

ECE212/212s Lecture Notes

Lecture: Chapter 2

Dr. Clay Gloster, Jr.

Additional Logic Gates

NAND, NOR, XOR, XNOR.

$$\text{NAND} = \text{not AND} = \overline{(X \cdot Y)}$$

X	Y	$X \cdot Y$	$\overline{(X \cdot Y)}$
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

$$\text{NOR} = \text{not OR} = \overline{(X + Y)}$$

X	Y	$X + Y$	$\overline{(X + Y)}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

$$\text{XOR} = X \oplus Y = \overline{X}Y + X\overline{Y}$$

$$\text{XNOR} = \text{NOT XOR} = \overline{X \oplus Y} = \overline{\overline{X}Y + X\overline{Y}}$$

X	Y	$X \oplus Y$	$\overline{X \oplus Y}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	0	1

Axioms of Boolean Algebra

1. The set B contains at least two elements a,b and $a \neq b$
2. Closure: for every element a,b $\in$ B,

1. $a+b \in B$

2. $a \cdot b \in B$

3. Commutative laws: For every a,b $\in$ B

1. $a+b=b+a$

2. $a \cdot b = b \cdot a$

4. Associative laws: For every $a, b, c \in B$

1. $(a+b)+c = a+(b+c) = a+b+c$

2. $(a \cdot b) \cdot c = a \cdot (b \cdot c) = a \cdot b \cdot c$

5. Identities

1. $a + 0 = a$

2. $a \cdot 1 = a$

6. Distributive laws: For every $a, b, c \in B$

1. $a+(b \cdot c) = (a+b) \cdot (a+c)$

2. $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$

7. Complement

1. $a + \bar{a} = 1$

2. $a \cdot \bar{a} = 0$

Laws and Theorems of Boolean Algebra

Idempotent theorem (Theorem 3) page 51

3. $X+X = X$

- 3D. $X \cdot X = X$

Duality Every Boolean function has a dual. It is derived by replacing all "AND" operations with "OR" and all "OR" operations with "AND".

Involution Theorem

4. $(X')' = X$

Simplification Theorems

9. $X \cdot Y + X \cdot \bar{Y} = X$

$$\begin{aligned} X \cdot Y + X \cdot \bar{Y} &= X \cdot (Y + \bar{Y}) \text{ Distributive} \\ &= X \cdot (1) \text{ Complementarity} \\ &= X \text{ Identities} \end{aligned}$$

9D. $(X + Y) \cdot (X + \bar{Y}) = X$

10. $X + (X \cdot Y) = X$

$$\begin{aligned} X + (X \cdot Y) &= X \cdot (1 + Y) \text{ Distributive} \\ &= X \cdot (1) \text{ Complementarity} \\ &= X \text{ Identities} \end{aligned}$$

10D. $X \cdot (X + Y) = X$

11. $(X + \bar{Y}) \cdot Y = X \cdot Y$

$$\begin{aligned} (X + \bar{Y}) \cdot Y &= (X \cdot Y) + (\bar{Y} \cdot Y) \text{ Distributive} \\ &= (X \cdot Y) + (0) \text{ Complementarity} \\ &= (X \cdot Y) \text{ Identities} \end{aligned}$$

11D. $(X \cdot \bar{Y}) + Y = X + Y$

DeMorgan's Law

$$\overline{X + Y} = \bar{X} \cdot \bar{Y}$$

$$\overline{X \cdot Y} = \bar{X} + \bar{Y}$$

Proving DeMorgan's Law using Truth Tables

X	Y	$\bar{X}$	$\bar{Y}$	$\bar{X} \cdot \bar{Y}$	$X + Y$	$\overline{X + Y}$
0	0	1	1	1	0	1
0	1	1	0	0	1	0
1	0	0	1	0	1	0
1	1	0	0	0	1	0

Note that the columns $\overline{X + Y}$ and $\bar{X} \cdot \bar{Y}$ are equal.

X	Y	$\bar{X}$	$\bar{Y}$	$X \cdot Y$	$\overline{X \cdot Y}$	$\bar{X} + \bar{Y}$
0	0	1	1	0	1	1
0	1	1	0	0	1	1
1	0	0	1	0	1	1
1	1	0	0	1	0	0

Note that the last two columns are equal.

