PEEK (65)

The Unofficial OSI Users Journal

$1.75

JUNE 1984
VOL.5, NO.6

P.O. Box 347
Owings Mills, Md. 21117
(301) 363-3268

INSIDE

OSI ROM ROUTINES - PART 2 2
DOUBLE SIDED DRIVES FOR OSI 3
BEGINNERS COLUMN 4
EPROM PROGRAMMER 8
BUILD RS-232 COMPATIBLE MODEM 14
BEXEC* PROG.(HOOKS INTO BASIC) 16
RT. JUST. FOR DISK WP65Ø2 V1.2 19
SIMPLE A-B SWITCH 19

Column One

What's new at ISOTRON? Because that is one of the most frequent questions we are asked, we will continue to give you everything that we can get our hands on. But you probably won't hear about the wondrous things promised for next January. ISOTRON is more careful than its OSI predecessors, out of justified fear of premature announcements and commitments that are not kept. So far, just about everything that ISOTRON has promised has come to be.

This month's news is not hardware related - not that there are no new things on the drawing boards for the fall. The big splash is Comdex, advertising, and "bundled" turn-key systems.

The ISOTRON booth is a big double-plus affair, manned by some seventeen staffers, showing off the Data South printers and Esprit terminals (sold under OSI's name). They will be running new OSI vertical market packages (a medical system is reported to be among the first and a newly improved word processor for 65Ø2), plus software packages by Cyma Software. A number of other packages are also under review. This is just part of the new "bundled" turn-key systems that ISOTRON has put together at reduced prices that should please both end-users as well as dealers.

President, Bob Lewis, has been busy burning the candle at both ends trying to put OSI back on the map. Now that their flashy four-color ads have been in Computer World, Computer Dealer, Retailer, Merchandise, and Business Computer Systems, plans are now under way for fall ads in consumer pubs (like "Inc." maybe). The list of new and potential dealers has been growing rapidly. We understand that these new dealers are not "basement" dealers, but, for the most part, existing dealers taking on the OSI line. Bob plans to visit each new dealer personally. As soon as the dealer list becomes available, you will see it in PEEK as a service to aid you in getting the supplies and assistance you need. So that's some of what is keeping OSI alive and well.

What keeps PEEK well is articles from you; the experts in the field. Your creativeness and understanding have produced articles of lasting importance and have made PEEK "the" reference source. What we need to do is convince those of you who have not written, that you do not have to be a literary genius, nor should you be shy about putting your thoughts, findings, experiences, or suggestions on paper. We will give you all the help we can to polish up any rough spots. Above all, don't feel that you don't know enough. You have been at it for quite a while and there must be something that you can share with a beginner (you would be surprised how many calls we get from beginners needing the simplest kinds of explanation). Of course, we still need the more advanced articles as well. While you are in the writing mood, let us hear from you as to what kinds of articles you want and need.

For some reason, the hardest type of articles for us to come by are those that are business oriented. It seems that you business users spend all of your time "using" the machine. But, I'll bet that 90% of you have a word processor and could zip out some very interesting comments in less time than it has taken to write this column!

Remember the "Software" issues last October and November? The response was good, and by popular demand, we are going to do it again. Just watch for the form at the back of next month's issue. Don't let us down!

In the meantime, we are putting together a plan to make this software available thru PEEK. That way we can provide authors with a marketplace and buyers with one common source for the programs that will make your machine perform. Those of you with programs for sale, please write.

Eddie

OSI ROM ROUTINES

PART 2

Part 1 published January,1984

by: Leroy Erickson
Courtesy of OSMOSUS NEWS
3128 Silver Lake Road
Minneapolis, MN 55418

This month's ROM routine is SYNMON page 2, the scanned keyboard driver. First though, is a little description of the hardware that it is scanning.

OSI's keyboard is layed out as an 8 by 8 matrix, or 8 rows of 8 columns. When a byte is written to address $DF00, each bit corresponds to one of the 8 rows. When address $DF00 is read, each bit corresponds to one of the 8 columns for the selected row. In each case, the lowest order bit is row or column 0, and the highest order bit is row or column 7. A bit being high for a write selects that row, and a bit being high for a read indicates that the corresponding key is down. (On a C1P, an inverter is not present which is in the C4P design, thus a bit being 0 selects a row or indicates a key strike.) To do a valid keyboard scan, only one row should be selected at a time. Multiple key strokes are indicated by multiple bits set in one row, or by 1 or more bits being set in more than one row. Figure 1 shows how the keys are layed out in the matrix. Note that there are only 52 defined keys, so there are 12 open spots in the matrix.

Now for the software! The routine starts with row 0 and scans successively higher rows until a non-zero value is detected. On this first scan, row 0 is ignored except for the 'escape' key. If a key in rows 1 through 7 is detected then the row and column numbers of the key are evaluated,

Copyright ©1984 by PEEK (65) Inc. All Rights Reserved.

published monthly

Editor - Al Peabody
Technical Editor - Brian Hartson
Circulation & Advertising Mgr. - Karin Q. Gieske
Production Dept. - A. Fusselbaugh, 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.

```
                ;
                ; ***************************************
                ; ***                                 ***
                ; ***       C4P BOOT ROM PAGE 2       ***
                ; ***                                 ***
                ; *** SCANNED KEYBOARD DRIVER ROUTINE ***
                ; ***                                 ***
                ; ***   Comments by  Leroy Erickson   ***
                ; ***           March  1982           ***
                ; ***                                 ***
                ; ***************************************
                ;
0213=           TMPCHR=$0213
0214=           WATCNT=$0214
0215=           CURCHR=$0215
0216=           LSTCHR=$0216
DF00=           KEYBRD=$DF00
                ;
FD00                    *=$FD00
                ;
FD00 8A         GETCHR  TXA             ; SAVE X & Y ON STACK
FD01 48                 PHA             ;
FD02 98                 TYA             ;
FD03 48                 PHA             ;
                                        ;
FD04 A901       NEWSCN  LDA     #$01    ; BEGIN FULL SCAN - SET ROW 1
FD06 8D00DF     ROWLUP  STA     KEYBRD  ; WRITE ROW SELECT
FD09 AE00DF             LDX     KEYBRD  ; READ COLUMN VALUES
FD0C D005               BNE     KEYFND  ; SKIP IF ANY KEY IS DOWN
FD0E 0A         NXTROW  ASL     A       ; ELSE, SELECT NEXT HIGHER ROW
FD0F D0F5               BNE     ROWLUP  ;  AND LOOP FOR 8 ROWS
FD11 F053               BEQ     CLEAR   ; SKIP WHEN ALL 8 ROWS ARE DONE
                                        ;
FD13 4A         KEYFND  LSR     A       ; GET ROW # / 2
FD14 9009               BCC     TRUCHR  ; IF NOT CTRL ROW, SKIP
FD16 2A                 ROL     A       ; IF CTRL ROW, RESTORE IT
FD17 E021               CPX     #$21    ; IS IT ESCAPE?
FD19 D0F3               BNE     NXTROW  ; NO, CHECK NEXT ROW
FD1B A91B               LDA     #$1B    ; YES, SET A = ESCAPE VALUE
FD1D D021               BNE     DEBNCE  ; GO DEBOUNCE IT
                                        ;
FD1F 20C8FD     TRUCHR  JSR     GETVAL  ; GET Y=ROW #
FD22 98                 TYA             ; A=ROW #
FD23 8D1302             STA     TMPCHR  ; SAVE IT TEMPORARILY
FD26 0A                 ASL     A       ; * 2
FD27 0A                 ASL     A       ; * 4
FD28 0A                 ASL     A       ; * 8
FD29 38                 SEC             ;
FD2A ED1302             SBC     TMPCHR  ; A = ROW # * 7
FD2D 8D1302             STA     TMPCHR  ; SAVE THIS VALUE
FD30 8A                 TXA             ; A = COLUMN IMAGE
FD31 4A                 LSR     A       ;  / 2
FD32 20C8FD             JSR     GETVAL  ; Y = COLUMN #
FD35 D02F               BNE     CLEAR   ; A NE 0 ==> MULTIPLE STRIKE
FD37 18                 CLC             ;
FD38 98                 TYA             ; ELSE, A = COLUMN #
FD39 6D1302             ADC     TMPCHR  ; A=7*ROW + COL
FD3C A8                 TAY             ; MOVE IT INTO Y
FD3D B9CFFD             LDA     KEYLST,Y ; GET INDICATED KEY VALUE
                                        ;
FD40 CD1502     DEBNCE  CMP     CURCHR  ; SAME AS LAST CHAR ?
FD43 D026               BNE     CLEAR1  ; NO, RESET FLAGS & SCAN AGAIN
FD45 CE1402             DEC     WATCNT  ; YES, DECR WAIT COUNT
FD48 F02B               BEQ     CHRFND  ; DONE, EXIT SCAN LOOP
FD4A A005               LDY     #$05    ; ELSE, WAIT 5 MILLI-SEC
FD4C A2C8       WATLP1  LDX     #$C8    ; INNER LOOP COUNT = 200
FD4E CA         WATLP2  DEX             ; WAIT 200 * 5 CLOCK CYCLES
FD4F D0FD               BNE     WATLP2  ;
FD51 88                 DEY             ; WAIT 5 * 1000 CLOCK CYCLES
FD52 D0F8               BNE     WATLP1  ;  ==> 5 MS ON A 1 MHZ SYSTEM
FD54 F0AE               BEQ     NEWSCN  ; THEN GO SCAN AGAIN
                                        ;
FD56 C901       TSTSHF  CMP     #$01    ; RIGHT SHIFT ?
FD58 F035               BEQ     UPRCAS  ; YES, SKIP AHEAD
FD5A A000               LDY     #$00    ; ELSE, SET Y=0
FD5C C902               CMP     #$02    ; LEFT SHIFT ?
FD5E F047               BEQ     ADJUST  ; YES, SKIP AHEAD
FD60 A0C0               LDY     #$C0    ; NO, SET Y = $C0
FD62 C920               CMP     #$20    ; CTRL ?
FD64 F041               BEQ     ADJUST  ; YES, SKIP AHEAD
                                        ;
FD66 A900       CLEAR   LDA     #$00    ; NONE OF THE ABOVE, CLEAR A
FD68 8D1602             STA     LSTCHR  ; CLEAR LAST CHAR
FD6B 8D1502     CLEAR1  STA     CURCHR  ; CLEAR CURRENT CHAR
FD6E A902               LDA     #$02    ; SET DEBOUNCE COUNTER
FD70 8D1402             STA     WATCNT  ;
FD73 D08F               BNE     NEWSCN  ; GO START A FRESH SCAN
                                        ;
FD75 A296       CHRFND  LDX     #$96    ; GET LARGE WAIT COUNT
FD77 CD1602             CMP     LSTCHR  ; IS THIS CHAR = LAST CHAR ?
FD7A D002               BNE     CHRFN1  ; NO, KEEP LARGE WAIT COUNT
FD7C A214               LDX     #$14    ; YES, GET 30CPS REPEAT COUNT
FD7E 8E1402     CHRFN1  STX     WATCNT  ; SET NEW REPEAT COUNT
FD81 8D1602             STA     LSTCHR  ; SAVE THIS CHAR AS LAST CHAR
FD84 A901               LDA     #$01    ; TEST CTRL/SHFT ROW
FD86 8D00DF             STA     KEYBRD  ;
FD89 AD00DF             LDA     KEYBRD  ;
FD8C 4A                 LSR     A       ; SHIFT LOCK ?
FD8D 9033               BCC     NOTLCK  ; NO, SKIP AHEAD
                                        ;
FD8F AA         UPRCAS  TAX             ; YES, SAVE COLUMN VALUES
FD90 2903               AND     #$03    ; STRIP OFF ALL BUT SHIFT KEYS
FD92 F00B               BEQ     TSTCTL  ; NO SHIFTS DOWN, SKIP
FD94 A010               LDY     #$10    ; ELSE, SET Y=$10
FD96 AD1502             LDA     CURCHR  ; GET CURRENT CHAR
FD99 100C               BPL     ADJUST  ; SKIP IF HI BIT NOT SET
```

Continued

```
107 FD9B A0F0            LDY     #$F0       ; ELSE, SET Y = - $10
108 FD9D D008            BNE     ADJUST     ; THEN SKIP
109                                         ;
110 FD9F A000    TSTCTL  LDY     #$00       ; NO SHIFTS, CLEAR Y
111 FDA1 E020            CPX     #$20       ; CTRL DOWN ?
112 FDA3 D002            BNE     ADJUST     ; NO, SKIP
113 FDA5 A0C0            LDY     #$C0       ; ELSE, SET Y = - $40
114                                         ;
115 FDA7 AD1502  ADJUST  LDA     CURCHR     ; GET CURRENT CHAR
116 FDAA 297F            AND     #$7F       ; STRIP HI BIT
117 FDAC C920            CMP     #$20       ; IS IT SPACE ?
118 FDAE F007            BEQ     ADJST1     ; YES, SKIP ON
119 FDB0 8C1302          STY     TMPCHR     ; ELSE, SAVE CHANGE VALUE
120 FDB3 18              CLC                ;
121 FDB4 6D1302          ADC     TMPCHR     ; CHANGE CHAR BY SHFT/CTRL VALUE
122 FDB7 8D1302  ADJST1  STA     TMPCHR     ;  AND SAVE NEW VALUE
123 FDBA 68              PLA                ; RESTORE Y & X
124 FDBB A8              TAY                ;
125 FDBC 68              PLA                ;
126 FDBD AA              TAX                ;
127 FDBE AD1302          LDA     TMPCHR     ; REGAIN NEW CHAR
128 FDC1 60              RTS                ; GO HOME ( WHEW! )
129                                         ;
130                                         ; * SHIFT LOCK NOT DOWN *
131                                         ;
132 FDC2 D092    NOTLCK BNE     TSTSHF     ; OTHER KEYS ARE DOWN, GO BACK
133 FDC4 A020            LDY     #$20       ; ELSE, SET LOWER CASE MOD
134 FDC6 D0DF            BNE     ADJUST     ; GO MODIFY THE CHAR
135                                         ;
136                                         ; * GET ROW & COLUMN NUMBERS *
137                                         ;
138 FDC8 A008    GETVAL LDY     #$08       ; SET FOR 8 BITS
139 FDCA 88      GETVL1 DEY                 ; DECR BIT COUNTER
140 FDCB 0A              ASL     A          ; SHIFT OUT LEFTMOST BIT
141 FDCC 90FC            BCC     GETVL1     ; LOOP UNTIL HIGHEST BIT FOUND
142 FDCE 60              RTS                ; THEN RETURN
143               ;
144               ; *** KEY VALUE LIST ***
145               ;
146 FDCF D0      KEYLST .BYTE $D0,$BB,$2F,$20,$5A,$41,$51 ; P ; /   Z A Q
146 FDD0 BB
146 FDD1 2F
146 FDD2 20
146 FDD3 5A
146 FDD4 41
146 FDD5 51
147 FDD6 2C              .BYTE $2C,$4D,$4E,$42,$56,$43,$58 ; , M N B V C X
147 FDD7 4D
147 FDD8 4E
147 FDD9 42
147 FDDA 56
147 FDDB 43
147 FDDC 58
148 FDDD 4B              .BYTE $4B,$4A,$48,$47,$46,$44,$53 ; K J H G F D S
148 FDDE 4A
148 FDDF 48
148 FDE0 47
148 FDE1 46
148 FDE2 44
148 FDE3 53
149 FDE4 49              .BYTE $49,$55,$59,$54,$52,$45,$57 ; I U Y T R E W
149 FDE5 55
149 FDE6 59
149 FDE7 54
149 FDE8 52
149 FDE9 45
149 FDEA 57
150 FDEB 00              .BYTE $00,$00,$0D,$0A,$4F,$4C,$2E ;    crlf O L .
150 FDEC 00
150 FDED 0D
150 FDEE 0A
150 FDEF 4F
150 FDF0 4C
150 FDF1 2E
151 FDF2 00              .BYTE $00,$FF,$2D,$BA,$30,$B9,$B8 ;  rb - : 0 9 8
151 FDF3 FF
151 FDF4 2D
151 FDF5 BA
151 FDF6 30
151 FDF7 B9
151 FDF8 B8
152 FDF9 B7              .BYTE $B7,$B6,$B5,$B4,$B3,$B2,$B1 ; 7 6 5 4 3 2 1
152 FDFA B6
152 FDFB B5
152 FDFC B4
152 FDFD B3
152 FDFE B2
152 FDFF B1
153                      .END
```

FIGURE 1: OSI Keyboard Matrix

	C7	C6	C5	C4	C3	C2	C1	C0
	D7	D6	D5	D4	D3	D2	D1	D0
	L7	L6	L5	L4	L3	L2	L1	L0
	7	6	5	4	3	2	1	0
ROW 7	1	2	3	4	5	6	7	
ROW 6	0	9	8	:	-	cr/lf		
ROW 5	.	L	O	P	cr			
ROW 4	M	E	R	T	Y	U	I	
ROW 3	B	D	F	G	H	J	K	
ROW 2	I	C	V	S	N	M	,	
ROW 1	0	A	Z	sp	/	;	P	
ROW 0	rpt	ctl	lsc					

converted into an index into a table, and the corresponding value from the table is loaded. The shift lock, shift, and control keys are taken into account to adjust the table value, and the resulting ASCII code is returned to the calling routine.

