13.35 Design the logic circuits, to provide a read and a write cycle for a 7489.

13.36 Refer to the 74S201 information in Fig. 13.26 and determine the following:

a. Minimum write-enable pulse width

b. Setup time, address to write-enable

c. Hold time, data from write-enable

13.37 Draw the logic diagram for a 256-word 8-bit memory using '201s.

Answers to Self-tests

1. A DRAM must be refreshed periodically.
2. EPROM stands for erasable-programmable read-only memory.
3. Cache memory is a small high-speed SRAM used inside a computer to speed up operation.
4. Even
5. Tape access time is too long!
6. Binary information is recorded on as magnetic film by magnetizing spots with two different orientations.
7. 780 nm.
8. CD-ROM 25% and more than 70%. CD-RW 15% and 25%.
9. 8.5 GB.
10. 145 (decimal) = 1001 0001 = $A_7A_6A_5A_4A_3A_2A_1A_0$
11. The cell at address 22—row 2 and column 2.
12. It refers to a ROM whose contents are established during the manufacturing process.
13. The triangle is the symbol for a three-state output.
14. It can be corrected by simply programming (adding) a 1 at word position Q_3. Note that you can add a 1 by programming (this is destroying a fuse link), but you cannot remove a programmed 1, since this would require replacing a fuse link.
15. Sixteen 4-bit words.
16. 1024, 4-bit words.
17. DRAM stands for dynamic random-access memory.
18. 16,384 × 1 bits
19. Through multiplexer-based look up table.

Digital Integrated Circuits

14

OBJECTIVES

- **Explain how diodes and transistors can be used as electronic switches**
- **Demonstrate an understanding of TTL devices, their parameters, how to drive them, and how to use them to drive external loads**
- **Be familiar with CMOS-devices and characteristics**
- **Understand TTL-to-CMOS and CMOS-to-TTL interfacing**

In 1964 Texas Instruments introduced transistor-transistor logic (TTL), a widely used family of digital devices. TTL is fast, inexpensive, and easy to use. In this chapter we discuss several types of TTL: standard, high-speed, low-power, Schottky, and low-power Schottky. You will learn about open-collector and tri-state devices because these are used to build buses, the backbone of modern computers and digital systems. Since TTL uses active-low as well as active-high signals, negative logic may be used as well as positive logic. Complementary metal-oxide semiconductor (CMOS) devices are chips that combine *p*-channel and *n*-channel MOSFETs in a push-pull arrangement. Because the input current of a MOSFET is much smaller than that of a bipolar transistor, cascaded CMOS devices have very low power dissipation compared with TTL devices. This low dissipation explains why CMOS circuits are used in battery-powered equipment such as pocket calculators, digital wristwatches, and portable computers.

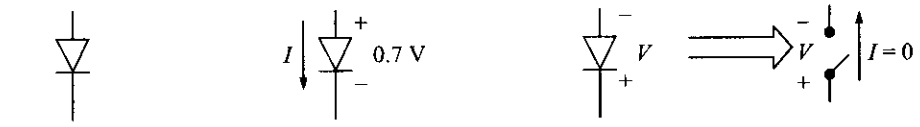
Since a knowledge of the subjects covered here is not prerequisite to any other chapter in this text, the material can be studied in part or in whole, at any time. An understanding of Ohm's law and familiarity with basic dc circuits are the only background needed.

14.1 SWITCHING CIRCUITS

The semiconductor devices used in digital integrated circuits (ICs) include diodes, *bipolar junction transistors (BJTs)* and *metal-oxide-semiconductor field-effect transistors (MOSFETs)*. The most popular transistor-transistor logic (TTL) in use includes the 7400 and the 74LS00 families; resistors, diodes, and BJTs are the elements used to construct these circuits. The 74C00 and the 74HC00 are the most widely used families constructed using MOSFETs. These two families of circuits are referred to as CMOS, since they use two different types of MOSFETs. In Chapter 1, we used the term *electronic switch* (see Fig. 1.7). Virtually all digital ICs in use today are silicon, so let's see how a silicon diode or transistor is used as an electronic switch.

The Semiconductor Diode

The symbol for a semiconductor diode (sometimes called a *pn* junction) is shown in Fig. 14.1a. The diode behaves like a *one-way* switch. That is, it will allow an electric current in one direction but not the other. We will use *conventional current flow* rather than *electron flow*. Figure 14.1b shows the direction of current through a diode—this is the *forward* direction. When conducting current, a silicon (Si) diode will have a nominal voltage of 0.7 V across its terminals as shown in Fig. 14.1b. In this condition, the diode is said to be *forward-biased*. Notice that the triangle in the diode symbol points in the direction of forward current—an easy memory crutch! It is not possible to pass current through the diode in the other direction—the *reverse* direction. When *reverse-biased*, the diode will act as an open switch as illustrated in Fig. 14.1c. To summarize:

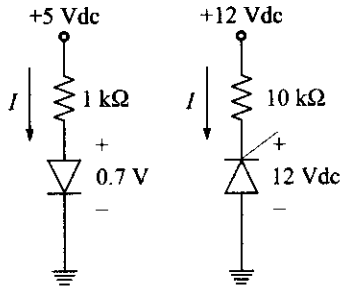


(a) Symbol (b) Forward bias (c) Reverse bias

Fig. 14.1 Semiconductor diode

1. When forward-biased, the diode conducts current, and the voltage across the diode terminals is about 0.7 Vdc.
2. When reverse-biased, the diode will not conduct current. The voltage across the diode terminals depends on the external circuit.

Example 14.1 For each diode in Fig. 14.2, determine whether the diode is forward- or reverse-biased. Determine the diode current I in each case.



(a) (b)

Fig. 14.2

Solution

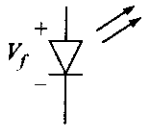
(a) The current direction is from +5 Vdc to ground, and thus the diode is forward-biased. The voltage across the diode terminals is 0.7 Vdc, and the diode current is found as

$$I = (5 - 0.7)/1\ \text{k}\Omega = 4.3/1\ \text{k}\Omega = 4.3\ \text{mA}$$

(b) The current direction is from +12 Vdc to ground, thus the diode is reverse-biased. The diode current is then $I = 0.0$ mA. There is no voltage across the 10-kΩ resistor, and thus the voltage across the diode terminals is 12 Vdc.

LEDs

The symbol for a light-emitting diode (LED) is shown in Fig. 14.3a. The arrows indicate light emission capability. The operation of an LED is similar to that of an ordinary diode. When forward-biased, it emits light in the visible spectrum and is thus used as an indicator. However, the voltage across the diode terminals when forward-biased (V_f) is somewhat greater than 0.7 Vdc. Typical LED forward voltages given in Fig. 14.3b show that V_f varies with the color of the emitted light. The color of the emitted light depends on the elements added to the semiconductor material during manufacturing.



(a) Symbol

	Red	Yellow	Green
V_f(V)	1.6	2.2	2.4

(b) Typical forward voltages

Fig. 14.3

Example 14.2 The diode in Fig. 14.2a is replaced with a *red* LED. What is the diode current?

Solution The diode is forward-biased, and the voltage across its terminals is about 1.6 Vdc (Fig. 14.3b). The diode current is then

$$I = (5 - 1.6)/1\ \text{k}\Omega = 3.4/1\ \text{k}\Omega = 3.4\ \text{mA}$$

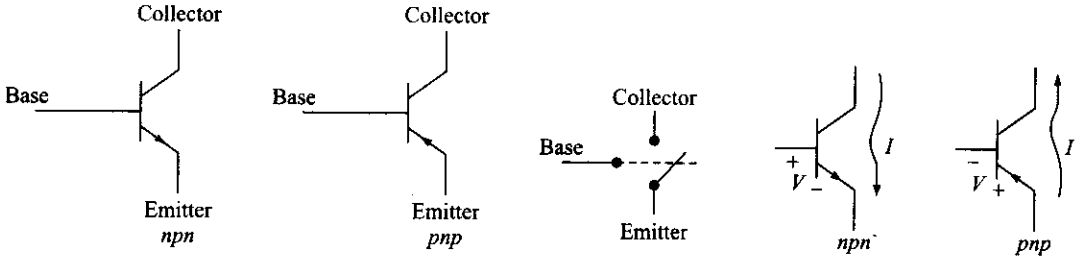
BJTs

The bipolar junction transistor (BJT) is available in two polarities (*npn* and *pnp*), as shown by the symbols in Fig. 14.4a. The BJT terminals are named *collector*, *emitter*, and *base*, as indicated. In Fig. 14.4b the BJT behaves as an electronic switch. The switch is activated by applying a voltage between base and emitter. Here's how it works:

1. The voltage between base and emitter is zero. The switch is open, and no current is allowed between collector and emitter. The transistor is off.
2. A voltage is applied between base and emitter. The switch is closed and a current is allowed between collector and emitter. The transistor is on. The voltage between emitter and collector (across a closed switch) is zero!

Since the BJT is available in two polarities—*npn* and *pnp*—the polarity of the applied base-emitter voltage must be as shown in Fig. 14.4c. For the *npn*, the base must be more positive than the emitter. The opposite is true for the *pnp*. This base-emitter voltage is applied across a forward-biased *pn* junction (a diode) and is thus limited to about 0.7 Vdc. Care must be taken not to exceed 0.7 Vdc, or the BJT may be destroyed.

The current through the *npn* transistor must be from collector to emitter as shown in Fig. 14.4c. For the *pnp*, current must be from emitter to collector. Notice that the current is in the direction of the arrow on the



(a) BJT symbols (b) Electronic switch (c)

Fig. 14.4

emitter—a good memory crutch! These polarities and current directions are important—*you should make every effort to commit them to memory!*

Example 14.3

a. Determine the current I and the voltage V_2 for the circuit in Fig. 14.5a if (i) $V_1 = 0$ Vdc, and (ii) $V_1 = +5$ Vdc.
b. Repeat part (a) for the circuit in Fig. 14.5b.

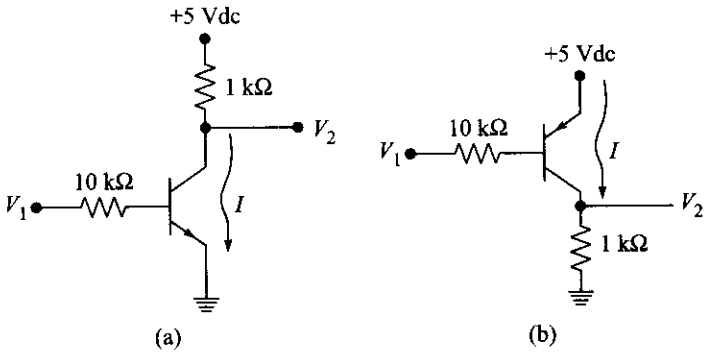


Fig. 14.5

Solution

a. $V_1 = 0$ Vdc. There is no current in the 10-kΩ resistor. Thus the voltage base-emitter is zero. The BJT is off (switch is open). The BJT current and the current in the 1-kΩ resistor is zero. The voltage V_2 is +5 Vdc.
$V_1 = +5$ Vdc. The base is more positive than the emitter—the BJT is on (switch is closed). V_2 is zero. The BJT current is $I = 5$ mA.
b. V_1 is 0 Vdc. The base is more negative than the emitter—the BJT is on (switch closed). V_2 is +5 Vdc. The BJT current is $I = 5$ mA.
V_1 is +5 Vdc. There is no current in the 10-kΩ resistor. The base is at +5 Vdc, and so is the emitter. Thus the voltage base-emitter is zero, and the BJT is off (switch open). The current $I = 0$ mA, and $V_2 = 0$ Vdc.

Let's look carefully at the results from Example 14.3. For both circuits, when $V_1 = 0$ Vdc, $V_2 = +5$ Vdc. Also, when $V_1 = +5$ Vdc, $V_2 = 0$ Vdc. Clearly V_2 is always the *inverse* of V_1—in other words, each circuit in Fig. 14.5 is an inverter! Either of these circuits can be used to implement the basic inverter introduced in Chapter 1 (Fig. 1.10).

MOSFETs

MOSFETs are available in two polarities (n-channel and p-channel) as shown by the symbols in Fig. 14.6a. MOSFETs operate as "depletion" or "enhancement" mode devices; the transistors in Fig. 14.6 are enhancement types. The MOSFET terminals are named *gate, source, drain, and body* as indicated. When the body is connected to the source, as is often the case with ICs, the simplified symbols are used. The MOSFET also behaves as the electronic switch in Fig. 14.6b. The switch is activated by applying a voltage between gate and source. Here's how it works:

1. The voltage between gate and source is zero. The switch is open, and no current is allowed between source and drain. The transistor is off.

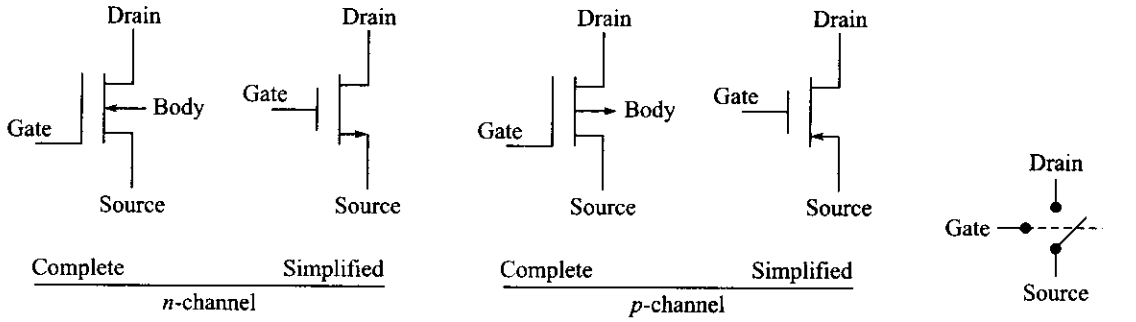


Fig. 14.6

2. A voltage is applied between gate and source. The switch is closed, and a current is allowed between source and drain. The transistor is on. The voltage between source and drain (across a closed switch) is zero!

Since the MOSFET is available in two polarities—*n*-channel and *p*-channel—the polarity of the applied gate-source voltage must be as shown in Fig. 14.6c. For the *n*-channel transistor, the gate must be more positive than the source. The opposite is true for the *p*-channel transistor. The current through the *n*-channel transistor must be from drain to source as shown in Fig. 14.6c. For the *p*-channel transistor, current must be from source to drain. Notice that the current is in the direction of the small arrow on the drain—a good memory crutch! These polarities and current directions are important—*you should make every effort to commit them to memory!*

Example 14.4 An *n*-channel MOSFET can be used to construct a simple inverter as shown in Fig. 14.7. Determine the current I and the output voltage V_2 if (a) $V_1 = 0$ Vdc, and (b) $V_1 = +5$ Vdc.

Solution

(a) $V_1 = 0$ Vdc. There is no current in the 100-kΩ resistor. Thus the gate-source voltage is zero. The MOSFET is off (the switch is open). The MOSFET current and the current in the 10-kΩ resistor are zero. The voltage V_2 is +5 Vdc.

(b) $V_1 = +5$ Vdc. The gate is more positive than the source—the MOSFET is on (the switch is closed). V_2 is zero. The MOSFET current is $I = 0.5$ mA. This circuit could also be used to implement the basic inverter introduced in Chapter 1 (Fig. 1.10).

The small size of a MOSFET on an IC is one of the great advantages of ICs constructed using MOSFETs. The 10-kΩ resistor in Fig. 14.7 requires a large area on an IC compared to a MOSFET. A second MOSFET

can be used in place of this resistor, as shown in Fig. 14.8. In this case, transistor Q_1 has its gate connected directly to its drain. When it is connected in this fashion, its behavior is similar to a resistor but shows little non-linearity compared to passive load, Q_1 is called an *active load*, and this circuit is a simple inverter.

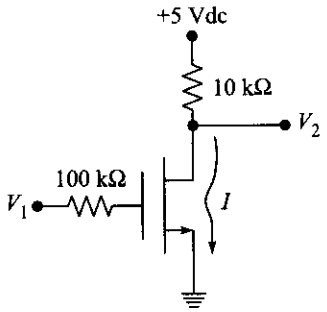


Fig. 14.7

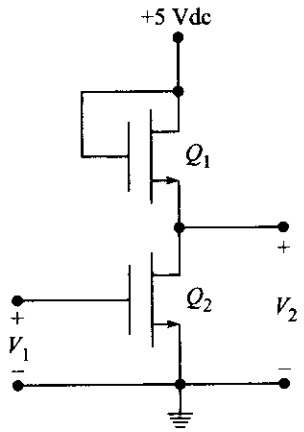


Fig. 14.8 An inverter with active load

Complementary Metal-Oxide-Semiconductor (CMOS) FETs

ICs constructed entirely with *n*-channel MOSFETs are called *NMOS* ICs. ICs constructed entirely with *p*-channel MOSFETs are called *PMOS* ICs. The 74C00 and 74HC00 families are constructed using *both* *n*-channel and *p*-channel MOSFETs. Since *n*-channel and *p*-channel MOSFETs are considered *complementary* devices, these ICs are referred to as *CMOS* ICs.

A CMOS inverter is shown in Fig. 14.9. Ideally, the characteristics of the *n* channel are closely matched with the *p* channel. This circuit is the basis for the 74C00 and 74HC00 families. Here's how it works:

1. $V_1 = 0$ Vdc. Q_n is off and Q_p is on. $V_2 = +5$ Vdc.
2. $V_1 = +5$ Vdc. Q_n is on and Q_p is off. $V_2 = 0$ Vdc.

Note that in the steady state (while not switching), one of the transistors is *always off*. As a result, the current I = 0 mA. When switching between states, both transistors are on for a very short time because of the rise or fall time of V_1. This is the only time the current I is nonzero. This is the reason CMOS is used in applications where dc power supply current must be held to a minimum—watches, pocket calculators, etc. *A word of warning:* If the input V_1 is held at +5 Vdc/2 = 2.5 Vdc, *both* transistors will be on. This is an almost direct short between +5 Vdc and ground, and it won't be long before both transistors expire! So don't impose this condition on a CMOS IC.

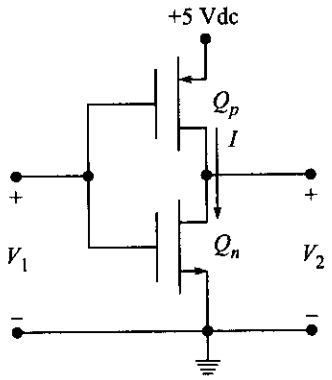


Fig. 14.9 A CMOS inverter

SELF-TEST

1. How does an LED differ from an ordinary silicon diode?
2. What are the two types of BJTs? What is the complement of an *n*-channel MOSFET?
3. An *npn* BJT is on when its base is more (positive, negative) than its emitter.
4. What is an active load in an NMOS IC?

14.2 7400 TTL

Standard TTL

Figure 14.10 shows a TTL NAND gate. The *multiple-emitter* input transistor is typical of the gates and other devices in the 7400 series. Each emitter acts like a diode; therefore, Q_1 and the 4-kΩ resistor act like a 2-input AND gate. The rest of the circuit inverts the signal so that the overall circuit acts like a 2-input NAND gate. The output transistors (Q_3 and Q_4) form a *totem-pole connection* (one *npn* in series with another); this kind of output stage is typical of most TTL devices. With a totem-pole output stage, either the upper or lower transistor is on. When Q_3 is on, the output is high; when Q_4 is on, the output is low.

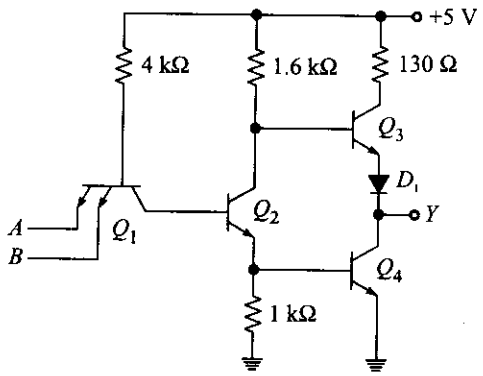


Fig. 14.10 Two-input TTL NAND gate

The input voltages A and B are either low (ideally grounded) or high (ideally +5 V). If A or B is low, the base of Q_1 is pulled down to approximately 0.7 V. This reduces the base voltage of Q_2 to almost zero. Therefore, Q_2 cuts off. With Q_2 open, Q_4 is off, and the Q_3 base is pulled high. The emitter of Q_3 is only 0.7 V below the base, and thus the Y output is pulled up to a high voltage.

On the other hand, when A and B are both high voltages, the emitter diodes of Q_1 stop conducting, and the collector diode goes into forward conduction. This forces Q_2 to turn on. In turn, Q_4 goes on and Q_3 turns off, producing a low output. Table 14.1 summarizes all input and output conditions.

Table 14.1 Two-Input NAND Gate

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

Without diode D_1 in the circuit, Q_3 will conduct slightly when the output is low. To prevent this, the diode is inserted; its voltage drop keeps the base-emitter diode of Q_3 reverse-biased. In this way, only Q_4 conducts when the output is low.

Totem-Pole Output

Totem-pole transistors are used because they produce a *low output impedance*. Either Q_3 acts as an emitter follower (high output), or Q_4 is on (low output). When Q_3 is conducting, the output impedance is approximately 70 ohms (Ω); when Q_4 is on, the output impedance is only 12 Ω (this can be calculated from information on

the data sheet). Either way, the output impedance is low. This means the output voltage can change quickly from one state to the other because any stray output capacitance is rapidly charged or discharged through the low output impedance.

Propagation Delay Time and Power Dissipation

Two quantities needed for later discussion are power dissipation and propagation delay time. A standard TTL gate has a power dissipation of about 10 milliwatts (mW). It may vary from this value because of signal levels, tolerances, etc. but on the average it is 10 mW per gate. The *propagation delay time* is the time it takes for the output of a gate to change after the inputs have changed. The propagation delay time of a TTL gate is approximately 10 nanoseconds (ns).

Device Numbers

By varying the design of Fig. 14.10 manufacturers can alter the number of inputs and the logic function. With only few exceptions, the multiple-emitter inputs and the totem-pole outputs are used for different TTL devices. Table 14.2 lists some of the 7400 series TTL gates. For instance, the 7400 is a chip with four 2-input NAND gates in one package. Similarly, the 7402 has four 2-input NOR gates, the 7404 has six inverters, and so on.

Table 14.2 Standard TTL

Device Number	*Description*
7400	Quad 2-input NAND gates
7402	Quad 2-input NOR gates
7404	Hex inverter
7408	Quad 2-input AND gates
7410	Triple 3-input NAND gates
7411	Triple 3-input AND gates
7420	Dual 4-input NAND gates
7421	Dual 4-input AND gates
7425	Dual 4-input NOR gates
7427	Triple 3-input NOR gates
7430	8-input NAND gate
7486	Quad 2-input XOR gates

5400 Series

Any device in the 7400 series works over a temperature range of 0 to 70°C and over a supply range of 4.75 to 5.25 V. This is adequate for commercial applications. The 5400 series, developed for the military applications, has the same logic functions as the 7400 series, except that it works over a temperature range of −55 to 125°C and over a supply range of 4.5 to 5.5 V. Although 5400 series devices can replace 7400 series devices, they are rarely used commercially because of their much higher cost.

High-Speed TTL

The circuit of Fig. 14.10 is called *standard TTL*. By decreasing the resistances a manufacturer can lower the internal time constants; this decreases the propagation delay time. The smaller resistances, however, increase the power dissipation. This design variation is known as *high-speed TTL*. Devices of this type are numbered 74H00, 74H01, 74H02, and so on. A high-speed TTL gate has a power dissipating around 22 mW and a propagation delay time of approximately 6 ns.

Low-Power TTL

By increasing the internal resistances a manufacturer can reduce the power dissipation of TTL gates. Devices of this type are called *low-power TTL* and are numbered 74L00, 74L01, 74L02, etc. These devices are

slower than standard TTL because of the larger internal time constants. A low-power TTL gate has a power dissipation of 1 mW and a propagation delay time of about 35 ns.

Schottky TTL

With standard TTL, high-speed TTL, and low-power TTL, the transistors are switched on with excessive current, causing a surplus of carriers to be stored in the base. When you switch a transistor from on to off, you have to wait for the extra carriers to flow out of the base. The delay is known as *saturation delay time.*

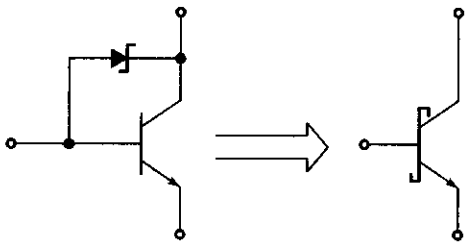


Fig. 14.11 Schottky diode prevents transistor saturation

One way to reduce saturation delay time is with *Schottky TTL.* The idea is to fabricate a Schottky diode along with each bipolar transistor of a TTL circuit, as shown in Fig. 14.11. Because the Schottky diode has a forward voltage of only 0.25 to 0.4 V, it prevents the transistor from saturating fully. This virtually eliminates saturation delay time, which means better switching speed. These devices are numbered 74S00, 74S01, 74S02, and so forth.

Schottky TTL devices are very fast, capable of operating reliably at 100 megahertz (MHz). The 74S00 has a power dissipation around 20 mW per gate and a propagation delay time of approximately 3 ns.

Low-Power Schottky TTL

By increasing internal resistances as well as using Schottky diodes, manufacturers have come up with a compromise between low power and high speed: *low-power Schottky TTL.* Devices of this type are numbered 74LS00, 74LS01, 74LS02, etc. A low-power Schottky gate has a power dissipation of around 2 mW and a propagation delay time of approximately 10 ns, as shown in Table 14.3.

The Winner

Low-power Schottky TTL is the best compromise between power dissipation and saturation delay time. In other words, of the five TTL types listed in Table 14.3, low-power Schottky TTL has emerged as the favorite of digital designers. It is used for almost everything. When you must have more output current, you can fall back on standard TTL. Or, if your application requires faster switching speed, then Schottky TTL is useful. Low-power and high-speed TTL are rarely used. if at all.

SELF-TEST

5. Draw the symbol for a Schottky transistor.
6. Which TTL family offers the lowest power and the fastest operation?

14.3 TTL PARAMETERS

7400 series devices are guaranteed to work reliably over a temperature range of 0 to 70°C and over a supply range of 4.75 to 5.25 V. In the discussion that follows, *worst case* means that the *parameters* (input current,

output voltage, and so on) are measured under the worst conditions of temperature and voltage. This means maximum temperature and minimum voltage for some parameters, minimum temperature and maximum voltage for others, or whatever combination produces the worst values.

Floating Inputs

When a TTL input is high (ideally +5 V), the emitter current is approximately zero (Fig. 14.12a). When a TTL input is *floating* (unconnected, as shown in Fig. 14.12b), no emitter current is possible because of the open circuit. Therefore, a floating TTL input is equivalent to a high output. Because of this, you sometimes see unused TTL inputs left unconnected; an open input allows the rest of the gate to function properly.

There is a disadvantage to floating inputs. When you leave an input open, it acts as a small antenna. Therefore, it will pick up stray electromagnetic noise voltages. In some environments, the noise pickup is large enough to cause erratic operation of logic circuits. For this reason, most designers prefer to connect unused TTL inputs to the supply voltage.

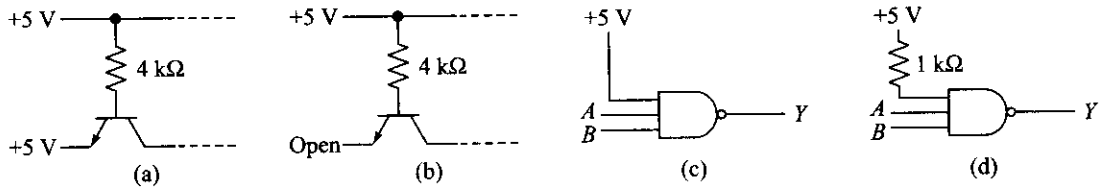


Fig. 14.12 **(a) High input, (b) Open is equivalent to high input, (c) Direct connection to supply voltage, (d) High input through a pull-up resistor**

For instance, Fig. 14.12c shows a 3-input NAND gate. The top input is unused, so it is connected to +5 V. A direct connection like this is all right with most Schottky devices (74S and 74LS) because their inputs can withstand supply overvoltages caused by switching transients. Since the top input is always high, it has no effect on the output. (*Note*: You don't ground the unused TTL input of Fig. 14.12c because then the output would remain stuck high, no matter what the values of A and B.)

Figure 14.12d shows an indirect connection to the supply through a resistor. This type of connection is used with standard, low-power, and high-speed TTL devices (74, 74L, and 74H). These older TTL devices have an absolute maximum input rating of +5.5 V. Beyond this level, the ICs may be damaged. The resistor is called a *pull-up resistor* because it serves to pull the input voltage up to a high. Most transients on the supply voltage are too short to charge the input capacitance through the pull-up resistor. Therefore, the input is protected against temporary overvoltages.

Worst-Case Input Voltages

Figure 14.13a shows a TTL inverter with an input voltage of V_i and an output voltage of V_o. When V_i is 0 V (grounded), it is in the low state and is designated V_{IL}. With TTL devices, we can increase V_{IL} to 0.8 V and still have a low-state input because the output remains in the high state. In other words, the low-state input voltage V_{IL} can have any value from 0 to 0.8 V. TTL data sheets list the worst-case low input as

$$V_{IL,\max} = 0.8 \text{ V}$$

If the input voltage is greater than this, the output state is unpredictable.

However, suppose V_i is 5 V in Fig. 14.13a. This is a high input and can be designated V_{IH}. This voltage can decrease all the way down to 2 V without changing the output state. In other words, the high-stage input V_{IH}

is from 2 to 5 V; any input voltage in this range produces a low output voltage. Data sheets list the worst-case high input as

$$V_{IH,\min} = 2 \text{ V}$$

When the input voltage is less than this, the output state is again unpredictable.

Figure 14.13b summarizes these ideas. As you see, any input voltage less than 0.8 V is a valid low-state input. Any input greater than 2 V is a valid high-state input. Any input between 0.8 and 2 V is indeterminate because there is no guarantee that it will produce the correct output voltage.

Worst-Case Output Voltages

Ideally, the low output state is 0 V, and the high output state is 5 V. We cannot attain these ideal values because of internal voltage drops inside TTL devices. For instance, when the output voltage is low in Fig. 14.13a, Q_4 is saturated and has a small voltage drop across it. With TTL devices, any output voltage from 0 to 0.4 V is considered a low output and is designated V_{OL}. This means the low-state output V_{OL} of a TTL device may have any value between 0 and 0.4 V. Data sheets list the worst-case low output as

$$V_{OL,\max} = 0.4 \text{ V}$$

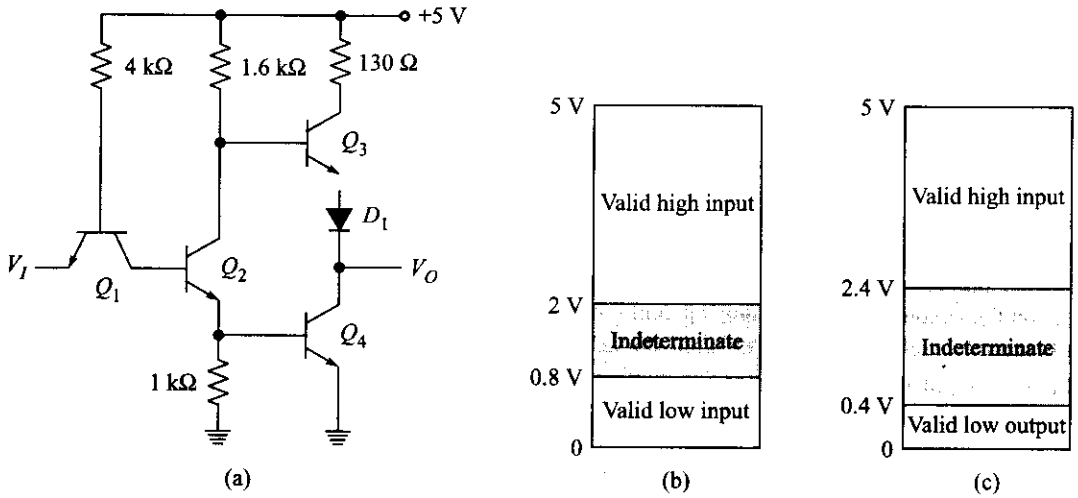


Fig. 14.13 **(a) TTL inverter, (b) TTL input profile, (c) TTL output profile**

When the output is high, Q_3 acts as an emitter follower. Because of the voltage drop across Q_3, D_1, and the 130-Ω resistor, the output voltage will be less than supply voltage. With TTL devices, the high-state output voltage is designated V_{OH}; it has a value between 2.4 and 3.9 V, depending on the supply voltage, temperature, and load. TTL data sheets list the worst-case high output as

$$V_{OH,\min} = 2.4 \text{ V}$$

Figure 14.13c summarizes the output states. As shown, any output voltage less than 0.4 V is a valid low-state output, any output voltage greater than 2.4 V is a valid high-state output, and any output between 0.4 and 2.4 V is indeterminate under worst-case conditions.

