



SEPTEMBER '84

I-M 1 IN A MILLION CLUB NATIONAL NEWSLETTER

INSIDE...

CONSCIOUS by RILL BORMAN
QUAD 'ARCADE'
MOVE FROM KAL BLOOM
Q & A FOLLOW UP
CLOSING DATA SNAP
DISK UTILITY PROGRAMS IN
REVIEW from BENHART
GEORGE KARABIN'S 'ALPHA
CENTURIAN' IN REVIEW
PROGRAM FROM A. PRESSOLA
SPOTLIGHT ON G.R. JONES



GEO-GRAFIX LIMITED

P.O. Box 54 • Arlington Heights, IL 60122

EDITOR-GEORGE BAKER
PRODUCED MONTHLY BY-GEO*GRAFIX LTD.
PRINTED BY-GLENN VOSS PRINTING

BULK RATE
U.S. POSTAGE
PAID
Washington, IL
Permit No. 258

 HARDWARE/SOFTWARE UNBELIEVABLE PRICES!!!!

- OFFER #
- 1 PAK-MAN, SUPERPROG, ESCAPE!! and 2 FREE APP PROGRAMS ALL 3 ONLY \$8.50 shipped free

 - 2 MS. PAK-MAN, POOSEBALL!! and 2 FREE APP PROGRAMS ALL 4 ONLY \$8.50 shipped free

 - 3 ANY 2 OF NINE AND 3 APP PROGRAMS ONLY \$7.50 shipped free

 - 4 2 BASIC CARTRIDGES \$25.00 EACH

 - 5 BK EXPANSION CARTRIDGES \$20.00 shipped free...ONLY 2 LEFT IN STOKG

 - 6 AS IS IM-1's INCLUDES TOP, BOTTOM, J CONNECTOR \$55.00 shipped free (CANADIANS ADD \$5 s/h)

 - 7 APP DIAGNOSTIC and FREE APP PROGRAM ONLY \$5.00

 - 8 ANY 1 OF MY GAMES: PAK-MAN, SUPERPROG, ESCAPE, MS.PAK, POOSEBALL ONLY \$5.00. INCLUDES FREE APP
- I JUST RECEIVED A NEW SHIPMENT OF SOFTWARE, SO IF I WAS PREVIOUSLY OUT OF A GAME YOU ORDERED, IT'S IN NOW. PLEASE LIST ALTERNATES, THESE GAMES ARE G O I N G P A S T!!!!
- 9 7 APP PROGRAMS ONLY \$5.50 shipped free

 - 10 12 APP PROGRAMS ONLY 10.50 shipped free

ORDER FORM MAKE CHECK PAYABLE TO: ERIC BECKETT
 8836 W. Waterford Sq. S.
 Greenfield, WI 53228

OFFER #	PRICE
1-----	\$8.50
2-----	\$8.50
3-----	\$7.50
4-----	\$25.00ea.
5-----	\$20.00ea.
6-----	\$55.00
7-----	\$5.00
8-----	\$5.00
9-----	\$5.50
10-----	\$10.50
<hr/>	
TOTAL \$	

APP PROGRAMS AVAILABLE

___ ELECTRONIC FILES	___ MUSIC COMPOSER
___ TYPING TUTOR	___ SPACE DESTROYERS
___ BUDGET MANAGER	___ HANGMAN
___ PERSONAL BUS.MACH.	___ SHOOTING GALLERY
___ BILLBOARD	___ CASINO
___ SPACE, SIZE, SURPACK	___ BASEBALL
___ MATH TUTOR	___ BOXING
___ THE WORD FACTORY	___ BACKGAMMON
___ SPELLING BUEL	___ GATENA
	___ JUMBLED UP THINGS

CANADIAN ORDERS PLEASE
 INCLUDE \$2.00 TO HELP WITH
 SHIPPING.

Please list 'A' for ALL ALTERNATES
 ___ PAK MAN ___ SUPERPROG ___ ESCAPE
 ___ MS. PAK ___ POOSEBALL

GENERAL NEWS

NEW PROGRAM COMING SOON!

PICTURES & MOTION(tm) by A. FRESSOLA, Fairfield, CT will be available soon. This program permits the creation, editing, and storing of HIGH RESOLUTION shapes, and allows the user to move shapes or entire pictures around the screen.

The complete instructions (12 pages) make it easy to create beautiful high res pictures and store them on tape for future display.

A complete review will appear next month.

FROGGY ON THE FREEMWAY DONATED!

Eric Beckett's popular HI RES program 'FROGGY' is now a part of the IM-1 CLUB PROGRAM LIBRARY.

For those who haven't played this fine game, here's your chance to obtain it for the price of processing from the club program library. As mentioned in earlier issues, the only charge is to cover postage and handling. This charge is \$5.00 for every 3 programs ordered.

Be sure to check the GENERAL NEWS in subsequent issues for other programs that can be ordered from the library.

A very SPECIAL THANKS to Eric for his generous contribution!

SCREENWRITER II--TEAMWORK!

Keith Phillips came to our aid when we experienced some trouble with his program. A new revised tape from Keith was in the mail promptly and we are now able to fill the requests for this program. Thanks to Keith for coming through again!

Also... Herbert Keith from Asheville, NC sent us revised instructions for SCREEN WRITER II on tape using Jim Clatfelter's 'COPYWRITER' program. It's really good to see club members working together with available programs to produce enhancements and improvements to other programs. Many thanks to Herbert for all the time and effort!

ALTERNATIVE KEYPAD NEEDED

Les Peterson from Nashua, NH has written requesting information regarding replacement of the original APP NUMERIC PAD with one of different manufacture. He is experiencing double entry problems. Can anyone help?

