Combinational Logic Design

Dr. Chandan Karfa
CSE IIT Guwahati

- Encoder
- Decoder

Source

- Chapter 4: M. M. Mano and M. D. Ciletti, Digital Design, 5th Ed., Pearson Education.

- Chapter 5: Z. Kohavi and N. Jha, Switching and Finite Automata Theory, 3rd Ed., Cambridge University Press, 2010.

Encoder

- An encoder has 2^n (or fewer) input lines and n output lines.
- The output lines, as an aggregate, generate the binary code

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

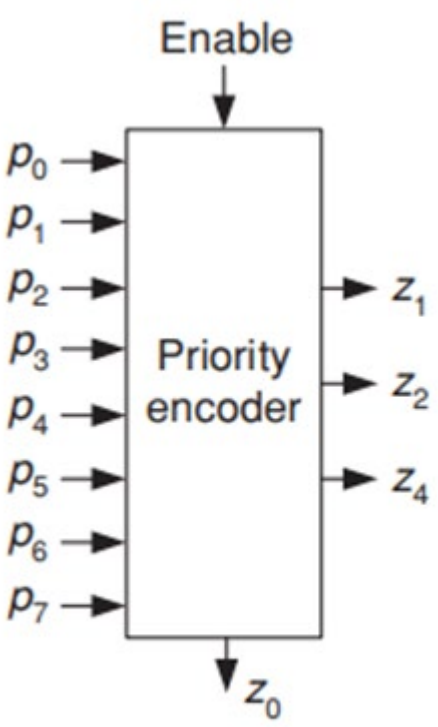
$$z = D_1 + D_3 + D_5 + D_7$$
$$y = D_2 + D_3 + D_6 + D_7$$
$$x = D_4 + D_5 + D_6 + D_7$$

Limitations: 1. Assumes only one input can be active at any given time. If two inputs are active simultaneously, the output produces an undefined combination.
2. Output with all 0's is generated when all the inputs are 0; but this output is the same as when *D*0 is equal to 1.

Priority encoders

- It is a device with n input lines and log2n output lines.
- Input lines represent request services.
- When two lines pi and pj , such that i>j , request service simultaneously, line pi has priority over line pj .
- The encoder produces a binary output code indicating which of the input lines requesting service has the highest priority.
- Pi indicates request service assuming value 1.

Design of priority encoder



(*a*) Block diagram.

Input lines								Outputs		
p_0	p_1	p_2	p_3	p_4	p_5	p_6	p_7	z_4	z_2	z_1
1	0	0	0	0	0	0	0	0	0	0
ϕ	1	0	0	0	0	0	0	0	0	1
ϕ	ϕ	1	0	0	0	0	0	0	1	0
ϕ	ϕ	ϕ	1	0	0	0	0	0	1	1
ϕ	ϕ	ϕ	ϕ	1	0	0	0	1	0	0
ϕ	ϕ	ϕ	ϕ	ϕ	1	0	0	1	0	1
ϕ	ϕ	ϕ	ϕ	ϕ	ϕ	1	0	1	1	0
ϕ	ϕ	ϕ	ϕ	ϕ	ϕ	ϕ	1	1	1	1

