

PEEK (65)

The Unofficial OSI Users Journal

P.O. Box 347
Owings Mills, Md. 21117
(301) 363-3267

★★ \$1.75 ★★

JUNE 1982
Vol. 3, No. 6

INSIDE:

THOUGHTS ON OS65U	2
MORE ON THE C1P	5
MINI HEAD LOAD	10
ADD MEMORY TO VIDEO	10
CASSETTE CORNER	13
COPY UTILITY	15

Column One

NEW MACHINE

I have just seen and worked with one of MA/COM OSI's new machines, a "250 I." Wow. It does not look or act like my old C3-D. Let me tell you about it.

The machine comes in a desk-height cabinet with a very nice looking formica woodgrain top. It is about 2 feet wide by 3 feet deep, and very handsome. I wouldn't mind having it in my office at all.

All components are in drawers which slide out easily for access. The top drawer contains a single floppy disk drive and the on/off and boot switches, both attractive rocker switches with enough heft that they will probably last forever. Notice I said the on/off switch is on the front of the machine! Simple things, but important. Inside this unit are the boards for the CPU, printer I/O and five user RAM partitions. On the back are the DB-25 plugs for the 5 user consoles.

The second drawer contains a 74 Mbyte hard disk drive. Both drawers are matched in color and style, white with black accents; together with the walnut-grain top, they would fit in quite well in most any office.

There is a single set of cooling slots on the front of each drawer. Most of the cooling air is drawn through and exhausted, quietly, by two large fans mounted on the back panel. Backed up to a wall (with a few inches left for air circulation), this unit

should stay nice and cool, and quiet enough to allow phone conversations in the same room.

The whole machine has a strong, heavy feel to it. It works as well as I have come to expect Ma/Com OSI computers to work, that is quite well indeed. Now it also looks sturdy.

Incidentally, this particular machine has the Universal Telephone Interface installed, and I used it to call up the PEEK(65) CBBS. Again, most impressive. After taking the menu selection "call any modem," I was asked for the number. Typing in

```
363 4867 <CR>
```

got me instant access to the CBBS, no problem. Its nice to dial from the same keyboard you will "talk" through.

The CBBS, incidentally, is improved. You can now send messages of any length, and they will be stored away on a scratch file. While each line is being put into the file, the board will send you a control-S, followed by a control-Q when it is ready for the next line. Actually, since it uses a hard disk, it is so fast all you need to do to transmit disk files to the board is send them as strings, with perhaps 6 nulls between each line (after the CR/LF) to give the board time to tuck it all away on disk.

If you are typing, of course, you can just ignore the control Q and S business and type

away as soon as you see the ":" prompt for each line.

I encourage you to use your CBBS. There is a lot of good information on the board, and will be even more after you start calling regularly. The number is (301) 363-4867, 5 PM to 7 AM Eastern time, 7 days a week. You have to pay for the call, but you don't have to pay an hourly charge as on Micronet or The Source. Let us know your problems, achievements, questions, answers, whatever. It is self-directing, and typing HELP at any time will let you know what you can do. Give us a try!!

The response to our call for articles has been gratifying, but a bit one-sided. Many technically oriented PEEKers have written some terrific articles on a variety of aspects of personal computing. Now what we need (are we never satisfied?) are articles on successful business installations of Ma/Com OSI computers. I know they are out there; I have made several myself (and will write about them). But we want to hear about yours.

Business articles need not be technical. In fact, it is probably better just to tell us what equipment you used, what software you bought/wrote, and what it does for you. Of course we want to hear about problems, and how you solved them (or didn't solve them -- maybe we can help). So dealers, consultants, businessmen, get out your word processor and let us know what you have been doing.

al

By: Colin Law
 P.O. Box 3819
 Auckland
 New Zealand

20 S\$=CHR\$(27):SU\$=S\$+CHR\$(31):SD\$=S\$+CHR\$(25)
 30 SC\$=S\$+CHR\$(28):SP\$=S\$+CHR\$(17):BP\$=CHR\$(7)
 40 CL\$=S\$+CHR\$(15):SE\$=S\$+CHR\$(24):SB\$=S\$+CHR\$(1)

Recent articles such as In-
 visible Passwords (March 1982)
 and Multiple Report Writer in
 DMS (January 1982) prompt me
 to offer a few of my own 65U
 routines, running DMS Nucleus
 on C3-OEM and C3A with Hazel-
 tine terminals.

Please note that because of
 copyright on the original pro-
 grams, my listings show only
 that part of the program to be
 changed, with perhaps one line
 either side.

PASSWORDS

(A) It's quite clear that you
 can't stop a skilled person
 from accessing data files, but
 I reckoned that by printing
 passwords on screen we would
 get a small degree of protec-
 tion for our staff leave data.
 The ARE YOU SURE routine
 (Listing 1) serves to discour-
 age unauthorized tampering and
 maybe to avoid accidental ac-
 cess to master records. Lines
 20-40 show some of the Haz-
 eltine functions which I
 abbreviate to strings at the
 start of a program.

S\$="escape" (CHR\$(27))
 SU\$=screen up (bright display)
 SD\$=screen down (normal
 brightness)
 SC\$=screen clear
 SB\$=screen blank
 SP\$=print position (to be
 followed by column & row
 numbers)
 SE\$=clear to end of screen
 SF\$=flashing display
 CL\$=clear to end of line
 BP\$=beep

When the operator attempts to
 enter the master file edit
 routine ARE YOU SURE is called
 first. An authorized operator

Copyright ©1982 by DBMS, Inc. All Rights Reserved.
 PEEK (65) is published monthly by DBMS, Inc.
 Owings Mills, MD 21117.

Editor - Al Peabody
 Technical Editor - Dickinson H. McGuire
 Asst. Technical Editor - Brian Hartson
 Circulation & Advertising Mgr. - Karin Q. Gieske
 Production Dept. - A. Füsselbaugh, Ginny Mays

Subscription Rates	
US (surface)	\$15
Canada & Mexico (1st class)	\$23
So. & Cen. America (Air)	\$35
Europe (Air)	\$35
Other Foreign (Air)	\$40

All subscriptions are for 1 year and are payable in advance
 in US Dollars.

For back issues, subscriptions, change of address or other
 information, write to:

PEEK (65)
 P.O. Box 347
 Owings Mills, MD 21117

Mention of products by trade name in editorial material or
 advertisements contained herein in no way constitutes
 endorsements of the product or products by this magazine
 or the publisher.

```

10000 :          REM ARE YOU SURE ^^^^^
10010 REM
10020 PRINTSP$;"au";CL$;SP$;"av";CL$;SP$;"au";
10030 PRINT" WARNING !!";BP$;" YOU MAY ENDANGER THE ACCURACY"
10040 PRINT" OF THE LEAVE RECORDS IF YOU PROCEED":PX$="PROHIBIT"
10050 WAIT64512,1:P=PEEK(64513)AND127:IFP=27GOTO10150
10060 IFP=47GOTO700
10070 PRINTSP$;"au";CL$;SP$;"av";CL$;SP$;"au";
10080 PRINT" TO PROCEED YOU MUST ENTER A PASSWORD AND YOU MUST"
10090 PRINT" BE CERTAIN THAT YOU KNOW WHAT YOU ARE DOING";BP$
10100 WAIT64512,1:P=PEEK(64513)AND127:IFP=27GOTO10150
10110 CK=CK+1:IFCK<2GOTO10140
10120 PRINTSC$:AA$="PLEASE LEAVE ME ALONE":GOSUB59000
10130 AA$="UNAUTHORISED PERSONS NOT ADMITTED":GOSUB59000:GOTO50180
10140 GOTO700
10150 PRINTSP$;"au";CL$;SP$;"av";CL$;SP$;"au";
10160 PRINT" ENTER YOUR NAME ";S$;CHR$(2);:INPUTNP$
10170 PRINTSD$;
10180 IFNP$="AL"ORNP$="DICK"ORNP$="BRIAN"ORNP$="KARIN"GOTO10200
10190 PRINTBP$:CLOSE:AA$="NOT AUTHORISED":GOSUB59000:GOTO700
10200 PRINTSP$;"au";CL$;SP$;"av";CL$;SP$;"au";"PASSWORD";S$;CHR$(2);
10210 INPUTPP$
10220 PRINTSD$;
10230 IFPP$<>"SECRET"THENAA$="NOT AUTHORISED":GOSUB59000:GOTO700
10240 REM
10250 PX$="PERMIT":GOTO1200
10260 REM
    
```

```

59000 FORI=1TO10:FORII=1TO100:NEXT:PRINTBP$:NEXT
59010 A2$="ERROR >>>: A3$=A2$ :GOTO60020
60000 : REM ERR MESSAGE
60010 A2$="SUGGEST YOU TRY AGAIN" : A3$="ERROR >>>> "+AA$
60020 A1$="YOUR DATA IS UNACCEPTABLE - DOES NOT COMPUTE"
60030 PRINTSD$;SP$;"t";CL$;SP$;"t";A1$:TT=500:GOSUB60060
60040 FORI=1TO3:TT=1500:PRINTSU$;A3$;SD$:GOSUB60060:PRINTA2$
60050 TT=500:GOSUB60060:NEXTI:RETURN
60060 FORT=1TOTT:NEXT:PRINTSP$;"u";CL$;SP$;"u";:RETURN
    
```

LISTING 2 'DIR' MODIFICATIONS

```

190 PRINT #DV, TAB(33);"ADDRESS"; TAB(44);"LENGTH";TAB(53);"PASSWORD"
200 FOR I=0 TO 60: PRINT #DV, "-";: NEXT: PRINT #DV

390 N$ = "": FOR I=0 TO 5: N$ = N$+CHR$(PEEK(RT+I)): NEXT
392 PX(2)=PEEK(RT+6):PX(4)=PEEK(RT+7)
394 PX(1)=INT(PX(2)/16):PX(3)=INT(PX(4)/16)
396 PX(2)=PX(2)-PX(1)*16:PX(4)=PX(4)-PX(3)*16
400 REM READ DIR BIT
    
```

```

700 PRINT #DV, TAB(32);DA; TAB(43);SZ;TAB(53);
702 IFPX(1)=15ANDPX(2)=15ANDPX(3)=15ANDPX(4)=5GOTO718
704 PX$(1)=CHR$(PX(1)+65):IFPX(1)=15THENPX$(1)="Z"
706 PX$(2)=CHR$(PX(2)+78):IFPX(2)=15THENPX$(2)="A"
708 PX$(3)=CHR$(PX(3)+65):IFPX(3)=15THENPX$(3)="Z"
710 PX$(4)=CHR$(PX(4)+78):IFPX(4)=15THENPX$(4)="A"
712 FORPT=1TO4:PRINT#DV, PX$(PT);:NEXT:PRINT#DV:GOTO720
718 PRINT#DV, "PASS"
720 IF DA+SZ > HA THEN HA = DA+SZ: REM UPDATE HIGHEST ADR
    
```

LISTING 3

KYUTIL MODIFICATIONS FOR 'RANGE'

```

16100 FORX=1TOTS:INPUT%1,WA(X):INPUT%1,TN(X):INPUT%1,CN$(X)
16110 IFTN(X)<>7THEN16160
16120 FORX1=2TOLN(CN$(X)):IFMID$(CN$(X),X1,1)=">"THEN16140
16130 NEXTX1:PRINT"FALSE RANGE ";CN$(X):GOTO16160
16140 C2$(X)=RIGHT$(CN$(X),LEN(CN$(X))-X1)
16150 CN$(X)=LEFT$(CN$(X),X1-1)
16160 NEXTX
16200 CLOSE 1
16300 RETURN
    
```


Z means anything from P onwards, you can probably figure out your original password.

KYUTIL MODIFICATIONS:

(A) One of the first problems I had with KYUTIL was that the list of conditions didn't include RANGE. Yes, I know that you can get a range by using 'greater than' and 'less than' but this means you must AND the conditions which immediately limits any further conditions to AND. For example you can't create a keyfile of all staff on salaries between 10000 and 20000 located in Auckland (AK) OR Wellington (WN). (KYUTIL either ANDs all conditions or ORs all conditions) listing 3 shows the changes to KYUTIL to allow condition number (7) RANGE.

The 25000 routine gets the condition and the range is saved as one string with ">" in between the bottom and top of the range. Around 16000 the program variables are restored from PASVAL and the ">" is found and bottom and top of range reconstituted. If there is no ">" then we print "FALSE RANGE" but I don't think this could ever happen (I haven't seen it yet) - still it's there as a protection. At 28000 is the actual conditional routine and up at 30000 is the subroutine to display your selections. Note that RANGE is INCLUSIVE.

(B) With over 2000 staff records of 19 fields of which 6 are date fields, it was important to save space if I was to fit the file onto one disk. I decided first to reduce dates from my original 8 characters (DD.MM.YY) to 6 (DDMMYY) and then to 4 (DMYY) - that saved about 48K of disk space. To allow KYUTIL to sort in date order I switched this to YYMD. Year is simply the last two digits of the year, month is letters 'A' to 'L', D is letters 'A' to 'Z' for 1 to 26 with lower case 'a' to 'e' for days 27 to 31. This means a small mod to KYUTIL to allow it to sort lower case. In its original form KYUTIL converts lower case to upper case for sorting - so that MACDONALD turns up next to MACDONALD for example instead of down the bottom after MYERS. I don't mind since we don't store any names in lower case - simply amend line 9935 to read:

```
FORI=97TOL22:AA$=AA$+CHR$(I)
:REM ALLOW LOWER CASE
```

Listing 5 continued

```
7000 REM
7010 FORI=2TOLN(FC$(6)):T(I)=ASC(MID$(FC$(6),I,1)):NEXT
7020 T=1:G=0:T1=2:IFT(2)<>32ANDT(2)<65THENT1=1
7030 G1$=LEFT$(FC$(6),T1):IFT1=1THENG1$=G1$+" "
7040 T2=T1+1:IFT(T1+1)<48ORT(T1+1)>57THENT2=T2+1
7050 T3=2:IFT(T2+1)<48ORT(T2+1)>57THENT3=1:GOTO7070
7060 G2$=G1$+MID$(FC$(6),T2,T3):GOTO7080
7070 G2$=G1$+"0"+MID$(FC$(6),T2,T3)
7080 IFVAL(RIGHT$(G2$,2))>20 THENT3=1:GOTO7070
7090 G3=VAL(MID$(FC$(6),T2+T3+T)):T4=INT(G3/10)+1
7100 G3$=G2$+RIGHT$(STR$(G3),T4)
7110 IFT(T2+1)>47ANDT(T2+2)>47THENT(T2+2)=9:T=0:GOTO7090
7120 IFG3=0ANDG=0THENT1=G=1:GOTO7090
7130 IFG3=0ANDT3=1ANDG=1THENT=0:G=G+1:GOTO7090
7140 IFG3=0ANDG<8THENG=G+1:T=T+1:GOTO7090
7150 IFLN(G3$)<6THENG3$=G3$+" ":GOTO7150
7160 IFG1$="GS"THENG2$=G1$+"P"+RIGHT$(G2$,1):G1$="G ":G3$=G2$+"1"
7170 REM
7180 RETURN
7190 REM
```

Of course you also need to modify your edit and report programs to translate the dates, but for me it meant the difference between fitting it all on one data disk or not and that's an important consideration. If you do use lower case in your data files, then you could use numbers, (say 1 to 5 ?), for days 27 to 31, or amend one of the other lines around 9900 to 'allow' some of the other characters to be sorted. Listing 4 shows sample routines for compression and decompression of dates.

(C) My several variations of KYUTIL are tailored for the specific program sets on each disk (saves all those tedious questions about file name, password, etc.) and they are all about half the size of the original KYUTIL since I had to pack things tight to make room for bigger WORK files (up to 70K). KYUTIL loads and sorts the key file on its own disk and then asks the operator to load a report writer disk into device B. The key file is then copied to the report disk.

(D) Another KYUTIL mod for an odd use - one field in use consists of 3 letters and 2 digits, but very often we want to load a key file with only the 3 letters (plus NAME field etc.) e.g. we want all AAA code in alphabetical order, ignoring the fact that they may be AAA01, AAA44, etc. This requires line 3120 added to KYUTIL to modify field 6 thus:

```
3120 AX(6) = 3: AY$(6) =
      "EST/3"
```

This limits the length of field 6 when used in the key to 3 characters and changes the field label to remind the operator of the limit. You

can modify any field in this way - AX() contains field lengths and AY\$() field labels. Remember that you still have the full field available for the conditions. (I have also done a mod there, but I'll save that for another time.)

