Camlin Page
Date / /

Number Base conversion -

1) Binary to Decimal -

$(\underset{3}{1}\underset{2}{0}\underset{1}{1}\underset{0}{0}.\underset{-1}{0}\underset{-2}{1}\underset{-3}{1})_2 \longrightarrow 2^3 + 2^1 + 2^{-2} + 2^{-3}$

$= 8 + 2 + 1/4 + 1/8$

$= 10 + 0.25 + 0.125$

$= 10 + 0.375$

$= 10.375$

2) Octal to Decimal

$(\underset{2}{3}\underset{1}{0}\underset{0}{6}.\underset{-1}{4})_8 = 3 \times 8^2 + 6 \times 8^0 + 4 \times 8^{-1}$

$= 192 + 6 + 4/8$

$= 198 + 0.5$

$= 198.5$

3) Hexadecimal to Decimal -

$(\underset{2}{3}\underset{1}{0}\underset{0}{6}.\underset{-1}{D})_{16} = 3 \times 16^2 + 6 \times 16^0 + 13 \times 16^{-1}$

$= 3 \times 256 + 6 + 13/16$

$= 768 + 6 + 0.8125$

$= \cancel{7}\ 774 + 0.8125$

$= 774.8125$

Scanned by CamScanner

$(153)_{10} \rightarrow (\ )_8$ Decimal to octal

$$
\begin{array}{lll}
8 & 153 & \\
8 & 19 & 1 \\
 & 2 & 3
\end{array}
$$

$= (231)_8$

$153 \rightarrow$

$(0.6875)_{10} \rightarrow (\ )_2$ Decimal to Binary

$0.6875 \times 2 = 1.375 \quad 1$

$0.375 \times 2 = 0.75 \quad 0$

$0.75 \times 2 = 1.5 \quad 1$

$0.5 \times 1 = 1$

$(0.1011)_2$

$(0.513)_{10} \rightarrow (\ )_8$ Decimal to Octal

$0.513 \times 8 = 4.104 \quad 4$

$0.104 \times 8 = 0.832 \quad 0$

$.832 \times 8 = 6.656 \quad 6$

$0.656 \times 8 = 5.248 \quad 5$

$0.248 \times 8 = 1.984 \quad 1$

$0.984 \times 8 = 7.872 \quad 7$

$(0.513)_{10} \rightarrow (0.406517\ldots)_8$

Scanned by CamScanner

Binary to octal

$\underbrace{101}_{5}\underbrace{101}_{5}\underbrace{101}_{5}\underbrace{101}_{5}.\underbrace{101}_{5}\underbrace{100}_{4}$

$(5555.54)_8$

Binary to Hexadecimal -

$\underbrace{0101}_{5}\underbrace{1110}_{E}\underbrace{0101}_{5}.\underbrace{1110}_{E}\underbrace{1100}_{C}$

$(5E5.EC)_{16}$

octal to Binary -

$(175)_8 = 001111101$

Hexadecimal to Binary -

$(395.A)_{16} = (001110010101.1011)_2$

Scanned by CamScanner

Camlin Page
Date / /

1's & 2's complement -

1's →

```
  1010100        1010100
- 1000100      + 0111011
               ① 0001111
                      +1
                 0010000  Ans
```

2's →

```
  1000100        1000100
- 1010100      + 0101100
                 1110000
```

no carry

− 0010000

$(BC25)_{16} + (1B2E)_{16}$

```
  ① ①
  BC25          246
+ 1B2E           49
  D753          295
```

$-(2BCD) + 53BA$

```
  4 2 1 11
  5 3 B A
  2 B C D
  2 7 E D
```

Scanned by CamScanner

27
2 3 1
1 1

Camlin Page
Date / /

Signed Numbers- Signed no. are those numbers which can represent both +ve & -ve no.

Representatn of Signed numbers-
The negative numbers in binary system is represented by the following 3 methods.

1) Signed magnitude
2) 1's complement
3) 2's complement

1) Sign Magnitude Representatn-
In Sign magnitude representatn, an extra bit is added at the MSB to represent sign of that number. this additional bit is usually known as Sign bit.

1 → for -ve
0 → for +ve

11011101 = (-93)

Signed magnitude Representatn can accomodate numbers in the range $-(2^{n-1}-1)$ to $+(2^{n-1}-1)$

n → no. of bits

Scanned by CamScanner

2) 1's complement representatn-

positive number are written as straight binary form with MSB = 0

→ if we take 1's complement of any number, resultant no. will be having same magnitude but opposite sign.

+18 00010010

−18 11101101

1's complement rep. can accomodate numbers in the range $-(2^{n-1}-1)$ to $+(2^{n-1}-1)$, $n \rightarrow$ no. of bits.

3) 2's complement Representatn-

positive no. are represented as straight binary no. with MSB = 0.

→ if we take 2's complement of any number the resultant number will be having same magnitude but opposite sign.

→ when the no. is −ve, the magnitude is ~~a~~ written by taking 2's complement of that no. and MSB = 1

+18 00010010

−18 11101110

Scanned by CamScanner

Advantage -
it contains only one type of 0. the other 2 representation have both a positive 0 and a -ve 0.

	Signed		1's comp.		2's comp.	
000	1000	0000	111	000(0)	~~1~~000	000
001	1001	0001	110	001(1)	111	001
010	1010	0010	101	010(2)	110	010
011	1011	0011	100	011(3)	101	011
100	1100	0100	011	100(-3)	100	100
101	1101	0101	010	101(-2)	011	101
110	1110	0110	001	110(-1)	010	110
111	1111	0111	000	111(-0)	001	111

(-0)	100	000	111	000	000	000
(-1)	101	001	110	001	111	001
(-2)	110	010	101	010	110	010
(-3)	111	011	100	011	101	011
					100	

(Since 2's complement representation has only single 0.) it should accomoradate one more no. than the other 2 sign representation. (therefore the 2's complement representation can accommodate numbers in the range $+(2^{n-1}-1)$ to -2^{n-1}

Scanned by CamScanner

Camlin Page
Date / /

floating point Number-
In this representation a number expressed by a fraction times some power of 10.

ex- 275.93801

$= \underbrace{0.27593801}_{\text{Mantissa}} \times \underbrace{10^3}_{\text{exponent/characteristic}}$

floating point no. is always represented in the following form.

