E X T E N D E D   A D O S - 3

Welcome to Extended ADOS-3! Since standard ADOS-3 is required for use of Extended ADOS-3, this documentation will assume prior familiarity with the ADOS-3 documentation.

One important difference between Extended ADOS-3 and standard ADOS-3 is that ADOS-3 can be used in RAM mode from disk, whereas Extended ADOS-3 must be burned into a 27128 EPROM, where it shares space with standard ADOS-3, for full use. The reason for this difference between Extended ADOS-3 and standard ADOS-3 is that the CoCo 3's Super Extended BASIC occupies the RAM above Disk BASIC, preventing a RAM copy of Extended ADOS-3 from being used there. Extended ADOS-3 uses the ROM "underneath" Super Extended Basic, switching in the ROMs whenever Extended ADOS-3 functions are needed and switching them out afterward. One advantage of this scheme is that compatibility conflicts are extremely rare, since nothing can have a conflicting use for these ROM addresses underneath Super Extended Basic.

We have, however, included some utilities on the Extended ADOS-3 disk that may be used individually under ADOS-3 to provide SOME of the functions in Extended ADOS-3. These utilities will be discussed later.   They are intended to provide an interim "sampler" of Extended ADOS-3's capabilities that you can explore prior to having Extended ADOS-3 burned into an EPROM.   The utilities that can be used in "stand-alone" fashion by RUNMing them are RAMDISK/BIN, MENU/BIN, WCOPY/BIN, LCOPY/BIN, DATE/BIN (optionally in conjunction with a real-time clock module CLOCKxx/BIN, using the sequence LOADM"DATE": RUNM"CLOCKxx"), and KEYRPT/BIN. These utilities can only be used one at a time, since they load to the same memory area as one another, with the exception that MENU/BIN can be used at the same time as any of the others.

Prior to getting started, make a backup of your Extended ADOS-3 disk to use as a work disk in configuring your EPROM. DO NOT WRITE TO THE ORIGINAL DISK; keep it write protected.

The Ramdisk

The Ramdisk is the only feature of Extended ADOS-3 that requires 512K for use; attempts at Ramdisk access in a 128K machine will give an IO error message.

This Ramdisk is the fastest we have seen for the CoCo 3. A good measure of Ramdisk speed is the "pure" Ramdisk operation of backing up one Ramdrive to another.   Extended ADOS-3 does a full 35-track Ramdrive-to-Ramdrive backup in 3 seconds. Its specialized formatting routines can DSKINI a Ramdrive in a fraction of a second.

Further, Extended ADOS-3's Ramdisk is unusual among CoCo 3 Ramdisk programs in that it maintains a two-byte checksum for each sector to protect data integrity and warn you when something is amiss. Whenever a sector is loaded in from a Ramdrive, the checksum is recomputed and compared with the stored checksum. If there is a discrepancy, an IO error message is given. However, there is also provision for overriding the protection given by the checksums, so that files may be loaded in from a damaged Ramdisk. The command CSUM OFF will turn off checksum verification; CSUM ON turns verification back on. Checksum verification is automatically turned on by a cold start.

All disk-related commands, including BACKUP, COPY, SAVE, DSKO$, DSKINI, etc., are usable with the Ramdrives. A minor difference between physical disk and Ramdisk commands is that a Ramdrive DSKINI command may be used within a BASIC program without interrupting program execution. It is not necessary to DSKINI a Ramdrive unless you wish to erase data previously contained on it, which using the Ramdrive as a BACKUP destination drive will also accomplish.

The capacity of ADOS-3's Ramdisk allows for two Ramdrives of 35 or 40 tracks, OR a single 80-track Ramdrive. This number of tracks is determined when you configure your Extended ADOS-3 EPROM with the ECUST/BAS program. (Extended ADOS-3, unlike standard ADOS-3, allows for different numbers of tracks on different drives.)

When the CoCo 3 is first powered up, the Ramdrives are automatically formatted and are immediately available for use. On subsequent cold starts, the presence of an already-formatted Ramdisk will be detected, and the Ramdisk contents will be left intact.

Since all Ramdisk data will be lost when the CoCo is turned off, make sure you copy any important data from the Ramdisk to a physical disk prior to powering down!

The Ramdisk should be fully compatible with programs that use 128K only, but will conflict with many 512K machine-language programs that use the same memory that the Ramdisk employs. (Note that many programs REQUIRE only 128K but will USE 512K if the extra memory is present, and thus may conflict with the Ramdisk.) So, you should not expect Ramdisk files to remain undamaged after using 512K programs, and you should be aware that attempts to save files to the Ramdisk from within a 512K program may cause the program to crash or otherwise create problems. The mere EXISTENCE of the Ramdisk should cause no problems for 512K programs, so long as you do not try to USE it from within the programs. Some 512K programs have a provision that allows the user to specify that they are to be run in 128K mode (or can be easily modified to enable this), which would allow Ramdisk compatibility at the cost of forgoing whatever benefits 512K mode allows.

