Lab # 4
Decoders and Multiplexers

Lab Format
- This is a **Individual Lab** so each student must design and test their own circuits.
- Students are free to assist each other in all labs.
- Each student must complete the Preliminary Work Section **before** lab begins. Preliminary Work will be checked in lab and will be part of the lab report grade.
- Each student must submit his or her own lab report.
- Lab reports will not be accepted until all required circuits have been demonstrated to the instructor.

A.   Objective
The objective of this laboratory is to investigate the design and use of decoders and multiplexers. Boolean functions will be implemented using both decoders and multiplexers.

B.   Materials
Breadboard
5V Power Supply
Wire, switches, etc.
74LS151 8 x 1 Data Selector (multiplexer)
74LS155 Dual 2 x 4 Decoder/Demultiplexer
74LS04 Hex Inverter
74LS08 Quad 2-input AND

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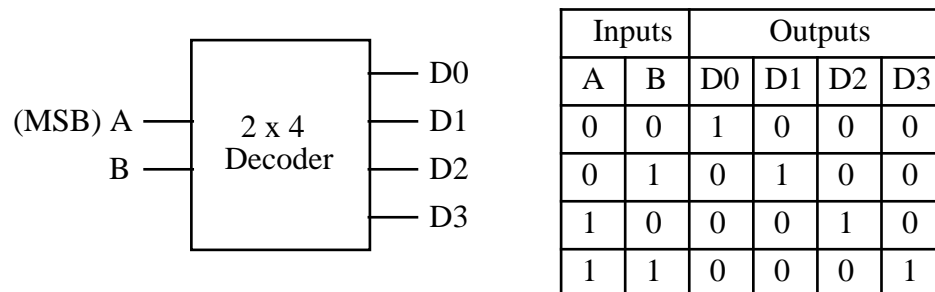
Inputs		Outputs			
A	B	D0	D1	D2	D3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

Figure 1: 2 x 4 decoder with active-HIGH inputs and outputs

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

> **_Active-HIGH decoder outputs are equivalent to minterms_**

For example, $D0 = m_0 = \overline{A} \bullet \overline{B}$ (for 2 inputs) or $D0 = m_0 = \overline{A} \bullet \overline{B} \bullet \overline{C} \bullet \overline{D}$ (for 4 inputs)

A combinational logic function that is expressed as a sum of minterms, therefore, can be implemented by summing decoder outputs, or

> **_$F = \sum(minterms) = \sum(active\text{-}HIGH\ decoder\ outputs)$_**

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.
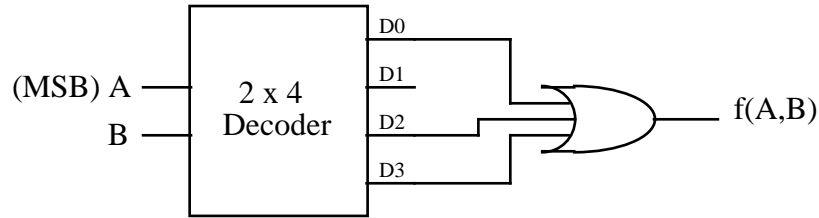

Figure 2: f(A,B) =Σ (0,2,3) implement using a 2 x 4 decoder

Some decoders, such as the 74155, have active-LOW outputs. Figure 3 shows a block diagram and a truth table for a 2 x 4 decoder with active-LOW outputs.
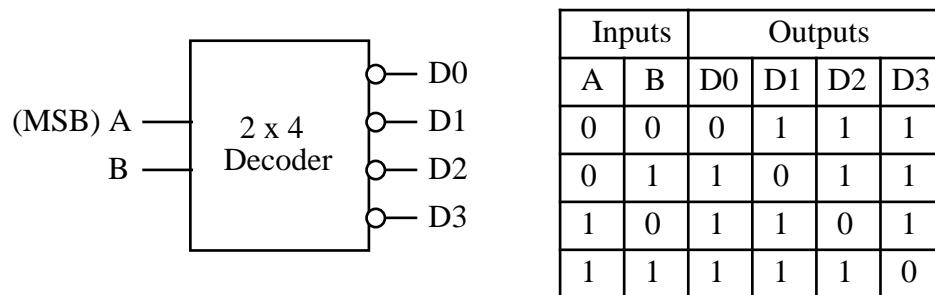

