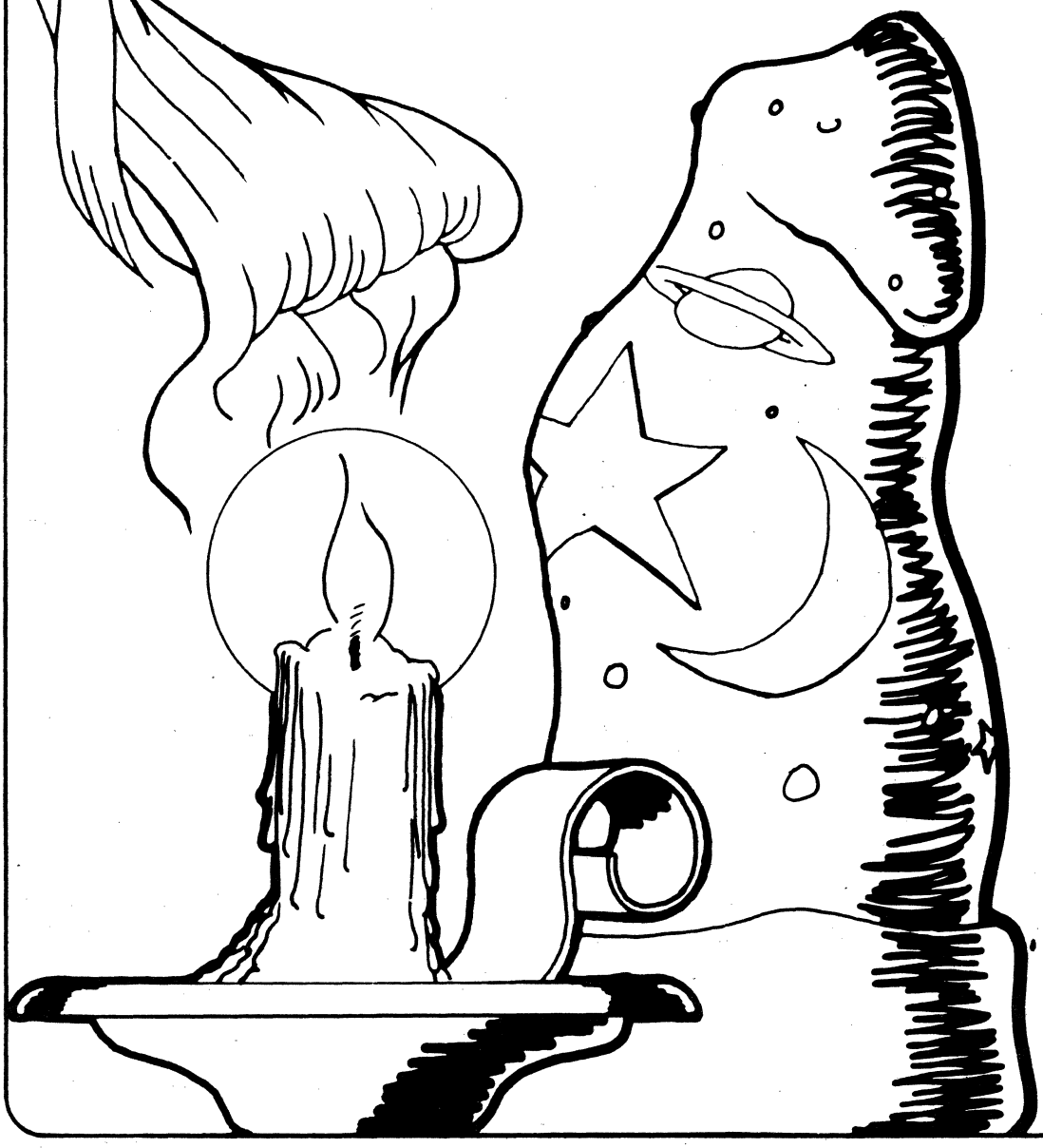
ARESCO Sorcerer's Source



VOL 1 ISSUE 3 DEC 1979 $2.00

SUBSCRIPTIONS, ADVERTISING RATES, AND OTHER SUCH INFORMATION

The SOURCE is an international User Group Newsletter dedicated exclusively to owners of the Exidy Sorcerer personal Computer. The Source will be published at irregular intervals, depending on the material sent in by subscribers for publication.

The SOURCE is published by ARESCO, Inc., 9143G Red Branch Road, Columbia MD 21045. All inquiries and correspondence should be sent to the postoffice box address rather than to the street address: ARESCO SOURCE, Box 1142, Columbia MD 21044.

POSTMASTER: Please send all address changes to the SOURCE at the above post office box.

Editor: Terry Laudereau
Co-Editor: Don Despres

Readers are encouraged to submit articles of general interest to other Sorcerer owners. Material submitted will be considered free of copyright, and all material in the publication is copyrighted by ARESCO. User Groups may circulate a copy, but may not make copies for other members of the group.

Material submitted for publication should be typewritten and single spaced, using new ribbon. If no typewriter is available, we can re-type your handwritten material, but we cannot be responsible for typographical errors which will, in spite of our best efforts, occasionally creep in.

SUBSCRIPTION RATES

The SOURCE is mailed 3rd class, and no charge is made for postage. Subscription rate is $15/10 issues, and all ten issues will be part of the same volume. Subscriptions do not overlap from one volume to another.

USA residents who wish to have their copies sent via first class mail may include $5/vol. for postage. Non-USA residents should include $10/yr for first class postage, $12.00/year for air-mail postage if desired.

Payment for subscriptions may be made in the form of cash, check, money order (all in US funds and drawn on a US bank), Master Charge, and VISA credit cards. No billing or COD is available, and purchase orders are not accepted unless accompanied by payment.

ADVERTISING RATES

Classified ads:	$10/3 lines
Full page display:	$85.
Half page display:	$45.
Quarter page disp:	$25

Payment must accompany camera-ready copy unless other arrangements have been made in advance.

DEALER RATES

Dealers may obtain copies of the ARESCO SOURCE at our cost of $1.20/copy plus shipping. Minimum order is 5 copies per order. All orders will be shipped either pre-paid or COD unless other arrangements have been made in advance.

INFORMATION REQUESTS

ARESCO will provide a free introductory copy of the SOURCE for interested readers, provided a stamped, self addressed, 6 x 9 envelope is provided (with 28¢ postage attached is included with the request.

TABLE OF CONTENTS

Volume I * Issue 3

December 1979

SOURCE

SOURCE

USING THE MONITOR COMMANDS FROM A BASIC PROGRAM

by Ian MacMillan
(Courtesy of the Australian Sorcerer Users Group - Thanks!)

Although the Exidy Monitor is quite a large program; of the order of 4K Bytes; it does not fully qualify as a "Monitor", having no provisions for examining the contents of the CPU registers, for example. However it is an extremely useful collection of Z80 routines, including, of course, the operating system. Actually, the whole thing is almost a kind of high level language interpreter, and it is in fact possible to run "monitor" programs consisting of sequences of "batched" Monitor Commands which have been recorded on tape.

