CP/M-80 R2.2 for the Epson QX-10 System

External Reference Specification

256K - Version B2.26

CHANGES FROM B2.25 DENOTED BY BOLD FACE TYPE

SECTIONS WHICH HAVE CHANGES ARE SHOWN IN BOLD FACE TYPE IN THE TABLE OF CONTENTS

CONTENTS

1.0 Introduction.....

2.0 Applicable Documents .....

3.0 Feature Description.....

3.1 Device Assignments.....

3.2 Video Display Console.....
3.2.1 Character Driver .....
3.2.2 Graphics Driver .....

3.3 Keyboard Support.....

3.4 Printer Support.....

3.5 Disk Subsystems Support.....
3.5.1 Floppy Disk Subsystem.....
3.5.2 RAM Disk Subsystem.....
3.5.3 Winchester Disk Subsystem.....
3.5.4 Hard Disk Partitioning.....

3.6 Operating System Configuration.....
3.6.1 Cold Boot Loader.....
3.6.2 GENCPM - System Generation.....
3.6.3 OEM Configuration Support.....

3.7 Serial Interface Support.....

4.0 System Level Description.....

4.1 Publications.....

4.2 Equipment Configuration.....

4.3 Interfaces with other Software.....

4.4 Reliability, Availability and Serviceability..

APPENDICES.....

APPENDIX A Memory Map .....

APPENDIX B XBIOS Interface .....

APPENDIX C Graphics Drawing Primatives .....

APPENDIX D System Data Page .....

APPENDIX E Console Control and Escape Sequences ....

E.1 TVI-920 Control and Escape Sequences.....
E.2 QX-10 Control and Escape Sequences.....

APPENDIX F HASCI Keyboard Mapping .....

APPENDIX G Extended Keyboard Support .....

G.1 System Keys.....
G.2 Function Keys.....

APPENDIX H Optional Diskette Formats .....

APPENDIX I Error and Status Messages .....

I.1 Printer Error and Status Messages.....
I.2 Floppy Disk Error and Status Messages.....

APPENDIX J Sample Font Table .....

LIST OF TABLES.....

Table 1-1 Device Correspondence.....
Table 1-2 Permitted Logical to Physical Assignments.....
Table 1-5 Memory Map.....
Table 1-6 Operating System Base Addresses.....

1.0 INTRODUCTION

This document describes the external features and characteristics of an implementation of the Digital Research Inc. CP/M-80 R2.2 operating system for the Epson QX-10 microcomputer. This version of the Specification reflects the 256K version of CP/M-80 R2.2 designated by Epson as release level B2.26.

The operating system product resulting from this implementation includes:

A system data page. The purpose of the data page is to provide specific locations on the system tracks for specific supported features. This eliminates the need to move code around to compensate for system changes. See Appendix G for an illustration of the System Data Page.

A customized Basic Input/Output System (BIOS), integrated with the unmodified CCP and BDOS components of CP/M-80.

A Reference Manual which supplements the Digital Research CP/M-80 documentation and describes to the user the unique features and facilities of this implementation and how it is used.

Source code listings of the customized BIOS and utilities.

A detailed specification of the product (this document) which when used in conjunction with the source listings provides the necessary documentation to support program maintenance.

The requirements used in developing this specification were taken from the CP/M-80 Version 2.2 documentation provided by Digital Research, the QX-10 technical specifications and marketing requirements provided by Epson America, Inc. This implementation is designed to support the various QX-10 system configuration options which include floppy disks, printers, video display, keyboard, memory and serial ports.

The objective of this implementation is to provide application program access to the various hardware facilities of the QX-10 microcomputer system with a user-friendly interface, while providing the functionality of CP/M-80 and its large applications software base to the Epson QX-10 System customer base.

2.0 APPLICABLE DOCUMENTS

The following documents are relevant to this customization:

CP/M Operating System Manual by Digital Research

QX-10 Firmware Specifications - Revision C1 (July 10,1982)

QX-10 System Specification - Revision C5

QX-10 Technical Manual - Principles of Hardware Operation

Operator's Manual Models 912/920 by Televideo Systems Inc.

3.0 FEATURE DESCRIPTION

3.1 DEVICE ASSIGNMENTS

3.1.1 GENERAL DESCRIPTION

The following tables provide the required correlation for the CP/M-80 logical devices, CP/M-80 physical devices and the QX-10 physical devices.

Table 1-1 Device Correspondence

CP/M-80 Physical Device	QX-10 Physical Device
CRT:	Input = Keyboard Output = Video Display
PTR:	Input = Keyboard
PTP:	Output = Video Display
LPT:	Parallel Printer Port
TTY:	Video Display
UC1:	Serial Port
UR1:	Serial Port
UR2:	Serial Port
UP1:	Serial Port
UP2:	Serial Port
UL1:	Serial Port
BAT:	Input = Current RDR Setting Output = Current LST Setting

Table 1-2 Permitted Logical to Physical Assignments

CP/M Logical Device Name		CP/M Physical Devices			
CON:	=	TTY:	CRT:	BAT:	UC1:
RDR:	=	TTY:	PTR:	UR1:	UR2:
PUN:	=	TTY:	PTP:	UP1:	UP2:
LST:	=	TTY:	CRT:	LPT:	UL1:

NOTE: Table 1-2 is established within CP/M-80 itself and is not created or modified by this implementation of CP/M-80

3.1.2 USER/OPERATOR INTERFACES

To display the current assignments in effect, or to make changes in the assignments, the User utilizes the various options contained in the CP/M-80 STAT command. The command STAT DEV: displays the current assignments in effect for the four CP/M-80 logical device names: CON:, RDR:, PUN: and LST:. The command STAT VAL: displays the possible CP/M physical device names that the User can assign to each CP/M logical device name as defined in Table 1-2 above. The SETUP program stores the current value of the IOBYTE in the system data page. This feature allows the User to establish a defined configuration which is restored at each cold boot.

3.1.3 ABORTS AND RECOVERIES

None.

3.1.4 ERRORS

None.

3.1.5 PERFORMANCE

Nominal.

3.1.6 OPERATIONAL CONSIDERATIONS

The QX-10 physical device assignments shown in Table 1-1 are implemented as part of the BIOS customization. The IOBYTE is a reserved byte in memory that keeps track of logical-to-physical device assignments. Display of the possible IOBYTE assignments (Table 1-2) and the current values in effect along with modification of the values in effect is accomplished through the standard CP/M-80 STAT utility which has not been modified for this implementation.

3.3 KEYBOARD SUPPORT

3.3.1 GENERAL DESCRIPTION

At the option of the User either the production or earlier HASCI keyboard is supported as the CP/M-80 R2.2 console input device. Keys such as Escape, Caps Lock, Control, Graph Shift and all typamatic keys are supported by the BIOS console routines. The BIOS maintains a 128 character type-ahead buffer regardless which keyboard is selected.

3.3.2 USER/OPERATOR INTERFACES

Keyboard options, such as typamatic keys or which keyboard is connected to the system, are handled by the configuration utility (SETUP) described in section 3.8. The B Release of CP/M-80 supports two keyboards-- HASCI-1 and HASCI-2. The HASCI-2 keyboard has one function key, EDIT, that is not available on the HASCI-1 keyboard. The keypads of the two keyboards do not correspond. (See keyboard diagrams in the appendix for specific differences.)

3.3.3 ABORTS AND RECOVERIES

None

3.3.4 ERRORS

None

3.3.5 PERFORMANCE

The BIOS manages the 128 character type-ahead buffer feature by running the keyboard in interrupt mode rather than using the normal polling technique.

3.3.6 OPERATIONAL CONSIDERATIONS

The keyboards supported by this implementation produce scan codes which must be mapped into the desired characters. The design of the keyboard mapping is table driven and does not preclude the support for other keyboard types which may be made available during later phases of the QX-10 product cycle.

The translated scan codes produce the full range of 8-bit characters (0-255) enabling generation of graphics characters. The Epson product line (QX-10, HX-20 and printer products) support a variety of graphics character sets. Therefore, flexiblity is provided at system generation time to link the particular keyboard tables which translate the proper video driver-keyboard-printer combination accurately.

The character video driver is only able to produce the QX-10 graphics character set on the console. The ideally suited keyboard tables will support the graphics characters in a manner similar to the unshifted and shifted graphics keyboard mappings as implemented in the Valdocs scheme.

The graphics video driver has the flexiblity of being able to display more than one font on the console. Therefore, the driver can support a variety of graphics character sets generated by the keyboard or supported by a printer. Support of the RX product is acheived on the QX by combining the graphics video driver linked with the RX font, and the keytables which generate the graphics set supported by the RX.

The hexidecimal keyboard mapping which generates the QX font is listed for each keyboard in Appendix A.

When a keyboard is selected via SETUP, the keyboard in effect is displayed as part of the CP/M "sign-on" message at either the next power-on or at the next system reset.

3.4 PRINTER SUPPORT

3.4.1 GENERAL DESCRIPTION

The Epson MX, FX and RX series printers, connected via the parallel interface, are supported by the customized BIOS. Other printers can be connected if they follow the industry standard Centronics compatible parallel interface or the RS-232C serial interface and one of the handshaking protocols supported by the BIOS serial support options described in section 3.7.

In the case where Epson printers are connected to the QX-10 via the parallel port, specific support can be identified in the SETUP utility. Note: this support is limited to proper status byte interpretation. Printer identification through SETUP does not affect the video driver - keyboard mapping configuration.

3.4.2 USER/OPERATOR INTERFACES

Support for the printers is limited to the basic BIOS calls, configuration management and error recovery. Support for the various printer special features is the responsibilty of the application software.

3.4.3 ABORTS AND RECOVERIES

Error reporting and error retry options are made available to Epson printers connected to the parallel port provided that proper printer selection is declared via the SETUP utility. If a non-Epson printer is utilized or the printer is connected to the serial port, aborts and recoveries are the responsibilty of the application software.

3.4.4 ERRORS

Errors are reported to the user on line 25. An explanation of the error messages is contained in Appendix B.

3.4.5 PERFORMANCE

As described in the specific printer model documentation, printers connected to the serial port cannot be supported at speeds greater than 1200 baud without making changes to the jumper options inside the QX-10 itself when hardware handshaking is used. Refer to the QX-10 Technical Manual, Principles of Hardware Operations, page 4-32, Table 4-8 for data relating to the jumpers.

3.4.6 OPERATIONAL CONSIDERATIONS

The assignment of the List Device for CP/M-80 R2.2 is accomplished by the logical device assignment option of STAT as described in section 3.1. The selection of printer options and handshaking methods are handled by SETUP as described in section 3.8.

3.5 DISK SUBSYSTEMS SUPPORT

The BIOS provides support for two standard disk subsystems on the QX-10: Floppy diskette subsystem and a RAM disk subsystem occupying two of the 64K memory banks. Optionally, the BIOS provides support for a winchester disk subsystem.

3.5.1 FLOPPY DISK SUBSYSTEM

3.5.1.1 GENERAL DESCRIPTION

The BIOS supports the two intergal 5 1/4" double-sided disk drives. The standard disk format for this implementation is 512 Byte sectors with 10 sectors per track. Provision is made in the BIOS to read or write (not format) industry standard diskette formats from the RIGHT drive. Utility programs are available which allow the User to reconfigure the RIGHT drive to allow for the importing of application software from other systems.

The BIOS writes all filled or partially filled write buffers to disk in the following situations:

1. Write operation to the directory.
2. Warm Boot

This version of the implementation can support other disk formats as well as the standard QX-10 format described above. The specific data for the QX-10 format and supported optional formats is given in Appendix D.

3.5.1.2 USER/OPERATOR INTERFACES

In order to read or write files from diskettes recorded in a supported optional format, the User must select the the format from the SETUP utility. After the utility has been run, the BIOS can read or write these diskettes only in the RIGHT drive. After reading the desired diskette, and before attempting to read standard QX-10 formatted diskettes, the User must reselect the QX-10 format with the SETUP utility. The QX-10 format is always in effect after a power-on or a system reset. Once the RIGHT drive has been alterred by the SETUP utility it remains in effect through a warm start (CONTROL C).

3.5.1.3 ABORTS AND RECOVERIES

Error reporting and error retry options are available to the User and are selectable from the SETUP utility. These options are as follows:

1. Diagnostic Error Messages
2. Standard Disk Error Retry

The error indications made available to the User during disk error recovery are a function of the status of the Disk Subsystem Options in effect at the time of the anomaly. There are four possible option conditions and the indications for them are summarized below:

1. If "Diagnostic Error Messages" and "Standard Disk Error Retry" are both enabled then the disk error recovery is attempted five (5) times and if the operation fails all five times, the User is queried on line 25 to select the appropriate response from the following list of possibilities:

(R)etry/(A)bort/(C)ontinue/(I)gnore

2. If "Diagnostic Error Messages" is disabled and "Standard Disk Error Retry" is enabled the disk error recovery is attempted five (5) times and if the operation fails all five times the error is reported to BDOS without the line 25 query for appropriate action.

3. If "Diagnostic Error Messages" is enabled and "Standard Disk Error Retry" is disabled the disk recovery is attempted one (1) time and if unsuccessful the User is queried on line 25 for the appropriate action.

4. If "Diagnostic Error Messages" and "Standard Disk Error Retry" are both disabled the disk recovery is attempted one (1) time and if unsuccessful the error is reported to BDOS without the line 25 query.

