

CodeHead Utilities Release #3 Notes

Release date: Tuesday, November 28, 1989

ADDED FEATURES IN CODE_RAM 2.0

There are quite a few new features added to the CodeHead RAM disk. If you are trying to implement these features as you read about them and are becoming confused, be aware that there is a tutorial at the end of this section which will help you put them all together. However it is not necessary that you use all of these features or any of them. Don't be afraid to simply run CodeHead RAM disk from the desktop and install or remove it manually. On that level it can be very simple to operate.

Removing the CodeHead RAM Disk:

CODE_RAM is now removable. When it is run from the desktop, the first button in the alert box will read "Remove" if there is currently a RAM disk installed, otherwise it will read "Install". The alert box will also show the current drive letter and size (if one exists). Removing a RAM disk will return all of its memory to the system. After removing a RAM disk, you will end up in the configure screen. If you do not wish to install another RAM disk, you should cancel installation by hitting Esc or Undo.

Automatic Re-installation:

CODE_RAM now has a reset-resident routine which will re-install itself without the need to be re-run. After installing CODE_RAM, a warm reset will re-install your RAM disk, showing the message, "Re-installing CodeHead RAM Disk: X". Of course, the 'X' will be replaced by whatever drive letter you've installed, unless you've actually installed it as drive X.

Drives A through Z:

That's right, you can install your RAM disk as any letter from A-Z! Note that the use of drives Q-Z may not work with all applications, although you can install desktop icons for drives Q-Z, or access them from the system file selector or the Little Green Selector by editing the directory line.

Piggybacking Drives:

You can also piggyback your CodeHead RAM disk on top of an existing drive! If you select a drive that is already being used, an alert box will ask you if you want to piggyback that drive. Access to piggybacked drives can be switched by a hot key. While holding a combination of Control/Left-Shift/Alternate, pressing the up arrow (the initial setting) will provide access to the RAM disk and pressing the down arrow will provide access to the previously existing drive. This means that if you've used up all 16 drive letters (and you don't want to use Q-Z), you can install a RAM disk on top of another drive, for instance, drive B.

Automatic Loading of the RAM Disk:

You can now automatically load the RAM disk with files and folders. This feature can be toggled on or off by the F5 key or from a command line. To use this feature, you must first have saved a CCP file using CodeCopy (also on the Utilities Disk). If CCP loading is turned on, while installing, CODE_RAM will look for a file called CODE_RAM.CCP in the current directory. NOTE: When executing from the AUTO folder, the current directory is the root directory of the boot drive. If CODE_RAM.CCP is found, it will be loaded into the RAM disk, creating all files and folders that were on the disk used to create the CCP file. If the CCP file was originally saved from a different sized disk, the boot sector will be retained so that no memory will be lost. If the CCP file is too large to fit in the RAM disk, a message will inform you and the CCP file will not be loaded. Switching CCP loading off will also force auto-shrink off (see below).

Automatic Shrinking:

The F9 key will toggle between 'Auto-shrink' and 'No shrink'. When set to shrink, the RAM disk size will be determined by CODE_RAM.CCP (if it exists) instead of the size selected. This allows you to conserve your memory by installing a RAM disk that is just large enough to contain the files you want loaded. When auto-shrink is selected, it will show as the selected RAM disk size (forcing the F5 CCP switch on if need be). Sizes will not be selectable by mouse or arrows. If you are saving your configuration with auto-shrink, the size of the RAM disk will be saved in the flexible 'X' size (see below).

Booting from the RAM Disk:

During its initialization, CODE_RAM will check to see if there is an AUTO folder present on the RAM disk with executable files in it (even if it has just loaded them through a CCP file). If so, it will execute all the AUTO programs there, providing a lightning fast bootup (make sure you also use PinHead). After the AUTO programs have executed from the RAM disk, the remaining programs in your original bootup AUTO folder will finish executing. When warm booting with CODE_RAM already installed, the system will execute the AUTO programs on the CodeHead RAM disk only, providing an even faster bootup (because it doesn't need to load CODE_RAM.CCP again).

