August

J F M A J J A S O N D

BUG ◇ BYTES

TIBUG

TI BUG

143 888

AUGUST 1984

CONTENTS

NEXT MEETING

31st AUGUST   7.30 pm

CALEDONIAN CLUB

O'CONNELL ST.

KANGAROO POINT

SHOP will open at 7.30pm.

MEETING will start at 8.00pm.

PLEASE NOTE: Books and Tapes

will be sold before the meeting

ONLY.

If you don't pay your dues this month this will be the last newsletter you
receive, so don't forget to post them off righht away to make sure you don't
miss an issue. For your convenience, there is a lift out page in the middle
of this newsletter for renewing your membership, and also for nominations
for the committee. Please consider carefully if you could take a position
on the committee. The club cannot function properly without all the
positions filled, and some of the present committe members have been
working hard for your benefit for 2 years or more, and would like a break.
Also the club can only benefit from new members bringing new ideas to the
committee. Don't leave it to someone else to volunteer. Even if you have
only been a member for a short while and are worried that you don't know
enough about the computer, it is amazing how quickly you learn about it
when you are on the committee. The only qualifications you need are
willingness and a few hours of your time, so ...how about it?
My apologies are due again. Bugs got in the programs again last month(Is
that why we cal it BUG-BYTES). This time it was all the RND statements that
wer wrong. Elsewhere in the magazine I will put a list of all the
corrections.
My thanks to all the people who have contributed to the newsletter this
month. Keep up the good work! What we really need are more articles of a
technical nature. So if you can help, please let me know.

COMMITTEE

PRESIDENT	Greg LANE	264 1562
VICE PRESIDENT	Tony WALLEN	349 6157
SECRETARY	Noel WILSON	205 7407
TREASURER	Garry CHRISTENSEN	284 1841
ASST. SECRETARY	Jacob GROENVELD	343 8116
PROGRAM LIBRARIAN	Alan SAUNDERS	204 41.
BOOK LIBRARIAN	Mike BRYZAK	345 1912
EDITOR	Sandra NICHELSEN	341 5667
PROJECT OFFICER	Marcel ARIAS	203 6512

JOYSTICKS AND EDITTING DON'T MIX
----

by Garry Christensen

ALWAYS DISCONNECT YOUR JOYSTICKS WHEN EDITTING. Why? The answer came
to me in a rather startling way. An onlooker was watching me put the
final touches on my latest programme. With his arm resting on the joystick,
he watched anxiously as my creation was about to come to life.

One last adjustment! FLASH! BEEP! TITLE SCREEN!

Oh no, what happened? Had I accidently pressed "quit" instead of the arrow
key? No, that couldn't be possible.

After much experimenting, we discovered the cause. If you push the
joystick foreward and hold the fire button down then press the function
key, the computer "quits" as the function key is released. why this
happens I haven't the slightest idea but the moral of the story is:-
"Disconnect the joystick while editting."

Although it is a freak accident, it happened to me. Better safe than
sorry. Don't let your time and effort disappear in a keystroke!

THE PHOENIX TO HATCH SOON. from SYDNEY NEWSDIGEST August

Cor-Comp Inc. the California based company that says it has developed the
succesor to the TI-99/4A,is beginning to firm its plans for delivery of the
new machine, the 99000.
The company is dubbed the PHOENIX after the mythical bird of ancient Egypt.
Here is a list of features of the 99000...

The basic system is expected to cost between $600 and $700 U.S.

KEYBOARD: choice of three styles (myler, word-processing with numeric
keypad, and typewriter styles). Each will be priced differently.

MEMORY: 64K RAM expandable in 64K increments to ONE MEGABYTE.

DISK CONTROLLER: built in card, capable of controlling up to four double-sided , quad-density drives.

RS232: built-in RS232 port.

CARTRIDGES: two slots for cartridges, compatible with all TI cartridge software.

PROCESSOR SLOT: one slot for a second processor chip, such as the MOSTEK 6502 used in APPLE and ATARI computers, the INTEL 8088 chip used in the IBM-PC and the Z80A used in the OSBORNE and TIMEX computers.

GAMES PORTS: two game controller ports, one Atari compatible, the other Apple compatible.

NEW PERIPHERAL EXPANSION BOX

As you probably already know, there are no more of the TI expansion boxes available. Thats the bad news, now for the good news. Cor-Comp Inc, have announced that their new mini PERIPHERAL BOX is to be released this month (August) in Australia through IMAGIC. It will include Minibox with RS232 card ($375), which is the same as the older style in price, but MODULE #2 will be arriving shortly which consists of 32K card with Disk Controller on ONE card. This card will support DOUBLE SIDED, DOUBLE DENSITY DISK DRIVES ($?). The new set-up looks like costing $549.95 U.S. without the Disk Drives. What you get will be the box with built-in RS232 (2 serial and 1 parallel ports) 32K Expansion RAM and a disk controller which will handle up to 4 (yes four) Double sided Double-density Drives. The flex cable interface will be a round cable that will hook to the side of the 99/4A via a small Ltype connector....unlike the large flat cable with TIOs PE box. The power supply is specially designed for low heat and high power.

The size of the new 99000 expansion system is about 3/4 size of the TI box. We understand that the new box power supply is capable of handling a Hard Disk Drive.

The formatted DS/DDdisk will store 360K bytes of information. With 4 DS/DD Drives hooked up.... the storage power adds up to 1.4 Megabytes of on-line storage... and that is without a Hard Disk!

The Disk Manager program will come on a 5 1/4" floppy diskette and will have many new enhancements added to it over the Disk Manager cartridge. The Disk

Controller also adds the following new commands and programming statements to the computer....

CALL PEEK(address,v,v,...v) peeks into CPU RAM
CALL POKE(address,dv,dv,..dv) pokes into CPU RAM.
CALL PEEKV(address,v,v,v,..v) peeks into VDP RAM.
CALL POKEV(address,dv,dv,..dv) pokes into VDP RAM.
CALL MGR loads the disk manager program.
CALL EXEC(address) Executes machine language code in ROM or RAM.
CALL MOVEM(type#1-4, from address, to address,# of bytes to move) This will move blocks of memory from one location to another. Type #1 - 4 can be numeric variable or direct number:
1=Move from VDP RAM to VDP RAM
2=Move from VDP RAM to CPU RAM
3=Move from CPU RAM or ROM to VDP RAM
4=Move from CPU RAM or ROM to CPU RAM

NOTE: v can be a numeric variable or string variable. d can be a direct number or string and address can be a variable or direct number. Single dimension numeric and string arrays may be used. Numerics and strings can be mixed in the same CALL.

LASTLY:this Disk controller has provisions for setting the head seek (step) times for all four of your drives. This will allow you to use some of the faster disk drives that are available.

SIZE and COSTS:
You may start out with just the RS232 and add the 32K Memory and Disk Controller chips later. This unit is approx 5 1/2" x 5" x 3" or about the size of 2 Speech Synthesizers. The disk Controller portion will have the same features as described above. SRP for the RS232 alone will be $149.95 U.S. With the 32K Memory and Disk Controller the SRP will be $399.95 U.S. These units will be sold in Australia by IMAGIC (Australia).

ATTENTION ADVENTIURE GAME PLAYERS

Are you having trouble with one of your adventure games? Are you completely bamboozled? If so, help is at hand. Write to Humphrey Lindley at 127 Crowley St.., ASPLEY Q 403-., enclosing a stamped addressed envelope, and outlining your problem, and Humphrey will send you a clue as to what to do next. Humphrey is making no charge for this service, so don't forget to enclose your stamped addressed envelope, and please, NO PHONE CALLS.