Profiles and Windows

The input characteristics of Fig. 14.13b are called the TTL input *profile*. Furthermore, each rectangular area in Fig. 14.13b can be thought of as a *window*. There is a low window (0 to 0.8 V), an indeterminate window (0.8 to 2.0 V), and a high window (2.0 to 5 V).

Similarly, Fig. 14.13c is the TTL output profile. Here you see a low window from 0 to 0.4 V, an indeterminate window from 0.4 to 2.4 V, and a high window from 2.4 to 5 V.

Values to Remember

We have discussed the low and high states for the input and output voltages. Here they are again as a reference for future discussions:

$$V_{IL,\max} = 0.8 \text{ V}$$
$$V_{IH,\min} = 2 \text{ V}$$
$$V_{OL,\max} = 0.4 \text{ V}$$
$$V_{OH,\min} = 2.4 \text{ V}$$

These are the worst-case values shown in Fig. 14.13b and c. On the input side, a voltage has to be less than 0.8 V to qualify as a low-state input, and it must be more than 2 V to be considered a high-state input. On the output side, the voltage has to be less than 0.4 V to be a low-state output and more than 2.4 V to be a high-state output.

Compatibility

TTL devices are *compatible* because the low and high output windows fit inside the low and high input windows. Therefore the output of any TTL device is suitable for driving the input of another TTL device. For instance, Fig. 14.14a shows one TTL device driving another. The first device is called a *driver* and the second a *load*.

Figure 14.14b shows the output stage of the TTL driver connected to the input stage of the TTL load. The driver output is shown in the low state. Since any input less than 0.8 V is a low-state input, the driver output (0 to 0.4 V) is compatible with the load input requirements.

Similarly, Fig. 14.14c shows high TTL output. The driver output (2.4 to 3.9 V) is compatible with the load input requirements (greater than 2 V).

Sourcing and Sinking

When a standard TTL output is low (Fig. 14.14b), an emitter current of approximately 1.6 milliamperes (mA) (worst case) exists in the direction shown. The conventional flow is from the emitter of Q_1 to the collector of Q_4. Because it is saturated, Q_4 acts as a *current sink*; conventional current flows through Q_4 to ground like water flowing down a sink.

However, when the standard TTL output is high (Fig. 14.14c), a reverse emitter current of 40 microamperes (μA) (worst-case) exists in the direction shown. Conventional current flows out of Q_3 to the emitter of Q_1. In this case, Q_3 is acting as a *source*.

Data sheets list the worst-case input currents:

$$I_{IL,\max} = -1.6\text{mA} \quad I_{IH,\max} = 40\ \mu\text{A}$$

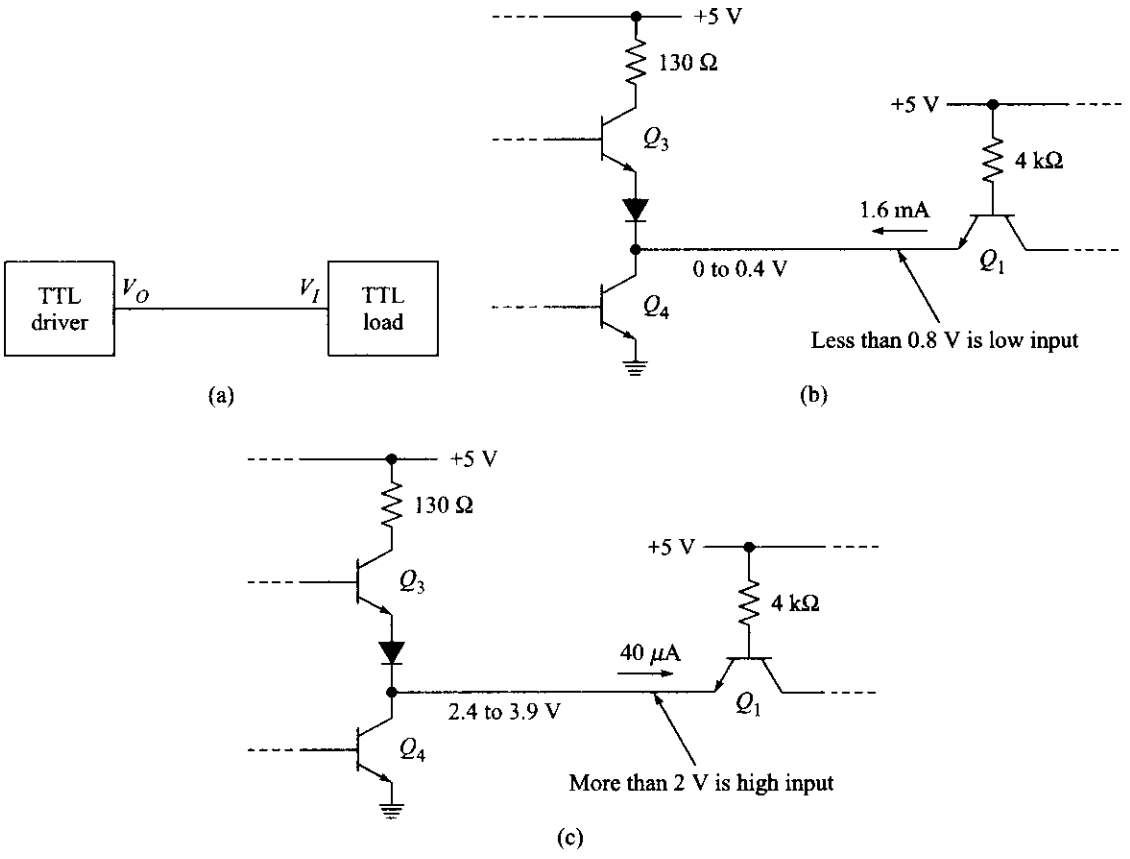


Fig. 14.14 **Sourcing and sinking current**

The minus sign indicates that the conventional current is out of the device; a plus sign means the conventional current is into the device. All data sheets use this notation, so do not be surprised when you see minus currents. The previous data tells us the maximum input current is 1.6 mA (outward) when an input is low and 40 μA (inward) when an input is high.

Noise Immunity

In the worst case, there is a difference of 0.4 V between the driver output voltages and required load input voltages. For instance, the worst-case low values are

$$V_{OL,\max} = 0.4 \text{ V} \quad \text{driver output}$$
$$V_{IL,\max} = 0.8 \text{ V} \quad \text{load input}$$

Similarly, the worst-case high values are

$$V_{OH,\min} = 2.4 \text{ V} \quad \text{driver output}$$
$$V_{IH,\min} = 2 \text{ V} \quad \text{load input}$$

In either case, the difference is 0.4 V. This difference is called *noise immunity*. It represents built-in protection against noise. Note that the concept of noise margin has been introduced in section 1.8 of Chapter 1.

Why do we need protection against noise? The connecting wire between a TTL driver and load is equivalent to a small antenna that picks up stray noise signals. In the worst case, the low input to the TTL load is

$$V_{IL} = V_{OL} + V_{noise} = 0.4\text{ V} + V_{noise}$$

and the high-stage input is

$$V_{IH} = V_{OH} - V_{noise} = 2.4\text{ V} - V_{noise}$$

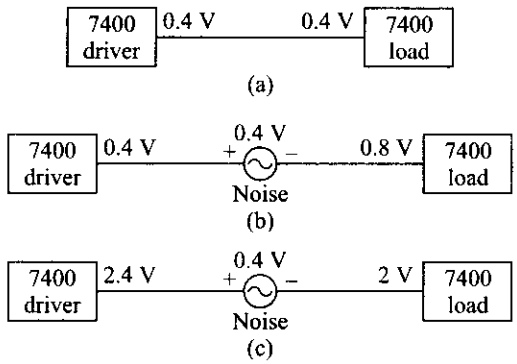


Fig. 14.15 **(a) TTL driver and load, (b) False triggering into high state, (c) False triggering into low state**

In most environments, the induced noise voltage is less than 0.4 V, and we get no false triggering of the TTL load.

For instance, Fig. 14.15a shows a low output from the TTL driver. If no noise voltage is induced on the connecting wire, the input voltage to the TTL load is 0.4 V, as shown. In a noisy environment, however, it is possible to have 0.4 V of induced noise on the connecting wire for either the low state (Fig. 14.15b) or the high state (Fig. 14.15c). Either way, the TTL load has an input that is on the verge of being unpredictable. The slightest additional noise voltage may produce a false change in the output state of the TTL load.

Standard Loading

A TTL device can source current (high output) or sink current (low output). Data sheets of standard TTL devices indicate that any 7400 series device can sink up to 16 mA, designated

$$I_{OL,max} = 16\text{ mA}$$

and can source up to 400 μA, designated

$$I_{OH,max} = -400\ \mu\text{A}$$

(Again, a minus sign means that the conventional current is out of the device, and a plus sign means that it is into the device.) As discussed earlier, the worst-case TTL input currents are

$$I_{IL,max} = -1.6\text{ mA} \quad I_{IH,max} = 40\ \mu\text{A}$$

Since the maximum output currents are 10 times larger than the input currents, we can connect up to 10 TTL emitters to any TTL output.

As an example, Fig. 14.16a shows a low output voltage (worst case). Notice that a single TTL driver is connected to 10 TTL loads (only the input emitters are shown). Here you see the TTL driver sinking 16 mA, the sum of the 10 TTL load currents. In the low state, the output voltage is guaranteed to be 0.4 V or less. If you try connecting more than 10 emitters, the output voltage may rise above 0.4 V under worst-case conditions. If this happens, the low-state operation is no longer reliable. Therefore, 10 TTL loads are the maximum that a manufacturer allows for guaranteed low-state operation.

Figure 14.16b shows a high output voltage (worst case) with the driver sourcing 400 μA for 10 TTL loads of 40 μA each. For this source current, the output voltage is guaranteed to be 2.4 V or greater under worst-case conditions. If you try to connect more than 10 TTL loads, you will exceed $I_{OH,max}$ and high-state operation becomes unreliable.

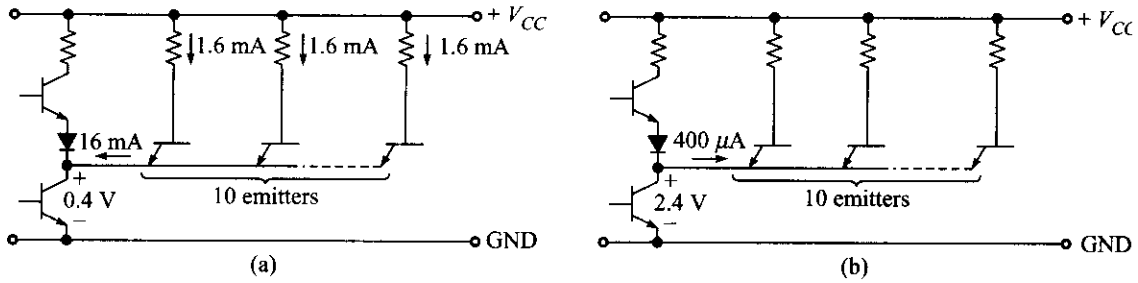


Fig. 14.16 **(a) Low-state fanout, (b) High-state fanout**

Loading Rules

Figure 14.17 shows the output-input profiles for different types of TTL. The output profiles are on the left, and the input profiles are on the right. These profiles are a concise summary of the voltages and currents for each TTL type. Start with the profiles of Fig. 14.17a; these are for standard TTL. On the left, you see the profile of output characteristics. The high output window is from 2.4 to 5 V with up to 400 μA of source current; the low output window is from 0 to 0.4 V with up to 16 mA of sink current. On the right, you see the input profile of a standard TTL device. The high window is from 2 to 5 V with an input current of 40 μA, while the low window is from 0 to 0.8 V with an input current of 1.6 mA.

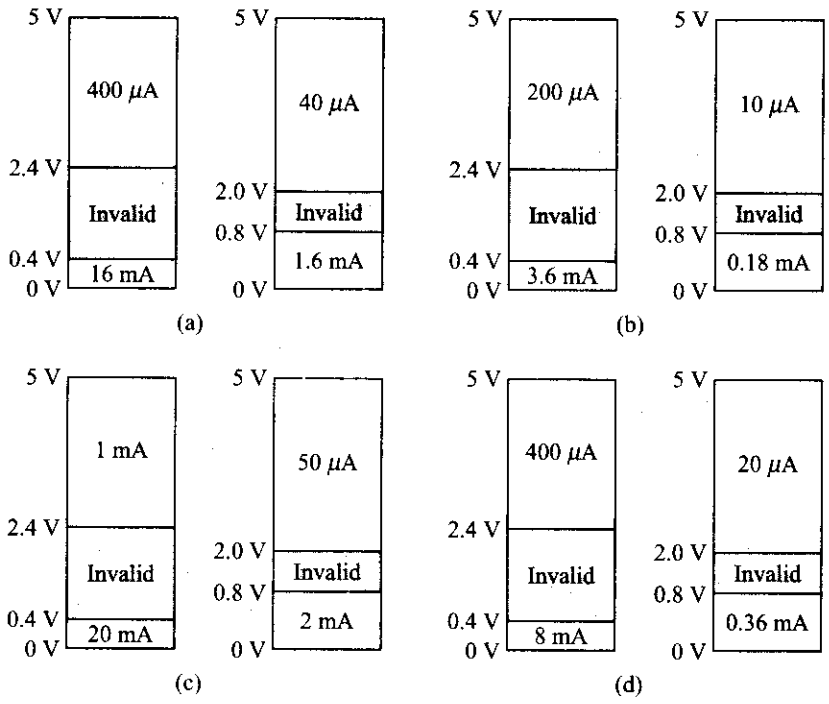


Fig. 14.17 **TTL output-input profiles: (a) Standard TTL, (b) Low-power TTL, (c) Schottky TTL, (d) Low-power Schottky TTL**

Standard TTL devices are compatible because the low and high output windows fit inside the corresponding input window. In other words, 2.4 V is always large enough to be a high input to a TTL load, and 0.4 V is always small enough to be a low input. Furthermore, you can see at a glance that the available source current is 10 times the required high-state input current, and the available sink current is 10 times the required low-state input current. The maximum number of TTL loads that can be reliably driven under worst-case conditions is called the *fanout*. With standard TTL, the fanout is 10 because one TTL driver can drive 10 TTL loads.

The remaining figures all have identical voltage windows. The output states are always 0 to 0.4 and 2.4 to 5 V, while the input states are 0 to 0.8 and 2 to 5 V. For this reason, all the TTL types are compatible; this means you can use one type of TTL as a driver and another type as a load.

The only differences in the TTL types are the currents. You can see in Fig. 14.17a to *d* that the input and output currents differ for each TTL type. For instance, a low-power Schottky TTL driver (see Fig. 14.17d) can source 400 μA and sink 8 mA; a low-power Schottky load requires input currents of 20 μA (high state) and 0.36 mA (low state). These numbers are different from standard TTL (Fig. 14.17a) with its output currents of 400 μA and 16 mA and its input currents of 40 μA and 1.6 mA.

Incidentally, notice that the profiles of high-speed TTL are omitted in Fig. 14.17 because Schottky TTL has replaced high-speed TTL, in virtually all applications. If you need high-speed TTL data, consult manufacturers' catalogs.

By analyzing Fig. 14.17*a* to *d* (plus the data sheets for high-speed TTL), we can calculate the fanout for all possible combinations. Table 14.4 summarizes these fanouts, which are useful if you ever have to mix TTL types.

Read Table 14.3 as follows. The TTL types have been abbreviated; 74 stands for 7400 series (standard), 74H for 74H00 series (high speed), and so forth. Drivers are on the left, and loads are on the right. Pick the driver, pick the load, and read the fanout at the intersection of the two. For instance, the fanout of a standard device (74) driving low-power Schottky devices (74LS) is 20. As another example, the fanout of a low-power device (74L) driving high-speed devices (74H) is only 1.

Table 14.3 Fanouts

TTL Driver	*TTL Load*				
	74	*74H*	*74L*	*74S*	*74LS*
74	10	8	40	8	20
74H	12	10	50	10	25
74L	2	1	20	1	10
74S	12	10	100	10	50
74LS	5	4	40	4	20

SELF-TEST

7. Why should TTL gate inputs never be left floating?
8. What is a TTL input profile?
9. What is the delay time for a 74LS04?

14.4 TTL OVERVIEW

Let's take a look at the logic functions available in the 7400 series. This overview will give you an idea of the variety of gates and circuits found in the TTL family. As a guide, Appendix 3 lists some of the 7400 series devices.

NAND Gates

The NAND gate is the backbone of the 7400 series. All devices in this series are derived from the 2-input NAND gate shown in Fig. 14.10. To produce 3-, 4-, and 8-input NAND gates, the manufacturer uses 3-, 4-, and 8-emitter transistors. Because they are so basic, NAND gates are the least expensive devices in the 7400 series.

NOR Gates

To get other logic functions the manufacturer modifies the basic NAND-gate design. For instance. Fig. 14.18 shows a 2-input NOR gate. Here Q_5 and Q_6 have been added to basic NAND-gate design. Since Q_2 and Q_6 are in parallel, we get the OR function, which is followed by inversion to get the NOR function.

When A and B are both low, the bases of Q_1 and Q_5 are pulled low; this cuts off Q_2 and Q_6. Then Q_3 acts as an emitter-follower, and we get a high output.

If A or B is high, Q_1 and Q_5 are cut off, forcing Q_2 or Q_6 to turn on. When this happens, Q_4 saturates and pulls the output down to a low voltage.

With more transistors, a manufacturer can produce 3- and 4-input NOR gates. (*Note*: A TTL 8-input NOR gate is not available.)

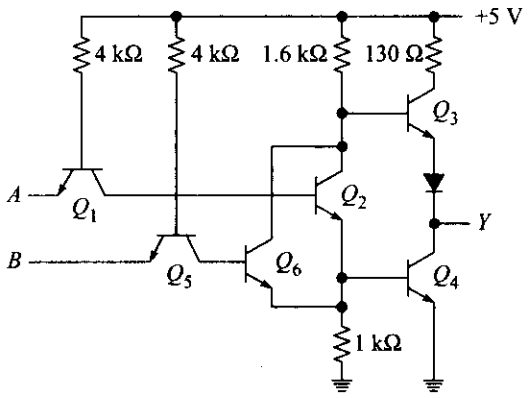


Fig. 14.18 TTL NOR gate

AND and OR Gates

To produce the AND function, another inverting stage is inserted in the basic NAND-gate design. The extra inversion converts the NAND gate to an AND gate. The available TTL AND gates are the 7408 (quad 2-input), 7411 (triple 3-input), and 7421 (dual 4-input).

Similarly, another inverting stage can be inserted in the NOR gate of Fig. 14.18; this converts the NOR gate to an OR gate. The only available TTL OR gate is the 7432 (quad 2-input).

Buffer Drivers

All IC buffer can source and sink more current than a standard TTL gate. As an example, the 7437 is a quad 2-input NAND buffer, meaning four 2-input NAND gates optimized to get high output currents. Each gate has the following worst-case currents:

$$I_{IL} = -1.6 \text{ mA} \quad I_{IH} = 40\ \mu\text{A}$$
$$I_{OL} = 48 \text{ mA} \quad I_{OH} = -1.2 \text{ mA}$$

The input currents are the same as those of a 7400 (standard TTL NAND gate), but the output currents are 3 times as high. This means that a 7437 can drive heavier loads. In other words, the fanout of a 7437 is 3 times that of a 7400. Appendix 3 includes several other buffer-drivers.

Hex schmitt-trigger (discussed in Chapter 7) inverter IC 7414 is shown in Fig. 14.19a.

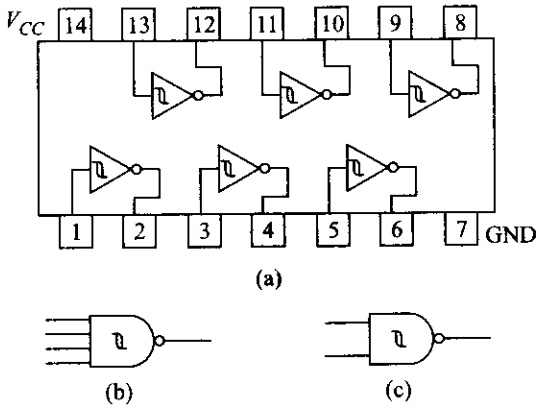


Fig. 14.19 **(a) Hex Schmitt-trigger inverters, (b) 4-input NAND Schmitt trigger, (c) 2-input NAND Schmitt trigger**

AND-or-INVERT Gates

Figure 14.20 shows the schematic diagram of an AND-OR-INVERT circuit. Here, Q_1, Q_2, Q_3 and Q_4 form the basic 2-input NAND gate of the 7400 series. By adding Q_5 and Q_6, we convert the basic NAND gate to an AND-OR-INVERT gate. Both Q_1 and Q_5 act as 2-input AND gates; Q_2 and Q_6 produce ORing and inversion. Because of this, the circuit is logically equivalent to Fig. 14.20b. This circuit represents 2-input, 2-wide AND-OR-INVERT gate. It is 2-input as each AND gate has 2 inputs and it is 2-wide because there are 2 AND gates at input stage. Figure 14.21a shows the schematic diagram of an expandable AND-OR-INVERT gate. As the name suggests, many such gates put together can expand the width at the input side. The difference between this and preceding AND-OR-INVERT gate (Fig. 14.20) is the collector and emitter pin brought outside the package.

Since Q_2 and Q_6 are the key to the ORing operation, we are being given access to the internal ORing function. By connecting other gates to these new inputs, we can expand the width of the AND-OR-INVERT gate.

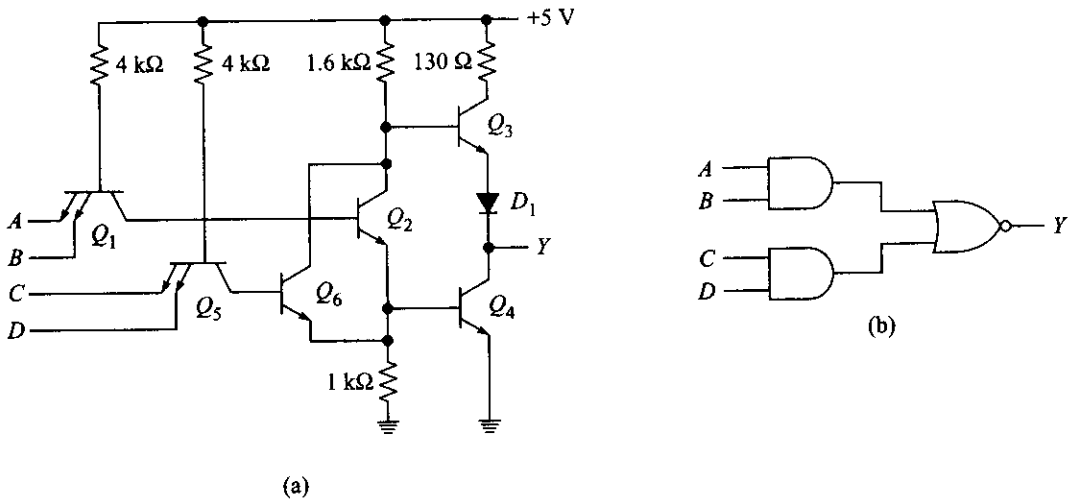


Fig. 14.20 **(a) AND-OR-INVERT schematic diagram, (b) Circuit**

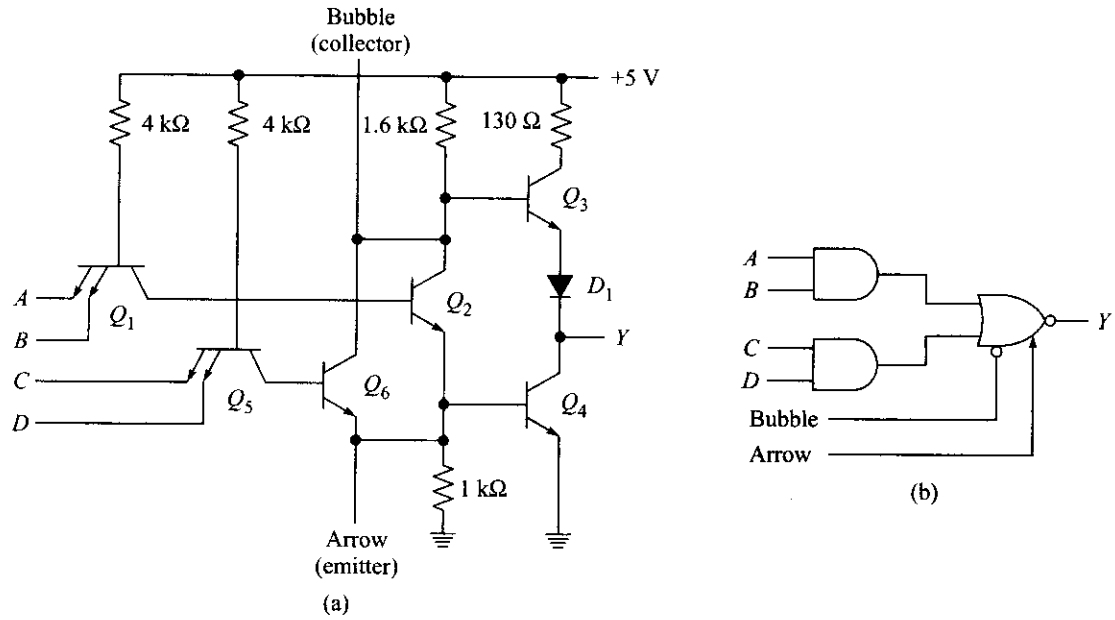


Fig. 14.21 **(a) Expandable AND-OR-INVERT gate, (b) Logic symbol**

Figure 14.21b shows the logic symbol for an expandable AND-OR-INVERT gate. The arrow input represents the emitter and the bubble stands for the collector. Table 14.4 lists the expandable AND-OR-INVERT gates in the 7400 series.

Table 14.4 **Expandable AND–OR–INVERT Gates**

Device	*Description*
7450	Dual 2-input 2-wide
7453	2-input 4-wide
7455	4-input 2-wide

SELF-TEST

10. What is the value of a buffer-driver?
11. What is an application for a Schmitt trigger?
12. What is the "width" of an AND-OR-INVERT gate?

14.5 OPEN-COLLECTOR GATES

Instead of a totem-pole output, some TTL devices have an *open-collector output*. This means they use only the lower transistor of a totem-pole pair. Figure 14.22a shows a 2-input NAND gate with an open-collector output. Because the collector of Q_4 is open, a gate like this will not work properly until you connect an external pull-up resistor, shown in Fig. 14.22b.

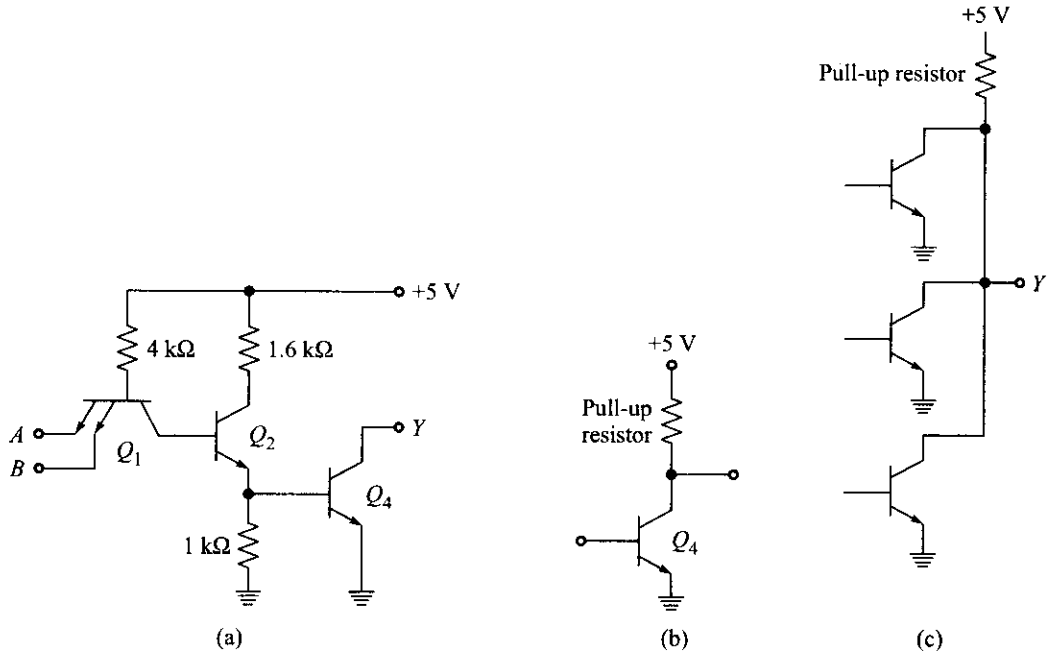


Fig. 14.22 **Open-collector TTL: (a) Circuit, (b) Pull-up resistor, (c) Open-collector outputs connected to a pull-up resistor**

The outputs of open-collector gates can be wired together and connected to a common pull-up resistor. For instance, Fig. 14.22c shows three TTL devices connected to the pull-up resistor. This is known as *wire-OR* (some called it wire-AND). A connection like this has the advantage of combining the output of three devices without using a final OR gate (or AND gate). The combining is done by a direct connection of the three outputs to the lower end of the common pull-up resistor. This is very useful when many devices are wire-ORed together. For instance, in some systems the outputs of 16 open-collector devices are connected to a pull-up resistor.

The big disadvantage of open-collector gates is their slow switching speed. Why is it slow? Because the pull-up resistance is a few kilohms, which results in a relatively long time constant when it is multiplied by the stray output capacitance. The slow switching speed of open-collector TTL devices is worst when the output goes from low to high. Imagine all three transistors going into cutoff in Fig. 14.22c. Then any capacitance across the output has to charge through the pull-up resistor. This charging produces a relatively slow exponential rise between the low and high state.

SELF-TEST

13. What must be connected to the output of an open-collector TTL gate?
14. Open-collector gates have (slower, faster) switching times.

14.6 THREE-STATE TTL DEVICES

Using a common pull-up resistor with open-collector devices is called *passive pull-up* because the supply voltage pulls the output voltage up to a high level when all the transistors of Fig. 14.22c are cut off. There is another approach known as *active pull-up*. It uses a modified totem-pole output to speed up the charging of stray output capacitance. The effect is to dramatically lower the charging time constant, which means the output voltage can rapidly change from its low to its high stage.

Why Standard TTL Will Not Work

If you try to wire-OR standard TTL gates, you will destroy one or more of the devices. Why? Look at Fig. 14.23 for an example of bad design. Notice that the output pins of two standard TTL devices are connected. If the output of the second device is low, Q_4 is on and appears approximately like a short circuit. If, at the same time, the output of the first device is in the high state, then Q_1 acts as an emitter follower that tries to pull the output voltage to a high level. Since Q_1 and Q_4 are both conducting heavily, only 130 Ω remains between the supply voltage and ground. The final result is an excessive current that destroys one of the TTL devices.

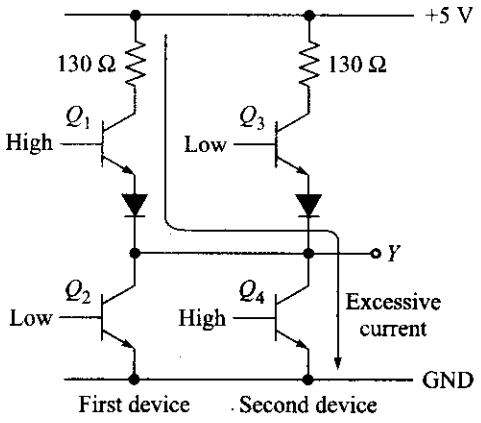


Fig. 14.23 **Direct connection of TTL outputs produces excessive current**

Low DISABLE Input

As you have seen, wire-ORing standard TTL devices will not work because of destructive currents in the output stages. This inability to wire-OR ordinary totem-pole devices is what led to *three-state (tri-state) TTL*, a new breed of totem-pole devices introduced in the early 1970s. With three-state gates, we can connect totem-pole outputs directly without destroying any devices. The reason for wanting to use totem-pole outputs is to avoid the loss of speed that occurs with open-collector devices.

Figure 14.24 shows a simplified drawing for a three-state inverter. When DISABLE is low, the base and collector of Q_6 are pulled low. This cuts off Q_7 and Q_8. Therefore, the second emitter of Q_1 and the cathode of D_1 are floating. For this condition, the rest of the circuit acts as an inverter: a low A input forces Q_2 and Q_5 to cut off, while Q_3 and Q_4 turn on, producing a high output. On the other hand, a high A input forces Q_2 to turn on, which drives Q_5 on and produces a low output. Table 14.5 summarizes the operation for low DISABLE.

Table 14.5 **Three-State Inverter**

Disable	*A*	*Y*
0	0	1
0	1	0
1	*X*	Hi-Z

High DISABLE Input

When DISABLE is high, the base and collector of Q_6 go high, which turns on Q_7 and Q_8. Ideally, the collector of Q_8 is pulled down to ground. This causes the base and collector of Q_1 to go low, cutting off Q_2 and Q_5.

