THE AMSTRAD USER

Happy Birthday to

— THE —
AMSTRAD
USER

3 YEARS OLD

- *New CPC Series: Graphics Programming - Viatel - Word Processors + Pedigree breeder's database type-in*
- *New PCW Protext series + PIP under CP/M Plus + PCW Finance Program type-in and more!*
- *Compatibles Corner + PC Help*

FOR THE NOVICE & EXPERIENCED USER

PACTRONICS CLASSIC WYSIWYGS

MASTERSCAN

A superb Scanner with which you can reproduce any picture on your PCW screen and print out as often as you like. Simple to use, the optical scanner clips onto the head of your printer in seconds. You can then incorporate these pictures in DESK TOP PUBLISHING programs.

KEMPSTON GRAPHICS MOUSE

A precision made, high resolution, two button Mouse. Dedicated Mouse Interface, GSX and RSX Mouse driving software, compatible with CP/M programs such as *DR. DRAW, DESK TOP PUBLISHER, WRITE HAND MAN, FLEET STREET EDITOR*, etc. and UNBELIEVABLY!!! these above two items from Pactronics include, as a bonus . . . .

. . . . MASTERPAINT

The total package to produce quality artwork and icons or pull-down menus. Scroll picture for larger images. 16 patterns, text fonts. Air brush for subtle shading. AND YOU CAN USE ANY OF **MASTERPAINT'S** TOOLS TO MANIPULATE AND FINE TUNE IMAGES FROM **MASTERSCAN.**

DESK TOP PUBLISHER

Just imagine what you can do with DESK TOP PUBLISHER with either or both **MASTERSCAN**, **MASTERPAINT and MOUSE**. Need we say more.

And don't forget Pactronics great range of other products - educational for the PCW, PC and CPC. Also our great range of new games - **TEST CRICKET** and **CLASSIC QUEST** Adventure Games, and last, but not least . . . .

. . . . PLAN-IT

Now you can use the power of your computer to organise your activities in a way never possible before. With **PLAN-IT**, a few keystrokes are all you need to keep track of your money, plan your budgets - and manage your time far more effectively. And you'll find it's so very simple to use!

Available from Grace Bros, Chandlers and computer retailers around Australia.

FOR THE NEAREST RETAILER IN YOUR STATE CONTACT
NSW: Pactronics Pty. Ltd., 33-35 Alleyne Street, Chatswood. (02) 407 0261
VICTORIA: Pactronics Pty. Ltd., 51-55 Johnston Street, Fitzroy. (03) 417 1022
QUEENSLAND: Pactronics Pty. Ltd., 12 Stratton Street, Newstead. (07) 854 1982
SOUTH AUSTRALIA: Baringa Pty. Ltd., (08) 271 1066 ext. 6132
WESTERN AUSTRALIA: Pactronics WA, 757 Canning H'way, Ardross. (09) 332 3807
MAIL ORDER: "Software To Go" - (02) 457 8289

Pactronics

THE AMSTRAD USER

Issue No. 37
February 1988

For Tape Subscribers, CPC programs appearing in this month's magazine can be found at the following approximate positions:

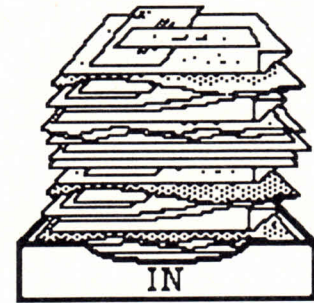|							
Side 1:	PEDIGREE	- 6	DIASPIN	-79	PLOTTER	-92
	SRCOLBIG	-103				
Side 2:	RELAB	-6	SC-HOUSE	-24	FORMATS	-36
	VORTAC	-84				

Letters

May I make a suggestion to all current and would-be contributors of software to TAU? (I will assume a favourable response.)

I occurs to me that a common approach in the program submitted may be beneficial to those readers of this magazine who transcribe programs from this magazine. I don't wish to suggest that my approach is the only one but it works for me and may serve as a starting point. "The point?" I hear you ask. There are several:

1. All programs should provide a return to either the operating system or to Basic rather than having to resort to the use of the STOP [Esc] key.

No, we are not starting a cookery column!

If you haven't already guessed by the picture on the front cover, we are entering our fourth year of publication.

1988 promises to be just as exciting, challenging and hard work as previous years.

With the national celebrations taking the forefront this year, we will also be celebrating the fact that your magazine now reaches over 8000 Amstrad users each month.

It is a fitting time to thank all contributors, past and present, our large readership and advertisers for their support.

2. Programs that reset the terminal characteristics, e.g. disable the cursor, set on reverse image, etc., should return the terminal to the default setting on exit; and

3. All programs should provide a means of terminating the program prematurely rather than by abuse of the STOP key, which has no effect if the OPTION RUN command is used.

The following program segments give the general idea:

```
200  REM * ACCEPT MOVE
210  INPUT a$; a$=UPPER$(a$)
220  IF a$="Q" THEN GOTO 1000
1000 REM * EXIT SUBROUTINE
1010 PRINT "<A> - Another game
     / <Q> - Quit"
1020 INPUT resp$; resp$=UPPER$
     (resp$)
1030 IF resp$="Q" THEN PRINT e$;
     "e";e$;"q";e$;"u";END
1040 IF resp$="A" THEN GOTO 10
1050 GOTO 1000
```

NB. "e$=CHR$(27)" needs to be included in the initialisation section of the program, ie. prior to using line 1030.

All the Basic games programs I have collected are located on one diskette together with a menu program. I also substitute 'RUN "MENU.BAS"' for "END" (line 1030 in the above example). As the menu program searches for programs with filetype ".BAS" and runs the selected option, I am able to run all of the programs on the diskette from the main menu and return to that menu on completion of the game. Included on the diskette in END.BAS which contains only one line "10 END". The presence of this program provides me with an option to exit to Basic if

required. The menu program includes the command OPTION RUN.

C.J. Faber, Croydon Park, NSW

The following modification to the MISER program published in the December 1987 issue is offered for readers who use an external cassette deck with a remote jack which the computer can use to control the cassette drive motor. I do not like having to remove the plug from this jack every time I want to fast forward, or rewind the tape; broken wires are eventually going to result from such manipulations. Three lines need to be altered, and four lines added to provide keyboard control on the CPC6128, as follows:

```
770  x=9:y=11:GOSUB 925
777  x=9:y=13:GOSUB 925
785  x=9:y=15:GOSUB 925
786  x=9:y=17:a$="8....MOTORON":GOSUB
     925
787  x=9:y=19:a$="9....MOTOROFF":GOSUB
     925
831  if a$="8" then out
     63200,16:GOTO 810:'Turn
     motor on
832  if s$="9" then out
     63200,239:'Turn motor off
```

There are two other observations about this program which may also prove useful to your readers. My cassette deck could not cope with the speed setting in line 105 of the published program; read errors occurred when attempting to recreate a disc from tape. Deleting this line cured the read error problem, but was excessively slow. Replacing with

```
105 SPEED WRITE 1
```

has so far proved quite successful.

Secondly, I do not trust my memory to faithfully recall the name that I used when archiving a disc after several months have passed. Of course, I should have written it down, but . . . . Fortunately, the sector map is saved as a standard AMSDOS tape file and the name can be determined by │TAPE.IN:CAT.

R.J. Williams, Black Forest, SA

I have a copy of Tascopy 464 and can

make very good screen dumps from graphics programs that I have written or taken from your magazine.

I have purchased games such as Gauntlet, Barbarian, Druid etc., and I would like a screen dump of the title screens.

How do I go about getting a dump onto my DMP-2000?

Jason Williams, Dianella, WA

Most games today are so long that there is a need to have the title screen separate from the main program. If this is the case there is usually a basic program with the name of 'DISC'. If you load this program and list it you will find where the program loads up the title screen. After you have found that line, insert the printer dump command, in your case '│DUMP'. This should load the title screen then dump it to the printer. Sometimes the basic program may be protected, in that case you will need to write your own screen loader.

This should explain why the CPC464

takes a long time to OPEN tape/disc buffers. (Anthony Trost's letter in TAU, December 1987).

Strings are stored in memory starting at HIMEM and growing downwards towards the variable store. If the first string used is "ABC", "A" will be at HIMEM-2, "B" at HIMEM-1, and "C" at HIMEM, with the string pointer stored in the variables area which starts at the end of the program. Either OPENIN or OPENOUT will reserve both an input and an output buffer of 2k each in the space HIMEM-4096 to HIMEM. The strings must be shifted down in memory to clear this space, so a garbage collection (consolidation of the string space) is performed, the strings are shifted, and their pointers are adjusted to reflect the new location.

The time taken to do all this is roughly proportional to the square of the number of active strings. A CLOSEIN/ OUT will shift the strings back to HIMEM if both buffers are closed.

The suggested procedure for avoiding the delay is to reserve the two buffers permanently at the beginning of the program:

CLASSIES

After all the cries of anguish when we published our intention __not__ to include classified ads, we expected a flood of $7.50 ads when we changed our minds. Perhaps we were right in the first place judging by the number appearing below. Still, we publish as promised, but now it's up to you - no input, no classies.

RE-INK your printer ribbons and save $$$. Send ribbon cartridge, $5 plus return postage to PO Box 126, Wardell, 2477, or write for more details. Prompt service.

FAMILY TREE for CPC6128, 5 generation trees to screen or printer. List cousins. Epson compatible. Uses BANKMAN. 3" disc, $25. 5.25" disc (40 trk), $20. Jeff Raphael, 3 Field of Mars Ave, Turramurra 2074 (02) 449 6537

```
10   'Suggested procedure for
     CPC464
20   CLOSEIN:CLOSEOUT 'abandon
     any open buffers
30   SYMBOL AFTER xx 'if re-
     quired, it must be done
     first
40   |TAPE 'no need to access the
     disc
50   OPENIN"" 'open buffers
60   MEMORY HIMEM-1 'move memory
     pointer while buffers are
     open
70   CLOSEIN 'buffers closed but
     remain available to OPEN
80   |DISC 'back to disc
```

This reduces the space available for string storage so that the garbage collection will occur more frequently during normal running, but there will be very little delay when the buffers are required. There is always a trade-off. I think the same procedure could be used on the 6128, but it would be of no benefit.

Petr Lukes, Toowoomba, Qld

In the January issue on page 13 (Galli-maufry VIII), was a Character by Character listing by B. Naff. I would like to submit the proggy again with some changes. Mr. Naff has a fine routine, but it is a pain having to type all the output into data statements. I thought that it would be far better to read all the output from disc instead - the advantage need not be stated - so I have altered the program accordingly. In line 10 I have removed RESTORE.
In line 20 the file to be read.
Line 30 reads until End of File.
Line 40 file contends into memory buffer. (Long files will be read in stages).
Line 50, 60, 70 ,80 ,90 and 100 are as in the original listing.
Line 110 all removed except WEND.
This is what it should look like:

```
10 MODE 1:CLS:ENV 1,15,1,7
20 OPENIN "b&a.
30 WHILE NOT EOF
40 LINE INPUT#9,a$
50 FOR t=1 TO LEN(a$)
60 PRINT MID$(a$,t,1);
70 FOR delay=0 TO 50
80 NEXT delay
90 IF MID$(a$,t,1)<>" " THEN
SOUND 1,50,2,10,1,1,1
100 NEXT t
110 WEND
120 CLOSEIN
```

This listing is without the REM state-

---

ments in lines 10,30,70 and 130 which appeared in the original. I used Tasword for my input file, but any WP can be used as long as proper ASCII files are produced.

Add border, paper and pen colour to choice for a fine routine to send for disc letters to friends.

P. Kent, Howrah, TAS.

Paul Kent (or is it Pam - sorry we can't read the signature) also sent in a solution to Anthony Trost but, unless we missed the point, we cannot see how it would dramatically speed things up. The suggestion was to replace the OPEN and CLOSE commands of both files with CALL commands, eg. CALL &BC77 for OPENIN and CALL &BC8C for OPENOUT. A CATALOGUE of the disc can be achieved with CALL &BC9B.
While writing, Paul/Pam asks if any knowledgeable programmer can provide a full explanation (with examples) of the OUT command.

464/6128 GREMLIN STRIKES AGAIN

More haste, less speed! We failed to check a couple of programs on the 464 before putting them to print.

BLACKOUT (Page 18/Jan 88)
Most subscribers received an amendment to this program with their January magazine. If you're not a subsciber and own a 464 you will want the following information to make it work.
Amend these lines as shown:
120 to read PLOT 780,0,2
601 to read PLOT 780,0,0
606 to read PLOT 780,0,2
721 to read PLOT 780,0,0
726 to read PLOT 780,0,2
990 to read PLOT 780,0,2
1310 to read PLOT 780,0,1

RNDXMAS [Revolving Message]
(Page 21/Dec 87)
Change the word "MOVE" to "PLOT" in line 40.

We eat humble pie!

Nationwide User Groups

As we enter out fourth year of publication, we take this opportunity to thank all past and present officers of User Groups, both home and abroad, for donating part of their time in helping to make the Amstrad User Group network one of the largest personal computer organisations in the country.

WESTERN AUSTRALIA

ALBANY AMSTRAD USER GROUP
President: Gerry Barr (098 41 6884)
Secretary: Steven Hands (098 44 7807)
Treasurer: Gavin Grose
Venue: Priess Street Centre, 14 Priess Street, Albany on the first and third Mondays of each month at 7.00 pm.
Mail: 20 Anuaka Road, Albany, WA 6330

AMSWEST (Perth)
President: Carl Hindle (09 419 1411)
Vice Pres: John Lansdown (09 342 3154)
Secretary: Saskia Quinn (09 444 8147)
Treasurer: Mario Ioppolo (09 444 7691)
Venue: Royal Institute for the Blind, cnr. Whately Cres. and Guildford Road, Maylands on the first and third Tuesdays of each month starting at 7.30.

AMSWEST (Blackwood) USERS GROUP
This small group is affiliated to AMSWEST (Perth). For more details contact George Muscat on (097) 61 1488.

ROCKINGHAM-KWINANA AMSTRAD USER GROUP
President: Bob Harwood
Vice-Pres: Keith Saw
Treasurer: Rob Macilroy
Secretary: Ben Hille (095 27 5246)
Venue: Cooloongup Primary School, Westerly Way, Cooloongup (Rockingham), every second Wednesday at 7.30 pm.
Mail: 29 Milgrove Ave., Cooloongup, WA 6168

SOUTHSIDE AMSTRAD USER CLUB
President: W. Van Der Kooi (09 271 1085)
Secretary: Steve King (09 354 2068)
Treasurer: Eric Tytherleigh (09 390 8865)
Venue: Huntigdale Primary Sch., Matilda Rd. Huntingdale every 2nd and 4th Wednesday of each month from 7.00 pm.
Mail: The Sec., Southside Amstrad Users Club, 61 Keslake Way, Parkwood, WA 6110.

AMSTRAD COMPUTER CLUB TOM PRICE
President: Colin Smith (091 89 2074)
Secretary: John Eliot (091 98 1735)
Treasurers: P. & C. Montgomery (091 89 2398)
Venue: Primary School every 2nd Wednesday night. Contact the above for more details.

