NOVEMBER SPECIALS

For the Atari(tm) Computer

JAWBREAKER On-Line Systems
Cat No. 3217 Atari, 24K, disk **$28.95**

STAR BASE HYPERION Quality Software
Cat No. 3224 Atari, 24K, cass **$19.95**
Cat No. 3225 Atari, 32K, disk **$21.95**

SURVIVAL / ADVENTURE - USA Software
Cat No. 3218 Atari, 32K, cass **$24.95**
Cat No. 3219 Atari, 32K, disk **$24.95**

Axlon
32K RAMCRAM
Memory expansion for your Atari computer. Assembled & tested, gold contacts and hi-impact plastic case. Order now and save $$. (reg. $189)
Cat No. 3003 Special **$164.95**

FILEMANAGER 800
A truly versatile database manager for your Atari 800 computer. Store and retrieve information exactly the way you want it.
Cat No. 3192 Atari 800, 32K, disk **$93.50**

SEND FOR OUR FREE CATALOGUE!!

Page after page of hardware and software products for your Apple® Atari® or TRS-80® . Hundreds of products offered at low, low SALE prices. Write or phone today and SAVE!!

HOW TO ORDER

Mention this ad and WE PAY SHIPPING & HANDLING (UPS ground, USA only). Write or phone. Pay by Check, M/C, Visa, or COD. (Add $1.40 addl. for COD) Offer expires Dec. 1, 1981.

HW ELECTRONICS

19511 Business Center Dr., Dept G11
Northridge, CA 91324
(800) 423-5387 (Outside Calif.)
(213) 886-9200 (In Calif.)

SOFTWARE for the ATARI 800* and ATARI 400* from QUALITY SOFTWARE

STARBASE HYPERION™
By Don Ursem

Become absorbed in this intriguing, original space simulation of war in the far future. Use strategy to defend a front line Star Fortress against invasion forces of an alien empire. You create, deploy, and command a fleet of various classes of space ships, while managing limited resources including power generators, shields and probes. Real time responses are sometimes required to take advantage of special tactical opportunities. Use of color, sound, and special graphics add to the enjoyment of this program. At least 24K of RAM is required.

On Cassette — $19.95 On Diskette — $22.95

NAME THAT SONG
By Jerry White

Here is great entertainment for everyone! Two players listen while the Atari starts playing a tune. As soon as a player thinks he knows the name of the song, he presses his assigned key or joystick button. There are two ways to play. The first way requires you to type in the name of the song. Optionally, you can play multiple choice, where the computer asks you to select the title from four possibilities. The standard version requires 24K of RAM (32K on diskette) and has over 150 songs on it. You also get a 16K version that has more than 85 songs. The instructions explain how you can add songs to the program, if you wish. Written in BASIC.

On Cassette — $14.95 On Diskette — $17.95

QS FORTH
By James Albanese

Want to go beyond BASIC? The remarkably efficient FORTH programming language may be just for you. We have taken the popular fig-FORTH model from the FORTH Interest Group and expanded it for use with the Atari Personal Computer. Best of all we have written substantial documentation, packaged in a three ring binder, that includes a tutorial introduction to FORTH and numerous examples. QS FORTH is a disk based system that requires at least 24K of RAM and at least one disk drive. Five modules that may be loaded separately from disk are the fig-FORTH kernel, extensions to standard fig-FORTH, an on-screen editor, an I/O module that accesses Atari's operating system, and a FORTH assembler.

Diskette and Manual — $79.95 Manual Only — $39.95

FOR OUR COMPLETE LINE OF ATARI SOFTWARE
PLEASE WRITE FOR OUR CATALOG

QS QUALITY SOFTWARE
6660 Reseda Blvd., Suite 105, Reseda, CA 91335
(213) 344-6599

ASK FOR QUALITY SOFTWARE products at your favorite computer store. If necessary you may order directly from us. MasterCard and Visa cardholders may place orders by calling us at (213) 344-6599. Or mail your check or bankcard number to the address above. California residents add 6% sales tax. *Shipping Charges:* Within North America orders must include $1.50 for shipping and handling. Outside North America the charge for airmail shipping and handling is $5.00. Pay in U.S. currency.

*Indicates trademarks of Atari.

Table 1.

FIELD NAME	OFFSET WITHIN IOCB (BYTES)	SIZE OF FIELD (BYTES)	PURPOSE OF FIELD
ICHID	0	1	SET BY OS. Index into device name table for currently OPEN file, set to $FF if no file open on this IOCB.
ICDNO	1	1	SET BY OS. Device number (e.g., 1 for "D1:xxx" or 2 for "D2:yyy")
ICCOM	2	1	The COMMAND request from user program. Defines how rest of IOCB is formatted.
ICSTA	3	1	SET BY OS. Last status returned by device. Not necessarily the status returned via STATUS command request.
ICBAL ICBAH	4	2	BUFFER ADDRESS. A two byte address in normal 6502 low/high order. Specifies address of buffer for data transfer or address of filename for OPEN, STATUS, etc.
ICPTL ICPTH	6	2	SET BY OS. Address minus one of device's put-one-byte routine. Possibly useful when high speed single byte transfers are needed.
ICBLL ICBLH	8	2	BUFFER LENGTH. Specifies maximum number of bytes to transfer for PUT/GET operations. *Note:* this length is decremented by one for each byte transferred.
ICAX1	10	1	Auxiliary byte number one. Used in OPEN to specify kind of file access needed. Some drivers can make additional use of this byte.
ICAX2	11	1	Auxilliary byte number two. Some serial port functions may use this byte. This and all following AUX bytes are for special use by each device driver.
ICAX3 ICAX4	12	2	For disk files only: where the disk sector number is passed by NOTE and POINT. (These bytes could be used separately by other drivers.)
ICAX5	14	1	For disk files only: the byte-within-sector number passed by NOTE and POINT.
ICAX6	15	1	A spare auxilliary byte.

```
        ...
        ...                         ;CONTINUE WITH MORE
                                     CODE
        ...
MSG     .BYTE   'THIS IS A MESSAGE',$9B
```

Just a very few notes on this routine: (1) If the command had been "GETREC," the OS would have gotten a line from the keyboard and put it into the "buffer" at MSG. (2) If the X-register had been set to $20 and if the printer had previously been OPENed at IOCB number 2, then *this same code* would have sent the message to the printer. (3) If the buffer length had been given as less than 18, the message would have been truncated to the specified length. That's all on I/O for this month. I hope you will hound your mailbox until your next issue of **COMPUTE!** arrives.

Bugs In BASIC

Several people have requested a list of all known bugs in Atari BASIC. The following list may not be complete, but it certainly enumerates all the bugs that may be considered "killers."

1. In the course of editing a BASIC program, sometimes the system loses all or part of the program and/or simply hangs. Often, turning power off and back on is the only solution. Contrary to popular belief, this condition is related to nothing except the size of the program that is being moved by a delete operation (*not* the size of the deleted line). FIX: NONE. Sorry about that. Just be sure and SAVE your programs often, especially if you are doing heavy editing.

2. String assignments that involve the movement of multiples of 256 bytes do not move the first 256 bytes. FIX: don't move multiples of 256 bytes. An easy way to accomplish this is to always move an ODD number of bytes. Usually, moving one extra byte is fairly easy to handle.

3. The cassette handler doesn't always properly initialize its hardware interface. Symptoms: ERROR 138 and ERROR 143. FIX: use an LPRINT before doing a CSAVE, etc. (This isn't a BASIC bug, but BASIC can be used to fix it.)

4. Taking the unary minus of a zero number (e.g., PRINT -0) can result in garbage. Usually this garbage will not affect subsequent calculations, but it does print strangely. FIX: don't use the unary minus in cases where there may be a doubt (e.g., use PRINT 0-x if 'x' might be zero).

5. Strange things can happen if you type in a program line longer than three screen lines long. Reason: the system editor device (E:) cuts off your input at three lines and gives it to BASIC, which processes it as is, and then E: gives the rest of your input to BASIC as the next line! FIX: don't try to put in program

*Indicates trademarks of Atari

lines bigger than three screen lines.

6. Using an INPUT statement without a variable (i.e., just '10 INPUT') does *not* cause a syntax error (it should) and may cause program lock-up when RUN. FIX: don't do it. (What did you expect? BASIC is in ROM, so *it* can't be fixed.)

7. Most keywords can be used as variable names. (Try this sometime: LET LET=5 : LET PRINT=3 : PRINT PRINT : PRINT LET ... it works!) Some cannot, and BASIC will tell you about them. But 'NOT' cannot be the first three letters of any variable name. Example:

10 LET NOTE=5 : PRINT NOTE

If you enter that line and then LIST it, you will get

10 LET NOTE=5 : PRINT NOT E

because in an expression NOT is a unary operator that is never seen as part of a variable name. (In the LET, only a variable name is expected, so NOT is never seen.) This is the only "poison" keyword in Atari BASIC. (Note the use of 'LET' in several instances above. Generally, assignment to a variable name which starts with a keyword requires the use of LET to avoid confusing the syntaxer.)

8. LOCATE and GET do not reinitialize their buffer pointer, so they can do nasty things to memory if used directly after some statements (e.g., they can change the line number of a DATA statement if used after a READ). FIX: reinitialize the pointer by using a STR$ function call (e.g., XX=STR$(0) works fine). Clumsy, but it works. PRINTing a numeric value works also (since PRINT calls STR$ internally). This fix is probably one you can ignore until it happens to you.

9. An INPUT of more than 128 bytes (from disk, cassette, etc.) will write into the lower half of page six RAM ($0600-$0657F). This is *not* a bug, it was designed that way. The lower half of page six was supposed to be available to BASIC, but someone at Atari forgot to tell someone else at Atari (and even two different memory maps in the Atari BASIC Reference Manual don't agree). As a consequence, both Atari and user programmers have come to regard all of page six as their own and have put small assembly language programs there. FIX: don't use the programs from $0600-$067F or don't INPUT such long strings.

There are a few other minor bugs (e.g., you can say DIM A(32766,32766) without getting an error message), but, by and large, they won't affect most programs. If anyone thinks they know of any other major bugs, let me know and I will try to provide a fix. Please let us know what topics you want covered.

IOCB field offset and name / Type of command	0 ICHID	1 ICDNO	2 ICCOM	3 ICSTA	4 BUFFER ADDRESS ICBAL	5 ICBAH	6 PUT-A-BYTE ADDRESS ICPTL	7 ICPTH	8 BUFFER LENGTH ICBLL	9 ICBLH	10 ICAX1	11 ICAX2	12 ICAX3	13 ICAX4	14 ICAX5	15 ICAX6	Mnemonic label used by Atari for this field
OPEN	•	•	3	•	FILENAME		•	•			SEE TEXT						OPEN
CLOSE	•		12	•													CLOSE
DYNAMIC STATUS		•	13	•	FILENAME												STATIS
GET TEXT RECORD			5	•	BUFFER				LENGTH								GETREC
PUT TEXT RECORD			9	•	BUFFER				LENGTH								PUTREC
GET BINARY RECORD			7	•	BUFFER				LENGTH								GETCHR
PUT BINARY RECORD			11	•	BUFFER				LENGTH								PUTCHR
	EXTENDED COMMANDS: DISK FILE MANAGER ONLY																
RENAME		•	32	•	FILENAME												RENAME
ERASE		•	33	•	FILENAME												DELETE
PROTECT		•	35	•	FILENAME												LOCKFL
UNPROTECT		•	36	•	FILENAME												UNLOCK
NOTE			38	•									SECTOR NUMBER		BYTE		NOTE
POINT			37	•									SECTOR NUMBER		BYTE		POINT

—LEGEND—

* — Set by OS when this comand is used.
BUFFER — 16-bit address of a data buffer.
FILENAME — 16-bit address of a filename.
LENGTH — length (in bytes) of a data buffer.
SECTOR NUMBER and BYTE — see text.

©

Santa Cruz Educational Software * Tricky Tutorials * Santa Cruz Educational Software * Tricky Tutorials * Santa Cruz Educational Software * Tricky Tutorials

ATARI™ GRAPHICS MADE EASY? YES?

FINALLY — SOMEONE HAS WRITTEN A SERIES OF PROGRAMS THAT DO ALL THE DIFFICULT CALCULATIONS FOR YOU WHILE DEMONSTRATING THE SPECIAL "TRICKS" THAT THE ATARI IS CAPABLE OF . . . .

TRICKY TUTORIALS(tm)

#1: DISPLAY LISTS — This program teaches you how to alter the program in the ATARI that controls the format of the screen. For example: when you say graphics 8 the machine responds with a large graphics 8 area at the top of the screen and a small text area at the bottom. Now, you will be able to mix the various modes on the screen at the same time. Just think how nice your programs could look with a mix of large and small text, and both high and low resolution graphics. This program has many examples plus does all of the difficult calculations!

#2: HORIZONTAL/VERTICAL SCROLLING — The information you put on the screen, either graphics or text, can be moved up, down or sideways. This can make for some nice effects. You could move only the text on the bottom half of the screen or perhaps create a map and then move smoothly over it by using the joystick.

#3: PAGE FLIPPING — Normally you have to redraw the screen every time you change the picture or text. Now you can learn how to have the computer draw the next page you want to see while you are still looking at the previous page, then flip to it instantly. You won't see it being drawn, so a complicated picture can seem to just appear. Depending on your memory size and how complicated the picture, you could flip between many pages, thus allowing animation or other special effects with your text.

#4: BASICS OF ANIMATION — Shows you how to animate simple shapes using the PRINT and PLOT commands, and also has a nice little PLAYER/MISSILE Graphics demo to learn. This would be an excellent way to start making your programs come alive on the screen.

#5: PLAYER MISSILE GRAPHICS — This complex subject will be demonstrated by starting with simple examples, and building up to a complete game and also an animated business chart on multiple pages! As always, the computer does most of the calculations. Ready Nov. 1981. Requires 32 k disk or tape and costs 29.95 (extensive manual included)

Tricky Tutorials (except #5) require 16k memory for cassette orders and 24k for disk. The price is $14.95 each. You may order 1,2,3, & 4 for $49.95!

MASTER MEMORY MAP(tm) — is really the key to using the ATARI's capabilities. We start out by explaining how to PEEK and POKE values into memory so that even new programmers can use this. Then, we give you over 15 pages of the memory locations that are the most useful. The information is condensed from both the ATARI's Operating System Manual and various articles and programs. It is, of course, useful even for experienced programmers as a reference. Also, we highly suggest that dealers offer this Memory Map to customers who request to be told how to use the power of the machine. We guarantee it will answer many of the questions you have about the machine. **$5.95**

THE GRAPHICS MACHINE!! — Turn your computer into an incredible graphics tool with advanced comands like circle, box, fill, polygon, line, help, etc. 3 colors in graphics 8 with instant text!!! Create colorful business charts or beautiful drawings and then save or retrieve them from disk in 5 SECONDS. YES, it's that fast! Needs all 48k, disk, and costs **$19.95**

MINI-WORD PROCESSOR — This is for those of you who have a printer, but don't need to spend $100 or more for a fancy word processor. It is suitable for simple editing of text, accepts most control characters for your printer, and text is stored on disk for easy retrival. The amount of text held depends on memory. Requires disk and 32k for **$14.95**

BOB'S BUSINESS — 14 small business type programs for home of office, all chosen from a nice menu. Supports printed output. 169 sectors of programs require 32k, tape or disk. **$14.95.**

KID's #1 — Includes the following: 1) TREASURE — search for the lost treasure while trying to keep from falling into the sea. Nice graphics if you find it!; 2) DIALOGUE — talk back to the computer about four subjects; 3) MATH QUIZ — Nice musical and graphical rewards for good scores. Parents input the level of difficulty.

KID'S #2 — A spelling quiz, a "scrabble" type game, and a version of Touch with the computer giving all the directions! Both Kid's programs require 16k or 24k disk and cost **$14.95** each.

* * * * * * * * * * * * * *

See your local Dealer, or send check to:

SANTA CRUZ EDUCATIONAL SOFTWARE
5425 Jigger Dr., Soquel, CA 95073
C.O.D. orders call (408) 476-4901
*Include $2.00 postage ($0.75 for Memory Map)
*In Calif. include 6% tax

Santa Cruz Educational Software * Tricky Tutorials * Santa Cruz Educational Software * Tricky Tutorials * Santa Cruz Educational Software * Tricky Tutor

---

SYNCRO, INC.

SOFTWARE DIVISION
31332 VIA COLINAS SUITE 107
WESTLAKE VILLAGE, CALIF. 91361

SOFTWARE FOR ATARI T.M. 400/800

MAR TESORO © 1981

A NEW TYPE OF ADVENTURE. SIT BACK WITH YOUR JOYSTICK AND DIVE THE DEEP BLUE SEA. RENT RIG AND CREW TO PLUNDER THE STRANGE TREASURES THAT AWAIT YOU.
JOYSTICK REQUIRED
24K Tape $16.95 — 32K Disk $21.95

MAZE OF DEATH © 1981

ARE YOU A SUPERAGENT? A FAST ACTION, ALL GRAPHIC GAME WITH PLENTY OF SPECIAL EFFECTS IN A MAZE LIKE UNDERGROUND COMPLEX. FIVE LEVELS OF PLAY IN REAL TIME.
JOYSTICK REQUIRED
16K Tape $12.95 — 24K Disk $17.95

DRAC IS BACK © 1981

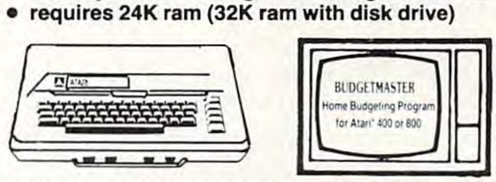
A NON-SCROLLING ADVENTURE TYPE GAME. SILVER BULLETS, WOODEN STAKES, CROSS. YOU'LL NEED THESE AND MORE TO DEFEAT THE GUEST IN DRAC'S CASTLE AND ESCAPE WITH HIS GOLD.
24K Tape $14.95 — 32K Disk $19.95

VISA AND MASTERCARD ACCEPTED.
CALIF. RES. PLEASE ADD 6% SALES TAX
PLEASE INCLUDE $1.50 SHIPPING
DEALER INQUIRIES WELCOME

SUNRISE SOFTWARE®

BUDGETMASTER™

- takes you, the first time budgeter, step by step through the task of setting up a home budget.
- allows for quick and easy manipulation of budget catagories, names, budget amounts, expense items, etc., always displaying all catagory names while making changes.
- allows for manipulation of figures for a whole year at a time without the need to save to tape or diskette.
- includes option for hard copy printout of the 39 different displays.
- stores information either on cassette or diskette.
- analyzes your budget and spending patterns to assist you with setting future budget amounts.
- requires 24K ram (32K ram with disk drive)



$29.95 (cassette) • $34.95 (disk) + $1.50 shipping

Ohio Residents Add Sales Tax

(DEALER INQUIRIES WELCOME)

SUNRISE SOFTWARE® • P.O. Box 25621 • Garfield Hts., Ohio 44125
(216) 581-9359

*Atari is a Trademark of Atari, Inc.

Atari Timing Delays

Editor's Note: Timer #4 will not work correctly due to an error in the Atari operating system. — RTM

Jim Clark
Seattle, WA

Timing delays are frequently needed in programs, and a common way to implement a delay in BASIC programs is to code a FOR-NEXT loop that does nothing other than to loop a specified number of times. FOR-NEXT loops are difficult to calibrate in Atari BASIC, however, because the time required for a loop will vary, depending on the location of the loop in the program and the size of the program.

I had a need for precise timing delays when I was writing a telephone dialer program for use with an autodial modem. One possible solution was to use Atari's built-in, realtime clock (for example, see "Real-Time Clock on the Atari" by Richard Bills, **COMPUTE!** #12, pg. 88.) Coding the delay would involve setting the clock to zero with POKE statements and then using PEEK statements in a loop to compare the current time with the amount of time that must pass. This procedure is awkward and I discovered that the Atari provides a better way.

In addition to the realtime clock, which measures time in the usual "forward" direction, the Atari also contains several countdown timers, which measure time "backwards" to zero from some initial value, which you set. When a timer has counted down to zero, the Atari "rings a bell," so to speak, just like a kitchen timer used for cooking. The timers are updated 60 times per second, which permits the coding of delays in multiples of 1/60 of a second (.0167 second).

One limitation to the use of the timers is that they can only be accessed from an assembly language program. There is no way to set a timer by way of POKE statements in BASIC. Program 1 shows an assembly language subroutine called TIMER which can be used to set system timer number three. The subroutine is POKEed into memory and then called with a BASIC USR statement. The example program asks for a value in seconds. The program multiplies the seconds by 60 to get the number of "ticks," which is used to set the timer. The TIMER subroutine sets the timer and then waits until it counts down to zero, at which point TIMER returns to the BASIC program. If you run the program and enter, say, 1 second, you will observe a one second delay, and then the program will ask for another delay value. Enter 10 and it will wait for 10 seconds until it asks again, and so on. You must enter a number greater than zero, and the longest delay you can use is 65,536 "ticks," which is 1,092 seconds, or about 18 minutes.

The listing has assembly language codes in DATA statements interspersed with REM statements in order to clearly show the assembly language subroutine. Lines 100 to 140 POKE the subroutine into page six, which is an area of RAM that is reserved for applications programs and which is not used directly by BASIC. You may have other uses for page six, so an alternative way to store the assembly language program is in a string array. You can put the lines shown in Program 2 into your program as an alternative to the lines with the same numbers in Program 1. In Program 2, the USR function is given the address of the string (ADR(TIMER$)) instead of the address of page six (1536) as in Program 1.

Program 2.

```
110 DIM TIMER$ (25)
120 FOR I=1 TO 25
130 READ BYTE:TIMER$ (I)=CHR$ (BYTE)
140 NEXT I
240 Z=USR(ADR(TIMER$),TICKS)
```

TIMER could be modified to return immediately, with the timer ticking, by deleting lines 650 through 690. (Also change the 24 in line 120 of Program 1 to 19). In this case, your BASIC program must PEEK location 554 to find out if the amount of time you set has passed. IF PEEK(554) =0 THEN time is up, otherwise IF PEEK(544)<>0 THEN time is not up yet.

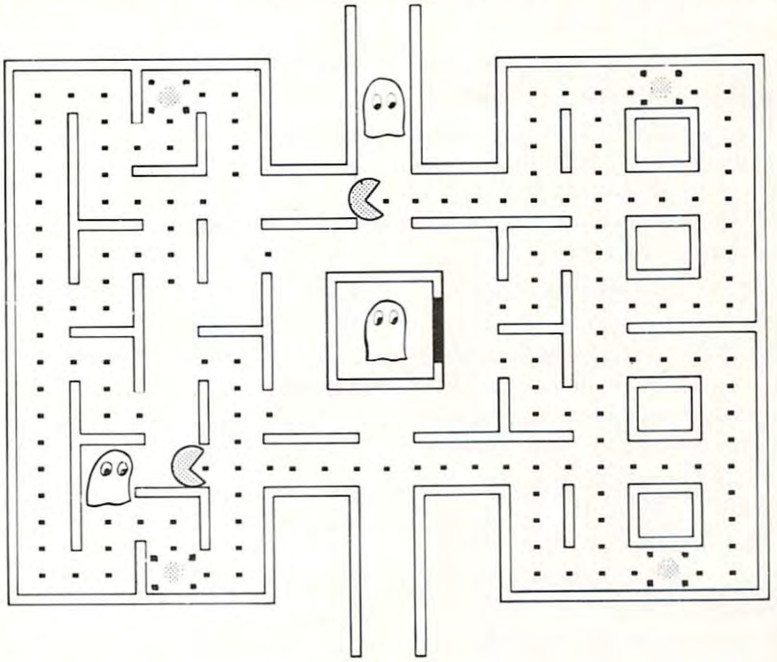
The Atari has five built-in timers, which creates the potential for control of very complex situations. Two of the timers "ring their bells" by calling a subroutine (which you can provide) and three "ring their bells" by setting a memory location to 0. Timer number three works this second way — the TIMER subroutine in Program 1 sets the memory location to a nonzero value at line 610, and then just repeatedly examines the location at line 650. When the value at the location goes to zero, the branch at line 670 will no longer be taken, and the subroutine returns to the BASIC program.

Timers one and two, which call a subroutine upon reaching zero, must have the subroutine entry address placed at a special location in memory, as shown in the following table:

Timer	Subroutine Entry Atari Name	Address Address
1	CDTMA1	$0226
2	CDTMA2	$0228

Timers three, four, and five use flags at the memory locations shown in the following table:

GHOST HUNTER™



THE ULTIMATE GAME OF HIDE AND SEEK.

In the beginning. . . there was Star Raiders.™

Then. . . came Missile Command.™

Now. . . Arcade PLUS introduces Ghost Hunter,™ the first in a new generation of software for the Atari® 400/800™ personal computers.

Your mission in Ghost Hunter™ is simple—rid the mansion on Huckleberry Hill of ghosts. . . before they get you!

Ghost Hunter™ begins where most computer games end!

- 51 game variations!
- One or two player game. . . you can play alone or head-to-head against another player!
- Choose from 16 different "floor plans" or let the computer randomly choose for you!
- Special features allow you to "Build-A-Hunter" and "Hide-Instantly"!
- Fast paced, 10 color action with 4-channel music and simulated sound effects!

Ghost Hunter™ is available on cassette or disk for Atari® 400/800™ computers with 16K minimum memory from your local Atari computer dealer. Or send $29.95 (cassette)/$34.95 (disk) + $2.50 postage and handling (CA residents please add 6% sales tax).

arcade PLUS

5276 Hollister Avenue Suite 208 Santa Barbara, CA 93111 (805) 683-2305

Timer	Atari Name	Flag Address
3	CDTMF3	$022A
4	CDTMF4	$022C
5	CDTMF5	$022E

The Atari Operating System provides a special subroutine called SETVBV to set the timers. (The need to use this subroutine is the reason that the timers can not be set directly from BASIC.) To use SETVBV, either put the address of the subroutine you want to have executed when the time is up into CDTMA1 or CDTMA2 (in the case of timer one or timer two,) or set one of the flags CDTMF3, CDTMF4, or CDTMF5 to a nonzero value (in the case of timers three, four, or five). You must then load the high-order byte of your countdown time into index register X, load the low-order byte of the time into index register Y, and load the number of the timer you wish to use (one through five) into the accumulator (register A). Finally, code a JSR SETVBV to set the timer. The address of the SETVBV subroutine is $E45C. Lines 430 through 640 in Program 1 illustrate this calling procedure.

One final note — you should probably not use timer one. The Atari Operating System itself uses this timer in its input-output routines. If you attempt to use it, you may interfere with the operation of your system.

Program 1.

```
10 REM DELAY.BAS
20 REM DEMONSTRATE USE OF ATARI COUNT-DO
WN TIMER FROM BASIC
30 REM BY JIM CLARK
100 REM PUT ASSEMBLY SUBROUTINE "TIMER"
ON PAGE 6
110 PAGE6=1536
120 FOR ADDR=PAGE6 TO PAGE6+24
130 READ BYTE:POKE ADDR,BYTE
140 NEXT ADDR
200 REM DEMONSTRATION OF CALL TO "TIMER"

210 PRINT "SECONDS DELAY";
220 INPUT SECONDS
230 TICKS=60*SECONDS
240 Z=USR(PAGE6,TICKS)
250 GOTO 200
300 REM SUBROUTINE TIMER
310 REM CALLING SEQUENCE FROM BASIC:
320 REM Z=USR(ENTRY,TICKS)
330 REM
340 REM RETURNS TO BASIC AFTER TICKS/60
SECONDS
350 REM SUBROUTINE IS COMPLETELY RELOCAT
ABLE
360 REM SO ORIGIN FOR ASSEMBLY IS NOT SH
OWN HERE
370 REM SYMBOL EQUATES TO ATARI OPERATIN
G SYSTEM:
380 REM CDTMF3=$022A       ;FLAG FOR SYST
EM TIMER #3
390 REM SETVBV=$E45C       ;ENTRY POINT F
OR SYSTEM ROUTINE
400 REM ;                  TO SET TIMERS
410 REM PLA
420 DATA 104
430 REM PLA                ;GET HIGH-BYTE
 OF TICKS
440 DATA 104
450 REM TAX
460 DATA 170
470 REM PLA                ;GET LOW-BYTE
OF TICKS
480 DATA 104
490 REM TAY
500 DATA 168
510 REM BNE TIMER          ;MAKE SURE TIC
KS>0
520 DATA 208,4
530 REM TXA                ;LOW BYTE=0, H
OW ABOUT HIGH BYTE?
540 DATA 138
550 REM BNE TIMER
560 DATA 208,1
570 REM RTS                ;TICKS=0, JUST
 RETURN
580 DATA 96
590 REM TIMER LDA #3       ;USE SYSTEM TI
MER #3
600 DATA 169,3
610 REM STA CDTMF3         ;SET THE FLAG<
>0
620 DATA 141,42,2
630 REM JSR SETVBV         ;START THE TIM
ER
640 DATA 32,92,228
650 REM LOOP  LDA CDTMF3   ;WAIT TILL THE
 FLAG GOES TO 0
660 DATA 173,42,2
670 REM BNE LOOP
680 DATA 208,251
690 REM RTS                ;TIMER HAS COU
NTED DOWN TO 0, RETURN
700 DATA 96
999 END
```

©

GREAT GAMES FOR ATARI

Gin Rummy 3.0

The classic computer Gin Rummy game now available for Atari, with color graphics and sound. Plays a regulation game, and a tough game that will hold its own against anyone. Keeps score to game level. Disk version keeps a running score in disk file so you can start another session where you left off. 24K Cass. IGR/C $19.95, Disk IGR/D $24.95.

Casino Blackjack/Counter

Play at a very realistic casino table — learn to beat the house at its own game, or just play for fun following the recommended bets. You play one of five hands (the computer plays the others), and practice card counting as the cards are dealt. Choose up to 6 decks to play against, and set the dealing speed to slow, medium or fast. 16K Cass. IBJ/C $19.95, Disk IBJ/D $24.95.

Concentration

Excellent full-color graphics, great fun to play on the screen. Where *was* that sailboat you saw two turns ago? Choose up to 15 pairs of figures—a smaller game is fascinating for children, and 15 pairs will challenge anyone. For 2 players. 16K Cass. ICO/C $14.95.

Atari 3-Game Pack

These three great Atari games — Gin Rummy, Blackjack and Concentration — on a single disk, and at less than the cost of the three separately. $49.95.

ALL PROGRAMS OPERATE WITH JOYSTICKS

AT YOUR DEALER OR DIRECT FROM:

MANHATTAN SOFTWARE
POST OFFICE BOX 1063
WOODLAND HILLS, CA 91365
California residents add 6% sales tax
24-hour Visa and MasterCard order line:
(213) 704-8495

RAM

For ATARI

Get the most from your ATARI 400. Memory expansion to a full 48K is now possible with our 48K Board. Expand your ATARI 800 with our 32K Board.

48K Board $299
32K Board $199

INTEC
Suite # 111
3387 Del Rosa Ave. North
San Bernardino, CA 92404
(714) 864-5269

CA residents add 6 percent tax

ATARI is trademark of ATARI, Inc.

ATARI™
400*& 800*

Computer Software

Announcing:

KAYOS

$34.95

IN 8K MACHINE LANGUAGE DISK OR TAPE.

*The longest lines at the Video/Electronics show in N.Y. were people waiting to play KAYOS!

CM 1001 — HORSEWORD DERBY**	**$19.95**
CM 1002 — AVOID**	**$19.95**
CM 1003 — TRIBOMB**	**$19.95**
CM 2001 — KAYOS	**$34.95**

**Requires 16K of RAM

TO ORDER SEND $1.50 SHIPPING

COMPUTER MAGIC LTD.
40 Wakefield Ave.
Port Washington, NY 11050
Dealer Inquiries
516-883-0094

educational software for the ATARI* educational software for the APPLE II+**

HIDDEN WORDS

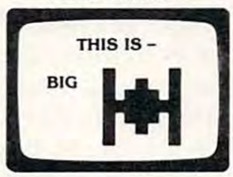
A word-search program for the ATARI*. Move your character over the letters that spell your word. Words may be read from left, right, up, down, or diagonally.

