NATIONAL UNIVERSITY OF SCIENCES & TECHNOLOGY
PAKISTAN

Chapter4: Combinational Logic

Lecture7- Study Encoders, Function Implementation using Encoders

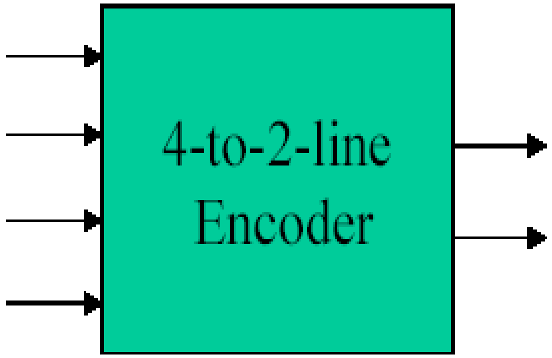
Engr. Arshad Nazir, Asst Prof
Dept of Electrical Engineering
SEECS

Objectives

- Study design and applications of Encoders
- Problem solving for Encoders

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



Encoder Design Example

- An example of encoder is octal-to-binary encoder
- It has eight inputs (one for each octal digits) and three outputs that generate the corresponding binary number
- It is assumed that only one input has a value of 1 at any given time
- The encoder can be implemented with OR gates whose inputs are determined directly from the truth table
- Output z is equal to 1 when the input octal digit is 1,3,5 or 7. Output y is 1 for octal digits 2,3,6 or 7 and output x is 1 for digits 4,5,6 or 7. These conditions can be expressed as by the Boolean functions as shown in the next slide.