MORE CLUB PROGRAMS ARRIVE!

Bill Bowman and Harry Brown have been busy this summer creating MORE programs for club members to use.

From Bill....COMCOST...a program that compares the cost of items when sold by the ounce or pound, displays the price differences, and gives a dollar figure for the BEST BUY. The program appears within this issue.

From Harry....A randomly generated POETRY PROGRAM, a calculation routine, and a LO RES scoring display routine.

Thanks to Bill and Harry for their continued programming contributions!

HOPE WE HAVEN'T MISSED YOU!

It has been a very interesting year so far and we have been busy almost every evening responding to letters from members and non-members alike. The load has slackened a bit lately so we want to insure that we have responded to ALL members who may have written us.

If YOU have written and HAVE NOT received a reply, please write again. If we have missed you, rest assured that it wasn't intentional. We'll get a reply to you promptly.

BYTE--CLUBS AND NEWSLETTERS

An article about the IM-1 IN A MILLION CLUB SHOULD be appearing in the SEPTEMBER issue of BYTE magazine in the CLUBS and NEWSLETTERS section.

A brief description was sent to BYTE about 4 months ago and we are looking forward to seeing it in September.



Q & A

FOLLOW UP

The following information was supplied to us by GLENN JONES in response to some of the questions appearing in the MAY Q & A TOSS UP section. Thanks Glenn for your help.

FROM KEN PASSAILAIGUE--CANADA

1. A way to SLOW DOWN THE LIST (scroll) so that it is more easily readable.

A. An excellent machine code scrolling routine was available from:

Carl Echols
112 Creekside Lane
Noblesville, IN 46060

Editors note: Carl IS NOT a member this year and we don't know if he is still involved with the IMAGINATION MACHINE.

FROM GARY SHELTON--BLOOMINGTON, IL.

"Can you tell me what happened to APF ELECTRONICS INC.? Did they go bankrupt or did they merge with another company, or are they still operating under another name?"

A. As far as I know, the company folded up and their assets were auctioned off.

ALSO FROM GARY SHELTON

"I would also like a description of an IM-2 and how it differs from the IM-1."

A. The IM-2 had the game unit built inside the console with ports in front for optional joysticks. Some of the internal ROM was slightly different, but it would run most IM-1 software. The unit included a 8B, SI-232, 16K RAM, and FI-100 with dual disks. It was to have level 2 BASIC and lower case, but these were never completed. The output was for a monitor rather than a TV.

FROM RICK THUES

The CALL instruction to READ THE KEYBOARD for an input and display the ASCII to the screen doesn't seem to work (CALL32975) HEX80CF. What built-in routine would provide this operation?"

A. The 80CF routine DOES read from the keyboard and store it in the A register, but the 8473 routine must be used to put it to the screen.

FROM CHRIS LINSCHOUTEN

"Has anyone successfully converted their IM-1 to work with a black and white or color monitor?"

A. The previous operators of the club..M B ENTERPRISES INC. were at one time offering the instructions to convert the IM-1 to work with a monitor. These instructions were provided by THOMAS FAIRBAIRNS.

PRODUCT REVIEW

DISKMOD from HEXMART SOFTWARE

Have you ever wanted to VIEW or CHANGE data residing on individual sectors on a disk? Have you ever wanted to look at information residing on a TI-99 or TRS 80 disk?

DISKMOD gives you the ability to do this on your IMAGINATION MACHINE.

The program will allow you to READ FROM or WRITE TO any sector which has been formatted by the INIT command, or by the INIT 40 PROGRAM.

For those of you who have ROMMON (c), its monitor program is supported by this program. For those of you WITHOUT ROMMON(c), DISKMOD will be slightly more difficult to use.

The program is MENU DRIVEN, SELF PROTECTED, and READS INFORMATION that it retrieves from the disk by looking at a buffer address.

A STEP command allows you to advance through the disk, sector by sector, until the desired sector is reached. At this point, the program will return to the main menu. The instructions include a good explanation of DISK FORMAT and useful machine calls. Included on the reverse side of the cassette is HEXMART'S INIT 40V program that initializes a disk with 40 tracks at 10 sectors per track. The program verifies all sectors after initialization is complete. This is an excellent safeguard feature that avoids saving important data on a disk that does not initialize properly.

Together, these programs allow you to expand your storage capability on disk and increase your knowledge of disk formatting, editing, and retrieval of data contained on disk.

DISKMOD provides unmatched versatility with its ability to manipulate disk recorded data.

DISK DIRECTORY from HEXMART SOFTWARE

While we're on the subject of DISK UTILITY PROGRAMS, when could be a better time to review an additional program from HEXMART SOFTWARE called DISK DIRECTORY.

The program is straight forward in its operation and instruction. Therefore, we HIGHLIGHT information taken directly from the instructions provided by HEXMART.

"Disk Directory is a utility program which will allow your computer to keep track of what programs you have and on which disk they are stored. The program uses a custom version of SUPERSORT (machine language sort routine) to sort your program list in a matter of seconds. The list which is created can be saved to disk as well as displayed on the screen or printer. The program is set up to handle a maximum of 200 titles and will do this on an 8K APP.

When you run the program, a menu of three choices will be displayed. The three choices are:

- C- create list of titles
- E- end session
- L- load list from disk

Press -C- to start loading program titles from your disks. If you had previously created a list and saved it to disk, you could press -L- to load that list. Press -E- to stop the program when you have finished using it.

If you press -C- the program will ask for your disk code. This is a 3 digit alpha-numeric code which you should have written on your disks so that you can identify them. It can be any code or number you want. You might use something as simple as 1,2,3, etc. or something more detailed such as G10 to indicate your game disk #10.