Also Q_3 is off because of the clamping action of D_1. In other words, the base of Q_3 is only 0.7 V above ground, which is insufficient to turn on Q_3 and Q_4.

With both Q_4 and Q_5 off, the Y output is floating. Ideally, this means that the Thévenin impedance looking back into the Y output approaches infinity. Table 14.5 summarizes the action for this high-impedance state. As shown, when DISABLE is high, input A is a don't care because it has no effect on the Y output. Furthermore, because of the high output impedance, the output line appears to be disconnected from the rest of the gate. In effect, the output line is floating.

In conclusion, the output of Fig. 14.24 can be in one of three states: low, high, or floating.

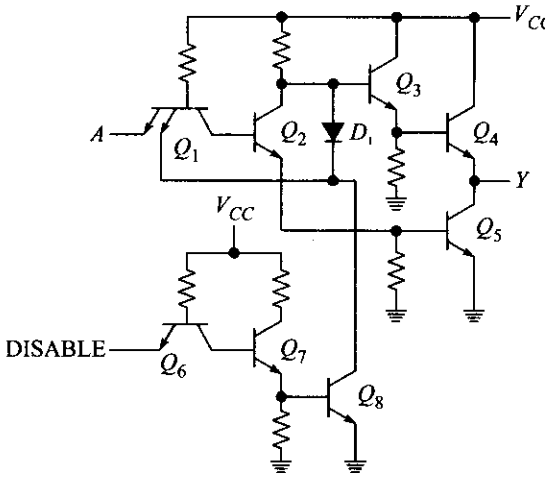


Fig. 14.24 **Three-state inverter**

Logic Symbol

Figure 14.25a is an equivalent circuit for the three-state inverter. When DISABLE is low, the switch is closed and the circuit acts as an ordinary inverter. When DISABLE is high, the switch is open and the Y output is floating or disconnected.

Figure 14.25b shows the logic symbol for a three-state inverter. When you see this symbol, remember that a low DISABLE results in normal inverter action, but a high DISABLE floats the Y output.

Three-State Buffer

By modifying the design, we can produce a three-state buffer, whose logic symbol is shown in Fig. 14.25c. When DISABLE is low, the circuit acts as a noninverting buffer, so that $Y = A$. But when DISABLE is high, the output floats. The three-state buffer is equivalent to an ordinary switch. When DISABLE is low, the switch is closed. When DISABLE is high, the switch is open.

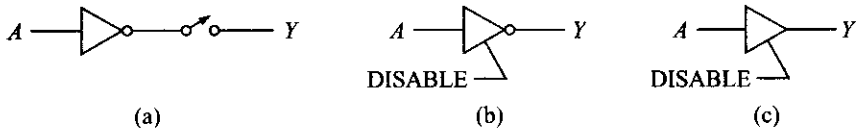


Fig. 14.25 **Three-state logic diagrams: (a) Equivalent circuit of inverter, (b) Logic symbol of inverter, (c) Logic symbol of buffer**

The 74365 is an example of a commercially available three-state hex noninverting buffer. This IC contains six buffers with three-state outputs. It is ideal for organizing digital components around a *bus*, a group of wires that transmits binary numbers between registers.

Bus Organization

Figure 14.26 shows some registers connected to a common bus. The three-state buffers control the flow of binary data between the registers. For instance, if we want the contents of register A to appear on the bus, all we have to do is make DISABLE low for register A but high for registers B and C. Then all the three-state

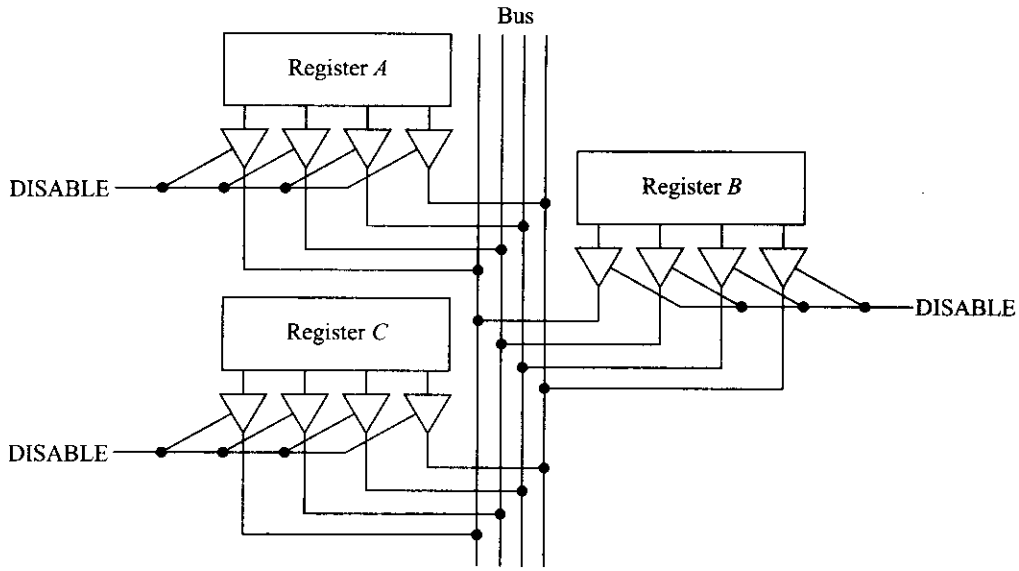


Fig. 14.26 **Three-state bus control**

switches on register *A* are closed, while all other three-state switches are open. As a result, only the contents of register *A* appear on the bus.

The idea in any bus-organized system is to make DISABLE high for all registers except the register whose contents are to appear on the bus. In this way, dozens of registers can time-share the same transmission path. Not only does this reduce the amount of wiring, but also it has simplified the architecture and design of computers and other digital systems. Refer to simple computer design discussed in Chapter 16.

SELF-TEST

15. Why are three-state gates used in conjunction with computer buses?

14.7 EXTERNAL DRIVE FOR TTL LOADS

To drive a TTL load with an external source, you need to satisfy the TTL input requirements for voltage and current. For standard TTL in the low state, this means an input voltage between 0 and 0.8 V with a current of approximately 1.6 mA. In the high state, the voltage has to be from 2 to 5 V with a current of approximately 40 μA. Let us take a brief look at some of the ways to drive a TTL load.

Switch Drive

Figure 14.27 shows the preferred method for driving a TTL input from a switch. With the switch open, the input is pulled up to +5 V. In the worst case, only 40 μA of input current exists. Therefore, the voltage

appearing at the input pin is slightly less than the supply voltage because of the small voltage drop across the pull-up resistor:

$$V_i = 5\text{ V} - (40\ \mu\text{A})(1\ \text{k}\Omega) = 4.96\text{ V}$$

This is well above the minimum requirement of 2 V, which is fine because it means that the noise immunity is excellent.

When the switch is closed, the input is pulled down to ground. In the worst case, the input current is 1.6 mA. This sink current creates no problem because it flows through the closed switch to ground. The noise immunity is fine because the input voltage is 0 V, well below the maximum allowable value of 0.8 V.

Size of Pull-Up Resistance

A pull-up resistance of 1 kΩ is nominal. You can use other values. Here are some of the factors to consider when you are selecting a pull-up resistor. In Fig. 14.27, the current drain with a closed switch is

$$I = \frac{5\text{ V}}{1\ \text{k}\Omega} = 5\text{ mA}$$

The smaller the pull-up resistance, the larger the current drain. At some point, too much current drain becomes a problem for the power supply, so you have to use a resistance that is large enough to keep the current drain to tolerable levels.

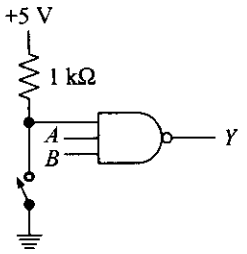


Fig. 14.27 **Switch drive for TTL input**

On the other hand, too large a pull-up resistance causes speed problems. The worst case occurs when the switch is opened. For instance, if the input capacitance is 10 picofarads (pF) in Fig. 14.27, the time constant is

$$RC = (1\ \text{k}\Omega)(10\text{ pF}) = 10\text{ ns}$$

The larger the pull-up resistance, the larger the time constant. A larger time constant means a slower switching speed because the input capacitance has to charge through the pull-up resistance.

Pull-up resistances between 1 and 10 kΩ are typical. They result in current drains and time constants that are acceptable in most applications.

Transistor Drive

Figure 14.28a shows another way to drive a TTL laid. This time, we are using a transistor switch to control the state of the TTL input. When V_i is low, the transistor is off and is equivalent to an open switch. Then the TTL input is pulled up to +5 V through a resistance of 1 kΩ. When V_i is high, the transistor is on and is equivalent to a closed switch. In this case, it easily sinks the 1.6 mA of input current.

The transistor inverts the control signal V_i. If this is objectionable, you can insert an inverter as shown in Fig. 14.28b. Now, the double inversion produces an in-phase control signal at the TTL input.

Operational Amplifier Drive

Sometimes, you want to use the output of an operational amplifier (OA) to control a TTL input. Because OAs typically use split-supply voltages of +15 and −15 V, you have to be careful how you connect to the TTL

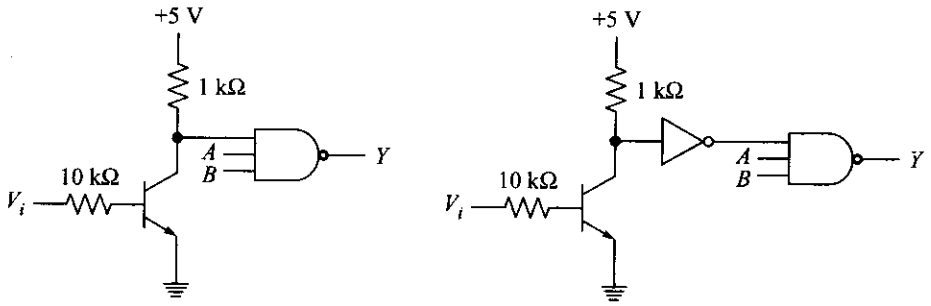


Fig. 14.28 **(a) Transistor drive for TTL input, (b) Inverter eliminates transistor inversion**

load. Figure 14.29 shows one way to use the output of a 741 to control a TTL input. The output of the OA ideally swings from +15 to −15V. The positive swing closes the transistor switch, producing a TTL input of approximately 0 V. The negative swing drives the transistor into cutoff, producing a TTL input of +5V.

Notice the diode in the base circuit. It protects the base against excessive reverse voltage. The data sheet of a 2N3904 indicates an absolute maximum base-emitter voltage rating of

$$V_{BE,\max} = -6 \text{ V}$$

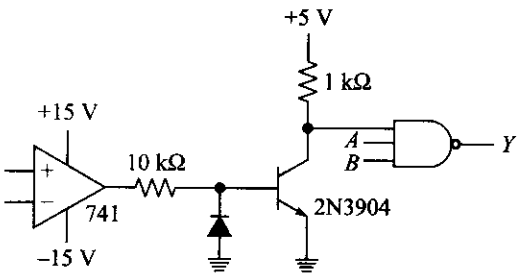


Fig. 14.29 **Op amp and transistor drive for TTL input**

Since the negative output of the OA approaches −15 V, we need to use a protective diode as shown between the base and ground. This diode clamps the base voltage at approximately −0.7 V on the negative swing.

Comparator Drive

Figure 14.30a shows the schematic diagram for a typical comparator, an IC that detects when the input voltage is positive or negative. Notice two things. First, a supply voltage of +15 V is typically used with this kind of device. Second, the comparator has an open-collector output transistor, Q_s. This sink transistor can be connected to any supply voltage.

Figure 14.30b shows how to connect an LM339 (typical comparator) to a TTL load. Because of the open-collector output, we can connect the output pin of the comparator to a supply voltage of +5 V through a pull-up resistance of 1 kΩ. When V_i is positive, the sink transistor goes off and the TTL input is pulled high. When V_i is negative, the sink transistor goes on and the TTL output is pulled low.

SELF-TEST

16. What factors influence the size of the resistor in Fig. 14.29?
17. What is the purpose of the diode in Fig. 14.31?

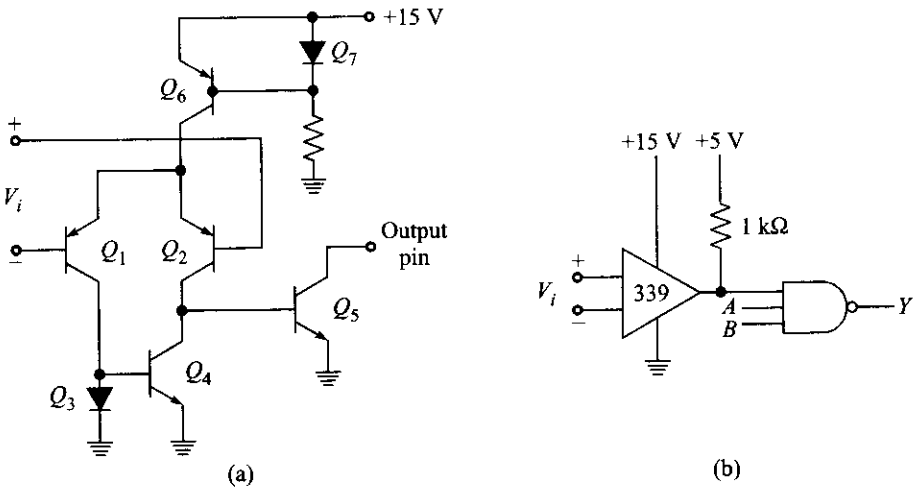


Fig. 14.30 **(a) Schematic diagram of comparator, (b) Interfacing an LM339 to a TTL input**

14.8 TTL DRIVING EXTERNAL LOADS

Because standard TTL can sink up to 16 mA, you can use a TTL driver to control an external load such as a relay, an LED, etc. Figure 14.31a illustrates the idea. When the TTL output is high, there is no load current. But when the TTL output is low, the lower end of R_L is ideally grounded. This sets up a load current of approximately

$$I_L = \frac{5\text{ V}}{R_L}$$

Since standard TTL can sink a maximum of 16 mA, the load resistance is limited to a minimum value of about

$$R_L = \frac{5\text{ V}}{16\text{ mA}} = 312\ \Omega$$

Driving an LED

Figure 14.31b is another example. Here a TTL circuit drives an LED. When the TTL output is high, the LED is dark. When the TTL output is low, the LED lights up. If the LED voltage drop is 2 V, the LED current for a low TTL output is approximately

$$I_L = \frac{5\text{ V} - 2\text{ V}}{270\ \Omega} = 11.1\text{ mA}$$

Supply Voltage Different from +5 V

If you need to use a supply voltage different from +5 V, you can use an open-collector TTL device. For instance, Fig. 14.32a on the next page shows an open-collector gate driving a load resistor that is returned to +15 V. Since an open-collector device can sink a maximum of 16 mA, the minimum load resistance in Fig. 14.32a is slightly less than 1 kΩ.

If you want more than 16 mA of load current, you can use an external transistor, as shown in Fig. 14.32b.

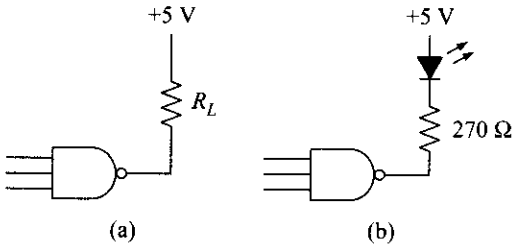


Fig. 14.31 **(a) TTL output drives load resistor, (b) TTL output drives LED**

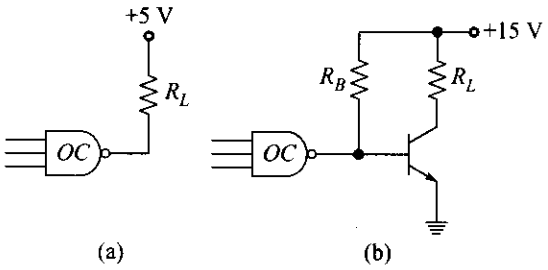


Fig. 14.32 **(a) Open-collector device allows a higher supply voltage, (b) Transistor increases current drive**

When the open-collector device has a low output, the external transistor goes off and the load current is zero. When the device has a high output, the external transistor goes on and the load current is maximum.

SELF-TEST

18. Could the resistor R_L in Fig. 14.34 be replaced with a red LED?

14.9 74C00 CMOS

National Semiconductor Corporation pioneered the 74C00 series, a line of CMOS circuits that are pin-for-pin and function-for-function compatible with TTL devices of similar numbers. For instance, the 74C00 is a quad 2-input NAND gate, the 74C02 is a quad 2-input NOR gate, and so on. This CMOS family contains a variety of small-scale integration (SSI) and medium-scale integration (MSI) chips that allow you to replace many TTL designs by the comparable CMOS designs. This is useful if you are trying to build battery-powered equipment. The 74HC00 series is the high-speed CMOS family.

NAND Gate

Figure 14.33 shows a CMOS NAND gate. The complementary design of input and output stages is typical of CMOS devices. Notice that Q_1 and Q_2 form one complementary connection; Q_3 and Q_4 form another. Visualize these transistors as switches. Then a low A input will close Q_1 and open Q_2; a high A input will open Q_1 and close Q_2. Similarly, a low B input will open Q_3, and close Q_4; a high B input will close Q_3 and open Q_4.

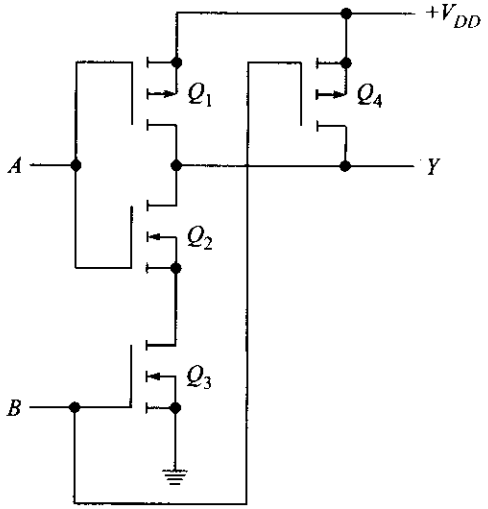


Fig. 14.33 **CMOS NAND gate**

In Fig. 14.33, the Y output is pulled up to the supply voltage when either Q_1 or Q_4 is conducting. The output is pulled down to ground only when Q_2 and Q_3 are conducting. If you keep this in mind, it simplifies the following discussion.

Case 1 Here A is low and B is low. Because A is low, Q_1 is closed. Therefore, Y is pulled high through the small resistance of Q_1.

Case 2 Now A is low and B is high. Since A is still low, Q_1 remains closed and Y stays in the high state.

Case 3 The A input is high and the B is low. Because B is low, Q_4 is closed. This pulls Y up to the supply voltage through the small resistance of Q_4.

Case 4 The A is high, and the B is high. When both inputs are high, Q_2 and Q_3 are closed, pulling the output down to ground.

Table 14.6 summarizes all input-output possibilities. As you can see, this is the truth table of a positive NAND gate. The output is low only when all inputs are high. To produce the positive AND function, we can connect the output of Fig. 14.33 to a CMOS inverter.

Table 14.6 CMOS NAND Gate

A	B	Y
Low	Low	High
Low	High	High
High	Low	High
High	High	Low

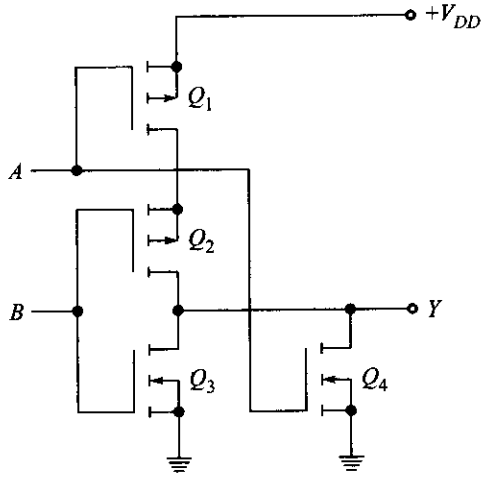


Fig. 14.34 CMOS NOR gate

NOR Gate

Figure 14.34 shows a CMOS NOR gate. The output goes high only when Q_1 and Q_2 are closed. The output goes low if either Q_3 or Q_4 is closed. There are four possible cases:

Case 1 The A is low, and the B is low. For both inputs low, Q_1 and Q_2 are closed. Therefore, Y is pulled high though the small series resistance of Q_1 and Q_2.

Case 2 The A is low, and the B is high. Because B is high, Q_3 is closed, pulling the output down to ground.

Case 3 The A is high, and the B is low. With A high, Q_4 is closed. The closed Q_4 pulls the output low.

Table 14.7

A	B	Y
Low	Low	High
Low	High	Low
High	Low	Low
High	High	Low

Case 4 The A is high, and the B is high. Since A is still high, Q_4 is still closed and the output remains low.

Table 14.7 summarizes these possibilities. As you can see, this is the truth table of a positive NOR gate. The output is low when any input is high.

Propagation Delay Time

A standard CMOS gate has a propagation delay time t_p of approximately 25 to 100 ns, with the exact value depending on the power supply voltage and other factors. As you recall, t_p is the time it takes for the output

of a gate to change after its inputs have changed. When two or more CMOS gates are cascaded, you have to add the propagation delay times to get the total. For instance, if you cascade three CMOS gates each with a t_p of 50 ns, then the total propagation delay time is 150 ns.

Power Dissipation

The *static power dissipation* of a device is its average power dissipation when the output is constant. The static power dissipation of a CMOS gate is in nanowatts. For instance, a 74C00 has a power dissipation of approximately 10 nanowatts (nW) per gate. This dissipation equals the product of supply voltage and leakage current, both of which are dc quantities.

When a CMOS output changes from the low state to the high state (or vice versa), the average power dissipation increases. Why? The reason is that during a transition between states, there is a brief period when both MOSFETs are conducting. This produces a spike (quick rise and fall) in the supply current. Furthermore, during a transition, any stray capacitance across the output has to be charged before the output voltage can change. This capacitive charging draws additional current from the power supply. Since power equals the product of supply voltage and device current, the instantaneous power dissipation increases, which means the average power dissipation is higher.

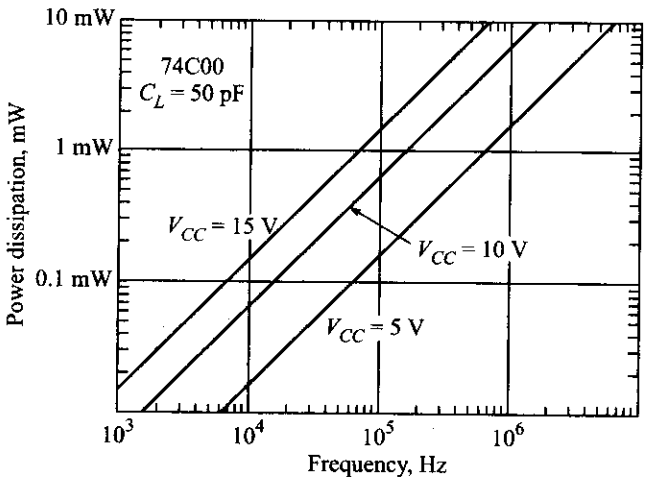


Fig. 14.35 **Active power dissipation of a 74C00**

The average power dissipation of a CMOS device whose output is continuously changing is called the *active power dissipation*. How large is the active power dissipation? This depends on the frequency at which the output is switching states. When the operating frequency increases, the current spikes occur more often and active power dissipation increases. Figure 14.35 shows the active power dissipation of a 74C00 versus frequency for a load capacitance of 50 pF. As you see, the power dissipation per gate increases with frequency and supply voltage. For frequencies in the megahertz region, the gate dissipation approaches or exceeds 10 mW (TTL gate dissipation). For CMOS to have an advantage over TTL, you operate CMOS devices at lower frequencies.

Another way to reduce power dissipation is to decrease the supply voltage. But this has adverse effects because it increases propagation time and decreases noise immunity. Although CMOS devices can work over a range of 3 to 15 V, the best compromise for speed, noise immunity, and overall performance is a supply voltage from 9 to 12 V. From now on, we assume a supply voltage of 10 V, unless otherwise specified.

Incidentally, notice the use of V_{CC} rather than V_{DD} for the supply voltage. This is a carryover from TTL circuits. You will find V_{CC} on the data sheets for 74C00 devices.

54C00 Series

Any device in the 74C00 series works over a temperature range of −40 to +85°C. This is adequate for most commercial applications. The 54C00 series (for military applications) works over a temperature range of −55 to +125°C. Although 54C00 devices can be substituted for 74C00 devices, they are rarely used commercially because of their much higher cost.

74HC00 Devices

The main disadvantage of CMOS devices is their relatively long propagation delay times. This places a limit on the maximum operating frequency of system. The 74HC00 series is a CMOS series of devices that are pin-for-pin and function-for-function compatible with TTL devices. These devices have the advantage of higher speed (less propagation delay time).

74HCT00 Devices

These are also high-speed CMOS circuits designed to be directly compatible with TTL devices. That is, they can be connected *directly* to any TTL circuit. Interfacing TTL and CMOS devices is discussed in Secs. 14.11 and 14.12.

CD4000 Series

RCA was the first to introduce CMOS devices. The original devices were numbered from CD4000 upward. This 4000 series was soon replaced by the 4000A series (called conventional) and the 4000B series (called the buffered type). The 4000A and B series are widely used; they have many functions not available in the 74C00 series. The main disadvantage of 4000 devices is their lack of pin-for-pin and function-for-function compatibility with TTL.

SELF-TEST

19. What kind of transistors are shown in Fig. 14.33?
20. Why is the NOR gate in Fig. 14.34a CMOS device?

14.10 CMOS CHARACTERISTICS

74C00 series devices are guaranteed to work reliably over a temperature range of −40 to +85°C and over a supply range of 3 to 15 V. In the discussion that follows, *worst case* means the parameters are measured under the worst conditions of temperature and voltage.

Floating Inputs

When a TTL input is floating, it is equivalent to a high input. You can use a floating TTL input to simulate a high input; but as already pointed out, it is better to connect unused TTL inputs to the supply voltage. This prevents the floating leads from picking up stray noise in the environment.

If you try to float a CMOS input, however, not only do you set up a possible noise problem, but, much worse, you produce excessive power dissipation. Because of the insulated gates, a floating input allows the gate voltage to drift into the linear region. When this happens, excessive current can flow through push-pull stages.

The absolute rule with CMOS devices, therefore, is to *connect all input pins*. Most of or all the inputs are normally connected to signal lines. If you happen to have an input that is unused, connect it to ground or the supply voltage, whichever prevents a stuck output state. For instance, with a positive NOR gate you should ground an unused input. Why? Because returning the unused NOR input to the supply voltage forces the output into a stuck low state. On the other hand, grounding an unused NOR input allows the other inputs to control the output.

With a positive NAND gate, you should connect an unused input to the supply voltage. If you try grounding an unused NAND input, you disable the gate because its output will stick in the high state. Therefore, the best thing to do with an unused NAND input is to tie it to the supply voltage. A direct connection is all right: CMOS inputs can withstand the full supply voltage.

Easily Damaged

Because of the thin layer of silicon dioxide between the gate and the substrate, CMOS devices have a very high input resistance, approximately infinite. The insulating layer is kept as thin as possible to give the gate more control over the drain current. Because this layer is so thin, it is easily destroyed by excessive gate voltage.

Aside from directly applying an excessive gate voltage, you can destroy the thin insulating layer in more subtle ways. If you remove or insert a CMOS device into circuit while the power is on, transient voltages caused by inductive kickback and other effects may exceed the gate voltage rating. Even picking up a CMOS IC may deposit enough charge to exceed its gate voltage rating.

One way to protect against overvoltages is to include zener diodes across the input. By setting the zener voltage below the breakdown voltage of the insulating layer, manufacturers can prevent the gate voltage from becoming destructively high. Most CMOS ICs include this form of zener protection.

Figure 14.36 shows a typical *transfer characteristic* (input-output graph) of a CMOS inverter. When the input voltage is in the low state, the output voltage is in the high state. As the input voltage increases, the output remains in the high state until a threshold is reached. Somewhere near an input voltage of $V_{CC}/2$, the output will switch to the low state. Then any input voltage greater than $V_{CC}/2$ holds the output in the low state.

This transfer characteristic is an improvement over TTL. Why? Because the indeterminate region is much smaller. As you can see, the input voltage has to be nearly equal to $V_{CC}/2$ before the CMOS output switches states. This implies that the noise immunity of CMOS devices ideally approaches $V_{CC}/2$. Typically, noise immunity is 45 percent of V_{CC}.

Also notice how much better defined the low and high output states are. When CMOS loads are used, the CMOS source and sink transistors have almost no voltage drop because there is almost no input

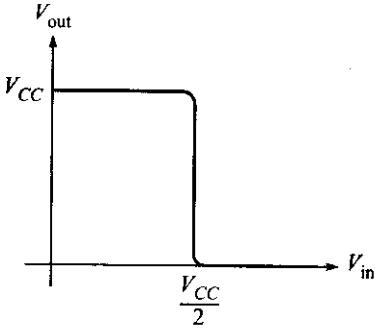


Fig. 14.36 **Typical transfer characteristic of a CMOS gate**

current to a CMOS load. Therefore, the static currents are extremely small. For this reason, the high output voltage is approximately equal to V_{CC}, and the low output voltage is approximately at ground. Stated another way, the logic swing between the low and high output states approximately equals the supply voltage, a considerable advantage for CMOS over TTL.

Compatibility

CMOS devices are compatible with one another because the output of any CMOS device can be used as the input to another CMOS device, as shown in Fig. 14.37a. For instance, Fig. 14.37b shows the output stage of a CMOS driver connected to the input stage of a CMOS load. The supply voltage is +10 V. Ideally, the input switching level is +5 V. Since the CMOS driver has a low output, the CMOS load has a high input.

Similarly, Fig. 14.37c shows a high CMOS driver output. This is more than enough voltage to drive the CMOS load with a high-state input. In fact, the noise immunity typically approaches 4.5 V (from 45 percent of V_{CC}). Any noise picked up on the connecting line between devices would need a peak value of more than 4.5 V to cause unwanted switching action.

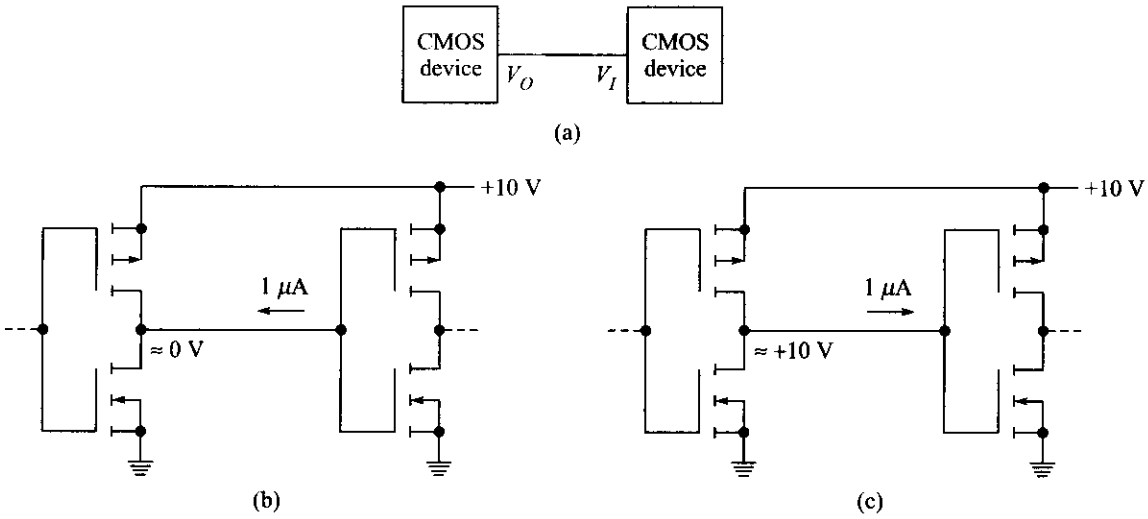


Fig. 14.37 **(a) Output of CMOS device can drive input of another CMOS device, (b) Sink current, (c) Source current**

Sourcing and Sinking

When a standard CMOS driver output is low (Fig. 14.37b), the input current to the CMOS load is only 1 microampere (μA) (worst case shown on data sheet). The input current is so low because of the insulated gates. This means that the CMOS driver has to sink only 1 μA. Similarly, when the driver output is high (Fig. 14.37c), the CMOS driver is sourcing 1 μ.A.

In symbols, here are the worst-case input currents for CMOS devices:

$$I_{IL,\max} = -1\ \mu A \quad I_{IH,\max} = 1\ \mu A$$

We use these values to calculate the fanout.