$M \cdot r^E$

M → Mantissa
r → The Radix / base
E → Exponent.

0.0027593801
0.2759380 × 10^{-2}
This procedure of expressing the MSB of the mantissa with a non zero digit is called normalization and floating point no. is said to be normalized.

Scanned by CamScanner

Camlin Page
Date / /

codes - when no. letters or words are represented by a special group of symbols, we say that they are being encoded, and the group of symbols is called code.

Binary coded Decimal - if each digit of a decimal number is represented by its 4 bit binary equivalent the result is a code called BCD. this is weighted code. the weights in the BCD code are 8, 4, 2, 1. therefore, it is also known as 8421 code.

Decimal	BCD
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

$(437)_{10} = (01000110111)_{BCD}$.

$\left.\begin{array}{l} 1010 \\ 1011 \\ 1100 \\ 1101 \\ 1110 \\ 1111 \end{array}\right]$ Invalid BCD code.

BCD Additn -

BCD digits are added as strict binary number, if the result is greater that 9 or carry is generated then 0110 (6) is added in the result to get the final result.

Scanned by CamScanner

Camlin Page
Date / /

ex Add 28 + 33 in BCD code.

28	0010 1000
+ 33	0011 0011
	0101 1011
	> 9
	+ 0110
	1 0001

1

0101 0001

0110 0001

61

928 + 419

1

1001	0010	1000
0100	0001	1001
11010	0100	1 0001 > 9
0110		+ 0110
1 0011		1 0111
1 3	4	7

0011
1
0100

$(1347)_{10}$

Scanned by CamScanner

Camlin Page
Date / /

Number System

Weighted no. sys.
- Decimal
- Binary
- Octal
- Hexadecimal
- BCD
- 2421

Unweighted no. sys.
- Gray code
- Excess-3 code

Excess-3 code –

→ This is a 4-bit unweighted code and can be obtained from the corresponding value of BCD code by adding 3 to each coded no.

→ This code is self complement in nature. i.e., ~~the~~ 1's complement of the coded no. yields 9's complement of the no. itself.

→ This code is used in arithmetic circuit bcoz of its property of self complementation

Scanned by CamScanner

Decimal	BCD	Excess-3
0	0000	0011
1	0001	0100
2	0010	0101
3	0011	0110
4	0100	0111
5	0101	1000
6	0110	1001
7	0111	1010
8	1000	1011
9	1001	1100

$$(27)_{10} \longrightarrow (01011010)_{\text{excess-3}}$$

$$\downarrow 9's \qquad\qquad \downarrow 1's$$

$$(72)_{10} \longleftarrow \underbrace{1010}\,\underbrace{0101}$$

Excess-3 Additn -

Digits are added as strait binary number and in the result 0011 (3) is subtracted if there is no carry out and if there is a carry out, add 0011 (3)

This code is used in arithmetic circuits bcoz of its property of self complementation.

Scanned by CamScanner

Camlin Page
Date / /

Add $(269)_{10} + (168)_{10}$ in Excess 3 code.

$(269)_{10} \rightarrow$ 0101 1001 1100
$(168)_{10} \rightarrow$ 0011 1001 1011
1000 10010 10111
−0011 +0011 +0011
1100 0110 11010

0101 1001 1100
0100 1001 1011
1010 10010 10111
−0011 +0011 +0011
0111 0110 1010
4 3 7

Ans $(437)_{10}$

Gray code:

→ In gray code each no. differs from its succeeding and preceding code by only one bit. it is also called unit distance code or reflected code.
→ it is not a weighted code.
→ it finds applicatn in i/p o/p devices.

Scanned by CamScanner

Camlin Page
Date / /

Advantage of gray code -

Binary to gray →

→ MSB in the gray code is same as corresponding digit in binary number.

→ Starting from left to right, add each adjacent pair of binary digits to get next.

→ If carry is generated then discard it.
neglect

$$
\begin{array}{cccc}
1 & 0 & 1 & 1 \\
\downarrow & \oplus & \oplus & \oplus \\
1 & 1 & 1 & 0
\end{array}
$$

Gray to Binary -

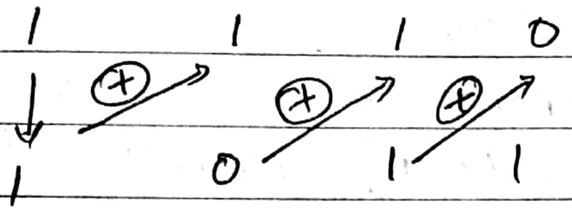
→ MSB in binary is the same as corresponding digit in gray code.

→ Add each binary digit generated to the next significant bit of the gray code.

→ Discard carry if generated.

Scanned by CamScanner

Camlin Page
Date / /



Application —

ASCII code —

→ The ASCII stands for American Standard code for Information Interchange it is a 7 bit code. The 8th bit is used for parity.

→ ASCII code widely used in small computers peripherals instruments and communication devices.

→ ~~The~~ A newer version of ASCII is ASCII-8. it is used in larger machines.

7 bits can represent upto - 128 characters
8 bits — " — - 256 char

a = 09~~7~~ → 0110000l

A = 065 → 01000001

2	97	
2	48	1
2	~~24~~	0
2	12	0
2	6	0
2	3	0
	1	1

Scanned by CamScanner

Camlin Page
Date / /

Boolean Algebra -

→ Boolean algebra is basically 2 values i.e. (0,1) functn.

