UTM
UNIVERSITI TEKNOLOGI MALAYSIA

SKEE1223: Digital Electronics

8– Medium Scale Integrated (MSI) Circuits

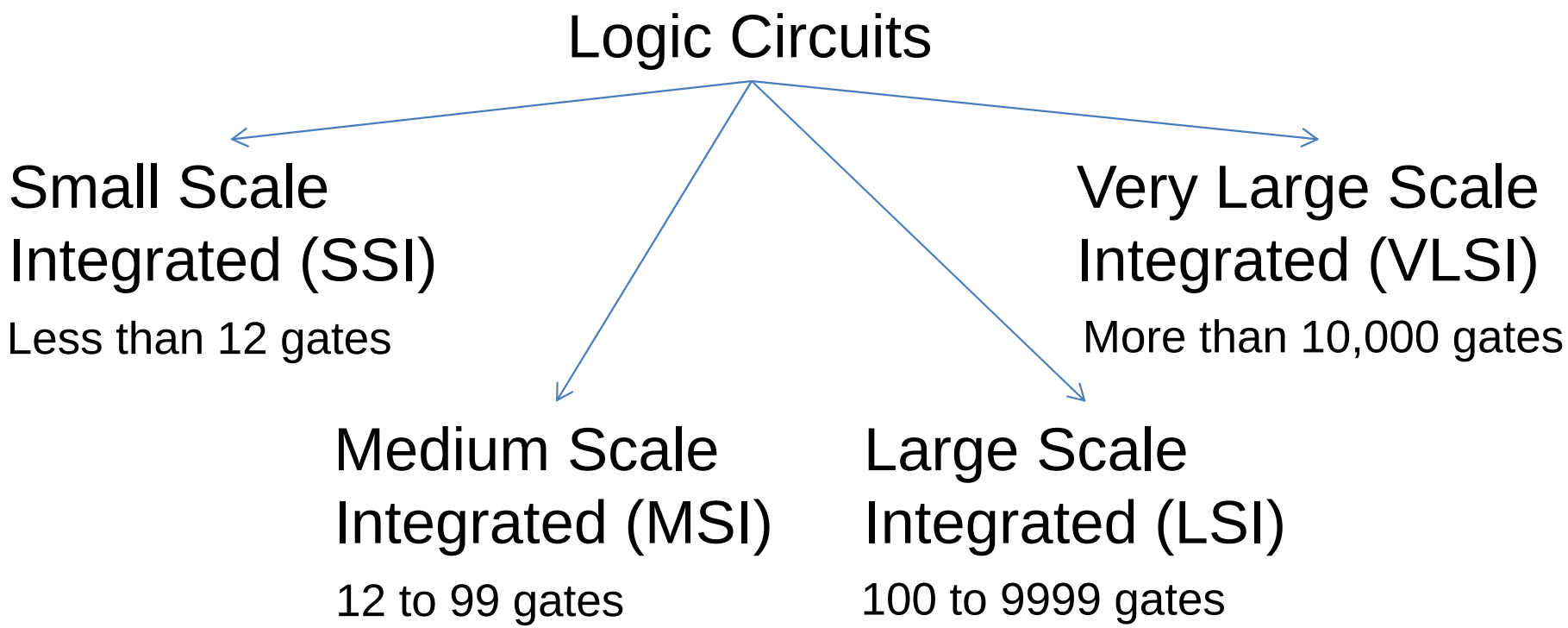
Dr Michael Tan Loong Peng
PhD (Cambridge)
Senior Lecturer
Faculty of Electrical Engineering
Universiti Teknologi Malaysia

MSI Circuits

- Multiplexers (Mux)
  - 2x1, 4x1, and 8x1 muxes
  - 74x151, 74x153, 74x157 devices
- Demultiplexers (Demux), Decoders, and Encoders
  - 74x138 and 74x139 decoders
  - Encoder, priority encoder and the 75x147 devices
  - BCD to 7-segment decoder and the 74x247 devices
  - Logic functions using muxes and decoders
- Adders and Comparators
  - Half, full and ripple carry adders
  - The 74x83 devices
  - Comparator and the 74x85 devices

MSI Circuits

- MSI (Medium Scale Integrated) circuits are logic circuits that contain 12 to 99 logic gates

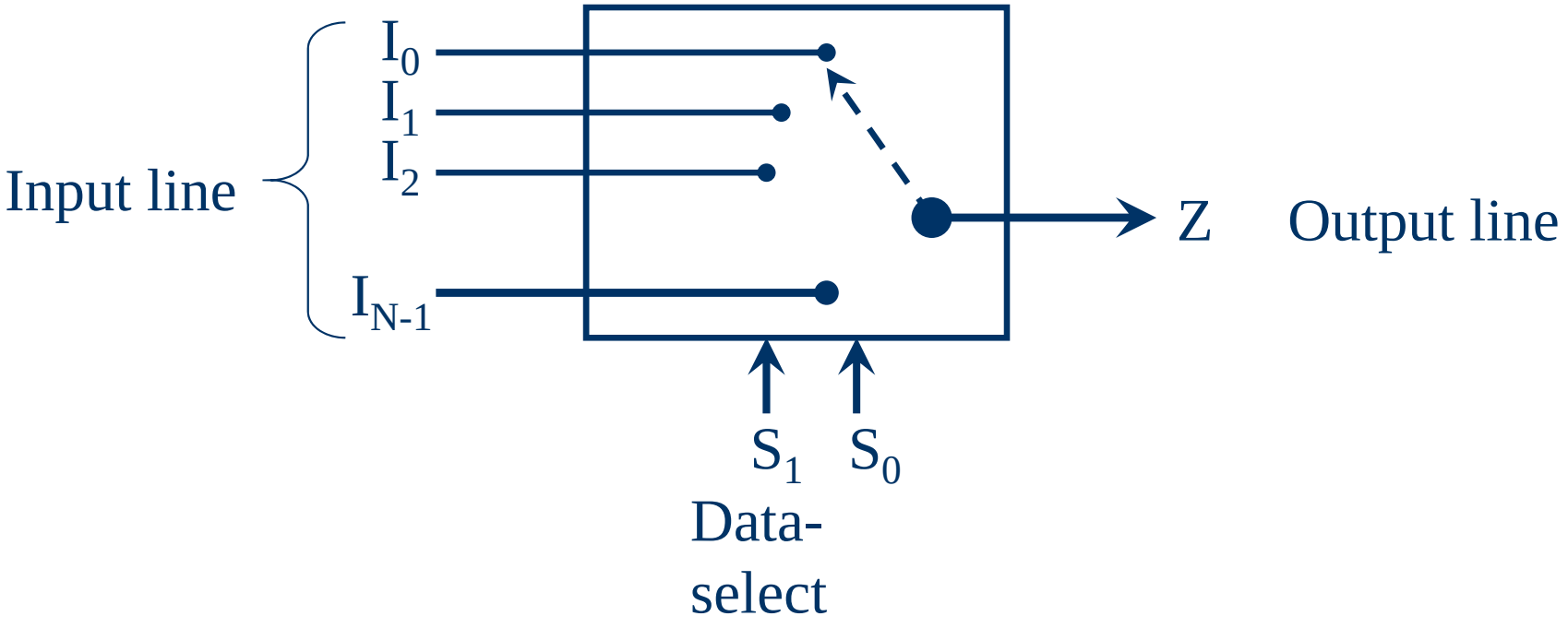


Multiplexers (Mux)

- A multiplexer (Mux) selects one data line from two or more input lines and routes data from the selected line to the output. The particular data line that is selected is determined by the select inputs.
- Mux is usually written as (Y)x1, where Y is the number of input data lines
- For example, a 4 input data line mux is written as 4x1 mux

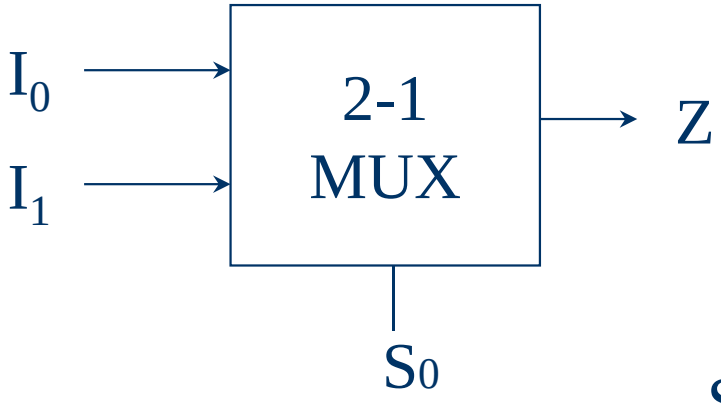
Multiplexer (Data Selector)

- Circuits with many inputs but only ONE output.
- At any time, only date of a particular input can be sent to the output.
- This is done using the control or SELECT bits (also called data selector).