Several of the facilities provided by the Monitor are of great value in normal programs, and if the program is written in Z80 language can be directly accessed by simply calling the routines.

Most users dabble inZ80 code only in a very small way, as the difficulties are considerable, especially in the absence of an editor/assembler (Yes,I know...),however,it is possible to use the routines from BASIC using the following technique.

The monitor command with it's associated addresses is directly POKEd into the Monitor Buffer. This is located from 8082 up in an 8K machine (16274 and 32658 in 16K and 32K). The address of the particular routine is entered into addresses 260 and 261, and the thing is set off by entering X=USR(0).

The program listing shows a MOve function implemented in BASIC but although it describes how to write the Data line using the character/ASCII table on page G1 in the "Short Tour of Basic", it fails to point out that the addresses POKEd in line 30 are specific to the MOve command, and have to be different for a different command.

The most useful "goto" addresses are listed below:

COMMAND	HEX ADDR	260	261
MOve	E562	98	229
DUmp	E4D3	211	228
SAve	E638	56	230
LOad	E7BA	186	231

The Monitor SET commands can be implenented more simply, with a simple POKE instruction. The use of these in a BASIC program can enable variable printing speeds, changes in current input and/or output ports,and I/O Baud rate, as the program runs, and a listing of the POKE addresses involved will be published in a separate article.

```
 1 REM   ******Program by IAN MACMILLAN******
 2 :
 3 REM    Demonstration of Monitor use from BASIC
 4 :
 5 REM  This shifts the top half of the screen to the bottom
 6 REM  half by using the MOve routine in the Monitor.  this
 7 REM  is done by poking MO F440 F443 into the monitor
 8 REM  buffer (Decimal coded in line 15), then poking the low
 9 REM  and high bytes of the MOve routine address into 260,261
10 REM  which is where the jump address for USR(0) is located.
11 REM  The monitor routines all end with C9, so the stack
12 REM  ensures a return to BASIC without special provision.
13 :
14 :
15 DATA 77,79,32,70,48,56,51,32,70,52,51,70,32,70,52,52,51,13
18 Z=32657: REM  for 8K Z=8081, for 16K Z=16273
20 FOR X=1 TO 18: READ Y: POKE(Z+X),Y:NEXT X
25 INPUT "Are you ready to move"; A$
30 POKE 260,98: POKE 261,229: V=USR(0)
40 :
60 REM  Any monitor command can be entered this way - just look
70 REM  up the Decimal Code for each character on page G1 of
80 REM  your"Short Tour of Basic" and make up line 15 accordingly
```

DO YOU SUFFER FROM UNWANTED SCROLLING?

by IM

(Courtesy of Australian Sorcerer Users Group - Thanks!)

Have you ever written a program using graphics with lots of INPUT statements? If you did you doubtless found that when the INPUT statements reached the bottom of the screen, the display scrolls, and woops! There go the graphics!

The answer is to keep the input prompts and responses on the top line of the video display by putting in a line:

```
PRINT CHR$(17)
```

after each input. This "homes" the cursor without clearing anything,and allows you to POKE into the display indefinitely without fear of losing your precious picture.

◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆
◆ PRELIMINARY ANNOUNCEMENT ◆
◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆

THE MIDWEST AFFILIATION OF COMPUTER CLUBS (MACC)
IS PROUD TO ANNOUNCE THAT OUR
FIFTH ANNUAL COMPUTERFEST (R) WILL BE JUNE 20 -22, 1980

AT FRANKLIN UNIVERSITY, GRANT & RICH, IN DOWNTOWN COLUMBUS OHIO.
PLEASE MAKE A NOTE OF THAT DATE !

175 ROOMS HAVE BEEN RESERVED FOR COMPUTERFEST V AT HOLIDAY INN
NEAR I-70 & I-71 IN DOWNTOWN COLUMBUS NEAR FRANKLIN UNIVERSITY.

FOR MORE INFORMATION, CONTACT -
JAMES CROWLEY 4008 RICKENBACKER AVE COLUMBUS OH 43213
CONVCHAIR (VIA INFOPLEX)
614-239-9931 (PHONE RINGS AT BUSINESS AND HOME)

◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆
◆ CALL FOR PARTICIPATION ◆
◆◆
◆ IN THE 5TH ANNUAL COMPUTERFEST JUNE 20 - 22, 1980 ◆
◆ ISSUED BY THE 1980 STEERING COMMITTEE ◆
◆◆

EVERYONE INTERESTED IN THE PERSONAL COMPUTING HOBBY OR
INVOLVED IN THE SMALL BUSINESS COMPUTER FIELD, IS INVITED TO :

LIST EVENTS YOU EXPECT US TO INCLUDE, OR VOLUNTEER AS A PANELIST.

PROVIDE THE NAMES & ADDRESSES OF CONTACTS FOR EACH PERSONAL
COMPUTER CLUB WHO WILL GET COMPUTERFEST INFORMATION TO THE MEMBERSHIP
AS SOON AS IT IS RELEASED, & ACT AS A LIASON FOR TWO-WAY INFO FLOW.

MORE PARTICIPANTS ARE NEEDED FOR THE STEERING COMMITTEE.
IDEALLY YOU SHOULD HAVE ACCESS TO A PERSONAL COMPUTER WITH A MODEM,
AND YOUR CLUB OUGHT TO JOIN MACC SO WE CAN GIVE YOU THE ACCESS CODES FOR
THE FREE INFOPLEX ELECTRONIC MAIL, DONATED BY COMPUSERVE TO MACC.
IF YOUR CLUB IS INTERESTED IN JOINING MACC, OR SEEKS MORE INFO ON MACC,
CONTACT DOUG TROUGHTON 508 COLONY DR WHEELERSBURG OH 45694 614-574-8152
PRESMACC (VIA INFOPLEX)

MORE PARTICIPANTS ARE DESIRED FOR THE FACILITIES MANAGEMENT
& HOST COMMITTEE, ORGANIZED BY THE AMATEUR COMPUTER SOCIETY
OF COLUMBUS OHIO (ACSCO) C/O RUSS RANSHAW 3586 SMILEY RD
HILLIARD OH 43026 614-876-9192 TRUSACSCO (VIA INFOPLEX) 70010,100 (MICRONET)

VOLUNTEERS ARE NEEDED TO ORGANIZE PUBLICITY & PROMOTION FOR
COMPUTERFEST 80. UNTIL A PENDING CONVERSION AT WORK TAKES AWAY
MY TIME, I'M ACTING AS TEMPORARY PUBLICITY CHAIRMAN.
WE EXPECT TO HAVE MORE VOLUNTEERS INVOLVED BEFORE I MUST DROP-OUT.
AL MACINTYRE 6750 SHENDANDOAH APT 5 FLORENCE KY 41042 606-525-6442
EDITOR FOR CINCINNATI COMPUTER CLUB (CCC).
EDCCC (VIA INFOPLEX) 70020,100 (MICRONET)

