Chapter 9
I/O Devices

This is an adapted version of portions of the text A Practical Approach to Operating Systems (2nd ed), by Malcolm Lane and James Mooney.

Revised: Mar. 7, 2011

9.1 INTRODUCTION

One of the major support services provided by an operating system is the management of I/O devices. Early operating systems provided little, if any, I/O service request capability. In many instances in simple disk operating systems, I/O support was provided by library subroutines written for a specific device, which were loaded into the user memory space. Multiprogramming operating systems that support multiple users and processes, however, cannot allow applications to directly control input/output. Hence, the operating system must provide I/O services to control all devices in the system; for example, to issue read/write commands, process device interrupts, and handle error detection and recovery for devices. These services are typically provided by OS modules called **device drivers**.

Before studying techniques for managing devices, it is important to gain some understanding of the characteristics of the devices themselves. This chapter provides a brief overview of the properties of the most common I/O devices in use today. This information provides a foundation for the study of device drivers and device management in the next chapter.

9.2 CHARACTERISTICS OF PHYSICAL DEVICES

Some Basic Issues

Implementation of device drivers for physical devices requires a thorough understanding of the characteristics of the specific devices being controlled. In this chapter we describe some common device types, and discuss some issues that should be kept in mind when confronting various types of devices. Here are some important considerations:

- devices differ greatly
- programming devices can be quite complex
- dynamic operation of devices is often ambiguously described in hardware documentation

- testing can be difficult, particularly testing of error recovery routines

It is important to understand a specific device thoroughly before attempting to design and implement a device driver for it.

The Variety of Device Characteristics

In addition to common operations such as read and write, many devices have unique control operations that must be supported in their device drivers. Printers, for example, may require special operations for selecting character sets, completing pages, and other control functions. Terminals often require operations to clear the screen or move a cursor to a specific position. Magnetic tape units are capable of many control operations like rewind, backspace record, and forward space record. Disks are capable of direct access to a particular track, and thus seek operations must be supported. While often hidden from the user, all these special operations and capabilities must be provided as operating system services that are used by higher-level software. Figure 9-1 outlines I/O data transfer and control operations for a variety of typical devices: a "typical" disk, a simple printer, a keyboard and mouse, and a bit-mapped display. The display is in a special category, since information is transferred by writing directly to a special memory.

DEVICE	CONTROL OPERATIONS	DATA TRANSFER OPERATIONS
Disk	Intialize Seek to selected block	Read block Write block
Printer	Initialize Load or select font Select paper tray	Initialize Print character Load font
Keyboard	Set status	Read character
Display	None	Store character or pixel
Mouse	None	Read position Read button status

Figure 9-1: I/O Operations for Typical Devices

A special type of device, discussed in detail in the previous chapter, is the clock. Although clocks are not exactly I/O devices, in many cases clock drivers are implemented in the same manner as device drivers. The main difference between a clock and an I/O device is that a clock performs no physical input or output; it only generates a signal to be used internally by programs.

9.3 CHARACTER DEVICES

A variety of common devices process information in the form of codes representing individual characters that can be displayed on a screen or printed on a page. These devices are collectively called character devices. Examples include keyboards, terminals, printers, and communication devices. These devices are relatively slow in data transfer to and from computer memory. The following subsections summarize the key properties of typical character devices.

Keyboards

A **keyboard** is a simple character device that sends a character code to the computer whenever a key is pressed. On most personal computers and workstations, the keyboard is an independent device used in conjunction with a memory-mapped display (see next section). On larger multiuser systems, the keyboard may be combined with a serial display device into a unit called a terminal (see below).

A keyboard contains circuits to translate each "keystroke" into a numeric value. These values may follow the ASCII encoding, or they may use a special extended code designed for a specific type of computer. In this case there may be separate codes generated when a key is pressed and when it is released, and there may be special codes for various key combinations. A keyboard driver may be responsible for translating these codes into values meaningful to the operating system and application software.

Terminals and Displays

A **terminal** is an independent device embodying a keyboard and a display device. Early terminals took the form of typewriters, with mechanical keyboards and a typing mechanism that printed lines of text on a continuous roll of paper. The most common of these devices was the teletype (produced by Teletype Corporation), often abbreviated `tty`. Today such devices are no longer used; however, the teletype model introduced in Chapter 3 still influences some user interfaces, and the term tty is widely used in UNIX environments as a name for any terminal device.

Today terminals have largely been replaced by personal computers running software that *simulates* the behavior of a true terminal device. However, it is still instructive to consider the underlying model on which this simulation is based.

Some terminals are capable of displaying graphics or text with various special attributes (bold, underlined, slanted, etc.) The most common terminals, however, display a basic set of characters based on the ASCII standard, with up to 24 lines of 80 characters on the screen at one time. This type of device supports the **alphanumeric** user interface model.

