

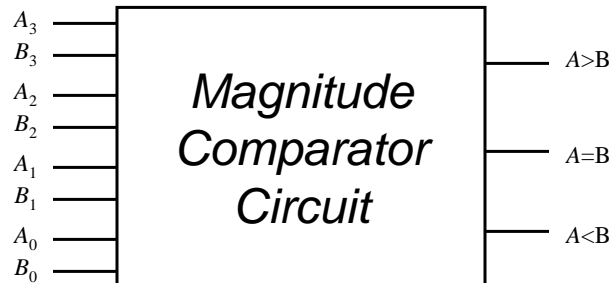
COMBINATIONAL BUILDING BLOCKS

- Many “Multi-gate Functions are Very Common
- These are Given Semi-standard Names
- Available in Many Design Tools
- Crucial for Modern Designer to Use without Going to the Gate Level
- Also Known as MSI Circuits

MAGNITUDE COMPARATOR

- Comparison of Two Binary Values
- Consider two n -bit Values A and B
- Need Circuit to Determine if:
 - $A > B$
 - $A = B$
 - $A < B$
- Consider 4-bit Values
 - A is $A_3A_2A_1A_0$
 - B is $B_3B_2B_1B_0$

MAGNITUDE COMPARATOR



Can Use the Equivalence Function (XNOR) to Determine $A=B$
 Define x_i to be the XNOR of Each Bit in A, B

$$x_i = A_i B_i + \bar{A}_i \bar{B}_i = \overline{A_i \oplus B_i}$$

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

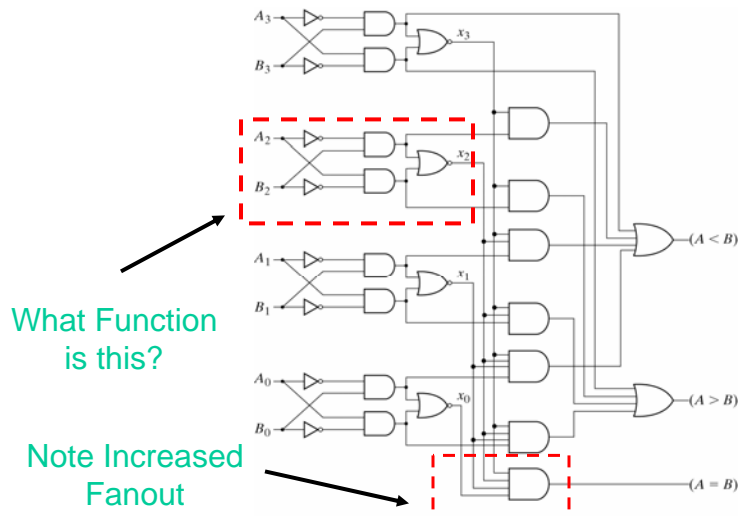
MAGNITUDE COMPARATOR

Can Build Truth Tables and use Maps to find
 the Following Relationships:

$$(A > B) = A_3 \bar{B}_3 + x_3 A_2 \bar{B}_2 + x_3 x_2 A_1 \bar{B}_1 + x_3 x_2 x_1 A_0 \bar{B}_0$$

$$(A < B) = \bar{A}_3 B_3 + x_3 \bar{A}_2 B_2 + x_3 x_2 \bar{A}_1 B_1 + x_3 x_2 x_1 \bar{A}_0 B_0$$

MAGNITUDE COMPARATOR



What Function is this?

Note Increased Fanout

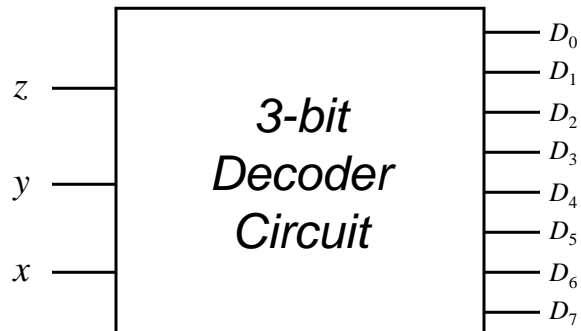
© 2002 Prentice Hall, Inc.
M. Morris Mano
DIGITAL DESIGN, 3e.