Inputs		Outputs			
A	B	D0	D1	D2	D3
0	0	0	1	1	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	1	0

Figure 3: 2 x 4 decoder with active-LOW inputs and outputs

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

> **_Active-LOW decoder outputs are equivalent 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, or

> **_$F = \prod(maxterms) = \prod(active\text{-}LOW\ decoder\ outputs)$_**

For example, if $f(A,B) = \Pi(0, 1, 3)$ then f (A,B)= D0 $\bullet$ D1 $\bullet$ D3 so f can be implemented by the circuit shown in Figure 4.
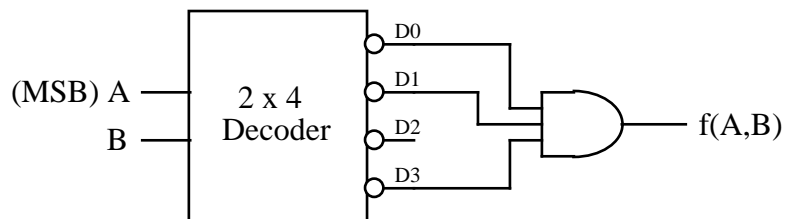

Figure 4: f(A,B) = Π(0,1,3) implement using a 2 x 4 decoder

2

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)

Procedure:
1) Draw the truth table
2) Determine which inputs will be connected to the select lines (note which is the MSB)
3) Express the output F in terms of the other input.
4) Draw the MUX logic diagram.

Example: Implement the function $f(A,B,C) = \Sigma(0, 3, 6, 7)$ using a 4 x 1 multiplexer. The multiplexer implementation table is shown below in Figure 5.

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

Connect A and B to select lines

Express F in terms of the other input (C)

MUX Input 0

$F = C'$ (so connect C' to MUX input 0)

$F = C$

$F = 0$

$F = 1$

Figure 5: MUX implementation table
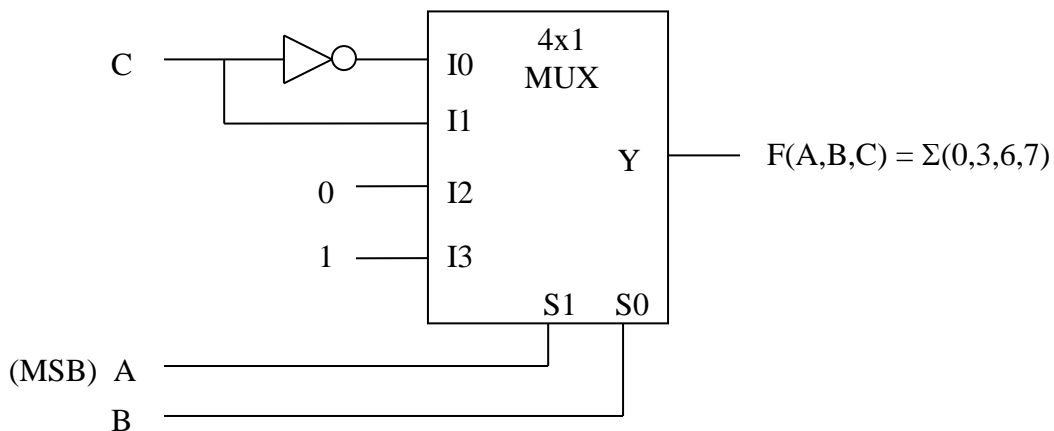
The circuit can be implemented as shown in Figure 6.

