

CHAPTER

9

MEDIUM SCALE INTEGRATION

In this chapter you will study medium scale integrated circuits. As mentioned previously, **MSI devices have between 12 and 100 transistors on a single piece of silicon.** You will study several commonly available types of MSI devices and some applications that would normally utilize MSI circuits.

9.0 INTRODUCTION

Upon completion of this chapter you should be able to:

9.1 OBJECTIVES

- Explain and **define MSI circuits.**
- Name **several common types of MSI circuits.**
- Know when to **use MSI circuit** vs other types of logic circuits.

A large number of devices are available as MSI logic circuits. These circuits are constructed using TTL or CMOS techniques. The logic devices used to this point have been SSI devices. SSI is fine for the simple circuits you have constructed, but MSI is frequently used for the more complex logic equations.

9.2 DISCUSSION

Alternatives to the use of MSI are the use of a read only memory (ROM) or programmable logic arrays (PLAs). This chapter will study some typical MSI devices and their applications.

9.2.0 Decoders

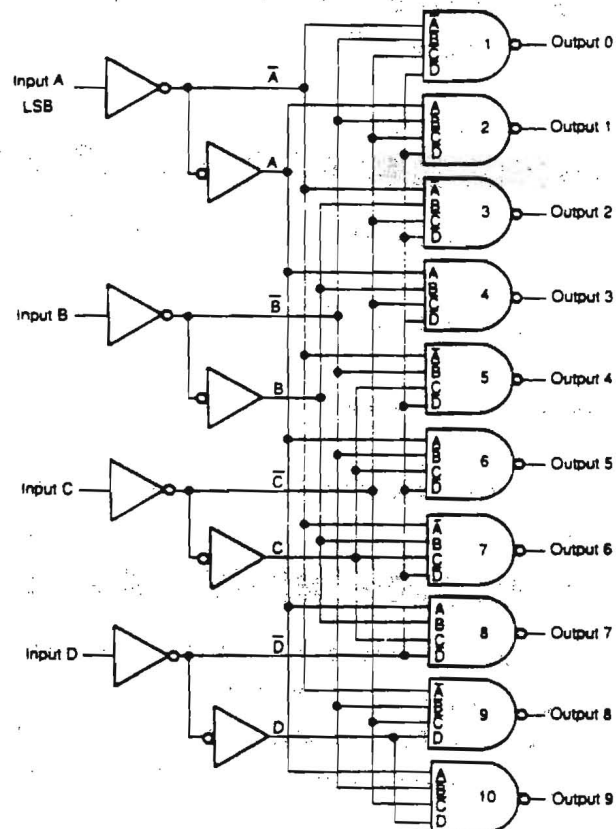
You may recall from your earlier studies that the basic circuit for decoding is the AND gate. The decoder function is to detect the presence of a particular binary number or word. The decoder input is a multi-bit binary word and the output is a binary line that indicates the presence or absence of a specific binary word.

A variety of types of decoders are available as TTL circuits. Two types of these decoders are studied in the following paragraphs. All of the TTL decoders work in the same manner as the devices you will examine.

9.2.1 BCD-to-Decimal Decoder

The BCD-to-decimal decoder is formed from ten NAND gates along with some additional input gating. Figure 9-1 has a logic diagram for the BCD-to-decimal decoder.

FIGURE 9-1. BCD to Decimal Decoder Logic Diagram.

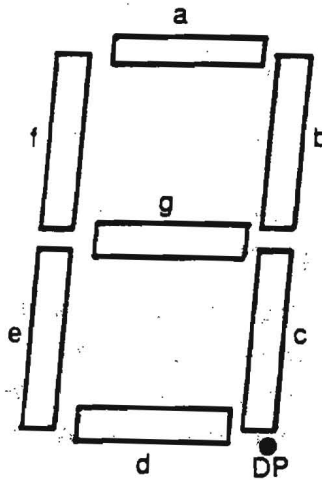


The MSI component that implements the BCD-to-decimal decode function is the 7442. This device is also known as a four-to-ten line decoder. Select one of the possible output states and convince yourself that the output will only be true for a specific set of input circumstances.

Notice that not all of the possible input states are decoded. Decoding all input combinations would result in erroneous decoder output should a number which is not between zero and nine appear on the input.

Another type of common decoder takes a BCD input and converts it to an output which will drive a seven segment display. You should already be familiar with seven segment displays from the laboratories using the LD-2.

A seven segment display is made from seven LED segments. These segments are segments a-g and are lettered clockwise from the top segment of the display with the g segment being the middle segment as shown in Figure 9-2.

