

AN ASSEMBLY LANGUAGE SUBROUTINE
By John Dow

In the last newsletter I tried to explain what assembly language is. This time, I include a complete example of a subroutine written by Jerry Rowell, past VP of our group and one of the more fluent assembly language programmers. The routine was written using the Dow Editor/Assembler, so the syntax is nonstandard in that there is a colon after each label and semi-colons are used instead of commas. The routine is called as follows from Basic:

CALL LINK("SORT",N,A())

where N is the number of (numeric) values in the array A to be sorted and where SORT has been entered into the REF/DEF table pointing to the first location

```

000 LWPI >70B8
004 CLR R0
006 LI R1;1
00A BLWP @NRF @E1 SIZE OF END
00E MOV @FAC;R1
012 MOV B R1;R2
014 SWPB R2
016 CI R2;>41
01A JNE L0
01C SWPB R1
01E CLR R2
020 MOV B R1;R2
022 SWPB R2
024 LI R3;100
026 LI R4;R2
02A MOV R3;R2
02C MOV B @>B34C;R0
030 SWPB R0
032 A R0;R2
034 MOV R2;@N
038 JMP L1
03A L0: SWPB R1
03C SWPB R2
03E MOV B R1;R2
040 SWPB R2
042 MOV R2;@N
046 L1: SRL R2;1
048 CI R2;0
04C JNE NXT
04E B $K17
050 NXT: MOV R2;@M8
054 MOV @N;R3
058 S R2;R3
05A LI R0;1
05E SWPB R0;@M8

```

```

062 L2: MOV @J8;@I8 I8=J8
068 L3: MOV @M8;@L8
06E A @I8;@L8 L8=I8+M8
074 MOV @I8;R0
078 LI R1;2
07C BLWP @NRF
080 LI R4;FAC
084 LI R5;X18
088 FIL: MOV #R4+;#R5+
08A CI R5;>712A
08E JNE FIL
090 MOV @L8;R0
094 BLWP @NRF
098 LI R7;FAC
09C LI R8;X18
0A0 LP: MOV #R7+;R4
0A2 MOV #R8+;R5
0A4 C R5;R4
0A6 JLT L4
0A8 JNE LP0
0AA CI R8;>712C
0AE JNE LP
0B0 LI R1;1
0B2 JLE L4
0B4 LP0: MOV @I8;R0
0B8 BLWP @NAS
0BC LI R4;FAC
0C0 LI R5;1
0C4 LP1: MOV #R5+;#R4+
0C6 CI R5;>712A
0CA JNE LP1
0CC MOV @L8;R0
0D0 BLWP @NAS
0D4 S @M8;@I8 I8=I8-M8
0DA MOV @I8;R6
0DE CI R6;0
0E2 JGT L3
0E4 L4: SWPB R1
0E6 LI R1;1
0EC JLE L2
0EE JMP L1
0F0 FAC: EQU >B34A
0F2 NRF: EQU >6044
0F4 NAS: EQU >6040
0FA N: EQU >6040
0FC RW: EQU >6040
0FE JM: EQU >6040
0FF L0: EQU >6040
100 X18: EQU >6040

```

Load the subroutine at location 7C00

REPRINTED FROM PITTSBURGH
PERIPHERAL

WRITING A PROGRAM

BY ROGER WILLS

A useful piece of software should be flexible enough that you can do a number of things with it. Unfortunately if you only have console ram available many such programs cannot be used because they occupy more ram than is provided. If you cannot afford to buy extra memory the answer is to write and modify your own programs. Usually one can write programs whilst working with the computer. This is alright for small programs, but it does not work for larger programs because you usually get lost in GOTOs and GOSUBs. You need to structure your program using a flow chart before actual line by line programming is initiated. It's also easier to work in X-BASIC (I had to convert the original program to BASIC so that all of the club members could possibly use it). Basically the program tracks all your purchases and commissions as well as calculates the value of your portfolio. The variables are explained in lines 100-190, and the arrays dimensioned information. 13 DATA statements are shown so that all LOOPS must use 13 to READ and PRINT. The menu (12 40-1340) gives 3 options, and accepts your input. All good programs contain statements like line 1310 which restrict the data input to the correct variable, rather than telling you that you have inserted the wrong variable. 230-1350 prints the original purchase information. Since the data frequently goes across the screen very quickly. The program also enables you to dump the data to a printer (lines 1380-1430). Note the STRING variable P\$ allows for any type of printer, baud rate etc. In order to calculate the value of the Portfolio you need to read the data and insert the current price of security on each prompt. RESTORE (line 1315) is needed to read the data a second time. I

used lines (240-260 to read the data and line 270 to avoid displaying the data a second time. Cash values are stored in the one line DATA statement at 1190. Next month we will look at the "sell side" of the portfolio. I hope some of you will find this program useful.

```
100 REM***INVESTMENT PORTFOLIO***
110 REM ****PART 1,LISTS ALL SECURI
    TIES,BUYING PRICES****
120 REM **** TOTAL MONEY INVESTED,
    AND CALCULATES PORTFOLIO VALUE.
130 REM W(50)=NUMBER OF UNITS
    PURCHASED
140 REM X$(50)=NAME OF UNITS
    PURCHASED
150 REM Q(50)=PRICE PER UNIT
    PURCHASED
160 REM Z(50)=OTHER COSTS FOR
    PURCHASES
170 REM INV(I)=AMOUNT INVESTED
180 REM TOTINV = TOTAL INVESTED
190 REM MARVAL=MARKET VALUE OF
    HOLDINGS IN THAT SECURITY
200 CALL CLEAR
210 DIM W(50),X$(50),Q(50),Z(50),
    INV(50),MARVAL(50)
220 GOSUB 1225
230 CALL CLEAR
240 FOR I=1 TO 13
250 READ W(I),X$(I),Q(I),Z(I)
260 NEXT I
270 IF RZ=2 THEN 510
280 PRINT "ORIGINAL PURCHASE
    INFORMATION"
290 PRINT
300 FOR I=1 TO 13
310 PRINT W(I);X$(I);Q(I);Z(I)
320 PRINT
330 NEXT I
340 FOR I=1 TO 13
350 INV(I)=(W(I)*Q(I))+Z(I)
360 NEXT I
370 FOR DELAY=1 TO 1000
380 NEXT DELAY
390 PRINT "AMOUNT INVESTED PER
    SECURITY"
400 FOR I=1 TO 13
410 PRINT X$(I);INV(I)
420 NEXT I
430 PRINT
440 TOTINV1=INV(1)+INV(2)+INV(3)+
    INV(4)+INV(5)+INV(6)+INV(7)
444 TOTINV2=INV(8)+INV(9)+INV(10)+
    INV(11)+INV(12)+INV(13)
448 TOTINV=TOTINV1+TOTINV2
450 PRINT "TOTAL INVESTED";TOTINV
```