3.5.1.4 ERRORS

Recoverable errors are reported to the user on line 25 according to the reporting options selected by the user during configuration option selection. Fatal errors are always reported both to the user and to BDOS. An explaination of the error messages is given in Appendix.

3.5.1.5 PERFORMANCE

Nominal.

3.5.1.6 OPERATIONAL CONSIDERATIONS

The system supports one standard QX-10 format and a number of additional formats on the RIGHT drive through the use of the SETUP utility which sets the disk format, density and double sided attributes. The two floppy disk drives are referred to as LEFT and RIGHT respectively by the SETUP and COPYDISK utilities.

On a system without hard disk support, the BIOS addresses the two diskette drives as "A" and "B". When hard disk support is available, disk designations can be altered at system generation time as well as cold boot time.

3.5.2 RAM DISK SUBSYSTEM

3.5.2.1 GENERAL DESCRIPTION

The customized BIOS supports one RAM disk occupying banks 2 and 3 of the QX-10 memory banks. The standard disk format for this implementation is 128 byte sectors with 128 sectors per track.

The system provides a locking mechanism to protect data in the RAM disk from being overlaid by data transferred via XBIOS Calls, to banks 2 and 3.

3.5.2.2 USER/OPERATOR INTERFACES

The RAM disk is configured as drive "M". This designation can be altered at system configuration time as well as by the drive assignment option of the SETUP utility. RAM disk support can be ENABLED or DISABLED through the SETUP utility.

There are three extended BIOS calls, XBIOS 48 - Move Block to another Bank, XBIOS 49 - Call Code in another Bank, and XBIOS 50 - Jump to Code in another Bank, that permit the user to move data and/or code to banks 2 and 3, and excute there. To execute one of these calls, the user must first execute XBIOS 46, Request Memory Bank, which insures that a contention for Banks 2 and 3 does not exist between the XBIOS calls and RAM disk usage.

Upon execution of XBIOS 46, the system checks the status of the RAM disk:

- If the RAM disk status is DISABLED, the requested bank is allocated to the requesting code with a success condition returned.
- If the RAM disk status is ENABLED, an error code is returned to the requesting code.

3.5.2.3 ERROR CONDITIONS

Two error conditions can occur:

- 1, RAM disk is ENABLED when XBIOS 46 is executed.

- 2, RAM disk is ENABLED and contains data, and user attempts to DISABLE RAM disk through SETUP.

If either condition occurs, the folloring message displays:

RAM disk data may be lost: (A)bort or (C)ontinue.

3.5.2.4 ABORTS AND RECOVERIES

If error condition 1:
-(A)bort warm boots with RAM disk still enabled.
-(C)ontinue disables RAM disk, allocates the requested bank.

If error condition 2:
-(A)bort leaves RAM disk ENABLED and warm boots.
-(C)ontinue DISABLES the RAM disk.

3.5.2.5 PERFORMANCE

Given the nature of a RAM disk, high speed data throughout is generally achieved.

3.5.2.6 OPERATIONAL CONSIDERATIONS

Because the RAM disk occupies a maximum of two storage banks, the capacity of drive "M" is limited.

The storage media is volatile. Files can be corrupted at system reset and will be lost at power off. Additional overhead can accrue at cold boot if the given implementation requires loading the RAM disk with files.

3.5.3 WINCHESTER DISK SUBSYSTEM

3.5.3.1 GENERAL DESCRIPTION

The BIOS supports one formatted, ten-megabyte, Comrex hard disk. When installed in the QX-10, the hard disk is divided into two five-megabyte logical drives, H1 and H2. Each logical drive has 512-byte sectors and 16 sectors per track.

3.5.3.2 USER/OPERATOR INTERFACES

The Comrex hard disk is accompanied by software and documentation that:

1. Specifies hardware installation.
2. Describes testing and maintenance procedures.

After physically attaching the hardware to the QX-10, the user logically installs the hard disk. By selecting the Drive Assignment Option of the SETUP utility the user can assign the two logical drives (H1 and H2) of the hard disk to any of the CP/M logical drives (A-P).

Note: The system default drives are A:=F1 and B:=F2, where F = floppy disk.

3.5.3.3 ABORTS AND RECOVERIES

Error reporting and error retry options are available to the user and are selectable from the SETUP utility. These options are as follows:

1. Diagnostic Error Messages
2. Standard Disk Error Retry

The error indications made available to the User during disk error recovery are a function of the status of the Disk Subsystem Options in effect at the time of the anomaly. There are four possible option conditions and the indications for them are summarized below:

1. If "Diagnostic Error Messages" and "Standard Disk Error Retry" are both enabled then the disk error recovery is attempted five (5) times and if the operation fails all five times, the User is queried on line 25 to select the appropriate response from the following list of possibilities:

(R)etry/(A)bort/(C)ontinue/(I)gnore

2. If "Diagnostic Error Messages" is disabled and "Standard Disk Error Retry" is enabled the disk error recovery is attempted five (5) times and if the operation fails all five times the error is reported to BDOS without the line 25 query for appropriate action.

3. If "Diagnostic Error Messages" is enabled and "Standard Disk Error Retry" is disabled the disk recovery is attempted one (1) time and if unsuccessful the User is queried on line 25 for the appropriate action.

4. If "Diagnostic Error Messages" and "Standard Disk Error Retry" are both disabled the disk recovery is attempted one (1) time and if unsuccessful the error is reported to BDOS without the line 25 query.

3.5.3.4 ERRORS

Recoverable errors are reported to the user on line 25 according to the reporting options selected by the user during configuration option selection. Fatal errors are always reported both to the user and to BDOS. An explanation of the error messages is given in Appendix B.

3.5.3.5 PERFORMANCE

The Comrex hard disk spins at 3600 rpm and transmits data at 5 million bits per second--approximately 10 times as fast as a floppy disk. The storage capacity of the Comrex hard disk is 10 to 15 times greater than that of a floppy disk.

3.5.3.6 OPERATIONAL CONSIDERATIONS

Sudden shocks and vibrations can damage the hard disk. Handle with care, particularly when moving it from place to place.

3.6.3 HARD DISK PARTITIONING

3.6.3.1 GENERAL DESCRIPTION

this section to be added

3.6 OPERATING SYSTEM CONFIGURATION

3.6.1 COLD BOOT LOADER

3.6.1.1 GENERAL DESCRIPTION

The CP/M-80 R2.2 operating system is loaded into memory in two steps. The QX-10 IPL ROM loads in the Cold Boot Loader into memory. The Cold Boot Loader loads the CP/M Loader which in turn loads CP/M-80.

The IPL ROM brings in the first 4K of data from track 0 of the diskette in the LEFT drive. Included in this 4K is the Boot Loader program. The Boot Loader program relocates itself down to location 100H (decimal 256) from the location F000H. The loader program then loads the CP/M Loader program into memory.

The CP/M Loader program is comprised of a condensed loader BDOS and a skeletal BIOS which provides disk I/O capablities to the loader. The loader searches the directory for file CPM2.SYS which is read into memory. The file contains load addresses and the entire CP/M structure. The loader 'loads' CP/M and jumps to the cold boot vector.

System configuration information, as defined by the CP/M-80 configuration utility SETUP, is also contained in the system track area. This information is not processed by the loaders.

3.6.1.2 USER/OPERATOR INTERFACES

When the system initiates the boot sequence the "INSERT DISKETTE" message is erased from the screen. The next message to be displayed is the sign-on message which is centered horizontally and displayed near the top of the screen. The exact format of the sign-on message is given below:

CP/M-80 R2.2 FOR THE EPSON QX-10

256K - VERSION B2.26

NOTE: The CP/M-80 A> prompt is shown for reasons of completeness and is not part of the sign-on message itself.

3.6.1.3 ABORTS AND RECOVERIES

Standard disk error recovery is utilized as defined in Appendix B.1.

3.6.1.4 ERRORS

If drive A is not ready, the boot ROM displays the message "INSERT DISKETTE". If a non-system diskette is inserted into drive A, the ROM will unsuccessfully attempt to read the boot loader and continuously repeats the "INSERT DISKETTE" message.

3.6.1.5 PERFORMANCE

Initial program load time takes approximately 17 seconds.

3.6.1.6 OPERATIONAL CONSIDERATIONS

The sign-on message and the CP/M-80 A> prompt are displayed in normal intensity. At the completion of the Boot Loader procedure the system is left in the normal intensity mode.

3.6.2 GENCPM - SYSTEM GENERATION

3.6.2.1 GENERAL DESCRIPTION

GENCPM2.COM is the system generation module for the B Release of the Epson CP/M-80. Its purpose is to modify the existing CPM2.SYS file, which contains the system BIOS, BDOS, and the CCP.

To perform system generation, GENCPM2 requires the presence of two files:

- The CPM2.SYS file for the existing system
- EBIOS.SPR, the System Page Relocatable file, which is created via selection of the B switch of the linker and contains the non-relocated BIOS Code.

3.6.2.2 USER/OPERATOR INTERFACES

OEM (User) decides which portion of the BIOS to assign to common RAM (CSEG) and which portion to assign to banked RAM (DSEG), and makes the appropriate assignments in source code using the CSEG and DSEG assembler directives.

OEM (User) makes system modifications with an editor, assembles the modified files with either RMAC (DRI) or M80 (Microsoft), links the modules with LINK80 (DRI) using the linker B switch to create EBIOS.SPR, and finally runs GENCPM2.COM. GENCPM2.COM relocates the DSEG and CSEG Code segments and appends the relocated BIOS onto the CMP2.SYS file.

If there are no errors, the Memory Map (displaying the locations of CSEG and DSEG) will be displayed. If there are errors, error messages will be displayed.

3.6.2.3 ABORTS AND RECOVERIES

GENCPM2 aborts on any error condition. There are no recovery techniques.

3.6.2.4 ERRORS

GENCPM2 checks for the presence of the CPM2.SYS and EBIOS.SPR files. If either of these files is not present, the appropriate message is displayed:

ERROR! EBIOS.SPR FILE NOT FOUND

ERROR! CPM2.SYS FILE NOT FOUND

If both CPM2.SYS and EBIOS.SPR are present, GENCPM2 takes the header information from EBIOS.SPR and checks the size of CSEG and DSEG.

If CSEG as coded in the EBIOS.SPR file exceeds the boundaries allotted to it, the following message is displayed:

ERROR! CSEG TOO LARGE

If DSEG as coded in the EBIOS.SPR file exceeds the boundaries allotted to it, the message is displayed:

ERROR! DSEG TOO LARGE

Other possible error messages are CLOSE ERROR, which indicates that the CPM2.SYS file could not be closed, and READ/WRITE errors in accessing EBIOS.SPR or CPM2.SYS.

3.6.2.5 PERFORMANCE

The error-checking capabilities of GENCPM2 protect the user from unintentionally overlaying code.

3.6.2.6 OPERATIONAL CONSIDERATIONS

GENCPM2.COM updates the CPM2.SYS file present on the currently selected drive. The CPM2.SYS file would need to be copied to the bootable system diskette.

To make additional copies of the operating system, the user runs the "Copy System Only" option of the Epson COPYDISK utility, which copies both the system tracks and the CPM2.SYS file to the destination diskette.

3.6.3 OEM CONFIGURATION SUPPORT

3.6.3.1 GENERAL DESCRIPTION

OEM support is a major feature of the "B" release. The BIOS code is divided into logic modules (patterned after CP/M 3.0 structure) which are linked together to form the CP/M environment. OEM supplied modules which adhere to a defined structure can be included in the link process and become an integral part of the CP/M environment provided to the end user.

3.6.3.2 USER/OPERATOR INTERFACES

A dummy module named OEMINIT is provided for the use of the OEM to customize the system. This routine has two entry points: ?OEMIC and ?OEMIW. The entry point ?OEMIC is called by the cold boot logic, while ?OEMIW is called by the warm boot logic.

The OEM may add whatever logic is required to customize the operating system to his specific requirements. For example the OEM can add an interrupt routine to the system to handle a new device such as a hard disk drive. Another routine can be added which initializes another device at warm boot.

Below are outlined the guidelines which an OEM must follow to integrate customized routines to the QX-10 CP/M BIOS:

- All interrupt routines must be initialized in the ?OEMIC routine. The start of the interrupt vector table can be determined by references to the system variable @IPBASE. The ?OEMIC logic must set the interrupt routine address in the appropriate vector location in the table.

- Interrupts must not be enabled in either the ?OEMIC or ?OEMIW routines.

- The interrupt routine initialization logic must set the mask value for the appropriate 8259. The OEM must know which 8259 is responsible for handling the interrupts for the device for which the routine is being provided. The interrupt mask for the first 8259 can be found at @8259A and the second at @8259B. The initialization logic must access the value at that location and "OR" in the necessary mask bit for that device. Upon return from ?OEMIC the system will set the interrupt masks for each 8259.

- The interrupt routine must be in the same context (address space) as the system interrupt vector. No provision has been made for the system to perform a bank switch to transfer control to an interrupt routine.

- An OEM interrupt routine is responsible for sending the EOI command to the appropriate 8259 after the interrupt has been serviced. The routine must exit with the RETI instruction.

- If at anytime the OEM's logic wishes to change the mask bit for a device controlled by the OEM'S routines, the appropriate mask byte may be accessed and changed by the procedure outlined above. The EOI routine may then call ?825CA or ?825CB the have the mask value changed.

