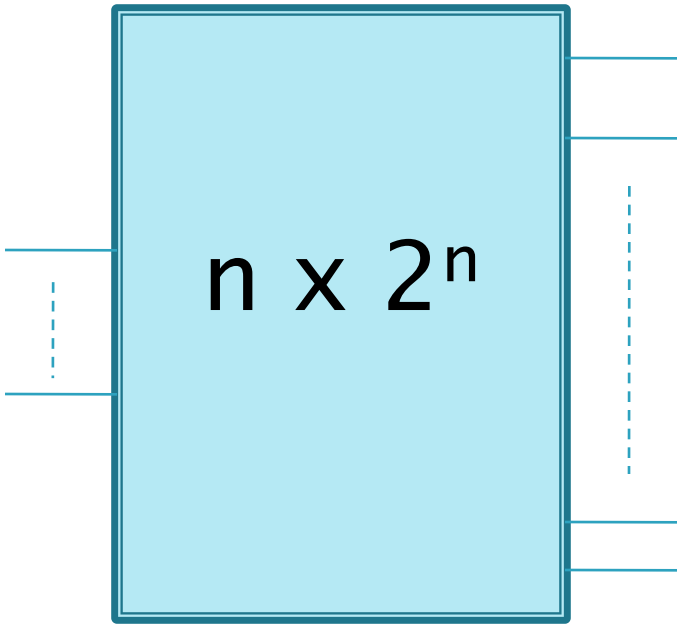
Digital Design

Lecture of week 7

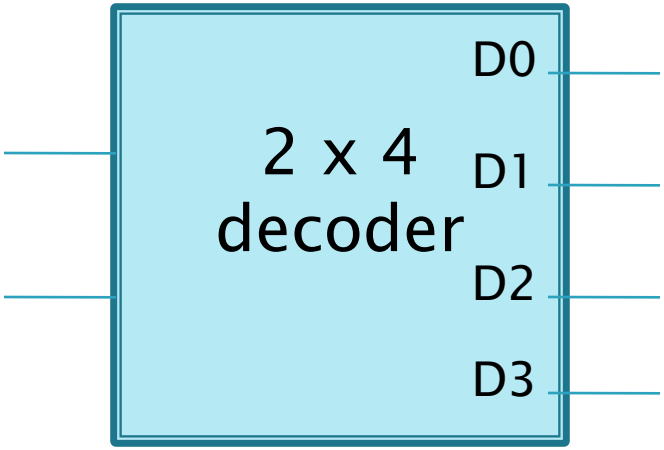
Dr Manal Tantawi

Decoders



Decoders

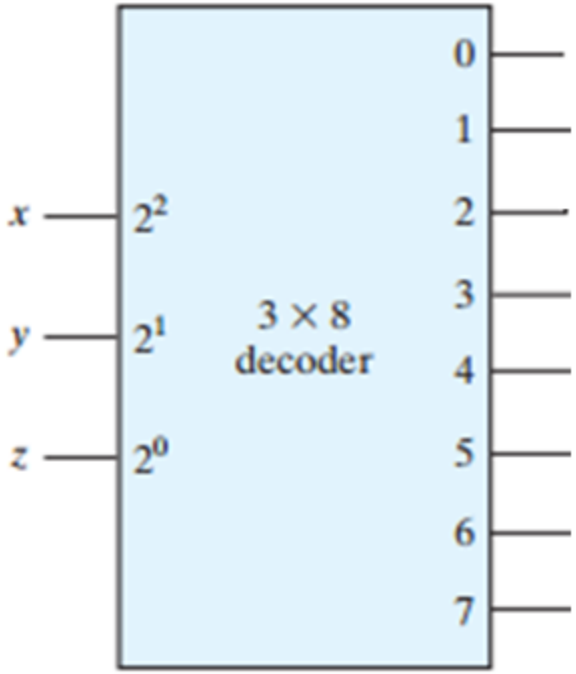
- 2 x 4 Decoder



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

Decoders

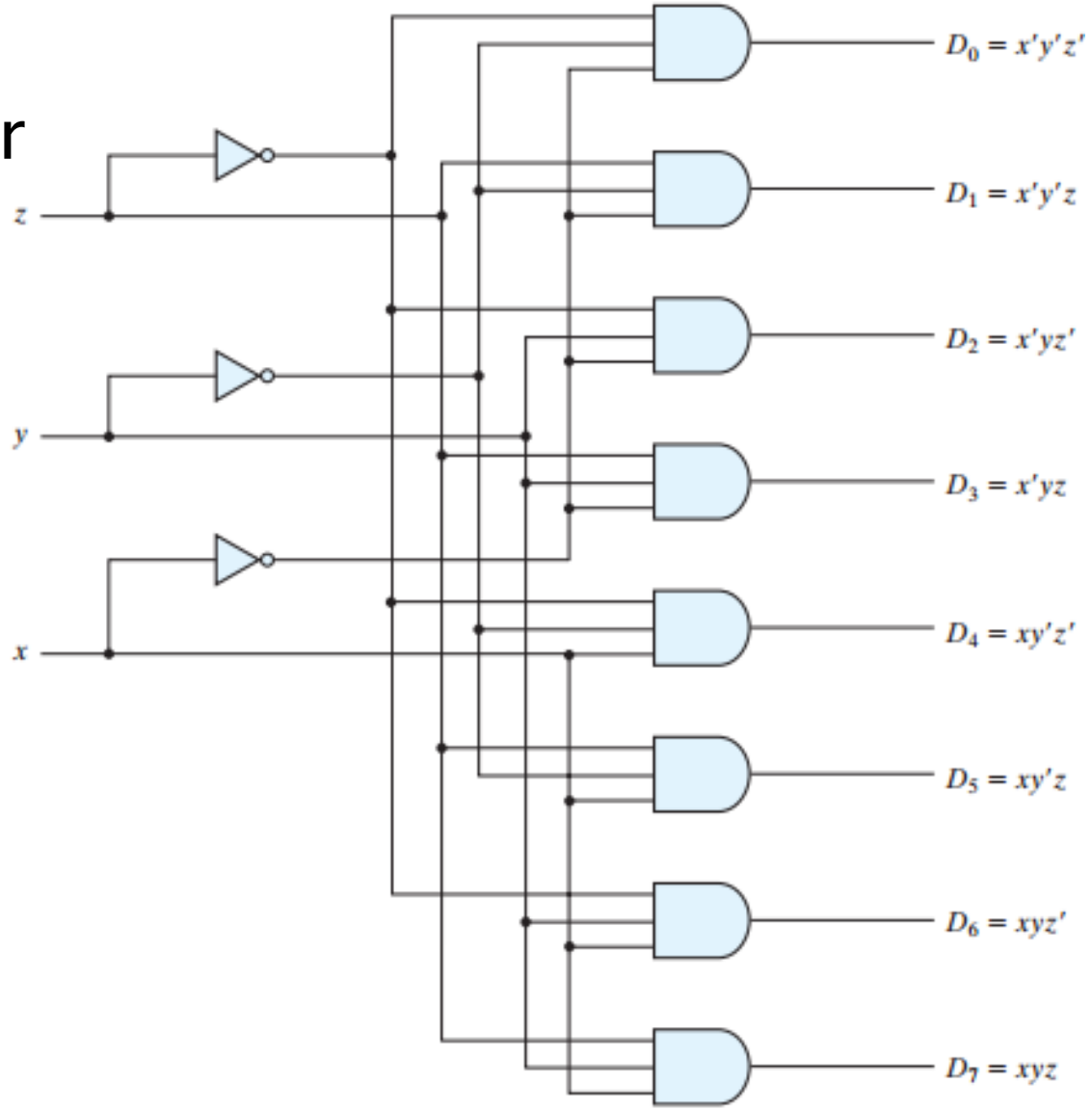
- 3 x 8 Decoder



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

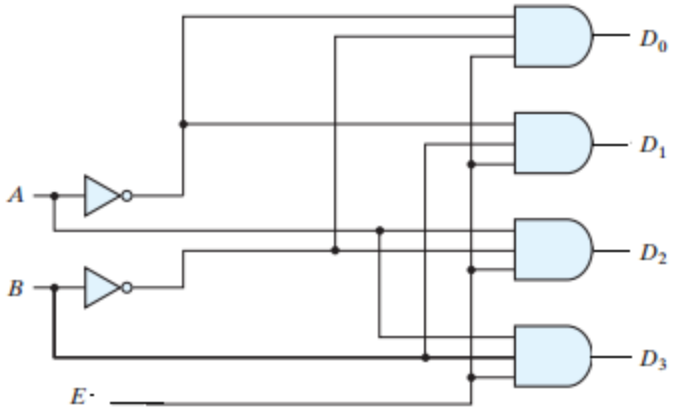
Decoders

- 3 x 8 Decoder



Decoders

