Setup Utility

Setup Utility

Important Notice

Microid Research, Inc. reserves the right to revise this document and make changes in the specifications of the product described herein at any time without notice and without obligation to notify any person of any such revision or change. Microid Research, Inc. makes no warranty, express, implied or statutory concerning this document or its use and assumes no responsibility for any omissions or errors which may appear in this document nor does it make a commitment to update the information contained in this document. **MICROID RESEARCH, INC. SPECIFICALLY DISCLAIMS LIABILITY FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES RESULTING FROM THE ACQUISITION, USE OR SALE OR OTHER TRANSFER, PERMITTED OR OTHERWISE, OF THIS DOCUMENT OR THE USE OF THE PRODUCTS DESCRIBED IN THIS DOCUMENT. MICROID RESEARCH, INC. SPECIFICALLY DISCLAIMS ALL WARRANTIES, EXPRESS, IMPLIED OR STATUTORY REGARDING THE FREEDOM OF THIS DOCUMENT OR THE PRODUCT DESCRIBED HEREIN FROM INFRINGEMENT OF ANY INTELLECTUAL PROPERTY RIGHT OF ANY THIRD PARTY. MICROID RESEARCH, INC. SPECIFICALLY DISCLAIMS ALL WARRANTIES AND REPRESENTATIONS, EXPRESSED, IMPLIED OR STATUTORY, REGARDING THIS DOCUMENT OR THE SPECIFICATIONS, QUALITY OR PERFORMANCE OF THE PRODUCT DESCRIBED IN THIS DOCUMENT, ITS MERCHANTABILITY OR ITS FITNESS FOR ANY PARTICULAR PURPOSE.**

MR BIOS is a registered trademark of Microid Research, Inc. Other trademarks are property of their respective owners.

Copyright © 1990,...,94 Microid Research, Inc.

No part of this document may be photocopied, reproduced, stored in a retrieval system, transcribed or translated to another language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, or otherwise, without the prior written consent of Microid Research, Inc.

ALL RIGHTS RESERVED

Table of Contents

Introduction 1

Summary Screen 4

Energy Management Setup Utility 10

Clock Setup Utility 14

Video Setup Utility 16

Keyboard Setup Utility 18

Floppy Drive Setup Utility 20

Fixed Disk Setup Utility 24

Low Level Format Utility 30

Boot Sequence Setup Utility 32

Ports Setup Utility 36

Security Setup Utility 38

Speed Setup Utility 42

Cache Setup Utility 44

Shadow RAM Setup Utility 48

DMA Setup Utility 52

Chipset Setup Utility 54

Appendix

Diagnostic Port 80H POST-CodesA-1

POST Beep Codes & Error MessagesA-2

Fixed Disk Parameter TablesA-5

Setup

Introduction

The custom features and hardware options in your PC/AT compatible computer are on-site selectable for maximum flexibility. You will need to *configure* these options through the built-in **Setup Utility** prior to using your computer for the first time.

The Setup Utility is a multi-screen, menu driven program. It has been customized for, and is contained within the BIOS EPROM(s) that reside on your motherboard. The information generated during a configuration session is recorded in special, low-power CMOS memory which is battery-maintained when the system is shut off. Since the battery may become discharged, it is recommended that you use **<PrtSc>** to make a hard-copy printout of each Setup Utility screen.

Invoking Setup Utility

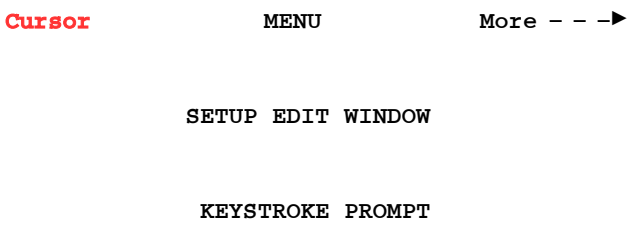
A procedure called Power-On-Self-Test (POST) is conducted each time the computer is booted. If the system status noted during POST cannot be reconciled with the Setup configuration stored in CMOS, then Setup Utility will be invoked **automatically**. Under more ordinary conditions, Setup can be manually invoked through the keyboard:

- Press **<Esc>** during poweron Memory Test
- Press **<Ctrl+Alt+Esc>** during runtime

While the memory size is scrolling on the CRT during *cold-boot*, you can press **<Esc>** to enter Setup. Analagous to the three key sequence **<Ctrl+Alt+Del>** that causes a system *warm-boot*, you can abort a current program and enter Setup by pressing **<Ctrl+Alt+Esc>**.

Exiting Setup Utility

To exit Setup (and boot the computer), press **<F10>**. All configuration changes edited in the various Setup screens are recorded into CMOS memory at this time. Be aware that *nothing is recorded until then*. Therefore, if you re-boot the computer or turn off the power (instead of pressing **<F10>**), these changes will be lost and the original configuration will remain unaltered. If you want to ***bail-out*** without storing any changes, then either (1) **press <Ctrl+Alt+Del>**, or (2) **turn off the power**.



Within Setup Utility

The Setup Utility screen is comprised of three sections, shown in the figure above. The top line contains a ***menu*** which lists individual configuration utilities. The central part of the screen contains the ***edit window*** where an individual utility is viewed and edited. The bottom line is a dynamic ***keystroke prompt*** which indicates currently valid keystrokes.

A reverse-video ***cursor*** is always present, either on the menu line or in the edit window. It directs your attention to the currently active field. Although several keys can be used to maneuver the cursor, the keyboard **<Arrow>** keys are generally used. As the cursor is moved across the menu from entry to entry, the edit window is updated to reveal each corresponding utility. When the cursor rests on an entry that you want to configure, move it downward into the edit window and begin editing the fields. The choices in the fields can generally be *scrolled* with the **<Space>** key, while a few fields require AlphaNumeric entry. Press **<PgUp>** when you are done, and the cursor will return to the menu.

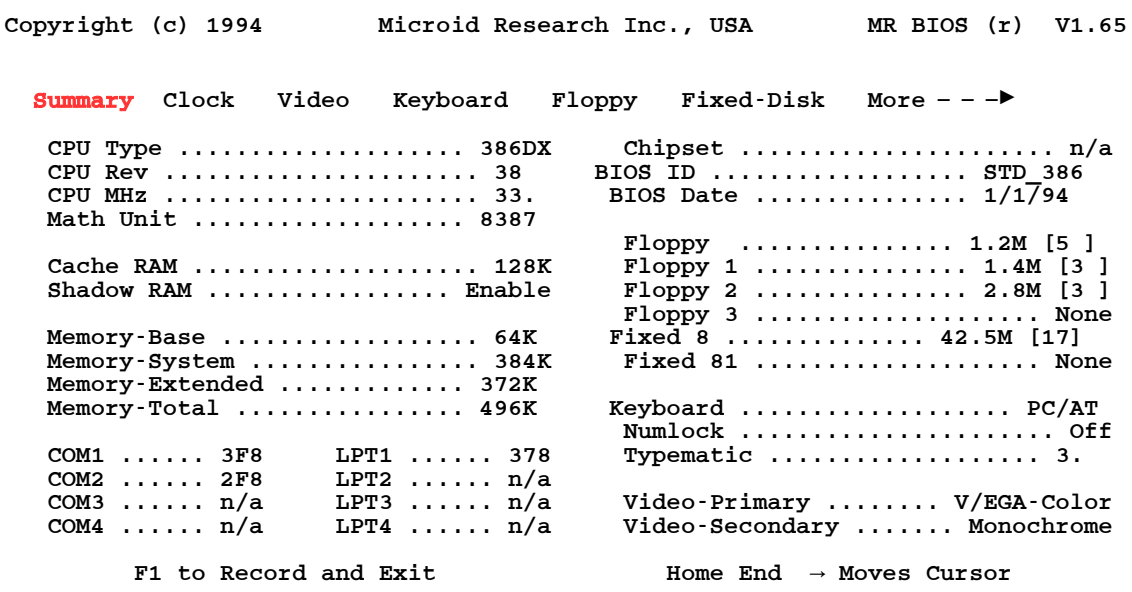
The following keys are available while the cursor is on the menu:

Right, Left Arrowright and left movement
Space, BackSpaceright and left movement
Tab, Shift-Tabright and left movement
Home, Endleftmost and rightmost entry
Down-Arrow, Enter, PgDnmove down into edit window
F10 record and exit Setup

These keys are generally available within the edit window:

Arrowsup, down, left, right movement
Space, BackSpacescroll choices in field
Plus, Minusscroll choices in field
AlphaNumericletters and numbers
Enter, (Esc)begin/end (abort) mode or A/N entry
Esc, PgUpexit edit window, back up to menu

(This page intentionally left blank)



Setup Summary Screen

SUMMARY SCREEN

This is the screen you are greeted with upon first entering the Setup Utility. It is a ***view-only*** report of the hardware components in your computer (although a few user selectable options can also be viewed here). This information is useful when you want to confirm that BIOS has properly recognized a newly installed component, or when you simply want to review your machine's content.

The following items comprise that part of the hardware environment which BIOS will autosense during POST. These resources *simply exist* in the system, and no Setup Configuration Utility is available or required to manage them.

- **CPU Type, Revision, MHz**
- **Math Unit (Coprocessor)**
- **Chipset (Core Logic)**
- **Memory-Base and Total**
- **Keyboard and PS2-Mouse**

The remainder of the items listed on the Summary Screen are affected by selections made elsewhere in the Setup Utility. There is some property or operation state of the resource that BIOS cannot determine or is optional. For example, the Floppy drive type (1.2M, 1.4M, etc) cannot be autodetected by BIOS, and requires your explicit selection. (See Floppy Setup Utility). As another example, the Shadow-RAM operation (when this feature is present) is fully user-configurable. In some designs, the Extended Memory is affected (decreased) by the amount allocated to Shadow-RAM. In general, it is a good practice to examine this Summary Screen after making each configuration change and prior to exiting the Setup Utility.

Again, the Summary Screen is ***view-only***; nothing can be changed here. If you want to make configuration changes or explore other Setup Utility screens, press **<Right Arrow>** to move the cursor rightward on the menu-line. To exit the Setup Utility (and boot the computer), press **<F10>**.

The following is a description of each field found on the Summary Screen.

CPU Type - This field shows the microprocessor (Central Processing Unit) in the system.

CPU Type ... 386DX	386DX CPU
CPU Type ... 486DX	486DX CPU
CPU Type ... 486BL	IBM Blue Lightning

CPU Rev - This field shows the model and revision code that is reported by the CPU. In general, the left two digits represent the CPU type, and the right two digits are the revision number. (286 CPUs do not identify themselves; this field does not appear).

CPU Rev 44	486DX, Rev 4
CPU Rev 35	386DX, Rev 5
CPU Rev 238	386SX, Rev 8

CPU MHz - This field shows the fastest operating frequency (clock) of the computer. Some 486 versions also indicate (1X) or (2X) oscillator usage.

CPU MHz ..... 25.	25 MegaHertz CPU speed
CPU MHz ....(2x) 33.3	33.3 MHz 486, 66.6 MHz Osc
CPU MHz ... (1x) 5.	50.0 MHz 486, 50.0 MHz Osc

Math Unit - The math unit is the type of numeric coprocessor(s) detected in the system.

Math Unit ... n/a	coprocessor not found
Math Unit ... 387SX	generic 387SX detected
Math Unit ... Cyrix EMC87	Cyrix high performance math
Math Unit ... Weitek 4167	Weitek unit detected
Math Unit ... Built-in	486DX math unit

Cache RAM - This field shows the configured state of the cache at boot-time.

Cache RAM ... n/a	no cache
Cache RAM ... Disable	cache off

Cache RAM ... Enable	on (CPU only)
Cache RAM ... 256K	cache on, 256K size

Shadow RAM - This indicates if *any* Shadow RAM (ROMs copied to, and executed from memory for enhanced performance) is enabled. When marked **Enable**, consult the Shadow Setup Utility to identify *which* regions are shadowed.

Shadow RAM ... n/a	shadow RAM not available
Shadow RAM ... Enable	shadowing in effect
Shadow RAM ... Disable	shadowing turned off

Memory-Base - This is the amount of Base Memory (below the 640K boundary) found to be in working order.

Memory-Base ... 64K	640 Kilobytes Base Memory

Memory-System - Many designs reserve a portion of memory, typically 384K, for special uses. Some may be allocated to Shadow RAM, and the remainder might automatically be *remapped* to the Extended Memory pool. This field shows the amount of memory retained for system use.

Memory-System ... K	none, or fully reallocated
Memory-System ... 384K	384 Kilobytes Special Memory

Memory-Extended - This is the amount of Extended Memory (above the 1 Megabyte boundary) found to be in working order.

Memory-Extended ... 372K	3 Megabytes Extended Memory

Memory-Total - This is the total amount of memory installed in the system. It is the sum of the three preceding quantities: Base + System + Extended = Total.

Memory-Total ... 496K	4 Megabytes Total Memory

COM1 (2,3,4) - These are the I/O addresses of the serial ports configured in the system. (More may actually be *available*. See **PORTS** Setup Utility).

COM1 ... n/a	serial port not present
COM4 ... 3F8	serial port at I/O 3F8

LPT1 (2,3,4) - These are the I/O addresses of the parallel (printer) ports configured in the system. (More may actually be *available*. See **PORTS** Setup Utility).

LPT1 ... n/a	parallel port not present
LPT3 ... 378	parallel port at I/O 378

Chipset - Most modern system boards contain a few, relatively large, surface-mounted ASIC components known as the *core logic chipset*. The chipset characterizes the properties of the system board in general.

Chipset ... n/a	none (discrete logic)
Chipset ... UMC 82C491	UMC 386/486 chipset
Chipset ... OPTi 596VHP	OPTi Pentium chipset

BIOS ID - This field identifies the BIOS firmware that is present on the system board. It should be referenced when reporting a problem or ordering an upgrade BIOS.

BIOS ID ... STD_286	original PC/AT
BIOS ID ... MICR3	Micronics discrete

BIOS ID ... OPTI54	for OPTi 596VHP

BIOS Date - This field reports the date the BIOS firmware was compiled. Note its format (dd/mm/yy or mm/dd/yy) is selected in the Clock Setup Utility.

BIOS Date ... 2/5/94	Feb 5, 1994 (USA)
BIOS Date ... 5/2/94	(International)

Floppy (1,2,3) - The Floppy Drives configured in the system are shown here. While Floppies 0 and 1 correspond to drives A: and B: respectively, the naming convention for Drives 2 and 3 varies with DOS versions.

Floppy ..... n/a	no floppy controller
Floppy ..... None	no drive selected
Floppy 1 ... 1.2M [5]	1.2Mb 5¼" drive
Floppy 2 ... 1.4M [3]	1.4Mb 3½" drive
Floppy 3 ... 2.8M [3]	2.8Mb 3½" drive

Fixed 8 and 81 - These fields show the configured Size and Type for Fixed Disk Drives 80 and 81 (usually C: and D: respectively).

Fixed 8 ..... n/a	no controller card
Fixed 8 ..... None	no drive selected
Fixed 8 .... 42.5M [17]	42.5Mb drive type 17
Fixed 81 ... 13.7M [47]	user programmable type 47

Keyboard - This is the type of keyboard identified during powerup.

Keyboard ... PC/XT	PC/XT type keyboard
Keyboard ... PC/AT	PC/AT type keyboard
Keyboard ... AT/PS2	AT/PS2 type keyboard

NumLock - This is the selected state of the NumLock key at bootup.

NumLock ... Off	cursor movement mode
NumLock ... On	numeric entry mode

Typematic - This is the Keyboard Typematic Repeat Rate (in characters per second). Possible values range from 2 to 30.0 cps.