Exercise $Z = A \overline{B} \overline{C} + A \overline{B} C + \overline{A} B C$

What is $\overline{Z}$?

$$\overline{Z} = \overline{A \overline{B} \overline{C} + A \overline{B} C + \overline{A} B C}$$

Let

$$x_1 = A \overline{B} \overline{C}$$

$$x_2 = A \overline{B} C$$

$$x_3 = \overline{A} B C$$

$$Z = x_1 + x_2 + x_3$$

$$\overline{Z} = \overline{(x_1 + x_2 + x_3)}$$

$$\overline{Z} = \overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3}$$

$$\overline{x_1} = \overline{A} + B + C$$

$$\overline{x_2} = \overline{A} + B + \overline{C}$$

$$\overline{x_3} = A + \overline{B} + \overline{C}$$

$$\overline{Z} = (\overline{A} + B + C) \cdot (\overline{A} + B + \overline{C}) \cdot (A + \overline{B} + \overline{C})$$

Logic Minimization

- Boolean Simplification
- Karnaugh Maps
- Quine McCluskey
- Espresso (CAD Tool)

Logic Minimization Using Boolean Simplification

Example Full Adder

A	B	Cin	Cout
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

$$Cout = \overline{A} B Cin + A \overline{B} Cin + A B \overline{Cin} + A B Cin$$

$$Cout = f(A, B, Cin) = m_3 + m_5 + m_6 + m_7 = \sum m(3, 5, 6, 7) \text{ (SOP)}$$

$$\begin{aligned}
Cout &= \overline{A}BCin + A\overline{B}Cin + AB\overline{C}in + ABCin \\
Cout &= \overline{A}BCin + A\overline{B}Cin + AB\overline{C}in + ABCin + ABCin \quad \text{Idempotent} \\
Cout &= \overline{A}BCin + A\overline{B}Cin + AB\overline{C}in + ABCin + ABCin + ABCin \quad \text{Idempotent} \\
Cout &= \overline{A}BCin + ABCin + A\overline{B}Cin + ABCin + AB\overline{C}in + ABCin \quad \text{Commutative} \\
Cout &= (\overline{A} + A)BCin + (\overline{B} + B)ACin + (\overline{C}in + Cin)AB \quad \text{Factoring} \\
Cout &= BCin + ACin + AB \quad \text{Complementarity}
\end{aligned}$$

Another Example

$$\begin{aligned}
f(X, Y) &= (X + Y)(X + \overline{Y}) \\
&= (X \cdot X) + (X\overline{Y}) + (YX) + (Y\overline{Y}) \quad \text{Distributive} \\
&= X + (X\overline{Y}) + (XY) \quad \text{Idempotent} \\
&= X(1 + \overline{Y}) + (XY) \quad \text{Idempotent} \\
&= X + (XY) \quad \text{Idempotent} \\
&= X(1 + Y) \quad \text{Operations with 1 or 0} \\
&= X \quad \text{Operations with 1 or 0}
\end{aligned}$$

Two Level Canonical Forms

- Sum of Products
- Product of Sums

A	B	C	f	$\overline{f}$	m	M
0	0	0	0	1	m_0	M_0
0	0	1	0	1	m_1	M_1
0	1	0	0	1	m_2	M_2
0	1	1	1	0	m_3	M_3
1	0	0	1	0	m_4	M_4
1	0	1	1	0	m_5	M_5
1	1	0	1	0	m_6	M_6
1	1	1	1	0	m_7	M_7