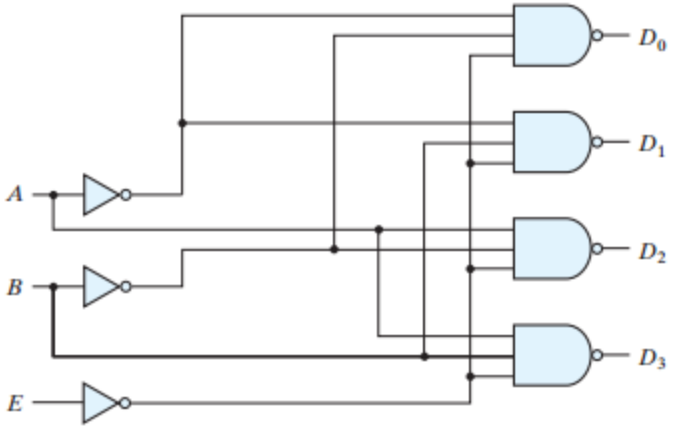
Active high decoder



A	B	D_0	D_1	D_2	D_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

E = 1 enabled
E = 0 disabled

Active Low Decoder (NAND decoder)



A	B	D_0	D_1	D_2	D_3
0	0	0	1	1	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	1	0

E = 0 enabled
E = 1 disabled

Implementing Functions using Decoders

- Design procedure

1) Define of inputs and outputs
2) Derive truth table
3) define suitable decoder
4) Implement each output using this decoder