The CONFIG command

CONFIG allows for the Ramdrives to be assigned to drive numbers that BASIC recognizes. More generally, CONFIG allows for the assignment of drive numbers 0 to 3 to various I/O channels. For example, you could assign Drive 0 to reference the first Ramdrive and Drive 1 the back side of your second double-sided drive. Here, it is useful to draw a distinction between LOGICAL and PHYSICAL drive numbers. LOGICAL drive numbers are the ones that appear in BASIC commands. They are constrained in Extended ADOS-3 to be between 0 and 3, just as in standard Disk BASIC. PHYSICAL drive numbers are dictated by your hardware configuration, including the settings of the drive select switches inside your drives. In general, physical drive zero refers to the drive closest to the CoCo, physical drive 1 the next closest, etc. A double-sided drive is considered as ONE physical drive, even though it may be referenced by TWO logical drive numbers, one for the front and one for the back. (When you configure your EPROM using ECUST/BAS, step rates and numbers of tracks will be assigned to PHYSICAL, not logical, drives, so that a 6-millisecond 80-track drive will be recognized as that regardless of what logical drive numbers you assign to it using the CONFIG command.) Hardware considerations dictate that a CoCo can be connected to a maximum of four single-sided or three double-sided drives.

Here is an example of the use of the CONFIG command:

CONFIG 0 1 B0 B1.

This command assigns logical drive numbers 0 to 3 respectively to physical drive 0 (front), physical drive 1 (front), physical drive 0 (back) and physical drive 1 (back). Here's another example:

CONFIG R0 R1 0 1

Here, logical drive numbers 0 to 3 are respectively assigned to Ramdrive 0, Ramdrive 1, physical drive 0 (front), and physical drive 1 (front). The four quantities appearing to the right of CONFIG can be the following: 0, 1, 2, or 3 (referencing up to four single-sided drives, or the front sides of double-sided drives); B0, B1, or B2 (referencing the back sides of up to 3 double-sided drives); and R0 and R1 (referencing two Ramdrives). If one or both Ramdrives have been declared as containing 80 tracks in the Extended ADOS-3 EPROM configuration procedure, both R0 and R1 will reference the same Ramdrive. CONFIG commands can be included within running programs, so effectively you can have simultaneous access to up to three double-sided drives, acting as 6 logical drives, PLUS two Ramdrives--the equivalent of EIGHT drives in all, each having up to 80 tracks worth of capacity. This feature should be quite valuable to anyone running a BBS (electronic bulletin board).

Extended ADOS-3 also has provision for a "shorthand" way of specifying up to four frequently-used CONFIG sequences. These can be referenced simply by CONFIG A, CONFIG B, CONFIG C, and CONFIG D. The definitions of CONFIGs A, B, C, and D are determined in configuring your Extended ADOS-3 EPROM. CONFIG A has a special importance, as that is the drive configuration that goes into effect whenever a cold start is done. The CONFIG command also has one more form: CONFIG all by itself, with no arguments. This command prints the current drive configuration on the screen. The DISABLE command causes the drive configuration to revert to whatever it was in the standard ADOS-3 that was incorporated into the Extended ADOS-3 EPROM.

The CONFIG command should be used sparingly, as it is possible to become confused as to what drive number references what I/O channel otherwise. To minimize the possibility of disastrous mistakes, a warning is given prior to executing a DSKINI or BACKUP command if anything other than the standard CONFIG A is in effect, indicating what drive(s) are involved and prompting for confirmation. However, no such warning is given on other disk-access commands, so BE CAREFUL if you are using the CONFIG command!

The program RAMDISK/BIN will allow you to try out the Ramdisk and the CONFIG command prior to burning your EPROM. RUNM"RAMDISK" will cause the Ramdrives to be initialized and will put CONFIG A into effect, which in this utility is defined as 0 1 R0 R1. That is, logical drive numbers 2 and 3 will access Ramdrives. The number of tracks assumed for each drive will be whatever your ADOS-3 is configured for. The CONFIG command will be recognized, so you can re-configure logical drive number assignments to be whatever you like. The RAMDISK/BIN program, like most of the other stand-alone utilities on the Extended ADOS-3 disk, will not survive a reset. However, if you cold start and re-execute RAMDISK/BIN, you will find the files previously saved to the Ramdisk are still there.

The MENU command

This command provides a directory display and allows for performing various operations (execute, LOAD, KILL, COPY, and SCAN) on a file by using the up and down arrow keys to point at the desired file and then typing a

single character representing the operation desired. It is a much more powerful version of the BOOT3/BAS utility that is included on the standard ADOS-3 disk. A list of the legal single-character subcommands is shown onscreen in the MENU display, so operation is mostly self-explanatory. You can try out this command prior to burning your EPROM simply by typing RUNM"MENU". When used from EPROM, MENU may take a drive-number argument (e.g., MENU 2), otherwise it initially accesses the default drive. (However, you can change from one drive to another once MENU is in operation by typing a desired drive number.) The MENU/BIN file serves a double purpose: it is incorporated into the Extended ADOS-3 EPROM and may also be used as a stand-alone RAM utility under standard ADOS-3. The MENU utility has its own separate configuration program, MENUCUST/BAS.