Truth Table: Octal to Binary Encoder

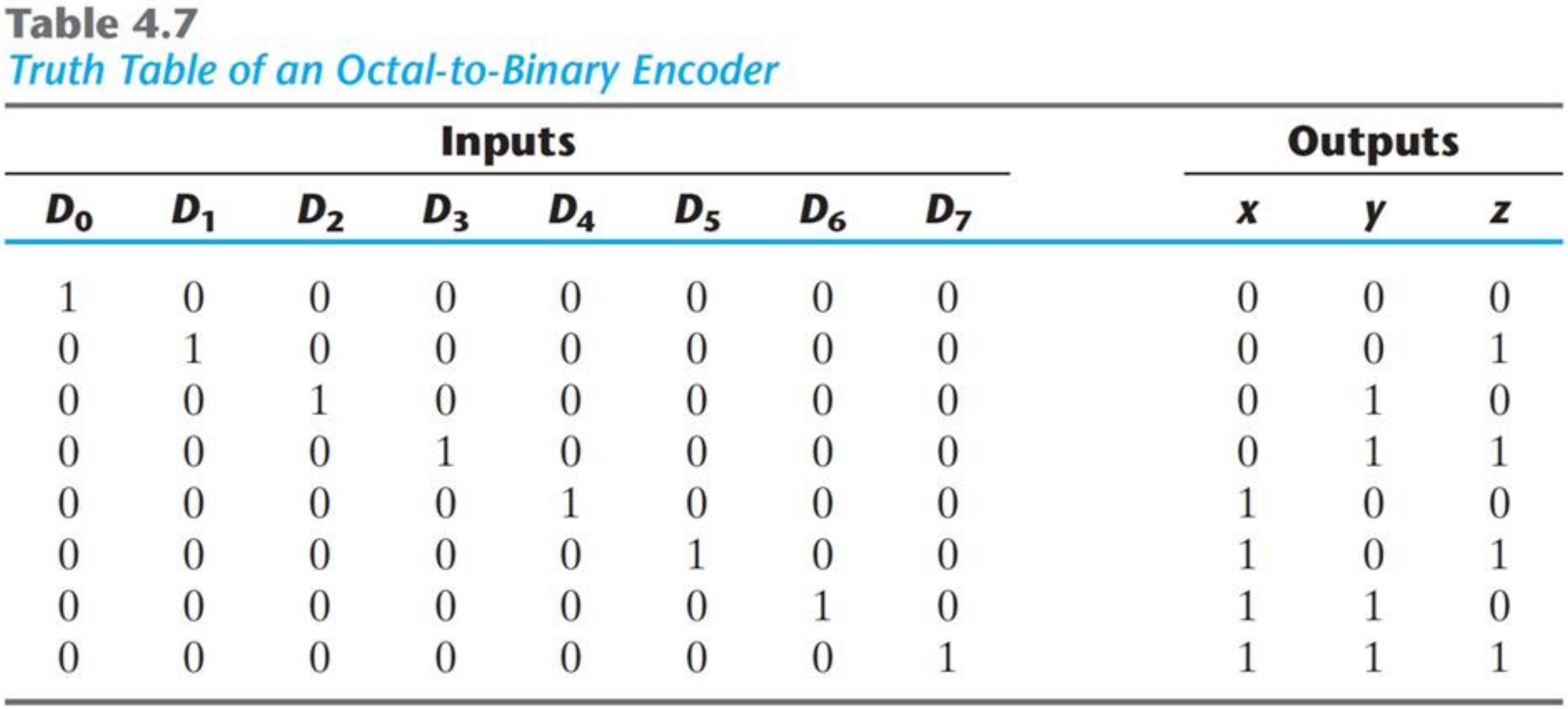


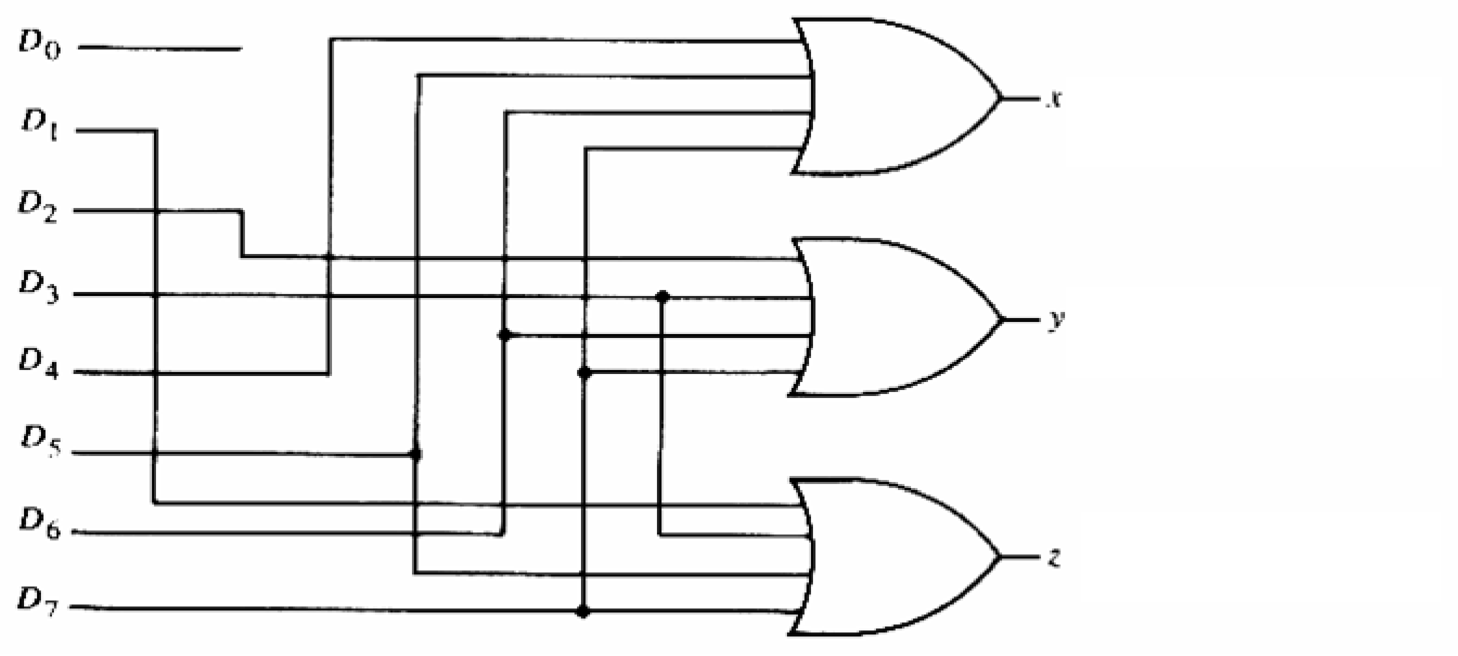
Table 4.7

Truth Table of an Octal-to-Binary Encoder

Inputs								Outputs		
D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	x	y	z
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

$z=D_1+D_3+D_5+D_7$ **$y=D_2+D_3+D_6+D_7$** **$x=D_4+D_5+D_6+D_7$**

Octal-to-Binary Encoder Implementation



$z = D_1 + D_3 + D_5 + D_7$ $y = D_2 + D_3 + D_6 + D_7$ $x = D_4 + D_5 + D_6 + D_7$

Previous Encoder Limitations

- The encoder defined in the last slide has the limitation that only one input can be active at any given time. If two inputs are active simultaneously, the output produces an undefined combination.
- For example
  - ➢ If input D_3 and D_6 are 1 simultaneously the output of the encoder will be 111 (see truth table and Boolean function for outputs). Since $z=D_1+D_3+D_5+D_7$ $y=D_2+D_3+D_6+D_7$ $x=D_4+D_5+D_6+D_7$
  - ➢ This 111 doesn't represent either binary 3 or binary 6
- To resolve this ambiguity, encoder circuit must establish an input priority to ensure that only one input is encoded.
- Another ambiguity is that when all inputs are zero, output with all zeros is generated. This is same as when D_0 is equal to 1. This discrepancy can be resolved by providing an output to indicate that at least one input is equal to 1