Along with input codes that represent actual characters, terminals usually accept a variety of codes that specify positioning and control information such as clearing the screen, inserting or deleting characters, moving the cursor (which represents the current position) to a specified screen location, etc. These codes are called control sequences or escape sequences. In general each terminal may have its own unique encodings for control sequences. However, most terminals are able to emulate a common terminal that was produced by Digital Equipment (now Hewlett-Packard), called the VT-100. As a result, the VT-100 model has become a standard that many programs follow, to ensure operation on the widest range of terminals. Other models that have been used include the Tektronics model, widely used for graphics, and the IBM 3270, commonly used to access IBM mainframes.

A powerful alternative to the terminal, a **memory-mapped display**, is used on personal computers and workstations. This type of display is not actually an output device, but a separate processor that shares memory directly with the CPU. The information in the memory may be viewed either as character codes or, more commonly, as graphic picture elements known as **pixels**. In the latter case the device is capable of displaying sophisticated graphics, not only text. Memory-mapped displays are capable of very high performance, but they can be used only where the display is closely connected to the processor itself.

A terminal is often viewed as two separate devices (display and keyboard), but it may also include some internal connections between these devices. In particular, when a key is pressed the terminal may have the ability to echo that key by displaying the appropriate character or control action directly on the screen. Direct echoing is normally disabled, however, because the OS usually requires a high degree of control over the effect that should actually appear on the screen each time a key is pressed.

Printers

Today's printers are of many different types. Impact printers rely on hammers or print heads, which strike a character onto paper using a ribbon. Impact printers produce low-quality output, but have long been used for certain types of specialized or high-volume work. Non-impact printers are widely used for general-purpose and high-quality output. The most common types are ink jet printers, which have relatively low cost, and laser printers, which can print at high speeds with very high quality.

Today's ink jet and laser printers are typically sophisticated computers in their own right, with complex internal programs for special functions and large memories for storing documents and character fonts. A significant task for device drivers may be to set up and program this internal memory. Thus device drivers for printers can be complex or simple depending on the level of support for special features within the driver itself.

Like terminals, printers may be attached to the computer using a serial interface. However, many printers make use of a parallel interface, in which all of the bits of a character code are transmitted to the printer at once. Higher speed serial interfaces, such as USB, are gradually replacing parallel interfaces in most instances.

Communication Devices

Many current computers are directly connected to various types of networks; most others have at least the ability to communicate via telephone lines. These abilities are supported by a wide variety of communication devices.

-

A modem, short for modulator/demodulator, is a device that converts digital computer data to the analog form required for transmission over telephone lines. Not too many years ago, the fastest modems operated at a speed of 1200 "baud", roughly equivalent to 120 characters per second. Today modems often operate at fifty times this speed. Modems are normally connected by a serial interface. They may have sophisticated internal processors to perform error checking and other important functions. They are programmed in a similar fashion to terminals.

Direct network interfaces take on many different forms; the familiar Ethernet interface is the most common example. These interfaces are outside the scope of our discussion. Since software may be responsible for sophisticated communication protocols, drivers for this type of device can be extremely complex.

9.4 BLOCK STORAGE DEVICES

Most file systems are maintained on magnetic storage devices. Implementation of device drivers to control such devices can be quite complex. A later chapter concentrates on file systems. The concern in this section is to understand the operation and control of magnetic storage devices. For practical working storage on present-day file systems, these devices are usually magnetic disks of some type. Magnetic tape was used for early and very simple file systems, but its usefulness is limited by its strictly serial nature. However, tape is still widely used for backup and long-term archiving of file systems, and for transporting files between computers. One non-magnetic technology that has rapidly gained importance is optical storage. Optical disks have a high initial cost, but they offer rapid access and extremely high capacity.

As will be seen in the discussion below, all of these storage devices most effectively transfer data in relatively large blocks rather than individual bytes. For this reason, they are often referred to as **block storage devices**.

Properties of Magnetic Devices

Magnetic tapes and disks of all types store information by a similar technology. Each device consists of a base of plastic material, on which a thin magnetic surface is deposited. This forms the basic storage medium. Information is recorded on this magnetic surface along a set of long, thin regions, called tracks. At various points on each track, a bit of information can be written by establishing a certain pattern of magnetic polarization. This is accomplished by maintaining a suitable electric current in an element called a **write head** as the desired spot is moved past it. This magnetization is permanent until disturbed, and it can be sensed by observing the current generated in a similar **read head** as the spot once again moves past. Physical movement of the storage medium is an essential requirement for this type of storage. This concept is shown in Figure 9-2.