Typematic ... Default	keyboard default unchanged
Typematic ... 3.	30 cps rate is selected

PS2-Mouse - In systems with dual PS2 style connectors (one for Keyboard, one for Mouse), this field indicates if a PS2-Mouse is present.

PS2-Mouse ... n/a	no mouse found
PS2-Mouse ... Yes	mouse is plugged-in

***Note** - This field replaces the Typematic field (above) when a PS2 keyboard subsystem is detected.*

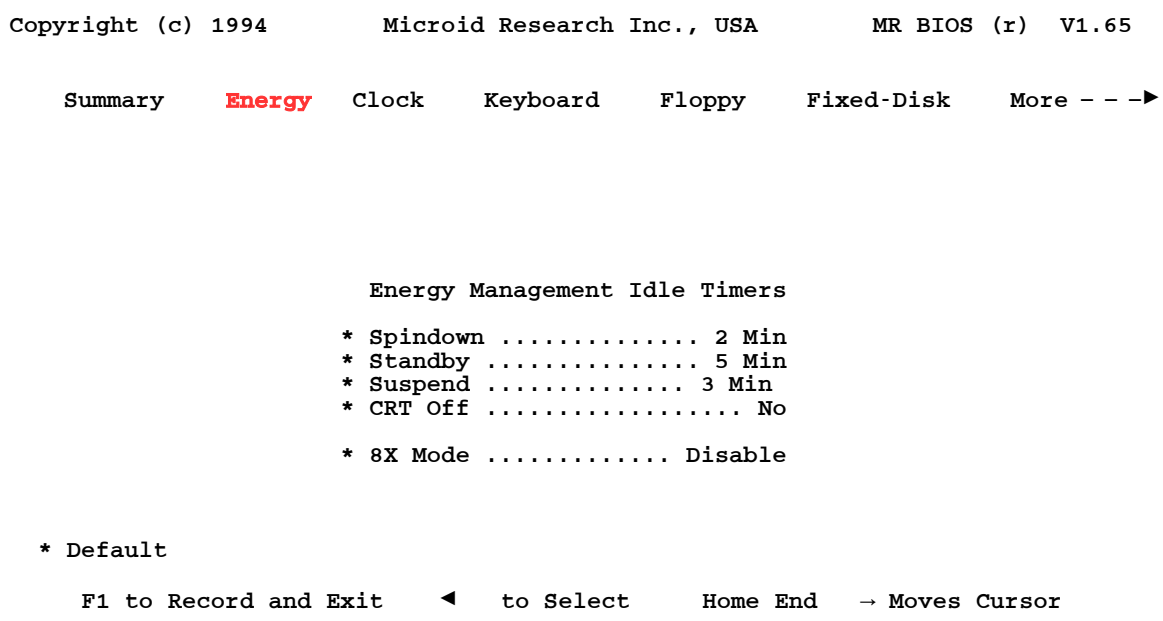
Video-Primary - This is the main Video Display Adapter which will be in use when control is passed to an Operating System at boot-time.

Video-Primary ... None	see Video Setup
Video-Primary ... Monochrome	black and white
Video-Primary ... V/EGA-Mono	enhanced graphics, b/w monitor
Video-Primary ... V/EGA-Color	enhanced graphics, color monitor

Video-Secondary - This field indicates the presence of a second Video Adapter. It will remain idle until activated by specialized software.

`Video-Secondary ... n/a` second video not found

(This page intentionally left blank)



Energy Management Setup Utility Screen

ENERGY MANAGEMENT

For compliance with the **EPA Energy Star** program, various energy conservation methods can be integrated into your computer. The objective is to automatically reduce power to devices like the fixed disk and CRT when they are idle, and to restore their full operation (with minimum delay and inconvenience) upon detection of activity. This is accomplished through use of *idle timers* and event monitoring techniques. Within this utility, you can select how rapidly the timers will expire, or, you can disable them altogether if you find they are interfering with your computing.

This implementation conforms to, and makes use of several industry standards:

- **APM** - Intel/Microsoft Advanced Power Management
- **ATA** - AT Attachments Specification (IDE Drive)
- **DPMS** - Display Power Management Signaling (VESA/Video)

APM coordinates the BIOS, Operating System, and Application Programs to act together as *participants* of Power Management. You will generally realize best power savings under DOS by using its APM driver, POWER.EXE, set to "ADV:". Under WINDOWS, use its Setup facility to enable POWER.DRV, then find the Power icon in the Control Panel, and select "ADVANCED" power management there.

Modern ATA compliant IDE Drives that provide **Spindown** services can be programmed to automatically turn off the spindle motor when the drive has been idle awhile. It will then wait *silently* for a new access to cause it to turn the motor back on.

VESA/DPMS compliant video adapters and monitors are designed to be used together as a matched set. Special signals exist between the pair that allow the CRT to be put into various low power states. These functions are generally not automatic, though, requiring manual instruction from the System BIOS.

Four power management *states* are defined by APM and DPMS:

- **Run** - Fully powered-up
- **Standby** - Reduced power, can instantly be returned to Run State
- **Suspend** - Minimum power, may be slow to return to Run State
- **Off** - Fully powered-down

The **Run** state is the natural, full-power state that you would expect when there is no power management at all. If the computer is allowed to sit idle for awhile, the Standby or Suspend timer will expire and that respective state will be engaged. Any activity (eg, keystrokes, mouse, modem, etc.) will reset those timers and restore the system to the Run state. If you want to disable power management altogether, disable *both* the Standby and Suspend timers.

The **Standby** state is an intermediate state, between Run and Suspend. It uses a short timer (a few minutes) to determine when the system should be considered *temporarily* idle. When no activity is occuring and the **Standby Timer** thus expires, measures taken to reduce power consumption may include slowing the CPU, and more noticably, blanking the CRT. The DPMS Standby (screen blanking) mode only *partially* powers-down the CRT, allowing instant recovery when computing activities are resumed.

The **Suspend** state is the final destination of power management, occurring after the system has sat idle for a significant period of time. It uses a long timer (up to one hour) to determine when the computer should be considered *unattended*. When the **Suspend Timer** expires, measures to severely reduce power may include slowing or halting the CPU, and of particular interest, the video will be disabled to the fullest extent possible. The DPMS Suspend mode induces a *cold* power-down of the CRT electronics, and it may require a lengthy warmup period (not unlike turning on your monitor in the morning) when computing is resumed.

VESA/DPMS defines a **CRT Off** mode that, taken in its most literal sense, can turn off the CRT power completely. When a monitor is turned off by this method, it may (or may not) require manual actuation of the power switch (pushbutton) on the CRT to later turn it back on. If you want the CRT to be fully turned off upon entering the Suspend state, select **Yes** in this field. Otherwise, select **No** to use the more ordinary DPMS Suspend mode.

An optional **8X Mode** field may appear in this utility that is used primarily during factory testing and demonstrations. It divides the timers by a factor of 8, accellerating the power management state transitions for quick observation. You will generally have no reason to enable this mode, and it should be left disabled. The 8X Mode is cancelled whenever the system is (re)booted; you will always find it disabled upon entering Setup Utility.

Spindown - Turn off the IDE Drive motor after **1**, **2**, or **5** minutes of inactivity. Or, disable the timer by selecting **Off** to leave the motor running indefinitely.

Spindown ... 2 Min	motor off after 2 min
Spindown ..... Off	motor always runs

Standby - Select a period of **2**, **5**, or **10** minutes inactivity after which the computer is considered to be *temporarily* idle and a mildly low power state will be put into effect. Or, disable this timer by selecting **Off** to prevent Standby mode altogether.

Standby ... 5 Min	Standby if idle for 5 min
Standby ..... Off	never go to Standby mode

Suspend - Select a longer period of **5 min** to **1 hr** inactivity after which the computer is considered to be *unattended* and severe power reduction steps occur. Or, disable this timer by selecting **Off** to prevent Suspend mode altogether.

Suspend ... 15 Min	Suspend if idle for 15 min
Suspend ..... Off	never go to Suspend mode

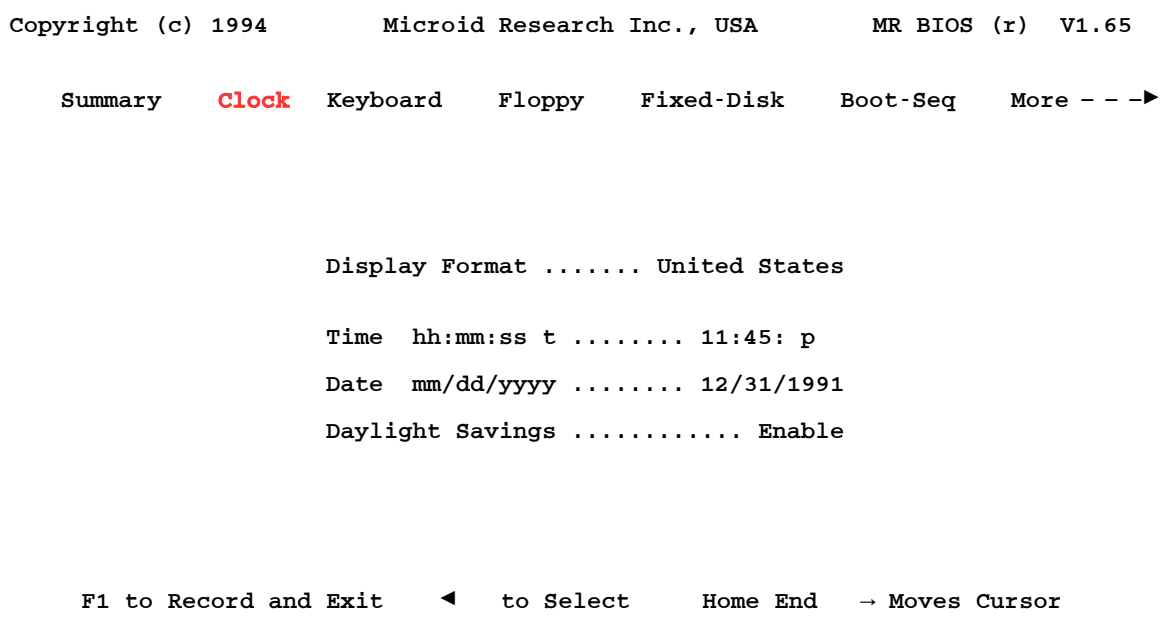
CRT Off - The CRT can be turned off altogether upon entering the Suspend state by selecting **Yes** here. Select **No** to use the standard, low-power Suspend mode.

CRT Off ... Yes	turn off idle monitor
CRT Off .... No	low-power mode only

8X Mode - For factory test and demonstration purposes, the power management state transitions can be accellerated by a factor of 8 by selecting **Yes** here. Select **No** for normal operation.

8X Mode ... Yes	fast test/demo mode
8X Mode .... No	normal operation

Note - *8X Mode is automatically cancelled and reverts back to normal* **(No)** *whenever the computer is (re)booted.*



Clock Setup Utility Screen

CLOCK SETUP

Your system board contains a Real Time Clock (RTC) in which the time and date are maintained. It is battery powered when the computer is shut off.

The RTC needs to be set with the initial time/date, and adjustments are required periodically for continued accuracy. Do not be alarmed if your wristwatch keeps better time than your computer. Variations in voltage (power-supply or battery) and other technical issues make it impractical to tune the RTC with the same degree of precision as a dedicated timepiece.

Display Format - The time and date can be selected to appear either in United States or International format, according to your preference.

United States	12 hour, mm/dd/yyyy
International	24 hour, dd/mm/yyyy

Time - The time is shown in the selected Display Format (above). If USA 12-hour format is selected, the time of day indicator **a** or **p** (am/pm) appears. Otherwise, it is a 24-hour clock. To change the time, move the cursor onto this field, then press **<Enter>** and edit. Upon completion, press **<Enter>** to record the new time.

Time hh:mm:ss t ... 2:27:35 p	12-hr am/pm (USA format)
Time hh:mm:ss 14:27:35	24-hr (International)

Date - The date is shown in the selected Display Format (above). USA format is mm/dd/yyyy, and International format is dd/mm/yyyy. To change the date, move the cursor onto this field, then press **<Enter>** and edit. Press **<Enter>** when done.

Date mm/dd/yyyy ... 5/14/1991	May 14, 1991 (USA format)
Date dd/mm/yyyy ... 14/5/1991	14 May, 1991 (International)

Daylight Savings - The RTC can be instructed to automatically correct the time on the two daylight savings days of the year. Altering this field will not cause an immediate change - the RTC adjusts the time only when a daylight savings transition occurs.

Daylight Savings ... Enable	RTC auto-adjusts time
Daylight Savings ... Disable	time not adjusted



Video Setup Utility Screen

VIDEO SETUP

Most systems possess a single Mono or VGA monitor, and BIOS configures the video automatically. If this describes your system, skip over this section. If your system contains two video adapters, a CGA adapter, or you want to run without video, this section will be of interest to you.

> **Note -** *BIOS autodetects all video cards in your system without the aid of the Mono/Color jumper (dipswitch) on the motherboard. Instead, that jumper is used as a master Security Override. (See Security Setup Utility).*

DUAL-MONITOR SYSTEMS

Your computer permits installation of two video adapters, provided one is monochrome and the other is color. Dual-video systems are useful in Computer Aided Drawing (CAD) and similar applications, allowing separation of text and graphics onto dedicated monitors.

When two monitors are present, one must be designated as the ***primary*** device which the Operating System (eg, DOS) and applications use for text display. The ***secondary*** adapter is placed into a standby state and remains black until activated by specialized software such as CAD. This Utility provides a convenient way to (re)assign the primary adapter, whereas this was historically accomplished with a jumper on the motherboard.

CGA ADAPTER CARDS

Early technology Color Graphic Adapter (CGA) cards produce severe screen ***snow*** (also called ***hash***) when accessed at full speed in 80-column AlphaNumeric mode (the standard text mode). Newer CGA cards do not have this speed limitation. When BIOS detects CGA, an additional field will appear on this screen to address this issue.

VIDEO-LESS OPERATION

To configure a system to operate without video, simply turn the power off and remove the video card(s). Upon turning the power back on, BIOS will detect the video change, beep once, and wait for you to press **<Enter>** to enter Setup. Press **<Enter> <F10>** (enter and leave Setup), or simply reset the computer. The system will boot thereafter without video.

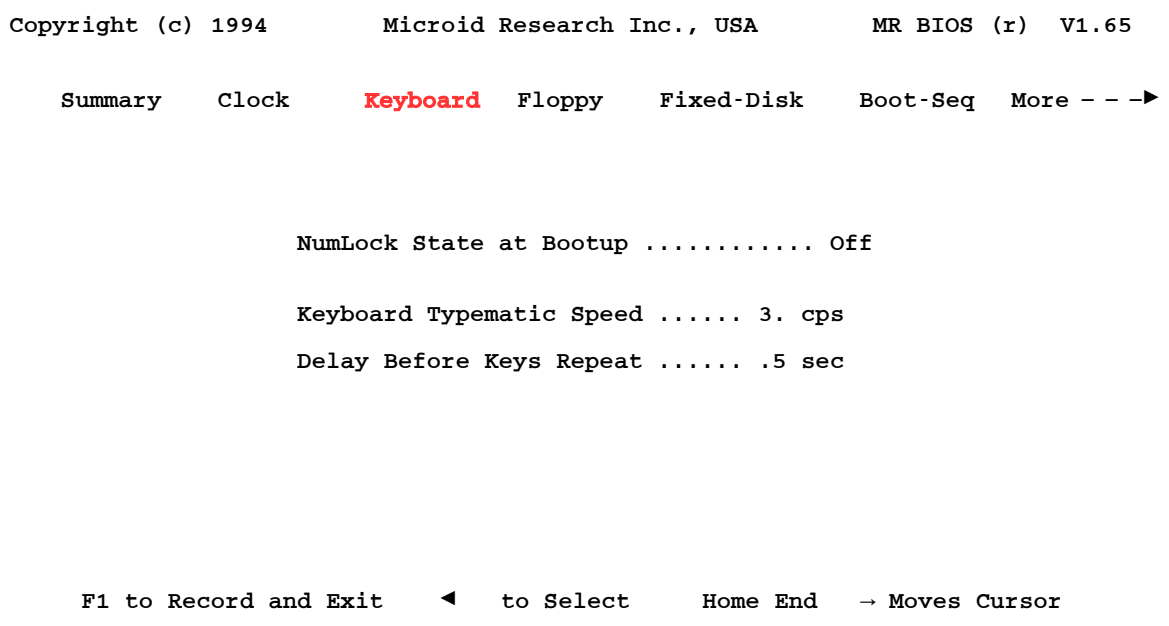
You can also *explicitly* set the Primary Video field to **None**, then press **<F10>** (to record), power down and remove the video card(s). Don't forget to remove the video card(s) prior to turning the power back on, else BIOS will complain of a hardware change and force you back into the Setup Utility.

