Magnitude Comparator

- A *magnitude comparator* is a combinational circuit that compares two numbers.
- The comparison of two numbers is an operation that determines if one number is greater than, less than, or equal to the other number.
- Let A and B are two number, and determines their relative magnitudes. The outcome of the comparison is specified by three binary variables that indicate whether A > B, A = B. or A < B.

Design of 1-bit magnitude comparator:

n=1; $2^{2n} = 2^{2x1} = 4$ combinations or entries.

Truth table of 1-bit comparator

Input		Output		
A	B	A<B	A=B	A>B
0	0	0	1	0
1	0	0	0	1
0	1	1	0	0
1	1	0	1	0

Logic equation for (A < B) , C = A'B

Logic equation for (A = B), D = A'B' + AB

Logic equation for (A > B), E = AB'

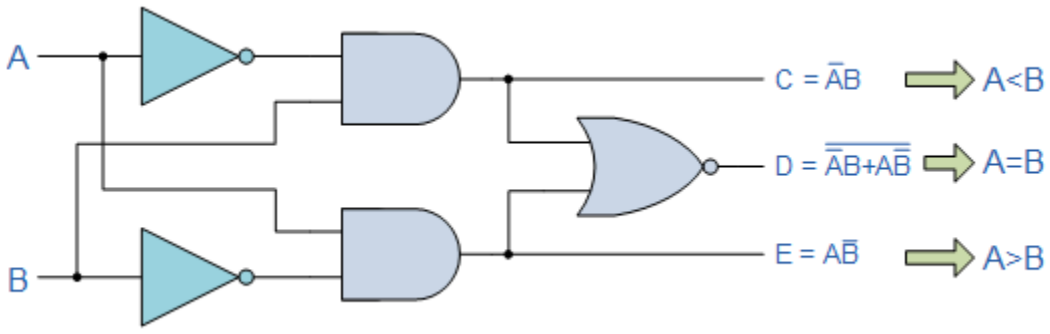
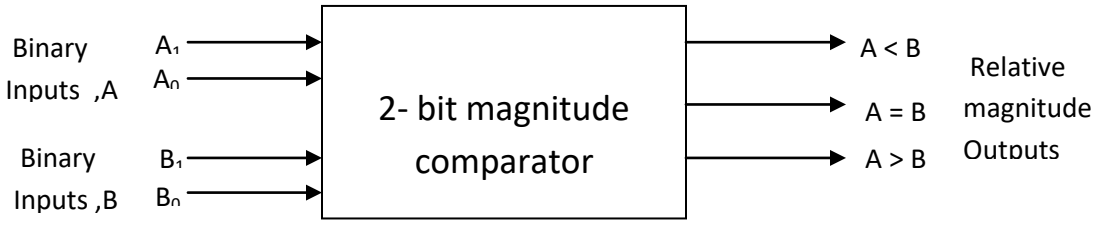


Figure: Logic diagram of 1-bit comparator

Design of 2- bit magnitude comparator:

We need 2^{2n} possible combinations in truth table of 2-bit magnitude comparator.
n=2; $2^{2n} = 2^{2x2} = 16$ entries

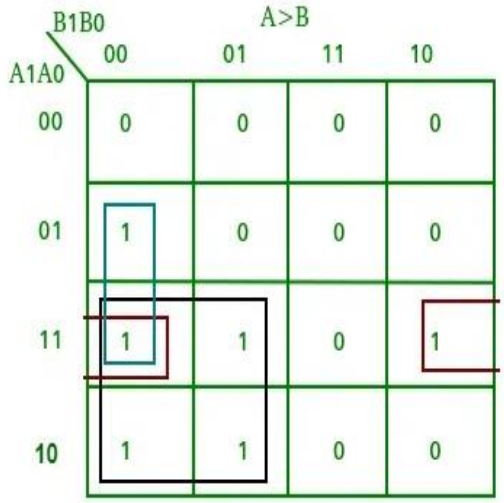


Comparator Inputs				Comparator Outputs		
A		B		A<B	A=B	A>B
A_1	A_0	B_1	B_0			
0	0	0	0	0	1	0
0	0	0	1	1	0	0
0	0	1	0	1	0	0
0	0	1	1	1	0	0
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	1	0	0
0	1	1	1	1	0	0
1	0	0	0	0	0	1
1	0	0	1	0	0	1
1	0	1	0	0	1	0
1	0	1	1	1	0	0

