

THE EPSON QX-10 USER'S GUIDE



James M. Hansen

The Epson QX-10 User's Guide

The Epson QX-10 User's Guide

James M. Hansen

Scott, Foresman and Company
Glenview, Illinois
Dallas, Texas Oakland, New Jersey Palo Alto, California
Tucker, Georgia London

Dedicated to Pat

Copyright © 1984 Scott, Foresman and Company.
All Rights Reserved.
Printed in the United States of America.

ISBN 0-673-15973-6

Library of Congress Cataloging in Publication Data

Hansen, James M., 1943-
The Epson QX-10 user's guide.

Includes index.

1. Epson QX-10 (Computer) I. Title.
QA76.8.E6H36 1984 001.64 84-2148
ISBN 0-673-15973-6

123456-MVN-898887868584

Technical Illustrations by Bob Aviles
Photography by Jim Hansen, unless otherwise noted

Contents

	Introduction The QX-10 and the Commodore Perry Connection	1
Chapter 1	A Brief History of the Epson Corporation	5
Chapter 2	A Walk Around the QX-10	14
	The Quest for the Standard Personal Computer	14
	The Standard Computer Emerges	15
	Enter the QX-10	15
	The QX-10 Keyboards	16
	The Main Typing Area	19
	The Editing Clusters	22
	The Numeric Keypads	24
	Function Groups	25
	Keyboard Interconnect	25
	The Main Electronics Case	26
	The Disc Drives	26
	System Reset Switch	26
	Main Power Switch	28
	The Back Apron	28
	The QX-10 Monitor Screen	30
	A Summary of Technical Specifications for the QX-10	31
Chapter 3	Inside the QX-10	32
	Part One: An Overall Survey of the QX-10	33
	Inside the Keyboard	34
	The Main Electronics	35
	The Power Supply and Fan	38
	The CRT Monitor	39

Part Two: The QX-10 Electronic Design 40

Main Electronics Board	40
Interfaces	43
<i>Serial Communications and the RS-232C Interface</i>	43
<i>Serial Data Transmission</i>	44
<i>The Centronics Parallel Printer Port</i>	45
<i>Parallel Data Transmission</i>	47
<i>The Light Pen Input</i>	48

Chapter 4 Operating Systems on the QX-10 51

CP/M	52
TPM II	52
Valdocs	53
CP/M Technical Details	53
CP/M Utility Programs	55
CP/M Transient Utility Programs	55
Memory Allocation on the QX-10	57
TPM II Technical Details	59
TPM II Resident Utility Programs	59
User Areas	60
TPM II File Protection	61
TPM II Transient Utilities	62
Comparing CP/M and TPM II	63
The Floppy Disk	63
Disk Formats	64
Care of Floppy Disks	66

Chapter 5 Applications Programs on the QX-10 67

Application Program Installation—A Few Notes	68
Sources of Applications Programs	68
PeachTree Software Available Through Epson	69
Text Editing—PeachText	69
<i>Editing Commands</i>	70
<i>Thesaurus Operation</i>	71
<i>Headers and Footers</i>	71
<i>Printing Commands</i>	71
<i>Overall Commentary on Peach Text</i>	72

PeachTree Spelling Proofreader	72
<i>Running the Spelling Proofreader</i>	73
<i>Dictionary Maintenance</i>	73
<i>Overall Commentary on Spelling Proofreader</i>	74
PeachCalc an Electronic Spreadsheet	74
<i>The Spreadsheet</i>	75
<i>General Comments on PeachCalc</i>	76
Mailing List Manager	76
<i>Mailing List Operation</i>	76
<i>Maintaining Data Files</i>	77
<i>Mailing Labels</i>	77
<i>Comments on Mailing List Manager</i>	78
CP+, or Making CP/M More Friendly	78
<i>CP+ Organization</i>	78
<i>General Comments About CP+</i>	79
Applications Programs Integration	79

Chapter 6 Valdocs 81

The HASCI Concept	83
Valdocs Versions	85
Upgrading Valdocs	86
Valdocs Operation	86
The Valdocs Editor	86
Editor Screen Display	87
Editor Operation	88
Block Graphics	89
Editor Menus	89
File Backup and Other Considerations	90
Data Recovery Procedures	90
The HASCI Keyboard	91
<i>The Main Typing Area</i>	92
Valdocs Editing Menus	95
Valdocs Operating Controls	97
System Controls	97
File Controls	98
Valdocs Applications	100
Valdocs—An Overview	101

Chapter 7	BASIC on the QX-10	102
	The Dartmouth Connection	102
	Microsoft BASIC	104
	BASIC for the QX-10	105
	Some Applications Utilities Using BASIC	107
	A Caution	107
	Redirecting LPRINT Output	108
	Outputting a Byte Directly to the Printer	110
	Serial Baud Rate Control	110
	Reading the Date and Time from MBASIC	112
	The QX-10 Speaker	114
Chapter 8	Peripherals for the QX-10	117
	Printers	117
	The Transtar 130 Daisy Wheel Printer	118
	Dot Matrix Printers	120
	<i>Integral Data Systems P 80 Printer</i>	122
	<i>The Epson FX-80</i>	123
	MODEMS for the QX-10	124
	<i>Hayes Smartmodem</i>	127
	<i>Epson CX-20 Acoustical Coupler</i>	128
	Votrax Speech Synthesizers	129
	<i>Graphic Plotters</i>	130
	Other Peripherals	132
Chapter 9	Expansion Cards and Accessories for the QX-10	133
	Hard Disks for the QX-10	133
	Hard-Disk Technical Matters	134
	File Backup	135
	Comrex CR-103 Communicator Modem Card	136
	Epson Serial Interface Expansion Card	137
	Epson IEEE Expansion Interface Card	137
	Epson Optical Fiber Expansion Interface Card	138
	Other Expansion Cards	138
Epilog		140
Appendixes		141

Acknowledgments

Special thanks are due many individuals without whom this book would not have been possible, especially Chris Rutkowski and Barbara Elman of Rising Star Industries, Bob Adam, Linda Blasnek, John Morgan and Bob Walker at Epson America, Marc Marlow of C & C Computer, and the folks at the Bohle Company who endured incredible harassment during the preparation of this manuscript.

Silvio Bedini, at the Smithsonian, who was of great assistance in researching the Seikosha origins, located the Japanese woodcuts made during the Perry visit to Japan. This book is one of the few to print them publicly in the United States.

Personal appreciation goes out to Pat Hansen, my wife, who spells far better than I and helped with the final preparation of this manuscript. And last, but certainly not least, a thank you to Mr. Kintaro Hattori, through whose efforts this is a book of fact, rather than fiction.

Jim Hansen

The Epson QX-10 User's Guide



Introduction

Any story about Epson's QX-10 computer would not be complete without briefly mentioning the contribution of Commodore Matthew Calbraith Perry, USN, 1794–1858, the father of the steam navy. Commodore Perry, you see, was sent on a diplomatic mission to Japan in 1852 and opened the country for international trade. This was no mean feat.

The Japan visited by Commodore Perry was vastly different from any other country at the time. Under the rule of the Tokugawa family, which assumed power in the early 1600s, Japan was closed to the outside world. (Some Dutch and Chinese traders were allowed into the port city of Nagasaki, but the rest of Japan had experienced virtually no contact with the world community since 1638, when foreign missionaries were expelled. Christianity was suppressed and foreign travel from Japan was banned on pain of death.)

Japanese society, protected from external influence for over 200 years, was a feudal system based on four classes. In order of privilege and social rank, they were the samurai,

Commodore Galbraith Perry as seen through the eyes of a contemporary Japanese artist in 1853. This photograph of the original woodcut drawing from the collections of the National Museum of American History, Smithsonian Institution, Washington, D.C.

peasants, artisans (craftsmen), and merchants. This system stemmed from the philosophy of Kung-fu-tse (also known as Confucius), the Chinese philosopher of about 500 BC. During this period Japan enjoyed a very rigid but peaceful and stable society.

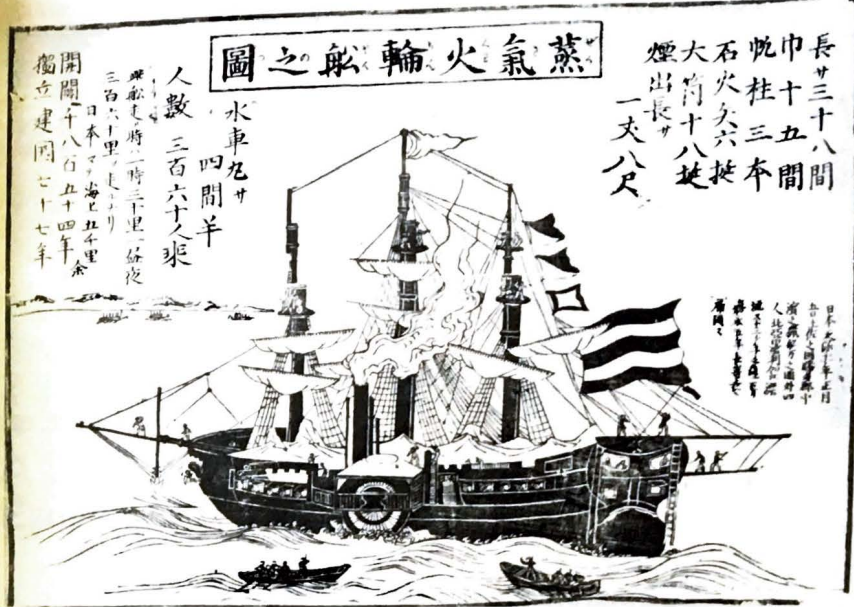
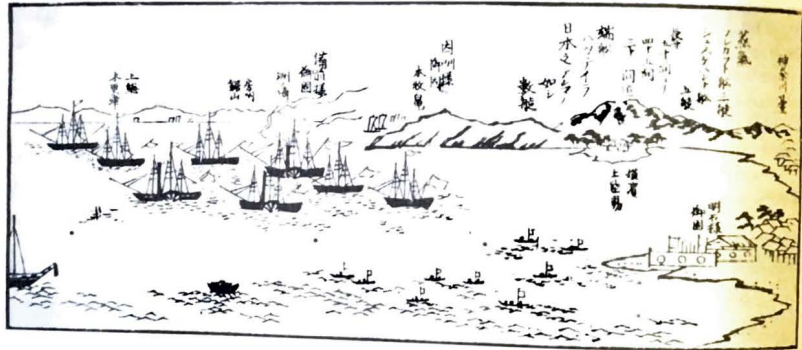
Commodore Perry arrived in Japan at the Bay of Yedo on July 8, 1853. On July 14, a letter to the Japanese Emperor from President Fillmore was delivered to commissioners of the Shogun. (It was enclosed in a gold box worth about \$1000 at the time.) Perry then left Japan under the pretext of diplomatic convenience, but actually to attend to matters concerning the Taiping Rebellion in China.

After finding that French and Russian admirals were soon planning to visit Japan, Perry returned to Yedo Bay the 13th of February, 1854, much earlier than he had desired. Since the location of Perry's previous visit was out of the range of his naval guns in the bay, and evidently feeling insecure, he attempted to move the negotiations to Yedo. The Japanese refused, and a compromise was reached to hold them in the suburb of Kanagawa.

In the Treaty of Kanagawa, signed on Friday, May 31, 1854, Japan agreed to international trade, opened the ports of Shimoda and Hakodate to American ships with an American Consulate in Shimoda, and established a few other general trading agreements. In return, Japan was given most-favored-nation status. Depending on who tells the story, the agreement was either negotiated with or forced upon the Japanese government.

In any event, political and social unrest stirred by Perry's visit ended the 250-year-old Tokugawa Shogunate (government) in 1866, when it was overthrown by samurai leaders. The Meiji Restoration, named after Emperor Meiji, who ruled after the uprising, completely changed Japanese society. Tokyo was soon established as the capital city, and the samurai class lost its privileges as the feudal system was rapidly torn down and replaced with a political and legal system capable of dealing with Western powers.

Perry's fleet in the Bay of Yedo. From the collections of the National Museum of American History, Smithsonian Institution, Washington, D.C.



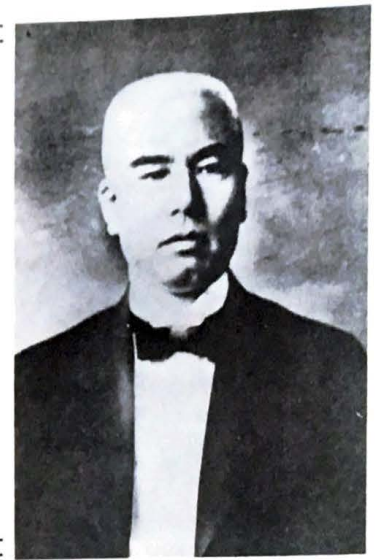
The *Powhatan*, Perry's sidewheel steam frigate. The Japanese reaction to Perry's visit ranged from shocked amazement at the etiquette of the "hairy ones" to thinly veiled hostility tempered by fascination of Western technology. This photograph of the original woodcut from the collections of the National Museum of American History, Smithsonian Institution, Washington, D.C.

And so it was in the midst of profound social change that the K. Hattori Trading Company was started in 1881 by the eldest son of an antiques salesman, to trade in clocks, watches, and music boxes. Today this company is still wholly owned by the Hattori family, but is certainly not your basic "Ma and Pa" store. K. Hattori owns three major corporations, which in turn hold others. Two of the smaller corporations are Epson Corporation (in Japan) and Epson America, Inc. (Epson America, Inc., in turn, owns Comrex, a subsidiary that sells computer peripheral products on a noncompetitive basis with Epson.)

The business ethics and practices used by these corporations reflect the personal integrity and standards of family pride that quite literally date back to the days of the samurai warriors, the Japanese equivalent of the American wild West. It is a matter of record, for instance, that all shipments of MX-80 printers to America were once stopped because somewhere in the United States a dealer had not paid a bill for \$47.

Commodore Perry did not live long after his visit to Japan—he died about five years later. But if he were alive today, he might be surprised that he had a hand in starting the company that makes his Seiko watch, the printer for his calculator, over 40 million mechanical “Swiss” watch movements a year, nearly 60% of all liquid crystal displays, and nine out of ten printer mechanisms world-wide, as well as the revolutionary QX-10 computer and Valdocs operating system first sold in the spring of 1983. Not bad for a sailor of the 1850s.

1



A Brief History of the Epson Corporation

After the Perry visit of 1853–1854, Japan began an accelerated program of modernization. Using Western models, Japan renovated its education, industry, and communications. The first Japanese railway opened in 1872 between Tokyo and Yokohama, and within a few years it and several others were operating at a profit.

In this same year, 1872, the ancient Japanese method of recording time was officially abolished, and European clocks and watches came into high demand. The first Japanese clock factory was started in 1875. By far the most successful of the early Japanese clock factories was the Seikosha factory, started in May of 1892 by the K. Hattori Company.

The K. Hattori Company started business in 1881 and began importing watches and clocks from the United States, principally from the Chicago area, in July of 1890. The founder of K. Hattori, Kintaro Hattori, was born in 1859, just five years after Admiral Perry departed Japan.

Kintaro Hattori, founder of Seikosha, one of Japan's earliest watch manufacturers, and the Hattori Trading Company, Ltd., which owns the “Seiko” group. He was born the year following Perry's death. Photograph courtesy Epson America Incorporated.



"Time Keeper" pocket watch, size 20, produced in 1895. This cylinder movement watch was among the first produced in Japan. Photograph courtesy Katsuhiko Sasaki, Tokyo National Science Museum.

In May of 1892, Hattori established the Seikosha factory, which produced about a dozen wall clocks in July of 1892. By December of 1893 Seikosha employed 90 workers, manufacturing wall clocks and metallic watch cases. The factory was powered by a five-horsepower steam engine.

A year later (December of 1894), employment was up to 173 workers and over 24,000 clocks were produced. In 1895, Seikosha began to export clocks to China, and at the same time began making the first all-Japanese watches, the "Time Keeper" series. These pocket watches featured the cylinder movement common in watches of the era.



"Seikosha" size 17 pocket watch manufactured in 1897. This was the first "Seiko" watch produced. The Japanese trend toward miniaturization is already evident. (Notice how much smaller it is than the Time Keeper in the previous photograph. Both watches are shown at 64% of original size.) In 1897 Seikosha employed 333 people and produced 63,720 watches and clocks. Photograph courtesy Katsuhiko Sasaki, Tokyo National Science Museum.



"Time Keeper" pocket watch, size 20, produced in 1895. This cylinder movement watch was among the first produced in Japan. Photograph courtesy Katsuhiko Sasaki, Tokyo National Science Museum.

In May of 1892, Hattori established the Seikosha factory, which produced about a dozen wall clocks in July of 1892. By December of 1893 Seikosha employed 90 workers, manufacturing wall clocks and metallic watch cases. The factory was powered by a five-horsepower steam engine.

A year later (December of 1894), employment was up to 173 workers and over 24,000 clocks were produced. In 1895, Seikosha began to export clocks to China, and at the same time began making the first all-Japanese watches, the "Time Keeper" series. These pocket watches featured the cylinder movement common in watches of the era.



"Seikosha" size 17 pocket watch manufactured in 1897. This was the first "Seiko" watch produced. The Japanese trend toward miniaturization is already evident. (Notice how much smaller it is than the Time Keeper in the previous photograph. Both watches are shown at 64% of original size.) In 1897 Seikosha employed 333 people and produced 63,720 watches and clocks. Photograph courtesy Katsuhiko Sasaki, Tokyo National Science Museum.

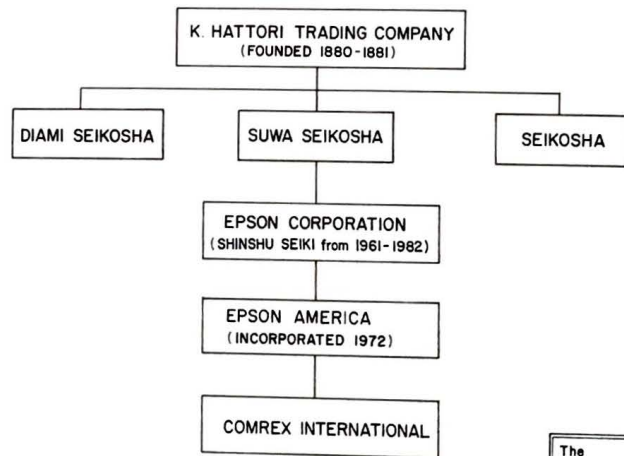
By the turn of the century, the factory was using a 60-horsepower steam engine, employed around 400 workers, and manufactured over 100,000 watches annually, some of them with as many as 17 jewels, lever escapements, and bimetallic balance wheels, which was the state of the art in 1901.

In 1917, the company name "K. Hattori" was changed to "K. Hattori Company, Ltd.," and Seikosha became "Seikosha Company Ltd." Today, after more than 100 years of operation, the Hattori Company is still family owned and has no outside stockholders. Precision mass production is actually the Hattori specialty. The smaller, finer, more precise and beautiful, the better. In fact, one translation of "Seikosha" could be "Precision Manufacturing Company."

Although many of the corporations owned by the Hattori family are still involved with the manufacture of time-keeping instruments, a music box line started in 1902 has long since been dropped. Most of the world's supply of "Swiss" watch movements are made by Seikosha. (These mechanical movements are installed in cases by other manufacturers and sold as "Swiss" watches.) All the watches sold under the "Seiko" trade name are made by various Hattori corporations.

The three parent corporations (Seikosha, Diami Seikosha, and Suwa Seikosha) held by K. Hattori have branched into many areas besides mechanical and electrical timekeeping. None of these corporations share management, money, planning, or product, and it is not unusual to find one competing directly with another in some market areas and cooperating in others.

Organizational chart of the K. Hattori Trading Company.



The Epson Family Tree

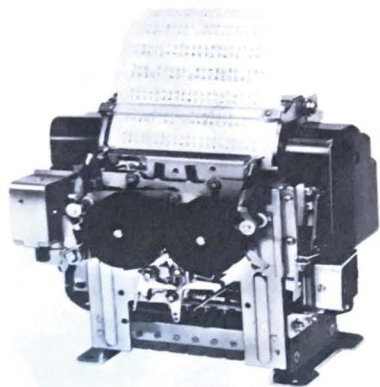
The main product of Seikosha is mechanical watches. In some product lines, hole tolerances are held to 5 microns (millionths of a meter) or less—true precision.

Seikosha also makes electronic and computer products and imports some printers into the United States. Several models have been introduced, but none has been as successful as those from Epson. The latest printer from Seikosha is imported by Transtar and is known as the Transtar 315 Color Graphics Printer. It prints about 50 characters per second and up to seven colors on a single pass.

Diami Seikosha makes watches that are mostly sold in the world market under the trade name "Seiko." (There is no actual Japanese company named "Seiko." Seiko is only a trade name used for marketing and subsidiary company names abroad.) Diami also builds an office computer and printer line sold in Japan and manufactures liquid crystal displays for everything from watches and calculators to computers and automotive dashboards.

Suwa Seikosha is the "high technology" member of the group. The quartz watch, originally developed here, was perfected over an eight-year period. Other products made by this corporation (and the eight other corporations owned by it) include TV watches, electric shavers, eyeglass lenses, CMOS integrated circuits, mechanical watches and parts, quartz crystal oscillators, cobalt magnets, and more. This company is of particular interest to us because it is the parent company of Epson Corporation.

The Epson Corporation was originally established in 1961 under the name Shinshu Seiki (Shinshu Precision Manufacturing Company) to provide precision watch parts for the Seiko watches. When the contract to build precision timers for the 1964 Tokyo Olympic



The EP-101 printer first made for use at the 1964 Tokyo Olympic Games. Photograph courtesy Epson America Incorporated.

games was awarded, Shinshu Seiki was asked to build a printer for them—Shinshu Seiki's first diversification.

The result, the EP-101, was a miniaturized, low-cost printer mechanism. It became the first practical printer for electronic calculators when it was placed on the market in 1968. Its small size, speed, and low power consumption have made it a continuing success—more than 100,000 of these printer mechanisms are still made each month, a phenomenal lifetime for this type of product.

The EP-101 brought Epson squarely into printing technology. Today Epson printer mechanisms are used in nearly every printing calculator made in Japan and elsewhere; there are over 15 "standard" printer mechanisms made to handle virtually every small printer application, in addition to a line of full-featured printers such as the FX-80 and FX-100 models. Epson can probably claim to make 90% of all printing mechanisms manufactured worldwide.

It was the EP-101 that gave Epson its name. Epson means literally "son of EP." For those interested in such nostalgia, a summary of specifications for the EP-101 can be found in Appendix A.

The first electronic watches using light-emitting diodes came on the market in 1970. The power requirements for LED displays made these watches inconvenient, since a button had to be pushed whenever the time was to be displayed. That year Shinshu Seiki started a four-year research program on liquid crystal displays, and in 1974 began commercial production. Practical, low-cost liquid crystal displays opened the door for the LCD watch, card-sized calculators, and wrist television sets (announced in the spring of 1983).

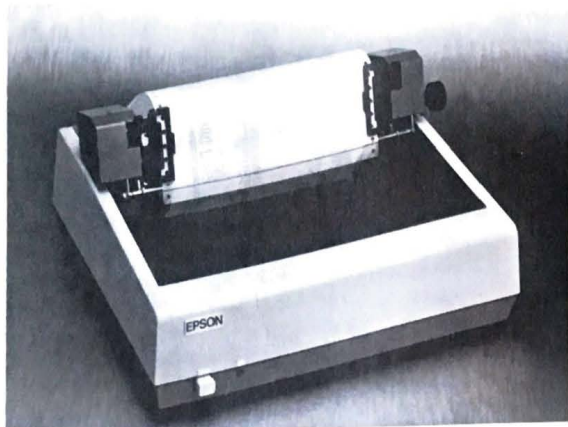
The Intel 8080, the first single-chip microprocessor powerful enough for major computing operations, went on sale in the United States during 1974. Although other microprocessors were available before this, notably Intel's 4004 and 8008, none had the right combination of speed, power, and electronic sophistication to make them practical for general computing applications.

Shinshu Seiki came to America the following year, in 1975, registering the trade name "Epson" and incorporating in California as Epson America. Offices were set up in Torrance, California, less than half a mile from Epson's present facilities. Epson America started with a staff of three. Yashuhiro Tsubota is presently the corporation President.

The year 1975 is special for microcomputerists as well. The January issue of *Popular Electronics* carried a home construction article describing the Altair computer kit. The Altair computer, probably more than any other single computer product, started the personal computer revolution.

As originally conceived, Epson America was to serve only as an OEM distributor of basic Shinshu Seiki wares, which, at the time, were printer mechanisms and liquid crystal displays. (OEM stands for "Original Equipment Manufacturer," meaning that they intended to sell only to other manufacturers.)

Epson's first dot matrix printer, the TX-80, was brought to market in 1978. The total development time for this 80-column printer was only three months. Later in the year, an



The TX-80 printer, Epson's first dot matrix printer export to the United States. This printer offered a seven dot high matrix (no lower case descenders), a solenoid-operated linefeed mechanism, and could print up to 58 lines per minute. Printing was strictly uni-directional, left to right. Photograph courtesy Epson America Incorporated.

eight-man design team began work on an improved version to be called the MX-80. The TX-80 was sold primarily to manufacturers. Although built with jewel-like precision and highly reliable (some say a bomb could be set off next to it and the bomb would fail), the TX-80 did not attract much attention in the United States. (The biggest customer for the TX-80 was probably Commodore, who used it as the system printer for the PET computer.)

Other events occurred during 1978 that would later influence the development of the QX-10 computer system. Technical Design Labs (TDL), an early personal computing company in New Jersey that specialized in S-100 computer hardware and sophisticated software to support these products, went out of business after achieving annual sales of about \$2.5 million.

That news was public. The news that was not publicized was the story of a computer system under development near the end of TDL's lifetime. It was called the "General," and through its development came the discovery of a phenomenon called "architectural stabilization."

The General was an advanced computer, but, even considering the cost of memory and other hardware available in 1978, would have been a fairly inexpensive system, about \$4000 retail. Although fully developed and ready for production, and with close to \$35 million in advance orders, the manufacturing startup costs were simply beyond the resources of TDL, and venture capital could not be raised. (The microcomputing business was considered "folly" by investors in those days.)

TDL's three founders, Chris Rutkowski, Roger Amidon, and Carl Galletti ceased operations in early spring of 1978. CDL, another New Jersey company, acquired software rights to many of TDL's products and still markets some of them.

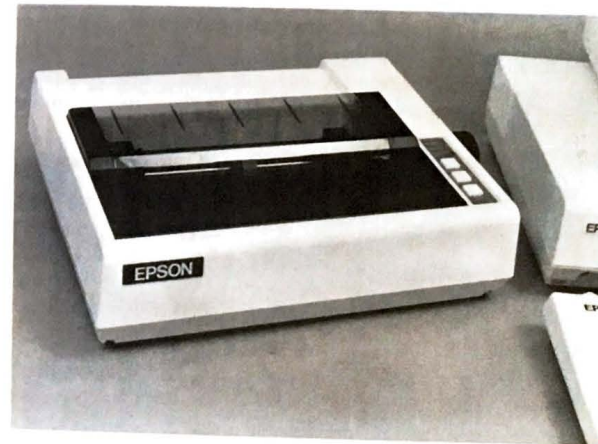
By January of 1980, Epson America had grown to employ 17 people. This was the year that Shinshu Seiki introduced the KX-1 desktop computer (marketed only in Japan) and the M-150 miniature dot matrix printer. In the fall of 1980, the now near-legendary MX-80 dot matrix printer, targeted for the world market, was introduced.

The MX-80 printer was not intended to be sold OEM; Shinshu Seiki had decided to enter the consumer marketplace with the Epson name. Chris Rutkowski, through a remarkable set of circumstances, was made Head of Marketing Research, and quickly established a dealer network of around 1000 stores during the summer of 1980. Advertisements were designed and placed in leading computing magazines.

The September, 1980, issue of *Byte* magazine carried the first full-page advertisement for the MX-80 on page 201. In the same issue, a mail-order dealer offered discounted prices of the Integral Data Systems "Paper Tiger" printer at \$895 and the Centronics 737 at \$849; another dealer offered the Epson TX-80 with graphics for \$795, and the MX-80, which had no graphics at the time, at the full retail price of \$645.

The MX-80 enjoyed tremendous acceptance. By the end of 1980, over 12,000 had been sold. By May-June of 1981, it captured first place as the best selling printer in the United States. Several factors combined to make the MX-80 popular: the printer was nicely styled, it was reliable, performance satisfied the market needs of the time, print quality was good, and the price was low compared to existing competition. But perhaps the most single important factor was distribution; the printer could be purchased almost anywhere.

The only feature lacking in the original MX-80 was graphics. This is a sensitive issue at Epson. One version of the story is that engineers in Japan refused to add graphics until someone in the United States disassembled the MX-80 firmware (read-only memories), patched in graphics code, and demonstrated it.



The MX-80 printer next to a QX-10 computer. This printer has become the world's most successful low-cost dot matrix printer. Until mid-1983, about 20,000 units a year were imported into the United States.

In any event, by one means or another, graphics capability was added to the MX-80. The September, 1981, issue of *Byte* carried the first Graftrax (as the graphics capability is called) advertisement. By this time, mail-order dealers were heavily discounting the Epson product line. One ad in the same issue of *Byte* listed the MX-80 with Graftrax at \$474; most others simply said, "Call for Price."

Development work on the QX-10 computing system began in midsummer of 1981. Members of the marketing arm of Epson American went to Japan with a 100-page marketing proposal, translated into Japanese, that outlined a product line of personal computers, suggested computer architecture, processor implementation, and suggested selling prices.

In November of 1981, a team of Japanese engineers arrived at Princeton, N.J., with an armload of hand-drawn schematics. There they met with three Epson employees: Chris Rutkowski, Roger Amidon, and Steve Semos. Various changes and modifications were suggested, and the first pass of the QX-10 electrical design was completed. The QX-10, except for modern high-density components and a few other minor details, bears a close functional resemblance to the General, which was abandoned at the collapse of TDL.

The conceptual design of the Valdocs software, used in the QX-10, was already under way. (More on it in Chapter 6.) The overall design of the HASCI (Human Applications Standard Computer Interface) keyboard design was already finished. A detailed description of it and the concepts behind Valdocs was published in the October and November 1982 issues of *Byte* magazine.

Rising Star Industries, an independent company, was incorporated in California shortly after the Princeton meeting. Its purpose was to develop the Valdocs software. The name Rising Star was chosen to symbolically show the partnership between the QX-10 hardware from Japan—Land of the Rising Sun—and the software from the United States—The Stars and Stripes.

With Epson's blessings, Chris Rutkowski left his Epson post—he was then the National Marketing and Sales Manager—in January, 1982, to become President of Rising Star.

The first QX-10 prototype computers arrived in America during the summer of 1982. By early fall, preliminary versions of Valdocs were being demonstrated and improved.



The QX-10 design team at Epson Corporation, Japan. The project leader, Mr. M. Akahane, is fourth from the left, with glasses. Taken from a photograph provided by Epson America, Incorporated.



Epson Corporation Headquarters in Japan. Epson employs about 2,600 people in Japan alone. Courtesy Epson America Incorporated.

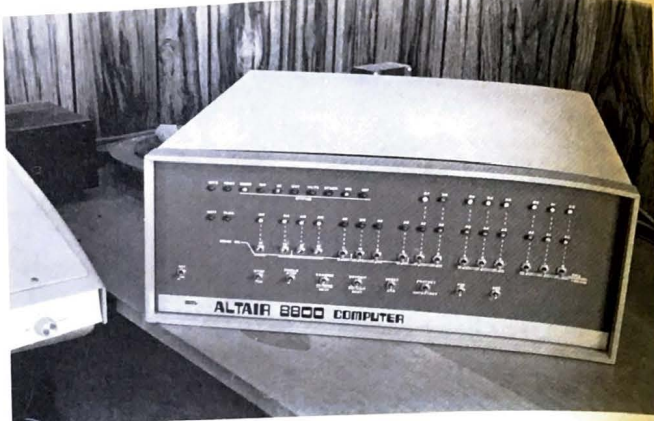
Also in 1982, Epson announced the HX-20 computer, a "briefcase" portable computer offering a built-in printer and microcassette tape system. It was developed and brought to market by a team of ten engineers over a two-year period.

With the maturing of the Epson brand name at hand (in 1982 Epson America contributed \$160 million of Shinshu Seiki's \$435 million in worldwide sales), Shinshu Seiki changed its name to Epson Corporation. This is a powerful statement of support to Epson America, especially in Japan where honor, pride, and dignity hold far different values than in the West. A close American corollary might be for Chevrolet to change its name to Holden, the name of its sister equivalent in Australia.

Epson America is now a rapidly growing company. In just a little over two years, it went from a 1980 work force of 17 people to more than 260 in 1983. (Epson Corporation, in Japan, employs more than 2600 workers.) Some product development and testing, as well as market research, is now done in the United States.

Virtually all software for the QX-10 has been developed in the United States, making this computer a truly international product combining the best of two countries—the American flair for invention and innovation and Japanese product refinement and quality mass production.

2



A Walk Around the QX-10

THE QUEST FOR THE STANDARD PERSONAL COMPUTER

The typical personal computer of 1975 was a rather cumbersome box filled with an assortment of plug-in cards and modules, lights, switches, and wires patterned after the minicomputers of the time. Everything was “unbundled.” Potential users had to choose how much memory to install, what interfaces to buy, and (when they became available) what kind of disk drives to use. Even the terminal, usually a CRT or teletype-like printing device, had to be bought separately.

Anyone who wanted to buy a computer in those days had to be a rugged individualist, preferably with a strong technical background. The buyer had to select compatible cards (or be able to make them compatible) for the system being assembled. Typically little or no help was available from the computer store or manufacturer who sold the goods.

The keyboards and screens of the computers were at first built external to the main computer. DecWriters, various Teletype machines, and other commercial grade computer terminals were common. The home television set was often pressed into service as a display device.

Altair Computer built from a kit, circa 1978. Computer courtesy Mike Ladam.

The Apple computer was one of the first computer systems with a keyboard and television interface included as a standard part of the computer. The Apple IIe still has the original 40 character display line (chosen because most television sets smudge smaller characters), although various “80-column display” adapter cards can be used with computer monitors.

Disc drives for floppy disks, until fairly recently, were considered “add-ons.” Sold separately, they were connected to the computer with one or more (usually tangled) cables.

THE STANDARD COMPUTER EMERGES

Fortunately, today’s computers are much easier to buy and use. High-performance personal computers intended for business and scientific use are usually packaged in one or two more or less standard configurations. This type of computer system has a detachable CRT screen (included with the purchase of the computer); an electronics case with two 5¼-inch double-sided, double-density disks inside; and a detachable keyboard. A variation of this configuration combines the display and electronics in a single case, locating the disk drives on the right side of the screen.

Although the internal workings of the standard computer are still being sorted out, there are usually five to seven slots inside for “options” (to be selected by the user), and built-in ports for a printer and a modem (an acronym for “modulator-demodulator,” a device used to transmit data over telephone lines).

Personal computers of the mid-1980s use either an 8- or 16-bit microprocessor; some feature various combinations of both. The difference between 8- and 16-bit computers is principally how much memory can be addressed, not necessarily speed. Eight-bit computers run most tasks about as fast as the larger 16-bit systems unless the job requires a lot of “number crunching,” a very large spreadsheet being a typical example. Sixteen-bit systems can address megabyte-sized memories, whereas 8-bit systems are normally restricted to a maximum of 64K bytes. (Specialized hardware and software can be used to provide “bank switching” for access to more memory.)

ENTER THE QX-10

The QX-10 falls into the category of the “three-box” computer. A lightweight 12-inch monitor sits atop the main electronics housing. The electronics case contains two disk drives, a Z-80A eight-bit processor, up to 256K bytes of main memory with bank switching, 128K bytes of video display memory, and an additional 2K bytes of battery-powered CMOS memory (used to save operating conditions during power dropouts or when the computer is turned off). The standard QX-10 also features built-in parallel printer and serial ports, five option slots, a battery-powered calendar-clock, and a light pen input.



The Epson QX-10 computer shown with the HASCI keyboard.

The QX-10 computer is attractively packaged in a cream-white, precision-molded plastic case. (So precise is the fit that the option card guides inside the accessory area were built into the top cover.) A finely textured finish on the plastic case dulls reflections, yet will not collect dirt. The QX-10 can easily be wiped clean with a damp cloth or cleaned with a mild detergent to remove stubborn accumulations.

THE QX-10 KEYBOARDS

Two keyboard layouts are available for the QX-10. At first glance, they look much like others on the market, except for the number of keys—there are over a hundred of them in each layout.

The overall keyboard design is attributed to Chris Rutkowski, leader of the American side of the QX-10 design team. It is modeled closely after standard office product typewriters, and touch typists will probably feel right at home on the QX-10. The keytouch is light with a reasonably tactile feel and quiet—there are no sharp clicking noises.

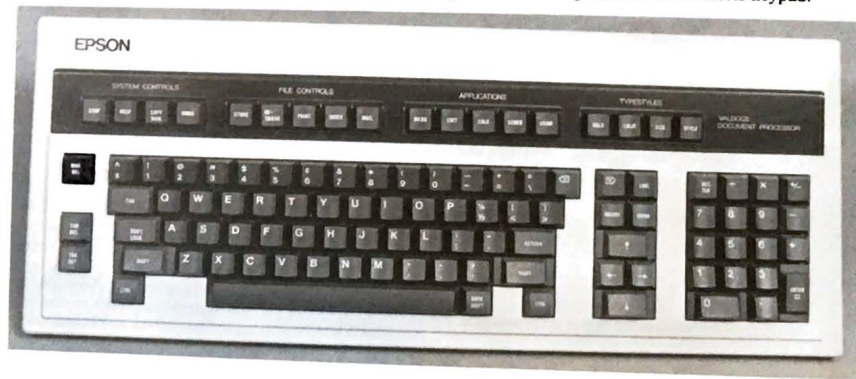
The keyboard first supplied with CP/M-based QX-10 computers used a standard ASCII configuration in the main typing area. (QX-10 computers for use only with CP/M are no longer sold in the United States. This is because CP/M can be used on any QX-10 computer configured for Valdocs.) The keyboard now supplied on all QX-10 computers sold in the United States is called the HASCI (Human Applications Standard Computer Interface) keyboard. It is specifically tailored for use with Valdocs. The main typing section of this keyboard is patterned after the standard office Selectric typewriter. A utility program called "SETUP," which is included with the QX-10 software, allows either keyboard to be selected and used.

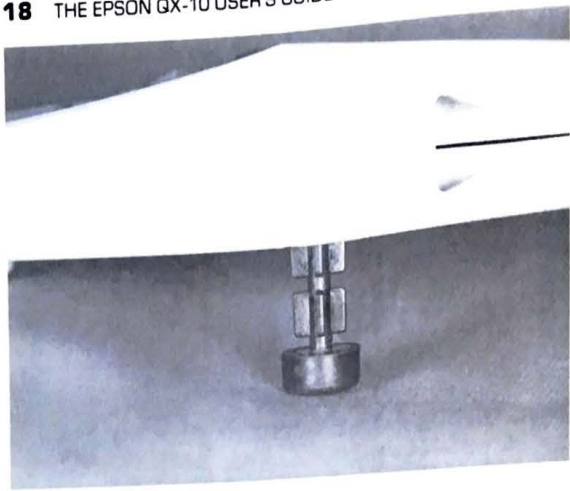


The original HASCI keyboard.

Two versions of the HASCI keyboard are manufactured. The first version is mechanically similar to the ASCII keyboard. The second, a more recent HASCI design, features simpler mechanical assembly, improved key sensitivity, and two additional keys. The layouts of all keyboards are grouped into four functional clusters: a main typing area, an editing cluster, a numeric keypad, and the function keys.

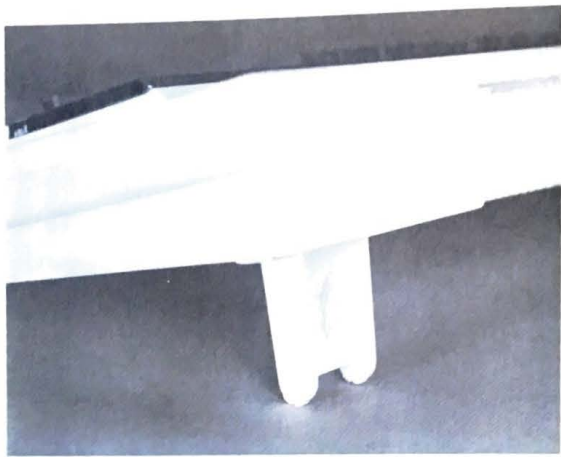
The "Brother" HASCI keyboard. This keyboard features sculptured keys and weighs less than the original. Notice the extra "EDIT" key in the applications area, and slightly different arrangement in the numeric keypad.