VOLUNTEERS ARE INVITED TO SENT SHORT ENTERTAINING ARTICLES TO THE MACC COMPUTERFEST PROMOTIONAL NEWSPAPER WHICH WILL BE EDITED, PRINTED, AND DISTRIBUTED BY THE SOUTHEASTERN MICHIGAN COMPUTER ORGANIZATION (SEMCO) C/O TERRY WILLIAMS 313-852-6546 1850 BROOKVIEW CIRCLE, BLOOMFIELD HILLS MI 48013
SEMCO HAS NUMEROUS ELECTRONIC BULLETIN BOARDS (CBBS) WHICH ARE LISTED IN THE SEPT 79 ISSUE OF THEIR NEWSLETTER (DATA BUS), OR SEMCO CAN ALSO BE CONTACTED BY TRUSSEMCO (VIA INFOPLEX)

VOLUNTEERS ARE ALSO NEEDED TO GET THE FOLLOWING COMMITTEES GOING :
(1) SOLICIT SPEAKERS & WORKSHOPS, AND SCHEDULE THE PROGRAM
(2) SOLICIT EXHIBITORS & ADVERTISING FOR MACC'S PROMOTIONAL NEWSPAPER, & THE COMPUTERFEST PROGRAM.
(3) BUDGETING COMMITTEE TO OVERSEE FINANCIAL ACCOUNTING
(4) CO-ORDINATE PRE-REGISTRATION & HOTEL RESERVATIONS & PRINT TICKETS.

MORE INFORMATION ABOUT COMPUTERFEST 80

COMPUTERFEST IS MORE THAN COMPUTERS AND COMPONENTS, SOFTWARE AND HARDWARE
IT IS A GATHERING AND DISSEMINATION OF IDEAS, WITH WORKSHOPS & PANEL DISCUSSIONS ON A WIDE VARIETY OF HARDWARE INTERESTS, COMPUTER LANGUAGES, AND COMPUTER APPLICATIONS, WHERE HOBBY SHOPS, HOBBYISTS, AND SMALL BUSINESSPEOPLE WHO HAVE MUCH IN COMMON, CAN SHARE INFORMATION.

LOUISVILLE'S COMPUTERFEST 79 HAD OVER 40 WORKSHOPS, SEMINARS, AND PANEL DISCUSSIONS, SUCH AS :

COMPUTER AIDED INSTRUCTION ; COMPUTER AIDS FOR THE HANDICAPPED : COMPUTER DRIVEN MUSEUM DISPLAYS ; ELECTRONIC NEWSPAPERS ; ETHICS ; HAM RADIO INTERFACES ; INVENTORY CONTROL ; ROBOTICS ; SIMULATION DESIGN TECHNIQUES ; WORD PROCESSING.
OUR IN-DEPTH COVERAGE OF ALL TOPICS IS MORE CRITICAL THAN A LARGE NUMBER OF SPECIALIZED MEETINGS AND LOTS OF EXHIBITORS.

ONE PURPOSE OF COMPUTERFEST IS TO TRADE INFORMATION ABOUT THE CAPABILITIES AND USES OF THE PERSONAL COMPUTER.

NUMEROUS SMALL BUSINESSMEN KNOW COMPUTERS ARE POWERFUL TOOLS, BUT ARE UNFAMILIAR WITH THE UNIVERSE OF ALTERNATIVES WHICH EXIST BETWEEN RADIO SHACK AND IBM, AND SEEK GUIDANCE BEFORE INVESTING.

LIKEWISE, NUMEROUS PERSONAL COMPUTER PROFESSIONALS HAVE A DREAM OF MAKING A LITTLE MONEY FROM OUR HOBBY BY SOFTWARE DEVELOPMENT, OR CONSULTATION WITH PEOPLE JUST GETTING STARTED.

COMPUTERFEST BRINGS BOTH INTERESTS TOGETHER FOR MUTUAL SATISFACTION.
% END OF TEXT

SETTING MONITOR PARAMETERS FROM "BASIC"

by Ian MacMillan & Devin Trussell

(Courtesy of Australian Sorcerer Users Group - Thanks!)

The monitor parameters set the display speed, Baud rate, tape header information, and select the input and output ports that are to be used by the programs running on the machine.

The "selection of Input or Output port" is somewhat misleading because what actually happens is that the routine in monitor that accesses the port in question is called. When O or I are set to an address (XXxx) it means that you have to provide, at that address, a machine routine that will look after whatever device you have in mind.

In order to set the parameters from BASIC, one or two bytes have to be POKEd into the appropriate locations in monitor RAM, which is located at the top of memory. The actual addresses depend on the amount of memory in the system, so that the following tabulates in term of 8,16 and 32 Kilobytes.

1. SEt S=XX...to set video display rate

	8K	16K	32K
POKE 0 to 255 into	8143	16335	32719

NOTE: 0 is fast; 255 is slow.

2. SEt T=X...set Baud rate

POKE 64 for 1200 Baud; 0 for 300	8142	16334	32718

3. SEt FF=XX...to set file type (in tape header) to XX

POKE 0 to 255 into	8158	16350	32734

NOTE: BASIC will only load file types B0 to BF.
File type D8 will not auto execute.

4. SEt X=YYyy...set auto-execute address in tape header

POKE decimal equivalent of desired program starting address bytes, YY in H; yy in L	8163/4 L/H	16355/6 L/H	32739/40 L/H

NOTE: Some uncertainty with this one.

5. Selecting Output Port

Pairs of bytes are POKEd into two addresses:

8K ... 8144,8145

16K ... 16336,16337

32K ... 32720,32721

The bytes to be poked, in the same order as the addresses above, are as follows:

```
SET 0=V     (video)  27 and 224
    0=P     (parlot) 33 and 224 (parallel output)
    0=L     (ce.dvr)147 and 233 (Centronics Printer)
    0=S     (outape) 18 and 224 (tape output)
    0=XXxx           xx and XX  (addr of special ouput routine)
```

NOTE: XX is the decimal equivalent of the most significant byte, and xx is that of the least significant byte.

6. Selecting Input Port

Pairs of bytes are POKEd into two addresses:

8K ... 8146,8147 16K ... 16338,16339 32K ...32722,32723

The bytes to be poked, in the same order as the addresses,are:

```
SET I=K  (keybrd)  28 and 235
    I=P  (parlin) 118 and 231
    I=S  (intape) 218 and 226
    I=XXxx         xx and XX
```

NOTE: We repeat the point that these are the addresses of routines, not ports.

We hope that this information will be useful, but must point out that much of it is untried, and has been derived from study of monitor dis-assemblies, and may contain errors of detail, or calculation. We would be greatly obliged if any such errors were brought to our attention.

