tidings

TIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIding
ngsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIc

TIHOME

tidings

TIHOME is not affiliated with Texas Instruments in
any way and is supported only by its subscribers

The opinions expressed in TIdings are not always those of TIHOME

TIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdi
dingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdinc

Contents

ngsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsT
sTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIdingsTIc

LANTERN

BATTLESTAR ATTACK   £10.   Ex-Basic & Joysticks

The view is through the cockpit window of your space fighter as you fly in low over an enemy battlestar. In the distance your home planet looks small and vulnerable as it slowly appears from behind a large red moon. Your task is to sink five photon torpedoes into the exhaust ports of the battlestar before it has a chance to annihilate your planet, whilst at the same time defending yourself against the mines homing in on your ship!
Four separate levels and hi-scores, 3D graphics, exciting effects and arcade style speed make this one of the best Ex-Bas games around!!!!

OPERATION MOON   £10.   Ex-Basic

You have been detailed to shuttle supplies and spares between special pads on the surface of the moon and a satellite workstation. In doing this you must also deal with a fiery comet and a host of banshee space bats which attempt to clamp themselves to your ship and thus prevent you from docking. You can defend yourself via the satellite but if one should land, beware, removing him can be tricky! Throughout the game you are using up supplies and may even die of starvation if you are not skilful enough!
Humour and frustration are nicely balanced in this excellent game as well as good graphics and ten levels of play.

SCROLMASTER   £10.   Mini-Memory

Scrolmaster is a powerful machine code utility program which adds five important new functions to support the standard console basic. These functions can be called up quite simply from a TI-Basic program via the 'Call Link' command.
Scrolmaster allows you to scroll the whole screen left, right, and down (print will scroll upwards), operate a split screen from any line you choose (except lines 1 and 24), and print a string of up to 32 characters at any row on the screen, instantly!
Whether used for display or games programs you can now scroll in four directions, use only a portion of the screen (e.g. as in Adventures) and, by manipulating spaces in strings, 'Print At' !

Send S.A.E. for a complete list of all our software.
Have you written a program?   Lantern Software offer good rates on software for the TI 99/4A.

LANTERN
SOFTWARE

LANTERN SOFTWARE 4, HAFFENDEN RD. TENTERDEN, KENT TN30 6QD.

EDITORIAL

A few badly collated copies of Vol. 3 No. 1 went out to members at the end of March, so if you were one of the unlucky recipients please return your copy and I will replace it for you.

A year's subscription for the 99'er costs £32 not £31. My mistake, sorry about that.

You will find some new contact addresses in the Close Routine. Check and see if there is one near you. Get in touch and see if the problem you are agonizing over was solved last week by someone just down the road.

Membership at the moment is just under 2000 and rising. I expect it to rise faster now that TI have started their special offers. And did I actually see a TV advert recently? Perhaps I have been having illusions — again.

Vol 3 No. 1 was such a large size because it was so late. I think it was too large, and future issues will probably be a little smaller, if only to save postage. It breaks my heart to pay £750 for stamps to send out all those copies.

I shall shortly be getting round to catching up with my correspondence, so if you have written to me and not had a reply, do not worry. This major development is possible because I now have another helper who deals with the copying and print-ing of all the programs. The computer operation, in other words. This leaves me free for other things.

A word to overseas members. Please do not send any currency except Sterling when paying for TIHOME services. My bank charge a punitive rate for converting foreign currency.

I have had a number of members phone me and report errors in 99/4A programs published in the commercial magazines. The first step is to contact the magazine. They may have the answer or put you in touch with the author. If you work out the correction, send it to me and I will publish it for the benefit of the rest of the members.

I am going up to Bedford shortly and I hope to see some of the new goodies that are on offer from TI. I will report on them in the next issue of Tidings.

I have just had the loft of my house turned into an office, so TIHOME now has a home, and I have my lounge back. This makes life a lot easier. We had to have the phone extended to the office so BT came and installed plug in phones. Two words of warning, sometimes the phones are left unplugged, so if you can't get through, persist. And the other thing is that if your Old English Sheepdog eats the plug off the end of the phone cable, it costs £19 for a new one. Cable, that is, not an Old English Sheepdog.

Good computing,

Paul Dicks.

TEXAS TI 99/4A JOYSTICK ADAPTERS

For a long time now the main problem of the marvellous Texas Instruments, TI 99/4A computer has been the joysticks which are compatable with it. We have consulted several TI joystick owners, and everyone of them without exception has been dissatisfied with their joysticks operation. Their main complaints were that the joysticks were sticky and insensitive and became difficult to handle when playing fast action games. Our own pair of joysticks were so hard to handle that we neglected them and reverted to using the TI keyboard, which was also difficult to handle.

From then on we decided to go about creating a new type of fully functional joystick which would be compatable with the TI. But unfortunately this was complicated and became too difficult to manufacture.

At this point we were comparing other computer joysticks to the TI joysticks. We found that there was a vast selection of joysticks for the Atari home computers. We then decided that it might be a good idea if we were to manufacture a small interface which would connect any of the numerous Atari joysticks to the TI. We came up with a joystick interface which connects the two pieces of hardware. With this interface the Atari joysticks are fully functional on the TI. This has put the pleasure back into fast action games on the TI. We have tested Master Point, Quick Shot, Le Stick and of course the traditional Atari joysticks, and they all function properly with the TI once the interface has been attached.

The item is selling for around £12 for the single joystick adapter and £16 for the deluxe double joysticks adapter. If you already own some Atari joysticks then these prices undercut the prices of the TI joysticks which sell for around £21.

For any further information please contact the address below, they will be happy to send you a data sheet on any additional information.

Write to: GRAHAM MICRO ACCESSORIES (joystick interface), 11-13 Long St., 3rd Floor, London E2 8HJ. Evening and weekend telephone enquiries: 04865-2887.

FOR SALE

A 99/4A FOR £99.4, EH?
ABSOLUTELY TRUE!
Details from P.N. WEBBER, 4 Firbank Road, St. Albans, Herts AL3 6LZ.
Telephone: 0727-35067.

* * * * * * * * * * * *

1.	Hunt the Wumpus (module), new	£10.00	
2.	Goblin Caves (cass.)	£ 1.50	
3.	3D-Maze (cass.)	£1.50	All 4 for £18 + P.& P.
4.	Teach yourself basic (double cass.)	£ 6.00	

Send to: D. Wheeler, 1 Guinness Trust, Loughborough Park, Brixton, London SW9.
Cheques made payable to: D. Heffernan.

```
*******************************************
CASSETTE SOFTWARE FOR THE TI99/4A (16K)
*******************************************
```

* <u>STARSHIP SUPERNOVA</u> (Text) (Adventure)

   Can you find the secret of the alien starship before it plunges
   into the Supernova?
   Includes a 'SAVE TAPE' facility.

* <u>ESCAPE THE MUGGER</u> (Graphics)

   Can you evade the mugger and get out of the room with the loot?
   An addictive game! Keeps a table of the ten highest scores and
   includes the option of key or joystick control.
   (A chase game similar to PACMAN).

* <u>NUCLEAR DISASTER</u> (Graphics)

   Can you save the scientists and seal the exits before the
   radiation guage gets to 100%. Highly frustrating! Several
   skill levels.

* <u>GOLF</u> (Graphics) 1 to 4 players. 1 to 18 Holes.

   A different course every time you play with a complete scorechart
   of each player at the end of the game.

* <u>MIND YOUR OWN BUSINESS</u> (Text) 1 to 4 players.

   Your in charge of a small business. Buy your stock in the
   morning and set your price for the best profits. (Don't be
   greedy!) Undercut your opponents and make a fortune.

* <u>OVERLORD</u> (Text)

   You are in charge of a Kingdom for a period of 10 years. You
   have peasants and land to grow food. Use your management skill
   to keep as many peasants alive as possible. Complete with
   natural disasters.

   ALL PROGRAMS £4.50 TO MEMBERS ONLY

   Send Cheque/PO to:-
                    P J Programs
                    231 Albert Road
                    JARROW
                    NE32 5RS

4

REVIEW

TI-WRITER/WORD PROCESSOR

The first thing I should mention before I whet too many appetites is that to use this new package you need the 4A console (the 4 won't work), the disc system, the 32k RAM and of course an 80 column printer. Rich Kids read on...

The first impression of this package is its similarity to the Editor/Assembler. It comes with a big 175 page manual, an SSS module and a disc. The latter contains the actual programs for the EDITOR and FORMATTER while the Module contains presumably the operating system. The manual is excellent in that unlike the Ed/Ass manual it is both a reference guide and a trainer, and practice texts are included on the disc.

The heart of the system is the EDITOR. This has many similarities with the Ed/Ass Editor, including the default screen colour of white on blue, although this can be changed. Like the Ed/Ass it uses a 3-window technique to display the full 80 columns on the 30 column screen. It has two modes: Command and Edit. In the Edit mode you carry out the usual file handling operations such as Load, Save and Print. If printing from the EDITOR you only get the format as typed in. To do all the clever things you use the FORMATTER. In the Command mode you also set Tabs and can Find, Move, Replace text and sundry other tricks. In the Edit mode you actually type in your text or correct it in various ways. The system uses a large number of FCTN and CTRL keys to access the editing and cursor control functions, which are too numerous to mention here. While in the Edit mode you have two Sub-modes to choose from: Wordwrap which is the default and Fixed. In the Wordwrap mode the Editor always puts as many words on a line as your margin tabs allow and reformats lines after insertions and deletions. You only use Carriage Return (ENTER) for paragraph endings. In the Fixed mode the text stays just as you have typed it. This is used for diagrams and other text where you don't want the format changed.

Once you have created a document you usually save it to disc since the FORMATTER can only operate on pre-stored files. (Incidentally the files are in DIS/VAR 80 format and thus compatible with the Ed/Ass).

The FORMATTER is perhaps badly named since it prints documents with numerous format commands, but these commands have to be inserted in the text with the EDITOR. There are too many format commands to list here but the important ones are:

- Justify the right margin
- Override margins set by the Editor.
- Fill/No fill which is like Fixed/Wordwrap in the Editor.
- Automatic line centring, underlining and double printing.
- Mailing list allowing text from a file to be inserted at preset points in the document.
- Automatic page heading or footing with Auto page numbering
- Page length control.

This has to be the best software package that TI have produced so far, but there are bound to be some bad points as well as good.

Some nice touches:

- The OOPS! command which restores a change or deletion if you hit the wrong key.
- The ability to alter the screen/text colours.
- An escape feature using CTRL U to transmit ASCII control codes 0-31 to the printer.
- The Transliterate command which allows a single character to be replaced by a sequence of ASCII codes when printed. Using this the longer Escape sequences of the printer can easily be handled. (Eg changing print font).

Some irritating features:

- Document files can be linked but not nested so standard texts cannot be inserted into documents.
- The FORMATTER always form feeds before starting to print. This achieves little except wasting paper.
- * The package can only work with a maximum of 80 columns, so even if you switch to condensed characters on the printer, you cannot use all the 132 columns. If you need 132 columns then you must go back to the comparatively limited features of the "Personal Report Generator".

Despite this I still give it a 5 Star rating and would recommend it to anyone who wants to make their 99 really earn its keep.

Peter Walker.

ASCOT '99 by Paul Burmiston. EXTENDED BASIC Command Module required.

This is a horserace game for 1-6 players with a maximum of 9 races a game. The computer randomly generates the names of six horses from the data statements inside the program. The form of each horse for its last five races is now displayed and from this the odds are calculated. With $1,000 you now bet and try to win as much money as possible.

When everyone has finished betting the race is on. The horses look like horses, not just a head moving along a line and the legs exactly simulate those of a horse galloping. When the race, which is 1½ laps long, has almost finished, the winning post appears, and the first past is the winner, unless disqualified (details with rules, contained in the program). The computer very accurately picks the winner and once the result has been announced, all the players losses or winnings are printed with the current state of his/her purse.

Very enjoyable with sound and brilliant use of the sprite graphics.

Send cheque/postal order £4.25 inc. p&p to:

Paul Burmiston, 19 Hereford Ave, Ollerton, NEWARK, Notts. NG22 9AE.

* 2021 note- although the screen display was based on a 40 column window on an 80 column line, it WAS possible to use TI Writer with 132 character lines of condensed print.

It is really surprising how fast time travels. It is now a year since I bought my own 99/4A machine – and since I joined TIHOME .... (thinks... will soon be time to renew subscription!)

My own reason for buying a computer was mainly due to "Information Technology Year" (remember that?)which was half-way finished, and I began wondering what could be in it for me. My experience with computers was limited to a one year stint I did as a computer control clerk at the age of 18 on Manchester Corporation's aged LEO III computer. This massive device occuplied almost two floors of Manchester's magnificent Gothic style town hall, used paper tape/card input and did its calculations on several banks of magnetic tape readers. Whilst it was calculating no other input/output system could operate... it was rather a slow and tedious process to say the least and I suspect that its memory size was no larger than many of today's popular micros!

Still – back to my purchase of the 99/4A ... There were a spate of thrillers on the T.V. around this time where the crooks were using computers for all sorts of nefarious activities – now as my full-time job is with the Probation Service, I thought that this opened several interesting possibilities.... I wandered around several shops selling micros and the two I was actively considering was the TI machine and the VIC 20. The VIC, of course, had only around 5K of memory against the TI 16K plus it looked and felt like a toy. However, it was £100 cheaper ... My dilemma was solved when I saw a tiny ad in The Guardian (note – all Probation Officers read the Guardian!) that Currys Micro-C were offering the TI machine at the same price as the VIC 20. Of course since then the price has dropped further and I think that this has been the main reason that sales of TI machines have only taken off recently, with the added result of a soaring membership of TIHOME. To give you some idea of how membership has increased, my first newsletter (actually dated 28th April 1982) was nothing more than 43 photo-copied sheets of poorly printed foolscap – compare that to the very professional effort of your last issue of TIdings. Paul Dicks was able to boast in that April 1982 edition that "Membership is around the 180 mark and still rising.." When I phoned Paul in April to ask him about something or other, he told me that membership was now at the 2000 mark (and still rising!) – WOW!! Some things however never change – even then, TIdings seemed to come much later than scheduled! Perhaps Paul could use some help – he does, after all have a full-time job himself on a big brother of the 99/4. I for one would be prepared to help out and I am sure that there are other volunteers eager to come forward – how about it Paul?

* _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ *

Probably you would have seen mentioned within these pages the International 99/4 Users-Group. This is really a great organisation to belong to. For an annual subscription of $ 12 you now get a bi-monthly magazine called Enthusiast '99 and on alternate months a bulletin called "The President's Letter". You also have access to literally hundreds of owner-written programs through the Software Exchange Library. This new Magazine which replaces the IUG newsletter from next month (May) will contain reviews of hardware and peripheral equipment

from both TI and third party sources as well as giving comparisons between TI products and other personal computers. Other sections will include "The Assembly Line" to help members to better understand the workings of TI's Mini-Memory Module and the Editor/Assembler, a Users-Group Spotlight (it seems that this will be restricted to those operating in the U.S. — BOO!), a "Member Spotlight" (and what are YOU doing with YOUR 99/4??), a "Women's View" by REGENA (I had always suspected that our mysterious author of 99'ER Magazine fame was female), a "Potpourri" of hints, tips, etc. and much more besides. The address to write to for a subscription is:

International 99/4 Users-Group, P.O. Box 67, Bethany, OK 73008, U.S.A.

* *

Observant readers of the last edition of TIdings might have noticed some TI-related items I had for sale. These included an Editor/Assembler manual which I had ordered from the States and intended to use with the Mini-Memory Module. However, panic set in as I realised that learning BASIC was causing me enough grief without getting out of my depth in assembly language. Needless to say the response to my advert was overwhelming. I sold it within 24 hours of TIdings appearing to a lucky Manchester member (no, it wasn't favouritism just because he was a fellow Mancunian — he really was the first). My apologies to all others who wrote to me — all your cheques/Postal Orders/cash/I.O.Us. have been returned. (Anyone out there interested in making me an offer for the Tombstone City and/or Addition Subtraction II Modules?? Phone me on 01-206 0796).

* *

Below you see an advert for a stand-alone 32K memory expansion which is one of a number of similar devices that are being advertised in 99'ER Magazine. I have written to all these companies to see if we as a users-group can obtain a bulk discount. However, I would like to know how many members would be interested in this, so please phone or write to me so that I will have something positive to negotiate with. I **do not** intend to do this on a business basis — any discount obtained will be passed on to members. Points to bear in mind: 1. The unit requires a power supply transformer suitable to our mains electricity i.e. 240v. 50cps down to the unit's 12v D.C. requirement using mini phono plug (these transformers are readily available in the U.K.) 2. The unit will have an 180 day warranty, but you must direct any claim arising from this direct to the supplier in the U.S.A. 3. Postage, duty, V.A.T. and the cost of me posting the unit to you will be extra. Nevertheless, I am hoping that if I can give the supplier a sufficiently high order, then the total cost to TIHOME members will be in the region of £125 at an exchange rate of $ 1.50. If you want to discuss this with me, please phone evenings or weekends.

* * * * * * *

MODEL 2000 32K MEMORY ADD ON (our most popular model)

This model connects to the I/O port on the right side of the TI 99/4A console. There is an I/O port on the right side of the unit which other peripherals may be connected to.
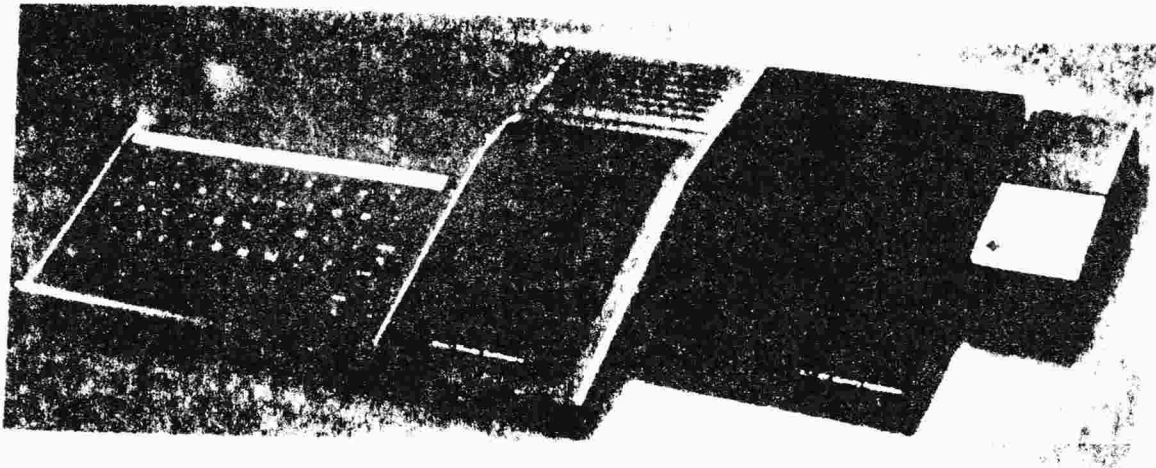
Model 2000 attached to Console and Speech Synthesizer

*TI's Peripheral Expansion Box or Hex-Bus adapter may be used if it is connected to the right of the MODEL 2000 32K MEMORY ADD ON.

*Other stand-alone peripherals such as the speech synthesizer, RS232 interface or Thermal Printer may be connected to the right side or left side of MODEL 2000.

SPECIFICATIONS

* Power supplied by a wall transformer.
* Power indicator LED on front.
* Attractive Black Anodized Aluminium box.
* Size 6 x 8 x 2-3/4 inches.

SUGGESTED RETAIL.... $ 190.00   Factory Discount Price... $160.00

SOFTWARE REVIEW TIME....SOFTWARE REVIEW TIME....SOFTWARE REVIEW

WARGAME (Tx Software)

This is the latest in a number of games put out by TX Software of 109, Abbotsweld, Harlow, Essex CM18 6TQ. All the games that I have seen by this company have been of a high programming quality and this is no exception. WARGAME is designed to run in Console Basic, but it can also be used in Extended Basic where speed is increased. Joysticks are not essential but I would personally recommend their use for a more enjoyable game.

The scenario is a detailed map with hills, forests, water and marshgrounds. Two opposing armies face each other. You may play against the computer at different levels of skill or play against other player. There are many tactical and strategic facets as you manoeuvre your tanks and artillery across the terrain with the object of destroying the enemy bases. This is by definition NOT a fast-moving arcade-style game, but one which needs logic skills — and an understanding of trajectories! Of course you get a new map for each new game which helps to make it even more interesting.

Yacobi Ratings:

Presentation On-Screen	* * * *	Value for Money	* * * * *
Program Quality	* * * * *	Ease of Use	* * *
Supplied Instructions	* * * * *	Interest Holding	* * * *

9

Conclusion: An excellent program for the thinking games player and good value at £6.00.

* _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ *

Well, that's all for this issue (will it make it to v.3 no. 2 in time??) except to say that due to a mishap I have a February copy of 99'ER Magazine surplus to requirements. £2.50 including postage — but phone me first to see if it has gone!

Happy Computing.

Arieh Yacobi.   56 Winchester Road, Kenton, Harrow,
Middlesex HA3 9PE. Phone: 01-206 0796.

LINE DRAWING ROUTINE
by
Ray Frowd

I am writing this article as part of the discussion of the use of the 99/4A's Hi-res graphics. The accompanying routine draws a straight line between any two points on the screen. It is in a fairly primitive form, but does contain some ideas that I have not seen before: and I hope some readers may find them useful.

The basis of this routine is that as a line is continuous, we need only use those hexadecimal codes that plot a continuous series of pixels. These are stored in the array P$; in lines 140 to 170.

The starting and finishing points are entered by a call key routine, in lines 180 to 320. As you know there are 192 x 256 pixels on the screen; so each point must be given two co-ordinates; first a row, then a column. Each co-ordinate must be entered as a three digit number; e.g. row 20 is entered as 020.

Lines 330 and 340; and, 360 and 370 calculate first the character position where the line starts, and the pixel where the line starts. Lines 350 and 390 calculate the two parameters that control the line:—

INC, the average number of pixels that should be plotted on each row
ALT, the number of rows that will be needed to plot the complete line

Working from the starting point the routine considers the screen one pixel row at a time. SGN(ALT) controls whether the line goes up or down. Lines 520 to 540 calculate W and Z. W is the number of pixels that should be plotted, but because only a whole number of pixels can be plotted it is necessary to calculate Z. Line 540 'remembers' the fractional part of a pixel that could not be plotted and carries it forward to the next line.

Line 560 calculates V, which is then used in line 570 to switch to one of 3 similar routines: Line 580 to 890 the line goes to the left. Lines 900 to 930 the line goes straight up or down. Lines 940 to 1250 the line goes to the right. These 3 routines all do the same thing. First they calculate whether the line is in the right or left half of a character position; then two further parameters M and N. M and N are used in the sub-routine starting at line 1260. M is the number of pixels to be plotted, and

N is the position of the first pixel. M and N are used to obtain the necessary hexadecimal code from the array P$. This is then inserted into the character definition string CH$, and then it is printed.

The procedure is repeated until ALT=0 which means the line is finished.

This program offers plenty of scope for further development. It can be adapted to draw curves. Also an adding routine is needed so that lines can cross.

Finally, to the curious I'm Robin Frowd's brother.

```
100 CALL CLEAR
110 B=3 2
120 Z$="000000000000000000"
130 COL=3
140 P$(1)="8421"
150 P$(2)="C63"
160 P$(3)="E7"
170 P$(4)="F"
180 FOR POINTS=1 TO 2
190 FOR CRD=1 TO 2
200 FOR H=1 TO 3
210 CALL KEY(0,K,S)
220 IF S=0 THEN 210
230 CALL HCHAR(23,COL,K)
240 K=K-48
250 HCD$=HCD$&STR$(K)
260 COL=COL+1
270 NEXT H
280 C(POINTS,CRD)=VAL(HCD$)
290 HCD$=" "
300 COL=COL+1
310 NEXT CRD
320 NEXT POINTS
330 ROW=INT(C(1,1)/8)+1
340 COL=INT(C(1,2)/8)+1
350 INC=(C(2,2)-C(1,2))/(C(2,1)-
C(1,1)+1)
360 R=C(1,1)-INT(C(1,1)/8)*8+1
370 L=C(1,2)-INT(C(1,2)/8)*8+1
380 CH$=Z$
390 ALT=C(2,1)-C(1,1)+1
400 R=R+SGN(ALT)
410 ALT=ALT-SGN(ALT)
420 IF R>0 THEN 470
430 ROW=ROW-1
```

```
440 R=8
450 B=B+1
460 CH$=Z$
470 IF R<9 THEN 520
480 ROW=ROW+1
490 R=1
500 B=B+1
510 CH$=Z$
520 W=W+INC
530 Z=INT(W)
540 W=W-Z
550 U=ABS(Z)
560 V=SGN(Z)*(SGN(ALT)-(ALT=0))
570 ON V+2 GOTO 580,900,940
580 IF L<5 THEN 700
590 IF U>(L-4)THEN 650
600 M=U
610 N=L-U-3
620 GOSUB 1260
630 L=L-U
640 IF ALT=0 THEN 1330 ELSE 400
650 M=L-4
660 N=1
670 GOSUB 1260
680 L=4
690 U=U-L
700 IF U>L THEN 810
710 M=U
720 N=L-U+1
730 GOSUB 1260
740 L=L-U
750 IF L>0 THEN 800
760 COL=COL-1
770 B=B+1
780 CH$=Z$
790 L=8
800 IF ALT=0 THEN 1330 ELSE 400
810 M=L
820 N=1
830 GOSUB 1260
840 COL=COL-1
850 CH$=Z$
860 B=B+1
870 U=U-L
880 L=8
```

```
890 GOTO 590
900 M=1
910 N=L+4*(L>4)
920 GOSUB 1260
930 IF ALT=6 THEN 1330 ELSE 400
940 IF L<5 THEN 1140
950 IF U>(9-L)THEN 1060
960 M=U
970 N=L-4
980 GOSUB 1260
990 L=L+U
1000 IF L<9 THEN 1050
1010 L=1
1020 COL=COL+1
1030 B=B+1
1040 CH$=Z$
1050 IF ALT=0 THEN 1330 ELSE 400
1060 M=9-L
1070 N=L-4
1080 GOSUB 1260
1090 COL=COL+1
1100 CH$=Z$
1110 B=B+1
1120 U=U-9+L
1130 L=1
1140 IF U>(5-L)THEN 1200
1150 M=U
1160 N=L
1170 GOSUB 1260
1180 L=L+U
1190 IF ALT=0 THEN 1330 ELSE 400
1200 M=5-L
1210 N=L
1220 GOSUB 1260
1230 U=U-5+L
1240 L=5
1250 GOTO 950
1260 AD$=SEG$(P$(M),N,1)
1270 LONE=(R-1)*2-(L>4)
1280 LTWO=LONE+2
1290 CH$=SEG$(CH$,1,LONE)&AD$&SE
G$(CH$,LTWO,15)
1300 CALL CHAR(B,CH$)
1310 CALL HCHAR(ROW,COL,B)
1320 RETURN
1330 GOTO 130
```

APEX TRADING LTD

QUALITY SOFTWARE AT SENSIBLE PRICES

All the programs listed below are on cassette and will run on the standard unexpanded machine. All tapes are covered by an unconditional replacement guarantee.

Order code	Title	Price
GM1	CATERPILLAR	£1.95
GM4	SNAKE	£2.95
GM5	HAUNTED HOUSE	£3.95
GM6	SORCERER'S CASTLE	£3.95
GM7	3-D MAZE	£3.95
GM8	NIM	£2.95
GM9	FRUIT MACHINE	£2.95
GM16	ISLAND ADVENTURE	£3.95
GM17	FORBIDDEN CITY	£4.95
GM18	PHARAOH'S CURSE	£4.95
GM19	DEVIL'S ISLAND	£4.95
GM21	CHALICE	£3.95
GM22	BOMBER	£3.95
BC10	Blank C10 computer cassettes	£0.50
	THE TEXAS PROGRAM BOOK (35 programs for the TI99/4A)	£5.95

NEW

All the above prices include VAT. Please add 50p p&p to orders under £6. Orders over £6 are post free. Send your order, plus cheque or P.O. to:

APEX SOFTWARE,
115, Crescent Drive South,
Brighton, BN2 6SB.
Tel. Brighton (0273) 36894

Access/Barclaycard holders may telephone their orders at any time.

Bab^bl_ing Br^{oo}k_s

I haven't seen the February issue of Tidings yet (April 3rd) but I understand it is being printed rather than reduction-Xeroxed, so we have climbed yet another rung in the ladder. The TIHOME membership must be around the 2000 mark now and growing in leaps and bounds, although few recent buyers seem to have heard of us. A warm welcome to those recently joining our happy throng, and do you realise just what you have let yourself in for?