Fig. 4-17 4-Bit Magnitude Comparator

DECODER

Given a Binary Value of n -bits Indicate Which One is Input

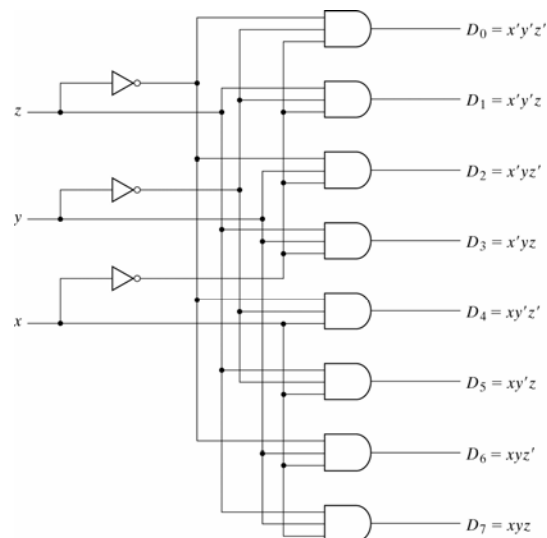
3-bit Example:



DECODER TRUTH TABLE

<u>Inputs</u>			<u>Outputs</u>							
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

DECODER CIRCUIT



© 2002 Prentice Hall, Inc.
M. Morris Mano
DIGITAL DESIGN, 3e.

Fig. 4-18 3-to-8-Line Decoder

DECODER CIRCUIT WITH ENABLE INPUT

- Decoder with Enable Line can be Viewed as a Demultiplexer
- More About Multiplexers to Come

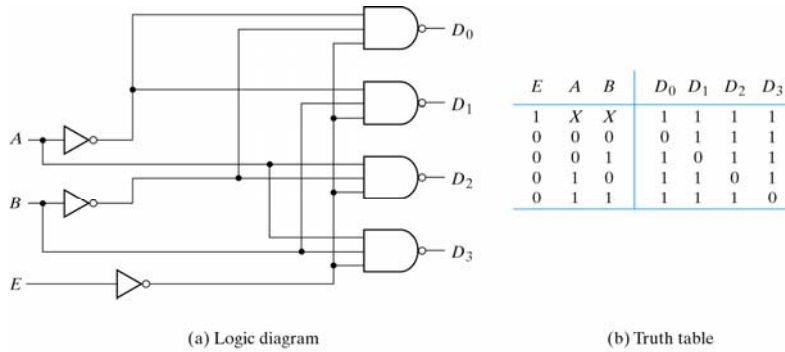
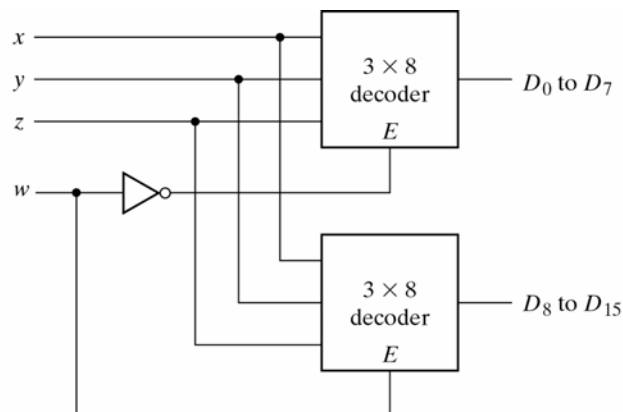


Fig. 4-19 2-to-4-Line Decoder with Enable Input

© 2002 Prentice Hall, Inc.
M. Morris Mano
DIGITAL DESIGN, 3e.

BUILDING LARGER DECODERS

This is a 4×16 Decoder Circuit



© 2002 Prentice Hall, Inc. Fig. 4-20 4 × 16 Decoder Constructed with Two 3 × 8 Decoders
M. Morris Mano
DIGITAL DESIGN, 3e.

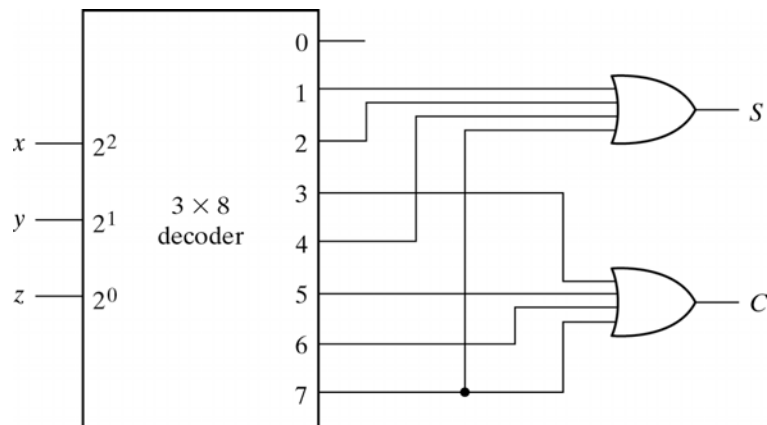
DECODERS FOR ANY COMB. CIRCUIT

- Decoder can be Used to Provide Output for all Minterms
- Use OR-Gates as Outputs from Decoder
- Consider Full-Adder Circuit:

