

# SORCERER'S APPRENTICE

VOL. I, No. 6

JAN. 1980



## MISCELLANEOUS NOTES

This issue will try to clear up some of the mystery surrounding the Sorcerer's serial port. Hopefully this will get those of you with serial printers and word processing pacs into business.

When I started this newsletter in May of '79, there seemed to be quite a bit of interest in starting a cassette based program exchange. Since setting things up, the situation has apparently changed - very few of you are sending in programs, or requesting them either, for that matter. I hope this doesn't mean that all those Sorcerers are not being used. Let me reiterate that we are interested in any type of program. If you have a use for a program you wrote, then chances are pretty good that someone else does as well. Whatever you write, send a copy along to Ralph Ruh, 623 Medill Ave., Lancaster, Ohio, 43130, and mention what program you would like to receive in exchange. The exchange benefits from your program, and you benefit by getting a free program. That's hard to beat, with commercial programs going for \$7. By the way, the more documentation you provide in your program (REM statements), the easier it is for someone else to modify your program to suit their exact needs.

The 48 K Sorcerers are now on the market. John Link, of Fremont, California, decided his was running a bit hot after five or six hours of continuous use. He cites no problems, but installed a small blower for added protection. From John: "The installation was easy, required no modification of existing hardware and provides quiet and adequate cooling. Using a small homemade tab made from a piece of fiberglass printed circuit board material, I was able to mount the blower to the top screw holding the line filter to the power transformer on the keyboard side. This places the blower just inside the side vent port on the Sorcerer's left side. Air is drawn into the port. The blower I used cost only \$1.39! (source: Technical Electronics Corp., P.O. Box 2361 Woburn, MA, 01888). I connected the blower to the AC line at the input side of the line filter."

Dan Fabiano of Wayne, Pennsylvania, wrote to tell me about repair times. The turnaround time is excellent, only 48 hours. Send units to be repaired to the attention of J.B. Wells at Exidy.

---

The Sorcerer's Apprentice is published approximately 8 times a year by The Sorcerer User's Group, c/o Dave Bristor, 1530 Washtenaw, Ann Arbor, MI, 48104. Meetings are held the second Thursday of every month at Newman Computer Exchange here in Ann Arbor. For a free copy of this newsletter and more info, send a self-addressed, stamped envelope. Donations of any kind are greatly appreciated.

Well, I'm still around. I'm not sure if that's good or bad.

Today, I thought I would concentrate on the hardware we are using, especially the printers. I mention this because I am hearing about a lot of the problems being encountered in this area and perhaps it might help if I pass along what does work with the EXWP Pac.

We are using two printers: for high speed drafts, we have the Comprint 912, equipped with a parallel board that runs with the Exidy. For slower speed, typewriter quality, we have an IBM Selectric II, equipped with the Escon interface unit.

The two printers serve two totally different functions but each does it very well. The Comprint runs at very high speed and prints a fully formed dot matrix character. It handles the full ASCII set and has lower case descenders. The printout is on an aluminized paper which photocopies very well. Its real virtue, though, is speed. I once timed it at 450 lines in three minutes flat. We use it around the office for drafts of long documents, which takes the strain off the typewriter. Incidentally, the aluminized paper makes excellent photocopies. The machine sells for about \$660.00.

The Escon interface is easy to work with and has turned the Selectric into an excellent typewriter printer. (Escon has obtained a letter from IBM which states that the installation will not void service agreements or new machine warranties.)

The Escon handles the full ASCII set, also, including backspacing and tabbing. (The PROM for ours does not include greater than/less than signs--it prints them as parentheses; otherwise, you get the full ASCII set for BASIC listings. Other PROM's are available.)

It runs slowly, about 12.5 characters per second. That's fast enough for us if we have the Comprint to back us up on drafts. We have the Selectric equipped with a sound cover and a pin-feed platen for continuous feeding of paper. The interface sells for approximately \$600.00 depending on installation and cable length.

Both units run on the parallel port of the Exidy and both run well in that mode.

You can obtain more information by writing Computer Printers, International, 340 E. Middlefield Road, Mountain View, California, 94043 or to Escon, 171 Mayhew Way, Suite 204, Pleasant Hill, California, 94523.