Enter the code you have chosen for your first disk. The program will tell you to press any key when you have placed your disk in disk drive #0. When you press any key, the computer will read the program titles from the disk in drive #0 and add your identification code to the end of the titles so you will know which disk they are on.

After the program has read the titles, it will ask you if you want to read titles from another disk. If you say yes, the program will ask for the next disk's code. If you say no, the program will sort the list which has been created and return you to yet a longer menu. The new menu has 6 choices:

- A- ADD TO LIST IN MEMORY
- D- DELETE DISK TITLES
- E- END SESSION
- L- LOAD LIST FROM DISK
- S- SAVE LIST TO DISK
- V- VIEW TITLES

SHORT PROGRAMS

```
5  POKE 2512:0: POKE 0195:60
6  DIM HN(11):GZ(2):P(2)
7  GOSUB 300
8  PRINT : PRINT SPC 150:"COMPARE PRODUCT COST": PRINT
11 PRINT : PRINT "COMPARISONS ARE IN OZ'S.": PRINT : INPUT "CONVERT FROM LBS.(Y-N)":HX
12 IF HX="Y" THEN 400
13 IF HX="N" THEN 19
14 IF HN(0)="Y" IF HN(1)="N" THEN MISC "N? N? N?" GOTO 7
15 GOSUB 300
16 PRINT "IF YOU CONVERTED TO OZ.":
17 PRINT : PRINT "A REMINDER:-": PRINT : PRINT "NO.1'S SPC (11)OZ(1) SPC (11)"OZ."
18 PRINT "NO.2'S SPC (11)OZ(2) SPC (11)"OZ.": PRINT
19 PRINT : PRINT "PROGRAM STARTS:-": PRINT
20 INPUT "OZ.PRODUCT (A)":"A
21 INPUT "PRICE PER UNIT PRODUCT (A)":"C
22 PRINT : PRINT
23 INPUT "OZ.PRODUCT (B)":"B
24 INPUT "PRICE PER UNIT PRODUCT (B)":"D
25 E=C/A:F=D/B
26 IF E=F THEN 90
27 IF F=E THEN 200
28 CALL 1794: POKE 4956:2: POKE 4951:0
29 PRINT "PRODUCT A IS N'S SPC (11)F SPC (11)"PER OZ.,"
30 PRINT "PRODUCT B IS N'S SPC (11)F SPC (11)"PER OZ.,"
31 IF F=E THEN 210
32 PRINT : PRINT "PRODUCT A IS N'S SPC (11)F SPC (11)"MORE PER OZ. THAN PRODUCT B": FOR I=1 TO 50: NEXT
33 MISC "/?/?/?": PRINT : PRINT "N'S SPC (11)F-F+0 SPC (11)"MORE FOR" SPC (11)A SPC (11)"OZ."
34 IF E=F THEN STOP
35 CALL 1794: POKE 4956:2: POKE 4951:0
36 GOTO 100
37 PRINT : PRINT "PRODUCT A IS N'S SPC (11)F SPC (11)"PER OZ.": PRINT "LESS PER OZ.THAN PRODUCT B"
38 MISC "/?/?/?": FOR I=1 TO 50: NEXT
39 FOR I=1 TO 100: NEXT
40 PRINT : PRINT "N'S SPC (11)F-F+0 SPC (11)"LESS FOR" SPC (11)A SPC (11)"OZ.,"
41 STOP
42 CALL 1794: POKE 4956:2: POKE 4951:0
43 RETURN
44 GOSUB 200: PRINT "CONVERT LBS.TO OZ'S.": PRINT
45 PRINT : INPUT "HOW MANY CONVERSIONS (1-2)":"L
46 PRINT : FOR N=1 TO L
47 PRINT N?
48 PRINT : INPUT "OZ.LBS.TO BE CONVERTED":"P(N)
49 J=P/16+1/6
50 GOTO J
51 NEXT
52 PRINT : PRINT "NO.1 CONVERSION:" SPC (11)A(1)
53 IF L=2 THEN 450
54 PRINT : PRINT "NO.2 CONVERSION:" SPC (11)A(2)
55 PRINT : INPUT "RETURN TO PROGRAM-PRESS RETURN ":"HX
56 GOTO 15
```

COM \$ COST

FROM BILL BOWMAN

Use this program to convert your items from ounces to pounds, or, compare the cost of two items that are priced by ounces or pounds. You might be surprised at what a few OUNCES costs you, and, how much you could save on a different sized item.

Spotlight

This month the SPOTLIGHT focuses on GLENN JONES from Tulsa, Oklahoma. Glenn has been making hardware and software for the IM-1 since the early days; games such as STAR TREK, PERZERKX, and connecting cables, memory expansion kits, and serial interfaces. Glenn has also been very active in the past clubs as well as the present one. The following information is printed exactly as it was received. Thank you Glenn for the INSIDE LOOK into your activity with the IM-1.

Glenn Jones here (alias B.A. Jones). I am a field engineer for a major computer peripheral company, am married to a very wonderful and patient computer widow, and have two daughters ages 2 and 9. I bought my IM-1 in March '81 on sale for \$399; retail price was \$599 for the "bare" machine. Since then, it's been apart more times than I could possibly count, making modifications and testing products.

My machine currently has 32K of RAM, a building block, an SI-232, PI-100 (APF parallel interface), R-8K RAM, and an FI-100 with two disk drives. There is a switch on the SI-232 to enable it when I want to use the VOLKSMODEM, and disable it when the PI-100 is to be used (either work with APF's built-in software routines). A switch on my PI-100 enables the standard 640X address or 670X for simultaneous printing with the modem. I am using an OKIDATA MICROLINE 88A printer, and have a TC-71 printer for "letter quality" printouts.