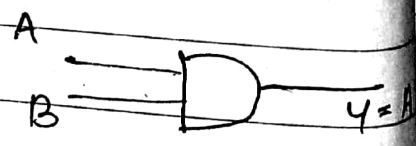
→ it had its applicatn to statements and sets which are either true or false. now a days it has wide applicatns to switching ckts, networks etc.

→ Basically there are 3 operations the case of Boolean algebra.

1) AND (·)
2) OR (+)
3) NOT (')

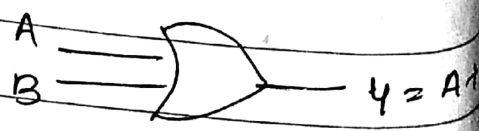
AND →

A	B	Y
0	0	0
0	0	0
1	0	0
1	1	1



OR →

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

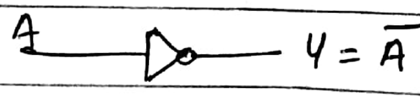


Scanned by CamScanner

Camlin	Page
Date	/ /

NOT →

A	Y
0	1
1	0



Boolean algebraic laws -

1) commutative law -

$$A + B = B + A \quad \text{and} \quad A \cdot B = B \cdot A$$

2) Associative law -

$$A + (B + C) = (A + B) + C = A + B + C$$

and $A \cdot (B \cdot C) = (A \cdot B) \cdot C = ABC$

3) Distributive law -

$$A(B + C) = AB + AC$$

$$\cancel{(A + B)(C + D) = AC + AD + BC + BD}$$

$$A + (B \cdot C) = (A + B)(A + C)$$

In boolean algebra, addition is distributive over the multiplication whereas in general algebra it is not so.

Camlin Page
Date / /

Boolean Algebraic Theorems -

1) AND operation Theorem.

$A \cdot A = A$

$A \cdot 0 = 0$

$A \cdot 1 = A$

$A \cdot \bar{A} = 0$

2) Involution Theorem -

$(A')' = \bar{\bar{A}} = A$

3) OR operation Theorem -

$A + A = A$

$A + 0 = A$

$A + 1 = 1$

$A + \bar{A} = 1$

4) De Morgan's Theorem.

There are 2 theorems -

$\overline{A \cdot B} = \bar{A} + \bar{B}$

$\overline{A + B} = \bar{A} \cdot \bar{B}$

5) Transposition Theorem

$(A + B)(A + C) = A + BC$

6) Distribution Theorem.

$A + BC = (A+B)(A+C)$

$AB + A\bar{B} + \bar{A}\bar{B}$

$A + \bar{A}\bar{B}$

$(A+\bar{A})(A+\bar{B})$

$\bar{A} + A\bar{B} = \bar{A} + \bar{B}$

$A + \bar{A}\bar{B} = (A+\bar{A})$

$= A +$

$(\bar{x}+y)(x+z)$

$\bar{x}z + yz$

$(A+B)(A+\bar{B})(\bar{A}+B)($

$= 0$

$(A+B+C)(\bar{A}+\bar{B}+C)(A$

$= (A+BC)$

$(A+B)(A+C)$

$A + AC + AB + B$

$A(1+C) + AB + B$

$A + AB + BC$

$A(1+B) + BC$

$A + BC$

Scanned by CamScanner

Camlin Page
Date / /

Consensus Theorem

→ This theorem is used to eleminate redundant term.

→ contains 3 variables

→ each variable is used twice.

→ only one variable is in complemented or uncomplemented form.

→ Then the related terms to that complemented or uncompleemented variable is the answer.

$AB + \bar{A}C + \underbrace{BC}_{\text{Redundant.}} = AB + \bar{A}C$

1) $A + AB = A$

2) $A \cdot (A+B) = A$

3) $A + \bar{A}B = (A+\bar{A})(A+B) = A+B$

4) $A \cdot (\bar{A}+B) = AB$

5) $AB + A\bar{B} = A$

6) $(A+B)\cdot(A+\bar{B}) = \underset{A+B\cdot\bar{B}}{\cancel{A\cdot(B+\bar{B})}} = A$

7) $A\cancel{B} + \cancel{\bar{A}}C \neq A$

$AB + B\bar{C} + AC = B\bar{C} + AC$

$AB + \bar{B}C + AC = AB + \bar{B}C$

$\bar{A}B + BC + AC = \bar{A}B + AC$

$(A+B)(\bar{A}+C)(B+C) = (A+B)(\bar{A}+C)$

Scanned by CamScanner

Camlin Page
Date / /

Transposition Theorem -

$(A+B)(\bar{A}+C) = AC + \bar{A}B$

P

$AB + \bar{A}C = (A+C)(\bar{A}+B)$
$= AB + \bar{A}C + BC = AB + \bar{A}C$

7) $AB + \bar{A}C = (A+C)(\bar{A}+B)$.

8) $(A+B)(\bar{A}+C) = AC + \bar{A}B$

9) $AB + \bar{A}C + BC = AB + \bar{A}C$

10) $(A+B)(\bar{A}+C)(B+C) = (A+B)(\bar{A}+C)$

consensus $AB + \bar{A}C + BC$

$ABC + AB\bar{C} + \bar{A}BC + \bar{A}\bar{B}C + ABC + \bar{A}BC$

$ABC + AB\bar{C} + \bar{A}BC + \bar{A}\bar{B}C$

$AB + \bar{A}C$

$(A+B)(\bar{A}+C)(B+C) = (A+B)(\bar{A}+C)$

$= (A\cancel{C} + \bar{A}B + \cancel{B}C)$

$= (AC + \bar{A}B)(B+C)$

$= ABC + AC + \bar{A}B + \bar{A}BC$

$= AC + \bar{A}B$

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

$AB + \bar{A}C + BC$

$\cancel{B}\cdot(A+C)\cdot(\bar{A}+B) + BC$

Q-> Simplify -> $(A\bar{B}(C+BD) + \overline{AB})C$

$(\overline{x+y})(\overline{\bar{x}+\bar{y}})$

Scanned by CamScanner

Camlin Page
Date / /

Complementary Theorem -
for obtaining complement expression-
1) change each OR sign by AND sign &
vice versa.
2) complement any '0' or '1' appearing in
expression.
3) complement the individual literals.
each variable

Duality Theorem -
→ Dual expression is used to convert
+ve logic to -ve logic.
→ In signal time domain to f domain or
frequency domain to time domain.