SOUTH AUSTRALIA

AMSOUTH AMSTRAD USER'S GROUP
President: Drew Ames (085 371 0151)
Treasurer: Bob Bleachmore (085 56 2048)
Secretary: Ross Kennewell (08 386 2737)
Venue: Christies Beach High School, Western Section, Beach Road, Christies Downs (adjacent to Staff Car Park off Mander Road) every 2nd Wednesday of each month at 7.30.
Mail: PO Box 612, Noarlunga Centre, SA 5168

AMSNORTH AMSTRAD USER'S GROUP
Organisers: J.T. Clarkin (08 262 6342)
R. Britton (08 258 7861)

Venue: Lacrosse Hall, Terama Street, Gepps Cross every Wednesday at 7.00 p.m.

AMSTRAD COMPUTER CLUB INC. (SA)
President: Frank Matzka (08 382 2101)
Vice-Pres: Andrew McDade (08 79 5414)
Treasurer: Les Jamieson (08 356 9612)
Secretary: Ross Barker
Venue: Church Hall, 15 Clayton Ave, Plympton between 6.30 and 9.00 each Tuesday.
Mail: PO Box 210, Parkholme, SA 5043

NORTHERN COMPUTING SOCIETY INC.
President: Grant Wilson (08 250 2760)
Treasurer: Percy Cook (08 248 1065)
Secretary: Judith Thamm (085 20 2377)
Venue: Salisbury North Primary School, cnr. Bagster & Woodyates Rds every Wednesday from 7.00.
Mail: PO Box 269, Two Wells, SA 5501

PORT LINCOLN AMSTRAD USERS GROUP
Contact: Rita Bascombe (086 82 1633)
Venue: Third Tuesday of each month from 8.00pm. Ring above number for address.

PORT PIRIE AMSTRAD USER GROUP
President: Doug Gowers (086 36 5206)
Treasurer: Dave Green (086 32 6834)
Secretary: Tim Eckert
Youth Rep: Mark Fusco (086 36 2452)
Venue: Education Ctr, 370 The Terrace, Port Pirie every 2nd and 4th Monday from 7.30 pm.
Mail: The Pt. Pirie Amstrad User Group, c/o D.T. Green, 207 Senate Rd., Pt. Pirie, SA 5540.

SOUTH EAST AMSTRAD USER GROUP (SA)
Contact: Neil Taylor (087 25 8068)
Venue: Mount Gambier from 1.00p.m. to 4.00p.m. on the 3rd Sunday of each month. Ring above number for address.

NORTHERN TERRITORY

DARWIN AMSTRAD USER GROUP
President: Kevin Bateman (089 32 1463)
Treasurer: Jeff Powis (089 27 5557)
Secretary: Kiem Le (089 32 1828)
Venue: Meetings are held twice monthly. Contact any of the above for more details.
Mail: 45 Priest Circuit, Gray, Palmerston, NT 5787

VICTORIA

CENTRAL AMSTRAD USER SOCIETY
President: Fred Gillen (03 580 9839)
Vice-Pres: Dennis Whelan (03 367 6614)
Treasurer: Doug Jones (03 560 8663)
Secretary: Craig Tooke (03 359 3736)
Venue: Corner of Church and Somerset Sts, Richmond on the first Sunday of each month commencing at 1.00 and generally twelve days later on a Friday evening at 7.00.

EASTERN AMSTRAD USER GROUP Inc.
President: J.L Elkhorne
Secretary: Bob MacDonald (03 878 7783)

Treasurer: Ron Dunn (03 277 7868)
Venue: St. Ninian's Church Hall, cnr. McCracken Avenue and Orchard Grove, South Blackburn on the 1st Sunday of each month from 1.00pm.
Mail: R.D. MacDonald, 6 Ashwood Drive, Nunawading, Vic 3131

GEELONG AMSTRAD USER CLUB
President: Reg Morse (052 43 3239)
Vice-Pres: Arthur Pounsett (052 78 2160)
Secretary: Ron Butterfield (052 50 2251)
Venue: South Barwon Community Services Ctr, 33 Mount Pleasant Road, Belmont on the first Wednesday of each month, starting at 7.30p.m.

GOULBURN VALLEY AMSTRAD USERS CLUB
President: Shad Aitken (058 52 1001)
Sec/Treas: Bill Brown (058 21 7569)
or (058 22 1011)
Venue: 98 Nixon Street, Shepparton on the first floor every third Wednesday from 7.30 pm.

LATROBE VALLEY AMSTRAD USER GROUP
President: Stan Hughes
Secretary: M.G. Donaldson (051 345 711)
Venue: Morwell Neighbourhood House, 17 Symons Crs., Morwell on the first Thursday of each month at 7.30pm.
Mail: PO Box 947, Morwell, Vic 3840

MARYBOROUGH AMSTRAD USER CLUB
President: Chad Banfield (054 68 1351)
Treasurer: Brendan Severino (054 61 3191)
Secretary: J. Fothergill (054 75 2667)
Venue: Maryborough CCC each week on Tuesday from 12.10 p.m. to 12.45 p.m.

MOUNTAIN DISTRICT AMSTRAD USER GROUP
President: Ian Poli (03 758 5282)
Treasurer: Lindsay Bell (03 758 9921)
Venue: Country Womens Association Hall, 4 Sundew Avenue, Boronia from 7.00 p.m. every second Monday of the month.
Mail: PO Box 132, The Basin, Vic 3154

NORTHERN AMSTRAD USER GROUP
Contact: Brian Ellis (03 469 4425 A/H)
Venue: Every three weeks in Brunswick West for CPC owners with a sincere interest beyond games.

SOUTHERN AMSTRAD USER GROUP INC.
President: Noel Sheard (03 786 5469)
Secretary: Bob Patterson (03 786 6976)
Treasurer: Christine Donaghey
Venue: Senior Campus at John Paul College, Frankston every third Tuesday from 7.30p.m.
Mail: The Sec., PO Box 100, Seaford, Vic 3198.

SUNBURY MELTON AMSTRAD USER GROUP
Contacts: Wayne Urmston (03 744 2719)
Norma McEntee (03 743 7104)
Venue: Contact above for more details.

WENDOUREE AMSTRAD USER GROUP
Contact: Brad Maisey (053 44 8356)
Venue: Cnr. Charles and Appleby Drive, Cardigan Village on the first Sunday of the month at 3.00 pm.

WESTERN COMPUTER CLUB
Venue: Fairbairn Kindergarten, Fairbairn Road, Sunshine on alternate Tuesdays from 6.30 pm.
Mail: PO Box 161, Laverton 3028.

ACT

CANBERRA AMSTRAD USER'S GROUP
Convenor: Paul Kirby (062 86 5460)
Secretary: Michael Hickey (062 58 5719)
Treasurer: Rod MacKenzie (062 54 7551)
Venue: The Oliphant Building, ANU, Canberra on the first Wednesday of each month from 7.30 pm.
Mail: PO Box 1789, Canberra , ACT 2601.

NEW SOUTH WALES

AM-USER's (North Ryde)
Contact: Lawrence Walters (02 888 1898)
Venue: Meeting Room at 2 Leisure Close, North Ryde from 7.30 p.m. on the first Tuesday of each month.

BLUE MOUNTAINS AMSTRAD USERS
President: Bob Chapman (047 39 1093)
Vice Pres: Dennis Shanahan (047 39 4568)
Treasurer: Peter Traish (047 53 6203)
Secretary: Christine Preston (047 51 4391)
Venue: Springwood Neighbourhood Centre, Macquarie Road, Springwood on the 4th Wednesday of each month at 8.00pm.

CENTRAL COAST AMSTRAD USERS CLUB
President: Lloyd Mitchell (043 88 2950)
Secretary: Ray Thompson (043 32 9095)
Venue: The Entrance Aquatic Club, Salt Water Reserve, Long Jetty every second Monday at 7.30 p.m. sharp.

COFFS HARBOUR AMSTRAD COMPUTER CLUB
President: Bruce Jones (066 52 8334)
Secretary: Don Donovan (066 52 6909)
Treasurer: Brian Claydon (066 49 4510)
Venue: Orara High School, Joyce Street from 7.00 on the first Friday of each month.

FAIRFIELD MICRO USER GROUP
Contact: Ekrem [after 6.30 pm] (02 609 6581)
Venue: Room 65, Canley Vale High School, Prospect Road, Canley Vale every third Wednesday from 7.00.

HAWKESBURY AMSTRAD USER GROUP
Contact: Terry Webb (045 76 5291)
Venue: Richmond Swimming Ctr, East Market St., Richmond every third Tuesday at 7.30 pm.

Please keep us notified of any changes to this listing. Amendments will be made in the next possible issue.

ILLAWARRA AMSTRAD USERS CLUB
President: Paul Simpson (042 27 1574)
Secretary: Ken Waegele (042 56 6105)
Publicity Off: Steve Parsons (042 96 3658)
Venue: AGA Gremania Club, Berkeley at 2.00 p.m. evey third Saturday.

LISMORE DISTRICT AMSTRAD COMPUTER CLUB
President: Max Muller (066 337 113)
Vice Pres: Nick Van Kempen (066 874 579)
Sec/Treas: Laurie Lewis (066 62 4542)
Venue: Goonellabah Public School, Ballina St. on the last Tuesday of each month from 6.30.
Mail: 20 Johnston Street, Casino, NSW 2470

S & W MILLER AMSTRAD USER'S CLUB
President: Wal Sellers (049 33 5459)
Secretary: Nikki Lee (049 33 5459)
Treasurer: Georgina Todd (049 66 2788)
Venue: Maitland Park Bowling Club, Maitland on the second Tuesday of each month at 7.30pm.

NEWCASTLE AMSTRAD USER GROUP
President: John Harwood
Treasurer: Erica Harwood
Secretary: Janet Bowen
Venue: Kotara Public School, Park Avenue, Kotara on the first Tuesday of each month. Contact the above for meeting times.
Mail: PO Box 18, Charlestown, NSW 2290

PCW AUSTRALIA GROUP
President: David Springett (02 660 4515)
Secretary: David Chamberlain (047 77 4396)
Venue: Burwood RSL Club, 96 Shaftsbury Road, Burwood every second Tuesday of the month at 7.30pm.
Mail: PO Box 97, Annandale, NSW 2038.

PORT MACQUARIE AMSTRAD USERS GROUP
Mail: Craig Tollis, Box 584, Pt. Macquarie, 2444.

SYDNEY AMSTRAD COMPUTER CLUB

President: Bob Knowles   (02 810 7373)
Secretary: Reed Walters   (02 560 9487)
Treasurer: Jim Chryss   (02 327 7872)
Venue: Newtown area on the 1st Saturday of every month for a normal club meeting and on the 3rd Saturday for the purposes of programming tutorials only. Both meetings commence at 2.00 p.m. For more details contact the Secretary or Treasurer between 6.00 p.m. and 9 p.m.

SYDNEY PC1512 USER GROUP

Contact: Geoff Craine   (02 76 6467) A/H
   (02 412 9213) B/H
Venue: To be arranged; meeting initially on the third Tuesday of each month at 7.00 pm.

QUEENSLAND

BRISBANE AMSTRAD COMPUTER CLUB

President: John O'Connor   (07 271 3350)
Vice Pres: John Digby   (07 351 2553)
Secretary: Bob Ashe   (07 355 5699)
Treasurer: Ivan Dowling   (07 269 8795)
Tech. Editor: Franz Hendrickx   (07 356 0633)
Venue: Main meetings in Room 15a of Junction ParkState School, Waldheim St., Annerley starting at 7.30p.m on the 1st Tues. of the month.. Another is held at Wynnum Central State H.Sch, Florence St., Wynnum Central on the 3rd Saturday of each month at 1.00p.m. The coordinator is Warren Kennedy (07 351 4232). A third is held at Newmarket State Sch., Banks St., Newmarket on the second Saturday of each month at 1.30p.m. The co-ordinator is Cherry Shrier (07 351 6179).
Mail: PO Box 167, Alderley, Qld. 4051

BUNDABERG AMSTRAD USER'S GROUP

President: Ray Babbidge   (071 72 1223)
Secretary: Clive Barrett   (071 71 3668)
Treasurer: Sheila Cole   (071 72 8884)
Venue: The third Tuesday of the month. For more details contact the above.
Mail: PO Box 865, Bundaberg, QLD 4670.

CABOOLTURE AMSTRAD USER GROUP

President: John D'Archambaud   (071 95 4860)
Secretary: Stephen Yench
Treasurer: Craig Deshon
Venue: Contact above number for more details.

CAPRICORN AMSTRAD USERS GROUP

President: Graeme Annabell   (079 27 4915)
Sec/Treas: Anthony Trost   (079 33 1951)
Venue: Waraburra State School, Johnson Road, Gracemere on the first Friday of each month at 7.00 pm.
Mail: 4 Sunrise Crescent, Gracemere, 4702

COMPUTER USER GROUPS OF AUSTRALIA

Pittsworth Branch
President: David Siebuhr
Contact: Ron Langton   (076 931 690)
Venue: Every first Tuesday of every month from 5 pm. at the St. Peter Lutheran Church Hall, Grand Street, Pittsworth.
Mail: CUGA, PO Box 166, Pittsworth, 4356

GOLD COAST AMSTRAD USER GROUP

President: Mark Abbott   (075 31 2114)
Treasurer: Pam Scott
Secretary: Mary Madaren
Venue: Benowa State High School, Mediteranean Drive, Benowa on the first Saturday of each month at 2.00 pm.
Mail: 17 Ewan Street, Southport, Qld 4215

HERVEY BAY - MARYBOROUGH AMSTRAD COMPUTER USER GROUP

President: Ian Jardine   (071 28 3688)
Vice-Pres: Gerhard Schulze
Sec/Treas: Les Patford   (071 28 9737)
Venue: The first Thursday of each month at 7.00 alternating between the Hervey Bay Senior College and Maryborough TAFE College.

Contact the above for more details.
Mail: Les Patford, PO Box 24, Torquay, Q 4657

IPSWICH AMSTRAD USER GROUP

Contact: Peter Wighton   (07 288 4571)
Venue: Every second Wednesday from 7.15 p.m. at Bremer High School, Blackstone Rd, Raceview

MACKAY AMSTRAD USER GROUP

Contact: Des Mulrealiey   (551 409)
   Ron Coates   (547 222)
Venue: Meet every second Sunday morning. Contact the above for location and time.

PENINSULA AMSTRAD CLUB (amalgamated with BACC)

President: Ivan Dowling   (07 269 8795)
Treasurer: Keith Johnston   (07 203 2339)
Secretary: Tracie Payne   (07 267 6645)
Venue: Kippa-Ring State School Library, Elizabeth Avenue every third Tuesday of the month at 7.30 pm.

SOUTHSIDE AMSTRAD USER GROUP (QLD)

President: Michael Toussaint   (07 200 5414)
Vice-Pres: Peter Incoll   (07 208 2332)
Secretary: Mick Howe   (07 209 1839)
Treasurer: Wayne Stephens   (07 287 2459)
Librarian: Carol Watts   (07 287 2882)
Venue: Loganlea State High School (in the Communications Room) every third Saturday of the month starting at 2.00 p.m. A Basic programming course is held fortnightly.
Mail: 10 Carramar St, Loganlea, 4204

TOOWOOMBA AMSTRAD USERS GROUP

President: Stephen Gale   (076 35 5001)
Vice-Pres: Priscilla Thompson   (076 35 5092)
Secretary: Adrian Dunsmore   (076 91 1561)
Treasurer: Edwin Gerlach   (076 33 1054)
Venue: Toowoomba Education Centre, Baker Street, Toowoomba on the 4th Monday of each month starting at 7.30 pm.