CONTINUED

```

460 FOR DELAY=1 TO 1000
470 NEXT DELAY
480 GOSUB 1350
490 GO TO 1225
500 PRINT
510 CALL CLEAR
520 PRINT "TYPE IN THE MARKET PRICE
OF EACH UNIT"
530 PRINT ::
540 PRINT X$(1)
550 INPUT AA
560 PRINT X$(2)
570 INPUT AB
580 PRINT X$(3)
590 INPUT AC
600 PRINT X$(4)
610 INPUT AD
620 PRINT X$(5)
630 INPUT AE
640 PRINT X$(6)
650 INPUT AF
660 PRINT X$(7)
670 INPUT AG
680 PRINT X$(8)
690 INPUT AH
700 PRINT X$(9)
710 INPUT AI
720 PRINT X$(10)
730 INPUT AJ
740 PRINT X$(11)
750 INPUT AK
760 PRINT X$(12)
770 INPUT AL
780 PRINT X$(13)
790 INPUT AM
800 PRINT
810 MARVAL(1)=W(1)*AA
820 MARVAL(2)=W(2)*AB
830 MARVAL(3)=W(3)*AC
840 MARVAL(4)=W(4)*AD
850 MARVAL(5)=W(5)*AE
860 MARVAL(6)=W(6)*AF
870 MARVAL(7)=W(7)*AG
880 MARVAL(8)=W(8)*AH
890 MARVAL(9)=W(9)*AI
900 MARVAL(10)=W(10)*AJ
910 MARVAL(11)=W(11)*AK
920 MARVAL(12)=W(12)*AL
930 MARVAL(13)=W(13)*AM
940 FOR I=1 TO 13
950 PRINT X$(I);MARVAL(I)
960 NEXT I
970 PORTVAL(1)=MARVAL(1)+MARVAL(2)
+MARVAL(3)+MARVAL(4)+MARVAL(5)
+MARVAL(6)
975 PORTVAL(2)=MARVAL(7)+MARVAL(8)
+MARVAL(9)+MARVAL(10)
980 PORTVAL(3)=MARVAL(11)+MARVAL
(12)+MARVAL(13)
990 PORTVALUE=PORTVAL(1)+PORTVAL(2)

```

```

+PORTVAL(3)
1000 PRINT
1010 PRINT "PORTFOLIO VALUE IS";
PORTVALUE
1020 PRINT
1030 REM***DATA STATEMENTS***
1040 DATA 100,A CORP,25.5,73
1050 DATA 100,12.5% B CORP
BOND,25,72
1060 DATA 70,C CORP LOAN 1992,40,52
1070 DATA 50,15% D CORP BONDS,32,79
1080 DATA 300,E CORP,123,105
1090 DATA 120,WARRANTS F CORP,63,42
1100 DATA 190,G CORP,56.5,167
1110 DATA 150,H CORP,26.5,31
1120 DATA 100,J CORP,87,74
1130 DATA 80,K CORP,38,60
1140 DATA 200,L CORP,27.75,132
1150 DATA 250,M CORP,68.75,234
1160 DATA 60,N CORP,89,270
1170 REM CALCULATE TOTAL PORTFOLIO
INCLUDING CASH
1180 READ C
1190 DATA 633
1200 PRINT "TOTAL PORTFOLIOI
INCLUDING CASH IS ";PORTVALUE+C
1203 FOR I=1 TO 1000
1205 NEXT I
1210 CALL CLEAR
1211 INPUT "PRINT THIS DATA?(Y/N)
":CH$
1212 IF CH$="N" THEN 1225
1213 PRINT
1214 INPUT "ENTER PRINTER'S NAME:
":P$
1215 OPEN #4:P$
1216 FOR I=1 TO 13
1217 PRINT #4:X$(I),MARVAL(I)
1218 NEXT I
1219 PRINT ::
1220 PRINT #4:"PORTFOLIO VALUE IS"
;PORTVALUE
1221 PRINT
1222 PRINT #4:"CASH IS";C
1223 PRINT #4:"TOTAL PORTFOLIO IS"
;PORTVALUE+C
1224 CLOSE #4
1225 CALL CLEAR
1226 PRINT " INVESTMENT
PORTFOLIO"
1230 PRINT ::
1240 PRINT "DISPLAY LIST OF
PURCHASES(1)"
1250 PRINT ::
1260 PRINT "CALCULATE CURRENT
MARKET VALUE OF PORTFOLIO(2)"
1270 PRINT ::
1280 PRINT "EXIT";"(3)"
1290 PRINT ::
1300 INPUT "SELECT YOUR OPTION":RZ

```

WRITING CONTINUED

```

1310 IF (RZ<1)+(RZ>3) THEN 1280
1315 RESTORE
1320 IF RZ=1 THEN 230
1330 IF RZ=2 THEN 230
1340 IF RZ=3 THEN STOP
1350 CALL CLEAR
1360 INPUT "PRINT THIS DATA?(Y/N)
":CH$
1370 IF CH$="Y" THEN 1380 ELSE
1225
1375 PRINT
1380 INPUT "ENTER PRINTER'S NAME:
":F$
1390 OPEN #3:P$
1400 FOR I=1 TO 13
1410 PRINT #3:W(I),X$(I),Q(I),Z(I)
,INV(I)
1420 NEXT I
1430 CLOSE #3
1440 RETURN

```

BIGGIES BITS

In the MARCH newsletter two programs on creating cassette data files were reprinted in part from the PIONEER VALLEY USERS GROUP. These are not correct since for LOOPS greater than 10 memory space must be reserved by DIMENSIONING THE ARRAYS. The routine shown demonstrates this point, and enables data to be loaded into the computer and printed onto the screen.

```

100 OPEN#1:"CS1",INTERNAL,FIXED
192,INPUT
110 DIM W(50),X$(50)
120 FOR I=1 TO 33
130 INPUT#1:W(I),X$(I)
140 PRINT:W(I),X$(I)
150 NEXT I
160 CLOSE#1

```

LETTERS

Dear CONNI,
I have a TI994/A computer. I am a Biologist and Teacher in the Universidad Veracruzana here in Xalapa Veracruz, Mexico.

I would like to make contact with your group, because in my country it doesn't exist. I would be very greatful if you could send me information about Membership, Newsletter subscription etc.

Thank you very much,
Biol. Armando Lopez R.
Xalpa, Veracruz Mexico

ANY Teachers want to correspond?
(ED).