To configure the MENU utility, follow the comments in MENUCUST/BAS, making changes using the BASIC line editor. The resulting version of MENU/BIN will be incorporated into your Extended ADOS-3 EPROM when you do your main configuring using the ECUST/BAS program to be described later.

One feature of MENU that is not visible in its onscreen instructions is this: holding down the SHIFT key while pressing the up or down arrow will speed the motion of the cursor through the filename display. Another non-displayed feature is that SHIFT-BREAK, instead of just BREAK, will take you to the 32-column screen width in exiting the MENU display.

Wild-card COPY and KILL

The wild-card feature allows you to copy or kill a number of files in a single operation. The asterisk (*) character is a wild-card character, in that it matches any series of characters in a filename or extension. Thus, the command COPY"*.*" TO 1 would result in all files on the default drive being copied to Drive 1. The command COPY"*.BIN" TO 1 would copy just files with a .BIN extension. KILL"*.BAS" would kill all BASIC files. COPY"T*.BAS" TO 1 would copy all BASIC files whose names begin with the letter T. (The asterisk will match all characters from the position where it is encountered up until the end of the filename or extension.) There is also a second type of wild-card character, the question mark. Question mark will match any SINGLE character in a filename. Thus COPY"T?.BAS" TO 1 would copy only two-letter BASIC filenames beginning with T. The destination filename, if one is given (the usual format would specify only a destination drive number) is ignored on a wild-card copy; the destination filename will be the same as the source one.

The COPY and KILL commands (whether wild-cards are used or not) may be followed by one or more options, signified by single letters separated by commas and added to the end of the command, e.g., COPY"*.BIN" TO 1,O,R. The options are defined as follows:

O   ("Output")--Causes filenames to be output to the screen as
     files are copied or killed.

A   ("Ask")--Displays each filename and prompts for a Y or N response
     prior to each copy or kill. BREAK may be used to abort the command.
     The especially useful commands COPY"*.*" TO 1,A and KILL"*.*",A will
     prompt for each file on the disk in turn.

K   ("Kill")--Affects the COPY command only; causes the source file
     to be killed after the copy is completed.

R   ("Replace")--Affects the COPY command only; if the destination
     file exists, it will be automatically replaced if the R option is
     selected; otherwise the user is prompted as to whether he wishes

to replace. The R option can be thrown on in the middle of a
wild-card copy by responding R instead of Y to the "Replace?"
question, so as to avoid a number of unwanted subsequent "Replace?"
prompts.

The utility WCOPY/BIN may be used to add the wild-card COPY and KILL
functions to standard ADOS-3

Changes to the BACKUP and DSKINI commands

The BACKUP and DSKINI commands have both been optimized for greater
speed. BACKUP now works twice as fast as before for a full disk, and DSKINI
has been speeded up by almost 30 percent.

Over and above the gain realized by the doubling of the speed of BACKUP
for a full disk is a further gain realized on account of the fact that in
Extended ADOS-3, tracks that do not contain any active files are not copied at
all, resulting in very substantial additional savings when backing up
partly-full disks. On account of this "GAT backup" feature (backup according
to the Granule Allocation Table), BACKUP will always be much faster than the
wild-card copy command COPY"*.*" as a method of copying all files on a disk.

The fast backup capability works especially well in conjunction with the
use of the Ramdisk, since a frequent Ramdisk operation is backing up a
physical drive to or from a Ramdrive.

In certain rare circumstances, such as backing up a disk with a damaged
Granule Allocation Table or backing up an OS-9 format disk, you may wish to
bypass the "GAT backup" feature and have all tracks copied. This may be done
using the "A" option ("A" for All) that has been added to the backup command,
e.g., BACKUP 0 TO 1,A.

A very useful additional option has also been added to BACKUP to provide
for FORMATTING of the destination disk prior to doing the actual backup. This
is the "F" option. The command BACKUP 0 to 1,F accomplishes both the
formatting and the backup operations with a single command. The "F" option is
prohibited on single-drive BACKUPs to avoid the risk of inadvertently
formatting the source disk.

A final option, less frequently useful, is the ability to specify how
many tracks are to be copied. Normally, the number of tracks copied will be
the number that the source and destination drives have been configured to
have. If there is a mismatch between the source and destination drives in
this regard (recall that in Extended ADOS-3, all drives need not have the same
number of tracks), an FC ERROR message will result. The number-of-tracks
option bypasses this requirement for a match between source and destination
drive and causes BACKUP to simply use the number of tracks specified. A
principal use of this command would be to allow a user who had all drives
configured for 40 tracks to make a backup of a 35-track diskette with BACKUP 0
to 1,35,F. (Note that the number-of-tracks option is here combined with the F
option, which will cause 35 tracks to be formatted on the destination disk
prior to the backup. Options may appear in any order.) Note that this
command does not alter the number of tracks the source or destination drives
are configured for; the disk will still have the wrong format for the drive
that it is in. (The NUTRAX2/BIN utility, a newer version of the NUTRAX/BIN
utility supplied with ADOS-3, could be used to transform the disk into a
40-track one.)