Implementing Functions using Decoders

Design a full adder using suitable decoder

1) Number of inputs $=3$ number of outputs $= 2$

2) Truth table

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

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

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

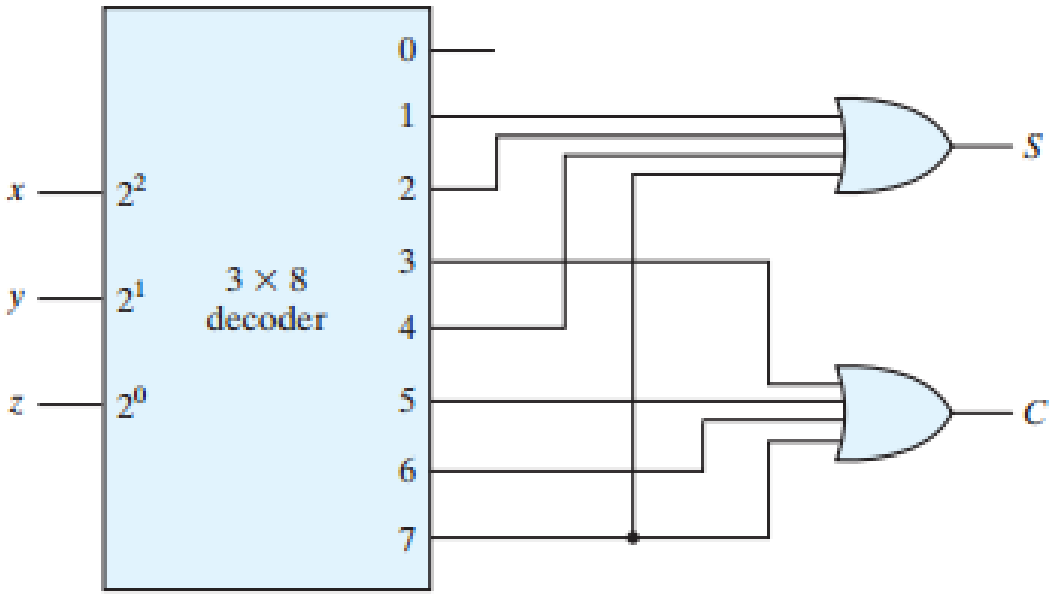
Implementing Functions using Decoders

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

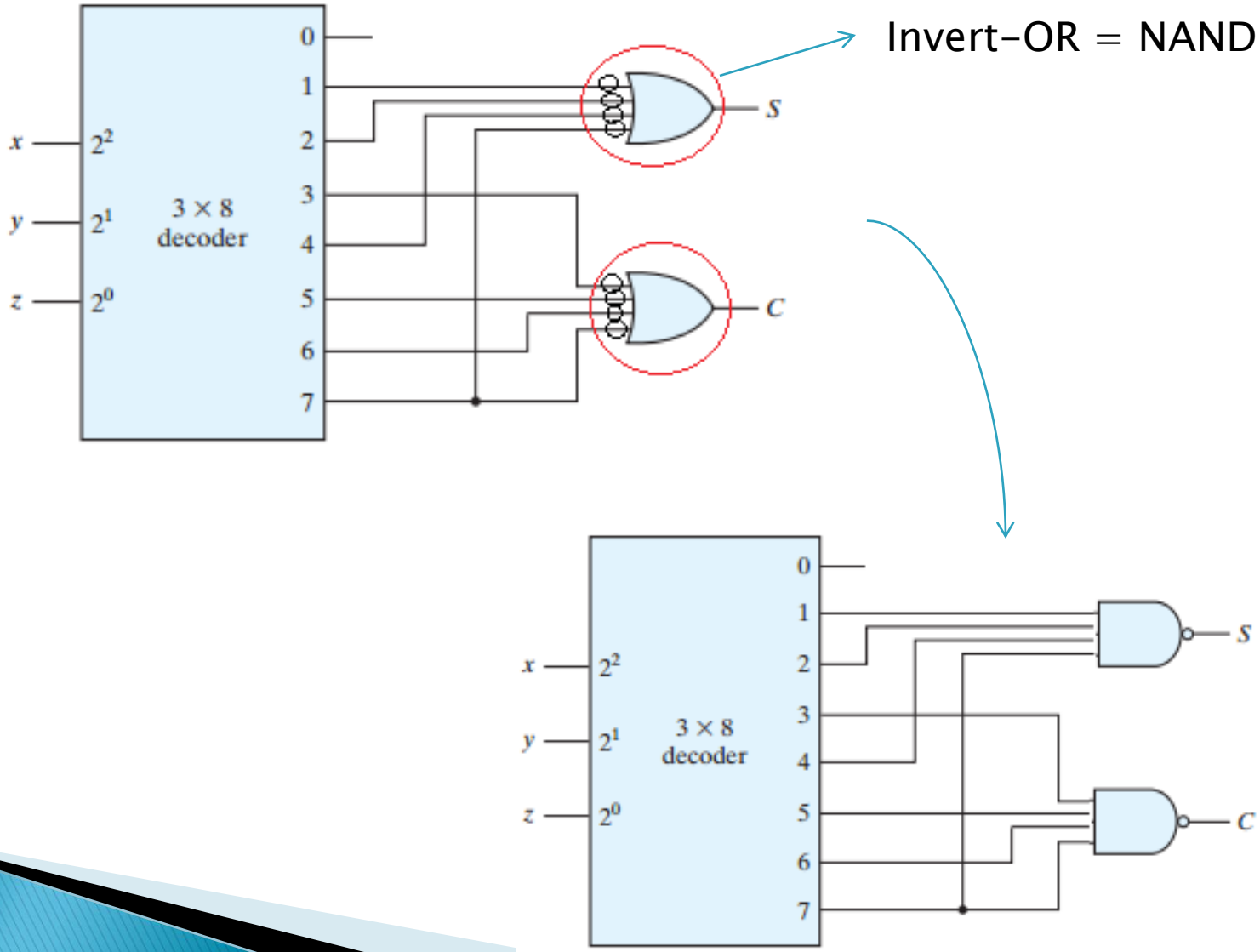
$$C(x, y, z) = \Sigma(3, 5, 6, 7)$$

3) Suitable decoder : 3 x 8 decoder

4)

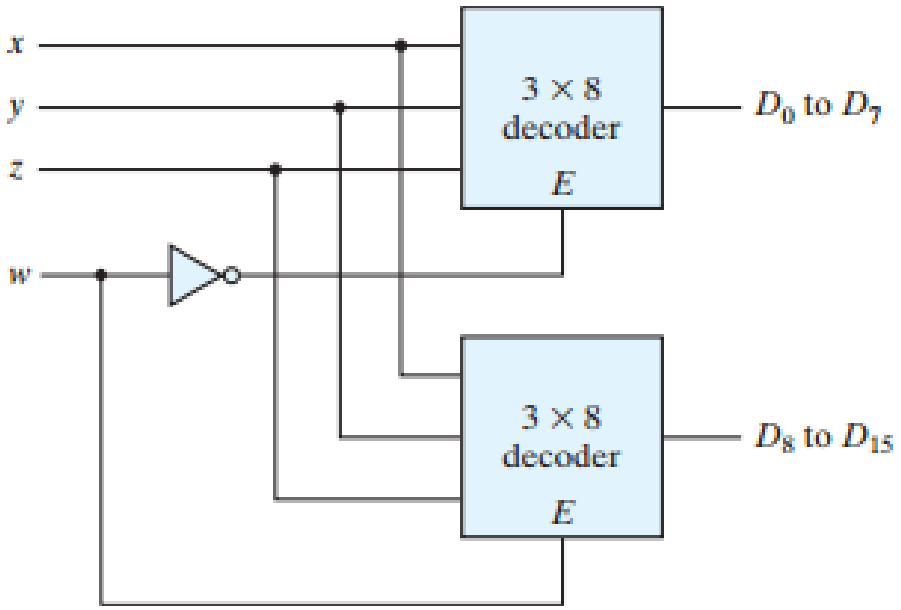


Using active low decoder



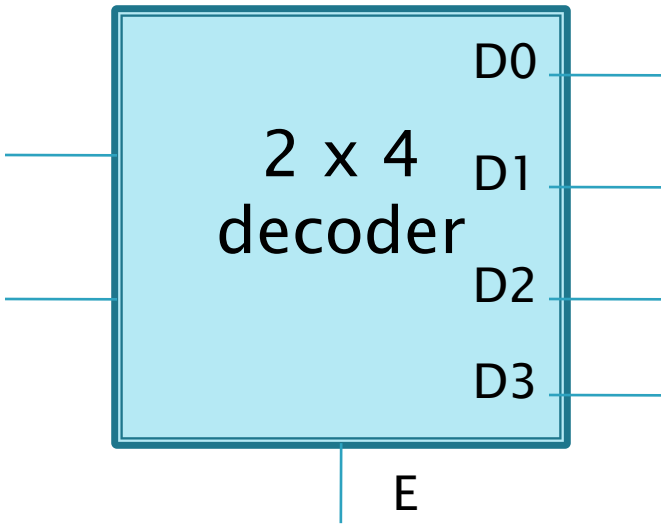
Construct 4x16 decoder using 3x8 decoders