TOWNSVILLE AMSTRAD USER GROUP

President: Ian Wallace   (077 73 1798)
Vice-Pres: Doug Selmes   (077 79 6011 xt 252)
Treasurer: Chris Nisen   (077 79 6299)
Secretary: Alister Buckingham   (077 73 3955)
Venue: Science Block of the Kirwan High School in Thuringowa Drive on the first and third Tuesdays each month at 7.30pm.

THE WARWICK AMSTRAD USER GROUP

President: Mrs. D. Christensen
Secretary: John Wode   (076 61 5176)
Treasurer: Neville Christensen
Venue: Warwick Education Centre on the first Saturday of each month from 3.00 p.m.

WEIPA AMSTRAD USERS CLUB

President: Andrew Seaborn
Vice-Pres: Dave Wootton
Treasurer: Frances Casey
Secretary: Gary Chippendale   (070 69 7448)
Venue: Noola Court in Weipa. Contact above for more details.
Mail: 15 Noola Court, Weipa, QLD 4874.

WESTERN SUBURBS AMSTRAD USERS GROUP

President: Peter Wighton   (07 288 4571)
Secretary: Jimmy James   (07 376 1137)
Contact: Keith Jarrot   (07 376 3385)
Venue: The Jamboree Heights State Primary School, 35 Beanland Street, Jamboree Heights at 1.30 p.m. on the first Saturday in each month.
Mail: Jimmy James, 36 Penong Street, Westlake, Brisbane 4074.

TASMANIA

SOUTHERN TASMANIAN AMSTRAD USER CLUB

President: Frank Self   (002 49 5499)
Secretary: Peter Campbell
Treasurer: Cindy Campbell
Publ. Off: Danny Brittain   (002 47 7070)
Venue: Elizabeth Matriculation College on the first Wednesday of each month from 7.30 pm.

NORTHERN TASMANIA AMSTRAD COMPUTER CLUB

President: Russell Lockett   (003 44 8972)
Treasurer: Keith Chapple   (003 26 4338)
Secretary: Shane Crack   (003 97 3298)
Publicity: Michael Watts   (003 31 1944)
Librarian: Patrick Salter   (003 97 3379)
Junior Del: Bobby Lockett   (003 44 8972)
Venue: Launceston Community College (opposite Park Street) in Room 11 on the first Saturday of the month at 5.00 pm.

N.W. COAST AMSTRAD USER'S CLUB

President: Peter Gibson   (004 24 7586)
Treasurer: Robert Simpson
Secretary: Karen Stevenson
Venue: Hellyer College, Mooreville Rd, Burnie on the third Friday of each month at 6.30.
Mail: Secretary, 112 Payne St., Burnie 7320

NEW ZEALAND

THE AMSTRAD COMPUTER CLUB OF CANTERBURY

Contact: Christine Linfoot   897 413
   Ian Orchard   524 064
Venue: Four Avenues School, cnr. Madras Street and Edgeware Road, Christchurch 1 on the fourth Wednesday of each month.
Mail: Box 23.082 Bishopdale, Christchurch, NZ.

WELLINGTON AMSTRAD USER GROUP

Contact: Tony Tebbs   791 072 (evgs)
Venue: Cafeteria, NZ Fisheries Research Division, Greta Point, on the first Monday of each month from 7.30 pm.
Mail: PO Box 2575, Wellington, New Zealand.

User Group Contact List

Please note that the following names are listed as contacts for new user groups and should NOT be viewed as a problem solving service.

NSW

Name	Location	Phone
Nick Rogers	Bogan Gate	(068) 64 1170
Chris Craven	Canowindra	(063) 44 1150
Trevor Farrell	Coolah/Mudgee area	(063) 77 1374
David Higgins	Inverell	(067) 22 1867
Paul Wilson	Moruya	(044) 74 3160
Frank Humphreys	Mummulgum	(066) 64 7290
Reuben Carlsen	North Sydney	(02) 957 2505
Stephen Gribben	Singleton	(065) 72 2732
Ken Needs	St. Ives	(02) 449 5416
Chas Fletcher	Toongabbie	(02) 631 5037
Nick Bruin Snr.	Tweed Valley	(066) 79 3280

VIC

Name	Location	Phone
Brian Russell	Ballarat	(053) 31 2058
C. van de Winckel	Ballarat	(053) 313 983
Rod Anderson	Camperdown	(055) 93 2262
Paul Walker	Heathmont	(03) 729 8657
Terry Dovey	Horsham	(053) 82 3353
Andrew Portbury	Leongatha	(056) 62 3694
R. Kernebone	Miildura	(050) 23 3708
Angela Evans	Mt. Evelyn	(03) 736 1852
Keith McFadden	Numurkah	(058) 62 2069
Maureen Morgan	Warnambool	(055) 67 1140

QLD

Name	Location	Phone
Beryl Schramm	Boyne Island	(079) 73 8035
Steven Doyle	Caloundra	(071) 91 3147
Ric Allberry	The Gap	(07) 300 1675
Neville Eriksen	Gladstone	(079) 78 2418
Kylie Telford	Goondiwindi	(076) 76 1746
D.F. Read	Ingham	(077) 77 8576

SA

Name	Location	Phone
Lindsay Allen	Murray Bridge	(085) 32 2340
Michael Spurrier	Murray Bridge	(085) 32 6984
Mrs. S. Engler	Penola	(087) 36 6029

WA

Name	Location	Phone
Graeme Worth	Scarborough	(09) 341 5211
P.M. Nuyens	Waroona	(095) 33 1179

TAS

Name	Location	Phone
Conal McClure	Scottsdale	(003) 52 2514

NT

Name	Location	Phone
G.P. Heron	Tiwi	(089) 27 8814

This revised list is based on information collected over the last four months. If your name appears and it shouldn't, please let us know.
For new readers: if you want to start a group in your area just drop us a line with the relevant details and we will add you to the list.

Gossip from the UK

Legal battle looms

Epson has issued a writ against Amstrad. The reason is to prevent Amstrad using the letters LQ on their latest batch of printers.

Epson, who has been manufacturing printers with an LQ suffix for the past four years, objects to Amstrad's use of LQ because they claim dealers have come to associate the designation LQ with Epson printers.

According to Andy Mackay at Amstrad's PR agency, "Amstrad will defend its position strenuously. Epson are wrong trying to put an injunction on the letters LQ when several other companies use NLQ on their printers." Andy added that Amstrad will continue selling one printer as the LQ3500. It

seems, then, that Amstrad and Epson are to be locked in a legal battle.

The situation is rather ironic for Amstrad because they took legal action, for a similar reason, several months ago, when they attempted to prevent companies using AMS in their product names. In that instance, it was because the suffix stands for Alan Michael Sugar.

Both cases revolve on the issue of "passing-off" - naming a product so as to gain an unfair advantage through confusion with someone elses product and reputation. Amstrad claim that LQ is a generic term meaning letter quality, and is in common usage and therefore can't be a trademark (in the same way as "hoover").

Rumour becomes fact

As leaked in last month magazine, the new Amstrad Lap-top computer was released at the Comdex exhibition in the States and by the time you read this will have been launched in the UK. There are two models designated the PPC512 and PPC640. They feature an enhanced

AT keyboard and 25kg portability. The PPC640 includes a switchable 2400/1200/300 baud modem. Both models have a lift-up top containing an 80 column LCD screen.
(More details, hopefully, in next months mag - Ed).
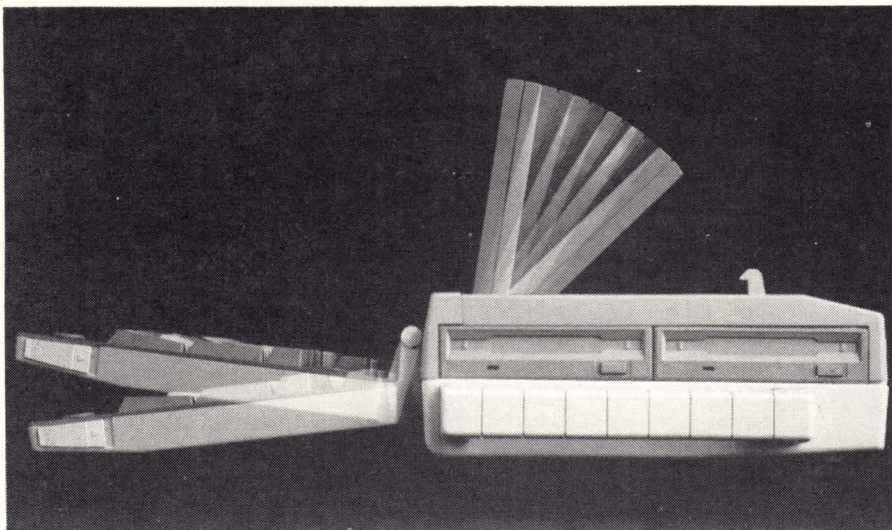
PC Games bonanza

Rainbird/Firebird have shown a sudden interest in the PC market with the announcement of at least five titles over the next month or so.

Just one comes from Firebird - that classic, and bound to be a huge seller, Elite which will be both Amstrad joystick and mouse driven. The others come from sister company Rainbird - The Pawn and Guild of Thieves - written by Magnetic Scrolls and Knight Orc, the three part adventure written by Level 9. Finally, Universal Military Simulator, a 3-D strategy war game which features five preprogrammed battles including Waterloo, Hastings, Gettysburg, Marston Moor and Arbela.

Loriciels has released Grand Prix 500cc, a motor cycle simulation race with a split-screen feature for two player games.

Electronic Arts on the other hand, lets you experience the thrill of driving several of the world's top sports cars in its game called Test Drive. They have also released a repackaged version of Starflight, one of the best PC games of all time. *(I wonder when we will see them over here - Ed).*

Portable PCW

Nabitchi Computing of Liverpool have just announced a portable version of the PCW8256. They have taken the insides of an 8256 and rehoused them in a custom-designed case with a new power supply and screen. The display is a 4" diagonal TV tube which can show the full 90 columns by 32 rows. Also included is the 8256's standard disc drive. It will, of course, run all PCW software unchanged, including LocoScript. The keyboard and printer are standard. The whole unit runs off rechargeable batteries which are claimed to last for up to 4 hours. continuous use. It can also be run off mains during which time the batteries are recharged.

Because the Nabitchi is a converted 8256, when it is purchased you get the unused full size display screen left over from the conversion. The cost of conversion is currently £200 + tax. *(We haven't heard that the same conversion will be available in Australia , but Alan Sugar is reported to be "looking at" plans for a portable PCW - Ed.).*

Sydney's PC Show rescheduled

Australian Exhibition Services Pty. Ltd. (AES) has decided to reschedule Australia's largest technology exhibition, PC88, Office Technology 88 and Communications 88, from 7-10th February to 20-23 March 1988. The triple event will still be held at Darling

THE 10th AUSTRALIAN
PERSONAL
COMPUTER
SHOW

DARLING HARBOUR SYDNEY

PC 88

The Australian International Office Technology Exhibition

OFFICE TECHNOLOGY 88
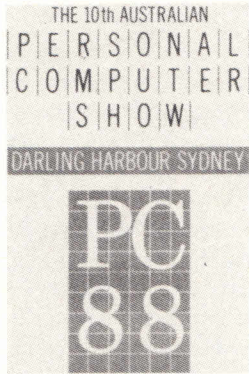
Harbour, in the same halls as in the previous tenancy.

AES Managing Director, Graeme Selby, said "Whilst we have at all times received and continue to receive assurances from the Sydney Convention and Exhibition Centre that the halls will be available for our February tenancy, there are nevertheless significant risks associated with being the first tenant, and doubts about the availability of all services. We have therefore decided to

eliminate the risk factor by accepting an excellent alternative tenancy which has now become available."

"March has always been our preferred tenancy for the PC Show, having staged the exhibition in March for the last five years, as we regard it as a more commercially effective time of the year than early February."

PC88 - 10th Australian Personal Computer Show - Office Technology 88 and Communications 88 will now be held from Sunday 20th March to Wednesday 23 March 1988 in Halls 3 and 4 at Darling Harbour.

THE AUSTRALIAN INTERNATIONAL ELECTRONIC COMMUNICATIONS AND INFORMATION TECHNOLOGY EXHIBITION

COMMUNICATIONS 88

PC Desk Top Publishing

Personal Computer Software (PCS) have released Imagemaster Version 2. The software is a PC system for image capture, forms design and page-makeup. It runs in conjunction with Wordcraft on an IBM XT, AT, PS/2 or 100% compatible with at least 512k RAM with a mouse installed. The Canon Laserbeam A2 or Series 2, the Hewlett Packard Laserjet Plus or Series 2 and the Kyocera (marketed in Australia by Imagineering) laser printers are all supported. The software also supports the following image scanners: HP Scanjet, Canon IX-8 or IX-12, Neotek-306A and AST Premium.

Version 2 of Imagemaster has an extensive range of new features including "On Line" help, an image library and tutorial and extended forms design capability. The system is driven by "pull down" menus and is extremely easy to install and use. It is priced at $888.

For more details contact PCS on (02) 923 2899.

Masterfile PC due soon in Australia

The highly successful "Masterfile" relational database package which has appeared in various forms for each of the Amstrad computers (464 through to PCW) will very shortly be available in Australia for the PC1512 and PC1640.

The PC version, like its predecessors, has been written entirely in machine code to gain the greatest possible speed. The relational database logic allows up to five files to be loaded at any one time; one for general access and update; the other 4 for read-only display.

It will have a file capacity of over 400k bytes in 512k machines and 582k in 640k machines. This gives, for example, enough space for 5800 or 7700 addresses respectively. There are many improvements over existing versions, even Masterfile 8000 for the PCW's which was hard to beat! The new

features include a 160 column display screen with automatic side-scroll when moving from field to field, a printed output of up to 160 columns and you can even print boxes in either PC characters or ASCII equivalent. Print formatted data can be diverted to disc, with or without printer controls, to feed a spooler or word-processor. For users with data privacy in mind, encryption is supplied as standard, with optional password protection on some functions. And there are many more improvements.

The Amstrad User has secured exclusive rights to market Masterfile PC in Australia. Full details will be announced in the March issue. The price is expected to be $199.00.

Dealer enquiries are welcome before the first shipment arrives.

Oops - no disc!

In our review of the book "Start Computing on the Amstrad CPC6128" written by Judith Thamm (see Page 57, Issue 35 - December 1987) we mistakenly advised that the price of the book

plus disc was "just $20". In fact, the book alone costs $20. The disc costs $12 if ordered with the book or $15 if ordered separately. Our apologies to Judith and anyone else who was inconvenienced by this error.

Get the picture

A new series on Graphics Programming for CPC machines

by Gary Koh

Have you ever wondered how to produce those fancy graphics displays that are to be found in all those commercial games? Or have you freaked out at the thought of trying to produce even a simple text display? Well, fear no more. This is the first of several articles (then again, the editor might have different ideas) on advanced aspects of graphics on the CPC range of computers. With any luck we will be covering Graphics routines, drawing shapes and lines, Sprites, Line Graphics, WIMP environments, Graphs and Masking.

So what are advanced graphics? Most of you out there would have an idea of what graphics are. Computer Graphics as it is, is very hard to define but here goes. Casually speaking a Graphic is a picture, design or diagram. This is allright if you limit graphics to a piece of paper but you start to run into problems with a computer. With a computer, not only can you produce graphics but you can also manipulate the screen. I call this screen manipulation. An example of screen manipulation is scrolling (and you thought that was graphics).

This is where we start running into problems. Defining the barrier between graphics and screen manipulation is quite a headache. Using the definition given earlier you could just as well call anything that involves screen manipulation, graphics.

