

Chapter 2:

OVERVIEW OF VALDOCS

The spirit of HASCI lives in Valdocs. The system is designed to be easy-to-use and easy-to-learn, whether you're a beginning user or a programmer interested in interfacing your application to the Valdocs environment. The following chapter provides a brief, general description of the system. Subsequent chapters cover each of the major components in greater detail.

BASIC HARDWARE

The Valdocs system is implemented on the Epson QX-10, which features a Z80A compatible microprocessor, 256 kilobytes of dynamic RAM, two double-sided, double-density 5 1/4-inch floppy disk drives, a 12-inch bit-mapped video display with a resolution of 640 by 400 pixels with 128 kilobytes of dedicated video RAM, and 2 kilobytes of battery-backed CMOS RAM.

The floppy disk drives are labeled A and B: A is the default system drive where (in Valdocs) all utilities, applications, and the operating system are stored; B is reserved for data files.

The standard QX-10 configuration also includes a programmable timer, Centronics-compatible parallel printer interface, RS-232C serial interface, CMOS clock and calendar, and the HASCI keyboard. Five card slots are included for installation of optional cards, such as a hard disk interface, RAM disks, modems, and a color CRT interface. (For a more detailed look at the QX-10 hardware, see Epson's QX-10 Technical Manual, available from your Epson dealer.)

SOFTWARE VERSION

The first chapter of this manual is valid regardless of which version of Valdocs you are using--the principles of HASCI apply to *all* Valdocs software. However, the remainder of the manual applies to Valdocs Version 1.18 *only*, so it's important to know which version of the system you're using.

Each time you boot Valdocs, you'll see a message that tells you which version you're running. You can also run the VERSION utility (included on the Valdocs system disk to check the versions of individual modules). VERSION.COM can be run from MENU. To find out which version of TPM you're running, exit to TPM and type "V".

BANK SWITCHING

The Valdocs system utilizes bank switching. Memory is divided into banks, which can be used independently, allowing the most used Valdocs modules to be resident in memory. This results in noticeable speed increases. Programs used less frequently can be loaded from the floppy disk onto a waiting bank as needed or requested.

Memory is divided into four banks (numbered 0 to 3) of 64 K each. 8K of each bank is reserved for Resident RAM (at the top of the 65K address), leaving only 56 K of usable RAM in each bank. Resident RAM contains the essential parts of the operating system.

The QX-10's standard 256 K of RAM is divided as follows:

	Bank 0	Bank 1	Bank 2	Bank 3
FFFF	-----	-----	-----	-----
E000	-----	-----	-----	-----
	Transient program memory	TPM operating system	Editor	Index
0000	-----	-----	-----	-----

The TPM operating system and the Valdocs system support are resident in Bank 1. (For further details, see Chapters 3 and 4.) The Editor (Chapter 6) and Indexer (Chapter 7) are resident Valdocs applications programs.

Bank 0 (also called the User's Bank) is reserved for transient programs. Any "outside" application programs run on Bank 0.

TPM OPERATING SYSTEM

TPM-II will run on any Z-80 based computer and is completely compatible with CPM. It does, however, have several advantages over CPM, and for this reason was chosen for the Valdocs environment. Whereas CPM was written specifically for the 8080 microprocessor--and for a language that is a subset of the Z-80 language--TPM was written specifically for the Z-80. TPM has greater flexibility, makes use of many functions that CPM cannot, and handles a wider variety of situations.

TPM is divided into two sections: the nucleus and the utilities. The nucleus, loaded when the system is first booted, is composed of I/O, file management, and housekeeping. The utilities, which include zzip and sysgen, are on the system disk.

Further discussion of TPM appears in Chapter 3. (Also see the *TPM-II User's Guide* for a complete and detailed description of TPM.)

VALDOCS SYSTEM SUPPORT

The Valdocs system support programs, which are extensions of TPM specifically tailored for Valdocs, are located on Bank 1. The most important of these programs is the Sysinit module which contains print spoolers and formatters, interrupt handling routines, and various buffers. Other support programs include the graphics drivers and video driver (see Chapters 4 and 5).

VALDOCS APPLICATIONS

HASCI concepts are best demonstrated by the Valdocs applications programs (see Chapter 1). Briefly, Valdocs is a set of integrated application programs--a Calculator, a word-processor (Editor), an Indexer, a calender (SCHED), a graphics program (DRAW), and an electronic mail system (MAIL).

The user "talks" to all applications via the *interaction window* (sometimes called the *menu window*), which must follow certain rules. These rules are detailed below.

RULES FOR MENUS

1. Menus always appear on the same place on the screen--that is, the screen has a "place for menus."
 - A. The menu is presented in the interaction window.
 - B. The uppermost two lines in the interaction window contain the title or instruction. This is left up to the programmer. For example: "Choose a function and enter the new values. Then press Store.", or "Choose a function by its letter or moving the cursor, then press RETURN". Or if it needs a title: "-VALDOCS MENU-", or "-VALDOCS COPYDISK-".
 - C. The choices within the menu should be aesthetically balanced. For example, if there are seven choices,

four will be presented on the left side of the menus and three on the right.

