EXPERIMENT 6

DECODERS AND MULTIPLEXERS

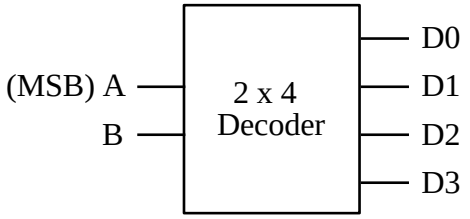
Objective

The objective of this laboratory is to investigate the use of decoders and multiplexers to implement combinational logic circuits.

Introduction

Decoders

A decoder is a combinational logic circuit that activates one of several output lines based on the input code (typically binary or BCD). Shown below in Figure 1 is a block diagram and a truth table for a 2-line-to-4-line (or 2 x 4) decoder that has active-HIGH inputs and outputs.



Inputs		Outputs			
A	B	D0	D1	D2	D3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

Figure 6.1 2 x 4 decoder with active-HIGH inputs and outputs

Note that functionally the outputs of the decoder above correspond to minterms. For example,

$D0 = m_0 = \overline{A} \bullet \overline{B} \bullet \overline{C} \bullet \overline{D}$. A combinational logic function that is expressed as a sum of minterms, therefore, can be implemented by summing decoder outputs.

For example, if $f(A,B) = \Sigma(0, 2, 3)$ then $f(A,B) = D0 + D2 + D3$ so f can be implemented by the circuit shown in Figure 2.

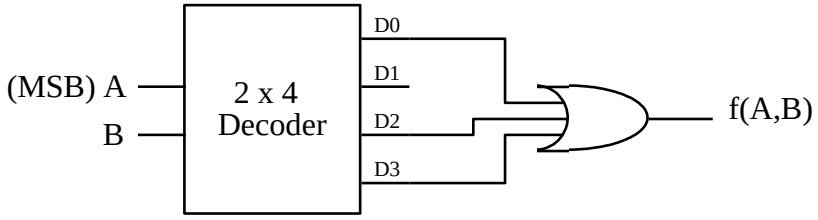
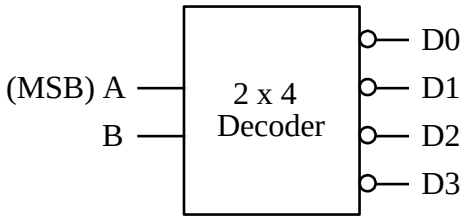


Figure 6.2 $f(A,B) = \Sigma (0,2,3)$ implement using a 2 x 4 decoder

Some decoders, such as the 74LS155, have active-LOW outputs. Figure 3 shows a block diagram and a truth table for a 2 x 4 decoder with active-LOW outputs.



Inputs		Outputs			
A	B	D0	D1	D2	D3
0	0	0	1	1	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	1	0

Figure 6.3 2 x 4 decoder with active-LOW inputs and outputs

Note that functionally the outputs of the decoder above correspond to maxterms. For example,

$D0 = \overline{m_0} = M_0 = \overline{\overline{A} \bullet \overline{B} \bullet \overline{C} \bullet \overline{D}} = (A + B + C + D)$. A combinational logic function that is expressed as a product of maxterms, therefore, can be implemented by ANDing decoder outputs. For example, if f(A,B) = Π(0, 1, 3) then f (A,B)= D0 • D1 • D3 so f can be implemented by the circuit shown in Figure 4.

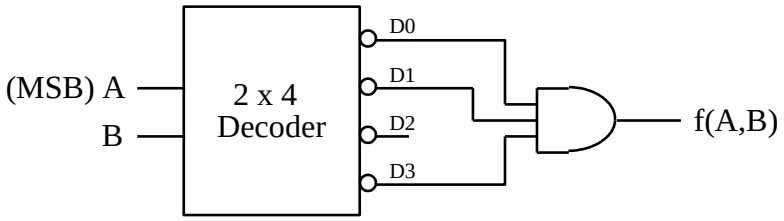


Figure 6.4 f(A,B) = Π(0,1,3) implement using a 2 x 4 decoder

Multiplexers

A ***multiplexer***, or ***data selector***, can be also be used to implement combinational logic circuits. A ***multiplexer implementation table*** is used to determine the input connections for the multiplexer.

A 2 x 1 multiplexer can be used to implement a function of 2 variables, such as f(A,B)

A 4 x 1 multiplexer can be used to implement a function of 3 variables, such as f(A,B,C)

A 8 x 1 multiplexer can be used to implement a function of 4 variables, such as f(A,B,C,D)

Example: Implement the function f(A,B,C) = Σ(0, 3, 6, 7) using a 4 x 1 multiplexer.

The multiplexer implementation table is shown below in Figure 6.5.

	(B'C') I_0	(B'C) I_1	(BC') I_2	(BC) I_3
A'	(0)	1	2	(3)
A	4	5	(6)	(7)
	A'	0	A	1

Figure 6.5 Multiplexer implementation table for f(A,B,C) = Σ(0,3,6,7)

Note that each minterm in f(A,B,C) is circled in the table. Connections for each input are determined as follows:

If no minterms are circled in a column, a logical 0 is connected to the input (Ex: $I_1 = 0$)
If the only one minterm is circled in a column, the input is equal to the variable shown to the left (Ex: $I_0 = A'$ and $I_2 = A$)
If both minterms are circled in a column, a logical 1 is connected to the input (Ex: $I_3 = 1$)

The circuit can be implemented as shown in Figure 6.

