NIBBLES & BITS

The Comprehensive Monthly Newsletter for ADAM Users

Route one, Box 29-G
Oak Hill, WV 25901

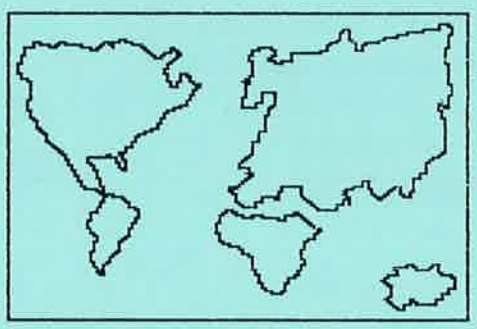
March 1987
vol: 1, nmb: 9
single issue: $3.50

Thought for the month: *"Success comes in cans; failure in cants."*

This issue includes 6 SmartBASIC program LISTs and 2 disassembled Z80 lists.

Designed and printed with the amazing ADAM™ computer.

ADAM ADAM ADAM ADAM ADAM
ADAM ADAM
ADAM ADAM
ADAM ADAM
ADAM ADAM
ADAM ADAM
ADAM ADAM
ADAM ADAM
ADAM ADAM
ADAM ADAM ADAM ADAM ADAM



The N&B Staff

EDITOR-IN-CHIEF:
Dr. Solomon Swift

DESIGN EDITOR:
Tim Whetstine

TECHNICAL EDITOR:
Chris Davidson

CIRCULATION DIRECTOR:
Tony Michaels

CONTRIBUTING EDITORS:
Janet Weston
Ted Johnson
Cindy Herrington

PUBLIC NOTICE

"NIBBLES & BITS" is published monthly by **DIGITAL EXPRESS.** Individual issues may be purchased for the current month or for a back issue for $3.50; the *premier* issue was July, 1986. The standard subscription rate for one year (12 issues) is $22.00 in the USA, its possessions, and Canada; and, the annual rate is $24.00 in other foreign countries. The standard subscription rate for six months (6 issues) is $12.00 in the USA, its possessions, and Canada; and, the semi-annual rate is $16.00 in other foreign countries.

We welcome contributions of original reviews, programs, articles, questions, and comments. Please include your subscription ID number from your mailing label on all written correspondence to us.

Your subscription ID number is on the first line of your mailing label (affixed to the newsletter). It is a 10 digit code. The first four digits are the month and year of the final issue in your current subscription. Following the ID number is a brief message. If this is your final issue, the message will read "FINAL ISSUE!!!". If this is your penultimate (next to last) issue, the message will read "TIME TO RENEW". Otherwise, the message will apprise you of the exact number of issues remaining in your subscription (excluding the current issue). Please verify this information each month.

To insure that you don't miss any issues, please renew early and let us know promptly of any address change. Please include your subscription **ID number** on the address change notification (you can get a free address change kit from your local US post office).

DISCLAIMER

EDITOR'S NOTE

Well ... it's March and this is our ninth issue. Looking back, I can honestly say that I've learned quite a bit about our fine ADAM computer in putting together each month's "NIBBLES & BITS". Judging from your letters, many of you have learned something about ADAM through our newsletter too. Thank you very much for your compliments and your words of encouragement.

Some might classify me as a bit of a hacker. A hacker, to me, is not necessarily an expert. Rather, the term more accurately describes an attitude toward programming than a particular level of expertise. If you'd rather peck on the keyboard than stare at the nightly prime time drivel, you can safely call yourself a hacker -- your friends and family probably already do.

In my opinion, most of our subscribers are enthused programmers. This is why a significant portion of each newsletter is devoted to concepts of programming. Even when the complexities of some our programs aren't easily understood, you can still benefit from using the program.

A few months ago, I mentioned in this column the important role of your mailed-in questions in the development of "NIBBLES & BITS". This month, several pages of the HACKER'S DELIGHT department is devoted to one subscriber's question. He wanted to know how to use the SmartKEYs in BASIC programs the way that Coleco does in their software.

I've called the demonstration program **EZmenu.** It uses the HGR2 (graphics mode 2) just like the Coleco packages. It even prints the SmartKEYs and messages at the bottom of the screen. You can use it as a central menu HELLO program to control your BASIC disks and data packs. It includes many nice features. It's a long one, so take breaks frequently when entering it to prevent typos. It's the first third party effort of this type that I've seen; I hope you like it.

EDITOR-IN-CHIEF
SOLOMON SWIFT

N&B NEWS

□□□ Our second collection of public domain SmartPAINT picture files is finished. It contains 13 hi-res pictures; that makes a total of 26 available now. With the "HGR Picture Manager" program LISTed last month, you don't have to have **ShowOFF I** to view the files.

□□□ We encourage our subscribers to write to us. But, if you'd like a written reply, please include an SASE (Self - Addressed, Stamped Envelope).

□□□ Despite several attempts, we had a problem with the line spacing for SmartPAINT with the Star printers. We would like to thank the following ADAMites who were instrumental in correcting the bug:

James N. Biggs
Ed Jenkins
James M. Killingsworth
Steven L. Major
Lee Smith

□□□ We've added both of Ben Hinkle's excellent Hacker's Guides to our product list. Every serious ADAM user needs both of these detailed books.

□□□ For this month's special, we've got a FREE gift for any product order greater than $15.00. You can have a FREE blank disk or a FREE "Things To Do priority note pad" with your order -- please specify which one you'd prefer. To qualify, your order must be postmarked prior to May 15th.

□□□ We had a few typos in last month's Z80 disassemblies. We would like to publicly thank David Kennedy, who pointed out the errors. We've reprinted "asmb37" in this issue. There were two errors with "asmb35". Address 63662 should have read **245 --- PUSH AF.** Address 63667 should have read **241 --- POP AF.** With "asmb36" address 63727 needed the label **bootOK.**

□□□ Beginning April 1st we're mailing all foreign subscribtions via First Class mail. Due to the higher postage the annual rate will have to increase to $30.00 for England, Israel, and Austrailia.

□□□ Our third quarterly collection of "NIBBLES & BITS" programs is completed. Each of these packages is available to our subscribers for only $4.95 on DDP or disk. Subscribers may also get one free by sending a blank disk or DDP with a return mailer and sufficient postage.

□□□ Don't forget about the SmartPAINT picture design contest. All entries must be postmarked prior to April 1, 1987.

□□□ Be sure to take a look at our PRODUCT LIST this month. We've lowered the price on data packs and disks.

ADAM NEWS

□□□ OrphanWare has completed their 256K memory expander for ADAM. Their address is:

OrphanWare
P.O. Box 324
Canal Fulton, OH 44614

□□□ **GraphixPix I** is a new graphics package from NIAD. It allows saving and loading pictures. It also supports PaintMaster picture files.

□□□ Walter's Software recently released RAMDSK which permits access to the 64K expander as a ramdisk; it works like a super fast disk drive with SmartBASIC V1.0. They are now completing a comparable program for use with SmartBASIC 2.0.

□□□ E&T SOFTWARE is considering plans to publish an ADAM only newsletter. Subscribers to their paper will get discounts on their extensive line of ADAM products.

□□□ Strategic Software has a new package for ADAM. We'll have more information on **MicroWORKS** in a future issue.

□□□ Thanks to support from ADAMites all around the world, several companies have new projects underway exclusively for ADAM.

EXPANDING YOUR SYSTEM

PRINTER ALTERNATIVES

(part 5)

What To Look For:

The primary aspect to consider in a second printer is compatibility with ADAM. The printer must have either a standard Centronics parallel interface or a standard serial RS-232 interface. Parallel interfaces are generally more common with printers.

Printer technology has advanced so rapidly in the last couple of years that many companies are offering incredible discounts on outdated models. When you see a $500 or $600 printer marked down to $49.95, you can safely assume that it is NOT state of the art. If possible, you should consider an Epson FX compatible printer. Epson is the leader in printer development, many companies offer printers with similar built - in software and features. As we've mentioned previously, the Panasonic KX series impact dot matrix printers are an excellent buy.

More On Escape Codes:

Last month we briefly discussed printer escape codes; special ASCII sequences that activate functions of a printer's built - in software. These coded sequences can turn on/off a variety of features. For instance, on many printers the sequence **27, 51, nn** will set the vertical line spacing to the user's specification. The user can substitute any value between '1' and '216' in the variable 'nn' to achieve precise spacing. For example, if 'nn' is given a value a '108', every line feed will roll the carriage exactly $^1/_2$ inch.

The manual that comes with a printer describes how to access the various features: bold printing, elongated characters, Pica pitch, Elite Pitch, compressed pitch, bit image graphics, etc. A second printer can truly add a new dimension to your experiences with ADAM.

ADAM™ USERS FORUM

The following questions and comments have been culled from recently received mail. The reader's input is a reasonable facsimile of the actual correspondence. My response, where applicable, is generally more detailed than my written response (if any) for the benefit of all readers. Please let us know if you want your street address listed when you send FORUM questions and comments.

SmartPAINT/PaintMaster COMPATIBILTIY

Is there any way to get rid of the PaintMaster icons on the left side of the screen when printing a hardcopy with SmartPAINT from ShowOFF I.

James N. Biggs
Fory Wayne, IN

In Response: Yes. After loading the converted PaintMaster file, enter a new title -- this will erase half of the icons. Then, switch to 'TEXT' mode. Use the space bar to quickly erase the remaining icons. Now press 'CNTL-B' to change the background color. Voila!!!

IMPROVING THE 'T' COMMAND

I am writing to tell you how much I like the 40-column program in the December issue, and to warn readers of a possible problem with it. Many of my programs use "t" as a variable. If the 40-column patch is used as is, as a HELLO program, ADAM will discard any program line that uses "t" as a variable. The following modification changes the command to "TEXT2".

```
7510 DATA 64,208,3,5,84,69,88,84,50
7520 FOR x=0 TO 8: READ mc: POKE x+787,mc: NEXT
7530 POKE 6421+2*64, 220: POKE 6422+2*65, 45
```

Ruth Mather
Beckley, WV

Hacker's Guide MACROS

Ben Hinkle's MACRO program in the HACKER'S GUIDE TO ADAM volume 2 works great. However, macro definitions can easily interfere with BASIC'S math routines (SPC, TAB, ABS, ASC, etc.). After running the program, try entering the SPC command in immediate mode. If the command functions incorrectly, increase the LOMEM setting in the program until the problem clears up. Remember, all programs that use RAM lower than the new LOMEM setting must be modified. Ben Hinkle's VIEWER program in 'volume one' can help establish LOMEM by looking at page 110.

David Giampietro
San Jose, CA

People/LINK UPDATE

For the past twelve months, I have been proud to be the Coleco ADAM Section Six Chairman in the Computer Club with PLink. The Computer Club is part of the telecommunications system known as American People / Link. This system can be reached via your ADAM's modem and your local TELENET, TYMNET, or DATAPAC (in Canada) phone numbers.

In Section Six of the Computer Club is a message base and a Data Library where Public Domain and FreeWare is available for you to download. You can also upload programs that you have written and want to share with other ADAMites.

FreeWare is software that the author/programmer has made available for you to use on a limited basis. One of the more interesting "FreeWare" programs to come along is a set of three data files that allow ADAM users that have only the ADAMLink II and SmartBASIC V1.0 to up / down load machine code (non - SmartWriter) files. You can download "VIDEOTUNES" songs, "PINBALL CONSTRUCTION" games, etc.

I hope to see your ID number online soon! While online, I go by the ID of DAVID E.C.. For information on how to get logged onto PLink, you may call their customer service number toll free: 1 - 800 - 524 - 0100. In Illinois (not toll free) call 1 - 312 - 870 - 5200.

David E. Carmichael
1325 North Meridian, Apt. 201
Wichita, KS 67203-4637

BIT BY BIT

LOW RESOLUTION GRAPHICS

(part 2)

Last month we discussed the **COLOR** command. It is used in the GR (lo-res) mode to assign the color of the next block to draw on the screen.