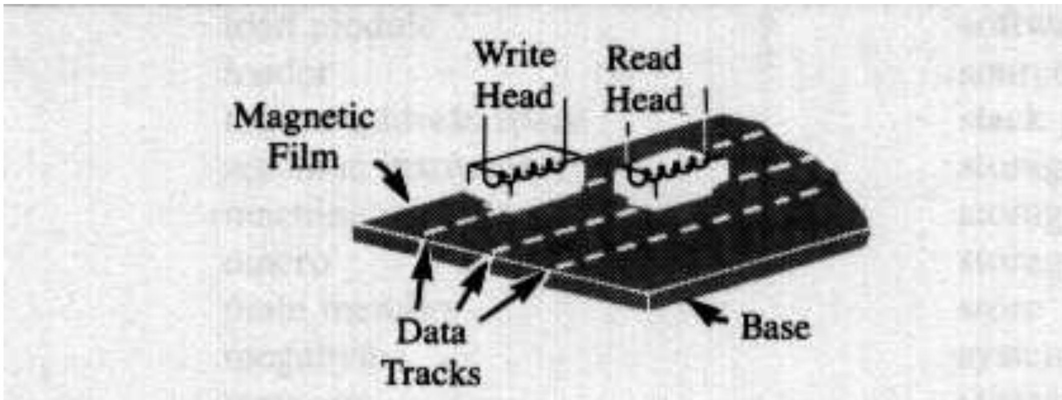

Figure 9-2: Magnetic Storage of Information

On most disks and some tapes, control information must be recorded initially in certain places to aid in locating the correct positions to read and write data. The process of writing this initial information, called **formatting**, must be carried out once before the device can be properly used.

Also on most magnetic devices, some space is set aside for check information to ensure the correctness of the stored data. This is necessary because storage devices sometimes develop flaws that can result in the loss of data. The interface hardware or device driver software must maintain this information and use it to verify the correctness of data that is read from the device. Also, device drivers must often provide the function of formatting these magnetic media.

Magnetic tapes and disks share some common properties that affect their use in file systems. The most important of these properties are:

- **non-random access**. To some degree, all devices fail to provide random access to any part of their contents at equal cost. The time required to reach a desired location depends on the starting location. The extreme serial nature of tapes is obvious. Non-random behavior in disks is less severe and varies widely with the type of disk. In most cases, however, the physical placement of information can have a significant effect on performance.

- **blocking**. Information on tapes and disks is organized into physical blocks, each containing a large number (typically hundreds) of bytes of data. In most cases blocks must be transferred to or from primary memory as a whole. It is not possible to access or modify a partial block.

Magnetic Tapes

Magnetic tapes consist of a continuous strip of a thin storage medium, which can contain a number of parallel tracks. A typical tape for a large computer system is shown in Figure 9-3.

This tape is one-half inch wide and 2,400 feet long, with nine tracks. It is stored on a large plastic reel, similar to a high-quality audio tape. Tapes are mounted on drives as needed, and may readily be changed.

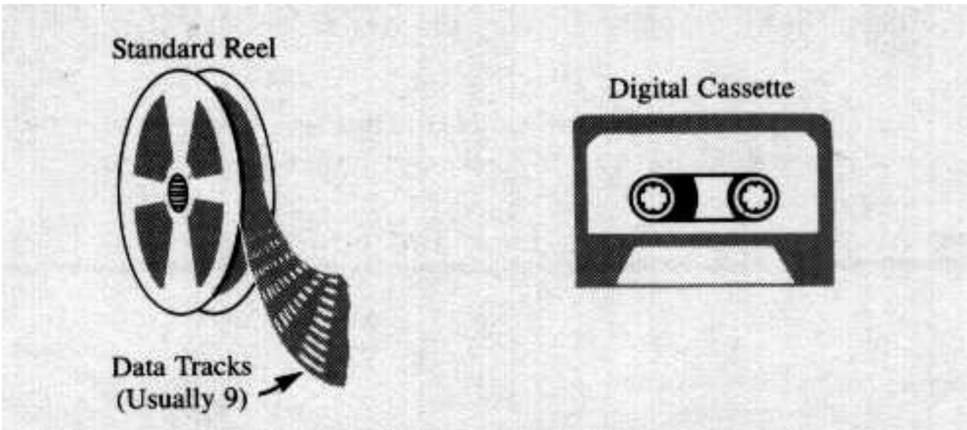

Figure 9-3: Magnetic Tape Structure

Eight-bit bytes of data are written in parallel on eight tracks; the ninth track receives a parity bit for error checking. Information may be written as densely as several thousand bytes per inch within a block; however, an inch or more may be required between blocks for control information, or simply as an erase gap to separate records. The size of each block may be chosen by the program that writes the data, and could vary from a few bytes to an entire tape.

In most magnetic tape systems, data can be located and read only when the tape is moving forward. To access data behind the present point, it may be necessary to rewind the tape to the beginning and then search forward. On a large tape this could take several minutes. It is often impossible to write data to a tape without destroying all existing data following that point. Although some tape systems have been devised to overcome some of these problems, as a whole they form a serious limitation.

Because of these limitations, practical use of tapes is confined to applications in which the entire tape will be read or written in sequence. Copying of entire file systems for backup or other purposes is a good example. Within its limits, tape offers a means of inexpensive and compact storage for unlimited amounts of data.

