HOCUS

Home Computer Users Spotlight

a monthly publication of the Milwaukee Area 99/4 Users Group

APRIL—1988

MILWAUKEE AREA USER GROUP
4122 GLENWAY WAUWATOSA WI 53222

President...D.Walden	5292173
Vice-Pres...J.Schroeder	2644735
Treasurer...P.Norton	4628954
Secretary...B.Kling	5295161
Librarian...E.VonDerEhe	5490593
Librarian...F.Pabian	3273618
Newsletter..G.Hitz	5350133
S.I.G.....Schroeder/Walden/Hitz	

Next Group Meeting - 2nd Saturday
May 14,1988 - 12 n..' til 4 PM
Wauwatosa S & L - [illegible] West State

North Sub-Meeting - 1st Tuesday
May 3, 1988 - 7 PM til 10 PM
Security S & L - 5555 N Pt Washington

South Sub-Meeting - 3rd Tuesday
April 19, 1988 - 7 PM til 10 PM
Franklin State Bank - 7000 So 76th

Membership Dues $10 - Family $15

<<<<< HOCUS NEWSLETTER INDEX >>>>>

Use TI-ARTIST Instances with Certificate99

First copy Certificate99 Disk2 Use DM1000 to change names CDATA21,CDATA22 TO CDATA21_P & CDATA22_P. Load TI-ARTIST. Now follow these steps: 1) TI-ARTIST, S)tore, L)load, filename CDATA21, D)draw, draw lines between pictures, Z)zoom, move outline over picture, FB zoom in, Ctrl<A> erase, repeat all 12, <space bar>, Fctn(=), 2) Enhancement, 6)load instance, move instance to space, FB drops picture in, repeat for all 12 instances, Fctn(=), 1) TI-ARTIST, D)draw, Ctrl(.) erase lines, S)tore, S)save, CDATA21, Now you may repeat the entire sequence for CDATA22. Use DM1000 to change names from CDATA21_P,CDATA22_P back to CDATA21 and CDATA22 and you're all set to use them in your Certificate99 program.

TI 32K MEMORY EXPANSION TEST PROGRAM AND LOCATOR BY JOE NUVOLINI AND JOHN WILLFORTH

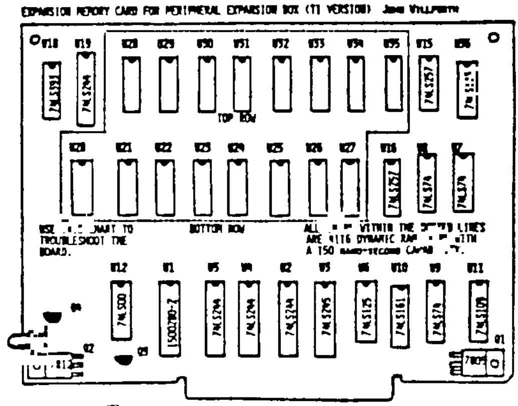
PROBABLY TWO YEARS OR MORE AGO JOE NUVOLINI WROTE AN EXTENDED BASIC PROGRAM TO USE IN TROUBLESHOOTING MEMORY PROBLEMS WITH YOUR TI MEMORY EXPANSION CARD (IN PEB). THE PROGRAM FUNCTIONED QUITE WELL IN ABOUT 80% OF THE MEMORY FAILURES RELATED TO THE 4116 CHIPS THEM[illegible]ES. BUT WHEN YOU HAD A STUCK "[illegible]" BIT, THE PROGRAM TOLD YOU THAT THE GOOD CHIPS WERE BAD AND THE BAD CHIP WAS OK. WELL I DID NOT CORRECT THAT DEFFICIENCY IN THE PROGRAM, BUT I WILL ISSUE A WARNING TO THE EFFECT THAT ON A PARTICULAR ROW BEING TESTED, IF YOU ARE TOLD THAT ALL ARE BAD BUT ONE, ASSUME THAT THAT ONE IS THE BAD CHIP. IF YOU WOULD LIKE TO KNOW WHY SEND A SASE, AND I WILL TELL YOU WHY THIS TRUE.

```
100 REM   WRITTEN BY JOE N[illegible]
OLINI (303) 596-6938, MO[illegible]
ED BY JOHN WILLFORTH
110 N=0
120 CALL CL[illegible]
[illegible] CALL SC[illegible](13)
[illegible]40 PRINT [illegible] MEMORY EXPANSIO
[illegible]ER        FOR THE MINI
ME[illegible]::::
150 PRINT "  SINCE PROGRAMS
[illegible] IN XB LOAD I
N[illegible]Y, [illegible]
PRO[illegible] SHOULD BE [illegible] IN I
F YOU [illegible] OWN"
160 PR[illegible] "A MINIMEMORY UNIT
.":
170 PRINT "  TO [illegible] THE INFO
[illegible]ON    PROV[illegible] BY THIS
[illegible]NTYOUR [illegible]RY E
[illegible] CARD [illegible] TWO
[illegible] 4116   CHIPS AT THE
[illegible]
180 PRINT :"ENTER :":"
1 TO [illegible] ROW ":"
2 TO [illegible]OM ROW":"
     3 TO [illegible]"
190 CALL KEY(0,K,S)
200 IF S=0 THEN 190
[illegible] IF K<49 [illegible] 190
[illegible] IF K>51 THEN 190
[illegible] R=K-48
240 IF R=1 THEN 700
250 IF R=2 THEN 720
260 IF R=3 THEN 680
270 IF R=1 THEN 300
[illegible] N=27
[illegible] GOTO 310
[illegible] N=35
[illegible] V=N
320 CALL CLEAR
330 IF R=1 THEN 350
340 GOTO 370
350 PRINT "TEST OF TOP ROW O
F 4116'S"
360 [illegible] 380
370 [illegible] :"TEST OF BOTTOM R
OW OF 4116'S"
380 PRINT :"Reading from rig
ht to left..":: 
390 FOR T=1 TO 2
400 FOR I=0 TO 7
410 IF T=1 THEN 440
420 IN=2^I
430 GOTO 450
440 IN=0
450 CALL L[illegible] A,IN)
460 CALL [illegible] A,D)
470 IF IN=[illegible] THEN 510
480 PRINT "        Chip U";STR
$(N);" is BAD"
4[illegible] [illegible]EEN(10)
[illegible]
5[illegible]               Chip U";STR
$(N);" is OK"
520 N=N-1
5[illegible] PRINT "Written = ";IN;"
[illegible]d = ";D
[illegible] I
[illegible] 740
[illegible] CALL CL[illegible]
[illegible] IF T=1 [illegible] 610
[illegible] PRINT " END OF SECOND PA
[illegible]::::::::
590 [illegible] 740
600 [illegible] 540
610 PRINT " END OF FIRST PAS
S"::::::::
620 GOSUB 740
630 N=V
640 N[illegible] T
650 [illegible]
[illegible] "PRESS ENTER TO CO
N[illegible]UE ":X$
670 GOTO 100
[illegible] CALL CLEAR
[illegible] END
700 A=-12[illegible]
710 GOTO [illegible]
720 A=12287
730 GOTO 270
740 FOR DELAY=1 TO 600
750 N[illegible] DELAY
760 RETURN
```

THIS PROGRAM WILL ONLY CHECK FOR THE COLUMN AND ROW TYPE FAILURES IN A CHIP (THE HIGHEST PERCENTAGE OF FAILURES), AND IS VERY RELIABLE IN TELLING YOU THE FAILING CHIP BY "UXX", WHICH CAN BE REFERENCED AGAINST THE CHART BELOW. A MUCH MORE EXTENSIVE PROGRAM WOULD HAVE TO BE WRITTEN TO CHECK EACH "BIT", AND THE SLOWNESS WOULD DISCOURAGE YOU IN USE OF IT. THIS PROGRAM SHOULD SAVE YOU MORE THAN $30. BY POINTING TO A 75¢ CHIP. YOU WILL ALSO NOTE THAT IT IS NOT IN EXTENDED BASIC ANY LONGER BUT IN BASIC FOR THE MINI-MEMORY. THIS MAKES IT POSSIBLE TO LOAD DIRECTLY FROM CASS. OR DISK AND RUN. IF YOU HAD A DEFECTIVE [illegible] MEMORY AND HAD THIS PROGRAM ON CASS. OR DISK, YOU WOULD NOT BE ABLE TO LOAD AND RUN IT, BECAUSE IT WOULD HAVE LOADED INTO THE EXPANSION MEMORY (WHICH IS BROKEN). BEING IN BASIC, YOU HAVE TWO OPTIONS. IF YOU WANT TO USE EXTEND BASIC, YOU'LL HAVE TO : "CALL INIT" AND "CALL LOAD(-31868,0,0) TO TURN "OFF" THE EXPANSION MEMORY FIRST, THEN LOAD THE PROGRAM ON THE LEFT INTO YOUR MACHINE FROM CASS. OR DISK. EVEN THOUGH YOU TURNED IT OFF, THE MACHINE CAN TEST IT! NEAT!

