REPRESENTATION OF ALPHANUMERIC INFO IN BINARY FORM ⇒ BINARY CODES

ONE UNIT OF ALPHANUMERIC INFO (A LETTER, SPECIAL CHARACTER, OR DIGIT) ⟷ CODING SYSTEM ⟷ A CODE WORD (A SEQUENCE OF BITS)

A CODE IS A COLLECTION OF CODE WORDS, $C_0, C_1, \ldots, C_q$ IN WHICH THERE IS A CODE WORD FOR EACH SYMBOL $S_0, S_1, \ldots, S_q$.

A) BINARY CODES FOR DECIMAL DIGITS

WEIGHTED CODES

- ARE POSITIONALLY WEIGHTED $2^3 2^2 2^1 2^0$
- e.g., **BCD** : BINARY CODED DECIMAL (OR **8421 CODE**)

⇒ THE CODE WORD FOR A DECIMAL DIGIT D IN BCD IS OBTAINED VIA $D = W_4 \times B_4 + W_3 \times B_3 + W_2 \times B_2 + W_1 \times B_1$
WHERE W_4, W_3, W_2, W_1 ARE WEIGHTS AND (B4 B3 B2 B1) = CODE WORD

NON-WEIGHTED CODES

- ARE NOT POSITIONALLY WEIGHTED
- e.g., **XS3** : EXCESS-3 CODE

⇒ CODE WORD FOR A DECIMAL DIGIT IN XS3 = CODE WORD FOR THE DECIMAL DIGIT IN 8421 + 0011
e.g. 0111 ⇐ 0100 + 0011
$(4)_{XS3}$ $(4)_{8421}$ $(3)_{8421}$

SOME BINARY CODES FOR DECIMAL DIGITS

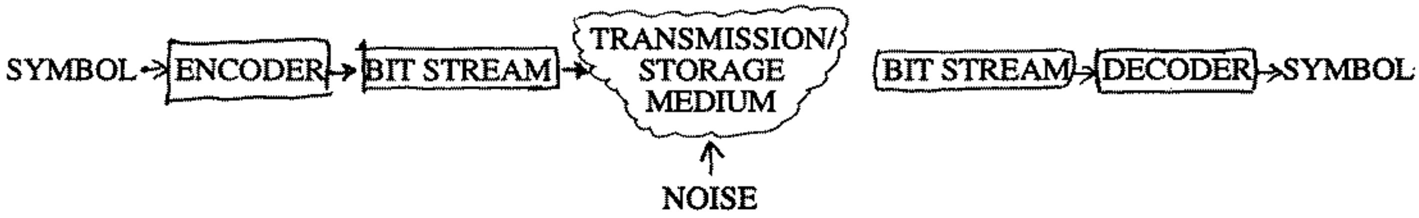
DECIMAL DIGIT	8421 CODE	XS3 CODE
0	0000	0011
1	0001	0100
2	0010	0101
3	0011	0110
4	0100	0111
5	0101	1000
6	0110	1001
7	0111	1010
8	1000	1011
9	1001	1100

$XS3 = (\text{Decimal} + 3)_2$

B) BINARY CODES FOR CHARACTERS

ASCII or EBCDIC (Covered in csi1101)

ENCODING/DECODING



OF BITS REQUIRED TO REPRESENT EACH SYMBOL DEPENDS ON

*FREQUENCY OF OCCURRENCE OF THE SYMBOL IN AVERAGE INFORMATION

$\therefore$ IF $S_0, S_1, \ldots, S_q$ ARE NOT EQUALLY LIKELY TO OCCUR $\Rightarrow$ VARIABLE LENGTH CODE

IF $S_0, S_1, \ldots, S_q$ ARE EQUALLY LIKELY TO OCCUR $\Rightarrow$ FIXED LENGTH CODE

*PROBABILITY OF SINGLE, DOUBLE, TRIPLE, ... ERRORS OCCURRING IN A PARTICULAR MEDIUM THROUGH WHICH CODE WORDS WILL BE TRANSMITTED/STORED

*WHETHER ERRORS IN A RECEIVED/RETRIEVED SEQUENCE SHOULD BE

DETECTED ONLY $\Rightarrow$ ERROR DETECTING CODE

OR CORRECTED $\Rightarrow$ ERROR CORRECTING CODE

<u>Parity Codes</u>

(Parity ≡ evenness or oddness of # 1s in a sequence.