Magnetic Disks

A magnetic disk is a circular base of support material on which a magnetic film is deposited. The base may be heavy and rigid, or more similar to magnetic tape material---thin and flexible. The diameter can range from a few inches to more than a foot. Tracks, formed on this base in a series of concentric circles, may be placed on both surfaces of a disk, or on one surface only. A disk operates by rotating at high speed, usually thousands of times per minute. If a suitable head is positioned over a track, it can read and write data at any position as the track

-

passes under it. On rigid disks, such disk heads ride on a cushion of air created by the rapid rotation of the disk platter. In this case, springs push the head(s) toward the disk, but the air cushion prevents the heads from touching the surface, the clearance being only a few thousandths of an inch.

Any particle, scratch, or other foreign body can break this cushion of air and result in a **head crash** if the surface of the disk and the head are damaged by this contact. On a flexible disk, a head often rides on the surface of the disk because the disk does not spin fast enough to provide this cushion of air. Hence, constant use results in wear of the disk surface and oxide collecting on the disk read/write head. Periodically, such disks should be replaced and the disk head(s) should be cleaned. Device drivers that can recognize and record the occurrence of intermittent, recoverable errors on such disks can warn of impending problems with these media.

Each track may contain one or more blocks of data. Usually each track is permanently divided into parts, either by mechanical marks or by formatting. Each part is called a sector. Generally, a sector holds a block of data and is the smallest addressable unit for reading and writing from a disk.

Some larger disk units include multiple disk platters mounted on a common spindle. All disks on such a unit rotate together.. On a unit with multiple surfaces, the set of tracks that are at corresponding positions on each surface is called **a cylinder**.

A number of types of disks are illustrated in Figures 9-4 through 9-6. Early disks were large and rigid, and could not be removed. These disks, equipped with a separate head for every track, are called **fixed-head** or **head-per-track** disks. Gradually a new type of disk became more common, the **moving-head** disk, in which a single head is provided for each surface. The heads are mounted on a moving arm, which can be positioned over any desired track. More recently, versions of the moving-head disk that use a small, thin, flexible base mounted in a paper envelope have been developed. These floppy disks, also called diskettes, are removable and inexpensive, and they have formed an important storage method for small computer systems. Each of these types is discussed below.

Many disk units, like tapes, can be physically removed and replaced. These disks may be used for long-term data storage.

Head-per-track Devices. Figure 9-4 shows the similarities in structure of a head-per-track disk, and a magnetic drum unit. Both of these storage devices were commonly used in older, large-scale computers. Instead of a set of concentric tracks, the drum has a set of tracks that form a single cylinder.

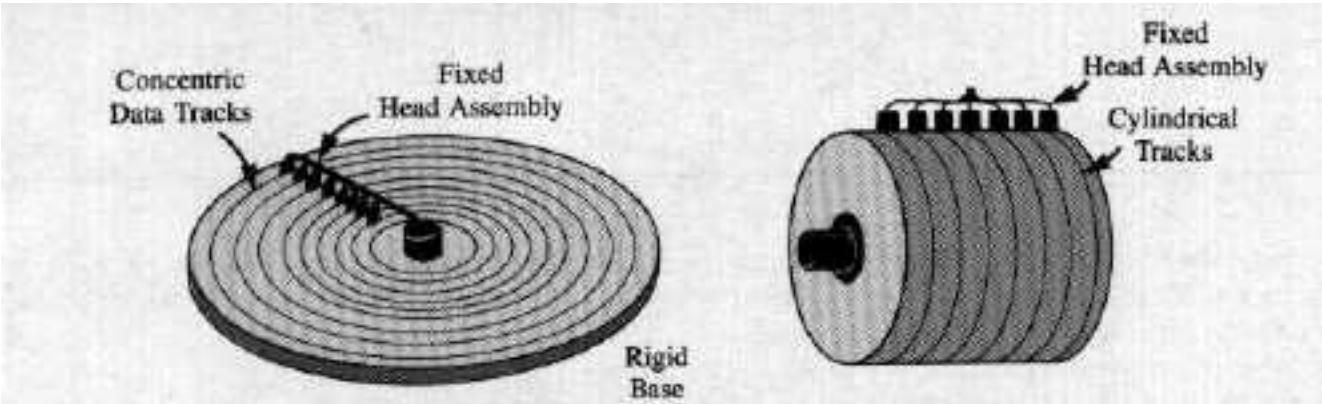

Figure 9-4: Example of a Head-per-Track Disk and Drum

A read and write head is permanently positioned over each track. This type of disk is very expensive, due to the large number of heads. Moreover, its storage capacity is limited. The spacing between tracks must be much larger than on moving-head devices to provide sufficient room for the heads.

On the positive side, head-per-track devices offer very fast data access, and access that comes closest to being truly random. For these reasons such devices are still used in some larger systems, especially in applications demanding very high performance.

Moving-head Disks. The structure of a typical moving-head disk is shown in Figure 9-5. A single head for each surface is mounted on a moving arm. The arms for each surface are tied together and move as a unit, so that at any time, all heads are positioned on the same cylinder.