I HOPE THAT THIS IS JUST WHAT YOU WERE LOOKING FOR TO GET YOUR OLD 32K RUNNING WELL AGAIN. THERE ARE COMPONENTS THAT CAN FAIL, MAYBE NEXT MONTH I'LL GIVE A LITTLE MORE INSIGHT ON TROUBLESHOOTING THE OTHER PROBLEMS. JFW

EXPANSION MEMORY CARD FOR PERIPHERAL EXPANSION BOX (TI VERSION) John Willforth



HOW TO REPAIR AN ELECTRONIC INSTRUMENT

STEP 1 - APPROACH THE AILING INSTRUMENT IN A CONFIDENT MANNER. THIS WILL GIVE THE INSTRUMENT THE MISTAKEN IDEA THAT YOU KNOW SOMETHING AND THAT YOU ARE NOT AFRAID OF IT. IT WILL ALSO IMPRESS ANYONE ELSE WHO HAPPENS TO BE LOOKING, AND IF THE INSTRUMENT SUDDENLY STARTS WORKING AGAIN, YOU WILL BE CREDITED WITH THE REPAIR.

STEP 2 - WAVE THE SERVICE MANUAL AT THE INSTRUMENT. THIS WILL MAKE IT ASSUME THAT YOU ARE AT LEAST FAMILIAR WITH THE SOURCE OF ALL KNOWLEDGE, AND STARTS THE INSTRUMENT TO THINKING THAT THERE IS EVEN THE SLIGHT POSSIBILTY THAT YOU CAN READ.

STEP 3 - IN A FORCEFUL AND DIRECT MANNER , RECITE OHMS LAW OR SOMETHING EQUALLY TECHNICAL - SOUNDING TO THE INSTRUMENT. (CAUTION: BEFORE TAKING THIS STEP, BE SURE TO CONSULT A RELIABLE SOURCE FOR THE CORRECT PRONUNCIATION OF OHMS LAW AND OTHER TECHNICAL WORDS.) THIS WILL INTIMIDATE THE INSTRUMENT AND PROVE THAT YOU INDEED DO KNOW SOMETHING. IF THIS PRODUCES NO IMMEDIATE REACTION, PROCEED TO STEP 4.

STEP 4 - JAR THE INSTRUMENT. THIS IS A PROGRESSIVE PROCEEDURE, STARTING WITH BOUNCING THE INSTRUMENT LIGHTLY ON THE BENCH, AND CULMINATING WITH DROPPING THE INSTRUMENT FROM A HEIGHT OF THREE TO SEVEN FEET (HIGHER IF THE INSTRUMENT IS PARTICULARLY FRAGILE). CAUTION MUST BE EXERCIZED HOWEVER; ALTHOUGH THE DROP METHOD IS A LONG-STANDING RECOGNIZED TECNIQUE OF INSTRUMENT REPAIR, ONE MUST BE CAREFUL NOT TO MAR THE FLOOR, OR THE CUSTODIAL STAFF WILL GET REALLY TICKED OFF AT YOU, IN WHICH CASE YOU ARE IN BIG TROUBLE.

STEP 5 - BRANDISH A LARGE SCREWDRIVER IN A MENACING MANNER. THIS WILL BADLY FRIGHTEN THE INSTRUMENT AND DEMONSTRATE YOUR INTIMATE KNOWLEDGE OF THE DEADLY SHORT CIRCUIT TECHNIQUE. TAP THE INSTRUMENT LIGHTLY WITH THE POINT OF THE SCREWDRIVER FOR SEVERAL SECONDS, JUST TO LET IT KNOW WHAT <u>COULD</u> HAPPEN IF IT FAILS TO WISE UP AND START WORKING.
IF THIS STILL FAILS TO ELICITE RESPONSE, PROCEED TO STEP 6.

STEP 6 - USING THE SCREWDRIVER, PRY THE BACK OFF THE INSTRUMENT (EVEN IF IT WAS DESIGNED TO OPEN FROM THE FRONT) AND EXPOSE THE INNARDS. CHOOSE A RANDOM LOCATION INSIDE AND STICK IN A TUBE - EVEN IF THE INSTRUMENT IS TOTALLY SOLID STATE. THIS WILL ACCOMPLISH TWO THINGS: IT WILL PROVE TO THOSE STANDING ABOUT WATCHING THAT YOU ARE INDEED INTIMATELY FAMILIAR WITH THE DESIGN, AND WILL ALSO CONFUSE THE INSTRUMENT GREATLY, THEREBY INCREASING YOUR PSYCHOLOGICAL ADVANTAGE.

STEP 7 - MAKE LOUD DISPARAGING REMARKS ABOUT THE DESIGNER OF THE INSTRUMENT, THE POOR QUALITY OF THE COMPONENTS, AND THE SLIPSHOD MANNER WITH WHICH IT WAS ASSEMBLED. USE LOTS OF EXPLETIVES. THIS MAY SERVE TO MAKE THE INSTRUMENT FEEL SUFFICIENTLY GUILTY TO START WORKING AGAIN, OR GET IT SO ANGRY AT YOU THAT IT STARTS TO WORK JUST TO SPITE YOU. BE SURE TO KEEP A FINGER ON THE INSTRUMENT AT ALL TIMES, SO THAT IF IT DOES START TO WORK, YOU WILL GET THE CREDIT, RATHER THAN LOOK SILLY.

STEP 8 - IF ALL ELSE FAILS, MAKE VARIOUS COMMENTS ON HOW YOUR TIME IS MUCH TO VALUABLE TO WASTE ON THIS STUPID THING AND WALK AWAY, HOPING TO SNEAK OUT OF THE BUILDING BEFORE ANYONE ELSE SPOTS YOU.