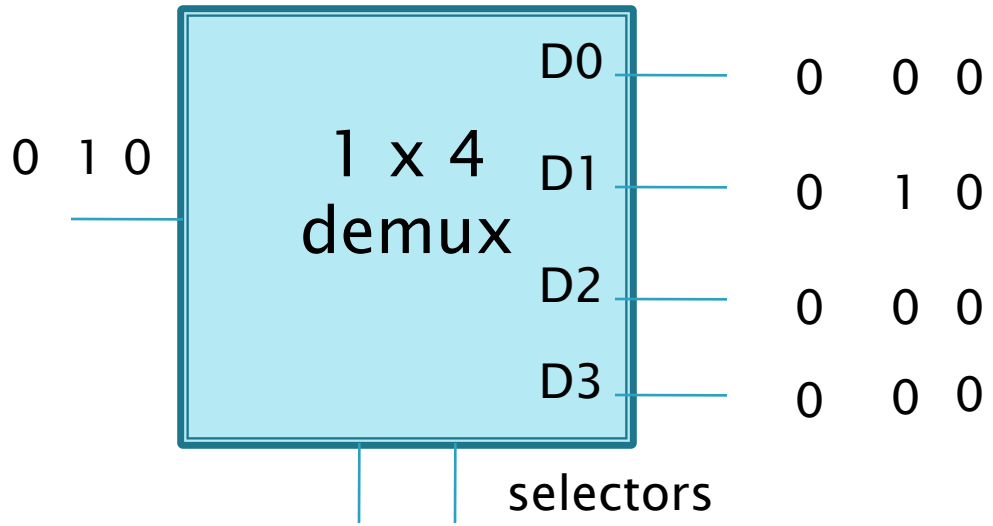
W	X	Y	Z
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	0	0
1	1	0	1
1	1	1	0
1	1	1	1

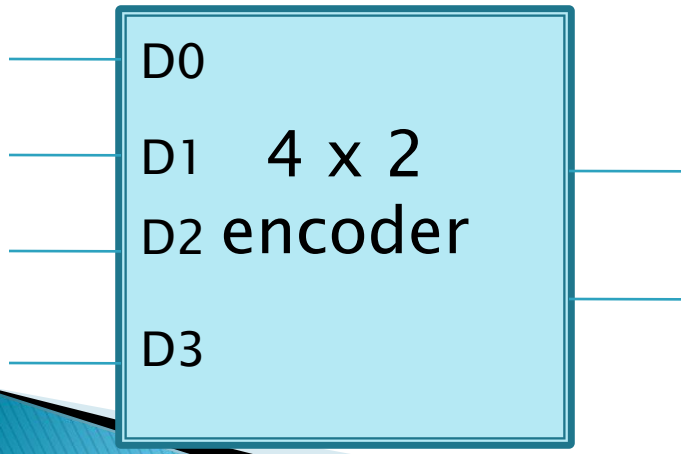


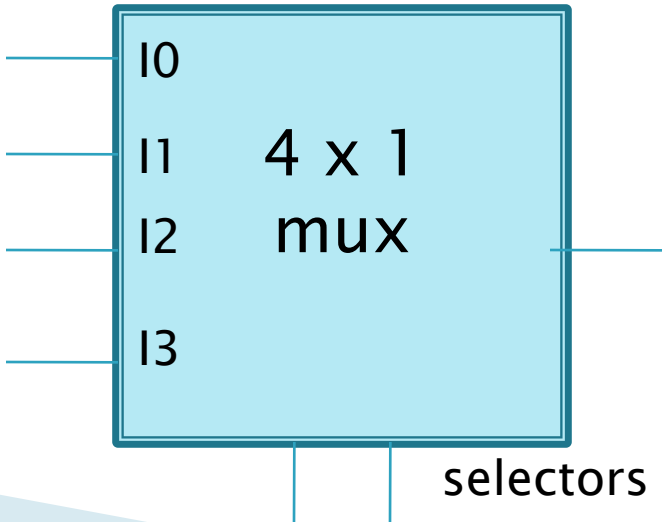
w	E0	E1
0	1	0
1	0	1

Decoders, Encoders, Demultiplexers (demux) and Multiplexers (mux)





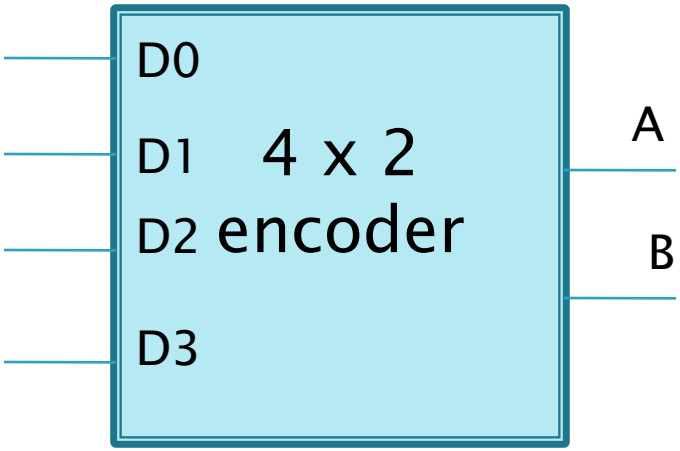




Encoder

- 4 x 2 encoder

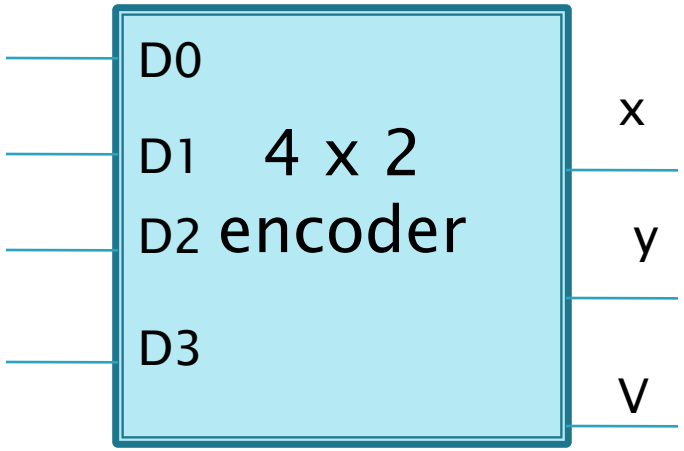
D_0	D_1	D_2	D_3	A	B
1	0	0	0	0	0
0	1	0	0	0	1
0	0	1	0	1	0
0	0	0	1	1	1