Multiple key closures in one row are detected and ignored as an error. Multiple key closures in multiple rows, though, are not detected. In that case, the lowest numbered row with a key down is found first and that key is evaluated. The higher numbered rows are never even reached to show the error.

Another problem in this routine concerns lower case characters. When shift lock is off, lower case characters are available, but nothing else on the keyboard is correct! The numeric and special character keys decode incorrectly, the left and right shift keys respond differently, and even return doesn't work properly!

Another property of this routine is that it needs four RAM locations as temporary storage. When OSI wrote it, they decided to use locations $0213 - $0216, so these spots must be accounted for in all programs which you write. BASIC, OS65D, and the Assembler/Editor all avoid or compensate for them.

Well, have fun looking at the code. For a challenge, see if you can see how the debounce and auto-repeat timing counters work. See you next month.

★

DOUBLE SIDED DRIVES FOR OSI

By: Robert S. Baldassano
4045 Ashbrook Circle
San Jose, CA 95124

My Shugart 800 drives that came with my 8PDF had given me long and trouble free service for over three years, but during a recent OSISIG conference, when Evan Pomerantz of OSMOSUS told me about a chance to buy new double sided Shugart 850's for approximately $120 each, I couldn't resist. Heck, I had seen numerous articles on putting new drives on an OSI, and I had the SAMS manuals to show me how to modify the boards, so it should be a piece of cake, right?

Well, I was a little slow in making up my mind, and by the time I decided to say yes these new drives were gone!

It wasn't long before Evan found a new deal for new surplus drives. They were a little more money, about $139, and had no guarantee, but I decided to take a chance.

I sent Evan the money, and he checked the drives out before sending them to me, but not on an OSI machine. He did see a set running on a C3 and copied down the jumpering for me. and sent me an 850 manual as well.

Soon my drives arrived, and they looked beautiful. These babies had Bi-Compliant heads and a track-to-track access time of 3 ms. I couldn't wait to make the swap.

Since the 850's were supposed to be plug compatible with the 800's, I decided to use my old power supply and case. The drives are the same size as 800's, but the mounting system is slightly different i.e. the 800's had smooth sides and these had stand off bosses. This required cutting about 1/2 inch away from both sides of the front opening of the case to make it 9 1/2 inches wide. A hacksaw and file did this nicely. The same bracket mount was useable, but the mounting holes in the bottom of the case had to be moved a little.

Power requirements are the same, 85-127 VAC and +24 and +5 VDC. I only required two small changes here. The AC connector had been changed to an AMP P/N 1-480701-0 and required putting AMP P/N 1-480700-0 male connectors in place of the old ones on the power supply. They were readily available in an electronic supply house. Also, my power supply and old drives had the DC return for both supplies on one pin of connector J5, and the 850's used a separate return. I guess you could jumper this on the drive, but I added the separate 24V return.

The machanics done, I then turned to the drives themselves. Evan had given me a list of jumpers to connect on the printed circuit board of the drives themselves. He didn't tell me to cut any traces but I should have known better (more on that later). When the drives were jumpered, I then turned to my 505 board and modified that. It is an easy task to do, but SAMS doesn't make it really clear. Also, although the modification is essentially the same on the 470, 505 Rev A, and 505 Rev B boards. the components are in different locations. For my board, a 505 Rev B, I made these changes:

* Cut connection from U4B pin 3 to 68B21P pin 8.

* Connect U4B pin 3 to 68B21P pin 15.

* Cut connection from 68B21P pin 15 to U5A pin 12.

* Tie U5A pin 12 to the 5V line through a 4.7K resistor (R62).

* Connect 68B21P pin 15 to U5A pin 13.

* Cut connection from U4B pin 6 to U5A pin 13.

The result of these modifications is you will have changed Select Drive 1 to Drive Select, and Select Drive 2 to Side Select. It is then necessary to modify the Paddle board as shown on page 18 of the OSI Tech Newsletter #27 dated April 18, 1980.

Well I did all this, and hooked everything up, turned on the system and ---------- NOTHING!!!

After much fooling around, I found if I only hooked up one drive it would work if I jumpered it as disk 1, but only A side would work. So it was back to the books and modifications looking for a clue.

I checked every solder joint I made on the 505 board, and even traced the connections using an ohmmeter. I did find a few connections that needed rework, but that was not the real problem. I then traced all 50 pins on the cable to see that they were modified according to the OSI guidance and connected to the proper pins on the drives. Here I got my first clue that something was wrong, all 24 pins on the paddle board went to the right places but now OSI pin 12 (ground) was tied to pin 26 on the 850's (Drive Select 1) and OSI pin 3 (now Drive Select) was tied to the 850's pin 32 (Drive Select 4).

The way a drive knows it is selected is for its Drive Select line to go low. If I had Drive 1 jumpered it would always be selected and the other drive would never work. I still couldn't see how this system would work with only one drive select line, an inverter of some kind seemed to be necessary.

I tried the drives again, this time jumpered as Drive 4. With both drives hooked up nothing worked, but with one drive up, I got sides A and C. Things were looking up! It was time for the OSISIG conference again, and this time everyone was on trying to help me solve the problem. I still was convinced some inverter or flip flop was necessary on the paddle board so that when one drive was selected the other would be off.

Well my answer was in the making. On the SIG that night, Bob Ankeney of Generic Computer Products told me he was getting a C2OEM that had 850's in it, and if I called him that Saturday, he would tell me how it was jumpered.

Saturday night I made the call and he read me off the list of jumpers. Both drives were jumpered as DS4, and the jumpers to two IC's also had traces cut (I did not know about these cuts). So I ran to my work bench and started to make the changes.

I had jumpered IC 3C pin 7 to IC 3C pin 12, and now cut the trace to pin 12 on 3C. I was also supposed to jumper IC 4B pin 10 to IC 2E pin 9 and cut the trace to 2E. Here I found that I had jumpered to IC 2B in error! Things would surely work now!

Another try and still no luck. I called Bob again to make sure I got it all down right. As we talked, he noted other jumpers he had not seen and most important of all another IC jumper. IC 4D pin 8 was connected to IC 2E pin 13, and the trace from IC 4D pin 11 to IC 2E pin 13 was cut on the back of the board. This was only done on the B drive. Here was the inverter I expected. Drive Select and Side Select are NORed in gate 4D, the output pin 8 going to pins 12 and 13 of 4D which flips the output. This modification bypassed the conversion so that when A drive is selected B drive is not and vice versa.

We quickly checked the other jumpers again. They were S,R, I,A,B,DS4,850,IW,TS,S2,IT,AF,R S,Y,C, and DS on BOTH boards.

I made this last modification, hooked everything up, turned on the power and AT LAST IT ALL WORKED!!

I am now the proud owner of one Meg on line storage and I learned a lot about disk drives as well. I hope this little story will help the rest of you who may want double sided drives too.

★

IN THE BEGINNING

By: L. Z. Jankowski
Otaio Rd 1 Timaru
New Zealand

The program to be discussed is a 'Mailing List'. It is easily adaptable to tape and ROM

TRY US!

WE MEAN BUSINESS

SC5/80-1 Computer System

With 1 Denver Board Multi Processor
Wired for 6 Users
Expandable to 14 Users

$6990.00

For Each Add'l Multi Processor User Add $1,000.00

- **Includes Super Utility Package**

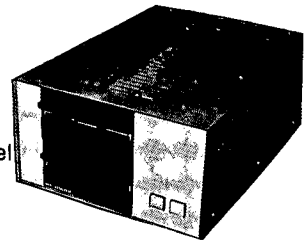
Plus our incredible new super system data base manager. With brand new Disk Tech One, 2 year warranty on internal hard disk components. Beautiful hand finished oak cabinet over steel frame. Instant access to all components. Available with casters or plastic feet

BEAUTIFUL AS WELL AS FUNCTIONAL

CLOSE OUT! C3-OEM

2 MHz/56K* With 6502 and Z80 Processors

With Centronics Parallel Printer Interface



While they Last!

$1499.00

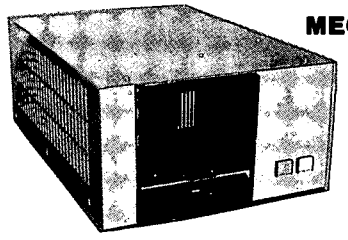
*Includes 8K Exec Memory for CP/M or Multi-User

- Buy for a Spare
- Add to a Spare H.D.
- Develop Programs

Spare 510CPU Board with Z80+6502 $169.00

MULTI PROCESSOR!

COMPLETE 10 MEGABYTE SYSTEM



$3999.00

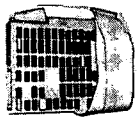
Boot From Hard or Floppy!

- 8" Hard Disk
- 8" Floppy Disk

Single user with Centronics parallel printer interface expandable to eight users.

Enclosed in table top cabinet as shown or mounted in deluxe floor cabinet, as above left. Add $400.00

NEW CONTROLLER/INTERFACE!



$599.00
Quantity 1

The new 9590 Controller replaces the OSI 590/525 set with 1 single board. Unlike others you've seen, it works perfectly with OSI or Denver boards, and all versions of 65U

It is compatible with old Style 592 Interface with 3 cable adapter, or attaches with single ribbon cable to our new 9592 H.D. Interface Board. Unlike others, our interface completely supports the ready/fault indicator on the 80 meg. hard disk.

9592
Quantity 1

$269.00

10 meg. Subsystem WOW!

$1999.00

Add to any existing OSI floppy based computer.

Just plug in one board and set this little gem on top or next to your computer and voila! Welcome to speed and convenience. Completely self contained with power supply. 9598 Hard Disk Controller also available separately at $699.00

12 SLOT BUS

Fits standard OSI mount holes can be connected in "T" or daisy chained by cable for expansion.

$79.00

DEALERS! **If you're still alive, call us and ask about our aggressive new price structure. We'll send you an unbelievable spares price list.**

SPACE-COM International
22991 La Cadena Drive, Laguna Hills, CA 92653 (714) 951-4648

BASIC, to 65D 3.2 and to 65D 3.3. The accent is on ease of use and flexibility for change. In fact, with minor changes the program could be used as a simple DBMS. Those are the two ways I use it.

Writing a long program is easy, but only if it is written in short blocks. If each block 'works' as the programmer intended, then the program must work. The big advantage of a block structure is that the various sections of the program are easily identified. Because this is true, program logic flow becomes obvious and testing the program is reduced to testing one small piece at a time. Editing is greatly simplified. Also, sections can be easily 'lifted' for use in other programs. All this and 8-bit OSI BASIC? Yes!

The 'Otaio Mailing List' (OML) was developed using these ideas. If you intend to type it in, use the line numbers as given. Leaving out all REM lines will save nearly 600 bytes. The spaces in the listing were inserted for clarity only. Spaces slow down long programs and reduce the number of statements that can be put on one line. The OML is useful for at least a file of 200 records and is adaptable to ROM or DISK BASIC. Features include sorting on any field, search and wild card search, and option on 2 column printout. Above all, it is easy to use!

THE PROGRAM

If the program is to run under DISK BASIC, create a 4-track file to hold the program, and a 2-track file to hold the mailing list data. (Run ZERO to fill the 2-track file). This should be sufficient for at least 100 names, addresses and zip codes. Now run CHANGE and ask for one buffer and 28 bytes before the BASIC workspace. Under 8" 65D 3.3 the BASIC workspace will then start at 18074; or at 15770 under 3.2. Subtract 27 from this value to get the value for X, in line 90. What's the 28th byte for? This byte (at $469A, $3D9A) holds the first null which starts off the BASIC program. So there's a bug in CHANGE? Yup! It is necessary to ask for one more byte in addition to what is required.