The **PLOT** command is used to place a single block on the screen. Normally, you'll set the COLOR first, then PLOT a block. For example, try this (in immediate mode):

```
GR      [return]
COLOR = 13      [return]
PLOT 10, 30      [return]
```

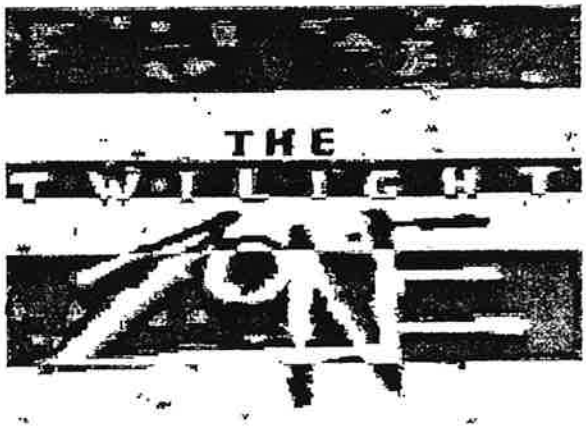
You'll see a small yellow rectangle appear in the lower left portion of the graphics window. Note that the comma that separates the two numbers with the PLOT command. The first number (0 - 39) specifies the horizontal coordinate. The second number (0 - 39) determines the vertical position.

The two programs in the next column demonstrate interesting applications of low-res graphics. The first program PLOTs a dark blue block in each screen corner. Then, it PLOTs one in the center of the screen. Next, it flashes the block between blue and white. Line number 170 controls the number of times the block flashes between the colors.

The next program illustrates a simple technique of animation. It moves a medium green block across the screen, row by row from the top of the screen to the bottom. Line number 150 erases each block after it's drawn. If you DELete this program line, the screen will fill with the medium green blocks.

```
 10 REM GRplayØ1
 20 REM flashing center block
100 GR
110 COLOR  = 2
120 PLOT Ø, Ø
130 PLOT Ø, 39
140 PLOT 39, Ø
150 PLOT 39, 39
160 PLOT 2Ø, 2Ø
170 FOR x = 1 TO 1Ø
180 FOR y = 1 TO 15Ø: COLOR  = 15
190 PLOT 2Ø, 2Ø: NEXT y
200 FOR z = 1 TO 15Ø: COLOR  = 2
210 PLOT 2Ø, 2Ø: NEXT z
220 NEXT x
230 PRINT " that's all!": END
```

```
 1Ø REM GRplayØ2
 2Ø REM simple animation of a block
1ØØ GR
11Ø FOR vv = Ø TO 39
12Ø FOR hh = Ø TO 39
13Ø COLOR  = 6: PLOT hh, vv
14Ø FOR x = 1 TO 25: NEXT x
15Ø COLOR  = Ø: PLOT hh, vv
16Ø NEXT hh: NEXT vv
2ØØ PRINT " that's all ...": END
```



BYTE-SIZED BASIC

POKES TO PLAY WITH

(part 9)

The Blocks Left Bug:

Have you noticed that the CATALOG command doesn't accurately display the Blocks Left? The problem is that DELETED files still occupy slots in the directory.

On an old datapack, run a CATALOG. Then, DELETE some unwanted program. Now CATALOG the medium again. The file was DELETEd and the Blocks Left value increased. But this is a deliberately erroneous figure. You can correct this bug with:

```
POKE 21298, 0
```

The default value of the address is 235. A routine used by the CATALOG command adds up the total blocks of DELETEd files. Then, it adds this value to the true free blocks figure. The POKE (above) simply circumvents the addition result.

The File Length Bug:

This one is also caused by the way the operating system (EOS) DELETEs files. In most cases, the CATALOG accurately reveals the block length of files.

But, if you DELETE a large file and then SAVE a small file that has a smaller length than the DELETEd file, the newer file uses all the excess blocks. There is NO simple way to correct this problem. However, you can change the CATALOG command so that it accurately reveals the blocks used and not the blocks assigned to a file. To overcome this BASIC limitation:

```
POKE 21370, 8
```

The default value of the address is 6. If you POKE a 2 into the address, the CATALOG command will display each file's start block on the medium, instead of the file's length.

EASTER DATE CALCULATOR

Have you ever wondered how the date of Easter is determined. Contrary to popular myth, the date has no direct relation to the Jewish penitential 40 days of Lent. Rather, Easter falls on the Sunday following the first full moon after the vernal equinox -- the first day of Spring. Yes ... that's right; it always falls between March 22nd and April 25th. Most frequently, Easter is celebrated in the month of April -- due to sheer probability.

Calculating the date of Easter involves adjusting the lunar calendar to determine a date of the solar calendar. Sounds complicated, huh? Don't worry, ADAM will do all the figuring for you. Just enter the program at the top of the next page and you can determine the date of Easter for **any** year. If you can get hold of an old calendar, check it out; it's 100% accurate.

SmartBASIC 2.0 FEATURES

Last month we delved a little further into our study of sprites using a simple demonstration program (page 9 of 02/87 N&B). Unfortunately, we don't have room this month for the page of blank sprite grids; we'll include it next month.

Let's continue our explanation of the program. Line 210 sets the byte flag so that DRAW works for sprites. Line 220 sets normal magnification for large sprites. Line 230 corrects the 2.0 sprite display sequence bug. Line 240 reveals an interesting trick.

Address 16788 is used as a flag for the presence of the 64K expander. Any nonzero value indicates the memory expander is accessible -- the SmartBASIC 2.0 bootstrap routine sets the value in this address.

Address 16789, is used as a flag for the version of BASIC. If the value at the address is zero, you are in STDMEM. If the value is not zero, you are in EXTMEM. The trick on line 240 simply tests which version of the interpreter is currently in RAM. Don't POKE into these addresses -- the results could be fatal.

On page 19 of our October 1986 issue, we listed the primary command table for BASIC 2.0 (in STDMEM). On page 9 of this issue we've listed the table for BASIC 2.0 in EXTMEM. The EXTMEM command re-loads the lower 16K of the interpreter. This is why so many POKEs are different in STDMEM and EXTMEM.

```
   1Ø REM Easter Date Calculator
  1ØØ TEXT: VTAB 2: HTAB 2: INVERSE: PRINT " Easter Date Calculator "
  11Ø NORMAL: VTAB 4: PRINT " Enter any negative number to"
  12Ø PRINT " exit the program.": PRINT
  13Ø PRINT: PRINT: PRINT " Calculate Easter's date for"
  14Ø INPUT " what year? ";yr$: yr% = VAL(yr$)
  15Ø IF yr% < Ø GOTO 1ØØØØ
  16Ø IF yr% < 1ØØ THEN  yr% = yr%+19ØØ
  17Ø w$ = " Easter date "
  2ØØ a% = yr%/19: a% = yr%-a%*19
  21Ø b% = yr%/4: b% = yr%-b%*4
  22Ø c% = yr%/7: c% = yr%-c%*7
  23Ø d% = 19*a%+24: d% = d%-INT(d%/3Ø)*3Ø
  24Ø e% = 2*b%+4*c%+6*d%+5: e% = e%-INT(e%/7)*7: PRINT: PRINT
  25Ø dt% = d%+e%-9: ON dt% <= Ø GOTO 3ØØ: ON dt% > Ø GOTO 4ØØ
  3ØØ dt% = dt%+31
  31Ø PRINT w$;yr%;": March ";dt%: PRINT: PRINT: GOTO 13Ø
  4ØØ IF dt% <> 26 GOTO 42Ø
  41Ø PRINT w$;yr%;": April 19": GOTO 13Ø
  42Ø IF dt% = 25 GOTO 44Ø
  43Ø PRINT w$;yr%;": April ";dt%: PRINT: PRINT: GOTO 13Ø
  44Ø PRINT w$;yr%;": April 18": GOTO 13Ø
 9999 END
1ØØØØ TEXT: PRINT " program terminated.": END
```

Easter commemorates the resurrection of Christ after His crucifixion on the cross.

Easter falls on the Sunday following the first full moon after the vernal equinox (between March 22 and April 25 inclusive).

SmartBASIC V2.0 PRIMARY COMMAND TABLE (extmem)

TOKEN	COMMAND	EXECUTION	PARAMETER ROUTINE(S)				
1		7419	15697				
2	GOSUB	9593	16488				
3	GOTO	9516	16488				
4	INPUT	10103	16296				
5	LET	7419	15697				
6	NEXT	9939	16318				
7	PRINT	8978	16330				
8	READ	10636	16325				
9	REM	9585	16554				
10	FOR	9714	15668	16699	15641	15848	
11	IF	9489	15753	15790			
12	DATA	9585	16551				
13	DIM	8154	16325				
14	ON	9550	15641	15964			
15	ONERR	9028	16727	16488			
16	STOP	7550					
17	RETURN	9638					
18	END	7231					
19	DEF	9373	15880				
20	CLEAR	9270					
21	RESUME	9460					
22	NEW	7528					
23	POP	9659					
24	RUN	7337	15987				
25	LIST	8597	15998				
26	TRACE	7518					
27	NOTRACE	7523					
28	DEL	8726	16002				
29	CALL	11094	15641				
30	CONT	7559					
31	CLRERR	9265					
32	GET	10497	16119				
33	POKE	11153	15641	16675	15641		
34	RESTORE	10619					
35	HOME	12234					
36	DRAW	12461	15641	15775			
37	XDRAW	12489	15641	15775			
38	FLASH	12199					
39	INVERSE	12204					
40	NORMAL	12209					
41	TEXT	12214					
42	GR	12219					
43	HGR	12224					
44	HGR2	12229					
45	HLIN	12312	15641	16675	15641	16713	15641
46	VLIN	12333	15641	16675	15641	16713	15641
47	HPLOT	12530	15857				
48	PLOT	12280	15641	16675	15641		
49	HTAB	12399	15641				
50	VTAB	12409	15641				
51	SHLOAD	7527					
52	RECALL	9585	16119				
53	STORE	9585	16119				
54	WAIT	11194	15641	16675	15641	15637	
55	SPEED	11925	16662	15641			
56	ROT	12510	16662	15641			
57	SCALE	12520	16662	15641			
58	COLOR	12250	16662	15641			
59	HCOLOR	12265	16662	15641			
60	IN	13092	16686	15641			
61	PR	13062	16686	15641			
62	HIMEM	12071	16647	15641			
63	LOMEM	11947	16647	15641			
64	0	7527					
65	0	7527					
7	?	8978	16330				
66	&	11223	16554				
67	7,2	9100	16554				

HACKER'S DELIGHT

TRANSFERRING DATA

Here's a routine that's used frequently in Z80 programming. It copies the value from one address to another address. There are several ways to accomplish this; but, using the accumulator is most common.

In BASIC:

```
POKE 17115, PEEK(17240)
```

In decimal Z80 format:

```
58, 88, 67,
50, 219, 66,
201
```

In mnemonics and hex code:

```
LD  A, ($4358)
LD  ($42DB), A
RET
```

EOS PROGRAMMED DELAY

One of the EOS routines (the fifth in the table of jump vectors) is a programmed delay. Its duration is 33.75 microseconds. In human terms, this is no delay at all. But for ADAM, it's a short eternity. The routine can be executed from the jump table by CALLing address 64572 (60, 252). The actual routine begins at address 63839 (95, 249). The routine uses 17 bytes. It requires no setup before the CALL and it doesn't return any values.

It is used by the "reset all devices" routine and the "initialize input / output processor" routine. You can study the code with **asmb39** on page 24. It uses two loops. The primary loop is controlled by the value in register "B". The secondary loop is controlled by the value in the "DE" register pair. It's a standard time delay.

You might want to use it in your programs. You can POKE a 255 into address 63844 and 63845 to get a half second delay. Then, you can change the value at address 63842 to manage the number of half second delays. For example, if you put an "8" in the address, the delay will last for about four seconds.

RAMDSK TIPS

Recently, Walter's Software released a ramdisk program for SmartBASIC 1.0, RAMDSK. If you use BASIC much and you have the 64K memory expansion card, you'll find the program very useful. The program is an excellent application of a powerful concept.

The program patches portions of both the EOS and SmartBASIC. It even makes a couple of improvements to BASIC. It uses Ben Hinkle's DATA / REM space bump fix. It corrects the RECOVER 'h' file bug. And, it automatically INITs disks to 160 blocks and DDP's to 255 blocks. It does have a couple of minor shortcomings though.

