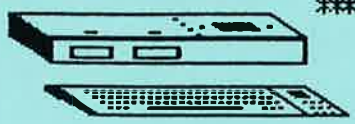
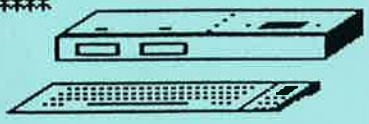
NIBBLES & BITS

The Comprehensive Monthly Newsletter for ADAM Users



P.O. Box 37
Oak Hill, WV 25901



June 1987
vol: 1, nmb: 12
single issue: $3.50

This issue includes 8 SmartBASIC program LISTs and 6 disassembled Z80 routines.

ColecoVision, ADAM, SmartBASIC, and SmartWriter are registered trademarks of Coleco Industries, Inc.
CP/M 2.2 is a registered trademark of Digital Research, Incorporated.

"NIBBLES & BITS" is printed in the USA. Copyright (c) 1987 by DIGITAL EXPRESS. All Rights Reserved.

DESIGNED and PRINTED with the amazing ADAM™ computer (using an Orphanware 64K expander, an Eve Electronics Centronics parallel interface, a Panasonic KX-P1080 dot matrix printer, ShowOFF I, and ShowOFF II).

NIBBLES & BITS

ADAM ADAM ADAM ADAM

ADAM ADAM ADAM ADAM

DIGITAL EXPRESS

The N&B Staff

EDITOR-IN-CHIEF:
Dr. Solomon Swift

DESIGN EDITOR:
Tim Whetstine

TECHNICAL DIRECTOR:
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 $30.00 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. All subscription issues are sent by U.S. mail, FIRST CLASS.

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 <u>each</u> month.

To insure that you don't miss any issues, please renew early and let us know promptly of any address change. <u>Please</u> 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

The editor(s) and publisher have exercised due care in the preparation of this newsletter. Neither the N&B staff, nor DEI, nor any contributors (of any capacity) make any warranty either expressed or implied with regard to the information contained herein, either by interpretation, use, or misuse. Reviews and opinions submitted by the readership at large do not necessarily reflect the opinions of the editor or staff. DEI has no affiliation with Coleco Industries. Unless stated otherwise, all correspondence shall be considered as "open to public review".

EDITOR'S NOTE

Summer is here!!! This is the time of year when many of us spend a little less time with the computer and a little more time in the sun. In fact, nearly everyone at DIGITAL EXPRESS is on vacation now; it's rather quiet around here.

Before you know it, though, we'll all be scurrying about getting next month's issue ready, writing more programs, etc. We have several innovative projects underway. It looks like DEI will be even more productive in our second year. Once again, I want to thank ALL of you for making our efforts so successful. Thanks also to everyone who sent us anniversary cards and good wishes.

I'll be brief this month so we can get this issue off to the print shops. If everything works out right, you should get the July issue in about 10 days. There is a strong possibility. I'm keeping my fingers crossed...

Solomon Swift
EDITOR-IN-CHIEF

THE MAD BOMBER

N&B NEWS

□□□ **TurboDISK 2.0** is now completed. It adds a ramdisk capability to SmartBASIC 2.0. TurboDISK 2.0 does not include a copy program as TurboDISK 1.0 does. The price is only $15.95 for non-subscribers and JUST $11.95 to subscribers of "NIBBLES & BITS".

□□□ We have added two fine software packages from Marathon Computer Press to our product list: **The Spanish Vocabularian** and **MegaUtil.**

□□□ Our fifth collection of SmartPAINT files is completed. This gives you 65 ready - to - use public domain pictures to choose from.

□□□ Last month we listed a few subscribers who have given valuable help to DIGITAL EXPRESS. Here are four more devoted ADAM users who have offered important contributions. Thank you VERY MUCH!!!

D.L. Decker
Pat Herrington
Dave McIntosh
Lee Smith

□□□ Our fourth quarterly collection of "NIBBLES & BITS" programs is completed. It is available to subscribers for only $4.95. This one contains all the programs LISTed in the April, May, and June issues.

□□□ DIGITAL EXPRESS has now released nine commercial software packages. These include: Intel-BEST 3.3, Intel-LOAD V1.0, Intel-LOAD V2.0, ShowOFF I, ShowOFF II, ShowOFF IIa, TurboDISK 1.0, FontPOWER, and TurboDISK 2.0. Many of you have purchased every package that we've released. In appreciation, we're offering a special bonus.

Any subscriber who has purchased seven of these commercial titles (either directly from us or from an authorized dealer) before July 31st will be entitled to a ten percent discount on his / her next purchase with DIGITAL EXPRESS. Just send photocopies of your receipts with your order and deduct the 10% from the subtotal amount on the order form.

□□□ It's time for our semi - annual special on back issues of "NIBBLES & BITS". Until August 14th you can get any back issues for only $2.95 each; that's a 16% savings off the standard price.

ADAM NEWS

□□□ Strategic Software has finished their **ProofREADER** spelling checker. See a review in this issue by Randal Bondi.

□□□ E & T SOFTWARE has a new business address. You can get their latest catalog of numerous products for the ADAM computer by writing to:

E & T SOFTWARE
1010 Westminster
Garland, TX 75040

□□□ D.L. Decker Enterprises is currently featuring a close - out special on ADAMlink modems. While quantities last, you can get the ADAMlink modem including the ADAMlink II and XMODEM software plus a free software title of your choice for only $59.95. In addition, the ADAM AutoDialer is being closed out for just $29.95 (including free programs). Their address is:

D.L. Decker Enterprises
Route 2, Box 15
Spring Mills, PA 16875-9720

□□□ Last month we mentioned two new releases from **VideoSongs**, "Potpourri" and "The Beatles", for use with **VideoTunes** by FutureVision. We'll have a review of these two titles in an upcoming issue. The folks at VideoSongs are currently working on a collection of Christmas tunes.

□□□ If you haven't accessed CompuServe in a while, you may be surprised to discover that the primary support for ADAM has moved. Now you can enter "GO CLUB" instead of "GO FAMFORUM". Two data libraries (collections of public domain software) are devoted to ADAM. **DL9** contains SmartBASIC and SmartLOGO programs; **DL10** contains CP/M 2.2 programs.

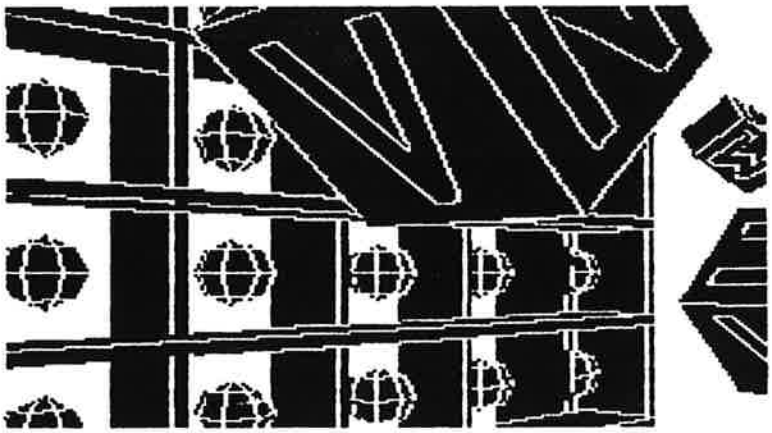
□□□ A couple of months ago we mentioned that ADAM Users Group 1986 has a great number of surplus items. They also publish a newsletter periodically for ADAM users. Issues are packed with valuable information; and best of all, subscriptions are FREE. Just send them your name and address. See this month's BULLETIN BOARD for their address.

□□□ Orphanware is selling a 300 / 1200 baud modem, AVATECH 1200, for just $169.95. This Hayes compatible external modem comes with an RS232 interface to connect it to ADAM and a version of MEX, a CP/M 2.2 program, to access the fast data transfer modem. They will even give you a $25 trade-in allowance for your ADAMlink modem. See this month's BULLETIN BOARD for Orphanware's address.

□□□ Terry Fowler of gHAAUG has a limited quantity of disk drive power transformers for the ADAM disk drive. These are exact replacements without the plastic top for only $9.95 (plus $2.50 for shipping). See the BULLETIN BOARD for his address.

□□□ For you modem users, here's a BBS that you should try:

The Hudson ADAM/Link
20 hours/day (10am to 6am EST), 7 days a week
Parameters: 7-1-even (300 baud)
Sysop: Fred Vicente
Address: P.O. Box 7266
North Bergen, NJ 07047
BBS phone: (201) 224-5764



TIDBITS

§§§ For most of the Coleco Super Games you can reset the "hall of fame" scores by pressing the number sequence "9 - 8 - 9" on the game controller keypad. This also works with Super SubRoc, the Coleco public domain game.

§§§ There are 16 treasures to be found in the Coleco public domain adventure, Troll's Tale. The objective of the adventure is to discover the word play on the title after finding all the treasures.

§§§ CP/M files that have the ".COM" extension are commands. You can execute the command by simply entering the file name (without the extension) and then pressing <return>.

§§§ You can get unlimited lives with Hard Hat Mac, the Coleco PD game, by pressing the "#" button on the game controller during the first screen. You can pause the game by pressing the "*" button on the game controller. The left fire button causes your character to jump. The right fire button releases the jack hammer.

