UPDATE XVM UTILITY MANUAL

DEC-XV-UUPDA-A-D

XVM Systems

digital

UPDATE XVM
UTILITY MANUAL

DEC-XV-UUPDA-A-D

digital equipment corporation · maynard. massachusetts

First Printing, December 1975

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license and may be used or copied only in accordance with the terms of such license.

Digital Equipment Corporation assumes no responsibility for the use or reliability of its software on equipment that is not supplied by DIGITAL.

The postage prepaid READER'S COMMENTS form on the last page of this document requests the user's critical evaluation to assist us in preparing future documentation.

The following are trademarks of Digital Equipment Corporation:

DIGITAL	DECsystem-10	MASSBUS
DEC	DECtape	OMNIBUS
PDP	DIBOL	OS/8
DECUS	EDUSYSTEM	PHA
UNIBUS	FLIP CHIP	RSTS
COMPUTER LABS	FOCAL	RSX
COMTEX	INDAC	TYPESET-8
DDT	LAB-8	TYPESET-10
DECCOMM		TYPESET-11

CONTENTS

FIGURES

TABLES

LIST OF ALL XVM MANUALS

The following is a list of all XVM manuals and their DEC numbers, including the latest version available. Within this manual, other XVM manuals are referenced by title only. Refer to this list for the DEC numbers of these referenced manuals.

BOSS XVM USER'S MANUAL	DEC-XV-OBUAA-A-D
CHAIN XVM/EXECUTE XVM UTILITY MANUAL	DEC-XV-UCHNA-A-D
DDT XVM UTILITY MANUAL	DEC-XV-UDDTA-A-D
EDIT/EDITVP/EDITVT XVM UTILITY MANUAL	DEC-XV-UETUA-A-D
8TRAN XVM UTILITY MANUAL	DEC-XV-UTRNA-A-D
FOCAL XVM LANGUAGE MANUAL	DEC-XV-LFLGA-A-D
FORTRAN IV XVM LANGUAGE MANUAL	DEC-XV-LF4MA-A-D
FORTRAN IV XVM OPERATING ENVIRONMENT MANUAL	DEC-XV-LF4EA-A-D
LINKING LOADER XVM UTILITY MANUAL	DEC-XV-ULLUA-A-D
MAC11 XVM ASSEMBLER LANGUAGE MANUAL	DEC-XV-LMLAA-A-D
MACRO XVM ASSEMBLER LANGUAGE MANUAL	DEC-XV-LMALA-A-D
MTDUMP XVM UTILITY MANUAL	DEC-XV-UMTUA-A-D
PATCH XVM UTILITY MANUAL	DEC-XV-UPUMA-A-D
PIP XVM UTILITY MANUAL	DEC-XV-UPPUA-A-D
SGEN XVM UTILITY MANUAL	DEC-XV-USUTA-A-D
SRCCOM XVM UTILITY MANUAL	DEC-XV-USRCA-A-D
UPDATE XVM UTILITY MANUAL	DEC-XV-UUPDA-A-D
VP15A XVM GRAPHICS SOFTWARE MANUAL	DEC-XV-GVPAA-A-D
VT15 XVM GRAPHICS SOFTWARE MANUAL	DEC-XV-GVTAA-A-D
XVM/DOS KEYBOARD COMMAND GUIDE	DEC-XV-ODKBA-A-D
XVM/DOS READER'S GUIDE AND MASTER INDEX	DEC-XV-ODGIA-A-D
XVM/DOS SYSTEM MANUAL	DEC-XV-ODSAA-A-D
XVM/DOS USERS MANUAL	DEC-XV-ODMAA-A-D
XVM/DOS V1A SYSTEM INSTALLATION GUIDE	DEC-XV-ODSIA-A-D
XVM/RSX SYSTEM MANUAL	DEC-XV-IRSMA-A-D
XVM UNICHANNEL SOFTWARE MANUAL	DEC-XV-XUSMA-A-D

PREFACE

This manual describes the operation and use of the UPDATE XVM (UPDATE) Utility Program.