* TIGERCUB SOFTWARE *

100 Different programs for the TI99/4A programed in TI Basic, No Peripherals needed.
GAMES-PUZZLES-EDUCATIONAL-MUSIC-D
SPLAYS-UTILITYS-ALL
Original-Debugged-User-Friendly-
Thoroughly Self-Documented. Only
\$3. ea! Minimum Order \$12.
CALL or WRITE for Catalog
(614) 235-3545, 156 Collingwood
Avenue Columbus, Ohio, 43213

INF-WARE INC

345 GLEN MEADOW ROAD
DUBLIN, OHIO 43017
889-9011

"MUSIC FROM STRINGS" X-BASIC

```

50 REM TI SYDNEY HOME USER GROUP
100 DISPLAY AT(12,6)ERASE ALL:"MUSIC FROM STRINGS"
110 A$="1A1E1K1K1K1K1E1A1A1E6K1K1P1U1U1U1U1_1_1_U6P1_1X1U1U1U1U1P1K1K1K1E1E1E1E
IP1K1I1E1A1A1K1P1U1P1E1I15K"
120 FOR A=1 TO 98 STEP 2 :: CALL SOUND(VAL(SEG$(A$,A,1))*200,(ASC(SEG$(A$,A+1,1)
)-36)*10,0):: NEXT A
130 DIM B(28):: C=262 :: FOR D=0 TO 27 :: B(D)=INT(C*1.059463094^D):: NEXT D ::
B(28)=32000 :: B$="cKcKcNeJaKcMcOcPeG=McKcMcKcJgK"
140 FOR F=1 TO LEN(B$)STEP 2 :: CALL SOUND((ASC(SEG$(B$,F,1))-95)*120,B(ASC(SEG$
(B$,F+1,1))-64),5):: NEXT F :: GOTO 110

```

BIGGIES BITS

Dear Biggie,

you asked for tips and contributions and although I do write assembler I did not have any simple examples available. However I found a need for string editing in basic which I thought you might want to use in your column.

For a general ledger program I wrote, all data is stored in a string array. This conserves memory, compacts casset storage, and gives a more flexible internal manipulation. However, editing is tough unless one intends to re-enter an entire line.

The following example can be entered as a subroutine in any program where character editing is desired. For maximum speed, constant screen update via print has been modified except where essential.

The beginning provides a SAMPLE STRING, the center section does the STRING MANIPULATION and the last section CALCULATES the CURSER PIONTER.

The program is quite short and easily followed with the REMarks included in the listing.

I enjoy the newsletter, but would very much like to see a catalog, and policy for program exchange.

I wish to thank the group for allowing me to show my printer at the last meeting.

sincerely yours
W Schadt

See the librarian at the July meeting, it looks as if we are all going to see a list and policy at last. Bring your printer any time. I'm sure others will be interested, and thanks for the program. (B)

```
50 CALL CLEAR
100 I=0
110 A$(I)="THIS IS A SAMPLE STRING TO EDIT.FOLLOW
      TI EDIT RULES"
120 GOSUB 150
130 PRINT "*****LET'S TRY AGAIN*****"
```

```
140 GOTO 110
150 P=1
160 I=0
170 IF LEN(A$(I))>1 THEN 190
180 A$(I)=" "
190 A$(I)=" "&A$(I)
200 REM SHOW INPUT+CALC CURSER
210 GOTO 640
220 IF (P+1)>LEN(A$(I))THEN 230 ELSE 240
230 A$(I)=A$(I)&" "
240 CALL KEY(0,KEY,STAT)
250 REM SHOW CURSER
260 CALL HCHAR(LIN,COL,95)
270 CALL HCHAR(LIN,COL,ASC(SEG$(A$(I),P+1,1)))
280 IF STAT=0 THEN 240
290 REM RT ARROW
300 IF KEY=9 THEN 310 ELSE 380
310 P=P+1
320 INS=1
330 IF (P+1)>LEN(A$(I))THEN 340 ELSE 660
340 INS=0
350 A$(I)=A$(I)&" "
360 GOTO 630
370 REM LEFT ARROW
380 IF KEY=8 THEN 390 ELSE 460
390 P=P-1
400 INS=1
410 IF (P+2)>LEN(A$(I))THEN 430
420 IF P<1 THEN 430 ELSE 660
430 P=1
440 GOTO 660
450 REM ENTER
460 IF KEY=13 THEN 470 ELSE 510
470 PRINT A$(I)
480 A$(I)=SEG$(A$(I),2,255)
490 RETURN
500 REM DEL
510 IF KEY=3 THEN 520 ELSE 560
520 A$(I)=SEG$(A$(I),1,P)&SEG$(A$(I),P+2,255)
530 A$(I)=A$(I)&" "
540 GOTO 630
550 REM INSERT
560 IF KEY=4 THEN 570 ELSE 590
570 INS=0
580 GOTO 660
590 A$(I)=SEG$(A$(I),1,P)&CHR$(KEY)&SEG$(A$(I),P+1+INS,255)
600 P=P+1
610 REM SHOW CHAR
620 CALL HCHAR(LIN,COL,KEY)
630 IF INS=1 THEN 660
640 PRINT A$(I)
650 REM CALC CURSER
660 LIN=24-(INT(LEN(A$(I)))/28)-5
670 LIN=LIN+INT(P/28)
680 COL=P
690 IF COL>27 THEN 700 ELSE 720
700 COL=COL-28
710 GOTO 690
720 COL=COL+3
730 GOTO 220
```

THE BEST OF JSC

(Original authors)

Jane McAshan/Mike Matula

Reprinted in part from JSC Newsletter. The summer consumer Electronics Show (CES) was held in Chicago on June 5-8 1983. Texas instruments is giving the Computr, Expansion box, Monitor & Speech Synthesizer a new look. The housings will be entirely made of plastic, and have the new standard "Computer Beige" color that has become popular in the industry. The NEW 994/A will show "Version 2.2" on the title screen.

TI had created quite a furor at the CES with what some of the manufacturers see as an attempt to make the TI994/A INCOMPATIBLE with third party Software. In advertisements in literature distributed at CES, TI pointed out that it holds a number of patents on an "auto incrementing memory" circuit and other circuits that are now being used or may be used in the TI hardware & software. They also indicated that they plan to make use of those circuits from now on. This means some software released or about to be released by third party software companys without authorization from TI, may not work on the future TI994/A computers. TI made the decision to start making use of the circuits because wants to maintain control over both the DISTRIBUTION and QUALITY of the software. They see it as a control/protection for customers, dealers and software authors as well for themselves.

Imagic announced an agreement with TI to release some of thier games for the 994/A and some other software companys may soon announce a similar agreement However Atari will not , it seems Ataris' plans run in disagreement with TIs insistance that the large installed base of 994/As' is thiers and thiers alone, to mine.

TI has not ALIENATED everyone though. E.T. and his adventures

on land will become available 4th qtr for a srp. of \$39.95. There are two other E.T. games planned for 4th qtr. as well "Word Radar" & "Word Invasion" developed by DLM for TI.

NINETY-NINER WARE is another of the companys who have chosen to stay with the TI. SCHOLASTIC INC. announced its line of software called "WIZWARE", for kids 6 to 16 with titles like "TURTLE TRACKS", "SOAREPAIRS" AND "MICROZINE". They like Ninety-Ninerware are cassette based. I could go on but the list is extraneous. In a nut shell, all of a sudden every body wants the new kid to be their friend. (remember when people said "you own a what?")

