



OS9 Software User's Manual
for the
BVME370/380/390/395
MC68030/040 CPU BOARDS

Manual Revision A April 6, 1998

For BVME3XX BOOTROMs revision O or greater.

This material contains information of proprietary interest to BVM Ltd. It has been supplied in confidence and the recipient, by accepting this material, agrees that the subject matter will not be copied or reproduced, in whole or in part, nor its contents revealed in any manner or to any person except to meet the purposes for which it was delivered.

DISCLAIMER

The information in this document has been checked and is believed to be entirely reliable, however no responsibility is assumed for inaccuracies. BVM Ltd reserves the right to make changes and/or improvements in both the product and the product documentation without notice. BVM Ltd does not assume any liability arising out of the application or use of any product described herein; neither does it convey any licence under its patent rights or the rights of others.

SOFTWARE LICENCE NOTICE

Any software that is provided as Copyright BVM Ltd is proprietary and confidential property of BVM Ltd, and each single copy is given on the agreed understanding that it is licensed for use on product combinations supplied by BVM Ltd or their appointed distributors only. The software product may not be copied (except for backup purposes), given away, rented, loaned, reproduced, distributed or transmitted in any way or form, in whole or in part, without written permission of BVM Ltd.

This applies to any merged, modified or derivative version of the software including, but not limited to, versions produced by customising, translating, reverse engineering, decompiling or disassembly.

This licence may be automatically terminated without notice if any of its provisions are breached. Reasonable legal costs may be awarded to the prevailing party in connection with this licence agreement. Use of, or accepted delivery of these products shall constitute your acceptance of the provisions of this licence agreement.

WARRANTY

A) BVM Ltd warrants that the articles furnished hereunder are free from defects in material and workmanship and will perform as specified by BVM Ltd for one year after the date of shipment.

B) All warranties and conditions, express and implied, statutory and otherwise, as to the quality of the goods or their fitness for any purpose are hereby excluded and with the exception of liability for death or personal injury caused by negligence as defined in the Unfair Contract Terms Act 1977 the seller shall not be liable for any loss, injury or damage arising directly or indirectly from the use, application or storage of such goods.

C) Subclause (B) above shall not apply where the buyer deals as a consumer as this expression is defined in the Unfair Contract Terms Act 1977.

D) The liability of BVM Ltd hereunder shall be limited to repair or replacement at the manufacturers discretion of any defective unit. Equipment or parts which have been subject to abuse, misuse, accident, alteration, neglect, unauthorised repair or installation are not covered by this warranty. BVM Ltd shall have the right of determination as to the existence and cause of any defect.

E) The warranty period of the replacement or a repaired product or part shall terminate with the termination of the warranty period with respect to the original product or part for all replacement parts supplied or repairs made during the warranty period.

F) Although BVM Ltd offer a high level of technical support and advice, due to the complex nature and wide application of product configurations it is the responsibility of the purchaser to be satisfied at the time of purchase that the products are suitable for the final application.

G) The term 'Software' used herein is defined as 'any program data or code in source or binary format recorded in or on any readable device or media'.

H) BVM Ltd will effect all reasonable effort to resolve accepted reproducible software errors reported within 12 months of purchase. Acceptance of an error shall solely be based on conformance to supporting specifications. Proper operation of earlier releases is not guaranteed.

WARNINGS

Do not lever out the EPROMs from the CPU board. The board uses surface-mounted devices extensively, which can be fractured by excessive force. Use proper EPROM extraction and insertion tools. Damage may result if users attempt to remove or fit EPROMs incorrectly.

Do not lever out memory modules from the CPU board. The board uses surface-mounted devices extensively, which can be fractured by excessive force. Memory modules are not a field-fit option, however it may be necessary to remove the memory module in order to fit 32-pin EPROM devices. Damage may

result if users attempt to remove or fit memory modules incorrectly.

Do not fit/remove the CPU device to/from the CPU board. Special tools are required to fit and remove these devices. Return the board to the factory if the device requires changing. Damage may result if users attempt to fit or remove the device. The CPU board uses devices sensitive to static electricity. Ensure adequate static electricity precautions are observed when handling the CPU board, EPROMs, and memory modules.

Copyright © 1994 by BVM Ltd.

OS-9 is a registered trademark of Microware Systems Corporation.