Figure 6: $F(A,B,C) = \Sigma(0,3,6,7)$ implemented using a 4x1 multiplexer
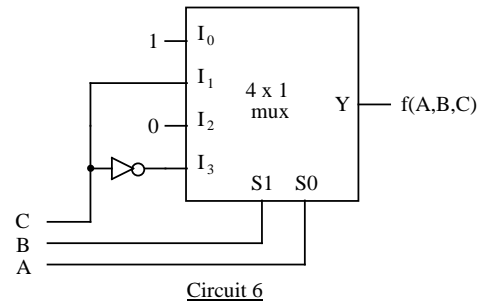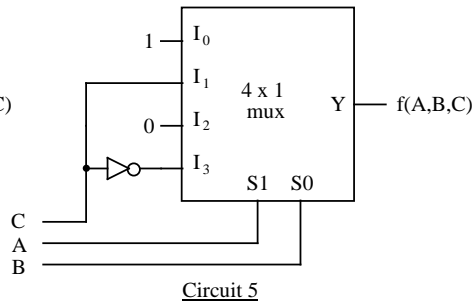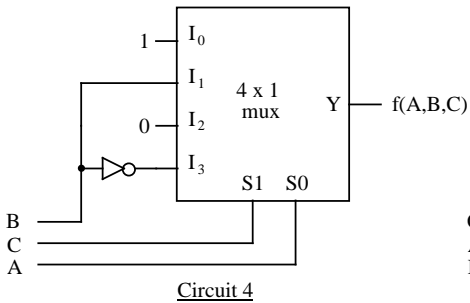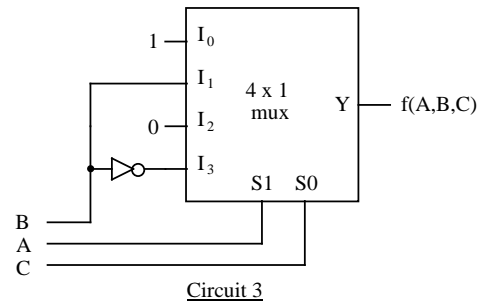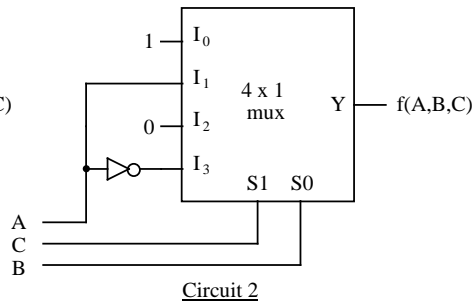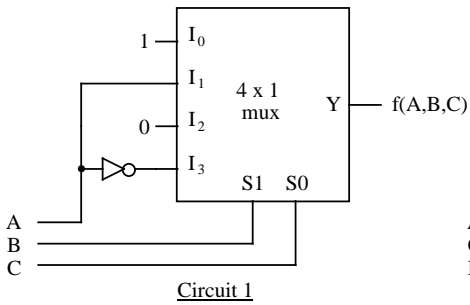
Keep in mind in the example above that bits A and B were connected to the select lines. If any other bits are connected to the select lines, then the implementation table needs to be rearranged.

3

D. **Preliminary Work**

Include instructions with each step.

1. **_3x8 Decoder Circuit_**: Function $f(A,B,C)$ is defined as follows: $f(A,B,C) = \Sigma(1^{st}$ 4 unique digits in your Student ID that are less than 8). If you do not have 4 digits that are less than 8, then add in as many of the following digits as are necessary: 7, 2, 6, 4. For example, if your Student ID is 1468413, then $f(A,B,C) = \Sigma(1,3,4,6)$. Express $f(A,B,C)$ as a sum of minterms and as a product of maxterms (shorthand notation).

2. **_Logic Diagram for 3x8 decoder circuit using PSPICE_**. Use PSPICE to generate a logic diagram that uses a 74155 3x8 decoder to implement $f(A,B,C)$ from the previous step. The logic diagram should include:
   - Display sssigned chip numbers and part numbers (For example: U1 – 74LS08)
   - If multiple gates are used, use as many as possible from a given IC (for example, use U1A, U1B, U1C, etc., rather than U1A, U2A, U3A, etc.)
   - Input switches (including DIP switches, resistors, etc)
   - Label all inputs and outputs. Label input A as the MSB.
   - Output LED, including current-limiting resistor
   - Include text on the schematic page with information such as your name, course number, lab number, and the function to be implemented.
   - **_Extra credit_** (5 points): If you simulate the circuit above using PSPICE, you can replace the dip switches with Digital Clocks. The output LED and current limiting resistor are also unnecessary. Be sure to use use the part LO if a 0 input is needed. The analog 0 ground cannot be used. Include the schematic and the output waveforms in your report.

