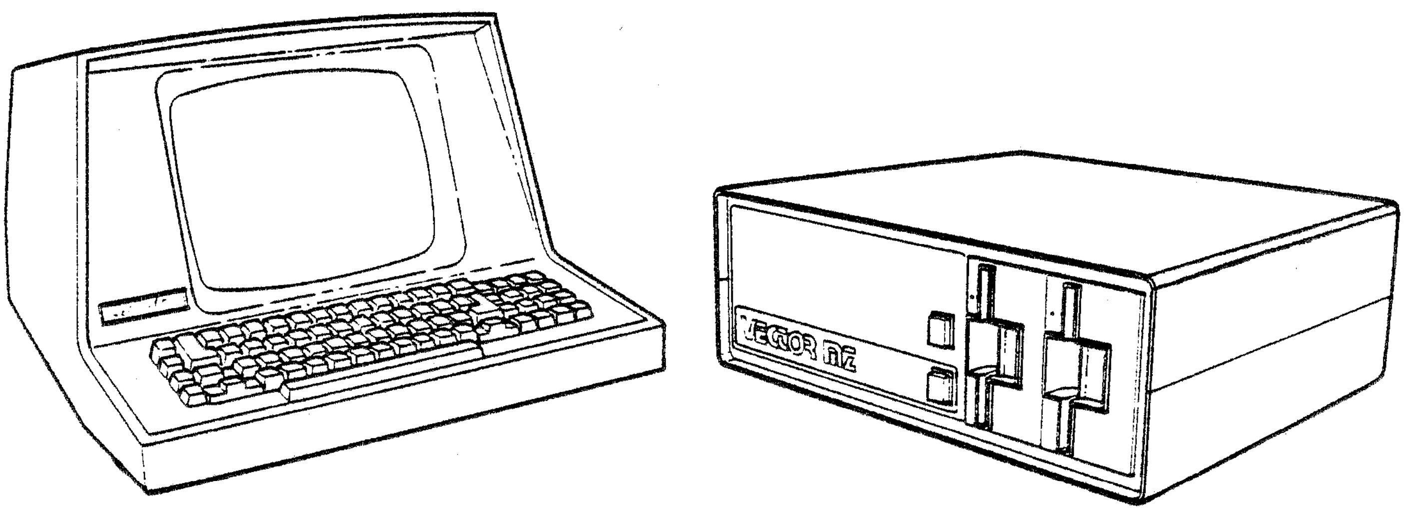
VECTOR
Time Share Multi-User System B

TECHNICAL
MANUAL



VECTOR
VECTOR GRAPHIC, INC.

VECTOR TIME-SHARE MULTI-USER SYSTEM B

CONVERSION PACKAGE

Version 3.3

TECHNICAL MANUAL

Revision A

June 24, 1980

Disclaimer

Vector Graphic makes no representations or warranties with respect to the contents of this manual itself, whether or not the product it describes is covered by a warranty or repair agreement. Further, Vector Graphic reserves the right to revise this publication and to make changes from time to time in the content hereof without obligation of Vector Graphic to notify any person of such revision or changes, except when an agreement to the contrary exists.

Revisions

The date and revision of each page herein appears at the bottom of each page. The revision letter such as A or B changes if the MANUAL has been improved but the PRODUCT itself has not been significantly modified. The date and revision on the Title Page corresponds to that of the page most recently revised. When the product itself is modified significantly, the product will get a new revision number, as shown on the manual's title page, and the manual will revert to revision A, as if it were treating a brand new product. EACH MANUAL SHOULD ONLY BE USED WITH THE PRODUCT IDENTIFIED ON THE TITLE PAGE.

Vector Time-Share Multi-User System B Conversion

TABLE OF CONTENTS

Section | Page

FOREWORD

Audience

This manual is intended for computer suppliers, or others with at least a moderate technical knowledge of small computers, and familiarity with the basic operation of the Vector Graphic system. It is NOT a training manual for end users.

Scope

This manual will describe how to convert a normal Vector System B to a multi-user system using the TMB Conversion package, and how to use the converted system.