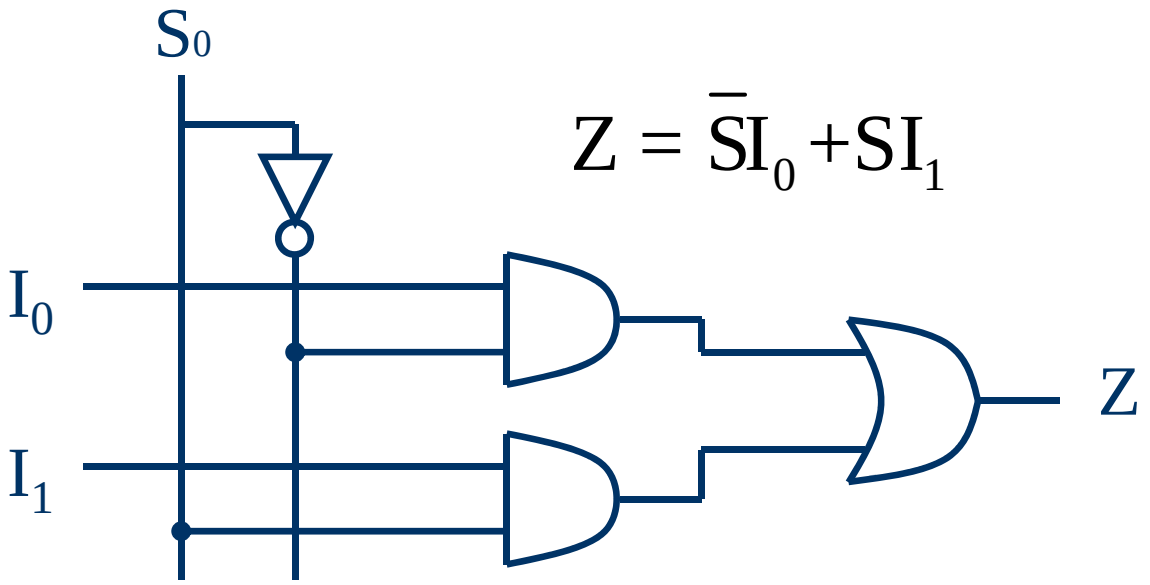


Multiplexer

- 2-input Multiplexer (2-to-1 MUX)

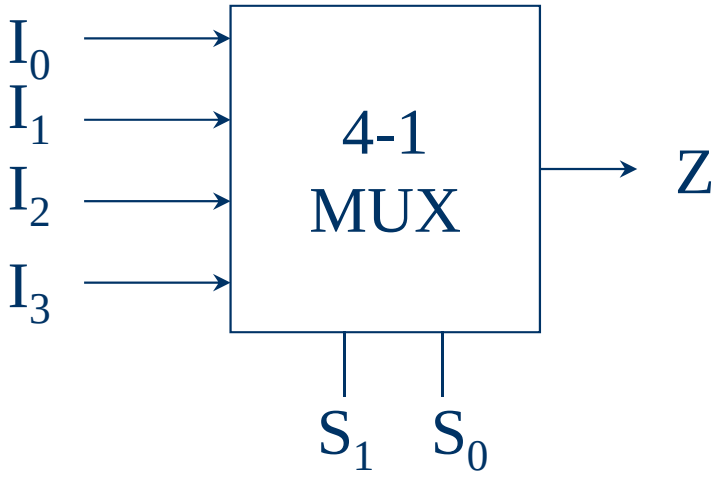


S	Output (Z)
0	$Z = I_0$
1	$Z = I_1$



Multiplexer

- 4-input Multiplexer (4-to-1 MUX)

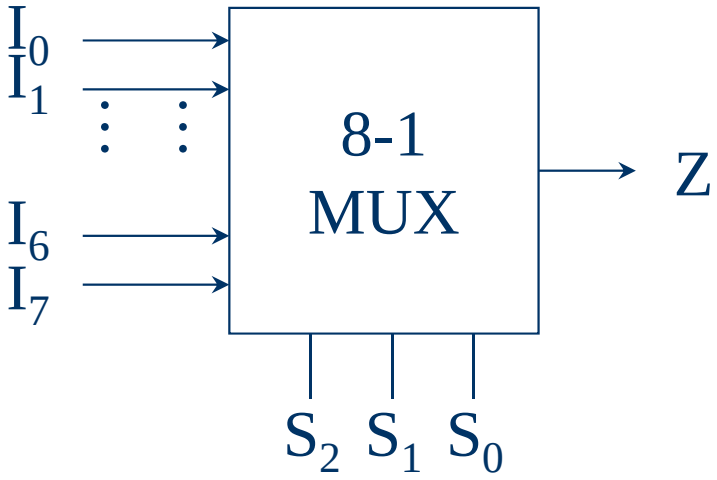


S_1S_0	Output (Z)
0 0	I_0
0 1	I_1
1 0	I_2
1 1	I_3

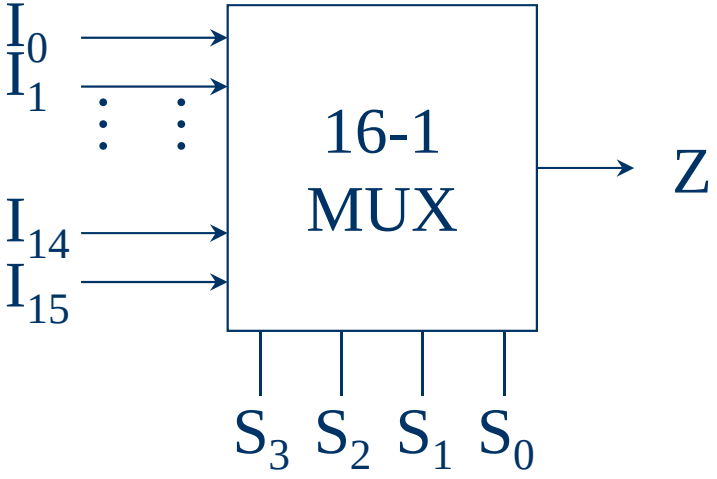
$$Z = \overline{S}_1\overline{S}_0I_0 + \overline{S}_1S_0I_1 + S_1\overline{S}_0I_2 + S_1S_0I_3$$

Multiplexer

- Design 8-input Multiplexer (8-to-1 MUX)

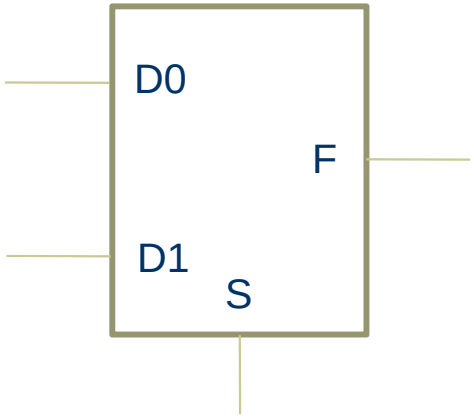


- Design 16-input Multiplexer (16-to-1 MUX)



Mux (cont.)

- 2x1 Mux (2 input data and 1 output data)



Symbol for 2x1 mux

Function:

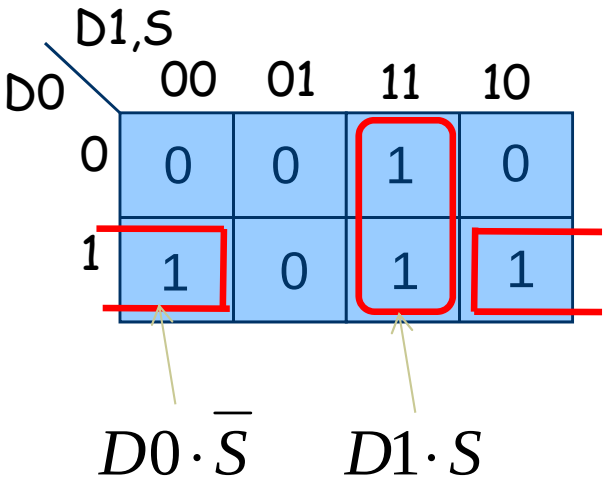
F = D0 when S = 0
F = D1 when S = 1

-How to design a 2x1 mux using basic gates?
- Using K-Map? What are the inputs and outputs?
- By inspection on its function?

