

TI*MES



TI*MES



TI*MES
NEWSLETTER

TI*MES

Biggest
ever
issue



Be74 (A) Computer and 3-C part items

The first step to a complete system can be your own cassette recorder.

WINTER

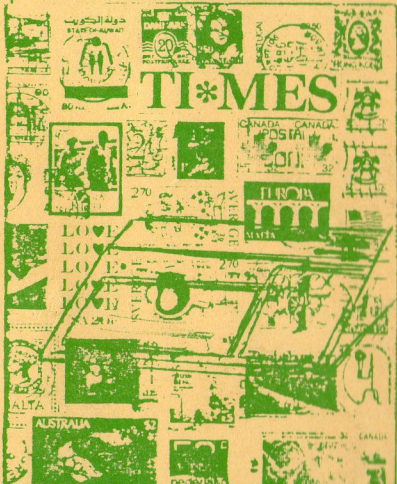
TI*MES
NEWSLETTER

TI*MES

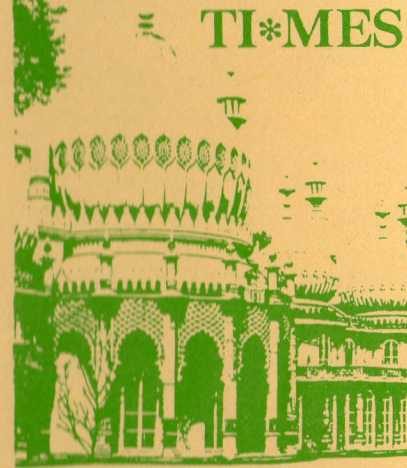
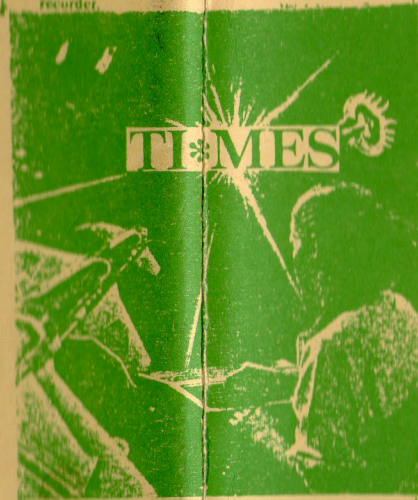
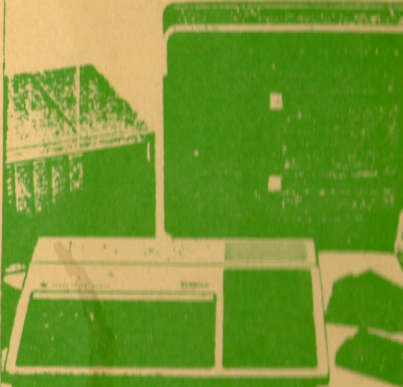


TI*MES
NEWSLETTER

TI*MES

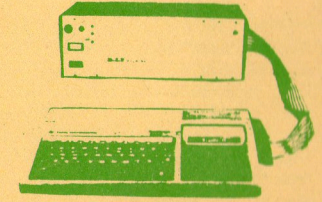


TI*MES



TI*MES

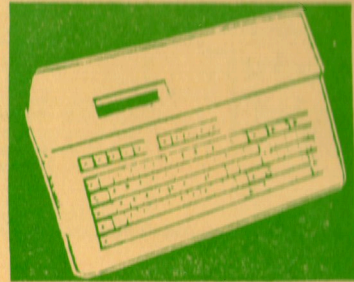
TI*MES



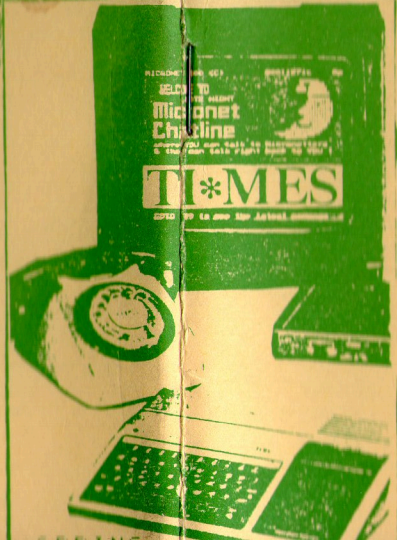
DIY EXPANSION TO BUILD YOURSELF



TI*MES



Myarc's 256K TI computer



TI*MES

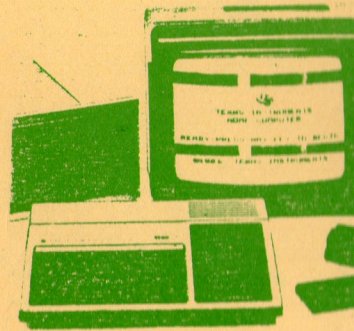
TI 99/4A NEWSLETTER

BRAND NEW! BRAND NEW! NEW!

- 10: NEW! 1984
- 11: Special Review
- 12: Your Letters
- 13: Features: computer arcade games
- 14: Programs: Jumping Jack
- 15: Reviews: For Under \$50
- 16: Round: How your keyboard is chosen
- 17: Super Computers, what to do
- 18: Links and Waps
- 19: Digital: review and your Hardware
- 20: Review: review & review page
- 21: Tip: Multiple Users
- 22: Classified page

TI 99/4A NEWSLETTER

TI*MES



TI*MES TI*MES TI*MES TI*MES TI*MES TI*MES TI*MES TI*MES TI*MES TI*MES
SPRING ISSUE NUMBER SIXTEEN

40, Barrhill, Patcham, BRIGHTON, East Sussex, BN18UF. Tel: 0273 503968 (evenings)

TI99/4a USERS GROUP U.K.

Great news! We have received an outstanding postbag of TI99/4a supporters. This means that for the first time ever, a National British TI99/4a Users committee will be elected. A list of persons willing to form the new TI group is printed in this issue. Please make every effort to attend the UK TI Faire and first AGM to lend your support and nominate your Committee and Chairperson.

Details of the UK TI Faire held in DERBY is devoted to a separate page. The 16th May also coincides with the TI Faires in Los Angeles USA and Ottawa CANADA. Subject to suitable telephone lines there is every chance of a live link up with both USA and CANADA via modem at DERBY!!!

You can see that the cover features fifteen issues of TI*MES published over the last four years. This issue (sweet sixteen) marks the start of a new era in the TI99/4a community. News of the MYARC 9640 is that delivery has been made in the USA. But not quite ready for the UK market?. Documentation of the system for both hardware and software will take some time to accomplish.

Next we see the dawning of the 99AT Expansion System, this sad to say is plagued with development problems. Those of you who subscribe to MICROpendium will have first knowledge of this project. By the way for those of you who don't

subscribe to MICROpendium can evaluate a small sample of the February issue. Look what you are missing.

Not in MICROpendium is an advertisement for the TRITON TURBO XT. The publisher John Koloen of MICROpendium has given us his view. However it does reflect a general view with regard to keyboard. But most groups in the USA very much welcome this addition to the TI99/4a community. Now TI99/4a can be part of the PC league. Congratulations TRITON.

Well, this is my last letter under the TI99/4a EXCHANGE banner, what happens in Derby is that this Group CONTINUES under the heading of TI99/4a USERS GROUP U.K. With a new organised committee all TI99/4a owners will benefit more than ever before. This leaves me to thank most sincerely all of you who took the trouble in filling in the questionnaire and confirming your continued support.

It really is good Tidings we did not desert you, because you will still have a 99er group that goes on and on and on..... Audrey and myself wish all the very best for the future.

Happy 99ing,



Clive Scally. Founder of UK TI USERS.

TI994a USERS GROUP U.K. and TI*MES magazine is supported only by its subscribers. This TI users Group is INDEPENDANT of Texas Instruments and is completely non profit making. TI*MES is published quarterly. The annual subscription runs May to May each year. Details from the Group secretary on application. Editorial etc is provided by group members, other WORLD WIDE TI User-Groups and other related sources. Views expressed are those of the writer and not necessarily those of TI994a Users Group U.K. Whilst efforts are made to ensure accuracy no responsibility can be accepted by TI 994a USERS GROUP UK as a result of the applying of such information found within the pages of TI*MES. You are invited to contribute copy for publication in TI*MES. If you would like to make a contribution please submit COPY ON A4 ONLY this MUST be TYPED AND SUPPORTED WITH A DISK OR TAPE if a program is included. Last DATE OF ACCEPTANCE OF COPY 30TH MAY FOR PUBLICATION in JULY. Unaccepted material will be returned ONLY if accompanied by a S.A.E. No material may be reproduced without credit to the author and TI994a USERS GROUP U.K. MICRO COMPUTERS OTHER THAN TI99 COMPATIBLES WILL NOT BE ADVERTISED FOR SALE IN TI*MES.

EDITED FROM....

THE MUSIC CORNER
BY JEFF GATLIN

NORTHEAST TARRANT HOME COMPUTER USERS' GROUP, HURST, TX.
January, 1984

Whenever I think about programming music, I think of the time it takes to key in all the CALL SOUND statements. My next concern is which programming trick to use to make the desired song 'sing' through the computer instead of spit and hiccup its way through. So far I have encountered 4 distinctive methods: 1) the CALL SOUND method, 2) the DATA method, 3) the GOSUB method, and 4) the ARRAY method.

CALL SOUND is simple: A series of CALL SOUND statements, each with different information, such as:

```
CALL SOUND(800,10,220,10,370,10)
CALL SOUND(800,165,5,247,5,415,5)
CALL SOUND(1600,110,0,227,0,440,0)
```

This produces smooth, precise sounds, but gets very tiresome for the nontypist. My first variation was setting up variables for all the notes to avoid constantly referring to the manual for the appropriate frequency. It worked, but didn't save any typing time or memory.

The DATA method involves setting up DATA statements which contain the notes, a READ statement to assign the notes to variables, a single CALL SOUND statement with variables read from data, and a clever FOR TO NEXT statement.
Example:

```
100 FOR REP=1 TO 4
110 READ A,B,C
120 CALL SOUND(400,A,0,B,0,C,0)
130 NEXT REP
140 RESTORE
150 GOTO 100
DATA 100,40000,40000,139,330,400,
147,40000,40000,165,277,440
```

The notes with frequencies of 40000 are used to create silence without having to key in a separate CALL SOUND statement with only one note. The disadvantage of this method is when you have a program with oodles of data (lots of notes), you occasionally get hiccups. The cause lies somewhere in the BASIC language. I've been told that basic generates garbage that has to be taken out occasionally. When the garbage is dumped, the computer hiccups causing the flow of the music to be interrupted. However, it usually takes quite a bit of data or an extreme tempo (speed) to cause hiccups.

The GOSUB method was introduced to me by Gerry Myers. It involves keying in one or more CALL SOUND statements (as needed) and follow them with a RETURN statement. Once done, your programming consists of redefining the CALL SOUND variables and adding a GOSUB statement to initiate the sound. Just to make things easier, set up variables for each note within an octave (see lines 100-110 below). Now instead of having a variable for every note, you alter the base variable (is this making sense). For example: BF is B flat, BF#2 is B flat 1 octave higher, BF#4 is 2 octaves higher, BF#8 is 3 octaves higher and so on. BF/2 is 1 octave lower, and BF/4 is 2 octaves lower. Explanations: well there is one, it's just that if I try to explain it correctly, you'll probably put down the article and grab the TV guide. Very simply: double the frequency of any note and you'll have a note one octave higher, halve the frequency of any note and you'll have a note one octave lower. Example:

```
100 B=493.88 :: AS,BF=466.16 :: A=440.00 :: GS,AF=415.30
:: G=392.00 :: FS,BF=369.99 :: F=349.23
110 E=329.63 :: DS,EF=311.13 :: D=293.66 :: CS,DF=277.18
:: C=261.63 :: R=40000 :: L=250
120 FOR REP=1 TO 3
130 X=G#2 :: Y=E#2 :: Z=C :: GOSUB 500
140 X=F#2 :: Y=D#2 :: GOSUB 500 :: X=E#2 :: Y=E#2 ::
GOSUB 500
150 Z=G/2 :: GOSUB 500 :: X=A#2 :: Y=F#2 :: Z=C :: GOSUB
500 :: GOSUB 500
160 X=BF#2 :: Y=G#2 :: Z=G/2 :: GOSUB 500 :: X=A#2 :: Y=F#2
GOSUB 500
170 X=BF#2 :: Y=G#2 :: GOSUB 500 :: GOSUB 500 :: GOSUB 500
180 X=A#2 :: Y=F#2 :: GOSUB 500
190 NEXT REP
200 X=G#2 :: Y=E#2 :: Z=C/2 :: GOSUB 510
499 END
500 CALL SOUND(L,X,0,Y,0,Z,0):: RETURN
510 CALL SOUND(L#7,X,0,Y,0,Z,0):: RETURN
```

In the above example, "L" is the defined length of the note. Notes longer than "L" can be lengthened with successive GOSUBs or altered in a separate SOUND statement (as in line 510). The "voices" are defined as X,Y,Z. X is the highest voice, Y is the middle, and Z is the bass. Although it is not necessary to keep them in that order, it does help make the editing of mistakes easier. Once a voice is defined, it will remain until you change it. This causes the illusion of sustained notes behind moving notes (lines 130, 140). Once again 40000 can be used for silence (defined as R but not used in the example).

The last method is the ARRAY method. It is similar to DATA in that all the notes are in DATA statements. However, instead of reading each voice and then playing them, the notes are read into ARRAYS, then played via a FOR TO NEXT statement. Even better, each array can be a musical line. The melody can be one array, countermelody in another, and bass line in a third.

Combining this method with a negative "duration" within

the SOUND statement creates a remarkably smooth and incredibly fast musical line. When a negative duration is specified, the previous sound is stopped and the new sound is started immediately. The first question that comes to mind is how do you use negative values without getting ridiculously fast music? Easy! Just put some sort of delay between the SOUND statements. In the Bach Invention example, I've used a math function that I saw used in a program by Robert Gagle. The statement "P=2.50" causes the computer to think for a few extra milliseconds before it plays the next sound. A higher number than fifty creates a longer delay and thus a lower number creates a shorter delay. Why? It's a mystery to me but it works like a charm so I don't complain. (To get an idea of how fast the AA can play,

change Line 240 to read "for N=1 to 104 :: SOUND(X,A(N),V1,B(N),V2):: NEXT N", remove line 270 and the program.)

Once you've completed your data statements, you write the data to a disk file to conserve program space. This can allow you to create programs that execute extremely long songs without running out of memory while in the middle of programming (it happened to me, really!)

In the example program, I've added the option of changing the volume of either voice while the program is running and without sacrificing the smoothness of execution (well, maybe a little, occasionally). Hope you enjoyed the program, and I hope this article has helped someone.

10 ! THIS PROGRAM USES 20 !ONE SOUND STATEMENT!!! 30 !AND TWO SIMPLE ARRAYS!! 40 ! 50 !PROGRAMMED BY 60 !JEFF GATLIN 100 CALL CLEAR 110 PRINT TAB(5):"INVENTIOK NO.13" 120 PRINT 130 PRINT TAB(7):"BY BACH" 140 FOR T=1 TO 5 :: PRINT :: NEXT T 150 PRINT "PROGRAMMED BY JEF F GATLIN" 160 PRINT 170 PRINT "CONTROL VOLUME OF VOICES USING '1'&'2' FOR	LOUDER AND '0'&'M' FOR S FTER" 180 DIM A(104):: DIM B(104) 190 FOR N=1 TO 104 :: READ A (N):: NEXT N 200 FOR N=1 TO 104 :: READ B (N):: NEXT N 210 X=-999 :: V1=10 :: V2=10 :: P=0 220 INPUT "READY? PRESS ENTE R.":UU\$:: GOTO 240 230 INPUT "PLAY AGAIN? PRESS ENTER.":UU\$ 240 FOR N=1 TO 104 :: CALL S OUND(X,A(N),V1,B(N),V2):: P= 2*50 250 CALL KEY(0,K,S):: IF K=4 9 THEN V1=V1-1 ELSE IF K=50 THEN V2=V2-1 ELSE IF K=61 TH EN V1=V1+1 ELSE IF K=67 THEN V2=V2+1 260 IF V1<0 THEN V1=1 ELSE I F V1>30 THEN V1=29 ELSE IF V	200 THEN V2=1 ELSE IF V2>38 THEN V2=29 270 NEXT N 280 GOTO 230 290 DATA 40000,659,880,1047, 988,659,988,1175,1047,1047,6 80,880,831,831,659,659 300 DATA 880,1047,1319,1047, 880,1047,740,880,1047,880,74 0,880,622,1047,988,880 310 DATA 831,988,1175,988,83 1,988,587,698,831,698,587,65 8,494,698,659,587 320 DATA 523,659,880,659,523 ,659,440,523,622,523,440,523 ,370,523,494,440 330 DATA 415,415,988,988,831 ,831,659,659,40000,659,880,1 047,988,659,988,1175 340 DATA 1047,880,1047,1319, 1175,988,1175,1397,1319,1047 ,1319,1568,1397,1319,1175,10 47	350 DATA 988,1047,1175,1319 1397,1175,1661,1175,1976,1 5,1047,1760,1397,1175,988, 75 360 DATA 831,988,1047,880, 9,880,988,831,880,659,523, 9,440,440,440,440 370 DATA 523,523,440,440,5 ,415,330,330,440,330,440,5 ,494,330,494,587 380 DATA 523,659,880,659,5 ,659,440,523,370,440,523,4 ,370,440,311,370 390 DATA 330,330,415,415,4 ,494,415,415,330,330,247,2 ,208,208,165,165 400 DATA 220,220,262,262,3 ,330,262,220,220,220,262,2 ,156,156,40000,40000 410 DATA 40000,494,415,330 94,494,415,294,262,262,330 30,208,208,330,330
---	--	--	--

TIP:

To speed up loading infocom games, don't use Extended Basic. Use Minimemory or EditorAssembler instead. To use these, select the load and run option and type DSK1.BOOT. When this is finished loading, press ENTER until you get the program name, then type START.

On the Minimemory, you will get an error after BOOT loads, but keep pushing enter and proceed as above.

(FROM HUGERS, June, 1985)



FLYING SAUCERS

```

100 REM ***FLYING SAUCERS***
110 REM (C)DAVID HOLMES 1984
120 REM MEMORY=5.2K.
130 CALL CLEAR
140 FOR L=128 TO 132
150 READ AS$
160 CALL CHAR(L,AS$)
170 NEXT L
180 DATA 98EFFF19,0019F7FF98,000
098EFFF19,00000019F7FF98,0000009
8EFFF19
190 CALL CHAR(35,"98EFFF19")
200 CALL CHAR(40,"18BA003333005A
18")
210 CALL CHAR(95,"00FF")
220 CALL CHAR(152,"001")
230 CALL CHAR(133,"18BA003333005
A18")
240 CALL CHAR(136,"00666666FFFF
F00")
250 CALL CHAR(137,"0000666666FF
FFF")
260 CALL CHAR(144,"242424242424
424")
270 RANDOMIZE
280 DEF NR=INT(RND*20)+3
290 DEF NC=INT(RND*32)+1
300 DEF NV=INT(RND*23)+5
310 DEF SR=INT(RND*12)+3
320 DEF N=INT(RND*3)
330 DEF J=(C1-C)/3
340 GOSUB 1760
350 HSC=0
360 M(0)=1
370 M(1)=-1
380 M(2)=0
390 PN=3
400 FG=3
410 SC=0
420 CL=8
430 PT=0
440 C1=5
450 C=10
460 EC=1
470 CALL CLEAR
480 FOR L=3 TO 8
490 CALL COLOR(L,4,1)
500 NEXT L
510 CALL COLOR(2,16,1)
520 CALL COLOR(16,SR,1)
530 CALL COLOR(13,16,1)
540 CALL COLOR(14,7,1)
550 CALL COLOR(15,11,1)
560 PR$="SCORE 0 HI~SCORE
"
570 ROW=1
580 COL=2
590 GOSUB 1710
600 GOSUB 2080

```



```

610 CALL HCHAR(2,1,95,32)
620 FOR L=6 TO 27 STEP 3
630 CALL HCHAR(24,L,136)
640 NEXT L
650 FOR L=1 TO 100
660 CALL VCHAR(NR,NC,152)
670 NEXT L
680 GOSUB 1520
690 FOR R=4 TO 24
700 CALL VCHAR(R-1,NV,152)
710 FOR D=128 TO 132
720 CALL VCHAR(R,C,D,P(0))
730 IF EC>0 THEN 840
740 CALL VCHAR(R,C+1,32,P(0)+1)
750 CALL VCHAR(R,C+3,D,P(1))
760 CALL VCHAR(R,C+4,32,P(1)+1)
770 CALL VCHAR(R,C+6,D,P(2))
780 CALL VCHAR(R,C+7,32,P(2)+1)
790 CALL VCHAR(R,C+9,D,P(3))
800 CALL VCHAR(R,C+10,32,P(3)+1)
810 CALL VCHAR(R,C+12,D,P(4))
820 CALL VCHAR(R,C+13,32,P(4)+1)
830 GOTD 930
840 CALL VCHAR(R,C-1,32,P(0)+1)
850 CALL VCHAR(R,C+3,D,P(1))
860 CALL VCHAR(R,C+2,32,P(1)+1)
870 CALL VCHAR(R,C+6,D,P(2))
880 CALL VCHAR(R,C+5,32,P(2)+1)
890 CALL VCHAR(R,C+9,D,P(3))
900 CALL VCHAR(R,C+8,32,P(3)+1)
910 CALL VCHAR(R,C+12,D,P(4))
920 CALL VCHAR(R,C+11,32,P(4)+1)
930 IF R<24-PN THEN 970
940 FOR L=0 TO 4
950 IF R+P(L)>23 THEN 1110
960 NEXT L
970 CALL KEY(O,K,S)
980 IF (K<50)+(K>57) THEN 1010
990 C1=K*3-144
1000 GOSUB 1280
1010 IF (C>5)*(C<15) THEN 1040
1020 EC=M(INT(C/10))
1030 GOTD 1050
1040 EC=M(N)
1050 C=C+EC
1060 IF H=PN*5 THEN 680
1070 NEXT D
1080 CALL HCHAR(R,C-2,32,34-C)
1090 GOSUB 1640
1100 NEXT R
1110 PR$="THE MARTIANS HAVE LAND
ED"
1120 ROW=12
1130 COL=5
1140 GOSUB 1710
1150 GOSUB 1970
1160 GOSUB 1640
1170 IF SC<HSC THEN 1200
1180 HSC=SC

```

```

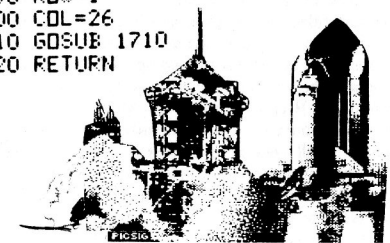
1190 GOSUB 2080
1200 PR$="PRESS ANY KEY FOR NEW
GAME"
1210 ROW=12
1220 COL=4
1230 GOSUB 1710
1240 CALL KEY(O,K,S)
1250 IF S=0 THEN 1240
1260 GOTD 390
1270 REM FIRE GUN
1280 CALL SOUND(-200,110,10,-4.5
)
1290 CALL VCHAR(24,C1,137)
1300 CALL VCHAR(24,C1,136)
1310 CALL GCHAR(R,C1,6)
1320 IF G<128 THEN 1460
1330 FG=R-1+P(J)
1340 CALL VCHAR(FG,C1,144,24-FG)
1350 CALL VCHAR(FG,C1,133,2)
1360 CALL VCHAR(FG,C1,32,24-FG)
1370 CALL SOUND(-100,110,4,-7,0)
1380 CALL SOUND(-200,110,8,-7,4)
1390 CALL SCREEN(SR)
1400 CALL SCREEN(SR)
1410 CALL SCREEN(2)
1420 H=H+1
1430 SC=SC+PT
1440 P(J)=P(J)-1
1450 RETURN
1460 CALL VCHAR(R,C1,144,24-R)
1470 CALL VCHAR(R,C1,40)
1480 CALL VCHAR(R,C1,32,24-R)
1490 CALL SOUND(-100,110,8,-7,4)
1500 RETURN
1510 REM RE-SET
1520 IF PN>19 THEN 1540
1530 PN=PN+1
1540 FOR L=0 TO 4
1550 P(L)=PN
1560 NEXT L
1570 H=0
1580 PT=PT+10
1590 CL=SR
1600 IF INT(CL/2)<>CL/2 THEN 159
0
1610 CALL COLOR(13,CL,1)
1620 RETURN
1630 REM DISPLAY SCORE
1640 PR$=STR$(SC)
1650 ROW=1
1660 COL=8
1670 GOSUB 1710
1680 CALL COLOR(16,SR,1)
1690 RETURN
1700 REM PRINT AT
1710 FOR L=0 TO LEN(PR$)-1
1720 CALL HCHAR(ROW,COL+L,ASC(SEE
G$(PR$,L+1,1)))
1730 NEXT L
1740 RETURN

```

```

1750 REM TITLES
1760 CALL SCREEN(2)
1770 PRINT "USE KEYS 2 TO 9 TO F
IRE GUNS":
1780 PRINT TAB(4);"### # # # #
# # ###":
1790 PRINT TAB(4);"# # # # #
# # #":
1800 PRINT TAB(4);"## # ### #
### #":
1810 PRINT TAB(4);"# # # # #
### # #":
1820 PRINT TAB(4);"# ### # #
# # ###":
1830 PRINT TAB(1);"### ### # # #
## ### ## ###":
1840 PRINT TAB(1);"# # # # # #
# # # # #":
1850 PRINT TAB(1);" # ### # # #
## ### #":
1860 PRINT TAB(1);" # # # # # #
# # # #":
1870 PRINT TAB(1);"### # # ### #
## ### # # ###":
1880 FOR L1=1 TO 10
1890 CALL COLOR(1,SR,1)
1900 RESTORE
1910 FOR L2=1 TO 4
1920 READ AS$
1930 CALL CHAR(35,AS$)
1940 NEXT L2
1950 NEXT L1
1960 REM SOUND
1970 FOR L=24 TO 0 STEP -0.5
1980 CALL SOUND(50,590,L,-3,L+3)
1990 CALL SCREEN(15)
2000 CALL SCREEN(2)
2010 NEXT L
2020 CALL SOUND(200,110,4,-7,0)
2030 CALL SOUND(300,110,8,-7,4)
2040 CALL SOUND(400,110,12,-7,8)
2050 CALL SOUND(500,110,16,-7,12
)
2060 CALL SOUND(4000,110,24,-7,2
0)
2070 RETURN
2080 PR$=STR$(HSC)
2090 ROW=1
2100 COL=26
2110 GOSUB 1710
2120 RETURN

```



MICROpendium

Mechatronic TI-Mouse

No squeaks from this input device

By JOHN CLULOW

The Mechatronic TI-MOUSE is a beautifully designed input device for the TI99/4A. It is priced a little lower than similar devices for Apple and Tandy computers and outperforms them in some respects.