When testing the program, I found that on a second RUN, the program would 'hang' when reading a sequential file off disk. 'CLEAR' in line 10 fixes that. The PRINT!(28) is a

```
1 PRINT!(28):PRINT"THIS IS A DEMONSTRATION PROGRAM.":PRINT:PRINT
2 PRINT"IT WILL NOW LOAD SEQUENTIAL FILE --> 'MASQ00'"
3 FORX=1TO4000:NEXT
4 REM ******************************************************
10 CLEAR:PRINT!(28):REM COPYRIGHT by L.Z. JANKOWSKI APRIL '84
20 TRAP2010:POKE2888,0:POKE8722,0:POKE2972,13:POKE2976,13:POKE2073,96
30 :
40 B$="       OTAIO MAILING LIST 4/84 by LZJ"
50 X=53509:FORI=1TOLEN(B$)
60 Y$=RIGHT$(B$,I):FORY=LEN(Y$)TO1STEP-1:POKEX+Y,ASC(MID$(Y$,Y,1))
70 NEXTY,I
80 :
90 X=18047:FORY=XTOX+26:READQ:POKEY,Q:NEXT
100 DATA 72,138,72,152,72,160,0,169,32,153,0,210,153,0,209,153
110 DATA 0,208,200,208,244,104,168,104,170,104,96:TB=40:V=2
120 :
130 N=200:P=5:Z=0:ST=10:S=64:F$=CHR$(12):R$=" ":S$="STOP":H$="HELP"
140 DIMD$(N,P):C$=CHR$(13)
150 N$(1)="Name      ":N$(2)="Address  ":N$(3)="City      "
160 N$(4)="Phone     ":N$(5)="Computer ":N$(P+1)="Record #"
170 :
174 REM ******************************************************
175 Y$="MASQ00":GOTO350
180 REM Main Menu
190 PRINT!(28):PRINTTAB(11)"* When in trouble type:-  HELP *":PRINT
200 PRINT:PRINT:PRINT"Records free ==>"N-Z" from "N:PRINT
210 :
220 PRINT"   MAIN MENU":PRINT"   ---------":PRINT
230 PRINT"1> LOAD  File":PRINT"2> SAVE  File":PRINT"3> PACK  Records"
240 PRINT"4> FIND":PRINT"5> EDIT":PRINT"6> SORT":PRINT"7> PRINT"
250 PRINT"8> APPEND":PRINT"9> LIST Erased Record #":PRINT"-> END"
260 PRINT:PRINT"Choice ? ";:GOSUB310:IFY$="-"THEN1970
270 IFY=0THEN190
280 PRINT!(28):ONYGOTO340,410,480,570,880,1280,1490,1710,1830
290 :
300 REM Get a Key
310 DISK!"GO 2336":Y$=CHR$(PEEK(9059)):Y=VAL(Y$):A=PEEK(9059)OR32
315 RETURN
320 :
330 REM LOAD A FILE
340 INPUT"* Sequential File Name ";Y$:IFY$=H$ORY$=""THEN190
350 PRINT:PRINT"* Loading from DISK now *":Y=Z+1
360 DISK OPEN,6,Y$:INPUT#6,X:Z=Z+X:IFZ>NTHENZ=Z-X:GOTO380
370 FORQ=YTOZ:FORC=1TOP:INPUT#6,D$(Q,C):NEXTC,Q
380 DISK CLOSE,6:GOTO190
390 :
400 REM SAVE A FILE
410 INPUT"* File Name ";Y$:IFY$=H$ORY$=""THEN190
420 PRINT:PRINT"* Saving to DISK now *"
430 DISK OPEN,6,Y$:PRINT#6,Z
440 FORQ=1TOZ:FORC=1TOP:PRINT#6,D$(Q,C)C$;:NEXTC,Q
450 DISK CLOSE,6:GOTO190
460 :
470 REM PACK FILE
480 PRINT"Is the File SORTED ? ";:GOSUB310:PRINTY$:IFA=121THEN510
490 IFY$=""ORA=104THEN540
500 PRINT:PRINT"* SORT File first Bub! *":PRINT:GOTO200
510 PRINT!(28):PRINTTAB(20)"* PACKING *"
520 Q=Z
530 IFLEFT$(D$(Q,1),2)="ZZ"THENZ=Z-1:FORY=1TOP:D$(Q,Y)="":NEXT:GOTO520
540 GOTO190
550 :
560 REM FIND A RECORD
570 PRINT!(28):Q$="?":F=0:K=0:PRINT"Print Records to Device # ? ";
580 GOSUB310:V=Y:IFV=0THENV=2
590 :
600 PRINTV:PRINT:PRINT"    FIND MENU":PRINT"    ---------":PRINT
610 FORC=1TOP+1:PRINTSTR$(C)"> by "N$(C):NEXTC:PRINT" -> EXIT"
620 PRINT:PRINT"Choice ? ";:GOSUB310:PRINTC$;:M=Y
630 IFY$="-"THEN190
640 IFM=P+1THEN800
650 IFM=0ORM>P+1THEN570
660 :
670 PRINT"Wild Card Search ? ";:GOSUB310:PRINTC$;:IFA=121THENK=-1
680 :
690 PRINT"Which  * "N$(M);: INPUT"  ";T$:IFT$=H$ORT$=""THEN570
700 I=LEN(T$):PRINT!(28):PRINTTAB(20)"* SEARCHING *"
710 :
720 FORQ=1TOZ:Y$=D$(Q,M):IFKTHENGOSUB830
730 R=LEN(Y$)-I+1:FORX=1TOR:IFT$=MID$(Y$,X,I)THENX=R:NEXTX:GOTO780
740 NEXTX
750 IFQ=ZANDF=0THENPRINT:PRINTT$;" << Not found Bub! >>":PRINT:PRINT
760 GOTO790
770 :
780 PRINT:PRINT"Record "Q"of"Z:PRINT:GOSUB1880:F=-1
790 NEXTQ:PRINT"Ready ? ";:GOSUB310:GOTO570
800 PRINT!(28):INPUT"* Record # ";Q:IFQ<1ORQ>ZTHEN800
810 FORQ=QTOQ:GOTO780
820 :
830 B$="":R=LEN(Y$):FORY=1TOR:R$=MID$(T$,Y,1):IFR$=Q$THENR$=Q$:GOTO850
840 R$=MID$(Y$,Y,1)
850 B$=B$+R$:NEXTY:Y$=B$:RETURN
860 :
870 REM EDIT A FILE
880 R=0:IFZ=0THEN190
890 FORQ=1TOZ:IFLEFT$(D$(Q,1),2)="ZZ"THEN1120
900 IFRTHEN1000
910 :
920 PRINT!(28):FORY=1TO7:PRINT:NEXTY:GOSUB1220:Y=18
930 PRINTTAB(Y+3)"EDIT MENU":PRINTTAB(Y+3)"---------"
940 PRINTTAB(8)"Change:-"
950 FORC=1TOP:PRINTTAB(Y)STR$(C)"> "N$(C):NEXTC:PRINTTAB(8)"or,":Y=Y+1
960 PRINTTAB(Y)"6> Next Record":PRINTTAB(Y)"7> Previous Record"
970 PRINTTAB(Y)"8> Erase Record":PRINTTAB(Y)"9> Random Select"
980 PRINTTAB(Y)"-> EXIT":PRINT:GOTO1010
990 :
1000 DISK!"GO 467F":GOSUB1220
```

continued

THE DATA SYSTEM

- Stored Report Formats
- Stored Jobs, Formats, Calcs.
- Multiple Condition Reports
- Multiple File Reports
- Calc. Rules Massage Data
- Up to 100 Fields Per Record
- User Designed Entry/Edit Screens
- Powerful Editor
- Merges - Append, Overlay, Match
- Posting - Batch Input
- Nested Sorts - 6 Deep
- Abundant Utilities

HARDWARE REQUIREMENTS: 48K OSI, Hard Disk, serial system, OS-65U 1.42 or Later; Space required: 1.3 megabytes for programs and data.

PRICE: $650.00 (User Manual $35.00, credited towards TDS purchase). Michigan residents add 4% sales tax. 30 day free trial, if not satisfied, full refund upon return.

TIME & TASK PLANNER

30 DAY FREE TRIAL — IF NOT SATISFIED, FULL REFUND UPON RETURN

- "Daily Appointment Schedule"
- "Future Planning List" - sorted
- "To Do List" - by rank or date
- Work Sheets for all Aspects
- Year & Month Printed Calendar
- Transfers to Daily Schedule

A SIMPLE BUT POWERFUL TOOL FOR SUCCESS

HARDWARE: 48K OSI, 8" floppy or hard disk, serial terminal system, OS-65U v. 1.3 or later.

PRICE: $300.00 (User Manual, $25.00, credited toward TTP purchase). Michigan residents add 4% sales tax.

FINANCIAL PLANNER

- Loan/Annuity Analysis
- Annuity 'Due' Analysis
- Present/Future Value Analysis
- Sinking Fund Analysis
- Amortization Schedules
- Interest Conversions

HARDWARE REQUIREMENTS: 48K OSI, 8" floppy or hard disk, serial terminal system, OS-65U v. 1.2 or later.

PRICE: $300.00 (User Manual, $25.00, credited toward Planner purchase). Michigan residents add 4% sales tax.

DEALERS: Your Inquiries Most Welcome

GANDER SOFTWARE

3223 Bross Road
"The Ponds"
Hastings, MI 49058
(616) 945-2821

"It Flies"

FROM THE FOLKS WHO BROUGHT YOU:
All This
THERE IS MORE COMING SOON:
Program Generator for TDS
Proposal Planner
Time and Billing A/R

```
1010 PRINT"Choice ? ";:GOSUB310:PRINTC$;:IFY$="-"THEN190
1020 IFY=0THEN880
1030 IFY>5THENR=-1:Y=Y-5:ONYGOTO1120,1140,1090,1180
1040 :
1050 PRINT"Change ";:PRINT"* "N$(Y)" to ";:INPUTY$
1060 IFY$=""ORY$=H$THEN920
1070 D$(Q,Y)=Y$:GOTO920
1080 :
1090 PRINT"Erase ?  ";:GOSUB310:PRINTC$;
1100 IFA=121THEND$(Q,1)="ZZ"+D$(Q,1):GOTO1120
1110 Q=Q-1
1120 NEXTQ:GOTO890
1130 :
1140 Q=Q-1:IFQ=0THENQ=Z
1150 IFLEFT$(D$(Q,1),2)="ZZ"THEN1140
1160 GOTO1000
1170 :
1180 INPUT"* Record # = ";Q:IFQ<1ORQ>ZTHENPRINT!(28):GOTO1180
1190 GOTO920
1200 :
1210 REM Write a Record to Screen
1220 X=53568:Y$="RECORD "+STR$(Q)+" of"+STR$(Z):GOSUB1250:X=X+S
1230 FORC=1TOP:Y$=D$(Q,C):GOSUB1240:NEXTC:RETURN
1240 IFY$=""THENY$=" "
1250 M=LEN(Y$):FORR=1TOM:POKEX+R,ASC(MID$(Y$,R,1)):NEXTR:X=X+S:RETURN
1260 :
1270 REM SORT FILE
1280 PRINT"SORT on which FIELD # ? ";:GOSUB310:PRINTY:B=Y
1290 PRINT:IFY=0ORY>PTHEN1460
1300 PRINT"Is the File partially Sorted on Field"B" ? ";
1310 GOSUB310:IFA=104ORA=45THEN1460
1320 PRINT!(28):PRINTTAB(20)"* SORTING *":IFA=110ANDZ>4THEN1390
1330 :
1340 FORQ=2TOZ:Y=Q:FORC=1TOP:X$(C)=D$(Y,C):NEXTC
1350 IFD$(Y-1,B)<=X$(B)THEN1370
1360 FORC=1TOP:D$(Y,C)=D$(Y-1,C):NEXTC:Y=Y-1:IFY>1THEN1350
1370 FORC=1TOP:D$(Y,C)=X$(C):NEXTC,Q:GOTO1460
1380 :
1390 I=(2^INT(LOG(Z)/LOG(2)))-1
1400 I=INT(I/2):IFI<1THEN1460
1410 FORQ=1TOI:R=Q+I:FORC=RTOZSTEPI:Y=C
1420 FORK=1TOP:X$(K)=D$(Y,K):NEXTK
1430 IFD$(Y-I,B)<=X$(B)THEN1450
1440 FORK=1TOP:D$(Y,K)=D$(Y-I,K):NEXTK:Y=Y-I:IFY>ITHEN1430
1450 FORK=1TOP:D$(Y,K)=X$(K):NEXTK:NEXTC,Q:GOTO1400
1460 GOTO190
1470 :
1480 PRINT FILE
1490 INPUT"*  # of copies of each Record ";L:PRINT:IFL<1THEN1680
1500 :
1510 SS=0:PRINT:PRINT"Two columns ? ";:GOSUB310:PRINTY$
1520 IFA=110ORA=45THEN1570
1530 IFA=104THENPRINT:GOTO1490
1540 PRINT:PRINT"Records must be PACKED. Hit <RETURN> if not."
1550 SS=-1:E=ST+1:TB=40
1560 :
1570 PRINT:PRINT"Device # ? ";:GOSUB310:IFY=0THEN1680
1580 PRINTY:V=Y:PRINT:PRINT"Ready ? ";:GOSUB310:PRINT:PRINT
1590 FORQ=1TOZ:FORX=1TOL:IFLEFT$(D$(Q,1),2)="ZZ"THEN1670
1600 :
1610 IFSS=0THEN1660
1620 IFQ=ETHENE=Q+ST*2:Q=Q+ST:PRINT#V,F$
1630 IFQ>ZTHEN1680
1640 GOSUB1920:GOTO1670
1650 :
1660 GOSUB1880
1670 NEXTX,Q
1680 V=2:GOTO190
1690 :
1700 REM APPEND RECORDS
1710 R=Z+1:IFR>NTHENPRINT"* No more space left *":GOTO200
1720 :
1730 FORQ=RTON
1740 PRINT!(28):PRINT"* To return to main menu type:-  STOP *":PRINT
1750 PRINT:PRINT"Record "Q"of"N:PRINT:PRINT
1760 :
1770 FORC=1TOP:PRINT:PRINT"* "N$(C)" " ;:INPUTD$(Q,C)
1780 IFD$(Q,C)=H$THENPRINT:PRINT:PRINT:GOTO1750
1790 IFD$(Q,C)=S$THENZ=Q-1:FORY=1TOP:D$(Q,Y)="":NEXTY:Q=N:C=P
1800 NEXTC,Q:GOTO190
1810 :
1820 REM ERASED RECORDS
1830 R$=" ":PRINT"These Records are 'erased' but recoverable:-":PRINT
1840 FORQ=1TOZ:IFLEFT$(D$(Q,1),2)="ZZ"THENR$=R$+STR$(Q)
1850 NEXT:PRINTR$:PRINT:PRINT"Ready ? ";:GOSUB310:GOTO190
1860 :
1870 REM Print a Record
1880 PRINT#V,D$(Q,1)TAB(32)Q:FORC=2TOP:PRINT#V,D$(Q,C):NEXTC
1890 PRINT#V:RETURN
1900 :
1910 REM Print a Record in 2 columns
1920 PRINT#V,D$(Q,1)TAB(32)Q TAB(TB)D$(Q+ST,1)TAB(TB+32)Q+ST
1930 FORC=2TOP:PRINT#V,D$(Q,C) TAB(TB)D$(Q+ST,C):NEXTC:PRINT#V
1940 RETURN
1950 :
1960 REM Restart
1970 PRINT:FORC=1TO5:PRINTTAB(10)"<<< To RESTART type:-  GOTO 190 >>>"
1980 NEXT:POKE2073,173:END
1990 :
2000 REM TRAP 1 goes here
2010 PRINT!(28):PRINT"* You made an INPUT (or DISK) error- try again!
2020 PRINT:GOTO200
```

SAMPLE OF "TRAP"

```
10 CLEAR : PRINT !(28): REM COPYRIGHT by L. Z. JANKOWSKI APRIL '84
20 TRAP 2010: POKE 2888,0: POKE 8722,0: POKE 2972,13: POKE 2976,13: POKE 2073,96
30 REM
90 X=18047: FOR Y=X TO X+26: READ Q: POKE Y,Q: NEXT
100 DATA 72,138,72,152,72,160,0,169,32,153,0,210,153,0,209,153
110 DATA 0,208,200,208,244,104,168,104,170,104,96: TB=40: V=2
120 REM
2010 PRINT !(28): PRINT "* You made an INPUT (or DISK) error-
Ok                                                    try again
```

screen-clear.

When entering data in response to the INPUT command, it is desirable that BASIC accept <RETURN> as a null and then continue to the next BASIC line. For ROM BASIC the fix is at $A944,5. Change $47 and $A6 to $54 and $A9. The changes merely bypass the null input check. For DISK BASIC the first two POKEs in line 20 do that. The next two allow ',' and ':' on INPUT. The final POKE disables CTRL-C. Line 20 illustrates the flexibility acquired by using DISK BASIC. With ROM BASIC changes can only be made by burning new EPROMs. (There is actually another solution. Save ROM BASIC to disk, call it back into RAM and now, .BD11G gives a COLD START; answer memory size with a number. Works well!)

The TRAP command is unique to 65D 3.3 and is extremely useful. Works rather like ON ERROR GOTO. All INPUT errors, DISK errors and even program syntax errors will be routed to line 2010.

Lines 90-110 contain a machine language partial screen clear. The screen is cleared from $D000 to $D2FF. ROM BASIC users can place this routine either at $0222 or POKE it into high memory and then protect it with POKE 133,LO byte : POKE 134,HI byte. The variable TB is the second column tab when printing records, and V stores the device number.