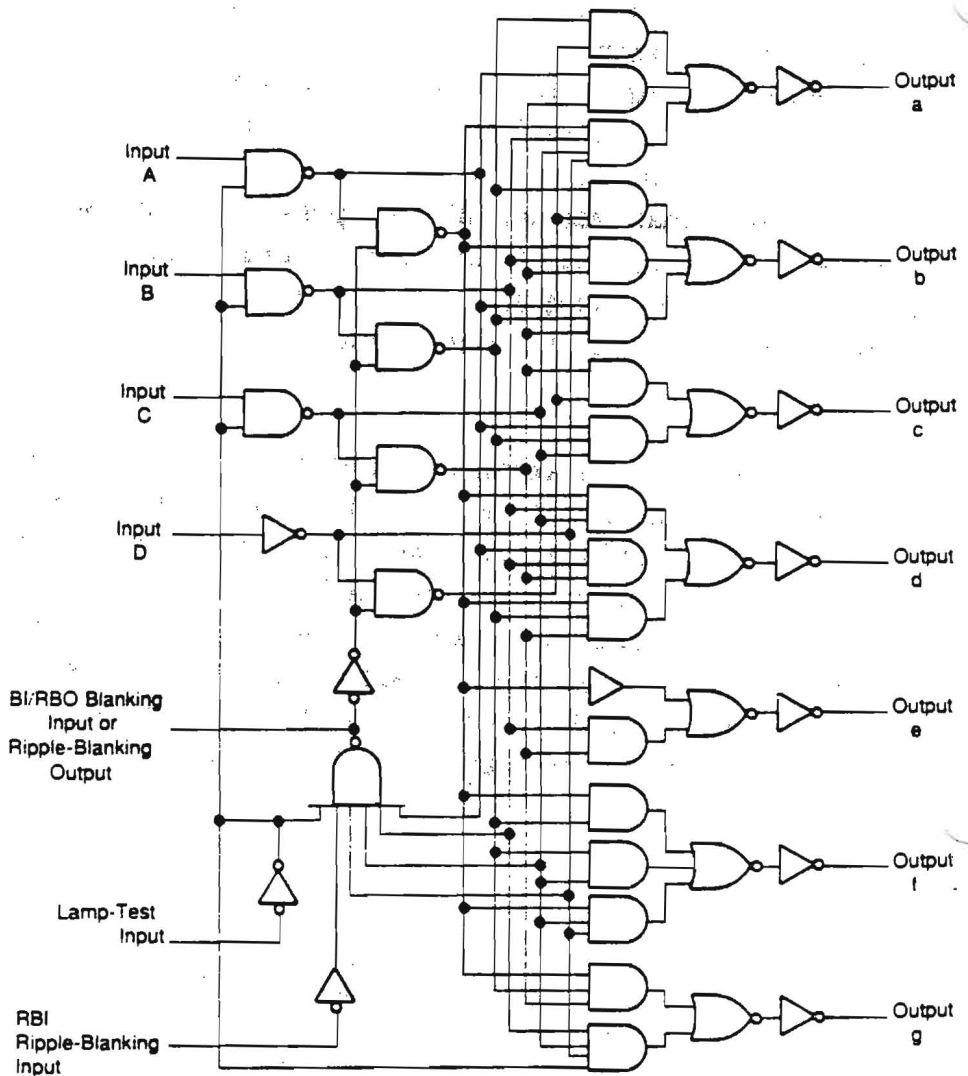


9.2.2 BCD-to-Seven Segment Display Decoders

FIGURE 9-2. Seven-Segment Display Labeling.

The decoder needed to perform this function has four input lines and seven output lines. A logic diagram for a BCD-to-seven segment display decoder is shown in Figure 9-3.

FIGURE 9-3. Logic Diagram BCD to Seven Segment Decoder-Driver IC.



The BCD-to-seven segment decoder function is performed by the 7447 TTL IC.

This device also provide open collector output drivers so that no additional circuitry is needed to drive a display. Again notice that this device does not decode all of the input combinations and in fact decodes even fewer cases than the BCD-to-decimal decoder.

9.2.3 Common Displays

Two types of simple displays are widely available for use with digital devices. One of these displays is the seven segment display made from LEDs. This type of display is used on the LF 2 logic designer. It is readily made compatible with TTL logic circuits. An LED functions much like any other diode except

that when it conducts sufficient current, the device will emit light. The two methods of driving an LED with a gate output are to:

1. Connect the gate output to the anode of the LED and the cathode of the LED to ground through a resistor. The LED will light when the gate output is HI. This is known as current sourcing operation. This type of operation is not the preferred method of biasing an LED to emit light when using TTL devices.
2. Connect the gate output to the cathode of the LED and the anode of the LED to Vcc through a resistor. In this configuration, the LED will light when the gate output is LO. This is the current sinking mode of operation. Current sinking operation is preferred since most TTL gates can sink more current than they can source. Note that when a gate is used to drive a LED that the current requirements of the LED are so high that no other TTL device can be driven from the output of the gate driving the LED.