Fanout

The fanout of CMOS devices depends on the kind of load being driven. In Sec. 14.11, we discuss CMOS devices driving TTL devices. Now we want to concentrate on CMOS driving CMOS. Data sheets for 74C00 series devices give the following output currents for CMOS driving CMOS:

$$I_{OL,\max} = 10\ \mu A \quad I_{OH,\max} = -10\ \mu A$$

Since the worst-case input current of a CMOS device is only 1 μA, a CMOS device can drive up to 10 CMOS loads. Therefore, you can use a fanout of 10 for CMOS-to-CMOS connections. This value is reliable under all operating conditions.

SELF-TEST

21. Why must care be taken when using CMOS devices?
22. What is the transfer characteristic of a CMOS inverter?

14.11 TTL-TO-CMOS INTERFACE

The word *interface* refers to the way a driving device is connected to a loading device. In this section, we discuss methods for interfacing CMOS devices to TTL devices. Recall that TTL devices need a supply voltage of 5 V, while CMOS devices can use any supply voltage from 3 to 15 V. Because the supply requirements differ, several interfacing schemes may be used. Here are a few of the more popular methods.

Supply Voltage at 5 V

One approach to TTL/CMOS interfacing is to use +5 V as the supply voltage for both the TTL driver and the CMOS load. In this case, the worst-case TTL output voltages (Fig. 14.38a) are almost compatible with the worst-case CMOS input voltages (Fig. 14.38b). *Almost*, but not quite. There is no problem with the TTL low-state window (0 to 0.4 V) because it fits inside the CMOS low-state window (0 to 1.5 V). This means the CMOS load always interprets the TTL low-state drive as a low.

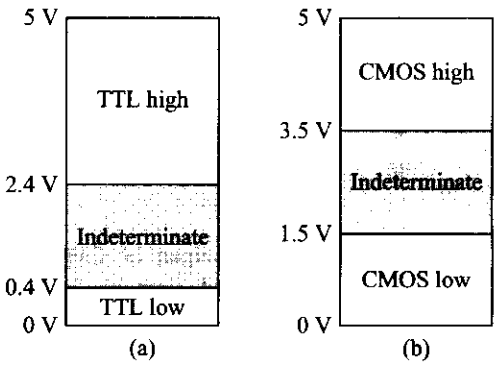


Fig. 14.38 (a) TTL output profile, (b) CMOS input profile

The problem is in the TTL high state, which can be as low as 2.4 V (see Fig. 14.38a). If you try using a TTL high-state output as the input to a CMOS device, you get indeterminate action. The CMOS device needs at least 3.5 V for a high-state input (Fig. 14.38b). Because of this, you cannot get reliable operation by connecting a TTL output directly to a CMOS input. You have to do something extra to make the two different devices compatible.

What do you do? The standard solution is to use a pull-up resistor between the TTL driver and the CMOS load, as shown in Fig. 14.39. What effect does the pull-up resistor have? It has almost no effect on the low

state, but it does raise the high state to approximately +5 V. For instance, when the TTL output is low, the lower end of the 3.3 kΩ is grounded (approximately). Therefore, the TTL driver sinks a current of roughly

$$I = \frac{5\text{ V}}{3.3\text{ k}\Omega} = 1.52\text{ mA}$$

When the TTL output is in the high state, however, the output voltage is pulled up to +5 V. Here is how it happens. As before, the upper totem-pole transistor actively pulls the output up to +2.4 V (worst case). Because of the pull-up resistor, however, the output rises above +2.4 V, which forces the upper totem-pole transistor into cutoff. The pull-up action is now passive because the supply voltage is pulling the output voltage up to +5 V through the pull-up resistor.

The gate capacitance of the CMOS load has to be charged through the pull-up resistor. This slows down the switching action. If speed is important, you can decrease the pull-up resistance. The minimum resistance is determined by the maximum sink current of the TTL device: $I_{OL,\max} = 16$ mA. In the worst case the supply voltage may be as high as 5.25 V, so the minimum resistance is

$$R_{\min} = \frac{5.25\text{ V}}{16\text{ mA}} = 328\ \Omega$$

The nearest standard value is 330 Ω, which you should consider the absolute minimum value for the pull-up resistor. And you would use this only if switching speed were critical. In many applications, a pull-up resistance of 3.3 kΩ is fine.

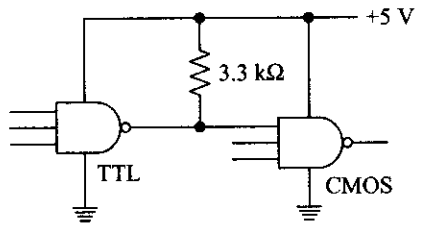


Fig. 14.39 **TTL driver and CMOS load**

Incidentally, the other inputs of the TTL driver and CMOS load (Fig. 14.39) are connected to signal lines not shown. Also, the use of 3-input gates is arbitrary. You can interface gates with any number of inputs. If more than one TTL chip is being interfaced to the CMOS load, connect each TTL driver to a separate pull-up resistor and CMOS input.

Different Supply Voltages

CMOS performance deteriorates at lower voltages because the propagation delay time increases and the noise immunity decreases. Therefore, it is better to run CMOS devices with a supply voltage between 9 and 12 V. One way to use a higher supply voltage is with an open-collector TTL driver (Fig. 14.40). Recall that the output stage of an open collector TTL device consists only of a sink transistor with a floating collector. In Fig. 14.40. this open collector is connected to a supply voltage of +12 V through a pull-up resistance of 6.8 kΩ. Likewise, the CMOS device now has a supply voltage of +12 V.

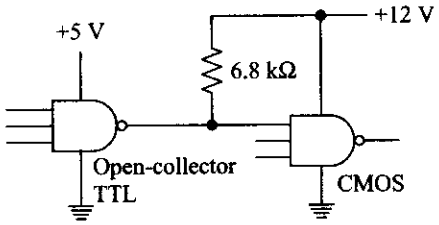


Fig. 14.40 **Open-collector TTL driver allows higher CMOS supply voltage**

When the TTL output is low, we can visualize a ground on the lower end of the pull-up resistor. Therefore, the TTL device has to sink approximately

$$I_{sink} = \frac{12\text{ V}}{6.8\text{ k}\Omega} = 1.76\text{ mA}$$

When the TTL output is high, the open-collector output rises passively to +12 V. In either case, the TTL outputs are compatible with the CMOS input states.

The passive pull-up in Fig. 14.40 produces slower switching action than before. For instance, with a gate input capacitance of 10 pF, the pull-up time constant is

$$RC = (6.8\text{ k}\Omega)(10\text{ pF}) = 68\text{ ns}$$

If this is a problem, reduce the pull-up resistance to its minimum allowable value of

$$R_{min} = \frac{12\text{ V}}{16\text{ mA}} = 750\ \Omega$$

Then the pull-up time constant decreases to

$$RC = (750\ \Omega)\ (10\text{ pF}) = 7.5\text{ ns}$$

CMOS Level Shifter

Figure 14.41 shows a 40109, called a *level shifter*. The input stage of the chip uses a supply voltage of +5 V, while the output stage uses +12 V. In other words, the input stage interfaces with TTL, and the output stage interfaces with CMOS.

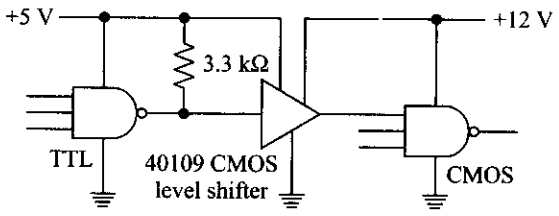


Fig. 14.41 **CMOS level shifter allows the use of 5-V and 12-V supplies**

In Fig. 14.41, a standard TTL device drives the level shifter. This produces active TTL pull-up to at least +2.4 V. Beyond this level, the pull-up resistor takes over and raises the voltage to +5 V, which ensures a valid high-state input to the level shifter. The output side of the level shifter connects to +12V (this can be changed to any voltage from 3 to 15 V). Since the CMOS load runs off of +12 V, it has better propagation delay time and noise immunity.

In summary, TTL has to run off of +5 V, but CMOS does better with a supply voltage of +12 V. This is the reason for using a level shifter between the TTL driver and the CMOS load.

SELF-TEST

23. What is the purpose of the 3.3-kΩ resistor in Fig. 14.41?
24. Where is a CMOS level shifter used?

14.12 CMOS-TO-TTL INTERFACE

In this section, we discuss methods for interfacing CMOS devices to TTL devices, Again, the problem is to shift voltage levels until the CMOS output states fall inside the TTL input windows. Specifically, we have to make sure that the CMOS low-state output is always less than 0.8 V, the maximum allowable TTL low-state input voltage. Also, the CMOS high-state output must always be greater than 2 V, the minimum allowable TTL high-state input voltage.

Supply Voltage at 5 V

One approach is to use +5 V as the supply voltage for the driver and the load, as shown in Fig. 14.42. A direct interface like this forces you to use a low-power Schottky TTL load (or two low-power TTL loads). Why? Because a low-power Schottky device has these worst-case input currents:

$$I_{IL,\max} = -360\ \mu A \quad I_{IH,\max} = 20\ \mu A$$

Data sheets for 74C00 devices list these worst-case output currents for CMOS driving TTL:

$$I_{OL,\max} = 360\ \mu A \quad I_{OH,\max} = -360\ \mu A$$

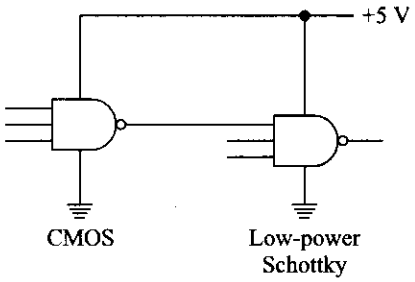


Fig. 14.42 **CMOS driver and low-power Schottky TTL load**

This tells us that a CMOS drive can sink 360 μA in the low state, exactly the input current for a low-power Schottky TTL devices. On the other hand, the CMOS driver can source 360 μA, which is more than enough to handle the high-state input current (only 20 μA). So the sink current limits the CMOS/74LS fanout to 1.

CMOS can also drive low-power TTL devices. The limiting factor again is the sink current. Low-power TTL has a worst-case low-state input current of 180 μA. Since a CMOS driver can sink 360 μA, it can drive two low-power TTL devices. Briefly stated, the CMOS/74L fanout is 2.

CMOS cannot drive standard TTL directly because the latter requires a low-state input current of −1.6 mA, for too much current for a CMOS device to sink without entering the TTL indeterminate region. The problem is that the sink transistor of a CMOS device is equivalent to a resistance of approximately 1.11 kΩ (worst case). The CMOS output voltage equals the product of 1.6 mA and 1.11 kΩ, which is 1.78 V. This is too large to be low-state TTL input.

Using a CMOS Buffer

Figure 14.43 shows how to get around the fanout limitation just discussed. The CMOS driver now connects directly to a CMOS buffer, a chip with larger output currents. For instance, a 74C902 is a hex buffer, or six CMOS buffers in a single package. Each buffer has these worst-case output currents:

$$I_{OL,\max} = 3.6\ \text{mA} \quad I_{OH,\max} = 800\ \mu A$$

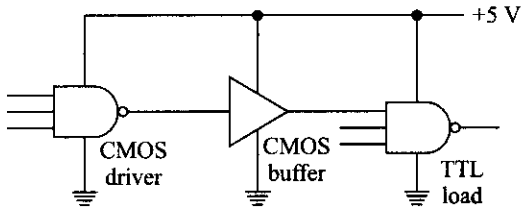


Fig. 14.43 **CMOS buffer can drive standard TTL load**

Since a standard TTL load has a low-state input current of 1.6 mA and a high-state input current of 40 μA, a 74C902 can drive two standard TTL loads. If you use one-sixth of a 74C902 in Fig. 14.43, the CMOS/TTL fanout is 2. Other available buffers are the CD4049A (inverting), CD4050A (noninverting), 74C901 (inverting), etc.

Different Supply Voltages

CMOS buffers like the 74C902 can use a supply voltage of 3 to 15V and an input voltage of −0.3 to 15 V. The input voltage can be greater than the supply voltage without damaging the device. For instance, you can use a high-state input of +12 V even though the supply voltage is only +5 V.

Figure 14.44 shows how to use the previous idea to our advantage. Here, the supply pin of the CMOS driver is connected to +12 V. On the other hand, the supply pin of the CMOS buffer is connected to +5 V to produce the TTL interface. Therefore, the input to the CMOS buffer will be as much as +12 V, even if its supply voltage is only +5 V. The fanout of this interface is still two standard TTL loads.

Open-Drain Interface

Recall open-collector TTL devices. The output stage consists of a sink transistor with a floating collector. Similar devices exist in the CMOS family. Known as *open-drain devices*, these have an output stage consisting only of a sink MOSFET. An example is the 74C906, a hex open-drain buffer.

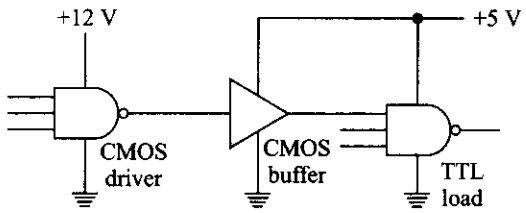


Fig. 14.44 **CMOS driver runs better with 12-V supply**

Figure 14.45 shows how an open-drain CMOS buffer can be used as an interface between a CMOS driver and a TTL load. The supply voltage for most of the buffer is +12 V. The open drain, however, is connected to a supply voltage of +5 V through a pull-up resistance of 3.3 kΩ. This has the advantage that both the CMOS driver and the CMOS buffer run off of +12 V, except for the open-drain output which provides the TTL interface.

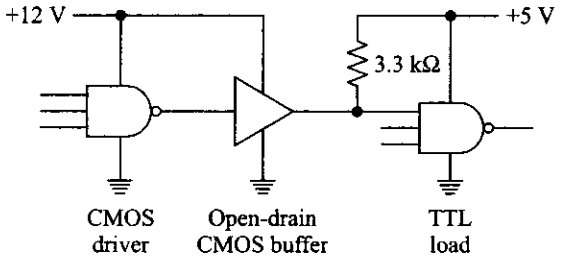


Fig. 14.45 **Open-drain CMOS buffer increases sink current**

SELF-TEST

25. Can a CMOS circuit drive a 74LS04 directly? What about a 7404?
26. CMOS output voltage levels are well within the profile of TTL input voltage levels. Why can't the CMOS drive the TTL directly?

14.13 CURRENT TRACERS

Figure 14.46a shows a solder bridge shorting a node to ground. When you trigger the logic pulser, the logic probe remains dark because the node is stuck in the low state. A logic pulser and logic probe will help you locate stuck nodes, but they cannot tell you the exact location of the short.

Figure 14.46b shows a *current tracer*, a troubleshooting tool than can detect current in a wire or circuit-board trace. Although it touches the wire, the current tracer does not make electric contract. Inside its

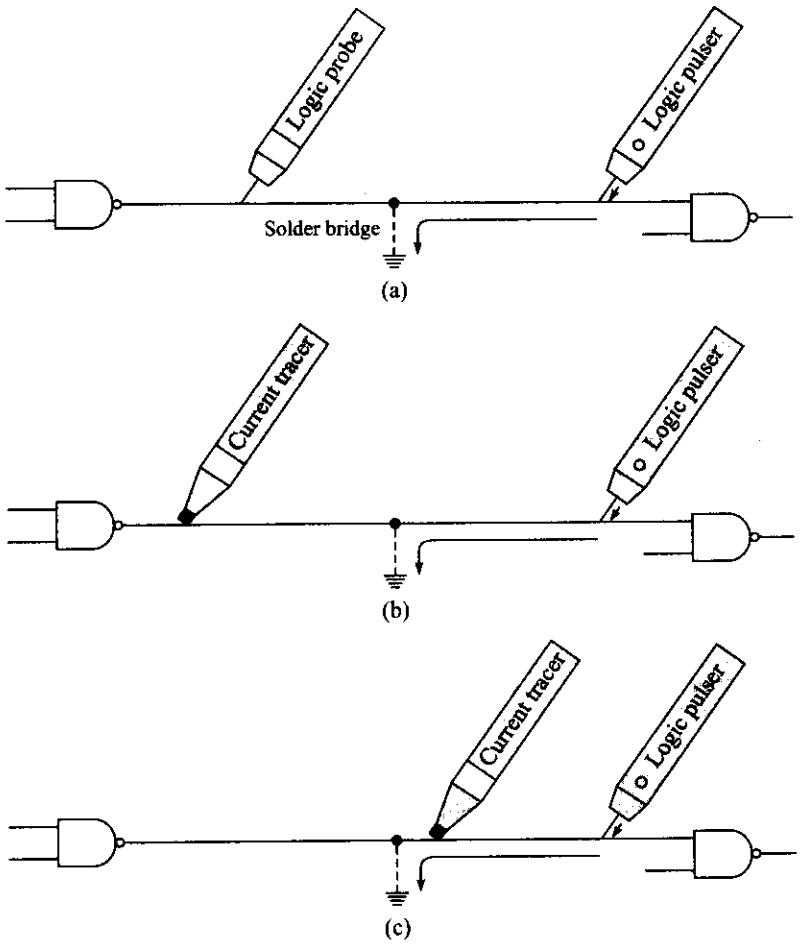


Fig. 14.46 **(a) Solder bridge shorts node to ground, (b) Current tracer will not detect any current, (c) Current tracer will detect current**

blunt insulated tip is a small pick-up coil than can detect the magnetic field around a wire carrying current. Therefore, if there is any current through the wire, the current tracer lights up.

In Fig. 14.46b, each time you trigger the logic pulser, a conventional current flows through its tip to ground along the path shown. The current tracer will not detect this current because it is touching another part of the wire.

If you move the current tracer between the ground and the logic pulser as shown in Fig. 14.46c, the current tracer will light up. The critical position for the current tracer is directly over the short. Move left, and the current tracer goes out. Move right, and it turns on. When this happens, you know the current tracer is directly over the trouble. During troubleshooting, a visual check at this critical location usually reveals the nature of the trouble (a solder bridge, in this discussion).

That's the basic idea behind a current tracer. You use the logic pulser and logic probe to find stuck nodes. Then you use the current tracer to locate the exact position along a trace where the short is.

SUMMARY

A chip is a small piece of semiconductor material with microminiature circuits on its surface. Small-scale integration (SSI) refers to chips with less than 12 gates. Medium-scale integration (MSI) means 12 to 100 gates per chip. Large-scale integration (LSI) refers to more than 100 gates on a chip.

The 7400 series is a line of standard TTL chips. This bipolar family contains a variety of compatible SSI and MSI devices. One way to recognize TTL design is the multiple-emitter input transistors and the totem-pole output transistors. The standard TTL chip has a power dissipation of about 10 mW per gate and a propagation delay time of around 10 ns.

By including a Schottky diode in parallel with the collector-base terminals, manufacturers produce Schottky TTL. This eliminates saturation delay time because it prevents the transistors from saturating on. Numbered from 74S00, these devices have a power dissipation of 20 mW per gate and a propagation delay time of approximately 3 ns.

By increasing internal resistances and including Schottky diodes, manufacturers can produce low-power Schottky TTL devices (numbered from 74LS00). A low-power Schottky TTL gate has a power dissipation of around 2 mW per gate and a propagation delay time of approximately 10 ns. Low-power Schottky TTL is the most widely used of the TTL types.

A floating TTL input is equivalent to a high input. Do not use floating TTL inputs when you are operating in an electrically noisy environment. Floating inputs may pick up enough noise voltage to produce unwanted changes in the output states.

A standard TTL gate can sink 16 mA and source 400 mA. Since the maximum input currents are 1.6 mA (low state) and 40 μA (high state), standard TTL has a fanout of 10, meaning that one standard TTL gate can drive 10 others. Fanout has different values when you mix TTL types.

Open-collector devices have only the pull-down transistor; the pull-up transistor is omitted. Because of this, open-collector devices can be wire—ORed through a common pull-up resistor. This connection is inherently slow because the time constant is relatively long.

Three-state devices have replaced open-collector devices in most applications because they are much faster. These newer devices have a control input that can turn off the device. When this happens, the output floats and presents a high impedance to whatever it is connected to. Three-state devices are widely used for connecting to buses.

A CMOS inverter uses complementary MOSFETs in a push-pull arrangement. The key advantage of CMOS devices is the low power dissipation. The main disadvantage is the slow switching speed.

The 74C00 series is a line of CMOS circuits that are pin-for-pin and function-for-function compatible with TTL devices. The static power dissipation of 74C00 devices is approximately 10 nanowatts (nW) per gate. Active power dissipation is higher because of the current spikes during transitions. Lower supply voltages increase the propagation delay time and noise immunity. Higher supply voltages increase the power dissipation. The best compromise is a supply voltage from 9 to 12 V. The 74HC00 series is a line of high-speed CMOS devices. The CD4000 series is another line of CMOS devices with many functions not available in the 74C00 series.

CMOS devices are guaranteed to work reliably over a temperature range of –40 to +85°C and a supply range of 3 to 15 V. Unused inputs should be returned to the supply voltage or to ground, depending on which connection prevents a stuck output. A floating CMOS input is poor design because it produces large power dissipation. CMOS devices have a fanout of 10 when driving other CMOS devices. By using level shifting, CMOS devices can be interfaced with TTL devices.

The 74HCT00 series is completely TTL-compatible, and special interfacing is not required.

GLOSSARY

- ***active load*** A transistor that acts as a load for another transistor.
- ***active power dissipation*** The power dissipation of a device under switching conditions. It differs from static power dissipation because of the large current spikes during output transitions.
- ***bipolar*** Having two types of charge carriers: free electrons and holes.
- ***bus*** A group of wires that transmits binary data. The data bus of a first-generation microcomputer has eight wires, each carrying 1 bit. This means that the data bus can transmit 1 byte at a time. Typically, the byte represents an instruction or data word that is moved from one register to another.
- ***chip*** A small piece of semiconductor material. Sometimes chip refers to an IC device including its pins.
- ***CMOS inverter*** A push-pull connection of *p*- and *n*-channel MOSFETs.
- ***compatibility*** Ability of the output of one device to drive the input of another device.
- ***interface*** The way a driving device is connected to a loading device. All the circuitry between the output of a device and the input of another device.
- ***fanout*** The maximum number of TTL loads that a TTL device can reliably drive.
- ***low-power Schottky TTL*** A modification of standard TTL in which larger resistances and Schottky diodes are used. The larger resistances decrease the power dissipation, and the Schottky diodes increase the speed.
- ***noise immunity*** The amount of noise voltage that causes unreliable operation. With TTL it is 0.4 V. As long as the noise voltages induced on connecting lines are less than 0.4 V, the TTL devices will work reliably.
- ***saturation delay time*** The time delay encountered when a transistor tries to come out of the saturation region. When the base drive switches from high to low, a transistor cannot instantaneously come out of hard saturation; extra carriers must first flow out of the base region.
- ***Schmitt trigger*** A digital circuit that produces a rectangular output. The input waveform may be sinusoidal, triangular, distorted, and so on. The output is always rectangular.
- ***sink*** A place where something is absorbed. When saturated, the lower transistor in a totem-pole output acts as a current sink because conventional charges flow through the transistor to ground.
- ***source*** The upper transistor of a totem-pole output acts as a source because conventional flow is out of the emitter into the load.
- ***standard TTL*** The basic TTL design. It has a power of dissipation of 10 mW per gate and a propagation delay time of 10 ns.
- ***three-state TTL*** A modified TTL design that allows us to connect outputs directly. Earlier computers used open-collector devices with their buses, but the passive pull-up severely limited the operating speed. By replacing open-collector devices with three-state devices, we can significantly reduce the switching time needed to change from the low state and the output state. The result is faster data changes on the bus, which is equivalent to speeding up the operation of a computer.

PROBLEMS

Section 14.1

14.1 For each assigned circuit in Fig. 14.47, determine the indicated current I and/or voltage V.

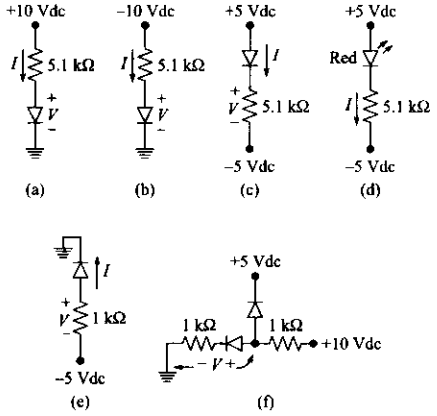


Fig. 14.47

14.2 From memory, draw the symbols for: diode, LED, *npn* BJT, *pnp* BJT.

14.3 Make a truth table for the circuit in Fig. 14.5a.

14.4 From memory, draw the simplified symbols for an *n*-channel MOSFET and a *p*-channel MOSFET.

14.5 For each assigned circuit in Fig. 14.48, determine the indicated current I and/or voltage V.

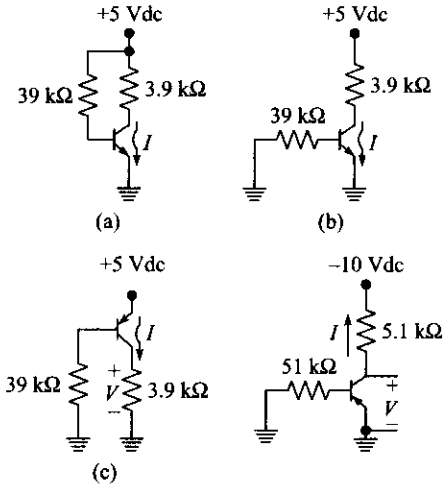


Fig. 14.48

14.6 Draw the circuit for a CMOS inverter.

14.7 For each assigned circuit in Fig. 14.49 on the next page, determine the indicated current I and/or voltage V.

Section 14.2

14.8 Figure 14.10 shows typical resistance values at room temperature. Here A is high, and B is grounded. Allowing 0.7 V for the base-emitter

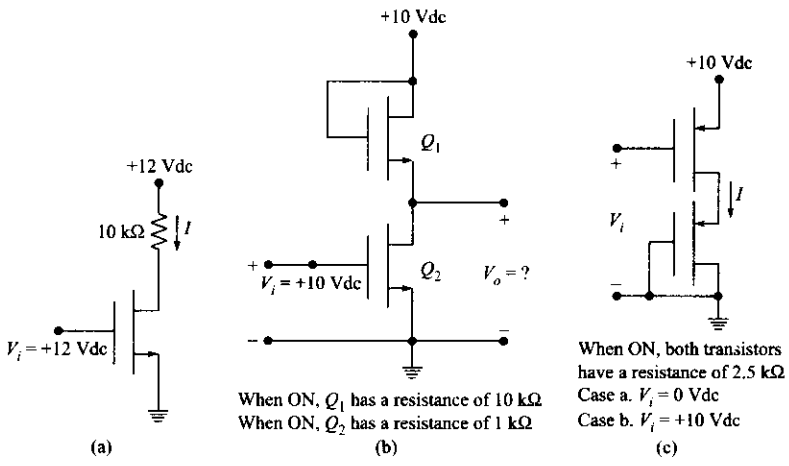


Fig. 14.49

voltage, how much current is there through the 4kΩ?

14.9 Suppose you need a TTL device with a power dissipation of less than 5 mW per gate and a delay time of less than 20 ns. What TTL type would you choose?

14.10 Use the values of Table 14.3 to calculate the total propagation delay time of three cascaded gates for each of the following TTL types:

a. Low-power
b. Low-power Schottky
c. Standard
d. High-speed
e. Schottky

Section 14.3

14.11 What is the fanout of a 74S00 device when it drives low-power TTL loads?

14.12 What is the fanout of a low-power Schottky device driving standard TTL devices?

14.13 What is the fanout of a standard TTL device driving a 74LS device?

14.14 The output of a 74LS04 is connected to the inputs of two 7404s, one 7400, and three 7410s. It seems to malfunction occasionally. What might be the problem?

14.15 What would be a simple "fix" for the circuit in Prob. 14.14?

14.16 A zero-rise-time pulse is applied to the input of a 74LS04. Its output drives a 74LS10. What is the delay time from the rising edge of the input pulse to the rising edge of the 74LS10 output?

Section 14.4

14.17 What is the fanout of a 7437 buffer when it drives standard TTL loads?

14.18 The input to a 7414 hex Schmitt trigger is a 2-V-peak sinewave. Sketch both the input and output voltages.

14.19 What is the output in Fig. 14.20 for these inputs?

a. $ABCD = 0000$
b. $ABCD = 0101$
c. $ABCD = 1100$
d. $ABCD = 1111$

14.20 Is the output Y of Fig. 14.50 low or high for these conditions?

a. Both switches open, A is low
b. Both switches closed, A is high
c. Left switch open, right switch closed, A is low
d. Left switch closed, right switch open, A is high

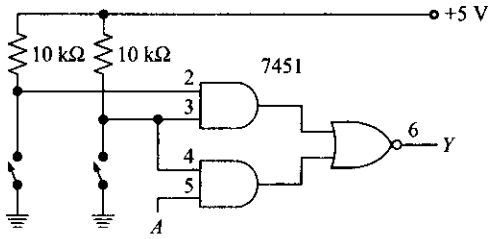


Fig 14.50

14.21 What is the value of Y in Fig. 14.51 for each of these?

a. $ABCD = 0000$
b. $ABCD = 0101$
c. $ABCD = 1000$
d. $ABCD = 1111$

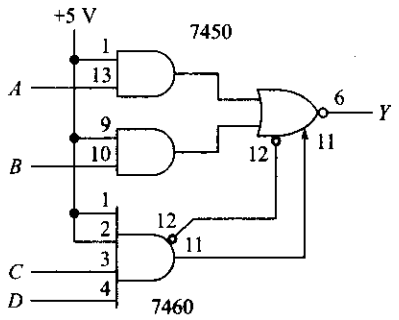


Fig. 14.51

Section 14.5

14.22 In Fig. 14.22a, $I_{OL,max} = 16$ mA. If three open-collector gates like these are wire-ORed together as shown in Fig. 14.22c, what is the minimum value of pull-up resistance needed to avoid destroying any device?

14.23 Suppose the total output capacitance is 20 pF in Fig. 14.22c. If the pull-up resistance equals 3.6 kΩ, what does the charging time constant equal?

Section 14.6

14.24 You want the contents of register *B* to appear on the bus of Fig. 14.28. What are the necessary disable values?

14.25 What are the three output conditions of a three-state gate?

14.26 Draw the logic symbol for a three-state inverter.

Section 14.7

14.27 In Fig. 14.52, what does output *Y* equal when each switch is open? When either switch is closed?

14.28 What is the current drain through the pull-up resistors when both switches are closed in Fig. 14.52? What is the time constant for each input when the switches are open?

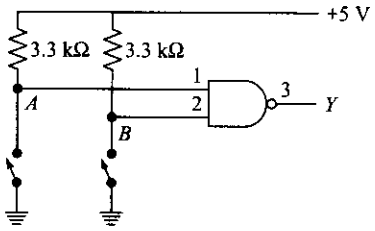


Fig. 14.52

14.29 In Fig. 14.53, what does the output *Y* equal when either switch is open? When both are closed? This is not a preferred method of driving TTL loads. Try to figure out two reasons why this circuit is not as good as the circuit shown in Fig. 14.52.

Section 14.8

14.30 In Fig. 14.54a, the TTL output voltage is 0.4 V, and the LED voltage is 2 V. What is the sink current when the LED is lighted?

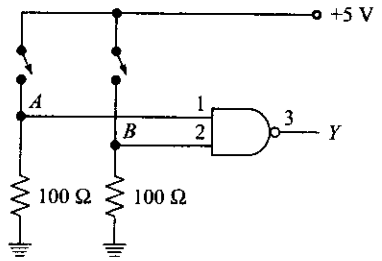


Fig. 14.53

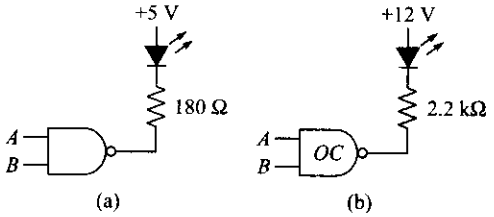


Fig. 14.54

14.31 What is the LED current in Fig. 14.54b if the LED voltage drop is 2 V and the TTL output is high? If the TTL output is 0.4 V, what is the LED current?

14.32 When switch *B* of Fig. 14.55 is closed, is the LED on or off? For this condition, what is the current in the LED?

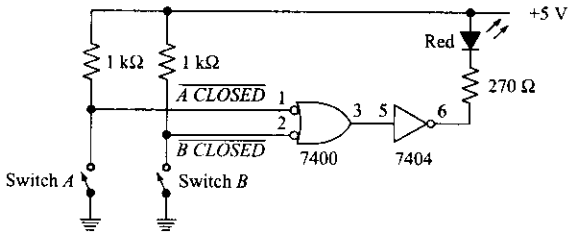


Fig. 14.55

Section 14.9

14.33 Three CMOS devices are cascaded. If each has a propagation delay time of 100 ns, what is the total propagation delay time?

