TABLE 6.7 CHARACTERISTICS FOR AN IBM-COMPATIBLE FLEXIBLE-DISK DRIVE

CHARACTERISTIC	SPECIFICATION
Capacity	400,000 bytes†
Rotational speed	360 rpm
Transfer rate	250,000 bits/s
Track-to-track access time	16–20 ms
Average access time	176 ms
Bit density	
Inner track	3268 bits/in.
Outer track	1836 bits/in.
Track density per inch	48
Number of tracks	77

†This is a maximum; with IBM's formatted recording system, only 250,000 bytes of data are recorded per disk.

is 0.012 in. wide, and standard track spacing is 48 per inch. The number of tracks is 77. The capacity of a surface using standard code and a bit density of 3268 bits/in. on the innermost track is about 400,000 bytes of 8 bits each. Table 6.7 shows some characteristic of this kind of flexible-disk system.

Disks are also made in 5¼- and 3½-in. sizes, and these minidisks are now the most popular. Several companies offer floppy-disk systems using adaptations of the flying-head concept, so the disk surface is not worn when used. Some floppy disks use an address system (like regular disks) in which the disk drive, track, and sector are given. However, other systems write *headers* on each block of recorded data on a track, and the header information is specified for each access.

The "IBM standard" system uses one complete track for formatting information. Sync bits and headers as well as check bits are interlaced with data on the remaining tracks. A complete description of IBM's formatting can be found in The IBM Diskette for Standard Data Interchange, IBM document GA21-9182-01.

To increase disk capacity, manufacturers now supply two-sided double-track-density (100 tracks per inch) and double-density (7 kbits/in.) drives. Less expensive drives normally have fewer tracks.

Table 6.8 shows some typical characteristics for small systems.

TABLE 6.8 FLOPPY-DISK CHARACTERISTICS

CHARACTERISTIC	8-IN. DISK DRIVE	5.25-IN. DISK DRIVE	3.5-IN. DISK DRIVE
Capacity (unformatted)			
Single density, kbytes	800	220	218.8
Double density, kbytes	1600	440	437.5
Average access time, ms	200	300	200
Transfer rates			
Single density, kbits/s	250	250	250
Double density, kbits/s	500	500	500
Number of tracks	154	100	70
Rotational speed, rpm	360	300	600
Track density per inch	130	130	145

MAGNETIC TAPE

6.12 At present, the most popular medium for storing very large quantities of information is magnetic tape. Although magnetic tape is not a desirable medium for the main high-speed storage of a computer because of its long access time, modern mass-production techniques have made the cost of tape very low. Thus vast quantities of information may be stored inexpensively. Furthermore, since it is possible to erase and rewrite information on tape, the same tape may be used again and again. Another advantage of magnetic tape is that the information stored does not "fade away," and therefore data or programs stored one month may be used again the next.

Another advantage of using magnetic tape for storing large quantities of data derives from the fact that the reels of tape on a tape mechanism may be changed. In this way the same magnetic-tape handling mechanism and its associated circuitry may be used with many different reels of tape, each reel containing different data.

There are four basic parts of a digital magnetic-tape system:

1 *Magnetic tape* This is generally a flexible plastic tape with a thin coating of some ferromagnetic material along the surface.

2 *Tape transport* This consists of a mechanism designed to move the tape past the recording heads at the command of the computer. Included are the heads themselves and the storage facilities for the tape being used, such as the reels on which the tape is wound.

3 *Reading and writing system* This part of the system includes the reading and writing amplifiers and the *translators*, which convert the signals from the tape to digital signals that may be used in the central computing system.

4 *Switching and buffering equipment* This section consists of the equipment necessary to select the correct tape mechanism if there are several, to store information from the tape and also information to be read onto the tape (provide buffering), and to provide such tasks as manually directed rewinding of the tape.

The tape transports used in most digital systems have two unique characteristics: the ability to start and stop very quickly and a high tape speed. The ability to start and stop the tape very quickly is important for two reasons. First, since the writing or reading process cannot begin until the tape is moving at a sufficient speed, a delay is introduced until the tape gains speed, slowing down operation. Second, information is generally recorded on magnetic tape in *blocks*, or *records*. Since the tape may be stopped between blocks of information, the tape which passes under the heads during the stopping and starting processes is wasted. This is called the *interblock*, or *interrecord*, *gap*. Fast starting and stopping conserves tape.[14]

Figure 6.30(*a*) shows a typical tape system. To accelerate and decelerate the tape very quickly, an effort is made to isolate the tape reels, which have a high inertia, from the mechanism that moves the tape past the recording heads. Figure

[14]There is a type of tape drive called a *streaming tape drive* which does not start or stop quickly but has a high tape speed. This is described in a later section.

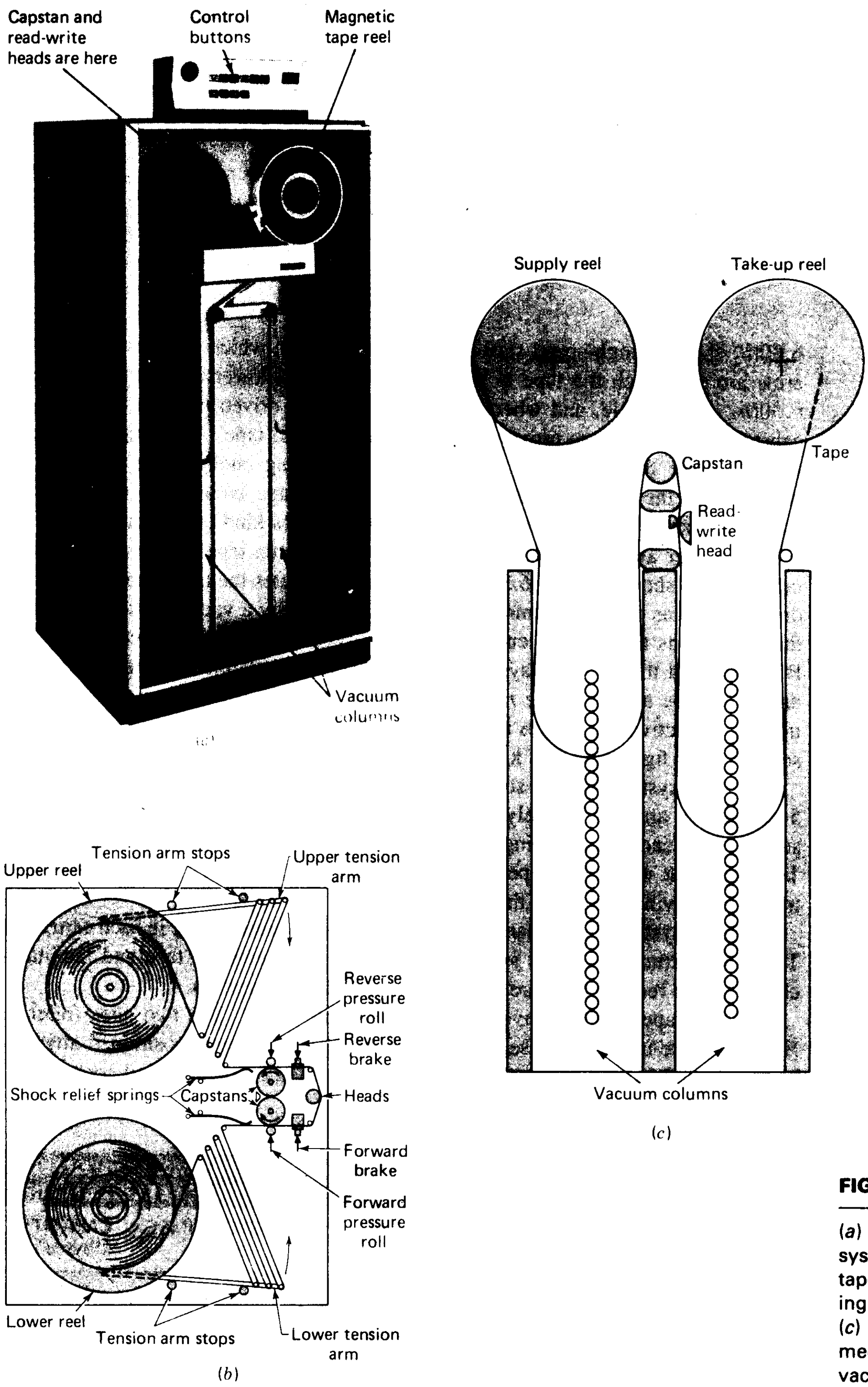


FIGURE 6.30

(*a*) IBM 3420 tape system. (*b*) Magnetic-tape mechanism using tension arms. (*c*) Magnetic-tape mechanism using vacuum columns.

TABLE 6.9 TYPICAL TENSION ARM TAPE DRIVE CHARACTERISTICS

REEL SIZE, in.	bits/in.	DATA RATE, kbits	START-STOP TIME, ms	CAPACITY, Mbytes	COST OF A TYPICAL REEL OF TAPE, $
7	1600	40	15	[illegible]	[illegible]
8.5	1600	60	10	[illegible]	[illegible]
10.5	1600	72 kbits/s at 45 in./s; 120 kbits/s at 75 in./s	8.33 [illegible] 75 [illegible]	[illegible]	[illegible]

6.30(*b*) shows a high-speed start-stop tape mechanism which uses a set of tension arms around which the tape is laced. The upper and lower tension arms in Fig. 6.30(*b*) are movable, and when the tape is suddenly driven past the heads by the capstan, the mechanism provides a buffering supply of tape. A servomechanism is used to drive the upper and lower reels, maintaining enough tape between the capstan and the tape reels to keep the supply of tape around the tension arms constant. Table 6.9 shows some characteristics of this kind of system.

Another arrangement for isolating the high-inertia tape reels from the basic tape drive is shown in Fig. 6.30(*c*). This system isolates the tape from the capstan drive by means of two columns of tape held in place by a vacuum. A servosystem then maintains the correct length of tape between the reel and the capstan drive. Both this and the previous systems use continuously rotating capstans to actually drive the tape and *pressure rolls* to press the tape against the capstan when the transport is activated. Brakes are also provided for fast stopping. Table 6.10 shows some typical figures on this kind of system.

When systems of this sort are used, the start and stop times can be less than 5 ms. These are, respectively, the times required to accelerate a tape to a speed suitable for reading or writing and the time required to fully stop a moving tape. The speeds at which the tapes are moved past the heads vary greatly, most tape transports having speeds in the range of 12.5 to 250 in./s.

Some systems have changeable cartridges with a reel of tape in each cartridge. The manufacturers of these systems feel that this protects the tape and facilitates changing the reels. These are discussed in a following section.

Most tape systems have two-gap read-write heads. The two gaps (refer to Fig. 6.31) are useful because, during writing, the read gap is positioned after the write gap and is used to check what has been written by reading and comparing.

TABLE 6.10 TYPICAL VACUUM COLUMN TAPE DRIVE CHARACTERISTICS FOR 10.5-IN. TAPE REEL SYSTEMS

SPEED, in./s	bits/in.	MAXIMUM DATA RATE, kbytes/s	START-STOP TIMES, ms	CAPACITY, Mbytes
50	1600	80	7.5	46
75	1600	120	5	46
200	1600	320	3	62
200	6250	1250	1.2	350

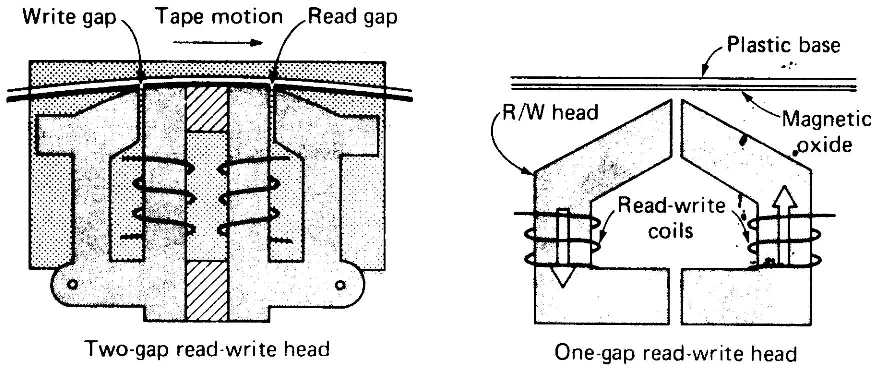


FIGURE 6.31

One- and two-gap tape heads.

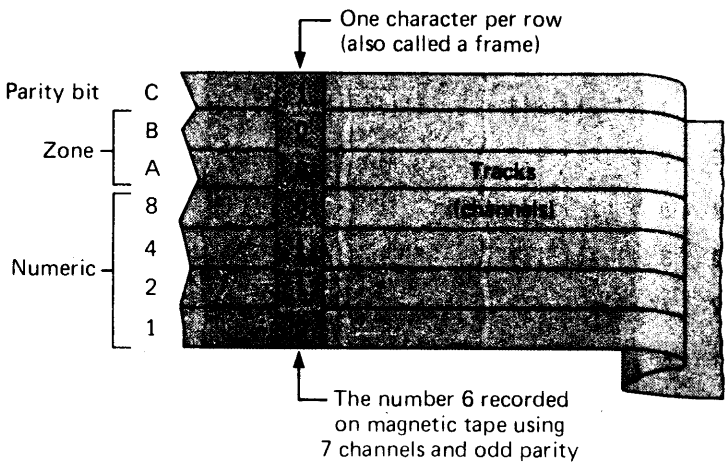
Tapes vary from $\frac{1}{4}$ to 3 in. in width; however, most tape is $\frac{1}{2}$-in.-wide 1.5-mil-thick Mylar tape. A 10.5-in. reel typically has 2400 or 3600 ft of tape. Generally about nine channels or tracks are used for $\frac{1}{2}$ in. of width. The surface of the tape is usually in contact with the read-write heads. Output signals from the read heads are generally in the 0.1- to 0.5-V range. The recording density varies; however, 200, 556, 800, 1600, 6250, and even 12,500 bits/in. per channel are standard.

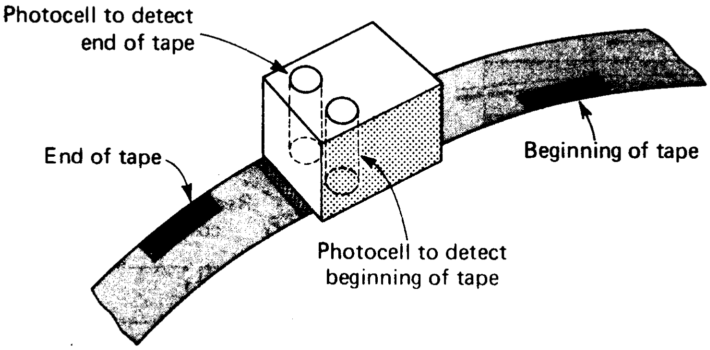
Data are recorded on magnetic tape by using some coding system. Generally one character is stored per row (refer to Fig. 6.32) along the tape. The tape in Fig. 6.32 has seven tracks, or channels; one of these is a parity bit, which is added to make the number of 1s in every row odd (we study this in the next chapter as well as the codes used for magnetic tape). Data are recorded on magnetic tape in blocks, with gaps between the blocks and usually with unique start and stop characters to signal the beginning and the end of a block.

A small piece of metallic reflective material is fastened to the tape at the beginning and end of the reel, and photoelectric cells are used to sense these markers and prevent overrunning of the tape [refer to Fig. 6.33(*c*)].

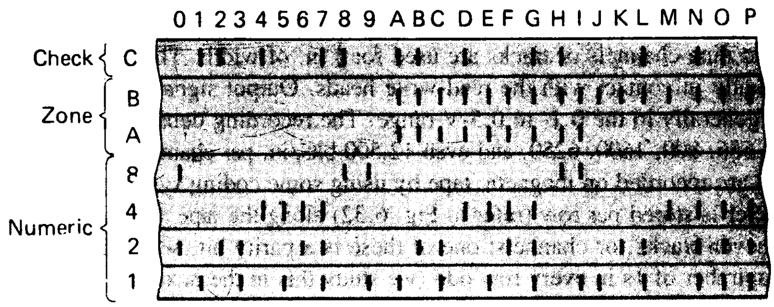
FIGURE 6.32

Basic layout of magnetic tape.

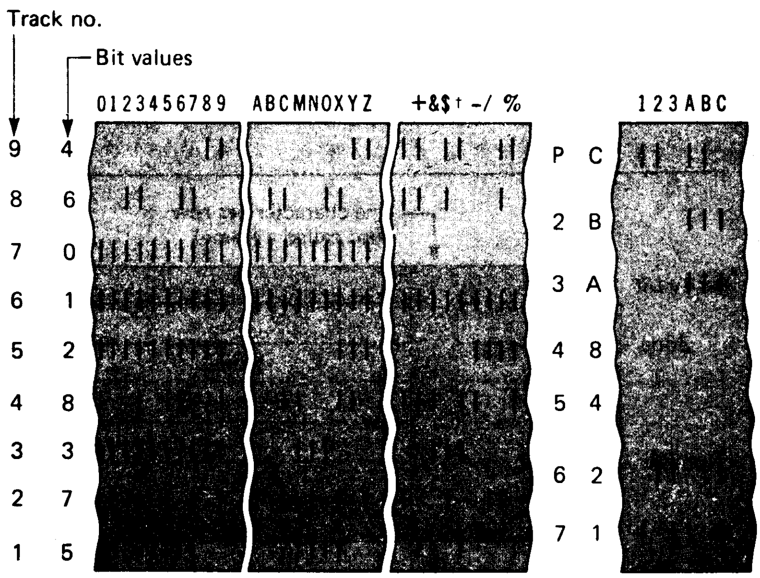




(a)



(b)



† The P bit position produces odd parity.

