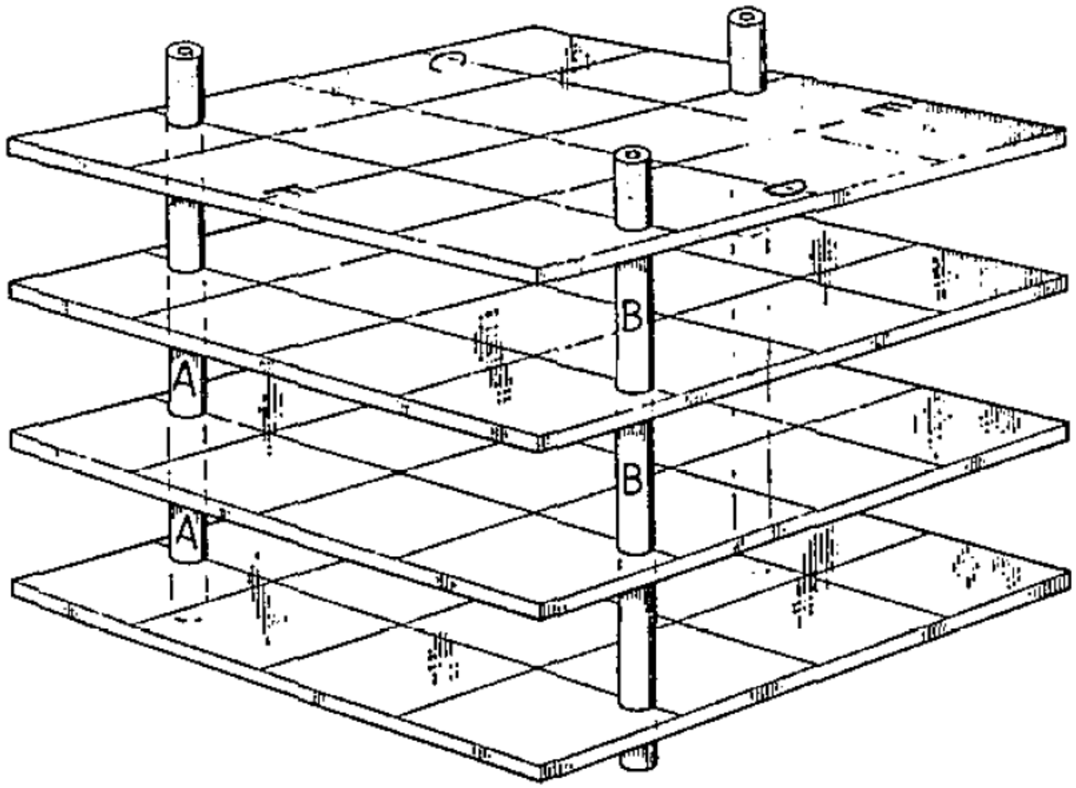


$$f_1 = \bar{x}_1 x_3 + x_1 \bar{x}_3 x_4 + x_1 \bar{x}_2 \bar{x}_3 x_5$$

[Figure 2.55 from the textbook]