3.6.3.3 ABORTS AND RECOVERIES

The responsibility of the OEM.

3.6.3.4 ERRORS

The responsibility of the OEM.

3.6.3.5 PERFORMANCE

Not applicable.

3.6.3.6 OPERATION CONSIDERATIONS

- The initialization logic in ?OEMIC cannot program the 8259 controllers. The OEM must use the system values for the 8259 initialized by the BIOS.

- An OEM supplied interrupt routine is responsible for saving and restoring CPU status and all registers which are used by the routine.

3.7 SERIAL INTERFACE SUPPORT

3.7.1 GENERAL DESCRIPTION

The RS-232 serial port, of the QX-10 system is supported by the customized BIOS. The serial port is supported as character stream input/output devices and can be connected, by the User, to CP/M-80 R2.2 logical devices with the STAT utility. The System Configuration Utility is used to set the port characteristics and protocol to be supported by the BIOS routines. The ETX/ACK or XON/XOFF software handshaking protocols as well as the RTS/CTS hardware handshaking are supported by the BIOS routines.

Text to be added discussing interrupt/polled drivers.

3.7.2 USER/OPERATOR INTERFACES

The user interfaces with the serial port support through SETUP utility described in section 3.8. Port characteristics are defined through SETUP.

3.7.3 ABORTS AND RECOVERIES

None.

3.7.4 ERRORS

Text to be added discussing interrupt/polled drivers.

3.7.5 PERFORMANCE

Text to be added discussing interrupt/polled drivers.

3.7.6 OPERATIONAL CONSIDERATIONS

Text to be added discussing interrupt/polled drivers.

4.0 SYSTEM LEVEL DESCRIPTION

4.1 PUBLICATIONS

User Manual

Informs the User of CP/M-80 R2.2 on the Epson QX-10 System of the specific characteristics of this customization of CP/M-80 R2.2 of which they should be aware. This would include the use of the keyboard and a description of the video display programming interface as well as unique utility programs.

Programmer's Manual

The External Reference Specification plus the Digital Research CP/M-80 documentation provides the technical information for the applications programmer who must customize some application program (such as full screen word processor) to the Epson QX-10 System.

Maintenance Manual

This document contains the technical specifications and program logic of the Customized BIOS and utilities for CP/M-80 R2.2 on the Epson QX-10 system. This manual would be used by those who are responsible for maintaining this customization of CP/M-80 R2.2. This ERS and the program source listings provide this information.

4.2 EQUIPMENT CONFIGURATION

HASCI keyboard, 2 diskette drives, Monochrome display, 256K RAM, 1 parallel port, and one RS232C serial port.

4.3 INTERFACES WITH OTHER SOFTWARE

Although not specifically addressed in this customization, the ground work is laid in place to not preclude the interface with the CP/M Plus, MP/M-80 and CP/NET-80 software products. Support for Synchronous communication capability using the serial ports is not addressed in this implementation. Additional marketing requirements are needed in order to externally specify this feature.

4.4 RELIABILITY, AVAILABILITY, SERVICEABILITY

All errors will be reported to the user. Recovery will be attempted if possible. Any unknown errors will cause a warm boot of CP/M-80 R2.2.

APPENDICES

APPENDIX A. Memory Map

The memory map for this implementation of CP/M-80 is defined as follows:

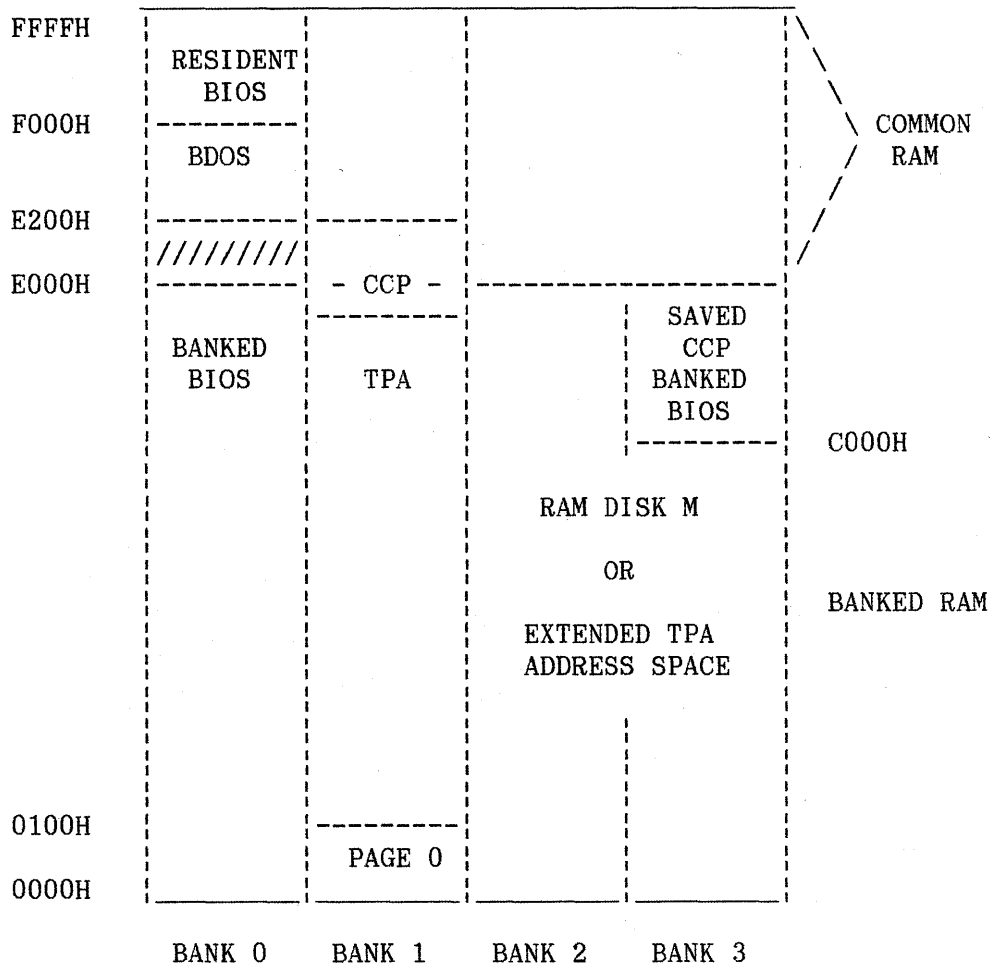


Table 1-5 Memory Map

MODULE	BASE ADDRESS
BIOS	F000H
BDOS	E200H
CCP	DA00H

Table 1-6 Operating System Base Addresses

The memory map in Table 1-5 indicates full utilization of 256K RAM. Bank 0 is reserved as the system bank. The BIOS is divided into two distinct sections. The resident section resides in common RAM and is responsible for intercepting BIOS calls, swapping in the system bank, and vectoring to the appropriate code in the banked BIOS. By segmenting the BIOS as shown, code overhead is reduced to a minimum enabling the TPA to increase to an acceptable size. The TPA extends into common RAM. The shaded area indicates the portion of common RAM overlayed by the TPA. If the application program residing in the TPA does not make any BDOS calls, the TPA can also overlay BDOS. Table 1-6 shows the base addressess of the BIOS, BDOS and CCP.

The RAM disk is mapped onto banks 2 and 3. The RAM disk is activated/deactivated through a SETUP selection. When the RAM disk is disabled, an intelligent program using XBIOS calls 48, 49 and 50 can use banks 2 and 3 as additional data and code sections.

APPENDIX B. XBIOS Interface

Table 1-7 below describes the entry points into the BIOS. The first 16 entry points correspond to the standard CP/M-80 implementation. Entries 16-29 are null entries which vector to a RET instruction producing a return to the invoking code. The last entry point, QXFUNC, is an Epson unique entry point through which the extended BIOS services are provided. QXFUNC is the 30th vector which in CP/M Plus is reserved for System Implementor. This provides a common entry point in both CP/M-80 and CP/M Plus.

No.	Entry Name	Function
0	BOOT	Cold Boot Entry
1	WBOOT	Warm Boot Entry
2	CONST	Console Input Status
3	CONIN	Console Input
4	CONOUT	Console Output
5	LIST	List Output
6	PUNCH	Punch Output
7	READER	Reader Input
8	HOME	Home Selected Disk
9	SELDSK	Select Disk
10	SETTRK	Set Track Number
11	SETSEC	Set Sector Number
12	SETDMA	Set DMA Address
13	READ	Read Sector
14	WRITE	Write Sector
15	LISTST	List Device Status
	...	Vector to Return Instruction
30	QXFUNC	QX-10 XBIOS Functions

Table 1-7 BIOS Entry Points

The QX-10 XBIOS Functions are described on the following pages.

The CP/M convention for accessing BDOS functions is adopted for accessing the XBIOS Functions. The XBIOS Functions are accessed by making a direct call to QXFUNC. The function number is passed in register C with the information address in the double byte pair DE. The XBIOS calls adhere to normal CP/M conventions concerning BIOS responsibilities in register maintenance.

- The XBIOS routines do not maintain registers. The results of registers upon return is is guaranteed only for registers returning result codes or specified data.

- The XBIOS routines use the alternate register set. An application using the alternater register set must save registers prior to making an XBIOS call.

- The XBIOS routines use the IX and IY registers. The application program must preserve the IX and IY registers prior to making an XBIOS call.

XBIOS 00	Floppy Disk Drive Function
XBIOS 01	Flush Active Floppy Disk Buffer
XBIOS 02	Disk Error Reporting
XBIOS 03	Get DRVTBL Address
XBIOS 04	Set Foreign Disk Format
XBIOS 05	Get System Data Page
XBIOS 06	Update System Data Page
XBIOS 07	Put System Data Page
XBIOS 08	Get Time
XBIOS 09	Put Time
XBIOS 10	Return BIOS Version Number
XBIOS 11	Display User Message on Status Line
XBIOS 12	Reserved
XBIOS 13	Load Font
XBIOS 14	Select Font
XBIOS 15	Read Custom Font
XBIOS 16	Write Custom Font
XBIOS 17	Save Current Font Number
XBIOS 18	Restore Current Font Number
XBIOS 19	Read International Font
XBIOS 20	Get Scan Code Table
XBIOS 21	Put Scan Code Table
XBIOS 22	Save Scan Code Table
XBIOS 23	Restore Scan Code Table
XBIOS 24	Get Scan Character
XBIOS 25	Put Scan Character
XBIOS 26	Read Scan Code Table
XBIOS 27	Write Scan Code Table
XBIOS 28	Set BIOS To Return Scan Code or Converted Character

XBIOS 29	Set Function Key Enable/Disable Mode
XBIOS 30	Select Function Key Table
XBIOS 31	Save Current Function Key Table Pointer
XBIOS 32	Restore Function Key Table Pointer
XBIOS 33	Get Selected Function Key Table
XBIOS 34	Put Selected Function Key Table
XBIOS 35	Save Selected Function Key Table
XBIOS 36	Restore Selected Function Key Table
XBIOS 37	Get Function Key
XBIOS 38	Put Function Key
XBIOS 39	Disable System Key
XBIOS 40	Enable System Key
XBIOS 41	Read Custom Function Key Table
XBIOS 42	Write Custom Function Key Table
XBIOS 43	Enable Keyboard Reset at Warm Boot
XBIOS 44	Perform Warm Boot Reset
XBIOS 45	Disable Keyboard Reset - XBIOS 43
XBIOS 46	Request Memory Bank
XBIOS 47	Release Memory Bank
XBIOS 48	Move Block to another Bank
XBIOS 49	Call Code in another Bank
XBIOS 50	Jump to Code in another Bank
XBIOS 51	Display RAM Disk Overwrite Message
XBIOS 52	Reserved
XBIOS 53	Reserved
XBIOS 54	PSET - Pixel Set/Reset
XBIOS 55	Set Viewport
XBIOS 56	Graphics Drawing Primatives
XBIOS 57	Direct Cursor Addressing
XBIOS 58	Direct String Display Function
XBIOS 59	Reserved
XBIOS 60	Set User Interrupt Vector
XBIOS 61	Clear User Interrupt Vector
XBIOS 62	Get Interrupt Mask
XBIOS 63	Enable Interrupt
XBIOS 64	Disable Interrupt
XBIOS 65	Not Available
XBIOS 66	Not Available
XBIOS 67	Not Available
XBIOS 68	Not Assigned
XBIOS 69	Not Assigned
XBIOS 70	Clear mode attributes in Character Driver
XBIOS 71	Clear mode attributes in Bitmap Driver
XBIOS 72	Set Cursor Shape
XBIOS 73	Load CCP into TPA memory
XBIOS 74	Move boot string into CCP
XBIOS 75-90	Not Assigned
XBIOS 91	Attach Internal Communications Driver
XBIOS 92	Detach Internal Communications Driver
XBIOS 93-127	Not Assigned

XBIOS 00 Floppy Disk Drive Function

Entry: A = Error Reporting Control
        0 - BIOS reports error
        1 - Error Returned to Calling Code

        C = 00
        DE = Parameter Buffer Address

Buffer Contents: byte 0 - Unit number 0,1
                  0: Left Drive
                  1: Right Drive

                  byte 1 - Function code
                  0: Home
                  1: Read data track
                  2: Write data track
                  3: Read system track
                  4: Write system track

                  byte 2 - Sector
                  0-31: system track
                  0-19: data track

                  byte 3 - Track
                  0-39

                  byte 4,5 - DMA Address (Intel format)
                  DMA buffer size
                  256 bytes: system track
                  512 bytes: data track

Return: A = 0 - operation successful
         A <> 0 - operation unsuccessful
              3 - read/write error
              4 - seek error
              5 - disk not ready error

XBIOS 01 Flush Active Floppy Disk Buffer

Description:
        This XBIOS call will write a 'dirty' disk buffer onto the affected diskette.