D. (Undo for prior menu) is placed on the bottom-most window frame. Unless, of course, Undo will not return the user to the prior menu!

E. (Undo to return to Editor) is placed on the bottom-most window frame when appropriate.

2. Menus are laid out so that choices can be made by one of two methods:

A. Type the first letter of the first word and press Return.

1. The first letter of the first word is always contained in brackets < >.

B. Move the cursor until it is over the letter and press Return.

1. The cursor is only allowed over a letter-choice. The cursor cannot go outside, above or below the letters. If there is a space between two letters and the user moves the cursor up, then the cursor moves directly to the next letter and skips that space.

3. Organize the menus such that the most common choices occur first in position and potentially destructive choices occur last.

A. If the user selects a potentially destructive option, such as "Delete a file or block", then another menu appears, warning the user what will happen, and again offering the option to delete or not, the cursor resting first on the non-destructive choice.

4. Make sure the wording in the menu is simple and completely understandable to the user. Avoid wording that is ambiguous or cryptic.

5. Limit the number of menu choices to eight, consisting of two columns of at most four selections.

For further details on the application programs and their data files, refer to Chapters 6, 7, and 8.

RESIDENT MODULES

1. INITIAL.INI: Initializing System Size: 2,048 bytes
Language: Assembly

Upon cold start, TPM loads INITIAL.INI at 100 Hex. INITIAL checks for needed modules to load Valdocs and turns control over to LOADUP.CHN. If the modules are all present, an A> will appear upon a cold start.

2. LOADUP.CHN: System Loading Size: 1,024 bytes
Language: Assembly

LOADUP.CHN sets up INDEX on bank #3, VALDOCS on bank #2, and VDRIVER on bank #1. After running SYSINIT, control is passed to the editor.

3. SYSINIT.CHN: Interrupts & Multi-Task Size: 18,048 bytes
Language: Assembly

SYSINIT.CHN is executed by LOADUP, and sets up Interrupt routines, Stop and Help keys, spooling functions, and other multi-tasking functions, such as the clock display.

4. VLOADER.CHN: Graphics Loader Size: 1,536 bytes
Language: Assembly

VLOADER.CHN loads the VDRIVER.CHN program onto Bank #1 and the four fonts contained in FONT.FNT. It redirects the the console output from the character driver contained in the ZAPPLE monitor to the VDRIVER interface, loaded at OFE00H.

5. VDRIVER.CHN: Video Driver Size: 10,240 bytes
Language: Assembly

VDRIVER.CHN is loaded on to bank #1 during initialization. It can be set up to run in either bit-mapped mode or character mode. The Video Driver can be manipulated through the operating system by the use of Call #39 and D-opcode commands (Drawing Operation Codes).

6. TPM.CHN: Video Emulator Size: 4,096 bytes
Language: Assembly

TPM.CHN is used presently for complete Televideo 920 emulation. With a selection made through SETUP.COM, a bit

is set in CMOS RAM, initiating TPM.CHN, which in turn loads the 920 emulation driver. The keyboard is also affected and becomes 920 compatible.

7. MENU.CHN: Applications & TPM Access Size: 8,192 bytes
Language: Assembly

MENU.CHN provides access to TPM, some applications programs within Valdocs (ie: VERSION.COM and SETUP.SYS), and all other application programs that can be run on the QX-10 along with Valdocs.

8. INDX.CHN: File System Size: 52,480 bytes
Language: C

INDX.CHN is loaded onto bank #3 and is resident throughout Valdocs operation. INDEX.CHN provides system hooks for TPM files and Valdocs files. Filenames may have up to 16 words and extension, unlike CPM's 11-character specification.

9. VALDOCS.CHN: Editor Size: 51,986 bytes
EDIT.HLP: Help File for Editor Size: 1,536 bytes
Language: Stoic

VALDOCS.CHN is loaded onto bank #2 of the QX-10 System RAM. It remains resident throughout the entire operation of the Valdocs system. The [CALC]ulator code is structured within the Editor in Stoic.

10. SCHD.CHN: Appointment Book Size: 51,968 bytes
SCHD.TIM: Data file for SCHD.CHN Size: 384 bytes
Language: C

11. DRAW.CHN: Graphs & Charts Size: 37,888 bytes
Language: Assembly

Valdocs application, loaded dynamically onto bank #0.

12. CPYD.CHN: Copy and Formatting Size 18,432 bytes
Language: Assembly

When initiated, CPYD.CHN is loaded on bank #0. It provides for single file transfers, disk management, diskette formatting, and copying.

13. MAIL.CHN: Telecommunications Size: 29,696 bytes
ADDRESS BOOK.MAL: Address Book Data Size: 7,424 bytes
Language: Assembly

Valdocs application, loaded dynamically onto bank #0.