PRACTICAL PROGRAMING PRACTICES

This Program will allow you to Load, Save, Edit, and print 4 inch by 6 inch postcards. To run this program you need the Extended Basic module, RS232 interface, and a printer. The program will also allow you to use grophics on your postcards provided that you are using the original TI Printer or one compatable with it such as the Gemini 10, SG-10, or a newer model. Give it a try. You may also change the print, line spacing, and darkness of the printer with this program. When editing a card, keep in mind that this program acts like TI-Writer in that you never see the entire postcard at once but toggle between two sides. Enjoy.

```
100 !+++++++++++++++!
110 !   "POSTCARD"   !
120 ! BY JOHN BEHNKE !
130 !REQUIRES X-BASIC!
140 !PRINTS 4x6 CARDS!
150 !   MARCH 1987   !
160 !+++++++++++++++!
170 DATA 235,231,236,233,234
,237,232,238,167,164,165,166
,168,174,173,175,190,239
180 AA=1 :: Z$="YNyn"
190 CALL CLEAR :: CALL SCREE
N(5):: FOR I=0 TO 14 :: CALL
 COLOR(I,16,1):: NEXT I
200 ON ERROR 1900
210 DISPLAY AT(6,11):"POSTCA
RD": :"        BY JOHN BEHNKE
"
220 DISPLAY AT(10,1):"
   MARCH 1987"
230 DIM A$(22),B$(22),C$(22)
,D$(18),E$(18),F$(5),G$(127)
240 F$(1)="DIS/FIX" :: F$(2)
="DIS/VAR" :: F$(3)="INT/FIX
" :: F$(4)="INT/VAR"
250 F$(5)="PROGRAM   " :: GO
TO 300
260 A,B,C,D,E,F,G,H,I,J,K,L,
M,Z=0 :: H$,I$,J$,K$,L$,M$,N
$,O$,P$=""
270 CALL DELSPRITE :: CALL C
HAR :: CALL HCHAR :: CALL KE
Y :: CALL VCHAR
280 CALL SPRITE :: CALL LOCA
TE
290 !@P-
300 CALL CHAR(126,"FFFFFFFFFF
0F0F0F0FFFFFFFF00000000FFFFF
FFF0F0F0F0F")
310 CALL CHAR(129,"F0F0F0F0F
0F0F0F")
320 CALL CHAR(130,"0F0F0F0F0
F0F0F0FF0F0F0F0FFFFFFFF00000
000FFFFFFFF")
330 CALL CHAR(133,"0F0F0F0FF
FFFFFFF")
340 H$=CHR$(27):: D$(1)=H$&"
B"&CHR$(4):: D$(2)=H$&"B"&CH
R$(5)
350 O$(3)=H$&"B"&CHR$(1):: D
$(4)=H$&"B"&CHR$(2):: D$(5)=
H$&"B"&CHR$(3)
360 O$(6)=H$&"W"&CHR$(1):: D
$(7)=H$&"W"&CHR$(0):: D$(8)=
H$&"G" :: O$(9)=H$&"H"
370 D$(10)=H$&"E" :: D$(11)=
H$&"F" :: D$(12)=H$&"2" :: D
$(13)=H$&"1"
380 O$(14)=H$&"O" :: D$(15)=
H$&"U"&CHR$(1):: D$(16)=H$&"
U"&CHR$(0)
390 FOR A=0 TO 17 :: READ B
:: E$(A)=CHR$(B):: NEXT A
400 CALL CHAR(138,"708888887
0000000207OF870200000002050S08
85020000000F8888888F8")
410 CALL CHAR(134,"0004023F0
20400001038541010100000000
808082A1C0800204OFC402")
420 CALL CHAR(142,"0088S020S
0880000007E7E7E7E7E7E"):: GO
TO 840
430 DISPLAY AT(1,1):"CTRL 9
to Abort" :: D$(16)=H$&"U"&C
HR$(0)
440 CALL HCHAR(2,3,127,28)::
 CALL HCHAR(24,3,132,28):: G
OSUB 740
450 FOR A=1 TO 21 :: GOSUB 7
40 :: ACCEPT AT(A+2,1)BEEP S
IZE(-28):A$(A)
460 IF SEG$(A$(A),1,1)=CHR$(
```

```
159)THEN 500
470 GOSUB 790 :: ACCEPT AT(A
+2,1)BEEP SIZE(-28):B$(A)
480 IF SEG$(B$(A),1,1)=CHR$(
159)THEN 500
490 NEXT A
500 DISPLAY AT(1,19):"Done?
N" :: ACCEPT AT(1,25)VALIDAT
E(Z$)BEEP SIZE(-1):I$
510 IF SEG$(A$(A),1,1)=CHR$(
159)THEN A$(A)=""
520 IF SEG$(B$(A),1,1)=CHR$(
159)THEN B$(A)=""
530 IF I$="N" OR I$="n" THEN
 440
540 GOTO 840
550 CALL CLEAR :: DISPLAY AT
(1,1):"PRINTER NAME: PIO"
560 ACCEPT AT(1,15)BEEP SIZE
(-14):J$ :: DISPLAY AT(1,1):
"HOW MANY: 1"
570 ACCEPT AT(1,11)VALIDATE(
DIGIT)BEEP SIZE(-4):C
580 CALL HCHAR(2,3,127,28)::
 CALL HCHAR(24,3,132,28):: G
OSUB 740
590 DISPLAY AT(1,18):"SCANNI
NG..." :: CALL SPRITE(#1,134
,16,16,9)
600 FOR A=1 TO 21 :: Z=LEN(B
$(A))
610 IF Z>0 THEN C$(A)=A$(A)&
RPT$(" ",28-LEN(A$(A)))&B$(A
)ELSE C$(A)=A$(A)
620 CALL LOCATE(#1,9+A*8,8):
: IF LEN(C$(A))=0 THEN 670
630 FOR D=126 TO 143 :: E=0
:: Z=LEN(C$(A)):: F=POS(C$(A
),CHR$(D),E+1)
640 IF F=0 THEN 660 ELSE C$(
A)=SEG$(C$(A),1,F-1)&E$(D-12
6)&SEG$(C$(A),F+1,Z-F)
650 IF F<LEN(C$(A))THEN E=F
:: GOTO 640
660 NEXT D
670 NEXT A :: OPEN #1:J$ ::
PRINT #1:H$&"C"&CHR$(0)&CHR$
(4)
680 DISPLAY AT(1,18):"A TO A
BORT" :: FOR A=C TO 1 STEP -
1
690 DISPLAY AT(1,10)SIZE(5):
A :: PRINT #1: : :: FOR G=1
TO 21 :: CALL KEY(0,D,H)
700 IF D=65 OR D=97 THEN 730
710 CALL LOCATE(#1,9+G*8,8)
720 PRINT #1:C$(G):: NEXT G
:: PRINT #1:CHR$(12):: NEXT
A
730 CLOSE #1 :: CALL DELSPRI
TE(#1):: GOTO 840
740 CALL VCHAR(2,31,32,23)::
 CALL HCHAR(2,2,126):: CALL
VCHAR(3,2,129,21)
750 CALL HCHAR(24,2,131):: D
ISPLAY AT(1,19):"LEFT SIDE"
760 DISPLAY AT(3,1):A$(1):A$
(2):A$(3):A$(4):A$(5):A$(6):
A$(7):A$(8):A$(9):A$(10)
770 DISPLAY AT(13,1):A$(11):
A$(12):A$(13):A$(14):A$(15):
A$(16):A$(17):A$(18)
780 DISPLAY AT(21,1):A$(19):
A$(20):A$(21):: RETURN
790 DISPLAY AT(3,1):B$(1):B$
(2):B$(3):B$(4):B$(5):B$(6):
B$(7):B$(8):B$(9):B$(10)
800 DISPLAY AT(13,1):B$(11):
B$(12):B$(13):B$(14):B$(15):
B$(16):B$(17)
810 DISPLAY AT(20,1):B$(18):
B$(19):B$(20):B$(21)
820 CALL VCHAR(2,2,32,23)::
CALL HCHAR(2,31,128):: CALL
VCHAR(3,31,130,21)
830 CALL HCHAR(24,31,133)::
DISPLAY AT(1,19):"RIGHT SIDE
" :: RETURN
840 CALL CLEAR
850 CALL HCHAR(1,12,126):: C
ALL HCHAR(1,13,127,8):: CALL
 HCHAR(1,21,128)
860 DISPLAY AT(2,10):CHR$(12
9)&"POSTCARD"&CHR$(130)
870 CALL HCHAR(3,3,132,28)::
 CALL HCHAR(3,12,131):: CALL
 HCHAR(3,21,133)
880 CALL HCHAR(3,2,132):: CA
LL HCHAR(3,31,132)
890 CALL VCHAR(4,2,129,20) :
 CALL VCHAR(4,31,130,20):: C
ALL HCHAR(23,2,127,30)
900 DISPLAY AT(5,6):"1 - LOA
D A CARD": :"      2 - SAVE A
 CARD"
910 DISPLAY AT(9,6):"3 - EDI
T A CARD": :"      4 - PRINT
A CARD"
920 DISPLAY AT(13,6):"5 - PR
INTER MODES": :"      6 - CLE
AR MEMORY"
930 DISPLAY AT(17,6):"7 - CA
TALOG A DISK": :"      8 - DI