Entry: C = 01

Return: A = 0 - operation successful
         A <> 0 - operation unsuccessful

XBIOS 02 Disk Error Reporting

Description:

This XBIOS call displays disk error message on the 25th line and waits for user response. Although it prompts user to enter (R) for retry, (A) for abort, (C) for continue and (I) for ignore, the program does not limit the entry to these 4 characters. If "A" is entered, the system will select drive A then do a warmboot. All other character will be converted to upper case (ANI 0DFH) and returned in register A. It is the responsibility of the application program to take care of retry, continue, ignore or other operations after receiving the character from user.

Entry: C = 02
A = Error Code
3 - read/write error
4 - seek error
5 - drive not ready
B = Error Drive Designator (0-15)

Return: A = Action Code
"R" - Retry
"C" - Continue
"I" - Ignore
Other characters entered by user
A = -1 indicates error in input parameters

XBIOS 03 Get DRVTBL Address

Description:

Drvtbl defines the physical to logical disk drive assignment. Sixteen entries are available for logical drive A to P. For existing physical drive, the address of its disk parameter table is put in the proper entry corresponding to the logical drive letter. Entries for non-existing drives are set to zero.

Entry: C = 03

Return: DE = Address of DRVTBL in common RAM

XBIOS 04 Set Foreign Disk Format

Entry: C = 04
DE = Pointer to DPB and NEC 765 parameter table.

Following are tables for QX-10 380K format.

```
@DT5B:  DW      80              ;SECTORS PER TRACK
        DB      4               ;BSH
        DB      15              ;BLM
        DB      1               ;EXM
        DW      190-1           ;BLOCKS-1
        DW      128-1           ;TOTAL DIR ENTRIES
        DW      192             ;DIR BLOCK ALLOCATION
        DW      32              ;CHECKED DIR ENTRIES
        DW      2               ;DIRECTORY TRACK

@PTBLB:
        DB      16              ;BLKSIZE/128
        DB      80              ;HSTBLK * HSTSPT
        DB      2               ;SECTOR SHIFT FACTOR
        DB      3               ;SECTOR MASK
        DB      10              ;# SECTORS/TRK
        DB      15              ;DISK GAP FACTOR
        DB      2               ;NEC 765 SECTOR FACTOR
        DW      512-1           ;DMA BYTE CNT FACTOR
        DB      4               ;SECTOR TRANSLATE FACTOR
        DB      0               ;LONG DISK FLAG
        DB      40              ;DISKETTE TRACK COUNT
```

XBIOS 05 Get System Data Page

Description:
Get a copy of current system data page in memory into a 128 byte buffer pointer by register DE.

Entry: C = 05
DE = Address of 128 byte buffer

Return: Contents of SDP moved to buffer

XBIOS 06 Update System Data Page

Description:
Put the content of the 128 byte buffer pointed by register DE into current system data page in memory.

Entry: C = 06
DE = Address of 128 byte buffer

Return: Contents of buffer moved to SDP

XBIOS 07 Put System Data Page

Description:
Copy the content of current system data page in memory into diskette. The bootable diskette must be in the left drive. System data page is located at physical track 0, sector 2 of the diskette. The description of each byte in the system data page can be found at appendix D.

Entry: C = 07

Return: A = 0 - operation successful
A <> 0 - operation unsuccessful

XBIOS 08 Get Time

Entry: C = 08
DE = Address of a 7 byte buffer

Buffer contents:
byte 0 - day of week (1-7)
byte 1 - month (1-12)
byte 2 - day (1-31)
byte 3 - year (0-99)
byte 4 - hour (0-23)
byte 5 - minute (0-59)
byte 6 - second (0-59)

Return: Date and Time information moved to buffer

XBIOS 09 Put Time

Entry: C = 09
DE = Address of a 7 byte buffer

Buffer contents: same as XBIOS 08

Return: Date and Time information moved from buffer and clock chip initialized.

XBIOS 10 Return BIOS Version Number

Entry: C = 0AH

Return: BIOS version number returned as ascii characters in registers DE and HL. Current version:

DE = B2
HL = 25

XBIOS 11 Display User Message on Status Line

Entry: C = 0BH
HL = Address of ascii string terminated with null byte.

Return: A = User keyboard response to status message. Depending on XBIOS 28, reg A contains keyboard scancode or converted character.

XBIOS 12 RESERVED

Entry: C = 0CH

Description: Reserved for future expansion. No operation.

XBIOS 13 Load Font

Description:

This XBIOS call is used to load a user defined font into STYLE font area. Each individual character requires 16 bytes of data. The following example shows how to build a character out of the 16 byte bit image.

The 16 byte data for standard font "B" is

41h,41h,21h,1fh,0,0,0,0,1fh,21h,41h,41h,21h,1fh,21h,41h

		bit	0	1	2	3	4	5	6	7
byte	data									
8	1f		x	x	x	x	x			
9	21		x					x		
10	41		x						x	
11	41		x						x	
12	21		x					x		
13	1f		x	x	x	x	x			
14	21		x					x		
15	41		x						x	
0	41		x						x	
1	41		x						x	
2	21		x					x		
3	1f		x	x	x	x	x			
4	0									
5	0									
6	0									
7	0									

The buffer size for the STYLE font is 0C00H. The first 0600H is designated to the STYLE font and the remaining 0600H to the STYLE BOLD font. The fonts are numbered from 00H to 5FH which corresponds to ASCII code 20H to 7FH. Refer to appendix J for fonts tables.

Entry: C = 0DH
DE = Address of buffer area containing font

Return: Font moved to STYLE font area

XBIOS 14 Select Font

Entry: C = 0EH
A = Font # to be selected
0: standard
1: standard bold
2: italic
3: italic bold
4: style
5: style bold
others: return with no font changed

Return: Font pointer updated to point to selected font

XBIOS 15 Read Custom Font

Description: Read STYLE fonts from system track into memory.

Entry: C = 0FH

Return: A = 0 - operation successful
A <> 0 - operation unsuccessful

XBIOS 16 Write Custom Font

Description: Write STYLE fonts from memory to system track.

Entry: C = 10H

Return: A = 0 - operation successful
A <> 0 - operation unsuccessful

XBIOS 17 Save Current Font Number

Entry: C = 11H

Return: The current font pointer is saved.

XBIOS 18 Restore Font Number

Entry: C = 12H

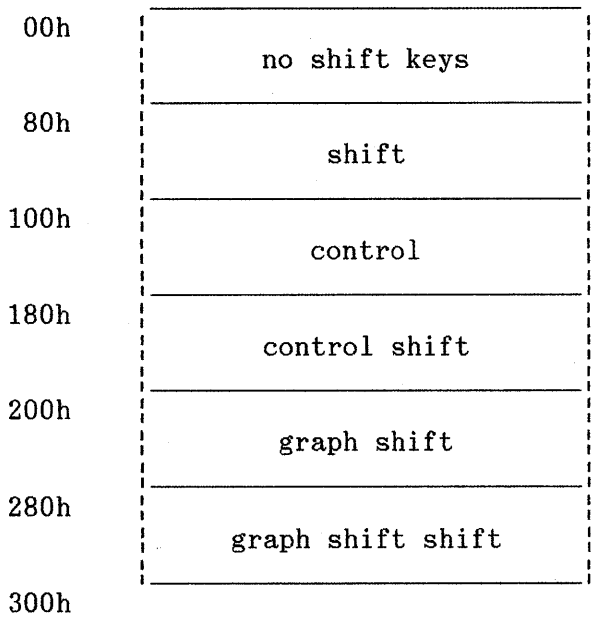
Return: The font pointer is restored.

XBIOS 19 RESERVED

Entry: C = 13H

Description: Reserved for future expansion. No operation.

XBIOS calls 20-28 deal with the keyboard scan code table. The size of the scan code table is 0300H bytes. The tables contain the maps to translate keyboard scan codes to 8-bit data bytes which are input into the BIOS console input buffer. System keys and Function keys are handled in other tables. The structure of the 0300H table is shown as follows:



Refer to Appendix F for scan code tables.

XBIOS 20 Get Scan Code Table

Entry: C = 14H
DE = Address of 1K buffer

Return: Keyboard Scan Code Table moved to buffer

XBIOS 21 Put Scan Code Table

Entry: C = 15H
DE = Address of 1K buffer containing user-defined scan code table.

Return: User scan codes moved over system scan code table

XBIOS 22 Save Scan Code Table

Entry: C = 16H

Return: Contents of System Scan Code Table copied to save area

XBIOS 23 Restore Scan Code Table

Entry: C = 17H

Return: Contents of scan code save area copied over System Scan Code (only if a XBIOS 22 function previously performed)

XBIOS 24 Get Scan Character

Entry: C = 18H
D = Shift Mask
000B : no shift keys
001B : shift
010B : control
011B : control shift
100B : graph shift
101B : graph shift shift
E = Scan Code
0 - 7FH

Return: A = Assigned Code

Example: If D = 0 , E = 43h
Then A = 61h
See appendix F.

XBIOS 25 Put Scan Character

Entry: C = 19H
D = Shift Mask
000B : no shift keys
001B : shift
010B : control
011B : control shift
100B : graph shift
101B : graph shift shift
E = Scan Code
0 - 7fh
A = Code to be assigned

XBIOS 26 Read Scan Code Table

Description:
Read scan code table from system track to memory.

Entry: C = 1AH

Return: A = 0 - operation successful
A = 1 - operation unsuccessful

XBIOS 27 Write Scan Code Table

Description:
Write the current scan code table from memory to system track.

Entry: C = 1BH

Return: A = 0 - operation successful
A <> 0 - operation unsuccessful

XBIOS 28 Set BIOS To Return Scan Code or Converted Character

Entry: C = 1CH
A = 0 - Return Converted Scan Code
A <> 0 - Return Scan Code

XBIOS calls 29 - 45 deal with function key tables. Refer to Appendix G for a listing of Function Keys.

XBIOS 29 Set Function Key Enable/Disable Mode

Entry: C = 1DH
A = 0 Enable Function Keys
<> 0 Disable Function Keys

Description: Enables/Disables function key interpretation. Warmboot always enables function keys.

XBIOS 30 Select Function Key Table

Entry: C = 1EH
A = Selection Code
0 - HASCI Table
1 - TVI-920 Table
2 - Custom Table

Return: A = 0 - operation successful
A <> 0 - operation unsuccessful

XBIOS 31 Save Current Function Key Table Pointer

Entry: C = 1FH

XBIOS 32 Restore Function Key Table Pointer

Entry: C = 20H

XBIOS 33 Get Selected Function Key Table

Entry: C = 21H
DE = Address of 1K buffer to hold CFK table

Return: Selected Function Key Table moved to buffer

XBIOS 34 Put Selected Function Key Table

Entry: C = 22H
DE = Address of 1K buffer containing a user defined PFK table

Return: User defined PFK copied over Selected CFK.

XBIOS 35 Save Selected Function Key Table

Entry: C = 23H

Return: Contents of Selected Function Key Table copied to save area

XBIOS 36 Restore Selected Function Key Table

Entry: C = 24H

Return: Contents of PFK save area copied over Selected Function Keys (only if a XBIOS 35 function previously performed)

XBIOS 37 Get Function Key

Entry: C = 25H
DE = Address of 25 byte buffer
A = Function Key # (0 - 22H)

Return: Selected function key string moved to buffer

XBIOS 38 Put Function Key

Entry: C = 26H
DE = Address of 25 byte buffer containing function key definition
A = Function Key # (0 - 22H)

Return: Funtion key string in buffer moved to Selected Function Key Table

XBIOS 39 Disable System Key

Entry: C = 27H
A = System Key Number

key no.	key label	function
0	bold	toggle bold fonts
1	italic	toggle italic fonts
2	size	reserved
3	style	toggle style fonts
4	control stop	system pause
5	control grph/shift stop	cold boot
6	control help	status/help display
7	control print	screen dump

Return: Selected System Key is disabled

XBIOS 40 Enable System Key

Entry: C = 28H
        A = System Key Number

        key numbers and definition same as XBIOS 39

Return: Selected System Key is enabled

XBIOS 41 Read Custom Function Key Table

Description:
        Read custom function key table from system track to memory.

Entry: C = 29H

Return: A = 0 - operation successful
         A <> 0 - operation unsuccessful

XBIOS 42 Write Custom Function Key Tables

Entry: C = 2AH

Return: A = 0 - operation successful
         A <> 0 - operation unsuccessful

XBIOS 43 Enable Keyboard Reset at Warm Boot

Entry: C = 2BH

Description: Operation causes XBIOS functions 23, 36 & 32 to be performed in that order at the next warm boot or XBIOS 44 request. Process resets the XBIOS 43 request. XBIOS 45 disables request made by XBIOS 43.