Anyway, the big story of the moment is not the 99/2, not the CC-40, nor is it the holding-over of the MATHS articles for a while. No, the big story is that one Babbling Brooks is getting hitched, acquiring a family of four on the way. Also, my address has changed (again) and I am once more available by phone. The new address is:—

29 Kestrel Crescent, Blackbird Leys, Oxford OX4 5DY.

and the phone is OXFORD 717985 but please don't call late (i.e. after say 10 pm). Coupled to this is an increase in 99 activity in the Oxford area and it is possible that a small Oxford group may be set up later this year. I have yet to arrange a suitable venue, and anyone will be welcome. As my own time will be scarce, once the group is up and running I will probably take a back seat, so any contenders for the post of coconut shy, sorry, group organiser should make themselves known. (And if you happen to pass through Oxford and you are a breaker, the handle is Flying Dutchman but I'm rarely on channel these days, although my much-better-half, Valentine, might catch you on 19!).

This Babble has temporarily lost the MATHS series, to make room for a number of extra goodies: the continuation of Gary Harding's Assembler series; more from Dutch fellow-enthusiast Bill van Kerkoerle (in unretouched original this time); a review of the NEC's book, 30 HOUR BASIC; the usual Beginners' BASIC — getting a little more complex this time; a report on the National Exhibition Centre's WHICH COMPUTER Show from NIGEL CLEMONS (More from Nigel next issue as well); some jottings on LOGO from teacher BETTY LUMLEY; the final part of the 3D Noughts and Crosses article — this time the listing!; and a short Soapbox on the whys and wherefores of using O and 0 in listings (and if that has confused you, think what it does to readers of the listings!)

Although there is nothing planned at the time of writing, no doubt there will be good reason to add a Postscript as well!

I'm still researching material for future Babbles, and SORTING and SEARCHING is a contender for the not--too-- distant future. There is also a shortie on Forth, Structured Programming, Life, The Universe, and Everything, somewhere in the pipeline, as well as a tentative foray into the interior of the 'soft' side of the Speech Synthesiser.

I have also been preparing short notes on Plotting (HiRes) and other Graphics routines, which were first covered back in issue 2.1, and which will eventually be available for the cost of two first class stamps and an envelope. The proposed

booklet on Hints & Tips is still just that: 'proposed', and the Great Brooks' BASIC and Programming Guide is still just a twinkle in the jaundiced eye of the author...

Which reminds me, there is a 4A book due to be published shortly (May/June); I will review it as soon as I can, unless someone else rings me and tells me to back off!

S O A P B O X

This issue's Soapbox is more of a warning about what may appear on the surface to be trivial, but which has caused at least one member considerable irritation. The source of the problem is the poor distinction on screen between the letter 'O' and the digit '0'. Most other manufacturers have distinguished between zero and O by placing a diagonal line through either one or the other, while TI simply square off their Os (which, on a poorly-adjusted TV cannot be distinguished from zeros). The extra-fast plotter given in the December '82 Tidings wouldn't run for this member because a DATA line had been entered incorrectly. Two letter Os had been accidentally used instead of two zeros, and as they appeared in a hexadecimal definition string the program kept crashing. It is a lot easier than you might think to make such a mistake. Another program owned by the same member also wouldn't run, and after seeking out the offending sections in one line, I was confronted by a whole sequence of erring letter Os and zeros having been used in the wrong context.

For example, the use of zero instead of letter O as a variable name, which either had an unwanted effect in the evaluation of INT(O), or caused a BAD NAME error in the use of O = O + B, where it should have been a letter O, but instead was a zero in both instances!

So, RULE NUMBER 441: do NOT use the letter O as a variable name, and try not to use O as the first letter in a name (in fact, avoid its use like the plague ANY-WHERE!). RULE NUMBER 442 is Watch Where You're Putting Your Fingers!

BEGINNERS' BASIC

Back in the early days when I conceived the Beginners' BASIC series, the original intention was to combine a slow, easy, introduction to the language with an equally-slow, easy introduction to 'programming'. Unfortunately, it has turned out that in order to begin the introduction to programming you really need quite a large base or background of BASIC commands from which to draw examples. This in turn requires the reader to understand the same base or background, and this has meant that I am having to spend the first few (= several) articles laying the foundation for that background.

As usual, if there is anything to be typed into your 99 (followed by ENTER), it will be 'indented' on the page, thus:—

        PRINT "FOR EXAMPLE"

If the indented material is on several lines, then use ENTER after every line (I may have caused some confusion in the past by not having made this point clear!), and don't forget that one line on this A5 page may become MORE than one line on your TV screen.

This issue we'll delve deeper into the NUMERIC side of things, examining the

functions which can be used with NUMBERS and NUMERIC VARIABLES.
These are:—

*	+	—	/	∧	
ABS	SGN	LOG	TAN	SIN	STR$
COS	ATN	SQR	EXP	INT	CHR$
RND	RANDOMIZE				

The more 'aware' or 'literate' of you might have noticed that technically STR$ and CHR$ qualify as STRING FUNCTIONS, but this issue is concerned with functions USING numeric variables or numbers.

The first row in the small table above show the asterisk (multiplication), addition, subtraction, oblique (division), and caret (exponentiation), known collectively as OPERATORS. The caret is perhaps the only one which needs some explanation, and a very detailed one was given in the early part of the MATHS series. For the purposes of brevity, if you are familiar with the notation 'X^3', it is written 'X∧3' in BASIC! Looking at the remaining functions, they are:—

ABS. This is short for ABSOLUTE, and is a way of removing the minus sign from a number (to put it simply). ABS(−2.7) is 2.7, while ABS(23) is still 23. You may not be able to see an immediate use for this function, but it can be very useful sometimes. I used it in my DRAW subroutine, given in an earlier issue of Tidings. You might use it thus:

```
A = ABS(−12.345)
PRINT A
```

SGN This is short for SIGNUM or SIGN, and can be used to tell you whether a number is negative, zero, or positive. Values of −1, 0, and 1 respectively are given. Again, it is a function which you might not immediately see a use for; again, it is one that I used in the DRAW subroutine back in volume 2.

```
PRINT SGN(−12.345); SGN(0); SGN(120)
```

LOG This again is an abbreviated name — for LOGARITHM — and a full explanation was given in the MATHS series earlier. If you know about these things, this function is, confusingly, the NAPIERIAN logarithm, otherwise shown as $\log_e$ or 1n in literature. It has a counterpart, the anti-logarithm, which we will see later.

```
PRINT LOG(10)
```

TAN This is one of the few TRIGONOMETRIC functions — the TANGENT — and it will be covered in detail in a future MATHS episode, along with some of the others which we will see in this issue. To confuse the issue, the function doesn't use degrees, but RADIANS.

```
PRINT TAN(.44)
```

SIN Another trigonometric function — SINE — which will also be covered in a future issue of the MATHS articles. It too uses RADIANS.

```
PRINT SIN(.523)
```

**STR$**   The STR$ function (which I pronounce as 'string-string') turns the internal representation of a number (i.e., the way in which the computer stores the number in its memory) into that of a string. This can be useful for separating a number up into sections, or for searching a number for a particular digit or group of digits. You may not find it that fascinating, but it can be very useful indeed. On screen, the only difference you will see is that in PRINTing, the STR$ version of a number loses its leading and trailing spaces — which can really only be noticed in direct comparison with the original number! Try this:—

        A = 123.456
        A$ = STR$ (A)
        PRINT A:A$

COS   Back to the trigonometrics again! This time it is the COSINE, once again using the dreaded RADIANS, and once again to be discussed in more detail in a future MATHS article.

        PRINT COS(1.05)

ATN   This is the only 'anti—' trigonometric function which TI saw fit to include; even the lowly Sinclairs have the full set. It is short for ARCTANGENT (sometimes shown on calculators as tan -1 which, if you have read the MATHS series, you will know is very wrong!) and gives you the reverse (if that makes sense) of TANGENT. ATN does have one additional use — it can save you a lot of typing (and remembering of digits) when you want to use PI, as 4*ATN(1) gives the same result.

        PRINT 4*ATN(1)

SQR   This is the SQUARE ROOT function, and gives you the number which, when multiplied by itself, will produce the number you used SQR on (if you see what I mean.....)
This is the BASIC equivalent of $\sqrt{}$

        PRINT SQR(144)

EXP   I mentioned earlier that there was an anti-logarithm — this is it. It is the EXPONENTIAL, which can be confused with EXPONENTIATION (yes, they do refer to the same action, but don't pick holes). You will see it as e^x on calculators and in the literature, where 'x' is the number to be 'anti-logged', and 'e' is a constant. The MATHS series gave a detailed explanation of this and other related topics.

        PRINT EXP(2.30258)

INT   Short for INTEGER, it can be used to separate the digits to the left of a decimal point from a 'whole number'. INT(123.456) is therefore 123. In other words, it lops off the FRACTION.

        PRINT INT(1.99)

18

By the way, you can obtain the counterpart to INT (carefully not provided on this supposedly scientific-orientated machine) called FRAC on calculators and on some other lesser machines, and you use INT to do it.

```
X = 1.99
PRINT X - INT(X)
```

**CHR$**   This one I pronounce 'charstring', and is associated with the dreaded ASCII codes which we have not yet fully covered. Give it a number between 0 and 255 and it will provide the character associated with that number. CHR$(65) is the letter 'A', while CHR$(32) is the 'space'.

```
PRINT CHR$(30)
```

Bet that has you a little confused.....

The last two take us into a rather murky area — that of 'pseudo-random numbers'.

RND   RND is a special function which I have included because it is used directly with numbers (or it can be used on its own). All the others use an ARGUMENT — i.e., there is usually something in brackets following the function, but not with RND.

```
PRINT RND
```

You should have got a FRACTION, produced 'randomly' internally.

```
PRINT RND;RND;RND
```

A different one each time! You can use this with INT to obtain ranges of numbers for use in games for example — like those involving DICE.

```
PRINT INT(RND * 6 + 1)
```

That short statement will produce the numbers 1,2,3,4,5 and 6 on what amounts to a random basis. Try this short program:

```
1  PRINT INT(RND * 6 + 1);
2  GOTO 1
```

Now RUN it — to stop you will need to use the BREAK key. There have been some highly interesting articles in the micro-press just recently on this subject, and unless anyone feels sufficiently moved to do so, I will attempt a foray into this field at a later date.

RANDOMIZE   The sequence of random fractions produced by RND can be made to repeat itself by using RANDOMIZE with a following number e.g., RANDOMIZE 7. Alternatively, the series can be made to start 'at random' simply by using RANDOMIZE on its own.

To try and give some idea of how this occurs, enter this short program:

```
NEW
1  RANDOMIZE 1
2  PRINT INT(RND * 6 + 1);
3  GOTO 2
```

RUNning this program as it stands will give one sequence of 'dice' numbers; altering line 1 will give a differing series. Using just RANDOMIZE will mean that a repeating series will not occur. Taking line 1 out altogether and RUNning several times will show that a series can be made to repeat also; the difference is that you can insert the RANDOMIZE n function anywhere in a program several times, and restart the series all over again.

If you have found this particular episode heavy going, do not be disheartened. Most of the functions here will be unlikely to be of use to you for some time to come, but it is useful for you to have some exposure to them so that you have at least come across their names!!

There is also little point in going deeply into their function and use at this stage, as that would largely be lost on you if you are a beginner in every sense of the word. There is plenty of time for you to get into the meatier information behind the 'TRIG' functions, for example, in later Tidings.

RADIANS, which probably mean as much to you as GRADIANS, will also get the BB treatment in later issues, so provided you keep up your membership of TIHOME you will get to suffer further over the months to come!

Next issue, family matters notwithstanding, we will go over the STRING FUNCTIONS, which may be of more interest to you (mostly because there are fewer of them!).

N.E.C. 30 HOUR BASIC : A REVIEW

Pete Brooks

The National Extension College has produced a course in BASIC for £5.95, claiming to teach you BASIC with or without access to a micro in 30 hours. The book is written by CLIVE PRIGMORE, and the copy I have was the fourth printing for 1982. Its ISBN number, if you want it, is 0 86082 269 9, and there are two versions of it available, so make sure you get the right one. The other version is specifically for the Sinclair ZX81.

This version is intended for all others (except the ZX80 — probably because that one doesn't have floating point maths and the other goodies, although there are probably still primitive micros about which suffer equally from a lack of facilities), but as it uses primarily the 'BBC Micro' and is intended for use with the late BBC Computer Programme (note spelling), you might guess that is is not ideal for the 99/4A.

First impressions are quite good. It is a hefty book, well-designed with an over-large plastic ring binding which enables you to open the thing and lay it flat by your micro if you have one. It is a little larger in area than Tidings, and about 1 cm thick. It contains about 256 pages (what a coincidence!) of very small type, some of which is high-lighted in a penetrating 'Kermit' green.

The blurb on the back of the cover as usual is filled with half-truths and down-right cobblers. For example, — "Microcomputers are the tool of the 80's. BASIC is the language that all of them use. So the sooner you learn BASIC, the sooner

you will understand the microcomputer revolution." ALL of them? So the Jupiter ACE, on which I have programmed in Forth, doesn't exist? And since when does learning a language have a direct bearing on your understanding of a hardware revolution?

Eventually they do come clean inside, and you learn that the book can't actually teach you all about BASIC — there is a part 2 yet to come — but what it does teach, it teaches well. The approach to arrays is novel, treating them initially as lists, although you may think otherwise.

It is a good book if you are used to the self-imposed discipline of isolated study; I would recommend that you use the book in a group otherwise, or better, with someone who has a smattering of any computer language and a simple understanding of the idea of 'programming'. (If you have a degree you'll find this book a doddle!)

There are differences between the versions of BASIC catered for in the book's text, and the supposed standard BASIC which the 4 and 4A uses, so if you are an abject beginner, you may have some problems here, notably with POS, VAL, RND, and so on, as they are interpreted differently, (POS doesn't figure at all!)

The book is divided into 11 sections: a 'How To Use This Court', 9 course units, and an Index. The text begins poorly — hands up anyone who knew what Microsoft BASIC was before they started programming. Er.. hands up anyone who knows what it is even now? Still, it does improve, and the high-lighting helps in many cases, although the use of the green bits isn't consistent — sometimes on the first use of a word or phrase, sometimes not.

There are lots of small routines to be keyed in and experimented with, and the early part of the book appears to run parallel with my own Beginners' BASIC, so it can't be ALL bad!!

I don't think I could recommend this book for anyone under say 16 who didn't really have more than a smattering of Computer Science under their belts, but it is a very good reference book on BASIC and programming, especially on the so-called Structured approach to programming. If you have some grey areas which the TI BASIC book hasn't fully reached, well, this isn't exactly Heineken but it'll do.

There are also a few introductory routines and explanations on Sorting & Searching, which to my mind are essential tools for any kind of serious programming, not that I'm suggesting that there is a kind of frivolous programming!

One pount I did notice which irritated me a little is that the text recognises RET and RETURN as keys for entering information, but CR or ENTER don't figure. Presumably the ZX81 version does at least mention NEWLINE.

A final hint: if you do buy the book, don't flick through it, work through it in a methodical manner, or you'll be put off by the apparent complexity toward the end. I have a feeling that trying to absorb a little of everything at a glance is precisely the thing which puts many budding programmers off; it's like flicking through a detailed car maintenance manual and expecting to understand the operation of a carburettor system as a result.

THE 'WHICH COMPUTER' SHOW : N.E.C. BIRMINGHAM

A report from Nigel Clemons

The Which Computer Show held at the NEC Birmingham (18 — 21 January 1983) was predominantly for the Business-system user and not the Home user, unless he/she has a few thousand pounds to spend.

We went first to the Apple stand to take a look at their new 16 bit machine, 'LISA' (as advertised in the Sunday Times, 23rd January). We looked high and low through the mass of bodies, but Lisa could not be found. Is this another case of the advertising man getting his facts wrong again?

Over on the IBM stand was their new personal computer, which was on official release in this country from the 18th January, about a year after its release in the USA. The only IBM Personal Computer I could get at was one that was not wired up, but I did like the feel of its keyboard and its looks. For anyone interested in further details see BYTE, January 1982 issue.

Osborne had their so-called portable system on display. I found the 3 x 2 inch (approx) CRT far too small for any real use, although an external VDU can be connected. I think that they would have been better off if they had designed this system with a larger, LCD, display, similar to the Casio, Epson, or Hewlett Packard portable systems.

The Epson HX20 portable computer was very neat and compact even with printer and extra RAM. The printer was similar to the type used with cash registers, giving very good results with text and graphics.

Kaga Electronics, a Japanese firm, had on display a range of VDUs with black and white, green, and orange, screens plus full colour models. From the available handouts it seems possible to connect these VDUs to any 'Home Computer' via a P —adaptor. They range in price from £99.50 to £416.00. The colour VDUs come in three models, from normal resolution and character specifications to super-high resolution and characters. All models are a cube measuring approximately 30 cm, mounted in a well-finished silver grey cabinet. The colour definition was far superior when compared with domestic TVs.

Finally, on the Texas stand, apart from their business system computers, they had a Porsche racing car, which was part of a competition they were holding. A TI 99/4A was being given away every day for the duration of the exhibition (hope I win one).

The business systems that TI produce come in 5 models, the two smaller models using our old friend the TMS 9900, and the other three using the new TMS 99000. As with most stands at the show, it was very difficult to get near the micros on display, but I did manage to talk to one of TI's reps about the 99/4A and the TMS 9900, with reference to the information they provide — viz., what has taken the place of the book: '9900 FAMILY SYSTEMS DESIGN & DATA BOOK', which has very useful chapters on instruction sets and assembly language for the TMS 9900, but seems to be out of print. Perhaps my request through the rep may bring forth some useful information.

TMS9900 ASSEMBLY LANGUAGE ON THE TI 99/4A HOME COMPUTER

Gary Harding.

Recap of Part 1

The first part of this series attempted to introduce some of the basic concepts involved in 9900 assembly language programming, by contrasting its (relatively) primitive facilities with those available in TI BASIC, including the restriction to 16-bit integers (whole numbers) in numeric computations. However, apart from a not-very-clear explanation of the difference in interpretation between signed and unsigned integers, not a great deal of ground was covered as far as actually producing assembly language programs is concerned. While there is still a lot to go through, I shall try to get a little closer towards that ultimate goal in Part 2. We continue where we left off last time — in the bowels of the 9900 CPU.

Part 2 : Welcome to the Machine

Towards the end of the last part, I introduced you to two of the 9900's three on-chip Processor Registers — the Program Counter (PC), which helps the 9900 find its next instruction in memory, and the Status Register (ST) which holds various system status information. These are basic features necessary on any processor, though, and are not unique to the 9900. What makes the 9900 special is its third and last on-chip register, the WORKSPACE POINTER (WP for short). To explain why, I'll have to go back in time to the earliest computers of the 1950's.

These machines (size of a room, huge power consumption, blah blah, cut to shot of James Burke with a silicon chip on his fingertip) were extremely simple in the way in which they processed data. At the heart of their CPU's (such as they were) was but a single register or ACCUMULATOR, which was the only place in the whole machine where you could get anything useful done (like simple arithmetic). Virtually all of the machine's instructions would operate only on the contents of this accumulator, which meant there was a huge amount of "traffic" between main memory and the accumulator as data were loaded into the accumulator, manipulated, and then had to be stored back in memory again before a fresh calculation could be carried out. And memory access was many times slower then than it is now. So, to increase performance, the number of accumulators available within the processor was increased, to two at first but later to eight and even sixteen. With a larger number of accumulators available, of course, it became possible to perform much more processing wholly within the CPU, storing any intermediate results in spare accumulators rather than back in memory, thus saving a considerable amount of time. Also, as the number of accumulators in the processor increased, it became possible to use them in different, more subtle, ways; for example, to hold the address of (a pointer to) a piece of data rather than the piece of data itself. The concept of single-purpose accumulators faded, and they became a set of GENERAL REGISTERS which could take on a variety of roles. Nowadays you will find all microprocessors come with a number of these general registers — for example, the Intel 8086 has four general-purpose 16-bit registers. However, they're still part of the hardware, built into the CPU itself.

The Texas Instruments TMS9900, introduced in 1976 as a single-chip version of

the processor used in TI's 990 series of minicomputers, has no such general registers in its hardware. Instead, it has a single register (the WP) which holds the start address of a block of sixteen consecutive 16-bit words of memory which act as a set of 16 general-purpose registers. The block of 16 words is called a WORKSPACE (Hence Workspace Pointer) and the registers are referred to as WORKSPACE REGISTERS, numbered 0 to 15. Although only one Workspace may be in use (or "active") at one time, the user may set up as many as he likes and switch between them by simply changing the WP. It may not be immediately apparent why you should want to do this but, as we'll find out later, it's a necessary and very useful feature of the 9900. The decision to take the registers out of the CPU and put them into RAM was taken against a background of steadily-increasing speed of memory access, where the original reasons for avoiding slow transfers between CPU and memory were becoming progressively less valid. In fact, the 9900's "memory-to-memory architecture" made it possible to perform many operations on data in memory (for example, adding one word to another) in a single instruction, without having to explicitly load or store an accumulator or register as an intermediate step. In this way, a single 9900 instruction may do the work of 2 or 3 Z80 instructions.

You may be wondering by now why we carry on using the concept of registers when, with processors as powerful as the 9900, we can perform complex manipulation of data in memory directly, i.e. without having to load the data into a register. Well, the reasons are many; for a start, instructions take up less room in memory when they operate on registers rather than on ordinary memory locations. For example, the instruction to subtract the contents of one Workspace Register from another takes up one 16-bit word (2 bytes), whereas the same subtraction operating on two words in memory takes up 3 words. This is because, in the latter case, the actual 16-bit addresses of the two words in memory have to be embedded in the instruction itself, to enable the CPU to locate them. If registers are used as operands, however, the CPU already knows vaguely where they are in memory (through the WP) and does not need to be supplied with a full 16-bit address to exactly locate them. In fact, as there are sixteen registers, it only needs four bits ($2^4=16$) to be able to generate each register's memory address for itself. This means that instructions involving registers will always be more compact — and (reason no. 2 for using registers) that means they will execute faster. The 9900 can only read instructions from memory one word at a time (along its 16-bit data bus), so it has to take a three-word instruction in three gulps, requiring three memory accesses, before it can execute the instruction. A one-word register — only instruction requires only one such "instruction fetch" and therefore takes less time overall to complete. In the subtraction example above, using registers takes about 5 micro-seconds, while the other method takes about twice as long.

Okay, that just about covers the Processor Registers (WP, PC, & ST) and the Workspace Registers. I hope you've got some idea of what they are and why they are used. Now I think we can start to get down to business by examining the anatomy of a 9900 Assembly Language instruction as you or I would write it. They're all fairly straight-forward and follow the same basic pattern, consisting of four fields separated by spaces :—

LABEL MNEMONIC OPERANDS COMMENT

24

In detail, the fields and their functions are as follows:—

1. LABEL :   This field is optional. Labels are just strings of characters beginning with an alphabetic character in the first character-position ("column 1") of the statement. The maximum allowable length of a label depends on whether you are using the Line-by-Line Assembler (LBLA) & Mini Memory or the Editor/Assembler. With the former, you are only allowed 1— or 2—character labels, whereas the Editor/Assembler allows up to 6 characters. In both cases, though, all characters after the first must be alpha-numeric (i.e. a letter or a number). If the label field is not used, column 1 must contain a space (followed by more spaces and/or the mnemonic field) or an asterisk (in which case the whole line is treated as a comment, i.e. ignored).

A label is usually placed in an instruction to perform the same function as a line number in TI BASIC, that is to make the instruction available as a destination for a jump instruction (like a GOTO). The Assembler does this by placing the label and the location of the associated instruction into its Symbol Table. Any subsequent jump instruction referring to the label can then be assembled properly by picking up the label's location from the symbol table. For this reason, labels are often called SYMBOLS, and the associated location is called the VALUE of the symbol. Symbols can also be used to label data areas in memory in much the same way as variable names in TI BASIC. If a label is included in an assembler directive which reserves a word of memory (e.g. a DATA directive) then it becomes possible to refer to that word in the operand field of an instruction by using the label or symbol. Exactly how this is done we'll see later when we examine SYMBOLIC ADDRESSING'

(NOTE: Page 7 of the Line-by-Line Assembler (LBLA) "manual" attempts to explain the difference between assembly language instructions and the superficially-similar Assembler Directives. It should be stressed that the former produce the machine code which is executed by the 9900 and the latter merely affect the process of assembly by reserving and initialising blocks of memory for use of data area (e.g. BSS & DATA) and determining the location of the assembled code (e.g. AORG & RORG). The Editor/Assembler has a comprehensive set of 28 Assembler directives, while the LBLA has a modest subset of the six most useful ones plus one of its very own (SYM) to allow you to look at its symbol table).

2. MNEMONIC : (n, a device designed to aid the memory). This field is compulsory unless the line is totally blank (not allowed in the LBLA) or wholly a comment (asterisk in position 1). It is sometimes called the op-code field, and holds the mnemonic (a kind of shorthand descriptive name) for one of the 9900's 69 machine instructions. For example, A is the mnemonic for the Add instruction, and MPY is the mnemonic for MultiPlY. Alternatively, the field may contain a mnemonic for an Assembler directive, such as BSS or EQU.

3. OPERANDS : Whether or not this field is required, and what form it takes, depends largely on the mnemonic which precedes it. Assembler directives' operands are usually very simple numeric expressions, text strings, or symbols with not much sophistication involved, so I won't bother running through them here. Assembly language instructions' operands are more complicated, however. Depending on the instruction, zero, one, or two (separated by a comma) operands

may be required. In general, the purpose of operands is to specify the memory location(s) of the data to be used by the instruction. In other words, while the mnemonic describes what is to be done, the operands specify where it is to be done. They do this through the use of ADDRESSING MODES, of which more in a minute, after I deal with the last field of the instruction.

4. COMMENT : This field follows the last operand, separated from it by at least one space, and is optional. Anything in the field is ignored by the Assembler, like REM statements in TI BASIC, and serves to annotate your program. However, unless you take Polaroids of the screen at regular intervals while writing your program, comments are not going to be of any permanent use if you are using the LBLA, as they disappear along with your source code (the assembly language instructions) when you exit from the LBLA. But normally, commenting your programs is very important — even if you're the only person who's ever going to see them!

Right! That didn't take long — now back to Addressing Modes. This is yet another (the last, I hope) fundamental concept that you'll have to grasp in order to be able to use any assembly language. Putting it in simple terms, instructions need data. To get that data, the CPU needs to know where it is, i.e. its address. And this information is provided through the operands in one of several different ways. These ways are the Addressing Modes. We already touched on the subject when we compared the speed of the subtract instruction with two kinds of operand. We automatically assumed that the same operation could be performed in more than one way. Well, of course, we were right thanks to the existence of five different addressing modes in which the 9900 can locate the data it wants. Most of the 9900's commonly-used instructions can have operands which use any of these five modes, and for those that can't there are 3 "special-purpose" modes which are specific to those instructions and catch everything that doesn't fit into the five main modes. These five general-purpose addressing modes are as follows:—

a) Workspace Register Addressing
b) Workspace Register Indirect Addressing
c) Workspace Register Idirect Addressing with Auto-Increment
d) Symbolic Memory Addressing
e) Indexed Memory Addressing

To allow me to give examples of their use, I'll use Subtract (S) as an example of an instruction which requires two operands, and Negate (NEG) as an example of an instruction which only needs one.

a) Workspace Register Addressing

This is the most straightforward of the general-purpose addressing modes, and is used when the data to be used by the instruction is contained in one of the 16 Workspace Registers (which it will be — or ought to be — most of the time). It's the simplest one to write, too. It's written as the register number (between 0 & 15) preceded by the letter R. Hence register 1 is referred to as R1, register 12 is R12, and so on. In point of fact, the "R" is optional, but is almost always used for the sake of clarity. So, for example, to subtract the contents of register 5 from the con-

tents of register 0, you write S R5,R0. And to negate the contents of register 10 (after a while you tend to just say "negate register 10"), you just write NEG R10.

b)   Workspace Register Indirect Addressing

The Concise Oxford Dictionary defines "indirect" as "not going straight to the point " (in which case, maybe this is an indirect series of articles?) And that's precisely how this addressing mode works. As in (a), you specify a Workspace Register and the 9900 uses its contents. But it doesn't stop there — it then interprets the 16-bit number in the register as an address, and uses the data at that address in the instruction. If that seems a little involved, it's probably just the way I've explained it. All it means is that the specified register contains the address of the data rather than the data itself. I'd say that was a pretty indirect method, wouldn't you? To use this addressing mode, you write the register just as in (a), but you precede it with an asterisk, for example *R1 or *R12. So to negate the word in memory whose address is held in register 10, you'd write the instruction NEG *R10. And to subtract the contents of register 5 from the word whose address is in register 0, you'd write S R5, *R0 (note that you can use a different addressing mode for each operand of a 2-operand instruction). Now, it may not be immediately obvious how indirection can be useful, but it's actually quite a handy way of addressing memory lying outside of the Workspace, especially if the same location is to be accessed many times in the same program. This is because loading an address into a register and from then on referring to it through indirection takes up less program space than referring to the address explicitly each time, using Symbolic Addressing (see mode (d)). In fact, the 99/4A's monitor program uses this economical method for its frequent writes to VDP RAM (i.e. the screen).

c)   Workspace Register Indirect Addressing with Auto-Increment

As you might expect, this is indirection with knobs on. The procedure involved here is exactly the same as in (b), but with one extra step involved. The extra step is simply the addition of either 1 or 2 to the contents of the specified Workspace Register AFTER the operand address has been read from it. One is added if the instruction is a byte instruction (i.e. an instruction operating on an 8-bit byte rather than a 16-bit word), and two is added if the instruction is a word instruction. The written form of an operand of this type is as in (b) but with a plus sign tacked on the end, e.g. *R1+ or *R12+. Continuing with our standard examples, the instruction S *R5+, R0 will subtract the word whose address is in register 5 from register 0, and then add 2 to the contents of R5 (the corresponding byte instruction SB *R5+, R0 would result in only 1 being added to R5). You should be able to work out for yourself what NEG *R10+ does.