§§§ There are three known fixes for the DATA / REM space bump bug with SmartBASIC V1.0. Each one overcomes the dreaded extra space; but there are some limitations. The four POKEs revealed by Ben Hinkle in "The Hacker's Guide to ADAM: Volume 2" cause a minor bug with the REM command. You can't use the command without some text after it. If you do, the interpreter adds stray ASCII values from the input buffer. The PatchWORK and the SmartBASIC 2.0 fixes force the DATA statement to be the last command on a program line; anything after the command is recognized as data. The Intel-BEST 3.3 fix is the only patch that doesn't have any of the (very minor) side effects.

§§§ ADAM's operating system sets up a 22 byte buffer for reading the keyboard while performing other tasks. Some machine code programs do not take advantage of this feature; SmartWriter does to some extent. JKL Utilities and ShowOFF II do also. And, SmartBASIC uses the feature in the immediate mode. To try it, SAVE a program to tape. While the tape is spinning, press the space bar and then type "catalog" and press <return>. You won't see anything on the screen yet. But, when the interpreter is finished SAVEing the file, it will execute the command as if you had just entered it.

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. For the benefit of all readers my reply, where applicable, is generally more detailed than any written reply. Unless the reader requests differently, street addresses are omitted.

A REAL-TIME CLOCK

I have an idea for an interesting program, a clock display in BASIC. I have a public domain wedge that works well, but I have to run a program to access the wedge and display the time. This is inconvenient. Would it be possible to write a program that would allow time display in BASIC the way that ADAMlink shows connect time?

Frank Curley, Jr.
Rochester, MN

IN RESPONSE: Yes, it is possible. In fact, this is one of the features that I plan to include in PowerBASIC. See the HACKER'S DELIGHT department of this issue for program that will accomplish this with SmartBASIC V1.0. The program does have a rather limited use as it works only while in the TEXT mode. But, the most interesting aspect of the routine is its simplicity. It works on the interrupt from the video chip so that it requires no additional hardware.

CHANGING ShowOFF II SCREEN COLOR

The black letters on a yellow screen with the ShowOFF IIa utility is a little difficult to read on my monitor. Is there a way to change colors?

Hector Sanchez
Corpus Christi, TX

IN RESPONSE: Yes. See the HACKER'S DELIGHT department of this issue for a program that will allow you to change screen colors with ShowOFF II or ShowOFF IIa.

VIDEO MODES

What is multicolor mode? Is this the same as BASIC's GR mode?

- name withheld -

IN RESPONSE: ADAM's video chip supports four distinct video modes. The most common of these is 32 column text or "graphics mode 1". A 40 column "text" mode is also available. Another mode corresponds to BASIC's HGR2; it is called "graphics mode 2". HGR and GR are just variations of this video mode. "Multicolor mode" is entirely different from the other three. It uses small blocks (4 by 4 pixels) to design graphics. I plan to start a workshop on this mode in August issue.

EXPANDING YOUR SYSTEM

PRINTER ALTERNATIVES

(part 7)

This articles continues the discussion of what to look for in a second printer from the April issue (page 7).

One cosideration is print speed. Dot matrix printers are much faster than letter quality printers, such as the SmartWRITER printer. Most of the lower priced printers have a print speed of 100 to 200 characters per second (CPS) in draft mode for the Pica pitch. Draft mode is good for quick prints of a working copy. When you're ready for the final copy, you'll most likely want to use "near letter" or "correspondence" quality. Many printers provide a selector switch for changing print qualities. With "near letter" quality, the dots are bearly discernable. The drawback of this improved hardcopy is a sacrifice of print speed. Near letter quality is generally about one - fifth as fast as draft quality.

There are two basic types of paper to use with your printer: "standard cut sheet" and "continuous form". Most impact dot matrix printers allow you to select the type of paper that you want to use. "Friction feed" is used to advance cut sheet paper; it works just like a standard typewriter. "Tractor feed" is used to advance continuous form paper. The printer has two devices called "tractors" which have a number of protruding pins that fit inside the evenly spaced holes at the left and right margins of the paper. You should look for fully adjustable tractors which allow for tractor feed adhesive labels, index cards, etc.

If you've never used a second printer, "print buffers" may be a new concept to you. A buffer is a temporary storage area. Most printers come with at least a 1K buffer (about a half page of text). This allows you to continue with other computer operations while the document is being printed. In reality, you should only consider a large print buffer if you do extensive printing. If you're a typical computer hobbyist, a large buffer will most likely be too expensive to make it cost effective. If you do get a printer with the standard one or two "K" buffer, be sure that the printer is expandable.

Nearly all printers today are factory equipped with built - in software called "firmware". This feature provides for changing certain functions of the printer when it receives a particular sequence of numbers. These sequences usually start with the number "27". In ASCII code this is the <escape> value. Thus, these printer codes are referred to as "escape codes".

Buffer size and print speed are the two factors that usually determine the price of a printer. Yet, the escape codes are the element that permits you to access the true power of the printer. The Panasonic 1080 and 1080i provide the most extensive escape code features of any comparably priced impact dot matrix unit.

In an upcoming issue, we'll start a workshop on accessing the escape codes using our PR#2 / PR#3 command (LISTed in the February 1987 issue, page 20). Escape codes allow you to change from Pica pitch to Elite pitch, use boldface printing, use double width characters, etc. You should be certain to get a printer that has Epson compatible escape codes. More next month...

BIT BY BIT

LOW RESOLUTION GRAPHICS

(part 5)

We received a lot of positive feedback in regard to last month's "breakout" game. The following explanation should help you to understand exactly what is happening as you play the game.

Line numbers 10 through 150 set up (or initialize) the variables and design the game screen. Line number 10 contains a short REMark statement. It's a good practice to include several of these programmer notes in your own creations. When you frist write a program, the notes may seem superfluous. But, when you come back to the program months later, you will find the REMarks very helpful in studying the LIST.

Line number 50 begins the error trapping and turns the cursor off. Turning the cursor off provides a much more attractive display. Line number 60 sets the GR screen colors. Line numbers 100 and 110 draw the game's borders. Line numbers 120 and 130 draw the two sections of bricks. Notice how the "x" value is used the determine both the brick color and the brick position on the screen. Line number 140 initializes the control variables. These are as follows:

ba = number of balls (minus one)
ht = horizontal ball position
p1 = first paddle block (from left)
sc = player score
vt = vertical ball position

Line number 150 completes the starting screen by executing three subroutines. You should use a subroutine whenever the particular set of lines will be used more than once.

Line numbers 200 through 900 control game play. As you can see, the game just executes six subroutines repeatedly. Originally the game was written to play against a timer. This final version was changed to permit scoring by hitting the bricks. Thus, line number 900 should read simply "GOTO 200".

Line numbers 910 through 930 display the message you see when you've run out of balls. Notice that the game exits by going to line number 2600. ALL exits from the game branch program execution to this line.

Line number 2000 is the routine that draws the player's paddle at the lower part of the graphics screen. The paddle is two blocks wide.

Line number 2100 draws the ball at the current screen position. The values of "ht" and "vt" must be set before this routine can be executed.

Line number 2200 prints the player's score. Line number 2210 prints the number of balls left.

Line numbers 2300 through 2360 determine the paddle position based on joystick movement. Page A-66 of the SmartBASIC manual gives a brief explanation of the "PDL(5)" function. Since we are only concerned with left and right movement of the paddle, we only need to use two of the PDL(5) values, *ie*, "8" for left and "2" for right.

Line numbers 2400 through 2440 print the screen that you see whenever you lose one ball. They also reset the vertical and horizontal ball movement offsets, *ie*, "vf" and "hf". If vf = -1, then the ball moves up. If vf = 1, the ball moves down. If hf = -1, the ball moves to the left. If hf = 1, the ball moves to the right. When the ball is first released, hf = 0 -- there is no left or right movement (until it strikes the paddle).

Line number 2500 erases the old ball from the screen. This routine is used in conjunction with the one on line number 2100 to continuously erase an old ball and draw a new one. This is what gives the ball the appearance of animation.

Line numbers 2600 through 2950 make the various sound effects. Without using programs like Intel-BEST 3.3 or SmartLOGO, the simplist method of getting sound is to PRINT CHR$(7). You can change the "bell" sound by POKEing values into address 17954.

Line numbers 3000 through 3600 control the ball's movement and the resultant effects. One of the most interesting aspects of this rather complex routine is the way that it RETURNs from a GOSUB. Rather than using the RETURN command, most of the possible exits just GOTO another subroutine that does end with a RETURN command. As you gain experience with BASIC, you'll understand how this convenient trick conserves memory and speeds program execution.

Next month, ACSII codes ...

BYTE-SIZED BASIC

POKES TO PLAY WITH

(part 12)

More on the ROT command:

This is the conclusion of Leonard Adolph's research notes on the ROT command. Once again, we thank Mr. Adolph for contributing these facts on one of BASIC's most abstruse commands.

ROT = 0 gives the following values:

16768 =	0	no left/right move
16769 =	248 (-8)	move up full distance
16770 =	8	move right full distance
16771 =	0	no up/down move
16772 =	0	no left/right move
16773 =	8	move down full distance
16774 =	248 (-8)	move left full distance
16775 =	0	no up/down move

These locations can be POKEd to change a shape's design without changing the "shape table". (The two programs on page 9 demonstate this trick.) SmartBASIC V1.0 comes with a default shape table containing one shape. This table occupies addresses 26574 through 26587. You can change the shape to a perfect square by POKEing a "5" into address 26576.