Sum of Products

If f is a function of A, B, and C, then

$$\begin{aligned}
m_0 &= \overline{A} \overline{B} \overline{C} \\
m_1 &= \overline{A} \overline{B} C
\end{aligned}$$

$$\begin{aligned}
m_2 &= \overline{A} B \overline{C} \\
m_3 &= \overline{A} B C \\
m_4 &= A \overline{B} \overline{C} \\
m_5 &= A \overline{B} C \\
m_6 &= A B \overline{C} \\
m_7 &= A B C
\end{aligned}$$

$$\begin{aligned}
f(A, B, C) &= \overline{A} B C + A \overline{B} \overline{C} + A \overline{B} C + A B \overline{C} + A B C \\
f(A, B, C) &= m_3 + m_4 + m_5 + m_6 + m_7 = \sum m(3, 4, 5, 6, 7) \\
\overline{f(A, B, C)} &= \overline{A} \overline{B} \overline{C} + \overline{A} \overline{B} C + \overline{A} B \overline{C} \\
\overline{f(A, B, C)} &= m_0 + m_1 + m_2 = \sum m(0, 1, 2)
\end{aligned}$$

Product of Sums

If f is a function of A,B, and C then

$$\begin{aligned}
M_0 &= A + B + C \\
M_1 &= A + B + \overline{C} \\
M_2 &= A + \overline{B} + C \\
M_3 &= A + \overline{B} + \overline{C} \\
M_4 &= \overline{A} + B + C \\
M_5 &= \overline{A} + B + \overline{C} \\
M_6 &= \overline{A} + \overline{B} + C \\
M_7 &= \overline{A} + \overline{B} + \overline{C}
\end{aligned}$$

$$\begin{aligned}
f(A, B, C) &= (A + B + C) \cdot (A + B + \overline{C}) \cdot (A + \overline{B} + C) \\
f(A, B, C) &= M_0 \cdot M_1 \cdot M_2 = \prod M(0, 1, 2) \\
\overline{f(A, B, C)} &= \prod M(3, 4, 5, 6, 7)
\end{aligned}$$

Incompletely Specified Function

BCD add one function

A	B	C	D	W	X	Y	Z
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	0	1	1	0	1	0	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	0	0	0	0
1	0	1	0	X	X	X	X
1	0	1	1	X	X	X	X
1	1	0	0	X	X	X	X
1	1	0	1	X	X	X	X
1	1	1	0	X	X	X	X
1	1	1	1	X	X	X	X

$$f_W(A, B, C, D) = \sum m(7, 8) + \sum d(10, 11, 12, 13, 14, 15)$$

$$f_X(A, B, C, D) = \prod M(0, 1, 2, 7, 8, 9) \prod D(10, 11, 12, 13, 14, 15)$$

Write the boolean equations for f_Y and f_Z in SOP and POS form respectively.

$$f_Y(A, B, C, D) = \sum m(1, 2, 5, 6) + \sum d(10, 11, 12, 13, 14, 15)$$

$$f_Z(A, B, C, D) = \prod M(1, 3, 5, 7, 9) \prod D(10, 11, 12, 13, 14, 15)$$

What is the on set for $f_Y(A, B, C, D)$? $\sum m(1, 2, 5, 6)$

What is the off set? $\sum M(1, 3, 5, 7, 9)$

What is the don't care set for $f_W(A, B, C, D)$? $\sum d(10, 11, 12, 13, 14, 15)$

Note Z:

$Z = \overline{D}$ after we replace the X's with 0's and 1's

Converting from minterm expansion to maxterm expansion

$$F(A, B, C) = \sum m(3, 4, 5, 6, 7)$$

$$F(A, B, C) = \prod M(0, 1, 2)$$

Converting from maxterm expansion to minterm expansion

$$F(A, B, C) = \prod M(0, 1, 2)$$

$$F(A, B, C) = \sum m(3, 4, 5, 6, 7)$$

Finding minterm expansion of complement of F given minterm expansion of f

$$F(A, B, C) = \sum m(3, 4, 5, 6, 7)$$

$$\overline{F(A, B, C)} = \sum m(0, 1, 2)$$

Finding maxterm expansion of complement of F given maxterm expansion of f

$$\overline{F(A, B, C)} = \prod M(0, 1, 2)$$

$$\overline{F(A, B, C)} = \prod M(3, 4, 5, 6, 7)$$

Deriving SOP and POS from any given expression

$$F(A, B, C) = A B + \overline{B} C + \overline{A} B \overline{C}$$

$$F(A, B, C) = A B (\overline{C} + C) + (\overline{A} + A) \overline{B} C + \overline{A} B \overline{C}$$