The best I could get on the "New computer" (you know the one), was that it did make it to the show, but was kept under lock & key for various & surdry reasons One of which might be, "TI will not announce any new hardware or software products unless it can be shipped within 60 days".

well from what I've been told and heard about CES "It wern't no big thing" TI I dare you to come to Columbus Ohio, you might find out that WE would help you put on a better show. Did anyone notice ANY other groups who

MEETING AGENDA

The meeting will beging at 9:30 AM on JULY 9th at the Martin Janis Senior Citizen Center on the Ohio State fair grounds.

After our regular meeting Irwin Hott, who is blind will demonstrate the use of Text to Speech using Terminal Emulator II & Speech Synthesizer. This should be an interesting demonstration.

Everyone is welcome to bring thier Equipment.

The software contest is still on and will be discussed at the meeting. If you wish to donate something to the library, Please bring it with you. And of course if you would like to become a member Art will be there to sign you up. See you there!

REM

NOTE; JSC is looking for a program to place a Space Shuttle on thier newsletter. They would like it to run on an Epson MX80. By the way guys Jim Peterson of CONNI is probably flattered that you reprinted his music subroutines from Biggies Bits.

NOTES OF INTEREST BY PAT SKEELE

BASIC

HOME EXPENSES AND INCOME

```
10 OPEN #1:"PI0"
20 CALL SCREEN(6)
30 CALL CLEAR
40 PRINT " EXPENSE/INCOME REPORT":::
41 FOR DELAY=1 TO 150
50 REM BY PAT SKEELE
55 NEXT DELAY
60 CALL CLEAR
70 CALL SCREEN(4)
80 INPUT "REPORT FOR (MONTH/YEAR) ":M$
90 PRINT
100 INPUT "NUMBER OF EXPENSE ITEMS? ":NE
110 PRINT
120 INPUT "NUMBER OF INCOME ITEMS? ":NI
130 DIM ES$(25),EA(25),IS$(25),IA(25)
140 CALL CLEAR
150 CALL SCREEN(8)
160 PRINT TAB(10);"EXPENSE DATA":::
170 FOR XE=1 TO NE
180 PRINT "# ";XE::
190 INPUT "EXPENSE SOURCE? ":ES$(XE)
200 PRINT
210 INPUT "AMOUNT? ":EA(XE)
220 TE=TE+EA(XE)
230 CALL CLEAR
240 NEXT XE
250 CALL CLEAR
260 CALL SCREEN(16)
270 PRINT TAB(10);"INCOME DATA":::
280 FOR XI=1 TO NI
290 PRINT "# ";XI::
300 INPUT "INCOME SOURCE? ":IS$(XI)
310 PRINT
320 INPUT "AMOUNT? ":IA(XI)
330 TI=TI+IA(XI)
340 CALL CLEAR
350 NEXT XI
360 CALL CLEAR
370 CALL CLEAR
380 CALL SCREEN(8)
390 PRINT TAB(4);"INCOME/EXPENSE STATEMENT"::
400 PRINT TAB(5);M$::
410 PRINT "EXPENSES";TAB(9);"(AMT)","INCOME";TAB(23);"(AMT)"
420 CALL HCHAR(24,1,45,32)
430 PRINT
```

```
440 IF NE>NI THEN 450 ELSE 470
450 XP=NE
460 GOTO 480
470 XP=NI
480 FOR X=1 TO XP
490 PRINT ES$(X),IS$(X);TAB(6);EA(X);TAB(18);IA(X)
500 NEXT X
510 CALL HCHAR(24,1,61,32)
520 PRINT
530 PRINT "TOTALS";TAB(6);TE;TAB(18);TI::
540 INPUT "DO YOU WANT THE ANALYSIS OF THE DATA(Y/N)":CH$
550 IF CH$="Y" THEN 560 ELSE 640
560 CALL CLEAR
570 CALL SCREEN(16)
580 PRINT " EXPENSE/INCOME ANALYSIS":::
590 TT=TI-TE
600 IF TT>TE THEN 610 ELSE 630
610 PRINT " YOU HAVE A SURPLUS OF $";TT;"THIS
MONTH.":::
620 GOTO 640
630 PRINT " YOU ARE IN DEBT $";TT;" THIS MONTH.":::
640 END
```

BASIC TENDERFOOT PART 2

by Niraj Shah

Remember that last month we were talking about how to STOP the execution of a program that lets the user assimilate the information at his own rate. The two methods are:

- 1) USING A DELAY LOOP
- 2) USING A CALL KEY STATEMENT

The first method was shown last month. This month we will go over the second method.

Using a CALL KEY statement lets the user tell the computer when he is done with the current data.

The only disadvantage is that it requires the user to give input to the computer even though his hands may be busy. Here is how to implement the second method:

```
100 REM USING A CALL KEY
STATEMENT
110 CALL CLEAR
120 PRINT "HELLO, MY NAME IS
TI-99/4A"::
130 PRINT "THIS REQUIRES YOU TO
TELL ME"::
140 PRINT "WHEN YOU ARE DONE
READING"::
```

CONTINUED

TENDERFOOT CONTINUED

```
150 PRINT "THIS SCREEN":  
160 CALL KEY(O,K,ST)  
170 IF ST=0 THEN 160  
180 PRINT "YOU JUST PRESSED A  
KEY!":  
190 END
```

The important lines in the above program are lines 160 and 170. Line 160 tells the computer to be on the look out for someone pressing a key on the keyboard.

If someone has pressed a key then the variable {K} is given a numeric value by the computer telling exactly what KEY was pressed. Also, the variable {ST} is assigned a NON-ZERO value by the computer. This means that the equality in line 170 is false and the computer will go on to line 180. But if a key was NOT pressed then the variable {ST} will be assigned a value of ZERO.

This causes the equality in line 170 to be true thus forcing the computer to go BACK to line 160 and keep on looking for a key to be pressed.

Remember that the variable {K} stands for which KEY was pressed and {ST} stands for the current STATUS of the keyboard (ie: if a key was pressed or not).

So, the effect of this second method is that the user presses any key to signal that he is done processing the information on the screen. One disadvantage to this method is that there is no way of telling the user to "PRESS ANY KEY TO CONTINUE".

Because if we do then the screen scrolls up and we end up losing the DATA on the screen. So, what is the solution? HUMAN NATURE!

Humans are naturally inquisitive creatures, so when they find that nothing is happening they will try to do something about it. In this case, the human will hit the keyboard or press <ENTER> to see what will happen. VOILA! The human did exactly what was desired of him without him knowing it!

NEXT MONTH: WHAT CAN AN
<IF-THEN-ELSE> BE USED FOR.

