EGR 270
Fundamentals of Computer Engineering
File: N270L4

Lab # 4
Decoders, Multiplexers, and 7-Segment Displays

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. Additionally, a 7-segment display will be driven using a commercially available BCD-to-7-segment decoders.

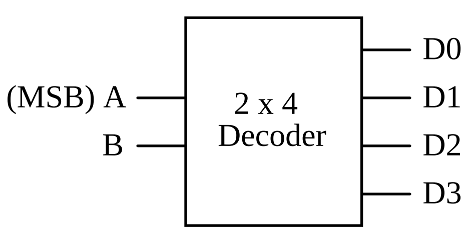
B. Materials

Breadboard
5V Power Supply
Wire, switches, etc.
Common-anode 7-segment display (MAN72 or other)
7447 BCD-to-7-segment decoder/driver (common anode)
74151 8 x 1 Data Selector (multiplexer)
74155 Dual 2 x 4 Decoder/Demultiplexer
Assorted AND, OR, NAND, NOR, XOR, and INVERTER IC's

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. 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.

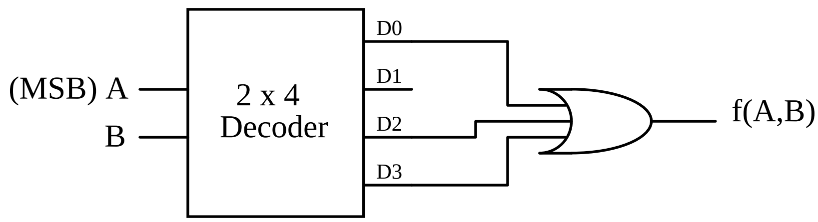
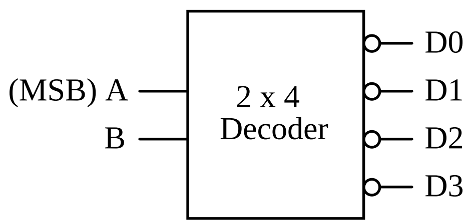


Figure 2: f(A,B) =Σ (0,2,3) implement using a 2 x 4 decoder

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. 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.

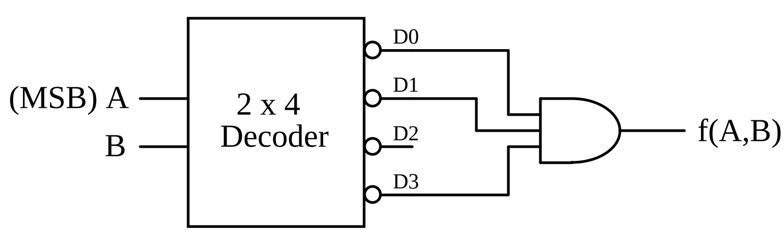


Figure 4: f(A,B) = Π(0,1,3) implement using a 2 x 4 decoder

BCD-to-7-segment decoders

One decoder of special interest is a BCD-to-7-segment decoder. Its purpose is to decode BCD inputs (the binary codes corresponding to the decimal values 0 - 9) in order to light the appropriate segments on a 7-segment display. The decoder, therefore, will need 7 outputs in order to control the 7 segments, which are labeled a through g. Diagrams of the decoder and a 7-segment display are shown below.

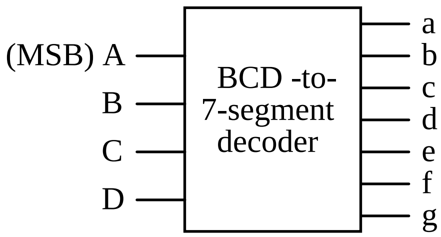
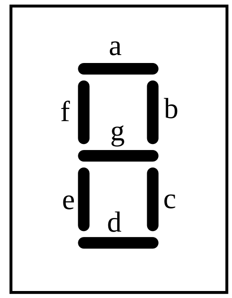


Figure 5: BCD to 7-segment decoder
(with active-HIGH inputs and outputs)



7-segment display

The decoder should function as follows:

if ABCD = 0000, the display should light the digit 0 (segments a, b, c, d, e, f)
if ABCD = 0001, the display should light the digit 1 (segments b, c)
if ABCD = 0010, the display should light the digit 2 (segments a, b, d, e, g)
.
if ABCD = 1001, the display should light the digit 9 (segments a, b, c, f, g)

