Osborne/McGraw-Hill

THE OSBORNE/McGRAW-HILL

CP/M®

USER GUIDE

THIRD EDITION

For all 8-Bit CP/M® computers

Thom Hogan

The Osborne/McGraw-Hill CP/M® User Guide

Third Edition

by Thom Hogan

Osborne/McGraw-Hill
Berkeley, California

Published by
Osborne/McGraw-Hill
2600 Tenth Street
Berkeley, California 94710
U.S.A.

For information on translations and book distributors outside of the U.S.A., please write to Osborne/McGraw-Hill at the above address.

THE OSBORNE/McGRAW-HILL CP/M® USER GUIDE

1234567890 DODO 8987654

ISBN 0-88134-128-2

Karen Hanson, Acquisitions Editor
Ralph Baumgartner, Technical Editor
Ellen Guethlein Silge, Copy Editor
KLT van Genderen, Text Design
Yashi Okita, Cover Design
Photography by Richard Cash and Harvey Schwartz

An earlier version of this book was reviewed for technical content by William Fairman of Faircom and Doug Huskey of Digital Research. Technical editors were Curtis A. Ingraham and Martin McNiff. Compositional editor was Denise E.M. Penrose.

Hogan, Thom, 1952-

The CP/M User Guide: For all CP/M-80 and CP/M-PLUS Users
Bibliography: p 267
Includes index.
1. CP/M (Computer program) 2. Operating Systems (Computers)
I. Title. II. Title: CP/M user guide

This book is dedicated to Lore Harp, Carole Ely, Steve Jobs, Steven Wozniak, Gary Kildall, and Seymour Rubenstein, who gave me the tools to write it.

Introduction

Your computer is not a single unit but an interrelated system of devices and programs. You must direct these components to carry out any program you wish to run. CP/M-80 and CP/M-PLUS are operating systems that do much of this job for you. They direct the activities of your computer's components and manage files that contain computer instructions or data.

Although CP/M-80 and CP/M-PLUS are complex computer programs, you can learn to use them even if you do not have any prior computer experience. This book introduces the novice computer user to the microcomputer system and examines CP/M's function within that system.

Chapters 1 and 2 provide the basic, practical information you need to get started, and they present a history of CP/M's development. Chapter 3 rounds out the fundamentals you need to know to use CP/M. Chapters 4 through 6 detail the CP/M-80 and CP/M-PLUS commands. Chapter 7 describes how other programs you purchase (or create) relate to CP/M; you'll be introduced to the concept of computer languages and application programs in this section. This is information you will use every day, so we recommend that you study the examples carefully. The beginning seven chapters of this book provide a solid foundation for understanding what CP/M is and how to use it.

Beginning with Chapter 8, the book goes beyond the functions normally needed by business or casual computer users and starts a discussion of emphasis on how these relate to assembly language programming. Chapter 11 combines the information presented in the previous three chapters to provide a step-by-step example of how to program in assembly language using CP/M.

The last chapter, Chapter 12, discusses CP/M's relatives: how they differ, what they do, and why you might be interested in them. An annotated bibliography provides directions for additional reading, and several other appendixes offer practical consumer information about CP/M-compatible programs, languages, and products.

Some special introductory comments regarding this revised edition of the *Osborne/McGraw-Hill CP/M® User Guide* are in order.

The primary concerns in rewriting, restructuring, and adding to the text for this third edition were to make the book more up-to-date and to include CP/M's latest generation, CP/M-PLUS. As happens with many works, the author had second (and third) thoughts about how to make some sections clearer. The CP/M-PLUS commands and information have been integrated into the text in a manner that allows owners of either CP/M-80 or CP/M-PLUS to make equal use of this book. For readers interested in learning about 16-bit versions of CP/M, Osborne/McGraw-Hill publishes a user's guide specifically for those operating systems.

Much effort has been expended to make this revision the most complete and accurate manual for 8-bit CP/M users. If you have a working computer system, read this book while seated in front of your computer. Try the commands and examples presented here; do not just read about them. You will be comfortable with CP/M much sooner than you expected.