Example of usage

```
20 POKE 32719,100
30 REM  SLOWS PRINTING SPEED ON SCREEN
40 POKE 16336,147: POKE 16337,233
50 REM  MAKES THE CURRENT OUTPUT PORT CENTRONICS
```

HOW TO READ OUT THE CHARACTER MEMORY STORAGE FOR ANY CHARACTER

by IM

(Courtesy of the Australian Sorcerer Users Group - Thanks!)

This is good stuff if you want to reproduce characters in large sizes, because it gives you the character formation patterns without having to re-generate them.

Every character is stored as a pattern of eight consecutive bytes in ROM or in the special user definable character RAM.

The expression for Y finds the address of the top line of the character(actually the address is -Y). The routine then looks at that, and the next seven bytes in memory, and prints them out. The bit pattern of these bytes form the shape of the character displayed.

```
10 INPUT "ENTER CHARACTER"; A$
20 Y=2048-8*ASC(A$)
30 PRINT "THE TOP LINE ADDRESS IS ";-Y
40 FOR X=Y TO (Y-7) STEP -1
50 PRINT PEEK(-X)
60 NEXT
70 GOTO 10
```

To user-define characters from BASIC, select your keytop, use the above routine to find the addresses of the eight bytes of the GRAPHIC/SHIFT character you are going to define. Then POKE the decimal values of the bytes required to make your bit pattern, into each location in turn. In a program, a DATA line and a READ in a loop very like the one in the program above will do the trick.

The "chunky graphics" routine in this issue can be used to create large characters using whole character blocks instead of little dots.

CHUNKY GRAPHICS

by IM

(Courtesy of the Australian Sorcerer Users Group - Thanks!)

The following subroutine(s) allow the equivalent of the TRS80 SET and RESET commands.

The horizontal axis allows X values from 1 to 64.
The vertical axis allows values from Y=1 to Y=30.
To SET ... GOSUB 990
To RESET ... GOSUB 1000

```
 990 Z=177: GOTO 1010
1000 Z=32
1010 P=3969-X-((Y-1)*64)
1020 POKE -P,Z
1030 RETURN
```

In your program, establish the values of X and Y and call the subroutine, which can be renumbered to suit yourself.

Terry- A couple of addendums to your software vendor list in Volume 1, Issue 1 of the Sorcerer.

I recently came across a fine book by Vic Tolomei on machine language for the Sorcerer. It is distributed by Quality Software, 6660 Reseda Boulevard, Suite 103, Reseda, CA, 91335. It has become an invaluable reference manual.

In addition, Quality Software distributes a good, high resolution program for plotting curves. It is available on cassette and runs without any difficulties. I only wish it were available on disk so that programs could be mersed.

On the negative side, Compumax Associates "General Ledser" program needs debugging and better video display. The instruction manual leaves much to be desired.

We too had trouble with our first Sorcerer operating with a cassette. However, the problem was in the connection to the "2nd layer" board within the Sorcerer.

We have placed five systems using the Sorcerer/Micropolis in local businesses. All are running reliably and with great praise from the users. -Barry Goldstein

NEW PRODUCT ANNOUNCEMENT

A new catalogue devoted exclusively to Educational Software for personal computers is being published by Queue, 5 Chapel Hill Drive, Fairfield, CT, and will contain educational doftware listings from numerous publishers.

Software listings will be separated by educational level and field, and by computer, and listings for all popular personal computers will be included.

All software can be ordered directly from Queue. For further information, contact Monica Kantrowitz, president, Queue, 5 Chapel Hill Drive, Fairfield, CT, 06432, or call (203) 372-6761.

Terry- I own a 16K Sorcerer and am presently having many problems in connecting a Heathkit WH14 serial printer to it. It seems that Exidy's serial port (RS-232) is not standard. When the driver program is run, the output is not being switched to the printer by the "SET" command. I would consider an interface from the parallel port, but I don't know how. I am looking towards newsletters and Sorcerer User Groups to see if anyone has ever had this problem and how to solve it. If you have any suggestions in resolving this problem, they would be more than welcome. -Dennis Lipovsky

NEW PRODUCT ANNOUNCEMENT

Textwriter
$125 (complete) or
$15 (bound 50-page user's manual)

Organic Software
1492 Windsor Way
Livermore, CA 94550
(415) 455-4034

Textwriter is a text formatting program that is perfectly suited to the printing of personalized form letters, reports and manuals, contracts and specifications, or books and articles. It personalizes form letters by replacing name and address symbols with values read from either a separate mail list file or from the keyboard. It makes the production of reports and manuals very easy by automatically generating a table of contents and alphabetized index, thereby eliminating the tedious task of manually changing the contents and specifications because it allows the user to reference frequently used paragraphs or sections by name for automatic insertion when the document is printed. It speeds the generation of books and articles because it allows chapters or sections to be segmented into separate files and then linked together at the time they are printed permitting editing without the overhead of reading the entire document. It also accumulates and properly places footnotes at the bottom of a page when required.

With all its features, Textwriter is useful for businessmen who generate personalized form letters from mailing lists; any one who writes manuals or reports requiring a table of contents and index; professionals who write contracts or specifications that share common sections; authors who write books or long articles; and hobbyists who need word processing but have a limited budget.

Textwriter is available on all commonly used floppy disk media in versions for use with CP/M and other similar systems such as IMDOS and CDOS; CP/M on TRS-80 or any other CP/M system based at 4200H; North Star DOS and Micropolis MDOS. It works with any terminal and printer; a special video output device is not needed as with other word processors.

DEFINING GRAPHICS KEYS

by Lee Meador

This is a program to define shapes of the graphic keys. When you see the prompt "THE KEY IS.__", enter a graphics key. Any key will do, but there are problems with the alphabetic keys. (I would appreciate any advice here.) Use these codes to produce the graphics you want:

"1" - make a white dot
"." - make a dark dot
"*" - don't change row
"^" - copy row above
"#" - don't change rest of the character