- For one or two players
- Four levels of play
- Comes with word base of the most frequently misspelled words.
- Requires JOYSTICKS

24K BASIC, Grades 1-8+ . . . **$17.50**

SPATIAL RELATIONS

For the ATARI*



Designed to help the child learn words that indicate size and placement. Units cover: big/small, tall/short, right/left, high/low, over/under, in/out. Requires one joystick. Minimal reading. SIDE TWO REQUIRES NO READING - Use with Type-'N-Talk™ speech synthesizer. 16K BASIC, Preschool-1 . . . . **$17.50**

Type-N'-Talk™*** speech synthesizer (VOTRAX). Let your computer talk with one of the easiest to use speech synthesizers on the market.
Requires RS-232 interface. **$375.00**

T.H.E.S.I.S.
P.O. Box 147 • Garden City, MI 48135
or call (313) 595-4722 for C.O.D.
Some T.H.E.S.I.S. software is now available for the APPLE II+
DEALER INQUIRIES WELCOME
Please add: $1.50 shipping/handling, $1.50 for C.O.D., Mich. res. add 4% sales tax.
WRITE FOR FREE CALALOG

*ATARI is trademark of Atari, Inc. **APPLE is trademark of Apple Computer, Inc. ***Type-'N-Talk is trademark of Votrax.

educational software for the ATARI* educational software for the APPLE II+** educational software for the ATARI* educational software for the APPLE II+** educational software for the ATARI* educational software for the APPLE II+** educational software for the ATARI* educational software for the APPLE II+** educational software for the ATARI* educational software for the APPLE II+** educational software

Printing Numbers That Make Cents

Tina Halcomb
Carrolton, TX

In the *Atari Basic Reference Manual* (page 39), the section on String Manipulations shows a way to print selected portions of a string. That is, sometimes we need to control the size of items that are being printed due to limited space. Or maybe we just want a prettier, more uniform format.

When dealing with numbers — complex computations and so on — accuracy can be very important. Atari gives several digits of accuracy and, in some cases, the numbers are expressed in terms of scientific notation.

If you were to write an accounting program you would want to have the figures as accurate as possible, but, in most cases, you wouldn't need any more than two decimal places printed. This is especially true if you are speaking in terms of dollars and cents. You might also want the fraction rounded to the nearest 100th. The following formula will do this for you:

A = INT((A + .005)*100)/100

It rounds the number off to the nearest 100th and disregards anything beyond two decimal places. This works very well, however, if you were to enter a number such as 79.00, the Atari automatically deletes the decimal point and trailing zeros. They would be necessary for a consistent printout. Lines 70 and 80 of the example program check for decimal point position and pad with zeros if necessary.

You can also line up the decimal points for a uniform columnar printout. Using an X and Y cursor position statement, subtracting the length of the string from the X (right boundary) coordinate will line the decimal points up very nicely. And, of course, you will increment the Y coordinate according to the desired horizontal spacing.

To see how all of this works, type the example program in and run it.

```
10 DIM B$(20)
20 Y=6
30 GRAPHICS 0
40 INPUT A
50 A=INT((A+.005)*100)/100
60 B$="00":B$(LEN(B$)+1)=STR$(A)
70 IF B$(LEN(B$)-1,LEN(B$)-1)="." THEN B$
   (LEN(B$)+1)="0":REM check for 1 decimal
   place
80 IF B$(LEN(B$)-2,LEN(B$)-2)<>"." THEN B$
   (LEN(B$)+1)=".00":REM check for 0 decimal
   places
100 GOSUB 200
110 GOTO 40
200 X=20-LEN(B$):Y=Y+1
210 POSITION X,Y:PRINT B$(3)
220 RETURN
```

©

Formatting Input

Kenneth J. Freese
East Meadow, NY

Data entry can frequently be simplified by formatting the screen as a table and entering the data into the table. This is in contrast to the usual method of serial prompt statements. Presented here are two simple routines for this technique. Both examples are for numeric input, but can easily be adapted for strings.

Program 1 utilizes the Keyboard (K:) as the input device. The data is entered without prompt. The variable RC in lines 150, 180 and 200 represents the right margin against which the data will be placed. The subroutine beginning at line 500 accomplishes the right justification (see **COMPUTE!** #10, pg. 84). Memory location 84 ("ROWCRS") holds the value of the current cursor row so line 520 will position the screen output on the proper line. One drawback is not seeing the input until the entire is keyed in and the RETURN is pressed. Another problem is that an inadvertently pressed key cannot be DELETEd so that correction requires a routine to alter the disk file.

Program 2 uses the Screen Editor (E:) as the input. This is the more familiar technique to most programmers since it is the default mode of the input statement. Since a semicolon cannot be used to "hold" the line after an input function, the POSITION statements in lines 150 and 160 use a PEEK(84)-1 to move the cursor up to maintain the same line. The screen display is not as appealing as in Program 1 because question marks precede every entry and there is no right margin justification. However, this may be a small price to pay to see entries as they are being typed and allowing for changes in entries prior to hitting the RETURN key.

Program 1.

```
100 DIM X$(10)
110 OPEN #2,4,0,"K:"
120 OPEN #3,8,0,"D:INFO"
130 ? "  NO          CODE          PROCEDU
RE"
140 FOR X=1 TO 37:? "-";:NEXT X:? :?
150 INPUT #2;NO:X=NO:RC=6:GOSUB 500
160 ? NO;
170 IF NO=0 THEN 240
180 INPUT #2;CODE:X=CODE:RC=19:GOSUB 500

190 ? CODE;
200 INPUT #2;PROCEDURE:X=PROCEDURE:RC=34
:GOSUB 500
210 ? PROCEDURE
220 ? #3;NO;",";CODE;",";PROCEDURE
230 GOTO 150
240 CLOSE #2:CLOSE #3
250 END
500 X$=STR$(X)
510 LC=RC+1-LEN(X$)
520 POSITION LC,PEEK(84)
530 RETURN
```

Program 2.

```
100 OPEN #3,8,0,"D:INFO"
110 ? "  NO          CODE          PROCEDU
RE"
120 FOR X=1 TO 37:? "-";:NEXT X:? :?
130 INPUT NO
140 IF NO=0 THEN 200
150 POSITION 15,PEEK(84)-1:INPUT CODE
160 POSITION 28,PEEK(84)-1:INPUT PROCEDU
RE
170 ? #3;NO;",";CODE;",";PROCEDURE
180 GOTO 130
200 CLOSE #3
```

©

Typing SHOOT

Since SHOOT was published in **COMPUTE!** #16, we've received numerous letters from readers who were having problems with it. Rest assured, if you type in the Boot Tape Maker correctly, it *will* make a perfect bootable copy of SHOOT.

It is not easy to type in a long computer program listing, especially one like SHOOT, with its huge table of DATA statements. Here are some tips on how to make it easier, and what to watch out for:

1. Some beginners do not fully realize that a program must be typed in *exactly* as listed. Seemingly trivial punctuation marks like the semicolon are absolutely necessary for proper execution.

2. Look out for numbers, too. For example, in line 200 of the Boot Tape Maker, were the number 125120 mistyped, the whole effort will be flagged as an error.

3. Remember, SHOOT must run on a 16K or greater Atari.

4. Make sure the characters in quotes on line 300 are hhh, reverse-video asterisk, LV, reverse-video d.

5. Before running the Boot Tape Maker, you might want to enter "LPRINT" to insure that the cassette pointers are correct.

6. We reprint the program here in larger type.

```
0 DIM H$(1),B$(2),AD$(4),A$(60),BUF$(1148)
1 POKOFF=4*1024-1
2 GOTO 100
10 IF H$>="0" AND H$<="9" THEN D=ASC(H$)-48:RETURN
12 D=ASC(H$)-55:RETURN
20 H$=B$(1,1):GOSUB 10:B=D:H$=B$(2,2):GOSUB 10:B=B*16+D:CHECK=CHECK+B:RETURN
30 B$=AD$(1,2):GOSUB 20:AD=B:B$=AD$(3,4):GOSUB 20:AD=AD*256+B:RETURN
100 GRAPHICS 0
110 TRAP 900:LINE=1000:LSUM=0
120 READ A$:IF A$="END" THEN 200
130 ? LINE:CHECK=0:B$=A$(1,2):GOSUB 20:NOB=B
140 AD$=A$(3,6):GOSUB 30:FAD=AD
150 FOR I=1 TO NOB:B$=A$(5+2*I,6+2*I):GOSUB 20
160 M=FAD+I-1-POKOFF:BUF$(M,M)=CHR$(B):NEXT I
165 SUM=CHECK-65536*INT(CHECK/65536)
170 AD$=A$(LEN(A$)-3,LEN(A$)):GOSUB 30
180 IF SUM<>AD THEN 900
185 LSUM=LSUM+SUM:LINE=LINE+10:GOTO 120
200 IF LSUM<>125120 THEN ? "Too many/few lines":END
205 CLOSE #1
210 OPEN #1,8,128,"C:"
220 IOCB=832+16
230 POKE IOCB+2,11
240 BUF=ADR(BUF$)
250 POKE IOCB+4,BUF-(INT(BUF/256)*256)
260 POKE IOCB+5,INT(BUF/256)
270 BUFLEN=LEN(BUF$)
280 POKE IOCB+8,BUFLEN-(INT(BUFLEN/256)*256)
290 POKE IOCB+9,INT(BUFLEN/256)
300 DUMMY=USR(ADR("hhh*LVd"),16)
310 CLOSE #1
320 END
900 ? "ERROR IN#";LINE:END
1000 DATA 18100000090010081018 60A93C8D02D3A916850AA910850B604C4E06A9
```

```
1010 DATA 1810181270707046001 8F070F070F070F070F070F070F070F070F00DF0
1020 DATA 18103070F070F07041191028432931393831204A20482050414C45076D
1030 DATA 181048564943488090829284948696889 8C808182838485868788 00A5A
1040 DATA 18106038380001020304050607087C7C0102030201FFFEFDFE00000615
1050 DATA 1810780102030405060708090A0B000100010001000100010100 0100E9
1060 DATA 181090FF000003060C1C3C7EFF00C06030383C7EFF0018181818 3C077E
1070 DATA 1810A87EFF0000F820F29E90F000001F044F79090F0000010D3F7F0844
1080 DATA 1810C01800000080B0FCFE18000018247E817E000001050008FF05070D
1090 DATA 1810D80000020A0118FE0A011003190020FD190020488A48A6B0E80708
1100 DATA 1810F0BD4C108D0AD48D1AD0A6B0AD08D02901F013A9009D62109D0B70
1110 DATA 1811086D10BD78101865B185B18D1ED0E886B0BD6210187D6D109D0ACE
1120 DATA 18112062108D00D0BD57108D12D0BD83108D08D068AA6840A5B1D00B40
1130 DATA 18113808A9808D03D24C901138E90185B1A98A8D03D2A205BD00180A4A
1140 DATA 18115018690109109D0018C91AD009A9109D0018CA4C4D11A5B7D00893
1150 DATA 18116827A6B5E886B5E00FD01EA20086B5A205BD0E18186901099 00A95
1160 DATA 1811809D0E18C99A9009A9909D0E18CA4C7811A6B6E886B6E03CD00C6F
1170 DATA 18119822A20086B6A5B7D02EA205BD0E1838E90109909D0E18C99F0A8B
1180 DATA 1811B0D009A9999D0E18CA4CA311A900A2061D0D18CAD0FA290FC90AA9
1190 DATA 1811C800D004A90185B7A900854DAD78024A4AAABD8E1085B3CA8A0B72
1200 DATA 1811E00A0A0AAAA000BD921099601AC899601AE8C8C010D0F0A5B20C55
1210 DATA 1811F81865B385B28D04D0A5B4F026AAA9009D8019CAF011A5B1D00DD2
1220 DATA 1812101286B4A9FF9D80198E00D24C2A1286B44C2A12A2008E00D20A10
1230 DATA 18122886B4A5B7D016AD8402D011A5B4D00DA96285B4A5B30A0A180BE0
1240 DATA 181240698485B2A9FF85B08D1ED04CD1E7A9A88D01D2A9808D03D20E26
1250 DATA 181258A9008D00D2A9308D02D2A280A9009DFF199D7F19CAD0F7A90CB3
1260 DATA 18127000A2089DFFCFCAD0FAA92E8D2F02A9188D07D4A9038D1DD00C27
1270 DATA 181288A9100D6F028D6F028D1BD0A90085B4A90185B7A9408D0ED40A7F
1280 DATA 1812A0A9108D3102A9198D3002A9108D0102A9EA8D0002A211A03508B7
1290 DATA 1812B8A906205CE4A9C08D0ED4A9C68DC402A9368DC502A9188DC60CD2
1300 DATA 1812D002A90A8DC702A214BD3710200E1409C09DFF17CAD0F2A5130AC1
1310 DATA 1812E8186903C513D0FCA214A9009DFF17CAD0FAA9108D0518A9500C3C
1320 DATA 1813008D0C18A9908D1318A90185B7A9088D1FD0AD1FD0C901D0060A1C
1330 DATA 181318202A144C4E12C906D0EFA900A2069D0D189DFF17CAD0F7A90ADB
1340 DATA 181330918D1118A9928D1218A9908D1318A9108D0518A90085B7850952
1350 DATA 181348B185B685B5A218A00020C513C8C008D0F8A207A0D0A9038D0C95
1360 DATA 1813602A02205CE4A9C08D0ED4AD2A02D0034C4D13A008A900196C091D
1370 DATA 1813781088D0FAC900D00AA9321865B185B14C4D13A5B7F0DCA5130C73
1380 DATA 181390186902C513D0FCA200BD0718291FDD0018F005B0084CB0130959
1390 DATA 1813A8E8E006D0EC4C0813A206BDFF17291F09409D0618CAD0F34C0B64
1400 DATA 1813C00813000008EC2138CC313AD0AD22907C906B0F70A0AAABD0975
1410 DATA 1813D8D210996D10BDD3109978 10BDD4109983 10A9009962 10BDD50BCF
1420 DATA 1813F010AAACC213A9088DC413BDAA1099001AE8C8CEC413D0F3980D45
1430 DATA 181408AAACC31360008C0D14A88A48982A2A2A2A2903AA98299F1D087A
1440 DATA 181420F6FEA868AA98AC0D1460A220A90C9D42032056E4A9149D450B11
1450 DATA 18143803A9759D4403A9039D4203A9089D4A03A9809D4B032056E40900
1460 DATA 181450A9009D4403A9109D4503A9789D4803A9049D4903A90B9D4208D9
1470 DATA 101468032056E4A90C9D42032056E460433A9B0652
1480 DATA END
```

©

Atari
Owners
Stay
Tuned
To
COMPUTE!
In
1982
For
The
Latest
In
Software
Overviews,
New Products,
Applications
Articles,
And
Support

TOLL FREE
Subscription
Order Line
800-345-8112
In PA 800-662-2444

GALACTIC CHASE™

The aliens have swept undefeated across the galaxy. You are an enterprising star ship captain—the final defender of space.

As the aliens attack, you launch a deadly barrage of missiles. Flankers swoop down on your position. Maneuvering to avoid the counterattack, you disintegrate their ships with your magnetic repellers.

As your skill improves, the attackers increase their speed. And as a last resort, the aliens use their invisible ray to slow the speed of your missile launcher.

GALACTIC CHASE provides Atari owners with the most challenging one or two person game in the galaxy.

ATARI ®

Atari 400/800 16k. Written in machine language. Requires joysticks.
Payment: Personal Checks—allow three weeks to clear.
American Express, Visa, & Master Charge—include all numbers on card. Please include phone number with all orders. 24.95 for cassette or 29.95 for disk plus 2.00 shipping. Michigan residents add 4%.

Check the dealer in your local galaxy. Dealer inquiries encouraged.

Galactic Chase © 1981 Stedek Software.

SPECTRUM
COMPUTERS

Dept C.
26618 Southfield
Lathrup Village, MI. 48076
(313) 559-5252

TextPlot

Charles Brannon
Editorial Assistant

TextPlot is a machine language graphics utility that lets you mix text and graphics. It is designed to work with the four-color graphics modes three, five, and seven. It will place any ATASCII character — upper/lowercase, graphics, numbers, and special symbols in normal or reverse field — on the graphics screen in any of three colors. The size of the characters varies in proportion to the pixel size: GRAPHICS 3 characters are *four times* as large as those in GRAPHICS 7, whose characters are the same size and proportions as those in GRAPHICS (text) mode two. Through standard display list modification, any of the three sizes of text can be mixed with the other modes. TextPlot enables you to use a total of *eight* text modes. (See the description of the "bonus" text modes later.)

Text On Graphics Lines

TextPlot, unlike the text modes, can be mixed on the same line with normal graphics. You can label charts and graphs, or quickly draw pictures with the graphics characters and then embellish them. TextPlot even works with an alternate character set, so you can design special "shapes" and move them around the screen for high-speed animation. The text in GRAPHICS mode three is *huge*, a real eyecatcher. Unlike the other text modes, TextPlot lets you position any character at any possible vertical resolution (although horizontally it's the same). And all this was without modifying the display list!

Luckily, TextPlot is easy to use. You load it into memory (it goes into the reserved memory at $600 hex) with a BASIC loader or BINARY LOAD, via DOS. You then select the graphics mode to use it in with the ordinary GRAPHICS command. (TextPlot works in either full-screen or window modes.) You then "plot" each character with the command:

```
A=USR(1536,chr,color,column,row)
```

Don't let this machine language call intimidate you. It merely enables a USeR command. The other variables for the function communicate with TextPlot. If you leave one out, or add an extra one, TextPlot will ring the bell to warn you.

CHR: The ASCII value of the desired character [like ASC("K")].

COLOR: The color of the character (just like the COLOR statement, 1–3).

COLUMN: The horizontal position of the character. This depends on the mode:

Table 1

Mode	Max Columns	Max Rows
3	5	16
5	10	40
6	10	88
7	20	88
8	20	184

Row: The vertical position of the character. This also depends on the mode (see Table 1), and is the line at which you want the character to start. Remember that each character is just eight lines of dots, so they can start at any pixel position vertically. The horizontal resolution is limited by the internal storage of graphics information on the screen.

So, to place a blue capital letter "A" on the screen in GRAPHICS mode three, at the second column and tenth row, use the command:

```
A=USR(1536,65,3,2,10)
```

65 is the ATASCII value of "A"; three is the color; two is the column; ten is the row. Strings of text can be placed on the screen as well:

```
DIM T$(20)
T$="That's Incredible!"
GRAPHICS 7+16
FOR I=1 TO LEN(T$)
  A=ASC(T$(I,I))
  V=USR(1536,A,1,I,2)
NEXT I
```

Notice that you can use any variable with the USR function, not just A.

Bonus Text Modes

TextPlot was designed for the four color graphics modes. Strange things can happen if you use it in any other mode. In modes six and eight, however, you will indeed get text. In GRAPHICS 6, the characters are the same size as those in GRAPHICS 5. There is a blank line between each row of dots in each character. A character plotted in COLOR 1 or COLOR 2 will also skip horizontally. COLOR 3 will create characters divided into "bands." The effect is similar to the IBM logo (See Figure 1). This same oddity results in "artifacting" in GRAPHICS 8. What does that mean? You get three colors of text in GRAPHICS 8! Depending on background and dot colors, COLOR 1 is purple, COLOR 2 is green, and COLOR 3 is white. The text is twice the width of GRAPHICS 0 characters, but the same height, just like GRAPHICS 1.

I have included a sample program that lets you type on the screen using a flashing cursor. It works in GRAPHICS mode seven. You can use all the standard keys, but only a few of the editing keys work. What can I say? It's not supposed to be a word processor! The lines from 20000 and up will place TextPlot into memory at page six. You can save them to disk or tape and merge them with other programs using the LIST/ENTER combination.

For Cassette	For Disk
Rewind cassette, press PLAY & RECORD, and enter:	Enter:
LIST "C:",20000,32767	LIST "D:TXTPLT.ENT", 20000,32767
Press RETURN twice.	and press RETURN.
To merge with a program already in memory:	
Rewind tape, press PLAY, enter:	Enter:
ENTER "C:"	ENTER "D:TXTPLT.ENT"
and press RETURN twice.	

Advanced readers may want to know how TextPlot works (if you haven't figured it out already). You are probably familiar with how to plot characters on the GRAPHICS 8 screen by PEEKing the character generator and then placing these bit patterns directly into the screen memory for GRAPHICS 8. It works because each byte in GRAPHICS 8 (and modes four and six, too) displays eight dots, or pixels. A one-bit in the byte means a "lit" pixel and a zero is a dark ("background") dot. The four color modes have to split the load between two bytes. Each byte displays four pixels. Two bits hold the color (binary): 01 color one, 10 color two, 11 color three. (See Figure 2.) TextPlot uses the character generator (indirectly through CHBAS, 756 decimal) to get the bit map and then "pulls" the byte accordian-style into two color bytes. Theoretically, any character could be a mixture of the three colors, but it's hard to implement and use. (Unless you use IR modes 4 or 5...)

Using TextPlot as a BINARY FILE

The Atari DOS lets you store machine language files on the disk and load them back, both by DOS menu selections. You can even have TextPlot load in automatically with the DOS, if you're sure you'll always need it. After placing TextPlot into RAM, go to DOS with the command: DOS. If you have DOS 2.0S, there will be a pause as the Disk Utility Package loads. The DOS menu should be displayed. Type K ‹RETURN›. After the prompt, enter:

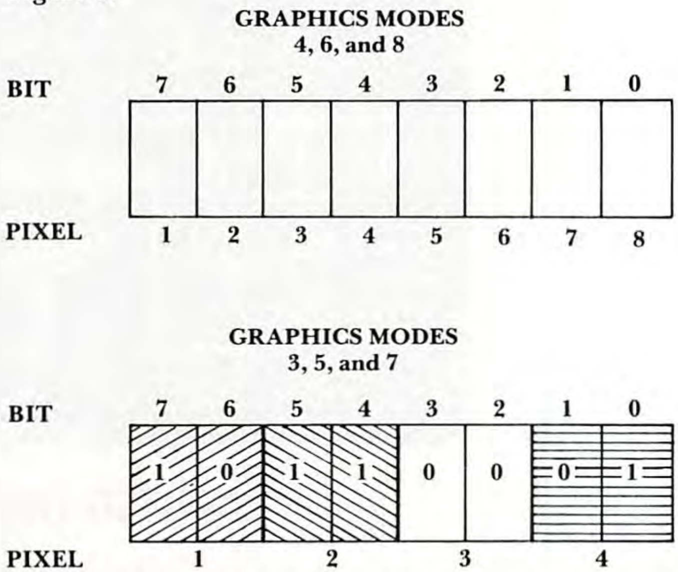
TXTPLT.OBJ,0600,06FF ‹RETURN›

If you want TextPlot to automatically load with DOS, enter:

AUTORUN.SYS,0600,06FF ‹RETURN›

instead. If you don't do this, you'll have to go to DOS and enter L (Load) and reply with TXTPLT.OBJ to load it and B ‹RETURN› to exit to BASIC.

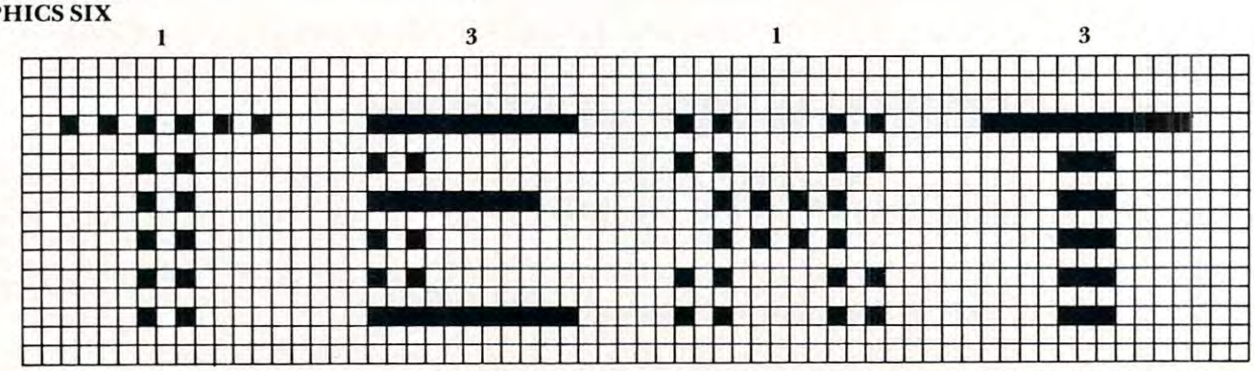
Figure 2.




```
10 REM SUPER SCREEN-TextPlot Demo
20 REM Use all the ATARI characters
30 REM including cursor up/down
40 REM left/right, backspace, RETURN,
50 REM etc.  Press CAPS/LOWR to
60 REM select upper or lower case,
70 REM as usual.  Atari Logo key
80 REM toggles reverse field.
90 REM Press console buttons for
       different colors
100 REM ESC switches modes (7 vs.8)
110 ML=1536
120 IF PEEK(ML)=0 THEN GOSUB 20000
130 XL=19:YL=11:DIM CHAR$(480),C$(480)
140 CHAR$(1)=" ":CHAR$(480)=" ":CHAR$(2)
=CHAR$:C$=CHAR$
```

Figure 1.

GRAPHICS SIX



```
150 GRAPHICS 7+G+16:OPEN #1,4,0,"K:"
155 IF G=1 THEN SETCOLOR 2,0,0:YL=23
160 LM=1:X=LM:Y=0:C=1
170 POS=X+Y*20+1:CHR=ASC(CHAR$(POS,POS))
:RVS=CHR:SC=ASC(C$(POS))-31
180 POKE 20,0:RVS=RVS+128:IF RVS>255 THE
N RVS=RVS-256
190 A=USR(ML,RVS,C,X,Y*8)
200 IF PEEK(764)<>255 THEN 300
201 T=PEEK(53279):IF T=6 THEN C=1
202 IF T=5 THEN C=2
203 IF T=3 THEN C=3
210 IF PEEK(20)<15 THEN 200
220 GOTO 180
300 A=USR(ML,CHR,SC,X,Y*8)
310 GET #1,KEY:E=0:DL=E
320 IF KEY>31 AND KEY<123 THEN 500
325 IF KEY=ASC("{CLEAR}") THEN CLOSE #1:
GOTO 140
326 IF KEY=ASC("{ESC}") THEN CLOSE #1:G=
1-G:GOTO 140
330 IF KEY=ASC("{UP}") THEN Y=Y-1:E=1
340 IF KEY=ASC("{DOWN}") THEN Y=Y+1:E=1
350 IF KEY=ASC("{LEFT}") THEN X=X-1:E=1
360 IF KEY=ASC("{RIGHT}") THEN X=X+1:E=1

380 IF KEY=155 THEN X=LM:Y=Y+1:E=1
390 IF KEY=ASC("{BACK S}") THEN X=X-1:KE
Y=32:DL=1
400 IF X<LM THEN X=XL:Y=Y-1
410 IF X>XL THEN X=LM:Y=Y+1
420 IF Y>YL THEN Y=0
430 IF Y<0 THEN Y=YL
440 IF E THEN 170
500 A=USR(ML,KEY,C,X,Y*8)
510 POS=X+Y*20+1:CHAR$(POS,POS)=CHR$(KEY
):C$(POS,POS)=CHR$(31+C)
520 IF DL=0 THEN X=X+1:IF X>XL THEN X=LM
:Y=Y+1:IF Y>YL THEN Y=0
530 GOTO 170
20000 ML=1536:FOR I=0 TO 252:READ A:POKE
 ML+I,A:NEXT I:RETURN
20010 DATA 104,240,10,201,4,240
20020 DATA 11,170,104,104,202,208
20030 DATA 251,169,253,76,164,246
20040 DATA 104,133,195,104,201,128
20050 DATA 144,4,41,127,198,195
20060 DATA 170,141,250,6,224,96
20070 DATA 176,15,169,64,224,32
20080 DATA 144,2,169,224,24,109
20090 DATA 250,6,141,250,6,104
20100 DATA 104,141,251,6,104,104
20110 DATA 141,252,6,14,252,6
20120 DATA 104,104,141,253,6,133
20130 DATA 186,166,87,169,10,224
20140 DATA 3,240,8,169,20,224
20150 DATA 5,240,2,169,40,133
20160 DATA 207,133,187,165,88,133
```

```
20170 DATA 203,165,89,133,204,32
20180 DATA 228,6,24,173,252,6
20190 DATA 101,203,133,203,144,2
20200 DATA 230,204,24,165,203,101
20210 DATA 212,133,203,165,204,101
20220 DATA 213,133,204,173,250,6
20230 DATA 133,187,169,8,133,186
20240 DATA 32,228,6,165,212,133
20250 DATA 205,173,244,2,101,213
20260 DATA 133,206,160,0,162,8
20270 DATA 169,0,133,208,133,209
20280 DATA 177,205,69,195,72,104
20290 DATA 10,72,144,8,24,173
20300 DATA 251,6,5,208,133,208
20310 DATA 224,1,240,8,6,208
20320 DATA 38,209,6,208,38,209
20330 DATA 202,208,228,104,152,72
20340 DATA 160,0,165,209,145,203
20350 DATA 200,165,208,145,203,104
20360 DATA 168,24,165,203,101,207
20370 DATA 133,203,144,2,230,204
20380 DATA 200,192,8,208,183,96
20390 DATA 169,0,133,212,162,8
20400 DATA 70,186,144,3,24,101
20410 DATA 187,106,102,212,202,208
20420 DATA 243,133,213,96,0,1
20430 DATA 28
```

©

Assembler Update

Eric Brandon

In my previous article, **Assembler in BASIC,** I presented a program that allowed you to enter, edit, and assemble machine language programs. One feature not included in the program was the ability to SAVE and LOAD your source code. I have since realized that this is a very useful capability to have, so I have written this short patch that gives the assembler SAVE and LOAD commands.

LOAD your old version, and type in the program lines you see below.

From now on, when in the menu, a 0 will allow you to LOAD a file from disk, and an S will allow you to SAVE a file to disk. The routine is written for DOS 1.0 disks. To use it with cassette type in:

```
11010 OPEN 8,1,0,FL
12010 OPEN 8,1,1,FL
```

Open last note — 8K users should not use this routine since their memory is quite full with the original program. If however, they can convince a friend to run it through the COMPACTOR program in **COMPUTE!** SEP/OCT 1980, p. 104, they may end up with more bytes free than when they started!

©

Maypole

Louis & Helen Markoya
Shelton, CN

Clyde Spencer's article on graphics modes 9, 10 and 11 **COMPUTE!** #14, pg. 120) is the first documentation I know of concerning these "mystery" modes. As the programs he wrote show, these modes offer great color versatility. For a very attention-getting display, add the following program to his *PALETTE* program with the subroutine TEN.

```
200 C=1
201 REM: START COLOR NUMBER
210 FOR I=1 TO 75 STEP 3
211 REM: DRAW TO THESE PLACES
220 COLOR C
221 REM: SETS COLOR VALUE FOR EACH LINE
230 PLOT 40,5
231 REM: TOP OF MAYPOLE
240 DR.I, 180
241 REM: MAYPOLE RIBBONS
250 C=C+1
251 REM INCREMENT COLOR NUMBER
260 NEXT I
261 REM: DRAWS NEXT RIBBON
270 A=PEEK (712)
271 REM: LOOK AT COLOR VALUE IN
    REGISTER 4
280 POKE 712, PEEK (710)
281 REM: POKE REGISTER 4 WITH LOWER
    VALUE IN REGISTER 2
290 POKE 710, PEEK (709)
291 REM: POKE REGISTER 2 WITH COLOR
    VALUE IN REGISTER 1
300 POKE 709, PEEK (708)
301 REM: POKE REGISTER 1 WITH COLOR
    VALUE IN REGISTER 0
310 POKE 708, A
311 REM POKE REGISTER 0 WITH ORIGINAL
    COLOR VALUE IN REGISTER 4
320 FOR I=1 TO 20:NEXT I
321 REM: GIVE TIME TO LOOK AT DISPLAY
330 GOTO 270
331 REM: ROTATE COLOR VALUES
    HIT BREAK TO STOP DISPLAY.
```

Color rotation is one effective way of creating animation easily. The above program not only rotates the Maypole's "ribbons," but also moves the color display from the top of the screen to the bottom. Enjoy and experiment: these bonus graphic modes certainly call for that. ©

Atari Program Listings

With this issue, **COMPUTE!** is starting a new, standardized Atari program listing format. All the editing and cursor-control characters are spelled out (e.g., CLEAR for clear screen) and surrounded by brackets.