The following program is a modified version for cassette and Gemini 10 Printer. It first appeared in December 1982, 99'er Magazine as "TEXSCRIBE" by DAVID G. BRADER. The author of this article has filed an affidavit with Spirit of 99, stating no misrepresentation is meant by this article. It is published only with the intent of showing the versatility of the aforementioned printer and is not for sale. (EDITOR)

MODIFIED WORD PROCESSING

With the <GEMINI 10> Printer

by Albert L. Allen

I know that many CONNI members have read the February 1983 issue of 99'er magazine pertaining to the GEMINI 10 printers. I have compared the the Gemini with many printers, in this price range, and found that it is one of the best on the market.

I have interfaced the Gemini and the TI/99-4A computer through the parallel port of the RS-232. The connecting cable wiring is simple and appeared in the same issue of 99'er mag. My cable was made up for me by some very helpful people at the Home Computer Store, 385 Main ST. Westerville at a very reasonable price.

Now I wanted to do some word processing, so I hunted up the 99'er magazine article 'TEXSCRIBE' in the Dec. 1982 issue. I read through the article and the program and found it was made for the TI printer. The program uses <CONTROL> functions to tell the printer how you want to print

your text. But I found this to be limited for all the functions that the Gemini 10 had to offer. So I modified the program to the word processor you see in this article.

It uses less than 10K of memory in Extended Basic and files are stored on cassette tape. The files may be stored on disk with the addition of a simple routine found in the 'TEXSCRIBE' article. All of the control functions can be entered anywhere in a line and they normally produce no visible character until the text is listed out; i.e. Control M 'TEXT' Control T will produce doublewide printing and cancel doublewide printing. You can use emphasized, double strike, italic, small or capital letters all intermixed on any line. You can also choose the number of times to print the text and to format a page of text at will. This program is comparable with anyone sold on the market in program form.

A brief discussion of the program is in order here. Line 130 sets the normal printer defaults; LB is line spacing of 1/6 inch, LC is 66 line per page, LD is header length set at 0, LE is left hand margin set at 0 and LG is the print front set at pica. Lines 140-180 set Control Functions. Lines 3130-3300 format printer functions for 'PAGE' printing with default values censored for change. Below is a list of CONTROL FUNCTIONS as printer commands.

The control functions are accomplished by;

PUSH	AND	
CONTROL G	=	BELL
CONTROL H	=	BACKSPACE
CONTROL I	=	HORIZONTAL TAB
CONTROL J	=	LINEFEED
CONTROL K	=	VERTICAL TAB
CONTROL L	=	FORM FEED
CONTROL M	=	CARRIAGE RETURN
CONTROL N	=	DOUBLE WIDE PRINTING
CONTROL O	=	CONDENSED PRINTING
CONTROL R	=	CANCEL CONDENSED PRINTING
CONTROL T	=	CANCEL DOUBLE WIDE PRINTING
CONTROL .	=	ESCAPE

The next functions require escape (CONTROL .) and then a figure.

PUSH	AND,	THEN	PUSH	
CONTROL .	E	=	EMPHASIZED PRINT	
CONTROL .	F	=	CANCEL EMPHASIZED PRINT	
CONTROL .	G	=	DOUBLESTRIKE PRINTING	
CONTROL .	H	=	CANCEL DOUBLESTRIKE PRINTING	
CONTROL .	S 0	=	SUPERSCRIPIT	
CONTROL .	S 1	=	SUBSCRIPT	
CONTROL .	T	=	CANCEL SUB/SUPERSCRIPIT	
CONTROL .	V 1	=	PRINT SLASHED ZERO	
CONTROL .	V 0	=	CANCEL SLASHED ZERO	
CONTROL .	- 1	=	UNDERLINE	
CONTROL .	- 0	=	CANCEL UNDERLINE	
CONTROL .	4	=	ITALIC PRINTING	
CONTROL .	5	=	CANCEL ITALIC PRINTING	

These control functions handle almost every function of the GEMINI 10 printer. The remaining functions are handled in the page formatting sub-routine of the program. Program listing follows;

CONTINUED

REM

As the editor of this publication well knows, good writers intentionally put errors in their work so readers, upon discovering them, may feel superior.

There is a new book on programming out by Steve Davis entitled PROGRAMS FOR THE TI HOME COMPUTER. It is published by Steve Davis Publishing, PO Box 190831, Dallas, Texas 75219 for \$14.95. This book is well worth the money and contains all kinds of practical programs. The book is also sold by TENEX in South bend, Indiana. You will be pleasantly surprised with this manual.

GEMINI 10 LISTING

```

100 REM *TEX/SCRIBE*
110 OPTION BASE 1
120 DIM A$(200),C$(31)
130 LB=24 :: LC=66 :: LD=0 :: LE=0 :: LG=1
140 DATA NUL,SOH,STX,ETX,EOT,ENQ,ACK,BEL,BS,HT,LF,VT,FF,CR,
    SO,SI,DLE,DC1,DC2,DC3,DC4
150 DATA NAK,SYN,ETB,CAN,EM,SUB,ESC,FS,GS,RS,US
160 FOR I=1 TO 31
170 READ C$(I)
180 NEXT I
190 CR$=CHR$(13)
200 WIDTH=80
210 CALL CLEAR
220 CALL SCREEN(15)
230 MEM$="<<< SORRY, MEMORY FULL >>>"
240 REM --- MENU
250 CALL CLEAR
260 DISPLAY AT(3,1):" ***  TEX-SCRIBE  ***
    Command functions"
270 DISPLAY AT(7,1):"1-Delete 2-Insert 3-Replace
    a line a line a line"
280 DISPLAY AT(10,1):"4-Clear 5-Add 6-Save
    file lines file"
290 DISPLAY AT(13,1):"7-List 8-Load 9-Replace
    file file a string"
300 DISPLAY AT(16,1):"0-Print the text of file"
310 DISPLAY AT(19,1):"*** -1= END OF PROGRAM ***"
320 DISPLAY AT(22,1):"HOW MANY CHARACTERS PER
    LINE DO YOU WANT?";WIDTH
330 ACCEPT AT(23,19)VALIDATE(DIGIT)SIZE(-3):WIDTH
340 IF (WIDTH<1)OR(WIDTH>132)THEN 320
350 PRINT
360 ON ERROR 3000 :: CALL SCREEN(15):: INPUT "ACTION ? ":M
370 IF M=-1 THEN 3120 :: M=M+1
380 IF (M<1)+(M>10)=-1 THEN 390 ELSE 410