DECIMAL DIGIT	8421	EVEN PARITY	ODD PARITY
0	0000	00000	00001
1	0001	00011	00010
2	0010	00101	00100
3	0011	00110	00111
4	0100	01001	01000
5	0101	01010	01011
6	0110	01100	01101
7	0111	01111	01110
8	1000	10001	10000
9	1001	10010	10011

Parity Bit

OTHER CODES
HAMMING CODES
HUFFMANN CODES
GRAY CODES
CYCLIC REDUNDANCY CODES

e.g., **TRANSMISSION OF OCTAL DIGITS USING ODD PARITY**

SENDER

- has $(6)_8 = (110)_2$ to send
- performs "parity generation" to find out the value of the parity bit, P i.e.,

Three-bit message x	y	z	Parity bit generated P
0	0	0	1
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	0

$\Rightarrow \quad P = 1$

- sends $(1101)_2$ ← parity bit

RECEIVER

- receives $(1101)_2$
- performs "parity checking" to find out if the #of 1's is odd in $(1101)_2$ i.e.,

Four-bits received x	y	z	P	Parity-error check C
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

- Determines there is no error & drops the parity bit to obtain $(110)_2$

SWITCHING ALGEBRA

Boolean Algebra is a set of elements B with two binary operators + and • that satisfy the following six axioms (Huntington postulates)

P1 B is closed with respect to + and • : i.e., $\forall$ a, b $\in$ B,
- a) $a + b \in B$
- b) $a \cdot b \in B$

P2 There exist identity elements, $0 \in B$ and $1 \in B$, for + and •, respectively : i.e., $\forall a \in B$
- a) $a + 0 = a$
- b) $a \cdot 1 = a$

P3 + and • are both commutative : i.e., $\forall$ a, b $\in$ B
- a) $a + b = b + a$
- b) $a \cdot b = b \cdot a$

P4 + (•) is distributive over • (+) : i.e., $\forall$ a, b, c $\in$ B
- a) $a + (b \cdot c) = (a + b) \cdot (a + c)$
- b) $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$

P5 $\forall a \in B$, there exists an element $a' \in B$ (which is called complement of a) such that
- a) $a + a' = 1$
- b) $a \cdot a' = 0$

P6 There exist <u>at least</u> two elements a, b $\in$ B such that $a \neq b$

Fundamental Identities of Boolean Algebra

- Universal Bound : $\forall a \in B$,
  - a) $a + 1 = 1$
  - b) $a \cdot 0 = 0$
- Absorption : $\forall$ a, b $\in$ B,
  - a) $a + (a \cdot b) = a$
  - b) $a \cdot (a + b) = a$
- Non-Cancellation : $\forall$ a, b, c $\in$ B,
  - a) $a + b = a + c \not\Rightarrow b = c$
  - b) $a \cdot b = a \cdot c \not\Rightarrow b = c$
- Idempotency : $\forall a \in B$,
  - a) $a + a = a$
  - b) $a \cdot a = a$
- Associativity : $\forall$ a, b, c $\in$ B,
  - a) $a + (b + c) = (a + b) + c$
  - b) $a \cdot (b \cdot c) = (a \cdot b) \cdot c$
- De Morgan : $\forall$ a, b $\in$ B,
  - a) $(a \cdot b)' = a' + b'$
  - b) $(a + b)' = a' \cdot b'$
- Involution : $\forall a \in B$, $(a')' = a$

Principle of Duality : Every algebraic identity deducible from postulates remain valid if
+ is replaced by •
• is replaced by +
0 is replaced by 1
1 is replaced by 0

BOOLEAN FUNCTIONS $f(x_1, x_2, ..., x_n)$

Any **boolean function** f which takes a **boolean constant** (i.e., 0 or 1) as its value is defined over n **boolean variables** (i.e., $x_1, x_2, ..., x_n$, each of which takes a boolean constant as its value).