Other characters, such as CTRL-T, the "ball" character, will be listed as the "normal" character within brackets: T . A series of identical control characters will be indicated by a number within the brackets, e.g. 5 DOWN for five cursor downs and 12 R for twelve CTRL-R's. Two control characters, {=} and {–} should be shifted. Any reverse-field text will be enclosed in vertical lines, | like this |. (Press the Atari logo key (Ⓐ) for each vertical line.) We expect that this convention will permit easy, unambiguous program typing.

THE OSI® GAZETTE

OSI Relocation Or What's NEW?

Mark Guzdial
Royal Oak, MI

After reading Elizabeth Deal's article on "Relocation of BASIC Programs on the PET" in **COMPUTE!** #13, I tried to implement the program relocation on my OSI C1P. After doing some POKEing around, both on and off-line, I found not only how very simple it was to do on an OSI machine, but also some observations about how the NEW command works.

Ms. Deal broke the relocation process down into four steps:

1. Set up one or more partitions
2. Adjust all BASIC pointers
3. Change the tape header information
4. Correct the forward pointers in the relocated program.

Since OSI machines store programs on tape in ASCII (not in the internal format) steps (4) and (5) are unnecessary. All one needs to relocate is to set up the partitions and to adjust the pointers.

In Williams and Dorner's *The First Book of OSI*, doing just that is described. In an example in that book, the authors show how a new workspace may be created by two (or three) POKEs and the NEW statement. One needs to adjust the pointer at $079, $007A (decimal 132,133), the beginning of BASIC workspace pointer, and to POKE in the initial null (Williams and Dorner do this with only two POKEs in the book, but normal use would probably entail three POKEs). So, if one needed to create a workspace at $1000, $0079 should be set to $01 (one byte past the initial null), $0071 should be set to $10, and a null should be inserted at $1000. After these POKEs, NEW should be typed to reset all other necessary pointers.

A normal LOAD should then load the program on tape into the new workspace, provided enough memory is left to hold it.

To return to a program LOADed into a new workspace, change the beginning of Basic workspace pointer with POKEs and type RUN.

What Else Was NEW Doing?

What I became interested in was the effect of the NEW upon the process. What other pointers were being reset? What else was NEW doing? Was it possible to relocate a BASIC workspace merely with POKEs?

I quickly found out that just leaving out the NEW only caused a machine crash. Taking a clue from Harvey Herman's article in the same issue of **COMPUTE!**, I tried modifying the single variable storage pointer, which is the same as the end of text pointer. In a normal cold start the end of text pointer ($007B, $007C) points to a memory address two bytes past whatever address $0079,7A points at. That is, after just cold-starting my C1P, $0079,7A points at $0301 (768) and $007B,7C points at $0303 (770). So, I tried setting my beginning of workspace to $1001 and my end of text pointer to $1003.

Well, it didn't crash — immediately. Obviously, there was something else the NEW statement did.

Checking around, I found that in a cold start, and in a NEW, addresses $0300 to $0302 are nulled out. So, I tried adding the POKEs to null $1001 and $1002 to the POKEs I had already.

Success! By POKEing the beginning of text pointer, the end of text pointer and nulling out the first three bytes of my new workspace, I could seemingly relocate my BASIC workspace without using the NEW command. But seven pokes seemed quite a bit more work than three pokes and a NEW statement, so I decided to see if any of the three nulls could be removed.

I found that the initial null could be left out to enter a program and to LIST it, but the RUN command generated a syntax error. If the null at $0302, or in the modified workspace $1002, was left out, the program would not LIST. But the null at $0301 or $1001 could be left out with no obvious problems.

$0301 and $0302 are pointers to the second line in the program in the 6502 format of least significant byte (LSB), most significant byte (MSB) for addresses. I knew that both being nulled indi-

OSI AARDVARK OSI

NOW MEANS BUSINESS!

WORD PROCESSING THE EASY WAY— WITH MAXI-PROS

This is a line-oriented word processor designed for the office that doesn't want to send every new girl out for training in how to type a letter.

It has automatic right and left margin justification and lets you vary the width and margins during printing. It has automatic pagination and automatic page numbering. It will print any text single, double or triple spaced and has text centering commands. It will make any number of multiple copies or chain files together to print an entire disk of data at one time.

MAXI-PROS has both global and line edit capability and the polled keyboard versions contain a corrected keyboard routine that make the OSI keyboard decode as a standard typewriter keyboard.

MAXI-PROS also has sophisticated file capabibilities. It can access a file for names and addresses, stop for inputs, and print form letters. It has file merging capabilities so that it can store and combine paragraphs and pages in any order.

Best of all, it is in BASIC (0S65D 5 1/4" or 8" disk) so that it can be easily adapted to any printer or printing job and so that it can be sold for a measly price.

MAXI-PROS – $39.95

NEW-NEW—NEW TINY COMPILER

The easy way to speed in your programs. The tiny compiler lets you write and debug your program in Basic and then automatically compiles a Machine Code version that runs from 50-150 times faster. The tiny compiler generates relocatable, native, transportable machine code that can be run on any 6502 system.

It does have some limitations. It is memory hungry – 8K is the minimum sized system that can run the Compiler. It also handles only a limited subset of Basic – about 20 keywords including FOR, NEXT, IF THEN, GOSUB, GOTO, RETURN, END, STOP, USR(X), PEEK, POKE, -,=,*,/, ( ), <>,Variable names A-Z, and Integer Numbers from 0-64K.

TINY COMPILER is written in Basic. It can be modified and augmented by the user. It comes with a 20 page manual.

TINY COMPILER – $19.95 on tape or disk

THE AARDVARK JOURNAL

FOR OSI USERS – This is a bi-monthly tutorial journal running only articles about OSI systems. Every issue contains programs customized for OSI, tutorials on how to use and modify the system, and reviews of OSI related products. In the last two years we have run articles like these!

1) A tutorial on Machine Code for BASIC programmers.
2) Complete listings of two word processors for BASIC IN ROM machines.
3) Moving the Directory off track 12.
4) Listings for 20 game programs for the OSI.
5) How to write high speed BASIC – and lots more –

Vol. 1 (1980) 6 back issues - $9.00
Vol. 2 (1981) 2 back issues and subscription for 4 additional issues - $9.00.

ACCOUNTS RECEIVABLE – This program will handle up to 420 open accounts. It will age accounts, print invoices (including payment reminders) and give account totals. It can add automatic interest charges and warnings on late accounts, and can automatically provide and calculate volume discounts.

24K and 0S65D required, dual disks recommended. Specify system.

Accounts Receivable. $99.95

* * * SPECIAL DEAL – NO LESS! * * *

A complete business package for OSI small systems – (C1, C2, C4 or C8). Includes MAXI-PROS, GENERAL LEDGER, INVENTORY, PAYROLL AND ACCOUNTS RECEIVABLE – ALL THE PROGRAMS THE SMALL BUSINESS MAN NEEDS. $299.95

P.S. We're so confident of the quality of these programs that the documentation contains the programmer's home phone number!

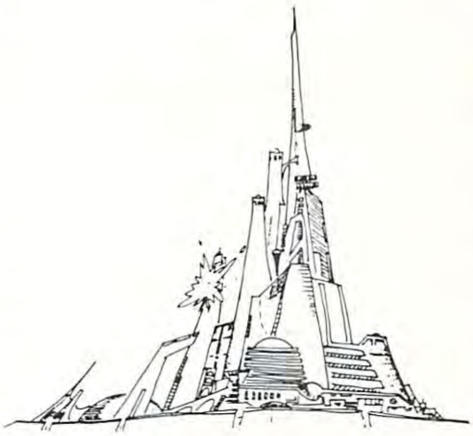
SUPERDISK II

This disk contains a new BEXEC* that boots up with a numbered directory and which allows creation, deletion and renaming of files without calling other programs. It also contains a slight modification to BASIC to allow 14 character file names.

The disk contains a disk manager that contains a disk packer, a hex/dec calculator and several other utilities.

It also has a full screen editor (in machine code on C2P/C4)) that makes corrections a snap. We'll also toss in renumbering and program search programs – and sell the whole thing for – SUPERDISK II $29.95 (5 1/4") $34.95 (8").

AND FUN, TOO!



BOOKKEEPING THE EASY WAY – WITH BUSINESS I

Our business package 1 is a set of programs designed for the small businessman who does not have and does not need a full time accountant on his payroll.

This package is built around a **GENERAL LEDGER** program which records all transactions and which provides monthly, quarterly, annual, and year-to-date PROFIT AND LOSS statements. GENERAL LEDGER also provides for cash account balancing, provides a BALANCE SHEET and has modules for DEPRECIATION and LOAN ACCOUNT computation.

GENERAL LEDGER (and MODULES) $129.95.

PAYROLL is designed to interface with the GENERAL LEDGER. It will handle annual records on 30 employees with as many as 6 deductions per employee.

PAYROLL - $49.95.

INVENTORY is also designed to interface with the general ledger. This one will provide instant information on suppliers, initial cost and current value of your inventory. It also keeps track of the order points and date of last shipment.

INVENTORY - $59.95.

GAMES FOR ALL SYSTEMS

GALAXIAN - 4K - One of the fastest and finest arcade games ever written for the OSI, this one features rows of hard-hitting evasive dogfighting aliens thirsty for your blood. For those who loved (and tired of) Alien Invaders. Specify system – A bargain at $9.95

NEW – NEW – NEW

LABYRINTH - 8K - This has a display background similar to MINOS as the action takes place in a realistic maze seen from ground level. This is, however, a real time monster hunt as you track down and shoot mobile monsters on foot. Checking out and testing this one was the most fun I've had in years! – $13.95.

NIGHT RIDER - You've seen similar games in the arcades. You see a winding twisting road ahead as you try to make time and stay on the road. NIGHT RIDER uses machine code to generate excellent high speed graphics - by the same author as MINOS.

NIGHT RIDER – $12.95 cassette only

THIEF - Another machine code goody for the C1P cassette only. You must use mobile cannon to protect the valuable jewels in the middle of the screen from increasingly nasty and trigger happy thiefs. Fast action and fun for one or two players. THIEF $13.95 on C1 cassette only!

SUPPORT ROMS FOR BASIC IN ROM MACHINES – C1S/C2S. This ROM adds line edit functions, software selectable scroll windows, bell support, choice of OSI or standard keyboard routines, two callable screen clears, and software support for 32-64 characters per line video. Has one character command to switch model 2 C1P from 24 to 48 character line. When installed in C2 or C4 (C2S) requires installation of additional chip. C1P requires only a jumper change. – $39.95

C1E/C2E similar to above but with extended machine code monitor. – $59.95

Please specify system on all orders

This is only a partial listing of what we have to offer. We now offer over 100 programs, data sheets, ROMS, and boards for OSI systems. Our $1.00 catalog lists it all and contains free program listings and programming hints to boot.

VISA

master charge

AARDVARK TECHNICAL SERVICES, LTD.
2352 S. Commerce, Walled Lake, MI 48088
(313) 669-3110

OSI OSI

cated the end of the program, i.e. there are no other lines to be pointed to. But the need for only one null at $0302 convinced me to start experimenting with nulls in the MSB of pointers later on in the program.

I found that regardless of what value the LSB contained, a null in the MSB of any line's pointer had the effect of deleting that line and all lines following.

For example, let's say that a multi-line program has been typed into an OSI C1P at the normal workspace of $0300, and the memory addresses $0301 and $0302 contain $0E and $03, respectively. This means that the second line of the program begins at address $030E, and the pointer to the third line is at ($030E,$030F). If we change $030F to a null ($00), we will not be able to LIST or RUN past the first line of the program. Everything from the second line on will be deleted. Except for changing the pointers to FREe the now unused memory, we have effectively NEWed most of our program. But, if we restored the value contained in $030F before it was nulled, we would also find our program restored, as long as we hadn't added any new lines of code which would overwrite the pseudo-NEWed code.

This means that during a RUN or LIST, BASIC does not check both the LSB and the MSB for nulls to indicate the end of the program-just the MSB. Actually, this makes sense since no real program line would exist in Page Zero memory (addresses that start with $00).

Though relocation without a NEW statement is still six pokes (four for the pointers and two for inserting nulls), it seems to me that knowing exactly what each POKE is doing is a lot safer than using such a dangerous command as NEW. Further, this information gives us a number of practical benefits such as the capability of a reversible NEW.

So these are the results of my excursion into OSI relocation. I hope users of both OSI and other BASIC systems may find my experiences to be helpful.

References

Deal, Elizabeth, "Relocation of Basic Programs on the PET", **COMPUTE!**, June 1981

Herman, Harvey B., "Memory Partition of Basic Workspace", **COMPUTE!**, Jan.–Feb. 1980 (original printing)

Williams and Dorner, *The First Book of OSI*, Aardvark Technical Services, 1980

©

Data Handling

Paul Muller
Ventura, CA

The OSI Superboard is an exceptional value for the computing power it provides. The 8K BASIC in ROM is ideal for writing number crunching programs which is my area of interest. After those first few weeks of wanton programming, however, I soon realized that an organized approach to data handling was needed. This article outlines a method I have used to structure the data handling chores for my numerical analysis programs. It is a series of BASIC routines which form a framework on which programs are built.

A completely flexible numerical program should be able to handle the following:

- Input data into the program from the keyboard
- Read data into the program off a tape
- Write data on to a tape
- Edit array data in the program
  - List any part of the string or array
  - Add elements to the string or array
  - Delete elements
  - Change elements
  - Insert elements

Program 1 demonstrates these data handling routines. In this simple example, up to 50 numbers can be placed into two one-dimensional arrays, X and Y. The data can be examined, manipulated, stored, or retrieved from tape without leaving the program. To anyone who has written a statistical program, the advantages of these routines soon become evident.

The program consumes about 1800 bytes as written (not including the REMark statements). The amount of memory required will depend on the number and size of the arrays being used. The really determined Superboarder will undoubtedly find a way to squeeze the program into even less memory.

Here is a step-by-step look at the commands available in this program. If it sounds complicated, don't worry. The program takes all of five minutes to master.

INPUT: Calling the INPUT routine allows you to enter the data into arrays X and Y from the keyboard. Simply input the values for each element in the arrays per the program query. The element number of each input is shown also. To exit from the INPUT routine, simply enter END,END in response to the query.

Now that you have entered some data into the program, you will want to check it for accuracy. Calling up the EDIT routine provides the necessary commands.

LIST asks for the beginning and ending array element numbers for the data you want listed. The output shows the element number in the left hand column, and the data in X and Y residing in those elements.

To change an incorrect entry, enter the element number and the correct values of X and Y in response to the program query.

To delete data in X and Y, enter only the element number in response to the DELETE query. The same procedure is used for the insert command, but you must enter X and Y values as well.

The ADD routine picks up where the INPUT routine left off. To exit, enter END,END in response to the query. Note that terminating the ADD routine takes you out of the EDIT mode. The CNTL function in EDIT does the same thing.

To write the data on a cassette, call up the TAPE routine. Upon hitting the WRITE command, the program will ask you to enter a title pad. Simply repeat a suitable title and date until you get an overflow signal. The title pad will be 72 characters long, and will be written on the tape ahead of the data. This serves two purposes; to identify the data and keep recorder noise between data files from being picked up by the program. (The title pad is read into the program in the READ routine, but not used.)

After the title pad is entered, the program will instruct you to start the tape. When the recorder is running, enter any number and hit RETURN; a listing of the data will follow the title pad. When the end of the data in the two arrays is reached, the program will prompt you to stop the tape. All of the SAVE and LOAD commands are taken care of by the program.

The READ routine is similar. Since you are entering new data, exit the program using the STOP command. Using the Superboard's LOAD command, locate the data file in the tape, then bring up the program. The READ routine will prompt you to start the tape, which should be in the title pad of the data file. Enter any number (quickly!), and hit RETURN. The title pad and data will begin listing on your terminal. When END,END is encountered in the data file, the program will prompt you to stop the tape. You can then edit or add to the data using the other routines in the program.

The RUN command simply jumps to the starting line of the analytical portion of the program. In this example, the mean value of the X and Y arrays is calculated and displayed. The STOP command allows you to exit the program.

This program is intended only to demonstrate one technique of handling data. There is a lot of room here for more sophistication. You may want to modify this program to allow the reading of several tape files into the same array. Or maybe you want to read data into different arrays. You could set up your program such that it would take in one set of data and write out a different set to be picked up by another program for further analysis. You could thus chain programs together in modules and increase the power of your Superboard far above its memory size. And who knows, maybe your friends with their dual disk drives and mega-byte memories will actually envy your low-cost, customized "tape operating system!"

```
10 REM***DATA HANDLING PROGRAM***
20 REM***BY PAUL MULLER 8/81*****
30 REM***DIMENSION ARRAYS***
40 DIM X(50),Y(50)
50 REM***START CONTROL ROUTINE***
60 FOR J=1TO5:PRINT:NEXT
70 PRINT"INPUT(1),EDIT(2)":PRINT"TAPE(3)
   ,RUN(4)"
80 INPUT"OR STOP(5)";C
90 ON C GOTO 110,200,480,780,850:GOTO60
100 REM***START INPUT ROUTINE***
110 N=0
120 REM***START ADD ROUTINE***
130 FOR I=N+1TO50
140 PRINT"X";I;",Y";I:INPUTX$,Y$
150 IFX$="END"THEN60
160 X(I)=VAL(X$):Y(I)=VAL(Y$)
170 N=N+1
180 NEXTI
190 REM***START EDIT ROUTINE***
200 PRINT:PRINT:PRINT
210 PRINT"LIST(1),ADD(2)";PRINT"DELETE(3)
    ,INSERT(4)"
220 INPUT"CHANGE(5),CNTL(6)";C
230 ONCGOTO250,130,300,360,430,60
240 REM***ARRAY LIST ROUTINE***
250 PRINT":INPUT"ENTER I1,I2";I1,I2
260 FORI=I1TOI2
270 PRINT" ";I;X(I);Y(I)
280 NEXTI:GOTO200
290 REM***DELETE ROUTINE***
300 PRINT:PRINT"ENTER I FOR"
310 INPUT"DELETED X, Y";I
320 FORG=ITON
330 X(G)=X(G+1):Y(G)=Y(G+1)
340 NEXT G: N=N-1: GOTO200
350 REM***INSERT ROUTINE***
360 PRINT:PRINT"ENTER I,X,Y":INPUTI3,I4,I
    5
370 FORG=I3TON+1
380 I6=X(G):I7=Y(G)
390 X(G)=I4:Y(G)=I5
400 I4=I6:I5=I7
410 NEXT G:N=N+1:GOTO200
420 REM***CHANGE ROUTINE***
430 PRINT:PRINT"ENTER I,X,Y"
440 INPUT I4,I5,I6
450 X(I4)=I5:Y(I4)=I6
460 GOTO200
470 REM***TAPE ROUTINE***
480 PRINT:PRINT"READ(1),"
490 INPUT"WRITE(2),CNTL(3)";C
500 ON C GOTO 640,520,60
510 REM***WRITE ROUTINE***
520 A=PEEK(15):POKE 15,72
530 PRINT"ENTER TITLE PAD":INPUT P$
540 INPUT"START RECORD";C:SAVE
550 PRINTP$
560 P$="-"
570 FORI=1TON
580 PRINTP$;",";X(I);",";Y(I)
590 NEXT I:PRINTP$;",END,END"
600 POKE 517,0
610 POKE 15,A:PRINT"STOP TAPE"
620 GOTO 480
630 REM***READ ROUTINE***
640 A=PEEK(15):POKE15,72
650 PRINT:INPUT"START PLAY";C
660 N=0:LOAD
670 INPUT P$
680 FORI=N+1 TO 50
690 INPUT P$,X$,Y$
700 IFX$="END"THEN 740
710 X(I)=VAL(X$):Y(I)=VAL(Y$)
720 N=N+1
730 NEXT I
740 POKE 515,0
750 PRINT"STOP TAPE":POKE 15,A
760 GOTO 60
770 REM***START ANALYTICAL PART OF PROGRA
    M***
780 Z=0:Z2=0
790 FORI=1TON
800 Z=Z+X(I):Z2=Z2+Y(I)
810 NEXT I
820 PRINT:PRINT"MEAN X=";Z/N
830 PRINT"MEAN Y=";Z2/N
840 GOTO 60
850 END
```

©

Keyboard Conversion Program For The OSI C1P

Ronald C. Whitaker
Salt Lake City, UT

One of the features advertised by OSI for the C1P and Superboard microcomputers is a lower case character set. While this feature is present, trying to make use of it can be both confusing and frustrating. When the shift lock is depressed, the alphabet is decoded as upper case characters and the upper row of keys produces numbers. The symbols on the upper row of keys are produced when either shift key is depressed. This works well when writing BASIC programs since lower case characters are not recognized by the BASIC interpreter except when used in PRINT statements. When the shift lock is released, the alphabet is decoded as lower case and the upper row of keys is decoded as garbage. The numbers can be obtained by depressing the left shift key. This also changes the alphabet to upper case. Depressing the right shift key produces only garbage. There's probably a reason somewhere for this bizarre decoding pattern, but it certainly seems illogical.

This problem with keyboard decoding didn't make too much difference to me until I tried to write a program to teach touch typing skills to my kids. Then it became very apparent that the keyboard must decode as much like a standard typewriter keyboard as possible. This means that, when the shift lock is depressed, the alphabet must decode as uppercase and the upper row of keys must produce the symbols. With the shift lock released, the alphabet will decode as lower case, and the upper row will produce numbers. Finally, both right and left shift keys must decode the same and produce upper case alphabet characters and symbols from the upper row of keys. To do this, I wrote a short machine language program to change the decoding.

This program makes use of the fact that, when a subroutine call is made to $F000, the subroutine polls the keyboard and the ASCII code of the next key pressed is placed in memory location 531. The keyboard conversion routine then examines this value and, if necessary, converts it to a standard typewriter character and places it back in memory location 531. The converted value is then PEEKed from memory by the BASIC program and POKEd to the screen. Below is a BASIC program which demonstrates the use of this subroutine. It works as follows:

Line 5 specifies the end of RAM available to BASIC

Lines 10–30 POKE the subroutine into the protected area

Line 40 clears the screen

"...it became very apparent that the keyboard must decode as much like a standard typewriter keyboard as possible."

Lines 100–115 set up the beginning of the line to be written

Line 120 loads zero page addresses 11 and 12 (USR vector) with the starting address of the machine language subroutine and calls the subroutine via the USR function.

Line 130 provides a line feed/carriage return function whenever the RETURN key is pressed.

Line 140 provides a true backspace whenever the RUBOUT key is pressed

Line 150 erases the page and starts the program at the top of the screen

It is necessary to protect the area at the top of RAM where the machine language subroutine is stored so that the BASIC program will not write over the top of it. The REMARK lines at the beginning of the program give values to use in lines 5 and 10 when other than 8K systems are used. I intentionally avoided the "free memory" from $0222 to $02FF because of its growing popularity with almost everyone who writes short machine code programs. This popularity has led to an ever increasing number of conflicts between otherwise useful programs.

If I were going to improve the program further I would do two additional things: first, I would write another machine language program to erase the page instead of using the PRINT/scroll erase that I have included here. Second, I would write another machine language program that would scroll the screen upward when the cursor reached the bottom of the page. However, these additions are beyond the scope of the rather simple demonstration program listed here and are left for the reader to implement.

The Keyboard Conversion program as described here suggests additional uses for the OSI C1P microcomputer which would be difficult and impractical with the standard keyboard decoding provided by OSI. Two that readily come to mind are typing tutor programs and character oriented word processors or text editors. The conversion subroutine has been an interesting exercise in machine language programming and use of the USR function. Hopefully, the reader will find it useful and I would be interested to hear of any uses found for it.

KEYBOARD CONVERSION PROGRAM

```
5 POKE 133,179:POKE 134,31
6 REM..FOR 16K SYSTEMS CHANGE LINE 5 TO
  "POKE 133,179:POKE 134,63"
7 REM..FOR 24K SYSTEMS CHANGE LINE 5 TO
  "POKE 133,179:POKE 134,95"
8 REM..FOR 32K SYSTEMS CHANGE LINE 5 TO
  "POKE 133,179:POKE 134,127"
10 FOR X=8116 TO 8192:READ Y:POKE X,Y:NEXT
11 REM..FOR 16K SYSTEMS CHANGE LINE 10 TO
   "FOR X=16308 TO 16384:"
12 REM..FOR 24K SYSTEMS CHANGE LINE 10 TO
   "FOR X=24500 TO 24576:"
13 REM..FOR 32K SYSTEMS CHANGE LINE 10 TO
   "FOR X=32692 TO 32768:"
15 DATA 32,0,253,173,0,223,201,255,240,34,201,253,
   240,50,173,19,2,201
20 DATA 47,208,6,169,63,141,19,2,96,201,59,144,1,
   96,201,48,176,1,96,56
25 DATA 233,16,141,19,2,96,173,19,2,201,92,144,1,
   96,201,45,176,1,96,56
30 DATA 233,32,141,19,2,96,173,19,2,201,107,144,1,
   96,201,81,176,217,96
40 FOR N=1 TO 30:PRINT:NEXT
100 PL=53411:S=0
110 IFPL>54171 THEN 40
115 POKE PL+S,128
120 POKE11,180:POKE12,31:X=USR(X):C=
    PEEK(531)
130 IFC=13THENPOKEPL+S,32:S=0:PL=PL+64:
    GOTO110
140 IFC=127ORC=159THENPOKEPL+S,32:S=S-1
    :POKEPL+S,128:GOTO110
150 IFC=10ORC=42THEN40
160 POKEPL+S,C
170 S=S+1:IFS>23THENS=S-1
180 GOTO110
```

©

THE PET® GAZETTE

Editor's Note: In this candid interview, **COMPUTE!***'s Assistant Editor, Richard Mansfield, and Commodore's new US Marketing Director, Kit Spencer, explore Kit's background and goals for Commodore US. During Kit's tenure with Commodore in England and Europe, Commodore reached a point where they held the lead in market share in every country where Kit held marketing responsibility. Many of you Commodore "old-timers" may remember those times as years of promised, but undelivered, increased support in the US. We're curious to see the impact of this and other reorientations of personnel and goals at Commodore US. At the age of 35, with seven years experience at Commodore, Kit himself qualifies as an "old-timer." We welcome him to the US. — RCL*

COMPUTE! Interview:

Kit Spencer – CBM's New US Marketing Director

COMPUTE!: *What was your background in Europe? What sort of things have you done in your life?*

KS: Where would you like to start — with my life in Commodore?

COMPUTE!: *Just a little bit earlier than that.*

KS: Should I take you through a very quick pointed history of Kit Spencer?

COMPUTE!: *Okay, that would be great.*

KS: I'm a physics graduate by original background, but I'm the worst physicist in the business. After leaving university, I worked in the Peace Corps for a year. In Uganda. Then I spent two years with a company called Bowmar which is in the packaging business, if you like. Largely in the marketing sales area with them. I then spent four years with the Phillips Corporation. Primarily in their hi-fi and TV division.

COMPUTE!: *Was this all in London?*

KS: This was, ah well, Cambridge which is about 80 miles north of London. But, basically, it was across the UK that I was concerned with. I used to work in — I was in their marketing group there. A very quick and sort of pointed history, this goes back, that must be 10 years now. We had a lovely boom time in color television. It's wonderful. We could sell every one we could make. But we said the market was going to turn down in a year's time. Let's look at what market we should get into — was there anything we could do? And one of the things that I was looking at was the calculator market at the time. They'd come down, I think, to under

$200. Well, we said we sell electronic products. We sell products around two hundred plus dollars. And I started looking around at the marketplace to see if it was right for us at the time. And I got rather excited at what I saw then. I saw, in that class of companies, three generations — those that grew up on basic electricity — cookers, freezers, lightbulbs, heating appliances, that sort of product. What we call the "white" goods, it was the massive generation of companies which grew out of that — probably General Electric over here, for example. It was the second generation that grew up out of the *valve* which was hi-fi, television, all those sort of things. Maybe RCA or something like that is an example here.

COMPUTE!: *Transistor.*

KS: Actually, the transistor was in between.

COMPUTE!: *You're thinking of the vacuum tube.*

KS: The vacuum tube?

COMPUTE!: *I'm trying to translate this for our audience here.*

KS: The vacuum tube.

COMPUTE!: *Right.*

KS: Which you know you used to have them in radios, you had them in hi-fi's, and in gramaphones. It led to a whole massive industry of recording. What we, in Europe, call the "brown goods" industry — televisions, hi-fi, radios. And most of the people around me, in fact, would say it was better for that business. At least it was a very exciting business when it grew up around wartime, in fact. And I said 'Hey there is something happening here coming out of semiconductors and display technologies.' And you could see a few products. You could see digital watches and they were around $450 for the Pulsar, if you remember that.

COMPUTE!: *That would be back about what year would you say?*

KS: Oh, this has got to be back maybe 10 years — 8 or 9 years. I came to the conclusion there is a whole new generation of companies — a whole new industry coming. And I remember thinking — I felt that within two years the price of calculators could drop to under $25. I was wrong, it went to $10. And I thought I was being very bullish on that. And, actually, I decided I wanted to be a part of the industry. And after about four years there, I became the UK marketing manager for Bowmar where Commodore was my biggest competitor for two years. And it was as if Bowmar in the U.S. didn't have a very happy time. Unfortunately, the UK company was the brand leader dividing the UK when Bowmar went into Chapter Eleven over here ... I got approached to join Commodore, at that time and I joined Commodore to run their UK market operation. That would be seven years ago. Since that time (running through my background with Commodore) I ran the UK marketing operation. It involves the marketing companies in Europe, which is extensive. I'm then general manager of the systems division for the one which sent out the PET computer from the launching pad. I spent the last six months as European marketing

"...who's to say that in the next few years liquid crystals won't start replacing the vacuum, the cathode ray tube?"

director. I moved across here about four weeks ago now.

COMPUTE!: *You know, you said something that interested me about the fact that you saw the three waves coming through. The electric, which is what they call "white goods."*

KS: Yes.

COMPUTE!: *The valve, (which I'm going to have to put in parenthesis as the vacuum tube) and then the third one. You called that one "brown goods." So what would be this third category? Microprocessors?*

KS: I guess so. "Silicon goods" almost could be said.

COMPUTE! *Okay.*

KS: Comes out of chips and displays.

COMPUTE!: *Displays? What do you mean by that?*

KS: The liquid crystal displays, LEDs...

COMPUTE!: *Do you see those as being the central aspect?*

KS: Not central, but a definitely associated area. I mean, if you look at the total industry which includes things like calculators in the marketplace and TV games and, ah, who's to say that in the next few years liquid crystals won't start replacing the vacuum, the cathode ray tube?

COMPUTE!: *Yes, that was what I was thinking, maybe...*

KS: And ... oh I'm sorry ... it was like seven years ago, I remember, I drew up a list of companies I thought would be leaders in the business into three categories. The first category I thought would make it; those that did a good job; or those that were unlikely to. Commodore was in the last category at that time.

COMPUTE!: *Well, that's an irony.*

KS: I've had to eat my words.

COMPUTE!: *Do you have anything to which you attribute Commodore's success in particular?*

KS: I think there are several things. Commodore is

CBM/PET? SEE SKYLES ... CBM/PET?

"They laughed when I sat down at my PET and immediately programmed in machine language... just as easily as writing BASIC."

With the new Mikro, brought to you from England by Skyles Electric works, always searching the world for new products for PET/CBM owners. A 4K machine language assembler ROM that plugs into your main board. At just $80.00 for the Mikro chip, it does all the machine language work for you; all you have to do is start laying down the code.

The Mikro retains all the great screen editing features of the PET...even all the Toolkit commands. (If you own a Toolkit, of course.) Sit down and write your own machine language subroutine. The program you write is the source code you can save. And the machine language monitor saves the object code. The perfect machine language answer for most PET owners and for most applications. (Not as professional as the Skyles Macro-TeA...not as expensive, either.)