First, all of the simple public domain binary BASIC converters and all of the commercial converters (including Intel-LOAD V1.0) CALL an address in the EOS for a Z80 RET. RAMDSK overwrites this address. When you try to BRUN a binary BASIC program, your system will lock up. Here's a simple correction: **POKE 64389, 201.** This will disable the ADAM printer, however. But, you can even BSAVE files to the RAMDSK for instantaneous retrieval. To enable the printer, just POKE a "205" into 64389.

Also, there is a minor bug in the ramdisk directory. If you try to store more than 35 files, you'll get an I/O error message. You can correct this minor oversight with the EZpatch program in this issue. Just POKE a "129" into address 27923 on the BASIC medium (the default value is 32).

To make the program compatible with our Intel-BEST 3.3, just POKE a "32" into address 65365 in RAM. To use our **EZkeys** enhancement with this powerful program, you need to make a couple of changes to our program (LISTed on page 19 of the January 1987 issue). On line number 8110, change the second DATA element from a "134" to a "95". On line number 8330 change the "23546" to "23507".

PR#2/PR#3 COMMANDS

Continuing from last month, the PR#2 command works just like the standard PR#1 command, *ie*, with a screen echo. The PR#3 command, on the other hand, doesn't display anything on the screen.

Another feature is included in the routine. SmartBASIC prevents the printing of some ASCII values. You can't send these codes to the printer via a normal 'PR' command. With this routine, all you need to do is POKE a value into address 1127 and then CALL 1126 to pass any values to the printer. This is great for graphics, escape codes, etc.

EZPatch

There are two ways of patching SmartBASIC. You can patch the interpreter after it's loaded into RAM -- this is usually accomplished via a HELLO program. With this option, you can not use any of the patched features until of the HELLO program has RUN.

You can also patch the actual SmartBASIC program on the DDP or disk. With this option, all the patches are in effect as soon as BASIC is booted. This is a very welcome time-saver. This month we've got just such a program, **EZpatch.**

There are few similar programs already in the public domain; but this one offers some unique differences. For instance, it includes some easy - to - use default changes, *ie*, screen color changes and the default (turnkey) drive. It also supports patching entire routines. It even includes a 'scan' option. Next month we'll LIST a similar program for SmartBASIC 2.0.

The program works by converting user friendly addresses into block locations. It will work on the BASICPGM regardless of its location on the DDP or disk.

With this program you can radically improve your SmartBASIC entering routines and changing address values permanently with ease. Be sure to use the program on a BASIC backup until you're certain there are no typos. Over the past several months we've revealed many patches. If you patch the PR#2/ PR#3 program to tape or disk, be sure to put a 'one' in address 1083 to prevent display of the Coleco title header.

DETERMINING DRIVE STATUS

The program on page 16 demonstrates a routine that determines whether or not the four ADAM drives contain media. It uses addresses 65532 through 65535 to store the four status readings.

65532 = tape one status
65533 = tape two status
65534 = disk one status
65535 = disk two status

The routine was adapted from the "EOS Boot System" algorithm which we examined in detail last month. The really nice feature about the routine is that it can be used anywhere in RAM; you can use it as a routine to CALL from a BASIC program or you can use it in a self-booting Z80 program. In the LIST on page 16, it starts at address 27600. When you CALL 27600, the addresses (mentioned above) will contain the status of each drive. If ADAM can not detect the drive at all, the status will be 'one'. If the drive power is off, the status will be '155'. If there is no medium in the drive, the status will be '255'. If there is a medium in the drive, the status will be the drive code, *ie*, '8' for tape one, '4' for disk two, etc.

EZmenu

The program that occupies pages 17 through 23 uses the HGR2 mode to display the SmartKEYs at the bottom of the screen the same way that SmartWriter does. The program is rife with advanced features: various sounds, SmartKEY labels, drive status checks, fast hi-res characters, etc.

Next month, we'll have a very similar program called **EZcopy** -- a simple utility for making back ups of your software. We'll go into more detail on the assorted routines next month. Using HGR2 mode entirely takes a lot of memory but you'll probably find that the program is well worth it.

You can use it to CATALOG, INIT, RENAME files, RENAME volumes, and RUN or BRUN programs (it automatically determines which is necessary).

```
 10 REM SmartBASIC 1.Ø editor
 20 REM Use with CAUTION!!!
 30 REM Use ONLY on a BASIC backup!!
 40 REM *** EZpatch 1.Ø ***
100 LOMEM :29696: IF PEEK(259) <> 195 GOTO 10000
110 POKE 16149,255: POKE 16150,255: mx% = 255
120 k$ = "BASICPGM"+CHR$(2)+CHR$(3): FOR x = 1 TO LEN(k$)
130 POKE x+27647,ASC(MID$(k$,x,1)): NEXT
140 DATA 62,Ø,17,Ø,108,33,160,253,205,204,252,50,255,255,201
150 FOR x = 27658 TO 27672: READ mc: POKE x,mc: NEXT
160 a1 = 27658: a2 = a1+1
170 FOR x = 1 TO 2: READ dv$(x): NEXT: DATA tape one,disk one
180 DATA common patches,single address patches
190 DATA routine patches,scan addresses,exit program
200 FOR x = 1 TO 5: READ m1$(x): NEXT
210 DATA 62,Ø,1,Ø,Ø,17,Ø,Ø,33,Ø,112,205,243,252,50,255,255,201
220 FOR x = 27673 TO 27690: READ mc: POKE x,mc: NEXT
230 b1 = 27673: b2 = b1+1: b3 = b1+6
240 DATA 62,Ø,1,Ø,Ø,17,Ø,Ø,33,Ø,112,205,246,252,50,255,255,201
250 FOR x = 27691 TO 27708: READ mc: POKE x,mc: NEXT
260 c1 = 27691: c2 = c1+1: c3 = c1+6
270 DATA TEXT colors,HGR colors,GR colors,default drive,main menu
280 FOR x = 1 TO 5: READ m2$(x): NEXT
290 FOR x = 1 TO 4: READ dv$(x): NEXT
300 DATA tape one,tape two,disk one,disk two
310 DIM z1(mx%),z2(mx%)
450 IF PEEK(64947) <> 28 AND PEEK(64936) <> 2 GOTO 500
460 dv% = PEEK(65534): GOTO 700
500 TEXT: VTAB 2: HTAB 2: INVERSE: PRINT " EZpatch 1.Ø": NORMAL
510 VTAB 4: PRINT " Use with EXTREME caution!!!"
520 VTAB 10: PRINT " Which drive has BASIC 1.Ø?"
530 VTAB 12: FOR x = 1 TO 2: PRINT " ";x;" = ";dv$(x): NEXT
540 GET k$: k% = VAL(k$): IF k% < 1 OR k% > 2 GOTO 10000
550 dv% = 2^(4-k%): POKE 65534,dv%
560 HOME: PRINT " To verify BASIC, insert the"
570 PRINT " ";LEFT$(dv$(k%),4);" in the drive and"
580 PRINT " press [return] ..."
590 GET k$: IF k$ <> CHR$(13) GOTO 10000
600 HOME: PRINT " verifying SmartBASIC ..."
610 POKE a2,dv%: CALL a1
620 IF PEEK(65535) = Ø GOTO 640
630 HOME: PRINT " BASIC not found!!!": END
640 IF PEEK(64947) = 28 GOTO 700
650 TEXT: PRINT " BASIC file length error!!!": END
700 HOME: PRINT " Which option do you prefer?": fb% = PEEK(64941)
710 PRINT: FOR x = 1 TO 5: PRINT " ";x;" = ";m1$(x): NEXT
720 GET k$: k% = VAL(k$): IF k% < 1 OR k% > 5 GOTO 10000
730 ON k% GOTO 1000,2000,5000,4000,10000
1000 HOME: PRINT " Which common patch?": PRINT
1010 FOR x = 1 TO 5: PRINT " ";x;" = ";m2$(x): NEXT
1020 GET k$: k% = VAL(k$): IF k% < 1 OR k% > 5 GOTO 10000
1030 ON k% GOTO 1100,1300,1500,1700,700
```

EZpatch 1.Ø LIST continued ...

```
1100 HOME: INPUT " TEXT background value? ";bk$: PRINT: PRINT
1110 bk% = VAL(bk$): IF bk% < Ø GOTO 700
1120 IF bk% > 255 GOTO 1100
1130 INPUT " TEXT NORMAL value? ";nm$: PRINT: PRINT
1140 nm% = VAL(nm$): IF nm% < Ø GOTO 700
1150 IF nm% > 255 GOTO 1130
1160 INPUT " TEXT INVERSE value? ";iv$: PRINT: PRINT
1170 iv% = VAL(iv$): IF iv% < Ø GOTO 700
1180 IF iv% > 255 GOTO 1160
1200 GOSUB 10100: rad = 17059: GOSUB 11000
1210 POKE 112*256+of,bk%: POKE c2,dv%: GOSUB 10300
1220 rad = 17115: GOSUB 11000: POKE 112*256+of,nm%: GOSUB 10300
1230 rad = 17126: GOSUB 11000: POKE 112*256+of,iv%: GOSUB 10300
1240 GOTO 1000
1300 HOME: INPUT " HGR background value? ";bk$: PRINT: PRINT
1310 bk% = VAL(bk$): IF bk% < Ø GOTO 700
1320 IF bk% > 255 GOTO 1300
1330 INPUT " HGR graphics window value? ";gw$: PRINT: PRINT
1340 gw% = VAL(gw$): IF gw% < Ø GOTO 700
1350 IF gw% > 255 GOTO 1300
1360 INPUT " HGR text value? ";tv$: PRINT: PRINT
1370 tv% = VAL(tv$): IF tv% < Ø GOTO 700
1380 IF tv% > 255 GOTO 1300
1400 GOSUB 10100: rad = 25431: GOSUB 11000
1410 POKE 112*256+of,bk%: POKE c2,dv%: GOSUB 10300
1420 rad = 25471: GOSUB 11000: POKE 112*256+of,gw%: GOSUB 10300
1430 rad = 25568: GOSUB 11000: POKE 112*256+of,tv%: GOSUB 10300
1440 GOTO 1000
1500 HOME: INPUT " GR background value? ";bk$: PRINT: PRINT
1510 bk% = VAL(bk$): IF bk% < Ø GOTO 700
1520 IF bk% > 255 GOTO 1500
1530 INPUT " GR graphics window value? ";gw$: PRINT: PRINT
1540 gw% = VAL(gw$): IF gw% < Ø GOTO 700
1550 IF gw% > 255 GOTO 1500
1560 INPUT " GR text value? ";tv$: PRINT: PRINT
1570 tv% = VAL(tv$): IF tv% < Ø GOTO 700
1580 IF tv% > 255 GOTO 1500
1600 GOSUB 10100: rad = 18607: GOSUB 11000
1610 POKE 112*256+of,bk%: POKE c2,dv%: GOSUB 10300
1620 rad = 18633: GOSUB 11000: POKE 112*256+of,gw%: GOSUB 10300
1630 rad = 18711: GOSUB 11000: POKE 112*256+of,tv%: GOSUB 10300
1640 GOTO 1000
1700 HOME: PRINT " Which default drive?": PRINT: PRINT
1710 FOR x = 1 TO 4: PRINT " ";x;" = ";dv$(x): NEXT: PRINT: PRINT
1720 GET k$: k% = VAL(k$): IF k% < 1 OR k% > 4 GOTO 1000
1730 IF k% = 1 THEN  dd% = 8
1740 IF k% = 2 THEN  dd% = 24
1750 IF k% = 3 THEN  dd% = 4
1760 IF k% = 4 THEN  dd% = 5
1800 GOSUB 10100: rad = 16641: GOSUB 11000
1810 POKE 112*256+of,dd%: POKE c2,dv%: GOSUB 10300
1820 GOTO 1000
```