$\Rightarrow f(x_1, x_2, ..., x_n)$

1) **SPECIFICATION OF $f(x_1, x_2, ..., x_n)$**

A) **TRUTH TABLE (TT)**

		x_1	x_2		x_n	f
0	→	0	0		0	
1	→	0	0		1	
$2^n - 1$	→	1	1		1	⇑

2^n COMBINATIONS OF 1'S & 0'S

VALUE OF f

In a TT for a boolean function $f(x_1, x_2, ..., x_n)$,

- each combination of values for $x_1, x_2, ..., x_n$ can be represented as a

MINTERM $L_1 \bullet L_2 \bullet ..... \bullet L_n$

which is a <u>product term</u> containing a literal L_i for each x_i, $1 \leq i \leq n$

or

MAXTERM $L_1 + L_2 + ... + L_n$

which is a <u>sum term</u> containing a literal L_i for each x_i, $1 \leq i \leq n$

where LITERAL L_i for x_i is either x_i or x'_i, $1 \leq i \leq n$

Minterms and Maxterms for Three Variables

			Minterms		Maxterms	
x	y	z	Term	Designation	Term	Designation
0	0	0	$x'y'z'$	m_0	$x + y + z$	M_0
0	0	1	$x'y'z$	m_1	$x + y + z'$	M_1
0	1	0	$x'yz'$	m_2	$x + y' + z$	M_2
0	1	1	$x'yz$	m_3	$x + y' + z'$	M_3
1	0	0	$xy'z'$	m_4	$x' + y + z$	M_4
1	0	1	$xy'z$	m_5	$x' + y + z'$	M_5
1	1	0	xyz'	m_6	$x' + y' + z$	M_6
1	1	1	xyz	m_7	$x' + y' + z'$	M_7

- $m'_i = M_i$ and $M'_i = m_i$ **(From De Morgan's Law)** e.g., $m'_0 = M_0$

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

B) BOOLEAN EXPRESSION (e)

f = e where e is a boolean constant, boolean variable, $e_1 \bullet e_2$, $e_1 + e_2$, e'_1, or e'_2

A boolean expression for $f(x_1, x_2, \ldots, x_n)$ can be formed

- AS A SUM OF MINTERMS for which the value of f is 1

$$\Rightarrow f(x_1, x_2, \ldots, x_n) = \sum_{i=0}^{2^n-1} (\alpha_i \bullet m_i) \qquad \text{where } \alpha_i \ (= 0 \text{ or } 1) \text{ is the value of f}$$

$\Rightarrow$ CSOP (CANONICAL SUM OF PRODUCTS) representation of f

- AS A PRODUCT OF MAXTERMS for which the value of f is 0

$$\Rightarrow f(x_1, x_2, \ldots, x_n) = \prod_{i=0}^{2^n-1} (\alpha_i + M_i) \qquad \text{where } \alpha_i \ (= 0 \text{ or } 1) \text{ is the value of f}$$

$\Rightarrow$ CPOS (CANONICAL PRODUCT OF SUMS) representation of f

e.g.,

CSOP $\quad f(x_1,x_2,x_3) = \sum m(1,4,5,6,7) = m_1 + m_4 + m_5 + m_6 + m_7$

$= (x_1'x_2'x_3)+(x_1x_2'x_3')+(x_1x_2'x_3)+(x_1x_2x_3')+(x_1x_2x_3)$

CPOS $\quad f(x_1,x_2,x_3) = \prod M(0,2,3) = M_0 \bullet M_2 \bullet M_3$

$= (x_1+x_2+x_3) \bullet (x_1+x_2'+x_3) \bullet (x_1+x_2'+x_3')$

2) CONVERSIONS AMONG REPRESENTATIONS OF $f(x_1, x_2, \ldots, x_n)$

A) TRUTH TABLE $\leftrightarrow$ EXPRESSION

	$x_1x_2x_3$	f	f'
m_0	0 0 0	0	1
m_1	0 0 1	1	0
m_2	0 1 0	0	1
m_3	0 1 1	0	1
m_4	1 0 0	1	0
m_5	1 0 1	1	0
m_6	1 1 0	1	0
m_7	1 1 1	1	0

CSOP $\quad f(x_1,x_2,x_3) = \sum m(1,4,5,6,7)$

CSOP $\quad f'(x_1,x_2,x_3) = \sum m(0,2,3)$

CPOS $\quad f(x_1,x_2,x_3) = \prod M(0,2,3)$

CPOS $\quad f'(x_1,x_2,x_3) = \prod M(1,4,5,6,7)$

B) CPOS $\leftrightarrow$ CSOP

a) INTERCHANGE SYMBOLS Σ and Π

b) LIST THOSE NUMBERS MISSING IN THE ORIGINAL

e.g., CPOS $f(x_1,x_2,x_3) = \prod M(0,2,3) \leftrightarrow$ CSOP $f(x_1,x_2,x_3) = \sum m(1,4,5,6,7)$