Then SAVE name FC00 FFFF to save your characters, and LOAD name to retrieve them later!

```
 10 REM **********FIND PLACE FOR PARTICULAR KEY**********
 20 AL$="1234567890:=~qwertyuiop[] asdfghjkl;\←zxcvbnm<>?"
 30 GOTO 500
100 INPUT "THE KEY IS ";K$
110 IF LEN(K$)<3 THEN GOSUB 150 : GOTO 300
120 KC$=MID$(K$,1) : K$=LEFT$(K$,1)
130 IF LEFT$(KC$,1)=" " THEN KC$=MID$(KC$,2): GOTO 130
132 COPY=VAL(KC$)
134 GOSUB 150
136 FOR I=0 TO 7
138 POKE -1024+CHAR*8+I, PEEK(-2018+COPY*8+I)
140 NEXT I
145 GOTO 700
146 :
147 REM*****************************************************
148 :
150 IF ASC(K$)>127 THEN CHAR=ASC(K$)-128 : GOTO 250
155 FOR I=1 TO LEN(AL$)
160 TK$=MID$(AL$,I,1)
170 TC=ASC(TK$)-32
180 IF TC<32 THEN TC=TC+16 : GOTO 180
190 TC$=CHR$(TC)
200 IF TK$=K$ THEN CHAR=I-1 : GOTO 250
210 IF TC$=K$ THEN CHAR=I-1+64 : GOTO 250
220 NEXT I
250 PRINT CHAR
260 RETURN
270 :
280 REM**********FORM CHARACTER FROM INPUT***************
290 :
300 PRINT CHR$(128+CHAR);"    76543210        76543210"
320 FOR I=0 TO 7
330 A$=""
332 BYTE=PEEK(-1024+CHAR*8+I)
334 FOR J=0 TO 7
336 IF INT(BYTE/2↑J)-INT(BYTE/2↑J+1))*2 THEN A$="1"+A# : GOTO 340
338 A$="."+A$
340 NEXT J
345 PRINT I;"  ";A$
346 NEXT I
347 FOR I=0 TO 7 :PRINT CHR$(23); : NEXT I
348 FOR I=0 TO 7
349 A$=""
350 FOR J=1 TO 15 : PRINT CHR$(19); : NEXT J : PRINT I;" ";
355 INPUT A$
360 A$=A$+"       "
370 BYTE=0
380 :
390 REM*************************************************
400 :
410 FOR J=0 TO 7
415 TK$=MID$(A$,J+1,1)
420 IF TK$="1" THEN BYTE=BYTE+2↑(7-J)
```

```
422 IF TK$="O" THEN BYTE=BYTE+2↑(7-J):REM * GRAPHICS +1
425 IF TK$="*" THEN 470
427 IF TK$="□" THEN 470 : REM * GRAPHICS + *
430 IF TK$="#" THEN 480
432 IF TK$="▫" THEN 480 : REM * GRAPHICS + #
435 IF TK$="↖" THEN BYTE=OBYTE : GOTO 460
437 IF TK$="↑" THEN BYTE=OBYTE : GOTO 460 : REM GRAPHICS + ^
450 NEXT J
451 REM*****************************************************
460 POKE -1024+CHAR*8+I,BYTE
465 OBYTE=BYTE
470 NEXT I
480 GOTO 700
490 :
495 REM**********SHOW ALL THE CHARACTERS*******************
499 :
500 PRINT CHR$(17);
501 FOR M=1 TO 30 : PRINT SPC(63) : NEXT M
502 PRINT CHR$(17);
503 PRINT
510 FOR I=32 TO 255 STEP 32
520 FOR J=1 TO I+31
530 PRINT CHR$(J);" ";
540 NEXT J : PRINT : PRINT : NEXT I
700 PRINT CHR$(17)
710 FOR I=1 TO 16 : PRINT : NEXT I
720 FOR I=1 TO 12 : PRINT SPC(39) : PRINT : NEXT I
730 PRINT CHR$(17)
740 FOR I=1 TO 16 : PRINT : NEXT I
750 GOTO 100
```

We asked Don Despres to enter, run, and comment on this program. Unfortunately, our printer isn't hooked up at the moment, so we couldn't give you a real printout, but we've proofed this several times - we think it's correct. Here are Don's comments:

I'd like to see Lee add a few more command codes to his excellent program. For instance, I believe Lee's program intends you to look at and modify only the graphic characters. I'd like to look at how Exidy puts all those dots together to to form the regular ASCII characters. Why? Well, if I don't like the shape of an Exidy letter, it would help to have the actual bit by bit dot pattern in ROM known to me before I try to design my own character. I know that 128 of the characters can't be changed, but if I could see the way they're put together, I'd know what to avoid instead of having to do it by trial and error.

I wonder if you can design your own letters what would be enough different from the standard patterns in ROM so you'd be able to have variable pitch or letter styles on the screen, just like the different IBM typewriter element balls.?

I had a few problems with the program. For example, I could not generate a graphic character on key (2/") using the SHIFT key. It generates a "?FC ERROR IN LINE 150". The Graphics key and the (2/") key worked okay.

I defined the (1/!) key as a box:
Lee's program labels it as key #64.
When I did an immediate command
PRINT ASC("□"), I got a response

GRAPHIC + SHIFT + (1/!) key

```
11111111
1......1
1......1
1......1
1......1
1......1
1......1
11111111
```

I got a response that said it was ASCII(192). This is confusing. Lee numbers the graphic keys from 0 - 127, and Exidy numbers them from 128 - 255. Of course, you might not care about comparing Exidy's # to Lee's #.

Likewise, I defined a funny looking large A and assigned it to the (A) key. Lee calls it key 90, but Exidy calls it 128 (via PRINT ASC("A "), using GRAPHIC + SHIFT + (A) key).

I tried to get fancy and modify the program to use □ and ■ instead of "1" and ".", but it didn't work. I was able to keep the box and the large A, but I couldn't keep the solid box.

Lee sent his program listing in on printer output. The printer uses only upper case letters, and Lee added a handwritten note that AL$ (line 20) was to have been entirely in lower case letters. This is fine, except that the Sorcerer considers the "@" a signal to ignore the line. So I entered all the characters in lower case except the "@". It didn't seem to hurt anything.

Lee's documentation tells me everything I need to know, but not everything I want to know about his program. For example, he says that the program works with the graphic key, but I find that:

1 Lower case letter key input reveals standard graphics
2 SHIFT + letter key input reveals user defined graphics
3 GRAPHIC + letter key input sometimes reveals user defined graphics.

I found that sometimes the characters wouldn't stay defined, especially when using letter keys rather than numeric keys.

I made the + key (on the numeric pad) a solid block; key #96, according to the program. Then pressed GRAPHIC + SHIFT + (+). I didn't get my solid block. So I went searching for it, all over the keyboard, and finally found it as GRAPHIC + SHIFT + (J).

So something is out of whack. The key I queried and thought I had redefined ended up redefining a key somewhere else! Why? It's driving me bananas!

ANNOUNCEMENT

Exidy announced another new software package: The Word Processor PAC[tm], which transforms the SORCERER into a dedicated word processing system. The software will support an inexpensive Selectric typewriter or the high-performance Diablo/Qume proportional spaced output printers.

Either an inexpensive cassette audio recorder or a mini-floppy disk may be used for data storage, and a 32K SORCERER will hold eight pages of text before saving to tape or disk is necessary. In the Edit mode, all standard functions of cursor control, pagination, and titling (and a lot of other features) are available to the user. The Command mode has the usual word processor functions, as well as tape merge with memory, line length set (15-120 characters), printer option, string search, and a display of unused space. File names can be up to eight characters in length.

The Word Processing PAC[tm] retails for $99, and Exidy says it's available 30 days ARO from them at 390 Java Dr. Sunnyvale CA, 94086.

