

KC 99'er

CONNECTION



A KANSAS CITY PUBLICATION

* Volume 7 KC 99'er BBS 436-9074 Issue 1 *

JANUARY 1988

REPORT ON THE FIFTH CHICAGO TI FAIRE NOVEMBER 7, 1987

COPYRIGHT 1987 by Warren Agee - Permission to reproduce for non-profit only.

I had the pleasure of attending the Fifth Annual Chicago TI Faire, one that is fast becoming the grand-daddy of national TI/Geneve gatherings... A good number of people showed up, and although I don't have the exact figures, the room seemed just as full as last year. This article gives a report on the Faire. Highlights some new products and a brief summary of all the great personages I had a chance to meet. Unfortunately, there were too many people to meet and not enough time to spend sufficient time at every single booth at the show, but I think I did manage to see most of the 'new exciting' products (to come out this year or so anyway.)

GENIAL COMPUTERWARE: This booth, headed by the ever-popular J. Peter Hoddie, was very busy all day long, and for good reason; they released PC-Transfer by Mike Dodd, which facilitates the transfer of data between the 4/A or Geneve and an MS-DOS system. It can format an IBM data diskette with a TI drive, and transfer files between the two systems, both ways. Corcomp has been selling a similar product for quite some time, but it's on a cartridge and cost twice as much as PC-Transfer, which sells for \$25.00 ... Also released was a new program called 'Graphics Expander' (by JPH) which takes a TI Artist or CGSD font and changes it size. Written entirely in Assembly, it allows you to stretch a font horizontally and vertically. You can save the result as EITHER a TI ARTIST or CGSD font. That means you can easily convert fonts to either format! Again, very slick and sells for \$10 ... Starting revelation, the J stands for James as was found out in a presentation by JPH as he demonstrated his company's new software. **IMPORTANT** Due to his untimely illness, Barry Iraver, the other half of Genial Computerware, was unable to attend this year's faire, and he was sorely missed. Everyone say a prayer for Barry that he return to us soon and in good health.

KYTE DATA: I only was able to get a brief demo from this company, but I thought it appropriate to mention Kyte Data's new 99AT Expansion System for both the 4A and Geneve. It looks like an IBM PC on the outside, but the innards have been rearranged to hold a large power supply. Up to 4 half-height drives (including a hard drive), and 5 slots for standard TI size cards made by TI, Myarc, Corcomp, Horizon, Foundation, Mechatronic, etc. Now with the Geneve, we need a new expansion box, and here we have

one. PRICE \$155 with 135 watt power supply and built-in system interface. Just plug in your Geneve and assorted cards, some drives, and you're all set.

DJUII Systems had their AVFD card running Multiplan in 80 columns. Asgard was there with their full line of software, and a smiling Chris Bobbit wearing an Asgard t-shirt. 'Twas a great pleasure to see Terrie Masters again at the LA 99'ers booth, and had the honor to meet Tom Freeman for the first time. Jack Riley was holding up the fort at the Myarc booth while being bombarded with questions about everything from "where is the HDCC?" to "why did you wear a polkadotted tie?" Peter Hoddie helped Jack at the tail end of the presentation by talking about MY-WORD 2. He would basically like to see MY-WORD completely user-definable. He did not specify a completion date, but estimates that perhaps it would be ready mid-to-late winter.

I met Mike Dodd, whom Peter describes as 'scary' he's so good. Only 15 and he does work for Myarc writing commercial software (like PC Transfer) and authors a Geneve Column for Micropendium. Now if J. Peter Hoddie calls Mike Dodd scary, that scare me even more! Great talent, and a nice guy to boot. That's it for the FACIS concerning the CHICAGO TI FAIRIE. I'm sure I've missed something, but there was a lot to see, both in booths and people.

<< PLEASE SUPPORT YOUR BBS call 436-9074 KC 99'er >>

FROM NEW HORIZONS' NEWLETTER ANIMATION IN EXTENDED BASIC
by Steve Schwartz Reprinted in LA 99ERS TOPICS Aug 87

One way to spice up Xbasic games or programs is to produce some animated effects. These are created by using SPRITES. Now in order to do this you need to create a set of patterns which show successive frames of the thing you wish to be animated. (like the frames of a motion picture) After placing the SPRITE(s) on the screen you just quickly change the SPRITE patterns with a CALL PATTERN statement.

The following short routine below produces an explosion on the screen. Why not give it a try! You might become creative and do some neat effects.

```
10 CALL CLEAR :: CALL SCREEN(2)
11 CALL MAGNIFY(2)
12 CALL CHAR(96,"000000000000802000200000000000000000000000004000080008")
13 CALL CHAR(104,"00000102000210020004110008000000000000000040000100040000104")
14 CALL CHAR(100,"00000000040002080108000200000000000000000080000400004000080")
15 CALL CHAR(108,"0000000210010420000200011400000000040100440140088220044002008")
16 CALL MAGNIFY(2)
17 CALL SPRITE(#1,65,16,100,100)
18 FOR A=1 TO 200 :: NEXT A
19 FOR I=96 TO 108 STEP 4
20 CALL PATTERN(#1,I)
21 CALL MAGNIFY(4)
22 CALL SOUND(500,-7,1)
23 NEXT I
24 CALL DELSPRITE(#1)
25 CALL KEY(O,K,S):: IF S=0 THEN 25 ELSE 16
```

LET'S TALK INTERRUPTS

Let me introduce myself. My name is John Phillips and I am a former game designer of Texas Instruments, Lubbock. Some of my creations are MUPPER, MOUNLINE, WORD KAPAK, and I worked on BURBERTIME, DEMON ATTACK, MUNCHMOBILE, JAWBREAKER, FACEMAKER, TREASURE ISLAND, ANGLER DANGLER, SLYMUIDS and just about everything else that came from LUBBUCK this past year. My forte is IMSY900 ASSEMBLY LANGUAGE and that is why I am delivering this article to you. I plan to deliver an assembly language article every month to you devoted /4A hackers out there. I still believe that the /4A is the best home computer on the market and I will quote that until the day I die. I would personally like to congratulate Don Veith for the task that he undertook: gathering all the /4A User groups together and uniting them with one common bond. I plan to support all the programmers that want to keep the /4A alive. If you have any questions concerning programming (either BASIC, EXTENDED BASIC, or ASSEMBLY LANGUAGE), just write to me and I'll do my best to get your problem resolved. I did it on all the packages just mentioned above. . . I'll do it for you, too!