The different ways of driving LEDs have resulted in two similar configurations of seven segment displays. The Common anode devices are made to be driven by current sinking logic. The Common cathode circuit is for use with current sourcing logic. LED displays can be driven from latches or can be multiplexed so that one decoder/driver can supply the needs of several digits of a display. When LED displays are multiplexed, drive current is modulated to the displays which can result in brightness problems if the drive pulses are too short.

The second type of display which is frequently encountered is the Liquid Crystal Display or LCD. This type of display is made from crystals which will change their light transmittance properties in the presence of an electric field. This means that the LCD requires a light source to be visible. Most LCDs receive enough room light to give good readability. Those that do not get enough room light for good readability often have a backlight which provides good readability in reduced ambient lighting.

The LCD works by changing the light's polarization. The LCD is constructed from two pieces of glass which sandwich the

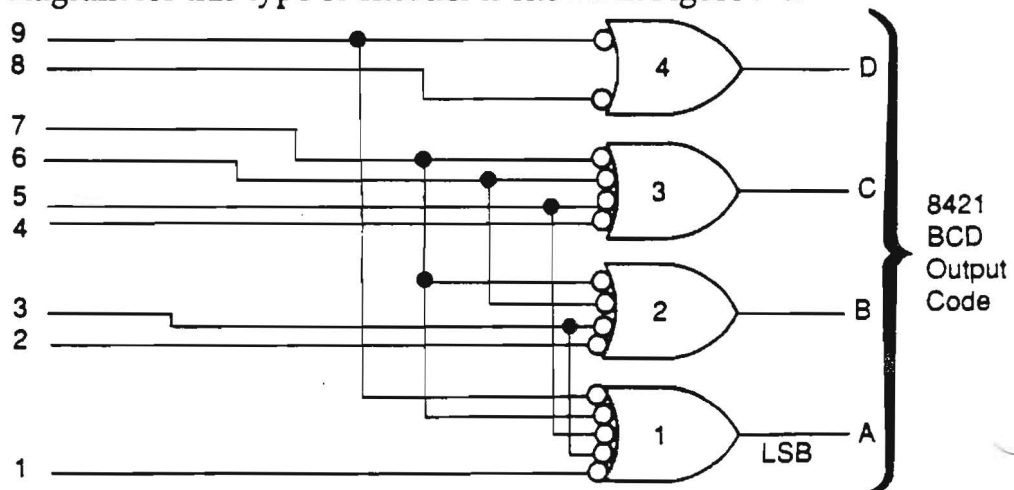
liquid crystal material. One of the glass plates is horizontally polarized to light and the other is vertically polarized. The LC material rotates the polarization of the incoming light by 90 degrees. This means that the incoming vertically polarized light is rotated in polarization by 90 degrees by the LC material and passed unobstructed through the horizontally polarized glass plate. When an electric field is applied the structure of the LC changes and the material will no longer rotate the light polarization. The vertically polarized light passing through the LC is blocked by the horizontal polarization of the other glass plate resulting in a dark area on the display. The LCD is more difficult to drive than the LED display. Several AC voltages with fairly strict timing requirements must be generated in order to drive the display. CMOS circuitry works well for driving LCD displays because of the wide range of supply voltages from which CMOS can be operated.

Complex LCD displays can best be driven by dedicated display driver circuits. One example of this type of circuit is the 4543B BCD-to-seven segment latch/decoder/driver for liquid crystals. This circuit will also drive LEDs, incandescent lamps, gas discharge displays and fluorescent displays.

9.2.4 Encoders

Encoders are circuits that accept one or more inputs and generate a multi-bit binary output. Decoders perform the opposite function from encoders in that decoders detect specific binary codes while encoders generate specific binary codes. Encoders are formed from NAND gates. One common decoder will convert from a decimal input to a BCD output. A logic diagram for this type of encoder is shown in Figure 9-4.

FIGURE 9-4. Decimal to BCD Encoder.



This type of encoder would be used to convert from a decimal keypad like a ten-key to a BCD output.

Another common type of encoder is the eight line to three line priority encoder. The logic diagram for this device is shown in Figure 9-5.