When more than one BACKUP is to be made of a source disk, there is a
substantial advantage to first backing up the source disk to a Ramdrive and
then using the Ramdrive as the source drive for the BACKUPs. (The ADOS-3

feature of being able to repeat a command by typing slash-enter-enter will come in handy for repeating the BACKUP command in this situation.) A BACKUP between a Ramdrive and a physical disk is about twice as fast as a physical disk to physical disk BACKUP.

If you have a single-drive system, you will (if you have 512K) probably want to do all BACKUPs in two steps: source disk to Ramdrive, Ramdrive to destination disk, thus avoiding the necessity for multiple disk swaps.

File dating--the DATE command and the DATE$ function

When the computer is turned on under Extended ADOS-3, you will be prompted to enter the date, unless you either (a) have a hardware real-time clock and the optional CLOCKxx/BIN module for that clock has been incorporated into your EPROM, or (b) have configured your EPROM so as to bypass the power-up date prompt. This date prompt is issued only at the first cold start after power-up; at subsequent cold starts, the previously-entered date string will be printed on screen immediately after your startup message. Only if a machine language program has destroyed both the previously entered date string and its RAM backup copy, stored in two different and seldom-used parts of 128K memory, will you be prompted a second time for a date.

The date string that you enter is used for three purposes:

1) It is saved in the directory entry when files are created, and is displayed when the DIR command is used.

2) It is used as a heading when the LLIST command is used to list an entire BASIC program, as opposed to listing a single line or range of lines.

3) It is returned as the value of the DATE$ function.

The date string may be up to 16 characters in length and has any format you like. In fact, there is no requirement that it contain a date at all, so you could use this feature to attach a short comment to a file's directory entry instead of a date, such as "Rainbow 6/87 p56". Files created by machine-language programs may or may not have a date string incorporated into their directory entries, depending on whether or not the programmer used BASIC's routine to create the directory entry.

Only the first 12 characters of the date string are displayed when the DIR command is given in 32-column mode, so as to confine each entry to a single screen line. DIRP will print the full 16 characters regardless of screen width.

If you wish to change the date string within a session, type the command DATE, which will cause you to be prompted for a new date string.

If you have a real-time clock, the date string (containing not only the date, but the time as well) will be obtained from that. The date string will be displayed in the format "12-Apr-89 11:23". (The ":23" would not be displayed in 32-column mode due to the 12-character truncation.) Times are displayed in 24-hour format. When the controller is used with the clock unplugged, Extended ADOS-3 will detect the clock's absence and will operate as if you had no clock. Thus, it will do no harm to incorporate a clock module into your EPROM in anticipation of possibly acquiring the corresponding real-time clock at a later date.

Extended ADOS-3 currently supports three different real-time clocks: the older Disto (CRC) clock used in the RTIME (2-in-1) adapter; the newer Disto clock used in their 3-in-1 and 4-in-1 adapters; and the SmartWatch, which is

offered by SpectroSystems (see "Real-time Clock Options Available for Extended ADOS-3" later in this documentation). The clock modules for these clocks are called CLOCKRT/BIN, CLOCK31/BIN, and CLOCKSW/BIN, respectively. To have a particular clock module incorporated into your EPROM, you must store a copy of the appropriate CLOCKxx/BIN file on your work disk under the name CLOCK/BIN prior to running ECUST/BAS.

The DATE command may also take a filename argument, e.g., DATE"XYZ.BIN" will cause the current date string to be appended to XYZ.BIN's directory entry. This feature is particularly handy for dating files created by machine-language utilities that do not incorporate the date string into the directory entry when saving a file. This form of the DATE command also has an "N" option ("N" for "No Clock") that is only of interest to users of the real-time clock hardware. The command DATE"XYZ.BIN",N will cause the date string that is appended to XYZ.BIN's directory entry to be one that has been previously entered in response to a date prompt (generated by the command DATE with no argument), instead of the date string being read from the clock. This would allow a real-time clock user to attach to a file a date string that did not represent the current date/time, which would be useful for backdating a file or attaching a comment.

The DATE$ function may be used in commands like PRINT DATE$ or Y$=DATE$ to print the current date string or assign it to a string variable. This capability is highly useful in allowing BASIC programs to attach a date to their output.

The utility DATE/BIN may be used under standard ADOS-3 to provide some of the above capabilities. This utility does not do any adding to the command set, so you will be unable to use the DATE command or the DATE$ function. You will be prompted for a date string the first time you EXEC the file, and this date string will be appended to the directory entry when saving a file and shown in the directory display, as well as being used as a heading in LLISTings. If you have a real-time clock, use the command LOADM"DATE": RUNM"CLOCKxx" in order to have the date string taken from the clock. This command cannot be used if the clock is to be used in a Rompack rather than in the disk controller, as customizing for Multipak slot selection is necessary for Extended ADOS-3 to recognize the clock in this case.