So to clear this matter up once and for all we will be defining graphics as a picture or design on the screen and screen manipulation as altering or reorganising the layout of a screen or part of a screen. What we will be discussing in this and later parts will be mainly graphics and a bit of screen manipulation.

By now you must be pretty confused by all this. Don't

worry. Just because it is called 'Advanced' does not mean it is grossly complicated. Neither does it mean that you need to be an advanced hacker to be able to produce such graphics. During our stroll into advanced graphics I will show you how rewarding it can be to know how to draw and produce advanced graphics.

About 7 years ago microcomputers, did not have very powerful graphics capabilities. One of the reasons for this was that there was not enough memory to hold a large screen. Most microcomputers of that period had about 16 Kb of memory (some had as little as 1 kb!). In order to conserve memory, they used character-mapped screens as opposed to the bit-mapped screens that we are used to these days.

With character-mapped screens, instead of being able to draw lines to form pictures, you have to define a lot of characters and print them on the screen. Although character-mapped screens use only a small amount of memory it is very hard to produce graphics on them.

This is a far cry from an Amstrad computer. When the Amstrad CPC 464 was first released back in late 1984 it was unique in that it did not have a text mode (in other words, a character-mapped screen). Most computers have what are called screen modes. Each mode is different from each other and each one can display either text or graphics in a variety of resolutions.

To understand this better lets have a look at mode 1. This mode has a resolution of 320 by 200 pixels and has 4 colours. Normally if a computer had a mode like this the top four fifths or so of the screen would be used for graphics. The bottom part of the screen is used for text and for typing commands in direct mode. However not all computers are like this. A few can mix text and graphics on screen but they still have graphics and text modes.

Your Amstrad has 3 different modes and each one allows you to freely mix text or graphics on the screen. Every other computer that I know of cannot mix graphics and text with the exception of the BBC model A or B. Although the BBC has both graphics and text modes it can mix text with graphics in the graphics modes.

	SCREEN MODE	No. OF COLOURS	RESOLUTION	WIDTH IN CHARACTERS	SCREEN SUM
Figure One	0	16	160*200 Pixels	20	160*200(pixels)*4(n/b)=128 000
	1	4	320*200 Pixels	40	320*200(pixels)*2(n/b)=128 000
	2	2	640*200 Pixels	80	640*200(pixels)*1(n/b)=128 000

Figure 1 shows you the attributes of each mode. As you can see, the higher the resolution of a mode, the less colours that can be displayed. To explain this, we will use mode 1 as an example. If you know anything about binary you would know that it takes 2 bits to store the information of 4 colours for each pixel. The 'screen sum' column in figure 1 shows you the sum for working out how many bits in a screen. As you can see, the end result is 128,000 bits, which makes 16,000 bytes. A screen actually occupies 16,384 bytes. There is an excess of 384 bytes.

In order to be able to produce graphics you need to understand how the screen is mapped and how it works. To start with you can think of the graphics screen as a piece of graph paper 640 pixels wide and 400 pixels tall. The real screen actually is only 200 pixels high. The reason why it is stretched to 400 pixels is so the screen's aspect ratio is in an appropriate unity. The graphics 'graph paper' uses Cartesian co-ordinates.

In case Maths is not your forte let me explain. Rene Descartes, a French philosopher and mathematician (1596-1650) devised this fantastic way of finding the location of an object using a grid. To do this you simply count x units left and y units up. The pair of pronumerials (x,y) refer to a position on the Cartesian plane. And no, I don't know why they're called x,y co-ordinates.

Using these co-ordinates you can specify any grid location on the screen. In order to draw on the screen you use an invisible graphics cursor. To make the cursor fly around the screen at your will you use the Move command. The format for the Move command is Move x,y. You should know what the x and y are. The Draw x,y,c command draws a line from the current graphics cursor position to the co-ordinates specified. The c parameter tells the computer what pen colour you want the line to be.

The plot x,y,c command moves the graphics cursor to (x,y) and plots a point there in pen colour c. The command Test (x,y) will give you the colour of the pixel (x,y). This may not sound very useful but it allows you to detect a collision between two objects.

These commands each also have relatives (no pun intended). These relatives are identical to the four commands we have just discussed except that they have a R slapped on to the end and that they address relative co-ordinates. Now Move and Co. address what are called absolute co-ordinates. To put it simply they use just any co-ordinate on the screen. However MoveR and Co. use relative co-ordinates.

When I first tried drawing lines I had no trouble using absolute co-ordinates but relative co-ordinates had me stumped for a while. Just so you don't get into the same mess as me I give you Listing 1. This is a mini tutorial on absolute and relative co-ordinates. This program draws a grid of lines which are meant to be streets.

Now to go around the streets from point A to point B by car you would have to move around by going up a few kilometres then left a few kilometres and so on. This is how relative co-ordinates work. Now, if you could fly in a plane from point A to B you could just move directly between the two points. This is how absolute co-ordinates work. This is about the best explanation of relative and absolute co-ordinates I can give you.

Listing One

```
100 ' Listing 1
110 ' Relatives & Absolutes
120 ' by Gary Koh(14)
130 ' The Amstrad User Feb(1988)
140 '
150 CALL &BBFF:INK 0,2:INK 1,0:INK 2,26:MODE 1:BORDER 2
160 WINDOW #2,14,40,16,25:PEN #2,1
170 FOR a=1 TO 193 STEP 32:MOVE a,1:DRAW a,193,1:NEXT
180 FOR a=1 TO 193 STEP 32:MOVE 1,a:DRAW 193,a:NEXT
190 FOR a=1 TO 17:READ x,y:x=x*16:y=y*16:MOVE x,y+4:DRAWR
0,-10,0
200 MOVE x-4,y:DRAWR 10,0:NEXT
210 PLOT 800,800,3:TAG:MOVE 5,20:PRINT"A";:MOVE 176,189:PR
INT"B";:TAGOFF
220 LOCATE 5,2:PEN 1:PRINT"This grid represents some stree
ts":PRINT
230 PRINT TAB(4)"To walk from point A to B you would"
240 PRINT TAB(4)"have to:-":PEN 2
250 PLOT 1,1,2:MOVE 1,1:FOR a=5 TO 14:LOCATE 15,a:READ sct
,ltm:PRINT"Move ";
260 ON sct GOSUB 280,290,300,310:PRINT"kilometres":FOR del
ay=1 TO 900:NEXT:NEXT
270 GOTO 320
280 PRINT"up";ltm;:DRAWR 0,ltm*32:RETURN
290 PRINT"down";ltm;:DRAWR 0,-ltm*32:RETURN
300 PRINT"left";ltm;:DRAWR -ltm*32,0:RETURN
310 PRINT"right";ltm;:DRAWR ltm*32,0:RETURN
320 WINDOW SWAP 0,2
330 PRINT"However if you were in a"
340 PRINT"plane, to get from point A to B you could just c
ross"
350 PRINT"over the roads like this."
360 MOVE 1,1:DRAW 193,193,3
370 PEN 3:PRINT:PRINT"Press any key to continue":PEN 1
380 CALL &BB18:CLS
390 PRINT"   Going by car from point":PRINT"A To B shows h
ow relative"
400 PRINT"co-ordinates work. Going by";:PRINT"plane from p
oint A TO B"
410 PRINT"show how absolute co-ordi-":PRINT"nates work."
420 PRINT:PRINT"   I hope this helps you to";:PRINT"unders
tand relative and"
430 PRINT"absolute co-ordinates.";
440 WHILE INKEY$="":WEND
450 DATA 0,1,2,1,4,1,7,0,7,2,7,4,4,5,6,5,3,6,7,6,9,8,11,8,
6,9,8,9,5,10
460 DATA 5,12,10,11,4,3,1,2,3,2,1,2,4,3,2,1,4,1,1,2,4,1,1,
1
```

Listing 2 draws a picture of a house. It draws the house by reading in values from data statements. If you want to alter the size of the house alter the value in line 150. If you want to see the lines being drawn un-Rem line 250. If you want to see the house slowly growing un-Rem lines 190 and 200 and rem or delete line 180.

At this stage don't get confused with the 'text' graph paper. There are two graphics 'systems' used by BASIC. One of them, the Graphics commands produces what I call 'real' graphics. The other, the Text commands produces 'text' graphics.

You have had so many articles on redefining characters, using the Symbol command and using control characters that you have probably been driven crazy by them so I will not be covering these areas much. In case you don't have all the issues of The Amstrad User here is some information on the Text commands.

The text commands use a graph paper whose size depends on what mode you are in. The graph paper is always 25 units high but the width depends on the mode you are in. Figure 2 also gives information about this.

Each character is made up of a grid of 8 by 8 pixels. Now a character in mode 2 would occupy a 8 by 8 grid of pixels but if you want to print it the same size (by sight) in mode 0 it would occupy a grid of 2 by 8 mode 0 pixels. Obviously you can't fit it into a smaller area of pixels therefore the amount of characters you can print across in each mode is limited by its resolution.

There are 255 characters stored in the ROM of your Amstrad. You can redefine these characters by moving them into RAM by using the Symbol After command. You can divide them into three groups. Characters 0-31 are called Control Codes and have a special function. Chapter 9 in the User Guide gives a table of these Codes. Characters 32-127 will be

the ones you'll be most familiar with. These characters are called the ASCII code. All the characters printed on your keyboard are ASCII codes. Characters 128-255 are Predefined graphics. You can alter any of these characters but the control codes are a bit troublesome to print which leaves you with characters 32-255 to define.

By now you should know enough to be able to draw something. As an exercise try writing a program to draw a picture. A good technique is to draw it out on graph paper first. Take Listing 2 as an example. I firstly drew the picture on graph paper (I used a grid of 40 by 26) and then transferred it into the data statements. Have fun with that and see you next month.

GARY'S GRAPHICS GLOSSARY

Pixel - *Short for Picture Element. The smallest point you can plot on the screen.*
Bit - *One of the two digits (1 and 0) that make up binary code.*
Byte - *A group of 8 bits.*
Screen Resolution - *The number of pixels a computer can control individually.*
Aspect Ratio - *The ratio of the width of the screen to the height of the screen. Eg. if the width is 30cm. and the height is 20cm. the aspect ratio is 3:2.*

Listing Two

```
100 ' Listing 2
110 ' Scaled House
120 ' by Gary Koh(14)
130 ' The Amstrad User Feb(1988)
140 '
150 CALL &BBFF:INK 0,11:INK 1,0:PEN 1:BORDER 11:MODE 1
160 s=2:' Scale value. Altering this    value changes the
scale of drawing. If  changing this to a minus value add t
he command ORIGIN 500,350
170 ' Locate and draw house
180 GOSUB 230
190 'scl=0.1:s=0.2:WHILE s<14:CLS:scl=scl+0.08:s=s+scl
200 'GOSUB 230:FOR delay=1 TO 200:NEXT:WEND
210 WHILE INKEY$="":WEND:END
220 ' Draw house
230 RESTORE 280:FOR a=1 TO 38:READ c,hx,hy
240 IF c=0 THEN DRAW hx*s,hy*s,1 ELSE MOVE hx*s,hy*s
250 'FOR delay=1 TO 200:NEXT
260 NEXT:RETURN
270 ' Data for drawing house
280 ' Outline
290 DATA 1,20,26,0,40,13,0,40,2,0,15,0,0,15,11,0,20,26
300 DATA 0,0,18,0,0,7,0,15,0,1,0,18,0,15,11,0,40,13
310 ' Door
320 DATA 1,25,1,0,25,9,0,30,9,0,30,1,1,29,5,0,29,6
330 'windows
340 DATA 1,2,15,0,6,13,0,6,8,0,2,10,0,2,15
350 DATA 1,9,12,0,13,10,0,13,5,0,9,7,0,9,12
360 DATA 1,17,9,0,22,9,0,22,4,0,17,4,0,17,9
370 DATA 1,33,11,0,38,11,0,38,6,0,33,6,0,33,11
```

Cheat Mode

Send your game-busters to:
**The Editor
(Cheat Mode)
1/245 Springvale Road,
Glen Waverley, Vic
3150**

Killer Ring

A poke for the disc version of Reaktor's shoot-em-up. Infinite lives and varying degrees of vulnerability are the flavours of the day. Type in the listing and save it to a disc (it can be the Killer Ring disc). Before running remove the options you don't require. For example if you don't want invulnerability to bombs, leave out line 200.

```
1 ' Killer Ring - disc
2 ' The Amstrad User Feb 1988
10 t=&9000
20 READ a$
30 IF a$="-1" THEN 60
40 POKE t,VAL("&"+a$)
50 t=t+1:GOTO 20
60 MODE 1:CALL &9000
70 DATA 06,08,21,1a,90,11,00,c0
80 DATA cd,77,bc,21,40,00,cd,83
90 DATA bc,cd,7a,bc,cd,22,90,c3
100 DATA 40,00,52,49,4e,47,2e,42
110 DATA 49,4e
140 '
150 ' infinite lives
160 DATA af,32,93,17
170 ' aliens don't kill
180 DATA 3e,c9,32,af,12
190 ' bombs don't kill
200 DATA 3e,c9,32,91,12
210 ' remove bombs
220 DATA 3e,c9,32,2d,0c
230 ' total invincibility
240 DATA 3e,c9,32,08,12
250 '
260 ' leave the remaining DATA
280 '       in the listing
290 DATA c9,-1
```

Chronos

Infinite lives for Mastertronic's budget shoot-em-up. The man to thank is Daniel Underwood.
Use Method 1.

```
1 ' Chronos - tape
2 ' by Daniel Underwood
3 ' The Amstrad User Feb 1988
10 OPENOUT"":MEMORY
&2FF:CLOSEOUT
20 LOAD"ALLCHR": POKE &738B,&FF
30 CALL &8FDC
```

Cobra

Ocean's game of the film has been subjected to humiliation by Nicholas Pavis. Infinite lives are up for grabs. Enter the poke using Method 1, but take care if you use a 664 or 6128: several data lines must be altered. See below.

```
1 ' Cobra - 464 Tape
2 ' by Nicholas Pavis
3 ' The Amstrad User Feb 1988
10 DATA cd,67,bb,21,14,bf,36
20 DATA c9,cd,00,bf,21,ca,4e
30 DATA 36,00,23,36,00,23,36
40 DATA 00,23,36,00,c3,00,41
50 DATA 21,08,22,36,6b,23,36
60 DATA 99,21,40,20,11,40,00
70 DATA 01,19,02,ed,b0,21,40
80 DATA 00,e5,21,00,bb,e5,21
90 DATA 19,02,e5,21,07,b8,e5
100 DATA 21,bb,02,e5,f1,21,ea
110 DATA b1,11,d9,b1,f3,c9
120 MEMORY &2000
130 FOR x=&BE00 TO &BE4B: READ a$
140 POKE x,VAL("&"+a$): NEXT
150 LOAD"": CALL &3A6A
160 LOAD"!",&2040: CALL &BE1C
```

If you own either a 664 or 6128 machine, then type in the major listing - making the alterations shown below:

```
5 ' Alterations for 664 and 6128
90 DATA 19,02,e5,21,1f,b1,e5
100 DATA 21,bb,02,e5,f1,21,e5
110 DATA b7,11,d4,b7,f3,c9
```

Auf Wiedersehen, Monty

Be careful when you type in this Method 1 poke. Leave out all REM statements, or it won't work.
Nick Pavis has been busy on Gremlin's game. You can choose to remove crushers, have infinite lives, have the air tickets in your possession or be a super-mole.
Some of the data has REMs following to indicate what action the data performs. If you don't want infinite lives then leave out line 130, for instance.

```
1 ' Auf Wiedersehen, Monty -
2 ' by Nicholas Pavis
3 ' The Amstrad User Feb 1988
10 MODE 1: a=&340
20 READ a$: IF a$="+" THEN 50
30 POKE a,VAL("&"+a$)
40 a=a+1: GOTO 20
50 CALL &340
60 DATA 21,00,00,11,00,c0
70 DATA 06,00,cd,77,bc,21
80 DATA 00,01,cd,83,bc,cd
90 DATA 7a,bc,21,62,03,22
100 DATA 8f,01,3e,c3,32,8e
110 DATA 01,c3,00,01
120 DATA 21,63,fe,36,c9,' remove
crushers
130 DATA 21,b4,ad,36,c3 ' infi-
nite lives
140 DATA 21,9e,a5,36,00 ' air
tickets
150 DATA 21,18,85,36,c9 ' super
mole
160 DATA c3,00,80,+
```

Craig Winfield has pointed out that the command POKE 800,201 will give you infinite lives in the Monty game. Just type it in before running the game - tape only.