Mux (cont.)

- Design of 2x1 mux using K-Map
  - Inputs: D0, D1, S
  - Output: F

D0	D1	S	F
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1



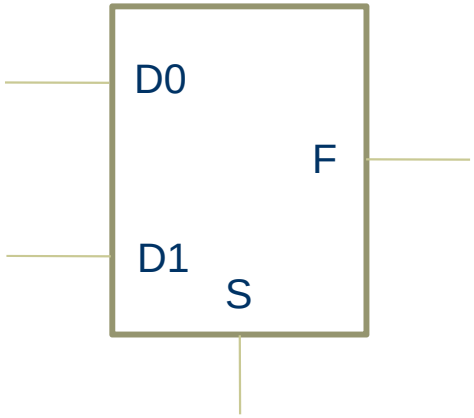
$D0 \cdot \overline{S}$ $D1 \cdot S$

$$F = D0 \cdot \overline{S} + D1 \cdot S$$

Can you draw the logic circuit?

Mux (cont.)

- Design of 2x1 mux by its function



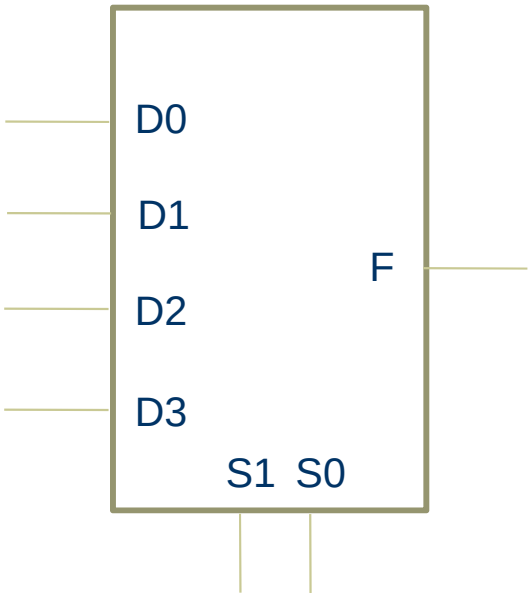
F = D0 when *S = 0* $\longrightarrow F = D0 \cdot \overline{S}$

F = D1 when *S = 1* $\longrightarrow F = D1 \cdot S$

Therefore, $F = D0 \cdot \overline{S} + D1 \cdot S$

Mux (cont.)

- 4x1 Mux



Function:

F = D0 when S1 = 0 and S0 = 0
F = D1 when S1 = 0 and S0 = 1
F = D2 when S1 = 1 and S0 = 0
F = D3 when S1 = 1 and S0 = 1

How can we design the 4x1 mux using basic gates?

- Using K-Maps? What are the outputs and inputs?
- By its function?

Mux (cont.)

- The 4x1 Mux has six inputs (D3, D2, D1, D0, S1, S0) and one output (F), therefore it is difficult/time consuming to use K-Maps

Looking at the function of the 4x1 mux,

F = D0 when S1 = 0 and S0 = 0 $\longrightarrow F = D0 \cdot \overline{S1} \cdot \overline{S0}$

F = D1 when S1 = 0 and S0= 1 $\longrightarrow F = D1 \cdot \overline{S1} \cdot S0$

F = D2 when S1 = 1 and S0 = 0 $\longrightarrow F = D2 \cdot S1 \cdot \overline{S0}$

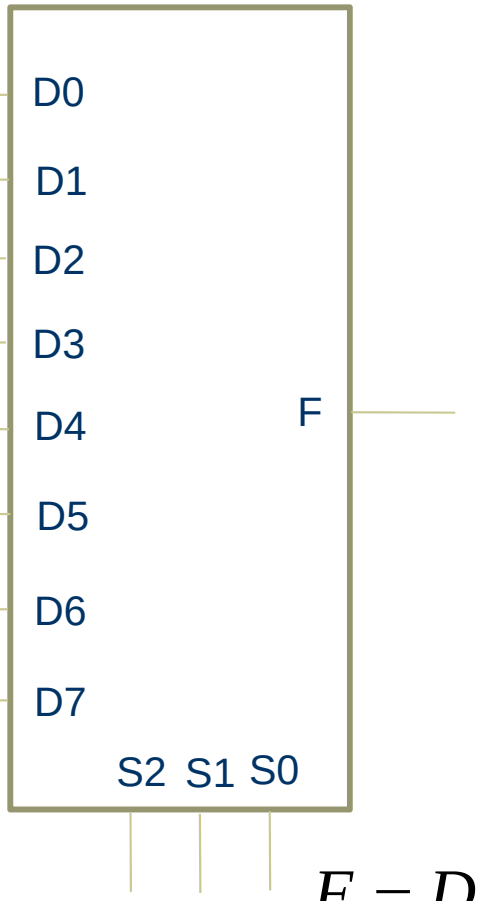
F = D3 when S1 = 1 and S0 = 1 $\longrightarrow F = D3 \cdot S1 \cdot S0$

Therefore, $F = D0 \cdot \overline{S1} \cdot \overline{S0} + D1 \cdot \overline{S1} \cdot S0 + D2 \cdot S1 \cdot \overline{S0} + D3 \cdot S1 \cdot S0$

Can you implement this logic function?

Mux (cont.)

- 8x1 Mux



Function:

F = D0 when and S2 = 0 and S1 = 0 and S0 = 0
F = D1 when and S2 = 0 and S1 = 0 and S0 = 1
F = D2 when and S2 = 0 and S1 = 1 and S0 = 0
F = D3 when and S2 = 0 and S1 = 1 and S0 = 1
F = D4 when and S2 = 1 and S1 = 0 and S0 = 0
F = D5 when and S2 = 1 and S1 = 0 and S0 = 1
F = D6 when and S2 = 1 and S1 = 1 and S0 = 0
F = D7 when and S2 = 1 and S1 = 1 and S0 = 1

What is the logic expression for F?

$$F = D0 \cdot \overline{S2} \cdot \overline{S1} \cdot \overline{S0} + D1 \cdot \overline{S2} \cdot \overline{S1} \cdot S0 + \ldots + D7 \cdot S2 \cdot S1 \cdot S0$$

Mux (cont.)