T = 0 0 f = ∞ 1

f = ∞ 1 T = 0 0

+ve logic →

logic 0 → 0V. logic 0 → -5V.

logic 1 → +5V. logic 1 → 0V.

-ve logic → logic 0 → 5V.

logic 1 → 0V.

Scanned by CamScanner

Camlin Page
Date / /

AND.

+ve logic AND gate →

0	0	0
0	1	0
1	0	0
1	1	1

-ve logic OR gate →

		Y
1	1	0
1	0	0
0	1	0
0	0	1

+ve logic AND gate = -ve logic OR gat

To find a dual expression.

1) convert AND ⟷ OR

1 ⟷ 0

keep variable as it is.

$$A B \bar{C} + A B C + \bar{A} B C$$

⇓ Dual

$$\cancel{A} (A + B + \bar{C}) \cdot (A + B + C) \cdot (\bar{A} + B + C)$$

→ for any logical expression, if we take 2 times Dual, we get same given expression as previous

→ if ~~we~~ ~~take~~ one time dual is same as the original expression

Camlin Page
Date / /

then it is called self dual.

$$AB + BC + AC \longleftrightarrow (A+B)\cdot(B+C)\cdot(A+C)$$
$$\Rightarrow (A+BC)\cdot(B+C)$$
$$\Rightarrow AB + AC + BC + BC$$
$$\Rightarrow AB + BC + AC$$

→ for n variables maximum possible self dual ~~statement~~ functn → $2^{2^{n-1}}$

→ with n variables, maximum possible distinct logic function $= 2^{2^n}$.

$n = 2 \longrightarrow A \cdot B$

AB	$A+B$	$\bar{A}$	0
$\bar{A}B$	$A+\bar{B}$	A	1
$A\bar{B}$	$\bar{A}+B$	B	$\bar{A}B + A\bar{B}$
$\bar{A}\bar{B}$	$\bar{A}+\bar{B}$	$\bar{B}$	$AB + \bar{A}\bar{B}$

Boolean Function

Canonical form
↓
all the terms contains each variable

$\bar{A}BC + ABC + \bar{A}B\bar{C}$

Standard form
↓
all the terms do not have each variable.

$A + BC + A\bar{B}C$

Scanned by CamScanner

Camlin Page
Date / /

Minterms and Maxterms -

→ n - binary variables have 2^n possible combinations and each of these possible combination is called Minterm or standard product.

→ Maxterm is the complement of corresponding "Minterm"

$$M = \overline{m}$$

Minterm ⇒ $\Sigma m(---)$
Maxterm ⇒ $\pi M(---)$

for 3 variable -

A B C	Minterm =	Maxterm
0 0 0	$m_0 = \bar{A}\bar{B}\bar{C}$	$A+B+C$
0 0 1	$\bar{A}\bar{B}C$	$A+B+\bar{C}$
0 1 0	$\bar{A}B\bar{C}$	$A+\bar{B}+C$
0 1 1	$\bar{A}BC$	$A+\bar{B}+\bar{C}$
1 0 0	$A\bar{B}\bar{C}$	$\bar{A}+B+C$
1 0 1	$A\bar{B}C$	$\bar{A}+B+\bar{C}$
1 1 0	$AB\bar{C}$	$\bar{A}+\bar{B}+C$
1 1 1	ABC	$\bar{A}+\bar{B}+\bar{C}$

Scanned by CamScanner

Camlin Page
Date

each individual term in std SOP form is called Minterm

SOP form-

→ The SOP expression usually takes the form of 2 or more variable ANDed together.

$$Y = \bar{A}BC + A\bar{C} + AB$$

→ SOP expression is used most often because it lends itself nicely to the development of truth tables and timing diagrams.

→ SOP form are used to write logical expression for the O/p becoming logic 1.

A	B	C	O/p (Y)
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

$$f(A,B,C) = \Sigma m(3,5,6,7)$$
$$Y = \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC$$

Scanned by CamScanner

POS (product of sum) forms.

The POS expression usually takes the form of 2 or more ORed variables within parantheses ANDed with 2 or more other variable within paranthses.

$$f(A,B,C) = \pi M(0,1,2,4)$$

$$Y = (A+B+C)(A+B+\bar{C})(\bar{A}+\bar{B}+C)(\bar{A}+B+C)$$

KARNAUGH MAP –

→ K-Map is a graphical tool to simp boolean expression. it

→ K-map technique may be used for any number of variables but it is generally used upto 5 or 6 varia beyond which it becomes very cumbers

In n-variables K-map there are 2^n squares. each square represents one minterm or maxterm.

