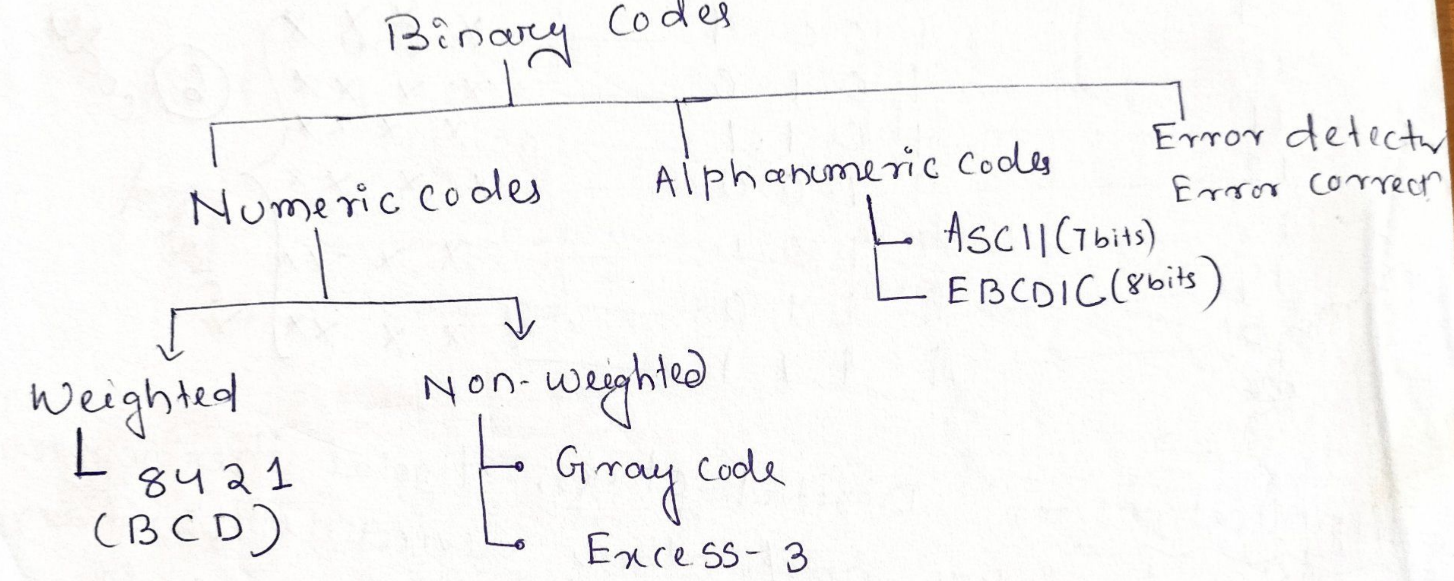
Binary Code.

code: group of symbols

code

Binary codes: Computer or other digital ckts process data into binary format. Data may be numeric, alphabet or sign. so various binary codes which are in 0's & 1's format are used to represent data



BCD It expresses each decimal digit (0-9) with binary code.

8421 is a type of BCD code.

Weights.

	BCD 8 4 2 1 → weights of each place b_3 b_2 b_1 b_0
0 —	0000
1 —	0001
2 —	0010
3 —	0011
4 —	0100
5 —	0101
6 —	0110
7 —	0111
8 —	1000
9 —	1010

0101 → $8 \times 0 + 4 \times 1 + 0 \times 2 + 1 \times 1 = 5$

Decimal	Binary equivalent	Binary coded de
0	0000	0000
1	0001	0001
2	0010	0010
3	0011	0011
4	0100	0100
5	0101	0101
6	0110	0110
7	0111	0111
8	1000	1000
9	1001	X X X X
10	1010	X X X X
11	1011	X X X X
12	1100	X X X X
13	1101	X X X X
14	1110	X X X X
15	1111	

⑥ code Invalid

Applications:- Digital clocks, digital thermometer digital meters, and other devices with seven segment desplay typically use BCD code

Binary Addition:

Step 1 Add the two BCD no using the rules binary addition.

Step 2 If 4-bit sum is equal to or less th 9 then it is a valid BCD number

Step 3 If a 4 bit sum is greater than 9 or if a carry is generated it is invalid resul Add 6 (0110) to the 4 bit sum. If a carry results when a '6' is added then add the

(2)

Gray code

Gray Code is unweighted and is no an arithmatic code. There are no specific code assigned to the bit position.

It exibits a single bit change from one code wor to the next in sequence.