In this manual, it is assumed that the reader is familiar with the XVM/DOS system and the contents of the XVM/DOS User's Manual.

Applicable XVM/DOS manuals useful in understanding and using UPDATE are the following:

XVM/DOS Users Manual
XVM/DOS Keyboard Command Guide
Linking Loader XVM Utility Program Manual
MACRO XVM Assembly Language Manual

CHAPTER 1

INTRODUCTION

1.1 GENERAL DESCRIPTION

The UPDATE XVM (UPDATE) Program provides users with the ability to create, examine and revise the binary library files of the XVM/DOS Monitor System. A library file is defined as any file containing one or more relocatable binary programs. The XVM/DOS Monitor's .LIBR BIN files, which contain the FORTRAN IV Object Time System (OTS) routines, reside in the BNK and PAG System User File Directory (UFD).

UPDATE alters files by performing replacement, deletion, or insertion operations. It can also be used to create new library files, list the contents of libraries, and copy programs out of libraries for use as individual files. Figure 1-1 shows the functional relationship of UPDATE to its input and output files. Library creating and updating functions are performed by the creation of a new file from programs copied, on command, from either primary (original file) or secondary (new programs) input devices to the output device. The original file on the primary input device is never altered. A listing of the names of programs contained in the library can also be obtained.

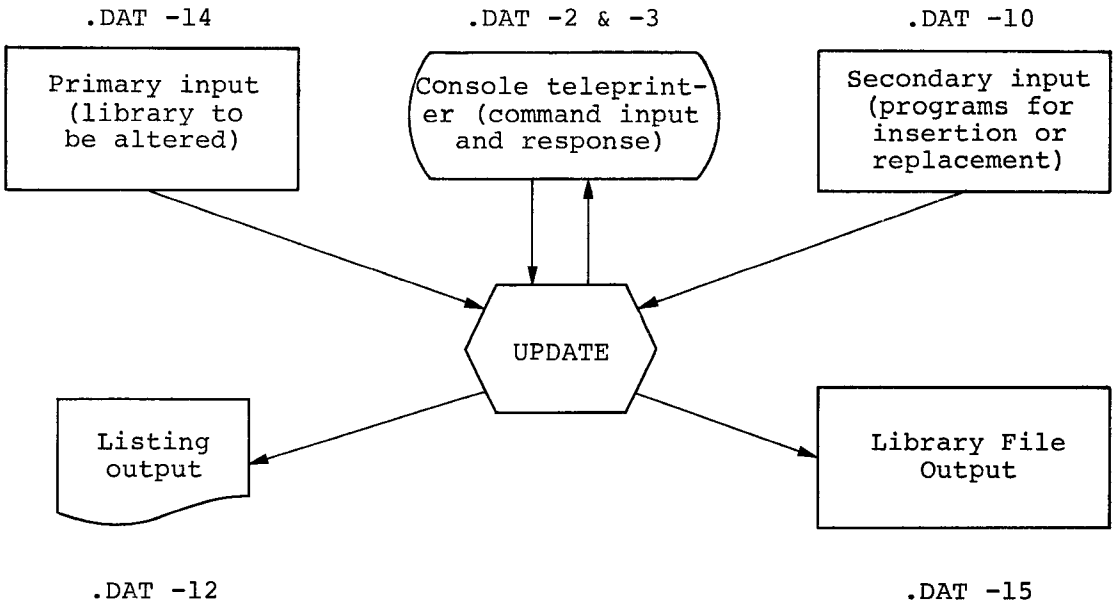

Figure 1-1
UPDATE's I/O Devices and Files

1.2 REFERENCE MATERIAL

The following manuals also contain information useful in understanding and using UPDATE:

XVM/DOS Users Manual
XVM/DOS Keyboard Command Guide
Linking Loader XVM Utility Program Manual
MACRO XVM Assembler Language Manual