Primary Video - Select the Video Adapter that will be active when the system is booted. Choices are limited to the present hardware. If two adapters are installed, the secondary monitor will remain black until accessed by special software.

Primary Video ... None	video-less operation
Primary Video ... Monochrome	black and white
Primary Video ... CGA - Fast	color graphics, no snow
Primary Video ... CGA - Snow	color graphics, slow
Primary Video ... V/EGA-Mono	advanced graphics, b&w
Primary Video ... V/EGA-Color	advanced graphics, color

CGA Gets Snow - Appearing only when a CGA card is detected, this field controls the CGA video access speed. Select **Yes** if screen interference is observed.

CGA Gets Snow ... Yes	slow access to fix snow
CGA Gets Snow ... No	fast video access



Keyboard Setup Utility Screen

KEYBOARD SETUP

Powerup settings for the **NumLock** key state and keyboard **Typematic** functions can be selected here according to your preference.

The **NumLock** state controls the operation of the numeric keypad found on the rightmost section of your keyboard. When disabled, the NumLock LED indicator will be dark, and the keys will produce special control functions (PgUp, PgDn, Home, End, Ins, Del, and cursors). When NumLock is enabled, the LED will be illuminated and the keystrokes will produce numbers. This utility only establishes the *initial boot-time* state; you can toggle it whenever you like simply by pressing **<NumLock>**.

Your PC/AT style keyboard has a built-in **Typematic** feature which automatically repeats the currently pressed key until it is released. An initial **delay** allows sufficient time to remove your fingers during ordinary typing. If the key is held down long, it will begin to repeat at a constant **speed**. Both of these parameters can be selected according to your preference, or, the keyboard can be left unprogrammed (default) to produce approximately 10 character per second (cps) repeat rate after 0.5 second delay.

NumLock State at Bootup - The initial NumLock state is programmable for cursor or numeric operation. **Off** selects cursor control, and **On** selects numeric entry.

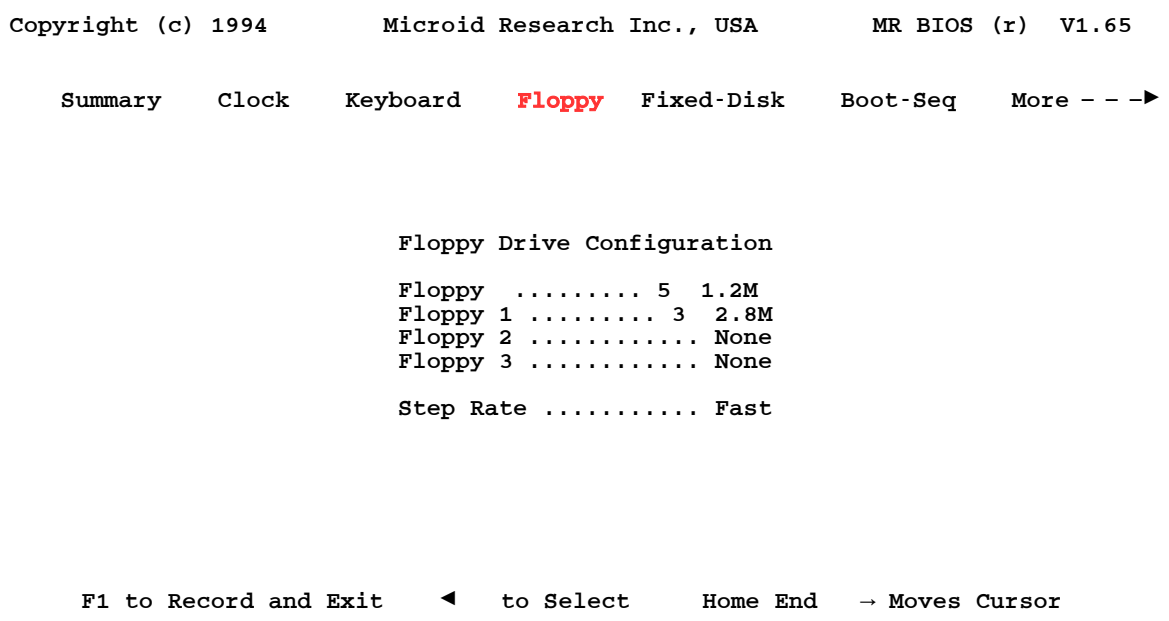
NumLock State at Bootup ... On	NumLock on (numeric)
NumLock State at Bootup ... Off	NumLock off (cursor)

Keyboard Typematic Speed - A key will eventually begin repeating after it is held down. You can select repeat rates from a very slow **2.0** cps up to a quick **30.0** cps, or you can leave it unprogrammed by selecting **Default** (approx 10 cps).

Keyboard Typematic Speed ... Default	native value
Keyboard Typematic Speed ... 3. cps	lively keyboard

Delay Before Keys Repeat - Choose a delay from **0.25** to **1.0** second that comfortably allows you to release the keys before they begin to repeat. (This field will display **Default** and cannot be changed if the Typematic field above is set to **Default**).

Delay Before Keys Repeat ... Default	native value
Delay Before Keys Repeat ... .5 sec	comfortable delay



Floppy Drive Setup Utility Screen (single controller card)

FLOPPY DRIVE SETUP

You may have as many as four diskette drives in your computer, referred to here as Floppy 0 through 3. The familiar names Drive A: and B: correspond to Floppy 0 and 1 respectively, while the drive letters for Floppy 2 and 3 depend on your version of DOS. Each drive in your system must be declared by specifying its ***type*** from this list:

- **360K 5¼" low density**
- **720K 3½" low density**
- **1.2M 5¼" high density**
- **1.4M 3½" high density**
- **2.8M 3½" extra density**

The ***step-rate*** (radial track-to-track speed of the recording heads) is also programmable. It should be set to **Fast** to exploit the improved performance of modern equipment. A **Slow** setting is provided for backward compatibility with original PC standards.

2.8M FLOPPY DRIVES

The BIOS in your computer fully supports 2.8M floppy drives. Many vendors are suppling retrofit software drivers to supplement systems that cannot support this latest technology. There is no need to use such a driver, and to do so will likely cause problems.

There is also built-in support for a second controller card. If your system does not have a 2.8M drive and you plan to upgrade, check first that your disk controller card has an

i82077 or **NSC8744** (or equivalent) Floppy Disk Controller chip (FDC). If not, you will need to add a secondary controller card. See Advanced Topics later in this section.

FOUR FLOPPY SUPPORT

The BIOS in your computer has built-in support for four floppy drives. Historically, BIOS support has been limited to two floppies and software drivers were used to extend it to four. There is no need to use such a driver here, and doing so will likely cause problems.

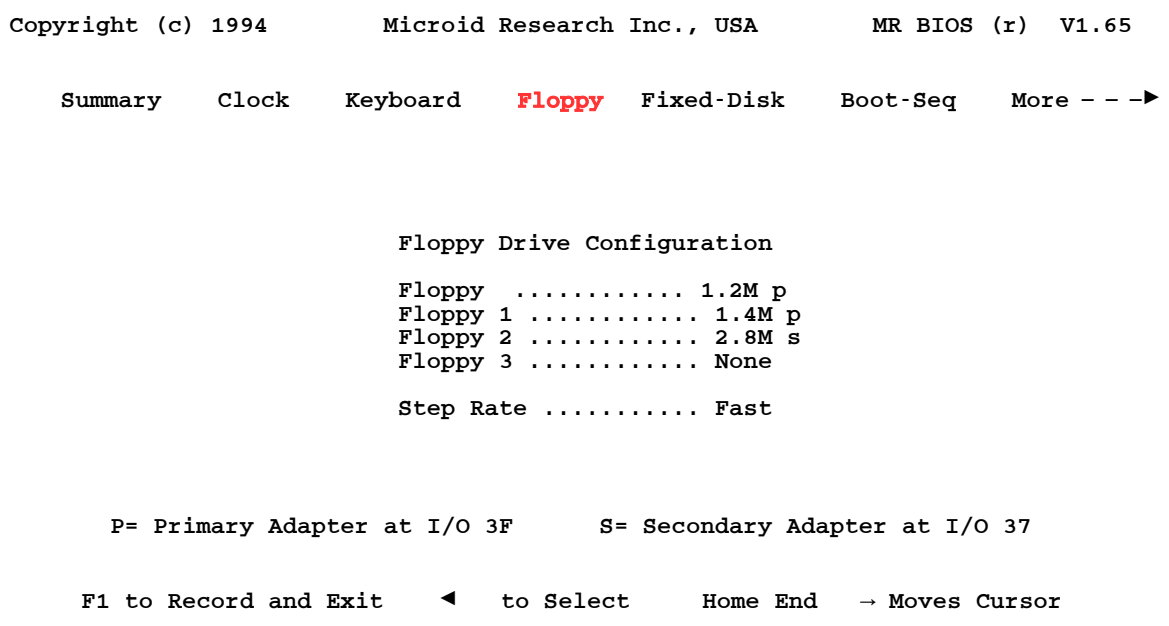
In addition to supporting four-floppy controller cards, this BIOS can also manage a pair of standard (dual-floppy) cards. If your system contains two cards (or you plan to add a second), refer to the Advanced Topics later in this section.

Floppy (1,2,3) - Indicate the drives present by specifying their types. Floppy 0 and 1 correspond to Drive A: and B: respectively. The drive letters for Floppy 2 and 3 depend on your Operating System.

Floppy ..... n/a	no adapter card
Floppy ..... None	marked absent
Floppy ... 5 36K	drive A: is 360K
Floppy 1 ... 3 2.8M	drive B: is 2.8M

Step-Rate - Track-to-track speed of the recording head should be set to **Fast** for best performance. **Slow** conforms to old standards.

Step-Rate ..... Fast	lively operation
Step-Rate ..... Slow	conservative



Floppy Drive Setup Utility Screen (two controller cards)

ADVANCED TOPICS

Most systems contain a single, ROM-less disk controller card. If this describes your system, skip over this section. (The primary/secondary message in the figure above only appears in systems with two controller cards. Floppy controllers with ROM programming will usually display a signon banner during POST indicating such).

FLOPPY CARDS WITH ROMS

ROM programming on floppy controller cards is intended to supplement a system whose BIOS cannot make use of the cards' special features. If the BIOS already has the target capabilities, the duplication in programming will usually result in malfunction. The BIOS in your computer already supports all *state of the art* floppy configurations. If you are upgrading your floppy subsystem for 2.8M or four-floppy operation, try to select a card without a ROM on it. Otherwise, you should disable or remove the ROM(s) according to the manufacturer's documentation.

SECONDARY CONTROLLER

Setup for two-controller systems proceeds almost exactly as described for single-controller systems. The only difference is that you also need to identify the controller card attached to each drive. The card attached to Floppy 0 (Drive A:) is called ***primary***, whereas the add-on card is called ***secondary***. They are indicated on this screen with abbreviations **p** and **s** respectively.

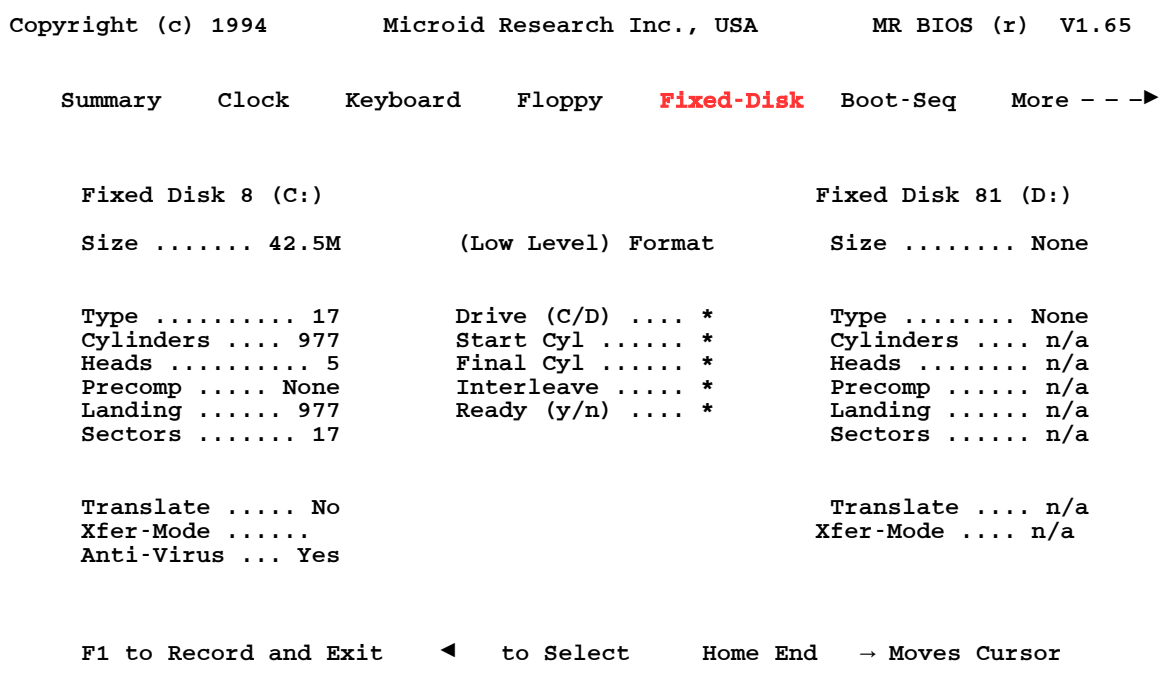
After the system is fully enclosed, it will not be apparent how to recreate the correct Setup settings if CMOS storage is somehow lost (eg, discharged battery). Be sure to use **<PrtSc>** to print a hard-copy of this screen.

The following technical information will be of interest when installing a second controller and cabling the drives:

- The primary floppy card resides at I/O address 3F0.
- The secondary floppy card resides at I/O address 370.
- Both cards share Interrupt Level 6, IRQ6.
- Both cards share DMA Request 2, DRQ2.
- Drives are assigned sequentially without gaps, Floppy 0,1,2,3.
- Drive-Select 0 (DS0) on the primary card is connected to Floppy 0.
- The numerically lowest drive on the secondary card responds to its DS0.
- Drive-Selects are assigned sequentially on each card, one per connected floppy.

Floppy (1,2,3) - Indicate each drive present by specifying its type and controller card. Primary and Secondary cards are designated by **p** and **s** respectively.

