MULTIPLEXERS

Experiment No. 7

Aim: Implementation of Multiplexers using gates and TTL ICs

Multiplexers: The term 'multiplexer' means "many in to one". Multiplexing is the process of transmitting a large number of information over a single line. The multiplexer(Data-Selector) has several Data-Input Lines and a single Output line. The selection of a particular input line is controlled by a set of selection lines. The block diagram of a multiplexer (MUX) with 'n' (D_n, D_{n-1}, ……., D_1, D_0) Input Lines, 'm' (S_m, S_{m-1},….., S_1, S_0) Select Lines, and one Output Line 'Y' is shown below.

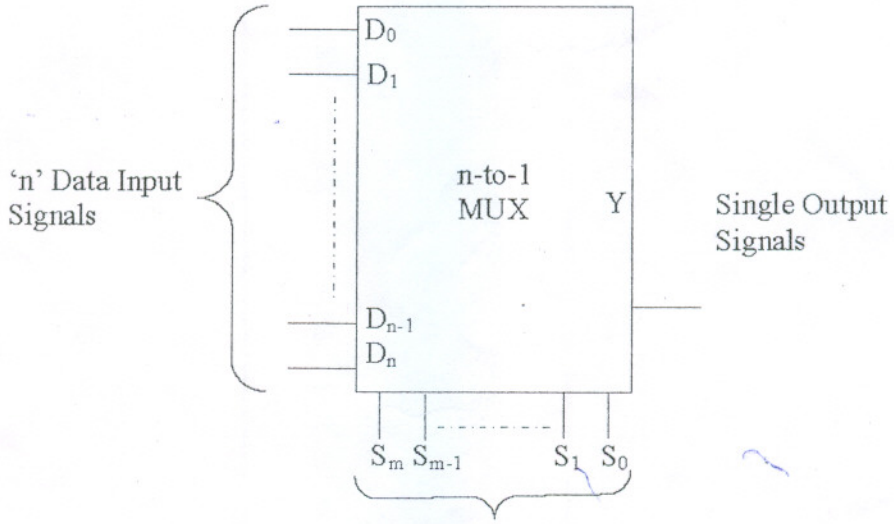

Fig.1. Block Diagram of n-to-1 MUX.

Activity 1. *2-to-1line Multiplexer*

A 2-to-1 MUX ha two data input lines, one select line, and one output line. Its logic symbol is shown in Fig.2, below.

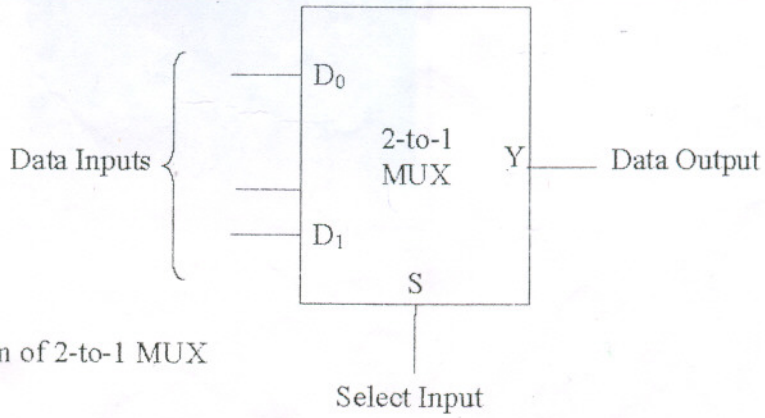

Fig.2. Block Diagram of 2-to-1 MUX

Logic Diagram:

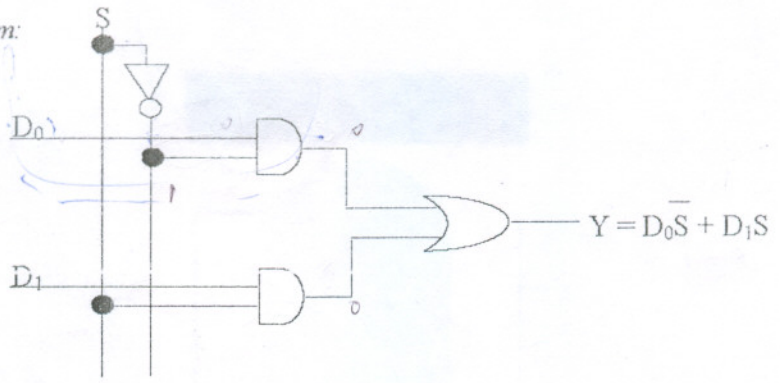

Fig.3. Logic Diagram of 2-to-1 MUX

Procedure

Step.1. Construct the circuit shown in the Fig.3.