The program at the top of page 9 shows what can be done by changing the left and right plot distances. The next program demonstrates changing distances and directions to considerably warp the original square. Mr Adolph adds, "I think anyone who likes to program games in BASIC can find uses for this information. The second program, in particular, has potential for an airplane simulator subroutine."

The first demo draws some concentric rectangles. Line number 100 sets the pointer to the default shape table, *ie*, (103 * 256) + 206 = 26574. Line numbers 120 and 130 assign the ROT data addresses to variable names. This makes the data table easier to work with.

The second demo program animates an airplane wing. Distorting the values in the ROT data table causes a very interesting effect with this one. It even adds some two-point perspective to the moving wing.

Correcting the GR and HGR color tables:

In our July issue (on page 12) we revealed a method to correct the GR and HGR color tables so that you would only have to learn one set of color values. These values are the ones that are actually used by the video chip. To recap:

correct HGR color table:
```
FOR x = 0 TO 15: POKE 18765+x, x: NEXT
```

correct the GR color table:
```
FOR x = 0 TO 15: POKE 18781+x, x: NEXT
```

Guy Cousineau of Ottawa, Canada has shown us an improved method of accomplishing the same results. His technique has two distinct benefits. First, it actually speeds up drawing slightly. Second, this method makes it much easier to change your SmartBASIC back-up; you only need to change nine addresses (as opposed to 32 addresses with our trick).

correct HCOLOR translation:
```
POKE 18728, 0: POKE 18729, 0: POKE 18730, 0
```

correct COLOR translation:
```
POKE 18735, 0: POKE 18736, 0: POKE 18737, 0
```

correct SCRN translation:
```
POKE 19256, 0: POKE 19257, 0: POKE 19258, 0
```

How to stop PR#1 screen echo:

Whenever you use the PR#1 command from SmartBASIC, each character is echoed on the screen. This can be a nuisance if it messes up your screen. The problem is that the interpreter routine to send characters to the printer ends by falling through to the routine that prints characters on the screen. We used a trick in last month's "label maker" program (page 14) that works around this screen echo. When you use this trick you stop screen printing. So when you're ready to go back to PR#0, you'll need to reset the default value for the address.

To disable screen echo:
```
POKE 12043, 201
```

To enable printing to the screen:
```
POKE 12043, 245
```

WAGES CALCULATOR

Sometimes we get so involved with advanced programming applications that we overlook some of the more simple applications. Consider the program on page 10. It computes gross pay based on the germane data which you input.

First the program asks you to enter the hourly rate of pay. Then, you enter the overtime pay factor. For time-and-a-half you enter "1.5", and so on. Next, you enter the number of standard pay hours worked. Concluding the inputs, you enter the overtime hours worked. In a flash, ADAM computes the base pay, overtime pay, and their sum (the gross pay).

```
 10 REM shape distortion demo
 20 REM by Leonard Adolph
 30 REM Flint, Michigan
100 POKE 16766, 206: POKE 16767, 103: POKE 26576, 5
110 POKE 25471, 17: HGR: HCOLOR  = 15: s = 20: ROT  = 0
120 DATA 16768,16769,16770,16771,16772,16773,16774,16775
130 READ ux, uy, rx, ry, dx, dy, lx, ly
140 FOR x = 1 TO 8: POKE rx, x
150 POKE lx, 256-x: SCALE  = s+2*x
160 DRAW 1 AT 125, 140+2*x: NEXT
```

```
 10 REM shape distortion demo #2
 20 REM by Leonard Adolph
 30 REM Flint, Michigan
100 POKE 16766, 206: POKE 16767, 103: POKE 26576, 5
110 POKE 25471, 17: HGR: HCOLOR  = 15: SCALE  = 15: ROT  = 0
120 DATA 2,1,0,255,254,253,252,251,250
130 FOR x = 1 TO 9: READ du: NEXT
140 DATA 16768,16769,16770,16771,16772,16773,16774,16775
150 READ ux, uy, rx, ry, dx, dy, lx, ly: RESTORE
200 POKE rx, 2: POKE lx, 252
210 FOR x = 250 TO 255: XDRAW 1 AT 125, 140
220 POKE ux, x: POKE dx, 255-x+3
230 DRAW 1 AT 125, 140: NEXT
300 FOR x = 0 TO 8: XDRAW 1 AT 125, 140
310 POKE ux, x: READ d: POKE dx, d: DRAW 1 AT 125, 140
320 NEXT: XDRAW 1 AT 125, 140
330 POKE uy, PEEK(uy)+1: POKE dy, PEEK(dy)-1
340 IF PEEK(uy) = 255 THEN  END
350 RESTORE: GOTO 210
```

June 1987

SUN	MON	TUE	WED	THR	FRI	SAT
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30				

July 1987

SUN	MON	TUE	WED	THR	FRI	SAT
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

```
 1Ø REM wages calculator
1ØØ TEXT: PRINT " This program will calculate"
11Ø PRINT " gross pay based on your"
12Ø PRINT " entries.": VTAB 6
13Ø PRINT " 1 = calculate gross pay"
14Ø PRINT " 2 = exit the program"
15Ø GET k$: k% = VAL(k$): IF k% < 1 OR k% > 2 GOTO 15Ø
16Ø IF k% = 1 GOTO 3ØØ
2ØØ TEXT: PRINT " program terminated.": END
3ØØ HOME: INPUT " enter base rate ($/hr): "; br
31Ø PRINT: INPUT " enter OT pay factor: "; ot
32Ø PRINT: INPUT " enter base hours worked: "; bh
33Ø PRINT: INPUT " enter OT hours worked: "; oh
4ØØ bb = bh*br: oo = ot*br*oh: gp = bb+oo: PRINT: PRINT: PRINT
41Ø PRINT " BASE PAY = "; bb
42Ø PRINT " OT PAY   = "; oo
43Ø PRINT " GROSS PAY= "; gp
5ØØ PRINT: PRINT: PRINT " press any key ...": GET go$: RUN
```

```
enter base rate ($/hr): 8.75

enter OT pay factor: 1.5

enter base hours worked: 40

enter OT hours worked: 10


BASE PAY = 350
OT PAY   = 131.25
GROSS PAY= 481.25


press any key ...
```

HACKER'S DELIGHT

TRANSFERRING DATA

(part 7)

Last month we discussed a simple routine that allows you to "fill" up to 255 consecutive addresses with a single value. This month's routine will fill any number of bytes that you specify. There are three set-up values for this one. Load the **HL** pair with the RAM address to start the filling. Load the **DE** pair with the number of bytes to fill. And, load the **B** register with the fill value. This routine is very convenient for clearing a large space of RAM.

You should select the area to clear with care. You don't want to erase the Operating System, for example. Also, you should NEVER try to erase the addresses that contain the routine. If you do, the system will crash.

This routine provides a good illustration of how fast z80 routines are. This is particularly evident when comparing it to a similar BASIC routine. Consider the program below which clears addresses 30000 through 39999. The BASIC version of RAM clearing takes about 34 seconds. The z80 equivalent is finished in about one-fourth of a second -- well over 100 times faster than BASIC.

The mnemonics and hex code are detailed at the top of the next column.

```
set up:
LD   HL, $7530
LD   DE, $2710
LD   B,  $FF

process:
LD   (HL), B
INC  HL
DEC  DE

check for completion:
LD   A,  D
OR   E
JR   NZ, $F9

exit:
RET
```

Decrementing a register pair doesn't have an effect on the **flags** register. The typical trick for overcoming this limitation is to move one byte of the double register into the accumulator and perform a logical **OR** with the other byte of the pair. A relative jump (signed displacement) of 249 ($F9) is a backwards move of seven bytes (256 - 249 = 7).

EOS JUMP TO SmartWRITER

The 62nd vector in the EOS jump table is a JumP to SmartWriter. It can be executed by CALLing address 64743 (231, 252). The actual routine begins at address 64148 (148, 250). It is nine bytes in length. Asmb#46 (at the top of page 12) details the routine. It requires no set up.

The routine bank switches the word processor in from the ROM chip, *ie*, it sends a zero to the bank switch port. When SmartWriter is in the lower bank of RAM, the routine just jumps to the start address, 256 ($0100).

```
 10 REM this program demonstrates how to clear a large
 20 REM area of RAM in Z80 code
100 LOMEM :29000
110 DATA 33,48,117,17,16,39,6,255,112,35,27,122,179,32,249,201
120 FOR x = 28000 TO 28015: READ mc: POKE x, mc: NEXT
200 TEXT: PRINT " 1 = clear RAM in BASIC"
210 PRINT " 2 = clear RAM in z80"
220 PRINT " 3 = exit program"
230 GET k$: k% = VAL(k$): IF k% < 1 OR k% > 2 GOTO 300
240 ON k% GOTO 1000, 2000
300 TEXT: PRINT " program terminated.": END
1000 GOSUB 3000: FOR x = 30000 TO 39999: POKE x, 0: NEXT
1010 GOTO 200
2000 GOSUB 3000: CALL 28000: GOTO 200
3000 HOME: PRINT " clearing addresses 30000 to"
3010 PRINT " 40000 ...": RETURN
```

TITLE (asmb#46):
EOS JUMP TO SmartWRITER

addr:	Label:	Value(s):	Op Code:	Comment:
64148	getbnk	58, 23,252	LD A, (64535)	;put zero in accumulator
64151	bnkswt	205, 20,253	CALL 64788	;CALL EOS bank switch
64154	start	195, 0, 1	JP 256	;start SmartWriter