1.3 SPECIAL SYMBOLS

Special symbols, when used in this manual, are defined as follows:

SYMBOL	REPRESENTS
↵	Carriage RETURN - LINE FEED operation
→|	TAB
␣	SPACE
[ ]	Optional Command String elements

CHAPTER 2

UPDATE COMMANDS AND OPERATIONS

2.1 LOADING PROCEDURE

2.1.1 I/O Device Assignments

As shown in Figure 1-1, UPDATE uses up to six .DAT slots. .DAT slots -2 and -3 are used by the console teleprinter and cannot be reassigned. The .DAT slots required by UPDATE vary with the type of operation being performed. For example, if a new library is to be created, only .DAT slots -10 (secondary input) and -15 (library file output) need be assigned. If a listing is also desired, then .DAT -12 must also be assigned. Similarly, if an update operation is to be performed where no programs are added and no listing is desired, then only .DAT -14 (primary input) and .DAT -15 (library file output) need be considered.

Because UPDATE uses the system macro .FSTAT to determine the existence or nonexistence of a named file, it is generally required that the A- version device handler be used, e.g., DTA. If only one file is to be open on DECtape, DTE or DTD may be used instead of DTA.

2.1.2 Calling UPDATE

Once the proper .DAT slot assignments have been made, UPDATE can be called by typing UPDATE↵ after the Monitor's $. When loaded, UPDATE identifies itself on the console teleprinter as follows:

```
UPDATE XVM Vnxnnn
>
```

and waits for a command from the user as described below.

2.1.3 CTRL Character Commands

UPDATE can be restarted at any time by typing CTRL/P. The current operation, of course, is aborted. CTRL/C also unconditionally terminates UPDATE operations but returns the user to Monitor control.

2.2 COMMAND DESCRIPTIONS

UPDATE's keyboard commands are divided into three functional categories as follows:

a. File Specification Commands
b. Manipulative Commands
c. Termination Commands

2.2.1 File Specification Commands

The first command to UPDATE each time it is loaded or restarted must be a file specification command. These commands specify the type of operations to be performed and the name of the file to be examined, modified, or created. The general form of the Command String format is:

OPTION DESIGNATOR ← FILENAME TERMINATOR

Sections 2.2.1.1, 2.2.1.2 and 2.2.1.3 describe the command string elements.

2.2.1.1 Options - The following paragraphs describe the options provided by UPDATE. Each option is specified in the command string by a single letter designator. Table 2-1 shows legal Option/Command combinations.

Table 2-1
Legal Option/Command Combinations

Options [1]	COMMANDS						
	INSERT	DELETE	REPLACE	FREE	END	CLOSE	KILL
U[LS]	X	X	X		X	X	X
N[LS]	X					X	X
G[LS]				X	X	X	X
L					X	X	X

[1]The S command is not logically useful by itself.

U Option — This option permits the updating of an existing library file by the deletion, insertion and replacement of the programs within it. The file to be upated resides on the device assigned to .DAT -14; programs to be inserted must be on .DAT -10; and the updated file is output to .DAT -15.

N Option — This option permits a new library file to be created from programs residing on the .DAT -10 device. The new library is output to .DAT -15.

S Option — The S Option causes UPDATE to remove the local symbol table from each program placed in the output file. This process shortens the length of a file by as much as 40%, thus conserving storage space and reducing library search time during loading. Non-library programs will occupy less memory when loaded with DDT XVM, since there will be no symbol table to load. (This is irrelevant for library programs because the Linking Loader and DDT do not load their local symbol tables. The single disadvantage of this option is that the removal of the symbol table precludes symbolic debugging with DDT and DDT's operation is then similar to DDTNS.