RABID by Cullhane Gibbs

Here is a good game to play if you're feeling bloodthirsty. It was written by one of our members, Cullhane Gibbs, who is only 13 years old. What you have to do is defend yourself from the rabid knife-wielding maniacs. You are equipped with a flamethrower. Good luck!

```
100 REM *********************
110 REM *      RABID       *
120 REM *BY CULLHANE GIBBS *
130 REM *IN EXTENDED BASIC *
140 REM *JOYSTICKS REQUIRED*
150 REM *********************
160 CALL MAGNIFY(2)
170 RANDOMIZE
180 CALL CHAR(47,"002277FFFF
2A0000")
190 CALL CHAR(94,"123422256F
800451")
200 CALL CHAR(64,"3838107CBB
292A28")
210 CALL CHAR(124,"20502070A
8B0A844")
220 CALL CLEAR :: CALL SCREE
N(2)
230 FOR COLOUR=2 TO 12 :: CA
LL COLOR(COLOUR,COLOUR+1,2):
: NEXT COLOUR
240 PRINT "    RABID":"BY C
ULLHANE GIBBS":"NEEDS EXTEND
ED BASIC":"AND JOYSTICKS" ::
 PRINT :: PRINT "INSTRUCTION
S:"
250 PRINT "AVOID    RABID,":
"KNIFE WEILDING MANIACS.":"T
O DEFEND YOURSELF"
260 PRINT "PRESS YOUR FIREBU
TTON":"TO RELEASE A SHORT FL
AME":"WHICH WILL BURN YOUR":
"ATTACKERS.":"NEW SCREEN STA
RTS WHEN ALL"
270 PRINT "ATTACKERS ARE KIL
LED.":"ONCE YOU ARE KILLED T
HE":"GAME IS OVER.":"YOU CAN
 FIRE ONLY":"IN ONE DIRECTIO
N-":"TO THE LEFT."
280 PRINT "PRESS ANY KEY"
290 FOR D=10 TO 50 :: CALL S
OUND(D,701,0):: CALL SOUND(-
100,-8,0):: CALL SOUND(-50,-
4,10):: CALL SOUND(-100,-2,0
):: NEXT D
300 CALL KEY(0,KEP,SEP):: IF
 SEP=0 THEN 300 ELSE 310
310 CALL CLEAR
320 PRINT "SCORECHART:" :: P
RINT
330 PRINT "@ MANIAC=100" ::
PRINT :: PRINT "* YOU" :: PR
INT
340 PRINT "PRESS ANY KEY TO
BEGIN"
350 CALL SOUND(-1000,-8,0)
360 PRINT :: PRINT :: PRINT
:: PRINT :: PRINT :: PRINT :
: PRINT :: PRINT :: PRINT ::
 PRINT
370 CALL KEY(0,K,S):: IF S=0
 THEN 370 ELSE 380
380 CALL CLEAR :: PRINT "
GET READY,PLAYER!" :: PRINT
:: PRINT :: PRINT :: PRINT
:: PRINT :: PRINT :: PRINT :
: PRINT :: PRINT :: PRINT ::
 PRINT :: PRINT :: PRINT
390 FOR TIME=1 TO 500 :: NEX
T TIME
400 SCORE=0
410 CALL CLEAR
420 CALL SPRITE(#11,124,4,70
,200)
430 FOR SPRIT=1 TO 4 :: CALL
 SPRITE(#SPRIT,64,5,121,89):
```

```
: CALL MOTION(#SPRIT,INT(RND
*10)+1,-INT(RND*10)+1):: NEX
T SPRIT :: MANIC=4
440 DISPLAY AT(1,3):"SCORE:"
;SCORE
450 CALL JOYST(1,X,Y):: CALL
 MOTION(#11,-Y*2,X*2):: CALL
 SOUND(-3,-3,0)
460 CALL POSITION(#11,YPOS1,
XPOS1)
470 CALL KEY(1,KE,ST)
480 IF ST=-1 AND XPOS1]24 TH
EN 490 ELSE 690
490 CALL SPRITE(#12,47,7,YPO
S1,XPOS1-16)
500 CALL COINC(#12,#1,20,A):
: IF A=-1 THEN 510 ELSE 540
510 CALL SOUND(1000,340,0)::
 CALL PATTERN(#1,94):: SCORE
=SCORE+100 :: MANIC=MANIC-1
:: CALL DELSPRITE(#1)
520 CALL DELSPRITE(#12)
530 IF MANIC=0 THEN 430 ELSE
 540
540 CALL COINC(#12,#2,20,A):
: IF A=-1 THEN 550 ELSE 590
550 CALL DELSPRITE(#12)
560 CALL SOUND(1000,340,0)::
 CALL PATTERN(#2,94):: SCORE
=SCORE+100 :: MANIC=MANIC-1
:: CALL DELSPRITE(#2)
570 CALL DELSPRITE(#12)
580 IF MANIC=0 THEN 430 ELSE
 590
590 CALL COINC(#12,#3,20,B):
: IF B=-1 THEN 600 ELSE 640
600 CALL DELSPRITE(#12)
610 CALL SOUND(1000,340,0)::
 CALL PATTERN(#3,94):: SCORE
=SCORE+100 :: MANIC=MANIC-1
:: CALL DELSPRITE(#3)
620 CALL DELSPRITE(#12)
630 IF MANIC=0 THEN 430 ELSE
 640
640 CALL COINC(#12,#4,20,C):
: IF C=-1 THEN 650 ELSE 700
650 CALL DELSPRITE(#12)
660 CALL SOUND(1000,340,0)::
 CALL PATTERN(#4,94):: SCORE
=SCORE+100 :: MANIC=MANIC-1
:: CALL DELSPRITE(#4)
670 CALL DELSPRITE(#12)
680 IF MANIC=0 THEN 430 ELSE
 700
690 CALL DELSPRITE(#12)
700 CALL COINC(#1,#11,16,T):
: IF T=-1 THEN 740 ELSE 710
710 CALL COINC(#2,#11,16,U):
: IF U=-1 THEN 740 ELSE 720
720 CALL COINC(#3,#11,16,V):
: IF V=-1 THEN 740 ELSE 730
730 CALL COINC(#4,#11,18,W):
: IF W=-1 THEN 740 ELSE 780
740 FOR DIP=1 TO 28 :: CALL
MOTION(#DIP,0,0):: NEXT DIP
:: CALL SOUND(-1000,-8,0)::
CALL PATTERN(#11,94):: FOR D
EL=1 TO 50 :: NEXT DEL
750 CALL DELSPRITE(#11):: DI
SPLAY AT(23,3):"GAME OVER-ST
ART AGAIN Y OR N" :: CALL KE
Y(0,P,S):: IF S=0 THEN 750 :
: IF P=ASC("n")THEN 770
760 IF P=ASC("y")THEN 400 EL
SE 750
770 END
780 GOTO 450
```

EXTENDED BASIC TUTORIAL with TONY MCGOVERN

from SYDNEY NEWS DIGEST

Part 2

```
100 DATA 1
110 READ X :: PRINT X :: READ X :: PRINT X
120 SUB NOTHING
130 DATA 2
140 SUBEND
```

When you RUN this program it makes no difference that the second data item is apparently located in the sub-program.IMAGEs behave likewise. On the other hand DEFed functions, if you care to use them, are strictly confined to the particular part of the program in which they are defined, be it main or sub. During the prescan DEFed names are kept within the allocatin process separately for each sub-program or the main program. Once again try a little programming to illustrate the point.

```
100 DEF X=1 :: PRINT X;Y :: CALL SP(Y) :: PRINT X;Y
110 SUB SP(Z) :: DEF X=2 :: Z=X :: DEF Y=3
120 SUBEND
```

This point is not explicitly made in the XB manual and has been the subject of misleading or incorrect comment in magazines and newsletters. A little reflection on how XB handle the details will usually clear up difficulties.

TI BASICs assign nominal values to all variables mentioned in a program as part of the prescan, zero for numeric and null for strings, unlike some languages (some Basics even) which will issue an error message if an unassigned variable is presumed upon. This means that XB can' work like TI LOGO which has a rule that if it finds an undefined variable within a procedure it checks the chain of CALLing procedures until it finds a value. However, unlike Pascal which erases all the information left within a procedure when it is finished with it , XB retains from CALL to CALL the values of variable entirely contained within the sub-program. The values of variables transferred into the sub-program through the SUB parameter list will of course take on their newly passed values each time the sub-program is CALLed. A little program will show the difference.

```
100 FOR I=1 TO 9 :: CALL SBPR(0):: NEXT I
110 SUB SBPR(A):: A=A+1 :: B=B+1 :: PRINT A;B
120 SUBEND
```

The first variable printed is reset to 0 each time SBPR is called, while the second, B, is incremented from its previous value each time. Array variables are stored as a whole in oone place in the program, within the main program or the sub-program in which the DIMension statement for the array occurs. XB doesn't tolerate attempts to re-dimension arrays, so information on arrays can only be passed down the chain of sub-programs in one direction. Any attempt by an XB sub-program to CALL itself, either directly or indirectly from any sub-program CALLed from the first, no matter how many times removed, will result in an error. Recursive procedures, an essential part of TI LOGO are NOT possible with XB sub-programs, since CALLing a sub-program does not set up a new private library of values.