(c)

FIGURE 6.33

Magnetic-tape coding. (a) Beginning and end of tape marking. (b) Magnetic recording of seven-track BCD code on tape. (c) Nine-track (EBCDIC) and seven-track tape data format comparison.

TABLE 6.11 VACUUM COLUMN DIGITAL TAPE TRANSPORT CHARACTERISTICS FOR THE IBM 3420-8

CHARACTERISTIC	SPECIFICATION
Data density	7 or 9 tracks, 1600 or 6250 characters/in.
Tape velocity	200 in./s
Start-stop time	3 ms at 200 in./s
Start-stop displacement	0.8214 in.
Reel diameter	10.5 in.
Tape	
Length	2400 ft
Width	0.5 in.
Thickness	1.5 mils
Rewind speed	300 in./s nominal

The codes used to record on tape vary, but two commonly used IBM codes are shown in Fig. 6.33(*b*) and (*c*). IBM standard tape is $\frac{1}{2}$ in. wide and 1.5×10^{-3} in. thick, with either seven or nine tracks. The seven-track code is shown in Fig. 6.33(*b*), where the 0s are simply blank and 1s are indicated by a vertical line. Figure 6.33(*c*) shows the nine-track code. Recording densities are 200, 556, 800, 1600, or 6250 bits (or rows) per inch [which means 200, 556, 800, 1600, or 6250 characters (or bytes) per inch, since a character is recorded in each row].

Some characteristics of a medium-priced vacuum column tape system are shown in Table 6.11. Table 6.12 gives the characteristics for an inexpensive tension arm Hewlett-Packard system.

TAPE CASSETTES AND CARTRIDGES

6.13 The changeable tape cassette used in the familiar home recorder is an attractive means for recording digital data. The cassettes are small, changeable, and inexpensive; they are frequently used in small and "home" computers. Unfortunately, the tape-moving mechanism in the conventional home tape cassette

TABLE 6.12 HEWLETT-PACKARD 7090E TAPE SYSTEMS

CHARACTERISTIC	SPECIFICATION
Number of tracks	9
Read-write speed	
2100-based systems	25, 37.5, 45 in./s
3000 system	4 in./s
Density	1600 characters/in. (8 bits/character)
Data transfer rate	72,000 characters/s maximum
Reel diameter	10.5 in. maximum
Tape (computer grade)	
Width	0.5 in.
Thickness	1.5 mils
Rewind speed	160 in./s
Start-stop time	8.33 ms (read after write) at 45 in./s
End of tape and beginning of tape reflective-strip detection	IBM-compatible

often used for small systems is not of sufficient quality for larger business and scientific computer usage. However, a number of high-quality digital cassettes with prices in the dollar region ($2 to $15 in general) have been developed. These are small—on the order of the familiar audio cassette—and have a similar appearance.

There are also larger tape cartridges which contain long strips of magnetic tape and which resemble large cassettes. These cartridges provide a more convenient way to package tape and greatly simplify the mounting of tape reels (which can be a problem with conventional reels of tape where the tape must be manually positioned on the mechanism). The tape cartridges also provide protection against dirt and contamination, since the tape is sealed in the cartridges.

A number of different digital cassettes and cassette drives are now in production, and each has different characteristics.[15] For example, TEAC offers a cassette drive with a tape speed of 15 in./s, 282 ft 0.15 in. of magnetic tape per cassette, and 1600-bits/in. tape density. The cassette can rewind in 48 s. A 22-in. reflective leader and trailer are used to mark the beginning and the end of the tape (a photodiode senses this strip).

Cartridges are a high-performance magnetic-tape storage medium. Several cartridge designs are available. These vary not only in performance capabilities, but also in the division of hardware between cartridge and transport.[16] The 3M cartridge and drive shown in Fig. 6.34 are representative. The cartridge contains 300 ft of $\frac{1}{4}$-in. tape capable of recording up to four tracks at 1600 bits/in. for a

[15]Standards organizations have attempted to develop standards for cassettes. The Phillips cassette is such a standard.
[16]For instance, the heads may or may not be included in each cartridge.

FIGURE 6.34

Digital cartridge and interface. *(3M Co.)*

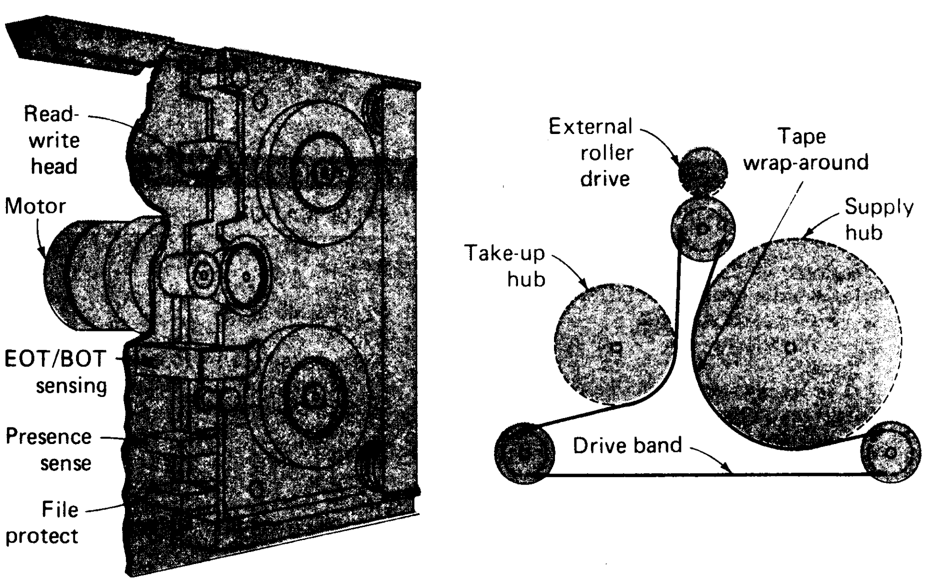


TABLE 6.13 SPECIFICATIONS OF 3M CARTRIDGE AND DCD-3 CARTRIDGE DRIVE

CHARACTERISTIC	SPECIFICATION
Operating speed	
Read-write	30 in./s forward and reverse
Fast forward, rewind, gap search	90 in./s forward and reverse
Packing density	6400 bits/in.
Transfer rate	48 kbits/s maximum
Interrecord gap	1.33 in. typical; 1.2 in. minimum per proposed ANSI standard
Maximum recommended start-stop rate	Three operations per second without forced-air cooling
Total speed variation	±4% maximum
Tape head	1-, 2-, 4-channel read-while-write heads available
Interface logic	TTL-compatible
Power	5 V dc ± 5%, ±18 V dc ± 5%

maximum storage capacity of more than 2×10^7 bits. The 3M transport operates at 30 in./s during reading or writing and at 90 in./s in the search mode. A novel elastic-band drive moves the tape and also supplies tape tension. Tape drive, hub, and guide components are referenced to the base of the cartridge and require no external guidance. Several new cartridge systems have been designed to back up Winchester disk drives.

Table 6.13 gives some specifications of the 3M cartridge and the 3M cartridge drive. The great popularity of the Winchester disk drives has led to the development of a new type of tape drive called a *streaming tape drive*. Because most Winchester drives have fixed disks, there is a need to *back up*, that is to write and store elsewhere, the contents of Winchester disks. This can be done with floppy disks, but they are expensive per bit and somewhat slow; with disk cartridges, but they do not have great capacity per cartridge; or with standard 0.5-in. tape drives, but these are liable to be expensive for small systems. Thus a small, inexpensive type of drive, the streaming tape drive, competes in the market for the Winchester backup function.

The streaming tape drive is small and uses 0.25-in. tape in a cartridge. An important design characteristic is that the tape is moved past the read-write heads by driving the reels, not by using a capstan as in most tape systems. This leads to fast tape movement, but starts and stops are slow and so interrecord gaps are long. As a result, a streaming tape drive can be used to transfer (or read) large quantities of data once the tape is in motion. Tape speeds of 90 in./s are common, and the tape lengths are about 450 ft per cartridge. Transfer rates can be as high as 100 kbytes/s. A single tape can store up to 67 Mbytes per cartridge. Densities of 6400 bits/in. along a track are common, and four or five tracks are generally used.

Table 6.14 compares some of the standard memory devices just described. Notice that these are standard systems, and some newer devices may exceed these characteristics. Table 6.15 gives some data on the newer low-cost devices suitable for minicomputers and microcomputers as opposed to more expensive devices. These are again representative figures for the latest systems.

TABLE 6.14 STORAGE MEDIA COMPARISONS

CHARACTERISTIC	5-in. REEL	PHILLIPS CASSETTE	3M-TYPE CARTRIDGE	FLOPPY DISK	LARGE HARD DISK FIXED MEDIA
Capacity, kbytes	18,500	550	2500	1600	571,000
Transfer rate, kbits/s	180	9.6	48	500	14,000
Number of tracks	9	2	4	200	600
Density, bits/in.	800	880	1600	3200	1600
Interrecord gap, in.	0.6	0.8	1.3	Not applicable	Not applicable
Mechanism cost, $	1000	400	500	100	30,000
Media cost, cents/byte	0.06×10^{-4}	1.2×10^{-3}	0.6×10^{-4}	2.6×10^{-4}	0.185×10^{-4}

TABLE 6.15 LOW-COST STORAGE SYSTEM CHARACTERISTICS

CHARACTERISTIC	CAPACITY	COST, cents/byte
Floppy disk	2.4 Mbits	2.6×10^{-4}
High-performance cassette	1 Mbyte	0.5×10^{-4}
Phillips cassette	1.44 Mbytes	1.2×10^{-4}
Low-performance cassette	200 kbits	20×10^{-4}
3M cartridge	11.5 Mbytes	0.6×10^{-4}
7-in. tape reels	40 Mbits	0.5×10^{-4}

MAGNETIC BUBBLE AND CCD MEMORIES

6.14 The secondary, or backup, memory devices that have been really successful so far have all been electromechanical devices (drums, disks, tape, etc.) which store bits as magnetic fields on a surface and rely on mechanical motion to locate the data. However, two devices for secondary storage having no moving parts are now being developed and have started to appear in some commercial applications. These are magnetic bubble and CCD memories.

Magnetic bubble memories are primarily competing with floppy disks, small disks, cartridges, and small tape devices. Bubble memories are more reliable (having no moving parts), consume less power, are smaller, and cost less per unit. However, disks have higher transfer rates, and the cost per bit is lower except for very small systems.

Bubble memories trace their history to research at the Bell Laboratories, which showed that bits can be stored as "bubbles" in a thin magnetic film formed on a crystalline substrate. A bubble device operates as a set of shift registers. The storage mechanism consists of cylindrically shaped magnetic domains, called *bubbles*. These bubbles are formed in a thin-film layer of single-crystal synthetic ferrite (or garnet) when a magnetic field is applied perpendicular to the film's surface. A separate rotating field moves the bubbles through the film in shift register fashion. The presence of a bubble is a 1; no bubble is a 0. The bubbles move along a path determined by patterns of soft magnetic material deposited on the magnetic epitaxial film.

To the user, the physics of the bubble memory's operation are less important than its operating characteristics. The memories appear as long shift registers which can be shifted under external control. Storage is permanent since if shifting is stopped, the bits in the memory will remain indefinitely.

To utilize the shift register characteristics better and reduce access time, the shift registers are generally made of only modest lengths of perhaps 50 to 100 bits. A memory package is liable to contain from a few hundred kilobits to several megabits.

The shift rate is relatively slow, perhaps 400 MHz, so access times are on the order of a few milliseconds. (Reading and writing are performed only at the ends of the shift register.)

Bubble memories require relatively complex interface circuitry, but IC manufacturers have produced reasonable IC packages for this purpose.

Charged-coupled devices (CCDs) are constructed by using IC technology. The bits are stored on capacitors as charges similar to the dynamic IC memories, except that the storage is arranged in a shift register configuration with the charge "packets" being shifted from cell to cell under clock control.

Since the storage mechanism is a charge on a capacitor, if shifting stops for very long (a few milliseconds), the charges will leak from the capacitors and the memory's contents will be lost.

CCD memories generally have from 500 kbits to several megabits of storage. The shift registers are read from and written into from the ends, so access time is dependent on shift register lengths. The shift rate is generally 200 to 500 kHz, and so for reasonable-length shift registers access times are in the milliseconds.

Since CCD memories use IC technology, they require less interface circuitry than bubble memories. The strategy involved in determining how long the shift registers should be for both bubbles and CCDs is based on a cost/performance analysis. A greater number of shorter loops results in faster access times, but more interface circuits and more complicated system usage strategies. Long loops give economy but long access times.

Both bubble and CCD technologies are in the early stages, but they are already considered competitive with the smaller, more conventional disk memories.

DIGITAL RECORDING TECHNIQUES

***6.15**[17] Although the characteristics and construction of such storage devices as tape recorders and magnetic disk storage devices may vary greatly, the fundamental storage process in each consists of storing a binary 0 or 1 on a small area of magnetic material. Storage in each case is dynamic, for the medium on which the information is recorded is moved past the reading or writing device.

Although the process of recording a 0 or a 1 on a surface may appear straightforward, considerable research has gone into both the development of the recorded patterns used to represent 0s and 1s and the means for determining the value recorded. There are two necessities here: (1) The packing density should be made as great as is possible; that is, each cell or bit should occupy as little space as possible, thus economizing, for instance, on the amount of tape used to store a given amount of information. (2) The reading and writing procedure should be

[17]Sections with asterisks can be omitted on a first reading without loss of continuity.

made as reliable as possible. These two interests are conflicting because when the recorded bits are packed more and more closely together, the distortion of the playback signal is greatly increased.

In writing information on a magnetic surface, the digital information is supplied to the recording circuitry, which then codes this information into a pattern that is recorded by the write head. The techniques used to write information on a magnetic medium can be divided into several categories: the *return-to-zero* (RZ) technique, the *return-to-bias* (RB) technique, and the *nonreturn-to-zero* (NRZ) technique. The methods for reading information written by using these techniques also vary. The basic techniques are described below, along with the recorded waveshapes and the waveshapes later read by the read heads and translated by the reading system.

RETURN-TO-ZERO AND RETURN-TO-BIAS RECORDING TECHNIQUES

***6.16** Figure 6.35 illustrates the return-to-zero recording technique. In Fig. 6.35(*a*) no current goes through the winding of the write head, except when a 1 or a 0 is to be recorded. If a 1 is to be recorded, a pulse of positive polarity is applied to the winding on the write head; and if a 0 is to be written, a negative pulse is applied to the winding. In either case, the current through the write-head winding is returned to zero after the pulse and remains there until the next bit is recorded. The second set of waveforms on this drawing illustrates the remanent flux pattern on the magnetic surface after the write head has passed. There is some distortion in this pattern because of the fringing of flux around the head.

If this pattern of magnetization is passed under a read head, some of the magnetic flux will be coupled into the core of the head. The flux takes the lower reluctance path through the core material of the head instead of bridging the gap in the head (Fig. 6.27). And when the amount of flux through the core material changes, a voltage will be induced in the coil wound around the core. Thus a change in the amplitude of the recorded magnetic field will result in a voltage being induced in the coil on the read head. The waveforms in Fig. 6.35(*a*) and (*b*) illustrate typical output signals on the read-head windings for each technique. Note that the waveform at the read head is not a reproduction of the input current during the write process, nor of the pattern actually magnetized on the magnetic material.

The problem is, therefore, to distinguish a 1 or a 0 output at the sense winding. Several techniques have been used. One consists in first amplifying the output waveform from the read winding in a linear amplifier. Then the output of this amplifier is strobed to determine whether a 1 or 0 was written. If the output from the read amplifier is connected to an AND gate and the strobe pulse is also connected as an input to the same AND gate, then the output will be a positive pulse when the recorded signal represents a 1.

It is important that the time pulse be very sharp and occur at the right time relative to the reading and writing of the bits.