Floppy ..... n/a	no adapter card
Floppy ..... None	marked absent
Floppy ..... 1.2M p	floppy on primary card
Floppy 3 ..... 2.8M s	floppy on secondary card



Fixed Disk Setup Utility Screen

FIXED DISK SETUP

Two related utilities are provided on this screen. The **Drive Definition** utility is used to specify the type of disks present, and is discussed here. The **Low-Level Format** function is not ordinarily used; its instructions are presented in the following section.

DRIVE DEFINITION

The properties of each fixed disk in your computer must be described to BIOS through this utility. Unlike floppy drives, where capacity is equivalent to drive type, several ***parameters*** are required here to uniquely identify a fixed disk. These parameters are obtained from the documentation furnished with the drive(s):

- **Number of Cylinders**
- **Number of Heads**
- **Sectors per Track**
- **Precompensation Cylinder**
- **Landing Zone (Park Cylinder)**

You do not need to understand these terms in order to successfully configure your drives. For an explanation, though, see the discussion at the end of this section.

PREDEFINED DRIVE TYPES 1-45

The fixed disk parameters are consolidated into a single ***type***, which is a numeric value in the range **1-45**. BIOS uses this number to select one of 45 built-in parameter tables (see appendix), each describing a different kind of drive. Since there are many hundreds of drives available today, this scheme remains as an artifact from early PC history.

USER PROGRAMMABLE TYPES 46 & 47

Two ***User Programmable*** drive types **46** and **47** enable you to manually enter custom parameters that describe any drive you have today or acquire in the future. Whenever the computer is booted, a table will be constructed in main memory with these parameters. If your computer has Shadow RAM capability and BIOS shadow is enabled, the table is generated in the EPROM image to assure 100% compatibility with all software products.

Note - *Novell 2.X requires BIOS to be shadowed when using types 46 & 47.*

AUTOCONFIGURE IDE DRIVES

Most IDE Drives (Integrated Drive Electronics) can be interrogated for their parameters. To do so, simply press **<?>** (Question Mark key). After a brief delay, the entire group of parameters will appear in their respective fields, and ***User Programmable*** Type **46** or **47** will automatically be selected. (Or an error will be reported if they cannot be sensed).

HIGH-SPEED DATA TRANSFER

Several options are available to increase data throughput. ***Standard*** transfer mode **0** conforms to original PC standards, and is compatible with all fixed disks. This method uses a hardware signal (called an *interrupt*) to transfer a single sector at-a-time. ***Poll*** mode **1** does not wait for the interrupt, but instead *polls* the drive for its readiness to transfer each sector. ***Block-Transfer*** mode **2** makes use of modern IDE drive capabilities to transfer a group of sectors in a single burst. While the preceding modes 0-2 perform industry standard 16-bit word transfers, ***32-Bit Block-Transfer*** mode **3** exploits the system board's 32-bit bus to achieve the highest transfer-rate possible.

SUPPORT BEYOND 1K CYLINDERS

If your fixed disk has more than 1K cylinders, you can instruct BIOS to ***translate*** the parameters. A drive with as many as 16K cylinders can be supported through this technique, otherwise the fixed disk interface is limited to the traditional 1K cylinders.

ANTI-VIRUS FEATURE

The ***Anti-Virus*** option is intended to provide a measure of protection against malicious programs which infect the main boot sector or low-level format (destroy) your data. Since viruses often gain entry when an infected floppy disk is booted, you should supplement this defense with the **C: 1st** boot order in the Boot-Sequence Utility. **WARNING: Many classes of viruses will not be detected, and even when a virus is detected, it may have already infected the disk, corrupted data, spread through a network, etc.**

Note - *You will need to disable this option while using certain fixed disk maintenance programs (eg, DOS FDISK), because their actions would be interpreted as a violation.*

Size - The capacity of the fixed disk is viewed here. This field may also be scrolled in numerical order to select any of the 47 drive types.

Size ... n/a	fixed disk controller not present
Size ... None	controller present, no fixed disk
Size ... 42.5M	disk configured with 42.5M capacity

Type - The numeric identifier that associates a drive with a set of parameters is selected here. While Types 1-45 reference predefined tables, values for Types 46 and 47 are programmable to describe any kind of drive. Press **<?>** for automatic setup.

Type ... n/a	fixed disk controller not present
Type ... None	controller present, no fixed disk
Type ... 1 - 45	predefined drive types
Type ... 46 & 47	user programmable drive types

Cylinders, Heads, ... Sectors - These five fields display the set of drive parameters corresponding to the selected drive Type (see above). When Type 46 or 47 has been selected, these fields can be edited.

Cylinders ... 967	number of cylinders
Heads 6	number of recording heads
Precomp 512	starting precompensation cylinder
Landing 967	landing zone cylinder to park heads
Sectors 17	number of sectors per track

Note - *If your documentation does not indicate a Precomp value, you can make* **None** *appear by entering the same number as the Cylinders value. Similarly, if your documentation does not indicate a Landing Zone, use the same value as* **Cylinders***.*

Translate - A large drive, up to 16K cylinders, can be fully utilized by enabling BIOS translation. Otherwise, the traditional 1K cylinder interface is employed.

Translate ... No	1K cylinders or less
Translate ... Yes	more than 1K cylinders

Xfer-Mode - Improved data throughput may result from one of the advanced Transfer Modes available. Modes 0-3 employ increasingly sophisticated techniques. Default Mode 0 complies with original PC standards.

Xfer-Mode ...	Standard
Xfer-Mode ... 1	Poll
Xfer-Mode ... 2	Block
Xfer-Mode ... 3	32-Bit Block

Anti-Virus - Write-protect the main boot sector by enabling this option. You will need to disable it prior to running fixed disk maintenance programs (eg, FDISK).

Anti-Virus ... Disable	allow writes to boot sector
Anti-Virus ... Enable	trap writes to boot sector

Discussion: Fixed Disk Naming Conventions

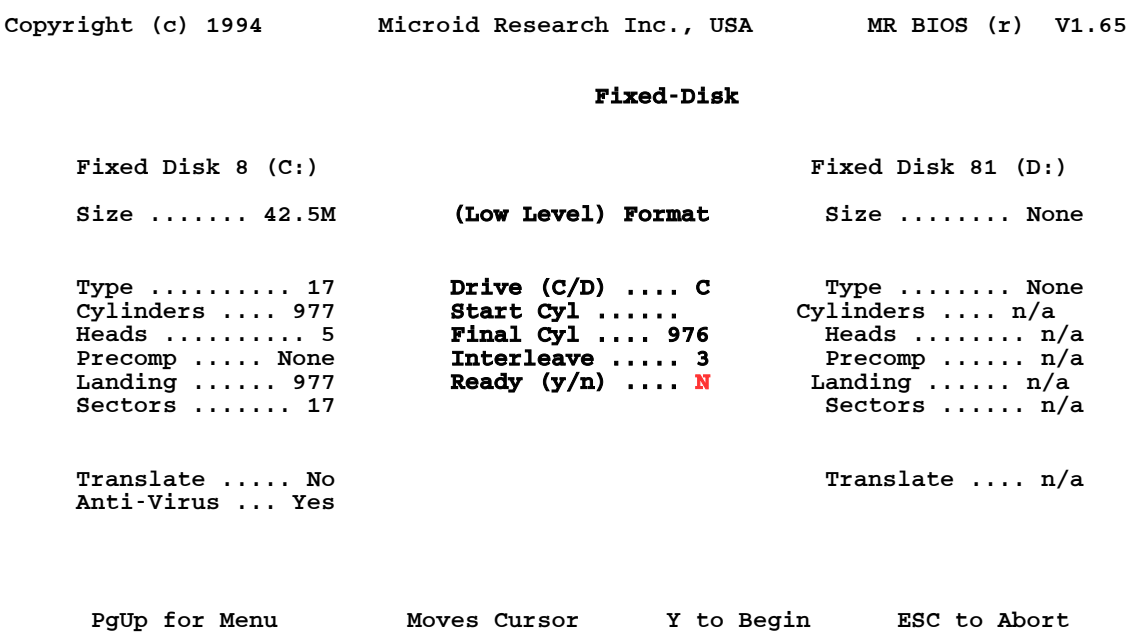
Although C: and D: are the most common names for the fixed disk units, it is possible to configure a system such that other drive letters are used. You may be familiar with the

concept of DOS Partitions, where a single fixed disk is subdivided into as many as four logical drives. In this case, they may be referenced as drives C, D, E, and F. As another example, you might have four floppy disk drives, named A-D. Your fixed disks might then appear as E and F. These examples demonstrate that DOS can assign the drive letters *dynamically*. In contrast, BIOS uses an invariant naming convention for the floppy and fixed disks. Floppy drives are referenced as Units 0,1,2,..., and fixed disks are accessed as Units 80,81,82,... Operating Systems and other low-level programming rely on this invariance when requesting disk services from BIOS. While you are configuring your drives through this Setup Utility, you should understand that BIOS is unaware of any partitions or other logical mappings which might affect the drive letters. The names C: and D: are used in order to simplify the discussion for the majority of readers. The actual physical Units 80 and 81 are being configured.

Discussion: Fixed Disk Parameters

A fixed disk unit contains one or more rigid disks with magnetizable surfaces. Data is recorded onto them by means of a recording **head**, where a single head is dedicated to each recording surface. The disks spin at very high speed (60 rotations *per second*), and the heads actually fly just above the sensitive media without touching it. Recording occurs in circular paths called **tracks**, and these tracks are subdivided into small, equal packets (typically 512 bytes) called **sectors**. Data is always written and retrieved in increments of whole sectors. Much like the rings seen in a tree trunk, concentric tracks are accessed by moving the heads to incremental positions along the disk's radius. The outer-most track is named Track 0, and the inner-most track nearest the spindle receives the highest number. Imagine a tin can passing through the entire stack of disks. Such a **cylinder** contains all tracks of the same number (name), and in fact, all the heads are attached to a single fixture which is stepped from cylinder to cylinder. As the fixture nears the center, the tracks have successively smaller circumferences and the data becomes increasingly crowded. In early technology fixed disks, special **precompensation** circuitry was enabled at a threshold track to predict and compensate for this density increase. Newer drives automatically manage this without any help from BIOS. Also, many older drives allowed the heads to rest on the recording surfaces when the drives were shut off. In order to transport the drive without damaging the media, the heads needed to be parked in a safe **landing zone**. Newer drives are designed to mechanically restrain the heads at power-off and therefore need not be parked. However, care should always be taken to avoid bumping and shock.

(This page intentionally left blank)



Low Level Format Utility Screen

LOW-LEVEL FORMAT UTILITY

WARNING: This utility can erase the entire content of your fixed disk. Its purpose is to establish the most primitive level of preparation which remains for the life of the disk. This procedure is usually performed in the factory and you generally need not be concerned with it. Unless you are absolutely certain that your fixed disk lacks this Low-Level Format, or are experiencing uncorrectable data errors that are attributable to Low-Level Format failure, you should never use this utility.

Before even considering the use of this utility, you should understand the concepts of the previous ***Drive Definition*** section, and your drive(s) must be declared properly.

Most PC computers use a DOS Operating System. In this environment, there are three steps (beyond Drive Definition and Low-Level Format) required to prepare a bootable disk:

- **FDISK utility to Partition the disk**
- **FORMAT utility to High Level Format the disk**
- **SYS utility to load DOS onto the disk**

If FDISK recognizes your fixed disk(s), then they are already Low-Level Formatted.

Note that DOS FORMAT is a High-Level Format procedure. This is different and should not be confused with Low-Level Format. The SYS step is optional. It can be combined into the FORMAT step by using the **/s** option on the command line (eg, FORMAT /s). It can be omitted altogether if the disk is not going to be booted. Consult your DOS

manual for instructions.

Low-Level Format Instructions

The Low-Level Format column (center of screen) is made accessible by pressing the pair of keys **<Ctrl+F>**, and is exited by pressing either **<Esc>** or **<PgUp>**. The cursor will skip over this column, remaining on the Drive Definition fields, until **<Ctrl+F>** is pressed.

Be aware that the number of heads, sectors, and the maximum cylinder are obtained from the Type parameters in the neighboring ***Drive Definition*** column. You must correctly configure the target drive before attempting to format it here.

The **Start** and **Final** cylinders can be selected from the range {0 to Total-1}. (If there are ***n*** cylinders on the disk, they are numbered **0** to ***n-1***). Notice the disk can be formatted in reverse order by selecting the Start cylinder to be greater than the Final cylinder.

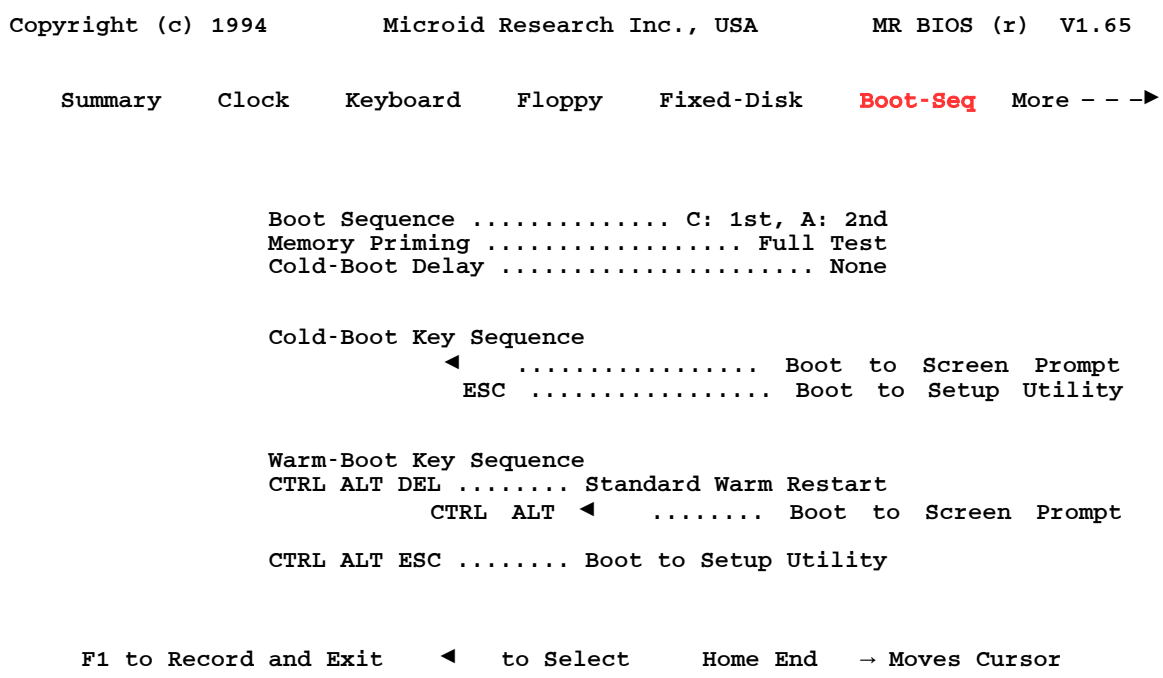
The **Interleave** field controls the order which the sectors are written onto the disk. A 1:1 interleave produces the intuitive sequence (1,2,3,...). 2:1 interleave produces (1,3,5,...). 3:1 interleave yields (1,4,7,...). Etc. While 1:1 interleave gives the best theoretical transfer rate, it will degrade performance unless you have a controller card with cache or track-buffer. If not, select the traditional 3:1 interleave.

Drive, Start Cyl, ... Interleave - Initially, these four fields will default to values that format the entire disk with a 3:1 interleave. The fields can be edited if a different cylinder range or interleave factor is desired. Many newer controllers recommend a 1:1 interleave, while traditional equipment uses 3:1 interleave.