Encoder

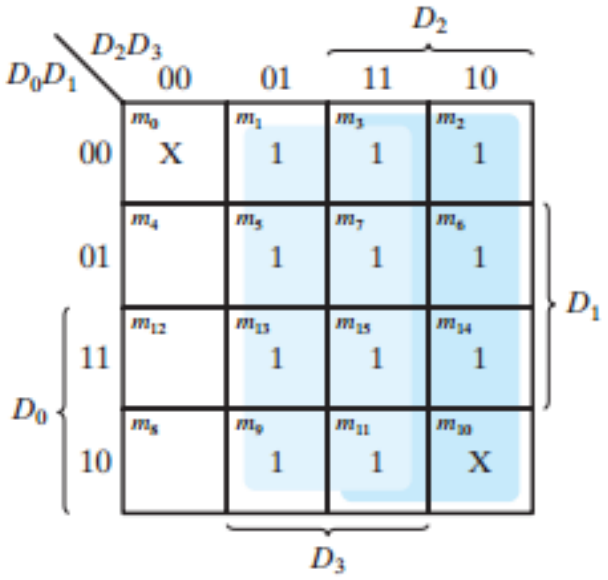
- 4 x 2 highest priority encoder

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

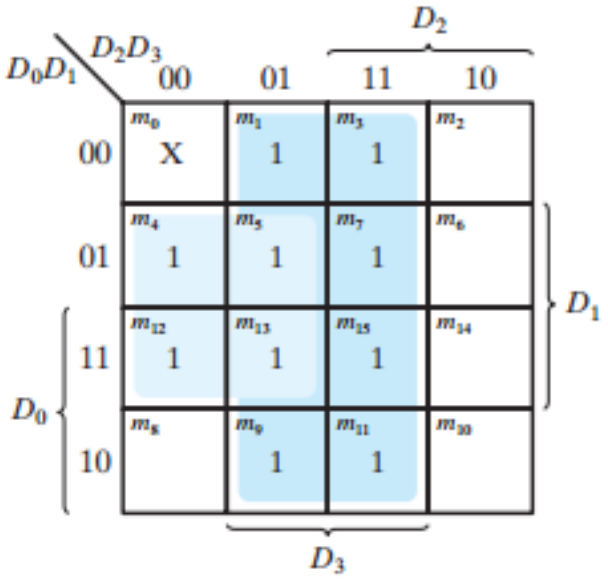


4 x 2 highest priority encoder

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

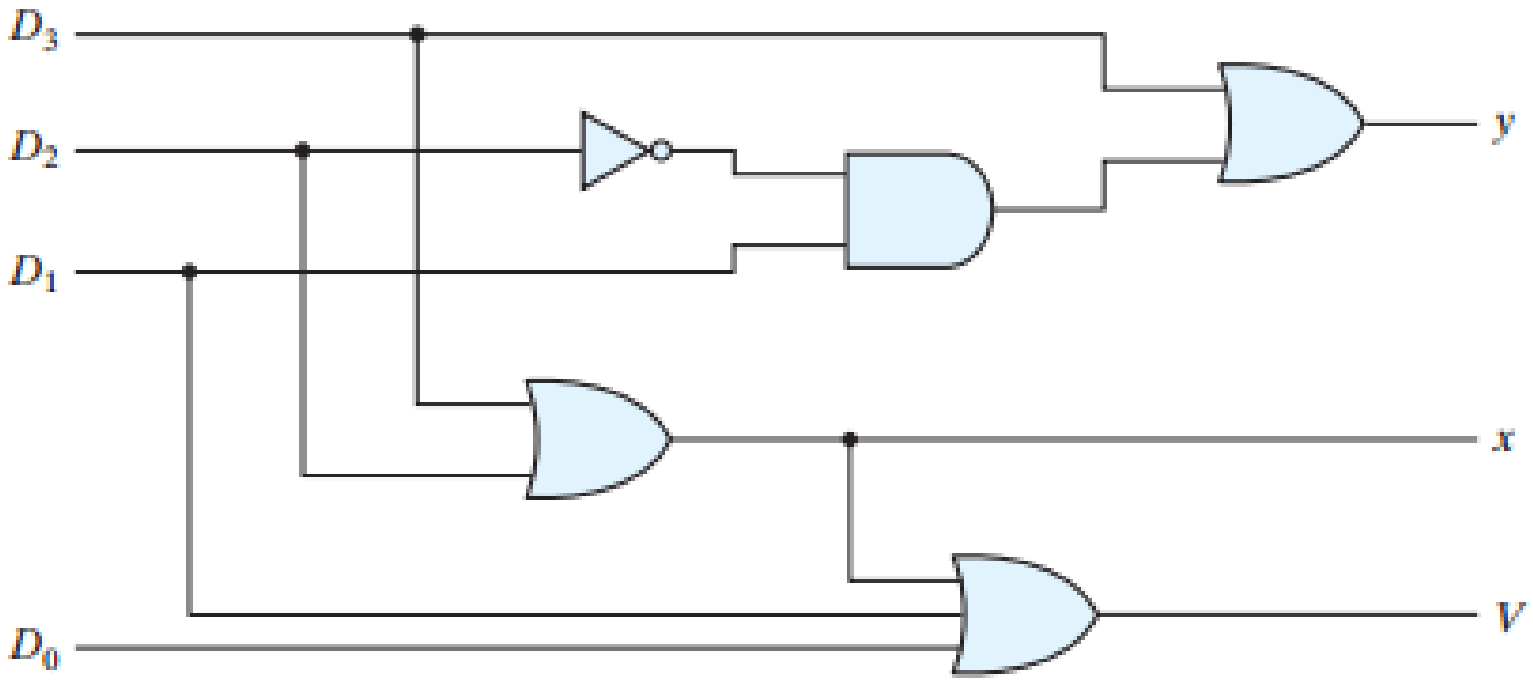


$x = D_2 + D_3$