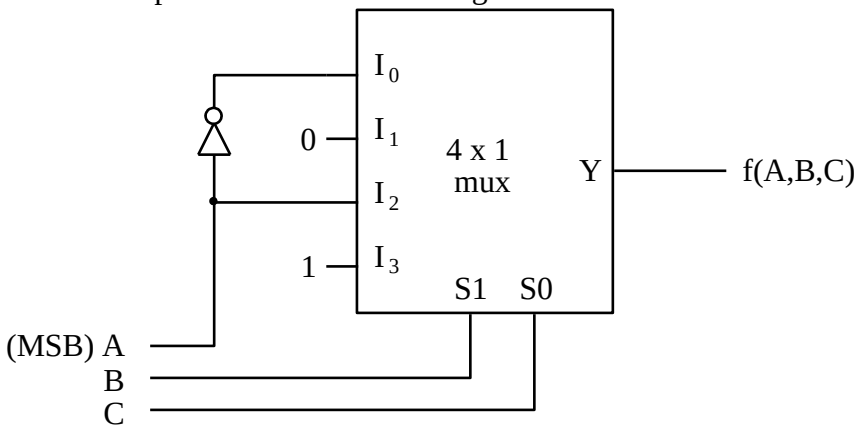


Figure 6.6 f(A,B,C) = Σ (0,3,6,7) implement using a 4 x 1 multiplexer

Keep in mind in the example above that bit A was the MSB. If another bit is the MSB, if the select lines are reversed, or if any bit except the MSB is connected to the inputs, then the multiplexer implementation table and the circuit will change.

For example, if the same function used above is implemented with input C connected to the inputs and inputs A and B to the select lines, then the multiplexer implementation table and the circuit will appear as shown below in Figure 7.

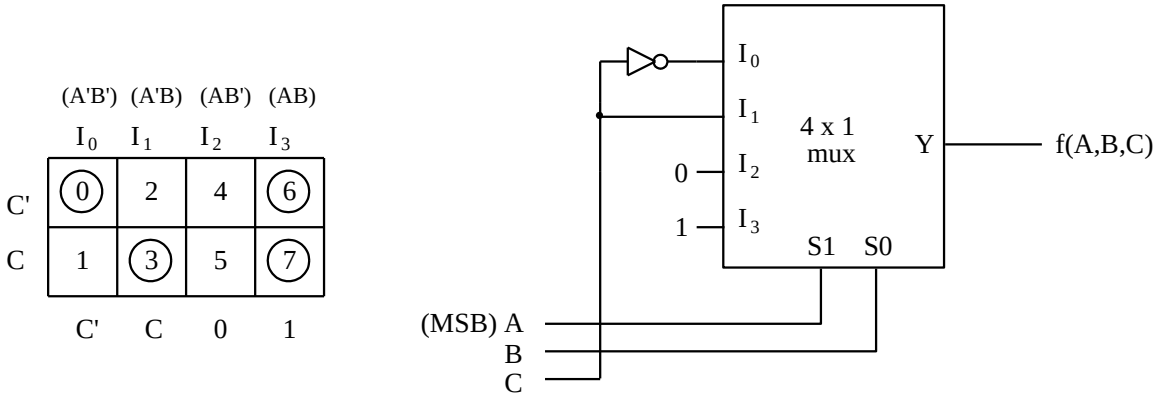


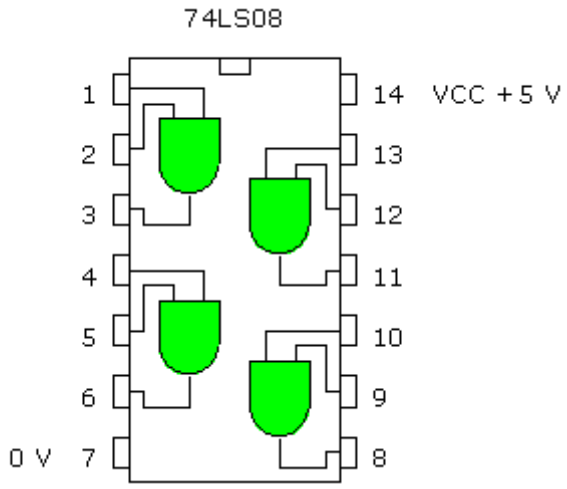
Figure 6.7 Alternate multiplexer implementation table and circuit

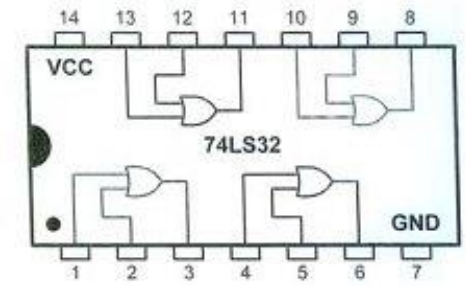
EXPERIMENTAL PROCEDURE

1) Implement 4x1 multiplexer by using minimum amount of logic gates.

Equipment List

1) 74LS32 TTL ***OR*** GATE IC
2) 74LS08 TTL ***AND*** GATE IC
3) Standard set equipments


74LS08
1
2
3
4
5
6
0 V 7
14 VCC +5 V
13
12
11
10
9
8


14 13 12 11 10 9 8
VCC
74LS32
GND
1 2 3 4 5 6 7