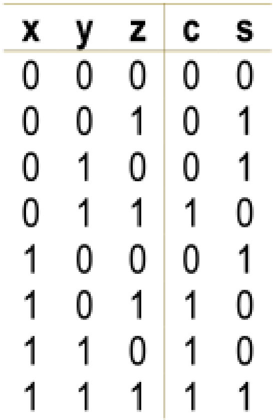
Application of Decoder, Encoder

By Engr. Rimsha

Combination Logic Implementation

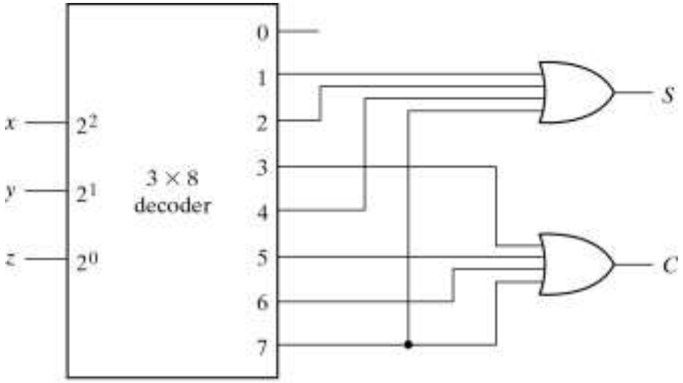
- A decoder provides the 2^n minterms of n input variables.
- Any function is can be expressed in sum of minterms.
- Use a decoder to make the minterms and an external OR gate to make the logical sum.
- In this way any combinational circuit with n inputs and m outputs can be implemented with an n-to-2^n line decoder and m OR gates. Such implementation needs that the Boolean function is expressed in sum of minterms
- For example: consider a full adder.
  - $S(x,y,z) = \Sigma(1,2,4,7)$
  - $C(x,y,z) = \Sigma(3,5,6,7)$



x	y	z	c	s
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Implementation of Full Adder with a Decoder

  - $S(x,y,z) = \Sigma(1,2,4,7)$
  - $C(x,y,z) = \Sigma(3,5,6,7)$
- There are three inputs and eight outputs so we need 3-to-8-line decoder
- Two OR gates are required for logical sum of the desired minterms



Implementation of Function with a Decoder Contd..

- $F_1(x,y,z) = \Sigma(0,1,3,5,6,7)$
- $F_2(x,y,z) = \Sigma(1,2,3,4,7)$

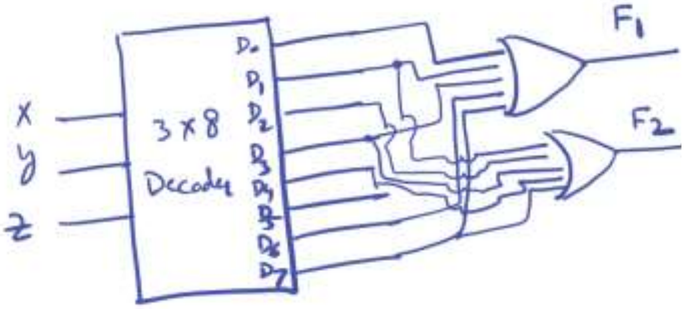
- A function with long list of minterms requires an OR gate with large number of inputs
- A function having a list of K minterms can be expressed in its complemented form F’ with 2^n-K minterms
- If the number of minterms in a function is greater than $2^n/2$ then F’ can be expressed with fewer minterms
- In such case it is advantageous to use a NOR gate to sum the minterms of F’. The output of the NOR gate complements this sum and generates the normal output F

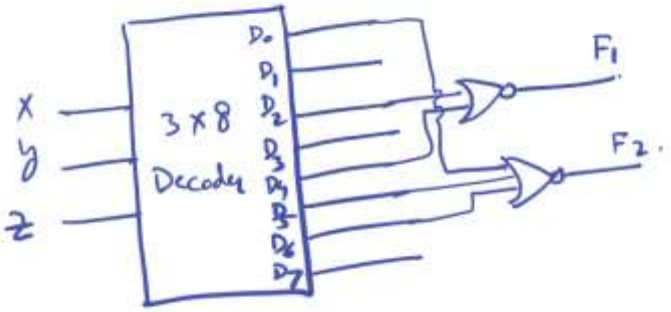
Implementation of Function with a Decoder Contd..