$$S(x, y, z) = \sum(1, 2, 4, 7)$$

$$C(x, y, z) = \sum(3, 5, 6, 7)$$

DECODER FULL-ADDER CIRCUIT



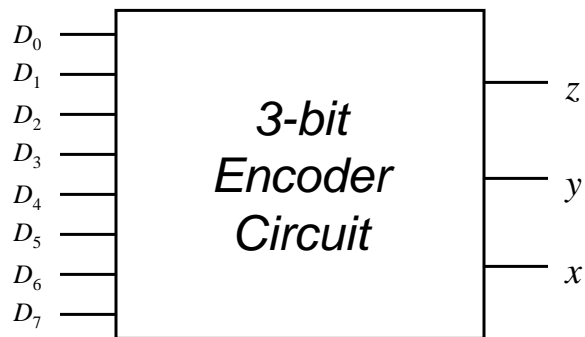
© 2002 Prentice Hall, Inc.
M. Morris Mano
DIGITAL DESIGN, 3e.

Fig. 4-21 Implementation of a Full Adder with a Decoder

ENCODER

Inverse (opposite) Function of the Decoder

3-bit Example:



ENCODER TRUTH TABLE

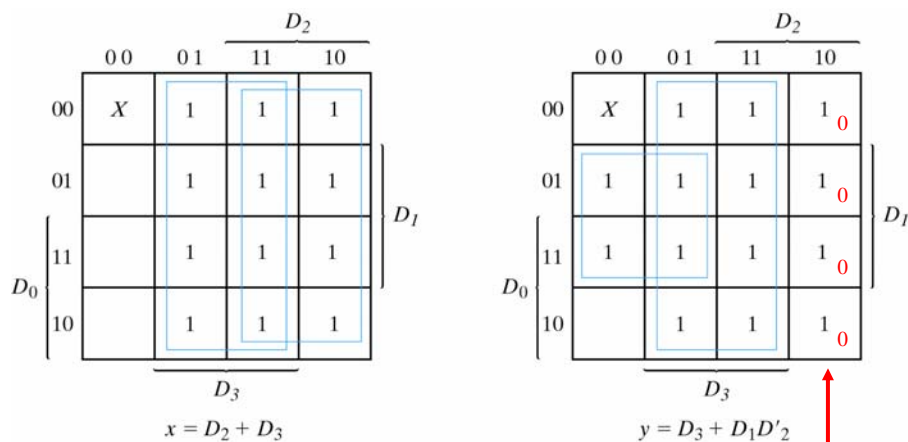
<u>Inputs</u>								<u>Outputs</u>		
D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	x	y	z
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

PRIORITY ENCODER TRUTH TABLE

<u>Inputs</u>				<u>Outputs</u>		
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

- V is the Valid bit

PRIORITY ENCODER MAPS



© 2002 Prentice Hall, Inc.
M. Morris Mano
DIGITAL DESIGN, 3e.