At the National Computer Conference (and the Summer Consumer Electronics Show), Exidy introduced a video/disk attachment for the SORCERER.

The swivel based unit attaches directly to the SORCERER's keyboard enclosure and contains a 12" video display and a dual mini-floppy disk system with data storage of 630K words.

Software included with the unit consists of the popular CP/M operating system, Z80 Assembler, Text Editor, Linking Loader, and Microsoft Disk Extended BASIC.

The unit is priced at $2995 retail directly from Exidy.

AUSTRALIAN USER GROUP SPEAKS UP

Frank Schuffelen, of the Australian User Group, has offered to exchange reprint rights with us, and we jumped at the chance. As you can see from some of the material in this issue, he and Ian MacMillan and Devin Trussell have a good thing going. We welcome the Group to our "international" fold, and look forward to a long and lively association with them! - Terry

ACKNOWLEDGEMENTS

ARESCO is deeply indebted this month to Don Despres of Columbia, who happened upon us quite by accident. Don entered programs into the SORCERER for us, so we could present you with running programs rather than error-prone, typed listings. Unfortunately, we didn't have our cassette cables with us - and we had our printer on another system - so after he got it loaded and running, we had to leave it overnight...and the inevitable happened. For the first time in weeks, we had a thunderstorm, lost power, and (of course) lost the program. Don patiently re-entered the whole thing.... and you'll find it in this issue.

KEYBOARD ACCESS WHILE THE PROGRAM RUNS

by Ian MacMillan

(Courtesy of the Australian Sorcerer User's Group - Thanks!)

A small machine language routine is poked into low memory at the start of the program. When called by a USR command, this calls the KEYBRD routine in the Monitor.

The USR line calls the routine until a character is received, and then moves on. The USR line is called each time a new character is needed in your program.

```
10 POKE 1,205 : POKE 2,24 : POKE 3,224 : POKE 4,50
20 POKE 5,8 : POKE 6,0 : POKE 7,201
30 POKE 260,1 : POKE 261,0 : L=USR(0)
 : IF PEEK(8)=0 THEN 30
40 A=PEEK(8)
50 PRINT CHR$(A)
60 GOTO 30
```

The demonstration program above simply prints keyboard entry in a vertical line. To incorporate this in your own program, note that lines 10 and 20 only need to be run once, to establish the machine language routine. Line 30 is the USR line, and the number at the end of the line must be a call to the line itself (if you use line 600, the IF...THEN must go to 600). Line 40 makes A equal to the ASCII value of the character entered; 50 converts this to a character, if that's what you need. The equivalent of line 60 is necessary to get successive entries.

The system works very well, but suffers from the disadvantage that the program will stop running while the key is pressed down.

(Editor's note: Ian hopes to publish an article that shows how to directly access the keyboard port and overcome the problem of the program halt while the key is being pressed.)

HOW NOT TO BE HEXED BY NEGATIVE ADDRESSES

by Ian MacMillan

(Courtesy of the Australian Sorcerer User's Group - Thanks!)

I you want to POKE or PEEK addresses in memory, you usually start by converting some address known in hex to decimal so you can use it from BASIC.

If your decimal address is greater than 32768, you must subtract 65536 to obtain a negative number (which will be the actual number you will POKE).

Similarly, if you have a negative decimal address, and you want to convert it to hex, you must first add 65536 to obtain the decimal number that you can convert to the hex address.

To: Leslie Zatz (re your letter in Vol 1, #2, page 3)
From: Don Despres

END OF LINE TYPING Yes, I agree. Typing program lines close to the 64 character length is difficult. I had that problem when I started, too. I'm anxious to see if someone responds to your request for a "bell" arrangement.

I have a couple of ideas which might help until someone does come up with that better "bell" idea, though. Prompted by your letter, I came up with this as a temporary aid. I took a long piece of regular string; put it on the right side of my monitor, and taped it top and bottom to the screen near the right edge of that amounts to 64 characters. (I also considered going to a stationery or art store to get one of those rolls of drafting tape with a thin black or white line. You could tape that directly on the screen). While most monitors and TV screens have a curved picture tube, either method should work for you. I happen to have the Exidy P31 monitor (the "green screen"), which has a fat clear plastic cover. Tape or string on the screen certainly isn't elegant, but it will do the trick until another reader shows us something better.

There's another routine you can use if you're writing a program which asks the user for input data. You can print a heading guide to aid the user in the correct placement of input fields (I'll give more details on this in the next issue of SOURCE).

TV INTERFERENCE Ay, yes. I've got that problem, too. The retail stores don't have a satisfactory answer for that complaint yet. It's rather important, because the family complains about certain channels when I turn my Sorcerer on. (Editor's note: Has anyone tried the ATV Research Microverter (an RF modulator) with the Sorcerer yet? I know it does kill all interference for APPLEs and VIPs, but since we have a monitor instead of a TV on the Sorcerer, we haven't tried it yet.)

I've seen some magazine ads for devices to take care of the problem - but do they really work? Has anyone tried one?

STRINGS OVER 66 CHARACTERS LONG Leslie, you didn't say how you were attempting to put in a string of over 66 characters, but let me tell you what I found. I was confused when I tried to get those long strings in BASIC as promised by the brochures and manuals. I got the same "lockup" problems (which destroy everything!). It has something to do with continuing to type your string after the cursor has gone into overflow for the next 64 character line on the screen without a carriage return. But CR tells Sorcerer you've finished input!

While Exidy's manual says you can have a string as long as 255 characters, it never explains how it's done. I happened upon the secret clue by reading other books & magazines. Actually, the information is in the "A SHORT TOUR OF BASIC" manual, pages 64 and 65, but you have to be more than a novice to pick it up.

You get long strings by using the string concatenation function (+) inside a program run. Here's a quick example for illustration.

```
 10 PRINT CHR$(12) : REM---CLEAR THE SCREEN
 20:
 30 CLEAR 1000 : REM---ALLOCATE A LARGE STRING SPACE IN MEMORY
 40 X$="ABCDEFGHIJKLMNOPQRSTUVWXYZ " Note the space at the end
 50 X$=X$+X$+"1234567890"             of the string here!
 60 PRINT"---X$=64 CHARACTERS---"
 70 PRINT X$ : PRINT
100:
110 Y$=LEFT$(X$,63)
120 X$=X$+X$+X$+Y$
220 PRINT "---NOW X$=255 CHARACTERS---"
230 PRINT X$
```

Line 40 makes X$ 27 characters long, including the space at the end of the string. Line 50 makes X$ 27+27+10 (64) characters long, and then all 64 characters are printed on the screen in line 70.

Line 110 chops off the last character in X$, stores Y$ as only 63 characters long. Then line 120 adds 64+64+64+63 to make the maximum string length of 255 characters. Remember, this string includes the 8 spaces (one for each of the original "X$"s used) in one large string.

Hope this helps. It is a long, tedious way to get at it, but until someone discovers a better way, I guess it will have to do.