All of this discussion of the behaviour of TI Extended Basic comes from with version 110 of XB on a TI-99/4A with 1981 title screen. Earlier versions and consoles are not common in Australia, but TI generally seems to take a lot of trouble to keep new versions of programs compatible with the old. On the other hand TI has also been very reticent about the details of how XB works. The Editor/Assembler manual has very little to say about it, less by far even than it tells about console Basic. I am not aware of any discussion of the syntax of the Graphics Programming Language (GPL), let alone of the source code for the GPL interpreter which resides in the console ROM of every TI-99/4A.

Another simple programming experiment will demonstrate what we mean by saying that XB sets up separate Basic programs for each sub-program. RUN the following:

```
100 X=1 :: CALL SBPR :: BREAK
110 SUB SBPR :: X=2 :: BREAK
120 SUBEND
```

When the program BREAKs examine the value of variable X by entering the command PRINT X, and then CONtinue to the next program BREAK, which this time will be in the main program,where you can once again examine variable values.

We will now summarize the properties of XB sub-programs as procedures in complete XB programs, leaving the details of joining up the various procedures to the next section.

(a) XB treats each sub-program as a separate program, building a distinct table of named(REFed) and DEFed variables for each.

(b) All DATA statements are treated as being in a common pool equally accessible from all sub-programs or the main program as are also IMAGE statements, CHARacters, SPRITEs,COLORs, and File specifications.

(c) All other informationis passed from the CALLing main or sub-program by the parameter lists in CALL and SUB statements. XB does not provide for declaration fo common variables available on a global basis to all sub-programs as can be done in some languages.

(d) Variable values confined within a sub-program are static, and preserved for the next time the sub-program is CALLed. Some languages such as Pascal delete all traces of a procedure after it has been used.

(e) XB sub-programs may not CALL themselves directly or indirectly in a closed chain. Subject to this restriction a sub-program may be called from any other sub-program.

(f) The MERGE command available in XB with a disk system (32K memory expansion optional) allows a library of XB sub-programs to be stored on disk and incorporated as needed in other programs.

NEXT MONTH, TONY CONTINUES HIS SPECIAL EXTENDED TUTORIAL SERIES, with SUBPROGRAM PARAMETER LISTS etc.

COMPETITION

PRIZES: MICROSURGEON, MOONSWEEPER, SUPER DEMON ATTACK, and FATHOM.

These are the very latest Imagic cartridges to arrive here in Australia each with a retail price of $29.95

In conjunction with IMAGIC (Australia) and TI.S.H.U.G (Sydney User Group) you have the opportunity to win a complete set of these games. There are 5 sets to be given away... 3 sets from IMAGIC and 2 sets from TI.S.H.U.G.

All you have to do is.... Purchase RETURN TO PIRATE'S ISLE then complete the game listing each and every move 'till it's completed.

SEND YOUR ENTRIES TO EITHER:

PIRATE'S ISLE COMPETITION   or   TI.S.H.U.G. COMPETITION
IMAGIC (AUSTRALIA)   P.O. BOX 595,
93 SOUTH CREEK ROAD,   MARRICKVILLE.
DEE WHY. N.S.W. 2099   N.S.W. 2204

KIDS KORNER

Here's a program that allows you to draw with the joystick, or if you prefer, you can use the U,D,L,R keys. It comes to us from one of our country members, Ian Smith from Mt Isa. This is one the kids will enjoy playing around with.

```
100 REM ******DRAWING******
110 REM ***BY LEN TURNER***
120 REM ***MODIFIED FOR ***
130 REM ***JOYSTICK NO 1***
140 REM ******** BY ******
150 REM *** IAN J SMITH ***
160 REM **** T.I.B.U.G. ***
170 REM  CHANGE LINE 270 TO
CALL KEY(O,Z,X)
180 REM  LINE 280 TO Z=85
190 REM  LINE 360 TO Z=68
200 REM  LINE 440 TO Z=82
210 REM  LINE 520 TO Z=76
220 REM  TO USE U,D,L,R KEYS
230 CALL CLEAR
240 CALL CHAR(128,"FFFFFFFF
FFFFFFFF")
250 R=1
260 C=1
270 CALL JOYST(1,X,Y)
280 IF Y=4 THEN 300
290 GOTO 360
300 R=R-1
310 IF R[1 THEN 330
320 GOTO 350
330 R=1
340 GOTO 270
350 GOTO 610
360 IF Y=-4 THEN 380
370 GOTO 440
380 R=R+1
390 IF R]24 THEN 410
400 GOTO 430
410 R=24
420 GOTO 270
430 GOTO 610
440 IF X=4 THEN 460
450 GOTO 520
460 C=C+1
470 IF C]32 THEN 490
480 GOTO 510
490 C=32
500 GOTO 270
510 GOTO 610
520 IF X=-4 THEN 540
530 GOTO 600
540 C=C-1
550 IF C[1 THEN 570
560 GOTO 590
570 C=1
580 GOTO0 270
590 GOTO 610
600 GOTO 270
610 CALL HCHAR(R,C,128)
620 GOTO 270
630 REM  IT WORKS IN EITHER
640 REM  BASIC OR EXT. BASIC
650 REM  CHANGE LINES 250,26
0
660 REM  TO R=12,C=14 TO STA
RT
670 REM  IN CENTRE OF SCREEN
```
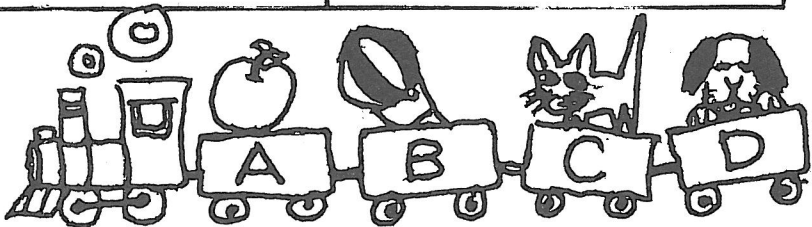
ALPHABET RECOGNITION