I. PERSPECTIVE

Version 3.3 of the Time-Share Multi-User Conversion Package for the System B ("TMB" Conversion Package) enables a Vector System B to support from 2 to 4 terminals, where each is a memory-mapped Vector Mindless Terminal. Once the conversion is made, you may add terminals on as the need arises, up to the maximum of 4 all together. The Time-Share Multi-User conversion is done only once to any given system, and from that point on you may add additional terminals.

The strength of the Vector time-share method is that almost any operating system or application running on a single-user System B will run on a terminal in the multi-user System B without respect for what is running simultaneously on other terminals. For example, you may have the Memorite word processing software running on one terminal, a Microsoft BASIC program under the CP/M operating system running on another terminal, and a CP/M Utility program running on a third. Or, you may simply have four terminals running Memorite. In short, each terminal behaves essentially as if it were a system all to itself.

Conversion to time-share multi-user does reduce the amount of RAM available to each user. Instead of the 56K available in recent System B's, each terminal has 48K. This may be a limitation on some applications software, but all of Vector's operating systems and most applications and languages work in time-share systems as well as single-user 56K systems.

Vector CP/M 2 in its normal 48K version will work fine in a time-share system. CP/M 1.4 will not work, but Vector will supply CP/M 2 to replace it. Also, a special version of Memorite software must be used, and the same is true for MDOS. This is due to the other changes to the system as required by the conversion. Vector cannot guarantee that an operating system not supported by Vector Graphic will work at all. If you are not sure whether a given application or language will work, inquire before making the conversion.

Existing standard 48K and 56K System B's with Bitstreamer II boards can be converted to time-share.

There are three versions of the TMB Conversion Package. The first is called the TMB and is used for converting an existing 56K System B that has a 64K Dynamic RAM board. (All standard 56K System B's are shipped with 64K boards.) It is supplied on 2-2708 EPROMs. The second version is identical to the first except it is supplied on 1-2716 EPROM. The third version is called the TMBU and is used for converting any system that does not have a 64K board, including 48K systems, and 56K systems that have been upgraded from 48K systems using an 8K board. Each of the versions are available on 1-2716 2K PROM or 2-2708 1K PROMs. The contents of the packages are as follows:

TMB Conversion Package (P/N 42-7191) for 56K systems having a 64K RAM board and a PROM/RAM board or a ZCB board strapped for 1K (2708 type) PROMs.

TMB Monitor PROM's (2-2708's)
MDOS Diskette and manual appropriate for time-share
TMB Conversion Package Technical Manual

TMB Conversion Package (P/N 42-7193) for 56K systems having a 64K RAM board and a ZCB board strapped for 2K (2716 type PROMs.

TMB Monitor PROM (1-2716)
MDOS Diskette and manual appropriate for time-share
TMB Conversion Package Technical Manual

TMBU Conversion Package (P/N 42-7192) for systems not having a 64K RAM board:

Same as above, plus

One 64K Dynamic RAM board
CP/M 2 System Diskette and manual

The 64K board included in the second package will be used by the existing terminal in place of all memory boards currently in the system. This board is needed not for its memory capacity but because it has the bank select capability needed for time-sharing. The Flashwriter II board in the system will also be used by that terminal. As you can see, the TMB conversion packages do not include any additional terminals or the supporting hardware required for each terminal. Each additional terminal may be ordered from the following list. The Part Number of the Add-On Terminal varies with the length of the cable:

Part Number	Description
99-7190	TST with 6 ft. cable
99-7193	TST with 12 ft. cable
99-7194	TST with 18 ft. cable

Each TST unit contains the following:

1 Vector Mindless Terminal
1 Mindless Terminal Cable
1 Flashwriter II board
1 CPU/FW Wireboard P/N 75-1910
1 64K board, populated to 48K

In short, when completed, there must be one Flashwriter board, one 64K board, and one Mindless Terminal for every terminal in a multi-terminal configuration.

Carry out the conversion process at the same time that you install one or more additional terminals. At a later date, should you wish to install further terminals, return to this manual and follow that part of the conversion procedure which applies to adding terminals to the system.

II. INSTALLATION

2.1 Strapping

Turn computer power off and remove the top cover.

Remove all printed circuit boards from the mother-board. Be careful to note how each of the I/O cables is attached to each board, because you will have to return them later.