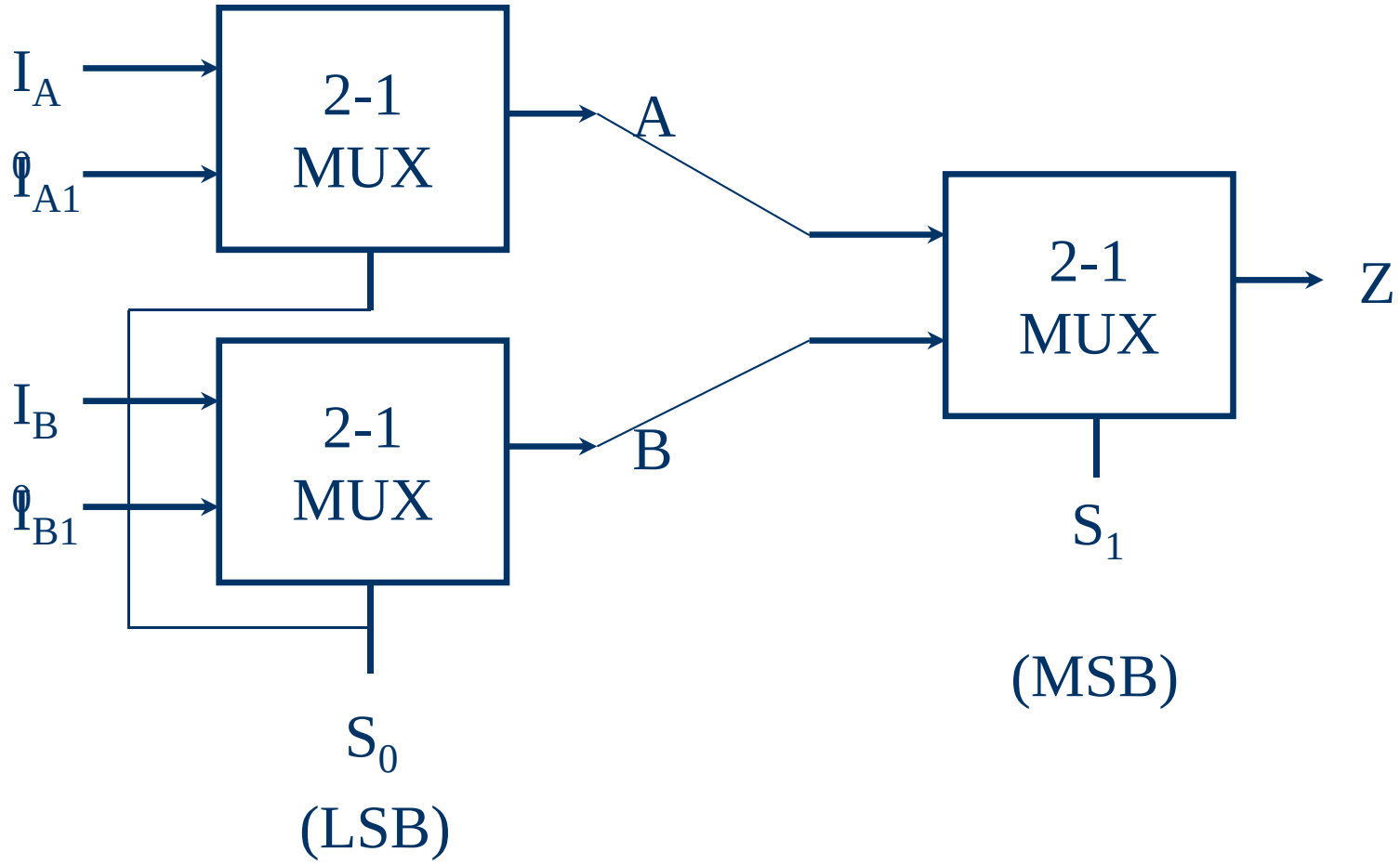
- How to design a 3x1 mux or a 7x1 mux?
  - 3x1 mux is structured as 4x1 mux
  - 7x1 mux is structured as 8x1 mux
- How many select bits is needed for 16x1 mux?
  - 4 select inputs (S3, S2, S1, S0)
- How many inputs does a 32x1 mux have?
  - 5 select bits and 32 input data lines (37 inputs)

Mux IC

- Mux (and other common logic blocks) can be bought as a packaged integrated circuits (IC)
- Commonly used IC is TTL and CMOS
- For example, an inverter IC in TTL is named 74LS04 (LS for Low Speed TTL)
- Inverter IC in CMOS is named 74HC04(HC for High Speed CMOS)
- 2x1 Mux IC: 74LS157/74HC157 (74x157)
- 4x1 Mux IC: 74LS153/74HC153 (74x153)
- 8x1 Mux IC: 74LS151/74HC151 (74x151)

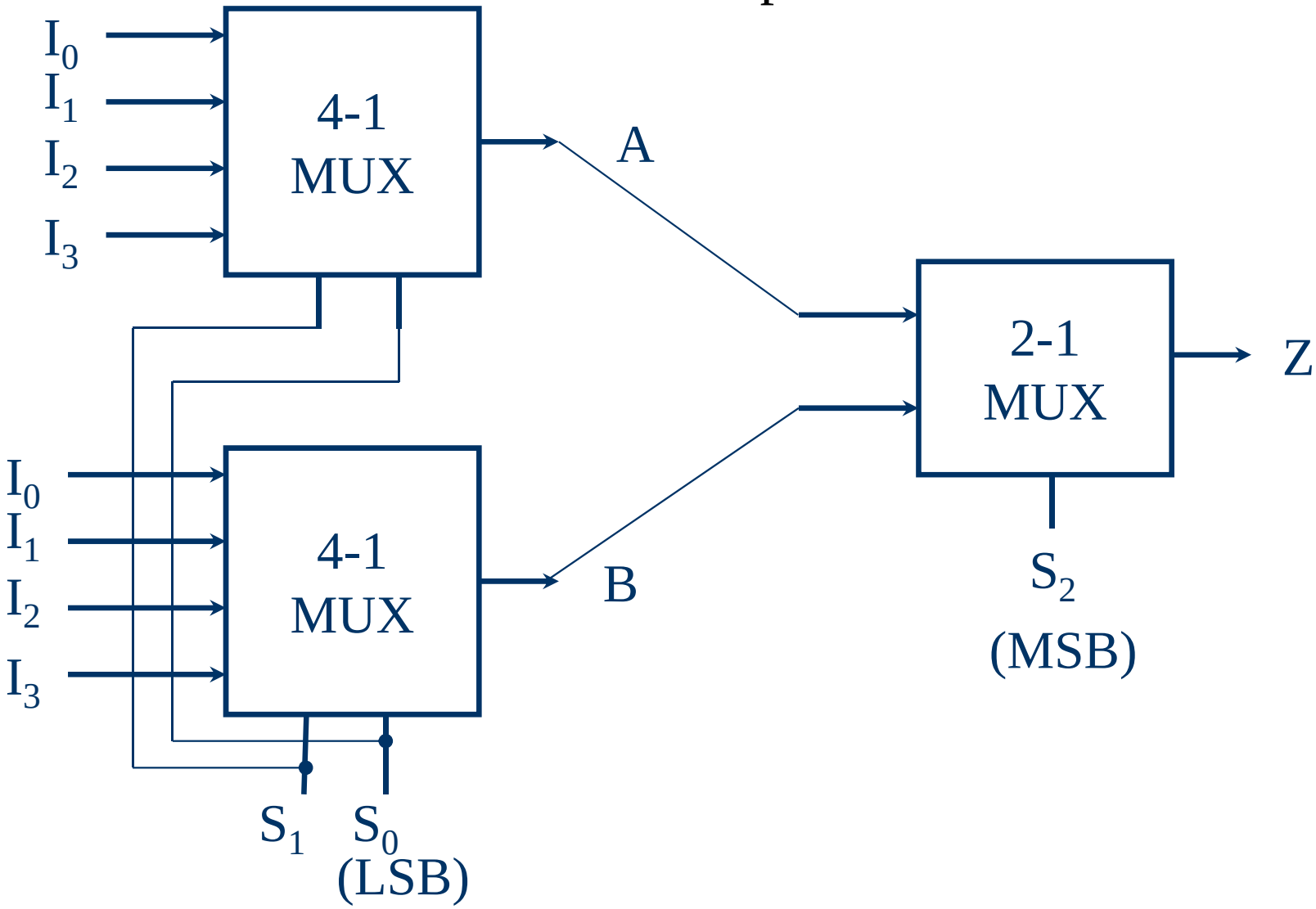
Multiplexer

- Use 2-1 MUX to implement 4-1 MUX.
- 3 units of 2-1 MUX is required.