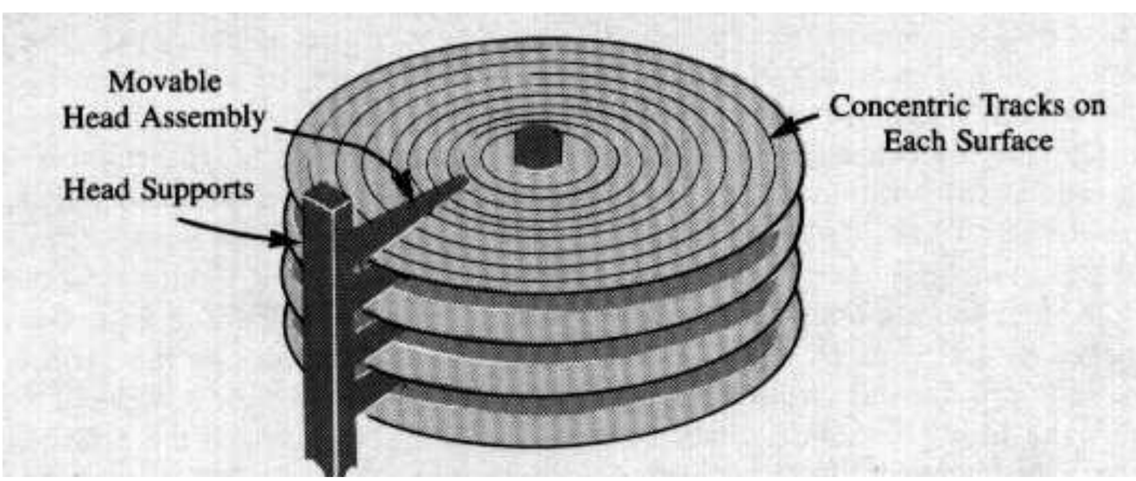

Figure 9-5: Example of a Moving-Head Disk

-

To access data on a moving-head disk, it is first necessary to move the heads to the proper cylinder. This introduces a delay, called **seek delay**, which is roughly proportional to the distance moved. This seek delay is a major factor limiting the speed of access.

Thus, moving-head disks have much slower average access speeds than head-per-track devices, and the actual access time is strongly dependent on the position. However, these devices are much less expensive, physically compact, and can have a high data capacity. Many moving-head disks are removable.

Newer types of moving-head disks include both the storage media and the read/write mechanism within a compact sealed unit. This type of construction is known as a **Winchester disk**. Because it is precision-made and sealed, this type of disk can have a high data density, low cost, and good reliability even when subject to physical shocks. This technology today allows the storage of as much as tens of gigabytes of data on a storage media only a few inches in diameter. This is the basis of low-cost, high-capacity "hard" disks on present-day computer systems.

Floppy Disks. The final type of magnetic disk is the floppy disk, also called a "flexible disk" or "diskette." The structure of a floppy disk is shown in Figure 9-6.

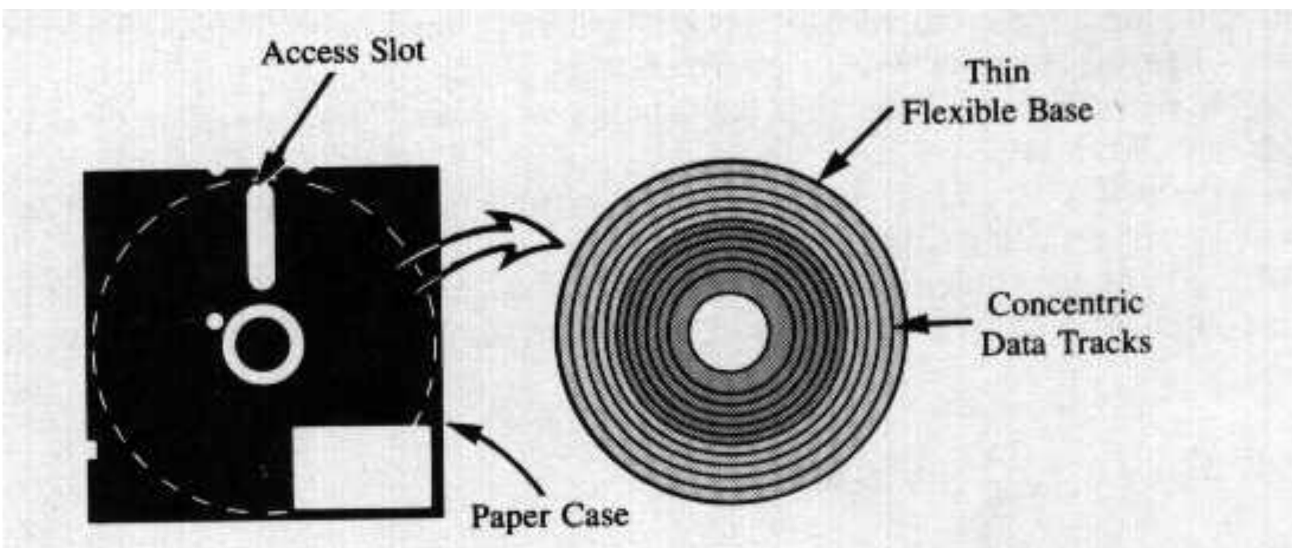

Figure 9-6: I/O Operations for Typical Devices

Floppy disks are a special type of moving-head disk, always removable, with a thin, flexible base. The disk is mounted inside a paper jacket, with a coating that allows it to rotate freely. A slot in the jacket allows data to be accessed as it passes by the slot.

