Lab Experiment # 09

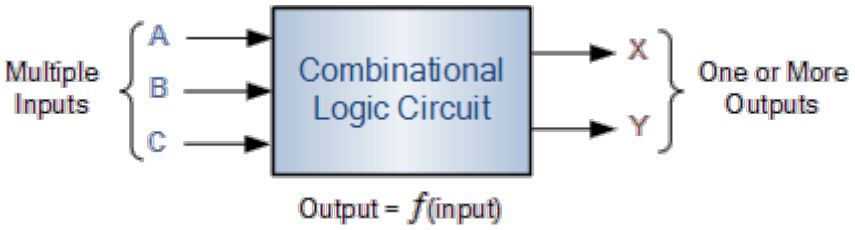
Building logic circuits using Multiplexers

Objectives

- **To learn how to build combinational logic circuits using multiplexers.**

Background

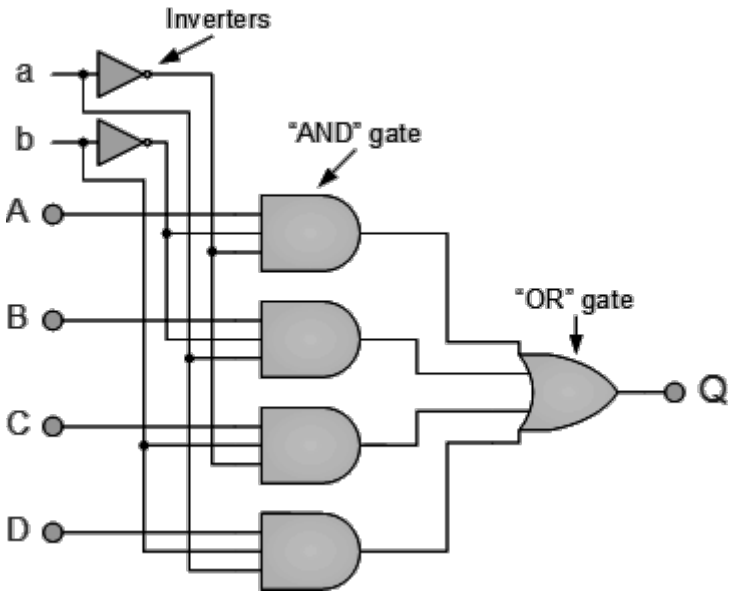
In a **Combinational Logic Circuit**, the output is dependant at all times on the combination of its inputs. Some examples of a combinational circuit include **Multiplexers**, **De-multiplexers**, **Encoders**, **Decoders**, **Full** and **Half Adders** etc.



A Multiplexer is a combination of logic gates resulting into circuits with two or more inputs (data inputs) and one output.

4 Channel Multiplexer using Logic Gates

The following circuit shows a 4x1 mux. Based on the binary value placed at the inputs "a" and "b", what will appear at the circuit output Q is one of the following values: A, B, C, or D.

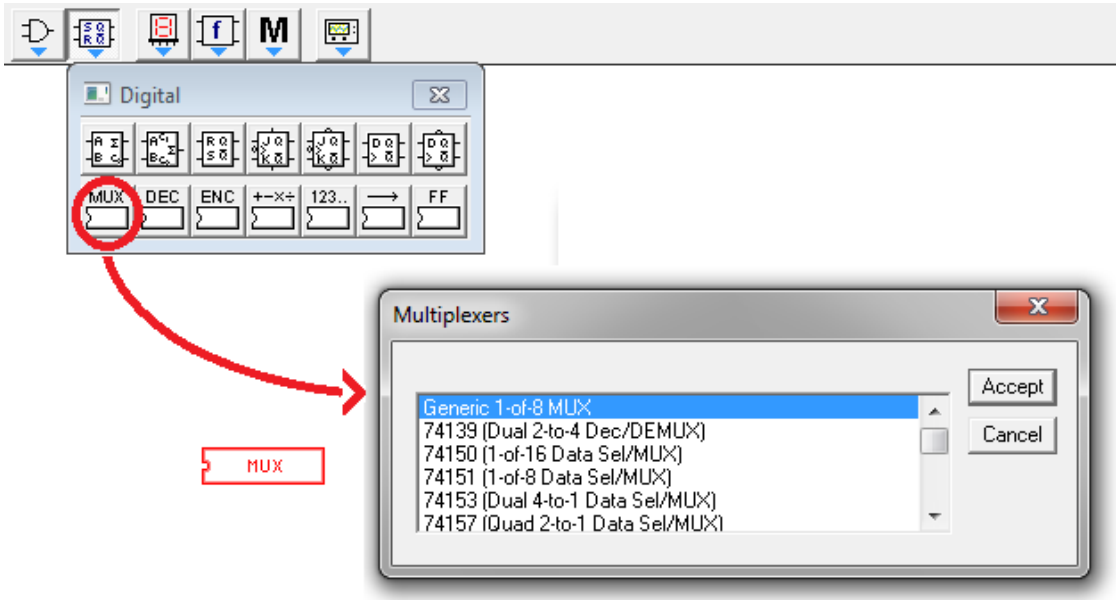


The circuit above is implemented based on the following truth table.