```

```

390 CALL CLEAR :: CALL SCREEN(12):: DISPLAY AT(1,1):"
    <<<< SELECTION ERROR >>>>"
400 GOTO 260
410 ON M GOSUB 2050,440,580,760,900,970,1090,1250,1500,1730
420 CALL SCREEN(4):: IF M=9 THEN CALL CLEAR :: GOTO 320
430 GOTO 350
440 REM DELETE A LINE
450 CALL SCREEN(10)
460 IF L>0 THEN 490
470 PRINT : " *** FILE EMPTY ***":
480 GOTO 570
490 GOSUB 2520
500 IF B=0 THEN 570
510 L=L-1
520 FOR I=B TO L
530 A$(I)=A$(I+1)
540 NEXT I
550 A$(L+1)=""
560 PRINT : "Deleted, file renumbered":
570 RETURN
580 REM INSERT A LINE
590 CALL SCREEN(10)
600 IF L>0 THEN 630
610 PRINT : " *** FILE EMPTY ***":
620 GOTO 750
630 PRINT : "Insert before"
640 GOSUB 2520
650 IF B=0 THEN 750
660 PRINT : "Enter new line-":
670 L_PROMPT$=STR$(B)
680 GOSUB 2390
690 L=L+1
700 FOR I=L TO B+1 STEP -1
710 A$(I)=A$(I-1)
720 NEXT I
730 A$(B)=INPUT$
740 PRINT : "Line added, file renumbered": :
750 RETURN
760 REM REPLACE A LINE
770 CALL SCREEN(10)
780 IF L>0 THEN 810
790 PRINT : " *** FILE EMPTY ***":
800 GOTO 890
810 PRINT : "Replace"
820 GOSUB 2520
830 IF B=0 THEN 890
840 PRINT : "Enter replacement line-":
850 L_PROMPT$=STR$(B)
860 GOSUB 2390
870 A$(B)=INPUT$
880 PRINT : "Line replaced": :
890 RETURN
900 REM CLEAR FILE IN MEMORY
910 FOR I=1 TO L
920 A$(I)=""
930 NEXT I
940 L=0
950 PRINT : : "--- MEMORY WORK AREA CLEARED": :
960 RETURN
970 REM ADD LINES TO FILE

```

CONTINUED

GEMINI 10 LISTING

```

980 CALL SCREEN(15)
990 CALL CLEAR
1000 PRINT : : "---- Ready for typing ---
      (enter ^^ to exit)": :
1010 L_PROMPT%=STR$(L+1)
1020 GOSUB 2390
1030 IF LEN(INPUT$)<>3 THEN 1050
1040 IF SEG$(INPUT$,1,2)="^^" THEN 1080
1050 L=L+1
1060 A$(L)=INPUT$
1070 GOTO 1010
1080 RETURN
1090 REM SAVE FILE
1100 CALL SCREEN(12)
1110 IF L>0 THEN 1140
1120 PRINT : : " *** FILE EMPTY ***": :
1130 GOTO 1240
1140 PRINT : : "Enter range of file to save.": :
1150 GOSUB 2640
1160 PRINT
1170 INPUT "SAVE TO                               i=cassette:":DEV
1180 IF DEV<>1 THEN 1210
1190 GOSUB 2910
1200 GOTO 1240
1210 PRINT
1220 CALL SOUND(500,220,1,659,1)
1230 GOTO 1170
1240 RETURN
1250 REM LIST FILE SOURCE
1260 CALL SCREEN(4)
1270 PRINT : "Enter the range of file
      source lines to print-":
1280 GOSUB 2640
1290 PRINT : :
1300 INPUT "Output to screen or printer?      (P/S)":P$
1310 PRINT : :
1320 IF (P$="P")+ (P$="p")=-1 THEN 1390
1330 FOR I=A TO B
1340 S=I
1350 GOSUB 2250
1360 PRINT I;S$;
1370 NEXT I
1380 GOTO 1490
1390 PRINT : : : " *** PRINTING SOURCE ***": : :
1400 OPEN #1:"PIO",OUTPUT
1410 PRINT #1:CHR$(15)
1420 FOR I=A TO B
1430 S=I
1440 GOSUB 2250
1450 PRINT #1:I;S$
1460 NEXT I
1470 PRINT #1:CHR$(18)
1480 CLOSE #1
1490 RETURN
1500 REM LOAD A FILE
1510 CALL SCREEN(12)
1520 IF L=0 THEN 1640
1530 CALL SOUND(500,220,1,659,1)
1540 PRINT : : : " *** FILE HAS DATA ALREADY IN IT.":
1550 PRINT : " Enter ""C"" to CLEAR file

```

Enter ""M"" to MERGE new file"

```

1560 IF (CH$="C")+ (CH$="c")=0 THEN 1600
1580 GOSUB 900
1590 GOTO 1640
1600 IF (CH$="M")+ (CH$="m")=-1 THEN 1640
1610 CALL SOUND(500,220,1,659,1)
1620 PRINT
1630 GOTO 1050
1640 PRINT
1650 INPUT "LOAD/MERGE from:                               i=cassette -?":DEV
1660 IF DEV<>1 THEN 1670
1670 GOSUB 2830
1680 GOTO 1720
1690 CALL SOUND(500,220,1,659,1)
1700 PRINT
1710 GOTO 1650
1720 RETURN
1730 REM REPLACE A STRING
1740 CALL SCREEN(10)
1750 PRINT : "String to be replaced"
1760 L_PROMPT%="?"
1770 GOSUB 2390
1780 R$=SEG$(INPUT$,1,(LEN(INPUT$)-2))
1790 D=LEN(R$)
1800 IF D>0 THEN 1830
1810 PRINT : "CAN'T REPLACE A NULL STRING": :
1820 RETURN
1830 N=0
1840 PRINT : "Enter replacement string"
1850 GOSUB 2390
1860 N$=SEG$(INPUT$,1,(LEN(INPUT$)-2))
1870 PRINT : "Enter search range."
1880 GOSUB 2640
1890 PRINT : : " *** SEARCHING ***": :
1900 FOR K=A TO B
1910 IF LEN(A$(K))<D THEN 2020
1920 CPOS=POS(A$(K),R$,1)
1930 IF CPOS=0 THEN 2020
1940 T$(1)=SEG$(A$(K),1,CPOS-1)
1950 T$(2)=SEG$(A$(K),CPOS+D,132)
1960 A$(K)=T$(1)&N$&T$(2)
1970 N=N+1
1980 PRINT "Changed line";K;"to-"
1990 S=K
2000 GOSUB 2250
2010 PRINT S$;
2020 NEXT K
2030 PRINT : : N;"changes made.": :
2040 RETURN
2050 REM PRINT TEXT FROM FILE
2060 CALL SCREEN(6)
2070 PRINT
2080 INPUT "PAGE PRINTING FORMAT Y/N?":AL$
2090 IF (AL$="Y")+ (AL$="y")=-1 THEN GOSUB 3130
2100 PRINT
2110 INPUT "NUMBER OF TIME TO PRINT TEXT":Z
2120 IF Z=0 THEN PRINT : : GOTO 360
2130 PRINT : "Enter line range to print-":
2140 GOSUB 2640