The mouse is about the size of a TI command module and has a motion sensing ball on the bottom. When it is moved on a flat surface, the computer generates a corresponding motion on the screen. Two keys on the mouse are used to make menu selections, draw, etc.

Joysticks and trackballs allow motion in one of eight directions. The mouse, on the other hand, provides accurate representation of speed and the full range of possible directions making it the most natural and easy-to-use input device available. Anyone who has been frustrated while trying to use a joystick or trackball with TI-Artist will be overjoyed with the performance of the TI-MOUSE.

The TI-MOUSE cable is plugged into a small interface box which, in turn, connects to the console joystick port. An AC adaptor supplies power to the interface. While the manual suggests an 8x11-inch section of table be available for the mouse, I found that an area of only 4x5 inches was sufficient. A mouse displacement of one inch translates into about 96 pixels on the screen; the minimum movement for full screen motion is about 2.5 inches horizontally and 2 inches vertically.

Three demo programs are provided. An Extended BASIC program displays a pocket calculator on the screen. The mouse moves a hand-shaped sprite to operate the calculator. An assembly language Breakout program shows how the mouse works in an arcade type game. Full source code is provided. The third program is a demo version of Mechatronic's TI-DOS; a disk manager program. The mouse moves an arrow-shaped cursor for icon selection of various disk functions.

Review

Report Card

Performance	A
Ease of Use	A
Documentation	A
Value	A
Final Grade	A

Cost: \$99.95

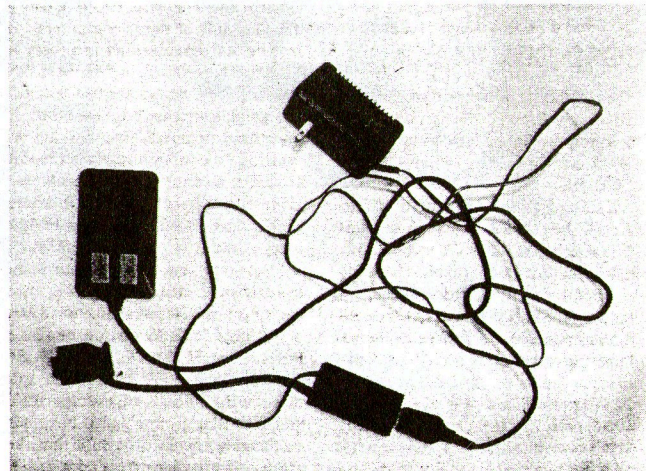
Manufacturer: Mechatronics GmbH, West Germany; distributed by T.A.P.E. Ltd., P.O. Box 4042, Ontario, CA 91761.

Requirements: console, memory expansion, disk system.

These examples illustrate how to include mouse routines in user-written programs. This is easy to do with the software provided. Loaded with a CALL LOAD statement, a program called MOUSE allows access to the device in Extended BASIC programs. Sprite number 1, used by the MOUSE program, can be assigned any desired color and pattern as usual. Three

CALL LINK routines are available: MAUS0 halts the main program until the sprite cursor is positioned as desired and the Mouse Key is pressed; MAUS1 is an interrupt routine which allows the Extended BASIC program to continue running while keeping track of the position of the mouse; and MCLR cancels the MAUS1 interrupt routine. MAUSE0 is analogous to an INPUT statement in BASIC; all processing stops until an entry is made. The interrupt routine, on the other hand, "automatically" keeps track of the mouse as the program continues execution. Interrupt routines, executed many times each second, tend to slow down operation of an Extended BASIC program so the MCLR routine should be executed whenever mouse input is not to be obtained. (An in-depth explanation of this principle can be found in J. Peter Hoddie's article "Attach a Sprite to a Joystick" in MICROpendium's Oct. '86 issue.)

CALL LOAD or CALL PEEK statements are used to read or assign mouse parameters, allowing a high (See Page 41)



MECHATRONIC MOUSE—

degree of control in Extended BASIC programs. A continuous set of nine bytes contain the data, allowing several parameters to be read or written with a single PEEK or LOAD. The parameters include the vertical and horizontal position of the sprite cursor, status of the mouse key, vertical and horizontal location of the "home" position and specification of the upper, lower, left and right boundaries. The sprite boundaries establish a "window" within which the sprite can be moved.

Evaluation with TI-Artist

Perhaps the most obvious use of the mouse would be with a program like TI-Artist. One of the many ways in which this program is unique is that it allows the user to specify a "DSR" (Device Service Routine) for new input devices. So I sent the mouse to Chris Faherty, author of TI-Artist, and he quickly modified the assembly program provided into a DSR. The only thing lost was the "rubber-band" effect.

In the process, Chris provided some feedback about the performance of TI-MOUSE relative to others he had worked with (the MAC mouse, Tandy, etc.). He said that the TI-MOUSE was much faster and much more responsive than others he had tried. He was also very happy with how easily the TI-MOUSE moved. He said some others were like sandpaper by comparison. Chris also told me that similar devices for other computers sold for 20 percent to 30 percent more than TI-MOUSE. TI-Artist users can obtain a copy of the DSR from T.A.P.E. or Insectbot.

Bill Sager, president of New Horizons Computer Club and TI-Artist user, offered to evaluate the mouse with TI-Artist. Bill said his kids (ages 6, 8 and 10) loved it and took to it right away. It greatly simplified use of TI-Artist for them.

He said that using the mouse speeded up TI-Artist somewhat and made it

much easier to draw curved lines, ovals, ellipses, etc. Drawing a straight line freehand was rather difficult, but the LINES, K-LINES and H/V LINES of TI-Artist could easily be used. In general, he said, he had much better control of the cursor.

Although it is possible to move the mouse too fast when drawing, causing it to skip pixels, Bill said that, as with many new tools, this just took a little practice to develop the right technique. The most prominent drawback was loss of the rubber-band effect, but Bill

was able to compensate the more he used the mouse. Also, because of the much greater responsiveness of the mouse, working in ZOOM mode was more difficult, but not impossible.

Over all, Bill felt that the mouse is a great addition to use of TI-Artist in that it eliminates the stiff action of the joystick and provides a more delicate touch when using the program.

The Mechatronic TI-MOUSE is a well-engineered product that is easy to use. I would highly recommend it to anyone in the market for a mouse.

Reviewed in MICROpendium

1984
 February: B-1 Nuclear Bomber, Tandon TM-100 Disk Drive, Void, Beamstalk Adventure, Microsurgeon, On Gaming, Database 50
 March: Star Trek. Escape From Balthazar, Garkon's Getaway, Sky Diver, Mail-Call, Prowler 8510 Printer
 April: Monthly Budget Master, Budget Master, Home Budget, Thief, Donkey Kong, Khe Sanh
 May: Companion Word Processor, Q*Bert, Mad-Dog I & II, Programs for the TI Home Computer
 June: Creative Expressions Accounts Receivable/Accounts Payable, CDC 9409 Disk Drive, Starship Concord, Lost Treasure of the Aztec, ASW Tactics II
 July: Theon Raiders, Introduction to Assembly Language for the TI Home Computer, Game of Wit, Pole Position
 August: TE-1200, Tower, Galactic Battle, Galaxy
 September: Wycove Forth, 99/4 Auto Spell-Check, QUICK-COPYer, Wizard's Dominion, Anchor Automation Mx XII Modern
 October: Killer Caterpillar, ZORK I, Defender
 November: 9900 Disk Controller Card/Manager, Super Bugger, Transar 1205 printer, Floppy-Copy, Data Base-X
 December: Gravity Master, Data Base Manager System, Learning 99/4A Assembly Language Programming

1985
 January: Super Sketch, Foundation Computing 128K Card, PFTERM-99, TI-Runner
 February: Super Extended BASIC, Beginning Assembly Language for the TI, ZORK II
 March: Morning Star Software CP/M Card, WDS/100 Winchester Disk Drive, Sketch Mate, BMC Color Monitor
 April: 9900 Micro Expansion System, Disk + Aid, Gemini

10X-15X
 May: Character Sets and Graphics Design, Draw 'N Plot
 June: GRAPHX, DATA BASE I
 July: Acorn 99, Advanced Diagnostics
 August: Model Dow-4 Gazelle, TI-Artist, PC-KEYS, Not-Polytopics' Bankroll
 September: Midnight Mason, Myarc 32K/128K Card, GRAP4X Companion
 October: 4A/TALK, Extended BASIC II Plus, XB Detective, Console Writer 2.1
 November: Foundation Z80A/80-column cards, 9900BASIC, Adventure Editor
 December: Display Enhancement Package, Triple Tech

1986
 January: BITMAC, Starcross
 February: Night Mission, Peripheral Diagnostic Module, BA-Writer
 March: Super Duper, Tunnels of Doom Editor, Business Graphs 99
 April: U.S. Open Tennis, PRBASE
 May: 4A Flyer, GRAM Kracker, Artist's Companion
 June: Myarc Disk Controller Card, Maximem
 July: Horizon RAMdisk, Old Dark Caves, Funwriter, TI99/4A Macro Assembler
 August: JOYPAINT 99, GPL Assembler, TI99/4A INTERN, GPL Linker
 September: Mechatronic 128K card
 October: TI-Forth Utilities, Co-Comp Memory Plus
 November: Submarine Commander, PEP, MAXRLE
 December: GK Utility I and II and GRAM Packer, X-10 Powerhouse, RAVE 99/101.

1987
 January: MG DISKASSEMBLER, Myarc XBII.

Back Issue Policy

Back issues of MICROpendium are available to subscribers only. Those wishing back issues may notify us of the issue(s) desired and include \$1.50 per issue desired in a check or money order. (U.S., Mexico and Canada; Texas residents add 6.25% sales tax.) For foreign airmail delivery, add \$2 per issue, 50 cents per issue surface mail. All prices listed are U.S. funds.

OUT OF STOCK: Vol 1, Nos. 1-2

P.O. Box 1343, Round Rock, TX 78680

JUST PLUG IT IN. IT'S READY TO USE!

MAKE THE IBM® CONNECTION TO YOUR TI-99/4A



TRITON
PRODUCTS COMPANY
ORDER CENTER
P.O. BOX 8123
SAN FRANCISCO, CA 94128-9986

SPECIFICATIONS

TRITON TURBO XT Personal Computer

MICROPROCESSOR: Intel 8088, 8/4.77 Mhz clock speed (software selectable).

OPERATING SYSTEM: MicroSoft® Disk Operating System (MS-DOS).

MEMORY: 256K RAM (Expandable to 640K).

DISK DRIVE: One 5¼" double-sided, double-density, 360K thin-line mini-floppy, 48 tracks per inch.

VIDEO: RGB/composite color graphics display adapter.

INTERNAL EXPANSION: Eight standard user-accessible IBM PC card slots.

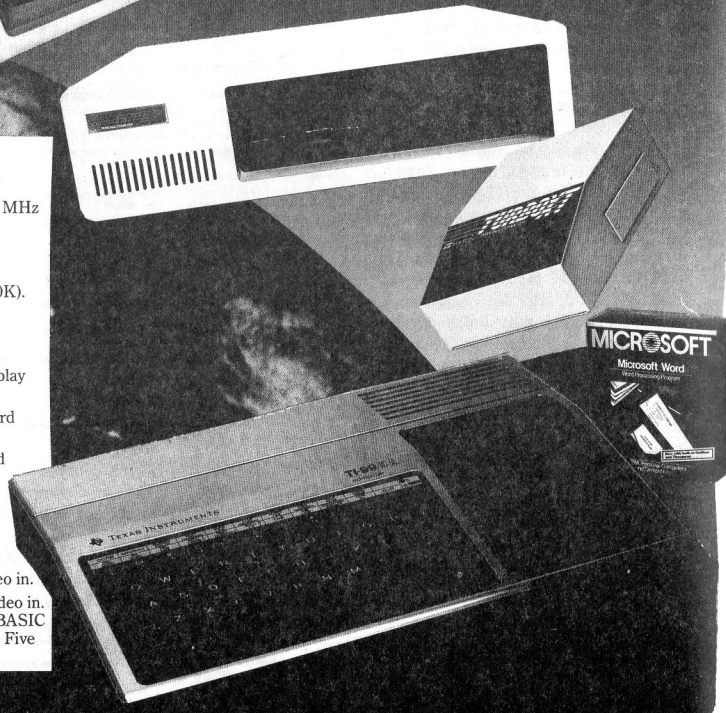
EXTERNAL CONNECTIONS: Standard parallel printer port, composite video, RGB, AC outlet.

POWER: 120VAC, 50-60 Hz.

BRIDGE BOX

4A MODE: Common video out, 99/4A video in.

XT MODE: Keyboard out, video in, XT Video in. Selectable power-up mode callable from TI BASIC or Extended BASIC. Concurrent processing. Five LED status display.



Triton schedules March 1 release of MG-designed PC clone system

The Triton Turbo XT Personal Computer, a product which allows IBM processing using a TI console, has been developed by MG for marketing by Triton Products of California. MG was formerly known as Miller Graphics.

The two-part system consists of a PC-type system unit that houses peripherals and cards and a bridge box that plugs into the side I/O port of the TI99/4A. A cable connects the bridge box to the PC unit.

The Turbo XT uses an Intel 8088 microprocessor with software switchable clock speeds of 8 and 4.77 Mhz. Also included is a color graphics card that supports RGB color and composite video, a floppy disk controller, one half-height DS/DD disk drive, parallel port and 256K RAM on the mother board. The mother board has sockets for up to 640K RAM. The system unit has 8 expansion slots, two of which are used by the CGA card and the disk controller, according to the manufacturer.

The bridge box has two modes: 4A and XT. The 4A mode includes a shared video output and a 99/4A input. The XT mode includes a keyboard output, video input and an XT video input. The box has five LED status lights.

The bridge box also contains a ROM for keyboard switching between 4A mode and XT mode and the ROM to convert the 4A key strokes into XT keycodes, the manufacturer says.

Mode switching from 4A to XT can be done through BASIC or XBASIC with CALL XT or by holding down FCTN CTRL ENTER on power up of the 4A, according to Triton.

Mode switching from XT to 4A is done by pressing the FCTN, CTRL and Enter keys, the manufacturer says.

The only items shared by the two systems are the 4A keyboard and the monitor. Although it is possible to get

80 columns out of a composite color monitor, the manufacturer says it is easiest to read with the color turned off in 80-column. The XT supports a 40-column mode. Graphics programs, such as games and drawing programs, work in 80 columns and most other software is readable, depending on the combination of foreground and background colors, the manufacturer says.

The manufacturer says that by not sharing the disk drives it is possible to do concurrent processing on the XT. According to Triton, the user can go into XT mode, start up communications software, log on to a BBS and start a download, then switch back to the 4A and do whatever he would like in 4A mode while the XT is still downloading from the BBS.

The manufacturer says the system has been tested on a number of 4A system configurations and found to be "very compatible," and says that since it is an IBM clone it is also fully compatible with both IBM software and IBM hardware, enabling the user to add any PC cards to the system.

Minimum 4A system requirements

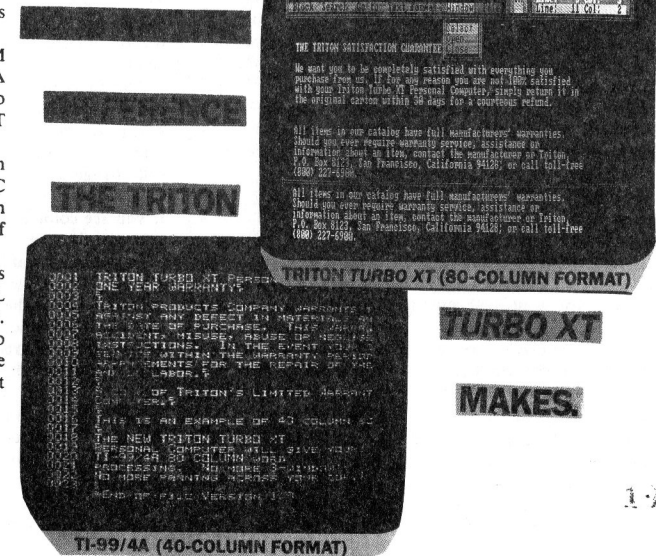
are a TI 99/4A console and a composite or RGB monitor.

Triton Products is handling the production of the bridge boxes and has contracted production of the Turbo XT.

The system has a 30-day money-back guarantee and a one year parts and labor warranty.

The cost for the basic system (Turbo XT, bridge box and cables) is \$499 plus \$19.90 for shipping and handling. Other configurations are available, including several that have a PC-type keyboard instead of a bridge box.

For additional information contact Triton at 1-800-227-6900, from 6 a.m. to 6 p.m. Monday through Friday and from 9 a.m. to 4 p.m. Saturday, Pacific time. Delivery of the product is scheduled to start March 1.



BASIC/XBASIC

Combining colors expands BASIC palette possibilities

By REGENA

When the TI99/4A was first being sold, it was the only computer that could have 16 colors on the screen in high resolution graphics.

Other computers may have advertised more colors, but they were limited in what color combinations could be used, or perhaps only four colors were really available in high resolution graphics. I got "spoiled" using my TI for graphics and had a lot of fun with the colors.

Now, of course, home computers are advertising even more colors (512 or 4096), and with the expanded memory capabilities of the present computers many more colors are available on the high-resolution screens. But wait—our little 16K computer still has a great capability for color graphics!

Are there really just 16 colors? Theoretically, yes, the BASIC commands allow for 16 color numbers. However, by using different combinations of colors, you can actually see many more colors. The short program included this month illustrates some of the colors.

First let's take a look at the color commands used in graphics in TI BASIC (or Extended BASIC). The colors are numbered from 1 through 16, and those numbers (or variables representing those numbers) may be used in the color commands. Color Number 1 is "transparent" and is actually whatever screen color you have at the moment. If you have a green (4) screen, then using Color Number 1 would be just like using Color Number 4. Therefore, we really only have 15 specified colors.

To specify a screen color, use CALL SCREEN(n) where n represents a color number. You may use this command anywhere in the program, and when the computer comes to the command the whole screen will change to the specified color. Even though we really have only 15 colors, you can actually see different shades because the color of the screen depends on how much printing or other characters are on the screen. For example, make a black screen, then start placing white stars on the screen. The more stars you place on the screen, the darker the screen color becomes.

Be careful with CALL SCREEN(2), which is a black screen. If you print in black you won't see the printing on a black screen. However, you can use this feature as a programming technique to make printing appear all at once. First change the screen to black and clear the screen, then print what you want, such as a title screen or a menu screen. Without changing any character colors you will be printing in black on the black screen and won't actually see the printing. After your printing is complete, change the screen color, such as CALL SCREEN(8). The screen changes from black to the specified color and you will see the printing.

The default colors for all characters are a black foreground with a transparent background. CALL COLOR(s,f,b) defines colors for characters. There are 16 color sets, each containing eight colors; "s" in the CALL COLOR statement specifies the color set number, "f" is the

foreground color and "b" is the background color. If you specify a background color other than transparent or the screen color, your characters will look like blocks on the screen with your defined colors. For example, change some of the alphabetic colors to black on white with CALL COLOR(7,2,16). Now print several words. Any letters in color set 7 will have the white background.

Just as the screen color appears to change as you have different things on the screen, the character colors will look different with different combinations of foreground and background colors. Now if you experiment with different foreground, background and screen colors, as well as surrounding colors, you'll really get a varying palette of colors. To illustrate this, I have written the following program for Color Combinations.

You will be asked to enter a screen color number, then a foreground color number, then a background color number. After the three numbers are entered, several patterns are shown on the screen. The first square has a checkerboard design alternating dots in a defined character. The second square shows a pattern of vertical lines and the third square shows a pattern of horizontal lines. The fourth and fifth squares use patterns of diagonal lines. You might wish to design your own patterns in the 8x8 character definitions.

The five patterns are printed with your specified foreground and background colors. The surrounding squares are printed in the color you specified as a screen color. I did not actually change the screen color but used that entry as the squares around the high-resolution patterns. You may prefer to adapt this program to actually change the screen color with CALL SCREEN(). The colors will appear slightly different than when you just use surrounding blocks.

Remember that whenever you use a CALL COLOR command, all characters currently on the screen are affected. This program continues to ask for screen color, foreground color and background color, so the patterns on the screen will change as you enter different numbers. After every three numbers entered, the patterns are drawn on the screen. If you see color combinations that you like, jot down the color numbers for use in your own programming. The program continues until you press FCTN-4 to stop.

Lines 120-180 clear the screen and print the instructions. Line 190 defines the first background color as transparent. Lines 200-260 define the high-resolution characters used in the patterns. Character 97 is the checkerboard pattern for the first square. Character 98 is the vertical line pattern for the second square. Character 99 is the horizontal lines pattern for the third square. Characters 100, 101 and 102 are used for the diagonal lines pattern in the fourth square. Character 103 is used for the diagonal lines pattern in the fifth square.

Line 270 defines L\$ for use in printing the surrounding colored blocks. Lines 280-400 ask for the color numbers,

BASIC/XBASIC—

then define those colors chosen. Lines 410-450 print the pattern blocks, then Line 460 branches back to Line 280 to continue asking for color numbers.

Please release the ALPHA LOCK key to types Lines 270, 430 and 440. The characters defined for high-resolution are in sets 9 and 10 so the lowercase letters can be used to PRINT the graphics rather than using CALL HCHAR and CALL VCHAR.