The way that it enables so much to be done in a single instruction makes the auto-increment facility quite powerful. However, its real usefulness is sometimes difficult for the assembly language student to cotton onto at first (I know I didn't!). A proper example will appear later on in this series, once we've learnt a few more instructions, but for now suffice to say that it's a very useful tool for searching tables, moving blocks of memory, and other similar situations where continued looping is required.

d)   Symbolic Memory Addressing

This is sometimes called Direct Addressing. You'll remember that earlier I said that if you label a location or instruction then the label (or SYMBOL) may be used elsewhere in your program to address that location. Well, this is how it's done.

This is the first addressing mode we've examined which doesn't rely on registers to hold either the operand or its address. This is because the full 16-bit address of the operand is actually contained in the instruction itself. As we saw earlier, this makes for a longer instruction, both in terms of the memory space taken up and the time taken to execute. However, there are times when using Symbolic Addressing is inevitable, for example when you run out of spare Workspace Registers and need to store data outside of the Workspace. Then, the advantage of Symbolic Addressing is that, even though the instruction requires you to supply the actual operand address, you can use a symbol instead. The Assembler then uses the value of that symbol as the operand address. This means that you don't even need to know where the operand is, just its "name". Using symbols in this way is very much akin to the use of variable names in TI BASIC, although of course here you are limited to 2 (or 6) characters for your "names". To use this addressing mode, you must define the symbol you refer to, otherwise you will end up with an "unresolved references" error. You can define a symbol by placing it in the label field of an instruction or Assembler directive. In most cases this will result in the symbol assuming the current value of the Assembler's "location counter" at that point, i.e. the address at which the labelled instruction or directive is assembled. However, with the EQU directive you can explicitly allocate any numeric value you like to a symbol — the effect is to load the symbol and its value into the Assembler's Symbol Table without otherwise affecting the assembly process. The format of the directive is LABEL EQU CONSTANT, where the constant is a decimal number (e.g. 968), a hexadecimal number (e.g. >35F (the ">" sign denotes hex)), or a previously-defined symbol. The Editor/Assembler allows an expression as well as constants, provided any symbols in it have been previously defined. But I digress! Back to our examples. The form of this type of operand is simply the symbol preceded by the "@" character, for example @FRED or @TABLE3. So S R5,@XYZ will subtract the contents of register 5 from the word at location XYZ, i.e. the word whose address is the value of symbol XYZ. If the symbol WHO has been given the value >1234 (by, for example, WHO EQU >1234) then NEG @WHO will negate the word at address 1234 hexadecimal. In fact, you could use a number in the operand instead of the symbol — NEG @ >1234 or NEG @4660 would have the same effect.

e)   Indexed Memory Addressing

The fifth and final general addressing mode is effectively a combination of the Symbolic and Workspace Register Indirect addressing modes, and enables you to "index" into tables (for example) by treating the operand address as the sum of two values — a fixed "base" address and an "offset" from that base which can be varied to enable different entries in the table to be accessed. The effect is somewhat similar to using subscripts (the "offsets") to access elements in a one-dimensional array in TI BASIC. Most other processors (such as the Z80) have dedicated Index Registers which are used for holding offsets and are difficult to use for anything else. The 9900 allows you to use any of your Workspace Registers (with the

exception of R0) as an "index register", and the "base" address is again embedded in the instruction as a symbol. So to write such an operand, you write the symbol as in (d) and follow it with the name of your chosen index register in brackets, e.g. @LABEL(R4) or @START(R2). The operand address is derived by the 9900 by adding the contents of the index register to the value of the symbol (which as I said is part of the instruction). So if you. have set up a table in memory and labelled the start of it with the symbol CODE, then you can access any entry in that table by simply loading a register, say R1, with the appropriate value and using @CODE(R1) as your operand. Setting R1 to zero will pick up the first entry in the table (remember, it is as if "OPTION BASE 0" were in effect), whilst setting R1 to 2 will pick up the second word (also, remember that all addresses refer to bytes, and words may only start on even byte addresses). This resembles quite closely the technique of using an index I to extract CODE(I) from an array CODE in TI BASIC.

Well, that's all I have to say for the moment on the subject of addressing modes. I hope you can see that it's very important to know how the different modes work and why they are useful. With the Workspace Register modes (a-c) it's really just a question of a register holding your data or a pointer to it, while the Symbolic and Indexed modes are really best thought of as variable names and simple arrays a la BASIC. Anyway, there will be a few more meaningful (I hope!) examples flying around when I finally get round to going through the TMS9900 instruction set. I plan to do that in Part 3, so stick around and maybe we'll actually get down to writing a program or two.

If you have any questions or comments on this series then write. My address is: **Flat 2, 12 Belmont, Lansdown Road, BATH BA1 5DZ.**

LOGO JOTTINGS

Betty Lumley

A short appreciation of TI's LOGO language by a primary school teacher. TI LOGO has two capabilities:

1. Drawing with an 'arrow' called a TURTLE
2. Designing scenes within which objects called SPRITES can move.

The language is simple, interesting, and well within the capabilities of the average 8 year old. Younger children with the right motivation and a little help have, in my experience, gained a great deal from using LOGO.

The TURTLE

The Turtle can be 'taught' to draw specific designs, and the sequence of instructions for one design is called a PROCEDURE. For example, to draw a square as a continuous line you would use: Forward 20 steps, Turn Right 90°, Forward 20 steps, Turn Right 90°, Forward 20 steps, Turn Right 90°, Forward 20 steps, Turn Right 90°, Stop. Call the square a BOX, and instructions become:

TO BOX FD20 RT90 FD20 RT90 FD20 RT90 FD20 RT90 END

On typing BOX the computer will draw a box. Children soon see the need for the quicker

REPEAT 4 (FD20 RT90)

Variable size can be introduced by

TO BOX N REPEAT 4 (FD:N RT90) END

Every time BOX is encountered the computer will execute it; consequently including BOX in a procedure will introduce the idea of RECURSION (or SELF - REFERENCE):

TO BOX REPEAT 4 (FD20 RT90) RT45 BOX END

The latter will produce squares rotated through 45° about the bottom left-hand corner. It will carry on until the error message OUT OF INK AT LEVEL 2 occurs. This introduces the need for tests. These are easy to incorporate:

TEST HEADING = 0 IFF BOX   (where IFF means IF FALSE)

So,

TO BOX REPEAT 4 (FD20 RT90) RT45
        TEST HEADING = 0
        IFF BOX
        END

will produce a pattern ending where it started.

The SPRITES

All the techniques for PROCEDURES can be used but with the exciting addition of scenery and movement. 16 colours can be summoned by name or number.

LOGO contains 5 predefined shapes (plane, rocket, truck, ball, square) but a further 27 can be designed using a grid and keys (no hexadecimal calculations!) These are the SPRITES.

Sprites have to be 'talked to'; they must be given a name or number, colour, heading, speed, and place on the screen. For example:

TELL SPRITE 1
CARRY : TRUCK
SETCOLOR : RED
SXY 20 20

A red truck will appear on screen when TRUCK is typed. To get motion, you use:

TO MOVE
SET HEADING 90
SET SPEED 40

Type TRUCK MOVE and the truck will move to the right. If other sprites are on screen, typing their name or number followed by MOVE will make them move to the right.

Sprites can be addressed in groups: TELL (1234) SET COLOR : BLACK will set

sprites 1, 2, 3 and 4 to black.

Using recursion and tests, many effects can be produced – jumps, zig-zags, colour changes, etc., etc.

Scenery is made using grids to design shapes called TILES. These can be placed on screen at will, using procedures to colour large or small areas.

LOGO has sound: BEEP produces one note for trucks etc., and morse can be produced by using BEEP with the WAIT command – WAIT sets up a length for the sound (or previous activity).

If there is an error in the program – for example a missing space between say BUS and MOVE – the computer prints TELL ME HOW TO following by the offending command:

TELL ME HOW TO BUSMOVE

I have used LOGO to teach 4 year old children, and they have absorbed the idea of what it means to program and how to instruct the machine to do what they want. This inter-action is invaluable and has caused older children to introduce their own programs, and to do this they have had to follow through the procedure and think logically. In the process they have also learned a great deal about angles and lines while designing beautiful patterns.

One of the main aims of education should be to help children to think logically and to adapt past experience; this LOGO does, and the children are enthralled and enjoy every minute.

They prefer to program rather than go out to play and even have to be turned out at the end of school!

_____

OPTIMAL EFFICIENT PROGRAMMING 3
=================================
IF'ZZZZZZ   OR THE WASPNEST

THOSE OF US WITH EXT.BASIC MAY
KNOW THE VALUE OF LOGICAL OPERA-
TORS AND MAY HAVE FAMILIARIZED
THEMSELVES WITH BOOLEAN ALGEBRA.
THOSE THAT HAVE NOT MAY COME
ACROSS PROGRAM LINES THAT MAKE
NO SENSE.
THIS ARTICLE I HOPE PROVOKES FUR-
THER INTEREST INTO THIS MATTER.

IN ORDER TO PREVENT ANY MISUNDER-
STANDING FIRST LET'S SETTLE A
FEW THINGS.
A BINAIRY NUMBER IS A CONSECU-
TIVE STRING OF DIGITS 0 AND 1
(BITS).EACH POSITION REPRESENTS
A STEPPED UP EXPONENTIAL VALUE
OF 2 STARTING WITH 0 RIGHT MOST.
SO: $0001_2 =1$. AS $2^0 =1$.
AND $0010_2 =2$. AS $2^1 =2$.
THE SMALL FIGURES TO THE LOWER
RIGHT INDICATE THE NUMERICAL SYS-
TEM.(10 DECIMAL.2 BINAIR.)
THE LEFT MOST POSITION INDICA-
TES THE SIGN: 0 FOR + 1 FOR -.
NOTE: IT'S CUSTOMARY TO DROP
THIS 0 WHEN DISCUSSING POSITIVE
NUMBERS ONLY.

IN THIS ARTICLE,TERMS LIKE FLAG
AND VALUE HERE DEFINED TO MEAN:
-FLAG: CONTROL VALUE IN'IF'
TRUE -1 FALSE 0.
-VALUE: DECIMAL VALUE OF AN EQUA-
TION OR NUMERICAL VARIABLE.
ALTHOUGH THE FLAG VALUES ARE DE-
CIMALS ALSO LET'S JUST CONSIDER
THEM TO BE CONTROLS.

NOTE: THE FACING TABLES WERE LIM-
ITED TO 8 FOR PRACTICAL REASONS.

TABLE 1-1C   LOGIC 'AND'
TABLE 2-2C   LOGIC 'OR'
TABLE 3-3C   LOGIC 'XOR'
TABLE 4-5A   LOGIC 'NOT'

GENERAL

NOTE THAT THE CONDITIONS IN THE
UPPER AND LOWER 2(TABLES 1-3) DO
DIFFER.
THE HEADINGS INDICATE A VALUE OR
FLAG TABLE.WHERE NESSECARY THE
PRECEDING CONDITION IS GIVEN.
1'IN ALL CASES C=0 TO START.
2'C IS'T THE SUM,BUT THE RESULT
OF BINAIRY TREATMENT OF THE DECI-
MAL VALUES IN A AND B.
3'THIS TREATMENT IS INDUCED BY
THE LOGICAL OPERATORS.
4'WHEN A BINAIRY STRING CONTAINS
ONE OR MORE 1'S THE FLAG IS -1.

'AND' TABLE 1.
BINAIRY CONDITION: WHENEVER A'1'
IN A AND B APPEARS IN CORRESPON-
DING POSITIONS A'1' IS TRANSFER-
RED TO C.

A=3.	1000.	A=6.	0110.
B=3.	0011.	B=3.	0011.
	----		----
C=0.	0000.	C=2.	0010.
FLAG	0	FLAG	-1

'OR' TABLE 2.
BINAIRY CONDITION: ANY'1' IN A
OR B IS TRANSFERRED TO C.

A=8.	1000.	A=6.	0110.
B=3.	0011.	B=3.	0011.
	----		----
C=11.	1011.	C=7.	0111.
FLAG	-1	FLAG	-1

'XOR' TABLE 3.
BINAIRY CONDITION:ANY'1' IN A
AND B NOT DUPLICATED IS TRANSFER-
RED TO C.

A=8.	1000.	A=6.	0110.
B=3.	0011.	B=3.	0011.
	----		----
C=11.	1011.	C=5.	0101.
FLAG	-1	FLAG	-1

THE FLAG TABLES 1A,2A AND 3A
SHOW THE CONTROL VALUE WHEN'IF'C
APPEARS FOLLOWING THE GIVEN'LET'.

THE TABLES 1B,1C,2B,2C,3B AND 3C
SHOW THE RESULTS WHEN THE SAME
RELATIONAL EXPRESSION IS USED IN
THE'IF'-STATEMENT.
WHEN THE STATEMENT IN'IF'WOULD
HAVE BEEN:A AND B TABLES 1 AND
1B WOULD HAVE BEEN IDENTICAL.

NOW TREATMENT IS DIFFERENT.
THE FIRST TERM READ IS: C=A
CONSIDERING: A=0 B=0
AS C WAS 0 TO START IN ALL CASES
C=A IS TRUE :-1. BINAIR 1111.
B=0   BINAIR 0000.

'AND'TREATMENT   0000.

TABLE 1C   FLAG   0

LET'S TAKE ANOTHER EXAMPLE
TABLE 2B : A=3 B=5
C=A IS FALSE: 0. BINAIR 0000
B=5   BINAIR 0101

'OR' TREATMENT   0101
            DECIMAL 5

TABLE 2C   FLAG   -1

'NOT'TABLE 4.
BINAIRY CONDITION: ALL BIT'S ARE
REVERSED.   EXAMPLE:
A=0   BINAIR 0000.
'NOT'TREATMENT   1111.
            DECIMAL-1
TABLE 4A   FLAG   -1

TABLE 4B.
ATTENTION: THE STATEMENT NOW IS:
IF C=NOT A   EXAMPLE:
A=4   BINAIR 0100

'NOT'TREATMENT   1011
C=0   0000.
C=NOT A FALSE   0
TABLE 4C FLAG   0

TABEL 5.
LIKE 4B BUT PRECEDING CONDITION
ALSO IS C=NOT A
THIS SUPERFLUOUS TABLE ILLUSTRA-
TES THE TRUTH OF 4B AS THAT IT
SHOWS THE RESULT OF THE REVERSED
ACTION.

MAY BE THE NUMBER LOGIC SHOWS
YOU THE WAY INTO SOLVING THE RID-
DLE OF THE BASIC'OR'S 'AND'S ETC.

GOOD LUCK AND BYE NOW.
BILL.

```
TABEL 1                    TABEL 2                    TABEL 3
VALUE C=A AND B            VALUE C=A OR B             VALUE C=A XOR B
A !          B            A !          B             A !          B
  ! 0 1 2 3 4 5 6 7 8       ! 0 1 2 3 4 5 6 7 8        ! 0 1 2 3 4 5 6 7 8
-------------------------  -------------------------  -------------------------
0 ! 0 0 0 0 0 0 0 0 0     0 ! 0 1 2 3 4 5 6 7 8     0 ! 0 1 2 3 4 5 6 7 8
1 !   1 0 1 0 1 0 1 0     1 !   1 3 3 5 5 7 7 9     1 !   0 3 2 5 4 7 6 9
2 !     2 2 0 0 2 2 0     2 !     2 3 6 7 6 7 10    2 !     0 1 6 7 4 5 10
3 !       3 0 1 2 3 0     3 !       3 7 7 7 7 11    3 !       0 7 6 5 4 11
4 !         4 4 4 4 0     4 !         4 5 6 7 12    4 !         0 1 2 3 12
5 !           5 4 5 0     5 !           5 7 7 13    5 !           0 3 2 13
6 !             6 6 0     6 !             6 7 14    6 !             0 1 14
7 !               7 0     7 !               7 15    7 !               0 15
8 !                 8     8 !                 8     8 !                 0
```

```
TABEL 1A                   TABEL 2A                   TABEL 3A
FLAG IF C                  FLAG IF C                  FLAG IF C
C=A AND B                  C=A OR B                   C=A XOR B
A !          B            A !          B             A !          B
  ! 0 1 2 3 4 5 6 7 8       ! 0 1 2 3 4 5 6 7 8        ! 0 1 2 3 4 5 6 7 8
-------------------------  -------------------------  -------------------------
0 ! 0 0 0 0 0 0 0 0 0     0 ! 0-1-1-1-1-1-1-1-1     0 ! 0-1-1-1-1-1-1-1-1
1 !  -1 0-1 0-1 0-1 0     1 !  -1-1-1-1-1-1-1-1     1 !   0-1-1-1-1-1-1-1
2 !    -1-1 0 0-1-1 0     2 !    -1-1-1-1-1-1-1     2 !     0-1-1-1-1-1-1
3 !      -1 0-1-1-1 0     3 !      -1-1-1-1-1-1     3 !       0-1-1-1-1-1
4 !        -1-1-1-1 0     4 !        -1-1-1-1-1     4 !         0-1-1-1-1
5 !          -1-1-1 0     5 !          -1-1-1-1     5 !           0-1-1-1
6 !            -1-1 0     6 !            -1-1-1     6 !             0-1-1
7 !              -1 0     7 !              -1-1     7 !               0-1
8 !                -1     8 !                -1     8 !                 0
```

```
TABEL 1B                   TABEL 2B                   TABEL 3B
VALUE IF C=A AND B         VALUE IF C=A OR B          VALUE IF C=A XOR B
A !          B            A !          B             A !          B
  ! 0 1 2 3 4 5 6 7 8       ! 0 1 2 3 4 5 6 7 8        ! 0 1 2 3 4 5 6 7 8
-------------------------  -------------------------  -------------------------
0 ! 0 1 2 3 4 5 6 7 8     0 !-1-1-1-1-1-1-1-1-1     0 !-1-2-3-4-5-6-7-8-9
1 !   0 0 0 0 0 0 0 0     1 !   1 2 3 4 5 6 7 8     1 !   1 2 3 4 5 6 7 8
2 !     0 0 0 0 0 0 0     2 !     2 3 4 5 6 7 8     2 !     2 3 4 5 6 7 8
3 !       0 0 0 0 0 0     3 !       3 4 5 6 7 8     3 !       3 4 5 6 7 8
4 !         0 0 0 0 0     4 !         4 5 6 7 8     4 !         4 5 6 7 8
5 !           0 0 0 0     5 !           5 6 7 8     5 !           5 6 7 8
6 !             0 0 0     6 !             6 7 8     6 !             6 7 8
7 !               0 0     7 !               7 8     7 !               7 8
8 !                 0     8 !                 8     8 !                 8
```

```
TABEL 1C                   TABEL 2C                   TABEL 3C
FLAG IF C=A AND B          FLAG IF C=A OR B           FLAG IF C=A XOR B
A !          B            A !          B             A !          B
  ! 0 1 2 3 4 5 6 7 8       ! 0 1 2 3 4 5 6 7 8        ! 0 1 2 3 4 5 6 7 8
-------------------------  -------------------------  -------------------------
0 ! 0-1-1-1-1-1-1-1-1     0 !-1-1-1-1-1-1-1-1-1     0 !-1-1-1-1-1-1-1-1-1
1 !   0 0 0 0 0 0 0 0     1 !  -1-1-1-1-1-1-1-1     1 !  -1-1-1-1-1-1-1-1
2 !     0 0 0 0 0 0 0     2 !    -1-1-1-1-1-1-1     2 !    -1-1-1-1-1-1-1
3 !       0 0 0 0 0 0     3 !      -1-1-1-1-1-1     3 !      -1-1-1-1-1-1
4 !         0 0 0 0 0     4 !        -1-1-1-1-1     4 !        -1-1-1-1-1
5 !           0 0 0 0     5 !          -1-1-1-1     5 !          -1-1-1-1
6 !             0 0 0     6 !            -1-1-1     6 !            -1-1-1
7 !               0 0     7 !              -1-1     7 !              -1-1
8 !                 0     8 !                -1     8 !                -1
```

```
TABEL 4                    TABEL 4B                   TABEL 5
VALUE C                    VALUE IF C=NOT A           VALUE IF NOT C
C=NOT A                    A ! 0 1 2 3 4 5 6 7 8      C=NOT A
A ! 0 1 2 3 4 5 6 7 8      ----------------------     A ! 0 1 2 3 4 5 6 7 8
                             ! 0 0 0 0 0 0 0 0 0      ----------------------
C !-1-2-3-4-5-6-7-8-9                                   ! 0 1 2 3 4 5 6 7 8

TABEL 4A                   TABEL 4C                   TABEL 5A
FLAG IF C                  FLAG IF C=NOT A            FLAG IF NOT C
C=NOT A                    A ! 0 1 2 3 4 5 6 7 8      C=NOT A
A ! 0 1 2 3 4 5 6 7 8      ----------------------     A ! 0 1 2 3 4 5 6 7 8
                             ! 0 0 0 0 0 0 0 0 0      ----------------------
C !-1-1-1-1-1-1-1-1-1-1-1                               ! 0-1-1-1-1-1-1-1-1-1
```

OPTIMAL EFFICIENT PROGRAMMING 4
=================================

AFTER THE PROBLEMATICAL/IF/THIS
TIME THE EFFICIENT/IF/OR BETTER
STILL THE/IF↓LET/CONVERSION.

TI 99/4(A) BESIDES ASCII CODES
RECOGNIZES A NUMBERED LIST OF TO-
KENS.
TOKENS ARE MEMORY ADDRESSES FOR
RESERVED WORDS USED IN RUN MODE.
THE LIST SHOWS BUILT IN FUNTION
CODES AND STATEMENTS LIKE:FOR,TO,
THEN,DATA,STR$,ETC.
THE INTERESTING ASPECT IS THAT
TOKENS USE ONE BYTE EACH AS GOES
FOR ASCII CODES.
THIS KNOWLEDGE ALLOWS US TO COM-
PARE THE REQUIRED MEMORY TO SOME
EXTEND.IN #7 OF THIS SERIES I'LL
GIVE YOU A BETTER BASE FOR BYTE
CALCULATION.

IN O.E.P.#2 I USED THE E.B.EXAM-
PEL: IF NOT X=1 TO SHOW YOU THE
BASIC EQUIVALENT:IF (X=1)=0.
INTERNALLY THIS WILL READ SOME-
THING LIKE: 132-200-7-183-88-190
-49-182-190-48
132=IF,200=LENGHTCODE,7=LENGHT,
183=(,ETC.

THE QUESTION HERE WAS WHETHER
/X=1/ EQUALS 0.SO WE CAN WRITE
IF X<>1 OFCOURSE ONLY INCASES
X IS EITHER 0 OR 1 !
THIS WOULD READ 132-200-4-88-191
-192-49 SAVING 3 BYTES.
MORE IMPORTANT SAVINGS CAN BE AT-
TAINED WHEN USING THE FOLLOWING
METHOD.

OFTEN/IF/IS USED TO POSITION A
FLAG
A1)IN BASIC:

```
300 IF M=1 THEN 303
301 H=G
302 GOTO 304
303 H=-G
```

A2)EX.BAS:
```
300 IF M=1 THEN H=-G ELSE H=G
```

THE RESULTS ARE THE SAME.
IN BOTH CASES A/LET/COULD HAVE
BEEN USED.

A3)LET:
```
300 H=G+2*G*(M=1)
```

THE ADVANTAGE FOR BASIC IS VERY
EVIDENT.BUT FOR EX.BAS.THERE IS
AN INTERESTING ADVANTAGE ALSO.
THE LIMITATIONS FOR/IF/S IN MUL-
TISTATEMENT LINES.ARE LIFTED AND
THAT'S REALLY SOMETHING AS MANY
A M.STMNT.LINE FINDS A PREMATURE
END DUE TO AN/IF/.
HOWEVER BE CAREFULL: STATEMENTS
FOLLOWING THESE/LET/S SHOULD NOT
BE IN CONFLICT WITH EITHER THE
/THEN/ OR /ELSE/ EXIT.

ALTHOUGH MANY VARIATIONS ARE POS
SIBLE.TWO MAIN GROUPS ARE WORTH
CLOSE ATTENTION.THE CHANGE OF
SIGN(EXAMPLE A)AND:0 OR <>0 (B)

B1)IN BASIC:
```
300 IF M=1 THEN 302
301 M=0
```

B2)EX.BAS.
```
300 IF M<>1 THEN M=0
```

B3)LET
```
300 M=1+(M<>1)
```

DON'T START USING/IF↓LET/S WITH-
OUT TESTING.DIFFERENT SHAPES ARE
NESSECARY WHEN M CAN HAVE VALUES
OTHER THAN 0 OR 1

THE WAY TO HANDLE THE IF↓LET CON
VERSIONS IS:TO WRITE THE FORM IN
ELSE.LEAVE SPACE THEN WRITE DOWN
THE CONDITION IN BRACKETS.FILL
THE GAP WITH THE VALUE REQUIRED
AND THE ARTHMETIC/OPERATOR.
SOMETIMES REVERSING THE RELATION-
AL EXPRESSION SIMPLIFIES THINGS.

NOW WE HAVE REACHED WHAT I CONSI
DER THE MOST INTERESTING CONVER-
SIONS.AS PETER BROOKS USES THEM
IN HIS KEYBOARD CONTR.PLOTTER I
ASSUME HE ELABORATELY EXPLAINED.
I THEREFORE WILL BE VERY CONCISE

MANY TIMES ROWS OF/IF/S ARE EN-
COUNTERED EACH FOLLOWED BY SOME-
WHAT SIMILAR STATEMENTS WITH A
/GOTO/IN LAST POSITION.MORE THAN
OFTEN THESE LINES CAN BE REBUILT
TO FORM A SINGLE BASIC LINE.

C1)IN BASIC:

```
300 CALL KEY(A,K,S)
301 IF S=0 THEN 300
302 IF K<>5 THEN 305
303 Y=Y-1
304 GOTO 313
305 IF K<>0 THEN 308
306 Y=Y+1
307 GOTO 313
308 IF K<>2 THEN 311
309 X=X-1
310 GOTO 313
311 IF K<>3 THEN 300
312 X=X+1
313 CALL HCHAR(Y,X,42)
314 GOTO 300
```

C2)EXT.BAS.(302-312)
```
302 IF K<>5 THEN IF K<>0 THEN IF
  K<>2 THEN IF K=3 THEN X=X+1 ELS
E 313 ELSE X=X-1 ELSE Y=Y+1 ELSE
  Y=Y-1
```

C3):LET (302-312)
```
302 Y=Y+(K=5)-(K=0)
303 X=X+(K=2)-(K=3)
```

WHEN A=1 KEYS/EDXS/FOR A=2 /IKMJ/
ARE OPERATIVE. THE WHOLE BECOMES
BETTER STILL WHEN THE ADDITIONAL
KEYS:RCZW/OR/O.NU/ARE REQUIRED.

```
302 Y=Y+(K=4)+(K=5)+(K=6)-(K=0)-
(K=14)-(K=15):: X=X+(K=2)+(K=4)+
(K=15)-(K=3)-(K=6)-(K=14)
```

AS TRANSLATING MEANS DOUBLE WORK
I SUGGEST YOU A COURSE IN DUTCH.
SO HERE WE GO:
TOT DE VOLGENDE KEER.
I.E."UNTILL NEXT TIME"
BYE   BILL.