Table Of Contents

1. Introduction	1
2. Board Configuration	2
2.1. Link Settings	2
3. OS9 Bootstrap	3
3.1. Bootstrap Menu	3
3.2. Bootstrap Reconfiguration	4
4. OS9 Resource Usage	5
4.1. Memory	5
4.2. Serial I/O	5
4.3. Parallel I/O	5
4.4. EEPROM	6
4.5. Real Time Clock/Timer	6
4.6. Disk I/O	6
4.7. Ethernet I/O	7
4.8. MC68040 MMU/CACHE/FPU	7
4.9. MC68030 MMU/CACHE/FPU	8
4.10. ABORT Handling	8
5. Other Software Support	9
5.1. Networking	9
5.2. TargetLink	9
5.3. MEM400 Flash Memory	9
Appendix A - Configuration EEPROM	10

1. Introduction

The BVME3XX is a versatile high-performance CPU board range using the Motorola MC68030/040 processor family. The board provides all of the commonly required facilities of an OS9 processing node for a diverse range of applications, and can be used in conjunction with a wide range of VMEbus boards to add features such as disk and network support.

The BVME370 range utilises the MC68EC030 CPU (no MMU), the BVME380 range utilises the MC68030, the BVME390 range utilises the MC68EC040 (no MMU or FPU) or the MC68LC040 (no MMU), and the BVME395 utilises the MC68040.

The BVME3XX is complimented by a full OS9 software port providing software facilities for the on-board devices. Both PROFESSIONAL and INDUSTRIAL OS9 versions are available, as well as customised versions including other OS9 functions.

Support is provided for two asynchronous serial I/O channels, a real time clock, tick timer, SCSI and floppy disk (with additional BVME130 board) and Ethernet (with additional BVME410 board) interfaces, and various other BVM and Microware software packages.

This manual provides details on the system configuration for running OS9 on the BVME3XX, details on the OS9 booting options, the BVME3XX resources required by OS9 and details on the facilities provided by the different software packages.

For further information on the hardware details of the BVME3XX range, refer to the BVME370/380 User's Manual: P/N 454-68380 and the BVME390/395 User's Manual: P/N 454-68390.

For further information on the hardware details of the memory module range for the BVME390/395, refer to the MEM390 User's Manual: P/N: 454-68391 and the MEM400 User's Manual: P/N 454-61400.

For further information on the hardware details of the BVME130 Floppy/SCSI board, refer to the User's Manual: P/N 454-48130 and for the BVME410 Ethernet board refer to the User's Manual P/N: 454-48414.

2. Board Configuration

2.1. Link Settings

ABORT SWITCH: the ABORT switch is used to call the bootstrap ROM debugger if LK1 (370/380) or LK4 (390/395) is fitted.

RESET SWITCH: the RESET switch will reset the board if LK7 (370/380/390/395) is fitted.

VMEbus RESET: the RESET switch will reset the VMEbus if LK6 pins 1 to 2 (370/380) or LK2 (390/395) is fitted, the VMEbus will reset the board if LK6 pins 3 to 4 (370/380 only) is fitted.

CACHE INHIBIT: normally the CPU's caches are enabled, but for debugging purposes the caches may be disabled by fitting LK9 (370/380) or LK8 (390/395).

EPROM SIZE: normally with 27C512 EPROMs, LK2 pins 2 to 3, 5 to 6, 7 to 8, 10 to 11 (370/380) or LK6 pins 1 to 2 (390/395) should be fitted (see relevant User's Manual).

3. OS9 Bootstrap

3.1. Bootstrap Menu

The OS9 port for the BVME3XX offers a variety of booting methods, described in the following table. Each bootstrap option has a "boot driver" number, used in the "auto-booting" priority setup, and an option name used for menu driven booting.

BOOT NUMBER	OPTION NAME	BOOT TYPE	DESCRIPTION
1	ro	ROM	locate the OS9 kernel and other modules in the on-board ROMs and execute the kernel.
2	ra	ROM to RAM	locate the OS9 kernel and other modules in the on-board ROMs, copy them all into RAM and execute the kernel in RAM.
3	rd	RAMDISK	read an OS9 bootfile from a ramdisk in the on-board SRAM (BVME390/395) or VMEbus SRAM at \$900000 (BVME370/380), locate and execute the OS9 kernel.
4	hd	SCSI HARD DISK	via a BVME130: read an OS9 bootfile from a SCSI CCS disk drive (synchronous or asynchronous) with a specified SCSI ID (default 2), locate and execute the OS9 kernel.
5	fd	FLOPPY DISK	via a BVME130: read an OS9 boot from floppy disk drive with drive select zero, locate and execute the OS9 kernel. Supports 3½" and 5¼" 1Mb drives, and TEAC 3½" 2Mb and 4Mb drives.
6	ft	SCSI TEAC FLOPPY	via a BVME130: read an OS9 boot from a 3½" SCSI floppy disk drive with a TEAC FC1 controller, with a specified SCSI ID (default 1), locate and execute the OS9 kernel. Supported drives are TEAC FD235HS-700 (2Mb) or FD235JS-500 (4Mb).
7	td	SCSI TAPE DRIVE	via a BVME130: read an OS9 bootfile from a SCSI TAPE with a specified SCSI ID (default 7), locate and execute the OS9 kernel. Supported drives are TEAC MT2ST or ARCHIVE VIPER.
8	bp	BOOTP	via a BVME410: read an OS9 bootfile from an INTERNET BOOTP server node with a matching Ethernet address entry, locate and execute the OS9 kernel.