TITLE (asmb#47):
INIT blocks fix

addr:	Label:	Value(s):	Op Code:	Comment:
58460	ckdisk	254, 7,	CP 7	;check for disk value
58462		48, 4,	JR NC, 4	;if not, then cktape
58464		30,159,	LD E, 159	;set disk volume size
58466		24, 10,	JR 10	;goto done
58468	cktape	254, 25,	CP 25	;check for tape value
58470		48, 4,	JR NC, 4	;if not, then setrdk
58472		30,255,	LD E, 255	;set tape volume size
58474		24, 2,	JR 10	;goto done
58476	setrdk	30, 63,	LD- E, 63	;set ramdisk volume size
58478	done	197,	PUSH BC	;store BC pair
58479		213,	PUSH DE	;store DE pair
58480		229,	PUSH HL	;store HL pair
58481		195, 40,243,	JP 62248	;continue EOS INIT

TITLE (asmb#48):
Directory Size Patch

addr:	Label:	Value(s):	Op Code:	Comment:
58484	size	246, 48,	OR 48	;convert to ASCII
58486		205,218, 46,	CALL 11994	;print directory size
58489		62, 32,	LD A, 32	;load ASCII space
58492		205,218, 46,	CALL 11994	;print space
58495	name	33, 16, 66,	LD HL, 16912	;set pointer to vol name
58498		205,140, 83,	CALL 21388	;print volume name
58501	done	195, 96, 47	JP 12128	;print a <return>

THE INIT BLOCKS FIX

At the top of page 16 of last month's issue we LISTed another module for the PatchWORK series. It corrects the EOS INIT function so that disks are INITed to 160 blocks, data packs are INITed to 256 blocks, and a ramdisk is INITed to 63 blocks. The patch is identical to the one used with **TurboDISK 1.0**.

Line number 10030 of the program allows you to POKE anywhere in standard RAM. The rest of the program may be considered as two integral components. The first component (line numbers 10100 and 10110) patches a jump over the beginning of the default INIT routine. The jump branches execution to the second component (line numbers 10120 through 10140). This one does all the work; it checks the current device code (stored in the accumulator) and changes the value of the **E** register (volume size) accordingly. Upon completion, this component then branches execution back the default INIT routine (continuing just after the patched jump).

This second component is detailed in asmb#47 on page 12 (previous page). The method of checking the current device code is rather simple. If it is less than seven, then it is a disk (code 4 or 5). If it is greater than seven and less than 25, then it is a tape (code 8 or 24). Any other value (hopefully 26) is assumed to be a ramdisk.

THE DIRECTORY SIZE PATCH

The second program on page 16 of last month's issue patches a part of the interpreter to slightly modify the CATALOG command. It displays the directory size just to the left of the volume name. And, the size is formatted in the column of the other file sizes.

This program has three components. The first one changes the word "volume" to "title". Line numbers 11100 thru 11120 accomplish this. The second component (line numbers 11200 through 11210) gets the attribute byte and masks out the upper bit (status bit) to determine the directory size. Then it just jumps to the third component.

This final section is detailed in asmb#48. It first converts the decimal directory size to an ASCII value (**OR 48**). Then, it prints the ASCII value, a space, and the volume name. It concludes by printing a <return>.

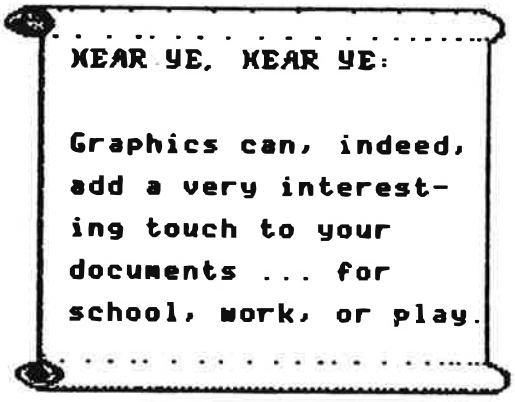
A TROLL'S TALE PATCH

"Troll's Tale", the Coleco public domain adventure, has a minor bug in the boot block. It will only load from one specified drive. The program on the next page (page 14) allows you to change the boot drive for the adventure. This is particularly useful for making a disk to data pack (or vice versa) archival copy.

Two bytes in block zero contain the default drive value. These are the 11th and 308th bytes. This program just provides a user - friendly method of changing the drive value.

CHANGING ShowOFF II SCREEN COLOR

The ShowOFF II menu screen is dark blue with white letters. The ShowOFF IIa menu screen is light yellow with black letters. Since the z80 program uses the 40 column text mode, only one byte controls the screen color. It is the 48th byte on block two. The program on page 15 permits you to easily change this screen color byte on the medium.



```
 10 REM patch to change the default drive for Troll's Tale
100 LOMEM :30000
110 DATA 62,4,1,0,0,17,0,0,33,0,108,205,243,252,50,255,107,201
120 FOR x = 29000 TO 29017: READ mc: POKE x, mc: NEXT
130 DATA 62,4,1,0,0,17,0,0,33,0,108,205,246,252,50,255,107,201
140 FOR x = 29100 TO 29117: READ mc: POKE x, mc: NEXT
150 DATA 4,5,8,24,disk one,disk two,tape one,tape two
160 FOR x = 1 TO 4: READ dv%(x): NEXT
170 FOR x = 1 TO 4: READ dv$(x): NEXT
200 TEXT: PRINT " This program allows you to set";
210 PRINT " the default drive for Coleco's";
220 PRINT " PD game, 'Troll's Tale'."
230 VTAB 10: PRINT " Which drive now contains"
240 PRINT " Troll's Tale?": PRINT
250 PRINT " 1 = tape one": PRINT " 2 = disk one"
260 PRINT " 3 = exit program"
270 GET k$: k% = VAL(k$): IF k% < 1 OR k% > 2 GOTO 300
272 IF k% = 1 THEN  POKE 29001, 8
274 IF k% = 2 THEN  POKE 29001, 4
276 GOTO 500
300 TEXT: PRINT " program terminated.": END
500 HOME: PRINT " press <return> to read the"
510 PRINT " medium ..."
520 GET go$: IF go$ <> CHR$(13) THEN  RUN
525 HOME: PRINT " reading block zero ..."
530 CALL 29000: IF PEEK(27647) = 128 GOTO 550
540 HOME: PRINT " read error on block zero!": END
550 IF PEEK(27658) = PEEK(27955) GOTO 600
560 HOME: PRINT " Troll's Tale not detected!!": END
600 pc = PEEK(27658): IF pc = 4 THEN  pc$ = dv$(1): GOTO 700
610 IF pc = 5 THEN  pc$ = dv$(2): GOTO 700
620 IF pc = 8 THEN  pc$ = dv$(3): GOTO 700
630 IF pc = 24 THEN  pc$ = dv$(4): GOTO 700
640 GOTO 560
700 HOME: PRINT " The current default drive for"
710 PRINT " Troll's Tale is "; pc$; ".": VTAB 6
720 PRINT " 1 = change default drive"
730 PRINT " 2 = restart this program"
740 GET k$: k% = VAL(k$): IF k% < 1 OR k% > 2 GOTO 300
750 IF k% = 2 THEN  RUN
800 HOME: PRINT " Select new default drive for"
810 PRINT " Troll's Tale."
820 FOR x = 1 TO 4: VTAB x+5: HTAB 2: PRINT x; "="; dv$(x): NEXT
830 GET k$: k% = VAL(k$): IF k% < 1 OR k% > 4 GOTO 300
840 nd% = dv%(k%): POKE 27658, nd%: POKE 27955, nd%
900 HOME: PRINT " press <return> to change the"
910 PRINT " default drive to "; dv$(k%); " ..."
920 GET go$: IF go$ <> CHR$(13) THEN  RUN
930 HOME: PRINT " changing default drive ..."
935 POKE 29101, PEEK(29001)
940 CALL 29100: IF PEEK(27647) = 0 GOTO 1000
950 HOME: PRINT " write error on block zero!!": END
1000 HOME: PRINT " drive changed.": PRINT
1010 PRINT " press any key ..."
1020 GET go$: RUN
```

```
 10 REM patch to change the screen color for ShowOFF II
100 LOMEM :30000
110 DATA 62,4,1,0,0,17,2,0,33,0,108,205,243,252,50,255,107,201
120 FOR x = 29000 TO 29017: READ mc: POKE x, mc: NEXT
130 DATA 62,4,1,0,0,17,2,0,33,0,108,205,246,252,50,255,107,201
140 FOR x = 29100 TO 29117: READ mc: POKE x, mc: NEXT
200 TEXT: PRINT " This program allows you to set";
210 PRINT " the screen color for"
220 PRINT " ShowOFF II or ShowOFF IIa."
230 VTAB 10: PRINT " Which drive now contains"
240 PRINT " ShowOFF II": PRINT
250 PRINT " 1 = tape one": PRINT " 2 = disk one"
260 PRINT " 3 = exit program"
270 GET k$: k% = VAL(k$): IF k% < 1 OR k% > 2 GOTO 300
272 IF k% = 1 THEN  POKE 29001, 8
274 IF k% = 2 THEN  POKE 29001, 4
276 GOTO 500
300 TEXT: PRINT " program terminated.": END
500 HOME: PRINT " press <return> to read the"
510 PRINT " medium ..."
520 GET go$: IF go$ <> CHR$(13) THEN  RUN
525 HOME: PRINT " reading block two ..."
530 CALL 29000: IF PEEK(27647) = 128 GOTO 550
540 HOME: PRINT " read error on block two!": END
550 IF PEEK(27694) = 1 AND PEEK(27696) = 7 GOTO 600
560 HOME: PRINT " ShowOFF II(a) not detected!!": END
600 pc = PEEK(27695): HOME
700 PRINT " The current color value is "; pc: PRINT
710 PRINT " 1 = change color"
720 PRINT " 2 = restart this program"
740 GET k$: k% = VAL(k$): IF k% < 1 OR k% > 2 GOTO 300
750 IF k% = 2 THEN  RUN
800 HOME: PRINT " enter letter color?"
810 INPUT " (1 - 15): "; ca$: ca% = VAL(ca$)
820 IF ca% < 1 OR ca% > 15 GOTO 800
830 VTAB 10: PRINT " enter background color?"
840 INPUT " (1 - 15): "; cb$: cb% = VAL(cb$)
850 ON cb% = ca% GOTO 800: IF cb% < 1 OR cb% > 15 GOTO 830
860 nc% = ca%*16+cb%: POKE 27695, nc%
900 HOME: PRINT " press <return> to change the"
910 PRINT " default color to "; nc%; " ..."
920 GET go$: IF go$ <> CHR$(13) THEN  RUN
930 HOME: PRINT " changing default color ..."
935 POKE 29101, PEEK(29001)
940 CALL 29100: IF PEEK(27647) = 0 GOTO 1000
950 HOME: PRINT " write error on block two!!": END
1000 HOME: PRINT " color changed.": PRINT
1010 PRINT " press any key ..."
1020 GET go$: RUN
```