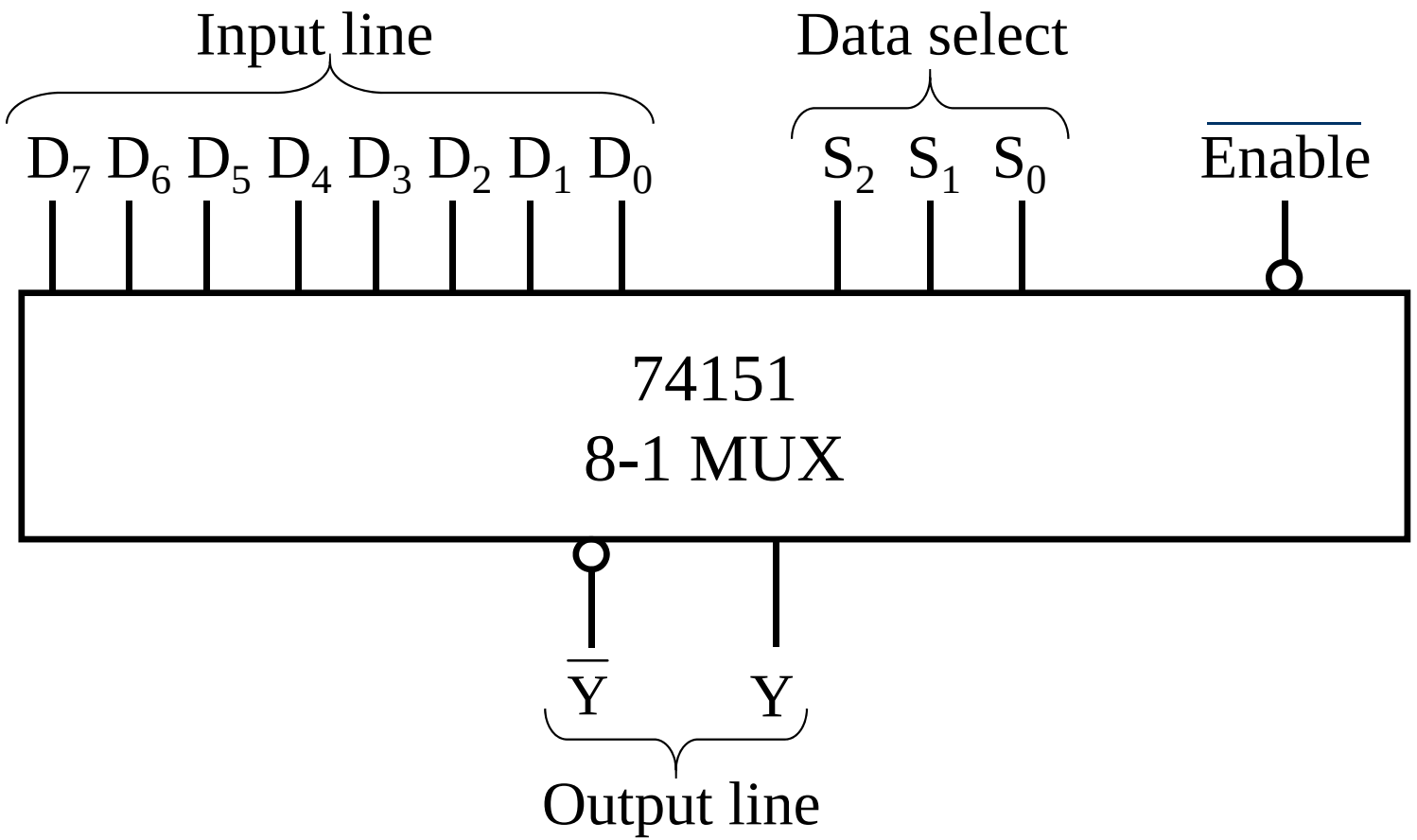
Multiplexer

- Use 4-1 MUX and 2-1 MUX to implement 8-1 MUX.



Multiplexer

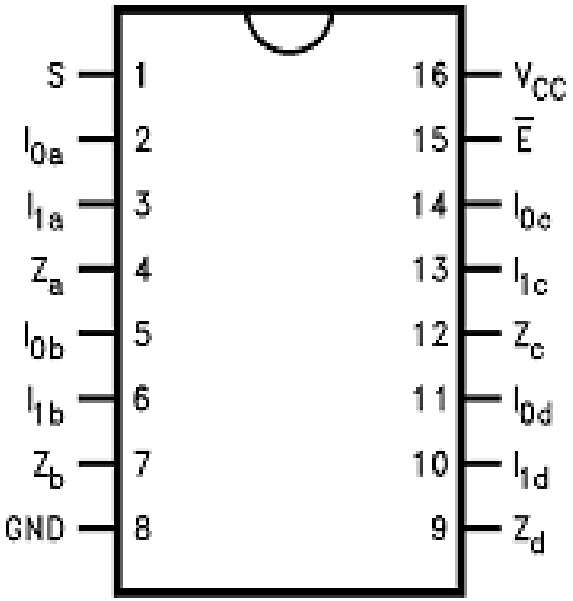
- The 74151 Multiplexer chip



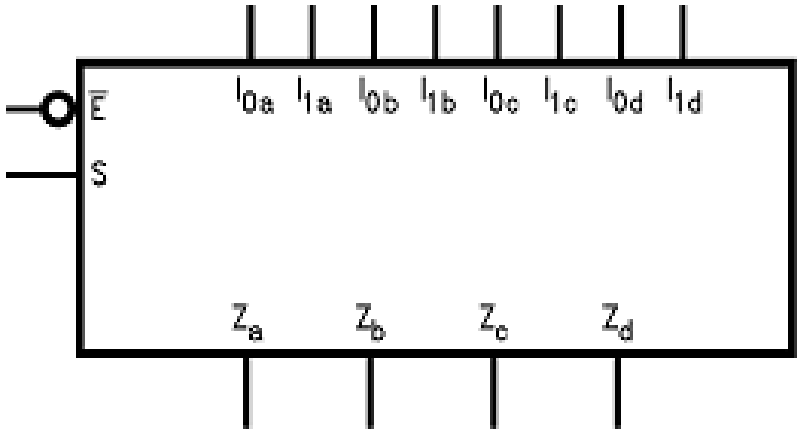
Mux IC (cont.)

- 74x157 Quad 2x1 Multiplexer

Connection Diagram



Logic Symbols



Contains four 2x1 mux with an active low enable (E) and common select (S) – refer datasheet

When S = 0 and E = 0; $Z_a = I_{0a}$, $Z_b = 1_{0b}$, $Z_c = 1_{0c}$, Zd = 1_{0d}

Mux IC (cont.)

- 74x157 Quad 2x1 Multiplexer (cont.)

Truth Table

Inputs				Outputs
$\overline{E}$	S	I_0	I_1	Z
H	X	X	X	L
L	H	X	L	L
L	H	X	H	H
L	L	L	X	L
L	L	H	X	H

H = HIGH Voltage Level

L = LOW Voltage Level

X = Immaterial

Output Z selects I_0 or I_1 depending on select S (with $\overline{E} = 0$)

Mux IC (cont.)

- 74x153 Dual 4x1 Multiplexer

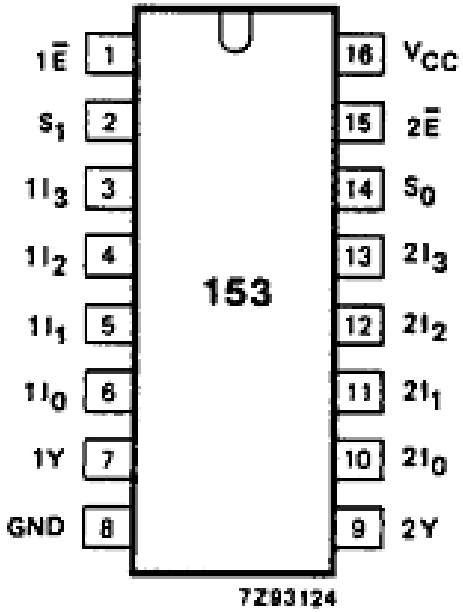


Fig.1 Pin configuration.

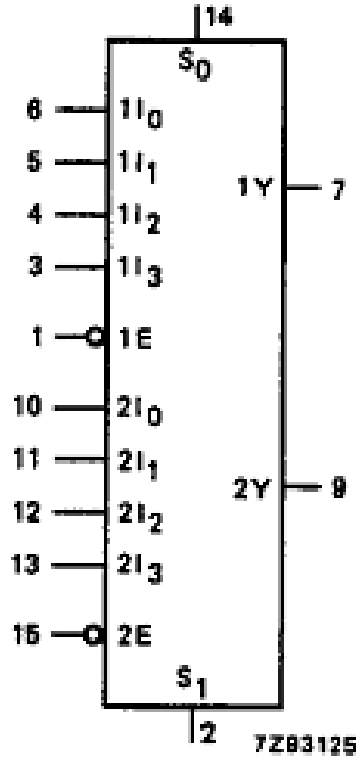


Fig.2 Logic symbol.

Contains two 4x1 mux with separate enable ($1\overline{E}$, $2\overline{E}$) and common select (S1, S0)

Mux IC (cont.)

- 74x153 Dual 4x1 Multiplexer (cont.)