XBIOS 44 Perform Warm Boot Reset

Entry: C = 2CH

Description: Performs the warm boot processes normally invoked by XBIOS 43. This function should be used by applications which trap the warm boot vector as a restart mechanism.

XBIOS 45 Disarm Keyboard Reset

Entry: C = 2DH

Description: Disarms reset function armed by XBIOS 43 call.

XBIOS 46 Request Memory Bank

Entry: C = 2EH
A = Bank # to request.

Return: A = 0 - operation successful
A = 1 - illegal bank # requested
2 - RAM disk is enabled and contains data
Code can use XBIOS 51 to report condition to user.

XBIOS 47 Release Memory Bank

Entry: C = 2FH
A = Bank # to release

XBIOS 48 Move Block to another Bank

Entry: C = 30H
DE = Parameter Buffer Address

Buffer Contents:
byte 0 - Source Bank (0 - 3)
byte 1 - Destination Bank (1 - 3)
bytes 2,3 - Source Address (Intel format)
bytes 4,5 - Destination Address (Intel format)
byte 6,7 - Byte Count (Intel format)

Return: A = 0 if successful
A = <> if not successful

XBIOS 49 Call Code in another Bank

Entry: C = 31H
DE = Destination Address
A = Destination Bank (1 - 3)

Return: The selected bank is enabled and program control is transferred to selected address. If an illegal bank request is made, an error message is displayed on the status line and a warm boot is performed. Calls can be nested 10 levels deep. Parameters can be passed on the stack.

XBIOS 50 Jump to Code in another Bank

Entry: C = 32H
DE = Destination Address
A = Destination Bank (1 - 3)

Return: The selected bank is enabled and program control is transferred to selected address. If an illegal bank request is made, an error message is displayed on the status line and a warm boot is performed.

XBIOS 51 Display RAM Disk Overwrite Message

Entry: C = 33H

Description: Displays the RAM Disk Overwrite Message. Allows the operator to abort the process or continue. Upon continuing, RAM disk is cleared and previous Bank requested competed.

XBIOS 52 Request Bank Error Overwrite

Entry: C = 34H
A = 0 - Requested Bank is enabled. Ramdisk is disabled.
<>0 - Requested bank is enabled. Ramdisk is left enabled. User beware.

XBIOS 53 RESERVED

Entry: C = 35H

Description: Reserved for future expansion. No operation.

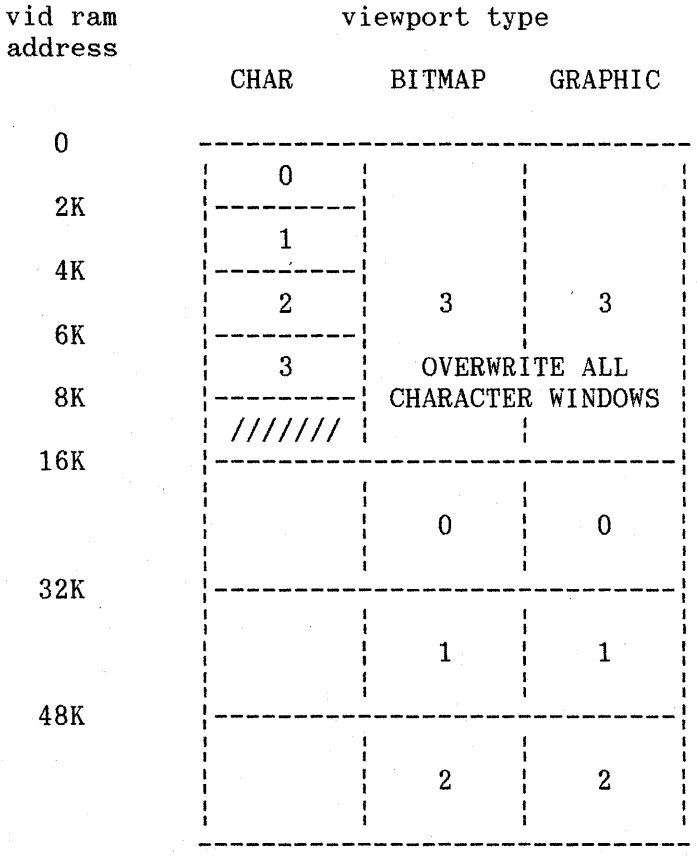
XBIOS 54 PSET - Pixel Set/Reset

Entry: C = 36H
DE = X coordinate (0-639)
HL = Y coordinate (0-399)
A = Operation
1 - Replace
2 - Overstrike
3 - Complement
4 - Erase

XBIOS 55 Set Viewport

Entry: C = 37H
A = screen width (80 or less sets the width to a default of 80 chars. Max = 639)
B = viewport number [0..3]
H = viewport type [0..2]
0 = Character mode
1 = Bit-map (non-graphic)
2 = Graphic
L = clear/no-clear flag [0..1]
0 = No clear
1 = Clear

Description: Sets current Video View Port



Viewport video ram mappings

XBIOS 56 Graphics Drawing Primatives

Description: This entry point is used to invoke the various GDP's supported by the BIOS. GPD's detailed in Appendix E.

Entry: C = 38H
B = Requested GDP
0 - Write Mode
1 - Set Pixel
2 - Read Position
3 - Line
4 - Arc
5 - Pie Slice
6 - Circle
7 - Rectangle
8 - Polyline
9 - Polygon
10 - Block
11 - Text
12 - Set Color
13 - Set Line Pattern
14 - Set Fill Pattern
15 - Hide Screen
16 - Display Screen
17 - Fill Screen

XBIOS 57 Direct Cursor Addressing

Entry: C = 39H
D = Row (0-23)
E = Column (0-79)

Description: Positions cursor to requested position.

BIOS 58 Direct String Display Function

Entry: C = 3AH
B = Direction to display character:
1 - Horizontal (Right)
2 - Vertical (Down)
D = Number of characters to display.
E = Character to display.

XBIOS 59 RESERVED

Entry: C = 3BH

Description: Reserved for future expansion. No operation.

XBIOS 60 Set User Interrupt Vector

Entry: C = 3CH
A = Interrupt Number to be set
B = Bank Where User Interrupt Service Routine Resides
DE = User Interrupt Service Routine Address

00 - Power Down Detect
01 - Software Timer #1
02 - External Interrupt INTF1
03 - External Interrupt INTF2
04 - Protected (Keyboard/Serial Port)
05 - CRT/Light Pen Interrupt
06 - Floppy Controller Interupt
07 - Protected (Slave Cascade)
08 - Printer
09 - External Interrupt #1
10 - Protected (Calendar Clock)
11 - External Interrupt #2
12 - External Interrupt #3
13 - Protected (Software Timer #2)
14 - External Interrupt #4
15 - External Interrupt #5
16 - Keyboard (Interrupt 4 Decoded)
17 - Serial Port (Interrupt 4 Decoded)

XBIOS 61 Clear User Interrupt Vector

Entry: C = 3DH
A = Interrupt Number

This routine restores the interrupt to the system default condition.

XBIOS 62 Get Interrupt Mask

Entry: C = 3EH

On Return:
DE = Interrrupt Mask

This routine returns the 8259 interrupt mask.

XBIOS 63 Enable Interrupt

Entry: C = 3FH
DE = Interrupt Mask

This routine enables the interrupts that correspond to the bits in DE that are set to 1 where the mask in register E affects the master 8259 and the mask in register D affects the slave 8259.

XBIOS 64 Disable Interrupt

Entry: C = 40H
DE = Interrupt Mask

This routine disables the interrupts that correspond to the bits in DE that are set to 1.

XBIOS 65,66,67 -- Not Available

Description: The interrupt handler has been modified in B2.25. XBIOS 65, 66 and 67 are unsupported. A message is displayed indicating the running program is not supported under B2.25 and then warm boots the system.

Entry: C = 41H,42H,43H

XBIOS 70 Clear field attributes in Character Driver

Entry: C = 46H

XBIOS 71 Clear field attributes in Bitmap Driver

Entry: C = 47H

XBIOS 72 Set Cursor Shape

Entry: C = 48H
HL = Address of array containing cursor shape

XBIOS 73 Load CCP into TPA memory

Entry: C = 49H

XBIOS 74 Move boot string into CCP

Entry: C = 4AH
HL = Address of ASCIIZ string.

Command is invoked at each warmboot.

XBIOS 75-90 Not Assigned

XBIOS 91 Attach Internal Communications Driver

Entry: C = 5BH

XBIOS 92 Detach Internal Communications Driver

Entry: C = 5CH

XBIOS 93-127 Not Assigned

APPENDIX C. Graphics Drawing Primatives

Primitive Number	Name	Parameter Block Definition		Comments
0	Write Mode	Mode Byte	: HL	0 = Replace 1 = Complement (XOR) 2 = Erase 3 = Overstrike (OR)
1	Set Pixel	X Coord	: HL	0..639
		Y Coord	: DE	0..399
2	Read Position	no parms		Returns: X Coord. : HL Y Coord. : DE
3	Line	Starting X	: HL	
		Starting Y	: DE	
		Ending X	: IX	
		Ending Y	: IY	
4	Arc	Center X	: word	
		Center Y	: word	
		Radius	: word	
		Starting Angle	: word	(0..359)
		Ending Angle	: word	(0..359)
5	Pie Slice	Fill Mode	: A	See note # 3
		Center X	: word	
		Center Y	: word	
		Radius	: word	
		Starting Angle	: word	(0..359)
		Ending Angle	: word	(0..359)
6	Circle	Fill Mode	: A	See note # 3
		Center X	: HL	
		Center Y	: DE	
		Radius	: IX	
7	Rectangle	Fill Mode	: A	See Note # 3
		Bottom Left X	: HL	
		Bottom Left Y	: DE	
		Upper Right X	: IX	
		Upper Right Y	: IY	

Primitive Number	Name	Parameter Block Definition		Comments
8	Polyline	Number of lines :	DE	
		Coord Array Addr:	HL	See Note # 1
9	Polygon	Fill Mode :	A	See note # 3
		# of vertices :	DE	
		Coord Array Addr:	HL	See Note # 1
10	Block	Mode byte :	word	0 = Write , 1 = Read
		X dimension :	word	
		Y dimension :	word	
		X position :	word	
		Y position :	word	
		Buffer address :	word	
11	Text	Text attribute :	A	See Note # 4
		X position :	HL	
		Y position :	DE	
		Text length :	IX	
		Text buf. addr. :	IY	
12	Set Color	Color number :	HL	Ordinal value (0..7) for current color
13	Set Line Pattern	Pattern :	HL	16 bit pattern
14	Set Fill Pattern	Pattern Address :	HL	Sets the current fill pattern
15	Hide Screen	none		Blanks current screen
16	Display Screen	none		Displays current screen
17	Fill Screen	Pattern :	HL	Pattern = 0, clear screen = 1, set to 1

Notes:

#1 The Coord. Array Address field used by both POLYLINE and POLYGON is actually a pointer to a variable length array (upto 255 maximum coordinate entries) with the following structure:

for POLYLINE:

```
Starting X      : word   coordinate of first line
Starting Y      : word
Ending   X(1)   : word
Ending   Y(1)   : word
  :      :        :
Ending   X(N)   : word
Ending   Y(N)   : word
```

N = number of vertices

for POLYGON:

```
Vertex #1, X coord.: word
Vertex #1, Y coord.: word
   :    :      :       :
Vertex #N, X coord.: word
Vertex #N, Y coord.: word
```

N = number of vertices

For a polygon call we assume that the figure is closed and that the line from vertex N to vertex #1 will be drawn.

#2 Primitive Graphic Routine Calling Convention:

For primitives number: 0,1,2,3,6,7,8,9,11,12,13,14,15,16,17

parameters are passed in the indicated register pairs in the parameter block definition column of the above table.

For primitives number: 4,5,10 (Arc, Pie Slice, Block)

HL - address of the parameter block; the structure of this block is dependent upon the particular graphic routine called.

All routines are accessed by passing the graphic routine number in register B ; register C has the XBIOS call value of 56.

#3 Fill Mode byte

The rectangle, circle, pie slice, and polygon primitves use a mode byte which indicates the following fill/boundary modes:

Bit Position:
- 0 - 0 = draw the boundary of the figure
  1 = DO NOT draw the boundary
- 1 - 0 = fill the figure with the current fill pattern
  1 = DO NOT fill the figure
- 2..7 - reserved for future use

#4 Text Attributes

Attributes for the text primitive are specified by bit positions in the attribute byte passed in register A. Following are the bit positions and their corresponding attributes. Setting a bit to 1 turns the atribute ON.