This program was written by L.K.TUTCHINGS to help his son learn the alphabet. It proved to be a big hit at his pre-school too. I think that all the pre-schoolers out there will love it too. It requires Extended Basic and if you have a speech synthesizer, you will also get speech with it.

```
100 CALL SCREEN(8)
110 FOR COL==3 TO 8 :: CALL C
OLOR(COL,2,1):: NEXT COL
120 DISPLAY AT(4,4)ERASE ALL
:"1 ALPHABET RECOGNITION" ::
 DISPLAY AT(6,4):"2 ALPHA AT
TACK"
130 DISPLAY AT(8,4):"3 CLOSE
 OF PROGRAM" :: DISPLAY AT(1
8,2):"PUSH No KEY OF YOUR CH
OICE"
140 CALL KEY(0,K,S):: IF S=0
 THEN 140 :: IF K=ASC("1"")TH
EN 150 :: IF K=ASC("2")THEN
470 :: IF K=ASC("3")THEN 850
 ELSE 140
150 DISPLAY AT(12,4)ERASE AL
L:"ALPHABET RECOGNITION" ::
FOR DE=1 TO 300 :: NEXT DE
160 CALL CLEAR
170 PRINT "THE IDEA IS TO PR
ESS THE KEY ON THE KEYBOARD
THAT       MATCHES THE LETTE
R THAT IS  GOING ACROSS THE
SCREEN"
180 PRINT "THE COMPUTER WILL
 LET YOU   KNOW IF IT IS COR
RECT. IF IT IS CORRECT THEN
ANOTHER    LETTER IS RANDOML
Y SELECTED"
190 PRINT "UNTIL YOU HAVE CO
RRECTLY GOT 40 RIGHT": : :"P
RESS ANY KEY TO START"
200 CALL KEY(0,K,S):: IF S=0
 THEN 200
210 CALL CLEAR
220 RANDOMIZE
230 FOR A=1 TO 40
240 X=INT(RND*25)+65
250 CALL SCREEN(2)
260 CALL MAGNIFY(2)
270 FOR C=5 TO 8
280 CALL COLOR(C,16,2):: NEX
T C
290 FOR Q=4 TO 25 STEP 4
300 DISPLAY AT(24,Q):CHR$(X)
:: DISPLAY AT(1,Q):CHR$(X)::
 NEXT Q
310 CALL SPRITE(#1,X,16,86,2
0,0,10)
320 CALL SAY(CHR$(X))
330 CALL KEY(0,K,S):: IF S=0
 THEN 330 :: IF K[]X THEN 36
0 ELSE 390
340 CALL DELSPRITE(#1):: NEX
T A
350 GOTO 100
360 DISPLAY AT(20,7):"WRONG
TRY AGAIN!"
370 CALL SAY("UHOH.THAT IS N
OT RIGHT, TRY AGAIN")
380 DISPLAY AT(20,7)SIZE(16)
:"               " :: GOTO
310
390 DISPLAY AT(20,12)SIZE(5)
:"RIGHT"
400 Z=INT(RND*5)+1 :: ON Z G
OTO 410,420,430,440,450
410 CALL SAY("GOOD WORK,GO S
OME MORE"):: GOTO 460
420 CALL SAY("THAT IS CORREC
T, CAN YOU DO IT AGAIN"):: G
```
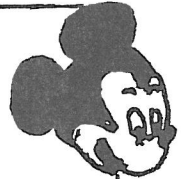
```
OTO 460
430 CALL SAY("RIGHT. GO AGAI
N")::  GOTO 460
440 CALL SAY("GOOD, WHY STOP
  NOW")::  GOTO  460
450 CALL SAY("YES.GO AGAIN")
460 DISPLAY AT(20,12)SIZE(5)
:"     " ::  GOTO 340
470 CALL CLEAR
480 DISPLAY AT(12,8):"ALPHA
ATTACK" ::  DISPLAY AT(20,2):
"WANT INSTRUCTIONS Y OR N?"
490 CALL KEY(0,K,S)::  IF S=0
  THEN 490 ::  IF K[]ASC("Y")T
HEN 550 ELSE 500
500 DISPLAY AT(2,2)ERASE ALL
:"THE OBJECT IS TO SHOOT DOW
N": :"THE ALPHABET IN ORDER"
510 DISPLAY AT(6,2):"USE THE
  JOYSTICK AND FIRING": :"BUT
TON.THERE ARE 52 BULLETS"
520 DISPLAY AT(11,1):"SO DO
NOT WASTE THEM...." ::  DISPL
AY AT(13,5):"GO TO IT! GOOD
LUCK!"
530 DISPLAY AT(20,5):"PUSH A
NY KEY TO START"
540 CALL KEY(0,K,S)::  IF S=0
  THEN 540 ELSE 550
550 CALL CLEAR
560 CALL SCREEN(2)
570 CALL MAGNIFY(1)
580 RANDOMIZE
590 FOR S=1 TO 26
600 R=INT(RND*120)+1 ::  C=IN
T(RND*246)+10 ::  CS=INT(RND*
15)+1
610 CALL SPRITE(#S,64+S,INT(
S/2)+3,R,C,0,CS)
620 NEXT S
630 CALL SPRITE(#27,94,16,17
0,128)
640 AMMO=52
650 FOR T=1 TO 26
660 FOR CL=3 TO 8 ::  CALL CO
LOR(CL,16,1)::  NEXT CL
670 CALL POSITION(#27,R,D)
680 CALL JOYST(1,X,Y)::  Y=0
690 CALL KEY(1,K,S)::  IF S=0
  THEN 730 ::  IF K=18 THEN CA
LL SPRITE(#28,46,16,R,D,-25,
0)::  CALL SOUND(100,-1,0)::
AMMO=AMMO-1
700 IF AMMO=0 THEN 820
710 CALL COINC(#T,#28,8,C)::
  CALL POSITION(#28,R1,D1)
720 IF C=-1 THEN 750 ::  IF R
1[9 THEN CALL DELSPRITE(#28)
ELSE 710
730 DISPLAY AT(23,2):"AMMO="
;AMMO
740 CALL MOTION(#27,-Y,X*6):
:  GOTO 670
750 CALL SOUND(250,-7,0)::  C
ALL DELSPRITE(#T)::  CALL DEL
SPRITE(#28)
760 DISPLAY AT(24,1+T):CHR$(
64+T)
770 IF T]26 THEN 790
780 NEXT T
790 CALL DELSPRITE(ALL)::  CA
LL CLEAR ::  CALL SCREEN(2)
800 DISPLAY AT(10,1):"WELL D
ONE WANT TO PLAY AGAIN" ::  D
ISPLAY AT(12,11):"Y OR N" ::
  DISPLAY AT(14,1):"YOU HAD";
AMMO;"BULLETS LEFT"
810 CALL KEY(0,K,S)::  IF S=0
  THEN 810 ::  IF K[]ASC("Y")T
HEN 100 ELSE 550
820 CALL DELSPRITE(ALL)::  CA
LL CLEAR ::  CALL SCREEN(2)
830 DISPLAY AT(10,1):"SORRY-
OUT OF AMMO PLAY AGAIN" ::  D
ISPLAY AT(12,11):"Y OR N"
840 CALL KEY(0,K,S)::  IF S=0
  THEN 840 ::  IF K[]ASC("Y")T
HEN 100 ELSE 470
850 DISPLAY AT(12,11)ERASE A
LL:"GOODBYE" ::  CALL SAY("GO
ODBYE")
860 FOR DE=1 TO 1000 ::  NEXT
DE ::  CALL CLEAR ::  END
```

SUBROUTINE TO PRINT LONG STRINGS.

Ever get annoyed when your program prints to the screen and sp
lits a word at the end of a line? The following routine locates the
spaces between words and starts a new line if the whole word won't fit
at the end of a line. The routine works wonders when string variables
of varying lengths are combined with text.

TRY THIS:-

```
100 INPUT "NAME? ":NAME$
110 INPUT "STREET? ":STREET$
120 INPUT "CITY?   ":CITY$
130 PRINT
140 M$="I BELIEVE YOUR NAME
IS "&NAME$&" AND YOU LIVE IN
 "&CITY$&". DO ANY OTHER "
150 M$=M$&NAME$&"S LIVE IN "
&STREET$&", "&CITY$&"? I WOU
LD LIKE TO VISIT "&STREET$&"
 "&NAME$
160 GOSUB 9000
170 PRINT
180 GOTO 100
```

SUBROUTINE:-

```
9000 IJ=1
9010 PS(IJ)=POS(M$," ",PS(IJ
-1)+1)
9020 IF PS(IJ)>28 THEN 9070
9030 IF (PS(IJ)=0)*(LEN(M$)>
28) THEN 9070
9040 IF PS(IJ)=0 THEN 9100
9050 IJ=IJ+1
9060 GOTO 9010
9070 PRINT SEG$(M$,1,PS(IJ-1
))
9080 M$=SEG$(M$,PS(IJ-1)+1,L
EN(M$)-PS(IJ-1))
9090 GOTO 9000
9100 PRINT M$
9110 RETURN
```

BY COL CHRISTENSEN

WORDSQUARE   BUG-BYTES/1

```
ERAWTFOSBTROPUG     ASSEMBLY      MODEM
TBYDTEHCEEPSSNE     BASIC         MODULATOR
IELDRNOISNAPXES     BIT           MODULE
BEOCSAMMROMLREC     BYTE          MONITOR
MCGOOOOONODATRI     CARTRIDGE     MULTIPLAN
UIODDNTBDBIRUTH     COMPUTER      PASCAL
LIAEIISUYRSEROP     CONSOLE       PERIPHERAL
TSMSNRLOEEKHEBA     DATA          PORT
IMAOSATTLLKPTAR     DISK          PRINTER
POMXTEURAETINSG     EXPANSION     SOFTWARE
LDWOEPMCADARIIT     FORTH         SPEECH
AUROMTSBACPERCE     GRAPHICS      TAPE
NLNORASTLFEPPRX     KEYBOARD      TEXAS
OECMPDAETYBMEMT     LOGO          TEXT
BYROMEMERHTROFS     MEMORY        WORD
```

Cross out in the wordsquare the letters of each word.
The remaining letters taken in order spell out an
important message to all our members. The Editor
would appreciate an early response.

errors. When debugging this program the things to look for are:-
1. Spelling errors.
2. Typing errors.
3. programming errors.