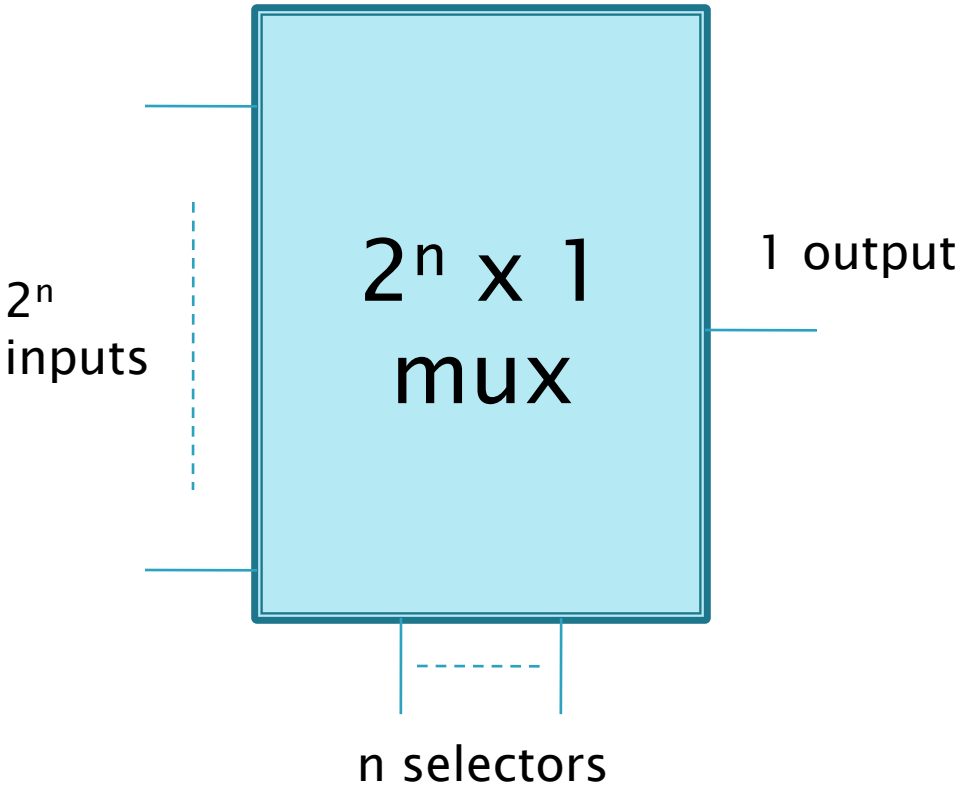


$y = D_3 + D_1D'_2$

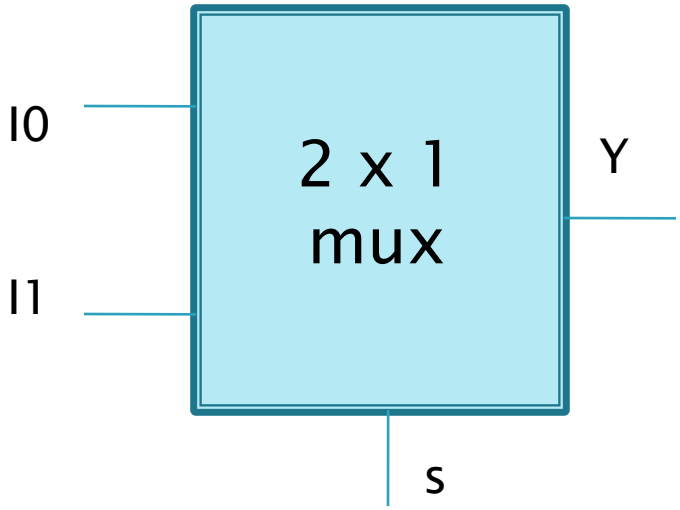
4 x 2 highest priority encoder

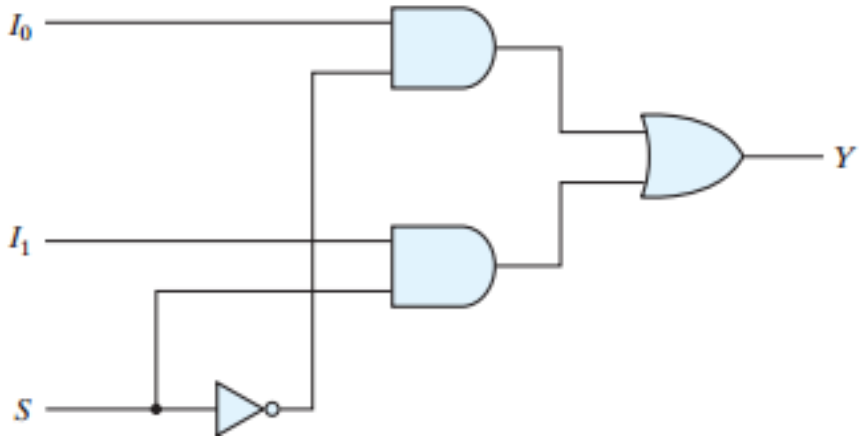


Multiplexers



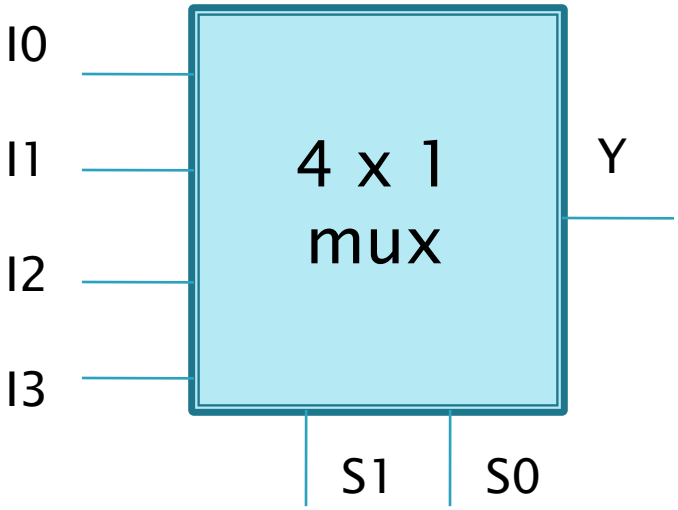
Multiplexers (2 x 1 mux)

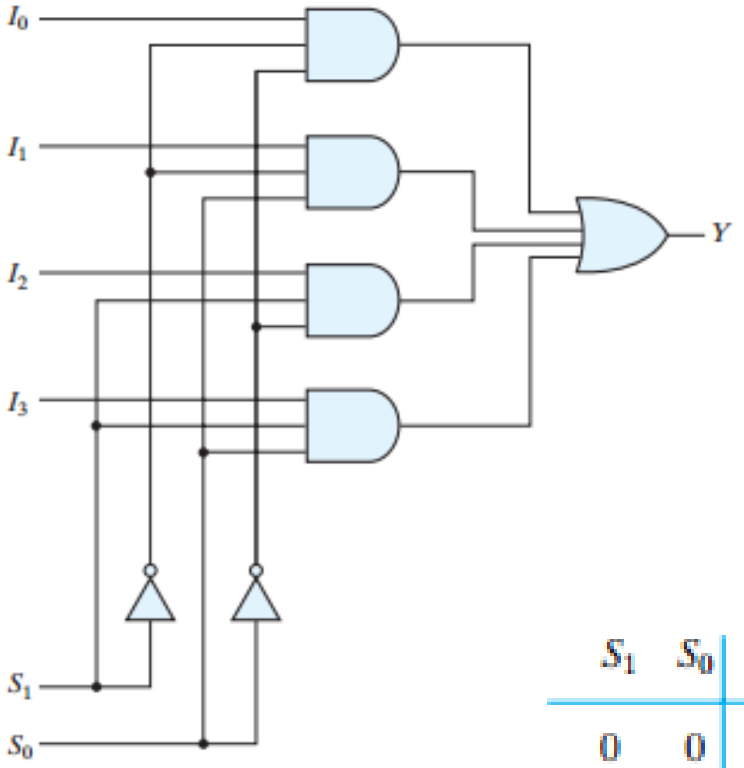




S	Y
0	I_0
1	I_1

Multiplexers (2 x 1 mux)