See you next month!

## "DATABASE"

I do not normally review software for two reasons: one, we do not use our Exidy for much besides word processing and two, because I am not that much of an expert at software. However, I have volunteered to do a review of "Database" (R G R Software Company, 623 Medill Avenue, Lancaster, Ohio, 43130, \$11.75, post-paid).

This program is a pleasure. It allows the user to set up ten fields of data and, believe it or not, the data can all be saved on cassette tape, alpha included. The author of the program has apparently solved the problem of cassette data save. The program fields will accept up to 255 characters, I am told, and just about all the instructions you need are built right into the program. The user can create a record, sort records alphabetically and numerically, update records, add records, insert new records, delete records and so forth. The sort feature alone makes the program well worth the money. It will also search for a four-number or four-letter string anywhere in the data.

We will be using this program as the basic program for our client time-keeping system. Of course, the read and write from cassette are slow but it is a lot cheaper than a disk system and should work well for anyone with an Exidy.

A O V E R T I S E M E N T

## RESTORING A BASIC PROGRAM AFTER 'NEW'

This 38 byte machine language program will restore a BASIC program after you have used the NEW command and then had second thoughts. The assembly code is given but you can easily key in the machine code using the ENTER command and then save it on tape. LOAD at any convenient location after getting out of BASIC by BYE (e.g. 10H), then just GO to the load address (e.g. GO 10). The program will then restore your BASIC program, return to BASIC automatically and issue a READY.

R. J. Lomax  
Ann Arbor, MI

EXIDY Z-80 ASSEMBLER

ADDR OBJECT ST #

		0001 ;***** RECOV *****
		0002 ;
		0003 ;RECOVERS A SORCERER BASIC PROGRAM
		0004 ;AFTER A 'NEW' COMMAND.
		0005 ;
		0006 ;COPYRIGHT (C) 1979 BY R. J. LOMAX
		0007 ;
		0008 ;USE: LOAD IN ANY CONVENIENT PLACE,
		0009 ; (PROGRAM IS RELOCATABLE).
		0010 ; GO TO LOAD ADDRESS.
		0011 ; PROGRAM IS RESTORED.
		0012 ;
		0013 ORG 0C
		0014 LD A,00 ;END OF LINE
		0015 LD HL,1D9H ;START OF 1ST LINE
		0016 LD BC,0 ;10000H BYTES
		0017 CPIR ;FIND 00
		0018 LD (1D5H),HL ;ADDR OF 2ND LINE
		0019 TEST LD A,H ;LAST LINE?
		0020 OR L
		0021 JR Z,DONE-\$ ;GO ON YES
		0022 LD E,(HL) ;GET ADDR NEXT LINE
		0023 INC HL
		0024 LD D,(HL) ;...INTO DE.
		0025 EX DE,HL ;BACK INTC HL
		0026 JR TEST-\$ ;TEST AGAIN.
		0027 DCNE INC DE ;ADDR AFTER...
		0028 LD (1B7H),DE ;..END TO 1B7
		0029 LD (1B9H),DE ;..AND 1B9, 1B8
		0030 LD (1BBH),DE
		0031 JP 0FFFFH ;TO BASIC WARM START
		0032 ;END

ERRORS=0000

CONE

0017 TEST

000D

SEARCH- program contributed by John Link, Fremont, CA

This routine can be appended to another BASIC program to search that program for any occurrence of a given string and indicate in which line(s) that string is found.

```
60000 A=469:INPUT "SEARCH FOR";A$  
60005 IF A$="LINE 0" THEN 60060  
60010 ST=PEEK(A)+256*PEEK(A+1)  
60015 LN=256*PEEK(A+3)+PEEK(A+2):A=A+4  
60020 IF LN=60000 THEN 60090  
60025 FOR X=A TO ST  
60030 IF PEEK(X)=ASC(LEFT$(A$,1)) THEN GOSUB 60040  
60035 NEXT:A=ST:GOTO 60010  
60040 FOR Y=1 TO LEN(A$)  
60045 IF PEEK(X-1+Y)<>ASC(MID$(A$,Y,1)) THEN RETURN  
60050 NEXT:PRINT LN:RETURN  
60055 A$="":FOR X=473 TO 600  
60060 IF PEEK(X)=0 THEN X=600:GOTO 60010  
60065 A$=A$+CHR$(PEEK(X)):NEXT  
60090 PRINT:INPUT "DELETE SEARCH";GH$  
60095 IF LEFT$(GH$,1)="Y" THEN POKE A-4,0  
60100 POKE A-3,0  
60105 END
```