There are two types of bootstrap ROMs available for the BVME3XX. For INDUSTRIAL OS9, only the bootstrap options "ro", "ra" and "rd" above are included to keep the bootstrap code size small. For PROFESSIONAL OS9 (or INDUSTRIAL+RBF OS9) all of the bootstrap options above are included.

The bootstrap menu is displayed initially if the EEPROM holds invalid data. Thereafter, it is entered by resetting the CPU board 5 times in quick succession (but allowing time for the bootstrap message to appear), or if it is selected from the menu itself.

The bootstrap menu allows booting directly from the menu for any of the above options. For those options without a SCSI ID, just type the option name. For those options requiring a SCSI ID, type the option name followed by the required SCSI ID, or type the option name alone, in which case the default SCSI ID shown above will be used.

The bootstrap menu also includes the "rc" reconfiguration option. This allows the required booting options to be stored in the configuration RAM for auto-booting, and other BVME3XX system parameters to be setup, see "3.2. Bootstrap Reconfiguration".

3.2. Bootstrap Reconfiguration

The bootstrap reconfiguration menu allows the BVME3XX system parameters to be set to individual user's requirements. In addition to the bootstrap options specified in "3.1. Bootstrap Menu", a number of other parameters may also be set up, as follows:

1. DEFAULT - the default settings are setup. See individual entries below for default values.
2. DEBUGGER - the debugger in ROM may be enabled or disabled. If enabled, when the system debugger is called whilst OS9 is running (e.g. by a system failure, ABORT switch, "break" command etc.), the debugger in the ROMs (if any) will be called. If disabled, when the system debugger is called directly the system will reboot, and if called from ABORT, it will be ignored (see "4.10. ABORT Handling"). DEFAULT: ENABLED.
3. BOOT STAGING - the initial debugger call may be enabled or disabled. If enabled, the debugger will be called immediately after reset. If disabled, the debugger will NOT be called immediately after reset. DEFAULT: DISABLED.
4. AUTOMATIC MENU - the bootstrap menu can be displayed at every bootstrap occurrence by enabling this option. DEFAULT: DISABLED.
5. BAUD RATE - the bootstrap ROM baud rate can be selected to be 9600, 19200 or 38400 baud. Note that this does not select the OS9 driver baud rate. DEFAULT: 9600.
6. ADDITIONAL BOOT RAM - normally the bootstrap ROM allocates only the first 512Kbytes of memory for bootstrap purposes. However if this is not enough, an additional 512Kbytes, 1.5Mbytes or 3.5Mbytes can be allocated. Note that this operates independently from the OS9 memory list. DEFAULT: NONE.
7. INITIATOR SCSI ID - the SCSI ID for a BVME130 as a SCSI initiator can be setup. Normally this is set to 0 for BVM supplied software. DEFAULT: 0.
8. BOOT DRIVERS - the "auto-booting" priority can be setup using the boot driver numbers. If the number denotes a SCSI boot driver, the SCSI ID must be appended to the boot driver number e.g. **42 5 1**: SCSI Hard Disk ID 2; Floppy Disk; ROM. DEFAULT: the default booting order, in order of priority is as follows for the full bootstrap ROM:

Boot from SCSI Hard Disk ID 2; Boot from Floppy Disk (drive select 0); Boot from ROM.

For the smaller bootstrap ROM, the default value is to boot from ROM only.

The above parameters are stored in the EEPROM (see "Appendix A - Configuration EEPROM"), and are referenced by the bootstrap software. The RAM is checksummed, and will be cleared if the checksum is invalid.