Many, many thanks to George and Susan Baker for providing us with the top quality newsletter we now have (this is the 5th different APF newsletter). Hopefully, we will be able to have them produce on a bi-monthly or quarterly basis again next year. It seems that the majority of us who remain are technically oriented, and are pretty much doing "our own thing". Therefore, we need to support the club, and each other by providing suggestions, questions, programs, and other information. The reasonable prices and consistently high quality of HEXMART's utility programs, GEORGE KARABIN's games, JIM CLATFELTER's business software, and many others are also very nice support to have.

My system is mostly used for word processing: writing letters and filing notes from Bible class. I would love to see someone come up with a "virtual storage" system for the disk to overcome the limited memory problem. I believe APF did this on their mailing list program, but I don't have a copy of the program, and understand it wasn't very good to begin with. At any rate, I found a means of getting an additional 8K of storage in the \$2000-\$3FFF address range, but since it is not "in line" with the rest of RAM, (\$A000-\$FFFF), is inaccessible by BASIC. It can be used for custom ROM, variable storage, machine code subroutines, or entire machine code programs. The entire 8K could even be "booted up" after power on, and used as ROM.

APF uses the entire 8K section to drive the PIA inside the game unit, when only the first 4 bytes are actually needed. Rather than add enough logic to fully decode the area, I opted to mount a switch on the MP-1000 which disables the PIA AFTER power on initialization. This means no joysticks, graphic mode changes, or sound utilization when the PIA is disabled. but it can be enabled any time the additional 8K RAM is not required. Another switch is mounted on the back of my MPA-10 console which is used to disable the 8K for power on init (to avoid interference with the PIA), then to enable it. The R-8K RAM cartridge

Spotlight

must be expanded to 16K, and the rest of the modification consists of logic changes which are done with unused gates of the TTL chips installed in my 24K expansion. See the classified ad for more details.

Since I have about \$1200 in APF equipment on my system which will do just about anything I would like for it to do with all of the software accumulated over the years, I'll be hanging onto it for some time. Besides, a new system today will be "obsolete" technologically in a few months, anyway! And there a few things I would still like to do (or see done) with the APF, like a color monitor modification and a program to "draw" & print the 64 block graphic characters available on some printers (which I am working on). I went to a meeting of the TULSA COMPUTER SOCIETY tonight, and they are doing some really neat things with computers to help the handicapped...fascinating!

Beam me up, Scotty!

SHORT PROGRAMS

The following short program was sent in by Alfred Pressola from Fairfield, Connecticut. "I would like to offer to your readers a program which I wrote that is a modification of a computer benchmark test known as the Sieve of Eratosthenes that appeared in the Jan. issue of *Byte Magazine*. As presented in the enclosed listing, the program will generate a list of prime numbers up to a stated maximum number selected by the user and will also indicate the interval between prime numbers whenever this interval is greater than all previous intervals of prime numbers. The program will also calculate the total number of primes found over the range of numbers selected, as well as the maximum interval between prime numbers within this range of numbers and where this interval occurred."

Thank you Alfred for your contribution to the club!

```
10 PAGE 24578,54: CALL 17046
20 COUNT=1:AI=2:NR=0
21 PRINT "SELECT HIGHEST NUMBER TO TEST": INPUT " FOR PRIMES",N1
22 NR=NR-31/2: IF NR=14688 THEN PRINT : PRINT : PRINT "SELECT A LOWER SUPPLY": PRINT : PRINT : GOTO 21
24 LD=57543- INT (NR)
30 FOR I=0 TO NR: PEEK LD+I,1: NEXT I: MUSIC "MS 46"
40 FOR I=0 TO NR
50 IF PEEK (LD+I)=0 GOTO 170
60 PRIME=I+1+I
70 PRINT PRIME,
75 A2=PRIME:18=NR-A1: IF 18=NR THEN NR=3N: MUSIC "M7": PRINT : PRINT "INTERVAL = ";NR-A2:A1:A4=A2
77 A1=A2
80 I=I+PRIME
95 IF I=NR THEN PAGE LD+I,0:K=K+PRIME: GOTO 90
100 COUNT=COUNT+1
170 NEXT I
190 PRINT : PRINT COUNT," PRIMES FOUND FROM": PRINT "A RANGE OF NUMBERS FROM": PRINT "3 TO ";(2NR+2)
195 PRINT "MAXIMUM INTERVAL WAS ";M1
197 PRINT "BETWEEN NUMBERS ";A1;" AND ";A4
200 END
280 L= PEEK (A1984)A256+ PEEK (A1985):E= PEEK (A1946)A256+ PEEK (A15447)
310 R=C-L: PRINT "MEMORY BURNING = ";M
```

DATA SHOP

This article marks the closing of the DATA SHOP for this year. We hope that you have enjoyed it and have gained some useful information from it.

Before closing, we would like to mention the most common problem with modems, and some solutions that may help you get your modem ON-LINE without delay.

Finally, we have drastically reduced some pages of data that were pulled from THREE separate BBS systems on the EAST COAST. These HARD COPY printouts of BBS systems as they are running may be of interest to those of you who may be considering modem operation with your IM-1. Please pardon the print quality. We wanted to get as many examples on a page that we could.

COMMON PROBLEM

"I got a modem, booked it up, dialed a bulletin board, and got GARBAGE on the screen just before I was disconnected!"

POSSIBLE REASONS

A number of things could cause this to happen.