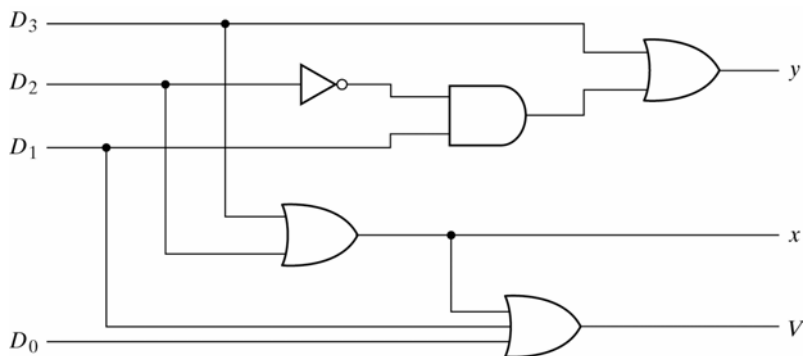
Fig. 4-22 Maps for a Priority Encoder

PRIORITY ENCODER CIRCUIT

$$x = D_2 + D_3$$

$$y = D_3 + D_1\bar{D}_2$$

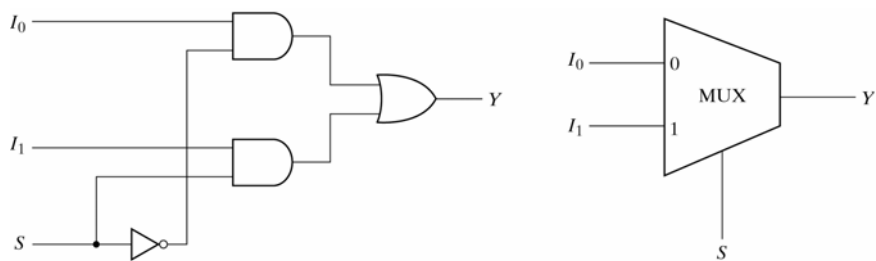
$$V = D_0 + D_1 + D_2 + D_3$$



© 2002 Prentice Hall, Inc.
M. Morris Mano
DIGITAL DESIGN, 3e.

Fig. 4-23 4-Input Priority Encoder

2-to-1 MULTIPLEXER CIRCUIT



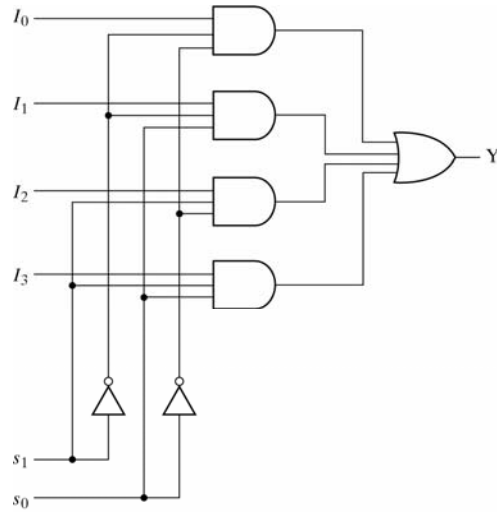
(a) Logic diagram

(b) Block diagram

Fig. 4-24 2-to-1-Line Multiplexer

© 2002 Prentice Hall, Inc.
M. Morris Mano
DIGITAL DESIGN, 3e.

4-to-1 MULTIPLEXER CIRCUIT



s_1	s_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

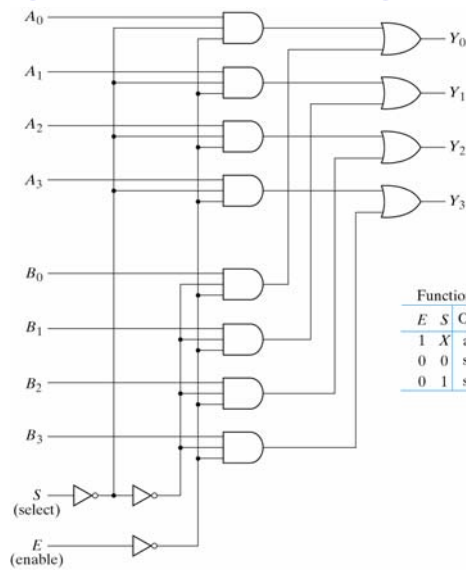
(b) Function table

© 2002 Prentice Hall, Inc.
M. Morris Mano
DIGITAL DESIGN, 3e.

(a) Logic diagram

Fig. 4-25 4-to-1-Line Multiplexer

QUADRUPLE 2-to-1 MULTIPLEXER



Function table		
E	S	Output Y
1	X	all 0's
0	0	select A
0	1	select B

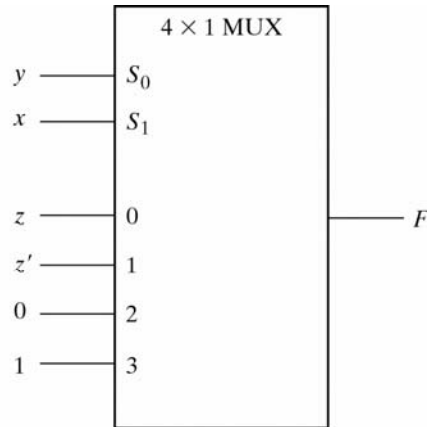
© 2002 Prentice Hall, Inc.
M. Morris Mano
DIGITAL DESIGN, 3e.