ZCB board

If your system is equipped with a ZCB board you must order the TMB package appropriate for your board. If your board has 2 P/ROMs in sockets U20 and U21, you should order P/N 42-7191. If your ZCB board has 1 P/ROM in socket U22, you should order P/N 42-7193. If your system has a Z-80 board, you may go directly to the next section on the PROM/RAM board.

Examine your ZCB board. Remove the 4.0 Monitor PROM(s) from their respective socket(s.) Replace with the PROM(s) supplied with the TMB conversion package. If you have the 2 PROM version, make sure to place the PROM labelled VTS 3.3 0 and VTS 3.3 1 in sockets U20 and U21 respectively. If you have the 1 PROM version, place it in socket U22.

Save the PROM(s) removed in case it is necessary to change the system back to its initial configuration.

Skip the PROM/RAM board instructions which follow and go directly to the section on the Micropolis Disk Controller Board.

PROM/RAM board

PROM's: TMB PROM's in sockets 8 and 9.
Address Jumpering: Block A (top row of PROM's) disabled, Block B at E000:

Explanation:

Examine the PROM/RAM board. Remove any PROM's in the top row of sockets. Remove the word processing "config" PROM's from sockets 9 and 11, if they are present. Remove <u>any</u> PROM's found in sockets 8 and 9.

This is all necessary because the word processing software for use in a TMB system does not use any PROM's, because the Extended Systems Monitor will be replaced by the TMB Monitor PROM's, and because the system will no longer have address space available for the top row of 8 PROM's. Sockets 10 and 11 can still be used for any purpose, however. Save all PROM's you removed, in case it is necessary to reconfigure the system back again.

Install the two TMB PROM's (labeled VTS 3.3 0 and VTS 3.3 1) in PROM sockets 8 and 9 respectively.

If the system was a 56K system already, the PROM/RAM board is already addressed properly. Otherwise, jumper areas are explained in the PROM/RAM board manual.

Micropolis Disk Controller Board

Addressed at F800.

If the system was 56K, the board will be addressed this way already. If not, you must readdress it as follows:

Find the four resistors above the number "4" printed at the bottom of the board. They are in the row of chips labeled "D" on the side of the board. Notice that below each resistor there are two jumper pads, one next to the left end and one next to the right end of the resistor. Then notice that there is a jumper connecting the top pair of pads and there is a jumper connecting the bottom pair of pads. Cut out the bottom jumper. Leave the top jumper in place.

Bitstreamer II Board

Implement the 55 Hz. real time clock by running a jumper from U4, pin 5 to jumper field "B", pads 1 and 2. Also run a jumper from jumper field "B", pad 3 to jumper field "A", pad 8.

Bitstreamer I Board (if in system)

Cannot be used with VTS 3.3. Must be replaced with Bitstreamer II board.

64K Dynamic RAM Boards

Disable the upper 16K block on all 64K boards which will be used in the system (1 per terminal). In Jumper area "D", cut the jumper from Pad 3 to Pad 5 (located on the rear of the board) and connect a jumper from pad 2 to pad 3.

Set the bank select switches on the 64K boards as follows:

1st board	0	
2nd board	1	
3rd board	2	(optional)
4th board	3	(optional)

Of course it does not matter which board you choose as 1st, and so on. To set the bank select switch (the red or blue dip-switch on the board) simply find the rocker having the desired number and press it down toward the number. Press all other rockers away from the desired number.

Flashwriter Video Display Boards

As with the 64K boards, there will be one Flashwriter Board per terminal in the system. The memory and keyboard ports on the Flashwriter boards must be jumpered as follows:

	Video Memory	Port Address	Jumper Area F	Jumper Area J	
1st board	F000	0,1	2-9	1-2	(Std. jumpering)
2nd board	C800	A,B	2-4	1-7	
3rd board	D000	C,D	2-5	1-8	(optional)
4th board	D800	E,F	2-6	1-9	(optional)

Cut away any pre-existing jumpers as necessary.