```

CONTINUED

GEMINI 10 LISTING

```
2160 PRINT "PRINTING TEXT ***":
```

```
2160 FOR J=1 TO Z
2170 OPEN #1:"PIO",OUTPUT
2180 PRINT #1:LB%;LC%;LD%;LE%;LF%
2190 FOR I=1 TO B
2200 PRINT #1:A$(I)
2210
2220 CLOSE #1
2230 NEXT J
2240 RETURN
2250 REM FORMAT SOURCE OUTPUT
2260 S#=A$(S)
2270 LS=LEN(S#)
2280 FOR O=1 TO LS
2290 P=ASC(SEG$(A$(S),O,1))+1
2300 IF P<33 THEN 2330
2310 IF P<130 THEN 2370
2320 P=P-128
2330 TS=LEN(S#)
2340 LM=LS-D
2350 PM=TS-LM
2360 S#=SEG$(S#,1,(PM-1))&C$(P)&SEG$(S#,(PM+1),LM)
2370 NEXT O
2380 NEXT I
2390 REM INPUT FROM KEYBOARD
2400 INPUT$=""
2410 LINPUT L_PROMPT$&"-":INPUT$
2420 IF LEN(INPUT$)<WIDTH THEN 2500
2430 PRINT : " *** LINE TO LONG ***":
2440 CALL SOUND(500,220,1,659,1)
2450 FOR W=WIDTH TO 1 STEP -1
2460 IF SEG$(INPUT$,W,1)=" " THEN 2480
2470 NEXT W
2480 INPUT$=SEG$(INPUT$,1,W)
2490 PRINT L_PROMPT$&"-":INPUT$
2500 INPUT$=INPUT$&CR$
2510 RETURN
2520 REM FIND AND DISPLAY LINE SUBROUTINE
2530 PRINT
2540 INPUT "Line number?":B
2550 IF (B>1)+(B<(L+1))=-2 THEN 2580
2560 PRINT : " *** NO SUCH LINE ***":
2570 GOTO 2540
2580 IF B=0 THEN 2630
2590 PRINT : "Old line";B;"reads -":
2600 S=B
2610 GOSUB 2260
2620 PRINT S#
2630 RETURN
2640 REM GET RANGE SUBROUTINE
2650 PRINT "The last line in file is";L:
2660 PRINT : "Enter the first line number,"
2670 INPUT "(Enter zero for whole file)":A
2680 IF A>0 THEN 2720
2690 A=1
2700 B=L
2710 GOTO 2820
2720 IF (A>0)+(A<(L+1))=-2 THEN 2750
2730 PRINT : " *** NUMBER OUT OF RANGE ***":
2740 GOTO 2660
2750 PRINT
```

```
2760 INPUT "Enter the last line number:":B
```

```
2770 IF B>A THEN 2800
2780 B=A
2800 IF B<L THEN 2820
2810 B=L
2820 REM CASSETTE LOAD SUBROUTINE
2830 OPEN #1:"CS1",INTERNAL,INPUT,FIXED 192
2840 INPUT #1:X::LL=L
2850 FOR I=LL+1 TO X+LL
2860 INPUT #1:A$(I)::L=L+1
2870 NEXT I
2880 CLOSE #1
2890 RETURN
2900 REM CASSETTE SAVE SUBROUTINE
2910 OPEN #1:"CS1",INTERNAL,OUTPUT,FIXED 192
2920 PRINT #1:((B+1)-A)
2930 FOR I=A TO B
2940 PRINT #1:A$(I)
2950 NEXT I
2960 CLOSE #1
2970 RETURN
2990 REM ERROR HANDLING AND RECOVERY SUBROUTINE
3000 ON ERROR 3060::CALL ERR(ECODE,ZAP)
3010 CALL SOUND(500,110,1,220,1,659,1)
3020 IF ECODE=39 OR ECODE=40 THEN 3030 ELSE 3040
3030 A$(L)=""::A$(L-1)=""::A$(L-2)=""::A$(L-3)=""
::A$(L-4)=""::L=L-4:
:PRINT::MEM$::GOTO 3080
3040 IF ECODE>82 AND ECODE<131 THEN PRINT : "<<< SORRY.
I/O ERROR >>>"::GOTO 3080
3050 PRINT : "<<< WEIRDD ERROR @#?! >>>"::GOTO 3080
3060 CALL ERR(ECODE,ZAP,ZIP,SPOT)
3070 PRINT "ERROR";ECODE;"IN LINE";SPOT
3080 ON ERROR 3100
3090 CLOSE #1::GOTO 3110
3100 CALL ERR(ECODE,ZAP)
3110 RETURN 350
3120 CALL CLEAR::GOTO 5000
3130 REM PRINT FORMAT INSTRUCTIONS
3140 CALL CLEAR::CALL SCREEN(3)
3150 DISPLAY AT(6,1):"LF SIZE ?/144:";LB
3160 DISPLAY AT(8,1):"PAGE LENGTH/LINES:";LC
3170 DISPLAY AT(10,1):"HEADER LENGTH/LINES:";LD:"1 TO 16"
3180 DISPLAY AT(12,1):"LH MARGIN SPACES:";LE
3190 DISPLAY AT(14,1):"1-PICA 2-ELITE 3-COND TYPE:
3200 ACCEPT AT(6,21)SIZE(-5)BEEP:LB
3210 ACCEPT AT(8,21)SIZE(-2)BEEP:LC
3220 ACCEPT AT(10,22)SIZE(-2)BEEP:LD
3230 ACCEPT AT(12,22)SIZE(-2)BEEP:LE
3240 ACCEPT AT(15,22)SIZE(-1)BEEP:LF
3250 LB%=CHR$(27)&CHR$(51)&CHR$(LB)
3260 LC%=CHR$(27)&CHR$(67)&CHR$(LC)
3270 LD%=CHR$(27)&CHR$(82)&CHR$(LD)
3280 LE%=CHR$(27)&CHR$(77)&CHR$(LE)
3290 LF%=CHR$(27)&CHR$(66)&CHR$(LF)
3300 CALL SCREEN(1)::RETURN
```

DO WE GET LETTERS!

by Conni

The following is a small list containing names of other User Groups presently corresponding with us. Although I love to get mail, I am asking for a volunteer to answer some of this mail, and keep us in touch with these other groups. We are starting a column of the best of other newsletters. RAISE YOUR HAND at the next meeting, this matter DEMANDS our attention. (WE know SOMEONE out there loves to write.

THE VALLEY 99'ER from PIONEER VALLEY U.G.

THE MSP (Minn/St. Paul) 99 U.G. (same name newsletter).

KENTUCKIANA 99/4 COMPUTER SOCIETY NEWSLETTER (Lou.Ky.)

DAYTONA 99ER'S (Same name group).

NATE'S NINETY-NINER (Nationwide Insurance group).

SAN GABRIEL VALLEY 99/4 USER'S GROUP (CALIF.)

CALL NEWSLETTER. (Atlanta 99/4A Users Group, Georgia).

BITS & BYTES (South bay TI Users Group, Las Gatos CA)

PERIPHERAL (Pittsburg Users Group, (PUG)).

JUG (Johnson Space Center (JSC) User Group).

THE FAMILY PROGRAMMER (Formerly TI Source & Logo News, Armonk NY).

TIC TALK (Rocky mountain 79ers, Littleton CO).

