Rev. 8/26/ 2009

ELEC 2210 - EXPERIMENT 3
Medium Scale Integrated (MSI) Circuits

The objectives of this Experiment:

In this lab you will learn to work with some simple MSI (medium scale integration) logic circuits. You will also be introduced to some of the circuits used to develop computer components (communications switches, read only memory, etc.). The objectives of this experiment include:

- Learn to use decoders / demultiplexers /output selectors,
- Learn to use multiplexers / data selectors,
- Learn how to connect an LED to a TTL output,
- Continue to build experience with digital simulation,
- Continue to develop professional communication skills.

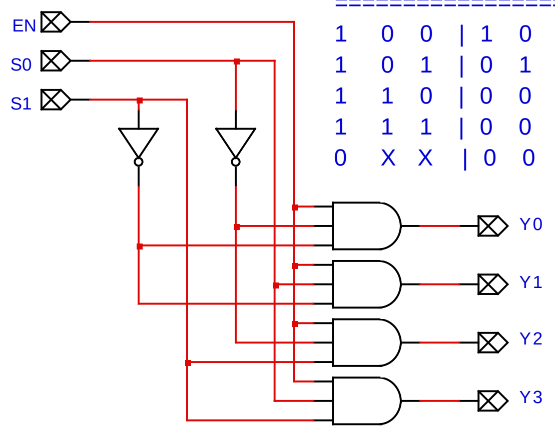
I. Introduction

Decoders/demultiplexers:

Digital systems often have sets of devices of which at most one should be active. For example, when a computer reads from memory chips, exactly one memory location needs to be chosen and "activated."

A *decoder* is a switch used to "turn on" exactly one item out of a set. A decoder made from four 3-input AND-gates is shown in Figure 1. The input signals are an enable signal *EN* and two select lines, *S0* and *S1*. Since the enable signal is directly wired to all four AND gates, when *EN=0* the outputs are all inactive (0). On the other hand, when *EN=1*, the select lines are connected such that exactly one gate is chosen to be ON based on the value of *S0* and *S1*. For example, look at the AND gate connected to *Y1*. Its top input (like all the other gates) is connected to *EN*. The middle input is connected directly to *S0*, and the bottom input is connected to *S1* through an inverter. Thus, $Y1=\overline{S1}\bullet S0\bullet EN$. (The dots indicate logical AND function.). That is, *Y1* is active when (*S1*,*S0*) = (0,1) (binary representation of the number 1) and *EN* is active. The remaining gates are connected in a similar fashion so that they are active when the appropriate number is put on the select lines *S0*, *S1*.

The decoder design shown in Figure 1 is called a 2-to-4 (2:4) decoder because it has two select lines and 2^2 = 4 output lines. Decoders are usually referred to by size in this fashion (n to 2^n decoders).



EN	S1	S0	Y0	Y1	Y2	Y3
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1
0	X	X	0	0	0	0

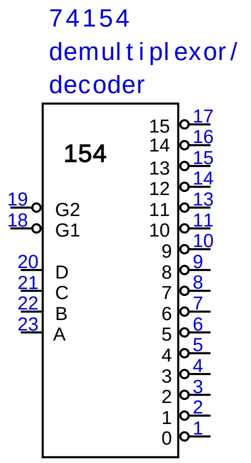


Figure 1. 2-to-4 decoder Figure 2: 74154 demultiplexer/decoder

Because decoders are used so often in digital systems, TTL manufacturers sell pre-packaged decoders such as the 74154 shown in Figure 2. Notice that the 74154 is a 4 to 16 decoder. The inputs are *D,C,B,A* where *D* is the "most significant bit" (i.e., *D*=1 means that one of lines 8,9,...,15 are selected). Notice also that the 75154 has two enable lines, $\overline{G1}$ and $\overline{G2}$. These inputs are connected inside the '154 so that *both* inputs must be active (LOW) for the chip to operate. That is, the chip is only active when G1 = 0 and G2 = 0.

Notice that the outputs of the 74154 are *active low*; that is, unlike the 2-4 decoder example above, the 74154 is built with NAND gates so that its active output is a 0 and all inactive outputs are 1's. (In all digital circuits, it is important to check whether specific outputs or inputs are active high or active low. Active low I/O will normally be indicated with an open circle, such as pins 1-19 on the '154 in Fig. 2.)

Read Only Memory (ROM):

One use for decoders is read-only memory (look-up table) design. Read only memory chips are commonly used for applications where a computer needs to save data values even when power is turned off. What can be done is to selectively OR the outputs of the decoder together so that the desired values appear on the outputs. For example, suppose that a lookup table was desired that took on the values shown below.