Check each board to insure that there is no jumper in Area L from pad 1 to pad 2. If you find such a jumper, cut and remove it. (This jumper was necessary to generate keyboard interrupts for the single user word processing software, but is not necessary in the multi-user configuration because interrupts are generated by the Bitstreamer II board.)

2.2 Mindless Terminal CPU/FW Wireboard (P/N 75-1910)

With each Mindless Terminal you will get a small interface circuit board having a DB-25 socket on one side, and three cables leading away from the board. The cables will be connected as follows:

cable	connected to
Flat ribbon cable	Flashwriter keyboard socket J1
5 wire bundle with 6 pin molex socket	Flashwriter video connector
3 wire bundle with lug connectors	Power supply

First, install each of interface boards on the back panel of the computer, with the DB-25 sockets pushed through the slots in the back panel. Use the provided screws to fasten them in place. If you are using four terminals, you may have to move the existing terminal interface board in the back panel in order to get them all on. A tight fit is expected. Work around any other connectors on the back panel, or move them if necessary. You may not however move any connections made to the rectangular molex connectors next to the fan on the back panel.

Connect the power supply cables from each of the Mindless Terminal interface boards to the computer power supply. It is very important that you connect the 3 colored wires to the correct locations in the computer power supply, because otherwise equipment damage can occur. The color coding, however, will eliminate any confusion. The colors in the power supply cable correspond with the colors of the wires used in the power supply. Connect the white wire ("ground") from each interface board to the screw having white wires leading from it on top of one of the capacitors. Since each capacitor has white wires leading from one of the two screws, it does not matter which capacitor is used.

Connect the red wire (+8V) from each interface board to the screw having red wires leading from it on top of one of the capacitors. In the same way, connect the purple wire (+18V) from each interface board to the corresponding screw on top of one of the capacitors.

2.3 Replacing the boards

Replace the boards in the following order in the mother-board, from front to back:

Disk Controller
CPU
PROM/RAM
Bitstreamer
64K boards
Flashwriter boards (as close as possible to the back panel)

It is important that the Disk Controller board be separated from the memory boards by several other boards.

Space the boards out as much as possible, except that the Flashwriter boards should be clustered together as close as possible to the back panel so that the interface cables can reach them.

Before inserting each of the Flashwriter Boards, connect one of the keyboard cables from one of the Mindless Terminal interface boards to the Flashwriter you are about to insert. The connector goes into the 3rd large socket down from the top of the board on the right side (J1.) The ribbon cable must point upward.

After all boards are in place, connect the video cable from each of the Mindless Terminal interface boards to the Flashwriter board already connected (via a keyboard cable) to the same interface board. Insert the 6-socket molex connector at the end of the video cable onto the 6-pin connector on top of the Flashwriter board, with the smooth side of the connector facing forward.

Replace all other cables in the system.

After double checking your connections, return the cover to the top of the system.

2.4 External connections

Plug each Mindless Terminal into one of the appropriate DB-25's at the rear of the computer, making sure you do not plug it into the RS-232C socket wired to the Bitstreamer board.

IMPORTANT: It does not matter which terminal is connected to which socket. All terminals in the multi-terminal system are completely equivalent.

III. OPERATION

3.1 Start up

When you turn the power on, a $ will appear at the top of each terminal. This is the TMB Monitor's prompt. Each terminal may now be operated as a separate system. For example, you may insert a CP/M diskette in the right-hand drive and press B on each of the terminals. The B may not appear on the screen immediately, but as quickly as possible, each of the terminals will boot up CP/M independently, using the same disk drive. You may now operate CP/M separately on each terminal.

For CP/M users: No special version of CP/M is needed. However, the CP/M diskette must be configured for a 48K system, not a 56K system. Therefore, if the system was previously a 56K system, you cannot use the 56K CP/M on the Personalized CP/M System Diskette, assuming that such a diskette was created when the system was installed. To make a new usable Personalized CP/M System Diskette, simply BACKUP a serialized CP/M diskette onto a fresh diskette. The CP/M on serialized diskettes is a 48K CP/M. You may then copy program and data files onto this diskette.