Fig. 4-26 Quadruple 2-to-1-Line Multiplexer

BOOLEAN FUNCTION IMPLEMENTED WITH MULTIPLEXERS

x	y	z	F
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

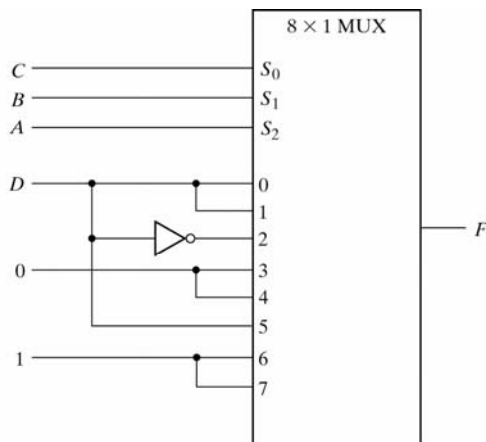
(a) Truth table



© 2002 Prentice Hall, Inc. Fig. 4-27 Implementing a Boolean Function with a Multiplexer
M. Morris Mano
DIGITAL DESIGN, 3e.

BOOLEAN FUNCTION IMPLEMENTED WITH MULTIPLEXERS

A	B	C	D	F
0	0	0	0	0
0	0	0	1	1
0	0	1	0	0
0	0	1	1	1
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1



© 2002 Prentice Hall, Inc. Fig. 4-28 Implementing a 4-Input Function with a Multiplexer
M. Morris Mano
DIGITAL DESIGN, 3e.

Tri-State Buffer

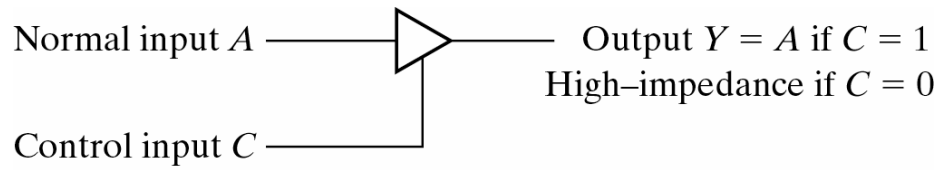


Fig. 4-29 Graphic Symbol for a Three-State Buffer

© 2002 Prentice Hall, Inc.
 M. Morris Mano
 DIGITAL DESIGN, 3e.

MULTIPLEXER IMPLEMENTATIONS

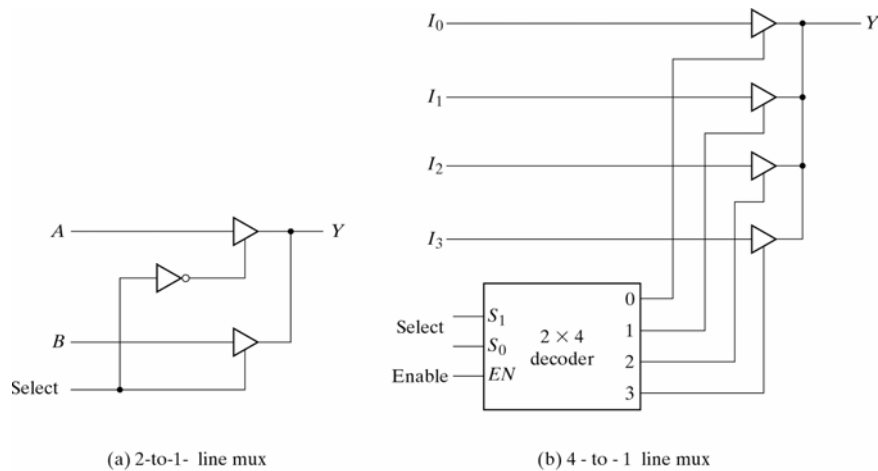
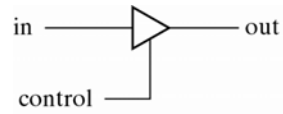


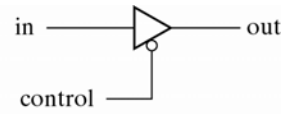
Fig. 4-30 Multiplexers with Three-State Gates

© 2002 Prentice Hall, Inc.
 M. Morris Mano
 DIGITAL DESIGN, 3e.

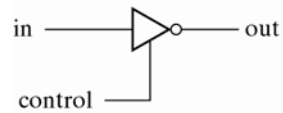
VARIOUS TRI-STATE BUFFERS



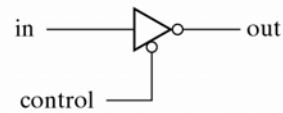
bufif1



bufif0



notif1



notif0

© 2002 Prentice Hall, Inc.
M. Morris Mano
DIGITAL DESIGN, 3e.

Fig. 4-31 Three-State Gates