Lines 470-630 contain the subroutine to receive the color numbers using CALL KEY. The key pressed has ASCII Code K and must be a number key. If the first number pressed is greater than 1, C1 is the color number and the computer returns to the main program. If the first number pressed is 0 or 1, the color number then can be a two-digit number and another CALL KEY is necessary.

```
500 CALL HCHAR(23,C,32)
510 IF (K<48)+(K>57) THEN 480
520 CALL HCHAR(23,C,K)
530 C1=K-48
540 IF C1>1 THEN 630
550 CALL SOUND(100,1400,2)
560 CALL KEY(0,K,S)
570 CALL HCHAR(23,C+1,63)
580 CALL HCHAR(23,C+1,32)
590 IF (K<48)+(K>54) THEN 560
600 IF (K=48)+(C1=0)=-2 THEN 550
610 CALL HCHAR(23,C+1,K)
620 C1=C1*10+(K-48)
630 RETURN
640 END
```

Color Combinations

```
100 REM COLOR COMBINATIONS
110 REM BY REGENA
120 CALL CLEAR
130 PRINT "COLOR COMBINATIONS"
140 PRINT : : "ENTER A SCREEN COLOR NUMBER"
150 PRINT : "A FOREGROUND COLOR NUMBER,"
160 PRINT : "AND A BACKGROUND COLOR"
170 PRINT : "NUMBER. PATTERNS WILL BE"
180 PRINT : "SHOWN USING THOSE COLORS.": : :
190 B=1
200 CALL CHAR(97,"55AA55AA55AA55AA")
210 CALL CHAR(98,"AAAAAAAAAAAAAAAA")
220 CALL CHAR(99,"FF00FF00FF00FF")
230 CALL CHAR(100,"9224499224499224")
240 CALL CHAR(101,"4992244992244992")
250 CALL CHAR(102,"2449922449922449")
260 CALL CHAR(103,"99CC663399CC6633")
270 L$="hhhhhhhhhhhhhhhhhhhhhhhhhhhh"
280 PRINT "SCREEN NUMBER:"
290 C=19
300 GOSUB 470
310 CALL COLOR(10,C1,C1)
320 PRINT : "FOREGROUND COLOR:"
330 C=22
340 GOSUB 470
350 F=C1
360 CALL COLOR(9,F,B)
370 PRINT : "BACKGROUND COLOR:"
380 GOSUB 470
390 B=C1
400 CALL COLOR(9,F,B)
410 PRINT : : :
420 PRINT L$
430 PRINT "haahhhbbhhhhcchhhdehhhhgh"
440 PRINT "haahhhbbhhhhcchhhfhhhhgh"
450 PRINT L$: : :
460 GOTO 280
470 CALL SOUND(100,1400,2)
480 CALL KEY(0,K,S)
490 CALL HCHAR(23,C,63)
```

The LEADING monthly devoted to the TI99/4A

MICROpendium really does keep you informed & urge you to support this

Subscription Fees

\$17 for 12 issues via domestic third class mail
 \$20.50 for 12 issues via domestic first-class mail
 \$20.50 (U.S. funds) for 12 issues Canadian or Mexican delivery
 \$23.50 (U.S. funds) for 12 issues foreign delivery via surface mail
 \$37.00 (U.S. funds) for 12 issues foreign delivery via air mail
 (Texas residents add 87 cents sales tax)

Send me the next 12 issues of MICROpendium. I am enclosing \$ in a check or money order in U.S. funds (Texas residents add \$1.06 cents sales tax.) Mail to: MICROpendium, P.O. Box 1343, Round Rock, TX 78680

Name _____
 Address _____
 City _____
 State _____ ZIP _____

Comments

What about Triton's PC clone?

1987 promises to be a big (and perhaps controversial) year for TI users, what with the imminent release of Myarc's 9640 and Triton/MG's Turbo-XT clone. We've got an article this month about the Turbo and were hoping to get a mini-review of the 9640, but that will come next month.

Although I have not used either of the new machines, my intention is to purchase only one—the Myarc 9640. I think it represents an innovative approach to system upgrading and at a reasonable cost. The Turbo-XT, on the other hand, doesn't break any new ground. In fact, it's nothing more than a garden variety PC clone, judging from how Triton describes it. The only reason we even published a story about it was because of the little bridge box that is used to connect the TI99/4A keyboard to the Triton system unit. Otherwise it is indistinguishable from any of the clones advertised in the pages of Computer Shopper.

One illuminating detail about the Turbo XT is that among the various configurations Triton offers, several do not include a bridge box, just a PC-type keyboard. Anyone who buys a Turbo with a bridge box will probably want to replace it and the 99/4A keyboard with a PC keyboard. Simply put, the 99/4A is a very poor keyboard to use with a PC. The absence of dedicated function keys and other keys to match a standard PC layout will make the Turbo XT a frustrating experience. My bet is that the bridge box will go the way of last year's Christmas tree as soon as purchasers realize that a PC keyboard can be had for less than \$70 virtually anywhere.

My recommendation to anyone in the market for a PC is to buy a PC, whether the Triton system or another. Forget about the bridge box, which is nothing more than a substitute for the PC keyboard a PC should have. You want one, buy the whole thing.

If it's expansion you want, you may want to wait, as we are, for the release of the 9640. This system will use your existing software and peripherals and provide ample room for growth. Right off the bat you'll get a PC-type keyboard, 9640 PEB card, enhanced version of TI-Writer and an 80-column version of Multiplan. Also Advanced BASIC III, a Pascal run-time program (can't be used for programming in Pascal) and Myarc's disk operating system. Beyond this, the system requires a analog RGB monitor but will operate with your existing peripherals. Like the Turbo XT, the 9640 won't have a cartridge port. Unlike the Turbo, you'll can dump your cartridges to disk and access them from there. (Devices such as MG's excellent GRAM Kracker won't be of use with the 9640).

Which is not to say that the TI99/4A needs to be replaced. The only computer I have at home is a fully-configured TI, and it's the machine I use for most of my business and recreational computing. It is still the computer I turn to late

at night when it's time to write an article or work on a spreadsheet. It's the computer I run our accounts receivable out of and it is the computer I use for telecommunications. (I'm not an aficionado of computer games. About the only game I play on any system is U.S. Open Tennis by Nicesoft, and that's because I play tennis.)

As a result of the introduction of these two machines I'm getting a larger than normal number of calls from programmers and others wondering where the heck the TI market is going. I may as well tell you what I tell them:

- I think those who use their TI's a lot will continue to use them, or a compatible.

- Those in the TI community will continue to watch what develops with the 9640 and the Turbo XT, though for the life of me I can't see why the Turbo XT should be viewed as anything more than a PC clone.

- Support for the TI will continue much as it has over the past three years. If the 9640 is successful, expect to see a flurry of new or enhanced TI99/4A software.

- Hobbyists and programmers throughout the western world will continue to experiment and produce scores of innovative products designed for exclusive use with the TI99/4A.

- Barring a natural disaster or accident, when 1988 rolls around, we will still be publishing a periodical devoted to the TI99/4A and compatibles.

THE NEW HARD DISK

As exciting as things get with these new computers, don't overlook Myarc's hard disk controller. For \$265, plus \$45 for cables, users can control hard disks with capacities of up to 240 megabytes. (Check with Myarc on hard disk specifications.) Myarc is offering a 20-megabyte hard disk with a 60 ms seek time. The package price is \$750. Of all the equipment I have connected to my TI, the piece I value most is the old WDS/100 hard disk. (It seems as if I'm plugging Myarc a lot in this column, but that's because the company is coming out with some very interesting products.)

FLIGHT SIMULATOR

Not-Polyoptic's Spad XIII flight simulator should be on the market this month. We've seen a version that is "75 percent" complete, and it represents an improvement over TI Flyer. In Spad XIII you control a World War I pursuit plane whose mission is to shoot down a couple of German planes and balloons and bomb a German airfield. Then you may fly a loop around the Eiffel tower and head back to your landing field. The program does a good job at duplicating the conditions of flying a very light plane. It is aerodynamically correct, and uses graphics and sound to good effect. I'm looking forward to seeing the final version.

—JK

The market for computer diskettes is tough and during this year will get even tougher. There will undoubtedly be companies dropping out along the way, a forecast made by industry observers on both sides of the Atlantic. The temptation of some suppliers will be to cut quality in order to cut costs and stave-off the day of their eventual demise. The danger for the user is obvious — poor quality diskettes can mean not just money wasted on short-lived media but a complete loss, or corruption, of valuable data for your business. The cost of that can only be imagined.

The market is heavily price driven and only very efficient manufacturers will continue to make good media under those circumstances. It does mean, however, that the difference between a good quality and a junk diskette can be counted in pennies and it is just not worth cutting corners buying media of dubious parentage.

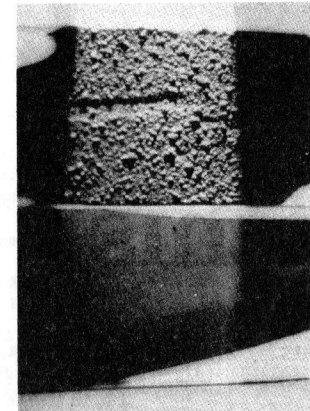
But how can you tell whether a diskette is good, bad or indifferent? Certainly a more expensive disc is not necessarily any better than a cheaper one! Surprisingly, you can make a start by looking at the jacket quite apart from the diskette itself.

The diskette jacket should have radiused edges on all four sides. This means that the disc drive entry slot is presented with a curved surface which will tend to find its own way in, in spite of the user's shaking hand. A diskette jacket with ordinary flat edges will present a vertical surface to the diskette slot which could — and invariably does — catch on the upper or lower edge of the disc drive slot itself. The result is a distorted diskette jacket which will eventually stop the diskette spinning freely and cause jamming, possible drive damage and certainly destruction of some of the information on the disc.

Of the 90 or so diskette manufacturers only seven provide the diskette jackets with properly radiused edges. The problem is that the radius fold is difficult to achieve in a high-volume manufacturing plant and the investment in the precision equipment is high as is the discipline needed among production line work people. However, the precaution is well worth taking. The radiused edge, also allows jacket thickness to be more precise. Too thin a jacket will cause the diskette to rub too much on the inside as it turns so wear will be excessive. Too fat a jacket allows the diskette to 'billow', catching the centre hole as it is taken in and out of the drive.

For other diskette quality checks it is best now to turn to the diskette manufacturer's literature. Any reluctance on the dealer's part to provide technical literature should be treated with obvious suspicion.

In that literature, thankfully subject to the strict Trades Descriptions Act, there will be a few figures among the host of performance statistics. Physical smoothness of the finished lubricated surface of the disc is an important assessment of manufacturing



Don't take the rough with the smooth

The floppy fiasco

accuracy. A smooth surface keeps the drive head and disc wear to a minimum and prolongs high performance read/write activities. This is because the drive head is continuously in touch with the disc surface and irregularities will easily 'bounce' the head away causing errors. Surface smoothness is usually quoted in terms of Centre Line Average and a figure of, or less than 0.76µ inches should be looked for. The photograph, in fact, shows the difference between a well burnished diskette surface and a badly finished one — well magnified of course but making the point quite dramatically. A delicate balance has to be achieved between diskette surface and read/write hardnesses — an imbalance will mean one wearing out the other.

Many so-called diskette manufacturers are not manufacturers at all. They simply buy bulk rolls of already coated material and stamp out circular diskettes from it. Others start with raw polyester, formulate magnetic coatings and check quality standards right through to the finished diskette in its jacket. It is in the original coating stage that one of the diskette's main qualities is formed — dispersion. This is the uniformity with which magnetic particles are spread on the surface and has the greatest bearing on signal consistency and therefore on read/write accuracy. A fine dispersion quoted as 1.77 x 10⁻⁶ µm nominal, is necessary to give consistent quality and signal amplitude averaging less than 125%

at 1f on track 00 and more than 80% at 2f on track 79. These kind of measurements, whether they mean anything to you or not, can easily be checked from the literature and are necessary in today's high coercivity diskettes now being used typically in the 3.5 inch and 5.25 inch sizes.

When squeezing one megabyte of data into a 3.5 inch diskette and 1.6 megabytes on to a 5.25 inch diskette even slightly uneven dispersion can cause a single bit of information to fall on an area which is insufficiently magnetised to record it. There are modern manufacturing methods which achieve these dispersion standards more rapidly than the old-fashioned wipe-on treatment once given to the magnetic coating and its associated binding agent.

The use of better quality bonding agents, manufacture in dust free environments and so on are all aspects which increase the quality of diskette life but are not capable of being verified when you go to buy a diskette from your retailer so there is no point in my dwelling upon them here. One last item which you will find in the diskette literature, however, is the quoted life to the diskette. Most manufacturers test diskettes on each track — though not many of them also test for errors between tracks — and will offer a greater or lesser form of guarantee.

Many manufacturers also quote a minimum specified life in terms of millions of passes. Five million passes, for example, may be quoted for 3.5 inch and 5.25 inch high coercivity diskettes. Taken literally, of course, the figure demonstrates a performance potential out of all proportion to the user's need. Five million passes is equivalent to accessing the same spot on the same disc once every working minute of a person's career. For the expert, however, the figure is taken as a guide from which he can calculate that the diskette will not deteriorate prematurely. The average user, however, should simply gain confidence from the fact that the manufacturer is not afraid to quote the minimum life of the disc in terms of 'head' passes.

Don't hope for the best

Hopefully this short discussion will have impressed upon readers that there is more to buying a diskette than picking a merchandised product from a shelf and hoping for the best. A few minutes spent with the retailer can at least identify makes of diskette which apparently fall short of the standards suggested here.

The Author would be interested to hear from readers who meet problems like this which are not satisfactorily answered by the dealer. It is in the manufacturers' interests to make sure that today's computer user treats magnetic media with the importance it deserves in spite of its lowest-ever price.

Don Miller, Wabash DataTech Intl.

Covering the TI99/4A, the Myarc 9640 and compatibles



RAMBLES by Stephen Shaw 1987

Hello again. Thanks to those of you who wrote in - but still no requests for articles on TI Basic or even Extended Basic, so it seems our readers have no interest in that area.

How did you fare in the telephone strike? Here in Manchester, we were hit a week earlier than the rest of the country and it dragged on a few days after as well. The result was a considerable decline in the number of lines into Manchester, leading initially to a false "engaged" signal, and later on to a deep-space type sound! Unfortunately operators outside Manchester did not seem to know of our problems...

The past quarter has seen another bundle of graphics and graphics programs arriving, on which more later! The principle freeware newcomer must be CREATIVE FILLING SYSTEM by Mark Beck, which I have to Jan 1987, "version 5", taking up THREE disks! CFS is a database program, with a whole variety of goodies on offer. One or two key words to search on, searching all fields or just a sorted field, can create subfiles, TI Writer value files, and one printed field can have the result of a math operation on two other fields. Plus vertical totals and sub totals. And so much more. The fields are of fixed length, so if you want to close the printing up you must make use of the TI Writer value file option. Nice program - for details of this and other Freeware/Public Domain programs please send me a blank disk and return postage- 10 Alstone Road, STOCKPORT, Cheshire, SK4 5AH.

This will appear before the big meeting in May, so let me just say- I'll be there, British Rail permitting. I won't be doing any disk copying or selling or anything, just milling around... and I'll wear a neat little name badge too! Hope to make it for 1pm.

And Richard- you'd better have a Geneve computer there! I can't afford one, but I'd surely like a look! (Review issue ham?). Now that Millers Graphics have effectively given us up - new name (MG), dropped Gramcracker, and involved with an IBM clone promotion which represents one of the poorest computer offers ever made... keep supporting Lou Philips at Myarc folks!

From the Ottawa User Group I have an excellent program that prints TI Writer files sideways. Some nice touches- the spacing between lines is adjusted to centre your letter and spread it out neatly. There are two print densities supported, and columns can be any number of characters across. This program uses either of the two rom resident character sets, or a disk based character set in a file set up for use with TI Writer, occupying 5 disk sectors of "PROGRAM" format.

Investigating the disks I have, I have found 5 character sets, which added to the two resident and two supplied, give NINE possible character sets. And in repairing some files for use with Sidewriter, I found out just how easy it was to prepare your own CHARAI file for TI Writer or Sideways Print.

MAKING YOUR OWN CHARAI FILE:

It helps if you start with a 5 sector CHAR file! You will need a utility to read and write disk sectors too. Place your CHARAI file onto a newly initialised disk. Now read Sector >22.

The character definitions occupy Sectors 22 to 25 (the other sector used is the directory sector). The first two words must be >0000 >0800 for Sideways Writer to use- TI Writer seems less fussy about the second word. The third word is the file length, and you should make it >03FD.

All the rest of the files is a sequential definition of characters 0 to 126, with each character occupying 4 words (or 8 bytes).

At the end of the first sector, the last word is the first two bytes defining character 31. The second sector begins with the remaining 6 bytes defining character 31. And so on.

The 4th to 7th word in the second sector defines the SPACE, Character 32: you will see this as >0000 >0000 >0000 >0000

Easy that one eh!

If you are used to using CALL CHAR, you will think of a character defined as 16 hexadecimal numbers (0 to F), with each pair of numbers representing one pixel row. To define a CHARAI file we must think of a character as being defined by just 8 hexadecimal numbers, each having two "digits".

Here are some equivalents:

```
CALL CHAR: CHARAI: !! CALL CHAR: CHARAI:
"01".....>01.....!!..."0F".....>0F
"10".....>10.....!!..."20".....>20
"1A".....>1A.....!!..."30".....>30
and so on.
```

Thus a vertical line, which we would define with:

```
CALL CHAR(N,"1010101010101010")
would appear in a CHARAI file as:
>1010 >1010 >1010 >1010
```

Not so hard is it! Now you can define your own fonts to appear on screen with TI Writer, or to print using the SIDEWAYS PRINTER program.

NB: TI Writer uses a 40 column screen, so character definitions for screen display must be in a 6x8 block, but the SIDEWAYS PRINT program can handle the full 8x8 block.

Farewell to John Rice, leaving us for Amstrad Country. Hope your Amstrad will last as long as my TI console- after your new Amstrad has been repaired that is...

MACHINE CODE:

Graham Marshall sent me two machine code utilities for Extended Basic to share with you through Rambles.

The first is a STRING ARRAY SEARCH.

Take an array A\$() with 30 elements, and you want to find out which element has the word "RAMBLES" in it. In extended basic you would use:

```
100 FOR T=1 TO 30
110 IF A$(T)="RAMBLES" THEN 150
120 NEXT T
130 PRINT "RAMBLES NOT FOUND"
140 STOP
150 PRINT "ELEMENT ";T;"HAS IT"
160 END
```

Which all takes time - especially if the string array has many more elements and the word we are looking for is at the end.

Using Graham's utility we would enter:

```
100 CALL LINK("SEARCH",A$( ),"RAMBLES",T)
110 IF T>0 THEN PRINT "ELEMENT ";T;"HAS IT"
120 END
130 PRINT "RAMBLES NOT FOUND"
140 END
```

All much faster.

I did have a little problem using this routine at first - it treats a nul element as a marker for the end of the array, and I filled the array, leaving no nuls! (a nul string is ""). The result was a BAD SUBSCRIPT error.

The routine is listed below in several formats:

1. Source code for XB, Ed/As or Myarc ExBas
2. As CALL LOADS for those of you without a disk drive
3. The object code as a text file if you prefer to key it in with TI WRITER - remember to save this with the PRINT FILE option, using the option F to save it in FIXED 80 format!

and finally, a demonstration program!

```
*****
*SEARCH ROUTINE IN EXTENDED BASIC*
*FORMAT-CALL LINK("SEARCH",ARRAY,*
*STRING,VARIABLE). *
*****
DEF SEARCH
*****
* FOLLOWING LINES ARE FOR *
* TI EXTENDED BASIC ONLY *
*****
NUMASG EQU >2008
STRREF EQU >2014
XMLLNK EQU >2018
*****
* IF YOU ARE USING ED/AS OR
* MYARC EX BAS THEN INSTEAD OF
* THESE THREE EQUATES YOU MUST USE:
* REF NUMASG,STRREF,XMLLNK
*****
FAC EQU >B34A
STATUS EQU >B37C
CIF EQU >20
COUNT DATA 0
STR1 BYTE 255 * ARRAY STORAGE
BSS 255
STR2 BYTE 255 * STRING STORAGE
BSS 255
EVEN
MYREG BSS 32
RTRN DATA 0
```

* ---> MORE MORE --->

I have seen ads for a product from Triton, in which MG (formerly Millers Graphics) are involved, which is pretending to make your TI an IBM compatible. No it doesn't. What they have done is take an IBM clone, connected the TI keyboard to it, and put in a switch so that the keyboard works as a normal TI with its own PEB and peripherals, OR the keyboard links to the IBM clone. What a really dotty idea. The two systems are quite separate, and merely share the keyboard, and the TI keyboard is hardly ideal to run IBM programs...

- Wanted: U.K. written commercially published programs, (no longer available ones!), on disk please, I no longer have tape capability. I bought as many programs as I could, but inevitably missed as many again. Programs which seem too long for disk storage may be helped by using CALL FILES(1) or if that doesn't work, you can OLD an XB program into TI Basic and save it to disk- you have a little more memory in TIB.

- Wanted- unwanted TI related books for collection purposes. (TI/Forth/C).

- Regret funds do not allow for direct purchase, but disks sent will be rerecorded with anything from the disk library free, while postage will be covered on books- tell me what you have before you send anything though! -And I do have everything Stainless Software published thanks!

```

*
SEARCH MOV R11,@RTN
      LWPI MYREG
      LI R0,0 * MOVE STRING TO BUFFER IN RAM
      LI R1,2 * STRING IS 2nd PARAMETER PASSED
      LI R2,STR2
      BLWP @STRREF

*
LOOP  LI R1,1 *MOVE STRING ARRAY TO BUFFER IN
      INC R0 *RAM
      LI R2,STR1
      LI R3,255
      SWPB R3
      MOVB R3,@STR1
      BLWP @STRREF

*
      MOV @STR1,R4 *COMPARE THE STRING TO THE ARRAY
      SWPB R4 *STRING
      ANDI R4,>00FF
      MOV R4,@COUNT * SET R4 TO LENGTH OF CURRENT ARRAY ELEMENT
      CI R4,0 * IS LENGTH ZERO?
      JNE CONT *IF THE ELEMENT IN THE STRING
      CLR R0 *IS "" THEN FINISH
      JMP FINISH

CONT  LI R1,-1 * TRY SETO R1 INSTEAD OF LI R1,-1
LOOP1 INC R1
      MOVB @STR1(R1),R2
      MOVB @STR2(R1),R3
      CB R2,R3
      JNE LOOP *IF NOT EQUAL GET NEXT ARRAY STRING
      C R1,@COUNT
      JNE LOOP1 *COMPARE NEXT TWO CHARACTERS IN THE STRING
FINISH MOV R0,@FAC
      BLWP @XMLLNK
      DATA CIF *WRITE TO THE NUMERIC VARIABLE
      LI R0,0
      LI R1,3 *SEND RESULT OUT TO THIRD PARAMETER IN CALL
      BLWP @NUMASG
      CLR R0 *DOES IT MATTER IF YOU OMIT THIS?
      MOVB R0,@STATUS
      LWPI >B3E0
      MOV @RTN,R11 *RETURN TO XB
      RT
      END

```

Did you see Mike O'Regans name in PCW? Mike was a founder member of the first UK User Group, and had many programs printed in the magazines. His name appeared in PCW as a founder member of the Psion Organiser User Group!!!
 Where are they now.... well, Pete Brooks and myself were founder members. Anyone else left from those early days???