The extension feet on the original HASCI and ASCII keyboards. The back of the keyboard can be set to three elevations by the legs, which can be locked into place with a quarter-twist.

The keyboards are supplied internationally in Danish, English, French, German, Italian, Spanish, and Swedish. A multilingual keyboard layout had not been announced at the time of this writing.



The extension legs on Brother keyboards can be set to give three angles of elevation by positioning them at an angle to the base of the keyboard.



The ASCII keyboard features removable caps on the function keys.

The Main Typing Area

The main typing areas of both keyboard designs provide double-sized (wider) keytops for functions normally struck by the little fingers—the tab, an illuminated shift lock, shift, and return keys on the Valdoks layout, and the tab, control, shift, backspace, and return keys on the ASCII layout. The ASCII keyboard also provides an illuminated CAPS LOCK key rather than the HASCI shift lock.



The main typing area of the original HASCI keyboard.



The main typing area of the Brother HASCI keyboard.

The HASCI keyboard includes all the standard Selectric keys in the usual typewriter locations. The two control keys (one on each side of the space bar) and the GRPH SHIFT key (described below), are the only "computer" keys added to the Selectric layout.

The differences in the HASCI and Selectric layouts are concentrated in the two upper corners. On the right side, the Selectric backspace key, an enlarged key on most typewriters, has been replaced with a backslash key and a normal-sized backspace key. The typewriter EXP (a manually controlled carriage return) and the oversized carriage return key are replaced with a double-sized carriage return key and two normal-sized keys for left and right brackets.

EPSON



The main typing area of the ASCII keyboard.

The top left corner replaces the enlarged TAB key on the typewriter keyboard with a key for the mathematical symbol "+/-".

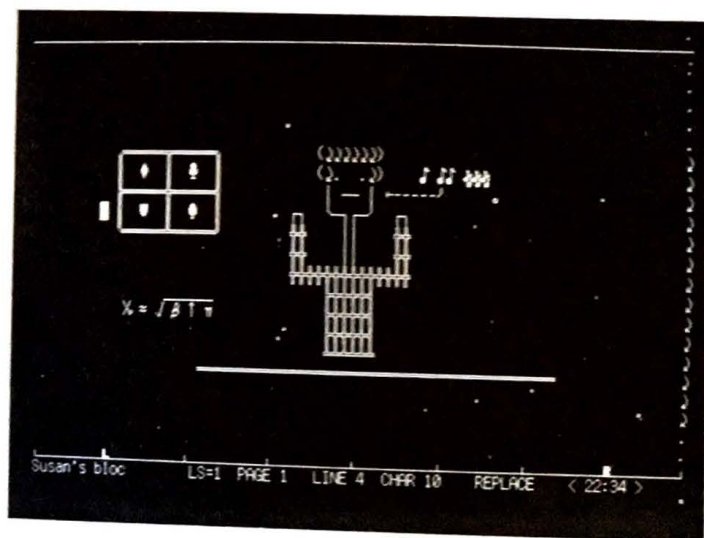
Both keyboards have a "GRPH SHIFT" key located near the right end of the space bar. The QX-10 uses this key as a "super-shift" to select two additional character sets for display on the screen. The GRPH SHIFT can be used alone to select Greek and mathematical symbols, or in combination with the SHIFT key to select block graphic symbols.

Both sets of graphics symbols are printable on Epson dot matrix printers, but may cause other brand printers to misbehave. These symbols are shown mapped against the standard keyboard characters in Appendix B.

Neither keyboard layout has a "repeat" key. Most keys automatically repeat when held down for about a second. Those that do not, such as the up and down arrow keys in the edit cluster, can be forced to repeat by holding down the control key with the desired key. The CP/M utility program "SETUP" can enable or disable the automatic key repeat feature.

The keyboard buffer (about 64 characters) will save repeated characters if screen operations, such as scrolling, cannot keep up. When this happens, whatever function was being performed will continue until the buffer is emptied. (Valdocs allows this sort of thing to be halted by use of the STOP key.)

The Valdocs man. The figures shown were drawn on the QX-10 screen using block graphic characters. Such characters can be mixed with actual text and printed using Valdocs versions 2 and above.



The Editing Clusters

The editing cluster in the ASCII layout features large up and down arrow keys with standard-sized left-right controls sandwiched between them in a logical and natural pattern. The INS (insert) and DEL (delete) keys, just above these, are the next most commonly used edit keys.

The exact function of the HOME and CLS keys, just above the INS and DEL keys, is dependent on the text processing software used in the QX-10, and may vary from one editor to the next. The HOME key is normally used to move the cursor to the top left of the screen, and the CLS key to clear the screen.

The HASCI editing cluster provides the same up-down, left and right arrow, and insert keys as the ASCII layout, but the WORD and LINE keys replace the ASCII DELETE and CLS keys. The erase key, marked with an X inside a large arrow pointing to the left, operates just like the backspace-erase key on self-correcting electric typewriters, backspacing and erasing the character under the cursor.

The last edit cluster key, marked with an X inside an arrow to the right, is located in the top left position in the editing cluster and replaces the ASCII HOME key. It erases characters to the right of the cursor.

Chapter 6, "Valdocs," explains the operation of the HASCI edit keys in more detail. Simply stated, the arrow and erase keys are used to move the cursor or erase characters in the indicated direction. If either the LINE or WORD key is pressed after these keys, the cursor moves or deletes characters to the end of the line or word in the direction of the arrow pressed before.



The original HASCI keyboard editing cluster and numeric keypad.



The Brother HASCI keyboard editing cluster and numeric keypad.



The ASCII keyboard editing cluster and numeric keypad.



The removable keycaps on the ASCII keyboard function keys. Custom legends can be inserted under the keycaps.

The Numeric Keypads

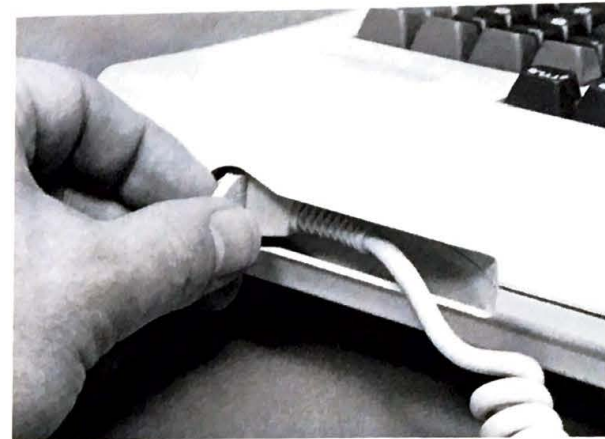
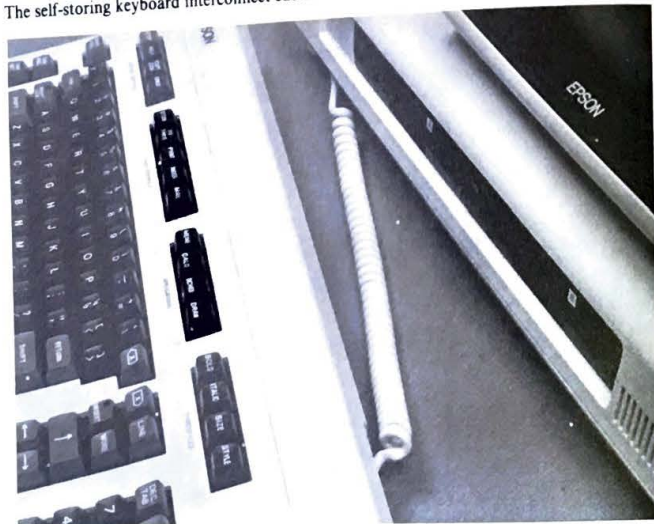
The numeric keypads are complete with all the keys necessary for calculator usage. Included in this cluster are multiply, divide, subtract, add, equal, comma, and decimal point keys; a large, strategically located return key, which is used for data entry, completes the pad.

The ASCII numeric keypad, with 19 keys, includes a single and triple 0 key, similar to some calculators. The multiply and divide symbols, * and /, commonly used by programmers, are provided because the screen used with CP/M does not have standard multiply and divide symbols.

The original HASCI keyboard features an 18-key HASCI numeric keypad with standard symbols for the multiply and divide keys. (The Valdocs screen can display these symbols directly.) A decimal tab key allows numbers to be entered with the decimal point automatically lined up on a predefined tab position. A standard single 0 completes this keypad.

The second (and newer) HASCI numeric keypad combines the equal and enter keys, and adds one for sign inversion, similar to that on many calculators. (This key changes the sign of the accumulated number.) The remainder of the numeric keypad is unchanged from the original HASCI version.

The self-storing keyboard interconnect cable.



Installing the keyboard interconnect cable. Notice the convenient tab on the cable connector. The connector, when installed, is recessed from the back of the keyboard, protecting it from accidental disconnect.

Function Groups

The ASCII keyboard provides 14 function keys with removable plastic caps. Customized legends can be placed under them for special applications. Four additional illuminated function keys labeled SF1 through SF4 (without the removable caps) are located over the numeric cluster.

The HASCI keyboard preassigns and labels all 17 function keys for operations performed by Valdocs. The keys are separated into four functional areas called System Controls, File Controls, Applications, and Typestyles. The functions of these keys are described in Chapter 6, "Valdocs."

Keyboard Interconnect

The keyboard is attached to the main electronics case with a coiled cable that will stretch to about four feet. The cable uses an eight-pin DIN connector with a folding tab for easy connection to the keyboard and electronics. The keyboard and electronics connectors are on opposite sides, so the cable is stretched diagonally between the two units. This results in a "self-storing" cable, since it will lay flat when the keyboard is pushed back against the electronics case.

THE MAIN ELECTRONICS CASE

This unit houses all the computing electronics, the two floppy-disk drives, and power supplies for both the computer and CRT screen. The case itself is made of three molded plastic parts: a bottom shell, the top, and the accessory bay cover, which forms a part of the top cover. The top shell also provides molded card guides for each of the five slots in the accessory bay.

The front of the unit has the keyboard jack in a cutout recess near the lower left corner, the two disk drives in the center, and the system reset switch just below the right disk drive.

The Disk Drives

The disk drives use a push button to load and unload floppy diskettes, rather than the usual door or "tongue." Floppy diskettes are loaded by inserting them label up until they click into place. The extended push button is then depressed, which loads the head and places the drive on line.

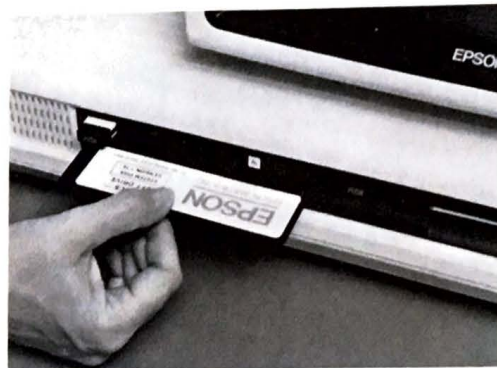
The drives are unloaded by pressing the push button once again, which pops the diskette out of the drive about $\frac{1}{2}$ inch, where it can be easily removed by hand. The disk drives use double-sided, double-density, $5\frac{1}{4}$ -inch diskettes and provide a formatted storage capacity of about 380K each.

System Reset Switch

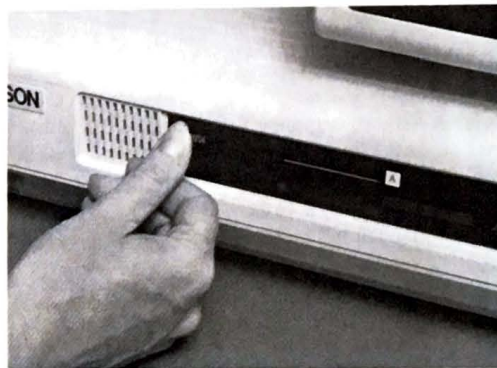
The system reset switch is not frequently used. Located below the right disk drive in a notch cut into the electronics case, it is easily reached, yet out of the way and unlikely to be accidentally activated.



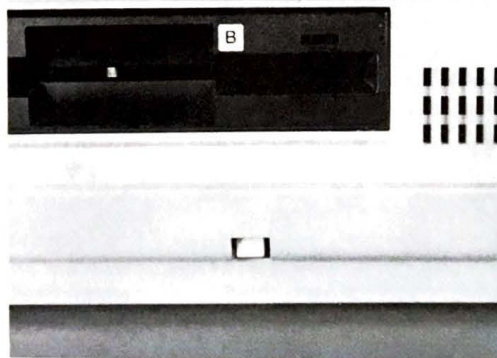
The QX-10 main electronics case.



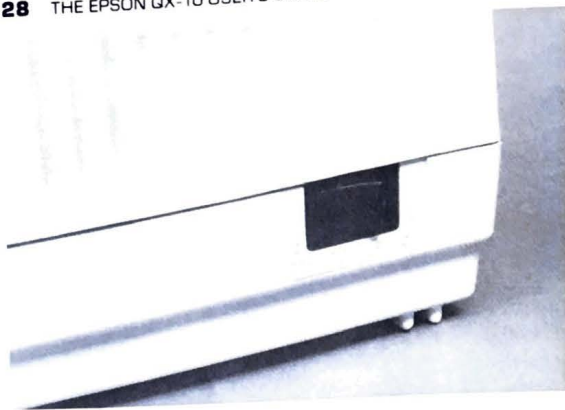
Diskettes are inserted into the drives, label up.



Once inserted all the way, diskettes automatically lock in place. The drive is enabled when the "PUSH" button is depressed. Diskettes are removed by pressing the "PUSH" button once again.



The reset switch, located in the front of the main electronics case.



The main power switch.

Probably the most common use for the manual reset is to quickly change operating systems without turning off the computer. The operating-system disks, for example TPM II and CP/M, can be swapped in the disk drive and the reset button pushed, avoiding the electrical trauma of shutting off the computer then immediately turning it back on.

The manual reset should be used with caution, since the effect is nearly the same as turning the computer off; anything in memory and not saved on disk will probably be lost when it is pushed.

Main Power Switch

The power switch is located near the right back corner of the QX-10, near the back of the electronics case. It is a standard rocker switch and, unless a switched power outlet is available, must be accessible during normal operation. Care should be used to ensure that books or other material likely to be stacked on the right side of the computer cannot fall against the switch and cause an accidental disruption of power.

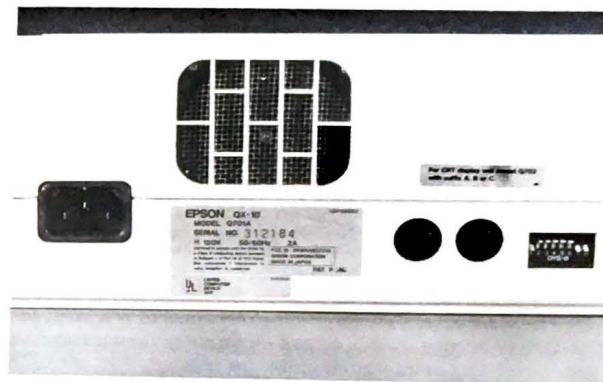
The Back Apron

The back apron of the QX-10 has a standard three-conductor power connector for the AC line cord, the exhaust from the internal cooling fan, a five-pin DIN jack for a light pen, an eight-pin DIN connector for power and video for the CRT screen, and a bank of eight DIP (Dual Inline Package) switches. (These switches are read when the QX-10 is first turned on for memory configuration information. Make sure that switch 1 is in the down position before turning on your computer.)

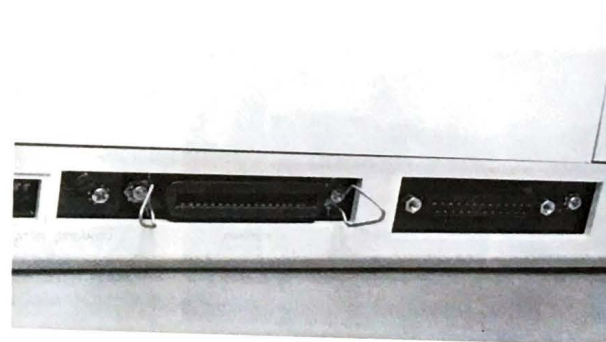
The volume control for the "bell," which is actually a small loudspeaker inside the case, is located next to the parallel printer interface connector. The RS-232C interface connector

(a standard DB-25S) is located next to the printer port. (Connection pinouts for these connectors are listed in Chapter 3, "Inside the QX-10.")

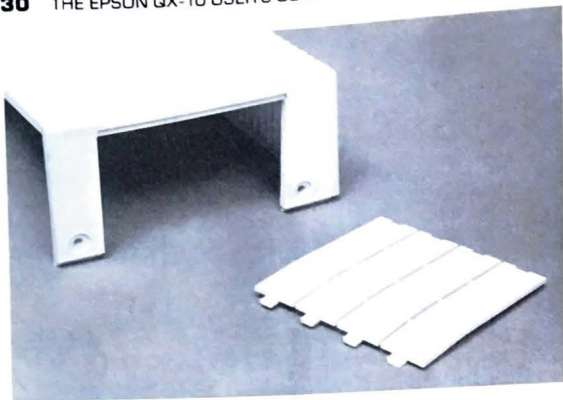
The accessory bay cover, held in place by two screws on the back apron, has removable panels for those accessories that require external connectors. Inside, the accessory bay has sockets for 192K bytes of dynamic random access memory (RAM—already installed in the Valdocs versions of the QX-10) and five sockets for option cards. Chapter 9, "Expansion Cards and Accessories for the QX-10," describes some of the accessory products available for the QX-10.



The back apron of the main electronics case showing the power cord connector, the fan, light pen and monitor connectors and the memory configuration DIP switches.



The back apron of the main electronics case showing the loudspeaker volume control, parallel printer, and serial interface connectors.



The expansion area cover. The cover is removed with two screws. The back dress plate is snapped apart to provide access only to those expansion cards requiring external connections.

THE QX-10 MONITOR SCREEN

The QX-10 monitor is housed in a lightweight plastic case. The majority of the weight of the monitor is in the faceplate of the 12-inch (diagonal measure) CRT, making the monitor case front-heavy.

The CRT screen uses a green phosphor and is matted to reduce glare and reflections. The QX-10 display itself is of particular note. The characters are well designed, sharp, and legible, resulting in less eye strain for the user.

The video signals and sweep speeds used in the QX-10 are not compatible with standard television monitors, and so it is not possible to use other monitors with the QX-10 computer.

Left: The QX-10 CRT monitor. *Right:* Rear view of the QX-10 CRT monitor.



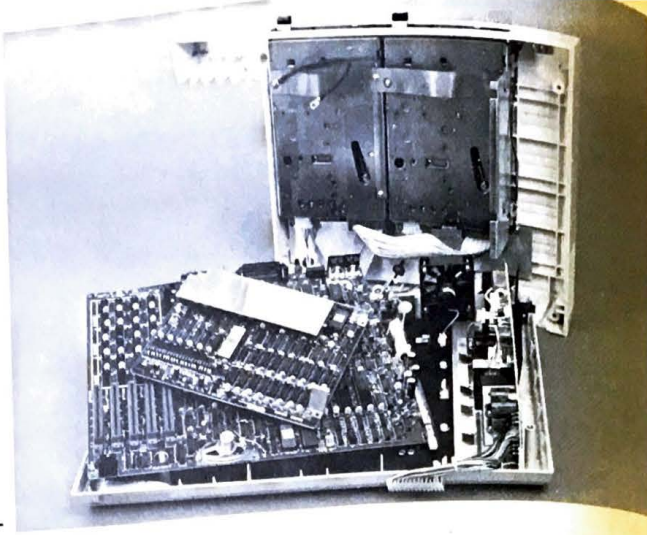
The back of the monitor case has an eight-pin DIN connector for the 27-inch power and signal cable that connects to the electronics case. A brightness control is "finger adjustable"; the focus control (which does not normally need adjustment) is recessed and requires a screwdriver for adjustment.

This completes our tour around the outside of the QX-10 computer. Chapter 3 will provide technical details of the computer hardware, system architecture, and interconnections for the serial, parallel, and light pen interfaces. If you are not interested in the electronics of the QX-10, Chapter 3 can safely be avoided without fear, but those with technical interests will find the sophistication of the QX-10 design interesting.

A SUMMARY OF TECHNICAL SPECIFICATIONS FOR THE QX-10

Microprocessor	<ul style="list-style-type: none"> • Z-80A (NEC μPD780AC-1) running at 4 MHz.
Memory	<ul style="list-style-type: none"> • 256K bytes maximum on main board, installable in 64K byte increments, total dependent on option selected. (256K bytes are supplied in systems sold in the United States.) • 2K bytes CMOS RAM with battery backup. • 32 to 128K bytes display memory depending on options selected. (128K bytes are supplied in systems sold in the United States.) • 4 to 32K bytes EPROM, depending on configuration.
Calendar/Clock	<ul style="list-style-type: none"> • CMOS time-calendar clock, battery backup.
Speaker	<ul style="list-style-type: none"> • Permanent magnet speaker usable as bell or for music.
Interfaces	<ul style="list-style-type: none"> • Parallel Centronics-compatible printer port. • Serial, RS-232C communications interface, fully programmable for baud rate, word size, and stop bits.
DMA Channels	<ul style="list-style-type: none"> • Light pen input for use with optional light pen.
Interrupts	<ul style="list-style-type: none"> • Seven available: CRT, floppy disk, and five option slots.
Counter/timer	<ul style="list-style-type: none"> • 15 levels of vectored interrupts.
Floppy disks	<ul style="list-style-type: none"> • Six channels • Two $5\frac{1}{4}$ inch drives, 380K bytes formatted each, 48 tracks per inch, double-sided, double-density.
Expansion	<ul style="list-style-type: none"> • Five expansion-card slots.
Monitor	<ul style="list-style-type: none"> • Green, long persistence 20 MHz high-resolution monitor.
Display	<ul style="list-style-type: none"> • Alphanumeric display: 25 lines, 80 column lines. Graphics: 640 dots horizontal by 400 lines.
Keyboards	<ul style="list-style-type: none"> • ASCII, 103 keys including 18 special function keys, 8 edit keys, and a 19-key calculator pad. • HASCI, 104 keys, including 17 special function keys, 8 edit keys and a 19-key calculator pad.
Cooling	<ul style="list-style-type: none"> • Forced air (internal DC-drive quiet fan).
Dimensions (inches)	<ul style="list-style-type: none"> • Electronics case 20 \times 13.4 \times 4 • Monitor 12.25 \times 13.4 \times 10.6 • Keyboard 20 \times 8.8 \times 1.9

3



Inside the QX-10

This chapter is dedicated to those who want to know "what is under the hood" of the QX-10. Once again the caution: mucking around the insides of this computer can be nothing but bad news. With the exception of NiCad battery replacement, which will probably be needed every three years or so (industry average), no internal service is normally required.

The QX-10 keyboard, monitor, and electronics cases should never be opened except for repairs. If you are unusually well qualified, have a good deal of digital test equipment and a copy of the QX-10 service manual, there is a chance that home repair is possible. Otherwise you should not attempt any kind of maintenance effort yourself, painful as that may be. Epson maintains an expanding network of service centers qualified to repair your QX-10. Those that become necessary. Those active as of midsummer 1983 are listed in Appendix C.

The first part of this chapter is devoted to the internal mechanical aspects of the QX-10. Photographs show virtually everything there is to see. You are encouraged not to attempt to take your QX-10 apart for comparison. Although it may have minor mechanical differences, they will not be significant and only reflect a different production run.

The second part describes in limited detail the electronic design of the computer. It is not meant to serve as a technical manual for either field-repair personnel or programmers, but will give a good idea of how the QX-10 works.

Exploded view of the QX-10.

32

This chapter concludes with a description of the serial, parallel, and light pen ports. It is intended to give a general understanding of the RS-232C serial interface standard, how the parallel printer interface operates, and basic information about the light pen interface. Detailed schematic diagrams are provided to help those with special interface requirements.

PART ONE: AN OVERALL SURVEY OF THE QX-10

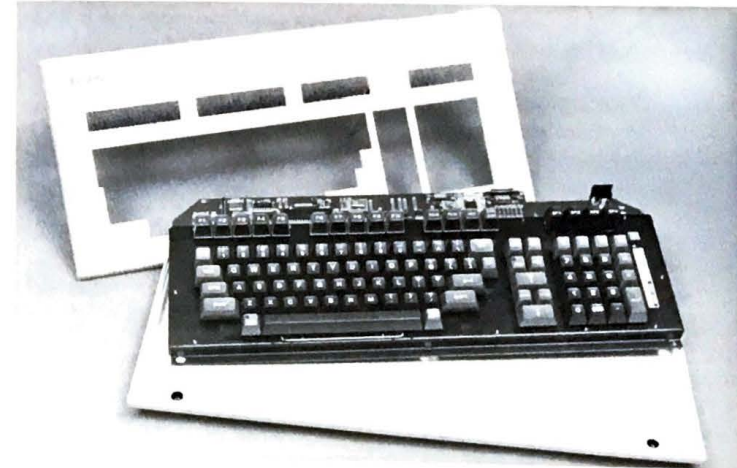
As mentioned in Chapter 2, the QX-10 is split into three main components consisting of the keyboard, the main electronics case, and the CRT monitor. The keyboard electronics contains an independent microprocessor for scanning the keys and controlling the light-emitting diodes (LEDs) used on illuminated keys. Keyboard data is sent serially to the main electronics case.

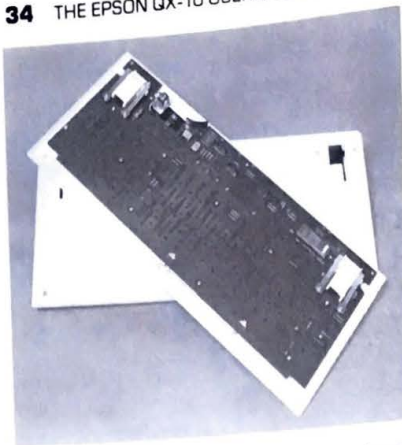
The main electronics case contains the disk drives, computing electronics, power supply, and a small fan that provides forced-air ventilation. All of the computing electronics is built on a single large printed circuit board, or motherboard. The display controller and screen memory are built on a small double-sided printed circuit board that is mounted "piggy-back" on top of the main board; the power supply, also a separate module, is mounted on edge, just inside the case next to the main power switch.

The two disk drives, mounted in the top of the electronics case, are 40-track, double-sided, double-density drives made by Epson.

The CRT monitor contains a single printed circuit board that receives the video and deflection drive signals from the QX-10. A 12-inch CRT with long-persistence green

Inside the ASCII keyboard.





Inside the Brother HASCI keyboard. Notice the legs are attached to the top cover through the printed circuit board, rather than the bottom case.

phosphor is used for the display screen. Anode high voltage for the CRT is derived from the horizontal drive, in the same manner as most television receivers. Power for the monitor is provided by the QX-10 main power supply. The CRT monitor is built for Epson by Hitachi.

Inside the Keyboard

Both the original HASCI and ASCII keyboards (made by Fujitsu) are mechanically and electrically similar, the only difference being the physical layout of the keys. They consist of top and bottom covers and an electronics module, built on a single printed circuit board. The two covers and electronics module are held together in a sandwich-like fashion with six screws through the bottom cover.

The newer HASCI keyboard, built in cooperation with Brother, features a case integrated with the key caps, providing a much lower profile and lighter weight.

All key contact switches are mounted directly on the top of the circuit board. The ASCII keyboard has a metal stiffening and support plate secured to the top of the printed circuit board to carry the individual key guides and bearings. (These guide the plungers from the key caps to the actual key switches.) Rubber bumper pads, mounted on the stiffener plate provide a "soft" key-travel stop for keys, such as the carriage return, that are likely to be "pounded."

The original HASCI keyboard used a plastic, rather than metal, stiffening plate. The plastic version both stiffened the printed circuit card and protected the switches and keyboard electronics from dirt or liquids spilled into the keyboard. The newer Brother keyboard eliminates the plastic stiffener, integrating the stiffening function into the keyboard top cover.

Key switches, accessible by removing the stiffener plate (or top cover on the Brother keyboard), are soldered directly onto the printed circuit board. Each is independent and can be individually replaced if necessary.

The keyboard electronics consists of an 8049 microprocessor, six small-scale integrated circuits, and a monolithic voltage regulator for on-board regulation of the logic power supply. All of the keyboard electronics are located along the top edge of the printed circuit board, above the stiffening plate (if applicable).

The microprocessor provides keyboard scan, control of the lights on illuminated keys, and a 64-character buffer to allow "type-ahead." (Some CP/M keyboard input routines and other software packages deliberately clear all characters from the keyboard before reading an input, effectively defeating the type-ahead feature.)

The Main Electronics

Access to the QX-10 electronics is obtained by first removing the accessory bay cover held in place by two screws on the back apron. The top cover is secured by two screws inside the accessory area and the two screws near the right corners, which are under cosmetic covers.

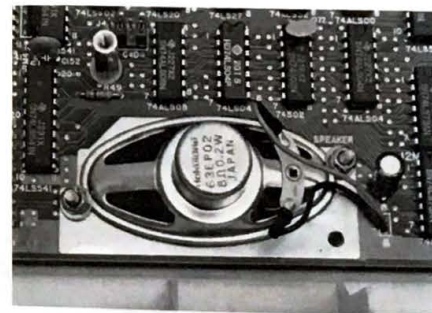
After all four cover screws have been removed, the top cover with the two disk drives attached can be lifted up, exposing a ground wire attached to the main printed circuit board next to the internal loudspeaker. This strap must be disconnected before the cover can be moved more than about six inches.

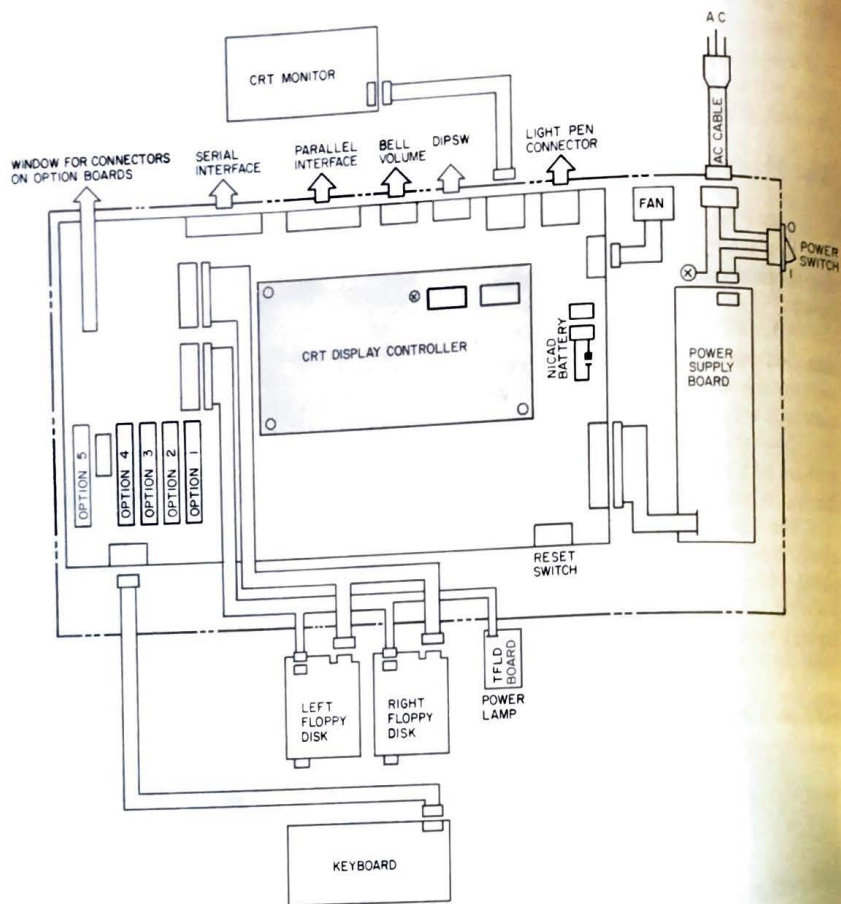
Once the ground strap is freed, the top cover can be lifted to stand vertically behind the bottom half of the case. When disk-drive power and data-cable connectors are disconnected, the top is completely freed.

The video controller circuit board, measuring $7\frac{1}{4} \times 9$ inches, is held to the main printed circuit board by four screws. Care should be exercised not to damage the connector underneath when it is separated from the main board.

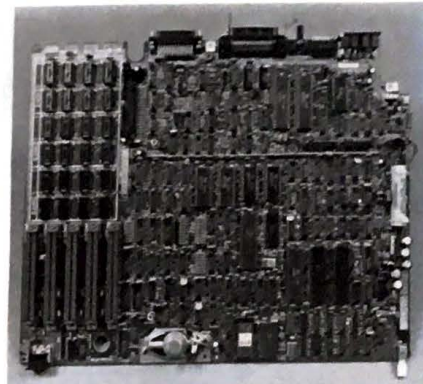
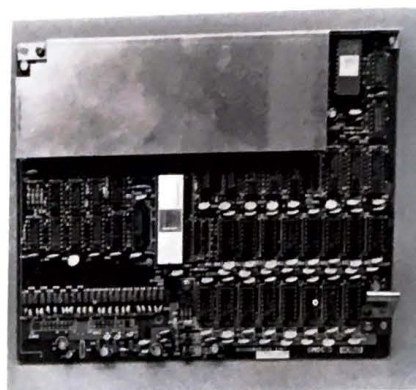
The main printed circuit board is secured to the bottom case by a combination of locking tabs molded into the case and five screws in the case bottom. A metal plate separates

Left: The QX-10 main electronics case. Right: The internal loudspeaker usable for both bell and under software control music and sound effects.





Internal cabling diagram of the QX-10.

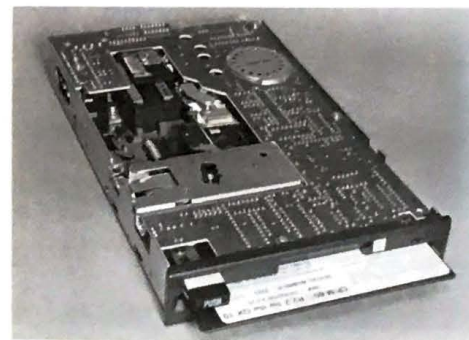


Left: The QX-10 video controller has more memory (128K bytes) than most personal computers have for the entire system. *Right:* The QX-10 main electronics board with video controller removed.

the main circuit board and power-supply card from the case bottom, providing both the UL-required fire barrier and shielding to minimize radio and television interference.

All connectors for the keyboard, power supply, and expansion slots, as well as for the light pen, serial, and parallel ports, are mounted directly on the main printed circuit board. Five-volt power is distributed across the central part of the board using a "bus rail" (visible in the photographs), which can be damaged or broken if the board is flexed. (Do not bend the main printed circuit board!)

The 3.6-volt NiCad battery, used to maintain the calendar-clock and 2K byte CMOS memory chip, can be replaced without removing the nylon restraints passing through the main printed circuit board if care is used when it is taken out. If the straps are cut, it may be necessary to remove the printed circuit card from the bottom case to replace them.



The QX-10 floppy disk drives. These drives are made by Epson.



The battery used for backup on the calendar-clock and 2K byte CMOS memory. Shown just below the battery is one of the printed circuit card retainers molded into the bottom section of the case.

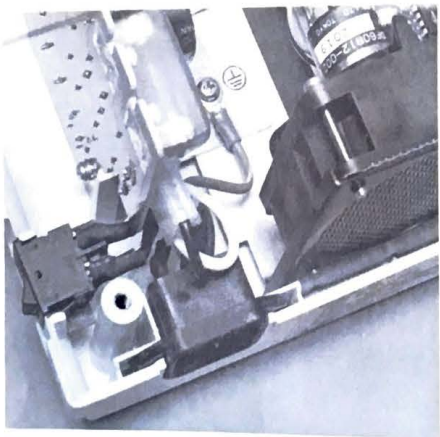
The Power Supply and Fan

The power supply is mounted on edge along the right side of the case next to the power switch and AC power cord. It is of a "switching mode" design, and so avoids the weight of a standard iron core transformer.

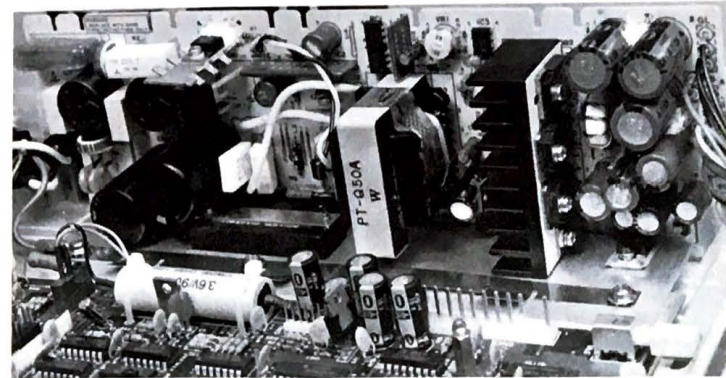
The standard industrial three-conductor AC line cord connector and main power switch are held in place by retainer mounting brackets molded directly into the case.

The metal fire barrier under the power supply is electrically bonded to the one under the main printed circuit board and the third (green) wire from the AC line cord.

The ventilation fan is powered from the 12-volt DC power supply and measures 2½ inches square. It is a ducted design with ten blades for quiet operation. A wire-mesh screen is installed between the fan and its mounting bracket to prevent damage to both the fan and wayward fingers from the outside.



The AC connector is mounted in a bracket molded into the QX-10 case.

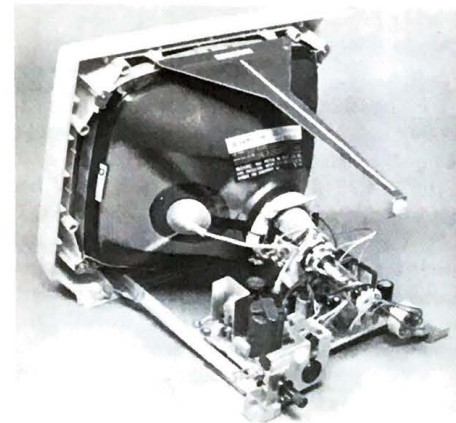


The switching mode power supply used in the QX-10 is mounted vertically.

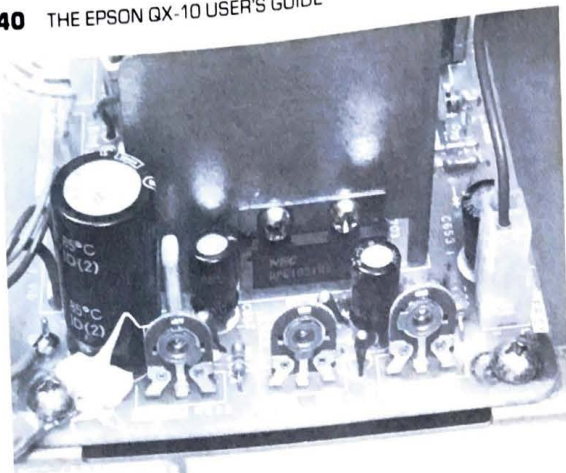
The CRT Monitor

The monitor case is held in place by four screws. Once these are removed, the entire cover slides off. The internal mechanical structure depends entirely on the case for rigidity and can easily be damaged by normal handling when the case is removed, or if any of the four case screws are left out after reassembly.

The monitor electronics is entirely contained on a single printed circuit card measuring 3½ × 7½ inches. Internally, there are controls for height, vertical linearity, vertical hold,



Internal view of the QX-10 CRT monitor.



Internal controls in the QX-10 monitor. In practice, these controls will not need any adjustment.

and an auxiliary brightness adjustment. In practice, none of the internal controls need adjustment.

The CRT monitor case should not be opened except by trained service personnel. The CRT's high anode voltage, mechanical delicacy when disassembled, and the ever present danger of CRT implosion due to accident or mishandling are dangers that should be avoided whenever possible.