A great learning experience for those new to machine language programming but who want to master it easily. Twelve-page manual included but we also recommend the book, "6502 Assembler Language Programming," by Lance A. Leventhal at $17.00 direct from Skyles.

Skyles guarantees your satisfaction: if you are not absolutely happy with your new Mikro, return it to us within ten days for an immediate, full refund.

Skyles Mikro Machine language assembler.. **$80.00**

"6502 Assembler Language Programming" by Leventhal.............................. **17.00**

Shipping and Handling........................*(USA/Canada)* **$2.50** *(Europe/Asia)* **$10.00**

California residents must add 6%/6½% sales tax, as required.

Skyles Electric Works
231E South Whisman Road
Mountain View, California 94041
(415) 965-1735

Visa/Mastercard orders: call tollfree **(800) 227-9998** (except California). California orders: please call **(415) 965-1735.**

SEE SKYLES...CBM/PET? SEE SKYLES ... CBM/PET? SEE SKYLES ... CBM/PET? SEE SKYLES ... CBM/PET? SEE

WATCH

THIS

SPACE

CBM/PET? SEE SKYLES ... CBM/PET?

"Look how fast I create these great graphic displays on my PET with the new PicChip... it's like home movies."

PicChip, the new ROM that took Europe by storm, available only from Skyles Electric Works in the U.S. and Canada.

PicChip, a ROM extension of the BASIC version III, BASIC 4.0 or BASIC 8032 interpreter that offers over 40 commands that allow you to create programs with dynamic graphics displays: plots, bar graphs, pictures; and rolling, scrolling, shifting and inverting. All instantly and easily added to your BASIC program.

The address for the 2000/3000 (which would require PicChip module PC2), for the 4000 (PC4), and for the 8000(PC8) is $A000...unless you have a Mikro, WordPro III or IV, or Jinsam, which occupy that same address. In those cases, you will need the PicChip on an interface board that would reside in address B800...for the 2000/3000 series (PCB2), above the Toolkit. For the 4000 (PCA4) and 8000 (PCA8), the Mikro or WoodPro would be switchable manually using the Skyles Socket-2-ME.

Skyles guarantees your satisfaction: if you are not absolutely happy with your new PicChip return it to us within ten days for an immediate, full refund.

PicChip from Skyles Electric Works (Please indicate PC2, PC4, PC8)............... **$60.00**

Complete with interface board (Please indicate PCB2, PCA4, PCA8)................ **80.00**

Shipping and Handling........................*(USA/Canada)* **$2.50** *(Europe/Asia)* **$10.00**

California residents must add 6%/6½% sales tax, as required.

Skyles Electric Works
231E South Whisman Road
Mountain View, California 94041
(415) 965-1735

Visa/Mastercard orders: call tollfree **(800) 227-9998** (except California). California orders: please call **(415) 965-1735.**

SEE SKYLES...CBM/PET? SEE SKYLES ... CBM/PET? SEE SKYLES ... CBM/PET? SEE SKYLES ... CBM/PET? SEE

WATCH

THIS

SPACE

a very misunderstood company. It is looked upon in many people's eyes here as a U.S. company and, in many ways, as a newish company, where, in fact, we have actually been going 23 years. And the majority of our business turnover is outside the USA. We are, in that sense, sort of a diversifying company. We are, as you probably know, the brand leader in every country in Europe. And, in fact, in Canada here, which is right next door to the States (*pause*) ... I'm sorry, I'm afraid I was interrupted ...

COMPUTE!: *You were describing what would characterize Commodore's success ...*

KS: One thing that our products have always been known for is value for money. And I've known Commodore now, both as having worked for Commodore and being a competitor of it for a few years before, and always our products have been value for money. They've always been at the leading edge of what's happening. I heard somebody: 'Oh Commodore is lucky. It brought the PET out. It got into the market at the right time.' As you know, it wasn't luck. We've been doing that consistently for the ten years I've been at Commodore. Whether it was a scientific calculator or whether it was a computer and, you know, hopefully we're still there doing it today. Be it the VIC, or whatever.

COMPUTE!: *Is there any significant difference in terms of Commodore's advertising? Do they advertise extensively in Europe and England as opposed to what they do here?*

KS: I think possibly it had a more consistent business approach in Europe than here. This has been, relatively speaking now, an oddly successful market so far. Obviously, I'm hoping that will change.

COMPUTE!: *Well, do you attribute that difference to anything that you can put your finger on?*

KS: A combination of things. One of which was a decision by the corporation. It's saying: 'How much resources have you got in the company? Where do you put them?' If one goes back to the beginning of the microcomputer marketplace, I think that we were all learning, we were all short of resources. Whoever you name in the business. It's so tough getting production, getting finance, everything else going in the business three or four years ago, that some people just put all their eggs into one basket here. We, if you like, set about perhaps a different strategy. We set about continuing to broaden the product range and we put a fair bit of energy into that. If you look at our range now, we've got the VIC, which I think is a very interesting product down on the true personal end. We've sort of taken the original 8K PET, developed what would the personal man want — he'd want color, sound, low cost, and expandability. And that's the VIC. We've said 'What's the difference between them?'

[A buyer at the other end] wanted a desirable keyboard, he wanted more memory backup, he wanted software, etc. and that's really the 8000 series. So we developed up the VIC range, the 4000 series, the 8000 and, now coming out, the SuperPET series. So now I think we have put together ... we've put a lot of our investments into getting a broad market range. And we've put our investments into a world-wide market as well. We did, you know, invest into developing up Germany, England, France, etc., which probably meant that we spent less of our corporate resources — both human and others — in the U.S., compared to some of our competitors. What we did start, just about a year ago, was the beginning of our strategy to increase our presence here in the USA. The heart of that, very much, was the regionalization policy — start to get closer to the dealers, etc. and we've started. About a year now I think it has been going on up there. And we really haven't got that in place. We've got the products in place and we've got the financing in place. We're beginning to look at now the stepping up of our marketing activities here to capitalize on the other factors.

COMPUTE! *Well, I'm sure you're aware that Commodore's developed a reputation, in the short amount of time that it's been selling computers in America, of having, as you point out, a strong price-performance ratio, but a woefully weak support system. And there have been a series of announcements of new distribution centers, new dealer set-ups, new hot-lines. There's a long list of more or less reorganizational announcements. I take it there's going to be another announcement.*

My question would be: 'Why would the dealers and the loyal Commodore supporters be likely, this time, to expect that the kind of full-fledged support that they can get — let's say with the Apple — would be forthcoming from Commodore?'

KS: I'm not going to promise a mountain. I'd promise achieved improvement. And I'd think if you'd look at some of the things: the regions have shown a considerable amount of progress. Speaking to a lot of the dealers here, we seem to have shown a fair amount of progress after the last four months. Some things have worked, some haven't. That's par for the business. But I think, increasingly, we're continuing to move in one direction. I think that occasionally we have over-promised and that maybe, for the best intentions, that's something we've got to be careful of. I think we've demonstrated our fair bid over here the last four months. I think we will continue to demonstrate that by the various things we are doing.

My job here is to build up, and support any operations to build up, our marketing policies. I'm not going to tell you that it's going to transform overnight. I hope to continue and accelerate what we've been doing for, really, the past 12 months, I think. And, hopefully, continue doing some of the things that we've shown we can do, elsewhere in the world.

COMPUTE!: *I suppose that's the curious thing. That there's this strange disparity between — I mean even in a demographic sense — the Canadian and American markets can't be all that dissimilar, and yet there's this strange difference in them in the way the marketing took place. Compare, for example, the Commodore in-house magazines. The* Transactor *in Canada and the American magazine. There is an entirely different reality to those two things. I guess, as you say, there is really no single cause. But, all the way down the line — advertising, dealers, support — everything seems to be quite distinctly different as soon as you cross the border up there. I imagine you are more or less addressing yourself to trying to bring America in line with the level of success, anyway, that Commodore experienced overseas?*

KS: Oh, yes. I would be unrealistic to say that we are going to take the number one position in the US overnight.

COMPUTE!: *But that's your goal?*

KS: No, my goal is to increase our position in the marketplace here. I don't think that we're going to be number one in the next six months. That would be unrealistic. I think the danger is if we often said we were going to do something which is promising too much. I very much expect to take a better, larger share of the marketplace. How large that is, we'll have to see. We've got a good product range. I believe that actually we are in a relatively good position to start building. We *have* got a good product range. I think that some of our competitors may be under greater attack when this competition comes in. Our product range does have a very good price-performance out there so we do have a position in that sense. I think that we have a big user base out there going for us which anybody new coming to business these days doesn't have going for them. There are a lot of PET users. A lot of things like **COMPUTE!** magazine, which write about our product, lend to our support. I think turning ... using our strength out there is going to be important.

COMPUTE!: *Well, that leads me to a somewhat ... actually I would like to revert to something I was thinking I might want to ask you about one of the first things you said. You've gone to university in physics and you didn't like physics? Or didn't like your work...*

KS: I did get my honors degree, but I didn't want to spend my life being a physicist.

COMPUTE!: *Right. And you found the transition to marketing, specifically marketing products having to do with electronics, a relatively easy one?*

KS: Oh, I think so, yes ... my personality and that fits in that area. I've been in the marketing of electronic durables now for about 12 years.

COMPUTE!: *I guess one thing that the readers would be interested in knowing is anything you have in the way of a forecast for what the next year, five years, or ten years is going to be like, for computing, at any level? Can you do that in a sentence?*

KS: In a sentence?

COMPUTE!: *No, no. I'm just kidding.*

KS: (Laughs) It's going to go in many different directions. I think it is a fallacy to say we're in one marketplace. I think that we have to recognize that we are in a variety of marketplaces. That there are

"...my goal is to increase our position in the marketplace here. I don't think we're going to be number one in the next six months. That would be unrealistic."

the business marketplaces, the educational marketplaces, there are the personal marketplaces. In virtually every ... the instrumentation market places, the PET is very strong in that area as you probably know. There is undoubtedly going to be a large increase in the usage of computers. There will be a large increase in the awareness of them. I think different manufacturers will begin to find different niches in the marketplaces. I believe that it will require traditional industry goals, but I believe it will require other skills, as well, to be successful in business. I think, no question, that value for money will continue to increase in this marketplace. That's been the market ever since I've known it for ten years.

The price-performance ratio will continue to increase. I think the emphasis in some areas will begin to shift from technology to marketing. I think that we've almost gone through the very first era of the rush of enthusiasm of the products' technicalities. And people are now beginning to see them as tools to be used; and, therefore, the types of dealers, the types of users, the types of software packages in industry will begin to change. The people, in my experience, now are less asking how we did this, how we buy etc., as much as what it will do for me. And that we're going to see distribution broadening and the use broadening.

COMPUTE!: *Do you see, for example, that a computer will be in as many homes as, say, televisions, at some point? Or will, in fact,* become *a television?*

KS: I think, ultimately, the telephones, the televisions, all the technologies are now converging. That, yes, there will be some conversion certainly, but it will take a number of years. It will take...it could be five years before it happens. I think the television is becoming interactive with the TV games, with computers. I think that there is a gen-

PETTERM™

Terminal Program

Intelligent Communication

Uploads and Downloads

In Machine Language

Use your PET as an intelligent terminal. Access timeshare systems nd networks such as the Source or Telenet. This is an interrupt driven ystem with buffered input/output. Unlike dumb terminals, the PET an do other tasks instead of having to wait for incoming data. This exibility allows many advanced features.

These are complete assembled hardware and software packages. ll include line editing/resend, auto-repeat, shift lock, output to IEEE rinter and much more. Delivered on cassette or disk. Also, available r Commodore 8010 modem or Livermore Star 488 modem.

PETTERM I All features above .... **$ 80.00**
PETTERM II All features of I, plus local text editor with down-loading capability **90.00**
PETTERM III All features of II, plus 80 132 column scrolling window for viewing formatted outputs wider than 40 columns **100.00**

FORTH

A Programming Revolution

FORTH Interactive high level compiler and operating system 5–10 times faster than PET BASIC. High level block structured language. This is a true fig-FORTH implementation for 16 and 32K PETs.

fig-FORTH, with editor, assembler **$50.00**
with floating point and strings **$60.00**
with floating point, strings, macro-assembler .. **$70.00**
cassette and disk interface available.

DEALERS INQUIRE

F S S

software for small computers

1903 Rio Grande
Austin, Texas
78705

1-512-477-2207

P.O. Box 8403
Austin, Texas
78712

Source: TCE995

eration of people growing up computer literate. I would say that within five years that the business environment the computer will become...will be looked upon like a copy of a typewriter. It is just a business tool. It's not the hallowed sanctuary of the data processor department. It's due to come. And, in the home, it's going to become another electronic tool like the TV, like the HiFi. I think there are interesting areas that will develop through the years in education and in things like computer-age construction. And, potentially, it can be a very interesting area.

COMPUTE!: *Do you foresee any danger that what's happened to most other American industries — such as the automobile, television, and appliance industries — that the Japanese will eventually get into the computer industry?*

KS: I undoubtedly believe we're in international business; that you cannot compete on just a one country area. The Japanese, I have the greatest respect for in electronics. They may find this market harder to get into than the others because of the software. I also see this competition in Europe. I think the companies, if they truly look just at America, will find themselves under increasing pressure. I think that, as myself and as a company, it's one of the reasons why we deliberately have tried to act internationally in terms of distribution and development.

For instance, we have a whole host of software in British, German, French, Spanish. You name it. And I would say, yes, I would think the share of the purely American manufacturers decrease, but will remain the dominant force. And, if you want to choose to include the American manufacturer, it's probably a good question because we have factories in Japan and in Germany. So I mean, in our case, part of our production is here in the States.

COMPUTE!: *We've spoken mostly about hardware. I'd like to ask you just a couple of questions about software. Do you feel that — well, this is actually a question about hardware — do you feel that the peripherals are mature? By that I mean, do you feel that the disk drive is essentially always going to be the disk drive it is today?*

KS: No, if you want one statement about the future from me, there's going to be even more change in the future than there has been in the past. It is, by no means, a static industry in terms of development.

COMPUTE!: *So you feel that there will be a product that will more or less replace the disk drive?*

KS: I believe, in time...I don't believe that anything will be obsolete overnight. But I believe, in time, we will continue to have changes. The greatest technical developments are still going on out there. Our biggest problem is keeping up with the marketing front.

COMPUTE!: *How do you feel in general about the future of software?*

KS: I think it will increasingly become more user-friendly. That's where the biggest breakthrough has got to come. I see the most important areas are use and assimilation by the person who is going to handle it. The days of the spoke software are gone in this industry. You'll have not just pre-packaged software — you have a lot of that. And also the quality has to continue to increase. But you have tools to develop the software much quicker and, increasingly, ones that can be done by the user. Products probably like OZZ, the Manager, things like that, are probably sort of the halfway stage between a purely pre-packaged product and a spoke product.

COMPUTE!: *One last question (unless you have something you think of that I haven't asked). What are the plans in the other direction, on the other side of things from the VIC? What are the plans for the SuperPET?*

KS: I'd say it's a very good market here. I think it's next in addition to our product range. I think it's going to be particularly suitable for areas in education where we're already extremely strong, higher education. But I think it's going to increase when they move us into some of the more traditional areas of business. I want to say that, so far, the micro has not gone into the traditional data processing. Small businessmen, inside the big company, the department manager who couldn't get time on the mainframe put in a micro and did word processing or put up financial planning or kept his own mailing list in his own office. I think that the SuperPET potentially bridges his gap.

I think the traditional computer industry has recognized the micro at last. Maybe IBM's entry signifies that. If you can't beat them, join them. I'm beginning to see rather traditional computer companies not having such a good time as they were. Rather traditional software houses were beginning to find it tougher. When the micro first came out, they could be poo-pooed a little bit. People said where there was no software, there was no service. All those were things that would come in time, but the potential for the machine was there. Now it's there.

I think the SuperPET is going to start appealing to some of the software audience out there — the ability to write in languages they know. That the DP department is beginning to look at micros. They are beginning to get involved with the specification of micros within their company at the department level and that they will. For the SuperPET offers them something very attractive: the ability to potentially transport software, the ability to use it potentially as an intelligent terminal. I think it begins to cross that gap between the traditional computer and the micro. And that could be a very important thing to us. ©

This is the second **COMPUTE!** *Overview. Believing that* **COMPUTE!** *readers deserve the most objective, comprehensive analysis possible, we select new products which are then sent out for Overview. We have established review panels, each consisting of three or more reviewers whose backgrounds qualify them to analyze new software or hardware products.*

To better prepare **COMPUTE!**'*s readers for their buying decisions, and to present fair reviews, we collect the independent opinions of the panelists into one Overview. Lawyers will test legal packages in the environment of a legal office, doctors will test medical systems, and this issue's Overview panel is made up of computing professionals who have each used Commodore computers since they were first introduced. We think that you will find* **COMPUTE!** *Overviews complete, balanced, and informative. — The Editors.*

COMPUTE! Overview: Power

It should be acknowledged at the start: the review panel was unanimous in concluding that POWER is an extraordinary product. It is a ROM chip which goes into the machine and adds several significant commands to the BASIC chips already inside. The key word here is *significant*. One panelist said that POWER made the PET "into an entirely new, much stronger computer." It is not inexpensive ($89.95), but the reviewers agreed that it was an excellent value. One commented, "I think that adding a disk drive or a printer are important expansions to a computer; adding POWER has some of that impact ... it greatly expands the programmer's capabilities."

POWER Commands

POWER was written by Brad Templeton; the manual, by Jim Butterfield. The documentation is written by a knowledgeable user, based on extensive experience (rather than written by a programmer as an afterthought). Butterfield uses his considerable knowledge of the PET system and POWER to make the manual clear and complete.

A major advantage which POWER possesses over a number of the chips available is that it performs in almost precisely the same way on 8032, 4032, and 3032 machines.

POWER consists of four thousand bytes of machine code which provide a great amount of help to the BASIC programmer. It solves many housekeeping problems. It also provides solutions to problems you didn't even know you had. Theoretically, POWER adds at least thirteen commands to PET's BASIC. The exact number is irrelevant in any comparison shopping, because several instructions permit adding your own commands, thus expanding the count well beyond the specified thirteen.

Some of the standard commands, also seen in similar products, are automatic line numbering, deleting all or portions of the BASIC text, and renumbering. The unique feature of this renumber command is that it can renumber not only the entire text, but also *any portion* of a program. Subsequent adding of extra program lines becomes a snap.

This is even more useful when employed in conjunction with POWER's disk or tape MERGE command. The latter is an appending routine which permits you to attach another program or a subroutine to a program already in memory. Once again, a powerful feature has been added: for complicated uses, one may, with little effort, fit subroutines into *any* place within an existing program. A machine code "dynamic keyboard" type of routine performs the job by reading an ASCII format subroutine created earlier. Program lines being appended from a tape or disk are included into the program in the same fashion as you would have typed them in from the keyboard.

Extraordinary Features

Power has "instant phrases." You may tell POWER that when you press shifted-A, for instance, what you really mean is this print loop:

```
FOR J = 1 TO 50: PRINT X(J): NEXT
```

It works like magic. Press shifted-A, the above line appears on the screen, you hit RETURN (after altering the line first, if you wish) and the values of the X array are displayed. Should you wish to use shifted-A for something else, you may redefine the key on the spot.

The unique, and very clever feature of this is that special REMs are used, at the start of the program, so that a BASIC program will continue to run even if POWER is not in place, and yet the phrases are available if required.

Both Instant Phrases and Instant Subroutines represent tremendous advances in capability. It would be ideal if Jim and Brad could be persuaded to produce some form of Advanced Users Supplement, or to update to the manual, dealing with this and the similar question of the more specialized aspects of using the XEC command to allow the disk unit, or some other peripheral to take over the machine in substitution for the keyboard. It is tantalizing to see simple examples. Why not some extremely clever ones too?

for the keyboard. It is tantalizing to see simple examples. Why not some extremely clever ones too?

This instant phrase feature of the keyboard is invaluable. Imagine never having to retype the monitor SAVE instruction. In this instance, assuming you are working within BASIC, you can define a key to mean S"prog",01,1234,5678. Since these single keystrokes stand for more than one keyword it could prove the most useful of the instant features.

On a smaller scale, most shifted keys have been defined to mean BASIC keywords, like OPEN, DATA, READ etc. (NEW has no key, a nice touch). You may change those default assignments, if you wish. Labels are provided for placing on the keys to help you remember which key means what. When you press a shifted-I, for example, the word INPUT appears on the screen. Shifted-L followed by RETURN lists the program.

The debugging potential of the POWER chip is enhanced by virtue of the fact that POWER features can be embodied in instant phrases. This has enabled the major design criterion which Brad Templeton had in mind, namely that program should be transportable and should not be POWER-dependent, to be implemented, whilst allowing the programmer full use of this attractive feature.

A major advantage which is available in the 80 column machine only is the command SEL I which gives the ability for an Instant Phrase to be accepted in response to an INPUT command. This is really great for debugging, because it implies, provided you hit the right key, that the data you input is identical every time.

And on a very large scale, keys can be defined to mean *whole sections* of a program. For example, Upgrade ROM may define a key to mean the well known disk error channel read routine. You may define a key to mean "execute the routine which begins at line 1000." You can convert numbers with it, draw with it, whatever you wish.

This feature is difficult to appreciate fully until you've used it, but it permits much flexibility. One of the panelists tested it by writing a "programming shell" which was LOADed into PET at the start of any session. This consisted of eleven lines (1-11) which held the REM statements which POWER uses to define keys for this feature. The subroutines themselves were placed in lines 50000+.

This easily added the following *single-keystroke* functions to BASIC. Clear screen and display the time for 10 seconds. Get the name and SAVE a program to disk. Get the name and LOAD a program from disk. Enable POWER. VERIFY a SAVE. Display Directory. List a program. Display instant-feature command keys. Delete lines 1-11 and 50000+ (removing the shell from whatever program was written). Search the program for a particular phrase, word, or variable. Translate between hex and decimal.

The panelist observed, "I wrote (as one of these instant subroutines) a short 'dynamic keyboard' routine to permit POWER's own commands to be used within subroutines. This allowed specified line deleting, searching, etc. Whatever you often need to do in BASIC programming you can assign to a shifted key and it's instantaneous. Machine language programmers can slip out of the monitor, press 'H' and enter a decimal number — the correct hex is immediately printed on the screen. Anything you want can be accomplished at a keystroke. It's like having a library of all your utility subroutines built into the computer. You can effortlessly add to BASIC; you can *personalize* your machine in a way you never imagined."

The "instant-key" feature, if not needed or if it interferes with something else you want to do, can be easily turned off.

XEC-Execute

Possibly the most extraordinary POWER command is XEC. One is reminded of COM files on a DEC computer. Sharp programmers may think of clever things to do with XEC that may not have yet occurred to the author. Merge is part of the XEC subset. XEC is a command that can turn PET into a powerful workhorse with exotic things like linking loaders. It is possible to set up an elegant system of program development aids or system execution requests consisting of a series of BASIC commands. These commands are placed on a tape or disk file and recalled by XEC. As the file is read, the statements are placed on the screen and executed on the spot, in the same fashion as if they were typed from the keyboard. This means, for instance, that if there is a repetitive sequence of events that you usually go through each day, you no longer need to type the commands. After buying POWER you can decide to set up such a system by typing the commands just once more and never again.

XEC (execute) is very powerful and totally unique. Because it is an extension of the technique pioneered by Brad Templeton for merging programs on cassette, there is a danger that it will be seen as merely a rather complicated merging method. This is totally wrong. The sequential file created is read by the computer and treated in every way as if the strings were being typed in at the machine. This implies that they will be acted upon as Direct Commands, and their capability is limited only by the fact that the special commands in POWER cannot be included. The manual gives a very tantalizing hint of writing programs which write programs.

POWER permits you to redefine the keyboard in almost any way you wish. For the technical and machine code people there are additional commands that can help, namely MLM and FIX. MLM simply makes a *call* to the monitor as opposed to a *break* obtained by SYS1024. This saves typing effort, crashing due to keybounce or bad typing, and in BASIC 4.0, guarantees that the CMD command remains enabled. FIX is a handy command that aligns BASIC pointers following a BASIC LOAD of machine code. Such LOADs often make a mess of the end of program pointer. You no longer have to worry about it. FIX finds and notes the actual end of the BASIC program. (FIX will not restore pointers after a NEW.)

POWER also provides find and replace functions. A program text can be searched for presence of a keyword, a character string, or a variable. It can be searched for occurrence of patterns or sequences of commands; for instance, a FOR-NEXT sequence. It can be searched for all occurrences of words containing a pattern without necessarily knowing what the rest of a word might be. POWER can replace a variable by a differently named variable and can change a character string to a different character string.

The ability to continue to SEARCH for a particular string, without having to define it again, merely by hitting @ is very convenient.

POWER includes a tracing command which permits you to watch the program executing at a slow pace. It is the best visual debugging show on the PET. POWER can display any or all of: line number, its statements, and the value of a last relevant variable. Three different speeds are permitted, including single step.

The versatility of the Trace command is enhanced by having special SYS commands with parameters enabling you to switch in and out specific types of Traces. This is particularly good in avoiding waste of paper when loops are running, and also makes it easy to bypass sections of the program which have already been successfully debugged. Complex programs can be debugged in stages using a GOSUB to a subroutine for the SYS commands switching the Trace on and off.

The manual shows how to avoid the problems of tracing GET commands. FIX will occasionally be very useful for resetting certain essential pointers which may have been clobbered by loading machine code programs into an unusual part of memory. The manual gives a very full explanation of how this command will prevent loss of variables or of the BASIC program.

A program stopped by a STOP key or by an error can be queried as to the reason, or more precisely, the location of the last executed place by use of the WHY command.

The variable dump command displays values of variables and functions and permits values to be changed on the screen, once again, an unusual feature. Array values are not displayed. Taken together, TRACE and WHY will also make BASIC easier to learn. There is no ambiguity any more about what is wrong, where it is wrong, or how it goes wrong during a RUN.

The chip adds more punch to the already excellent PET screen editor. Cursor keys and INST/DEL keys repeat at a very fast rate. Up and down cursor keys can also be used for scrolling the program. Yes, scrolling the program, WordPro fashion, in both directions. LIST need no longer be used. The scrolling motion is as smooth as PET usually provides in the up direction and perhaps only slightly choppy in the down direction on 40 column PETs, though this may be an illusion due to double BASIC lines on the 40 column screen, coming in backwards on top.

A useful feature of the repeating cursor key and instant key provisions (IRQ driven) is that they disengage when POWER takes notice of LOAD, SAVE and RUN commands. Consequently, the program runs fast and it is no longer necessary to resort to the usual RUN/STOP trick of restoring the IRQ vector prior to input and output. It is easy to re-enable the repeating key feature. Pressing RETURN or issuing any POWER command does the job. The entire chip can be disabled by OFF.

There are two items that might cause trouble for some people. First, a "delete all program" command exists and is written "DEL-". It is equivalent to NEW. Users of TOOLKIT™ might wipe out a program unintentionally since they are accustomed to putting in the line number or getting an error message if the number is missing. ("DEL" does give error message). Second, the Universal Wedge and the old DOS SUPPORT program conflict in the hookup to the CHRGET routine and are, therefore, disabled. Users of Basic 4.0 systems will not feel the loss of the Wedge program. Users of the Upgrade systems without convenient shorthand disk commands might miss it. However, the tradeoff seems obvious: POWER gives far more than the Wedge and, with POWER, you can *add* single-stroke disk commands to replace the Wedge anyway.

The Book

The documentation provided with POWER ought to be considered the standard against which future documentation is measured. It is correct, unambiguous, redundant, thorough, coherent, and fun to read. The redundancy greatly assists understanding. Most of the time, Butterfield shows a command and explains it in one or two sentences. Then, in case you didn't get it on the first pass, he rephrases the explanation to make it absolutely clear. Abundant examples follow each command. Where appropriate, he points out and explains syntax which is different from that used in similar products. Just about every possible, usual, or unusual use of a command, together with its pitfalls, if any, is described. A separate chapter is provided for a list of several unexpected events that might occur. One gets a feel for how the program functions by reading that list and by reading Jim's explanations and methods of preventing trouble and misuse.

Apart from several innocent typos, no errors were discovered. The book contains a table of contents, index, and top-of-the-page indicators of contents on that page, making it easy to find things. It is 74 pages long.

Reading the book, one not only quickly learns how to use POWER, but also gets to see it from the perspective of the whole PET. In the case of the XEC command, a hint of a warning is given to the effect that PET just might make us all obsolete by writing its own programs. Short programs are also provided. For instance, there exist two disk-related routines for users of the Upgrade ROM system, which can be attached to the instant action keyboard. Various instant keyboard illustrations are amusing and educational.

In an unusual twist in documentation practice, the book also includes a listing of the key locations in the chip, making it possible for machine code people to interface to POWER. A user may also add his own commands to expand POWER's power. It should be possible, for instance, to reinstate the WEDGE program back into existence, even though the interfacing addresses conflict.

Command Set Summary

AUTO (line #) (increment) — automatic line numbers as a program is being entered.

DEL (line range) — delete any range of lines (syntax similar to LIST).

DUM — provides a list of all program variables which can then be, if necessary, changed on the screen.

FIX — restores BASIC pointers.

MLM — a call entry to the monitor.

OFF — kills POWER.

REN (increment) (new starting number) (line range) renumbers all or a range of program line numbers.

SEL (R) or (K) or (P) — turns on or off a number of the "instant key" features.

TRC (L) or (N) or (T) or (NT) or (LT) — enters the various Tracing modes.

WHY — prints the line in the program where an error stopped the program (or the STOP key was pressed). It turns on reverse field over the error in the line.

XEC (file number) — turns control of the computer to a tape or disk file.

@(line range) — search.

] (replacement) (line range) — search and replace.

General Overview

•Panelist #1: "By way of summary: POWER is a unique program giving you many useful commands. It is easy to learn to use. It is easy to expand and customize. It is meant to make programming easy. It is 'friendly' in use. It works smoothly and without a glitch.

"Needless to say, Brad Templeton should be congratulated for an extremely difficult job done imaginatively and Jim Butterfield gets a little gold star for writing the most unique set of instructions. More power to both."

•Panelist #2: "The user will find this package very helpful when writing and debugging BASIC programs. Some of the commands are not as useful as others and some will take a little getting used to. On balance, however, this is an excellent value for the price."

•Panelist #3: "With the possible exception of my $400 word processor, I can think of no software product which matches this one: I will use it often every time I'm at the computer. I tried to balance my comments by looking for things to complain about. I can't find any.

"Usually I'm a curmudgeon about this sort of "helper" software. I never used the wedge because I found it more trouble to load than the time it saved. I avoided other CHRGET "aids" because they, too, seemed rarely worth using. POWER is the exception. It contains such a range of *necessary* programming tools (and permits the user to fashion his own tools so easily) that I would compare it to BASIC's PRINT command: after you use it you can't imagine programming without it.

"One final comment. In the large manual, Butterfield does for POWER what he's been doing for the PET in **COMPUTE!** for years. He lucidly teaches you the essentials and then takes you beyond, into new tricks and methods you'd likely never have thought of yourself. This whole package is, without qualification, a splendid addition to the computer."

•Panelist #4: "Power offers so many facilities, that the user, whether beginner or expert, can grow in capability by using it.

It is worth emphasizing the fact that this is no mere run-time improvement to extend BASIC, nor a mixture of such and programmer's aids, but rather a very carefully thought-out set of aids to the serious user of BASIC, designed by a truly expert programmer, to cram as much power as possible into a 4K chip, with as much care taken in choosing what to leave out, as what to include."

POWER. Professional Software Inc., 166 Crescent Road, Needham, MA 02194. $89.95. Versions available for the 2001 (graphics or business), 4000, or 8000 CBM computers. ©

The PET Speaks

Kenneth Finn
Bedford, NY

The following article gives the Commodore PET computer (all ROMs) the ability to reproduce speech from audio cassette tapes. Even though the sound quality is very limited, the principle will allow intelligible speech and the program is by no means optimized — much can be done with it.

This program was taken originally from an idea of Robert Bishop in *Microcomputing*. He wrote about the idea for the APPLE computer. His original program was 18 bytes long. In order to adapt the principle for the PET many changes had to be made. This is evidenced by the present size of 81 bytes. Another problem was that the PET has no built-in speaker like the APPLE so I had to substitute the "CB2 sound" convention which many present PET users have.