Nemesis

Neil McCauley and Antony Porter have bumped heads and produced a poke for Konamis' shoot-em-up.
Enter it using Method 1 for double laser and missile fire. You will also have a shield, which should help you somewhat on your journey.

```
1 ' Nemesis - tape
2 ' by Neil and Antony
3 ' The Amstrad User Feb 1988
10 ENT -1,1,9,1
20 ENT -3,2,-6,1,2,6,1
30 ENT -5,1,-,2,1
40 ENV 1,2,-1,2
50 ENV 3,2,3,1,3,2,1
60 ENV 5,2,-1,1
70 ENV 7,10,-1,5,30,0,1,10,1,10
80 ENT -7,2,-1,1,2,1,1,5,0,2
90 ENV 8,2,2,2,5,-1,2,1,-7,5
100 ENT -8,5,0,1,2,-1,1,2,1,1
110 MODE 1: BORDER 0
120 FOR t=0 TO 15: READ a: INK
t,a: NEXT
130 DATA 0,6,11,23,6,7,8,9,11
140 DATA 12,13,15,16,18,24,26
150 LOAD"!nemesis", &C000
160 OPENOUT"y": MEMORY 999
170 LOAD"NEMCODE",1000
180 POKE &9682,212:POKE &96AA,&20
190 FOR t=550 TO 980: POKE t,0:
```

```
NEXT
200 CALL 38341
```

Army Moves

Here's a welcome poke from Kevin McCaughey. It gives you various methods of cheating in both parts of Imagine's shoot-em-up. Use Method 1 and blast away to your heart's content. The first listing is for Army Moves part one - you can have immunity from bullets or enemy craft (or both) and infinite lives.

```
1 ' Army Moves (part 1) — tape
2 ' by Kevin McCaughey
3 ' The Amstrad User Feb 1988
10 N=0: CHK=0: L=70: MODE 1
20 FOR I=&A000 TO &A088
30 READ A$: J=VAL("&"+A$): POKE
I,J
40 N=N+1: CHK=CHK+J
50 IF N=8 THEN N=0: READ AMOUNT:
IF CHK<>AMOUNT THEN PRINT"ERROR
IN DATA LINE";L: END ELSE IF N=0
THEN CHK=0: L=L+10
60 NEXT I
70 DATA AF,CD,0E,BC,21,AF,39,
11,&360
80 DATA 00,00,3E,16,CD,A1,BC,CD,
&34B
90 DATA 6A,3A,21,40,A1,11,00,
00,&1B7
100 DATA 3E,16,CD,A1,BC,21,40,
A1,&380
110 DATA 01,26,02,3E,8D,CD,78,
A0,&2D9
120 DATA 21,86,A1,01,E0,01,3E,
3D,&2A5
130 DATA AE,CD,78,A0,21,40,A1,
11,&3A6
140 DATA 00,BB,01,26,02,F3,ED,
B0,&374
150 DATA DD,21,00,BF,11,4F,00,
CD,&2EA
160 DATA 67,BB,DD,21,00,C0,11,
00,&2F1
170 DATA 40,CD,67,BB,DD,21,1E,
05,&350
180 DATA 11,86,8D,CD,67,BB,AF,
06,&3C8
190 DATA F6,ED,79,21,3B,0B,77,
21,&35B
200 DATA 8B,12,77,21,4C,1C,3E,
C3,&29E
210 DATA 77,CD,37,BD,FB,C3,90,
92,&518
220 DATA F5,AE,77,F1,23,0B,C6,
0A,&409
230 DATA F6,80,5F,78,B1,7B,20,
F0,&489
240 DATA C9
250 INPUT "INFINITE LIVES(YN) :"
,Y$:IF UPPER$(LEFT$(Y$,1))="N"
THEN POKE &A066,0
260 INPUT "HARMLESS BULLETS (Y/N)
: ",Y$:IF UPPER$(LEFT$(Y$,1))="N
" THEN POKE &A06A,0
270 INPUT "IMMUNITY FROM ENEMY
CONTRAPTIONS:",Y$:IF UPPER$(LEFT
$(Y$,1))="N" THEN POKE &A070,0
280 CALL &A000
```

The second listing, for the second part of Army Moves gives infinite men.

```
1 ' Army Moves (part II) — tape
2 ' by Kevin McCaughey
3 ' The Amstrad User Feb 1988
10 N=0:CHK=0:L=70:MODE1
20 FOR I=&A000 TO &A085
30 READ A$:J=VAL("&"+A$):POKE I,J
40 N=N+1:CHK=CHK+J
50 IF N=8 THEN N=0:READ AMOUNT: IF
CHK<>AMOUNT THEN PRINT"ERROR IN
DATA LINE";L: END ELSE IF N=0 THEN
CHK=0:L=L+10
60 NEXT I
70 DATA AD,CD,0E,BC,21,AF,39,
AA,&360
80 DATA 00,00,3E,16,CD,A1,BC,
CD,&34B
90 DATA 6A,3A,21,40,A1,11,00,
00,&1B7
100 DATA 3E,16,CD,A1,BC,21,40,
A1,&380
110 DATA 01,26,02,3E,8D,CD,75,
A0,&2D6
120 DATA 21,86,A1,01,E0,01,3E,
3D,&2A5
130 DATA AE,CD,75,A0,21,40,A1,
11,&3A3
140 DATA 00,BB,01,26,02,F3,ED,
B0,&374
150 DATA DD,21,00,BF,11,4F,00,
CD,&2EA
160 DATA 67,BB,DD,21,00,C0,11,
00,&2F1
170 DATA 40,CD,67,BB,DD,21,1E,
05,&350
180 DATA 11,58,8E,CD,67,BB,01,
00,&2E7
190 DATA F6,ED,49,3E,C9,21,D9,
92,&4BF
200 DATA 77,AF,21,69,08,77,CD,
37,&333
210 DATA BD,FB,C3,58,93,F5,AE,
77,&580
```

```
220 DATA F1,23,0B,C6,0A,F6,80,
5F,&3C4
230 DATA 78,B1,7B,20,F0,C9
240 INPUT "INFINITE MEN(Y/N):",Y$
:IF UPPER$(LEFT$(Y$,1))="N" THEN
POKE &A06D,0
250 CALL &A000
```

Aliens

P.R. Tonks from Walsall read a previous batch of tips for the Electric Dreams game and thinks his method is much simpler.

Move Ripley quickly, and without firing a shot, to room 247, just before the Queen's chamber. Then repeat for one other character. Now blow the door to the chamber and kill everything that moves. Try to kill an alien in front of a wall of growth. This allows you to clear the wall while keeping the dead alien in view, and no other aliens will set off your tracker.

You'll need to use both characters to clear all the aliens and growth - just switch when one runs low on ammo. Once you've got the first company message just switch control to one of the four characters at the start and you'll get the second message, completing the game.

Into the Eagle's Nest

Glenn Pinnock has run a gauntlet of guards to tell us that if you type DAS CHT on the hi-score table of the Pandora game it gives you infinite health. Pity you can still get trapped by running out of ammo.

Short Circuit

Peter Featherstone has sent in a couple of pokes for Ocean's search-n-shoot game. The first poke is for part one of the game - which unlocks doors and prevents certain security checks.

The second poke, for section two, gives No 5 invulnerability to everything but water and mudpits. Both pokes use Method 1.

```
1 ' Short Circuit (pt I) — tape
2 ' by Peter Featherstone
3 ' The Amstrad User Feb 1988
10 MODE 1: MEMORY 14766: LOAD ""
20 FOR f=&BE00 TO &BE4B: READ a$
30 POKE f,VAL("&"+a$): NEXT
40 CALL &BE00
100 DATA 3e,c3,32,e2,39,21,14,be
110 DATA 22,e3,39,21,ff,ab,11,40
120 DATA 00,c3,af,39,3e,a8,32,00
```

```
130 DATA 02,21,45,e5,22,01,02,f3
140 DATA F1,c9,dd,21,00,bf,11,41
150 DATA 00,cd,4e,bc,3e,c3,32,1b
160 DATA bf,21,3a,be,22,1c,bf,c3
170 DATA 00,bf,3e,18,32,7e,0d,32
180 DATA 63,0c,21,18,fc,22,9e,0d
190 DATA c3,0a,02,00

1 ` Short Circuit (pt II) — tape
2 ` by Peter Featherstone
3 ` The Amstrad User Feb 1988
10 MODE 1: MEMORY 14766: LOAD ""
20 FOR f=&BE00 TO &BE63: READ a&
30 POKE f,VAL("&"+a$): NEXT
40 CALL &BE00
100 DATA 3e,c3,32,e2,39,21,14,be
110 DATA 22,e3,39,21,ff,ab,11,40
120 DATA 00,c3,af,39,3e,a8,32,00
130 DATA 02,21,45,e5,22,01,02,f3
140 DATA f1,c9,dd,21,00,bf,11,4f
150 DATA 00,cd,4e,bc,3e,c3,32,1b
160 DATA bf,21,3a,be,22,1c,bf,c3
170 DATA 00,bf,dd,21,33,0b,11,cd
180 DATA 60,cd,c9,03,af,32,7c,0c
190 DATA 32,46,20,3e,18,32,06,1f
200 DATA 32,dd,1e,32,98,1e,32,fc
210 DATA 1b,3e,c9,32,07,1e,c3,0a
220 DATA 02,00,00,00
```

The Curse of Sherwood

Infinite lives and permanent use fo the map are provided by Stephen Stokes for Mastertronic's game. Enter using Method 1

```
1 ` Curse of Sherwood — tape
2 ` by Stephen Stokes
3 ` The Amstrad User Feb 1988
10 a=&BF00
20 READ v$
30 IF v$="*" THEN CALL &BF00
40 POKE a,VAL("&"+v$)
50 a=a+1:GOTO 20
60 DATA 06,00,11,00,c0,cd,77,bc
70 DATA 21,c0,0d,cd,83,bc,cd,7a
80 DATA bc,21,7b,a6,11,df,ab,01
90 DATA bc,98,ed,b8
100 DATA 3e,00,32,ac,66 `lives
110 DATA 3e,00,32,6f,75 `map
120 DATA c3,0a,aa,*
```

Scooby Doo

Mad Monk (?) fresh from the asylum has sent in a weird batch of pokes for Elite's cartoon-like game. Remove the pokes you don't want. Disk only.

```
1 ` Scooby Doo — disk
2 ` by the Mad Monk
3 ` The Amstrad User Feb 1988
```

```
10 OPENOUT"y": MEMORY &16C9
20 :USER,1: LOAD"SCHOOB1",&16CA
30 LOAD"SCHOOB2",&68E8
40 POKE &7C29,&C9 `remove monks
50 POKE &7860,&C9 ` can't fall
60 POKE &7808,&C9 ` wide ladders
70 POKE &77B3,&C9 ` spirit plane?
80 POKE A7838,&C9 ` go thru walls
90 POKE &77F6,&c9 ` Scooby skates
100 CALL &66E8
```

Hydrofool

It's that Peter Featherstone chappie once more. This time he has delivered a prevent-Sweevo-from-rusting poke for FTL's game. Use Method 1.

```
1 ` Hydrofool — tape
2 ` by Peter Featherstone
3 ` The Amstrad User Feb 1988
10 MODE 1: OPENOUT"y":MEMORY &7CF
20 LOAD"": POKE &865,&C3
30 POKE &866,0: POKE &867,&BE
40 FOR f=&BE00 TO &BE38: READ a$
50 POKE f,VAL("&"+a$): NEXT
55 CALL &7d0
60 DATA 3e,c3,32,75,75,21,23,be
70 DATA 11,1c,83,01,0c,00,ed,b0
80 DATA 21,2f,be,11,47,96,01,0d
90 DATA 00,ed,b0,0e,ff,21,34,71
100 DATA cd,16,bd,20,41,4c,41,4e
110 DATA 20,53,55,47,41,52,20,28
120 DATA 30,35,33,32,20,36,31,33
130 DATA 30,35,34,29
```

Amaurote

Mastertronic's graphical superb MAD game has been hacked by Peter Feather-stone. Infinite lives and bombs have been supplied. You can fire a bomb even if there is one already on the screen - the first one will disappear, mind. Method 1.

```
1 ` Amaurote
2 ` by Peter Featherstone
3 ` The Amstrad User Feb 1988
10 MODE 1
20 FOR f=&BE00 TO &BE23
30 READ a$
40 POKE f,VAL("&"+a$)
50 NEXT
60 CALL &BE00
100 DATA 21,00,40,11,00,31
110 DATA 3e,16,cd,a1,bc,30
120 DATA f3,21,16,be,22,24
130 DATA 40,c3,00,40,af,32
140 DATA 44,9d,32,a0,8d,32
150 DATA 97,8d,c3,d4,26,00
```

Grand Prix Simulator

Several people have contacted us with the cheat for the code Masters racing game. All you have to do is type the word TRACK followed by the letter of the track you want to start on. A to N. Then while holding down the letter for the track, press 1 or 2 to start the game.

Chronos

Several people have also contacted us with a tip for the Mastertonic shoot-em-up. When you've got a high score type Jing It Baby into the table and a new option will appear on the menu. This gives you a megalaser.

Poke methods for tape

Here is how to input the majority of Cheat Mode tape pokes. The instructions for each poke tell you which of the two different methods to use. If you have a 664 or 6128, first type l tape.

Method 1

Rewind the game tape to the beginning. Now type in the poke listing. Then type RUN and press the Enter key. (Don't use the key marked CTRL or Control; that would stop the poke from working.) Press the Play key on the cassette deck, then any key on the main keyboard - the spacebar will do nicely. The tape should now start to play through in the normal way.

Method 2

For this method you have to skip the first bit of the game program. To do that, start by rewinding the game tape to the beginning. Now type in the listing. Then type CAT and press Enter. Start the tape by pressing Play and then any key. Then watch the screen.

Soon you'll get the message FOUND SOMETHING BLOCK 1. It doesn't matter what the something actually is; this will vary from one game to another. If the Cheat Mode instructions just tell you to skip the first block, you should stop the tape here.

If the instructions tell you to skip several things, stop the tape when the found message comes up for the last thing you're trying to skip.

Once you've stopped the tape, press Escape, type RUN and press Enter. Now press Play on the tapedeck and any key on the keyboard to start the tape running.