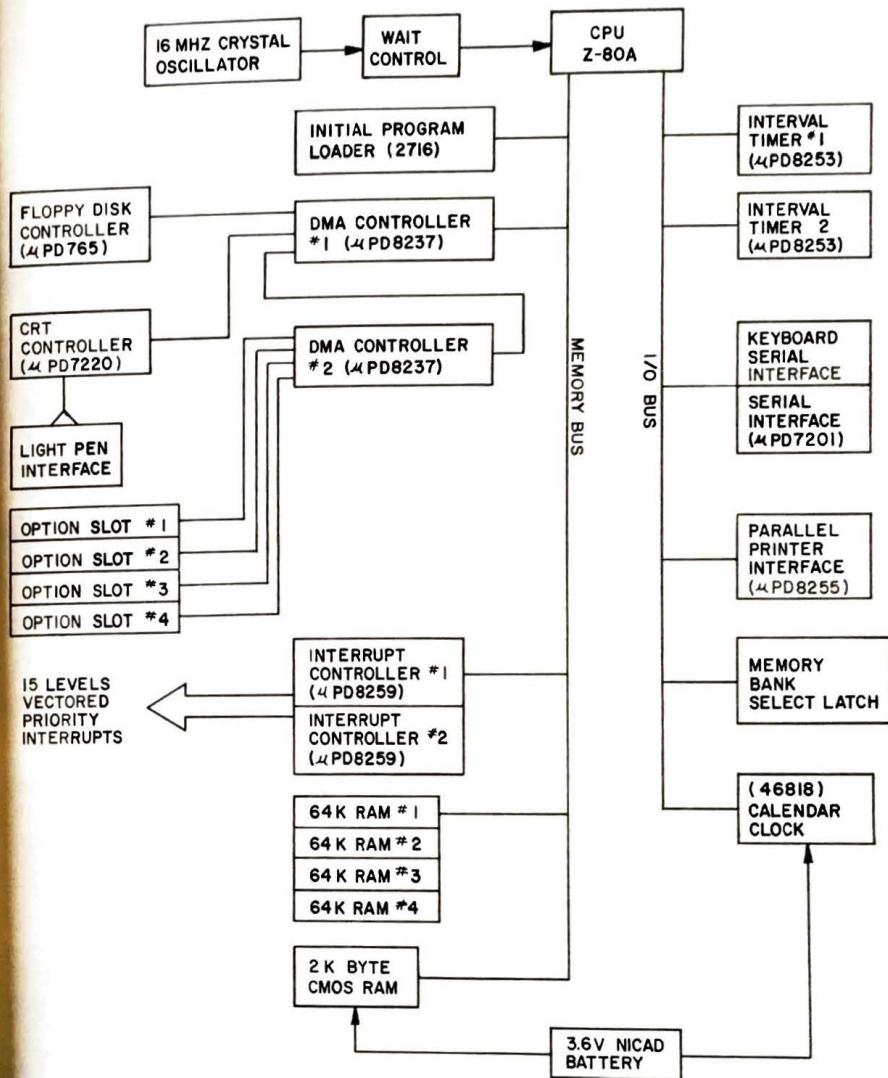
PART TWO: THE QX-10 ELECTRONIC DESIGN

Main Electronics Board

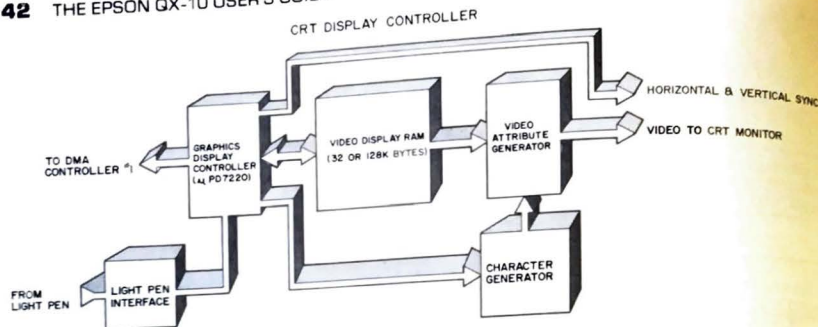
The QX-10 is built around a Z-80A microprocessor running at 4 mHz. Depending on configuration, up to 256K bytes of software-controlled bank selected memory (using 64K byte dynamic RAM) is available as well as 2K bytes of battery supported CMOS RAM. (The CMOS RAM is selected as a "fifth" bank and is used to store system configuration information and other status data when the QX-10 is turned off or unexpectedly loses power.)

Two NEC uPD8253-5 counter-timer integrated circuits provide timing for the serial interface baud rate clock, the keyboard serial interface clock, the bell duration and frequency, and three unassigned timers for program use. All are accessible for use if you are an expert programmer. Chapter 7, BASIC on the QX-10, demonstrates how some can be modified to suit personal needs.

Two uPD8259 interrupt controllers provide 15 levels of priority interrupt. All 15 are assigned to functions in the priorities shown in Appendix D.



A block diagram of the QX-10 electronics.

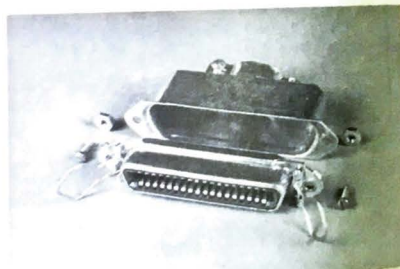
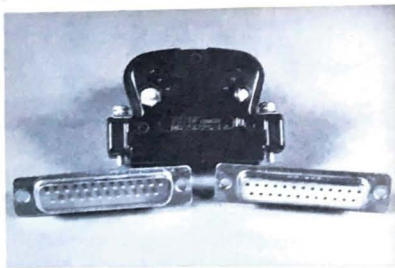


A block diagram of the QX-10 video display controller.

A NEC uPD765 is used for floppy disk control. Data for the floppy disks, the CRT display, and four of the option slots are controlled through two uPD8237 DMA (Direct Memory Access) controllers. These provide automatic high-speed data transfer without loading the Z-80A CPU.

The CRT display controller uses a uPD7220 graphics display controller integrated circuit and a separate display memory. (The display memory is in addition to the Z-80A main memory. Computers configured for CP/M have 32K bytes of display memory, whereas Valdocs versions have 128K bytes.) Since the display controller has its own memory and is fed data via DMA (Direct Memory Access), very little CPU overhead is required for and screen operations. Special hardware in the display controller provides display zoom and pan operations.

Left: Cinch DB-25 series connectors used for RS-232C connectors. The hood (connector cover) is shown in the background. *Right:* Typical connector used for Centronics parallel printer interfaces.



All other I/O operations use the normal Z-80A I/O bus. This includes control and access to the keyboard, serial RS-232C and parallel printer interfaces, the calendar-clock and light pen, the bank of eight DIP switches (on the back apron), the counter-timers, the memory bank switching latch, and various other status and control registers for the CRT and floppy disk controller chips.

A listing of some (but not all) of the I/O addresses most likely to be of interest is included in Appendix E; additionally, Chapter 7, BASIC on the QX-10, shows how the counter-timers can be programmed to control the serial interface baud rate clock and change the bell frequency and duration. A word of caution is necessary here: it is possible to disable the keyboard or cause data loss on floppy disks by randomly (or accidentally) writing to the I/O ports. Furthermore, it is possible to program the uPD7220 video controller in such a manner as to cause physical damage to the CRT display monitor. Do not tamper with unknown I/O addresses!

Interfaces

The QX-10 provides a standard RS-232C and Centronics-compatible parallel interface for use with printers, modems, terminals, and other data devices that use these interface standards. A light pen input is also available for use with a light pen that will be available in the future.

The RS-232C interface is used whenever serial data transmission is required. Without going into detail, this means that a single signal wire carries all the individual data bits that make up and define a character. Serial data transmission is typically used whenever information is sent over telephone lines or outside the immediate area near the computer. The advantage of serial data transmission (over parallel) is that the electrical interface and interconnecting cables are usually simpler. The major disadvantage is that transmission speed is often slower.

The Centronics-compatible parallel interface is used to send data to printers. This type of interface is normally used for distances of 10 or 15 feet, although it can sometimes go farther. It is called a parallel interface because a separate wire is used for each of the eight bits that define a character.

The Centronics interface is named after the printer manufacturer in Hudson, New Hampshire, that developed the interface in the early 1960s for its products. It wasn't then (and isn't now) based on an official ANSI electrical interface standard. As a result, it is sometimes possible to find computer-printer combinations that don't always behave properly.

Serial Communications and the RS-232C Interface

The RS-232C standard specifies a 25-pin connector (Cinch 25P and 25S or equivalent mating connectors are typically used), signal logic (the exact meaning of signals used on the interface), the pin assignment for each signal, and the voltage levels for each signal on the connector.

The standard calls for voltages to be bipolar. A signal is usually positive when off and negative when on. The voltages may be somewhere between 3 and 25 volts when positive, and -3 to -25 volts when negative. (Any voltage within these two ranges is considered acceptable.)

Another RS-232C rule is that any or all of the signal leads may be shorted to each other for an indefinite period of time with no damage to the devices connected to the interface. This means that poorly prepared or worn cables are not allowed to cause equipment damage. The QX-10 meets this requirement by using special RS-232 interface integrated circuit drivers designed for this purpose.

The RS-232C standard can be met with as few as three signal lines: ground, send data, and receive data. Additional control lines are used in conjunction with various types of modems. In most cases, only a few of these control signals are used for a given type of modem.

The QX-10 supports 12 RS-232C signal and control lines for use with various modems. These are described in Table 3-1. (The extra signal lines are used to control the data being sent to and from the QX-10, as well as to let it know something about the condition of the devices connected.)

Serial Data Transmission

Serial character transmission is almost entirely controlled by hardware. The CPU inside the QX-10 simply checks several status bits derived by the serial interface controller chip (an NEC μ PD7201) from the various RS-232C control lines, and stores a character in the controller output register if all is well. If the peripheral device is not ready, the QX-10 waits until it is before sending the character.

The controller chip takes care of the electrical hassle of adding the start, stop, and parity bits to the eight-bit characters provided by the QX-10, and sends them out sequentially over the transmit data lines. On the receive side, the interface chip reads a serial incoming character, checks it for proper parity, and sets appropriate status flags to inform the QX-10 that a character has been received and is available to be read.

The particular controller chip used in the QX-10 (the μ PD7201) is very flexible and, with proper QX-10 software, can service a number of standard telecommunication protocols. Those with expert programming skills and high-performance telecommunication requirements will be able to effectively apply the QX-10 to these applications. The QX-10 also provides other signals for use with complex high-speed modems on pins 11, 15, 17, and 24. These are not supported in existing QX-10 software, and are mentioned for reference only. Those with interests along these lines should consult the Epson QX-10 programmer's and technical service manuals.

The QX-10 keyboard also uses an RS-232C interface to communicate with the CPU at 1200 baud (120 characters per second). The keyboard speed is set internally by keyboard ROM, but the QX-10 serial keyboard interface, like the serial port, is programmable.

Table 3-1. THE QX-10 RS-232C SERIAL INTERFACE

RS-232C pin	Signal Name	Description	Signal Direction
1	Frame Ground	Protective ground for AC power faults	Both
2	Tx Data	Data Transmitted from QX-10	From QX-10
3	Rx Data	Data Received by QX-10	To QX-10
4	RTS	Request To Send; when active, the QX-10 wants to send data	From QX-10
5	CTS	Clear to send; when active, the device connected to QX-10 is ready to receive data from the QX-10	To QX-10
6	DSR	Data Set Ready; when active, tells the QX-10 that the modem is ready for operation (exact meaning varies with modem)	To QX-10
7	GND	Signal Ground; separate from frame ground (this is the common reference point for all signals on the RS-232C interface)	Both
8	DCD	Data Carrier Detect; this signal indicates that a modem at the other end of a telephone line has answered and is ready to go	To QX-10
20	DTR	Data Terminal Ready; indicates the QX-10 is ready for service	From QX-10

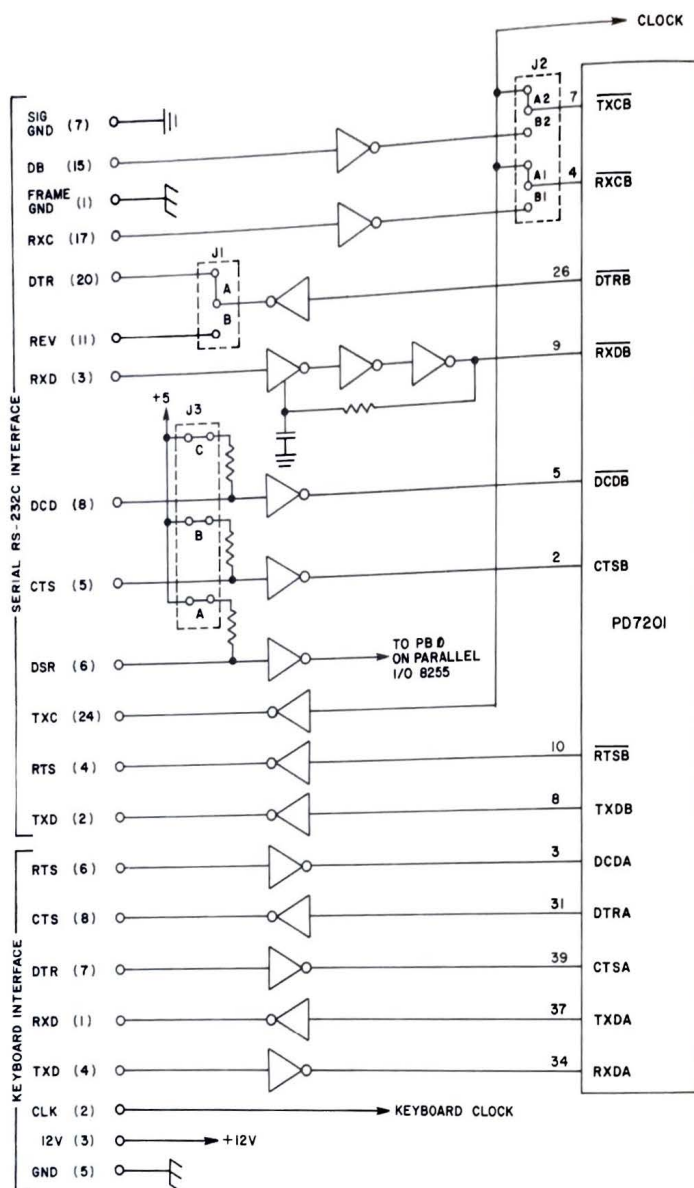
Chapter 6, "BASIC on the QX-10," provides information on serial baud rate programming. The keyboard speed should not be changed since the keyboard will be locked out if the QX-10 interface is not set to 1200 baud. However, it is theoretically possible to substitute another device, perhaps a computer, in place of the normal QX-10 keyboard for specialized control applications.

The Centronics Parallel Printer Port

The QX-10 printer port uses a 36-pin AMP 552742-1 connector, standard for most Centronics-compatible parallel printer interfaces. This connector can usually be found in mail-order catalogs and Radio Shack stores for those wishing to make up their own cables.

Printer interface cables should be kept as short as reasonably possible. Cables sold by Epson, for example, are only about three feet long, severely limiting printer placement. Longer cables can be purchased or made by those with reasonable soldering skills.

The original Centronics printer cables used a twisted-pair of conductors for each signal and control line to control electrical noise. One conductor of each pair was used for a "signal" ground and the other was used for a signal. The result was a 34 conductor cable, which is expensive and difficult to make. Commercial (and most homemade) cables today



Schematic diagram of serial interfaces used by the QX-10 for the keyboard and RS-232C interface.

are usually made with standard cables having no twisted pairs. These often provide only one or two ground lines and result in an electrically noisy cable that must be fairly short to work well.

The signal and connector pinout for the Centronics interface is described in Table 3-2. Its operation is electrically more complex than a serial interface because of computer-printer "handshaking" of control signals performed as each character is sent to the printer.

Parallel Data Transmission

Transmission of a single character to a printer is started when the QX-10 first checks the status lines to insure that the printer is turned on, has paper, and is operational. If this check is satisfactory, a character is placed on the interface data lines and the strobe line (on pin 1) is pulsed.

The printer reads the character when the strobe is sensed, and sets the busy signal on pin 11. This tells the QX-10 not to send another character until later. After the printer has finished processing the character (which might take as little as a few millionths of a second, or up to several seconds if the character causes a line to print), it removes the busy signal and pulses the acknowledge line on pin 10. The acknowledge pulse tells the QX-10 that the printer is ready for the next character, and the process is repeated.

Chapter 6, "BASIC on the QX-10," describes how single characters can be sent directly to the printer interface under software control.

Table 3-2. THE QX-10 PARALLEL PRINTER INTERFACE

Pin	Signal	Pin	Signal
1	Strobe	19	STR. Ground
2	Data 0	20	Data 0 Gnd
3	Data 1	21	Data 1 Gnd
4	Data 2	22	Data 2 Gnd
5	Data 3	23	Data 3 Gnd
6	Data 4	24	Data 4 Gnd
7	Data 5	25	Data 5 Gnd
8	Data 6	26	Data 6 Gnd
9	Data 7	27	Data 7 Gnd
10	Acknowledge	28	Ack. Ground
11	Busy	29	Busy Ground
12	Paper Out	30	Prime Gnd
13	Printer Select	31	Input Prime
14	Auto Line feed	32	Fault
15		33	Ground
16	Ground	34	
17	Chassis Gnd	35	Printer On
18		36	

Signal definitions for Table 3-2:

- Strobe—Forces printer to read the character on the data lines
- Data 0 to Data 7—Data lines carrying the character code
- Acknowledge—Signal from printer indicating that the last character was received and processed
- Printer Select—A signal line from the printer that indicates it is selected; normally, this line is always activated by the printer
- Auto Line Feed—Epson printers can use this line to cause the printer to automatically add a line feed after every line
- Input Prime—Also called "Init" and used to reset the printer to initial power-on conditions
- Fault—Indicates an error condition in the printer, usually that it is out of paper
- Printer On—Indicates the printer is turned on

The Light Pen Input

The light pen input on the QX-10 can be used to determine the relative location of the pen on the monitor screen. A light pen is actually a light-sensing pen, and is used with internal QX-10 electronics controlling the scan of the monitor screen.

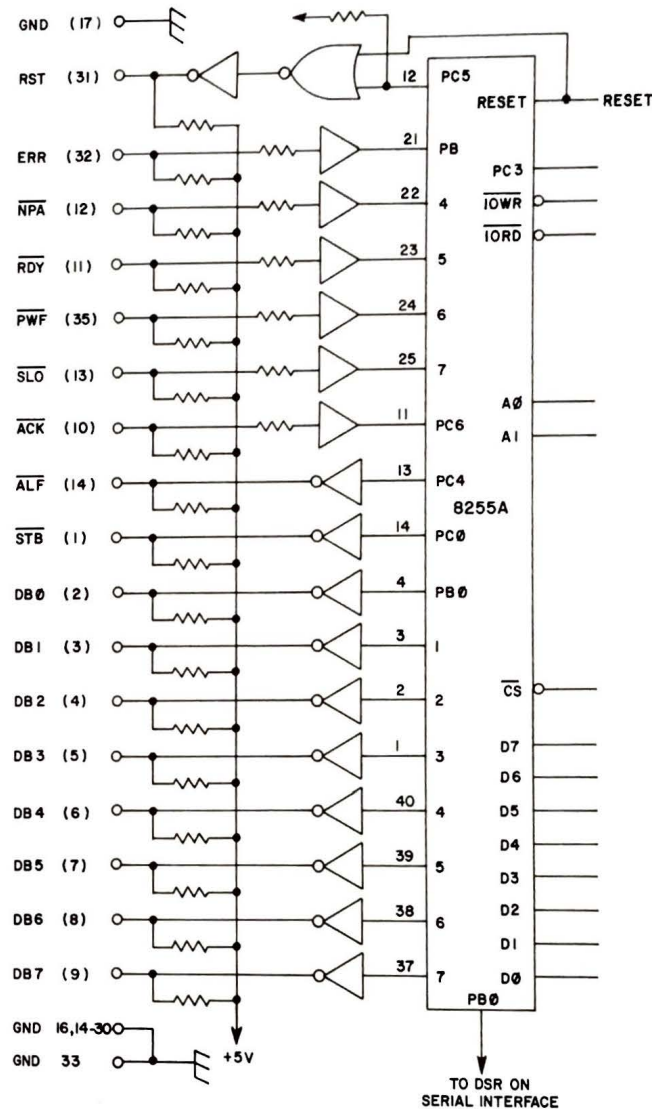
The QX-10 display controller maintains a set of counters that are used to control which dots in which character in which row are to be displayed at any given instant. This is all done in a single integrated circuit, an NEC μ PD7220, located on the video display controller card.

Since the controller knows which dots are being displayed and where on the screen they are located, it can determine the light pen position when it is placed over an illuminated dot. This is the basic operating principal of most light pens.

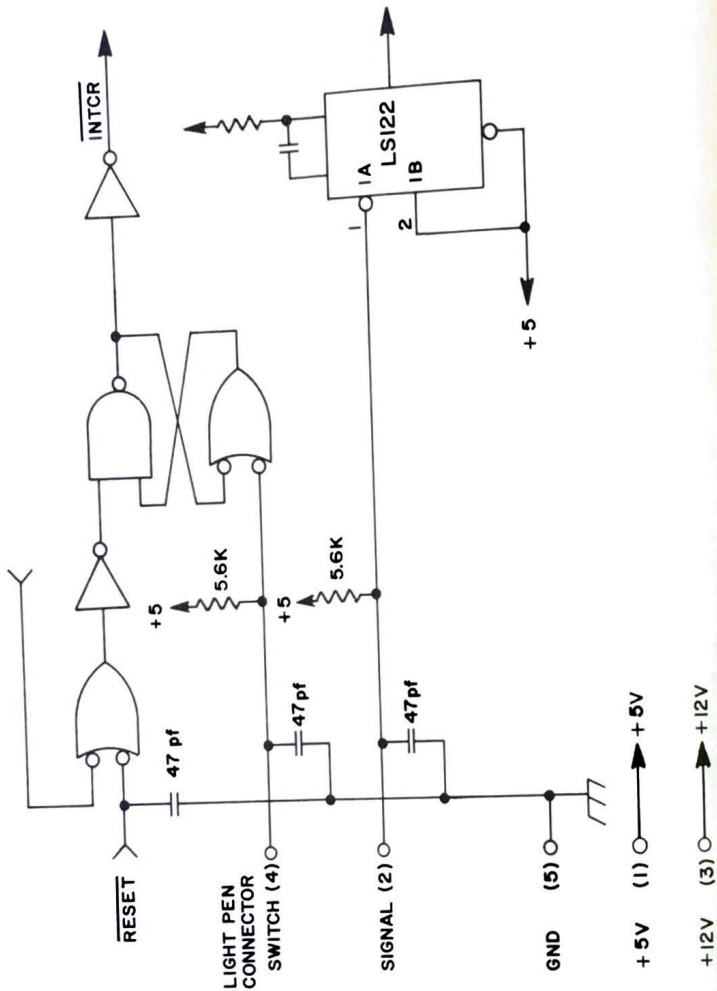
The QX-10 light pen connector provides two input signal lines. The first, intended for a switch contact closure to ground, is activated by pressing the light pen stylus against the CRT monitor screen at the location desired. An interrupt to the QX-10 CPU is generated by the switch closure. System software used to service the interrupt then interrogates the μ PD7220 for a valid light pen reading.

The second signal, from a photocell inside the light pen, causes the display controller to compute the location of the pen at the same time the switch closure interrupts the CPU. If the controller determines the location is valid (and not just noise from room light), a status bit is set and the pen location is saved for the Z-80A to read as a part of the light pen interrupt service routine.

This completes our internal technical look at the QX-10. Those with curiosity beyond this coverage should contact Epson for copies of the Technical Service Manual and QX-10 Programmers Manual.



Schematic diagram of Centronics compatible parallel printer interface on the QX-10.



Schematic of the light pen interface.

4



Operating Systems on the QX-10

Operating Systems are probably the least understood programs used on personal computers. Every computer, whether large or small, makes use of an organized group of programs collectively called an "operating system." On large computer systems, the operating system is usually called the "executive"; on very small computers it is called "the monitor."

Regardless of the size of the computer, the purpose of its operating system is to make the computer and its peripherals conveniently available to the user. Files on disks are carefully tended and protected from harm. Output for your printer is formatted and printed. Tables, pointers, and directories in memory are automatically updated and protected. Peripherals such as hard disks, graphics tablets, or plotters that need special attention are all managed and controlled, largely without human intervention, through the operating system.

Programs in the operating system are used to control the basic, elementary functions of a computer. Many of these functions are hidden from sight and go unnoticed. Others, usually on disk-based computers, are major programs that are called into use like a "regular" program. A hidden program might enter individual characters from a keyboard, read or write chunks of data from a disk, or check out the computer when it is first turned on. Still others maintain directories of disk files and control the use of memory and countless other jobs necessary to make a computer useful.

Books and disks for the operating systems and environments supported by the QX-10.

More "visible" fragments of operating systems include programs that format and prepare new floppy disks for use, compare files, or list the names of files on a disk. These programs are collectively referred to as "utilities." Some of them, such as assemblers, machine language monitors, special editors, and debuggers are used to write and troubleshoot complex software. Only people with technical interests will find them useful. Other utilities, such as disk copy or compare programs, are used more frequently.

Epson provides both CP/M and TPM II disk operating systems for the QX-10. CP/M (Control Program for Microcomputers), is an internationally popular operating system for small business and scientific computers. It was written in 1975 by Gary Kildall, who eventually formed Digital Research to market and further develop microcomputer operating systems. CP/M version 2.2 is provided with all versions of the QX-10 computer.

CP/M

CP/M is successful for a variety of reasons. It was among the first disk-based operating systems available for microcomputers. Although originally written for the 8080 microprocessor, the Z80, which came out about a year later, can also run 8080 programs. CP/M was therefore compatible with any computer system using either of these two very popular microprocessor chips.

But perhaps more importantly, CP/M was the first disk-based operating system to allow the power of the microprocessor to be fully realized, and yet be easily adaptable for nearly any 8080-, 8085-, or Z80-based computer system. A small section of code called BIOS (Basic Input/Output System) was designed to be adapted by computer manufacturers (and advanced hobbyists) to accommodate their particular computer system design. The source code for BIOS, unlike the main body of CP/M, was published for this specific reason.

Over the years since CP/M was introduced, several versions have been released. Programs that ran on the early versions (release 1.4 and earlier) must be modified to run on version 2.2. However, CP/M is largely the same disk operating system it was in 1975. The software base that runs with this operating system includes thousands of programs that range from accounting packages to structural engineering software.

TPM II

The second operating system used on the QX-10 is called TPM II (Transient Program Manager II). Valdoks uses this operating system to manage the QX-10 hardware. The TPM II was developed by Roger Amidon over a four-year period and is "upward compatible" with CP/M programs. (Programs written to run with CP/M will also work on TPM II, but not necessarily the other way around.) TPM II provides operational features not available on CP/M such as bank memory switching (to manage the 256K byte main memory in the QX-10), improved disk storage and packing, large file indexing, and many other improvements, refinements, and features needed by Valdoks.

TPM II is written in Z80 assembly language with the expanded Z80 instruction set, rather than with the older 8080 set used by CP/M. TPM II is provided by Epson only to support Valdoks. It was chosen because of operating features and performance advantages over CP/M.

VALDOCS

A high-level "operating environment" called Valdoks (VALuable DOCumentS) is provided with all QX-10 computers sold in the United States that are equipped with 256K bytes of memory and the HASCI keyboard. Valdoks was developed by Chris Rutkowski and Rising Star Industries. Conceptual work on it began in 1981. Epson owns all versions of Valdoks up to version 2.0 and licenses the later versions.

Valdoks represents a radical departure from the way software and hardware have traditionally been joined. The QX-10 was designed to specifically support Valdoks, and Valdoks was written to support the QX-10. Valdoks is a collection of both utility programs for file and printer control, applications programs such as text processors, and schedulers and graphics generators. All are written to work together in conjunction with the specially designed HASCI keyboard.

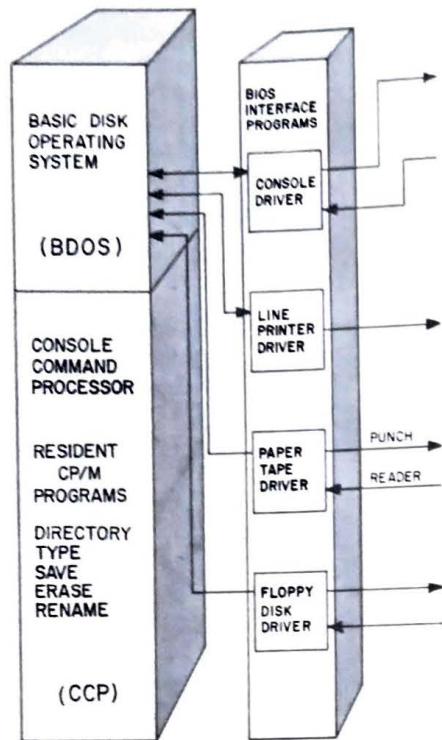
Valdoks is totally consistent within itself. Utility and applications programs that run under Valdoks operate the same way. Such is not the case when independently written commercial programs are combined to form a new system. Such programs, which are nearly always written without considering other programs on the market, often use conflicting terminology, control methods, and file types.

Programs included with Valdoks are menu driven with easily understood descriptions within the menus. All have detailed instructions available at any time at the stroke of the HELP key. (In fact, nearly 40K bytes of disk storage are used to contain all the help files in Valdoks version 1.) Valdoks is so advanced and streamlined that an operating manual is not strictly required, since it is available from the computer through the HELP key. Valdoks is covered in detail in Chapter 6.

CP/M TECHNICAL DETAILS

CP/M is logically split into three general sets of software. BIOS (Basic Input/Output System) is a collection of hidden programs that control input and output to the computer system console (the QX-10 keyboard and screen), line printer, disk drives, paper-tape reader and punch (the serial port on the QX-10), and other peripherals that might be attached to the computer.

BIOS is the only part of CP/M that is machine-dependent. This means that none of the rest of CP/M needs to know anything about the type or brand of computer that is running it. The source code of BIOS is published by Digital Research so that manufacturers can



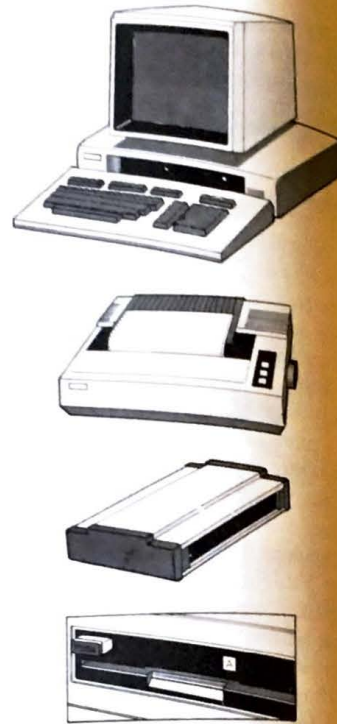
Hardware and software relationships in CP/M operating system.

modify it to run on their own computer systems. (Advanced hobbyists also frequently do the same on their homebuilt systems.)

All data going to or coming from peripherals such as the floppy disks, printers, terminals, screens, and keyboards of computers using CP/M passes through the BIOS. BIOS is actually a software interface that connects BDOS (Basic Disk Operating System) to the specific hardware design of the computer and the equipment driven by the computer.

BDOS is the actual core of CP/M and controls all disk operations, such as reading and writing blocks of data, finding (or creating) space on a disk for information to be stored, maintaining file directories on the disk, and hundreds of other "invisible" chores related to these tasks.

The Console Command Processor (CCP) is the part of CP/M that "talks" to the user.



The main function of the CCP is to collect commands from the console (keyboard and screen of the QX-10), check them for proper syntax (make sure they were typed and formatted correctly), then pass them to BDOS where they are executed.

The QX-10 is supplied with a program called CP+ that replaces the original CCP and makes CP/M more "friendly" (easy to use) by controlling operations with a series of "menus." There is more on CP+ in Chapter 5.

The first illustration in this chapter shows the relationship between the various CP/M programs and the QX-10 computer. Notice how all computer input and output passes through BIOS and into BDOS. It may then pass back out through BIOS to another peripheral. One example of this taking place is the COPYDISK command. (Data is read from one disk drive through BIOS into BDOS, then sent back through BIOS to the other drive, thus making a copy of the original disk.)

CP/M Utility Programs

The QX-10 version of CP/M provides 19 utility programs. Six of them are "resident," and the remaining thirteen are "transient." Resident programs are those that are always in memory and can be used without having to pull them in from the disk every time they are needed. Transient programs are those that are normally kept on disk and have to be read into memory every time they are used.

The six resident programs are DIR (print the directory—a list of all the files—of a disk), TYPE (list a text file from a disk on the screen), SAVE (save data or programs onto a disk), ERA (delete or erase a program or file currently on disk), REN (rename a file on disk), and USER (provide 15 "user areas" in the disk directory—see "User Areas" later in this chapter.) These programs can be used by applications programs since they are in memory all the time. Such is not the case with the transient utility programs.

CP/M Transient Utility Programs

Transient utility programs are usually larger, less often used programs that are kept on disk because they consume so much memory. These utility programs cannot normally be used while the computer is running another program such as BASIC because of memory conflicts.

Transient programs and other data stored on floppy disks are bundled into blocks of information called "files." Files are referenced by, logically enough, "filenames." In both the CP/M and TPM II operating systems, a filename can consist of up to eight characters, and is followed by a period and a three-letter "extension."

An attempt is usually made to give a descriptive name to files to indicate the contents (of a data file) or the intended application (in the case of a program). Filename extensions are used in some cases by both CP/M and TPM II to indicate what can be done with the file. For example, a ".TXT" or ".DOC" file cannot be run as a program, since these are, by

convention, text or document files containing no program. A listing of some common filename extensions used by TPM II (CP/M also recognizes some of them) is given in Appendix H. Valdocs, by comparison, allows considerably greater freedom in assigning filenames, as will be seen in Chapter 6, "Valdocs.")

Several of the original CP/M utility programs supplied by Digital Research have been combined into easier-to-use programs by Epson or are not supplied as a part of the CP/M software package. All original CP/M utility programs are mentioned here, regardless of whether they are supplied by Epson. More thorough descriptions of the original CP/M utilities can be found in the *CP/M Primer* (supplied with each QX-10 by Epson) and the CP/M literature written by Digital Research. Those readers with specific technical interests who need the missing utilities should contact their Epson dealer.

STATUS provides information about disks, files, or peripherals attached to the computer. It can report the remaining disk space, sizes of files, and file types, and can change the input-output assignments of your computer. (This feature is very important if, for example, a serial printer were to be used in place of a line printer connected to the parallel port. STAT alters the normal destination (or source) of data by changing the logical connections between BDOS and the various interface programs in BIOS.)

PIP (Peripheral Interchange Program) is used primarily to move files from one disk to another (copy them). It can also be used to move a file from any peripheral device to another. For example, a file could be sent directly to a printer over telephone lines through a modem attached to the serial port on the QX-10.

SYSGEN, which was originally provided with CP/M, is not supplied by Epson. It is functionally replaced by the Epson-supplied utility COPYDISK, and is listed for reference only. It is a program used to copy CP/M from one diskette to another. SYSGEN could be considered a specialized form of PIP that is useful only to replicate CP/M itself.

MOVCPM, which is also not supplied by Epson, is a program used to copy CP/M that is similar to SYSGEN, except that it specifies how much memory the newly copied CP/M is allowed to use. This has the effect of protecting part of memory from CP/M and is useful primarily to those with technical interests.

ED is a "system editor" that can be used to create files for use with ASM (the 8080 microprocessor assembler). When CP/M first came on the market, there were no word processors or text editors available for disk-based systems. ED, while useful, is limited and somewhat cumbersome to use by today's standards.

ASM is used to write programs in 8080 assembly language. Although this assembler generates 8080 code that will run on a Z80, many OPCODES (operation codes) for the Z80 (used in the QX-10) are not available. This program is of use only to those interested in low-level high-performance software within the limits of the 8080 microprocessor instruction set.

DDT (Dynamic Debugging Tool) is a program used to troubleshoot software during development. As with ASM, most non-technical people will find little use for it.

LOAD and DUMP allow you to load machine language files from disk to memory or to display them on the screen using DUMP. These are more software development tools that are useful only to people with technical interests.

FORMAT (not provided by Epson) is functionally replaced by COPYDISK. It is a program that is usually provided by a computer manufacturer to format diskettes. As such, it is not actually a part of CP/M, although it falls in the utility category. (See "The Floppy Disk" elsewhere in this chapter for more about disk formats and the need for them.)

SETUP, which is an Epson supplied utility, allows a number of QX-10 functions to be changed. Among these, disk error diagnostic messages can be turned on or off, either the ASCII or HASCI keyboards can be selected, and keyboard auto-repeat can be disabled. The serial port can be programmed for line protocol, data bits and parity, and baud rate. It also allows automatic CP+ entry at power up to be disabled.

TIMESET is supplied with the QX-10 by Epson. It allows the internal calendar-clock to be set from the keyboard.

COPYDISK, an Epson supplied utility, can format and copy a disk. The copy program can be set to copy just the operating system (CP/M itself), all the data files, or both. Operation is menu driven and fully automatic. Once started, the program runs to completion without interruption. This program does not verify that the disk has been copied correctly.

INSTALL+, a utility supplied by Taurus Software, is used to enable CP+ operation. Chapter 5 describes CP+.

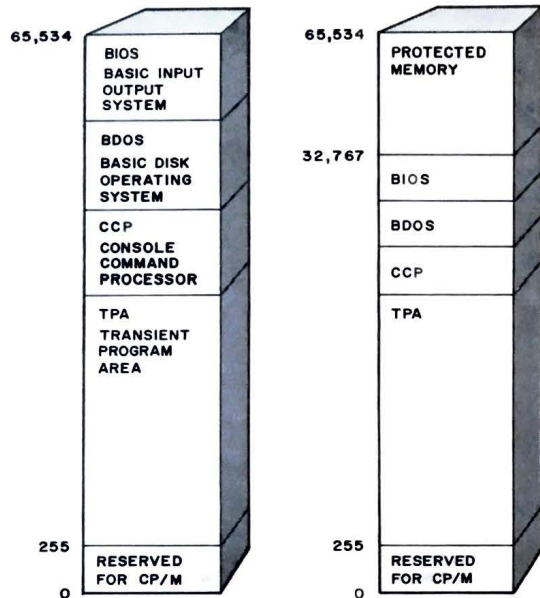
Memory Allocation on the QX-10

One of the primary functions of an operating system is to manage the memory available on the computer. CP/M is capable of managing only 64K bytes (actually 65,535 bytes) of memory, despite the 256K byte main memory available in the Valdocs version of the QX-10. The reason for this is that CP/M was written for the 8080 microprocessor, which can directly address only 64K bytes of memory.

Special hardware inside the QX-10 allows the use of four separate blocks of 64K bytes of memory and a separate fifth bank of 2K bytes of CMOS RAM with battery backup. One of the reasons that Valdocs uses TPM II rather than CP/M is that TPM II provides bank memory controls, whereas CP/M version 2.2 does not.

In the QX-10, CP/M partitions 64K bytes of memory into five distinct sections. Figure 4-2 shows the general arrangement. Starting at the bottom of memory, CP/M reserves 256 locations, addresses 0 through 255, for its own use. These locations hold system variables. It is important to realize that these locations are the same on every computer running CP/M; the numbers stored at these locations may change depending on input-output assignments, memory capacity of the computer, and other variables, but the logical values assigned to these addresses are constant.

At location 256, the TPA (Transient Program Area) begins. Any program that runs under CP/M is loaded and run from this address, and can be considered a transient program. Microsoft BASIC, for example, is loaded starting at location 256 and extends upward. BASIC programs (programs written in BASIC) are loaded above the point that BASIC stops. The transient utilities mentioned earlier operate in the same area of memory, and for that reason are usually not available for use by applications programs run under CP/M.



Memory map when using CP/M 2.2. MOVCPM utility can relocate CP/M below the top of a 64K byte block of memory, protecting the memory above CP/M from normal use.

CP/M itself is loaded from the top of usable memory. In the case of the QX-10 with 64K bytes of memory, BIOS is loaded from location 65,535 and extends downward.

BDOS follows, and below that, the Console Command Processor (CCP). The CCP can be overwritten if the transient program needs the memory space, since it normally is not used when an application program is running. BDOS can also be overwritten if no disk or file accesses are required by the application program. If the CCP or BDOS have been overwritten, CP/M will automatically reload them from the disk if control is returned to CP/M through a special program called the "warm boot." (The term "boot" or "booting" refers to bringing a computer to life by running a sequence of programs, each loading and running a longer and more complicated program until the operating system is finally in place and takes charge.)

In the early days of computing, a short program was entered into a computer by way of switches on a maintenance console. This program loaded a paper tape, which then read a program from a magnetic tape, which then loaded and started the operating system program.

A "cold boot" reloads the entire operating system as if from scratch, and a "warm boot" brings in only selected portions of a program.

The utility program MOVCPM is used to make a new copy of CP/M, but with an artificial upper address limitation set according to specific needs. The original purpose of MOVCPM was to allow CP/M to run on computer systems with memory capacities less than 64K bytes. Since the QX-10 has a full 64K byte memory for CP/M, the need for MOVCPM is diminished.