Priority Encoder

- A priority encoder is an encoder circuit that includes the priority function.
- The operation of the priority encoder is such that if two or more inputs are equal to 1 at the same time, the input having the highest priority will take precedence
  - D_3 has the highest priority
  - D_0 has the lowest priority
- Valid bit indicator (V) is set to 1 when one or more inputs are equal to 1. If all inputs are 0, there is no valid inputs and V is equal to 0. The other two outputs are not inspected when V equals 0 and are specified as don’t care conditions

Priority Encoder: Truth Table

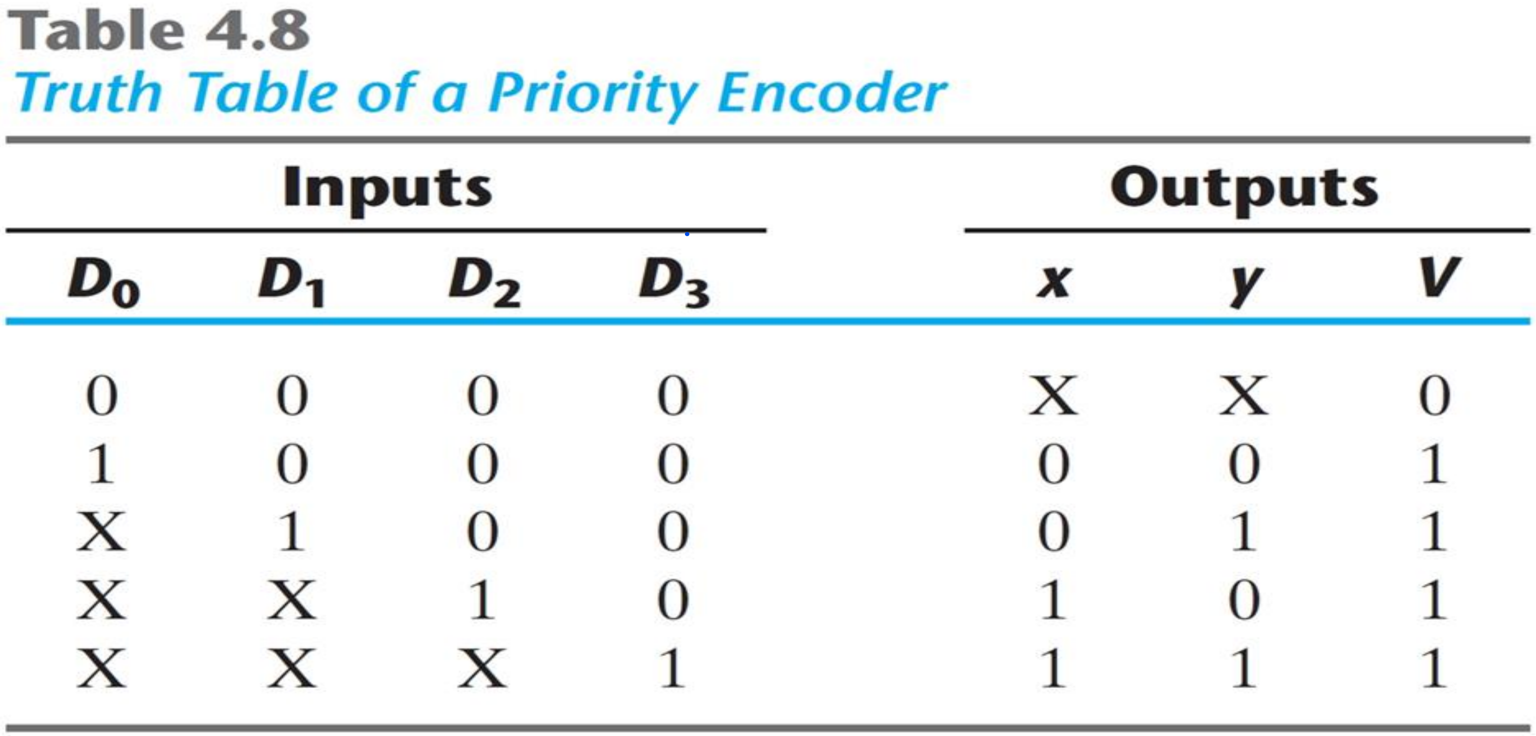


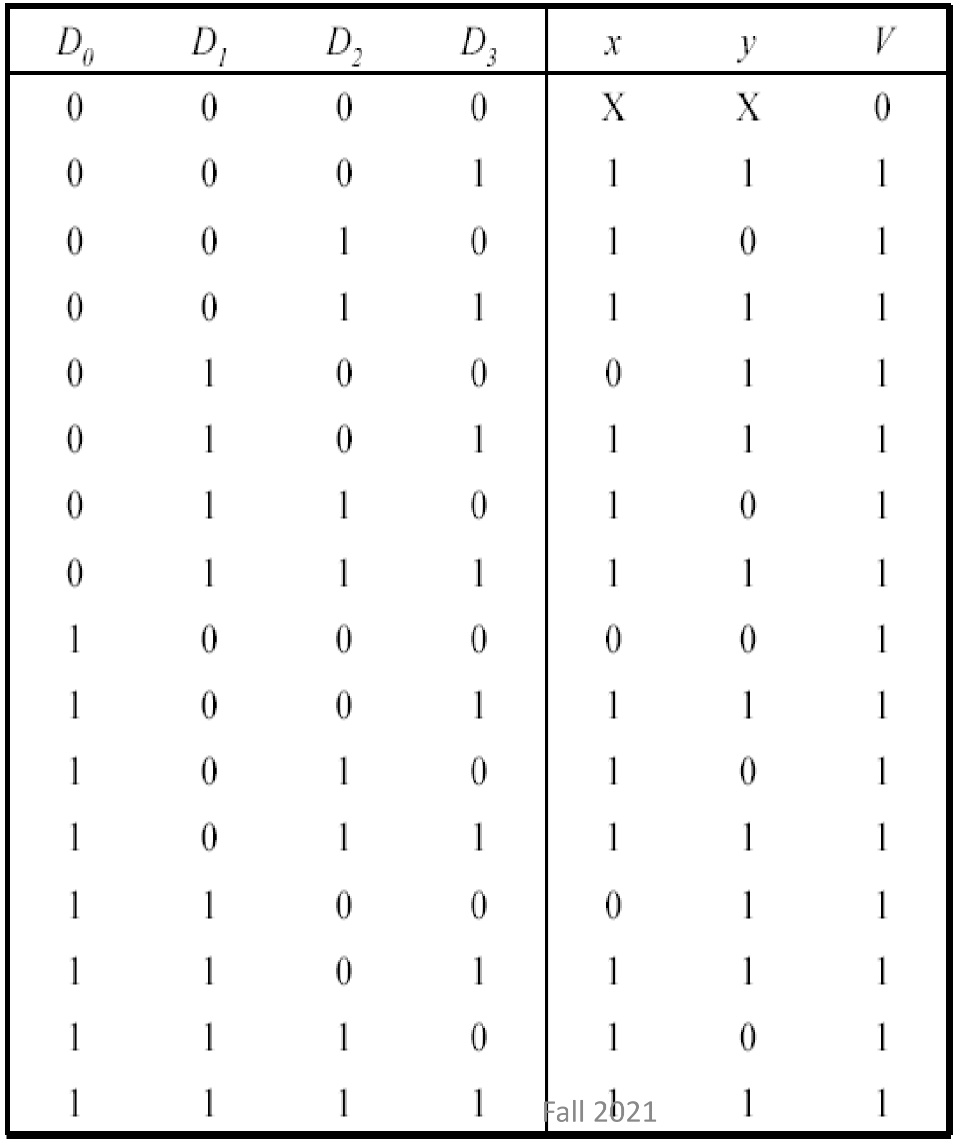
Table 4.8
Truth Table of a Priority Encoder

Inputs				Outputs		
D_0	D_1	D_2	D_3	x	y	V
0	0	0	0	X	X	0
1	0	0	0	0	0	1
X	1	0	0	0	1	1
X	X	1	0	1	0	1
X	X	X	1	1	1	1