BAUD RATE--The baud rate on your SI-232 or DC-232 was not matched to your modem, or the rate at the other end of the connection. Most BBS systems answer at 300 BAUD and, if they have the capability, let you SHIFT UP in speed (1200) after the introduction. Some SMARTER systems will SAMPLE your initial BAUD RATE and SHIFT UP or DOWN accordingly.

PARAMETERS--Serial WORD LENGTH and PARITY did not match the BBS system. This is an area that should have been standardized long ago. It causes the biggest headaches with home computer users, not only with modem operation, but with PRINTER operation as well.

Solutions to word length (data and stop bits in a serial stream) were provided in an earlier issue of the newsletter in regard to printer control. The problem becomes apparent in MODEM operation when dialing into another computer that has a different arrangement than the IM-1. The TERM-1 program is fixed in its format of serial data and therefore limits the user to access BBS systems that ARE compatible with it. Parity only becomes a problem when a certain system REQUIRES ODD or NO PARITY in its transmission or reception of data. There ARE a few around!

Any of these items mentioned thus far could cause strange looking characters to appear on your screen, or no characters or control at all.

If at all possible, it would be worth it to contact the SYSOP (system operator) of the BBS system that you plan to access BEFORE trying it. This is not very adventurous, but will save you some unnecessary charges on your phone bill. Most SYSOPS who maintain active boards will be glad to answer any questions you may have.

BBS IN TROUBLE--Some systems are not maintained regularly and eventually get into trouble. We have one locally that has a habit of answering you cordially, taking your name and number, and then goes absolutely WACKED before your eyes! A fitting example of HI TECH at its WORST! A good rule of modem sensibility is, before suspecting YOUR system to be in trouble, TRY DIALING ANOTHER BOARD!

IMPROPER SYNC--If BOTH ends do not sync up together in the allotted time to do so, problems will occur. Generally NO DATA will be received and your screen will remain blank. However, in the silence of the connection CIRCUIT NOISE can cause erroneous characters to be displayed on your screen. Normally the connect tone is long enough to establish the proper connection.

FULL/HALF DUPLEX/ECHO--Repeated characters (doubles) such as HHELLLLOO are a result of one end being in HALF DUPLEX when transmitting data and the other end ECHOING the data back. The half duplex mode displays each character typed onto the screen and at the same time transmits the character to the other end. If the other end is conditioned to ECHO received data back to the originating end (YOU), it will send each character in order back IMMEDIATELY. These returned characters will be placed adjacent to the others on your screen thus producing the DOUBLE display. To overcome this unwanted characteristic, the transmitting party should change to FULL DUPLEX operation. This operation will transmit each character as it is typed but will not display it. When the other end ECHOS the character, it will be then be displayed on your screen by itself.

DATA SHOP

If you need help in getting ON-LINE with your modem, or would like to establish a hook-up with another IN-1 to test with or exchange data with, please write ahead of time and let us know when you would like to schedule. If you have a printer and would like to gather a few programs over the phone, the club library is at your disposal. By the way...if you happen to know of any good systems that other club members would enjoy getting into, please let us know. We'll pass the information along to others in the newsletter.

Appearing below are reduced pages of information gathered from the phone line when hooking up to three separate BBS systems that are OPEN TO THE PUBLIC. GOOD LUCK WITH YOUR MODEM!!

WELCOME TO ONLINE
DE OPERATE MODEM OF ME
A TIME TO BE
SUPPORTIVE AND
- ALL WELCOME AND ENCOURAGED TO
TO HAVE REQUEST
- ALL WELCOME OFFLINE AND AVAILABLE
TO MEET WITH ME OFFLINE
OR OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM
OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM
OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM

LOCAL PROGRAMS SECTION
- ALL WELCOME OFFLINE AND AVAILABLE
TO MEET WITH ME OFFLINE
OR OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM
OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM
OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM

WELCOME TO ONLINE
DE OPERATE MODEM OF ME
A TIME TO BE
SUPPORTIVE AND
- ALL WELCOME AND ENCOURAGED TO
TO HAVE REQUEST
- ALL WELCOME OFFLINE AND AVAILABLE
TO MEET WITH ME OFFLINE
OR OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM
OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM

WELCOME TO ONLINE
DE OPERATE MODEM OF ME
A TIME TO BE
SUPPORTIVE AND
- ALL WELCOME AND ENCOURAGED TO
TO HAVE REQUEST
- ALL WELCOME OFFLINE AND AVAILABLE
TO MEET WITH ME OFFLINE
OR OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM
OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM

WELCOME TO ONLINE
DE OPERATE MODEM OF ME
A TIME TO BE
SUPPORTIVE AND
- ALL WELCOME AND ENCOURAGED TO
TO HAVE REQUEST
- ALL WELCOME OFFLINE AND AVAILABLE
TO MEET WITH ME OFFLINE
OR OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM
OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM

WELCOME TO ONLINE
DE OPERATE MODEM OF ME
A TIME TO BE
SUPPORTIVE AND
- ALL WELCOME AND ENCOURAGED TO
TO HAVE REQUEST
- ALL WELCOME OFFLINE AND AVAILABLE
TO MEET WITH ME OFFLINE
OR OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM
OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM

IMPORTANT NOTICE
Please read instructions for a new important addition!
- ALL WELCOME AND ENCOURAGED TO
TO HAVE REQUEST
- ALL WELCOME OFFLINE AND AVAILABLE
TO MEET WITH ME OFFLINE
OR OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM
OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM

WELCOME TO ONLINE
DE OPERATE MODEM OF ME
A TIME TO BE
SUPPORTIVE AND
- ALL WELCOME AND ENCOURAGED TO
TO HAVE REQUEST
- ALL WELCOME OFFLINE AND AVAILABLE
TO MEET WITH ME OFFLINE
OR OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM
OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM

WELCOME TO ONLINE
DE OPERATE MODEM OF ME
A TIME TO BE
SUPPORTIVE AND
- ALL WELCOME AND ENCOURAGED TO
TO HAVE REQUEST
- ALL WELCOME OFFLINE AND AVAILABLE
TO MEET WITH ME OFFLINE
OR OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM
OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM

WELCOME TO ONLINE
DE OPERATE MODEM OF ME
A TIME TO BE
SUPPORTIVE AND
- ALL WELCOME AND ENCOURAGED TO
TO HAVE REQUEST
- ALL WELCOME OFFLINE AND AVAILABLE
TO MEET WITH ME OFFLINE
OR OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM
OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM

WELCOME TO ONLINE
DE OPERATE MODEM OF ME
A TIME TO BE
SUPPORTIVE AND
- ALL WELCOME AND ENCOURAGED TO
TO HAVE REQUEST
- ALL WELCOME OFFLINE AND AVAILABLE
TO MEET WITH ME OFFLINE
OR OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM
OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM

WELCOME TO ONLINE
DE OPERATE MODEM OF ME
A TIME TO BE
SUPPORTIVE AND
- ALL WELCOME AND ENCOURAGED TO
TO HAVE REQUEST
- ALL WELCOME OFFLINE AND AVAILABLE
TO MEET WITH ME OFFLINE
OR OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM
OFFLINE. I WILL BE OFFLINE
FROM 10:00 PM TO 10:00 PM

PRODUCT REVIEW

ALPHA CENTURION by George Karabin

"In a far corner of the galaxy Alpha Centauri lies the planet Nibar. It is of strategic importance to the confederation because of its vast deposits of the element X-287; the principle component of fuel that has enabled "WARP" speed and intergalactic space travel."

This information appears in part from the cover of George Karabin's ALPHA CENTURION instruction booklet.

Putting planet Nibar, element X-287, and warp speed aside for a moment, let's take a look at the actual components and layout of the game.

The layout of the MI RES screen is very similar to SPACE DESTROYERS with rows of aliens (landers, tie fighters, and pods) appearing in the upper portion of the screen. These creatures move side to side as with the SPACE DESTROYERS program, and it's up to the player(s) to eliminate as many aliens as possible. Your joystick controls your weapon which is called an X-WING FIGHTER and allows you to fire photon torpedoes at the aliens. Some MI RES SHAPES appear BELOW your X-WING FIGHTER. These make up the city that you must protect from destruction.

At this point...the similarity with SPACE DESTROYERS ENDS!

Your X-WING FIGHTER is equipped with a full load of PHOTON TORPEDOES and a 4 element shield. The object is to protect the city, fire photons, and destroy as many aliens as possible before they destroy 3 of your fighters, or wipe out the buildings below. The aliens drop bombs which the X-WINGS can stop, but unlike SPACE DESTROYERS, the aliens have additional features that make the game much more interesting.

At random, aliens from the first row descend (one at a time) upon the city. These 'LANDERS' are suicidal; inact on crashing into a building unless destroyed. If a gap in the formation of the alien groups occurs, the alien 'TIE FIGHTERS' (second row), or the 'PODS' in the third row will descend in a random pattern.

The TIE FIGHTERS are not suicidal, but if they exit the screen in their side-to-side movement, they will re-appear again in the formation.

The 'PODS', like the 'LANDERS' will crash into the buildings unless stopped!

So instead of plain old-fashion bombs being dropped on you as in the SPACE DESTROYERS game, ALPHA CENTURION contains a SWARM of dropping bombs and diving aliens that truly challenge your firing skills and reflexes.

Scoring is determined by the amount and type of aliens destroyed, PLUS each building that is saved at the end of each round. HIGH SCORE is also displayed. The game comes with complete instructions, 8K, and has a 2 player option.

CLASSIFIED

*****WANTED*****

APP HARDWARE in good working condition. Will pay cash.

J. Field (212) 434-0781

or write to:

J.H. Field

Box 292, Midwood Sta.

Brooklyn, NY 11230

*****EXPANSION*****

24K MEN EXPANSION INSTRUCTIONS requires building block & BB-K RAM; comes w/detailed instr's, illust's, & schem's—only \$14.95

32K EXPANSION INSTRUCTIONS for use w/above.....\$9.95

BOTH 24K & 32K INSTRUCTIONS

FOR EXPANSION.....\$19.95

EIGHT 4116 RAM I.C.'s FOR EITHER

EXPANSION.....\$12.95

G.R. JONES, 419 S. 105 E. Pl.

TULSA, OKLAHOMA 74128

THE ARCADE

OK...it's time to MOVE! Eric has laid the ground work and has recently supplied us with an outline for the remainder of the year. It is extonsive! Therefore, this issue becomes the beginning of LARGE ARCADE articles. There's a lot of room in the newsletter this month so we are able to EXPAND this section as never before. Begin by loading in the BASIC program that was given last month, enter the ML monitor mode (CALL28672) and enter the following ML data. A complete step by step description follows. This ML data appears exactly as it was received so that we don't risk the chance of an incorrect entry as before.