Custom Formats

Throw away your CP/M discs, use this Basic Formatter from Petr Lukes

I wrote over a disc directory when I first used CP/M FORTH, which uses the whole disc as its memory. Recently I did it again while testing routines for a direct-access file, which does not take any notice of the AMSDOS file structure, so I decided to use a non-standard format.

That led to this utility which optionally formats a disc so that it can be read and written to only by the program using the non-standard sector numbers. The utility can, of course, produce the standard AMSDOS formats, and it could be modified to format more than the normal forty tracks (at your own risk), and possibly adapted to format double-sided discs. In any case, I find it more convenient to format without having to load CP/M.

The disc is formatted by writing a header for each sector and filling the sector with filler bytes. As each track is formatted, its individual sectors are read back and checked to see that they contain the required number of filler bytes. Any faulty sectors would be reported, but this is unlikely with good quality discs.

The AMSDOS ROM contains the CP/M routines to access any sector, to select one of the standard formats, and to format any track. The complicated part is how to access them from BASIC. Their names are numbers from 1 to 9 and since their addresses are found by calling KL_FIND_COMMAND, the name must have the high bit set (so READ_SECTOR is named 4+128). Their addresses are fixed in any one ROM, but may differ in other versions of AMSDOS ROM, so the program finds them each time it is run. The routines are detailed in the FIRMWARE Manual.

Since they are ROM routines, the correct ROM must be selected before they are called. The easiest way to call them is via the special routine at RST 3, which requires an in-line address pointing to a three-byte block which contains the address of the routine to be called and the number of the ROM containing it. I have used the four bytes of the faradd integer array: the zero-th element (two bytes) holds the routine address, the first byte of faradd(1) has the ROM number while the second byte is zero. The address in faradd(0) can be changed to correspond with the routine to be accessed. The in-line address is then the address of the first element, ie. @faradd(0). This address is passed to the high level routine which sets up the required registers and finally calls the relevant ROM routine. In this way the msecio routine can be used both to read and write a particular sector, depending on the address in faradd(0). (The write routine is not used by this program.)

It all sounds very involved, and it is, but no more than doing it all in reserved memory. Using arrays (they MUST be integers, as MUST be most of the simple variables used) for storage of

the code avoids worrying about using memory which may be used by other routines. The ability to pass values into the array elements makes it much easier to write relocatable routines. However, it should be remembered that arrays will be shifted in memory when a new (not previously defined) simple variable is created, so addresses of any arrays should be passed just before a routine is called.

Even if you do not need a formatter, you may find use for some of the techniques. The use of integer arrays for storage of code is, of course, not new; I first saw it on a Tandy Model 1, nearly ten years ago.

```
10 zn$="FORMATS : Disc formatting utility for AMSTRAD CPCs
. LKS 871011"+CHR$(10)+CHR$(13)+"P. Lukes, 26 Noll St., TO
OWOOMBA, 4350."
20 INK 0,26:INK 1,0:PAPER 0:PEN 1:MODE 2:PRINT zn$
30 ' Initialize
40 DEFINT a-t:DEFSTR u-z:dr=&FF:er=&FF:er1=&FF:secn=&FF:tr
acn=&FF'create variables
50 buf=&C000'display memory as buffer
60 fsel=3+128:secr=4+128:secw=5+128:form=6+128'Command nam
es: select format, read/write sector, format. Locations ar
e used to store addresses after they are found by mfcom
70 DIM msecio(12),mfsel(6),mver(12),faradd(1),dpb(1),forms
(19)'Arrays for machine language, 'far address' for RST 3,
 disc parameter block address, format
80 dpb(0)=UNT(PEEK(&BE42)+PEEK(&BE43)*256):dpb(1)=dpb(0)+6
4'Address of Disc Parameter Block for drive A and B. DDI-1
 FIRMWARE gives the address as being at &be40
90 ' mfcom : find command, save address and ROM select. By
tes which will be replaced by addresses or values are show
n as FF. Padding bytes (00) are used to align addresses to
 array elements. Enter all data as two-digit hex numbers.
100 x="ld hl,address of command {0}{1}":DATA 00,21, FF,FF
110 x="push hl    call kl_find_command":DATA e5,cd, d4,bc
120 x="pop de  ret nc=unsuccessful":DATA d1,d0
130 x="ex de,hl  ld (hl),e  inc hl  ld (hl),d : save addre
ss":DATA eb,73, 23,72
140 x="ld a,c  ld (faradd(1)),a : save ROM select {7}{8}":
DATA 79,32, FF,FF
150 x="ret":DATA c9,00
160 FOR a=0 TO 9:READ x,y:mfcom(a)=VAL("&"+y+x):NEXT a:mfc
om(8)=@faradd(1)
170 faradd(1)=&FF:mfcom(1)=@fsel:CALL@mfcom(0):IF faradd(1
```

```
)<>7 THEN STOP'find Select_format (my 464 address=&c039)         540 dr=ASC(dr$)-65:mfsel(3)=dr:mform(1)=dr
180 faradd(1)=&FF:mfcom(1)=@secr:CALL@mfcom(0):IF faradd(1       550 PRINT"Drive "dr$" selected"
)<>7 THEN STOP'find Read_sector (&c03c) AMSDOS ROM is no 7       560 ' Format disc
.                                                               570 PRINT"FORMAT disc in drive "dr$:PRINT"Will overwrite a
190 faradd(1)=&FF:mfcom(1)=@secw:CALL@mfcom(0):IF faradd(1       ny data on disc!"
)<>7 THEN STOP'find Write_sector (&c03f) Not used for form      580 PRINT"VENDOR, DATA, and IBM are standard AMSDOS format
atting                                                          s.":PRINT"VENDOR is SYSTEM without the boot tracks."
200 faradd(1)=&FF:mfcom(1)=@form:CALL@mfcom(0):IF faradd(1       590 PRINT"FILE is a special format which AMSDOS cannot nor
)<>7 THEN STOP'find Format_track (&c042)                        mally read.":PRINT"It is DATA format with different sector
210 ERASE mfcom'no longer needed                                 numbers and filler byte."
220 ' msecio : read/write a sector                             600 INPUT"Select format (Vendor/Data/Ibm/File) : ";x
230 x="ld a,(drive) : 0=drive A, 1=drive B {0}{1}":DATA 00      610 sectr=9:skew=5:secp=2:IF x="V"THEN sec1=&41 ELSE IF x=
,3a, FF,FF                                                      "D"OR x="F"THEN sec1=&C1 ELSE IF x="I"THEN sectr=8:sec1=&1
240 x="ld e,a  ld hl,(buf) {2}{3}":DATA 5f,2a, FF,FF            :skew=1 ELSE PRINT CHR$(7);:GOTO 600
250 x="ld a,(secn) : sector {4}{5}":DATA 00,3a, FF,FF           620 faradd(0)=fsel:mfsel(1)=sec1:mfsel(3)=dr:mfsel(5)=@far
260 x="ld c,a  ld a,(tracn) : track {6}{7}":DATA 4f,3a, FF      add(0):CALL@mfsel(0)'Select_format, one of the standard AM
,FF                                                             SDOS formats
270 x="ld d,a  RST 3  inline addr of 'far address' {8}{9}"      630 IF x="F"THEN sec1=&F1:POKE dpb(dr)+15,sec1:POKE dpb(dr
:DATA 57,df, FF,FF                                              )+19,&A3'set FILE format : first sector no.=&f1,filler byt
280 x="ld (err),a : report error {10}{11}":DATA 00,32, FF,      e=pound sign
FF                                                              640 mver(5)=PEEK(dpb(dr)+19)'filler byte for verification
290 x="ret {12}":DATA c9,00                                     650 'Table for Format_track, stored in forms(): for each s
300 FOR a=0 TO 12:READ x,y:msecio(a)=VAL("&"+y+x):NEXT a:m      ector: track no., head no., sector no., sector size as 2^(
secio(1)=@dr:msecio(3)=@buf:msecio(5)=@secn:msecio(7)=@tra      secp+7)=2^9=512 bytes
cn:msecio(11)=@er                                               660 IF skew=5 THEN FOR a=0 TO sectr*2 STEP 4:forms(a+1)=se
310 ' mfsel : select format                                    cp*256+sec1+a\4:forms(a+3)=forms(a+1)+skew:NEXT a'sector n
320 x="ld a,first-sector-number {0}{1}":DATA 00,3e, FF,FF       umber and size
330 x="ld e,drive {2}{3}":DATA 00,1e, FF,FF                     670 'Sector interleave for other than IBM: 1,6,2,7,3,8,4,9
340 x="RST 3  address  ret {4}{5}{6}":DATA 00,df, FF,FF,        ,5
c9,00                                                           680 IF skew=1 THEN FOR a=0 TO sectr*2 STEP 2:forms(a+1)=se
350 FOR a=0 TO 6:READ x,y:mfsel(a)=VAL("&"+y+x):NEXT a          cp*256+sec1+a\2:NEXT a'No sector interleave for IBM
360 ' mform : format a track                                   690 PRINT"Display is used as buffer"
370 x="ld e,drive {0}{1}":DATA 00,1e, FF,FF                     700 PRINT"Press ENTER/RETURN to format disc in drive "dr$"
380 x="ld hl,addr of table {2}{3}":DATA 00,21, FF,FF             as "x;:INPUT" ",x
390 x="ld a,(tracn) {4}{5}":DATA 00,3a, FF,FF                   710 FOR tracn=39 TO 0 STEP-1:PRINT:PRINT"Formatting track
400 x="ld d,a  RST 3   address {6}{7}":DATA 57,df, FF,FF        &"HEX$(tracn,2);
410 x="ld (err),a : report error  ret {8}{9}{10}":DATA 00,      720  FOR secn=0 TO sectr*2 STEP 2:forms(secn)=tracn:NEXT s
32, FF,FF, c9,00                                               ecn'track, head(=0)
420 FOR a=0 TO 10:READ x,y:mform(a)=VAL("&"+y+x):NEXT a:mf      730  er=&FF:faradd(0)=form:mform(3)=@forms(0):mform(7)=@fa
orm(5)=@tracn:mform(9)=@er                                      radd(0)
430 ' mver : verify a sector after formatting                  740  CALL@mform(0):PRINT" error: "BIN$(er,8)'Format_track
440 x="ld hl,buf {0}{1}":DATA 00,21, FF,FF                     750  faradd(0)=secr
450 x="ld bc,512 = sector size {2}{3}":DATA 00,01, 00,02       760  FOR secn=sec1 TO sec1+sectr-1'read sectors
460 x="L1: ld a, filler byte {4}{5}":DATA 00,3e, FF,FF         770   er=&FF:msecio(9)=@faradd(0):CALL@msecio(0):er1=er
470 x="cpi = compare,inc hl,dec bc {6}":DATA ed,a1             780   er=&FF:CALL@mver(0)
480 x="jr nz,L2 = not equal {7}":DATA 20,04                    790   PRINT"Track &"LOWER$(HEX$(tracn,2))" Sector &"LOWER$
490 x="ld a,b  or c  jr nz,L1 {8}{9}":DATA 78,b1, 20,f5        (HEX$(secn,2));
500 x="L2: ld (err),a. ret {10}{11}{12}":DATA 00,32, FF,FF     800   PRINT" r/w error: "BIN$(er1,8)
, c9,00                                                        810   IF er=0 THEN PRINT"Verified"ELSE INPUT"Faulty ",x
510 FOR a=0 TO 12:READ x,y:mver(a)=VAL("&"+y+x):NEXT a:mve      820 NEXT secn:NEXT tracn
r(1)=buf:mver(11)=@er                                          830 INPUT"Format another (n/y) ";x:IF UPPER$(x)="Y"THEN 52
520 ' Select drive                                             0
530 dr$="":WHILE dr$<"A"OR dr$>"B":INPUT"Select drive (A o
r B) : ",dr$:WEND
```

Firing-up CP/M - 5

A closer look at some of the utility programs supplied with your CP/M system disc

More PIP than meets the eye?

After reading through previous articles on firing-up CP/M, you should be fairly happy with PIP and the various things it can do. One important point that has not been mentioned is the possibility of having extra parameters following the PIP command. Well, as promised last month, here is a brief rundown on the more important of these parameters and their functions.

You'll remember that PIP uses the star as a 'wildcard' - so *.com means any filename with extension '.com' and *.* means any filename, any extension. To copy all the files from the disc in drive A to the disc in drive B you would type

```
PIP B:=A:*.*
```

which is very handy. However, using the extra or optional parameters available, even more control is available to you. For example, type [V] directly following, like this

```
PIP B:=A:*.*[V]
```

and all the copied files will be verified - the system will check that they have been copied correctly.

Other parameters include:

[C] Confirm (in CP/M version 3.1 only). This will display a message asking you for confirmation before copying each file - especially useful where you have given ambiguous filenames (for example *.*). You will be given the choice to copy or not to copy.

[E] Echo all transfers to the screen. Messages such as 'Writing to drive B:' appear to let you know what is happening.

[L] Lower-case only. This will convert all capital letters to lower-case.

[U] Upper-case. This has the opposite effect to [L], in that it converts all lower-case characters to upper-case - ie. capitals. (The terms come from the days when printers set type by hand and sorted individual metal or wooden letters into two cases, an upper and a lower.)

[H] Hexadecimal characters are assumed. All characters in the file are checked to be hexadecimal; an error message results if not.

[Pn] Page length n. Formfeeds are inserted (a new page is started) after every n lines of text. The default is 60 - useful if copying to a printer.

[Qstring^Z] Quit. A file will be copied until the specified string of characters is encountered, and then closed with an end-of-file marker. For example [Qarea^Z] will copy everything up to, but not including, the word 'area'. (Be sure to type ^Z, which is CP/M's end-of-file marker, as Control-Z - not as the two separate characters arrow and Z.)

[R] Read system file(s). Sys files can be copied easily with this option. These files are hidden from the directory - see the articles on STAT.

[Sstring^Z] Start. A file will be copied from the first occurrence of the string, until the end of the file. Thus, [Sknow^Z] will start copying the file from 'know'. The string will be included in the copy unlike its counterpart [Qstring^Z].

[V] Verify. As mentioned previously, this verifies the data between source and destination files, checking that a perfect copy was made.

[X] Copy. Non-Ascii files (for example, Basic and binary files) can be copied using this option.

Watch the syntax of these extra parameters. If using more than one of them, the correct form is:

```
PIP LST:=DEMO.TXT[UP 57Q
hello^Z]
```

This command will send to the printer the file DEMO.TXT in upper case, inserting a formfeed after every 57 lines and finishing on the word 'hello'. Don't worry if you have not quite got the hang of it. Trial-and-error is the best method of teaching yourself!

That ends the list of the more useful extended PIP parameters. As you can judge for yourself, PIP is a very versatile program. It may not be the most user-friendly, but then CP/M itself was never renowned for that!

Other programs on the system disc

The rest of firing-up CP/M this month will be devoted to explaining how, why and when to use numerous application programs on the system disc that came with your computer.

I'll tackle the simpler of these programs first and gradually climb the ladder, till the worst offenders have been properly dealt with.

DISCCOPY.COM and friends

DISCCOPY, as the name implies, will let you copy discs - in CP/M, data or IBM formats. This is only useful for single-drive systems. If you have dual disc drives, you also have a program entitled COPYDISC. These files are available under version 2.2 of CP/M.

DISCKIT3 is a useful file to have around with CP/M Plus instead of Disccopy and Copydisc. It allows copying of a complete disc and also caters for formatting discs in system, data or vendor format. (If you desire IBM format you will have to use FORMAT as found on version 2.2 system discs - more of that later.)