For MDOS and CP/M users: Two or more users may not be writing new material to the same diskette at the same time. One user must complete the write operation AND close the file, before another can write to the same diskette. Further, even if only one user is writing on a diskette, no other user may be attempting to query the same file until the first user closes the file. The other user may, however, query different files on the same diskette. These limitations arise from the fact that each user operates on the diskette independently, without having any way to inform the other of the effects of changes to the diskette until the file is closed.

The above will only be a problem when several users must access a common data base, because otherwise it is wiser to assign different users to different disk drives. If you do require a common data base, it is an excellent idea to schedule the main file update activity to a time when no other users are quering the file. Entry of new material can take place concurrently with query if the software is configured so that new material is placed in a separate temporary file on a differerent diskette, which is later merged into the main data base.

For MDOS and Memorite users: You can boot them up just like CP/M, but you must have the special versions of these operating systems for use in the Time-Share system.

For users of External Communications, available in 56K systems: The Monitor E command, allowing communications over an RS-232C port, is not available in a TMB system.

3.2 The Monitor Executive

When the $ prompt is on the screen, you can use the Monitor executive commands.

Do not press the RESET key on the front panel if you want to make use of the Monitor executive commands from one of the terminals, because the RESET key will return all of the terminals to the Monitor executive. Instead, use the command appropriate to the operating system running on that terminal to return to the Monitor executive. For example, in CP/M's executive, type MONITOR. In some operating systems, such as MDOS and MZOS, you will have to jump to the Monitor's warm-start address E01B.

The Monitor executive, though running out of PROM, is a re-entrant program, meaning that any of the terminals may be using it independently of the others. The commands available in the TMB Monitor are similar to those of the Extended Systems Monitor, version 3.1, except that commands E, K, R, and W do not exist in the TMB Monitor. They are as follows:

A <ADR1> <ADR2> - ASCII DUMP

Memory contents from ADR1 through ADR2 will be displayed as ASCII characters, i.e. the most significant bit of each 8-bit byte will be ignored, and values less than 20 which represent non-printing control characters will be printed as spaces. This command is useful for examining files such as those created by the lineditor, BASIC or MEMORITE. ASCII strings embedded in object code are easy to recognize.

B - JUMP TO BOOTSTRAP LOADER

Typing this command will cause a jump to location F800H which is the bootstrap loader. This will cause the operating system in the right-hand drive to be loaded into memory and to take control. (Applies to MDOS, CP/M, and Memorite, but not to MZOS, which can only be booted via Memorite.)

C <ADR1> <ADR2> <ADR3> - COMPARE BLOCKS

A byte-by-byte comparison will be made between the block of memory data starting at ADR1 and ending at ADR2 and a block of identical length starting at ADR3. The differences will be printed out with the address, the byte in the first block and the byte in the second block. This command is useful to compare two ersions of a program or to verify that proms have been programmed correctly.

D <ADR1> <ADR2> - DUMP IN HEX

Memory contents from ADR1 through ADR2 will be displayed as pairs of hexadecimal characters. The left character in each pair represents the four most significant bits of the memory location. The display may be halted and interrupted as described above.

F <ADR1> <ADR2> <BYTE1> <BYTE2> - FIND TWO BYTES

This memory range from ADR1 through ADR2 will be searched for the particular code combination BYTE 1 BYTE 2. This is useful for locating particular commands or jump addresses. For example, if you wish to change a control character (say control D) in a program you may try FE 04, which is CPI 04 since this is a common way of testing input characters. If you wish to find all locations that call or jump to a particular address, say C700, then search for 00C7. There is no guarantee that each location displayed is valid object code - it may be part of a data table, ASCII string, or second and third bytes of a three byte instruction.

G <ADR1> - GO TO AND EXECUTE

This command will cause a jump to ADR1 to execute a program or user subroutine. As with all Monitor jump commands, the address contained on the stack is the Monitor warmstart, so that if the user routine at ADR1 ends in "RET", program execution will return to the Monitor. Virtually unlimited stack space is available (up to 1K), but of course, pushing more registers on the stack than are popped will defeat the return feature with undesirable effects.

I <PORT> - INPUT FROM A PORT

Execution of this command will cause the CPU to execute an "IN PORT" instruction and the accumulator contents immediately following this to be displayed. This command is useful in checking out peripheral equipment. Only those ports used by the terminal, cassette interface, etc., will contain interesting values. All others will read FF since the data bus will be floating when the "IN" command is executed.