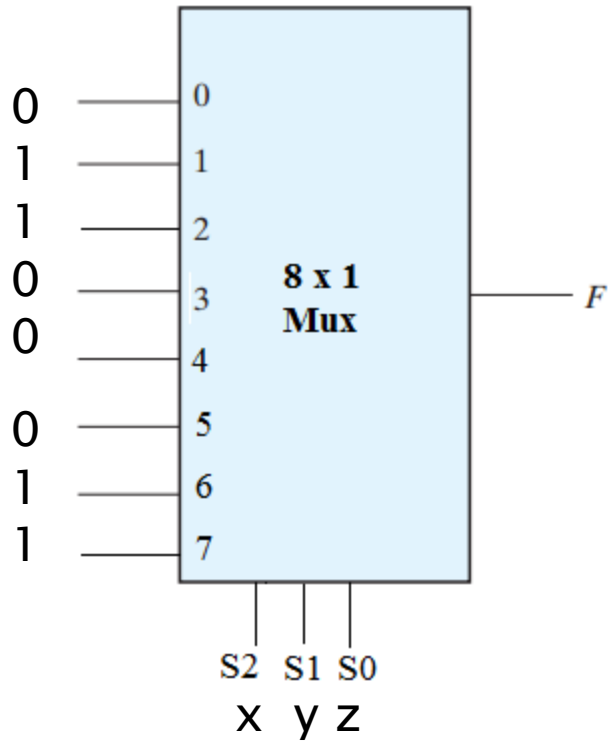




S_1	S_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

Implementing functions using mux

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

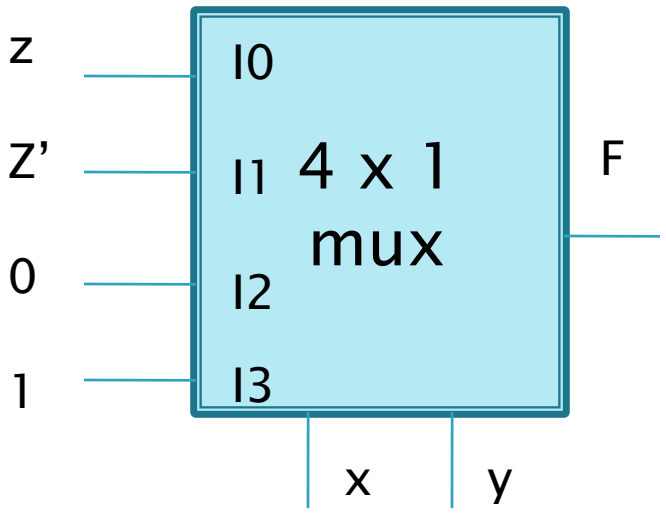


Number of selection lines equal number of function variables -1

Implementing functions using mux

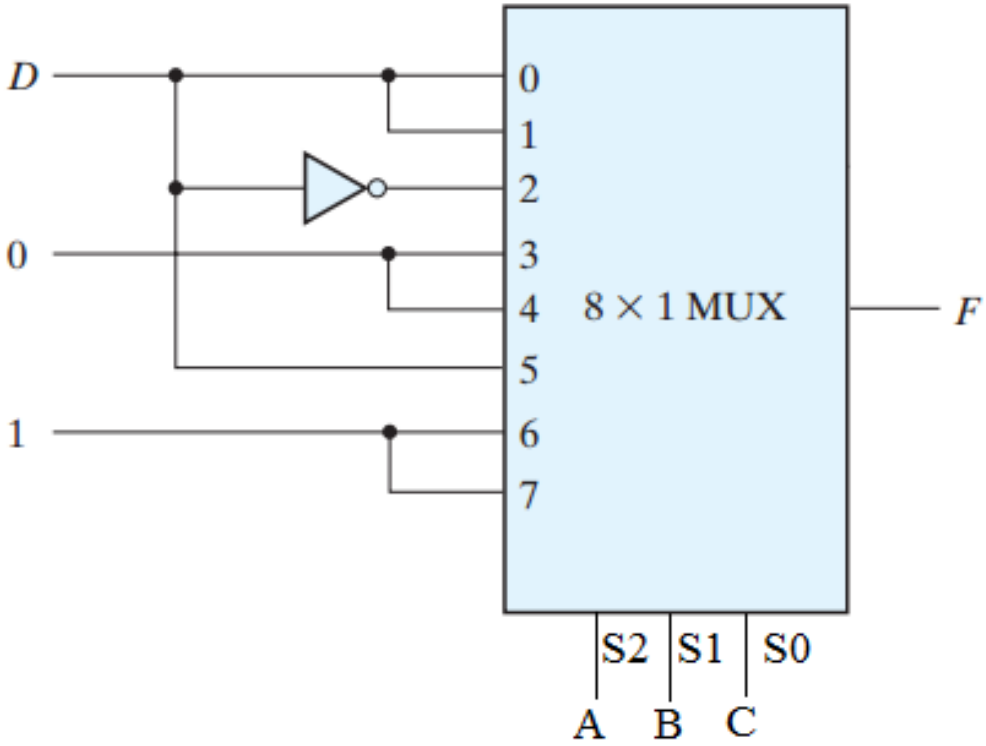
Number of selection lines equal number of function variables -1

x	y	z	F	
0	0	0	0	I0 = $F = z$
0	0	1	1	
0	1	0	1	I1 = $F = z'$
0	1	1	0	
1	0	0	0	I2 = $F = 0$
1	0	1	0	
1	1	0	1	I3 = $F = 1$
1	1	1	1	



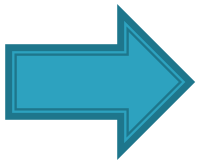
Implementing functions using mux

A	B	C	D	F	
0	0	0	0	0	$F = D$
0	0	0	1	1	
0	0	1	0	0	$F = D$
0	0	1	1	1	
0	1	0	0	1	$F = D'$
0	1	0	1	0	
0	1	1	0	0	$F = 0$
0	1	1	1	0	
1	0	0	0	0	$F = 0$
1	0	0	1	0	
1	0	1	0	0	$F = D$
1	0	1	1	1	
1	1	0	0	1	$F = 1$
1	1	0	1	1	
1	1	1	0	1	$F = 1$
1	1	1	1	1	



Implement F(A, B, C, D) = SUM (1, 3, 4, 11, 12, 13, 14, 15) using 8 x1 mux put A, B and D on selections