Camlin Page
Date / /

$\cancel{(A\bar{B})}$

$(A\bar{B}(C+BD) + \overline{AB})\,C$

$= (A\bar{B}C + A\bar{B}\cdot BD + \bar{A}+\bar{B})\,C$

$= A\bar{B}C + \bar{A}C + \bar{B}C$

$= \bar{B}C + \bar{A}C$

$= C(\bar{A}+\bar{B})$

$(\overline{x+y})\,(\overline{\bar{x}+\bar{y}})$

$= (\bar{x}\cdot\bar{y})\cdot(x\cdot y)$

$= 0$

$= \bar{A}\bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}\bar{C}D + A\bar{B}\bar{C}\bar{D} + A\bar{B}\bar{C}D$

$= \bar{A}\bar{B}\bar{C}(D+\bar{D}) + A\bar{B}\bar{C}(D+\bar{D})$

$= \bar{A}\bar{B}\bar{C} + A\bar{B}\bar{C}$

$= \bar{B}\bar{C}(A+\bar{A})$

$= \bar{B}\bar{C}$

$ABCD + AB\overline{CD} + \overline{AB}CD$

$ABCD + AB\overline{CD} + ABCD + \overline{AB}CD$

$AB + CD$

~~can~~ obtain canonical form -

$f(x,y,z) = [(x+y') + (y+z')']' + yz$

$= (x+y')'\cdot(y+z')'' + yz$

$= \cancel{x+}\,(x'\cdot y'')\cdot(y+z') + yz$

$= (x'\cdot y)\cdot(y+z') + yz$

$= \cancel{y}\; x'y + x'yz' + yz$

Scanned by CamScanner

Camlin Page
Date / /

2 Variable K-Map –

2 Variable K-map has 4 cells and hence 4 minterms (maxterms)

A \ B	$\bar{B}$ 0	B 1
$\bar{A}$ - 0	m_0	m_1
A - 1	m_2	m_3

→ for SOP

A \ B	B ← 0	$\bar{B}$ ← 1
A ← 0	M_0	M_1
$\bar{A}$ ← 1	M_2	M_3

3 Variable K-map –

A \ BC	$\bar{B}\bar{C}$ 00	$\bar{B}C$ 01	BC 11	$B\bar{C}$ 10
$\bar{A}$ 0	m_0	m_1	m_3	m_2
A 1	m_4	m_5	m_7	m_6

A \ BC	B+C 00	$B+\bar{C}$ 01	$\bar{B}+\bar{C}$ 11	$\bar{B}+C$ 10
A ← 0	M_0	M_1	M_3	M_2
$\bar{A}$ ← 1	M_4	M_5	M_7	M_6

Scanned by CamScanner

4-Variable K-Map-

AB \ CD	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}

Representation of truth table on K-MAP—

A	B	Y
0	0	1
0	1	0
1	0	1
1	1	0

	0	1
0	1	0
1	1	0

Looping groups of 2 →
looping a pair of adjecent 1's in a K-map eliminates the variable that appears in complemented and uncomplemented form

	$\bar{B}\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$
$\bar{A}$	0	1	0	0
A	0	1	0	0

ABC - 3
$\bar{B}C$

$\bar{A}$	0	0	0	0
A	0	1	1	0

AC

Scanned by CamScanner

Camlin | Page
Date / /

		$\bar{B}\bar{C}$			$B\bar{C}$
$\bar{A}$	0				
A	1	1			1

$A\bar{C}$

$AB \backslash CD$	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$			1	1
$\bar{A}B$				
AB				
$A\bar{B}$	1			1

$$A\bar{B}\bar{D} + \bar{A}\bar{B}C$$

Looping groups of four (Quads)
looping a quad of 1's eliminates the 2 variables that appear in both complemented and uncomplemented form.

$\bar{A}$	0	0	0	0
A	1	1	1	1

$Y = A$

$AB \backslash CD$	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$				
$\bar{A}B$				
AB	1	1	1	1
$A\bar{B}$				

$Y = AB$

Scanned by CamScanner

Camlin Page
Date / /

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$				
$\bar{A}B$		1	1	
AB		1	1	
$A\bar{B}$				

$Y = BD$

	$\bar{C}\bar{D}$			$C\bar{D}$
AB	1			1
$A\bar{B}$	1			1

$Y = A\bar{D}$

	$\bar{C}\bar{D}$			$C\bar{D}$
$\bar{A}\bar{B}$	1			1
$A\bar{B}$	1			1

$Y = \bar{B}\bar{D}$

Looping groups of Eight (octets)

1	1	1	1
1	1	1	1

looping an octet of 1's eliminates the 3 variables that appear in both comp & uncomp. form

Q find out simplified boolean function -

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

1) By sop →

$$f(A,B) = \bar{A}\bar{B} + \bar{A}.B + A\bar{B}$$
$$= \bar{A} + \bar{B}$$

2) BY K-map.

A \ B	$\bar{B}$ (0)	B (1)
$\bar{A}$ 0	1	1
A 1	1	0

$$f(A,B) = \bar{B} + \bar{A}$$

2^i, $i = 0, 1 \ldots$

$2^0 = 1$

$2^1 = 2$ (pair)

$2^2 = 4$ (quad)

$2^3 = 8$ (octet)

Camlin Page
Date / /

Simplify boolean functn _ using K-map →

$F = \bar{A}\bar{B} + \bar{A}B + A\bar{B}$

A \ B	$\bar{B}$ 0	B 1
$\bar{A}$ 0	1 (0)	1 (1)
A 1	1 (2)	0 (3)

Simplify boolean functn

$f(A, B) = \Sigma(0, 1, 2)$

Simplify boolean functn.