★

EPROM PROGRAMMER

By: David Tasker
111 Bass Highway
Tasmania, Australia 7303

PARTS LIST:

Sockets
2 x 16 pin sockets. *
2 x 40 pin sockets.
1 x 14 pin socket.
1 x 28(24) pin socket. *
* These may be Zero insertion force sockets, (Z.I.F.). It is recommended that the 28(24) pin socket be a ZIF type as this is used for the EPROM device. The 28(24) pin socket

may be either a 28 pin or a 24 pin socket. 28 pin is preferred as this will allow 2764 EPROMS to be programmed.

Integrated Circuits
2 x 6821 PIA devices.
1 x 7404 TTL device.
1 x LM317 variable voltage regulator. **

Transistors and diodes.
4 x 2N3904 (BC 107) Q1, Q3, Q4 and Q6
2 x 2N3906 (BC177) Q2 and Q5.
3 x LEDs in different colors for D6, D7, and D8. Suggested colors are D6 Red, D7 Green, D8 Yellow.
4 x IN4001 or similar for D1 to D4. **
1 x IN914 for D5.

Resistors and capacitors.

R1	4.7k ohm.	R2	220 ohm. **
R3	220 ohm.	R4	10k ohm.
R5	10k ohm.	R6	10k ohm.
R7	10k ohm.	R8	10k ohm.
R9	10k ohm.	R10	27k ohm.
R11	10k ohm.	R12	4.7k ohm.
R13	470 ohm.	R14	47 ohm.
R15	10k ohm.	R16	100 ohm.
R17	470 ohm.	R18	2.7k ohm.

VR1 5k ohm Tab set pot'. **

C1 330uf. 47volt. **
C2 10 uf Tag.
C3 10 uf Tag.
C4 1 uf Tag. 47volt. **
C5 1 uf Tag. 47volt. **
C6 1 uf Tag. 47volt. **
C7 0.1uf.
C8 0.1uf.

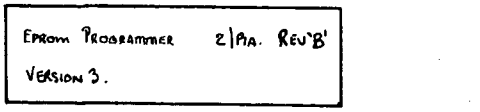
Points marked as OUT 1 to 4 may have multiple 0.1" in-line header pins inserted here to facilitate access of the PIA output lines.

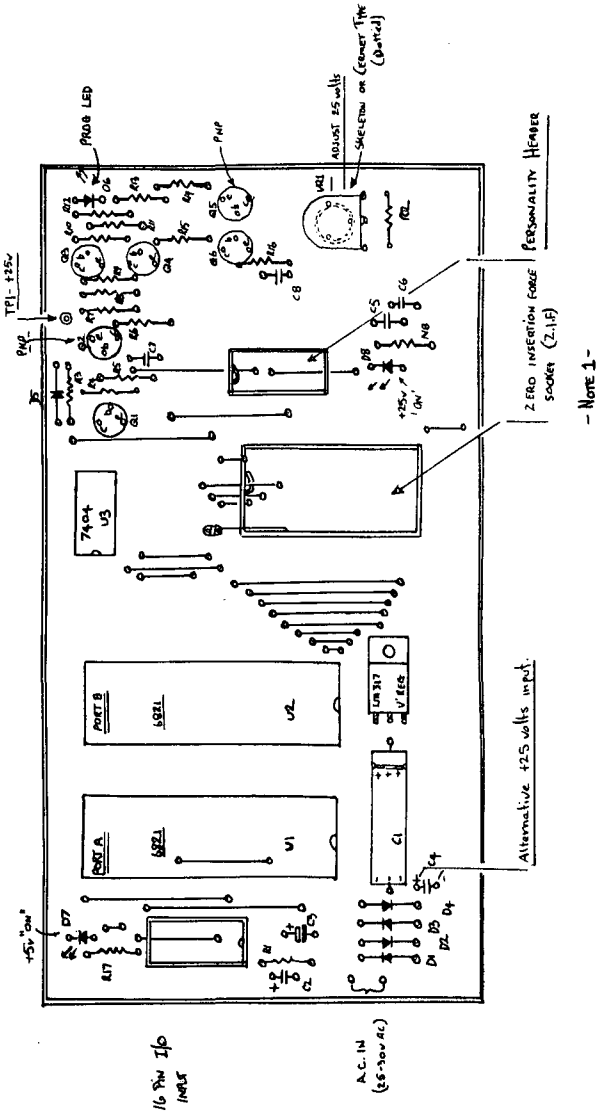
Items marked "**" may not be required if an external +25 volt supply is available and which can be connected at TP1 or at the point marked "0" (the center pin) of the LM317 which would not be fitted.

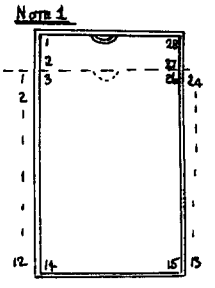
This programmer for I/O Bus is designed to plug into the 16 pin I/O Bus that can be found on many OSI Challenger Computer Systems. If your computer does not currently support the I/O Bus, then you should obtain the correct expansion adaptor.

There are a number of expansion boards available, but nearly all use variations of the standard 40 pin expansion socket system that OSI favors on its C1 and Superboard.

This programmer can be used on any computer that has the software driver routine installed. A description of the software requirements are given in the following notes.







24 pin devices i.e. 2708, 2716, 2732 MUST BE PLACED IN LOWEST 24 PINS.

OVERVIEW OF HARDWARE AND SOFTWARE

The EPROM programmer requires two latched ports at $C704-C707 and $C708-C70B. These ports are provided by the two PIA chips on the programmer board. The address selected is in keeping with OSI Input/Output (I/O) allocation. You may choose to have the programmer somewhere else within your memory, however, this would mean a change of software. A list of memory locations is included that would require a change to implement this.

The programmer is designed to program only the single 5volt supply EPROMS, e.g. 2708, 2716, 2532 (TMS), 2732 and 2764. It is versatile enough to be able to read most ROMs such as OSI BASIC ROMs CHARGENerator and Monitor ROMs.

HOW IT WORKS

In the 16 pin socket called "Personality Header" a series of links are put in which sets up the programmer hardware for the particular device that you wish to read or program.

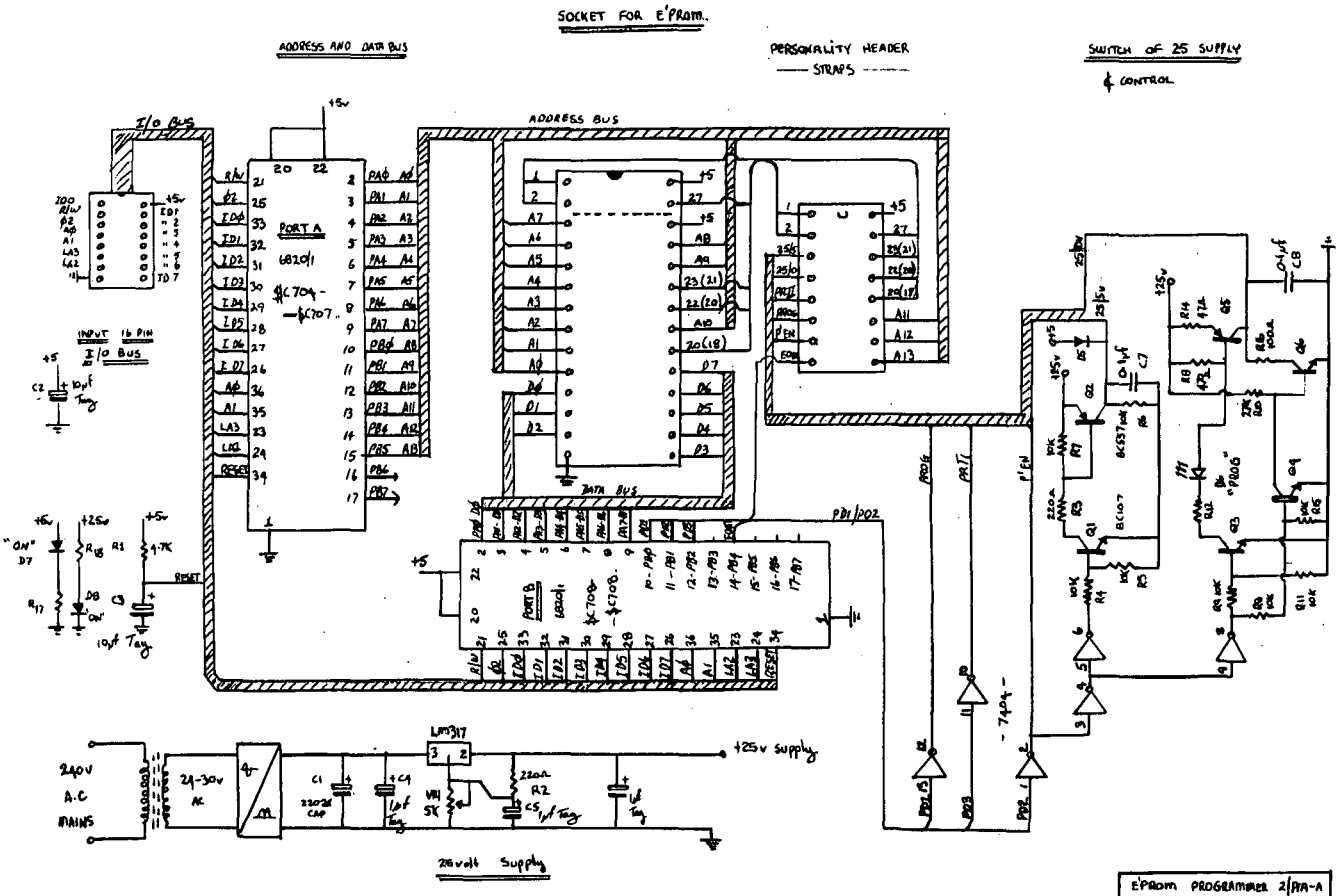
On Port A PA0 to PB7 (16 lines) are set as outputs and we present a binary setting which represents the address of the first location in the device. Data to the device is provided by PA0-PA7 of PORT B. The A side of port B can be inputs or outputs depending if we are reading or writing. Port B -PB0 and PB1 are the two control leads that set our read/write, chip select and also programming pulses.

As each data is presented and the addresses are incremented as required, the control leads will, as required, select the device and control the 25 volt programming pulse. The timing is all controlled by the software.

As the address is incremented Port bit PB4 will eventually go high as the EOM (end of memory) signal will be provided by the particular address line that you have it connected to. The software as set up will continue to program till it reaches EOM in software, then halts and relies on EOM from PB4 to exit the software read/write routine.

Switching and control of the device, incrementing of address and data changing are all transparent to the user and is looked after by the software. The programer is MENU driven.

MENU driven simply means that the screen display will show you a choice of functions each time that the programmer has finished a task.



SCHEMATICS CONTINUED

Isotron announces the Ohio Scientific WORKSYSTEM™

A MULTI-USER COMPUTER SYSTEM AT A WORKSTATION PRICE $6,995.*

Now you can fully equip your business or professional office with a complete multi-user computer system for $6,995.

Step up in performance but not in price. WORKSYSTEM breaks you out of the single-user bind. It's versatile, big league installation, preconfigured for three users who can perform different tasks all at the same time. The ideal system to grow with your growing organization.

Here's what you get

Your WORKSYSTEM is powered by an Ohio Scientific Microcomputer with 20 Megabyte Hard Disk storage—more than 40 times the capacity of the largest, densest floppy.

Expandable to four users, the system includes . . .

- ***Display Terminal,** * tilt-and-swivel, with easy-to-read 14" non glare screen and a full range of easy to use functions.*
- ***Letter-quality printer** turns out clean, crisp business correspondence.*
- ***Word processing package,** user-friendly, greatly enhances office productivity through its speed, versatility and editing features.*
- ***Data Base Management System** lets users file and access information quickly and fully.*

Ohio Scientific: leader and pioneer

WORKSYSTEM is the latest example of Ohio Scientific's widely recognized mastery in designing exceptional computer systems which are proving their worth in thousands of growing business and professional offices.

Dealers who solve your problems

Isotron, Inc., the American subsidiary of the Swedish industrial giant Investment AB Beijer, combines the technology and integrity of Ohio Scientific hardware and software with a network of dealers who analyze and meet your needs for today and tomorrow.