J - JUMP TO LOADED DOS

This command permits easy return to MDOS at 04E7H, or if not present the jump will be to 0000H, the CP/M warm start location.

M <ADR1> <ADR2> <ADR3> - MOVE MEMORY BLOCK

The data contained in memory starting at ADR1 and ending at ADR2 is moved to memory locations starting at ADR3. This command is useful for moving a program from a temporary storage location to its correct address. If there is an overlap of the two memory areas, interesting results are obtained. For example, M 6000 7BFF 6400 will cause the block of data from 6000 through 63FF to be repeated 8 times from 6000 through 7FFF, since by the time location 6400 is read, it has been overwritten with data from 6000. This is useful for bank programming of proms, or for creating repeating instruciton sequences for test purposes.

N - NON-DESTRUCTIVE MEMORY TEST

Memory locations starting at 0000 are read and the data temporarily stored. The memory location is then tested to see if 00 and FF can be written and read correctly. This continues after rewriting the original data until the first error is detected, whereupon the address is displayed followed by the data written into memory and what was read from it. This command is most useful for checking how much memory a system contains. For example, if the system contains 16K of memory, 4000 00 FF should be printed, indicating that there is no memory at address 4000. Since the test is non-destructive to data in memory, it can be used at any time. If there are only 4 terminals, then the reading will be correct, because there will be no Flashwriter board at C000.

O <PORT> <DATA> - OUTPUT TO PORT

The two hex digits "DATA" are loaded into the accumulator and the instrucion "OUT PORT" is executed. This command is useful for checking out peripheral equipment. For example, if a serial printer is used, 0 06 41 will cause an "A" to be printed, since 41 is the hex ASCII code for "A". The TMB Monitor will not allow output to port 40, since this is the bank-select port of the 64K RAM boards.

P <ADR1> - PROGRAM MEMORY

The contents of memory at ADR1 are displayed followed by a dash. If the space bar is depressed, no modification will be made to this location and the next location will be displayed. Typing two valid hex digits will cause that value to be substituted into memory and the next location to be displayed. The return key will only cause a CRLF to be echoed. To terminate the procedure, depress ESCape or type a non-valid hex character other than space or return.

Q <ADR1> <ADR2> - COMPUTE CHECKSUM

The MOD 256 checksum of memory contents in the address range specified is computed and displayed. This command is useful for checking proms or files to see if anything has changed. Any source file or program written in pure code (it does not write on itself) will have the same checksum as when it was loaded. While debugging assembly language programs, it is useful to be able to verify that a program being debugged has not written garbage in the source file or assembler.

S <ADR1> <ADR2> <BYTE> - SEARCH FOR SINGLE BYTE

This is similar to the "F" command, except that only one byte is searched for instead of two. An example of the use of this command is to display all locations in a program where an output to a port occurs (D3). The address of each location will be displayed followed by "D3" and the next byte (the port number).

T <ADR1> <ADR2> - TEST MEMORY

This is an extremely useful command, especially when first setting up a system. This command permits thorough testing of the system memory. A portion of a 64K byte pseudorandom number sequence is written into memory from ADR1 through ADR2, and the exact same sequence is regenerated from the initial point and compared with what is read from memory. If all locations compare, another portion of the sequence is used to repeat the test which continues until it is interrupted. Any memory errors are displayed with the address, what was written into memory and what was read from memory, respectively. This information is all that is needed to pinpoint a malfunctioning memory chip. This test is quite exhaustive if used for at least 10 cycles and is far superior to incrementing or complementing tests which may not reveal addressing problems. The only area of system memory that cannot be tested with this routine is the few bytes required for the stack and video flags in the vicinity of FFD0 on the PROM/RAM board.

U - JUMP TO 2B00

This command permits easy return to programs in the BDOS application area.

X <ADR1> <ADR2> <ADR3> - EXCHANGE MEMORY BLOCKS

A block of memory from ADR1 through ADR2 is exchanged with an equal length block starting at ADR3. This command is useful in comparing the operation ot two versions of a program, or for rapid switching of portions of a program without destroying the original. A loaded BASIC program can be exchanged with another if care is used to include the stack area (usually below the top of allowed memory).