BANK SWITCHING

The Z80 can only access 64K (65536 bytes) of RAM at any one instant. A technique known as "bank switching" allows the programmer to shift 32K (32768 bytes) sections of memory in or out of direct access thereby permitting access to far more than 64K bytes. Each of these 32K sections is referred to as a "bank".

Bank switching should not be confused with accessing the 16K of video RAM. ADAM sends values to and receives values from the video chip as an input / output operation. This difference is more than simple semantics. The I/O operations between standard RAM and VRAM employ single byte transfers and an entirely different principle of micro-electronics is involved.

Back to bank switching ... there are two different types of banks, *ie*, low RAM and high RAM. The lower bank occupies addresses "0" thru "32767". The upper bank occupies addresses "32768" thru "65535".

The standard ADAM provides four lower banks and four upper banks. These eight banks equip ADAM with 16 possible memory combinations. To change banks, you just send an appropriate value to the bank switch port (127, $7F) -- there are 256 ports. Upon receiving the value, the specified bank is instantly switched in and the previous bank is switched out.

In actual practice, bank switching is a little complex. It involves juggling banks to prevent the system from crashing. There are several reasons for this. One of these is the NMI interrupt from the video chip which executes the routine at address 102 ($66). Another obstacle is that a system boot places the EOS in the upper bank of standard RAM. Still another is the location of the current stack. And, you must be certain that your z80 routine to access a bank is stored in the opposite end of RAM. If your routine is in a lower RAM bank, then you can only switch in an upper bank (and vice versa).

More next month ...

THE DIGITAL CLOCK PROGRAM

The program on the next page (page 17) creates a real-time clock for SmartBASIC V1.0. The display looks very similar to the connect time display for ADAMlink. The progam does have a handicap, though. Changing to GR, HGR, or HGR2 modes will stop the clock counter. Thus, you'll need to stay in TEXT mode for it to keep accurate time.

There are many possible applications for this program. You could just use it as a simple clock display. You could use it to sound an alarm tune at pre-determined intervals. You could also use it as a timer to determine how long you've been at the keyboard. And, with extra hardware accessed through a serial or parallel interface, you could use the clock counter to perform a variety of tasks. One of these could be to turn house lights on or off.

The BASIC program sets up two different machine code routines. The first of these (which occupies addresses 27606 thru 27699) controls the timer. The second one (which occupies addresses 27700 thru 27840) allows you to print the current time. When you use these routines, make certain that you don't overwrite these addresses with other routines.

Six addresses are reserved as a small data table for the counter. These are:

27600 = am/pm indicator
27601 = hour value
27602 = minute value
27603 = seconds value
27604 = vertical cursor position
27605 = horizontal cursor position

The process of keeping time is very simple. Asmb #49 (on page 18) details the z80 code of the routine. It justs increments the "seconds" value once every second. When the value is greater than 59, it increments the "minute" value and resets the "seconds" value. When the "minute" value is greater than 59, it increments the "hour" value and resets the "minute" value, and so on. A "1" is used to indicate "pm"; and, a "0" is used to indicate "am".

The program allows you to set the current time. When you want to see the time, you can CALL 27700. This may be a minor inconvenience. But, if you were to have the routine continually display the time, BASIC programming would be slowed considerably.

The default location for the display is in the upper, left-hand corner of the screen. You can set your own vertical position by POKEing a VTAB value into address 27007. And, you can set a new horizontal position by POKEing an HTAB value into address 27012.

The program even has an added bonus. It corrects the bug with BASIC's RND command. You now have true randomization.

If you use the program to sound an alarm periodically, you should PEEK addresses in the data table (addresses 27600 - 27603) to determine time values. Please let us know if come up with an interesting application for this one — we can pass it along to other readers.

```
 10 REM onscreen digital clock
100 LOMEM :28000: POKE 16149, 255: POKE 16150, 255: POKE 171, 201
110 HOME: PRINT " reading data ..."
200 DATA 245,197,213,229,221,229,221,33,208,107
210 DATA 221,52,3,62,59,221,190,3,48,46,221,54,3,0
220 DATA 221,52,2,62,59,221,190,2,48,32,221,54,2,0
230 DATA 221,52,1,62,12,221,190,1,48,18,221,54,1,1
240 DATA 221,52,0,62,1,221,190,0,48,4,221,54,0,0
245 DATA 42,64,63,35,34,64,63
250 DATA 221,225,225,209,193,241,201,0
255 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0
260 DATA 42,105,66,34,212,107,14,1,205,107,102,14,19,205,79,102
265 DATA 62,32,205,218,46,58,209,107,254,10,48,5,62,48,205,218,46
270 DATA 237,91,209,107,22,0,205,167,50
280 DATA 62,58,205,218,46
285 DATA 58,210,107,254,10,48,5,62,48,205,218,46
290 DATA 237,91,210,107,22,0,205,167,50
300 DATA 62,58,205,218,46
305 DATA 58,211,107,254,10,48,5,62,48,205,218,46
310 DATA 237,91,211,107,22,0,205,167,50
320 DATA 62,32,205,218,46
330 DATA 58,208,107,254,1,40,4,62,97,24,2,62,112,205,218,46
340 DATA 62,109,205,218,46,62,32,205,218,46
350 DATA 58,212,107,60,79,205,107,102,58,213,107,79,205,79,102,201
399 DATA -1
400 start = 27606: tot = 0
410 READ mc: IF mc = -1 GOTO 430
420 POKE start, mc: start = start+1: tot = tot+mc: GOTO 410
430 IF start = 27841 GOTO 460
440 IF start > 27841 THEN  PRINT " too many data entries!": END
450 PRINT " too FEW data entries!": END
460 IF tot = 23784 GOTO 1000
470 PRINT " incorrect data total!": END
1000 TEXT: PRINT " This program creates a real-"
1010 PRINT " time clock for ADAM in TEXT"
1020 PRINT " mode.  CALL 27700 to get": PRINT " display."
1030 VTAB 6: PRINT " am OR pm?"
1040 INPUT " (0=am, 1=pm): "; m$
1050 m% = VAL(m$): IF m% < 0 OR m% > 1 GOTO 1040
1060 POKE 27600, m%
1100 PRINT: INPUT " enter hour (1-12): "; ho$
1110 ho% = VAL(ho$): IF ho% < 1 OR ho% > 12 GOTO 1100
1120 POKE 27601, ho%
1200 PRINT: INPUT " enter minute (0-59): "; mn$
1210 mn% = VAL(mn$): IF mn% < 0 OR mn% > 59 GOTO 1200
1220 POKE 27602, mn%
1300 PRINT: INPUT " enter second (0-59): "; sc$
1310 sc% = VAL(sc$): IF sc% < 0 OR sc% > 59 GOTO 1300
1330 POKE 27603, sc%: POKE 171, 0
1340 POKE 172, 195: POKE 173, 214: POKE 174, 107: POKE 159, 60
1400 CALL 27700
```

TITLE (asmb#49):