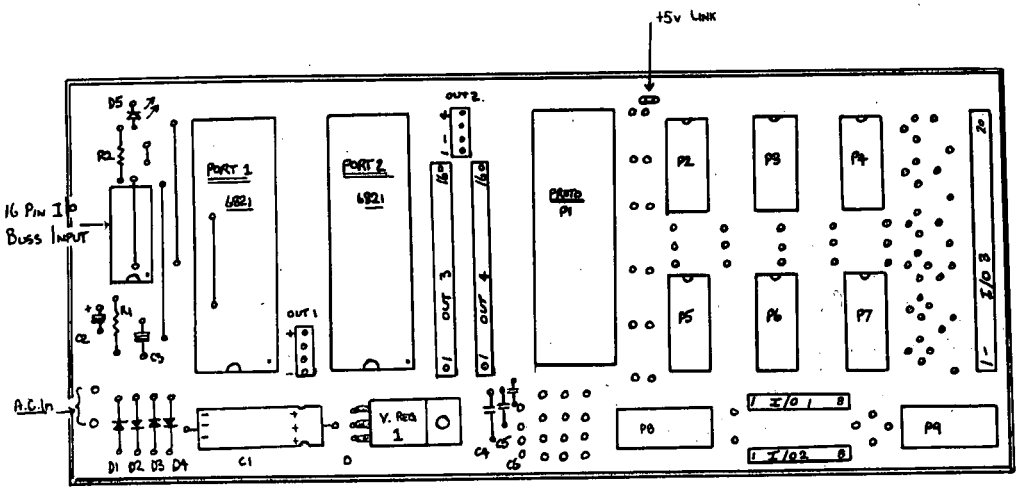
ISOTRON

140 SHERMAN ST.
FAIRFIELD, CT 06430
(203) 255-7443
TLX-756436

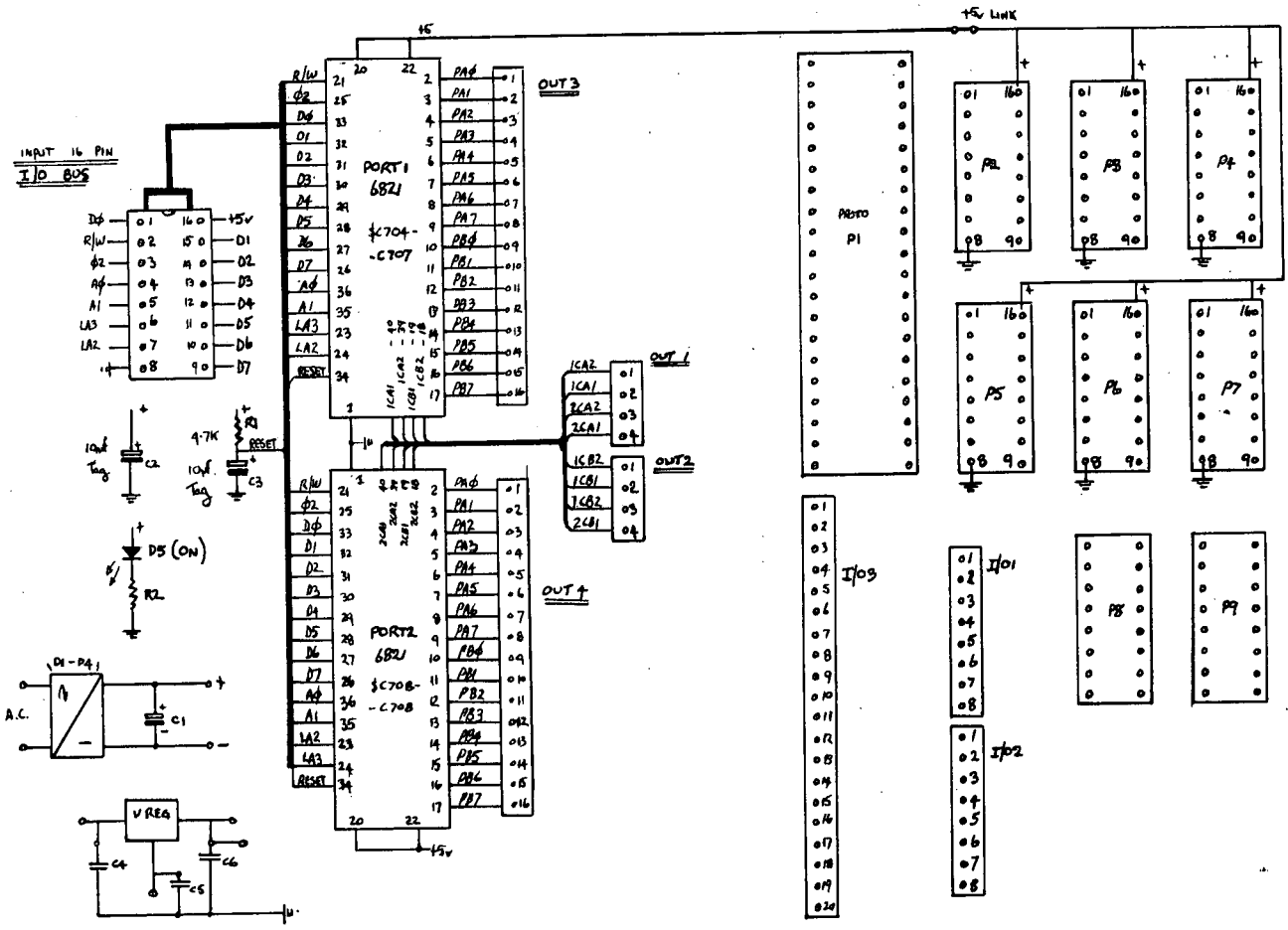
Call your authorized Isotron dealer

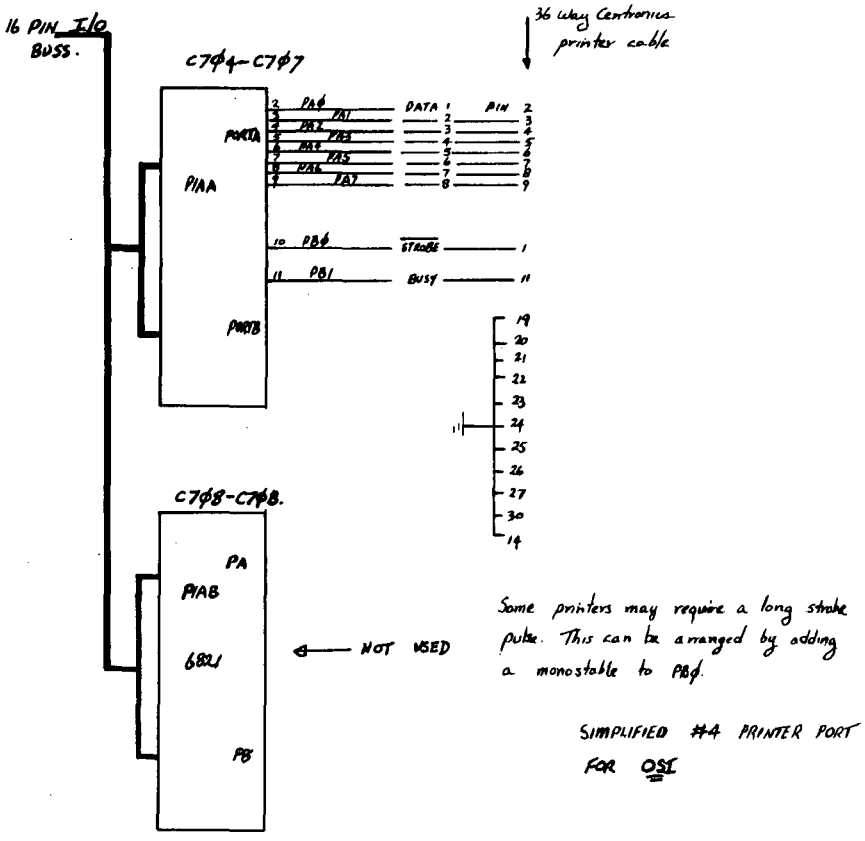
*Additional Terminals optional.
*Multi-Processing System also available.

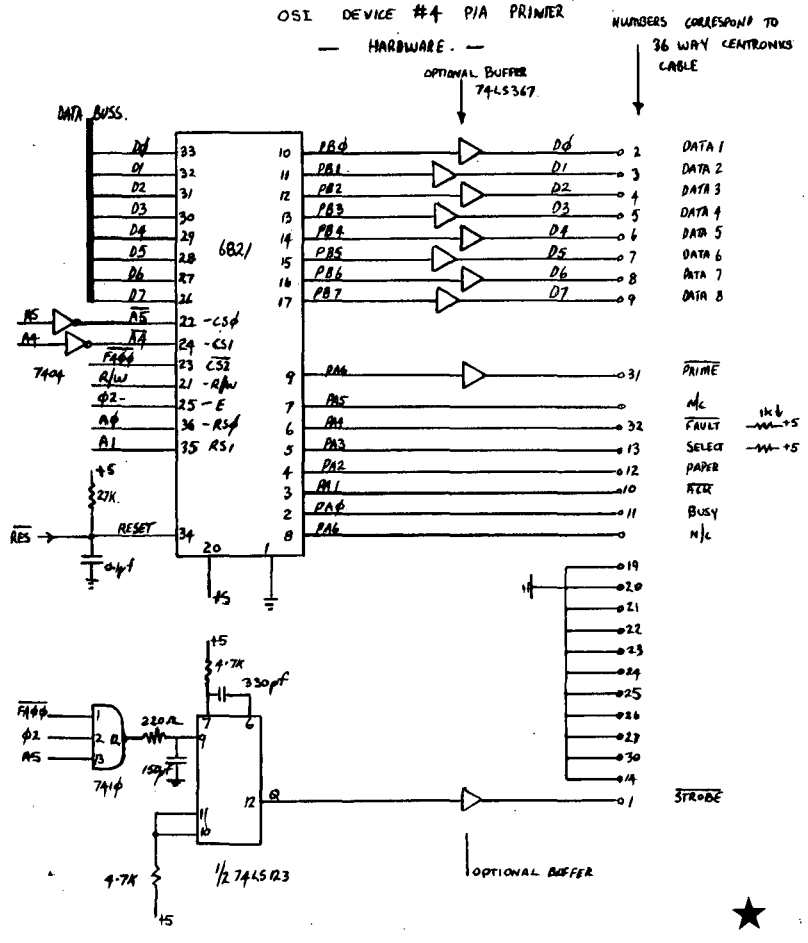
16 Pin I/O Buss
2-6821 Ports + Prototyping



2x6821 Port + Protocard for 16 Pin I/O. Bus.







DISK DRIVE RECONDITIONING

WINCHESTER DRIVES

NEW OSI CD23 (SA4008)

FLAT RATE CLEAN ROOM SERVICE.
(parts & labor included)

Shugart	SA4008	23meg	$550.00
Shugart	SA1004	10meg	$450.00
Seagate	ST412	10meg	$350.00
Seagate	ST506	5meg	$290.00

FLOPPY DRIVE FLAT RATES
Parts & Labor Included

8" Single Sided Siemens	$150.00
D&E Series	
8" Double Sided Siemens	$170.00
P Series	
8" Double Sided Remex	$225.00
8" Single Sided Shugart	$190.00
8" Double Sided Shugart	$250.00

ONE WEEK TURN AROUND TYPICAL

90 Day warranty on Floppy & Large Winch.
1 Yr. Warranty on 5" & 8" Winchesters.

Write or call for detailed brochure

Phone: (417) 485-2501

FESSENDEN COMPUTERS
116 N. 3RD STREET
OZARK, MO 65721

NEW SOFTWARE

BETA/65

Programming System
Editor/Interpreter

plus 55 functions

featuring

Mixed-Precision Arithmetic

CALL with Two Arg. Types

LINK by Name or Address

Concurrent User Entry

Programming Manual. . .$30

Integer System v.2.3. . . . .$90

MicroGram Systems

SR 3 Box 62 La Honda CA
94020 Tel. (415) 747-0811

BUILD A CHEAP RS-232 COMPATIBLE MODEM

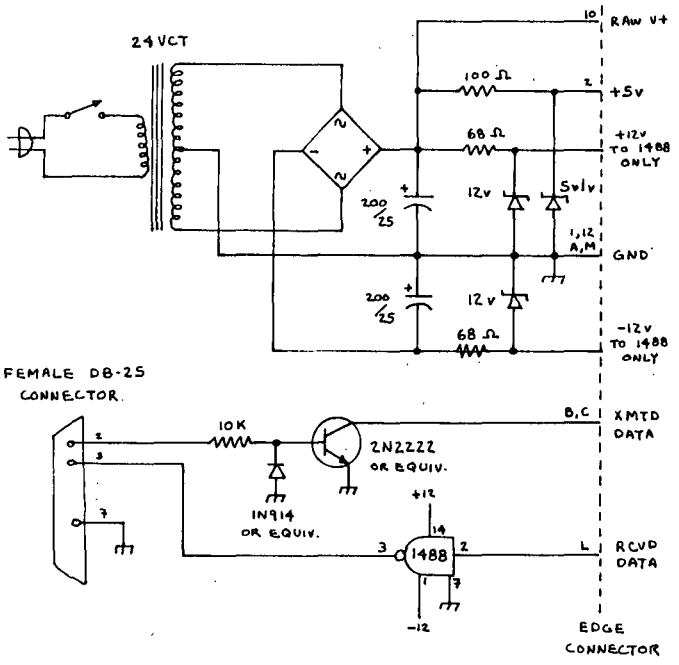
By: Jim McConkey
7304 Centennial Road
Rockville, MD 20855

Many computer users are getting into computerized telecommunications today for both work and pleasure. Some just enjoy communicating with others, say on the OSI SIG on Compuserve. Banks are now offering bank-by-personal computer services to their customers. MCI allows you to send mailgrams from your home computer. Some people (myself included) are able to work at home, at least part time, on their home computers and have to communicate with their offices periodically. All of these applications require a modem.

The modem, which serves as a link between your computer and the phone lines, is the first item required to get into telecomputing. Ever since the introduction of the inexpensive (about $50) modem for the Commodore 64, I have wondered whether it could be adapted for use with my C1P. This modem only works at up to 300 baud and lacks fancy features like automatic dialing, but it is just fine for my uses.

FIGURE 1

RS-232/HESMODEM INTERFACE AND POWER SUPPLY



NOTE: ALL PARTS EXCEPT 1488 ARE AVAILABLE AT RADIO SHACK.

MEM *PLUS*

BARE ..... $ 75	32K $300	52K $415
16K $200	40K $350	56K $440
24K $250	48K $390	64K $490

MEM+ Options Include:

- OSI compatible floppy disk controller — add $85
- RTC — Real Time Clock — day, date and time with lithium battery backup — add $85
- Centronics parallel printer interface with software for OS65D and OS65U — add $65
- High reliability sockets for memory chips — add 15%
- RTC only (OSI CA-20 replacement) — $195

All boards feature solder mask, silkscreen, **gold-plated** edge connectors and a one year warranty.

07514 51893

Generic Computer Products

High Resolution Color Graphics

Our new Color Plus board provides 256 x 192 high-resolution graphics with 15 colors. Two 8-bit resolution joystick interfaces are included. Software extensions to OS65-D BASIC provide a superset of APPLE II graphics instructions.

Color Plus connects to the standard 48-pin bus or the 16-pin bus.

Pricing:

CP-8 for C8 or C3 computers:	$195
CP-4 for C4 computers (5V only):	$245
CP-bare Bare board with software:	$ 75

VISA, MasterCard, personal checks and C.O.D.s all accepted. Add $5 per board for shipping and handling.

To order, or for more information, contact:
Fial Computer
5221 S.W. Corbett
Portland, Oregon 97201
(503) 227-7083

!!!!! MORE GOOD NEWS !!!!!
From
D.B.I., INC.

The reliable Denver Board has found a new home.
It has become the heart of a NEW,

Multi-User/Multi-Processing

SUPER-MICRO COMPUTER SYSTEM

Built exclusively by D.B.I., INC.

* Up to 16 Users
* SCSI interface (connect to most SCSI devices on market)
* ERROR CORRECTING controller (no more intermittent read errors)
* LOW POWER consumption
* RUNS existing OS-65U OPERATING SYSTEMS and SOFTWARE
* NEW DB-DOS runs existing OS-65U SOFTWARE, but has changes and extensions, to take advantage of the MICRO's features (faster)
* SUPER QUIET
* PRINTER INTERFACES (both serial and parallel)
* REAL TIME CLOCK (day, date, 24 hour, day of the week)
* NEW SIZE (approximately the same size as an average terminal)
* TRANSPORTABLE, about 40 LBS. (single user 5 megabyte)
* SHIPPABLE (can be sent by U.P.S.)
* DISK SIZE available NOW! includes 5, 10, 20, 40 & 80 megabytes.
* Taking orders for 160 megabyte systems (35 mil/sec access)
* LOW COST PER BYTE BREAK-THROUGH, single user 5 meg-sug-$4995.

CALL YOUR DBI DEALER TODAY FOR FURTHER DETAILS!

DBi, inc.

p.o. box 7276
denver, colorado 80207
phone: (303) 428-0222

Curiosity overtook me and I finally got a Hesmodem (also for the Commodore 64) in December and figured out that it could be adapted. I decided to make it RS-232 compatible, so that it would work with any computer, and build in a power supply (these modems are meant to get their power from the Commodore). The parts cost to adapt a Hesmodem is about $10, less if you have a well stocked junk box. I have not tried it, but I think this adaptation will also work with a Vicmodem (Commodore's version). This adaptation requires no modification of the Hesmodem itself, preserving it for possible use with a Commodore or for resale if you decide to move up to 1200 baud at a later date.

The Hesmodem (and presumably the Vicmodem) is based on the Texas Instruments TMS99532 single chip modem, which requires very few external parts. The Hesmodem also contains on-board supply voltage regulation and circuitry to meet FCC regulations for direct connection to the phone lines. The only connections to the Hesmodem are power and TTL-compatible data send and receive lines. Figure 1 shows the schematic of the interface. The transformer, bridge rectifier, zeners and associated resistors and capacitors provide the various required voltages. The transistor is used to convert the bipolar RS-232 line to TTL levels. The collector line is pulled up on the Hesmodem. This receiver will also work with the ClP's 0 to +5 volt output on the modem and RS-232 ports. The 1488 is used to turn the TTL level data output line from the Hesmodem into proper RS-232 levels. Construction was done on a modified 44 pin edge connector board, which was cut down as shown in Figure 2 to provide the 24 pin edge connector required by the modem.