One last comment: The terms CP/M, CP/M-80, and CP/M-PLUS are used in this book with specific meanings. Whenever we refer to CP/M, we are writing about *all* versions of CP/M. However, whenever we refer to CP/M-80 or CP/M-PLUS, we are referring to specific (and different) versions of the operating system.

Books, like computer programs, are never completely error-free. The author and publisher invite your comments and criticisms.

This book is the work of the author and the publisher; it has not been reviewed, authorized, or endorsed by Digital Research.

th
Palo Alto
1984

Contents

CHAPTER

1 Computers, CP/M, and Operating Systems

This chapter presents the basic information that allows you to understand what an operating system like CP/M is, why it is necessary, and how it relates to the specific parts of the computer. You'll also learn a few computer terms and information about the diskettes used by CP/M.

CP/M is a disk operating system for microcomputers produced by a company named Digital Research. When CP/M was first introduced, its creator indicated that the initials stood for "Control Program/Monitor." Since then, however, it has become more popular to refer to it as "Control Program for Microcomputers."

Versions of CP/M are available for a wide variety of microcomputers from a number of different sources, and the introduction of new versions of CP/M can be expected when new central processors are invented. CP/M-80 and CP/M-PLUS can be used on almost any microcomputer that uses the 8080, 8085, or Z80 central processor unit and has 8-inch or 5 1/4-inch floppy disk drives. If the numbers confuse you or the term *central processor* has no meaning to you, read on; the concepts should be clear to you by the time you finish this chapter.

THE FUNCTION OF CP/M WITHIN A MICROCOMPUTER SYSTEM

Before starting to use CP/M, it is important to understand the functions served by CP/M within a microcomputer system. If you know what is going on and why CP/M is necessary, you are less likely to make mistakes.

In this section, we describe the function of CP/M (or for that matter, any operating system) within a computer system. This description assumes an elementary understanding of microcomputers and how they function. If you need more information about microcomputer systems, see *An Introduction to Microcomputers, Volume 0, The Beginner's Book* by Adam Osborne (Osborne/McGraw-Hill, Berkeley, California, 1978).

A microcomputer system is illustrated in Figure 1-1. The system shown is typical of configurations that you may encounter. It includes the microcomputer itself, a keyboard, a video display, a pair of disk drives, and a printer.

Numerous permutations abound. You might have a single integrated terminal (keyboard and display) or a display separate from the keyboard. The keyboard could be part of the microcomputer while the display is separate, or the keyboard, display, and microcomputer may be packaged together. For the purposes of this book it does not matter how many "boxes" the basic components come in, just as long as you have the basics.

Figure 1-2 is a functional diagram of the components of a typical microcomputer.

Microcomputers spend much time transferring information between the microcomputer and various other components of the system. This process of transferring information is called "I/O" (short for Input/Output) and is done through "ports" (the channels through which the input and output must be routed). A disk controller is simply a specialized I/O port, but we show it as a separate functional block in Figure 1-2 because it is a unique type of I/O device.