A SHORT TOUR OF BASIC- ERATA - PARTS 4 and 5 Thanks to Marvin Weingast

#### (L2) The function USR(X)

④

The function USR allows a BASIC program to call a machine language subroutine. The format for the call is:

```
LET <variable>=USR(0)
```

Here, 0 is used as a dummy argument, and <variable> is a dummy variable. You can use any legal variable as the dummy, but remember that its value will change when the USR function is called.

When Standard BASIC first takes control of the Sorcerer, the machine code C3 (hexadecimal) is written in memory address 0103 hex (259 decimal). This is the first byte of a three-byte jump instruction. Addresses 0104 and 0105 hex (260 and 261 decimal) are the other two bytes of the jump instruction --you must load the starting address of your machine language subroutine into 0104 and 0105 hex before you call the USR function.

Use the POKE command to load your starting address into 0104 and 0105 hex. Put the low-order byte into 0104 and the high-order byte into 0105. Remember that you must use decimal notation with the POKE command.

---

Example:

100 POKE 260,16

puts 10 hex (16 decimal)  
into address 0104 hex  
(260 decimal)

110 POKE 261,0

puts 0 into address 0105  
hex (261 decimal)

200 LET A=USR(0)

calls the machine language  
subroutine starting at  
address 0010 hex

---

Standard BASIC has its own stack, separate from the Monitor stack; the USR subroutine uses the BASIC stack. So long as your USR routine leaves the stack unaltered, a Z80 RET command will return control to BASIC. Addresses 0000 through 00FF hex (0 through 255 decimal) are not used by BASIC or the Monitor; you can safely use these addresses as scratchpad RAM for your subroutine, or to store the subroutine.

The simplest way to pass parameters between a BASIC program and a USR subroutine is by means of PEEK and POKE commands. Use POKE to put a decimal number into a decimal address; your Z80 subroutine can then pick the number (in hexadecimal) out of that address (also hexadecimal). Use PEEK to read a number into your BASIC program, after it has been deposited by your USR subroutine.

PAGE 7  
BLANK

The Exidy Monitor - part II - by Dave Bristor

Last time, I gave some introduction as to why anyone would want to do anything in machine language and also some references to look into. I hope you did take a look at at least one of them; it will be helpful.

Recall the purpose of our machine language subroutine: it must scan the keyboard and return the value of the key pressed, or return a null if no key is down. There is a similar routine in the monitor, the KEYBRD routine at E018 (hex). When called, KEYBRD returns the value of the key in the down position in the Z-80's "A" register. Since BASIC cannot access the A register, after a call to KEYBRD we can store it away. The following routine accomplishes this:

<u>ADDR</u>	<u>OBCODE</u>	<u>LABEL</u>	<u>SOURCE</u>
0000	CD 18 E0	START	CALL KEYBRD
0003	32 0A 00		LD (VALUE),A
0006	C9		RET
0007		VALUE	DEFS 01

A little explanation is in order. The first column is the address of a byte of memory. Each byte in memory has a unique address, ranging from 0 to 65535 (0 to FFFF hex). Thus the machine language subroutine starts at 0. This is called its entry point. The second field is labeled OBCODE, for object code. Object code is the language that the Z-80 processor understands. In this case, CD tells the computer to "call" (as in GOSUB) the subroutine whose address is E018 (note! that the digits are reversed!). This happens to be the address of the keyboard routine mentioned earlier. The LABEL field is of little importance as it concerns this statement, but remember that we have to store that value from the A register somewhere. VALUE, the second label, is the address of the location in memory where it will be put. The last field, SOURCE, is assembly language. If we had the development pac, we would give it as input the LABEL and SOURCE columns and get back the other two fields. Since I'm assuming you don't have a development pac, we will have to do our own assembly. This is called hand coding. As I said earlier, familiarity with the Z-80's instruction set is imperative here, because how do you expect to be able to "talk" to it unless you speak its language? The way to write a program is to figure out what you need done, write down (in assembly language) all the source code and labels you need, then translate to machine code by hand. This involves looking up a particular instructions op-code, and writing it down in the OBCOD column along with any necessary operands. There is no opcode for the DEFS mnemonic (things in the first half of the SOURCE column are called mnemonics, the second half has operands) because we are not interested in what starts out at VALUE, only at what is there after a call to KEYBRD.