The purpose of this article is to inform you about the ever-present INTERRUPTS that occur every 1/60 of a second within the /4A. They are not magical, but they certainly do create a whole lot of havoc for the beginning assembly language programmer. I'll try and explain what they're all about.

One problem with the 11-99/4 was the inflexibility of the VDP interrupt routine. In an effort to make some previously impossible applications possible, the VDP interrupt routine was modified. This routine performs 6 basic functions:

1. Sprite motion
2. Auto-sound
3. System reset key (function =)
4. Screen timeout
5. GFL timer increment
6. Storing VDP status in GPL status block

On the 11-99/4A, the first three of these functions are optional. The last three are not. The ability also exists to execute an additional interrupt

routine. The execution of each phase of the resident interrupt routine depends upon a bit flag in CPU RAM (>8300->83FF) location >83C2. If the most significant nybble of this byte is 0, all phases of the interrupt routine will execute. Setting any of the bits in this nybble will disable some part of the interrupt handler as follows:

```
bit 0 - disable sprite motion,
auto-sound, system reset key
bit 1 - disable sprite motion
bit 2 - disable auto-sound
bit 3 - disable system reset key
| | | | |
| a l l | motion | sound |quit key|
| | | |
```

For instance, if I wanted to disable the QUIT key, the assembly code to accomplish this desired feat is:

```
LI R0,>0100
MOVB R0,@>83C2
```

For you EXTENDED BASIC PROGRAMMERS, an accidental function = is disastrous. You can prevent this by the following 2 immediate statements:

```
CALL INIT
CALL LOAD(-31806,16)
```

VOILA! the QUIT key no longer functions. Use a load value of 0 to enable the QUIT key again.

After the routine has handled sprite motion and auto-sound, and has checked for the system reset key, the timing functions are performed and the VDP status is stored. The interrupt handler then checks whether there is an external routine to be executed.

The existence of an external routine is determined by the word value at CPU RAM >83C4. If this location is zero, there is no external routine and the interrupt handling is complete. A non-zero value in this location is assumed to be a pointer to another interrupt routine. Control is passed to that location.

At this point in the processing, the workspace pointer indicates the GPL workspace (>83E0). If the routine is to use this workspace (a call to KSCAN, for example), the values in certain registers must be preserved as follows:

1. If the routine is operating in the GPL environment, the values in registers 13, 14, and 15 must be preserved.
2. If the routine is operating in the BASIC or EXTENDED BASIC environment

(loading a program using the CALL LOAD), the values of registers 9, 10, 13, 14, and 15 must be preserved. Also, register 8 (>B3F0) of the GPL workspace must be cleared regardless of whether or not that workspace is used.

An external routine may conclude by returning to the console routine with the GPL workspace active. Right now register 8 is cleared and control is returned to the point at which the interrupt occurred. If preferred, an external routine may return directly to the point at which the interrupt occurred by loading the interrupt workspace pointer (>B3C0) and performing a RIWP instruction. I highly suggest this method!

Here is an example of an interrupt-driven program for you to review and enter in using the EDITOR/ASSEMBLER. This should clear up any confusion from the above article. Until next time . . .

```

DEF INT
REF VMBW
*
MYWS      B55 >20
MYWS2     B55 >20
TIMER     DATA 0          TIMER
INTWS     EQU >B3C0       INTERRUPT
WORKSPACE
GPLWS     EQU >B3E0       GPL WORKSPACE
*
MYNAME    TEXT "JOHN PHILLIPS"
BLANKS    TEXT "
*****
*****
* PROGRAM TO DEMONSTRATE EXTERNAL
INTERLUPT CODE.
*
*****
*****
INT      LIM1 0          DISABLE
INTERLUPTS
      LI R0,INTERR      GET ADDRESS OF
EXTERNAL INTERRUPT ROUTINE
      MOV R0,>B3C4      TELL INTERRUPT
HANDLER THERE IS ANOTHER ROUTINE
      LIM1 2          ENABLE
INTERLUPTS AGAIN
*
INT      LWPI MYWS      LOAD MY
WORKSPACE FOR PROGRAM
LOOP     JMP $          SAME AS "10
GOTO 10" IN
BAS.L

```

```

*****
*****
* EXTERNAL INTERRUPT ROUTINE. IF YOU
FOLLOWED THE ABOVE CODE, YOU WILL *
* REALIZE THAT IT DOES NOTHING EXCEPT AN
INFINITE LOOP. THIS INTERRUPT *
* CODE WILL PRINT MY NAME OUT AND ERASE
IT EVERY 1/60th OF A SECOND. *
*****
*****
INTERR    LWPI MYWS2      LOAD ANOTHER
WORKSPACE SO MY MAIN ONE ISN'T DESTROYED
      MOV @TIMER,@TIMER IS TIMER ZERO?
      JEQ TIMEUP        NO, SO EXIT
THIS ROUTINE
      DEC @TIMER        YES, SO PRINT
MY NAME
      JMP INTRTN        AND EXIT THIS
ROUTINE
TIMEUP    LI R0,60       RESET COUNTER
      MOV R0,@TIMER    AND SAVE IN RAM
VARIABLE
      LI R0,12*32+9    CENTER OF
SCREEN
      LI R1,MYNAME     RAM ADDRESS OF
MY NAME
      LI R2,14         14 BYTES TO
WRITE
      BLWP @VMBW       WRITE MY NAME
      LI R1,BLANKS     NOW POINT TO
BLANKS
      BLWP @VMBW       ERASE MY NAME .
. . DON'T BLINK!
INTRTN    LWPI INTWS    RESTORE
INTERLUPT WOKSPACE
      RTWP             RETURN TO MY
MAIN PROGRAM
*
      END INT          LOAD AND GO
OPTION