THREE — DIMENSIONAL NOUGHTS AND CROSSES : DOCUMENTATION

1380-1400	ASSIGN TO LEVEL, ROW, AND COLUMN VARIABLES THE KEY — PRESSES ACCEPTED
1410	TRANSFORM 3D CO-ORDINATES TO 1D
1420	SECOND VALIDATION
1430	BRANCH IF MOVE ACCEPTED
1440-1500	NAUGHTY, NAUGHTY!
1440-1490	CLEAR TEXT LINE AND OUTPUT 'CHEAT!' WITH ADMONISHING TONES
1500	JUMP TO HUMAN PLAYER'S MOVE (1260)
1510-1560	MOVE ACCEPTED. PUT ON BOARDS, UPDATE RELEVANT LINESUM ELEMENTS, RESET CO-ORDINATES AND DISPLAY THEM, AND CLEAR TEXT OUTPUT LINE
1570-1610	INITIATE SEARCH FOR NEXT MOVE TO BE MADE. LOOP THROUGH ALL LINESUMS, COMPARING WITH SUCCESSIVE STRATEGY VALUES. IF A MATCH IS FOUND, BRANCH OUT OF LOOP
1620-1740	IF THE LOOP IS COMPLETED, THEN NO BRANCH HAS OCCURRED. THIS MEANS THAT NO MATCH HAS BEEN FOUND, SO THE GAME IS THEREFORE A DRAW. REQUEST A DECISION FROM THE HUMAN PLAYER WHETHER FURTHER GAMES ARE TO BE PLAYED, WAIT FOR KEY — PRESS AND VALIDATE. IF NO FURTHER GAMES, THEN CLEAR THE SCREEN AND END
1750-1800	CLEAR TEXT LINE, RESET LINESUM ARRAY, RESET CO-ORDINATES TO ZERO AND DISPLAY THEM. JUMP TO RE-ENTRY POINT (960)
1810	IF JUMP WAS MADE BECAUSE OF MATCH WITH STRATEGY ARRAY ELEMENT 1 (16) I.E. HUMAN PLAYER WIN, BRANCH TO 2100 TO NOTIFY
1820-1970	NOTIFY PLAYER OF COMPUTER'S MOVE, FIND FIRST EMPTY POSITION IN STORED WINLINE CO-ORDINATE SEQUENCE, PLACE ON SCREEN WITH NOTIFICATION
1980	IF BRANCH WAS BECAUSE OF 'NORMAL' MATCH THEN BRANCH AGAIN TO SECOND RE-ENTRY POINT (1230)
1990-2090	COMPUTER HAS WON. NOTIFY VIA TEXT LINE, RUN THROUGH WINNING CO-ORDINATE SEQUENCE, DISPLAYING CO-ORDINATES, AND CHANGING CORRESPONDING SHAPES ON SCREEN BOARDS TO INDICATE WINNING LINE. JUMP TO 1650 TO REQUEST A DECISION ON FURTHER GAMES
2100-2130	HUMAN PLAYER HAS WON. CLEAR TEXT LINE, PRINT NOTIFICATION, AND GO TO RE-ENTRY TO REQUEST DECISION ON FURTHER GAMES

```
100 CALL CLEAR
110 OPTION BASE 1
120 RANDOMIZE
130 DIM B$(76),C$(76),L$(76),U$(
64)
140 DEF R4=INT(RND*4)+1
150 GOTO 500
160 FOR F=1 TO LEN(M$)
170 CALL HCHAR(G,F+2,ASC(SEG$(M$
,F,1)))
180 NEXT F
190 RETURN
200 L=0
210 R=0
220 C=0
230 CALL HCHAR(18,16,ASC(STR$(L)
))
240 CALL SOUND(100,990,0)
250 CALL HCHAR(20,16,ASC(STR$(R)
))
260 CALL HCHAR(22,16,ASC(STR$(C)
))
270 FOR E=1 TO 100
280 NEXT E
290 RETURN
300 L=INT((U-1)/16)+1
310 I=U-(L-1)*16
320 R=INT((I-1)/4)+1
330 C=I-(R-1)*4
340 RETURN
350 DATA 16,3,12,8,4,2,1,0
360 DATA 23420461,27384916,26391
352,22430164,22261830,38423446,2
3271931,39433547,22232124,383937
40,26272528
370 DATA 42434144,22380654,27431
159,26421058,23390755,22390556,2
6430960,23380853,27421257,224202
62,23430363
380 DATA 26381450,27391551,22271
732,38433348,23262029,39423645,0
1160611,49645459,04130710,526155
58,04491934
390 DATA 16613146,01521835,13643
047,01612141,04642444,13492537,1
6522840,49525051,61646263,010402
03,13161415
400 DATA 49615357,52645660,04160
812,01130509,16643248,13612945,0
1491733,04522036,40245608,442860
12,47316315
410 DATA 46306214,41255709,37215
305,34185002,35195103,55545653,5
9586057,59556351,58546250,070608
05,11101209
420 DATA 11071503,10061402,40443
648,47464845,41374533,34353336,2
4282032,31303229,25212917,181917
20
430 DATA 04293537434851,21435968
,22436067,0131333843475
2,1748586
5,13296568,16316567,19475365,184
85766,15316668
440 DATA 14296667,20475466,03313
639444850,23445668,24445567,0229
3440444749,25517576,05355976,073
36076,27527376
450 DATA 09375875,04050913172125
,01070916192227,09385373,1139577
5,03051115182327,02071114202425,
11405473
460 DATA 27507475,05365674,07345
574,25497374,26517172,06335972,0
8356072,28526972,10395871,020610
13192326
470 DATA 03081016172428,10405369
,12375771,01061215202128,0408121
4182226,12385469,28507071,063456
70,08365570
480 DATA 26496970,02303339414551
,23415964,24416063,033235404146 5
2,19455861,13306164,16326163,174
65361,20455762
490 DATA 15326264,14306263,18465
462,01323437424550,21425664,2242
5563,04303638424649
500 FOR A=1 TO 8
510 READ S$(A)
520 NEXT A
530 FOR A=1 TO 76
540 READ C$(A)
550 L$(A)="0"
560 NEXT A
570 FOR A=1 TO 64
580 READ U$(A)
590 NEXT A
600 CALL SCREEN(12)
610 CALL CHAR(96,"00422418182442
")
620 CALL CHAR(97,"00183C66663C18
")
630 CALL CHAR(104,"0042241818244
2")
640 CALL CHAR(112,"0000001818")
650 CALL COLOR(9,16,6)
660 CALL COLOR(10,16,10)
670 CALL COLOR(11,14,16)
680 M$="     3-D NOUGHTS & CROSSE
S"
690 G=2
700 GOSUB 160
710 M$="      PETER BROOKS 1980"
720 G=3
730 GOSUB 160
740 M$="LEVEL: 1    2    3    4"
750 G=7
760 GOSUB 160
770 M$="ROW:"
780 G=10
790 GOSUB 160
800 M$="      1234 1234 1234 123
4"
810 G=13
820 GOSUB 160
830 M$="           COLUMN:"
840 G=14
```

```
850 GOSUB 160
860 M$="LEVEL:"
870 G=18
880 GOSUB 160
890 M$="ROW:"
900 G=20
910 GOSUB 160
920 M$="COLUMN:"
930 G=22
940 GOSUB 160
950 G=16
960 FOR A=9 TO 12
970 FOR B=9 TO 24 STEP 5
980 CALL HCHAR(A,B-1,A+40)
990 CALL HCHAR(A,B,112,4)
1000 NEXT B
1010 NEXT A
1020 GOSUB 200
1030 L=R4
1040 R=R4
1050 C=R4
1060 M$="MAY I GO FIRST ?(Y/N) "
1070 GOSUB 160
1080 CALL KEY(0,K,S)
1090 IF (K=78)+(K=89)THEN 1100 E
LSE 1080
1100 CALL HCHAR(16,3,32,30)
1110 IF K=78 THEN 1260
1120 U=L*16+R*4+C-20
1130 M$="MY MOVE"
1140 GOSUB 160
1150 GOSUB 230
1160 FOR I=1 TO 5
1170 CALL HCHAR(8+R,5*L+C+3,112)
1180 FOR J=1 TO 100
1190 NEXT J
1200 CALL SOUND(-10,1000,0)
1210 CALL HCHAR(8+R,5*L+C+3,96)
1220 NEXT I
1230 FOR A=1 TO LEN(U$(U))STEP 2
1240 L$(VAL(SEG$(U$(U),A,2)))=ST
R$(VAL(L$(VAL(SEG$(U$(U),A,2))))
+1)
1250 NEXT A
1260 GOSUB 200
1270 CALL HCHAR(16,3,32,30)
1280 M$="YOUR MOVE"
1290 GOSUB 160
1300 FOR A=1 TO 3
1310 CALL KEY(0,K,S)
1320 IF S<=0 THEN 1310
1330 IF (K<49)+(K>52)THEN 1310
1340 CALL HCHAR(16+2*A,16,K)
1350 D(A)=K-48
1360 CALL SOUND(-20,990,0)
1370 NEXT A
1380 L=D(1)
1390 R=D(2)
1400 C=D(3)
1410 U=L*16+R*4+C-20
1420 CALL GCHAR(8+R,5*L+C+3,H)
1430 IF H=112 THEN 1510
1440 CALL HCHAR(16,3,32,30)
1450 M$="CHEAT!"
1460 GOSUB 160
1470 CALL SOUND(300,139,0)
1480 CALL SOUND(450,110,0)
1490 CALL HCHAR(16,3,32,30)
1500 GOTO 1260
1510 CALL HCHAR(8+R,5*L+C+3,97)
1520 FOR A=1 TO LEN(U$(U))STEP 2
1530 L$(VAL(SEG$(U$(U),A,2)))=ST
R$(VAL(L$(VAL(SEG$(U$(U),A,2))))
+4)
1540 NEXT A
1550 GOSUB 200
1560 CALL HCHAR(16,3,32,30)
1570 FOR A=1 TO 8
1580 FOR B=1 TO 76
1590 IF S$(A)=L$(B)THEN 1810
1600 NEXT B
1610 NEXT A
1620 CALL HCHAR(16,3,32,30)
1630 M$="GAME DRAWN"
1640 GOSUB 160
1650 FOR E=1 TO 100
1660 NEXT E
1670 CALL HCHAR(16,3,32,30)
1680 M$="ANOTHER ?(Y/N) "
1690 GOSUB 160
1700 CALL KEY(0,K,S)
1710 IF (K=78)+(K=89)THEN 1720 E
LSE 1700
1720 IF K=89 THEN 1750
1730 CALL CLEAR
1740 END
1750 CALL HCHAR(16,3,32,30)
1760 FOR A=1 TO 76
1770 L$(A)="0"
1780 NEXT A
1790 GOSUB 200
1800 GOTO 960
1810 IF A=1 THEN 2100
1820 M$="MY MOVE"
1830 GOSUB 160
1840 FOR F=1 TO 7 STEP 2
1850 U=VAL(SEG$(C$(B),F,2))
1860 GOSUB 300
1870 CALL GCHAR(8+R,5*L+C+3,H)
1880 IF H=112 THEN 1900
1890 NEXT F
1900 GOSUB 230
1910 FOR I=1 TO 5
1920 CALL HCHAR(8+R,5*L+C+3,112)
1930 FOR J=1 TO 100
1940 NEXT J
1950 CALL SOUND(-10,880,0)
1960 CALL HCHAR(8+R,5*L+C+3,96)
1970 NEXT I
1980 IF A>2 THEN 1230
1990 CALL HCHAR(16,3,32,30)
2000 M$="I WIN!"
2010 GOSUB 160
2020 FOR F=1 TO 7 STEP 2
2030 U=VAL(SEG$(C$(B),F,2))
2040 GOSUB 300
```

```
2050 GOSUB 230
2060 CALL HCHAR(8+R,5*L+C+3,104)
2070 CALL SOUND(100,770,0)
2080 NEXT F
2090 GOTO 1650
2100 CALL HCHAR(16,3,32,30)
2110 M$="YOU WIN!"
2120 GOSUB 160
2130 GOTO 1650
```

VARIABLE + SUBPROGRAM LIST + FUNCTIONS

OPTION BASE: 1
DEFINED FUNCTIONS : R4
NUMERIC : A, B, C, F, G, I, K, L, R, S, U
STRING : M$

ARRAYS :   NUMERIC :   D() 3 elements (NB not explicitly DIMensioned,
                                          so 10 reserved)
            STRING :   B$() 76 elements
                            C$() 76 elements
                            L$() 76 elements
                            S$() 8 elements
                            U$() 64 elements

SUBPROGRAMS : CHAR
                    CLEAR
                    COLOR
                    GCHAR
                    HCHAR
                    KEY
                    SCREEN
                    SOUND

LISTed SIZE (Extended BASIC) : 5424 bytes
RUNning SIZE (Extended BASIC) : 8655 approximately

_____

POSTSCRIPT

I reckon I could set myself up with a crystal ball, my predictions are so good. I forecast the need for a Postscript in the Introduction, and lo!, here it is!

A few things have happened which are worth mentioning. One is that the book noted in the introduction (LEARNING TO USE THE TI 99/4A) is now available (11th April) and I have placed my order for a copy. The company, READ-OUT BOOKS & SOFTWARE, have adverts in a number of mags; I've placed mine through COMPUTING TODAY'S book service. Delivery should be within about 14 days apparently.

The two new weeklies, PERSONAL COMPUTER NEWS and HOME COMPUTING

WEEKLY, have been on the go now for long enough that comparisons might be made between them and the existing weekly, POPULAR COMPUTING WEEKLY, and I'm afraid that PCWeekly loses hands down, on just about every count. PCN (offspring of the original PCW) has the most professional format, and is better than some of the monthlies. HCW carries articles by one DAVID GRAY (TIHOME member ??) — although his LoRes (Low Resolution) graphics program seemed unnecessarily complex to me (perhaps deliberate — these magazines don't know up from down half the time, and if you can couch your simple statement in sufficient jargon and 'doublespeak', they'll pay quite well!), — and seem to be heading for the 'regular feature' position. PCN on the other hand published recently a 'clinic' on the 4A, and although the information was of no real use (it kept telling you why you couldn't use PEEK and POKE on the 99s), at least it was something a little more elevated than the usual 'I think the 13 decimal places more than makes up for the lack of just about every other standard facility' crud.

PCN has also been publishing occasional letters from STEPHEN SHAW (you could have used the original A = A + 8, Stephen!), so I'm currently getting both, and PCWeekly is about to get the axe.

GARY HARDING sent his article in having used the impact-dot-matrix printer which TI are now shoving out (a mere £400), and although the quality is quite good you probably won't get to see it if Tidings is now being printed (haven't seen a copy yet).

Incidentally, I hear rumours that the printer TI chose ceased production shortly after they chose it, which seems in character....

Puzzle: why are NORTHWISH LTD in PCWeekly (24-30 March) advertising for 'arcade games' — type programs with an illustration of a 4A + Speech Synthesider — yet it DOESN'T list TI programs in its request ???

I've been playing with CP/M — you may have seen mentions of this operating system in the mags — and I find it difficult to believe that ANYONE would willingly put up with such sheer unadulterated @@@@. It has to be THE most nauseous system I have ever seen; how on earth did it manage to get accepted? Editing on the 99s is sheer heaven when compared with CP/M, and no matter how much it may have been patched to make it a little more 'user-friendly', nothing can disguise the fact that it is worse than terrible.

GARY HARDING has sent me an amendment to his DISASSEMBLER program which appears in the software collection, and once I have checked the changes with him (his original carries different line numbers from the library version) I will publish them for any member who has received a copy and wishes to update it.

If I had enough room here, I would paraphrase SCIENCE DIGEST on the subject of the latest 'intelligent cars' — you know, like British Leyland's Maestro only better. There is this American serial, you see, called Knight Rider, and if you haven't seen it count your blessings. The thing is, it's not that technology has caught up with fiction, it's that fiction is now trailing BEHIND technology because our fiction writers are not with it; odd how the serial's car can't locate itself while FORD's new machines will use GEO-STATIONARY EARTH ORBIT SATELLITES to pinpoint their positions to within ridiculous limits (and so will many others, Japanese especially).

The 3D Noughts & Crosses listing might leave you a little bemused if you haven't been following the tortuous explanations over the last few months — it might leave you bemused even then — but if you take the time to key it in you should have fun with it for at least ... ooh, five minutes ... Before anyone raises an eyebrow, the version in the software collection is an advanced one, using a more powerful algorithm. It is continually under revision, and I have another, more compact, and possibly even faster, version in the pipeline. Anyone with a Mini-Memory Module might care to try their hand at translating the 1980 game into 9900 code...

Well, that doesn't even begin to cover all the subjects buzzing around in my head, but time and space permit no more this issue. If you think that some of the buzzing might manage to help you out with a problem, drop me a line. Now that postage costs have got even dafter, a return stamp would be much appreciated, but don't bother about providing an envelope — you won't be able to guess just how much woffle your letter will generate....

Good programming,

Pete Brooks.

Post-Post Script

Well, it's the end of April and I've finally managed to see a copy of Tidings — the format of which I like very much, incidentally, — and unfortunately there have been many slips 'twixt cup and lip, as our contributions have been retyped by the printers, and a whole array of omissions, mis-spellings, and complete hashes have been carefully produced. Poor old Mike O'Regan lost out most of all because he'd used different type faces as an illustrative example, and of course the printing cocked that up.

Apart from the occasional omissions of symbols and letters from my articles (which shouldn't cause too much grief) there was one GLARING error. Bill van Kerkoerle's contributions were halved, as an entire sheet of A4 with example program and full explanation was left out!

I have included another copy of the sheet for Paul's printers and hope that it helps to clear any mysteries up.

The page lies between paragraphs 2 and 3 on page 20 ov V3.1 Tidings (and 'look' ought to have been 'loop' — a little gem courtesy of the printers), and the article ought to have had a '1' at its head, to go with the '2' which appears lower down on that page — Bill sent two articles, if you remember.

One other omission was an asterisk (multiply symbol) at the bottom of page 13, between the '10' and the '('. The omission of these symbols (and the relational operators) is quite serious, and the printers should be made aware of their importance. And my omission was to fail to explain the equation's operation in this issue! The number you will never see is a zero, due to the fact that the program causes the value which 'A' represents to cycle from 1 to 10, and no matter what value you initially set A to, it will eventually enter the cycle again. (If you can't see the use of such a daft equation, don't worry, you soon will!)

Finally of course was the conflicting information about my abode, caused by my moving house in between publications. The telephone number is spaced wrongly in the contacts page — the code is 0865, and the actual number on my dial is 717985, but so far noone has had any trouble getting through. Can I ask you (as a precaution) not to ring after 10 p.m., please — it would be appreciated.

Those sorting routines looked quite good, and I hope to explain how some of them work soon, as well as giving again the Quicksort, and a 'D-sort', a special form of Bubble sort which is very fast also.

Pete Brooks (yet again).

OPTIMAL EFFICIENT PROGRAMMING

Bill van Kerkoerle

Hello there. My name is Guillaume van Kerkoerle, and I have offered to write a series of articles for Tidings. They will be the equivalent of my contribution to the Dutch Tijdingen. As I am unaware of your experience, for some of you I will no doubt be opening doors which have already been opened! I will take that risk on the assumption that there are parallels between the groups in our two countries — namely that 75% of TI users are still at a Beginner's level.

My subject this time is a result of observing that often the length of a program is inversely proportional to its output. For those like myself not used to typing, shorter programs are a blessing and besides work faster and claim less memory space. Who hasn't come across 'MEMORY FULL' through too large arrays, too little space for ASCII characters, etc.?

Examine the market for RAM Expansion, and you'll come away disgusted with the price, but while more memory may offer greater possibilities, it may not be necessary if you optimise a program's use of memory. For short programs this is overkill, I agree, but generally a good short program will eventually be added to, to make it multi-functional for example, and that is where efficient programming comes in.

Who's Afraid Of For-Next Loops?

The first rule is: never use the variables controlling loops carelessly, as every variable consumes memory. The variable used for a loop can also be used as a counter, a component in an equation, an array subscript, or part of a conditional expression. Its exit value can be used as a choice or flat in subsequent statements, or it can be used again as the control variable in the next loop.

Are you also the victim of the suggestion that a loop may not be jumped into? It may, of course, provided you take care — if you jump in after the FOR you MUST jump out before the relevent NEXT. The best thing is to use the same flag variable for both actions. The following is an example of such a construction. The formula used in line 170 is illustrative only. An explanation of the operation of the routine is also given.

42

```
100 INPUT "CHOICE 1/2/3!":G
105 PRINT "CHOICE G=";G,,,,,,
110 X$="YOU SEE!"
120 IF G<2 THEN 170
130 FOR R=1 TO 3
140 PRINT R;
150 IF G<>3 THEN 170
160 FOR C=1 TO 4
170 A=100*R+C
180 ON G GOTO 240,220,190
190 PRINT TAB(C*7);A*R*C;
200 NEXT C
210 IF G=3 THEN 230
220 PRINT TAB(6);A;TAB(13);A*2;T
AB(20);A*3;TAB(27);A*4
230 NEXT R
240 PRINT A;R;C,X$,,,,,,,,
250 GOTO 100

RUN
CHOICE G= 1

   0   0   0         YOU SEE!


RUN
CHOICE G= 2

  1     100     200     300     400
  2     200     400     600     800
  3     300     600     900    1200
300    4   0      YOU SEE!


RUN
CHOICE G= 3

  1     101     204     309     416
  2     402     808    1218    1632
  3     903    1312    2727    3648
304    4   5      YOU SEE!
```

Description

Column loop C (160) and Row loop R (130) straddle the formula (170). The value of G will select a route through this routine.

G = 1: Clear passage through both loops. The TRACE would give: 100, 105, 110, 120, 170, 180, 240, 250...... so A = O, R = O, and C = O.

G = 2: Executes the R loop, but jumps into the C loop and out again before NEXT C. TRACE is 100, 105, 110, 120, 130, 140, 150, 170, 180, 220, 230, 240, 250...
The table values are printed (140) and (220) while (240) shows the exit values for A, R, and C.

G = 3: Uses both loops and prints (140), (190), and (240).

I used IF G and ON G on purpose; with a sensible formula or block of formulae this construction allows:

1. Single use of the formula/formulae
2. Printing a 4-item table with 1 choice variable
3. Printing a 1-item table with 2 variables at choice

I do realise that I have not covered everything I could about FOR NEXT statements, but I don't think it is a good idea to go into detail first time round. There will be, therefore, a follow-up series on IF starting with Logical Operators (Boolean Maths) concluding with conversions for statements of the type IF A = B THEN LET C = D, the latter being one of my personal favourites.

2.   The IF statement is a very useful one, which few of us are sufficiently skilful to do without. For the programmer, IF is a valuable asset saving time and work. The TI BASIC IF may transfer control to line numbers, whereas under Extended BASIC both THEN and ELSE may contain a string of statements as well, although there are a few which may not be used — FOR NEXT is one. This is logical, considering that any program flow must be able to find its way from the FOR to the NEXT unconditionally. Let us use a pictorial representation to clarify the execution:

nnnn IF X = 1 THEN 140 ELSE 180:—

THEN 140

ELSE
180
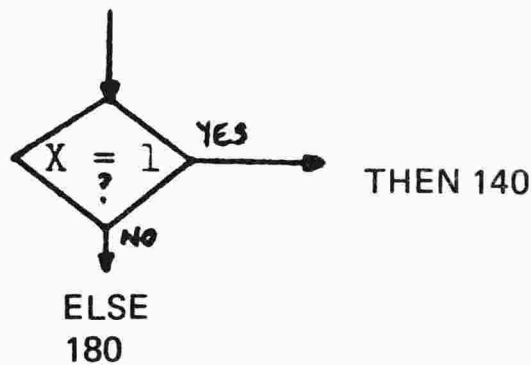
In other words, if (X = 1) is TRUE (i.e., the value of X is 1), the flag value is —1, and if it is FALSE, the value is 0. For the rest of this item, to reduce typing, let B stand for TI BASIC and EB for Extended Basic.

Logical Operators

EB: provides 4 logical operators. They are: NOT, XOR, AND, OR. They are available in B but need to be written differently — just remember that —1 is TRUE and 0 is false.

EB: IF NOT X = 1 THEN ...   i.e., if X = 1 is FALSE, then ...
      B: IF (X = 1) = 0 THEN ...

EB: IF X = 1 XOR Y = 1 THEN ...   i.e., branch if one or the other is TRUE
                                    but not both.
      B: IF (X = 1) + (Y = 1) = —1 THEN ...

EB: IF X = 1 AND Y = 1 THEN ...   i.e., branch only when both are
                                    TRUE
      B: IF (X = 1) + (Y = 1) = —2 THEN ...

EB: IF X = 1 OR Y = 1 THEN ... i.e., branch if either is TRUE
      B: IF (X = 1) + (Y = 1)   = 0 THEN ...

X, Y, or 1 may be replaced by any expression. The same goes for relational expressions and arithmetic operators. Be careful though when using multiply and divide and remember that the computed result must comply with your requirements. You can make all kinds of interesting combinations, but test them fully before using them. For example, if you use IF (X = 1) + (Y = 1) + (Z = 1) = —2 THEN ... it means that 2 out of the three must be TRUE.

There is another construction in EB: IF X = 1 THEN IF Y = 1 THEN IF Z = 1 THEN 150 ELSE 180 ELSE 260 ELSE 110. Pictorially this is explained thus:—
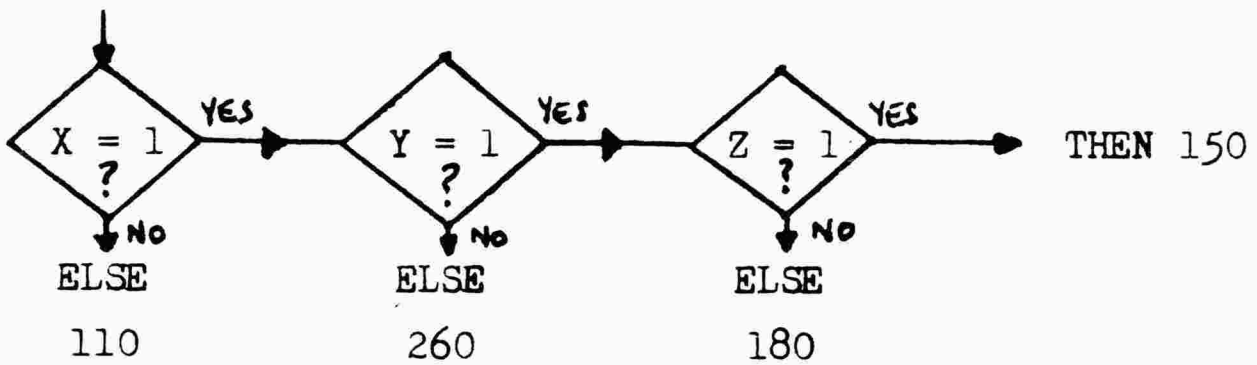
To implement this kind of construction in B requires that it be broken down thus:—

```
100   IF X < > 1 THEN 110
101   IF Y < > 1 THEN 260
102   IF Z =  1 THEN 150 ELSE 180
```

Note that in the EB line above if the line following is numbered 110 the trailing ELSE can be omitted.

Risks with IFs

Sometimes bugs can drive you mad and take the fun out of programming entirely. Often the fault lies hidden in an erring IF statement — not even seasoned programmers are immune to this. Using EB for example, using C = A AND B and IF C THEN ... is not the same every time as IF A = B THEN ... When C contains a non-zero value (specifically − 1), this will work, but the result of C = A AND B is not always the same as the result of A AND B in an IF statement. This problem is caused by Boolean arithmetic and is luckily not one which B programmers can experience.

2021 note: Parts 3 and 4 of this article are on the page with the number 32 on the bottom.