I'm going to stop here with this part. Next issue will detail the use of the monitor in entering, looking at, and executing machine language subroutines. In the meantime, try adding this twist to the above routine: instead of returning a null, make the program (machine language) stay in a loop until a key is pressed. (Hint use the Z flag status, and see the Exidy manuals).

## PROGRAM REVIEWS

Staley's Sorcerer Software (22 Stayman Court, Lafayette, IN, 47905) has quite a selection of programs to offer. I have Touchdown, Blackjack, U.S. Map, and Function Plot.

Touchdown is a great game. Living in a fraternity, this one got lots of use during the football season, and was received very well. The program is strong on graphics, and plays a good game of football. The human player can play against the Sorcerers, or against another human opponent. Give your friends a chance at this one, and it may be several hours before you can get back to your computer. Note: there is a bug in the copy of the program I have that eventually may cause the computer to crash. For some reason the ball will sometimes be spotted off the field. Play continues, off the field, until the ball starts munching through other memory off the screen and crashes the computer. (Don't know why yet, but some people have more fun doing this than continuing the game.)

Blackjack is a superb graphics program. The cards are drawn out quite rapidly on the screen. Several players are allowed, making it a fun game to play at parties. Betting is included, just like at a real casino. The graphics alone are worth the price if you are planning on any other card games.

U.S. Map is another very graphic program. It can draw the outline of the United States, and then draw the outline of any state in its proper position. Alaska and Hawaii are included, but drawn outside the U.S., of course. The program can be used as a game, a quiz of state capitals, as well. As such, it can be used to teach children the locations of our states and their capitals. (Children in this sense can apply to anyone using the computer, I learned some things I didn't know just testing the program out).

If you ever need a plot of anything, and need it now, then I just hope you have Function Plot on hand. This program can plot any two-dimensional function, whether it be polar or in terms of x and y. The function doesn't even have to be entered in in BASIC! The running program accepts your function as input, then proceeds to translate it into BASIC and evaluate it over the range of x's you have specified. The scaling of the y axis can be manual or automatic. This marvelous program has excellent plotting resolution and even allows two functions to be plotted simultaneously.

All of the above programs may be ordered directly. Touchdown is available for \$14.95, others cost \$9.95.

USING DIABLO PRINTERS- from Herman Schwedinger, 3 Millbrook Road,  
New Paltz, NY, 12561

Herman sent in his experience with the Diablo model 1640 and 1650 printers. The hook up is as follows:

"The Diablo did require switching pins 2 and 3, as Exidy's instructions indicated. However, the printer required the following changes also. We pulled out pins 4, 5, 6, and 7 from the male plug at the end of the Diablo's cable, and prevented them from plugging into the Sorcerer's serial port. Then, after pulling pins 6 and 7 out of the plug, we shorted them by simply tying a copper wire around both pins. We did not short 4 and 5 but wrapped them individually in insulating tape. Folding the wires carefully into the plug, we reconnected the face plate and thus there were no loose wires visible.

"That's all there is too it. We do not know if this procedure will work on Diablo 1620....Also, has anyone expanded their Sorcerer from 32K to 48K, using the S-100 box? Finally, has anyone used the Sorcerer to sort and update bibliographic files?"

If anyone can help out in these areas, write directly to the address above or to me.