For those of you who have not heard about it, purchase a small audio amplifier such as the Radio Shack #277-1088 two hundred milliwatt solid state speaker-amplifier. This is then wired to pins N and M of the user port via a card edge connector. These pins are the CB2 line and ground. This setup can make musical sounds also using the internal shift register of the 6522 VIA. This article will not explain this techniques because it has been often described. This simple setup which has many uses.

Basically, the small program shown below does the following things:

1. It sets the interrupt to stop further interrupts.

2. It starts the motor on cassette #1.

3. It tests to see if the stop key is pressed.

4. It clears the 6520 Keyboard PIA and sets CA1 for a low-high transition.

5. It then waits for that transition. This means that the cassette has sensed some sound. This also represents a digitization of the analog signal from a voice tape. Also note here that the PET cassette electronics has a Schmidt trigger in it which also stops analog signals cold.

6. When it detects the shift, it sets CB2 on the 6522 high. Note that this is tricky since other bits in the 6522 PCR must be left alone.

7. It then sets up CA1 for a high-low transition and sets CB2 low when it occurs. It then returns to step #3 for another cycle.

8. Lastly, when the STOP key is pressed, it turns off the motor and returns via a break instruction.

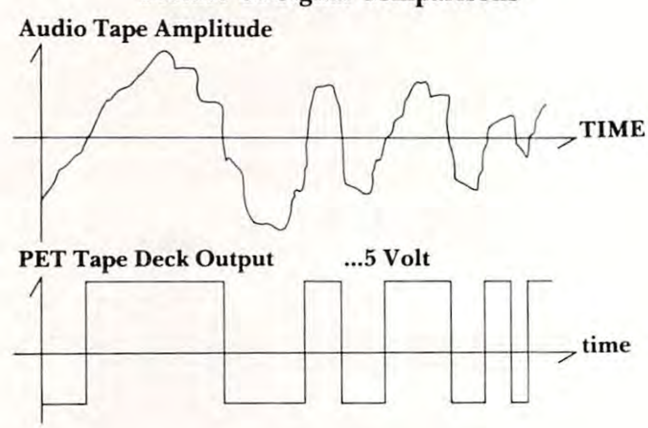
To use the program, first have it in the second cassette buffer. To enter the program, type SYS1024. Then type .M 033A 0390 (or, for 4.0, M 0360 03B0). Then enter the numbers right over the numbers on the screen, hitting RETURN after each line is changed. Then issue the monitor GO command (i.e. .G 033A) and you will hear some semblance of sound through the CB2 amplifier. The STOP key on the PET will get you out of the routine.

What is next for this routine? Well, speeding up the main loop will improve the sampling rate and, thus, the sound quality. Also, putting a simple low-pass filter between the PET and the amplifier will eliminate the switching noise. Other improvements such as adding a way for a BASIC program to time a message and then stop the routine would make it quite useful.

A big question is what to do with it? The answers are many. First it can be used to provide audio directions after a digital program load. Second it can become a speak and spell type of demonstrator. Third it can provide speech processing once limitations in the sampling rate/memory requirement are solved. (Hal Chamberlin where are you?)

I have just begun experimenting with this routine and I am looking forward to discovering both new uses and new modifications to it. The routine also gives many clues on how to use the cassette IO and the CB2 line which are useful all by themself. Happy Experimenting.

Picture Of Signal Comparisons



Program 1: Original and Upgrade PETs

```
.:  Ø33A 78 A9 35 8D 13 E8 A9 F9
.:  Ø342 8D 1Ø E8 A9 1Ø 2D 12 E8
.:  Ø34A DØ Ø3 4C 84 Ø3 AD 1Ø E8
.:  Ø352 A9 3E 8D 11 E8 2C 11 E8
.:  Ø35A 1Ø FB AD 1Ø E8 AD 4C E8
.:  Ø362 Ø9 EØ 8D 4C E8 AD 1Ø E8
.:  Ø36A A9 3C 8D 11 E8 2C 11 E8
.:  Ø372 1Ø FB AD 1Ø E8 AD 4C E8
.:  Ø37A 29 1F Ø9 CØ 8D 4C E8 4C
.:  Ø382 4Ø Ø3 A9 3D 8D 13 E8 58
.:  Ø38A ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ
```

OLD PET OWNERS NEVER DIE

☆ ....THEY JUST CALL ON US! ☆

Our innovative line of hardware products protect the investment in your original version 4K/8K PET by extending its life and usefulness. Our software works on most PET/CBMs.

• 2114-TO-6550 RAM ADAPTER HUNDREDS SOLD!

Replaces up to eight 6550 RAMs with low cost, reliable 2114s—one at a time. Use two units for total 6550 replacement. Never worry about RAM availability again!

PHB-001	Bare Board	$11.95
PHK-001S	Full Kit (No 2114s)	$17.95
PH-001S	Assembled (No 2114s)	$24.95

($2.50 postage/handling per order)

• 4K MEMORY EXPANSION

Installs easily internally. Uses eight 2114Ls. Write protect provisions. Second expansion needs external +5V power. "Soft-ROM" applications. Bigger programs!

PHB-002	Bare Board	$16.00
PHK-002	Full Kit (No 2114s)	$29.00
PH-002S	Assembled (No 2114s)	$42.00
PH-002	Assembled with 2114s	$89.00

($3.00 postage/handling per order)

• "REAL WORLD" SOFTWARE

For most PET/CBMs. Word Processor, Mailing List, Catalog, Satellite Tracker, Morse Code Keyer, Frequency Counter. Professional results at low cost!

$17—$25

Write for Free Catalog

PROMPT SHIPMENT! California Residents, add 6% Tax

SATISFACTION GUARANTEED Foreign Shipping Higher

OPTIMIZED DATA SYSTEMS

Dept. C, P.O. Box 595 – Placentia, California 92670

DEALER INQUIRIES INVITED

PET/CBM Trademark of Commodore Business Machines

Let Us...

flick your VIC

Vic 20 memory expansion — you get 8k and 176x176 graphics for only $59.95

12VDC Power Converter — use the Vic in your car, camper, etc. $49.95

Brass Pounder — learn morse code the new and exciting way! $5.95

Music Machine — play songs from sheet music using the Vic's sound. $5.95

Vic Monitor — lets you do machine code programming on the Vic! $6.95

Talking Vic — lets the Vic speak the numbers one to ten! $8.95

Joysticks ....$24.95, **Schematics** $7.95

* * * * * * * * * * * * * * * * * * * *

OSI Rip Off 4 $5.95, Battle Zone $5.95, Adventure Game $9.00

TRS-80 Robot Simulator program $7.95

Add $2.50 per order to cover shipping and handling.

micro systems by
PETTED
hardware with a soft touch

P.O. Box 21851,
Milwaukee, Wisconsin 53221

A Wooden Computer?

Not from Commodore!

So why should the desk look like wood? A pleasant cream and charcoal trimmed desk looks so much better with Commodore systems. One look and you'll see. Interlink desks are right. By design.

The specifications only confirm the obvious:

- •Cream and charcoal color beautifully matches the Commodore hardware and blends with your decor.
- •An ideal 710 mm (28") keyboard height yet no bumping knees because a clever cutout recesses the computer into the desk-top.
- •High pressure laminate on both sides of a solid core for lasting beauty and strength.
- •Electrostatically applied baked enamel finish on welded steel legs—no cheap lacquer job here.
- •T-molding and rounded corners make a handsome finish on a durable edge that won't chip.
- •Knocked down for safe, inexpensive shipment.
- •Patented slip joints for quick easy assembly.
- •Leveling glides for uneven floors.
- •Room enough for a Commodore printer on the desk, yet fits into nearly any den or office niche— H: 660 mm (26") W: 1170 mm (46") D: 660 mm (26").
- •Matching printer stand available with slot for bottom feeding.

Price: $299

In short, as Commodore dealers, we won't settle for anything that looks good only in the catalog! Our customers won't let us. They don't buy pictures. And neither should you. This is why **we will let you use one of our desks for a week** and then decide. If for any reason you don't like it, just return it in good condition for a cheerful refund.

If your Commodore dealer doesn't carry our desks yet, send a check for $299 and we will ship your desk freight paid!

Name____________________

Address__________________

City__________ **St**____ **Zip**____

Interlink, Inc., Box 134, Berrien Springs, MI 49103
Master Charge and Visa welcome. Call our order line:
616-473-3103

Program 2: 4.0 PETs

```
.
.:  Ø36Ø 78 A9 35 8D 13 E8 A9 F9
.:  Ø368 4D 1Ø E8 A9 1Ø 2D 12 E8
.:  Ø37Ø DØ Ø3 4C AA Ø3 AD 1Ø E8
.:  Ø378 A9 3E 8D 11 E8 2C 11 E8
.:  Ø38Ø 1Ø FB AD 1Ø E8 AD 4C E8
.:  Ø388 Ø9 EØ 8D 4C E8 AD 1Ø E8
.:  Ø39Ø A9 3C 8D 11 E8 2C 11 E8
.:  Ø398 1Ø FB AD 1Ø E8 AD 4C E8
.:  Ø3AØ 29 1F Ø9 CØ 8D 4C E8 4C
.:  Ø3A8 66 Ø3 A9 3D 8D 13 E8 58
.:  Ø3BØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ
.
```

Program 3: Source Code

033A	78			SEI		Set interrupt
933B	A9	35		LDA	#$35	
033D	8D	13	E8	STA	$ E813	Start motor for cassette #1
0340	A9	F9		LDA	#$F9	
0342	8D	10	E8	STA	$ E810	Test for STOP key pressed
0345	A9	10		LDA	#$10	
0347	2D	12	E8	AND	$ E812	
034A	D0	03		BNE	$034F	Branch if no key pressed
034C	4C	84	03	JMP	$ 0384	Go to End of program if key pressed
034F	AD	10	E8	LDA	$ E810	Clear 6520
0352	A9	3E		LDA	#$3E	
0354	8D	11	E8	STA	$ E811	Set CA1 for Low/High transition
0357	2C	11	E8	BIT	$ 036F	
035A	10	FB		BPL	$ 0357	Wait for transition
035C	AD	10	E8	LDA	$ E810	Clear 6520
035F	AD	4C	E8	LDA	$ E84C	
0362	09	E0		ORA	#$E8	Set CB2 HIGH
0364	8D	4C	E8	STA	$ E84C	
0367	AD	10	E8	LDA	$ E810	Clear 6520
036A	A9	3C		LDA	#$3C	
036C	8D	11	E8	STA	$ E811	Set CA1 for High/Low transition
036F	2C	11	E8	BIT	$ E811	
0372	10	FB		BPL	$ 036F	Wait for transition
0374	AD	10	E8	LDA	$ E810	Clear 6520
0377	AD	4C	E8	LDA	$ E84C	
037A	29	1F		AND	#$1F	Set CB2 LOW
037C	09	C0		ORA	#$C0	
037E	8D	4C	E8	STA	$ E84C	
0381	4C	40	03	JMP	$ 0340	Jump always for next transitions
0384	A9	3D		LDA	#$3D	
0386	8D	13	E8	STA	$ E813	Stop motor on Cassette #1
0389	58			CLI		Clear interrupt
038A	00			BRK		Return to monitor via break ©

Machine Language: Monitoring Progress

Jim Butterfield
Toronto, Canada

A Monitor is a simple thing. It's the built-in program that allows you to read and change memory. Yet beginners find that developing skills in using their monitor systems can be a major hangup in learning how to cope with machine language. It's partly mechanics, and partly insight.

First Functions

All monitors allow memory to be displayed and changed. The display is usually in hexadecimal, and beginners are urged to learn about hexadecimal numbering systems. The command varies from system to system. On the KIM, you press the AD button and enter the address; memory contents are immediately displayed, and the user can "walk through" memory using the "+" key to step to the next location. The SYM asks you to enter the address first and then press the MEM key to see the contents. The AIM displays memory locations four at a time, and screen-oriented systems usually allow any number of locations to be seen; in the PET it's multiples of eight which means that if you ask for ten addresses to be displayed you'll get sixteen.

There's always a command to change memory, although sometimes it doesn't look like one. On the KIM, it's a mode change: you switch to data entry mode by pressing the DA button and, from that point on, you're entering changes. On the PET, memory is changed by first displaying the contents on the screen and then moving the cursor back and typing over; it's similar to screen editing in BASIC. Remember to press RETURN to make the screen change permanent. Most other machines ask for a specific change-memory command to be typed — on the Apple, it's a colon, for example.

So we can read memory and change memory: that's indispensable if we want to get a program into the machine. The next question is how to run the program, and how to stop it after it's run.

STOP and GO — Traffic Control

The usual command to get a program to run is GO, often abbreviated by the letter G. GO does a number of fascinating things before it actually goes to the selected program; to explain them, we'll need to reverse the order and discuss STOP first.

When a program is Stopped, there's a lot of useful information inside the microprocessor chip. It's not in memory; it's in the chip where you can't get at it, and it will soon be destroyed by other data. It's vital to be able to inspect this data which is held in "registers" within the chip; so the monitor's STOP sequence arranges to dump these values to a fixed place in memory. Now you can look at them if you wish by inspecting memory.

Suppose we have Stopped a program and we want it to continue: that often happens during program testing. We issue the command for GO — but our program wouldn't run properly if the information in the registers were lost ... and the monitor itself has been using and changing those registers as it worked. What to do? The GO command, just before going to the desired location, will bring back the register values that were dumped into memory and restore the microprocessor exactly. Then it goes to your program. Clever? No — vital.

How do we Stop a program? There are several ways. The Break instruction in machine language (BRK — code 00) will almost invariably take you to the Monitor stop program. Some machines are equipped with a STOP key which interrupts the program. Advanced monitors allow the definition of Breakpoints — locations where the program will flip to the Monitor.

Registers

The Stop routine causes the contents of the microprocessor's registers to be dumped to memory. You may examine them there, and you also may change them. Don't become confused: these locations are not the registers themselves; they are just copies made at Stop time.

Many monitors allow registers to be inspected by means of a special command — R is popular. Once again, there is always a means of changing the register image; this is useful when you are checking out a program and want to make shortcuts or special tests.

There aren't a lot of registers in the 6502. There's the Program Counter, usually abbreviated PC, which tells you where the program was operating when it stopped. There are three data registers, A, X, and Y which are the only places in the microprocessor where you hold data. There's the Stack Pointer, which tells you where the temporary stack values are being kept: for example, if SP contains value EC, you should look in locations 01ED to 01FF for the values themselves. Finally, there's the Program Status Word, often abbreviated PSW or SR for Status Register. This last contains several flags; it's often useful to be able to break the hexadecimal value to binary so as to see individual flags. For example, if I see a value of B9 I know immediately that the Carry flag is set. Can you see how? To work it out, you'll need to know that the code for the flags is NVXBDIZC. Don't try to pronounce it — use it to decode the bits.

The PET monitor has an oddity: among the registers it shows a value called IRQ. This isn't a register at all; it's a vector located in memory (at hex 90 and 91, if you must know).

Saving And Loading

On many systems, the only means of saving memory to tape or disk and loading it back to memory are provided by the Monitor. When Saving memory you must give the start and end addresses of memory. Know your system: many Monitors require that the end address be one beyond the last location saved. When loading, the Monitor usually knows where the data came from and puts it back; some Monitors allow loading to a different location.

Some Monitors don't write a program name to tape; on others you must supply the name. When a Monitor is capable of saving to more than one device (e.g., either of two tapes or disk) you'll have to furnish information on where you want your data saved.

Quitting

Many systems are Monitor-oriented. They started as tiny systems with no language but the Monitor and, when extra features were added, the Monitor still was dominant. In this case, you must leave the Monitor with a GO command to enter another system. To return to BASIC on an Apple II system, for example, you're asked to type in the curious command 3D0G. This has nothing to do with man's best friend, but is a command to GO to location hexadecimal 3D0 where BASIC takes over.

The PET considers BASIC as standard and the Monitor as an "extra;" the Monitor returns to BASIC with the simple command: X (exit). Other systems make provision in the Monitor for invoking other systems — the AIM, for example, has special control keys for its text editor, its assembler, and BASIC.

Extras

Most monitors can be extended; some have extra features built in. Some of the features you may find in your system, or may wish to add with an "extension" package or to write yourself include: Breakpoints, single-stepping, slow step, disassemblers, tiny assemblers, memory search, memory fill, tape check, memory block move.

It's quite a luxury to have these extra features. Keep in mind that they're icing on the cake ... you can work well, if not quite as comfortably, with just a basic Monitor. And remember that extra features take up extra memory. It doesn't help to have an 18K super Monitor when you have an 8K machine... ©

Directory For 3.0

Stephen Meirowsky
Peabody, KN

A very important part of any disk system is the ability to get the directory of your disks at any time. Commodore has provided a very nice DIRECTORY command for 4.0 BASIC. But what if you don't have 4.0 BASIC? On the Upgrade BASIC machines you must either use LOAD"$0",8, which would destroy your current BASIC program, or use the DOS Wedge from Commodore which can only be used in direct mode. To solve the problem of getting a directory during a program RUN, one must use some other method.

This program allows you to display the directory at any time you want in your program. This comes in handy when loading or saving files from a disk. The ideas for this program came from the article "PET/CBM Disk Formats" **COMPUTE!** #13. In this article, the author says "Remember that you can read the BAM, header, and directory blocks as a single file name of $0 or $1. To read the data contained in these blocks you must use the GET# command since there are no carriage return delimiters". I followed this article and the tables listed in the article to make this program.

Lines 50100-50180 are for the title. If the title is not wanted or needed then replace the lines with this line. 50100 VV=253:GOSUB50320.

I did not check for any disk errors in this program. If needed, modify to your own personal needs. This program works on the 2040 disk and should work on the 3040 and 4040 disk systems.

```
READY.

 50000 REM
 50010 REM DIRECTORY FOR 3.0
 50020 REM
 50030 REM STEPHEN MEIROWSKY
 50040 REM
 50050 INPUT"DRIVE #0,1";M
 50060 A$=STR$(M):B$="I"+A$
 50070 OPEN 14,8,15:PRINT#14,B$
 50080 F$="$"+RIGHT$(A$,1)+",SEQ"
 50090 OPEN 1,8,2,F$
 50100 GET#1,G$:FO=ASC(G$)
 50110 VV=140:GOSUB50320
 50120 PRINT:PRINTM"▮"CHR$(34);
 50130 FORWW=144TO159:GET#1,G$:PRINTG$;
       :NEXT:PRINTCHR$(34)" ";
 50140 VV=2:GOSUB50320:PRINTG$;
 50150 GET#1,G$:PRINTG$" ";
 50160 IFFO=1THENPRINT" 1"
 50170 IFFO=65THENPRINT"2A"
 50180 VV=91:GOSUB50320
 50190 FORYY=1TO8:A$="":GET#1,G$
 50200 IFG$=""THENVV=28:GOSUB50320:GOTO50290
 50210 IFG$=CHR$(13)THEN50310
 50220 TT=ASC(G$):VV=1:GOSUB50320
 50230 FORWW=3TO18:GET#1,G$:A$=A$+G$:NEXT
 50240 VV=9:GOSUB50320:PRINTASC(G$);
 50250 PRINTTAB(5)CHR$(34)A$CHR$(34);
 50260 IFTT=129THENPRINTTAB(24)"SEQ"
 50270 IFTT=130THENPRINTTAB(24)"PRG"
 50280 GET#1,G$
 50290 IFYY<>8THENVV=1:GOSUB50320
 50300 NEXT:GOTO50190
 50310 CLOSE1:CLOSE14:END
 50320 FORWW=0TOVV:GET#1,G$:NEXT:RETURN
```

©

Editor's Note: This program will work on all PETs. For 4.0 (if using disk) you might want to relocate it. — RTM

Inversion Partitioning

David O. Williams
Toronto, Canada

Several people have written for **COMPUTE!** about methods of putting more than one BASIC program into the PET's memory. (See the articles by Elizabeth Deal, Charles Brannon and Harvey Herman in **COMPUTE!** #13.) The methods they use involve "partitioning" the memory into several blocks, each usually 8K in size, and arranging the pointers so that each block becomes the entire working memory space in which a program can be stored and run. While it is working in one block, the pointers prevent the PET from affecting the memory in the other blocks in any way. Programs which are stored there are undisturbed, waiting for the pointers to be realigned by an appropriate machine language routine.

This technique suffers from several disadvantages. The tape SAVE and LOAD commands do not work when a BASIC program is not in the usual position in memory, so various tricks such as saving through the machine language monitor have to be used. A far more serious problem is that each program is confined to 8K of memory. Any program which needs a total of more than 8K, including the space it uses for variables, arrays, and strings, will crash with an OUT OF MEMORY ERROR. Also, any program which performs a lot of string manipulations will run slower in a small memory space because of the greater frequency of garbage-collection delays. There may be plenty of spare memory in the other blocks which is effectively useless.

Providing you don't want more than two programs in your PET, you can avoid all of these problems by using the machine language routine here. It resides in the second cassette buffer, and its main effect is to turn the entire contents of the memory which is accessible to BASIC upside down! A BASIC program which is in the normal place at the bottom of memory is thus moved to the top. (It is also turned upside down, but that doesn't matter since no attempt is made to run it in that position.) A second program can then be written or loaded from tape into the bottom of memory. When this program is RUN it has access to all the unoccupied memory between the two programs which it can use for variables, arrays, and strings. (The machine language routine adjusts the pointers so that the second program cannot over-write the text of the first.) When you want to return to the first program, simply run the machine language routine again. It moves the first program back to the normal position, stores the second one at the top of memory, and adjusts the pointers so that the first program has access to all the working space without over-writing the second one. In this way, you can interchange the two programs as many times as you want. At any time, the program which is in the "running position" at the bottom of memory can be handled in all the normal ways for BASIC programs. You can SAVE it, edit it, delete it with NEW, LOAD another program, anything you want.

It all sounds very simple in theory, and in practice it isn't much more difficult. First, engage the machine language monitor by entering SYS 1024, then display the contents of the second cassette buffer with the command ".M 033A 03F1". The resulting display will look very like the listing except that the numbers in the body of the table will be different. Move the cursor to the first line of the table, change all the numbers to match the listing and press RETURN at the end of the line. Do the same thing to the second line, and so on to the end of the table. You have now written (or at least copied) a machine language program. Save it on tape (before you run it — this is important) with the monitor still engaged by using the command .S "EXCHANGE",01,033A,03F2. Press PLAY and RECORD, as requested by the machine, and wait for the cursor to reappear. (In the future you can get to this point by putting this tape into the deck, entering the monitor, and giving the simple command " .L". You will be told to press PLAY, and the tape will load into the correct memory locations.) Now exit the monitor by entering " .X". The familiar word READY will appear, and you are back to BASIC.

Now all you have to do to invert the memory is enter SYS 826. The process of turning the whole memory over takes a second or so, then the word READY will reappear.

Incidentally, Exchange will work with memories of any size, so even 8K PET owners can use it to get two short programs into their machines. Exchange will also leave intact any machine language you may have located in high RAM, *provided* that Exchange is initialized (i.e. run for the first

time) *after* the high-RAM program is in place and protected with the top-of-BASIC-memory pointer.

Have fun with it....!

```
Ø33A A2 ØØ EØ ØØ DØ 1D A5 35
Ø342 29 FE 48 38 E9 Ø1 8D 78
Ø34A Ø3 68 48 18 6A 69 Ø1 8D
Ø352 94 Ø3 68 69 Ø3 8D AA Ø3
Ø35A 8D B5 Ø3 AØ ØØ A5 Ø1 48
Ø362 A5 Ø2 48 A5 14 48 A5 15
Ø36A 48 A9 Ø1 85 Ø1 A9 Ø4 85
Ø372 Ø2 A9 FE 85 14 A9 3F 85
Ø37A 15 A1 Ø1 48 A1 14 81 Ø1
Ø382 68 81 14 E6 Ø1 C6 14 E4
Ø38A Ø1 DØ EE E6 Ø2 C6 15 A5
Ø392 15 C9 21 DØ E4 68 85 15
Ø39A 68 85 14 68 85 Ø2 68 85
Ø3A2 Ø1 38 A9 FF E5 2A 48 A9
Ø3AA 43 E5 2B 48 A9 FF E5 34
Ø3B2 85 2A A9 43 E5 35 85 2B
Ø3BA 68 85 35 68 85 34 EC 3D
Ø3C2 Ø3 DØ 17 A9 Ø4 8D 3D Ø3
Ø3CA 8E Ø1 Ø4 8E Ø2 Ø4 85 29
Ø3D2 85 2B A9 Ø1 85 28 A9 Ø3
Ø3DA 85 2A A5 2A 85 2C 85 2E
Ø3E2 A5 2B 85 2D 85 2F A5 34
Ø3EA 85 3Ø A5 35 85 31 6Ø ØØ
```

©

A Personal News Service

Ed Steinfeld
Hudson, NH

It's easy to get your home computer to type letters, maintain an inventory, store financial data, and play chess. But try to make it into a dumb terminal — that's difficult!

I have owned a PET 2001 and a TNW 232/488 (dual RS232 port) for about three years and used the RS232 port only to interface my Selectric. (I now own an 8032 CBM, 4040 disk, 2022 printer, TNW 232/488, Harris 1030 Selectric, and an acoustical coupler.) A few weeks ago I reread Jim Butterfield's article, "Basic CBM 8010 Modem Routines" in issue 7 of **COMPUTE!**. The 8010 programming is similar to the programming for the TNW 232/488 device. I entered the code with a few modifications, connected an acoustical coupler to the RS232 port, dialed a Digital Equipment Corporation (DEC) VAX-11/780, and I was on-line to another computer.

The VAX expected a number of characters I couldn't type from the CBM keyboard. I had to define a few keys for the VAX CONTROL characters. This was easy because Jim Butterfield used a table-driven translator in his program. This translator was used to convert CBM ASCII into standard ASCII and vice versa. I used this same table to convert the RVS, OFF, Left Arrow, and Up Arrow keys into CONTROL C, CONTROL Z, CONTROL S, and CONTROL P keys. These control keys abort programs, terminate input, and control scrolling. I got all this to work properly in about three weeks. Once the program worked, I contacted the Dow Jones News/Retrieval Service (DJNRS) (800-257-5114). Dow Jones & Company have an information service available to the home or business. This news service is not unlike the press wire services except the subscriber selects the information he wants rather than receiving everything being sent on the wire. The news includes what you might expect from the publishers of *The Wall Street Journal* and *Barron's*, stock and financial data. It also has general news, industry news, corporate profiles — 10-K extracts, financial statistics, and historical stock quotes. Current day quotes with a 15-minute delay on stocks, options, and corporate bonds as well as current day updates for U.S. Treasury issues, mutual funds, and the national over-the-counter market are all available.

I was given a password to try the service for a weekend. By Monday I had subscribed. My 8032, with a little software, can log into the Dow Jones News/Retrieval Service (DJNRS), request stock quotes and up-to-the-minute news, store it all on my disk, and print it on the 2022 printer. I now have my own personal news service.

The program I use to access the DEC VAX (VMS operating system) and PDP-11/70 (RSX-11M PLUS) software is the same software I use to talk to the DJNRS. However, Dow doesn't require any of the CONTROL characters, it only uses the question mark to stop output. All commands on the DJNRS are punctuation keys. The program first prompts with the terminal attributes that tell the various operating systems the characteristics needed for proper operation of the terminal. The DJNRS is available via a TYMNET system and requires the user to type a single character called the terminal identifier when he first accesses the system. (In some areas, the DJNRS is accessed through a TELENET service.) The terminal identifier lets the host software know when extra fill characters are required after carriage returns and when a line feed is needed between lines. I have found the identifier 'A' works well with my software.

On DEC, two fill characters are required after every return. Then the program asks if the terminal session should be written to disk and, if so, which drive and what file name. If you write the session to the disk, everything on the screen will be written into the file (both what you type as well as what is sent by the host computer). Even with this activity, the CBM can run at 300 baud without missing a character.

After you select the disk option, the program clears the screen except for the top line. This line in reverse video has the telephone numbers of the two systems I use. This leaves 24 lines for display data. Twenty-four lines is the standard display area of most video terminals and is the default page size for both the DJNRS and DEC systems.

A second program, PRINT TERMINAL, is used to print the terminal session. I save myself money by recording the data on the disk and later printing it after disconnecting from the service. The charges for DJNRS are by minute of connect time and the type of data being accessed. There is a one-time connect fee of $50 and the non-prime-time minute rate is between 15 and 20 cents.

The TERMINAL EMULATOR program has one gotcha when you use the option of writing to the disk. I have no way to stop the program other than to press the STOP key. Since I can't determine when I'm through, there is no way to close the disk file. So after I press the STOP key, I must remember to type DCLOSE (CLOSE 7 on PETs) to close the disk file. If I use some other key to determine when the session is finished, more code would have to be added to the lines that decode the input and output. More code would not allow the CBM to run at 300 baud. If you modify the 'business part' of the program, you may begin to lose characters. I have added about all that can be added and still keep up with the hosts. Options such as Command-O, Basic Toolkit, and the Wedge cannot be used because they slow the execution speed of the TERMINAL EMULATOR program.

With the exception of the 80-character reverse video line and setting the window, this terminal emulator program will work equally well with 40 column PETs and CBMs. In fact the news data displayed by DJNRS is 40 columns by 24 lines. You would think they expected all Commodore owners to be DJNRS customers. A Commodore 8010 modem may be used instead of the RS232 device and an acoustical coupler.

When using the DJNRS you can select headlines of current general news or specific headlines where a particular company is mentioned in the article. If one of the headlines is of interest, you then request to have the article displayed. You don't have to waste time reading every article to see if a company you're interested in is mentioned. The attached terminal session printout shows the login, Commodore's stock quotes between April 15 and May 1, an article in which Commodore is mentioned, followed by a list of headlines of articles in which Commodore is mentioned.

With this simple TERMINAL EMULATOR program, your PET or CBM can become a dumb terminal and can access any number of information services. Now your home computer is a window to general and financial news services and you can read the news as fast as they receive it — your own Personal News Service.