Non-auto-booting Hard Disks:

If CODE_RAM.CCP is not found in the current directory, CODE_RAM will try to find it in the root of drive C (if drive C: exists). This means that even if your hard disk doesn't autoboot, you can still have a fast bootup. To set up this way, your floppy boot disk should have an AUTO folder containing your hard disk driver program, followed by CODE_RAM. After CODE_RAM runs, CODE_RAM.CCP will be loaded from drive C, and the AUTO programs will run from the RAM disk (with one last look at drive A to check for more AUTO programs). Make sure to include the hard disk driver in your RAM disk AUTO folder for later warm boots. (You're welcome, Astra owners.)

Loading Accessories from the RAM Disk:

You can also load your accessories from the RAM disk (these guys think of everything). Accessory loading can be switched between the RAM disk and the normal location by using the F8 key, or from the command line. If you are loading ACCs from the RAM disk and you change resolutions on a color monitor, the accessories will not be reloaded.

Blitter Usage for More Speed Yet:

RAM disk reads and writes will now take advantage of the blitter if available and you have selected it. Blitter usage can be toggled on and off with the F6 key or from a command line. Speed advantage varies between 5%-20%. When the blitter is used, extreme movement of the mouse during reads and writes sometimes causes a key click or two with no ill effects.

Floppy Sizes for the RAM Disk:

You may also install one of four different preset floppy disk sizes. These are shown on the configure screen and correspond to F1-F4. (F10 is now used to toggle between save/no save).

Flexible Size:

There is another new size on the configure screen, corresponding to the letter X. This is a flexible size (denoted by '~') that is initially set to 200K. You may change this size by hitting F7 and typing in a new size. It may also be changed by using a command line which sets a size and saves the configuration. This allows you to use any size RAM disk, not just the preset 64K increments set up in CODE_RAM.

Saving your Configuration:

The F10 key will enable or disable saving of your settings. (This was formerly done with the F1 key). Each time the RAM disk program is run, saving is turned off. You must turn it on to save your configuration. Items which are saved are: RAM disk size, drive letter, installation state (whether or not to install), CCP loading, blitter state, flexible size, ACCs load location, and auto-shrink setting. Your program must be named CODE_RAM.PRG in order for the settings to be properly saved. Saving can also be enabled from a command line.

Summary of Function Keys:

- F1 - Install an image of a single sided, 9 sector per track floppy.
- F2 - Install an image of a single sided, 10 sector per track floppy.
- F3 - Install an image of a double sided, 9 sector per track floppy.
- F4 - Install an image of a double sided, 10 sector per track floppy.
- F5 - Turn CCP file loading on or off.
- F6 - Turn blitter usage on or off.
- F7 - Edit the flexible X size.
- F8 - Switch between loading ACCs from RAM disk and normal location.
- F9 - Switch between auto-shrinking and selected size.
- F10 - Save configuration or not.

Cold Booting:

When you run CodeHead RAM Disk from the desktop, you may select 'Cold Boot' from the alert box, after which you will get one more warning before doing a cold boot. On a warm reset, if you hold Control/Left-Shift/Alternate (after your hard drive installs), you will be asked if you want to do a cold boot. You'll find there are times when you need to do this as CODE_RAM sticks like glue. A cold boot is also available from a command line.

Delay Setting:

If you use SETDELAY.PRG to save a delay of 0 into CODE_RAM, the configure screen will not even flash on the screen (from the AUTO folder), but rather just show the installation (or non-installation) message. You may, however, still interrupt the process by pressing any key before CODE_RAM runs.

Command Line Control:

CODE_RAM will now accept a command line. There are a number of ways to pass a command line to CODE_RAM. You can install CODE_RAM as a TTP (TOS Takes Parameters) file. Then when you run it, a box will come up allowing you to type in the command line you want to use. Better yet, if you have CodeHead's HotWire (1.2 or later), you can install CODE_RAM in the HotWire menu as many times as you like, each with a different command line. Then you can just click on the item you like and CODE_RAM will do its thing. Command lines may also be implemented on other ST shells. See the manual for your shell to find out if it passes command lines.