14.34 A 74C00 has a load capacitance of 50 pF. If the supply voltage is 10 V, what is the active

power dissipation per gate at each of the following frequencies?

a. kHz
b. 10 kHz
c. 100 kHz

14.35 Explain the difference between 74C00, 74HC00, and 74HCT00 devices.

14.36 What are CD4000 series ICs?

Section 14.10

14.37 Figure 14.56a shows how to drive a CMOS device from a switch. As you see, the input does not float in either state. Is the output low or high when the switch is open? Is it low or high when the switch is closed?

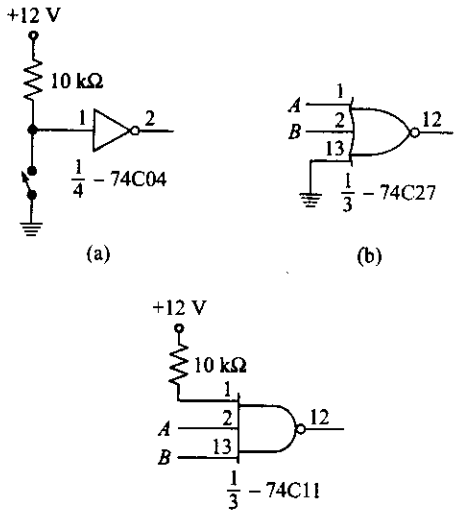


Fig. 14.56

14.38 Pin 13 is an unused input in Fig. 14.56b. As you see, it is grounded. Is the output low or high for each of these conditions?

a. *A* and *B* both low
b. *A* low and *B* high
c. *A* high and *B* low
d. *A* and *B* both high

14.39 If pin 13 is returned to the supply voltage instead of grounded in Fig. 14.56b, the circuit is useless as a NOR gate. Why?

14.40 Pin 1 is an unused input in Fig. 14.56c. As you see, it is returned to the supply voltage through a pull-up resistor. Is the output low or high for each of these conditions?

a. *A* and *B* both low
b. *A* low and *B* high
c. *A* high and *B* low
d. *A* and *B* both high

14.41 If pin 1 is grounded instead of returned to the supply voltage in Fig. 14.56c, the circuit is useless as a NAND gate. Why?

Section 14.11

14.42 If the CMOS input is low in Fig. 14.57, what is the sink current in the TTL driver? If the CMOS input is high, how much voltage drop is there across the 1.5 kΩ if the gate current is 1 μA (worst case)?

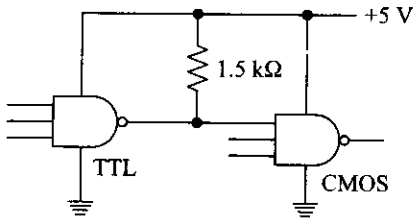


Fig. 14.57

14.43 Ideally, how much current does the open-collector driver of Fig. 14.58 have to sink when its output is low?

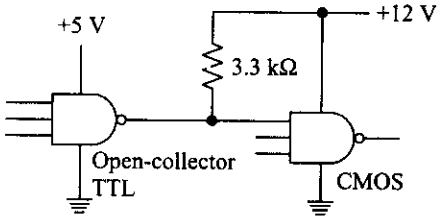


Fig. 14.58

14.44 What is the smallest acceptable value for the 3.3-kΩ resistor in Fig. 14.39 if the TTL device is a 7410 (look at maximum sink current)? What if the 7410 were replaced with a 74LS10?

14.45 The TTL in Fig. 14.40 is a 74LS10. Could a second CMOS circuit be added at the output of the 74LS10 without violating any loading rules? How many could be added?

Section 14.12

14.46 Ideally, what is the sink current I in Fig. 14.59? If the TTL load has a high-state input current of 40 μA, what is the voltage drop across the 2.2 kΩ?

14.47 If the input capacitance of the TTL load is 10 pF, what does the pull-up time constant equal in Fig. 14.59?

14.48 What is the maximum sink current for the 74C906 in Fig. 14.59?

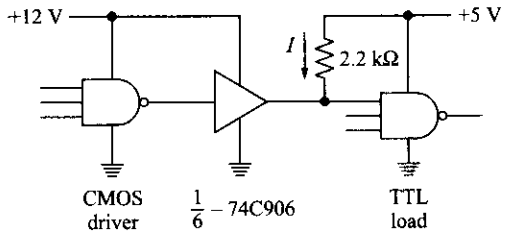


Fig. 14.59

Section 14.13

14.49 In Fig. 14.57, a logic probe indicates that the lower end of the pull-up resistor is stuck in the low state. Using a logic pulser and current tracer, you detect a current in the wire between this resistor and the TTL output. Which of the following is a possible trouble?

a. 1.5 kΩ shorted
b. 1.5 kΩ open
c. Open trace between the TTL output and the resistor
d. TTL sink transistor shorted

14.50 The lower end of the pull-up resistor (Fig. 14.57) is stuck in the high state. With a logic pulser and current tracer, you detect a current in the wire between this resistor and the CMOS input. Which of the following is a possible source of the trouble?

a. CMOS input trace shorted to supply voltage
b. CMOS input grounded
c. TTL output trace open
d. TTL output shorted to ground

Answers to Self-tests

1. The voltage across a forward-biased LED is larger.
2. *npn* and *pnp*. A *p*-channel MOSFET is the complement of an *n*-channel MOSFET.
3. Positive.
4. The active load in an NMOS IC is an *n*-channel MOSFET used in place of a resistor.
5. See Fig. 14.11.
6. 74LS00—low-power Schottky.
7. They are subject to unwanted noise that may switch the circuit.
8. It specifies acceptable high and low input voltage levels.
9. From Table 14.3, it is 10 ns.
10. It can sink or source more current than a standard gate.
11. It will produce an output waveform with very fast rise and fall times.
12. The "width" refers to the number of AND gates.
13. A resistor to +5 Vdc
14. Slower
15. They are used to facilitate connecting two or more gate outputs in parallel.
16. The factors are dc power supply current and switching time.
17. Its purpose is to protect the base-emitter of the transistor from excessive voltage.
18. No, you must also have a resistor in series with the LED to limit current; otherwise, when the transistor is on, the diode and/or the transistor will burn out.

19. Q_1 and Q_4 are p-channel MOSFETs. Q_2 and Q_3 are n-channel MOSFETs.
20. The NOR gate is CMOS because it uses both *n- and p-channel transistors.*
21. The thin oxide layer connected to the gate is easily damaged by static electricity.
22. The transfer characteristic of a CMOS inverter is a plot of input voltage versus output voltage (Fig. 14.38).
23. Its purpose is to raise the minimum TTL high output level above the lowest allowable CMOS high input level.
24. A level shifter is used between a TTL gate driving a CMOS gate; it is used to make their high and low levels compatible.
25. Yes, no
26. The CMOS has current sink and source limitations.

Applications

15

OBJECTIVES

- **Understand the multiplexing techniques used with LED displays**
- **List and describe the main sections of a frequency counter**
- **Explain how a time measurement circuit can be designed**
- **Be familiar with the basic features of the ADC0804 A/D converter**
- **Be familiar with the basic features of the ADC3511 microprocessor compatible A/D converter**
- **Discuss how to construct a digital voltmeter using the National Semi-conductor ADD3501 chip**

This chapter is intended to tie together many of the fundamental ideas presented previously by considering some of the more common digital circuit design encountered in industry. The multiplexing of digital LED displays is considered first since it requires the use of a number of different TTL circuits studied in detail in prior chapters. Digital instruments that can be used to measure time and frequency are considered next, and the concept of display multiplexing is applied here.

A number of applications using the popular ADC0804 are presented. An intergrating-type converter, the microprocessor-compatible ADC3511, is studied in detail. Then a similar converter, the ADC3501, is used to construct a digital voltmeter. In most of the applications considered, specific TTL part numbers have been specified, but in the interest of clarity, detailed designs including pin numbers have not been provided. However, it is a simple matter to consult the appropriate data sheets for this information.

In some cases, a specific part number has not been assigned; an example of this is the 1-MHz clock: oscillator shown in Fig. 15.14, or a divide-by-10 counter in the same figure. In such cases, it is left to you

to select any one of a number of divide-by-10 circuits, or to choose an oscillator circuit such as discussed in a previous chapter, on the basis of availability, cost, ease of use, compatibility with the overall system, and other factors.

15.1 MULTIPLEXING DISPLAYS

The decimal outputs of digital instruments such as digital voltmeters (DVMs) and frequency counters are often displayed using seven-segment indicators. Such indicators are constructed by using a fluorescent bar, a liquid crystal bar, or a LED bar for each segment. LED-type indicators are convenient because they are directly compatible with TTL circuits, do not require the higher voltages used with fluorescents, and are generally brighter than liquid crystals. On the other hand, LEDs do generally require more power than either of the other two types, and *multiplexing* is a technique used to reduce indicator power requirements.

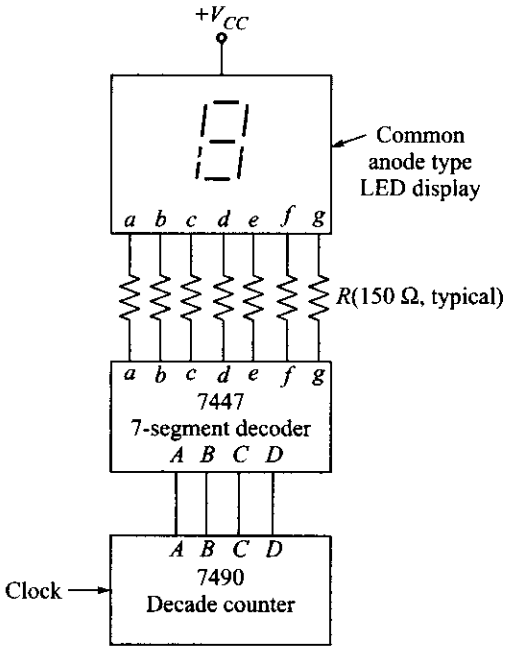
The circuit in Fig. 15.1a is a common-anode LED-type seven-segment indicator used to display a single decimal digit. The 7447 BCD to seven-segment decoder is used to drive the indicator, and the four inputs to the 7447 are the four-flip-flop outputs of the 7490 decade counter. Remember that the 7447 has active low outputs, so the equivalent circuit of an illuminated segment appears as in Fig. 15.lb. A 1-Hz square wave applied at the clock input of the 7490 will cause the counter to count upward, advancing one count each second, and the equivalent decimal number will appear on the display.

A similar single decimal digit display using a common-cathode-type LED indicator is shown in Fig. 15.2a. The seven-segment decoder used here is the 7448; its outputs are active high, and they are intended to drive buffer amplifiers since their output current capabilities are too small to drive LEDs directly. The seven *npn* transistors simply act as switches to connect $+V_{CC}$ to a segment. When an output of the 7448 is high, a transistor is on, and current is supplied to a LED segment. The equivalent circuit for an illuminated segment is shown in Fig. 15.2b. When an output of the 7448 is low, the transistor is off, and there is no segment current and thus no illumination.

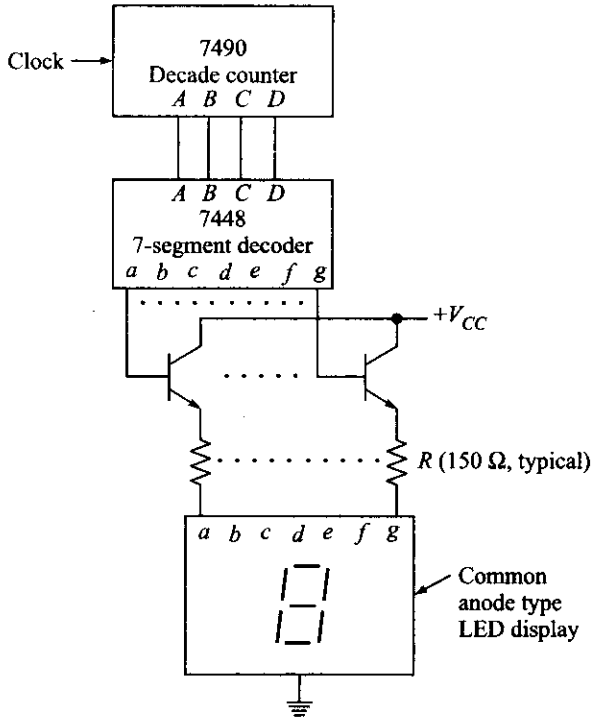
Let's take a look at the power required for the single-digit display in Fig. 15.la. A segment is illuminated whenever an output of the 7447 goes low (essentially to ground). If we assume a 2-Vdc drop across an illuminated segment (LED), a current $I = (5 - 2)/150 = 20$ mA is required to illuminate each segment. The largest current is required when the number 8 is displayed, since this requires all segments to be illuminated. Under this condition, the indicator will require $7 \times 20 = 140$ mA. The 7447 will also require about 64 mA, so a maximum of around 200 mA is required for this single digit display. An analysis of the display circuit in Fig. 15.2 will yield similar results.

A digital instrument that has a four-digit decimal display will require four of the circuits in Fig. 15.1 and thus has a current requirement of $4 \times 200 = 800$ mA. A six-digit instrument would require 1200 mA, or 1.2 A, just for the displays! Clearly these current requirements are much too large for small instruments, but they can be greatly reduced using multiplexing technique.

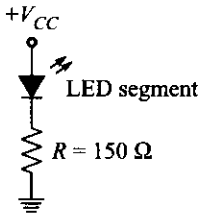
Basically, multiplexing is accomplished by applying current to each display digit in short, repeated pulses rather than continuously. If the pulse repetition rate is sufficiently high, your eye will perceive a steady illumination without any flicker. (For instance, hardly any flicker is noticeable with indicators illuminated using 60 Hz.) The single-digit display in Fig. 15.3a has +5 Vdc (and thus current) applied through a *pnp* transistor that acts as a switch. When DIGIT is high, the transistor (switch) is off, and the indicator current is zero. When DIGIT is low, the transistor is on, and a number is displayed. If the waveform in Fig. 15.3b is used as DIGIT, the transistor will be on and the segment will display a number for only 1 out of every 4 ms. Even though the display is not illuminated for 3 out of 4 ms, the illumination will appear to your eye as if it



(a) Single decimal-digit display

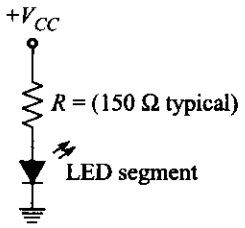


(a)



(b) Equivalent circuit for an illuminated segment

Fig. 15.1

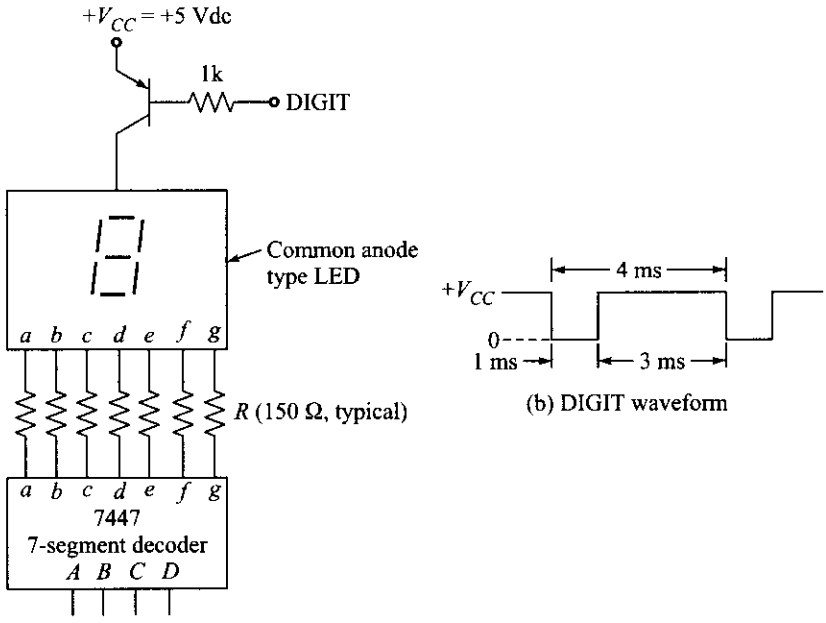


(b) Equivalent circuit for an illuminated segment

Fig. 15.2

were continuous. Since the display is illuminated with a pulse that occurs once every 4 ms, the repetition rate (*RR*) is given as *RR* = 1/0.004 = 250 Hz. As a guideline, any *RR* greater than around 50 or 60 Hz will provide steady illumination without any perceptible flicker. The great advantage here is that this single-digit display requires only one-fourth the current of a continuously illuminated display. This then is the great advantage of multiplexing!

Let's see how to multiplex the four-digit display in Fig. 15.4a. Assume that the four BCD inputs to each digit are unchanging. If the four waveforms in Fig. 15.4b are used as the four DIGIT inputs, each digit will be illuminated for one-fourth of the time and extinguished for three-fourths of the time. Looking at the time line, we see that digit 1 is illuminated during time t_1, digit 2 during time t_2, and so on. Clearly, $t_1 = t_2 = t_3 = t_4$



(a) Multiplexed display

(b) DIGIT waveform

Fig. 15.3

= $T/4$. The repetition rate is given as $RR = 1/T$, and if the rate is sufficient, no flicker will appear. For instance, if t_1 = 1 ms, then T = 4 ms, and RR = 1/0.004 = 250 Hz.

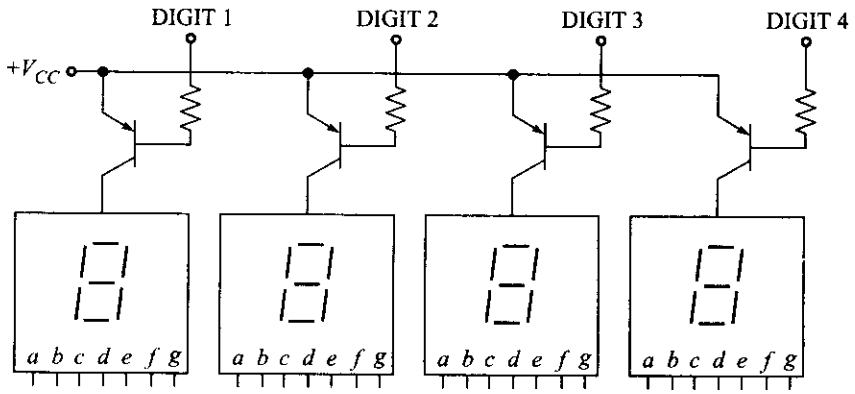
Now, here is an important concept; an illuminated digit requires 200 mA, and since only one digit is illuminated at a time, the current required from the $+V_{CC}$ supply is always 200 mA. Therefore, we are illuminating four indicators but using the current required of only a single indicator. In fact, in multiplexing displays in this way, the power supply current is simply the current required of a single display, no matter how many displays are being multiplexed!

Example 15.1 Explain the timing for a six-digit display that has a repetition rate of 125 Hz.

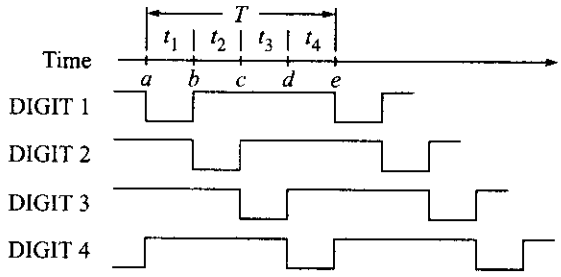
Solution An RR of 125 Hz means that all digits must be serviced once eyery $\frac{1}{125}$ = 8 ms. Dividing the time equally among the six digits means that each digit will be on for $\frac{8}{6}$ = 1.33 ms and off for 6.67 ms. Note that as the pulse width is decreased, the display brightness will also decrease. It may thus become necessary to increase the peak current through each segment by reducing the size of the resistors R in Figs. 15.1 and 15.2.

Example 15.2 The circuit in Fig. 15.3 show how to multiplex a common-anode-type display. Show how to multiplex a common-cathode-type display.

Solution The *npn* transistor in Fig. 15.5 is used as a switch between the cathode of the display and ground. When the transistor is on, current is allowed to pass through a segment for illumination. When the transistor is off, no current is allowed, and the segment cannot illuminate. The DIGIT waveform is shown in Fig. 15.5b. Notice that a positive pulse is required to turn the transistor on, and the display will be illuminated for 1 out of every 4 ms.



(a) A multiplexed 4-digit display



(b) Control waveforms

Fig. 15.4

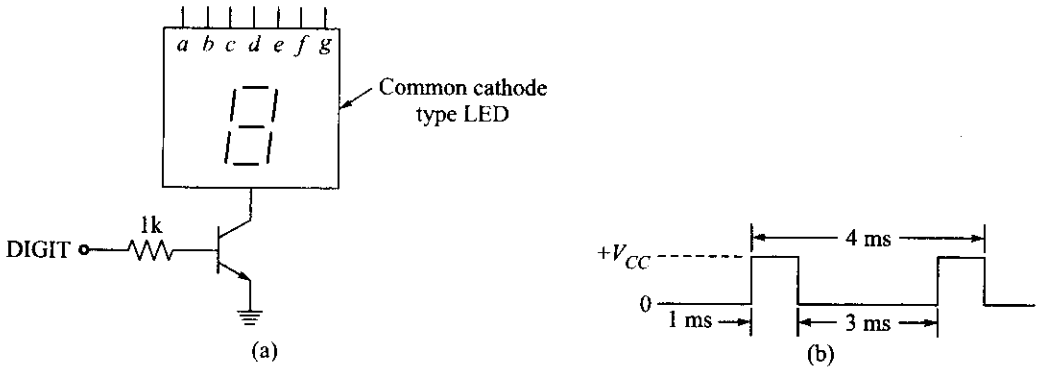


Fig. 15.5

The flip-flop outputs of a 7490 decade counter are used to drive the seven-segment decoder-driver in Figs. 15.1 and 15.2, and as long as the counter is counting, the displays will be changing states. It is often more desirable to periodically strobe the contents of the counter into a four-flip-flop latch and use these latches to drive the seven-segment decoder-driver. Then the four BCD inputs to the decoder-driver as well as the display will be steady all of the time except when new data is being strobed in. The circuit shown

in Fig. 15.6 is a complete single-digit display that will indicate the decimal equivalent of the binary number stored in the 7475 quad *D*-type latch by the positive STROBE pulse. It is also capable of being multiplexed by use of the DIGIT input.

Example 15.3 What are some possible methods for generating the DIGIT control waveforms shown in Fig. 15.4b?

Solution Reflecting back on topics covered in previous chapters, a number of different methods come to mind—for instance:

1. A two-flip-flop counter with four decoding gates
2. A four-flip-flop ring counter
3. A two-flip-flop shift counter with four decoding gates
4. A 1-of-4 low-output multiplexer

Can you think of any others?

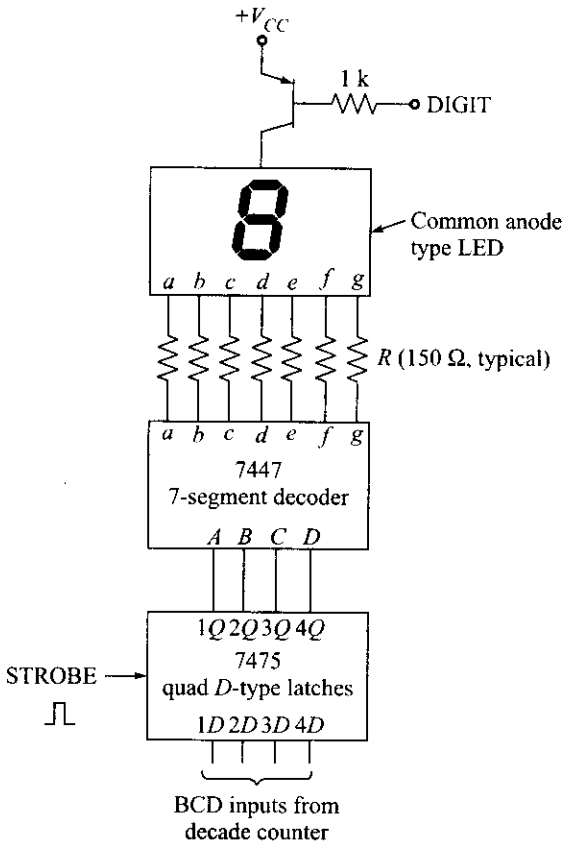


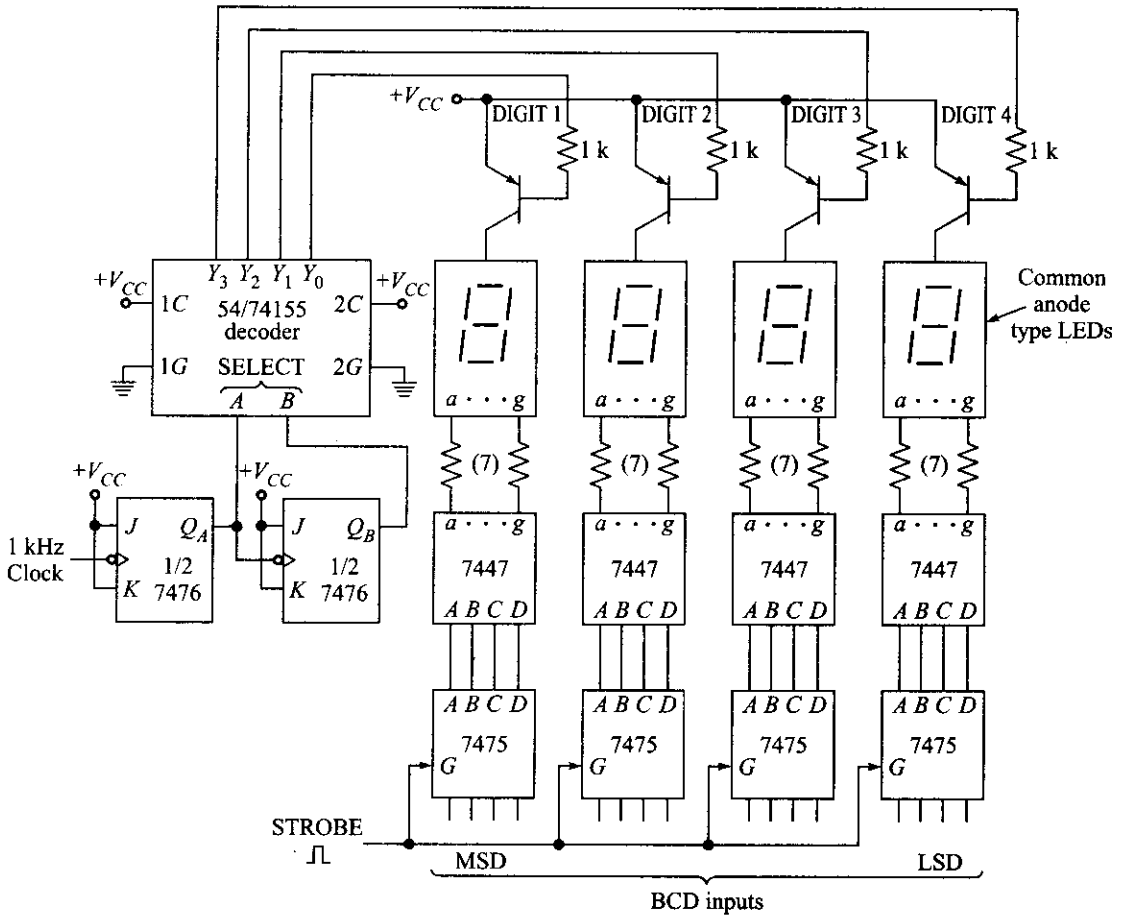
Fig. 15.6

The four-digit display in Fig. 15.7a on the next page uses four of the decimal digit displays in Fig. 15.6, and they are multiplexed to reduce power supply requirements. Notice that DIGIT 1 controls the LED on the left and this is the most-significant digit (MSD). The right display is controlled by DIGIT 4, and this is the least-significant digit counter (LSD). If we assume that the decimal point for this display to be at the right, the LSD is the units digit, and the MSD is the thousands digit. This circuit is capable of displaying decimal numbers from 0000 up to 9999. The 54/74155 is a dual 2-line to 4-line decoder-demultiplexer, and it is driven by a two-flip-flop binary counter called the *multiplexing counter*. As this multiplexing counter progresses through its four states, one and only one of the 54/74155 output lines will go low for each counter state. As a result, the DIGIT control waveforms exactly like those shown in Fig. 15.4b will be developed. You might like to review the operation of decoder-demultiplexers as discussed in Chapter 4.

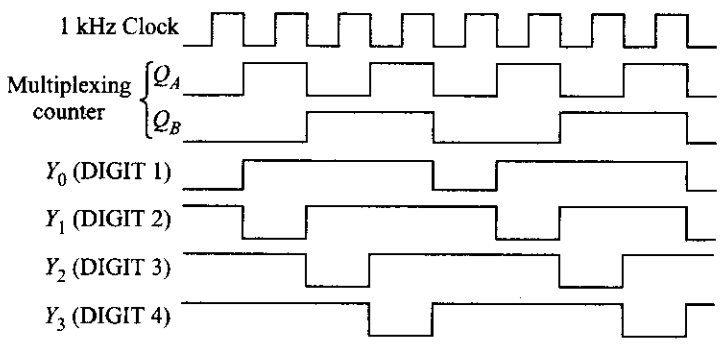
A savings in components as well as power can often be realized if the four inputs (*ABCD*) to the seven-segment decoder in Fig. 15.6 are multiplexed along with the DIGIT control. The four-digit display in Fig. 15.8 uses two 54/74153 dual 4- to 1-line multiplexers to apply the four outputs of each 7475 sequentially to a single seven-segment decoder. Here's how it works.

The BCD input data is stored in four 7475 *D*-type latches labeled 1, 2, 3, and 4. Latch 1 stores the MSD, and latch 4 stores the LSD. The 4-bit binary number representing the MSD is labeled $1_A1_B1_C1_D$. For instance, if the MSD = 7, then $1_A1_B1_C1_D = 0111$.

Each 74153 contains two multiplexers, and the four multiplexers are labeled *A*, *B*, *C*, and *D*. The *A* and *B* SELECT lines of the two multiplexers are connected in parallel and are driven by the multiplexing counter



(a) Four-decimal-digit multiplexed display



(b) Waveforms

Fig. 15.7

(exactly as in Fig. 15.7). When the SELECT inputs are $AB = 00$, the number 1 line of each multiplexer will be connected to its output. So, the multiplexer outputs connected to the 7447 decoder will be $1_A1_B1_C1_D$, which is the binary number for the MSD. This binary number is decoded by the 7447 and applied to all the LED displays in parallel. However, at this same time, DIGIT 1 is selected by the 74155 decoder, so the MSD will be displayed in the leftmost LED display. All the other displays will be turned off.

Now, when the multiplexing counter advances to count $AB = 01$, the number 2 line of each multiplexer will be selected, and the binary number applied to the 7447 will be $2_A2_B2_C2_D$, which is the next MSD (the hundreds digit). The decoded output of the 7447 is again applied to all the displays in parallel, but DIGIT 2 is the only LOW DIGIT line, so the "hundreds" digit is now displayed. (Again, all other displays are turned off during this time.)

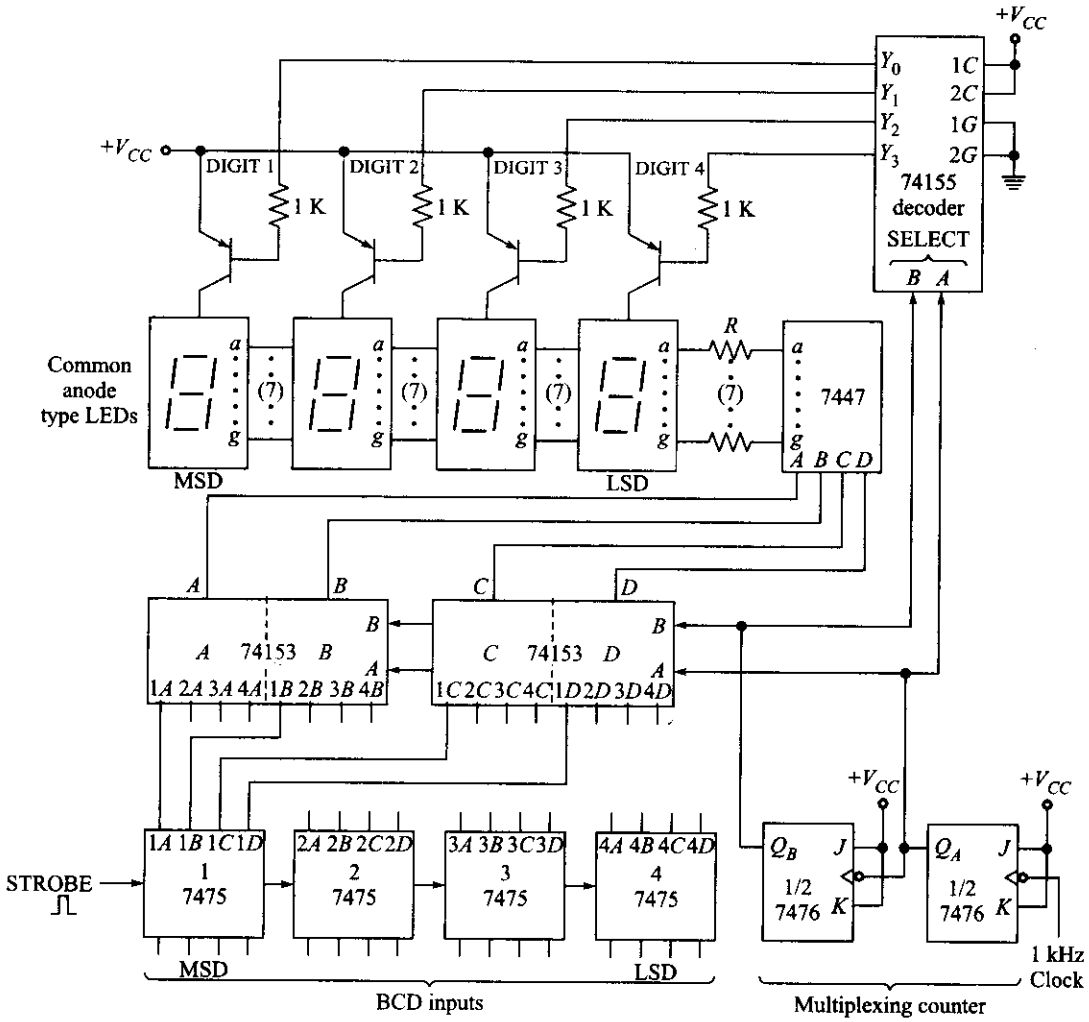
In a similar fashion, the tens digit will be displayed when the SELECT inputs are $AB = 10$, and the units digit will be displayed when the SELECT inputs are $AB = 11$. Notice that only one digit is displayed at a time, and the $RR = 250$ Hz, so no flicker will be apparent. Again, we are illuminating four digits but the power supply current is the same as for a single, continuously illuminated digit. At the same time, there is a modest saving of two chips. The savings in components increases as the number of decimal digits in the display increases.