$$F(A, B, C) = A B \overline{C} + A B C + \overline{A} \overline{B} C + A \overline{B} C + \overline{A} B \overline{C}$$

$$F_d(A, B, C) = (A + B)(\overline{B} + C)(\overline{A} + B + \overline{C})$$

$$F_d(A, B, C) = (A + B + (\overline{C} C))((\overline{A} A) + \overline{B} + C)(\overline{A} + B + \overline{C})$$

$$F_d(A, B, C) = (A + B + \overline{C})(A + B + C)(\overline{A} + \overline{B} + C)(A + \overline{B} + C)(\overline{A} + B + \overline{C})$$

Karnaugh Maps Tools used for Logic Minimization

Implicant A single element of the on-set or any set of elements that can be grouped together in a K-MAP.

Prime Implicant Implicants with as few literals as possible (maximal group size)

Essential Prime Implicant A prime implicant that is the only prime implicant to cover a particular element of the on-set.

Algorithm for finding minimized expression using Kmaps

1. find all prime implicants (maximal groups)
2. find all essential prime implicants
3. find minimum set of non-essential implicants to cover the remaining elements of the on-set.

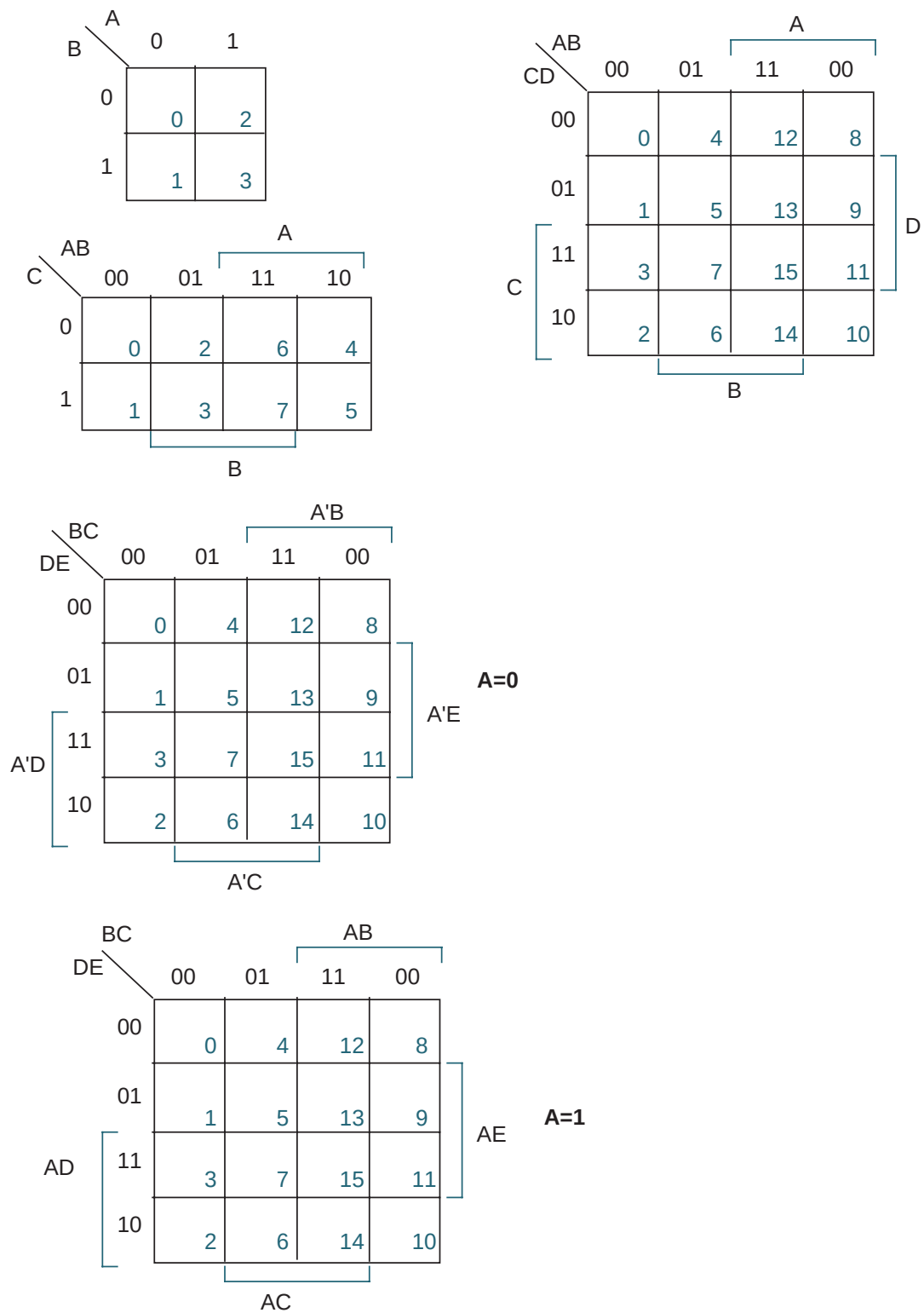
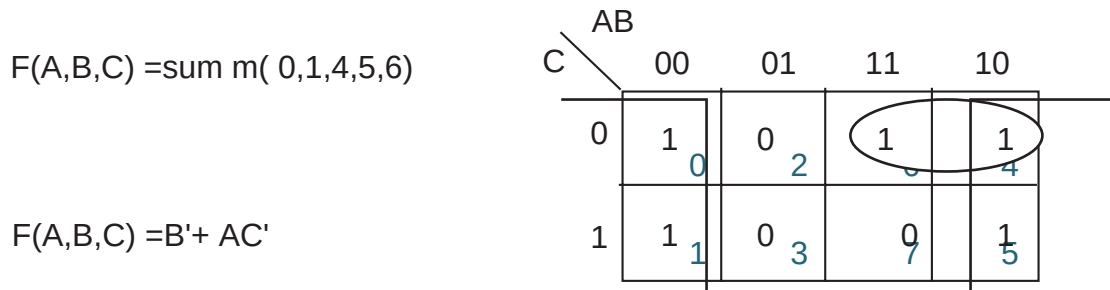
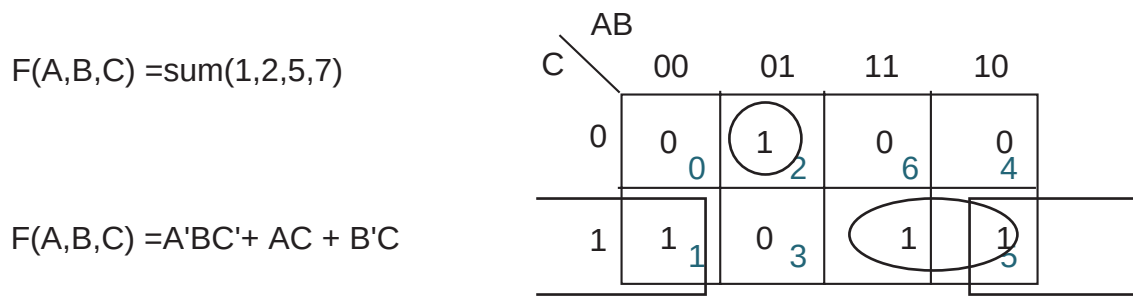


Figure 1: KMAPS for 2,3,4, and 5 variable functions

3 variable K-Maps



POS: Circle zeroes and use Demorgan's Law!!!!