Assemble with R option only for TI ExBas! Myarc ExBas can handle C option as well!

```

1 REM SEARCH ARRAY UTILITY      BY GRAHAM MARSHALL
2 REM REQUIRES XB + 32K RAM     USE:          CALL LINK("SEARCH",ARR
AY$( ),FIND$,ELEMENT)
100 CALL INIT
110 CALL LOAD(16376,83,69,65,82,67,72,39,24)
120 CALL LOAD(8194,39,158,63,248)
130 CALL LOAD(9460,0,0,255,0,0,0,0,0,0,0,203,20,203,53,203,78,203,231,204,71)
140 CALL LOAD(9482,204,150,204,228,205,29,205,75,205,96,33,131,35,253,38,184,40,183,41,182,42,195)
150 CALL LOAD(9504,43,193,44,179,45,194,47,196,58,181,59,180,60,191,61,190,62,192,94,197,255,58)
160 CALL LOAD(9526,58,130,65,84,240,71,79,133,73,70,132,79,78,155,79,82,186,80,73,221,84,79)

```

!!! MORE MORE -----> !!!

```

170 CALL LOAD(9548,177,255,65,66,83,203,65,76,76,236,65,78,68,187,65,83,67,220,65,84,78,204)
180 CALL LOAD(9570,66,89,69,3,67,79,78,1,67,79,83,205,68,69,70,137,68,73,77,138,69,78)
190 CALL LOAD(9592,68,139,69,79,70,202,69,88,80,206,70,79,82,140,73,78,84,207,76,69,78,213)
200 CALL LOAD(9614,76,69,84,141,76,79,71,208,77,65,88,223,77,73,78,224,78,69,87,0,78,79)
210 CALL LOAD(9636,84,189,78,85,77,4,79,76,68,5,80,79,83,217,82,69,67,222,82,69,77,154)
220 CALL LOAD(9658,82,69,83,6,82,78,68,215,82,85,78,169,83,71,78,209,83,73,78,210,83,81)
230 CALL LOAD(9680,82,211,83,85,66,161,84,65,66,252,84,65,78,212,86,65,76,218,88,79,82,188)
240 CALL LOAD(9702,255,66,65,83,69,241,66,69,69,80,238,67,65,76,76,157,255,0,82,36,214,68)
250 CALL LOAD(9724,65,84,65,147,5,130,16,54,152,20,32,179,22,8,194,2,5,136,136,8,32,108)
260 CALL LOAD(9746,21,3,19,2,217,20,0,1,136,2,32,108,17,239,6,160,43,124,16,237,216,32)
270 CALL LOAD(9768,32,179,131,117,16,204,216,32,32,202,131,117,16,200,6,160,39,52,4,224,32,20)
280 CALL LOAD(9790,194,1,162,32,32,126,136,8,255,212,20,4,192,72,160,224,32,126,16,9,160,224)
290 CALL LOAD(9812,32,126,136,3,255,212,17,3,192,224,255,212,6,3,192,67,4,224,32,130,4,224)
300 CALL LOAD(9834,32,106,4,224,32,18,194,224,32,6,19,2,192,67,5,129,4,96,34,86,6,160)
310 CALL LOAD(9856,39,52,4,224,32,20,194,1,98,32,32,126,17,5,192,72,96,224,32,126,17,2)
320 CALL LOAD(9878,16,230,4,193,4,195,16,227,6,160,46,218,4,224,32,20,200,1,32,8,4,32)
330 CALL LOAD(9900,162,222,4,224,32,162,136,1,255,212,17,4,7,32,32,162,192,65,227,16,220)
340 CALL LOAD(9922,6,160,39,44,4,96,36,20,6,160,39,44,6,1,128,193,17,2,4,96,34,86)
350 CALL LOAD(9944,4,224,32,20,192,195,22,1,4,193,192,193,17,253,16,246,4,224,32,20,152,32)
360 CALL LOAD(9966,32,183,32,166,19,5,152,32,32,183,32,168,19,1,4,91,7,96,32,162,19,1)
370 CALL LOAD(9988,5,129,200,1,32,88,200,32,46,220,32,94,6,160,37,138,16,223,0,0,200,11)
380 CALL LOAD(10010,39,22,2,224,38,246,2,0,0,2,1,0,2,2,2,37,246,4,32,32,20)
390 CALL LOAD(10032,2,1,0,1,5,128,2,2,36,246,2,3,0,255,6,195,216,3,36,246,4,32)
400 CALL LOAD(10054,32,20,193,32,36,246,6,196,2,68,0,255,200,4,36,244,2,132,0,0,22,2)
410 CALL LOAD(10076,4,192,16,12,2,1,255,255,5,129,208,161,36,246,208,225,37,246,144,194,22,223)
420 CALL LOAD(10098,136,1,36,244,22,246,200,0,131,74,4,32,32,24,0,32,2,0,0,2,1)
430 CALL LOAD(10120,0,3,4,32,32,8,4,192,216,0,131,124,2,224,131,224,194,224,39,22,4,91)
440 CALL LOAD(10142,7,79)
450 END

```

→ MORE

PARCO LATEST...
 If you haven't seen a copy of Personal Computer World recently, you may have missed the latest telephone number for PARCO. Here it is:
 Telephone 01888625435665

Final advice on peripherals: If you are going to keep your TI and are prepared to spend the necessary on a PEB, buy a disk controller now before they disappear! It may soon be too late... and with a disk drive you have access to a huge range of very cheap programs which really open up your TI to its fullest capacity...

Now the OBJECT code:

```
002AA      A0000B0000BFF00A0003A0102BFF00A0103A0202A0222B00007F33BF      0001
A0224BC80BC0222B02E0C0202B0200B0000B0201B0002B0202C0102B04207F345F      0002
A023AB2014B0201B0001B0580B0202C0002B0203B00FFB06C3BDB03C000027F30EF      0003
A0250B0420B2014BC120C0002B06C4B0244B00FFBCB04C0000B0284B00007F304F      0004
A0266B1602B04C0B100CB0201BFFFFB0581BDOA1C0002BD0E1C0102B90C27F292F      0005
A027CB16DFB8B01C0000B16F6BC800B834AB0420B201BB0020B0200B00007F2E0F      0006
A0292B0201B0003B0420B2008B04C0BD800B837CB02E0B83E0BC2E0C02227F2CFF      0007
A02AB8045B7FD90F      0008
50224SEARCH7FD16F      0009
:          99/4 AS      0010
```

Remember to save this using PRINT FILE with the F option, eg:
PF
F DSK1.SEARCH/0

And a demo:







```
100 DIM STRING$(20)
110 CALL PEEK(12000,A):: IF A=1 THEN 140
120 CALL INIT :: CALL LOAD("DSK2.SEARCH")
130 CALL LOAD(12000,1)
140 FOR A=1 TO 14 :: READ STRING$(A):: NEXT A
150 CALL LINK("SEARCH",STRING$(A),"SPIDER",A):: PRINT A
160 DATA DOG,CAT,HORSE,LION,ELEPHANT,TIGER,MONGOOSE,DEER,OTTER,SPIDER,ANTEATER,E
AGLE,SHARK,GORILLA
170 END
```

Want a printer? If you don't fancy any of the EPSON range, Richard Speed tells me that his AMSTRAD DMP2000 (connected to a Myarc RS232) works perfectly.

And on to the next piece of code- in normal 32 column text mode you can have sprites whizzing around, and call use CALL COINC to see if they are near a specific screen location. However, what if the sprite is to whiz around within the boundaries of a maze? Or has to detect a hazardous graphic character? This little routine will tell you if your sprite hits an "on" pixel on the screen. The source code is for TI ExBas only - but as above you only need to change a few EQUATES to REFS to use it with Myarc XB.

```
*****
* PIXEL COINC UTILITY
*****
DEF COINC
*
VSBW EQU >2020 * \
VMBW EQU >2024 * \
VSBR EQU >2028 * Change these to REFS
VMBR EQU >202C * for Myarc ExBas
NUMREF EQU >200C *
NUMASG EQU >2008 * /
XMLLNK EQU >2018 */
FAC EQU >834A
GPLWS EQU >83E0
MYREG BSS 32
```

* MORE MORE ---->

	Protect Proteger Protéger Schützen 保護		Never Nunca Jamais Nie 絶対禁止
	No No Non Falsch 注意		10°C → 52°C 50°F → 125°F
	Insert Carefully Insérer Insérer avec soin Sorgfältig Einsetzen 挿入注意		Never Nunca Jamais Nie 絶対禁止

Syd advised me I could EAT his giant adventure disk if I wanted to- but Sydney, the disk jacket bears a very specific warning NOT to munch a floppy....^^

*
RTRN DATA 0
COINC

```
MOV R11,@RTRN
LWPI MYREG
BL @GETNUM
DEC @XPOS
DEC @YPOS
BL @CDINK
BL @SETNUM
LWPI GPLWS
MOV @RTRN,R11
RT
```

```
*
ONE DATA >4001,>0000,>0000,>0000
XPOS DATA >7F
YPOS DATA >60
XDIV DATA 0
YDIV DATA 0
XREM DATA 0
YREM DATA 0
SPOS DATA 0
PPOS DATA 0
DECB DATA 8
DECS2 DATA 32
```

```
*
COINC MOV @XPOS,R2
CLR R1
DIV @DECB,R1
MOV R1,@XDIV
MOV R2,@XREM
```

```
*
MOV @YPOS,R2
CLR R1
DIV @DECB,R1
MOV R1,@YDIV
MOV R2,@YREM
```

```
*
MOV @YDIV,R1
MPY @DECS2,R1
A @XDIV,R2
MOV R2,@SPOS
```

```
*
MOV @SPOS,R0
BLWP @VSBR
ANDI R1,>FF00
SWPB R1
MPY @DECB,R1
A @YREM,R2
MOV R2,@PPOS
```

```
*
MOV @PPOS,R0
BLWP @VSBR
ANDI R1,>FF00
SWPB R1
LI R2,>0080
CLR R3
SRCLP C R3,@XREM
JEQ FINSRC
INC R3
SRC R2,1
JMP SRCLP
```

* MORE MORE ---->

- Review Time!

RAPID COPY by Barry Boone, (of Oklahoma, no relation to Bob Boone of Ottawa!) released by TEXAMENTS for US\$15, runs from XB, Ed or TIM, needs 32k and disk system.

One of the fastest disk copying programs around- faster than the track copy using Miller's Diagnostics package. How about 3 seconds to copy a SSSD disk using two drives? Not bad. Now use quite heavily for the disk library copying chores.

Add extra to cover overseas mail- say US\$3 or so. TEXAMENTS, 53 Center Street, PATCHOGUE, New York, USA, 11772.

This supplier is very reliable and FAST.

(I have a report that this program will not run with the Myarc MFES system, although it will run with the Myarc disk controller card for the TI PEB)

JOY PAINT 99 from Great Lakes Software, 804 E Grand River Ave Howell, MI, USA, 48843. US\$40.

Yes, yet another graphics program, and at a premium price too. How does it differ from GRAPHX or TIARTIST?

TI ARTIST: includes conversion program for GRAPHX.

Supported by MAX-RLE

Considerable number of FONTS and INSTANCES available.

Ready access to fonts and instances in XB.

Can be driven with Super Sketch tablet.

Can be driven by Tandy mouse.

VERY fast and easy to use FONTS.

Fast and easy to use INSTANCES and SLIDES.

GRAPHX: Supported by MAX-RLE, Fair number of fonts and clip-art.

Includes ELLIPSE capability.

Very easy to use with pull down menus.

Joy Paint fails on these but has:

Aerosol (air-brush) effect

Several "fill" patterns not just solid colour

Disk directory function. Ability to UNDO last operation.(neat).

Claims to use joystick input only, but you DO need to type in file names!

Dislikes about JoyPaint: The major joystick-selected menu is on extreme screen left. On my tv set, I cannot see the icons. NOT the end of the world, but a rotten location choice.

Fonts are CUT AND PASTED from full-alphabet screens, even slower and far more tiresome than GRAPHX.

No clip-art or instances - again you must use CUT AND PASTE.

First load the full page containing what you want, cut out what you want, load your screen, and paste the cut out into it. Again and Again and Again for text...

At present no conversions, but I understand an extra disk is to be sold with conversion programs.

CIRCLES are located by defining- would you believe- a CORNER! of the square which would surround the circle. This I detest!

My overall impression is that this package may be better for pure graphics, thanks to the air-brush and fill effects, but if you are going to use any text, forget it! The price differential inclines me rather heavily towards continuing to recommend TI ARTIST.

HOWEVER: I do know that some US owners prefer JoyPaint to TI Artist. Horses for Courses perhaps.

750

```

*
FINSRC SZC R1,R2
        CI R2,0
        JNE CONC
        LI R2,1
        RT
CONC LI R2,0
        RT
*
GETNUM CLR R0
        LI R1,1
        BLWP @NUMREF
        BLWP @XMLLNK
        DATA >12BB
        MOV @FAC,@XPOS
        LI R1,2
        BLWP @NUMREF
        BLWP @XMLLNK
        DATA >12BB
        MOV @FAC,@YPOS
        RT
SETNUM CI R2,1
        JEQ RTRN1
        CLR @FAC
        CLR R0
        LI R1,3
        BLWP @NUMASG
        RT
RTRN1 CLR R1
ONELP MOVB @ONE(R1),@FAC(R1)
        INC R1
        CI R1,8
        JNE ONELP
        LI R1,3
        CLR R0
        BLWP @NUMASG
        RT
        END

```

Thanks are due to Peter Barker for sending two specialised data base programs- a label printer and a ham radio log. Part of the disk library if you are interested! It is nice to see one UK TI programmer writing material for our computer!

Thanks to Richard Speed for sending a program for private review, happy to help out where I can.

Thanks to Sydney Michel for sending his huge Adventure program, "The Big Texas Spy Adventure" with - I am told - 158 locations! Requires XB, 32k and disk system, and is in the Disk Library.

(As there was room on the disk I also included the ADULT adventure by Stephen Peacock, which requires the ADVENTURE module... so far I have not found anything more risque than the need to empty an overfull bladder, but then I am an appalling adventurer...).

-Are there any more appalling adventurers? I now have directions for completing ZORK 1, not hints, just a list of what to type in, so you can enjoy the textual content of the adventure, and having made it to the end you can go back and do all the silly things which result in such interesting output...

-Thanks to Edward Shaw for sending a utility program for private review.

And THANKS to John Stocks for another excellent demonstration of a hi res plot using mini memory- the speed up by use a look up table for trig, writing your own square root routine, and using integers, is enormous. Contact John if you have a mini mem and would like a look- send a cassette and return postage, address is last two issues.

And thanks to J S Dunning for a utility program which shows you a two character high number or upper case letter together with its two hex codes - if there is room I will try to get it into this issue, else it is on disk UTILITY-12 from me at usual prices.

... yes this has been a good quarter for post, and more UK written programs than at any other time!

DEMONSTRATION PROGRAM:

```

100 CALL CHAR(42,"8000000000000000")
110 CALL INIT :: CALL LOAD("DSK2.GPIXEL")
111 PRINT : : :
115 PRINT "USE JOYSTICK TO MOVE BLACK DOT. WHEN IT IS OVER A PIXEL THAT IS ON 1
WILL APPEAR AT THE BOTTOM OF THE SCREEN IF NOT A ZERO WILL BE SHOWN "
120 FOR A=1 TO 12 :: CALL COLOR(A,7,1):: NEXT A :: CALL SPRITE(#1,42,2,100,128):
: Y=100 :: X=128
130 CALL JOYST(1,DX,DY)
140 X=X+DX/4 :: Y=Y-DY/4
150 CALL LOCATE(#1,Y,X)
160 CALL LINK("COINC",X,Y,A):: DISPLAY AT(24,3):A
170 GOTO 130
180 END

```



```

1 REM NEW SPRITE COINC DETECTION UTILITY
2 REM RETURNS 1 WHEN SPRITE COINCIDES WITH AN "ON"
3 REM IN NORMAL 32 COLUMN MODE OF COURSE!
BY GRAHAM MARSHALL
PIXEL ON THE SCREEN
REQUIRES XB + 32K
100 CALL INIT
110 CALL LOAD(16376,67,79,73,78,67,32,37,22)
120 CALL LOAD(8194,38,56,63,248)
130 CALL LOAD(9460,0,0,0,0,0,0,0,0,0,203,20,203,53,203,78,203,231,204,71)
140 CALL LOAD(9482,204,150,204,228,205,29,205,75,205,96,0,0,200,11,37,20,2,224,3,6,244,6,160)
150 CALL LOAD(9504,37,218,6,32,37,68,6,32,37,70,6,160,37,88,6,160,38,6,2,224,131,224)
160 CALL LOAD(9526,194,224,37,20,4,91,64,1,0,0,0,0,0,0,127,0,96,0,0,0)
170 CALL LOAD(9548,0,0,0,0,0,0,0,0,0,0,0,32,192,160,37,68,4,193,60,96,37,84)
180 CALL LOAD(9570,200,1,37,72,200,2,37,76,192,160,37,70,4,193,60,96,37,84,200,1,37,74)
190 CALL LOAD(9592,200,2,37,78,192,96,37,74,56,96,37,86,160,160,37,72,200,2,37,80,192,32)
200 CALL LOAD(9614,37,80,4,32,32,40,2,65,255,0,6,193,56,96,37,84,160,160,37,78,200,2)
210 CALL LOAD(9636,37,82,192,32,37,82,4,32,32,40,2,65,255,0,6,193,2,2,0,128,4,195)
220 CALL LOAD(9658,136,3,37,76,19,3,5,131,11,18,16,250,64,129,2,130,0,0,22,3,2,2)
230 CALL LOAD(9680,0,1,4,91,2,2,0,0,4,91,4,192,2,1,0,1,4,32,32,12,4,32)
240 CALL LOAD(9702,32,24,18,184,200,32,131,74,37,68,2,1,0,2,4,32,32,12,4,32,32,2,4)
250 CALL LOAD(9724,18,184,200,32,131,74,37,70,4,91,2,130,0,1,19,8,4,224,131,74,4,192)
260 CALL LOAD(9746,2,1,0,3,4,32,32,8,4,91,4,193,216,97,37,60,131,74,5,129,2,129)
270 CALL LOAD(9768,0,8,22,249,2,1,0,3,4,192,4,32,32,8,4,91,39,52)
280 END

```

```

00144 A0000A0020B0000BC80BC0020B02E0C0000B06A0C00E6B06207F339F 0001
A003C0050B0620C0052B06A0C0064B06A0C0112B02E0BB3E0BC2E0C00207F2F7F 0002
A0046B045BB4001B0000B0000B0000B007FB0060B0000B0000B0000B00007F365F 0003
A005CB0000B0000B0000B00020BC0A0C0050B04C1B3C60C0060BC801C00547F305F 0004
A0072BC802C0058B0A0C0052B04C1B3C60C0060BC801C0056BC802C005A7F2BBF 0005
A008BB0C060C0056B3860C0062BA0A0C0054BC802C005CBC020C005CB04207F2CDF 0006
A009EB2028B0241BFF00B06C1B3860C0060BA0A0C005ABC802C005EBC0207F29DF 0007
A00B4C005EB0420B2028B0241BFF00B06C1B0202B00B0B04C3B8803C00587F2DBF 0008
A00CAB1303B0583B0B12B10FAB4081B0282B0000B1603B0202B0001B045B7F2F7F 0009
A00E0B0202B0000B045BB04C0B0201B0001B0420B200CB0420B201BB12BB7F31AF 0010
A00F6BC820B834AC0050B0201B0002B0420B200CB0420B201BB12BB8C8207F2EBF 0011
A010CB834AC0052B045BB0282B0001B1308B04E0B834AB05B1B0281B0008B16F97F2D3F 0012
A0122B0420B2008B045BB04C1BD861C0048B834AB05B1B0281B0008B16F97F2D3F 0013
A0138B0201B0003B04C0B0420B2008B045B7F868F 0014
50022C0INC 7FD44F 0015
: 99/4 AS 0016

```

END □

DISK LIBRARY: All disks SSSD. One pound per disk copying fee plus one pound post and packing per order. Send disks in good packing! Floppies count as two disks!

RAG MAC macroassembler is revised to 1987 and now uses THREE disks.
C99 REL 3 (Vn 2.1) plus support files occupies FOUR disks.
DM1000 Vn 3.5 is on one disk, SOURCE CODE on two disks.
FUNLWRITER revised to Feb 87 at press time, on TWO DISKS.
RLE now up to 19 disks! Loaders on RLE/1 and RLE/13.

TI990POLY is one one disk.
A MOUNTAIN OF DISKS AVAILABLE! SEND BLANK DISK AND RETURN POSTAGE FOR LISTING!
10 ALSTONE ROAD STOCKPORT CHESHIRE SK4 5AH


```

Z034 LI R1,>7E00
MOV R4,R0
BLWP @VSBW
Z034C INC R4
MOV *R0,R1
CI R1,>1F00
LI R0,163
LI R1,2030
LI R2,4
BLWP @VMBW
MOV @Z029,R13
MOV R13,R0
BL @Z046
LI R0,172
LI R1,2030
LI R2,4
BLWP @VMBW
MOV R14,R0
BL @Z046
LI R0,181
LI R1,2030
LI R2,4
BLWP @VMBW
MOV R15,R0
BL @Z046
LI R0,195
LI R1,2030
LI R2,4
BLWP @VMBW
LI R4,8*32
LI R5,2
MOV @Z020,R6
CI R6,0
JEQ Z035
CI R6,32
JLE Z034
LI R6,32
BLWP @VMBW
AI R4,2
MOV R4,R0
LI R1,>3D00
BLWP @VSBW
INC R4
CLR R1
MOV @Z021(R5),R0
BLWP @VSBW
CI R1,>1F00
JLE Z036A
CI R1,>7F00
JHE Z036A
MOV R4,R0
BLWP @VSBW
JMP Z036B
Z036A LI R1,>7E00
MOV R4,R0
BLWP @VSBW
Z036B INCT R5
AI R4,5
DEC R6
JNE Z036C
* ----> more!

```

```

Z037 BL @Z044
CI R1,>5000
JEQ Z039
CI R1,>0D00
JEQ Z038
JMP Z037
Z038 CLR R0
LI R1,2026
LI R2,768
BLWP @VMBW
RTWP
*****
* OPEN PRINTER DEVICE. IF NO PRINTER *
* PUT "*" IN FRONT OF "BLWP @DSRLNK" *
* AND "DATA B" *
*****
Z039 LI R0,2022
LI R1,2025
LI R2,28
BLWP @VMBW
LI R0,2022+9
MOV R0,@B356
*****
BLWP @DSRLNK *****
*****
DATA B
LI R1,>0300
LI R0,2022
BLWP @VSBW
LI R0,2028
LI R1,>2000
LI R2,80
Z040 MOV R1,*R0+
DEC R2
JNE Z040
LI R3,24
CLR R4
MOV R4,R0
LI R1,2028
LI R2,32
BLWP @VMBW
LI R0,2023
LI R1,2028
LI R2,80
BLWP @VMBW
LI R0,2022+9
MOV R0,@B356
BLWP @DSRLNK
DATA B
AI R4,32
DEC R3
JNE Z041
LI R0,2022+9
MOV R0,@B356
LI R0,2022
LI R1,>0100
BLWP @VSBW
BLWP @DSRLNK
DATA B
B @Z037
*****

```

```

*****
Z042 LI R0,767
LI R1,>2000
Z043 BLWP @VSBW
DEC R0
JNE Z043
RT
*****
Z044 BLWP @KSCAN
CLR R1
MOV @B375,R1
RT
*****
Z045 TEXT @123456789ABCDEF
Z046 LI R7,2030
LI R8,4
Z047 MOV R0,R9
ANDI R9,>F000
SRL R9,12
MOV @Z045(R9),*R7+
SLA R0,4
DEC R8
JNE Z047
RT
*****

```

END

As one of the original UK owners of a TI99/4, I shall continue to use my TI99/4A for QUITE some time. Other computers may be more powerful, but inevitably need more programming time to make them perform their tricks. And I like to program, not just use purchased programs! My programming has by no means exhausted the capabilities of my TI yet. Thank you to everyone who has helped ME to discover some of the secrets of the TI - irritating questions have often driven me to explore some dim dark alley to find surprising fruits! Hope to see you in Derby!

"Where is scratch pad ram?"- Scratch pad ram is the ONLY area of ram addressed with a 16 bit data line. It is FAST. It can be found from >8300 to >83FE, and parts of it are used by the operating system. Different bits are used depending on how you are using the console! In case you thought EASY BUG was part of the Mini Memory module- yes, it is in the module, but it does not RUN from there- for greater speed it is transferred to the scratch pad area, where it resides from >8370 to >83FF >8300 TO >8348 IS USED BY BASIC (XB) and so seems to be a reasonable place to put a pure machine code utility. >834A TO >83FE seem to be fully utilised for most environments.

"I know how IF (A=2)+ (B=4) THEN works, but what does: IF (A=2)+ (B=4)=-2 THEN... mean..." This one goes back to the pioneering work of Pete Brooks in the early days... IF... THEN... can be considered to take the form: IF NOT ZERO THEN...ELSE IF ZERO THEN ... So: IF A THEN 100 will go to 100 if A has any value other than zero. The computer treats an equals sign in a LOGICAL manner, treating the result as zero or minus one. So, 2=2 is TRUE, and takes a value MINUS ONE 4=2 is NOT TRUE and takes a value of ZERO Now look at the question again. If A=2 the value of this equate is: -1 If B=4 the value of this equate is: -1 And the sum of these is: -1+ -1=-2

What if one of the statements is untrue? Then one equates will be minus one, and the other will be zero, which adds up: 0+ -1=-1.