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

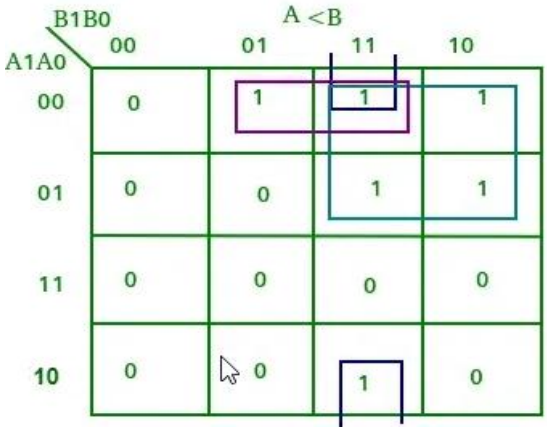
Truth Table of 2-bit magnitude comparator

K-map



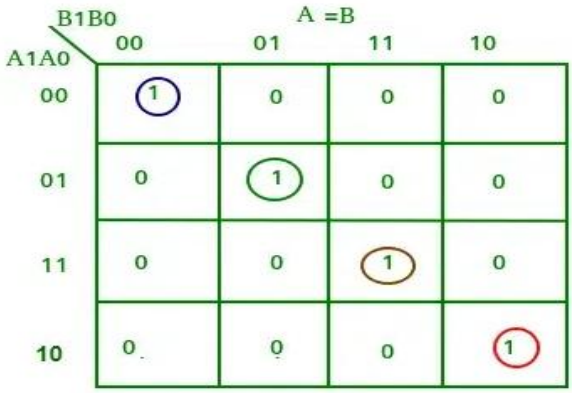
$A > B : A_1B_1' + A_0B_1'B_0' + A_1A_0B_0'$

K-map



After solving the K-map

$A < B : A_1'B_1 + A_0'B_1B_0 + A_1'A0'B0$



For (A = B)