```

ANYBODY NEEDING JOHN PHILLIPS' ASSISTANCE ON THEIR PROGRAMS OR ANSWERS TO YOUR QUESTIONS, PLEASE WRITE TO THE ADDRESS BELOW:

JOHN PHILLIPS G & A
 99ERS USERS GROUP ASSN.
 3535 SO. H ST. #93
 BAKERSFIELD, CA 93304

PARAPHRASED MINUTES OF THE DEC. KC 99'ER MEETING

Our meeting was opened by Walter Blood on DEC. 13, 1987. After his welcoming to the visitors against us. The floor was opened to questions, problems and news to be shared.

A few questions were brought forth dealing with printers. Then a problem of the widget locking up. A trouble in the expansion box power supply and then the trouble of loading the Atari diskette. Many good suggestions were given from the other members.

The new business was next on the agenda: A new company that produces software for the 4/A and Genve is looking for people to mail their catalog sometime in JAN. Corcomp is offering T-shirts and tote bags having TI savings on them. Walter mentioned he is going to place another diskette order. Allowing all to participate if needed. An updated library catalog was brought in by John Dilly, but not system was here today to copy it. The amon order has come in and Walt will get a them copied by the next meeting or you can call him and prearranged to get your set.

The election of new officers was brought to the floor and Bill Talley wished to have the duties of each officer explained. This was done and then nominations for president was opened. Walter Blood was nominated by Gary Burns and Bill Talley ceased the nomination. Therefore Walter Blood is the new President of the KC 99'er Users Group. Next come vice-president and John Dilly was nominated. He too was elected to continue his office of vice-president. Then came the secretary. Steven DeGeare was nominate by Gary Burns and nomination ceased. The secretary elected for 1988 was Steven Debeare. Next came the treasurer. A recent new member Tom was nominated and he was elected as our new treasurer.

The issue of our next Swap-N-Shop was next on the agenda and a discussion arosed as too how many and when the next one shall be. The members chose the second sunday of April 1988 for the date of it with the option to run another later on if the members decide so.

The final item was the sponsership of our BBS the KC 99'ER operated by Gary Burns. A group of 4 members: Walter, Gary, Bill and Frank were chosen to bring a proposal before the membership no later than the FEB meeting. The meeting was adjorned, then we had a break before our Xbasic class began its session.

sincerely Steven (secretary)

'TURNING WHITE'
from WORDPLAY Newsletter

This neat little program comes to us from the Twin Tiers User Group. It is easy to type in. Just load this program and it will remain in memory until you turn off the computer. It will be especially handy for debugging when you type in programs as it will distinguish the letters apart from the numbers and punctuation. I've tried it (so has this editor) and find that there is less chance of error when typing in long CALL Characters codes.

You can keep it separate or use as a merge file with low numbers.

(I myself will keep it on my Utilities Diskette -- editor)

```
10 REM Turns all numeral and punctuation white by Harry Wilhelm
20 REM Turn on by CALL LOAD(-31804,63) Turn off by CALL LOAD
   (-31804,0)
30 CALL INIT
40 CALL LOAD(16126,2,224,38,0,2,0,8,17,2,1,63,36,2,2,0,3,4,32,32,36,
   2,224,131,192,3,128)
50 CALL LOAD(16164,240,240,240)
60 CALL LOAD(-31804,63)
```

TIPS FROM THE TIGERCUB

#45

Copyright 1987

TIGERCUB SOFTWARE
156 Collingwood Ave.
Columbus, OH 43213

Distributed by Tigercub Software to 11-99/4A 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 150 original programs in basic and extended basic, available on cassette or disk, now reduced to just \$2.00 each, plus \$1.50 per order for cassette or disk and PPA. Cassette programs will not be available after my present stock of blanks is exhausted.

Descriptive catalogs, while they last, \$1.00 which is deductible from your first order.

Tigercub Full Disk Collections, reduced to \$10 postpaid. Each of these contains either 5 or 6 of my regular \$2 catalog programs, and the remaining disk space has been filled with some of the best public domain programs of the same category. I am now 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, NUNE GAMES, MUND GAMES, ELEMENTARY MATH, MIDDLE/HIGH SCHOOL MATH, VOCABULARY AND READING, MUSICAL EDUCATION, KALEIDOSCOPIES AND DISPLAYS

MUTS & BULTS (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.

MUIS & MULLS 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.

XXXXXXXXXXXXXXXXXXXXXXXXXXXX
* MUIS & MULLS NO 15 NOW *
* ready, another full disk *
* of 140 new merge-format *
* utility subprograms, all *
* compatible with the pre- *
* vious. With 11 pages of *
* documentation. \$15 odd. *
XXXXXXXXXXXXXXXXXXXXXXXXXXXX

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.

Here is a versatile printer utility which will accept all printer control codes, print in 1 to 5 columns with choice of column separation and margin width, allow alternate margins and pause at end of page to turn paper over, and will load and print a diskfull of

files one after another. It is set up for the Gemini 10X and may require modification for other printers.