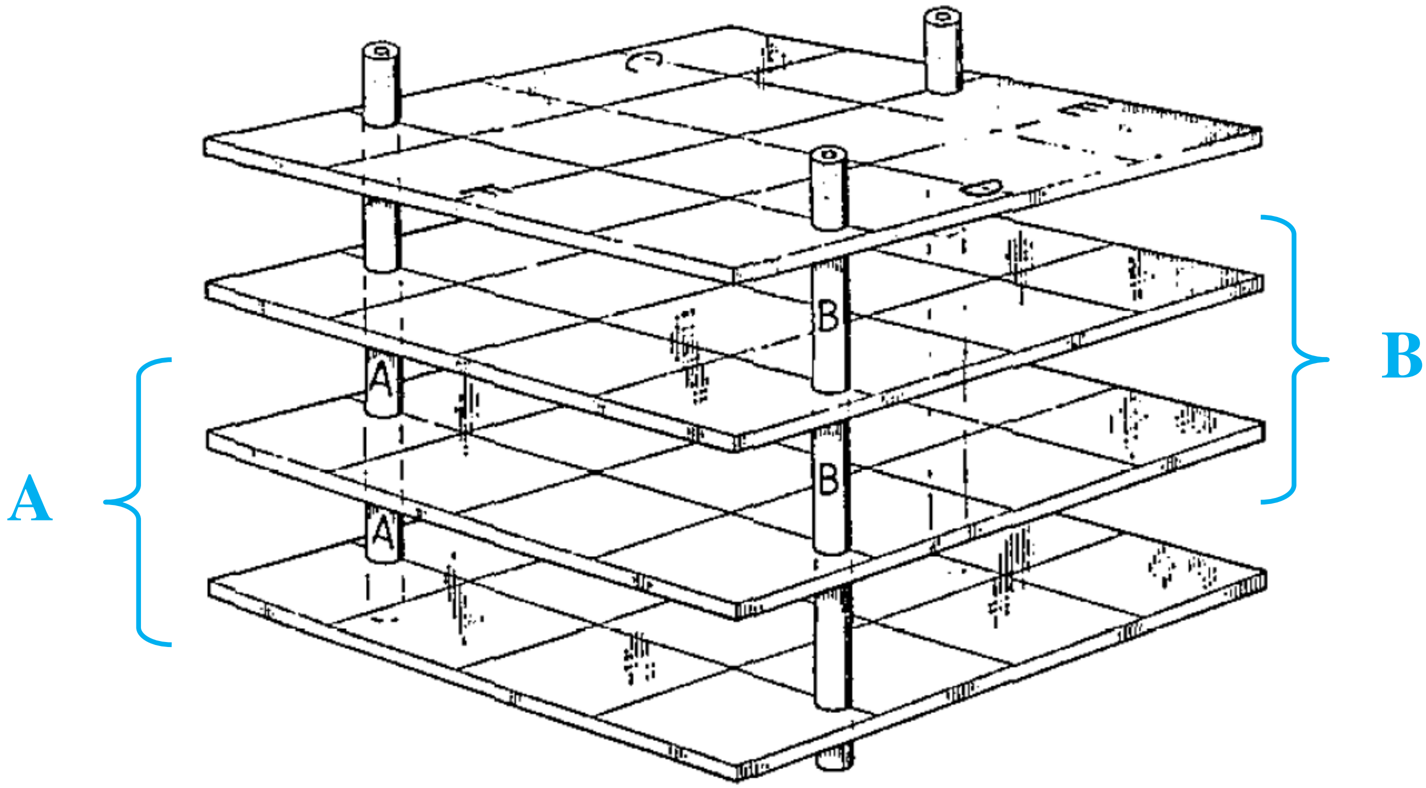
Six-Variable K-Map

A six-variable Karnaugh map



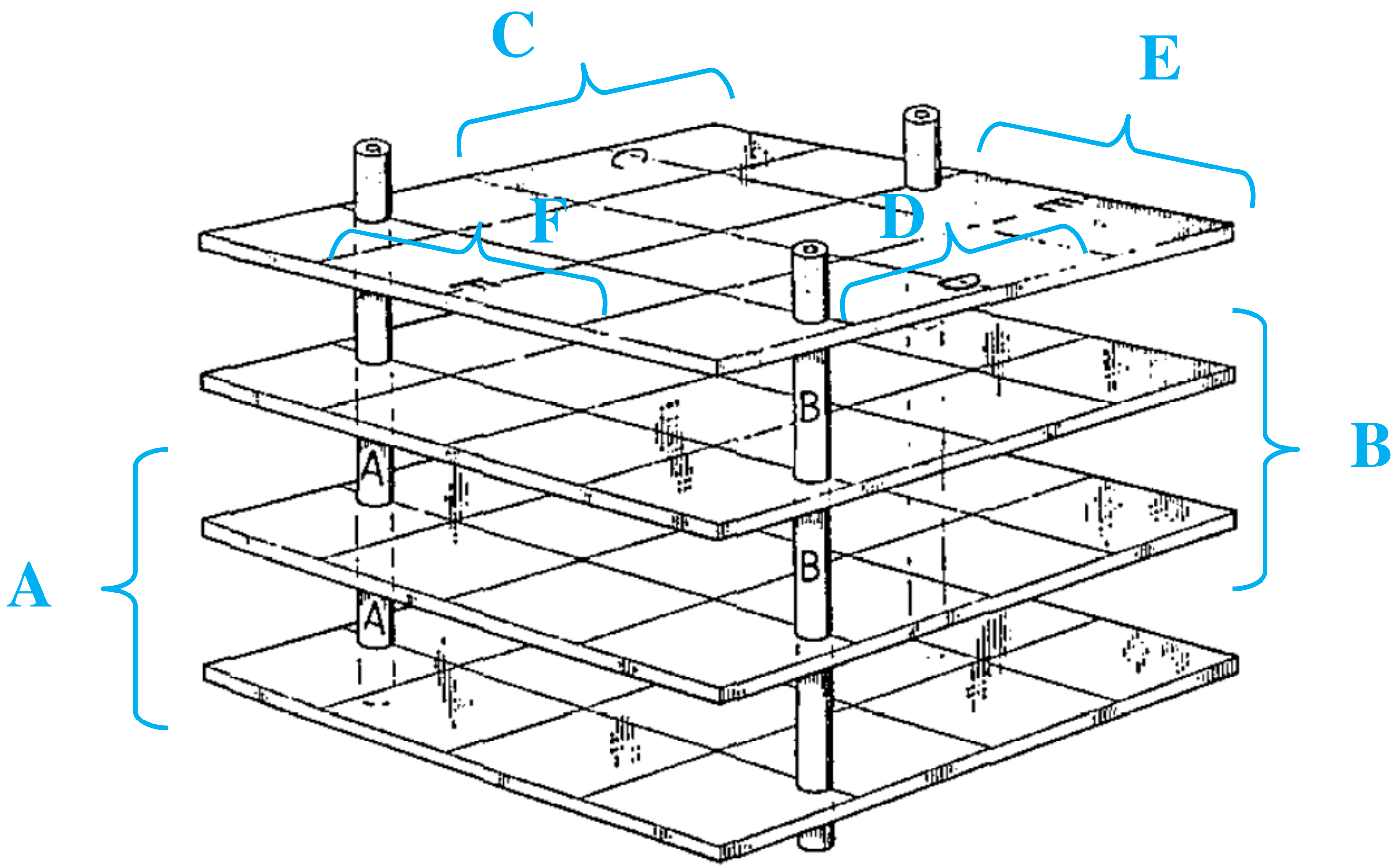
[Figure 16, in Karnaugh 1953]

A six-variable Karnaugh map



[Figure 16, in Karnaugh 1953]

