Binary to Gray Conversion

- Gray Code MSB is binary code MSB.
- Gray Code MSB-1 is the XOR of binary code MSB and MSB-1.
- MSB-2 bit of gray code is XOR of MSB-1 and MSB-2 bit of binary code.
- MSB-N bit of gray code is XOR of MSB-N-1 and MSB-N bit of binary code.

8421 BCD code (Natural BCD code):

Each decimal digit 0 through 9 is coded by a 4 bit binary no. called natural binary codes. Because of the 8,4,2,1 weights attached to it. It is a weighted code & also sequential . it is useful for mathematical operations. The advantage of this code is its case of conversion to & from decimal. It is less efficient than the pure binary, it require more bits.

Ex: 14→1110 in binary

But as 0001 0100 in 8421 ode.

The disadvantage of the BCD code is that , arithmetic operations are more complex than they are in pure binary . There are 6 illegal combinations 1010,1011,1100,1101,1110,1111 in these codes, they are not part of the 8421 BCD code system . The disadvantage of 8421 code is, the rules of binary addition 8421 no, but only to the individual 4 bit groups.

BCD Addition:

It is individually adding the corresponding digits of the decimal no,s expressed in 4 bit binary groups starting from the LSD . If there is no carry & the sum term is not an illegal code , no correction is needed .If there is a carry out of one group to the next group or if the sum term is an illegal code then 6_{10}(0100) is added to the sum term of that group & the resulting carry is added to the next group.

Ex: Perform decimal additions in 8421 code

(a)25+13

In BCD	25=	0010 0101
In BCD	+13 =	+0001 0011
	38	0011 1000

No carry , no illegal code .This is the corrected sum (b). 679.6 + 536.8

679.6 =	0110	0111	1001	.0110 in BCD		
+536.8 =	+0101	0011	0010	.1000 in BCD		
1216.4	1011	1010	0110	. 1110	illegal codes	
	+0110	+ 0011	+0110	. + 0110	add 0110 to each	
	(1)0001	(1)0000	(1)0101	. (1)0100	propagate carry	
	/	/	/	/		
	+1	+1	+1	+1		
	0001	0010	0001	0110	.	0100
	1	2	1	6	.	4

BCD Subtraction:

Performed by subtracting the digits of each 4 bit group of the subtrahend the digits from the corresponding 4- bit group of the minuend in binary starting from the LSD . if there is no borrow from the next group , then 6_{10}(0110)is subtracted from the difference term of this group.

(a)38-15

In BCD	38=	0011	1000
In BCD	-15 =	-0001	0101
	23	0010	0011

No borrow, so correct difference.

.(b) 206.7-147.8

206.7 =	0010	0000	0110	.	0111	in BCD
-147.8 =	-0001	0100	0111	.	0110	in BCD
58.9	0000	1011	1110	.	1111	borrows are present
-0110	-0110 .		-0110		subtract 0110	
		0101	1000	.	1001	

BCD Subtraction using 9's & 10's compliment methods:

Form the 9's & 10's compliment of the decimal subtrahend & encode that no. in the 8421 code . the resulting BCD no.s are then added.

EX: 305.5 – 168.8

```
305.5  =     305.5
-168.8=      +83.1                9's comp of -168.8
             ___  _ _
             (1)136.6
                 +1               end around carry
               136.7              corrected difference
305.5₁₀  =   0011   0000 0101  .   0101
+831.1₁₀ =  +1000   0011 0001  .   0001        9's comp of 168.8 in BCD
__ _ _      ________________
            +1011   0011  0110 .   0110     1011 is illegal code
            +0110                               add 0110
            ______________________________
         (1)0001    0011   0110  .  0110
                                         +1    End around carry
            ______________________________
            0001    0011   0110  .  0111
                                = 136.7
```

Excess three(xs-3)code:

It is a non-weighted BCD code .Each binary codeword is the corresponding 8421 codeword plus 0011(3).It is a sequential code & therefore , can be used for arithmetic operations..It is a self-complementing code.s o the subtraction by the method of compliment addition is more direct in xs-3 code than that in 8421 code. The xs-3 code has six invalid states 0000,0010,1101,1110,1111.. It has interesting properties when used in addition & subtraction.