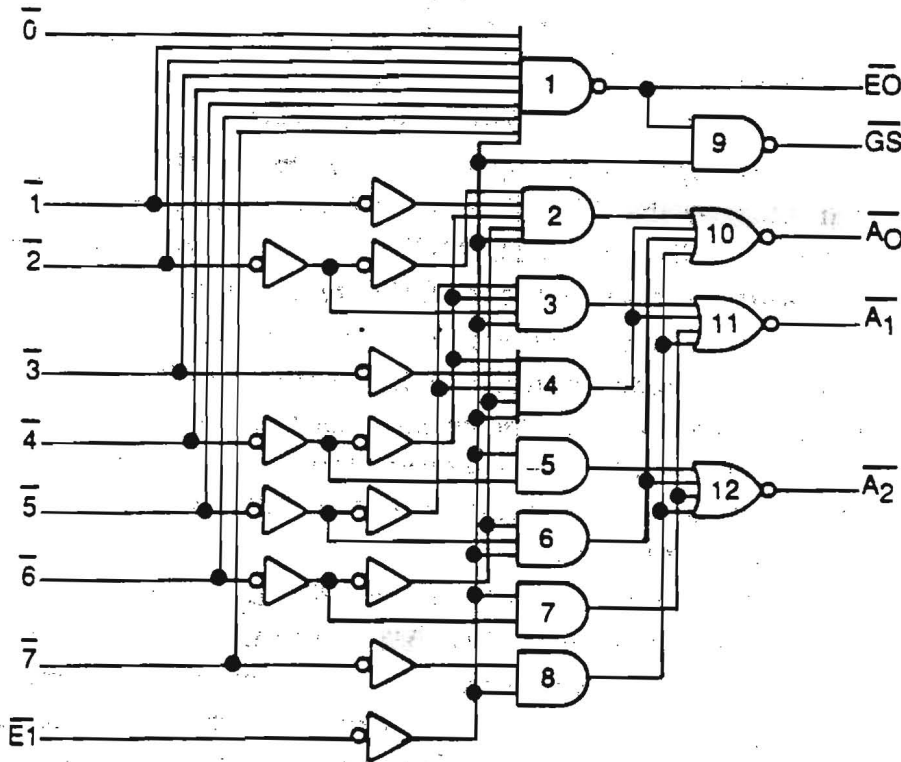


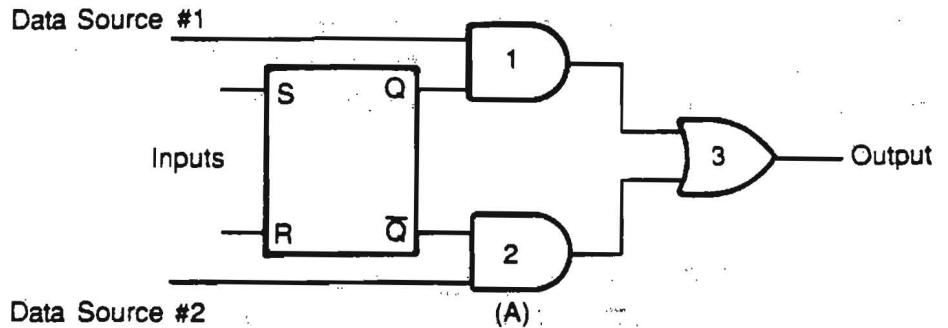
FIGURE 9-5. 8-Line to 3-Line Priority Encoder.

This device is available as an MSI IC as the 74148. The device will perform as a simple encoder as long as only one input is active (LO). When two inputs are active the device will encode the highest order input hence the name priority encoder. This type of circuit is frequently used in binary code generation since these devices are easily cascaded.

A multiplexer is an electronic device that allows the routing of any of several inputs to a single output. Both analog and digital multiplexers are available. The emphasis in this chapter is the digital multiplexer or binary data selector. The logic diagram for a simple digital multiplexer is shown in Figure 9-6.

9.2.5 Multiplexers

FIGURE 9-6. Simple Multiplexer Logic Diagram.



The AND gates are used to select which data source is routed through the OR gate to the output. The AND gates are controlled by the S/C flip-flop. When the Q output of the flip-flop is HI (flip-flop SET), AND gate 1 is enabled and the data from source 1 is applied to the OR gate for output.

The multiplexer will have several data inputs, one data output and some control inputs. Simple multiplexers, like the one described above, can be paralleled to form multiplexers for any size of word. For example, four 2-bit multiplexers can be paralleled to form a multiplexer that will select and output one of two four-bit words.

More complex multiplexers with up to 16 inputs and one output are available as TTL MSI circuits. Multiplexers can be used for applications other than data selectors. A multiplexer can be used for parallel to serial conversion by sequencing through the data inputs. This type of application is shown in Figure 9-7.