$A_1'A_0'B_1'B_0' + A_1'A_0B_1'B_0 + A_1A_0B_1B_0 + A_1A_0'B_1B_0'$

$= A_1'B_1' (A_0'B_0' + A_0B_0) + A_1B_1 (A_0B_0 + A_0'B_0')$

$= A_0B_0 + A_0'B_0') (A_1B_1 + A_1'B_1')\backslash$

$= (A_0$ Ex-NOR $B_0) (A_1$ Ex-NOR $B_1)$

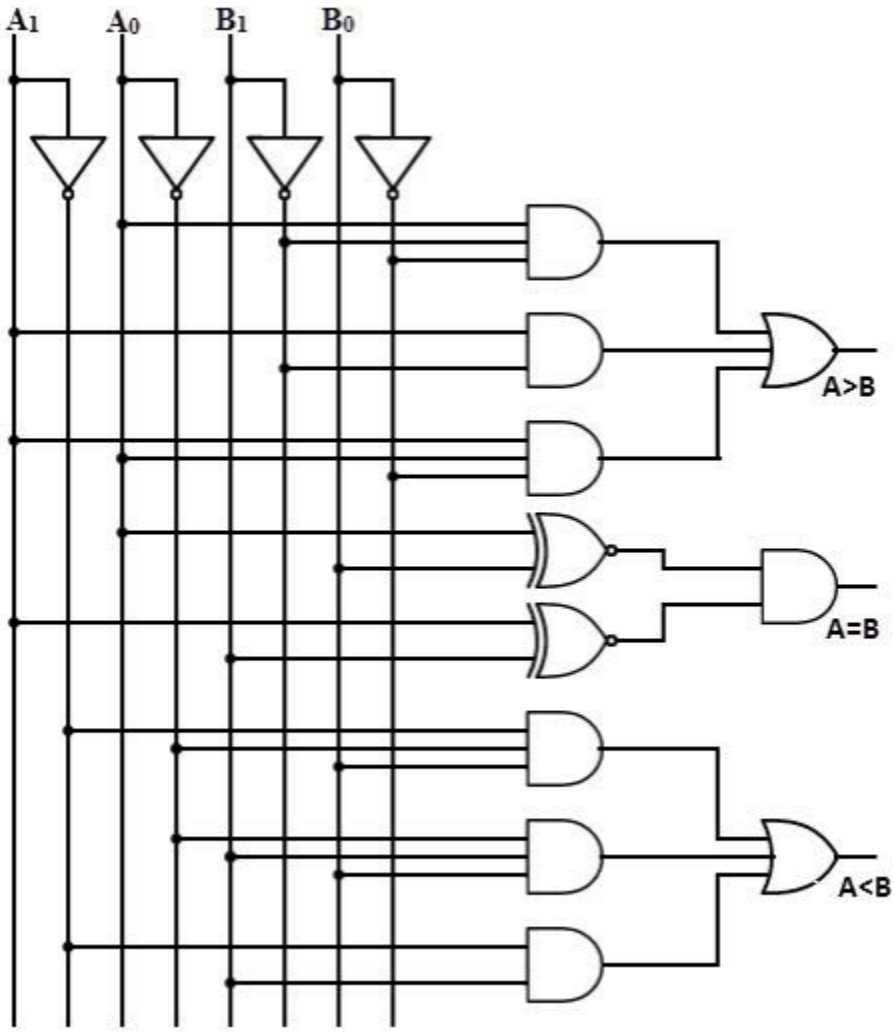


Figure: Logic diagram of 2-bit magnitude comparator

Second approach to design n-bit comparator

For 3-bit or more than 3–bit we will not follow the truth table approach because for 4-bit magnitude comparator, the required no of possible combinations are

$2^{2n} = 2^{2x4} = 256$ entries. It's is not easy to fill these possible combinations in the truth table.

The other approach is to use some design equations for n-bit comparator

For n-bit magnitude comparator

$(A=B) = x_{n-1}. x_{n-2}. x_{n-3}\ldots\ldots. x_0$

$(A>B) = G_{n-1} + x_{n-1} G_{n-2} + x_{n-1} x_{n-2} G_{n-3} + \ldots\ldots + x_{n-1}. x_{n-2}\ldots.. x_1 G_0$

$$(A<B) = L_{n-1} + x_{n-1} L_{n-2} + x_{n-1} x_{n-2} L_{n-3} + \ldots\ldots + x_{n-1}. x_{n-2} \ldots\ldots x_1 L_0$$

Here G and L are used to represent greater than(A>B) and less than(A<B) respectively.

Design of 2-bit magnitude comparator using second approach

Let the two number are A and B

In 2- bit comparator, A and B can be represented in their binary bits as given below

$$A = A_1 A_0$$
$$B = B_1 B_0$$

In a 2-bit comparator, the condition of A=B can be achieved in the following two cases.

1. If $A_1 = B_1$
2. If $A_0 = B_0$

This equality conditions can be expressed logically by Ex-NOR function

$$x_i = \overline{A_i}\,\overline{B_i} + A_i B_i \text{ (for n-bit comparator)}$$

Where, i = 0 to (n-1);

n is the binary inputs bits of the comparator .