LCOPY and LMOVE

The commands LCOPY and LMOVE allow for moving or copying ranges of lines or single lines within BASIC programs. The syntax is the same for both. Here are some examples:

```
LMOVE 120-150 TO 200
LMOVE 120-150 TO 200,5    an increment can optionally be specified
LCOPY 250 TO 400
```

If the destination line exists, the source line(s) are inserted immediately after it. If no increment is specified, one is calculated so as to obtain the widest possible spacing of the lines to be inserted, up to a maximum increment of 10, consistent with fitting the transferred lines into the available gap between the destination line and the line that follows it. The best spacing results if you use as the destination line the existing line that you want the transferred lines to be inserted after. Note that if you move lines, you must change any line number references to these lines in the rest of your program, as this is not done automatically.

The utility LCOPY/BIN may be used to add the LCOPY and LMOVE commands to standard ADOS-3. This utility has the side effect of temporarily disabling the PAINT and COLOR commands in order to substitute LCOPY and LMOVE for

them.

Screen snapshot feature

This feature provides for a screen dump ("snapshot") of the current text screen to be sent to the printer. It is activated by holding down the J, K, and L keys simultaneously, and may be used either from direct mode or when a BASIC program has paused at an INPUT or LINEINPUT statement. (The input that the BASIC program has paused to receive may be entered when the screen print is completed; the screen print will not disrupt program execution.) The screen snapshot feature is an alternative to using the ADOS-3 "PRT ON" command that causes all screen output to be echoed to the printer.

Parallel printer output--the POUT ON/POUT OFF commands

The command POUT ON causes printer output to be sent to a parallel port instead of the usual serial port. Currently supported are the Disto (CRC) parallel port and the parallel port included in the older J & M controllers. Some machine-language programs may not allow the parallel port to be used. In some cases (e.g., Telewriter-128), the parallel port can be used only if the serial port is disconnected from any modem or other device that would otherwise fool the program into thinking the printer was not online. Disto owners should not assume that machine-language programs that were unable to use the parallel port under CDOS will also be unable to use it under Extended ADOS-3; Extended ADOS-3 will allow parallel printer port use in some cases where CDOS does not.

80-track drives and the SKIP ON/SKIP OFF command

The command SKIP ON will allow 5-1/4 inch 80-track drives to read, write, and format standard 35-track (or, configurably, 40-track) disks. The command has no effect on drives not configured as 80 tracks in Extended ADOS-3, and no effect on Ramdrives. In SKIP mode, Extended ADOS-3 causes an 80-track drive to step twice instead of once in moving from one track to the next, thus mimicking the action of a 40-track drive.

Hardware considerations dictate certain limitations to this technique. Although the inter-track spacing accurately mimics that of a 40-track drive, the narrower head of the 80-track drive creates a magnetic data pattern that is physically narrower than that created by the head of a 40-track drive, which may create problems in some cases. Experience has shown that an 80-track drive in SKIP mode rarely has any difficulty in reading a 40-track disk, but that some 40-track drives may have difficulty in reading disks created on 80-track drives using SKIP mode. A particularly difficult case arises when the 80-track drive is used in SKIP mode to write to a disk that had some previous information written to it with a 40-track drive. The narrower head may fail to completely erase the previous contents of the wider track laid down by the 40-track drive, so that I/O errors may occur when an attempt is made to read the disk later on a 40-track drive. Therefore, it is recommended that you not use SKIP mode to write to a disk that was formatted on a 40-track drive.

Key Repeat

Extended ADOS-3 will cause a key that is held down to repeat. The delay between when the key is pressed and when repeat starts, and the interval between successive repeats, can be configured. The utility KEYRPT/BIN was included not only to demonstrate the key repeat feature but also to allow you to adjust these delay constants to your liking. Instructions for this adjustment are given in the comments in the main configuration program ECUST/BAS.

The key repeat feature is most useful in program editing, using a repeating spacebar to move the cursor forward, and a repeating backspace to move it backwards. In combination with ADOS-3's use of the up, down, and right arrows for editing, key repeat provides an editing environment that combines the best features of a screen editor and a line editor.

Configurable cold-start actions

Extended ADOS-3 can be configured to do one of two actions (or no action at all) on cold start, contingent on whether the space bar is or (configurably) is not being held down. The two actions are (1) do a DOS command, or (2) look for a BASIC program named SYS/BAS on a specified drive, and run it if it is found. (If SYS/BAS does not exist, the drive simply spins momentarily, then the usual date prompt is given, if required.) Selecting to have the action performed if the space bar is NOT being held down provides a means for an unattended CoCo to reboot whatever program it is running, such as a BBS, if there is a power interruption.

The COLS command

The command COLS prints a line on the 40 or 80-column screen that looks like this

```
0----+----1----+----2----+----3----+----4----+----5----+----6---
```

and extends for the full width of the screen. Its purpose is to aid BASIC programmers in using the LOCATE command by identifying where the various column positions are. The COLS command is not usable on the 32-column screen.

The PEEP command

This command, a configurable alternative to error trapping in the original ADOS and provided as a utility on the ADOS-3 disk, is included in Extended ADOS-3. See page 5 of the original ADOS manual for a description. Some confusion has existed concerning exactly what this command is supposed to do. It provides a means of rapidly scanning through memory; however, its display is of a graphical sort that is not readily intelligible except to those familiar with how various byte values are displayed in the 32-column text mode. It is useful in locating text strings, which are readible only if they consist of uppercase letters and of numbers. It can be used to display PMODE graphics. It is of particular use to machine-language programmers in providing an overview of the contents of memory. The version of PEEP that is in Extended ADOS-3 automatically switches to the 32-column mode when the command is given, as PEEP only supports the CoCo 1/CoCo 2 display modes, and reverts to the original screen width when PEEP is exited by the BREAK key.