The techniques used to multiplex the four-digit display in Fig. 15.8 on the next page are easily expanded to displays that have more than four decimal digits. It is necessary only to increase the size of the multiplexing counter and to replace the 74153 multiplexer with one that has a greater number of inputs. It is also a simple matter to alter the design to accommodate common-cathode-type LEDs instead of the common-anode types used here. (See the problems at the end of this chapter.)

All the display circuits discussed here are frequently constructed and used, but you should be aware that there are LSI chips available that have all the multiplexing accomplished on a single chip; examples of this are the National Semiconductor MM74C925, 926, 927, and 928. The MM74C925 shown in Fig. 15.9 is a four-digit counter with multiplexed seven-segment output drivers. The only external components needed are the seven-segment indicators and seven current-limiting resistors. In fact, a four-digit counter is even included on the chip! A positive pulse on the RESET input will reset the 4-bit counter, and then the counter will advance once with each negative transition of CLOCK. A negative pulse on LATCH ENABLE will then latch the contents of the counter into the four 4-bit latches. The four numbers stored are then multiplexed, decoded, and displayed on the four external seven-segment indicators. A simplified diagram is given in Fig. 15.9b. Notice that this is a common-cathode-type display.

15.2 FREQUENCY COUNTERS

A frequency counter is a digital instrument that can be used to measure the frequency of any periodic waveform. The fundamental concepts involved are illustrated in the block diagram in Fig. 15.10. The counter and display unit are exactly as described in Sec. 15.1. A GATE ENABLE signal that has a known period t is generated with a clock oscillator and a divider circuit and is applied to one leg of an AND gate. The unknown signal is applied to the other leg of the AND gate and acts as the clock for the counter. The counter will advance one count for each transition of the unknown signal, and at the end of the known time period, the contents of the counter will equal the number of periods of the unknown signal that have occurred during t. In other words, the counter contents will be proportional to the frequency of the unknown signal. For instance, suppose that the gate signal is exactly 1 s and the unknown input signal is a 750-Hz square wave. At the end of 1 s, the counter will have counted up to 750, which is exactly the frequency of the input signal.

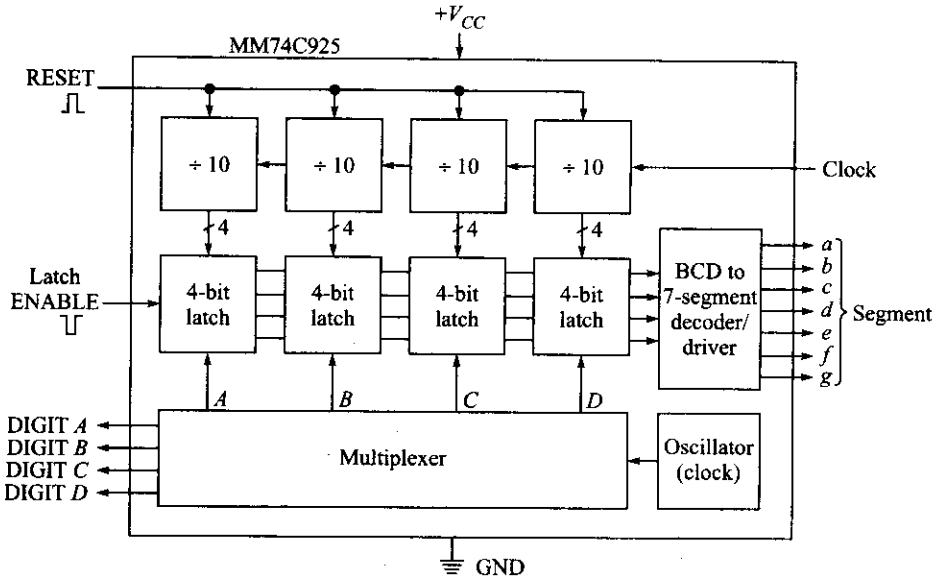


Note: For clarity, only the connections for the *MSD* are shown (1*A*, 1*B*, 1*C*, 1*D*), but the other 12 connections must be made between the input latches and the 74153s.

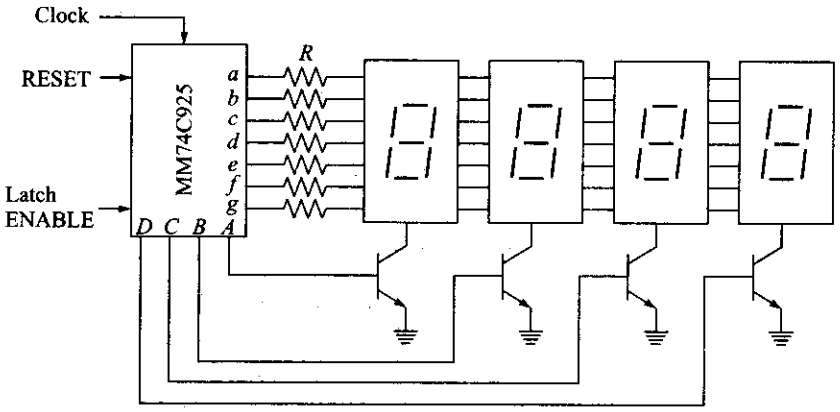
Fig. 15.8

Example 15.4 Suppose that the unknown input signal in Fig. 15.10 is a 7.50-kHz square wave. What will the display indicate if the GATE ENABLE time is $t = 0.1$ s? What if $t = 1$, and then 10 s?

Solution When $t = 0.1$ s, the counter will count up to 7500 (transitions per second) × 0.1 (second) = 750. When $t = 1$ s, the counter will display 7500 (transitions per second) × 1 (second) = 7500. When $t = 10$ s, the counter will display 7500 (transitions per second) × 10 (seconds) = 75,000. For this last case, we would have to have a five-decimal-digit display.



(a)



(b)

Fig. 15.9

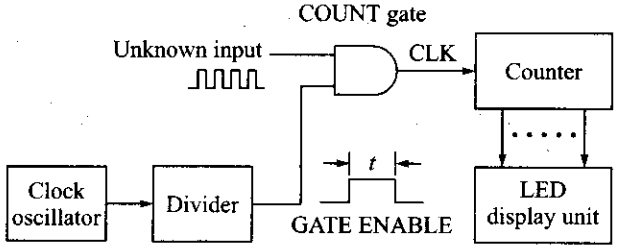


Fig. 15.10 **Basic frequency counter**

In Example 15.4, the contents of the counter are always a number that is proportional to the unknown input frequency. In this case, the proportionality constant is either 10, 1, or $\frac{1}{10}$. So, it is a simple matter to insert a decimal point between the indicators such that the unknown frequency is displayed directly. Figure 15.11 shows how the decimal point moves in a five-decimal-digit display as the gate width is changed. In the top display, the unknown frequency is the display contents multiplied by 10, so the decimal point is moved one place to the right. The middle display provides the actual unknown frequency directly. In the bottom display, the contents must be divided by 10 to obtain the unknown frequency, so the decimal point is moved one place to the left.

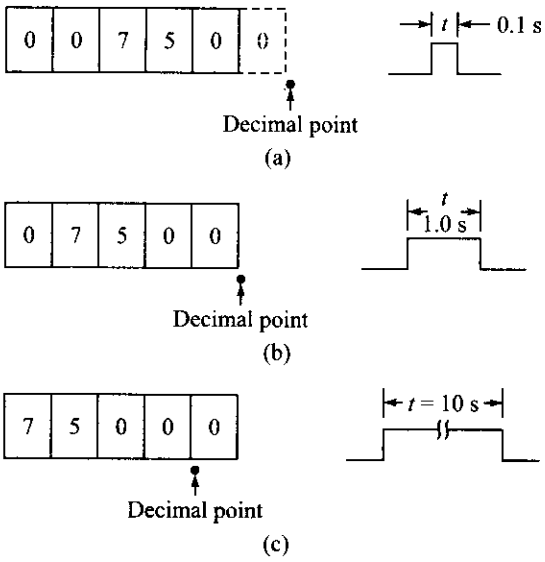


Fig. 15.11 Decimal point movement for Example 15.4

The logic diagram in Fig. 15.12 on the next page shows one way to construct a four-decimal-digit frequency counter. The AMPLIFIER block is intended to condition the unknown input signal such that INPUT is a TTL-compatible signal—a series of positive pulses going from 0 to +5 V dc. When allowed to pass through the COUNT gate, INPUT will act as the clock for the COUNTER. The COUNTER can be constructed from four decade counters such as 54/74160s, and it can then be connected to a multiplexed LED DISPLAY such as the one shown in Fig. 15.8. Or, COUNTER and DISPLAY can be combined in a single chip such as the MM74C925 shown in Fig. 15.9.

The DIVIDER is composed of six decade counters (such as 54/74160s) connected in series. Its input is a 100-kHz square wave from OSC CLOCK, and it provides 10-, 1-, and 0.1-Hz square wave outputs that are used to generate the ENABLE-gate signal.

When the 1-Hz square wave is used to drive the GATE flip-flop, its output, Q, is a 0.5-Hz square wave. Output Q will be high for exactly 1s and low for 1s, and it will thus be used for the ENABLE-gate signal. Notice that the 10-Hz signal will generate a 0.1-s gate and the 0.1-Hz signal will generate a 10-s gate. Let's use the waveforms in Fig. 15.12 to see exactly how the circuit functions.

A measurement period begins when the GATE flip-flop is toggled high—labeled *START* on the time line. INPUT now passes through the COUNT gate and advances the COUNTER. (Let's assume that the counter

is initially at 0000.) At the end of the ENABLE-gate time t, the GATE flip-flop toggles low, the COUNTER ceases to advance, and this negative transition of Q triggers the 74121 one-shot. Simultaneously, $\overline{Q}$ goes high, and this will strobe the contents of COUNTER into the DISPLAY latches. There is a propagation delay time of 30 ns minimum through the 74121, and then a negative RESET pulse appears at its output, $\overline{X}$. This propagation delay assures that the contents of COUNTER are strobed into DISPLAY before COUNTER is reset. The RESET pulse from the '121 has an arbitrary width of 1 μs, as set by its R and C timing components. The end of the RESET pulse is the end of one measurement period, labeled END on the time line.

For a 1.0-s gate, the decimal point will be at the right of the units digit, and the counter will be capable of counting up to 9999 full scale, with an accuracy of plus or minus one count (i.e. 1 part in 10^4). With a 10-s gate, the decimal point is between the units and the tens digits, and with a 0.1-s gate, the decimal point is one place to the right of the units digit.

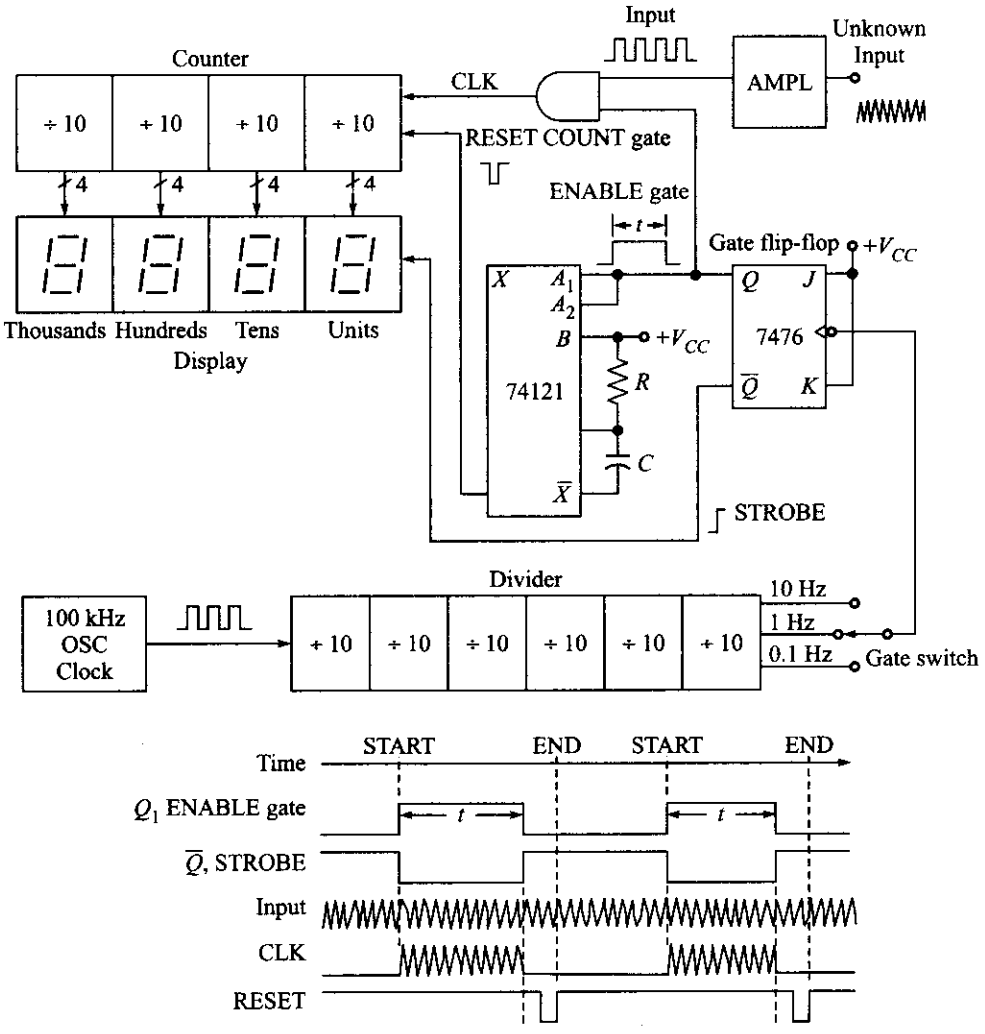


Fig. 15.12 **Four-decimal-digit frequency counter**

The CLOCK oscillator is set at 100 kHz, and this provides an accuracy on the ENABLE-gate time of 1 part in 10^4 with the 0.1-s gate. Thus the accuracy here is compatible with that of the COUNTER.

Example 15.5 Explain what would happen if the instrument in Fig. 15.12 were set on a 1-s gate time and the input signal were 12 kHz.

Solution Assuming that the counter began at 0000, the display would read 200 at the end of the first measurement period. It would then read 400, then 600, and so on at the end of succeeding periods. This is because the counter capacity is exceeded each time, and it simply recycles through 0000.

The design in Fig. 15.12 shows one method for constructing a frequency counter using readily available TTL chips, but you should be aware that there are numerous chips available that have all, or nearly all, of this design on a single chip, for instance, the Intersil ICM7226A. You will be asked to do a complete design of a frequency counter based on Fig. 15.12 in one of the problems at the end of this chapter.

15.3 TIME MEASUREMENT

With only slight modifications, the frequency counter in Fig. 15.10 can be converted into an instrument for measuring time. The logic block diagram in Fig. 15.13 illustrates the fundamental ideas used to construct an instrument that can be used to measure the period of any periodic waveform. The unknown voltage is passed through a conditioning amplifier to produce a periodic waveform that is compatible with TTL circuits and is then applied to a *JK* flip-flop. The output of this flip-flop is used as the ENABLE-gate signal, since it is high for a time t that is exactly equal to the time period of the unknown input voltage. The oscillator and divider provide a series of pulses that are passed through the count gate and serve as the clock for the counter. The contents of the counter and display unit will then be proportional to the time period of the unknown input signal.

For instance, if the unknown input signal is a 5-kHz sine wave and the clock pulses from the divider are 0.1 μs in width and are spaced every 1.0 μs, the counter and display will read 200. Clearly this means 200 μs, since 200 of these 0.1-μs pulses will pass through the COUNT gate during the 200 μs that ENABLE-gate signal is high. Naturally the counter and the display have an accuracy of plus or minus one count.

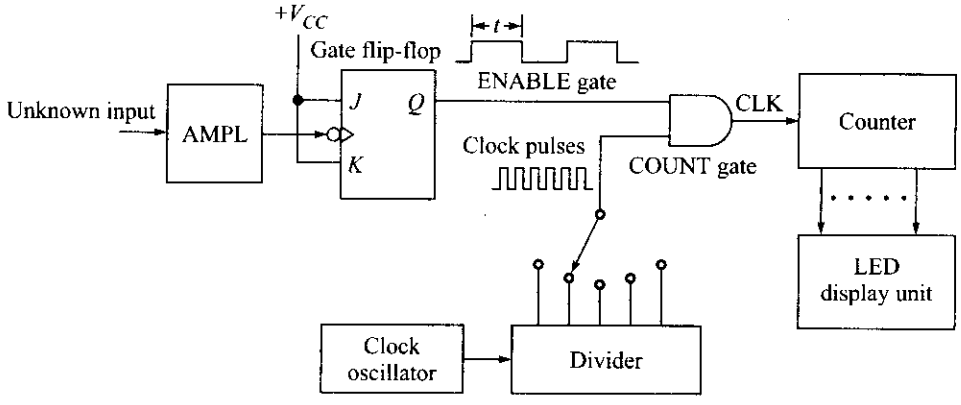


Fig. 15.13 **Instrument to measure time period**

Example 15.6 Suppose that the counter and the display unit in Fig. 15.13 have five-decimal-digit capacity and the divider switch is set to provide a 100-kHz square wave that will be used as clock pulses. What will the display read after one ENABLE-gate time t, if the unknown input is a 200-Hz square wave?

Solution Assume that the counter and the display are initially at 00000. A 200-Hz input signal will produce an ENABLE-gate time of $t = \frac{1}{200} = 5000\ \mu s$. The 100-kHz square wave used as the clock is essentially a series of positive pulses spaced by 10 μs. Therefore, during the gate time t, the counter will advance by, $\frac{5000}{10} = 500$ counts, and this is what will be viewed in the displays. Since each clock pulse represents 10 μs, the display should be read as $500 \times 10 = 5000\ \mu s$—this is the time period of the unknown input.

Example 15.7 Explain the meaning of an accuracy of plus or minus one count applied to the measurement in Example 15.6.

Solution An accuracy of plus or minus one count means that the display could read 499, 500, or 501 after the measurement period. This means that the period as measured could be 4990, 5000, or 5010 μs—in other words, 5000 plus or minus 10 μs. Since a single count represents a clock period of 10 μs, this instrument can be used for measurement only within this limit. For more precise measurement, say, to within 1 μs, the clock pulses would have to be changed from 10- to 1-μs spacing.

The circuit in Fig. 15.14 is a four-decimal-digit instrument for measuring the time period of a periodic waveform. It is essentially the same as the frequency instrument in Fig. 15.12 with only slight modifications. First, the CLOCK has been increased to 1 MHz, and DIVIDER is composed of a buffer amplifier and three decade counters. This will provide clock pulses for COUNTER with 1-, 10-, and 100-μs as well as 1-ms spacing. The unknown input is conditioned by AMPLIFIER and is then applied to the GATE flip-flop to generate the ENABLE-gate signal. STROBE and RESET are generated and applied as before. Notice that a single instrument for measuring both frequency and period could be easily designed by using a 1-MHz clock with a divider that has seven decade counters and some simple mechanical switches.

Example 15.8 Explain the DISPLAY ranges for the four-decimal-digit period measurement instrument in Fig. 15.14.

Solution With CLOCK pulses switched to the 1-μs position, each count of COUNTER represents 1 μs. Therefore, it has a full scale of $9999 \pm 1\ \mu s$. On 10 μs, it has a full scale of $9999 \times 10 = 99{,}990 \pm 10\ \mu s$. Full scale on the 0.1-ms position is $9999 \times 0.1 = 999.9 \pm 0.1$ ms. Full scale on the 1-ms position is 9999 ± 1 ms.

An interesting variation on the instrument in Fig. 15.14 is to use it to measure the time elapsed between two events. There will be two input signals, the first of which sets the ENABLE gate high and begins the count period. The second signal (or event) resets the ENABLE gate low and completes the time period. One method for handling this problem is to use the first event to set a flip-flop and then use the second event to reset it. Of course, both input signals must be first conditioned such that they are TTL-compatible. You are given the opportunity to design such an instrument in one of the problems at the end of the chapter.

15.4 USING THE ADC0804

Stand-Alone Operation

The ADC0804 was briefly introduced in Sec. 12.8. Figure 12.29 shows how to connect the ADC0804 for stand-alone operation, and it is repeated here for convenience. The recommended supply voltage is $V_{CC} = +5$

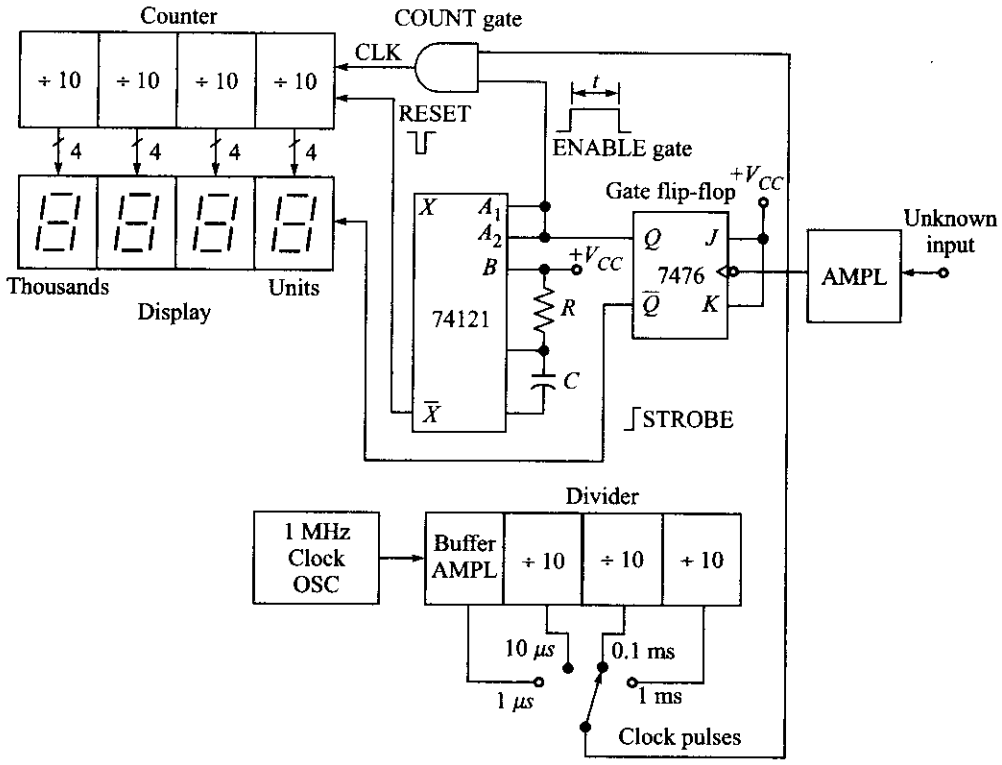


Fig. 15.14 **Four-decimal-digit period measurement instrument**

Vdc. The external resistor R and capacitor C establish the frequency of the internal clock according to

$$f \cong 1/(1.1\,RC) \tag{15.1}$$

Pin 9 is an input for an external reference voltage V_{ref}. If pin 9 is left open, the reference voltage is set internally at $V_{CC}/2$. The analog input voltage is applied between pins 6 and 7. With pin 7 connected to ground, the allowable input voltage range is from 0.0 to +5.0 V.

This ADC is designed for use with the 8080A CPU (central processing unit) chip set, composed of the 8080A microprocessor, the 8228 system controller, and the 8224 clock. It can also be used directly with the 8048 MPU (microprocessor unit). The inputs $\overline{WR}$, $\overline{INTR}$, $\overline{CS}$ and $\overline{RD}$ are microprocessor control signals. In the stand-alone mode of operation, $\overline{WR}$ and $\overline{INTR}$ are connected directly to ground. $\overline{CS}$ and $\overline{RD}$ are momentarily grounded with a push-button switch to initiate a conversion. The converter will digitize the analog voltage present at the input at the instant the push button is depressed. It will then continue to convert additional analog input voltage levels at approximately 100-μs intervals.

The digitized value of an input voltage is presented as an 8-bit binary number on pins 11 through 18, with pin 11 the most significant bit (MSB). An input voltage of 0.0 V has a digitized output of 0000 0000 (OOH). The digital output 1111 1111 (FFH) represents a full-scale input of +5.0 V. The digitized output is accurate to ±1 LSB. Since the full-scale input of 5.0 V is represented by $2^8 = 256$ bits, 1 bit (the LSB) is equivalent to an analog voltage of 5.0 V/256 = 19.53 mV.

Example 15.9 Refer to the ADC0804 in Fig. 12.29, repeated below for your reference.

(a) What is the digital output for an analog input of 2.5 V?
(b) The digital output is 0010 0010 (22H). What is the analog input?

Solution

(a) 2.5 V is one-half full scale. The digital output is then 1000 0000 ±1 bit ($2^7 = 128$). As a check, 128 × 19.53 mV = 2.5 V

(b) $(2^5 + 2^1) \times 19.53$ mV = $(32 + 2) \times 19.53 = 0.664$ V.

Span Adjust

As shown in Fig. 12.29, the ADC0804 functions nicely for analog input voltages between 0.0 and +5.0 V. But what if the input voltage range is only from 0.0 to 2.0 V? In this case, we would like the full-scale input to be 2.0 V rather than 5.0 V. Fortunately, this is quite easy to do with the ADC0804! Simply connect an external reference voltage V_{ref} to pin 9 that is *one-half* the desired full-scale input voltage. Another term for the full-scale input voltage range is *span*. In this case, set $V_{ref} = 2.0\text{ V}/2 = 1.0$ Vdc. In general terms,

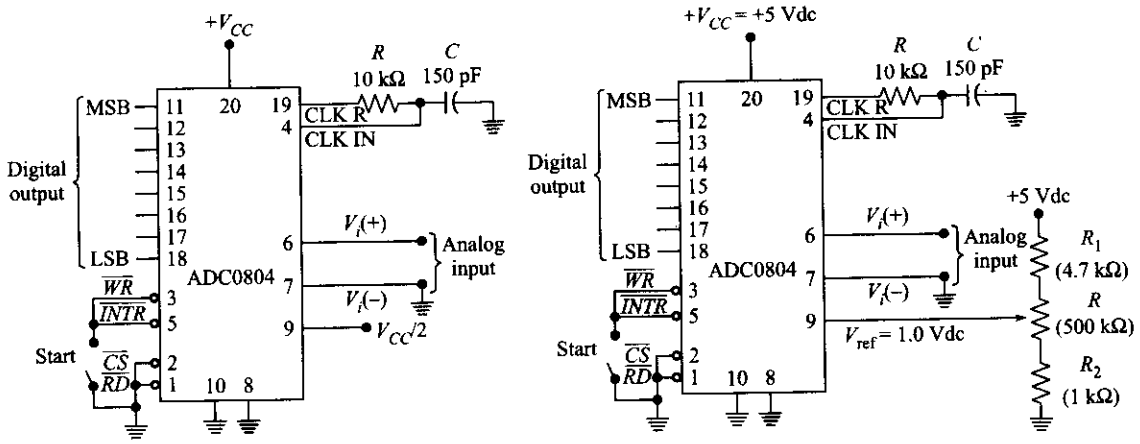
$$V_{ref} = \text{full-scale analog input voltage}/2 = \text{span}/2 \tag{15.2}$$

In Fig. 15.15, a simple resistive divider is used to generate the reference voltage V_{ref}. Here's an expression to use with this divider:

$$V_{ref} = V_{CC}\frac{R_2 + R/2}{R_1 + R_2 + R} \tag{15.3}$$

As an example, let's apply Eq. (15.3) using the circuit values given in Fig. 15.15.

$$V_{ref} = +5\text{ Vdc}\,\frac{1\text{ k}\Omega + 0.25\text{ k}\Omega}{1\text{ k}\Omega + 4.7\text{ k}\Omega + 0.5\text{ k}\Omega} = 1.01\text{ Vdc}$$



Stand-alone operation (Fig. 12.29 repeated)

Fig. 15.15

In this case, $V_{ref} = 1.0$ Vdc. The 500-Ω potentiometer will allow fine adjustment. So the full-scale analog input voltage is then 0.0 to 2.0 V. An input voltage of 2.0 V will convert to a digital output 1111 1111 (FFH). The LSB is equivalent to 2.0 V/256 ≡ 7.8 mV.

Zero Shift

The ADC0804 can also accommodate analog input voltages that are offset from zero. For instance, suppose we wish to digitize an analog signal that is always between the limits +1.5 V and +4.0 V, as illustrated in Fig. 15.16a. The span of this signal is (4.0 – 1.5) V = 2.5 V. So we would use Eq. (15.2) to find

$$V_{ref} = \frac{\text{span}}{2} = \frac{2.5 \text{ volts}}{2} = 1.25 \text{ Vdc}$$

This reference voltage (1.25 Vdc) is applied to pin 9.

Now we connect pin 7 to the *lower limit* of the input voltage. This lower limit is called the *OFFSET*. In general terms,

$$\text{OFFSET at } V_{in} = \text{analog input lower limit} \tag{15.4}$$

In Fig. 15.16b, we have used two voltage dividers, one for $V_{ref} = 1.25$ Vdc and one for $V_{i-} = 1.5$ Vdc. For this circuit an analog input voltage of 1.5 V will be digitized as 0000 0000. An input of 4.0 V will convert to 1111 1111. This LSB is then equivalent to 2.5 V/256 ≅ 9.77 mV.

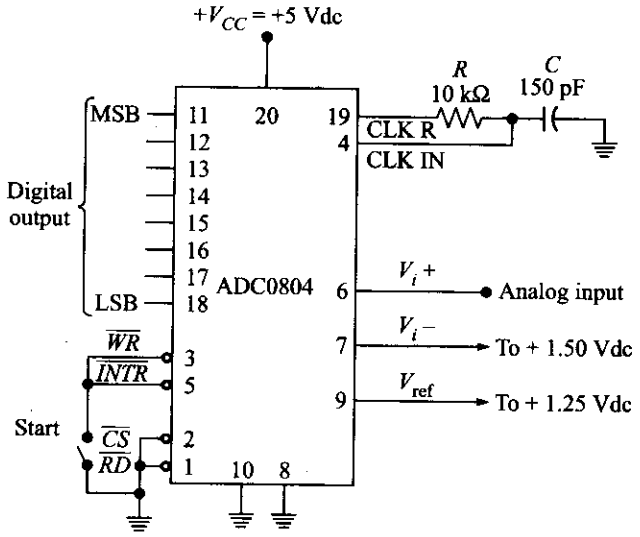


Fig. 15.16

Positive and Negative Input Voltages

Up to now, we have only considered positive analog voltage levels. How can we handle both positive and negative input signals? For instance, suppose we wish to digitize analog voltages that vary from –5 to +5 Vdc. One solution is given in Fig. 15.17. The technique is to use a resistive voltage divider (R and R) at the input pin 6. Pin 7 is connected to ground.