DIGITAL CLOCK TIMER

```
addr:  Label:  Value(s):          Op Code:          Comment:

27606  setup   245               PUSH AF           ;store AF pair
27607          197               PUSH BC           ;store BC pair
27608          213               PUSH DE           ;store DE pair
27609          229               PUSH HL           ;store HL pair
27610          221,229           PUSH IX           ;store IX pair
27612          221, 33,208,107   LD   IX, 27600    ;set index pointer
27616  SecAlg  221, 52,  3       INC  (IX+3)       ;move ptr to 27603
27619           62, 59           LD   A,  59       ;set "second" limit
27621          221,190,  3       CP   (IX+3)       ;check current value
27624           48, 46           JR   NC, 46       ;if less, then RndPtc
27626          221, 54,  3,  0   LD   (IX+3), 0    ;reset "seconds"
27630  MinAlg  221, 52,  2       INC  (IX+2)       ;move ptr to 27602
27633           62, 59           LD   A,  59       ;set "minute" limit
27635          221,190,  2       CP   (IX+2)       ;check current value
27638           48, 32           JR   NC, 32       ;if less, then RndPtc
27640          221, 54,  2,  0   LD   (IX+2), 0    ;reset "minutes"
27644  HrAlg   221, 52,  1       INC  (IX+1)       ;move ptr to 27601
27647           62, 12           LD   A,  12       ;set "hours" limit
27649          221,190,  1       CP   (IX+1)       ;check current value
27652           48, 18           JR   NC, 18       ;if less, then RndPtc
27654          221, 54,  1,  1   LD   (IX+1), 1    ;restart "hours"
27658  PmAm    221, 52,  0       INC  (IX+0)       ;move ptr to 27600
27661           62,  1           LD   A,  1        ;check "pm" indicator
27663          221,190,  0       CP   (IX+0)       ;check current value
27666           48,  4           JR   NC, 4        ;if less, then RndPtc
27668          221, 54,  0,  0   LD   (IX+0), 0    ;set "am" indicator
27672  RndPtc   42, 64, 63       LD   HL, (16192)  ;get RND seed
27675           35               INC  HL           ;increment seed value
27676           34, 64, 63       LD   (16192), HL  ;reset RND seed
27679  DONE    221,225           POP  IX           ;retrieve IX pair
27681          225               POP  HL           ;retrieve HL pair
27682          209               POP  DE           ;retrieve DE pair
27683          193               POP  BC           ;retrieve BC pair
27684          241               POP  AF           ;retrieve AF pair
27685          201               RET               ;exit routine
```

MORE HGR COLORS

Have you ever wanted to be able to use more than ADAM's standard 15 colors? There is a simple trick known as "shading" that can give you limited access to a much wider choice of colors. The program on page 20 illustrates this technique.

As you'll see when you RUN the program, you can obtain various shades of gray, pink, red, green, blue, and yellow. The trick involves printing a small checkered pattern giving different color values to the background and foreground. For our example, the values "170" and "85" are used to create the checkered bit - map pattern. Any two whole integers whose sum equals 255 will work. The routine created with line numbers 140 thru 160 transfers the special bit - map to VRAM. Line numbers 170 and 180 write the color values to VRAM. The DATA on lines 200, 220, 240, and 245 just provide some pre - set color selections.

As is stands, this program is only a demonstration of the shading technique. With a little ingenuity, you could add some very impressive color combinations to your graphics screens using the same routines.

MORE ON EZcalendar

EZmenu (March issue), EZcopy (April issue), and EZcalendar (May issue) share many of the same routines. Here, we'll discuss some of these powerful graphics tricks.

Several months ago, we explored a very simple method of transferring the bit - maps of fonts to the graphics screen. Although this technique worked satisfactorily, it provided rather slow processing when printing a string (several consecutive characters) in the graphics window. To speed up the calculations and bit manipulations, we later created a couple of z80 routines. With these, you can print in the HGR window as fast as BASIC prints on the TEXT screen.

The routine for printing the strings (line numbers 220 thru 280) occupies addresses 28706 thru 28796. It is detailed in ASMB#51 on the bottom of page 21. For ease of use, the routine requires a data table to keep track of the assorted values. We set aside addresses 65517 through 65535 for this data table.

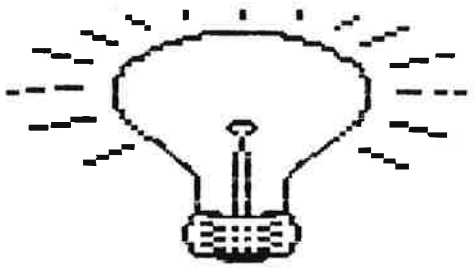
The data table is arranged as follows:

65535 = disk two status
65534 = disk one status
65533 = tape two status
65532 = tape one status
65531 = destination drive
65530 = source/current drive
65529 = high byte of font table pointer
65528 = low byte of font table pointer
65527 = nothing -- keep at zero
65526 = repeat counter value
65525 = VTAB position
65524 = HTAB position
65523 = high byte of ASCII pointer
65522 = low byte of ASCII pointer
65521 = color value
65520 = nothing -- keep at zero
65519 = offset for get bit-map bytes
65518 = last block to copy
65517 = first block to copy

The routine from line numbers 30000 thru 30060 POKE the values into the data table. It also puts the ASCII string in RAM and CALLs the z80 routine to print a string inside the graphics window.

Asmb #50 (at the top of page 21) doubles the length of the bit-mapped font table. This allows a program to quickly print double length characters.

We'll go into more details on the various routines in upcoming similar programs ...



```
 1Ø REM demo program to show 255 HGR colors!!!
1ØØ POKE 16149, 255: POKE 16150, 255: LOMEM :28ØØØ
11Ø POKE 25431, 1: POKE 25471, 25: POKE 25568, 241: HGR
12Ø FOR x = 22Ø*256 TO 221*256-1 STEP 2
13Ø POKE x, 17Ø: POKE x+1, 85: NEXT
14Ø DATA 17,Ø,32,1,Ø,1,33,Ø,22Ø,2Ø5,26,253,2Ø1
15Ø FOR x = 27ØØØ TO 27612: READ mc: POKE x, mc: NEXT
16Ø FOR x = 32 TO 52: POKE 27602, x: CALL 27ØØØ: NEXT
17Ø DATA 62,Ø,33,Ø,Ø,17,Ø,1,2Ø5,38,253,2Ø1
18Ø FOR x = 27622Ø TO 27631: READ mc: POKE x, mc: NEXT
19Ø z = 19: DIM g(z), b(z), r(z)
2ØØ DATA 58,59,51,5Ø,6Ø,35,34,44,2Ø4,33,193,49,199,39,55,2Ø3,2Ø2,43,42,198
21Ø FOR x = Ø TO z: READ g(x): NEXT
22Ø DATA 94,95,78,79,91,92,74,75,87,71,85,84,68,81,65,7Ø,86,77,93,76
23Ø FOR x = Ø TO z: READ b(x): NEXT
24Ø DATA 158,142,11Ø,153,137,1Ø5,152,1Ø2,136,134,1ØØ,1Ø2,145,129,155
245 DATA 139,1Ø7,157,141,1Ø9
25Ø FOR x = Ø TO z: READ r(x): NEXT
3ØØ HOME: PRINT " 1 = define color by row"
31Ø PRINT " 2 = preset colors": PRINT " 3 = exit"
32Ø VTAB 23: HTAB 1: GET m$: m% = VAL(m$)
33Ø ON m% < 1 OR m% > 3 GOTO 32Ø: HOME: ON m% GOTO 4ØØ, 6ØØ, 35Ø
35Ø PRINT " program terminated.": END
4ØØ INPUT " enter color (Ø - 255): "; co$: co% = VAL(co$)
41Ø ON co% < Ø OR co% > 255 GOTO 4ØØ: HOME
5ØØ INPUT " enter row (1 - 2Ø): "; ro$: ro% = VAL(ro$)
51Ø IF ro% < 1 OR ro% > 2Ø GOTO 5ØØ
52Ø POKE 27621, co%: POKE 27624, ro%-1: CALL 27622Ø: GOTO 3ØØ
6ØØ PRINT " 1 = green": PRINT " 2 = blue": PRINT " 3 = red"
61Ø VTAB 23: HTAB 1: GET m$: m% = VAL(m$)
62Ø FOR x = Ø TO z: POKE 27624, x
63Ø IF m% = 1 THEN  co% = g(x)
64Ø IF m% = 2 THEN  co% = b(x)
65Ø IF m% = 3 THEN  co% = r(x)
66Ø POKE 27621, co%: CALL 27622Ø: NEXT
67Ø HOME: PRINT " press any key for menu ..."
68Ø GET go$: GOTO 3ØØ
```

TITLE (asmb#50):

Double Bit-Map Length

```
addr:  Label:  Value(s):      Op Code:        Comment:

28685  setup    33,  0,108    LD   HL, 27648  ;set source pointer
28688           17,  0,212    LD   DE, 54272  ;set destination pointer
28691            1,  0,  4    LD   BC, 1024   ;set byte count
28694  loop     126           LD   A,  (HL)   ;get bit-map value
28695           18            LD   (DE), A    ;xfer value to dest. adr.
28696           19            INC  DE         ;increment dest. address
28697           18            LD   (DE), A    ;repeat value xfer
28698           19            INC  DE         ;set up for next loop
28699           35            INC  HL         ;continue set up
28700  check    11            DEC  BC         ;see if done
28701          120            LD   A,  B      ;continue check
28702          177            OR   C          ;adjust flag register
28703           32,245        JR   NZ, 245    ;if not done, then "loop"
28705  exit    201            RET             ;exit routine
```

TITLE (asmb#51):