FIGURE 1-1. Typical microcomputer system

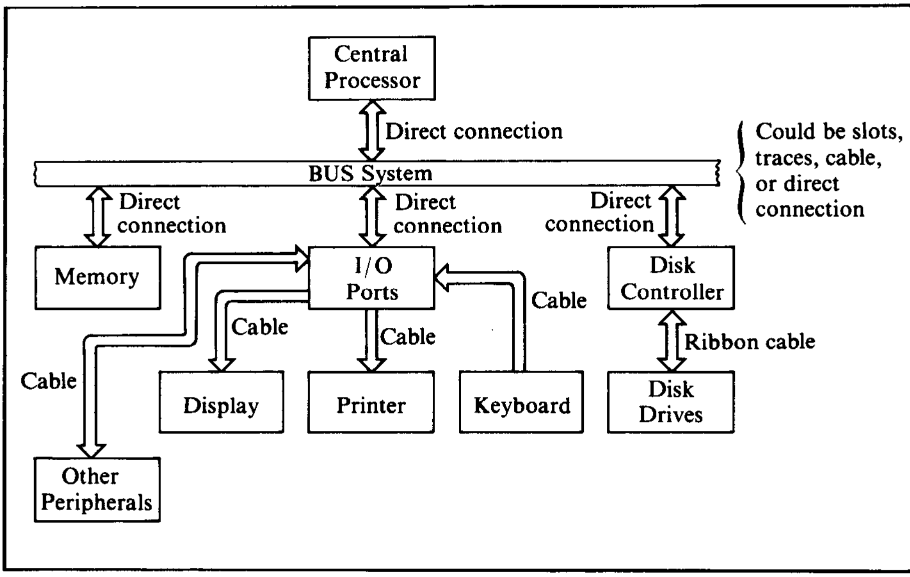

FIGURE 1-2. Functional diagram of a typical microcomputer

How does a microcomputer know what components are present and how to control them? The answer is that a set of instructions, called an *operating system,* is necessary for the microcomputer to function.

CP/M is such an operating system. By using appropriate CP/M commands, you can transfer data from a diskette to the microcomputer, print data from memory on a printer, or perform any operation that the microcomputer system is physically capable of handling.

In order to perform these microcomputer system functions for a wide variety of different configurations, CP/M (like most other operating systems) ignores the physical components that make up the microcomputer system, dealing instead with logical components. In other words, rather than addressing a printer, the operating system assumes that a "listing device" is present. Likewise, rather than reading information directly from a modem, the operating system assumes that the input comes from an "auxiliary device."

The manufacturer of your microcomputer system usually ensures that the system's actual physical units connect properly to the logical units CP/M uses. If you put together a system by "mixing and matching" components from several manufacturers, you may have to make some changes to CP/M yourself. These program modifications are invisible if made correctly, but if the proper changes are not made, CP/M may work inaccurately, if at all.

As an operator of a microcomputer system, you may occasionally be concerned with physical as well as logical units. For example, you may have the option of sending program output to a specific printer or display. Likewise, you may have the option of typing input at the keyboard or of telling the computer to receive input over a telephone line from a remote terminal. You can make such physical unit choices easily using the appropriate CP/M commands described later in this book. For the moment, you need only understand the general function of CP/M or of any operating system. You do not need to understand CP/M's specific activities in order to use it.

An operating system like CP/M is itself a computer program that must be executed by a microcomputer. Being a program, CP/M must be written in a programming language. The programming language that a microcomputer understands is determined by the microprocessor (sometimes called *central processing unit,* or *CPU* for short) that the microcomputer contains.

A microprocessor is a very small and unassuming device; Figure 1-3 shows a typical microprocessor. The microprocessor is a microcomputer's most important component—it actually translates the instructions that constitute a program and causes the action associated with each instruction. Some microprocessors can execute CP/M, but others cannot.

CP/M was initially written for the 8080 microprocessor designed by Intel. Since the later 8085 and Z80 microprocessors also execute 8080 instructions, CP/M-80 and