But MOVCPM can also be used to provide a block of memory above BIOS that CP/M "doesn't know about," and will be unused by CP/M and any application program that doesn't specifically use this memory area. The only detrimental side effect is that memory thus reserved is actually taken out of the TPA because CP/M is simply moved down the address scale. The "protected" upper memory can still be used as always, but CP/M, as an operating system, does not know it is there, and will leave it alone.

TPM II TECHNICAL DETAILS

TPM II is an operating system that is totally compatible with CP/M and about 20% faster when performing equivalent tasks. It also provides software controls for bank switching the 256K bytes of main memory in the QX-10, as well as other features, services, and utility programs that are not available on CP/M version 2.2.

Originally supplied only to support Valdocs, TPM II is not presently available as an independent operating system, although this may change in 1984. In nearly all cases, TPM II can be used interchangeably with CP/M. From a user's point of view, there are no operational differences between the two operating systems. TPM II and a newer version, TPM III, are available through Rising Star Industries.

Any program for use with either CP/M version 1.4 or 2.2 can be run directly using TPM II. CP/M versions 1.4 and 2.2 have a number of differences, and many programs written for the 1.4 version cannot run on 2.2. If a "CP/M compatible" program is encountered that does not run properly on TPM II, it was probably written for CP/M version 1.4.

These can be run on TPM II by simply changing the filename extension from .COM to .SYS. TPM II automatically switches to the CP/M version 1.4 mode when .SYS programs are run, and to CP/M version 2.2 when .COM files are executed. Programs are then run in the same way as on the respective version of CP/M. Regardless of which mode TPM II is currently running, all of its expanded features (many of which are described below) are available for use by any program running under it.

TPM II Resident Utility Programs

A total of ten resident commands can be executed by TPM II. The Directory, ER, SA, Rename, and TY commands are identical in form and function to the five offered by CP/M version 2.2. The other five TPM II utilities are either not available on CP/M or must be run

Table 4-1. CP/M and TPM II Resident Commands

CP/M	TPM II	Description
DIRectory	Directory	List files contained on current disk drive
ERase	ERase	Delete a disk file
—	Mask	Display only files of specified protection levels
—	Protect	Protect and control file use
REName	Rename	Change the name of a file
SAVe	SAve	Save a memory image as a file
—	SPace	Report remaining storage available on specified disk
—	Time	Print current time and date on the QX-10 console
TYPE	TYpe	Print a text file on the console
—	User	Print current user number on specified disk drive (see text for information on user numbers)
USER N	N:	Select user area number N

Note: These commands can be executed by typing only the letters shown in capital letters.

as a non-resident utility program. Table 4-1 shows a comparative listing of all CP/M and TPM II resident commands along with a brief functional description. TPM II resident commands can be abbreviated to as few letters as will allow them to remain unambiguous.

User Areas

CP/M version 2.2 and TPM II both provide several features of larger computers. Whereas TPM II's file directory system will allow about 1000 entries, floppy disk storage available on the QX-10 cannot possibly hold that many. Hard disk drives available for the QX-10 can, and so the concept of "user areas" or "user numbers" is provided to help keep track of the large number of files that can be maintained in such storage facilities.

Both CP/M version 2.2 and TPM II provide user areas. CP/M offers 15 areas controllable from the console compared to TPM II's 255. These allow files associated with a given task to be collected and used exclusively. Files in one user area are not available in another, providing a degree of protection from accidental destruction while at the same time "uncluttering" the directories. Files not belonging to a particular user area are not listed in the directory for that area.

Perhaps an example would help to illustrate the significance of this feature. Suppose a collection of files containing market performance data and evaluation programs for a stock portfolio were kept in user area 1, and chapters for a book being written were in user area 2. When logged into user area 1, only the stock market portfolio files are accessible, and thus it is not possible to unintentionally destroy or alter the contents of the book files that are saved in user area 2. The effect is nearly the same as having two disk drives—one for user area 1 and one for user area 2.

When TPM II is initialized, the system defaults to user area 0, making all files on the disks available, as does CP/M. A TPM II user area is selected by typing the command N,

where N is a user number between 0 and 255. CP/M selects user areas with the command USER N, where N is a user number between 0 and 15. User areas are a "nice to have" feature that need not be used or worried about until needed.

TPM II File Protection

Files on TPM II can be directly protected from alteration or deletion using file protection. Seven protection levels can be assigned (compared to CP/M's single protection, READ ONLY), and are listed in Table 4-2. The protection level for each file is shown in parentheses next to the filename in directory listings. The purpose of file protection as used in TPM II is to prevent the loss of files, rather than to keep someone from snooping into them.

Protection is applied to any file already saved by typing the command:

P FILENAME:EXT <N> (CR)

where:

P is the command for protection

FILENAME:EXT is the name of the file and TPM II extension to be protected

<N> is the protection level (between 0 and 7) enclosed between the brackets shown

(CR) is a carriage return used to end the command

The Mask command is used in conjunction with the Directory command to display only files with a particular protection. The normal or default mask value is 0, allowing all files to be displayed. A listing of files with a protection code greater than 5, for example, can be obtained by first setting the mask by typing:

Mask > 5 (CR)

When the directory is listed after this mask has been set, only the files with protection levels greater than 5 will be displayed. A mask remains in effect until it is changed to another value or until the computer is turned off.

Table 4-2. TPM II FILE PROTECTION

Code	Protection Meaning
7	File only appears in the directory and cannot be referenced
6	File may be executed, but not erased
5	File may be read, but not erased
4	File may be appended (added to), but not erased
3	File may be appended or erased
2	File may be replaced or erased
1	File marker only, no protection
0	Default value, no protection

TPM II Transient Utilities

Although the release version of TPM II was not available at the time of this writing, many very useful utilities were in preparation. Several utilities have comparable functions to those found in CP/M; others are completely new and are used in conjunction with Valdocs. The TPM II transient utilities available in midsummer of 1983 are listed in Table 4-3.

The ALTDISK utility (not available at the time of this writing) is of particular interest. It allows most CP/M diskettes to be used directly on the QX-10 without regard to their original system configuration or format. Such disks can be copied into the TPM II standard (and thus QX-10) without problems.

Files from other computer systems (such as the IBM PC) can be read into the QX-10 for processing or use. Files can also be written to disk in a variety of computer system formats (the Osborn I computer is another example). The primary restriction is that the disk must be soft sectored and a derivative of the IBM 3740 format standard.

Other programs convert text files directly into indexed Valdocs files and convert Valdocs files for use by other word processors. Utilities for use with Valdocs are also available to help restore indexes that have been damaged or destroyed.

Table 4-3. SOME TPM II TRANSIENT UTILITIES

Name	Function	Nearest CP/M Equivalent
BAUDSET	Set serial speed	SETUP
BILD	Rebuild, if possible, a Valdocs Index	Not available
COMPARE	Compare two files	Not available
CPYD	Copy a disk	COPYDISK
D7	A disk examine/change utility program	Not available
FILES	Display disk directory in hexadecimal	Not available
IOMOD	Change input-output assignments	STAT
KEYSWAP	Change ASCII/ASCII keyboards	SETUP
LIST	Print a sorted list of files and file length contained in current User Directory	Not available
MODEM 7	A version of MODEM7 in public domain for moving files over telephone lines	In public domain
SET-TIME	Print time and date or set time and date of internal calendar-clock	TIMESSET
SYSGEN	Copy operating system	SYSGEN
VERS	Print TPM II version, revision and serial number on QX-10 screen	Not available
WIDTH	Set number of columns in directory display	Not available
ZDDT	Dynamic debugger	DDT
ZPIP	Peripheral interchange program	PIP

Note: This list is incomplete since TPM II was not released at the time of writing.

COMPARING CP/M AND TPM II

These two operating systems share many traits. Both use the same command syntax, file organization, and structure. Both are constructed in three major blocks. CP/M's, BIOS, BDOS, and CCP correspond directly to TPM II's PIOS (Physical Input/Output System), LIOS (Logical Input/Output System), and ICP (Interactive Command Processor).

TPM II has an advantage in using Z80 assembly language rather than CP/M's 8080 code. This makes it faster and more efficient while providing greater functionality. TPM II's major operational advantages are the ability to run both CP/M versions 1.4 and 2.2 programs, and to be able to manage the full 256K byte main memory of the QX-10. (Little or no CP/M version 2.2-based software is presently available to make use of memory sizes above 64K bytes.)

Another major advantage of TPM II is the technical implementation of the USER command. CP/M version 2.2 has provided minimal support for this command (transient programs generally ignore user areas). Separate copies of files must be saved in every user area in which they are used. This consumes an enormous amount of disk space and makes it difficult to update such files. TPM II allows a single file to be listed in multiple directories. Thus, only one file need be updated for all directories.

Despite the CP/M compatibility of TPM II, it is possible for CP/M-based programs to be written that cannot be run on TPM II. Compatibility between these operating systems is possible because user programs are expected to make standard calls for operating system services using a specific technical protocol. Undoubtedly, in a software base as large as CP/M's, there are programs that have violated this arrangement. TPM II cannot possibly run such a program with a nonstandard protocol since internally it is unique and depends entirely on the standard software interface to manage and run CP/M programs.

Many of the "fancy" TPM II utility programs mentioned earlier could easily have been written for CP/M, and many may actually run just as well under it. They have been mentioned here not to show TPM II at an advantage, but because they are either an integral part of TPM II or are distributed with it.

TPM II is a powerful operating system that uses the CP/M format. Although it has been tuned to specifically support the QX-10 computer and Valdocs software, many users may choose it over CP/M because of the flexibility and increased performance it offers.

THE FLOPPY DISK

A floppy disk got its name because data is stored on a flexible mylar disk rather than the flat metal platter used in rigid disk system. The disk itself is coated with magnetic iron oxide that shows a sharp distinction between its magnetized and non-magnetized states, which is the primary difference between magnetic iron oxide and coatings used for audio tapes.

The outer protective cover of a floppy disk provides some rigidity for handling and

protection for the disk inside. Holes in the cover provide access to the disk for the disk drive head and spindle. A notch cut in one side of the cover gives the disk controller permission to write on the disk. The disk is "write protected" by covering the notch with an opaque piece of tape (it is optically sensed). Barring a hardware failure, a write-protected disk is safe from being erased or modified by the computer.

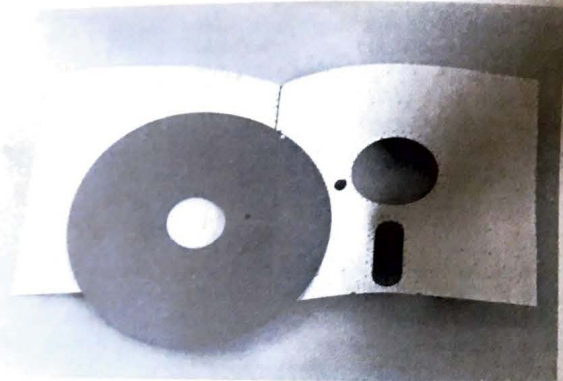
The mylar disk itself is typically .002 inches thick, and its flatness is carefully controlled. Better quality diskettes have a reinforced hub in the center to limit the wear caused by loading and spinning the disk. The disk fits loosely inside the protective cover, which is lined with a specially treated paper that continually cleans and lubricates the disk as it turns.

The disk spins at six revolutions per second. Maximum track latency (the longest it will take to read or write a given block of data on a track) is one-sixth of a second; average latency is half that.

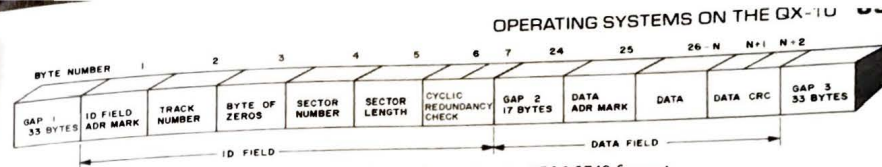
The small hole near the hub is used to provide a physical "index." Sophisticated disk controllers make use of this hole to determine whether the disk drive is actually turning. Utility programs can measure the speed of the drive spindle by using the index hole as a timing reference.

Disk Formats

Data is stored on a disk in a series of concentric rings. The QX-10 uses 40 of these rings, or tracks, on each side of the disk. The tracks are numbered from the outside (track 0) inward (to track 39). The purring noise made by the QX-10 disk drives as files are referenced is caused by the head stepping across the disk to access other tracks.



The inside of a typical floppy disk. The specially treated paper liner helps keep dust from contaminating the disk surface.



The format of a single sector as recorded on a floppy disk using the IBM 3740 format.

Each track is logically divided into a series of segments called "sectors." A sector is divided into two fields. The first, the ID (identification) field, comprises 7 bytes that tell the controller which track and which sector within that track is about to be referenced. Other bytes in this field indicate the sector length and provide a "magic number" that allows the controller to determine whether the data just read was correct. (This is called the CRC or Cyclic Redundancy Check number.)

Following the CRC number, a gap of 17 bytes is provided to allow the controller time to decide whether the sector and track were the ones it wanted to read or write. A data address mark follows, indicating that the following bytes are actual data to be read. The last two bytes in a sector make up another CRC to ensure that the data is read correctly. A gap of at least 33 bytes separates sectors.

The disk format just described is based on the IBM 3740 format. The most usual format difference from one computer system to the next is simply the number of bytes contained in each sector. If a disk controller is set up to read 256 bytes in each sector, and it finds 512, the CRC number will not match and a disk read error will be generated.

Before a disk can be used by a computer to store information, every sector position in each track must be recorded in the sector format just described. The data written during this operation can be "zeros" or some other pattern; it is only used to fill each sector as a placeholder. This operation, which consists of writing the sector formats throughout an entire disk, is called "formatting."

Formatting is an operation that makes electronic "bins" on the disk where blocks of data can be stored. Without these electronic bins, it would be impossible to know where information could be found. Formatting also permanently erases any information previously saved on a disk, and after a disk has been formatted, there is no hope of recovering anything written on it before.

It is sometimes possible to recover full use of a disk that has been slightly damaged by formatting it again. If the damage was largely superficial or if the disk was exposed to stray magnetic fields, simply reformatting it (after any recoverable information has been copied onto another disk) may restore it to full use.

The disk format used by the QX-10 consists of 256 byte sectors on tracks 0 and 1, and ten 512 byte sectors on tracks 2 through 39. Both sides of the disk are used for data storage. Tracks 0 and 1 are used to store the operating system (either CP/M or TPM II) and the directories. The remainder is considered the actual disk storage capacity.

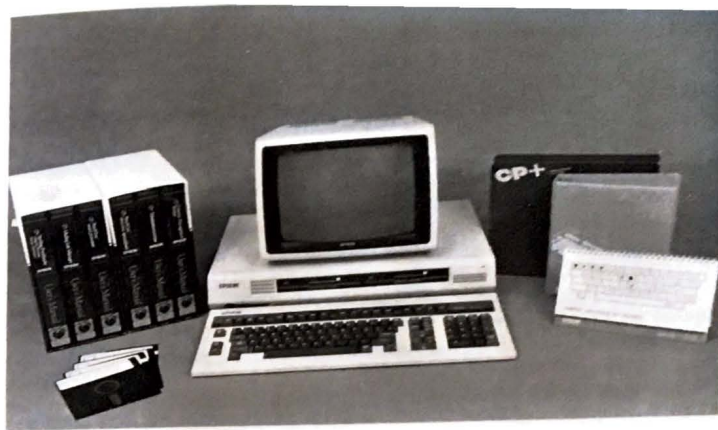
Care of Floppy Disks

The rules for the proper care of floppy disks are quite simple:

1. Do not touch the actual disk through the oblong slots in the cover. This may put an oily fingerprint on the surface, which will attract dirt, causing undue wear on your disk drive head and eventual data corruption on the disk itself. It is a good idea, before inserting a diskette into a drive, to center the hub hole in the center cutout of the jacket, touching only the inside edge of the hub hole.
2. Provide a dust-free storage case for disks not in use. Keep floppy disks in the dust jacket provided with each disk when they are out of the storage case. The rule here is simply "keep them clean."
3. Do not bend a floppy disk, or do anything to it that will cause it to be bent. This includes placing anything on top of it, writing on it forcefully, or placing it on radiators or in direct sunlight where heat may cause the plastic diskette to warp. Labels should be written first, then placed on the jacket for identification. Put labels in a corner of the diskette so that when they are removed there is less danger of bending the disk inside. (The disk does not reach into the corner areas.)
4. Keep anything magnetic away from your disks. This includes horseshoe magnets, dynamic microphones, tape degaussers, screwdrivers or other tools possibly magnetized in use, and power transformers for electronic equipment, which may be out of sight under a thin plastic case. (The QX-10 is safe from this problem.) Avoid placing floppy disks near the deflection coils of the tube at the back of the CRT monitor. Powerful magnetic fields are often generated from these coils.

If these simple rules are followed, a typical floppy disk will provide reliable data storage for years of service.

5



Applications Programs on the QX-10

Applications programs are those programs that cause a computer to perform a useful job, as opposed to utility programs that are used for computer housekeeping. Application programs can turn an ordinary computer into an accounting machine, a text processor, or the flight deck of an interstellar battle cruiser. There are no strict rules for what constitutes an application program. An application program for one user may be considered a utility by another.

This chapter provides a quick look at some of the application programs available for the QX-10 through Epson.

Moving or converting programs intended for other computer systems can be difficult. Unless there are compelling reasons otherwise, it is a good idea to purchase programs already configured for the QX-10. Program conversion can consume large amounts of time, especially if unfamiliar technical problems are encountered.

In midsummer of 1983, Epson provided CP+, a program used to make CP/M into a menu-driven operating system, and four volumes of software from PeachTree Software, Inc.

The QX-10 and applications programs supplied by Epson.

In Europe, where the marketing thrust of the QX-10 has been entirely on CP/M-based systems, a wide variety of QX-10 configured software is advertised.

APPLICATION PROGRAM INSTALLATION—A FEW NOTES

The process of "installing" a program on a computer could be compared to the installation of a garage door. Although standard doors can be selected that nearly fit just as they come from the manufacturer, considerable modification is often required to make one suitable for your particular garage.

Programs that are to be installed in the QX-10 computer system must be written for either an 8080 or Z-80A computer. Data bases (information in the form of text or numbers) can come from nearly any computer system, but there are special (and different) problems related to the installation of a data base.

Most applications programs and data bases suitable for QX-10 use will be CP/M compatible. This ensures that at least the "outside fit of the garage door" is close. Application programs that make use of any special features of a particular computer or terminal, such as graphics or unique hardware, will probably have to be extensively modified before they can run on the QX-10 (or any other computer, for that matter).

Programs that make use of special features, such as calendar-clocks, special interfaces, or joy-sticks will present problems whenever they are moved to new computer systems. There is no standard by which such features are built into computer hardware, and installation difficulties may well preclude the use of such programs by all but "techies."

Applications programs in general, but word processors in particular, must know how characters are displayed on the computer monitor and how they can be controlled. This often goes beyond simply relying on the BIOS section of CP/M to display the text.

When the QX-10 is used with CP/M, the display is made to act like a Televideo 920 CRT terminal. (The technical term for this is "terminal emulation," which means that the screen will respond to the same control characters as a particular terminal.) When word processors and other programs that rely on special screen controls are installed on the QX-10, they must be set for the Televideo 920 or the display will not behave correctly.

If all the preceding seems complex and technical, it should. Program installation is not easy. If the project is "for fun," it can be considered very educational, although it is time consuming and frustrating in many cases. But if a particular program is needed for work, considerable time and effort will be saved by purchasing a ready-to-run program on the QX-10. Additionally, a professionally installed program has the advantage of being thoroughly checked out, and gives some assurance that there are no "loop holes."

SOURCES OF APPLICATIONS PROGRAMS

Nationwide, there are many dealers and distributors who specialize in selling software already configured for use. Among the largest is Lifeboat Associates, 1651 Third Avenue, New York City, NY (10028). Other software vendors can be found through advertisements

in various computer magazines such as *Byte*, *Microcomputing*, *Creative Computing*, and others.

Other software sources and topics of interest to Epson computer owners is available through the Epson Computing Group, an independent computer-users group not affiliated with Epson. It publishes a monthly newsletter and has an electronic bulletin board (a computer dial-up number that you can call if you have a modem for your QX-10). (The Epson Computing Group, 400-2 DeYoung, Marion, IL 62959, (618) 993-3600.) The Epson Connection is another independent newsletter, publishing news and articles about Epson computer products. Published bimonthly as a User's Group Newsletter, the Epson Connection is available for \$18 per year. (Epson Connection, P.O. Box 14027, Detroit, MI, 48214.)

PEACHTREE SOFTWARE AVAILABLE THROUGH EPSON

When the QX-10 was released for sale in the United States in late April of 1983, the only applications programs available were a set of six PeachTree volumes. These included PeachText (the word processor used by the author to write this book), Spelling Proofreader (a spelling checker), PeachCalc (an electronic spreadsheet program), Mailing List Manager, Telecommunications (for electronic mail), and the Calendar Management System. Since then, Epson has not released or endorsed further CP/M applications programs, and has stopped officially distributing the Telecommunications and Calendar Management System packages.

PeachTree programs are sold with no Epson or QX-10 specific information, and the manuals are written more for experienced computer users than for those just starting out. All PeachTree programs have a tutorial section at the beginning to help familiarize new users with the software, but very little "hand holding" is done. Depending on user experience, some of the above programs can be mastered with a couple hours of concentrated practice and study; others will take longer.

All PeachTree manuals are supplied in green 1½-inch three-ring binders, that make it convenient to add notes to the manual.

The PeachTree software packages supplied for the QX-10 are also available for a wide variety of computer systems, including the IBM PC. These programs offer good operating characteristics (they are menu driven) and all are documented and operated in the same general manner. Procedural experience with one PeachTree program is generally applicable to the others.

Text Editing—PeachText

PeachText is a quality text processor that is easy to use, yet has powerful features such as print spooling, form-letter capabilities, and other advanced file handling features that can lie dormant and out of the way until needed. The editor comes with an exceptionally easy to use thesaurus (a program that provides alternative words) that can be installed on the editor disk if desired.

When initialized, PeachText presents a well-organized menu that allows a choice of eight functions and notifies the user of the type of printer and terminal it is currently configured to use. (PeachText can drive several types of printers, including daisy-wheel, letter-quality printers such as the Transtar 130 and Epson FX-80 dot matrix printers, which are described in Chapter 7.) The terminal configuration information is a holdover from CP/M, which allows a variety of terminals to be used.

Operational choices from the introductory menu include **EDit** a file, **PRint** a text file (PeachText calls these "document" files), **COpy** a document, **DElete** a document, **REname** a file, **DIsplay** a directory, **??** for help, and **ENd** to exit PeachText and return to CP/M.

The directory, file copy, rename, and delete commands are similar in function to those provided by CP/M, but are automatic and easier to use. File names used by PeachText have the same eight-character length limitation as those in CP/M, and are given the extension .DOC for DOCUMENT to distinguish them from files made by other editing systems. (An example of a document filename might be "CHAPTER5.DOC". The three letters to the right of the period, DOC, make up the filename extension.)

Whenever an existing document is re-edited, PeachText automatically changes the original filename extension to BAK for backup, providing a single backup copy of every file edited prior to the latest changes. (If there is a previous "BAK" version of the file being edited, it is replaced when the current file is stored at the end of the session.) This quite obviously doubles disk storage requirements, but provides limited protection from catastrophic loss due to machine or operator error.

Editing Commands

The PeachText editor offers a powerful set of commands for editing text and reviewing files. Conditional operators (such as the **IF** statement), variables, and counters control the merging of files during printing, which allows good flexibility when sending letters to a mailing list.

The cursor location is controlled with the QX-10 edit cluster arrow keys. Text can be scrolled forward or backward a line or screen page at a time. The cursor can be sent to the top or bottom of the document using various function keys at the top of the keyboard.

Text deletion in a file is performed by the keyboard backspace key. Individual characters are deleted by the edit-cluster delete key, which removes the character under the cursor, then moves text in from the right of the screen to replace it. Function keys can be set to delete one word or everything on the line that is to the right of the cursor.

Text entry can be done on a replacement basis (anything typed simply replaces the characters under the cursor) or by "insertion," whereby keyboard data is inserted in the character position to the left of the cursor, moving any text under and to the right of the cursor to the right. A "full" insert mode is selectable from a function key to "make a hole" in the current text file for large insertions of text.

PeachText allows blocks of text to be "marked," then moved, copied, deleted or extracted, and placed in another file.

The search function can scan for partial words or complete phrases; searches can be done once, or repeated with a single keystroke as many times as required. PeachText searches only in the forward direction, from the current cursor position toward the end of the file.

The replace function is actually part of the search command. First text to be replaced is entered as if a search for it were to be made, followed by the text to replace it. Replacement can be conditional on a case-by-case basis, or global throughout the entire text file.

The ASCII keyboard HOME key (delete right key on the Valdocs keyboard) moves the cursor to the left of the screen on the first stroke, and to the top left corner if it is pressed a second time. The CLS key (Valdocs keyboard LINE key) is used to evoke the thesaurus, which is provided with the Epson version of PeachText.

Thesaurus Operation

The thesaurus, once placed on the editor disk by procedures outlined in the manual, is used by placing the cursor over the first character of the desired word and depressing the CLS key. The thesaurus file is automatically referenced and alternatives are listed on the screen. The arrow keys are used to select the desired replacement. Selected replacement words are automatically inserted into the text in place of the old word.

The thesaurus source file takes 204K bytes of disk space. This, combined with other PeachText files (PeachText itself is 26K bytes long), uses the better part of a QX-10 disk. Most users will probably put their text files on separate disks anyway, and so this is no major concern.

Headers and Footers

PeachText makes provision for automatic page headers and footers throughout text. These can use all the usual variables maintained during printing such as line, page, and column numbers. Commands allow headers and footers of any reasonable size to be centered or placed on the left or right side of a page.

Printing Commands

Printing commands are normally embedded in the text file. (They are preceded by an "escape" character, which is a normal printable character chosen by the user to indicate the start of a command sequence.) The text as seen on the QX-10 is not in the same form as the printed version. (The Valdocs editor is a "what-you-see-is-what-you-get" editor, where the screen is an exact equivalent of what will be printed.)

Commands entered in the text file can stop printing to solicit special text input; display a message on the screen; cause text to be underscored or boldfaced (by printing it twice); and change margins, line spacing, and a host of other variables related to printing text. The same commands can be entered from the keyboard during print time.

PeachText supports a wide variety of printers, including those made by Epson. Text can

When initialized, PeachText presents a well-organized menu that allows a choice of eight functions and notifies the user of the type of printer and terminal it is currently configured to use. (PeachText can drive several types of printers, including daisy-wheel, letter-quality printers such as the Transtar 130 and Epson FX-80 dot matrix printers, which are described in Chapter 7.) The terminal configuration information is a holdover from CP/M, which allows a variety of terminals to be used.

Operational choices from the introductory menu include **EDit** a file, **PRint** a text file (PeachText calls these "document" files), **COpy** a document, **DElete** a document, **REname** a file, **DISplay** a directory, **??** for help, and **ENd** to exit PeachText and return to CP/M.

The directory, file copy, rename, and delete commands are similar in function to those provided by CP/M, but are automatic and easier to use. File names used by PeachText have the same eight-character length limitation as those in CP/M, and are given the extension **.DOC** for **DOCument** to distinguish them from files made by other editing systems. (An example of a document filename might be "CHAPTER5.DOC". The three letters to the right of the period, **DOC**, make up the filename extension.)

Whenever an existing document is re-edited, PeachText automatically changes the original filename extension to **BAK** for backup, providing a single backup copy of every file edited prior to the latest changes. (If there is a previous "BAK" version of the file being edited, it is replaced when the current file is stored at the end of the session.) This quite obviously doubles disk storage requirements, but provides limited protection from catastrophic loss due to machine or operator error.

Editing Commands

The PeachText editor offers a powerful set of commands for editing text and reviewing files. Conditional operators (such as the **IF** statement), variables, and counters control the merging of files during printing, which allows good flexibility when sending letters to a mailing list.

The cursor location is controlled with the QX-10 edit cluster arrow keys. Text can be scrolled forward or backward a line or screen page at a time. The cursor can be sent to the top or bottom of the document using various function keys at the top of the keyboard.

Text deletion in a file is performed by the keyboard backspace key. Individual characters are deleted by the edit-cluster delete key, which removes the character under the cursor, then moves text in from the right of the screen to replace it. Function keys can be set to delete one word or everything on the line that is to the right of the cursor.

Text entry can be done on a replacement basis (anything typed simply replaces the characters under the cursor) or by "insertion," whereby keyboard data is inserted in the character position to the left of the cursor, moving any text under and to the right of the cursor to the right. A "full" insert mode is selectable from a function key to "make a hole" in the current text file for large insertions of text.

PeachText allows blocks of text to be "marked," then moved, copied, deleted or extracted, and placed in another file.

The search function can scan for partial words or complete phrases; searches can be done once, or repeated with a single keystroke as many times as required. PeachText searches only in the forward direction, from the current cursor position toward the end of the file.

The replace function is actually part of the search command. First text to be replaced is entered as if a search for it were to be made, followed by the text to replace it. Replacement can be conditional on a case-by-case basis, or global throughout the entire text file.

The ASCII keyboard **HOME** key (delete right key on the Valdocs keyboard) moves the cursor to the left of the screen on the first stroke, and to the top left corner if it is pressed a second time. The **CLS** key (Valdocs keyboard **LINE** key) is used to evoke the thesaurus, which is provided with the Epson version of PeachText.

Thesaurus Operation

The thesaurus, once placed on the editor disk by procedures outlined in the manual, is used by placing the cursor over the first character of the desired word and depressing the **CLS** key. The thesaurus file is automatically referenced and alternatives are listed on the screen. The arrow keys are used to select the desired replacement. Selected replacement words are automatically inserted into the text in place of the old word.

The thesaurus source file takes 204K bytes of disk space. This, combined with other PeachText files (PeachText itself is 26K bytes long), uses the better part of a QX-10 disk. Most users will probably put their text files on separate disks anyway, and so this is no major concern.

Headers and Footers

PeachText makes provision for automatic page headers and footers throughout text. These can use all the usual variables maintained during printing such as line, page, and column numbers. Commands allow headers and footers of any reasonable size to be centered or placed on the left or right side of a page.

Printing Commands

Printing commands are normally embedded in the text file. (They are preceded by an "escape" character, which is a normal printable character chosen by the user to indicate the start of a command sequence.) The text as seen on the QX-10 is not in the same form as the printed version. (The Valdocs editor is a "what-you-see-is-what-you-get" editor, where the screen is an exact equivalent of what will be printed.)

Commands entered in the text file can stop printing to solicit special text input; display a message on the screen; cause text to be underscored or boldfaced (by printing it twice); and change margins, line spacing, and a host of other variables related to printing text. The same commands can be entered from the keyboard during print time.

PeachText supports a wide variety of printers, including those made by Epson. Text can

be automatically centered, justified (even right margins), proportionally spaced, and printed bidirectionally if the printer supports these features.

Text can also be printed directly from the editor to a file or printer. A print spooler, which is a program that allows editing to continue while a file is printed, is available for additional convenience, but its operation is somewhat complex.

Overall Commentary on PeachText

PeachText is a competent text processing system. Most users will probably find this program entirely adequate for their needs, but automatic footnoting and indexing are not included. The PeachText manual outlines a laborious procedure that works, but is inadequate for all but very small files. (The notes must be manually typed in each time the document is printed.)

The optional thesaurus is fast and easy to use, taking only seconds to extract whatever alternate words are available. However, it uses so much disk space that there isn't much room for anything but PeachText, and so most files created by PeachText will have to be kept on a separate disk. A bug in the thesaurus program causes a carriage return to be placed at the end of any line in which a word has been automatically replaced from the list of alternatives. It has to be removed by the user.

PeachText is easy to learn and use, especially for those with previous text processing experience. It is a comfortable text processor that works very well with the QX-10 keyboard and screen. It is entirely predictable and reliable and has no bad habits that might cause operator difficulties.

PeachTree Spelling Proofreader

After a text processor, probably the next most often used personal computer software package is one of the many spelling checkers on the market. The PeachTree Spelling Proofreader works well in conjunction with files prepared by PeachText and other word processing systems.

The PeachTree Spelling Proofreader available for use with the QX-10 is based on the Random House Dictionary and contains 44,290 words. Versions of this program are available with dictionaries between 21,000 and 85,000 words in size, limited only by the disk storage space. Documents of any size can be checked.

Words can be added or taken out of the standard dictionary, and special dictionaries can be created at any time.

This spelling checker works by scanning the entire document and compiling a list of all the different words used. These are the "unique" words mentioned below. The Proofreader then checks the dictionary for each of the unique words. If the word isn't found in the dictionary, it is put in a list that can be reviewed after all the words have been checked.

Spelling Proofreader is menu driven. An entire document is checked and faulty words are marked by entering little more than the document name to be checked. After that, only single keystrokes are necessary to respond to the spelling checker's results.

When the program is run, the main menu is displayed on the screen. Choices are made by typing a single character: S for spell checking a document; M for dictionary maintenance (to add words, sort them into the main dictionary, merge dictionaries, and more); C for change default tables (more on these later); ? for help; and E to exit Spelling Proofreader.

Running the Spelling Proofreader

A document is "spell-checked" by running the program SP.COM, the main spelling checker program. Option S (Spell-check a document) is selected, and the name of the document file to be examined is entered. A status screen is then displayed, which shows the number of words found in the file, the number and percentage of unique words (those words not in the dictionary that occur one or more times), the number and percentage of spelling "errors," and the percentage of proofing done.

When the spellcheck is completed, a second menu is used to control word review. Options in this menu allow words not found by the checker to be added to the dictionary, marked in the original text file, ignored, or listed on the screen or printer. Help is also available. Although the word list is normally reviewed a word at a time in forward progression, it is also possible to reverse the list to re-examine words already reviewed.

Words are added to the dictionary during word review by simply typing A (for add) when the desired word is examined. Many users put familiar PeachText editing commands in the dictionary so they aren't continually reported as spelling errors.

A misspelled word is marked by selecting the "M" option. This automatic function causes the last character of a misspelled word to be replaced in the source document by a "[" symbol. The actual marking is done after all the words have been reviewed for accuracy. After the spell-checking session is over, the search function in PeachText can be used to locate all the misspelled words for correction. (The marking symbol can be changed by the dictionary management program.)

An alphabetized listing of all the misspelled words is printed on either the screen or printer by the L command.

After all the incorrectly spelled words have been examined and processed, the "C" option is selected to terminate the review and return to the main menu. If any words have been marked for correction, one last chance is given before they are actually marked in the source file. The original, unmarked source file is saved as a backup file (with the file extension .BAK), and the marked version is given the usual .DOC extension, causing it to become the current version.

Dictionary Maintenance

The dictionary maintainer is a collection of programs provided as a part of the Spelling Proofreader. It is menu driven and is used to add or delete words, combine or subtract dictionaries, reorganize a dictionary that has had words added to it, and to list the words in a dictionary.

Special words used by various professions can be collected in a single dictionary and added to or deleted from the main Random House Dictionary by use of the combine and subtract functions.

Words added to a dictionary are generally not merged into the main part of the dictionary until a preset limit of about 250 is reached. This organization is important if the Spelling Proofreader is to operate at top speed. Reorganization is automatically performed when it is required, but can also be initiated by the user.

A listing of words in a dictionary can be printed with the list command. This command is normally used to spot-check a part of a dictionary to determine its content.

Default tables are used to control how a document is checked. They select the dictionary and marking character to be used. They also control whether an automatic backup copy of a document being marked is to be made.

Overall Commentary on Spelling Proofreader

This program is a useful accessory to the PeachText text processor. It is easy and simple to use because of good menu design and the help options. However, several options available on other spell checking programs are missing on Spelling Proofreader. These include a separate "lookup" capability of referring to the dictionary when partial or misspelled words are found and an automatic correction feature, which corrects the source, rather than just marking it.

Special care must be used when adding words to the dictionary. If misspelled, they must be removed and re-entered properly.

Users of this spelling checker (and others) should keep in mind that these checkers are unable to check grammar. If the word "too" is typed, and should have been "two," no error will be reported.

Other writing aids that provide valuable information on style and more in-depth usage checks are now coming onto the market. These can be purchased separately and can be used in conjunction with PeachText and Spelling Proofreader. These programs help check punctuation; identify overused, awkward, and redundant phrases; and provide statistics on word usage. Because of memory size limitations, these programs are often cumbersome, slow, and incomplete, and do not entirely replace manual proofreading.

PeachCalc, an Electronic Spreadsheet

Electronic spreadsheets represent one of the few programs that can be said to have been spawned entirely by the microcomputer market. The first spreadsheet program, VISICALC, has become a legendary example of success, outselling virtually every other program on the market except Microsoft's BASIC.

A spreadsheet program could be considered a large grid consisting of vertical columns and horizontal rows. Each intersection of a column and row is called a "block," into which a

relationship to other blocks or a value is inserted. This relationship is very flexible. In the latest spreadsheets, it can include complex scientific formulas and text.

Spreadsheet operation can be simply illustrated by considering just three blocks, which will be called block A1, B1 and C1. If we say that the number in block C1 is the sum of the numbers held in blocks A1 and B1, and if a five is placed in A1 and a 3 in B1, the computer would automatically put 8 (5 + 3) in block C1. Should block B1 be changed to an 18, the sum in C1 is instantly recomputed and changed to 23. This is a simple example, but serves to explain the basic operation of an electronic spreadsheet.

Applications for electronic spreadsheets abound. Accountants set them up for uses ranging from personal finance to large corporate budgets. Business executives use them for business planning. Spreadsheets work particularly well in "what if . . ." applications. When a single block is changed, the entire spreadsheet is instantly updated, showing the effect everywhere in the grid.

The Spreadsheet

PeachCalc features a grid consisting of 254 rows and 63 columns. The rows are numbered (1 to 254), and the columns are alphabetized (A through Z, then AA through ZZ, and finally BA through BK). This gives a total of 16,002 blocks.

Each block can contain either alphabetic information (such as labels or notes) or numeric data. Inter-block relationships can include addition, subtraction, multiplication, division, partial sums, minimums and maximums, square roots, trigonometric functions, and more.

Operation of the spreadsheet is similar in many ways to the PeachText editor. A large block cursor is positioned around the screen using the edit arrow keys. When the cursor reaches a screen margin the spreadsheet moves horizontally or vertically, as required, to display more of it. A "readout" of the relationship of the block under the cursor is given in the status lines under the grid. This relationship can be changed at any time by simply typing in a new one.

PeachCalc provides all the usual save, load, and copy commands, allowing a spreadsheet to be saved and recalled later. Since the QX-10 screen is not large enough to display the entire spreadsheet at one time, it can also be split vertically or horizontally to show two separate areas of the spreadsheet. The windows can be locked so they scroll together, or one can be held stationary while the other is moved. This feature is useful since changes can be made in one area of the spreadsheet and the results can be observed in another.

Histograms can be displayed on the screen instead of the usual alphanumeric data normally shown in the blocks. These graphs are somewhat crude in comparison with normal graphics, but provide a reasonable visual representation of the numbers involved.

The PeachCalc worksheet can be sent to the QX-10 screen, printer, or disk file for later use. (Output disk files use the special extension .PRN to distinguish them from those generated by the SAVE command. Output files are text files and can be listed or edited as desired.)

General Comments on PeachCalc

This is a good general spreadsheet system. It is reasonably easy to use, but most users will need a few days' practice with it before becoming thoroughly proficient in its use. Unlike the other PeachTree programs discussed in this book, PeachCalc is not menu driven, nor is its operation "intuitive." The manual must be studied. Successful use of this program depends on an in-depth understanding of its operation. Printer commands in particular are poorly documented in the manual.

This spreadsheet program is average in performance; it is neither a blazing ball of fire, nor terribly slow. Many other spreadsheet programs feature a larger grid, but this one is of sufficient size to serve the majority of users.

Perhaps the most useful feature that is lacking is a "PeachCalc Plotter" similar to "VisiPlot" or similar packages. The lack of good graphics capability in this spreadsheet is its most limiting feature.

Mailing List Manager

The Mailing List Manager from PeachTree is menu driven and quite flexible. It can be used to maintain mailing labels and lists (as its name implies), and can also maintain files of lists of parts, bibliographies, travel itineraries, or any other data that can be kept in a list format.

Files of data can have up to 32,765 records (a record being an item, such as a name or street address, for instance). Due to the disk storage limitations, a file with this many records is probably impractical unless a hard disk is available (see Chapter 9, "Expansion Cards and Accessories for the QX-10").

Mailing List Operation

The Mailing List Manager allows the data fields of a label to be defined by the user in any way suitable for his needs. Operation of the system begins by actually defining the address file. This file can have as many as 14 fields for information of your choice. Each field to be used is given a title and the maximum allowable number of characters in it. No actual information is entered here; only the format of the information to be stored is given.