The first floppy disks were eight inches in diameter. Newer types used smaller sizes, most commonly 5-1/4 inches or 3-1/2 inches. Various formatting techniques have been used to obtain

-

high data density. Early disks could store only a few hundred K bytes. Standard 3-1/2 inch disks can store over a megabyte of data.

Floppy disks are often mounted on inexpensive drive mechanisms that rotate only when needed. On such devices, there is a startup delay for each new block transfer. As a consequence, it may not be possible to read all the sectors of a track in their physical order during a single rotation of the disk. This leads to the practice of sector interleaving. The sectors on each track are numbered in such an order that one or more sectors are skipped between consecutively numbered positions. This technique defines an order by which sectors can be accessed in the fewest possible revolutions.

Floppy disks are very slow to access, and highly non-random. Because of this they are not suitable for general-purpose working storage, which must be randomly accessed with high frequency. They perform quite well for occasional use. Because floppies are removable and very inexpensive, they form a low-cost alternative to tape for archival storage of moderate quantities of data.

Optical Storage

A technology that has emerged as a competitor to magnetic storage is that of **optical disks**. A well-known example of optical storage is the audio compact disk (CD) and the closely related CD-ROM. More recently these have been joined by the closely related digital video disk (DVD) both for video storage and and data storage. Typical CDs have a capacity of about 640 megabytes, while that of the DVD is over two gigabytes.

These devices consist of a base of glass or similar transparent medium coated by a thin film of material, the optical properties of which can be changed by a laser beam's heat. As the disk is scanned with a laser, the surface is modified in a prescribed way, forming patterns of bubbles and pits. These features reflect light in a different way than the original surface. In the closely related magneto-optical technology, a laser beam controls the magnetic properties of the disk rather than its optical properties.

Originally, optical disks were read-only devices. Today both read-only and rewritable technologies are available.

Unlike the concentric tracks on magnetic disks, data on optical disks are written on one long spiral track as on audio disks. Each block occupies the same length on the track. This allows the highest data density, but complicates block location; in addition, the rotation speed must be varied as the access position changes between inner and outer tracks.

To read information another laser beam scans the surface, and the reflected pattern is detected by suitable sensors. The beam is reflected at a different angle where data marks have been formed on the surface. The structure of an optical disk is illustrated in Figure 9-7.

Optical storage offers the advantages of very high density and high reliability. A disk the size of a CD-ROM can store hundreds of megabytes. Once properly recorded, information on an optical disk cannot deteriorate. The principal disadvantages are high cost and slower speed than many magnetic devices.