- A function with long list of minterms requires an OR gate with large number of inputs
- A function having a list of K minterms can be expressed in its complemented form F’ with 2^n-K minterms
- If the number of minterms in a function is greater than $2^n/2$ then F’ can be expressed with fewer minterms
- In such case it is advantageous to use a NOR gate to sum the minterms of F’. The output of the NOR gate complements this sum and generates the normal output F

- $F_1(x,y,z) = \Sigma(0,1,3,5,6,7)$
- $F_2(x,y,z) = \Sigma(1,2,3,4,7)$

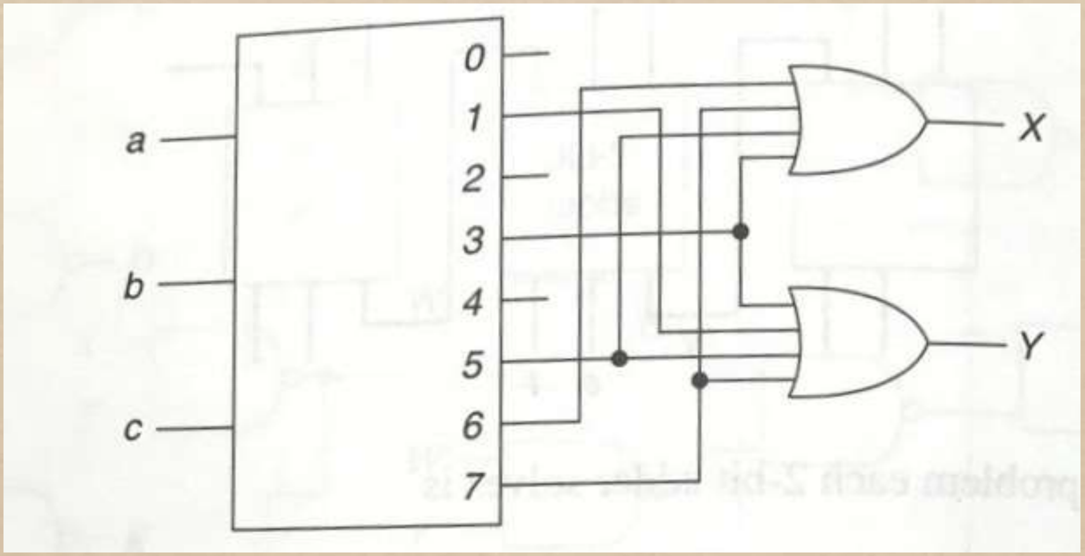
- $F_1'(x,y,z) = \Sigma(2,4)$
- $F_2'(x,y,z) = \Sigma(0,5,6)$