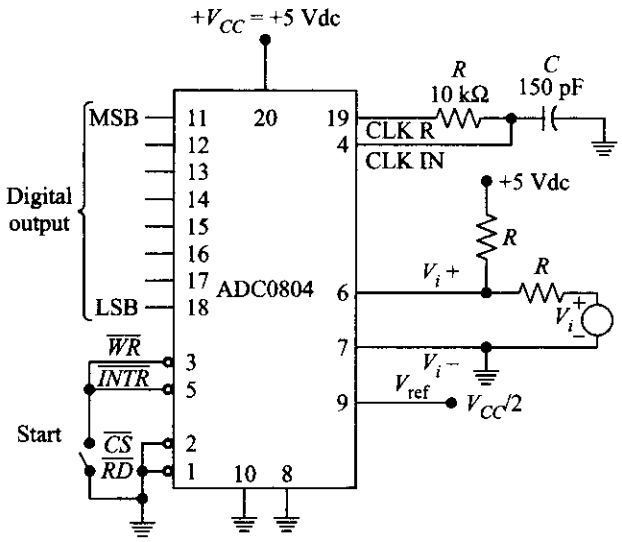


Fig. 15.17

A little thought will reveal the following:

1. V_i = −5 V. Then V_{i+} = 0.0 V. The digitized output is 0000 0000 (00H).
2. V_i = 0.0 V. Then V_{i+} = +2.5 V. The digitized output is 1000 0000 (80H). This is mid scale.
3. V_i = +5 V. Then V_{i+} = +5.0 V. The digitized output is 1111 1111 (FFH). This is full scale.

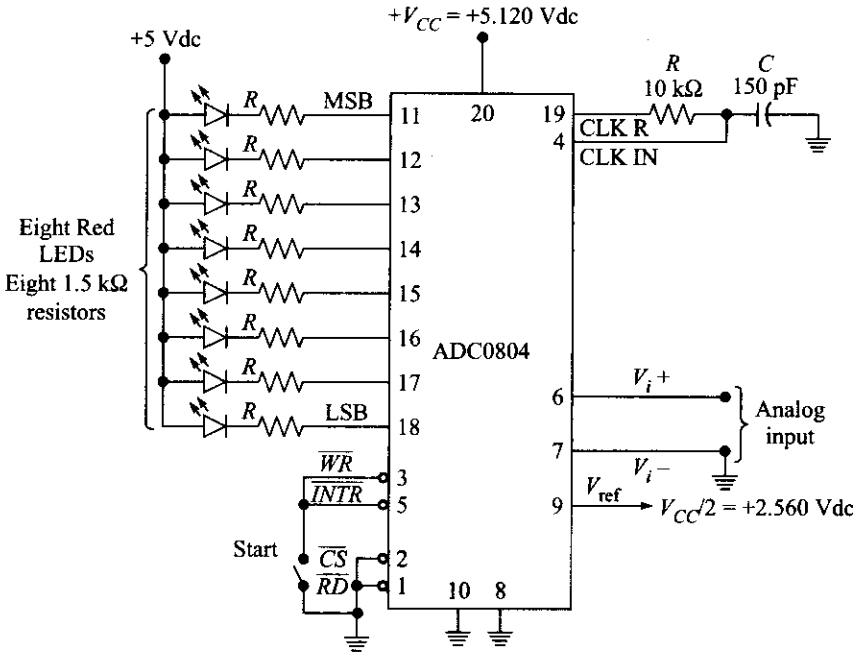
The span at V_{i+} is clearly 5.0 V. OFFSET is not required, since the voltage at V_{i+} varies between 0.0 V and +5.0 V. Notice that a *negative* input voltage, V_i, always produces a 0 for the MSB of the digital output (with the possible exception of 0.0). A *positive* input voltage, V_i, always produces a 1. for the MSB of the digital output (again, with the possible exception of 0.0). In this case, the LSB is equivalent to 10 V/256 = 39.01 mV.

Testing

When using an ADC0804, it may become necessary to test it for proper operation, for example, before initial installation or perhaps to troubleshoot a suspected malfunction. There are many different testing procedures for A/D converters, some of which are quite complex and computer-controlled. However, a rapid and simple test is to apply a known analog input voltage while monitoring the digital outputs. The test circuit in Fig. 15.18 on the next page can be used for this purpose. Notice that the dc supply voltage has been adjusted carefully to a value V_{CC} = 5.120 Vdc. Also, V_{ref} has been set at $V_{CC}/2$ = 2.560 Vdc. These values have been chosen so that the LSB has a weight of 5.120 V/256 = 20 mV. This eliminates any round off error and makes the arithmetic easier!

A checkerboard-type test is used to activate each output. Here's how to do it:

1. Apply an input voltage to produce the digital output 1010 1010 (AAH). The required input is (128 + 32 + 8 + 2) 20 mV = 3.400 V.
2. Apply an input voltage to produce the digital output 0101 0101 (55H). The required input is (64 + 16 + 4 + 1) 20 mV = 1.700 V.



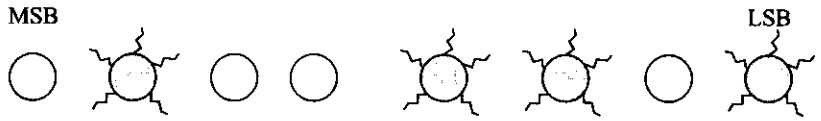
Note: Illuminated LED ≡ output = low = 0
Extinguished LED ≡ output = high = 1

Fig. 15.18

These two tests will activate all eight outputs in both states. This is not a comprehensive test, but it will detect any faults in the outputs, and it will thus give a reasonable degree of confidence in the operation of the A/D converter. Note carefully that a digital output 1 (high) will extinguish the LED. A low ouput (a 0) will illuminate an LED. So,

Illuminated LED ≡ low ≡ 0
Extinguished LED ≡ high ≡ 1

For example, the output 1011 0010 is "seen" as



SELF-TEST

1. Why are the external *R* and *C* in Fig. 12.29 needed?
2. What is the purpose of the START button in Fig. 12.29?
3. The digital output or an ADC0804 is 1100 0011. What is this in hexadecimal?
4. For the ADC0804 what do the terms *span* and *OFFSET* mean?

15.5 MICROPROCESSOR-COMPATIBLE A/D CONVERTERS

A fundamental requirement in many digital data acquisition systems is an A/D converter that is simple, reliable, accurate, inexpensive, and readily usable with a minicomputer or microprocessor. The National Semiconductor ADC3511 is a single-chip A/D converter constructed with CMOS technology that has $3\frac{1}{2}$-digit BCD outputs designed specifically for use with a microprocessor, and it is available for less than $9! The 3511 uses an integrating-type conversion technique and is considerably slower than flash-type or SAR-type A/D converters. It is quite useful in digitizing quantities such as temperature, pressure, or displacement, where fewer than five conversions per second are adequate. The pinout and logic block diagram for an ADC3511 are shown in Fig. 15.19.

Only a single +5-Vdc power supply is required, and the 3511 is completely TTL-compatible. This A/D converter is a very high precision analog device, and great care must be taken to ensure good grounding, power supply regulation, and decoupling. It is important that a single GROUND point be eatablished at pin 13, to eliminate any ground loop currents. Voltage V_{CC} on pin 1 is used to apply +5-Vdc power. A 10-μF 10-Vdc capacitor is connected between pin 2 and GROUND; this capacitor, and the internal 100-Ω resistor shown on the logic block diagram are used to decouple the dc power used for the analog and digital circuits. Voltage V_{SS} on pin 22 should also be connected directly to GROUND.

The conversion rate of the chip is established by a resistor R connected between pins 17 and 18 and a capacitor C connected between pin 17 and GROUND. The clock frequency developed by these two components is given by $f = 0.6/RC$ and should be set between 100 and 640 kHz. This is the clock signal used to advance the internal counters thal finally store the digitized value of the analog input voltage.

The analog signal to be digitized is applied between $+V_i$ and $-V_i$, pins 11 and 10, respectively. Negative signals are handled automatically by the converter through the switching network at the input to the comparator. A conversion is initiated with a low-to-high transition of START CONVERSION on pin 7. The waveform, CONVERSION COMPLETE, on pin 6 will go low at the beginning of a conversion cycle and then return high at the end of a conversion cycle. Connecting pin 7 to $+V_{CC}$ will cause the chip to continuously convert the analog input signal. The using edge of the waveform on pin 6 indicates that new digital information has been transferred to the digit latches and is available for output.

The digitized analog signal is contained in the converter as four BCD digits. The LSD, or units digit, is $D_1C_1B_1A_1$, the tens digit is $D_2C_2B_2A_2$, the hundreds digit is $D_3C_3B_3A_3$, and the MSD is $C_4B_4A_4$. All digits can store the BCD equivalent of decimal 0, 1, 2, ... , 9, except the MSD. The MSD can have values of only decimal 0 or decimal 1. It is for this reason that the 3511 is called a $3\frac{1}{2}$ digit device—the MSD is referred to as a *half-digit*. The 16×4 MUX is used to multiplex one digit (4 bits) at a time to the outputs according to the input signals D_0 and D_1 as given in Fig. 15.20. For instance, when $D_1D_0 = 00$, the LSD appears on the four output lines 23, 22, 21, and 20. A low-to-high transition on DIGIT LATCH ENABLE (DLE), pin 19, will latch the inputs D_1D_0, and the selected digit will remain on the four output pins until DLE returns low. The polarity of the digitized input analog signal will also appear on pin 8, SIGN. The 3511 has a full-scale count of 1999, and if this count is exceeded, an overflow condition occurs and the four digit outputs will indicate *EEEE*.

The heart of the analog-to-digital conversion consists of the comparator, the *D*-type flip-flop, and an *RC* network that is periodically switched between a reference voltage V_{ref} and ground. When the output of the *D*-type flip-flop, Q, is high, the transistor designated as SW_1 is on, and the other transistor designated as SW_2 is off. Under this condition, the capacitor C charges through R toward the reference voltage (usually +2.00

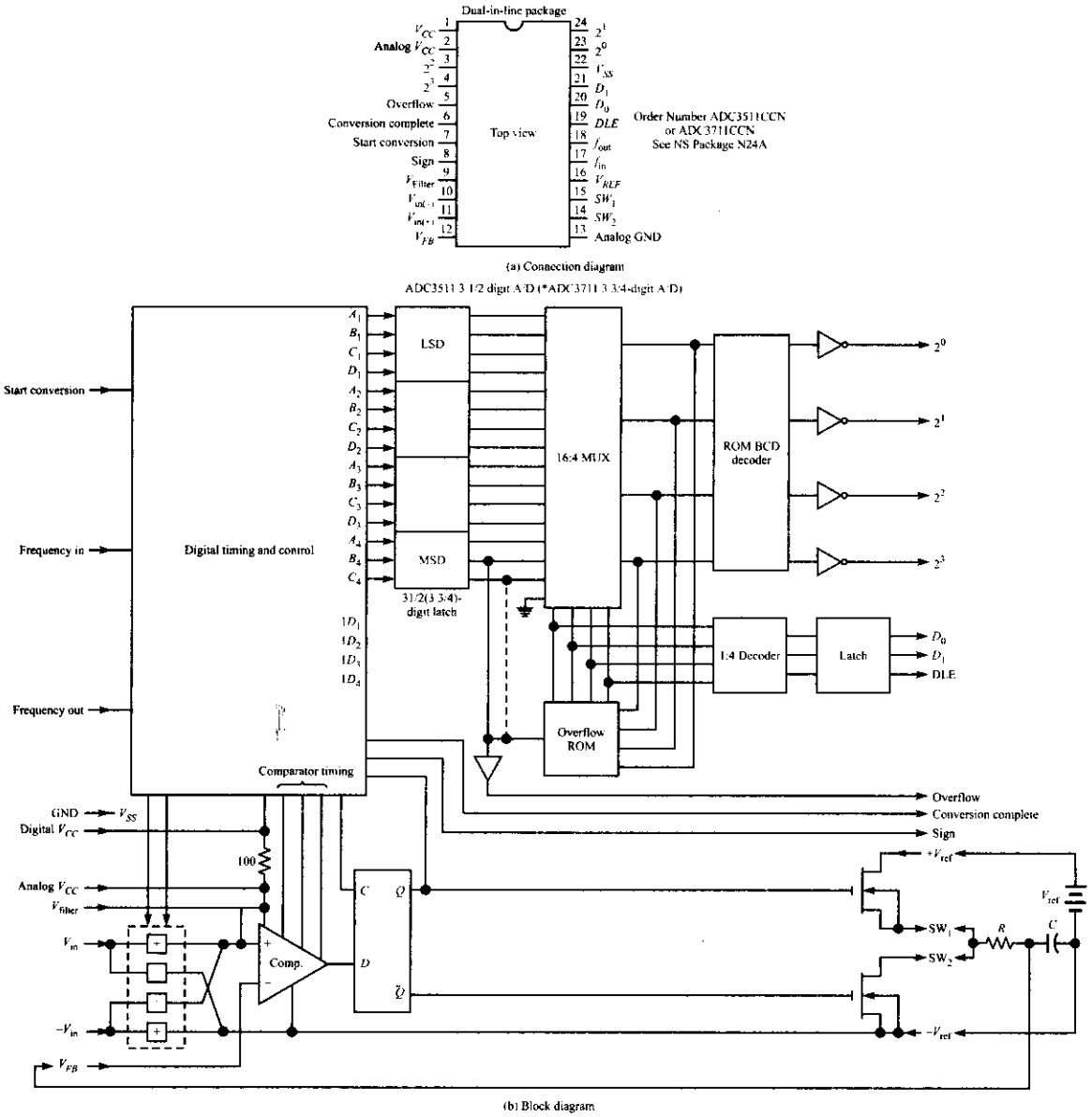


Fig. 15.19 National Semiconductor ADC3511 (3711)

Vdc), and the capacitor voltage V_{fb}, is fed back to the negative input terminal of the comparator. When V_{fb} exceeds the analog input voltage, the comparator output switches low, and the next clock pulse will set Q low. When Q is low, SW_1 is off and SW_2 is on. The capacitor now discharges through resistor R toward 0.0 Vdc, As soon as V_{fb} discharges below the analog input voltage, the comparator output will switch back to a high state, and this process will repeat.

These components form a closed-loop system that will oscillate—that is, a rectangular waveform as shown in Fig. 15.21 will be produced at SW_1 and SW_2 (pins 15 and 14).

The duty cycle of this waveform is given as

$$\text{Duty cycle} = \frac{t_c}{t_c + t_d}$$

and its dc value is given as

$$V_{dc} = V_{ref} \times \text{duty cycle}$$

This dc voltage will appear at V_{fb} and the closed-loop system will adjust itself such that

$$V_i = V_{dc} = V_{ref} \times \text{duty cycle}$$

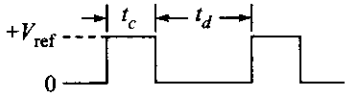
or

$$\frac{V_i}{V_{ref}} = \text{duty cycle} = \frac{t_c}{t_c + t_d}$$

DIGIT SELECT Inputs			
DLE	D_1	D_0	Selected DIGIT
L	*L*	*L*	DIGIT 0 (LSD)
L	*L*	*H*	DIGIT 1
L	*H*	*L*	DIGIT 2
L	*H*	*H*	DIGIT 3 (MSD)
H	*X*	*X*	No change

L = Low logic level
H = High logic level
X = Irrelevent-logic level
The value of the selected digit is presented at the 2^3, 2^2, 2^1, and 2^0 outputs in BCD format.

Fig. 15.20 ADC 3511 (3711) control levels



$$\text{Duty cycle} = \frac{t_c}{t_c + t_d}$$

Fig. 15.21 Waveforms at SW_1 and SW_2 (pins 15 and 14 respectively) for the ADC3511

The maximum allowable value for the analog input voltage is V_{ref}. When the input is equal to V_{ref}, the duty cycle must be equal to 1.0 ($t_d = 0$) and Q is always high. If the input analog signal is 0.0. the duty cycle must be zero ($t_c = 0$), and Q is always low. For an analog input voltage between 0.0 and $+V_{ref}$, the duty cycle is some value between 0.0 and 1.0.

The waveform Q at the output of the D-type flip-flop has exactly the same duty cycle as V_{fb}, and it is used to gate a counter in the converter. The counter can only advance when Q is high, and the gating is arranged such that for a duty cycle of 1.0, the counter will count full scale (1999), and for a duty cycle of 0, the counter will count 0000. For any duty cycle between 0.0 and 1.0, the counter will count a proportional amount between 0000 and 1999. In fact, the exact COUNT relationship is given as

$$\text{COUNT} = N \times \frac{V_i}{V_{ref}}$$

where N is the full-scale count of 2000.

Example 15.10 An ADC3511 is connected with a reference voltage of +2.0 Vdc. What will be the count held in the counter for an analog input voltage of 1.25 Vdc? What must be the duty cycle?

Solution Using the expression given above, we obtain

$$\text{Count} = 2000 \times \frac{1.25}{2.00} = 1250$$

The duty cycle must be

$$\text{Duty cycle} = \frac{V_i}{V_{ref}} = \frac{1.25}{2.00} = 0.625$$

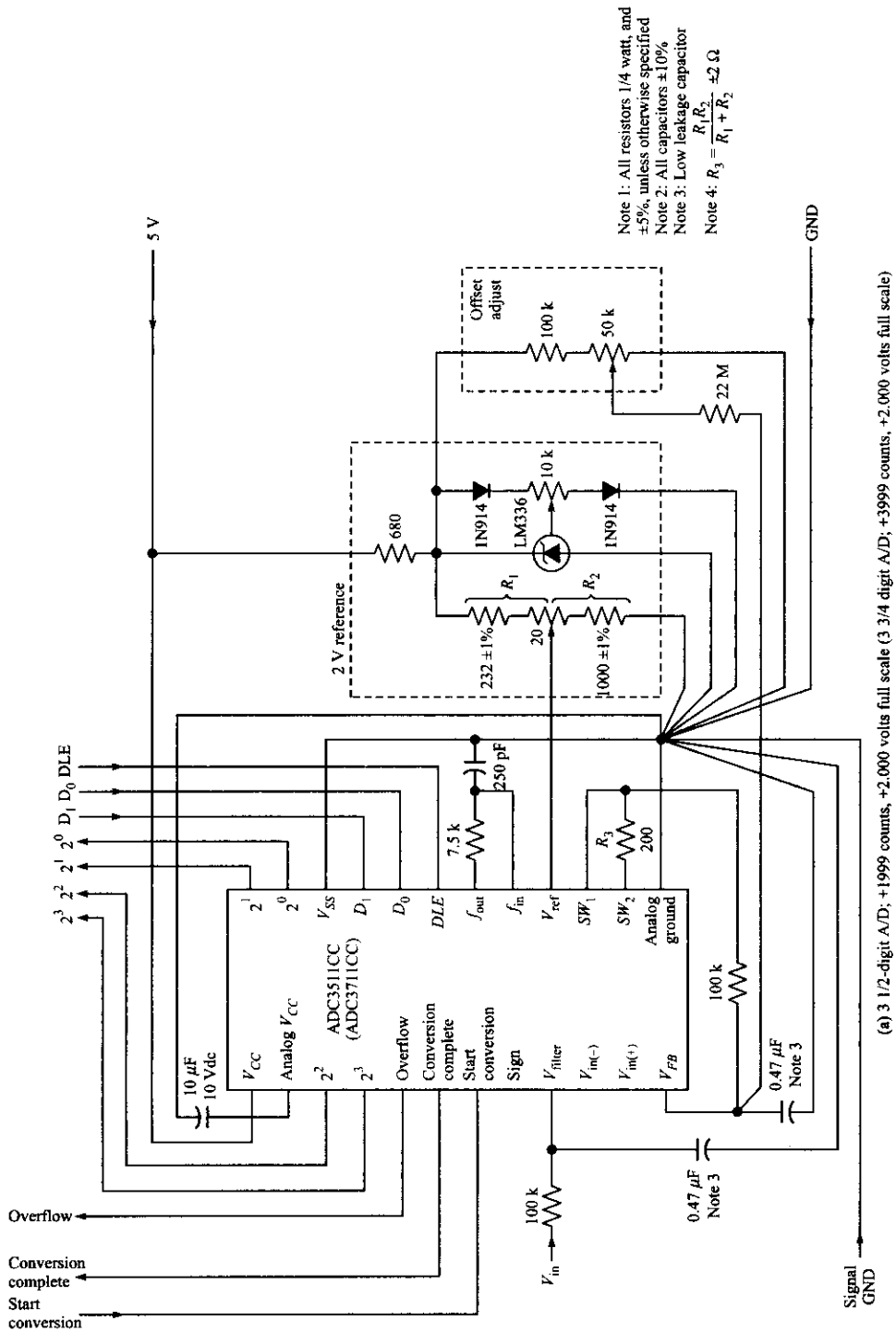
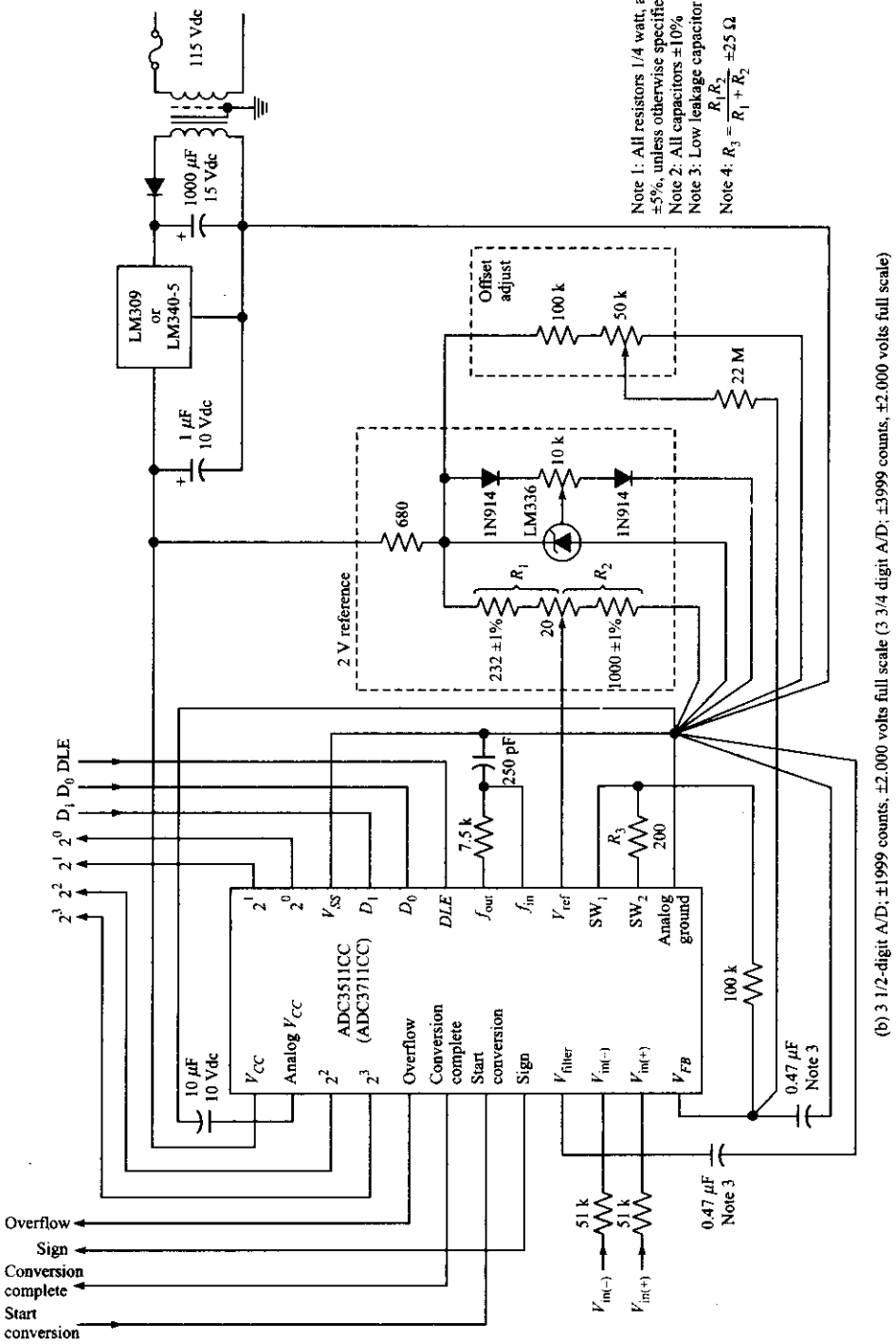


Fig. 15.22 (From National Semiconductor Data Acquisition Handbook; continued on next page)

(a) 3 1/2-digit A/D; +1999 counts, +2.000 volts full scale (3 3/4 digit A/D; +3999 counts, +2.000 volts full scale)



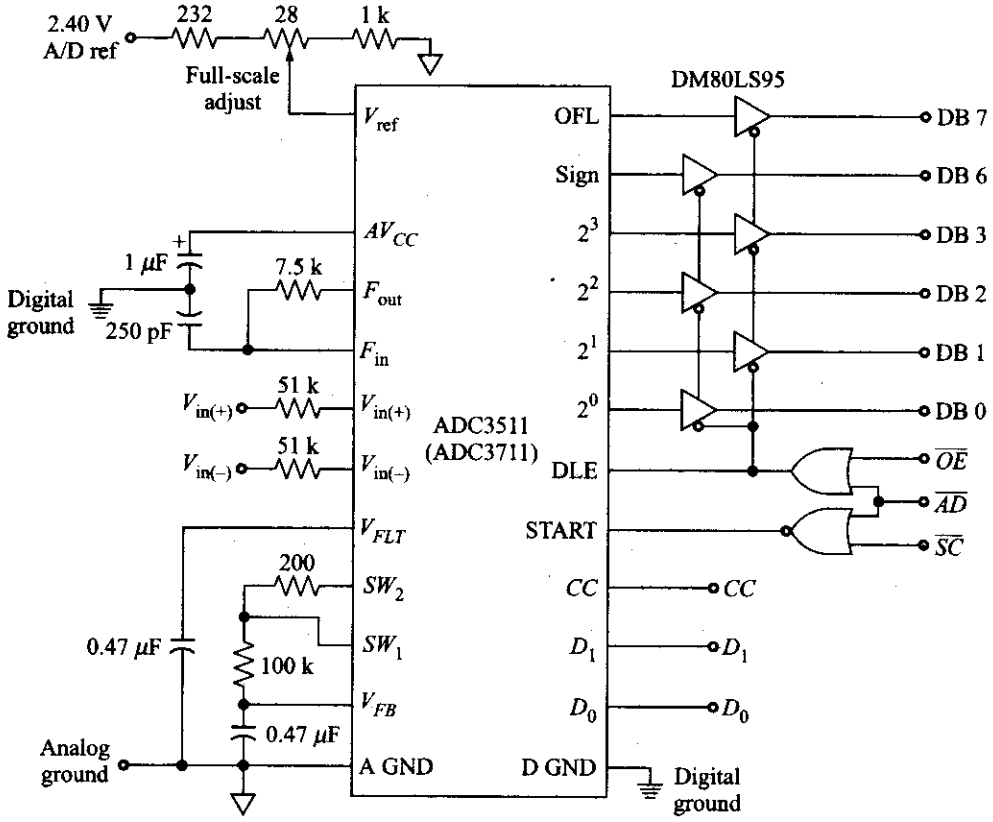
(b) 3 1/2-digit A/D; ±1999 counts, ±2.000 volts full scale (3 3/4 digit A/D; ±3999 counts, ±2.000 volts full scale)

Fig. 15.22 (Continued)

The circuit in Fig. 15.22a shows an ADC3511 (or an ADC3711) connected to convert 0.0 to +2.00 Vdc into an equivalent digital signal in BCD form. The 3511 converts to 1999 counts full scale and thus has a 1-bit resolution of 1 mV. The 3711 converts to 3999 counts full scale and has a 1-bit resolution of 0.5 mV. The circuit in Fig. 15.22b utilizes an isolated power supply such that the converter can automatically handle input voltages of both polarities—from +2.0 to −2.0 Vdc.

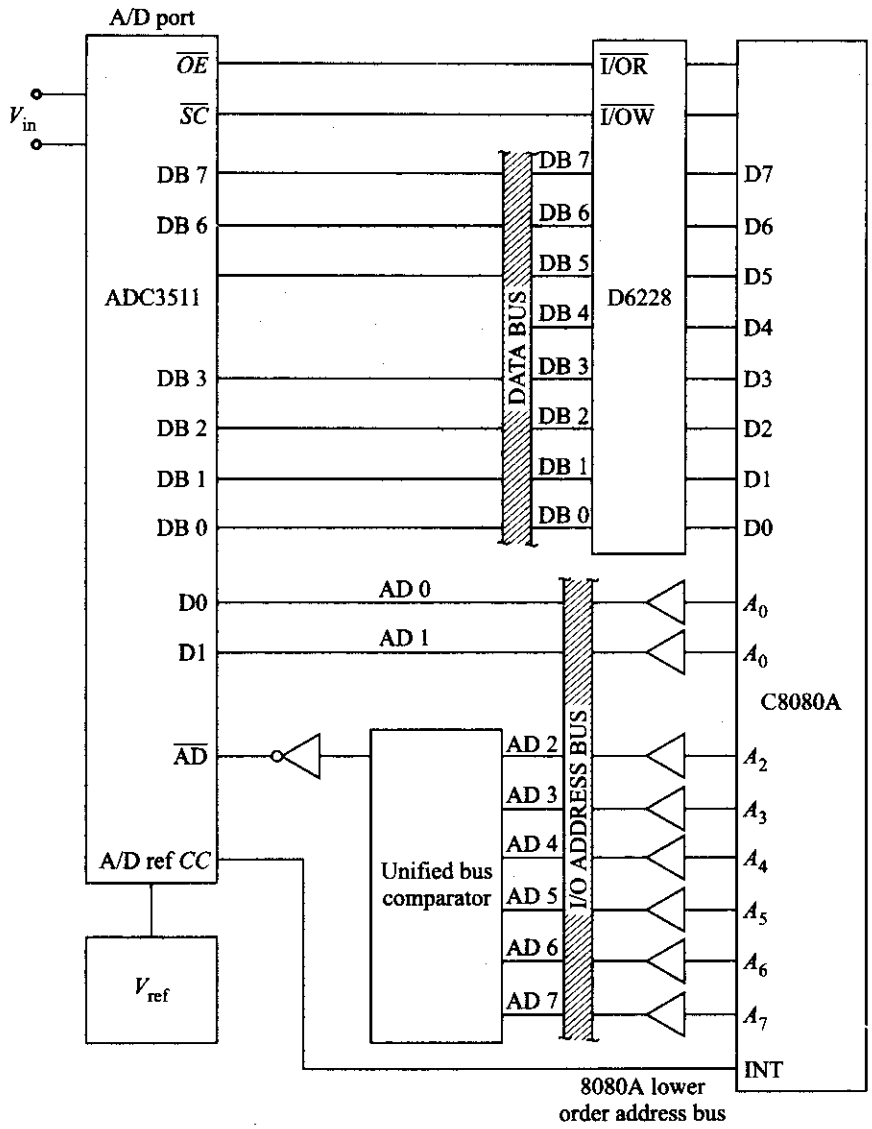
For both circuits, the reference voltage is derived from a National Semiconductor LM336, indicated by dotted lines. This is an active circuit that will provide 2.000 Vdc with a very low thermal drift of around 20 ppm/° C.

A complete circuit used to interface the ADC3511 with an 8080A microprocessor is shown in Fig. 15.23 below. Three-state bus drivers (DM80LS95) are used between the 3511 digital outputs and the microprocessor data bus, and the OR-gate–NOR-gate combination is used for control. The analog input is balanced with 51-kΩ resistors, and the 200-Ω resistor connected to SW_1 is chosen to equal the source resistance of the voltage reference; this will provide equal time constants for charging or discharging the 0.47-μF capacitor.



