1- Optimize the following Boolean functions *F* together with the don't-care conditions *d*:

F A B C D = *m* 0 1 7 13 15 , *d A B C D* = *m* 2 6 8 9 10

2- Design a 16-to-1-line multiplexer using a 4-to-16-line decoder and a 16 2 AND-OR.

3- Perform the arithmetic operations (_36) _ (_24) and (_35) _ (_24) in binary using signed 2s complement representation for negative numbers in 8 bits.

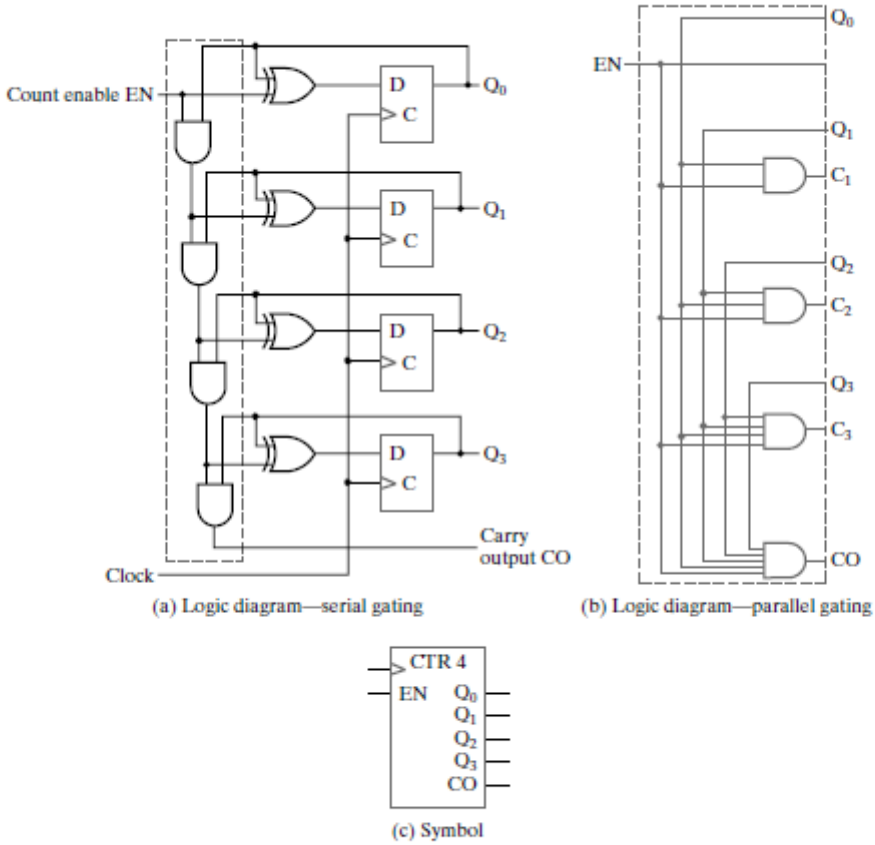
4- The following binary numbers have a sign in the leftmost position and, if negative, are in 2s complement form. Perform the indicated arithmetic operations and verify the answers.
(a) 100111 _ 111001 **(c)** 110001 _ 010010
(b) 001011 _ 100110 **(d)** 101110 _ 110111
Indicate whether overflow occurs for each computation.

5- Design a combinational circuit that compares two 4-bit unsigned numbers *A* and *B* to see whether B is greater than A. The circuit has one output *X*, so that *X* =1 if *A* < *B* and *X* = 0 if *A*>= *B*.

6- Design a sequential circuit with two *D* flip-flops *A* and *B* and one input *X*. When *X* = 0, the state of the circuit remains the same. When *X* = 1, the circuit goes through the state transitions from 00 to 10 to 11 to 01, back to 00, and then repeats.

7- Use *D* flip-flops and gates to design a binary counter with each of the following repeated binary sequences:
(a) 0, 1, 2 **(b)** 0, 1, 2, 3, 4, 5

8- **(a)** Using the synchronous binary counter of Figure 13 and an AND gate, construct a counter that counts from 0000 through 1010.
(b) Repeat for a count from 0000 to 1110. Minimize the number of inputs to the AND gate.



□ **FIGURE 13**
4-Bit Synchronous Binary Counter