Bit position	Attribute
7	Reverse Video
6	Underline
5	(Unused)
4	Super-script
3	Sub-script
2	Style Font
1	Italics Font
0	Bold Version of a Font

(i.e. a value of 83h will allow the text to be written from the BOLD, ITALICS Font in REVERSE Video)

APPENDIX D. System Data Page

Entry	Offset	Bytes	Description
EYEBALL	00	20	Visual locator
RELEASE	21	1	Bios Release Level - 'B'
REVLVL	22	3	Bios Revision Level - '225'
ERREC	25	1	Disk Error Recovery 0 - disabled 1 - enabled
ERRDIAG	26	1	Disk Diagnostics 0 - disabled 1 - enabled
CLOCK	27	1	Date - Time Display 0 - disabled 1 - enabled
CRTWRAP	28	1	Auto Line Wrap 0 - disabled 1 - enabled
CBLINK	29	1	Cursor Blink 0 - disabled 1 - enabled
KEYBOARD	30	1	Connected Keyboard 1 - HASCI 1 2 - HASCI 2 3 - QX-16
TYPAMAT	31	1	Typamatic 0 - disabled 1 - enabled
IOBYTE	32	1	CP/M IOBYTE

Entry	Offset	Bytes	Description
SIO1BD	33	1	7201 Baud Rate 0 - none 1 - 50 2 - 75 3 - 110 4 - 134 5 - 150 6 - 300 7 - 600 8 - 1200 9 - 1800 10 - 2400 11 - 3600 12 - 4800 13 - 7200 14 - 9600
SIO1PRO	34	1	7201 Protocol 0 - none 1 - ETX/ACK 2 - RTS/CTS 3 - XON/XOFF
SIO1PAR	35	1	7201 Parity Bits 0 - none 1 - odd 2 - even
SIO1STP	36	1	7201 Stop Bits 0 - 1 1 - 1.5 2 - 2
SIO1DTA	37	1	7201 Data Bits 0 - 5 1 - 6 2 - 7 3 - 8
AUTO	38	1	Turnkey Cold Boot 0 - disabled 1 - enabled
DISPM	39	1	Display Unit Bit 7 Masking 0 - disabled 1 - enabled
SERM	40	1	Serial Port Bit 7 Masking 0 - disabled 1 - enabled

Entry	Offset	Bytes	Description
SHFTLOK	41	1	SHIFT LOCK Interpretation 0 - Caps Lock 1 - Shift Lock
FUNCTBL	42	1	Function Table Selector 0 - HASCI 1 - TVI-920 2 - Custom
RAMDISK	43	1	RAM Disk Active 0 - disabled 1 - enabled
PRINTER	44	1	Parallel Port Printer Handshake Support 0 - none 1 - undefined 2 - centronics 3 - FX-80 4 - FX-100 5 - RX-80 6 - RX-100 7 - MX-80 8 - MX-100 9 - LQ-1500 10 - Comrex CR-1 12 - Comrex CR-2 13 - Comrex CR-3
DSKFMT	45	1	Right Diskette Format 0 - QX10 - 380k 1 - QX10 - 300k 2 - MFCP/M 3 - IBM 1 4 - IBM 2
VIDTYP	46	1	CRT Driver enabled 0 - Character 1 - Bit Image
PARMASK	47	1	Parallel Port 7 bit masking 0 - disabled 1 - enabled
SERINT	48	1	Serial Port Interrupt Handler Enabled 0 - Polled Driver 1 - Interrupt Driver
SSHEET	49	1	Single Sheet Mode Support 0 - disabled 1 - enabled

Entry	Offset	Bytes	Description
SYSDRV	50	1	System Boot Drive 0 - Left Floppy <>0 - Hard Disk
SDPLOT	51	1	Screen Dump to Plotter 0 - Not Installed. <>0 - Installed. (Ramdisk disabled).
SETSD	52	1	Screen Dump Attribute Byte
INTLFONT	53	1	International Fonts Loaded 0 - Not Installed <>0 - Installed.
SCRNSAVER	54	1	Screen Saver 0 - disabled 1 - enabled
QXVER	55	1	QX-machine type 0 - QX-10 1 - QX-16
fill	56	1	not used
GABY1	57	1	Program disk assignment
GABY1	58	1	Data disk assignment
GABY2	59	1	Modem type
GABY4	60	1	Phone type
GABY5	61	1	GABY return drive
GABY6	62	1	GABY expert level
GABY7	63	1	GABY system warmboot string flag
DRVASG	64	16	Disk Drive Assignments at Cold Boot 1 - Left Diskette 2 - Right Diskette 3 - RAM Disk 4 - Logical Hard Disk 1 5 - Logical Hard Disk 2
CURASG	80	16	Current Drive Assigments
AUTOLEN	96	1	Cold Boot Turnkey String Length 0-24
AUTOBOOT	97	24	Cold Boot Turnkey String
VALID	127	1	System Data Page Initialized Byte

APPENDIX E. Console Control and Escape Sequences

E.1 TVI-920 CONTROL AND ESCAPE SEQUENCES

FUNCTION	CTRL/ESC SEQ	HEX CODE	CHR DRV	GRPH DRV
Bell	CTRL G	07	X	X
Carriage Return	CTRL M	0D	X	X
Cursor Left	CTRL H	08	X	X
Cursor Right	CTRL L	0C	X	X
Cursor Down	CTRL J	0A	X	X
Cursor Up	CTRL K	0B	X	X
Cursor Home	CTRL ^	1E	X	X
Address Cursor	* ESC = rc	1B,3D rc	X	X
Read Cursor	ESC ?	1B,3F	X	X
Linefeed	CTRL J	0A	X	X
Tab	CTRL I	09	X	X
Backtab	ESC I	1B,49	X	X
Clear Screen	CTRL Z	1A	X	X
Clear All To Space	ESC +	1B,2B	X	X
Character Insert	ESC Q	1B,51	X	X
Character Delete	ESC W	1B,57	X	X
Line Insert	ESC E	1B,45	X	X
Line Delete	ESC R	1B,52	X	X
Reverse Video	ESC j	1B,6A	X	X
End of Reverse Video	ESC k	1B,6B	X	X
Start Underline	ESC l	1B,6C	**	X
End Underline	ESC m	1B,6D	**	X
Start Blink Field	ESC ^	1B,5E	X	**
Start Blank Field	ESC	1B,20	X	**
End Blink/Blank	ESC q	1B,71	X	**
Erase EOL with Spaces	ESC T	1B,54	X	X
Erase EOP with Spaces	ESC Y	1B,59	X	X
Lock Keyboard	ESC #	1B,23	X	X
Unlock Keyboard	ESC "	1B,22	X	X
High Intensity Off	ESC (	1B,28	X	X
High Intensity On	ESC)	1B,29	X	X
New Line	CTRL _	1F		
Set Block Mode	ESC B	1B,42		
Set Conversation Mode	ESC C	1B,43		
Protect Mode On	ESC &	1B,26		
Protect Mode Off	ESC '	1B,27		
Set Column TAB	ESC 1	1B,31		
Clear TAB	ESC 2	1B,32		
Clear All TAB	ESC 3	1B,33		
TAB	ESC i	1B,69		

FUNCTION	CTRL/ESC SEQ	HEX CODE	CHR DRV	GRPH DRV
Send Line Unprotected to Cursor Position	ESC 4	1B,34		
Send Page Unprotected to Cursor Position	ESC 5	1B,35		
Send Whole Page Unprotected	ESC S	1B,53		
Send Line All to Cursor Position	ESC 6	1B,36		
Send Page All to Cursor Position	ESC 7	1B,37		
Send Whole Page	ESC s	1B,73		
Clear All to NULL	ESC *	1B,2A		
Clear Unprotected to NULL	ESC :	1B,3A		
Line Erase to NULL	ESC t	1B,74		
Page Erase to NULL	ESC y	1B,79		
Toggle Page	ESC K	1B,4B		
Load Cursor (PAGE,ROW,COLUMN)	ESC -	1B,2D		
Read Cursor (PAGE,ROW,COLUMN)	ESC /	1B,2F		
Auto Flip On	ESC v	1B,76		
Auto Flip Off	ESC w	1B,77		
Extension Port On	ESC @	1B,40		
Extension Port Off Page Print Mode On	ESC A	1B,41		
Print Page	ESC P	1B,50	X	X

X - Function implemented.

* NOTE: Direct cursor positioning is accomplished by the following escape sequence:

ESC = Row+20H Col+20H where Row = 1-24, Col = 1-80

** NOTE: These attributes will have no effect but will result in a space (blank) being written in their place. This will insure proper cursor positioning relative to the TVI-920.

E.2 QX-10 CONTROL AND ESCAPE SEQUENCES

FUNCTION	CTRL/ESC SEQ	HEX CODE	CHR DRV	GRPH DRV
Set Font	ESC X 1 n	1B,58,31,n		X
Load Font-character	ESC X 2 n	1B,58,32,n		X
Sub-script	ESC X 3	1B,58,33		X
Super-script	ESC X 4	1B,58,34		X
Normal-script	ESC X 5	1B,58,35		X
LED On	ESC X 6 n	1B,58,36,n	X	X
LED Off	ESC X 7 n	1B,58,37,n	X	X
~~Foreground-Color~~	~~ESC X 8~~	~~1B,58,38,n~~		~~X~~
~~Background-Color~~	~~ESC X 9~~	~~1B,58,39,n~~		~~X~~
Set Reverse Video	ESC X A	1B,58,41		X
Reset Reverse Video	ESC X B	1B,58,42		X
Set Underline	ESC X C	1B,58,43		X
Reset Underline	ESC X D	1B,58,44		X
Cursor on	ESC X E	1B,58,45	X	X
Cursor off	ESC X F	1B,58,46	X	X

X - Function implemented.

E.3 CONTROL AND ESCAPE SEQUENCE DESCRIPTIONS

Set Font - select a font from the set of available fonts

Escape Sequence	: ESC	X	1	N
Hex Codes	: 1B	58	31	00..06

N is the number of the selected font from the following set:

FONT	N
Standard Font	00h
BOLD Standard Font	01h
Italics Font	02h
BOLD Italics Font	03h
User defined Style Font	04h
BOLD Style	05h

Load Font-character - load an 8 X 16 pattern for a selected character in the user defined style font

Escape Sequence	: ESC	X	2	ordinal value	P1...P16
Hex Sequence	: 1B	58	32	00..127	-

Ordinal Value is the ASCII code for the character that is being defined.

P1 to P16 are 16 pattern bytes of the character.

PLEASE SEE GRAPHICS CHARACTER DEFINITION APPENDIX FOR FURTHER DETAILS.

Sub-Script - write all subsequent characters in subscript mode

Escape sequence	: ESC	X	3	
HEX Code	: 1B	58	33	

All subsequent characters are selected from a special half-height character font until the normal-script escape sequence is issued.

Super-Script - write all subsequent characters in super-script mode

Escape sequence	: ESC	X	4	
HEX Code	: 1B	58	34	

All subsequent characters are selected from a special half-height character font until the normal-script escape sequence is issued.

Normal-Script - change to the currently selected font

Escape sequence	: ESC	X	5	
HEX Code	: 1B	58	35	

If in Sub- or Super-script mode, will revert to normal mode. Subsequent characters are written from the currently selected font.

LED ON - turn the selected LED on

Escape sequence	: ESC	X	6	N
HEX Code	: 1B	58	36	01..08

N is the number of the selected LED; numbered from 1 to 8

LED OFF - turn the selected LED off

Escape sequence	: ESC	X	7	N
HEX code	: 1B	58	37	01..08

N is the same as for LED ON above

Set Reverse Video - all subsequent characters are written in reverse video mode

```
Escape sequence : ESC   X   A
HEX Code        : 1B    58  41
```

This is the mode attribute version of reverse video. All subsequent characters are written in reverse video until a reset reverse video escape sequence is issued.

Reset Reverse Video - cancels reverse video mode (if active)

```
Escape sequence : ESC   X   B
HEX Code        : 1B    58  42
```

All subsequent characters are written in standard mode

Set Underline - all subsequent characters are underlined

```
Escape sequence : ESC   X   C
HEX Code        : 1B    58  43
```

All subsequent written characters are underlined until a RESET Underline escape sequence is sent.

Reset Underline - cancels underline mode (if active)

```
Escape sequence : ESC   X   D
HEX Code        : 1B    58  44
```

All subsequent characters are written in standard mode.

Cursor On - allows the cursor to be displayed

```
Escape sequence : ESC   X   E
HEX Code        : 1B    58  45
```

Cursor is displayed

Cursor Off - cursor is not displayed

```
Escape sequence : ESC   X   F
HEX Code        : 1B    58  46
```

Cursor is not displayed.