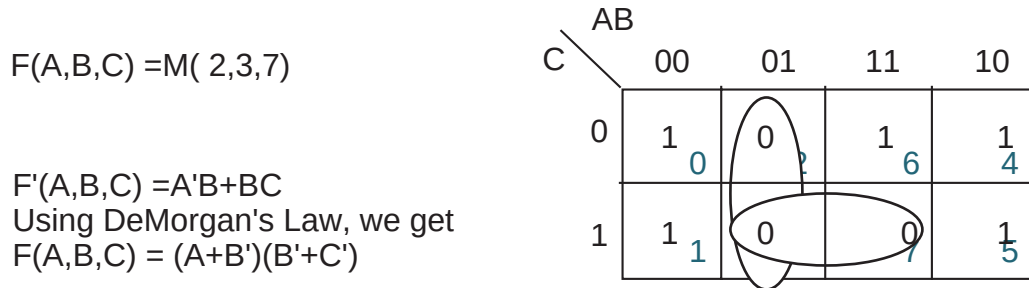
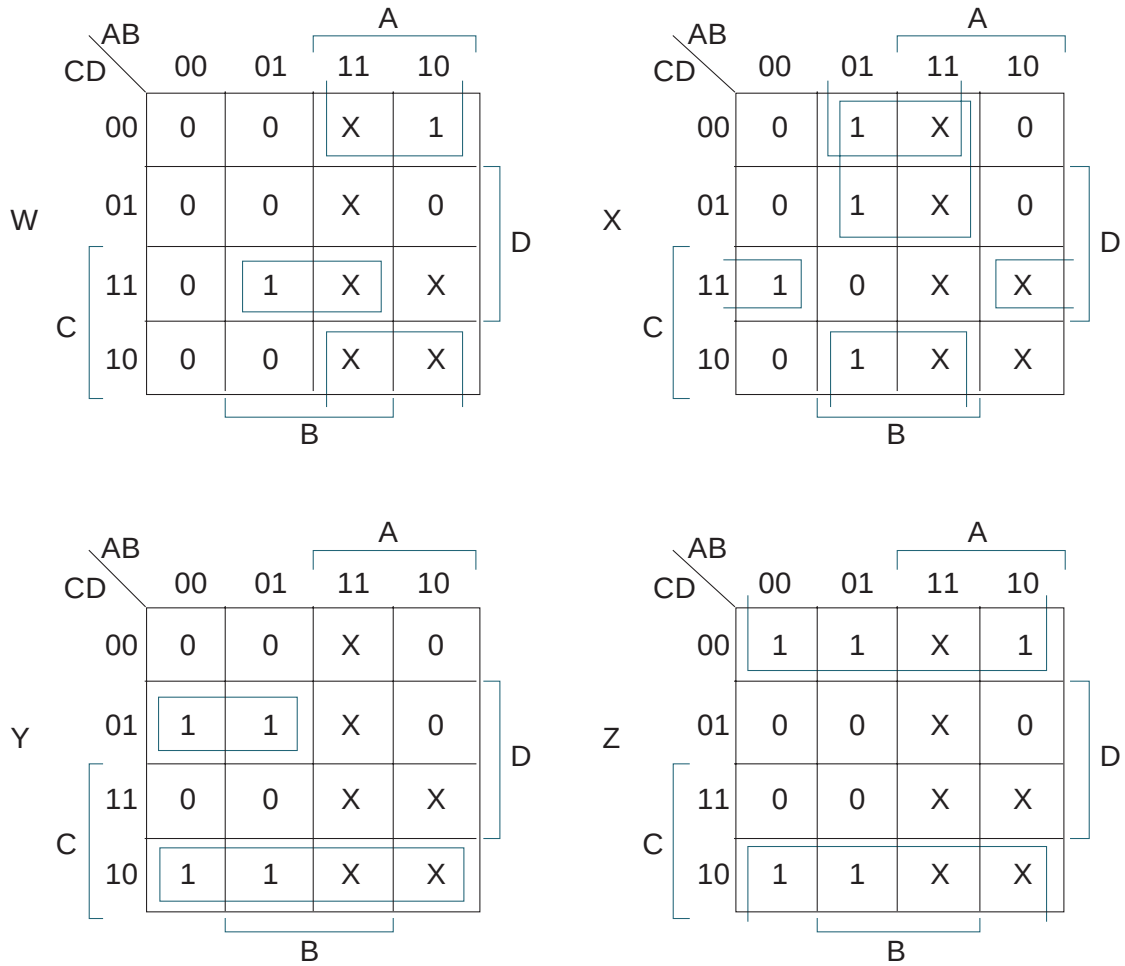