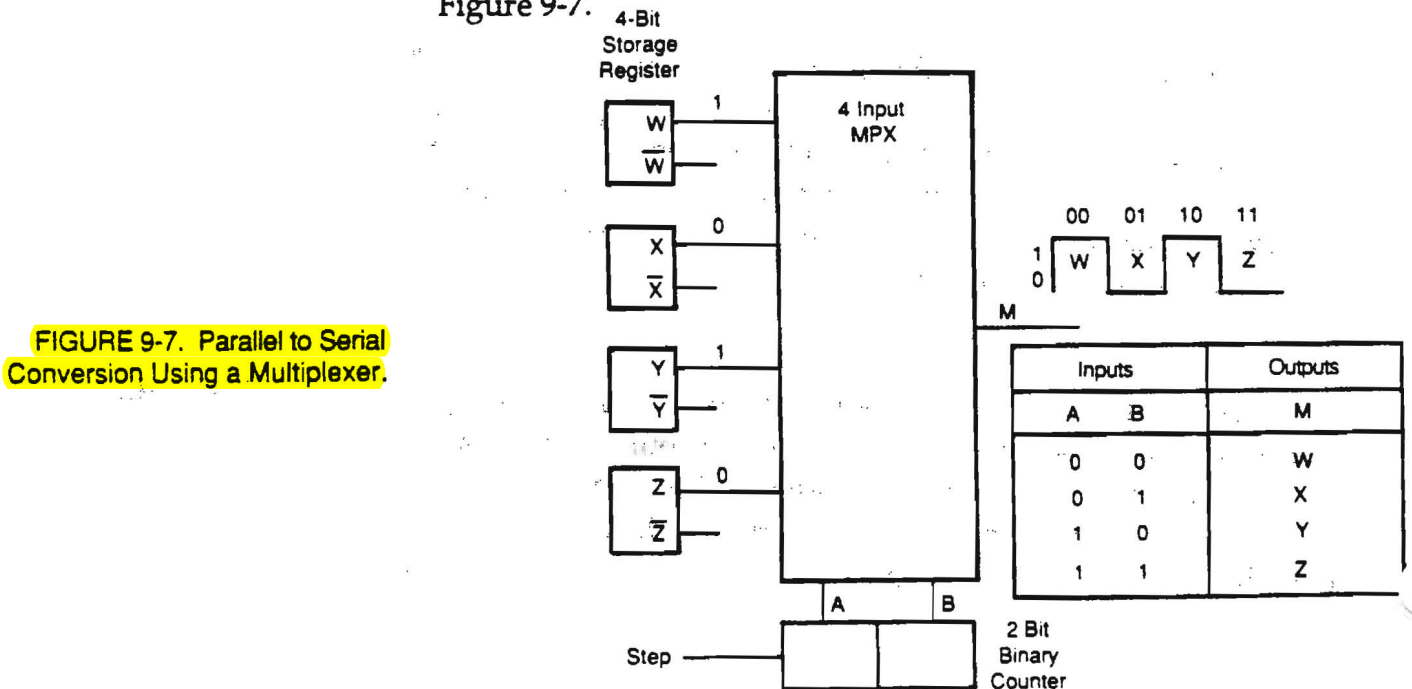
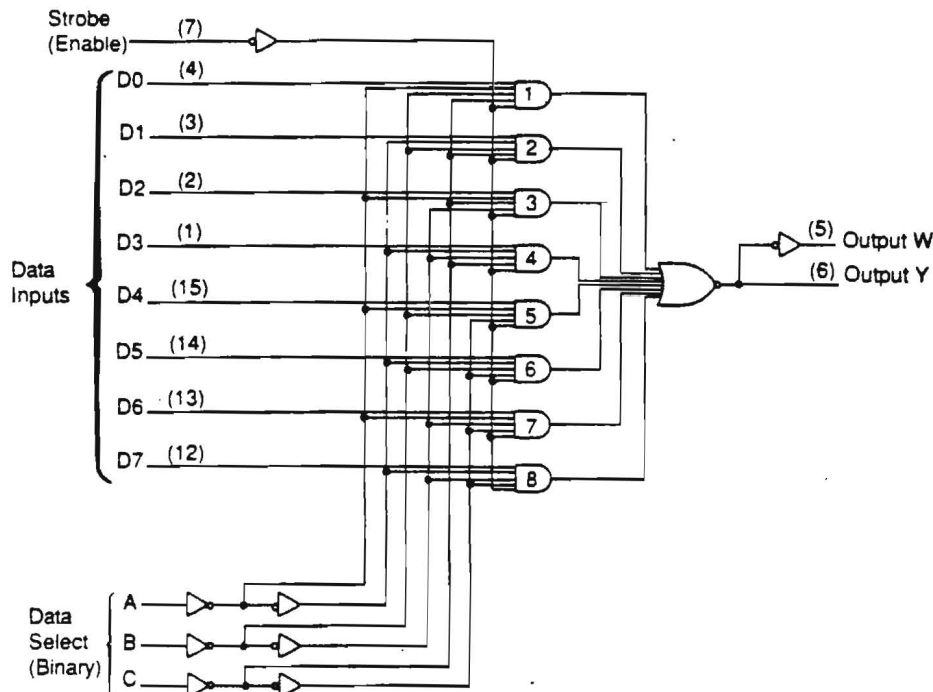


FIGURE 9-7. Parallel to Serial Conversion Using a Multiplexer.

The sequencing needed to drive the multiplexer control lines can be provided by the parallel output of a 2-bit counter. The data to be converted are stored in a 4-bit register.

An application related to parallel to serial conversion is the generation of a binary word. This application works the same as parallel to serial conversion except that the input data are fixed. A third application of digital multiplexers is the generation of boolean logic functions. The logic diagram for an eight input multiplexer is shown in Figure 9-8.