Table 1 – ROM Specification Example.

INPUTS		**Decoder Outputs (Active HIGH)**				**Desired ROM Outputs**	
S1	*S0*	*Q3*	*Q2*	*Q1*	*Q0*	*A1*	*A0*
0	0	0	0	0	1	1	1
0	1	0	0	1	0	0	1
1	0	0	1	0	0	0	1
1	1	1	0	0	0	1	0

For this example, our ROM lookup table has four entries. Each row in Table 1 is one entry. The goal is to output (A1, A0) when the corresponding row is selected by (S1, S0). This can done with the circuit shown in Figure 3. Since an OR gate only needs one input to be active in order for the output to turn on, the circuit is built by simply connecting a wire corresponding to each 1 in the table. For example, when *S1=1 and S0=1*, Q3 becomes active (Q3=1) and all other lines are zeros. Thus A1 = 1, as desired.

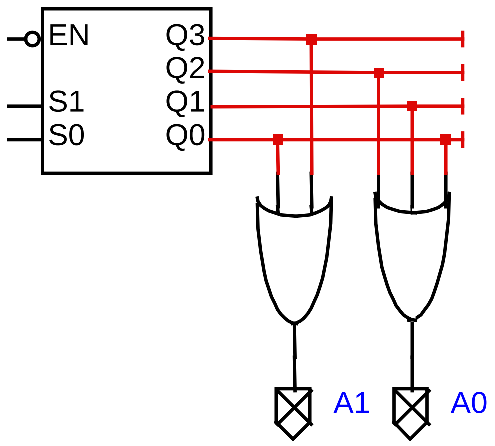


Figure 3: ROM example

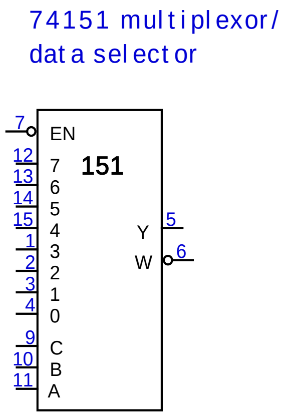


Figure 4. 74151 multiplexer/data selector

Multiplexers:

Digital circuits often use a single wire to connect several different items together. For example, the CPU in a computer is connected through a single bundle of wires to its memory chips, CD-ROM, disk drive, etc. In order to control who gets to communicate with the CPU, it is necessary to have a "switchboard" to manage the connections. A *multiplexer* (also called a *data selector*) is used as a switch to connect several inputs lines to a single output line. For example, the 74151 in Figure 4 can be used to switch between 8 inputs, numbered 0-7, that can be connected to the output Y (W is the inverted output). The multiplexer has a group of data select lines (A, B, C) that are used to choose which input line is connected to the output. An additional input line, $\overline{EN}$, is used to enable/disable the chip. When it is active (=0) then the multiplexer connects the output *Y to* the input line indicated by the select lines. When it is inactive (=1), then *Y*=0, regardless of the other input values. The select lines are used as a 3 bit binary number to specify which of the input lines is connected to the output *Y*. C is the most significant bit, followed by *B*, and *A* is the least significant select bit. Thus, *CBA*=110 means that input line 6 is connected to the output *Y*. Multiplexers are referred to by size as "*n* to 1" multiplexers, where $n=2^j$ is a power of two. For example, the 74151 above is an 8 to 1 (or just 8-1) multiplexer. A multiplexer always has $\log_2(n)$ select lines; the 74151 has $\log_2(8) = 3$ select lines. A 4-1 multiplexer has two select lines; a 16-1 multiplexer has 4 select lines.

Demultiplexers

A demultiplexer performs the inverse operation of a multiplexer: it takes a single input line and connects it to exactly one output line, all other output lines being inactive. If you take a look at the 2-4 decoder in Fig. 1, you'll see that it also serves as a 1-4 demultiplexer when you use the EN line as the input signal. For example, suppose *S1*=1 and *S0*=0 so that output line *Y2* is selected and all other outputs are inactive. If *EN*=1, then output *Y2* is also 1. Conversely, if *EN*=0, then so is *Y2*. In other words, a demultiplexer and a decoder are exactly the same thing.

Look back at the 74154 4-16 active low decoder chip in Figure 2. This chip has two enable lines, not just one enable line. The reason for this is simple: if the chip is used as a demultiplexer, one of the enable inputs is used for the signal input while the other is used as an enable. On the other hand, if the chip is used as a decoder, the two inputs can just be wired together to act as a single enable.