HARD COPY AND THE WORD PROCESSOR- from Phil Wilkinson, 755 Pt San Pedro Rd, San Rafael, CA, 94901

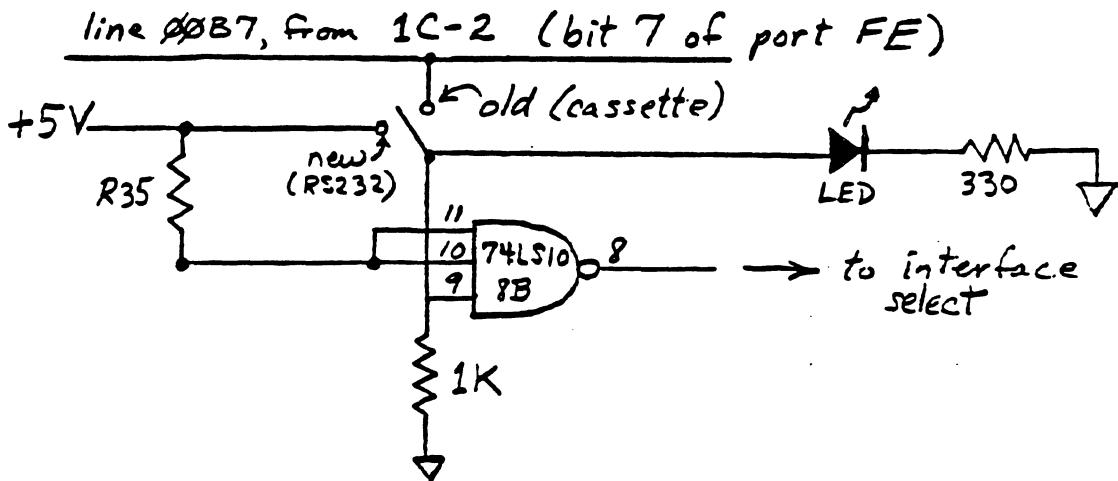
"Enclosed is a short driver program that works perfectly with my Word Processor pac. The latter supports user written drivers if one tells it where to look for them by putting the starting address into 7E7 (low byte) and 7E8 (high byte). No extraneous characters at all with this one. There is a delay built into slow output down enough to suit my 15 char/sec Selec-tric. The duration of the delay is easily varied."

0000: F5	PUSH AF	SAVE ACCUMULATOR, FLAGS
0001: F5	PUSH AF	
0002: FD 7E 3D	LD A,(IY+3D)	GET BAUD RATE
0005: F6 80	OR 80	ENABLE RS232
0007: D3 FE	OUT(FE),A	
0009: FD 77 45	LD (IY+45),A	
000C: F1	POP AF	RESTORE CHARACTER
000D: CD 12 E0	CALL E012	SEND TO RS232
0010: DB FD	IN A,(FD)	
0012: CB 47	BIT 0,A	WAIT TILL UART
0014: 28 FA	JRZ FA (0010)	DONE
0016: 3E 25	LD A,25	
0018: OE FF	LD C,FF	DELAY
001A: OD	DEC C	ROUTINE
001B: 20 FD	JRNZ FD (001A)	
001D: 3D	DEC A	
001E: 20 F8	JRNZ F8 (0018)	
0020: F1	POP AF	RESTORE CHARACTER
0021: C9	RET	RETURN