Spelling errors include statements that were not typed as per the syntax.
Typing errors include "I" for "1" and "O" for zero. Programming errors are
using a statement out of sequence or values too small or too big. This type
of bug is the one programmers most often make.

```
100 REM THIS PROGRAM HAS          200 CALL CHAR(162,"8I42241818
110 REM BEEN INSTALLED WITH       3C2400")
120 REM SEVERAL BUGS              210 FOR ROW=25 TO 1 STEP 1
130 REM IT IS YOUR JOB TO         220 FOR CHAR=160 TO 162
140 REM FIND THEM                 230 CALL HCHAR(ROW,16,CHAR)
150 CH99/USERS GROUP              240 CALL SOUND(-60,-5,0)
160 REM BY D.STOREY               250 NEXT ROW
170 CALL CLAER                    260 NEXT CHAR
180 CALL CHAR(160,"0000669818     270 RETURN
362400")                          280 END
190 CALL CHAR(161,"0042A51818
362400)
```

EDITOR/ASSEMBLER

BASE CONVERSION ROUTINES FOR EDITOR/ASSEMBLER OR MINIMEMORY

By Phil West   T.I.U.P.

Called from BASIC

CALL LINK("HEXDEC",HEXVAL$,DECVAL)

CALL LINK("DECHEX",DECVAL,HEXVAL$)

Where HEXVAL$ is a string of 1 to 4 HEX Characters.

where DECVAL is a numeric variable.

Editor Assembler Only

```
    DEF   HEXDEC, DECHEX         Define program names
    REF    STRREF,STRASG,NUMREF   Reference Utilities
    REF   NUMASG,XMLLNK,ERR
IF EQU   ]2300                   Value for E/A CIF Routine
```


Minimemory Only

```
    AORG   ]7D80                 AORG to this address in M/M
SR EQU   ]604C                  Minimemory Utility Equates
SA EQU   ]6048
NR EQU   ]6044
NA EQU   ]6040
ER EQU   ]6050
XM EQU   ]6018
IF EQU   ]7200                  Value for M/M CIF Rouutine
```

COMMON

Note 1. If you are using the Editor/Assembler use the REF enclosed in brackets instead of the Label.

NOTE 2. If you are using the Line-by-Line Assembler do not attempt to include any of the comments in this listing.

NOTE 3. To use these programs with the Editor/Assembler you must use the following statements in a Basic program or in command mode.

CALL INIT

CALL LOAD("DSK1.BSCSUP",DSK1.BASE") Where "DSK1.BASE" is the Object File produced by the Assembler.

```
FI EQU   ]1200                  Value for CFI routine
BV EQU   ]1300                  Bad Value for Error Utility
FC EQU   ]834A                  Floating point Accumulator
OV EQU   ]8354                  Floating Point Error Code
ST EQU   ]837C                  GPL Status Byte
LB DATA  ]0403                  Max Length Byte & Error check
VL DATA  0,4,8,12,0             Values to Shift R4 in HEXDEC
TX TEXT  '0123456789ABCDEF'     Comparison String
BF BSS   6                      String Buffer
```

Convert Hexadecimal String to Number

```
HD CLR   R0                     Simple Variable R0-0
   CLR   R5                     Clear R5-Integer Result
   LI    R1,1                   First Parameter in Link List
   LI    R2,BF                  CPU Address of String Buffer
   MOVB  @LB,*R2                Put Maximum Length Byte in Buff
   BLWP  @SR        (STRREF)    Get String from Basic
   SRL   R2,8                   Move to LSB of R2
NC CLR   R3                     Clear Pointer to TX String
NH CB    @BF(R2),@TX(R3)        Compare Inpt String to TX String
   JNE   A3                     No Match Found
   MOV   R2,R1                  Copy R2,R1
   SLA   R1,1                   Multiply R1*2
   MOV   @VL(R1),R0             Get value to Shift R4-Put in R0
   MOV   R3,R4                  Copy R3,R4
   SRC   R4,0                   Shift R4 to get Integer Value
   SOC   R4,R5                  Put value in R5
   JMP   D2                     Proceed to next Char in Input
A3 INC   R3                     No Match-point Next Char TX
   CI    R3,16                  End of Tx String?
   JLT   NH                     No Compare Next Char
D2 DEC   R2                     Yes-point to next Char in Input
   JNE   NC                     If not finished - Repeat
```

```
        MOV    R5,@FC                      Copy R5 into Floating Point ACC
        BLWP   @XM         (XMLLNK)        Convert integer to floating point
        DATA   IF
        CLR    R0                          Simple Variable R0-0
        LI     R1,2                        Second Parameter in Link List
        BLWP   @NA         (NUMASG)        Assign Numeric Value to BASIC
BK MOVB @VL,@ST                            Clear GPL Status Byte
        B      *R11                        Return to BASIC


Convert Decimal Number to Hexadecimal String


DH CLR  R0                                 Simple Variable R0-0
        LI     R1,1                        First Parameter in Link LIst
        BLWP   @NR         (NUMREF)        Get Number from Basic
        BLWP   @XM         (XMLLNK)        Convert Number to Integer
        DATA   FI
        CB     @OV,@LB+1                   Check for overflow Error
        JNE    OK                          If OK Proceed
        LI     R0,BV                       Error detected - Load R0-BV
        BLWP   @ER         (ERR)           Report Error to BASIC
OK MOV  @FC,R5                             Copy Integer value into R5
        LI     R6,4                        Load R6-Number of Chars
        LI     R7,BF                       LOad R7-Output Buffer
        MOV    R7,R2                       Copy R7,R2
        MOVB   @LB,*R7+                    Put length Byte in Buffer
M5 MOV  R5,R3                              Copy R5,R3
        SRL    R3,12                       Discard 12 LSB Bits
        SLA    R3,8                        Move value to MSB
        CI     R3,]0900                    Is  Value [9 ?
AD AI   R3,]3000                           Add ]30 to get ASCII code
        MOVB   R3,*R7+                     Put ASCII Char into Buffer
        SRC    R5,12                       Rotate R5 for next Char
        DEC    R6                          Decrement Char count
        JNE    M5                          If not finished-Continue
        CLR    R0                          Simple Variable R0-0
        LI     R1,2                        Second Parameter in Link List
        BLWP   @SA         (STRASG)        Assign String to BASIC
        B      @BK                         Return


Minimemory Only
```

```
    AORG    ]7FEO              Put Program Names in REF/DEF Table
    TEXT    'HEXDEC'
    DATA    HD
    TEXT    'DECHEX'
    DATA    DH
    AORG    ]701E              Update Last Free Address in
    DATA    ]7FEO              Minimemory Pointer


 Editor Assembler Only


HEXDEC     EQU HD              Equate Program Names to entry points
DECHEX     EQU DH


 Common


 END                          End of Source Listing
```

THE TI.S.H.U.G. BULLETIN BOARD from SYDNEY NEWDIGEST

The Sydney Users Group now have their very own Bulletin board available to
all you lucky people with modems. It features Electronic Mail and Shopping,
Latest News, both local and overseas, Programming hints, and Down-loadable
Software, and coming soon, will be Full Speech. This will be the only BBS
in the Southern Hemisphere to have full clear spoken text as it is displayed
on your screen. This means that if you have a speech synthesizer, your
TI-99/4A will TELL YOU all the information you wish to hear and read.

This system will be on during the following days and times. Do not try to
log on outside of these hours as the computer used for this BBS is also used
for the production of the SYDNEY NEWSDIGEST and personal use.

SUNDAY: 9AM to 12 MIDNIGHT (except 2nd Sunday of month)

MONDAY: 7AM to 7PM then 8.30PM to MIDNIGHT

TUESDAY: as above

THURSDAY: 7AM to 10PM

DOWN-LOADABLE SOFTWARE

LOAD: for tose with disk drives, this version of a LOAD program by Russell
Welham will come in very handy. Save it as LOAD and when you go Extended
Basic, it will automatically produce a Catalogue of the contents of your
disk then enable you to load and run a program with one press of a key.

23

BUGLETYPE: is a combination of music and typing tutor. The faster you type, the clearer the music becomes.

SURVIVAL: fight for your life.

CIRCUS: An arcade game... you have to bounce up and pop the balloons, but be careful not to crash to the ground.

TOTAL CATALOG: Your very own DISK CATALOGUING program, to keep a printed record of all your disk titles. This program will sort then ask you if you would like 1, 2, or 3 columns of listings on your printer. A very good program...enables you to store around 300 titles in it.

