BCD ADDER

In BCD code each decimal digit, 0 through 9, is coded by a 4 bit binary number. It implies there is no BCD representation for the decimal numbers greater than 9.

To add two BCD numbers add the 4 bit BCD code groups for each decimal digit position using ordinary binary addition. For these sums the sum is less than or equal to decimal 9, i.e., the sum is in proper BCD form and no correction is needed. When the sum of two digits is greater than 9, a correction is required i.e. 0110 to be added to that sum, to produce the proper BCD result. This will produce a carry to be added to the next decimal position.

A BCD adder circuit able to operate in accordance with the above points is shown in fig 2.34.

The BCD addition as like simple binary addition is performed by 4 bit parallel binary adder

(1) The sum outputs S0, S1, S2, S3, S4 can range from 0000 to 1001. (when both the BCD numbers (1001) are equal to decimal 9.

If the sum is between 0000 to 01001 no correction Logic is required

S_4	S_3	S_2	S_1	S_0	Decimal Number
0	1	0	1	0	10
0	1	0	1	1	11
0	1	1	0	0	12
0	1	1	0	1	13
0	1	1	1	0	14
0	1	1	1	1	15
1	0	0	0	0	16
1	0	0	0	1	17
1	0	0	1	0	18

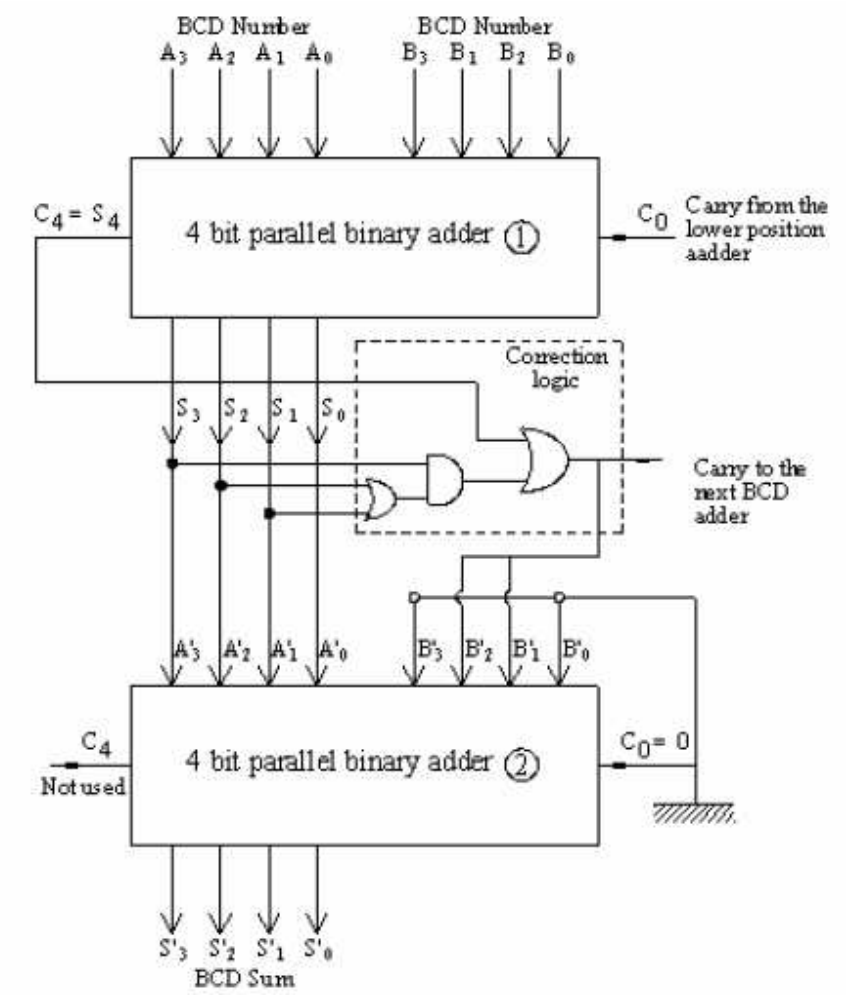
The correction logic required is determined by logic output ‘y’, that will go High only when the sum is greater than 01001. From the table above we can observe that “y” will be high for either of the following conditions.

(1) Wherever S3 = 1 and either S1 or S2 or both are 1 (Sums 10 to 15).

(2) Whenever S4 = 1.

This condition can be expressed as y = S3 (S2 + S1) + S4. When y = 1, it is necessary to add 0110 to the sum bits, and to generate a carry. Fig - 2.34 above shows the circuit of BCD adder with the correction logic y.