100 DIM M\$(400),F\$(50)
110 GOTO 150
120 K,ST,SET,S,P\$,P,CL,DW\$,S
S\$,I\$,D\$,E\$,NC,CW,TC,TA,TX,A
V,CS,S\$,LT,A\$,LSP,LP,RA,OK\$,
M\$,X,F\$(),SL,F,IP,M\$(),T\$,F
LAG,J,PP,LTS
130 CALL CLEAR :: CALL KEY :
: CALL COLOR :: CALL SCREEN
:: CALL SOUND
140 !@P-
150 CALL CLEAR :: CALL KEY(3
,K,ST):: ON MARKING NEXT
160 FOR SEI=0 TO 14 :: CALL
COLOR(SET,2,B):: NEXT SET ::
CALL SCREEN(1)
170 DISPLAY AT(3,6):"TIGERCUB
& PRINTALL":TAB(7):"Copyri
ght 1987":TAB(6):"Tigercub S
oftware" !programmed by Jim
Peterson
180 DISPLAY AT(12,1):"May be
distributed without":"restr
iction providing that":"no p
rice or copying fee is":"cha
rged."

190 DISPLAY AT(18,7):"TURN P
RINTER ON!"
200 DISPLAY AT(20,8):"PRESS
ANY KEY" :: DISPLAY AT(20,8)
:"press any key" :: CALL KEY
(0,K,S):: IF S=0 THEN 200 CL
SE CALL CLEAR
210 DISPLAY AT(12,1):"PRINTE
R DESIGNATION?" :: ACCEPT AT
(14,1)BEEP:P\$:: IF PUS(P\$,
".LF",1)=0 THEN P\$=P\$&".LF"
220 ON ERROR 230 :: OPEN #1:
P\$,VARIABLE 255 :: ON ERRUR
STOP :: PRINT #1:CHR\$(27);"E
" :: CALL CLEAR :: GOTO 240
230 DISPLAY AT(20,1):"CANNOT
OPEN PRINTER!" :: RETURN 21
0

240 DISPLAY AT(12,1):"PRINT
SIZE?": (1) PICA:" (2)
ELITE":(3) CONDENSED"
250 ACCEPT AT(12,13)VALIDATE
("123")SIZE(1):P :: PRINT #1
:CHR\$(27);"B";CHR\$(P);
260 !The values 80, 96 and 1
56 in the next line are the
maximum number of pica, elit
e and condensed characters p
er line on Gemini 10X

270 !Change as necessary for
your printer!
280 CL=(P=1)*80+(P=2)*96+(P=3)*136 :: CL=ABS(CL)
290 DISPLAY AT(12,1)ERASE AL
L:"DOUBLE-WIDTH? (Y/N) N" ::
ACCEPT AT(12,21)SIZE(-1)VAL
IDATE("YM")BEEP:DW\$:: IF DW
\$="Y" THEN PRINT #1:CHR\$(27)
;"M";CHR\$(1);:: CL=CL/2
300 DISPLAY AT(12,1)ERASE AL
L:"SUPERSCRIPT? (Y/N) N" ::
ACCEPT AT(12,20)SIZE(-1)VALI
DATE("YM")BEEP:SS\$:: IF SS\$
="Y" THEN PRINT #1:CHR\$(27);
"S";CHR\$(0);
310 DISPLAY AT(12,1)ERASE AL
L:"ITALICS? (Y/N) N" :: ACCE
PT AT(12,16)VALIDATE("YM")SI
ZE(-1)BEEP:I\$:: IF I\$="Y" T
HEN PRINT #1:CHR\$(27);"4";
320 DISPLAY AT(12,1)ERASE AL
L:"DOUBLE-STRIKE? (Y/N) Y" ::
ACCEPT AT(12,22)VALIDATE("Y
M")SIZE(-1)BEEP:O\$:: IF O\$
="Y" THEN PRINT #1:CHR\$(27);
"6";
330 IF P<3 AND P<4 THEN DI
SPLAY AT(12,1):"EMPHASIZED?
(Y/N) Y" :: ACCEPT AT(12,19)
VALIDATE("YM")SIZE(-1)BEEP:E
\$:: IF E\$="Y" THEN PRINT #1
:CHR\$(27);"E";
340 DISPLAY AT(12,1)ERASE AL
L:"NUMBER OF COLUMNS? (1-5)"
:: ACCEPT AT(12,26)VALIDATE
("12345")SIZE(1)BEEP:NC
350 DISPLAY AT(12,1):"COLUMN
WIDTH (NUMBER OF": "CHARAC
TERS?" :: ACCEPT AT(14,13)VA
LIDATE(DIGIT)BEEP:CW
360 TC=NC&CW :: TA=CL-TC ::
TX=TC+NC&2-2
370 IF TX<=CL THEN 390 :: DI
SPLAY AT(18,1):STR\$(NC)&" co
lumn of "&STR\$(CW)&" charac
ters":"plus 2-column spacing
equals"
380 DISPLAY AT(20,1):STR\$(TC
)&" characters; maximum":"av
ailable in print size":"sele
cted is "&STR\$(CL)&":"****
Please reselect****" :: GOTO
240
390 AV=INT(TA/(NC-1)): DIS
PLAY AT(12,1)ERASE ALL:"COLUM
N SEPARATION?":"MINIMUM 2":"
MAXIMUM "&STR\$(AV)&" AVAILAB
LE ":"2"
400 ACCEPT AT(15,1)VALIDATE(

```

DIGIT)SIZE(-2)BEEP:CS :: IF
CS<2 OR CS>AV THEN 32/67 ELS
E SS=NP(S(" ",CS)
410 IA=IA-CS*(NC-1):: IF TAK
Z THEN 430
420 DISPLAY AT(12,1)ERASE AL
L:"LEFT MARGIN WIDTH?": "MA
XIMUM "&SIX*(IA)&" AVAILABLE
" :: ACCEPT AT(12,2)VALIDATE
E(DIGIT)BEEP:LI :: IF LI>IA
THEN 420
430 DISPLAY AT(12,1):"ALTERN
ATING LEFT/RIGHT": "MARGIN?
(for pages to be:"later re
produced on both":sides) (Y
/N) "
440 ACCEPT AT(16,14)VALIDATE
("Y")SIZE(-1):AS
450 LSP=12 :: DISPLAY AT(10,
1):" ": " ": "LINES PER PAGE?
60": " " " " " " :: ACCEPT
AT(12,17)VALIDATE(DIGIT)SI
ZE(-3):LP :: IF LP<0 THEN 4
90
460 DISPLAY AT(12,1):"LINE S
PACING - 72 INCH": :: DISPLAY
AT(11,16):" " " " :: ACCEPT AT
(10,16)VALIDATE(DIGIT)BEEP:L
SP
470 IF LP/(INT(72/LSP))>11.5
THEN DISPLAY AT(20,1):"WON'
T FIT!" :: GOTO 450
480 PRINT #1:CHR$(27);"A";CH
R$(LSP):
490 KN=IA-LT
500 DISPLAY AT(12,1)ERASE AL
L:SIX*(NC)&" columns of":SIX
*(LN)&"-character width": "le
ft margin of "&SIX*(LI)&" sp
aces"
510 DISPLAY AT(15,1):SIX*(LP
7)&" lines per page": "with "&
SIX*(LSP)&"72 line spacing"
520 DISPLAY AT(17,1):SIX*(CS
7)&" spaces between columns":
"front margin of "&SIX*(KM)&
" spaces": "UK? (Y/N) Y"
530 ACCEPT AT(20,11)VALIDATE
("Y")SIZE(-1)BEEP:UKS :: IF
UKS="N" THEN 240
540 DISPLAY AT(12,1)ERASE AL
L:"PAUSE AT END OF PAGE? N"
:: ACCEPT AT(12,2)VALIDATE(
"Y")SIZE(-1):UUS
550 DISPLAY AT(1,1)ERASE ALL
:"INPUT FILENAMES TO BE": "PR
INTED.": "PRESS ENTER WHEN DU
NE"
560 IX=I+1 :: DISPLAY AT(IX+3,
1):"FILENAME? DSK" :: ACCEPT