HIDDEN SHAPES: A test of skill (good luck. YOu'll need it).

LEARNOTE: A music education program.

MURDER AT KINGS CROSS: WHO DID IT? Play the game and find out. A text adventure

TOWERS OF HANOI: How good is your memory? you need brains for this one... that does it - no-one will complete this one (chuckle chuckle).

MAINSCREEN: Colour Graphics and Music. Fun to watch and great to show off.

INVASION OF THE VOLKSWAGONS: Fast reflexes are needed for this one...to prevent the bugs from dropping on you.

BOWL MATH: This is a cutegraphics Education game...the only thing wrong with it is the bowling ball. Somehow it lost it shape. Correct this error and let Shane know on Electronic Mail or Chat mode.

The software disk will be changed about once a monthwith the cream of the club software - usually before the Club Library is able to make it available. Some software will be exclusive to the BBS users.

Now I have got you all interested how do you go about it? First, you need an RS232 card, a modem, and Terminal Emulator cartridge. Providing you have these, all you need do is dial 02 560 0926 and log on as a guest. If you want to become a regular user, you will have to fill out an application form, which you can obtain by writing to TI.S.H.U.G. AT P.O. BOX 595 MARRICKVILLE. N.S.W. 2204. It is a free service to all TIS.H.U.G. MEMBERS, but I do not know how much if anything they will be charging Members of other user groups from Interstate. For further details ring SHANE ANDERSEN on (02) 29 ₁631 ing working hours.

```
EEEEE  N   N  L        AAA    RRRR   GGGG   EEEEE  RRRR
E      NN  N  L       A   A   R   R  G      E      R   R
E      NN  N  L       A   A   R   R  G      E      R   R
EEEE   N N N  L       AAAAA   RRRR   G GGG  EEEE   RRRR
E      N  NN  L       A   A   R  R   G   G  E      R  R
E      N  NN  L       A   A   R   R  G   G  E      R   R
EEEEE  N   N  LLLLL   A   A   R   R   GGG   EEEEE  R   R
```

As the heading suggests, this program creates this type of print display from INPUT text and directs it either to a printer or to a diskfile for direct access by TI-WRITER or EDITOR-ASSEMBLER as I have done in writing this.   The program accepts up to 7 lines of text, enough to fill a page of paper when using standard Pica printstyle and 1/6 inch line spacing.

```
100 REM*********************
110 REM*  ENLARGED PRINT   *
120 REM* by C.Christensen   *
130 REM*********************
140 CALL CLEAR
150 DIM B$(20)
160 DISPLAY AT(5,1):"HOW MAN
Y CHARACTERS PER LINE": :"DO
ES YOUR PRINTER ACCEPT?80"
170 ACCEPT AT(7,26)SIZE(-3)B
EEP VALIDATE(DIGIT):LONG
180 DISPLAY AT(12,3):"SAVE T
O  I. DSKI": :TAB(12);"2. PR
INTER": : : :TAB(4);"YOUR CH
OICE? (1 OR 2)"
190 ACCEPT AT(18,26)VALIDATE
("12")BEEP:CHOICE
200 IF CHOICE=1 THEN 230
210 DISPLAY AT(21,1):"PRINTE
R DEVICENAME IS:-" :: ACCEPT
 AT(23,3)BEEP:DEVICE$
220 OPEN #1:DEVICE$,VARIABLE
LONG :: GOTO 250
230 DISPLAY AT(21,3):"FILENA
ME? DSK1." :: ACCEPT AT(21,
19)SIZE(10)BEEP:FILE$
240 OPEN #1:"DSK1."&FILE$,DI
SPLAY ,VARIABLE 80
250 REM ***SET UP SCREEN***
260 CALL CLEAR
270 CALL COLOR(14,1,1)
280 CALL HCHAR(1,1,140,768)
290 CALL COLOR(1,16,16)
300 FOR I=9 TO 15 :: CALL HC
HAR(I,12,32,INT(LONG/8)):: N
EXT I
310 FOR I=1 TO 12 :: CALL CO
LOR(I,2,16):: NEXT I
320 FOR I=1 TO 7
330 ACCEPT AT(8+I,10)SIZE(-I
NT(LONG/8)):SC$(I):: NEXT I
340 M=7 :: FOR I=7 TO 1 STEP
-1 :: IF SC$(I)="" THEN M=M
-1 ELSE 360
350 NEXT I
360 DISPLAY AT(20,3)SIZE(18)
:"ANY CHANGES? Y/N N" :: ACC
EPT AT(20,20)SIZE(-1)VALIDAT
E("YNyn"):YN$
370 CALL HCHAR(20,1,140,32)
380 IF YN$="Y" OR YN$="y" TH
EN 320
390 DISPLAY AT(22,1):"HOW MA
NY BLANK LINES TO BE":"LEFT
ON THE PAPER BEFORE THE":"FI
RST PRINT LINE?"
400 ACCEPT AT(24,19)VALIDATE
(DIGIT)BEEP:LINES
410 DISPLAY AT(22,1):"HOW MA
NY BLANK LINES BENEATH": :"E
ACH ROW?[MINIMUM OF 1]"
420 ACCEPT AT(24,25)SIZE(-2)
VALIDATE(DIGIT):SPACES
430 IF SPACES<1 THEN 420
440 DISPLAY AT(22,1):"THAT M
AKES A TOTAL NUMBER": :"OF";
M*8+LINES-1+(M-1)*(SPACES-1)
;"LINES. O.K.?  Y"
450 ACCEPT AT(24,21)VALIDATE
("YN")SIZE(-1):YN$
460 IF YN$="N" THEN 390
470 FOR I=22 TO 24 :: CALL H
CHAR(I,1,140,32):: NEXT I
480 DISPLAY AT(20,5)SIZE(5):
"ROW" :: DISPLAY AT(20,15)SI
ZE(11):"PIXEL ROW"
490 FOR I=1 TO LINES-1 :: PR
INT #1 :: NEXT I
500 REM **EACH ROW OF TEXT**
510 FOR ROW=1 TO M
520 A$=SC$(ROW)
530 REM ****DIM CHARPATS****
540 FOR LOOP=1 TO LEN(A$)
550 CALL CHARPAT(ASC(SEG$(A$
,LOOP,1)),B$(LOOP))
```

25

```
560 NEXT LOOP                          740 IF D$="8" THEN E$=Z$&"
570 REM **HEX STRINGS FOR***               " :: GOTO 820
       ******ONE ROW*******            750 IF D$="9" THEN E$=Z$&"
580 FOR HEX=1 TO 8                         "&Z$ :: GOTO 820
590 FOR CH=1 TO LEN(A$)                 760 IF D$="A" THEN E$=Z$&" "
600 C$=C$&SEG$(B$(CH),HEX*2-                &Z$&" " :: GOTO 820
1,2)                                    770 IF D$="B" THEN E$=Z$&" "
610 NEXT CH                                 &Z$&Z$ :: GOTO 820
620 REM **BINARY STRINGS FOR            780 IF D$="C" THEN E$=Z$&Z$&
     ***EACH PIXEL ROW***                   " " :: GOTO 820
630 FOR BIN=1 TO LEN(C$)                790 IF D$="D" THEN E$=Z$&Z$&
640 Z$=SEG$(A$,INT((BIN+1)/2                 " "&Z$ :: GOTO 820
),1)                                    800 IF D$="E" THEN E$=Z$&Z$&
650 D$=SEG$(C$,BIN,1)                        Z$&" " :: GOTO 820
660 IF D$="0" THEN E$="    "            810 IF D$="F" THEN E$=Z$&Z$&
    :: GOTO 820                              Z$&Z$ :: GOTO 820
670 IF D$="1" THEN E$="   "&            820 F$=F$&E$
Z$ :: GOTO 820                          830 NEXT BIN
680 IF D$="2" THEN E$="  "&Z            840 DISPLAY AT(20,9)SIZE(1):
$&" " :: GOTO 820                       STR$(ROW):: DISPLAY AT(20,25
690 IF D$="3" THEN E$="  "&Z            )SIZE(1);STR$(HEX)
$&Z$ :: GOTO 820                        850 PRINT #1:F$
700 IF D$="4" THEN E$=" "&Z$            860 F$=""
&"  " :: GOTO 820                       870 C$=""
710 IF D$="5" THEN E$=" "&Z$            880 NEXT HEX
&" "&Z$ :: GOTO 820                     890 IF ROW=N THEN 920
720 IF D$="6" THEN E$=" "&Z$            900 FOR I=1 TO SPACES-1 :: P
&Z$&" " :: GOTO 820                     RINT #1 :: NEXT I
730 IF D$="7" THEN E$=" "&Z$            910 NEXT ROW
&Z$&Z$ :: GOTO 820                      920 CLOSE #1 :: END
```

LETTERS TO THE EDITOR

Dear Sandra,

Please find enclosed a program that might be fun for the kids (even big
ones). I have one request, has anybody found out how to screen dump to a
printer yet? Also the WRITER program from Bug-bytes June has a few 'bugs in
it, line 240, no = in if L$..., line 190 the [ ] are confusing.