The COLD command

The COLD command causes a cold start.

CSUM OFF/CSUM ON

As mentioned previously in the section about the Ramdisk, CSUM OFF causes Ramdisk checksum verification to be bypassed; CSUM ON restores verification.

Miscellaneous Niceties

1. Additional keystroke macros beyond those defined in ADOS-3 may be added in ECUST/BAS. Also, more space is available for your startup message.

2. If you interrupt a SCAN or SCANP command with the break key, files will be closed so that no AO error occurs on the next disk access. Printing is also shut off. This fixes a minor problem with ADOS-3 that was not remedied because of lack of room in the ADOS-3 EPROM for the fix.

3. A bug in the COPY command that caused a crash if an I/O error was encountered has been fixed. This bug exists in both versions of standard Disk BASIC as well as in ADOS and ADOS-3.

4. The presence of a write protect tab will now give a ?WP ERROR immediately on an attempted write, instead of after several unsuccessful retries.

5. The FILES bug in Disk Basic 1.0 and in ADOS and ADOS-3, which would sometimes cause an error message when a FILES statement was encountered in a program, has been fixed.

Compatibility

Experience to date indicates that Extended ADOS-3 shares ADOS-3's excellent compatibility characteristics. One "problem" program we have run into is Microcom's Word Power 3 word processor. In order to run Word Power 3 under Extended ADOS-3, you must first make it fully compatible with standard ADOS-3 as follows: delete line 25 of WORD/BAS that requests ADOS-3 users to issue a DISABLE:DLOAD command. In versions 3.0 and 3.1, replace this line with the following:

    25 POKE&HA282,&HBD:POKE&HA283,1:POKE&HA284,&H67:VERIFY OFF

In version 3.2, the line need not be replaced at all. You will then be able to run Word Power 3 under Extended ADOS-3 by first booting up a copy of standard ADOS-3 from disk, then running Word Power 3. The author of Word Power 3 is currently cooperating with SpectroSystems to ensure that the next release of the program will be directly compatible with Extended ADOS-3.

There is also a minor change that must be made to CoCo Max 3, versions 3.1 and 3.5, in order to achieve compatibility with Extended ADOS-3 if you have 512K. In the program BOOT/BAS (and also in BOOTRS/BAS, if you have version 3.5), insert line 165 reading as follows:

    165 IF PEEK(&HFE80) THEN POKE&H5DE6,&HA4:POKE&H5DEF,&H80 (for version 3.1)
    165 IF PEEK(&HFE80) THEN POKE&H5DE9,&HA4:POKE&H5DF2,&H80 (for version 3.5)

This change will not affect compatibility with standard Disk BASIC or with other enhanced DOSes. (CoCo Max 3 also requires DISABLE to be used.)

There is a fix needed for the Ultimaterm terminal program, if you are using an RS232 Pak. If you wish to use Ultimaterm with Extended ADOS-3, version 4.1 as patched for use with ADOS-3 is recommended. This version is available on Delphi and elsewhere. If you are using an RS232 Pak, Ultimaterm will crash on attempted disk I/O unless you make a small hardware modification to your Multipak. This modification ("strapping the interrupts") is frequently done by OS-9 users for the purpose of making the RS232 Pak slot-independent. It consists of connecting together the pin 8's on the four Multipak slots. When this has been done, Ultimaterm will work correctly IF you go to the Miscellaneous Options menu and set the slot number for the RS232 Pak to be the same slot that the disk controller is in, rather than the actual RS232 Pak slot.

Because the DSKINI routine in Extended ADOS-3 is rewritten for greater speed and moved to a different location, there may be incompatibilities with

utilities that have a "format" feature and that make ROM calls into the midst
of the DSKINI function. One such utility is the NUTRAX/BIN utility included
with ADOS-3. Included on the Extended ADOS-3 disk (and on recent ADOS-3
disks) is NUTRAX2/BIN, a newer version of NUTRAX that is compatible with
Extended ADOS-3. Fortunately, programs that do formatting are rare; most
such programs are probably "fast backup" utilities whose capabilities are no
better than those built into Extended ADOS-3.

In general, compatibility problems should be approached as with standard
ADOS-3: first try DISABLE, then DISABLE plus reset (or, equivalently,
DISABLE:DLOAD). If these do not work, try running the program after booting
up standard ADOS-3 from disk, as with Word Power 3, possibly throwing in a
DISABLE for good measure. Also, you should be aware of the fact that some
programs, especially those written for the CoCo 1 and 2, may require that they
be booted up from 32-column mode. (This is the reason for the "Exec 32
Screen" option in the MENU utility.)