Therefore, the formula: (A=2)+ (B=4) will have a non-zero value if EITHER of the equates is true, OR if both of them are true. The plus sign can be thought of as OR: IF A=2 OR B=4 THEN..., which will carry out the transfer if either or both conditions are met. more....---->

If however we wish to transfer control only if both conditions are met, we need an equivalent to AND: IF A=2 AND B=4 THEN... Try: IF (A=2)*(B=4) THEN... If both are true: -1 * -1 =+1 = non zero If only one is true: -1 * 0 = 0. Problem solved.

So what of that strange form we started with? IF (A=2)+ (B=4)=-2 THEN... IF BOTH are TRUE the answer is -2 as we have seen! If ONE is true, the answer is -1 If NONE are true the answer is ZERO.. Therefore this oddball form is another way of saying AND! I cannot think of any reason for preferring this form to the use of the multiply (asterisk) sign.

"Where is the Program Counter and is it accessible"... this is a machine code question! The 9900 CPU can address 16 software registers, which you can place anywhere in CPU RAM. It also has three HARDWARE registers, The Program Counter, the Workspace Pointer, and the Status registers. Details can be found in the Editor Assembler manual, pages 39 onwards. According to the EXPLORER documentation, R14, >83DC, contains the "Return PC for context switch (RTWP)"

There are several utilities on sale which allow you ready access to the registers, and of course the Fairware program SUPER BUG by Edgar Dohmann. I have version 1, and version 2 is available direct from Ed in Texas. The command to use is R. SUPER BUG will work with Mini Memory and 32k ram, and requires a disk system.

***** "What the @@ is the AID key intended for.... for all those modules which intend you to get instructions by pressing AID! (FCTN 7)! It has its own use in Mini Memory Easy Bug (see p 64 of the MiniMem manual!). AND it can be used in a BASIC program by using CALL KEY, as AID returns a key value of 1. In Myarc XB, the AID key can also be used for ACCEPT and INPUT, instead of ENTER, and places a value of 1 in the variable TERMCHAR. *****



FAST COUNT

This little piece of machine code SOURCE CODE is to illustrate how to use the SCRATCH PAD RAM for a machine code program, and quite nicely shows just how fast it can be. Just assemble this and show it to your friends with C64s or Spectri....

```
* FAST COUNT
DEF START
REF VMBW,VMBR
JMP START * HUH?
CT
BSS 6
START LWPI >B3F0
LI 1,LD * move from LD
LI 2,>B300 * move to >B300
LI 3,234 * move 234 bytes
TP
MOV *1+,*2+
DEC 3
JNE TP
B @>B300 * now the program
* is at >B300
* let's go there!
*
LD LI 4,>3A00 * 10 =chr$(5B)
LI 5,>3030 "00" =ascii for zero
LI 6,>3100 "10"
*
MOV 5,@CT
MOV 5,@CT+2 * MAKES CT
MOV 5,@CT+4 * = "000000"
JMP AD
*
RS SWPB 0
MOVB 0,@>BC02
SWPB 0
MOVB 0,@>BC02
NOP
RL MOVB *1+,@>BC00
DEC 2
JNE RL
*
AD LI 0,>4000+370
INC @CT+5 *Increase last digit
CB @CT+5,4 *Is last digit 9+1?
JEQ ND *if so go to ND
*
* -----> more ---->
```

```
* more ---->
* PRINT WITHOUT CARRY
*
SWPB 0
MOVB 0,@>BC02
SWPB 0
MOVB 0,@>BC02
NOP
MOVB @CT+5,@>BC00
JMP AD
*
* DIGIT CARRY
*
ND MOVB 5,@CT+5
AB 7,@CT+4
DEC 0
LI 1,CT+4
LI 2,2
CB @CT+4,4
JNE RS
MOVB 5,@CT+4
AB 7,@CT+3
DEC 0
DEC 1
INC 2
CB @CT+3,4
JNE RS
MOVB 5,@CT+3
AB 7,@CT+2
DEC 0
DEC 1
INC 2
CB @CT+2,4
JNE RS
MOVB 5,@CT+2
AB 7,@CT+1
DEC 0
DEC 1
INC 2
CB @CT+1,4
JNE RS
MOVB 5,@CT+1
AB 7,@CT
DEC 0
DEC 1
INC 2
CB @CT,4
JNE RS
MOVB 5,@CT
JMP RS
END
```

```
* PHEW
* THIS ONE RUNS IN THE
* SCRATCH PAD RAM WHICH IS
* 16 BIT ADDRESSED
```



THIS CODE FIRST APPEARED IN THE MAGAZINE OF THE INTERNATIONAL USER GROUP IN BETHANY.

Code written by Bill Granos.

REVIEW!

THE PRINTERS APPRENTICE by Mike McCann, from McCann Software, P O Box 34160, Omaha, NE, USA, 68134. Price US\$23.

What an interesting program this is! I am referring here to Version 2 by the way- version 1 was downright difficult to use!

TPA as it is known is another utility written in FORTH, and none the worse for that.

What does it do?

1. You can "set up" a page to print, with different typefaces, and with pictures inserted, and print in one pass- slowly! Text is from ordinary DFBO files, (TI Writer CAN output DFBO!), while pictures originate from TI ARTIST. OR there are text and picture editors included.

2. Your DFBO text can be printed on its own, using the special fonts supplied or which you create. If your printer can do it, fonts can be printed in 1x, 2x, 4x or high speed 2x density. Optional right justification, not by inserting whole spaces but by adding single pixel spaces- much neater! Option hyphenator feature. Fonts may be "over/under strike" which has "nlq" capability, or at least very close to it.

3. You control the default spacing between letters and the width of the space from the program, unlike TI-Artist fonts where letter spacing is fixed in the font file.

4. Zzzzzz. Using a 160cps printer it takes AGES! to print a page. Phew. Nice effect though.

5. You could do something similar with TI ARTIST and scissors and paste, but it might take longer to get the same text effect!

6. There is a utility provided which allows you to transfer TI Artist fonts up to 24 pixels high into a format that TPA can use- to produce micro-justified text from your TIW files!

7. OVERALL a useful program if you have a SERIOUS need to mix text and pics OR to neatly print in different fonts. I insert SERIOUSLY due to the graphics design work which you must still do, and the printing time required. Powerful. Dont forget to add extra overseas postage- say \$5 for this one.

The documentation for this program is pretty awful- I was able to use the FORMATTER option straight away using the fonts supplied, but it took me two days to sort out how to design my own fonts!

My ancient Epson does not have NLQ print available, but with this program, I can add it myself, AND design my own font entirely. Everyone else uses a slashed zero, but I prefer to use the IBM standard, which just has a dot in the middle!

(There's a story to this one- I paid for my copy with a replacement Kenwood beater! Supplied at HALF the price Harrods quoted and five times faster! This form of trade is called BARTER!!!)

PARCO LATEST (Clarification):

The advert in PCM magazine, headed in large letters "PARCO IS YOUR BEST CHOICE" is placed by a company quite some distance away- PEI CHOW INDUSTRY CO LTD IN TAIWAN.

If you do not have an ARGOS store near you (they seem to have several dozen around the country...) you are missing an EXCELLENT price on 5 1/4" disks- a box of ten disks (who needs bulk order!!) -DSDD quality, sold retail on 16 day approval! - vat inclusive, 70p each. Checking out PCM, you are struggling to get that price on BULK disks.

One of our members sent me some for copying (free testing!) and they ran VERY smoothly. So often, the El Cheapo disks I am sometimes sent make a noise which can best be described as worrying...

So, if you want disks, find an Argos store!

....think twice before sending large sums of money to companies you have not heard of for disks- they just might not be there when you ask where your disks are... with WH Saith now selling computer cassettes for 75p, disks are now a cheaper medium than cassette- a long cry from when I bought my first disks!

Several members have told me that CALL LOAD(-31962,255) does work on their consoles- well, it doesn't work on mine! so don't assume it will work on every console!

Remember I mentioned that TI Writer disk files could have various sizes before you got BUFFER FULL? Guess what- a little bird tells me that when text is loaded into the buffer, it is run length encoded... now there's a technique I've heard of before...

To put it another way, a whole lot of blank spaces, or dashes, or anything else, dont appear as a whole lot of.... they appear as a code to indicate "X number of things here please" which takes up a bit less room...

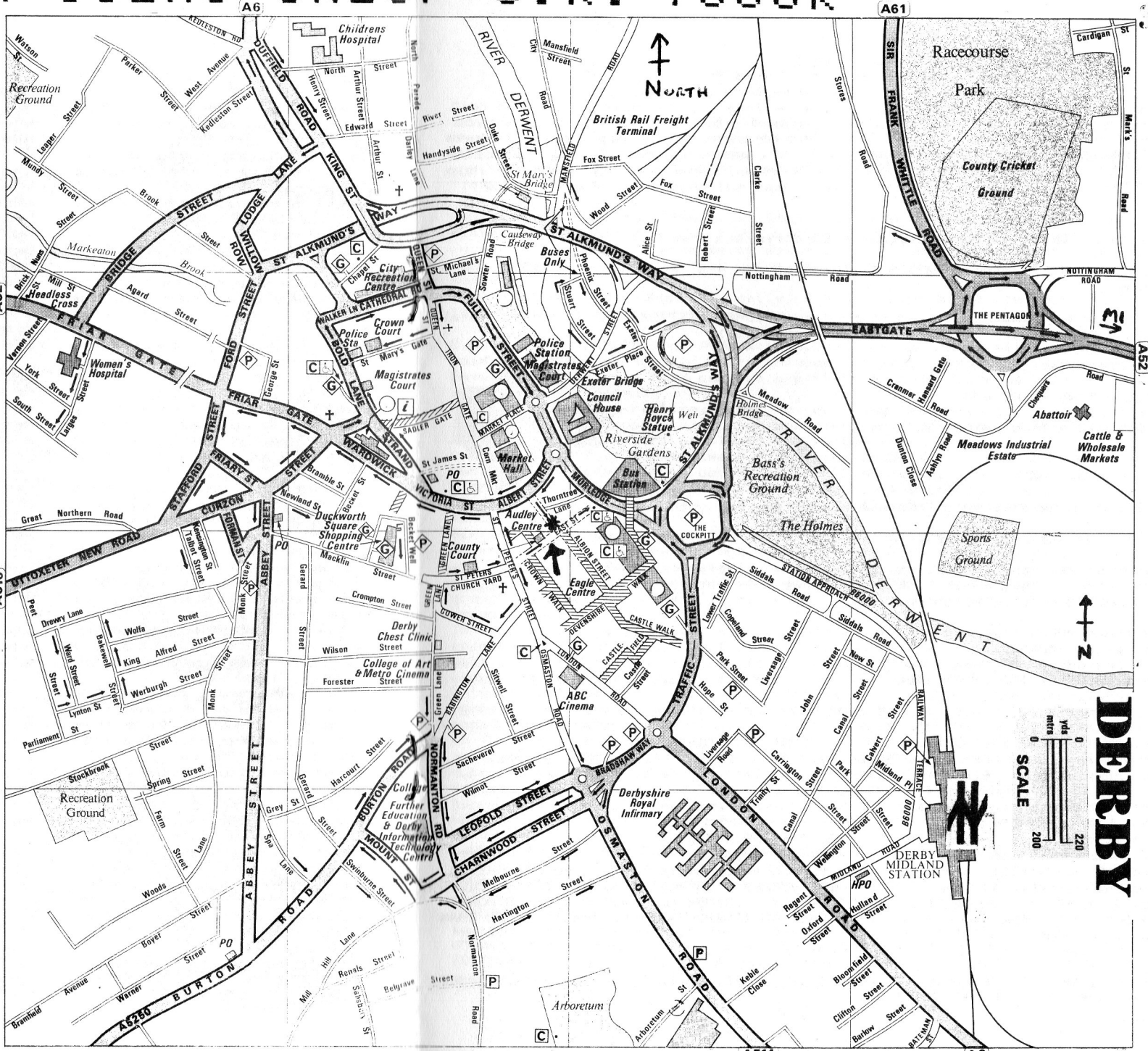
As I have no idea what is going to follow this issue of TI*MES at this moment, I make no promises about what you will find in this issue of Rambles- I shall try to find bits and pieces in general request and get as many in as possible, but it is not the time to start a series! Your letters are always welcome, and I hope you can appreciate the financial need which makes me ask you to send a stamped addressed envelope if you would like a reply.

I have quite a collection of documents here to help me answer questions, but please dont ask me anything too technical, as I am no expert as far as machine code or hardware are concerned!

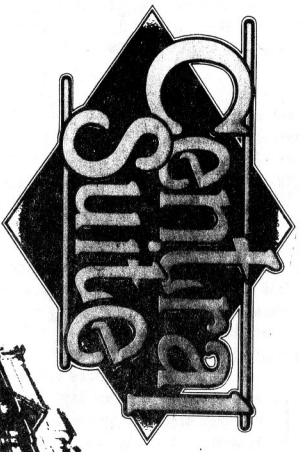
And the disk library will remain available as long as possible- AT LEAST to December 1987 and hopefully beyond, depending upon demand. A library listing is available ON DISK ONLY - just send a blank disk and return postage for a copy.

TI 99/4A USERS GROUP U.K. TUGUK

TI 99/4A USERS GROUP U.K.

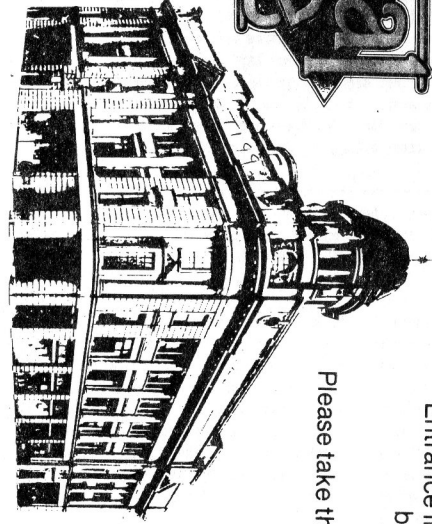


GRAND T I USERS FAIRE DERBY



* Exchange Street Derby

Central Hall

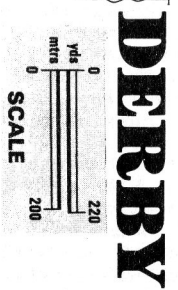


Entrance in Exchange Street,
between Homestyle
and Albion Travel.
Please take the lift or stairs to the
second floor

- Official Car Park Free
- Multi-storey Car Park
- Parking Available On Payment
- One-way Street
- Convenience



Crown Copyright
Reserved



DERBY

TI USERS FAIRE
DERBY 16TH MAY 1987

Doors open 10am to 6pm free to members and family.

This is another great event not to be missed. Not only will you have your new group, TI99/4A USERS GROUP U.K. but an excellent chance for you to have a clear out. Bring that TI jumble sell or swap at the Faire. Bargains galore, and all funds for the new group. Please come and support. Help needed.

Replies from questionnaires.

TOTAL of 146 replies from 358 sent out give a 48% reponse. THANKYOU very much.

As regards remaining a member of a NATIONAL BRITISH TI99/4a GROUP, Total 100% SUPPORT. (Percentages of forms returned.)

Attendance at the FAIRE-

WILL ATTEND FAIRE 47%

UNABLE TO ATTEND FAIRE 28%

DON'T KNOW 25%

Willing to help serve on the new committee, not yet nominated.

TOTAL reponse to join 17%.

Names and telephone numbers of members willing to help form the new committee. They will ask you to cast your vote on the day for or against nomination of the following.

Chairman VACANT.

Secretary VACANT.

Rambles Editor STEPHEN SHAW. (STOCKPORT).

Editors CHRISTINA MEHEW (TEL 0469 8404) and ALAN BAILEY (TEL 01 508 1053)

International Editor VACANT.

Treasurers PETER CROSS (TEL 0322 522809) and PETER WALKER (TEL 0707 873778).

Technical MIKE GODDARD (TEL 0678 520737) COLIN HINSON (0767 40503) TONY

BOWDEN (0279 730594) Reserves SIMON DORRIDDOTT (0234 58981).

Programming GEOFFREY COAN (TEL 061 427 4440) PETER WALKER (0707 873778)

BBS/Prestel NEVILLE BOSWORTH (TEL 073 441714)

R.T.T.Y. PETER BARKER (TEL 0794 8286).

Membership secretary VACANT

Disk Library VACANT

Cassette Library MAURACE RYMILL (TEL 021 458 4970).

Faire organisation GERALD COLLINS (0202 433704).

General committee members to fill above vacancies subject to nomination.

RICHARD SIERAKOWSKI (0672 54975) LEO HUGHES (PAGHAM 266188) BRYAN CLOUD

(0473 464996) JOHN BUTCHER (0293 884204) L.KINGSTON (0483 464996) W.WORGER

(ALDRSHOT) T.ANDERSON (GLASGOW) E.G.SMITH (RUGBY 74111 EXT 248) EDWARD SHAW (0538 360382)

ANYONE WANT A LIFT FROM SCOTLAND TO THE DERBY FAIRE?? Please contact JAMES PARKINSON (GLASGOW 041 775 1750)

Your Letters.

Alan Rutherford WILMSLOW, Cheshire writes... When I answer the small adverts I discover everything has gone... is TI*MES posted separately? Is there anything you can do about this?

ED Yes Alan, most small adverts are placed two or three months in advance of publication. Chances are they would be sold. As regards distribution of TI*MES, they are all posted the same time.

Robert Priest who owns the Audio Shop near the famous Lanes in Brighton stated... "I had a colour T.V. hooked up to my TI99/4a to display Adverts in my shop window... because it was a T.V. I was informed that I needed a T.V. Licence. Now I am using a mono monitor. I hope to use a colour monitor soon."

ED I have seen your shop, it really is a wonderful sight to see that the TI99/4a runs 7 days a week advertising!.

E.R.Irving BRADFORD on AVON writes.... You have done sterling service in guiding fledgling computer users through the early stages of enthusiasm and have consolidated a hard core of expertise on this super machine (TI99/4a), which I fully commend.

As I think you now realise, due to dropping memberships and increasing diversity, the groups are losing momentum, which I think is mainly due to a loss of direction attributable to the following factors.

A computer is what the software / firmware makes it, so all non disc drive owners have been left behind and so have probably "defected" or given up TI*MES on the basis that involved articles on the intricacies of the likes of TI Writer and FORTH are of no interest as they are so application specific.

Similarly, all new modules / cards away from the TI standard are of limited interest. Whatever the current manufacturers hype says, all other computers are different and not compatible because they are different to benefit from being different. Therefore if I wanted a MYARC I would join a MYARC users group and would not expect it to be using up valuable pages in my TI Users group magazine.

To regain popularity I think there must be two main objectives which are to open-up the capabilities of the BASIC functions and to clearly categorize sections as "4 front" splendidly does, so that contributors define their areas and so may be encouraged to achieve breaking boundries but also gives valuable information on areas of interest.

Plenty more ideas and some knowledge, if you want any from a solid supporter.

ED Thankyou for putting pen to paper, I do not doubt a single word, however as Stephen Shaw will confirm, there are very few people who communicate or submit ideas articles etc of Basic interest. I think the group would benefit from the ideas you have and if you are attending the TI FAIRE in Derby, put your views across, better still join the Committee. By the way REGENA who writes for MICROpendium works without expansion.

DAVE G.HEWITT (DIY Fame) HAS MOVED TO 53 KENNET CLOSE, BERINSFIELD, OXFORD. Dave writes... Apologies to anybody who has written and not had a reply If your letter was returned perhaps you could write to me again at the new address.....

ERRORS in TI*MES issue 15. Page 21 PROGRAM Line 110 the first two numbers in the CHR\$ statements should be 117 not 125 and 50 not 0 as shown. The effect of this bug is to produce a line 32000 not line 30002 as required.

ED Thanks for your letter, we have quite a fan mail for your DIY RS232 Monitors expansion etc, hope for another article for next issue TI*MES.

TI-LINES

As some subscribers will know, I have been running what used to be OXON TI USERS and is now THE INTERNATIONAL TI USER GROUP since April 1984. I began the group as a service for Oxfordshire owners, and it has gone on expanding ever since.

I attempted to turn the group into a "proper" club after the first two months, but, because of a total lack of response, I had no choice but to turn it into what is known as a "commercial" User group. In my case, this does not mean that the group takes a business-like view of its subscribers (which has a rather nasty feel to it, as if the subscriber was of secondary interest), but that it is declared to the Inland Revenue as an extension of my professional activities, which carries certain tax advantages. It also means that I am not carrying out a hobby, and that means that the banks are more helpful than they might otherwise have been!

In addition, other companies with which I deal regard ITUG not as a collection of frivolous amateurs, but as a group of serious enthusiasts which are to be taken seriously. All too often the TI owner is regarded as insignificant - witness the lack of published material in the popular computing press.

The "serious enthusiasts" label doesn't mean that we turn a stern eye upon games players, or on the young, but that we attempt to provide as wide a range of services as we possibly can, while still encouraging the serious use of the machine and its facilities.

ITUG is therefore not a club, and while it hasn't yet done anything other than make a considerable loss each year, it is supposed to eventually become self-financing (one way of saying that any profits get ploughed back into the group).

I always try to make any potential subscriber aware that ITUG is not a club, and I also try to make that clear from time to time in the newsletter.

It is my intention to continue running ITUG for as long as there are a dozen TI owners sufficiently enthusiastic to want me to, and I am working hard to make sure that ITUG's facilities are always improving.

What I propose is that in order to maintain the cohesive structure that has been TI-EXCHANGE and which has been of such immense service to its members, that I take on the membership of TI-EXCHANGE (if they are willing) and incorporate them within ITUG.

What will this mean for the average TI-EXCHANGE member?

First, an annual subscription of £11 for the UK, and £12 for all the overseas subscribers (that's for the new ITUG subscription year starting on JUNE 1st.).

Second, a monthly newsletter in a format similar to TI*MES (i.e., A5 booklet), which currently runs to 32 pages a month, and could rise to 40 pages if the number of members is large enough.

Third, access to at least three volumes of back issues of International TI-LINES, comprising some 1000+ pages of information.

Fourth, access to a private software collection (the former TIHOME SOFTWARE COLLECTION which has been revamped and now consists of 160 programs with a 56 page catalogue).

Fifth, access to a library of Public Domain/Freeware/Fairware/Shareware items (largely catering for disk drive owners) which is continually growing.

Sixth, cheap repair and servicing of their hardware.

Seventh, from time to time we get our hands on small stocks of modules and other items, which are now only available to current ITUG subscribers.

Eighth, a hard core of enthusiastic hardware and software buffs who are always looking for new projects for the home constructor.

Ninth, a number of equally enthusiastic, but not necessarily expert or genius, contacts who have put themselves forward to act as focal points for ITUGers local to them (or far away even).

Tenth, a commitment to provide all Users with as much information as they are ever likely to need. Some ITUGers have told me that in the past they have skipped over articles which they found uninteresting, only to retrace their steps some months later and be pleasantly surprised by the same articles - because their interests and experiences have changed in the meantime.

Eleventh, ITUG is currently in the throes of setting up the first of what may be several Bulletin Boards, which are intended to act as focal points for TI Users.

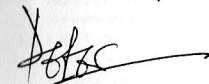
Contributors who currently write in TI*MES are more than welcome to submit their material for publication in TI-LINES - we don't want to repeat the TIHCUC fiasco where all the dedicated contributors were alienated by a rather rude organisation!

Alternatively, as has happened with the MALTBY TI USERS, and the WEST MIDLANDS TI USERS, if TI-EXCHANGE members who live near each other decide to organise themselves into a formal (or a loose) association, one or more members could subscribe to ITUG and act as representatives of their own group. This would enable them to have a local club run on a shoestring budget, and yet still have a monthly magazine to help keep them in touch with everyone else.

I hope to attend the Derby meeting in May should anyone have any comments or criticisms of this proposal to merge one group into the other; it is the best way that I can think of to keep everyone together under one banner. Alternatively, respondents can either ring me on Oxford 510822 evenings, or write to me at the address above.

I have gabbled on for far too long (sounds familiar), but I hope that I have managed to make clear the gist of what I'd like to do.

All the best,


Baldie Pete Brooks

96 Banbury Road, OXFORD OX2 6JT

We received this open letter from Peter Brooks. TI-LINES is an excellent monthly, worth considering. The choice is yours. Those of you who do not wish to renew membership with this Nationwide users group could consider supporting an alternative, ie form a local TI99/4a group. It is my wish you will give serious thought to supporting all those hard working folks helping 99ers make the most of our orphan TI99/4a Home Computer.