Figure 2: 3 variable KMAPS

Example BCD increment by one

A	B	C	D	W	X	Y	Z
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	0	1	1	0	1	0	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	0	0	0	0
1	0	1	0	X	X	X	X
1	0	1	1	X	X	X	X
1	1	0	0	X	X	X	X
1	1	0	1	X	X	X	X
1	1	1	0	X	X	X	X
1	1	1	1	X	X	X	X

BCD Incrementer (KMAPS)



$$W = B C D + A D'$$

$$Y = A' C' D + C D'$$

$$X = B C' + B D' + B' C D$$

$$Z = D'$$

Figure 3: KMAPS for BCD Incrementer

Solution

$$\begin{aligned}
 f_W(A, B, C, D) &= \sum m(7, 8) + \sum d(10, 11, 12, 13, 14, 15) \\
 &= B C D + A \overline{D} \\
 f_X(A, B, C, D) &= \sum m(3, 4, 5, 6) + \sum d(10, 11, 12, 13, 14, 15) \\
 &= \overline{B} C D + B \overline{C} + B \overline{D} \\
 f_Y(A, B, C, D) &= \sum m(1, 2, 5, 6) + \sum d(10, 11, 12, 13, 14, 15) \\
 &= \overline{A} \overline{C} D + C \overline{D} \\
 f_Z(A, B, C, D) &= \sum m(0, 2, 4, 6, 8) + \sum d(10, 11, 12, 13, 14, 15) \\
 &= \overline{D}
 \end{aligned}$$

Example Two Bit Comparator

A	B	C	D	F_1	F_2	F_3
0	0	0	0	1	0	0
0	0	0	1	0	1	0
0	0	1	0	0	1	0
0	0	1	1	0	1	0
0	1	0	0	0	0	1
0	1	0	1	1	0	0
0	1	1	0	0	1	0
0	1	1	1	0	1	0
1	0	0	0	0	0	1
1	0	0	1	0	0	1
1	0	1	0	1	0	0
1	0	1	1	0	1	0
1	1	0	0	0	0	1
1	1	0	1	0	0	1
1	1	1	0	0	0	1
1	1	1	1	1	0	0