DIAMOND MINER – LANTERN SOFTWARE

As you have no doubt noticed, DIAMOND MINER is not a particularly short program to key in! I do hope, however, that some of you will make the effort since several of the routines demonstrated in this program are quite interesting and have wider applications than simple (but addictive!) games. Some points made will not be new to the more experienced users but I hope that there will be something for everyone within its tight little body. Anyway here goes with the listing and please forgive the (temporary) lack of printer.

```
100     RANDOMIZE
110     CALL CLEAR
120     CALL SCREEN(2)
130     D = O
140     X = 1
```

```
150      Y = 32
160      B$ = "SCORE="
170      C$ = ""              (i.e. two " )
180      DEF R1 = INT(RND*23)+1
190      DEF R2 = INT(RND*32)+1
200      DEF A$(D) = B$ & STR$(D) & C$
210      FOR I=1 TO 16
220      CALL COLOR(I, 12, 1)
230      NEXT I
240      CALL CHAR(97, "3C42A599665A2418")
250      CALL COLOR (9, 16, 1)
260      CALL CHAR(104, "1866181C2A081436")
270      CALL COLOR (10, 9, 1)
280      CALL HCHAR(1, 1, 126, 736)
290      CO = 1
300      RA = R1
310      RB = R2
320      CALL GCHAR(RA, RB, G1)
330      IF G1 < > 126 then 300
340      CALL HCHAR(RA, RB, 97)
350      CALL KEY(O, K, S)
360      A = X
370      B = Y
380      IF CO+1 = 736 THEN 390 ELSE 410
390      CALL SOUND(-1000, 5000, 2)
400      GOTO 280
410      IF (K= 83)*(Y>1) THEN 510
420      IF (K = 68)*(Y<32) THEN 540
430      IF (K = 69)*(X>1) THEN 570
440      IF (K = 88)*(X<23) THEN 600
450      CALL GCHAR(X, Y, G)
460      CALL HCHAR(A, B, 32)
470      CALL HCHAR(X, Y, 104)
480      IF G = 97 THEN 630
490      IF G = 32 THEN 710
500      GOTO 350
510      Y = Y-1
520      CO = CO+1
530      GOTO 420
540      Y = Y+1
550      CO = CO+1
560      GOTO 430
570      X = X-1
580      CO = CO+1
590      GOTO 440
600      X = X+1
610      CO = CO+1
```

2021 note: "Smart quotes" in these listings should be replaced with TI's ")

```
620      GOTO 450
630      FOR I = 110 TO 1110 STEP 500
640      CALL SOUND(-1000, I, 2)
650      NEXT I
660      D = D+1
670      FOR I = 1 TO LEN( A$(D) )
680      CALL HCHAR(24, I+1, ASC(SEG$(  A$(D) , I 1)))
690      NEXT I
700      GOTO 300
710      FOR I = 210 TO 110 STEP -10
720      CALL SOUND(-500, I, 2)
730      NEXT I
740      FOR I = 2 TO 20
750      CALL SOUND(-1000, 1000, I)
760      NEXT I
770      CALL HCHAR(X, Y, 32)
780      IF D> HD THEN 790 ELSE 800
790      HD = D
800      B$ = "H-SCO="
810      C$ = "PRESS ANY KEY TO PLAY"
820      FOR I = 1 TO LEN( A$ (HD) )
830      CALL HCHAR(1, I+1, ASC(SEG$ (HD) , I, 1)))
840      NEXT I
850      CALL KEY(O, K, S)
860      IF S THEN 100 ELSE 850
870      END
```

The rules of DIAMOND MINER are quite simple. Using the E, X, S and D keys you must cause your man to burrow through the rock and collect as many diamonds as possible. Once a tunnel has been dug, however, you may not re-enter or cross that tunnel again so that it is most important to plan out your route very carefully. If you successfully clear all the rock then you lose the last diamond but start all over again in a new section of the mine. Due to lack of time (the game had to be written in a single evening) and to the need to keep the program relatively simple, no routine has been added to speed up the placing of the diamond, resulting in an occasional lengthy pause at the end of each page. However, I'm sure that someone out there can come up with something that will do the job and send it in!

VARIABLES in the program are assigned like this:— D is the number of diamonds collected :: HD is the highest score to date :: X and Y are the miner's position on the screen :: A and B are the old position of the miner :: CO counts the number of rocks burrowed away.

In lines 180 to 190 DEF has been used to generate random numbers. The effect of using DEF is that every time the computer encounters either R1 or R2 it performs the operation on the right side of the equals sign. This is particularly useful in handling cumbersome equations as well as in saving precious memory space.

Line 200 I have included not so much as a good approach to printing strings or saving memory but simply to demonstrate the little known fact that DEF will handle string functions as well as numeric.

What line 200 actually does is to make A$ (D) equal to the string B$, plus the current value of D (turned into a string by STR$), plus the string C$. This means that by changing the values in B$, D, or C$ you can use A$(D) to represent any string you choose!

210 to 230 colors the character set yellow.

240 to 270 defines the miner, and the diamond and colors them.

280 uses a single command to fill the screen.

300 to 340 makes RA and RB equal to fixed random numbers. The position RA, RB is then checked using GCHAR for a rock. If one is present then a diamond is printed if not then some more random numbers are selected.

350 to 500 is the main loop, that is, the part of the program which communicates with the user and therefore governs its performance. The first thing it does is to call up a keyboard scan to see if you've pushed anything. The rest of the loop then runs through a series of operations based on the results of the scan. If these operations take a long time then the response to pressing a key will be sluggish and vice versa. The trick, therefore, is to cram as much as you can into the main loop whilst keeping its running speed as high as possible.

Within the loop, lines 410 to 440 demonstrate the fact that Stephen Shaw has advertised so well recently that by using the ' * ' or the ' + ' in an IF THEN ELSE statement you can add the boolean functions AND and OR. So, for instance, line 410 could be read:— IF K = 83 AND Y>1 THEN 510.

480 to 490 check and branch out of the loop if you have hit a diamond or a space.

500 to 620 alter values of X and Y according to which key has been pressed and adds 1 to CO as each move must correspond to the loss of one rock.

670 to 690 sees the first occurrence of A$(D) our user defined string function, here it is being used to print the new score after a diamond has been collected. Line 670 first finds the total length of the string, then, for each letter or number in the string line 680 adds one to the horizontal print position and then prints the character. This is the best method of achieving a ' print at ' command in TI—BASIC.

710 to 760 show two sound commands embedded in FOR NEXT loops. This is a very effective way of generating sound and worth experimenting with, since by varying the complexity of the loops it is possible to produce some very interesting effects. 740 to 760 is the chime loop by the way.

770 to 840 demonstrates the print at in operation again but this time printing the high-score and using different values for B$ and C$.

860 shows an example of how the TI evaluates expressions. If an expression such as A >B is true then that expression is given the value −1, if it is false then the value is 0. Therefore the statement IF S THEN... is

really saying IF S is true THEN... or IF S is not equal to O THEN...

One final point on the subject of saving memory. When you first start programming this doesn't seem very important but it is useful to get into the habit so that when you come to write a long program you are already skilled in the art of byte saving. You may have noticed that in every FOR NEXT loop I have used the same variable ' I ', this has saved a quite impressive 40 bytes!

S.W. SORSBIE,
LANTERN SOFTWARE

AFTERTHOUGHTS

---

by: Roger Culshaw   84, Meadow Way   Walton   Stone   Staffs   0785-812016

Welcome back to adventureland. Having read my last article I hope you have all now indulged in some fantasy role-playing. In describing the various games in my last article I did omit 'Treasure Trap'. This is a role playing game with a difference. You play it for real!

To play, you must journey to a castle in Cheshire. There you will be equipped with weapons and armour and dispatched on your quest into the depths of the castle's dungeons. The monsters you meet fight back and it is a good idea to wear brown trousers. I play every other month and my 'other self' Orlando Niblock has so far been killed twice. If you fancy killing Orcs for real give me a ring (evenings) and I'll give you the details.

Now back to our saga of the computerised adventure. Having seen 'Tron' I now have marvellous visions of little men running around inside my trusty TI although knowing my luck some of them are running in circles! In the last issue we discussed the fundamentals of role playing games and discussed the limitations of achieving the same result on the computer.

Looking more closely at the main elements of an adventure program we can identify four main components (see Fig 1).

Player:
We need to keep a record of our trusty adventurer. How many wounds has he got? What spells does he know? What objects is he carrying? What weapons has he got? How many days provisions? etc etc.

Environment:
Where is the adventure to take place? What is the description?

Objects:
What objects are in the environment
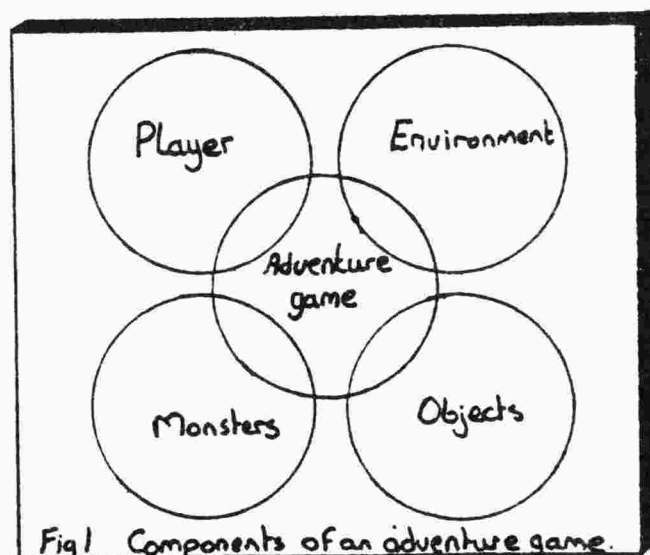
Fig 1   Components of an adventure game.

that our player can pick up, use, or which have an effect on him/her or things he/she meets.

Monsters:
All adventures need nasties (and not so nasties) for our player to meet and to interact with.

Lets start by looking at the environment and deciding just what information is relevant or adds to the game. There are of course a vast number of environments we could choose. This is one of the tremendous joys of adventure games. The scenarios are only limited by our imagination. But somehow I'm going to have to write something in this article so lets simplify the situation. Basically adventures can take place outside or inside. Outside or wilderness adventures in my experience take a lot of planning and for the time being I shall ignore them. That leaves us with inside adventures. It doesn't really matter what we are inside of, it could be a castle, a temple, a spaceship or a mountain. Our inside adventure will inevitably be made up of a number of individual encounters that take place in rooms. At last I get to what I wanted to talk about ROOMS!!!!!

Let's consider the items of information that might have an effect on our trusty adventurer and that can also relate to the room. First of all can he see anything at all. The room may be dark. In this case the program would have to prevent him seeing a description until he has lit a torch. (In some role-playing games elves, like our friend here have infravision and may well be able to see even if the room is lit). Once we have established that something can be seen we must then provide some sort of description.

How big is the room: This may be important to enable mapping or if the program takes position of players, monsters and objects into account.

Are there any doors? There must be the door through which we entered (unless we were teleported here!) but are there any others? If there are where are they? Are they open or locked? Are they secret and therefore not currently visible? What room do we enter if we pass through the door?

What is the general description? The general description of the room and its occupants is a very important part of an adventure program. It is only through description that the author of the adventure can fire the imagination of the player. Unfortunately written or graphical descriptions gobble up memory. In my opinion this is the most serious limitation of computer driven adventure games. For an example lets just compare a written description in one of my D&D scenarios with its probable computer counterpart. I will set the scene first as it does give an opportunity to illustrate a typical scenario....

"Crunior Woodleaf was on his first adventure. He had been a guard at the temple of Truth for the past four years. Two days ago a party travelling towards the temple had been ambushed by the powers of darkness. This had proved to be a disaster since the party was bringing a powerful artifact which was to aid in the eternal struggle against darkness and evil. Crunior was a member of the party who had been sent to recover it. After tracking the ambushers for several days, the party had found themselves moving deeper and deeper into the Mountains of Morn, a

land noted for the manifestations of beings of darkness. Eventually the trail led to a large cave in the side of one of the larger mountains. (Here we leave the 'outside' adventure and enter the 'inside') The party waited till nightfall and entered (Which was silly since it was heavily guarded and they were supposed to search for a secret way in which would have been safer). They were lucky and after many near scrapes and some ferocious battles in which several of their party were lost, they eventually found a secret door to the room where they suspected the relic ought to be. (OK. Here comes the room description....)

The secret door opens silently after pressing the catch. A wiff of stale air blows out from the dimly lit room beyond. The room is 15ft NS and 10ft EW your door is in the centre of the south wall. The room is lit by a strange red glow that seems to be coming from the eyes of a statue in front of the north wall. The statue appears to be that of a humanoid with a disproportionate head. Indeed the head is about twice as large as it ought to be. The room has a musty smell and there are several large spiderwebs in the corners of the room. On both east and west walls hang tapestries. These depict some horrific scenes of torture and fighting and all seem to show a black humanoid figure involved in the most brutal of acts. An air of forboding and evil pervades this room. In front of the statue laying on the floor is a gold casket. Somehow the gold shines, of its own accord, repelling the red light. As you enter the room the eyes of the statue turn towards you....."

That is some 150 words. A horrific thought for our computer if we multiply that by say 30 rooms. There wouldn't be much memory left for our program. True we could use some form of text compression techniques and I hope to discuss some of these in later articles but for the time being we must prune the number of words and at the same time make sure that we convey all the useful information...

"You enter a 10' x 15' room.
Visible: Black humanoid statue with red eyes
                Golden casket.

Statue looks at you ... What now?"

It certainly loses something in the translation. We could add more words but this would be at the expense of the number of rooms. So we can store descriptions

but they will have to be abbreviated. (Or plug plug NOT contained in the computer but stored on paper and referred to by the computer. This is the concept behind my own adventure games to be released by Time Travels very shortly).

So size, doors, description what else do we need? Well we need to know what objects are currently in the room and what other occupant is in the room and perhaps even whether that occupant is still alive. This is important if our program is to allow the adventurer to return to a room where they have have killed something.

So all in all a lot of information we need to know about just one room. Let's make a rough total of the array elements we would need to specify the room... lit(1) size (2) doors (4) description no (1) objects (4) occupant (1) occupant status (1).

14 elements per room (This is only a rough idea we could for example store the status of the monster in the array that describes the monster but it does begin to give us some idea of the memory we need). If we assume that we have 30 rooms then we need 420 element positions just to store the room data. Remember that one array element takes 8 bytes of memory and you can see we have used nearly a ¼ of our memory just on this room array. We still need to add the description, player array, object array and monster array and of course the program!

Arrays

Well from the above it should be obvious that ordinary use of arrays is not possible. What can we do instead? Well one traditional solution is to use string arrays to store numerical values. Remember that the reason one numerical array element occupies 8 bytes is because a numerical variable is stored to 14 digit accuracy by our computer. That is not really needed to store whether or not a room is lit. '1' could represent lit where '0' could represent dark. If we stored this value in a string array then we could always establish the value of that array position using the VAL(X$) command. Indeed we could store all the information about one room in a single element of a string array and use the SEG$(.....) command to establish the individual items.

e.g.   10 ROOM$(1)="12311112312120934"
        20 LIT=VAL(SEG$(ROOM$(1),1,1))
        30 PRINT LIT
        RUN
        1

How would this affect memory usage. Well our room did use 14x8 bytes to store its information, that is 112 bytes. It takes 2 bytes to store a letter (this is how it seems to me, can anyone tell me why?) so counting the letters in our string array element we have now used 17x2 bytes or 34, this is at the expense of a slightly more complex program line to get the values we need.

There is another way we can reduce the amount of memory needed even more. Lets examine again just what is stored in by the computer when it stores a number. Well first of all there are 14 digits, there is also a sign + or − and also the exponent eg $\times 10^{32}$ etc. We could represent a lit room by making its numerical value + and

an unlit room as a —ve number. Using the SGN(x) would then give us the required value. Now what about the 14 digits. Look at the diagram below which shows a 14 digit number but that number is a composite number made up from others.

Here I am storing 10 items of information in one numerical variable position which as we have said before occupies 8 bytes. The problem is decoding that number when we want to use the details of the room. Consider this program.

```
10 X=12345678901234
20 I=INT(X/10↑13)
30 X=X-(I*10↑13)
40 A=INT(X/10↑12)
50 PRINT I,A
```
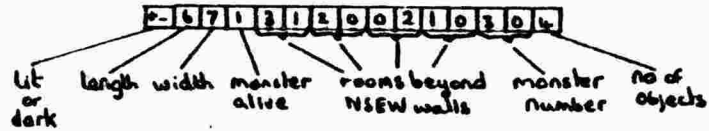

It would not be too difficult to write a For..Next loop to do this and generate all 14 digits separately. Why don't you try and then try to write a program that produces the first 4 single digits and the 5 double digit numbers as per my original example. I'll give you my version of this in the next issue. Hope you understand all that.

In the next issue we will conclude our look at arrays and maximising the memory use and having examined the environment in outline we will take a closer look at our player.

Comment:

I am currently amazed at the numbers of computers coming on to the market at the moment. With the recent launch of Oric, Lynx and upcoming electron and now TI launching their new machine I don't know where the customers all are. Anyway at least if the cheaper peripherals work on the 99 then I might be able to afford a printer. Our esteemed leader rang me after presenting my last article to tell me about the 1 A=A+8 2 GOSUB 1 routine to determine memory used. (Thanks Paul) This is a bit quicker than my maximum array size idea in the last issue. However it does only tell you how much space your program occupies not how much it uses whilst it runs. This has led me to experiment using both methods to come up with some rule of thumb. I haven't got one yet except to say that it seems to depend on the length of string variables and the number of nested GOSUBS. Can anyone tell me any more about what extra space the computer needs to run a program?

It now looks as if a few more people are producing cassette games. I counted five adverts in the last copy of C&V games (inc my own) so perhaps things are looking up. Is anyone else playing seventh empire and can you beat a score of —890!!!!!
                    Bye for now Roger

TI NEWS

Robin Frowd, TI UK

Firstly, my congratulations to TI HOME for yet another advance in the layout of TIDINGS. As I sit here looking at the last edition (March), I see that the type face is the same all the way through — indeed, a noteable first; this edition might almost become a collectors' item.

As many of you will remember from March's edition of TIDINGS, I announced a new computer, new software, new peripherals including a new Cassette Recorder, as well as there being some information on the new Compact Computer, the CC40, so I am sorry if this month does not live up to that as much, but never the less, there are some new packages arriving from the United States over the next few weeks:

TI WRITER

For those of you who want full word processing, TI Writer is for you. It is a complete implementation that one journalist compared to a main frame version.

MICROSOFT MULTIPLAN

At last, a full spread sheet program (like VISICALC), this time from the stable of MICROSOFT. Again, this is a complete implementation giving 63 columns and 256 rows. I have had a good look at it and it is very impressive indeed.

OTHELLO

The classic board strategy game brought to your TV screen. A game for more than one person, or just you versus the computer. I, personally, prefer these types of games to plain arcade games since they hold their interest longer.

ALPINER

A superb mountain climbing game featuring not one, but 2 voices, (Speech Synthesiser is optional). You have to climb progressively higher mountains with sundry obstacles, like bears, tigers, falling rocks etc., that impede your progress.

TUNNELS OF DOOM

This is a cartridge and cassette package, and was originally referred by us as "Has to be seen to be believed". Now I have seen it, I can tell you that it is a graphics adventure game, and very good too.

For later this year, around August, there will be an inexpensive adapter called the "Hex Bus Adapter" that will allow TI99/4A owners to use the micro peripherals that were designed for use with the CC40. Currently there are 3 of these peripherals:

RS232   A standard interface containing both a serial RS232, and parallel centronics interface. With this you can connect the TI99/4A to a range of printers etc. that are commercially available.

WAFERTAPE   A fast storage device that has many of the advantages of disk systems, but without the high price.

PRINTER/PLOTTER   A 4 colour printer/plotter that will enable you to list your programs, plot graphs and do multi-coloured graphics.

All these peripherals are designed as low cost additions to our computer range. Approximate prices are as follows:

RS232	£119.95
Wafertape	£109.95
Printer	£139.95

(Prices include VAT).

Currently there are 2 special offers in the High Street, one for new owners and one for new and existing owners.

Firstly, any new purchaser of a TI99/4A can claim from Texas Instruments the following items:

A pair of Joysticks
Beginners Basic Tutor
Connect Four Cartridge

(Offer closes 2 July 1983).

These items together are worth over £50.00, so watch out, Mr. Dicks, you may be getting a few more members!

Secondly, and perhaps of more interest to members of TI HOME, anybody buying 6 Solid State Software Cartridges between 30 April and 2 July can claim from us either a new TI Program Recorder, or a Solid State Speech Synthesiser.

I think that is about it for the moment, however, you should find in another part of this excellent magazine a letter from one Kirsty Clift. Kirsty actually works for me, but her letter is a personal one which she wrote off her own back, and perhaps is another example of a Human Being working at TI!

═══
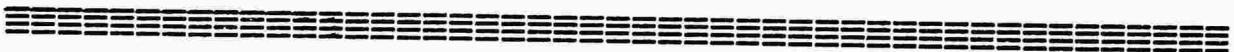
53 Leyburn Close, Urpeth Grange,
Ouston, Co. Durham, DH2 1TD.
Dear Mr. Dicks,   4th May 1983.

I am writing to tell you that I have beaten the score of G. Tapp on the TI Invaders on the Downright Nasty Level. G. Tapp's score was 12,337 in the March 1983 edition of 'Tidings'. My score is 14,154, my previous score was 12,960.

Yours sincerely, Dominic R. Huffer (age 14¾)

* * * * *

257 Monton Road, Monton, Eccles,
Dear Mr. Dicks,   Manchester M30 9PS.

I have recently read the March issue of Tidings. One of the letters I read in it, I saw that Karen Warner (13) had scored 85,770 on the TI Munchman. Like her, I have no joysticks and I scored 147,580 on Munchman by using the keyboard. I would be interested if this is a record.

Yours sincerely, Stephen Leung (14)

Texas Instruments Finds Potential Fault In Its Transformer

DALLAS—Texas Instruments Inc. has detected a potential defect in the electrical transformer sold with its 99-4A home computer that could cause electrical shocks or computer damage.

The company is taking steps to protect current and future owners of the computer against the hazard.

Texas Instruments said the possibility of shock is remote and that "under normal usage, it is extremely unlikely" the hazard could arise However it has suspende shipments of the computer to replace th transformer or include an adapter cord wi the computer that would prevent the haz from arising Texas Instruments also s is discussing with the Consumer Pr Safety Commissi plan to distrib daptor cords tc nown owner 4A

The com

Shocker confined to the US

The Insider column of the March 1983 publication of *Infomatics* contains a short article about Texas Instruments' 99/4A home computer under the heading 'Texas Shocker'.

Reference is made to the various defects attributable to the transformers accompanying that product in such a way as to give the manifestly false impression that the fault lay in *all* transformers supplied with the computer, both in the US and Europe. This view is further reinforced by the wording of the last sentence of the article which states that a safety adaptor cord "may go to *all* users".

The European version of the transformer of both 200 and 240 volts conforms fully to all European standards and presents no risk to consumers. The problem identified in the US is confined solely to the 110 volt version which is only available for use in North America.

V Sonsino,
Texas Instruments,
Bedford.

Texas shocker A 99/4A home comput transformers can ov shocks and damag while a safety adap

Only in the US

RAMBLES by Stephen Shaw

This edition of Rambles has been rushed off towards the end of April 83, without seeing the February issue which is somewhat late, in the hope that the April issue won't be quite so delayed!

Consequently this section is shorter than usual and some news and comment has had to be held over to the June issue of TIdings.

New members — who may have been waiting quite some time for the February issue! — should note that the adverts placed elsewhere in this issue are by me in a private capacity and not as a section of the club!

If you have any problems, please write in and where possible they will be dealt with in TIDINGS, but if you want a direct reply PLEASE enclose an SAE and understand it will be a few days before you have a reply!!!!

If you have made some discoveries, do please share them!

— CASSETTE OPERATION: At certain times you are asked to press R, C or E. Paul Karis has kindly shown me that these keys are scanned at many other times also.

When the message 'Press cassette stop and then press enter' appears, you may instead press R C or E with the usual effect.

If you ENTER 'SAVE CS1' by accident, you may cancel the request by pressing E when the messages:
'Rewind... then press ENTER' or
'Press...Record...then press ENTER' appear.

This information can save some time under certain circumstances!

— I have based a few programs on games found in GAMES and PUZZLES magazine, which went into liquidation twice and resurfaced as THE GAMER, which has also gone under — for the third and last time! However a completely new magazine is now out, available on subscription only — Bimonthly £5.50 p.a. 'Games Gazette' contains interesting articles and reviews on board games. These can provide useful information on general gaming techniques. The publisher is: Games Gazette, 4 Old Barn Road, Leybourne, MAIDSTONE, Kent.

— Budding games publishers considering advertising their programs in the commercial magazines may be interested in the response rates I've had — Computer & Video Games comes out well ahead with an average of 150 plus responses per month. Of the weeklies, Home Computing Weekly does very well but Popular Computing Weekly not so good.

— Did you see TI's TV ads in May? Pretty expensive. They seem to be establishing a lot of dealerships at present with video shops — if you shop there, you can probably help the owners with the information YOU have: why not offer your services as a part time technical adviser? At least one TIHOME member is already supplementing

his income this way!

— Lots of rumours floating around about the 99/2 and possible price decreases on the 4A. Nothing reliable at present but it will be interesting to see what TI is selling this coming Christmas and at what price!

FEBRUARY has been a time of bugs —

My apologies to anyone who tried to phone me in early Feb — the high winds in these parts broke the wire to our 'phone, and although you will have received a ringing tone, my phone didn't ring.

If any members have purchased STAR TREK by Norton Software from me (before February 23rd 1983) please would they amend the following line — 2320 should be: KLENRGY(KLFL) = KLENRGY(KLFL)−INT((0.25+0.25*RND) *WENRGY/KLDIST)

(The final bracket has to be moved otherwise the program bombs if you use phasors on the Klingons. This program has been on sale in the US for 2 y ears and here for nearly a year — and only now does someone complain about it bombing!!!)

Anyone with the program in EXTENDED BASIC should return it for a free replacement.

Rumours of new TI prices were correct — but we did NOT expect increases! The new TRADE price for some peripherals is higher than the retail price of suitable alternatives (80 col printer, disk drive).

The new HEX BUS peripherals may be available when you read this. Your Computer has suggested £150 for the new printer — it is US$200 in the USA so that seems reasonable. I'm not too sure about £50 for the TI PROGRAM RECORDER. REM: The 99/4A REQUIRES an adaptor to use the new peripherals — catalogue number PHP-1300 with a US price of $60.

Have any members made the 99'er Program 'Cyber Dice' work as described? It appears to have a fair number of bugs sprinkled around — not to stop it running, but to stop it working as described.

Cassette Tapes — I use tapes from a number of sources, mainly DATA ASSETTE & (up til now!) BIBI MAGNETICS. On my last BiBi order we had lots of admin problems, and when the tapes finally arrived, a full 30% failed to record anything! That is a very high failure rate, and one I cannot afford to risk again — it wastes a lot of time recording a program and then finding the tape blank.

REVIEW: Industrial quality joysticks now available!

How does this differ from the TI Controller? It uses microswitches instead of the coated plastic TI use, and gives a more reliable operation, especially on diagonals. The shaft is metal — unlike the plastic TI shaft which is liable to fracture.

The speed of operation is controlled by the key-scan, but the more reliable operation does give a slightly better user response. You can use quite gentle pressure — you don't have to force it to operate! and thus reduce operator fatigue.

The sample reviewed had a short lead, but I understand that this is to be longer on production models. The firing button, placed on the front left surface — near the base — was quite difficult to operate and it is likely to be moved to another position. The controller is in a hefty plastic box of good construction (those four microswitches are quite large!) and is likely to outlast the TI Controllers. Recommended for serious play. This controller can be used with one hand by placing it on a sturdy surface — so you COULD write a program using two of them at the same time (impossible with TI Controllers!) but note that a second controller is an optional extra! ONE joystick only supplied....

Supply: £19.50 from some dealers or by post from: Arcade Hardware, 211 Horton Road, Fallowfield, Manchester M14 7QE. (post free).

MISCELLANY

Popular Computing Weekly has described the price of the 99/4A in the States as £98 — compared to £95 for the launch price of the Spectrum. (In the US!).
At the time of writing, the Spectrum was £125 in the UK (INC 15% VAT!!!)

Can you see TI reducing the 99/4A to £130? It's not far off...

The originally quoted figure for the 99/2 (£70-£80) seems too high considering the competition, but compares well with the US launch price (£66 excluding sales tax). Last due date heard — July.

* * * * * * * * * * * * *

Taylor-Wilson have been criticised by another writer in an earlier TIDINGS. I can only agree — no response at all to two orders sent to them.