L Option — This option lists the contents of the new or updated library output to .DAT -15 on the device assigned to .DAT -12. If no other options are selected, a listing of the named library residing on the primary input device (.DAT -14) can be obtained. The format of the listing is shown in Figure 2-1. The first column of the listing contains the program name. The second column contains the source extension and is output only for programs assembled under the MACRO XVM Assembler (which outputs loader code 33_8 for this purpose). The third column specifies the size (in octal) of each program. The last column contains the command line echo of all Manipulative Commands issued, thus providing a complete log of the current updating operations.

G Option — This option primes UPDATE to accept the FREE Manipulative Command, which allows individual programs to be extracted from libraries as separate files.

LIBRARY FILE LISTING FOR file name PAGE n

PROGRAM NAME	SOURCE EXTENSION	PROGRAM SIZE	ACTION
NAME2	EXT2	34	REPLACE NAME1,NAME2
NAME2	EXT2	104	
NAME4	EXT4	423	INSERT NAME4

Figure 2-1
Format of Listing File Output

2.2.1.2 Filename - A filename consists of any combination of from 1 to 6 alphanumeric characters. The filename extension is always assumed to be BIN. If no filename is specified, the name .LIBR BIN is assumed by the program.

2.2.1.3 Command Terminator - Either a Carriage RETURN (↵) or an ALT MODE keyboard entry is used to terminate the file specification command. If a ↵ terminator is used, control is returned to the UPDATE program when work on the named file is complete. If an ALT MODE terminator is used, control is returned to the Monitor when updating of the named file is complete.

2.2.2 Manipulative Commands

After acceptance of a file specification command, UPDATE indicates its readiness to accept commands for the manipulation of the contents of the named file by typing a > .

In responding to the commands described below, UPDATE operates on library files in a sequential, program-by-program manner. Processing always begins at the beginning of the file (prior to the issuance of the first manipulative command). All programs in a file, which are not deleted or replaced, are copied into the output file automatically. The input file is never altered. The user should exercise care in issuing these commands, since UPDATE has no provision for "backing up" in a file to manipulate a program already passed by. The user, therefore, should first obtain a listing (L option followed by a CLOSE) of the file before he attempts to modify it.

The five manipulative commands, DELETE, REPLACE, INSERT, FREE and END, are described in Sections 2.2.2.1 through 2.2.2.5.

2.2.2.1 DELETE (D) Command - The D command causes the deletion of a named routine or a series of routines from the specified file. A series is specified by typing the names of the first and last routines of the series. The deletion is carried out by copying all the routine or file elements in the file up to the deleted routines into the output file and then positioning the input file just before the program which follows the last routine deleted.

Form: D[ELETE]␣prog1[,prog2]

The DELETE command may be used only when the U option has been requested in a file specification command.

2.2.2.2 REPLACE (R) Command - The R command causes programs in the original file to be replaced by new programs on the secondary input device. All of the programs in the original file, up to the element or routine to be replaced, are copied onto the output device. The new program(s) (replacement) is then copied onto the output file from a secondary input device. Upon completion, the output file is positioned immediately after the replaced routine.

If one program name is specified, UPDATE replaces the named program with a new program having the same name. If two program names are specified, the program with "name1" is replaced by the program with "name2".

Form: R[EPLACE]␣name1[,name2]

The REPLACE (R) command may be used only when the U option has been specified in a file specification command.

2.2.2.3 INSERT (I) Command - The I command directs UPDATE to insert programs contained on the secondary input device, at any point, into the original file as it is copied onto the output file device. On completion of an insert operation, the output file is positioned at a point immediately after the inserted program. If one name is used in the command, the named program is inserted into the file at the current position. If two names are specified, the "name1" program is inserted into the file after the program specified by "name2".

Form: I[NSERT]␣name[,name2]

The I command with two names is used only when the U option has been previously specified in a file specification command. The I command with only one name can be used when either the U or N option has been given.

2.2.2.4 FREE (F) Command - The F command directs UPDATE to free or extract a named program from the library file named in a G option file specification command. The extracted program is transferred to the output device as a file of the same name.