For 2 –bit comparator the, i= 0, 1

$$x_1 = \overline{A_1}\,\overline{B_1} + A_1 B_1$$
$$x_0 = \overline{A_0}\,\overline{B_0} + A_0 B_0$$

The logic expression for (A=B) can be represented by

$$E = x_1 . x_0$$
$$E = (\overline{A_1}\,\overline{B_1} + A_1 B_1).(\overline{A_0}\,\overline{B_0} + A_0 B_0)$$
$$E = (A_1 \; XNOR \; B_1).(A_0 \; XNOR \; B_0)$$

Case 2: (A<B)

,$L = L_{n-1} + x_{n-1} L_{n-2} + x_{n-1} x_{n-2} L_{n-3} + \ldots\ldots + x_{n-1}. x_{n-2} \ldots\ldots x_1 L_0$

Here n=2 ; $L_i = \overline{A_i}\, B_i$ therefore $L_1 = \overline{A_1}\, B_1$ and $L_0 = \overline{A_0}\, B_0$

$L = L_{2-1} + x_{2-1} L_{2-2} + x_{2-1} x_{2-2} + \ldots\ldots + \ldots\ldots x_1 L_0$

$= L_1 + x_1 L_0$

$L = \overline{A_1}\, B_1 + (\overline{A_1}\,\overline{B_1} + A_1 B_1).\overline{A_0}\, B_0$

Case 3: (A>B)

$$G = G_{n-1} + x_{n-1} G_{n-2} + x_{n-1} x_{n-2} G_{n-3} + \ldots\ldots + x_{n-1}. x_{n-2} \ldots\ldots x_1 G_0$$

Here n=2 ; $G_i = A_i \overline{B_i}$ therefore, $G_1 = A_1 \overline{B_1}$ and $G_0 = A_0 \overline{B_0}$
$G = G_1 + x_1 G_0$ (G is representing greater than)

$$G = A_1 \overline{B_1} + (\overline{A_1}\,\overline{B_1} + A_1 B_1).\, A_0 \overline{B_0}$$

The logic diagram of 2 –bit magnitude comparator is given below

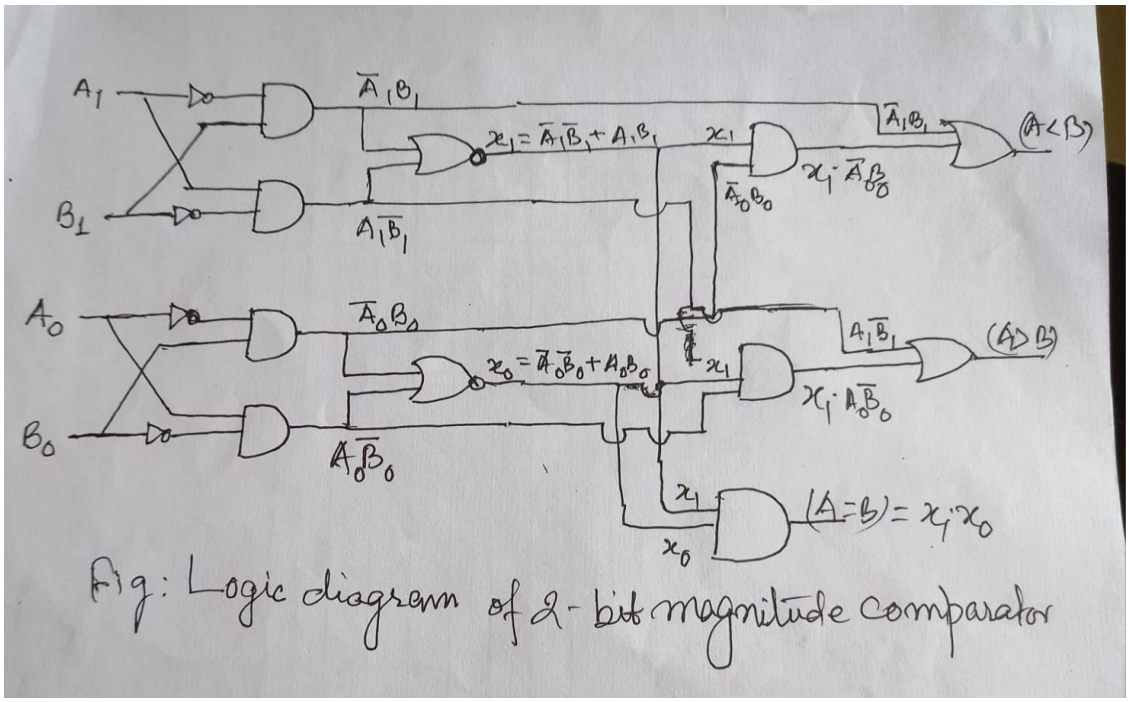
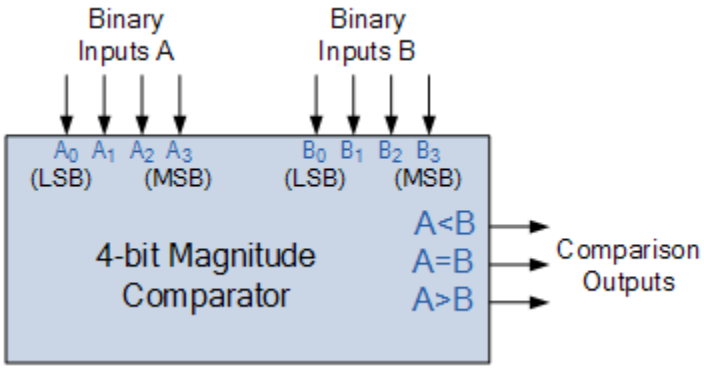