Here are a couple of useful pokes that allow you to enable or disable
specific features that may be a source of compatibility problems. Both pokes
below are already accomplished by giving the DISABLE command; however,
knowledge of them may allow "partial" disables that will allow you to keep
valuable features of Extended ADOS-3 in operation while still achieving
compatibility with certain programs normally requiring a full DISABLE.

1) POKE&H10E,&HBC. This poke shuts off the key repeat feature, which may
be re-enabled by poking back the value originally found in location $10E.

2) POKE&H97B,&H55. This poke shuts off Ramdisk access and drive
configurability, as well as a few things having to do with the dating of
files. POKE&H97B,0 restores functioning. A particularly useful combination
is DISABLE:POKE&H97B,0, which will restore access to the Ramdisk even though
DISABLE is otherwise in effect.

Yet another occasionally useful poke, for either standard or Extended
ADOS-3, is POKE &HC16E,0. This poke shuts off the true lowercase on the
32-column screen and renders lowercase as inverse characters as in the
standard CoCo display. (DISABLE:DLOAD will also accomplish this.)

Configuring Your Extended ADOS-3 EPROM

The program ECUST/BAS is used for configuring Extended ADOS-3. However,
you should first configure MENU/BIN using the MENUCUST/BAS program as
mentioned earlier. You will need to place on your work disk a configured copy
of standard ADOS-3 created using the CUST3/BAS utility on the ADOS-3 disk,
using the name ADOS3T/BIN for this file. Alternately, you can run the program
SAVEROM/BAS to create such a file using the copy of standard ADOS-3 that you
are currently running from EPROM. If you have separately purchased a
CLOCKxx/BIN file for a real-time clock, place that on the work disk also,
under the name CLOCK/BIN, so that it will be included in Extended ADOS-3. As
with ADOS-3, configuration is accomplished by using the BASIC line editor to
make changes in the configuration program ECUST/BAS, following instructions
given in the comments of that program. Places where you may want to make
changes are preceded by comment lines beginning with four asterisks. When you
have done your editing of ECUST/BAS, simply type RUN. The program will create
and save a file called EPROM/BIN suitable for burning into a 27128 EPROM. It
will also prompt you for a filename for saving your edited version of
ECUST/BAS.

In configuring the number of tracks on each drive, it is recommended
that you not mix 35 and 40-track drives, but stick with one or the other.
This is because of the potential for getting your 35 and your 40-track disks

mixed up. As noted in the supplement to the original ADOS documentation, if you save a file to a 40-track disk using a 35-track configured DOS (or, in this case, using a drive configured for 35 tracks), you will destroy the portion of the GAT corresponding to the upper five tracks.

Burning and Installing Your Extended ADOS-3 EPROM

Extended ADOS-3 users may have their EPROM/BIN files burned into a 27128 EPROM at a cost of $15 by sending a disk containing the file to either Jim Smith, 2241 Bahama Drive, Miramar, FL 33023 (phone 305-966-7143), or to Marty Goodman, 1633 Bayo Vista Avenue, San Pablo, CA 94806 (phone 415-233-8032). These prices are current as of April, 1989. Be sure to include the serial number of your Extended ADOS-3 with your order. The price includes the chip, the burning, and shipping.

If you have a Tandy or early J&M controller that has only a 24-pin ROM socket, you will need an adapter in order for the controller to be able to accommodate the 28-pin 27128 EPROM. (Unfortunately, there is no 24-pin EPROM with enough capacity to accommodate Extended ADOS-3.) SpectroSystems has such an adapter available to purchasers of Extended ADOS-3, at a cost of $10. Please include $2 shipping and your Extended ADOS-3 serial number. Installation requires the soldering of one wire to one of the lands of the disk controller's edge connector, a relatively easy soldering job even for a novice.

If you have the Tandy FD502 controller, you already have a 28-pin socket; however, the controller needs a minor modification in order to be usable with a 27128 EPROM. This modification consists of severing the trace connecting pin 26 of the ROM socket to the adjacent pin 27, and connecting pin 26 to pin 37 on the controller's edge connector.

Third-party controllers are generally directly compatible with the use of a 27128 EPROM; however, in some cases a jumper must be moved to accomplish this. Consult the instructions that came with your controller, or inquire of the manufacturer.

Real-time Clock Options Available for Extended ADOS-3

The drivers for the Disto real-time clocks, CLOCKRT/BIN and CLOCK31/BIN, are available for $5 plus shipping from SpectroSystems.

SpectroSystems also has available the SmartWatch, a clock that is usable in any make of controller with a 28-pin DOS socket. The SmartWatch is a chip that plugs into the DOS socket, and your Extended ADOS-3 EPROM plugs into it. (Due to the extra height, the controller's cover may not be able to be fully screwed down, but the SmartWatch has a low enough profile that the controller can still be plugged into the CoCo with no difficulty.) You can inspect a SmartWatch at any Radio Shack, where is it catalog number 25-1033, list price $39.95. The SmartWatch has its own lithium battery, rated for 10 years of service, for maintaining the time independently from the computer. SpectroSystems is offering it for $35 plus shipping, including an Extended ADOS-3 driver (CLOCKSW/BIN), a program for setting the clock (CLOCKSW/BAS), and an OS-9 Level 2 driver written by Jim Smith. When installing the SmartWatch, make certain that its notch faces the same direction as the notch in the EPROM.