Clive S.

A Light Pen
(from Oct. 86 issue of The San Francisco 99ers)

What you will need is a 9-pin D-plug that will easily fit in the joystick port. These are easily found at Radio Shack. Wire, the braided stuff to go between the pen and plug. A CDS photo cell and a junk flair pen.

First, cut the writing end off the pen and cut it. Drill a hole in the other end of the pen that two wires will fit through. Now take the cap and poke the end out of it.

Now solder two lengths of about 40 inches of wire to each lead on the CDS cell. Run these through the cap and pen and glue the CDS cell in the cap hole. I filled my pen with silicone rubber. Takes 24 hours to set up and smells, but it holds great.

The two wires are soldered to pins 9 and 7 of the D-plug. You can make two. Hook the second pen to #2 and #5 of the D-plug.

And here is the software:

```

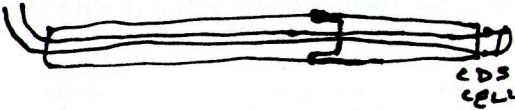
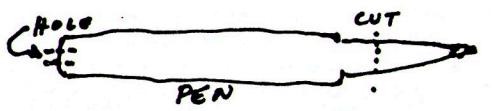
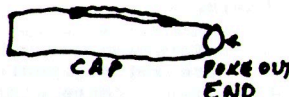
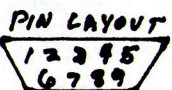
300 CALL CLEAR :: CALL SCREEN(2)
310 FOR X=1 TO 10 :: CALL COLOR((X,5-11*(X)8),1)::
NEXT X
320 RANDOMIZE
330 FOR X=0 TO 2
340 CALL CHAR(96+8*X,"3C7EFFFFFF7E3C")
350 CALL COLOR(9+X,1,1)
360 NEXT X
370 PRINT " 'hp'hp 'hp 'hp'hp' hp' ' p'h
'p'h hp' h hp' h p'h hp' h hp' h p'
h"
380 PRINT " 'hp' h hp' h p'h hp' h hp' h
p'h hp' ' p'h p'h"
390 PRINT "'hp'hp 'hp p'h":::
400 PRINT : "'hp'hp'hp'hp'hp'hp'hp'hp'p TI 99
LITE PEN hh TOUCH DOT TO CONTINUE. p'ph'p'p
h'ph'ph'ph'ph'"
410 CALL JOYST(1,X,Y):: IF X<0 OR Y<0 THEN 400
420 FOR C=1 TO 3
430 CALL COLOR(9,7-4*(C=1)-8*(C=2),1)
440 CALL COLOR(10,7-4*(C=2)-8*(C=3),1)
450 CALL COLOR(11,7-4*(C=3)-8*(C=1),1)
460 NEXT C
470 GOTO 410
480 CALL SOUND(100,440,0):: CALL CLEAR :: SC=0
490 DISPLAY AT(12,9):"EASY HARD"
500 DISPLAY AT(16,7):"SELECT DIFFICULTY" :: DISPLAY
AT(18,1):"EASY=LARGE DOTS, HARD=SMALL."
510 CALL HCHAR(12,9,112):: CALL HCHAR(12,19,104)
520 CALL COLOR(11,16,16,10,2,2)
530 FOR I=1 TO 10
540 CALL JOYST(1,X,Y):: IF X=4 THEN CALL MAGNIFY(2)::
GOTO 610
550 NEXT I
560 CALL COLOR(11,2,2,10,16,16)
570 FOR I=1 TO 10
580 CALL JOYST(1,X,Y):: IF X=4 THEN CALL MAGNIFY(1)::
GOTO 610

```

```

590 NEXT I
600 GOTO 520
610 CALL CLEAR
620 CALL SPRITE(#1,96,16,92,124)
630 DISPLAY AT(16,6)BEEP:"TOUCH DOT TO START"
640 CALL JOYST(1,X,Y):: IF X=4 THEN 640
650 CALL SOUND(-100,220,5):: CALL SOUND(-100,880,0)
660 CALL CLEAR
670 FOR L=1 TO 20
680 CALL SPRITE(#1,96,16,INT(RND0)+1,INT(RND0)+10)
690 N=0
700 CALL JOYST(1,X,Y)
710 IF X=4 THEN 730
720 N=N+1 :: GOTO 700
730 SC=SC+N :: CALL SOUND(-100,440,5)
740 DISPLAY AT(1,1):"SCORE ";SC
750 FOR X=1 TO 100 :: NEXT X
760 NEXT L
770 FOR Z=1 TO SC STEP 10
780 CALL SOUND(-100,Z+110,0)
790 NEXT Z
800 CALL CLEAR :: CALL SPRITE(#1,96,16,150,123)
810 DISPLAY AT(10,6):"YOUR SCORE IS ";SC :: DISPLAY
AT(18,3):"TOUCH DOT TO PLAY AGAIN."
820 IF SC<150 THEN 830 ELSE DISPLAY AT(12,6):" YOU
CAN'T FOOL ME! YOU CHEATED!" :: GOTO 900
830 IF SC<200 THEN 840 ELSE DISPLAY AT(12,1):" VERY
GOOD!" :: GOTO 900
840 IF SC<225 THEN 850 ELSE DISPLAY AT(12,1):" WOW!
YOU NAMED RAMBO?" :: GOTO 900
850 IF SC<250 THEN 860 ELSE DISPLAY AT(12,1):" NOT
BAD,BUT YOU PAY SOMEONE TO SUAT FLIES FOR YOU?"
:: GOTO 900
860 IF SC<300 THEN 870 ELSE DISPLAY AT(12,1):" YOU
NEED PRACTICE!" :: GOTO 900
870 IF SC<350 THEN 880 ELSE DISPLAY AT(12,1):" HAVE
YOU CONSIDERED CHECKERS?" :: GOTO 900
880 IF SC<400 THEN 890 ELSE DISPLAY AT(12,1):" HELLO!
ANYBODY AWAKE OUT THERE?" :: GOTO 900
890 DISPLAY AT(12,1):"TRY POINTING THE PEN AT THE DOT"
900 FOR Z=1 TO 500 :: CALL JOYST(1,X,Y):: IF X=4 THEN
930
910 NEXT Z
920 CALL CLEAR :: DISPLAY AT(12,1):" DOTS ALL FOLKS!"
:: END
930 CALL DELSPRITE(ALL):: GOTO 400

```



" BRAZOS VALLEY 99'ERS "

PRINTER COMMANDS

(energizes or turns on)

	10X	SG-10	MX-80	FX-80	KX-P1091	OKIDATA
ITALICS	127 52	127 52	*****127 52	127 52	127 52	*****
ELITE	127 66 2	127 66 2	*****127 77	127 77	127 77	! 28
CONDENSED	127 15	127 15	127 15	127 15	127 15	! 29
PICA	127 66 1	127 66 1	*****127 80	127 80	127 80	! 30
EXPANDED	127 87 1	127 87 1	127 14	127 87 1	127 87 1	! 31
SUPERSCRIPIT	127 83 0	127 83 0	***** 27 83 0	127 83 0	127 83 0	127 74
SUBSCRIPIT	127 83 1	127 83 1	*****127 83 1	127 83 1	127 83 1	127 76
NEAR LETTER	*****127 65 4	*****127 120 1	*****127 110	127 49	127 49	127 49
EMPHASIZED	127 69	127 69	*****127 69	127 69	127 69	127 84
UNDERLINE	127 45 1	127 45 1	*****127 45 1	127 45 1	127 45 1	127 67
DOUBLE STRIKE	127 71	127 71	127 71	127 71	127 71	127 72
SLASHED ZERO	*****127 92 1	*****	*****	*****	*****	*****
1/8 LINE SP.	127 48	127 48	127 48	127 48	127 48	127 56
1/6 LINE SP.	127 50	127 50	127 50	127 50	127 50	127 54
7/72 LINE SP.	127 49	127 49	127 49	127 49	127 49	*****
n/72 LINE SP.	127 65 n	127 65 n	127 65 n	127 65 n	*****	*****
n/144 LINE SP.	127 51 n	127 51 n	*****	*****	*****	127 37 57 n
n/216 LINE SP.	*****	*****	*****127 51 n	*****	*****	*****
TOP MARGIN	127 82 n	127 82 n	*****	*****	*****	*****
BOTTOM MARGIN	127 78 n	127 78 n	127 78 n	127 78 n	*****	*****
LEFT MARGIN	127 77 n	127 77 n	*****127 108 n	*****	*****	*****
RIGHT MARGIN	127 81 n	127 81 n	*****127 81 n	*****	*****	*****
COLUMN WIDTH	*****	*****127 81 n	*****	*****	*****	*****
PAGE LTH. LINES	127 67 n	127 67 n	27 67 n	127 67 n	*****	*****
PAGE LTH. INCHES	127 67 0 n	127 67 0 n	*****127 67 0 n	*****	*****	*****
PAPER OUT "OFF"	127 56	127 56	127 56	127 56	127 56	*****
PROPORTIONAL	*****127 112	*****127 112	127 111	*****	*****	*****
RESET PRINTER	127 64	127 64	*****127 64	*****	*****	24

ANNOUNCING

THE ALL-NEW, SUPER-DUPER, HANDY-DANDY, 98 CENT, DO-IT-YOURSELF, WAXPAPER

R. L. E. DIGITIZER!

BY: RAY KAZMER

When I saw my first R.L.E., I thought, "GOLLLLL-LEEEE!" I'd SHORE like to draw ME a pit-chur like THAT!! Then I found out that it takes something called a "digitizer" to make an R.L.E. and THOSE things could cost a LOT more than my '66 Chevy (fer-shirrrrr!) Since my TI-ARTISTIC talents were FAR from perfect, I decided I'd try to make a CHEAP digitizer, one which required very little talent to use, but would yield a fairly good R.L.E.

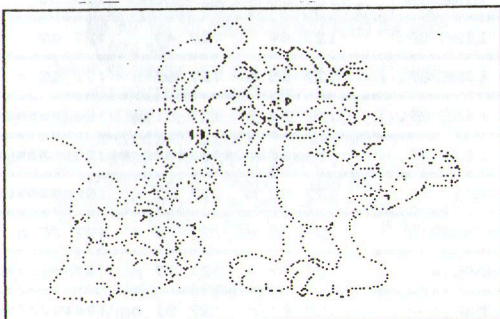
"Tracing" a picture, then sticking the paper to my TV screen, so I could move TI-ARTIST's cursor under it (drawing as I went) seemed a good idea, but regular tissue paper wouldn't let me see my cursor CLEARLY enough! I tried "plastic wrap," which certainly DID allow me to see the cursor but wouldn't hold ANY kind of ink! Besides, one touch and it was all SMUDGE, SMUDGE, SMUDGE! And you know how it LOVES to "cling to itself!" Mur-der!

While shopping, I spotted a roll of WAXPAPER (98 cents for 100 feet) AND a (9"X12") cardboard folder (with "pockets" inside) used by school kids. Though the folder was way too big for my TV screen, the drawings of ODIE and GARFIELD on the cover (my favorites!) seemed to be just about right!

At home, I taped a hunk of waxpaper onto the folder, then QUICKLY traced over every line, "etching" the image into the waxpaper with a mechanical pencil (with the lead retracted.) THAT WAS A MISTAKE!!! If you decide to try my "digitizer" yourself, trace with GREAT CARE! Make your tracing as ACCURATE as possible! Care NOW, will save you LOADS of "correcting time" later, when you are completing your "on-screen" master-piece! Be SURE to hit ALL lines, BEFORE you remove the waxpaper copy from your "original."

Next, load TI-ARTIST and put a "frame" around the drawing screen, which helps to align the copy vertically, and can be erased later. Be SURE the copy lies WITHIN this frame, THEN tape it to your screen.

THIS PART IS MOST IMPORTANT! Find a comfortable position, "head-on" to the screen, and begin to "outline" the copy, by placing "DOTS" BEHIND the waxpaper lines. (See sample) DO NOT shift your head sideways! That causes DISTORTION and is HARD to repair later!



AGAIN, the same words of CAUTION apply when placing the dots as when you were making your WAXPAPER tracing, which is: TAKE YOUR TIME! Do NOT rush to finish it fast! CAREFULLY place each dot, as CLOSE to the "center" of each line, as possible! Although this will SEEM like a long, TEDIOUS job to you (and it IS) try to think of it as "building a strong foundation."

There is NO WAY you can follow a "traced" line by just pushing your joystick and mashing the fire-button! You'll see the cursor "weave all over the road" like a drunk driver! Before trying to make your first WAXPAPER R.L.E., plan to spend several hours with it. Be patient! Persevere! Your determination and care WILL be rewarded with a real work of art! (AMEN!)

It gets easier now as you play "connect the dots." You may find the ZOOM feature a real help with this. Another tip: SAVE the picture frequently! If you make a major boo-boo, you won't lose a TOO much time and sweat by simply reloading the SAVED picture, rather than struggling to repair it.

The FINAL STEP is to give your picture a good "polishing," OR what I had referred to earlier as "correcting time." If you took the time to do all the first steps PROPERLY and your picture is now "connected" simply view "THE BIG PICTURE" and all the "rough spots" will LEAP RIGHT OUT at you!! Adding or erasing a single pixel here and there, is all that remains. It sounds simple, doesn't it? (THIS is the HARDEST part!) After you've done all the "correcting" you THINK you can find, SAVE it, then store it away someplace (for a week or two) THEN reload it and compare your picture to the original. If you can't find ANYTHING else wrong with it, it is DONE! (Use MAX-RLE to convert your TI-ARTIST "PICTURE_P" file into a MAX-RLE.)

Some last tips: DON'T strive for ABSOLUTE PERFECTION! That's IMPOSSIBLE! (Garfield's "stripes" nearly ran me up a wall!!) BUT, by the same token, if you've waited those two weeks and you spot another "flaw," DO attempt fixing it! IF (due to limitations inherent in our consoles or TI-ARTIST, OR due to approaching blindness) you CAN'T fix it (after trying for five or six years) make up some "logical sounding" excuse, when you debut the master-piece. If you make it "high-tech" enough, ANYBODY will buy it! MY winning line is: "Well, NOBODY can draw a PERFECT, curved zig-zag line!"

So, here it is! My COMPLETED work of art! It's NOT a 100% PERFECT copy of the original but what can you expect from a console with an overloaded framistan in it's quadilop?!

There are TONS of "copiable" pictures, for your "WAXPAPER R.L.E. DIGITIZER!" (Coloring books for children, atlases, magazines, calanders, etc.,) and if any 99'ERS out there, try doing some PLAYBOY stuff well, I'd appreciate a copy, (before I go totally blind!)

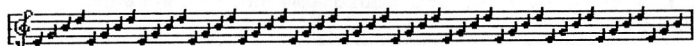
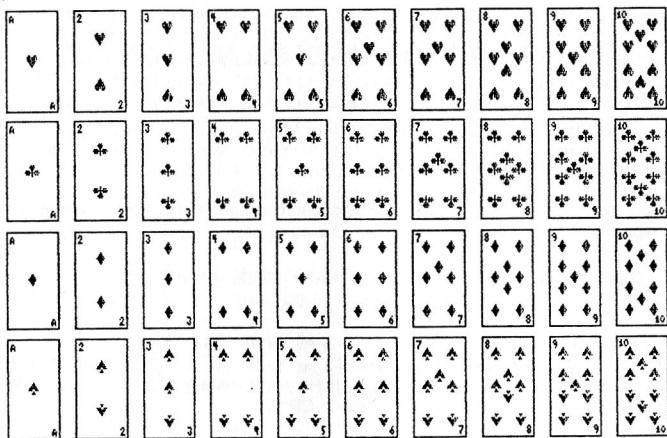


After ALL THAT WORK, it's time for some FUN! Here's a RIDDLE for all you sharp-eyed TI-RUNNER players. WHERE (in TI-RUNNER) do the initials "IBM" appear on screen? HERE'S A CLUE: Play the game up to Level 28, then look in the bricks, but don't look TOO CLOSELY, or you MIGHT miss them!) R.K.

Stephen Shaw welcomes YOUR pictures, and will pass on to Ray Kazmer any digitised pictures from Playboy!



TI WRITER GRAPHICS



BY JAMES STRINGFELLOW

THESE GRAPHICS WERE PRINTED DIRECTLY FROM THE KEYBOARD USING TI WRITER AND EPSON MX80 PRINTER

Here is a program which should be useful for people using graph paper to design graphics for TI-WRITER.

It is for use with the EPSON 80 printer and should be easy to convert. I have included at the end of the program some changes to be made if your printer head has the top wire numbered 64 instead of 1 as with the EPSON 80.

The screen is 14 dots high and 16 dot rows long which should make quite large characters.

You can inverse your graph by pressing "M" for mirror.

After saving to a disk you can load into your TI-WRITER for printing with your letter.

```

10 !*****
12 !*
14 !* TIWRITER FONT MAKER *
16 !*
18 !* VERSION 1 1987 *
20 !*
22 !* James Stringfellow *
24 !*
26 !*****
27 !
100 CALL CLEAR :: CALL SCREE
N(5):: CALL CHAR(128,"FFB181
B1818181FF",122,RPT$("0",16)
,129,RPT$("F",16))
110 FOR I=5 TO 12 :: CALL CO
LOR(I,2,1):: NEXT I :: CALL
COLOR(13,2,11,0,5,5,1,1,13,
3,2,13,4,2,13):: P$=RPT$(CHR
$(128),16)
120 X=64 :: CH=128 :: FOR R=
4 TO 17 :: DISPLAY AT(R,4):S
TR$(X);TAB(7);P$: X=X/2 ::
IF X<1 THEN X=64
130 NEXT R :: CALL VCHAR(1,3
1,28,96):: Z$=RPT$(CHR$(32),
28)
140 DISPLAY AT(21,1):"zMovez
withzzWzEzRzSzDzZzXzCz": "zQz
Togglezonzoffzzzzzstopzzz": "
zKclearzzPzprintzzMzmirrorz
": "zzzzzzzzzzzzzzzzzzzzzzzz
zzz" :: M$,M2$=""
150 DISPLAY AT(1,7):"zTIWRIT
ERzFONTsz": Z$:Z$: DISPLAY
AT(18,1):Z$:Z$:Z$: R=4 ::
C=9 :: K=75 :: CALL SPRITE(
1,128,16,25,65)
160 REM
170 CALL KEY(3,K,S):: Y=(K=6
9 OR K=82 OR K=87)-(K=88 OR
K=90 OR K=67):: X=(K=83 OR K
=87 OR K=90)-(K=68 OR K=82 O
R K=67):: IF K=81 THEN 190 ::
IF K=80 THEN 200 :: IF K=7
5 THEN 120
180 IF K=64 THEN 370 :: IF K
=77 THEN 330 :: R=R+Y :: C=C
+X :: R=R-(R<4)+(R>17):: C=C
-(C<9)+(C>24):: CALL HCHAR(R
,C,CH):: CALL LOCATE(#1,R*8-
7,C*8-7):: GOTO 170
190 CH=CH+1+(CH=129)*2 :: CA
LL HCHAR(R,C,CH):: FOR I=1 T
O 30 :: NEXT I :: GOTO 170
200 GOSUB 350 :: FOR C=9 TO
24 :: X=64 :: FOR R=4 TO 10
:: CALL GCHAR(R,C,G):: IF G=
129 THEN A=A+X
210 X=X/2 :: NEXT R :: FOR J
=1 TO LEN(STR$(A)):: CALL VC
HAR(J,C,ASC(SEG$(STR$(A),J,1
))):: NEXT J :: M2$=M2$&CHR$(
A):: A=0 :: NEXT C

```

```

220 FOR C=9 TO 24 :: X=64 ::
FOR R=11 TO 17 :: CALL GCHA
R(R,C,G):: IF G=129 THEN A=A
+X
230 X=X/2 :: NEXT R :: FOR J
=1 TO LEN(STR$(A)):: CALL VC
HAR(17+J,C,ASC(SEG$(STR$(A),
J,1))):: NEXT J :: M$=M$&CHR
$(A):: A=0 :: NEXT C :: CALL
DELSPRITE(ALL):: CALL MAGNIF
Y(1)
235 CALL SPRITE(#1,128,16,18
5,233):: DISPLAY AT(24,6):"z
zzzzzzDoublezDensityzN" :: A
CCEPT AT(24,28)VALIDATE("YN"
)SIZE(-1):Q$: :: K=75 :: IF Q
$="Y" THEN K=K+1
240 OPEN #1:"RS232.BA=4800"
250 PRINT #1:CHR$(27)&"A"&CH
R$(7)
260 PRINT #1:CHR$(27)&CHR$(K
)&CHR$(16)&CHR$(0)&M2$&CHR$(
13)
270 PRINT #1:CHR$(27)&CHR$(K
)&CHR$(16)&CHR$(0)&M$: :: CLO
SE #1
280 DISPLAY AT(24,6)SIZE(23)
:"PrintzagainzyeszorznzN" ::
ACCEPT AT(24,28)VALIDATE("
YN")SIZE(-1):Q$: :: IF Q$="Y"
THEN 235
290 DISPLAY AT(24,6):"zzzzzzz
zzzzSavezyeszorzn" :: ACCEPT
AT(24,28)VALIDATE("YN")SIZE
(-1):Q$: :: IF Q$="N" THEN 14
0
300 CALL LOCATE(#1,185,145):
: DISPLAY AT(24,2):"zzzFilen
amezDSK" :: ACCEPT AT(24,17)
SIZE(-12):F$: :: IF F$="" THE
N 140 :: OPEN #2:"DSK"&F$,FI
XED 80
310 PRINT #2:CHR$(27)&"K"&CH
R$(16)&CHR$(0)&M2$
320 PRINT #2:CHR$(27)&"K"&CH
R$(16)&CHR$(0)&M$: :: CLOSE #
2 :: GOTO 140
330 FOR R=4 TO 17 :: X=64 ::
FOR C=9 TO 24 :: CALL GCHAR
(R,C,CH):: IF CH=128 THEN CA
LL HCHAR(R,C,129,1)ELSE IF C
H=129 THEN CALL HCHAR(R,C,12
8,1)
340 NEXT C :: NEXT R :: CH=C
H+1+(CH=129)*2 :: GOTO 140
350 DATA 80,76,69,65,83,69,3
2,87,65,73,84
360 CALL DELSPRITE(#1): CAL
L MAGNIFY(2):: FOR I=2 TO 22
STEP 2 :: READ Y :: CALL SP
RITE(#I,Y,16,I*8-7,I+I*8-7+1
6):: NEXT I :: RESTORE :: RE
TURN
370 END

```


Tom Arnold - Secretary/Coordinator

77 Lavina Crescent Hamilton, Ontario L9C 5S8 (416) 385-5576

RANDOM
by
W. M. Johnson
 No 331
 UK TI User Group
 Is a member in good standing of
 the Channel 99/Hamilton
 Users-Group

CH 99 HAM
 Users-Group

The random number is probably one of the most useful aspects of a computer. There are of course scientific arguments as to what constitutes random, but at this stage I am not interested in the philosophical attributes of electronics.

Random can be used for many things, but first you must understand what RANDOM is, then you can worry about how to use it. Below is a short program that will run in basic or extended basic that produces a series of random numbers that could be used to pick lottery numbers, or possibly several other uses, such as the English soccer pools.

If you just want to use it for Lotto or such, type it in and off you go. If you would like to know how it works, then read on.

```

100 CALL CLEAR
110 REM *****
120 REM * INPUT *
130 REM *****
140 INPUT "NUMBERS PER TICKE
T ":TOTAL
150 INPUT "NUMBER FROM 1 TO
":TOP
160 INPUT "HOW MANY TICKETS
":Z
170 CALL CLEAR
180 REM *****
190 REM *PICK RND NUMBER*
200 REM *****
210 FOR LOOP=1 TO Z
220 RANDOMIZE
230 FOR I=1 TO TOTAL
240 A(I)=INT(RND*TOP)+1
250 FOR X=1 TO I-1
260 IF A(X)=A(I)THEN 240
270 NEXT X
280 NEXT I
290 REM *****
300 REM *ANSWER PRINTER*
310 REM *****
320 FOR J=1 TO TOTAL
330 PRINT A(J);
340 NEXT J
350 PRINT
360 PRINT
370 NEXT LOOP
380 END

```

Random on the T. I. computer is a number that is less than ONE. That is to say a fraction, a metric fraction. If you ask your computer "PRINT RND ". You will see that it will print a decimal fraction less than one. The fact that it is less than one makes it very useful, and versatile.