```

```

AT(IX+3,14)SIZE(-12)BEEP:F$(
X)
570 IF F$(X)=" THEN X=X-1 :
: GOTO 600 ELSE F$(X)="DSK"&
F$(X)
580 ON ERROR 590 :: OPEN #2:
F$(X):: CLOSE #2 :: GOTO 560
590 ON ERROR STOP :: CALL 50
UND(1000,110,0,-4,0):: DISPL
AY AT(20,1):"CANNOT OPEN "F
$(X):: X=X-1 :: RETURN 560
600 SL=1
610 F=F+1 :: IF F>X THEN 700
:: ON ERROR 620 :: OPEN #2:
F$(F),INPUT :: DISPLAY AT(22
,1):"READING ":F$(F):: ON ER
ROR STOP :: GOTO 630
620 CALL SOUND(1000,110,0,-4
,0):: DISPLAY AT(20,1):"COUL
D NOT OPEN "&F$(F):: STOP
630 FOR IP=SL TO LP*NC :: LI
NPUT #2:M$(IP):: IF LEN(M$(I
P))=0 THEN 670 :: IF NC>1 AN
D POS(M$(IP),CHR$(13),1)<>0
THEN M$(IP)=SEG$(M$(IP),1,LE
N(M$(IP))-1)
640 IF ASC(M$(IP))>126 OR AS
C(M$(IP))<32 THEN IP=IP-1 ::
GOTO 680
650 IF LEN(M$(IP))<CN THEN
670 :: M$(IP)=SEG$(M$(IP),1,
CN):: CALL SOUND(1000,110,0,
-4,0):: DISPLAY AT(12,1):M$(
IP):" OVER";CN;"CHARACTERS":
"TRUNCATED TO ":F$(X)"
660 CALL KEY(3,K,S):: IF S=0
THEN 660 ELSE IF K<>89 THEN
STOP
670 M$(IP)=M$(IP)&RPTS(" ",C
N-LEN(M$(IP)))
680 IF EOF(2)=1 THEN CLOSE #
2 :: SL=IP+1 :: GOTO 610
690 NEXT IP :: IF EOF(2)=1 T
HEN CLOSE #2 :: GOTO 720 ELS
E GOTO 720
700 ON ERROR 710 :: FLAG=1 :
: FOR J=IP+1 TO NC:LP :: M$(
J)=" :: NEXT J :: GOTO 720
710 STOP
720 PP=PP+1 :: IF PP/2=INT(PP
/2)AND AS="Y" THEN LT=RPT$(
" ",KN)ELSE LT=RPTS(" ",LT
)
730 FOR J=1 TO LP :: ON NC 6
OSUB /50,760,770,780,790 ::
NEXT J :: PRINT #1:CHR$(12):
: SL=1 :: IF F>X THEN STOP E
LSE IF UUS="N" THEN 630
740 DISPLAY AT(24,1)BEEP:"PR
ESS ANY KEY TO CONTINUE" ::