Fig: Logic diagram of 2-bit magnitude comparator

4- bit magnitude comparator

Similarly In a 4-bit comparator, A and B can be represented in binary as given equation below

$$A = A_3A_2A_1A_0$$
$$B = B_3B_2B_1B_0$$



The condition of A=B can be achieved in the following four cases.

1. If $A_3 = B_3$
2. If $A_2 = B_2$
3. If $A_1 = B_1$
4. If $A_0 = B_0$

This equality conditions can be expressed logically by Ex-NOR function

$x_i = \overline{A}_i\overline{B}_i + A_iB_i$ (for n-bit comparator)

Where, i = 0 to (n-1);

Here n is the comparator input bits to the corresponding number. For 4–bit comparator the magnitude comparator i= 0, 1,2,3

- The equality of the two numbers, A and B, is represented in a combinational circuit by an output binary variable by the symbol (A = B). This binary variable is equal to 1 if the input numbers, A and B, are equal, and it is equal to 0 otherwise.
- For the equality condition to exist, all x_i variables must be equal to 1. This can be performed by an AND operation of all variables:

$$(A = B) = x_3\, x_2\, x_1\, x_0$$

the binary variable (A = B) is equal to 1 only if all pairs of digits of the two numbers are equal.

- To determine if A is greater than or less than B, we inspect the relative magnitudes of pairs of significant digits starting from the most significant position.
- If the two digits are equal, we compare the next lower significant pair of digits. This comparison continues until a pair of unequal digits is reached.
- If the corresponding digit of A is 1 and that of B is 0, we conclude that A > B. If the corresponding digit of A is 0 and that of B is 1, we have that A < B.
- The sequential comparison can be expressed logically by the following two Boolean functions:

$$(A > B) = A_3B'_3 + x_3A_2B'_2 + x_3x_2A_1B'_1 + x_3x_2x_1A_0B'_0$$

$$(A < B) = A'_3B_3 + x_3A'_2B_2 + x_3x_2A'_1B_1 + x_3x_2x_1A'_0B_0$$

These above equations can be find from the equations also.

- the symbols (A > B) and (A < B) are binary output variables which are equal to 1 when A > B or A < B, respectively.

- The logic diagram of the 4-bit magnitude comparator is shown in Fig. The four x outputs are generated with equivalence (exclusive-NOR) circuits and applied to an AND gate to give the output binary variable (A = B).
- The other two outputs use the x variables to generate the Boolean functions listed above. This is a multilevel implementation and, as clearly seen, it has a regular pattern. The procedure for obtaining magnitude comparator circuits for binary numbers with more than four bits should be obvious from this example. The same circuit can be used to compare the relative magnitudes of two BCD digits.