SPLAY COLORS"
940 DISPLAY AT(21,6):"9 - EX
IT PROGRAM"
950 DISPLAY AT(24,3):"Use Ar
row Keys To Select"
960 CALL HCHAR(3+AA*2,7,[illegible])
970 CALL KEY(0,H,D):: IF D=0
```

```
 THEN 970
900 IF H=88 OR H=120 OR H=69
 OR H=101 THEN CALL HCHAR(3+
AA*2,7,32)
910 IF H=88 OR H=120 THEN AA
=AA+1
1000 IF H=69 OR H=101 THEN A
A=AA-1
1010 IF AA=0 THEN AA=9
1020 IF AA=10 THEN AA=1
1030 IF H<>13 THEN 960
1040 ON AA GOTO 1050,1050,13
10,550,1100,1270,1440,1830,1
780
1050 DISPLAY AT(24,1):"FILEN
AME: DSK1.CARD" :: ACCEPT AT
(24,14)BEEP SIZE(-15):K$
1060 K$="DSK"&K$ :: OPEN #1:
K$ :: FOR A=1 TO 21
1070 IF AA=1 THEN LINPUT #1:
L$ :: A$(A)=SEG$(L$,1,28)::
B$(A)=SEG$(L$,29,28)
1080 IF AA=2 THEN PRINT #1:A
$(A);B$(A)
1090 NEXT A :: CLOSE #1 :: G
OTO 840
1100 DISPLAY AT(1,8)ERASE AL
L:"PRINTER MODES": :"PRINTER
 NAME: PIO"
1110 ACCEPT AT(3,15)BEEP SIZ
E(-14):J$
1120 DISPLAY AT(5,1):"<N> NL
Q ON":"<N> NLQ OFF":"<N> PIC
A MODE":"<N> ELITE MODE"
1130 DISPLAY AT(9,1):"<N> CO
NDENSED MODE":"<N> EXPANDED
MODE ON"
1140 DISPLAY AT(11,1):"<N> E
XPANDED MODE OFF":"<N> DOUBL
E STRIKE ON"
1150 DISPLAY AT(13,1):"<N> D
OUBLE STRIKE OFF"
1160 DISPLAY AT(14,1):"<N> E
MPHASIZED ON":"<N> EMPHASIZE
D OFF"
1170 DISPLAY AT(16,1):"<N> 1
/6  INCH LINE SPACING  <N> 7
/72 INCH LINE SPACING"
1180 DISPLAY AT(18,1):"<N> 1
/8  INCH LINE SPACING  <N> U
NI-DIRECTION PRINT ON"
1190 DISPLAY AT(20,1):"<N> U
NI-DIRECTION PRINT OFF <N> E
NTER OTHER LINE SPACING"
1200 FOR A=5 TO 20 :: ACCEPT
 AT(A,2)VALIDATE(2$)BEEP SIZ
E(-1):I$
1210 IF I$="Y" OR I$="y" THE
N OPEN #1:J$ :: PRINT #1:D$(
A-4);:: CLOSE #1
1220 NEXT A :: ACCEPT AT(21,
2)VALIDATE(2$)BEEP SIZE(-1):
I$
1230 IF I$="N" OR I$="n" THE
N 840
1240 DISPLAY AT(21,1):"NEW L
INE SPACING -12/72 inch"
1250 ACCEPT AT(21,19)VALIDAT
E(DIGIT)BEEP SIZE(-2):I :: O
PEN #1:J$
1260 PRINT #1:K$&"A"&CHR$(I)
:: CLOSE #1 :: GOTO 840
1270 FOR A=1 TO 21 :: A$(A),
B$(A)="" :: NEXT A :: GOTO 8
40
1280 DISPLAY AT(24,1):"
ARE YOU SURE?  N"
1290 ACCEPT AT(24,21)VALIDAT
E(2$)BEEP SIZE(-1):I$ :: IF
I$="N" OR I$="n" THEN 84
0
1300 CALL CLEAR :: END
1310 DISPLAY AT(1,7)ERASE AL
L:"SPECIAL GRAPHICS"
1320 DISPLAY AT(4,3):CHR$(12
6)&" - FCTN W     "&CHR$(127)
&" - FCTN V"
1330 DISPLAY AT(6,3):CHR$(12
8)&" - CTRL ,     "&CHR$(129)
&" - CTRL A"
1340 DISPLAY AT(8,3):CHR$(13
0)&" - CTRL B     "&CHR$(131)
&" - CTRL C"
1350 DISPLAY AT(10,3):CHR$(1
32)&" - CTRL D    "&CHR$(133
)&" - CTRL E"
1360 DISPLAY AT(12,3):CHR$(1
34)&" - CTRL F    "&CHR$(135
)&" - CTRL G"
1370 DISPLAY AT(14,3):CHR$(1
36)&" - CTRL H    "&CHR$(137
)&" - CTRL I"
1380 DISPLAY AT(16,3):CHR$(1
38)&" - CTRL J    "&CHR$(139
)&" - CTRL K"
1390 DISPLAY AT(18,3):CHR$(1
40)&" - CTRL L    "&CHR$(141
)&" - CTRL M"
1400 DISPLAY AT(20,3):CHR$(1
42)&" - CTRL N    "&CHR$(143
)&" - CTRL O"
1410 DISPLAY AT(24,4):"PRESS
 ANY KEY TO EDIT."
1420 CALL KEY(0,0,H):: IF H=
0 THEN 1420
1430 GOTO 430
1440 DISPLAY AT(1,9)ERASE AL
L:"CATALOG DISK" :: CALL HCH
AR(2,1,132,32)
1450 DISPLAY AT(5,1):"DRIVE?
```

```
 (0-5): <1>"
1460 ACCEPT AT(5,16)VALIDATE
("012345")BEEP SIZE(-1):M$
1470 IF M$<>"0" THEN 1490 EL
SE DISPLAY AT(5,1):"DEVICE N
AME? <                >"
1480 ACCEPT AT(5,15)BEEP SIZ
E(-13):M$
1490 M$="DSK"&M$&"." :: DISP
LAY AT(6,1):"OUTPUT TO PRINT
ER? (Y/N): N"
1500 ACCEPT AT(6,27)VALIDATE
(2$)BEEP SIZE(-1):I$ :: IF I
$="y" THEN I$="Y"
1510 IF I$<>"Y" THEN 1530 EL
SE DISPLAY AT(6,1):"PRINTER
NAME: PIO"
1520 ACCEPT AT(6,15)BEEP SIZ
E(-14):J$
1530 DISPLAY AT(8,1):"READIN
G...": : : : : :"A TO ABORT"
1540 CALL HCHAR(10,2,126):::
CALL HCHAR(10,3,127,28):: CA
LL HCHAR(11,2,129)
1550 CALL HCHAR(10,31,128)::
 CALL HCHAR(11,31,130):: CAL
L HCHAR(12,2,131)
1560 CALL HCHAR(12,3,132,28)
:: CALL HCHAR(12,31,133)
1570 OPEN #1:M$,INPUT ,RELAT
IVE,INTERNAL :: INPUT #1:N$,
G,G,D :: FOR A=1 TO 127
1580 CALL KEY(0,J,H):: IF J=
65 OR J=97 THEN 1640
1590 INPUT #1:D$,K,L,M :: IF
 LEN(D$)=0 THEN 1640
1600 G$(A)=D$&RPT$(" ",11-LE
N(D$))&STR$(L)&RPT$(" ",5-LE
N(STR$(L)))&F$(ABS(K))
1610 IF ABS(K)<5 THEN P$=" "
&STR$(M):: G$(A)=G$(A)&SEG$(
P$,LEN(P$)-2,3)
1620 IF K<1 THEN G$(A)=G$(A)
&" Y"
1630 DISPLAY AT(11,1):G$(A):
: NEXT A
1640 CLOSE #1
1650 CALL CLEAR :: IF I$="Y"
 THEN OPEN #1:J$
1660 PRINT TAB(9);"A TO ABOR
T";TAB(9);"P TO PAUSE": :
1670 PRINT M$;" - DISKNAME=
";N$:"AVAILABLE=";D;"USED=";
G-D
1680 IF I$="Y" THEN PRINT #1
:M$;" - DISKNAME= ";N$:"AVAI
LABLE=";D;"USED=";G-D
1690 PRINT :" FILENAME  SIZE
  FILETYPE  P---------- ----
 ---------- -";
1700 IF I$="N" THEN 1720
1710 PRINT #1:" FILENAME  SI
ZE  FILETYPE  P":"----------
 ---- ---------- -"
1720 FOR A=1 TO A :: CALL KE
Y(0,D,H):: IF D=65 OR D=97 T
HEN 1760
1730 IF D=80 OR D=112 THEN 1
810
1740 PRINT G$(A):: IF I$="Y"
 THEN PRINT #1:G$(A)
1750 NEXT A
1760 IF I$="Y" THEN CLOSE #1
1770 PRINT :"LIST AGAIN (Y/N
)?"
1780 CALL KEY(0,D,H):: IF H=
0 THEN 1780
1790 IF D=89 THEN 1650
1800 GOTO 840
1810 FOR D=1 TO 50 :: NEXT D
 :: CALL KEY(0,D,H):: IF H=0
 THEN 1810
1820 GOTO 1740
1830 DISPLAY AT(24,1):"SCREE
N COLOR? (2-16): <2 >"
1840 ACCEPT AT(24,24)VALIDAT
E(DIGIT)BEEP SIZE(-2):I
1850 DISPLAY AT(24,1):"LETTE
R COLOR? (2-16): <16>"
1860 ACCEPT AT(24,24)VALIDAT
E(DIGIT)BEEP SIZE(-2):A
1870 IF A=1 THEN 1830
1880 CALL SCREEN(I):: FOR I=
1 TO 14 :: CALL COLOR(I,A,1)
:: NEXT I
1890 GOTO 840
1900 CALL CLEAR :: PRINT "ER
ROR, RE-BOOTING..." :: RUN
```