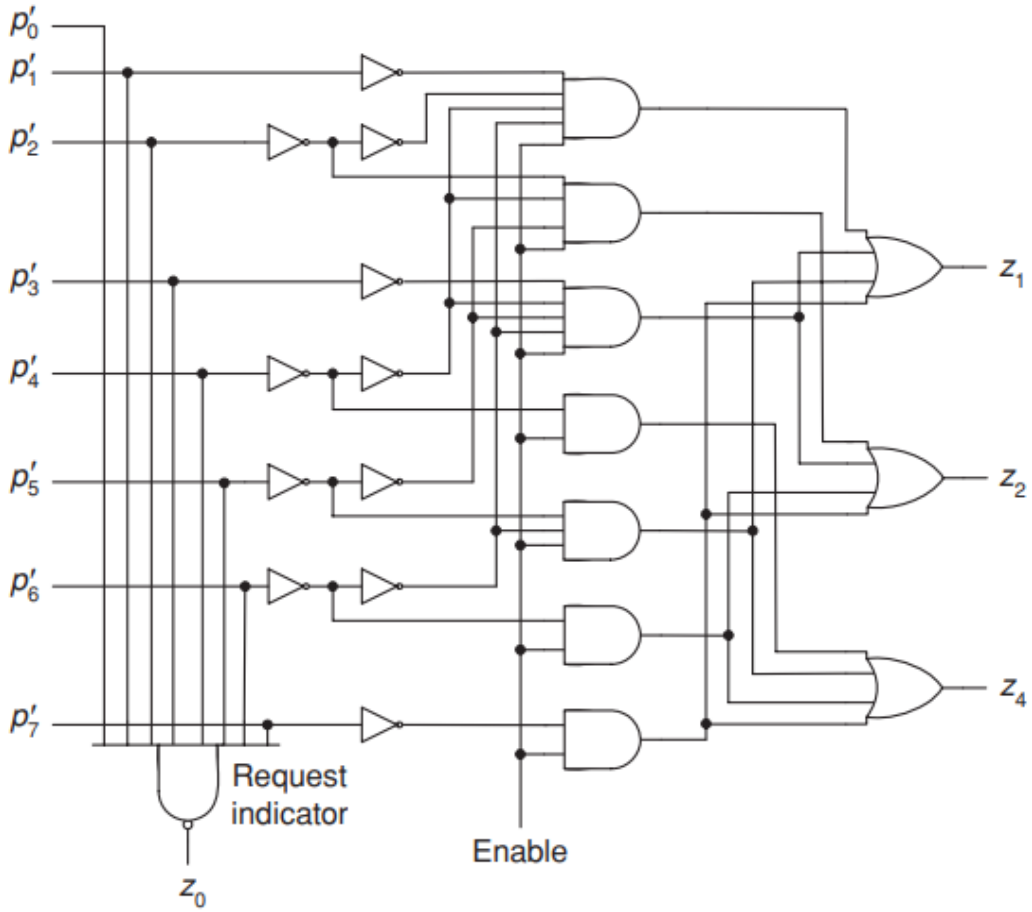
(*b*) Truth table.

$$z_4 = p_4 p_5' p_6' p_7' + p_5 p_6' p_7' + p_6 p_7' + p_7.$$

$$z_4 = p_4 + p_5 + p_6 + p_7.$$

$$\begin{aligned} z_2 &= p_2 p_3' p_4' p_5' p_6' p_7' + p_3 p_4' p_5' p_6' p_7' + p_6 p_7' + p_7 \\ &= p_2 p_4' p_5' + p_3 p_4' p_5' + p_6 + p_7, \\ z_1 &= p_1 p_2' p_3' p_4' p_5' p_6' p_7' + p_3 p_4' p_5' p_6' p_7' + p_5 p_6' p_7' + p_7 \\ &= p_1 p_2' p_4' p_6' + p_3 p_4' p_6' + p_5 p_6' + p_7. \end{aligned}$$

Implementation of encoder



(c) Logic diagram.

$$z_4 = p_4 + p_5 + p_6 + p_7.$$

$$z_2 = p_2 p'_4 p'_5 + p_3 p'_4 p'_5 + p_6 + p_7,$$

$$z_1 = p_1 p'_2 p'_4 p'_6 + p_3 p'_4 p'_6 + p_5 p'_6 + p_7.$$

- The circuit also has an Enable signal
- It contains an output z_0 that indicates whether any requests are present. Specifically, $z_0 = 0$ if there is no request and $z_0 = 1$ if there are one or more requests present.

Decoders

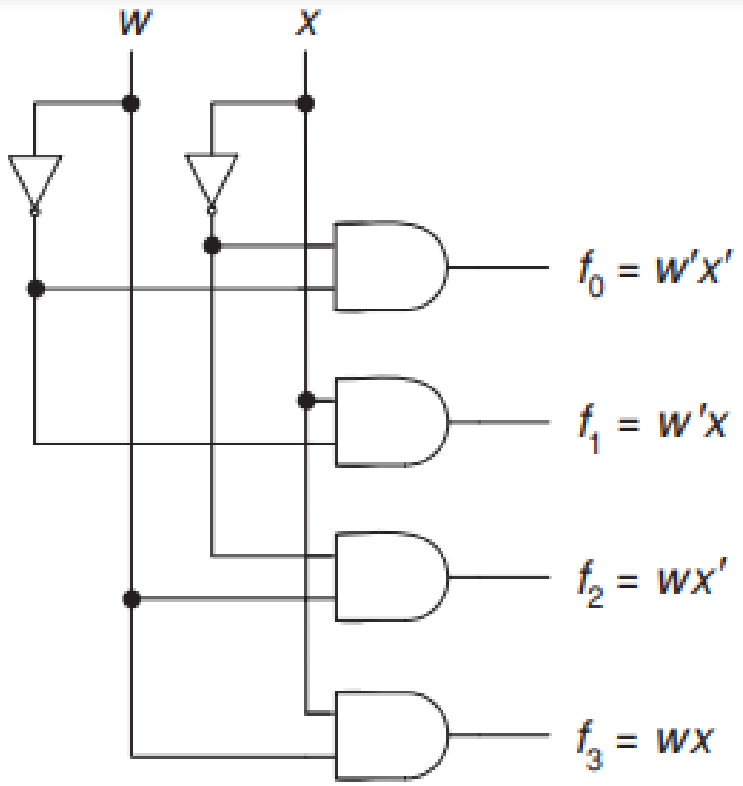
- It is a combinational circuit with n inputs and at most 2^n outputs.