EZpatch 1.Ø LIST continued ...

```
 2ØØØ HOME: PRINT " Change which BASIC address"
 2Ø1Ø INPUT " (256 - 28671): ";ch$: ch% = VAL(ch$)
 2Ø2Ø IF ch% < Ø GOTO 7ØØ
 2Ø3Ø IF ch% < 256 OR ch% > 28671 GOTO 2Ø1Ø
 2Ø4Ø HOME: GOSUB 1Ø1ØØ
 2Ø5Ø rad = ch%: GOSUB 11ØØØ
 21ØØ HOME: PRINT: PRINT " PEEK(";rad;") = ";pk
 22ØØ VTAB 1Ø: PRINT " Do you want to change the"
 2211 PRINT " value ('y' for yes): ";: GET k$
 2221 IF k$ <> "y" AND k$ <> "Y" GOTO 2ØØØ
 23ØØ VTAB 1Ø: HTAB 2: PRINT "Enter the new value for"
 2311 PRINT " address ";rad;":"
 2321 VTAB 11: HTAB 16: INPUT " ";nn$: nn% = VAL(nn$)
 2331 IF nn% < Ø GOTO 7ØØ
 2341 IF nn% > 255 GOTO 23ØØ
 2351 POKE 112*256+of,nn%: POKE c2,dv%
 2361 POKE c3,x2: VTAB 2Ø: GOSUB 1Ø1ØØ: CALL c1
 2371 IF PEEK(65535) = Ø GOTO 2ØØØ
 2381 HOME: PRINT " write error on block ";x2;"!": END
 4ØØØ HOME: PRINT " What address for scan start"
 4Ø1Ø INPUT " (256 - 28671): ";st$: st% = VAL(st$)
 4Ø2Ø IF st% < Ø GOTO 7ØØ
 4Ø3Ø IF st% < 256 OR st% > 28671 GOTO 4Ø1Ø
 4Ø4Ø HOME: GOSUB 1Ø1ØØ
 4Ø5Ø FOR x = st% TO 28671: rad = x: GOSUB 11ØØØ
 41ØØ st$ = STR$(x): pk$ = STR$(pk): ft$ = CHR$(31)
 411Ø IF pk > 31 AND pk < 127 THEN  ft$ = CHR$(pk)
 412Ø PRINT SPC(6-LEN(st$));st$;SPC(5-LEN(pk$));pk$;"  ";ft$
 413Ø rf = PEEK(64885): IF rf <> 27 GOTO 415Ø
 414Ø PRINT: PRINT: GOSUB 1Ø2ØØ: GOTO 7ØØ
 415Ø IF rf <> 144 AND rf <> 152 GOTO 417Ø
 416Ø POKE 16136,Ø: GOTO 43ØØ
 417Ø IF rf <> 128 GOTO 419Ø
 418Ø x = st%-1: PRINT: PRINT: GOTO 43ØØ
 419Ø IF rf <> 16Ø GOTO 422Ø
 42ØØ x = x-11: IF x < st%-1 THEN  x = st%-1
 421Ø PRINT: PRINT: GOTO 43ØØ
 422Ø IF rf <> 162 GOTO 43ØØ
 423Ø x = x+9: IF x > 28671 THEN  x = 28671
 424Ø PRINT: PRINT
 43ØØ POKE 64885,Ø: NEXT x: GOTO 414Ø
 5ØØØ HOME: PRINT " These codes apply when using"
 5Ø1Ø PRINT " the routine editor:": PRINT: PRINT
 5Ø2Ø PRINT " B = back up one address"
 5Ø3Ø PRINT " E = enter value at address"
 5Ø4Ø PRINT " Q = quit editing"
 5Ø5Ø PRINT " R = review routine/addresses"
 5Ø6Ø PRINT " S = skip one address": PRINT: PRINT: PRINT
 5Ø7Ø PRINT " Start editing at what"
 5Ø8Ø INPUT " address (256 - 28671)? ";ea$: ea% = VAL(ea$)
 5Ø9Ø IF ea% < Ø GOTO 7ØØ
```

EZpatch 1.Ø LIST continued ...

```
 5100 IF ea% < 256 OR ea% > 28671 GOTO 5080
 5200 HOME: GOSUB 10100: FOR x = Ø TO mx%-1: z1(x) = ea%+x: xx = x
 5210 rad = z1(x): GOSUB 11000: IF x = Ø THEN  HOME
 5215 PRINT " address = ";z1(x)
 5220 PRINT " value   = ";pk
 5230 PRINT " (B, E, Q, R, or S)? ";
 5240 GET k$: IF k$ = "B" OR k$ = "b" GOTO 5500
 5250 IF k$ = "Q" OR k$ = "q" GOTO 5330
 5260 IF k$ = "R" OR k$ = "r" GOTO 6500
 5270 IF k$ = "S" OR k$ = "s" GOTO 7000
 5280 IF k$ <> "E" AND k$ <> "e" GOTO 5240
 5300 PRINT: PRINT " POKE ";z1(x);",";: INPUT " ";pe$
 5302 IF pe$ = "" GOTO 5300
 5304 IF LEFT$(pe$,1) < "Ø" OR LEFT$(pe$,1) > "9" GOTO 5300
 5310 z2(x) = INT(VAL(pe$)): IF z2(x) < Ø OR z2(x) > 255 GOTO 5300
 5320 PRINT: NEXT x
 5330 PRINT: PRINT: PRINT: IF x = Ø GOTO 700
 5340 PRINT: PRINT " 1 = abort BASIC changes"
 5350 PRINT " 2 = enter this routine"
 5360 IF x = mx% GOTO 5380
 5370 PRINT " 3 = continue editing"
 5380 GET k$: k% = VAL(k$): IF k% < 1 OR k% > 3 GOTO 10000
 5390 IF x = mx% AND k% = 3 GOTO 10000
 5392 IF k% = 1 THEN  RUN
 5394 IF k% = 3 THEN  x = xx-1: GOTO 5320
 5400 HOME: PRINT " entering this routine,": GOSUB 10100
 5410 FOR z = Ø TO xx-1: rad = z1(z): GOSUB 11000: POKE c2,dv%
 5440 POKE 112*256+of,z2(z): GOSUB 10300: NEXT z: RUN
 5500 x = xx-2: IF x < -1 THEN  x = -1
 5510 PRINT: GOTO 5320
 6500 PRINT: PRINT: IF xx > 1 GOTO 6520
 6510 PRINT " nothing to review": x = xx-1: PRINT: PRINT: GOTO 5320
 6520 PRINT: PRINT: FOR y = Ø TO xx-1
 6530 z1$ = STR$(z1(y)): z2$ = STR$(z2(y)): ft$ = CHR$(31)
 6540 IF z2(y) > 31 AND z2(y) < 127 THEN  ft$ = (CHR$(z2(y)))
 6550 PRINT SPC(6-LEN(z1$));z1$;SPC(5-LEN(z2$));z2$;"  ";ft$
 6600 rf = PEEK(64885): IF rf <> 27 GOTO 6620
 6610 PRINT: PRINT: x = xx-1: GOTO 5320
 6620 IF rf <> 144 AND rf <> 152 GOTO 6700
 6630 POKE 16136,Ø
 6700 POKE 64885,Ø: NEXT y: x = xx-1: PRINT: PRINT: GOTO 5320
 7000 PRINT: GOTO 5320
10000 HOME: PRINT " program terminated."
10010 POKE 64947,Ø: POKE 64936,Ø: END
10100 PRINT " one moment please ...": RETURN
10200 PRINT " press any key to continue ..."
10210 GET k$: RETURN
10300 POKE c3,x2: CALL c1: ON PEEK(65535) <> Ø GOTO 2380: RETURN
11000 x1 = INT((rad-256)/1024): x2 = x1+fb%
11010 ON PEEK(b3) = x2 GOTO 11040: POKE b2,dv%: POKE b3,x2: CALL b1
11020 IF PEEK(65535) = 128 GOTO 11040
11030 HOME: PRINT " read error on block ";x2: END
11040 y1 = INT(rad/256): y2 = rad-y1*256
11050 y3 = (y1-1)-x1*4: of = y3*256+y2
11060 pk = PEEK(of+112*256): RETURN
```

```
 1Ø REM demo program to determine drive status
1ØØ LOMEM :28ØØØ
2ØØ DATA 62,4,2Ø5,126,252,4Ø,5,5Ø,252,255,24,21
21Ø DATA 62,4,2Ø5,84,252,253,126,2Ø,23Ø,15,254,3,62,4,56,2
22Ø DATA 62,255,5Ø,252,255
23Ø DATA 62,5,2Ø5,126,252,4Ø,5,5Ø,253,255,24,21
24Ø DATA 62,5,2Ø5,84,252,253,126,2Ø,23Ø,15,254,3,62,5,56,2
25Ø DATA 62,255,5Ø,253,255
26Ø DATA 62,8,2Ø5,126,252
27Ø DATA 62,8,2Ø5,84,252,253,126,2Ø,23Ø,15,254,3,62,8,56,2
28Ø DATA 62,255,5Ø,254,255
29Ø DATA 253,126,2Ø,254,48,48,4,62,24,24,2,62,255
3ØØ DATA 5Ø,255,255,2Ø1,-1
31Ø start = 276ØØ
32Ø READ ml: IF ml = -1 GOTO 35Ø
33Ø POKE start,ml: start = start+1
34Ø total = total+ml: GOTO 32Ø
35Ø ON start = 27709 GOTO 4ØØ: wr = start-27709
36Ø IF wr > Ø THEN  PRINT " ";wr;" entries too many!!": END
37Ø PRINT " missing ";(ABS(wr));" entries!!": END
4ØØ IF total = 12721 GOTO 5ØØ
41Ø PRINT " incorrect data total!!!"
42Ø PRINT " offset = ";total-12721: END
5ØØ TEXT: INVERSE: PRINT " DRIVE STATUS DEMO": NORMAL
51Ø VTAB 4: INVERSE: PRINT " disk one": VTAB 6: PRINT " disk two"
52Ø VTAB 8: PRINT " tape one": VTAB 1Ø: PRINT " tape two": NORMAL
53Ø GOSUB 1ØØØ: VTAB 16: PRINT " 1 = scan drive status"
54Ø VTAB 18: PRINT " 2 = exit program"
55Ø GET key$: IF key$ = "1" THEN  GOSUB 1ØØØ: GOTO 55Ø
56Ø TEXT: PRINT " program terminated.": END
999 END
1ØØØ a1$ = "no such drive": a2$ = "drive power off"
1Ø1Ø a3$ = "NO medium in drive": a4$ = "medium IN drive"
1Ø2Ø CALL 276ØØ: id = 65532: GOSUB 15ØØ: VTAB 4: HTAB 11: PRINT st$
1Ø3Ø id = 65533: GOSUB 15ØØ: VTAB 6: HTAB 11: PRINT st$
1Ø4Ø id = 65534: GOSUB 15ØØ: VTAB 8: HTAB 11: PRINT st$
1Ø5Ø id = 65535: GOSUB 15ØØ: VTAB 1Ø: HTAB 11: PRINT st$
1Ø6Ø RETURN
15ØØ IF PEEK(id) = 1 THEN  st$ = a1$: RETURN
151Ø IF PEEK(id) = 155 THEN  st$ = a2$: RETURN
152Ø IF PEEK(id) = 255 THEN  st$ = a3$: RETURN
153Ø st$ = a4$: RETURN
```

```
 10 REM EZmenu
 20 REM by DIGITAL EXPRESS
 30 REM All graphics!!!
 40 REM simple file control utility
 50 REM works with BASIC 1.Ø, Intel-BEST 3.3, and RAMDSK
 60 REM also works with BASIC 2.Ø (extmem and stdmem)
100 LOMEM :32768: ONERR  GOTO 61000: IF PEEK(259) = 195 GOTO 115
110 cj% = 17240: POKE 1648,255: POKE 1649,255: GOTO 120
115 POKE 16149,255: POKE 16150,255: cj% = 17115
120 TEXT: PRINT " one moment please ...";: DIM pn$(40)
125 cc% = PEEK(cj%): IF cc% = 240 THEN  cc% = 27
130 DATA 17,Ø,Ø,33,Ø,108,1,Ø,4,205,29,253,201
140 FOR x = 28672 TO 28684: READ mc: POKE x,mc: NEXT: CALL 28672
150 DATA 33,Ø,108,17,Ø,212,1,Ø,4,126,18,19,18,19
160 DATA 35,11,120,177,32,245,201
170 FOR x = 28685 TO 28705: READ mc: POKE x,mc: NEXT: CALL 28685
180 DATA Ø,16,16,16,16,16,16,Ø,Ø,68,68,68,68,68,68,Ø
190 DATA Ø,146,146,146,146,146,146,Ø,Ø,145,145,138,138,132,132,Ø
200 DATA Ø,34,34,20,20,8,8,Ø,Ø,137,137,81,81,33,33,Ø
210 FOR x = 27656 TO 27703: READ mc: POKE x,mc: NEXT
212 DATA Ø,4,2,127,2,4,Ø,Ø
214 FOR x = 27864 TO 27871: READ mc: POKE x,mc: NEXT
220 DATA 237,91,242,255,26,254,Ø,200,254,13,200
230 DATA 245,58,239,255,71,241,33,Ø,Ø,95,22,Ø,25,16,253
235 DATA 237,75,248,255,9,58,246,255
240 DATA 71,58,244,255,79,175,129,16,253
245 DATA 95,58,245,255,61,198,32,87,237,75,246,255
250 DATA 213,205,26,253,209,107,122,214,32,103,58,241,255
260 DATA 237,91,246,255,205,38,253,58,244,255,60,50,244,255
270 DATA 42,242,255,35,34,242,255,24,165
280 FOR x = 28706 TO 28796: READ mc: POKE x,mc: NEXT
290 FOR x = 65517 TO 65535: POKE x,Ø: NEXT: POKE 65526,8
300 IF PEEK(259) = 195 THEN  POKE 25431,cc%: POKE 25471,cc%: GOTO 304
302 POKE 24695,cc%: POKE 24847,cc%
304 HGR2: pt = 28706
310 DATA 62,187,17,Ø,20,33,Ø,Ø,213,205,38,253,209
320 DATA 175,33,Ø,32,205,38,253,201
330 FOR x = 28797 TO 28817: READ mc: POKE x,mc: NEXT: c1 = 28797
335 POKE c1+1,cc%
340 DATA 62,153,17,248,Ø,33,8,20,213,205,38,253,209
350 DATA 175,33,8,52,205,38,253,201
360 FOR x = 28818 TO 28838: READ mc: POKE x,mc: NEXT: c2 = 28818
370 DATA 62,119,17,248,Ø,33,8,21,245,213,229,205,38,253
380 DATA 225,209,241,36,245,213,229,205,38,253,225,209,241
390 DATA 36,213,205,38,253,209,33,8,53,175,213,229,205,38,253
400 DATA 225,209,36,175,213,229,205,38,253,225,209,36,175,195,38,253
410 FOR x = 28839 TO 28896: READ mc: POKE x,mc: NEXT: c3 = 28839
420 DATA 62,4,17,40,Ø,33,Ø,21,229,213,245,205,38,253
430 DATA 241,209,225,36,229,213,245,205,38,253
440 DATA 241,209,225,36,195,38,253
450 FOR x = 28897 TO 28927: READ mc: POKE x,mc: NEXT
460 d1 = 28897: d2 = d1+1: d3 = d1+6
470 FOR x = 4 TO 6: READ m1$(x): NEXT: DATA hue,dir,exit
475 DATA 6,20,62,128,211,224,120,211,224,62,146,211,224
480 DATA 17,Ø,10,27,122,179,32,251,5,16,234,62,159,211,224,201
485 FOR x = 28928 TO 28956: READ mc: POKE x,mc: NEXT
490 s1 = 28928: s2 = s1+1: s3 = s1+15
495 DATA 62,226,211,224,62,240,211,224,17,Ø,175,27,122,179
```