A fundamental characteristic of RZ recording [Fig. 6.35(*a*)] is this: For a 1, the output signal during the first half of each bit time will be positive with regard to the second half; for a 0, the first half of the output signal during each bit time

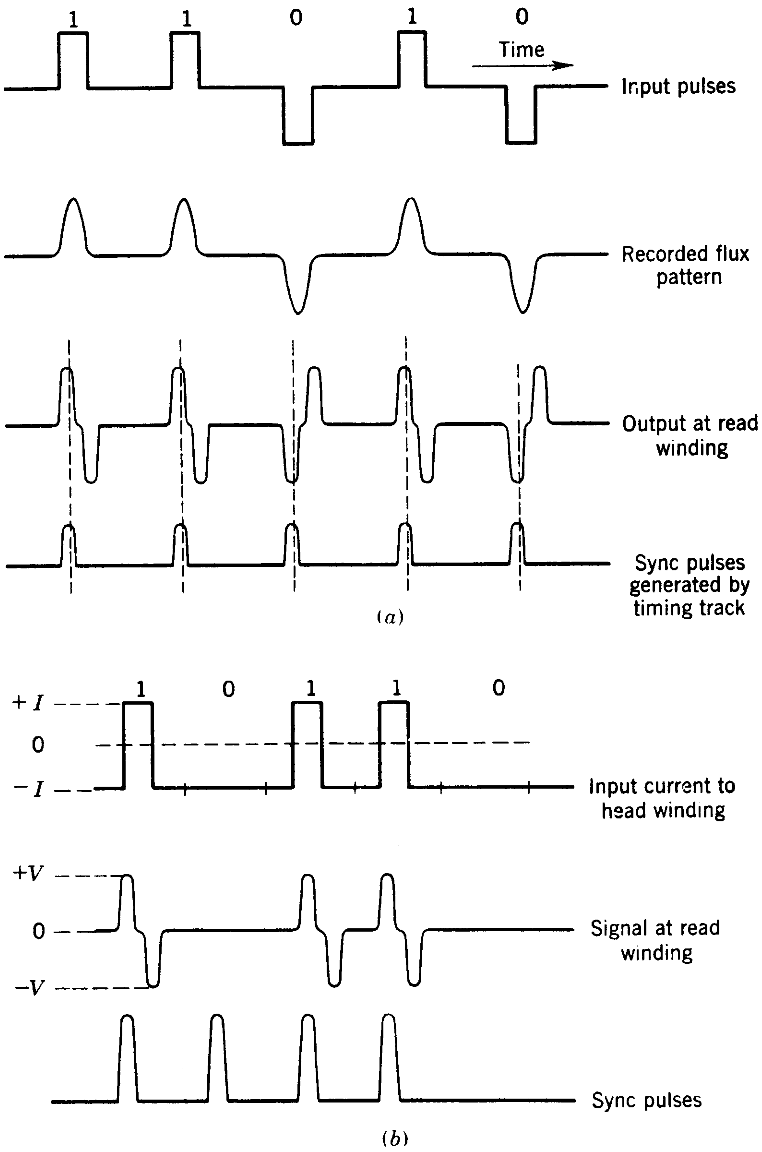


FIGURE 6.35

Recording techniques. (*a*) Return-to-zero recording. (*b*) Return-to-bias technique.

will be negative with regard to the second half of the signal. This fact is sometimes exploited in translating the signal read-back.

In the RZ system in Fig. 6.35(*a*), the magnetic field returns to zero flux when a 1 or a 0 pulse is not present. This makes it impossible to write over information which has previously been written unless the position of each cell is very accurately located. If a 0 pulse is written directly over a previously recorded 1, the flux generated will reverse the polarity of the recorded field only if the write head is in exactly the right position when the 0 is recorded. So the timing of the writing of information is very critical for this system, and it is rarely used.

The second method for recording information is the *return-to-bias system*, illustrated in Fig. 6.35(*b*). In this case, the current through the winding maintains the head saturated in the negative direction unless a 1 is to be written. When a 1 is written, a pulse of current in the opposite direction is applied to the winding at the center of the bit time. The outputs at the sense winding are also illustrated in the figure. In this case, there will be an output at the sense winding only when a 1 is written. This output may be amplified and strobed just as in the previous case. The timing here is not so critical when information is being written over, because the negative flux from the head will magnetize the surface in the correct direction, regardless of what was previously recorded. The current through the winding in this case, and in all those which follow, is assumed to be sufficient to saturate the material on which the signals are being recorded. A primary problem here concerns sequences of 0s. For magnetic tape, either a clock track must be used or the code used must be such that at least one 1 occurs in each line of the tape. Notice that this is because only 1s generate magnetic flux changes and, therefore, output signals at the read head.

NONRETURN-TO-ZERO RECORDING TECHNIQUES

***6.17** Figure 6.36 illustrates three recording techniques, each of which is classified as a nonreturn-to-zero system. In the first, the current through the winding is negative through the entire bit time when a 0 is recorded and is positive through the entire bit time when a 1 is recorded. So the current through the winding will remain constant when a sequence of 0s or 1s is being written and will change only when a 0 is followed by a 1 or when a 1 followed by a 0 is written. In this case, a signal will be induced in the sense winding only when the information recorded changes from a 1 to a 0, or vice versa.

The second technique illustrated is sometimes referred to as a *modified nonreturn-to-zero*, or *nonreturn-to-zero mark* (NRZI), technique. In this system the polarity of the current through the write winding is reversed each time a 1 is recorded and remains constant when a 0 is recorded. If a series of 1s is recorded, then the polarity of the recorded flux will change for each 1. If a series of 0s is recorded, no changes will occur. Notice that the polarity has no meaning in this system; only changes in polarity matter. Therefore, a signal will be read back only when a 1 has been recorded. This system is often used for tape recording when, in order to generate a clock or strobe, a 1 must be recorded somewhere in each cell along the tape width. That is, if 10 tracks are recorded along the tape, then one of these must be a timing track which records a sequence of 1s, each of which defines a different set of cells to be read; or the information must be coded so that a 1 occurs in each set of 10 cells which are read. Alphanumeric coded information[18] is often recorded on tape, and the code may be arranged so that a 1 occurs in each code group.

The third nonreturn-to-zero technique in Fig. 6.36 is sometimes called a *phase encoded, biphase-mark, Harvard, Manchester*, or *split-frequency* system. In this case a 0 is recorded as a $\frac{1}{2}$-bit-time negative pulse followed by a $\frac{1}{2}$-bit-time

[18]Alphanumeric codes are described in Chap. 7.

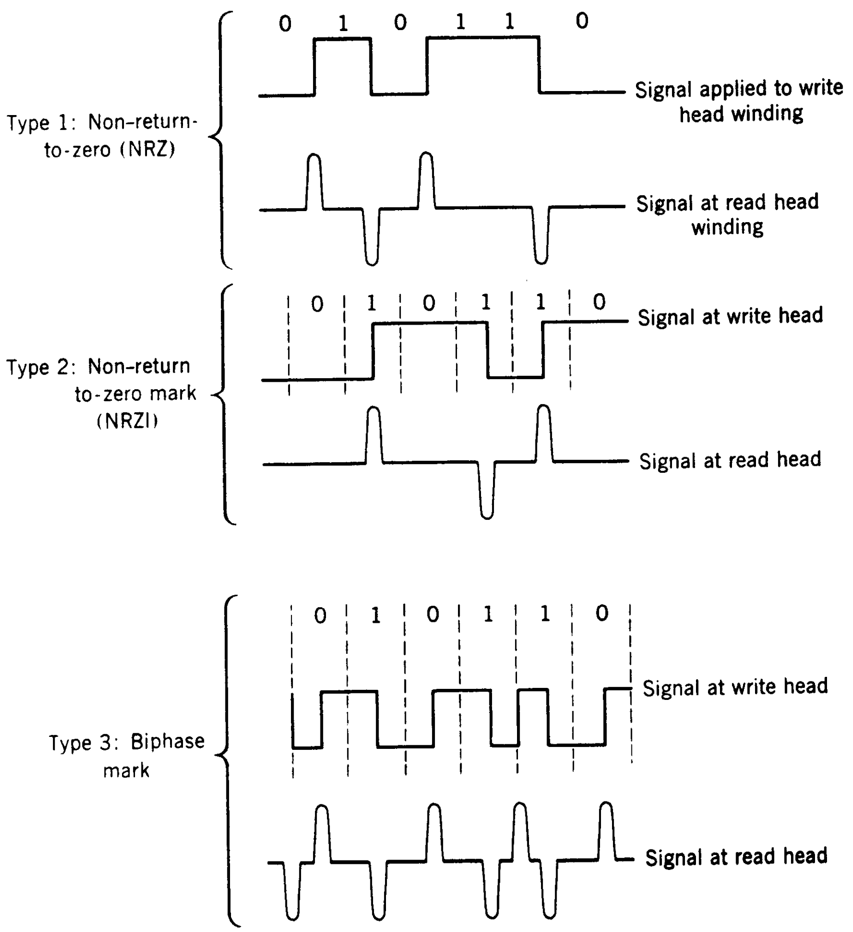


FIGURE 6.36

Three types of non-return-to-zero recording.

positive pulse, and a 1 is recorded as a $\frac{1}{2}$-bit-time positive pulse followed by a $\frac{1}{2}$-bit-time negative pulse. This technique is often used in high-speed systems.

The reading of information which has been recorded consists of two steps. First the output from the read head is amplified, and then the amplified signals are translated by logic circuitry. Figure 6.37 shows a translation technique for the first nonreturn-to-zero system illustrated in Fig. 6.36. The output signals may be either from the output flip-flop or from serial pulses. The sync pulses occur each time a cell passes under the read heads in the system.

The flip-flop (Fig. 6.37) responds to positive pulses only. Positive-pulse signals at the recording head will therefore "set" the flip-flop to 1. The inverter at the *c* input will cause negative pulses to be made positive. These positive pulses then will clear the flip-flop. The output of the flip-flop may be used directly by the computer; or pulse outputs can be generated by connecting an AND gate to the 1 output, delaying the sync pulses, and connecting them to the AND gate. Also, a serial representation of the number stored along the surface may be formed.

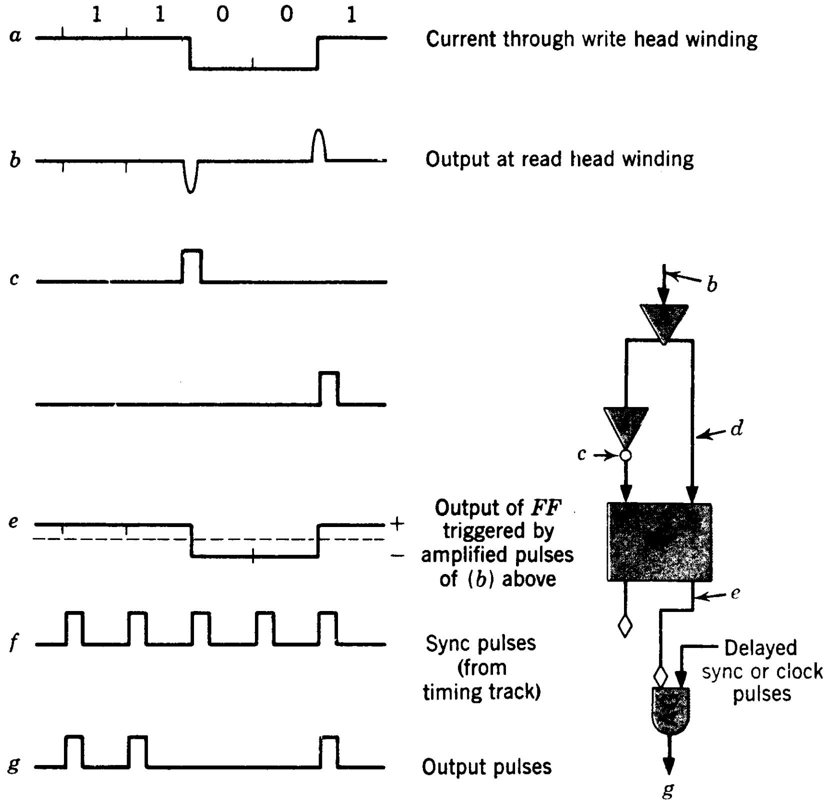


FIGURE 6.37

Nonreturn-to-zero recording.

SUMMARY

6.18 The memory devices in a computer are organized according to their speed. The fastest are in the central processor, the next fastest are in the main memory, and the slower but less expensive devices are used as backup, or secondary, memories.

IC memories are generally organized by some variation of a two-dimensional selection scheme. They can use static (flip-flop) or dynamic memory cells. Large IC memories can be formed from a number of IC chips by using decoders and select inputs to organize the selection process. IC memories can be readily connected to the buses now used to interface memory drives to a central processing unit, and examples of this were shown.

Read-only memories are similar to read-write memories in organization but provide permanent or semipermanent storage of their contents Some ROMs have these contents built in by the manufacturers while others can be changed by the user as desired.

Disk memories have many desirable properties. Fixed-disk Winchester memories are now very popular as are changeable floppy-disk memories. The principles, operating characteristics, and organization of these memories were discussed.

Magnetic-tape memories provide low cost per bit with longer access times than disk memories. Tape memories range from small, low-cost, entertainment-type cassette memories to large, high-speed tape drives with large reels of tape. New cassette-type streaming tape drives are often used to back up Winchester disks. The operating characteristics and recording formats for tape memories were presented along with costs and their comparison with disk memories.

QUESTIONS

6.1 Determine the number of AND gates and OR gates used in a two-dimensional and a one-dimensional IC memory by using the techniques in Figs. 6.4 and 6.9. The memory is to have 1 bit per word and 16 words. (Show how you got your numbers.)

6.2 Determine the complete OR gate and AND gate decoder count for an IC memory with 4096 words of 1 bit each, using the selection schemes in Figs. 6.4 and 6.9. (Show how your numbers are derived.)

6.3 The interface circuitry for dynamic memories is more complicated than for static memories. However, the cost per bit of actual memory is less for dynamic memories. As a result, for small memories static devices are less expensive and for larger memories dynamic devices are less expensive. If the interface for a static memory costs $1.00 and for a dynamic memory $10.00, and static memory bits cost $0.005 per bit while the dynamic memory bits cost $0.002 per bit, determine how many bits must be in a memory before the dynamic memory is less expensive.

6.4 Consider decoder matrices which are rectangular, but not square. For instance, to encode a 256-bit memory, we might use a "square" 16×16 matrix in two-dimensional form or an 8×32 two-dimensional rectangular array. Show that keeping the array "as square as possible" will reduce the number of AND gates in the decoders.

6.5 What are the sizes of the decoders in Fig. 6.18?

6.6 As the size of the memory goes up, the advantage of using a two-dimensional selection scheme increases with regard to the number of AND gates used for the decoders. The two-dimensional memory, however, requires more complicated memory cells. For a 4096-bit memory with a single output bit, compare the number of AND gates in the decoder for linear and for two-dimensional memories and also the number of gates in the linear memory cells versus the two-dimensional memory cells. Try to draw some conclusion regarding which is more economical.

6.7 Does the two-dimensional selection system in any way slow down operation compared to the linear selection scheme?

6.8 A 64K-word memory is to be assembled with chips having 4096 bits each, using the chip in Figs. 6.18 and 6.19. Explain how a memory bus can be connected so that the full 64K-word memory can be implemented (showing only 1 bit in the memory word).

6.9 There is a scheme whereby as the power declines, the contents of the IC memory are read into a secondary (disk or tape) memory. When the power is returned, the data are reentered into the IC memory. It is important that the power not decline too much before the transfer of information can be made. As a result, it is generally necessary to have some sort of backup power to carry the memory through until the contents of the IC memory can be read out. Calculate how long this would take for a 256K memory with a cycle time of 500 ns.

6.10 Question 6.9 concerned dumping an IC memory on a disk in case of power failure. It is also necessary to find a spare area on the disk and to transfer the contents of the memory onto the disk. For one of the disk memories described in the chapter, discuss this problem for the preceding 256K 500-ns IC memory.

6.11 Several microcomputers come with a basic 256K-word 8-bit memory. How many flip-flops are in (*a*) the memory address register and (*b*) the memory buffer register?

6.12 Draw the waveforms for recording the binary sequence 101, showing the signal applied to the write-head winding and the signal at the read-head winding for the type 1, type 2, and type 3 nonreturn-to-zero recording techniques.

6.13 A block diagram of an INTEL MOS LSI memory chip is shown in Fig. 6.10. Lines A_1 through A_8 are for the 256 addresses, and R/W tells whether to read or to write (a 0 on this line is READ, a 1 is WRITE). The CHIP SELECT disables the memory for a 1 input and enables the memory with a 0 output. Discuss the construction of a 256-word 8-bit memory using these packages. How many packages are required? Each flip-flop in the address register (external) would have its 1 output connected to how many chips?

6.14 Draw the schematic for a many-to-one decoder matrix with inputs from 4 flip-flops and 16 output lines. Use the same basic configuration as is illustrated in Fig. 6.5(*a*).

6.15 The INTEL package in Fig. 6.10 is representative of several IC manufacturers' products. Show how the CHIP SELECT can be used to add words to a memory.

6.16 What are the primary advantages of bipolar memory over MOS memory?

6.17 Explain the operation of the digit line in a linear-select core memory.

6.18 Explain the difference between a dynamic and a static MOS memory.

6.19 Contrast the parallel tree and balanced decoder networks for a 32-output decoder. Figure the number of diodes used for each and the delay incurred because of the number of gates a signal must pass through for each, assuming diode AND gates are used.

6.20 Given eight 8-bit registers $A, B, C, \ldots, H$, show how a transfer circuit can be made by using multiplexers so that the contents of any register can be selected by a 3-bit register S and transferred to an 8-bit register X.