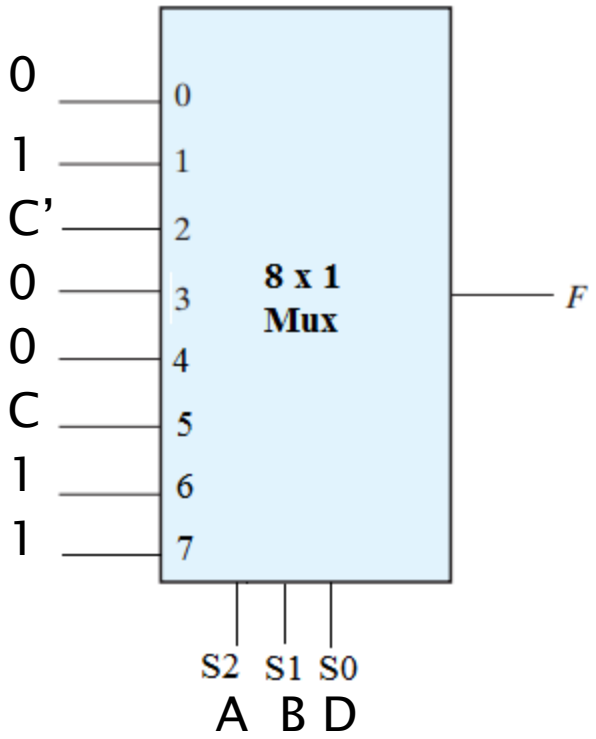
A	B	C	D	F
0	0	0	0	0
0	0	0	1	1
0	0	1	0	0
0	0	1	1	1
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1



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

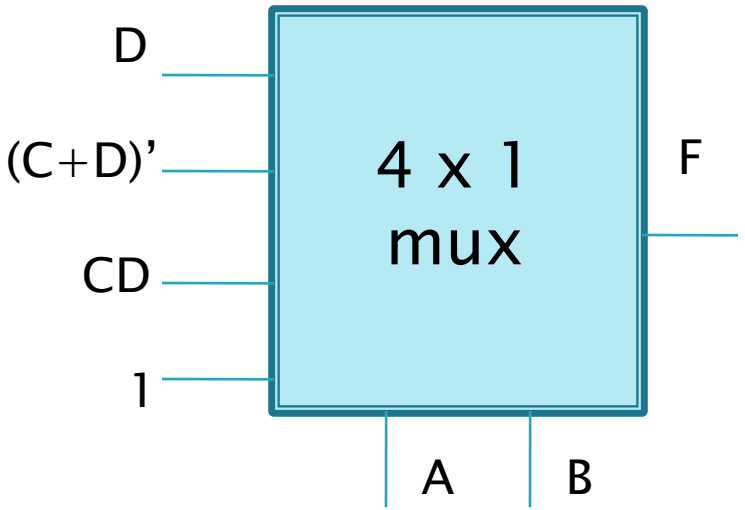
Implement F(A, B, C, D) = SUM (1, 3, 4, 11, 12, 13, 14, 15) using 8 x1 mux put A, B and D on selections

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

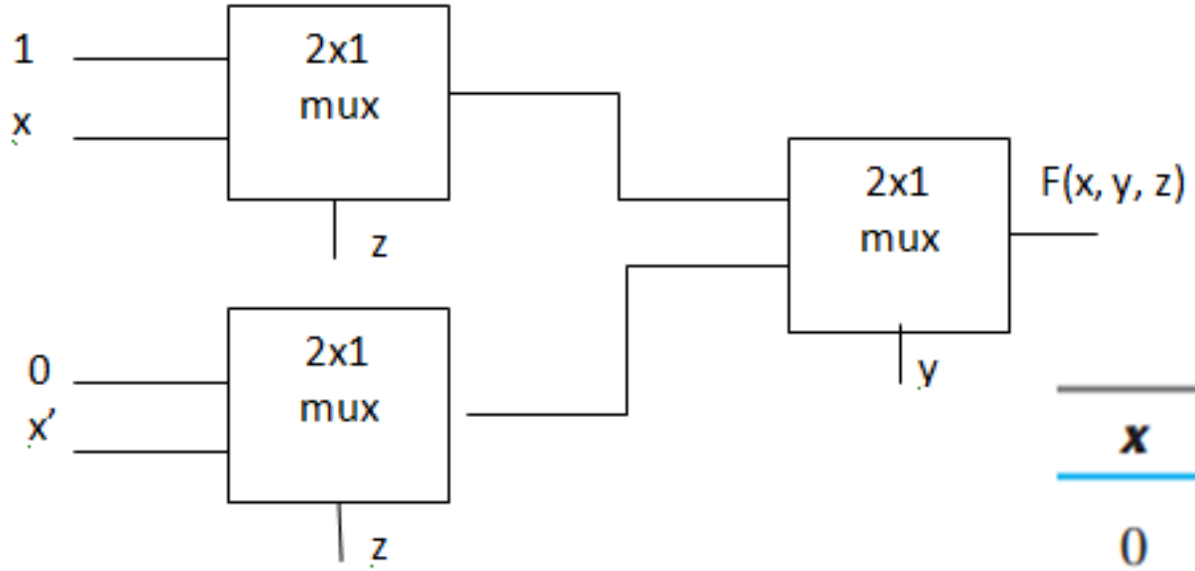


Implement F(A, B, C, D) = SUM (1, 3, 4, 11, 12, 13, 14, 15) using 4 x1 mux

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



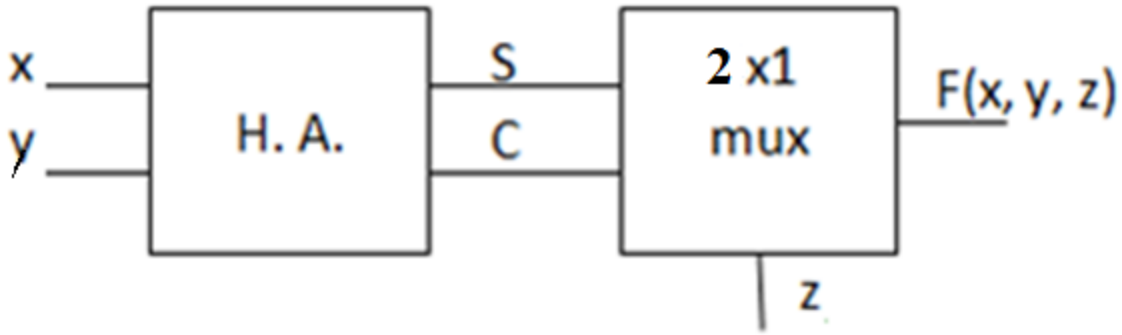
Analyze the following circuit find F



F(x, y, z) = SUM (0, 3, 4, 5)

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

Analyze the following circuit find F



x	y	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

F(x, y, z) = SUM (2, 4, 7)

x	y	z	F
0	0	0	0 (s)
0	0	1	0 (c)
0	1	0	1 (s)
0	1	1	0 (c)
1	0	0	1 (s)
1	0	1	0 (c)
1	1	0	0 (s)
1	1	1	1 (c)

Next Lecture we will begin chapter 5
thank you