3) **EVALUATION OF $f(x_1, x_2, ..., x_n)$**
is based on
- a given combination of values for $x_1, x_2, ..., x_n$
- a known precedence among operators ( ) , ′ , • , +

A) **BOOLEAN EXPRESSION**
Given CSOP for $f(x_1,x_2,x_3)$ and values for x_1, x_2, x_3

e.g.,

$$f(x_1,x_2,x_3) = \sum m(1,4,5,6,7) = m_1 + m_4 + m_5 + m_6 + m_7$$
$$= (x_1'x_2'x_3)+(x_1x_2'x_3')+(x_1x_2'x_3)+(x_1x_2x_3')+(x_1x_2x_3)$$

Given CPOS for $f(x_1,x_2,x_3)$ and values for x_1, x_2, x_3

e.g.,

$$f(x_1,x_2,x_3) = \prod M(0,2,3) = M_0 \cdot M_2 \cdot M_3$$
$$= (x_1+x_2+x_3) \cdot (x_1+x_2'+x_3) \cdot (x_1+x_2'+x_3')$$

B) **TRUTH TABLE**
Given TT for $f(x_1,x_2,x_3)$ and values for x_1, x_2, x_3

$x_1x_2x_3$	f
0 0 0	0
0 0 1	1
0 1 0	0
0 1 1	0
1 0 0	1
1 0 1	1
1 1 0	1
1 1 1	1

4) SIMPLIFICATION OF $f(x_1, x_2, ..., x_n)$ (FROM CANONICAL TO STANDARD FORMS OF EXPRESSIONS)

STANDARD FORMS OF EXPRESSIONS

SOP (SUM OF PRODUCTS):
NOT ALL PRODUCT TERMS ARE MINTERMS
e.g., $f(x,y,z) = x'y'z + xy' + yz'$

POS (PRODUCT OF SUMS):
NOT ALL SUM TERMS ARE MAXTERMS
e.g., $f(x,y,z) = (x + y) \bullet (x' + y + z) \bullet (x + y')$

A) ALGEBRAIC SIMPLIFICATION

I) CSOP → SOP

e.g.,

$$\begin{aligned} f(x,y,z) &= \Sigma m(0, 2, 3, 4, 6, 7) = m_3 + m_2 + m_6 + m_4 + m_0 + m_7 \\ &= x'yz + x'yz' + xyz' + xy'z' + x'y'z' + xyz \\ &= x'yz + x'yz' + xyz' + xy'z' + x'yz' + x'y'z' + xyz \\ &= (x'y) \bullet (z + z') + (x + x') \bullet (y + y') \bullet z' + xyz \\ &= x'y + z' + xyz \end{aligned}$$

II) CPOS → POS

e.g.,

$$\begin{aligned} f(x,y,z) &= \Pi M(2, 4, 5, 6, 7) = M_4, M_5, M_6, M_7, M_2 \\ &= (x'+y+z) \bullet (x'+y+z') \bullet (x'+y'+z) \bullet (x'+y'+z') \bullet (x+y'+z) \\ &= [(x'+y+z)\bullet(x'+y+z')\bullet(x'+y'+z)\bullet(x'+y'+z')]\bullet[(x+y'+z)\bullet(x'+y'+z)] \\ &= (x' + yy' + zz') \bullet (y' + z + xx') \\ &= x' \cdot (y' + z) \end{aligned}$$

B) GRAPHICAL SIMPLIFICATION (KARNAUGH MAPS)

* ANY 2^k ADJACENT SQUARES for k=0,1,2,...n IN AN n -VARIABLE MAP PRESENTS AN AREA THAT GIVES A TERM OF n-k VARIABLES
IF n = k $\Rightarrow$ ENTIRE AREA OF THE MAP $\Rightarrow$ IDENTITY FUNCTION

* IN AN n-VARIABLE MAP, EACH SQUARE HAS n ADJACENT SQUARES

m_0	m_1
m_2	m_3

$x \backslash y$	0	1 (y)
0	$x'y'$	$x'y$
1 (x)	xy'	xy

m_0	m_1	m_3	m_2
m_4	m_5	m_7	m_6

$x \backslash yz$	00	01 (z)	11 (y, z)	10 (y)
0	$x'y'z'$	$x'y'z$	$x'yz$	$x'yz'$
1 (x)	$xy'z'$	$xy'z$	xyz	xyz'

m_0	m_1	m_3	m_2
m_4	m_5	m_7	m_6
m_{12}	m_{13}	m_{15}	m_{14}
m_8	m_9	m_{11}	m_{10}