* * * * * * * * * * * *

A director of a firm of 'Computer Specialists' thinks that the US 60Hz mains may stop US programs on cassette loading in the UK — and also thinks computer programs sold on tape should load on any cassette player regardless of any system difficulties — and also thinks it reasonable to demand a full refund after reporting a tape would not load THREE MONTHS after purchase..... hmmmm.

* * * * * * * * * * * *

MAIL ORDER BUYING   (Applicable to everything, not just computers!)

a.   IS YOUR ADDRESS STATED ? (LEGIBLY!!!!)
b.   HAVE YOU ENCLOSED PAYMENT? (if required)

It is difficult to know what to do when you receive cash with no return address... a cheque can at least be returned to the bank. An empty envelope is a mystery!

c.   Are you sure you have ordered what you want? Very few firms supply on approval these days (except for specialist mail order companies). Software is almost never sold on approval due to piracy problems and fairly low margins.

d.   If anything goes wrong — contact the supplier SOON or you may lose any rights you have under the Sale of Goods Act. Be polite and your chances of fair treatment are much better.

If you are immediately rude or tell lies — you stand little chance. Remember your supplier is in business to make money! and give him a fair break.

e.   Problems with computer software are almost always due to a problem recorder

— dirty head, magnetised head, misaligned head, poor speed control. By all means try other recorders, but be sure not to be too long before you report problems. SOME suppliers insist on problems being reported within 7 or 10 days.

Although a supplier is only responsible if the product is faulty, you may be able to obtain a refund if your recorder is unsuitable — just ask nicely (and beware — future supply may be difficult!)

If you have difficulty loading one tape, there is little point asking a supplier for a different program — he may think you are 'trying it on'. No harm in asking for a re-recording of the same program though....

Try to put yourself in the position of your supplier — are YOU being fair to him? If so, he is very likely going to treat you in a fair and generous manner!!

FROM THE U.S.

Samuel D. Pincus writes regarding Assembly language:
"First — the notorious errors in the Editor/Assembler manual. Most of them are minor but there are a few that will kill you. In no case should you plan on using the extended utilities called GPLINK or XMLLNK. They only work under a restricted set of circumstances. For example, the Workspace pointer must point to hex 20BA (called USRWS) and NOT to your own workspace area. In addition, some of the registers in that workspace must have certain values or else these calls will not return to your program. This restriction means that BLWPs will not work in 'auto-start' programs unless you load the special values into the USRWS registers. Unfortunately we have not yet been able to figure out what the correct values should be.

"In addition there is a mis-print in the Extended BASIC equates on page 416 of the manual. The equates for NUMREF is 200C (hex) not 2000 (hex) as stated.

"Another error in the Editor/Assembler deals with disk file handling. It states that a call to a bad device will set bit 2 in the STATUS byte (837C hex) on. In fact, I could not get this byte to mean anything. It never seems to be touched by the file management routines in any way when using the DSRLNK routines to read disk files. I tried a lot of different techniques to see if I could get the error routines to work automatically but never managed to do it. For example, if I cleared byte 837C and then tried to read a non-existant disk drive (i.e. drive 4) the status byte stayed clear. If anyone in the UK finds the scret, let me know.

"By the way, I know a lot of people who are doing great things with the TI editor/ assembler language, but all of them tell me the same thing: avoid using the TI utilities to do things because they may not work and you can drive yourself crazy trying to "debug" your program.

"A very amazing feature of using Assembly language routines to process arrays created by Extended or TI BASIC is that unless you use the routine called STRASG to re-create a string within an array, BASIC will not recognise it! I wrote a high speed Assembler sort to be used by Extended Basic. Instead of re-writing the data in its new locations within the array I merely changed the 'string pointers' for the array without moving the data. This is a normal micro-computer technique which I have used successfully on my TRS80 computer. At the very end of the sort, the

pointers are set to where I have pointed them during the routine. As soon as BASIC did its 'garbage collection' (to free up unused string space) my pointers were reset back to where they had been before I began my sort!

As a matter of fact, the only way I could get BASIC to recognise my sort was to use STRSG in writing out data. Of course, this significantly slows the sort speed. You can imagine how long it took me to 'debug' my program. Every time I would review the sort process, everything looked fine and the data was perfectly sorted. Within a few seconds however the program would act as though the sort never took place! My advice then is that you had better use NUMASG and STRASG when creating data within an assembly language subroutine.

"...An important point to remember is that whenever the E/A manual talks about using BASIC, the author means TI BASIC only. You have to read very carefully through the appendices to see how to interface an assembly language routine to Extended Basic.

"Another thing to pay close attention to when writing assembly language routines called by TI or EXTENDED BASIC is the 'screen offset'. If you look at the VDP RAM maps and the VDP register pointers set up by Basic (Chapter 21.1 in the E/A manual), you will realise that both Basics overlap the pattern table with the screen image table. They get away with this by 'offsetting' the character patterns so that you never need to access a pattern that would reside where the screen image table is located. This means that the pattern for ASCII character 32 is stored in the location where the pattern for ASCII character 128 should be and so on.

In other words, BASIC automatically adds hex 60 to each ASCII character before writing it to the screen. This is why TI BASIC cannot access the full 256 ASCII character set. If your assembler program has been called from BASIC, you must also add hex 60 to each character set up in a DEFM statement, because the character pattern has been loaded in with the offset by BASIC. Of course, there are other ways around this, but adding hex 60 is the easiest method I have come up with."

(Sam Pincus is a contributing editor to 99'er magazine).

****

TI BASIC SPRITES WITH MINI MEM

Following on from last months note on using sprites with MMM, Mr. Pincus points out that in TI BASIC, hex 300 to 31F is used for the colour table, while in Extended Basic it is used for sprite attributes. As only 17 to 32 are used in TI BASIC, there are some 17 bytes available (decimal 768 to 784) which we can use for TI BASIC sprites — as soon as we move out of this area we are playing with the colour table!

(reminder — use

CALL POKEV(768,98,128,163,1,208)   )

for 3 sprites this would be:

CALL POKEV(768,98,128,163,1,20,40,164,1,130,170,165,1,208)

and that last 208 will be at location 780.

Extension past this is possible but can lead to strange results under certain circumstances.

In Extended Basic, the sprite velocities are held at hex 780 to 7ff, and we can POKEV these in TI BASIC too — except that in TI BASIC this area is used for the value stack (unless you have defined characters 97 upwards, in which case their definitions reside here). The velocity of each sprite occupies 4 bytes —the first two refer to the velocity and the last two are used by the computer — they appear to count up from 0 to 15 repeatedly, at a rate controlled by the velocity.

Thus to move our 4 sprites, use:

CALL POKEV(1920,1,1,0,0,128,1,0,0,200,4,0,0)

Then tell the computer it has to move 4 sprites with:

CALL LOAD(−31878,4).

CURSOR COLOR:: Now — referring back to last month, when we redefined the cursor using the MINIMem, lets change the COLOUR!

Try   CALL POKEV(783,240) in a program e.g.

    1.   CALL POKEV(783,240)
    2.   INPUT A$

Notice anything? Now you can have a truly personlised cursor!
The number 240 has to be broken down to hexadecimal format to mean anything — it is FFFF0000.

Hex FFFF is 15 (white as logic 0=1)   (foreground color)
Hex 0000 is 0   (transparent- 1)   (background colour)

* * * * * * * * * * * * *

I received a very interesting letter from Mr. B. Austwick of Coylton in Ayrshire — as his queries will be of interest to many of our readers here are some short replies:

REVIEWS WANTED — of all software on modules or cassettes — I've tried Barry — but very few seem interested! I offered free programs to reviewers — 3 replies and so far no reviews (offer closed!)
I've sent programs to Computer & Video Games — and they can't find anyone to review them! Its all up to you — the reader. If you buy software from any source, why not review it? Send your reviews in to Paul Dicks or Pete Brooks or Mike O'Regan (not me — I'm an interested party!).

Additional tape recorder to add to list of those which are suitable — Ferguson Revolver Radio Cassette c.£29.

Magazine holder — In the last but one issue I recommended the INMAC COPY HOLDER — Barry has suggested you try making your own, by using a BOOTS anglepoise-type lamp — remove the lamp holder and screw a wooden platen to the plate.
(The INMAC looks very like a lamp of this type. The platen holder is of acrylic with a top clip and sliding transparent cursor, adjustable for different thicknesses of copy).

PRINTER — At present you MUST have an RS232 interface to connect any printer. This means you need either —

The BOX, RS232 card, plus printer OR
The hex-bus RS232 peripheral, 99/4A adaptor plus printer OR
The hex-bus 4 colour printer & adaptor (MAY need RS232???)

The hex-bus system comes with the CC-40 and 99/2 — no news of availability yet. An adaptor to use the 99/4A with these is available in the States for about US$70 — it is a TI product so should arrive here sometime.

PRINT AT in Extended Basic allows you to PRINT at any screen location — for example PRINT AT(2,3):"TEST" will print TEST with the first T in row 2, colum 3 (28 column screen).

USING allows you to justify and shorten amounts — in the following example I have used the @ sign, but you really use a hash mark — not available on my typewriter — it is SHIFT 3 on the computer.

PRINT USING "TOTAL @@@.@@":A
If A is equal to 3745.77 an error message is generated — too many digits before the decimal!
If A is equal to 5.876 then the print will be: TOTAL 5.87

USING is only available in Extended Basic.

A form of PRINT AT is available in TI BASIC if you have the PERSONAL RECORD KEEPING or STATISTICS modules inserted — a brochure is available from TI HOME for 50p describing this and other enhancements to TI BASIC using these modules.

PLOT — The 99/4A does have a hi-res mode, and you could write your own utility to PLOT — but you need assembly language to do it — and you therefore need some CPU RAM. As the hi-res screen occupies 12k, this really means you need the 32k RAM expansion. A utility program is available in the states for US$20.
In BASIC you can use a pseudo hi res plot, but it is limited by the number of definable characters and VERY slow.

File processing was dealt with briefly in the last issue.

* * * * * * * * * * * * * * * * *

TI GAMES MODULES:

TI INVADERS: HIGH SCORE:
Have you a high score for TI INVADERS: Can you beat **37,667**.
This score is reported by K.E. Storry of Flitwick, Bedford, at level 2. Mr. Storry asks: 'has anyone got past the 500 on the bonus ship yet — i.e. hit the invisible xxx or got past the wave of flashing ones that come after the flying birds?'

MUNCH MAN: Cheating(!)
Would you like to START at level 20 — with ONE munchkin? You can. To debug a program, a programmer needs devices to enable him to play the higher levels without the boredom of playing through the lower levels first — and the device in Munchman is reported as being still there for YOU to use!

Mr. J.L. Vaughn has reported to CREATIVE COMPUTING (March '83) that you can enter your chosen level as follows:

As soon as the Munch Man screen appears, hold the SHIFT key down and press 8 then 3 then 8 (holding shift down all the time!!) Now input answers to th three screen prompts—

    i)   0,1, or 2 — the ROUND

    ii)   0 to 19 — the SCREEN (0=screen 1 etc)

    iii)   1 to 9 — the number of Munch Men you are playing with.

From the time the screen appears you have no more than 3 seconds to do all that — so don't hesitate.

If you lose, you go back to level 1, but you can use QUIT and repeat the above procedure.

(Mr. Vaughn is a TI Product Support Rep in DeKalb, Illinois)

Although screen 20 (input 19!!) is VERY DIFFICULT, you can WIN your way all the way up to level 60 (? who can ?)

Barry Austwick of Coylton has asked for a software library — a subject which has been dealt with at some length in the pages of POPULAR COMPUTING WEEKLY. Even for more popular computers the software producers are distinctly unhappy — for the 99/4A with its very limited user base, a software library would very quickly halt production of new programs —

Why? With only a small user base, a software author cannot expect large sales (In the December peak, my best selling program sold just 38 copies — the 2nd best was well below that), resulting in somewhat higher prices.

The Author requires SOME incentive to spend the time to perfect and polish his program — with no incentive he will stop writing (or at least stop offering his programs to other users).

About half the cost of my programs goes directly to the author — so a library would need to pay at least half my sale price on each occasion a tape is lent, to keep the programmer happy — and a borrower would be unlikely to pay that much! Don't forget a library still has to cover its operating costs — and these are not low!

I am not in favour of a software library for cassette programs — I consider it would have a badly damaging effect on the supply of software.

NEW PRODUCTS:

DUST COVERS: Sherborn Designs, Victory House, 8A The Rank, North Bradley, TROWBRIDGE, Wilts. Tel: 02214 4425.
£3.95. Also made to measure service.
(From ad in Popular Computing Weekly — covers not actually seen)

TI MODULE HOLDERS: Holder for 18 cassettes OR TI Modules. In new INMAC catalogue. Stands on desk/table and measures 9¾" x 8¼" high x 10" deep. Smoked plastic.
Unlike most cassette holders (which don't work with TI MODULES very well) this one is specifically advertised as for them. £33.00 + vat.

CASSETTE FILING: Six cassettes held neatly in a plastic page for a standard 3-ring binder. Very neat on the shelf. Also INMAC —
One page £3.50 + vat.   Three pages £9.00 + vat.
Ten pages £25.00 + vat.
A binder 1¾" thick is £4.00 + vat. Pages to hold 20 5¼" disks are £7.50 per pack + vat.

INMAC catalogue free from INMAC(UK) Ltd., 18 Goddard Road, Astmoor Industrial Estate, RUNCORN, Ches, WA7 1QF.

BUGS   **BUGS**

I was a little surprised to receive three programs in one week — from three different programmers — all with the same bug. And to discover a 4th program also with the bug —

UNRESOLVED GOSUBS. When you GOSUB to a subroutine, you MUST leave the subroutine with a RETURN. If you exit with any line transfer (IF...THEN or GOTO) then the return address remains stored in memory. A few rounds of that and you will receive a MEMORY FULL error.

This is a particularly hard bug to spot (especially when you are trying to debug someone elses program!) but it is an easy bug to avoid when you are writing the program.

If you GOSUB ensure there is ALWAYS a RETURN!!!!!

You can transfer to other lines in the subroutine, but you must ensure that you don't build up a vast library of stored return addresses! A careful approach to programming should ensure this bug does not remain the most common — it really is hard to get rid of once you let it in!

NEW PROGRAMS FOR MARCH 1983:   **ADVERT**

67

≡≡≡

Rutland Court, Ponders End,
Enfield, Middlesex EN3 4BJ.

Dear Sir,

Having just received the March '83 edition of 'Tidings' and had a quick look through
it, I see that there are several cassette recorders recommended for the TI99/4A.

Myself I have tried 5 different models, Toshiba, Phillips, York, Waltham and last,
but not least the Bush 3150, price £26.99, variable on where purchased. It has
sockets for 3 Jacks (as needed) A.L.C. and only tone control and no counter, but
it is the best I've come across so far and also the tapes, once recorded, will play
back on other recorders.

Yours faithfully,   M.J. Wright.

✱✱✱✱✱✱✱✱✱✱✱

16 Aycliffe Place, Springwell Estate,
Gateshead, Tyne & Wear NE9 7BY.

Dear Sir,

I have recently sent a letter to your club asking how to become a member, since I
have sent that letter to you I have found that I have lost all interest in my TI com-
puter. Through having little time to spare and being unable to find any suppliers
of software for my computer in my area. The computer is still in original packing
since I have only used it two or three times. In the letter I received from your club
I heard you were expanding, so I thought I may offer you the chance of buying
my computer for the reasonable price offer of £90.00 or nearest offer.

Please if you do not wish to purchase my computer could you please give me the
address of any firm or person that you know would be interested in such an offer.
I paid £180. for my computer and find it a shame to see it packed away when
somebody could use it instead of it being packed away forever.

Yours faithfully,   A. Bell.

70

BOMBER – for unexpanded TI/99/4A in TI BASIC

You are the pilot of a bomber. First select a level of difficulty:

1. Incredibly easy
2. Moderate
3. Perilous
4. Impossible

Then press the space bar to bomb. You get lower until you either land or crash into a skyscraper. You get three planes.

```
100  CALL  CLEAR
110  INPUT "O.K. BOMBER, WHAT DIFFICULTY DO YOU REQUIRE
            (1–4)? ":G
120  ON G GOSUB 1110,1140,1170,1200
130  CALL CHAR(40,"0000101038FEFEFE")
140  C=1
150  S=0
160  CALL CHAR(39,"000033FFFF33")
170  CALL COLOR(2,5,1)
180  CALL CHAR(41,"FEFEC6C6C6C6FEFE")
190  RANDOMIZE
200  CALL CLEAR
210  FOR B=3 TO 30
220  A=INT(RND*H+J)
230  CALL VCHAR(A,B,40)
240  CALL VCHAR(A+1,B,41,23–A)
250  NEXT B
260  CALL CHAR(42,"000203FFFF0302")
270  X=1
280  Y=2
290  CALL HCHAR(X,Y,–1,32)
300  CALL HCHAR(X,Y,42)
310  CALL KEY(O,K,ST)
320  IF K=32 THEN 680
330  Y=Y+1
340  IF Y=32 THEN 390
350  CALL GCHAR(X,Y+1,Z)
360  IF Z=40 THEN 440
370  IF Z=41 THEN 440
380  GOTO 290
390  CALL HCHAR(X,Y–1,32)
400  Y=2
410  X=X+1
420  IF X=23 THEN 600
430  GOTO 300
440  CALL SOUND(500,–7,2)
450  CALL CLEAR
460  FOR M=1 TO 5
```

```
470   PRINT "KERR-UNCH:"
480   NEXT M
490   PRINT "SCORE SO FAR";S
500   FOR N=1 TO 300
510   NEXT N
520   C=C+1
530   IF C<4 THEN 200
540   CALL CLEAR
550   PRINT "YOU SCORED";S
560   PRINT "LESS THAN";360/G;":MORE PRACTICE NEEDED"
570   PRINT 360/G;"-";900/G;":NOT BAD AT ALL!"
580   PRINT "OVER";900/G;"FAB!"
590   END
600   CALL SOUND(400,400,2)
610   CALL CLEAR
620   PRINT "SUPERHERO! YOU LANDED"
630   S=S+100
640   PRINT "YOUR SCORE? WITH BONUS IT IS";S
650   FOR F=1 TO 300
660   NEXT F
670   GOTO 530
680   W=X+1
690   CALL VCHAR(W,Y,39)
700   CALL GCHAR(W+1,Y,V)
710   IF V=40 THEN 770
720   IF V=41 THEN 770
730   IF W>22 THEN 880
740   CALL VCHAR(W,Y,32)
750   W=W+1
760   GOTO 690
770   IF W>18 THEN 840
780   FOR E=W TO W+4
790   CALL SOUND(50,-3,2)
800   CALL VCHAR(E,Y,32)
810   NEXT E
820   S=S+4
830   GOTO 890
840   CALL SOUND(150,-3,2)
850   Q=24-W
860   CALL VCHAR(W,Y,32,Q)
870   S=S+Q
880   CALL VCHAR(W,Y,32)
890   CALL HCHAR(X,Y,32)
900   IF Y>24 THEN 980
910   FOR P=Y TO Y+8
920   CALL GCHAR(X,P,Z)
930   IF Z=40 THEN 440
```

```
940  IF Z=41 THEN 440
950  NEXT P
960  Y=Y+8
970  GOTO 300
980  FOR U=Y TO 30
990  CALL GCHAR(X,U,Z)
1000 IF Z=40 THEN 440
1010 IF Z=41 THEN 440
1020 NEXT U
1030 X=X+1
1040 Y=Y-22
1050 FOR L=2 TO Y
1060 CALL GCHAR (X,L,Z)
1070 IF Z=40 THEN 440
1080 IF Z=41 THEN 440
1090 NEXT L
1100 GOTO 300
1110 H=7
1120 J=16
1130 RETURN
1140 H=8
1150 J=14
1160 RETURN
1170 H=9
1180 J=12
1190 RETURN
1200 H=10
1210 J=10
1220 RETURN                              William Hanson.
```

ARCADE GAME SCENE

It's no secret that most of the games currently offered by Texas Instruments on Modules are direct descendants from amusement machines. Munchman and Invaders are two prime examples. In this, and future articles I hope to cover the past, present and future developments on the games available in amusement arcades.

In the beginning, circa 1970 a small company saw a computer generated game and decided that this could be successfully marketed as a stand alone unit, with integral display, paddles and of course a coin slot replacing the start button. This was a phenomenal success and the company, no longer small went from strength to strength to riches. They were Atari, and even now produce the most novel, new and most entertaining games. Pong died the death it deserved, but it had served it's function. People wanted to play video games.

Atari had paved the way for a new industry, investors and inventors flocked to produce newer and more exciting games. They might have tried, but most were dismal failures. Very few pre 1977 games are still played, even as £2.00 tapes for the humble micro.

The industry nearly died, with amusement machine operators, sick of high outlays and poor takings, refusing to invest further in video games. Then came the first of the blockbusters. BREAKOUT. Yes once again Atari had set the amusement machine industry alight. People couldn't push their 10p's quarters and 100 yes into the machines fast enough. Atari couldn't make the machines fast enough and at this point occurred a problem that has been the plague of every form of software, from video games to records. Piracy.

Unlicensed and illegal copies of Breakout were appearing worldwide, and with the software laws being in an even greater mess than they are now, there was little Atari could do to protect their investment.

But all good things come to an end and the fickle public grew tired of knocking out bricks with a bat and ball.

In mid 1978 came the greatest explosion in video games that will ever occur. From Atari came the game Asteroids which filled cashboxes fuller than any game before or since. At one point the Atari factory was making over 4000 of these machines per week. But even this amazing success was eclipsed by the arrival from the Far East. Taito, a company producing unremarkable games in Japan, introduced the game now synonymous with the video game. Space Invaders had arrived. It caught the imagination of everyone, from the kids playing truant to the Chancellor of the Exchequer (who never did carry out the threat of taxing the video game out of existence).

The Japanese had a foothold at last which was later to turn into a stranglehold. They had already destroyed the Motorcycle industry, reduced the car industry, consumer electronics and plastics to former shadows of themselves and by God they were determined that they were to ruin the video games industry too. Ultimately they did indeed cause the second cardiac failure of the industry. For eighteen months it seemed that anyone in the Far East with access to a soldering iron was churning out variations of Space Invaders.

No game, no matter how good, lasts forever, and at this point the Japanese with their new dominance as market leaders, didn't know what to do next. Sooner than relinquish the grip they had, they proceeded to flood the world with what I termed Space invaders in colour with knobs on.

Colour was the main breakthrough. Until 1979 nearly all games were in black and white. Now the new medium was more important than the games. Games of unbelievable mediocrity were grabbing the public eye merely because at some point in the game the player was rewarded with a pretty display. Games like Galaxian, Astro-fighter, Sherriff, Red Tank, Tomohawk 777, Deep Scan, Lunar Rescue, Space-Chaser and others even more immemorable followed in a flood. During this aggresive sales drive, good machines were never allowed to receive the acclaim they deserved, and the likes of Sega's Monaco Grand Prix were dismissed along with the unworthy.

Not that the Japanese were entirely to blame. With the notable exception of Atari, The Americans weren't doing themselves any favours. Exidy's Crash and Targ and Gremlins Invinco seemed to be the pointer to the lack of innovation to come. Once again the industry was in the doldrums. And once again it was Atari to the rescue.

Missile command put an end to the left right fire movement. With three fire buttons and the novel Trak ball, Missile Command was the first genuinely new game in almost two years. This time the revival was coming too late. Operators were heartily sick of the poor quality hardware and software being thrown at them from Japan and Missile Command, fine game that it still is, never received the acclaim or sales that it deserved.

Hot on its heels did come something new. Battle Zone was the first game to feature true 3—D Graphics, although only in wire frame form. Sadly the game, rather than the concept wasn't good enough and even the mighty Atari seemed stumped for new ideas.

It is debatable as to whether the next revival heralded from Japan or the States. Pac-man was certainly the last game to become a household name. Just as in previous years when there was a lull, along would come a saviour good enough to rekindle the enthusiasm of players and operators alike. Pac-man has since sold more copies than even Space Invaders or Asteroids, and for the first time a souvenir industry was spawned to accommodate the need to spend more than just time playing machines. You could buy Pac-man toys, write with Pac-man pens on Pac-man stationery, sleep soundly on a Pac-man pillow case and every other item that the little gobbler could show his face on was duly conceived, made, marketed and sold by their thousand. The most remarkable event was C.B.S. making a television series based on the hungry yellow mouth. But Pac-man was the end. Never again would the public's imagination be caught by images on a cathode ray tube.

If the reader feels I have shown a heavy bias toward Atari, this is only because I believe in their products. Both in terms of game quality and the standard of hardware and documentation, they are streets ahead of everyone else. If you received any documentation with any Japanese game it was a miracle. Mind you it rarely

did any good. I only once received a manual in English. My Japanese is non existant. I should like to point out that I have never worked for or received any money from Atari and that my sole relationship with them is the same as yours. A customer. Not that Atari are perfect. I have one major criticism of them, and this has remained constant for as long as I have done business with them. Their prices are such that Texas Instruments are very firmly in the bargain basement area. The same applies equally to the home computer. Atari have very much followed Texas Instruments when it comes to pricing. Witness the halving of prices of the Atari 400 in only six months. Software prices are the other way round however. Games for the Atari home computers were cheap long before T.I. realised that people weren't interested in forking out £40.00 + for a game of space invaders. If T.I. were a little more forthcoming about the internal goings on of the TI/99/4 I'm sure software prices could fall still further as third party software houses could make their own modules. I for one flatly refuse to fork out the exhorbitant sums required for the means to run machine code. I know that Thorn EMI will shortly introduce Submarine commander and River Rescue for the TI99/4 but I'm sure that with access to certain information I could make the modules ready programmed.

How is all this talk of arcade games relevant to the home micro scene?

Well, remember that nearly all games you play on the TI99/4 started life as a printed circuit board whose sole function in life was to put a picture on a screen with the intention of persuading a player to part with his money. There are exceptions to this. Tombstone City, Zero-zap and Chisholm Trail are just three examples, and I hope there will always be people prepared to devote time and effort to making and marketing good **original** software. But copying and improving on arcade games is not a bad start. After all, anyone playing a game in a pub or arcade has paid to play it and as such deserves an entertainment value from that game.

Once something becomes successful, the analysts move in and try to find reasons why. Usually in an attempt to destroy the concept. Social workers blamed Space Invaders for a dramatic rise in truancy. They also criticised the video game for the increase in destitution amongst the unemployed, a fact which looked at retrospectively was the nonsense responsible for successive attempts to ruin the industry by taxing it to the point where it was no longer economic to operate video games. Psychiatrists were divided. One group found the games to be mind damaging whereas another decided that helping unsure kids to regain their confidence by letting them kill a few aliens or rescue a damsel in distress did nothing but good. The old timers felt they ruined their local with their constant pings, zaps and kerrangs. They also felt the same way when the Juke box all bar totally removed the traditional pianist from the pub. Yet I get asked many times by the older set for their favourite records. Someone, somewhere has finally accepted the Juke box, they will do the same for the video game. (Enter a younger generation of traditionalists?)

In the next article I hope to come up to date with developments taking place in Japan, the new generation of programmable video systems and the worsening piracy situation.

If you feel the urge to copy an arcade game, beware! The laws on software piracy are gradually becoming clearer. Many new games are licenced from the original software house and without the appropriate licence you could lay yourself wide open to prosecution for trying to sell someone else's software without having first paid the appropriate royalty. Your arguments that you did not copy but translate will have little effect on an unsympathetic judge.

Until the next edition
HAPPY ZAPPING.   Howard Greenberg.

NOTES FROM THE NORTH

Get Rich Quick!

Some of you will have seen an advert for a compiler for the Sinclair Spectrum giving the following "Benchmark" improvements:

	Spectrum Basic (Secs)	Spectrum Compiler (Secs)	TI99/4A Basic (Secs)
BM1	4.8	0.15	3.0
BM2	8.7	0.09	9.0
BM3	21.1	1.10	24.0
BM4	20.4	0.99	24.8
BM5	24.0	1.11	26.2
BM6	55.3	1.91	61.9
BM7	80.7	2.14	84.6

and all for £14.95.

There must be a fortune waiting for the first person to write something comparable for the TI with a potential market of, say, a million worldwide. TI are hardly likely to market one as the attraction of their other languages would begin to dininish. After all, my PACMAN copy would even be marketable at, say, 10 times the speed! I have a nasty feeling that such a program for our machine would require mini-memory or even Mem.Exp(ensive).