EZmenu LIST continued ...

```
  500 DATA 32,251,62,255,211,224,201
  510 FOR x = 28957 TO 28977: READ mc: POKE x,mc: NEXT
  520 e1 = 28957: e2 = e1+1: e3 = e1+10
  530 FOR x = 1 TO 6: READ m2$(x): NEXT
  535 DATA yelw,blue,red,grn,cyan,done
  540 IF PEEK(259) = 195 THEN  cx = 18765: GOTO 544
  542 cx = 25360
  544 FOR x = 0 TO 15: POKE x+cx,x: NEXT
  550 dr% = PEEK(16821): IF PEEK(259) <> 195 THEN  dr% = PEEK(16781)
  560 FOR x = 4 TO 6: READ m3$(x): NEXT: DATA drv,dir,done
  590 FOR x = 1 TO 6: READ m4$(x): NEXT: POKE 65530,dr%
  600 DATA next,get,ren,init,vol,done
  610 FOR x = 5 TO 6: READ yn$(x): NEXT: DATA no,yes
  620 DATA 62,4,205,126,252,40,5,50,252,255,24,21
  630 DATA 62,4,205,84,252,253,126,20,230,15,254,3,62,4,56,2
  640 DATA 62,255,50,252,255
  650 DATA 62,5,205,126,252,40,5,50,253,255,24,21
  660 DATA 62,5,205,84,252,253,126,20,230,15,254,3,62,5,56,2
  670 DATA 62,255,50,253,255
  680 DATA 62,8,205,126,252
  690 DATA 62,8,205,84,252,253,126,20,230,15,254,3,62,8,56,2
  700 DATA 62,255,50,254,255
  710 DATA 253,126,20,254,48,48,4,62,24,24,2,62,255
  720 DATA 50,255,255,201
  730 FOR x = 28978 TO 29086: READ mc: POKE x,mc: NEXT: dv = 28978
  740 FOR x = 5 TO 6: READ dv$(x): NEXT: DATA scan,done
  750 DATA 62,0,1,0,0,17,0,0,33,0,0,205,243,252,50,255,107,201
  760 FOR x = 29087 TO 29104: READ mc: POKE x,mc: NEXT
  770 r1 = 29087: r2 = r1+1: r3 = r1+6: r4 = r1+10
  780 DATA 62,0,1,0,0,17,0,0,33,0,0,205,246,252,50,255,107,201
  790 FOR x = 29105 TO 29122: READ mc: POKE x,mc: NEXT
  800 w1 = 29105: w2 = w1+1: w3 = w1+6: w4 = w1+10
  810 DATA 1,0,7,205,32,253,201
  820 FOR x = 29123 TO 29129: READ mc: POKE x,mc: NEXT: c4 = 29123
 1000 GOSUB 30100: ww$ = "Welcome to": b1% = 0: b2% = 212: di% = 16
 1010 vt% = 2: ht% = 11: co% = cc%: GOSUB 30000
 1020 vt% = 3: b1% = 8: GOSUB 30000
 1030 ww$ = "EZmenu": b1% = 0: ht% = 13: vt% = 5: GOSUB 30000
 1040 vt% = 6: b1% = 8: GOSUB 30000
 1050 ww$ = "A simple file control"
 1060 b1% = 0: b2% = 108: vt% = 9: ht% = 6: di% = 8: GOSUB 30000
 1070 ww$ = "utility developed by": vt% = 11: ht% = 6: GOSUB 30000
 1080 ww$ = "DIGITAL EXPRESS.": vt% = 13: ht% = 6: GOSUB 30000
 1100 ww$ = "using BASIC "
 1110 IF PEEK(259) = 195 THEN  wa$ = "V1.0": GOTO 1200
 1120 wa$ = "2.0 "
 1130 IF PEEK(16789) = 0 THEN  wa$ = wa$+"(stdmem)": GOTO 1200
 1140 wa$ = wa$+"(extmem)"
 1200 ww$ = ww$+wa$+" ...": GOSUB 40000: be% = 4
 1210 FOR z = be% TO 6: wd$ = m1$(z): GOSUB 40100: NEXT
 1220 GOSUB 31000: ON sd% GOTO 2000,5000,60000
```

EZmenu LIST continued ...

```
2000 ww$ = "select screen color ...": GOSUB 40000: be% = 1
2010 FOR z = be% TO 6: wd$ = m2$(z): GOSUB 40100: NEXT
2020 GOSUB 31000: IF sk% = 6 GOTO 1100
2030 IF sk% = 1 THEN  cc% = 27
2040 IF sk% = 2 THEN  cc% = 244
2050 IF sk% = 3 THEN  cc% = 246
2060 IF sk% = 4 THEN  cc% = 19
2070 IF sk% = 5 THEN  cc% = 23
2100 POKE c4+1,cc%: CALL c4
2110 POKE c1+1,cc%: POKE c1+13,201: CALL c1: POKE c1+13,175
2200 IF PEEK(259) = 195 GOTO 2250
2210 z(1) = 17184: z(2) = 17240: z(3) = 24695: z(4) = 24847
2220 z(5) = 24695: z(6) = 24847: GOTO 2300
2250 z(1) = 17059: z(2) = 17115: z(3) = 18607: z(4) = 18711
2260 z(5) = 25431: z(6) = 25568
2300 FOR x = 1 TO 6: POKE z(x),cc%: NEXT: GOTO 2020
5000 dd% = PEEK(65530): GOSUB 32000: CALL c1
5010 ww$ = "current drive: "+ww$: GOSUB 40000: be% = 4
5020 FOR z = be% TO 6: wd$ = m3$(z): GOSUB 40100: NEXT
5030 GOSUB 31000: ON sd% GOTO 6000,7000,1000
6000 w1$ = "select a base drive ...": GOSUB 34000
6010 IF dd% = 0 GOTO 5000
6020 POKE 65530,dd%: GOTO 5000
7000 TEXT: HGR2: pn$(1) = "": dn$ = "": CALL c1
7005 ww$ = "getting directory ...": GOSUB 40000
7010 POKE r2,PEEK(65530): POKE r3,1: POKE r4,116: CALL r1
7020 IF PEEK(27647) <> 128 GOTO 35000
7030 IF PEEK(29709) <> 85 GOTO 35000
7040 IF PEEK(29710) <> 170 GOTO 35000
7050 bz% = PEEK(29708): IF bz% > 128 THEN  bz% = bz%-128
7060 IF bz% > 3 GOTO 35000
7070 IF bz% = 1 THEN  GOTO 7500
7100 POKE r3,2: POKE r4,120: CALL r1
7200 IF bz% = 2 GOTO 7500
7300 POKE r3,3: POKE r4,124: CALL r1
7500 POKE pt+6,0: pg% = 1: st% = 29696
7510 dd% = PEEK(65530): GOSUB 32000: ww$ = ww$+" name:"
7520 GOSUB 40000: POKE pt+6,3: POKE 29707,3
7530 ht% = 18: GOSUB 30020: fc% = 0: st% = st%+52: GOSUB 37000
7600 st% = st%+26: IF PEEK(st%+12) = 1 GOTO 7800
7610 IF PEEK(st%+12) = 4 OR PEEK(st%+12) = 20 GOTO 7600
7612 IF fc% > 35 AND pg% = 1 GOTO 7800
7614 IF fc% > 39 AND pg% > 1 GOTO 7800
7620 fc% = fc%+1: pn$(fc%) = "": FOR x = 1 TO 11
7630 IF PEEK(x+st%-1) = 3 GOTO 7700
7640 pn$(fc%) = pn$(fc%)+CHR$(PEEK(x+st%-1)): NEXT
7700 ht% = 3: IF INT(fc%/2) = fc%/2 THEN  ht% = 19
7710 vt% = INT(fc%/2+.5): co% = cc%: GOSUB 30020
7720 GOTO 7600
```