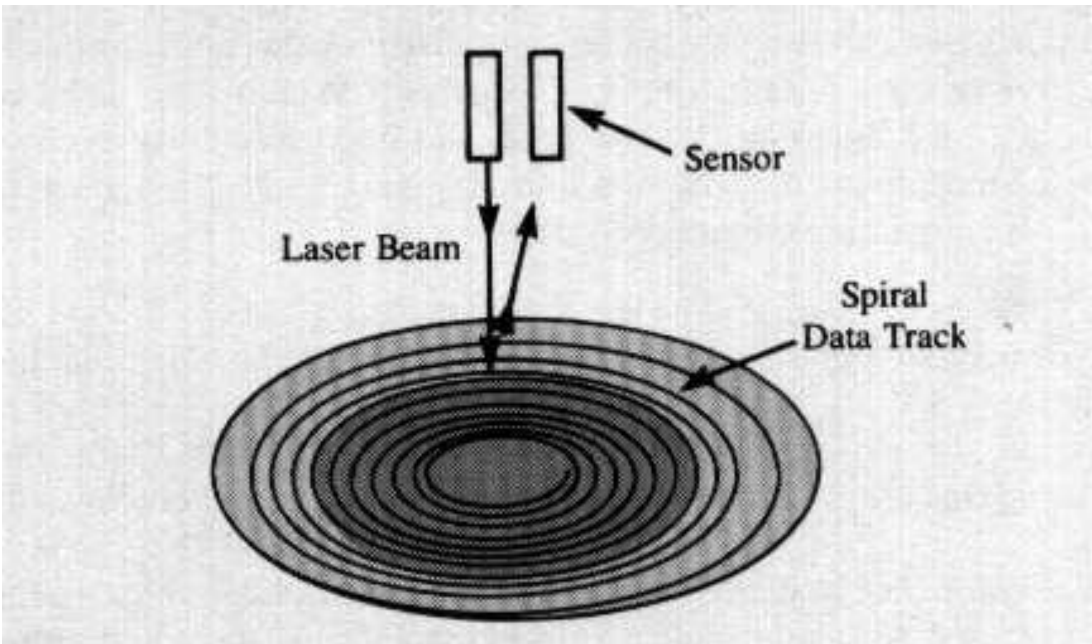

Figure 9-7: An Optical Storage Disk

Abstract Block Storage Devices

Many of the storage devices we have discussed have a similar basic structure, despite their many differences. It is possible to treat most devices uniformly by defining a model for an **abstract block storage device**, the properties of which are assumed to be simple and consistent, even if the physical device is more complex. Such a device is considered to contain an address space consisting of an ordered sequence of blocks, all the same size. The size is chosen to be reasonably consistent with the physical devices to be used; typical choices are 512 or 1024 bytes.

The sequence of blocks is assigned to the physical device in such a way that accessing a consecutive sequence of blocks in order will be efficient. It is understood that random access may be more costly and that the exact cost varies with the device.

This approach, used by UNIX and other OSs, allows software that works with storage devices to assume a common set of characteristics, even if the actual devices vary widely. A possible reduction in performance in some cases is accepted as the price of such independence from the structure of the physical device.

9.5 OTHER DEVICES

This section briefly overviews a few additional device types that do not fit clearly into either of the categories discussed above.

Pointing Devices

Most present-day computers are equipped with a pointing device, which is used to select a particular position on a display screen. The most common such device is the mouse, a device which records the movements of a rolling ball on a surface. Other devices with a similar purposes include trackballs and joysticks. Each of these devices supplies the processor with a pair of integers, representing horizontal and vertical positions, whenever a sufficient amount of movement has occurred. The values provided by mice and trackballs are relative positions, while those provided by joysticks are absolute.

Besides position data, each of these devices includes one or more buttons which can be pressed, indicating that some special action is to be taken at the current position.

Device drivers for a pointing device are concerned primarily with keeping track of the selected screen position, and responding to button clicks. Inputs from these devices are typically handled by an interrupt mechanism.

Multimedia Devices

Increasingly some current systems are provided with devices that input and output information in forms very different from the traditional character codes. These include devices for input and output of sound and of both still and video images. Usually, such devices include extensive hardware capabilities for translating and compressing information. The processor sees this information as a sequence of digital integers, which may have to be generated or accepted at very high speeds.

Programming these devices is extremely complex, and outside the scope of our study.

General Analog Devices

Computers are often used to control and monitor physical processes in laboratories, factories, etc., as well as in embedded applications like automobiles and appliances. In this case inputs and outputs often take the form of physical signals with a continuous range of real-number values. Devices called analog-to-digital converters and digital-to-analog converters are used to transform these signals to bit sequences understandable by the processor. Often these devices have strict timing requirements, which makes their programming very complex.

-

Virtual Devices

Some device drivers manage simulated devices which do not correspond directly or one-to-one to any physical device. These devices are called virtual devices or software devices. For example, disk units may be simulated by allocating a part of main memory to store data as if they were a disk unit, primarily to improve the performance of application software in small systems with relatively slow disks. These simulated disks are called RAM disks (random access memory disks). In other cases a single physical disk, may be divided up into several logical or virtual disks, each one viewed by users as an individual device.

As yet another example, a common device now found in many computer systems is a spool device, or spooler, which captures output intended for a printer or similar device, and stores it in a buffer until the device is ready to accept it. This is another type of virtual device that improves system performance and usability because users do not have to wait for a printer to finish before being able to use the computer for another task.

9.6 I/O INTERFACES

Processors cannot be expected to understand the detailed characteristics of all possible I/O devices, and a typical device may be used with many different types of processors. Linkage between a processor and a device must be provided by a hardware unit called an I/O interface.

The interface commonly takes the form of a circuit card to be plugged into the processor, and is designed to suit both the specific processor and the specific device. It may smooth out and simplify many of the details of device control. The task of a device driver is actually to program and monitor the interface, rather than the device itself.

Simple Interfaces

Most of you will quickly become familiar with the terms **serial interface** and **parallel interface** when working with microcomputers. These simple interfaces help transfer data to devices such as printers and terminals, one character at a time. The serial interface transmits the bits in a character on a single wire one bit at a time, while the parallel interface transmits all the bits at once over multiple wires. An example of a simple serial interface connection is shown in Figure 9-8.