APPENDIX F. HASCI-1 Keyboard Mapping

KEY	SCAN CODE	AL	AU	CL	CU	GL	GU	
STOP	73	F01	00	S05	00	00	00	
HELP	72	F02	F19	S06	00	00	00	
COPYDISK	71	F03	F20	00	00	00	00	
UNDO	01	F04	F21	00	00	00	00	
STORE	03	F05	F22	00	00	00	00	
RETRIEVE	04	F06	F23	00	00	00	00	
PRINT	05	F07	F24	S07	00	00	00	
INDEX	06	F08	F25	00	00	00	00	
MAIL	07	F09	F26	00	00	00	00	
EDIT	09	F10	F27	00	00	00	00	N/A HASCI-1
MENU	08	F11	F28	00	00	00	00	
CALC	0A	F12	F29	00	00	00	00	
SCHED	0B	F13	F30	00	00	00	00	
DRAW	0C	F14	F31	00	00	00	00	
BOLD	0E	F15	F32	S01	00	00	00	
ITALIC	0F	F16	F33	S02	00	00	00	
SIZE	1F	F17	F34	S03	00	00	00	
STYLE	1E	F18	F35	S04	00	00	00	
MAR REL	74	1B	1B	1B	1B	00	00	
^	75	BB	5E	00	1E	A0	AC	
! 1	76	31	21	11	01	A1	AD	
@ 2	61	32	40	12	00	A2	AE	
# 3	62	33	23	13	03	A3	AF	
$ 4	63	34	24	14	04	A4	A9	
% 5	64	35	25	15	05	A5	AA	
6	65	36	C4	16	00	E6	E4	
& 7	66	37	26	17	07	DD	DA	
* 8	67	38	2A	18	0A	ED	EA	
(9	68	39	28	19	08	FD	FA	
) 0	69	30	29	10	09	F6	F4	
_ -	6A	2D	5F	0D	1F	60	C6	
+ =	6B	3D	2B	1D	0B	7C	C7	
\	6C	5C	BD	1C	00	7E	C0	
¦X>	6D	08	08	08	08	00	00	

KEY	SCAN CODE	AL	AU	CL	CU	GL	GU
TAB	77	09	09	09	09	00	00
Q q	51	71	51	11	11	A6	B8
W w	52	77	57	17	17	A7	A8
E e	53	65	45	05	05	B0	B9
R r	54	72	52	12	12	B1	BA
T t	55	74	54	14	14	E2	E0
Y y	56	79	59	19	19	F2	F0
U u	57	75	55	15	15	E9	F9
I i	58	69	49	09	09	DE	DB
O o	59	6F	4F	0F	0F	EE	EB
P p	5A	70	50	10	10	FE	FB
	5B	BE	BF	00	00	D7	D8
[<	5C	3C	5B	1C	1B	7B	C2
] >	5D	3E	5D	1E	1D	7D	C3
TAB REL	41	00	00	00	00	00	00
SHIFT LOCK	42	00	00	00	00	00	00
A a	43	61	41	01	01	CC	CD
S s	44	73	53	13	03	B2	B3
D d	45	64	44	04	04	BC	B4
F f	46	66	46	06	06	B7	B6
G g	47	67	47	07	07	E3	E1
H h	48	68	48	08	08	F3	F1
J j	49	6A	4A	0A	0A	E8	F8
K k	4A	6B	4B	0B	0B	DF	DC
L l	4B	6C	4C	0C	0C	EF	EC
: ;	4C	3B	3A	1B	1A	FF	FC
" '	4D	27	22	07	02	D7	C1
RETURN	4E	0D	0D	0D	0D	00	00
TAB SET	78	00	00	00	00	00	00
SHIFT	86 87	00	00	00	00	00	00
Z z	33	7A	5A	1A	1A	CE	CF
X x	34	78	58	18	18	AB	B5
C c	35	63	43	03	03	D0	D4
V v	36	76	56	16	16	D1	D5
B b	37	62	42	02	02	D2	C5
N n	38	6E	4E	0E	0E	D3	D6
M m	39	6D	4D	0D	0D	E7	E5
, ,	3A	2C	2C	0C	0C	C8	CA
. .	3B	2E	2E	0E	0E	C9	CB
? /	4F	2F	3F	0F	0F	F7	F5
SHIFT	84 85	00	00	00	00	00	00
CTRL	8A 8B	00	00	00	00	00	00
SPACE BAR	32	20	20	20	20	20	20
GRPH SHIFT	8C 8D	00	00	00	00	00	00
CTRL	8E 8F	00	00	00	00	00	00

KEY	SCAN CODE	AL	AU	CL	CU	GL	GU	
<X¦	6E	7F	7F	7F	7F	00	00	
LINE	6F	0A	0A	0A	0A	00	00	
INSERT	5E	00	00	00	00	00	00	
WORD	5F	1E	1E	1E	1E	00	00	
UP ARROW	3C	0B	12	0B	12	00	00	
<--	3D	08	01	08	01	00	00	
-->	3E	0C	06	0C	06	00	00	
DOWN ARROW	3F	0A	03	0A	03	00	00	
DEC.TAB	2F	00	00	00	00	00	00	
DIVISION	2E	2F	2F	2F	2F	00	00	
X	2D	2A	2A	2A	2A	00	00	
-	1C	2C	2C	2C	2C	00	00	+/-
7	2B	37	37	37	37	00	00	
8	2A	38	38	38	38	00	00	
9	29	39	39	39	39	00	00	
+	2C	2B	2B	2B	2B	00	00	-
4	1B	34	34	34	34	00	00	
5	1A	35	35	35	35	00	00	
6	19	36	36	36	36	00	00	
=	28	3D	3D	3D	3D	00	00	+
1	27	31	31	31	31	00	00	
2	26	32	32	32	32	00	00	
3	25	33	33	33	33	00	00	
ENTER	15	0D	0D	0D	0D	00	00	ENTER =
0	17	30	30	30	30	00	00	
.	16	2E	2E	2E	2E	00	00	

LEGEND:

- AL - NO SHIFT KEYS
- AU - SHIFT
- CL - CONTROL
- CU - CONTROL SHIFT
- GL - GRAPH SHIFT
- GU - GRAPH SHIFT SHIFT

APPENDIX G. Extended Keyboard Support

G.1 System Keys

System Keys invoke a BIOS function to occur at the time the key is depressed. Initial System Key functions are:

```
CTRL BOLD     - Toggle Bold Font (Graphics Driver Only)
CTRL ITLALIC  - Toggle Italics Font (Graphics Driver Only)
CTRL SIZE     - Size Key (reserved for future use)
CTRL STYLE    - Toggle Style Font (Graphics Driver Only)
CTRL STOP     - System Pause
CTRL GRPH SHIFT STOP - Reset System (Cold Boot)
CTRL HELP     - System Status Display
CRL PRINT     - Screen Dump
```

The user cannot add or change System Keys. The System Key definitions are set at System Generation time. An application program can disable (and re-enable) a System Key in order to allow the key to be redefined as a function key so that the application can process it (a wordprocessor may wish to provide font capabilities and would require the font keys).

The VAR and OEM can redefine the System Keys described above as well as adding new System Keys. There is no upper limit.

E.2 Function Keys

Functions Keys differ from the normal set of keys in that when depressed a Function Key will insert a string of characters (as opposed to one character) in the keyboard buffer. Any key on the keyboard may be assigned as a Function Key. Tables within the keyboard support code designate a unique scan code and shift keys combination (shift, control, graph shift) as a Function Key. Three tables for function key mappings are provided:

HASCI - a QX-10 implementation which highlights QX-10 features and keyboard layout.

TVI920 - a TVI-920 implementation which provides function key definitions as defined by Televideo for programs requiring the TVI-920 mapping.

CUSTOM - a general function key table which the user can modify through SETUP or an application program. These modifications can be temporary or can be made permanent.

VAR's and OEM's have the ability to generate a baseline function key mappings that support their particular environment.

The user cannot modify the tables which define the particular scan codes that identify the function keys. However, the user can modify the character strings produced by the function keys.

As with System Keys, Function Keys can be enabled/disabled by an application program. Additionally, function keys can be protected such that modification by SETUP is disabled.

Following is the TVI-920 function key table.

```
;*****************************************************************
;
;               FUNCTION KEY SCAN CODE TABLE - TVI-920
;
;       BYTE 1 - SCAN CODE
;       BYTE 2 - ATTRIBUTES
;                BITS 0-2 - SCAN MASK
;       BYTES 3,4 - OFFSET INTO STRING TABLE
;
;*****************************************************************

;       TELEVIDEO-920 FUNCTION KEYS TABLE
;       UNSHIFTED FUNCTION KEYS

@FK920:
        DB      73H,00H
        DW      0 * FKOFFSET
        DB      72H,00H
        DW      1 * FKOFFSET
        DB      71H,00H
        DW      2 * FKOFFSET
        DB      01H,00H
        DW      3 * FKOFFSET
        DB      03H,00H
        DW      4 * FKOFFSET
        DB      04H,00H
        DW      5 * FKOFFSET
        DB      05H,00H
        DW      6 * FKOFFSET
        DB      06H,00H
        DW      7 * FKOFFSET
        DB      07H,00H
        DW      8 * FKOFFSET
        DB      09H,00H
        DW      9 * FKOFFSET
        DB      08H,00H
        DW      10 * FKOFFSET
        DB      0AH,00H
        DW      11 * FKOFFSET
        DB      0BH,00H
        DW      12 * FKOFFSET
        DB      0CH,00H
        DW      13 * FKOFFSET
        DB      0EH,00H
        DW      14 * FKOFFSET
        DB      0FH,00H
        DW      15 * FKOFFSET
        DB      1FH,00H
        DW      16 * FKOFFSET
        DB      1EH,00H
        DW      17 *FKOFFSET
```

```
;UNSHIFTED FUNCTION KEYS

        DB      72H,01H
        DW      18 * FKOFFSET
        DB      71H,01H
        DW      19 * FKOFFSET
        DB      01H,01H
        DW      20 * FKOFFSET
        DB      03H,01H
        DW      21 * FKOFFSET
        DB      04H,01H
        DW      22 * FKOFFSET
        DB      05H,01H
        DW      23 * FKOFFSET
        DB      06H,01H
        DW      24 * FKOFFSET
        DB      07H,01H
        DW      25 * FKOFFSET
        DB      09H,01H
        DW      26 * FKOFFSET
        DB      08H,01H
        DW      27 * FKOFFSET
        DB      0AH,01H
        DW      28 * FKOFFSET
        DB      0BH,01H
        DW      29 * FKOFFSET
        DB      0CH,01H
        DW      30 * FKOFFSET
        DB      0EH,01H
        DW      31 * FKOFFSET
        DB      0FH,01H
        DW      32 * FKOFFSET
        DB      1FH,01H
        DW      33 * FKOFFSET
        DB      1EH,01H
        DW      34 *FKOFFSET

FKOFFSET EQU    25                      ;OFFSET MULTIPLIER
```

```
;*********************************************************
;
;       PROGRAMMABLE FUNCTION KEY STRINGS
;
;       BYTE 1 - ATTRIBUTES
;               BIT 7   - ENABLED/DISABLED
;               BIT 6   - PROTECTED
;               BIT 5   - CARRIAGE RETURN DELIMITED
;               BIT 0-4 - STRING LENGTH (MAX = 24)
;       BYTES 2-25 - STRING
;
;*********************************************************

@FS920:
        DB      80H                             ;STOP
        DS      24
        DB      25H,'SETUP                  '   ;HELP
        DB      28H,'COPYDISK               '   ;COPYDISK
        DB      22H,01H,40H                     ;UNDO
        DS      22
        DB      22H,01H,41H                     ;STORE
        DS      22
        DB      22H,01H,42H                     ;RETRIEVE
        DS      22
        DB      22H,01H,43H                     ;PRINT
        DS      22
        DB      22H,01H,44H                     ;INDEX
        DS      22
        DB      22H,01H,45H                     ;MAIL
        DS      22
        DB      22H,01H,46H                     ;MENU
        DS      22
        DB      22H,01H,47H                     ;EDIT
        DS      22
        DB      22H,01H,48H                     ;CALC
        DS      22
        DB      22H,01H,49H                     ;SCHED
        DS      22
        DB      22H,01H,4AH                     ;DRAW
        DS      22
        DB      80H                             ;BOLD
        DS      24
        DB      80H                             ;ITALIC
        DS      24
        DB      80H                             ;SIZE
        DS      24
        DB      80H                             ;STYLE
        DS      24
```

```
;SHIFTED FUNCTION KEYS

        DB      80H                             ;HELP
        DS      24
        DB      80H                             ;COPYDISK
        DS      24
        DB      22H,01H,60H                     ;UNDO
        DS      22
        DB      22H,01H,61H                     ;STORE
        DS      22
        DB      22H,01H,62H                     ;RETREIVE
        DS      22
        DB      22H,01H,63H                     ;PRINT
        DS      22
        DB      22H,01H,64H                     ;INDEX
        DS      22
        DB      22H,01H,65H                     ;MAIL
        DS      22
        DB      22H,01H,66H                     ;MENU
        DS      22
        DB      22H,01H,67H                     ;EDIT
        DS      22
        DB      22H,01H,68H                     ;CALC
        DS      22
        DB      22H,01H,69H                     ;SCHED
        DS      22
        DB      22H,01H,6AH                     ;DRAW
        DS      22
        DB      80H                             ;BOLD
        DS      24
        DB      80H                             ;ITALIC
        DS      24
        DB      80H                             ;SIZE
        DS      24
        DB      80H                             ;STYLE
        DS      24

        DS      9       ;TAKE UP SPACE FOR 1K BOUNDARY ON DISK I/O
```

APPENDIX H. Optional Diskette Formats

FORMAT	DENSITY	SIDES/ DISK	TRKS/ SIDE	SECTORS/ TRACK	BYTES/ SECTOR	SYSTEM TRACKS
QX-10 380K	DOUBLE	2	40	10	512	2
QX-10 300K	DOUBLE	2	40	16	256	2
MFCPM	DOUBLE	2	40	16	256	4
IBMD	DOUBLE	2	40	9	512	2
IBMS	DOUBLE	1	40	8	512	2

The IBM formats above refer to CP/M-86 implementations.

APPENDIX I. Error and Status Messages

There is one printer error message described in Appendix I.1, and four disk error messages described in Appendix I.2. All situations have in common the fact that the User is prompted for certain decisions; namely, for the printer the User is prompted for a decision to: Correct the printer fault or "(A)bort/(I)gnore output until warmboot" and for the disk the User is prompted to select a decision from four alternatives "(R)etry/(A)bort/(C)ontinue/(I)gnore".

The effect of each decision is as follows:

(R)etry - Attempt I/O operation again.

(A)bort - Cancel current operation and program. Invoke a system warmstart.