Print Bit-Mapped String

```
addr:  Label:  Value(s):          Op Code:          Comment:
28706  setup   237, 91,242,255   LD   DE, (65522)  ;set pointer to string
28710  LOOP     26               LD   A,  (DE)     ;get ASCII value
28711  check   254,  0           CP   0            ;check if done
28713          200               RET  Z            ;if zero, then exit
28714          254, 13           CP   13           ;check if done
28716          200               RET  Z            ;if <CR>, then exit
28717  mult1   245               PUSH AF           ;store accumulator
28718           58,239,255       LD   A,  (65519)  ;get bitmap offset
28721           71               LD   B,  A        ;xfer offset to "B"
28722          241               POP  A            ;retrieve ASCII
28723           33,  0,  0       LD   HL, 0        ;reset HL pair
28726           95               LD   E,  A        ;prepare multiplier
28727           22,  0           LD   D,  0        ;continue prep
28729  mloop1   25               ADD  HL, DE       ;repeat adding
28730           16,253           DJNZ 253          ;if not done, then mloop1
28732  getptr  237, 75,248,255   LD   BC, (65528)  ;get start of bit-maps
28736            9               ADD  HL, BC       ;compute font location
28737  mult2    58,246,255       LD   A,  (65526)  ;get repeat factor
28740           71               LD   B,  A        ;xfer factor to "B"
28741           58,244,255       LD   A,  (65524)  ;get HTAB position
28744           79               LD   C,  A        ;xfer HTAB to "C"
28745          175               XOR  A            ;reset accumulator
28746  mloop2  129               ADD  A,  C        ;repeat adding
28747           16,253           DJNZ 253          ;if not done then mloop2
28749  VRMpos   95               LD   E,  A        ;store horz scrn pos
28750           58,245,255       LD   A,  (65525)  ;get VTAB position
28753           61               DEC  A            ;correct VTAB
28754          198, 32           ADD  A,  32       ;offset for pattern table
28756           87               LD   D,  A        ;complete VRAM position
28757          237, 75,246,255   LD   BC, (65526)  ;get repeat factor
28761          213               PUSH DE           ;store VRAM address
28762          205, 26,253       CALL 64794        ;EOS write to VRAM
28765          209               POP  DE           ;retrieve VRAM address
28766  VRMcol  107               LD   L,  E        ;xfer DE to HL
28767          122               LD   A,  D        ;prep for subtract
28768          214, 32           SUB  A,  32       ;change to color address
28770          103               LD   H,  A        ;conclude xfer DE to HL
28771           58,241,255       LD   A,  (65521)  ;get color value
28774          237, 91,246,255   LD   DE, (65526)  ;get repeat factor
28778          205, 38,253       CALL 64806        ;EOS fill VRAM (color)
28781  prep     58,244,255       LD   A, (65524)   ;get HTAB position
28784           60               INC  A            ;increment HTAB
28785           50,244,255       LD   (65524), A   ;store new HTAB
28788           42,242,255       LD   HL, (65522)  ;get crnt strng ptr
28791           35               INC  HL           ;increment pointer value
28792           34,242,255       LD   (65522), HL  ;store new ptr value
28795  repeat   24,165           JR   165          ;goto LOOP
```

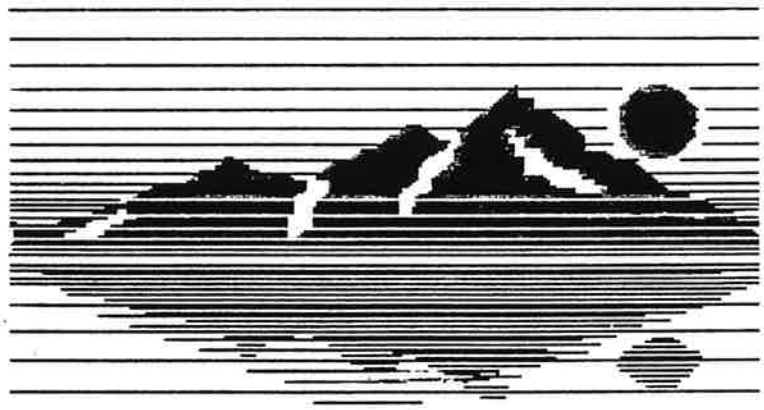
HACKER'S CONTEST #8

Congratulations to Guy Cousineau of Ottawa, Canada as the winner of Hacker's Contest #7.

The "NIBBLES & BITS" Hacker's Contest is a bi - monthly competition. The winner of each contest is randomly selected from the correct responses postmarked within the specified dates. No individual shall be named the winner for more than two consecutive contests. The winner of each contest shall be awarded a $25.00 purchase credit with DIGITAL EXPRESS and given a FREE three month extension to his / her "NIBBLES & BITS" subscription term. Decisions of the judges are final.

Responses for this particular contest shall be considered valid if, and only if, they are postmarked after June 1, 1987 and prior to July 31, 1987. The winner shall be announced in the August issue of "NIBBLES & BITS".

The challenge: Write a SmartBASIC V1.0 program (it may include machine code in DATA statements) which will scroll an HGR screen vertically up or down.



PRODUCT:	ProofREADER
MANUFACTURER:	Strategic Software
MEDIA TYPE:	data pack
GRAPHICS/SOUND/DESIGN:	n/a
INSTRUCTIONS:	95
USEFULNESS vs. PRICE:	80
RECOMMENDATION:	recommended
PRICE:	34.95 (M.W. Ruth)
RATED BY:	Randal Bondi Allison Park, PA

ProofREADER is a spelling checker for SmartWRITER, MultiWRITE or any other ASCII files. Unlike their earlier attempt with SmartSPELLER, it is written entirely in machine language and checks the file very quickly.

The file you are checking must be on a tape with a one block directory. When ProofREADER encounters a word in your file that is not in its dictionary, it displays it on the screen and lists several suggested spellings and gives you the choice of using one of them. You can enter the correct spelling yourself or if it is correct, you can leave it unchanged and add it to the dictionary.

The dictionary that comes with it contains about 5500 words; and, you may add words to it while checking a document, or by typing them in manually, or from a data file. The dictionary will hold a total of about 7000 words. When you add words to the dictionary you must save the new dictionary on a tape other than ProofREADER. When it finishes checking the file, it tells you how many words the file contains, how many you changed, and saves a corrected version on tape.

My biggest complaint is that when you are checking the file, it displays the incorrect word out of context. This makes it hard to tell what some typos were supposed to be. Overall, I think this is a very good program and I believe it would be a nice addition to anyone's software library -- especially if your spelling is as poor as mine is (and you can live with a small dictionary).

PRODUCT:	TurboDISK 1.0
MANUFACTURER:	DIGITAL EXPRESS
MEDIA TYPE:	data pack/disk
GRAPHICS/SOUND/DESIGN:	95/95/99
INSTRUCTIONS:	95
USEFULNESS vs. PRICE:	100
RECOMMENDATION:	higly recommended
PRICE:	19.95
RATED BY:	D.L. Decker President of D.L. DECKER ENTERPRISES

TurboDISK 1.0 is an excellently written ADAM utility which will allow you to use the 64K expansion card (sold separately) as a fast disk drive. TurboDISK 1.0 allows the user to turn the 64K card into a "RAM DISK", which permits incredibly fast storage and retrieval of programs while using SmartBASIC V1.0. One nice thing about the program is that unlike other ADAM utilities which ignore various BASIC commands, this one makes full use of ALL SmartBASIC V1.0 commands. In addition, it is compatible with other DIGITAL EXPRESS products such as Intel-LOAD V1.0, Intel-BEST 3.3, and the PatchWORK series from N&B. Also, TurboDISK 1.0 includes a copy program called "TurboCOPY" and a BASIC improvement program called "EZkeysII".

The documentation provided with this software describes all the functions in a precise manner. "TurboCOPY" is an excellent back-up program which uses a 62K buffer! Several programs were duplicated using the utility and ALL worked flawlessly. The great aspect about using the 64K expander in backup work is that fewer media swaps are necessary. For instance, when copying disks using a single drive, only 3 source / destination media swaps were required. Furthermore, "TurboCOPY" allows the user to store it to the ramdisk for later usage. Another nice feature was the CATALOG, INIT, RENAME, and DELETE functions. Also, unlike other copy programs, "TurboCOPY" was written using HI-RES graphics, SmartKEYS, and keyclicks.

If this sounds like a "sales pitch", I guess it is. This package is one which every ADAM owner with the 64K expansion card shouldn't be without. The really appealing fact about this software is the price -- $19.95 for a ram disk ability from BASIC and a 62K buffer backup program. In conclusion, I highly recommend this program due to its HIGH QUALITY & LOW PRICE.