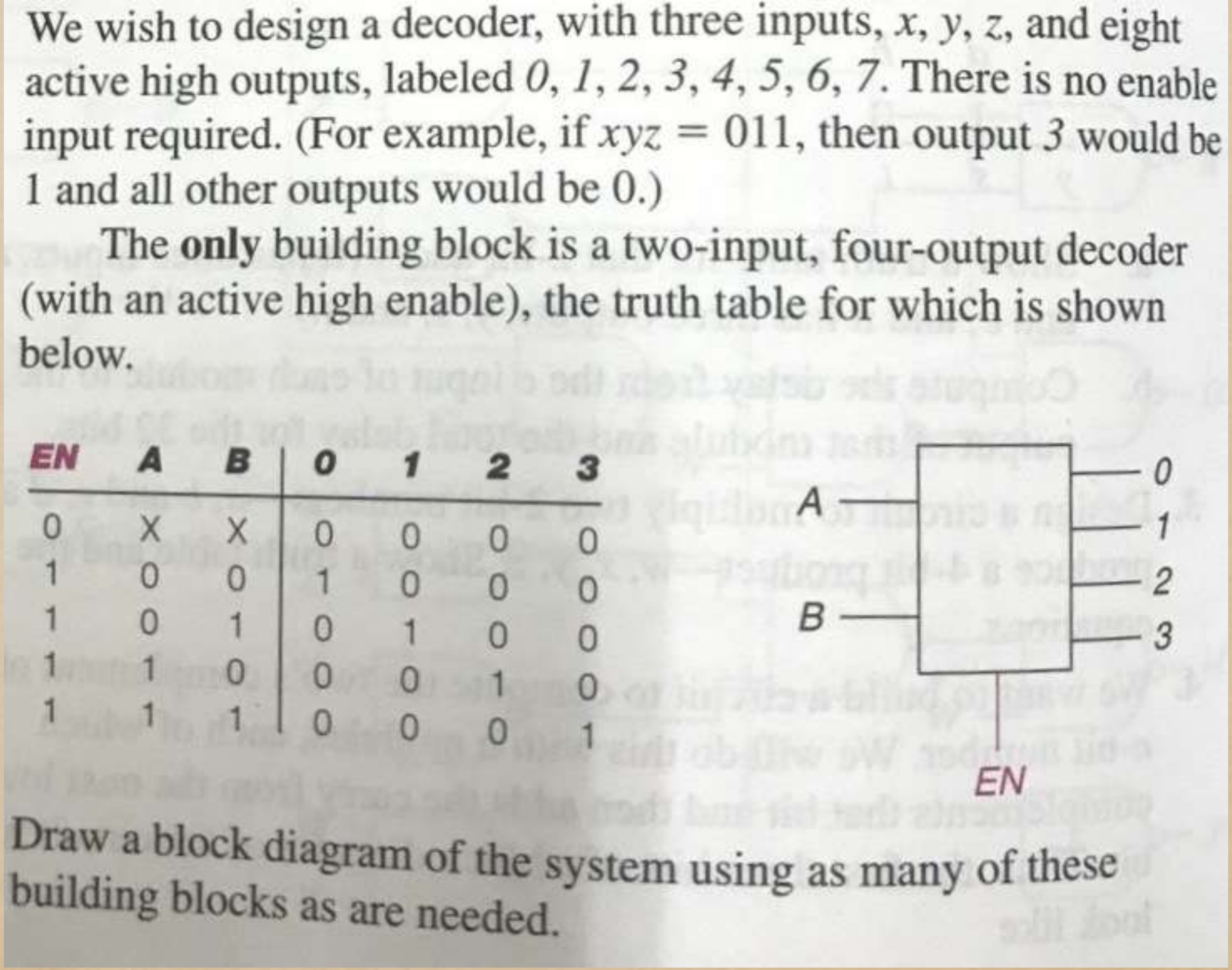
Practice Problems

P1 : Consider the following circuit with an active high output decoder. Draw a truth table for X and Y in terms of a, b and c



Practice Problems Contd..

P2



We wish to design a decoder, with three inputs, *x*, *y*, *z*, and eight active high outputs, labeled *0*, *1*, *2*, *3*, *4*, *5*, *6*, *7*. There is no enable input required. (For example, if *xyz* = 011, then output *3* would be 1 and all other outputs would be 0.)

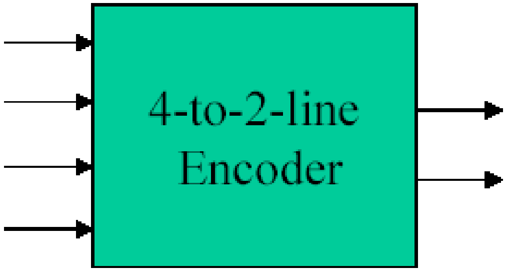
The **only** building block is a two-input, four-output decoder (with an active high enable), the truth table for which is shown below.

EN	*A*	*B*	*0*	*1*	*2*	*3*
0	X	X	0	0	0	0
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1

Draw a block diagram of the system using as many of these building blocks as are needed.

Encoders

- An encoder is a digital circuit that performs the inverse operation of a decoder.
- An encoder has 2^n (or fewer) input lines and n output lines.
- The output lines generate the binary code corresponding to the input value



End of Lecture