Encoders:
An encoder outputs a binary number indicating which of its inputs is active; if more than one input is active, then the output depends on the priority that the encoder was designed to assign to its input lines. (Encoders will not be used in this experiment.)

Prelab

1. Design a decoder-based ROM similar to Fig. 3 for the look-up table below.

INPUTS		**OUTPUTS**	
S1	*S0*	*A1*	*A0*
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

2. Obtain data sheets for the 74151 multiplexer and the 74154 demultiplexer from the internet. Any manufacturer or variety is acceptable, but your data sheets should include the pin-out for the DIP packages (16-pin for the ‘151, 24-pin for the ‘154). You should keep these data sheets for use in the experiment after your instructor verifies that you have them. Helpful tip: the datasheets used in this writeup are (as of 1/25/2009) at:
   http://www.national.com/ds/54/54LS151.pdf
   http://www.national.com/ds/DM/DM54LS154.pdf

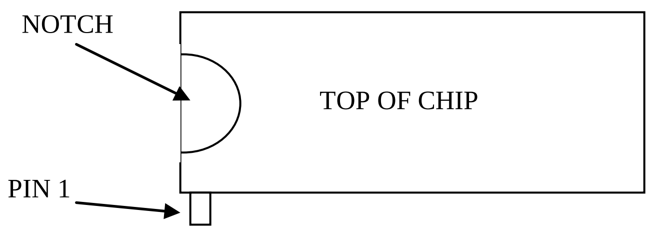
3. Construct a pin connection list similar to Table 2 to guide your testing of the 74151.

4. Use LogicWorks to connect and test the 74151. Print out your LogicWorks circuit and include it as part of your pre-lab.

Lab Exercise:
You will use the following equipment and components:

- Bit Bucket digital logic breadboard system
- Digital Multimeter
- The usual assortment of 16 banana-plug cables, and a tub of hookup wires.
- If any of these are missing or non-functional, let your lab instructor know.
- Other components listed in the table below.

Qty	Part # or value	Description	Instructions
1	330 Ω resistor	orange-orange-brown	Measure and record the value
1	LED	red	Identify the anode and cathode, as instructed by your GTA.
1	74151, 16-pin DIP	8-input multiplexer (MUX), TTL	Identify pin 1 (see drawing)
1	74154, 24-pin wide DIP	4–16 decoder, TTL	Identify pin 1 (see drawing)



(1) Connect the 74154 decoder and verify correct operation.

The pinout diagram is shown in Figure 5.

(a) Make the connections listed in Table 2.

(b) Verify the correct operation of the 74154 by setting the input bits DCBA to at least three different values of your choosing, and confirming that the correct output is active. To confirm an output is active, connect it to the logic probe or to any convenient LED in the bit bucket display area. Then you should see it respond when you press the pushbutton connected to G2. Record your input values and the observed outputs in a table in your report.

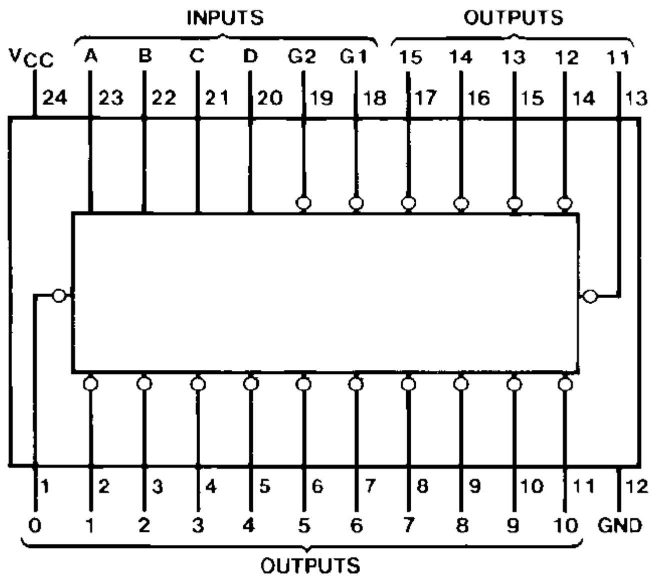


Figure 5. Pinout Diagram for the 74154 4–16 decoder.

(2) Connect an LED to a TTL output.

In this step, you will learn how to connect an LED to a TTL output. In order to limit the LED current and to protect the chip, it is necessary to put a resistor in series with an LED. For a standard TTL output, it is desirable to limit the current to approximately 8 mA. For a standard red LED, this can be accomplished by using a 330 Ω resistor as shown in Figure 6.

(a) Set the inputs *DCBA* to 0000 to activate output 0. Verify that this output is active as described in Step 3.

