

EXPERIMENT 8

Medium Scale Integration (MSI) Logic Circuits

Introduction

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.).

Experiment Objectives:

- Learn to build and use decoders / demultiplexers /output selectors,
- Learn to build and use multiplexers / data selectors,
- Continue to build experience with the Bit Bucket digital breadboarding system and LogicWorks digital simulation,
- Develop professional communication skills, including accessing data using the internet.

Bring to Lab:

Your completed Pre-Lab. Turn this in when you get to lab.

Theory: MSI Logic Circuits

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, $Y1 = \overline{S1} \cdot S0 \cdot EN$. (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).

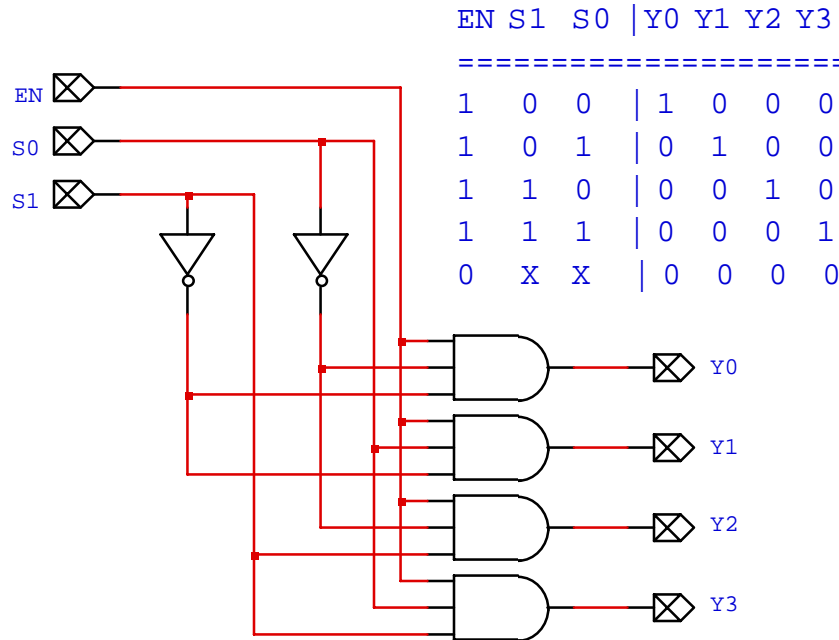


Figure 1. 2 to 4 decoder

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 74154 has two enable lines, $\overline{G1}$ and $\overline{G2}$. These inputs are connected inside the '154 so that *both* inputs must be active for the chip to operate. That is, the chip is only active when $\overline{G1} = \overline{G2} = 0$.

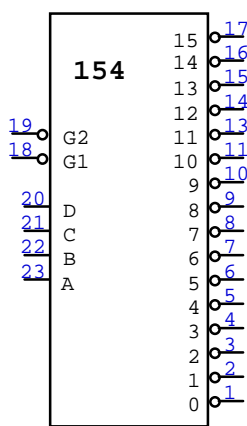


Figure 2: 74154 demultiplexer/decoder

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 in Table 1.

Table 1. Decoder-base ROM lookup table

INPUTS		OUTPUTS	
<i>S1</i>	<i>S0</i>	<i>A1</i>	<i>A0</i>
0	0	1	1
0	1	0	1
1	0	0	1
1	1	1	0

For this example, our ROM lookup table has four entries. Each row in Table 1 is one entry, ($n=2$ select lines means you can select one of $2^n=4$ rows) with two bits per entry. The goal is to output (*A1*, *A0*) when the corresponding row is selected by (*S1*, *S0*). This can be 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 $S1=S0=1$, *Q3* becomes active ($Q3=1$) and all other lines are zeros.

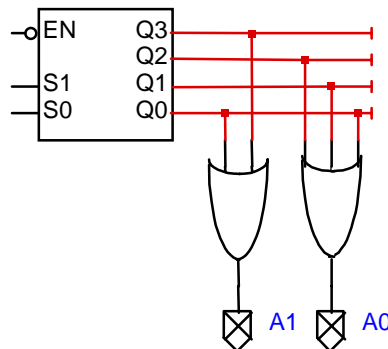


Figure 3: ROM example

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.

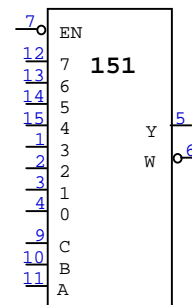


Figure 4. 74151 multiplexer/data selector

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.)

Your Name _____

Prelab Questions (10 points)

Answer these questions before coming to lab and turn them in when you arrive. You may do your work on separate paper (for example you might want to do your work on a computer), but please attach your work to this sheet for submission.

- Use inverters and AND gates to design a 3-8 decoder with active high inputs. Draw your design neatly (use a straight-edge). Instead of drawing by hand, you may wish to use LogicWorks and print out the circuit after you verify that it works correctly.
- Design a decoder-based ROM similar to Fig. 3 for the look-up table below.

INPUTS		OUTPUTS	
<i>SI</i>	<i>SO</i>	<i>AI</i>	<i>AO</i>
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

- 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: Go to the Texas Instruments website <http://www.ti.com> and search for *74*151 and *74*154. The '*'s are wildcard characters that can stand for any sequence of characters. Not all the online data sheets have the same quality. Some of the older ones are almost illegible on the screen. The data sheets for TI's HC, AS, and ALS families are much better than the older S or LS.