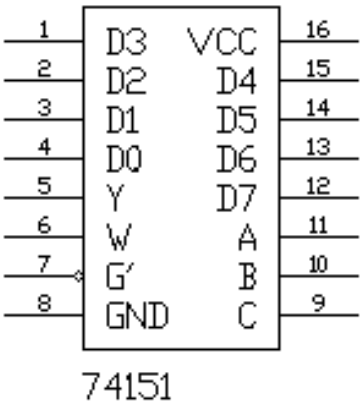
a	b	Q
0	0	A
0	1	B
1	0	C
1	1	D

Drawing Multiplexers in EWB:

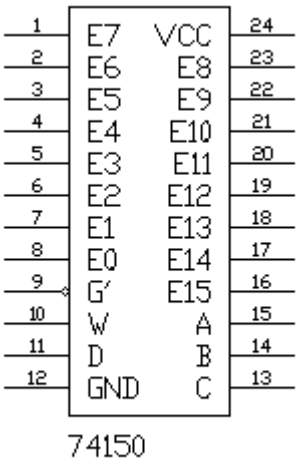
Task: Draw the previous lab examples using EWB, follow the steps below to implement Multiplexers and Decoders.



Then choose 74151 (1-of-8 Data Sel/MUX)from the list:



You may also choose 74150 (1-of-16 Data Sel/Mux) as follows



NOTE: the "A" line in the multiplexer is the least significant bit, while "C" is the most significant bit.

Data selector/multiplexer truth table:

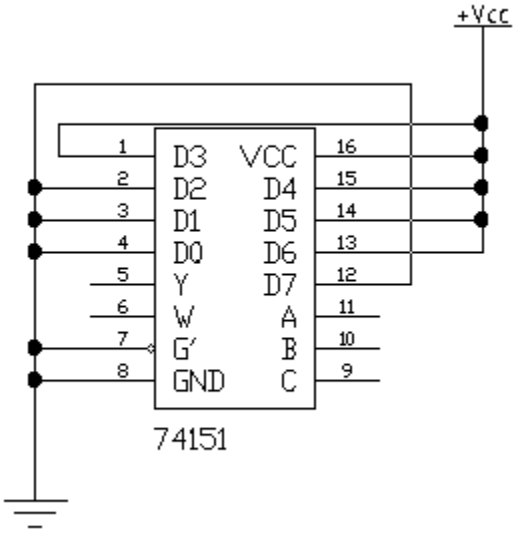
Select			**Strobe**	**Outputs**	
C	**B**	**A**	**G'**	**W**	**Y**
x	x	x	1	1	0
0	0	0	0	D0'	D0
0	0	1	0	D1'	D1
0	1	0	0	D2'	D2
0	1	1	0	D3'	D3
1	0	0	0	D4'	D4
1	0	1	0	D5'	D5
1	1	0	0	D6'	D6
1	1	1	0	D7'	D7

Multiplexers can be used to synthesize logic functions

4-to-1 MUX can realize any 3-variable function, 8-to-1 MUX can realize a 3-variable or 4-variable function, in general 2^n-to-1 MUX can realize an (n +1)-variable and n-variable function.

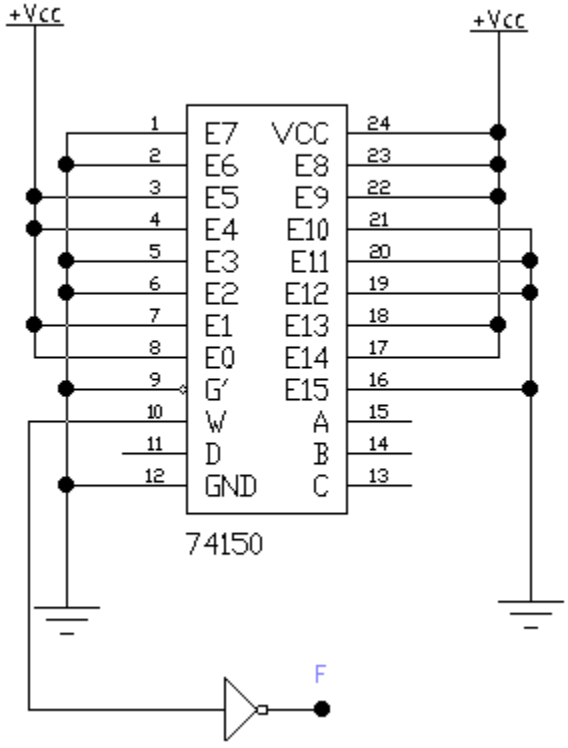
Example: realizing functions using Multiplexers