- Its characteristic property is that, *'for every combination of input values, only one output value will be equal to 1 at any given time.'*

- Applications of decoders in digital technology:
  1. Used to route input data to a specified output line.
  2. Decode memory address
  3. They may be used for data distribution i.e. demultiplexing.
  4. they are also used for implementing arbitrary switching functions.

Decoders

- The figure shows a basic 2-to-4 decoder.
- If w and x are the input variables then each output corresponds to a different minterm of two variables.



(*a*) A 2-to-4 decoder.

3-to-8 Decoder

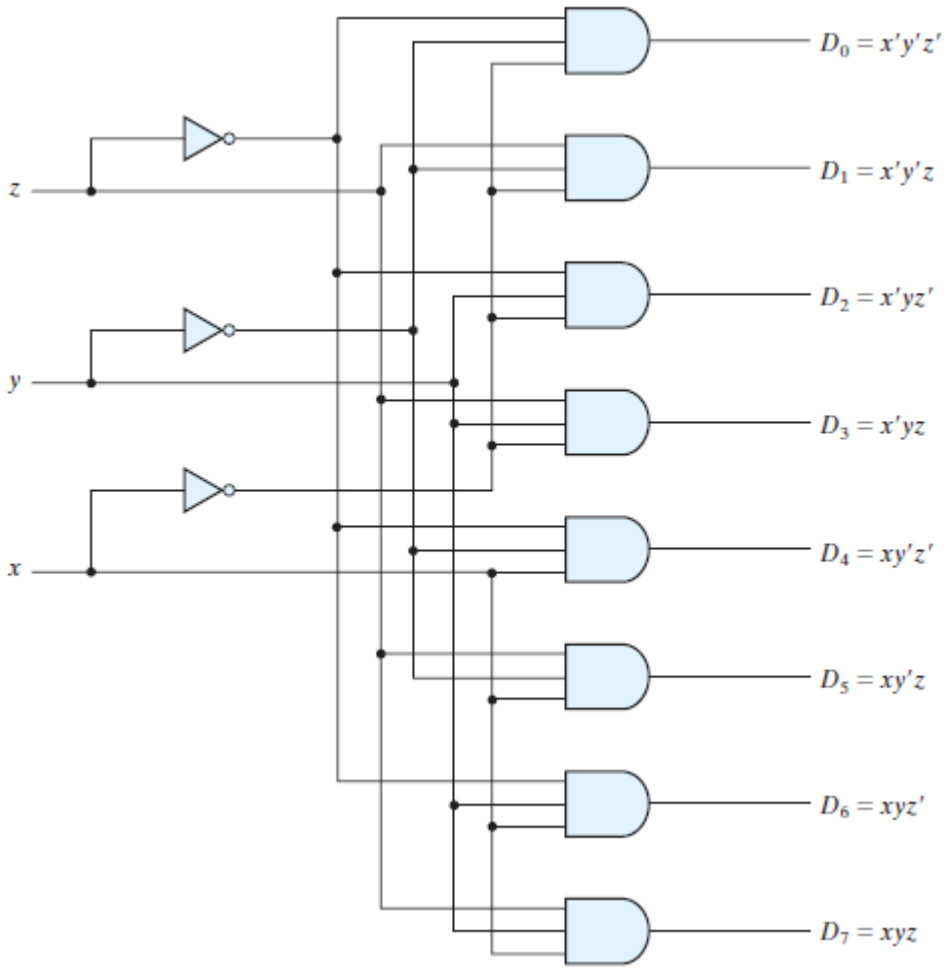
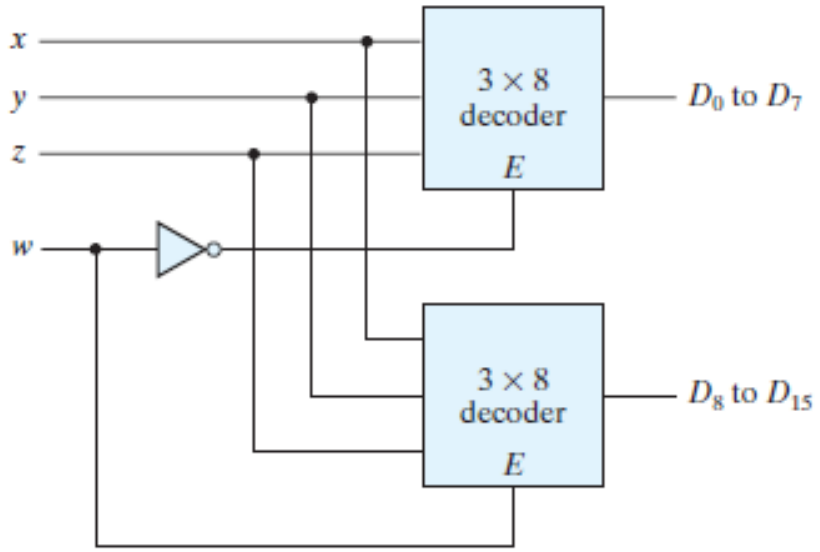