TIPS FROM THE TIGERCUB

#46

TIGERCUB SOFTWARE
156 Collingwood Ave.
Columbus, OH 43213

Distributed by Tigercub Software to TI-99/4A Users Groups for promotional purposes and in exchange for their newsletters. May be reprinted by non-profit users groups, with credit to Tigercub Software.

Over 130 original programs in Basic and Extended Basic, available on cassette or disk, NOW REDUCED TO JUST $1.00 EACH!, plus $1.50 per order for cassette or disk and PP&H. Minimum order of $10.00. Cassette programs will not be available after my present stock of blanks is exhausted. The Handy Dandy series, and Color Programming Tutor, are no longer available on cassette.
Descriptive catalogs, while they last, $1.00 which is deductable from your first order.

Tigercub Full Disk Collections, reduced to $5 postpaid. Each of these contains either 5 or 6 of my regular catalog programs, and the remaining disk space has been filled with some of the best public domain programs of the same category. I am NOT selling public domain programs - they are a free bonus!
TIGERCUB'S BEST, PROGRAMMING TUTOR, PROGRAMMER'S UTILITIES, BRAIN GAMES, BRAIN TEASERS, BRAIN BUSTERS!, MANEUVERING GAMES, ACTION REFLEX AND CONCENTRATION, TWO-PLAYER GAMES, KID'S GAMES, MORE GAMES, WORD GAMES, ELEMENTARY MATH, MIDDLE/HIGH SCHOOL MATH, VOCABULARY AND READING, MUSICAL EDUCATION, KALEIDOSCOPES AND DISPLAYS

NUTS & BOLTS (No. 1), a full disk of 100 Extended Basic utility subprograms in merge format, ready to merge into your own programs. Plus the Tigercub Menuloader, a tutorial on using subprograms, and 5 pages of documentation with an example of the use of each subprogram. Reduced to $15.00 postpaid.
NUTS & BOLTS NO. 2, another full disk of 108 utility subprograms in merge format, all new and fully compatible with the last, and with 10 pages of documentation and examples. Also $15 postpaid.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% NUTS & BOLTS #3 is now        %
% ready, another full disk      %
% of 140 new merge-format       %
% utility subprograms, all      %
% compatible with the pre-      %
% vious. With 11 pages of       %
% documentation, $15 ppd.       %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

TIPS FROM THE TIGERCUB, a full disk containing the complete contents of this newsletter Nos. 1 through 14, 50 original programs and files, reduced to $10 ppd.
TIPS FROM THE TIGERCUB VOL. 2, another diskfull, complete contents of Nos. 15 through 24, over 60 files and programs, also just $10
TIPS FROM THE TIGERCUB VOL. 3, another 62 programs, tips and routines from Nos. 25 through 32, $10 postpaid.
TIPS FROM THE TIGERCUB VOL. 4, another 48 programs and files from issues 33 through 41, also $10 postpaid.

TIGERCUB CARE DISKS #1, #2 & #3, three full disks of text files, mostly of lessons on programming in XBasic, $5 per disk postpaid.