(C)ontinue - Continue operation and return error code to BDOS.

(I)gnore - Alter disk I/O return code to indicate a sucessful I/O operation and return to BDOS.

(I)gnore output until warmboot - Ignore any printer output request until next system warmstart.

I.1 PRINTER ERROR AND STATUS MESSAGES

Following is the printer error and status message:

Printer Not Ready Correct fault or (A)bort/(I)gnore output until warmboot

Cause: The printer is either:

disconnected
connected but powered off
connected, powered on but offline
connected, powered on, online but out of paper

I.2 FLOPPY DISK ERROR AND STATUS MESSAGES

The following are the floppy disk error and status messages:

DRIVE X - SELECT ERROR: (R)etry/(A)bort

Cause: A non-existent drive has been selected.

DRIVE X - SEEK ERROR: (R)etry/(A)bort/(C)ontinue/(I)gnore

Possible Causes:
- a bad diskette
- corrupted data
- erroneous program instruction

DRIVE X - RD/WR ERROR: (R)etry/(A)bort/(C)ontinue/(I)gnore

Possible Causes:
- a bad diskette
- corrupted data
- the diskette has a write protect tab on it

DRIVE X - NOT READY: (R)etry/(A)bort/(C)ontinue/(I)gnore

Possible Causes:
- the 'PUSH' button on the disk drive has not been depressed
- a diskette has not been inserted in the selected drive
- the selected drive does not exist

Appendix J. Fonts Tables

The fonts table for the standard font is attached as follows.

```
;
; Standard Font
;
spc 00H,00H,00H,00H,00H,00H,00H,00H,00H,00H,00H,00H,00H,00H,00H,00H
!   08H,00H,00H,08H,00H,00H,00H,00H,08H,08H,08H,08H,08H,08H,08H,08H
"   00H,00H,00H,00H,00H,00H,00H,00H,6CH,6CH,24H,24H,00H,00H,00H,00H
#   24H,24H,24H,00H,00H,00H,00H,00H,00H,24H,24H,24H,7EH,24H,24H,7EH
$   48H,49H,3EH,08H,08H,00H,00H,00H,08H,08H,3EH,49H,09H,09H,3EH,48H
%   64H,92H,92H,61H,00H,00H,00H,00H,46H,29H,29H,16H,10H,08H,08H,04H
&   51H,21H,22H,5CH,00H,00H,00H,00H,0CH,12H,12H,0CH,04H,0AH,09H,51H
'   00H,00H,00H,00H,00H,00H,00H,00H,18H,18H,10H,08H,04H,00H,00H,00H
(   10H,10H,20H,40H,00H,00H,00H,00H,40H,20H,10H,10H,08H,08H,08H,08H
)   04H,04H,02H,01H,00H,00H,00H,00H,01H,02H,04H,04H,08H,08H,08H,08H
*   08H,00H,00H,00H,00H,00H,00H,00H,08H,49H,2AH,1CH,08H,1CH,2AH,49H
+   08H,08H,08H,00H,00H,00H,00H,00H,00H,00H,08H,08H,08H,08H,7FH,08H
,   0CH,0CH,08H,04H,02H,00H,00H,00H,00H,00H,00H,00H,00H,00H,00H,00H
-   00H,00H,00H,00H,00H,00H,00H,00H,00H,00H,00H,00H,00H,00H,3EH,00H
/   04H,02H,02H,01H,01H,00H,00H,00H,40H,20H,20H,10H,10H,08H,08H,04H
0   41H,41H,22H,1CH,00H,00H,00H,00H,1CH,22H,41H,41H,41H,41H,41H,41H
1   08H,08H,08H,08H,00H,00H,00H,00H,08H,0CH,0AH,08H,08H,08H,08H,08H
2   04H,02H,01H,7FH,00H,00H,00H,00H,1CH,22H,41H,41H,40H,20H,10H,08H
3   40H,41H,22H,1CH,00H,00H,00H,00H,1CH,22H,41H,40H,20H,1CH,20H,40H
4   20H,20H,20H,20H,00H,00H,00H,00H,20H,30H,28H,28H,24H,22H,21H,7FH
5   40H,41H,22H,1CH,00H,00H,00H,00H,7FH,01H,01H,01H,1DH,23H,40H,40H
6   41H,41H,22H,1CH,00H,00H,00H,00H,1CH,22H,41H,01H,01H,1DH,23H,41H
7   10H,08H,08H,08H,00H,00H,00H,00H,7FH,40H,40H,20H,20H,20H,10H,10H
8   41H,41H,22H,1CH,00H,00H,00H,00H,1CH,22H,41H,41H,22H,1CH,22H,41H
9   40H,41H,22H,1CH,00H,00H,00H,00H,1CH,22H,41H,41H,41H,62H,5CH,40H
:   00H,0CH,0CH,00H,00H,00H,00H,00H,00H,00H,00H,0CH,0CH,00H,00H,00H
;   00H,18H,18H,08H,04H,00H,00H,00H,00H,00H,00H,18H,18H,00H,00H,00H
<   04H,08H,10H,20H,40H,00H,00H,00H,40H,20H,10H,08H,04H,02H,01H,02H
=   00H,00H,00H,00H,00H,00H,00H,00H,00H,00H,00H,7EH,00H,00H,00H,7EH
>   10H,08H,04H,02H,01H,00H,00H,00H,01H,02H,04H,08H,10H,20H,40H,20H
?   00H,00H,08H,08H,00H,00H,00H,00H,1CH,22H,41H,41H,20H,10H,08H,08H
@   51H,51H,51H,2EH,00H,00H,00H,00H,1CH,22H,41H,40H,50H,5CH,52H,51H
A   41H,41H,41H,41H,00H,00H,00H,00H,1CH,22H,41H,41H,41H,41H,7FH,41H
B   41H,41H,21H,1FH,00H,00H,00H,00H,1FH,21H,41H,41H,21H,1FH,21H,41H
C   41H,41H,22H,1CH,00H,00H,00H,00H,1CH,22H,41H,41H,01H,01H,01H,01H
D   41H,41H,21H,1FH,00H,00H,00H,00H,1FH,21H,41H,41H,41H,41H,41H,41H
E   01H,01H,01H,7FH,00H,00H,00H,00H,7FH,01H,01H,01H,01H,3FH,01H,01H
F   01H,01H,01H,01H,00H,00H,00H,00H,7FH,01H,01H,01H,01H,3FH,01H,01H
G   41H,41H,62H,5CH,00H,00H,00H,00H,1CH,22H,41H,41H,01H,01H,71H,41H
H   41H,41H,41H,41H,00H,00H,00H,00H,41H,41H,41H,41H,41H,7FH,41H,41H
I   08H,08H,08H,08H,00H,00H,00H,00H,08H,08H,08H,08H,08H,08H,08H,08H
J   10H,11H,11H,0EH,00H,00H,00H,00H,10H,10H,10H,10H,10H,10H,10H,10H
K   09H,11H,21H,41H,00H,00H,00H,00H,41H,21H,11H,09H,05H,03H,03H,05H
L   01H,01H,01H,7FH,00H,00H,00H,00H,01H,01H,01H,01H,01H,01H,01H,01H
M   41H,41H,41H,41H,00H,00H,00H,00H,41H,63H,63H,55H,55H,49H,49H,41H
```

```
N   51H,61H,61H,41H,00H,00H,00H,00H,41H,43H,43H,45H,45H,49H,49H,51H
O   41H,41H,22H,1CH,00H,00H,00H,00H,1CH,22H,41H,41H,41H,41H,41H,41H
P   01H,01H,01H,01H,00H,00H,00H,00H,1FH,21H,41H,41H,21H,1FH,01H,01H
Q   41H,51H,22H,5CH,00H,00H,00H,00H,1CH,22H,41H,41H,41H,41H,41H,41H
R   21H,21H,41H,41H,00H,00H,00H,00H,1FH,21H,41H,41H,21H,1FH,11H,11H
S   41H,41H,22H,1CH,00H,00H,00H,00H,1CH,22H,41H,41H,02H,0CH,30H,40H
T   08H,08H,08H,08H,00H,00H,00H,00H,7FH,08H,08H,08H,08H,08H,08H,08H
U   41H,41H,22H,1CH,00H,00H,00H,00H,41H,41H,41H,41H,41H,41H,41H,41H
V   14H,14H,08H,08H,00H,00H,00H,00H,41H,41H,41H,41H,41H,22H,22H,22H
W   55H,55H,22H,22H,00H,00H,00H,00H,41H,41H,41H,41H,41H,49H,49H,49H
X   22H,22H,41H,41H,00H,00H,00H,00H,41H,41H,22H,22H,14H,08H,08H,14H
Y   08H,08H,08H,08H,00H,00H,00H,00H,41H,41H,41H,22H,22H,14H,14H,08H
Z   04H,04H,02H,7FH,00H,00H,00H,00H,7FH,40H,20H,20H,10H,10H,08H,08H
[   08H,08H,08H,08H,78H,00H,00H,00H,78H,08H,08H,08H,08H,08H,08H,08H
\   10H,20H,20H,40H,40H,00H,00H,00H,01H,02H,02H,04H,04H,08H,08H,10H
]   08H,08H,08H,08H,0FH,00H,00H,00H,0FH,08H,08H,08H,08H,08H,08H,08H
^   00H,00H,00H,00H,00H,00H,00H,00H,08H,14H,22H,41H,00H,00H,00H,00H
_   00H,00H,00H,00H,00H,0FFH,00H,00H,00H,00H,00H,00H,00H,00H,00H,00H
`   00H,00H,00H,00H,00H,00H,00H,00H,18H,18H,08H,10H,20H,00H,00H,00H
a   42H,42H,42H,0BCH,00H,00H,00H,00H,00H,00H,00H,3CH,42H,40H,7CH,42H
b   42H,42H,22H,1EH,00H,00H,00H,00H,02H,02H,02H,1EH,22H,42H,42H,42H
c   02H,02H,44H,38H,00H,00H,00H,00H,00H,00H,00H,38H,44H,02H,02H,02H
d   42H,42H,44H,78H,00H,00H,00H,00H,40H,40H,40H,78H,44H,42H,42H,42H
e   01H,41H,22H,1CH,00H,00H,00H,00H,00H,00H,00H,1CH,22H,41H,41H,7FH
f   08H,08H,08H,08H,00H,00H,00H,00H,30H,08H,08H,08H,3EH,08H,08H,08H
g   42H,64H,58H,40H,40H,44H,38H,00H,00H,00H,00H,58H,24H,42H,42H,42H
h   42H,42H,42H,42H,00H,00H,00H,00H,02H,02H,02H,3AH,46H,42H,42H,42H
i   08H,08H,08H,08H,00H,00H,00H,00H,08H,08H,00H,08H,08H,08H,08H,08H
j   10H,10H,10H,10H,10H,0CH,00H,00H,10H,10H,00H,10H,10H,10H,10H,10H
k   0AH,12H,22H,42H,00H,00H,00H,00H,02H,02H,02H,42H,22H,12H,0AH,06H
l   08H,08H,08H,08H,00H,00H,00H,00H,08H,08H,08H,08H,08H,08H,08H,08H
m   49H,49H,49H,49H,00H,00H,00H,00H,00H,00H,00H,36H,49H,49H,49H,49H
n   42H,42H,42H,42H,00H,00H,00H,00H,00H,00H,00H,3AH,46H,42H,42H,42H
o   41H,41H,22H,1CH,00H,00H,00H,00H,00H,00H,00H,1CH,22H,41H,41H,41H
p   42H,42H,26H,1AH,02H,02H,02H,00H,00H,00H,00H,1AH,26H,42H,42H,42H
q   42H,42H,64H,58H,40H,40H,40H,00H,00H,00H,00H,58H,24H,42H,42H,42H
r   04H,04H,04H,04H,00H,00H,00H,00H,00H,00H,00H,74H,0CH,04H,04H,04H
s   40H,40H,42H,3CH,00H,00H,00H,00H,00H,00H,00H,3CH,42H,02H,0CH,30H
t   08H,08H,08H,30H,00H,00H,00H,00H,08H,08H,08H,3EH,08H,08H,08H,08H
u   42H,42H,42H,0BCH,00H,00H,00H,00H,00H,00H,00H,42H,42H,42H,42H,42H
v   14H,14H,08H,08H,00H,00H,00H,00H,00H,00H,00H,41H,41H,41H,22H,22H
w   55H,22H,22H,22H,00H,00H,00H,00H,00H,00H,00H,41H,41H,49H,49H,49H
x   24H,24H,42H,42H,00H,00H,00H,00H,00H,00H,00H,42H,42H,24H,24H,18H
y   14H,14H,08H,08H,04H,04H,02H,00H,00H,00H,00H,41H,41H,41H,22H,22H
z   04H,02H,01H,7FH,00H,00H,00H,00H,00H,00H,00H,7FH,40H,20H,10H,08H
{   0H,10H,10H,10H,60H,00H,00H,00H,60H,10H,10H,10H,10H,10H,0CH,10H
|   08H,08H,08H,08H,08H,00H,00H,00H,08H,08H,08H,08H,08H,00H,00H,00H
}   04H,04H,04H,04H,03H,00H,00H,00H,03H,04H,04H,04H,04H,04H,18H,04H
~   00H,00H,00H,00H,00H,00H,00H,00H,06H,49H,30H,00H,00H,00H,00H,00H
```