FIGURE 9-8. 8-Input MUX Logic Diagram.



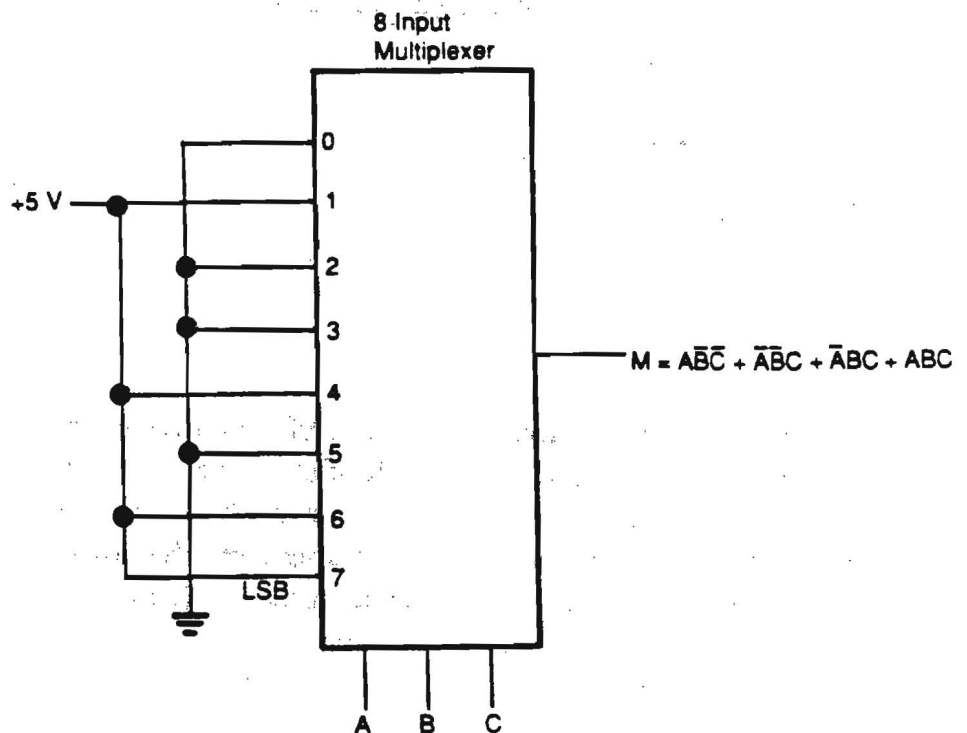
Notice that the output is the sum of products form of the data select inputs. This allows the implementation of any 3-bit logic function by inspection without reduction of the logic equations. Suppose that we wish to implement the equation $f = ABC + \bar{A}BC + A\bar{B}C + \bar{A}\bar{B}C$. In order to accomplish this you need to know which inputs correspond to each boolean function. The functions generated for each corresponding data bit are summarized in TABLE 9-1.

TABLE 9-1. Boolean Products of an 8-Bit Multiplexer.

DATA INPUT	LOGIC OUTPUT
D0	$\overline{A}\overline{B}\overline{C}$
D1	$\overline{A}B\overline{C}$
D2	$\overline{A}B\overline{C}$
D3	$\overline{A}B\overline{C}$
D4	$\overline{A}B\overline{C}$
D5	$\overline{A}B\overline{C}$
D6	$\overline{A}B\overline{C}$
D7	ABC

Since all of the logic terms are implemented by the multiplexer, you only need to choose the desired functions. In order to select the function ABC , the D7 input must be HI. If any input is LO that logic combination is put into the summation gate as a zero or LO and is not represented on the output. To implement the logic function mentioned above, a logic one or HI would be applied to the D1, D4, D6, and D7 inputs while the other data inputs are held LO. This circuit is demonstrated in Figure 9-9.

FIGURE 9-9. Multiplexer Used as a Boolean Function Generator.



Other more complex boolean functions can be generated by using input variables instead of fixed input levels. For example, a four-bit truth table can be implemented utilizing an eight input multiplexer by using the D (fourth) data bit and its complement as data inputs.

This type of circuit is known as a folded data selector circuit. This type of circuit is desirable since implementing a four-bit truth table with fixed logic would require a sixteen input multiplexer which is a much more complex and costly circuit.

A demultiplexer performs the opposite function of a multiplexer. The demultiplexer has one input and several outputs. A demultiplexer is also known as a data router or data distributor. A simple two output demultiplexer is shown in Figure 9-10.