After the address file has been defined, that is, the labels for the various fields have been named along with the allowable lengths, the indexes for this particular file must be selected. These are the fields that are to be used to "look up" items in the file. If a mailing list was being set up, the index might be set to the name field of the file. Three separate indexes can be used if needed, allowing additional sorts by state and city, for example.

The last variable to be chosen is the number of records (entries) that will be placed in the file. This information is used to reserve file space on the disk; the maximum number of records available is determined by the amount of space used by the various fields just defined. If a particularly large list were to be made, it may be more practical to split the file between two disks, treating each separately.

Maintaining Data Files

Data is added to the address titles using the UA (Update Address File) menu option. Operation is simple because the name of each field defined for the file is neatly displayed on the screen. Information to be kept in each field is typed onto the screen. When every field is entered, the entire record is stored in the file. This process is repeated until the session is complete.

Entries in any record can be changed or deleted by calling up the record, "tabbing" to the field to be changed, deleting the field, and typing in the new value. Any record on file can be examined by simply typing the field entry for that particular record, provided that it was defined as an "indexed" field. In the mailing list example, a record for a particular person could be displayed by typing the name, the city, or the state.

It is not necessary to enter the complete name to recall a record for editing. It is only necessary to enter the minimum letters necessary to make it unique from the other names maintained on file. Mailing List Manager uses only the first 23 characters of an indexed field, although the file can contain as many characters as defined.

Mailing Labels

Mailing labels of nearly any size or style can be printed by Mailing List Manager. A comprehensive, yet easy to use, label format editor allows any field to be printed anywhere on the label, and any label format to be used. (Typical standard labels have two labels per row and ten rows per page. This standard can be changed by entering the arrangement of whatever label stock is to be used.)

The actual data to be printed on the labels is arranged with a screen editor that shows the available fields from the file being used as the source of the label data. An outline of the label to be printed is shown on the screen. The cursor is positioned to the location of each selected field to be printed; a control-P (depressing the control key and the P key simultaneously) places the field at the cursor, and a control-E erases it for repositioning. Once the label format is established, it can be saved for later use.

Labels are printed under the supervision of a program called PM (for Produce Mailing). A comprehensive screen menu defines the type of stock, the format, the sort being used (perhaps ZIP code and state would be most common for labels), and the output device. Output can be sent directly to a printer, the CRT console of the QX-10, or a disk file for later printing.

An important feature included by Mailing List Manager is that of a "mailing profile." This is actually a mask that must be matched if a record is to be included or excluded in the printing. For example, a special mailing for a particular town might be desired from a mail list that included several states.

Profiles can be set up in a variety of ways. It is possible to set up profiles to make use of dates, keywords in descriptive fields, and numbers from the fields. For example, a mailing could be generated that included all women between the ages of 24 and 36 who live in Colorado, except those in Denver.

Comments on Mailing List Manager

This is a very useful and adaptable program for use with mailing lists or other information adaptable to this manipulation. Mailing List Manager can sort any of three index fields and produce accurate lists with complex requirements of what is or isn't to be printed from the source file.

The limitations of this program have little to do with its primary objective, which is to manage and produce mailing lists. It is not particularly convenient to manually change or add data to the address files, nor is it possible to "drive" this program with a disk file generated by another program, which limits its use for rapidly changing data such as active inventories.

However, data files maintained by this program can be used by PeachText to generate "personalized" form letter mailings. Complete information on performing this function is given in the Mailing List Manager and PeachText manuals.

Operation of this program is controlled by well-designed screen menus and is easily learned.

CP+ or Making CP/M More Friendly

This program could be considered both a utility program and an applications program. Distributed by Epson as a part of the CP/M documentation, CP+ is intended to make CP/M into a menu driven operating system. CP+ comes already installed on the CP/M master disk and immediately takes control of the CP/M operating system.

Briefly, CP+ takes the place of the CCP (Command Console Program) that normally resides at the bottom of CP/M. It replaces the CCP with a collection of screen menus. These menus control everything from PIP, which is used to copy files from one disk to another, to program selection and the running of ordinary applications programs.

CP+ comes in a special ring binder that includes a tutorial that provides practice while stimulating disk operations so there is no possibility of losing data. The manual explains in detail all 18 screen menus and how they work, although many users will immediately grasp the concepts involved and choose to learn CP+ directly from use.

CP+ Organization

When the QX-10 is initialized with CP+, a start-up screen is displayed. This screen allows a selection of seven options:

1. Help—Displays a brief description of the following choices.
2. Select and Run a program—Displays the current CP/M user number directory (in several pages if necessary). Programs can be selected and run with a bare minimum of single key strokes.
3. Print Commands—If a fully formatted file is ready for printing, CP+ provides a print spooler mechanism to keep your printer busy while you use the computer for other tasks at the same time.

4. File Commands—Files can be copied, renamed, and erased. The operating system can be copied, and disks can be formatted. This program automates parts of the PIP, SYSTEM, and FORMAT programs described in Chapter 4.

5. Review Commands—Allows you to look at file contents, the disk directory, and the file description catalog (CP+ allows short file descriptions to be saved along with the actual files), the print queue (useful if you have scheduled several files to be printed by the spooler), and the formats in which they are to be printed. (Only the top, bottom, and side margins can be changed.)

6. The X, or Exit Option—Allows CP+ to be abandoned and replaced by standard CP/M operation.

7. The Final Option—Allows the CP/M user area to be changed.

General Comments about CP+

Those who are new to computers will find CP+ a genuine aid. No knowledge of any of the various processes involved with PIP, FORMAT, or SYSTEM is required, for instance, when copying disks or files. CP+ also provides much the same services as does the Epson utility program COPYDISK, with which disks can be formatted and copied. It is also very convenient to be able to run a file by simply looking it up in menu and making a single keystroke selection. The print spooler can also be a benefit if files to be printed are already formatted and ready for printing. Another handy feature is that a catalog is maintained by CP+ that allows short descriptions of files to be kept. These are automatically displayed with the directory display.

Those are the good points of CP+. These must be balanced against the relative disadvantages of speed (CP+ adds at least another 20 seconds or so to the time it takes to exit a program and return to the system-prompt menu, from which another program can be selected and run). Another potential source of annoyance is that the directory is displayed with only six files on each page. Disks with large directories will use many pages. These must be searched, page by page, when a particular program or data file is needed. Fortunately, pages can be rapidly changed, and so the process is fairly rapid.

The jury is still out on CP+, but most users with a little experience will probably choose to disable it by use of the SETUP program after the newness has worn off and the obvious slow-down becomes objectionable.

Applications Programs Integration

One of the clear-cut advantages of using applications programs from a single vendor such as PeachTree (there are many others) is that there are usually common traits shared by all programs from the same "family." Knowledge and experience on one package can, even if in a small way, be used on other programs. In other words, experience is *transferable* from one application to another.

CP+ largely eliminates the need to learn relatively complex procedures associated with PIP, FORMAT, and SYSTEM programs. A selection is made from a menu, such as copy

all files, and the computer does the rest, prompting for proper disk insertion and the like. Forget for the moment the extra time CP+ takes to initialize; any program can be speeded up and improved.

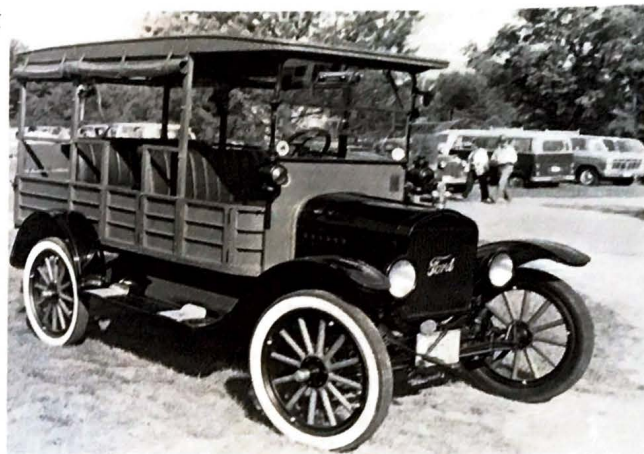
The point is that the applications program just described attempt to make the computer both consistent and simple to use through a collection of menus and programs that automate most of the internal functions.

Valdocs, the subject of Chapter 6, represents a revolutionary change in the way that computers are programmed and used. The whole concept is woven around the complete integration of the human interface, computer functionality (software), and hardware, all the while keeping in mind that to be successful, the human element is the master and the computer is the slave. It has taken the idea of system integration (simplification and automation) several steps beyond what can be done with "off the shelf" software today.

Another major conceptual difference between Valdocs and most other applications programs systems is that working knowledge developed in one area of use is immediately and completely applicable to others throughout the system. Great effort is made to avoid forcing a user to learn anything about what goes on inside the computer. The computer, in the Valdocs philosophy, is simply another tool or appliance. Most television viewers have no need to know anything about the internal design of a television set, so why should a computer user need to know anything about the insides of what is basically another household appliance?

Valdocs, the subject of Chapter 6, is the product of many years of study of the human-computer relationship, which promises to radically simplify the operation and use of personal computers to the point that they may someday actually become the "mind amplifiers" of science fiction promise.

6



Valdocs

The Valdocs operating system available for the QX-10 is unique at the present time. "Valdocs" is a combination of the words "VALuable DOCuments." It is both an operating system and a set of applications programs which, in version 3, will provide most of the software needed by the majority of all computer users. The interface standard used by Valdocs is the HASCI (Human Applications Standard Computer Interface) and the Valdocs keyboard, which, appropriately enough, is called "the HASCI keyboard."

The HASCI concept is derived from a series of studies performed by Chris Rutkowski during the late 1970s to early 1980s. During his relationship with TDL (see Chapter 1), he noticed that consumer products requiring mass production went through a common evolutionary development.

He found that a newly discovered product would usually be built at first more or less for its own sake by pioneering technologists. In the case of the automobile, which is often cited, early cars were hand built more or less for the thrill of it. Far more tinkering and repair was done than actual travel.

Later, other technically inclined but less ambitious individuals prevailed and limited hand-production of the early cars began. Cars of this period were by and large unreliable,

A restored 1925 Yellow Knight.

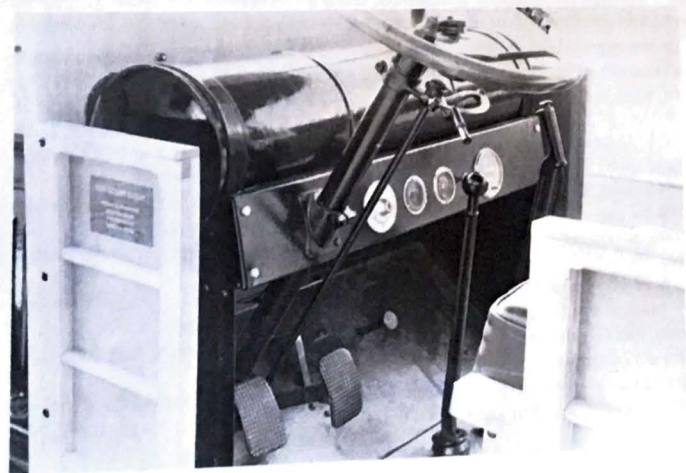
expensive, and not terribly useful, but exciting none the less. Word of mouth and popular press coverage soon brought notice to the public, and the demand for cars increased, despite the rigors of actually owning and operating such an advanced product.

Early car owners were inconvenienced, to say the least. Every car was fundamentally different, except for the wheels—they were all round. However, some cars had three wheels, and others had four, five, or six wheels. Operating controls were just as varied. Reins, tiller bars and steering wheels were used in every combination with hand brakes and foot brakes, floor and hand throttles, and steam, electric, and gasoline engines. Every car was an adventure to drive (or endure, depending on how devoted you were). The increased demand created more competition and lower prices.

Once the price of the automobile began to fall, increased public awareness created a demand by people who wanted an automobile that worked, was reliable, and didn't need a master mechanic on hand every time it was used. Industry was forced to oblige or remain forever small.

Automotive development was then taken over by more professional, well-financed companies. A "standard" configuration for automobiles was developed. This standardization has remained in effect with little change for more than 50 years. This phenomenon was dubbed "architectural stabilization" by Rutkowski.

Although the driver's controls are not entirely the same as on today's cars, the Yellow Knight could be driven by anyone used to a stick shift. This vehicle was manufactured about the same time the standard automotive architecture was emerging.



We could consider the architectural essence of an automobile to be a body of sufficient size to hold two or more people, transported by four pneumatically inflated tires, gear driven by a petroleum burning reciprocating engine of some sort, and steered by the front wheels with a "steering wheel." The driver's seat is in the front of the vehicle on either the left or right side, depending on what country it is to be used in. A collection of foot-operated controls are placed in front of the driver to control the engine speed and braking. Optionally, a third foot-control operates a clutch for cars equipped with manual transmissions. A gear or transmission selector is conventionally placed on the floor of the vehicle inboard from the driver or on the steering column. Vehicle instrumentation is placed more or less directly in front of the driver, below the lower edge of the windscreen.

Once this architecture was reached and standardized (in the mid-1920s), automotive development began to concentrate on improvements in performance, service, reliability, and ease of use. Drivers of cars built during and after this period are able to switch from one to another without having to learn how to drive all over again. Knowledge and driving experience gained on one vehicle are directly applicable to the operation of other vehicles of the same class.

This same evolutionary trend, which in the case of the automobile took more than 30 years, is now taking place in today's microcomputer industry. The personal computer architecture "stabilized" in only eight years or so. Large, well-financed companies such as Tandy, IBM, and Epson saw a mass market for personal, small business, and scientific computers and are now concentrating development on software to make them easier to use. Meanwhile, what hardware development there is will be focused on increasing the speed and capacity of the computing machine itself; computers will get smaller, faster, and more powerful, but will remain recognizable for the foreseeable future.

THE HASCI CONCEPT

HASCI is a concept, technique, and philosophy by which a human being is interfaced with an infinitely malleable tool. It recognizes the strengths and weaknesses of both humans and computers of the state of the art. When successfully implemented, the interface becomes transparent to the human, who, it is hoped, will not even notice its presence.

The obvious and recognized strengths of a computer include the performance of high-speed repetitive tasks, numerical calculation, sorting, merging, transforming, sifting, and searching information or symbols of all kinds. The computer, however, is disadvantaged when it must determine the significance of the results of any of the above tasks.

Humans happen to be very good at tasks at which today's computers are weakest—reasoning. A shorthand way of describing the human-computer relationship might be to use Rutkowski's description of a computer as a "symbol manipulator" and of the human as a "pattern recognizer."

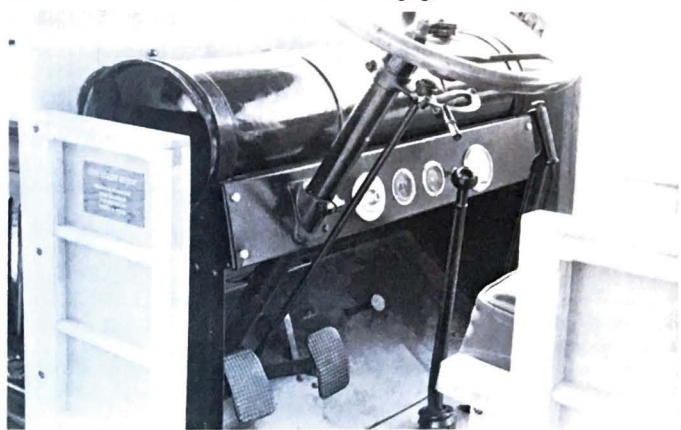
expensive, and not terribly useful, but exciting none the less. Word of mouth and popular press coverage soon brought notice to the public, and the demand for cars increased, despite the rigors of actually owning and operating such an advanced product.

Early car owners were inconvenienced, to say the least. Every car was fundamentally different, except for the wheels—they were all round. However, some cars had three wheels, and others had four, five, or six wheels. Operating controls were just as varied. Reins, tiller bars and steering wheels were used in every combination with hand brakes and foot brakes, floor and hand throttles, and steam, electric, and gasoline engines. Every car was an adventure to drive (or endure, depending on how devoted you were). The increased demand created more competition and lower prices.

Once the price of the automobile began to fall, increased public awareness created a demand by people who wanted an automobile that worked, was reliable, and didn't need a master mechanic on hand every time it was used. Industry was forced to oblige or remain forever small.

Automotive development was then taken over by more professional, well-financed companies. A "standard" configuration for automobiles was developed. This standardization has remained in effect with little change for more than 50 years. This phenomenon was dubbed "architectural stabilization" by Rutkowski.

Although the driver's controls are not entirely the same as on today's cars, the Yellow Knight could be driven by anyone used to a stick shift. This vehicle was manufactured about the same time the standard automotive architecture was emerging.



We could consider the architectural essence of an automobile to be a body of sufficient size to hold two or more people, transported by four pneumatically inflated tires, gear driven by a petroleum burning reciprocating engine of some sort, and steered by the front wheels with a "steering wheel." The driver's seat is in the front of the vehicle on either the left or right side, depending on what country it is to be used in. A collection of foot-operated controls are placed in front of the driver to control the engine speed and braking. Optionally, a third foot-control operates a clutch for cars equipped with manual transmissions. A gear or transmission selector is conventionally placed on the floor of the vehicle inboard from the driver or on the steering column. Vehicle instrumentation is placed more or less directly in front of the driver, below the lower edge of the windshield.

Once this architecture was reached and standardized (in the mid-1920s), automotive development began to concentrate on improvements in performance, service, reliability, and ease of use. Drivers of cars built during and after this period are able to switch from one to another without having to learn how to drive all over again. Knowledge and driving experience gained on one vehicle are directly applicable to the operation of other vehicles of the same class.

This same evolutionary trend, which in the case of the automobile took more than 30 years, is now taking place in today's microcomputer industry. The personal computer architecture "stabilized" in only eight years or so. Large, well-financed companies such as Tandy, IBM, and Epson saw a mass market for personal, small business, and scientific computers and are now concentrating development on software to make them easier to use. Meanwhile, what hardware development there is will be focused on increasing the speed and capacity of the computing machine itself; computers will get smaller, faster, and more powerful, but will remain recognizable for the foreseeable future.

THE HASCI CONCEPT

HASCI is a concept, technique, and philosophy by which a human being is interfaced with an infinitely malleable tool. It recognizes the strengths and weaknesses of both humans and computers of the state of the art. When successfully implemented, the interface becomes transparent to the human, who, it is hoped, will not even notice its presence.

The obvious and recognized strengths of a computer include the performance of high-speed repetitious tasks, numerical calculation, sorting, merging, transforming, sifting, and searching information or symbols of all kinds. The computer, however, is disadvantaged when it must determine the significance of the results of any of the above tasks.

Humans happen to be very good at tasks at which today's computers are weakest—reasoning. A shorthand way of describing the human-computer relationship might be to use Rutkowski's description of a computer as a "symbol manipulator" and of the human as a "pattern recognizer."

Text processing (the composition and rearrangement of textual information) is a good example of how this relationship works. A human conceives a literary concept, taken as a whole as a pattern of symbols and communicates it to a computer. Under human direction, the computer blindly manipulates the information without any conceptual understanding of the material until the human recognizes a satisfactory pattern. After editing, the text is ordered printed or stored for later use.

The human interest in this example is limited to the text, not in the details of how the information is processed, stored, or printed. The matters of substance in this case are really the following:

1. How the information can be placed in the computer with the least effort.
2. Once there, how it can be altered and processed according to personal whim with the least possible effort.
3. How it can be put away or saved for future use with the least effort.
4. How it can be recalled and/or printed with the least effort.

None of the above issues are related to any specific computer. Indeed, the human doesn't care if the device actually used is called a computer or a bulldozer. The only interest is in using a tool, in this case a text processor, not in learning to service it or the computer in any way. Furthermore, when it is time to buy another computer, the human does not want to have to learn how to use it all over again. Once the interface to a computer is learned, the experience and operational skills already acquired should be directly transferable to any computer.

The fundamental HASCI concept, then, is to make the computer automatic and easy to use, presenting no logical or artificial encumbrances to the human operator. At the same time, it provides computational service in a consistent, unobtrusive, and rational manner so that an average person looking at the computer controls can readily put the computer to use. On a broader scale, the concept will make it possible to transfer operating experience from one computer to another. This is presently not possible because of the plethora of operating systems and non-standardized keyboards, displays, and software, most of which cannot be operated "intuitively". Considerable technical study of the internal workings of the individual computer system is required before competent use of most personal computers can be made.

Outwardly, HASCI dictates that computer controls are to be labeled in such a way as to give an intuitive functional understanding of their use. Controls are organized into logical blocks of related functions. All are clearly labeled in function and produce immediately observable effects. Humans are not required to know or understand anything about the internal workings of the machine or software itself. If help is required, it is instantly accessible on the screen; mistakes are "undo-able." The computer automatically "backs everything up" so that it is nearly impossible to accidentally lose data. If a power failure occurs, the computer "remembers" where it was and automatically restores the screen, all files, and the cursor to their original states when power returns. This is part of what the HASCI interface is all about.

VALDOCS VERSIONS

VALDOCS 85

The Epson QX-10 is the first computer to be built to support the HASCI concept, which is embodied in the Valdocs operating system. The first public Valdocs release (version 1.14) was shipped to the United States during May of 1983. Version 1.14 was replaced three months later with version 1.16, which was superseded in the fall of 1983 by 1.18. Only minor functional changes were made between these releases.

Version 1.16 was developed to increase the editor display speed. It added underscore and multiple line-feed capability, which was lacking in version 1.14. (It also corrected several bugs, which are inevitable in any major software release, including one that was colorfully named the "Phantom Stomper.")

Version 1.18 significantly increased display speed and added a selectable non-bit-mapped display mode for even more speed. (The normal Valdocs editor CRT display is "bit-mapped," meaning that the screen is used in the graphics, or bit-mapped mode. Characters are actually "plotted" rather than displayed as characters. Bit-mapping allows special characters and character attributes such as boldface and underscore to be displayed on the screen, but at a discernible reduction in display speed.)

Version 1.18 also freed the cursor so it could be positioned vertically anywhere on the screen. It allowed the indexer to be turned off for greater file handling speed during store and retrieve operations. (These features are described and explained later.)

Overall, three models of Valdocs were developed, each with increasing speed and capability.

Version 1 (1.14, 1.16, and 1.18, already released) features a bit-mapped screen with an editor suitable for many general text processing needs, including boldface, italic, and underscored characters; a print spooler so printer operations do not interfere with other computing needs; electronic mail; an internal on-screen calculator; and a business systems graphics package. A calendar-scheduler completes the major software offerings.

Version 2, to be introduced in the first half of 1984, includes all the software packages found in version 1, the Mail address book, which can be merged with documents for "personalized" form letters, automatic computer network log-on, improved business graphics with overlays, advanced scientific bar-graph capabilities, an electronic spreadsheet, user definable functions, and more. The editor includes major speed improvements, super- and subscripting, automatic indexing and footnoting, condensed and enlarged characters, proportional spacing on the screen, and more.

Version 3, which will be introduced in 1985, takes Valdocs near its evolutionary limit. Here Valdocs virtually becomes a programming language (to be called FPL—Forms Processing Language). The editor, data base manager, spreadsheet, and indexer are tightly interwoven with the business graphics system. Complete personalized business filing systems can easily be developed by average users using version 3's FPL.

Valdocs applications programs, such as a three-dimensional CAD/CAM graphics package, spelling checkers, and data base managers will start appearing between versions 2 and 3.

Text processing (the composition and rearrangement of textual information) is a good example of how this relationship works. A human conceives a literary concept, taken as a whole as a pattern of symbols and communicates it to a computer. Under human direction, the computer blindly manipulates the information without any conceptual understanding of the material until the human recognizes a satisfactory pattern. After editing, the text is ordered printed or stored for later use.

The human interest in this example is limited to the text, not in the details of how the information is processed, stored, or printed. The matters of substance in this case are really the following:

1. How the information can be placed in the computer with the least effort.
2. Once there, how it can be altered and processed according to personal whim with the least possible effort.
3. How it can be put away or saved for future use with the least effort.
4. How it can be recalled and/or printed with the least effort.

None of the above issues are related to any specific computer. Indeed, the human doesn't care if the device actually used is called a computer or a bulldozer. The only interest is in using a tool, in this case a text processor, not in learning to service it or the computer in any way. Furthermore, when it is time to buy another computer, the human does not want to have to learn how to use it all over again. Once the interface to a computer is learned, the experience and operational skills already acquired should be directly transferable to any computer.

The fundamental HASCI concept, then, is to make the computer automatic and easy to use, presenting no logical or artificial encumbrances to the human operator. At the same time, it provides computational service in a consistent, unobtrusive, and rational manner so that an average person looking at the computer controls can readily put the computer to use. On a broader scale, the concept will make it possible to *transfer* operating experience from one computer to another. This is presently not possible because of the plethora of operating systems and non-standardized keyboards, displays, and software, most of which cannot be operated "intuitively". Considerable technical study of the internal workings of the individual computer system is required before competent use of most personal computers can be made.

Outwardly, HASCI dictates that computer controls are to be labeled in such a way as to give an intuitive functional understanding of their use. Controls are organized into logical blocks of related functions. All are clearly labeled in function and produce immediately observable effects. Humans are not required to know or understand anything about the internal workings of the machine or software itself. If help is required, it is instantly accessible on the screen; mistakes are "undo-able." The computer automatically "backs everything up" so that it is nearly impossible to accidentally lose data. If a power failure occurs, the computer "remembers" where it was and automatically restores the screen, all files, and the cursor to their original states when power returns. This is part of what the HASCI interface is all about.

VALDOCS VERSIONS

The Epson QX-10 is the first computer to be built to support the HASCI concept, which is embodied in the Valdocs operating system. The first public Valdocs release (version 1.14) was shipped to the United States during May of 1983. Version 1.14 was replaced three months later with version 1.16, which was superseded in the fall of 1983 by 1.18. Only minor functional changes were made between these releases.

Version 1.16 was developed to increase the editor display speed. It added underscore and multiple line-feed capability, which was lacking in version 1.14. (It also corrected several bugs, which are inevitable in any major software release, including one that was colorfully named the "Phantom Stomper.")

Version 1.18 significantly increased display speed and added a selectable non-bit-mapped display mode for even more speed. (The normal Valdocs editor CRT display is "bit-mapped," meaning that the screen is used in the graphics, or bit-mapped mode. Characters are actually "plotted" rather than displayed as characters. Bit-mapping allows special characters and character attributes such as boldface and underscore to be displayed on the screen, but at a discernible reduction in display speed.)

Version 1.18 also freed the cursor so it could be positioned vertically anywhere on the screen. It allowed the indexer to be turned off for greater file handling speed during store and retrieve operations. (These features are described and explained later.)

Overall, three models of Valdocs were developed, each with increasing speed and capability.

Version 1 (1.14, 1.16, and 1.18, already released) features a bit-mapped screen with an editor suitable for many general text processing needs, including boldface, italic, and underscored characters; a print spooler so printer operations do not interfere with other computing needs; electronic mail; an internal on-screen calculator; and a business systems graphics package. A calendar-scheduler completes the major software offerings.

Version 2, to be introduced in the first half of 1984, includes all the software packages found in version 1, the Mail address book, which can be merged with documents for "personalized" form letters, automatic computer network log-on, improved business graphics with overlays, advanced scientific bar-graph capabilities, an electronic spreadsheet, user definable functions, and more. The editor includes major speed improvements, super- and subscripting, automatic indexing and footnoting, condensed and enlarged characters, proportional spacing on the screen, and more.

Version 3, which will be introduced in 1985, takes Valdocs near its evolutionary limit. Here Valdocs virtually becomes a programming language (to be called FPL—Forms Processing Language). The editor, data base manager, spreadsheet, and indexer are tightly interwoven with the business graphics system. Complete personalized business filing systems can easily be developed by average users using version 3's FPL.

Valdocs applications programs, such as a three-dimensional CAD/CAM graphics package, spelling checkers, and data base managers will start appearing between versions 2 and 3.

UPGRADING VALDOCS

The present Epson policy on upgrading Valdocs is that, within a given version, updates are free; simply take the original disk in to any Epson dealer and have it copied. There will be a nominal charge for moving to higher versions, but, at the time of this writing, the charge had not been determined. Users should question any charges for updates within a given version because this is not present Epson policy.

What follows is an introduction to Valdocs operation. It is meant to offer only general notes and comments about Valdocs version 1 since versions 2 and 3 were not released at the time of this writing. Exact procedural information can be found in the Valdocs operating manual and through the HELP key on the HASCI keyboard.

VALDOCS OPERATION

Valdocs is brought into operation by inserting the Valdocs program disk into drive A and a data disk into drive B. (A data disk is a standard formatted disk. It can contain CP/M files or remain blank, but must be formatted.) Valdocs takes a considerable time to run the QX-10 system diagnostic checks and load, usually about 60 seconds.

The first menu encountered is the "hello menu," which indicates the Valdocs version, Rising Star copyright notice, and time of day. This menu stays on the screen until any key is typed, at which time the Valdocs editor is entered.

The editor could be considered the normal "home" position for Valdocs. As soon as it is active, a document can be started by simply typing it on the keyboard. Experienced computer users will find this incredibly "unnatural" because no operating system functions had to be used. Neither the editor, object file names, nor anything else is called up; everything happens automatically.

Valdocs allows any computer function to be used at any time. It is not necessary to store a document being prepared before, for example, checking the electronic mail in-basket. All files are automatically saved and protected as necessary. After the application or other task has been completed, Valdocs automatically returns to the editor with the original document and cursor in the same place they were when interrupted. Experienced users will find this somewhat uncanny.

Valdocs will be explored here by first examining the editor and how it operates, followed by a look at the various functions available through the control clusters spread across the top of the HASCI keyboard. As will be seen, Valdocs and the HASCI keyboard work together very well, and the relationship between them and the user is cordial and harmonious.

THE VALDOCS EDITOR

As just mentioned, the "normal" or default state of Valdocs is the edit mode. It can be left at any time to perform other tasks, which return control to the editor when completed.

The editor is used to compose and edit text documents. It is of the "what-you-see-is-what-you-get" variety, meaning that whatever shows on the screen will be copied onto paper when the print command is given. This, of course, depends on the type of printer used with the QX-10. At present, only the Epson FX-80 and FX-100 printers will print text *exactly* as it is seen on the QX-10 screen, although other dot matrix printers and various letter-quality printers that support the Diablo 1620 code conventions can be used as well.

If non-FX-80/100 printers are used, none of the "fancy" font features or graphics are printable. These include condensed, expanded, and block graphics characters (available in Valdocs versions 2 and 3); italic or other character sets; proportional spacing; graphic output from the business graphics package, which is available through the DRAW key; and output from the three-dimensional CAD/CAM applications package.

Later versions of Valdocs will make some effort to support other printers by providing a facility in SETUP (a program described later) to allow limited code translation, but if the printer does not fully support graphics using standard Epson coding, or have a downloadable font capability matching the Epson protocol, certain Valdocs printing features will be lost.

Editor Screen Display

The editor screen presents a page boundary marker (a horizontal line on the screen where the top or bottom of the printed page will appear), a status line at the bottom of the screen, and a vertical column of dots along the right margin. Each dot represents an empty line.

If special features such as underscored or boldface characters have been used in a line, the dot is replaced by an asterisk (*); a line ended with a carriage return, which is normally used only to end a paragraph or to provide vertical space between text, is indicated with a down arrow (↓). A "normal" line with only standard characters in it has no dot.

Although the margin usually expected by the editor is 80 columns, lines up to several hundred characters in width can be typed if desired. If text runs past the 80-column mark, the dot is replaced with a "+" to indicate the extended line. When the cursor (a full character-sized nonblinking block) moves past the 80th column, all text on the screen is automatically shifted to the left to expose the extended line. Everything is shifted back to the normal position when the cursor is returned anywhere within the normal margin boundaries.

The status area at the bottom of the screen shows the current left and right margins, and any tabs set along a line, which are marked in ten-character increments. The time of day and number of files currently in the print spooler, current line and character position, and current line spacing are listed under this line.

Text files created by the Valdocs editor can be of any reasonable size. Files greater than about 3K bytes in length are automatically scrolled in and out of memory into temporary disk files as necessary to accomplish the editing functions requested by the user. The maximum practical file length is roughly half the storage of whatever disk system may be in use.

UPGRADING VALDOCS

The present Epson policy on upgrading Valdocs is that, within a given version, updates are free; simply take the original disk in to any Epson dealer and have it copied. There will be a nominal charge for moving to higher versions, but, at the time of this writing, the charge had not been determined. Users should question any charges for updates within a given version because this is not present Epson policy.

What follows is an introduction to Valdocs operation. It is meant to offer only general notes and comments about Valdocs version 1 since versions 2 and 3 were not released at the time of this writing. Exact procedural information can be found in the Valdocs operating manual and through the HELP key on the HASCI keyboard.

VALDOCS OPERATION

Valdocs is brought into operation by inserting the Valdocs program disk into drive A and a data disk into drive B. (A data disk is a standard formatted disk. It can contain CP/M files or remain blank, but must be formatted.) Valdocs takes a considerable time to run the QX-10 system diagnostic checks and load, usually about 60 seconds.

The first menu encountered is the "hello menu," which indicates the Valdocs version, Rising Star copyright notice, and time of day. This menu stays on the screen until any key is typed, at which time the Valdocs editor is entered.

The editor could be considered the normal "home" position for Valdocs. As soon as it is active, a document can be started by simply typing it on the keyboard. Experienced computer users will find this incredibly "unnatural" because no operating system functions had to be used. Neither the editor, object file names, nor anything else is called up; everything happens automatically.

Valdocs allows any computer function to be used at any time. It is not necessary to store a document being prepared before, for example, checking the electronic mail in-basket. All files are automatically saved and protected as necessary. After the application or other task has been completed, Valdocs automatically returns to the editor with the original document and cursor in the same place they were when interrupted. Experienced users will find this somewhat uncanny.

Valdocs will be explored here by first examining the editor and how it operates, followed by a look at the various functions available through the control clusters spread across the top of the HASCI keyboard. As will be seen, Valdocs and the HASCI keyboard work together very well, and the relationship between them and the user is cordial and harmonious.

THE VALDOCS EDITOR

As just mentioned, the "normal" or default state of Valdocs is the edit mode. It can be left at any time to perform other tasks, which return control to the editor when completed.

The editor is used to compose and edit text documents. It is of the "what-you-see-is-what-you-get" variety, meaning that whatever shows on the screen will be copied onto paper when the print command is given. This, of course, depends on the type of printer used with the QX-10. At present, only the Epson FX-80 and FX-100 printers will print text *exactly* as it is seen on the QX-10 screen, although other dot matrix printers and various letter-quality printers that support the Diablo 1620 code conventions can be used as well.

If non-FX-80/100 printers are used, none of the "fancy" font features or graphics are printable. These include condensed, expanded, and block graphics characters (available in Valdocs versions 2 and 3); italic or other character sets; proportional spacing; graphic output from the business graphics package, which is available through the DRAW key; and output from the three-dimensional CAD/CAM applications package.

Later versions of Valdocs will make some effort to support other printers by providing a facility in SETUP (a program described later) to allow limited code translation, but if the printer does not fully support graphics using standard Epson coding, or have a downloadable font capability matching the Epson protocol, certain Valdocs printing features will be lost.

Editor Screen Display

The editor screen presents a page boundary marker (a horizontal line on the screen where the top or bottom of the printed page will appear), a status line at the bottom of the screen, and a vertical column of dots along the right margin. Each dot represents an empty line.

If special features such as underscored or boldface characters have been used in a line, the dot is replaced by an asterisk (*); a line ended with a carriage return, which is normally used only to end a paragraph or to provide vertical space between text, is indicated with a down arrow (↓). A "normal" line with only standard characters in it has no dot.

Although the margin usually expected by the editor is 80 columns, lines up to several hundred characters in width can be typed if desired. If text runs past the 80-column mark, the dot is replaced with a "+" to indicate the extended line. When the cursor (a full character-sized nonblinking block) moves past the 80th column, all text on the screen is automatically shifted to the left to expose the extended line. Everything is shifted back to the normal position when the cursor is returned anywhere within the normal margin boundaries.

The status area at the bottom of the screen shows the current left and right margins, and any tabs set along a line, which are marked in ten-character increments. The time of day and number of files currently in the print spooler, current line and character position, and current line spacing are listed under this line.

Text files created by the Valdocs editor can be of any reasonable size. Files greater than about 3K bytes in length are automatically scrolled in and out of memory into temporary disk files as necessary to accomplish the editing functions requested by the user. The maximum practical file length is roughly half the storage of whatever disk system may be in use.

After a document is composed and edited, it is named and stored in a disk file through the STORE key, described later. Files are normally printed directly from disk files with the print spooler, which is controlled by the PRINT key, but a screen dump, which includes the status line and everything else displayed on the screen, is always available by typing "control-PRINT" (holding the control key down and pressing the PRINT key). Screen dumps are not allowed on non-Epson printers because of non-compatible graphics and downloadable character fonts.

Editor Operation

Operation of the editor is simple. Just start typing as if using a typewriter. Words that "hang" over the end of a line are automatically moved to the beginning of the following line. (This is called "word-wrapping.") The return key is used only to end paragraphs.

The TYPESTYLES function keys are used to select BOLDFACE or ITALIC fonts, the only type faces available in version 1. (Both the normal and italic fonts can be boldfaced.) The spacing between lines is controlled by the SIZE key. (It advances from 1 to 2 to 3 and back to 1 with each stroke of the key.) The style key causes text to be underscored.

The keyboard can be shifted into upper case like a typewriter by pressing the SHIFT LOCK key, at which time a red indicator in the key is illuminated. Normal keyboard operation is returned by typing either SHIFT key. The keyboard can also be locked into the "all caps" mode, in which all alphabetic characters are printed in upper case, but numbers and punctuation remain normal. (Holding the control key down while typing the STYLE key.) Whenever a control character is to be typed, first (or simultaneously) depress either control key on the keyboard, then the key.

The Valdocs version 1 cursor cannot be positioned anywhere on the screen that does not already have a printable character on it. As a result, it is necessary to use the space bar or other keys to move the cursor across the screen on a given line for the first time. Tabs, once set in the standard typewriter fashion, can be used to rapidly move the cursor across "blank" areas of the screen. This limitation is removed in versions 2 and 3.

Cursor control on the Valdocs editor is very flexible. The four arrow keys move the cursor in the indicated direction. The vertical arrows do not have auto-repeat, as do most of the other keys, but will repeat if a control key is held down with them. Since the screen scrolling cannot keep up with the repeat mode, it is likely the text will scroll past the intended line. Such scrolling can be instantly stopped using the STOP key.

The cursor can be directed in the left and right directions by two differing sets of editing keys. The two arrow keys simply move it, character by character, in the indicated direction. The two other arrow keys, those with an X marked inside, move the cursor as before, but delete characters located under the cursor as it moves.

The cursor can also be moved a word at a time or to the end of a line. For example, to move it to the right end of a line, first type the right arrow key, then the edit key marked "LINE." If the delete key with the right arrow had been pressed instead, the entire line of

characters to the right of the cursor would have been erased. The same operation can be performed in the left direction by using the left arrow keys instead.

Text can be scrolled vertically either downward (toward the end of the document) or upward (toward the beginning) by holding down the shift key while typing the down or up arrow keys, respectively. Text is moved 22 lines with each key stroke, about a page at a time.

Numeric text can be entered with the calculator keypad, which causes all entries to line up automatically on either the right digit or the decimal point. This is done by first setting a tab at the desired location of the decimal point or digit, then using the DECIMAL TAB key on the numeric keypad to tab to position. Numbers then entered on either the keypad or main typing area will be properly aligned.

Be sure to use the DECIMAL TAB key for this operation; the normal TAB key works like that of a regular typewriter and does not influence numeric entry. (Numbers in either rows or columns can be automatically added with the Valdocs Calculator, described later.)

Other typical text processing facilities such as margin settings, line centering, justification, searching and changing text, and more are controlled from the various menus, which are described elsewhere.

Block Graphics

Block graphics characters can be displayed on the editor screen just as any regular text character, but cannot be printed as graphics (they appear as regular text characters) in version 1; Valdocs versions 2 and 3 can print them intermixed with text. Block graphics characters include many foreign and special mathematical characters, as well as a large assortment of shapes that can be used to draw boxes around special text to emphasize or isolate it from the main body.

Two "keyboards-worth" of block graphics characters are available. The first set is accessed by holding down the GRPH SHIFT key while typing the desired character. The second set is selected by holding down both the GRPH SHIFT and the standard SHIFT keys while typing the character. A cross-reference chart for block graphics characters is provided in Appendix B.

Editor Menus

Valdocs version 1 provides seven menus. These control a wide variety of editor functions and are described in detail elsewhere. Only one is not available to all skill levels, and is mentioned in detail here because of its effect on editor operation. (Skill levels can be set through the MENU key and are described in detail later.)