The other ingredient required for telecommunications, once you have a modem, is the terminal software to drive it. OSI at one time supplied a simple "dumb" terminal routine on cassette with each ClP and it is probably still available. This cassette version would also work under OS-65D. There are at least two modem programs available which run under HEXDOS. One is my adaptation/expansion of OSI's cassette "dumb" terminal routine for HEXDOS and the other allows spooling received data to a HEXDOS disk file. Both are in the public domain and both are available from the HEXDOS User's Library (c/o Vern Heidner, 1440 Co. Rd. 110 N., Mound, MN 55364). I will also supply my adaptation of OSI's program to interested HEXDOS users. Send me $5 and a disk. I'll copy the program onto your disk, along with the HEXASM source code for the machine language subroutine, and return the disk to you. To protect Steve's copyright, please send the disk WITHOUT track zero! If anyone has written a smart terminal package that allows both up and down-loading and local command execution for OS-65D of HEXDOS, please write in and tell us about it.

There you have it! An inexpensive way to go on-line. If you don't know who to call, there are several books listing public bulletin boards and explaining telecomputing, and Computer Shopper always has a long listing of free bulletin boards to get you started.

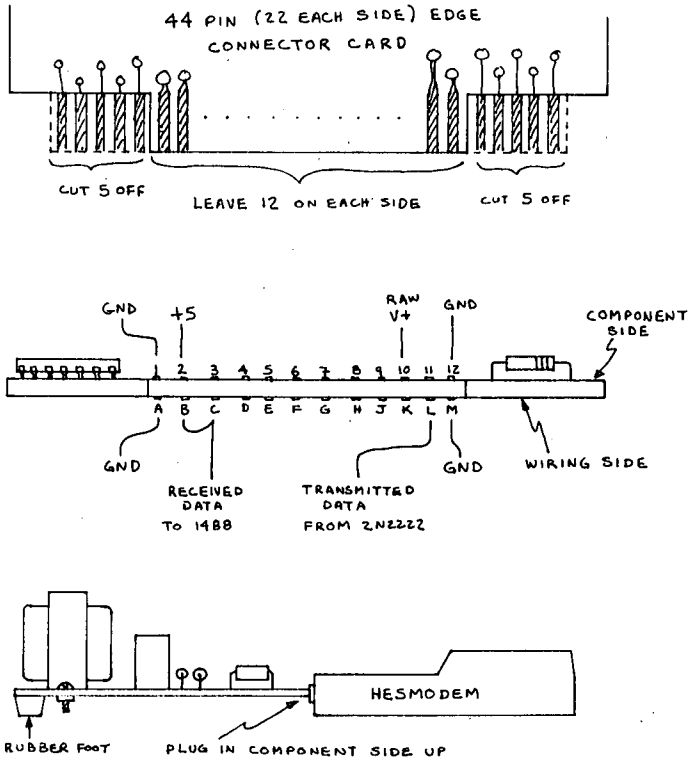


FIGURE 2

BOARD DETAILS

★ ★

BEXEC* PROGRAM

REQUIRED FOR "HOOKS INTO BASIC" by RICK TRETHEWEY

See December 1983 and January 1984 issues of PEEK(65)

```
10 X=PEEK(10950):POKE8993,X:POKE8994,X:POKE14170,16
20 POKE14172,8:POKE2888,0:IFPEEK(57088)=223THENPOKE9794,37
30 MAXMEM=PEEK(8960):DEST=MAXMEM-9:IFPEEK(532)=156THEN1040
40 POKE133,DEST-1:CLEAR:DEST=PEEK(133)+1:GOSUB580
50 HEX$="0123456789ABCDEF":A1=INT(DEST/16):A2=DEST-A1*16
60 AD$=MID$(HEX$,A1+1,1)+MID$(HEX$,A2+1,1)+"00"
70 TT$=RIGHT$(STR$(TT+100),2)
80 DISK!"CA "+AD$+"="+TT$+",1"
90 IFDT<>8THENDEST=DEST+8:TT=TT+1:GOTO50
100 :
110 REM INSTALL PATCH INTO LET FOR NEW KEYWORDS
120 DEST=PEEK(133)+1:FORK=2470TO2476
130 READY:IFY=182THENPOKEK,DEST:NEXTK
```

Continued

D&N MICRO PRODUCTS, INC.

3702 N. Wells St.
Fort Wayne, Ind. 46808
(219) 484-6414

TERMS $3.00 shipping, Foreign orders add 15%, Indiana residents add 5% sales tax.

COMPUTER

MICRO-80 COMPUTER

Z-80A CPU with 4Mhz clock and CP/M 2.2 operating system. 64K low power static memory. Centronics parallel printer port. 3 serial ports. 4" cooling fan. Two 8" single or double sided floppy disk drives. IBM single density 3740 format for 243K or storage, double density format for 604K of storage. Double sided drives allow 1.2 meg on each drive. Satin finish extruded aluminum with vinyl woodgrain decorative finish. 8 slot backplane, 48 pin buss compatible with OSI boards.

MODEL 80-1200 **$2995**
2 8" Single sided drives

MODEL 80-2400 **$3495**
2 8" Double sided drives

MICRO-65 COMPUTER

6502 CPU with 2Mhz clock and DOS-65 operating system. 48K of low power static memory. 2 serial ports and 1 Centronics parallel port. 2 8" single or double sided drives. Satin finish extruded aluminum with vinyl woodgrain finish. 8 slot backplane, 48 pin buss compatible with OSI. Will run OSI 65D and 65U software.

MODEL 65-1 **$2995**
2 8" Single sided drives

MODEL 65-2 **$3495**
2 8" Double sided drives

BP-580 8 Slot Backplane . . . . . **$ 47**
OSI 48 pin Buss compatible

MEM-CM9 MEMORY/ FLOPPY CONTROLLER

24K memory/floppy controller card uses 2114 memory chips, 1 8K and 1 16K partition. Supports OSI type disk interface

24MEM-CM9 . . . . . **$325**
16MEM-CM9 . . . . . **$260**
8MEM-CM9 . . . . . **$180**
BARE MEM-CM9 . . . . . **$ 50**
Controller on assembled unit add . . . . . **$ 90**

BIO-1600 Bare IO card . . . . . **$ 50**
Supports 8K of memory, 2 16 bit parallel ports, 5 serial ports, with manual and Molex connectors.

PRINTERS

Okidata

ML82A, 120 cps, 10" . **$409**
ML83A, 120 cps, 15" . **$895**
ML84 Parallel, 200 caps, 15" . **$1150**

C. Ioth

8510AP Prowriter, parallel . . . **$419**
120 cps, correspondence quality
8510APD Prowriter, serial **$585**
F10-40PU Starwriter, parallel **$1319**
Letter quality daisy wheel
F10-40RU Starwriter, serial . . **$1319**
F10-55PU Printmaster **$1610**
parallel, Letter quality daisy wheel
F10-55RU Printmaster, serial **$1610**

DISK DRIVES AND CABLES

8" Shugart SA801 . . . . . **$385**
single sided
8" Shugart SA851 **$585**
double sided
FLC-6 6 ft cable from D&N **$69**
or OSI disk controller to 8" drive
5¼" MPI B51 disk drive with . . **$450**
cable, power supply and cabinet. Specify computer type.
FLC-5¼ cable for connection . **$75**
to 5¼ drive and D&N or OSI controller, with data separator and disk switch. Specify computer type

HARDWARE

OSI COMPATIBLE

IO-CA10X Serial Printer Port . . **$125**
Specify Device #3 or #8
IO-CA9 Parallel Printer Port . . **$150**

CMOS-MEM

64K CMOS static memory board, uses 6116 chips, 3 16K, 1 8K and 2 4K blocks, Partitionable for multi-user, OSI type disk controller, 2 IO mapped serial ports for use with D&N-80 CPU. Ideal way to upgrade from cassette to disk.

64K CMOS-MEM **$490**
48K CMOS-MEM **$390**
24K CMOS-MEM **$250**
16K CMOS-MEM **$200**

Controller add . **$ 90**
2 IO mapped serial ports add. **$125**
on assembled memory board
Z80-IO 2 IO mapped serial **$160**
ports for use with D&N-80 CPU card
FL470 Disk Controller **$155**
Specify 5¼ or 8" drive

STANDARD CP/M FOR OSI

D&N-80 CPU CARD

The D&N-80 CPU allows the owner of an OSI static memory computer to convert to Industrial Standard IBM 3740 single density disk format and CP/M operating system. Double density disk operation is also supported for 608K of storage on an 8" diskette. When used with a 5¼" disk system 200K of storage is provided. Includes parallel printer and real time clock. Also available for polled keyboard and video systems. Compatible with C2, C3, C4 and 200 series OSI computers.

D&N-80- P **$349**

CP / M 2.2 . . . . . **$150**

64K CMOS-MEM with D&N-80 CPU card **$450**

HARD DISK DRIVER **$140**
Allows D&N-80 CPU board to control OSI 40 or 80 meg hard disk unit. Will not destroy OSI files. Will also allow for a true 56K CP/M system. Specify 40 or 80 meg drive.

BUSS TRANSFER **$135**
Allows for D&N-80 and OSI CPU to be in the computer at the same time. Toggle switch provides for alternate CPU operation.

DISK TRANSFER **$100**
Utility program to transfer OSI CP/M format disk to IBM 3740 single density format. Will also transfer IBM to OSI format.

SYSTEM HARDWARE REQUIREMENTS

D&N-80 CPU, D&N FL470 or OSI 470 controller, 48K memory at 0000-BFFF, 4K memory at D000--DFFF, two disk drive cables.

FORMAT TRANSFER **$15**
You supply software on 8" diskette D&N will transfer OSI CP/M format to IBM 3740 CP/M format. Can also transfer IBM 3740 CP/M format to OSI CP/M format. Original diskette returned.

```
140 POKEK,Y:NEXTK
150 DATA 32,0,182,234,234,234,234
160 :
170 MAXMEM=PEEK(8960)
180 :
190 REM- ENABLE NAMED GOSUBS AND GOTOS
200 POKE 2215,134: POKE 2216, MAXMEM-1
210 :
220 REM- ENABLE IF...THEN....ELSE
230 POKE 532,156 : POKE 533, MAXMEM-1
240 :
250 REM- ENABLE HEX PRINT
260 DA=8643:IFPEEK(121)=58THENDA=8379
270 POKE DA,57 : POKE DA+1,MAXMEM
280 :
290 REM INSTALL PATCH TO EVAL TO ALLOW HEX EXPRESSIONS
300 GOSUB430
360 :
370 REM DISABLE "REDO FROM START"
380 POKE 2893,55 : POKE 2894,8
390 X=PEEK(133):POKE8960,X:GOTO1040
400 :
410 REM- USE "CALL" TO PATCH TO EVAL
420 :
430 DEST=PEEK(133)*256:K=0
440 READA:POKEDEST+K,A:K=K+1:IFA<>96THEN440
450 CALL DEST:RETURN
460 DATA 169,76        :REM-  LDA #$4C
470 DATA 141,195,13    :REM-  STA $0DC3
480 DATA 169,27        :REM-  LDA #$1B
490 DATA 141,196,13    :REM-  STA $0DC4
500 DATA 172,0,35      :REM-  LDY $2300
510 DATA 136           :REM-  DEY
520 DATA 140,197,13    :REM-  STY $0DC5
530 DATA 169,234       :REM-  LDA #$EA
540 DATA 141,198,13    :REM-  STA $0DC6
550 DATA 96            :REM-  RTS
560 :
570 REM- FIND FILE CONTAINING HOOKS OBJECT CODE
580 D=11897:F$="BASIC+":DT=PEEK(11716):TRAP900
590 DEF FNA(X)=10*INT(X/16)+X-16*INT(X/16)
600 DT=FNA(DT):DT$=RIGHT$(STR$(DT+100),2):S=1
610 DISK!"CA 2E79="+DT$+","+RIGHT$(STR$(S),1)
620 FORI=DTOD+255STEP8:F1$=""
630 FORJ=0TO5:F1$=F1$+CHR$(PEEK(I+J)):NEXTJ
640 IFF1$=F$THENTT=FNA(PEEK(I+J)):RETURN
650 NEXTI:IFS<2THENS=S+1:GOTO610
680 :
890 REM- EXECUTIVE FOR FAILURE TO FIND "BASIC+"
900 POKE741,76:POKE750,78:POKE2073,173
910 PRINT" OS-65D V3.3":PRINT:CLEAR:X=FRE(X)
920 IFX<0THENX=X+65536
930 PRINTX;"BYTES FREE":END
940 :
1040 clr$=CHR$(27)+CHR$(28)
1050 PRINTclr$;:R*:Q*5:PRINTTAB(18);:D*:PRINT:TRAP0
1060 PRINT"(1) Create a New File"
1070 PRINT"(2) Delete a File"
1080 PRINT"(3) Rename a File"
1090 PRINT"(4) Invoke the Assembler/Editor"
1091 PRINT"(5) Invoke the Extended Monitor"
1092 PRINT"(6) Exit"
1100 PRINT:INPUT"    Your Choice";y$:k=VAL(y$)
1110 IFk<1ORk>5THENTRAP0:END
1120 ONkGOTO1130,1170,1200,1250,1251
1130 INPUT"File name ";f$
1140 GOSUB1260
1150 INPUT"How many tracks ";nt
1160 make f$,nt:GOTO1050
1170 INPUT"File Name ";f$
1180 GOSUB1260
1190 kill f$:GOTO1050
1200 GOSUB1260
1210 INPUT"Old File Name ";of$
1220 INPUT"New File Name ";nf$
1230 rename of$ TO nf$:GOTO1040
1240 FORk=1TO3000:NEXTk:RUN
1250 ASM
1251 EM
1260 INPUT"Drive (A/B) ";dr$:s*dr$:RETURN
```


Introducing

SCRIBE

WORD PROCESSOR

OS-65U 1.42 < Floppy/Hard Disk, Level 1 or Level 3

and DENVER BOARDS

- INTERFACED TO OS-DMS FILES
- AUTOMATIC WRAP AROUND
- COMPLETE EDITING CAPABILITIES
  - FULL CURSOR CONTROL
  - INSERT & DELETE TEXT
  - SEARCH/SEARCH & REPLACE
- USER FRIENDLY MANUAL
- AND MUCH MORE

IHS COMPUTER SERVICES
Route 1 Box 201B Port Republic, VA 24471
(703) 249-4833

$195.00

FOR SALE

Ohio Scientific
C3B

with 48K Memory

9 Track Tape
Interface

1 or 2
Winchester
80MB Drives

Rack Included

Reasonably Priced

Call:
Scott Field
901-458-4662

OSI repairs

C-2, C-3, & CD Series
200 Series

- board level service on:
- power supplies
- 8" floppy drives
- cpu, memories, etc.
- gold molex contacts
- custom printer cables

(1 week turnaround typical)