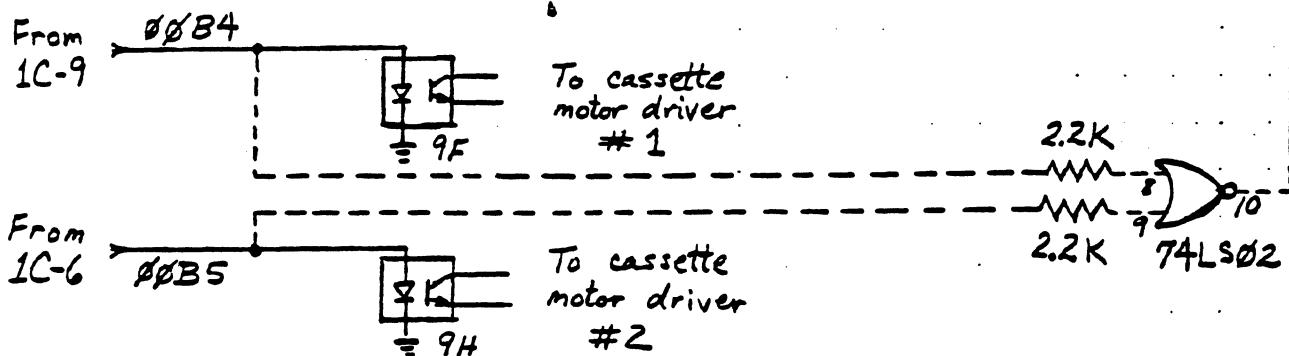
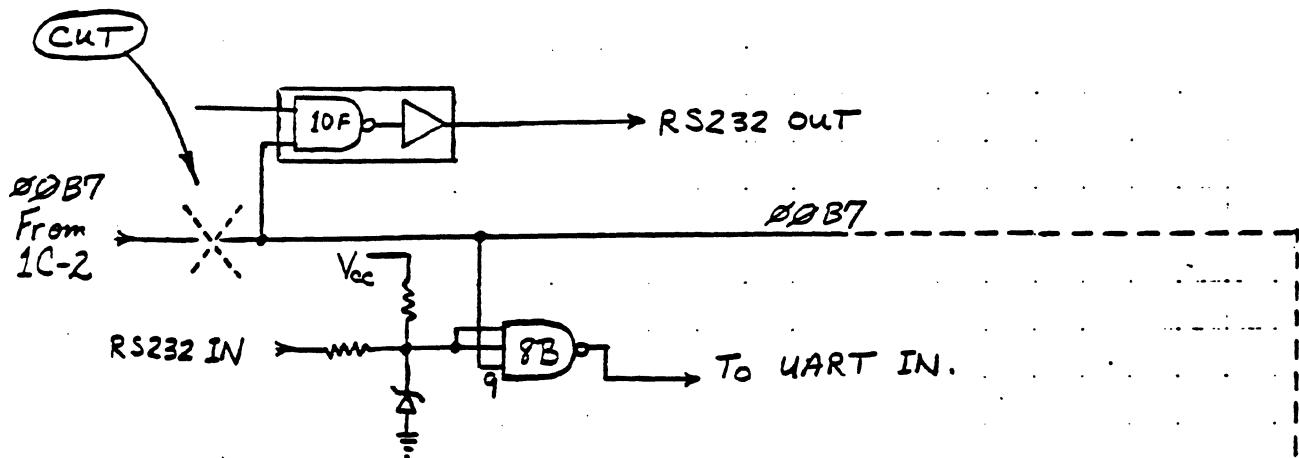
## THE EXIDY SERIAL PORT

First off, let me say that if you want to run a serial device off your Sorcerer, the easiest fix is more than likely to be buying the parallel to serial converter that Mark Longley is selling (see his ad in this issue). I have heard nothing but praise for it from friends. But if you're like me, you have other uses planned for that parallel port, and would like to be able to use the serial port built in. Well, there is hope, my friend. Most of what follows is from Bryan Lewis (Putnam Valley, NY).

The problem with the serial port stems from the fact that it is cheaper to have one part do many things than to have many parts each doing their own thing. Thus thought Exidy when they decided to use the same Z-80 I/O port, #254, for the serial interface status and for the keyboard scan. In other words, you can tell the Sorcerer that you want the serial interface to be RS-232 instead of cassette, but as soon as the keyboard is scanned again, you're back into cassette mode. Bryan sent the following simple hardware modification that uses only a switch, two resistors, and an LED. The drawing refers to drawing 7 in the Sorcerer Technical Manual.



Now something for the circuit wizards out there. Bryan discovered that the only time we really want to be in cassette mode is when we are using the cassettes themselves. Note that when we are using the cassettes, one of the two motor control lines changes status. In other words, it should be possible to control selection of RS-232/cassette based on the motor control lines. The following diagrams Bryan's modification:



A few days later I received a note from Bryan telling me to change the 2.2 K resistors to 4.7 K's to improve reliability. A few days later still, I received word that he had given up on this "perfect fix". There seemed to be a loss of reliability under the following conditions: 1200 Baud, using automatic cassette motor control, trying to read a tape written under the same conditions. (Other people's tapes read OK.) I hope that some hardware genius can pass along the information to correct this situation and make Bryan's fix really "perfect". The track seems to be right, but the wheels aren't fitting quite right.

continued from page 4

sort (which, by the way, do need to be renumbered).