Program 1.

```
100 REM**"TERMINAL EMULATOR  3 MAY 1981 ¬
     ¬ COPYRIGHT (C) 1981"
110 REM**"ED STEINFELD"
120 REM**"31 RICHMAN ROAD"
130 REM**"HUDSON,  NH  03051"
140 REM*********************************
     ¬**********************************
     ¬*
150 REM**"CBM ASCII TO ASCII AND ASCII ¬
     ¬TO CBM ASCII CONVERTER"
160 REM**"THIS PROGRAM WILL EMULATE A ¬
     ¬VT52 WITHOUT CURSOR CONTROL.
170 REM**"USE TERMINAL IDENTIFIER 'A'"
180 REM**"EITHER A TNW 232D, OR 488/232 ¬
     ¬WITH AN ACOUSTICAL COUPLER
190 REM**"OR A CBM 8010 MODEM MAY BE ¬
     ¬USED.
200 TE=22:REM"     INSERT THE UNIT ¬
     ¬NUMBER OF RS232 OR MODEM DEVICE."
210 O=0:A$="":B$="":C$="":Q=0:I=0:J=0:
     ¬K=0:Q$="":DIM F(255),T(255)
220 PRINT"hhh              TERMINAL ¬
     ¬EMULATOR BY E.F. STEINFELD (C) ¬
     ¬1981"
230 FORJ=32 TO 64: T(J)=J:NEXT:T(13)=13:
     ¬T(18)=3:T(20)=127:T(7)=7:T(8)=20
240 FORJ=65 TO 90:K=J+32:T(J)=K:NEXT:
     ¬T(9)=9
250 FORJ=91 TO 95:T(J)=J:NEXT
260 FORJ=193 TO 218: K=J-128:T(J)=K:NEXT
270 T(160)=32:T(141)=13:T(94)=19:
     ¬T(95)=17
280 FORJ=0 TO 255:K=T(J):IF K THEN ¬
     ¬F(K)=J:F(K+128)=J
290 NEXT:F(94)=94:F(95)=95:F(8)=20:
     ¬T(146)=26:F(27)=147
300 F(141)=13
310 POKE1020,0:POKE59468,14
320 OPEN5,TE
330 A$="    TERMINAL EMULATOR         "
340 PRINT"hhhr DJNS TEL. NO. 889-8618 ¬
     ¬*** DEC TEL. NO. 884-1707"A$
350 PRINTCHR$(15):FORI=1TO80:PRINTCHR$(9
     ¬)CHR$(137);:NEXT:PRINT:REM"CLEARS ¬
     ¬TABS
360 T$=CHR$(137):PRINTTAB(8)T$TAB(16)T$T
     ¬AB(24)T$TAB(32)T$TAB(40)T$TAB(48)T
     ¬$;
370 PRINTTAB(56)T$TAB(64)T$TAB(72)T$:
     ¬REM "SETS TABS EVERY 8 COL."
380 PRINT"      TYMNET   TERMINAL ¬
     ¬IDENTIFIER 'A'
390 PRINT"      VMS        SET TERMINAL/CRF
     ¬ILL=2
400 PRINT"      RSX-11M  SET TERMINAL ¬
     ¬HFILL:2
410 PRINT "↓      RVS = CTRL C, OFF = ¬
     ¬CTRL Z, ^ = CTRL S, ^ = CTRL P"
420 PRINT "↓      AFTER LOGOFF HIT ¬
     ¬rSTOPf TO TERMINATE THE PROGRAM."
430 INPUT"↓DO YOU WISH TO WRITE THIS ¬
     ¬SESSION TO THE DISK (Y OR N)   ¬
     ¬N←←←";Q$
440 Q$=LEFT$(Q$,1):IFQ$="N"THENQ=0:
     ¬GOTO520
450 IF Q$<>"Y"GOTO430
460 PRINT:PRINT"DON'T FORGET TO TYPE ¬
     ¬DCLOSE AFTER YOU STOP."
470 INPUT"↓DRIVE NO. (0 OR 1) _0←←←";D:
     ¬IFD<0ORD>1THEN470
480 INPUT"FILE NAME___←←←";Q$:IFQ$="_"TH
     ¬EN480
490 Q=1
500 DOPEN#7,D(D),(Q$)+",W"
510 IFDS<>0THEN PRINTDS$:END
520 PRINT"h";
530 GETA$:IFA$THENPRINT#5,CHR$(T(ASC(A$)
     ¬));
540 GET#5,A$:IFST=OANDA$<>B$THENC$=CHR$(
     ¬F(ASC(A$))):PRINTC$;:IFQTHENPRINT#
     ¬7,C$;
550 GOTO530
READY.
```

ECLECTIC SYSTEMS CORPORATION

Authorized Commodore service center
Repair of the complete line of Commodore products
In a hurry? Check our modular exchange program

HARDWARE:	
CBM 8032 Computer, 80 Column	$1195
CBM 8050 Disk Drive	1395
CBM 4032 Computer, 40 Column	995
CBM 4040 Disk Drive	995
CBM 4022 Printer	649
CBM VIC 20 Computer	263
CBM VS100 Cassette	68
PET to IEEE Cable	33
IEEE to IEEE Cable	39
BASF Diskette, Box of 10	30

SOFTWARE:	
OZZ	$299
Wordcraft 80	299
Tax Preparation System	380
IRMA	380
Dow Jones Portfolio Management System	115
Personal Tax	55
Pascal	229
Assembler Development Package	77
Wordpro 4+	329

Order TOLL FREE 1+800-527-3135

10 AM to 4 PM CDT Monday through Friday

Texas residents call 1+214-661-1370

VISA, MASTER CHARGE, MONEY ORDERS, AND C.O.D. "Certified Check" accepted.
Units in stock shipped within 24 hours, F.O.B. Dallas, Texas.
All equipment shipped with manufacturer's warranty.

Residents of Texas, Louisiana, Oklahoma City and Tulsa, Oklahoma must add applicable taxes.

Eclectic shortly will be announcing products that are designed to work with CBM systems.

1. ROMIO: two RS232 ports—three parallel ports—26K EPROM memory-managed alternate character set, software controlled—EDOS (extended DOS).
2. Terminal program (options with ROMIO)
3. EPROM programmer
4. Front-end processor
5. Additional firmware to be announced

Be sure to write the address below for more information; dealer inquiries welcome.

P.O. Box 1166 • 16260 Midway Road • Addison, Texas 75001 • (214) 661-1370

Table 1.

```
please type your terminal identifier
-1040-134-
please log in: dow1;
to>  host is online

WHAT SERVICE PLEASE????
djns
ENTER PASSWORD
WWWWWWWWWWWWWWWW
MMMMMMMMMMMMMMMM
@@@@@@@@@@@@@@@@
※   ※
DOW JONES NEWS/SERVICES  COPYRIGHT 1981
DOW JONES & CO., INC.
NOW..FREE TEXT SEARCHABLE NEWS DATABASE
LOW INTRODUCTORY RATE CALL 800-257-5114
ENTER QUERY
※   ※;1cbu p1

DOW JONES HISTORICAL
STOCK QUOTE REPORTER SERVICE

STOCK 1CBU

 DATE       HIGH       LOW      CLOSE   VOL(100/S)
04/15/81    36        34 3/8    36         752
04/16/81    38 3/8    36 3/8    38 3/8    1717
04/20/81    42 3/8    37 3/4    42        1173
04/21/81    43 1/2    41 1/2    42        1481
04/22/81    44        40 1/2    40 1/2    1152
04/23/81    42 3/8    39 7/8    40 1/2     993
04/24/81    42 1/2    39 3/4    42 1/2    1079
04/27/81    42 3/4    40 3/8    40 5/8     489
04/28/81    40 1/2    37 5/8    39 1/2     928
04/29/81    39        37 1/8    37 3/4    1156
04/30/81    38 3/8    36 1/8    36 1/8     591
05/01/81    37 5/8    35 3/4    37         363

※   ※.cbu
 N  CBU        01/02 AN 1/3
  /AAPL  CBU   TAN   DGN  /OFF    /
    05/01 SOFTWARE PIRATES A WORRY
  (WJ) FOR PERSONAL COMPUTER MAKERS
  THE BOOMING MARKET IN PERSONAL
COMPUTERS OWES ITS VITALITY NOT ONLY TO
THE LOW PRICES OF THE MACHINES
THEMSELVES BUT ALSO TO THE QUALITY OF
THE PROGRAMS, OR SOFTWARE. NOW, MANY IN
THE EMERGING PERSONAL COMPUTING
INDUSTRY ARE WORRIED BY A RECENT RISE
IN WHAT  SOME CALL SOFTWARE PIRACY.
INCREASINGLY, INDIVIDUALS ARE
REPRODUCING  MICROCOMPUTER PROGRAMS
THAT AREN'T  SUPPOSED TO BE DUPLICATED.
  MOST OF THE PROGRAMS ARE COPYRIGHTED,
AND THE AUTHORS HAVE USED SOPHISTICATED
RECORDING TECHNIQUES TO GUARD AGAINST
COPYING. MOST PEOPLE MAKING THE
DUPLICATES SEEM TO BE DOING SO AS
INSURANCE IN CASE SOMETHING GOES WRONG
WITH THEIR ORIGINAL, AS IT OFTEN DOES.
  BUT MANY OTHERS ARE GIVING COPIES
AWAY BY THE DOZEN, AND SOME ARE SELLING
※   ※
 N  CBU        01/02 AN 2/3
THEM.
  BUT MANY OTHERS ARE GIVING COPIES
AWAY BY THE DOZEN, AND SOME ARE SELLING
THEM.
  BY ONE ESTIMATE, UNAUTHORIZED COPYING
HAS RISEN FROM ECONOMIC INSIGNIFICANCE
TO CUT SALES OF THE $200-MILLION-A-YEAR
MICROCOMPUTER SOFTWARE PUBLISHING
INDUSTRY BY $12 MILLION TO $36 MILLION
ANNUALLY.
  AT PRESENT, THE COPYING AFFECTS ONLY
SUCH RELATIVELY SMALL MAKERS OF
PERSONAL DATA-PROCESSING EQUIPMENT AS
APPLE  COMPUTER, COMMODORE
INTERNATIONAL AND  TANDY CORP. AND THE
EVEN SMALLER  COMPANIES THAT WRITE AND
PUBLISH MOST OF THE PROGRAMS FOR SUCH
MACHINES. BUT IT  ALSO COULD HINDER NEW
VENTURES BY THE  LARGER, TRADITIONAL
COMPUTER COMPANIES,  SUCH AS DATA
GENERAL, DIGITAL EQUIPMENT,
INTERNATIONAL BUSINESS MACHINES AND
WANG LABORATORIES THAT ARE EXECTED TO
※   ※
 N  CBU        01/02 AN 3/3
ENTER  THE PERSONAL COMPUTER MARKET
WITH  PRODUCT INTRODUCTIONS AT THE
NATIONAL  COMPUTER CONFERENCE OPENING
MONDAY IN  CHICAGO.
  TO SOME DEGREE, INDUSTRY OBSERVERS
SAY, SOFTWARE PUBLISHERS BROUGHT THE
COPYING PROBLEM ON THEMSELVES BY
FAILING TO PROVIDE ADDITIONAL OR
REPLACEMENT  COPIES OF IMPORTANT
PROGRAMS QUICKLY. IN SOME CASES, USERS
HAVE COME TO DEPEND ON UNCOPYABLE
PROGRAMS THAT WENT AWRY, ONLY TO FIND
THAT THE PROGRAM TOOK WEEKS TO
REPLACE.
  "ANY COMPANY THAT DOESN'T PROTECT ITS
USERS BETTER THAN THAT DESERVES EVER
BOOTLEG COPY MADE," SAYS ALLAN
TOMMERVIK, MANAGING EDITOR OF SOFTALK,
A MAGAZINE FOR OWNERS OF APPLE
COMPUTERS.
※   ※
 NO PAGE
※   ※.cbu 01
 N  CBU       01/02
 AN 05/01 SOFTWARE PIRATES A WORRY
  (WJ) FOR PERSONAL COMPUTER MAKERS
 AM 04/22 COMMODORE INTL OUTLOOK -2-
  (DJ)
 AL 04/22 COMMODORE INTL SEES 4TH QTR
  (DJ) AS BEST IN COMPANY HISTORY
 AK 04/22 COMMODORE INTL EARNS -2-
  (DW)
 AJ 04/22 COMMODORE INTL LNET 66C A SHR VS 38C
 AI 04/13 UP & DOWN WALL ST: IBM
  (BN) SAID TO HAVE SMALL COMPUTER
 AH 04/02 COMMODORE INTERNATIONAL
  (DW) UNVEILS MICROCOMPUTER
 AG 03/26 TANDON CORP RECEIVES
  (DW) $9.3 MILLION ORDER
 AF 03/05 COMMODORE INTERNATIONAL GETS
  (DW) $6.2 MILLION ORDER FROM ATARI
 AE 03/04 IBM SAID TO WEIGH MATSUSHITA
  (WJ) AS MICROCOMPUTER SUPPLIER
 AD 03/03 COMMODORE INTL ISSUES
  (DJ) STATEMENT ON BARRON'S STORY
※   ※
 N  CBU       02/02
 AC 03/02 COMMODORE INTERNATIONAL
  (BN) PROSPECTS UNCERTAIN
 AB 02/18 COMMODORE INTL. SEES HIGHER
  (DW) FISCAL 1981 NET -2-
 AA 02/18 COMMODORE INTL SEES FISCAL
  (DW) 1981 NET OF MORE THAN $2.40 A SH
※   ※disc
```

©

FOR/NEXT GOSUB/RETURN, And The Stack

Jim Butterfield
Toronto, Canada


```
100 print "stack sniffer      jim butterfield"
110 data 186,189,3,1,160,1,217,-11,240,4,136,16,248,96
    ,185,-9,32,210,255
120 data 185,-7,32,210,255,169,32,32,210,255,138,24,
    121,-5,141,-1,138,121
130 data -3,170,188,4,1,189,3,1,132
140 data 300,107,137,133,136,32,148,220 : remark
    – orig
150 data 300,27,55,133,54,32,206,220 : remark – upgr
160 data 300,66,55,133,54,32,120,207 : remark – 4.0
170 data 169,13,32,210,255,174,-1,208,189,0,129,
    141,76
180 data 83,80,66,18,7,14,1,0
190 r=peek(65534)
200 v=52: if r=107 then v=134
210 a=peek(v)+peek(v+1)*256
220 b=a-80:b%=b/256
230 poke v,b-256*b%:poke v+1,b%
240 poke v-4,b-256*b%:poke v-3,b%
250 for j=0 to 79
260 read x:t=t+x
270 if x<0 then y=a+x:x=int(y/256):poke j+b,y-256
    *x:j=j+1
280 if x<>300 goto 320
290 read x:t=t+x:if x=r goto 310
300 for k=1 to 6:read x:t=t+x: next k:goto 260
310 for k=1 to 5:read x:t=t+x:poke j+b,x:j=j+1:
    next k: read x:t=t+x
320 poke j+b,x
330 next j
340 print"analyze stack with sys";b
350 print"(checksum =";t;")"
360 print"checksum should be 9965"
```

FOR...NEXT loops are quick and easy to code. GOSUB to a subroutine with RETURN at the end works very nicely. They never give you trouble. Except, once in a while, very subtle trouble. An ?OUT OF MEMORY when you know you have lots of memory left ... a ?NEXT WITHOUT FOR when you can see the FOR in the listing. What's going on here?

We'll try to establish a few rules by means of experiments. Type in the program with the elegant name of Stack Sniffer. Run it, and it will park itself out of the way in high memory; it will remain there until you power down. Make a note of the SYS address; we're going to use it quite a few times. If the checksum is wrong, check your data statements: there must be an error in there.

Checking Out The Program

Type NEW. Now type in:

CBM/PET? SEE SKYLES ... CBM/PET?

"Should we call it Command-O or Command-O-Pro?"

That's a problem because this popular ROM is called the Command-O-Pro in Europe. (Maybe Command-O smacks too much of the military.)

But whatever you call it, this 4K byte ROM will provide your CBM BASIC 4.0 (4016, 4032) and 8032 computers with 20 additional commands including 10 Toolkit program editing and debugging commands and 10 additional commands for screening, formatting and disc file manipulating. (And our manual writer dug up 39 additional commands in the course of doing a 78-page manual!)

The Command-O extends Commodore's 8032 advanced screen editing features to the ultimate. You can now SCROLL up and down, insert or delete entire lines, delete the characters to the left or right of the cursor, select TEXT or GRAPHICS modes or ring the 8032 bell. You can even redefine the window to adjust it by size and position on your screen. And you can define any key to equal a sequence of up to 90 key strokes.

The Command-O chip resides in hexadecimal address $9000, the rightmost empty socket in 4016 and 4032 or the rearmost in 8032. If there is a space conflict, we do have Socket-2-ME available at a very special price.

Skyles guarantees your satisfaction: if you are not absolutely happy with your new Command-O, return it to us within ten days for an immediate, full refund.

Command-O from Skyles Electric Works **$75.00**

Complete with Socket-2-Me **95.00**

Shipping and Handling .... *(USA/Canada)* **$2.50** *(Europe/Asia)* **$10.00**

California residents must add 6%/6½% sales tax, as required.

Skyles Electric Works
231E South Whisman Road
Mountain View, California 94041
(415) 965-1735

Visa/Mastercard orders: call tollfree **(800) 227-9998** (except California). California orders: please call **(415) 965-1735.**

WATCH

THIS

SPACE

CBM/PET? SEE SKYLES ... CBM/PET?

"You mean this one little Disk-O-Pro ROM will give my PET twenty-five new commands?

And for just $75.00? Why, that's only $3.00 a command!"

The Disk-O-Pro in any PET with Version III (BASIC 2.0) ROMs **(### COMMODORE BASIC ###)** will give 19 software compatible disk instructions*: 15 identical with the new BASIC 4.0 (or with 8032 ROMs) compatible with both old and new DOS. Plus 4 additional disk commands...including appending **(MERGE)**, overlaying **(MERGE #________)** and **PRINT USING**, allowing formatting output of strings and numbers on the PET screen or on any printer.

**NOTE: Old DOS doesn't recognize three of the commands.*

Those are just 3 of the important commands—and there are 7 more beauties—on your Disk-O-Pro that have never been available previously to PET/CBM users. (Skyles does it again!)...Beauties like the softtouch key **(SET)** which allows you to define a key to equal a sequence of up to 80 keystrokes; like **SCROLL** whereby all keys repeat as well as slow scrolling and extra editing features; like **BEEP** which allows you to play music on your PET.

The Disk-O-Pro is completely compatible with the BASIC programmer's Toolkit. The chip resides in the socket at hexadecimal address $9000, the rightmost empty socket in most PETS. And for the owners of "classic" (or old) PETS, we do have interface boards.

(For those owning a BASIC 4.0 or 8032, even though the Disk-O-Pro may not be suitable, the Command-O is. Just write to Skyles for additional information. Remember, we have never abandoned a PET owner.)

Complete with 84-page manual written by Greg Yob...who was having so much fun that he got carried away. We had expected 32 pages.

Skyles guarantees your satisfaction: if you are not absolutely happy with your new Disk-O-Pro ROM chip, return it to us within ten days for an immediate full refund.

Disk-O-Pro from Skyles Electric Works **$75.00**

Complete with interface board (for "classic" PETS) **95.00**

Shipping and Handling .... *(USA/Canada)* **$2.50** *(Europe/Asia)* **$10.00**

California residents must add 6%/6½% sales tax, as required.

Skyles Electric Works
231E South Whisman Road
Mountain View, California 94041
(415) 965-1735

Visa/Mastercard orders: call tollfree **(800) 227-9998** (except California). California orders: please call **(415) 965-1735.**

WATCH

THIS

SPACE

```
100 FOR J=1 TO 50
200 FOR M=7 TO 3
```

Type the SYS command that Stack Sniffer wants. The system should respond READY. That means that there are no unclosed loops or unfinished subroutines. We haven't run the program yet.

Conclusion 1: You don't build loops or subroutines until you run the program.

Now type RUN. The system will answer READY immediately. Now give the SYS command. You should get:

```
LP IN 200
LP IN 100
```

Note that the loops are reported from the innermost out.

Conslusion 2: Loops stay open if they are not closed, even if the program is finished.

Add line 150 to the program:

```
150 GOSUB 200:STOP
```

Type the SYS command without RUNning. You'll get READY.

Conclusion 3: Adding or changing a line clears all loops, just as it clears all variables.

Say RUN, and then give the SYS command again. You'll get three lines back:

```
LP IN 200
SB IN 150
LP IN 100
```

Conslusion 4: Loops and Subroutines seem to "nest," one within another.

By this time, Stack Sniffer has been checked out well enough that we know it is OK.

Loops Within Subroutines

Now add the line: 300 RETURN. If you read through the program, you will see that the program calls the subroutine, returns from it, and then stops. The loops at 100 and 200 have never been closed. What will be on the stack? Type RUN, then give the SYS and see.

What? The loop at 200 has been closed!

Conclusion 5: If a FOR loop is opened within a subroutine, it will be terminated when the subroutine returns.

This can be a good way to clean out a FOR-...NEXT loop structure that is no longer needed.

Loops Within Loops

Delete line 150. Type in line 300 NEXT J. LIST the program and see what you think will happen to the loops when you RUN. Now try it.

Result: no loops, even though we didn't close the inner one.

Conclusion 6: Closing an outer loop always closes any loops inside.

Type in 150 IF J=7 THEN STOP. In this case, we're going to stop the program before the outer loop has completed its cycle. What will happen to the inner loop? RUN and then give SYS and see.

Result: the J loop is open, but the K loop is closed. How come? It seems that the NEXT J must close the K loop.

Conclusion 7: Performing NEXT on an outer loop always closes all loops inside.

This rule is what makes it impossible for you to incorrectly nest loops. If you were to type: FOR X=1 TO 4:FOR Y=1 TO 3:PRINT"H":NEXT X:NEXT Y you'd get a ?NEXT WITHOUT FOR error message. Do you see why?

Early Exit From Loops

Say NEW and start over. This time, we'll enter the program:

```
300 FOR A=3 TO 9
350 FOR B=4 TO 15
400 IF A+B=20 GOTO 600
450 NEXT B
500 NEXT A
550 STOP
600 FOR F=1 TO 7
650 NEXT F
```

Think about this one. This kind of coding can happen regularly when you are looking things up in a table. When you find what you want (line 400) you exit the loop and go on to other business.

RUN the program and then give the SYS call.

Surprised? The loops at 300 and 350 were never closed. They are still active.

Conclusion 8: Jumping out of a loop leaves the loop active. This can run you out of stack space very quickly.

What can we do about this? Let's try the following, which doesn't seem to do anything significant: change lines 600 and 650 to read: FOR A ... and NEXT A. RUN and SYS.

Wow! We've cleaned everything up. How did we do it? The only change was in the name of the loop variable. A was previously used for the outermost loop: re-opening a loop using A closed the old one and closed the B loop too.

Conclusion 9: Re-opening a loop closes the previous loop activity and all inner loops.

Picky point for technical tyros: this always works except when you try to do it after calling a subroutine: you may remember that the subroutine "separates" the FOR/NEXT loop entries.

Interlude

We've arrived at a very important coding moral: if you jump out of a loop, be prepared to do something about it. Otherwise, the loop stays around and this can result in one of two possible problems.

First, your stack (where the loop information is kept) will fill up and you'll get ?OUT OF MEMORY even though it seems you have lots of memory. Annoying.

Alternatively, you may accidentally open a loop using the same variable name much later in the program. What's worse (pay careful attention here), by this time we're inside some genuine loops that we are using. Now: as far as BASIC is concerned, you're re-opening the outermost loop; it's outermost because it was opened a long time ago and all subsequent loops were fitted inside. Think about that, and the enormity of what's going to happen will dawn on you ... BASIC will close absolutely every loop it's got before opening the one you're asking for. You'll get ?NEXT WITHOUT FOR when you try to exercise one of the outer loops. Insidious.

How do you police this problem? It's not practical to insist that you never jump out of a loop until it's completed. You have several alternatives:

*Do this kind of loop activity in a subroutine. When you return, the loop will be closed automatically.

*Name all your outermost loops by the same name, I. Then name the next level inside with the same name, J ... and so on. This ensures that every time you exit from a loop you will soon open another with the same name.

*Write a dummy loop on the same variable immediately after you exit. FOR Y=1 TO 1: NEXT Y will do the trick nicely. The Y loop will end up good and closed.

Subroutines

If you write your subroutines correctly (jump into the beginning, have a neat RETURN by itself at the end, don't jump out), you won't have any trouble. The thing that beginners forget in the heat of the moment is that you can't just GOTO out of a subroutine directly into your main coding.

Type NEW and enter the following program:

```
100 GOSUB 300
110 STOP
300 X=X+1
310 IF X=1 THEN GOSUB 300
```

We deliberately have left off the RETURN to see what's happening here. This is an example of coding beloved to some clever types, but utterly horrible to most of us. It's called recursive programming, with the emphasis on the curse. What that means is that the subroutine at 300 calls itself. Do it and see what happens with SYS.

Hmmm. Both calls are neatly sitting there ... the one from 100 and the subsequent one from 310.

Conclusion 10: Subroutines don't work like loops. Calling the same one twice doesn't clean the old one out.

Add 320 RETURN and RUN it again. Call SYS. Check the value of X and see if it is what you expect.

Conclusion 11: Recursive subroutines will work, if you're that fool hardy.

Now we're ready for our final experiment in this series. Type new and enter the single line:

```
500 GOSUB 500
```

This is the ultimate in recursive subroutines. The subroutine calls itself, and then calls itself ... and so on. It will give you an ?OUT OF MEMORY incredibly fast. Adding a RETURN won't help; the stack is used up in milliseconds. RUN it, read the bad news, and then give the SYS command.

Huh? The stack is clean. How can that be? Well ...

Conclusion 12: Some error messages clean the stack, removing all GOSUB calls and FOR... NEXT loops.

Those of you who love speed contests: type RUN, and then see if you can hit the RUN/STOP key quick enough to catch the program before it bombs. It can be done, but you'll need super nimble fingers. If you do stop it in time, you might like to give the SYS command and see all those subroutine calls stacked up.

Conclusion

FOR ... NEXT and GOSUB/RETURN have a few subtleties. It's well to be aware of them.

It doesn't hurt to keep a copy of Stack Sniffer around and try them on your own programs. How many loops or subroutine calls have you left open? Look through your programs, and draw your own conclusions... ©

New Listing Conventions For CBM

New machines — VIC and "FAT-40" — and 4.0 BASIC have added a host of new editing functions and color control codes. To make **COMPUTE!**'s program listings as easy as possible for you to type in accurately, we will list CBM programs in a new, simpler way. Starting in this issue, you will see that our previous method — reconstructed cursor symbols — has been replaced by bracketed words. [DOWN] will mean the cursor-down key. [3 LEFT] will mean three cursor-lefts, and so on.

We will continue to split program lines with the ˉ symbol. It signals that the line is continued below and prevents any spaces from being hidden. All shifted characters and graphics are represented by their underlined non-graphics equivalent. Line 110 in David Swaim's article, "High Resolution Bar Graphics for the PET," is a good example of the new conventions. We hope you will agree that this change will simplify your typing of programs. Let us know how you feel.

Here is a table of the new conventions:

Key To COMPUTE!'s CBM Listings

All Machines

Clear Screen	{CLEAR}
Home Cursor	{HOME}
Cursor Up	{UP}
Cursor Down	{DOWN}
Cursor Right	{RIGHT}
Cursor Left	{LEFT}
Insert Character	{INST}
Delete Character	{DEL}
Reverse field on	{RVS}
Reverse field off	{OFF}

CBM 8032/"FAT 40"

Set Window Top	{SET TOP}
Set Window Bottom	{SET BOT}
Scroll Up	{SCR UP}
Scroll Down	{SCR DOWN}
Insert Line	{INST LINE}
Delete Line	{DEL LINE}
Erase to Beginning	{ERASE BEG}
Erase to End	{ERASE END}
Toggle TAB	{TGL TAB}
TAB	{TAB}
ESCape key	{ESC}

©

Nuts And Volts:
Build Your Own Controllers: Part III

Gene Zumchak
Buffalo, NY

In the first installment of this series of articles on building your own controllers, I outlined what you would need as a minimum to develop independent controller systems. Specifically, you need a development system consisting of a computer system with some kind of machine language operating system, an EPROM PROGRAMMER, and an EPROM emulator. A console computer system like an APPLE or PET will do quite well as the computer system, provided some machine language capability has been added. That is, PEEK and POKE are not sufficient for reading and writing memory. While hand assembly of programs can get the job done, the use of text editor and assembler is preferred.

EPROM programmers are available from several sources for a wide range of prices. EPROM emulators, however, are less common. In the second installment of this series, the design of an EPROM emulator was provided. In Part III, we will begin to look at what is required for the target system itself.

Basically, there are two extreme approaches you can take; you can buy everything, or build everything. While the first may look attractive to someone with no hardware experience, even if you can buy a suitable CPU board, there will undoubtedly be some custom I/O that you will have to provide. On the other hand, if you can afford the few extra bucks for a ready-made CPU board, the savings of your time may make the ready-made board a bargain. The advantages of doing it all yourself are usually compactness (low parts count) and low parts cost. You ought to be able to put together a complete system for about $100.

The cost for a ready-made CPU board will start at about $100. John Bell and Brachman Associates are two outfits that come to mind with boards in this range. If your application requires that it be field-programmable, then you should consider a board with a built-in keyboard/display and/or a serial interface. A versatile board having these features and a reasonable price is the SYM. The SYM has a powerful 4K machine language monitor. Two 6522s are availble for I/O and a third can be plugged in. Sockets for extra RAM and EPROM are also available, and the SYM can be jumpered to reset into your program. If ASCII style I/O or hard copy is necessary, the AIM-65 should be considered, although you are now talking over $400.

Power Supply

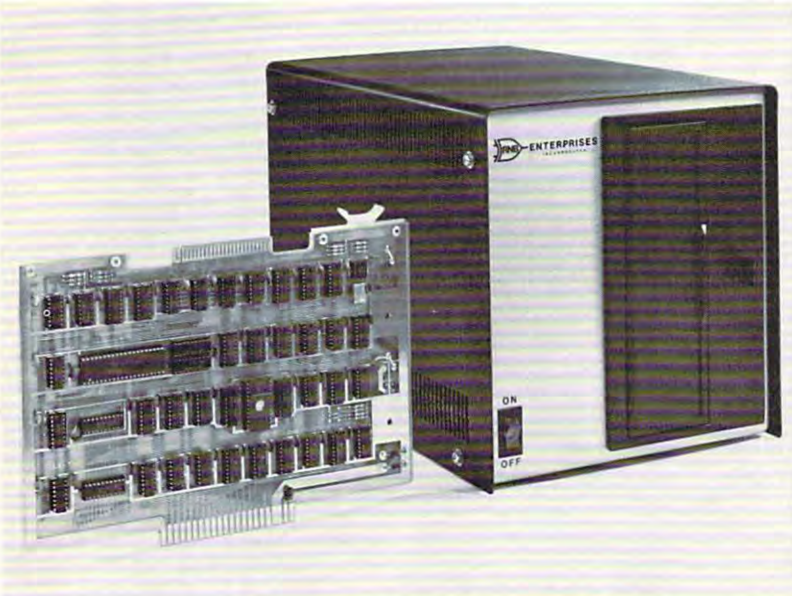
Whether you build your own board, or use a ready-made, single-board computer like the SYM, you'll need to provide a power supply. Kits and assembled 5-volt supplies are reasonably priced, or you can build your own.

Obtaining a regulated voltage from a raw DC voltage is easy with a three-terminal voltage regulator IC. The 7805 can provide one amp if adequately heatsunk and costs less than a buck. Obtaining the raw DC is only a little more work. Having to bring high-voltage AC into your board or system can be hazardous, but this can be avoided by buying what is called a "wall" transformer. This is a molded box containing a transformer that plugs directly into an AC socket. The output is low-voltage AC or DC. For currents 500 ma or less, the rectifiers and filter capacitor are usually included and the output is a DC voltage. Such supplies have been used with calculators for a long time. For larger currents, only AC is available and you will have to provide the rectifiers and filters. Wall transformers are available from a number of mail order outfits. For most controller applications, a single five-volt supply with a one-amp capacity will probably be adequate. Figure 1 shows such a supply made with an AC wall transformer.

While the 7805 can provide 1 amp with adequate heatsinking, since they are so inexpensive, a better idea is to share the load among two or more. Note: this does not mean paralleling the outputs. Instead, the five-volt loads are split up, each piece getting its own regulator. By running the regulators considerably below their capacity, heatsinking can be reduced or eliminated. Another possibility is using a regulator with a larger capacity. The LM323, for example, comes in a metal power

VAK-7 8″ FLOPPY DISK SYSTEM

FOR AIM-65, SYM-1



The VAK-7 Disk System was specifically designed for use with AIM-65 and SYM-1 Microcomputer Systems. The VAK-7 is a complete full size (8″) Floppy Disk System. This system will Read, Write and Format IBM Single and Dual Density diskettes. The VAK-7 is available with both Single and Dual Sided Disk Drives. Each Disk Drive comes with its own cabinet and Power Supply. The VAK-7 can handle up to 4 disk drives, totaling more than 4.98 Megabytes of storage.

The VAK-7 Disk System incorporates both advanced hardware and innovative software designs. The addition of the VAK-7 produces a very powerful and useful computer system. Unlike most other disk systems, there is no requirement for the user to provide RAM to hold the Disk Operating System software. No valuable time is wasted loading in the DOS from cassette tape, because the VAK-7 DOS software is in onboard ROM. The VAK-7 is located above the 32K boundary (8000 HEX), leaving the user with a full 32K bytes of contiguous address space for his own use.

AIM-65—Allows the user to save and load object code thru the AIM Monitor; to load, save, and append Text thru the AIM Editor; to load, save, and append Basic Programs thru the BASIC INTERPRETER; to assemble directly from disk single or multiple file programs.

SYM-1—Allows the user to save and load Files for use with the SYM Monitor, SYM Basic, and RAE-1.