DISCKIT2 is another version available in CP/M 2.2, which works with only a single bank of 64k memory, as opposed to Disckit3's using both banks in the CPC 6128. Disckit2 is consequently slower.

If you need to verify that one disc is identical to another, then use DISCCHK (for single-drive users) or CHKDISC (dual drives).

We come now to copying individual

files. Users of two drives should use PIP (which has been dealt with in great depth). For those with a one-drive system, Pip will prompt you when to swap discs if you have CP/M Plus. The following utility is supplied for single-disc users with CP/M version 2.2.

FILECOPY will copy only files that have a R/W and DIR status (read-write and directory status are explained in previous articles). You type:

FILECOPY filename

You can copy more than one file if you use the '>' and '*' wildcards.

Copying cassettes to disc

Included on the system disc are a couple of programs called CLOAD and CSAVE. Cload reads a cassette file and writes it to disc. Csave quite appropriately reads a file from disc and outputs it to cassette.

Cload needs two parameters, separated by spaces or comma:

```
CLOAD "cassettefile" discfile
```

The cassette filename to read is enclosed in double quotes (") - if a filename is not present, the first file on cassette will load. If a disc filename is not set, it will be saved to disc under the same name as that of the cassette filename. (This must be a maximum of 8 characters to be compatible with the CP/M operating system.)

The programs transferred are saved as Ascii files, so don't expect them to work correctly without modification. Another point to beware is that protected files cannot be transferred.

CSAVE is very similar to Cload, but there is a third optional parameter to deal with this time. The first two behave in precisely the same manner as Cload's. The third deals with cassette write speed: a '0' for 1000 baud and a '1' for 2000 baud - makes sense:

```
CSAVE discfile, "cassettefile",
writespeed
```

That ends the copying saga, and the various files that will do the job.

Formatting - as simple as 1, 2, 3

FORMAT, supplied on CP/M version 2.2 discs, has the ability to format a disc in any of four ways. These are System, Vendor, Data and IBM format.

To select a particular format, type a parameter after the Format command: S, V, D or I. For example:

```
FORMAT D
```

causes a disc to be data-formatted. Just typing Format (and return) will use System format as default.

A system-formatted disc will have two reserved system tracks copied onto it. These contain part of the CP/M operating system.

DISCKIT2 and 3 also have functions for formatting a disc, but make no provision for IBM format. All formats are selected from menus when using the DISCKIT programs - very nice and simple!

To SETUP or not - is there a doubt?

The setup file allows users to configure discs to meet their particular basic requirements. These parameters or setup values are held on a special sector (configuration sector) on the system track.

A few of the items SETUP allows you to modify include the sign-on string, keyboard translations, drive characteristics and baud rates for the serial interface.

Typing setup will invoke several messages, most asking you whether to change an option or proceed to the next. Here is an outline of them:

Initial command buffer. You can store 128 characters of commands to be executed automatically. For example, if you want a directory of the disc every time you boot up CP/M, do this when asked to enter the 'new initial command buffer': type dir ^M and Enter. You should be familiar with DIR by now; the ^M (Control-M, not two separate characters) stands for Enter or Return.

Using control codes, it is possible to change border, paper, ink, mode and window size among other things. It is even possible to have a program execute as soon as CP/M has been entered - try and figure that one out for yourself.

Sign-on string. This is a string of characters - up to 253 - that is output to the screen when CP/M is invoked.

If you enter 'Utilities disc 1986 - Jack Jones' as the sign-on string, exactly that will be printed every time you enter CP/M. Personalised discs - what more could you wish for!

Printer power-up string is very similar to the sign-on string. As the name implies, the message you type here will be sent to the printer each time you boot up. This could be a text message or, perhaps more usefully, a set of control codes to set up your printer.

I'm most thankful for the option *Keyboard translation table*, which lets me redefine the codes generated when I press certain keys. Each line in the table has four columns to fill in: key number, normal, shift and control.

Key numbers refers to the actual key to redefine. There should be a comprehensive list at the back of your Amstrad User Instructions - or it's handily printed on the 6128's case. The remaining three columns allow different values to be generated when a key is pressed on its own, with the Shift key or with Control.

A nice example would be to set up the cursor keys (which happen to have codes 8,0,2 and 1) so that application programs will reconise them. Here they are set for WordStar, which in its ordinary unfriendly way (considering the letters, 'unfriendly' may be the wrong word!) would expect you to use Control with the keys S, E, X and D - Ascii 19, 5, 24 and 4:

Key code	Normal	Shift	Control
8	19	1	141
0	5	18	142
2	24	3	143
1	4	6	144

If you wish to alter the keyboard translations, you are presented with a menu closely resembling the following:

A - Add key translation to table.
D - Delete key translation.
C - Clear translation.
F - Finish translation.

To enter the first line that I gave as an example, you would type A, 8, 19, 1, 141.

We'll have to look at the remaining Setup questions in a future instalment. They are rather technical, and this month's space is full. To be safe, best to say no to the rest of the options.

However, before you go, look at the last two unanswered questions.

Do you want to update your system disc? (Y/N). I would suggest that you go through all the options a number of times - till you are confident of what you are changing and the effects they will have on your disc before answering yes to this question.

The final question asks whether or not you wish to restart CP/M. Well, it would be nice to see what you have done - because the whole purpose of Setup is to make life with discs more pleasant.

Simulating Navigational Computers

by Peter Doutch

How many readers of this magazine have got the ambition to become an airline pilot. I can see it now. I put my hand on the joystick (yoke as it is called) of the 747, trundle down the runway and in two minutes I am in the air. I then push a whole lot of buttons and then let the computer fly the plane for me. What a life, relying on the computer to do everything. I am not sure how all the computers on board a 747 work except one, the most important. The VORTAC computer.

On most airplanes today, either light or commercial, is a little computer called the VORTAC computer (well, that's what I'll call it). So what is this computer and what does it do? I will explain.

The VORTAC computer stands for "VHF omni range system (VORTAC)" which is a navigational computer. Around the world are a number of control towers (I'm not sure how many) that give out a signal which is picked up by the VORTAC computer. From this signal the computer can calculate the magnetic course and distance to a given airport for the pilot.

Figure 1
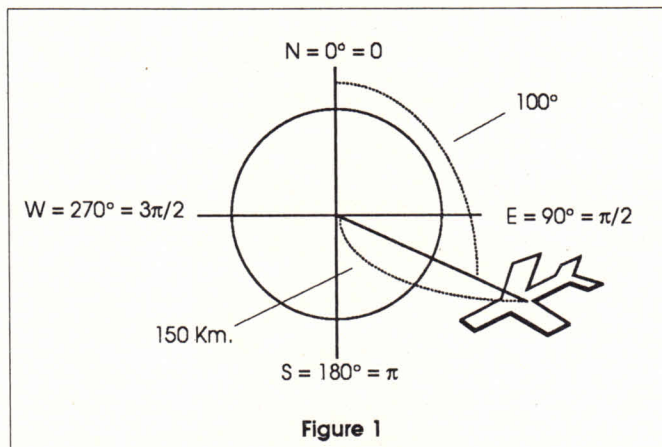
The pilot then sets his course given to him by the computer. So how does it work?

The concepts behind the operation of this navigational system are simple enough for a form five student to understand. The basic information the pilot receives in the cockpit is his position relative to the VORTAC, given in polar co-ordinates.

The pilot in the illustration Fig 1 would describe his position (obtained from his radio instruments) as being "on the 100° radial of the South Gippsland (SGT) VOR, 150 Km out."

The pilot could calculate that by turning right, and flying a course of 260° he would arrive at SGT.

This is good, the pilot knows how to get to the VOR. However the location of the VOR is not always the same as the location of the destination airport. If the pilot flew to the VOR then to the airport, problem solved. Yes the problem is solved, but living in the eighties and using sophisticated computers (like an AMSTRAD), we can instruct the pilot to steer a course from his current position which will take him straight to the destination airport.

Before we write a program to tackle this feat it is best if we work it out mathematically, there are various different ways of doing it such as this one:

PROBLEM 1

The aircraft is 389 Km. out on the 12° radial from the VOR. The destination airport is 500 Km. out on the 176° radial. We wish to find the course the pilot has to steer to reach the destination airport and the distance the plane has to travel.

If you refer to Fig 2 the following values are as such: s1=389, e=12°, s3=500 and d=176°.

Step 1

Angle 'a2' is found by subtracting angle 'e' from angle 'd', hence a2 = 176° - 12° = 164°.

Step 2

To find the distance 's2' we must use the COSINE RULE, I will not go into what the cosine rule is.

$$s2 = \sqrt{s3^2 + s1^2 - (2 * s3 * s1 * COS(a2))}$$

Hence, when we use figures we get:

$$s2 = \sqrt{500^2 + 389^2 - (2 * 500 * 389 * COS(164))} = 880.48384$$

Step 3

To find angle 'a3' we must first find the SINE and COSINE of 'a3'. If you were to do it with a scientific calculator you could work out the value of 'a3' by using the inverse sine button, however the computer does not have an inverse sine command. Instead it has an inverse tangent command commonly known as Arc-Tangent. The mathematics for using arc-tangents is shown below:

$$sina3 = (s3 * SIN(a2)) / s2$$
$$cosa3 = \sqrt{(1 - sina3^2)}$$
$$a3 = ATN(sina3 / cosa3)$$

With figures we get:

$$sina3 = (500 * 0.2756373) / 880.48384 = 0.156526$$
$$cosa3 = \sqrt{(1 - 0.156526)} = 0.9876738$$
$$a3 = ATN(0.156526 / 0.987638) = 9.0053101$$

STEP 4

The last step is to find the angle 'x', this is done by adding angle 'e' to 180° then subtracting angle 'a3'.

$$x° = 180° + e° - a3°$$
$$x° = 180° + 12° - 9.0053101° = 182.99469$$

So the output to the pilot would be to steer a course of 183° radial and travel a distance of 880 Km.

NOTE: *When I was writing down all these figures, they came from a calculator with an 8 digit display. The Amstrad is a little bit more accurate in its maths so don't be alarmed if you get slightly different answers.*

Now that we know the maths behind writing such a program, lets do it. Guess what? I have made your job easier and have already written down one. It can be found under listing 1. The original program I wrote was coded in Turbo Pascal. Then as very few people use Pascal (well I don't think many do) I had to convert my code into the BASIC.

If you can follow this article, you will have no problem following the program, it is very straightforward.

Simulations like this one are so very straightforward and show how powerful your Amstrad really is. If you have any suggestions for other different types of simulations you wish to see appear in this mag, just drop in a letter to the Editor of the mag. I'm sure he will pass them on.

Figure 2
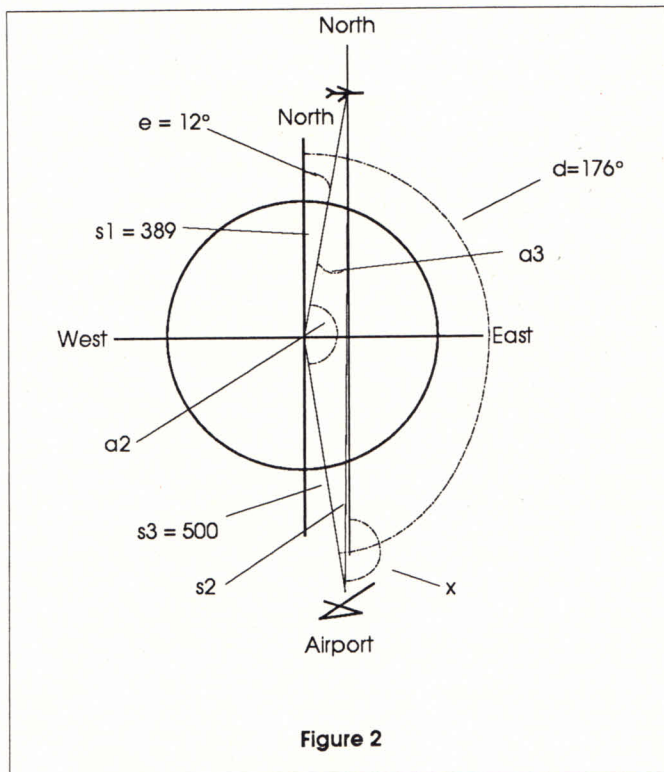
Listing 1

```
100 'Program     Vortac;
110 'Variables Used
120 '        s1,s2,s3,a2,a3
130 '        e,d,x,sina3,cosa3 : All Reals;
140 '        loop              : An Integer;
150 '        ans               : A string of characters;
160 GOTO 800
170 'Procedure Draw_Box;
180   FOR loop=(x1+1) TO (x1+1-1)
190     LOCATE loop,y1:    PRINT CHR$(154);
200     LOCATE loop,y1+w:  PRINT CHR$(154);
210   NEXT loop
220   FOR loop=(y1+1) TO (y1+w-1)
230     LOCATE x1,loop:    PRINT CHR$(149);
240     LOCATE x1+1,loop:  PRINT CHR$(149);
250   NEXT
260   LOCATE x1+1,y1+w:    PRINT CHR$(153);
270   LOCATE x1+1,y1:      PRINT CHR$(156);
280   LOCATE x1,y1:        PRINT CHR$(150);
290   LOCATE x1,y1+w:      PRINT CHR$(147);
300 RETURN
310 'End of Draw_Box........
320 '{---------------------------------------}
330 'Procedure Collect_Inputs;
340   WINDOW#0,4,76,4,20:PAPER 1:PEN 0:CLS
350   LOCATE 13,2:PRINT "Enter Radial Of VORTAC On Which A
ircraft Lies.";
360   LOCATE 16,6:  PRINT "Enter Distance Of Aircraft From
 VORTAC.";
370   LOCATE 13,10: PRINT "Enter Radial Of VORTAC On Which
 Airport Lies.";
380   LOCATE 17,14: PRINT "Enter Distance Of Airport From
VORTAC.";
390   INK 1,0: LOCATE 35,4: INPUT "",e
400   LOCATE 35,8: INPUT "",s1:LOCATE 35,12: INPUT "",d:LOCA
TE 35,16: INPUT "",s3:PEN 1:PAPER 0:CLS
410 RETURN
420 'End of Collect_Inputs........
430 '{---------------------------------------}
440 'Procedure Set_Up_Screen;
450   PAPER 0:PEN 1:MODE 2
460   x1=2:y1=2:1=78:w=21:GOSUB 180 :' DRAW_BOX........
470   LOCATE 34,2: PRINT CHR$(24);" ^ VORTAC ^ ";CHR$(24);
480   x1=33:y1=1:1=13:w=2:GOSUB 180 :' DRAW_BOX........
490   LOCATE 39,22: PRINT "By";
500   LOCATE 23,23: PRINT SPACE$(10);"PETER T. DOUTCH.";SP
ACE$(8);
510   x1=22:y1=21:1=34:w=4:GOSUB 180:' DRAW_BOX........
520   LOCATE 24,24: PRINT CHR$(164);" 1988, SPELLBOUND PRO
DUCTIONS. ";
530 RETURN
540 'End of Set_Up_Screen........
```

Words Work - 1

The first part of a new series providing inside information on popular wordprocessors

Here you are. A section of the magazine devoted to the word-processor user. Every month we'll look at certain aspects of word-processing: speeding things up, making a neater job, tips, tricks and answering your queries. To remain sane we have decided to limit the number of word-processors covered: the more popular such as Protext, Tasword and Wordstar will be looked at. We kick off this month with a look at setting up.

It is important to have your word-processor tuned to your taste. This makes life easy and production a lot faster. Things such as colour choice and key combinations may seem trivial, but if you constantly use a word-processor these trivial irritations become unbearable.