The Quirk menu, which is accessed by typing a control-Q while in the editor, provides selections related to editor operation. Quick Display disables the "bit-mapped" mode of operation and speeds up the editor display. Only the "standard" character font can be displayed while using this display mode. (There are no italics, boldface, underscore, or block

graphics.) The Quick display can be used on any existing text, but will display everything with "normal" characters, regardless of what they actually look like.

"Fancy" characters such as boldface and italic cannot be typed while in the Quick mode, and will not show as such unless the display is switched to normal. The Quick Display mode can easily be identified by the cursor—it changes to a small flashing box, rather than the large steady block used by the normal display mode.

The other editing change provided by the Quirk menu is CURSOR UNLOCK. When selected, the cursor can be moved vertically on the screen while the text remains stationary. Additional text is moved onto the screen when the cursor is "moved against" the top or bottom of the screen. This mode removes the page boundary indicators normally provided; these can quickly be restored for inspection by "relocking" the cursor.

File Backup and Other Considerations

Valdocs provides good file control and relatively fast backup capabilities. A current file can be protected from power loss or accidental system reset by typing any function key (the MENU key is the fastest) and then the UNDO key. This simple 20-second procedure can save untold amounts of grief, self-recrimination, and hours of frantic recovery work (which may not be successful) if it is used on a regular basis.

While switching to a Valdocs function, the message "Protecting Current Work" is displayed. Once this message disappears, the computer can be shut off or power can fail without loss of the current document, even though it has not been formally saved.

When the UNDO key is typed to return to the editor, the current document will reappear on the screen with the cursor located at the same place it was earlier. If for some reason power was lost after the backup was performed, the computer will automatically reload the current file and position the cursor where it was when the last backup was done, regardless of the length of the power disruption.

It is generally not safe to remove a disk from the QX-10 unless you have just backed up the file or used the control-M menu to change disks. There are no hardware facilities on the QX-10 disk drives that allow Valdocs to prevent disk removal when conditions are unsafe, or to detect when they were removed.

The prudent user will not change disks, even if "nothing was changed," unless the above safety procedures have been exercised. Valdocs uses a host of temporary files for its own purposes. Users are not advised of when or how many are in use at a given time. "Unauthorized" disk changes can result in the loss of index files and other data important to Valdocs; sometimes such data is recoverable (after hours of very difficult work), and other times the data is destroyed beyond recovery.

Data Recovery Procedures

Valdocs keeps several categories of files on the data disks. Some files, called index files, contain the filenames and other housekeeping data that are normally used by Valdocs. These use the TPM II-CP/M filename extension .NDX. Files of user data use the extension .VAL.

(These are basically text files with a few embedded control characters.) Files created by the DRAW program are stored using the extension .GRF for graphics. Other types of files are used but are not discussed here.

Two general types of data loss can occur: part of the index system can be damaged, or actual data files can be corrupted or lost. Either type of loss can be caused by changing disks at an unfortunate moment.

TPM II offers a utility called BILD, which is sometimes able to partially rebuild a damaged index system. The procedure is relatively straightforward: place the damaged disk in drive B and the TPM disk in drive A, reset the QX-10, then run the BILD program. The full results, if any, will not be known until Valdocs is run again.

If the BILD program is not effective, a second, longer and more painful procedure is required.

Lost data may be recoverable by first making a list of all the filenames on the damaged disk with the VAL extension. This can be done with either TPM II or CP/M using the DIR command, which displays all the files on the disk. If using CP/M, be sure to record the entire filename accurately. (TPM II users can print the directory by typing a control-P to turn on the printer before listing the directory.)

Valdocs files are named using a coding scheme involving the date and sequence number (the number of files saved on that date) to generate unique filenames. A typical Valdocs data filename might be 83206002.VAL. (It must be agreed that filenames like this are not "user friendly," but users who don't randomly change disks without telling Valdocs beforehand will never need to look at them.)

Install the damaged disk in the right drive. Set Valdocs to the Expert mode with the SETUP program available through the MENU key. Using the Valdocs editor, select the Quirk menu. From this menu, select the R function (retrieve a non-indexed file). Enter a filename from your list when prompted; after the file has been loaded into the editor, inspect it for content. The recovered file can be saved in the usual manner using the STORE key, where a new index filename must be supplied.

Once stored as an indexed file, the original file, (the one loaded into the editor with the Quirk menu) should be deleted. The entire process should be repeated until all files have been recovered. The author has determined from personal experience that this process is incredibly enlightening; one such encounter is sufficient to last a good long while and is a powerful inducement for good operating practices.

Rising Star recognizes the difficulties of file recovery and is developing a utility program that will simplify the above procedure. The program will read and display a small segment of a file, then allow the user to either discard it or provide a suitable name and store it as an indexed file.

The HASCI Keyboard

The HASCI keyboard supplied with all QX-10 computers that are equipped to run Valdocs consists of a main typing area patterned closely after the standard office Selectric typewriter, an editing cluster to the right of this, and a numeric keypad on the right side of

the keyboard. All Valdocs operating system controls are located in functionally related groups of keys across the top of the keyboard. These groups are labeled, left to right, the System Controls, the File Controls, Applications, and Typestyles.

The Main Typing Area

In addition to the normal typewriter keys, there are several control keys such as are found only on computers. The typing and editing control keys on the HASCI keyboard are:



Margin Release—This key serves the same purpose as the margin release on a standard typewriter. When the cursor on the screen is stopped by a left or right margin, it can be released to travel to the screen boundaries with this key. Valdocs version 1 does not allow the margin on a single line to be changed more than once in a single editing session.



Tab Set—This key is used to set a tab at the current cursor position. Tabs are indicated on the margin status line at the bottom of the edit screen by a small "T" on top of the line.



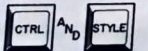
Tab Release—Tab stops can be removed by placing the cursor on the tab stop and then typing this key.



Decimal Tab—The decimal tab, located on the top left corner of the numeric key pad, causes the cursor to be moved to the next present tab location. Any numbers following it are aligned over the tab position. Those numbers without decimal points are arranged so the left-most digit is one character to the left of the tab position. Any numbers with a decimal point are automatically positioned with the decimal point over the tab location.



Shift Lock—This key operates and performs the same way as on a typewriter. The shift characters on all keys are printed. This key is lighted when active. Normal operation is restored when either shift key is depressed.



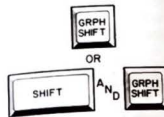
Control-STYLE—When the control key is depressed and the STYLE key is typed, the alphabetic section of the keyboard reverses the shift key. Unless the shift key is depressed, all alphabetic characters become capitalized. Lower case characters can be typed if the shift key is held down while they are entered. Normal operation is restored when Control-STYLE is typed a second time.



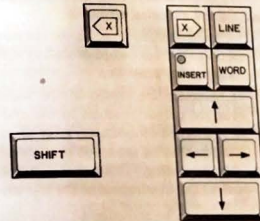
CTRL—The control key is used with the letters O, P, K, L, M, B, and Q to select editor control menus and to cause the edit cluster up and down arrow keys to automatically repeat.

(If the screen then scrolls past the desired location, use the red STOP key on the top left corner of the keyboard to stop scrolling.) Additional control functions can duplicate most of the edit cluster functions. (See Table 6-1.)

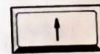
Graphics Shift or Graphics Shift and Shift—These key combinations take the main keyboard out of the normal alphabetic mode to provide two different sets of block graphics characters. See Appendix B for character design and keyboard location.



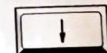
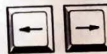
The cursor position is controlled by the edit cluster keys plus the erase key (the top left corner key on the main typing area) and the main typing area shift key.



These keys have specific and very carefully controlled functions as follows:



The arrow keys are used to move the cursor one space or line in the indicated direction.



If the WORD key is typed immediately following a left or right arrow, the cursor is moved to the beginning of the next word in the indicated direction.



If the LINE key is typed immediately after a right or left arrow, the cursor is moved to the indicated end of the line.

the keyboard. All Valdocs operating system controls are located in functionally related groups of keys across the top of the keyboard. These groups are labeled, left to right, the System Controls, the File Controls, Applications, and Typstyles.

The Main Typing Area

In addition to the normal typewriter keys, there are several control keys such as are found only on computers. The typing and editing control keys on the HASCI keyboard are:



Margin Release—This key serves the same purpose as the margin release on a standard typewriter. When the cursor on the screen is stopped by a left or right margin, it can be released to travel to the screen boundaries with this key. Valdocs version 1 does not allow the margin on a single line to be changed more than once in a single editing session.



Tab Set—This key is used to set a tab at the current cursor position. Tabs are indicated on the margin status line at the bottom of the edit screen by a small "T" on top of the line.



Tab Release—Tab stops can be removed by placing the cursor on the tab stop and then typing this key.



Decimal Tab—The decimal tab, located on the top left corner of the numeric key pad, causes the cursor to be moved to the next present tab location. Any numbers following it are aligned over the tab position. Those numbers without decimal points are arranged so the left-most digit is one character to the left of the tab position. Any numbers with a decimal point are automatically positioned with the decimal point over the tab location.



Shift Lock—This key operates and performs the same way as on a typewriter. The shift characters on all keys are printed. This key is lighted when active. Normal operation is restored when either shift key is depressed.



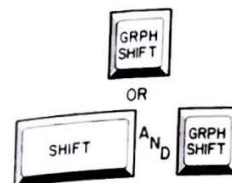
Control-STYLE—When the control key is depressed and the STYLE key is typed, the alphabetic section of the keyboard reverses the shift key. Unless the shift key is depressed, all alphabetic characters become capitalized. Lower case characters can be typed if the shift key is held down while they are entered. Normal operation is restored when Control-STYLE is typed a second time.



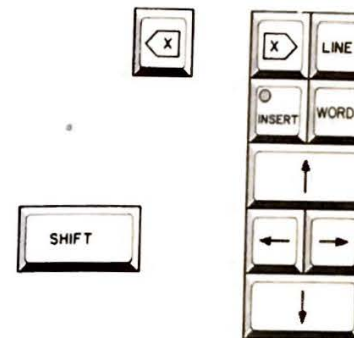
CTRL—The control key is used with the letters O, P, K, L, M, B, and Q to select editor control menus and to cause the edit cluster up and down arrow keys to automatically repeat.

(If the screen then scrolls past the desired location, use the red STOP key on the top left corner of the keyboard to stop scrolling.) Additional control functions can duplicate most of the edit cluster functions. (See Table 6-1.)

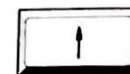
Graphics Shift or Graphics Shift and Shift—These key combinations take the main keyboard out of the normal alphabetic mode to provide two different sets of block graphics characters. See Appendix B for character design and keyboard location.



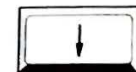
The cursor position is controlled by the edit cluster keys plus the erase key (the top left corner key on the main typing area) and the main typing area shift key.



These keys have specific and very carefully controlled functions as follows:



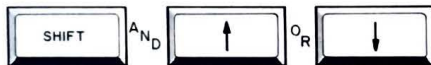
The arrow keys are used to move the cursor one space or line in the indicated direction.



If the WORD key is typed immediately following a left or right arrow, the cursor is moved to the beginning of the next word in the indicated direction.



If the LINE key is typed immediately after a right or left arrow, the cursor is moved to the indicated end of the line.



Text can be scrolled up or down 11 lines (half a page) by holding down either shift key while typing the up or down arrow keys.



The insert key is used to select the typing mode of the keyboard. When the key is lighted, anything typed is inserted before the character under the cursor. The text is moved to the right as needed. When the insert key is pressed and the indicator light is off, anything that is typed replaces the text under the cursor. The two modes are indicated in the status display at the bottom of the editor screen. The two modes alternate with each press of the key.



The delete keys erase characters under the cursor as it moves in the indicated direction.



Characters under the cursor are erased in the indicated direction to the next word boundary if the WORD key is pressed after either erase key.



Characters are erased in the indicated direction from the current cursor position to the end of the line if the LINE key is typed immediately after an erase key. The screen may "fall behind" if the erase keys are held down long enough to start repeating. This can be stopped immediately with the red STOP key located at the top left of the keyboard.

Many of the edit cluster functions can be activated by using control keys. The functions listed below are implemented on Valdoks version 1 editors. Notice that the functions "Top of Document" and "Bottom of Document" have no direct edit cluster key equivalent, and that only the right erase keys are available as control functions.

Table 6-1. EDITOR CONTROL KEY FUNCTIONS

Control Key	Function
A	Move cursor left a word
S	Left arrow
D	Right arrow
F	Move cursor right one wordspace
Q	Delete right character
W	Move cursor to top of document
E	Up arrow
R	Prior page
T	Delete right one word
Y	Delete right end of line
Z	Move cursor to bottom of document
X	Down arrow
C	Next page

Valdoks Editing Menus

Valdoks version 1 editing provides seven menus for screen formatting and other controls. A complete listing of all the selections in each menu is given in Table 6-2. The following is a brief overview of each menu and what it can do. The menus are called to the screen by typing the control version of each of the letters listed below:

B—the Block menu. The Block menu allows a single block of text to be marked and extracted and stored as a separate file on disk, to be moved, deleted, or copied to other areas of the document. Up to nine separate "placemarks" can be placed throughout a file. The cursor can be directed to them more or less instantly.

K—the K menu. In Valdoks version 1 it is identical (and undocumented) to the M menu described below. This menu will be used with the spelling checker functions in Valdoks versions 2 and 3.

L—the Look menu. This menu provides search and replace functions for the Valdoks editor. The cursor can also be moved to the top or bottom of the document.

M—the Miscellaneous menu. This menu makes it safe to change disks, reports the amount of disk space remaining on the data disk, erases the current document being worked on, deletes text to the end of the current document, turns justification on and off (this squares the right margin of paragraphs), and can turn off automatic word-wrap.

O—the On-Screen Format menu. Top, bottom, left, and right margins are set here. Lines can be centered and non-breakable spaces created. (A non-breakable space is used when two adjacent words must appear next to each other. Ordinarily, they could be split if one happened to fall at the end of a line and was wrapped to the following line. The unbreakable space makes the two words appear as one when word-wrapping is applied.)

P—the Page menu. This menu allows the cursor to be placed on a specific page, a new page to be started or an old page to be ended to avoid "widows and orphans" (single lines or line fragments at the tops and bottoms of pages). It also defines the length of pages in lines.

Q—the Quirk menu. This menu is available only when Valdoks is operated in the expert mode. The expert mode is selected through the MENU Applications key by the SETUP program. The Quirk menu is used to list the directory of the data disk using the TPM-style format, log in a different drive, format a non-Valdoks document (one created with some other editor such as PeachText) so the Valdoks editor can work on it, select the "quick" display described earlier, fetch non-Valdoks files for editing and store a Valdoks file for use by non-Valdoks programs, free the cursor from the centerline of the screen (described earlier), and turn the indexer on or off to allow files to be saved and loaded more quickly. It does this with standard TPM II-CP/M filenames.

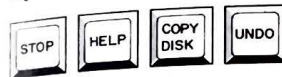
The following is a detailed listing of all selections available through the Valdoks editor menus. Menu selections are made by typing the first character within the brackets of the desired selection followed by a carriage return, or by positioning the cursor with the edit cluster arrow keys.

Table 6-2. VALDOCS VERSION 1 EDITOR MENUS

Control B—Blocks	
<M>ark block (beginning or end)	<C>opy block to cursor location
<G>o to placemark 1-9	<R>elocate block to cursor
<P>lacemark setting	<D>elete/Unmark block
<U>nmark placemark	<S>tore block as document on disk
Control L—Look For	
<L>ook-for text	eginning of document
<A>gain repeat LOOK-FOR or REPLACE	<E>nd of document
<C>onditionally replace text	<R>eplace text
Control M—Miscellaneous	
<C>hange data disk	<D>elete to end of document
<S>pace available on data disk	<J>ustify on/off
<E>rase screen, refile original document	<W>ord-wrap on/off
Control O—On-Screen Editing	
<C>enter line on/off	<R>ight margin setting
<N>on-break space	<T>op of page margin
<L>eft margin setting	ottom of page margin
Control P—Page	
<G>o to specific page	
<N>ew page (go to top of new page)	
<I>f within 'n' number of lines from end, new page	
<S>et length of paper to 'n' number of lines	
Control Q—Quirks	
<D>irectory of disk	<R>etrieve non-indexed file
<L>og in different disk drive	<S>tore as non-indexed file
<F>ormat entire document	<C>ursor locked in center
<G>uick Display	<I>ndexer active

VALDOCS OPERATING CONTROLS

The Valdocs operating environment is managed, as mentioned before, by three clusters of functionally related control keys. These are labeled SYSTEM CONTROLS, FILE CONTROLS, and APPLICATIONS. In most cases, each of these keys performs menu-driven functions that are obvious from the label on each key. The exceptions are the HELP key, which provides instant on-screen help throughout Valdocs, and the UNDO key, which will be described shortly.

System Controls

The System Control group is made up of four keys: STOP (colored red), HELP, COPY DISK, and UNDO. The STOP key can be used in any program to stop current execution and is similar in some ways to the break key found in other computer systems. In the Valdocs system, this key is generally used to stop or abort most listings currently being printed, halt screen scrolling, stop a disk copy operation, and when typed while holding down the control key, instantly blank the screen. (Normal screen operation returns as soon as any printable key is typed.)

The HELP key provides detailed, on-screen assistance in nearly every part of Valdocs. Some menus are considered self-explanatory, but in most cases, help is available. This limits the need (and size) of a Valdocs manual. In version 1.8, the HELP file consumes about 38K bytes of disk space. If no use of HELP is anticipated, the file HELP.TXT can be deleted to recover this space on the Valdocs program disk in disk drive A.

The DISK COPY key allows disks to be copied, formatted, and verified, but not compared. Single files or the whole disk can be copied. The menu selection "Make a Data Disk" is the translation of "format a disk." The selection "Disk Management" allows disk directories to be recovered, new directories to be made, checks to be run for bad sectors on the disk, and deleted files to be recovered. Disk management should be used only by experienced computer users and is not meant for the weak at heart. It is possible to destroy disk data with this program.

The Undo key is a powerful key for use with the Valdocs editor. It allows text that was recently deleted to be returned (un-doing the deletion). The UNDO key is also extensively used to exit menus when no selections are desired.

File Controls



The File Controls section is made up of five or six keys, depending on keyboard model. (The newer Brother keyboard provides a separate EDIT key. It allows the edit function to be entered directly, without use of the UNDO key. Each key is identified with a fairly self-explanatory label. The STORE key is used to store documents prepared by the editor; the RETRIEVE key allows a file to be selected and loaded from disk storage; the PRINT button causes files or screens to be printed; the INDEX key is similar in Valdocs version 1 to the RETRIEVE function (and may actually be used by the RETRIEVE and PRINT functions, as explained later), but in versions 2 and 3 has expanded functions; and finally, the MAIL key selects functions related to the electronic mail system.

STORE is used whenever a document is to be placed in storage on a disk. Operation is simple and self-explanatory: press the store button. A listing of all the latest files stored on the disk is displayed, and a filename is requested. If a document already on disk was being edited, the original name is supplied. A new name can be typed over it or a return can be typed to replace the original file with the edited version. No automatic file backup system is available.

Valdocs indexed files can have very long filenames, up to 16 words in two lines, plus 32 "noise" words such as "and," "the," "but," and "with." Valdocs files are called "indexed" files; the index references are kept in an index, while the real file, the one containing the actual file data, is kept in a standard TPM II-CP/M file with the extension .VAL. (The TPM II filename is made up of the current date and a sequence number based on the number of files created on that particular date. A standard TPM II file is called a non-indexed file. TPM II and indexed files can be passed in and out of Valdocs by use of the Quirk menu in the editor.)

The RETRIEVE key is used to fetch a file from disk storage and place it in memory for use by the editor. When this key is activated, the current document (if any) is placed in temporary storage, and a screen showing the latest files stored is displayed. If an entire filename is typed, it is loaded into the editor immediately. However, if the exact name of the file is not recalled, any word or fragment of word that is in the indexed filename can be typed. The file indexer will then automatically present an alphabetical list of all files having that word or fragment in the filename; the desired file can then be selected by positioning the cursor next to it and typing a return.

The PRINT key is used to control printing operations using the built-in printer spooler. As many as 7 files can be simultaneously scheduled for printing in version 1. Any text or graphics file can be printed this way, but catalogs and file indexes cannot be scheduled; the QX-10 cannot be used for other tasks while these are being printed.

The print spooler allows the starting and ending pages to be defined and multiple line spacing to be chosen, and defines the number of copies (1 to 99) to be printed and the first page number. If an Epson printer is used, the correspondence-quality mode can also be used for improved print quality. A single sheet mode is also available to allow insertion of letterhead or single sheet paper between pages.

The STOP key presents a menu that can be used to stop the printing of the current file or all files presently in the spooler.

Once a file has been scheduled for printing, Valdocs returns control to the editor. Any other computing job can be performed except disk copy, which would use the disk drive where the file being printed is stored.

The INDEX key in Valdocs version 1 works much like the RETRIEVE key, except that files are chosen by moving a cursor next to the filename and pressing the return key. Filenames can be sorted for examination and easier selection by sequence number (the order in which they were created and saved), by a straight alphabetical listing, or by a cross-referenced listing (when the filenames are arranged alphabetically by each word in the filename). This is a powerful aid to memory when the exact filename cannot be recalled, but a topic or word used in the filename is remembered. The INDEX menu also allows files to be renamed and deleted.

The INDEX key can be used from the editor in place of the RETRIEVE key. The difference is that RETRIEVE requires the entire filename to be typed in correctly; if this weren't done, Valdocs would control to the indexer, which allows file selection by moving the cursor.

When files are loaded with either the INDEX or RETRIEVE controls, they are actually *inserted* into the current file where the cursor was located when the file was retrieved. This is a powerful and useful feature. Blocks of "standard" text, such as a letter opening, body, and closing, can be inserted into a new document.

The MAIL key provides access to the electronic mail system built into Valdocs. The MAIL system menu allows inspection of the "inbasket" and "outbasket," (logs automatically kept by the MAIL system of incoming and outgoing files), use of a "person-to-person" network, address book maintenance, and of course, mail to be sent. Any file on the QX-10 can be sent as mail and scheduled to be automatically transferred at any time of day.

The inbasket and outbasket are actually logs kept automatically by the mail system to show what files or messages have been received or sent. The person-to-person network allows the QX-10 to be used as a terminal when dialing into other computer systems or bulletin boards.

The address book is used to record the name, address, voice, and computer telephone numbers of people and businesses. The address book is kept on either the Valdocs program disk or any data disk. A mailing label program built into the address book menu allows mailing labels to be designed and printed directly from the book. Users can make as many different address books as desired simply by recording them on different disks. The only restriction on size or number of entries is the space on the actual disk.

Valdocs Applications



The two HASCI keyboards available for the QX-10 differ in the applications area. The original model had only the MENU, CALC, SCHED, and DRAW keys located here. A fifth key labeled EDIT was inserted between the menu and CALC keys on the newer Brother keyboard. Since the EDIT key is not used by Valdocs version 1, operation is unaffected.

The MENU key is used to select and run programs that are not necessarily a part of Valdocs, to obtain an alphabetized listing of all the files on a disk, and to exit to the TPM II operating system (if Valdocs has been set to the Expert mode). Valdocs applications programs can also be selected and run from the MENU key. Only three applications programs were initially available with version 1: RESTART, a quick way to re-initialize Valdocs; SETUP; and VERSION (used to give the revision date of each Valdocs file).

SETUP is a program that allows the date and time to be set (this can also be done in a more user-friendly fashion with the scheduler), the time format to be switched between 12 and 24 hour displays, the serial port communications modes to be set, experience level for Valdocs to be selected, the terminal type for the QX-10 to be changed (normally it is used in the Televideo 920 mode), the type of printer used with the QX-10 to be defined, and to specify the type modem used to communicate on the telephone lines to be specified.

Since it is possible to set the ports into an incommunicative state, inexperienced users will probably not want to tamper with any of these settings. Version 1 supports only a limited number of terminal modes, modems, and printers. Users should contact local dealers before purchase of products not specifically supported by Valdocs before deciding what to buy.

The CALC key presents an instant four-function calculator (plus memory) on the screen of the QX-10. It can be used to automatically add columns or rows of numbers and can be used to enter numbers in files being edited. The Brother HASCI keyboard added a sign inversion key (+/-) and removed the = key from the original QX-10 numeric pad. Later versions of Valdocs will include a full-function scientific calculator option.

The SCHED key is used to call up an electronic calendar-notebook system. It provides an appointment book and notepad, an event timer, and a general purpose calendar-clock. The appointment and note page for any date or group of dates can be printed using the "Itinerary" option.

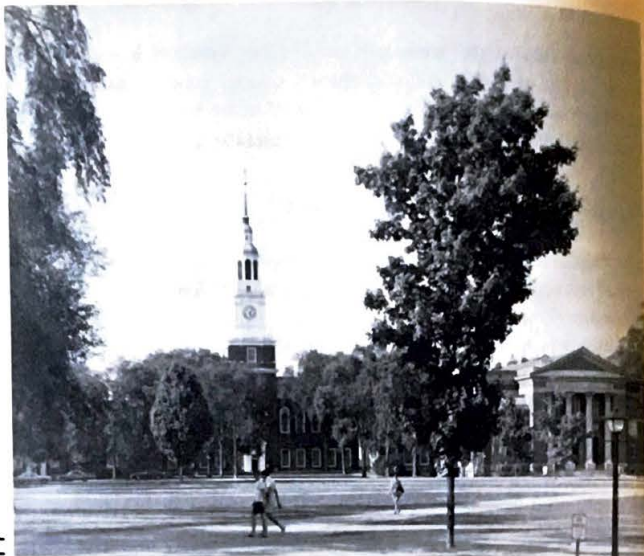
The DRAW key presents a flexible business graphics program that can make pie charts, and bar, line, and scientific graphs automatically from information entered from the keyboard. The results can be displayed on the QX-10 screen, stored as files, and printed on any Epson dot matrix printer. Operation is simple: data is entered, one element at a time, in response to prompts requested by the DRAW program.

Although Valdocs version 1 does not allow the output of this program to be used by the

editor, or processed for size or orientation, versions 2 and 3 will permit these graphs to be even more automatically generated, scaled, rotated, and inserted in standard text files. Additionally, version 3 will provide a powerful three-dimensional CAD/CAM graphics package tied into the data base management system.

VALDOCS—AN OVERVIEW

Valdocs is a dynamic evolving computer system that will eventually be incorporated on other computer systems. It is a genuine innovation because of its uncommonly non-technical approach to computing. For instance, the editor is automatically set up in such a manner that many users will never have to change any of the settings or use the menus. But advanced and more complex features are instantly available through means written in "English." The ultimate goal of Valdocs, when it reaches version 3 sometime in late 1984 or early 1985 is to "provide 90% of the entire software needs of 90% of all computer users".



BASIC on the QX-10

THE DARTMOUTH CONNECTION

The BASIC language was originally developed under a National Science Foundation Grant by John G. Kemeny and Thomas E. Kurtz, two mathematics professors at Dartmouth College, located in Hanover, New Hampshire. These two computer pioneers started their work at Dartmouth on a Burroughs LGP-30 computer (which had only a drum memory) in 1959.

In 1963, the trustees of Dartmouth decided to introduce the computer as a regular part of the Liberal Arts program at a time when the major programming language in use was FORTRAN (FORmula TRANslator), and computers were run in "batch mode". What batch mode meant was that each program to be run was first punched on a deck of cards (usually by the user) and hand carried to the computer center. There, the cards waited in

Dartmouth College, home of the BASIC language. This scene shows Baker Library, the center of the campus. College Hall is located just to the left of this picture; it was the location of the first Dartmouth computer, where BASIC and the beginnings of DTSS were developed. The present computer center, Kiewit, is located just behind Baker Library. Photograph courtesy Dartmouth College.

turn until they were loaded and run on the system as a separate task. Twenty-four hours later, the cards and any resulting output were picked up.

As can easily be seen, any error that caused the program to stop would delay it until the next day. At that time, the user would discover and correct the mistake, perhaps a missing period or a typographical error, then start the process over again. It could take weeks to get meaningful output from a moderately complex program.

Dartmouth concluded that the logistics of seven or eight hundred students making daily pilgrimages to the computer center would make a batch processing system impractical, and so elected to develop a time-sharing system that, during the early 1960's, was on the leading edge of software technology.

Two faculty members and a group of undergraduate research students developed the beginnings of what is now called "DTSS" or the Dartmouth Time Sharing System. It was originally run on a donated General Electric 235 computer in the basement of College Hall and first placed in operation in the fall of 1964. This computer system was replaced with a much larger GE 635 when the computer center was moved into the Kiewit building (a donation of Peter Kiewit, a Dartmouth Alumnus and founder of a large construction firm bearing his name in Nebraska).

After the time-sharing system was well developed, General Electric and Dartmouth went separate ways. General Electric's version became known commercially as the GE Mark II Timesharing System, and Dartmouth's, of course, was (and still is) called DTSS.

The FORTRAN programming language, although powerful, was considered too complicated for beginning computer users (average Dartmouth students), and so BASIC (Beginners All-Purpose Symbolic Instruction Code) was written by Kurtz and Kemeny for ease



Professors John Kemeny (right) and Tom Kurtz (left), authors of the BASIC language. Kemeny has served as President of Dartmouth and Kurtz as Director of the Kiewit Computation Center. Both have since returned to the Mathematics Department at Dartmouth. Photograph courtesy Dartmouth College.



The Kiewit Computation Center is a familiar sight to nearly every student at Dartmouth. Only a very small percentage of students manage to survive four years at Dartmouth College without using the computer in at least one course. Photograph courtesy Dartmouth College.

of use. The reason BASIC has been so successful is that it is written in several levels, even today. The first level of the latest Microsoft offering, for example, is no more complicated to use than Dartmouth's BASIC of 1964. It can be mastered by nearly anyone in less than an hour or so.

BASIC was made exceptionally "user friendly" by 1964 computer language standards. This meant that, for the first time, the programmer did not have to worry about formats, number types, field and file specifications, and the host of other things necessary for operation in a large computer environment.

Another major innovation was the use of "meaningful" error messages. Instead of a message like "Error 5B\$F at C491 ** FE20/RR 6D52" the user was treated to "Missing Parenthesis in Line 140." BASIC was written from the start to be used by beginners.

(We can also probably give Dartmouth at least partial credit for pioneering the popular use of the computer for games applications. This was—and still is—considered good training; it helps reduce the fear some people have of computers while teaching the use of the system. Dartmouth was also one of the first educational institutions to treat the computer as a library resource. Terminals abound in virtually every nook and cranny of the institution, and anyone connected to the college (as well as visitors) is given free access to the computer from any terminal in public areas. Dartmouth is a true "hacker's paradise.")

MICROSOFT BASIC

The BASIC most commonly found on microcomputers is one of the many versions from Microsoft. The two founding fathers of Microsoft are Paul Allen and Bill Gates. (When the first BASIC for a microprocessor was written, Gates was attending Harvard; Allen, who was a couple of years older than Gates, was already employed at Honeywell.)

These two individuals were probably prototypes of today's typical teenage "computer hacker"; both became interested and involved with early computers (Digital Equipments PDP-10) since junior high school (Gates was in the seventh grade when he first used a time-sharing computer), and had been toying with the idea of writing a BASIC for microprocessors since the Intel 8008 was announced in the spring of 1972.

Intel introduced a series of microprocessor chips, starting with the 4004. It was originally developed in 1969 for Busicom, a Japanese calculator company that is now out of business. The chips were placed on the open market for non-calculator applications in 1971. The 8008 (developed from work done for Computer Terminals Corporation, now Datapoint, a CRT terminal manufacturer) was introduced in April of 1972. Finally, two years later, the 8080 was introduced in April of 1974. MITS, a small company in Albuquerque, introduced the Altair computer kit (\$350) in the January and February issues of *Popular Electronics* in 1975.

During February and March of 1975, Gates and Allen worked 18 hours a day in 312 Daniels (a dorm room at Harvard) to write an 8080 emulator for the PDP-10 available at Harvard (an emulator is a program that makes one computer act like another—in this case it reduced a PDP-10 to an 8080). A BASIC interpreter was then developed that could run on the emulator. It required less than 4K bytes of memory.

When the interpreter was finally running, Allen flew it to MITS, on the way writing a program to load the program (it was punched on a roll of paper tape) into an Altair computer. When Allen arrived at MITS, the BASIC program had never been run on anything other than the homemade emulator and had never even been loaded on an Altair computer.

By all accounts, the loader program written on the airplane and BASIC both ran, a remarkable feat. Two months later, Allen left Honeywell and Gates went on leave from Harvard (his current status there). The two formed Microsoft, which is now a very successful software company.

BASIC FOR THE QX-10

Two versions of Microsoft BASIC are presently available for the QX-10. In Europe, it is called "MFBASIC," and in the United States it is called "Microsoft BASIC Revision 5.22." A third, more powerful BASIC is available from Rising Star Industries, authors of the VALDOCS Software.

The United States BASIC is a "straight" Microsoft version and is not adapted in any way for the QX-10; not even the time and date from the internal calendar-clock can be directly referenced by it. There are no graphics commands for use with the QX-10 screen, and the line editor is the same confounding Microsoft editor that has been supplied for years. None of the editing keys found on the QX-10 keyboard are used by the BASIC line editor. Users of this version of QX-10 BASIC will have to put up with an editor and BASIC that is only as convenient to use as it was in 1980.

The European MFBASIC is an extended version of the one supplied in the United

States. Comprehensive graphics abilities have been added; this addition makes excellent use of the 640 × 400 line resolution of the QX-10 screen. A proper screen editor that uses the keyboard insert and delete keys has also been added.

The "MF" in MFBASIC stands for "MultiFont." In Europe, the standard QX-10 includes a MultiFont character generator card that contains six 128K byte read only memories and an 8049 CPU. Firmware for the card is contained in a single 2716 EPROM.

When the MultiFont card is activated by the European CP/M program MFONT (or automatically by MFBASIC), 1 of 16 fonts can be displayed on the screen. (See Table 7-1 for a listing of these fonts.) The fonts are selected by the SF function keys that are located above the calculator keypads on the ASCII keyboard. The QX-10 screen width is cut down to 40 character rows when any of the "special" character sets generated by the MultiFont card are used. The special font characters, once inserted into a listing or text file, are carried into files as normal characters. They are printable on standard Epson printers, using the graphics mode as required to print them.

The Rising Star BASIC available from Rising Star Industries beginning in the early spring of 1984 is a powerful, high performance BASIC interpreter with both X-Y and turtle graphics, exploiting the QX-10 graphics capability to the fullest. Requiring the TPM operating system, this RSI BASIC is priced in the \$100 range.

The origins of RSI BASIC go back nearly as far as Bill Gate's Microsoft versions. Originally written in the latter part of 1977 for TDL (see Chapter 1), it is optimized for the Z-80A microprocessor. Combined with a highly efficient floating point arithmetic package, this full-featured BASIC provides over 170 keywords and commands.

RSI BASIC offers a unique combination of features, giving it a flavor all its own. Although definitely not a "Microsoft Clone," most BASIC programs written for MBASIC can easily be modified to run on the QX-10, using RSI BASIC. The most obvious exceptions will be those with graphics output.

The graphics implementation in RSI BASIC provides a rich assortment of features not found in any other BASIC. Commands for QX-10 computers equipped with the color option (to be made available during 1984) allows brilliant, high resolution color graphics output. Even without color, the QX-10 resolution of 640 by 400 dots is double that of most competing computer systems.

Graphics commands provide three independent screens; each can be displayed separately. Screen 1 is a bit-mapped screen used to display both text and graphics data. Screen 2 is normally a standard alphanumeric screen with 25 lines of 80 characters each, but on color QX-10 computers it is also bit-mapped for both alphanumeric and graphic display. Screen 3 is a special alphanumeric screen. Using the DISPLAY command, it can be programmed to have between 0 and 12 lines of 80 characters each and can be overlaid on either screen 1 or 2. All screens can be scrolled.

A wide variety of additional graphics commands allows circles, arcs, and rectangles to be automatically drawn at very high speed, due in part to the graphics controller inside the QX-10. Enclosed areas can be automatically filled with solid colors or predefined patterns.

Table 7-1. MULTIFONT*

Standard Font	Light Italic
Bodoni	Bodoni Italic
Helvetica Light	Helvetica Light Italic
Helvetica Medium	Helvetica Medium Italic
Old English	Old Germany
Flash Bold	American Typewriter Extended
American Typewriter Medium	Sans-Serif Shaded
Broadway	Commercial Script

*The above fonts can be selected by QX-10 computers using the MultiFont option card, which is not presently sold in the United States. Font selection is made by pressing one or more of the SF function keys in combination on the ASCII keyboard.

Screen dump commands can save a whole screen or a defined area from the screen to a disk file for later use. Screen data can also be sent directly to the FX-80 printer for full or partial screen dumps.

RSI BASIC is one of the very few that offers both full-featured, standard color graphics and a turtle-graphics implementation. Turtle-graphics commands allow both relative and absolute turns, moves, and draws. (A relative move example is "MOVE 100," meaning "move 100 units from the current position on the present heading"; an absolute example is "MOVETO (100,200)" meaning "move from present position to location X=100, Y=200.")

Many common DOS commands are available directly from RSI BASIC using the same command names and formats normally used in DOS. A "RESAVE" command is available to automatically make the current disk file into a backup copy, then save the current BASIC program under the same filename.

There are many other interesting BASIC commands and conveniences available that set RSI BASIC apart from MBASIC. While screen editing is not available in RSI BASIC, the line editor does allow use of the entire QX-10 edit cluster, a real relief from MBASIC's primitive approach. QX-10 users interested in programming will no doubt want to investigate RSI BASIC and Helios, a Forth-like language with all the features of Forth-83, also available from Rising Star.

SOME APPLICATIONS UTILITIES USING BASIC

A Caution

The following programs operate directly on the QX-10 hardware. If these programs are run, care must be used to ensure that they are correctly entered into the QX-10. Do not use the POKE or OUT instructions indiscriminately. Under certain conditions, it is possible to cause physical damage to your QX-10 CRT monitor by misprogramming the 7220 CRT controller chip. Make sure the POKE statements in your program are entered correctly before running it.

Redirecting LPRINT Output

BASIC can be used to directly control a few of the input/output functions of the QX-10 computer. (Note: RSIBASIC includes commands that do this automatically for you.) It might be desirable, for example, to direct output normally sent to the parallel printer to the serial port. This is especially true if a dot matrix printer is already connected to the printer port and some other output device, such as a plotter or speech synthesizer, was used on the serial port.

The BASIC program shown in Listing 7-1 demonstrates how CP/M's IOBYTE can be changed under program control to automatically send data output by the BASIC LPRINT command to either the parallel or serial port. The IOBYTE, which is maintained in memory location 3, is used by CP/M to control where output for various peripheral devices should be sent. Normally, this byte is altered by the STAT or SETUP utility program from CP/M, but it can also be changed using the BASIC POKE command.

Memory location 3 is split into four sections, each consisting of two bits. These bits are set to specifically control "logical" devices consisting of a listing device, a punch, a tape reader, and a console. In the QX-10, these logical devices are assigned to the physical devices as shown in Table 7-2.

The IOBYTE is organized into four two-bit code groups as follows:

IOBYTE: LL PP RR CC

LL directs the listing output

PP directs the punch output

RR directs the reader input

CC selects the console device

The following two bit codes select the physical device:

00 selects the QX-10 keyboard-screen

01 selects the QX-10 keyboard-screen

10 selects the parallel printer port

11 selects the serial port

Table 7-2.

Logical Device	QX-10 Physical Device
Listing Device	Parallel Printer Port
Punch	Video Display
Tape Reader	Keyboard
Console Input	Keyboard
Console Output	Video Display

The program in Listing 7-1 shows how these selections, when incorporated into IOBYTE, redirect the output generated by the BASIC LPRINT command to the console screen, to the serial port, or to the parallel printer port.