FUNCTION TABLE

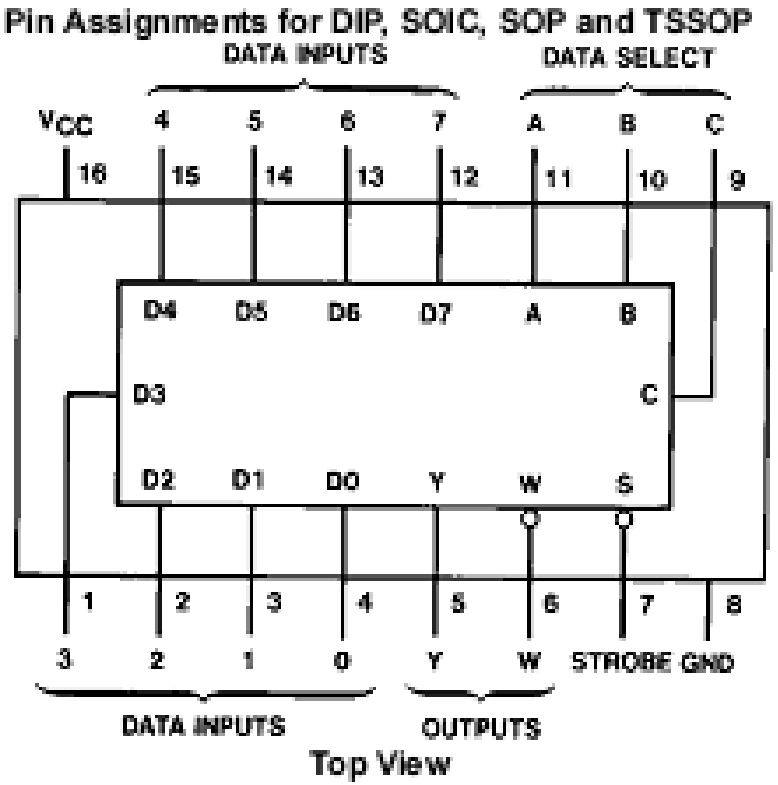
SELECT INPUTS		DATA INPUTS				OUTPUT ENABLE	OUTPUT
S_0	S_1	nI_0	nI_1	nI_2	nI_3	$n\overline{E}$	nY
X	X	X	X	X	X	H	L
L	L	L	X	X	X	L	L
L	L	H	X	X	X	L	H
H	L	X	L	X	X	L	L
H	L	X	H	X	X	L	H
L	H	X	X	L	X	L	L
L	H	X	X	H	X	L	H
H	H	X	X	X	L	L	L
H	H	X	X	X	H	L	H

Output nY selects nI_0, nI_1, nI_2, or nI_3 depending on S_1 and S_0 (with $n\overline{E} = 0$)

Mux IC (cont.)

- 74x151 Single 8x1 Multiplexer

Connection Diagram



Truth Table

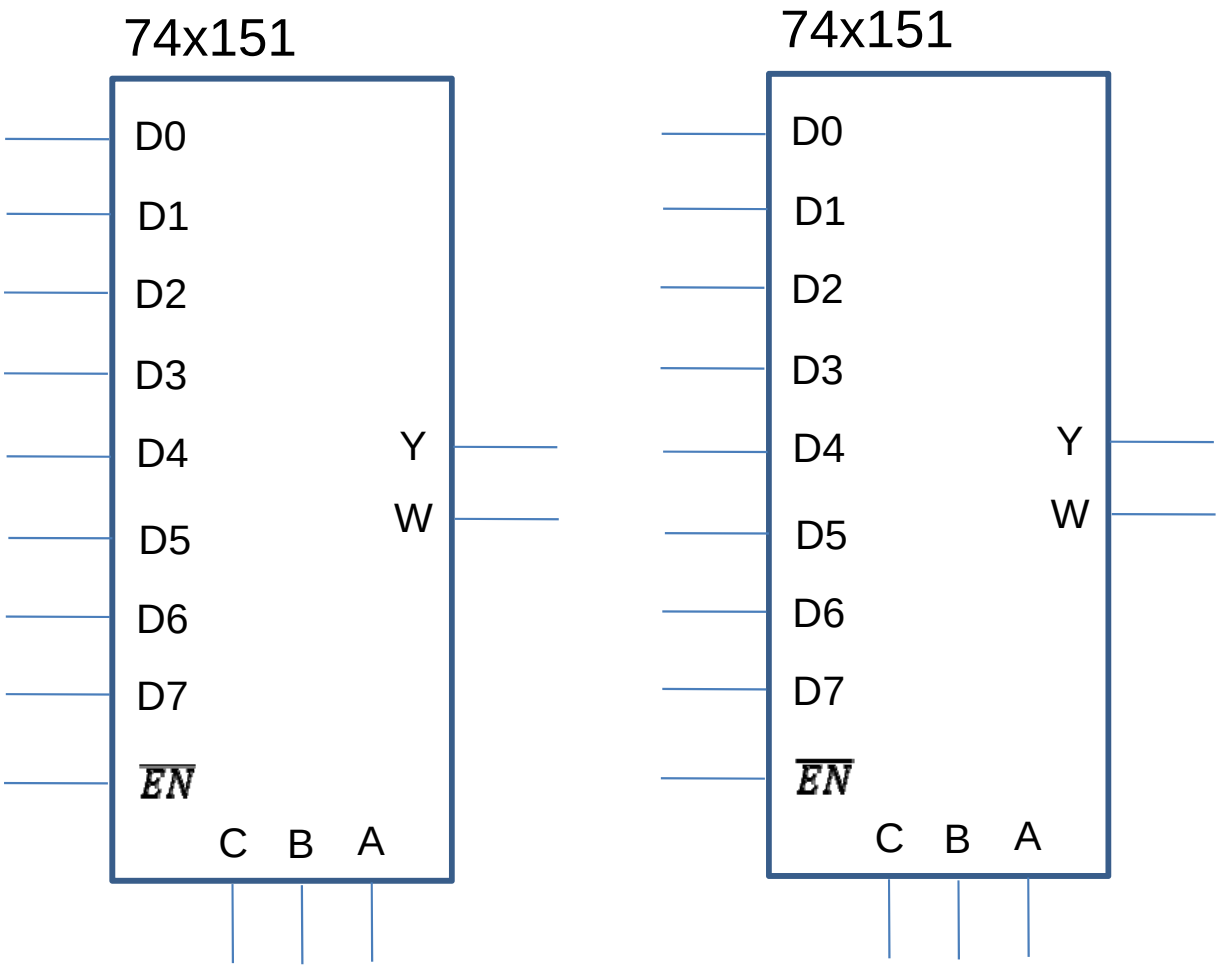
Inputs				Outputs	
Select			Strobe		
C	B	A	S	Y	W
X	X	X	H	L	H
L	L	L	L	D0	$\overline{D0}$
L	L	H	L	D1	$\overline{D1}$
L	H	L	L	D2	$\overline{D2}$
L	H	H	L	D3	$\overline{D3}$
H	L	L	L	D4	$\overline{D4}$
H	L	H	L	D5	$\overline{D5}$
H	H	L	L	D6	$\overline{D6}$
H	H	H	L	D7	$\overline{D7}$

H = HIGH Level, L = LOW Level, X = Don't Care
D0, D1...D7 = the level of the respective D input

Output Y selects D0 to D7 depending on select bits A, B, and C (with S = 0)

Mux IC (cont.)

- How to design a 16x1 mux using two 74x151 IC's and basic gates?