```

```

CALL KEY(0,K,S):: IF S=0 THE
N 740 ELSE DISPLAY AT(24,1):
" " :: GOTO 630
750 PRINT #1:LT&M$(J)&CHR$(
10):: RETURN
760 PRINT #1:LT&M$(J)&S&M$(
J+LP)&CHR$(10):: RETURN
770 PRINT #1:LT&M$(J)&S&M$(
J+LP)&S&M$(J+LP*2)&CHR$(10
):: RETURN
780 PRINT #1:LT&M$(J)&S&M$(
J+LP)&S&M$(J+LP*2)&S&M$(J
+LP*3)&CHR$(10):: RETURN
790 PRINT #1:LT&M$(J)&S&M$(
J+LP)&S&M$(J+LP*2)&S&M$(J
+LP*3)&S&M$(J+LP*4)&CHR$(10
):: RETURN
This is an improved
version of the math program
in TIPS #36.
100 CALL CLEAR :: RANDOMIZE
110 B=INT(5*8RD+2):: IF B=B2
THEN 110 ELSE B2=B
120 F=INT(5*8RD+2):: IF F=F2
THEN 120 ELSE F2=F
130 D=INT(5*8RD+2):: IF D=D2
THEN 130 ELSE D2=D
140 X=F*B*B*D
150 BB=INT(5*8RD+2):: IF BB=
BB2 OR BB=B THEN 150 ELSE BB
2=BB
160 DD=INT(5*8RD+2):: IF DD=
DD2 OR DD=D THEN 160 ELSE DD
2=DD
170 F=F*BB*DD
180 DISPLAY AT(3,1)ERASE ALL
:"IF";B;"BOYS CAN CATCH";X;"
FROGS IN";D;"DAYS,"
190 DISPLAY AT(6,1):"NOW MAN
Y FROGS CAN";BB;"BOYS": "CATC
H IN";DD;"DAYS?"
210 ACCEPT AT(7,19):M
220 IF M=F THEN DISPLAY AT(9
,1):"THAT'S RIGHT!": :: GOTO
110
230 DISPLAY AT(9,1):"NO, THA
T'S WRONG."
240 DISPLAY AT(11,1):"IF";B;
"BOYS CAN CATCH";X;"FROGS IN
";D;"DAYS"
250 DISPLAY AT(13,1):"THEN O
NE BOY CANCATCH";X/B;"FROGS
IN";D;"DAYS"
260 DISPLAY AT(15,1):"AND ON
E BOY CAN CATCH";X/B/D;"FROG
S IN ONE DAY."
270 DISPLAY AT(17,1):"SO, IF
ONE BOY CAN CATCH";X/B/D;"F

```

```

ROGS IN ONE DAY,"
280 DISPLAY AT(19,1):"THEN";
BB;"BOYS CAN CATCH";X/B/D*BB
;"FROGS IN ONE DAY"
290 DISPLAY AT(21,1):"AND";B
B;"BOYS CAN CATCH";X/B/D*BB*
DD;"FROGS IN";DD;"DAYS."300
DISPLAY AT(24,1):"PRESS ANY
KEY" :: CALL KEY(0,K,S):: IF
S=0 THEN 300 ELSE 110

```

Here's an idea for an unusual title screen -

```

100 CALL CLEAR :: FOR SET=1
TO 8 :: CALL COLOR(SET,1,1):
: NEXT SET :: CALL CHAR(100,
"0",101,"0")
110 X$(0)="4043241818244202"
:: X$(1)="4021261818648402"
:: X$(2)="2020131C38C80404"
:: X$(3)="1010101FF8080808"
:: X$(4)="081010907E111020"
120 X$(5)="080808FB1F101010"
:: X$(6)="0404C8381C132020"
:: X$(7)="0284641818262140"
130 AS=RPT$(CHR$(100)&CHR$(1
01),13):: FOR R=1 TO 24 :: C
=C+1+(C=2)*2 :: DISPLAY AT(R
,C):AS :: NEXT R
140 CALL VCHAR(1,29,1,168)
150 CALL SCREEN(2):: CALL CO
LOR(9,5,16):: FOR S=1 TO 8 :
: CALL COLOR(S,16,2):: NEXT
S
160 DISPLAY AT(5,5):" TIGERC
UB SOFTWARE ":::: DISPLAY AT(
8,6):" SQUIRMY SCREEN "
170 FOR J=0 TO 7 :: CALL CHA
R(100,X$(J)):: CALL CHAR(101
,X$(7-J)):: NEXT J
180 CALL KEY(0,K,S):: IF S=0
THEN 170

```

MEMORY FULL

Jim Peterson

THE BLOOD BANK

Walter H. Blood
2032 North 32nd Street
Kansas City, Kansas 66104

JANUARY 1988

```

100 DIM BD$(15,15),CM$(13),DM$(2),N$(
15)
110 CALL CHAR(136,"00/F40404F48484800
FF0000FF00000000FE0202F2121212"&RPT$(
"48",8))
120 CALL CHAR(128,"00/E7E7E7E7E7E7E0000
00242424240000")
130 CALL CHAR(140,RPT$("12",8)&"48484
84F4040/F00000000FF0000FF00121212F202
02FE00")
140 BK$=CHR$(128):: CK$=CHR$(129):: P
$=CHR$(46):: SP$=CHR$(32)
150 CU$=CHR$(8)&CHR$(9)&CHR$(11)&CHR$(
10)
160 UL$=RPT$(CHR$(75),4):: DM$(1)="AC
RUSS" :: DM$(2)=SP$&"DOWN"&SP$
170 FOR L=1 TO 15 :: N$(L)=STR$(L)::
IF L<10 THEN N$(L)=SP$&N$(L)
180 NEXT L :: FOR L=1 TO 13 :: READ C
M$(L):: NEXT L
190 GOSUB 1190
200 CALL CLEAR :: DISPLAY AT(3,6):"CR
OSSWORD PUZZLER":::"1> Create your
own crossword":::"2> Work on existing
puzzle"
210 DISPLAY AT(10,1):"3> Create a puz
zle using":TAB(4):"Family Computing d
ata":::"4> Solve a puzzle":::"5> Quit
"
220 DISPLAY AT(18,7)BEEP:"Selection:"
230 GOSUB 970 :: DISPLAY AT(18,18):K$
:: IF K<49 OR K>53 THEN CALL SOUND(1
00,400,0):: GOTO 230 ELSE MS=VAL(K$)
240 CALL CLEAR :: ON MS GOTO 250,250,
690,260,950
250 L$="BAEDPS" :: W=0 :: IF MS=2 THE
N 270 ELSE GOSUB 990 :: GOTO 380
260 L$="EASDP" :: W=1
270 CALL CLEAR :: PRINT "What is the
name":"of the puzzle?"::"DSK"::: ACC
EPT AT(24,4)SIZE(12)BEEP:F$ :: PRINT
F$
280 IF F$="" THEN 200 ELSE F$="DSK"&F
$
290 GOSUB 1000 :: PRINT :::"Loading p
uzzle" :: OPEN #1:F$,INTERNAL,INPUT :
: INPUT #1:I :: IF W=0 AND I=1 THEN 3
40
300 FOR Y=1 TO 15 :: FOR X=1 TO 15 ::
INPUT #1:BD$(X,Y):: NEXT X