FINDING DATA

We have a file of salary scales which has some 30 groups under one or two letter codings - DO, E, PP, PS, TP etc. Each group scale has up to 19 grades and each grade has up to 12 steps. The first field of each record is the group, grade and step, the second is the actual salary. The top step of a scale is of course the maximum available for that grade. The file is created in descending order of grades and steps. In practice we usually require to find the salary and salary maximum for a given grading. To speed up the FIND process separate the grading field, for example TP10/05 into three parts thus: group=TP;group+grade=TP10; and group+grade+step=TP1005. Assuming that the data file is open on channel 1, then first FIND"TP". This is found in a header of 30 entries and once found INPUT the index immediately beside it, reset channel index to that figure, which is where the "TP" grades begin. Then FIND"TP10" and INPUT the salary maximum - i.e. the first salary in TP10 grade is the maximum, then without resetting the index, FIND"TP1005" and INPUT the salary for that step. Now print the salary in the format \$9000/9500 which means salary \$9000, maximum \$9500. With over 60 entries in some of the salary scales this gives a valuable saving in time in reports involving several hundred records by leaping through the file to the right

area. An alternative is to read the 30 groups and indexes into an array to avoid the need for the first FIND. I'm sure you could apply this same principle to other data files which exist in clearly defined groups. Listing 5 is a program extract showing some of the routines.



A HARDWARE SMORGASBORD

By: Eugene E. Baldwin
10650 North 75th Street
Longmont, CO 80501

More ClP modifications? Yep - some of us just cannot leave well enough alone. My OSI just celebrated its first birthday this spring and we are becoming better acquainted by now.

The unit was purchased in a nearby city and handed to me in the carton as may have often been the case. Everyone is aware of the older problems due to poor instruction books, factory indifference, and others. (Fortunately this has changed a few months ago and new manuals are very nice, the factory actually talks to us, and many of the dealers are quite good!) What I did to my unit follows.

The first problem was trying to load from the cassette. Sound familiar? After the dealer couldn't help, I played with all kinds of ideas and again proved the value of magazines such as this to obtain help and find out what others were doing. The present cassette input is shown in fig. 1.

Shielded wire was used from the input/output jacks to the Molex connector J2 with a resistor across the input jack to present a load to the cassette recorder output stage. A small A.C. voltmeter was mounted in the panel to monitor the level of incoming data and makes it much easier to set various recorders up the first time. Parallel with this is a small speaker with a volume control. The total resistance is high enough to avoid loading, but gives a very good indication of type of data coming in. A good tape has a very distinct sound to it, and if the recorder or tape are not good, one can often recognize a problem.

A few recorders oscillate when both cables are connected, but

BE A KEY WITNESS

M/A-COM OSI, the innovator in micro-computer-based systems, invites you to witness the unveiling of our KEYFAMILY Systems at NCC '82.

Before your eyes, the KEYFAMILY will unlatch new capabilities for hardware expansion, for software compatibility, and for dynamic growth from single-user to multi-user to networking.

And whether you're a business computer user or business computer dealer, our complete line of new personal computers and work stations, time-sharing and multi-processing computers and networking systems will open your eyes to a new world of possibilities.

Be a key witness. See us in Booth 1111 and make the right decision.

The KEYFAMILY from



M/A-COM Office Systems, Inc., Seven Oak Park, Bedford, MA 01730, (617) 275-4440.

most work fine. I have done 2 ClP's and can use any of five different cassette players. A bad load is very uncommon any more, even at the higher baud rate to be mentioned later. A simple trick to avoid confusion between the I/O cables is to paint the ends, put a colored piece of tape on them, etc. I cut a small piece of shrink tubing in black and red and placed it over the plugs with matching colors at each end, then marked the jacks with felt-tip pens of the same color.

Next in the line was the R.F. I. (Radio-Frequency Interference) problem. The metal case was one factor in choosing the Cl-P originally. I removed the original power cord and using a "nibbler" tool enlarged the hole enough to accept a commercial line filter. This was installed, being sure a good ground was made to the case. This requires purchase of a matching power cord or two. In the same area, a small A.C. outlet was installed, and connected to the switched side of the line. This outlet was bypassed with a pair of .003 MMF disk ceramic capacitors, (again for

R.F.I. protection) while at the power switch a varistor G.E. # V130LA10A was placed across the line for some transient protection. While working in this area, I went ahead and installed a cooling fan in the holes OSI was good enough to provide. While most 3" fans would work, my choice was a PAMOTOR Model 8500D which is very quiet and thin enough that the main circuit board does not contact it. The fan will only be needed if extra memory, etc. is added. The factory power supply is a nominal 3.5 Amp. unit which has been modified with increased input capacitors and then re-rated at 4.5 amp. While it works well at the 4.5 Amp. level, too much heat buildup occurs and the fan really does the job.

By this time the bottom cover had been removed several times and the screws were showing the wear and tear. The next step was to obtain a dozen 8-32 x 1/2 screws with knurled heads, permitting removal without a screwdriver. This idea was also used to hold the main circuit board in place, with the added touch of a small nylon washer between the

QUALITY SOFTWARE FOR TRS-80 COLOR AND OSI



ADVENTURES!!!
For TRS-80 COLOR and OSI. These Adventures are written in BASIC, are full featured, fast action, full plotted adventures that take 30-50 hours to play. (Adventures are inter-active fantasies. It's like reading a book except that you are the main character as you give the computer commands like "Look in the Coffin" and "Light the torch.")

Adventures require 16K on COLOR-80 and TRS-80. They sell for \$14.95 each.

ESCAPE FROM MARS
(by Rodger Olsen)

This ADVENTURE takes place on the RED PLANET. You'll have to explore a Martian city and deal with possibly hostile aliens to survive this one. A good first adventure.

PYRAMID (by Rodger Olsen)
This is our most challenging ADVENTURE. It is a treasure hunt in a pyramid full of problems. Exciting and tough!

TREK ADVENTURE (by Bob Retelle)
This one takes place aboard a familiar star-ship. The crew has left for good reasons — but they forgot to take you, and now you are in deep trouble.

NEW!!
CIRCLE WORLD — We got Kzinti and puppeteers and problems. Our newest and biggest adventure. Requires 12k on OSI and 16K on TRS-80 Color.

NUCLEAR SUB — You are trapped in a nuclear sub at the bottom of the ocean. Escape and even survival is in doubt. Plotted by three of the most sadistic — I mean "creative" minds in adventure programming.



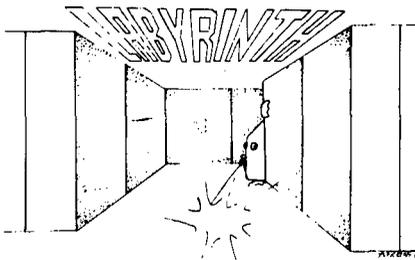
VENTURER!—A fast action all machine code Arcade game that feels like an adventure. Go berserk as you sneak past the DREADED HALL MONSTERS to gather treasure in room after room, killing the NASTIES as you go. Great color, high res graphics, sound and Joystick game for the TRS-80 Color or OSI machines. (black and white and silent on OSI.) Tape only. \$19.95.

BASIC THAT ZOOMS!!
AT LAST AN AFFORDABLE COMPILER FOR OSI AND TRS-80 COLOR MACHINES!!! The compiler allows you to write your programs in easy BASIC and then automatically generates a machine code equivalent that runs 50 to 150 times faster.

It does have some limitations. It takes at least 8K of RAM to run the compiler and it does only support a subset of BASIC—about 20 commands including FOR, NEXT, END, GOSUB, GOTO, RETURN, END, PRINT, STOP, USR(X), PEEK, POKE, *, /, +, -, X, X =, VARIABLE NAMES A-Z, A SUBSCRIPTED VARIABLE, and INTEGER NUMBERS FROM 0 - 64K.

TINY COMPILER is written in BASIC. It generates native, relocatable 6502 or 6809 code. It comes with a 20 page manual and can be modified or augmented by the user. \$24.95 on tape or disk for OSI or TRS-80 Color.

LABYRINTH — 16K EXTENDED COLOR BASIC — With amazing 3D graphics, you fight your way through a maze facing real time monsters. The graphics are real enough to cause claustrophobia. The most realistic game that I have ever seen on either system. \$14.95. (8K on OSI)



PROGRAMMERS!
SEE YOUR PROGRAM IN THIS SPACE!! Aardvark traditionally pays the highest commissions in the industry and gives programs the widest possible coverage. Quality is the keyword. If your program is good and you want it presented by the best, send it to Aardvark.



QUEST — A NEW IDEA IN ADVENTURE GAMES! Different from all the others. Quest is played on a computer generated map of Alesia. Your job is to gather men and supplies by combat, bargaining, exploration of ruins and temples and outright banditry. When your force is strong enough, you attack the Citadel of Moorlock in a life or death battle to the finish. Playable in 2 to 5 hours, this one is different every time. 16K COLOR-80 OR TRS-80. ONLY \$14.95.



SPACE ZAPPER — Protect your central Star Base from ships that attack from all four sides. Fast reflexes are required as the action speeds up. Great for kids or Dads. This game has high speed high resolution graphics and looks as if it just stepped out of the arcades. — 16K extended or 32K disk. BASIC TRS-80. Color only. \$14.95.

Please specify system on all orders

This is only a partial listing of what we have to offer. We have arcade and thinking games, utilities and business programs for the OSI and TRS-80 Color. We add new programs every week. Send \$1.00 for our complete catalog.



TRS 80 COLOR **AARDVARK - 80** **2352 S. Commerce, Walled Lake, MI 48088**
(313) 669-3110

OSI

screw head and the circuit board. Since I carry the unit back and forth to work a lot a set of stick-on rubber feet was placed on one side and a folding handle on the other side. With the removable cords it now becomes quite portable. Rubber feet on the bottom prevent any sliding around and help guarantee air circulation.

Cassette control consists of a miniature center-off toggle switch which energizes either of two 5 volt reed relays whose contacts are brought out of the rear panel but insulated from it. The diagram is shown in fig. 2. I wanted to use the RTS line for other things, this for me at least, works well.

Small red light-emitting diodes make good indicators while requiring little current so one was mounted above the keyboard and connected via a 180 ohm current-limiting resistor to the 5 volt supply and serves as a power "on" indicator. A second one was mounted just right of the keyboard and will serve to indicate that the SHIFT LOCK key is in the depressed position.

RS-232 interfaces are quite common and very useful so the next item completed was installing the "D" connector (25 pin, female) in the opening provided on the rear panel. A miniature 2 pole-3 position switch was installed above and to the right of the keyboard.

Perhaps you have noticed - but so far the main circuit board has not been tinkered with! So much for that. OSI includes all RS-232 circuitry on the board and it is shown in their service manual with dotted lines for the traces and all parts shown. It's just that the owner must install these parts. This is easy so I'll not dwell on this. I do strongly suggest that sockets be used for any added I.C.s to make replacement a lot easier. Many pieces of gear will operate without the -9 volts or so, but a tiny supply may be made or purchased to convert the +5 volts to -9 to -12 at a few ma. of current. I bought a ready made one for less than fifteen dollars from a surplus house and mounted it on the blank power supply area with a small dab of RTV. Short lengths of wire connect to ground, the +5 buss, and the -9 connection which is PIN 7 of 13.

Detailed step by step instruc-

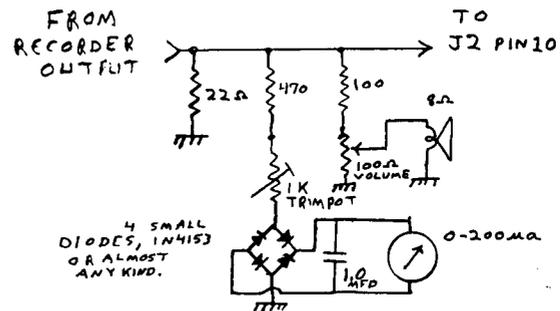


FIG. 1 AUDIO INPUT ADDITIONS

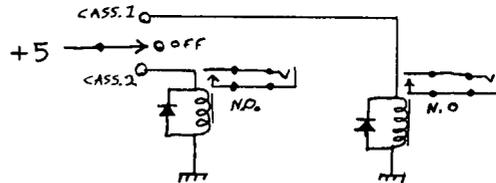


FIG. 2 CASSETTE CONTROL

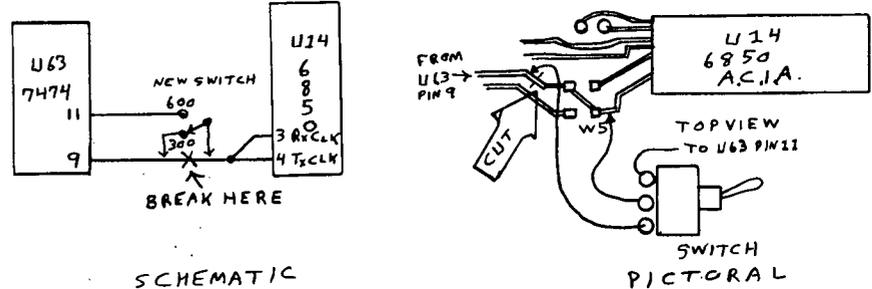


FIG. 4 BAUD RATE MODIFICATION

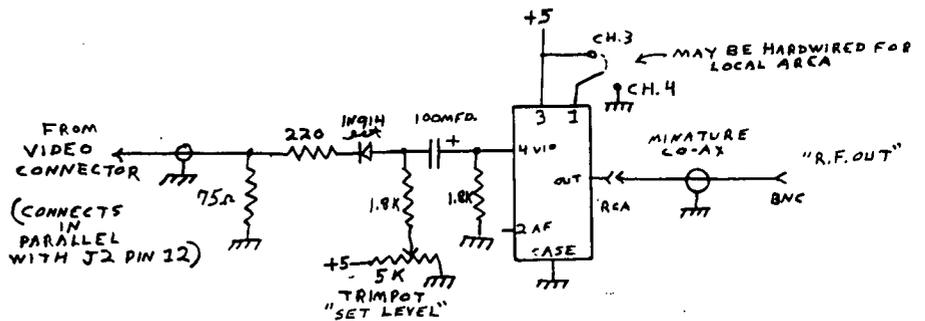


FIG. 4 R.F. MODULATOR

tions have appeared in this and other magazines and may also be obtained from Aardvark Technical Services, 2352 S. Commerce, Walled Lake, MI 48088, for a nominal sum. They have an excellent offering for the OSI fan and even the catalog is interesting reading.

Liberal use was made of rub-on lettering to number the jacks, the foil side of the I.C.

sockets, and the new parts on front and rear panels. Polaroid (T.M.) print coater is good protection for these.

A small, but useful, change was to replace the video output connector, formerly a "RCA type" with a BNC type as many commercial monitors use these (and my ham station does too!). A "T.V. modulator" kit from Radio Shack was stripped and the small shielded R.F.

portion mounted inside the rear panel and the output of this on channel 3 or 4 is connected via 50 ohm miniature coax cable to another BNC connector. This way a modulator is always available if needed but the higher quality picture makes a good monitor desired. The schematic diagram is shown in fig. 3.

Many articles and ideas on increased baud rate have appeared, but my choice was one in a letter by Curtis Preston of Shalimar, Florida and is sketched in fig. 4. A single pole-double throw switch has been mounted just left of the Challenger 1P logo plate with short leads down to the ACIA chip area.

A small foil cut is made near W5 next to U24, the 6850 A.C.I.A. chip. This cut (fig. 4) breaks the 300 baud clock line between pin 9 of U63 which is a 7474 flip-flop and pins 3 and 4 of U14. Pin 11 of U63 has a signal two times the frequency of that on pin 9 or 600 baud! The switch either restores the line to the A.C.I.A. at 300 baud or connects it to pin 11 of U63 at 600 baud.

I.C. sockets and jacks may be used as is or cut into sections for form disconnect points and liberal amounts of

colored wire help keep things in order now and if notes are taken and saved one will still make sense of it a long time later.

More memory soon was needed but the 610 board seemed a bit much so the 8K memory with I/O ports was ordered from Aardvark. Assembly was quite easy as complete instructions are furnished. The only problem was a typo in the memory test program. A phone call to Mr. Olsen resolved this and it was up and running with 16K total. The mounting consisted of small screws and spacers thru the top of the cabinet with two #22 wires for +5 and ground leading to the power supply and a 12" ribbon cable with 40 pin DIP plugs on each end to connect to J1 on the main board. The I/O connections have not been made yet.