For a 16x1 Mux,

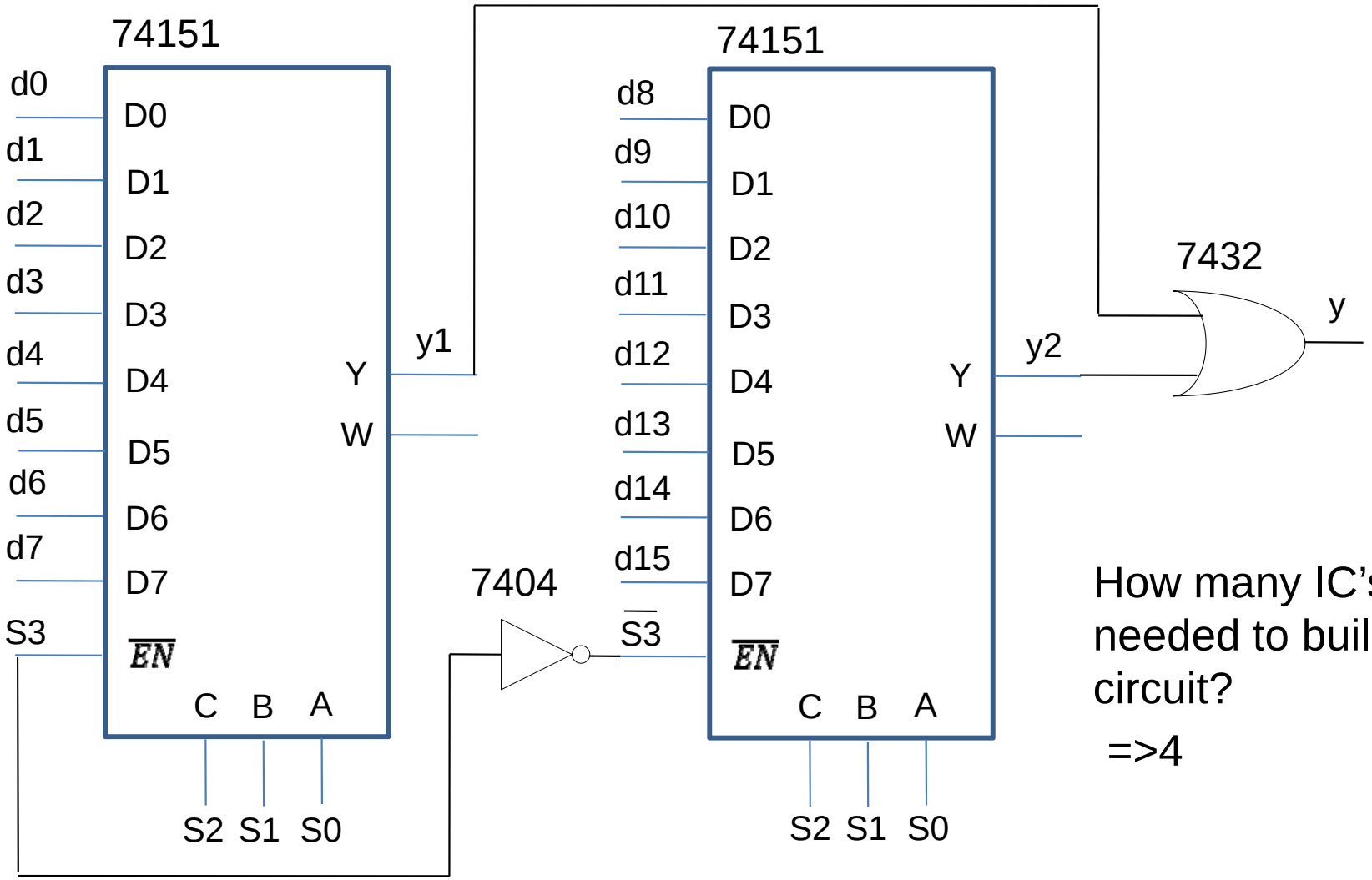
-How many select bits are needed?
=>4

-How many data inputs?
=>16

-How many outputs?
=> 1

Mux IC (cont.)

Assume the 16x1 mux data input is d0 to d15, select input S3 to S0 and output y

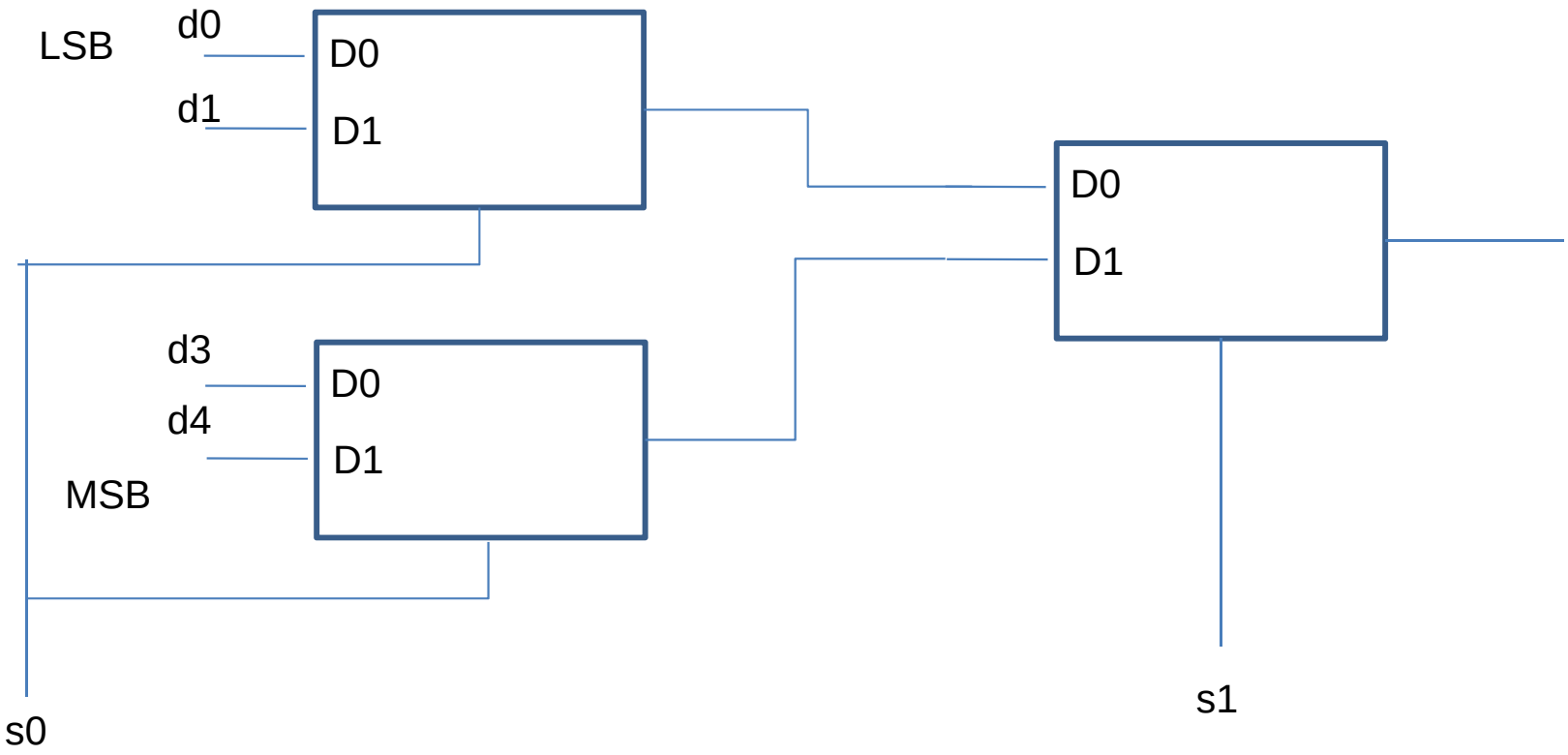


How many IC's are needed to build this circuit?

=>4

Mux IC (cont.)