If you take the fraction produced by the computer and multiply it by another number, what would be the result? SIMPLE, the result must be less than the multiplier because we are multiplying a fraction. For example if the random number were 0.999999, and we multiply by ten, as you should have learned in school, move the decimal point one to the right. The answer then is 09.99999. Or if you like 9 and a bit.

What I am trying to show is that the greatest number available from a random will always be, at best only a smidgen less than the multiplier, at worst it could be very small, less than one. For example if the random number produced by the computer were 0.000001. And we multiply by 100000, which is a very large number the result would be 0.1, which is still less than one whole one.

Is this making any sence? So far all we have proved is that a random number can be controlled by multiplying it by a control number. Then the random result will always be between just less than the control number and a bit more than nothing.

Let's look at it another way. If we wanted to produce a random number between 1 and 32, (the character width of the screen), how would we set about it? Again it is so simple, first we employ the notation (INT). Int, is short for integer, which means every thing to the left of the decimal place. For example 99.222 as an integer is 99. As there is nothing to the right of the decimal place, we may as well ignore it.

Why don't we try and see if this really works! Type on your computer PRINT INT(12.9999999) then press enter. All you get is twelve, right! Now we'll do something really fun filled. Type on you computer, PRINT INT(RND) and press enter . The answer is nothing or "0". the reason being RND will always be on the right on the decimal place, always less that one.

Let us not be disconcerted, for now we have discovered the secret of generating a number between given limits. We still want to produce a number from 1 to 32. Multiplying random by 32 will produce a number between almost nothing and almost 32. NOW think on this ALMOST nothing INTEGRERED is nothing plus one will always be one. Almost 32 INTEGRERED will be 31. BUT 31+1=32.

So as I said it's simple, to get a number between 1 and 32 the formula, or statement is INT(RND)+1. (Take the integer of the random and add one.) This can be used to create any random whole number INT(RND*5)+1 will produce a whole number between 1 and 5. See how easy it is. Now that you have a full understanding of random I shall endeavour to explain how the lottery number picker works.

Line 240 is the only line of any importance this one actually calculates the number between the predetermined limits. A(I) is the variable that will become loaded with the random number. TOP is only a variable designating the top number allowable .

LINE by LINE explanation:-

```

100 clears the screen.
110 REMark statement outlining subroutine title.
120 REMark statement "title of subroutine".
130 REMark underlining title.
140 Gives a value to TOTAL equal to how many numbers you want on the ticket.
150 Another input statement loading the variable TOP, this sets the high number limit.
160 The last input statement loading the variable "Z". Z represents how many tickets you want the computer to produce.
170 Clears all the inputs off the screen, ready for the tickets to be printed.
180 REMark outline for title of subroutine.
190 REMark Title.
200 REMark underline.
210 Loop represents how many times the generator will be asked to produce a ticket to the total number of "Z".
220 A random number produced by the computer will always be in a set sequence, (just the nature of the beast). RANDOMIZE will cause the generator to chose a random number at random, thus never producing the same sequence twice.
230 sets up how many numbers will be on each ticket.
240 The random generator.
250 A loop to test each number against those already produced.
260 Compares the random number A(X) to the one just produced, if a duplicate exists the program control will revert back to line 240 and try another number.

```

270 Continue the X loop.
280 Continue the I loop when the X loop is finished.

290 REMark.
300 REMark.
310 REMark.

320 Sets up the loop to print the total amount of numbers on each ticket. TOTAL is the variable that has already been loaded.

330 Prints the variable A(J) (the random number) in a horizontal line, because of the ";".

340 Continue the J loop to the TOTAL number.

350 Breaks the horizontal line print instruction, because it is not followed by a ";".

360 Just a blank space.

370 Next loop, to print the next ticket, when all tickets are printed the program stops as there is no line 380. Though a line 380 could be added.

380 GOTO 140

Now you know as much as I do about the generation of random numbers. Let's see if you can come up with some real unusual uses for this type of thing.

END



TIT-BITS

```
#####
!
! Try this program out for
! size. The SPRITE demo
! is fantastic.
!
! EXTENDED BASIC ONLY
!
#####
```

```
100 ! ***** SPIRACLE *****
110 !
120 ! BY GEME KRACKEDWITS
130 !
140 !
150 CALL CLEAR
160 CALL SCREEN(2)
170 CALL MAGNIFY(3)
180 A$="07182040800000000000
00400201007E010040202010101
010101020204JBE0"
190 B$="071F3F7E7FFFFFFF
FF7F7F3F1F07E0FBFCFEFF
FFFFFFFFEFCEFB0"
200 CALL CHAR(140,A$)
210 CALL CHAR(136,B$)
220 CALL SPRITE(0,136,14,92
,124)
230 X=00
240 Y=127
250 R=00 :: P=2
260 FOR I=0 TO 2*PI STEP (2*
PI)/10
270 A=X+R*COS(I)
280 B=Y+R*SIN(I)
290 P=P+J
300 CALL SPRITE(0P,136,I*2+3
,A,B,0,0)
310 NEXT I
320 V=20
330 CALL SPRITE(01,140,16,00
,90)
340 FOR I=0 TO 2*PI STEP (2*
PI)/36
350 A=INT(V*COS(I))
360 B=INT(V*SIN(I))
370 CALL MOTION(01,A,B)
380 NEXT I
390 IF C=1 THEN 400 ELSE C=C
+1 :: GOTO 340
400 CALL DELSPRITE(01):: P=P
-1
410 V=2 :: C=0
410 FOR I=0 TO 2*PI STEP (2*
PI)/18
420 A=INT(V*COS(I))
430 B=INT(V*SIN(I))
440 P=P+1 :: IF P>20 THEN P=
3
450 CALL MOTION(0P,A,B)
460 V=V+.05
470 NEXT I
480 IF C=1 THEN 490 ELSE C=C
+1 :: GOTO 410
490 C=C :: GOTO 250
```

```
100 CALL CLEAR
110 CALL SCREEN(16)
120 PRINT " M A T H A M A
T I C S "
130 PRINT
140 PRINT
150 PRINT
160 PRINT
170 PRINT TAB(12);"BY VDN KL
IMPEL"
180 PRINT
190 PRINT TAB(17);"TICHUG"
200 GOSUB 2260
210 INPUT "PLEASE ENTER YOUR
NAME?":NAME$
220 GOSUB 2310 REM DELAY
230 CALL CLEAR
240 PRINT TAB(11);"OK ";NAME
$
250 PRINT
260 PRINT TAB(9);"YOUR CHOIC
E"
270 PRINT
280 PRINT
290 PRINT
300 PRINT "1. INSTRUCTIONS"
310 PRINT
320 PRINT "2. MULTIPLICATION
TABLES "
330 PRINT
340 PRINT "3. RANDOM MULTIPL
ICATIONS"
350 PRINT
360 PRINT "4. ADDITION "
370 PRINT
380 PRINT "5. SUBTRACTION"
390 PRINT
400 PRINT "6. DIVISION "
410 PRINT
420 PRINT
430 PRINT
440 PRINT
450 PRINT
460 PRINT "(1,2,3,4,5 OR 6)"
;
470 INPUT CHOICE
480 ON CHOICE GOSUB 500,700,
950,1170,1300,1620,2510
490 GOTO 2400
500 REM INSTRUCTIONS
510 CALL CLEAR
520 PRINT TAB(9);"INSTRUCTIO
NS"
530 PRINT
540 PRINT " THIS PROGRAM
ALLOWS YOU TO CHOOSE THE MAT
```

```
HS FUNCTION "
550 PRINT
560 PRINT " YOU WILL BE G
IVEN A CHOICE OF FUNCTIO
NS AND ALL YOU HAVE TO DO IS
PRESS THE APPROPRIATE ";
570 PRINT "ANSWER."
580 PRINT
590 PRINT " TO STOP THE P
ROGRAM AT ANY TIME JUST ENT
ER '000' AS YOUR ANSWER."
600 PRINT
610 PRINT " TO RETURN TO
THE INDEX LIST AND CHANGE T
HE MATHS FUNCTION, JUST TY
PE IN THE WORD 'INDEX' AS Y
OUR ANSWER"
620 PRINT
630 PRINT
640 PRINT
650 PRINT
660 PRINT "(PRESS ANY KEY TO
CONTINUE)";
670 CALL KEY(0,KEY,STATUS)
680 IF STATUS=0 THEN 670
690 GOTO 230
700 REM MULTIPLICATION
TABLES.....
710 CALL CLEAR
720 PRINT TAB(9);"RIGHT ";NA
ME$
730 PRINT
740 PRINT
750 PRINT "WHICH MULTIPLICAT
ION TABLE WOULD YOU LIKE TO
DO ";
760 INPUT TABLE
770 GOSUB 2310
780 CALL CLEAR
790 N=0
800 N=N+1
810 PRINT TAB(6);"TRY THIS O
NE ";NAME$
820 PRINT
830 PRINT
840 PRINT TAB(9);N;"X";TABLE
;
850 INPUT "=:ANS$
860 IF ANS$="INDEX" THEN 230
870 LET ANS=VAL(ANS$)
880 IF ANS=000 THEN 2400
890 IF ANS<N+TABLE THEN 930
900 GOSUB 2000
910 PRINT
920 GOTO 800
930 GOSUB 1900
940 GOTO 840
950 REM RANDOM MULTIPLICATIO
NS
960 CALL CLEAR
970 PRINT TAB(6);NAME$;
980 PRINT " TRY THIS ONE";
990 PRINT
1000 PRINT
1010 RANDOMIZE
1020 FIRST=INT(10*RAND)+1
1030 SECOND=INT(10*RAND)+1
1040 PRINT TAB(9);FIRST;
```

```
1050 PRINT "X";
1060 PRINT SECOND;
1070 PRINT "=";
1080 INPUT ANS$
1090 IF ANS$="INDEX" THEN 23
0
1100 LET ANS=VAL(ANS$)
1110 IF ANS=000 THEN 2400
1120 IF ANS<FIRST*SECOND TH
EN 1150
1130 GOSUB 2000
1140 GOTO 960
1150 GOSUB 1900
1160 GOTO 1040
1170 REM ADDITION
1180 CALL CLEAR
1190 PRINT TAB(6);"TRY THIS
ONE ";NAME$
1200 RANDOMIZE
1210 FIRST=INT(100*RAND)+1
1220 SECOND=INT(100*RAND)+1
1230 PRINT
1240 PRINT
1250 PRINT TAB(9);FIRST;
1260 PRINT "+";
1270 PRINT SECOND;
1280 PRINT "=";
1290 INPUT ANS$
1300 IF ANS$="INDEX" THEN 23
0
1310 LET ANS=VAL(ANS$)
1320 IF ANS=000 THEN 2400
1330 IF ANS<FIRST+SECOND TH
EN 1360
1340 GOSUB 2000
1350 GOTO 1190
1360 GOSUB 1900
1370 GOTO 1230
1380 REM SUBTRACTION
1390 CALL CLEAR
1400 PRINT TAB(6);"TRY THIS
ONE ";NAME$
1410 PRINT
1420 PRINT
1430 RANDOMIZE
1440 FIRST=INT(100*RAND)+1
1450 SECOND=INT(100*RAND)+1
1460 IF SECOND>FIRST THEN 14
30
1470 PRINT TAB(6);FIRST;
1480 PRINT "-";
1490 PRINT SECOND;
1500 PRINT "=:";
1510 IF FIRST<SECOND THEN 13
00
1520 INPUT ANS$
1530 IF ANS$="INDEX" THEN 23
0
1540 LET ANS=VAL(ANS$)
1550 IF ANS=000 THEN 2400
1560 IF ANS<FIRST<SECOND TH
EN 1590
1570 GOSUB 2000
1580 GOTO 1390
1590 GOSUB 1900
1600 GOTO 1470
1610 RETURN
1620 REM DIVISION BY A SET
```

```
FIGURE
1630 CALL CLEAR
1640 PRINT TAB(9);"OK ";NAME
$
1650 PRINT_
1660 PRINT
1670 PRINT "WHICH NUMBER DO
YOU WISH TO DIVIDE BY";
1680 INPUT TABLE
1690 GOSUB 2260
1700 CALL CLEAR
1710 GOSUB 2310
1720 PRINT
1730 CALL CLEAR
1740 RANDOMIZE
1750 FIRST=INT(100*RAND)+1
1760 IF FIRST/TABLE<INT(FIR
ST/TABLE)THEN 1700
1770 PRINT TAB(9);"OK ";NAME
$
1780 PRINT
1790 PRINT TAB(8);"TRY THIS
ONE"
1800 PRINT
1810 PRINT
1820 PRINT
1830 PRINT TAB(3);FIRST;
1840 PRINT "DIVIDED BY";
1850 PRINT TABLE;
1860 PRINT "=:";
1870 IF FIRST/TABLE<INT(FIR
ST/TABLE)THEN 1730
1880 IF FIRST<=TABLE THEN 17
00
1890 INPUT ANS$
1900 IF ANS$="INDEX" THEN 23
0
1910 LET ANS=VAL(ANS$)
1920 IF ANS=000 THEN 2400
1930 IF ANS<FIRST/TABLE THE
N 1960
1940 GOSUB 2000
1950 GOTO 1700
1960 GOSUB 1900
1970 GOTO 1830
1980 REM WRONG ROUTINE.
1990 CALL SOUND(250,110,2,-6
,2)
2000 GOSUB 2260
2010 PRINT TAB(6);"YOU ARE W
RONG, ";NAME$
2020 PRINT
2030 PRINT
2040 PRINT TAB(6);"TRY THAT
ONE AGAIN"
2050 PRINT
2060 PRINT
2070 RETURN
2080 REM CONGRATULATION
SUBROUTINE
2090 CALL CLEAR
2100 PRINT TAB(8);"THAT'S CO
RECT ",TAB(12);NAME$;,"TA
B(8);"CONGRATULATIONS"
2110 CALL SOUND(100,440,2)
2120 PRINT
2130 CALL SOUND(100,550,2)
2140 PRINT
```

```
2150 CALL SOUND(100,440,2)
2160 PRINT
2170 CALL SOUND(100,660,2)
2180 PRINT
2190 CALL SOUND(100,880,2)
2200 PRINT
2210 FOR I=3 TO 16
2220 CALL SCREEN(1)
2230 NEXT I
2240 CALL CLEAR
2250 RETURN
2260 REM SCROLLING ROUTINE
2270 FOR I=1 TO 8
2280 PRINT
2290 NEXT I
2300 RETURN
2310 REM DELAY ROUTINE
2320 FOR DELAY=1 TO 50
2330 NEXT DELAY
2340 RETURN
2350 REM SWAP ROUTINE
2360 HOLD=FIRST
2370 FIRST=SECOND
2380 SECOND=FIRST
2390 GOTO 1470
2400 CALL CLEAR
2410 PRINT TAB(3);"HOPE YOU
ENJOYED YOURSELF"
2420 PRINT
2430 PRINT
2440 PRINT TAB(12);NAME$
2450 PRINT
2460 PRINT
2470 PRINT TAB(11);"GOOD BYE
"
2480 PRINT
2490 PRINT
2500 PRINT
2510 END
#####
!
! Conversion routine to
! change any lower case
! character into it's
! upper case equivalent.
!
! Can be stored in the
! MERGE format and used
! as a subroutine.
!
#####
100 ! CONVERSION ROUTINE
110 NM$=""
120 FOR I=1 TO LEN(A$)
130 CH$=SEG$(A$,I,1)
140 CH=ASC(CH$)
150 IF (CH<123)AND(CH>96)THE
N 160 ELSE 170
160 CH=CH-32
170 NM$=NM$&CHR$(CH)
180 NEXT I
190 PRINT NM$
200 A$=""
210 RETURN
```

DATA TRANSFER

faintest idea which were necessary, which weren't, whether the defaults would do, and so on. In the end, I set the ones I knew, and used the defaults given for a Hayes 388 modes.

Add to that the fact that I was unsure about the cable connections, and you have all the ingredients for endless frustration. Fortunately, I had corresponded with TI in the US on another matter, and decided that seeing they had been so helpful in that case, and as they had designed both computers, they ought to be able to give me the correct wiring for the cable. Accordingly, I wrote again to them. A long time had passed, and I had nearly given up. During this time, nothing I had tried had worked, when out of the blue, came a letter from TI with the required information.

Joyously, I soldered up the cable, and tried it out. Well, I did get a line of (PE)<XF> and so on, which may have meant something, but TELL stopped dead after that. I nearly cried, for by his time, I had found a perfectly wonderful word processor called "Word Perfect", version 4.1, which must have been designed with thesis writers in mind.

A quick diversion here is in order, for there may be readers who are considering word processors. Word Perfect has a lot of wonderful features, but a few are:

- a) The ability to enter footnotes and endnotes into the text, and to have them printed out at the bottom of the page, or end of the article, (as the case may be), automatically numbered;
- b) The ability to automatically generate an index and a title page of chapter headings;
- c) On-screen columns of text, changeable at will - if three columns doesn't look right, change to two!;
- d) A spelling checker that takes wild-card characters and phonetic spelling;
- e) A thesaurus;
- f) Mathematical capability;
- g) The ability to create a database, and selectively choose parameters for reports;
- h) Many variations on the header/footer theme;
- i) The ability to work on two files concurrently, and see them both on-screen, and be able to cut and paste between them;
- j) The "flush right" key - e.g. type in the date at the head of a letter, on the left margin, press the "flush right" key, and hey presto, the date is now hard up against the right margin;
- k) The above is unnecessary, for the program will automatically enter the date, in any of several formats;

and much, much more.

So, I was now in the situation where I really wanted to transfer those files, and in desperation, I did what I should have done ages ago, and that was to ring Bernie Eisner. Bernie, ever helpful, suggested that the protocols of TELL may not be acceptable to Open Access, and opined that a program he well called "4A/TALK" by DataBiologics Inc. may be worth a try. He sent it to me and....

All I can say is ..HOORAY!!!! IT WORKS!!!!

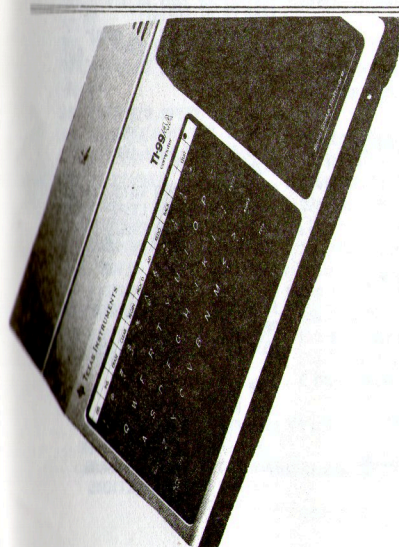
There has not been a more satisfying moment in my dealings with computers - for years, when that text appeared on the 4A screen, and then on the screen of the PRO. So that's how this article was sent to Bernie.

I also wish to acknowledge the assistance of Patrick Hicks, Consumer Relations, Technical Communications, Texas Instruments, from whose letter I quote below:

"The following pin assignments are designed to connect a TI 99/4A Home Computer to a TI Professional Computer via a null modem cable.

99/4A	TI Pro.
1-----	-----1
2-----	-----2
3-----	-----3
4-----	-----4
5-----	-----5
6-----	-----6
7-----	-----7
8-----	-----8
9-----	-----9
0-----	-----0

* (End of quote.)



PROGRAMMING MUSIC IN TI BASIC by TEXAS INSTRUMENTS

Programming music on the TI-99/4A with the CALL SOUND command can produce delightful results.

To get started, you'll need to know how to read notes and their lengths, have a copy of the music you want to reproduce and locate the musical tone frequencies chart on page III-7 of the "User's Reference Guide."

The CALL SOUND command line consists of Duration (length of the note), Frequency (numerical equivalent of the note) and Volume (loudness or softness of the note); 100 CALL SOUND (Duration, Frequency, Volume)

If you wanted to program more than one tone at the same time, the line would look like this:

100 CALL SOUND (Duration, Frequency, Volume, Frequency2, Volume2, Frequency3, Volume3)

Whenever you program two or more notes together, the duration remains the same and is listed only once in the line.

The duration of a note depends on the time signature of the music. The signature tells you how many beats are in each measure and is indicated at the beginning of the music. For your first program, you may want to start with music written in a 4/4 signature (four beats to the measure).

To determine the duration, first choose the length of each measure. For example, if you choose 960 (a fast tempo) as the length of each measure, a quarter note would be 240 (1/4 of 960), a half note would be 480 (1/2 of 960) and a dotted half note would be 720 (3/4 of 960).

The frequency is the actual note. Frequencies for notes are listed in the "User's Reference Guide." The frequency for middle C, for example, is 262. The frequency for high C is 523.

Volumes range from 0 (loudest) to 10 (most quiet).

If you programmed a line to play middle and high C together as half notes at a fairly loud volume, the line would look like this:

100 CALL SOUND (480,262,1,523,1)

A program for "Deck the Halls" follows. Remember, you can make the music play faster or slower by changing the duration of each measure and of each note.