ADDITIONAL COMMANDS:

ACTIVATE A DELETED FILE
COMPRESS A DISK
RENAME A DISK FILE
COPY A DISK
FORMAT A DISK
DELETE A DISK FILE
INITIALIZE A DISK
LIST CATALOG

SPECIFICATIONS:

- Completely assembled, tested, and burned in.
- Occupies address 8000–8FFF for AIM-65, $9000–9FFF for SYM-1, or $E000–EFFF for KIM-1.
- IBM Format; Single Density (128 bytes/sector); Dual Density (256,512, or 1024 bytes/sector).
- All ICs are in sockets.
- Fully buffered address and data bus.
- Standard KIM-4* BUS (both electrical pin-out and card size).
- Designed for use with a regulated power supply, but has provisions for adding regulators for use with an unregulated power supply.
- Dimensions: Board—10″ wide x 7″ high (including card-edge). Cabinet—9.25″ wide x 10″ high x 16″ deep.
- Power Requirements: +5v DC @ 2 Amps.
  117 AC 60Hz @ 2 Amps.
- Shipping Weight: 25 lbs.

**KIM-4 is a product of MOS Technology/C.B.M.*

PRICING:

VAK-7 **$1,299.00**
Controller and One Single-Sided Drive

VAK-7A **$599.00**
Additional Single-Sided Drive with Cabinet and Power Supply

VAK-7B **$1,599.00**
Controller and One Dual-Sided Drive

VAK-7C **$899.00**
Additional Dual-Sided Drive with Cabinet and Power Supply

CALL OR WRITE FOR FREIGHT CHARGES

RNB ENTERPRISES INCORPORATED

4030 N. 27th Avenue, Suite D
Phoenix, AZ 85017
(602) 265-7564

master charge VISA

transistor package and can supply three amps when heatsunk. The case or mounting tab for most positive, three-terminal regulators is the ground connection, permitting them to be attached directly to a grounded chassis or case.

With a 5-volt only EPROMS, like the 2716, being so cheap, it is now relatively easy to avoid parts that require other than a single five-volt supply.

The CPU

A controller will minimally consist of a CPU, an EPROM, some RAM, and some I/O. The first three items need not vary much from application to application, and you may even wish to lay out a standard controller board. The I/O, of course, will be a function of the application. If you are a dyed-in-the-wool 6502 fan, the choice of CPU is easy. While the 6502 is available in 28-pin packages (6503,6504, and 6505), the couple of square centimeters in board space saved is minimal. While you won't need all of the address space of the 6502, it will make decoding easier. Besides, since they are more common, 6502s are cheaper anyway. If you are open minded, and have assembled programs for more than one processor, you may consider other processor types, perhaps the Z-80 or the 6809, or a ROMless single-chip type. Since we are emulating program EPROM and not the processor, our development system can be used with any processor that can use EPROMs. It is a good idea to pick a processor that has built-in clock circuitry and only needs five volts. For convenience, we'll assume that we are using the 6502.

The 6502 can generate its own two-phase clock in a number of ways. The simplest is to use an RC network. While this method saves the price of a crystal, the loss of the accurate crystal time-base precludes accurate timing using either software or programmable timers, and is a false economy. A crystal may be connected in either a series or parallel mode. The series connection used by the SYM and KIM is shown in Figure 2a. Some asymmetry between the two phases usually occurs with this connection. Symmetry can be assured by applying a TTL square wave to the phase-zero input. The AIM uses this method starting with a four MHz crystal and a classical oscillator circuit. A pair of flip-flops are used to generate quadrature one MHz signals from the four MHz clock. (Only one is used.) A more straightforward circuit that merely divides by two twice is shown in Figure 2b. This circuit permits a two MHz clock to be selected for use with a two MHz 6502.

If some I/O device requires its own crystal of a specific frequency, it may be possible to get a suitable CPU clock from it. For example, a clock of about .9 MHz can be obtained by dividing the 3.58 MHz color-burst frequency by four. Use of a clock frequency that gives other than one-microsecond clocks makes counting time less easy.

The only control output signals required by memory devices will be $\phi 2$ and R/W. These are applied directly to family devices, and can be combined (in the proper plolarity) to provide read/write signals for non-family I/O. This will be illustrated later. The READY and interrupt inputs should be pulled up, whether used or not, with 3.3K resistors.

Most controllers are designed to be automatically reset when power is applied. The circuit of Fig. 3 will provide a power-on reset and also permits a manual reset capability.

RAM

Some RAM will be necessary for stack. There are many choices. The Motorola 68A10 contains 128 bytes for less than $3. The 6532 contains 128 bytes of RAM, two programmable ports, and a timer ($10). Since the timer has no free-running mode, the 6522 is probably a better choice for a port chip, and it has two timers. A pair of 2114L 1K x 4 chips ($6) will provide 1K of memory and is probably the safest bet.

Figure 4 shows the two basic ways of controlling the chip select and write enable pin of a 2114L. In 4a., the chip select is decoded from addresses only. The write enable gets a write strobe fabricated from R/W and $\phi 2$. In 4b., the write enable pin gets the R/W direction signal. This means that write strobing action must be applied via the chip select input. This is accomplished by gating an ordinary address select with $\phi 2$ as shown. The gating may be omitted, if the address select is generated using $\phi 2$ as a component. The scheme of 4b. should be used if the 6502's data bus is buffered in order to avoid bus contention during $\phi 1$ of a write operation.

In the next installment, I'll talk about I/O choices.

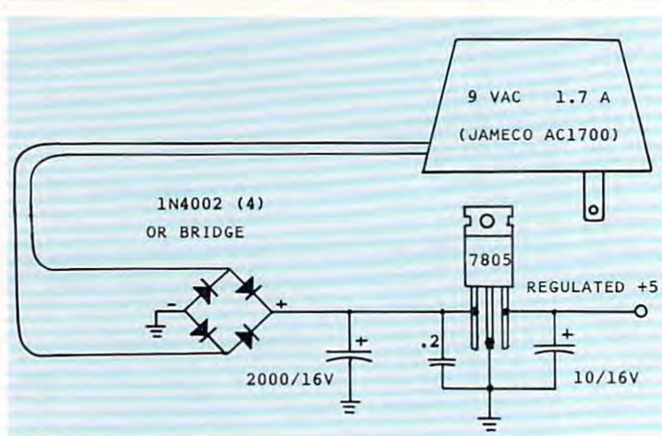
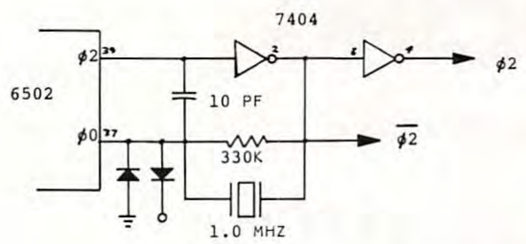
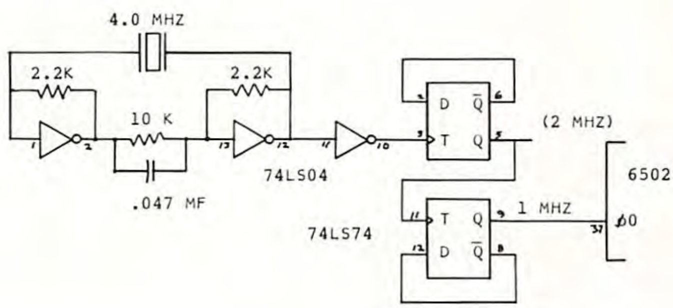


Figure 1. Regulated 5-volt supply using wall transformer.



a. Series Crystal Configuration.



b. Applied TTL Square Wave

Figure 2. Clock Generation for 6502.

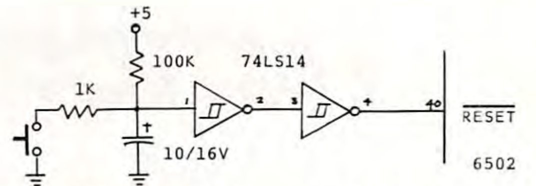
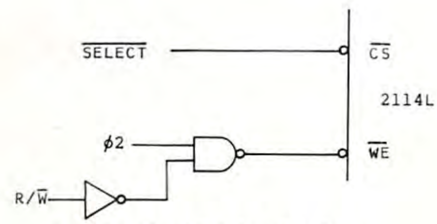
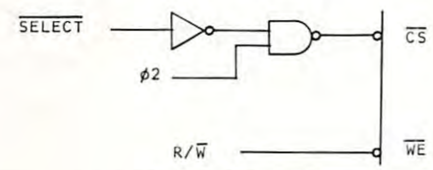


Figure 3. Manual and Power-on RESET for 6502.



a. Strobed Write Enable



b. Strobed Select

Figure 4. Control Signals for 2114L Static RAM.

©

Review:

DOS/65 – A Disk Operating System

(6502 Software)

Harvey B. Herman
Associate Editor

Ever since I installed an 8" disk on my KIM system (see **COMPUTE!** #11) I have been using KMMM disk operating software (DOS), sold by Willi Kusche. Recently, I saw an ad for a general purpose 6502 DOS, sold by Richard Leary.

From the beginning, Richard handles your order in a systematic and professional manner. His software can be run on a variety of machines, so you must give him, for example, the type of disk controller (Versafloppy in mine) and the location of the console I/O routine (my system uses standard KIM with echo defeated).

The package I promptly received in the mail contained an extensive instruction manual, a loader on cassette tape, and a diskette recorded with the operating system, an assembler/text editor and several utility programs. The utility programs include (among others): disktest, copy, debug, and format (not as yet for the Versafloppy, however). Users will appreciate that he has included the source code of many of the utility programs so they can be easily modified or enhanced if desired.

I had some trouble getting the system up for the first time as my read head is not aligned properly with track 0. Most people will not have this problem. The procedure required to bring the software up is trivial. First, load the tape. Second, run from location $200 (for the KIM system). However, using DOS/65 properly is a little more difficult than getting the first prompt ('A›'). It definitely helps to be familiar with CP/M, as Richard has emulated most of its functions. I found it much easier to read his manual after I had done some reading about CP/M. (So that's what "A›" means!)

An important advantage of DOS/65 is that it normally comes with an integral text editor and assembler. As a test, I typed in a program which transfers a table of data from one memory location to another. The editor features a variety of commands which allow easy correction of errors. It is even possible to edit programs which are larger than available memory. I had no trouble with any of the commands and in a short time I had created a file suitable for input to the assembler.

Running the two-pass assembler is a snap. The program just grinds away on its own and, in no time, two output files are produced. The first, a printer file, can be listed at some later time if desired. The second, a so-called "KIM" file, can be made into an executable file by a standard DOS command or loaded with the utility debugger program. I tested the later method and confirmed (by the debugger's disassembler) that the program was indeed in memory. I was surprised at how smoothly everything went once I got past some mental hangups.

DOS/65 appears to me to be a quite sophisticated program. It is complicated enough that, I believe, it should only be considered by serious assembly language programmers. It can be used with Microsoft BASIC, but the user may have to develop his own software interface. I confess this was beyond me. However, I feel certain Richard will offer his assistance if others are interested. The program does have a few awkward areas. For example, the user must specify, in decimal, the number of pages to save. However, some of that may be due to emulation of CP/M. An experienced user should have no problems with this software.

Richard A. Leary
1363 Nathan Hale Drive
Phoenixville, PA 19460 *$100*

©

LO•BALL COMPUTERS LO•BALL COMPUTE

New Products

Home Accounting System For The Atari

Sunnyvale, CA — A new home accounting system for the ATARI® 800™ Home Computer will replace the ATARI Accountant™, a small business accounting system, which will not now be offered for sale.

The new system will be lower-priced and easier to use than the current system, since it will be designed expressly for the home market. It will be ready for delivery in the first calendar quarter of 1982. The ATARI 815™ Dual Disk Drive, which was required for use of the professional accounting package, will not be offered for sale.

"The major thrust of our marketing efforts is toward use of our products in the home. We feel that our resources are better spent developing products aimed specifically at this market and segments that support this market, such as institutional education. It makes the most sense to convert the existing accounting package for home use," Roger H. Badertscher, president of Atari's Computer Division, said.

The new system will work with the ATARI 810™ Disk Drive, which is priced at $599.95. The dual disk drive was priced at $1,499.95, and offered "more capacity and a higher price than we feel is necessary for most home applications," Badertscher added.

No further details of the new accounting system are available at this time, except that it will include those features of the previous system that are best for the home market.

Business Package For Accounting And Tax Planning

Dakin5 Corporation's newest business application software package, The Depreciation Planner™ has been released to coincide with recent federal tax changes.

The Depreciation Planner is for use on the Apple microcomputer. It is designed to keep track of depreciable assets for accounting and tax planning purposes. This comprehensive package incorporates both the previous depreciation methods (to be used for assets purchased before January 1981), as well as the new depreciation methods (to be used for assets purchased after January 1981). It is faster than manual record keeping. It reduces chances of error and alleviates excess paperwork.

The user determines cost, salvage amount, useful life, and special restrictions or conditions pertaining to the asset and depreciation method. Once these figures are entered, The Depreciation Planner will automatically keep track of each asset.

The Depreciation Planner provides the following features:

— Automatically calculates current month depreciation, year-to-date and life-to-date amounts.

— Provides a printed list of assets in five formats.

Tag numbers
Location
Depreciation type
Depreciation Method
Asset Life

— Prints a list of all items that are potentially eligible for investment tax credit in each fiscal year.

— Prints a depreciation projection report that lists the projected current year and next year depreciation for each asset.

— Has a unique modeling feature that allows the user to experiment with any or all forms of depreciation for each asset. The information can be printed to allow thorough evaluation and comparison before a final selection is made.

The Depreciation Planner can work independently or will interface with The Controller® or The Business Bookkeeping System™. If used with either The Controller or The Business Bookkeeping System, The Depreciation Planner will automatically post depreciation amounts to the General Ledger and update current month, year-to-date and life-to-date amounts for each asset.

The documentation includes a complete glossary and modeling workbook to introduce users to the concept of depreciation and to give them hands-on experience with the software package.

The Depreciation Planner will provide significant benefits not only to businesses, but also to accountants. Accountants can use The Depreciation Planner to record all assets and depreciation information for their individual clients.

Owners of The Depreciation Planner may take advantage of Dakin5 Corporation's toll-free customer service line. This service, which is available to users of all Dakin5 products, is designed to answer user as well as dealer questions.

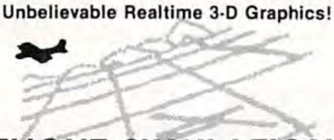
For further information contact Dakin5 Corp., 7475 Dakin St., 4th fl., Denver, CO 80221. (303) 426-6090.

The Controller is a registered trademark

THE PROGRAM STORE

Our Most Popular

Unbelievable Realtime 3-D Graphics!



FLIGHT SIMULATION

From Sub-Logic
A graphic tour de force that will truly captivate you. If you haven't seen this brilliant program, you haven't fully utilized the capabilities of your computer! During FLIGHT SIMULATION, you instantly select instrument flight, radar, or a breathtaking pilot's-eye-view. But be sure to strap yourself in -- you're liable to get dizzy!

Once you put in some air time learning to fly your computer head for enemy territory and try to bomb the fuel depot while fighting off five enemy warplanes. Good Luck!

Apple II 16K tape....$25.00
Apple II 32K disk....$33.50
TRS-80 16K tape......$25.00

MICROSOFT ADVENTURE

Prepare yourself for the adventure of your life as this classic fantasy/logic game takes you into the world of the Colossal Cave. Your computer is your guide as you search for treasures, solve puzzles, explore, and avoid the dangers that lurk within.

Complete version of the original Adventure, originally written for the DEC PDP-10 in FORTRAN. The program has been translated to bring you all the enjoyment in your home computer.

TRS-80 (32K disk), Apple (32K disk) ~~$29.95~~ Special

SPECIAL SAVE $10: $19.95 thru 12/31!

DEATH-MAZE 5000

From Med Systems
A new breed of adventuring! Venture through a graphically represented 3-D maze, with halls that could dead end -- or recede to infinity. Step through the doors or drop into the pits. Will you encounter monsters and mayhem, or will you be treated to useful objects and information? Will you ever get out alive?

TRS-80 (16K tape), Apple (32K tape) $14.95
TRS-80 (32K disk) $19.95

Also Available: ASYLUM for TRS-80 (16K tape) $14.95

TRS-80® · APPLE® · ATARI®

The Program Store carries more than 1000 programs for TRS-80, Apple, and Atari. If the program you want is not listed here, call us: we probably have it in stock!

NEW

VOYAGE TO VALKYRE

By Leo Christopherson from AOS
Combine the animation and music techniques pioneered by Christopherson with the challenge of his first fast-moving arcade game and you have VOYAGE TO VALKYRE!

You speed through a magical maze guarded by ferocious birds that swoop down to attack if you don't get them first. To list all the play and options of this exciting game would take the 16 pages of instruction included.

Tape: TRS-80 (16K) $34.95
Disk: TRS-80 (16K), Apple (48K) $39.95

LUNAR LANDERS

From Adventure & Stoneware
These are realtime action simulations of a descent to the Moon. In common, they all have super graphics, realistic movement and control, and sound. The skillful are rewarded with high scores, the clumsy can look forward to spectacular crashes!

LUNAR LANDER: TRS-80 & Atari (16K), $19.95
TRANQUILITY BASE: Apple (32K disk), $24.95

FILE MANAGER 800

From Synapse Software
An extremely powerful and versatile database manager for use in both professional and personal applications. You define the format of the records to be filed and FILE MANAGER 800 gives you full control over sorting, searches, and retrieval.

You can store up to 1000 records on each disk and the program will allow up to four drives. Access to any record takes less than 1.5 seconds, and most commands can be entered with self-prompting single keystrokes.

Atari (40K disk)...$99.95

NEW

Crush, Crumble and Chomp!

From Epyx
It's a monster movie, and you are the monster! You can be The Glob, Kraken, Mantra, Mechismo, Arachnis, or Goshilla -- or even design your own "custom" monster (disk version only). This hilarious action game is loaded with graphics and sound as you practice your villany. With 6 monsters, 4 cities, and 5 game objectives, you get a choice of more than 100 possible scenarios. A monster's life is not all carnivorous crunching, though: The combined resources of the police, science, and armed forces are bent on your destruction.

TRS-80 (16K tape or 32K disk), Apple (48K disk)...$29.95
Coming soon for Atari.

GALACTIC SAGA

By Douglas Carlston from Broderbund/A.I.
Take control of the Galactica as you navigate through an uncharted 3-D universe. In GALACTIC EMPIRE, you attempt to unify a star system that is randomly created each time you play. TRS-80 (16K tape) $14.95, Apple (48K disk) $24.95, Atari (32K tape) $19.95.

GALACTIC TRADER pits your bartering skills against those of the other inhabitants as you try to accumulate riches and power. But watch out for the assassins and the energy cartel -- they're out to getcha! TRS-80 (16K tape) $14.95, Apple (48K disk) 24.95.

Diplomacy and deviousness play equal parts in GALACTIC REVOLUTION. It's a game that combines tactics, social manipulation, and Machiavellian ruthlessnes. For more intrigue, this game allows more than one player. Sound effects. TRS-80 (16K tape) $14.95, Apple (48K disk) 24.95.

TAWALA'S LAST REDOUBT makes you the rebel leader. You must intercept and decipher the messages of the dethroned Emporer Tawala in order to initiate an assault on his exile kingdom. Apple (48K disk) $24.95. Coming soon for TRS-80.

Control The Whole Universe -- Get All Three!

Visit our other stores: Seven Corners Center · Falls Church, VA & W. Bell Plaza · 6600 Security Blvd. · Baltimore, MD

TO ORDER CALL TOLL FREE 800 424-2738 For information Call (202) 363-9797

THE PROGRAM STORE
4200 Wisconsin Avenue NW, Dept. U-II Box 9609
Washington, D.C. 20016

MAIL ORDERS: Send check or M.O. for total purchase price, plus $1.00 postage & handling. D.C. residents, add 6% tax. Charge card customers: include all embossed information on card.

Get the most from your micro with software and accessories from one of the world's largest selections.

The Program Store

CALL TOLL FREE 800 424-2738

EPYX COMPUTER GAMES THINKERS PLAY

TEMPLE OF APSHAI
The first of the DunjonQuest series, and still one of the most popular. In exploring over 200 rooms in the magical labyrinth, you will encounter more than 30 kinds of fearsome monsters guarding over 70 treasures. Some of the treasures will help you in your quest, but you must still watch out for the many monsters and traps that spring out from the walls and shadows.
Apple (48K disk) TRS-80 (16K tape, 32K disk) $39.95

LOWER REACHES OF APSHAI: Four more levels. Requires above program. $19.95

HELLFIRE WARRIOR
Sequel to APSHAI, this thriller gives you four levels to explore with sixty rooms each. New monsters, new traps, new challenge!
Apple (48K disk), TRS-80 (16K tape, 32K disk) $39.95

KEYS OF ACHEREON: Four more levels. Requires above program. $19.95

MORLOC'S TOWER
A completely different 6-level "dunjon" hides treasures and trouble. Defeating the dreaded Morloc takes wit and determination.
Apple (32K tape, 48K disk), TRS-80 (16K tape, 32K disk), Atari (32K tape or disk) $19.95

DATESTONES OF RYN
Rex the Reaver and his band of 24 cutthroats have made off with the treasured datestones. They're hiding in an labyrinth of caves in the Haunted Mountains. It is your job to find them before they find you.
Apple (32K tape, 48K disk), TRS-80 (16K tape, 32K disk), Atari (32K tape or disk) $19.95

TUESDAY MORNING QUARTERBACK
An amazingly realistic football game that lets you captain the team of your choice against the team you choose for the computer. 96 possible offensive/defensive combinations are available. A database includes the teams' rosters as well as their strengths and weaknesses -- just like the real thing!
Apple (48K disk), TRS-80 (48K disk) $29.95

STARQUEST

RESCUE AT RIGEL
First of the "Starquest" series, it's an interplanetary adventure. The evil High Tollah has captured ten prisoners; if you don't find and save them they will be transformed into enemy aliens -- and set out after you!
Apple (32K tape, 48K disk), TRS-80 (16K tape, 32K disk), Atari (32K tape or disk) $19.95

STAR WARRIOR
You're on your own, light years from Earth in intergalactic space. If you thought saving the princess in RESCUE AT RIGEL was a challenge, wait 'til you pit yourself against STAR WARRIOR! Order the second edition of the Starquest Series now.
Apple (32K tape, 48K disk), TRS-80 (16K tape, 32K disk), Atari (32K tape or disk) $19.95

MICROCOMPUTER GAMES

From Avalon Hill
Fine war and strategy games for the home computer that pit your skill against the program. 16K tape unless otherwise noted.

MIDWAY CAMPAIGN - relive the battle as you control our naval forces. TRS-80, Apple, Atari (32K) $14.95

B-1 NUCLEAR BOMBER - avoid MiGs & missiles as you pilot this advanced aircraft. TRS-80, Apple, Atari (16K) $14.95

NORTH ATLANTIC CONVOY RAIDER - lets you simulate the Bismark convoy raid of 1941. TRS-80, Apple, Atari (16K) $14.95

NUKE WAR - choose espionage or arms buildup to control a nuclear confrontation. TRS-80, Apple, Atari (16K) $14.95

PLANET MINERS - one to four players compete, staking claims in the solar system. TRS-80, Apple, Atari (24K) $14.95

CONFLICT 2500 - star battle in the 26th century for up to 10 players. TRS-80, Apple, Atari (32K) $14.95

LORDS OF KARMA - adventure in a land where good deeds score more than treachery. TRS-80, Apple, Atari (40K) $20

TANKTICS - combines gameboard & computer in a strategic WWII tank battle. TRS-80, Apple, Atari (24K) $23.95

MAJOR LEAGUE BASEBALL - play solitaire or head-to-head, using names & statistics of real players. Apple (48K disk) only, $24.95

THE PROGRAM STORE
Franchises Available

SCOTT ADAMS'

ADVENTURE

By Scott Adams from Adventure International
Twelve different adventures make up this acclaimed series. Written in machine language for fast response, they support lower case (if installed) and have over 100 words in their vocabularies.

Until you've played an Adventure, you can't appreciate the hours of challenge and fun built into each program. Each tests your powers of reason and deduction as you attempt to accomplish your mission using the implements you have, find or devise. Tape for TRS-80 16K, Apple 24K, Atari 24K (Adventures 1-9 only for Atari).

1. ADVENTURELAND - caves, pits, magic words and the dragon... $19.95
2. PIRATE'S ADVENTURE - go from your London flat to Treasure Island... $19.95
3. MISSION IMPOSSIBLE - complete your mission or the reactor is doomed... $19.95
4. VOODOO CASTLE - Save Count Cristo from his fiendish curse... $19.95
5. THE COUNT - when you awaken in Transylvania, watch your neck... $19.95
6. STRANGE ODYSSEY - find alien treasures at the edge of the galaxy... $19.95
7. MYSTERY FUN HOUSE - try to leave the most bizarre funhouse you've seen... $19.95
8. PYRAMID OF DOOM - search the labyrinth for ancient treasures... $19.95
9. GHOST TOWN - just because something moves doesn't mean it's not dead... $19.95
10. SAVAGE ISLAND #1 - first part of a new series for advanced adventurers...$19.95
11. SAVAGE ISLAND #2 - part two: requires the purchase of part one (above)...$19.95
12. GOLDEN VOYAGE - you have just 3 days to rejuvenate the near-dead king...$19.95

Three Adventures on disk (TRS-80 32K, Apple 48K), choose:
1-3 4-6 7-9 10-12 $39.95 per disk

SARGON II

By Dan & Kathe Spraklen from Hayden
Acclaimed the best of the microcomputer chess programs. SARGON II came in third in the 9th North American Computer Chess Championship, playing against much bigger machines! You haven't really played chess against your computer until you've tried this brilliant program.

Tape: TRS-80 (16K), Apple (24K) $29.95
Disk: TRS-80 (32K), Apple (24K) $34.95

Prices Subject to Change

THE PROGRAM STORE · Dept UH Box 9609 · 4200 Wisconsin Ave, NW · Washington, D.C. 20016

Item ____ Price ____ Postage $1.00 Total ____ name ____ addr ____
☐ CHECK ☐ VISA city ____ state __ zip ____
☐ MASTERCARD MC Bank # ____ Card # ____ Exp ____

of Dakin5 Corporation. The Depreciation Planner and The Business Bookkeeping System are trademarks of Dakin5 Corporation. The Controller is marketed by Apple Computer, Inc. Apple is a registered trademark of Apple Computer, Inc.

80 Column Adaptor

EXECOM CORP. announced a new product for Commodore Business Machines' PET/CBM series computers. The product is an 80 column adaptor, circuit board and ROM combination, that allows the user to switch between the original, 40 column display, and the new 80 column display, from the keyboard, or through program control. Price of this modification is $275.00 plus installation. This circuit board and ROM combination is for the PET/CBM computers that do not have a CRT, or display controller chip, in the 2000/3000/4000 series models, which are designed for version 3.0, or 4.0 Basic. The installation involves cutting circuit traces and installing 4 sockets, thus making it a product that should be purchased from a dealer, if you are not technically oriented. EXECOM CORP. offers the installation for $75.00. This requires the user to send in their computer circuit board for modification.

For further information contact Execom Corp., 1901 Polaris Ave., Racine, WI 53404.

Information Systems Education To Be DPMAEF Conference Theme

Chicago: The Education Foundation (EF) of the Data Processing Management Association (DPMA) has announced that it will sponsor a National Conference on Information Systems Education to be held March 22-24, 1982 in Chicago. The meeting will consist of one-day in-depth Workshops followed by a two-day general Conference.

On the first day, invited experts will conduct special Workshops in major areas of importance in Information Systems Education. The Conference, which begins on the second day, will deal with issues of practical importance to the providers, recipients, and end users of Information Systems Education. It will provide comprehensive and authoritative updates on the DPMA Model Curriculum, reports on practical experiences in implementing Information Systems Education, and approaches to overcome financial and other obstacles to implementing such programs.

The DPMAEF was established in 1975 by the Association for the purpose of expanding educational opportunities for systems professionals and to conduct research and programs of benefit to the DP industry,

commodore spectacular

8032-32K 80 COL CRT	
REG. $1495	**$1095**
4032-32K 40 COL CRT	
REG. $1295	**$995**
4016-16K 40 COL CRT	
REG. $995	**$795**
8050-DUAL DISK 950K	
REG. $1795	**$1345**
4040-DUAL DISK 343K	
REG. $1295	**$995**
C2N-CASSETTE DRIVE	
REG. $95	**$69**
4022-80 COL PRINTER	
REG. $795	**$629**
8024-MANNSMAN TALLEY	
REG. $1995	**$1595**
25CPS-STARWRITER	
REG. $1895	**$1445**
CBM-IEEE MODEM	
REG. $279	**$229**
VOICE SYNTHESIZER	
REG. $395	**$329**
VIC 20 ★★★★	
REG. $299	**$269**
PET TO IEEE CABLE	
REG. $39.95	**$34**
IEEE TO IEEE CABLE	
REG. $49.95	**$39**

SOFTWARE

	LIST	YOU PAY
OZZ—The Information Wizard	$395	$299
Wordcraft 80	$395	$299
IRMA—Info Retrieval & Mgmt. Aid	$495	$399
Dow Jones Portfolio Mgmt.	$149	$119
Pascal Development Pkg.	$295	$229
EBS—Receivables, Inventory	$750	$579
BPI—General Ledger	$395	$299
Word Pro 3—40 Column	$250	$179
Word Pro 4—80 Column	$375	$269
Word Pro 4 Plus	$450	$329

CUSTOM PRINTED CONTINUOUS MINI-FORMS

- LOOK PROFESSIONAL
- LOW PRICES FOR SMALL RUNS

1000-11"x9½" BLACK

PARTS	LETTERHEAD	INVOICE	STATEMENT
1	37.50	—	37.50
2		63.50	63.50
3		79.50	
5		115.50	

- SPECIAL LOGOS, 2 COLOR PRINTING & SPECIAL STOCK AVAILABLE

MINI-FORMS PRICES & ORDERS:
(800) 523-3442
PA & West of Rockies (215) 455-9705

Main Line Computer Discount

P.O. BOX 170
ST. DAVIDS, PA 19087

MAIL ORDER PHONE
800-345-8112
IN PA CALL 1-800-662-2444

- PREPAID ORDERS SHIPPED FREE
- VISA & MASTERCARD ADD 3%
- COD-UPS
- PA RESIDENTS ADD 6% PA TAX

educators, business, government, and the public.

DPMAEF has issued a "Call for Papers" for this Conference. For a copy of the "Call" or for further information on the Conference, contact the Conference Manager, USPDI, 12611 Davan Drive, Silver Spring, MD 20904. Telephone (301)622-0066.

Low Cost Floppy Disk For Rockwell AIM

The PEDISK II Floppy Disk System from CGRS Microtech, Inc. is now available for the Rockwell AIM microcomputer. This system, originally developed for the Commodore PET, now provides the AIM owner with a high performance mass storage peripheral. It is available with either 5¼" or 8" drives with a storage capacity to 858 Kbytes in a three drive system. The 8" drive offers standard IBM 3740 compatibility allowing data exchange with most microcomputers and minicomputers.

A new software package, ADOS, provides a full set of disk utilities including format, copy, display, patch and directory. Additional disk commands allow the user to load and save programs, data files and source files. A convenient user interface is provided to utilize the disk system in custom software. Several software tools are available for use with the AIM/PEDISK. These include full FORTH+, a versatile high level language, a Macroassembler/Editor for machine language programs and a full BASIC interface to allow the disk to be used with AIM BASIC.

The single drive dual density Model 540-1 offers 143 Kbytes of storage and sells for $595.00. The dual drive quad density Model 580-2 offers 572 Kbytes of storage and sells for $1195.00. For additional information, contact your dealer or CGRS Microtech, P.O. Box 102, Langhorne, PA 19047. 215-757-0284.

Hi-Res Graphics For Atari 400/800 Computers

Newbury Park, CA — Versa Computing, Inc. announces the release of *GRAPHICS COMPOSER*, the complete joystick/paddle graphics software package for ATARI 400/800 Computers.

With GRAPHICS COMPOSER, you use paddles or joystick to draw a picture outline on hi-res screen Mode 8 or 7. Then use color fill-in, color brushes and add Text to complete your graphic designs. *Save your graphics to disk or cassette.*

GRAPHICS COMPOSER allows easy creation of PLAYER/MISSILE shapes which may be used in other programs. The

we beat the price...