Y - KEYBOARD ECHO

This command causes keyboard input to be echoed directly to the video driver and can be used for demonstration purposes. An ESCape returns to the Monitor.

Z <ADR1> <ADR2> <DATA> - ZERO OR FILL MEMORY

The memory block from ADR1 through ADR2 is filled with the byte "DATA". This is useful for setting memory to Zero. The end of a file or assembled program will stand out more clearly if memory is first zeroed. For test purposes, single instructions can be executed continuously so that bus waveforms are more easily interpreted. This is done by filling a block of memory with a repeated instruction sequence with a jump to the start of the block so that the program loops continuously.

3.3 Programming in the TMB system

A jump table at the beginning of the Monitor can be used to access several routines:

E000 - The normal cold entry point to the Monitor Executive, this is a jump to the initialization routine which initializes all terminals and initializes an 8251 USART through I/O port 7. The USART is set for an X16 band rate factor. This entry point returns all terminals to the Monitor executive. To return only the terminal you are on to the executive, use the E01B entry point.

E01B - Monitor restart point. Causes the terminal to return to the Monitor executive without reseting the whole system.

Other entry points are:

Locat.	Name	Function	Destroyed
E003	KEYTST	See if character is waiting in circular buffer. If character is ready, A=40H, Z=false. If character is not ready, A=0, Z=true.	A, PSW
E006	KEYDATA	Input data from circular buffer. A=input character. If not input exists, it will force a task swap until a character is input.	A
E009	CRT	Output a character in A to video terminal.	None
E00C	ESC	Test for ESC key depression. Causes a jump to the Monitor executive if so.	A, PSW
E00F	STATUS	Return information on task. A: Video cursor position, x coordinate B: Video cursor position, y coordinate C: Job # D,E: Video board address H,L: Not affected	A, BC, DE
E012	CTRLC	Test for a Control-C False: A=0 and Z=True True: A=3 and Z=False	A
E015	CLEARIN	Reset task's circular buffer to empty	BC, HL
E018	SWAP	Forces a switch to the next task. Upon return, execution begins at the location following the call.	None

Control-C

The control-C character has a special meaning to the KEYDATA routine. If it finds the control-C character in the circular buffer, it empties the buffer, places the control-C character in the buffer, and sets the flag in the control-C table. The programmer can thus sense a control-C keyboard depression either by normal input using KEYDATA, or by using the CTRLC routine. Use the CTRLC routine to check whether a control-C has been depressed, before inputing from KEYDATA, in order to prevent the normal control-C check from emptying the input buffer. The programmer will then have to clear the control-C from the buffer before proceeding.

Interrupts

Since the TMB Monitor uses interrupts to cause a job swap, the programmer must not disable interrupts for any significant amount of time. Some delay must be tolerated when using disks and other time-dependent peripherals, which have to disable interrupts for short periods, but the delay periods should be kept to the shortest time possible.

Disk files

When operating under TMB Monitor, each user is running in what appears to be his own system. Each user manipulates the disk independently of each user. When configuring an application using TMB, the programmer has the responsibility to insure that multiple tasks updating the same disk maintain disk integrity.

3.4 Video driver

The purpose of the video driver is to accept a stream of ASCII codes, and to write them into the screen memory in the proper place, interpreting certain non printing control codes in a special way.

Control codes can be sent to the video driver by higher level software, or can be generated by the keyboard by holding the control (CTRL) key down while a letter key is pressed. Control codes have values between 0 and 31, and are 64 less than the codes for the corresponding upper case letters. To demonstrate the features of the video driver, type Y after the Monitor prompt, and any keyboard generated code will be echoed to the video driver. The following control codes are interpreted as special functions, while all others are ignored:

(B) HOME THE CURSOR
(D) CLEAR THE SCREEN AND CURSOR HOME
(E) PRINT THE CODE IN B REGISTER
(H) NON DESTRUCTIVE BACKSPACE (also BACKSPACE)
(I) TAB OVER TO THE NEXT 8 MULTIPLE (also TAB)
(J) LINEFEED (also LF)
(M) CARRIAGE RETURN (also RETURN)
(N) NO CURSOR
(R) CURSOR DOWN
(T) TOGGLE REVERSE VIDEO
(U) CURSOR UP
(W) CURSOR LEFT
(Z) CURSOR RIGHT

The underline, or back arrow character, (5F) is a destructive backspace.