Table 4.6
Truth Table of a Three-to-Eight-Line Decoder

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

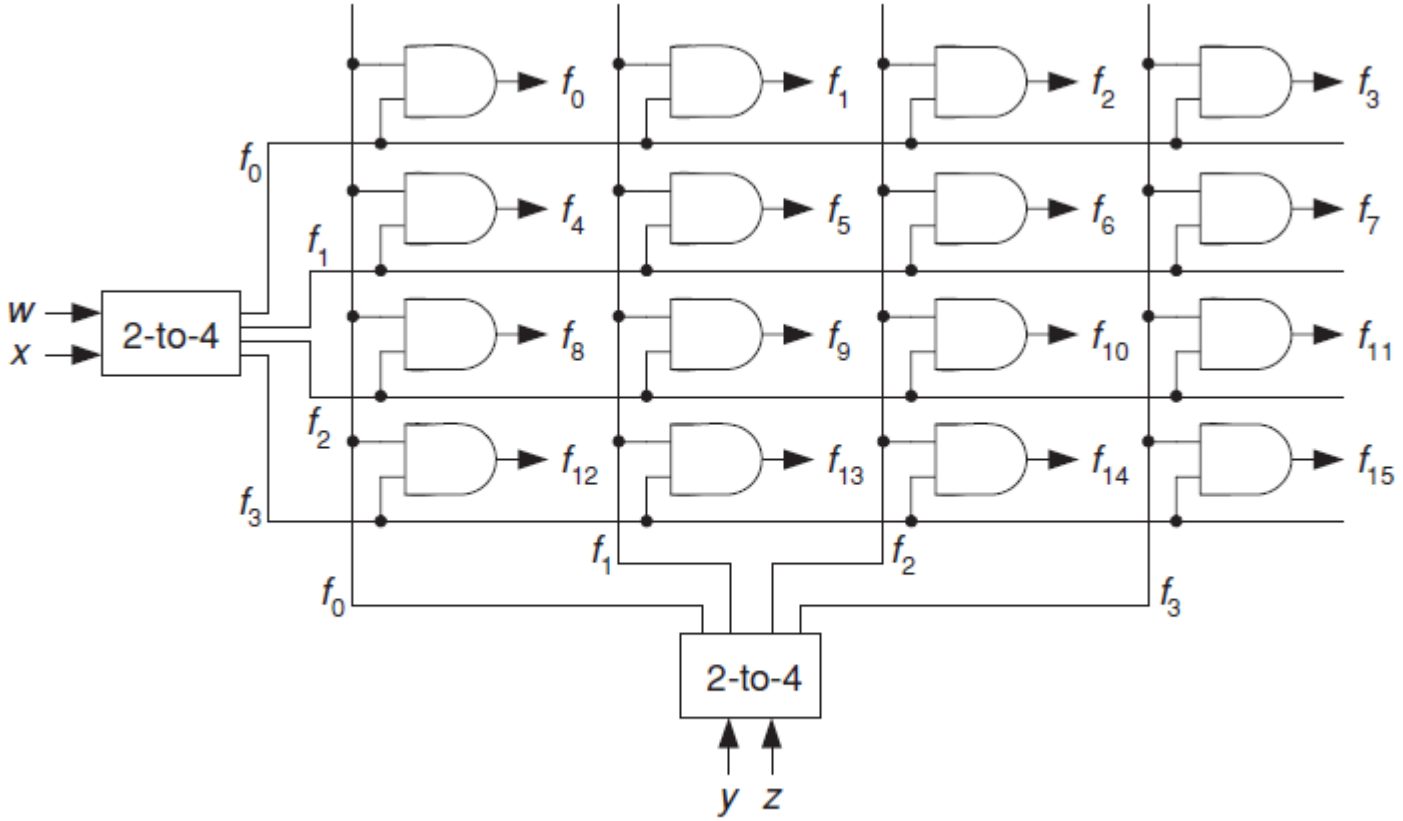
4-to-16 Decoder

- Using two 3-to-eight decoder



4-to-16 Decoder

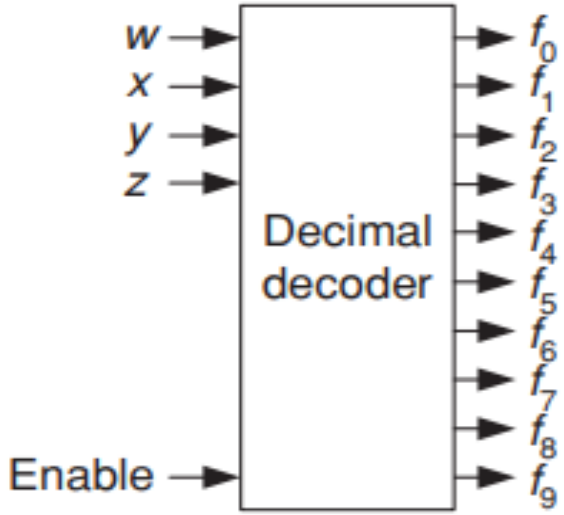
- Using 2-to-4 decoders and switching matrix



(*b*) Design of a 4-to-16 decoder.