The SmartWatch can also be used in a Rompack, if you have a Multipak. This option is needed by those with controllers having 24-pin DOS sockets, since the SmartWatch will not fit properly into the 28-to-24-pin adapter. The Rompack also might be attractive for those who wish to avoid using the SmartWatch in the controller in order to allow the controller's cover to be

able to fit snugly. The cost of the Rompack is $10 plus shipping. If the SmartWatch is used in a Rompack, you must modify your copy of SETSW/BAS according to the comments in the program, in addition to doing the necessary customizing for slot selection in ECUST/BAS.

Please included $2 shipping and your Extended ADOS-3 serial number with all orders.

Acknowlegments

The wild-card COPY/KILL feature is based heavily on a similar utility written by Roger Bouchard, and is used with his permission. The MENU feature is descended from the BOOT3/BAS utility written by Mike Tolbert and modified by Roy Cosby.

I am also indebted to Marty Goodman, Tim Koonce, Jim Smith, Mike Ward, and various people on Delphi who made valuable suggestions that were incorporated into the development of Extended ADOS-3.

Appreciation is extended to Tony DiStefano and CRC Computers, Inc. for providing hardware and associated source code.

Note: If you have purchased the Extended ADOS-3 driver for the SmartWatch, the SAME DISK that contains this driver also contains the OS-9 files necessary for using the SmartWatch under OS-9 Level 2. Under OS-9, you will be able to access the following files: getclk.bin, a program to read the SmartWatch; swset.bin, a program to set the SmartWatch; swclock.bin, a system module for the SmartWatch that can be incorporated into your boot file; and ipatch, a patching utility. Also included are various .ipc files for modifying getclk, swset, and swclock to work with the clock in Slots 1, 2, or 3 of a Multipak Interface (MPI) by using ipatch.

I. getclk and swset

    If you are using the SmartWatch in the disk controller, with the controller either in MPI Slot 4 or plugged directly into the CoCo, copy getclk.bin and swset.bin into your CMDS directory under the names getclk and swset.

    If the SmartWatch is contained in a Rompack that is to be used in some other MPI slot than Slot 4, you must use ipatch to modify getclk and swset for the appropriate slot. With your boot disk in /d0 and the clock driver disk in /d1, give the following commands to modify getclk and swset for Slot 3:

        chx /d1
        chd /d1
        ipatch getclk3.ipc getclk.bin /d0/cmds/getclk -v
        ipatch swset3.ipc swset.bin /d0/cmds/swset -v
        chx /d0/cmds
        chd /d0
        attr cmds/getclk pe e
        attr cmds/swset pe e

    For Slots 1 or 2, use the appropriately-numbered versions of getclk and swset instead of getclk3 and swset3.

    getclk has the following syntax: getclk <n>& where n = 1 to 60. Simply typing getclk will cause the OS-9 system time to be updated using the SmartWatch on a one-time basis. Typing getclk <n>& will read the clock and update every <n> minutes. For example, to update the system time automatically every 5 minutes, type: getclk 5&

    Once having placed the appropriate version of getclk in your CMDS directory, you will probably want to modify your startup file by deleting the setime command and substituting a getclk or getclk <n>& command in order to have the system time obtained from the SmartWatch at startup.

    Typing swset will allow you to set the SmartWatch.

II. swclock

    As an alternative to using getclk, experienced OS-9 users may instead wish to replace the CLOCK module in their boot file with swclock.bin and make up a new boot file using OS9Gen. Prior to doing this, ipatch should be used in the same manner as above to modify swclock for the correct MPI slot, if a slot other than 4 is to be used, e.g.;

        ipatch swclock3.ipc swclock.bin /d0/cmds/clock -v

    Once a new boot disk has been created, the clock module may be deleted from the CMDS directory. Upon boot, the system clock will be updated from the SmartWatch every 60 seconds.

    Support is provided in the driver for the OS-9 direct page variable D.Daywk corresponding to the day of the week that is entered at the "Enter Day:" prompt of the swset utility.

    These programs are copyright (c) 1989 by James W. Smith Jr., 2241 Bahama Drive, Miramar, Florida 33023. Distribution rights are granted to SpectroSystems of Miami, Fl. This software has NOT been released to the public domain. It may be FREELY distributed if, and only if, all copyright notices remain intact. You MAY NOT sell this software or otherwise profit from it.

INSTALLATION: The SmartWatch may be installed in a disk controller with a 28-pin ROM socket simply by piggybacking the disk ROM chip on top of it, making sure the notch faces the proper direction. For OS-9 use only (but not for use with Extended ADOS-3), it can be used in a controller with a 24-pin ROM socket by plugging it in with pins 1, 2, 27, and 28 hanging off the end, provided that you connect the hanging-off pin 28 to pin 26. The SmartWatch can also be used in a Rompack, available for $10 plus $2 shipping from SpectroSystems, 11111 N. Kendall Drive, Suite A108, Miami, Fl 33176.