6.21 In larger machines, when the ac power drops below a certain level, the contents of the control unit and arithmetic-logic unit are dumped on tape or disks so that the computer can be restarted with no loss of data. If IC memories are used, their contents must be dumped also. Explain why.

6.22 Explain the operation of the memory cell in Fig. 6.3.

6.23 Explain the operation of the memory cell in Fig. 6.8.

6.24 When linear selection is used for IC memories, individual cells tend to be simpler than for two-dimensional cell select systems, but the decoders tend to be more complicated. Explain why.

6.25 Why are floppy disks more used than hard disks in microcomputers?

6.26 The memory in Fig. 6.19 uses the organization in Fig. 6.11 and has linear-select memory cells. This simplifies individual memory cell complexity while increasing sense and write circuitry. Explain why this is cost-effective for a memory of this type.

6.27 Dynamic memories that require external refreshing introduce extra complexity into computer operation. Why?

6.28 Explain the advantages and disadvantages of dynamic IC memories.

6.29 In designing a 256-word 8-bit memory, pin 1 is connected to pin 1 for each container of the chip shown in Fig. 6.10. This applies to all *A* inputs and to chip-select bits and R/W, but not to data out or data in. Why?

6.30 Show how to expand the 256-word 1-bit memory in Fig. 6.10 to a 2048-word 1-bit memory by using the CHIP SELECT and a three-input, eight-output decoder.

6.31 Which of the tape units described will transfer data fastest, and how fast can characters and bits be read or written?

6.32 Make up a formula to calculate how many bits a second can be read from a disk memory that revolves at a rate of r revolutions per second and has b bits per track. What will the average latency time be?

6.33 Assuming that a disk with 1500 bits/in. rotates at a speed of 2400 rpm and that we read from eight tracks simultaneously, how many bits per second can be read from one of these disks?

6.34 Explain seek time and latency for disk memories.

6.35 A magnetic-tape system has seven tracks for each $\frac{1}{2}$-in. width of tape. The packing density per track is 1280 bits/in., and the tape is moved at a speed of 75 in./s. If the tape width is 0.5 in., how many bits may be read per second?

6.36 Fixed-head disk memories reduce total access time by avoiding either seek time or latency. Which is avoided and why?

6.37 Contrast the floppy-disk figures for the 5- and the 8-in. systems.

6.38 Tape cassettes and tape cartridges have advantages and disadvantages compared to conventional tape systems. Cite several of each.

6.39 The table comparing storage media brings out some of the contrasts in price and performance for memory systems. What devices would you choose for the following kinds of computers and why?

(*a*) Microcomputers (*b*) Minicomputers (*c*) Large computer systems

6.40 Draw a diode ROM as in Fig. 6.24 which translates from excess-3 code to BCD.

6.41 An IC memory could be made three-dimensional by breaking the MAR into three pieces and having three decoders. The memory cell would be more complex, however. Design a 4096-word memory of this type, comparing it with the two-dimensional memory.

6.42 Design four locations in a ROM by adding gates to the following diagram. The signal C_1 is to be a 1 in locations 01111 and 11110 and a 0 otherwise; the signal C_2 is to be a 1 in locations 01111 and 10000 and a 0 otherwise; the signal C_3 is to be a 1 in 01101, 11110, and 01111 and a 0 otherwise. A given decoder output line is high when selected.

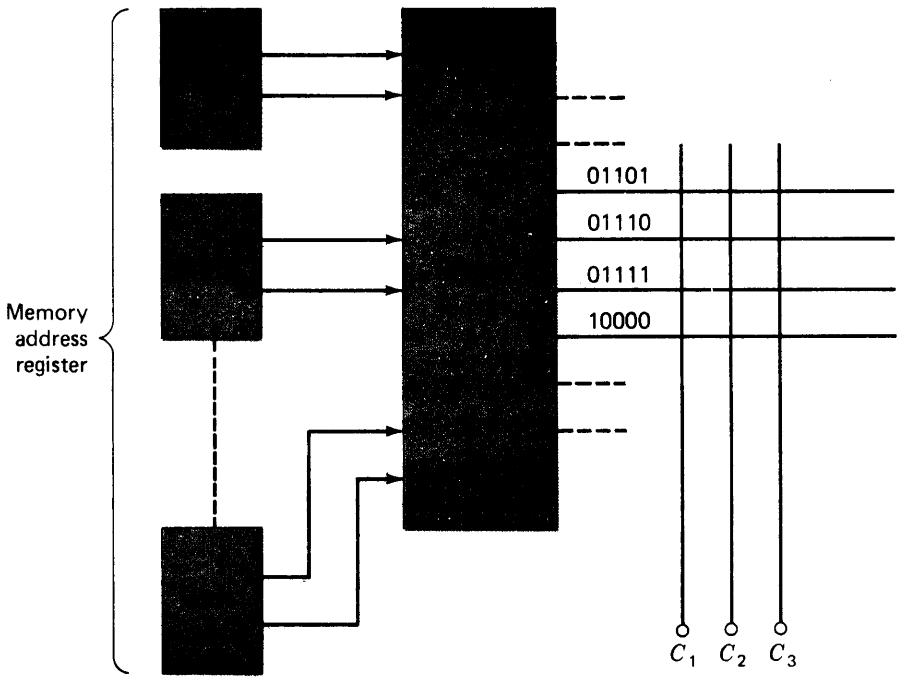


FIGURE 6.42

6.43 Here are some data on a PDP-11 disk pack:

Number of cylinders: 203
Tracks per cylinder: 2
Bytes per track: 6144
Disk rotation time: 40 ms

Seek time S to move N cylinders
For $N = 0$ to 8, $S = 6 + 2N$
For $N = 9$ to 24, $S = 16 + 3N/4$
For $N > 24$, $S = 26 + N/3$

Discuss the total search time for finding a specific piece of data, considering the heads to be positioned in the center cylinder when the search order is given. A cylinder is the set of all tracks which can be read from or written on for a given position of the head-positioning mechanism.

6.44 Compare a conventional disk memory with the floppy-disk memory with regard to operating characteristics and costs.

6.45 Memory systems, disk packs, tape drives, cassettes, and floppy disks all have different characteristics. However, in general the units with lower entry prices (that is, lower unit prices) have higher bit prices. Explain this and give examples, using figures in the text.

6.46 *Hard sectoring* refers to a disk system in which sectors are determined by some mechanical technique. For instance, sectors on some floppy disks are determined by punching a number of holes around the disk, and a sector begins when a hole occurs in the disk. *Soft sectoring* refers to a technique in which headers are written at the beginning of sectors, and so the reading circuitry locates sectors and information without the use of mechanical devices. Whta are the advantages and disadvantages of these systems?

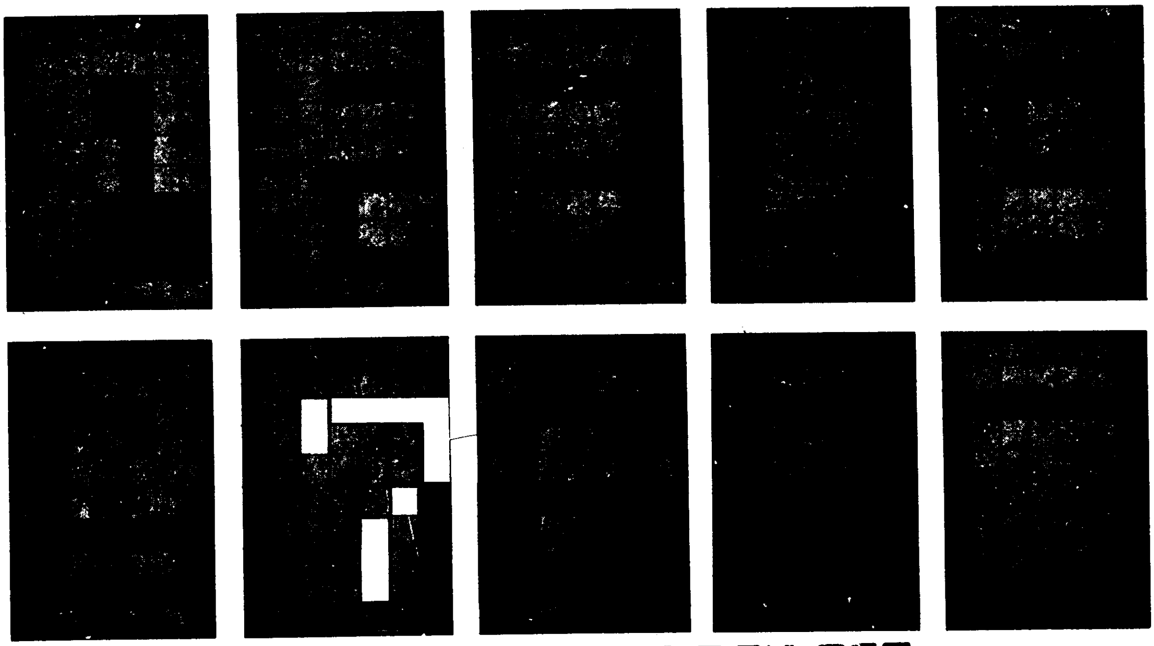
6.47 The very high-speed high-bit-packing-density tape drives use an encoding technique in which bits are encoded in groups. For instance, in one commonly used technique 4 input bits are encoded into 5 bits. Since the 16 possible combinations of the 4 bits which can occur in the data are mapped into only 16 of the 32 possible combinations of 5 bits, these sixteen 5-bit patterns can be carefully selected so that the recording characteristics are optimal. When the 5 bits are read back, they are changed back to the original input data. This plus the use of powerful error-correcting codes allows a packing density of 6250 bits/in. (and sometimes more). Give some advantages and disadvantages of a complicated encoding and decoding scheme such as this one with regard to tape drive mechanisms and user characteristics.

6.48 Formulate a memory system for a microprocessor which has 64K of ROM, 256K of RAM, and an initial backup memory of 0.5 Mbyte with a possibility of expanding to 5 Mbytes. Choose the memory devices you think would be reasonable and justify your choice economically and from a performance viewpoint.

6.49 Bubble and CCD memories are generally considered to be competitive with floppy disk and small disk packs and are useful in replacing these devices, because disks require mechanical motion for reading and writing and are thus less reliable than straight integrated circuitry or nonmoving media. In what applications might bubble or CCD memories be particularly useful?

6.50 A dynamic memory must be given a refresh cycle 128 times each 2 ms. Each refresh operation requires 150 ns, and a memory cycle is 250 ns. What percentage of the memory's total operating time must be given to memory refreshes?

6.51 If a memory is made up of dynamic memory chips, then the time lost to refreshing can be reduced by breaking the memory into sections, or banks, and when one bank is addressed, refreshing another bank. Sketch how this might be arranged by a memory controller chip.



INPUT-OUTPUT DEVICES

In order to use a computer, it is first necessary to insert the program and the data via the input devices. The input devices to a personal computer consist of keys on a keyboard. Numeric and alphabetic data are introduced via these keys. Figure 7.1 shows a keyboard for a personal computer.

When the program and data have been read and the necessary operations completed, it is necessary for the computer to deliver the results. A great variety of output devices are available. Business applications generally require that the results be printed in tabular form or perhaps on a series of checks, as in a payroll accounting operation. Scientific results are more likely to consist of numeric data which must be clearly printed with little chance of error (such accuracy is also of prime importance in computers used to compute payroll checks) or graphs showing the results of the calculations. For any of these applications, one type of output device may be more desirable than another. However, a great many applications require that the outputs from the computer be printed on a piece of paper. This requires printers, ranging from electromechanical typewriters, which print one letter or digit at a time, to high-speed printers capable of printing a hundred or more characters at a time. Figure 7.1 shows a small printer and an oscilloscope display, the two output devices used most. There are several important input-output devices in addition to keyboards, printers, and oscilloscopes, however, and these are introduced in this chapter.

This chapter is organized as follows. First, input devices are discussed, followed by output devices. Then digital-to-analog (D-to-A) and analog-to-digital (A-to-D) converters are described, followed by a section on data collection.

FIGURE 7.1

Personal computer.

OBJECTIVES

1 Input-output devices provide for communication between computers and humans, computers and computers, and computers and external sensors and effectors (robots, for example). The various devices such as tape, cards, keyboards, oscilloscopes, and printers are discussed along with their characteristics.

2 Alphanumeric codes provide for machine interfaces with humans where keyboards and printers can be used to input and output information. Error-detecting codes provide for protection against errors. Both these subjects are discussed, and examples are given.

3 Digital-to-analog and analog-to-digital converters provide important interfaces between the outside world and computers. The principles of operation of these devices, examples, and general characteristics are presented along with a discussion of data collection from analog inputs.

PUNCHED TAPE

7.1 Punched tape was one of the first popular media for storing the programs and data to be read into a digital machine. When the first large computers were designed, telegraph systems had been using perforated paper tapes for some time, and so devices for punching and reading paper tapes already had been fairly well developed. The tape comes in many types and sizes. A medium-thickness tape has been used a great deal, and oiled tapes and plastic tapes are employed also. The widths of the tapes have varied from $\frac{1}{2}$ to 3 in. Information is punched into the

PUNCHED TAPE

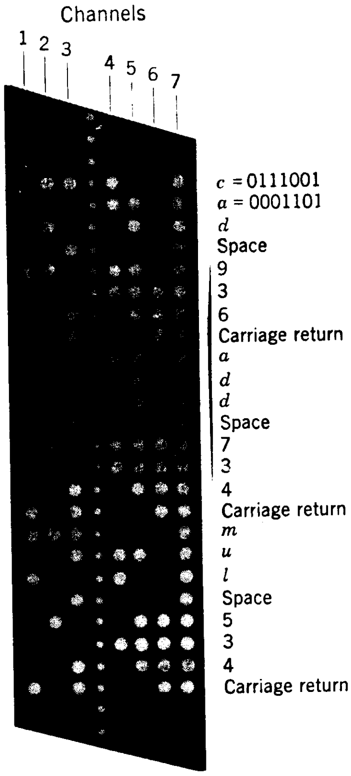


FIGURE 7.2

Punched paper tape.

tape a line at a time. Figure 7.2 illustrates a section of a punched tape. Multiple channels are used (just as on magnetic tape, a channel runs lengthwise along the tape), and a single character or code is punched as a pattern of bits in each lateral line.

The preparation of these paper input tapes is sometimes referred to as *keyboarding*. In this step the operator of a tape punch is presented with a copy of a program or input data. The operator then punches a number of holes into the tape. The holes represent, in coded form, the input information to the machine. The tape punch may be one of a number of types. The more popular devices resemble a typewriter, and the keyboards of these tape punches contain conventional symbols, similar to those on an ordinary typewriter.

When a key on the tape-punch keyboard is depressed, the binary-coded symbol for the character selected is punched into the tape, and then the tape advances to the next line. In most cases, the tape punch also prints on a separate piece of paper, in the same manner as a typewriter, the character that was punched, as well as printing the characters along the rows of the tape. There is then a typewritten copy of the program, which may be checked for errors, in addition to the paper tape punched with the coded symbols. This printed copy of the program is referred to as the *hard copy*. Many tape punches are able to read a perforated tape and to type printed copy from this tape. A punched section of tape may be placed in the tape reader attached to the tape punch, and a typed copy of the information which was punched in the tape may be made.

Figure 7.3 shows a code which has been used frequently for paper tape systems. In the eight-channel code shown in Fig. 7.3, eight channels run lengthwise along the tape. A hole in a given one of these channels represents a 1, and the absence of a hole a 0. The 1, 2, 4, and 8 channels are used to represent the digits for 0 to 9. Thus 0s, or no holes, in positions EL, X, and 0 indicate that the encoded character is a digit with the value given by the sum of the positions 8, 4, 2, 1, in which there are holes. The check position is used for an *odd-parity check*. Its value is determined so that the number of 1s, or holes, in each character is odd. The 0 and X positions are used in conjunction with the 8, 4, 2, 1 positions to encode alphabetic and special characters.

The code has two special features. A punch in the end-of-line position indicates the end of a record on the tape. When there are punches in all seven of the positions, this indicates a blank character called *tape feed,* and the tape reader skips over such positions. This is useful for correcting mistakes in keyboarding or in editing tapes, because an erroneous character can be eliminated by simply punching in the nonpunched positions.