Has anyone encountered the message ?MO ERROR? The manuals make no mention of this particular error. How about REDO FROM START? I have never seen this displayed but it is in the BASIC microcode. If anyone has the answers to these questions or has other helpful information please write to The Sorcerer's Apprentice as I am sure it would be greatly appreciated.

0010: 21 80 F0 11 81 F0 01 7F 07 36 20 ED B0 C9

Screen clearing program

```

10C :
110 :      REM PRINT FORMAT ("PRINT USING")
120 :      REM ADAPTED FROM AN ARTICLE IN 'KILOBAUD'
130 :      REM BY TCM BASSETT -- AUGUST 1979
140 :
150 PRINTCHR$(12):PRINTTAB(1);"PRINT FORMATTING DEMONSTRATION"
160 PRINT:INPUT "COLUMN NUMBER OF RIGHT MARGIN":B
170 PRINT:INPUT"TYPE THE NUMBER TO BE FORMATTED":A
180 X1=A:GOSUB 30000
190 PRINT
200 PRINT TAB(B-X3);X$
210 GOTO 170
220 :
230 :
240 :      REM THE CODE BEGINNING WITH LINE 30000 IS THE FORMAT
250 :      REM PORTION OF THE PROGRAM -- THE PRECEDING CODE IS
260 :      REM FOR DEMONSTRATION PURPOSES. TO USE THE PROGRAM,
270 :      REM DELETE ALL LINES PRIOR TO LINE 30000, AND THEN
280 :      REM MERGE THE SUBROUTINE WITH YOUR HOST PROGRAM AND
290 :      REM RENUMBER THE COMBINED PROGRAMS. SET X1 EQUAL TO
300 :      REM THE VARIABLE TO BE FORMATTED, CALL THE SUB-
310 :      REM ROUTINE, AND THEN PRINT TAB(RIGHT MARGIN-X3);X$
320 :
330 :      REM NEGATIVE NUMBERS ARE PRINTED IN PARENTHESES (12.50)
340 :      REM IN ACCORDANCE WITH ACCOUNTING PRACTICES. TO
350 :      REM PRINT NEGATIVE NUMBERS WITH A TRAILING MINUS
360 :      REM SIGN, CHANGE LINE 30130 TO:
370 :      REM           X$=X1+X$+"-"
380 :
390 :      REM TO ADD DOLLAR SIGNS TO FORMATTED NUMBERS, CHANGE
400 :      REM LINE 30100 TO:
410 :      REM           X$="$"+X1+X$
420 :      REM AND LINE 30130 TO:
430 :      REM           X$="($"+X1+X$+")"
440 :
450 :      REM PROGRAMS TO MERGE AND RENUMBER BASIC PROGRAMS
460 :      REM FOR THE SORCERER ARE AVAILABLE FROM:
470 :
480 :      REM           THE SORCERER USERS GROUP
490 :      REM           RALPH G. RUH
500 :      REM           623 MEDILL AVE.
510 :      REM           LANCASTER, OH 43130
520 :
530 :      REM SUBROUTINE BEGINS HERE
540 :
30000 X4=SGN(X1):X1=INT(ABS(X1)*100+.5)/100:X$=""
30010 X1$=STR$(X1):X2=LEN(X1$)-1:X1$=MID$(X1$,2,X2)
30020 FOR L=1 TO X2
30030 X2$=MID$(X1$,L,1)
30040 X3=L
30050 IF X2$=".0" GOTO 30080
30060 NEXT L
30070 X$=".00"
30080 IF X3=(X2-1) THEN X$="0"
30090 IF X4=(-1) THEN 30130
30100 X$=X1$+X$
30110 X3=LEN(X$)
30120 RETURN
30130 X$="(" + X1$ + X$ + ")"
30140 X3=LEN(X$)-1
30150 RETURN

```

O

B  
P

C