The circuit consists of three basic parts. The two BCD numbers A3 A2 A1 A0 and B2, B2 B1 B0 are added together in the adder (1), to produce the sum S4 S3 S2 S1 S0. The logic gates shown implement the expression for y. The adder (2) will add the correction 0110 to the sum bits, only when y = 1, producing the final BCD sum output represented by . The y is also the carry out that is produced when the sum is greater than 01001 of course, when y = 0 there is no carry and no addition of 0110. In such asses , = S3 S2 S1 S0.



Comparators

In the digital computer, one has to carry out not only the arithmetic operations but also the logical operations. One of the logical functions is to know whether a given number is greater than, equal to or lesser than another number. A comparator is a circuit that compares the values of two numbers.

To compare the two single bit numbers (A and B) a single bit comparator is used. The truth table for the comparator action is given below.

A	B	X A>B	Y A = B	Z A < B
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0

Fig: Single bit comparator

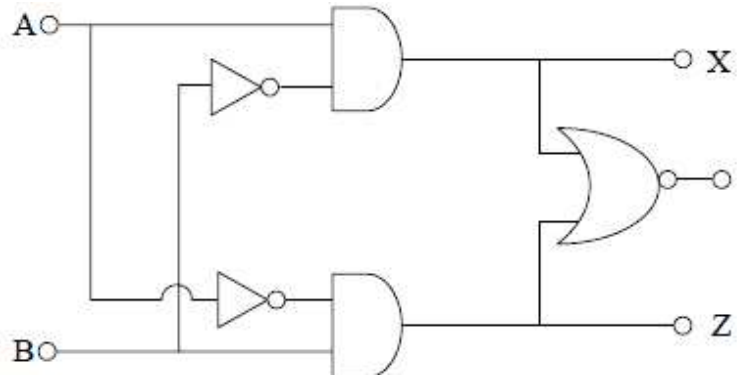


Fig: Logic diagram of single bit comparator

For the above fig X and Z are same but Y is obtained by nor operation between X and Z.

2. TWO BIT COMPARATOR

Let the numbers to be compared be A (A_1 A_0) and B(B_1 B_0).

A will be greater than B when any of the following conditions is satisfied.

(1) When $A_1 = 1$, $B_1 = 0$ irrespective of the other bit;

(2) If $A_1 = B_1$ and $A_0 = 1$; $B_0 = 0$.

A will be equal to B when the following condition is satisfied.

(1) $A_1 = B_1$; $A_0 = B_0$.

A will be lesser than B when any one of the following conditions satisfied.

(1) When $A_1 = 0$, $B_1 = 1$ irrespective of the other bits.

(2) If $A_1 = B_1$ and $A_0 = 0$; $B_0 = 1$.

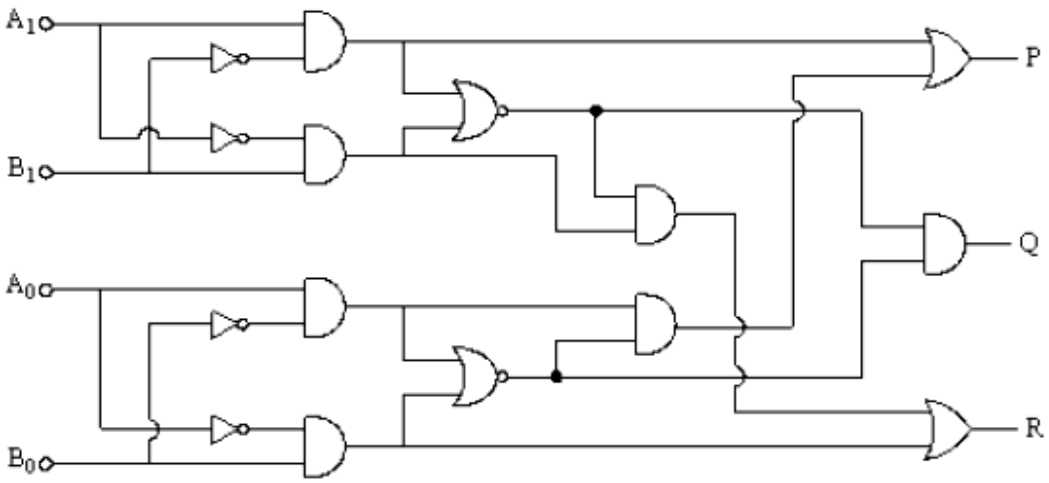


Fig: Logic diagram of two bit comparator