(a) Dual polarity A/D requires that inputs are isolated from the supply. Input range is ± 1.999 V

Fig. 15.23 (From National Semiconductor Data Acquisition Handbook; continued on next page)



(b) Single channel A/D interface with peripheral mapped I/O

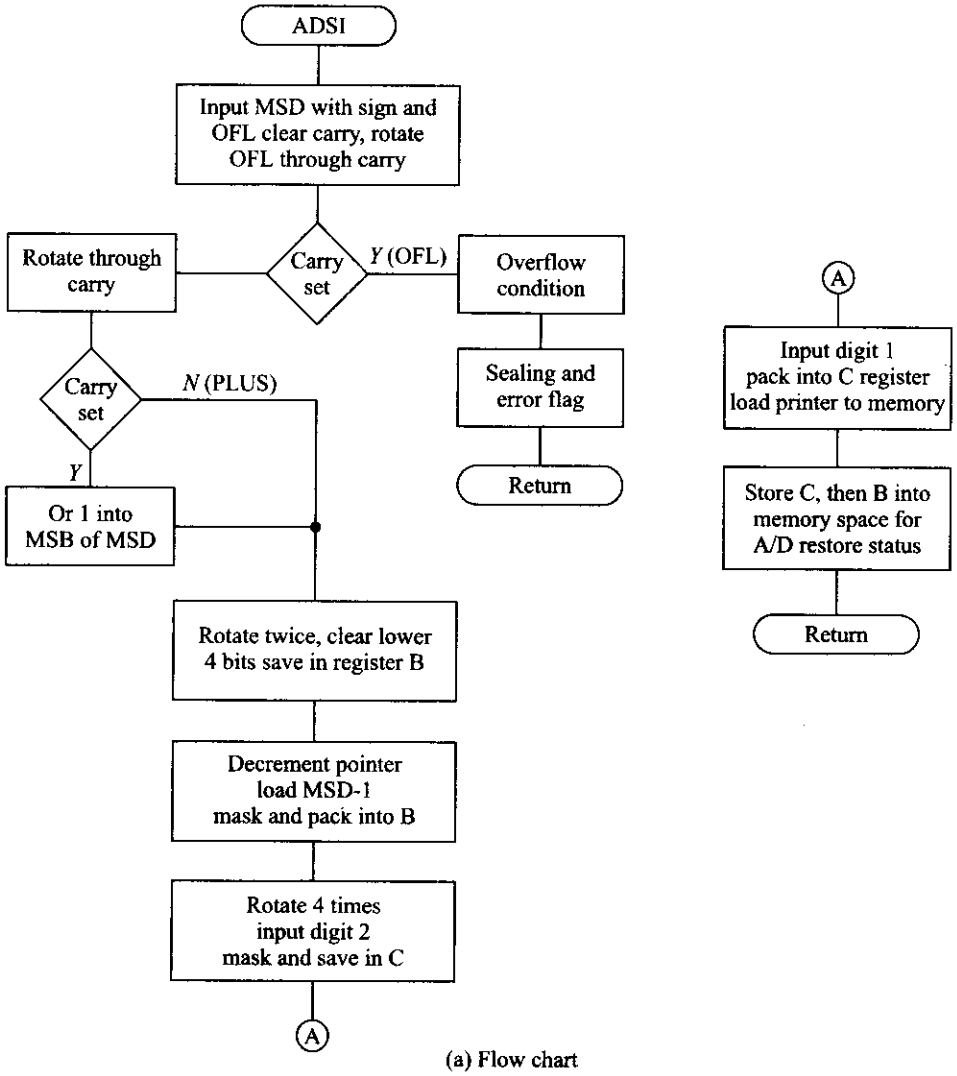
Fig. 15.23 **(Continued)**

In this application, the 3511 is a *peripheral mapped device*, which means that it is selected by an address placed on the address bus by the 8080A. The *unified bus comparator* is used to decode the proper address bits and select the ADC3511 with a low level at the $\overline{AD}$ input of the two control gates.

The CONVERSION COMPLETE output from the 3511 is used as an INTERRUPT signal to the 8080A, telling it that a digitized value is available to be read into the microprocessor. The receipt of an INTERRUPT signal causes the 8080A to read in the MSD (4 bits), the overflow (OFL), and the SIGN. If an overflow condition exists (OFL is high), an error signal is generated and the 8080A returns to its prior duties. Otherwise,

the SIGN bit is examined and stored in the MSB of digit 4 (the LSD); a negative value is denoted by a 1 in this position. The 4 bits of the LSD, that now contains the sign bit, are shifted into the upper half of the 8080A data byte. (Note that the 8080A works with 1 byte, i.e. 8 bits, of data at a time on the data bus.) The 4 bits of digit 3 are then shifted into the lower half of this byte. In a similar fashion, digits 2 and 1 are shifted into the second byte, and the four digits are now stored in the 8080A memory.

It is beyond the intent and scope of this text to include the programming required on the 8080A to interface with the ADC3511, but the flow chart and service routine given in the National Semiconductor *Data Acquisition Handbook* are included in Fig. 15.24 for the convenience of those who might presently utilize the circuit. Additional information is available in the National Semiconductor handbook.



(a) Flow chart

Fig. 15.24 **(From National Semiconductor Data Acquisition Handbook; continued on next page)**

```
Label  Opcode  Operand  Comment
ADIS:  PUSH    PSW      :A/D interrupt
                        service
       PUSH    H        :save
       PUSH    B        current status
       IN      ADD 4    :input A/D digit 4
       IN      ADD 4    :delay
       ORA              :RESET carry
       RAL              :rotate OFL through
                        carry
       JC      OFL      :overflow condition
       RAL              :rotate sign through
                        carry
       JC      PLUS     :positive input
       ORI     20H      :OR 1 into MSB
                        negative input
PLUS:  RAL              :shift
       RAL              :into position
       ANI     FO       :make lower bits
       MOV     BA       :save in B
       IN      ADD 3    :input digit 3
       IN      ADD 3    :delay
       ANI     OF       :mask higher bits
       OR      B        :pack into B
       MOV     B, A     :save in B
       IN      ADD 2    :input digit 2
```

```
Label  Opcode  Operand  Comment
       IN      ADD 2    :delay
       RAL              :rotate
       RAL              :into
       RAL              :upper
       RAL              :4 bits
       ANI     FO       :mask lower bits
       MOV     C, A     :save in C
       IN      ADD 1    :in digit 1
       IN      ADD 1    :delay
       ANI     OF       :mask upper bits
       OR      C        :pack
       MOV     C, A     :save in C
       LXI     H, ADMS  :load printer to A/D
                        memory space
       MOV     M, C     :save C in memory
       INX     H        :point next
       MOV     M, B     :save B in memory
       OUT     ADD 1    :start new
                        conversion
       POP     B        :restore
       POP     H        :previous
       POP     PSW      :status
       EI               :ENABLE interrupts
       RET              :return to main program
```

(b) Routine 1, single channel interrupt service routine

Fig. 15.24 (Continued)

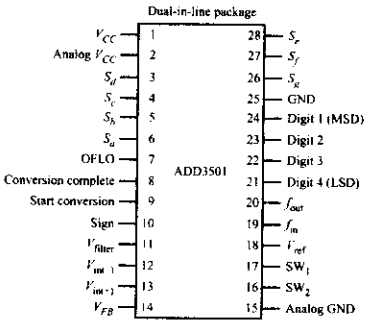
Example 15.11 Explain why it is acceptable to place the sign bit of a voltage digitized by the ADC3511 (or 3711) in the MSB of the MSD.

Solution The full-scale count for the 3511 is 1999 and for the 3711, is 3999. So, the largest value possible for the MSD in either case is 3 = 0011. Clearly the MSB is not needed for the magnitude of the MSD. It is thus convenient to specify a positive number when this bit is a 0 and a negative number when this bit is a 1.

15.6 DIGITAL VOLTMETERS

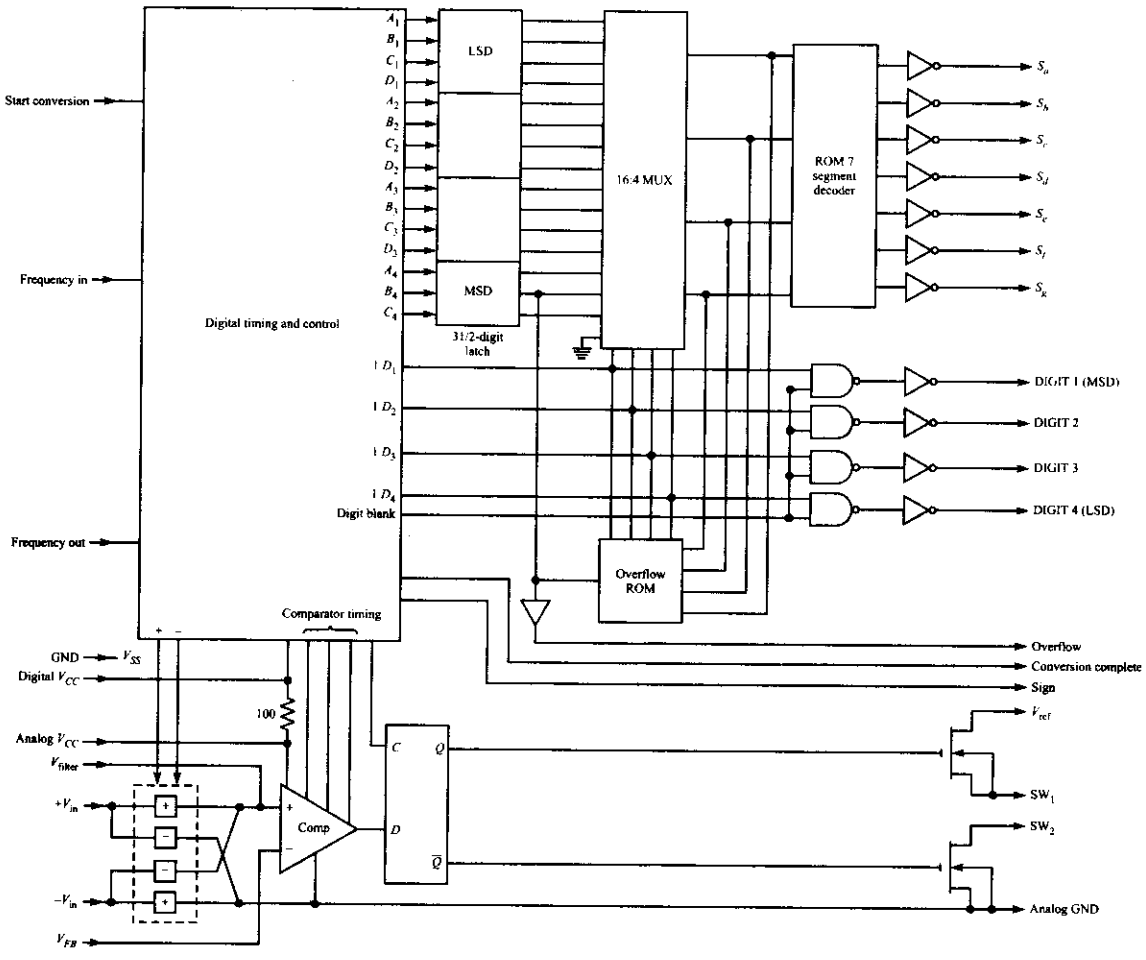
The ADC3511 (or 3711) discussed in the previous section can be used as a digital voltmeter, but it is usually more convenient to have a circuit that will drive seven-segment LED displays directly. The National Semiconductor ADD3501 is a $3\frac{1}{2}$-digit DVM constructed using CMOS technology and available in a single dual in-line package (DIP). It operates from a single +5-Vdc power supply and will drive seven-segment indicators directly. The ADD3501 is widely used as a digital panel meter (DPM) as well as the basis for constructing a digital multimeter (DMM) capable of measuring voltage, current, and resistance, and it is available at a nominal price.

The connection and logic block diagrams for an ADD3501 are shown in Fig. 15.25. The only difference between this device and the ADC3511 are the outputs. There are seven segment outputs , S_a, S_b, ... , S_g, and the four digit outputs, DIGIT 1, ... , DIGIT 4. These outputs are fully multiplexed and are designed to drive a



Order Number ADD3501CCN See NS Package N28A

Block diagram



ADD3501 3 1/2-digit DVN block diagram

Fig. 15.25 National Semiconductor ADC3501

common-cathode-type LED display directly. All the other inputs and controls are identical to the previously discussed ADC3511. The 3501 has a full-scale count of 1999 for a full-scale analog input voltage of +2.00 Vdc. A resolution of 1 bit thus corresponds to 1 mV of input voltage.

The circuit shown in Fig. 15.26 on next page shows how to use an ADD3501 as a digital voltmeter that has a full-scale analog input voltage of +2.00 Vdc. The LM309 is a voltage regulator used to reduce jitter problems caused by switching. The NSB5388 is a $3\frac{1}{2}$-digit 0.5-in common-cathode LED display. The LM336 is an active circuit which is used to provide the 2.00-Vdc reference voltage. When using this configuration, it is important to keep all ground leads connected to a single, central point as shown in Fig. 15.26, and care must be taken to prevent high currents from flowing in the analog V_{CC} and ground wires. National Semiconductor has carefully designed the circuit to synchronize the multiplexing and the A/D conversion operations in an effort to eliminate switching noise due to power supply transients.

Example 15.12 What is the purpose of the 7.5-kΩ resistor and the 250-pF capacitor connected to pins 19 and 20 of the ADD3501 in Fig. 15.26?

Solution These two components establish the internal oscillator frequency used as the clock frequency in the converter according to the relationship $f_i = 0.6/RC$. In this case, $f_i = 320$ kHz.

The DVM in Fig. 15.27 on page 589 is modified slightly in order to accommodate analog input voltages of either polarity, and also of different magnitudes. Power for the circuit is obtained from the 115-Vac power line through an isolation transformer, and the analog input is now applied at $V_{in}(+)$ and $V_{in}(-)$.

Scaling the analog input voltage for different ranges is accomplished by changing the feedback resistor between SW_1 on pin 17 and V_{FB} on pin 14, or using a simple resistance divider across the analog input. First look at the 2.00-Vdc range, since this is the normal full-scale range for the 3501. In this position, the range switch connects 100 kΩ as the feedback resistor, and the analog input goes directly to pin 13 [V_{i+}]. Also notice that the decimal point is between the 1 and the 8, giving 1.999 Vdc as a full-scale reading.

On the 0.2-Vdc scale, the range switch still applies the analog input voltage directly to pin 13. but the reference voltage at SW_1 is reduced by a factor of 10 by a resistive voltage divider before being used as a feedback voltage. The resistive divider is composed of a 90-kΩ resistor R_1, and a 10-kΩ resistor R_2. The voltage developed at the node connecting these two resistors is 0.1 V_{ref}, and so the full-scale voltage is also reduced by a factor of 10. The 90-kΩ resistor R_3 is used to keep the charging time constant essentially the same on all ranges. The time constant is given as RC = 100 kΩ × 0.47 μF. Notice that the decimal-point position has moved to pin 7 on the NSB5388 to give a full-scale reading of 199.9 mV.

On the 200-Vdc position, the range switch puts back the original feedback resistor, but the analog input voltage is reduced by a factor of 100 with a resistive voltage divider composed of 9.9-MΩ and a 100-kΩ resistor. The analog input to the 3501 is thus still 2.00 Vdc full scale even though the actual input signal is 200 Vdc full scale. The decimal point will be placed on pin 7 of the NSB5388 to give a full-scale reading of 199.9 Vdc.

On the 20-Vdc full-scale position, the range switch still uses the input voltage divider to reduce the input signal by a factor of 100, but the feedback resistor is also used to effectively increase the full scale by a factor of 10. The net result is that the 3501 will count full scale when the analog input voltage is 20 Vdc. Notice that the decimal point is now applied to pin 6 of the NSB5388 to give a full-scale reading of 19.99 Vdc.

The circuit shown in Fig. 15.28 on page 590 is a complete DMM taken from the National Semiconductor *Data Acquisition Handbook*. It utilizes the ADD3501 and is capable of measuring both dc and ac currents and voltages as well as resistances. The ranges and accuracies of the instrument are given in Fig. 15.29.

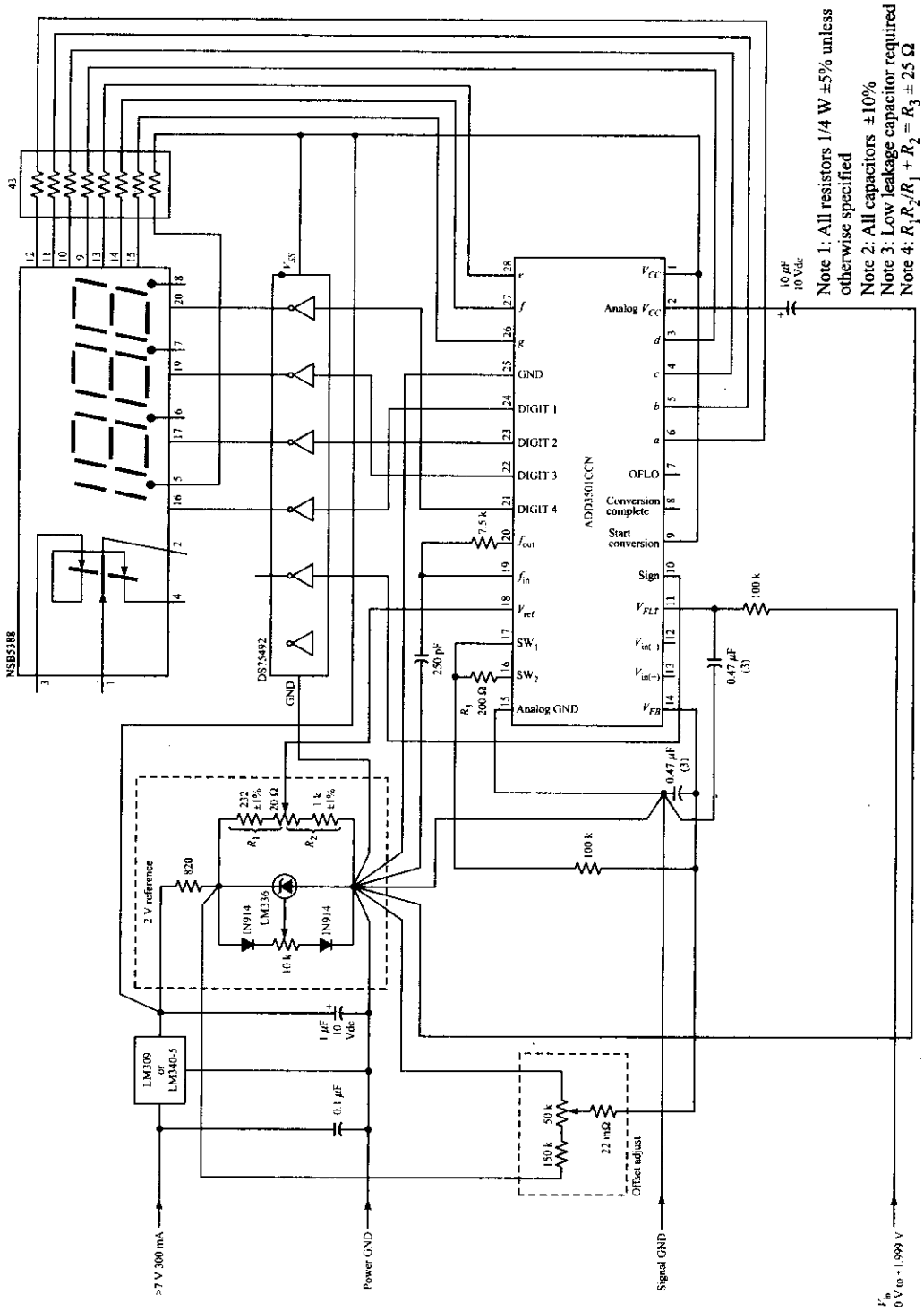


Fig. 15.26 $3\frac{1}{2}$ digit DPM, +1.999 Vdc full scale, (National Semiconductor)

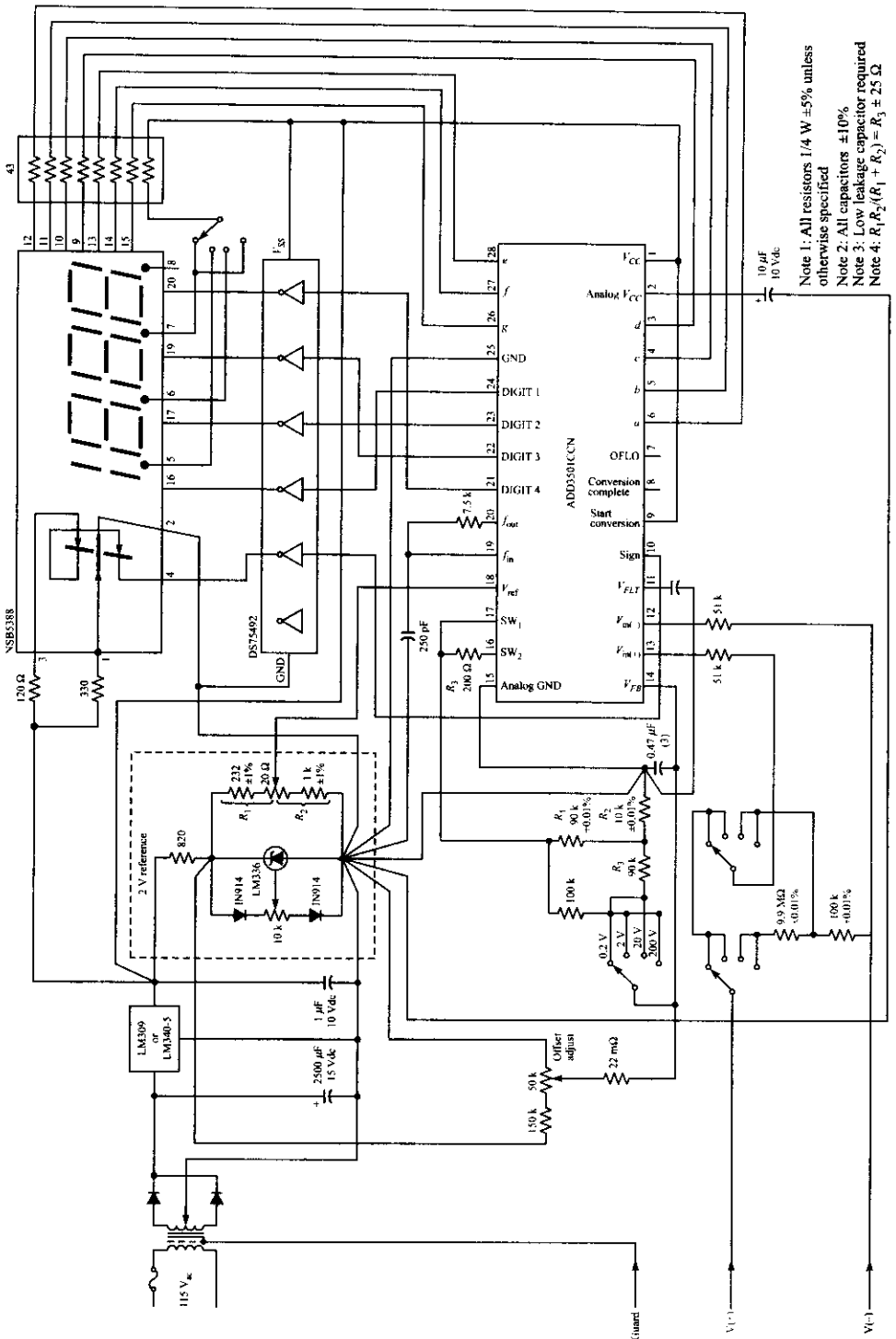


Fig. 15.27 $3\frac{1}{2}$ digit DVM, Four decade; ±0.2, +0.2, ±20, and ±200 Vdc (National Semiconductor)

Technical specifications

DC volts	< ± 1% accuracy
ranges	2 V, 20 V, 200 V, 2 kV
input impedance	2 V range, > 10 MΩ 20 V to 2 kV range, 10 MΩ
AC RMS volts	< ± 1% accuracy
ranges	2 V, 20 V, 200 V, 2 kV (40 to 5 kHz sinewave)
DC amps	< ± 1% accuracy
ranges	200 μA, 2 mA, 20 mA, 200mA, 2A
Ohms	< ± 1% accuracy
ranges	200 Ω, 2kΩ, 20kΩ, 200kΩ, 2 MΩ

Note 1: All V_{cc} connections should use a single V_{cc} point and all ground/analog ground connection should use a singic ground/analog ground point.
Note 2: All resistors are 1/4 watt unless otherwise specified
Note 3: All capacitors are ±10%
Note 4: All op amps have a 0.1 μF capacitor connected across the V+ and V-- supplier
Note 5: All diodes are 1N914

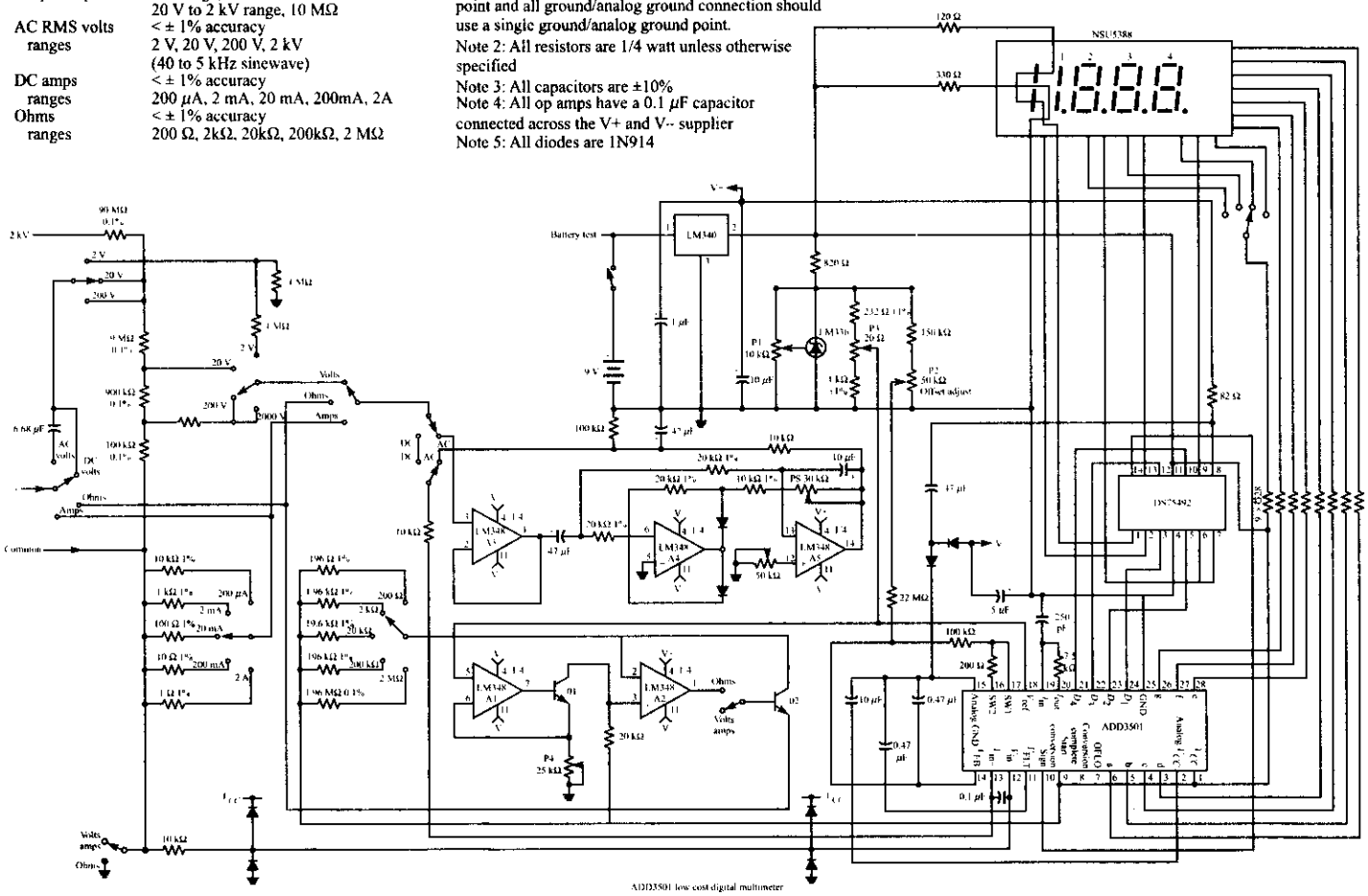


Fig. 15.28 A low-cost DMM using the ADD 3501 (National Semiconductor Data Acquisition Handbook)

The different dc and ac voltage ranges are accommodated by a resistive voltage divider at the analog input. Alternating-current voltages are measured by using the three operational amplifiers A_3, A_4, and A_5 to develop a dc voltage that is proportional to the root-mean-square (RMS) value of the ac input voltage.

Measurement mode	Range					Frequency response	Accuracy	Overrange display
	0.2	2.0	20	200	2000			
DC volts	–	V	V	V	V	–	≤ 1% FS	± OFLO
AC volts	–	V_{RMS}	V_{RMS}	V_{RMS}	V_{RMS}	40 Hz to 5 kHz	≤ 1% FS	+ OFLO
DC amps	mA	mA	mA	mA	mA	–	≤ 1% FS	± OFLO
AC amps	mARMS	mARMS	mARMS	mARMS	mARMS	40 Hz to 5 kHz	≤ 1% FS	+ OFLO
Ohms	kΩ	kΩ	kΩ	kΩ	kΩ	–	≤ 1% FS	+ OFLO

Fig. 15.29 Performance of the DMM in Fig. 15.28

A series of current-sensing resistors are used to measure either dc or ac current. The current to be measured is passed through one of the sensing resistors, and the DMM digitizes the voltage developed across the resistor.

The DMM measures resistance by applying a known current from an internal current source (operational amplifiers A_1 and A_2) to the unknown resistance and then digitizing the resulting voltage developed.

For those interested in pursuing this subject, complete details for the construction and calibration of this DMM are given in the *National Semiconductor Data Acquisition Handbook*.

SUMMARY

The primary objective of this chapter is to demonstrate the use of many of the most fundamental principles discussed throughout the text by considering some of the more common digital circuit configurations encountered in industry. Multiplexing of LED displays, time and frequency measurement, and use of digital voltmeters of all types are widely used throughout industry. Although our coverage is by no means comprehensive, it will serve as an excellent introduction to industrial practices.

The problems at the end of this chapter will also provide a good transition into industry. They are in general longer than previously assigned problems. All the necessary information required to work a given problem may not be given—this is intentional since it will require you to seek information from industrial data sheets. However, the problems are more of a design nature, and usually deal with a practical, functional circuit that can be used to accomplish a given task; as such, they are much more interesting and satisfying to solve.

PROBLEMS

In order to solve some of these problems, you may have to consult product data sheets that are not included in this text. It is intended that you discover a source for such information.

15.1 Pick one of the solutions suggested in Example 15.3 and do a detailed design, including part numbers and pin numbers.

15.2 Design a four-decimal-digit multiplexed display like the one in Fig. 15.7, but use common-cathode-type LEDs. Use a basic circuit like the one in Fig. 15.2, but you will now need to generate DIGIT waveforms that have positive pulses.

15.3 How often is each digit in Fig. 15.7 serviced, and for what period of time is it illuminated? Extinguished?

15.4 Design a multiplexed display like the one in Fig. 15.8 having eight decimal digits. Use a three-flip-flop multiplexing counter and four 74151 multiplexers.

15.5 Specify a ROM that could be used in place of the 7447 in Fig. 15.8. Draw a circuit, showing exactly how to connect it.

15.6 Design a four-decimal-digit display using 54/74143 and common-anode LEDs.

15.7 Using Fig. 15.12 as a pattern, design a four-digit frequency counter using 54/74143s and 54/74160s. Use a 1.0-MHz clock, and provide 0.1-, 1.0-, and 10.0-s gates. Specify the frequency range for each gate.

15.8 Following Fig. 15.12 as a guide, design a four-digit frequency counter using National Semiconductor MM74C925.

15.9 Design a circuit to measure "elapsed time" between two events in time—for instance, the time difference between a pulse occurring on one signal followed by a pulse occurring on another signal. Use as much of Fig. 15.14 as possible, but consider using a set-reset flip-flop in conjunction with the two input signals.

15.10 Combine the circuits in Figs. 15.12 and 15.14 into a single instrument. Use a 1.0-MHz clock and seven decade counters, define the scales and readouts carefully.

15.11 What is the internal clock frequency of the ADC0804 in Fig. 12.29 if the capacitor C is changed to 100 pF?

15.12 In stand-alone operation, how often does the ADC0804 do an A/D conversion?

15.13 Assuming that V_{CC} = +5.0 Vdc, determine the digital outputs of an ADC0804 for analog inputs of:

a. 1.25 V, b. 1.0 V

c. 4.4 V

15.14 Assuming that V_{CC} = +5.0 Vdc, in Fig. 12.29, determine the analog input voltages that will produce a digital output of:

a. 1000 1100 b. 25H

c. 0001 1000?

15.15 The ADC0804 in Fig. 12.29 is to be used with an analog signal that varies between 0.0 and +3.3 V. Determine a new value for V_{ref}.

15.16 The ADC0804 in Fig. 15.16b is to be used with an analog signal that varies between +2.2 and +3.3 V. Determine a new value for V_{ref} and V_{i-}.

15.17 Use the ADC0804 and design a stand-alone circuit to digitize an analog voltage that ranges between +0.25 and +5.0 V.

15.18 Use the ADC0804 and design a stand-alone circuit to digitize an analog voltage that ranges between –2.5 and +2.5 V.

15.19 Design a resistive voltage divider to use with the ADC3511 such that it will digitize an analog input voltage of 20 Vdc as full-scale voltage input. What is the resolution in millivolts for this design?

15.20 Design a voltage divider such that the DVM in Fig. 15.27 will measure full-scale voltages of 2.0, 20.0 and 200.0 Vdc without changing the feedback resistor. Leave the feedback resistor at 100 kΩ. Draw the complete design. Is it possible to achieve a full scale of 0.2 Vdc for this circuit without changing the feedback resistor?

Answers to Self-tests

1. R and C are needed to set the internal clock frequency.
2. Depressing the START button begins the A/D conversion process.
3. C3H
4. Span is the range of input voltage. OFFSET is the lowest value of analog input voltage.