Step.2. Apply the Data Inputs (D0 & D1) and Select Input (S) signal and record the Outputs.

Observations: *Truth-Table*

Data Select Inputs	Data inputs		Output	Comment on Output (Write D_1 / D_0)
S	D_1	D_0	Y	
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

Activity 2. *Implementation of 2-to-1 line MUX with Enable / Disable input*

Connect an extra Enable signal 'E' to the 2-to-1 MUX, as shown in Fig.4.

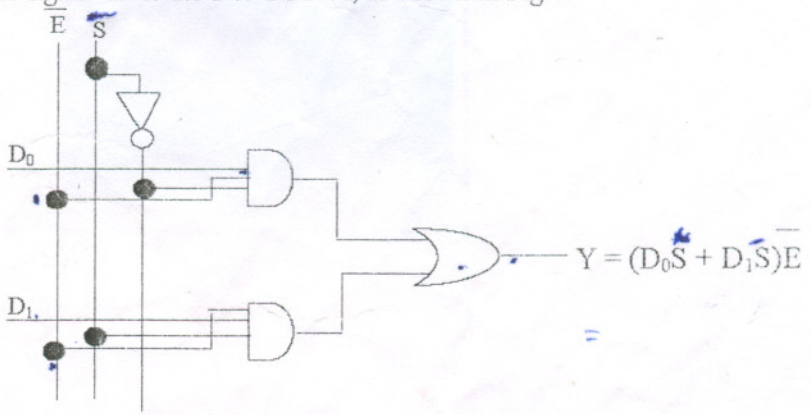

Fig.4.

Observations: *Truth-Table*

Enable Input	Select Inputs	Data inputs		Output	Comment on Output (Write D_1 / D_0)
E'	S	D_1	D_0	Y	
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		
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		

Write:

1. When Enable Input is HIGH, the Output of the MUX is.....(Enabled / Disabled).

2. When Enable Input is LOW, the Output of the MUX is ..... (Enabled/Disabled).

Activity 3. *2-to-1 MUX using IC 74157*

The Logic Symbol of the IC 74157 (Quad 2-Input Multiplexer) is shown in Fig.5., below.