ADAM USERS' GROUPS

OREGON

Portland ADAM Users
Craig Frerichs
P.O. Box 1081
Portland, OR 97207

PENNSYLVANIA

The (717) ADAM Users
Steve Chamberlain
120 East 4th Street
Bloomsburg, PA 17815

TENNESSEE

Midsouth ADAM Users
Roger Burford
Lot 142, NAS MHP
Millington, TN 38053

TEXAS

gHAAUG
Terry Fowler
P.O. Box 800801
Houston, TX 77280-0801

ADAM Users of EL Paso
Dick Lewin
6308 Falling Star
El Paso, TX 79912

VIRGINIA

Central Virginia AUG
Thomas J. Kelly
Route 664, 3B
Earlysville, VA 22936

Norfolk ADAM Group
Gerald M. Steen
1000 Rockbridge Avenue, #144
Norfolk, VA 25308

Washington D.C. AUG
Jim Tyson
1811 St. Roman Drive
Vienna, VA 22180

BULLETIN BOARD

- interested in meeting other ADAM users -
Dave McIntosh
7 Monsarrat Crescent
London, Ontario
Canada N5Y 4Y7

Song Files for VideoTunes
Send SASE to:
VideoSongs
Box 151
Harmony, IN 47853

ADAM Software
FutureVision
P.O. Box 34
North Bellerica, MA 01862

ADAM Software and Hardware
ALPHA - 1
1671 East 16th Street, Suite 146
Brooklyn, NY 11229

PRINTING SERVICES - NEWSLETTERS, ETC.
Ted Gocal, Gannon University
121 West 7th Street
Erie, PA 16541
Compuserve ID#: 75226, 226

ADAM Software
Murdock Games
2645 Wilbanks Circle
Dallas, GA 30132

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

ADAM Software
MARATHON COMPUTER PRESS
P.O. Box 68503
Virginia Beach, VA 23455

- magazine -
COMPUTER SHOPPER
P.O. Box 1419
Titusville, FL 32781-9988

computer peripherals
BTE Computers, Inc.
14644 North Cave Creek, #6
Phoenix, AZ 85022
(602) 867 - 8962

Introducing . . .
Font POWER

"MICRO" FONTS
ABCDEFGHIJKLMNOPQRSTUVWXYZ
ABCDEFGHIJKLMNOPQRSTUVWXYZ
0123456789

"roman" fonts
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
0123456789

"cozy" fonts
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
0123456789

"script" fonts
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
0123456789

"bold" fonts
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
0123456789

Font POWER

© 1987 by DIGITAL EXPRESS

for use with

ADAM™

THE COLECOVISION
FAMILY COMPUTER SYSTEM

FontPOWER is the latest release from DIGITAL EXPRESS. Its use of graphics and sound is so sophisticated that we've dubbed its release as the beginning of our second generation of ADAM software. You'll see SmartKEYS at the bottom of the screen just like Coleco packages. You'll hear Coleco-like sound routines. You'll see the directory of files displayed on a graphic file folder just like Coleco software. Almost instantly after pulling the reset switch, you'll see a detailed graphic title screen just like the Coleco packages. And so much more. The package is so professional that you may be tempted to think that it is a Coleco program.

Enough about the embellishments, what will FontPOWER do for you? The main program is a very user - friendly utility for designing your own font sets. The font design grid looks just like the SmartLOGO sprite design grid. Never before has creating your own character sets been so easy on ADAM.

FontPOWER comes complete with EIGHT new font sets. Take a look at the previous page (page 25) to see the quality of these characters. You can use any set in your own programs -- a very impressive touch. And, the set you select is stored above the BASIC interpreter so that there is no overhead for your own programs. You can even have the INVERSE characters with the default font design and the NORMAL characters with the new set of your choice.

But, there's more. A simple routine is included that will let you use any of the sets in your own HGR, HGR2, or GR screens -- put (eye - catching) text right next to your graphics. Plus, you even get three different 'shape tables' of font sets for special text manipulation in your graphics, such as ROTating characters.

FontPOWER comes with a detailed, easy - to - understand user's guide. In no time at all you can add a very distinctive new touch to your own programs. How much is the powerful package? ONLY $12.95 to "NIBBLES & BITS" subscribers. And, just $16.95 to non - subscribers. Get FontPOWER today -- you've never seen a third party program for ADAM that so closely emulates Coleco's design techniques.

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 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

☐☐☐ BASICaide (rev2) (Mr. T. Software)
* several SmartBASIC 1.0 enhancements including a new "CHAIN" command for merging programs and a new "BIN" command that executes the built-in function for converting SmartBASIC 1.0 programs to LOAD up to 12 times faster

>>> $11.95 (each) for non-subscribers
>>> $9.95 (each) for N&B subscribers

☐☐☐ TurboDISK 1.0 (by DIGITAL EXPRESS)
* creates a ramdisk ability from SmartBASIC V1.0; corrects INIT blocks and BSAVE short buffer; includes TurboCOPY -- a utility for controlling files and copying media with a 62K copy buffer

>>> $24.95 (each) for non-subscribers
>>> $19.95 (each) for N&B subscribers

☐☐☐ FontPOWER (by DIGITAL EXPRESS)
* utility using Coleco-like graphics for designing your own font sets; 8 font sets including "script", "roman", "cory", & "bold"; shows you how to use font sets in high or low resolution graphics; plus three font shape tables for use in HGR or HGR2 mode

>>> $16.95 (each) for non-subscribers
>>> $12.95 (each) for N&B subscribers

☐☐☐ MegaUtil (by MARATHON COMPUTER PRESS)
* an excellent collection of varied programming aids; includes ByteWriter (block editor), CopyWriter (media back-up utility), PD modules, programming tips, more +++

>>> $32.95 (each) for non-subscribers
>>> $27.95 (each) for N&B subscribers

☐☐☐ TurboDISK 2.0 (by DIGITAL EXPRESS)
* creates a powerful ramdisk ability for SmartBASIC 2.0

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

RECREATION/GAMES SOFTWARE

☐☐☐ MageQuest (rev 2) (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

☐☐☐ Lab Mouse (by REEDY SOFTWARE)
* exciting game that puts you in the role of a laboratory mouse stuck in a maze; all hi-res graphics; five skill levels; auto-loading

>>> $13.95 (each) for non-subscribers
>>> $11.95 (each) for N&B subscribers

☐☐☐ Entertainment Pack (by REEDY SOFTWARE)
* three challenging computer classics (connect 4, blockade, and slide puzzle); great graphics; fast animated sprites; one or two players

>>> $16.95 (each) for non-subscribers
>>> $14.95 (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 UTILITY SOFTWARE

☐☐☐ ShowOFF I (by DIGITAL EXPRESS)
* self-booting graphics design package (enter text, draw polygons, save pictures, etc.) with a variety of print options (preset for Epson FX / IBM 5152 printer codes); printing graphics requires a Centronics parallel interface for printer

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

☐☐☐ ShowOFF II (by DIGITAL EXPRESS)
* machine code print enhancements for SmartWriter (adds CONTROL features to SmartWriter) and SmartBASIC; requires Centronics parallel interface, a Panasonic KX 1080 or 1080i printer, and a 64K expander

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

☐☐☐ ShowOFF IIa (by DIGITAL EXPRESS)
* very similar to ShowOFF II except that it is compatible with any dot matrix printer that supports EPSON FX escape codes; works with the EPSON and STAR line of printers and the Okimate 20; does not include line justification commands or internal document margin control

>>> $19.95 (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
* set 04: all the programs from 04/87 thru 06/87

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

COLECO COPYRIGHTED SOFTWARE

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

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

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

>>> $17.95 (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
* Sony brand 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
* black ink, just like the one that came with your ADAM

>>> $5.75 (each) for non-subscribers
$15.95 (for 3) for non-subscribers
>>> $5.25 (each) for N&B subscribers
$14.75 (for 3) for N&B subscribers

□□□ Panasonic printer ribbons
* black ink, nylon, approximately one million characters, fits these models: 1080, 1080i, 1090, 1091, 1091i, and 1092

>>> $6.95 (each) for non-subscribers
>>> $5.45 (each) for N&B subscribers

□□□ multipurpose adhesive labels
* white, tractor feed, $3\frac{1}{2} \times \frac{15}{16}$, 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

□□□ word processing computer paper
* white, tractor feed, $9\frac{1}{2} \times 11$, fan fold, 20 lb. wt., clean edge, one part, single column

>>> $4.25 (250 sheets) for non-subscribers
>>> $3.45 (250 sheets) for N&B subscribers

EDUCATIONAL SOFTWARE

□□□ The Spanish Vocabularian
(by MARATHON COMPUTER PRESS)
* a unique program for ADAM; includes electronic dictionary; includes 1600 words; expandable to 7400 words; quizzes; printed study sheets; report cards

>>> $18.50 (each) for non-subscribers
>>> $15.95 (each) for N&B subscribers

□□□ Quikfax Quest (by DIGITAL EXPRESS)
* 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

■■■■■■■■■■

□□□□□ 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 August 15, 1987.

■■■■■■■■■■

DEI Public Domain Facts

You may get any of the volumes described below on DATA PACK or DISK for ONLY $5.95 as an 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 calendar 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.

Public domain software is offered as a quick, inexpensive means for you to expand your ADAM software library. Note, however, that public domain software is not necessarily of commercial quality. Although we do attempt to winnow out flawed programs, there is no guarantee of quality regarding these packages.

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 -- 154K of 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 -- 108K of files.

SmartPAINT Files 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&Bpix001": 13 different HGR picture files.
"N&Bpix002": 13 different HGR picture files.
"N&Bpix003": 13 different HGR picture files.
"N&Bpix004": 13 different HGR picture files.
"N&Bpix005": 13 different HGR picture files.

Coleco Unreleased Titles LIBRARY

"SmartBASIC 2.0": Improved interpreter; 49K program; works with or without the 64K expander; includes new commands STDMEM, EXTMEM, MERGE; plus more ...

"Pinball Construction/Hardhat Mac": Best of Electronic Arts; latest version with two demo pinball games; 1 to 4 players with Pinball Construction; one or two players with Hardhat Mac.

"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 this 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 the 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 or may be used with the Pinball Construction Set.

"N&B-PBgames01": 10 pinball games.

"N&B-PBgames02": 10 pinball games.

Miscellaneous Collections LIBRARY

"MWplus01": A collection of improvements to MultiWrite by Strategic Software. Requires Multiwrite software. Written by Jim Guenzel.

"N&Bacalc01": several paradigm and other files stored in ADAMcalc format; contributed by Terry Fowler; 148K of files.