Sokol Electronics Inc.
474 N. Potomac St.
Hagerstown, Md. 21740
(301) 791-2562

```
9 REM RJ V1.2 by LZ JANKOWSKI
10 PRINT !(28): POKE 133,73: POKE 8960,73: INPUT "First TR# of File ";TR$
11 TR=VAL(TR$): DISK !"CA 4A00="+TR$+",1": N=PEEK(18948): IF N=1 THEN 15
12 IF N>9 THEN PRINT : PRINT "TOO MANY TRACKS!": LIST -15: END
13 FOR X=1 TO N-1: READ AD$(X): NEXT : FOR Z=1 TO X-1: TR=TR+1
14 TR$=RIGHT$(STR$(TR+100),2): DISK !"CA "+AD$(Z)+"="+TR$+",1": NEXT
15 DATA 5500,6000,6800,7600,8100,8C00,9700,A200
16 CLEAR : PRINT : GOTO 880
20 W$="": F=1: T$=CHR$(125): A$="PpFfMmDdTtCcSsKkWw": L=LEN(A$)
30 K=18949: R1=140: R2=80: DISK !"IO ,"+FG$: PRINT TAB( P): REM K=$4A05
40 IF T THEN GOSUB 460
45 GOTO 70
50 IF N<LEN(T$) THEN 100
60 T$=""
70 N=1: U=0: FOR Q=K TO K+R1: C$=CHR$(PEEK(Q)): T$=T$+C$: U=U+1
80 IF U>R2 AND C$=" " THEN K=K+U: GOTO 100
90 NEXT : T$=T$+CHR$(1): K=K+U
100 Q=FRE(X): IF C$="#" THEN 610
110 IF ASC(C$)=64 THEN GOSUB 480: GOTO 910
120 IF ASC(C$)=125 THEN GOSUB 480: GOSUB 400: W=O-P+J: GOSUB 540: GOTO 310
130 B=0: X=LEN(L$)+LEN(W$): IF LEN(W$)>W AND G AND LEN(L$)>W THEN GOSUB 280
140 IF X>W THEN GOSUB 170: L$="": GOTO 290
150 IF X>W-2 THEN B=1
160 GOTO 290
170 M=P: U=1
180 IF MID$(L$,U,1)=" " THEN U=U+1: GOTO 180
190 R=LEN(L$): IF R=0 THEN RETURN
200 IF MID$(L$,R,1)<>" " THEN Q=0: GOTO 230
210 R=R-1: IF R=0 THEN RETURN
220 GOTO 200
230 FOR X=U TO R: IF MID$(L$,X,1)=" " THEN Q=Q+1
240 NEXT : A=W-R: FOR Y=1 TO R: F$=MID$(L$,Y,1): IF Y>U AND F$=" " THEN 270
250 IF M=P AND G=0 OR F=1 THEN GOSUB 520
260 PRINT F$;: NEXT : GOTO 280
270 H=INT(A/Q): X=H+1: FOR Z=1 TO X: PRINT " ";: NEXT : A=A-H: Q=Q-1: NEXT
280 G=0: GOTO 390
290 L$=L$+W$: IF B THEN L$=L$+" ": GOTO 310
300 IF LEN(L$)<W THEN L$=L$+" "
310 W$="": IF N>=LEN(T$) THEN 50
320 C$=MID$(T$,N,1): N=N+1: IF C$=" " THEN 320
330 IF C$="" THEN C$=" ": GOTO 50
340 IF C$="#" OR ASC(C$)=125 OR ASC(C$)=64 OR ASC(C$)=1 THEN 50
350 W$=W$+C$: C$=MID$(T$,N,1): IF C$=" " THEN 50
355 IF C$="" THEN C$=" ": GOTO 50
360 N=N+1: IF C$="#" OR ASC(C$)=125 OR ASC(C$)=64 OR ASC(C$)=1 THEN 50
370 GOTO 350
380 G=0: GOSUB 480: L$="    "
390 PRINT : F=F+1: W=O-P+J
400 IF S=0 OR F=1 THEN 420
410 FOR Z=1 TO S: PRINT : F=F+1: NEXT
420 IF F<=PL THEN RETURN
430 IF LEFT$(H$,1)="N" THEN 450
440 DISK !"IO ,02": PRINT : INPUT "# Hold NEXT Page ";H$: PRINT
445 DISK !"IO ,"+FG$: GOTO 460
450 PRINT CHR$(12)
460 F=1: IF T THEN PRINT TAB( J+INT(O-3)/2)T: T=T+1: PRINT : F=F+2
470 RETURN
480 IF LEN(L$)+LEN(W$)>W THEN GOSUB 170: L$=""
490 GOSUB 520: PRINT L$;: L$=""
500 IF TI THEN PRINT W$;: TI=0: RETURN
510 PRINT W$: F=F+1: RETURN
520 IF I THEN 550
530 IF E=27 THEN E=0: RETURN
540 M=0: PRINT TAB( P): RETURN
550 IF D=0 THEN PRINT TAB( I): GOTO 590
560 IF V>I-P THEN V=0: X=I-V: W=W-P: GOTO 580
570 X=I-V-P: IF V=I-P THEN 590
580 FOR M=1 TO X: PRINT ".";: NEXT
590 M=0: I=0: D=0: RETURN
600 GOSUB 480: GOSUB 450: RETURN
610 C$=MID$(T$,N,1): N=N+1: FOR Z=1 TO L: E$=MID$(A$,Z,1): IF C$<>E$ THEN NEXT
620 ON INT((Z+1)/2)GOSUB 380,600,630,650,660,690,770,780,835,840: GOTO 310
630 GOSUB 480: GOSUB 400: GOSUB 875: P=J+Q: W=O-P+J: G=0: RETURN
650 D=-1
660 GOSUB 860: GOSUB 875: I=Q: IF I<V+P THEN GOSUB 390
670 W=O+J-I: G=0: RETURN
690 IF G THEN E=27
700 IF LEN(W$)>=W THEN GOSUB 390: GOSUB 540
710 GOSUB 860: GOSUB 875: W=W-V: IF W=0 THEN GOSUB 390: GOSUB 540
730 PRINT CHR$(Q);: IF Q=44 THEN PRINT " ";: W=W-1
740 IF Q<>27 THEN G=-1
750 IF Q=58 OR Q=32 OR Q=44 OR Q=35 OR Q=64 OR Q=93 OR Q=94 THEN W=W-1
760 E=Q: RETURN
770 GOSUB 480: S=VAL(MID$(T$,N,1)): N=N+1: RETURN
780 GOSUB 860: DISK !"IO ,02": PRINT : INPUT E$: C$=RIGHT$(T$,LEN(T$)-N+1)
790 PRINT : IF MID$(E$,LEN(E$),1)<>" " THEN E$=E$+" "
800 IF C$<>" " THEN T$=E$+C$: GOTO 820
810 T$=E$
820 W=W-V: G=-1: N=1: IF MID$(T$,1,1)="#" THEN E=27
830 DISK !"IO ,"+FG$: RETURN
835 GOSUB 480: GOSUB 400: GOSUB 875: O=Q: W=O-P+J: GOSUB 540: G=0: RETURN
840 GOSUB 480: GOSUB 540: FOR Z=1 TO O-P+J: PRINT C$;: NEXT : PRINT : F=F+1
850 GOSUB 390: GOSUB 540: RETURN
860 TI=-1: Z=LEN(L$)+LEN(W$): IF Z>W THEN Z=LEN(W$)
870 V=Z: GOSUB 480: RETURN
875 Q=10*VAL(MID$(T$,N,1))+VAL(MID$(T$,N+1,1)): N=N+2: RETURN
880 INPUT "Copies ";C: INPUT "Line Spacing ";S: PRINT : PRINT "PAGE:": PRINT
890 INPUT "Margin ";P: INPUT "Width  ";W: INPUT "Length ";PL: J=P: O=W
900 INPUT "Number ";T: INPUT "Hold   ";H$: PRINT : S=S-1: TT=T: SS=S: HH$=H$
905 PRINT : INPUT "DISK!'IO ,nn'. (08 is DV#4). nn= ";FG$: GOTO 20
910 PRINT : C=C-1: S=SS: T=TT: P=J: W=O: H$=HH$
920 IF C THEN PRINT CHR$(12): GOTO 20
930 PRINT CHR$(12): DISK !"IO ,02": END
```

★ ★

RIGHT-JUSTIFIED TEXT FOR DISK WP6502 V1.2

By: L. Z. Jankowski
Otaio Rd 1 Timaru
New Zealand

This is a BASIC program which recognizes all the embedded commands except #Bbnn. There is one extra command, #Wwnn. This command changes the line width to provide a variable right margin.

If too many different #Ttnn commands follow each other too closely in the text then only the 'last' #Ttnn command will be recognized. A number of such commands could be replaced by one set of printer control commands.

Lines 10 to 15 of the program read the required WP6502 file into RAM from where it is processed.

SIMPLE A-B SWITCH

By: Fred S. Schaeffer
84-55 Daniels Street #4F
Jamaica, NY 11435

I don't want to take business away from those manufacturers that make fancy 'A-B' type switches, but there is a better way to solve your primitive switching problems. My problem was to have two computers input into a single peripheral (e.g. printer, terminal of modem). Of course, it is just as easy to switch cables most of the time except that mine are rather inaccessible.

The IN-cables (those into the switch box/figure 1) come from I/O serial boards in Unit 1 (a S-100 type computer which I'm in the process of populating) and Unit 2 (my existing OSI equipment). If you are using pin 1 (frame ground) that should then be strapped to the 2nd cable (or both to) to pin 1 of the Female DB25 in the switch box. It is probably not even necessary to have pin 7 (signal ground) switched. Suffice it to say that both pin 2/3 and pin 5/7 DPDT switches must be thrown together and BOTH must enable either unit 1 OR unit 2.

Pin 5 is CTS; it is, however, only part of the handshaking formula. There seems to be no 'standard' way of wiring here; if you have no pin 5 connected (to a board) then it shouldn't matter whether the SPST switch is off or on, but I read some-

where that you can simulate handshaking to the peripheral by looping p5 to p8. So experiment...just be sure you don't connect the other end to a +V pin in your equipment.

I cannot claim credit for the wiring of the null switch. Dick Brannin of E. Williston, NY thought that one out. Basically, what happens here is that pin 2 and 3 from either unit 1 or 2 feeds into a 'common' or the middle tabs of a DPDT switch. By wiring as in figure 1, it effectively switches 2 and 3 on the one hand, or leaves p2=2 and p3=3 on the other hand.

Some printers need lead 20 connected; that becomes a problem because OSI's 525 board isn't necessarily wired for that. I use a Sooperspooler (Compulink Inc) which takes care of that and other problems.

The entire switch is in a small plastic equipment box, measuring about 4-1/2"L. x 2-1/2"W. x 1"H. Total cost excluding cable was $20.00 incl. 2 male DB25 with hoods that are plugged into the back panel of each computer. That is a bargain because most commercially available AB switches start at about $100. Usually, that gives you 25 switched lines, but no facility for a null modem. The switch I made can also connect two computers together with the null switch enabled.

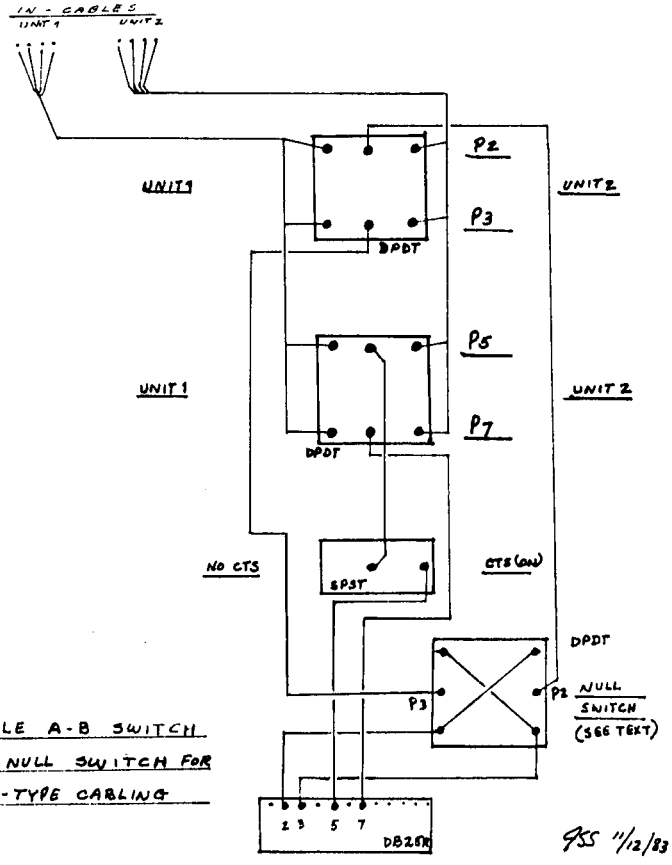


FIG. 1.
SIMPLE A-B SWITCH
WITH NULL SWITCH FOR
RS232-TYPE CABLING

READER PROFILE

ED:

I started working on OSI systems in 1978 on a C2-8S with 8-inch drives and OS65D V2.0. I purchased my own C2-4P DMF in 1979 and have since accumulated a couple of systems using boards from OSI, D&N, CCS, MIS, GENERIC & ORION. I've done several hardware and software additions on my own.

The hardware projects include the following:

1. A 'Head End Connector' interface so that I could connect a standard OSI EPROM burner.

2. A 'Head End' adaptor for a 'Hand Powered' paper tape reader.

3. Modifying the OSI disk interface to be switchable from $C000 to $C100 so that I could have both 8-inch & 5-inch drives active on the system at the same time and be able to boot from either.

4. A change to the 540A & 540B video boards to drive monitors with separated video inputs.

5. Modifying the 502 CPU board to replace the BASIC-IN-ROM with 2kx8 RAM chips.

The software projects have included the following:

1. Writing a keyboard/video driver with true upper/lower case input, full-screen cursor control, re-transmit from screen and print from screen. It replaced the standard keyboard-in and video-out calls of OS65D so it was directly useable from BASIC, Editor/Assembler, Extended Monitor and anything else that runs under OS65D.

2. A Modem driver integrated into the keyboard/video handler. This meant that with 1 command I could vector modem input to OS65D and/or OS65D output to the modem, or could run in local mode to OS65D, or as a dumb terminal to the modem. It provides no high level support such as phone number files, split screen (such as Rick Trethewey's nice package), etc., but is perfect for dialing up my system at work so I can work from home, and for doing file transfers to/from any remote system.

3. Regenerating source for OSI's Editor/Assembler and WP2. They both use a common interpreter and a lot of common code (in fact, WP2 still has a lot of unuseable Assembler code buried in it!). I've totally re-arranged WP2 and added some features.

4. Relocating OS65D from $2300 to $0300, a much more sensible location to me. I used Tom Berger's commented source of OS65D for help in this.

5. An extensive re-write of the XPLO package. If you're unfamiliar with this software, it's a very good structured programming language for 6502 systems. The language is very similar to Intel's PL/M language. The whole system as it is sent to you works fine on any OSI system with 8-inch disks and at least 32k of memory. It is composed of the XPLO compiler, the I2L interpreter, a text editor, and some small utilities. What I

didn't like about it is that the editor, compiler, interpreter, your source code and your executable code all had to be resident in memory at the same time! This restricted program size tremendously! I eliminated the editor, removed the compiler from being permanently resident, and converted the compiler to read its input from disk and generate the compiled output to disk. It also accepts files in WP2 format so that's what I use for my editor. I relocated the whole system to use the moved OS65D, so now my memory map has OS65D from $0300 to $1200, the I2L interpreter from $1200 to $2200, and user programs from there to the top of memory (about 39k on a 48k system). I also developed several utilities in XPLO including a directory utility package (this combined and extended the distributed XPLO utilities), a disk COPY program which copies whole disks, track ranges or files on 1 or 2 drive systems, a cross-reference program for Assembler or XPLO files, and numerous other programs. (The original work on regenerating the source of the interpreter was split with Tom Berger and most of the Compiler source was regenerated by him. The interpreter re-write and compiler extensions were mostly mine).

I would like to offer this package to anyone who wanted it, but since the original is still commercially available from the 6502 Program Exchange, I probably can't do that. If you have purchased the original and would like to have a copy of mine, send me proof of purchase, a disk and some postage, and it's yours.