TAPE READERS

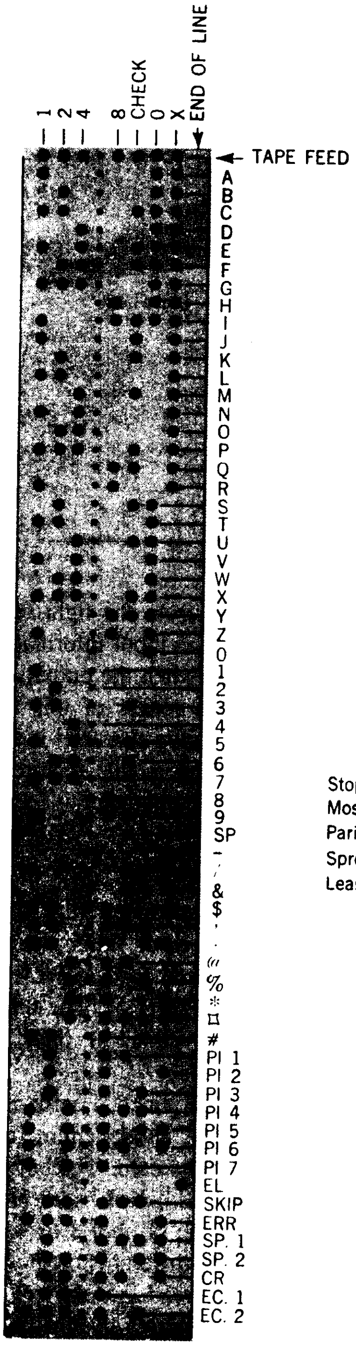
7.2 The function of the paper tape reader is to sense the coded information punched in the tape and deliver this information to the computer. Most tape readers used in teletype and office equipment are electromechanical devices. In many of these devices, mechanical *sensing pins* are used to determine the symbol punched ino each line of the tape. In a system of this type, there will be a sensing pin for each information channel, plus a means of moving the tape and positioning it for reading. The tape is not moved continuously, but only a single line at a time; is stopped while the coding is sensed; and then is moved to the next line. The motion of the sensing pins operates a switch, the contacts of which are opened or closed, depending on whether there is a hole in the tape. Another type of reader uses a *star wheel* to sense the absence or presence of holes in the tape, as shown in Fig. 7.4.

When the input is to a digital machine, the motion of the tape through the reader generally is controlled by the computer. Each time the tape is to be advanced and a new character read, the computer will supply the reader with a pulse which will cause it to advance the tape to the next character. To read characters as fast as possible, generally a line will be read at the same time as the advancing pulse is transmitted. Since there will be a delay owing to inertia before the tape is actually moved, the reading of the state of the sensing relays will occur during this delay period. In this case, when a stop character is sensed, the reader will proceed to the next character before actually stopping.

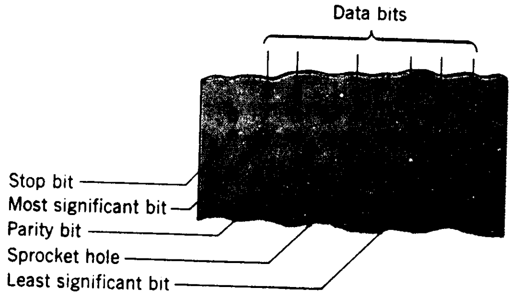
To speed up the reading process, tape readers use photoelectric cells or photodiodes to read the characters punched into the tape. In this case, a light-sensitive cell is placed under each channel of the tape, including the tape feed hole, or sprocket channel. A light source is placed above the tape, so that the light-sensitive element beneath the hole in the tape will be energized and will produce

7

TAPE READERS



(*a*)



(*b*)

FIGURE 7.3

(*a*) Punched tape with eight-channel code. (*b*) Format for eight-channel tape.

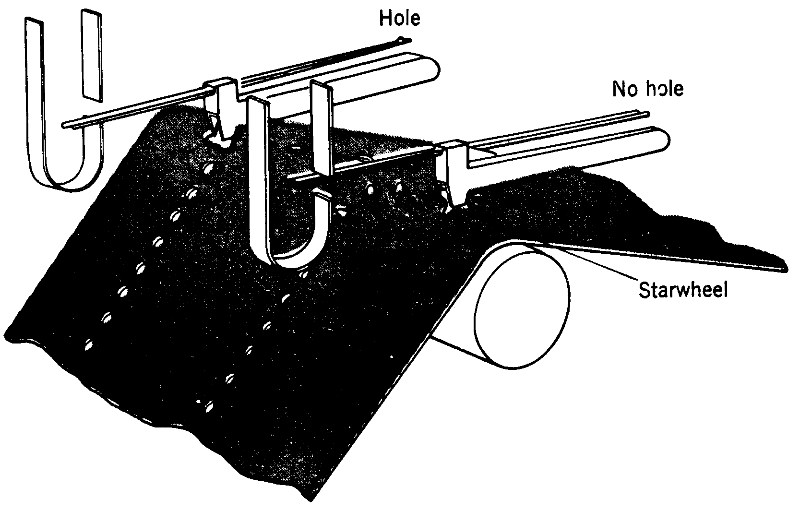


FIGURE 7.4

Star-wheel mechanism for reading perforated tape.

a signal indicating the presence of the hole. The signals from the light-sensitive elements are then amplified and supplied to the computer as input information.

The tape feed hole is used in this case to determine when the outputs of the light-sensitive elements are to be sensed. The tape in a reader of this type is generally friction-driven and is moved continuously until a stop character is sensed. Extremely fast starting and braking of the tape are very desirable features, and most readers are capable of stopping the tape on any given character.

PUNCHED CARDS

7.3 A widely used input medium has been the punched card. While there are a number of sizes of punched cards, the most frequently used card at present is a 12-row 80-column card $3\frac{1}{4}$ in. wide and $7\frac{3}{8}$ in. long (see Fig. 7.5). The thickness of the cards varies, although at one time most cards were 0.0067 in. thick. There is now a tendency to make the card somewhat thinner.

Just as with tape, there are numerous ways in which punched cards may be coded. The most frequently used code is the *Hollerith code,* an alphanumeric code in which a single character is punched in each column of the card. The basic code is illustrated in Fig. 7.5. As an example, the symbol A is coded by means of a punch in the top row and in the 1 row of the card, and the symbol 8 by a punch in the 8 row of the card. There are other types of cards with different hole positions, just as there are many ways of preparing the cards to be read into the computer. The most common technique is very similar to that for preparing punched tape, in that a card-punch machine with a keyboard like that of a typewriter is used, as shown in Fig. 1.8. The card punch usually also makes a hard copy of the program as it is punched into the cards. Generally, the card punch also prints the characters punched into a card on the face of the card itself. In this way, a card may be identified without examining the punches. Each character is usually printed at the top of the card directly above the column in which the character is punched.

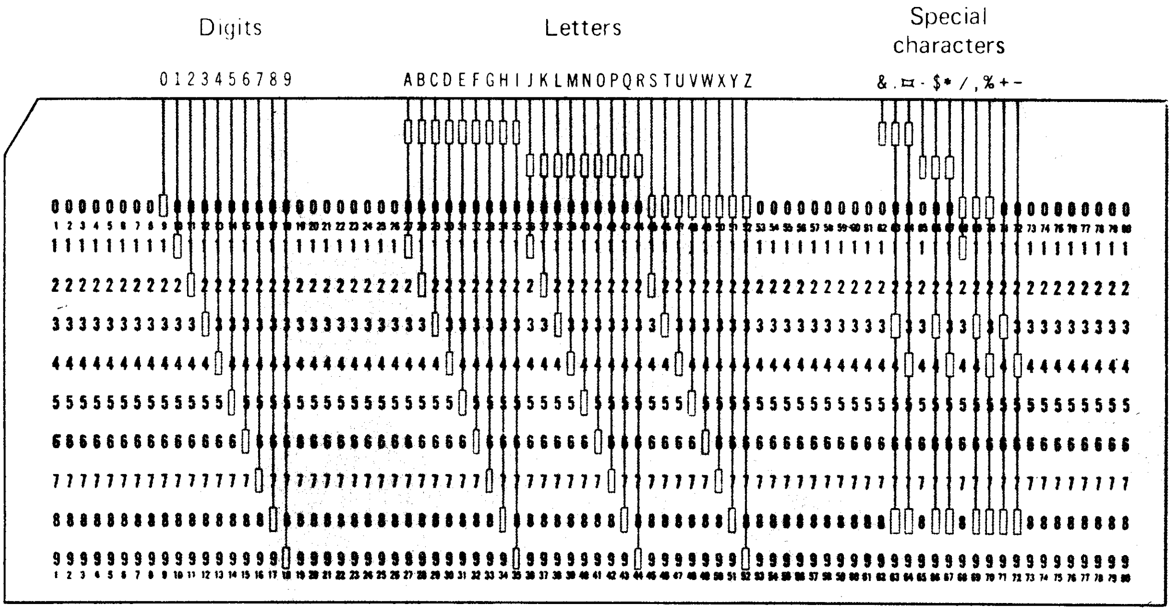


FIGURE 7.5

Punched card with Hollerith code.

The card punch contains a hopper in which the blank cards are stacked. The operator of the card punch then causes a card to enter the punching area, and the program's list of instructions or the data to be processed are punched into the card. The card punch punches the card laterally, a column at a time, starting at the left. If a key of the card punch is depressed, the code for the character is punched into a column of the card, and then the card is moved so that the next column on the right is under the punch.

Figure 7.6 shows IBM card punches, widely used devices. These have a small memory capable of holding all the data that can be punched into two 80-column records and six program cards. The key-punch operator keyboards the data into the small memory and can backspace and change characters until the data keyboarded are correct. After all the data are in the small memory, an *enter data* key is depressed, and the card is punched from the characters in the memory. Facilities for controlling the format of the card are included by means of IC logic.

CARD READERS

7.4 Most card readers are electromechanical devices which read the information punched into a card, converting the presence or absence of a hole to an electric signal representing a binary 0 or 1. The punched cards are placed in a hopper, and when the command to read is given, a lever pushes a card from the bottom of the stack. Generally, the card is moved lengthwise over a row of 80 *read brushes*. These brushes read the information punched along the bottom row of the card. If a hole is punched in a particular row, a brush makes electric contact through the hole in the card, providing a signal which may be used by the computer. Then the

FIGURE 7.6

IBM 929 card punches.

next row up is read, and this process continues until all rows have been read, after which the next card is moved into position on the brushes.

Faster card readers are constructed by using photoelectric cells under the 12 punch positions along a column and an illuminating source above the card. As each column on the card is passed over the 12 photoelectric cells, whether a given position is punched is determined by the presence or absence of light on the corresponding cell. Card readers operate at speeds of 12 to 1000 cards per minute.

Notice that when cards are read a row at a time, programs must "decipher" the characters punched in the card. Often "binary" decks are used, particularly for programs, where the computer words are distributed along rows rather than in columns and data may be loaded directly as the rows are read.

ALPHANUMERIC CODES

7.5 Data and programs are almost invariably entered in alphanumeric form, and the internal operation of computers, particularly those which involve business records, makes extensive use of alphanumeric codes. Because of the diversity of applications

	000	001	010	011	100	101	110	111
0000	*NULL*	① DC_0	b̸	0	@	*P*		
0001	*SOM*	DC_1	!	1	*A*	*Q*		
0010	*EOA*	DC_2	"	2	*B*	*R*		
0011	*EOM*	DC_3	#	3	*C*	*S*		
0100	*EOT*	DC_4 (Stop)	$	4	*D*	*T*		
0101	*WRU*	*ERR*	%	5	*E*	*U*		
0110	*RU*	*SYNC*	&	6	*F*	*V*		
0111	*BELL*	*LEM*	'	7	*G*	*W*		
1000	FE_0	S_0	(	8	*H*	*X*	Unassigned	
1001	*HT* / *SK*	S_1	)	9	*I*	*Y*		
1010	*LF*	S_2	*	:	*J*	*Z*		
1011	V_{TAB}	S_3	+	;	*K*	[		
1100	*FF*	S_4	(Comma) ,	<	*L*	\		*ACK*
1101	*CR*	S_5	–	=	*M*	]		②
1110	*SO*	S_6	.	>	*N*	↑		*ESC*
1111	*SI*	S_7	/	?	*O*	←		*DEL*

Example: 100 0001 = *A*

b_7 ---------------- b_1

FIGURE 7.7

American Standard Code for Information Interchange.

and the many viewpoints on codes and code construction, many different alphanumeric codes have been suggested and used.

Fortunately, there has been an attempt to standardize on an alphanumeric code which will be agreeable to both manufacturers and users, and the American National Standards Institute has published an American Standard Code for Information Interchange (ASCII). This code is now the most used, and most major manufacturers are using the code so that their equipment will be compatible with that of other manufacturers. This code is shown in Fig. 7.7. Notice that the decimal digits are represented by the normal 8, 4, 2, 1 code preceded by the three binary digits 011. So decimal 1 becomes 0110001, decimal 2 is 0110010, decimal 7 is 0110111, etc. To expand the code, the letter A is 1000001, B is 1000010, etc. There are various codes such as "end of messages," "who are you," "skip," "carriage return," etc., which are very useful in communications systems and in editing data processes in computers. These are shown in Fig. 7.8. As a result, this is the most frequently used code for intercomputer communications systems.

Many IBM computers, and a number of other manufacturers' computer systems, use the extended BCD interchange code (EBCDIC) shown in Fig. 7.9. This code is the second most used code after the ASCII.

The abbreviations used in the figure mean

NULL	Null Idle	*CR*	Carrige return
SOM	Start of message	*SO*	Shift out
EOA	End of address	*SI*	Shift in
EOM	End of message	DC_0	Device control ①
			Reserved for data
			Link escape
EOT	End of transmission	DC_1-DC_3	Device control
WRU	"Who are you ?"	*ERR*	Error
RU	"Are you . . . ?"	*SYNC*	Synchronous idle
BELL	Audible signal	*LEM*	Logical end of media
FE	Format effector	S_0-S_7	Separator (information)
HT	Horizontal tabulation		Word separator (blank, normally non-printing)
SK	Skip (punched card)	*ACK*	Acknowledge
LF	Line feed	②	Unassigned control
V/TAB	Vertical tabulation	*ESC*	Escape
FF	Form feed	*DEL*	Delete Idle

FIGURE 7.8

Special characters in ASCII.

CHARACTER RECOGNITION

7.6 Techniques for data entry extend in many directions. The reading of handwritten or typewritten characters from conventional paper appears to offer an ideal input system for many applications. The systems currently in use are primarily as follows:

1 *Magnetic ink character reading (MICR)* The recording of characters by using an ink with special magnetic properties and with characters having special forms was originally used in quantity by banks. The American Banking Association settled on a type font, and several of their characters are shown in Fig. 7.10. A *magnetic character reader* "reads" these characters by examining their shapes, using a 7 × 10 matrix; it determines, from the response of the segments of the matrix to the magnetic ink, which of the characters has passed under the reader's head. This information is thus transmitted to the system. The determination of the character which is read is greatly facilitated by the careful design of the characters and the use of the magnetic ink.

2 *Optical character reading (OCR)* This area takes one of two forms. In the first, a special type font (or fonts) is used to print on conventional paper with conventional ink. The printed characters are examined by passing them under a strong light and a lens system, which differentiates light (no ink) from inked areas, and a logical system which attempts to determine which of the possible characters is being examined. The systems in actual use depend heavily on the fact that only a limited number of characters in a particular font are used, but such systems are still quite useful. The standard type font agreed on by the ANSI optical character committee is shown in Fig. 7.11.

Of course, the ideal system would be able to adapt to many different type fonts. Some systems, particularly one developed by the post office, even read handwritten characters. The limited success of these systems is due to the many

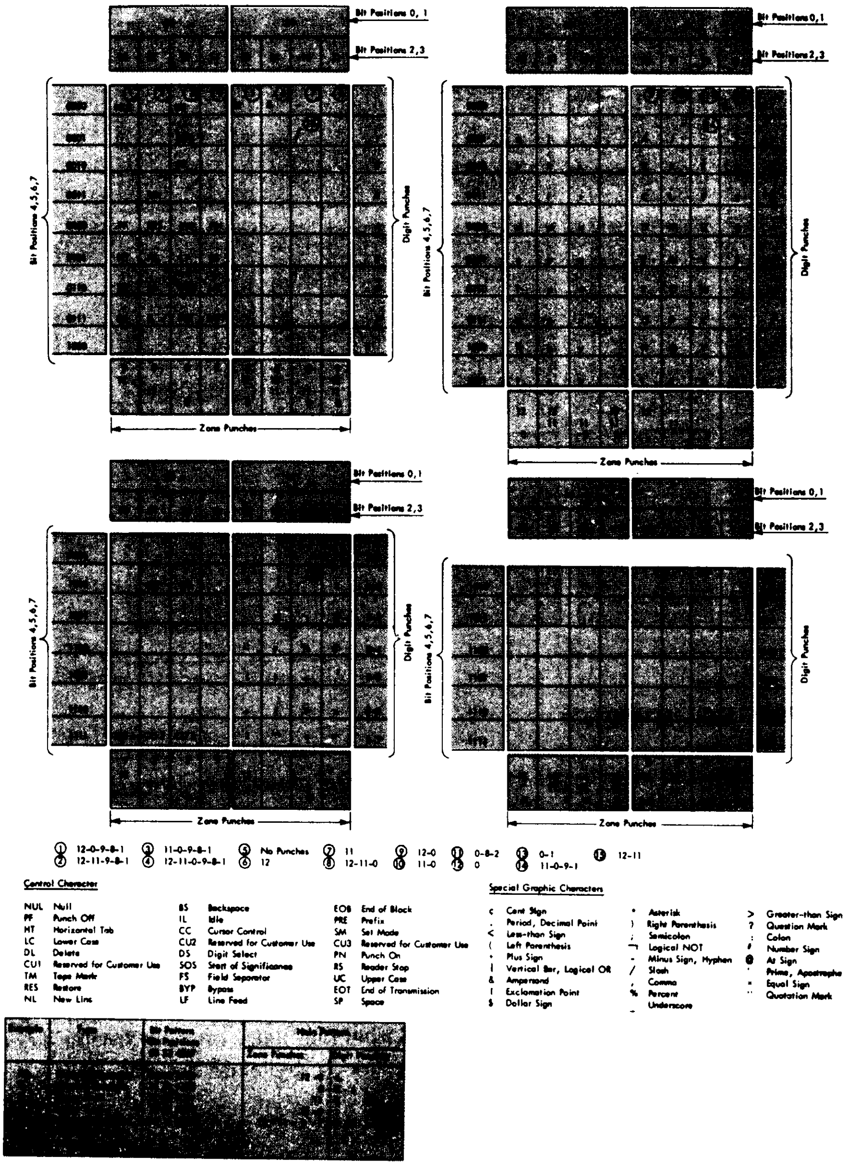


FIGURE 7.9

Extended BCD interchange code (EBCDIC).

shapes that a given character can have. Consider the ways you can write an *a* and the similarity between a handwritten *a* and an *o* or a *b* and an *f*. These problems are increased by the optical reader's difficulty with the porosity of paper, ink smearing at the edges of lines, etc. Much work continues in this area, and much more is needed, but the advantages of such systems continue to cause this work to be sponsored and performed.