This one is explained in lines 180-190. I think that it will run on any Gemini printer.

```
100 DIM B(25,12),B$(25),CH$(
12),L$(12)
110 GOTO 150
120 S,K,T$,C$,V,J,A,CH$(),X,
X$,B$(),B(X,J),T,M,Q$,L$(),C
,C1$,C2$,L,M$
130 CALL CLEAR :: CALL COLOR
 :: CALL SCREEN :: CALL CHAR
 :: CALL KEY :: CALL NUMTH
140 !@P-
150 !SEGMENTED BAR GRAPH
     by Jim Peterson 10/87
160 CALL CLEAR :: FOR S=1 TO
 12 :: CALL COLOR(S,2,8):: N
EXT S :: CALL SCREEN(5):: DI
SPLAY AT(3,10):"TIGERCUB" ::
 DISPLAY AT(5,6):"SEGMENTED
BAR GRAPH"
170 CALL CHAR(95,"3C4299A1A1
99423C"):: DISPLAY AT(7,12):
"_ 1987" :: DISPLAY AT(9,2):
"For free distribution but n
o";"price or copying fee may
 be";"charged."
180 DISPLAY AT(14,2):" Will
output to a Gemini";"printer
 e horizontal bar-";"graph o
f up to 25 bars, each";"segm
ented into up to 12"
190 DISPLAY AT(18,1):"values
, with a title for";"each an
d optionally with a";"table
of identification of";"the s
egment symbols."
200 DISPLAY AT(24,8):"" :: D
ISPLAY AT(24,8):"PRESS ANY K
EY" :: CALL KEY(0,K,S):: IF
 S=0 THEN 200
210 ON WARNING NEXT
220 DISPLAY AT(12,1)ERASE AL
L:"GRAPH TITLE?" :: ACCEPT A
T(14,1):T$ :: T$=RPT$(" ",17
-LEN(T$)/2)&T$ :: C$=CHR$(27
)
230 DISPLAY AT(16,1):"HOW MA
NY SEGMENTS PER BAR?" :: ACC
EPT AT(16,27)VALIDATE(DIGIT)
SIZE(2):V :: IF V=0 OR V>12
THEN 230
240 !@P+
250 DATA 239,229,168,251,173
,175,184,236,169,250,160,207
260 !@P-
270 FOR J=1 TO V :: READ A :
: CH$(J)=CHR$(A):: NEXT J
280 DISPLAY AT(3,1)ERASE ALL
:"Type END when finished"
290 X=X+1 :: IF X)25 THEN 33
0
300 CALL NUMTH(X,X$):: DISPL
AY AT(12,1):"Title of "&X$&"
 bar?" :: ACCEPT AT(14,1):B$
(X):: IF B$(X)="END" OR B$(X
)="end" THEN 330
310 FOR J=1 TO V :: CALL NUM
TH(J,X$):: DISPLAY AT(16,1):
X$&" segment value?" :: ACCE
PT AT(18,1)VALIDATE(NUMERIC)
:B(X,J):: T=T+B(X,J):: NEXT
 J
320 M=MAX(M,T):: T=0 :: GOTO
 290
330 X=X-1 :: DISPLAY AT(20,1
):"Print labels? Y/N" :: ACC
EPT AT(20,19)VALIDATE("YN")S
IZE(1):Q$ :: IF Q$="N" THEN
350
340 FOR J=1 TO V :: CALL NUM
TH(J,X$):: DISPLAY AT(22,1):
X$&" label?" :: ACCEPT AT(24
,1):L$(J):: NEXT J
350 C=120/M :: C1$=C$&"B"&CH
R$(1)&C$&"G"&C$&"E" :: C2$=C
$&"B"&CHR$(3)
360 OPEN #1:"PIO",VARIABLE 2
55 :: PRINT #1:C$&"D" :: PRI
NT #1:C$&"E"&C$&"G"&C$&"M"&C
HR$(6)
370 PRINT #1:CHR$(14)&T$&CHR
$(20)&"";RPT$(CHR$(229),70);
 :: PRINT #1:C$&"J"&CHR$(1
0)
380 FOR J=1 TO X :: PRINT #1
;B$(J)&C2$ :: FOR L=1 TO V :
: M$=M$&RPT$(CH$(L),INT(B(J,
L)*C+.5)):: NEXT L
390 PRINT #1:RPT$(CHR$(232),
LEN(M$)):: PRINT #1:M$ :: PR
INT #1:M$ :: PRINT #1:RPT$(C
HR$(231),LEN(M$))
400 M$="" :: PRINT #1:C1$;::
 NEXT J :: IF Q$="N" THEN ST
OP
410 PRINT #1:"";""
420 FOR J=1 TO V :: PRINT #1
:C2$&RPT$(CHR$(232),10):: PR
INT #1:RPT$(CH$(J),10)&C1$&"
     "&L$(J):: PRINT #1:C2$&
RPT$(CH$(J),10):: PRINT #1:R
PT$(CHR$(231),10):: NEXT J
430 !@P+
440 SUB NUMTH(N,N$):: IF FLA
G=1 THEN 520 :: FLAG=1 :: RE
STORE 480
450 GOTO 480
```

```
460 J,ONE$(),TEEN$(),TEN$(),
N,N$
470 !@P-
480 DATA first,second,third,
fourth,fifth,sixth,seventh,e
ighth,ninth,tenth
490 DATA eleventh,twelfth,th
irteenth,fourteenth,fifteent
h,sixteenth,seventeenth,eigh
teenth,nineteenth
500 DATA twenty,THIRTY,FORTY
,FIFTY,SIXTY,SEVENTY,EIGHTY,
NINETY
510 FOR J=1 TO 10 :: READ BN
E$(J):: NEXT J :: FOR J=1 TO
 9 :: READ TEEN$(J):: NEXT J
 :: FOR J=2 TO 9 :: READ TEN
$(J):: NEXT J
520 IF N<11 THEN N$=ONE$(N):
: SUBEXIT
530 IF N<20 THEN N$=TEEN$(N-
10):: SUBEXIT
540 IF N/10=INT(N/10)THEN N$
=SE$(TEN$(N/10),1,LEN(TEN$(
N/10))-1)&"ieth" :: SUBEXIT
550 N$=TEN$((INT(N/10))&"-"&O
NE$((N/10-INT(N/10))*10)
560 !@P+
570 SUBEND

  And  a little something
educational -

100 DIM M$(100)
110 GOTO 150
120 S,J,M$(),A$,Z$,K,W$(),X,
Y,ADV$,A,Q$
130 CALL CLEAR :: CALL COLOR
 :: CALL SCREEN :: CALL CHAR
 :: CALL KEY :: CALL ADVERB
 :: CALL SOUND
140 !@P-
150 CALL CLEAR :: FOR S=0 TO
 12 :: CALL COLOR(S,2,8):: N
EXT S :: CALL SCREEN(5):: BI
SPLAY AT(3,2):"ADJECTIVE TO
ADVERB  V.I.3"
160 CALL CHAR(64,"3C4299A1A1
99423C"):: BISPLAY AT(5,6):"
@ Tigercub Software":!:" For
 free distribution with  no
charge or copying fee."
170 FOR J=1 TO 100 :: READ M
$(J):: A$=A$&CHR$(J):: NEXT
J :: Z$=A$ :: CALL KEY(3,K,S
)
180 W$(1)=" If adjective end
s in Y,     change the Y to
ILY." :: W$(2)=" If adjectiv
e ends in C,    add ALLY."