Lack of a cursor control and problems attempting to correct mistakes without retyping a complete line became rather frustrating and led some of us to order a chip called a C1S from Aardvark. This is a replacement monitor ROM for the C1S at less than \$40 and is chock full of goodies! All kinds of nice editing features such as insert, delete or change a character, cursor control, real backspace, support for video modifications

and much more. They also sell a C1E/C2E for a little more money which even includes an extended machine code monitor. This probably is what I should have purchased since it was not much more expensive. We considered the chip to be money well spent. All it usually takes is just plug it in and follow the simple instructions which are furnished.

Lots of time was spent studying ideas for more characters per line using various video modifications. Several things were considered including cost, complexity, number of foil cuts to be made, pros' and cons' of "piggybacked" chips, and vendor support. Some study decided that two of us would try different approaches so one of my friends purchased a kit from a source in Canada, while I chose the "expensive" one from Aardvark as it appeared to me to be more simple and I'd had good luck with their items in the past. My total cost was about twelve dollars including a new crystal. A full set of instructions is furnished.

Since the C1S chip directly supports the video change, no "video driver" cassette is involved. Just turn the unit on and start using it. Everything can be controlled from

OSI Disk Users

Double your disk storage capacity Without adding disk drives

Now you can more than double your usable floppy disk storage capacity—for a fraction of the cost of additional disk drives. Modular Systems' DiskDoubler™ is a double-density adapter that doubles the storage capacity of each disk track. The DiskDoubler plugs directly into an OSI disk interface board. No changes to hardware or software are required.

The DiskDoubler increases total disk space under OS-65U to 550K; under OS-

65D to 473K for 8-inch floppies, to 163K for mini-floppies. With the DiskDoubler, each drive does the work of two. You can have more and larger programs, related files, and disk utilities on the same disk—for easier operation without constant disk changes.

Your OSI system is an investment in computing power. Get the full value from the disk hardware and software that you already own. Just write to us, and we'll send you the full story on the DiskDoubler, along with the rest of our growing family of products for OSI disk systems.

Modular Systems

Post Office Box 16 D
Oradell, NJ 07649.0016
Telephone 201 262.0093

™DiskDoubler is a trademark of Modular Systems.

Business System Users

Get Mainframe Software features with
“Computer Business Software”
“CBS”

the **INSTALLABLE** Accounting Package
for Floppy Disk and Hard Disk Systems -
Single or Multi User

Featuring: Accounts Receivable with Sales Analysis
Inventory with Product Analysis
Order Entry with Invoicing
Accounts Payable with Checkwriting
General Ledger with Financial Statements
Payroll with Labor Distribution

- **Thoroughly Documented**
- **Already Installed**
- **Fully Supported**
- **Ready Now**

Call **1-800-843-9838** for your free documentation kit, price schedule and sales brochure.

MICRO SOFTWARE
INTERNATIONAL

3300 South Madelyn, Sioux Falls, South Dakota 57106
“Mainframe Software for your Micro”

the keyboard.

After this much was typed, I got a good buy on a 610 board complete with power supply so the Aardvark board was removed and sold to a friend owning a Superboard 2. He has used it for some time now and is very pleased with it.

With 32K RAM, I took the plunge and ordered my first disk drive from OSI. A home built case and power supply provide for a second drive soon.

Imagine my surprise to find the disk would not work! After some digging, I found that the previous video mod. caused wrong frequencies to be fed to the disk controller circuit. So I removed the mod. and went back to the old display. Further work will be required, since the better display is desirable.

Even without these changes, OSI has come to be one of the biggest bargains around and, since their purchase by M/A-COM, I feel they will be a potent contender in the personal and home computing field even more.

As you may guess, the initial let down has been reversed and much of this has been due to OSI themselves, the folks at Aardvark, PEEK (65), and those brave souls who write for publications such as this. Next comes a printer and modem, as we (the computer and myself) become best friends.

I can't promise any of these ideas will work for you, but they have done fine for me. By using published as well as commercial ideas there is little chance of failure and often a letter is enough to get any needed help. Naturally, not all mods need be done at once. More memory takes more power which in turn takes more cooling and almost any change in the video display will require some video driver change, but in this case the CIS chip took care of that even before the change was made. For the hardware person, these are all easy things to do, but my advice to others is to be sure of your ability then go ahead. Perhaps something like the feet and handles might be all you wish to do at this time. No matter what one does in the way of repairs or modifications, the important thing is to KEEP A RECORD! A week or a year later it will be the most important thing in your file.

Also, in case of sale or perhaps trade-in once we get that far, proper documentation will be a very valuable item.



AUTOMATIC HEAD LOAD/UNLOAD FOR OS65DV3.2 MINI-FLOPPY SYSTEMS

by: Owen R. Brock
Route 5, Box 911
Burleson, Texas 76028

For all you OS65DV3.2 users who have been complaining about the fact that the MF version of OS65D does not provide disk drive head load/unload, here is an operating system patch that will automatically load and unload the disk drive head each time the drive is accessed.

This patch requires only eleven bytes of memory, and can be loaded at any convenient location. Since OS65DV3.2 has some unused memory in the additional page required by the MF version, location \$326E was chosen.

The following assembly listing shows all of the changes to the system required to install the patch:

```
26BC 4C6E32 JMP $326E
2754 A920 LDA #S20
2756 0D02C0 ORA $C002
2761 A9DF LDA #SDF
2763 2D02C0 AND $C002
27C0 D094 BNE $2756
326E 85FA STA $FA
3270 205427 JSR $2754
3273 A5FA LDA $FA
3275 48 PHA
3276 4CBF26 JMP $26BF
```

For those who would prefer to poke these changes into memory when BEXEC* is run, the following can be added to BEXEC* to affect the required changes:

```
10 POKE9916,76:POKE9917,110
20 POKE9918,50:POKE10069,32
30 POKE10070,13:POKE10082,223
40 POKE10083,45:POKE10177,148
50 FORI=12910TO12920
60 READA:POKEI,A:NEXT
70 DATA 133,250,32,84,39,165
80 DATA 250,72,76,191,38
90 DISK!"GO 2761"
```

These changes to OS65DV3.2 accomplish the head load and unload function by toggling bit five of the floppy disk control word. Since this is the side select bit, these changes will not work if your system includes disk drive C or D, but for single or dual drive systems it will work without difficulty. That is to say that I have not encountered any problems with it.



Try it and see if you like all those strange clicking sounds coming from your disk drive.

ADD 2K OF MEMORY TO YOUR OSI VIDEO SYSTEM

By: Jeff Easton
9416 Washington
Brookfield, IL 60513

The OSI 540B video board has two 2K blocks of memory strapped at \$D000 and \$E000 for video and color respectively. The video memory is eight bit wide memory while the color memory is only four bits wide. This article will show you how to add the additional four bits of color memory enabling this space to be used for machine language subroutines, temporary program storage, indirect file space, etc. The color is still operational of course.

You will need three IC chips and a foot or two of wire. The IC chips needed are two 2114L memory chips and one 8T26 bus driver chip.

Refer to the schematics of your 540B board. At the top of figure two, shown below, is the schematic of the color memory. If you have the Sams manual, the schematic is in the upper right corner of page 36. The three chips we are interested in are U1H, U1F, and U1G. We will exactly duplicate this portion of the circuit but for the upper four data bits D4 thru D7.

The easiest way to add the three additional chips is to piggyback them to U1F,G and H soldering down all leads except those associated with the data lines. This way we can pick up all of the enable, address, and power lines easily.

Here are the steps needed to add the three chips:

- 1) Open your computer and remove the video board. Disconnect the ribbon cable that goes to the keyboard and the molex connector that goes to the phono jacks at the back of the computer.

- 2) Place the video board in front of you with bus connector facing to the left. The three chips to be piggybacked are at the lower left corner in a horizontal row. See figure 1 below. Remove these

Z-FORTH IN ROM by Tom Zimmer

5 to 10 times faster than Basic. Once you use it, you'll never go back to BASIC!
source listing add

\$ 75.00
\$ 20.00

OSI FIG-FORTH True fig FORTH model for OS65D with fig editor named files, string package & much more

\$ 45.00

TINY PASCAL Operates in fig-FORTH, an exceptional value when purchased with forth.

TINY PASCAL & documentation
FORTH & TINY PASCAL

\$ 45.00
\$ 65.00

SPACE INVADERS 100% machine code for all systems with 64 chr. video. Full color & sound on C2, 4P & 8P systems. The fastest arcade program available.

\$ 14.95

PROGRAMMABLE CHARACTER GENERATOR

Use OSI's graphics or make a complete set of your own! Easy to use, comes assembled & tested.
2 Mhz. boards

\$ 99.95

\$109.95

PROGRAMMABLE SOUND BOARD

Complete sound system featuring the AY-3-8910 sound chip. Bare boards available.

\$ 74.95

\$ 29.95

32/64 CHARACTER VIDEO MODIFICATION

Oldest and most popular video mod. True 32 chr. C1P, or 32/64 chr. C4P video display. Also adds many other options.

\$ 39.95

ROMS!!!

Augment Video Mod with our Roms. Full screen editing, print at,selectable scroll, disk support and many more features.

Basic 4 & Monitor
Basic 3
All 3 for

\$ 49.95

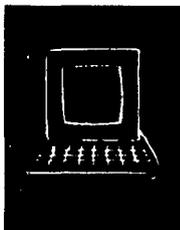
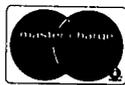
\$ 18.95

\$ 65.00

65D DISASSEMBLY MANUAL. by Software Consultants
First class throughout. A must for any 65D user.

\$ 24.95

NUMEROUS BASIC PROGRAMS, UTILITY PROGRAMS AND GAMES ALONG WITH HARDWARE PROJECTS. ALL PRICES ARE U S FUNDS. Send for our \$1.50 catalogue with free program (hardcopy) Memory Map and Auto Load Routine.



OSI Software & Hardware

3336 Avondale Court
Windsor, Ontario, Canada N9E 1X6
(519) 969-2500

3281 Countryside Circle
Pontiac Township, Michigan 48057
(313) 373-0468

progressive computing

chips and place them in conductive foam.

3) Take the three chips to be added and bend out the following pins on each about 45 degrees. 2114L's: bend out pins 11 thru 14. These are the data pins. 8T26: bend out pins 2 thru 7 and pins 9 thru 14. The only remaining unbent pins should be the power and data direction pins (1,8,15 and 16).

4) Now take these three chips and piggyback them to the chips you removed from the video board observing pin 1 locations on each. Tack all unbent pins of top chip to corresponding pins of bottom chip with solder. Reinstall chips.

5) Now we have to pick up the upper four data lines from the board and route them to the piggyback buffer chips input pins. See figure 2 for the locations of the upper four data lines. Notice that you will have to drill one hole in the board to pick up D4 from a foil running on the underside of the board. Drill the hole close to the foil coming from molex connector pin #9, this is D4. The other data lines have holes readily available. Run wires from these holes to pins 3,6,10 and 13 of the 8T26. It doesn't matter which data line goes to which input pin of the chip as long as you can keep the inputs and outputs of the chip straight. For simplicity I would recommend the same wiring convention as shown in the schematics, but substituting D4 for D0 and D5 for D1 etc.

6) Now wire the 2 output pins for each data line together. The 2 output pins are on either side of the input pin. Example: if D4 goes in on pin 3 then the D4 output pins are 2 and 4 and should be wired together. Also, these two pins must be wired to a data input pin on the RAM chip. Wire the two RAM chips in parallel.

7) After you have all four data lines wired, plug the cables back into the video board and plug the video board back into the computer.

To make sure that you wired everything correctly, load the Extended Monitor and fill \$E000 thru \$E7FF with \$FF and then dump those locations. If those locations aren't \$FF then you either made a wiring error or you burned out an IC by leaving the soldering iron

Figure #1

540B Board.

↑ BACKPLANE ↑

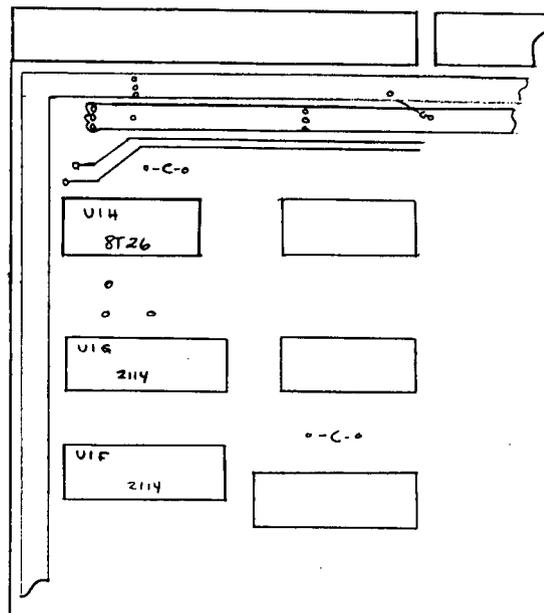
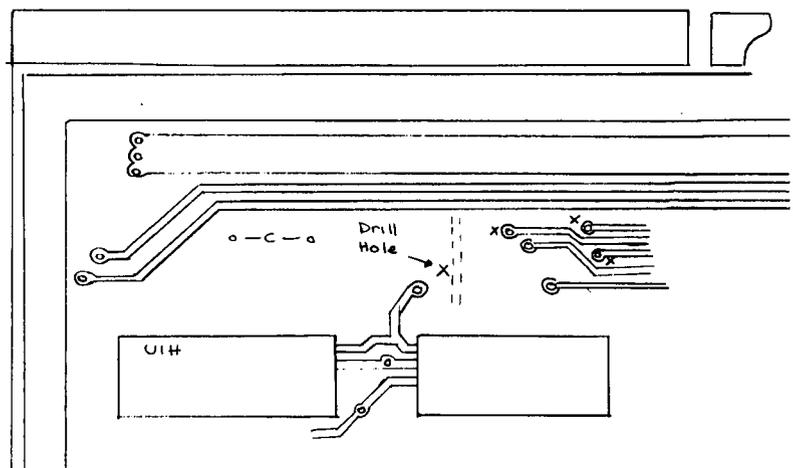


Figure #2

Data Bus Connections

X = hole for data line

↑ BACKPLANE ↑



on the legs too long. Go back and check your work.

I made this circuit mod to my computer and it works fine. It took me about three hours to build the circuit. I have found that this is a convenient place to put my machine language subroutines like screen clear, etc. And as long as you leave the computer on, the program will stay there because this area of

memory is protected from BASIC.

As you probably already figured out, you can add another 2K at \$E800 thru \$EFFF because this area is unused. Also, you could add a third 2K block at \$F000 thru \$F7FF depending on whether you had any peripherals addressed here or not. These would be harder to implement because address decoding and board space is not

readily available.

One final note, OS65D 3.3 seems to take control of the color memory for special color printing functions so it will erase whatever you store there when 3.3 is booted. This doesn't occur with versions 3.1 or 3.2. I haven't found a way to disable this feature as of yet but maybe someone out there has. Any ideas?



CASSETTE CORNER

By: David A. Jones
8902 SW. 17th Terrace
Miami, FL 33165

If you have wanted to dabble in assembly language programming but haven't been sure just where to jump in or were waiting for an easy but still useful program to start on, I offer the following hexadecimal to decimal conversion routine to get your feet wet.

Most of the work is done by existing routines in the ROM monitor or ROM BASIC. All we have to do is the necessary housekeeping prior to branching to the ROMs. Five existing routines are used, 4 of which were mentioned in last month's column. The 5th, which I call 'CONV' is in the BASIC ROM and converts a 16 bit binary number into decimal and displays it on the screen.

When we talk about binary we are really talking about the only number system the computer understands. All other numbering systems are used by humans to keep numbers at a comprehensible level and must be converted or translated to binary before the processor can work with them. Octal, decimal and hexadecimal are the 3 most common and all can be represented in binary. Octal 010243 is easier for us to recognize than 0001000010100011. This same number in hex is 10A3 and in decimal is 4259. Usually hexadecimal numbers are preceded with a dollar sign (\$). If we split the above number in half, take the first part (00010000) and call it \$10 and load it into the accumulator of the 6502 and the second half (10100011) call it \$A3 and put it in the X register of the 6502, then 'convert' will translate the number to decimal and print it on the screen for us.