6. The best available assembler for OSI that I've found is the A/65 Assembler. Unfortunately, when I got a copy, it didn't always work on my files, so again I regenerated and commented the source. I then optimized the input handler and disk handler, rewrote the symbol table sort routine, extended the file linking abilities to include switching between drives or volume (it waits for you to mount alternate disks), and added conditional assembly. This assembler is now effectively limited in speed only by the motion of the disk drives between tracks. Again, if you've bought the original, I can give you a copy of my version.

Future projects that I would like to do include moving to an 80 column video display, a REAL keyboard (from a Sperry UTS-20), and trying out DOS-65 (I bought a copy, it won't boot and I haven't had time to find out why). I also want to move OSI's EPROM program driver from BASIC to my XPLO system.

Leroy Erickson
Roseville, MN 55113

* * * * *

ED:

Enclosed herein is a check to renew my PEEK(65) subscription from a still loyal OSI user. I say this because it has become a monthly habit of mine to open to the back pages of PEEK or BYTE and count the number of people selling their OSI's and to shake my head and wonder. My machine, Eddie (Electronic Digital Data Interface Engine), who happens to be an OSI C2-4P 8" floppy system will always be with me. Mainly because I've put too darn much work into modifying him.

Eddie consists of:

- a 502 microprocessor board, a 540 video board, a D&N floppy controller/memory board, a 527 memory board, a front panel display board, and a CA-20 I/O board mounted on an eight slot backplane and enclosed in a 12"x12"x16" white and black plexiglass and metal cabinet.

- a Shugart SA-801R floppy drive mounted in a smaller cabinet.

- a detached keyboard (the 540 in a separate cabinet).

- an Epson MX-80 printer.

- various CA-20 connected peripherals including a 3 octave organ keyboard, Eprom burner, Steve Ciarcia's Sweet Talker voice synthesizer, complex sound generator, A/D converter (16 channel), stepper motor controller, and solid state AC switches.

Sometime in the near future, I hope to get a Corvus 10 megabyte Winchester going with my system. Since this drive comes with an intelligent controller, the physical interface should be relatively simple (such as the design that appeared in the Oct. '83 issue of Dr. Dobb's Journal), but the software patches to my operating system are still difficult (for me, anyway). Has anyone in OSI land had any experience with a project of this sort?

Finally, I would like to mention my first choice on my ISOTRON wish list. Namely, an upgrade for current OSI users based on the new Western Design CMOS 16 bit version of the 6502 (one version being pin-to-pin compatible with the old 6502!)

Douglas M. Petersen
Fresno, CA 93726

LETTERS

ED:

Maybe there are some fellow C4P computerists out there like me who are enjoying Dwo Quong Fok Lok Sow's software. I am using the WP6502 word processor with great satisfaction. There is one feature of it that I wish I could change. When you wish to print the text that you have written, the program gives you the option to choose whether you want the pages automatically numbered or not. The default choice is 01, in which case the pages are numbered beginning with page 01. However, in the preponderence of cases, as in the case of this letter, my text consists of only one page, and I don't want it numbered. In this case I select the option 00, and the page numbering is withheld. This works ok, but it means more keying for me. I would rather that the default choice would be 00, because that is the choice I make most frequently.

Does anyone know how to modify this program? Is Dwo Quong still in business? I have had no success in communicating with him or them or whatever.

Carl M. King
Sarasota, FL 33579

Carl:

We also have tried communicating with Dwo Quong on your behalf, and like you have been unsuccessful. I fear the news it not good. We do not have an answer for you, but hope one of our readers can be of help.

Peek Staff

* * * * *

ED:

I'm responding to the letter by J. F. McConkey III at Rockville. MD. The Superboard II/C1P is extremely easy to interface to a modem.

He can build up the RS-232 port if he likes, or do as I did, tap TTL data I/O lines of the ACIA (consult OSI schematics).

Also, I would like to respond to the two hardware expansion articles by Messrs. Cortes and Tasker in the Jan. 84 issue.

Mr. Cortes goes to the trouble of addressing each slot on the expansion board. Why? Does it ultimately save some decoding on the board in that slot?

Mr. Tasker's memory board duplicates some of the features I have on my home-brew expansion. However, he added two ICs that are really unnecessary. Those are IC8 and 9, 8T28 buffers. The two found on the 600 board are more than adequate for the task. But if buffering should be necessary, my parts catalog tells me that one 74LS245 would be cheaper and more effecient. The same applies to memory chips - one TMM2016 will replace four 2114s at about half the cost.

Lastly, why does he use a 1K resistor in series with +5 volts going to the inputs of IC-6? My TTL data sheets indicate that the chips can handle a high input up to +7 volts.

Bruce Showalter
Abilene, TX 79601

* * * * *

ED:

I recently purchased a CA-22 analog I/O board from another OSIer who never used it. I use it in conjuction with a modular analog synthesizer for audio processing and analysis. I have developed a spectrum analysis package (mucho graphics) which will analyze waveforms of any number of steps (samples) and print the computed data on a Gemini 10X, if desired. Requirements are 48K, polled keyboard with DAC, and a 5 1/4 inch disk drive. Anyone interested should send a 5 1/4 inch disk, with either return postage ($1.00) or software on the diskette (I'm interested in anything and everything). There is more to the package than I have described.

My question is, do any of your readers have the adventure 'Volcano of Kanthor'? I tried to buy it from Orion before they went under but they no longer handled it. I would gladly send the $50.00 list price to Mr. Bassman for it. Can any of your readers help?

Jack Deckard
3808 Laguna Dr.
Columbus, OH 43232

* * * * *

ED:

Reference: "System Disk Utility for OS65D" by David L. Kuhn, April 1984 issue.

I gave Mr. Kuhn's Utility a try on our C2 OEM and found that a change was required in four lines to make this "nifty" program work on our unit.

As follows:

```
220 Disk!"CA 2E79=08,1"
230 Disk!"SA 08,1=2379/1"
235 DISK!"CA 2E79=08,2"
250 Disk!"SA 08,2=2379/1"
```

Also, I have changed line 210 via "MEDUMP" (to find the new address) because on our disks we prefer "CREATE" and "DIRSRT" directly behind "BEXEC*".

Dick Wilkinson
Fairview, TN 37062

* * * * *

ED:

How does the "merge" command work or how do you merge a set of files in WP-6502 (The Chinese word processor)? When I go out of the processor by using "exec," it drops me into 65U and I can load or save files but there is no explanation for merge in the manual. I gather it is a 65U command function rather than WP-6502. Can anyone help? My copy does not have "fileclerk" if that helps.

Neil Dennis
Bliss, NY 14024

Neil:

Your WP-6502 must be a true antique. Every version we can remember has had in its screen menu Load, Save, & Merge, which appends the next file to be loaded to the back of the current file. The command is in WP-6502 not OS65U and simply does not reset the RAM pointer when making the next Load.

Peek Staff

* * * * *

ED:

Thank you for printing my letter in the April issue of PEEK(65). What I was interested in, however, was not how to rewrite Apple programs to run on the OSI. Rather, I wanted to find out what kind of hardware changes can be performed to make the OSI "Apple-compatible," i.e., run the Apple programs as written. Taking this a step further, how about IBM-compatibility. Anyone out there working on this?

T. J. Hirasuna
Yonkers, NY 10703

* * * * *

ED:

Re my article on a Tax Preparation program published in the Apr. 84 issue of PEEK(65), I am sorry to say a few minor bugs slipped by me, none of them major, thank God, but annoying to some I am sure.

I have corrected all these bugs, and have enclosed a new machine listing.

The corrections were made on the following lines:

Corrections on lines 185 and 415 now allows the proper filing status adjustment to be printed on Schedule A if printed directly after the file is loaded. Before it was not saved in the file and would have printed 00.

Corrections to lines 255,260, 270,505,515, and 770 correct round off errors that might occur.

Changes in lines 385,440,495, 520,600,645,1040, and the addition of line 522 serve two purposes. First in the old listing, if a person did not use Schedule B, but entered interest and dividend data directly on the 1040, he would have found that it disappeared when he printed it out. These changes correct that problem and speed up the calculations by eliminating unnecessary runs through the whole calculation set.

Finally, line 15 has been changed to reflect the revision.

Next year, I plan improvements to the program, and will keep you advised if you are interested.

Thank you for your continued support.

Robert S. Baldassano
San Jose, CA 95124

* * * * *

SEE LISTING ON NEXT PAGE

Corrections made. **"TAX PREPARATION V83.1"**

by: Robert S. Baldassano

```
15  PRINT!(20)&(5,10)!(1)"TAX PREPARATION V83.1":Q$=CHR$(34)

185 INPUT#6,D1$,D2$,D4$,D5$,D6$,D1,D7$,D2,N9:DISK CLOSE,6:POKE13026,171

255 PRINTUSINGU$"TOTAL TAXES    ";FNA(I(58))
260 PRINTUSINGU$"TOTAL PAYED    ";FNA(I(66))

270 PRINTUSINGU$!(31,2)BL$;FNA(ABS(A)):IFA<=0THENPRINT!(2,5,2)

385 S1=1:S2=1:S3=70:GOSUB525:GOSUB975:GOSUB665:GOSUB1045:GOTO100

415 PRINT#6,D2:PRINT#6,N9:DISK CLOSE,6:GOTO100

440 S1=2:S2=71:S3=100:GOSUB525:GOSUB975:GOSUB665:GOSUB1045:GOTO100

495 S1=3:S2=101:S3=133:GOSUB525:GOSUB730:GOSUB975:GOSUB1045:GOTO100

505 GOSUB760:PRINTUSINGU$!(1)"TOTAL INTEREST";TAB(44);FNA(I(4))

515 PRINTUSINGU$!(1)"TOTAL DIVIDENDS";TAB(44);FNA(I(5)):GOSUB1230

520 S1=4:S2=134:S3=DN:GOSUB525:GOSUB795:GOSUB975:GOSUB665:GOSUB1045
522 GOTO100

600 A=LEN(Z$):IFA=0THENRETURN

645 RETURN

770 PRINTL$(A);"  ";I$(A);TAB(44);:PRINTUSINGU$;FNA(I(A)):GOTO780

1040 I(32)=I(32)+I(30)+I(31):I(33)=I(22)-I(32):I(34)=I(33):RETURN
```

* *

ED:

I am replying to a letter in the January 84 issue by Leroy Erickson that stated a BASIC IN ROM serial system cannot be supported. The ROM I have is a CM7157N and it supports SERIAL BASIC IN ROM with the following jumpers: 2 to 8 and 3 to 13 and the ROM seems to be similar to his "SYNMON."

I have always used a serial system and when I purchased my first CPU board (a 502), it came with the BASIC ROM set and the above jumpers. My first problem was to find a way to save programs. The BASIC IN ROM expects the cassette port to be a 430 board, which was no longer available at the time, so I built my own on a prototype board. Later on I added an 8" disk and used my 430 board for the serial printer port. After getting tired of the slow transfer rate I added hand shaking and changed the baud rate. I'm enclosing a schematic of my changes for this mod.

This will sound like desertion, but I recently purchased a LOBO MAX80 that runs CPM and LDOS and I wanted to switch the 8" drives between the two systems but came up with a problem. The LOBO uses a common drive ready line which OSI uses separate lines. After some experimentation, I found that by cutting the ready line to drive 2 on the controller board and connecting the two lines on the 6820 PIA together, I could use the same drives on both systems. The second schematic shows how this was done.

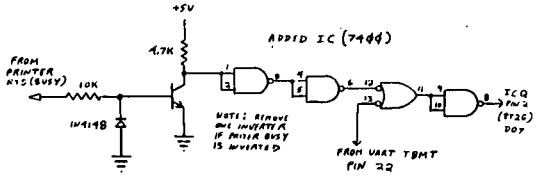


FIGURE 1

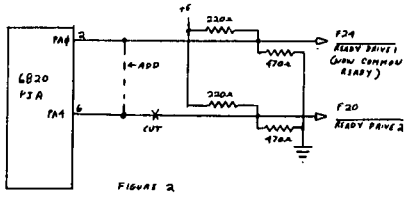


FIGURE 2

I hope the above can help someone who may have similar problems.

Robert H. Foltz
Bronx, NY 10461

AD$

C2 serial, 56K dual 8" drives, 1 parallel & 2 serial ports, OS-65D, OS-65U, WP2, some diagnostic software. Much documentation. $750 FOB. Also, UCSD Pascal with FORTRAN ($225); Sanders' SORT/MERGE ($40), RESEQ ($20); WP6502 ($25); 125 used diskettes ($100); 10 plastic storage cases ($20). $1050 FOB for everything. For complete list, send SASE: Roy Harris, Rt. 2, Box 340, Charlottesville, VA. Phone (804) 293-3605.

* * * * *

Send for free catalog, Aurora Software, 37 South Mitchell, Arlington Heights, IL 60005.

* * * * *

FOR SALE: Dual 8" Siemens drives single sided, in OSI case with power supplies. Asking $200 including shipping. Also new OSI 542 keyboard $50 with case, $40 without. Robert H. Foltz, 1911 Mulford Ave., Bronx, NY 10461. (212) 863-0964.

* * * * *

USED OSI, C2. Extra board in processor. Dual disk drive (has problems). Software: word processing, accounting, planner, report writer. We used the report writer package successfully for three years to handle our membership data and mailings. Seven manuals provided and several years of PEEK(65). $300 or best offer. Council on Municipal Performance, (212) 243-6603.

* * * * *

Good prices on collection of OSI equipment and accessories. Send SASE for complete list. Ricky Peterson, 206 Pine Valley, Warner Robins, GA 31093.

* * * * *

WANTED: 610 & 620 boards, mini floppies, and other items for SBII expansion. Write with price to Paul Harris, Apt 5A, 7 Hamilton Rd., Morristown, NJ 07960.

* * * * *

FOR SALE: C8PDF 48K polled keyboard, 65U & 65D, disks & manuals for video editor, A/R, A/P, G/L, Payroll, Inventory, Assembler, Extended Monitor, Misc. $1100 or best offer. John Payne, 4610 Don Juan, Abilene, TX 79605. 915-695-2196 after 6 PM CST.

* * * * *

FOR SALE: OSI 525 Memory Board with 16K RAM (2114). $70.00, including shipping. Larry Wagner, Rt. 2, Defiance, OH 43512, 419-782-3660.

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:

GOODIES for OSI Users!

PEEK (65)
The Unofficial OSI Users Journal

P.O. Box 347 • Owings Mills, Md. 21117 • (301) 363-3268

( ) **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 — $15.00 $ _____

( ) **C2/C3 Sams Photo-Facts Manual.** The facts you need to repair the larger OSI computers. Fat with useful information, but just — $30.00 $ _____

( ) **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 $ _____

BOOKS AND MANUALS (while quantities last)

( ) **65V Primer.** Introduces machine language programming. — $4.95 $ _____

( ) **C4P Introductory Manual** — $5.95 $ _____

( ) **Basic Reference Manual** — (ROM, 65D and 65U) — $5.95 $ _____

( ) **C1P, C4P, C8P Users Manuals** — ($7.95 each, please specify) — $7.95 $ _____

( ) **How to program Microcomputers.** The C-3 Series — $7.95 $ _____

( ) **Professional Computers Set Up & Operations Manual** — C2-OEM/C2-D/C3-OEM/C3-D/C3-A/C3-B/C3-C/C3-C' — $8.95 $ _____

( ) Cash enclosed ( ) Master Charge ( ) VISA

Account No. _____ Expiration Date _____

Signature _____

Name _____

Street _____

City _____ State _____ Zip _____

TOTAL	$_____
MD Residents add 5% Tax	$_____
C.O.D. orders add $1.65	$_____
Postage & Handling	$ 3.50
TOTAL DUE	$_____

POSTAGE MAY VARY FOR OVERSEAS