*** ATTENTION PEOPLE ***

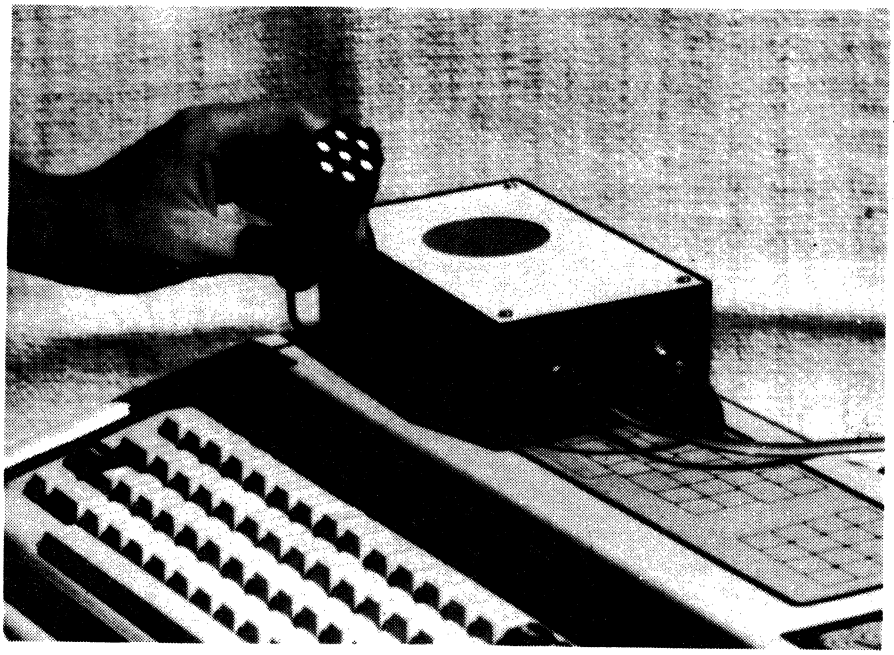
We have been really fortunate to have been contacted by a group of Sorcerer users in Australia (who sent in much of the material we've printed in this issue). We hope the efforts of these folks spur our American and European readers into writing material to share with the Australian group (we don't want to let these good people outshine us, do we?).

Remember - we'll print the next issue as soon as we have enough material in the house to go to press. Any body got any ideas, comments, suggestions, discoveries, experiences to share?

One of the things we'd like to see is reviews of products you've purchased for use with your Sorcerer. And comments regarding your experiences with dealers. Who are the "good guys" out there? Who are the "bad guys"?

Also - Exidy promised to send us a copy of the new Word Processor Pac so we could review it for you - we haven't got it yet, but Paul Terrell says "as soon as possible", so we should have it soon!

COGNIVOX™



Voice input & output for the Exidy SORCERER•

$ 149.00

- Recognizes up to 16 words
- 16 word voice response vocabulary
- Easy two pass training
- Up to 98% recognition accuracy
- Generates music and sound effects
- Excellent software support

COGNIVOX is a unique new accessory for your Sorcerer (16K) computer that opens up a whole new area of man-machine interaction. For the first time speech recognition and voice response are combined in a single low cost voice I/O terminal.

Imagine being able to play a video game with your Sorcerer by calling out your moves! And just as easily you can ask your computer to read its memory to you out loud - a vocal memory dump! The possibilities are endless and fascinating.

The COGNIVOX system includes both high quality hardware and an extensive software collection. The hardware comes complete with microphone and audio amplifier/speaker in a beautiful color coordinated enclosure, ready to plug in the I/O port of your Sorcerer.

The software provided consist of the basic driver programs plus 2 applications program paks (a total of 8 programs) plus two sophisticated voice operated video games (VOTHELLO and VOICETRAP) plus a Talking Calculator program that converts your Sorcerer into a four function floating point calculator that talks!

All these, plus an excellent user's manual can be yours for the amazingly low price of $149 (please add $4 for shipping in US, Cal. add 6% tax). Although the cost of COGNIVOX is lower than the cost of either a speech recognizer or voice output alone, you will be very pleasantly surprised with the high quality of both our hardware and software. Our low prices come from technological breakthroughs in speech processing, not by cutting corners.

Send in your order today and add an amazing new dimension to your Sorcerer computer.

VOICETEK P.O. Box 388, Goleta, CA 93017

• SORCERER is a trademark of Exidy Inc.

RENUMBER

by Devin Trussell

The program here must be appended to the program which is being renumbered. It is executed by a "RUN 63000" command, and is based on the "Line Renumbering On The PET" article which appeared in the March, 1979 issue of Personal Computing. It has been converted to operate on the SORCERER, and three enhancements have been added:

IF...THEN and IF...GOTO statement sequences are correctly handled

The beginning line number is user specified

The increment between line numbers is user specified

The second enhancement enables mutually exclusive line numbers to be given to two programs, in preparation for splicing them together. There are two main restrictions to the program:

Only 256 line numbers can be handled; this can be altered by changing the dimension of OL in line 63010

The increment between line numbers must be less than 256

An outline of the program follows. (A modest familiarity with hexadecimal numbers on the part of the reader is assumed.)

OL is an array containing the old line numbers. The Nth element in the array is the Nth line number in the original program.

OM is the index for array OL. The final value of OM is the total number of lines corrected (see line 63510).

NH and NL are high and low order bytes respectively of the new line numbers.

AH and AL are the high and low order bytes respectively of the hexadecimal address of the beginning of the next line in the program. They are initialized as 1 and 213 (which is 01D5).

AD (see line 63020) is the decimal value of AH and AL.

LL and LH are the second and third bytes respectively following AD, and contain the low and high order bytes of the current line number.

Lines 63000-63019 are program initialization. In lines 63020 to 63070, the old line numbers of the program are changed to the desired new values. By means of a table lookup (see line 63800), the internal references (ie, numbers following THEN (ASCII 162), GOTO (ASCII 137) and GOSUB (ASCII 141) statements) are corrected in lines 63500-63890. The program will give a "NO ROOM" message (see line 63830) if there is insufficient

space to insert the new internal line numbers. The solution then is to manuall correct the lines so affected. In most cases, this chore can be avoided if one or two spaces are inserted ahead of the internal line numbers when the original program is entered. The program will give a "COULDN'T FIND" message if an internal line number reference in the original program cannot be matched with an original line number.