190 W$(3)=" If adjective end
s in LL,    just add Y."
200 W$(4)=" If adjective end
s in LE,    preceded by a con
sonant,     drop the E and ad
d Y."
210 W$(5)=" If the word ends
 in E       preceded by a con
sonant,     preceded by a vow
el, just    add LY."
220 W$(6)=" This word is an
exception  to the rule - the
 adverb is WHOLLY."
230 W$(7)=" If the adjective
 does not   end in C,E,LL or
 Y, always  just add LY."
240 W$(8)=" This is an excep
tion to the rule. The prefe
rred adverb   form is BRYLY."
250 W$(9)=" If the adjective
 ends in E  preceded by a vo
wel, drop   the E and add LY
."
260 W$(10)=" If the adjectiv
e ends in E preceded by a co
nsonant     other than L, ad
d LY."
270 RANDOMIZE :: X=INT(RND$L
EN(Z$)+1):: Y=ASC(SEG$(I$,I,
1)):: Z$=SEG$(Z$,1,X-1)&SEG$
(Z$,X+1,255):: IF LEN(Z$)=0
THEN Z$=A$
280 ACCEPT AT(24,1):M$(Y)
290 CALL ADVERB(M$(Y),ADV$,A
)
300 BISPLAY AT(12,1):" Type
the adverb form of -" :: BIS
PLAY AT(15,1):M$(Y):: BISPLA
Y AT(16,10):"" :: ACCEPT AT(
15,15)BEEP:Q$
310 IF Q$=ADV$ THEN BISPLAY
AT(18,10):"CORRECT!" :: GOTO
 240
320 CALL SOUND(100,110,5,-4,
5):: BISPLAY AT(20,1):W$(A):
"":"" :: GOTO 300
330 !@P+
340 DATA BUE,COOL,SOLE,STOIC
,FRANTIC,COMIC,ABLE,FULL,POO
R,HANDY,SORE,SOCIAL,PENAL,SL
OW,HIGH,LOW
350 !@P-
360 DATA FRISKY,PLAYFUL,HEAL
THY,ROUGH,BUSY,SILLY,SICK,SM
ART,SORE,FAIR,ANGRY,BARE,TIR
ED,WISHFUL,ACTUAL
370 DATA HASTY,LONE,HECTIC,O
FFICIAL,MAGIC,MAGICAL,MATHEM
ATIC,LOGIC,TRAGIC,PATHETIC,T
RAUMATIC
380 DATA DRAMATIC,AUTOMATIC,
AROMATIC,EQUAL,SERIAL,BASIC,
USUAL,FAVORABLE,UNSTABLE,LEG
IBLE
390 DATA HECTIC,LIVE,WARY,VI
SIBLE,TERRIBLE,HORRIBLE,VIVI
D,FAMCY,EASY,VILE,WICKED,BLO
ODY,SHODDY
400 DATA NOBLE,HAPPY,LEGAL,M
ERRY,JOLLY,CRAZY,CASUAL,CARE
FUL,FOOLISH,FAMOUS,GAY,GUILT
Y
410 DATA HOPEFUL,HATEFUL,TIM
IB,BRAVE,BEAUTIFUL,DRY,NICE,
LARGE,PAINFUL,SINFUL,SORROWF
UL,SIMPLE,WILLFUL
420 DATA MENTAL,MORAL,PALE,W
HOLE,HUNGRY,FINAL,FORMAL,TRU
E,AMPLE,DOUBLE
430 !@P+
440 SUB ABVERB(M$,ADV$,A)::
L=LEN(M$):: E$=SE$(M$,L,1)::
 F$=SE$(M$,L-1,2):: B$=SEB
$(M$,L-1,1):: P$=SEB$(M$,L,L
-1):: H$=SE$(M$,L-2,1)
450 IF ASC(SEG$(M$,1,1))<97
THEN A$="ALLY" :: I$="ILY" :
: L$="LY" :: Y$="Y" :: V$="A
EIOU" ELSE A$="ally" :: I$="
ily" :: L$="ly" :: Y$="y" ::
460 IF M$="WHOLE" THEN ADV$=
"WHOLLY" :: A=6 :: SUBEXIT
470 IF M$="DRY" THEN ABV$="B
RYLY" :: A=8 :: SUBEXIT ELSE
 IF F$="LL" OR F$="II" THEN
ADV$=M$&Y$ :: A=3 :: SUBEXIT
480 IF E$="C" OR E$="c" THEN
 ABV$=M$&A$ :: A=2 :: SUBEXI
T ELSE IF E$="Y" OR E$="y" T
HEN ADV$=P$&I$ :: A=1 :: SUB
EXIT
490 IF E$<>"E" AND E$<>"e" T
HEN 530
500 IF B$="L" OR B$="l" THEN
 IF POS(V$,H$,1)<>0 THEN ADV
$=M$&L$ :: A=5 :: SUBEXIT EL
SE ADV$=P$&Y$ :: A=4 :: SUBE
XIT
510 IF POS(V$,B$,1)<>0 THEN
ADV$=P$&L$ :: A=9 :: SUBEXIT
520 IF POS(V$,SE$(M$,L-2,1)
,1)=0 THEN ADV$=M$&L$ :: A=1
0 :: SUBEXIT ELSE ADV$=M$&L$
 :: A=5 :: SUBEXIT
530 ADV$=M$&L$ :: A=7 :: SUB
END

100 !MOCKINGBIRD TINYGRAM by
    Jim Peterson. Tap  your
    tune on the 1 to 0 keys
      (tuned A through C)
110 !Then press any other
    key to hear it repeated
120 DATA 220,247,262,294,330
,349,392,440,494,523
130 FOR J=1 TO 10 :: READ N(
J):: NEXT J :: J=0 :: DIM T(
50,2)
140 CALL KEY(5,K,S):: IF S=0
 THEN 140
150 ON ERROR 190
160 CALL KEY(5,K,S):: IF K=-
1 THEN 160 :: K=K-(K=48)*10
:: T(J,1)=N(K-48):: CALL SOU
ND(-999,T(J,1),0)
170 IF K=K2 THEN T(J,2)=T(J,
2)+1 :: GOTO 150
180 K2=K :: J=J+1 :: GOTO 16
0
190 FOR X=0 TO J-1 :: CALL S
OUND((T(X,2)+1)*400,T(X,1),0
,T(X,1)*1.01,0):: NEXT X ::
J=0 :: GOTO 140

  A little subprogram to add
a  bit  of  variety  to your
"PRESS ANY KEY" routine.

1 CALL CLEAR :: CALL PRESSKE
Y(24)
30000 SUB PRESSKEY(R)
30001 C=C+1 :: IF C=16 THEN
30002 :: DISPLAY AT(R,1):""
:: DISPLAY AT(R,C):"PRESS AN
Y KEY" :: DISPLAY AT(R,C):"p
ress any key" :: CALL KEY(0,
K,S):: IF S=0 THEN 30001 ELS
E 30003
30002 C=C-1 :: IF C=0 THEN 3
0001 :: BISPLAY AT(R,1):"" :
: DISPLAY AT(R,C):"PRESS ANY
 KEY" :: DISPLAY AT(R,C):"pr
ess any key" :: CALL KEY(0,K
,S):: IF S=0 THEN 30002
30003 DISPLAY AT(R,1):"" ::
SUBEND

  And  a new way to wipe the
screen -

1 CALL CORNERWIPE(30)
29000 SUB CORNERWIPE(CH):: F
OR T=1 TO 24 :: CALL HCHAR(T
,3,CH,T+4):: CALL HCHAR(25-T
,32-T,CH,T):: NEXT T :: CALL
 CLEAR :: SUBEND

      MEMORY FULL

      Jim Peterson
```