                          Ian J.Smith, Mt Isa

Dear Ian,

Thank you for the program, it will be a good one for KIDS KORNER. As to
your problems, I haven't found a screen dump program yet, however, if
someone out there has one, and would like to let me have it, I will put it
in the Newsletter
As to tthe bugs in Writer, perhaps you don't realize that we use the Square
brackets, [and] in place of the greater than and less than signs,
character nos 60 and 62 in the ASCII code. Unfortunately, the Daisy wheel
printer that we use to print up the program listings does not have those
characters. Line 240 therefore reads IF L$ less than greater than "" then
270, and line 190 IF (S=0)+(K less than 49 + K greater than 56)THEN 190.
hope this clarifies the situation. We are trying to get a new daidy wheel
with all the ASCII characters on it.

2 6   Sandra Nichelsen, Editor

CONVERSION BY J.GROENVELD

This program was written by one of our newest members, J.Groenveld, who has only had his computer for 3 weeks. Thank you very much for your contribution, and perhaps some of our longer standing members can follow your example and contribute something that they have written.

```
100 REM ****************
110 REM *  CONVERSIONS *
120 REM *BY J.GROENVELD*
130 REM ****************
140 CALL CLEAR
150 PRINT "IN THIS PROGRAM Y
OU CAN DO  THE FOLLOWING CON
VERSIONS:-"
160 PRINT :"(1)..MILES TO KI
LOMETRES":"(2)..KILOMETRES T
O MILES":"(3)..MILES PER GAL
LOM"
170 FOR DELAY=1 TO 2000
180 NEXT DELAY
190 CALL CLEAR
200 LET KLM=8
210 LET MLS=5
220 LET GAL=(4*L)
230 LET L=.21997
240 PRINT "CONVARSATION - (1
)MLS -KLMS           (2
)KLMS -MLS           (3
)MPG"
250 INPUT "YOUR CHOICE,1,2,0
R 3?":X
260 IF X]3 THEN 240
270 ON X GOTO 280,350,4000
280 CALL CLEAR
290 PRINT
300 INPUT "MILES=":A
310 PRINT
320 LET C=((A*KLM)/MLS)
330 PRINT C: :
340 GOTO 240
350 CALL CLEAR
360 INPUT "KILOMETRES=":A2
370 LET D=((A2*MLS)/KLM)
380 PRINT D: :
390 GOTO 240
400 CALL CLEAR
410 INPUT "DISTANCE=":M
420 PRINT
430 INPUT "FUEEL=":F
440 PRINT :
450 LET Z=M/F
460 PRINT :
470 PRINT "MPG=";Z
480 FOR DELAY=1 TO 500
490 NEXT DELAY
500 CALL CLEAR
510 GOTO 240
520 END
```

BANKCARD

T.I. WRITER Saver.

T.I. WRITER has an annoying feature in that it always feeds a sheet of paper when printing with the text formatter. Many newsletters have suggested a way to avoid this waste of paper. You turn off your printer while answering the question screen, you answer Y to PAUSE AT END OF PAGE? then turn the printer back on. I think this procedure is hazardous to your printer. The most vulnerable period in the life of an intergrated circuit after the initial "burn in period" is the time when the power supply is turned on or off, generating spikes in the battery lines.

As an alternative to turning the printer on and off, I use the following procedure which makes use of the fact that when not using a header on your page, the formatter after form-feeding a page line feeds to the sixth line on the page. Even when using a header it will line feed two lines.

Fit the paper into the printer in the place where you want the top of the page to be, but don't engage the feed sprockets into the paper. Answer the questions as normally. After you press enter, after the STOP AT END OF PAGE question, the printer will start to feed a page out but as the sprocket is not engaged with the paper it will not waste a page, then when the printer begins to line feed press the on/off line button on your printer, this will stop the printer so you may engage the sprockets in the paper. Pressing the on/off line button again allows the printer to resume. After printing a few pages using this method you can easily judge where to position your paper in the printer to compensate for the lost linefeeds.

The above procedure works well on my EPSOM printer, but with printers that can friction feed paper, it is best to not insert the paper into the printer until after the form-feed stage, as the paper tends to be dragged through by the platen.

Bruce W.Carew.

TIMES June 1984

QUESTIONNAIRES

Included elsewhere in this newsletter is the result of the questionnaire in February's Newsletter. The winner of the Home Computer Magazine promised as a prize is JACK L'ESTRANGE. We will be contacting you Jack to arrange for you to get the magazine. Thank you all for taking the time to fill them out. They will be a great help to the committee.

MODULE LIBRARY

The committee has decided to start a module library as soon as funds are available from people renewing their membership. We have decided to set the fees as follows:

$5 JOINING FEE payable once only

$30 REFUNDABLE DEPOSIT

$2 per MONTH RENTAL plus postage if required

If you think you would be interested in joining, please let us know. The quicker you renew your membership and let us know of your interest, the sooner we will be able to start up.

DATES TO REMEMBER

BRISBANE MEETING	31st AUGUST	7.30 pm
TOOWOOMBA MEETING	29th AUGUST	7.30 pm
IPSWICH MEETING	14th SEPTEMBER	7.30 pm
NORTHSIDE MEETING	2nd SEP.EMBER	10.00 am
SOUTHSIDE MEETING	11th SEPTEMBER	7.30 pm

TE-II OUTPUT FILES

For those of you who use the TI Terminal Emulator II and save data to an external device such as a cassette or disk here is an Extended Basic program that will allow you to retrieve the data you have saved.

From Enthusaist '99.
TIMES June 1984

```
100 INPUT"device and file na
me":FILES
110 OPEN#1:FILE$,INPUT,DISPL
AY,FIXED 80
120 LINPUT:A$
130 IF EOF(1) THEN 160
140 PRINT A$
150 GOTO 120
160 CLOSE#1
170 END
```

RESULTS OF QUESTIONNAIRE - 1984

From a total of close to 200 newsletters sent to members early in 1984, 54 questionnaires were completed and returned. Thank you all sincerely for your time and effort. The information has now been compiled and passed on to your committee. The lucky winner of a free magazine is

Below is a brief summary of the results of the questionnaire. Replies were received from 15 people under 25 years of age and 37 members over 2 . Of these, the majority lived in the Brisbane metropolitan area. There were an even proportion of 'new' and 'old' members. Approximately ½ of the total worked in computer related occupations.

We had a great response to our plea for helpers! Lets hope you are all still available and willing when you're called on in the near future.

Over half our members owned or inted to buy Speech Synthesizer, Disk drives, Peripheral Expansion boxes and 32k memory expansion. The other peripherals weren't quite as popular. Although we seem to always be hearing of problems with cassette players, over half our members have had no problems at all! The biggest problem is load/save and solutions seem to be perserverence with volume control.

Nearly all members have or intend to buy Extended Basic and half of you have or intend to have Editor/Assembler. Most own 3-6 cartridges. The most popular are Extended Basic, Parsec, Chess, Munchman and TI Invaders.

As many users class themselves as beginners as intermediate programmers. Hey you 13 with technical knowledge, we have 44 who want to share it. And almost everybody wants help to learn programming.

In order of priority, your uses for your computers are Education, games, computer literacy and business. Most of you use your computers 3-6 hours per week. Household uses are many and varied including tax records, fuel consumption, recipes, sports records, household budget and inventory etc.

Most of you were reasonably happy with our newsletter at the time of answering the question aires and even more of you intend to contribute. NOW is the time! There were numerous comments and suggestions for improvements to the newsletter and they have been passed on.