4. OS9 Resource Usage

4.1. Memory

During the bootstrap phase, the bootstrap software expects to find RAM at address \$80000 in the address map, and expects to find ROM at address 0 in the address map. The first 512Kbytes, 1Mbytes, 2Mbytes or 4Mbytes (bootstrap menu selected) of RAM is scanned for memory by the bootstrap software, and up to 512Kbytes of ROM is scanned for valid OS9 modules. Any further ROM or RAM can be scanned by the OS9 kernel acting on the memory list found in the "init" module.

For the BVME370/380 the first 1 or 4Mbytes of the address space from \$80000 are scanned for on-board RAM, and the next 7.5 or 4.5Mbytes up to \$8FFFFFF can be scanned for VMEbus A24:D16 memory.

For the BVME390/395 all 8 of the on-board memory pages are scanned for RAM and ROM (from \$nn080000 to \$nn880000), and from \$FE000000 to \$FE900000 can be scanned for VMEbus A24:D16 memory. The second page of the ROM can also be scanned from \$01000000 to \$01080000.

The top 7Mbytes of VMEbus A24:D16 is normally reserved for memory-mapped I/O boards. The "init" module needs to be modified to enable VMEbus (and additional ROM) memory scanning, and conditional equates are included in the "systype.d" definitions file for this purpose.

The BVME390/395 SRAM can be used as a non-volatile ramdisk at address \$FC000000. An "r1" device descriptor is supplied for this purpose, and this ramdisk can also be used to store a bootfile for the bootstrap software.

The BVME370/380 can be used with VMEbus SRAM (e.g. using a BVME065) to provide a non-volatile ramdisk at address \$900000. An "r1" device descriptor is supplied for this purpose, and this ramdisk can also be used to store a bootfile for the bootstrap software.

4.2. Serial I/O

During the bootstrap phase, the "console" serial I/O is via the BVME3XX serial channel A, which is connected through JP1 (lower connector). The same serial channel is also used for I/O whenever the system debugger is called. When operating in this mode, the serial channel runs at 9600, 19200 or 38400 baud (bootstrap menu selected), 8 data bits, 1 stop bit, no parity, and is polled I/O (non-interrupt).

When OS9 is running, the "console" I/O is via the same serial channel as above, using the "sc3XX" device driver and "term" device descriptor under interrupt control. The standard settings are as above, but may be altered by changing the device descriptor. Secondary serial I/O is via the BVME3XX serial channel B, which is connected through JP2 (upper connector). The OS9 device driver is also "sc3XX", using the "t1" device descriptor, also under interrupt control.

The "sc3XX" device driver uses the MC68681 serial controller device resources, which should not generally be used for any purposes other than alternative BVM serial I/O packages such as BVM serial networking for NFM, PCLINK, ISP etc.. Refer to the documentation supplied with these packages for further information.

4.3. Parallel I/O

When OS9 is running the parallel port may be used as a printer using the "pp3XX" device driver and "pp" device descriptor under interrupt control connected through JP3. This driver operates only as a printer output driver.

The "pp3XX" device driver uses the on-board parallel port and MC68681 resources, which should not generally be used for other functions.

4.4. EEPROM

When OS9 is running the EEPROM may be used for parameter storage using the "nvr3XX" device driver and "nvr" device descriptor as an OS9 SCF driver.

The device is sequentially read, and "re-wound" to the beginning by a "SS_Reset" call or being re-attached. Out of the 128 bytes available, the last 8 are reserved for the bootstrap ROM (see "Appendix A - Configuration EEPROM"), leaving 120 bytes available for user requirements.

4.5. Real Time Clock/Timer

The DS1215 contains the calendar information which is read by the OS9 kernel using the "ck3XX" clock module. This clock module also uses the MC68681 timer to provide a 10mS system heartbeat when OS9 is running.

During the bootstrap phase, the MC68681 timer is also used for BOOTP based booting.

4.6. Disk I/O

The addition of a BVME130 SCSI/Floppy Controller Board, gives the BVME3XX OS9 disk bootstrap and driver support. During the bootstrap phase, the floppy disk interface and SCSIbus can be used to interface to disk or tape storage devices for loading bootfiles, and a subset of the OS9 drivers is provided in the bootstrap ROMs.

When OS9 is running, all SCSI I/O uses the "scsi130" low-level device driver in conjunction with the relevant high-level device driver (i.e. "rbSCSI" for CCS hard disk drives, "sbMT2ST" for TEAC tape etc.). All floppy disk I/O is through the "rb130" driver. Device descriptors are normally "h0" for the hard disk drive, "d0" for the floppy disk drive and "mt0" for the tape drive, and the default device "dd" (normally whichever device is being used for bootstrap).