```

```

310 NEXT Y :: CLOSE #1 :: IF W=0 OR T
=1 THEN 380
320 FOR X=1 TO 15 :: FOR Y=1 TO 15 ::
IF BD$(X,Y)<>BK$ THEN BD$(X,Y)=P$
330 NEXT Y :: NEXT X :: GOTO 390
340 CLOSE #1 :: CALL CLEAR :: CALL SO
UND(100,400,0)
350 PRINT TAB(11):"Sorry!"::"This fi
le is not loadable"::"from Menu Choi
ce 2."
360 PRINT "Select <4> Solve a puzzle
"::"to load it."::
370 GOSUB 1010 :: GOTO 270
380 IF MS<4 THEN CX,CY=1 :: GOTO 410
390 CX,CY=0 :: FOR Y=1 TO 15 :: FOR X
=1 TO 15 :: IF BD$(X,Y)=P$ THEN CX=X
:: CY=Y :: X,Y=15
400 NEXT X :: NEXT Y :: IF CX=0 THEN
BD$(1,1)=P$ :: CX,CY=1
410 CALL CLEAR :: GOSUB 1160
420 FOR L=1 TO 15 :: CALL HCHAR(1,5+L
,64+L):: NEXT L
430 FOR Y=1 TO 15 :: DISPLAY AT(2+Y,1
)SIZE(2):N$(Y)
440 FOR X=1 TO 15 :: CALL HCHAR(2+Y,5
+X,ASC(BD$(X,Y))): NEXT X :: NEXT Y
450 SK=2 :: T=4
460 FOR L=1 TO 13 :: T=T+1 :: IF L=5
AND MS=4 THEN T=T-1 :: GOTO 470 ELSE
DISPLAY AT(T,20):CM$(L)
470 NEXT L :: DISPLAY AT(19,4):"Use <
+ctn>Arrow": " keys to move cursor"
480 DISPLAY AT(2,22):N$(CY)&CHR$(64+C
X):: DISPLAY AT(3,21):DM$(SK/2)
490 CALL HCHAR(2+CY,5+CX,32):: CALL H
CHAR(2+CY,5+CX,129):: CALL HCHAR(2+CY
,5+CX,ASC(BD$(CX,CY)))
500 CALL KEY(S,K,S):: IF S=0 THEN 490
ELSE K$=CHR$(K):: DK=POS(CU$,K$,1)
510 IF DK<>0 THEN GOSUB 1030 :: GOTO
480 ELSE IF K<129 THEN 670
520 K=K-64 :: K$=CHR$(K):: CK=POS(C$,
K$,1):: IF CK=0 THEN CALL SOUND(100,4
00,0):: GOTO 490
530 IF W=0 THEN ON CK GOTO 550,560,57
0,560,590,660
540 ON CK GOTO 570,560,660,560,590
550 K$=BK$ :: GOTO 680
560 SK=CK :: GOTO 480
570 CALL CLEAR :: CALL SOUND(200,400,
0):: PRINT "This puzzle has not been
"::"SAVED!"::
580 PRINT "Are you sure you want"::
"to EXIT?" :: GOSUB 1100 :: IF K$="Y"
THEN 200 ELSE 410
590 CALL CLEAR :: PRINT TAB(7):"Press
any key":TAB(3):"when printer is rea
dy."
600 CALL KEY(0,K,S):: IF S=0 THEN 600
ELSE OPEN #1:"F10" :: PRINT :::" F