`Drive (C/D) ... C`	target drive
`Start Cyl .....`	starting cylinder
`Final Cyl ... 966`	ending cylinder
`Interleave .... 3`	interleave factor

Ready (y/n) - Press **Y** with the cursor on this field to begin ***immediately and without hesitation***. You can abort at any time by pressing **<Esc>**, however, any formatting that has already occurred cannot be undone.

`Ready (y/n) ... N`	Press "Y" to begin



Boot Sequence Setup Utility Screen

BOOT SEQUENCE SETUP

Three system startup functions are configured here.

Boot Sequence specifies the order your disk drives are accessed when loading the Operating System. While the most popular boot order is **C: 1st**, you may also select **A: 1st**, **Network 1st**, **Auto-Search**, or **Screen Prompt** according to your preference. The majority of this section is devoted to describing these options.

Memory Priming specifies the memory test that executes during powerup; normal, fast, or skip it altogether. The **Full Test** should normally be selected. It conducts a rigorous memory test at the rate of approx. 1 MB/sec. This is relatively slow when compared to approximately 8 MB/sec achieved by **Quick Scan**, which does little other than *prime* the memory and parity by writing zeros to it. You can bypass the memory test entirely by selecting **Skip Test** here, or by striking **<SpaceBar>** during the POST memory test.

Cold-Boot Delay can provide additional powerup time required by some slow mechanical devices. As an example, some IDE drives malfunction if accessed within a few seconds after powerup. They can be accommodated by enabling a several second delay. Unless you experience such problems, though, disable the delay by selecting **None**.

BOOT SEQUENCE

Each time you turn on your computer or press **<Ctrl+Alt+Del>** to restart it, an Operating

System (such as DOS) is loaded from one of the disk drives. When several drives are installed in your computer, you can select the *order* that BIOS searches the disks for the Operating System.

Because the traditional boot devices are Floppy A: and Fixed Disk C:, DOS and most other O/S can only boot A: or C:. Yet you can arrange through this utility to boot *any* disk in the system, say B: or D:. In order to do this, though, BIOS needs to reassign the drive letters to remain compatible with tradition. In general, a Floppy that boots is named A:, and a Fixed Disk that boots is named C:.

A: 1st, C: 2nd

Historically, only A: or C: could be booted, and the order these drives were checked was not selectable. Floppy A: would be booted if it contained a diskette. Otherwise, Fixed Disk C: would be booted. This is the industry standard **A: 1st, C: 2nd** order.

Auto-Search

In computers with more than one Floppy, **Auto-Search** extends the standard **A: 1st** order to search *all* Floppies 0,1,2,3 (**A:, B:,...**) before defaulting to Fixed Disk C:. This is useful if Drive A: is a different size than the diskette you want to boot. The Floppy that ultimately boots becomes A:, and the other Floppies are renamed in a round-robin fashion. In a typical two Floppy system, this reduces to simply swapping A: and B: assignments.

C: 1st, A: 2nd

In computers with a Fixed Disk, the **C: 1st, A: 2nd** order can be selected to bypass the Floppy Drive access. This option promotes fastest bootup, and eliminates the annoying *non-bootable diskette* error that occurs when a diskette is unintentionally left in the Floppy Drive. It also eliminates one opportunity for a virus to infect your computer.

From time to time, you may still need to boot a Floppy. This can be done conveniently (*without entering Setup*) by pressing **<Ctrl+Alt+Enter>**. See **Screen Prompt**, below.

Network 1st

It is a common practice in businesses to interconnect PC's through cables and hardware that is known collectively as a *network*. Each computer in the network contains a Network Adapter card, and often, this Adapter contains a *boot-ROM* that loads the O/S directly from the network (instead of booting from disk). In these installations, Fixed Disk C: is usually left unbootable because it would be accessed (booted) before the boot-ROM gains control. The **Network 1st** option bypasses all disk access completely, allowing you to boot directly from the network even when Fixed Disk C: is bootable.

From time to time, you may still want to boot from disk. This can be done conveniently (*without entering Setup*) by pressing **<Ctrl+Alt+Enter>**. See **Screen Prompt**, next.

Screen Prompt

A menu can be made to appear on the CRT which requests your explicit selection of the boot device. When **Screen Prompt** is programmed as the default boot method here, this menu will be displayed each time the computer is booted. You can also invoke this **Screen Prompt** during runtime (eg, from DOS) by pressing **<Ctrl+Alt+Enter>**. Be aware that this has the same effect as **<Ctrl+Alt+Del>** - it aborts any current program and warm-boots the computer.

```
Press F1 to Boot Floppy
     F2 to Boot Fixed
     F3 to Boot Network
```

Pressing **F1** actives the **Auto-Search** mode described above, and **F2** activates the **C: 1st** mode. Appearing only when a Network Boot-ROM has been detected, the **F3** option will activate the **Network 1st** option. You may also press any key **A-F** to explicitly select drives **A: - F:** (according to DOS 5.0 conventions), or the letter **N** to boot from Network. When explicitly selecting a disk A-F, note that whichever Floppy is booted becomes A:. Whichever Fixed Disk is booted becomes C:.

Preparing Fixed Disk D:

In order to make Fixed Disk D: bootable, you will need to SYS it (put DOS on it), and have FDISK mark its partition *Active*. Unfortunately, FDISK refuses to do it, complaining "Only partitions on Drive 1 can be made active". To get around this, have a bootable floppy ready with FDISK on it, then try to boot drive D: (by invoking the Screen Prompt, then pressing "D"). When this fails and the Screen Prompt reappears, boot the floppy. At this point, you will discover that BIOS has swapped C: and D: in the same fashion as if D: had successfully booted. Now you activate the partition on drive C:, and FDISK is *happy*.

Boot Sequence - The order which the drives are accessed at boot-time is programmed in this field. Select the option that best serves your needs.

Boot Sequence ... A: 1st, C: 2nd	floppy 0 then fixed
Boot Sequence ... C: 1st, A: 2nd	fixed then floppy 0
Boot Sequence ... Auto-Search	all floppies, then fixed
Boot Sequence ... Network 1st	network boot-ROM
Boot Sequence ... Screen Prompt	select from menu

Note - *A prompted boot can also be manually invoked by pressing* **<Ctrl+Alt+Enter>** *during runtime (from DOS), or by pressing* **<Enter>** *during the cold-boot memory test. (This is a convenient way to occasionally boot from floppy when the* **C: 1st** *order is selected here as the default).*

Memory Priming - During Cold-Boot, BIOS conducts a rigorous memory test to verify its integrity and also to prepare it for use. In large memory systems, bootup time can be significantly reduced by selecting **Quick Scan** to initialize the memory without extensive testing, or **Skip Test** to bypass testing altogether.

Memory Priming ... Full Test	complete memory test
Memory Priming ... Quick Scan	initialize only

`Memory Priming ... Skip Test` no test whatsoever

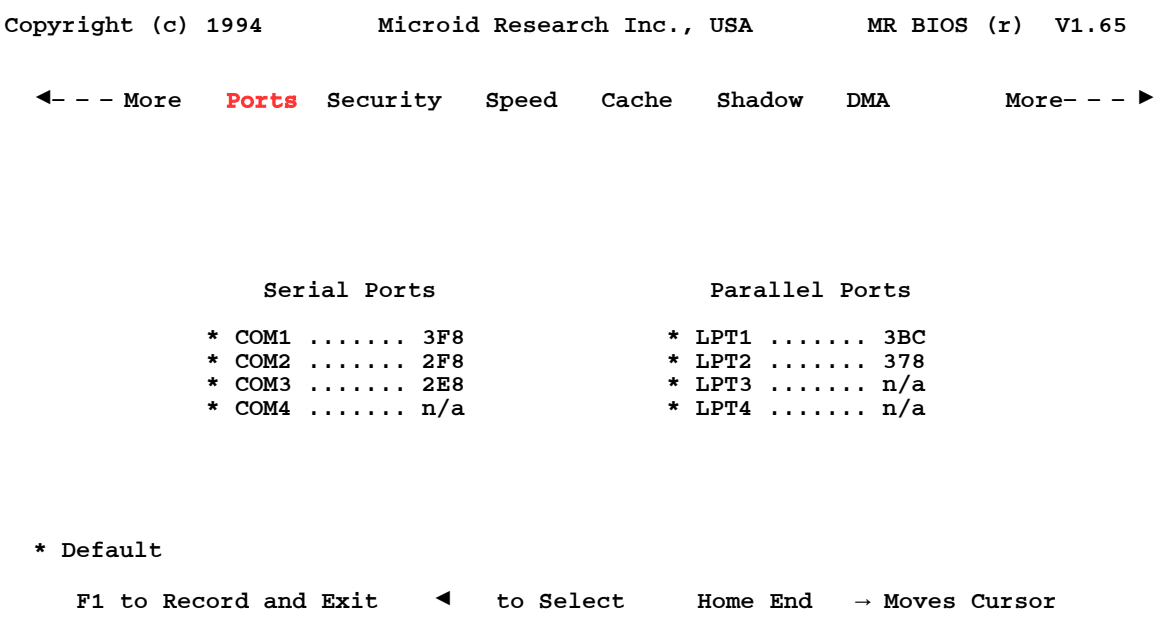
Note - *Press **<SpaceBar>** during cold-boot Memory Test to terminate it.*

Cold-Boot Delay - A delay before Power-On-Self-Test (POST) may be needed to allow proper initialization of various slow mechanical devices. This is especially true of certain IDE drives that are unprepared for the unusually swift execution of this BIOS. If you experience powerup difficulties, try a delay of 1 to 30 seconds.

`Cold-Boot Delay ... None` no delay before POST

`Cold-Boot Delay ... 5 Sec` 5 secs before POST

Note - *Unless required, select **None** to avoid inducing an unnecessary delay.*



Ports Setup Utility Screen

PORTS SETUP

This utility allows you to view, and optionally to change the Serial Port and Parallel Port configuration of your computer. During poweron, BIOS identifies these ports and *assigns* them the device names COM1-COM4 and LPT1-LPT4 according to industry conventions. In special cases, you may want to rearrange the default assignment, or even to disable the ports altogether. Usually, no changes need be made, and you should simply confirm that an asterisk (*) appears next to each field (default settings).

When viewing this screen, the left-hand side of each field shows the familiar device name COMx or LPTx. The right side reveals the hardware I/O Port Address that is currently associated with the COM/LPT name. (Or "n/a" if no port is assigned there). While the cursor illuminates a target field, you can scroll the possible I/O Port Address choices by pressing **<SpaceBar>**. Unless you are simply disabling a port, the rearrangement will involve modifying two or more fields. At some intermediate point in the process, a single I/O Address may exist in multiple fields. Prior to exiting this utility, make sure no duplicate I/O Address assignments remain, else they will be deleted.

COM1 (2,3,4) - The I/O Port Address associated with each Serial ("COM") Port can be changed here, or a Port can be disabled altogether by selecting **n/a**.

COM1 ... 3F8	primary port
COM2 ... 2F8	secondary port
COM3 ... 2E8	another port
COM4 ... n/a	disabled

LPT1 (2,3,4) - The I/O Port Address associated with each Parallel ("LPT") Port can be changed here, or a Port can be disabled altogether by selecting **n/a**.

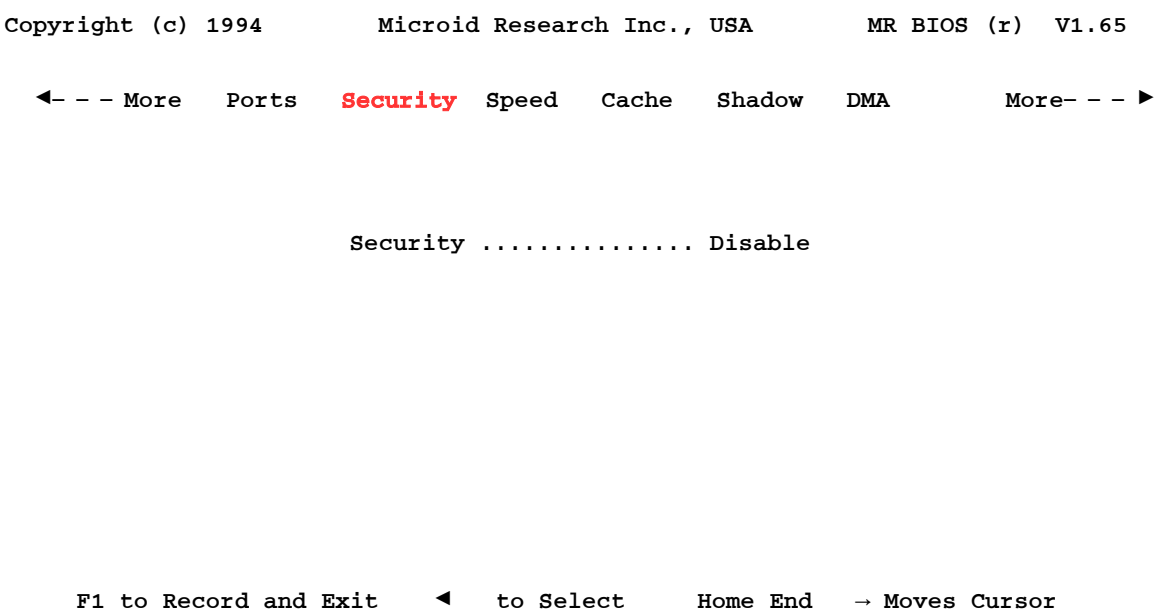
LPT1 ... 3BC	on mono card
LPT2 ... 378	primary port
LPT3 ... 278	secondary port
LPT4 ... n/a	disabled

Discussion: Defacto-Standard Serial Ports

The original PC computers were equipped with up to two serial ports, although the system BIOS was actually designed to service as many as four. The *primary* serial adapter uses a block of I/O Addresses based at 3F8, and the *secondary* port resides at I/O Address 2F8. These original two serial ports are typically accessed using logical device names **COM1** and **COM2**, together being referred to as the ***standard*** serial ports. In order to support more than two serial devices at a time (eg, printers, modems, FAXes, scanners), the PC industry has come to recognize a group of additional I/O ports known collectively as ***defacto-standard*** ports. They are often (mis)documented as **COM3** and **COM4** with the assumption that they will be installed in systems that already contain two serial ports. This leads to the (confusing) table of the six most widely available serial ports:

Name	I/O	IRQ
COM1	3F8	4
COM2	2F8	3
COM3	3E8	4
COM4	2E8	3
COM3	3E0	4
COM4	2E0	3

During poweron, BIOS *dynamically* allocates the logical device names **COM1-COM4** to each serial port it encounters, in numerically increasing order. Eg, if only one serial port is present, it is named COM1 (without regard for the I/O Address it occupies). Thus, the COM device name cannot be predicted by an adapter card manufacturer before you install it in your computer. The characteristic property of the port is its hardware I/O Address.



Security Setup Utility Screen

SECURITY SETUP

Three Password Security options are available to protect your computer from unauthorized use: ***Setup Only***, ***Powerup/Setup*** and ***Bootup/Setup***. When one of these Security levels is armed and that condition is triggered, the user will be prompted to enter a Security Code (password). Three chances are given to match the key password that was originally established in this Utility. Upon failure, the system is halted and an alarm is sounded.

Within this utility, you can:

- **Arm Security and Establish a Password**
- **Disarm Security (Eliminate Password)**
- **Change the Security Option**
- **Change the Password**

The **Video Jumper** (dipswitch) on the motherboard functions as a **Master Override**, either permitting or unconditionally disabling this Security feature. (It is not needed by BIOS for video detection). You can use this override in the event you forget your password.

- **MONO -** Unconditionally disables the Security Feature.
- **COLOR -** Permits Security to be enabled through this utility.