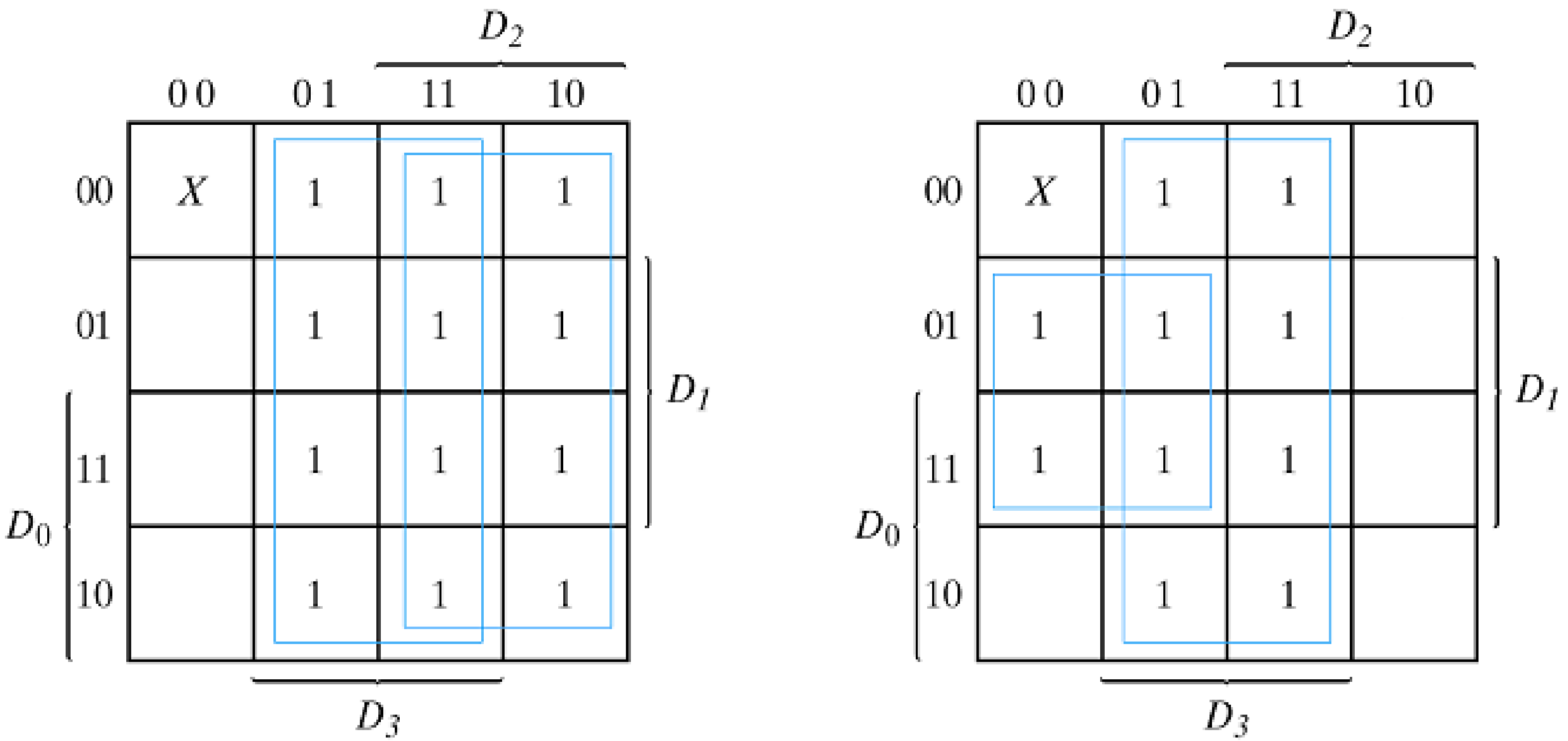
x: don't-care conditions in the output, used in the inputs to condense truth table, replaced by both 0 and then 1

V: valid output indication, implemented by OR function

Priority Encoder: Expanded Truth Table



D_0	D_1	D_2	D_3	x	y	V
0	0	0	0	X	X	0
0	0	0	1	1	1	1
0	0	1	0	1	0	1
0	0	1	1	1	1	1
0	1	0	0	0	1	1
0	1	0	1	1	1	1
0	1	1	0	1	0	1
0	1	1	1	1	1	1
1	0	0	0	0	0	1
1	0	0	1	1	1	1
1	0	1	0	1	0	1
1	0	1	1	1	1	1
1	1	0	0	0	1	1
1	1	0	1	1	1	1
1	1	1	0	1	0	1
1	1	1	1	1	1	1