Form: F[REE]␣name

The F command is legal only when a G option has been previously specified in a file specification command.

2.2.2.5 END (E) Command - The END command causes the output file to be positioned at the end of the final routine of the original file. All remaining programs in the original file, beginning at its current position, are copied into the output file. The END command is a convenient method of positioning the file to be modified in order that new routines from the secondary input may be appended to the file via the INSERT (I) command.

Form: E[ND]

The END command is logically useful only when the U option has been specified.

2.2.3 UPDATE Termination Commands

2.2.3.1 CLOSE (C) Command - The C command is the normal terminator for any of UPDATE's operations. When used with the U option, it causes all of the remaining programs in the primary input file to be copied onto the output file starting at the current position of the file. When used with the N and L options, the C command closes the output file and completes the output of the listing file. If a file name is not specified with this command, the output file is given the name originally specified in the last file specification command.

Form: C[LOSE] ␣ [filname]

2.2.3.2 KILL (K) Command - The KILL command can be entered whenever UPDATE is awaiting command input and a file has been opened. When a new library file is being created (N option) or an old one updated (U option) and the KILL command is typed, the effect is as if all transactions prior to opening the file had not happened. Any insertions, deletions, or replacements are nullified. When a library file is opened, using the G option, and files are created from it using the FREE command, typing KILL does not delete the FREEd files, but it does close the library input file. To indicate that UPDATE has

restarted after a KILL command, it re-identifies itself on the terminal, i.e.,

```
UPDATE XVM Vnxnnn
>
```

and awaits further command input.

Form: K[ILL]

CHAPTER 3

ERROR CONDITIONS AND RECOVERY

UPDATE performs comprehensive error checking to assure the integrity of the input and output files as well as correctness of operator command strings. The program contains a large repertoire of messages to specifically identify the nature of the error. Table 3-1 lists all of UPDATE's error messages accompanied by an explanation of each and appropriate recovery procedures. The general rule for recovery, particularly when an incorrect option was selected or an unintentional end of library file occurred, is to type CTRL/P to restart UPDATE; then issue a new command string. IOPS messages may also be printed (e.g., IOPS4 for Line Printer).

Table 3-1
Update Error Messages

Message	Meaning	Recovery
FILE NOT FOUND COMMAND IGNORED	The file named in an R or I command could not be found on the directoried device assigned to .DAT -10.	Check that the device is properly assigned and that the desired file is contained therein.
END OF FILE FOUND FILE POSITIONING COMPLET, PGM UNIT NOT FOUND	The device assigned to .DAT -10 is nondirectoried. In attempting to perform an R or I command, UPDATE encountered either an end-of-file or an end-of-medium indicator in its first read to that device.	Check that the paper tape is correctly positioned in the input device. With paper tape, this error can easily occur if one forgets to load the paper tape reader.
EOF REACHED BY SEARCH COMMAND IGNORED	The program named in an F command was not found in the library.	Check that the name specified in the F command is correct and that the library does indeed contain the desired program (use L command to examine library).
END OF FILE FOUND	The program named in an F command did not contain a .END statement. The library is contaminated.	Obtain another copy of the library file and try again.
UNRECOVERABLE READ ERROR ON .DAT -1n - DYNAMIC KILL where, n=Ø or 4	A checksum, parity or buffer overflow error was detected (via the logical record header word 1) while reading from the specified input device.	Obtain another version of the offending file and try again.

