CARTDISK PLUS

A Soft Unlimited Product

This version of CARTDISK PLUS is a preliminary version that is only available to DISKEDIT owners. It includes an upgrade for DISKEDIT 2.0, so make sure that your current version of DISKEDIT says 2.0 or greater before purchasing this product (you are safe if your version of DISKEDIT performs an RPM test when it boots up.)

This documentation and the CARTDISK PLUS program are protected by U.S. copyright laws, and may not be copied for any other use than personal backup protection. Copying this software for redistribution or trade is a criminal offense and punishable by fines and imprisonment. Likewise, use of this software for software piracy is also illegal. Soft Unlimited developed this tool for personal backup protection only, and may not be held responsible for illicit use of the program.

This product is sold as-is. Soft Unlimited will, however, replace defective disks at no charge, if returned within 10 days of purchase. Disks returned after 10 days must be accompanied by $5.00 to cover handling and postage for replacement. So, don't forget to make a backup of the program as soon as you get your disk. A program called PCOPY is included on the CARTDISK PLUS disk to allow you to backup CARTDISK PLUS. Run PCOPY the same way you did with Diskedit (boot up the CARTDISK PLUS disk with BASIC inserted, wait until DISKEDIT C.1 starts running, then press the BREAK key and then type RUN "D:PCOPY"). PCOPY will prompt you for the source disk, so just press return. When you see the FORMAT? prompt, place a formatted disk in the drive and press return again. You now have a duplicate of CARTDISK PLUS. Please don't start passing copies of CARTDISK PLUS around. I am selling my software for what I consider a reasonable price, because I am gambling that the lower price will lower the tendency to trade. No single SOFT UNLIMITED program will ever have a suggested retail price of more than $25.00, unless piracy destroys our profit margin.

DISCLAIMER: Just to be safe, I must caution you that Soft Unlimited can not be held liable for any damages you may sustain to your computer and/or cartridges while using CARTDISK PLUS or the procedures outlined herein. You also may void your warranty on your machine. I personally have never had any problems, but you never know.

```
*****************************************
* DISKEDIT and ULTRACOPY may be ordered *
* FROM:                                 *
*   SOFT UNLIMITED                      *
*   3546 PILGRIM LANE                   *
*   PLYMOUTH, MN 55441                  *
*   DISKEDIT-$24.95 Suggested Retail    *
*   ULTRACOPY-$20.00 Suggested Retail   *
*   CARTDISK-$24.95 Suggested Retail    *
*****************************************
```

CARTDISK PLUS
BY
TODD BURKEY

(Preliminary Documentation)

CARTDISK PLUS is a fully assembly language cartridge backup program that will copy both 8K and 16K ROM cartridges. CARTDISK PLUS is full of firsts for the ATARI. You purists out there will appreciate the fact that CARTDISK PLUS is fully backupable and that it in fact requires that the first thing you do is make a backup. That is first #1-this is the only program (so far) that you can buy where the original will refuse to run, but all of the backups will. Those of you familiar with being only able to back up 8K cartridges by inserting them in the right slot, will appreciate that CARTDISK PLUS lets you use the left slot (the same as you do when you normally use the cartridge.) The two best features of CARTDISK PLUS, however, are: 1) It is quick load (important for cartridges that freeze the system a lot), and 2) it automatically searches out most of the possible protection schemes that are used on cartridges (finding the places where the cartridge may try to modify itself). Diskedit (a special version) is used to let you decide in one short session which parts of the disk code are actually protection and which are valid commands. The software has been set up so that no assembly language experience is required to back up most cartridges and minimal experience is required for others. Soft Unlimited can't guarantee that this program will back up all cartridges, but CARTDISK PLUS was recently tested on 17 different products and all 17 could be copied to disk. 9 of the 17 worked the first time with no fixes required, and 5 of the remaining 8 worked after the auto-scan session with DISKEDIT was performed. The final 3 cartridges required extensive assembly experience to back up.