$f(A, B, C) = \Sigma(2, 3, 6, 7)$

A \ BC	00	01	11	10
0	(0)	(1)	1 (3)	1 (2)
1	(4)	(5)	1 (7)	1 (6)

$f(A, B, C)\ f = B$

$f(A, B, C) = \Sigma m(0, 1, 3, 5, 7)$

A \ BC	00	01	11	10
$\bar{A}$ 0	1 (0)	1 (1)	1 (3)	(2)
A 1	(4)	1 (5)	1 (7)	(6)

$f(A, B, C) = C + \bar{A}\bar{B}$

Scanned by CamScanner

Simplify →

$f(A, B, C, D) = \Sigma(1, 2, 3, 4, 7, 9, 10, 12)$

$2^4 = 16$ cells

AB \ CD	00	01	11	10
00	0	1 1	1 3	1 2
01	1 4	5	1 7	6
11	1 12	13	15	14
10	8	1 9	11	1 10

$f(A,B,C,D) = \cancel{\bar{B}}\ \bar{B}\bar{C}D + \bar{B}C\bar{D} + B\bar{C}\bar{D} + \bar{A}CD$

$(\bar{A}\bar{B}D + \bar{A}\bar{B}C)$*

Q→ $4 = m_3 + m_7 + m_9 + m_{10} + m_{11} + m$

$2^4 = 16.$

AB \ CD	00	01	11	10
00	0	1	1 3	2
01	4	5	1 7	6
11	12	13	1 15	14
10	8	1 9	1 11	1 10

Camlin Page
Date / /

f~~(A, B,~~

$Y = \cancel{A}' CD + A\bar{B}D + A\overline{BC}$

$f(A,B,C,D) = \Sigma m(1,5,6,7,11,12,13,15)$

AB \ CD	00	01	11	10
00	0	1 (1)	3	2
01	4	1 (5)	1 (7)	1 (6)
11	1 (12)	1 (13)	1 (15)	14
10	8	9	1 (11)	10

→ Redundant term

$\cancel{\bar{A}\bar{B}C} + \cancel{AB\bar{C}} + \cancel{ABC} + \cancel{A}$

$A\cancel{B}\bar{C} + \cancel{\bar{A}}\bar{B}\cancel{C} + \bar{A}\cancel{B}\cancel{C} + \cancel{A}B\cancel{C}$

$\bar{A}\bar{C}D + AB\bar{C} + ACD + \bar{A}BC$

$f(A,B,C,D) = \Sigma m(2,3,6,7,8,9,10,11,12,13,14,15)$

AB \ CD	00	01	11	10
00	0	1	1 (3)	1 (2)
01	4	5	1 (7)	1 (6)
11	1 (12)	1 (13)	1 (15)	1 (14)
10	1 (8)	1 (9)	1 (11)	1 (10)

$C + A$

Q → (0,1,2,4,5,6,8,9,12,13,14) → $y' + w'z' + xz'$

Dont care conditn.

Q simplify → $f = A'B'C' + B'CD' + A'BCD' + AB'C'$

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

Scanned by CamScanner

$f(A,B,C,D) = \Sigma(2,3,4,5,13,15) + \Sigma d(8,9,10,11)$

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

$f(A,B,C,D) = AD + \overline{A}B\overline{C} + \overline{B}C$

$f(A,B,C,D,E) =$
$\Sigma m(0,5,6,8,9,10,11,16,20,24,25,26,27,29,31)$

$2^5 = 32$ cells.

AB \ CD	00	01	11	10
00	1, 0	1	2	3
01	4	1, 5	7	1, 6
11	12	13	15	14
10	1, 8	1, 9	1, 11	1, 10

	00	01	11	10
00	1, 16	17	19	18
01	1, 20	21	23	22
11	26	1, 29	1, 31	30
10	1, 24	1, 25	1, 27	1, 26

Scanned by CamScanner

AB \ CDE	000	001	011	010	110	111	101	100
00	1_{0}	$_{1}$	$_{3}$	$_{2}$	1_{6}	$_{7}$	1_{5}	$_{4}$
01	1_{8}	1_{9}	1_{11}	1_{10}	$_{14}$	$_{15}$	$_{13}$	$_{12}$
11	1_{24}	1_{25}	1_{27}	1_{26}	$_{30}$	1_{31}	1_{29}	$_{28}$
10	1_{16}	$_{17}$	$_{19}$	$_{18}$	$_{22}$	$_{23}$	$_{21}$	1_{20}

$$B\bar{C} + \bar{A}\cdot\bar{C}\bar{D}\bar{E} + ABCE + \bar{A}\bar{B}CD\bar{E}$$
$$+ \bar{A}\bar{B}C\bar{D}E + A\bar{B}\cancel{C}\bar{D}\bar{E}$$

Q→ Obtain the canonical form →

$$f(x, y, z) = [(x+y') + (y+z')']' + yz$$

$$= [x + y' + y'\cdot z]' + yz$$
$$= [x + y'(1+z)]' + yz$$
$$= (x+y')' + yz$$
$$= x'\cdot y + yz$$
$$= x'y + yz$$
$$= x'y(z+z') + yz(x+x')$$
$$= x'yz + x'yz' + xyz + \cancel{x'yz}$$
$$= x'yz + x'yz' + xyz$$

Q Express the above boolean functn ~~into~~ ~~a of its minterms.~~ sum of minterms.

$$f(x,y,z) = \overset{0\,1\,1}{x'yz} + \overset{0\,1\,0}{x'yz'} + \overset{1\,1\,1}{xyz}$$

$$= m_3 + m_2 + m_7$$

$$= \Sigma \cancel{m} (0\ 2, 3, 7)$$

Q → change sop form into pos form -

Q express $(x+y)(x+z')$ in canonical form. in

$$f(x,y,z) = (x+y)(x+z)$$

$$= (x+y+z\cdot\bar{z})\cdot(x+z+y\cdot\bar{y})$$

$$f = (\overset{0}{x}+\overset{0}{y}+\overset{0}{z})(\overset{0}{x}+\overset{0}{y}+\overset{1}{\bar{z}})\cdot(\overset{0}{x}+\overset{0}{y}+\overset{0}{z})\cdot(\overset{0}{x}+\overset{1}{\bar{y}}+\overset{0}{z})$$

prod. ~~sum~~ of maxterms

$$f = M_0 \cdot M_1 \cdot M_2$$

$$f = \pi(0,1,2)$$

change sop into pos form

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

pos $\longrightarrow$ sop

$$f(x,y) = (x + y')(x' + y)(x' + y')$$
$$= xy + x'y'$$
$$= (x + y')\cdot(x' + yy')$$
$$= (x + y')\cdot x'$$
$$= x\cdot x' + x'y'$$
$$= x'y'$$