Note that inputs ABCD = 1010 to ABCD = 1111 correspond to illegal inputs. The designer may wish to treat these inputs as “don’t cares” or perhaps generate a blank display or special unique symbols. If the illegal inputs are treated as “don’t cares”, the truth table would look as follows:

Inputs				**Outputs**						
A	**B**	**C**	**D**	**a**	**b**	**c**	**d**	**e**	**f**	**g**
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	0	0	1	1
1	0	1	0	x	x	x	x	x	x	x
1	0	1	1	x	x	x	x	x	x	x
1	1	0	0	x	x	x	x	x	x	x
1	1	0	1	x	x	x	x	x	x	x
1	1	1	0	x	x	x	x	x	x	x
1	1	1	1	x	x	x	x	x	x	x

Only 2 of the 7 required K-Maps for the 7 outputs are shown below.

AB \ CD	00	01	11	10
00	1	0	1	1
01	0	1	1	1
11	x	x	x	x
10	1	1	x	x

K-map for segment a

AB \ CD	00	01	11	10
00	1	1	1	1
01	1	0	1	0
11	x	x	x	x
10	1	1	x	x

K-map for segment b

The corresponding outputs are:

$$a = A + C + B \bullet D + \overline{B} \bullet \overline{D}$$

$$b = \overline{B} + C \bullet D + \overline{C} \bullet \overline{D}$$

Once the expressions for all 7 outputs are obtained, the circuit can be implemented.

There are several commercially available BCD-to-7-segment decoder IC's. The 7448 is such a device with active-HIGH outputs. The 7447 is a similar device with active-LOW outputs. Recall from earlier experiments that LEDs can be lit using either active-HIGH or active-LOW outputs. Thus, there are two types of 7-segment displays: common-cathode displays which require active-HIGH outputs to light the display and common-anode displays which require active-LOW outputs to light the display. Note that each of the 7 LEDs that make up a 7-segment display require a current-limiting resistor. The two types of displays and decoders are illustrated below.

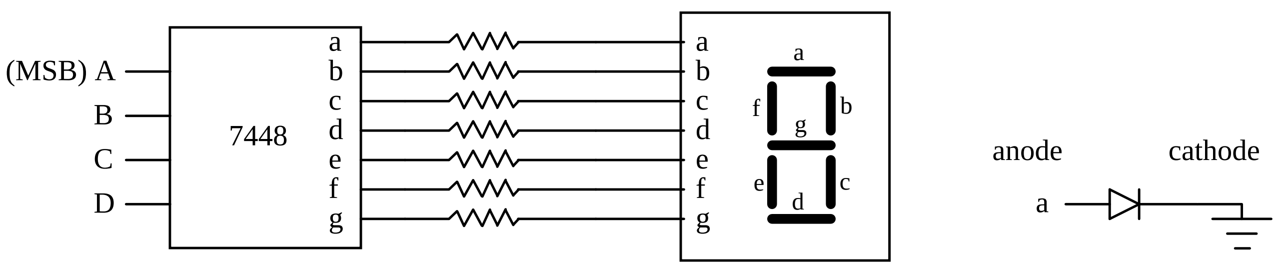


Figure 6: BCD to 7-segment decoder (with active-HIGH and outputs)

Common-cathode 7-segment display

Typical segment

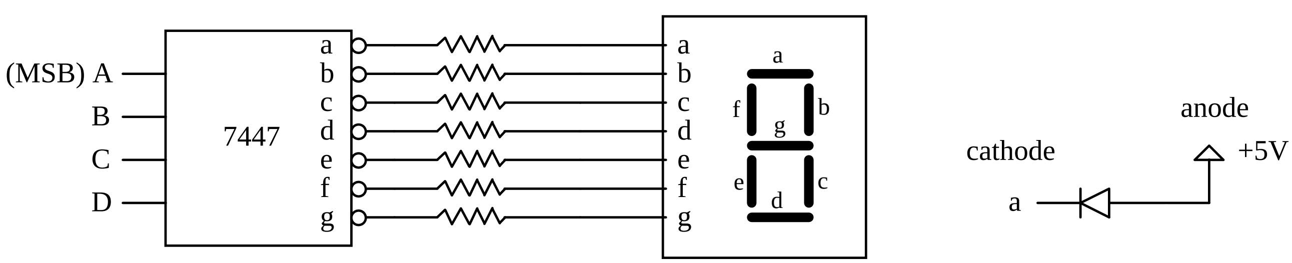


Figure 7: BCD to 7-segment decoder (with active-LOW outputs)