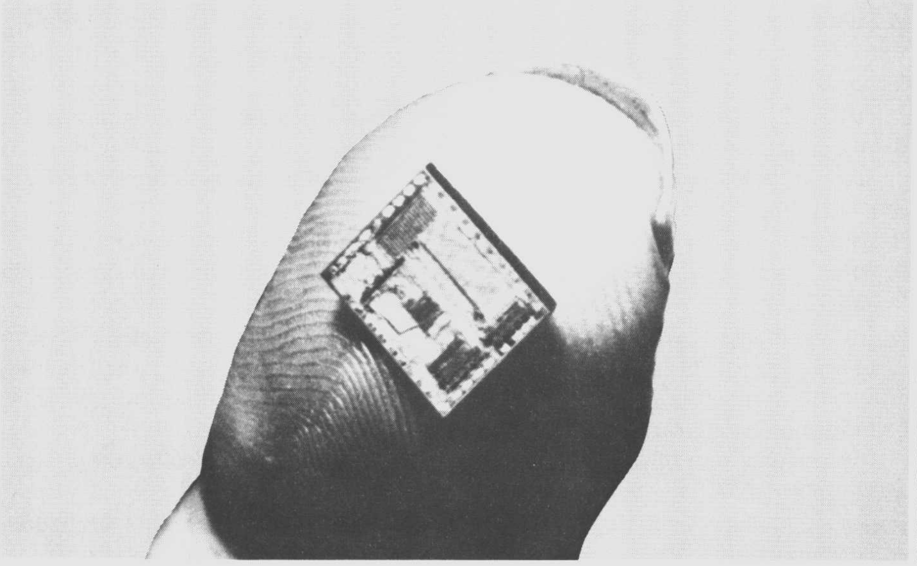

FIGURE 1-3. Typical microprocessor

CP/M-PLUS run on microcomputers containing either of these two microprocessors. In addition, in early 1981 Digital Research introduced CP/M-86 for the 8086 and 8088 microprocessors. A fuller discussion of the different types of CP/Ms appears in the next chapter.

SOME USEFUL TERMS

Like most disciplines, computing has a vocabulary of its own. So that we can be as precise and clear as possible throughout the rest of this book, we will present a short glossary of the more common terminology we will be using. Specifically, we want to make sure that you understand the "units" of information we discuss. When one talks about language, one uses the terms *words, sentences, phrases,* and *paragraphs.* Similarly, when one talks about computers, one talks about *bits, bytes, sectors,* and *tracks.* If you are familiar with computers and the difference between bytes and bits, and so on, you might wish to skip ahead to the next section. Newcomers to computing should study the following information carefully.

BIT

A bit is the smallest piece of information a computer can maintain. All of a computer's data is stored internally as a series of 1's and 0's. A single bit is a single 1 or 0. You will encounter this term later in this book when the technical details of CP/M are discussed.

BYTE

The storage capacity of a microcomputer's memory, or its disks, is always described as some number of bytes. A byte is a memory unit capable of storing a single character. For example, the letter A or the digit 1 could be stored in one byte of memory. Numbers without decimal points (termed *integers*) are usually stored in two consecutive bytes of memory, while numbers with decimal points (termed *floating point numbers*) are stored in four or more consecutive bytes of memory per number.

Memory size is usually expressed not as thousands of bytes but as some number of *K* bytes. 1K equals 1024. All computers are binary machines; in other words, they count in twos. For example, you get the number 1024 if you double 2 to give 4, then double 4 to give 8, and keep on doubling in this fashion ten times.

A byte, by the way, consists of eight consecutive bits. Since a bit can contain only a 1 or a 0 value, a little binary arithmetic soon shows you that a byte has 256 possible values. In most microcomputers each possible byte value is assigned to one symbol, letter, or digit. A "1," for instance, is stored by the computer as a byte value of 49, while an "A" is stored by the computer as a byte value of 65. The "code" that determines what value is assigned to what letter, digit, or symbol is called the *ASCII* code (which stands for American Standard Code for Information Interchange—see Appendix A).

TRACK

When information is stored on a cassette tape, it is stored as a single track of data down the length of the tape. When information is stored on a disk, the surface of the disk is a series of concentric circles called *tracks*. The outermost circle is referred to as track 0; while on standard 8-inch CP/M disk systems, the innermost concentric track is called track 76.

SECTOR

Each concentric track of information on a diskette is further subdivided into units called *sectors*. On standard 8-inch CP/M disk systems each sector stores 128 bytes (characters) of information, and there are 26 sectors on each track. The smallest unit of information that CP/M manipulates on a diskette's surface is one sector. Manipulations of a single byte at a time are done in the computer's memory and not directly on the diskette.

Figure 1-4 illustrates the concept of sectors and tracks.