3. **_8x1 Multiplexer Circuit_**. Design a multiplexer circuit using the 74151 to implement the function $f(A,B,C,D) = \Sigma(1^{st}$ 4 unique digits in your EmplID + minterms 10, 12, 13). In particular:
   - State $f(A,B,C,D)$ as a sum of minterms (shorthand notation) as described above.
   - Show the **_multiplexer implementation table_** to implement this function.

4. **_Logic Diagram for 8x1multiplexer circuit using PSPICE_**. Use PSPICE to generate a logic diagram that uses a 74151 8x1 multiplexer to implement $f(A,B,C,D)$ from the previous step. The logic diagram should include:
   - Display assigned chip numbers and part numbers (For example: U1 – 74LS04)
   - Input switches (including DIP switches, resistors, etc)
   - Label all inputs and outputs. Label input A as the MSB.
   - Output LED, including current-limiting resistor
   - Include text on the schematic page with information such as your name, course number, lab number, and the function to be implemented.
   - **_Extra credit_** (5 points): If you simulate the circuit above using PSPICE, you can replace the dip switches with Digital Clocks. The output LED and current limiting resistor are also unnecessary. Be sure to use use the part LO if a 0 input is needed. The analog 0 ground cannot be used. Include the schematic and the output waveforms in your report.

5. ***Exercise using 4x1 Multiplexers***. Draw the multiplexer implementation table or a truth table for each circuit below and determine the output function f(A,B,C) in sum of minterms (shorthand) form.

Circuit 1
1 — I_0
I_1 4 x 1
0 — I_2 mux   Y — f(A,B,C)
I_3
S1 S0
A
B
C
Circuit 1

Circuit 2
1 — I_0
I_1 4 x 1
0 — I_2 mux   Y — f(A,B,C)
I_3
S1 S0
A
C
B
Circuit 2

Circuit 3
1 — I_0
I_1 4 x 1
0 — I_2 mux   Y — f(A,B,C)
I_3
S1 S0
B
A
C
Circuit 3

Circuit 4
1 — I_0
I_1 4 x 1
0 — I_2 mux   Y — f(A,B,C)
I_3
S1 S0
B
C
A
Circuit 4

Circuit 5
1 — I_0
I_1 4 x 1
0 — I_2 mux   Y — f(A,B,C)
I_3
S1 S0
C
A
B
Circuit 5

Circuit 6
1 — I_0
I_1 4 x 1
0 — I_2 mux   Y — f(A,B,C)
I_3
S1 S0
C
B
A
Circuit 6

E. **Laboratory Work**

1. Construct the ***3x8 decoder circuit*** shown in the PSPICE logic diagram of step 2 in the Preliminary Work. Note any changes. Test the circuit to be sure that it produces the correct output for each input. Record the truth table. Demonstrate proper operation of the circuit to the instructor.

2. Construct the ***8x1 multiplexer circuit*** shown in the PSPICE logic diagram of step 4 in the Preliminary Work. Note any changes. Test the circuit to be sure that it produces the correct output for each input. Record the truth table. Demonstrate proper operation of the circuit to the instructor.

F. **Report**

Remember that each lab report should have the following four sections. Also see additional notes below.

Title Page
Preliminary Work

*   Include instructions

Lab Results

*   Include all measured results (truth tables).
*   Include step numbers and titles or headings that make it clear what is being shown.
*   Explain any changes that were made to the circuits as shown in the PSPICE schematics.

Discussion/Conclusion

*   Discuss each circuit tested in lab.
*   Compare the implementation of combinational logic circuits using basic logic gates (Labs 1 & 3) to implementing combinational logic circuits using decoders and multiplexers (Lab 4). Which method is easiest?
*   Explain the difference between implementing a function using a decoder with active-HIGH outputs and a decoder with active-LOW outputs.

*   What have you proven or demonstrated by completing the experiment?