The "scsi130" and "rb130" device drivers use all of the facilities of the BVME130 controller under interrupt control. Three VMEbus interface modes are supported: DMA direct to main memory (mode 1), DMA to local memory followed by DMA to main memory (mode 2), or DMA to local memory followed by CPU access to local memory (mode 0). Normally for best performance SCSI devices use mode 1, and floppy disk drives use mode 2.

The SCSI devices are connected through the BVME130 J1 or J2 connector. If the BVME130 is at the end of the SCSIbus, then it's terminators should be fitted. The floppy disk drives are connected through the BVME130 J3 connector.

BVM SCSI drivers support most popular SCSI CCS hard disk drives, the TEAC MT2ST tape drive range, the ARCHIVE VIPER tape drive range and the HP DAT tape drive range.

The "rb130" driver supports most 1Mb 3½" and 5¼" floppy disk drives, but BVM recommends the 3½" TEAC FD235H (2Mb) or FD235J (4Mb) floppy disk drives for higher capacity use. The different floppy disk formats can be changed using the BVM "rbmode" utility.

4.7. Ethernet I/O

The addition of a BVME410 Ethernet Controller Board gives the BVME3XX OS9 network bootstrap and driver support. During the bootstrap phase, the Ethernet can be used for loading bootfiles using the INTERNET BOOTP protocols supported by OS9. The BVME410 Ethernet Address (allocated by the IEEE) is stored in it's on-board EEPROM, and is broadcast by BOOTP. Any BOOTP server recognising this address will respond, and a bootfile will be loaded.

When OS9 is running, all Ethernet I/O uses the "net410" low-level device driver in conjunction with the relevant high-level device driver (i.e. pcINET for PCLINK, nfmNET for NFM, ispNET for ISP etc.). Device drivers are normally "pc0" for PCLINK, "n0" for NFM, and "le0" for ISP. The "net410" device driver uses all of the facilities of the BVME410 controller under local DMA and interrupt control.

Connection to Ethernet is through the BVME410 P2 (Cheapernet/Thinnet) or P3 connector (AUI/MAU) to a transceiver. Ensure that the P2 connection uses the correct 50Ω 'T'-connector and the line is terminated at the

ends with 50Ω terminators. Selection between Cheapernet/Thinnnet and the AUI/MAU networks is made using LK1, 2 and 3 (see BVME410 User's Manual).

4.8. MC68040 MMU/CACHE/FPU

The BVME390/395 can be supplied with the MC68EC040, MC68LC040 or MC68040 processors, known in OS9 software terms as the BVME390, BVME393 and BVME395 respectively.

The MC68040 is the full MPU, including MMU and FPU support. BVM uses Microware's "ssm" (System Security Module) for memory management, and for setting up the memory and I/O caching regions. Caching regions for user processes are defined in the "init" module, created from the entries in the "systype.d" file.

The MC68LC040 excludes the FPU, but still includes the MMU. Thus support for this is similar to the MC68040, and based on the "ssm" module.

The MC68EC040 excludes both the MMU and the FPU. BVM uses it's own "acm" module to set up the memory and I/O caching regions for both user and system processes.

The BVME390 address map is split into three regions for caching purposes: the first 256Mbytes (\$00000000 to \$0FFFFFFF) for main memory; the top 256Mbytes (\$F0000000 to \$FFFFFFF) for I/O; and the remainder (\$10000000 to \$EFFFFFFF) for alternate caching where possible.

The bootstrap software sets up the top 256Mbytes of the address map for the I/O region to non-cached bus-serialised supervisor-state data accesses. This includes accesses to VMEbus A24:D16 and A16:D16 for I/O boards, as well as on-board I/O.

The "ssm" module sets up the rest of the address map (other than the I/O region) to full copyback caching for supervisor-state data accesses (instruction caching is writethrough). User-state data accesses are set up by "ssm" using the cachelist supplied in the "init" module. This sets up user-state accesses to the first 256Mbytes of the address map for copyback caching, the top 256Mbytes as non-cached bus-serialised, and the remainder for writethrough caching.

The "acm" module sets up user and supervisor-state accesses to the first 256Mbytes of the address map for copyback caching, the top 256Mbytes as non-cached bus-serialised, and the remainder for writethrough caching.

The Microware "syscache" module is used to control the caching, allowing user and supervisor software to control the caching (enable/disable, flushing for data/instruction caches).