SETUP ONLY

As the name suggests, the **Setup Only** option prevents unauthorized access to the Setup Utility. When the Setup Utility is invoked, a Security prompt will appear at the

bottom of the Summary Screen. Access to other Setup menus (ie, the entire configuration) will be denied unless a valid Security Code (password) is entered at this prompt.

POWERUP/SETUP

In addition to restricting Setup Utility access, the **Powerup/Setup** option also prevents unauthorized entry into the computer from powerup. Immediately after a system powerup (or pushbutton reset), the Security Prompt will appear in the center of the screen. Access to the computer will be denied unless a valid Security Code (password) is entered.

BOOTUP/SETUP

The most extreme setting, **Bootup/Setup**, triggers the Security Prompt *every* time the computer is booted. Beyond Setup and Powerup protection, this Security level also includes the standard warm-start **<Ctrl+Alt+Del>**.

KEYSTROKES

When a password is being typed, each keystroke is echoed to the CRT as an asterisk (*). Practically any combination of 0 to 10 characters can be specified, including Function keys, Shift, Alt and Ctrl sequences. Notice that alphabetic characters are *case sensitive* (eg, "B" is different from "b"). Three special keys are reserved for editing:

- **BackSpace** - Delete last typed character
- **Enter** - Final keystroke of password
- **Esc** - Restart (or abort in special cases)

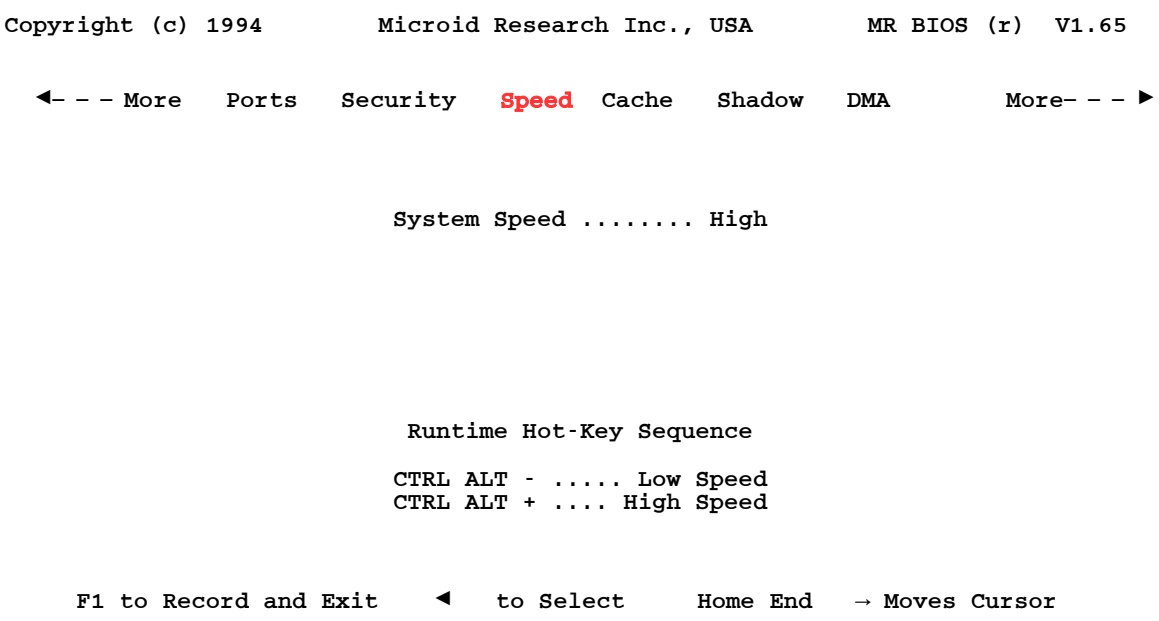
Security - The currently selected Security option appears in this field. Other choices are selected by scrolling through the options, then pressing **<Enter>** when the desired state is visible. If Security is becoming armed or the Code is being changed, you will be prompted to establish a new password (see below).

`Security ... Disable`	security disarmed
`Security ... Setup Only`	protect configuration
`Security ... Powerup/Setup`	both powerup & config
`Security ... Bootup/Setup`	warm-boot too
`Security ... Change Code`	change password

Select Security Code - Type a new password consisting of 0 to 10 characters, followed by **<Enter>**. You will then be instructed to retype it for verification. If you abort the mode by pressing **<Esc>**, all changes are abandoned.

`Select Security Code: **********`	enter 0 to 10 chars
`Verify Security Code: **********`	enter same code again

(This page intentionally left blank)



Speed Setup Utility Screen

SYSTEM SPEED SETUP

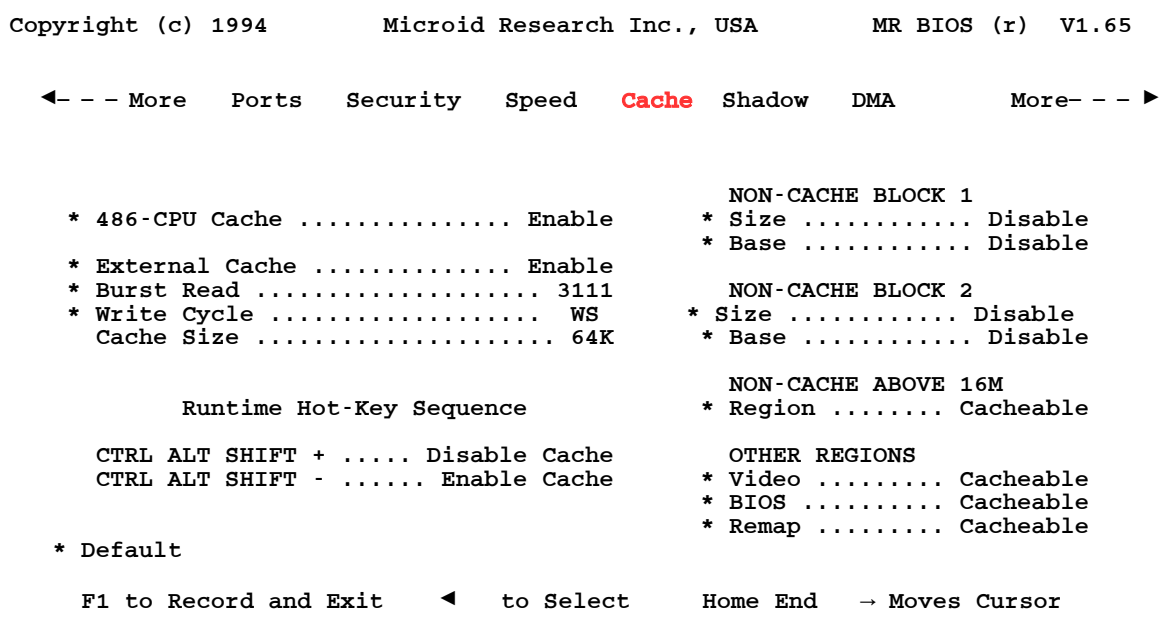
CPU speed control is a *bells-and-whistles* feature that is becoming impractical (and often impossible) to implement in today's high performance systems. If your computer does not offer this feature, you can achieve a similar effect by disabling the cache. (See the Cache Setup Utility). Otherwise, this section may be of interest to you.

The operating speed of the CPU is configured here. **High** speed (turbo) is the usual choice, since this will maximize the system performance. The other option, **Low** speed, reduces performance to simulate the original, slower PCs. The selection you make here will be effected each time the system is booted.

You can change the speed *on the fly* during normal runtime by pressing **<Ctrl+Alt+Plus>** for High speed, or **<Ctrl+Alt+Minus>** for Low speed. If your computer has a front panel turbo button, either this Setup selection or the pushbutton will dominate. Experiment with the four possible combinations to learn how your system reacts.

System Speed - Select the computer's operating speed according to your preference.

System Speed ... Low	slow speed
System Speed ... High	fast (turbo)



Cache Setup Utility Screen (sample)

SYSTEM CACHE SETUP

If this Cache Setup Utility does not appear in your system, skip over this section. If it does, but an optional cache is not installed in your (286 or 386) system, then simply make sure all fields here are disabled. Otherwise, the information here will be of interest to you.

The figure appearing above is an example, and may not exactly match your Cache Setup Utility screen. Cache implementations vary significantly from system to system, requiring a *custom* Setup Utility to fully exploit the unique properties of each particular design. The information in this section is therefore intended to describe general concepts and terms that will be encountered within this Utility.

The purpose of a cache is to optimize system performance by increasing throughput of the memory subsystem. This is achieved through use of a small quantity of high speed Static RAM (SRAM). As data is fetched from slower main memory, it is copied into the faster SRAM. Subsequent references to this data are directed to the SRAM, occurring more swiftly than is possible from main memory.

The main objective of this Utility is to allow you to **enable** the cache(s) in your system. Depending on the complexity of your particular design, you may also be able to customize the cacheable ranges, exclude certain regions from being cached, control SRAM access timing, or even select the cache algorithm. As a general rule, you will

obtain best results by making all memory present in your computer cacheable, disabling any non-cache blocks, and selecting the most aggressive timing parameters.

286 & 386 SYSTEMS

While 286 systems rarely (if ever) possess a cache, many 386 designs offer an optional cache that is integrated into motherboard logic. In these designs, a **System Cache** field will be present to enable/disable the cache.

486 & 586 SYSTEMS

These CPUs contain a small, but extremely efficient **X86-CPU Cache**. There may also be a supplemental **External Cache** implemented in motherboard logic, analogous to the 286/386 design discussed above. You will generally be able to control both these caches independently.

CACHE SIZE

This field indicates the amount of SRAM in the system-board cache. (In 486+ systems, this is the *External* cache size). Usually, BIOS autodetects this value and it is ***view-only***.

TIMING / WAIT STATES

Cache accesses occur in increments of CPU timing units called **Wait States**. Numerically lower WS values cause faster access and therefore better performance. (Eg, 0 WS is faster than 1 WS). Cache SRAM has a minimum access time requirement that must be enforced, otherwise the system will malfunction. Thus, the objective here is to select the ***least*** number of Wait States that will not cause SRAM failure.

BURST TIMING

486 CPUs optimize external memory fetches by reading a group of four Double-Words (DWORDS) in a quick succession known as a **Burst**. The first of the four memory cycles includes delivery of the starting memory address, whereas the system board logic predicts the latter three cycles' address sequence. In general, each of the final three cycles will require an equal amount of time, and the first cycle will take longest. Timing is specified by a four digit number XYYY, where X represents the first cycle, and Y represents each of the final three cycles. Burst timing is given in increments of CPU timing units called **T-States**, with the fastest possible timing being 2111.

586 CPUs not only perform Burst-Reads, but can also perform Burst-Writes. A separate Burst-Write field may therefore be present for these CPUs.

ASSOCIATIVITY

Since cache is smaller than main memory, old data in the cache will be replaced by new data from time to time. If this happens persistently, a condition called *thrashing* exists which reduces the efficiency of the cache. A design with programmable **Associativity** will allow you to select an improved cache algorithm based on the simple concept that *if one is good, two are better*. A **2-Way-Set-Associative** cache reduces data thrashing

seen with the simpler **Direct-Map** design. However, the added complexity of the 2-Way design can slow its response, requiring additional Wait States in high-speed systems. In this case, the advantage is lost and possibly the Direct-Map option will perform better.

WRITE-THRU / WRITE-BACK

The singlemost important function of cache is the accelleration of Memory Read cycles. A **Write-Through** cache accomplishes this with little consideration for Memory Write cycles, which are always directed System Memory, and, simultaneously to Cache if it is a *hit*. If the write to System Memory could be omitted, further throughput improvement would be realized. While it is impossible to avoid System Memory writes altogether, a **Write-Back** (or Copy-Back) cache suppresses them until absolutely necessary. Ordinary write-hits are captured in cache and are not issued to System Memory. Whenever the CPU fetches data that is not in the cache (a *miss*), a two step *line replacement* occurs where first the cache line is flushed to memory, followed by reading the new data into the cache. A read-miss *penalty* is incurred when the write cycle is induced. By monitoring those cache lines that have become modified by writes (are *dirty*), unnecessary write-back (and its penalty) can be eliminated. In practice, it is found that very little is gained by qualifying dirty cache lines. Many modern designs have abandoned it, employing simpler (and more cost effective) **Blind-Write-Back** strategy.

NON-CACHE BLOCKS

In most cases, you will want to **disable** this function. A **Non-Cache Block** is a region of memory that is specifically excluded from entering the cache. This defeats the operation of cache in that region, and programs will suffer diminished performance when accessing that memory. Under rare circumstances, however, this may be exactly what is needed to make a malfunctioning program work.

> **Note -** *Systems with an Austek cache controller should construct a Non-Cache Block over the video buffer at A0000-BFFFF.*

NON-CACHE ABOVE .../ CACHEABLE RANGE

If there are provisions to specify the maximum cacheable range, be sure to specify an address range no greater than the actual amount of memory in your computer. Usually, BIOS autoselects this field correctly, and an asterisk (*) will appear next to the correct value when it is selected.

CACHEABLE REGIONS

In general, it is desirable to **enable** all regions that can be cached. You may want to experiment to see if improved performance actually results. Also, you will be able to disable the region should a software compatibility problem occur.

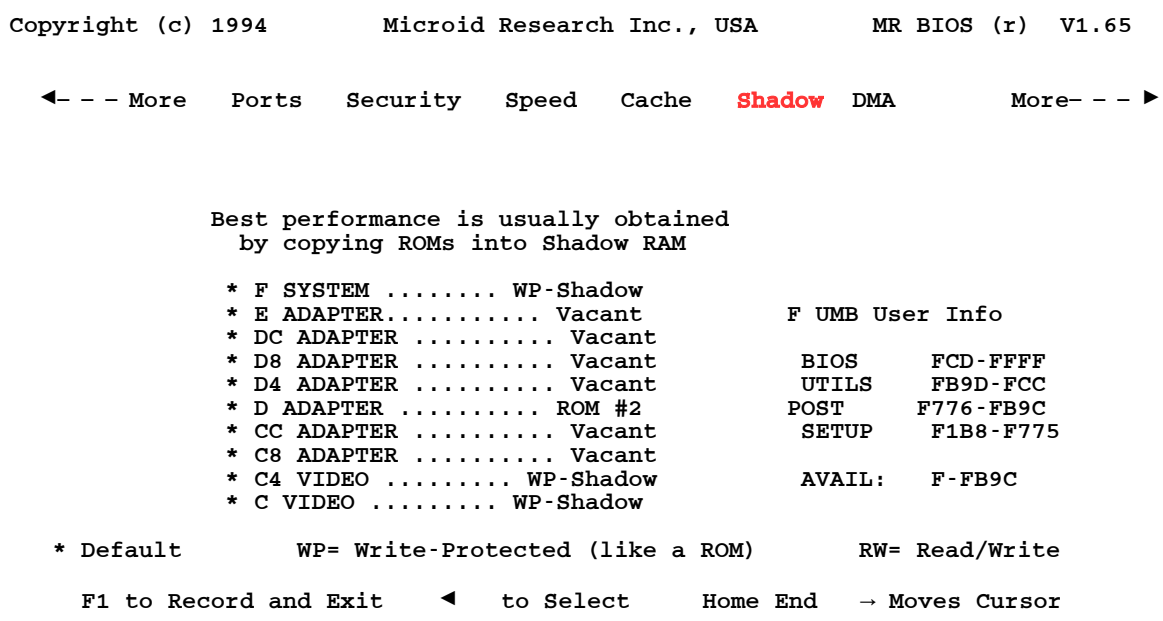
CACHEABLE VIDEO

When present, this field refers to the Video Adapter ROM that contains code to operate your VGA (or equivalent) graphics card. Usually, this selection will have no effect unless the video ROM is also Shadowed. (See Shadow-RAM Setup, segments C000 and C400).