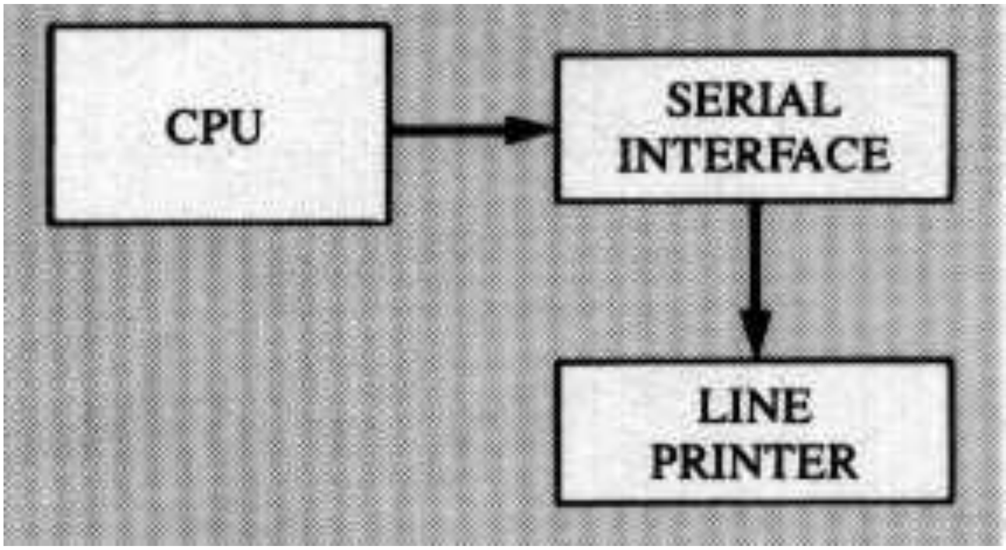

Figure 9-8: A Simple Interface for Character-by-Character Transfer

I/O Controllers

More complex I/O interfaces are often called **I/O controllers**, or I/O control units. Controllers provide features such as:

- direct memory access (DMA) data transfer, which allows for transfer of data between memory and a device without program intervention
- support of multiple devices (for example, multiple disk units)
- support of overlapped control operations, such as seek operations on disks, which can be initiated on one device while another device is performing another control operation.

Some controllers even provide for reading or writing on one device while one or more other devices are performing a control operation. Using such a controller, only one unit attached to the control unit can be busy transferring data at a given instant in time. In other words, the control unit can only select and control one device at a time for data transfer.

Programming controllers can be more complex than programming simple interfaces, because of the multiple devices that must be supported on one controller. A device driver will support a single controller, but must be able to keep track of the control and status of each device attached to the controller. Figure 9-9 illustrates a controller used to control multiple devices.

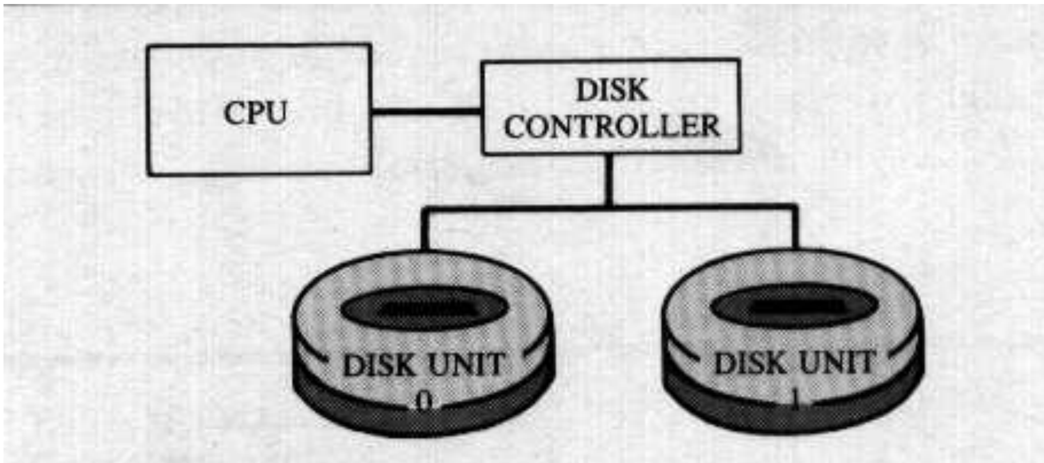

Figure 9-9: An I/O Controller Controlling Several Devices

I/O Processors

Another mechanism for device management, common on large mainframes, is the use of I/O processors. In the context of IBM mainframes, such a processor is called a channel. Multiple I/O controllers can be attached to a single channel. Channels function like special-purpose CPUs, with their own machine language. They can execute a series of I/O commands contained in a channel program, which in the 370 architecture resides in main memory. The address of this channel program is sent to the channel by a start I/O instruction. The channel command words (CCWs) in the channel program are read from memory using DMA techniques; the channel executes these commands using the selected controller. This obviously adds another complexity to the implementation of device drivers in such an environment. Three interacting physical devices must be considered:

- the I/O channel (which executes channel programs and controls data transfer between controllers and main memory)
- the I/O controllers (which control individual devices)
- the I/O devices (which actually perform appropriate input, output, and control operations)

FOR FURTHER READING

An overview of optical storage technologies is given by Asthana[1994].