Table 3-1 (Cont.)
UPDATE Error Messages

Message	Meaning	Recovery
NAMED FILE ALREADY ON OUTPUT DEVICE DO YOU WISH TO CONTINUE (Y/N)	The program name specified in an F or C command is the same as that of a file residing on the .DAT -15 (output) device.	If a Y is typed, UPDATE replaces the file on the current file. If Y is not typed, UPDATE will respond with COMMAND IGNORED. The user may then type any other command, presumably a CLOSE command with another file-name, or abort execution of UPDATE and return control to the Monitor. If the latter is done, the updated file will be present on the output device with the file-name extension WRK.
NOT A COMMAND	Unrecognizable command.	Retype the command.
VALID ONLY IN U MODE COMMAND IGNORED	A DELETE or REPLACE or an INSERT command (with 2 arguments) was issued prior to selecting the U option.	Issue a new command string and select the U option.
VALID ONLY WITH U OR N MODE COMMAND IGNORED	An INSERT command was issued but neither the U nor the N option had been selected.	Issue a new command string and select either the U or the N option as desired.
VALID ONLY IN GET MODE COMMAND IGNORED	A FREE command was issued but the G option had not been selected.	Issue a new command string and select the G option.

Table 3-1 (Cont.)
UPDATE Error Messages

Message	Meaning	Recovery
BUFFER OVERFLOW - DYNAMIC KILL	The current operation caused an overflow of UPDATE's internal (free memory) buffer.	Return to the Monitor and assign smaller device handlers to provide more free memory.
ILLEGAL COMMAND STRUCTURE COMMAND IGNORED	No program name specified after INSERT, DELETE, REPLACE, or FREE command.	Retype command using program name.
BAD SECONDARY INPUT -DYNAMIC KILL	A data error was detected during input from .DAT -10.	Locate and correct the error; then try again.
BAD DATA - NO PROGRAM NAME	The file being read from the .DAT -10 device does not contain a program name (loader code 23).	Check that the .DAT -10 device is the correct device and that the desired input file is there. If the above checks correctly, the file must be reassembled.
PROGRAM NAME MISSING - DYNAMIC KILL	In performing a search through the library on the .DAT -14 device, the end of file was reached but no program name (loader code 23) could be found.	Check that the .DAT -14 device is correctly assigned and that the desired input file is there. If the above checks correctly, then the library file is contaminated and must be reconstructed.

Table 3-1 (Cont.)
UPDATE Error Messages

Message	Meaning	Recovery
PROGRAM NAME DISCREPANCY FILENAME -- filename PROGRAM NAME -- program name SOURCE EXT -- source extension DO YOU WISH TO CHANGE INPUT (Y/N)	The file named in an I or R command, when the secondary input is from a nondirectoried device, is not the same as the program name (loader code 23).	If a "Y" is typed, UPDATE responds with "CHANGE INPUT AND ↑P". If N is typed, UPDATE responds with: "DO YOU WISH TO USE FILENAME (Y/N)". If Y is typed, the filename is used, otherwise the program name is used. In either case, UPDATE will type SOURCE EXT WANTED and expect the user to respond by typing in a filename extension. If, instead, the user types only a Carriage RETURN, the extension will be omitted from the program unit.
PROGRAM NAME DISCREPANCY FILENAME -- filename PROGRAM NAME -- program name SOURCE EXT -- source extension DO YOU WISH TO ACCEPT COMMAND (Y/N)	The file named in an I or R command when the secondary input is from a directoried device, is not the same as the program name (loader code 23).	If anything other than Y is typed, UPDATE responds: "COMMAND IGNORED". If Y is typed, UPDATE replies: "DO YOU WISH TO USE FILENAME (Y/N)". If a Y is typed the filename is used, otherwise the program name is used. In either case, UPDATE asks: "SOURCE EXT WANTED". The user may then type a filename extension followed by a Carriage RETURN. If the user types only a Carriage RETURN, the extension will be omitted from the program unit.

CHAPTER 4

EXAMPLES USING UPDATE

The following examples demonstrate typical command sequences using the various options and commands provided by UPDATE. User responses are underlined.