When the computer is running and waiting for an input from

```

10 0000 ;HEXX, 6-15-81
20 0000 ;BY DAVID A. JONES
30 0000 ;REVISED 4-15-82
40 0000 ;
50 0222 *=$0222
60 0222 TEMP =$FC      TEMPORARY ACCUMULATOR
70 0222 XTMP =$FF     TEMPORARY X REGISTER
                        STORAGE
80 0222 CONV =$B95E
90 0222 LEGAL =$FE93
100 0222 ROLA =$FE93
110 0222 INKEY =$FEED
120 0222 INPUT =$FFBA
130 0222 BASOUT=$FFEE
140 0222 ;
150 0222 C908      CMP #$08      IS IT CONTROL H?
160 0224 F003      BEQ HEXX     BRANCH TO HEXX IF IT IS
170 0226 4CBAFF    JMP INPUT    REVERT TO NORMAL FLOW
                        IF NOT
180 0229 ;
190 0229 A924      HEXX LDA #'$      LOAD PROMPT
200 022B 20EEFF    JSR BASOUT   DISPLAY IT
210 022E A204      LDX #4       LOAD COUNTER
220 0230 20EDFE    INNN JSR INKEY   GET CHARACTER
230 0233 20EEFF    JSR BASOUT   DISPLAY IT
240 0236 2093FE    JSR LEGAL   MAKE IT A HEX NIBBLE
250 0239 86FF     STX XTMP    SAVE COUNTER
260 023B A200     LDX #0     PREPARE X FOR ROLA
270 023D 20DAFE    JSR ROLA    ROL NIBBLE INTO TEMP
                        ACCUMULATOR
280 0240 A6FF     LDX XTMP   RESTORE COUNTER
290 0242 CA       DEX          COUNT CHARACTER
300 0243 D0EB     BNE INNN   GET NEXT CHARACTER IF
                        MORE TO COME
310 0245 ;
320 0245 A6FC     LDX TEMP   PUT LOW BYTE IN THE X
                        REGISTER
330 0247 A5FD     LDA TEMP+1  AND HIGH BYTE IN THE
                        ACCUMULATOR
340 0249 205EB9   JSR CONV   CONVERT THE NUMBER
350 024C A90D     LDA #$0D   LOAD CARRIAGE RETURN
360 024E 20EEFF   JSR BASOUT DISPLAY IT
370 0251 A90A     LDA #$0A   LOAD LINE FEED
380 0253 20EEFF   JSR BASOUT DISPLAY IT
390 0256 60       RTS          BACK TO CALLING ROUTINE
400 0257 ;

```

the user, the program flow is through a vector stored at \$0218 which directs it to the input routine at \$FFBA. If we redirect the program flow to check for a control H before going on to \$FFBA, we can call our program automatically. POKE 536,34:POKE 537,2.

Refer to the following listing as you read the explanation. If a control H was indeed typed we branch to HEXX, otherwise we continue on to INPUT at \$FFBA. This check and branch is done in line 150 and 160. If we take the branch we should let the user know that an input is expected and display a prompt to let him know what kind of input. Lines 190 and 200 do this.

We want 4 hex characters to be typed in so we set up the X register to keep track of how many have been entered so far. After setting the X register we get the first character with the JSR INKEY at line 220 and then immediately display it with the JSR BASOUT instruction. Numbers from the

keyboard or serial port appear in a code called ASCII which uses 7 bits to represent each character. We are only interested in numbers contained in the hexadecimal set so 4 bits are sufficient and we can strip off the excess with the routine LEGAL.

ASCII representation of the number 1 is 0110001 in binary which we can split into 2 groups 011 and 0001. All of the information we need is contained in the second group or nibble so we can discard the first and return from LEGAL with 00000001 in the accumulator. This works for 0 to 9. If the character is a letter in the range of A to F we still strip off the first nibble but then add 1001 (9) to the second nibble to cover all of the possible combinations from 0000 to 1111. If the input character is not a legal hex character, an M for example, LEGAL sets the first bit to a 1 which is the machine's way of expressing a negative number. A test could be made to branch back and get

OSI COMPATIBLE PRODUCTS

56K 2-MHz Ultra Low Power CMOS Static Memory Board MEM-56K \$850

Partially Populated Boards (Specify address locations required) ... MEM-48K \$750
 MEM Board uses the new 2K-Byte Wide Static RAM chips which are MEM-32K \$550
 2716 EPROM compatible. Any 2K byte memory segment can be MEM-24K \$450
 populated with RAM or EPROM (or left empty for use of Address Space MEM-16K \$350
 by another board). Fully expandable to any memory size you will ever MEM- 8K \$250
 need. No special addressing requirements, just solder in extra sockets MEM- 4K \$200

Extra 2K RAM Memory Chip \$24

Optional Parallel Printer Port -P \$120

Optional Calendar/Clock Software available in EPROM -T \$ 25

Both options (Disk software mods provided for use of 6522 V/A on printer). -PT \$125

EXAMPLE USES:

C4P & C8P: Expansion to 4K RAM of Basic workspace.
Parallel Printer Port — Reserve Serial Port for MODEM
Calendar/Clock Displaying on unused portion of screen.
 Space for 5.75K of **Enhanced System Monitor EPROMS**.

All of this on 1 Board, using only one of your precious slots. Software for Enhanced System Monitor capabilities is continuously being developed and improved. As new EPROM Monitors are available, you may upgrade to them for any price differential plus a nominal \$10 exchange fee. Another possibility is to fill any portion of the memory with Basic Programs in EPROM for **Power-on Instant Action**. This custom EPROM programming service is available at \$25 per 2716 (Includes EPROM). Extra copies at \$15 for each EPROM.

C4P-MF & C8P-DF:

Memory expansion to 48K.
 Add 6K Memory above BASIC for special software requirements.
Parallel Printer Interface and/or **Displaying Calendar/Clock**.
 Add 1.75 K **Enhanced System Monitor ROM**.

C3: Up to 56K of Memory Expansion — can be addressed for Multiuser.
 (Optionally, each user can have his own **Dedicated Printer Port**).

C1P, C4P & C8P FLOPPY DISC CONVERSIONS:

Memory/Floppy Board (Includes MI48P1 ROM) MEM F-16K \$450
C1P-600 Board Adapter & Cable A600/48 \$ 50
Additional Memory/Printer/Times (See MEM Board Prices)
5 1/4" Drive/Case/Power Supply & Cable to MEMF Board FD5 \$399

IEEE-488 INTERFACES AND SOFTWARE:

The General Purpose Instrumentation Bus (GPIB Controller interface is available for all OSI Computers. Machine code GPIB Drivers are linked to Basic to provide easy control of IEEE-488 instruments which is equal to the best of Hewlett-Packard Controllers and far superior to most others. Basic Commands for Serial Poll, Parallel Poll, IFC Clear, full Local/Remote Control, Respond to SRQ Interrupts, Send Trigger, do Formatted Input/Output, Direct Memory Input/Output and MORE. Interface includes IEEE-488 Ribbon Cable/Connector.

GPIB Controller Interface for C2, C3, C4 and C8 Systems GPIB 4-488 \$395

GPIB Software for OS-65D (Add -8 for 8" or -5 for 5") GPIB 488-D \$ 70

GPIB Software for OS-65U GPIB 488-U \$100

GPIB Software on two 2716 EPROMS for ROM based systems GPIB 488-R \$100

Add Optional **Parallel Printer Interface** to GPIB 4-488 -P \$120

Add Optional **Calendar/Clock** to GPIB 4-488 -T \$ 25

Add 2K RAM to GPIB 4-488 (Specify location, \$4000-\$8FFF & \$D000-\$EFFF available) -M \$25

GPIB Controller for C1P, includes Software, Clock, All Features of ROMTERMS, & space for 6K EPROM. GPIB 6-488R \$395

Add Optional **Parallel Printer Interface** to GPIB 6-488R -P \$120

EPROMS:

C1P ROM with 24/48 Col Display for Series II, Smart Terminal, Line Editing, Corrected Keyboard Screen Clear and More ROM-TERM II \$59.95

C1P ROM with 24 Col Display, Other ROM-TERM II Features, Disk Boot, and ROM/Disk Basic Interchange ROM-TERM \$59.95

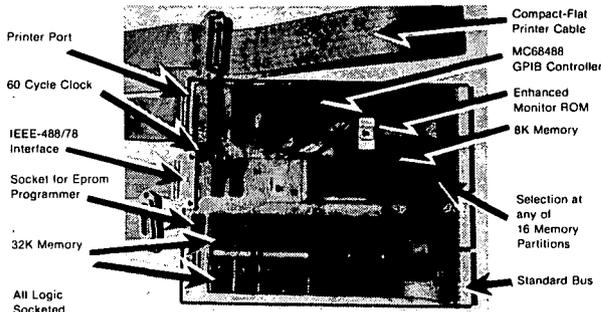
C4P-MF/C8P-DF Disk warm start, changed IRQ Vector and just flip switch for Serial or Video System with Corrected Keyboard SYNKEY \$39.95

ENHANCED MONITOR ROMS FOR USE ON GPIB 4-488 & MEM BOARDS:

Expanded Support for C4P & C8P Featuring Calendar/Clock, Line Edit, Smart Terminal, Memory Files, Parallel Printer Control, Corrected Keyboard, All Features of ROMTERMS, Disk Support with Warm Start and More MI48P1 \$59.95

Expanded C2 Monitor with Calendar/Clock Software, Hard Disk Boot, Warm Start and Multi-User Control for C2 Systems MIC2-1 \$59.95

IEEE-488 CONTROLLER INTERFACE



THE GPIB 4-488 INTERFACE BOARD CONVERTS ANY OSI COMPUTER INTO AN IEEE-488 INSTRUMENT BUS CONTROLLER!

BENEFITS — Provides a Sophisticated Instrumentation Controller at very low cost (often saving thousands of Dollars). The combination of IEEE-488 Instrumentation Controller and High Capacity Hard Disk file storage available on OSI Computer systems is available at a fraction of the cost required by the nearest competitor. The IEEE-488 Bus, also known as the GPIB, HP-IB or IEC-625 is the most popular International Standard for connecting instrumentation systems. This 16-line bus is designed to interconnect and control up to 15 instruments at a time. Currently, over 2000 different instruments are available to work on this bus. They include: Plotters, Digitizers, Printers, Graphic Displays, Recorders and a multitude of specialized Test/Masurement/Control Equipment.

EPROM-ABLE — Can be used with a C4-P to create a dedicated IEEE-488 controller.

C2-D MULTIPLE USER SYSTEMS

SAVE — 2 and 3 user Time Sharing Systems are available on the C2-D Winchester Disk Computer at a considerable cost savings from C3 Multiple User Systems. The 3 user C2-D System can be expanded to include a word processing printer, 4 other parallel printers and 3 serial printer interfaces.

COMPATIBLE — The special C2-D Multi-User Executive Program is 100% compatible with OS-65U V1.4. The Multi-User Real Time Clock, Memory Partition Control and IRQ Interrupt Management are done on the Micro Interface Memory Board. Thus, the CPU board is not modified and remains in factory condition.

CONVERSIONS — The Up-Grade of your existing C2-D Computer to Multiple User Configuration is also available. Call for details.

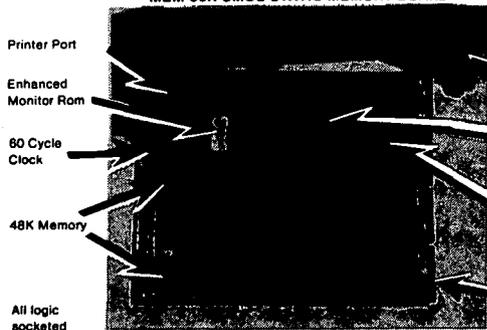
FLOPPY DISK UPGRADES FOR C1P, C4P & C8P

Our Memory/Floppy Board provides easy conversion of 502 and 600 CPU Computers to Floppy Disk Operation. The **MEMF** Board has a floppy disk interface which includes a data separator and the ability to automatically lift the disk drive heads — your floppy disk lifetime will be extended many times. You will retain the cassette interface for your existing software; which can easily be converted to Disk.

This **MEMF-16K** Board is populated with 16K RAM (50K possible) and has features of the MEM CMOS Static Memory Board with an added floppy interface. The low power memory means extra power supply not required. **ROM Basic** is retained even when Board is populated for 48K Disc Basic. An optional Parallel Printer Port and Real Time Calendar/Clock is on board.

Complete Ready to Run conversion kits with 5 1/4" or 8" Disk Drives are available.

MEM-56K CMOS STATIC MEMORY BOARD



ULTRA-LOW POWER — By using CMOS Static RAM Memory, the total power consumption is about 1/2 Amp at 5 Volts when populated for 48K. In fact, most of power is used by the Address Line Buffers and the Data Transceivers.

MULTI-USER — Can be addressed for any of the 16 multi-user memory partitions. The low power and single memory board/partition simplify installation and provide a typical \$1400 saving for a 3-user system.

MICRO-INTERFACE
 3111 SO. VALLEY VIEW BLVD., SUITE I-101
 LAS VEGAS, NEVADA 89102
 Telephone: (702) 871-3283

Check with your local Dealer or Order Direct.
 Phone orders accepted.
TERMS: Check/Money Order/Master Charge/VISA
 Sent POSTPAID ON PREPAID ORDERS.
 Foreign Orders: Prepaid only.
 Add 5% for handling/shipping.

another one here if desired, but in the interest of simplicity I didn't include the test here. The BMI instruction would be the logical choice to use if you're interested enough to pursue it.

Since we typed in a valid character we want to store it in our temporary accumulator at \$FC,FD. This is done by ROLA. We know that the routine ROLA uses the X register and that X is our counter so we preserve the contents of X by storing them in XTMP temporarily and recalling them when we get back from ROLA with lines 250 and 280. We loaded a character so now we must count it. This is done by decrementing X. When X reaches 0, we know we have loaded all 4 characters. If X is not 0 we must branch back and get the next one. This is done by the BNE INNN instruction. After all 4 characters have been loaded we take the contents of our temporary accumulator and put the low byte (\$A3) in the X register and the high byte (\$10) in the accumulator and jump to the convert routine 'CONV', as mentioned above. The number is converted and displayed but the cursor is left at the space after the last number of the answer. To clean things up a bit, a carriage return is sent to the display routine followed by a line feed which positions the cursor at the beginning of the next line. The RTS at the end sends us back to the routine which called for an input with a line feed in the accumulator.

The capability of interrupting the input vector routine is a quite powerful feature of the OSI computers and I have built a very usable operating system around it. Control A clears the screen, control B calls the checksum loader, etc., all the way to control Z. With everything stored in EPROM it is always instantly available. The output vector at \$021A can also be modified as can the control C, LOAD and SAVE vectors.

GAMES

Starzap REV 1.0 - Pretzelland - BASIC w/sound - \$9.95. The second thing I did when I got this game was to modify the keyboard routine to the Pretzelland standard. The first was to try to play it like it was. Why Pretzelland chose to vary from their own standard on this one I don't know. I definitely have trouble switching from game to

game and reprogramming my reflexes.

Once past this point, the game can be enjoyable if you keep a good attitude and don't let the limitations of BASIC get in your way. Controls don't always respond quickly and sometimes you get it from all sides at once so there is nothing you can do, but if you get a rhythm going and don't try to anticipate the action, you can fare pretty well. Successive games can result in scores like 600, 2200, 150, which demonstrates that more than skill is involved. Could use a little bit of revision. Not a rip off, but not for the serious player either. A 5 for REV 1.0.

Some thoughts on games in general. I don't like games with multiple levels of play. They don't add much to the game and they make comparisons between players difficult. If the game really does take some practice, maybe reducing the score value along with the skill required to keep things even would be acceptable. Then once selected, the player should be relieved of the burden of answering the skill level question each time he answers yes to the 'play again' prompt. On games where you can lose very quickly (Starzap for example) this becomes very tiring.

On games with sound, the normal keyboard routine should be bypassed between games and a scan such as used during play should be used so as to mute the annoying tone while waiting for the answer to the 'play again' prompts.

I forwarded these comments to Bob Retelle of Pretzelland but other authors and distributors should also take note. By the way, Bob informs me that although he normally charges \$1.00 for a catalog (refundable with first order) he will waive this charge for readers of 'Cassette Corner' if you mention it.