$$F_1 = 1 \text{ if } AB = CD$$

$$F_2 = 1 \text{ if } AB < CD$$

$$F_3 = 1 \text{ if } AB > CD$$

Design Example: Two Bit Comparator

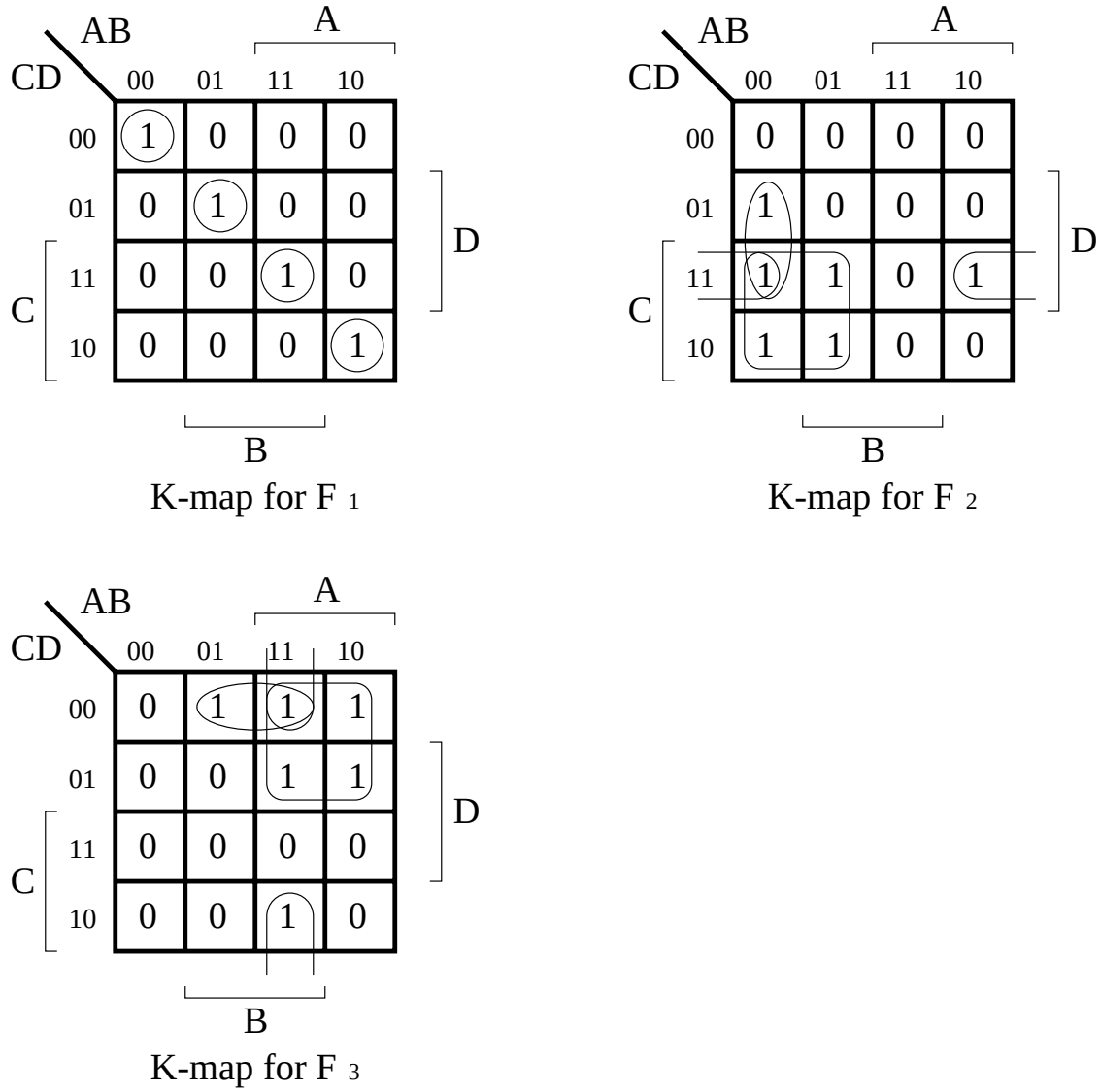


Figure 4: KMAPS for Two Bit Comparator

$$F_1 = \overline{A} \overline{B} \overline{C} \overline{D} + \overline{A} B \overline{C} D + A B C D + A \overline{B} C \overline{D}$$

$$F_2 = \overline{A} \overline{B} D + \overline{A} C + \overline{B} C D$$

$$F_3 = B \overline{C} \overline{D} + A \overline{C} + A B \overline{D}$$

Introduction to Espresso

Introduction

Espresso can be used for two-level Boolean function minimization. It combines many of the best heuristic techniques developed in earlier CAD programs such as *mini* and *presto*. It can be obtained from the University of California, Berkeley. However, it is available in the group of CAD software tools available on Eos at N. C. State University. In order to obtain access to espresso, add the CAD software tool locker using the following command:

```
add octtools
```

You can then run espresso using the command

```
espresso infile > outfile
```

where *infile* is the name of a file containing the inputs for a given Boolean function and *outfile* is the name of the file used for the output from espresso. You can also use the standard input to input the commands to espresso since it is written to accept the standard input (the keyboard) and to write to the standard output (the crt). The next section gives examples of using espresso to minimize Boolean functions.

Espresso Examples

Consider the example given in Fig. 2.56 on page 90 of the text (Contemporary Logic Design by Randy H. Katz) where

$$F(A, B, C, D) = \sum m(4, 5, 6, 8, 9, 10, 13) + \sum d(0, 7, 15) \quad (1)$$

The input for espresso is given by the following

```
.i 4
.o 1
.ilb A B C D
.ob F
.p 10
0100 1
0101 1
0110 1
1000 1
1001 1
1010 1
1101 1
0000 -
0111 -
1111 -
.e
```

The output from espresso is given by

```
.i 4
.o 1
.ilb A B C D
.ob F
.p 3
1-01 1
10-0 1
01-- 1
.e
```

Thus, we have

$$F(A, B, C, D) = A\overline{C}D + A\overline{B}\overline{D} + \overline{A}B \quad (2)$$