TEAM STATISTICS PROGRAM FOR MINI-MEMORY

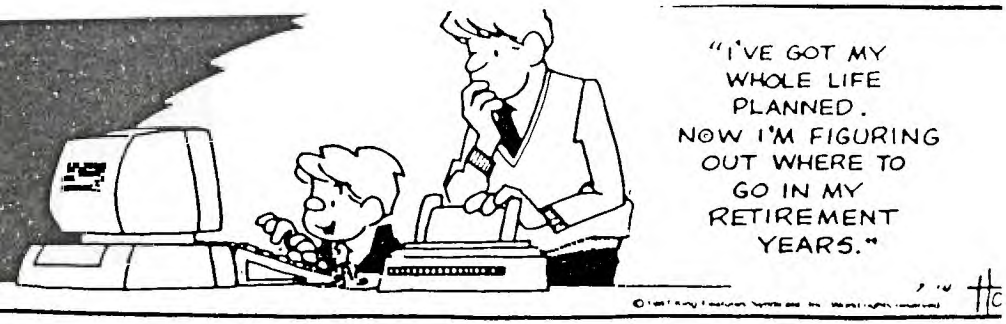
```
100 CALL CLEAR
110 PRINT "***BASEBALL STATI
STICS***": : :
120 INPUT "ENTER DATE OF LAS
T GAME      ":DATE$
130 PRINT "ON 'AB' PROMPT, P
RESS 0 TO":"SKIP THE BATTER,
 OR PRESS":"99 TO REDO THE L
AST BATTER"::
140 PRINT "TO SET UP FILES F
OR 1ST TIME":"PRESS 1.":"OTH
ERWISE, PRESS ENTER."
150 PRINT "MINIMEMORY IS REQ
UIRED"
160 DIM NAME$(15),AB(15),H(1
5),BB(15),SO(15),B(15),AV(15
),OBAV(15)
170 DIM NEWAB(15),NEWHITS(15
),NEWBB(15),NEWSO(15),NEWBAS
ES(15),SAC(15)
180 FOR I=1 TO 15
190 READ NAME$(I)
200 NEXT I
210 CALL KEY(0,K,S)
220 IF S=0 THEN 210
230 CALL CLEAR
240 IF K<>49 THEN 940 ELSE 8
20
250 REM LIST STATS TO SCREEN

260 FOR I=1 TO 15
270 PRINT NAME$(I):"AB";AB(I
);TAB(10);"HITS";H(I);TAB(20
);"BB";BB(I)
280 PRINT "SO";SO(I);TAB(10)
;"BASES";B(I);TAB(20);"%";AV
(I)
290 IF I=7 THEN 310
300 IF I=15 THEN 320 ELSE 34
0
310 PRINT "PRESS ANY KEY TO
CONTINUE"
320 CALL KEY(0,K,S)
330 IF S=0 THEN 320
340 NEXT I
350 RESTORE
360 GOTO 1060
370 REM INPUT NEW DATA
380 A=0
390 A=A+1
400 PRINT NAME$(A)
410 INPUT "AB ":NEWAB(A)
420 IF (NEWAB(A)=0)+(A>=15)=
-2 THEN 860
430 PRINT AB(A)
440 IF NEWAB(A)=0 THEN 750
450 IF NEWAB(A)=99 THEN 460
ELSE 540
460 A=A-1
470 AB(A)=AB(A)-NEWAB(A)
480 H(A)=H(A)-NEWHITS(A)
490 BB(A)=BB(A)-NEWBB(A)
500 SO(A)=SO(A)-NEWSO(A)
510 AB(A)=AB(A)+SAC(A)
520 B(A)=B(A)-NEWBASES(A)
530 GOTO 400
540 AB(A)=AB(A)+NEWAB(A)
550 PRINT AB(A)
560 INPUT "HITS ":NEWHITS(A)
570 H(A)=H(A)+NEWHITS(A)
580 PRINT H(A)
590 INPUT "BB ":NEWBB(A)
600 BB(A)=BB(A)+NEWBB(A)
610 AB(A)=AB(A)-BB(A)
620 PRINT BB(A)
630 INPUT "SO ":NEWSO(A)
640 SO(A)=SO(A)+NEWSO(A)
650 PRINT SO(A)
660 INPUT "SACRIFICES":SAC(A
)
670 AB(A)=AB(A)-SAC(A)
680 IF NEWHITS(A)=0 THEN 720
690 INPUT "BASES ":NEWBASES(
A)
700 B(A)=B(A)+NEWBASES(A)
710 PRINT B(A)
720 AV(A)=INT(H(A)/AB(A)*100
0)
730 AB(A)=AB(A)+BB(A)
740 PRINT AV(A)
750 OB=H(A)+BB(A)
760 PRINT AB(A)
770 OBAV(A)=INT(OB/AB(A)*100
0)
780 PRINT OBAV(A)
790 IF A<15 THEN 390 ELSE 86
0
800 DATA JASON,MARC,JACK,TED
,CHASE,NATHAN,JIMMY,JOHN,JON,
DANNY,ROMEL,CARRIE,MICKEAL,C
RES,CRAIG
810 REM SET UP FILES FOR FIR
ST TIME
820 PRINT "THIS WILL ERASE T
HE FILES...ARE YOU SURE? (Y/
N)"
830 INPUT YN$
840 IF YN$="Y" THEN 860 ELSE
 940
850 IF YN$="y" THEN 860 ELSE
 940
860 OPEN #3:"MINIMEM",RELATI
VE,OUTPUT,INTERNAL
870 FOR I=1 TO 15
880 PRINT #3:AB(I),H(I),BB(I
),SO(I),B(I),AV(I)
890 NEXT I
900 PRINT #3:99,99,99,99,99,
99
910 CLOSE #3
920 GOTO 1070
930 REM INPUT DATA FROM FILE
S
940 OPEN #3:"MINIMEM",RELATI
VE,INPUT ,INTERNAL
950 FOR I=1 TO 15
960 INPUT #3:C,D,E,F,G,J
970 IF C=99 THEN 1050
980 AB(I)=C
990 H(I)=D
1000 BB(I)=E
1010 SO(I)=F
1020 B(I)=G
1030 AV(I)=J
1040 NEXT I
1050 CLOSE #3
1060 CALL CLEAR
1070 PRINT ">1< PRINT OUT ST
ATS":">2< LIST STATS ON SCRE
EN":">3< ENTER A GAME'S STAT
S"
1080 PRINT ">4< READ OR CHAN
GE AN           INDIVIDUAL'S
 STATS":">5< END"
1090 PRINT :"(JUST KEEP ENTE
RING GAMES  SEQUENTIALLY, T
HEY'RE AUTO- MATICALLY SAVED
 EACH TIME.)"
1100 PRINT :"ENTER THE PLAYE
R'S NAMES IN THE DATA STATEM
ENTS IN LINE 760-THE DIMENSI
ONS ARE SET-UP FOR 15 NAMES"

1110 PRINT ::"NOTE: YOU MUST
 RUN THROUGH  THE LINE UP ON
E TIME EVERY  TIME THE PROGR
AM IS RUN FOR"
1120 PRINT "THE ON BASE AVER
AGES TO BE  CALCULATED": :
1130 INPUT Z$
1140 IF Z$="1" THEN 1200
1150 IF Z$="2" THEN 250
1160 IF Z$="3" THEN 380
1170 IF Z$="4" THEN 1330
1180 IF Z$="5" THEN 1610 ELS
E 1070
1190 REM LIST STATS TO PRINT
ER
1200 OPEN #4:"PIO"
1210 PRINT #4:CHR$(27)&CHR$(
71)
1220 PRINT #4:"BATTING AVERA
GES AS OF ";DATE$: :
1230 PRINT #4:"NAME";CHR$(9)
;"AT BATS";CHR$(9);"HITS";CH
R$(9);"BB";CHR$(9);"BASES";C
HR$(9);"S.O.";CHR$(9);"AVE."

1240 PRINT #4:CHR$(9);"ON BA
SE AVE"
1250 PRINT #4
1260 FOR I=1 TO 15
1270 PRINT #4:NAME$(I);CHR$(
9);AB(I);CHR$(9);H(I);CHR$(9
);BB(I);CHR$(9);B(I);CHR$(9
);SO(I);CHR$(9);AV(I);
1280 PRINT #4:CHR$(9);OBAV(I
)
1290 PRINT #4
1300 NEXT I
1310 CLOSE #4
1320 GOTO 1060
1330 CALL CLEAR
1340 INPUT "PLAYER'S NAME? "
:Z$
1350 RESTORE
1360 FOR I=1 TO 15
1370 READ NAME$(I)
1380 IF NAME$(I)=Z$ THEN 144
0
1390 NEXT I
1400 PRINT "NAME NOT FOUND I
N DATA FILE"
1410 CALL SOUND(2000,110,1)
1420 CALL SOUND(100,440,1)
1430 GOTO 1070
1440 OPEN #3:"MINIMEM",RELAT
IVE,UPDATE,INTERNAL
1450 INPUT #3,REC I-1:U,V,W,
X,Y,Z
1460 CLOSE #3
1470 PRINT "TOTAL AT BATS",U:"
HITS",V,,:"WALKS","STRIKE O
UT",W,X::"BASES",AVERAGE,Y,Z

1480 PRINT : :">1< RETURN":
">2< CHANGE THESE STATS"
1490 INPUT Z$
1500 IF Z$="2" THEN 1510 ELS
E 1060
1510 OPEN #3:"MINIMEM",RELAT
IVE,UPDATE,INTERNAL
1520 INPUT "AT BATS ":AB(I)
1530 INPUT "HITS ":H(I)
1540 INPUT "WALKS ":BB(I)
1550 INPUT "STRIKE OUTS ":SO
(I)
1560 INPUT "BASES ":B(I)
1570 AV(I)=H(I)/(AB(I)-BB(I)
)
1580 PRINT #3,REC I-1:AB(I),
H(I),BB(I),SO(I),B(I),AV(I)
1590 CLOSE #3
1600 GOTO 1060
1610 END
```



TI-99/4A

HOME COMPUTER SPECIALISTS

WE HAVE WHAT YOU NEED FOR YOUR TI!

COMPETITION COMPUTER PRODUCTS

2629 W. NATIONAL AVE. MILWAUKEE, WIS. 53204
(near the Mitchell Park Domes)

STORE HOURS; MON THRU FRI 10-6 SAT 10-3

672-4010

BANKCARDS - CHECKS - DISCOVER CARDS WELCOME!

WOW! DS/DD DISKS .54 EACH!

GENUINE TI JOYSTICKS $10 PER PAIR!
(with this flyer only - regularly $29.95 - while supply lasts)

PICK UP YOUR COPY OF OUR CATALOG SOON

WE WILL BUY ANY TI HARDWARE OR SOFTWARE YOU NO LONGER NEED - CALL!

WE CUSTOM BUILD IBM COMPATIBLE COMPUTERS & TAKE TI ITEMS IN TRADE.

FEATURING PANASONIC & STAR MICRONICS PRINTERS FROM $189!

NEW AND USED TI99/4A COMPUTERS AVAILABLE!

EXPANSION SYSTEMS AVAILABLE - NEW AND USED!

* HUGE SOFTWARE INVENTORY - MORE IN STOCK THAN EVER BEFORE! * CALL US FOR TECHNICAL HELP. WE WILL HELP YOU WITH YOUR PROBLEMS. WE WILL TRY TO MEET OR BEAT ANYBODY'S PRICES. REMEMBER THAT WE ARE HERE TO HELP IF YOU HAVE A QUESTION OR PROBLEM. WE WANT YOUR BUSINESS AND WE'LL PROVE IT!

OUR 21ST YEAR IN BUSINESS IN MILWAUKEE AT 27TH & NATIONAL!

TED, GENE, MIKE, JIM AND JERRY

1/88