Excess-3 Addition:

Add the xs-3 no.s by adding the 4 bit groups in each column starting from the LSD. If there is no carry starting from the addition of any of the 4-bit groups , subtract 0011 from the sum term of those groups (because when 2 decimal digits are added in xs-3 & there is no carry , result in xs-6). If there is a carry out, add 0011 to the sum term of those groups(because when there is a carry, the invalid states are skipped and the result is normal binary).

EX: 37 0110 1010
+28 +0101 1011
— — – – – – – – – –

65 1011 (1)0101 carry generated
+1 propagate carry
— – – – – – –

1100 0101 add 0011 to correct 0101 &
-0011 +0011 subtract 0011 to correct 1100
— – – – – – – – –

1001 1000 $=65_{10}$

Excess -3 (XS-3) Subtraction:

Subtract the xs-3 no.s by subtracting each 4 bit group of the subtrahend from the corresponding 4 bit group of the minuend starting form the LSD .if there is no borrow from the next 4-bit group add 0011 to the difference term of such groups (because when decimal digits are subtracted in xs-3 & there is no borrow , result is normal binary). I f there is a borrow , subtract 0011 from the differenceterm(b coz taking a borrow is equivalent to adding six invalid states , result is in xs-6)

Ex: 267-175

267 = 0101 1001 1010
-175= -0100 1010 1000
— – – – – – – – –

0000 1111 0010
+0011 -0011 +0011
____

0011 1100 +0011 $=92_{10}$

Xs-3 subtraction using 9's & 10's compliment methods:
Subtraction is performed by the 9's compliment or 10's compliment
Ex:687-348 The subtrahend (348) xs -3 code & its compliment are:

9's comp of 348 = 651
Xs-3 code of 348 = 0110 0111 1011
1's comp of 348 in xs-3 = 1001 1000 0100
Xs=3 code of 348 in xs=3 = 1001 1000 0100

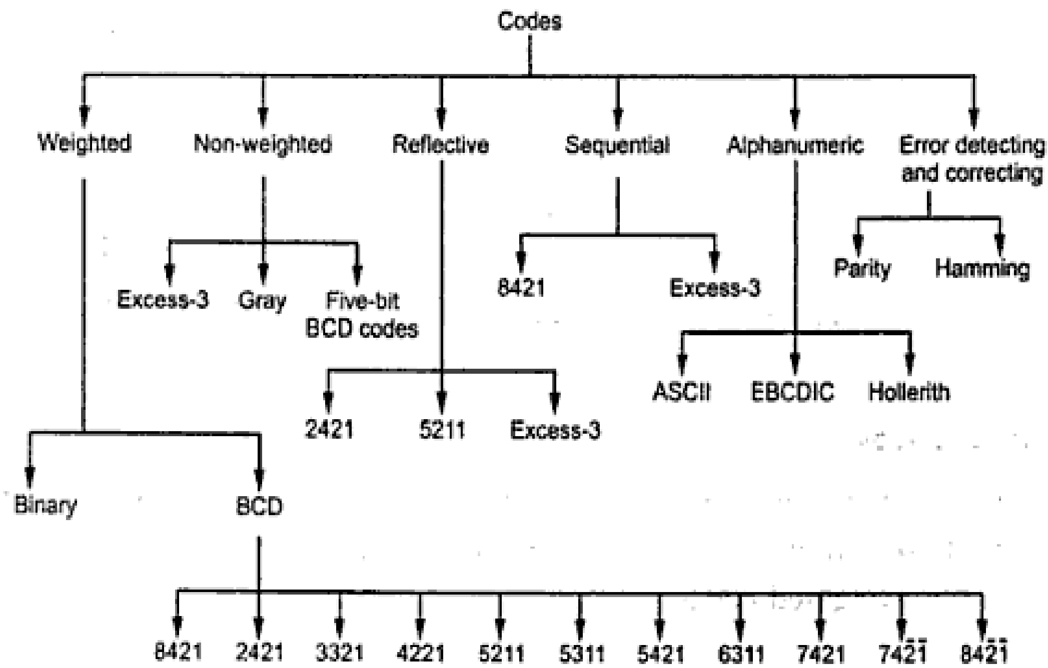
```
   687              687
  -348     →       +651 9's compl of 348
 ________
   339             (1)338
                      +1 end around carry
                   ___ _
                      _
                    339          corrected difference in decimal

   1001      1011      1010      687 in xs-3
   +1001     1000      0100      1's comp 348 in xs-3
   ___ _   _ _ _ _ _ _ _   _ _
   _ (1)0010 (1)0011   1110   carry generated
//
+1         +1                    propagate carry
___ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _-
(1)0011    0010   1110
                        +1       end around carry
___ _ _ _ _ _ _ _ _ _ _ _ _ _ _

   0011      0011      1111      (correct 1111 by sub0011 and
   +0011     +0011     +0011     correct both groups of 0011 by
   ___ _ _ _   _ _ _ _   _  _  _   adding 0011)
               _ _
                        _
   0110      0110      1100      corrected diff in xs-3 = 330₁₀
```