Commands must begin with a dash (minus), "-". Commands may be in upper or lower case and may optionally contain spaces (except immediately after the "-"). All commands will override the current settings saved in CODE_RAM, but if an option is not specified, the saved setting will be used. A description of the commands is as follows:

- N No warning will be given before removing the current RAM disk or doing a cold boot.
- D<A-Z> Drive identifier (-DP = use drive P)
- I<size> Install a RAM disk using the current drive letter. Size may be specified in K (100 = ~100,000 bytes), F1-F4 for the preset floppy sizes, or 's' for auto-shrink in which case the size will be determined by CODE_RAM.CCP, if found. If there is no size specified, then the last size saved will be used. If a RAM disk is currently installed, it will first be removed (with or without warning).
- R Remove the current RAM disk (if one exists). An alert box will confirm your actions unless warnings have been disabled.
- W Write this configuration into the program (CODE_RAM.PRG)
- C Cold boot (with or without warning).
- L<0,1> Load CCP file (-L1) or don't load CCP file (-L0)
- A<R,N> Load ACCs from RAM disk (-AR) or load ACCs normally (-AN)
- B<0,1> Blitter usage off (-B0) or on (-B1)
- Q Quit - when removing, this will exit the program rather than leaving you in the configure screen.

Here are some sample command lines. Note that spaces are allowed (except after the '-') and upper or lower case letters may be used.

- N-R The current RAM disk will be removed without warning and you will end up in the configure screen waiting for input.
- C You will be asked if you're sure you want to do a cold boot.
- dB -i302 A 302K RAM disk will be piggybacked onto drive B. If there was a RAM disk already present, you will be warned.
- W-dq-R-i A RAM disk will be installed as drive Q with the last size that was saved and this configuration will be saved. The '-R' is not needed and you will be warned if another RAM disk was present.
- n -r -q~ The current RAM disk will be removed without warning and the program will exit immediately. The '~' on the end will tell HotWire not to stop for your input. This command line is perfect for use as HotWire's auto-run program if memory is precious.
- L1-IS-AR-B1 Load CCP file, install with auto-shrink, ACCs load from RAM disk, and use the blitter.
- i400-l0-an-b0 Install 400K RAM disk, ignore CCP file, ACCs load from normal location, and don't use the blitter.

Miscellaneous:

There is special programming in CODE_RAM to work with FATSPEED. If you use FATSPEED.PRG, make sure you do not rename it or it will not install correctly.

You can now exit from the final prompt of CODE_RAM (when run from the desktop) with a keypress OR the mouse button.

A SIMPLE (?) TUTORIAL FOR PUTTING THIS ALL TOGETHER

No doubt reading about all these features boggles the mind a bit. The following description assumes that you are using HotWire and describes my own setup:

1. Boot up in your normal way, install CODE_RAM.PRG in the HotWire menu, and run it. Click on 'Install' to bring up the configure screen and choose a drive letter by pressing <Control>A-Z. I use 'P'. Hit F5 to enable CCP loading, F6 is you have a blitter, F8 to load ACCs from the RAM disk, and F10 to save the settings. Then select a size large enough to contain your AUTO programs and accessories (mine take 228K) by moving the cursor with the mouse or arrow keys and pressing return or clicking the left mouse button, or just press the key corresponding to the size you desire. The default size of 383K is a good starting point. Don't worry, we'll be changing the size later.

2. Create an AUTO folder on the RAM disk and copy the AUTO programs you want to use in the order that you want them to run. MaxiFile is a good way to do this, using the 'No Sort' option. My AUTO programs are (in running order)--FatSpeed, PinHead 1.4, Desk Manager 2.7, Cartridge RAM Disk Driver, Tempelmon Debugger, Macro Utilities, G+Plus, Little Green Selector, Folder300.PRG, CodeHead Spooler, and HotWire. If you use Desk Manager 2.7, you should configure it (by pressing Help as you run it) to look for its AUTO programs on the drive you've chosen. It's also best to set the default to bypass so that it will only come up if you press a key first. In fact, this feature was added specially for this kind of speed-booting. You may want to create a DESKMGR folder inside the AUTO folder and copy your HIGHDESK.INF, MEDDESK.INF, and LOWDESK.INF files into it, depending on what monitor(s) you use. Then copy an ASSIGN.SYS file (if you use G+Plus/GDOS) and your PINHEAD.DAT file to the root directory of the RAM disk and any accessories you wish to load. I use only MultiDesk and load all other ACCs into it. That's all I use. You can use more, or less, as you desire.