Listing 7-1.

```

10 REM PROGRAM TO DEMONSTRATE HOW TO DIRECT LINE PRINTER OUTPUT
15 REM TO OTHER PORTS.
20 REM THIS OUTPUT WILL GO TO THE LINE PRINTER
30 POKE 3,129 : REM SELECT LINE PRINTER OUTPUT
40 GOSUB 170
50 REM THIS OUTPUT WILL PRINT ON THE SCREEN
60 POKE 3,1 : REM SELECT SCREEN FOR LINE PRINTER OUTPUT
70 GOSUB 170
80 REM THIS OUTPUT WILL GO TO THE SERIAL PORT
90 POKE 3,193 : SELECT SERIAL PORT
100 GOSUB 170
110 REM NOW SET OUTPUT BACK TO PARALLEL PORT
120 POKE 3,129
130 LPRINT "THAT'S ALL, FOLKS!"
140 STOP
150 REM
160 REM
170 REM SUBROUTINE TO OUTPUT A TEXT MESSAGE TO PRINTER
180 LPRINT "NOW IS THE TIME FOR ALL GOOD MEN TO RAISE THEIR VOICES"
190 RETURN
200 REM
210 REM
220 REM
1000 REM THIS PROGRAM DEMONSTRATES HOW THE IOBYTE IN MEMORY
1010 REM LOCATION 3 CAN REDIRECT LINE PRINTER OUTPUT TO OTHER PORTS.
1020 REM THE CHANGE IS MADE BY USE OF THE POKE COMMAND
1030 REM
1040 REM          POKE 3,N
1050 REM WHERE 3 IS THE MEMORY LOCATION OF THE IOBYTE AND N IS ONE
1060 REM OF THESE NUMBERS:
1070 REM
1080 REM 129 --- OUTPUT GOES TO THE PARALLEL PRINTER PORT
1090 REM 193 --- OUTPUT GOES TO THE SERIAL RS232C PORT
1100 REM 1 --- OUTPUT GOES TO THE QX-10 SCREEN
1110 REM
1120 REM YOUR BASIC PROGRAM SHOULD RESTORE THE IOBYTE TO 129 BEFORE
1130 REM TERMINATION

```

Outputting a Byte Directly to the Printer

It is sometimes necessary to output a character or byte of information directly to a printer without using the normal BASIC PRINT statement. This is especially the case when graphics data is being printed. RSIBASIC provides commands to output untouched character strings for just these applications.

The PRINT statement in BASIC may try to maintain the length of printed lines by inserting carriage returns and line feeds; it may also modify, delete, or add characters that are "control" codes. (These might include carriage returns, linefeeds, or formfeed characters, among others.) In graphics output, all characters are "ordinary data" and must be output exactly as sent, without additions, deletions, or changes. Unfortunately, the PRINT output routines often don't know that, and the result is usually chaos.

The program shown in Listing 7-2 demonstrates how a byte of data can be output directly to the printer port without handshaking (a hardware consideration), and without any possible interference from BASIC. Some peripheral products may require a delay to be established between each character output to take the place of handshaking; in this case, the subroutine shown in the remark statements can be used.

Listing 7-2.

```
10 REM A PROGRAM TO OUTPUT DATA DIRECTLY TO THE PARALLEL PORT
20 REM NO HARDWARE HANDSHAKING IS USED
30 FOR N = 33 TO 68 : REM OUTPUT ASCII CHARACTERS 33 TO 68 (! TO D)
40 OUT 20,N : REM OUTPUT CHARACTER CODE "N" TO PORT 20, THE DATA PORT
50 OUT 22,0 : REM NOW GENERATE A "STROBE" TO CAUSE THE DATA TO BE READ
60 OUT 22,1 : REM BY THE DEVICE CONNECTED TO THE PRINTER PORT
70 REM IF A DELAY IS NEEDED (ALL CHARACTERS NOT RECEIVED BY THE PRINTER)
80 REM INSTALL IT HERE: FOR X=1 TO P:NEXT P WHERE P IS A DELAY NUMBER
90 NEXT N
100 LPRINT " THIS IS THE END OF THE PROGRAM"
110 END
```

Serial Baud Rate Control

Although the SETUP utility supplied with CP/M makes it possible to change the serial baud rate of the QX-10 port, it is sometimes more convenient to change this directly from inside a BASIC program. The procedure is simple, but great care must be exercised because the port can be set to an "in-between" speed that cannot be used by standard devices.

The serial baud rate in the QX-10 is controlled by one of the six interrupt timers that are built into the QX-10 electronics. The rate is altered by simply changing a two-byte value of a register inside the timer chip. The program shown in Listing 7-3 automatically runs

through seven standard baud rates, beginning with 9600 (960 characters per second) and down to 110, ten characters per second.

A table within the program shows the actual values sent to the timer. These are broken into a two-byte figure by the program and are sent in sequence to the timer. Characters sent after the register has been loaded will go out at the new baud rate.

Listing 7-3.

```
10 REM A PROGRAM TO DEMONSTRATE SERIAL SPEED CONTROL
20 REM IF YOUR SERIAL DEVICE DOES NOT HAVE ALL THE SPEEDS
30 REM OUTPUT BY THIS PROGRAM IT WILL PRINT GARBAGE
40 REM AT THOSE SPEEDS. BEWARE!
50 REM
60 REM SELECT SERIAL PORT AS LINE PRINTER
70 POKE 3,193
80 FOR R = 1 TO 7
90 READ A$,X : REM A$ IS THE BAUD RATE FOR SCREEN ID, X IS A NUMBER
100 PRINT "SET YOUR SERIAL DEVICE TO ";A$;" BAUD."
110 PRINT "HIT ANY KEY WHEN READY."
120 IF INKEY$="" GOTO 120 : REM WAIT FOR KEYSTROKE
130 REM
140 REM CALCULATE 2 BYTE BINARY NUMBER FOR TIMER AND
150 REM LOAD THEM INTO TIMER
160 REM
170 OUT 6,X-(INT(X/256)*256) : THE FIRST NUMBER BASED ON "X"
180 OUT 6,INT(X/256)
190 REM
200 REM NOW OUTPUT THE TEST TEXT
210 REM
220 LPRINT "HELLO SPORTS FANS. THE CURRENT BAUD RATE IS ";A$;".";CHR$(7)
230 PRINT "TEXT TO SERIAL PORT NOW RUNNING ";A$;" BAUD."
240 PRINT:PRINT
250 REM
260 REM NOW REPEAT THE RUN AT A NEW SPEED
270 REM
280 NEXT R
290 REM
300 REM
310 PRINT "END OF PROGRAM REACHED"
320 GOTO 430
330 REM THIS IS THE DATA BASE FOR BAUD RATES
340 REM
350 DATA 9600, 13 : REM 9600 BAUD
```

```

360 DATA 4800, 26 : REM 4800 BAUD
370 DATA 2400, 52 : REM 2400 BAUD
380 DATA 1200, 104 : REM 1200 BAUD
390 DATA 300, 416 : REM 300 BAUD
400 DATA 150, 832 : REM 150 BAUD
410 DATA 110, 1134 : REM 110 BAUD
420 REM
430 REM SET SERIAL PORT TO 1200 BAUD DEFAULT SETTING
440 OUT 6, 104
450 OUT 6, 0
460 REM SET PARALLEL PORT TO PRINTER PORT. OMIT IF APPROPRIATE
470 REM
480 POKE 3, 129
490 END

```

Reading the Date and Time from MBASIC

Although no MBASIC commands are available to read the date and time from the QX-10 internal calendar-clock, it is possible for a small subprogram to supply this function. (These functions both work on Rising Star BASIC.)

The QX-10 uses a 46818 CMOS calendar-clock integrated circuit to provide both a calendar and clock function. This chip is powered by the normal power-supply regulated voltage when the QX-10 is in use. It is maintained during off periods by an internal rechargeable NiCad battery.

The 46818 is mapped on the I/O data bus at locations 61 and 62 (3D and 3C in hexadecimal notation). It provides 14 bytes of data and status and 50 bytes of additional general-purpose random access memory, which is also battery powered while the QX-10 is turned off.

Operation of this chip is as follows: the address of the date-time register (or memory location inside the 46818) to be referenced is first stored at I/O address 61. (In BASIC, use the OUT command.) The data contained in the register or memory location can then be read or written at I/O location 62 using BASIC's INP or OUT commands. Table 7-3 lists the values contained in each of the 14 registers of the 46818.

The program shown in Listing 7-4 shows how to read the date and time from the 46818. In practice, an MBASIC program should first read the seconds and then everything else needed, then perform a second read of the seconds. If a calendar-clock update were made during the time BASIC took to read the chip, the seconds will not compare and a second set of readings should be made; otherwise, all data will be good.

It should be noted that VALDOCS uses many of the memory locations inside the clock chip; tampering with them may cause VALDOCS to change the system SETUP last used when running VALDOCS.

BASIC ON THE QX-10 113
Table 7-3. 46818 CALENDAR-CLOCK ADDRESSING

I/O Location	Register Content
61	Register Address (Write Only)
62	Register Data (Read-Write)
On Chip Addresses:	
0	Second Data
1	Alarm Second Data
2	Minute Data
3	Alarm Minute Data
4	Hour Data
5	Alarm Hour Data
6	Day of Week
7	Day of Month
8	Month
9	Year
10	Status Byte (floppy motor timer, reference clock, update status; do not tamper with this location)
11	Status Byte (interrupt enable, mode settings)
12	Interrupt Status Byte
13	Memory Valid Status
14-63	Random Access Memory

Listing 7-4. Reading the Date and Time

```

10 REM PROGRAM TO READ THE DATE AND TIME
20 REM FIRST READ OUT ALL REGISTERS INTO AN ARRAY
30 REM TO MAKE THEM EASY TO GET AT
40 REM
50 DIM A(9)
60 FOR N=0 TO 9
70 OUT 61,N : REM OUTPUT THE CHIP ADDRESS TO I/O 61
80 A(N) = INP(62) : REM INPUT THE CHIP DATA INTO ARRAY
90 NEXT N
100 REM
110 REM NOW CHECK SECONDS DATA
120 REM
130 OUT 61,0 : REM SELECT SECONDS
140 IF INP(62)<> A(0) THEN 60 : REM DO IT AGAIN IF NO MATCH
150 REM
160 REM SECONDS MATCH, DATA IS GOOD
170 REM NOW PRINT DATE AND TIME
180 REM

```

```

190 PRINT "THE TIME IS: ";
200 REM
210 A=A(4):GOSUB 1000
220 PRINT " ";
230 A=A(2):GOSUB 1000
240 PRINT " ";
250 A=A(0):GOSUB 1000
260 PRINT
270 REM
280 REM THE DATE CAN BE PRINTED WITH A SIMILAR ROUTINE
290 REM
300 END
1000 REM SUBROUTINE TO PRINT THE TIME IN VARIABLE A
1010 B=INT(A/16) : REM CONVERT TO DECIMAL FIRST
1020 A=A - (B*6)
1030 A$=STR$(A)
1035 REM THEN PRINT LEADING ZERO FOR TWO DIGIT NUMBER
1040 IF A$=2 THEN A$="0"+RIGHT$(A$,1)
1050 PRINT RIGHT$(A$,2);
1060 RETURN

```

The QX-10 Speaker

The QX-10 uses a small loudspeaker for the audible alert or "bell." The speaker can be easily programmed in BASIC for frequency and duration; volume is controlled by the manual control on the back apron of the main electronics cabinet. Rising Star BASIC provides commands to directly control the bell pitch and duration.

Two of the six available QX-10 interval timers are used to control the duration and frequency of the bell tone. The actual chip used is a NEC 8253. Each 8253 provides three separate counters. Two are used for the speaker-bell, one for the RS-232 baud rate clock, and one for the keyboard baud rate clock. The two remaining timers are available for use by applications programs.

When programming the bell tone and duration, make sure that the correct timer is being set. If the keyboard or serial port timer setting is modified, it will probably be "locked out" and unusable until the system is manually reset or until the QX-10 is powered down. (Serial port operation can be restored by use of the SETUP utility or the BASIC program given earlier to set the baud rate.)

The speaker 8253 timers are I/O mapped with the duration control register at port 3 and the data register at port 0. The speaker frequency-control register is at port 7, and the data register is at port 4.

The 8253 is programmed by selecting the timer to be used (with the control register), then writing a number representing the duration or frequency in the data register. The 8253 uses a 16-bit number for timing, and so two numbers are usually sent to the data register in succession using the OUT instruction. The first number output is the least significant, and the last is the most significant.

The duration timer is operated from a 1200 Hz clock (derived from the keyboard baud rate clock), and is set to about 100 msec (a tenth of a second) when the QX-10 is turned on. The frequency timer runs from a 2 MHz clock, and is set to about 1 kHz during system initialization.

Duration time is calculated by this formula:

$$\text{Time} = N * 1 / 1200$$

Where N is a 16-bit number. (It must be broken into two eight-bit numbers for use by the OUT instruction.)

The speaker frequency is based on this formula:

$$\text{Frequency} = 1 / (N * 0.0000005)$$

Where N is a 16-bit number.

Listing 7-4.

```

10 REM PROGRAM TO CHANGE THE DURATION OF THE
20 REM QX-10 BELL TONE
30 REM
40 INPUT "KINDLY INPUT A NUMBER: ",D
50 IF D < 256 THEN 100 : REM KEEP THE NUMBER SHORT
60 PRINT "KEEP THE NUMBER BELOW 256"
70 GOTO 40
100 REM PROGRAM THE DURATION TIMER, BEEP THE BELL
110 OUT 0,D
120 OUT 0,0 : THE MSB WILL ALWAYS BE 0 IN THIS
130 REM PROGRAM
140 PRINT CHR$(7)
150 GOTO 40 : DO IT AGAIN
160 END

```

```

10 REM PROGRAM TO CHANGE THE FREQUENCY OF THE BELL
20 REM
40 INPUT "INPUT A NUMBER: ",N
50 REM
60 REM CONVERT NUMBER N TO TWO 8 BIT NUMBERS
70 REM FOR USE BY THE OUT STATEMENT

```



```

80 REM
90 A = (D-(INT (N/256)*256) : REM THE LEAST SIGNIFICANT NUMBER
100 B = INT (N/256) : REM THE MOST SIGNIFICANT NUMBER
120 REM
130 REM NOW OUTPUT NEW FREQUENCY AND BEEP THE BELL
140 REM
150 OUT 4,A
160 OUT 4,B
170 PRINT CHR$(7)
180 GOTO 40
190 END

```

8



Peripherals for the QX-10

A "peripheral" is a device or contrivance connected to a computer to make it more useful. Peripheral products include printers, plotters, light pens, modems, and more. This chapter mentions several common peripheral devices and describes in very brief terms how they can be connected to the QX-10 and used.

The author does not specifically endorse the particular products shown here or suggest that they are the most suitable for use with the QX-10 in a particular situation. The intent is simply to demonstrate the ease and flexibility of the QX-10 to support a wide range of computer products.

PRINTERS

A printer is probably the most common peripheral product used on any computer. Printers for the personal computing market come in a wide variety of styles, technologies, and price ranges. Printers can be grouped in two major categories: fully formed character printers and dot matrix printers.

The fully formed character group includes letter-quality printers, such as those made by

A peripheral collage. Most common computer peripherals can be used with the QX-10.



The typing element used in daisy wheel printers.

Qume, Diablo, NEC, and more recently, by Comrex, Brother, and Transtar. These are generally called "daisy wheel" printers since the overall form suggests a daisy flower. The typing element is actually a spoked wheel with letter elements at the end of each spoke.

Operation of the printer is easy to understand. The daisy wheel is spun so the character to be printed is in position. A hammer then strikes the back of the spoke, making an impression through the ribbon and onto the paper. Although these printers are slower and usually more expensive than dot matrix models, the typical output is of typewriter quality.

The throughput or speed of these printers is usually measured in characters per second (cps) and varies from about 12 cps for low cost models (\$700-\$900 price range in 1983) to around 60 cps for the high-performance models that cost \$2000 or more. This compares favorably with the older, less reliable Selectric typewriter-based terminals (13 cps) that were previously used for letter-quality applications.

The lower cost models, such as the units described here, are supplied with either a serial RS-232C or a Centronics parallel interface. The choice of interface may present a problem for some users.

It is usually desirable to have a faster dot matrix printer for non-critical output (usually supplied with a parallel interface) and the daisy wheel printer for high-quality correspondence output. Users who have both types of printers and who will be using electronic mail (which also uses the serial port) will probably choose to connect the high-speed dot matrix printer to the parallel port, the daisy wheel printer to the serial port, and use an expansion interface card (described in Chapter 9) to support electronic mail.

The Transtar 130 Daisy Wheel Printer

The Transtar 130 (T 130) daisy wheel printer is representative of the latest generation of low-cost printers in this category. These printers are physically larger and heavier than the

faster and less expensive dot matrix printers typically used on personal computers. (This printer weighs about 31 pounds compared to 16 pounds for the Epson FX-80 dot matrix printer.)

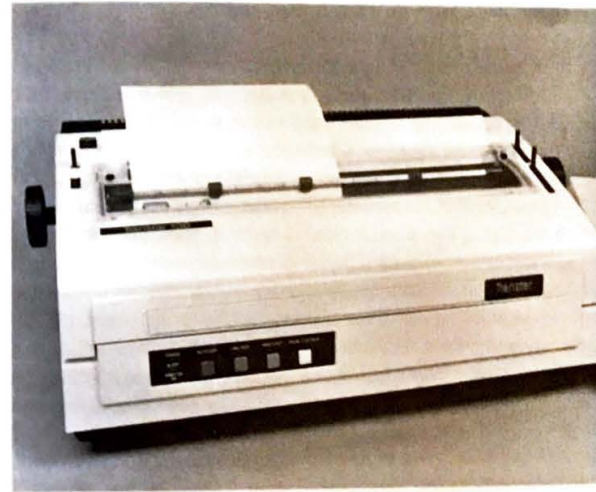
The T 130 offers exceptionally convenient manual single-sheet paper loading. (At this writing no automatic feeder was available for it.) Paper and envelopes are inserted in the back of the platen in the same manner as that on a typewriter. The paper can be positioned by use of the platen knobs on either side of the unit. The left knob can be pushed in (toward the printer) to release the platen for variable adjustment as in a typewriter.

Controls on the front panel can be used to automatically position freshly loaded paper to one of four preset positions, move paper up one line (line feed), eject a printed page, and temporarily stop printing operations.

The print speed of these printers is rated at 18 cps. Except for the lack of a two-color ribbon, the T 130 printers are Diablo 1610/1620 compatible. (Valdocs supports the Diablo 1620 protocol.)

All the standard high-quality printer functions are available on the T 130. Print densities of 10 (pica) or 12 (elite) characters per inch are switch selected, but under program control can be proportionally spaced or compressed up to 15 characters per inch. Super- and subscripting and graphics are supported. Printing can be bidirectional if it is supported by the printer driver software.

These printers use standard Olivetti fabric and single-strike carbon ribbons. Eight different typefaces (fonts) are listed in the manual. These typing elements are available through Transtar, although Silver-Reed EX 55 typewriter wheels are listed as being compatible.



The Transtar T 130 daisy wheel printer.

**Table 8-1. TRANSTAR 130
SERIAL BAUD RATES***

300	2400
600	4800
1200	9600

*These serial baud rates are selectable by switch selection inside the printer.

This printer is sold with either a serial or parallel interface. (Transtar printers support only one type of interface, not both.) Except for the interface, the printers are mechanically identical. The parallel interface is the standard Centronics-compatible parallel interface. The serial RS-232C interface operates at six different standard baud rates (Table 8-1).

Since serial data is normally sent to the printer faster than it can be printed, the printer must be able to tell the computer to stop sending when the 2K byte internal buffer is nearly full. This is done by direct signal line control (the T 130 uses the Data Terminal Ready control line on the RS-232C interface) or the XON/XOFF or ETX/ACK protocol (where the printer sends a standardized control character to the computer to start and stop data transmission).

The manual supplied with T 130 printers is well written and provides good technical descriptions of the interface and how to configure it. Clear programming examples also show how various printing features are activated and used.

The print quality of the T 130 printer is excellent, and there is no smearing; it compares favorably with that of a standard office Selectric typewriter. The printer can print an original and four copies, enough for nearly all office applications.

Dot Matrix Printers

Dot matrix printers were first used in low-performance, low-duty cycle applications, such as ticket printers. First used in the early 1960s, this printing technology has steadily increased in print speed performance, quality, and acceptance. Characters are formed by a series of dots, much like those on a sports scoreboard.

This printing technology is usually divided into impact and non-impact categories. Impact printing uses a print head that drives 7 or more wires into the ribbon and paper, each producing a dot. Characters are formed by selectively firing needles of the head, usually arranged in vertical columns, as the head is moved horizontally across the paper.

Most modern dot matrix printers now print bidirectionally and feature "overlapped" dots, allowing more uniform lines to be printed. (This reduces the "dotty" look of early dot matrix printer output.) Dots are overlapped by staggering the print wires in the head. Although staggering improves the appearance of the characters, it also reduces the overall height from the standard 0.1 inches to about .09 inches.

The current trend in impact dot matrix printer technology is to add more wires, allowing "near typewriter" quality and higher speed. Some now offer color by shifting colored



The print wires of a dot matrix printer head. The wires are typically about .012 inches in diameter. This particular head was used in an old Centronics printer.

ribbons and automatic single sheet paper feeding. At present, none can automatically handle envelopes.

Non-impact printers offer exceptionally quiet operation at the expense of multiple copy capability. Early printers of this category (1970 era) were built into portable terminals such as the Texas Instruments "Silent 700" series.

Thermal, non-impact printers use specially treated paper and a thermal print head. The paper for them is usually available only on rolls, typically has a "funny" feel, often won't lie flat (it has a built-in curl), and sometimes fades or discolors if it is left in the sun. Many considered the early heads relatively unreliable because they often needed replacement after six months to a year of heavy use.

Electrostatic printing, although still using special paper, was quiet and very fast and could produce good graphics copy. These printers were (and still are) used primarily in large computer installations but have no penetration into the personal computer market.

The latest technology to appear in non-impact printing is the so-called "wax printer." It offers exceptional print contrast, brilliant color, low cost, and reliability. The process combines a semiconductor thermal print head that melts dots from a wax-like ribbon onto the paper. The ribbons, which are similar to typewriter carbon ribbons in cost and texture, can be made in nearly any color. Printers of this type were first shown in quantity at the National Computer Conference in 1983.

The disadvantages of non-impact printing are the print quality (the characters tend to be more "dot-matrixish"), the requirement for special papers (and possibly ribbons), and no more than single copy printing capability. The wax printers will no doubt steadily improve the appearance of printed characters. They promise to lower the initial cost of dot matrix printing even further.



The Integral Data Systems P-80 printer. This printer has a semi-automatic sheet feeder (paper is inserted one sheet at a time) and prints from a four color ribbon. An optional automatic sheet feeder is available for this printer.

Integral Data Systems P 80 Printer

This printer is an example of a high-performance dot matrix printer. It offers two fonts (a letter-quality and draft font), 200 character per second printing, graphics, four colors (including black), and automatic single-sheet paper handling.

Other internal features are automatic text justification (it can automatically make the right margins of text square by use of intercharacter spacing adjustments), automatic proportional spacing, a high-quality graphics mode (84 dots per inch both vertically and horizontally), eight character sets (See Table 8-2.), horizontal and vertical tabs, and a variety of head-positioning commands.

The IDS printer also comes in a 132-column version. Except for the carriage width, it is identical in operation and performance to the P 80.

The P 80 offers both an RS-232C and Centronics parallel interface. The serial interface controls data flow with both the DTR and XON/XOFF protocols. A nearly unique feature of the IDS printers is the large internal buffer of over 3000 characters. The printer can take these characters at communication speeds and free the computer for other use while the buffer contents are printed.

Table 8-2. INTEGRAL DATA SYSTEMS P80 CHARACTER SETS

US ASCII	Spanish
English 1	German
English 2	Swedish
French	Norwegian

Special mention should be made about paper handling on the P 80. The printer can take standard perforated stock fed from the back, through the bottom or front of the printer. Single sheet paper can be printed by moving a lever inside the printer to engage pinch rollers.

Single sheets of paper are inserted in the front of the printer, where they are grabbed by the friction rollers and positioned for printing. The paper is ejected at the top of the printer. No extra software support is required to operate the printer in this mode; it automatically waits for paper. Any form length can be used since the printer senses the bottom edge of the paper as it passes through.

An automatic paper feeder is also available for use with the P 80. The printer is placed on top of the feeder. A paper tray that is similar to that on an office copier is filled with standard typewriter paper or letterhead stock. The printer picks up paper from the tray, prints on it, and ejects it at the back when finished, all automatically.

Users who contemplate using this printer (or any non-Epson FX series dot matrix printer for that matter) should remember that certain Valdocs features such as special characters, direct screen copies, and graphics printing will not be possible with the P 80. Furthermore, a special interface cable will probably have to be made since the P 80 does not use a standard Centronics-type connector. But if speed is of primary importance, the P 80 or other "fast" printers will prove perfectly acceptable for use with the QX-10.

Integral Data Systems was acquired by Data Products in the spring of 1983, and is now considered a part of Data Products Special Products Group. They now also produce color printers for OEM accounts for such industry grants as IBM and Burroughs.

CP/M compatible test formatters that take full advantage of the high-performance features of the P 80, even allowing color to be easily imbedded into text, are available. (The Rainbow Writer, Aptec, 80 Townsend St., Pepperell, MA 01463 is one such. The Rainbow Writer accepts files written by standard text processors, such as PeachText, interprets commands for the printer written in plain English, and drives the printer accordingly.)

The Epson FX-80

The FX-80 is an example of a low- to medium-priced impact dot matrix printer that is totally compatible with the QX-10 and Valdocs software. It is a much improved version of the original MX-80 printer first imported into the United States in the early 1980s. The FX-80 features faster printing (now 160 characters per second), down-loadable fonts (new or different character sets can be programmed into the FX-80 from the QX-10), proportional spacing, and more.

Users of the QX-10 should be aware that the MX and RX series of printers are not strictly compatible with many Valdocs features. Some of them are not available in Valdocs version 1 but are coming in versions 2 and 3. Financial considerations aside, the FX-80 or FX-100 (the "wide-body" version of the FX-80) are better suited for use with the QX-10 and Valdocs than the MX-80/MX-100 printers.

Compared to the MX-80, the FX-80 is far more flexible and much faster. It features roman and italic print fonts, six print pitches, 16 print mode combinations, proportional



The Epson FX-80 printer. It and the wide-body version, the FX-100, are the only two printers fully compatible with all Valdocs features including graphics and multiple character sets.

spacing, super- and subscripting, underscoring, down-loadable character fonts, six different high-resolution graphics densities, and more.

Downloaded character sets are those that have been designed or are stored on a computer and sent to the printer when needed. Valdocs uses these to provide many of the new character sets coming in later versions. Printers without this function will not be able to print documents prepared with the Valdocs editor when special features (e.g., block graphics to "box" special text passages or new character sets) are used.

Block graphics are not available on the FX-80 unless downloaded as special characters. About 2K bytes of RAM is provided in the FX-80 to store the downloaded characters, but can also be used to provide a print buffer, much the same as mentioned for the P 80, when no external character sets have been loaded.

The FX-80 can also take single sheets of cut paper although loading is sometimes a little difficult. Care must be used, especially with standard perforated stock since the feed throat is restricted. It may be necessary to take the cover off the printer to clear jams caused by torn scraps of paper.

The FX-80 is a good choice of printer for general applications on the QX-10 because it is both supported by Valdocs, and "upward compatible" with the MX-80. CP/M programs that use an MX-80 will be able to use the FX-80 as before, but with faster printing and quieter operation.

The Epson FX-100 is basically a 132-column version of the FX-80. Except for the wider body to accommodate forms in the 13.5 inch range, all features are the same for both printers.

MODEMS FOR THE QX-10

The term "modem" is from modulator-demodulator. This equipment is required for computer communications because of the limitations of the voice telephone network. Voice telephone lines are designed to use analog signals. (In this case, analog means a variable,

PERIPHERALS FOR THE QX-10 125
constantly changing signal in the frequency range of 300 to 2500 Hz. This is the frequency range where most intelligibility in human speech is carried.)

Although the output of computers varies, the frequency range and waveform—the manner in which it varies—are not at all suited for direct use on standard telephone lines. The solution for low data rate applications is to use the computer output to change the frequency of a tone. This is done by circuits called "modulators"; at the far end of the telephone line a "demodulator" recovers the original computer-type signal. Tones can be sent over standard telephone lines with ease.

Several types of modulation schemes are used depending on the data rate to be sent over the telephone system. If the rate is 30 characters per second or less, a simple oscillator can be modulated directly. Modems of this type are usually said to be "Bell 103 or 113A compatible," since particular tone frequencies were standardized by Dataphones of these models sold by Bell Telephone.

The Bell 103 standard makes use of two standard tone pairs, which are listed in Table 8-3. Two types of modems are used with this standard. The "originate" modem transmits one set of tones while listening on another. The "answer" modem listens to the set of tones transmitted by the originate modem, and sends on the other set. The terms "originate" and "answer" are historical in nature. When Bell Telephone had the direct connect monopoly, it provided all the "answer" modems, those that could answer a telephone ring. Then, all acoustical couplers were used on the "originate" side (where a human originated the call by dialing) of the communication.

Some acoustical couplers provide an originate-answer switch to select which set of tones will be sent. Operationally it makes no difference which of a pair of modems is set to originate as long as one "originates" and the other "answers".

Data rates up to 120 characters per second are handled by specialized, more complicated modems using either the Vadic 3400 or Bell 212A standard. Neither of these standards are compatible with each other, and, typically, 300 baud (30 character per second) data can not be reliably transmitted on these modems since they are specifically designed for the



The legendary MX-80 printer from Epson. This printer can be used with Valdocs, but does not have down-loadable fonts or proportional spacing available in the Valdocs editor.

Table 8-3. BELL 103 TONE PAIRS*

Originate	Answer
Send Tones	
1070	2025
1270	2225
Receive Tones	
2025	1070
2225	1270

*These tones are in Hz; the choice of tone sent is based on the RS-232C voltage level output by the computer. (A positive voltage usually represents a "0"; a negative voltage usually represents a "1.")

higher data rate and use sophisticated internal data processing techniques that are optimized for it. Low-speed data is often sent by use of a standard low-speed modem that is built into the system and is automatically selected when needed.

Modems can be connected to a telephone line in the United States directly or acoustically by individual users who purchase them on the open market. In parts of Europe such connections are illegal unless the equipment is provided by a government agency. (Bell Telephone once held the same sort of monopoly until the "Carterphone" case was brought to trial—it allowed private acoustical couplers to be used—and in the late 1970s when direct connection to the telephone network through FCC-approved interfaces was finally permitted.)

Acoustically coupled modems are almost exclusively limited to 30-character-per-second operation because of the wide variances in carbon microphones used in telephone instruments. Acoustical couplers simply use a small loudspeaker to transmit the tones into the telephone mouthpiece and a microphone to listen to the tones coming from the earpiece. Operation is usually not as reliable with these types of modems, but they are portable (some are even battery operated) and are the least expensive types available, often below \$150.

Direct connect modems, as the name suggests, are connected directly to the telephone line, usually through a telephone "modular" jack. Often these modems are supplied with two jacks in back, one to be connected to the telephone wall outlet through a cable supplied, and the other for the normal telephone connection. (Modular jacks and other telephone accessories are commonly available through Radio Shack and similar electronic supply stores.)

The QX-10 hardware can be used with nearly any modem at any speed, but Valdocs currently supports only the Comrex internal direct modem (described in Chapter 9) and the DC Hayes "Smartmodem" series. Both of these modems provide automatic answer and dialing (both touch-tone and pulse), and when used with the Valdocs electronic mail system, provide reliable, automatic transmission of data files around the world.



The Hayes Smartmodem. This modem is available in both 300-baud and 300/1200 baud models. This and the Comrex CR-103 internal modem (see Chapter 9) are the only modems fully supported by Valdocs version 1.

Hayes Smartmodem

The Smartmodem manufactured and sold by DC Hayes is available in both 300 baud or 1200 baud models. The 1200 baud model incorporates a 300 baud modem that is automatically switched on when the slower rate is needed.

These modems connect directly to the QX-10 serial port using only the RS-232C connections shown in Table 8-4. They are controlled and communicate to the QX-10 through an assortment of commands that are automatically sent by the electronic mail system in Valdocs.

Before these modems can be used with Valdocs, several internal switch settings must be made. The switches are located behind the plastic front cover, and the settings are described by the SETUP program in Valdocs. Once set, the modem can be located in any convenient position since it does not require any attention during operation.

Table 8-4.
RS-232C CONNECTIONS TO HAYES SMARTMODEMS*

Modem	QX-10
2 Transmit Data	2 Receive Data
3 Receive Data	3 Send Data
7 Ground	7 Ground

*Only the above signals need be connected for proper operation with the QX-10 and Hayes Smartmodems.



The Epson CX-20 modem is a lightweight, battery-operated, portable, 300-baud acoustically coupled modem. Although usable with the QX-10, it is more likely to find application with the HX-20 computer, also made by Epson.

Epson CX-20 Acoustical Coupler

This small, battery operated acoustical coupler is well suited for portable operation. It features both originate/answer operation and full/half duplex operation. (Full duplex operation, in this case, means that the echo of whatever characters are sent comes from the device or computer at the other end of the telephone line. Half duplex means that it is sent back from the coupler. Most computers operate in the full duplex mode. Characters sent to them through telephone lines are not displayed on the terminal screen unless they are sent back by the computer.)

Operation of the coupler is simple. The coupler is first turned on, and the originate/answer and full/half duplex switches set to the proper positions. The computer telephone number is then dialed. After the phone has been answered and the "beep" from the answering modem is heard, the telephone hand set is placed in the coupler cups with the dial cord extending in the direction indicated by the logo on the CX-20. The "READY" light on the CX-20 will illuminate, indicating that the communications link is established.

Users who become dependent on the CX-20 should make sure to have the charger for the internal NiCad batteries available. (They provide about four hours of operation and take four to six hours to fully charge.) The power light on the CX-10 blinks when about 10 minutes of use remain.

Connection of the CX-20 to the QX-10 requires the same cable connections as used on the Hayes MODEMS listed in Table 8-4.

VOTRAX SPEECH SYNTHESIZERS

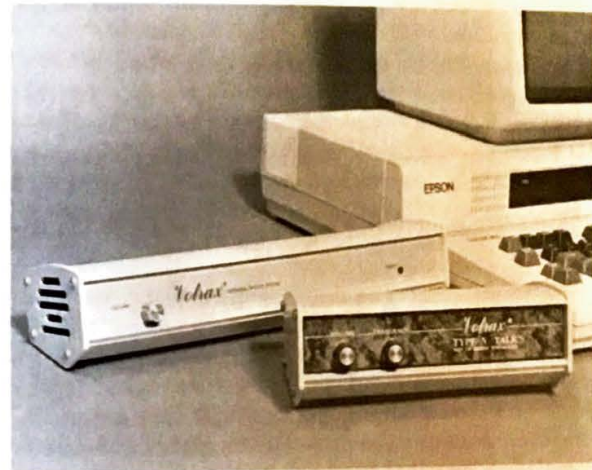
Computers can be used to communicate with humans by use of speech synthesizers. This is a useful form of communication for people who for one reason or another cannot read (such as children or the visually handicapped), when the CRT screen is not visible, or when an audible alert consisting of more than a "beep" is needed.

Two types of speech synthesizers are currently marketed. The first, based on linear predictive coding (LPC), is basically an "electronic recording." It provides high-quality speech, but cannot "say" anything that hasn't already been recorded. The recording process is usually done using a good-sized minicomputer, and requires a lot of heavy-duty number crunching to compress the speech. Playback of the speech requires a dedicated microprocessor chip. LPC speech, once recorded and put into custom-made read only memories, is relatively cheap. (This technique was first used commercially with Texas Instruments "Speak 'N Spell" electronic toy.)

The second type of speech synthesis is called "true" synthesis. A complex sound generator (called a "phoneme generator") is programmed to produce sounds that form recognizable speech. At present, speech from such synthesizers is distorted, and animals such as dogs and cats do not recognize it as a form of human speech without some training.

Phonemes are the basic building blocks of speech. They consist of noises such as "SHHH," "OOO," and "AHH." The English language takes about 64 different phonemes while other languages use more or fewer. True synthesis requires precise sequential control of selected phonemes in order to make good speech.

Although such speech is still very "mechanical" (robotic) in sound and is easily identified



The Votrax speech synthesizers. The Personal Speech System, in the back, offers a wide variety of speech options, including sound effects and an internal alarm clock function that can go off like Big Ben and conclude with a pre-entered verbal message.

as being machine generated, children using computer-aided instruction have been found to do much better when the computer "talks like a robot" rather than like a human. Adults don't seem to object to it, but generally would prefer a "real" voice, given the choice.

Still, true synthesis offers a number of advantages over LPC. It is versatile, and can "speak" in nearly any language, and while sometimes not as intelligible as LPC, it is easily understood once the listener has had a chance to become accustomed to it. Additionally, interesting and varied sound effects can be programmed, including chimes, jet planes, explosions, and "ray guns."

Votrax, who has pioneered true synthesis, manufactures two low-cost speech synthesizers, both usable on the QX-10. (Votrax also makes a full line of "professional" speech systems for industrial applications.) The first, the "Type 'N' Talk" has been available for nearly two years. It is a "straight" synthesizer that can be sent ordinary English text that can be "spoken" with about a 60% accuracy.

The second synthesizer, the Personal Speech System, is an advanced model providing operating features such as an alarm clock, music, sound effects, and additional control of speech amplitude and inflection. Additional hardware features not found in Type 'N' Talk include a large buffer (about 3500 characters of speech can be loaded at one time), a built-in loud speaker, and both serial and parallel interfaces.

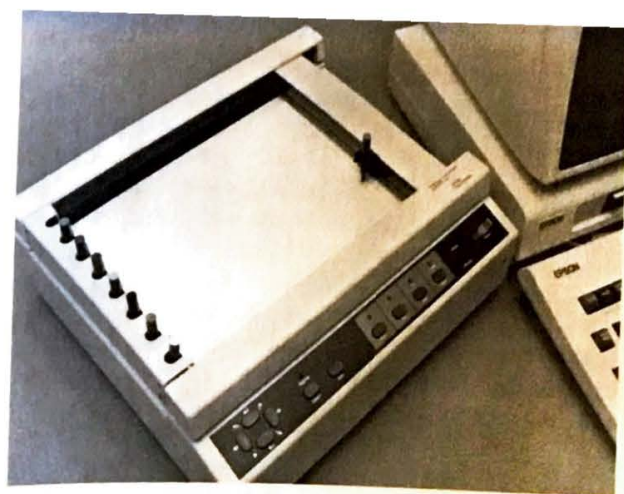
The Personal Speech System features a 90% text to speech accuracy. Accuracy in speech synthesizers means "how well does it pronounce what it was sent?" Both Type 'N' Talk and the Personal Speech System use a "synthesis by rule" technique to translate English text into speech. Since the English language is riddled with exceptions, pronunciation is bound to fail occasionally.

For example, the Type 'N' Talk will pronounce "machine" as "ma-chine". If the word "mosheen" is used instead, proper pronunciation results. The Personal Speech System is much more accurate than Type 'N' Talk, but still needs correction once in a while.

Either of these two synthesizers can be effectively used on the QX-10 through the serial port. Probably the most common use will be under control of a BASIC program where output is switched from the QX-10 printer to the serial port, using the techniques described in Chapter 7.

Graphic Plotters

A graphic plotter is a useful computer peripheral when high-precision drawings are required. Although in many cases the graphics output of dot matrix printers can be used, at present the line quality of a modern plotter cannot be matched with dot matrix printing techniques. The reason is that a plotter draws a continuous line using a pen; dot matrix printers must draw lines with a series of dots. These are smaller, making such graphic output more difficult to read and copy. Further, dot matrix graphic output must make discrete "steps" when a line does not run absolutely horizontally or vertically. (This phenomenon is called "aliasing" and naturally occurs whenever an image is reproduced with a scanning



The IBM Instruments XY/749 Digital Plotter is about the same size as the FX-80 printer. It can automatically draw circles, axes and lines. It "prints" text at about 3 characters per second.

technique (such as television or dot matrix printing; the only way it can be eliminated is to increase the resolution to make steps so fine that they cannot be seen.)

Dot matrix printers can be used to draw shaded images, such as digitized television-like photographs. Digital plotters cannot shade such images (and would have a hard time drawing them because of excessive detail), although areas of a drawing can be filled with a solid color or any cross-hatch pattern that can be plotted. Only few dot matrix printers can provide color printing (such as the IDS printers described earlier), while nearly any graphics plotter can provide color output by simply changing pens.

The IBM Instruments XY/749 Digital Plotter is an example of a modern, high-performance graphics plotter that can easily be used with the Epson QX-10 computer. The plotter can be ordered with a Centronics-compatible parallel interface, an RS-232C serial interface and an IEEE-488 interface. It can be connected to a standard QX-10 through either the printer or serial port with standard cables available at nearly every computer store.

This particular plotter allows automatic computer-controlled selection of one of eight pens held in a rack along the left side of the plotting surface, an 8½-by-11 inch area. Paper is electrostatically held flat to the plotting table. The pen is moved by DC motors coupled with a wire cable/pulley arrangement. Overall, the plotter is about the same size as one of the Epson printers, and power consumption is around 20 watts.