Q express the boolean functn in a sum of minterms —

$$f(x,y,z) = (x + y)\cdot(x + z') + (y + z')$$
$$= x + yz' + (y + z')$$
$$= x + y + z'(1 + y)$$

Scanned by CamScanner

Camlin Page
Date / /

$$= x + y + z'$$

$\hookrightarrow M_1$

$$= x \cdot 1 + y \cdot 1 + z' \cdot 1$$

$$= x(y+y')(z+z') + y \cdot (x+x')(z+z') + z'(x+x')(y+y')$$

$$= x(yz + yz' + y'z + y'z') + y(xz + xz' + x'z + x'z') + z'(xy + xy' + x'y + x'y')$$

$$= xyz + xyz' + xy'z + xy'z') + xyz + xyz' + x'yz + x'yz') + xyz' + xy'z' + x'yz' + x'y'z'$$

$$= m_7 + m_6 + m_5 + m_4 + m_7 + m_6 + m_3 + m_2 + m_6 + m_4 + m_2 + m_0$$

$$\Sigma (0, 2, 3, 4, 5, 6, 7)$$

Camlin Page
Date / /

Dual of

$$f(A, B, C, D E) = (A+B+C)\, D'E$$

$$= A \cdot B \cdot C + D' + E.$$

Obtain complement →

$$f = \overline{(AB + CD) \cdot E}$$

$$= \overline{AB + CD} + \overline{E}$$

$$= \overline{AB} \cdot \overline{CD} + \overline{E}$$

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

$$=$$

Tabulation method –

It is also known as the Quine Mccluskey method. The K-map method is convenient as long as the number of variables are limited to five or six. As the variables increases, excessive no. of square prevents the selection of adjecent squares. The tabulation method overcomes this difficulty.

Scanned by CamScanner

Camlin Page
Date /

Simplify the following boolean function by using the tabulation method $\rightarrow \Sigma(0,1,2,4,5,6,8,9,12,13,14)$

Ans $\rightarrow$ $f(A,B,C,D) = C' + A'D' + BD'$

~~S~~ find out minterms.

$$f(A,B,C,D) = \Sigma(0,1,2,4,5,6,8,9,12,13,14)$$

$$C' + A'D' + BD'$$

$$(C'+A')\cdot(C'+D') + BD'$$

B B A

$$(BD' + \underbrace{C' + A'}_{A})\cdot(BD' + C' + D')$$

BC

$$(D' + C' + A)\cdot(C'+A'+B)\cdot(C'+A'+D')$$

$$\Rightarrow (A+C+D')(\bar{A}+\bar{C}+D')\cdot(A'+B+C'+D)\cdot$$
$$(A'+B+C'+D')\cdot(A'+B+C'+D')\cdot$$
$$(A'+B'+C'+D')$$

$$\Rightarrow (A+B+C+D')\cdot(A+B'+C+D')\cdot(\bar{A}+B+\bar{C}+$$
$$(\bar{A}+\bar{B}+\bar{C}+\bar{D})\cdot(A'+B+C'+D)\cdot$$
$$(A'+B+C'+D')_{\times}\cdot(A'+B+C'+D')\cdot$$
$$(A'+B'+C'+D')_{\times}$$

find no. of minterms.

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

K-map.

wx \ yz	00	01	11	10
00	1_{0}	1_{1}	$_{3}$	1_{2}
01	1_{4}	1_{5}	$_{7}$	1_{6}
11	1_{12}	1_{13}	$_{15}$	1_{14}
10	1_{8}	1_{9}	$_{11}$	$_{10}$

$\Sigma m (0, 1, 2, 4, 5, 6, 8, 9, 12, 13, 14)$

S-2 Group binary representatn of the minterms according to the no. of 1's contained. ie. from the groups containing no 1's, one 1, two 1's and so on.

~~Minterm~~

Q Minimize logic functn $f(A, B, C, D) = \pi(4, 6, 10, 12, 13, 15)$

AB \ CD	00	01	11	10
00	$_{0}$	$_{1}$	$_{3}$	$_{2}$
01	0_{4}	$_{5}$	$_{7}$	0_{6}
11	0_{12}	0_{13}	0_{15}	$_{14}$
10	$_{8}$	$_{9}$	$_{11}$	$\boxed{0}_{10}$

$(\bar{B} + C + D) \cdot (A + \bar{B} + D) \cdot (\bar{A} + \bar{B} + \bar{D}) \cdot (\bar{A} + B + \bar{C} + D)$

Scanned by CamScanner

Camlin Page
Date /

Minterms	A	B	C	D	Group
0	0	0	0	0	
1	0	0	0	1	
2	0	0	1	0	
4	0	1	0	0	
5	0	1	0	1	
6	0	1	1	0	
8	1	0	0	0	
9	1	0	0	1	
12	1	1	0	0	
13	1	1	0	1	
14	1	1	1	0	

Grouping minterms according to the no. of

Minterms	A	B	C	D	Group.
0	0	0	0	0	0
1	0	0	0	1	
2	0	0	1	0	1
4	0	1	0	0	
8	1	0	0	0	
5	0	1	0	1	
6	0	1	1	0	
9	1	0	0	1	2
12	1	1	0	0	
13	1	1	0	1	
14	1	1	1	0	3

Camlin Page
Date / /

S2 → Any 2 minterms which differ from each other by only one variable can be combined.

→ The variable eliminated during the matching combination of minterm groups of 2 is denoted by a – in its original position.