Common-anode 7-segment display

Typical segment

Note that the common-cathode display has all cathodes common (tied together) to ground. Thus, a HIGH input "forward biases" the diode and it emits light. (A diode is forward biased when a positive voltage is placed across it from anode to cathode.) The common-anode display, on the other hand, has all anodes common (tied together) to the supply voltage. Thus, a LOW input forward biases the diode and it emits light.

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) = Σ(0, 3, 6, 7) using a 4 x 1 multiplexer.
The multiplexer implementation table is shown below in Figure 5.

Connect A and B to select lines — Express F in terms of the other input (C)

A	B	C	F	
0	0	0	1	F = C' (so connect C' to MUX input 0) (MUX Input 0)
0	0	1	0	
0	1	0	0	F = C
0	1	1	1	
1	0	0	0	F = 0
1	0	1	0	
1	1	0	1	F = 1
1	1	1	1	

Figure 5: MUX implementation table

The circuit can be implemented as shown in Figure 6.

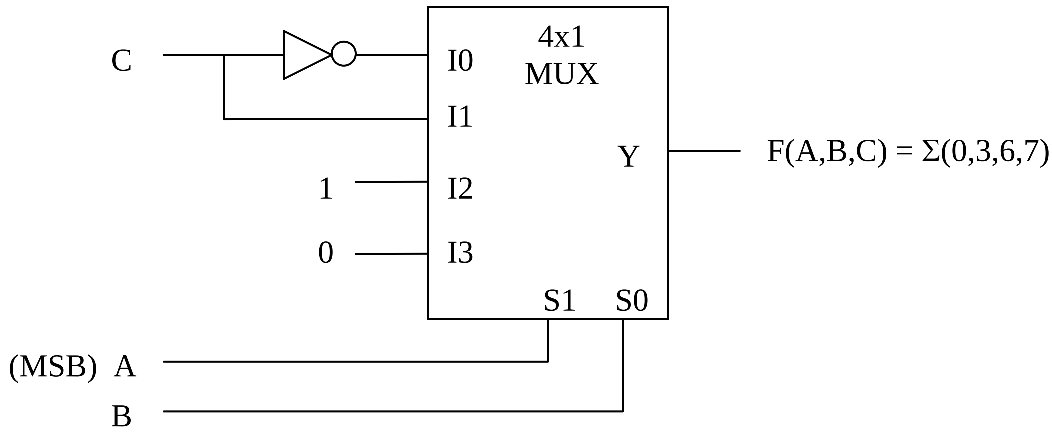


Figure 6: $F(A,B,C) = \Sigma(0,3,6,7)$ implemented using a 4x1 multiplexer

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.

Example: In class show the MUX implementation table for the example above if B and C are connected to the select lines.

D. **<u>Preliminary Work</u>**

1. Generate **full circuit documentation** (pinouts, IC diagram, and logic diagram) for a circuit consisting of a 7447 used to drive a common-anode 7-segment display. Include current-limiting resistors and input switches.

2. Check the data sheet for the 7447 to determine the function of $\overline{LT}$, $\overline{BI}$ / $\overline{RBO}$, $\overline{RBI}$. Write out a clear description of each.

3. Show how to connect four 7447 IC's (using a simple block diagram) such that leading zeros will not be displayed (for example, the displays will show (blank)607 rather than 0607).

4. Check the data sheet for the 7447 to determine the pattern that is lit on the 7-segment display for each of the illegal inputs. Illustrate the results with sketches.

5. Function f(A,B,C) is defined as follows: f(A,B,C) = Σ(1st 4 unique digits in your EmplID 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 EmplID is 1468413, then f(A,B,C) = Σ(1,3,4,6). Express f(A,B,C) as a sum of minterms and as a product of maxterms.