9.2.6 Demultiplexers

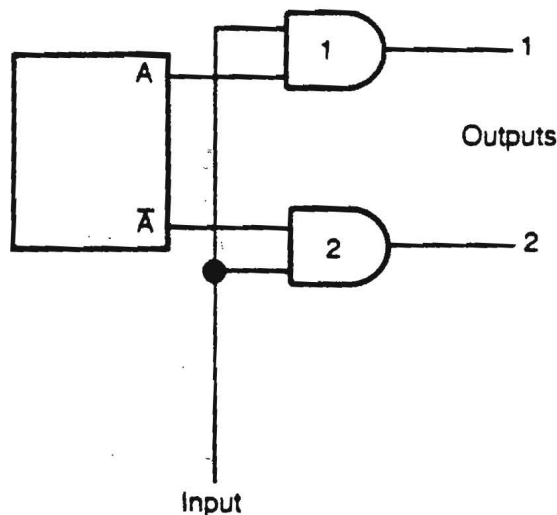
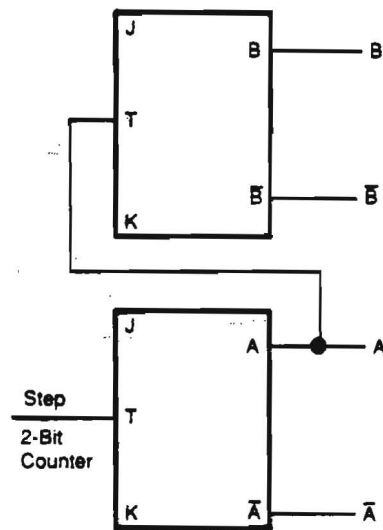
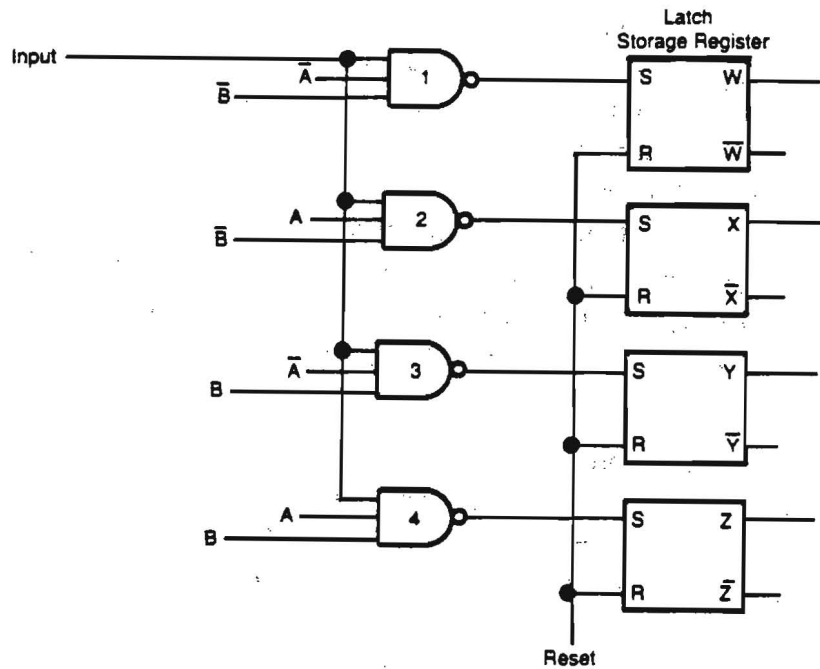


FIGURE 9-10. Simple Demultiplexer.

The AND gates route the data under control of the latch. The input data is connected to all of the and gates. The number of control inputs is determined by the number of possible outputs so that a demultiplexer with four outputs will have a 2-bit control input. The demultiplexer can be used for serial to parallel conversion. This type of circuit is demonstrated in Figure 9-11.

FIGURE 9-11. Serial to Parallel Conversion Using a Demultiplexer.



The parallel output of a 2-bit is used to drive the control circuits for this application. The counter is incremented at the same rate as the input serial data stream changes data values. Each gate in succession is enabled and stores one bit from the serial input data.

A close examination of the data distributor circuit will reveal that the circuit is basically a decoder circuit where the decode gates have a common input. For this reason, a standard MSI decoder is often used as a data distributor.

9.2.7 3-State Registers

The multiplexer and demultiplexers just discussed often receive their inputs from or deliver their outputs to some sort of register. In bussed systems where the potential for input from multiple data sources exists, 3-state registers provide a convenient method of providing these inputs and outputs.

3-state registers are available in any and all of the common register formats (PIPO, SISO, PISO, SIPO) and are available in packages with four, eight and sixteen bit widths. These registers operate in the same manner as the registers that you are familiar with except that they also have the capability of providing a high impedance output state.

In this chapter you have studied MSI circuits. You learned what MSI circuits are and some of their applications. You have studied decoders and their applications of detecting specific data and driving displays. You also learned about encoders and some of their applications.

9.3 SUMMARY

Multiplexers and their applications as data selectors, parallel to serial converters, and boolean equation generators were discussed. You studied demultiplexers and some typical applications of these devices and how some demultiplexers can be implemented with decoders. The final topic covered was the use of 3-state logic in conjunction with the circuits studied in this chapter.