TI LOGO

Like some other people, I expect, I was very impressed with LOGO on the BBC Horizon programme earlier in the year and actually went ahead and bought it. This was a very big step since the "Box" and Memory Expansion are required. The language is certainly interesting and for my young children very good as they can get things happening on the screen easily and quickly. They seem more attracted by the 32 sprites than turtle graphics. There are a number of points, many of which, I think, deserve an answer from TI.

1.   The manual is abysmal. I know the language is one of discovery but eg should we really have to try and discover how to do list processing when this is one of the central features?

2. It is clear from the Turtle Graphics that the graphics mode on the 4A which allows individual pixel colours is not being used. i.e. Each **character** only allows background and foreground colours. The turtle also runs "out of ink" fairly quickly, my guess being when it has touched 256 characters.

3. Attracted by the sprites (I don't have EB yet) I have tried to write a simple video game. On the face of it this should not be hard but I have found that any program which depends on the user responding with keys seems to run out of memory quite quickly. In fact the User memory seems very small considering there could be a total of around 100K ROM and RAM in use. There are no examples of users interacting with a program in the manual.

A few tips:—

— F is allowed as an abbreviation for FIRST although this is not in the accepted abbreviations list.

— As well as the normal square brackets keys, shift 4 and 5 (on the 4A) also give them. Square brackets are, of course, used a lot in LOGO and I've found this a time saver.

— Well worth taking the time to list (PA) the programs on the sample cassette as you will learn far more about actually using LOGO from these than from the manual.

Finally, a major irritation. Stephen Shaw has been mentioning LOGO 2 for two or three TIDINGS now and I saw the product advertised in the December 82 issue of an American magazine. TI UK will **not** acknowledge its existence even to major stockists like ABC and in the meantime people like me are ripped off for what I am sure will turn out to be an inferior product.

Program Reviews

Apex Trading, GM4-Space Attack and Snake £2.95
This looks like a clear case of you get what you pay for! Both these programs are fairly primitive TI BASIC games and do not hold the attention for long. Space Attack involves moving a white spare around the screen on to Aliens appearing at random and then "firing" to erase them. Does at least offer different levels of difficulty but is fairly ponderous. Snake is a familiar type of program with a continually growing snake moving around the screen eating spiders and trying to avoid both rocks and itself. The action is fast for a TI BASIC game but not especially exciting.

Apex are at least offering cheap software for our machine and should be encouraged. They will consider publishing your programs but I expect royalties will be low. You should also beware of using your best kept secrets as the programs will LIST.

Parsec
This program is superb! The game follows the well tried format of a craft flying over a varying landscape being attacked by assorted objects. The graphics, sound and speech are excellent and the action very fast. Further complications are asteroid

belts and having to negotiate a refuelling tunnel if you've had one craft for too long. Best to vary the "lift" for these. The manual is very full for a game and indicates that there are 16 levels to get through. So far I am up to number 3! The program shows how good our machine can be. It must be worth TI writing many more games, given the likely sales of good ones like PARSEC, or subcontracting the work to big specialist houses. As someone with young children I welcome the huge new range of educational software announced by Robin Frowd in the last TIdings but I am worried that the flow of games seems to have virtually dried up.

Yahtzee — Library No. 050

An excellent version of the game. TI BASIC can be adapted to games like this very well since no fast action graphics are needed. Music and comments complement the game well and there can be no point buying the TI module. The game makes a pleasant change from zapping aliens.

Note: Air Attack (Library 072) needs Extended Basic. (alter your lists).

Suppliers etc.

Well done to Akhter Instruments of Harlow, Essex who are offering the Disc Drive at £200 and the Controller at £145, carriage free. At last the peripheral prices are beginning to be reasonable. Akhter also advertise that they will not be beaten on price. See their adverts in PCW.

Well done also to Dixons of Middlesbrough who had a 99/4A available for use (no other computer) complete with speech, PARSEC and the new joysticks. This configuration should be made compulsory for all TI dealers — very impressive.

New joysticks you might be saying. Are these the "new" ones that I bought only last Christmas? Well, no — TI have been quick to see the inadequacy of those and have now brought out a far better version with a smaller base, shorter stick mounted on a ball (very easy and smooth) and a side mounted fire button. If you are about to buy some wait for these.

TI BASIC

TI BASIC is a strange animal and these columns are full of comments about its shortcomings. Let us, for once, look at some of the good things. Very few micros have NUM, RES and TRACE facilities on the basic machine. Recent magazine listings, none of them very short, have included techniques to:—

- — Mix text and graphics on the Dragon
- — Stop a LIST in the middle on a VIC
- — Design your own characters on a Spectrum

You wouldn't believe how tedious the POKEing of Graphics and Sound commands can be on other machines and did you know that VIC programs stop while Sound commands are being enacted?

It pays to count your blessings now and again.

Last Edition

Thanks to Don Weerasinghe of Shepherds Bush for pointing out the error in my

"mirror images" program in the last issue.

130   READ (X,Y,Z) should, of course, have been

130   READ X,Y,Z

                                        sorry!

My only excuse is that, like many people, I suppose, I use other languages at work and occasionally forget to totally flip to TI when I get home!

In this program I asked whether you could see how to relate X, Y or Z to N to cut down the DATA. The answer was to change X to N+4 and remove the numbers 5 to 13 from the DATA list (each third number).

Video Chess

I have an interest in computer chess and subscribe to a specialist magazine but already owning a dedicated chess computer have not bothered to buy the TI module. Is it any good? Most of the reviews I see are fairly disparaging which is disappointing since you can buy a whole chess computer for less than the price of the module. The size and power of the 99/4A should produce a reasonable game.

Perhaps readers could send in sample games or examples of brilliancies or blunders. Even more interesting has anyone played a 'match' against other programs. eg VIC Sargon.

Eventually I hope to have a go at writing a program but this will only be worth it when I can afford Assembly language.

ONE YEAR WITH THE TI 99/4A

Having read all that advice about 'decide what you want to use a computer for' (and how do you know that before you make a start) I chose the TI in preference to the VIC 20, Acorn Atom, Sinclair, Video Genie etc. because:

a.   It was available
b.   The Texas label gave it a good pedigree
c.   Good RAM, colour, graphics, sound, display and keyboard
d.   Very good modules — all be it pricey
e.   Very good manuals — so important to the novice
f.   Potential for both games and programming

One year later have thos perceptions been justified? On the whole, yes, and I would add that the ease with which programs are saved and loaded from a Boots CR325 recorder (price £19.95) must be an advantage over so many systems which have trouble.

What then are the disappointments?

a.   Sour grapes that what cost me £300 is now sold at £150
b.   The prohibitive cost of adding any real enhancements

Joysticks have been my sole purchase.

c.   The limited line length of 28 Chs constricts the design of any program with text, eg columnar reports

d.   The lack of sales has discouraged interest by the computing magazines in chat, hints and programs. (Full marks then to Paul Dicks for TIdings). Ditto software other than of TI origin

e.   Lack of machine code programming and PLOT/DRAW commands.

f.   A design to deter addition of any non TI peripherals.

I was encouraged by TI's determination to generate sales before Christmas by the price slash (under orders from the US?) but do they seriously expect to sell any printers at an effective cost of £580 even from our good friends AB&C Computers, or a disc system at £600? It simplifies the sales projection at Manton Lane of course. TI's price structure is now completely unbalanced and we must all swear an oath to buy nothing until there is at least a 50% reduction on all the plug-in goodies. 'Curb excess profits of Multinations', 'Time for Texas to talk turkey', etc. I hope the first clever TI owner to use adapt other peripherals will tell us all about it. I understand that the new TI matrix printer is a 're-badged' Epson MX80 FT-3 and the problem hinges on providing the RS232 interface. When suitable alternatives are found I hope TIHOME will take a full page ad. in PCW to tell the world. You can have £10 from me.

Finally, what would I like to learn more about through the pages of TIdings?

a.   A primer on machine code (like the excellent beginner's Basic)

b.   More on pretty graphics with listings.

c.   Software reviews — especially non-TI

d.   A page of simple tips for beginners.

e.   Simple DIY repairs, eg. If the screen exhibits a curious partial display when switching on suspect that the consol switch has insufficient travel. Remove the base, disassemble the switch and carefully shave the offending plastic part with a very sharp knife. End of problem. (No call-out fee, just send me the £10 service charge!).

David Burch.

****

COPY HOLDER

I am sure that the INMAC 'Suspension Copyholder' reported in December is very good if you want to spend £29 but the design illustrated on the next page works very well at a fraction of the cost. It is stable with quite heavy manuals and the ¼ round beading on the ledge keeps pages open. There is sufficient clearance for a cassette recorder underneath.

After I had cut the wood to size my 10 year old son Robert assembled it using ½" and ¾" moulding pins and wood glue and passed a Cub's test in the process. Care must be taken that the bottom edges stand squarely on a surface without rocking.

Be warned though — it also gets used as a recipe holder, music stand and for model assembly instructions.

David Burch.

1"

14"

4mm ply

3"

4mm ply

6"

5/8"square softwood (2 off)

4mm ply

5/16" softwood

1/4 round beading

11"

12"

1¼"
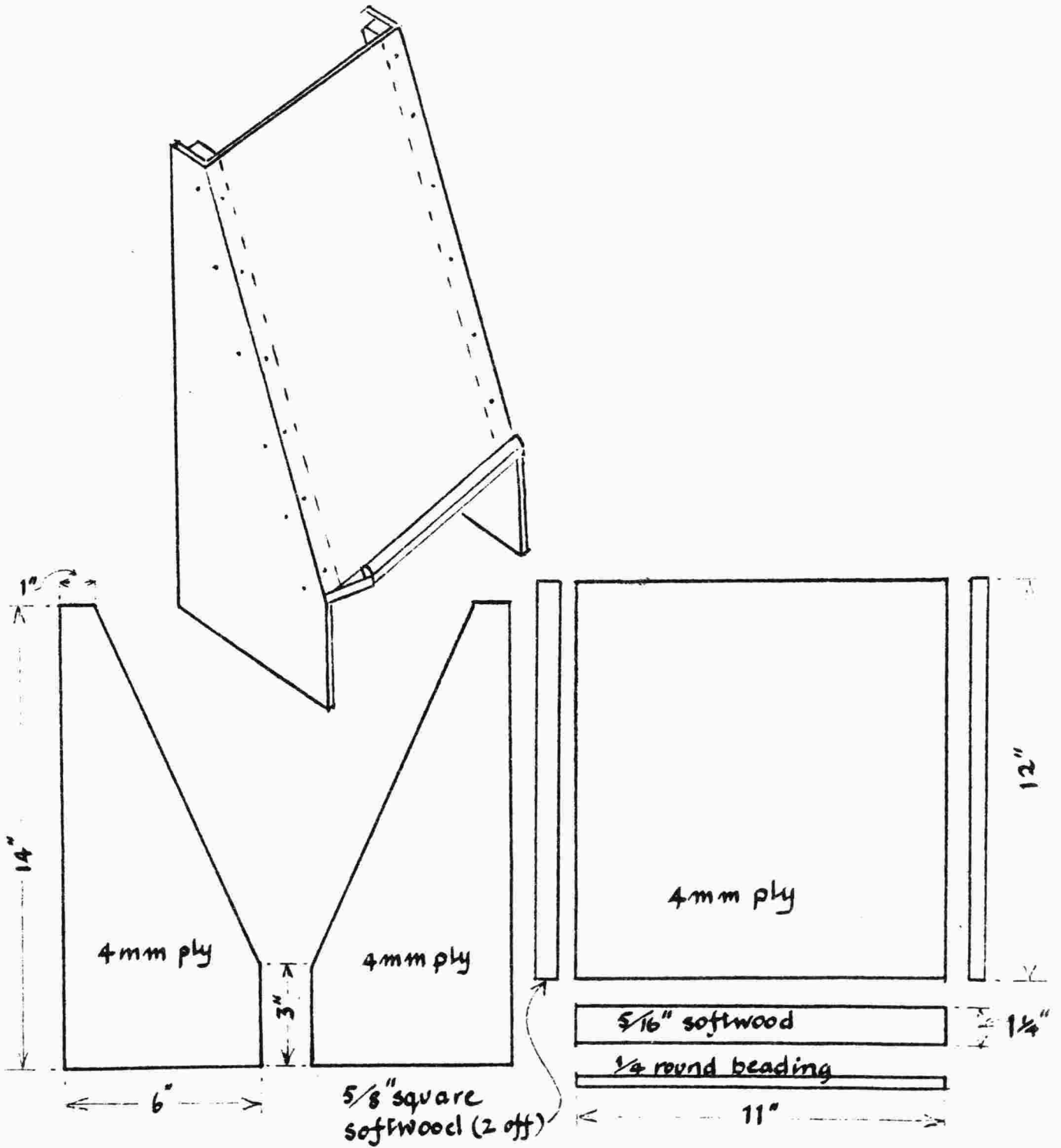
L E T T E R S

=======================================

2FL, 16 Viewforth,
Bruntsfield, Edinburgh.

27th March 1983.

Dear Mr. Dicks,

Firstly, thanks for the excellent new format magazine. It is extremely interesting to find out how other people are coping with the TI 99/4A. I was prompted by Jeff Freeman's experiences to tell you about my attempts to have my home computer repaired by TI in Bedford.

The sad story started in early January after having received the adventure module as a Christmas present. Having completed Pirate rather rapidly, I was attempting to discover the secret of savage Island (1) when the screen of my B/W television went blank and a warbling noise was heard. The televison was still working and so sadly I concluded that the computer had cooked a chip. I returned this to TI after a phone call to a very pleasant young lady in the service department. Two months later the computer was returned with an invoice saying that it had been serviced. It still produced the same blank screen and noise. Oh dear, I thought, it must be the PAL modulator which I had omitted to send. To be sure this time I took the unit to my local supplier and proved that the console was working well. I therefore sent an apologetic letter to TI for having wasted their time (and mine) and sent them the PAL modulator for repair. You guessed it! A month later, it was returned with the usual service note and lo and behold, the same noise and picture were present. I had foolishly assumed that since the console light was lit that the power supply was working. Removal of four screws revealed a basic transformer with two power supplies and fuses on an otherwise bare circuit board (look for yourself!) with one of the fuses blown. **£@!!!!!! I said, all that time without my computer for a simple fuse. I would again like to apologise to TI for wasting their time and hope that someone else may benefit from my experience.

During the period that the computer was away, I received Parsec and Alpiner (I enclose brief reviews if you are interested) which I think reveal the differences between the 99/4 and 4A.

Finally I would like you if possible to include the following advert in the next issue of TIDINGS.

FOR SALE	Tombstone City Module		£12
	Car Wars Module		£15
	Video Games 1 Module		£12
WANTED	Statistics Module	up to	£35
	(or swap for the above)		

REVIEWS

PARSEC – This is a game of the Defender type combined with Scramble and features excellent graphics. There is a scrolling screen and flashing star background. Various types of ship appear at the right of the screen and move at great speed toward your ship. You are equipped with a rapid fire laser and can manoeuvre in all directions. Various ships either try to ram you or shoot at you. Your laser overheats if used too often. If you manage to pass an asteroid field then you move to the next set of ships. At various stages it is necessary to refuel through a very narrow refueling tunnel. There are 16 levels each with new hazards. I have managed to reach level 3 only. As an added bonus there is vocal encouragement for good play and for warnings. Overall the graphics and smooth fast action make this game as good if not better than the equivalent arcade games. Extremely addictive!

ALPINER – This game is unlike any arcade game and is basically a variety of maze game. The alpiner is a figure of a mountain climber which "climbs" as the screen scrolls vertically. The idea is to reach the tops of six different peaks, higher each time. There are a number of static hazards e.g. tree trunks and pine trees and a number of mobile ones e.g. rock falls and skunks amongst others, each causing the climber to forfeit a life or several metres of climb. At various stages, targets appear for bonus points. The first few levels are easy, but hazards increase at later levels. The game is not as interesting to play as Parsec but graphics and speech effects are amusing. If Parsec gets 10 out of 10 then Alpiner gets 7. Both are a distinct improvement graphically on previous module games and unfortunately only work on the 99/4A machine.

Yours sincerely,   Neil Frazer.

****

(from:
R.P. Hammill B.Ed.Hons.
Headteacher)

SAINT EDWARD'S SCHOOL,
Wivern Place, Runcorn. Tel: 72317.
29th March '83.

Dear TIHome,

Another holiday is upon us so there is time for a letter from blackboard land. I thought some of you might be interested to hear some of the things that are happening in the world of Educational Computing as they filter through the mind of someone in school.

You have probably seen those adverts by now..."Eight out of ten Primary Schools are choosing the BBC micro". Well here is the background. As a result of an earlier government initiative every Secondary School should now have a micro. Most have several. Presently a scheme is operating whereby the Department of Industry provides half the cost of a micro package for each Primary School prepared to take advantage of the offer. The objective is that every Primary School should have a micro within the life of the scheme. Then, bitten by the bug, they might find their own funds to buy future micros.

Three British machines are offered through the scheme. The B.B.C. Model B, the Sinclair Spectrum 48K. and the Research Machines 380Z. Each must be bought as part of a package including a monitor, tape recorder, leads and some introductory software. Clive Sinclair offers some extra goodies like a free printer and a Tidy Tray expansion-box.

Most Primary Heads I know are very much at the learning stage and will admit that they don't know enough about micros to make an informed choice. Most of us are prepared to be guided by Regional Advisers. In some cases Local Education Authorities make a policy decision that they will administer the scheme only where a particular choice is made.

The Spectrum is generally ruled out by virtue of its chatty keyboard. It is so portable that it is a security risk and let's face it, there is also a snobbish tendency to dismiss anything with the Sinclair name as belonging to the toys and hobbyist end of the market. It isn't fair this last fact but there it is. Teachers who have experience of Educational Computing through Secondary experience or college courses have generally used the Apples, Pets, 380Z s and suchlike so they are coloured in their view.

The 380Z is itself ruled out of most Primary schools by the price. There is only the Beeb left if we want to take advantage of the half price scheme. Even the Beeb works out at £400.00 since the D.O.I. money pays for the monitor and bits and pieces. Many of us are worried not so much about how to pay for the first micro but how to pay for the next ones since the pound for pound scheme applies only to the first. The Beeb is going to cost £400.00 a throw and then some for extra monitors, tapes etc. The much publicised Econet system is not cost effective until you link up more than ten micros. Well I ask you ... How can a school of a hundred or so children with only £1200.00 a year to spend on books, materials, stationery .... everything manage to exploit the micro revolution at those prices?

Capital expenditure on Education continues to rise but our buying power continues to fall. Schools like my own have opted for a low cost entry into the use of micros. Yet the Beeb bandwagon is underway and there is no point in fighting it. Eventually I will buy at least one Beeb for this school but the time is not ripe. Why does a TI 99 4A fan intend to buy a Beeb? Well firstly there is the tube. This is a gizmo inside the Beeb which, we are told, will permit almost limitless low cost expansion. Secondly there is the support which is becoming available as more and more schools develop software of their own and make it available for colleagues. Thirdly there is the prospect of free Telesoftware. (Mind most of what you see on Ceefax at present is far from impressive!) Fourthly there is a marvellous extended BASIC WHich is less offensive to programming purists than most BASICs. (Mind I only know two so called 'purists' and they are deathly bores who write brain numbing programs anyway). Now there is another thing in favour of the Beeb stable. Acorn are soon to release a new micro called the Electron (Elektron?) which may be out by the time you see this letter. I hear that it will have 32K RAM and will run all BBC programs. It will be cheaper because it will lack a lot of the interfaces which the Beeb carries as standard (Rows of the things all over the bottom and sides!) and it will retail for about £130.00 it is rumoured. A school might have one Beeb as a developmental machine for program writing and three or four Electrons for

classroom use. I doubt whether it will live up to the rumours myself.

For the moment we are happy to carry on with our trusty TI 99 4As.

You might be interested to know how the TI ("Tiny" to the children) is used in our school. Firstly none of us are interested in teaching programming as an end in itself. Even now many University Lecturers wish their students had not acquired the restrictive concepts of BASIC programming before undertaking University level computing courses. I do teach some simple programming but only as a means of promoting clear sequential thinking. Equally we can and do use recipe writing, craft instructions writing, map-direction giving and other tasks to this same end. Programming, at the user end of a computer, will be obsolete in the near future. The need to think quickly, sequentially and purposively is eternal.

A major aim for a primary teacher is to widen childrens' experience of life. Here we try to give them a variety of experiences of the micro ... helping them to see that this is the most open ended multi-purpose tool in history. We explore its possibilities for routine jobs, simplifying and speeding task execution. We explore its possibilities for entertainment. We try to make use of the micro for C.A.L. (That's an "in term" at the moment standing for Computer Assisted learning). All sorts of skills can be practised and concepts can be developed whilst a child works or plays with a micro.

In the earliest days (not so long ago!) I was bringing my own TI into school. Fired by the amazing release of enthusiasm for nouns, verbs and the like which was generated by the "Beginning Grammar" module I set about writing programs of the drill and test type. They were simple maths programs for the most part. Surprise, surprise the children even wanted to work through these, to me boring, programs.

Then I began to see the snide comments in the Educational Press about how micros opened up a whole world of possibilities but boring teachers like me were using them as tools to maintain the status quo and nothing exciting was happening at all. O.K. I thought, I'm not going to give up using the micro for drill because drill is necessary for some things whatever the trendies say. On the other hand I would try not to use it only for drill.

My first use of the micro for creative purposes was to get the lower juniors to design space invaders and make simple one line alterations to one of my programs so that they could see their own shape on the screen. They coped easily with this and a nice wall display resulted. The topic raised all sorts of side issues; scale, coordinate grids, binary and hexadecimal notation. Soon the children were trying to repeating patterns on the screen and developing characters that tesselated or joined to form interesting shapes.

The next step was to write instructional programs. O.K. teaching should be about personal contact and the teacher is remote from the child when he passes on his instructions through a micro. Remember, though, that we delay the application of our work to the child when we give him a duplicated sheet or a text book. It seems the same to me. Anyway we did wax philosophical about this one in the

staffroom but we gave it a try and a program on "Place Value" was produced. This first instructional program teaches the child how numbers have different values depending whether they come in the units column or the tens or the hundreds. It also tests their understanding of the content. The program was intended as an electronic text book and exercise with immediate feedback, colour and sound. It was submitted to the TIHome Library and accepted. (I think.) Anyway as a result of the programs I sent to TIHome we received sixteen new programs within the week. (Thanks, Paul. I don't know how you can do it and work at the same time!) Now we had a program library!

The children were thrilled with all the new things we could do on our computer. So was I! Paul later sent us a pile of his own programs and a module (Household Money Management) to further extend our capabilities.

Meanwhile I had been working in the dark on an Adventure style program to stimulate research work. The answers to various clues could only be found by using books and instruments in the school library. I say this was done in the dark because at that time I had never seen an Adventure program. Now I have and I must admit that it was the wrong label to put on my effort. Still within our school and with me close at hand it has proved a very useful program. (Also in our users' library I think).

The real Adventure program I have seen comes from Apex Trading. I have bought two of their programs and can recommend them. Their programs are all for the unexpanded machine in TI BASIC. 'Goblin Caves' has three dimensional graphics. The goblins, pits of molten lava and the monster are not shown graphically. You die just as you enter their caves so you don't get a chance to see them. It is very playable and, of course, is listable so you can lift the subroutines which handle the graphics. On the reverse side is a free program of an anagram game. This firm deserves to succeed. Sadly it is a tricky market for TI software. If buyers make unauthorised copies for their friends such firms who must have very little profit margin will simply cease to write for our machine. The genuine adventure is called 'Island Adventure' and has pirates, trolls, wolves and other hazards as well as handy and hidden objects and passwords and so on. I really enjoyed it. You DON'T need the TI Adventure module for Apex Adventures. I paid £3.95 for 'Island Adventure' and £4.95 for 'Goblin Caves'. These are the two dearest they do. Others range from about £1.95 and many have several games. 'Island Adventure' carries 'Four in a Row' on the reverse side.

Enclosed with this letter is my attempt at a genuine Adventure style program. It is called 'Space Station Adventure'. It is written in Exbas since this allows protection against listing and cheating. The logical structure is cribbed unashamedly from the Apex 'Island Adventure' but the layout, locations, scenario, plot, items to use, hazards, codes and objectives are all original so I trust Apex would not consider this a breach of copyright.

Thanks to the contributor who suggested going to Boots to look for end of line bargains I managed to get a statistics module for only £10.00... certainly not to be sneezed at. At this sort of price it makes a good buy even if you aren't interested in statistics since as Tidings readers will know this module interacts with the Basic

ROM to produce Enhanced Basic. In its normal mode this module is a delight to use. It won't teach you about statistics but if you are already au fait with statistical methods and require the use of descriptive, analytical and inferential tools then this module is the answer. Now that we are obliged to reveal to our Governors the results of standardised tests in schools this module is a godsend. All I need now is a second-hand thermal printer to go with it so that my time isn't wasted transcribing from the screen.

I have had an order placed for Extended Basic since October or November. Last week I found one in Curry's in Warrington. There's a Texas on display there but the salesman who is only in on Saturdays tries to demonstrate its capabilities by inputting a programme to work out averages! I watched two prospective customers get that glazed eyed look .... Anyway I bought the module from him even though I couldn't wangle a discount. At £80.00 it cost more than it would from the shop where my order was placed but a bird in the hand.... Funnily I checked with the original firm, Computer City in Widnes, and they told me that they had had it in stock but were out of it again. That is why they got a cancelled order. They had promised to get in touch if one came in and I'd checked every week since before Christmas! It's a pity because the proprietor is working his legs off building good customer relations but when firms grow suddenly things can slip. Anyway if any members visit the shop which is part of the Spectrum chain I suggest they ask for Mr. or Mrs. Hargreaves by name until they get their staff teething troubles sorted out.

What do I think of Exbas? Well I make no secret at that price I was tempted not to bother and to get a new computer instead. But I've got this unhealthy obsession with the TI 99 4A and I can't let go. Now with Exbas version 110 I'm thrilled to have stayed on. Don't let anyone tell you Exbas is slow! It might be slow on benchtests but for screen handling it's like lightning. The sprites are amazing. I will be happily occupied through many insomniac nights getting to grips with the 40 new and extended calls and commands. I have been keeping a magazine program for some time hoping that one day I'd have the Exbas module to implement it. I typed it in and couldn't believe my eyes. The sprites look like machine coded graphics (which I suppose they are really since CALL statements seem to call up machine code subprograms). The program is called "Target". A Spectrum owning friend and a Beeb owning friend wouldn't believe the evidence of their eyes at first. They couldn't see how such speed and smoothness could be achieved through BASIC.

Some of you might have seen the "Horizon" broadcast about LOGO the computer language which is designed to provide a learning environment for children and older people. (I particularly like it because it shows little keyboard maestros using OUR micro).

Well after seeing the "Horizon" programme I can see that my approach to micros is pretty formal and my 'enlightened' programs would be pretty unimpressive to some of the LOGO pioneers. But really the LOGO language just raises all the old arguments which have been the staple diet of Primary Educationalists over the years. Should we give more time for experience which the child directs for himself

or should we determine his direction for him and provide tasks which lead him to learn what we want him to learn? Well this Horizon tape really brings these issues to the fore.

The only way I'll ever get anything off to you is if I knock out a paragraph whilst I'm passing the typewriter so these thoughts do not come out in any particular order. They are just the musings of a TI user who happens to have a particular involvement with kids and schools.

If these thoughts are of any interest to fellow TI homers then I'll try to make a regular contribution.

Here is a short program of the Lunar Lander/Apollo type. If you have not got the Lunar Lander from our TIHome library you must get it. The graphics are excellent and the game is very challenging. This offering is very simple but I haven't managed to land yet. I've adapted it from a ZX81 program by a colleague of mine. There is plenty of scope for improving it.

```
10   CALL CLEAR
20   PRINT "MOONLANDER"
30   H=500
40   V=50
50   T=0
60   F=120
70   PRINT :::::: "TIME"; T, "HEIGHT"; H
80   PRINT :: "VELOCITY"; V, "FUEL"; F
90   IF F=0 THEN 140
100  PRINT "BURN? (0-30)"
110  INPUT B
120  IF B<0 THEN 125 ELSE 130
125  B=0
130  IF B>30 THEN 145 ELSE 150
135  B=30
140  IF B>F THEN 145 ELSE 150
145  B=F
150  V1 = V-B+5
160  F=F-B
170  IF (V1+V)/2>=H THEN 220

180  H=H-(V1+V)/2
190  T=T+1
200  V=V1
210  GOTO 70

220  V1=V+(5-8)*H/V

230  IF V1>5 THEN 235 ELSE 240
```

) *Sets starting values*
) *for time, height,*
) *speed and fuel*
) *and*
) *Prints them*

————— *If fuel is exhausted computer jumps down prog bypassing input section which asks you for a burn. It then prints running commentary of approach to surface.*

Gets number from you and check that it is within correct limits.