SINGLE DISK COPY UTILITY

By Jeffrey T. Dripps
Box 118 DeHaven Road
Beaver Falls, PA 15010

Owning a C1P-MF, or any other OSI computer with only one disk drive presents a problem when it comes to backing up a disk. OSI, evidently in an attempt to sell more drives, includes only a drive to drive

copy utility under 65D V3.2 and less. To the low budget computer owner with one disk drive, the only feasible solution is to write a disk copy utility in BASIC. This may prove adequate for a while but, a good copy routine that will copy multiple sector tracks requires a fairly large program. This in turn limits the amount of free memory to which disk tracks can be read to. Making a backup using a Basic utility can be a tedious process of inserting one disk after another in the drive until the specified track range is copied.

This frustrating task spurred me to do what OSI should have done, that is, write a machine language single disk copy utility for OS-65D V3.2 and less. The utility shown here will copy single tracks or any track range specified from track 0 to as many tracks as your memory will hold. Also, included in the utility are some rather unique features. Interested? Read on...

1. This routine displays on screen the track number, page count and the location in memory that each track is read to/written from.

2. The utility is exitable, that is possible to exit the utility after a disk read, re-enter the extended monitor, make changes to the data, then re-enter the copy utility and write the edited version to the backup disk.

3. It will "look" at the track prior to reading it to memory. If there is no data on the track it will skip over the blank track(s) so as not to waste memory by copying a blank track.

4. This utility will copy multiple sector tracks and will initialize the track prior to writing the track to disk.

5. The most important feature of this utility is that most of the subroutines called by the utility are already part of the 65-D operating system. This avoids duplication of already existant code and keeps memory requirements to a minimum. The small size of the utility permits its location in a convenient place above O.S. and outside of user memory. Because of its efficient use of memory, this utility will copy a disk in half the number of "passes" that a Basic copy utility would require.

Getting it on disk...

All of the routines named in the source listing are either part of the 65-D.O.S. or are part of the source except 'clear'. This is a M.L. screen clear I have burned in ROM at \$FCD6. Anyone having a screen clear at another address may change line 620 appropriately or if you have no screen clear at all, omit this line and also lines 750, 810, 1120, 1240, and 3490.

The source listing shown will require 32K of memory for assembly as it requires a memory offset of \$7000. That is...

.M 7000
.A 3

This will assemble the object code to memory at \$7200 for 6 pages. Then to save the object code (I used track #13). .ISA 13,1=7200/6 Note: For users with less memory the utility will probably assemble in a "minimum system" by leaving out most of the comment lines.

Putting it to use...

The copy utility is very straightforward and really requires no explanation. To use it 'exit' to the O.S. Kernel then type.

A*CA 0200=13,1
A*GO 0200

Note: There is no memory checking done by the copy utility. That is, if you specify a track range to be copied and the number of pages of data on these tracks exceeds the available RAM in your system the utility will not notice this and it will read as many tracks as you specify whether there is memory at that address or not. This is not really a problem though as the screen will display the address of each track read to memory. If you make a mistake and copy more tracks than your available RAM will hold all you will have to do is re-copy any tracks that were read from above available RAM. Also, if someone would want to add memory checking to the utility it would be quite simple, I did not as I was trying to keep the utility under seven pages in length.

How's and Why's...

The copy utility loads to \$0200 for 6 pages. This is above the area used by the extended monitor. This allows you to use the copy utility and the extended monitor interactively, provided the EX.

The procedure is as follows:

A* EM
:ICA 0200=13,1 (call the copy utility)
:@ 05A4 /09 3A (change to read first
specified track to mem.
at \$3A00 and up)
:EXIT
A* GO 0200 (jump to copy utility)

Then proceed as in the example below:

OS-65D V3.X
Single Disk Copy Utility

Specify Starting Track:0 (for example 0)

Specify Ending Track:3 (for example 3)

INSERT MASTER DISK!

Are You Ready(Y/N):Y (enter Y)

Reading Track:00 To:\$3A00 For:8 pages
With:\$2200 As The Load Vector

R/W Track:01 To:\$4200 For:8 pages

R/W Track:02 To:\$4A00 For:8 pages

R/W Track:03 To:\$5200 For:8 pages

INSERT DESTINATION DISK!

Are You Ready(Y/N):N (enter N)

A*RE EM
:@XXXX

(This will allow
exit to the
Kernel)

(change any
locations
desired)

(copy re-entry
point)

:EXIT

A*GO 023D

INSERT DESTINATION DISK!

Are You Ready(Y/N):Y (enter Y)

Writing Track:00 From:\$3A00 For:8 pages
With:\$2200 As The Load Vector

R/W Track:01 From:\$4200 For:8 pages

R/W Track:02 From:\$4A00 For:8 pages

R/W Track:03 From:\$5200 For:8 pages

Again(Y/N):N

DISK COPY COMPLETE
RE-BOOT SYSTEM

A*

Editor's NOTE: A neat piece of diligent work. Even though OS65D V3.2 now has a single copy routine, it does not give the user the opportunity to alter contents before writing the copy disk.

MON. is enabled prior to calling the copy utility. The copy utility in its normal configuration is "set" to read the first track specified to \$0900. If it is an 8 page track (such as track 0) the track will occupy memory from \$0900-\$1100. This will over-write the assembler but, the EX. MON. at \$1700 will still work satisfactorily. If you specify two tracks (such as

track 0-track 1) then track 1 will over-write the EX. MON. code making it useless. In short, if you must call and edit multiple tracks, you may change the starting page (Location \$05A4) in the copy utility to any page after the O.S. (ie. page 33 and up).

See listing next month.



LETTERS

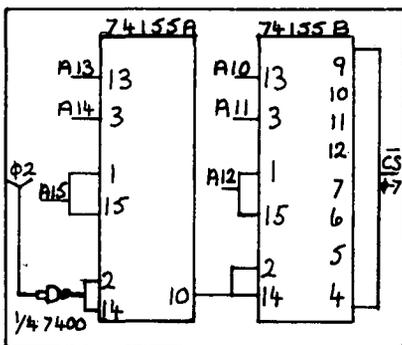
ED:

In answer to Frank Aguilar's letter (p.21, March 1982), I have a memory map for the ClP-MF in the back of my factory supplied user's manual. It shows quite a bit of uncommitted memory space where he could insert a PROM.

Such as, 8000H to 9FFFH (8K) or E000H to EFFFH (4K). Then again, he could stick some ROM into space allowed for RAM.

If he wants to preserve certain ClP routines, he could copy them into his new PROM or "burn" a new monitor PROM which contains the pointers to the old code he wants to use. His term "non-ClP code" is ambiguous. If the code isn't used by the ClP, then what's it there for?

At this time, let me make a further correction to my article on Cl/Superboard memory expansion (August 1980). The two chip select decoders should be wired as follows:



The memory expansion is now functioning as expected- quite well.

Bruce Showalter
Abilene, TX 79601

ED:

Throughout the many issues I have of PEEK (65), (and Aardvark J.) I have seen the plea for information on how to jumper the 610 board clock divider into the PIA for DOS operation. This was little concern to me since I was using my SB II with 610 at 32K with cassette, plus GE Terminet Printer. However, I am now going to add 5-1/4 BASF Floppies. Expected a few interface problems but just realized that I had no data on the clock patch. Went back

and reviewed all issues of PEEK and the Journal. I just couldn't find the answer! I suppose the routine answer is to contact your local OSI dealer. In my case there is no OSI dealer within 175 miles of my town. (And those who I have contacted during my travelling are sales oriented with no time or knowledge for technical assistance).

Therefore, this letter is to ask if you have in your files any data on the 610 clock to PIA patch. If so, I enclose a SSAE for any information you can provide. If you have no data, perhaps you could find a place in your LETTERS column to add this plea.

Neither have I seen 610 board J-3 pinout nomenclature. Your December '81 issue, page 3, shows pinout nomenclature for the 470 board, but it does not exactly agree with my 610 and other sparse data I have. Do you have the nomenclature for the 610?

After reading your February issue, I noted on page 19 that the Southeastern Minnesota Group is working with BASF floppies. Is it possible to contact them for some help?

William J. Beebe
Pittsfield, MA 01201

William:

Please note that the following J-3 pinout is almost identical to the 470 board pin. Also a cable that will connect an industry standard 5-1/4" floppy drive that is ANSI standard will interface directly with no ? problems. Currently OSI is using MPI (9754 Derring Ave., Chatsworth, CA 91311) model 51 drives. The cable is an OSI catalog A13 and retails for \$21.00. The cable plugs onto J-3 and the floppy drive. The only modification may be to the select and ready lines as OSI doesn't use the industry standard pinout.

610 PINOUT

1. HEAD LOAD
2. LOW CURRENT
3. SELECT DRIVE 1
4. FAULT RESET
5. STEP
6. STEP IN
7. ERASE ENABLE
8. WRITE ENABLE

Flat Rate

DISK DRIVE OVERHAUL

One Week Turnaround Typical

Complete Service on Current Remex, MPI Siemens and Shugart Floppy Disk Drives.

FLAT RATES

8" Double Sided Remex	\$170.00*
8" Single Sided Siemens	\$150.00*
5 1/4" M.P.I. Drive	\$100.00*

Other Rates on Request.

*Broken, Bent, or Damaged Parts Extra.

YOU'LL BE NOTIFIED OF

1. The date we received your drive.
2. Any delays and approximate time of completion.
3. Date drive was shipped from our plant.
4. Repairs performed on your drive.
5. Parts used (* and description).
6. Any helpful hints for more reliable performance.

90 day warranty.
Ship your drive today.
Write or call for further details.

We Sell Parts
PHONE (417) 485-2501
FESSENDEN COMPUTER SERVICE
116 N. 3RD STREET OZARK, MO 65721

AT LAST...

...FOR OSI

For investors and financial managers

Stock portfolio analysis
\$150.00

- in your office - instant valuations
- compound growth measurement
- pertinent company operating statistics
- input data electronically

Stock financial statement analysis
\$250.00

- input your interpretation of financial data
- analyze up to 10 years of data
- see mean, trend and stability
- data stored on DBM system
- input data electronically

On-line data retrieval
\$50.00

Accounting package
\$150.00

DBM system
\$200.00

for 8" floppy/hard disc under OS65U

write for details

Genesis Information Systems, Inc.
P.O. Box 3001 • Duluth, MN • 55803
Phone 218-724-3944

9. WRITE DATA
10. SEPERATED CLOCK
11. SEPERATED DATA
12. GROUND
13. GROUND
14. NC
15. NC
16. NC
17. INDEX
18. SELECT DRIVE 2
19. WRITE PROTECT
20. READY DRIVE 2
21. SECTOR
22. FAULT
23. TRUCK 00
24. READY DRIVE 1

The patch for the clock is as follows: connect U11 pin 1 to U10 pin 9; U11 pin 9 to pad no.3 of the 4 square pads below U10 and U11 (left to right component side up) U72 pin 18.

The address for the Southeastern Minnesota Group is: OSMOSUS NEWS, 128 Silver Lake Rd., Minneapolis, MN 55418.

If you have any more questions, please write to us.

Brian
Asst. Tech. Ed.

* * * * *

ED:

Thanks to you and Ron Mosley for Ron's article ("OS-65U V1.2, Level 3") in the February 1982 issue of PEEK (65). We applications programmers working under Level 3 received some valuable information.

I would like to add some comments based on my own experiences. The numbered section headings correspond to those in the original article:

2. User-Defined Inputs

The SECURE INPUT routine is very clever, not only because it prevents sensitive data (such as a password) from being echoed to the screen, but because it prints random characters in place of the ones being typed. It does call for quite a few POKES to

system memory, and I would like to offer a simpler, though admittedly less clever, alternative. The following example prompts the user for a password but echoes nothing to the screen:

```
10 PRINT "ENTER PASSWORD: ";
20 POKE 11686,0: REM** Disable
  echo to any output device
  **
30 INPUT PWS
40 POKE 11686,1: REM** Restore
  echo to screen **
```

This method does allow spaces in the password. You "oldtimers" with long memories might recognize this routine. I published it in the October 1979 OSIO Newsletter.

4. Level 3 Printer Paging

Ron mentions in his article how his POKES to printer control locations 14457 and 15908 messed up another user who was using the printer when the POKES were executed. As the article explains, this occurs because there is one system-wide line counter under Level 3. Our experience was similar, but we discovered the cause in a slightly different manner:

We have a four-user system and use Digital Technology's Hard Disk Executive, which contains some unique programs and has modified versions of some of the standard OSI utility programs. As with all Level 3 systems, we turn on the computer, boot it, bring up Level 3, and key in the system date and time. After all of this done at the system terminal (User #0), the Hard Disk Exec's modified version of BEEXEC* prints the following prompt on all terminals: "Start-up option (or C/R for normal start-up):". A carriage return here causes the system menu to be printed on the terminal screen.

Now, suppose user X walks in at 7:30 AM, boots the system and starts printing a report, all from the system terminal. If the other terminals remain unused, they are sitting there with the prompt described above on their screens. We print some very long reports, so the printer is still going strong at 8:00 AM when user Y walks in. User Y sits down at one of the unused terminals and sees the start-up prompt on the screen. Quite naturally, he/she hits the RETURN key to get the system menu. What happens? The line counter is reset! That is, the system line counter,

stored in memory location 15908, is changed back to 601. This of course destroys the paging that was being performed for the report on the printer.

I haven't explored the problem deeply enough to determine where the line counter is being reset. Our temporary solution is to get to the system menu on all terminals before starting any reports. Since rebooting a specific user doesn't cause this problem, we only have to worry about this when the entire system is booted.

I'll pass along any further discoveries on this subject. It might only be a problem for those of us using D.T.'s Hard Disk Executive. Come to think of it, I haven't even tried Ron's modification to program "LEVEL3" (renamed "L3 74" for C3-B owners using the Hard Disk Executive). Maybe that will do the trick.

Brian Goodhart
Rockville, MD 20852

* * * * *

ED:

To all of you who have the Sanders' Terminal Program, AND who also have both serial and parallel printers in use... Take a look at Sanders' BEEXEC* particularly lines 400-740. I put part of this in my BEEXEC* for OSDMS (the part pertaining to port 5 and 8). Now I can select either Serial (port8) or Parallel (port5) when doing a printout. I also amended STAT03 (this previously appeared in PEEK (65) so that I can either run it as originally programmed (in my case for Serial) or, as in PEEK (65), for my parallel Centronics 737, allowing for automatic font selection, i.e. when 132 col. selected, the C737 selects the condensed font and when 80 col. selected, the C737 selects 10cpi. The Serial printer I am now using is a 25cps Itoh Starwriter. It took some maneuvering to get the dip switches right, but it appears that at the low speed of 300 baud (that's what I have port 1 on my 550 set for) handshaking lines (pin 20) are not needed. Likewise, the modem on port 2 also has no handshaking and (as is evident from this message) presents no problem either. When you set your 550 for a faster speed, then apparently handshaking is needed. I tried it at 2400 baud with the Itoh and it just wouldn't work although the

Itoh can be switched for 2400 (plus other lower speeds). But to get back to Sanders' efforts - those port selections POKES are great! The POKES as well as the "change" (in hex) in OSI Tech Notes number 24, page 11, are NO GOOD. One more note: if you select Serial, and you want to LIST or PRINT to a DV#, you still have to use DV#5. At least in my case this is so.

Fred Schaeffer
Jamaica, NY 11435

* * * * *
ED:

Can you recommend any reference material for OS-65U?

My C8P-DF came with version 1.2, but the dealer provided only a copy of the Operators Manual for version 1.1, dated June 1978. He said that was the only documentation available. [He was right - Al.]

My specific problem at the moment involves how to use two disk drives at once. From your phone bill, it looks like we all need more reference manuals. The articles in PEEK (65) are helpful, but perhaps a bibliography on where to find what, would let us new users catch up to the old-timers more quickly. That might be one way to reduce the phone bill!

Jim A. Wakeford
Birmingham, AL 35243

Jim:

65U is in a state of flux, with new versions coming out frequently. By summer, it should be settled down with a

decent new manual. Have patience - the new versions are worth waiting for!

To use both disk drives in BASIC, do something like this:

```
100 DEV"A"
110 OPEN "FILE1","PASS",1
120 DEV"B"
130 OPEN "FILE2","PASS",2
```

Thereafter, PRINT#1 or INPUT#1 will automatically go to FILE1 on the A drive, PRINT#2 or INPUT#2 will automatically go to FILE2 on drive B. To create a file on B, LOAD"CREATE","PASS, then type DEV"B and RUN.

Meanwhile, you probably would find the following reading material helpful.