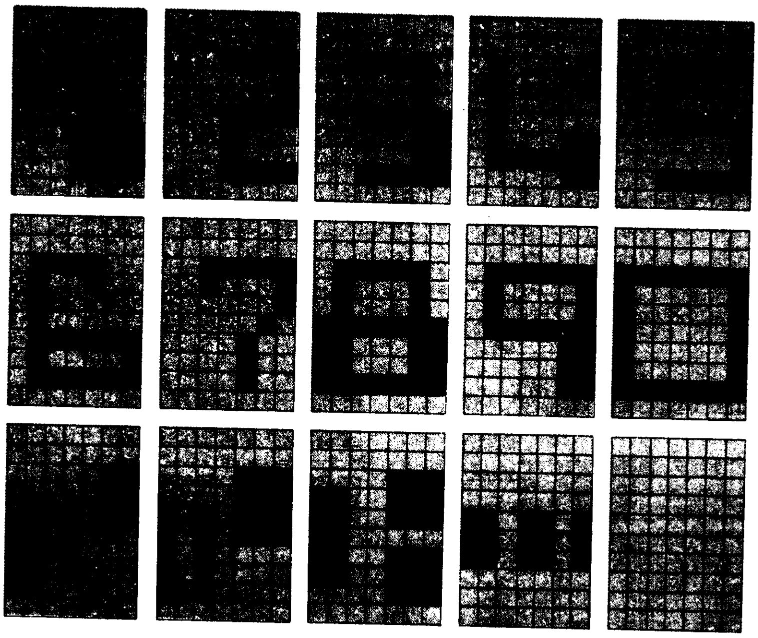


FIGURE 7.10

A magnetic reader character set.

OUTPUT EQUIPMENT

7.7 Although many types of output equipment are used in the computer industry, the most popular form of output from a computer is undoubtedly the printed word, and printers and oscilloscopes are the most used output devices. There are many other output devices, however. For instance, loudspeakers and other audio devices are used to generate music and voices. Banks commonly use computer-driven audio to give balances and other information to tellers.

Lights are sometimes used to indicate the states of the storage devices of the principal registers of the machine (the accumulator, selected in-out registers, etc.). Such lights are generally used as troubleshooting aids, often to troubleshoot the operation of the machine. Figure 7.12 shows the lights on the console of an IBM

FIGURE 7.11

Type font for optical character recognition.

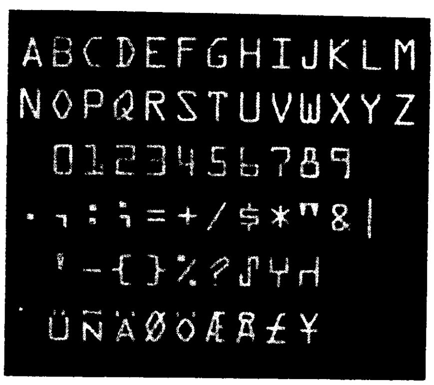


FIGURE 7.12

Console of a large computer system. (*IBM Corp.*)

computer. Each light is connected to a flip-flop output (via a light driver). The outputs of most of the important flip-flops are monitored by using the lights. Various sets of flip-flops can be selected by using the rotary switches at the bottom of the panel, and the toggle switches and pushbuttons can be used to enter data into flip-flop registers and to sequence operations of the computer (as well as to start and stop the computer). The lights can often be driven by the same logic line used in the system.

PRINTERS

7.8 A most convenient and useful method by which the computer can deliver information is by means of printed characters. For the sake of convenience, the printer should have the ability to print alphabetic characters, decimal digits, and common punctuation marks.

The process in printing is the inverse of the encoding procedure in which a key corresponding to an alphanumeric character is depressed, causing a coded binary character to be punched into a tape or card. In this case, coded groups of

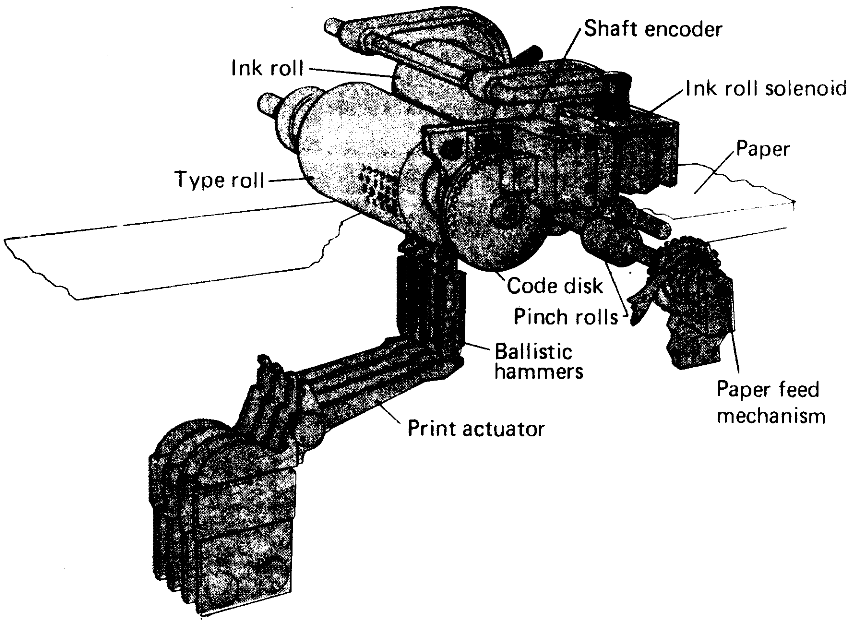


FIGURE 7.13

Line-at-a-time drum printer.

binary bits are delivered to the printer, which decodes them and then prints the correct characters. The basic binary-code groups may contain 7 through 9 bits, depending on the coding for alphanumeric characters that the printer provides.

The information delivered to a printer operated online will be in the form of electronic signals directly from the computer. If the printer is operated offline, the reading and decoding of data stored on punched tape, punched cards, or magnetic tape may be a part of the printing operation. Since the electronic circuitry of a computer is able to operate at speeds much higher than those of mechanical printing devices, it is desirable that a printer operated online be capable of printing at a very high speed. Even if the printer is operated offline, speed is highly desirable, since the volume of material to be printed may be quite large.

An example of a fast printer in which the raised characters are distributed around a *print wheel* that revolves constantly is shown in Fig. 7.13. In this case the print wheel does not contain moving parts, but consists of a motor-driven drum with a number of bands equal to the number of characters printed per line. A set of all the characters which are used is distributed around each band. The print wheel is revolved continuously. When the selected character is in position, the print hammer strikes the ribbon against the paper and thus against the raised character on the print wheel located behind the paper. A printer of this type requires a decoder and a memory for each character position along the line as well as a character-timing encoder for each position, which determines when the selected character is in position. Printers of this type can print up to 1250 lines per minute with 160 characters per line.

In Fig. 7.13 the print wheel is continually revolving, and when a selected character passes by the position where it is to be printed, the print actuator pushes the chosen ballistic hammer against the paper, forcing the paper against the selected character on the print wheel. The print wheel is continually inked by an ink roll, and no ribbon is used. In this system a code disk and shaft encoder are used to tell which character is currently in a position to be printed. Shaft encoders are discussed in a later section. Figure 7.13 shows that the paper is moved horizontally after each line is printed.

Figure 7.14 shows an IBM printer in which the paper is moved vertically in front of a chain of raised characters of type. This chain is continually moving in the horizontal direction so that each of the 48 different characters continually passes by each of the 100 printer's positions in each line. (Other numbers of characters per line are availble.) When a character to be printed passes the position where it is to be printed, the armature hammer magnet is energized, striking a hammer and forcing the paper against the type at that position. An inked ribbon is placed between paper and type so that the character is impressed on the paper in ink.

To provide inexpensive printers for minicomputers and microcomputers, lower-speed "character at a time" printers are used. Consider the print wheel or drum type of printer shown in Fig. 7.13. The drum can be reduced to a small cylinder with only one set of characters. This single rotating cylinder can be moved across the paper along with a single hammer, making an inexpensive character-at-a-time printer [see Fig. 7.15(*c*)]. Sometimes the characters are distributed around a "daisy wheel," as shown in Fig. 7.15(*b*). A hammer then drives the selected character against the paper when it is in position.

The use of pins which are driven against the paper to print characters also provides low-cost printing. Sometimes the pins are arranged in a complete dot matrix, but generally only a single column of seven pins is provided, and this is moved across the paper, requiring five positions per character [see Fig. 7.15(*a*)]. Character generator logic must sequence the striking of the pins as the print head is moved across the paper.

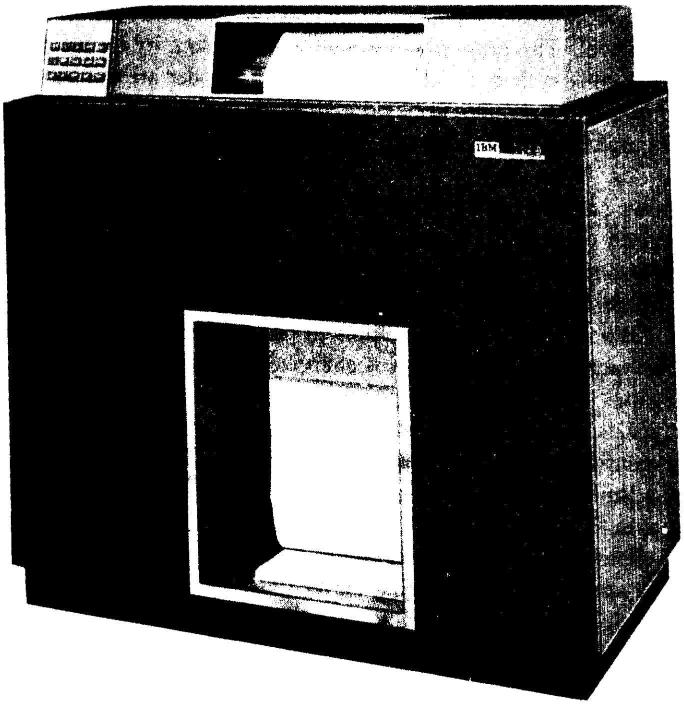
Microprocessors are used to control the timing and other functions required in printers. Character buffering also can be provided by microprocessors.

Many inexpensive printers use thermal or electrostatic papers (see below). The printers are inexpensive; but since the paper is not "ordinary" paper, its costs are generally greater than for regular paper. However, because of the volume of production, the special papers required are quite reasonable in price.

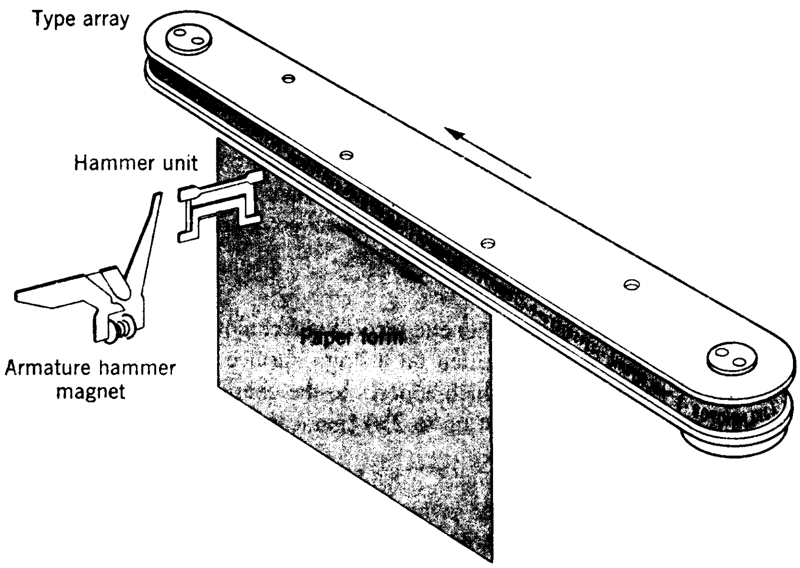
The natural limitations of speed in electromechanical devices and cost considerations have led to the development of printers called *nonimpact printers*. These printers fall primarily in the following categories:

1 *Electromagnetic* By using magnetic recording techniques, a magnetic image of what is to be printed can be written on a drum surface. Then this surface is passed through magnetic powder which adheres to the charged areas. The powder is pressed onto the paper. Speeds of up to 250 characters per second are obtained in such systems.

2 *Electrostatic* For electrostatic printers the paper is coated with a nonconducting dielectric material which holds charges when voltages are applied with writing "nibs" (heads). These heads write dots on the paper as it passes, as shown in Fig. 7.15(*a*). Then the paper passes through a toner which contains material



(a)



(b)

FIGURE 7.14

(*a*) High-speed line printer. (*b*) Type chain for the printer.

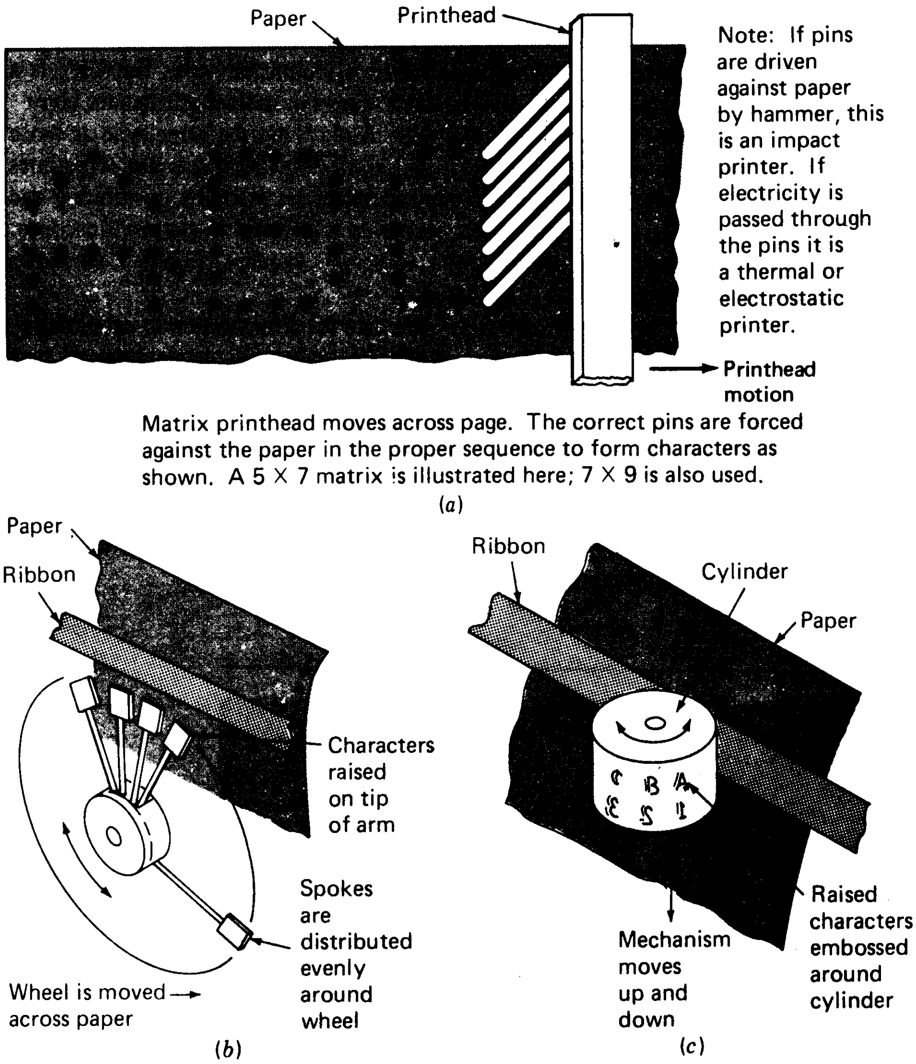


FIGURE 7.15

Impact printer mechanisms. (*a*) Matrix printhead. (*b*) Daisy wheel printer. (*c*) Cylinder printer.

with colored particles carrying an opposite charge to that written by the nibs; as a result, particles adhere to the magnetized areas, forming printed characters.