Decimal	Binary	Gray code
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100

Binary to gray code

The most significant bit in gray code is Same as the binary number.

Going from left to right add each adjacen pair of binary code bit to get the next gray c bit. Discard carries.

e.g Binary

MSB
1 0 1 1 0

1 1 1 0 1 gray

Gray to Binary:

(1) Copy the MSB bit.

(2) Add each binary code bit generated to the gray code bit in the adjacent position. Discard carr

Convert each decimal no to BCD. 4|

(a) 35 (b) 98 (c) 170 (d) 2469

0011 0101 ● 10011000 0001 0111 000 0010 0100 0110 0100

Convert BCD to decimal

(a) $\underbrace{1000}_{8}\underbrace{0110}_{6}$ (b) $\underbrace{0011}_{3}\underbrace{0101}_{5}\underbrace{0001}_{1}$ (c) 1001 0100 0111 0000

9 4 7 0

Add the following BCD nos:—

(a)
$\overset{3}{0011} + \overset{4}{0100}$
0111 (7)

(b)
0010 0011
0001 0101
0011 1000

(c) 86 + 13 = 99
10010110
00010011
1001 1001
(9) (9)

(d)
0100 0101 0000
0100 0001 0111
1 000 0110 0 111

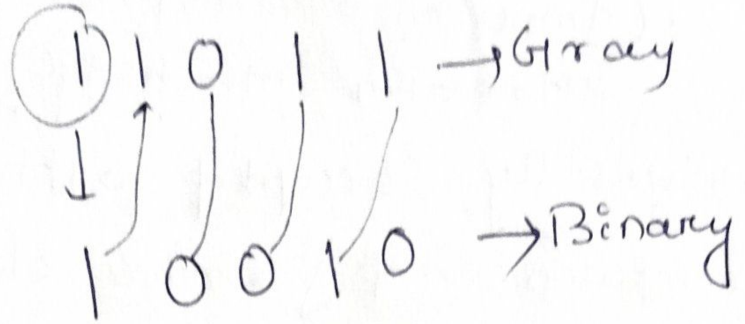
(e) 1001 + 0100
1001
0100
1101 (invalid)
1101
0110
10011
0001 0011

(f) 1001 + 1001
= 10010
0110
0001 1000

(g) 00010110 + 00010101
0001 0110
0001 0101
0 0 0 1 1011 ↳ invalid
+110
0011,0001

(h) 01100111
+ 01010011
0110 0111
01010011
1011 1010 ↳ invalig
01100110
0001 0010 0000

e.g



eg

Convert binary to gray – 11000110

Convert gray to binary – 10101111

Problems

(1) Convert the following decimal to BCD
(a) 104 (b) 359 (c) 1051 (d) 186 (e) 210
Convert each to binary and compare the no of bits required.

(2) Convert each ~~decimal~~ BCD to decimal
(a) 10000000 (b) 1000110111 (c) 11001100 1100111

(3) Add the following BCD numbers:
(a) 1000 + 0110 (b) 0111 + 0101 (c) 00100101 + 00100111
(d) 10011000 + 10010111 (e) 010101100001 + 011100001000

(4) Add the no. first converting each to BCD.
(a) 28 + 23 (b) 65 + 58 (c) 113 + 101 (d) 295 + 157

(5) What is the binary weight of ~~ea~~ each 1 in the following BCD
(a) 0010 (b) 1000 (c) 0001 (d) 0100.

ASCII Code : (American standard ~~for~~ code for information Interchange)

It is a universally accepted alphanumeric code used in most computers and other electronic equipment.

Most computer keyboards are standardized with the ASCII. When you enter a letter, number or a Control command the corresponding ASCII code goes into the computer.

ASCII has 128 charecters and symbols represented by a 7 bit binary code. 94 are graphic charectercistic, that can be printed. 34 are non-printing charecters, used for various ~~function~~ control function.

94 charecters include — 26 (A to Z)
26 (a to z)
10 (0 to 9)
32 (special char e.g *, $, &)

Similarly e.g of control charecters are "null", "line feed", "start of text", "escape" etc.

e.g

A = 1 0 0 0 0 0 1

a = 1 1 0 0 0 0 1

ESC = 0 0 1 1 0 1 1

In addition to 128 standard ASCII charecteres there are an additional 128 charecters that were adopted by IBM for use of the following things. — Greek letters, Math symbols, Currency symbols etc. It is termed as Extended ASCII code. It uses 8 bit representation $\boxed{2^8 = 256}$