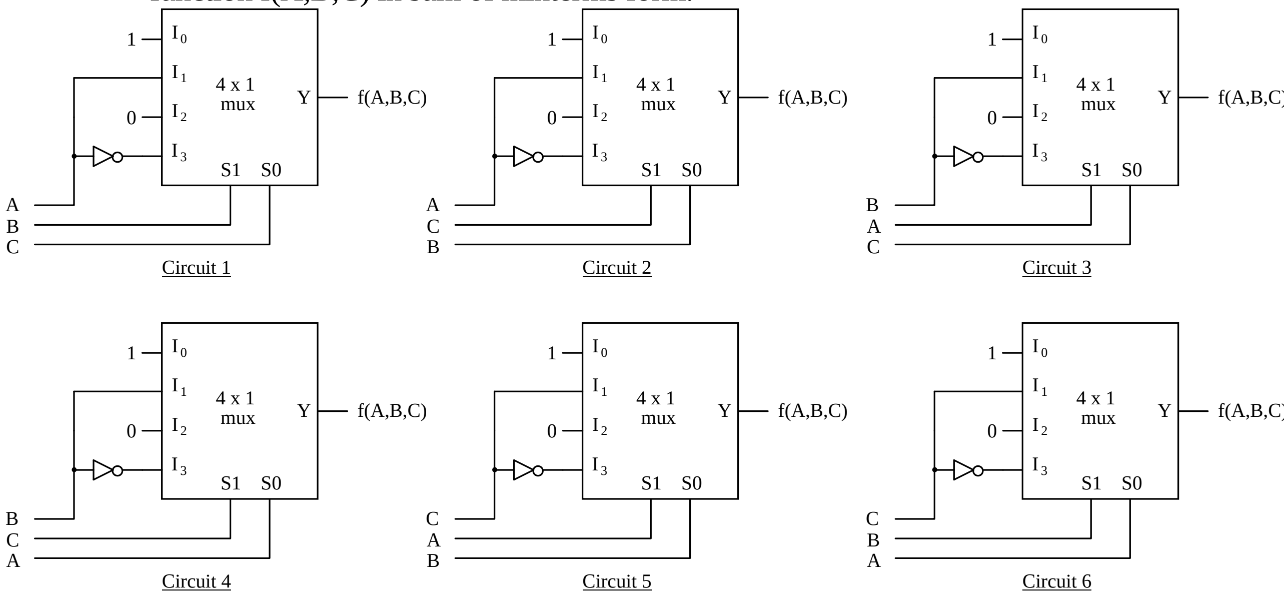
6. Design a decoder circuit using the 74155 configured as a 3 x 8 decoder to implement the product of maxterms expression shown above. Generate **full circuit documentation** for your designed circuit, including input switches and an LED for the output. Be sure to show how each enable line is to be connected on the 74155. Label input A as the MSB of the circuit diagram

7. The 74156 is similar to the 74155 except that it has open-collector outputs. Since open-collector outputs allow for wire-ANDing, draw the logic diagram for the designed circuit of step 6 using a 74156.

8. Design a multiplexer circuit using the 74151 to implement the function f(A,B,C,D) = Σ(1st 4 unique digits in your EmplID + minterms 10, 12, 13). Show the multiplexer implementation chart along with **full circuit documentation** for your designed circuit, including input switches and an LED for the output. Label input A as the MSB of the circuit diagram.

(continued)

9. Draw the multiplexer implementation table for each circuit below and determine the output function f(A,B,C) in sum of minterms form.



E. **Laboratory Work**

1. Construct the circuit of step 1 in the Preliminary Work according to the logic diagram generated. Note any changes. Test the circuit for all 16 possible input switch combinations to verify proper operation.

2. Perform further tests on the circuit of step 1 as follows:
   a) Test the $\overline{LT}$ input. Verify that it works as expected.
   b) Set $\overline{RBI}$ LOW and display a 0 on the 7-segment display. Check the value of $\overline{BI}/\overline{RBO}$. Set $\overline{RBI}$ LOW and display a 3 on the 7-segment display. Check the value of $\overline{BI}/\overline{RBO}$. Explain the results in your report.
   c) Display a 0 on the 7-segment display. Switch the $\overline{RBI}$ input from HIGH to LOW and note the effect. Explain the results in your report.

3. Construct the decoder circuit of step 6 in the Preliminary Work according to the logic diagram generated. 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.

4. Construct the multiplexer circuit of step 8 in the Preliminary Work according to the logic diagram generated. 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.