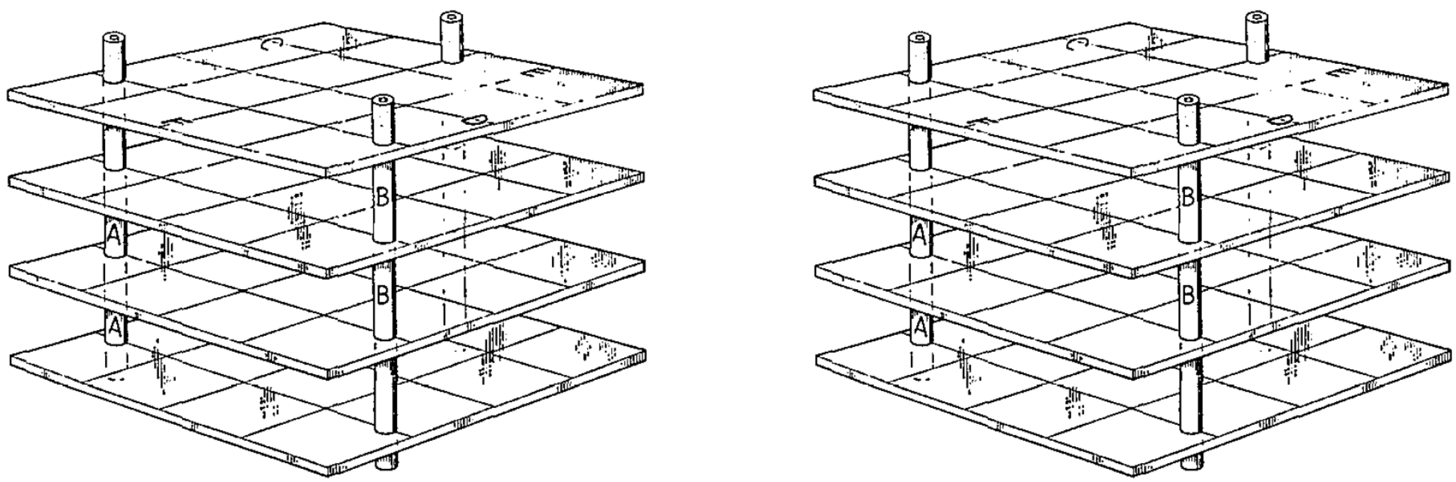
A six-variable Karnaugh map



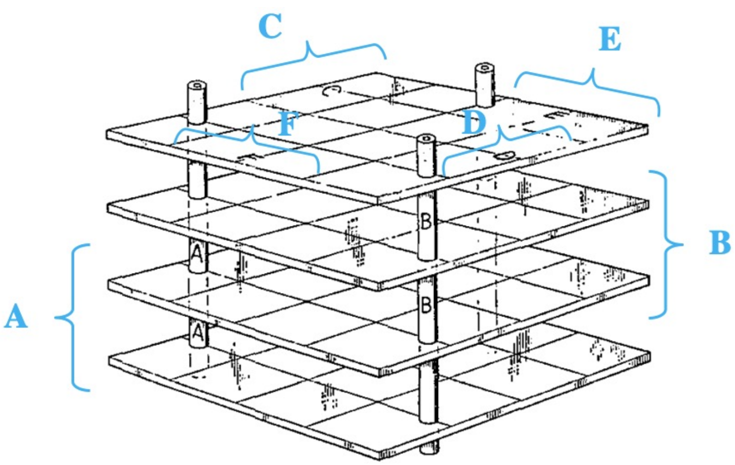
[Figure 16, in Karnaugh 1953]

Seven-Variable K-Map

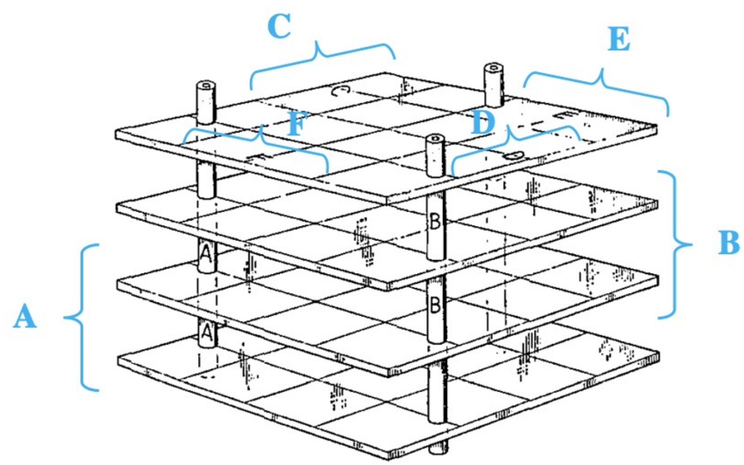
A seven-variable Karnaugh map



A seven-variable Karnaugh map



G = 0



G = 1

[Suggested in Karnaugh 1953]

Questions?

THE END