

## EXPLORING

by Faye Deere

A word to the newcomer: understand that CP/M 2.2 is not a system you can use without first doing a lot of reading. But it's well worth the time investment.

CP/M provides the capability to write sophisticated machine-language programs. All of Coleco's cartridge games are written in CP/M, as well as most of the programs for the Adam.

Opening up a whole new world for Adam owners, CP/M gave them a multitude of well-written public domain and commercially written software. However, CP/M has never been a user-friendly environment, so you must be very precise in what you do, and you have to keep a tape/disk with the CP/M system on it in the A> drive.

Remember that your A> drive is always the drive that you boot the CP/M system from. It doesn't matter if you boot from the first or second tape drive, or the first or second disk drive. The boot drive is A> drive. If you make a mistake, the CP/M system will abort, and look for the system on the disk in the A> drive. If it doesn't find it, your Adam will lock up on you. Because of this, it helps if you have more than one drive if you plan to work with CP/M a lot. In fact, when you come right down to it, two drives are really a necessity.

### Getting Messages

If you are using a tape drive, a message you will probably see frequently is "Missing block, adjust media, abort, retry." You get this message when you are trying to read or write to a data pack. CP/M has an error-checking routine it uses to search for errors and bad blocks. For some reason, the data packs generate this message quite often.

Don't panic. Most of the time, the only thing you'll have to do is remove the tape, smack it smartly against the heel of your hand, and

drive if you plan on using CP/M extensively. The second thing, which you'll discover very quickly, is that you must hit a Ctrl-C (warm boot) if you put a different tape/disk in the drive than the one with which you started. If you don't warm boot, CP/M will not recognize it. Again, if you don't have the CP/M system on the disk in A> drive, the system will crash, and you will have to reboot.

Another thing: if you try to read or write to a drive that is empty, your system will crash. Be sure you have a disk/tape in the drive before you hit that return key, or it's reboot time again.

If you want to put the CP/M system on a tape/disk, you can do this by using SYSGEN.COM. If you have files on the tape/disk, don't worry. SYSGEN will put the system on the tape/disk without destroying the files already present. I put the system on all of my CP/M tapes/disks. This is much easier than trying to remember which ones have it and which ones don't. It doesn't take up that much room, and it saves the time used up by rebooting. It often saves a lot of work, too.

If you have to reboot, you will lose what you were working on. Most of the time I keep a disk with the system and the utilities I constantly use in drive A>, and the disk I am currently working on in drive B>. This way I have everything I need, right where it can do me the most good.

CP/M is very hard on tape drives because of the extensive reading and writing done to load and copy CP/M files. This is another reason that I suggest getting a disk drive if you plan on working a lot in CP/M. They seem to hold up better than the tape drives under extensive usage, and are a heck of a lot faster to use.

Before using a CP/M file of any type, read the DOC or README files first. Trying to use a file in CP/M if you don't know what you're doing is like driving blind-

Remember that your A> drive is always the drive that you boot the CP/M system from. It doesn't matter if you boot from the first or second tape drive, or the first or second disk drive. The boot drive is A> drive. If you make a mistake, the CP/M system will abort, and look for the system on the disk in the A> drive. If it doesn't find it, your Adam will lock up on you. Because of this, it helps if you have more than one drive if you plan to work with CP/M a lot. In fact, when you come right down to it, two drives are really a necessity.

#### Getting Messages

If you are using a tape drive, a message you will probably see frequently is "Missing block, adjust media, abort, retry." You get this message when you are trying to read or write to a data pack. CP/M has an error-checking routine it uses to search for errors and bad blocks. For some reason, the data packs generate this message quite often.

Don't panic. Most of the time, the only thing you'll have to do is remove the tape, smack it smartly against the heel of your hand, and reinsert. It should now work, unless the tape has not been formatted correctly or is defective. This problem usually only happens if a disk has a bad block. You can eliminate this hazard by verifying the blocks when you format the disk. This is one reason that I recommend getting a disk

the files already present. I put the system on all of my CP/M tapes/disks. This is much easier than trying to remember which ones have it and which ones don't. It doesn't take up that much room, and it saves the time used up by rebooting. It often saves a lot of work, too.

If you have to reboot, you will lose what you were working on. Most of the time I keep a disk with the system and the utilities I constantly use in drive A>, and the disk I am currently working on in drive B>. This way I have everything I need, right where it can do me the most good.

CP/M is very hard on tape drives because of the extensive reading and writing done to load and copy CP/M files. This is another reason that I suggest getting a disk drive if you plan on working a lot in CP/M. They seem to hold up better than the tape drives under extensive usage, and are a heck of a lot faster to use.

Before using a CP/M file of any type, read the DOC or README files first. Trying to use a file in CP/M if you don't know what you're doing is like driving blindfolded—it can cause a lot of grief! You should actually make a hard copy of the files and keep them for reference. They frequently come in very handy. To make a hard copy, do a Ctrl-P, then type: TYPE (filename). This sends the file to the printer at the same time it appears on your

# DAM ADAM ADAM ADAM ADA/ G CP/M 2.2

screen. Also, before you start working in CP/M, you may want to use a utility like TRUE30.COM to shorten the length of the printed line. CP/M was built for an 80-column screen and, in order to read it, you must scroll across your screen using the Control key. TRUE30 shortens the

MS-DOS.