EDMONTON USER GROUP

(Edmoton, Alberta Canada)

99/4 USERS OF AMERICA (Flint MI).

99/4 SGV U.G. PRINTOUT (San Gabriel Valley) W Covina CA)

TI994/A U.G. (Baltimore MD).

P.N.T.I.U.G. (Eugene OR).

CMIX (Columbus MicroComputer Information Exchange)

ACSCO (Amature Computer Society of Central Ohio). **WHEW!!**

O.K. Let's hear from you writers Out there. (C)

TREASURERS REPORT

New Members @ \$15.00 = \$405.00

Newsletter expenses -\$226.05

Net Increase since last report

\$178.95

Previous Ballance \$83.00

Bal. as of June 25 \$261.95

Income from Cassett tape \$255.00

Unpaid Expenses (Thompson Co.)

\$193.00

Net proceeds \$62.00

TOTAL ASSETS as of June 25

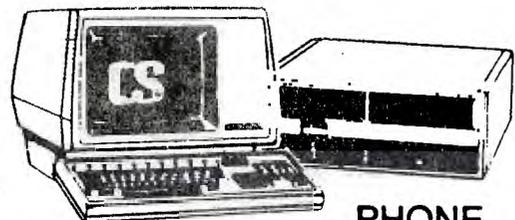
\$323.95

Art Morgan

COMPUTERS

Central Ohio's authorized Texas Instruments Full-service Dealer.

- Professional computers • Printers
- Business Systems • Software
- Terminals • Warranty service • Home computers



PHONE
888-9287

CS **COMMANDER
SYSTEMS, INC.**

F.L.A.G.S. Center-Schrock Rd. at Cleveland Ave.
Just north of I-270 exit

**AUTHORIZED
DEALER**

**TEXAS INSTRUMENTS
COMPUTER SYSTEMS**

BOOK REPORT

Several new books are turning up on the shelves of book stores around the country. Two new ones are

USING & PROGRAMMING THE TI994/A, & TI994/A GAME PROGRAMS. by Fredrick Holtz Publishers list \$34.90 (2 Vol set). Tab book Club Price \$19.95 (2 Vol. set). (I just happen to be a member of that one).

JUG'S Newsletter carried a report on B. Dalton Bookseller, (ours is located in Eastland Mall) carrying a large number of books... NOT JUST FOR KIDS by E CARLSON-RESTON Publishing Co./ PROGRAMS FOR THE TI994/A HOME COMPUTER by STEVE DAVIS / 36 TEXAS INSTRUMENTS TI994/A PROGRAMS FOR HOME, SCHOOL & OFFICE by LEN TURNER / 101 PROGRAMS & TIPS & TRICKS FOR THE TI994/A HOME COMPUTER by LEN TURNER.

Tom Fosson has 3 books he would like to sell (Childrens Books), I wrote them down & misplaced the titles. But, you can catch him at the July meeting. In any case I'm still checking and will let you know whenever I find something interesting to report. happy book hunting,...(B)



IS 2 BETTER THAN 1

OR IS MORE LESS

A short time ago TI introduced a Peripheral Expansion Box, (P-Box). This was to take the place of all the peripherals daisy-chained out to the side four feet, & possibly give you a place to put your Monitor. But that's not what I'm here to talk about.

Next to all those slots in the P-Box, all the way to the right, (if you are looking at the front of the box), is a space that is bigger than the rest. This is where the T.I.-supplied DISK drive gets installed. Notice I said "DISK DRIVE" (singular). Yes I'm going to address the subject of "SKINNY DISKS" (Plural), or Half-Height Disk drives.

Now that there has been a special price on the P-box, (FREE) many more of you could afford to own one. So you hurried out and bought an "RS232", "disk controller card", and one other item from THE LIST. Why didn't you buy the "32K RAM or the "DISK DRIVE"? Because by nature man is always trying to get a FREE LUNCH...So...You thought You would send away for the "128K" memory, and the "SKINNY DISKS". Good thinking, More memory than you need, and the equivalent to 4 Disk drives. (half height/ double sided). Of course getting all of this equipment to work is another matter...THAT'S what I'm here to talk about. (We'll talk about 128K RAM some other time.

First let me say, I don't think T.I. EVER meant it's P-box to house more than one Disk drive, (especially another brand). Where did I get this ridiculous idea? Look at figure # 1, Notice four 1N4002 Rectifiers used to supply 16VDC (VOLTS D.C.) UNregulated to the cards and the 7812 Voltage regulator that in turn supplies 12VDC to the Disk drive. This particular rectifier has a rating of 1AMP continuous, or 2AMPS @ 50% duty). The 7812 V.R. has a rating of 1.5 AMPS. The TI standard-Supplied Disk drive, (MP1B51) draws 1.5 AMPS Power up surge, and 0.8 AMPS running. The BEST Half-Height Disk Drives draw 0.7 AMPS running...EACH...Both drives start and run Simultaneously on power up.

(drives do stop running if there are no disks in them). This is part of the DSR, (Device Service Routine). OK. back to the subject at hand. All of this "Draw of amperage from the power supply leaves you with little if any power for the other peripherals in the P-Box! What WILL you do?

The simple solution is to buy TI brand or TI recommended Disk drives, but that's not what you want to hear.

Another solution is to replace the 1N4002 with a 1N5046 which has a 3 AMP rating, and the 7812 voltage regulator with a 78H12 which has a higher rating. You

may still have to leave two slots empty in your P-box to prevent overloading the power transformer.

Whether you burn up your P-box by overloading it or have a professional make the changes for you, TI will certainly NOT honor the warranty. CAUTION: Electronics are expensive as well as dangerous even at low voltages. Have a professional make these changes for you.

And now back to the "SKINNY DISKS", QUME & SHUGART are two of the brands in question. Although these companies manufacture an excellent product, I have't come accross any information regarding thier compatabiliy with the P-box. They however do require the same amount of power each as a full size disk drive. this would seemingly make them incompatable.

Pete Crowell of Rocky Moutian 99ers says, "he is successfully using TEAC mod (#55B) without any

side effect". He also mentioned in his article that TEAC may have an availabilty problem until next year.

The only other brand mentioned as compatable is Hitachi (Same power requirements as Teac). The two Major differences being, they are Beige instead of Black, (I could live with that), and they are Double Sided Double Density. The later does not matter to the TI controller, which will only handle Single Density on both sides.

IS TWO BETTER THAN ONE? Add them up, 1 Double Sided Disk Drive + 1 Double Sided Disk Drive= 4 Disk Drives.

This article in part previously appeared in the Pittsburg Peripheral may 83, & the Rocky Moutian 99ers newsletter June 83, (Pete Crowell). TEAC, QUME, SUGART, HITACHI, & TI ARE REGISTERED TRADE MARKS OF THIER RESPECTIVE COMPANYS.

DRIVE CAREFULLY...Pat S.



FIGURE 1