Note: "II" is used to specify the line number increment. It should not be confused with the number eleven. "II" occurs in lines 63019, 63050, and 63820.

```
63000 REM ** RENUMBER PROGRAM **
63001 :
63002 REM ** BY DEVIN TRUSSELL
63003 :
63004 REM ** AUSTRALIAN SORCERER USER'S GROUP
63005 :
63010 DIM OL(255) : OM=0 : AL=213 : AH=1
63015 INPUT "NEW LINE NUMBERS BEGIN AT ";ZZ
63017 NH=INT(ZZ/256) : NL=ZZ-NH*256
63019 INPUT "LINE NUMBER INCREMENT IS ";II
63020 AD=256*AH+AL : LL=PEEK(AD+2) : LH=PEEK(AD+3) : OL=256*LH+LL
63030 IF OL=63000 GOTO 63500
63040 OL(OM)=OL : OM=OM+1
63050 POKE AD+2,NL : POKE AD+3,NH : NL=NL+II : IF NL>255 THEN
NL=NL-256 : NH=NH+1
63060 IF OM>255 GOTO 63500
63070 AL=PEEK(AD) : AH=PEEK(AD+1) : GOTO 63020
63080 :
63500 REM * LINE NUMBERS ALTERED.
63501 REM * INTERNAL REFERENCES NOW CHANGED.
63502 :
63510 PRINT OM;"LINE NUMBERS CORRECTED" : OM=OM-1 : L=468
63520 L=L+4 : LN=256*PEEK(L)+PEEK(L-1) : IF LN=63000 THEN PRINT
"FINISHED" : END
63530 L=L+1 : CH=PEEK(L) : IF CH=0 THEN 63520
63540 IF (CH<>137) AND (CH<>141) AND (CH<>162) THEN 63550
63565 IF CH=32 THEN 63560
63570 IF (CH>47) AND (CH<58) THEN GOSUB 63700 : GOTO 63560
63580 IF N$="" THEN 63530
63590 IF CH=44 THEN GOSUB 63800 : GOTO 63550
63600 GOSUB 63800 : IF CH=0 THEN 63520
63610 GOTO 63530
63620 :
63700 N=CH-48 : N$=N$+RIGHT$(STR$(N),1) : RETURN
63710 :
63800 J=-1 : N=VAL(N$) : FOR I=0 TO OM : IF OL(I)=N THEN J=1 :
GOTO 63890
63810 NEXT I : IF J=-1 THEN PRINT "COULDN'T FIND LINE #";N;"IN
TABLE" : GOTO 63890
63820 NL=II*J+ZZ : NL$=STR$(NL) : NL$=RIGHT$(NL$,LEN(NL$)-1)
63830 IF LEN(NL$)<(L-LO-1) THEN PRINT "NO ROOM TO REPLACE ";N
;"WITH";NL : GOTO 63890
```

```
63840 IF [redacted]NL$)<(L-LO-1) THEN NL$=NL$+" " : GOTO 63840
63850 FOR I=LO+1 TO L-1 : N$=MID$(NL$,I-LO,1) : N=VAL(N$)+48:
POKE I,N
63860 IF N$=" " THEN POKE I,32
63870 NEXT I
63890 N$="" : RETURN
```

(Editor's Note: Devin Trussell is a member of the Australian SORCERER User's Group. Thanks to Frank Schuffelen, USA SORCERER Users have access to all that "down under" expertise!)

ANNOUNCEMENT

Exidy has announced the Development PAC^{tm}, which allows Z-80 assemblies, editing, and debugging.

The Assembler is a two-pass utility and its I/O can be vectored to any device driver within the SORCERER. Source and object code can be spooled to accomodate programs of indefinite length. Absolute assemblies and pseudo operators are also supplied.

The line-oriented Editor allows forward cursor positioning, line deletion, line insertion, input and output of source code to any device driver, and spooling. The edit buffer is left intact for immediate use by the assembler.

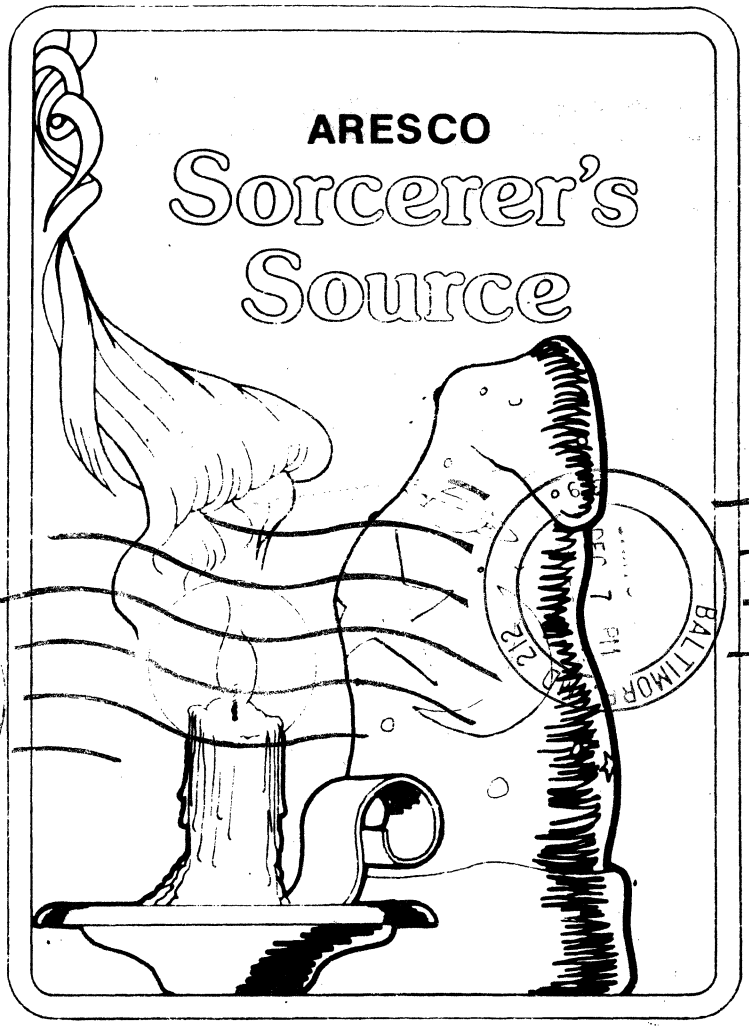
The debugger can display and/or modify any RAM location or Z80 register. It will execute a program with breakpoints and generally serve as a very useful tool to isolate program bugs.

The Development PAC^{tm} will retail for $99. Exidy says it's available 30 days ARO (after receipt of order) from them, at their new address: 390 Java Drive, Sunnyvale, CA 94086.

COMPUTER MART OF WALTHAM LOSES BRUCE McGLOTHLIN TO EXIDY CORP.

It's true! Bruce left the Computer Mart (and the Exidy Monitor Newsletter) to work in Exidy's marketing division. Paul Terrell explained, "We aren't certain yet exactly what Bruce will be doing for us...but we can use someone with his experience and knowledge in almost any department in the company!"

Good luck Bruce! Sorry we weren't able to reach you for your comments.



MERRY CHRISTMAS!

HAPPY NEW YEAR!

from all of us
at ARESCO, Inc.

ARESCO
P.O. Box 1142
Columbia, MD 21044
THE PAPER VIPER RAINBOW SOURCE

THIRD CLASS MAIL

SOURCE