Minterms	A	B	C	D
(0,1)	0	0	0	–
(0,2)	0	0	–	0
(0,4)	0	–	0	0
(0,8)	–	0	0	0
(1,5)	0	–	0	1
(1,9)	–	0	0	1
(2,6)	0	–	1	0
(4,5)	0	1	0	–
(4,6)	0	1	–	0
(4,12)	–	1	0	0
(8,9)	1	0	0	–
(8,12)	1	–	0	0
(5,13)	–	1	0	1
(6,14)	–	1	1	0
(9,13)	1	–	0	1
(12,13)	1	1	0	–
(12,14)	1	1	–	0

Scanned by CamScanner

8-(b) All the minterms in the adjacen[t] in above table are compared when minterms in one section are compared with those of the next section down only, the dashes must be in the same bit pos[ition] in the groups of two and o[nly] one variable must differ.

combination of minterm a groups of 4

Minterms	A	B	C	D
(0,1,4,5)	0	—	0	—
(0,1,8,9)	—	0	0	—
(0,2,4,6)	0	—	—	0
(0,4,1,5)	0	—	0	—
(0,4,2,6)	0	—	—	0
(0,4,8,12)	—	—	0	0
(0,8,1,9)	—	0	0	—
(0,8,4,12)	—	—	0	0
(1,5,9,13)	—	—	0	1
(1,9,5,13)	—	—	0	1
(4,5,12,13)	—	1	0	—
(4,6,12,14)	—	1	—	0
(4,12,6,14)	—	1	—	0

Camlin Page
Date / /

(8, 9, 12, 13)	1	—	0	—
(8, 12, 9, 13)	1	—	0	—

S-4 The comparing process should be carried out again in above table. In this case both dashes must be in the same bit position and only one other variable must be differ for matching.

We observe that generally in each group 2 terms are same therefore only one is to be taken in each group for matching.

Combination of minterms groups of 8.

(0, 1, 4, 5, 8, 9, 12, 13)	—	—	0	—
(0, 1, 8, 9, 4, 5, 12, 13)	—	—	0	—
(0, 2, 4, 6)	0	—	—	0
(0, 4, 1, 5, 8, 9, 12, 13)	—	—	0	—
(0, 8, 4, 12, 1, 5, 9, 13)	—	—	0	—
(4, 6, 12, 14)	—	1	—	0

Scanned by CamScanner

S-5 Repeat the process of grouping until the exhaustive search is completed.

prime Implicants table.

	A	B	C	D	Term
(0, 1, 4, 5, 8, 9, 12, 13)	–	–	0	–	$\bar{C}$
(0, 2, 4, 6)	0	–	–	0	$\bar{A}D'$
(4, 6, 12, 14)	–	1	–	0	BD'

→ prime Implicants.

$f(A,B,C,D) = C' + A'D' + BD'$

EPI table →

S-6

PI terms	Minterms	0	1	2	4	5	6	8	9	12	13	14
✓ C'	(0, 1, 4, 5, 8, 9, 12, 13)	X	X		X	X		X	X	X	X	
✓ A'D'	(0, 2, 4, 6)	X		X	X		X					
✓ BD'	(4, 6, 12, 14)				X		X			X		X

In this example there are 7 minterms whose columns have a single cross.

1, 2, 5, 8, 9, 13 & 14

0, 4, 6, 12

EPI = C', A'D' & BD'

required minimized functn

$f(A,B,C,D) = A'D' + BD' + C'$

Scanned by CamScanner

ex→ Determine the

Minimize the function

$f(w,x,y,z) = \Sigma(1,4,6,7,8,9,10,11,15)$

Determination of prime Implicants: 0001, 0100, 0110, 0111, 1000, 1001, 1010, 1011, 1111

	(a) wxyz	(b)	wxyz	(c)	wxyz
1	0001 ✓	~~1,9~~	-001		
(1) 4	0100 ✓	4,6	01-0	8,9,10,11	10--
~~6~~	~~0110~~	8,9	100- ✓	8,10,9,11	10--
8	1000 ✓	8,10	10-0 ✓		
6	0110 ✓	6,7	011-		
(2) 9	1001 ✓	9,11	10-1 ✓		
10	1010 ✓	10,11	101- ✓		
(3) 7	0111 ✓	7,15	-111		
11	1011 ✓	11,15	1-11		
(4) 15	1111 ✓				

Prime Implicants—

	w x y z	Terms
1,9	-001	$x'y'z$
4,6	01-0	$w'xz'$
6,7	011-	$w'xy$
7,15	-111	xyz
11,15	1-11	wyz
8,9,10,11	10--	wx'

Scanned by CamScanner

Camlin Page
Date / /

EPI table –

Terms	Minterms	1	4	6	7	8	9	10	11	15
✓ $x'y'z$	1, 9	X					X			
✓ $w'xz'$	4, 6		X	X						
$w'xy$	6, 7			X	X					
xyz	7, 15				X					X
wyz	11, 15								X	X
✓ wx'	8, 9, 10, 11					X	X	X	X	

EPI → $x'y'z$
$w'xz'$
wx'

1, 9, 4, 6, 8, 9, 10, 11 → covered

7, 15 → not covered.
↳ xyz

$f(w, x, y, z) = x'y'z + w'xz' + wx' + xyz$

Scanned by CamScanner

Camlin Page
Date / /

$f(A,B,C) = \Sigma m(0, 1, 5, 6, 7)$

$A \backslash BC$	$\bar{B}\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$
$\bar{A}$	1	1		
A		1	1	1

$f(A,B,C) = \underline{\bar{A}\,\bar{B} + AB + AC}$

$= \underline{\bar{A}\bar{B} + AB + \bar{B}C}$

$\downarrow$

EPI — 2

PI — 4

I = 5

→ each minterm is known as Implicant.

→ PI is product term which is obtain by compining max. possible cell.

→ EPI is a PI which is possible to combine in only one way & there is no alternative way.