100 CALL SOUND (360,523,1,440,1)
200 CALL SOUND (120,466,1,392,1)
300 CALL SOUND (240,440,1,349,1)
400 CALL SOUND (240,392,1,330,1)
500 CALL SOUND (240,349,1,294,1)
600 CALL SOUND (240,392,1,330,1)
700 CALL SOUND (240,440,1,349,1)
800 CALL SOUND (240,349,1,262,1)
900 CALL SOUND (120,392,1,330,1)
1000 CALL SOUND (120,440,1,349,1)
1100 CALL SOUND (120,466,1,392,1)
1200 CALL SOUND (120,392,1,330,1)
1300 CALL SOUND (360,440,1,349,1)
1400 CALL SOUND (120,392,1,294,1)
1500 CALL SOUND (240,349,1,262,1)
1600 CALL SOUND (240,330,1,262,1)
1700 CALL SOUND (480,349,1,262,1)
1800 CALL SOUND (360,523,1,440,1)
1900 CALL SOUND (120,466,1,392,1)
2000 CALL SOUND (240,440,1,349,1)
2100 CALL SOUND (240,392,1,330,1)
2200 CALL SOUND (240,349,1,294,1)
2300 CALL SOUND (240,392,1,330,1)
2400 CALL SOUND (240,440,1,349,1)
2500 CALL SOUND (240,349,1,262,1)
2600 CALL SOUND (120,392,1,330,1)
2700 CALL SOUND (120,440,1,349,1)
2800 CALL SOUND (120,466,1,392,1)
2900 CALL SOUND (120,392,1,330,1)
3000 CALL SOUND (360,440,1,349,1)
3100 CALL SOUND (120,392,1,294,1)
3200 CALL SOUND (240,349,1,262,1)
3300 CALL SOUND (240,330,1,262,1)
3400 CALL SOUND (480,349,1,262,1)
3500 CALL SOUND (360,392,1,330,1)
3600 CALL SOUND (120,440,1,349,1)
3700 CALL SOUND (240,466,1,392,1)
3800 CALL SOUND (240,392,1,330,1)
3900 CALL SOUND (360,440,1,349,1)
4000 CALL SOUND (120,466,1,392,1)
4100 CALL SOUND (240,523,1,440,1)
4200 CALL SOUND (240,392,1)
4300 CALL SOUND (120,440,1,349,1)
4400 CALL SOUND (120,494,1,392,1)
4500 CALL SOUND (240,523,1,392,1)
4600 CALL SOUND (120,587,1,392,1)
4700 CALL SOUND (120,659,1,392,1)
4800 CALL SOUND (240,698,1,440,1)
4900 CALL SOUND (240,659,1,392,1)
5000 CALL SOUND (240,587,1,349,1)
5100 CALL SOUND (480,523,1,330,1)
5200 CALL SOUND (360,523,1,440,1)
5300 CALL SOUND (120,466,1,392,1)
5400 CALL SOUND (240,440,1,349,1)
5500 CALL SOUND (240,392,1,330,1)
5600 CALL SOUND (240,349,1,294,1)
5700 CALL SOUND (240,392,1,330,1)
5800 CALL SOUND (240,440,1,349,1)
5900 CALL SOUND (240,349,1,262,1)
6000 CALL SOUND (120,587,1,349,1)
6100 CALL SOUND (120,587,1,349,1)
6200 CALL SOUND (120,587,1,349,1)
6300 CALL SOUND (120,587,1,349,1)
6400 CALL SOUND (360,523,1,330,1)
6500 CALL SOUND (120,466,1,392,1)
6600 CALL SOUND (240,440,1,349,1)
6700 CALL SOUND (240,392,1,330,1)
6800 CALL SOUND (480,349,1,294,1)
6900 END

 * The competition winning
 * programs. This one is
 * by William Edmond, and
 * won the first prize.
 #####



```

100 REM ** MOTORBIKE **
110 REM ** DEALING **
120 REM ** BY **
130 REM ** WILLIAM **
140 REM ** EDMOND **
150 REM ** T.I.U.P. **
160 REM ** DATE: **
170 REM ** 15/4/86 **
180 REM ** COMPETION **
190 REM **READ DATA**
200 DIM MODEL$(7),PRICE(7)
210 FOR ELEMENT=1 TO 7
220 READ MODEL$(ELEMENT),PRIC
E(ELEMENT)
230 NEXT ELEMENT
240 CALL CLEAR
250 CALL SCREEN(14)
260 PRINT TAB(7);"MOTOR BIKE
"
270 PRINT TAB(7);" DATABASE
"
280 PRINT
290 PRINT TAB(11);"BY"
300 PRINT TAB(8);"WILLIAM"
310 PRINT TAB(9);"EDMOND"
320 FOR INTERVAL=1 TO 9
330 PRINT
340 NEXT INTERVAL
350 CALL SOUND(2000,-4,5)
360 CALL SOUND(3000,-4,5)
370 CALL SOUND(1700,-8,5)
380 FOR TIME=1 TO 900
390 NEXT TIME
400 CALL CLEAR
410 CALL SCREEN(13)
420 PRINT TAB(7);"SELECTION
LIST"
430 PRINT TAB(7);"=====
===="
440 PRINT TAB(6);"(1)....INS
TRUCTIONS"
450 PRINT TAB(6);"(2)....SEE
ALL BIKES"
460 PRINT TAB(6);"(3)....SEE
THE PRICE OF"
470 PRINT TAB(4);"A CERTAIN
MODEL"
480 PRINT TAB(6);"(4)....SEE
MODELS BELOW"
490 PRINT TAB(4);"A CERTAIN
PRICE"
500 PRINT TAB(6);"(5)....END
"
510 FOR SPACE=1 TO 3
520 PRINT
530 NEXT SPACE
540 CALL SOUND(100,440,6)
550 PRINT "YOUR CHOICE : "
    
```

```

560 CALL KEY(0,K,5)
570 IF S=0 THEN 560
580 IF K=49 THEN 650
590 IF K=50 THEN 860
600 IF K=51 THEN 950
610 IF K=52 THEN 1160
620 IF K=53 THEN 1540
630 GOTO 560
640 REM
650 REM **INSTRUCTIONS**
660 CALL CLEAR
670 CALL SCREEN(4)
680 PRINT TAB(6);"INSTRUCTIO
NS"
690 PRINT TAB(6);"=====
===="
700 PRINT "THIS PROGRAM WILL
PROVIDE "
710 PRINT "YOU WITH ALL THE
RELEVANT"
720 PRINT "INFORMATION DURIN
G THE"
730 PRINT "PROGRAM."
740 PRINT "YOU WILL NEED TO
ENTER"
750 PRINT "THE NAMES OF MODE
LS OR "
760 PRINT "PRICES ALONG THE
WAY"
770 PRINT "AND THIS INFORMAT
ION CAN BE"
780 PRINT "GOT BY PRESSING -
2- IN THE"
790 PRINT "MAIN SELECTION LI
ST."
800 PRINT
810 PRINT
820 PRINT TAB(4);"PRESS ENTE
R TO CONTINUE"
830 CALL KEY(0,K,5)
840 IF S=0 THEN 830
850 GOTO 400
860 REM **ALL BIKES**
870 CALL CLEAR
880 CALL SCREEN(12)
890 PRINT TAB(4);"MODEL
PRICE"
900 PRINT
910 FOR INDEX=1 TO 7
920 PRINT MODEL$(INDEX),PRIC
E(INDEX)
930 NEXT INDEX
940 GOTO 1360
950 REM **MODEL'S PRICE**
960 CALL CLEAR
970 CALL SCREEN(10)
980 INPUT "MODEL OF BIKE *:B
IKEMODEL$
990 N=0
1000 M=0
1010 FOR N=1 TO 7
1020 IF MODEL$(N)=BIKEMODEL$
THEN 1110
1030 IF MODEL$(N)=BIKEMODEL$
THEN 1130
1040 IF MODEL$(N)=BIKEMODEL$
THEN 1150
1050 IF M<>0 THEN 1150
1060 NEXT N
    
```

```

1070 PRINT
1080 PRINT "MODEL UNLISTED"
1090 GOTO 1150
1100 PRINT
1110 PRINT MODEL$(N),PRICE(M
)
1120 GOTO 1030
1130 M=1
1140 GOTO 1050
1150 GOTO 1360
1160 REM **BELOW A CERTAIN
PRICE **
1170 CALL CLEAR
1180 CALL SCREEN(16)
1190 INPUT "WHAT IS THE SEAR
CH PRICE *-PRICETO FIND
1200 Y=0
1210 PRINT
1220 FOR N=1 TO 7
1230 IF PRICE(N)<PRICETO FIND
THEN 1280
1240 IF PRICE(N)<PRICETO FIND
THEN 1300
1250 NEXT N
1260 IF Y=0 THEN 1330
1270 GOTO 1360
1280 PRINT MODEL$(N),PRICE(N
)
1290 GOTO 1240
1300 Y=1
1310 GOTO 1250
1320 GOTO 1360
1330 PRINT "NO BIKES UNDER T
HAT PRICE"
1340 GOTO 1320
1350 END
1360 REM **SELECTIONS2**
1370 PRINT
1380 PRINT
1390 PRINT TAB(6);"SELECTION
S : "
1400 PRINT TAB(6);" 1)...
.RETURN TO MAIN MENU"
1410 PRINT TAB(5);"2)....END
"
1420 PRINT
1430 CALL KEY(0,K,5)
1440 IF S=0 THEN 1430
1450 IF K=49 THEN 400
1460 END
1470 DATA "HONDA CB250",1750
1480 DATA "HONDA CB360",1820
1490 DATA "HONDA CB500",2340
1500 DATA "HONDA CB750",2560
1510 DATA "SUZUKI 6L900",265
0
1520 DATA "SUZUKI 6L1000",27
80
1530 DATA "SUZUKI 6L1100",29
40
1540 END
1550 REM **END**
1560 REM IF MORE BIKES ARE
1570 REM TO BE ADDED,REFER
1580 REM TO USER'S MANUAL
1590 REM FOR INSTRUCTIONS
1600 REM **BYE**
    
```

TRAFFIC COP

BY T.I.U.P., THE TI-99/4A HOME COMPUTER USERS GROUP OF PERTH,

```

70 CALL CLEAR
80 REM *****
90 REM *
100 REM * TRAFFIC COP *
110 REM *
120 REM *****
130 PRINT "TRAFFIC COP *"
140 REM
150 CALL MAGNIFY(3)
160 CALL GRAPHICS
170 CALL INSTRUCTIONS
180 CALL CLEAR
190 SP=0 :: CR=0
200 CALL SETUP(SP)
210 CALL CARS(SP)
220 CALL CDINC(ALL,H):: IF H
THEN CALL CRASH(CR)ELSE 250
230 CALL SOUND(-100,1000,5)
240 IF CR>9 THEN 320 ELSE 21
0
250 CALL KEY(3,K,5)
260 CT=CT+1 :: IF CT>75 THEN
CALL FASTER(CT,SP,BST)
270 IF K=83 THEN CALL MOTION
(0,0)ELSE CALL MOTION(0,1
,0,SP)
280 IF K=68 THEN CALL MOTION
(0,0)ELSE CALL MOTION(0,2
,0,SP)
290 IF K=69 THEN CALL MOTION
(0,0)ELSE CALL MOTION(0,3
,0,SP)
300 IF K=88 THEN CALL MOTION
(0,0)ELSE CALL MOTION(0,4
,-SP,0)
310 GOTO 220
320 CALL CLEAR
330 DISPLAY AT(6,1);"YOU CAU
SED 10 CRASHES, WITH"
340 DISPLAY AT(8,1);"THE TRA
FFIC RUNNING AT"
350 DISPLAY AT(10,1);SP;"MPH
"
360 DISPAY AT(12,2);"BEST SP
LED SO FAR IS";BST;"MPH"
370 DISPLAY AT(16,6);BEEP;"PL
AY AGAIN? (Y/N)"
380 CALL KEY(3,K,5):: IF S=0
THEN 380
390 IF K=78 THEN END
400 IF K<>89 THEN 370
410 CALL CLEAR :: GOTO 170
420 SUB GRAPHICS
430 CALL SCREEN(2)
440 CALL CHAR(128,"FF101010
101010FF")
450 CALL CHAR(129,"818181FFB
1818181")
460 CALL CHAR(130,"010101010
10204FB")
470 CALL CHAR(131,"808080808
040201F")
480 CALL CHAR(132,"1F2040808
0808080")
490 CALL CHAR(133,"FB0402010
1010101")
500 CALL CHAR(136,"000000FFF
F")
510 CALL CHAR(137,"181818181
8181818")
520 CALL CHAR(136,"00000000FF
FFF3F3F3F3FF7F00000000000000
00FFFEFF1F1F1FFFEFF")
530 CALL CHAR(100,"00000000FF
7FFFBFBFBFB7FF000000000000000
0000FFFCFCFCFCFFFEFF")
540 CALL CHAR(104,"080F0F0F0
F0C0C0C0F0C0C0F0F0F0F0F0F0F0
F0F0303030F0F03030F0F0F0E0"
)
550 CALL CHAR(108,"070F0F0F0
C0C0F0C0C0C0F0F0F0F0E0F0F0
F0F03030F0F03030F0F0F0F0D0"
)
560 CALL CHAR(112,"014163333
F31F1F1F1F3F7B6383030100818
3C6EEFCFCFCFBFBFC9E0701")
570 CALL COLOR(13,16,2,14,16
,1)
580 FOR I=2 TO 8
590 CALL COLOR(I,16,1)
600 NEXT I
610 SUBEND
620 SUB INSTRUCTIONS
630 DISPLAY AT(10,8);"TRAFFI
C COP."
640 DISPLAY AT(13,2);"DO YOU
WANT INSTRUCTIONS?"
650 DISPLAY AT(15,11);BEEP;"
(Y/N)"
660 CALL KEY(3,K,5):: IF S=0
THEN 660
670 IF K=78 THEN CALL CLEAR
:: GOTO 750
680 IF K<>89 THEN 650
690 CALL CLEAR
700 DISPLAY AT(4,2);"USE THE
ARROW KEYS (ESD,X)"
710 DISPLAY AT(6,3);"TO STOP
THE TRAFFIC AND"
720 DISPLAY AT(8,1);"PREVENT
COLLISIONS. WHEN TEM"
730 DISPLAY AT(10,3);"CRASHE
S HAVE OCCURED YOU"
740 DISPLAY AT(12,2);"WILL B
E REMOVED FROM DUTY."
750 DISPLAY AT(18,4);"PRESS
ANY KEY TO PLAY."
760 CALL KEY(3,K,5):: IF S=0
THEN 760
770 SUBEND
780 SUB SETUP(SP)
790 DISPLAY AT(4,1);"TRAFFIC
DISPLAY AT(6,3);"COP"
810 DISPLAY AT(4,21);"SPEED"
820 DISPLAY AT(6,20);SP;"MPH
"
830 DISPLAY AT(20,1);"TI-99/
4A"
840 DISPLAY AT(20,21);"CRASH
ES"
850 CALL HCHAR(9,1,128,32)
860 CALL HCHAR(15,1,128,32)
870 CALL VCHAR(1,13,129,24)
880 CALL VCHAR(1,19,129,24)
890 CALL VCHAR(1,14,32,120)
900 CALL HCHAR(10,1,32,160)
910 CALL HCHAR(9,13,130)
920 CALL HCHAR(9,19,131)
930 CALL HCHAR(15,13,133)
940 CALL HCHAR(15,19,132)
950 FOR I=2 TO 30 STEP 3
960 CALL HCHAR(12,1,136,2)
970 NEXT I
980 FOR I=1 TO 22 STEP 3
990 CALL VCHAR(1,16,137,2)
1000 NEXT I
1010 SUBEND
1020 SUB CARS(SP)
1030 IF SP>20 THEN SP=SP-4
1040 DISPLAY AT(6,20);SP;"MP
H"
1050 CALL SPRITE(01,96,14,75
,1,0,SP)
1060 CALL SPRITE(02,100,16,9
,8,250,0,-SP)
1070 CALL SPRITE(03,108,6,1
,128,SP,0)
1080 CALL SPRITE(04,104,3,19
,0,105,-SP,0)
1090 SUBEND
1100 SUB CRASH(CR)
1110 CALL MOTION(01,0,0,02,0
,0,83,0,0,04,0,0)
1120 CALL SPRITE(09,112,9,88
,120)
1130 CALL SPRITE(010,112,12,
88,120,5,5,011,112,12,88,120
,-5,-5,012,112,12,88,120,5,-
5,013,112,12,88,120,-5,5)
1140 FOR I=0 TO 20
1160 CALL SOUND(-200,-7,1,11
0,145)
1170 CALL COLOR(09,12)
1180 NEXT I
1190 CALL DELSPRITE(ALL)
1200 CR=CR+1
1210 DISPLAY AT(22,23);CR
1220 SUBEND
1230 SUB FASTER(CT,SP,BST)
1240 CALL SOUND(200,-2,0,500
,5)
1250 SP=SP+2
1260 IF SP>BST THEN BST=SP
1270 CT=0
1280 DISPLAY AT(6,20);SP;"MP
H"
1290 SUBEND
    
```

LETTERS TO THE EDITOR MAY BE

ADDRESSED TO :- T.I.U.P.

PO BOX 246, MOUNT LAWLEY

WESTERN AUSTRALIA 6050.



 * ARCADE HARDWARE *
 * 211, HORTON ROAD *
 * FALLOWFIELD *
 * MANCHESTER *
 * M14 7QE *
 * 061-225-2248 *

Around the middle of June 1987, Arcade Hardware will cease to exist. After June, I will be permanently resident outside the United Kingdom and so it will be impossible to carry on trading. Customers should note the following. Not all items are new. Not all new items are boxed, although wherever possible, manuals will be included with non boxed stock.

Where an item is marked HG, cheques should be made payable to HOWARD GREENBERG and not Arcade Hardware. Access cannot be accepted on these items.

MODULE LIST	CURRENT	From 18 April	From 20 May
MIDNIGHT MASON	£ 5.00	£ 3.00	£ 2.00
MUNCHMAN	£ 5.00	£ 3.00	£ 2.00
TOMBSTONE CITY	£ 5.00	£ 3.00	£ 2.00
MINUS MISSION	£ 5.00	£ 3.00	£ 2.00
EARLY LEARNING FUN	£ 5.00	£ 3.00	£ 2.00
NUMBER MAGIC	£ 5.00	£ 3.00	£ 2.00
NUMBER READINESS	£ 5.00	£ 3.00	£ 2.00
FROG JUMP	£ 5.00	£ 3.00	£ 2.00
STAR MAZE	£ 5.00	£ 3.00	£ 2.00
EXTENDED BASIC	£49.95	£35.00	£20.00
MINI MEMORY (used)	£35.00	£35.00	£20.00
EDITOR/ASSEMBLER (used)		£35.00	£20.00 HG
MULTIPLAN	£25.00	£25.00	£20.00 HG
TI-WRITER (used)			£30.00 HG
CONNECT FOUR	£ 5.00	£ 3.00	£ 2.00
FATHOM	£ 5.00	£ 3.00	£ 2.00
TREASURE ISLAND	£10.00	£ 7.50	£ 5.00
EXTENDED BASIC no manual	£20.00	£20.00	£10.00
RETURN TO PIRATE ISLE	£10.00	£ 7.50	£ 5.00
THE ATTACK (used)	£ 5.00	£ 3.00	£ 2.00
EARLY READING no manual	£ 5.00	£ 3.00	£ 2.00
SLYMOIDS	£10.00	£10.00	£10.00
KING OF THE CASTLE	£ 5.00	£ 3.00	£ 2.00
HONEY HUNT	£ 5.00	£ 3.00	£ 2.00
CHISHOLM TRAIL	£ 5.00	£ 3.00	£ 2.00
TI-INVADERS	£ 5.00	£ 3.00	£ 2.00

PUBLICATIONS

STEVE DAVIS PROGRAMS FOR TI HOME COMPUTER	£14.95	£10.00	£ 7.50
101 PROGRAMMING TIPS	£ 4.00	£ 3.00	£ 2.00
GET PERSONAL WITH TI99/4A	£ 5.00	£ 4.00	£ 3.00
FAVOURITE PROGRAMS EXPLAINED	£5.00	£ 4.00	£ 3.00
SMART PROGRAMMERS GUIDE TO SPRITES	£ 6.95	£ 5.00	£ 3.00
THE ORPHAN CHRONICLES	£10.00	£ 7.50	£ 5.00

PRODUCT LIST	CURRENT	18 April	From 20 May
ADVENTURE TAPES (to suit Adventure module)			
MYSTERY FUN HOUSE	£ 3.50	£ 3.00	£ 2.00
VOODOO CASTLE	£ 3.50	£ 3.00	£ 2.00
THE COUNT	£ 3.50	£ 3.00	£ 2.00
GHOST TOWN	£ 3.50	£ 3.00	£ 2.00
STRANGE ODYSSEY	£ 3.50	£ 3.00	£ 2.00

INFOCOM ADVENTURES (requires disc + 32k)			
ZORK II	£25.00	£18.00	£10.00
ZORK III	£25.00	£18.00	£10.00
SORCERER	£25.00	£18.00	£10.00
DEADLINE	£25.00	£18.00	£10.00
WITNESS	£25.00	£18.00	£10.00
INFIDEL	£25.00	£18.00	£10.00
SUSPENDED	£25.00	£18.00	£10.00
STARCROSS	£25.00	£18.00	£10.00
HITCH HIKERS GUIDE TO GALAXY	£25.00	£18.00	£10.00

MILLERS SOFTWARE			
ADVANCED DIAGNOSTICS	£19.95	£12.50	£10.00
EXPLORER	£24.95	£18.00	£10.00
DISKASSEMBLER	£19.95	£12.50	£10.00

HARDWARE			
MYARC RS232 card	£125.00	£85.00	£50.00
MYARC Disc Control Card	£185.00	£100.00	£75.00
MYARC 128k card	£230.00	£150.00	£100.00

1 only complete peripheral system comprising box, 32k card, disc controller + 1 internal drive. £250.00 HG.

1 only Shinwa CPA 80+ printer			£175.00 HG.
1 only Juki 2200 daisywheel printer			£175.00 HG.
AXIOM parallel interface	£ 75.00	£ 50.00	£ 35.00
CASSETTE CABLE	£ 7.50	£ 5.00	£ 3.50
CARTRIDGE EXPANDER (widgit)	£35.00	£ 28.00	£ 20.00
CASSETTE N GAME FILE	£ 22.50	£ 18.00	£ 10.00
holds 18 modules/tapes			
SPEECH SYNTHESISER	£ 35.00	£ 25.00	£ 20.00
SINGLE SIDED DISC DRIVE	£ 75.00	£ 75.00	£ 75.00 s/h
DOUBLE SIDED DISC DRIVE	£ 85.00	£ 85.00	£ 85.00 s/h
ALPHACOM 42 thermal printer	£65.00	£ 65.00	£ 65.00 s/h

DISC STORAGE BOXES	£ 3.00	£ 3.00	£ 3.00
--------------------	--------	--------	--------

NEW FROM MONTY SCHMIDT:

GPL LINKER V1.1 RunTime Version

now \$49.95
w/Linker \$59.95
plus Intern \$69.95
add \$3 shipping

GPL Linker is an ingenious program that places the power of Graphics Language Programming (GPL) at your command. No extra hardware is required beyond standard 32k and disk system. In short, Linker creates runnable program files from compressed (or un-compressed) GPL Assembler object files. You can then run these programs with "Option 5 Run Program Files" of the Editor Assembler Module.

Up to 24k GPL programs can be developed and run on standard 32k systems. Included in the run time version are two demonstration programs and "CONVERT," a public domain conversion program that converts MS BASIC statements to TI BASIC statements. Price: \$21.00 CDN funds \$15.00 US funds.

ENHANCED GPL Assembler V2.1

NOW with high memory loader package

UNLOCK ALL THE SECRETS! New GPL Assembler Version 2.1 available exclusively through RYTE Data.

This program provides the power to write, edit and assemble true GPL programs for the TI 99/4A. Create code that accesses console operating system routines directly. Develop programs that use the GPL Interpreter and all the features of the TI 99/4A.

This package includes the GPL Assembler disk, printed documentation, GPL tips and hints, update support service and commented GROM/ROM listings (with the book "INTERN"). An example for a command module type GPL program is included with source, object and list files on disk.

Requires: 32k memory, disk drive(s), TI Editor Assembler package. Printer/RS-232 recommended.

GPL price Reduction

R/D COMPUTING Technical Newsletter

with Bill Gronos on assembly!

We have a vision. Our vision is one of continued TI 99/4A support. We're dedicated to the power of the machine. From the novice to the experienced computer user; for management, home, education, entertainment or advanced applications our publication "R/D COMPUTING" is for you. TI never revealed all the important inner workings of the 99/4A. We bring you this vital information every month.

A major feature of R/D COMPUTING is the regular "upgrade projects." These electronic construction projects are designed to give the 99/4A owners more features and improvements. For example, it is possible to increase the speed of your computer with a very simple part and switch. Each month we present new circuits, diagrams and projects for your computer.

From the moment your new subscription arrives at your home, you will have access to critical technical information that makes your computer more valuable, powerful and versatile.

We believe that the TI 99/4A deserves new products, innovative hardware, software, information and a dedicated technical publication. This is what makes a computer "viable" in the fast paced microcomputer industry. Now that the 99/4A has been 'opened up,' all the secret information is available. You can have all these benefits and more each month. **SUBSCRIBE NOW!**

\$14/year - back issues 3-21 available

THANK YOU! Our business has grown 300% this year. To show our appreciation we are giving away hundreds of dollars in TI products to 99/4A owners or users groups. To enter drawing, (no purchase necessary) send your name and address on a postcard to RYTE Data. For subscribers to R/D Computing we are giving away XBII plus, 32k memories, GPL Assembler package, etc. **Enter your subscription today!**

Prices listed in U.S. funds.

New catalogue available.



Designed for the CorComp Clock Peripheral—Triple Tech Card or Stand-alone models. This utility package provides more functions for use in your Extended Basic programs. Direct access to the clock ROM at assembly speed gives you these features: three independent timers to set and read; alarm function; two interrupt routines to display time and date on screen with CTRL T—continuously or on your

command; all time and date displays are in 12 or 24 hour format using TEXT. This program also allows the week, date and time to be set independently rather than all together.

Program disk is not copy protected to allow you full use in your Extended Basic programs. Package includes disk and instructions. Only \$17.95 plus \$2

XBII plus

As reviewed in Micropendium October 1985. This command module gives you all the features of Extended Basic PLUS 40 new commands.

Totally compatible with TI's XB, this enhanced version gives your programs more power to access your 99/4A. Commands such as MLOAD, MSAVE, VPEEK, VPOKE, GPEEK are superior to most other Basic environments. Various demo programs and new applications using high resolution graphics make this module a "must" for Extended Basic users. Comes complete with a 95 page manual. Requires console and 32k. \$75.00 (US) plus \$2 shipping.

BASIC V1.1 Compiler

New Basic Compiler that is finally easy to use! Supports virtually all Basic and Extended Basic commands in

existing programs. Simply load and compile programs from a menu driven directory on your screen. No extensive re-writing, variable declarations or conversions are required. Compiler produces code-list in one pass containing all variable addresses and jump list. Package includes Extended Basic Loader, Floating Point Loader, Integer Loader, Disk Menu program and DSR program for the Compiler support. This Compiler cannot unravel DEF statements and stops on the END statement—no SUB's allowed. TRACE, BREAK, ON ERROR, CALL LOAD and CALL LINK may produce execution errors. Requires 32k, disk. Price: \$20.00 plus \$2 shipping (US funds).

Ryte Data

(705) 457-2774



MILLENNIUM COMPUTERS

210 MOUNTAIN STREET,
HALIBURTON, ONTARIO K0M 1S0
TELEX 06-986766 TOR. ATTN: RYTE DA