Experiment with the keys. There are special keys on the keyboard to generate some of the codes such as RETURN, TAB and linefeed (LF). If you are using the Vector Graphic Keyboard or Mindless Terminal, there are also keys for the cursor control and BACKSPACE. A few of the functions are not self explanatory. The underline or back arrow causes the characters to be erased as the cursor passes over them, in contrast to BACKSPACE or control Control H which merely moves the cursor back. A Control D sets the reverse video flag to normal in addition to clearing the screen and homing the cursor. A Control T will then toggle the reverse video flag from normal to reverse and back without printing on the screen.

In some cases it is desireable to print the symbol for a control code on the screen. This can be done in assembly language programs by putting the code for the symbol in the B register and calling the video driver with Control E (05) in A.

Control N requires the most explanation. This causes the cursor to disappear by reversing the video back to normal. Repeated Control N will cause the cursor to blink, and could be used in a keyboard input routine to generate a blinking cursor. A more important function is to allow random X Y positioning of the cursor by either BASIC or assembly language programs. After outputting Control N, the CURPOS location at FFDB can be modified to contain the new cursor X position, and LINENO at FFDC can be modified to contain the line number. Zero is at the far left and at the top of the screen respectively. Writing 32 and 12 in these locations would cause the next character to be written in character position 32 and line number 12 (the 13th line). Exceeding 79, 23 for the Flashwriter II or 63, 15 for the Flashwriter I will cause undesired results. The next printing character will restore the cursor, or this can be done with another Control N. If a cursor motion control is output before restoring the cursor, a second cursor will be left behind.

The video driver provides an extensive range of special controls, however, they must be incorporated into the software generating the video stream to be meaningful. For instance a piece of software that merely echoes all characters as they go into its input buffer will allow cursor motion on the screen, but this will probably be meaningless to the software.

3.5 Keyboard code conversion

Due to limitations in the keyboard encoder chip, the [] key on Vector Graphic keyboards is not encoded properly. The correct code is generated by a conversion routine in the Monitor's KEYBRD routine. The codes for backslash and tilde are also produced by the control and control shift mode of this key.

[] KEY CONVERSION:

MODE	KEYCODE	CONVERTED CODE	ASCII SYMBOL
unshifted	F1	5B	[
shifted	E1	5D	]
control	B1	5C	®
control shift	A1	7E	™

One of the cursor control keys is also converted from 60H to 15H which is interpreted correctly by the video driver.

IV. THEORY OF OPERATION OF TIME-SHARING

The TMB Monitor allows the computer to allocate its resources to multiple tasks. CPU allocation is on a timed basis, with each task receiving a sixteen millisecond time slice. Up to four tasks are supported, with each task occupying a block of main memory. To facilitate the addressing of the Z80 memory space, the bank select feature of the 64K Dynamic RAM board is used. Each task is executed in a separate memory board, which are selected on a round robin basis.

Every effort has been made to make the Supervisor function transparently to the user. Operation of the Vector computer remains unchanged for each of the terminals. Upon power up or system RESET, the Monitor's first function is to determine how many tasks will be active in the system. To perform this function, an initialization routine attempts to access memory at the predefined video terminal addresses. If memory is found at these addresses, the supervisor allocates a task corresponding to the video terminal. After locating all active tasks, the initialization routine causes each active task to execute the Monitor executive program. From this executive, the user is able to perform the various commands, including booting up an operating system.

Each task is allocated system resources on a round robin basis. There are two conditions which will cause the system to swap: an interrupt originating from the interval timer, or when the data input routine is called and there is no input to be returned. The latter allows the system to concentrate on tasks that are not I/O bound, that is, are not waiting for keyboard input. When either of these conditions exists, the Monitor saves all important CPU information in the current task's Machine State Save Stack, calls up the FW information for the next task, and enables the memory board for that task.