1. Basic Reference Manual ROM, 65D & 65U
2. OS-65U Primer
3. C8P Users Manual

Al

* * * * *
ED:

I have the lines 5000-5320 keyed in STAT03. Now, if you'll be specific about:

"Just about the only other part needed is an extra line which asks, after you have input all the information for a new report, whether you want to save that report on the list of reports. If you do, another couple of simple lines look at position 0 in the data file, where the true end of file is kept, write your information on the new report just past that point, then update the end of data file information at position 0."

I'll use the heck out of this. Otherwise, I'll keep answering the questions and getting the same reports over and over.

Jeanne English
Jackson, MS 39211

Jeanne:

Never again will I fail to list ALL lines of a suggested program change! You are the third person to have this problem. My sincere apologies, and the correct listing:

```
5000 REM RUN PREVIOUS REPORT?
5005 C=0
5007 DEV"A"
5010 OPEN"STAT3B",1
5020 EF=10000
5030 FIND"^^",1
5040 IF INDEX(1)>EF THEN
5045   5200
5050 C=C+1:KP(C)=INDEX(1):REM
5055   POSITION OF A HEADER
5060 INPUT#1,H$:H$=MID$(H$,3)
5070 PRINT C;" ";H$
5080 GOTO 5030
5200 INPUT"RUN ONE OF THESE
5205   AGAIN";AG$
5210 IF LEFT$(AG$,1)="Y" THEN
5215   5300
5230 RETURN
5300 INPUT"WHICH ONE";NN
5310 INDEX<1>=10:PRINT#1,KP
5315   (NN)
5320 CLOSE:RUN"STAT3A"

4000 REM DUMP REPORT PARAMS TO
4005   SUBMIT FILE
4010 REM
4020 REM
4030 INPUT"SAVE THIS REPORT ON
4035   THE LIST OF REPORTS";AL$

4560 PRINT #1,R1$(X): PRINT
4565   #1,R2$(X): GOTO 4590:
4570   REM T1=1, T2=7, T3=1
4575   PRINT #1,R1(X): PRINT
4580   #1,R2(X): GOTO 4590:
4585   REM T1=1, T2=7, T3=2
```

NEW
FULLY DOCUMENTED
COMPREHENSIVE

A Payroll Package designed with the following features:

- DMS compatible employee files
- Hard or soft disks
- Multi-company usage
- Support of serial or parallel printer
- Configurable for various CRTs

Program includes:

- Up to 52 taxing authorities
- Check printing routine
- Comprehensive list of deductions
- Generation of forms 941 & W2
- Provision for multiple department code charges
- Crediting for vacation and sick hours earned

A PAYROLL PACKAGE

DMS COMPATIBLE

Morganstein Consultants
13329 Woodruff Court
Germantown, Maryland
20874

A detailed Manual of Instructions is available for review. The Manual's price of \$10.00 is deductible from the cost of the Payroll Package. Price includes 1 year's free update on program improvements! Alternative individualized check printing routines optional. Total Payroll Package \$390.00 check or money order.

Dealer Inquiries Invited Allow 2 weeks for delivery

digital technology

BUS-II LEVEL I BOOKKEEPING & ACCOUNTING SYSTEM

The BUS-II turn-key multi-client accounting package is the leading OSI business software package. BUS-II Version 32 includes five principle modules:

	Inst. Price	List Price
GENERAL LEDGER	\$1200	\$599
ACCOUNTS RECEIVABLE (a)	1000	599
ACCOUNTS PAYABLE (a)	1000	599
ORDER ENTRY W/ INVENTORY (a) (b)	1000	599
PAYROLL (no extra charge for optional versions)	1200	799
01 - STANDARD MULTI-STATE OPERATION		
02 - CPA FIRMS & SERVICE BUREAUS		
03 - RESTAURANTS		
04 - COMMISSION SALES		
05 - CONTRACTOR'S JOB-COST ACCOUNTING		

The Accounts Receivable, Accounts Payable, and Order Entry W/ Inventory are completely interactive with the BUS-II General Ledger. Two optional specialized packages (completely interactive) are also available.

CPA EXTENSIONS (see below)
POINT-OF-SALE TERMINAL W/ INVENTORY (see below)

The BUS-II CPA EXTENSIONS Package provides special features for accountants and bookkeepers. The POS-1 Point-of-Sale Terminal package enables the operator to use the computer system's video terminal as an on-line "electronic cash register."

Note: BUS-II operates on floppy-disk or hard disk-based systems running the OS-65U operating system (single-or multi-user). Multi-client use can accommodate any number of client companies on floppy disk systems or hard disk systems with H/D/E (required for hard disk use). BUS-II LEVEL I files are limited in size for floppy disk backup; floppy disk operation continues in case of hard disk failure.

BUS-II "SOFTWARE EXCHANGE" SPECIAL

Users of other business software packages who wish to upgrade to Digital Technology's full-function BUS-II BOOKKEEPING & ACCOUNTING SYSTEM can, in many cases, "trade in" their old, unusable, or unsupported software for full rebate of the original purchase price (up to \$750.00). The "SOFTWARE EXCHANGE" offer includes virtually all of the previously-available OSI business software packages. Contact Digital Technology or your dealer for more information.

BUS-II LEVEL II (EXPANSION TO BUS-II LEVEL I)

BUS-II LEVEL II is designed for much larger businesses. Expanded file size and special operations allow virtually unlimited numbers of accounts and transactions. BUS-II LEVEL II requires BUS-II LEVEL I. Minimal backup is data cassette (tape) or floppies--although multiple Winchester disk operation is recommended (provides ability to continue computerized bookkeeping functions in case of hard disk failure.) H/D/E Hard Disk Executive is required.

	Inst. Price	List Price
GENERAL LEDGER (c) (d)	\$	\$399
ACCOUNTS RECEIVABLE (c) (d)	600	399
ACCOUNTS PAYABLE (c) (d)	600	399
ORDER ENTRY W/ INVENTORY (c) (d)	600	399

CPA EXTENSIONS PACKAGE

CPA EXTENSIONS is designed for public accounting firms. A number of special operations are provided: "bankers" Balance Sheet and Profit and Loss statement with summarization and consolidation options, Statement of Changes in Financial Position, Statement of Changes in Components of Working Capital, Cash Flow Analysis, Departmentalized Sales Analysis, Asset Depreciation Schedule (compatible with TAXMAN-1040), and Loan Amortization Schedule. In addition, a pre-processed or "after-the fact" payroll system is provided.

CPA EXTENSIONS is interactive with BUS-II 32 BOOKKEEPING & ACCOUNTING SYSTEM

CPA EXTENSIONS (a) Inst. Price \$2400 List Price \$1500

POINT-OF-SALE TERMINAL

POS-1 is an on-line multi-store point-of-sale terminal program with integrated inventory designed for cash register emulation. POS-1 controls cash drawer and ticket printer (or system printer). Automates taxable or nontaxable sales, cash transactions, and credit sales (with verification operations). POS-1 also allows the use of industry-standard bar code readers with the point-of-sale terminal system through a "Siamese port-- on the C2 or C3 CPU card. (Extra serial port NOT needed except in multi-user operation.) Configured for industry-standard RS232 bar code "wand" (INTERMEC) or "window" (SPECTRA-PHYSICS).

POS-1 is interactive with the BUS-II V 3.1 BOOKKEEPING & ACCOUNTING SYSTEM.

POS-1 POINT-OF-SALE TERMINAL (a)(b)
Inst. Price \$1600 List Price \$1199

TAXMAN-1040 PERSONAL INCOME TAX PREPARATION

TAXMAN-1040 is designed for tax practitioners and public accountants. TAXMAN-1040 is the leading tax package for OSI microcomputers--the package has been installed on OSI, Hewlett-Packard, DEC and IBM systems. Designed and supported by CPA tax experts. This package automatically prepares FORM 1040 and 32 schedules. Support includes annual forms, tax tables, and computational revisions in accordance with Federal Tax Law changes.

TAXMAN-1040 Inst. Price \$3600 List Price \$2399

H/D/E HARD DISK EXECUTIVE

Digital Technology's implementation of H/D/E is the answer to AMCAP's HDM. Digital Technology's H/D/E provides user functions not found on HDM of similar products: ability to copy from any user "system" to another; automatic recovery in case of "back-up to floppy" or "restore from floppy" utility failures, allowing the user 3 options: (1) ignore error, (2) abort to menu, (3) try again; use of both "A" and "B" floppy drives to back-up hard disk files; and automatic back-up diskette initialization. H/D/E operates on any OSI Winchester disk system from 7 - 80 megabytes. Re-use of hard disk space is provided. Superior to AMCAP's hard disk manager in every respect (and Digital Technology software does not self-destruct).

NOTE: H/D/E is required when installing any Digital Technology business applications packages on OSI hard disk systems.
H/D/E HARD DISK EXECUTIVE List Price \$399

OS-DMX DATABASE MANAGEMENT SYSTEM

Command-oriented OS-DMS compatible database management system. OS-DMX operates under the OS-65U V1.2 operating system (single- or multi-user). Features such as control files, extensive operating commands and the innovative HELP function, make this one of the most usable--as well as powerful--systems available for microcomputers. OS-DMX may be used instead of, or in addition to, OS-DMS Nucleus, Query, Sort; OS-DMX will replace virtually all of the specialized OS-DMS modules-- and in most applications will provide greatly improved performance.

OS-DMX Database Management System buyers will receive (no extra charge) a number of "extras" previously sold separately:

DMX-MAIL	Mailing List Management (FEB 82)
DMX-STAT	Comprehensive Statistical Analysis package (JULY 82)
DMX-COPY	Edit Database Structure after the fact (FEB 82)
DMX-MERGE	File Merge Operation (FEB 82)
DMX-TUTOR	450-Pg Tutorial w/ Demo Data Diskette (AVAILABLE)

In addition, DMX-SORT operations will be upgraded to machine-code sorting for faster operation. There will be no need to purchase high-speed sort programs separately.

OS-DMX DATABASE MANAGEMENT SYSTEM
Inst. Price \$1600 List Price \$1199

BISYNC-80/HASP

BISYNC-80/HASP is a full-function Multileaving Workstation package which allows communication with any remote CPU that supports a HASP Multileaving Workstation, and, as such, is ideally suited to Remote Job Entry applications.

OS-BISYNC-80/HASP (e)(f) List Price \$1195

BISYNC-80/3270

BISYNC-80/3270 is a full-function IBM 3270 terminal emulator which allows the microcomputer to communicate over point-to-point telephone lines with any IBM S/360, S/370, or S/30xx CPU that provides standard IBM support for one of the following:

IBM 3275	Model 2
IBM 3271	Model 2 or control unit w/ attached 3277 Model 2
IBM 3284	or 3286 printer

OS-BISYNC-80/3270 (e)(f) List Price \$895

BISYNC-80/3780

BISYN-80/3780 is a full-function IBM 2780/3780 emulator allowing the microcomputer to communicate over point-to-point telephone lines with any CPU or device that provides standard IBM support for:

IBM 2780	Models 1, 2, 3 or 4
IBM 3780	w/ or w/o 3781 card punch
IBM CPU	to CPU BSC communications

OS-BISYNC-80/3780 (e)(f) List Price \$895

BISYNC-80/ASYNC

BISYNC-80/ASYNC is a full-function asynchronous communications package which allows microcomputers to communicate asynchronously with a mainframe or other microcomputers. This package is an ASYNC adaptation of BISYNC-80/3780 terminal emulation program, providing asynchronous communications at 75 to 9600 baud, using full IBM BISYNC protocol.

OS-BISYNC-80/ASYNC (e)(f) List Price \$195

OS-BISYNC-80 SYNCHRONOUS INTERFACE ASSY

List Price \$395

NOTE: The prices shown in this catalog are estimates only; contact your OSI dealer for quotations. The "suggested installed price" reflects a typical business installation and includes reasonable allowance for software installation, minor program adaptation or customization, operator training, dealer support, back-up, etc. The "reference" or "list" price reflects a base price for the software for comparison purposes, exclusive of dealer installation and support.

Digital Technology, Inc., is the largest independent supplier of OSI software with hundreds of business packages in use around the world. Digital Technology software is sold by a growing number of conscientious OSI dealers and OEMs. Every package is backed by the finest support program in the microcomputer industry. All "bugs" are fixed free of charge. Updates (fixes to bugs, minor enhancements, new product announcements) are provided to all dealers and licensed users free of charge. And upgrades to new versions are encouraged (at nominal charge).

Digital Technology's software is user-obtained. In fact, no one else provides such extensive features as on-line documentation, idiot-proof prompting, and operator's manuals that are comprehensive, detailed, and accurate. All Digital Technology software systems allow the operator to "set" the programs to the type of video terminal and printer used. The operator selects the terminal and printer types from the list provided in the "TERMINAL & PRINTER OPTIONS" program. Screen formatting and printer control are provided automatically yet may be redefined through user subroutines.

digital technology

inc.

P.O. BOX 178590
SAN DIEGO, CA 92117
(714) 270-2000

REQUIREMENTS

- BUS-II LEVEL I or LEVEL II G/L req'd
- BUS-II LEVEL I or LEVEL II A/R req'd
- Corresponding BUS-II Level I module(s) req'd
- H/D/E req'd
- C3 CPU W/ 56K RAM & OS-CP/M or Lifeboat Associates CP/M req'd
- SYNCHRONOUS INTERFACE ASSY req'd

```

4580 PRINT %1,QQ(X+4): REM
      T1=2, T2=1-6, T3=1 OR
      T3=2
4590 NEXT X
4600 IFLEFT$(AL$,1)="Y" THEN A
      =INDEX(1):INDEX<1>=0:
      PRINT%1,A
4605 CLOSE: DEV CHR$(DV(2)+65)
      : RUN "STAT3A","PASS",2

```

Al

* * * * *

ED:

Recently, after many years of "Big Iron" data processing, I was blessed with an OSI C3 OEM. With the aid of a business acquaintance, I have been able to put together a system of programs on OS-65U in BASIC that provides support for my office staff. Now, however, I would like to get CP/M running and I am completely lost.

For some reason the dealer provided me with a serial interface to my NEC printer via a non-OSI board, which I believe is a UART (it runs on Device Number 3 under OS-65U). The version of CP/M I received with the hardware is 1.4. I have been told that I can get updated to 2.2 but when I checked with Lifeboat they told me that because of an agreement with OSI, they can-

not update.

My questions are these: 1) How can I get an updated copy of CP/M most economically, 2) How do I get printer support on a UART, and 3) Can you put me in touch with any documentation or other OSI C3 users who have a similar system so I can compare notes?

Frederick A. Starck
Bellevue, WA 98004

Fred:

Contact your dealer about updating your CP/M. The documentation which comes with CP/M 2.2 includes an assembly source code listing of the device drivers for I/O. With this listing and DDT you should be able to find the UART driver (called a CA-6 board in the listing) and change your I/O BYTE (location \$03) to direct output thru the UART, which is probably at \$FB00.

Who else has done this, readers?

Al

* * * * *

ED:

I own an Ohio Scientific Challenger C2-4P microcomputer with a 48 line bus system. I also own a Micropolis 1053-II floppy disk drive that was designed for a Z80 or an S-100 bus. Could you give me an idea of the problems I stand to face if I try to interface the two, or could you direct me to some place with either information for hardware/software necessary to accomplish this project?

D. Kirkpatrick
Capitola, CA 95010

P.S. Can you tell me where to get information on writing a program that will check to see how much addressable memory space is available in my computer?

Kirkpatrick:

The Micropolis 1053-II drive will interface to your C2-4P using a 470 board and an A12 cable assembly. See your local dealer for these and for OS65D version 3.3.

Your C2-4P comes standard with 8K RAM and can be expanded to 32K. When you boot your computer it comes up with a message part of which indicates

BACH-MANN

ONLY \$27.50 and FULLY GUARANTEED! My kids say it is BETTER THAN ATARI PAC MAN!®

FAST ASSEMBLY PROGRAM 'RUN' LIKE BASIC. BEWARE THE INTELLIGENT ROBOTS WHILE YOU TRY TO EAT THE MAZE.

REQUIRES OS-65U AND SERIAL TERMINAL WITH CURSOR ADDRESSING. SUPPLIED ON 8-INCH DISK. EASILY CONFIGURED TO YOUR TERMINAL OR STATE TYPE WHEN ORDERING. COMPLETE INSTRUCTIONS INCLUDED... YOU CAN'T GO WRONG.

ALL OF SANDERS' SOFTWARE INCLUDES
30 DAY SATISFACTION GUARANTEED
OR MONEY BACK WARRANTY

J AND T ASSOCIATES
2338 RIVIERA DRIVE
VIENNA, VA. 22180

®trademark Midway Mfg. Co.