	00	01	11	10
00	X	1	1	1
01		1	1	1
11		1	1	1
10		1	1	1

$x = D_2 + D_3$

	00	01	11	10
00	X	1	1	
01	1	1	1	
11	1	1	1	
10		1	1	

$y = D_3 + D_1D'_2$

Fig. 4-22 Maps for a Priority Encoder

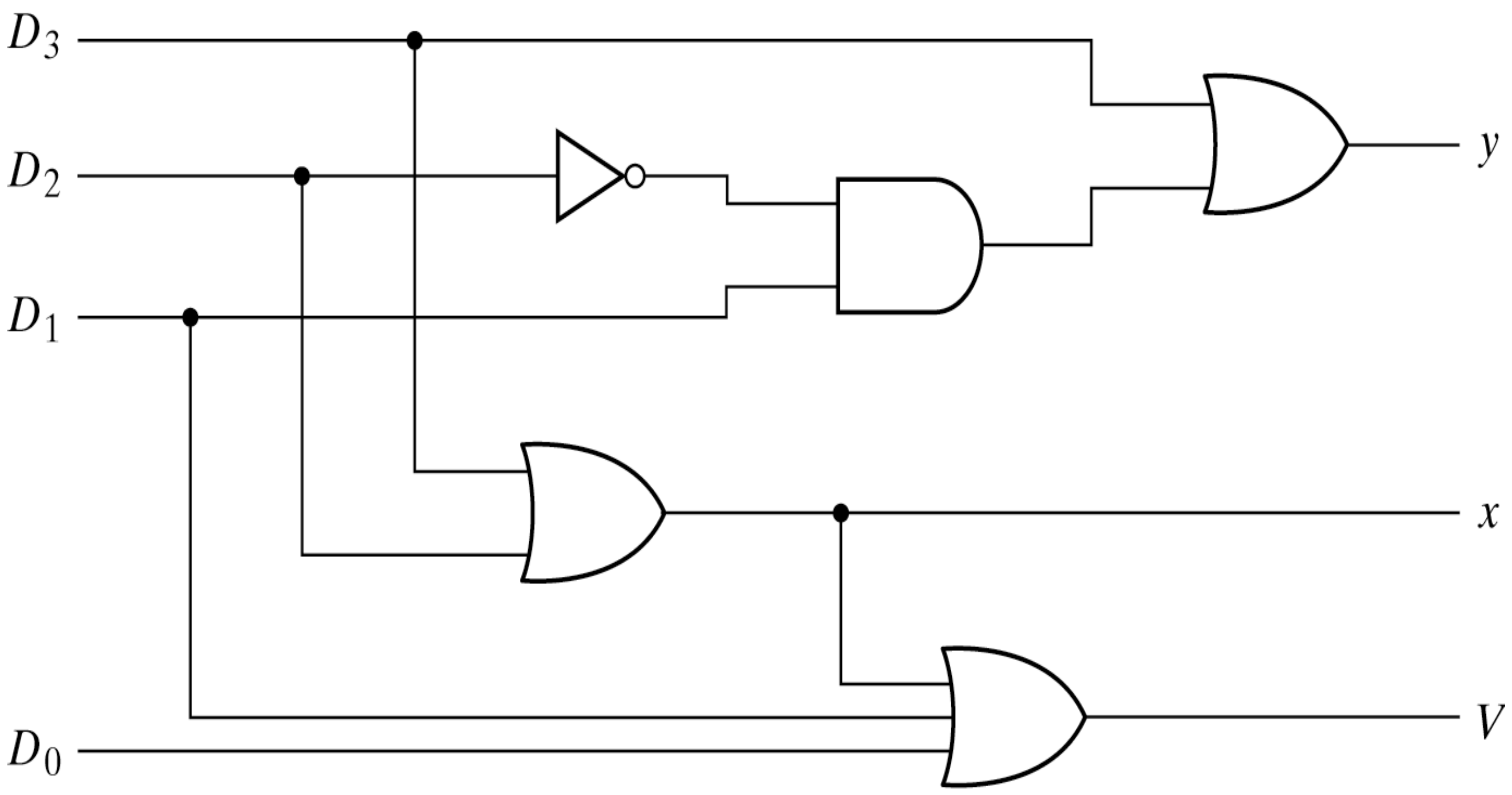
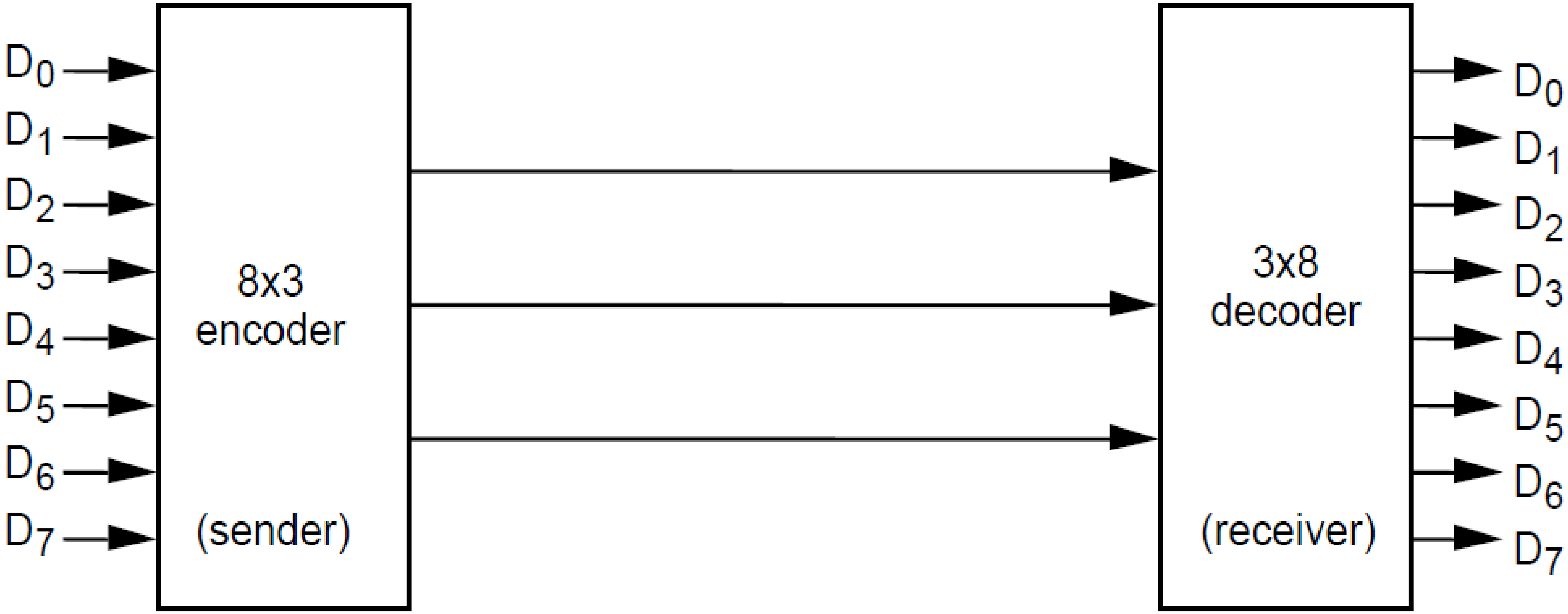