```



```

finding puzzle ..."
610 PRINT #1: TAB(5);: FOR L=1 TO 15
  :: PRINT #1: CHR$(64+L)&SP$;: NEXT L
  :: PRINT #1
620 PRINT #1: TAB(3); RPT$( "#", 33)
630 FOR Y=1 TO 15 :: PRINT #1: N$(Y)&
  #"&SP$;
640 FOR X=1 TO 15 :: T$=BD$(X,Y):: IF
  T$<>CHR$(128) THEN PRINT #1: T$&SP$; EL
  SE PRINT #1: "*"&SP$;
650 NEXT X :: PRINT #1: "#"&N$(Y):: NE
  XT Y :: PRINT #1: TAB(3); RPT$( "#", 33):
  : CLUSE #1 :: GOTO 410
660 GOSUB 1110 :: IF F$="" THEN 410 E
  LSE 200
670 GOSUB 980 :: IF (K$<"A" OR K$>"Z"
  )AND K$<>P$ THEN CALL SOUND(100,400,0
  ): GOTO 490
680 BD$(CX,CY)=K$ :: CALL HCHAR(2+CY,
  5+CX,ASC(K$)): DK=5K :: GOSUB 1030 :
  : GOTO 480
690 CALL CLEAR :: GOSUB 990 :: D(SPLA
  Y AT(22,1): "Input Family Computing da
  ta" :: ZY=1 :: H$=UL$
700 FOR L=1 TO 4 :: H$=H$&","&UL$ ::
  NEXT L :: CD$=""
710 DISPLAY AT(1+ZY,1): CHR$(64+ZY)&SP
  $&H$ :: L$=RPT$(CHR$(95),20):: ZX=0
720 DISPLAY AT(17,1): "CENTER= move
  to next line": "CTRL E= exit w/o sav
  ing"
730 CALL HCHAR(1+ZY,5+ZX+INT(ZX/4),32
  ): CALL HCHAR(1+ZY,5+ZX+INT(ZX/4),12
  9): CALL HCHAR(1+ZY,5+ZX+INT(ZX/4),A
  SC(SEG$(L$,ZX+1,1)))
740 CALL KEY(0,K,S):: IF S=0 THEN 730
  ELSE K$=CHR$(K):: DK=PUS(CD$,K$,1)::
  IF DK<1 OR DK>2 THEN 760
750 ZX=ZX+(DK=1)-(DK=2):: ZX=ZX+((ZX=
  20)-(ZX=-1))*20 :: GOTO 730
760 IF K<129 THEN 780 ELSE K=K-64 ::
  GOSUB 980 :: IF K$="E" THEN CALL SOUN
  D(400,300,0):: GOTO 200
770 CALL SOUND(100,400,0):: GOTO 730
780 IF K$<>CHR$(13) THEN 930 ELSE GOSU
  B 1090
790 FOR L=LEN(L$)-3 TO LEN(L$):: Z=AS
  C(SEG$(L$,L,1)): IF Z<48 OR Z>57 THE
  N L$=SEG$(L$,1,LEN(L$)-4)&"0000" :: L
  =LEN(L$)
800 NEXT L :: CV=VAL(SEG$(L$,LEN(L$)-
  3,4)): CI=0
810 FOR L=1 TO 16 :: CI=CI+ASC(SEG$(L
  $,L,1)): NEXT L
820 IF CI=CV+(CV>5000)*5000 THEN 840
  ELSE CALL SOUND(150,300,0)
830 DISPLAY AT(17,1): "You have made a
  tvoo": "in line "&CHR$(64+ZY)&P$ :: G
  OSUB 1010 :: GOSUB 1090 :: ZX=0 :: GO
  TO 720

```

```

840 IF CV>5000 THEN 850 ELSE CD$=CD$&
  SEG$(L$,1,16):: ZY=ZY+1 :: IF ZY<14 T
  HEN 710 ELSE 870
850 T=16 :: FOR L=1 TO 15 STEP 2 :: I
  F SEG$(L$,L,1)="Z" THEN T=L-1 :: L=15
860 NEXT L :: CD$=CD$&SEG$(L$,1,T)
870 FOR L=1 TO LEN(CD$)-1 STEP 2 :: X
  $=SEG$(CD$,L,1):: Y$=SEG$(CD$,L+1,1)
880 BD$(ASC(X$)-64,ASC(Y$)-64)=BK$ ::
  NEXT L
890 GOSUB 1110 :: IF F$="" THEN CALL
  SOUND(100,300,0):: GOTO 890
900 CALL CLEAR :: PRINT "Press <M> fo
  r Menu or":;TAB(7);"<S> to solve puz
  zle"
910 GOSUB 970 :: IF K$<>"M" AND K$<>"
  S" THEN CALL SOUND(100,400,0):: GOTO
  910 ELSE IF K$="M" THEN 200
920 MS=4 :: W=1 :: C$="EASDP" :: GOTO
  410
930 IF (K$<"A" OR K$>"Z")AND(K$<"0" O
  R K$>"9") THEN CALL SOUND(100,400,0)::
  GOTO 730
940 CALL HCHAR(1+ZY,5+ZX+INT(ZX/4),K)
  :: L$=SEG$(L$,1,ZX)&K$&SEG$(L$,ZX+2,L
  EN(L$)-ZX-1):: ZX=ZX+1+(ZX=19):: GOTO
  730
950 CALL SOUND(300,300,0):: PRINT "Ar
  e you sure": "you want to quit?" ::
  GOSUB 1100 :: IF K$="N" THEN 200
960 CALL CLEAR :: END
970 CALL KEY(S,K,S):: IF S=0 THEN 970
980 K$=CHR$(K-(K>96)*(K<123)*32):: K=
  ASC(K$):: RETURN
990 FOR X=1 TO 15 :: FOR Y=1 TO 15 ::
  BD$(X,Y)=P$ :: NEXT Y :: NEXT X :: R
  ETURN
1000 CALL SOUND(100,400,0):: PRINT :
  "Please make sure disk": "is in disk dr
  ive and": "door is closed.": :: GOSU
  B 1010 :: RETURN
1010 DISPLAY AT(24,1)BEEP: "Press any
  key to continue";
1020 CALL KEY(3,K,S):: IF S=0 THEN 10
  20 ELSE RETURN
1030 UN DK GOTO 1040,1040,1060,1060
1040 CX=CX+(DK=1)-(DK=2)
1050 CX=CX+((CX=16)-(CX=0))*15
1060 CY=CY+(DK=3)-(DK=4)
1070 CY=CY+((CY=16)-(CY=0))*15
1080 IF BD$(CX,CY)=CHR$(128)AND W=1 T
  HEN 1030 ELSE RETURN
1090 CALL HCHAR(17,1,32,96):: CALL HC
  HAR(24,3,32,28):: RETURN
1100 GOSUB 970 :: IF K$<>"Y" AND K$<>
  "N" THEN CALL SOUND(100,400,0):: GOTO
  1100 ELSE RETURN

```

***** continued next month *****