-
1. Define a MSI circuit.

9.4 REVIEW QUESTIONS

2. What alternatives are available for the implementation of logic circuits?

3. What is the basic decoder circuit?

4. Name the basic encoder circuit.

5. Define a digital multiplexer.

6. Describe a digital demultiplexer.

7. Give two applications of decoders.

8. What are three applications of multiplexers?

9. Name an application of demultiplexers.

10. How many control lines are needed for a 16 input multiplexer?

In this lab exercise you will begin your study of MSI circuits. In this lab exercise you will also study the 4-line to 10-line decoder. This circuit is often used as a BCD to decimal decoder. It can also be used as a decoder for any number of binary inputs less than four such as 3-line to 8-line decoders and 2-line to 4-line decoders.

LAB EXERCISE 9.1
Decoders
Objectives

LD-2 Logic Designer

Materials

74LS42 4-Line to 10-Line Decoder

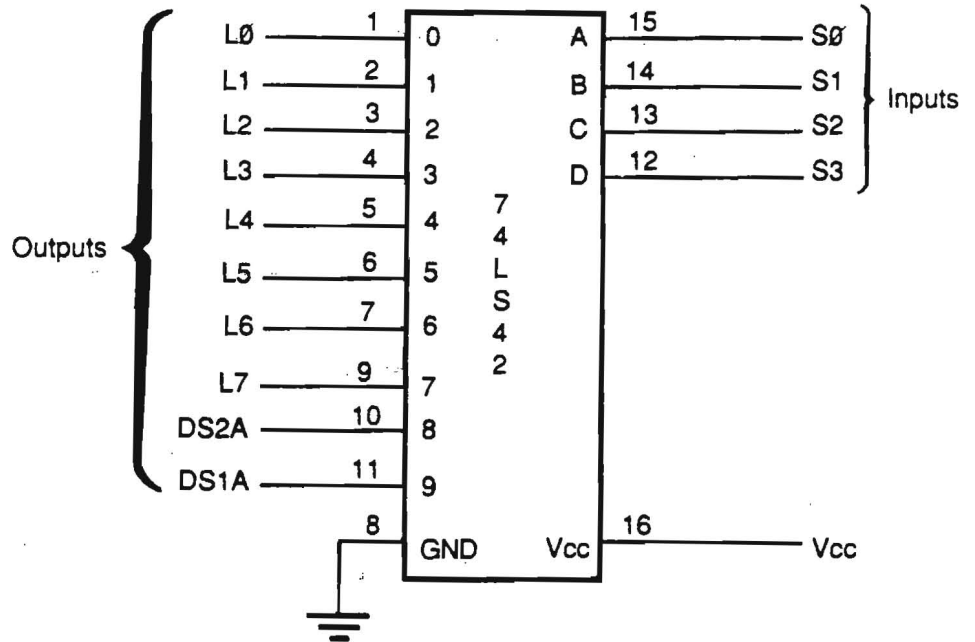
Jumper Wires

TTL Data Book

1. Wire DS1 DE and DS2 DE to 5 VDC. Now wire the circuit shown in Figure 9-12. This is the circuit for a BCD/DEC decoder.

Procedure

FIGURE 9-12. BCD/DEC Decoder.



2. Place S0-S4 to LO. Turn on power. D1 and L1-L7 should light. DS1 and DS2 should show a count of 1.
3. Use S0-S3 as binary inputs. The LEDs L0-L7 are the decoded decimal outputs 0-7. DS2 is the 8 output and DS1 is the 9 output. Observe the circuit's operation and record your observations.
4. Now use S0-S2 as inputs and L0-L7 as the decoded outputs. This circuit is a 3-line to 8-line decoder.
5. Leave this circuit connected while you answer the following questions.

Questions

1. What happens to the outputs when a count greater than nine is entered onto the binary input lines?
-
-

2. What type of circuit could you use to tell that an invalid input had been entered on the S0-S3 inputs?

3. What type of circuit would you need to use to detect invalid inputs for the 3-line to 8-line decoder?

4. Are the outputs of the 74LS42 LO true or HI true?

5. The LD-2 has an internal display driver so that the outputs of the 74LS42 are driving a TTL load and not a directly connected LED. Can the 74LS42 be used to drive an LED indicator directly? Explain your answer.

In this lab exercise you will investigate the use of the 7447 BCD-TO-SEVEN-SEGMENT DECODERS/DRIVERS. You will also learn about the use of seven-segment displays. You will study implementing simple displays.

LAB EXERCISE 9.2
Decoder/Drivers
Objectives

LD-2 Logic Designer

7447 BCD-TO-7-Segment Decoders/Drivers

TTL 312 Common-Anode 7-Segment Display

Jumper Wires

Materials