(b) Obtain a resistor marked 330 ohms (orange-orange-brown). Measure the actual resistance with the DMM, and record the value. Obtain a red LED.

(c) Turn off the power, and connect your LED and resistor to output 0 as shown in Fig. 6. Also connect output 0 to the logic probe (bit-bucket built-in LED).

(d) Turn on the power, and verify that your LED comes on when output 0 is low, and turns off when output 0 is high. If not, perhaps your LED is in backwards.

(e) Using the DMM, measure the voltage across the resistor when the LED is on. Using Ohm's law, calculate the LED current. Compare your calculation with the desired value of approximately 8mA stated earlier.

(f) Using the DMM, measure the output high and output low voltages on at least three different outputs. Record the output voltages both with and without your LED/resistor connected. Compare your results to the manufacturer's specifications in the data sheet and comment.

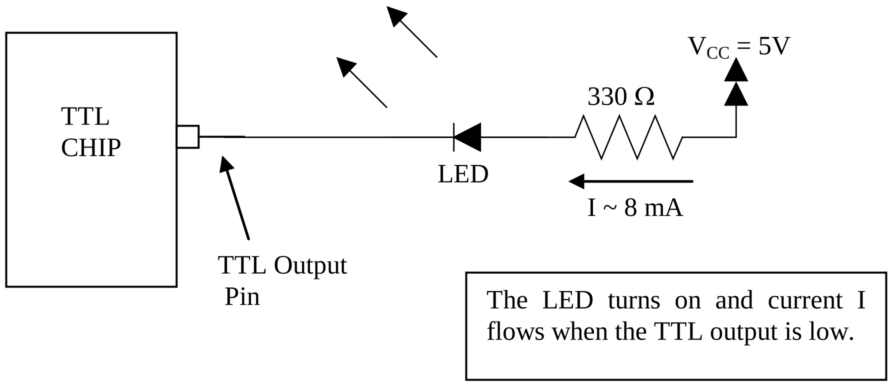

TTL CHIP
TTL Output Pin
LED
V_{CC} = 5V
330 Ω
I ~ 8 mA
The LED turns on and current I flows when the TTL output is low.

Figure 6. Connecting an LED to a TTL output.

Table 2. Connection list for the 74154. Active LOW I/O pins are shown with an asterisk (*)

Pin	Label	Function	Connect to	Notes
24	VCC	power	+5 V	
12	GND	ground	ground	
18	G1*	chip is enabled when G1 and G2 are low	ground	
19	G2*		Pulse switch $\overline{A}$	This will be "data" input.
20	D	MSB of 4-bit select input	Data switch SW3	
21	C	select input	Data switch SW2	
22	B	select input	Data switch SW1	
23	A	LSB of 4-bit select input	Data switch SW0	
Pins 1–11, 13–17	Outputs 0–10, 11–15	outputs	As instructed in the procedure	All outputs are active low.

(3) Connect the 74151 8-input Multiplexer.

The pin-out diagram and pin descriptions for the 74151 are shown in Fig. 7

(a) Follow the LogicWorks circuit and connection table from your pre-lab to connect the 74151.

(b) To verify correct operation, select at least 3 inputs using A-B-C and observe that they are transmitted to the output Y.

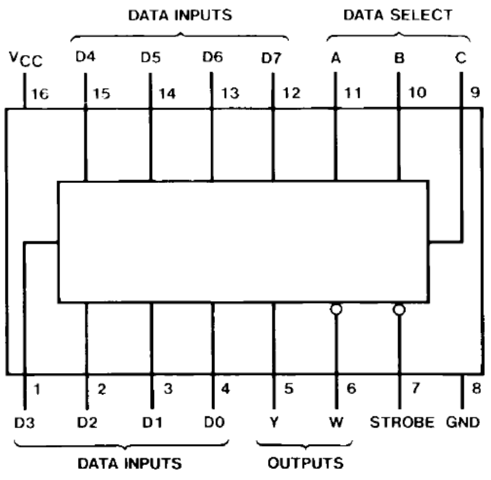


Figure 7. Pin-out and pin descriptions for the 74151 8-input multiplexer.

(4) Cleanup

DO NOT PUT RESISTORS, CAPACITORS, CHIPS, OR ANY OTHER COMPONENTS IN THE WIRE TUBS.

(a) Turn off the power to the Bit Bucket.

(b) Turn off the DMM.

(c) Disassemble your circuit and place all wires back in the wire tub.

(d) Put all chips and other components back in the proper bins.

(e) Clean up your workstation and discard any trash.