Surprisingly most of you owned none or only 1-2 club tapes. I'm sure if you haven't got hold of some by now, there will be plenty available. Most of you want to attend 4 or more day workshops per year. Perhaps that has changed slightly now with the growth of sub-group meetings but you indicated that you are prepared to pay for a professionally run workshop with competent tutors available.

We have numerous ideas for activities and topics for meetings and the numbers indicate a combination of sub-group, special interest group and central meetings to be most popular. We have taken notice of your comments 'Mail Only' members and will advise you more of what is going on. If it is decided that fund raising is necessary, we have many ideas.

Well that's basically it - a brief summary of what you had to say. Thanks again for your participation. The information is in the hands of the committee now and hopefully some changes will be made soon to cater for your needs.

Shirley Peppler.

Here is a programme to plot high resolution graphs. It comes from a User group in england.

```
100 REM   *******************

110 REM   * HIGH RESOLUTION *

120 REM   *X/Y GRAPH PLOTTER*

130 REM   *                 *

140 REM   * BY JOHN STOCKS   *

150 REM   * FROM T.I. GROUP *

160 REM   *     ENGLAND     *

170 REM   *******************

180 CALL CLEAR
190 GOTO 1020
200 Y=12*EXP(-X/10)*COS(X)
210 GOTO 370
220 CALL SCREEN(4)
230 R=12
240 C=1
250 CALL CLEAR
260 CALL CHAR(33,"0000000000
000080")
270 CALL HCHAR(R,2,33,30)
280 CALL VCHAR(1,C,33,24)
290 M=143
300 OPTION BASE 1
310 DIM H(16),I$(16)
320 FOR A=3-C TO 32-C
330 FOR B=1 TO 8
340 X=A+(B-1)/8
350 IF X=0 THEN 410
360 GOTO 200
370 IF (INT(Y)[R-24)+(INT(Y)
]R-1)THEN 410
380 E(B)=INT(8*(Y-INT(Y)))
390 F(B)=R-INT(Y)
400 GOTO 430
410 E(B)=8
420 F(B)=1
430 NEXT B
440 Q=1
450 FOR K=1 TO 8
460 IF G(K)[]0 THEN 530
470 G(K)=Q
480 FOR L=K+1 TO 8
490 IF (G(L)[]0)+(F(L)[]F(K)
)THEN 510
500 G(L)=Q
510 NEXT L
520 Q=Q+1
530 NEXT K
540 FOR B=1 TO 8
550 D(E(B)+1,B,G(B))=1
560 D(9,8,G(B))=F(B)
570 IF (R-F(B)=0)+(A+0)THEN
590
580 GOTO 600
590 D(1,1,G(B))=1
600 NEXT B
610 FOR K=1 TO 8
620 IF D(9,8,K)=0 THEN 970
630 FOR L=8 TO 1 STEP -1
640 N=8*D(L,1,K)+4*D(L,2,K)+
2*D(L,3,K)+D(L,4,K)
650 J=8*D(L,5,K))+4*D(L,6,K)+
2*D(L,7,K)+D(L,8,K)
660 H(17-2*L)=N
670 H(18-2*L)=J
680 NEXT L
690 FOR L=1 TO 16
700 IF H(L)[10 THEN 720
 710 ON 16-H(L)GOTO 740,760,7
80,800,820,840
720 I$(L)=STR$(H(L))
730 GOTO 850
740 I$(L)="F"
750 GOTO 850
760 I$(L)="E"
770 GOTO 850
780 I$(L)="D"
```

17

```
790 GOTO 850
800 I$(L)="C"
810 GOTO 850
820 I$(L)="B"
830 GOTO 850
840 I$(L)="A"
850 NEXT L
860 P$=I$(1)&I$(2)&I$(3)&I$(
4)&I$(5)&I$(6)&I$(7)&I$(8)&I
$(9)&I$(10)&I$(11)&I$(12)&I$
(13)&I$(14)&I$(15)&I$(16)
870 CALL CHAR(M,P$)
880 CALL HCHAR(D(9,8,K),A+C,
M)
890 M=M-1
900 IF M=33 THEN 1010
910 FOR L=1 TO 9
920 FOR J=1 TO 8
930 D(L,J,K)=0
940 NEXT J
950 NEXT L
960 NEXT K
970 FOR L=1 TO 8
980 G(L)=0
990 NEXT L
1000 NEXT A
1010 GOTO 1010
1020 CALL SCREEN(4)
1030 CALL CHAR(64,"00FF00000
0000000")
1040 FOR K=1 TO 8
1050 CALL COLOR(K,1,4)
1060 NEXT K
1070 READ A$
1080 IF A$="X"THEN 1110
1090 PRINT A$
1100 GOTO 1070
1110 FOR K=1 TO 8
1120 CALL COLOR(K,2,4)
1130 NEXT K
1140 FOR K=1 TO 3000
1150 NEXT K
1160 GOTO 220
1170 DATA "     HIGH-RESOLUT
ION       *************
***          X/Y  GRAPH P
LOT       "
1180 DATA "    ************
***       ",,,EDIT 200 TO SPE
CIFY Y=F(X)  ********,
1190 DATA EDIT 230 TO LOCATE
 X AXIS   ********,"
 (ROWS 1 TO 24)",
1200 DATA EDIT 240 TO LOCATE
 Y AXIS   ********,"
 (COLUMNS 3 TO 32)",
1210 DATA RUN 220 TO BYPASS
CAPTION   ********,,,X
```

DEBUGGING YOUR BUGS by David Storey TISHUG

The following program is full of bugs. It's your job to find them. The
program when running will show a bird-like animal fly up the screen then it
will clear the screen and start over again.

Next month, i will give you the answers to this, and give you a new program
to work on.

My defination of bugs are spelling errors, typing errors and programming

SUB - GROUP NEWS

NOGGERA GROUP - Contact OWEN HARVEY 355 9317
f you live near Enoggera, you"ll be interested in hearing that Owen wishes
o start a group based there. Owen feels that the Northside meeting at
ashville are too far away for the people from Enoggera, Everton Park area
nd he envisages a small group meeting once a month in members' homes.

OUTH COAST GROUP - Contact SIMON REID (075) 38 4262
imon is waiting to here from you people on the South coast in order to
rganise a meeting. Please contact him as soon as possible.

OOWOOMBA - Contact GREG MCRAE (076) 34 4280
eetings are held on the last Wednesday of each month at Greg's house, 5
udgee Crt, Toowoomba at 7.30pm. All are welcome.

PSWICH - Contact JOHN HOLLAND 281 4526
eetings are held on the 2nd Friday of each month at the NORTH IPSWICH STATE
CHOOL at 7.30pm. Everyone is welcome.

ORTHSIDE - Contact HUMPHREY LINDLEY 263 6161 or COL CHRISTENSEN 284 7783
eetings are held on the 1st Sunday of each month at the NASHVILLE STATE
CHOOL (near Sandgate) from 10am until 3pm. You don't have to stay for the
hole day, just an hour or so if you want to. All are welcome.

OUTHSIDE - Contact SANDRA NICHELSEN 341 5667
eetings are held on the 2nd Tuesday of each month at MACGREGOR HIGH SCHOOL
lackwattle St, Macgregor (Upper Mt Gravatt) at 7.30pm. Everyone is
elcome.

SSEMBLY - Contact MARCEL ARIAS 203 6512
o meetings are held at present.

ARE YOU HAVING TROUBLE WITH THE REMOTE ON YOUR CASSETTE PLAYER

If this is your problem, help is at hand. John Holland is making up some
Polarity Reversers for sale to club members. These should be quite inexpensive,
(probably $2 to $2.50). If you are having trouble getting your remote to work,
this will almost certainly fix the problem, so make sure you get along to the
next meeting to put your order in.

31

For Sale —— Wanted

FOR SALE:- PRINTER Star brand. Dot matrix. 80 c.p.s. 10".
parallel. tractor and friction feed. text buffer. 300 sheets of
fan fold paper included.
Make an appointment for a free demonstration.
Phone (07) 349 6157

FOR SALE:- FULL SYSTEM includes the following: CONSOLE (black 4A).
SPEECH SYNTHESIZER. P.E. BOX. DISK DRIVE and CONTROLLER. 32K
MEMORY EXPANSION. RS232 CARD. and the following software:
MULTIPLAN. VIDEO CHESS. EXTENDED BASIC. BOX OF DISKS with
programs. Will separate Console and Extended Basic. $1000
Phone Marcel (07) 207 6512