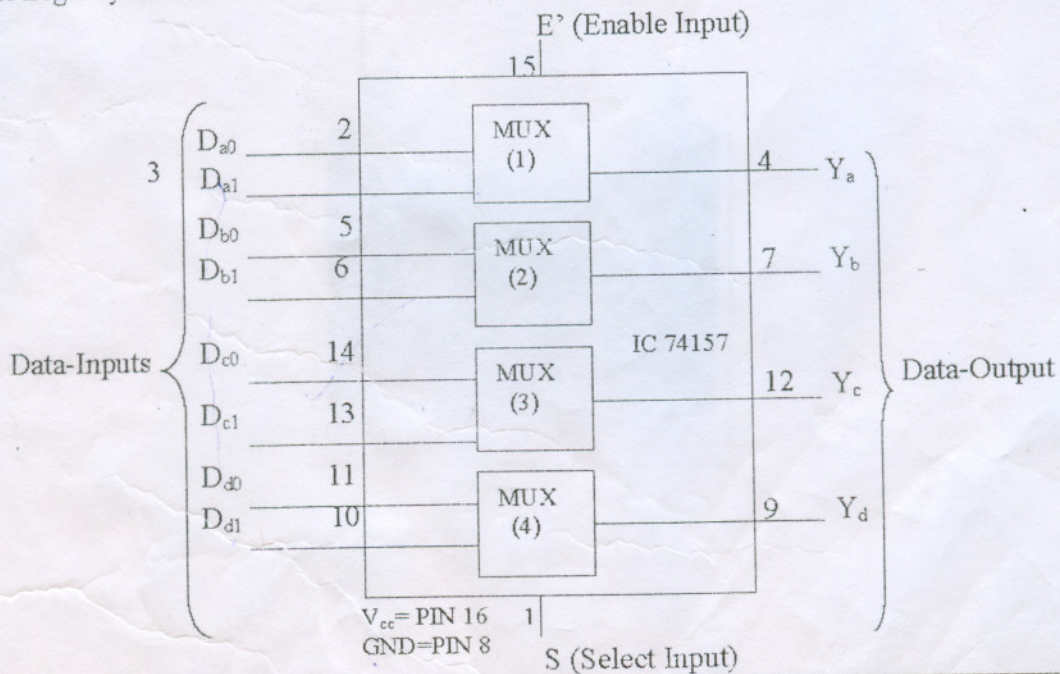

Fig.5.

Procedure

Step.1. Choose one multiplier (say MUX1) out of the four multipliers, as shown in the Fig.5.

Step.2. Make the proper connections of the Data Inputs, Enable Input, Select Input, and Output.

Step.3. Apply the Enable Signal and the Select Signal to the MUX and take observations.

***Observations**: Truth-Table*

Enable Input	Select Input	Data Inputs		Output	Comment on Outputs Y_a (D_{a0} / D_{a1})
E'	S	D_{a0}	D_{a1}	Y_a	
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		

Activity 4. Implementation of 4-to-1line Multiplexer using IC 74153

The Logic Symbol for the IC 74153 (Dual 4-to-1 Multiplexer) is shown in Fig.8, below.

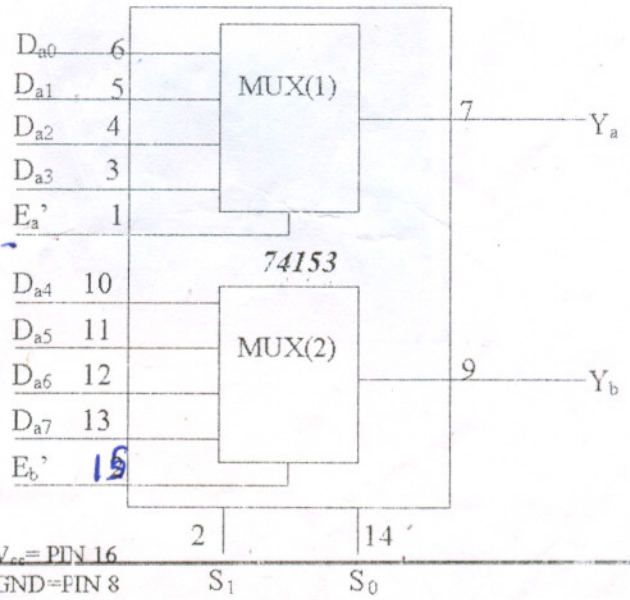

Fig.6. V_{cc}= PIN 16
GND=PIN 8

Procedure

Step.1. Choose either MUX1 or MUX2 to perform the experiment.

Step.2. Make the proper connections of the Data Inputs, Enable input, Select Input, and Output.

Step.3. Apply the Enable signal and Select Signal to the MUX.

Step.4. Take observations.

Observations: Truth-Table

Enable Input	Select input		Data Inputs				Output	Comment on Outputs Y_a
E'	S_1	S_0	D_{a0}	D_{a1}	D_{a2}	D_{a3}	Y_a	($D_{a0}/D_{a1}/D_{a2}/D_{a3}$)
1	X	X	X	X	X	X	0	Don't Care
0	0	0	0	0	0	D		
0	0	1	0	0	D	0		
0	1	0	0	D	0	0		
0	1	1	D	0	0	0		

Activity 5. *Implementation of 8-to-1 line MUX using Dual 4-to-1 line MUX*

Higher order (more number of inputs) multiplexers can be implemented using lower order (lesser number of inputs) multiplexers. For example,

A 4-to-1 MUX can be implemented using two 2-to-1 MUX,

An 8-to-1 MUX can be implemented using four 4-to-1 or two 2-to-1 MUX,

A 16-to-1 MUX can be implemented using eight 2-to-1 or four 4-to-1 or two 8-to-1 MUX.