3. Next install CodeCopy in the HotWire menu and select 'Command Line' in the 'Options' box. Enter a descriptive title; mine says, "Save Boot Drive P". Click on OK and run it. When the command line box comes up, enter the following: "-B0 -DP -R -SC:\CODE_RAM.CCP-Q~". This will turn off the bell, select drive P (use the drive you wish), read the disk, save it as C:\CODE_RAM.CCP, and quit. The tilde (~) is to keep the command line from coming up every time you run it. When you click on OK, CODECOPY will go into action, saving your RAM disk before you know it. If you need to later edit the command line, hold Control while selecting it. Remember to save your HOT file for later use.

4. Now you should copy CODE_RAM.CCP into the same directory as CODE_RAM.PRG. Install CODE_RAM into HotWire with the following command line: "-W -N -DP -IS~". This will re-install your RAM disk as drive P, auto-shrinking it to the correct size, load your CCP file and save the configuration into itself.

5. Now run SETDELAY.PRG, locate your working copy of CODE_RAM.PRG and save a delay of 0.

6. Copy your customized CODE_RAM.PRG into your C:\AUTO folder and rename all the other PRGs to PRX. MaxiFile works great for this, using shift to select all the PRGs. You might want to also use AUTO_ORG.PRG to make CODE_RAM.PRG the first program in your AUTO folder (for optimum speed).

7. At this point (whew) you should be ready to try it out. Use the HotWire cold boot combination to check it out. If all has gone well, you'll see your hard drive installation message, followed by CODE_RAM's installation message, telling you it's shrinking to fit and loading CODE_RAM.CCP (which only takes about 2 seconds). Look out now, because your AUTO programs are going to run so fast you won't be able to read their installation messages. In fact, you'll want to use NO_CLEAR.PRG to remove the screen clearing from your AUTO programs, producing a smooth scroll that you can almost read.

If you can't afford to keep CODE_RAM in memory all the time, you can install CODE_RAM.PRG as HotWire's auto-run program, with the command line "-n-r-q~". This will remove the RAM disk with no warning and quit, returning the memory it used to the system.

Because I own a hardware, cartridge RAM disk, I use the CodeHead RAM disk only for bootup purposes. There's no reason that you can't make CODE_RAM large enough for both bootup purposes and your normal work purposes, be they word processing, programming, or whatever. For this type of use, you'll want to disable auto-shrink.

While testing the features described here and doing some timings with the latest ICD hard disk caching software (4.2.1), I've come to realize that their new software is almost as fast as booting off of the CodeHead RAM Disk. If you are using this ICD software, you may decide that it is fast enough that you don't want to bother with all the setup involved with CODE_RAM. Bear in mind that you can still use many of the other features available. Don't forget that you can set up various CODECOPY command lines to load in different RAM disk setups simply by clicking on their HotWire entry.

ADDED FEATURES IN CODECOPY 1.2

NOTE: In case you didn't have CODECOPY 1.1, see below for those features.

A bell pitch setting of 0 will eliminate the bell sound, also eliminating the usual pause before proceeding.

You can now abort a command line operation by holding the Alternate key.

There is no minimum memory size required (previously it required enough for a single-sided disk). You will be told when you read a disk or load a CCP file if there is not enough memory. This allows use of CODECOPY with smaller memory sizes as long as the disk to be copied doesn't have too many files on it.

Drives A-Z are now allowed. This allows you to load and save CODE_RAM disks beyond P.

Two new command line options have been added:

- f Format disk (original omitted due to danger of overwriting, now included to allow use of command lines for copying floppies). You will be warned before formatting unless you use the '-n' option.
- n No warnings will be given before formatting (command line only) or writing to a disk with a different size boot sector. This command must precede the Format command (-f) to cancel the warning.

**** IMPORTANT WARNING **: BE EXTREMELY CAREFUL WHEN DISABLING WARNINGS IN THE COMMAND LINE. IT IS POSSIBLE TO OVERWRITE AN ENTIRE HARD DISK PARTITION IN A SPLIT SECOND IF YOU USE THE WRONG DRIVE LETTER!!!!**

A bug was fixed that previously allowed you to load a CCP file that was too big for the current memory size.