CACHEABLE REMAP

The term REMAP refers to the unused Shadow-RAM that is allocated to the top of the

Extended Memory. BIOS automatically performs such a remapping to optimize memory use. When this field appears, it is because the system board manages this block of memory differently than the rest. If you do not explicitly make it cacheable, then it will not be cached. Note that this field has no effect unless memory has in fact been remapped.



Shadow RAM Setup Utility Screen

SHADOW RAM SETUP

Most modern PC style computers provide a mechanism to increase the execution speed of programming that resides in Read-Only-Memory (ROM). This mechanism involves copying the ROM into faster main memory, then substituting that memory image for the original ROM. Memory that is dedicated to this use is called **Shadow-RAM**. BIOS and VGA Adapters are two main examples of ROMs that demonstrate significant performance gains when they are shadowed.

Since ROMs are by definition Read-Only, it is usually desirable to **Write-Protect** the Shadow-RAM. However, Shadow-RAM can also be used as general purpose memory by certain programs, in which case it should be enabled as **Read-Write** memory. While most Adapter ROMs can be shadowed either way, some permit only the RW or WP option, and a rare few cannot be shadowed at all. You may need to experiment a little.

When viewing this Setup screen, you will observe that unshadowed segments display **Vacant** if no Adapter ROM is present there. Unshadowed segments containing a ROM will indicate **ROM #n**, where **n** is a number from 0-9. Each ROM is assigned a unique number so they can be distinguished. Note that if a single ROM spans several segments, then the same ROM #n will appear multiple times. ROM #0 is always the system BIOS.

Shadow-RAM is obtained from a gap in the otherwise contiguous memory space of the computer. The 384K region between the 640K and 1 Megabyte boundaries is occupied not by memory, but instead by ROMs, video memory, and possibly other system-level

devices. The memory that *should* appear there is simply inaccessible and unused. One way to make use of this lost memory is to activate it as Shadow-RAM. Certain designs can also ***remap*** a portion of this 384K into the Extended Memory pool, provided it is not already enabled as Shadow-RAM. In most designs with this capability, remap will be prevented if *any* Shadow segment is enabled in the D000 through E000 regions. View the Extended Memory field in the Setup Summary screen to see how (or if) a particular Shadow configuration affects your computer's memory.

F SYSTEM - The system BIOS occupies this 64K segment. For best results, always enable WP-Shadow here.

F SYSTEM ..... ROM #	BIOS is NOT shadowed
F SYSTEM ... WP-Shadow	read-only BIOS shadow

Note - *Fixed Disk Types 46 & 47 (User Programmable) can be used with Novell 2.X only when F000 is shadowed.*

C8 thru E ADAPTER - If present, Adapter ROMs for non-video devices (eg, disk controllers or LAN adapters) are found in one or more of these seven segments. While C800 - DC00 are 16K segments, E000 is 64K.

E ADAPTER ..... Vacant	no ROM present
DC ADAPTER ..... Vacant	"
D8 ADAPTER ..... Vacant	"
D4 ADAPTER ..... Vacant	"
D ADAPTER ... RW-Shadow	writable shadow
CC ADAPTER ..... ROM #2	ROM not shadowed
C8 ADAPTER ..... Vacant	no ROM present

C & C4 VIDEO - Video Adapter ROMs (eg, VGA) usually occupy both of these 16K segments. They appear collectively as **ROM #1** when shadow is disabled. Best performance is usually obtained when shadow is enabled here.

C4 VIDEO ..... ROM #1	video ROM continued
C VIDEO ..... ROM #1	unshadowed video ROM
C4 VIDEO ..... WP-SHADOW	video shadow continued
C VIDEO ..... WP-SHADOW	start of video shadow

UPPER MEMORY BLOCKS

Many programs are confined to Base Memory because Extended Memory is difficult to access. (It requires *Protected Mode* software). A maximum of 640K can be installed as Base Memory, thus limiting the amount of device-drivers, TSRs, and programs that may co-reside in this space. Various *Memory Manager* programs are able to effectively increase the Base Memory by reclaiming unused gaps between the 640K and 1 MegaByte boundaries. When program memory is mapped into this region (traditionally reserved for BIOS, ROMs and the like), it is referred to as an **Upper Memory Block (UMB)**.

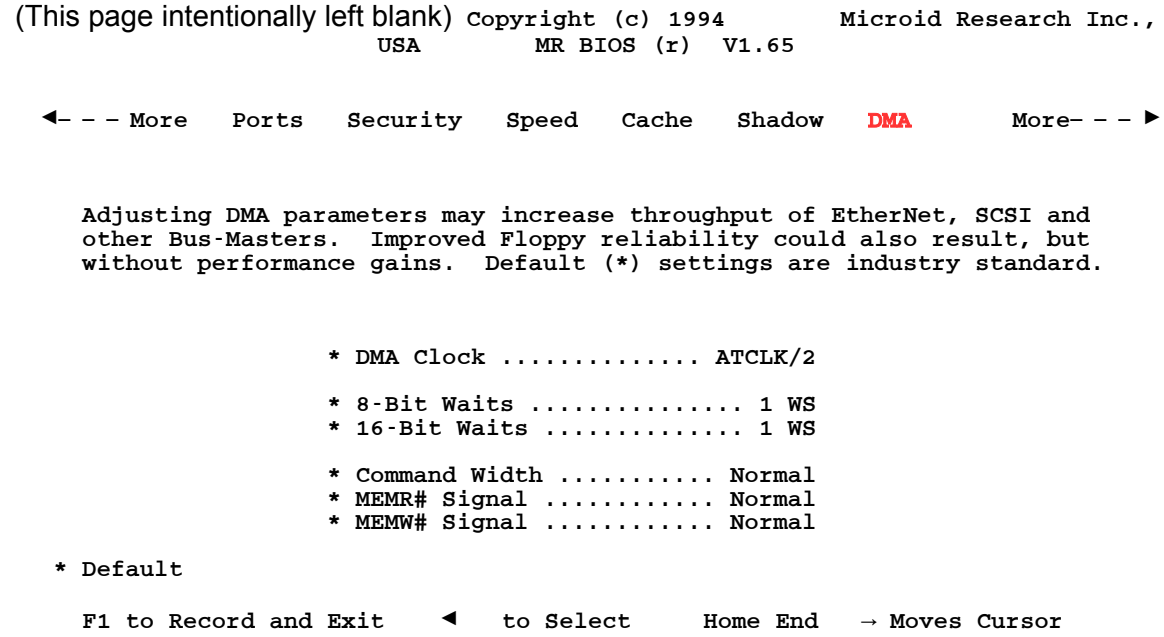
F UMB User Info - Approximately the bottom 3/4 of the BIOS EPROM contains expendable code that may be reclaimed as UMB space. A view-only breakdown of its content appears on this Setup screen for your convenient reference. You will need to furnish this information to your Memory Manager software in order to

create a UMB in the F000 BIOS region.

BIOS ..... Fzzz-FFFF	vital runtime BIOS
UTILS ..... Fyyy-Fzzz	speed & cache on/off
POST ..... Fxxx-Fyyy	powerup code
SETUP ..... Fwww-Fxxx	Setup Utility
AVAIL ..... F-Fyyy	available for UMBs

The critical runtime programming in the **BIOS** section cannot be assigned to UMB space. The **UTILS** section should not be assigned to UMBs either, since the speed and cache hot-key functions **<Ctrl+Alt+Plus>** and **<Ctrl+Alt+Minus>** are contained in this section. Both the **POST** and **SETUP** regions may always be reclaimed for UMBs since they contain only powerup and boot-time code. The **AVAIL** field shows the entire recommended range beginning at the bottom of the F000 EPROM.

(This page intentionally left blank)



DMA Setup Utility Screen

DMA SETUP

This menu selection appears only in systems containing an 82C206 Integrated Peripheral Controller (IPC), or equivalent part. In such systems, this Utility permits optional fine tuning of the DMA subsystem. Usually, no adjustments need be made, and you should simply confirm that an asterisk (*) appears next to each field (default settings).

All PC computers contain a Direct Memory Access (DMA) controller, either as a discrete component or incorporated in large-scale logic. This device allows various peripherals to bypass the system CPU and access memory directly. In typical installations, DMA is used only by the floppy disk controller. Add-on cards that *might* use DMA services include Networks, SCSI, tape drives, and special feature adapters like music synthesizers.

In certain cases, you will be able to improve throughput to/from an add-on card that uses DMA by adjusting parameters here. Or, an adapter card that refuses to operate correctly may be made to work by adjusting DMA timing more to its liking. As a general rule, high performance cards may benefit from a fast DMA **Clock**, and slower cards may require more **Wait States**. The other parameters are extremely technical, and you will need to consult timing diagrams in order to fully understand them.

DMA Clock - The clock to the DMA controller is derived from ATCLK (AT-Bus Clock).

ATCLK divided by 2 is the industry standard (default) setting. You can select the unreduced clock, ATCLK divided by 1, for faster operation.

*** DMA Clock ... ATCLK/2**	standard DMA clock
DMA Clock ... ATCLK/1	fast DMA clock

8-Bit Waits - The number of Wait States in an 8-bit DMA cycle is programmable from 1 (the default) to a maximum of 4. This field affects floppy operation.

*** 8-Bit Waits ... 1 WS**	standard wait state
8-Bit Waits ... 4 WS	maximum wait states

16-Bit Waits - The number of Wait States in a 16-bit DMA cycle is programmable from 1 (the default) to a maximum of 4. This field does ***not*** affect floppy operation.

*** 16-Bit Waits ... 1 WS**	standard wait state
16-Bit Waits ... 4 WS	maximum wait states

Command Width - A "Normal" DMA transfer cycle requires 4 clocks to complete, but can be optionally "Compressed" to 3 clocks.

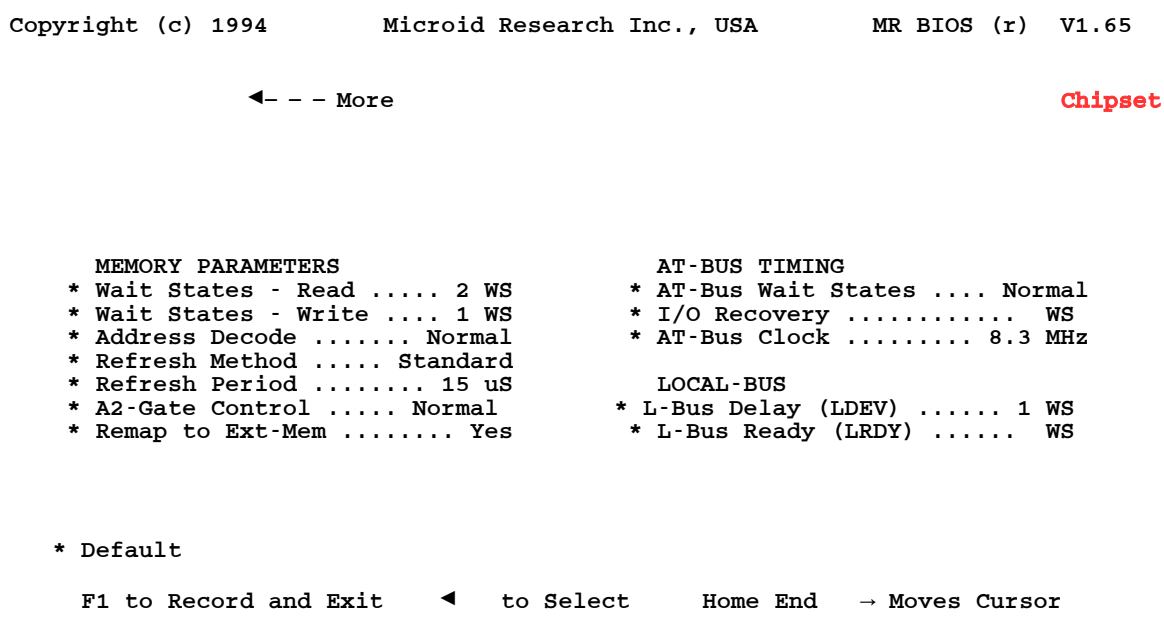
*** Command Width ... Normal**	default width
Command Width ... Compress	compressed width

MEMR# Signal - The MEMORY READ control signal can be configured to start one clock cycle earlier than its normal position.

*** MEMR# Signal ... Normal**	standard MEM READ
MEMR# Signal ... Early	start MEM READ early

MEMW# Signal - The MEMORY WRITE control signal can be configured to start one clock cycle earlier than its normal position.

*** MEMW# Signal ... Normal**	standard MEM WRITE
MEMW# Signal ... Early	start MEM WRITE early



Chipset Setup Utility Screen

CHIPSET SETUP

If this Chipset Setup Utility does not appear in your system, skip over this section. When it is present, this Utility permits optional fine tuning of certain chipset parameters that are very technical in nature. The preset values denoted with an asterisk (*) are designed to provide efficient, trouble-free operation, but can be changed according to custom needs. Usually, no adjustments are needed and you should confirm that an asterisk appears next to each field (default settings).

The figure appearing above is an example, and may not exactly match your Chipset Setup Utility screen. Chipsets vary significantly from system to system, requiring a *custom* Setup Utility to fully exploit the unique properties of each particular design. The information in this section is therefore intended to describe general concepts that will be encountered within this Utility.

In modern designs, practically the entire PC/AT motherboard logic is contained in just a few large chips known simply as the ***chipset***. Its primary responsibility is to make your computer operate in a PC/AT compatible fashion.

The chipset is typically programmable by BIOS, allowing it to be adapted to a broad range of designs and operating environments. Most chipset functions are presented elsewhere in the Setup Utility as standard BIOS features. The more technical core-logic functions, including Memory, AT-Bus, and Local-Bus timing are managed here.

MEMORY TIMING

Dynamic Random Access Memory (DRAM), the main memory in your computer, is rated according to minimum access-time requirements. If the CPU is allowed to access the DRAM too swiftly, the data will become corrupted and/or a *Parity Failure* will be reported. In reverse, if the DRAM is accessed very slowly, system performance will suffer. The objective is therefore to select the most aggressive timing that doesn't result in failure.

DRAM access speed is usually selected in terms of a CPU timing unit called a **Wait State**. Numerically lower WS values cause faster access, thus better performance. (Eg, 0 WS is faster than 1 WS). Another way that memory timing is specified is by its **nS Rating** (nanosecond). Common values range from 60 to 100 nS, and again, numerically lower values refer to faster DRAM.

Upon receiving a Memory Command from the CPU, the chipset enters into an **Address Decode** phase that can last several Wait States. During this period, the chipset translates the *linear* CPU address into a memory *bank-select* and *multiplexed* (Row and Column) address. (DRAM chips operate as a matrix). Afterwards, the **Read** or **Write Command** phase is executed, where the actual memory access occurs. Note that Address Decode timing depends on chipset characteristics and is independent of the DRAM access requirements. However, many chipsets do not allow you to program the Address Decode timing directly, but instead, automatically increase it when greater Read/Write WS are selected.

PAGEMODE and INTERLEAVE

Several memory access *methods* can enhance performance and are independent of the DRAM timing requirements.

Pagemode operation keeps the memory activated after a memory cycle is completed, allowing subsequent accesses in that neighborhood (*page*) to proceed without delay. In a few systems, you can program a **RAS-Timeout** value to control the maximum amount of time the DRAMs are kept activated. The standard timeout is 10 uS (ten microseconds), but even if the timer wasn't present, Refresh cycles (see below) will generally deactivate the memory within a reasonable amount of time.