SMART TERMINAL SOFTWARE

\$35.00

A COMPLETE MODEM COMMUNICATIONS PACKAGE FOR OS65U SERIAL SYSTEMS. VERY EASY TO USE. INCLUDES A FULL RANGE OF KEYSTROKE CONTROLS FOR DUPLEX, DELAY, FILE HANDLING, AND SIMULTANEOUS PRINTER OPERATION. OVER 200 USERS ARE SENDING AND RECEIVING PROGRAM AND DATA FILES, OR JUST COMMUNICATING. IN DAILY USE WITH CDC AND IBM MAINFRAME WORK. SUPPLIED ON 8 INCH DISK WITH COMPREHENSIVE MANUAL. ASSEMBLY PROGRAM WITH BASIC DRIVER.... EASILY CONFIGURED TO ANY ACIA OR UART ADDRESS, OR STATE DESIRED CONFIGURATION WHEN ORDERING.

BROWSE THRU OS-65U

BROWSE -- a professional software system for examining and modifying 65U disk data sets. This SCREEN-ORIENTED utility presents a "window" into any disk area, displays each byte in graphics and hex, which may be modified with BROWSE's screen editor. BROWSE easily, quickly, and reliably installs operating system patches, dumps files, displays file processing test results, and repairs data files. With BROWSE, every byte on a 65U disk is accessible. BROWSE's command set includes OFFSET, EQUATE, PRINT, CALCULATE, FORWARD SCROLL, BACKWARD SCROLL, JUMP, and ZAP. BROWSE supports two disk access methods for maximum versatility. This very powerful software was developed by CSS in consultation with J & T Associates. BROWSE is delivered complete with diskette, on-line users guide, and start-up instructions for the low introductory price of \$30.00- send orders to:

CSS, 2402 Freetown Drive, Reston, VA 22091

how much RAM is available for BASIC use. In an 8K machine this is 7423 bytes. The rest of the 8K is used by BASIC in ROM as a stack and scratch pad.

Brian
Asst. Tech. Ed

* * * * *
ED:

On a ClP-MF, is there any way to have a program stored on an OS65D disk but run under ROM BASIC as PICO-DOS does?

It would be nice to take advantage of the higher storage capabilities of the 65D and not have to rewrite programs written for ROM BASIC (keyboard scan, control C off, etc.).

I think it would be very beneficial to have an article tabulating the differences between ROM and disk BASIC locations, at least for the popular PEEKs and POKEs, and showing what is required to convert a program. It would help those who started with a tape system and converted to disk.

Thanks for any help you can give me.

Richard List
Pittsburgh, PA 15221

Readers:

Who can help?

Al

* * * * *
ED:

This letter is in reference to Mr. Atchley's article entitled "Invisible Password Program" which appeared in the March 1982 issue. The program performs the same function as POKeing the Output Distributor (at location 11686) to 0 before an INPUT and POKeing the original value back into the Output Distributor after data has been entered: the data entered through the terminal keyboard is not echoed back to the screen. (Under OS-65U only. - Al)

Mr. Atchley's program provides a means of limiting the number of characters which can be entered. If the Output Distributor is used as described above with OS-65U version 1.3, a limit can be placed on the number of characters to be entered by using the Extended Input feature.

Either method provides some security for passwords. Both methods also have what I consider to be a basic flaw: the cursor does not move across the screen as data is being entered. Some computer systems I am familiar with send a blank or forward space cursor command to the CRT each time a character is entered as evidence to the operator that a key on the keyboard was indeed depressed. If someone has a method for "blind" password entry, I would appreciate his/her sending it to PEEK (65) for publication. (See Jim Sanders' article, PEEK (65), May 1980, p.14. - Al)

Sample program for "blind" password entry using Output Distributor facilities:

```
10 A = PEEK(11686):REM SAVE
CURRENT OUTPUT DISTRIBUTOR
VALUE
20 PRINT "ENTER PASSWORD";:REM
REQUEST ENTRY OF PASSWORD
30 POKE 11686,0:REM DISABLE
ECHO OF INPUT TO TERMINAL
40 INPUT PW$:REM GET
OPERATOR'S RESPONSE
50 POKE 11686,A:REM RESTORE
OUTPUT DISTRIBUTOR VALUE
60 IF PW$ = "PASS" THEN ... :
REM TEST OPERATOR'S ENTRY
```

Sample program for "blind" password entry using Output Distributor facilities and Extended Input to limit number of characters entered:

```
10 A = PEEK(11686):REM SAVE
CURRENT OUTPUT DISTRIBUTOR
VALUE
20 PRINT "ENTER PASSWORD";:REM
REQUEST ENTRY OF PASSWORD
30 POKE 11686,0:REM DISABLE
ECHO OF INPUT TO TERMINAL
40 INPUT [4,"A"] PW$:REM GET
OPERATOR'S RESPONSE
50 POKE 11686,A:REM RESTORE
OUTPUT DISTRIBUTOR VALUE
60 IF PW$ = "PASS" THEN...
:REM TEST OPERATOR'S ENTRY
```

David A. Weigle
Morton, IL 61550

* * * * *
ED:

Can you give a list about language and hardware tools supported by CP/M in OSI C3-A?

Jose Garcia
Venezuela

Jose:

Lifeboat Associates (1651 Third Ave., NY, NY 10028) and Westico, Inc. (25 Van Zant St., Norwalk, CT 06855) both offer all their CP/M programs in OSI disk format including MBasic, Cobol, Fortran, Z-80

Macro Assembler, DBase II, Wordstar and lots more. Write to either one for a catalog.

Al

* * * * *
ED:

The college where I teach has a C2-4P and a C2-8P. I have obtained a 430B board so hope to be able to output an RS-232 signal. Does anyone know what kind of software, if any, is necessary to emulate a terminal? [Check out the J&T Assoc. ad in PEEK - Brian]

The C2-8P now will only go into machine language when I reply to the H/D/M? query.

Bruce Hughes
Pukalani, HI 96788

Bruce:

The problem is probably with the shift lock key or in that area. The routine in ROM that decodes the input boot character, defaults to the machine language monitor program if the input character is not a capital H or D. So look in this area. It is not likely, but you could possibly have a bad monitor ROM or address decoding problem. The addresses are:

FD00 - FDFF HARD DISK (H)
FF00 - FFFF FLOPPY DISK (D)
FE00 - FEFF 65A MONITOR

Brian
Asst. Tech Ed

* * * * *
ED:

In the March 1982 issue, Frank Aguilar asked about Monitor Rom locations vital to his ClP.

I believe the No. 3 issue of the OSI Small Systems Journal in September 1980 Micro has answers to his questions. If he can't locate a copy in Laredo, I will be happy to make him a copy of mine.

S.C. Dodd
Alamogordo, NM 88310

S.C.:

Also, of course, PEEK (65) offers the complete set of Small Systems Journals bound for \$15.00 (plus \$3.00 shipping).

Al

* * * * *

ED:

In regard to a letter sent in by Mr. Morris, discussing string building during concatenation, this seems to be a problem caused by Microsoft, Inc.

It seems to happen in ALL Microsoft 6502 Basic's versions for OS65D version 3.2 and 3.3 to this date, as far as I know. This "bug" does not seem to occur in OSI 65U (latest version). This problem seems to HAUNT most all of the commodore line of computer basic's, CBM-PET...??? (not positive about Vic-20).

As stated by Jim Butterfield, in his article in COMPUTE!, September 1981, the newest 4.0 disk basic for CBM does have the problem as described. I also have tried the CBM 4.0 basic, and it does not seem to be an isolated "bug" only on just one computer system.

It's anybody's guess how many other versions of MICROSOFT BASIC's have this flaw. I have tested TRS-80 Level III (ROM version) and The Color-80 computers at a local Radio Shack Store, and they both seem to work fine. Is it possible that ONLY some of the 6502 BASIC's have this problem?

Also, as stated by Jim Butterfield, in his article in COMPUTE!, September 1981; if Mr. Morris's program listed in PEEK (65), January 1982, is modified by changing line number 120, from:

```
120 Z$=Z$+"X"+"="
```

TO:

```
120 Z$=Z$+"X":Z$=Z$+"="
```

the problem will not occur at all. Very interesting, to say the least. I just thought that other PEEKer's might like to know about this.

Al Adams
Midland, MI 48640

* * * * *

ED:

In response to Roger Clegg's letter published in the March 1982 issue, I am submitting two routines which will round financial data to two trailing decimal places. This has been used under 65U V1.2, Level I or III:

```
97 REM=====
98 REM SUB: ROUND VARIABLES
99 REM=====
```

```
100 X=VAL(X$)
110 IF X=0 THEN X$="
    0.00":GOTO 130
120 POKE 9734,129:X$=STR$(X+
    .005*SGN(X))
130 RETURN
```

Some notes about this routine follow:

(1) The POKE is explained in the OSI Tech Notes. It's purpose is to force the \$R, \$L "Money-Mode" upon the next string variable encountered following the POKE. Using the 129 value will return a right justified, two trailing decimal, rounded number, represented by a 13 character string X\$. Using the 128 value will produce a similar value, although being left justified and a varying length.

(2) The entry point at line 100 will convert a string X\$ to a numeric X before proceeding. When the entry at line 110 is used, the value for X is presumed to be set prior to entering the routine.

(3) This POKE is "self-canceling" (i.e. it does not require another POKE to reset the system to "normal" conditions). One must remember that it operates on the first string set after the POKE.

This routine will first round up positive numbers by first adding .005 and rounds down negative numbers by adding -.005. The result is then truncated beyond the second trailing decimal. All negative signs will all line up in the 1st character position while the decimals will line up in the 12th and 13th character positions. If you do not care whether the signs line up and/or you wish the string X\$ to be set to some other length, use the following routine:

```
10 B$="          ":L=10
97 REM=====
98 REM SUB: ROUND VARIABLES
99 REM=====
100 X=VAL(X$)
110 IF X=0 THEN X$="0.00":
    GOTO 130
120 POKE 9734,128:X$=STR$(X+
    .005*SGN(X))
130 X$=RIGHT$(B$+X$,L)
140 RETURN
```

The above routine will permit you to specify the length of X\$ by setting L in line 10. Note that L must be greater than or equal to 5 or the smallest anticipated value (counting the sign, the first digit, the decimal point, and the two trailing digits) or else you may truncate the

front of the number. The sign will appear immediately to the left of the value, not in the 1st character position. I hope that either of these routines can be useful to Mr. Clegg.

With regards to the setting of the variable "L" (the length of the string X\$) for the second routine. L should be set to 5 for the values up to 9.99, to 6 for values from 10.00 up to 99.99, etc.

I enjoy PEEK (65) very much and have found many useful routines and advice in the past. Perhaps you or your readers can help with a bug discovered in BPSORT, a machine language sort routine purchased from BPS of New York City.

After some minor layout changes in the OSI Key file, we were able to find a real performance improvement over the slow, basic sorting program used in OS-DMS (naturally since BPSORT is a machine language sort within RAM). However, when sorting a keyfile containing 256 records (or 512 or any integer multiple of 256), I have found that the sort ignores the last record found in the keyfile. The result is a bad sort, with the last record (originally) left in its original position in the sorted result. There must be a sign mistake somewhere in the code since the sort works perfectly on any other number of records. Any help would be appreciated.

Finally, I wish to empathize with Mr. McKay's comments regarding OSI local support. Our original OSI supplier also went "by the boards" and further local support cannot be assured. Thank god that OSI now answers their phone when users call for aid. Our compliments to Mr. Jim Halverson for his help in the past; we have found that Jim will return our calls with helpful advice whenever we call.

Ronald Bartlett
Pittsburgh, PA 15224

Ronald:

Thanks very much for the routines. Regarding BPSORT: it was written by Phillip Woellhoff, who is now with Westico, Inc., 25 Van Zant St., Norwalk, CT 06855. He just told me he knows the bug and the fix. Write or call him and you will find him quite helpful. Or, get KYUTIL (available from DBMS, Inc.)

and forget your key file problems!

Al

* * * * *

ED:

To shorten OS65D V3.3 BEXEC* to allow for additional statements. <free 120+ positions>

Add line 60:

60 GOTO 100

Add line 70 thru 74:

```
70 INPUT" and depress RETURN
  (1-6 characters)";A$
71 IF A$=" " THEN 74
72 IF LEN(A$)>6 THEN 70
73 IF LEN(A$)<6 THEN A$=A$+
  ": GOTO 73
74 RETURN
```

Retype line 2016:

2016 GOSUB 70

Retype line 3016:

3016 GOSUB 70

Delete line 3017 and 3020

Retype line 3045:

3045 GOSUB 70

Delete line 3047 and 3050

Retype line 4016:

4016 GOSUB 70

Delete line 4017 and 4020

Change line 5105:

```
5105 K$="TUTORIAL":L$="BLANK"
      :GOTO 5506
```

Delete 5110

Change line 5505:

```
5505 K$="BLANK":L$="TUTORIAL"
```

Change line 5506:

```
5506 PRINT"REMOVE YOUR "K$"
      DISKETTE FROM DRIVE A
      AND":PRINT
```

Change line 5510:

```
5510 PRINT"REPLACE IT WITH
      YOUR"L$"DISKETTE.":PRINT
      :GOTO10200
```

Also, when using PRINT#1,(80) V3.3 screen dump with black and white 540 video board (IE C2-4P/C8) - POKE 12813,0 - to disable inverted character stripes - POKE 12813,<ORIGINAL VALUE> to enable.

Also, when desiring a response

from OSI on any V3.3 questions, state the serial number of your OS65D 3.3 copy. I have received 2 responses this way, versus none without stating my OS65D V3.3 SN#.

Craig D. Lombard.
Olympia, WA 98506

* * * * *

ED:

To quote a sentence of your subject article "When I saw this, I was very excited." After reading the article while eating lunch, I decided immediately after returning to the office I would key in your program routine, in STAT03 and then put it into my DMS-Inventory package (in place of INVL).

Before having to give up to go to other business demands, I spent two days trying to get it to work.

Your article was very good to both clarify what your program was doing plus a simple listing for me to key in. However, as I have learned with most OSI, there is some "simple" thing missing. Unfortunately, what is simple to most is usually monumental and time consuming to me.

Would it be possible for you to give me a clue as to the following so that I can use the program?:

- 1) location to put the code
- 2) the simple code to include

Also, the ability to remove report parameters if desired would be beneficial to this program.

Should line 5210 be AG\$ instead of AG?

C. Alan Skoog
Chadwick, IL 61014

Alan:
Mea Culpa, Mea Culpa. I start off writing an article for inexperienced computer users, then assume they will have enough experience to finish the job!

Here are the steps needed to save any new report you input on the list of reports:

- 1) Retype line 4030 as line 4039. Don't change the line, just retype it with the new number. The reason is that when you are all through selecting the options for the report, it goes to line 4030 to run the actual report writer program.

Now enter these lines:

```
4030 INPUT"Save this report on
      the list of reports";SA$
4031 IF LEFT$(SA$,1)<>"Y"THEN
      GOTO 4039:REM JUST RUN
4032 OPEN"STAT3B","PASS",1
4033 INDEX<1>=10:INPUT%1,REDF
4034 GOTO 4120:REM SAVE
```

This section of code will ask you if you want to save the report on the list of reports, then if you say you do by answering with anything which starts with "Y" it will skip to a later section of the program and stuff the information you have selected into STAT3B past the previous end of the data in the file.

Removing reports from the list is a bit harder, but not much. It should be done in another, separate program. Here is a program which will do the job:

```
10 PRINT"PROGRAM TO REMOVE
      REPORTS FROM THE LIST"
20 OPEN"STAT3B","PASS",1
25 WI=44:REM WRITE INDEX
30 FIND"^^",1
35 IFINDEX(1)>1E8THEN200
40 INPUT%1,A$:RI=INDEX(1)
50 PRINTA$
60 INPUT"KEEP THIS ONE";QA$
70 IFLEFT$(QA$,1)<>"Y"THEN30
75 INDEX<1>=WI:PRINT%1,A$
      WI=INDEX(1)
80 INDEX<1>=RI:INPUT%1,A$
90 RI=INDEX(1)
100 IFLEFT$(A$,2)="^^"
      THEN50
130 GOTO 80
200 CLOSE:END
```

What is happening here? First, we open up the file with the report parameters, and line 30 finds the first report title (titles are marked with "^^" at the beginning of the title). We then read in the title as A\$, set a number called RI (for the Read Index, the position in the file where we are reading data). The Write Index has already been set at 44, the position where the first title is always located in the file.