Fig. 4-23 4-Input Priority Encoder

Decoders and Encoders



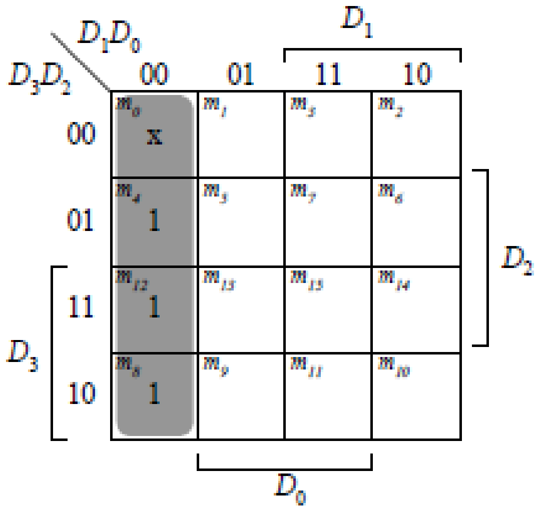
Main function of encoder and decoder

Problem Solving Session

Problem: 4-29

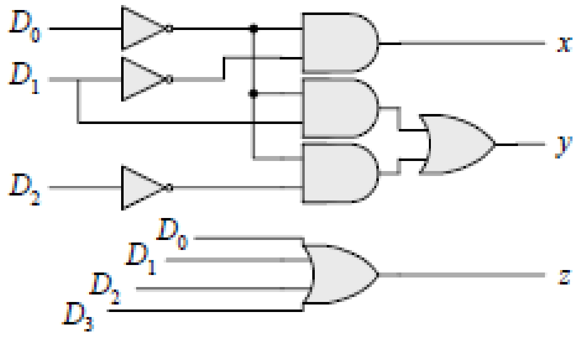
Design a 4-input priority Encoder with the inputs as in Table 4.8, but with the input D_0 having the highest priority and the input D_3 having the lowest priority.

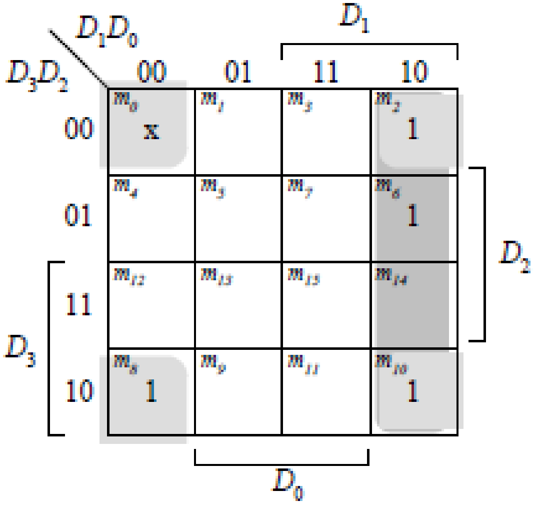
Inputs				*Outputs*		
D_3	D_2	D_1	D_0	X	Y	Z
0	0	0	0	x	x	0
x	x	x	1	0	0	1
x	x	1	0	0	1	1
x	1	0	0	1	0	1
1	0	0	0	1	1	1



D_3D_2 \ D_1D_0	00	01	11	10
00	m_0 x	m_1	m_3	m_2
01	m_4 1	m_5	m_7	m_6
11	m_{12} 1	m_{13}	m_{15}	m_{14}
10	m_8 1	m_9	m_{11}	m_{10}