Word interleave causes accesses to alternate between two or more memory banks in a round-robin fashion. The recovery time needed for the first bank is absorbed by the access time of the second. (And so forth). **Block interleave** is similar, except the interleave occurs in increments of memory *pages*. As is suggested, this method works best when Pagemode is enabled.

MEMORY REFRESH

General purpose memory is called *Dynamic* (as opposed to *Static*) because it requires periodic recharging to **refresh** its content. The CPU cannot access memory during these brief refresh cycles, and system performance is degraded slightly. You may be able to improve system performance by selecting parameters that reduce refresh overhead.

The time interval, or **period**, required between refresh cycles depends on the type of memory being used. Common 256K DRAMs require 15 uS (microseconds), and many 1 Meg DRAMs only need refresh each 30 uS. *Slow Refresh* DRAMs can wait 60 uS or longer. Empirically, most DRAM can be deprived of refresh for much longer periods than their ratings indicate.

The **method** used to refresh the memory can also affect overall performance. **Standard** refresh cycles begin with a process called *Hold arbitration*, and proceed when the CPU releases control of the memory. **Burst** refresh attempts to reduce overhead by performing several refresh cycles during a single Hold sequence. **Hidden** refresh can refer to the way a refresh cycle is formed, or to a technique where the CPU is monitored for the most opportune moment to initiate refresh.

A20-GATE CONTROL

The 8088 CPU in the earliest PC computers was able to access only 1 Mb (MegaByte) of memory. An attempt to address beyond 1 Mb would result in *wraparound* to the lowest memory, and several programs were known to have exploited this property. The next generation 80286 CPU had four additional address pins (A20-A23), allowing access up to 16 Mb. The wraparound effect would now occur at 16 Mb, so something needed to be done to accommodate these early programs. A hardware component called an **AND-Gate** was installed on CPU Address **A20** (thus **A20-Gate**) that could be turned off to induce 1 Mb wraparound, or turned on to permit access to the entire 16 Mb memory space. For economy, a spare pin on the keyboard controller (8042 KBC) was used to control this A20-Gate, under direction of System BIOS or advanced software aware of its existence.

In modern systems, 8042 controlled A20-Gate severely limits throughput to high memory. It is often responsible for erratic keyboard and system behavior seen with early Novell and other Operating Systems. While an *optimized 8042* will help, some chipsets implement a **Transparent-A20** scheme that emulates the 8042 A20 function much more swiftly and reliably. Hardware **Fast-A20** permits forcing A20 enabled, but generally requires that BIOS manually activate it.

REMAP TO EXTENDED-MEMORY

The region between 640K and 1Mb is dedicated to special system resources, including Video buffers, Adapter ROMs, and System BIOS. The 384K of DRAM that would *naturally* appear there is inaccessible and essentially, it is wasted. It may be possible to enable this memory in its natural location as Shadow-RAM. (See Shadow-RAM Setup Utility). As another alternative, some chipsets can **Remap** a portion of this 384K to the very top of memory, increasing the total Extended Memory in the system. Often, the un-Shadowed memory is Remapped *automatically* and you need not be concerned with it. In other designs though, you can control Remap manually from within this utility. To view the final memory configuration, refer to the System and Extended Memory fields in the Setup Summary window.

AT-BUS TIMING

A *bus* is a group of lines (wires) that carry signals on a printed circuit board (PCB).

There are various buses on your motherboard; a Data Bus, an Address Bus, and so on. The industry standard bus that is connected to the 16-bit Adapter Card slots in your computer is called the **AT-Bus**. Its signal definition is the basis of the PC/AT compatible peripheral card industry. A 32-bit, high-performance *Local-Bus* extension to the AT-Bus is discussed in the next section.

The motherboard supplies a **Clock** frequency to the AT-Bus that governs the rate which data is exchanged between them. Although the industry standard AT-Bus clock rate is defined to be approximately 8 MHz (MegaHertz), many newer peripheral cards can be run at considerably higher rates for improved performance.

The width of an AT-Bus cycle can be increased in increments of AT-Bus clocks by adding **Wait States**. While this effectively reduces the bus performance, it is not quite the same as decreasing the Clock speed.

The delay between Input/Output cycles is referred to as **I/O Recovery** time. Many AT-Bus devices (especially IDE drives) require an extra delay between back-to-back cycles when running with an increased AT-Bus clock or in a high speed system.

LOCAL-BUS TIMING

The original 80286-based PC/AT ran at 6-8 MHz, often using the CPU clock to drive the 16-bit AT-Bus *synchronously*. While modern 32-bit systems run at much higher speeds, the AT-Bus remains 16-bit / 8 MHz to allow continued use of peripheral cards that were designed to this specification. Ironically, the *bus-conversion* and *asynchronous* clocking now needed for AT-Bus compliance has restricted advancement of the very peripherals it is intended to support.

To overcome this bottleneck, the PC video industry has introduced a 32-bit extension to the AT-Bus, called **VESA Local-Bus**. It connects directly to the CPU's local signals and clock, theoretically being able to run as fast as 40 MHz without Wait States. To obtain this speed, the system board chipset allows these Adapter Cards the first opportunity to *intercept* CPU cycles. The card claims the cycle by immediately asserting a *local-device* signal called **LDEV**, but the system board may need a Wait State to receive this signal. The Adapter Card asserts a *local-ready* signal called **LRDY** when it has completed the bus cycle. It may need to be re-synchronized with the system board clock (ie, a Wait State) due to signal skew.

(This page intentionally left blank)

Appendix

DIAGNOSTIC PORT 80H POST-CODES

0

0

/

0

0

H

- Cold-Boot commences. (Not seen with warm-boot).0

/

0

1

H

- HOOK 00. OEM specific, typically resets chipset to default.0

/

0

H

- Disable critical I/O: 6845s, 8237s, 765, and parity latches.0

3

/

0

H

- BIOS checksum test.0

4

/

0

4

H

- Page register test. (Ports 81-8F).0

5

/

0

5

H

- 8042 (Keyboard Controller) Selftest.0

6

/

0

H

- Gang Port Init: 8237 m/s, 8254 ch2/1, RTC REG F/A, 8259 m/s.0

7

/

0

7

H

- HOOK 01. OEM specific, typically disables cache, shadow.0

8

/

0

H

- Refresh toggle test (PORTB).0

9

/

0

H

- Pattern test master/slave 8237s, eight 16-bit regs each.1

0

/

0

A

H

- Base 64K memory test.1

/

0

B

H

- Pattern test master/slave 8259 mask regs.1

/

0

C

H

- 8259 / IRQ tests, purge powerup ints.1

3

/

0

D

H

- 8254 channel-0 test and initialization.1

4

/

0

E

H

- 8254 channel-2 toggle test, test speaker circuitry.1

5

/

0

F

H

- RTC tests/inits: Init REG-B, write/readback NVRAM, PIE test.1

9

/

H

- HOOK 02. OEM specific, select 8MHz bus.1

6

/

0

H

- Video Initialization.1

7

/

1

H

- CMOS Checksum test.1

8

/

H

- Signon msg, Accept KB BAT, perform 1st try KB init, cold-boot delay.2

0

/

4

H

- Size/Test base memory (low 64K already done).2

/

5

H

- Perform 2nd try KB init, if necessary.2

/

H

- HOOK 03. OEM specific. Size/Test cache.2

3

/

7

H

- Test A20 gate, off then on.2

4

/

H

- Size/Test extended memory.2

5

/

H

- HOOK 04 and Size/Test system memory ("special" OEM memory).2

6

/

A

H

- Test RTC Update-In-Progress and validate time.2

7

/

B

H

- Serial port determination, off-board/on-board.2

8

/

C

H

- Parallel port determination, off-board/on-board.2

9

/

D

H

- Coprocessor determination/initialization.3

0

/

E

H

- Floppy controller test/determination, cmos validation.3

/

F

H

- Fixed Disk controller test/determination, cmos validation.3

/

0

H

- Rigorous CMOS parameter validation, display other config changes.3

3

/

1

H

- Front-Panel lock check, wait for user to acknowledge errors.3

4

/

H

- Set NumLock, Password-Security Trap, dispatch to Setup-Utility.3

5

/

H

- HOOK 05. OEM specific.3

6

/

4

H

- Set typematic rate.4

0

/

H

- HOOK 06. OEM specific, typically enables shadow, cache, turbo.3

7

/

5

H

- Floppy subsystem initialization.3

8

/

H

- Fixed subsystem initialization.3

9

/

7

H

- ACK errors, set primary adapter video mode.4

/

H

- Disable A20-gate, set low stack, install C800-E000 ROMs.4

/

A

H

- ACK errors, set video mode, set DOS time variables from RTC.4

3

/

B

H

- Enable parity checking and NMI.4

4

/

C

H

- Install E000 ROM.4

5

/

D

H

- ACK errors.4

6

/

E

H

- HOOK 07. OEM specific. Log-in EMS (if built-in).4

7

/

F

H

- Pass control to INT19 (boot disk).

BEEP CODES AND MESSAGES

***Beep Codes** - L=low tone and H=high tone*

Port 80H	Beep Code	Error Message
03/03H	LH-LLL	ROM-BIOS Checksum Failure
04/04H	LH-HLL	DMA Page Register Failure
05/05H	LH-LHL	Keyboard Controller Selftest Failure
08/08H	LH-HHL	Memory Refresh Circuitry Failure
09/09H	LH-LLH	Master (16 bit) DMA Controller Failure
09/09H	LH-HLH	Slave (8 bit) DMA Controller Failure
10/0AH	LH-LLLL	Base 64K Pattern Test Failure
10/0AH	LH-HLLL	Base 64K Parity Circuitry Failure
10/0AH	LH-LHLL	Base 64K Parity Error
10/0AH	LH-HHLL	Base 64K Data Bus Failure
10/0AH	LH-LLHL	Base 64K Address Bus Failure
10/0AH	LH-HLHL	Base 64K Block Access Read Failure
10/0AH	LH-LHHL	Base 64K Block Access Read/Write Failure
11/0BH	LH-HHHL	Master 8259 (Port 21) Failure
11/0BH	LH-LLLH	Slave 8259 (Port A1) Failure
12/0CH	LH-HLLH	Master 8259 (Port 20) Interrupt Address Error
12/0CH	LH-LHLH	Slave 8259 (Port A0) Interrupt Address Error
12/0CH	LH-HHLH	8259 (Port 20/A0) Interrupt Address Error
12/0CH	LH-LLHH	Master 8259 (Port 20) Stuck Interrupt Error
12/0CH	LH-HLHH	Slave 8259 (Port A0) Stuck Interrupt Error
12/0CH	LH-LHHH	System Timer 8254 CH0 / IRQ0 Interrupt Failure
13/0DH	LH-HHHH	8254 Channel 0 (System Timer) Failure
14/0EH	LH-LLLLH	8254 Channel 2 (Speaker) Failure
14/0EH	LH-HLLLH	8254 OUT2 (Speaker Detect) Failure
15/0FH	LH-LHLLH	CMOS RAM Read/Write Test Failure
15/0FH	LH-HHLLH	RTC Periodic Interrupt / IRQ8 Failure
16/10H	LH-LLHLH	Video ROM Checksum Failure at Address XXXX Mono Card Memory Error at Address XXXX Mono Card Memory Address Line Error at Address XXXX

		Color Graphics Card Memory Error at Address XXXX Color Graphics Card Address Line Error at Address XXXX
17/11H	(None)	Real Time Clock (RTC) Battery is Discharged
17/11H	(None)	Battery Backed Memory (CMOS) is Corrupt
18/12H	LH-HLHLH	Keyboard Controller Failure
20/14H 24/18H 25/19H	LH-LHHLH	Memory Parity Error
20/14H 24/18H 25/19H	LH-HHHLH	I/O Channel Error
20/14H 24/18H 25/19H	(None)	RAM Pattern Test Failed at XXXX Parity Circuit Failure in Bank XXXX Data Bus Test Failed: Address XXXX Address Line Test Failed at XXXX Block Access Read Failure at Address XXXX Block Access Read/Write Failure: Address XXXX Banks Decode to Same Location: XXXX and YYYY
18/12H 21/15H	(None)	Keyboard Error - Stuck Key Keyboard Failure or no Keyboard Present
23/17H	LH-LLLHH	A20 Test Failure Due to 8042 Timeout
23/17H	LH-HLLHH	A20 Gate Stuck in Disabled State (A20=0)
23/17H	(None)	A20 Gate Stuck in Asserted State (A20 Follows CPU)
26/1AH	LH-LHLHH	Real Time Clock (RTC) is Not Updating
26/1AH	(None)	Real Time Clock (RTC) Settings are Invalid
30/1EH	(None)	Diskette CMOS Configuration is Invalid Diskette Controller Failure Diskette Drive A: Failure Diskette Drive B: Failure
31/1FH	(None)	Fixed Disk CMOS Configuration is Invalid Fixed Disk C: (80) Failure Fixed Disk D: (81) Failure Please Wait for Fixed Disk to Spin Up
32/20H	(None)	Fixed Disk Configuration Change Diskette Configuration Change Serial Port Configuration Change Parallel Port Configuration Change Video Configuration Change Memory Configuration Change Numeric Coprocessor Configuration Change
33/21H	(None)	System Key is in Locked Position - Turn Key to Unlocked Position
41/29H	(None)	Adapter ROM Checksum Failure at Address XXXX

FIXED DISK PARAMETER TABLE

Type	Size (formatted)	Cylinders	Heads	Precomp Cylinder	Landing Zone	Sectors per Track
1	10.7 Mb	306	4	128	305	17
2	21.4 Mb	615	4	300	615	17
3	32.1 Mb	615	6	300	615	17
4	65.5 Mb	940	8	512	940	17
5	49.1 Mb	940	6	512	940	17
6	21.4 Mb	615	4	None	615	17
7	32.2 Mb	462	8	256	511	17
8	31.9 Mb	733	5	None	733	17
9	117.5 Mb	900	15	None	901	17
10	21.4 Mb	820	3	None	820	17
11	37.2 Mb	855	5	None	855	17
12	52.1 Mb	855	7	None	855	17
13	21.3 Mb	306	8	128	319	17
14	44.7 Mb	733	7	None	733	17
15	0.0 Mb	0	0	None	0	0
16	21.3 Mb	612	4	0	663	17
17	42.5 Mb	977	5	300	977	17
18	59.5 Mb	977	7	None	977	17
19	62.4 Mb	1024	7	512	1023	17
20	31.9 Mb	733	5	300	732	17
21	44.7 Mb	733	7	300	732	17
22	21.9 Mb	733	5	300	733	17
23	10.7 Mb	306	4	0	336	17
24	42.9 Mb	805	4	None	805	26
25	72.5 Mb	925	9	None	925	17
26	104.9 Mb	776	8	None	776	33
27	44.6 Mb	1024	5	512	1024	17
28	71.3 Mb	1024	8	None	1023	17
29	71.6 Mb	823	10	None	823	17
30	159.8 Mb	1224	15	None	1223	17
31	98.0 Mb	1024	11	None	1024	17
32	133.7 Mb	1024	15	None	1024	17

33	44.6 Mb	1024	5	None	1024	17
34	10.7 Mb	612	2	128	612	17
35	80.2 Mb	1024	9	None	1024	17
36	71.3 Mb	1024	8	512	1024	17
37	42.8 Mb	615	8	128	615	17
38	71.6 Mb	823	10	256	823	17
39	42.2 Mb	809	6	128	809	17
40	42.8 Mb	820	6	None	820	17
41	42.5 Mb	977	5	None	977	17
42	42.7 Mb	981	5	None	981	17
43	71.6 Mb	823	10	512	823	17
44	72.2 Mb	830	10	None	830	17
45	119.7 Mb	917	15	None	917	17
46	User programmable drive parameters					
47	User programmable drive parameters					