In this experiment we will consider the design of 8-to-1 MUX using two 4-to-1 MUX.

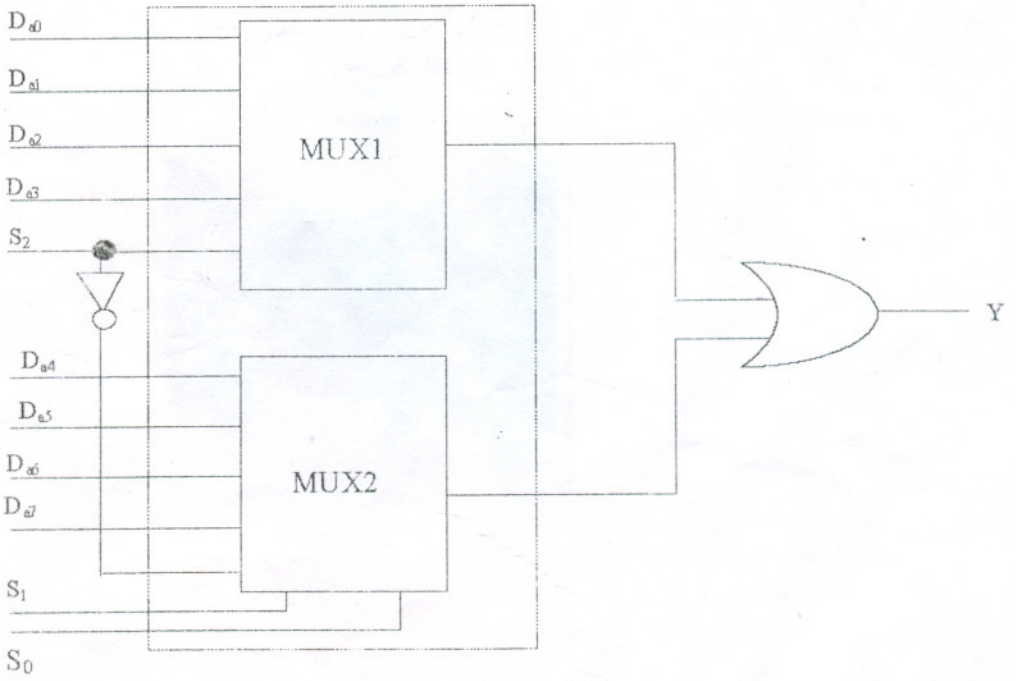

Fig.7.

Procedure

To select one of the 8 inputs, three select lines (S_2, S_1, S_0) are required. Among the three select lines, the Least Significant two select lines (S_1, S_0) are connected with the two select inputs of the multiplexers. The Most Significant select line (S_2) is directly connected to the Enable input of the MUX1 while the same is connected to the Enable input of the MUX2, see Fig.8. Therefore,

1. When $S_2 = 0$, MUX1 is selected as MUX2 is disabled and the data inputs D_0 to D_3 are multiplexed to the output Y.

2. When $S_2 = 1$, MUX2 is selected as MUX1 is disabled and the data inputs D_4 to D_7 are multiplexed to the output Y.

Also, note that the outputs of the MUX1 and MUX2 are ORed using an OR gate to generate output Y.

Observations: *Truth-Table*

Enable Input	Select Input			Data Inputs								Output	Comments on Outputs
E'	S2	S1	S0	D_{a0}	D_{a1}	D_{a2}	D_{a3}	D_{a4}	D_{a5}	D_{a6}	D_{a7}	Y	
1	X	X	X	X	X	X	X	X	X	X	X	0	Don't Care
0	0	0	0	D	0	0	0	0	0	0	0		
0	0	0	1	0	D	0	0	0	0	0	0		
0	0	1	0	0	0	D	0	0	0	0	0		
0	0	1	1	0	0	0	D	0	0	0	0		
0	1	0	0	0	0	0	0	D	0	0	0		
0	1	0	1	0	0	0	0	0	D	0	0		
0	1	1	0	0	0	0	0	0	0	D	0		
0	1	1	1	0	0	0	0	0	0	0	D		