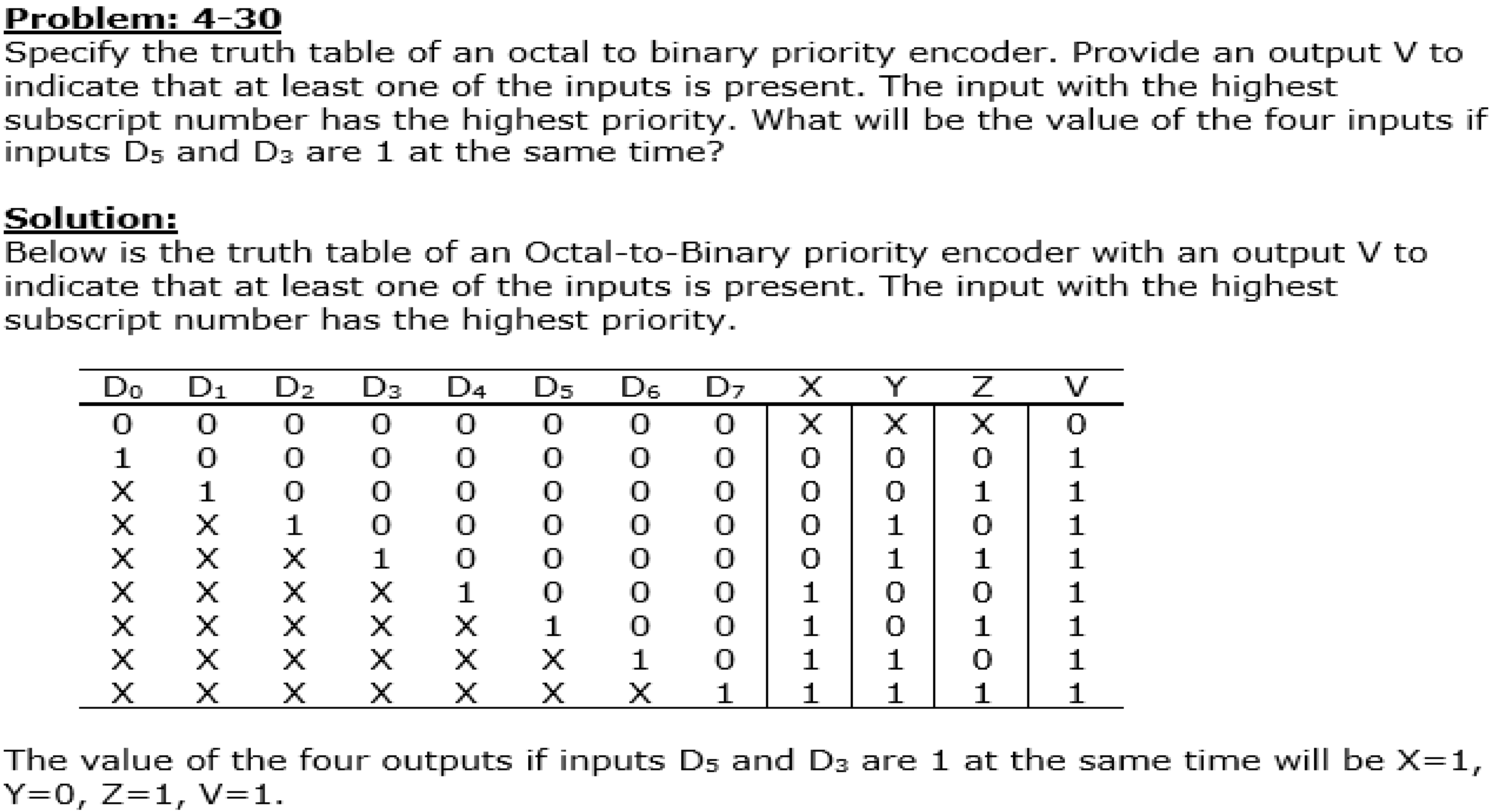
$$v = D_0 + D_1 + D_2 + D_3$$





D_3D_2 \ D_1D_0	00	01	11	10
00	m_0 x	m_1	m_3	m_2 1
01	m_4	m_5	m_7	m_6 1
11	m_{12}	m_{13}	m_{15}	m_{14}
10	m_8 1	m_9	m_{11}	m_{10} 1

$$y = D'_0D_1 + D'_0D'_2$$



Problem: 4-30

Specify the truth table of an octal to binary priority encoder. Provide an output V to indicate that at least one of the inputs is present. The input with the highest subscript number has the highest priority. What will be the value of the four inputs if inputs D_5 and D_3 are 1 at the same time?

Solution:

Below is the truth table of an Octal-to-Binary priority encoder with an output V to indicate that at least one of the inputs is present. The input with the highest subscript number has the highest priority.

D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	X	Y	Z	V
0	0	0	0	0	0	0	0	X	X	X	0
1	0	0	0	0	0	0	0	0	0	0	1
X	1	0	0	0	0	0	0	0	0	1	1
X	X	1	0	0	0	0	0	0	1	0	1
X	X	X	1	0	0	0	0	0	1	1	1
X	X	X	X	1	0	0	0	1	0	0	1
X	X	X	X	X	1	0	0	1	0	1	1
X	X	X	X	X	X	1	0	1	1	0	1
X	X	X	X	X	X	X	1	1	1	1	1

The value of the four outputs if inputs D_5 and D_3 are 1 at the same time will be X=1, Y=0, Z=1, V=1.

The End