Exclusive One Year Warranty Included

800™ $779

WITH ONE YEAR EXTENDED WARRANTY

ATARI®
Computers for people.™

ATARI 810 DISC DRIVE

$444

Available without warranty for even less.

ATARI SOFTWARE

CX401 General Accounting	$399.00
CX402 Accounts Receivable	$399.00
CX403 Inventory Control	$399.00
CX404 Word Processor	$119.00
CX405 PILOT	$68.00
CX413 MICROSOFT BASIC	$68.00
CX4101 Invitation to Programming 1	$17.00
CX4102 Kingdom	$13.00
CX4103 Statistics	$17.00
CX4104 Mailing List	$17.00
CX4105 Blackjack	$13.00
CX4106 Invitation to Programming 2	$20.00
CX4107 Biorhythm	$13.00
CX4108 Hangman	$13.00
CX4109 Graph It	$17.00
CX4110 Touch Typing	$20.00
CX4111 Space Invaders	$17.00
CX4112 States & Capitals	$13.00
CX4114 European Countries & Capitals	$13.00
CX4115 Mortgage & Loan Analysis	$13.00
CX4116 Personal Fitness Prog.	$59.00
CX4117 Invitation to Programming 3	$20.00
CX4118 Conversational French	$45.00
CX4119 Conversational German	$45.00
CX4120 Conversational Spanish	$45.00
CX4121 Energy Czar	$13.00
CX4125 Conversational Italian	$45.00
CX6001 U.S. History	$23.00
CX6002 U.S. Government	$23.00
CX6003 Supervisory Skills	$23.00
CX6004 World History	$23.00
CX6005 Basic Sociology	$23.00
CX6006 Counseling Proced.	$23.00
CX6007 Principal of Act.	$23.00
CX6008 Physics	$23.00
CX6009 Great Classics	$23.00
CX6010 Business Comm.	$23.00
CX6011 Basic Psychology	$23.00
CX6012 Effective Writing	$23.00
CX6014 Principals of Econ.	$23.00
CX6015 Spelling	$23.00
CX6016 Basic Electricity	$23.00
CX6017 Basic Algebra	$23.00
CX8106 Bond Analysis	$20.00
CX8107 Stock Analysis	$20.00
CX8108 Stock Charting	$20.00
CXL4001 Education System Master	$21.00
CXL4002 Basic Computing Language	$46.00
CXL4003 Assembler Editor	$46.00
CXL4004 Basketball	$24.00
CXL4005 Video Easel	$24.00
CXL4006 Super Breakout	$30.00
CXL4007 Music Composer	$45.00
CXL4009 Chess	$30.00
CXL4010 3-D Tic-Tac-Tow	$24.00
CXL4011 Star Raiders	**$32.00**
CXL4012 Missle Command	**$32.00**
CXL4013 Asteroids	**$32.00**
CXL4015 TeleLink	$20.00
Compuhome	$74.95
Visicalc	$149.00
Letter Perfect (Word Processor)	$119.00
Source	$89.00

Atari® Peripherals:

400 16K	$329.00
410 Recorder	$59.00
822 Printer	$359.00
825 Printer	$ CALL
830 Modem	$159.00
850 Interface	$ CALL

Atari® Accessories

New DOS 2 System	$21.00
CX70 Light Pen	$64.00
CX30 Paddle	$18.00
CX40 Joy Stick	$18.00
CX853 16K RAM	$89.00
Microtek 16K RAM	$75.00
Microtek 32K RAM	$169.00

commodore

CBM 8032 $1099

4016	$799.00
4032	$999.99
8096	$1795.00
CBM4022 Printer	$629.00
Tally 8024	$1699.00
CBM C2N Cassette Drive	$69.00
CBM4040 Dual Disk Drive	$999.00
CBM8050 Dual Disk Drive	$1349.00

VIC 20 $259

Vic-TV Modual	$19.00
Vic Cassette	$69.00
Vic Disk Drive	$ Call
Vic 6 Pack program	$44.00

CBM Software

WordPro3 Plus	$199.00
WordPro4 Plus	$299.00
Commodore Tax Package	$399.00
Visicalc	$149.00
EBS Accts. Rec./Inventory Interactive Syst.	$595.00
BPI General Ledger	$329.00
OZZ Information System	$329.00
Dow Jones Portfolio	$129.00
Pascal	$239.00
Legal Time Accounting	$449.00
World Craft 80	$289.00
Word Check	$180.00
Create-A-Base	$219.00
Power	$89.00
Socket-2-Me	$20.00
Jinsam	Call

Disks

CX8100 Blank Disk (5)	$22.00
Sycom Blank Disk (10)	$29.00
Maxell Blank Disk (10)	$36.00
Maxell Blank Disk (10)	$46.00

Printers

Epson MX-70	Call for Prices
Epson MX-80	Call for Prices
Epson MX-80 FT	Call for Prices
Diablo 630	Call for Prices
Nec 5530	$2495.00
TEC 1500 Starwriter 25cps	$1495.00
TEC 1500 Starwriter 45cps	$1795.00

No Risk, No Deposit On Phone Orders, COD or Credit Card, Shipped Same Day You Call *

* on all in stock units

IN PA, CALL (717) 327-9575

(800) 233-8950

To Order:

Phone orders invited (800 number is for order desk only). Or send check or money order and receive free shipping. Pennsylvania residents add 6% sales tax. Add 3% for Visa or M.C. Equipment is subject to price change and availability without notice. Please call between 11 AM & 6 PM.

VISA

COMPUTER MAIL ORDER
501 E. 3RD ST., WILLIAMSPORT, PA 17701

GEOMETRIC FIGURES program lets the user define circles, triangles, polygons, parallelograms, and even trigonometric curves!

Loading routines are provided so that pictures can be used in other programs or traded with friends.

GRAPHICS COMPOSER is available on cassette or disk for only $39.95, and requires 32K RAM. Call Versa Computing at (805) 498-1956, or ask your local computer retailer for GRAPHICS COMPOSER.

Hayden Announces DENTISTAID™

DENTISTAID, the lowest priced dental office management program on the market is designed to streamline all major time-consuming tasks performed in the dental office.

This menu-oriented program will automatically print standard ADA insurance forms, prequalification and actual services, monthly statements, patient recall notices, accounts receivable aging reports, daily summary of work performed and payments received, daily, weekly, monthly, and yearly totals, production analysis using standard ADA insurance forms, prequalification and actual services, monthly statements, patient recall notices, accounts receivable aging reports, daily summary of work performed and payments received, daily, weekly, monthly, and yearly totals, production analysis using standard ADA categories and display of individual accounts.

One of the primary advantages of DENTISTAID is the simplicity of operation used to increase office efficiency through automatic generation of many forms and reports that give the dentist better practice control.

With many systems installed in the past two years, this error-free, operator-proof program can be installed in a dental office in two days.

DENTISTAID is a trademark of the Hayden Book Company, Inc., located at 50 Essex Street, Rochelle Park, NJ 07662.

New Books

Experiments in Artificial Intelligence for Small Computers,
by John Krutch
Howard W. Sams & Co.
112 pages, $8.95 softbound

This new book is a resource for anyone wanting to conduct interesting and exciting experiments on the small computer.

The author presents programs written in Microsoft's Level II BASIC that can be easily converted to most BASIC dialects.

The book begins with an explanation of artificial intelligence — its scope and problem areas. A short BASIC program that involves moving a chess king on a small chessboard is used to illustrate the discussion. Game-playing programs, with checkers as a major topic, are presented in the second chapter.

Problem-solving is covered with emphasis on a program that predicts a human player's choices. The author then focuses on reasoning, primarily by means of a program which stores data and makes deductions from this data. Also, creativity is treated in the form of computer poetry and computer-generated prose.

The book devotes a chapter to natural-language processing or verbal communication. It is illustrated by the program DOCTOR, which simulates a psychiatrist's counsel.

An appendix describes BASIC keywords to help make the book's programs more easily translated to the version of BASIC to which the user is accustomed.

Your Own Computer (2nd ed.)
by Mitchell Waite & Michael Pardee
Howard W. Sams & Co.
224 pages, $7.95 softbound

In this book, authors Mitchell Waite and Michael Pardee have removed much of the complexity and mystery that surrounds the microcomputer and have succeeded in producing a simple, easy-to-understand book about these amazing devices.

This new edition provides the newcomer with the knowledge and confidence needed to use today's personal computer. The text has been updated and explained to include a chapter that objectively compares 30 popular personal and small business computers now on the market.

What are personal computers? Where did they come from? What's coming in 1990? An introductory chapter provides these answers for you.

Personal computing applications for your home, office, or classroom are explained. The chapter on programming has been expanded in this edition to help you understand the heavy emphasis on computer software.

To help you better understand computer concepts, a glossary of the most often encountered buzz words, complete with definitions and a practice sentence using the word, is presented in Chapter Two. Also included is a list of key acronyms.

The book presents the nuts and bolts — from input/output units and peripherals to buses and memories — that make up the personal computer.

The authors give tips on how you can get started in computing for your home, business, or hobby. An appendix on computer number systems is included for your reference.

Webster's Microcomputer Buyer's Guide
by Tony Webster
Hayden Book Co.
326 pages, $25.00 peperback

This is a detailed reference guide listing approximately 113 private vendors for microcomputer users.

The book's 16 chapters contain four parts: Theory and Application, Independent Software Vendors, Microcomputers and Microcomputer Systems and CRT Displays, Printers and Printing Terminals.

Part I introduces ten chapters, the first five concerning theory and application. Also, outlined are the differences between microprocessors, microcomputers and microcomputer systems. The remaining five chapters examine the future potential of microcomputers, provide guidelines for selecting microcomputers and the capabilities of microcomputers in word

Lyco Computer Marketing & Consultants

We Specialize in Quality, Knowledge, Service, and Microcomputers

717 - 435-5197

ATARI HARDWARE

800	$744.00
400	$329.00
410 CASSETTE RECORDER	$57.00
810 DISK DRIVE	$444.00
850 INTERFACE	$159.00
825 PRINTER	$525.00
830 PHONE MODEM	$135.00

PROGRAMMING SOFTWARE

CX4101 INVITATION TO PROG.	$17.00
CX4106 INVITATION TO PROG. 2	$19.00
CX4117 INVITATION TO PROG. 3	$19.00
CXL4003 ASSEMBLER EDITOR	$42.00
CXL4002 ATARI BASIC CART	$42.00
CXL4015 TELELINK	$19.00

EDUCATIONAL SOFTWARE

CXL4001 EDUCATIONAL SYSTEM	$19.00
CX 6001 U. S. HISTORY	$22.00
CX 6002 U. S. GOVERNMENT	$22.00
CX 6003 SUPERVISORY SKILLS	$22.00
CX 6004 WORLD HISTORY	$22.00
CX 6005 BASIC SOCIOLOGY	$22.00
CX 6006 COUNSELING PROCED.	$22.00
CX 6007 PRINCIPLES OF ACCT.	$22.00
CX 6008 PHYSICS	$22.00
CX 6009 GREAT CLASSICS	$22.00
CX 6010 BUS. COMMUNICATION	$22.00
CX 6011 BASIC PSYCHOLOGY	$22.00
CX 6012 EFFECTIVE WRITING	$22.00
CX 6014 PRINCIPLES OF ECON.	$22.00
CX 6015 SPELLING	$22.00
CX 6016 BASIC ELECTRICITY	$22.00
CX 6017 BASIC ALGEBRA	$22.00
CX 4108 HANGMAN	$12.00
CX 4112 STATES & CAPITALS	$12.00
CX 4114 EUROPE COUNT. & CAP.	$12.00
CX 4121 ENERGY CZAR	$12.00
CX 4123 SCRAM	$12.00
CX 4102 KINGDOM	$12.00

ATARI

800 16K	$744.00
400 16K	$329.00

ENTERTAINMENT

CXL4004 BASKETBALL	$29.00
CXL4005 VIDEO EASEL	$29.00
CXL4006 SUPER BREAKOUT	$29.00
CXL4009 CHESS	$29.00
CXL4010 3-D TIC-TAC-TOE	$29.00
CXL 4011 STAR RAIDERS	$30.00
CXL 4012 MISSILE COMMAND	$30.00
CXL4013 ASTEROIDS	$30.00
CX 4015 BLACKJACK	$12.00
CX 4111 SPACE INVADERS	$16.00

ACCESSORIES

CX 30 PADDLES	$17.00
CX 40 JOYSTICKS	$17.00
CX853 16K RAM	$85.00
****** 32K RAM by MICROTEK	$169.00

WE CARRY MANY OTHER LINES OF MICROCOMPUTERS YOU CAN CALL FOR PRICES ON:

CPU's	PRINTERS
COMMODORE	TEC
ALTOS	DIABLO
TEXAS INSTRUMENT	HEC
ZENITH	EPSON
INERTEC DATA	ANADEX
AND OTHERS	AND OTHERS

At Lyco Computers we offer our expert services to help customers make their first computer purchase schools establish a computer program, or evaluate multiterminal systems.

CALL US AT 717 - 435-5197

to help evaluate your needs, or if you wish to make a purchase.

Lyco Computers
P.O. Box 10
COGAN STATION, PA 17728

Note the prices in this advertisement are subject to change processing and education.

Part II, covering Chapters 11 and 16, provides descriptions of many software operating systems, utilities, languages and application packages from independent vendors. Part III provides a summary of many major microcomputer systems available, under the headings, Overview, Central Processing Systems, Peripherals, Software, Pricing and Head Office. Part IV lists in detail the specifications of some of the popular CRT displays, printers and printing terminals available for connection to microcomputers.

Learning With Computers
by Alfred Bork
Digital Press
304 pages, $25.00 hardbound

Learning With Computers is a key publication for anyone interested in designing, using, or studying the computer as an educational tool. It focuses specifically on using the computer to build a more responsive learning environment.

An original collection of articles, *Learning With Computers* offers a comprehensive overview for using the computer as an interactive teaching and learning device. Written in a non-technical style, it covers such crucial topics as:

- Using programs prepared by others
- Personal computers versus timesharing systems
- Types of hardware and the efficient use of hardware
- How computer graphics can assist learning
- Authoring graphic dialogs
- Attitudes towards learning simulations
- On-line tests and quizzes
- Motivation for computers in the classroom
- Conversational language criteria
- Computer availability
- Self-pacing and student choice
- Course management

Instant (Freeze-Dried Computer Programming in) BASIC
by Jerald Brown
Dilithium Press
200 pages, $10.95

The 2nd Edition keeps the style and flavor of the 1st edition. BASIC, a computer programming language, is favored by most beginners because it is easy to learn and use, and because of the proliferation of programs written in BASIC. The author, Jerry Brown, appreciates who his audience is (the absolute beginner) and writes specifically for them.

The 2nd Astounding! Edition has more annotations, additions activities and text, and has been updated. In the tradition of the first edition:

- It is microcomputer oriented for Microsoft-like versions of BASIC as used on the Apple, TRS-80, PET, Microexpander, and any brand of computer using Microsoft BASIC 80, with annotations for Northstar BASIC, Atari BASIC, and DEC BASIC Plus.
- There is no heavy math.
- There are neat little boxed summaries of BASIC statements throughout the book.
- There are end-of chapter activities to see how well the reader is learning BASIC.
- It is full of graphics to relieve boredom and stimulate interest.

Instant BASIC is for every beginner, and is especially suited for the educational market.

File System For Floppy Disks Described In Brochure

Los Altos, CA — Amaray Corporation announces the immediate availability of a four-page, two-color brochure describing its DiskBank floppy disk filing and storage system.

Available for either 5¼" or 8" floppy disks, each DiskBank unit holds ten floppy disks in a dust-free, secure environment. The DiskBank's interlocking, modular design allows the user to structure the system horizontally or vertically with maximum space efficiency.

The DiskBank's design allows it to be used either on a desk top or in a file drawer application.

Contact Rich Koch at Amaray Corp., 2251 Grant Road, Suite H, Los Altos, CA 94022 (415) 968-2840.

ATARI

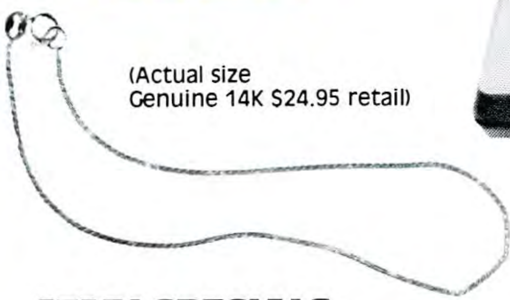
Ω CHRISTMAS SPECIALS!

ATARI

SAVE now on ATARI especially these products!!

We carry the complete line of ATARI Hardware, Software Peripherals, and accessories.

FREE GOLD!



(Actual size Genuine 14K $24.95 retail)

Buy an ATARI 800 16K and receive a 14K Gold Bracelet FREE!!

ATARI SPECIALS

ATARI 400 16K Personal Computer	$349

ATARI PERIPHERALS

410 Program Recorder	$ 60
810 Disk Drive	$449
820 40 Column Printer	$299
822 40 Column Thermal Printer	$349
825 80 Column Printer	$599
830 Acoustic Modem	$159
850 Interface Module	$139

ATARI ACCESSORIES

CX852 8K RAM Memory Module	$39.95
CX853 853 16K RAM Memory Module	$89.95
CX30-04 Paddle Controller (pain)	$15.00
CX40-04 Joystick Controller (pair)	$15.00

With any purchase of ATARI Hardware or Software over $500 you can buy the bracelet for only $9.95.

PRICES ARE SUBJECT TO CHANGE W/O NOTICE.

WE CARRY THE COMPLETE LINE OF ATARI SOFTWARE.

CALL FOR THE SUPER CHRISTMAS SAVINGS!!!

WEST COAST
1-800-235-3581
3533 Old Conejo Rd. #102
Newbury Park, CA 91320
1-805-499-3678
CA. TOLL FREE 1-800-322-1873

OMEGA SALES CO.

EAST COAST
1-800-556-7586
12 Meeting Street
Cumberland, RI 02864
1-401-722-1027

Low Power IEEE-488 Controller/Talker/ Listener Module Plus Custom I/O

Ontario, CA, September 29, 1981 — A General Purpose Interface Bus (GPIB) module which meets all IEEE-488 specifications, with control software, a wire-wrap section for custom input/output, operating on a single +5V DC power source, is now available from General Micro Systems, Inc.

The module, GMS6514, is fully compatible with the Motorola EXORciser/ Micro Module and Rockwell AIM 65 and SYSTEM 65 bus structures. It has an on-board device address switch and a special GPIB connector to allow bus control with minimum effort. Using low power Schottky technology, the module also offers over voltage and reverse polarity protection.

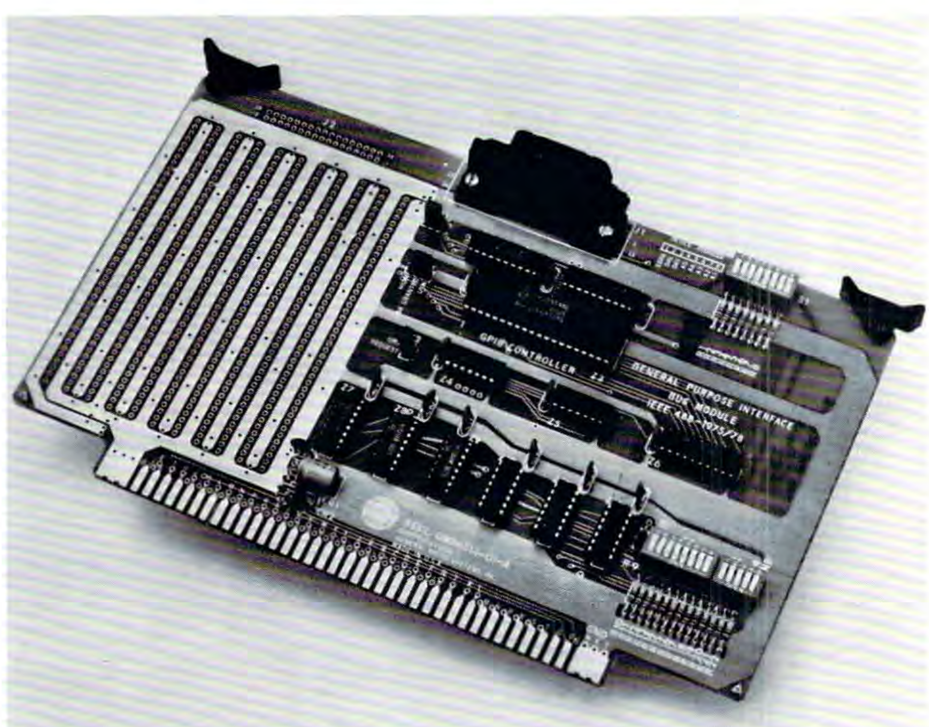
The GMS6514 is designed with a TMS9914 LSI bus controller device from Texas Instruments, which meets all specifications for signal levels and timing for IEEE-488, 1975/78 standards. It has pass control and system control capabilities with device clear and trigger functions, parallel and serial poll, service request and remote/local selection with local lockout.

The board has base address and enable/disable switches. A device address switch, with secondary addressing capability, is accessible from the top of the module.

A large wire-wrap section for custom interface circuitry has provisions for a 34-pin connector. A software package, written for the Rockwell SYSTEM 65 and AIM 65, gives complete IEEE-488 bus control.

With a 72-hour burn-in and a full year warranty, the GMS6514 module is $250 in single piece quantity. Available off-the-shelf in the US, Canada, South America and Europe, normal quantity discounts apply.

General Micro Systems, Inc., located at 1320 Chaffey Ct., Ontario, Calif. 91726, (714) 621-7532, designs and manufactures a family of microcomputer modules and systems directly compatible with the 6500/6800 bus, plus associated software. ©



EAST/WEST

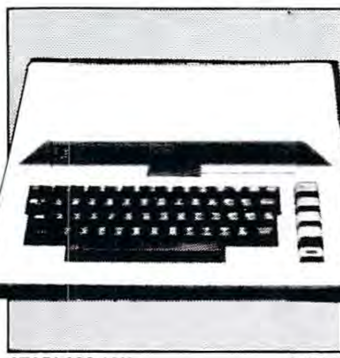
LOOK NEXT MONTH FOR OUR NEW SUPER SPECIALS FOR 1982!

SAVINGS!

CHRISTMAS SPECIALS!

OMEGA SALES CO.

LOOK CLOSELY AT THESE CHRISTMAS SAVINGS!!



ATARI 800 16K $ 749

EPSON MX-80 $ 469

EAST COAST
1-800-556-7586

WEST COAST
1-800-235-3581

PRICES ARE SUBJECT TO CHANGE W/O NOTICE.

OKIDATA MICROLINE-83A

INTERTEC SUPERBRAIN 64K RAM	**$2799**
QD SUPRBRAIN	2999
NEC5510 SPINWRITER (7710)	**2345**
NEC 5520 SPINWRITER (7720)	2695
NEC 5530 SPINWRITER (7730)	2345
NEC 12" MONITOR	229
NEC COLOR 12" MONITOR	339
NEC PC 8023 Printer 100 CPS Tractor & Friction	639
OKIDATA MICROLINE-80A	**399**
OKIDATA MICROLINE-82A	529
OKIDATA MICROLINE-83A	769
DIABLO 630	1995
APPLE II PLUS 48K	**1139**
APPLE DISK w/3.3 DOS Controller	525
APPLE DISK w/o Controller	449
EPSON MX-80	**469**
INTERFACES: IEEE $55, TRS-80 $35, APPLE INTERFACE & CABLE $90, RS-232 $70	
HAZELTINE 1420	799
NORTHSTAR HORIZON II 32K QD	2925
ANADEX DP-9500/9501	1249
TELEVIDEO 912C	**669**
TELEVIDEO 920C	729
TELEVIDEO 950	929

We Accept C.O.D.'s • Stock Shipments Same Day or Next • No Surcharge for Credit Cards • All Equipment Factory Fresh w/MFT Warranty •

We carry the complete line of Personal Software.

$749

CBM 8032 COMPUTER	**$ 1149**
CBM 8050 DISK DRIVE	1349
CBM 4032 COMPUTER	1029
CBM 4040 DISK DRIVE	1029
CBM 402	649
CBM VIC-20	269
LEEDEX / AMDEK 100	**139**
LEEDEX / AMDEK 100G	169
LEEDEX / AMDEK COLOR-1 13" Color Monitor	329
MICROTEK 16K RAMBOARD for Atari	**79**
MICROTEK 32K	149
ATARI 800 16K	**749**
ATARI 400 16K	349
ATARI 810 Disk Drive	449
ATARI 82040 Column Printer	299
ATARI 822 40 Column Thermal Printer	349
ATARI 825 80 Column Printer	599

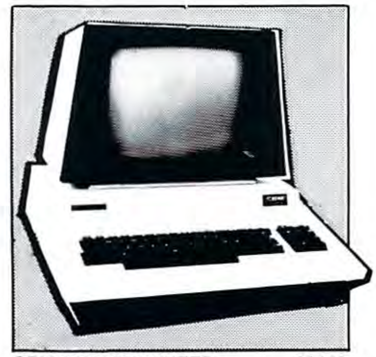
OMEGA SALES CO.

EAST COAST
1-800-556-7586
OMEGA SALES CO.
12 Meeting St.
Cumberland, RI 02864
1-401-722-1027

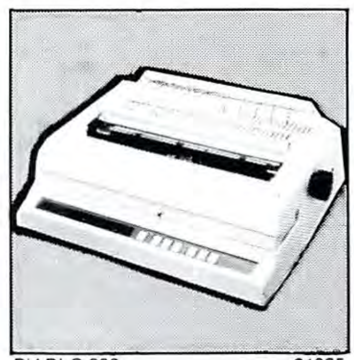
WEST COAST
1-800-235-3581
OMEGA SALES CO.
3533 Old Conejo Rd. #102
Newbury Park, CA 91320
1-805-499-3678
CA. TOLL FREE 1-800-322-1873

OMEGA SALES CO.

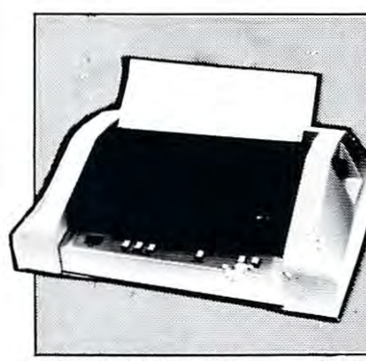
CHRISTMAS SPECIALS!



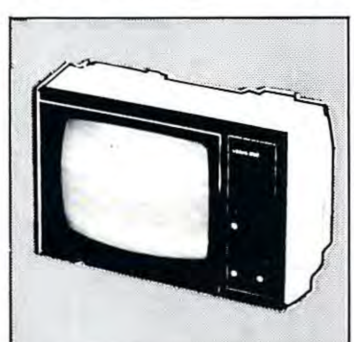
CBM 8032 COMPUTER $1149



DIABLO 630 $1995



ANADEX DP-9500/9501 $1249



LEEDEX/AMDEK COLOR-1 13" $ 329

Advertiser's Index

BEFORE YOU BUY A FINE PRINTER, READ THE FINE PRINT.

That's where you'll find the prices—in the tiny type under the brief descriptions.

Which is only appropriate. Because with C. Itoh's full line—from the workhorse business printers to the racehorse word-processing printers—the descriptions could be much bigger.

But the prices couldn't be much smaller.

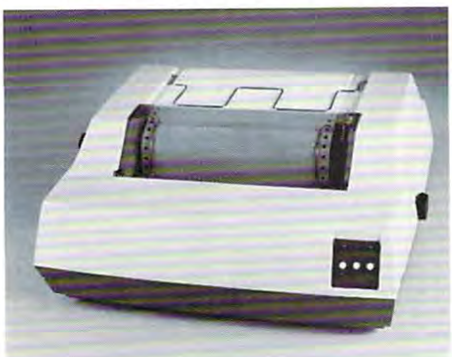
Moreover, as part of the Leading Edge Inventory Bank,™ our printers (along with dozens of other popular computer products) are available for 24-hour "withdrawal." Within a day of when we receive your order, your order will be on its way from our nearest warehouse (Boston, New York, Los Angeles, etc.), thus saving you both time and freight. So you can use our space, and our money, to inventory your product, and still be sure of having exactly what you need, exactly when you need it.

All the printers on this page are warranteed by Leading Edge™ for 3 months on parts and labor.

For the name of your nearest dealer—just pick up the nearest telephone.

TOLL-FREE 1-800-343-6833.

In Massachusetts, call collect (617) 828-8150.



Big business
(Comet II)

The first 15" printer to retail for under a grand. Full 136 columns for standard-width accounting reports, giving you easy-to-read 10 cpi printout up to 66 lines deep. And, like with the Comet I, ribbons are available everywhere. Retail: about $995.

Beyond business
(Starwriter I)

Lowest-priced letter-quality printer on the market. Comes complete and ready to use, requiring no changes in hardware or software. Industry-standard daisy wheels and ribbon cartridges. Plug-in compatible with a wide variety of systems. 25 cps; 163 columns; precise character placement; full graphics capability. Retail: about $1,895.

Busy business
(Starwriter II)

The racehorse of letter-quality daisy wheel printers. Typical 45 cps throughput is about equal to (or, in bi-directional mode, actually *faster* than) "thimble" printers rated at 55 cps. Like the Starwriter I, it uses easily available daisy wheels and ribbon cartridges, and it's compatible with software packages, like Wordstar and Magic Wand, programmed for systems like the Qume Sprint 5. Retail: about $2,395.

LEADING EDGE.™

Leading Edge Products, Inc., 225 Turnpike Street, Canton, Massachusetts 02021.

DEALERS: For immediate delivery from the Leading Edge Inventory Bank,™ call toll-free, **1-800-343-6833.**
In Massachusetts, call collect (617) 828-8150, Telex 951-624.

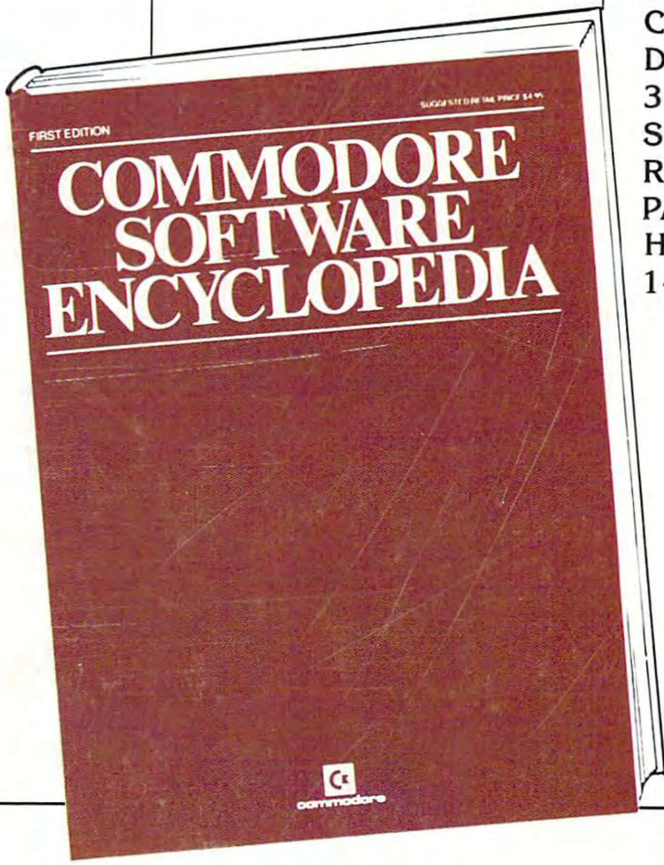
Commodore Software Encyclopedia

The COMMODORE SOFTWARE ENCYCLOPEDIA is now available from your authorized COMMODORE Dealer, for . . .

$4.95

The next edition of the COMMODORE SOFTWARE ENCYCLOPEDIA will be available in 90 to 120 days. If your software is not listed and you would like to have it listed in the next edition, please submit details to:

Software Department
Commodore Business Machines, Inc.
Computer Systems Division
300 Valley Forge Square, 681 Moore Road, King of Prussia, PA 19406.
HOTLINE Number
1-800-523-5622



commodore