A412	CE	code X reg. immediatly	3A	81	compare
13	02	middle of	3B	57	ASCII W
14	EE	screen	3C	26	Branch if \neq 0
15	FF	store X reg. Im.	3D	03	offset
16	A4	permanent address wher	3E	78	jump to routine
17	10	joyetick location	3F	A4	move
18	86	load acc. A In.	A440	50	W
19	CF	white block	41	01	no operation
1A	A7	etore acc A indexed	42	01	" "
1B	00	off eet	43	01	" "
1C	BD	jump to subroutine ext.	44	30	return to basic
1D	41	right	45	FE	load X reg. ext.
1E	D9	joyetick	46	A4	permanent joystick
1F	25	branch if carry is set	47	10	location
A420	01	offset for branch	48	86	load acc. A Im.
21	39	return to Basic	49	20	code for background
22	B6	load acc. A ext.	4A	A7	store acc A ind.
23	01	where joystick	4B	00	offset
24	F2	data is located	4C	08	increment index reg.
25	81	compare acc. A	4D	7E	jump
26	4E	ASCII code for North	4E	A4	draw square
27	26	Branch if not equal to 0	4F	7B	routine
28	03	Branch offset	A450	FE	load X reg. ext.
29	7E	jump ext	51	A4	Permanent joystick
2	A4	move N	52	10	location
2	5B	subroutine	53	86	load acc. A Im.
2	81	compare acc. A im.	54	2-	background code
2	53	ASCII for eouth	55	A7	store acc. A Ind.
2	26	branch if \neq to 0	56	00	offset
2	03	branch offset	57	09	decrement index reg
30	7E	jump ext.	58	7E	jump
31	A4	move S'	59	A4	draw square
32	6B	routine	5A	7B	routine lo
33	81	compare acc. A Im.	5B	FE	load X reg ext.
34	45	ASCII for E	5C	A4	joyetick
35	26	branch if \neq 0	5D	10	location
36	03	branch offset	5E	86	load acc. A
37	7E	jump to routine	5F	20	background code
38	A4	move E	A460	A7	store acc. A ind.
39	95	routine	61	00	offset
			62	86	load acc. A
			63	20	decrement variable
			64	09	decrement index reg.
			65	4A	decrement acc. A
			66	26	branch if \neq to 0
			67	FC	offset
			68	7E	jump
			69	A4	draw square
			6A	7B	routine
			6B	FE	load x reg. ext.
			6C	A4	joystick
			6D	10	location
			6E	86	load acc. A
			6F	20	background

THE ARCADE

```
A4 70 A7 store acc. A ind.
71 00 offset
72 86 load acc. A
73 20 decrement variable
74 08 increment X reg.
75 4A decrement acc. A
76 26 branch if ≠ 0
77 0C offset
78 01 NOP
79 01 NOP
7A 01 NOP
7B 01 NOP
7C 01 NOP
7D 01 NOP
7E 86 load acc. A
7F 0F white square code
A4 80 Q7 store acc A ind.
81 00 offset
82 FF STORE X reg. ind.
83 Q4 joystick
84 10 location
85 B6 load acc. A ext.
86 A4 low byte of
87 11 joystick location
88 4A decrement acc. A
89 81 compare acc. A ind.
8A 0C compare value
8B 26 branch ≠ 0
8C 0F offset
8D FE load X reg. ext.
8E A4 joystick
8F 10 location
A4 90 86 load acc. A
91 20 background code
92 A7 store A ind.
93 00 offset
94 08 increment X reg.
95 B6 load acc. A
96 0F white block
97 A7 store acc. A ind
98 00 offset
99 FF store ind. reg
9A A4 joystick location
9B 10 location
9C C6 load acc. B
9D FF 255 dec.
9E B6 decrement acc. A
9F FF branch if ≠ 0
A0 4A offset
A1 26 decrement acc. B
A2 FD branch if ≠ 0
A3 5A offset
A4 26 NOP
A5 FB NOP
A6 01 NOP
A7 01 NOP
A8 01 NOP
A9 39 return
```

After entering this program triple check it carefully because it is very easy to make a typing error. The program is about 150 bytes long. It would be a good idea to go back to BASIC (G8894) and CSAVE to tape before continuing. Then if your program bombs out because of a bug, you can GLOAD it back in and make the necessary changes to it.

The length of the program may seem like a lot but it's not bad considering that reading the joysticks and moving the object is a major part of a game program. Many routines coming up are this long or longer. Example, most of my games used the entire 8K of memory and they were around 90% ML which is THOUSANDS of commands. Don't worry. You can do quite a lot in 8K and in most cases you won't need all of it.

Here is a line by line description of what is going on. Our BASIC program is short. The first few lines are REM statements containing A's where our ML program will go. Line 25 is a simple POKE to turn off the audio on your tape deck. At 30 we call a sub-routine that clears the screen. Line 40 goes to our ML program. At the end of our ML program we come back to line 50 which goes to 40 again. Remember we use this loop instead of doing it all in ML because 100% ML programs can't be stopped with the break key. This means that if it doesn't work, or you want to modify it, you would have to reset your computer and reload it.

Here is a line by line description of the ML program.

We start off at A412 by loading the index register immediately with 02EE. This address is about the center of the screen. The at A415 we store the index register at A410. What we are doing is using A410 and the next byte to hold a variable. This will be the current location of the white square. At A418, we load the acc. with the code for a white square and at A41A, we store the white square indexed or at 02EE then at A41C, we jump to the right joystick routine. At A41D, we branch if the carry bit was set. It branches over the return code at A421. So what we do is return to BASIC if the joystick wasn't used. If the joystick WAS used, we continue on to A422 where we load the A acc. with the data located at 01F2.