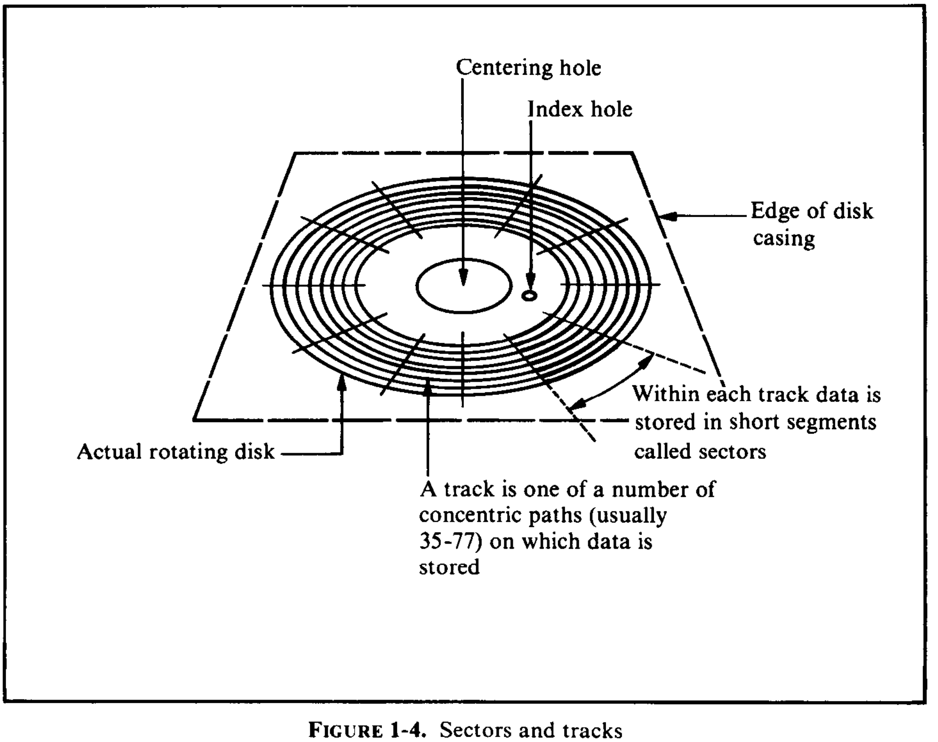

FIGURE 1-4. Sectors and tracks

DISKETTES

There is more to diskettes than just tracks and sectors, however, and it is important that you understand as much about diskettes as possible, since so much of CP/M's use revolves around them. Let us take a closer look at the terminology of diskettes.

A floppy diskette is the flimsy removable medium used by a disk drive, which is a mechanical device used to write information to and read information from diskettes. Disk drives that utilize polished metal "platters" instead of removable floppy diskettes are *hard disk drives* (because the medium is hard). For the sake of conciseness, we will normally abbreviate floppy diskette to "disk," disk drives to "drives," and hard disk drives to "hard disks."

The backbone of any disk operating system is, of course, the disk itself. It may be difficult to comprehend that 150 single-spaced typed pages (the equivalent of 30 hours of typing) can be stored on one flimsy magnetic disk. Unfortunately, one can easily forget to take the proper steps to protect the information stored on the disk. Before discussing detailed information about CP/M, we will pause for a quick course on disk types, care, and usage.

Comparing Disks

Walk into a computer store and ask for a disk, and the salesperson will ask you what kind you need. For an idea of the possibilities, here are some features to consider:

- 8-inch versus 5 1/4-inch disks
- Single-sided versus double-sided disks
- Single-density versus double-density disks
- Soft-sectored versus hard-sectored disks
- 10-sector versus 16-sector hard-sectored disks
- Write-protect notch versus no write-protect notch.

This is a confusing array of choices, and there are as many brands as there are types. We could not possibly list all of the combinations. Diskette manufacturers publish long lists of compatible diskettes, computers, and disk drives. Check with any reputable computer store to learn which diskettes to use with your microcomputer. Better yet, check with the vendor who sold you your computer.

A brief summary of the more popular types of diskettes and microcomputers is provided in Appendix B.

Describing Disks

Disks, as mentioned previously, are flimsy. That is why they are sometimes called "floppies." If you have an extra disk around, you may want to get it, as we are about to take a "disk tour."

You notice first that the disk is accompanied by a paper envelope. This envelope protects about two-thirds of the disk from such data killers as dirt, liquids, and thumbprints. Since it is a thin envelope, it provides limited protection, so be careful. Many disk manufacturers print handling tips on the back of the envelope. Read any such information; someday it may mean the difference between retyping for five hours and spending a relaxing evening at home.