****
* General Procedure *
****

! Step 1-Gettin Set Up !

Actually, the first step in backing up a cartridge is to back up CARTDISK+ using the procedure outlined on the front page of this manual to get a working copy of CARTDISK+. CARTDISK PLUS was designed to allow you to perform all of the backup tasks without ever switching disks in the drive; hence the need to use backups. Next, find a wood pencil or something to insert into the power shut-off slot on your ATARI (located on the front right side under the cartridge door.) The only way to get the data from a cartridge to disk (apart from rewiring your ATARI) is to push the cartridge into the correct slot while the computer is running and a program is booted that can copy from memory to disk. The reason for this is that if you boot up your computer with a cartridge installed (other than BASIC and a few others,) the cartridge will take over and not allow the copy program to boot up. What this means is that you have to have

the shut-off switch de-activated and your cartridge door open while you boot up with CARTDISK PLUS. So wedge something into the shut-off slot, insert the CARTDISK PLUS backup diskette in the drive, and power on the system. The system will quickly boot up and CARTDISK PLUS will be ready to go.

Step 2-Copying the Cartridge

Now for the hard part. It took me 6 months to get up enough nerve to push a cartridge into my computer while the power was on, but now it seems almost natural to have the computer on with the door open. The hardest part in inserting the cartridge is to get it in without glitching the system. If you put the cartridge in too slowly or at an angle, it is very likely that the system will hang. The best method of inserting a cartridge is to position the cartridge right above the contacts and then press it quickly and firmly in position. If the system does hang (the flashing dots on the screen will freeze), then turn off the power, remove the cartridge, and reboot the disk. If the system doesn't hang, CARTDISK PLUS will let you know when it sees the cartridge in memory and then you can just hit a key to let it start copying out to disk. As it is copying, you will see where in memory the program saw the cartridge in (this will tell you what size the cartridge is-8K or 16K) and the program will inform you when it is searching for absolute and indirect stores into the ROM addresses (more about this later). When the program is done copying, it will automatically do a reset and start running the cartridge you inserted in the slot.

Step 3-Cross Your Fingers

Turn off the power, remove the cartridge, and reboot the CARTDISK PLUS disk. Press the T key to load and run the cartridge data that was saved to disk. If there were no protection mechanisms, the software will boot up as if you had inserted the cartridge. If there appears to be garbage on the screen or the system hangs, then follow the steps outlined in the next section. Otherwise, skip to the section on making DOS files.

Step 4-Fixing The Code

This section is the only reason that DISKEDIT owners are the first people that can buy CARTDISK PLUS. The CARTDISK PLUS system is actually composed of several components that all work together in an integrated fashion. As seen in the previous steps, the autoboot portion of CARTDISK PLUS will automatically store the cartridge data out to the disk in a reserved area. It will also store the addresses within this code that are potential candidates for being the locations of cartridge protection schemes. These addresses are stored to another reserved section on the disk. An overview of how the CARTDISK PLUS diskette is structured is shown in appendix A. Once the backup program is done backing up the cartridge to disk, it is time

for the modified DISKEDIT to take over.

CDISKEDIT (for Cartridge DISKEDIT) is basically DISKEDIT with a special feature added that lets it automatically scan the CARTDISK PLUS disk for the cartridge data and the possible protection locations. CDISKEDIT boots up and acts just like DISKEDIT, with the exception that the Z key will invoke an auto-read of the cartridge data and then prompt you for what type of protection you wish to look at first.