Decimal decoder

- Not all decoder with n inputs has exactly 2^n outputs.
- It converts information from BCD to decimal.
- It has four inputs: w, x, y and z.
- w is the most significant and z the least significant digit, and 10 outputs, f0 through f9, corresponding to the decimal numbers.



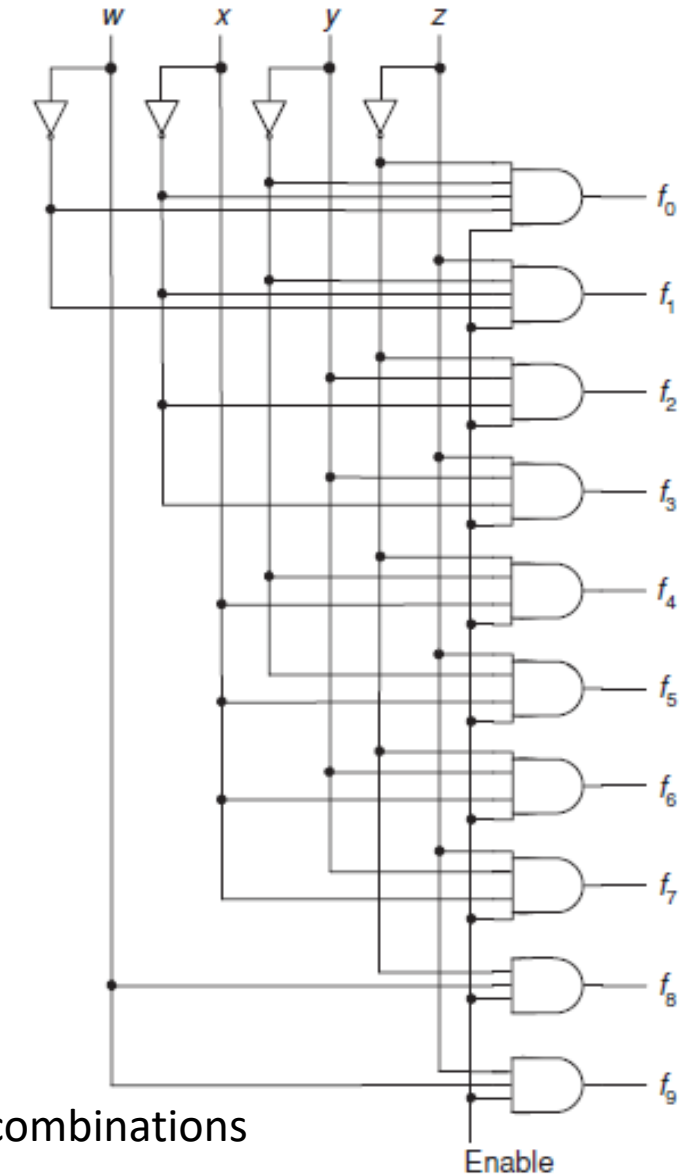
(*a*) Block diagram.