Omitted from the Release 2 README file was the fact that command lines are executed in the order entered, as if you had executed them manually. For instance, a Read operation (-r) must precede a Write operation (-w). Also you can abort from a format operation by hitting Esc.

CAPSLEFT.ACC and CAPSRITE.ACC

Many people have requested that CapsLock be able to save its setting so that it could come up on the left side of the screen. Since CapsLock is quite small, we decided to supply two versions, CapsLeft and CapsRite, each defaulting to its respective side of the screen. Now you can choose which one you want to use. They can both still be swapped to either side of the screen.

AUTO ORGANIZER AND ZERODISK NOW FORCE A MEDIA CHANGE

Auto Organizer and ZeroDisk will now force a media change after doing their respective tasks. We had a couple of reports of problems with these programs. Although we couldn't duplicate these problems, we've added the media force routine in hopes that it would put an end to these reports. If you have any problems with either Auto Organizer or ZeroDisk, please let us know.

NO_CLEAR.PRG

NoClear will allow you to change your AUTO programs so that they will not clear the screen, but rather produce a smooth scrolling effect during bootup. We've included this FreeWare program for your convenience. See the file NO_CLEAR.TXT to learn of its use.

CodeHead Utilities Release #2 Notes

Release date: August 28, 1989

NEW FEATURES IN MULTIFILE 1.2

Communication with the Little Green Selector

Included on your CodeHead Utilities disk (in the SHARWARE folder) is a replacement for the GEM item selector, called the Little Green Selector. This program (known from here on as LGSELECT) is shareware by Charles F. Johnson, and version 1.3 of LGSELECT now incorporates a way to directly call MultiFile 1.2, without leaving the file selector. This gives you full access to all MultiFile capabilities any time you use the file selector! In order to make this work, MultiFile must be installed as a desk accessory, either in MultiDesk or as an ordinary accessory. See the documentation for LGSELECT (included in a text file on your CodeHead Utilities disk) for more details.

Save Configuration

The 'Save Config' button has moved to the main MultiFile box. (It used to be in the 'More' box.)

Force Media Change

You can now set 'media changed' status on any drive by holding the Alternate key when you select a new drive for the 'Source Path'. This is very handy for people with removable media drives, like the Supra FD-10 (a Konica mechanism) or the Syquest removable hard drives, since these types of drives sometimes don't sense it when you change their disk (or cartridge). Some STs also have built-in floppy drives that don't always recognize a disk change; the 'Force Media Change' feature of MultiFile will solve this problem. (NOTE: MultiFile always forces a media change when you type the Esc key to reread a directory.)

The 'More' box

Three new options have been added to the MultiFile 1.2 'More' box.

1) 'Show Hidden Files' - When this button is selected (highlighted), MultiFile 1.2 will display files which are marked as 'hidden' or 'system' files. These types of files are normally invisible to the ST desktop and most programs that read a disk directory. 'Hidden' or 'system' files will appear in MultiFile's box as 'half-tone' or 'disabled' text, so you can easily distinguish them from normal files. (NOTE: you can change a file's attributes, including its 'hidden' status, from the 'File Information' box. See the 'MultiFile' section in your CodeHead Utilities manual for more details about the 'File Information' box.)

When the 'Show Hidden Files' button is not selected, 'hidden' and 'system' files will not appear in MultiFile's directory.

2) 'Exit After Save' - When this button is selected, MultiFile 1.2 will exit immediately after you save your configuration with the 'Save' button in the main MultiFile dialog box. Depending on the setting of the 'Exit After Save' warning (see below), you will either see a prompt or exit without any further ado.

3) 'Save Current Paths' - When this button is selected, MultiFile 1.2 will save the current source and destination paths (shown at the top of the MultiFile dialog box) when you click on the 'Save' button.

New Options in the File Information Box

The box which appears when you right click on a file now contains six new buttons.

1) 'Normal' - If a file is marked as a 'hidden' or 'system' file, clicking this button will cause it be un-hidden, and restore it to 'normal' status so that it will be visible from the GEM desktop.