3 *Thermal printers* An electric pulse can be converted to heat on selected sections of a printing head or on wires or nibs. When this heat is applied to heat-sensitive paper, a character is printed, as shown in Fig. 7.15(*a*).

4 *Ink jets* Some printers direct a high-velocity stream of ink toward the paper. This stream is deflected, generally by passing it through an electrostatic field such as that used to deflect beams in oscilloscopes. In some systems the ink stream is broken into droplets by an ultrasonic transducer.

The wide variety of printers available and the continuing development of new ideas make the printer field a fascinating area. The cost-performance tradeoffs often make selection of a satisfactory printer difficult.

CATHODE-RAY-TUBE OUTPUT DEVICES

7.9 Cathode-ray tubes (CRTs) are often used as output devices. The CRT is a very fast and inexpensive output device, but it does not deliver permanent copy.

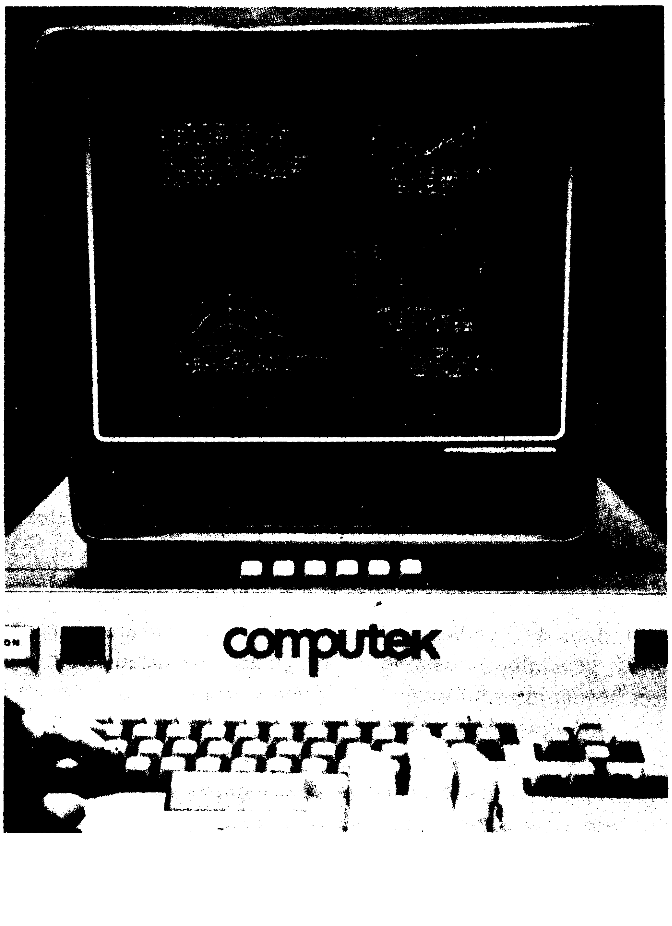
The cathode-ray tubes used in computer displays are the same type as those used in oscilloscopes and television sets, and entire television sets are sometimes used. For these display systems, the displayed points are made by positioning and turning on an electron beam in the tube, just as in television sets or oscilloscopes. The displays are called *CRT, oscilloscope,* or *scope,* displays. Figure 7.16 shows a cathode-ray-tube display with a keyboard.

Many systems have been used to encode data for transmission to oscilloscope displays. The simplest system consists of simply sending the coordinates of each point to be displayed. The oscilloscope electronics then position the beam.

The above system is simple but very inefficient, in that many points must be transmitted, which takes considerable computer time. (In most of these oscilloscope displays, the points must be illuminated at least 30, and generally 60, times each second. There are, however, *storage oscilloscopes,* which hold displayed points and do not require refreshing. These must be erased, however, before rewriting, and are relatively expensive at present.)

FIGURE 7.16

Cathode-ray-tube display. (*Computek, Inc.*)



Because of the problem of repeatedly sending data points to the display device, generally the display-device electronics are independent of the computer and on separate boards which include a memory which stores the data and then generates the display from the last data transmitted, changing or updating the display only when the computer so directs.

Complicated but efficient techniques for data transmission to oscilloscope display electronics are generally used in inexpensive displays. If only alphanumeric data are to be displayed, the ASCII code is used for the transmission of each character to be displayed. The display electronics then converts from each 8-bit coded ASCII character to the sequence of points required to display the character.[1] A memory capable of storing the alphanumeric character codes for the entire oscilloscope face is then included in the display electronics, so that the computer proper must send characters only once, and the oscilloscope electronics refreshes the oscilloscope from the memory. Typically the displays have about 25 lines of 40 to 80 characters per line. Lines for graphic displays such as that shown in Fig. 7.16 are often drawn using a technique in which the computer sends the difference in coordinates between the electronic beam's present location and line segment to be drawn.

Microcomputers generally include display electronics which generate CRT displays using the standard television scanning procedure (or that for television monitors). If only alphanumeric characters are displayed, a memory of about 16K bits (for 25 lines of 80 characters) is required to store the information to be displayed. When a special graphics feature is included so the microcomputer user can make graphics drawings, the display memory must be on the order of 128K to 900K bits (for color).

The individual points in a graphics display are called *pixels*. IBM uses 320 or 640 pixels per line on the scope face and 200 lines. Two or four bits are required for each pixel in a color mode of operation. The memory size required to store a display can be calculated from numbers such as these. From these calculations the additional memory and complexity required for graphics displays versus alphanumeric displays becomes clear.

MAGNETIC-TAPE OUTPUT OFFICE OPERATION

7.10 Magnetic tape is often used as an intermediate storage medium for offline output equipment operation in large systems.

The advantage is in the speed-to-cost ratio attainable. A high-speed magnetic-tape recorder will be much faster than a high-speed printer so the computer will be less slowed down in recording results. The final printing may then take place on a printer, which interprets the recorded tape and prints the final result. This is another case of offline operation, in which the final computer operation is performed at the speed of a faster output device, and the computer is, therefore, not slowed down as much as it would be if a printer were used. Even the magnetic-tape recorder devices, however, are inherently much slower than the computer's internal circuitry.

[1]Oscilloscopes are quite often scanned a line at a time, as in television sets, and so some conversion electronics is required. IC packages to generate characters for CRT displays are made by several companies.

7.11 The process of transferring information into the machine and from the machine is especially liable to error. Although card and tape readers are constructed with the highest possible regard for correct operation, and the occurrence of errors is relatively infrequent, still errors occur, and it is desirable to detect them whenever possible. To facilitate the detection or correction of errors, two classes of codes have been invented: error-detecting codes and error-correcting codes. The first type of code enables the equipment to detect the errors which occur in the coded groups of bits, and the second type of code corrects the errors automatically.

Both error-detecting and error-correcting codes require that redundant information be sent along with the actual information being processed. The most commonly used type of error-detecting code is undoubtedly the parity-check code. Parity-check codes are commonly used for card and tape readers and for the storage of information on magnetic tape.

Parity checking

The parity check is based on the use of an additional bit, known as a *parity bit*, or *parity-check bit*, in each code group. The parity bit associated with each code group in an *odd-parity-bit* checking system has such a value that the total number of 1s in each code group plus the parity bit is always odd. (An *even-parity-bit* checking code has a parity bit such that the sum of the 1s in the code group plus the parity bit is always an even number.) The example shown in Table 7.1, which uses an 8, 4, 2, 1 code, has an odd parity bit which makes the sum of the 1s in each code group an odd number.

If a single error occurs in transmitting a code group—for instance, if 0011, 1 is erroneously changed to 0010, 1—the fact that there is an even number of 1s in the code group plus the parity bit will indicate that an error has occurred. If the values of the parity bits had been selected so that the total sum of the 1s in each code group plus the parity bit were even instead of odd, each parity bit would be the complement of the parity bit shown above, and the code would be an even-parity-bit checking code.

The technique of parity checking is doubtless the most popular method of detecting errors in stored code groups, especially for storage devices such as magnetic tape, paper tape, and even core-and-drum systems.

TABLE 7.1

DECIMAL	BCD	ODD PARITY BIT
0	0000	1
1	0001	0
2	0010	0
3	0011	1
4	0100	0
5	0101	1
6	0110	1
7	0111	0
8	1000	0
9	1001	1

If the parity-bit system is used, an additional bit must be sent with each code group. As another example, if the 7-bit ASCII is used, each line of a punched tape will have an additional hole position which will contain a 1 or a 0. When the tape is read by the tape reader, each code group will be examined, together with the parity bit; and in an odd-parity-bit system, an alarm will be generated if the number of 1s in a group is even.

This type of checking will detect all odd numbers of errors. Suppose that an even-parity check is used as in Table 7.1 and the code group to be sent is 0010; the parity bit in this case will be a 1. If the code group is erroneously read as 0110, the number of 1s in the code group plus the parity bit will be odd, and the error will be detected. If, however, a double error is made and 0010 is changed to 0111, the error will not be detected since the number of 1s will again be even. A parity-bit check will only detect odd numbers of errors. (The above rule will also apply when the parity bit is in error. For instance, consider an odd-parity-bit checking system where 0010 is to be sent and the parity bit is 1. If the parity bit is changed to 0, the number of 1s in the code group plus the parity bit will be even, and the error will be detected.)

There are many types of error-correcting codes, and some very clever and sophisticated coding schemes are used in both communications and computer systems. For instance, magnetic tape is a memory device that is especially prone to errors. Most errors are due to either imperfections in the tape or foreign matter which gets between reading head and the tape, causing the tape to be physically pushed away from the reading head and the recorded signal to be incorrectly interpreted. Such errors are said to be caused by *dropout*. These errors tend to lie in a single track, and several clever codes have been used to detect and correct such errors.[2]

KEYBOARDS

7.12 The important data entry device is the keyboard. In some cases keyboards are used to enter data into punched cards or punched tape, for example; in other cases keyboards enter data directly into a computer. The most familiar keyboards are on terminals which include either a printer or an oscilloscope (CRT) display. Sometimes small keyboards with only a few keys (similar to touch-tone telephone dialing keyboards) are used in industrial applications. These small assemblies of keys are generally called *keypads*.

When an operator depresses a key, electric signals must be generated which will enable the computer (or other device) to determine which key was depressed. This is called *encoding*. The encoding process is dependent on the mechanism used to make the individual keys in the keyboard.

The most direct method for encoding is based on the use of keyboard switches, which contain a switch similar to the pushbutton switch used in many electric devices. Figure 7.17 shows such a switch. When the plunger is depressed, the contacts of the switch in the housing are closed, and the two terminals at the output

[2]A study of error-correcting and error-detecting codes in some depth may be found in W. W. Peterson, *Error Correcting Codes*, and in G. Birkhoff and T. C. Bartee, *Modern Applied Algebra*.

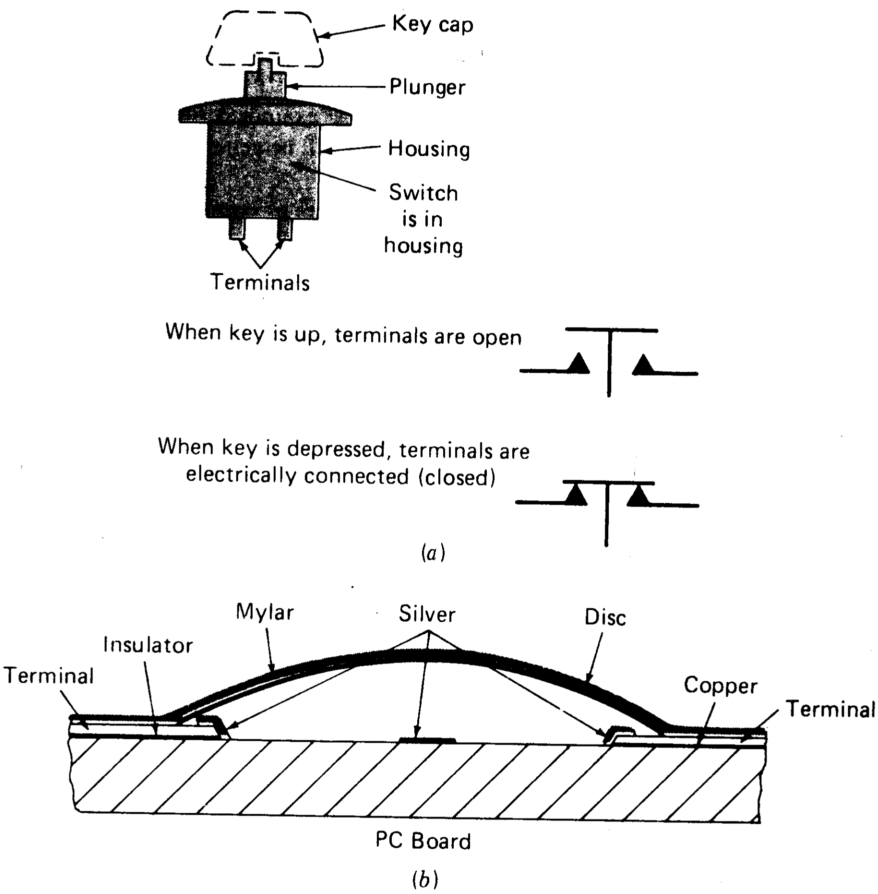


FIGURE 7.17

(*a*) Keyboard switch.
(*b*) Keypad switch.

are effectively connected. When the plunger is up (key is not depressed), the switch in the housing is open, and the terminals are not electrically connected.

To encode a keyboard by means of electromechanical switches, diodes (or transistors) can be used. Figure 7.18 shows the layout for encoding three keys into the ASCII illustrated in Fig. 7.8. An odd-parity-check bit has been added at the right end to make an 8-bit character. Each of the horizontal wires on the drawing is normally at a 0 logic level. When a key is depressed, however, the switch in its housing is closed, and this connects the wire to a logic level of 1. Thus if the *A* key is depressed, the value 10000011 will appear on the output lines, for example, because a diode connects the horizontal wire to the *A* switch to the leftmost and two rightmost vertical wires. This is the ASCII for *A* with a parity bit on the right end.

It is a good idea to load the values on the output lines into a flip-flop register before the computer reads the outputs. This has the advantage of storing the values until the computer can read them, particularly if the keyboard operator raises the key before the computer can respond. Figure 7.19 shows a scheme in which a flip-flop is used on each output, and a strobe is generated to load the flip-flops, by

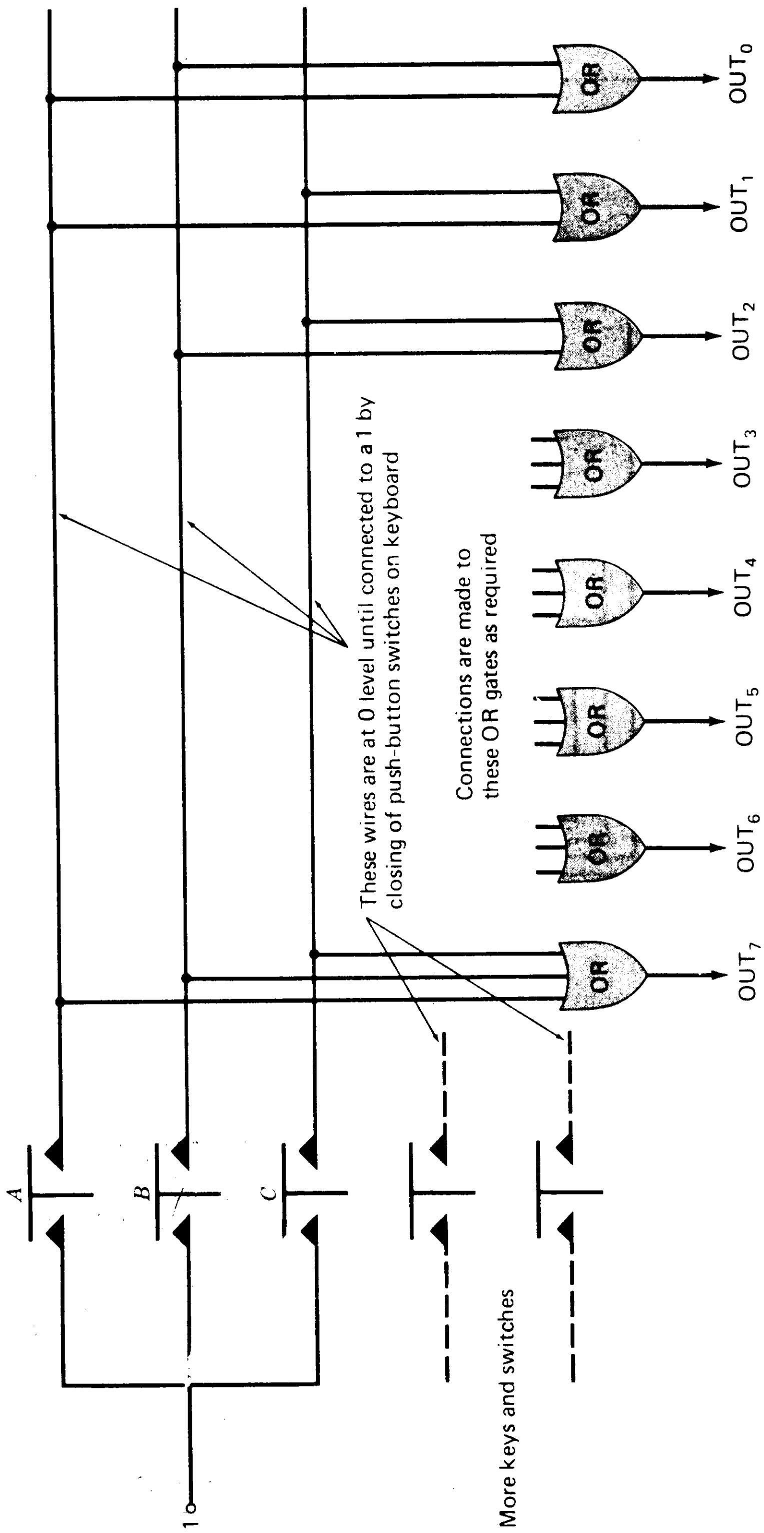


FIGURE 7.18

Encoding a keyboard.