The function
F=A'BC+AB'+AC'
Can be implemented using an 8-1 mux as follows



Example: realizing functions using Multiplexers

The function
F= A'C'+B'C'+C'D+ABCD'
Can be implemented using an 16-1 mux as follows



Example: realizing a 4-variable function using 8-to-1Multiplexer

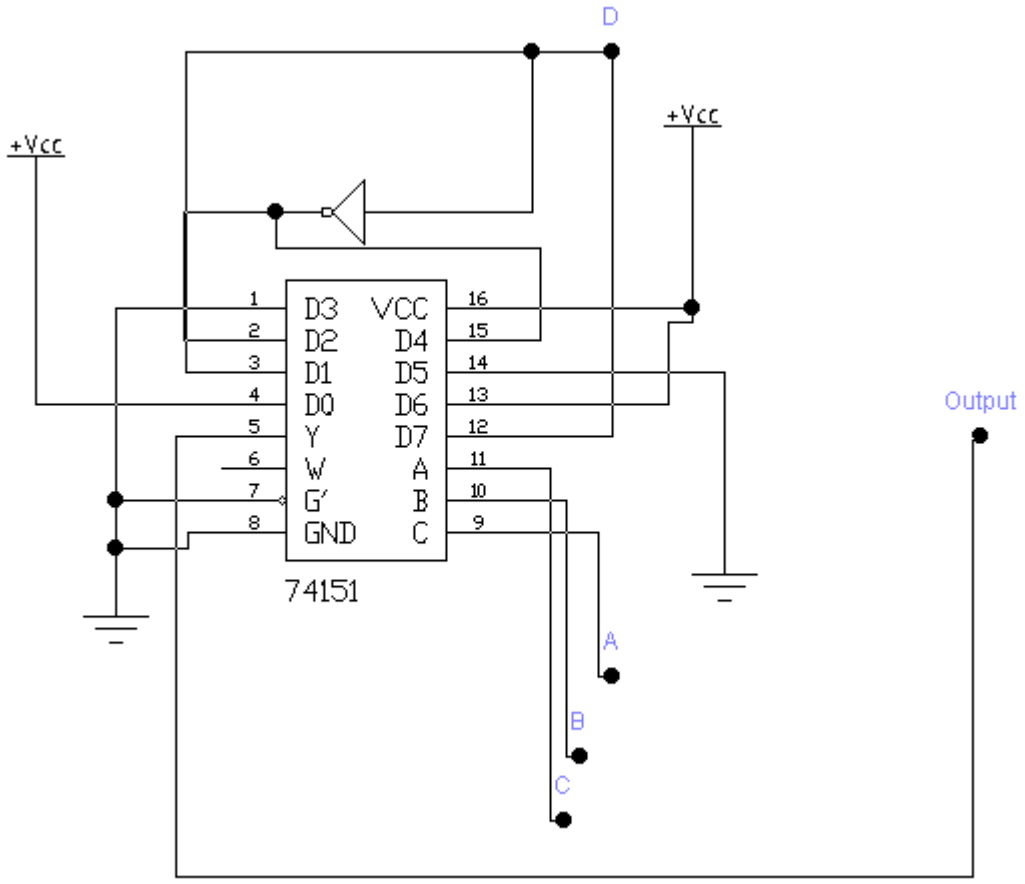
F(A, B, C, D) = A'B'C'D'+A'B'C'D+A'B'CD+A'BC'D'+AB'C'D'+ABC'D'+ABC'D+ABCD

Truth table:

	A	**B**	**C**	**D**	**F**	
0	0	0	0	0	1	F=1
1	0	0	0	1	1	

2	0	0	1	0	0	F=D
3	0	0	1	1	1	
4	0	1	0	0	1	F=D’
5	0	1	0	1	0	
6	0	1	1	0	0	F=0
7	0	1	1	1	0	
8	1	0	0	0	1	F=D’
9	1	0	0	1	0	
10	1	0	1	0	0	F=0
11	1	0	1	1	0	
12	1	1	0	0	1	F=1
13	1	1	0	1	1	
14	1	1	1	0	0	F=D
15	1	1	1	1	1	

To implement this function using EWB, you draw the following circuit:



Lab Tasks

Task 1: Implementing single-output circuits using muxes

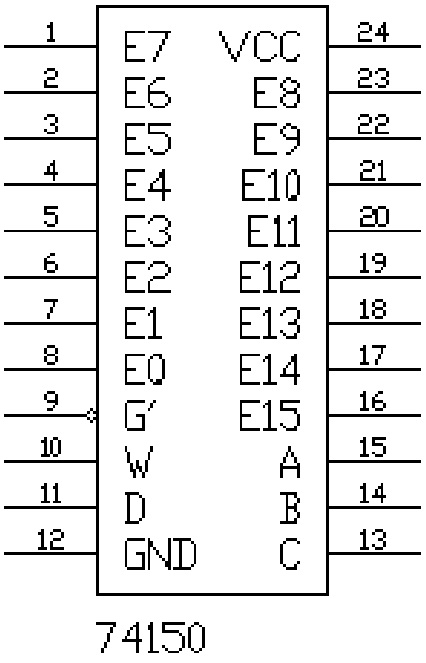
Implement the following function using one 8x1 multiplexer
F(A, B, C, D) = A'B'C'D'+A'B'C'D+A'B'CD+A'BC'D'+AB'C'D'+ABC'D'+ABC'D+ABCD

Note: this example has already been solved above. Just draw the circuit using EWB.

Task 2: Implementing single-output circuits using muxes

Implement the following function using one 16x1 multiplexer

F(A, B, C, D) = A'C'B+AB'C'+B’C'D+ABCD'

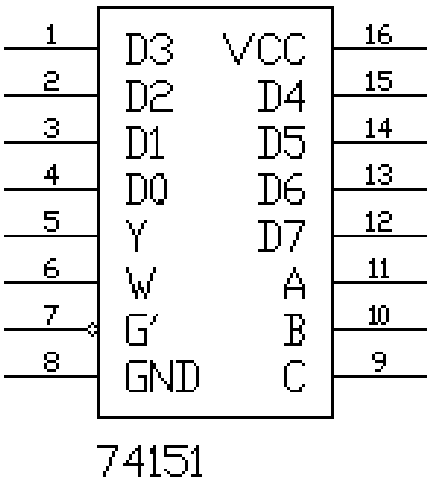


Task 3: Implementing single-output circuits using muxes

Implement the following function using one 8x1 multiplexer

F(A, B, C, D) = A'C'B+AB'C'+B’C'D+ABCD'

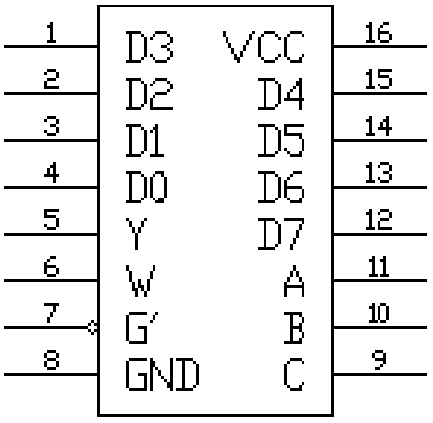
	A	B	C	D	F	
0	0	0	0	0		
1	0	0	0	1		
2	0	0	1	0		
3	0	0	1	1		
4	0	1	0	0		
5	0	1	0	1		
6	0	1	1	0		
7	0	1	1	1		
8	1	0	0	0		
9	1	0	0	1		
10	1	0	1	0		
11	1	0	1	1		
12	1	1	0	0		
13	1	1	0	1		
14	1	1	1	0		
15	1	1	1	1		



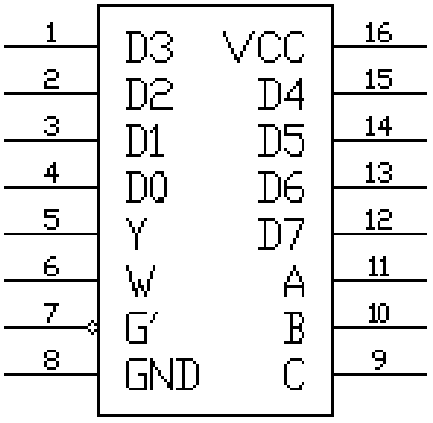
Task 4: Problems with verbal description

Design a combinational circuit (using two 8-to-1 multiplexers) with three inputs, and one output to implement the following function.

	A	B	C	D	F
0	0	0	0	0	0
1	0	0	0	1	1
2	0	0	1	0	0
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	0
6	0	1	1	0	1
7	0	1	1	1	1
8	1	0	0	0	1
9	1	0	0	1	0
10	1	0	1	0	1
11	1	0	1	1	0
12	1	1	0	0	1
13	1	1	0	1	0
14	1	1	1	0	0
15	1	1	1	1	1


1
2
3
4
5
6
7
8
D3
D2
D1
D0
Y
W
G'
GND
VCC
D4
D5
D6
D7
A
B
C
16
15
14
13
12
11
10
9

74151


1
2
3
4
5
6
7
8
D3
D2
D1
D0
Y
W
G'
GND
VCC
D4
D5
D6
D7
A
B
C
16
15
14
13
12
11
10
9

74151