2) 'Hidden' - Click this button to 'hide' files so that they are invisible to the GEM desktop. Be careful if you start hiding lots of files; remember where you put them, since you won't be able to see them from the GEM desktop. (This feature can be very handy if you're worried about the security of certain files.)

3) 'System' - See the description of 'Hidden' above.

4) 'Show' - Click on this button to show the selected file on screen. This function operates much like the GEM desktop's 'Show' feature. You can scroll forward through a text file by typing Return (to scroll one line at a time) or the space bar (to scroll one page at a time). You can exit the 'Show' function at any point by typing Q, Control-C, or Undo.

Unlike the GEM desktop, MultiFile's 'Show' function also lets you use the mouse buttons to control scrolling through a file. If you press the left mouse button, the display will scroll continuously, one line at a time, until you let go of the button. If you press the right mouse button, the display will also scroll one line at a time, but not until you let go of the button. The left button should be used when you want to move quickly through a file, and the right button when you need fine control. You can exit the 'Show' function at any point by pressing both mouse buttons together.

5) 'Print' - When you click on this button, MultiFile will send the selected file to your printer. Type any key to stop printing and return to the file information box.

6) 'FastLoad' - This feature is intended to be used with Atari's new TOS 1.4. TOS 1.4 has a special feature that allows some programs to load faster; to use this feature, you need to set a special value in one of the bytes of a file's "header". When you click on 'FastLoad' MultiFile reads the header of the selected file, displays the status of that file's 'fastload bit', and lets you turn the fastload bit on or off. MultiFile will not let you set the fastload bit in a file that is not an executable program. Setting the fastload bit will have no effect if your machine does not contain the TOS 1.4 ROM chips.

(NOTE: Not all programs will work correctly when they are run with their fastload bits set. See the TOS 1.4 documentation for more details.)

New Warning Options

There have been three new options added to MultiFile 1.2's 'Warnings' list.

1) 'Wild Card Delete' (Type W) - Normally, even if you turn the 'Erase Files' warning off, MultiFile will still warn you once once before erasing a group of files (a "wildcard delete"). Turn off the 'Wild Card Delete' warning if you don't even want this "last chance" prompt to appear. However, if you turn this warning off, BE CAREFUL! You will get no warning at all when you click the 'Erase' button - MultiFile will just start immediately erasing all selected files. Unless you're very sure of what you're doing, we suggest that you leave this warning turned on.

2) 'Save Config' (Type G) - This option determines whether or not you will be warned before saving a configuration.

3) 'Exit After Save' (Type X) - If you choose to exit MultiFile after saving a configuration (see 'Exit After Save' in 'The More Box' above), this option determines whether or not you will be warned before exiting.

ADDED FEATURES IN CODECOPY 1.1

CodeCopy will now write a unique serial number to each disk copy. This eliminates the possibility that the system will get confused when swapping disks that were created by CodeCopy. The checksum no longer includes the serial number so you will get a different checksum than CodeCopy 1.0 would give you. Checksumming has also been sped up.

You will now be warned if CodeCopy finds an executable boot sector. Of course, many times (such as an autobootable drive C) it is normal to have an executable boot sector, but you might want to examine the possibility of a virus if you get this message when reading a floppy:

This disk has an executable boot sector! Check it for a virus if you're unsure about it.

CodeCopy now comes up with the main menu prompt rather than requiring you to select bell pitch and drive. A default bell pitch of 4 is used and drive A is selected.

You may now select any drive from A-P. This is intended for use with RAM disks although you *may* have a hard disk partition that is small enough to read into your computer. Now you can quickly read a RAM disk and save it (on your hard disk) as a file. If you want to insure that your file will be as small as possible you should use a disk optimizer (such as TuneUp or Turbo Kit) to move all files to the bottom of the disk (optimize for reading).

WARNING: To help avoid accidentally overwriting a hard drive partition, CodeCopy will only allow "Writing" to a drive with an identical boot sector to that which is being written (unless it is drive A or B). Nevertheless, be **EXTREMELY CAREFUL** when using the "Write" command as it has great potential for losing data in the blink of an eye.