The MC68040 has an on-chip floating point co-processor, and the "fbsp040" floating point support module is used to emulate the MC68882/1 un-implemented instructions. The MC68LC040 and MC68EC040 use the "fpu" floating point emulation module to emulate floating point instructions.

4.9. MC68030 MMU/CACHE/FPU

The BVME370/380 can be supplied with the MC68EC030 or MC68030 processors, known in OS9 software terms as the BVME370 and BVME380 respectively (BVME372 and BVME382 for 4Mb versions).

The MC68030 is the full MPU, including MMU support. BVM uses Microware's "ssm" (System Security Module) for memory management.

The MC68EC030 excludes the MMU. No "ssm" module is required with this processor, but no harm is caused by using it, as accesses to the MMU are ignored.

The Microware "syscache" module is used to control the caching, allowing user and supervisor software to control the caching (enable/disable, flushing for data/instruction caches). With either processor type the board normally caches memory accesses below address \$900000.

The BVME370/380 can accept a MC68882 floating point co-processor which will handle all floating point support. If the MC68882 is not fitted the "fpu" floating point emulation module is used to emulate floating point instructions.

4.10. ABORT Handling

Both the ABORT switch and the VMEbus share the autovector level 7 interrupt vector (31). The ABORT switch is intended to call the ROM debugger if present in the ROMs, whereas a VMEbus autovector level 7 interrupt is usually intended to warn of impending system power loss.

The bootstrap software will normally call the bootstrap ROM debugger if it is enabled, but will call an OS9 autovector level 7 handler installed by user application software (if any) when any autovector level 7 interrupt occurs with the debugger disabled (see "3.2. Bootstrap Reconfiguration").

ABORT is enabled by fitting LK1 (370/380) or LK4 (390/395).

The CPU board requires special configuration to support VMEbus autovector level 7 interrupts.

5. Other Software Support

5.1. Networking

BVM supplies separate network software for Ethernet support on the BVME410 and serial I/O facilities on the CPU board. The disks include load files ready for use with the BVME410, and full documentation in the ".doc" and/or ".inf" text files.

Microware's OS9/ISP Internet support package is available. The Ethernet version also supports BOOTP booting from Internet BOOTP servers, and the serial version also includes support for SLIP protocols.

Microware's OS9/NFS Network File System package is available for Unix system compatibility, and runs on top of OS9/ISP.

Microware's OS9/NFM Network File Manager is available giving full file system connection between OS9 nodes.

BVM's PCLINK support package is also available, giving file system access between OS9 and PC nodes.

BVM's IOF File Manager support is available for "raw packet" access to the network media.

All of the above protocols may run concurrently through the BVME410 Ethernet or CPU board serial connections, and are also compatible with the BVME4000 SBC running the same protocols.

5.2. TargetLink

The BVM TargetLink package is available for all BVM CPU's, including the BVME370/380/390/395. This package provides the documentation, command files and examples required to build ROM based embedded versions of OS9.

TargetLink also includes facilities for downloading OS9 modules via a serial port and debugging applications in a target environment.

5.3. MEM400 Flash Memory

BVM supplies separate software for flash memory support on the BVME390/395 using the MEM400 memory module. This operates with the BVM NVF file manager. It can be used for data logging purposes (similar to tape) and can also be used to store OS9 modules for ROM based systems. The disks include load files ready for use with the MEM400, and full documentation in the ".doc" and/or ".inf" text files.

Appendix A - Configuration EEPROM

Address	EEPROM Contents
0 to \$77	Available for user requirements via "/nvr" and "nvr3XX".
\$78	Boot Driver [0] 7-5 = ID, 4-0 = driver no.
\$79	Boot Driver [1] 7-5 = ID, 4-0 = driver no.
\$7A	Boot Driver [2] 7-5 = ID, 4-0 = driver no.
\$7B	Boot Driver [3] 7-5 = ID, 4-0 = driver no.
\$7C	7-5 = Own SCSI ID, 4-0 = Switches.
\$7D	7-6 = Baud Rate, 5-4 = Boot RAM, 3-0 = reserved.
\$7E	reserved
\$7F	CHECKSUM of \$78 to \$7E

Bit	Switch Meaning (1 = enable)
0	Debugger boot staging.
1	Debugger calling.
2	Auto menu selection.

Code	Baud Rate Value
0	9600 (default)
1	19200
2	38400
3	38400

Code	Add. Boot RAM Code
0	None (default)
1	512Kbytes
2	1.5Mbytes
3	3.5Mbytes