Example 1

To update a user created library called USRLIB:

```
UPDATE XVM V1A000
>U←USRLIB↵                  /FILE SPECIFYING COMMAND MUST BE FIRST
>I NAME2,NAME3↵             /INSERT ROUTINE NAME2 AFTER NAME3
>R NAME4,NAME5↵             /REPLACE ROUTINE NAME4 WITH NAME5
>D NAME6↵                   /DELETE ROUTINE NAME6 FROM FILE
>C↵                         /CLOSE USRLIB

UPDATE XVM V1A000
>
```

Example 2

To create an updated version of the system library (.LIBR) by replacing the BCDIO routine:

```
$A PR -10↵                  /ASSIGN PAPERTAPE READER TO
                            /SECONDARY INPUT
$UPDATE↵                    /CALL UPDATE

UPDATE XVM V1A000
>U←↵                        /SPECIFY "UPDATE" OPTION
                            /(NOTE: .LIBR BIN ASSUMED)
>R BCDIO↵                   /REPLACE BCDIO WITH NEW VERSION
                            /FROM PAPER TAPE READER
```

```
>C ↵                          /CLOSE THE FILE (.LIBR BIN)
                              /NEW .LIBR ON .DAT -15

UPDATE XVM Vnxnnn
>
```

To complete the update of a library file, it is necessary to use the PIP XVM Utility Program to delete the old library file from the .DAT -14 device and to transfer the new library from the .DAT -15 device to the .DAT -14 device.

Example 3

To remove all local symbols from the programs in an existing library and also obtain a listing:

```
UPDATE XVM Vnxnnn
>LS←MYNAM ↵                   /SELECT "LS" OPTIONS AND FILE MYNAM BIN
>C ↵                          /CLOSE TO COMPLETE OPERATIONS

UPDATE XVM Vnxnnn
>
```

Example 4

To create a new library called LIB13, delete all local symbols and obtain a listing:

```
UPDATE XVM Vnxnnn
>NLS←LIB13 ↵                  /SELECT OPTIONS "NLS" AND FILE LIB13 BIN
>I PROG1 ↵                    /INSERT PROG1
>I PROG2 ↵                    /INSERT PROG2 AFTER PROG1
>I PROG3 ↵                    /INSERT PROG3 AFTER PROG2
>C ↵                          /CLOSE TO COMPLETE OPERATIONS

UPDATE XVM Vnxnnn
>
```

Example 5

To obtain copies of the BCDIO and AUXIO routines from the file .LIBR BIN:

```
UPDATE XVM Vnxnnn
>G←┘                    /SELECT "G" OPTION FOR FILE .LIBR BIN
>F BCDIO┘               /EXTRACT BCDIO
>F AUXIO┘               /EXTRACT AUXIO
>C┘                     /CLOSE .LIBR BIN

UPDATE XVM Vnxnnn
>
```

INDEX

UPDATE XVM
Utility Manual
DEC-XV-UUPDA-A-D

READER'S COMMENTS

NOTE: This form is for document comments only. Problems with software should be reported on a Software Problem Report (SPR) form.

Did you find errors in this manual? If so, specify by page.

Did you find this manual understandable, usable, and well-organized? Please make suggestions for improvement.

Is there sufficient documentation on associated system programs required for use of the software described in this manual? If not, what material is missing and where should it be placed?

Please indicate the type of user/reader that you most nearly represent.

- ☐ Assembly language programmer
- ☐ Higher-level language programmer
- ☐ Occasional programmer (experienced)
- ☐ User with little programming experience
- ☐ Student programmer
- ☐ Non-programmer interested in computer concepts and capabilities

Name _____ Date _____

Organization _____

Street _____

City _____ State _____ Zip Code or Country _____

If you require a written reply, please check here. ☐

Please cut along this line.

---------- Fold Here ----------

---------- Do Not Tear - Fold Here and Staple ----------

FIRST CLASS
PERMIT NO. 33
MAYNARD, MASS.

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

Postage will be paid by:

digital

Software Communications
P. O. Box F
Maynard, Massachusetts 01754

digital

digital equipment corporation

PRINTED IN U.S.A.