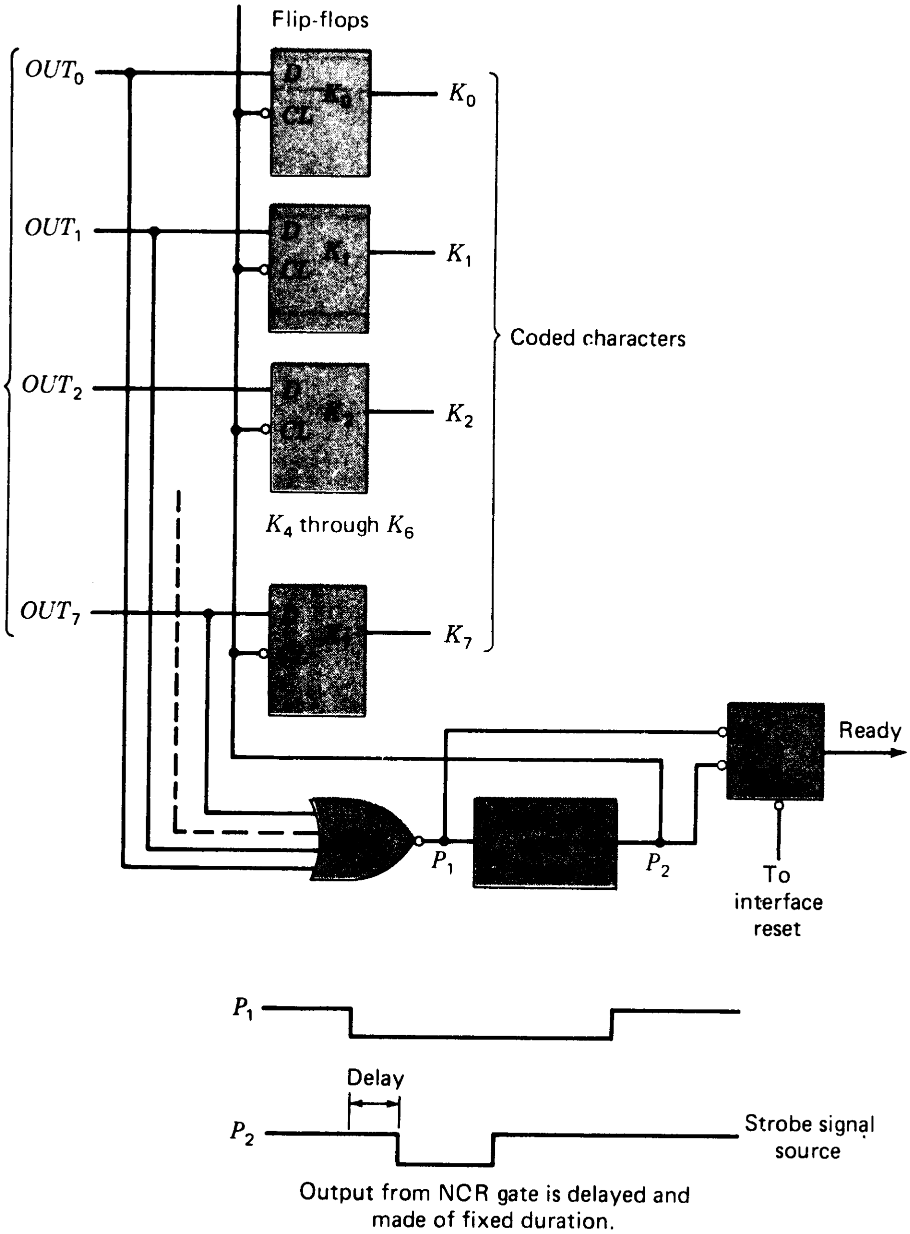


FIGURE 7.19

Keyboard buffer for interface.

using a delayed inverted pulse signal generated whenever any one of the output lines from the encoder goes high. The delay is inserted to compensate for signal *skewing*, where signals arrive at the output lines at different times because of differing delays through the wires in the system. The delay must be adequate to accommodate the largest delays that may occur. Also, the length of the strobe pulse should be short compared to the shortest time a key might be depressed. (A delay of 1 ms and a pulse of 1 ms would be reasonable.)

Here we are assuming that the switch contacts do not bounce, as is the case with some switches. If the contacts do bounce, the output signals must be "smoothed," and various circuits are available. When reed relays are used (and these are sometimes used), there is little need for this. (The switches are often momentary-contact switches which generate a closure of relatively fixed duration.)

In Fig. 7.19 the strobe signal is also used to load a flip-flop, called *ready*, which will be used in an interface design in the next chapter. Notice that the encoding scheme shown here requires that there be a 1 in the code for each character (so that the strobe pulse will be generated).

The keyboard market is very large, and so many kinds of keyboards are now made as manufacturers compete to see who can produce a lower-cost, more reliable, more durable keyboard. The basic division of keyboards is (1) the electromechanical keyboard, which includes the switch type just explained and (2) the solid-state keyboard.

There are several basic mechanisms for solid-state keyboards. Capacitor types have mechanisms which vary the capacity of a capacitor when a key is depressed. These are low-cost keys, often used in keypads and other cost-conscious keyboards. Hall-effect keyboards are more expensive, but have long life and good key feel, as do ferrite-core and photooptic keyboards. Each of the basic mechanisms has different problems with regard to encoding the keyprinters' output into a coded form usable by a computer. (The references include discussions of the encoding techniques.)

The encoding technique is often based on a two-dimensional array of keys and wires instead of the "linear" array shown in Fig. 7.18 for reasons of economy. (This is discussed in the Questions.) IC packages for encoding are made by several manufacturers and can include *smoothing* or *debouncing* for contacts and sometimes *key rollover* protection, which protects against two keys being depressed at the same time. (This can happen when an adjacent key is inadvertently depressed or when the next character is struck before a key is released.)

Chapter 8 discusses interfacing keyboards and printers in some detail. However, the overall physical setup for interfacing a mini- or microcomputer is discussed now.

Figure 7.20 shows the general layout for a microcomputer. Printed-circuit boards contain the logic for the CPU, which consists of the arithmetic-logic unit (ALU), the control section, and the high-speed (IC) memory. The high-speed memory is connected to the bus, and the control section controls the high-speed memory by using signals it places on the bus. (The bus in Fig. 7.20 consists of a set of wires running under the printed-circuit boards. Printed-circuit-board connectors are mounted to these wires, and the printed-circuit boards are plugged into these connectors, thus making connection to the logic on the boards.)

To interface input-output devices with the CPU section boards, *interface boards* are connected to the bus. The boards contain the logic gates and flip-flops to read from and write onto the bus and to control and interface the input-output devices.

In a typical system, in order to interface a keyboard, a keyboard interface board is connected to the bus, by a connector, and then a cable runs from this board to the keyboard. Similar boards are used to interface a tape unit, a disk drive, etc.

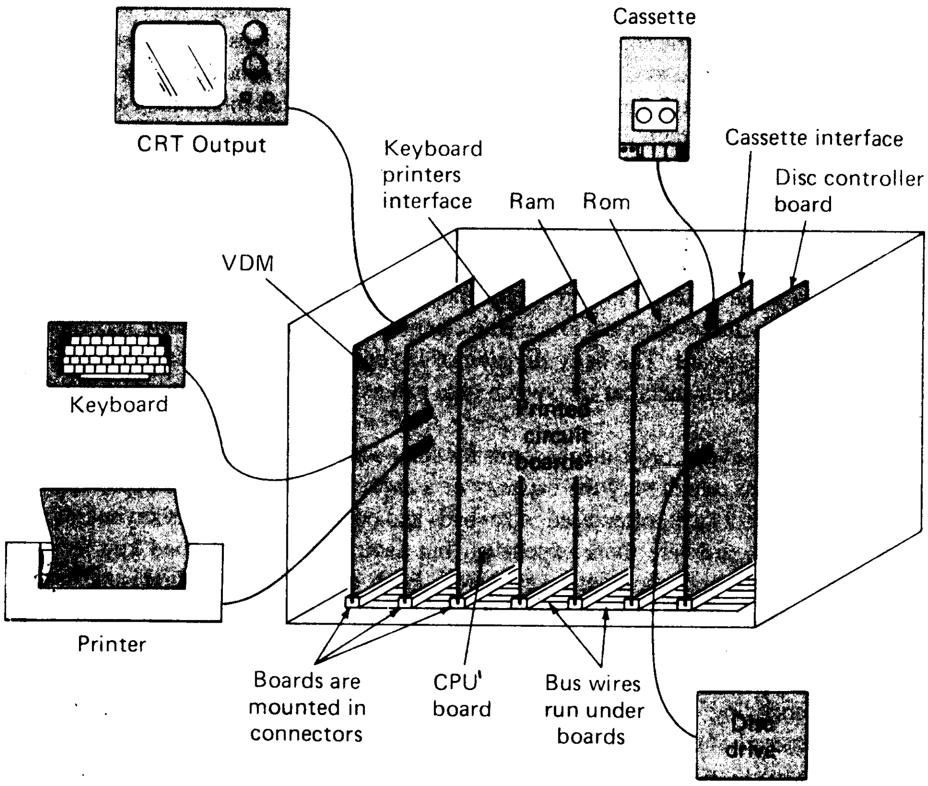


FIGURE 7.20

Microcomputer interface layout for input-output devices.

Notice that communication between input-output devices and the CPU utilizes the interface boards and the bus. In very small systems the CPU and one, two, or more interface chips may be placed on a single board, with the bus on the same board. Cables then connect this board to the input-output devices.

TERMINALS

7.13 When a keyboard is combined and packaged together with an oscilloscope display or a printing mechanism and suitable electronics is provided so that characters struck on the keyboard can be entered into a computer and the computer can also have the ability to print on the oscilloscope or printing mechanism, the keyboard and printer or keyboard oscilloscope are called a *terminal*. Terminals are widely used to input programs and data to computers.

Figure 7.16 shows a terminal consisting of an oscilloscope and a keyboard. Characters struck on the keyboard will appear on the oscilloscope face, and output signals are also provided which are suitable for computer usage. A computer can print on the oscilloscope providing for two-way communications. The primary disadvantage is that no hard copy, or record of what is typed by the terminal operator or printed by the computer, remains when the terminal is turned off.

Terminals range from minimal systems containing almost no memory to *smart* terminals in which a small microcomputer or minicomputer is included. Smart terminals provide many facilities for the user, sometimes including text editing, input formatting, and checking for typing errors made by operators.

Oscilloscope-keyboard combinations almost always contain a small memory in which the present contents of the screen are stored, so that the screen can be refreshed by the terminal electronics instead of requiring the computer to continually rewrite data on the screen. In this case, a small memory and some electronics are required, and microprocessors are often used to control the terminal's operation. By enlarging the memory capabilities and program in the microprocessor, a smart terminal can be formed without too much additional cost.

Terminals generally generate output data in serial form and accept input data in serial form. The formats, coding, and electrical properties of the signals generated vary from terminal to terminal, but there are now strong movements to standardize the interface designs.

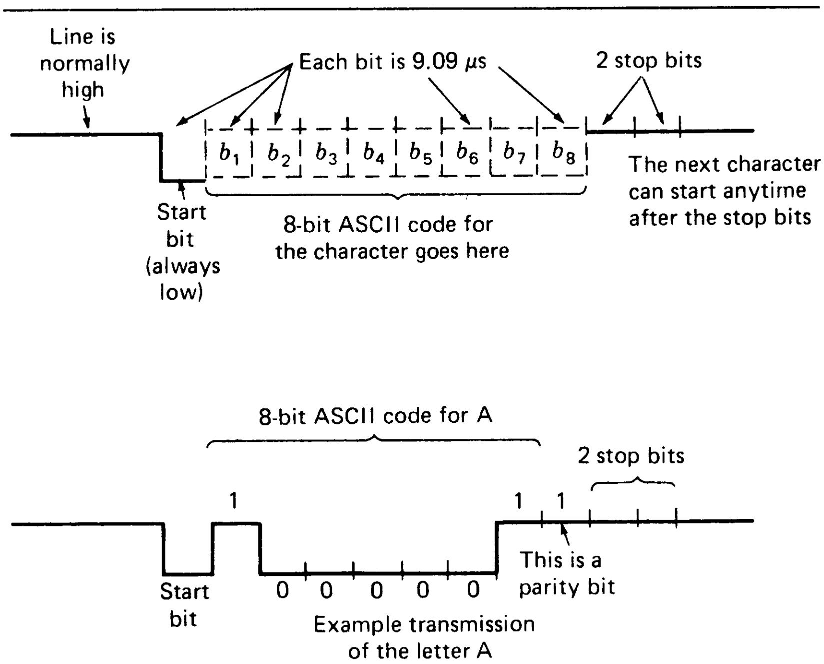
Most of the terminals in present-day use generate an 11-bit output in serial form whenever a key on the keyboard is depressed. This same stream is generally used to activate the printer mechanism or oscilloscope. In most cases, when a key is struck, the code for the character is read by the computer, which must then write this character back into the terminal. This is called *echoing* the character. Microcomputers generally echo characters.

Figure 7.21 shows how the present standard character transmission operates. The output line from the keyboard is normally in the high[3] state, but when a key is depressed, a *start bit* at a low level is generated. This start bit is followed by

[3]Communications people call this a *mark* value and the other level a *space*.

FIGURE 7.21

Sending a character using the standard code.



the proper 8-bit ASCII character, and then two *stop bits* at the high level are inserted before another character can be started (that is, before another start bit can be generated). The start bit, each stop bit, and the bits in the ASCII character are each of the same time duration. For most current systems this is $\frac{1}{110}$ s for each; and since 11 such time periods, or *bit times*, are required, a single character requires $\frac{11}{110}$, or $\frac{1}{10}$, s.

Several other speeds are currently used, including 300, 600, and 1200 bits/s. In each case, the same character construction with start and stop bits is used, and so character rates of $\frac{11}{300}$, $\frac{11}{600}$, and $\frac{11}{1200}$ s are attainable.

The character transmission system described here is called *asyncnronous transmission,* for the character and start bits can occur at any time. *Synchronous transmission* systems have the bits clocked into fixed time periods, and characters are placed in fixed positions in the bit stream. These systems require both bit timing and character timing to be established between the transmitting device and receiver and thus are more complicated. Character transmission can be at higher rates, however. (Since start and stop bits are not required, the character beginnings and ends are established by the system.) As a result, high-speed data communication is generally in synchronous form.

The output levels and interface requirements for the coded characters are explained in the Questions, and several standards are noted.

When terminals are operated at some distance from a computer, the telephone system is often used to provide the necessary communications link. If a terminal is to be operated into the telephone system, special devices are needed to translate logic levels produced by the terminal into signals acceptable for telephone-line transmission.

Figure 7.22 shows a terminal with an *acoustic coupler*. The handset from the telephone is placed in this acoustic coupler. When a key is depressed, the bits comprising the character are converted to audio tones by a small loudspeaker in the coupler. Generally one frequency is used for a 0 and another for a 1. These signals enter the transmitter part of the handset and are transmitted into the telephone line. At the computer end these frequencies are received by a microphone connected to a handset and then converted back to 1s and 0s in electrical form. The electrical logic levels output at the receiver are therefore replicas of the signals originally generated at the terminal.

The computer "talks" back to the terminal by using an acoustic coupler at its end of the telephone line, and the acoustic coupler at the terminal converts the computer's signals back to logic levels which are used to drive the display.

When output signals in logic-level form are converted directly to electric signals suitable for telephone transmission by electronic circuitry (instead of acoustic coupling), the converting device is called a *modem*. A modem can convert logic levels to electric signals for the telephone system and can convert received signals from the telephone line back into logic levels. This means, of course, that the connections to the telephone line are made electrically (generally into a telephone jack) and directly into the telephone line, and a handset is not used. It also means that the electric signals must comply with telephone system regulations. The design of modems which will (1) send bits through telephone lines at high speeds, (2) make few errors in transmission, and (3) comply with telephone company regulations is a highly developed and interesting scientific area.