EZmenu LIST continued ...

```
7800 IF fc% < 35 AND pg% = 1 THEN  dn$ = "end"
7805 IF fc% < 39 AND pg% > 1 THEN  dn$ = "end"
7810 IF pn$(1) <> "" GOTO 8000
7820 IF pg% <> 1 GOTO 8000
7830 POKE pt+6,0: ww$ = "NO FILES": vt% = 3: ht% = 3: co% = cc%
7840 GOSUB 30000: ww$ = "ON THIS": vt% = 4: GOSUB 30000
7850 IF pg% > 1 THEN  ww$ = "BLOCK!!!": vt% = 5: GOSUB 30000: GOTO 8000
7860 ww$ = "MEDIUM": vt% = 5: GOSUB 30000
8000 POKE pt+6,0: be% = 2: GOSUB 38100
8002 IF bz% > 1 AND dn$ <> "end" THEN  be% = 1
8004 IF pn$(1) = "" THEN  be% = 4
8006 IF pg% > 1 AND dn$ = "end" THEN  be% = 1
8010 FOR z = be% TO 6: wd$ = m4$(z): GOSUB 40100: NEXT
8015 ar$ = "yes": IF pn$(1) = "" THEN  ar$ = "no": GOTO 8025
8020 ht% = 2: vt% = 1: co% = 31: ww$ = CHR$(27): GOSUB 30000
8025 GOSUB 31000: IF sk% >= 160 AND sk% <= 163 GOTO 8500
8030 ON be% = 1 GOTO 8050: ON be% = 4 GOTO 8060
8040 ON sd% GOTO 11000,12000,9000,10000,5000
8050 ON sd% GOTO 15000,11000,12000,9000,10000,5000
8060 ON sd% GOTO 9000,10000,5000
8500 ho = ht%: vo = vt%: IF fc% = 1 THEN  GOSUB 30600: GOTO 8730
8510 IF sk% = 160 THEN  vo = vo-1
8520 IF sk% = 161 THEN  ho = ho+1
8530 IF sk% = 162 THEN  vo = vo+1
8540 IF sk% = 163 THEN  ho = ho-1
8550 IF ht% = 2 AND vo < xu THEN  vo = xd: GOTO 8700
8560 IF ht% = 18 AND vo < yu THEN  vo = yd: GOTO 8700
8570 IF ho = 19 THEN  ho = 2: GOTO 8700
8580 IF ho = 3 AND vo = xd AND xd > yd THEN  vo = yd: ho = 18: GOTO 8700
8590 IF ho = 3 THEN  ho = 18: GOTO 8700
8600 IF ht% = 2 AND vo > xd THEN  vo = xu
8610 IF ht% = 18 AND vo > yd THEN  vo = yu
8620 IF ho = 17 THEN  ho = 2: GOTO 8700
8630 IF ho = 1 AND yu = 0 THEN  GOSUB 30600: GOTO 8730
8640 IF ho = 1 THEN  ho = 18: GOTO 8700
8700 ww$ = CHR$(32): co% = cc%: GOSUB 30000
8710 ht% = ho: vt% = vo: ww$ = CHR$(27): co% = 31: GOSUB 30000
8720 GOSUB 30500
8730 ar$ = "yes": GOTO 8025
9000 ww$ = "enter INIT name:": GOSUB 40000: GOSUB 32300
9010 vt = 21: ht = 19: ml = 1: ll = 10: lf$ = "!": hf$ = "z": GOSUB 55000
9020 nn$ = b$: ww$ = "directory length?": GOSUB 40000: GOSUB 32300
9030 vt = 21: ht = 22: ml = 1: ll = 1: lf$ = "1": hf$ = "3": GOSUB 55000
9040 dn% = VAL(b$): ww$ = "volume length?": GOSUB 40000: GOSUB 32300
9050 vt = 21: ht = 19: ml = 1: ll = 3: lf$ = "0": hf$ = "9": GOSUB 55000
9060 vn% = VAL(b$): IF vn% > 255 THEN  GOSUB 30600: GOTO 9050
9065 POKE 25305,vn%: POKE 25308,dn%
9070 dd% = PEEK(65530): GOSUB 32000
9080 ww$ = "INITializing "+ww$+" ...": GOSUB 40000
9090 PRINT CHR$(4);"INIT ";nn$: GOTO 7000
```

EZmenu LIST continued ...

```
10000 ww$ = "enter VOLume name:": GOSUB 40000: GOSUB 32300
10010 vt = 21: ht = 21: ml = 1: ll = 11: lf$ = " ": hf$ = "z": GOSUB 55000
10020 FOR x = 1 TO LEN(b$)
10030 POKE x+29695,ASC(MID$(b$,x,1)): NEXT
10040 POKE x+29695,3
10050 POKE w2,PEEK(65530): POKE w3,1: POKE w4,116
10060 ww$ = "changing volume name ...": GOSUB 40000
10070 CALL w1: GOTO 7000
11000 GOSUB 39000: GOSUB 39100
11010 rr$ = "RUN ": IF ft$ = "H" THEN  rr$ = "BRUN "
11020 TEXT: PRINT " getting ";pn$;" ...";
11030 dr% = PEEK(65530): IF PEEK(259) = 195 THEN  POKE 16821,dr%
11040 IF PEEK(259) = 210 THEN  POKE 16781,dr%
11050 PRINT CHR$(4);rr$;pn$
12000 GOSUB 39000
12020 co% = 31: ww$ = pn$: POKE pt+6,0: ht% = ht%+1: GOSUB 30000
12030 ww$ = "enter NEW name:": GOSUB 40000: GOSUB 32300
12040 vt = 21: ht = 22: ml = 1: ll = 10: lf$ = "!": hf$ = "z": GOSUB 55000
12050 nn$ = b$: ww$ = "RENAMEing file ...": GOSUB 40000
12060 GOSUB 39100
12070 dr% = PEEK(65530): IF PEEK(259) = 195 THEN  POKE 16821,dr%
12075 IF PEEK(259) = 210 THEN  POKE 16781,dr%
12080 PRINT CHR$(4);"rename ";pn$;",#"
12090 PRINT CHR$(4);"rename #, ";nn$: GOTO 7000
15000 fc% = 0: CALL c3: pg% = pg%+1: IF pg% > 3 GOTO 7500
15010 IF pg% > bz% GOTO 7500
15020 IF pg% > 1 AND dn$ = "end" THEN  dn$ = "": GOTO 7500
15100 st% = (pg%+28)*1024-26: POKE pt+6,3: GOSUB 37000
15110 ar$ = "yes": vt% = 1: ht% = 2: GOTO 7600
30000 FOR x = 1 TO LEN(ww$): POKE x+27599,ASC(MID$(ww$,x,1))
30010 NEXT: POKE 27599+x,0
30020 POKE 65524,ht%: POKE 65519,di%: POKE 65521,co%
30030 POKE 65528,b1%: POKE 65529,b2%: POKE 65525,vt%
30040 IF PEEK(pt+6) = 0 GOTO 30060
30050 GOSUB 36000: CALL pt: RETURN
30060 POKE 65522,208: POKE 65523,107: CALL pt: RETURN
30100 CALL c1: CALL c2: CALL c3: HCOLOR  = 1: HPLOT 7,0 TO 255,0
30110 HPLOT 7,158 TO 255,158: HPLOT 7,0 TO 7,158
30120 HPLOT 255,0 TO 255,158: RETURN
30300 POKE s2,20: POKE s3,10: CALL s1: RETURN
30400 POKE s2,8: POKE s3,25: CALL s1: RETURN
30500 POKE s2,2: POKE s3,30: CALL s1: RETURN
30600 POKE e2,226: POKE e3,150: CALL e1: RETURN
30700 POKE e2,228: POKE e3,20: CALL e1: RETURN
30800 POP: GOSUB 30300: CALL c2: CALL c3: GOTO 1000
31000 GET sk$: sk% = ASC(sk$)
31010 GOSUB 31100
31020 IF sk% > 134 THEN  GOSUB 30600: GOTO 31000
31030 IF sk% = 27 GOTO 30800
31040 IF sk% < be%+128 THEN  GOSUB 30600: GOTO 31000
31050 sk% = sk%-128: sd% = sk%-be%+1: GOTO 30300
31100 IF sk% <= 134 THEN  RETURN
31110 IF ar$ = "yes" GOTO 31200
31120 sk% = sk%-8: RETURN
31200 IF sk% >= 160 AND sk% <= 163 THEN  POP: ar$ = "no": RETURN
31210 RETURN
```

EZmenu LIST continued ...

```
32000 IF dd% = 4 THEN  ww$ = "disk one"
32010 IF dd% = 5 THEN  ww$ = "disk two"
32020 IF dd% = 8 THEN  ww$ = "tape one"
32030 IF dd% = 24 THEN  ww$ = "tape two"
32040 RETURN
32100 ww$ = STR$(bk%): IF bk% > 99 THEN  RETURN
32110 IF bk% > 9 THEN  ww$ = " "+ww$: RETURN
32120 ww$ = " "+" "+ww$: RETURN
32300 ww$ = "press [RETURN] after typing..."
32310 ht% = 2: vt% = 23: co% = 23: GOTO 30000
33000 CALL c2: CALL c3: CALL dv: ee = 0
33010 IF PEEK(65534) <> 8 THEN  ee = ee+1: GOTO 33030
33020 z = 1: wd$ = "ddp1": GOSUB 40100
33030 IF PEEK(65535) <> 24 THEN  ee = ee+1: GOTO 33050
33040 z = 2: wd$ = "ddp2": GOSUB 40100
33050 IF PEEK(65532) <> 4 THEN  ee = ee+1: GOTO 33070
33060 z = 3: wd$ = "dsk1": GOSUB 40100
33070 IF PEEK(65533) <> 5 THEN  ee = ee+1: GOTO 33090
33080 z = 4: wd$ = "dsk2": GOSUB 40100
33090 RETURN
34000 GOSUB 33000: dd% = 0: IF ee <> 4 GOTO 34100
34010 ww$ = "all drives empty ..."
34100 ON ee = 4 GOTO 34110: ww$ = w1$
34110 vt% = 21: ht% = 2: co% = 25: GOSUB 30000
34120 FOR z = 5 TO 6: wd$ = dv$(z): GOSUB 40100: NEXT
34130 be% = 1: IF ee = 4 THEN  be% = 5
34140 GOSUB 31000: IF sk% = 5 GOTO 34000
34150 IF sk% = 6 THEN  RETURN
34160 IF sk% = 1 AND PEEK(65534) <> 8 THEN  GOSUB 30600: GOTO 34130
34170 IF sk% = 2 AND PEEK(65535) <> 24 THEN  GOSUB 30600: GOTO 34130
34180 IF sk% = 3 AND PEEK(65532) <> 4 THEN  GOSUB 30600: GOTO 34130
34190 IF sk% = 4 AND PEEK(65533) <> 5 THEN  GOSUB 30600: GOTO 34130
34200 IF sk% = 1 THEN  dd% = 8: RETURN
34210 IF sk% = 2 THEN  dd% = 24: RETURN
34220 IF sk% = 3 THEN  dd% = 4: RETURN
34230 dd% = 5: RETURN
35000 ww$ = "can not access directory!!!"
35010 GOSUB 40000: vt% = 23: ht% = 2: co% = 23
35020 ww$ = "press any key for menu...": GOSUB 30000: GOSUB 30600
35030 GET key$: GOSUB 30300: GOTO 5000
36000 POKE 65523,st%/256: POKE 65522,st%-PEEK(65523)*256: RETURN
37000 CALL c1: HCOLOR  = 1: HPLOT 8,0 TO 8,158
37010 HPLOT 124,0 TO 124,158: HPLOT 132,0 TO 132,158
37020 HPLOT 255,0 TO 255,158: RETURN
```