The Gray code (reflective –code):

Gray code is a non-weighted code & is not suitable for arithmetic operations. It is not a BCD code . It is a cyclic code because successive code words in this code differ in one bit position only i.e, it is a unit distance code.Popular of the unit distance code.It is also a reflective code i.e,both reflective & unit distance. The n least significant bits for 2^n through 2^{n+1}-1 are the mirror images of thosr for 0 through 2^n-1.An N bit gray code can be obtained by reflecting an N-1 bit code about an axis at the end of the code, & putting the MSB of 0 above the axis & the MSB of 1 below the axis.

Reflection of gray codes:

Gray Code				Decimal	4 bit binary
1 bit	2 bit	3 bit	4 bit		
0	00	000	0000	0	0000
1	01	001	0001	1	0001
	11	011	0011	2	0010
	10	010	0010	3	0011
		110	0110	4	0100
		111	0111	5	0101
		101	0101	6	0110
		110	0100	7	0111
			1100	8	1000
			1101	9	1001
			1111	10	1010
			1110	11	1011
			1010	12	1100
			1011	13	1101
			1001	14	1110
			1000	15	1111



Binary codes block diagram

Error – Detecting codes: When binary data is transmitted & processed,it is susceptible to noise that can alter or distort its contents. The 1‘s may get changed to 0‘s & 1‘s .because digital systems must be accurate to the digit, error can pose a problem. Several schemes have been devised to detect the occurrence of a single bit error in a binary word, so that whenever such an error occurs the concerned binary word can be corrected & retransmitted.

Parity: The simplest techniques for detecting errors is that of adding an extra bit known as parity bit to each word being transmitted.Two types of parity**:** Oddparity**,** evenparity forodd parity, the parity bit is set to a _0‘ or a _1‘ at the transmitter such that the total no. of 1 bit in the word including the parity bit is an odd no.For even parity, the parity bit is set to a _0‘ or a _1‘ at the transmitter such that the parity bit is an even no.

Decimal	8421 code	Odd parity	Even parity
0	0000	1	0
1	0001	0	1
2	0010	0	1
3	0011	1	0
4	0100	0	1
5	0100	1	0
6	0110	1	0
7	0111	0	1
8	1000	0	1
9	1001	1	0

When the digit data is received . a parity checking circuit generates an error signal if the total no of 1‘s is even in an odd parity system or odd in an even parity system. This parity check can always detect a single bit error but cannot detect 2 or more errors with in the same word.Odd parity is used more often than even parity does not detect the situation. Where all 0‘s are created by a short ckt or some other fault condition.

Ex: Even parity scheme

(a) 10101010 (b) 11110110 (c)10111001

Ans:

(a) No. of 1‘s in the word is even is 4 so there is no error
(b) No. of 1‘s in the word is even is 6 so there is no error
(c) No. of 1‘s in the word is odd is 5 so there is error

Ex: odd parity

(a)10110111 (b) 10011010 (c)11101010

Ans:

(a) No. of 1‘s in the word is even is 6 so word has error
(b) No. of 1‘s in the word is even is 4 so word has error
(c) No. of 1‘s in the word is odd is 5 so there is no error

Checksums:

Simple parity can‘t detect two errors within the same word. To overcome this, use a sort of 2 dimensional parity. As each word is transmitted, it is added to the sum of the previously transmitted words, and the sum retained at the transmitter end. At the end of transmission, the sum called the check sum. Up to that time sent to the receiver. The receiver can check its sum with the transmitted sum. If the two sums are the same, then no errors were detected at the receiver end. If there is an error, the receiving location can ask for retransmission of the entire data, used in teleprocessing systems.

Block parity:

Block of data shown is create the row & column parity bits for the data using odd parity. The parity bit 0 or 1 is added column wise & row wise such that the total no. of 1‘s in each column & row including the data bits & parity bit is odd as