Example 2: Design a 4x1 mux using three 2x1 mux only

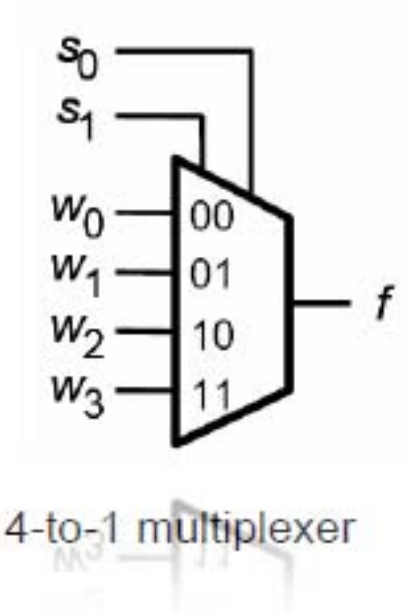


Slide 27

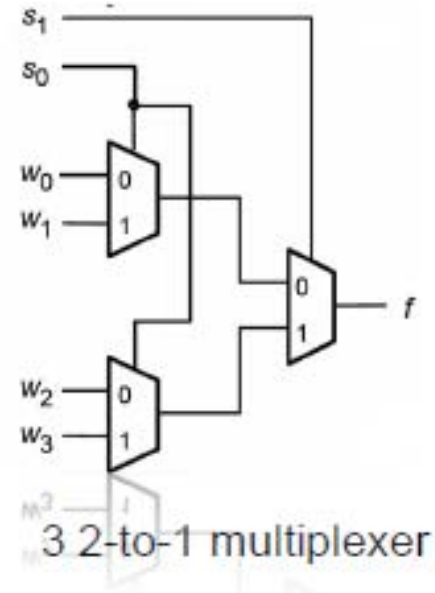
M1 Michael, 10/2/2014

Mux IC (cont.)

s_1	s_0	f
0	0	w_0
0	1	w_1
1	0	w_2
1	1	w_3



4-to-1 multiplexer



3 2-to-1 multiplexer

Larger multiplexer can be built based on small multiplexers such as 4-to-1 or 2-to-1 multiplexer, e.g., a 4-to-1 multiplexer can be built with 3 2-to-1multiplexers,a 8-to-1 multiplexer can be composed of 2 4-to-1 and a 2-to-1 multiplexers, and a 16-to-1 multiplexer can be set up using 5 4-to-1 multiplexers.

Slide 28

M1 Michael, 10/2/2014

Mux IC (cont.)

Example 3: Design a 32x1 mux using two 16x1 mux and one 2x1 mux

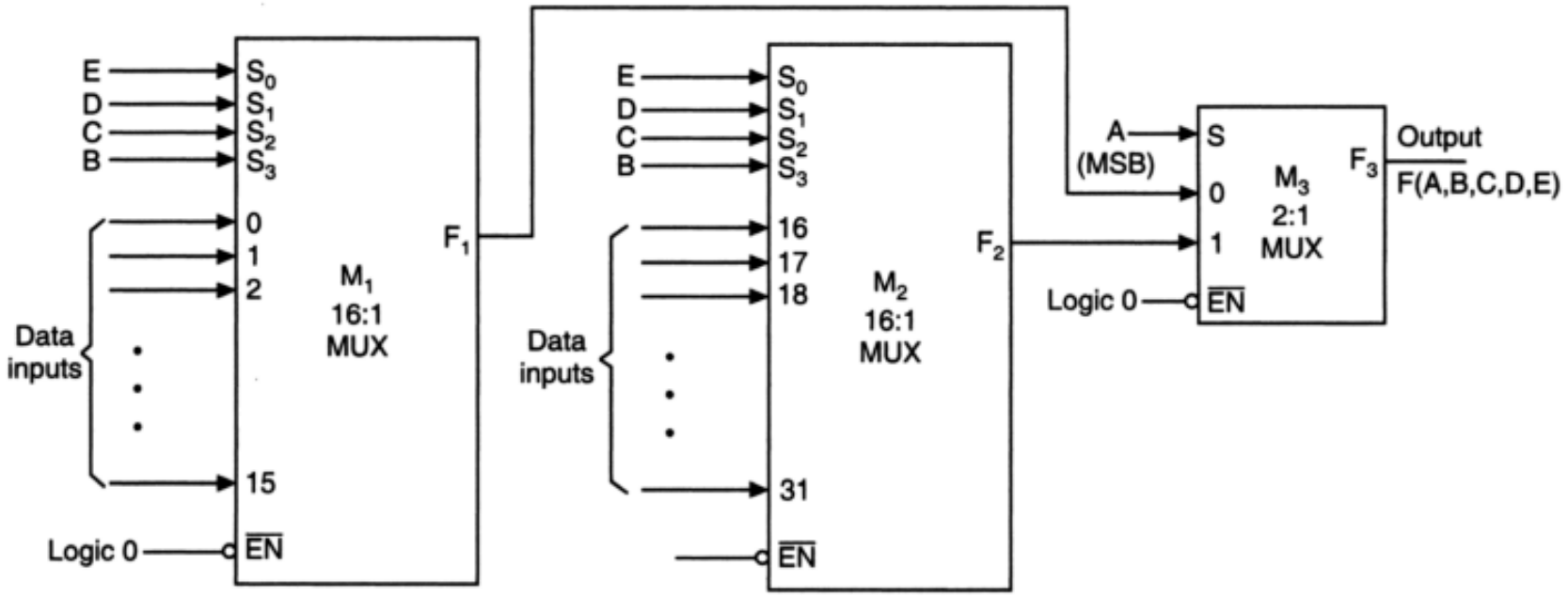


Figure 7.94 32:1 mux using two 16:1 muxs and one 2:1 mux.

Slide 29

M1 Michael, 10/2/2014

Exercise 1

- Using the 74151 Multiplexer chip to implement 16-1 MUX
- Solution:
  - Use 2 units of 74151, one NOT gate (for MSB data select) and one OR gate (for output line)

Exercise 2

- Refer to the truth table of Decoder Exercise 2. Implement the output of the truth table using
  - 8-1 MUX only
  - 4-1 multiplexer only with other basic gates if required.
    - If you are allowed to use only **ONE 4-1 multiplexer** and basic gates, you will have to use the ***Entered Variable*** method. Solution is shown in next page.
    - If you are allowed to use more than one MUX, then you may use the cascading method.

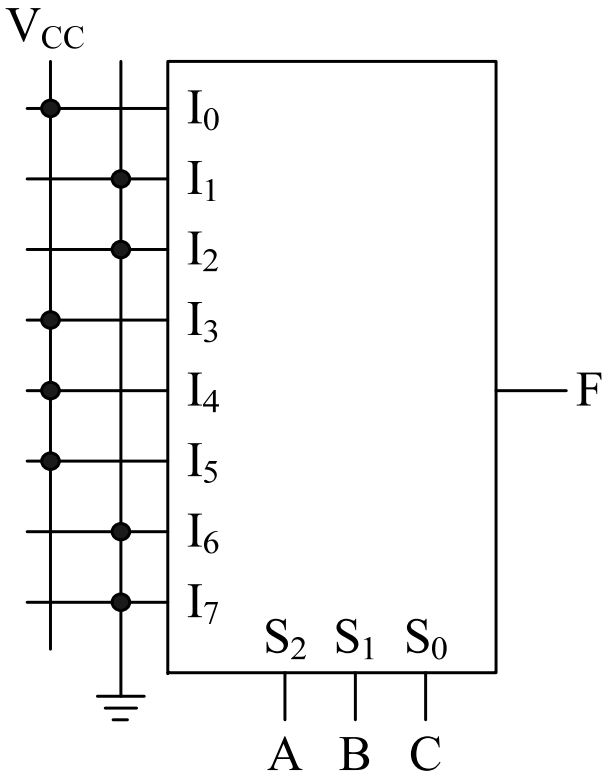
Solution Exercise 2

- 8-1 MUX only

The simplest way:
MSB of Boolean function is connected to MSB of the select bit and so on.

If ABC = 000, output of Boolean function F = 1;
$S_2S_1S_0$ also = 000, output of the multiplexer F = I_0.
Therefore I_0 is connected to V_{CC} (logic 1).

A	B	C	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



Solution Exercise 2

- 4-1 MUX only : Using ***Entered Variable*** method

-Only two inputs of the Boolean function can be connected to the select bits.
-Therefore, need to modify the truth table.
- Normal truth-table has outputs 0, 1 or X. Entered variable truth-table has input variable at the output column. A and B are the inputs, C is the entered variable.

A	B	C	F	
0	0	0	1	$F = \overline{C}$
0	0	1	0	
0	1	0	0	$F = C$
0	1	1	1	
1	0	0	1	$F = 1$
1	0	1	1	
1	1	0	0	$F = 0$
1	1	1	0	

A	B	F
0	0	$\overline{C}$
0	1	C
1	0	1
1	1	0

Exercise 3

- Implement F(A,B,C,D) = BD’ + AC’ + BC’D+ABCD using one 4-1 MUX and NAND gates only. A and B must be connected to the select bits S_1 and S_0 respectively.

Solution Exercise 3

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

- F(A,B,C,D) = BD’ + AC’ + BC’D+ABCD

A	B	F
0	0	0
0	1	$\overline{CD}$
1	0	$\overline{C}$
1	1	1