7. LIST will send a file to your dot matrix printer. You cannot use this command with an Adam printer.

8. The COPY command is now built in, so you no longer need COPY.COM on your system tape/disk.

---

*CP/M 2.2 is a system well worth the necessary investment of time.*

---

line to 30 characters, so you can see everything on the screen without scrolling. This makes things a lot easier. You can't use this utility with some programs, however. WordStar and Nevada Basic need the full 80 columns to work properly.

## Enhancements to CP/M

Before we go any further, I want to tell you about a new CCP that corrects a number of bugs in the BIOS of Adam's CP/M system. This was sent to me a couple of years ago by Larry Sparks. Now I don't want to confuse you before you even get started, but I think these enhancements to the CP/M 2.2 system are a great help, and make operating the system much easier. Here are the things it changes:

1. The DIR command will give you an alphabetical listing and the length of each file. You also get a listing of your free space.

2. The TYPE command will now print one screen of information and then stop until a key is pressed. You don't have to use Ctrl-S or Ctrl-Q anymore.

3. Drive and User area changes are now possible as with ZCPR. To change drives, enter the new drive (M:). To change User areas, enter the new User area (1:). To change both, enter both (M2:). The USER command is deleted.

9. The size of the RAM disk (M:) is increased to 61K usable. Drive M: will now survive the Reset switch, in case your Adam freezes and you have to use it. Save your work to the RAM disk as you go, and even Reset won't cause you to lose it!

10. This BIOS supports the EVE SP-1 and its equivalents.

11. This BIOS supports a second device, such as the Orphanware 80-column card.

12. This patch fixes a bug, so that what's in the disk buffer is always written to disk/tape before the system resets. This is what caused the trouble with Edfile, and now the read-random function will work correctly.

13. This patched CP/M system can easily be modified to support both single-sided and double-sided disk drives simultaneously. These patches are the work of Tony Morehen of Canada who wrote FileManager for us. We have this patch in our public domain library, as I'm sure do many other groups. I suggest that, if possible, you get it and patch your CP/M 2.2.

If you take a look at one of your CP/M directories, you might find some .LBR files, which are library files. This means that a group of files, all pertaining to the same subject, have been saved together to



**CLASSIC COMPUTERS****ADAM AD****◆ EXPLORING CP/M 2.2**

saved in this manner will have a "Q" in the middle of the extension (AQM or DQC).

To store a group of files into a library file, you will need a utility such as NULU.COM. To "de-library" a file, you will need DELIB.COM. To squeeze a file, you need the SQZ.COM utility, and to un-squeeze them, you will need USQZ.COM. You should not try to squeeze a COM or small files, since you will save very little storage space, and it wouldn't be worthwhile. When you use the "squeeze" and "library" utilities together, you save a considerable amount of space, and this makes a big difference when you are transferring files by modem. As the saying goes, "time is money."

Before you start to DELIB a library file, copy the library file to a fresh disk. This way, you will have room for each file as it is released from the library file. Also, if you make a mistake and lose everything, you will still have the library file on another

disk so you can start over from the beginning. Don't laugh: it can happen.

Once you have the files in the library separated, you may want to copy them all to another disk. This can be accomplished very easily with another utility that I like very much, called NSW.P.COM or NEWSWEEP. This utility allows you to mark all the files you want transferred, and walk away while your Adam copies all the marked files to the other disk. This saves a lot of time, since you don't have to do each file at a time. While you don't have to stay there and watch, you might check on it once in a while. If there is a bug in a file, or if you run out of room on the receiving disk, the program will abort and sit there until you come back.

Now we need to talk a little bit about assembling a CP/M machine language program. Relax—it's not as hard as it sounds.

First, you have to create an ASM file consisting of the machine-lan-

# DAM ADAM ADAM ADAM ADA

guage assembler instructions that make up the program. The following program that I found in an old NIAD newsletter, which gives you a 30-character column, is a good example of machine-language assembler instructions: `ORG 100h  
LXI H,OEA68h  
MVI A,30 (or any length you want)  
MOV M,A  
JMP 00  
END.`

There are two ways to create a file containing this program. One way is to type the program in SmartWRITER, and then use the ADAM command to transfer it to CP/M. The second way is to use a CP/M editor, such as CREATE3, to type in the file. Remember that you can turn off the SmartKEYS if they take up too much room at the bottom of your screen by pressing the Shift and Undo keys at the same time. Just repeat this process to bring them back.

After typing in the program, save it in a new disk/tape under whatever filename you like, with an .ASM extension. Be sure you zap it

with the ADAM command first if you typed it in SmartWRITER. Now transfer the ASM.COM and LOAD.COM programs to the disk on which you have saved the new file. Let's pretend you named the above program WIDE30.ASM. To assemble the program, type: `B:ASM  
WIDE30:BBZ.`

The first letter, B:, could be an A:, or whichever drive contains ASM.COM. If the file to be assembled is on a disk in your A> drive, the above extension would read .ABZ, which means that the file you typed (WIDE30.ASM) is in the A> drive and that you're saving it to a disk in your B> drive. If WIDE30.ASM is in A> drive and you want to save the finished file back to the A> drive, the extension would read .AAZ.

If there have been no errors up to this point, type: `LOAD  
WIDE30.HEX.` This will take the NEX file that ASM.COM created and turn it into a .COM file, which is now ready to run. You did it!