$wx \backslash yz$	00	01 (z)	11 (y, z)	10 (y)
00	$w'x'y'z'$	$w'x'y'z$	$w'x'yz$	$w'x'yz'$
01 (x)	$w'xy'z'$	$w'xy'z$	$w'xyz$	$w'xyz'$
11 (w, x)	$wxy'z'$	$wxy'z$	$wxyz$	$wxyz'$
10 (w)	$wx'y'z'$	$wx'y'z$	$wx'yz$	$wx'yz'$

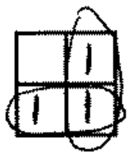
GIVEN f = e in CSOP, OBTAIN MINIMAL SOP

Procedure-CSOP-to-SOP

1 - MARK f ON THE MAP
(i.e., PUT 1 INTO SQUARES THAT CORRESPOND TO MINTERMS FOR WHICH f TAKES VALUE 1)

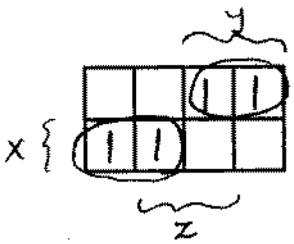
2 - OBTAIN A MINIMAL e FOR f IN SOP BY COVERING ALL 1-SQUARES WITH MINIMAL NUMBER OF LARGEST POSSIBLE AREAS OF SIZE 2^k, $k = 0, 1, 2,..$ FORMED BY COMBINING ADJACENT 1-SQUARES

e.g., $f(x,y) = \sum m(1, 2, 3)$



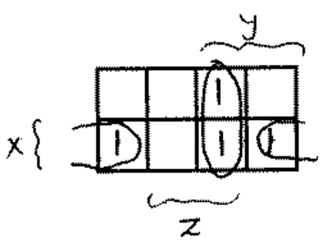
$f(x,y) = x + y$

e.g., $f(x,y,z) = \sum m(2, 3, 4, 5)$



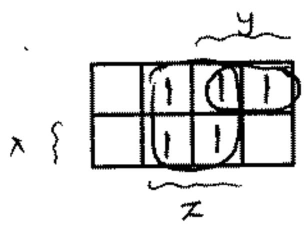
$f(x,y,z) = xy' + x'y$

e.g., $f(x,y,z) = \sum m(3, 4, 6, 7)$



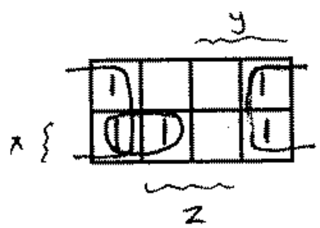
$f(x,y,z) = xz' + yz$

e.g., $f(x,y,z) = \sum m(1, 2, 3, 5, 7)$



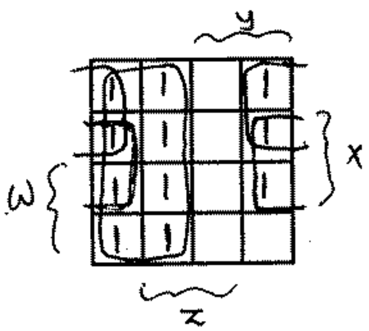
$f(x,y,z) = z + x'y$

e.g., $f(x,y,z) = \sum m(0, 2, 4, 5, 6)$



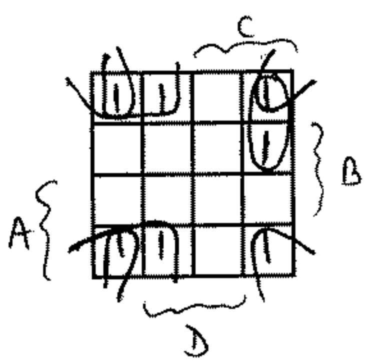
$f(x,y,z) = z' + xy'$

e.g., $f(w,x,y,z) = \sum m(0, 1, 2, 4, 5, 6, 8, 9, 12, 13, 14)$



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

e.g., $f(A,B,C,D) = \sum m(0, 1, 2, 6, 8, 9, 10)$



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