Considerable firmware inside the plotter makes it easy to use. Characters can be automatically "plotted" (at three characters per second) in virtually any size and angle or slant. The plotter can automatically draw circles, axes, several variations of dotted lines, curves, characters, and of course, can select colors. Additionally, it can also digitize drawings. A sight (used in place of a pen) is manually positioned over a point of interest. The coordinates of that point can then be sent to the host computer. Repeatable accuracy of the plotter is .1 mm (with the same pen or digitizing sight).

Like the Votrax Synthesizers described earlier, probably the best way to use this plotter is through the serial port of the QX-10. (Additional serial ports are available on an option card, which is discussed in Chapter 9.)

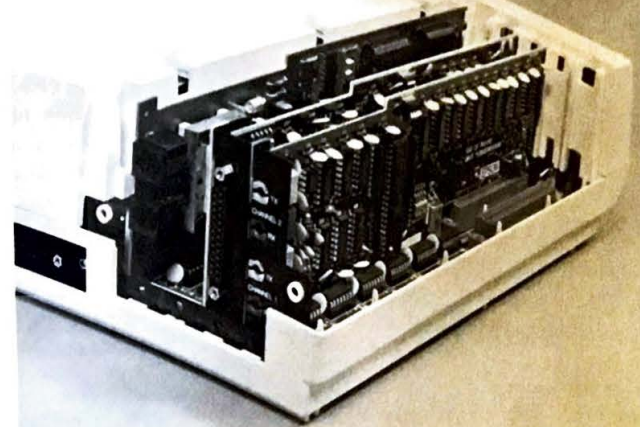
Graphics plotters are "high-end" peripheral, which means that they are specialized and not meant for everyone's use. Prices vary from around \$900 for low-cost, low-performance models to nearly as much as you wish to pay. The IBM unit described here is priced at about \$1800; others in the same league are available from Tektronix, Hewlett-Packard, and others. Generally, price dictates the speed and accuracy of the plot as well as the bed size, and therefore the maximum size of a drawing. Increasingly, software features such as lettering, automatic circles, arcs, dotted lines, and the like are "standard" no-cost frills.

Generally speaking, plotters make "prettier" graphics displays and are easier to use and program than most dot matrix printers. At present there is little commercially available software to drive them; most users write their own plotting software in BASIC, Pascal, C or Forth. They are quiet in operation and fun to watch as they go about their business. Unless you have a specific need, most often related to scientific or research fields or want to make computer-generated overhead projection transparencies, it is unlikely that a plotter is in your future.

OTHER PERIPHERALS

Nearly any imaginable peripheral device can be used with the QX-10, although it is easy to see the limitation of a single parallel and serial port. Another unseen limitation is that of software. Unless a pre-packaged program (probably written for use with CP/M) is available, users of peripheral products not supported directly by Valdocs or Epson will have to write support software themselves.

9



Expansion Cards and Accessories for the QX-10

Several expansion cards for the QX-10 are available directly from Epson and Comrex. Additionally, hard-disk drives for the QX-10 are available from both Comrex and Quality Computer Services. Undoubtedly, as the QX-10 gains acceptance, other manufacturers will offer other expansion and peripheral options.

HARD DISKS FOR THE QX-10

Hard-disk units come with an interface card to be installed in any slot in the QX-10 expansion interface. The hard-disk drives themselves are connected to the QX-10 with a 40 conductor ribbon cable. The Comrex cable is nearly five feet long and the QCS is about three feet long. Both allow good flexibility in drive placement.

The Comrex drive housing is an aluminum enclosure about 8 inches wide, $5\frac{1}{4}$ high, and $14\frac{3}{4}$ deep. The QCS cabinet is 12 inches wide, 5 high, and about $12\frac{1}{2}$ deep. Both require an AC power outlet. The Comrex unit weighs about 12 pounds, and the QCS about the same, depending on disk capacity. Both are painted to match the QX-10.

Several expansion cards installed in the QX-10 accessory bay.



The Comrex ComFiler CR-1510 hard disk.

The Comrex Comfiler CR-1510 has a 10-megabyte capacity; QCS offers capacities ranging from six to 53 megabytes of storage, including models with removable 5 megabyte cartridges.

Both of these units are well constructed and appear to be equal in terms of quality. The QCS unit is a little more "boxey" and takes a wider space on the operating table than the Comrex system, which is long and narrow. The QCS system has been used for software development on the QX-10 by Rising Star, and has a proven track record. The Comrex unit came on the market during the summer of 1983.

These systems both work well with the QX-10, and there is little or no difference in operation between the two. The choice of which to buy should be made on the basis of storage requirements, price, and lastly, cosmetics. Reliability of these units is such that service will probably not be needed for years. QCS offers an extended warranty policy based on a percentage of the cost of the unit, and Comrex charges for time and materials.

Hard-Disk Technical Matters

Hard disks (also called "Winchester disks") offer much greater storage capacities than floppy disk systems. Although the general concept for data storage on all rotating disks is similar to that described for floppy disks earlier, there are several important differences in the construction of hard disks.

Unlike the flexible disk, which is exposed to outside contamination, hard disks use precision plated disks operated in a sealed, clean environment. The recording heads are held off the disk surfaces only a few millionths of an inch, using the air pressure caused by the rotation of the disk. The close proximity of the head to the disk platter allows greater

magnetic densities to be recorded and is the key to the high densities obtainable with hard disks. (The head-to-disk tolerance is so tight that a cigarette smoke particle caught between the disk and head would cause the head to bounce, crashing into the disk surface, likely causing permanent damage to the disk surface.)

When a hard disk system is shut down, the heads "land" on the disk surfaces. Operationally, the system would be better left powered on 24 hours a day to prevent needless landings and the subsequent risk of scratching the disk surfaces.

Hard disks rotate at 3600 revolutions per minute, compared to about 300 for a floppy disk. Data transmitted to a hard disk, then, is over 10 times faster than a floppy, and so file operations using the disks take considerably less time.

Larger capacity hard disks have several disks stacked on top of each other with a space between each platter for the read-write heads. These are selected and switched automatically by the disk controller. Except for the obvious increase in disk capacity, there is no operational difference between them and floppy disks.

File Backup

One of the biggest problems facing the user of hard disks is what to do about backing up valuable files. Although exceptionally reliable, sooner or later the disk system will fail. When such a failure occurs, it is possible that all data on the disk will be lost forever.

Unfortunately, no easy alternative is presently available. Floppy disk backup can be used by simply copying important files to them, but 10 megabytes, the capacity of the Comrex CR-1510, if entirely backed up, would take over 25 QX-10 floppy disks. "Streamer" tape systems are available for minicomputer systems, and may eventually be available for



The QCS DSK-23 20 megabyte hard disk.

mass storage systems on microcomputers, but at present these are too expensive for the personal computing market. (A streamer tape system continuously runs, once started, as data is "streamed" to it. Data is read back in the same fashion, being continuously read and placed on a disk. Capacities of the tape cartridges used are suitable for hard disk backup.)

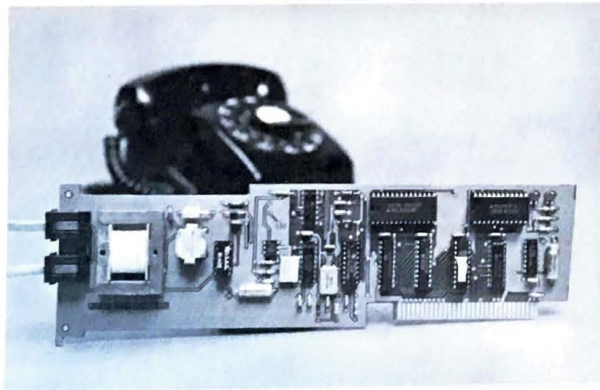
Users at present have one of these backup options: (1) Keep all working data files on hard disk with copies of all (or only important) files on floppy disks; (2) Use a second hard disk system for backup use; (3) Wing it and don't back anything up.

COMREX CR-103 COMMUNICATOR MODEM CARD

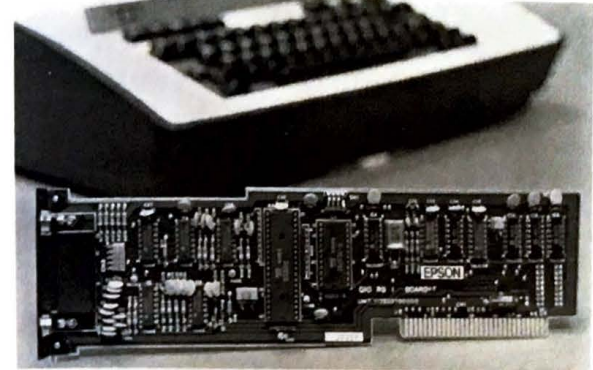
The CR-103 is a single expansion card for use with the QX-10. It is a 300 baud modem card offering both standard rotary pulse and touch tone dialing with M.C.I. and SPRINT capability, automatic answering, and a speaker to monitor the line activity during call initiation.

Bell 103 compatible, this direct connect modem can be configured for either originate or answer.

This option card was designed to provide low-cost, medium-performance telecommunications. It is manufactured and marketed in the United States by Comrex. Users who have known bad or marginal telephone lines may need to use high-performance (and consequently higher priced) units such as the DC Hayes modems described in Chapter 8. This is particularly important since the court-ordered deregulation of AT&T and the subsequent lower reliability of the nation's telephone network. The CR-103 will perform quite adequately for users with "average" telephone lines at a very attractive price. Valdocs fully supports this modem with error-correcting software under the "MAIL" key. The CR-103 comes from Comrex with a support disk for use with CP/M that includes a version of the popular CP/M telecommunications program MODEM.



The Comrex CR-103 Communicator expansion modem for the QX-10.



The Epson serial interface expansion card. It can provide two RS-232C interfaces for common use, or a single RS-232C interface with reverse channel operation.

EPSON SERIAL INTERFACE EXPANSION CARD

This card provides two RS-232C interfaces for use with peripheral equipment. Any baud rate can be software selected on the card, which provides communications clocks independent of the built-in serial port.

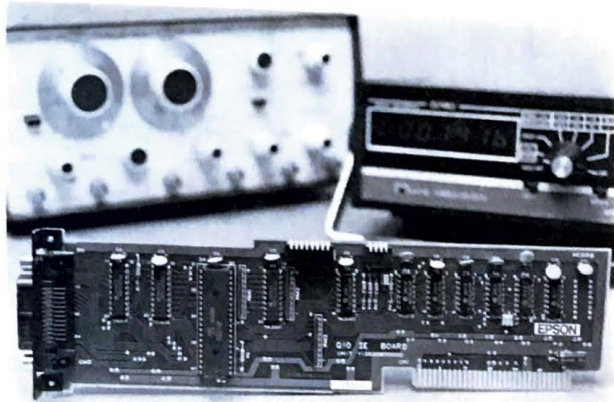
A single "D" connector is provided. Epson has provided a full RS-232C interface including "reverse channel" operation for specialized communications needs. A second, standard RS-232C port can be software defined in place of the reverse channel if desired, using pins 18 and 25 (undefined in the RS-232C specification) on the D connector for Data Set Ready and Data Terminal Ready, respectively.

Both this card (and the internal, standard serial port) can support standard asynchronous communications as well as Bi-sync and SDLC synchronous signaling.

EPSON IEEE 488 EXPANSION INTERFACE CARD

The IEEE 488 bus system is used in industry and scientific applications to allow computers to control and monitor various combinations of laboratory and industrial test equipment. The IEEE interface is complex and not universally found in personal computing systems, but is supported by all major manufacturers of so-called "smart" laboratory instruments and most scientific minicomputers.

This card supports full source and acceptor handshaking, talker, listener, service request, and remote/local functions. Additionally it provides parallel pole 1 and 2 message configurations, device clear, and trigger and controller functions. The NEC uPD7210 integrated circuit is used to provide the controller intelligence.



The Epson IEEE-488 interface card is most often used in laboratory and industrial applications.

EPSON OPTICAL FIBER EXPANSION INTERFACE CARD

This card is used in applications requiring high data rates and/or absolute electrical isolation between the QX-10 and an instrumentation area. The card is connected to peripheral devices using a standard Hewlett Packard HFBR-0500 Fiber Optic Link system.

Data is transmitted and received at 500,000 bits per second using the standard bi-synchronous SDLC communications protocol with Manchester coding. The card itself can be installed in any expansion slot except slot 5. Because of the high data rate, direct memory access (DMA) is required for the use of this card.

This expansion card is intended primarily for use in scientific, research, and local area network business and industrial applications. At present there is no published software for support of this card in any of these applications, although Epson is working on a local area network system.

OTHER EXPANSION CARDS

Epson, Comrex, and other independent companies are developing additional QX-10 expansion interface cards that will further increase the flexibility of the QX-10 computer. Two of these are a RAM-DISK card which allows large sections of disk data to be held in a special memory treated like a disk (providing an enormous increase in apparent disk access time), and a universal floppy disk controller board. (It allows virtually any floppy disk drive to be used with the QX-10.)



The Epson Fiber-Optic interface card.

Other cards and systems were under development at the time of this writing. Given the obvious scientific and industrial thrust of the first QX-10 expansion cards and the business and personal computing applications of CP/M and Valdocs, it would appear that the QX-10 will become a "universal" computer useful in almost every walk in life.

EPILOG

Given the QX-10 computer, which by December 31, 1983, had sold over 50,000 units in the United States since its introduction in May of that year, what can be expected of Epson in the coming months and years?

One guess would be that in keeping consistent with the **apparent** Epson trend of marketing a full line of printers, the same would be expected of its computers. At the end of 1983 Epson was selling the HX-20 "notebook" computer, a portable that can easily be carried in a briefcase, and the QX-10.

The HX-20's competition is presently Radio Shack's Model 100 and another similar unit, the NEC PC-8201. The most obvious difference between the HX-20 and these two computers is the HX-20's built-in printer and cassette tape system (the other two offer neither) and the screen size. The HX-20's screen displays only four lines of 20 characters each compared to the larger screens (six lines of 40 characters) of the other two systems.

It would not be unreasonable to expect an expanded version of the HX-20 in the near future with direct "hooks" to transfer data between it and the QX-10, or perhaps a capability to run CP/M and Valdocs.

The QX-10 design could be simplified with the use of **custom integrated** circuits, which would improve the hardware performance and reduce the cost of the system at the same time. Valdocs makes extensive use of the disk system and **needs** every bit of the 256K bytes of memory provided in the QX-10. Perhaps we will see another "QX-10" type of computer of about the same capacity as the present system, but at lower cost and perhaps smaller size.

A dual processor computer system able to run both CP/M and MS-DOS (IBM) applications could be another possibility. A "high-end" 16-bit system could be under development that will more fully support Valdocs system requirements. If guesses were to be made, a system based on the Intel 80186 microprocessor with perhaps a megabyte or two of onboard memory and at least one and perhaps **two internal hard disk drives** would be a reasonable start.

As to software development, Epson has made a tremendous investment in **Valdocs**, and given the size of the undertaking, it has been a success thus far. Epson, like all the corporations owned by the Hattori Trading Company, makes a living by selling products made by mass production. The object is to make products that can be sold by the millions. If any computer system is to be sold in numbers that large, it must be simple enough for everyone to master.

Valdocs and the QX-10 promise to make it so.

Appendixes

APPENDIX A EPSON PRINTER EP-101 SPECIFICATIONS

This was the first printer made by Epson, used by Seiko in timers made for the 1964 Tokyo Olympic Games, and gave Epson its American name (meaning "Son of EP"). Descendants of the EP-101 are still in production.

Model EP-101

Mechanism: Fully formed characters, impact printing
Print Rate: 2.8 lines per second
Number of Characters per Column: 16
Character Size: .11 inches high by .07 wide
Line Spacing: .2 inches
Paper Feed: Solenoid operated
Size: 5.3 inches high by 6.45 wide by 4 deep
Power: 17 volts DC, 410 ma (maximum during printing)

APPENDIX B BLOCK GRAPHICS KEYBOARD MAPPING ON THE HASCI KEYBOARD

These graphics characters are generated by holding down the GRPH key or the GRPH and shift keys while typing the character in the NORMAL column.

Valdocs version 1 cannot print block graphics characters displayed on the screen. Versions 2 and above will allow these characters and other graphics data to be intermixed both on the screen and on printed text output.

NORMAL	GRAPH SHIFT	SHIFTED GRAPH SHIFT	NORMAL	GRAPH SHIFT	SHIFTED GRAPH SHIFT	NORMAL	GRAPH SHIFT	SHIFTED GRAPH SHIFT
±	⊠	♥	A	■	□	Q	π	∞
1	Ⓐ	♦	B	§	⌚	R	≥	»
2	Ⓜ	♣	C	Ⓢ	+	S	≈	≡
3	Ⓝ	♠	D	÷	┌	T	■	▣
4	λ	△	E	≤	◀	U	┐	▩
5	μ	⊙	F	✓	↻	V	Ⓜ	Ⓞ
6	Ⓡ	Ⓡ	G	■	Ⓜ	W	Ⓞ	∑
7	Ⓡ	Ⓡ	H	■	Ⓜ	X	Ⓞ	Ⓞ
8	Ⓡ	Ⓡ	I	Ⓡ	Ⓡ	Y	■	■
9	Ⓡ	Ⓡ	J	Ⓡ	Ⓡ	Z	■	○
0	Ⓡ	Ⓡ	K	Ⓡ	Ⓡ	.	→	↓
-	Ⓡ	Ⓡ	L	Ⓡ	Ⓡ	,	←	↑
=	Ⓡ	Ⓡ	M	Ⓡ	Ⓡ	/	↙	↘
1/2	Ⓡ	Ⓡ	N	Ⓡ	Ⓡ	;	↘	↙
<	Ⓡ	Ⓡ	O	+	Ⓡ	'	↙	↘
>	Ⓡ	Ⓡ	P	+	Ⓡ	\	↘	↙

APPENDIX C EPSON DISTRIBUTORS IN THE UNITED STATES

Depot level maintenance, technical support, and information is available through Epson distributors.

New York Area
 Computronics Inc.
 200 Benton St.
 Stratford, Connecticut 06497
 (203) 377-7171

Boston Area
 C & C Computer Concepts Inc.
 65 Bay State Drive
 Braintree, Massachusetts 02184
 (617) 843-7932

Philadelphia Area
 Mid-Atlantic Computer Products, Inc.
 219 Rittenhouse Circle
 Keystone Park
 Bristol, Pennsylvania 19007
 (215) 788-3499

Chicago Area
 J B N Distributing Co.
 106 Garlich Drive
 Elk Grove, Illinois 60007
 (312) 952-8222

Detroit Area
 Microprocess Computer Systems, Inc.
 13500 Reeck Road
 Southgate, Michigan 48195
 (313) 281-2800

Minneapolis Area
 Great Northern Data Systems, Inc.
 3050 Ranchview Lane
 Minneapolis, Minnesota 55441
 (612) 559-0992

St. Louis Area
 Delta Computer Systems Dist., Inc.
 2666 Metro Boulevard
 Maryland Heights, Missouri 63043
 (314) 291-6300

Atlanta Area
 Pre-empt Computer Systems Dist., Inc.
 5600 Oakbrook Parkway
 Norcross, Georgia 30093
 (404) 447-1414

Dallas Area
 Southwest Tele-Computing Corp.
 6000 Campus Circle
 Irving, Texas 75062
 (214) 659-0353

Los Angeles Area
 Amber West Corp.
 1000 Ortega Way
 Placentia, California 92670
 (714) 632-9560
 (213) 320-4858

San Francisco Area
 M B C Marketing Corp.
 385 Oyster Point Blvd.
 So. San Francisco, California 94080
 (415) 589-4950

Denver Area
 Western States Computer Distributing Co.
 5867 N. Broadway
 Denver, Colorado 80216
 (303) 296-9377
 (800) 221-0119 (Colorado Only)

APPENDIX D QX-10 INTERRUPT PRIORITIES

Highest	Powerdown Detection
^	Software Timer #1
^^	External (option) INTF 1
^^^	External (option) INTF 2
:	Keyboard/Internal RS-232C Port
:	CRT/Light Pen
:	Floppy Disk Controller
:	Parallel Printer Interface
:	Accessory Slot 1
:	Calendar-Clock
:	Accessory Slot 2
:	Accessory Slot 3
:	Software Timer #2
:	Accessory Slot 4
Lowest	Accessory Slot 5

APPENDIX E SOME COMMONLY USED ADDRESSES IN THE QX-10

(All addresses in hexadecimal notation)

CAUTION: Do not tamper with any I/O or Memory location unless you are sure of what you are doing. Physical damage can occur in the QX-10 monitor if the CRT display controller is misprogrammed. Details of the control register operation of the various timers, interrupt handlers, and controllers can be found in the QX-10 Technical Manual and integrated circuit handbooks published by NEC and other semiconductor manufacturers.

SELECTED MEMORY LOCATIONS

Memory Address	Function
0000	Jump to LIOS/BDOS to Start Operating system
0003	I/O byte used to direct input/output on system
0100	The starting location of applications programs
I/O Address	
00	Loudspeaker Timer
01	Timer #1—not assigned in QX-10 system
02	Timer #2—not assigned in QX-10 system
03	Command Register (8253) (Timers at location 00, 01, 02)
04	Loudspeaker Frequency
05	Keyboard Clock
06	RS-232C Serial Port Clock
07	Command Register (8253) (Timers at location 04, 05, 06)
08	Master (8259) Interrupt Controller
0C	Slave (8259) Interrupt Controller
10	Keyboard Data
11	RS-232C Data
12	Keyboard Command Register (7201)
13	RS-232C Command Register (7201)
14	Printer Data
15	Printer Status
16	Printer Control
17	Command Register (8255) for Printer
18	DIP Switches (on back apron)
30-37	Floppy disk addresses — stay clear
38-3B	Screen control addresses — stay clear
3C	Clock Data
3D	Clock Address

APPENDIX F SOURCES OF EPSON INFORMATION

Epson Computing Group
400-2 DeYoung
Marion, Illinois 62959
Telephone : (618) 993-3600
Bulletin Board: (618) 997-3220

Epson Connection
PO Box 14207
Detroit, MI 48214
Telephone: (313) 822-0090

Epson Corporation
3415 Kashiwa Street
Torrance, California 90505
Main Telephone: (213) 539-9140
Valdocs Hotline: (213) 603-0819
(800) 556-6721

Rising Star Industries, Inc.
24050 Madison
Suite 113
Torrance, California 90505
(213) 378-9861

Publication:
The Rising Star
9211 Nagle Ave.
Arleta, California 91331
(213) 378-9861
(Published bi-monthly. Content
about Valdocs, Rising Star and
Epson products and personali-
ties: \$24.00 per year.)

**APPENDIX G MANUFACTURERS AND VENDORS
MENTIONED IN THE TEXT**

Epson America
3415 Kashiwa Street
Torrance, California 90505
(213) 539-9140

Rising Star Industries, Inc.
24050 Madison St.
Torrance, California 90505
(213) 378-9861

Integral Data Systems
Rt. 13
Milford, New Hampshire 03055
(603) 673-9100

Votrax
500 Stephenson Highway
Troy, Michigan 48084
(313) 588-2050

Transtar
Box C-96975
Bellevue, Washington 98009
(206) 454-9250

Comrex Internatiuonal Incorporated
3701 Skypark Drive
Suite 120
Torrance, California 90505
(213) 373-0280

Peachtree Software Incorporated
3445 Peachtree Road, N.E., 8th Floor
Atlanta, Georgia 30326
(404) 239-3000

Quality Computer Services
178 Main Street
Metuchen, N.J. 08840
(201) 548-2135

IBM Instruments, Inc.
Department 74P
P.O. Box 332
Danbury, Connecticut 06810
(203) 243-7054

APPENDIX H TPM II FILE EXTENSION NAMES

VAL	Valdocs Document File
BAK	Backup Document File
MAL	Mail File
NDX	Index File
DAT	Valdocs Data File
TMP	Temporary Valdocs Data File
INI	Files used to initialize system
FNT	Font files (for Tpestyles key)
HLP	Help files (can be deleted if never referenced)
CHN	Valdocs program files
TIM	Time and Date files for Scheduler
SYS	TPM II system files
COM	TPM II command files

Index

Accessories, 133-139
Accessory bay, 29
AC connector, 38
Acoustical couplers. *See* Modems
Addresses, common, 145
Allen, Paul, 104-105
Altair computer, 9, 14, 105
ALTDISK program, 62
Amidon, Roger, 10, 12, 52
Apple IIe, 15
Applications, 25
Applications programs, 67-80
 CP+, 78-79
 electronic spreadsheet, 74-76
 installation of, 68
 integration of, 79-80
 mailing list manager, 76-78
 PeachTree, 69-78
 sources of, 68-69
 spelling proofreader, 72-74
 text editing, 69-72
ASCII keyboard, 16, 18, 19
 editing cluster in, 22, 23
 function groups of, 23, 25
 inside of, 33, 34
 main typing area of, 20
 numeric keypad of, 23, 24
ASM program, 56
Assembly language, 56

Back apron, 28-29
BASIC, 57, 102-116
 applications utilities using, 107-116
 development of, 102-103
 MFBASIC, 105-106
 microsoft, 104-105
 Microsoft BASIC Revision 5.22, 105
 for QX-10, 105-107
 RSI BASIC, 105, 106-107
Batch mode, 102-103
Battery, 32, 37, 38

BAUDSET program, 62
BDOS (Basic Disk Operating System), 54, 55, 58
Bell:
 loudspeaker for, 35
 programming frequency and duration of, 114-116
 volume control for, 28, 114
Bell 103 standard tone pairs, 125, 126
Bell 212A standard, 125
BILD program, 62, 91
BIOS (Basic Input/Output System), 52, 53-54, 55, 58
Block graphics, 89
Block graphics keyboard mapping, 142
Block menu, 95, 96
Booting, 58-59
Brother keyboard, 17, 18
 editing cluster in, 23
 inside of, 34
 main typing area of, 20
 numeric keypad in, 23
Bus rail, 37

Cables:
 keyboard interconnect, 24, 25
 printer, 45-47
Cabling diagram, internal, 36
CALC key, 100
Calendar-clock, 112-114
CCP (Console Command Processor), 54-55, 58
Centronics interface, 42, 43, 45-47, 49
Clocks, 5-6
CLS key, 22, 71
CMOS RAM, 40, 57
Cold boot, 59
COMPARE program, 62
Comrex, 3, 147
Comrex ComFiler CR-1510 hard disk, 134, 135
Comrex CR-103 communicator modem card, 136

Connectors, 28-29, 37, 38, 42
 COPYDISK key, 97
 COPYDISK program, 55, 57, 62, 79
 Counter-timer integrated circuits, 40
 CP/M (Control Program for Microcomputers),
 16, 52, 53-59
 memory allocation and, 57-59
 TPM II vs., 63
 user areas, 55, 60-61
 utility programs for, 55-57
 CP+, 55, 57, 67, 78-79
 CPYD program, 62
 CRC (Cyclic Redundancy Check) number, 65
 CRT display controller, 42
 CRT screen, 30-31, 33-34, 39-40
 CTRL key, 92-93
 CURSOR UNLOCK, 90
 CX-20 acoustical coupler, 128

Daisy wheel printers, 118-120
 Dartmouth College, 102-104
 Data address mark, 65
 Data recovery, 90-91
 Data transmission:
 parallel, 47-48, 49
 serial, 44-45
 Date, 112-114
 DDT program, 56, 62
 DECIMAL TAB key, 89, 92
 DEL key, 22
 Diami Seikosha, 8
 Dictionary, 72, 73-74
 DIP switches, 28, 29
 Direct Memory Access (DMA), 42
 Directory command, 55, 59, 60
 DIR program, 55, 60
 DISK COPY key, 97
 Disk drives, 26, 27, 33, 137
 Disk formats, 64-65
 Disk read error, 65

Disks. *See* Floppy disk; Hard disks
 Display. *See* CRT screen
 DISPLAY command, 106
 Dot matrix printers, 120-121, 130-131
 DRAW key, 87, 100
 D7 program, 62
 DTSS (Dartmouth Time Sharing System), 103
 Dual Inline Package switches. *See* DIP switches
 DUMP program, 56

Editing, text, 69-72
 Editing clusters, 22-23
 Editing commands, 70-71
 EDIT key, 98, 100
 Editor, Valdocs, 86-96
 control key functions, 94
 keyboard of, 91-94
 menus, 89-90, 95-96
 operation, 88-89
 screen display, 87-88
 ED program, 56
 Electronics, main, 35-37
 diagram of, 41
 Electronics board, main, 40-43
 Electronic spreadsheet, 74-76
 Epson America, Inc., 3, 9, 13, 147
 Epson Computing Group, 69
 Epson connection, 69
 Epson Corporation, history of, 3, 8-13
 Epson distributors, 143
 Epson FX-80 printer, 87, 119, 123-124
 Epson FX-100 printer, 87
 Epson IEEE 488 expansion interface card, 137,
 138
 Epson MX-80 printer, 10, 11-12, 124, 125
 Epson optical fiber expansion interface card,
 138, 139
 Epson P-101 printer, 8, 9, 141
 Epson serial interface expansion card, 137
 ERA program, 55, 60

Erase key, 22
 ER command, 59, 60
 Expansion area, 29, 30
 Expansion cards, 136-139

Fan, 38
 Fields, 65
 File backup:
 with hard disks, 135-136
 in Valdocs, 90
 File commands, 79
 File controls, 25, 98-99
 Filename extensions, 55-56
 Filenames, 55-56
 File protection, 61
 FILES program, 62
 Floppy disk, 26, 27, 63-66
 care of, 66
 formats for, 64-65
 Floppy disk control, 42
 Footers, 71
 FORMAT program, 57
 Formatting, 65
 FORTRAN, 102, 103
 FPL (Forms Processing Language), 85
 Function keys, 23, 25
 FX-80 printer, 87, 119, 123-124
 FX-100 printer, 87

Galletti, Carl, 10
 Gates, Bill, 104-105
 GE Mark II Timesharing System, 103
 General computer, 10
 Graphic plotters, 130-132
 Graphics:
 block, 89
 RSI BASIC and, 106-107
 GRPH SHIFT key, 20, 21, 89, 93

Hard disks, 133-136
 HASCI (Human Applications Standard
 Computer Interface), 12, 81
 concept of, 83-84
 Valdocs versions of, 85
 HASCI keyboard, 16-25
 block graphics keyboard mapping on, 142
 design of, 16-18
 editing cluster in, 22-23
 inside of, 34
 main typing area of, 19-21
 numeric keypad in, 22, 24
 Valdocs editor and, 91-94
 Hattori, Kintaro, 5-6
 Hattori Trading Company, 3, 5, 7
 Hayes Smartmodem, 127
 Headers, 71
 HELP key, 97
 HOME key, 22, 71
 HX-20 computer, 13

IBM instruments XY/749 Digital Plotter, 131
 ICP (Interactive Command Processor), 63
 ID (identification) field, 65
 IEEE 488 expansion interface card, 137, 138
 Index files, 90-91
 INDEX key, 98, 99
 INS key, 22
 INSTALL+ program, 57
 Integral Data Systems (IDS) P 80 printer,
 122-123
 Intel 8008, 105
 Intel 8080, 9, 105
 Interfaces, 42, 43-50
 light pen and, 43, 48, 50
 parallel, 45-48, 49
 RS-232C, 28-29, 42, 43-44, 45, 127
 serial, 44-45
 See also Expansion cards
 Interrupt controllers, 40

Interrupt priorities, 144
I/O bus, 43
IOBYTE, 108-109
IOMOD program, 62

Kemeny, John G., 102, 103
Keyboard buffer, 21
Keyboards, 16-25
 block graphics keyboard mapping, 142
 design of, 16-18
 editing clusters in, 22-23
 function groups in, 23, 25
 inside of, 33, 34-35
 interconnect of, 24, 25
 main typing area of, 19-21
 numeric keypads in, 22, 23, 24
 Valdocs editor and, 91-94
Keypads, numeric, 22, 23, 24
KEYSWAP program, 62
Kiewat, Peter, 103
K menu, 95
Kurtz, Thomas E., 102, 103

Languages. *See* BASIC; FORTRAN
Lifeboat Associates, 68
Light pen, interface with, 43, 48, 50
Linear predictive coding (LPC) speech, 129
LINE key, 22, 71, 88, 93
LIOS (Logical Input/Output System), 63
LIST program, 62
LOAD program, 56
Look menu, 95, 96
Loudspeaker, 35
 adjusting volume of, 28, 114
 programming frequency and duration of,
 114-116
LPC speech, 129
LPRINT command, 108-109

Mail address book, 85
Mailing labels, 77
Mailing List Manager, 76-78
Mailing profile, 77
MAIL key, 98, 99
Main electronics case, 26-31
 back apron, 28-29
 disk drives, 26, 27
 inside of, 33, 35-37
 main power switch, 28
 system reset switch, 26-28
Main power switch, 28
Manufacturers' addresses, 147
Margin release key, 92
Mask command, 60, 61
Memory, electronics of, 40
Memory allocation, 57-59
Memory locations, selected, 145
Memory map, 58
MENU key, 89-90, 91, 100
Menus, 89-90, 95-96
MFBASIC, 105-106
Microprocessors:
 early, 9
 for personal computers, 15
 for QX-10, 40
Microsoft BASIC, 104-107
 See also BASIC
Microsoft BASIC Revision 5.22, 105
Miscellaneous menu, 95, 96
MITS, 105
Modems, 124-128
 Epson CX-20 acoustical coupler, 128
 Hayes Smartmodem, 127
MODEM 7 program, 62
Monitor screen. *See* CRT screen
MOVCPM program, 56, 58, 59
MOVETO command, 107
MultiFont character generator card, 106, 107
MX-80 printer, 10, 11-12, 124-125

NiCad battery, 32, 37, 38
Numeric keypads, 22, 23, 24

On-Screen Format menu, 95, 96
Operating systems, 51-66
 CP/M, 52, 53-59, 63
 floppy disk, 63-66
 memory allocation, 57-59
 TPM II, 52-53, 59-63
 Valdocs, 53
OUT command, 107

Page footers, 71
Page headers, 71
Page menu, 95, 96
Parallel data transmission, 47-48, 49
PeachCalc, 74-76
PeachText, 69-72
PeachTree software, 69-78, 147
 electronic spreadsheet, 74-76
 mailing list manager, 76-78
 spelling proofreader, 72-74
 text editing, 69-72
Peripherals, 117-132
 graphic plotters, 130-132
 modems, 124-128
 printers, 117-124, 125
 speech synthesizers, 129-130
Perry, Commodore Matthew Galbraith, 1-4
Personal computers, development of, 14-15
Personal Speech System, 130
Phoneme generator, 129
PIOS (Physical Input/Output System), 63
PIP program, 56, 62
Plotters, graphic, 130-132
Pocket watches, 6
POKE command, 107, 108
P-101 printer, 8, 9, 141
Power supply, 38, 39
Printers, 117-124, 125
 daisy wheel, 118-120
 dot matrix, 120-121, 130-131
 Epson FX-80, 123-124
 Epson P-101, 141
 history of, 8, 9-12

Integral Data Systems P 80, 122-123
 interface with, 42, 43, 45-49
 Transtar 130 daisy wheel, 118-120
 Valdocs editor and, 87
Printing commands, 71-72, 78
PRINT key, 88, 98
Print spooler, 72, 78, 99
PRINT statement, 110
Proofreading, 72-74
Protect command, 60

QCS DSK-23 20 megabyte hard disk, 134, 135
Quick display, 89-90
Quirk menu, 89-90, 95, 96
QX-10:
 features of, 15-16
 history of, 12-13
 technical specifications for, 31

RAM (Random Access Memory), electronics
 of, 40
Rename command, 59, 60
REN program, 55, 60
Repairs, 32
RESAVE command, 107
Resident programs, 55, 59-60
RETRIEVE key, 98, 99
Review commands, 79
Rising Star Industries, 12, 53, 91, 105, 106, 107,
 147
RS-232C interface, 28-29, 42, 43-44, 45, 127
RSI BASIC, 105, 106-107
Rutkowski, Chris, 10, 11, 12, 16, 53, 81, 83

SA command, 59, 60
SAVE program, 55, 60

SCHED key, 100
 Screen dump, 88, 107
 Screen memory, 33
 Scrolling, 94
 Sectors, 65
 Seikosha Company Ltd., 7-8
 Seikosha factory, 5, 6
 Seiko watches, 4, 6, 8
 Selectric keyboard, 20-21
 Semos, Steve, 12
 Serial baud rate control, 110-112
 Serial data transmission, 44-45
 SET-TIME program, 62
 SETUP program, 16, 21, 57, 62, 79, 91, 100, 108
 SHIFT key, 88, 89
 SHIFT LOCK key, 88, 92
 Shinshu Seiki, 8-9, 11, 13
 SIZE key, 88
 Smartmodem, 127
 SP command, 60
 Speaker. *See* Loudspeaker
 "Speak 'N Spell," 129
 Specifications, 31
 Speech synthesizers, 129-130
 Spelling Proofreader, 72-74
 Spreadsheet, electronic, 74-76
 STAT program, 62, 108
 STATUS program, 56
 STOP key, 97, 99
 STORE key, 88, 91, 98
 Streamer tape system, 135-136
 STYLE key, 88, 92
 Suwa Seikosha, 8
 Switches:
 DIP, 28, 29
 main power, 28
 system reset, 26-28
 Switching mode power supply, 38, 39
 SYSGEN program, 56, 62
 System Controls, 25, 97
 System reset switch, 26-28

TAB key, 89
 Tab Set key, 92
 Technical Design Labs (TDL), 10
 Televideo 920 CRT terminal, 68
 Terminal emulation, 68
 Text editing, program for, 69-72
 Thesaurus, 71, 72
 TI command, 60
 Time, 112-114
 "Time Keeper" pocket watch, 6
 TIMESET program, 57, 62
 Time sharing, 103
 TPA (Transient Program Area), 57
 TPM II (Transient Control Manager II), 52-53, 59-63
 CP/M vs., 63
 file extension names, 148
 file protection in, 61
 resident utility programs for, 59-60
 transient utility programs, 62
 user areas, 60-61
 Transient programs, 55-57, 62
 Transtar 130 daisy wheel printer, 118-120
 Transtar 315 Color Graphics Printer, 8
 Treaty of Kanagawa, 2
 True synthesis, 129-130
 TX-80 printer, 9, 10
 TY command, 59, 60
 "Type 'N' Talk," 130
 TYPE program, 55, 60
 TYPESTYLES function key, 25, 88

 UNDO key, 90, 97
 User areas, 55, 60-61
 User command, 60
 USER program, 55, 60-61
 User's Group Newsletter, 69
 Utility programs:
 CP/M, 55-57

resident, 55, 59-60
 for TPM II, 59-60, 62
 transient, 55-57
 Vadic 3400 standard, 125
 Valdocs, 12, 53, 81-101
 applications, 100-101
 data recovery procedures, 90-91
 editor, 86-96
 File controls, 98-99
 HASCI concept in, 85
 HASCI keyboard, 91-94
 operation, 86
 other applications programs systems vs., 80
 System controls, 97
 Valdocs Development Language (VDL), 107
 Vendors' addresses, 147
 Ventilation, 38
 VERS program, 62

Video display controller, 35, 37, 42, 43
 VISICALC, 74
 Votrax speech synthesizers, 129-130

Warm boot, 58-59
 Watches, 4, 5-7, 8, 9
 WIDTH program, 62
 Winchester disks, 133-136
 WORD key, 22, 93
 Word-wrapping, 88

ZDDT program, 62
 ZPIP program, 62

The Epson QX-10 User's Guide will show you how to get the QX-10 to do what you want it to.

The author, a professional hardware-software engineer with a long background in personal computers, thoroughly explores the architecture, capabilities, and uses of the QX-10. He gives you a brief history of the Epson computer and the corporation and personalities that built it, takes you through the machine's internal features, and explains how to use its applications programs and peripherals most easily and effectively.

You'll learn how to

- Use the Valdocs operating system
- Use PeachTree software's text editor, spelling checker, mailing list manager, and spreadsheet programs
- Run BASIC programs on the QX-10
- Make interface cables for the QX-10
- Run any CP/M program on the TPM II operating system

- Evaluate and select printers, modems, speech synthesizers, plotters, and expansion cards

For your convenience, **The Epson QX-10 User's Guide** also includes a complete listing of Epson Distributors and service centers, and over 80 detailed photographs showing virtually every part of the computer. This book's complete, accurate technical information and practical advice make it an invaluable tool for QX-10 owners and users.

James M. Hansen, an independent microcomputer consultant, has been involved with computers and electronics most of his life. He was formerly Telecommunications Manager of the Kiewit Computation Center at Dartmouth College, and in 1976 he established his own corporation, Polytronics, to provide consulting and custom electronics design services. A resident of New Boston, New Hampshire, Hansen has published numerous articles in popular computer magazines.