Two types of protection are currently searched out (absolutes and indirect stores), but before I describe these I had better describe how a cartridge can be protected. A cartridge will usually protect its code from being executed in RAM by attempting to store data on top of itself. When the ROM is inserted, stores to ROM memory locations are essentially null operations (the ROM memory can't be changed. If RAM is inserted, however, and you have loaded the code from disk and it tries to write over itself, it will be successful and likely crash. For those of you unfamiliar with assembly, you can get the same effect by using the poke command. You can't hang BASIC by poking into memory locations that the BASIC cartridge resides in, but you can definitely screw things up by poking to RAM; hence the need to find the spots in cartridges that may be intentional writes to what is normally a ROM address. There are two ways in assembly language to store out to memory. Using absolute stores is similar to using a POKE command in BASIC-you just POKE an address with a number. Using indirect stores is more like doing a PEEK of several memory locations to get the address that you are going to POKE to-all in one command in assembly. Luckily, the 6502 only has a few (10) commands that fit the category of possible stores that can change ROM memory, so it wasn't hard for me to write software that would extract the locations of these commands from the cartridge that is being copied. Unluckily, 6502 code is such that it is sometimes impossible to tell whether a particular byte is actually a command or part of another command. This is the reason DISKEDIT was modified. The user will have a measure of control over determining what is actually code and what is not. The user interacts with diskedit via using the Z command. DISKEDIT will prompt you as to whether you want to look at the possible absolute stores or indirects. After you type either an A or an I, DISKEDIT will load the game into memory, and then load the store locations that were found during copying. Once everything is loaded, DISKEDIT will automatically disassemble the code around the first address in its list of locations, and prompt you as to whether you want to have DISKEDIT automatically nullify the code (NOP), delete it from the list of possibilities (if it was not really a valid store), save the location for looking at later, or changing the range of disassembly (+/- a user defined number of bytes.) By pressing the A, S, D, or C keys respectively, you can select any of these options. You may also press Q quit the session and start up again later. When all of the absolutes have been scanned (or you press Q), DISKEDIT will save out all of the changes you instructed it to make. If you think you have found all of the checks, you can retry rebooting (without a cartridge) and find out for sure. For those of you unfamiliar with assembly code, I have included two examples below. The one on the left contains an actual store into a memory address. You would tell DISKEDIT to auto-NOP this instruction. The example on the right contains the same command (8D 20 B0 = STA $B020), but it is just coincidence and the bytes are actually part of

two other commands.

```
8090: A9 22    LDA #$22      9233: A8       TAY
8092: 85 E1    STA $E1     9234: A9 8D    LDA $8D
8094: A9 01    LDA #$01    9236: 20 B0 A0 JSR $A0B0
8096: 8D 20 B0 STA $B020   9238: A9 02    LDA #$02
8099: AD 20 B0 LDA $B020   923A: 85 E2    STA $E2
```

For your reference, the 6502 codes that are possible absolute stores are 8D, 9D, 99, 8C, CE, DE, EE, FE, 0E, 1E, 2E, 3E, 4E, 5E, 6E, 7E, and 8E. The indirect stores are 81 and 91. Also remember that the ROM addresses will be between A000-BFFF for 8K cartridges and 8000-BFFF for 16K cartridges.

Step 5-Making DOS Files

This step should not be attempted until you have a working boot version. To make a DOS file out of the boot version, just boot up on the work disk with Diskedit, hit break, and then RUN"D:CRTBLOAD". This program will automatically load your game into memory and ask you for the file name that you want your DOS file to be called (don't forget to add the D:). Once you have given the name, it will automatically write the DOS file out to disk. With luck, you will be able to boot up on a DOS disk and then just do a binary load (using the L option) of the file. This program isn't really a supported feature of CARTDISK, since I eventually plan on working things around a multi-file non-dos quick load disk. That is scheduled for a future release.

APPENDIX A: Disk Structure

This section is for reference only. The CARTDISK diskette contains the following sector allocations:

Sector	CONTENTS
1-3	Selection menu/DISKEDIT run
4-42	DOS
43-194	DISKEDIT, PCOPY, & CRTBLOAD
512-644	Cartridge Storage area
646-648	Cartridge Boot Loader
649	DOS Load File Header
650-700	CARTCOPY PLUS assembly routine
701-703	DOS BOOT Sectors
704-711	Absolute location storage
712-720	Indirect location storage

I will let you try to figure out how I can tell whether the disk is an original or a backup. Also, good luck in trying to find the internal serial numbers.