yz \ *wx*	00	01	11	10
00	f_0	f_4	ϕ	f_8
01	f_1	f_5	ϕ	f_9
11	f_3	f_7	ϕ	ϕ
10	f_2	f_6	ϕ	ϕ

(*b*) Map.

yz \ wx	00	01	11	10
00	f_0	f_4	ϕ	f_8
01	f_1	f_5	ϕ	f_9
11	f_3	f_7	ϕ	ϕ
10	f_2	f_6	ϕ	ϕ

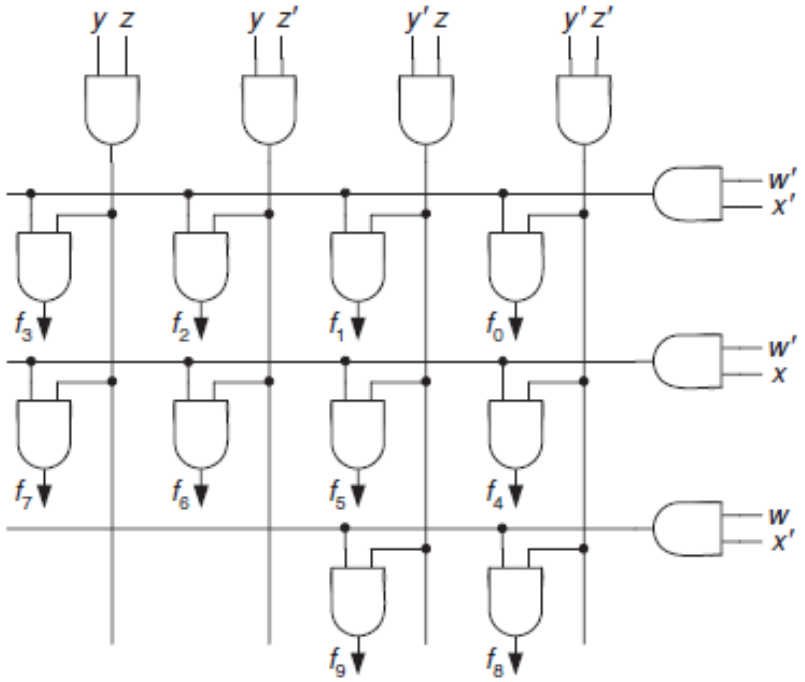
(*b*) Map.



(*c*) Logic diagram.

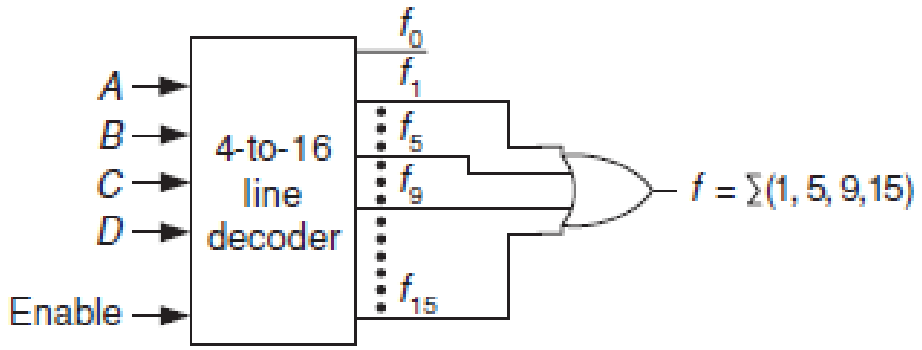
taken advantage of the don’t-care combinations

BCD-to-Decimal Decoder using gate matrix



Switching function realization using decoder

- A decoder with exactly *n* inputs and 2^n outputs can also be used to implement any switching function.
- Each output of such a decoder realizes one distinct minterm.
- Thus, by connecting the appropriate outputs to an OR gate, the required function can be realized.



Demultiplexer

- It is a decoder with one data input and n address inputs.

- It directs the input data to any one of the 2n outputs, as specified by the n-bits input address.