Next, we ask if you want to keep this report. If you don't say yes, we just go back to line 30 and FIND the next report title.

If you do want to keep this report, we have to read all the strings from the file with the information on the report and then re-write them back toward the beginning of the file. That is why we do the INPUT%1 and PRINT%1, and why we reset the RI and WI each time.

Note that the only way to get out of this program is to go past the end of the data in the file, at which time the next FIND command will come back with the INDEX set at 1E9 (1,000,000,000) and the program will CLOSE the file and END. It might also be nice to be able to get out "in the middle" in case you knew you would want to keep all the rest of the files, but that would require enough more code that it makes my head hurt. Readers, how would you do it?

Al

* * * * *

ED:

I appreciate the kind words for my SMARTERM program published occasionally in PEEK (65). This program was sold and supported through my local OSI dealer, Computer Connections of Farmington, MI, for the past year in both eight inch and minifloppy versions. Unfortunately, this dealer has dropped the OSI line and has therefore lost all interest in my terminal routines. A review of the sales for last year shows insufficient demand to warrant the interest of another software vendor especially when one considers the level of support, required by users attempting to get computers, modems, telephone lines, printers, remote services, and software all working together simultaneously! I have an eight inch disk based system and can therefore provide copies on eight inch floppies, but am unable to support minifloppy users. Computer Connections was kind enough to provide me with their existing program copies, so I do have a few minifloppy disks available. I propose the following:

I will provide copies of SMARTERM on minifloppy diskettes as long as the supply lasts for twenty bucks a copy including mailing costs. Requests must be prepaid.

I release SMARTERM into the public domain and invite your readers to make copies for other users.

I started work on the first version of SMARTERM over two years ago. Since then numerous improvements have been incorporated. Currently I am using a version I call STOS for Smart Terminal Operating System. SMARTERM was written primarily in BASIC with embedded machine code to do the fast work. STOS is written

entirely in assembler and resides in the region of memory used for the BASIC Interpreter, the Assembler/Editor, etc. That way the entire memory space normally used for working storage can be used as a buffer and the normal DOS commands can be used to save and recall files. Instead of SMARTERM's menu, STOS includes a command interpreter that currently recognizes over twenty-five commands including synonyms, and more can be easily added. STOS can send or receive files from or to the buffer and can convert simple WP2 files to ASCII for transmission. If anyone wants a copy of STOS for a polled keyboard eight inch disk system, I will send one to them in return for twenty-five bucks. Users who have a need for terminal software for non polled keyboard configurations should note that Jim Sanders is advertising Smart Terminal Software for OSI Serial systems for \$27.50! I wish to retain marketing rights for STOS for a bit, at least until OSI comes out with something better.

I invite users of either SMARTERM or STOS to share experiences or problems with me. One of the most convenient ways for me to communicate is via the EMAIL service on Compuserve. My user Id. is 70150,144.

Phil Lindquist
8892 Cooley Lake Rd.
Union Lake, MI 48085

* * * * *

ED:

I have been a HEXDOS user for about a year. When I received HEXDOS version 2.3, my system consisted of 12k memory and the CEGMON ROM. I had also ordered equipment to upgrade my system to 16K and a single disk system.

When I had everything installed, HEXDOS version 2.3 would not work with the CLE. The way I found that out was by loading PICO-DOS okay. Well I switched ROMS back to SYNMON and oh boy I had a 3K disk operating system on a 16K CLE WOW!

The editing facilities were not like those in the CLE, but it worked. I immediately changed all my programs from tape to disk.

To make a long story short, I was very happy when Mr. Hendrix made me aware of the new version 3.0 for the CEGMON

ROM, and that it would also allow for dynamic naming of any input/output file. I have also used the renumberer, and have found it to be very easy to use with the proper instructions.

I have converted PIMS (Personal Information Management System) by Scelbi to my system. I have also wired for the realtime clock, but I have not used it yet.

With the additional information that you have provided that I was not aware of, I intend to install the disk monitor control and the tone generator.

A final note. With the forthcoming assembler and FORTH, I will still be a very happy user.

William H. Moses
Mars, PA 16046

* * * * *

ED:

I would like to recommend the book "Beyond Games: System Software for Your 6502 Computer" by Ken Skier. I was most interested in Skier's general MOVE utility. This utility allows a block of memory to be moved from anywhere to anywhere without over-writing itself even when the source and destination blocks overlap. For example, a section of code can be moved up one byte in memory as well as one byte down. Some unusual effects are obtained by moving the video memory at \$D000 to \$D800. The video output can be scrolled sideways by doing a block move up or down one byte, or diagonally by moving 65 bytes.

The fast machine language MOVE utility is useful in video effects for games. For example, the entire screen can be moved in the direction indicated by a joystick to give the feeling of steering through a star field. By moving the top of the screen up and the bottom half down, the illusion of traveling forward is created. Skier has taken some trouble to make the routine fast.

If you want to merely use a good set of utility programs, buy OSI's extended monitor. If you want to understand utilities and incorporate these into your machine language programs, then get a copy of "Beyond Games".

Earl Morris
Midland, MI 48640 * * * * *

ED:

In the February issue of PEEK (65), page 3, you ran an article on USR(X). In the article it states that when passing values it will not work right on OS65D3.3 because of a bug. The fix was put into the article at \$4800 on up and one track on a mini-floppy disk of 8 pages has 2048 bytes of data which will end at \$5000. The article said to make the correction at \$51C2 and \$51C3 which is out of the range of the track we just loaded. What are the right locations?

Frank Soto
CBBS

Frank:

Mr. Magerman informs us that:
"The OSI Technical Support

Group says that no patch is needed for the 5-1/4" floppy copy of OS65D V3.3. HOWEVER, one of our members checked his copy and found that the patch WAS required. If it is, one should proceed as follows:

CA 4A00=05,1
@ 51C2
51C2/B2 18
51C3/20 12
SA 05,1=4A00/8

A1

* * * * *

ED:

I sent in two number converters (ref. PEEK letters, Dec. '81 and Jan. '82). Each of these may be shortened by three or four characters.

For: "IFH\$<>" write "IFH\$"
For: "IFD<>0" write "IFD"

Harry Hawkins
Burton, SC 29902

* * * * *

ADS

UNIVERSE (by Galaxian author) is an action game for all OSI systems on tape for rom systems and disk for 65D users. See the review in Cassette Corner that rates this program as worth your money. \$14.95 for tape or 5-1/4" disk. Other Quality Programs: EDIT-ALL is a full screen editor for 65D systems. Never retype a line again. Just type over (or dynamically insert/delete) text and our editor does the rest. Also includes full cursor control, screen clear and scroll windows. \$19.95 for 5-1/4" disk, \$22.95 for 8". Works with OS65D v3.x. TRACER will speed up your debugging of machine code programs. Trace and single step your system with break points in rom or ram. Works with OSI Extended Monitor. \$12.95 for 5-1/4" disk. All programs by: Dave (I don't write junk) Pompea. Soon to come: Alien Defenders - all machine code with smart bombs, scanner, scrolling mountains & more. Rom Dos II - replacement for OSI Pico Dos. Adds more storage per disk, faster loading and other usefull features. DMP Systems, 319 Hampton Blvd., Rochester, NY 14612.

* * * * *

OSI CLP with 8K memory, Aardvark monitor ROM and BASIC ROM to fix the garbage collection routine. The RS232 components have been added to the 600 board, but the necessary switch has not been installed. Will sell for \$300. Owen R. Brock, Route 5, Box 911, Burleson, TX 76028

* * * * *

FOR SALE: 5 year collection of OSI Equipment including two CLP's, a broken 610 board and 2 disc drives. There are many more items and many original programs both on disc and tape. Please send SASE for a listing and price to E.H. Brown, P.O. Box 2211, Warner Robins, GA 31099.

ED:

In reference to my letter published in the April 1982 issue, and the program submitted therewith. I discovered an error in the program which

caused me to rework it completely. The following is the new, improved version.

William E. Shawcross
Cambridge, MA 02138

```
10 REM -----
20 REM UTILITY TO SET UP A 5510 SPINWRITER AT $F500 AS DEVICE 5
30 REM (CENTRONICS-COMPATIBLE PARALLEL INTERFACE)
40 REM -----
50 REM
100 PRINT:INPUT"SPINWRITER OR PRINTRONIX AS DEVICE #5 (S OR P)";Q$
110 IF LEFT$(Q$,1)<>"S"THEN GOSUB3000:GOSUB4000:GOTO1000
120 REM *** THE FOLLOWING POKES INITIALIZE THE PIA AT $F500 ***
130 REM *** AND CHANGE THE I.O. DISTRIBUTOR DESTINATION ***
140 POKE62720,0:POKE62721,0:POKE62723,0:POKE62722,255:POKE62721,4
150 POKE62723,4:POKE62722,255:POKE11695,0:POKE15149,245
160 POKE15835,245:POKE15838,245:POKE15841,245:POKE15846,245
170 POKE15851,245:POKE15854,245:POKE15881,245:POKE15904,245
180 PRINT:PRINT"THE SPINWRITER IS NOW DEVICE #5":GOSUB4000:PRINT
190 REM -----
200 REM *** SET UP PITCH AND LINE SPACING ON SPINWRITER ***
210 INPUT"DO YOU WANT TO CHANGE FROM PICA PITCH (10 CPI), 6 LPI";Q$
220 IF LEFT$(Q$,1)<>"Y" THEN PRINT#5,CHR$(27)+"1L";:GOTO 1000
230 INPUT"ELITE PITCH (12 CPI) ";Q$
240 IF LEFT$(Q$,1)="Y" THEN PRINT#5,CHR$(27)+"1J";
250 INPUT"8 LPI ";Q$
260 IF LEFT$(Q$,1)="Y"THENPRINT#5,CHR$(27)+"1U";
1000 END
3000 REM -----
3010 REM *** THESE POKES RESET DEVICE 5 TO $F400 ***
3020 POKE15149,244:POKE15835,244:POKE15838,244
3030 POKE15841,244:POKE15846,244:POKE15851,244
3040 POKE15854,244:POKE15881,244:POKE15904,244
3050 RETURN
4000 REM -----
4010 PRINT:INPUT"RESET COUNTER TO TOP OF FORM ";Q$
4020 IF LEFT$(Q$,1)<>"Y" THEN GOTO 4040
4030 POKE 15908,60 : REM RESETS COUNTER TO TOP OF FORM
4040 RETURN
5000 REM =====
5010 REM FOR PITCH P (8 TO 120) AND LPI L (3 TO 48) USE THE
5020 REM FOLLOWING BY FIRST RUNNING SPIN, THEN TYPING "RUN 5000"
5030 REM -----
5040 INPUT"PITCH (8-120) ";P:INPUT"LINES PER INCH (3-48) ";L
5050 PRINT#5,CHR$(27)+"1"+CHR$(64+INT(120/P))+CHR$(27)+"1";
5060 PRINT#5,CHR$(79+INT(48/L));
5080 END
```

* * * * *

TACKLE YOUR PROGRAMMING PROBLEMS HEAD ON!



If operating bugs and "unsolvable" coding problems have got you dodging a big project, here's help . . . Quality software products that'll let you tackle those problems and move on to something easier.

At Software Consultants, we know the kinds of problems you often face with OSI gear, so we designed our software accordingly. Most of our products are modifications and/or extensions of OSI packages, but some are complete replacements. All are easily extensible for your own custom applications. We feel they're the finest OSI software products around - and the most reasonably priced.

PRODUCT ROUNDUP

OS-65D V3.2 DISASSEMBLY MANUAL

A super-complete manual that has it all. 50 pages of disassembly listings, complete and clear comments on all lines, 10 pages of computer generated cross reference listings, and more! Praised by many OSlers who couldn't believe it til they bought one. A deal at \$25.95.

REF COMMAND UNDER BASIC

A complete, cross reference utility that'll find and list any BASIC line number, variable, or numeric constant. It's available under 65D or 65U and comes on 8" floppies. This one will save your sanity, and cut out hours of wasted time. Yours for \$31.95.

SPOOLER/DESPOOLER UTILITY

A useful utility that feeds backed-up data to your printer for normal output, and leaves your screen free for other work by intercepting data bound for the printer and temporarily storing it on hard disk. Written in super fast machine language. Interfaces with serial and parallel printers. Only \$69.95.

FIG FORTH UNDER OS-65U

The only other language available for your 65U system. You get lots of extras, like terminal oriented editor, printer and terminal tools, and much more. Under multi-user, runs BASIC simultaneously too. Unbelievable at \$89.95.

VIDEO ROUTINE

This convenient extension for OS65D V1.2 really spices up your video system with little niceties like 24 separate control codes, horizontal and vertical plotting, and many variable screen parameters. Software extensions are available to connect this with the graphics resolution booster. The routine alone is \$25.95, with extensions, \$29.95.

GRAPHICS RESOLUTION BOOSTER

An ingenious piece of hardware that increases the graphic resolution per character by 8 times. With this device, your circles will be rounder and your curves smoother. A slick addition to your C4/8P system. Priced at \$49.95, or, with the video routine and extensions, \$79.95.

Unlike the majority of other software vendors, we offer our customers copies of source code (on floppies) for any of our products they've purchased. For a nominal fee of \$12, covering postage & handling, we'll send you the source code you choose.

Foreign orders please add \$10 postage fee for air mail. Dealer inquiries welcome. Tennessee residents add 6% sales tax. COD orders will incur a \$1.50 handling charge. We ship via UPS, so please include street address with your order.

*Write or call us today with your order, or ask
for our free product catalog
and get all the details.*

 **SOFTWARE
CONSULTANTS**
HARDWORKING software

6435 Summer Avenue
Memphis, TN 38134
901/377-3503

PEEK (65)

The Unofficial OSI Users Journal

P.O. Box 347
Owings Mills, Md. 21117

BULK RATE
U.S. POSTAGE
PAID
Owings Mills, MD
PERMIT NO. 18

DELIVER TO:

AGVNH980
1977

GOODIES for OSI Users!

DBMS, INC.

P.O. Box 347
Owings Mills, Md. 21117
(301) 383-3267

- C1P Sams Photo-Facts Manual.** Complete schematics, scope waveforms and board photos. All you need to be a C1P or SII Wizard, just \$7.95 \$ _____
- C4P Sams Photo-Facts Manual.** Includes pinouts, photos, schematics for the 502, 505, 527, 540 and 542 boards. A bargain at \$15.95 \$ _____
- C2/C3 Sams Photo-Facts Manual.** The facts you need to repair the larger OSI computers. Fat with useful information, but just \$39.00 \$ _____
- OSI's Small Systems Journals.** The complete set, July 1977 through April 1978, bound and reproduced by PEEK (65). Full set only \$15.00 \$ _____
- Terminal Extensions Package** - lets you program like the mini-users do, with direct cursor positioning, mnemonics and a number formatting function much more powerful than a mere "print using." Requires 65U. \$50.00 \$ _____
- RESEQ** - BASIC program resequencer plus much more. Global changes, tables of bad references, **GOSUBs** & **GOTOs**, variables by line number, resequences parts of programs or entire programs, handles line 50000 trap. Best debug tool I've seen. **MACHINE LANGUAGE - VERY FAST!** Requires 65U. Manual & samples only, \$5.00 Everything for \$50.00 \$ _____
- Sanders Machine Language Sort/Merge** for OS-65U. Complete disk sort and merge, documentation shows you how to call from any BASIC program on any disk and return it or any other BASIC program on any disk, floppy or hard. Most versatile disk sort yet. Will run under LEVEL I, II, or III. It should cost more but Sanders says, "...sell it for just..." \$89.00 \$ _____
- KYUTIL** - The ultimate OS-DMS keyfile utility package. This implementation of Sander's SORT/MERGE creates, loads and sorts multiple-field, conditionally loaded keyfiles. KYUTIL will load and sort a keyfile of over 15000 ZIP codes in under three hours. Never sort another Master File. \$100.00 _____

Cash enclosed Master Charge VISA
 C.O.D., add \$1.50

Account No. _____ Expiration Date _____
 Signature _____
 Name _____
 Street _____
 City _____ State _____ Zip _____

TOTAL \$ _____
 MD Residents add 5% Tax \$ _____
 C.O.D. orders add \$1.50 \$ _____
 Postage & Handling \$ 3.00
 TOTAL DUE \$ _____
 POSTAGE MAY VARY FOR OVERSEAS