EZmenu LIST continued ...

```
38100 xu = 1: xd = INT(fc%/2+.5)
38110 yu = 1: IF fc% <= 1 THEN  yu = Ø
38120 yd = xd-1: IF INT(fc%/2) = fc%/2 THEN  yd = xd
38130 RETURN
39000 ww$ = CHR$(32): co% = cc%: GOSUB 30000
39010 ff = vt%*2: IF ht% = 2 THEN  ff = ff-1
39020 pn$ = pn$(ff): RETURN
39100 pn% = LEN(pn$): ft$ = RIGHT$(pn$,1)
39110 pn$ = LEFT$(pn$,pn%-1): RETURN
40000 CALL c2: CALL c3: ht% = 2: vt% = 21: co% = 25
40010 b1% = Ø: b2% = 10B: di% = 8: GOTO 30000
40100 z$ = CHR$(z): POKE d2,5: IF INT(z/2) = z/2 THEN  POKE d2,4
40110 POKE d3,((z-1)*5+2)*8: CALL d1: ww$ = " "+z$+" "
40120 ht% = (z-1)*5+3: vt% = 22: co% = 31: GOSUB 30000
40130 co% = 21: IF PEEK(d2) = 4 THEN  co% = 244
40140 ww$ = wd$: vt% = 24: GOSUB 30000
40150 ON z = 6 GOTO 30400: RETURN
55000 ww$ = "": FOR x = 1 TO 11: ww$ = ww$+CHR$(95): NEXT: b$ = ""
55010 ht% = ht: vt% = vt: co% = 25: GOSUB 30000
55020 GET a$: a% = ASC(a$)
55030 IF a% = 27 GOTO 30800
55060 IF a$ = CHR$(13) AND LEN(b$) >= m1 GOTO 55180
55070 IF a$ >= lf$ AND a$ <= hf$ GOTO 55120
55080 IF a$ <> CHR$(8) AND a$ <> CHR$(163) THEN  GOSUB 30600: GOTO 55020
55090 IF b$ = "" THEN  GOSUB 30600: GOTO 55020
55100 IF LEN(b$) = 1 THEN  b$ = "": GOSUB 55190: GOSUB 30700: GOTO 55020
55110 b$ = LEFT$(b$,LEN(b$)-1): GOSUB 55190: GOSUB 30700: GOTO 55020
55120 IF LEN(b$) < 11 GOTO 55160
55150 GOSUB 30600: GOTO 55020
55160 ww$ = a$: ht% = ht: vt% = vt: GOSUB 30000
55170 b$ = b$+a$: GOSUB 30500: ht = ht+1: GOTO 55020
55180 GOTO 30300
55190 ww$ = CHR$(32): ht% = ht-1: vt% = vt
55200 ht = ht-1: GOSUB 30000: ww$ = CHR$(95): ht% = ht
55210 vt% = vt: GOTO 30000
60000 TEXT: PRINT " program terminated."
60010 PRINT: PRINT " type NEW to clear RAM": END
61000 er% = ERRNUM(Ø): TEXT: VTAB 2
61010 PRINT " Error number ";er%
61020 PRINT " encountered!!!": PRINT: PRINT
61030 PRINT " program terminated.": END
```

TITLE (asmb#37):

PR#2/PR#3 algorithm

addr:	Label:	Value(s):	Op Code:	Comment:
1102	outCHR	245,	PUSH AF	;save Accum
1103	check	219, 64,	IN A, (64)	;get status byte
1105		203, 71,	BIT 0, A	;check status bit
1107		40, 250,	JR Z, 250	;if not ready then check
1109		241,	POP AF	;retrieve Accum
1110		211, 64,	OUT (64), A	;send byte to printer
1112		201,	RET	;RETurn from routine
1113	entry2	205, 11, 47,	CALL 12043	;CALL BASIC display char
1116	entry3	205, 78, 4,	CALL 1102	;send byte to outCHR
1119	chkCR	254, 13,	CP 13	;check for [return]
1121		192,	RET NZ	;if not then RETurn
1122		62, 10,	LD A, 10	;load line feed ASCII
1124		24, 2,	JR 2	;skip 2 bytes
1126		62, 0,	LD A, nn	;load special value
1128		195, 78, 4	JP 1102	;send byte to outCHR

TITLE (asmb#39):

EOS Programmed Delay

addr:	Label:	Value(s):	Op Code:	Comment:
63839	Store	197,	PUSH BC	;save BC pair
63840		213,	PUSH DE	;save DE pair
63841	Lp1set	6, 1,	LD B, 1	;set up first loop count
63843	Lp2set	17, 1, 0,	LD DE, 1	;set up second loop count
63846	begin	27,	DEC DE	;start loops
63847		122,	LD A, D	;prepare zero check
63848		179,	OR E	;continue zero check
63849		32, 251	JR NZ, 251	;if not zero the 'begin'
63851		16, 246	DJNZ 246	;if B<>0 then 'Lp2set'
63853	done	209,	POP DE	;retrieve DE pair
63854		193,	POP BC	;retrieve BC pair
63855		201	RET	;exit routine

ADAM ENTERTAINMENT SOFTWARE

LOW PRICES QUALITY SOFTWARE GREAT REVIEWS

LAB MOUSE (NEW!) - Lab Mouse puts you in the role of a laboratory mouse stuck inside a maze. You see the maze from the mouse's perspective and you must find the cheese! Utilizes hi-resolution graphics with sprites, offers five skill levels, and every maze is different! Destined to be a top seller.....ONLY $12.95 DDP / $10.95 DISK

MAGEQUEST (TOP SELLER!) - MageQuest is a super graphic adventure by Brian Miguel. In this game you take the role of a mage on a quest to get back the nine Wards of Power that were stolen by the evil Entwon. The graphics are in detailed hi-res and sprites are used for the superbly animated mage and monsters. Have the time of your life exploring the seemingly endless array of rooms. Includes 9 levels plus 3 "Solo Adventures".....ONLY $15.95 DDP / $13.95 DISK

SOLO ADVENTURE PACK VOLUME 2 (for use with MageQuest) - Six more MageQuest adventures.....ONLY $10.95 DDP / $ 8.95 DISK

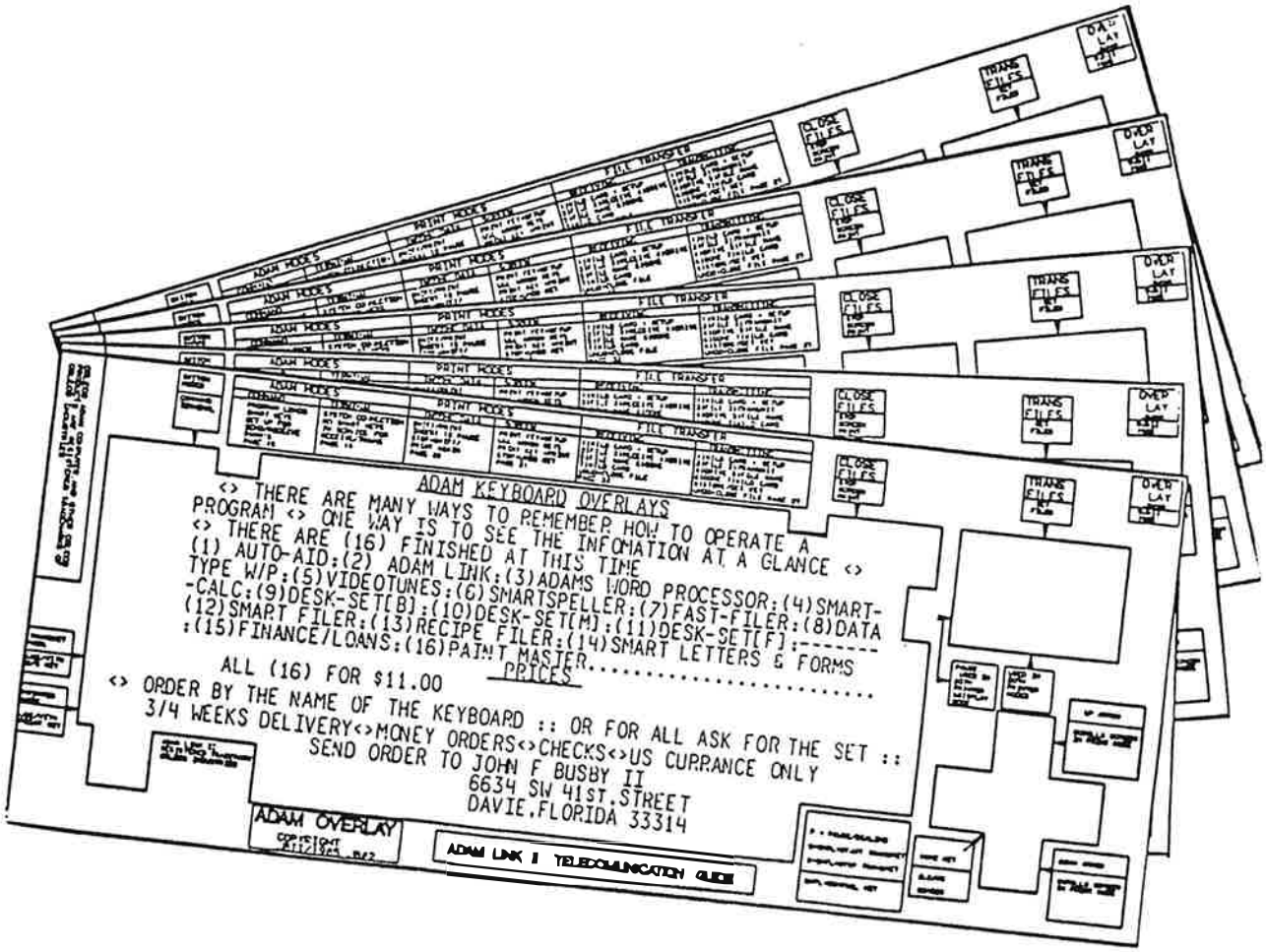
SOLO ADVENTURE PACK VOLUME 3 (for use with MageQuest) - Six more MageQuest adventures.....ONLY $10.95 DDP / $ 8.95 DISK

THE ENTERTAINMENT PACK 1 - The Entertainment Pack is a collection of three super graphic games with arcade sound. Includes: Connect 4, Blockade, and Slide Puzzle. The package is auto-loading and all of the games are loaded from a main menu. This package will provide the whole family with hours of fun!.....ONLY $15.95 DDP / $13.95 DISK

THE REEDY LIBRARY - This excellent package includes several quality programs: Michigana Jones (a great text adventure), VideoPAINT (drawing program), TextEDITOR, Picture Subroutine Maker, and more.....ONLY $19.95 DDP / $17.95 DISK

Send a self-addressed, stamped envelope for our current catalog of all our ADAM products. To order, send check or money order to:

REEDY SOFTWARE
10085 60th Street
Alto, MI 49302



PRODUCT REVIEWS

PRODUCT:	Easy Come - Easy Go
MANUFACTURER:	MMSG
MEDIA TYPE:	datapack/disk
GRAPHICS/SOUND/DESIGN:	97
INSTRUCTIONS:	93
USEFULNESS vs. PRICE:	95
RECOMMENDATION:	highly recommended
PRICE:	19.95
RATED BY:	staff

Quoting from the instructions, "Easy Come - Easy Go is a collection of financially based program modules designed to compute, summarize, and report how a given sum of money, whether borrowed or saved, is affected by interest rates and time. We have designed these modules to be easy to use and we believe they can help you better manage your financial resources."

That's a very accurate description of this financial package. It's very practical and easy to use. The package is self-booting and starts with a nice graphics design. Sound is also employed. Hardcopy options support the ADAM printer or a parallel interfaced dot matrix. "Easy Come - Easy Go" is a nicely organized package with several advanced features; it is well worth the money.

PRODUCT:	Business Pack I
MANUFACTURER:	E & T SOFTWARE
MEDIA TYPE:	datapack/disk
GRAPHICS/SOUND/DESIGN:	95
INSTRUCTIONS:	93
USEFULNESS vs. PRICE:	96
RECOMMENDATION:	highly recommended
PRICE:	18.95
RATED BY:	staff

This package includes three programs designed for the small business owner. Two programs are for entering and printing address files. You have several options to choose from.

The other program is an excellent inventory control program. You can let ADAM search for the items that you need to order. It calculates quantity on hand, total sold, etc. from the items that you enter. You can keep track of stock numbers, product descriptions, your cost, retail cost, quantity on hand and quantity on order.

This is a fine package; the inventory program alone is worth the cost of the whole package. We reviewed the latest version which uses SmartBASIC 2.0 (included) and the programs are speed - loaded.