CodeCopy 1.1 will now execute a command line. All keyboard commands except 'Format' can be used in the command line. This means you can tell CodeCopy in advance what you'd like it to do. There are a few different ways to pass a command line to CodeCopy. You can install CodeCopy as a TTP (TOS Takes Parameters) file, or rename it CODECOPY.TTP. Then when you run it, a box will come up allowing you to type in the command line you want to use. Better yet, if you have CodeHead's HotWire (1.2 or later), you can install CodeCopy in the HotWire menu as many times as you like, each with a different command line. Then you can just click on the item you like and CodeCopy will do its thing. Command lines may also be implemented on other ST shells. See the manual for your shell to find out if it passes command lines.

Commands must begin with a dash (minus), "-". Commands may be in upper or lower case and command lines may optionally contain spaces. A description of the commands is as follows:

- b<x> Set the bell to pitch x (x=0 to 9)
- d<x> Set drive to drive x (x=A to P, or 2)
- r Read the current disk
- w Write to current disk (must read or load something first)
- v Verify the current disk (must read or load something first)
- l<file> Load the specified file name
- s<file> Save the specified file (must read or load something first)
- q Quit the program

The Bell and Drive commands require a single data character following the command and the Load and Save commands require a filename (don't include the "<" or ">"). The filename may be a full path (including the drive and directory, i.e. C:\RAMS\G_SAVE.CCP) or just a filename, in which case the current directory will be searched.

If you are still totally confused by all of this (I know I am), here are some sample command lines:

- dG -sD:\RAMFILES\DRIVE_G.CCP Reads drive G and saves it as a file.
- da-l utils.ccp -w Loads "UTIL.CCP" from current directory and writes it to disk in drive A. (Disk must already be formatted).
- B3 -r Sets bell pitch 3 and reads disk in drive A.
- L C:\SAVES\DRV_H.CCP-DH-W-Q Loads a file, writes it to drive H, and quits.

Notice that upper and lower case may be mixed and use of spaces is quite flexible (no spaces are allowed between "-" and the command letter however).

The use of command lines can give you some incredibly powerful possibilities. If you have HotWire, you can set up a screenful of entries which could do things like save RAM disks to your hard disk, load RAM disks from your hard disk, start up CodeCopy with a master disk file loaded and ready to make floppies. Another possibility is to have HotWire auto-run a CodeCopy with commands to load up your RAM disk. Be careful though because a warm reset could automatically write over your CodeHead RAM disk, or another reset-proof RAM disk.

SETDELAY PROGRAM

The SETDELAY.PRG program can be used to alter the automatic timeout delay in CodeHead RAM Disk. The delay time is normally set to about 5 seconds but may be altered anywhere from 0 to 20 seconds. Any decimal value is acceptable, i.e. 2.34 seconds.

CAPSLock NOW SWAPPABLE

You can now choose to display the CapsLock status in either the upper left corner or the upper right corner of the screen. Click on "Swap" to move the indicator to the opposite corner.

LITTLE GREEN SELECTOR REPLACES START SELECTOR

The Start Selector (STSELECT.PRG) is no longer included (as stated on the manual's back cover). It has been replaced by the far more powerful Little Green Selector (LGSELECT.PRG) which can be found in the SHAREWARE folder, along with PinHead, ArcShell, and Desk Manager. Note that these programs are shareware and you are expected to send a contribution if you use them regularly. The documentation for these programs is included in the same folder.

CODEHEAD SOFTWARE UPDATE POLICY

At CodeHead, we update our software frequently; but due to postal expenses we are not able to notify all of our registered users on a regular basis. You may find out the latest version number of any CodeHead Software product by calling us at (213) 386-5735, or by finding us on GEnie, CompuServe, or Delphi.

The latest update for any product may be obtained by returning your original master disk plus \$5.00 to:

CodeHead Software
P.O. Box 74090
Los Angeles, CA 90004

From time to time we offer free updates for certain versions (if we've released something with bugs, etc.). You will be informed (or your money returned) if this is the case.

Please, remember that software theft hurts EVERYONE. If you have a legitimate copy of our software, please accept our sincere thanks for purchasing our product.

If you're reading this document with an illegal copy of our software (you know if you are), you might want to give some thought to the consequences of your actions. We will not continue to produce software for the ST if we can't make a living at it. If you steal a copy of any of our programs, besides breaking a federal law, you're also quite literally stealing the food right out of our families' mouths. Think about it.