Carefully pull the disk out of the envelope. (**Note:** It slides right out. If it looks as if you must cut something open to get inside, you are mistaking the disk "sleeve" for the envelope.) The disk has a square vinyl sleeve that protects a thin circular disk. The sleeve has a circular hole in the center (as does the disk surface inside), and there is an oblong cutout at one edge of the sleeve (see Figure 1-5).

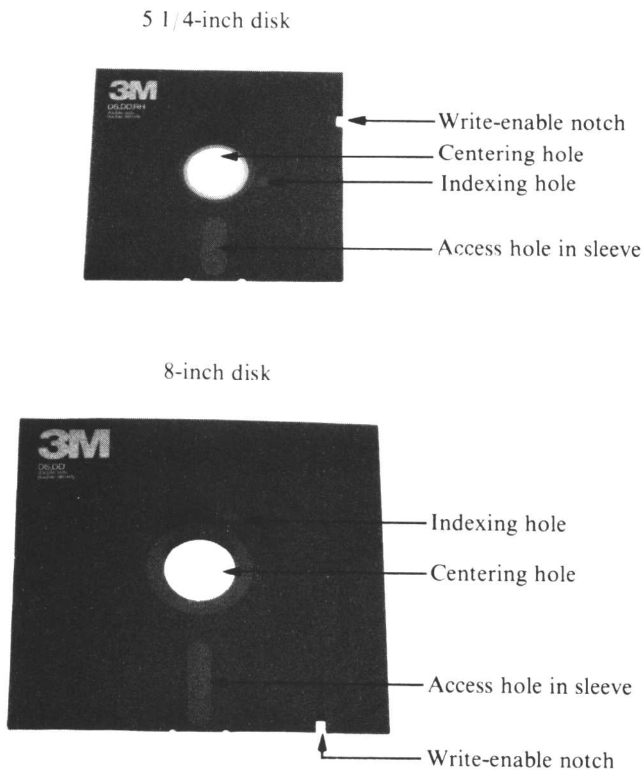

FIGURE 1-5. 5 1/4-inch and 8-inch disks

Also, there is a smaller hole just to one side of the central hole.

Identify these parts:

CENTERING HOLE

The disk drive mechanism locks onto this hole to spin the disk inside the sleeve.

INDEXING HOLE

The disk drive looks here to find the starting sector (and, in the case of hard-sectored disks, each individual sector) for each track on the disk. Imagine a line drawn across the disk surface at this point; the drive waits for this starting line and then counts characters of information from there.

ACCESS HOLE

The head of the disk drive comes in contact with the magnetic surface through this cutout. The head moves back and forth in this opening, from track to track. Note that there are access holes on each side of the disk.

NOTCH

This is a write-protect notch. Writing on the disk means adding information to the disk.

A point of confusion arises here: On 8-inch disks, if the write-protect notch is covered up, you can write on the disk; if the notch is left uncovered, you cannot write on it. On 5 1/4-inch disks, if the notch is uncovered, you can write on the disk; if the notch is covered, the disk is protected against writing. (You may want to circle the appropriate section of this paragraph for future reference.)

To compound the problem further, the write-protect notch on 8-inch disks is located at the bottom of the disk, while that on the 5 1/4-inch disks is located on the side of the disk (see Figure 1-5).

As if this were not enough, some manufacturers buy disks that have no write-protect notch at all. Software vendors who sell programs on 5 1/4-inch media usually buy such disks so that you can never make any changes to the original disk you receive.

Handling Disks

Now that you are familiar with the parts of the disk, it is time to learn how to handle it.

First the don'ts:

Never touch the disk surface. You may handle the vinyl sleeve when necessary, but do not touch the actual magnetic surface of the disk. No matter how clean your hands appear, even a slight residue may prohibit your computer from reading some data.

Keep disks away from magnets. Silly instruction, you say? You think you have no magnets where you work? Think again. If your computer has a video terminal (one