LOCAL ADAM™ USERS GROUPS

INDIANA

Harold L. Shaw
350 Broken Arrow Court
Indianapolis, IN 46234

KANSAS

David E. Carmichael
1325 North Meridian, Apt. 201
Wichita, KS 67203

KENTUCKY

Keith Bowman
P.O. Box 434
Alexandria, KY 41001

MICHIGAN

ADAM Network
P.O. Box 85
East Detroit, MI 48021

MINNESOTA

Bill Rahn
12426 - 15th Street South
Afton, MN 55001

Downtown Minneapolis AUG
Thomis C. Gilmore
1424 West 33rd Street
Minneapolis, MN 55408

NEBRASKA

Omaha ADAM Users Club
Norman Castro
809 West 33rd Avenue
Bellevue, NE 68005

BULLETIN BOARD

ADAM Software
LVAC
P.O. Box 81146
Las Vegas, NV 89103

ADAM Hardware
Capital Software
P.O. Box 370
St. Louis, MO 63032

ADAM Software
C.M.E.
P.O. Box 339
Eastlake, CO 80614

ADAM Hardware
EVE ELECTRONIC SYSTEMS
320 Union Street
Millis, MA 02054

ADAM Software
Walters Software
Route 4, Box 289 - A
Titusville, PA 16354

ADAM Software
MMSG
P.O. Box 1112
Broomfield, CO 80020-8112

CP/M Conversions
PIVAR COMPUTING SERVICE
165 Arlington Heights Road
Buffalo Grove, IL 60089

PRODUCT LIST

PROGRAMMING UTILITY SOFTWARE

☐☐☐ Intel-BEST 3.3 (by DIGITAL EXPRESS)
* makes over three dozen changes to SmartBASIC V1.0; includes nine very user friendly MUSIC commands

>>> $24.95 (each) for non-subscribers
>>> $18.95 (each) for N&B subscribers

☐☐☐ Intel-LOAD V1.0 (by DIGITAL EXPRESS)
* converts BASIC 1.0 programs to LOAD up to 12 times faster; stays in RAM; onscreen help; two BSAVE options

>>> $15.95 (each) for non-subscribers
>>> $11.95 (each) for N&B subscribers

☐☐☐ Intel-LOAD V2.0 (by DIGITAL EXPRESS)
* converts BASIC 2.0 programs to LOAD up to 12 times faster; stays in RAM; onscreen help; two BSAVE options; works only in STDMEM

>>> $15.95 (each) for non-subscribers
>>> $11.95 (each) for N&B subscribers

☐☐☐ SmartBEST V1.0 (by DATA DOCTOR)
* makes several changes to SmartBASIC V1.0; not compatible with Intel-BEST 3.3

>>> $16.95 (each) for non-subscribers
>>> $14.95 (each) for N&B subscribers

☐☐☐ SmartTRIX I (by DATA DOCTOR)
* a set of 10 user friendly programming utilities; includes two very nice sprite programs; 60 page manual; disk and DDP version not compatible

>>> $29.95 (each) for non-subscribers
>>> $24.95 (each) for N&B subscribers

COLECO COPYRIGHTED SOFTWARE

☐☐☐ SmartLOGO (data pack only)
* Coleco's version of the popular language; 350 ++ pages

>>> $29.95 (each) for non-subscribers
>>> $24.95 (each) for N&B subscribers

☐☐☐ CP/M 2.2 (data pack only)
* Coleco's version of the popular operating system; configured for ADAM; 250 ++ pages

>>> $34.95 (each) for non-subscribers
>>> $29.95 (each) for N&B subscribers

☐☐☐ SmartFiler (data pack only)
* Coleco's general purpose database program; 38 pages

>>> $17.95 (each) for non-subscribers
>>> $14.95 (each) for N&B subscribers

RECREATION/GAMES SOFTWARE

☐☐☐ MageQuest (by REEDY SOFTWARE)
* superb graphic adventure; includes 9 levels of play in the main adventure plus 3 solo adventures; additional solo adventures are available from REEDY SOFTWARE

>>> $16.95 (each) for non-subscribers
>>> $14.95 (each) for N&B subscribers

☐☐☐ TRIVIAPAC I (by Mr. T. Software)
* 1200 questions; 6 categories; one to four players; graphics and sound; many hours of fun; DDP version only

>>> $17.95 (each) for non-subscribers
>>> $14.95 (each) for N&B subscribers

☐☐☐ KID'S TRIVIAPAC (by Mr. T. Software)
* 1080 questions; 6 categories; one to four players; graphics and sound; many hours of fun; DDP version only

>>> $17.95 (each) for non-subscribers
>>> $14.95 (each) for N&B subscribers

☐☐☐ Strategy Strain (by DATA DOCTOR)
* nine intellectually challenging computer classics; graphics and sound; superb Star Trek adventure

>>> $18.95 (each) for non-subscribers
>>> $14.95 (each) for N&B subscribers

☐☐☐ Quikfax Quest (by DATA DOCTOR)
* three academic quizzes; includes study mode (on screen and hardcopy); US capitals, world capitals, and Chemistry elements

>>> $18.95 (each) for non-subscribers
>>> $14.95 (each) for N&B subscribers

"NIBBLES & BITS" SOFTWARE

☐☐☐ N&B binder set 01 (by DIGITAL EXPRESS)
* all six issues from 07/86 thru 12/86 in a sturdy 3-ring binder; includes two DDP's or two disks containing all the programs

>>> $29.95 (each) for non-subscribers
>>> $24.95 (each) for N&B subscribers

☐☐☐ N&B issue programs (by DIGITAL EXPRESS)
* set 01: all the programs from 07/86 thru 09/86
* set 02: all the programs from 10/86 thru 12/86
* set 03: all the programs from 01/87 thru 03/87

>>> $9.95 (each) for non-subscribers
>>> $4.95 (each) for N&B subscribers

GUIDES/BOOKS/INSTRUCTIONS

☐☐☐ The Hacker's Guide to ADAM (vol one)
* Ben Hinkle's in-depth guide to the technical aspects of exploring ADAM; 60 pages; 18 programs

>>> $12.95 (each) for non-subscribers
>>> $10.95 (each) for N&B subscribers

☐☐☐ The Hacker's Guide to ADAM (vol two)
* Ben Hinkle's detailed guide to SmartBASIC V1.0; 110 pages; HELLO program includes several BASIC enhancements

>>> $12.95 (each) for non-subscribers
>>> $10.95 (each) for N&B subscribers

☐☐☐ Hacker's Guide software (by Ben Hinkle)
* all the programs from volumes one and two

>>> $5.95 (each) for non-subscribers
>>> $4.95 (each) for N&B subscribers

☐☐☐ EZ Ref 101 (by DIGITAL EXPRESS)
* approximately 700 Z80 instructions listed in NUMERICAL sequence; 9 pages; decimal, hex, op codes, operands

>>> $2.45 (each) for non-subscribers
>>> $1.95 (each) for N&B subscribers

☐☐☐ EZ Ref 102 (by DIGITAL EXPRESS)
* approximately 700 Z80 instructions listed in ALPHABETICAL sequence; 9 pages; decimal, hex, op codes, operands

>>> $2.45 (each) for non-subscribers
>>> $1.95 (each) for N&B subscribers

☐☐☐ Pinball Construction/HardHat Mac Guides
* 40 pages of instructions for the popular public domain package

>>> $2.45 (each) for non-subscribers
>>> $1.95 (each) for N&B subscribers

MISCELLANEOUS UTILITIES

☐☐☐ ShowOFF I (by DIGITAL EXPRESS)
* self-booting graphics design variety of print options (preset for Panasonic KX series and Star SG 10/15 and Star NX-10 printers); requires Centronics parallel interface for printer

>>> $29.95 (each) for non-subscribers
>>> $24.95 (each) for N&B subscribers

☐☐☐ ShowOFF II (by DIGITAL EXPRESS)
* a collection of machine code enhancements for SmartWriter and SmartBASIC; requires Centronics parallel interface, a Panasonic KX series printer, and a 64K expander

>>> $19.95 (each) for non-subscribers
>>> $14.95 (each) for N&B subscribers

MISCELLANEOUS SUPPLIES

☐☐☐ Coleco/LORAN digital data packs
* designed and formatted by Loranger Manufacturing

>>> $4.95 (each) for non-subscribers
> $39.95 (for 10) for non-subscribers
>>> $3.95 (each) for N&B subscribers
> $33.95 (for 10) for N&B subscribers

☐☐☐ Plain Label digital data packs
* formatted by E & T SOFTWARE

>>> $3.95 (each) for non-subscribers
> $33.95 (for 10) for non-subscribers
>>> $2.45 (each) for N&B subscribers
> $18.95 (for 10) for N&B subscribers

☐☐☐ Plain Label 5.25" disks for ADAM
* double sided, double density, with envelope

>>> $.89 (each) for non-subscribers
> $7.95 (for 10) for non-subscribers
>>> $.59 (each) for N&B subscribers
> $4.95 (for 10) for N&B subscribers

☐☐☐ SmartWriter printer ribbons
* just like the one that came with your ADAM

>>> $5.45 (each) for non-subscribers
> $15.45 (for 3) for non-subscribers
>>> $4.95 (each) for N&B subscribers
> $13.45 (for 3) for N&B subscribers

☐☐☐ multipurpose adhesive labels
* white, tractor feed, 3 1/2 x 15/16, fan fold, single column

>>> $2.95 (for 500) for non-subscribers
> $5.45 (for 1000) for non-subscribers
>>> $2.25 (for 500) for N&B subscribers
> $3.95 (for 1000) for N&B subscribers

■■■■■■■■■■

☐☐☐☐☐ Unless otherwise noted, all software is available on disk or datapack.

☐☐☐☐☐ All DIGITAL EXPRESS media is warrantied to be free from defects in materials and workmanship. If the storage medium proves defective at any time, return it to us for repair or replacement (at our descretion).

☐☐☐☐☐ The product prices listed herein may be subject to change after April 15, 1987.

■■■■■■■■■■

PUBLIC DOMAIN SOFTWARE

DEI Public Domain Facts

You may get any of the volumes itemized below on data pack or disk for ONLY $5.95 as a N&B subscriber, or for just $9.95 as a non - subscriber. Subscribers also have an option to get a volume FREE (limit three per calander month); this option does NOT apply to the volumes in the Coleco Unreleased Titles Library.

Here's how to get one FREE. (1) Contribute an original program for any library. (2) Send a signed statement that the program is not copyrighted. (3) Send the program on DDP (digital data pack) or disk; one DDP or disk for each volume that you want to exchange. And, (5) include a return mailer with sufficient postage or send $2.50 for shipping costs.

SmartBASIC V1.0 Library

You must boot your own SmartBASIC first in order to use the volumes in this library. All programs will speed load. Each volume (except the utility volumes) is controlled by a user friendly ramdisk (does NOT require the 64K expander) central menu.

N&Bgames01: An assortment of text adventures, board games, and animation games -- 130K of files.

N&Bgames02: An assortment of text adventures, board games, and animation games -- 155K or files.

N&Bgraph01: A variety of graphics displays and music programs -- 88K of files.

N&Bmath01: Several scientific and financial math programs -- 114K of files.

N&Butil01: Intended for more advanced programmers this volume includes programming utilities -- 100K of files.

SmartPAINT pictures Library

In order to view/use the volumes in this library you should have SmartPAINT (from ShowOFF I) or the HGR Picture Manager program in the February 1987 issue of "NIBBLES & BITS" (page 16).

N&Bpix01: 13 different HGR picture files stored in SmartPAINT format.

N&Bpix02: 13 different HGR picture files stored in SmartPAINT format.

Coleco Unreleased Titles Library

SmartBASIC 2.0: Improved interpreter; 49K program; works with or without the 64K expander.

Pinball Construction/Hardhat Mac: Best of Electronic Arts (latest version with two demo pinball games).

ADAMLink II: Supports uploading and down loading of SmartWriter compatible files; includes U/D instructions; requires the ADAMLink modem.

Jeopardy: The extremely popular ADAM game; just like the game show; great graphics; hall of fame; one to three players.

Super SubRoc: 90K arcade-type game; super graphics; hall of fame; one or two players.

Troll's Tale: Easy to play graphic/text adventure; supports one player; disk and DDP versions NOT compatible.

CP/M 2.2 Library

The volumes in the library require that you boot your own CP/M 2.2 package first.

CP/Mgames01: 30 games.

CP/Mgames02: 25 games.

Test/Music: System tester (requires 64K expander) and a hodgepodge of music samples -- from an unreleased Coleco cartridge program.

Pinball Games Library

Each volume in this library is self-booting and may be used with the Pinball Construction Set.

N&BPBgames01: 10 pinball games.

N&BPBgames02: 10 pinball games.

N&BPBgames03: 10 pinball games.

Miscellaneous Collections Library

MWplus01: A collection of improvements to MultiWrite by Strategic Software. Written by Jim Guenzel, a N&B subscriber.