— *calculates new speed (velocity—*
— *calculates new fuel level*
— *checks if distance travelled in last go is greater or equal to your height above moon. If it is you've landed*
— *calculates new height*

— *new speed*
— *goes back for next update and perhaps input if line 90 permits*
— *calculates your speed on touch down and checks what kind of landing it gives you.*

```
235  PRINT "YOU CRASHED – ALL DEAD!"
236  STOP
240  IF (V1 > 1)*(V1 <=5) THEN 245 ELSE 250
245  PRINT "HARD LANDING" :: "SOME INJURIES"
246  STOP
250  PRINT "GOOD LANDING"
```

Happy Computing,

Ray Hammill.

* *

SOME LETTERS FROM THE PUPILS AT SAINT EDWARD'S SCHOOL.

Dear Tihome,

I would like to tell you about our computer. My favourite programs are 'cat and mouse' and 'Blind maze'. On 'cat and mouse' there is a maze with a cat and a mouse. There are three speeds, fast, medium, and slow. Some times you get muddled up because you can't find the right keys. The computer tells you to type a letter or it gives you a sum to do. If you are quick you can press the right answer in time. On 'Blind maze' you see a picture of a man in a maze. Then the screen goes blank and you have to remember where the walls are. You have to steer the man through the maze, sometimes you crash because the maze is invisible.

Love from Sarah Hughes, aged 9.

* * * * * *

Dear Tihome

I am writing to tell you about one of the best programs we have in our school. It is called "cat and mouse". It is good because when the mouse does not move the cat moves closer to the mouse; then you think your going to lose but some times you win. The sound of when the cat eats you is munch, munch and when the mouse wins the t.v. goes red and blue. It is a very good thing to play with. I wish I had a computer at home. At school we have got Isola, Apollo, and Star Trek, Obstacle, Blind Maze, Yahtzee, Towers of Hanoi, Kingdoms, Goblin Caves, Island Adventure, Connect four, ten pin bowls, pools, Beginning Grammar, Gavaquinho, fantasy. My favourite is Yahtzee. You gave our school that. Thank you for the very nice games you gave us; they are very good.

Love from Denise (9 yrs old)

* * * * * *

Dear Tihome,

I would love to have a computer at home. My dad won't let me have one yet. The best games are cat and mouse and coonect four. I like 'Battle at Sea' as well. The

computer is hard to beat. If you squeeze all your battle ships into the top corner he sometimes gets mixed up and he doesnt know what to do next.

Love from Annemarie.

******

Dear Tihome,

"Cat and mouse" is one of the most popular games we have. It is good for learning how to type. You go mad sometimes when you can't get the right keys quick enough. It is very exciting. Sometimes you think you are going to lose but you win and it feels good. In cat and mouse there is a maze with a cat and a mouse, the screen is blue. The walls are red, the cat is black and it look a bit like a dog. The mouse is white. The bottom of the screen is white and you see black letters appear on it. You have to be very quick to press the right letter. If you are too slow the cat catches up on the mouse a bit. If you make too many mistakes the cat eats the mouse.

Love from Jennifer Eaton (age 8½)

===

2 Cheshire Road, Albany Estate,
Norton, Stockton-on-Tees, Cleveland TS20 2RP
1 May 1983

Dear Mr. Dicks,

I have had my TI99/4A computer for a month now, and as a complete beginner to computers, I am beginning to understand the language! I have discovered a mistake in the User's Reference Guide on page 94 in connection with the JOYST Sub-program. Line 190 should read:

190 B= −Y x 1.6 plus 12.2   not   190 B= Y x 1.6 plus 12.2

Sorry, my typewriter does not have an asterisk or addition sign! Also, has anyone else seen the new Joysticks! I was in Dixons at Middlesbrough on Friday, and they had them plugged in the cassette cable port. After pointing out their error, I tried them out with the PARSEC module. They are more compact than the older ones, being slightly larger than two swan vestas matchboxes. The fire button is on the top left hand side, rather like the scalectrix controllers used to be. The actual stick part is smaller too. The ball joint at the bottom seems to be slightly larger, giving a wider movement. The top of the stick has a small sphere attached to it, thus not causing any more "congested thumb!"

Yours faithfully,   D. Copeland.

===

17 Wagtail Close, Twyford,
Reading RG10 9ED.
9th April 1983.

Dear Paul,

I would like to take this opportunity to thank you and the other members of the team for the excellent work you have done. Tidings has certainly improved my

knowledge of computer programming — when I bought my 99/4A I did not know the first thing about computers other than the fact that it was a subject that was rapidly becoming an essential part of life.

In my work as a lecturer in Management subjects I was finding that so many of the people I met were meeting with the same problem. The only way to learn anything useful about computers is to use one — you cannot really learn enough from books to understand how they work, you have to experience the problems and frustration caused by putting in the wrong punctuation or word!!!

I could not have made much progress without the aid of your magazine plus the often long-delayed arrival of the 99'er magazine from the States. It was through 'tidings' that I contacted Peter Brooks who has been a great help to me — even to the point of visiting my home and demonstrating some of the modules and peripherals which are available.

Like many other 99'er users I find that the Texas pricing policies are a bit way out when compared with other manufacturers or even to the prices charged by Texas in the States. I would like a disk system and printer but at the moment they are too costly — but appear to be moving in the right direction. In a rash moment I purchased a Mini-Memory module because I had visions of using assembly language programs — the literature did not say that without the Editor-Assembler manual it would be almost impossible. Even programs given in the 99'er do not appear to work for me — so if there is any member of TIHOME who can write programs suitable for the Mini-Memory I would be interested to see them published.

Thank you once again, and please keep up the good work.

Yours sincerely, Allen (Burt).

---

12 Matlock Road,
Watton, Chesterfield.

Dear Mr. Dicks,

I have just read the letter from Karen Warner in the March 1983 issue of tidings. I am writing to tell you that my highest score on the Munchman is 137,910 and I have reached maze 47. I managed this score using the keys which I find much easier than the joysticks.

My highest score in the TI Invaders is 9010 on merely aggressive. My highest score on Tombstone City is 172,900 on day 32 (novice).

My family and I have also managed to find both treasures on the Pirate Adventure game but have not yet worked out how to get the second one.

I like your magazine very much.

Yours sincerely,
Ian Watts, age 12.

71A Biggleswade Road, Upper
Caldecote, Bedfordshire.

Dear Mr. Dicks,

I have just received the March issue of 99'er and will pass on items that might be of general interest.

1.   TI is marketing a Hex-bus Adapter (TI PHP1300 — $59.95) which will allow the "cheap" 99/2 peripherals to be connected to the TI 99/4A.

2.   Several other companies are now marketing peripherals for the 99/4A:

    a.   32K Memory card for expansion box $130.00
    b.   128K Memory card for expansion box $220.00
    c.   32K Stand alone card $160.00
    d.   Single-sided, 40 track drive $199.00
    e.   Double-sided, 40 track drive $265.00

3.   An article in the February issue that the following companies will be making modules for the 99/4A: Walt Disney, 20th Century Fox, CBS, Thorn-Emi, Milton Bradley, Parker Brothers.

4.   Another company "Funware" is already selling modules for the TI 99/4.

5.   I just received a manual from TI ($17.00) called TI-99/4A Console Technical Data. It contains a block diagram, Memory Allocation, Timing Diagrams, Component Location Diagram and Schematic Diagrams for the console.

6.   I have had several occasions to deal with the TI people in Bedford and have talked to several of the dealers in this area and agree with your conclusions that the real "problem" with the TI computer is located in BEDFORD.

<div align="right">Sincerely yours,   Howard M. Slomer.</div>

<div align="right">

35 Manor Cres, Grimethorpe,
Barnsley, S. Yorks S727AE
Tel (0226) 712041

</div>

COUNTY CONTACTS REPORT (Yorks)

Dear Paul,

Not much to report as yet, still it is early days and with telephone contacts, etc. I find that most owners in this area are not TIHOME members. I hope to do something about this in the future.

Most calls I have received have been from owners with the usual difficulties in loading and saving programs. They nearly all finished with the question, "what recorder do you recommend" as I have had no problems with the cheap one I use, a VANICA, and have not used any other make, I would be grateful if anyone with more experience than myself (which is limited), for a list of which recorders are compatable with the TI99/4a, plus which to avoid. I have started a list from info given in back copies of 'tidings', but this is not a complete list.

Another query I will throw for comment, comes from a local radio Ham. He wants information, and advice on 'RITTY' and the TI 99/4A. Has anyone linked the two, if so comments would be appareciated. It seems that 'Sinclair' have a section of their users club for Hams, one can understand this, with Sinclair himself being a Radio Ham, who writes articles in the 'Practical Wireless' on a regular basis. He has a special interest in this section of his users club. Surely, with the TI, being a superior machine to the Sinclair, there are some 'Hams' out there who would like to start a similar section within TIHOME, if so, drop me a line and I'll put you in

touch. While on the subject of radio, I read an interesting article in another magazine. The writer and a couple of friends were both CB Radio freaks, and also had the same make of micro. They experimented on passing programs to each other via their CB radio, and the writer gave a good description on how to do it. I too have a friend who lives about five miles from my home, we both use CB, we made a couple of attempts at passing a program over the airwaves and had no problems at all. With all the interferences, both natural and man made, we get on 27 megs you Hams should find no problems in passing programs over on 2 meters ect, they tell me its interference free up there. Anyone wanting info, a stamped address envelope would be appreciated.

Finally, anyone in my area, who is interested in getting together, to work out mutual problems, or swop programs, or even just to talk shop. Drop me a line, or phone me and we'll try to arrange a venue. If there is enough interest, we may be able to start a local users club. My address is at the top of this report, if anyone is in the area, you're always welcome, the wife makes a good pot of tea.

<div align="center">Sincerely,
Albert Kearford.</div>

<div align="right">6 Cedar Close, Louth,
Lincs LN11 0EH.</div>

Dear Mr. Dicks,

I see that Computer and Video Games have done again something that caused a few moans months ago. As with Stephen Shaw's 'Pompeii' they've printed part of a listing in one colour and put it on the same colour, making it indecipherable (or nearly). In the MAY issue they've printed a program for a ski-ing game in the above fashion. Why is it that they only seem to do this to Texas programs?

After some very close range examination I managed to read the program lines, but then found more than enough bugs and typing errors. I've attached a list of the indecipherable lines and also what I think are corrections of the other faults for anyone who hasn't the time or inclination to sort it out. I don't suppose it's exactly right, but at least the program will run.

Probably we shouldn't criticise too much the errors in the magazines. They do provide, for absolute novices, such as myself, a very useful exercise and give some insight into programming.

<div align="right">Ian Giles.</div>

```
1490    CALL COLOR(11,3,16)
1500    CALL COLOR(10,2,16)
1510    CALL COLOR(13,7,16)
1520    CALL SOUND(100,M.0)
1530    CALL SOUND(100,M+5,0)
1540    NEXT M
1550    CALL CLEAR
1560    CALL COLOR(1,12,9)
1570    CALL CHAR(33,"FFFFFFFFFFFFFFFF")
```

```
1580    FOR X=3 TO 14
1590    CALL HCHAR(X,9,33)
1600    CALL HCHAR(X,25,33)
1610    NEXT X
1620    CALL HCHAR(8,9,33,16)
1630    CALL HCHAR(14,9,33,16)
1640    FOR N=4 TO 9
1650    CALL COLOR(N,2,9)
1660    NEXT N
1670    K$="WE HAVE"
1680    Z=9
1690    GOSUB 1350
1700    K$="A WINNER"
1710    Z=11
1720    GOSUB 1350
1730    INPUT "DO YOU WANT TO FLY AGAIN ":T
1740    IF T$="Y" THEN 120
1750    STOP
1760    CALL HCHAR(X,Y,E)
1770    FOR N=170 TO 220 STEP 8
1780    CALL SOUND(100,N,0)
1790    CALL SOUND(100,N-50,0)
1800    NEXT N
1810    FOR N=5 TO 9
1820    CALL COLOR(N,2,16)
1830    NEXT N
```

****

```
220     CALL CHAR(105,"303098781810907F")
280     FOR N=0 TO 40
350     FOR N=0 TO 30
        Delete Lines 390 to 410 incl.
720     IF C=120 THEN 2030
730     IF (C=70)+(C=72)+(C=73)+(C=78)+(C=83)THEN 1470
770     CALL HCHAR(22,2,128,32)
780     CALL HCHAR(22,2,129,32)
980     CALL CHAR(35,"3C4299A1A199423C")
1410    DATA 5,30,1,7,30,2,2,5,3,28,30,4,27,30,5,1,3,6,9,10,
        6,22,6,29,30,6,1,23,7
1420    DATA 30,30,7,1,7,8,2,3,8,30,30,8,1,6,9,17,22,9,1,4,9,
        30,30,9,1,1,10,30,30,10,1,1,11,29
1430    DATA 30,11,1,1,12,10,11,12,29,30,12,1,1,13,9,14,13,22,
        30,13,1,1,14,5,15,14,21,30,14,1,1
1450    DATA 3,8,3,5,5,5,5,8,2,11,4,11,4,17,6,17,3,22,5,22,7,24,
        7,28,9,28,9,25,12,23,10,23,11,19
1460    DATA 13,19,10,15,12,15,8,10,10,10,10,7,12,7,13,2,13,4
        16,4,16,2,17,9,19,12,17,12
1740    IF T$="Y" THEN 120
2260    CALL HCHAR(X,Y,120)          2420  IF T$="Y" THEN 120
```

5.5.1983.

Feedback, Practical Computing,
Quadrant House, The Quadrant,
Sutton, Surrey SM2 5AS.

Dear Sir,

TI 99/4A games — May 1983.

I have designed microcomputers, I have built microcomputers, I write about microcomputers, and I have twelve computers at the present count, including the TI 99/4A, but I was incensed by the load of rubbish, which purported to be a review, written by Jack Schofield. My two boys, aged now 18 and 20, are my experts when it comes to games testing, having spent their formative years commuting daily via Waterloo on their way to school, and Waterloo housed Space Invaders.

I have also had Atari Invaders at home for a good two years, but my boys tell me that in their opinion the TI Invaders is easily the best. Schofield's description of the game shows that he only has the slighest acquaintance with the TI version, in that he never discovered the 16 other hoards of invaders which descend on successive screens, the blue demons, the white demons, the flashing green demons, each wave more and more difficult to destroy. My experts tell me that TI is much more dem anding than Atari. To the best of TI's knowledge, no one has yet reached the 17th screen.

TI's invader movement is not slow, the implementation is written in machine code and the speed of movement is slowed by delay loops so that the player has a chance of getting beyond the second wave. And the "downright nasty" is exactly that, the beasts know where you are, and from past experience they know where you are going, and they aim at you rather than drop their bombs at random.

Why is no mention made of Parsec, a space-craft game with outstanding graphics and demanding enormous skill to outwit and destroy the enemy? If you add the TI speech synthesiser, your on-board computer talks to you and warns of approaching dangers. It has a delightful female voice, surely unique in home computer games. And that is in addition to the battle sounds which the game also generates.

The 99 was designed as a family home computer, and the very great range of software available reflects the same philosophy, it caters for all the family from the very youngest to the oldest, male and female. Do not expect, Mr. Schofield, to have to be drunk to enjoy some of the games, you only have to be young enough. To judge them all from the one standpoint is a pointless exercise.

As to that bit at the start which suggests that shortage of independent software is a limitation, where did Mr. Schofield get that bit of mis-information from? In TI's software catalogue, which is already way out of date, I counted 32 under the TI Logo and 240 from independent organisations, in the Entertainment section alone, and there is a huge amount of software in the four other sections. How about the many adventure games by Scott Adams? What do you say to the.....

Such well-known names as Scott Foresman, 20th Century Fox, CBS, DLM, Collins, Thorn-EMI, Milton Bradley, and Parker Brothers will be producing software for game cartridges running on the 99. Are they not independent? Then there is the monthly glossy "99'er", a quality magazine devoted entirely to programming aspects on the 99, with many program listings in each issue, and containing advertisements from many of the independent software and hardware producers.

No, Mr. Schofield, yours was a very poor review, and there are 1,500 members of the TI user's club in Britain who know that you were wrong. The 99 is a very powerful home computer with a very extensive range of good software applications, and capable of a great deal of hardware expansion. One American company is now offering 40 Megabyte hard disk for the 99/4A, and the 99 can make very good use of it.

Finally, this letter was produced on, and printed from, my 99/4A, using another of those cheap plug-in cartridges called the TI-WRITER. And for those who are interested, my machine has twin discs, extended memory, serial and parallel interfaces, speech synthesiser, modem communications facilities, and I program in Assembler and FORTH. Not at all bad for a machine whose basic price is a hundred and fifty pounds.

<div align="right">

Yours sincerely,

Frank Dale.

</div>

<div align="right">

62 Egerton Avenue, Hextable,
Swanley, Kent BR8 7LQ.
1st May 1983

</div>

Firstly, congratulations on the new look 'Tidings'. It is bigger, better AND more legible..... a superb effort!

Next, belated thanks to Paul Dicks for his invaluable assistance in getting my strange little program into print in the April edition of Personal Computing Today. At the time, I had no printer, and Paul was most kind in ensuring that the computer print-out (which P.C.T. prefer) reached them by their deadline. My apologies, Paul, for having to 'phone you 3 times in my panic to make sure everything was O.K.!

To those people who saw the article in P.C.T. (and probably didn't understand a word of it!) also go my apologies. I am quite sure that my instructions and explanations leave a lot to be desired. I would have preferred it also if the magazine had printed a little more of the text that I sent with the program. This would have explained that the program was in fact written for a friend's 12 year old son, and was only sent to P.C.T. because its rather frivolous content seemed appropriate to the season (t was sent off before Christmas!) It would also have mentioned the fact that I am (as is probably only too obvious) a very inexperienced programmer. Take heart, therefore, all you newcomers; if you can write a program as badly as I, provided you have something a little unusual to offer, you too may find yourselves in print, with the benefit of the not inconsiderable financial reward as well! In my case, P.C.T. probably liked the idea of me putting my own money on the line! As it turned out (and as I expected) nobody claimed my £10, in all likelihood due, once again, to those rotten instructions of mine!

Best wishes and Happy computing.

<div align="center">

David Vincent.

</div>

ASTASOFT TI-99/4A SOFTWARE

STARWARRIOR: An EPIC ADVENTURE for one to four players. Travel the GALAXY in search of powerful CRYSTALS with which you may enter the STRONGHOLD of the SCAILIENS in order to DESTROY them. On your VOYAGE you will encounter many dangers include: SPACE PIRATES, SLAVERS, DRAKWOLDS, SLIME SLUGS, and METEOR STORMS. You may purchase FOOD, FUEL, CREW, etc. or hire a doctor in case of SPACE PLAGUE. If you are desperate you may obtain help at a MISSIONARY OUTPOST. The GALAXY is different each game and the permutations are almost infinite. FOUR LEVELS. £7.00.

Send a crossed cheque or P.O. made payable to E.A. STANLEY, to

ASTASOFT, c/o 4 VICTORIA COURT, 38 VICTORIA ROAD, SOUTH WOODFORD, LONDON E18 1LH.

ARCADE HARDWARE

GENUINE ARCADE JOYSTICKS FOR THE TI 99/4 and 4A

These joysticks are the very same as used on thousands of Arcade machines throughout the country. In normal (and abnormal) usage they have proved indestructible, whilst retaining the virtues of a joystick, i.e. they should move in the direction wanted immediately, with no stiffness or worries about breakage.

These joysticks are made from a steel frame and shaft, with microswitches designed for a life greater than 10,000,000 operations. The only penalty you pay for this robustness is in looks. One thing the Arcade Hardware joystick cannot claim is that it is pretty. Mounted using bolts in a sturdy but unpretentious grey plastic case, nobody will admire it for it's aesthetic appeal. It will be admired by the person who wants to play to the best of his or her ability without the limitations imposed by certain mass produced joysticks.

The Arcade Hardware joystick (single unit only) is available by mail order from:

ARCADE HARDWARE, 211 Horton Road, FALLOWFIELD, Manchester, M14 7QE.

Price £19.50 inclusive of tax and postage.

NEW NEW NEW NEW NEW NEW NEW NEW NEW NEW NEW NEW NEW

Thorn EMI have finally released two new modules for the TI 99/4A.

These are: Submarine commander
                River rescue

Both these modules are now available from Arcade Hardware priced £24.50 each inclusive of postage and packing.

Arcade Hardware promise none of the delays that have made ordering goods by mail such an unpleasant and frustrating business. If you don't receive your package within 14 days (usually 4) I promise to return your cheque with the goods ordered.

TX SOFTWARE, 109 Abbotsweld, harlow, Essex CM18 6TQ.

Four in a Row
Often called Connect Four. The program allows the computer to play a fast, challenging game. A two player option is also included. Full graphics and sound are included.

Battlefront (Extended Basic required)
A fast moving 'arcade' type game involving a three dimensional display, tanks and guided missiles. Fire and be fired at. This game compares with Atari and Apple programs in terms of speed and graphics. Either joysticks or the keyboard may be used with this game.

This tape also includes FREEWAY and TARGET. These Extended Basic games are as equally exciting. FREEWAY involves manoeuvres on a six lane highway. Use the joystick to change lane, speed up or slow down. Again there is a 3-D display. TARGET is a shorter program that portrays a city with tanks roaming the streets. An extended version of the program recently published in Computer and Video Games.

Wargame
A detailed map is displayed on the screen with hills, water, forests and marsh. Two opposing armies face each other. You may play against the computer at different levels of skill or play against another player. Many tactical and strategic facets as you manoeuvre your tanks and artillery across the terrain with the object of destroying the enemy bases. Hills increase the range of your guns, but slow you down, woodland is good for defence etc. Every game will be different.

Joysticks are not essential, but are recommended for moving and firing. To fire and hit requires real skill, especially if the range is great. Instructions included.

Chess File
A program for serious players. It does not play, but allows games to be played on the screen using high resolution graphics. Tape files may be constructed of opening variations and games, which can then be replayed in any order. Special storage techniques allow up to 30 complete games and 100+ openings to be stored in memory at any one time and replayed in any order. An excellent learning tool with totally clear and unambiguous graphics. An instruction program is included with the main program.

German Whist
The computer plays a skilful version of this two player card game. Considerable ability is required to play better than the computer. Bridge and whist addicts generally will find this game absorbing. Features include single key entry when playing cards and very clear graphics. Speech is also provided for those with appropriate facilities, although the program will also run on the unexpanded machine. The ability to replay hands means that you can continually try to improve your performance.

3–D Maze
The computer displays a three dimensional view of a random maze. The object is to find your way through different levels of the maze using the clues provided.

However, time is not on your side. A two-dimensional view of a limited area around you is also available... at a cost. Graphically excellent.

Graphic Pairs
Based on the traditional memory game of pelmanism, but with a difference. A graphic treat as you attempt to use your memory to beat the cunning computer in a crazy car race. The penalty of failure is amazing. This program includes the TX skill meter which makes it suitable for everyone from four year olds to those with photographic memories. The graphics include verything from spaceships to robots.

Hangman
Everyone knows this game, but this program turns it into a challenging and educational experience. A vast vocabulary is built into the program and it is also possible to create your own data tapes of words. Thus it is possible to specialise in certain subjects. An ideal learning tool for children, but at its highest levels can outplay the smartest adults. As for the graphics....

All programs are recorded on quality cassette tapes and loading should be satisfactory on any reasonable cassette player.

The above programs will be added to in the future, but only by the addition of top quality programs. TX Software programs will continue to be highly selective.

Prices are as follows:

TX1	Four in a Row	£4.00	TX5	German Whist	£4.00
TX2	Battlefront	£6.00	TX6	3–D Maze	£5.00
TX3	Wargame	£6.00	TX7	Graphic Pairs	£5.00
TX4	Chess File	£5.00	TX8	Hangman	£5.00

In addition the following package is being sold:

TX1, TX5 supplied on one cassette   £6.00

All of the above prices are fully inclusive and include documentation where necessary. All programs are of course guaranteed.

T.C.H.S. TI EXCHANGE

40 BARRHILL, PATCHAM, BRIGHTON, SUSSEX BN1 8UF.

WHY CONTINUE TO BUY EXPENSIVE MODULES WHEN YOU CAN JOIN BRITAIN'S FIRST TI/99/4A SOFTWARE EXCHANGE CLUB?

Just write to me enclosing S.A.E. telling me what T.I. modules you have to exchange and we will advise you when an exchange can be made. We ensure all modules are in good condition before exchanging. Only TI/99/4A modules complete with instruction booklet are accepted. NOTE: WO DO NOT EXCHANGE CASSETTE PROGRAMS.

* *

DUST COVERS: to protect your TI 99/4A computer from that enemy of electronics @ £3.50 + 25p p&p.

CASSETTE LEADS: CS1 only @ £5.95 incl.

COMPUTER CASSETTES: Top quality C12 in library cases — three for £1.95 + 43p p&p.

APEX SOFTWARE: We supply these adventure and games tapes to TI HOME members.

NEW*NEW*NEW*NEW   **THE TEXAS PROGRAM BOOK by Vince Apps**
                                        (Apex Trading)

Contains 35 programs for games, home and business use with the TI 99/4A written for the home user. These programs are both fun and educational.

                    Price: £5.95 + 35p p&p.

WRITE TODAY: T.C.H.S. TI EXCHANGE,
                    "FORTY", BARRHILL AVENUE,
                    PATCHAM, BRIGHTON,
                    SUSSEX BN1 8UF.

Please enclose S.A.E. and advise us of your requirements for exchange/purchase.

TCHS TIHOME TCHS TIHOME TCHS TIHOME TCHS TIHOME TCHS TIHOME

2021 note: When TIHOME was passed to TIHCUC, many TIHOME members moved to TCHS (run by Clive Scally) who published TI*MES - still in publication in 2021

CLOSE ROUTINE

Well, there we are, another issue of Tidings. I hope you found something in it for you. Here are some more County Contacts:—

HEREFORED & WORCESTER
M.A. Morris, 32 Eastwood Drive, Kidderminster, DY10 3AW.   0562-60408

NORTHANTS
John Carter, 16 Sherwood Ave, Northampton, NN2 8PZ.   0604-842760

WILTS
David Kimberlin-Wyer, 39 River Park, Marlborough, Nr.Swindon.   0672-52548

SURREY
E.G. Faulkner, 39 Oakwood Avenue, Mitcham, Surrey.   01.640 4778
David Burch, 57 Dorling Drive, Ewell, Surrey.

AVON
Brian Boyde-Shaw, 7 Riverway, Nailsea, Avon, BS19 1HZ.   0272-851337
Gary Harding, Flat 2, 12 Belmont, Lansdown Rd, Bath, BA1 5DZ.

MIDDLESEX
P.J. Walker, 1 Joanne Ct., 2A Queen Annes Gdns, Enfield, EN1 2JQ.

HERTS
G.L. Jones, 4 Grovelands Avenue, Hitchin, Herts, SG4 0QT.

GLAMORGAN
P.W. Boswell, 103 Roundwood, Cardiff, CF3 7PH.

DYFED
Ross O'Neill, 2 Westfield Dr, Honeyborough, Neyland, SA73 1SB.

TYNE & WEAR
David O'Doherty, 23 Newlyn Dr, Bilton Hall Est., Jarrow, NE32 3TW.
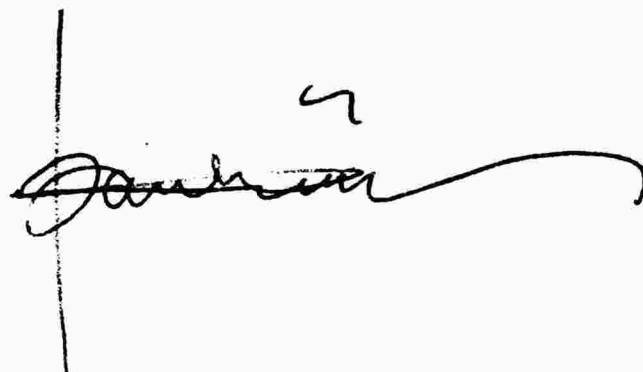
NORTH NOTTS
James Davies, 18 Huthwaite Road, Sutton in Ashfield, NG17 2GW.

WILTS
Bruce Merrett, 2 Harlestone Road, Stratton St. Margaret,
                         Swindon, SN3 4ED.   0793-822639

Good luck and Best Wishes, and may your 99 never go TILT.

Yours