THE ARCADE

At A425 we compare the value in A to the ASCII code for NORTH (4E). The compare instruction works like this (VALUE in O1P2) -4E. The result affects the condition code register so right after it we can do a branch if =0 at A427. If the value in acc. A is equal to 4E (N), then we don't branch and jump to N routine. If it is NOT, we branch to A42C where we compare acc. A to the S ASCII code or (53). Just like N and S, we do the same compare for E at A433 and west at A43A. If it isn't west, we end up at A441 which is a NOP command and these type of commands do NOTHING at all. They are used to leave holes in your program in case you want to add a routine at a later date. If you don't leave holes, it can be very frustrating later when you seed room in the middle. We finally come to A444, which is the return code which takes us back to our BASIC program at line 30. Notice around A437, if the joystick was pressed EAST, we jump to A445. The first thing that we do there is load the X register with the address at A410. A410 is where we store the current joystick location. Next at A448, we load the A accumulator with the ASCII code for background color (20). Then at A44A, we store the acc. at the address in the X register, which is Q2EE the first time. This erases whatever was at Q2EE. At A44C, we increment the value of the X register once, which moves it EAST. Then at A44D, we jump to a routine that draws the white square, checks to see if the value 0D is in the current joystick location A410, then it delays a little and returns to our BASIC program. The reason we check to see if our white square is at 000D or 010D is because if we return to BASIC with a 0D in our program, it will stop running and you will get an illegal line statement. Again, this is because the APF reads 0D as an end of line statement. At A480, we store the white square and at A482, we store the value in the X register at A410 (our joystick location). Remember before we jumped to this routine, in the last routine we loaded the index register with what was at A410. Then we added or subtracted from it depending on the joystick movement and jumped to the routine. When you jump to a routine, the registers are unchanged so at A482, we store the new value in the X register back in A410.

At A485 we load the A acc. with whatever is at A411. This is the low byte of our joystick location, and this is where we check it for HEX 0D. We can't compare our number to 0D because we can't use 0D, so we decrement acc. A at A488 and then check for 0C instead of in A48A. If acc. A isn't equal to 0C, we branch from A48B ahead 0F bytes or to A49C where we start the delay. If the A acc. is equal to 0C we then have to change it. One quick way is used, at A4ED we load the X register with what is at A410. At A490, we load acc. A with the background color and store A indexed which erases the white square. At A494, we increment the X register once. At A495, we load the A acc. with a white square and at A497, store it indexed. Next at A499, we store the new value in X at A410. We just increased the 0D to 0E, then go on to the delay at A49C. This delay is different from the one in our other example. It works a lot like it. First we load acc. B with FF, then at A79E we load acc. A with FF. In this routine we will decrement acc. from FF or 255 to zero, 255 times. This is the maximum for this delay but later we'll speed it up a little bit at a time until we reach the speed we want. At A4A1, we check to see if the A acc. is equal to zero. If it is not it branches back to A4A0 where it keeps decrementing A until it does equal zero. Then it goes to A4A3, which decrements the B acc. Then we check to see if the B acc. is equal to zero. If it is not we branch backwards to A49E where we load acc. A again and start all over again. When B is finally equal to zero, we go to A4A6 where we see three NOP commands. These again are just to leave room. At A4A9 we finally return to BASIC. Before you run this program, again, be sure to save it to tape. When you first run it, the action is very slow. One thing to point out is, don't move your square off the top of the screen or off the bottom. If you do, you will be actually storing the square at RAM addresses other than the screen memory. You could end up writing over your program. Later on we'll put a border around the screen so we can't go off it. For now we'll just concentrate on staying on the screen. If the program seems to be working all right, we can go ahead and shorten the delay. Change A49D to 20 and A49F to 35 and RUN.

THE ARCADE

That wraps it up for this month. If anyone has questions regarding this material, please send them in. We will compile them and forward them to Eric Beckett. We ask that you send them to us so that we can keep track of all problem areas that may exist. We will then attempt to clarify these problems in future issues for the benefit of others who may have similar problems.

Appearing below is a useful chart that provides decimal, hex, and ASCII values that can be used when POKING characters to the screen. An example of POKING the ASCII character D would be: POKES12,4 The 4 would be in DECIMAL when in the BASIC mode. This command would cause the character D to be displayed in the upper left corner of the screen.

If you are using MACHINE LANGUAGE which requires a HEX value, the value that would be required would be 04 in HEX. For an ENHANCED character, the value would be 44 in HEX.

THE ARCADE

		HEX VIDEO		HEX VIDEO	
DECIMAL	ASCII	NORMAL	ADVANCE	DECIMAL	ASCII
0	●	0	40	32	Space
1	A	1	41	33	'
2	B	2	42	34	"
3	C	3	43	35	^
4	D	4	44	36	&
5	E	5	45	37	§
6	F	6	46	38	•
7	G	7	47	39	'
8	H	8	48	40	(
9	I	9	49	41)
10	J	0A	5A	42	*
11	K	0B	4B	43	+
12	L	0C	4C	44	,
13	M	0D	4D	45	-
14	N	0E	4E	46	.
15	O	0F	4F	47	/
16	P	10	50	48	0
17	Q	11	51	49	1
18	R	12	52	50	2
19	S	13	53	51	3
20	T	14	54	52	4
21	U	15	55	53	5
22	V	16	56	54	6
23	W	17	57	55	7
24	X	18	58	56	8
25	Y	19	59	57	9
26	Z	1A	5A	58	:
27	[1B	5B	59	;
28	\	1C	5C	60	<
29]	1D	5D	61	>
30	^	1E	5E	62	>
31	~	1F	5F	63	?

THE ARCADE

FROM BASIC 1. POKE DECIMAL WILL DISPLAY ASCII POKES66,25
 2. ADD 64 TO UNIMAL WILL DISPLAY ASCII BLACK OR YELLOW
 POKE #93,60 WILL CHANGE ASCII TO BLACK OR ORANGE

FROM MACHINE 1. STORE NUMERICAL TO SCREEN ADDRESS WILL DISPLAY ASCII
 2. STORE ROTHSIDE VIDEO HEXIDECIMAL FOR REVERSE VIDEO