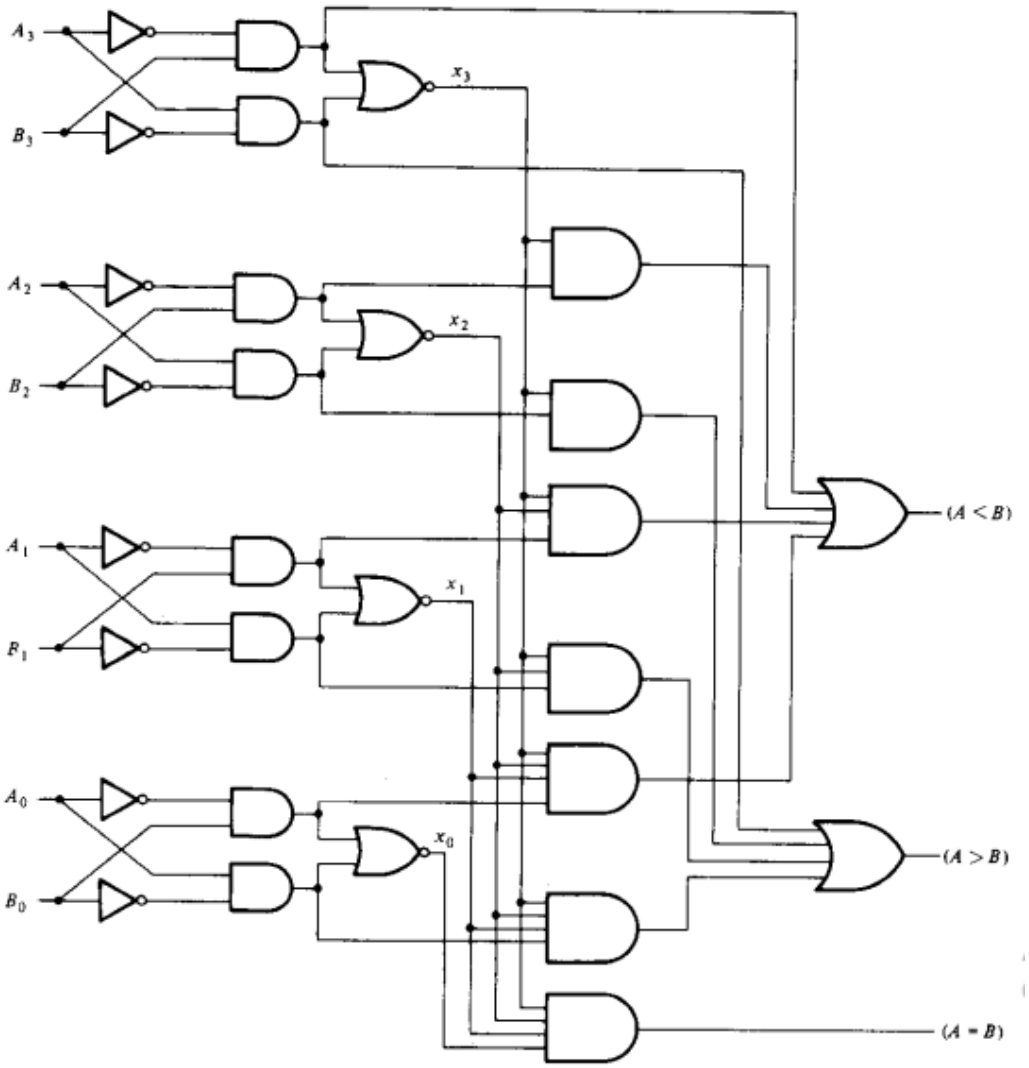


Figure: Logic diagram of 4-bit magnitude comparator