Tuning Tasword
Various versions of Tasword exist: Tasword 464, Tasword D and Tasword 6128. They all have a Basic loader which makes it easy to set them up.

Load the Basic Tasword loader and type LIST. You should see Basic commands such as KEY, BORDER and INK. If you wish to customise Tasword it is best to remove all these commands.

Tasword works in Mode 2, which means only two colours are allowed on the screen: one background colour (PAPER) and one foreground (PEN). If you want the text black, and the border and paper in white, you would need to add a new line to the loader, like this: 23 INK 0,26: BORDER 26: INK 1,0.

If you're not sure how the Ink and Border commands work, a quick scan through the User Instructions won't go amiss. However, here are a few pointers:

the number following BORDER refers to the colour; the first number after INK refers to the inkpot (in this case there are only two, the background and foreground). The second number indicates the colour. The CPC manual has a list of numbers and the colours they represent.

Using KEY
When word-processing, it is handy to have often-used phrases or sequences of control codes available at the touch of a single key. This effect can be implemented using Basic's KEY command. For example, try entering this line:

```
24 KEY 0,"The Amstrad User" +
CHR$(13) + "Strategy Publications" +
CHR$(13) + "1/245 Springvale Road" +
CHR$(13) + "Glen Waverley" +
CHR$(13) + "Vic 3150" + CHR$(13).
```

This causes our full address to appear just by pressing f0. The "CHR$(13)" is of course the Ascii code for end-of-line.

Naturally, you can place whatever you like in the function keys. Remember you have 32 of them: Shift and Control states also count. Take care that the content of an individual key doesn't exceed 120 characters - otherwise you'll get an error message.

Once you have added all the bits and pieces to the Tasword loader, save it. Whenever you load Tasword, it will be set to your requirements.

It is possible to alter everything described earlier from within Tasword, but unfortunately it's only temporary. There is no option for saving the alteration: every time you load Tasword you will have to configure it to your liking - what a pain. You must go into Basic to do it permanently.

On the other hand..... the manual goes into considerable detail describing how to customise and then save Tasword. As the whole operation is menu-driven, it may be a safer way than directly amending the Basic program.

Playing with Protext
Arnor's Protext is similar to Tasword in that it has a small Basic loader before the main program. Again you can create a personalised loader if you wish - in fact, Arnor encourages this.

A green screen: a black background (that is, INK 0,0) and white text (INK 1,26) is the easiest to work with. Before the final document is put through to the typesetter, it needs codes (just as the dot-matrix printer needs control codes for italics and bold). Phototypesetter codes are usually more wordy: the programmable function keys come in handy. For example, to produce slanted print we use the command [/]; [|] stands for letters upright again. In the Protext loader we define certain function keys to produce these and other effects.

The Protext manual lists special codes if you want to build various keystrokes onto a function key: Shift-⇨, for instance, takes the cursor to the next word, and in Protext this is CHR$(247). By making good use of function keys you can save a good deal of time and effort.

It is very easy to make tiny alterations in the setup of Protext during operation: typing quit or q from its command mode returns you to Basic - not only that, but you'll find the loader program still there: just type LIST. From here you can alter colours, redefine keys or program function keys. Typing |P returns you to Protext - neat.

Discovering Pyraword
Pyraword is customised in a different way from Tasword or Protext: everything is altered from within Pyraword and saved as a User Options file. This means you can have numerous setup files and can pick a particular one for a particular job.

Nine function keys are available under Pyraword, and each can hold 32 characters. Pressing f0 displays their contents;

the small Enter key lets you alter them. As Pyraword needs two keystrokes to get to the top or bottom of a document, it makes sense to use a function key for this purpose.

From edit mode, Control-Z takes you to command mode. From here you can alter all sorts of things: pressing J or K will change the paper and pen colours respectively. L or S will load or save the User Options: colours, tab stops and keystroke memories. When Pyraword loads, it searches for a User Options file called Wpsvopys.Usr (not mentioned in the manual). If you save your alterations under this name then there is no need to load them manually.

Wordstar under 2.2

This section deals with creating an everyday Wordstar work disc. It will instruct you how to auto-start Wordstar, select the colours and define keys, among other things. Before going any further, you will need a CPM 2.2 system disc and a disc containing Wordstar.

Boot up your system disc and type SETUP at the A> prompt. Eventually a message will appear: Initial command buffer...is this correct? (Y/N). Before answering, insert your Wordstar disc into the drive. Then press N. This allows you to enter a command that will automatically execute when you type |CPM at the beginning of the day.

Your entry should look like this: WS↑M. As you know, WS is the filename for Wordstar. The up-arrow (↑) tells the system a control-key instruction follows; in this case Control-M means Return or Enter.

Once you've typed that lot, press the Return key. A confirmatory message will appear - just press Y. Another option will come to view: Sign-on string...Correct? All the options will give you the option of altering them. Say yes-it's correct in each case unless stated otherwise.

From here, colours can be selected and text displayed on boot-up. This could be a useful sign-on string:
↑\@ww↑\a@@↑]wwWordstar work disc↑M↑JCustomised by Fred Sugar↑M↑Jversion 2.11↑M↑J|↑M↑J.

This will set the paper and border colour to white and text to black. It will also display a message. By all means use your own messages and set the colours that you find comfortable to work with.

The next option, when you get to it, is Printer power-up string... You can use this to send control-characters or text to the printer - for example, to switch on NLQ (near-letter quality) or italic print. Control characters again are prefixed with "↑". Here's a suggestion: ↑[@Wordstar printer initialisation↑M↑↑M↑J. It resets the printer and prints a short message.

Your printer must be on-line when you boot CPM. If it isn't, CPM waits until it is.

Custom keys

The next function is possibly the most important. It lets you alter key definitions. Wordstar uses the standard CPM cursor-movement keys: Control with S, D, E and X for left, right, up and down. Not only are these keys a pain to remember, but they are awkward to access. How much nicer to define the cursor cluster (which wasn't invented when Wordstar was born) to do this job. Easily done from the option Keyboard translation...

The Cursor Diamond
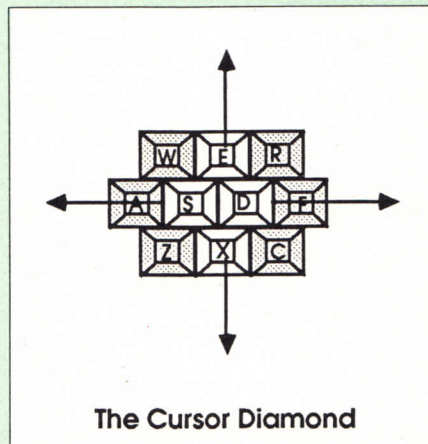
Selecting this option produces a sub-menu:
```
A - add key translation (key no.,
normal, shift, control)
D - delete key translations (key no)
C - clear all translations
F - finish translations
```
We are interested in only A and F. Type exactly the following, pressing Return at the end of each line:
```
A 8,19,1,141
A 0,5,18,142
A 2,24,3,143
A 1,4,6,144
A 17,123,91
A 19,125,93
A 16,7,7,20
F
```
The A at the beginning of each line indicates the option you wish to use. The next number refers to a key number. If your machine is a 664 or 6128 look at the top of the disc drive: there is a handy diagram of the keyboard and key numbers. 464 owners will find it in the User Instructions. The rest of the numbers on the line tell the machine what character to print when the key is pressed on its own, with Shift or with Control - in that order. At the back of the User Instructions manual is a table showing characters and their corresponding Ascii codes.

The 8 in the first line is the key-number for ← on the keypad. The next number, 19 is the character that will appear when the left-arrow is pressed: Control-S, Wordstar's "cursor left". Shift-← is programmed by the next number: Control-A (Ascii 1) is used by Wordstar to move to the left by a word. The last figure, affecting Control-←, is above 127 which means it isn't an Ascii character. Wordstar and other CPM applications will ignore this.

Function keys

Certain functions in Wordstar need two or three keypresses - Control-QR to get to the top of the text, for example. It is very easy to hit the wrong key or forget the correct sequence. This is where the next Setup option, keyboard expansions..., comes in handy. It works on the same principle as Basic's KEY command: you can define the function keys to produce a string of characters when you hit one of them.

Entering function-key definitions is done the same way as defining keys. Some useful definitions:
```
A 0,↑KS
A 1,↑K↑J|↑F
A 2,↑K↑B
A 3,↑K↑K
```
The first number refers to the function key (0 to 9). Following is the key sequence. In our example, f0 will save the current file and return you to the editor, f1 will show the disc catalogue from which you can delete a file. The other two are for start- and end-block markers.

And that was the last Setup option of any real use to the Wordstar user. Answer Y to all other questions. The penultimate option will save all the changes to your Wordstar disc. And finally CPM will re-boot letting you see all your alterations. Experiment till your system suits you.

The urge to explain

Ian Sinclair, successful author of many books including CPC and PCW tutorials, has some words of encouragement for technical writers.

Most people think of 'writing a book' as writing a novel, but the truth of the matter is that the vast majority of books published are non-fiction. Armed with your PCW, all you need to make a little money from your writing is expertise in your chosen area, an organised mind and the ability to express yourself clearly on paper.

There is a widely-held belief that anyone who has knowledge of a subject can write a technical book or article on that subject, but a quick glance at some of the manuals that come with computers and their software should dispel that fantasy. The unfortunate fact is that technical writing is a skill that is acquired by experience, and it's not always easy to obtain the experience. It's rather like Catch-22 - your work will be accepted when you have the experience, and you never get the experience until your work has been accepted. How, then, do you get yourself into the business of technical writing?

The obvious necessity is to have some technical expertise. This doesn't mean that you should be an expert, far from it. Only too often the expert is the worst possible technical writer because he or she never realises that readers are not experts. Certainly if you are writing for post-graduates you need to be an acknowledged expert in your field, perhaps the acknowledged expert. For any other purposes, and particularly if you are writing for the general public, what you can get over is a lot more important than what you know, always assuming, of course, that you know enough to understand the topic for yourself.

In short, don't be intimidated just because you don't understand everything about your subject, but make sure you know really thoroughly about what you are explaining. If you want to write about car servicing you don't need to know how to replace a clutch, but you certainly should know how to adjust the cable and check for correct operation.

Practice makes perfect

Technical know-how also implies have-done. You quite definitely must have practical experience of what you are writing about, because unless you have tried something for yourself you don't really know about it. You may be quite convinced after reading a manual for a piece of software that you know how it works, but it's almost certain that when you start using it for yourself you will find that your understanding of the manual does not always correspond with what actually happens. Would you know from reading the LocoScript manual that you cannot replace each occurrence of 'definitely' by 'definitely' (underlined, Brian) because the under-lining command cannot be placed into a search-and-replace? One fellow-writer came unstuck on that one not long ago, and I still have painful memories of assuming that the commands on the Amstrad CPC664 computer would follow a predictable pattern. Never assume that anything will be what it seems, particularly if it has any connection with computing.

A foot in the door

How do you get the sort of experience that is useful for technical writing? Technical teaching is by far the best grounding that anyone can have for learning how topics have to be explained. As any technical teacher will tell you, the first thing you discover when you start teaching a subject is just how little of it you really understand, as distinct from how much you remember. The second thing you learn the hard way is that you don't know how well you have communicated something until you find what someone else has made of it, and this is a form of feedback you don't get when you write. In particular, it gets you out of the habit of using jargon words like 'boot', 'syntax' and all the rest, before you have explained them thoroughly, a major fault in a lot of technical writing.

We can't all get this type of experience, even part-time with evening classes, so how do the rest of us get started? You should start in a small way, which has the great advantage of getting your name known. Articles in magazines perform this function admirably, and there are few technical authors I know who have not made a start in this way. Writing articles gives you experience in gathering facts, organising them in a logical sequence and explaining methods - the key parts of technical writing. In addition, magazine editors are more inclined to take an interest if they think you have some useful expertise. There will be more about magazine article writing in an article (how else?) in a future issue.

Booking your place

Breaking into books is something you do once you have made your name with articles, because by then you can write a preliminary letter to an editor that can include references to some of your published work. In this preliminary letter it is important to include an outline of the proposed book, with chapter headings and summary of contents, number of words and illustrations. Even then, it can be as much a matter of luck as good judgement in finding an editor who wants he sort of title that you are offering. Don't expect it to be easy, and don't feel that because your proposed book doesn't fit one editor's list that it won't fit another. You do not have to have written the book at this stage, because if you have made a thorough job of the outline synopsis the book will not be difficult or take long to write.

To start with, then select your publisher with some care. This may seem obvious, but I knew someone who submitted a book on Fresh-water Ecology to Mills & Boon. To some extent, selecting a publisher is easier for the technical writer, because you are likely to have come across books from the publishers who are most active in your field of expertise. You can find out a publisher's address by going into a bookshop or library and looking on the back covers or copyright pages of their titles.

You should get a reply to your preliminary letter fairly soon, and if the publisher is interested you will be expected, unless you are very well-known, to submit a sample chapter. This need not be the first chapter of your book, but it helps if it is, because the first chapter is usually the most difficult to write.

The sample

The first chapter is where you want your readers to start, so that it's here that you outline what level of knowledge you assume the reader already has so that he/she can decide very quickly if the book will be suitable. The less you assume that the reader knows, the more readers will find your book useful. For example, don't assume that because your readers possess a PCW computer they must surely understand how to use CP/M. Chapter 1 sets the tone of the book, and should take more time than any other chapter in the book, even if you are not submitting a sample. It should be laid out in a logical sequence, broken up into reasonably short paragraphs, with sub-titles as needed. The printing should be on A4, double-spaced, one side only. The assessment of this sample will take time, because it has to be sent to someone who can judge it technically, perhaps another author. If you hear no more after a month, however, it does no harm to ring up and enquire.

Reading a sample means that a publisher has invested some time and money on your idea, so that if your book proposal has any merit, but your writing techniques could be improved, you are likely to be told rather than have the proposal turned down completely. If you get a quick rejection at this stage, it's likely that there is a serious flaw in your sample, and you will have to re-draft it before you try again with another publisher. Sooner or later, if there is any merit in the idea, you will receive a contract in the post. Some contracts specify an advance, which will be returnable if you don't deliver. Watch also for the royalty figure - some contracts specify 10% of cover price, which is good, but others state 10% of money received, which amounts to 4% or less of cover price and means that large sales do not translate into a lot of cash for you.

You then have to submit the rest of your work. Even if delivery on disc is called for, always print a paper copy even if only for your own eyes, because faults that seem to pass un-noticed on the screen will scream at you from a piece of paper. Make sure that everything is as you want it, because alterations to a book once it has been set into print are ruinously expensive, even in these days of computer typesetting.

Once you send off the manuscript start planning your next book, with the experience of the first behind you. The important thing is to keep up a continuous effort, because if your first book succeeds you will need a second one to pay the tax bill, and if the first one does not succeed you will want to try again anyhow.

The rewards

Finally, what can you expect to get out of all this work? The most important answer is satisfaction, knowing that you have imparted some useful information to readers. You will not make a fortune unless you happen to hit a lucky jackpot, and you should not expect to be able to become a full-time author overnight. It took me 23 years of part time work to get to the full-time stage, and it can still be a precarious life. Don't expect any perks like computers on loan, though if you are writing in other technical fields you may find firms more co-operative. You have at least one advantage that was denied to me in my early days - a computer that is an excellent word-processor and which can also be used for your book-keeping and data-processing. That's progress.
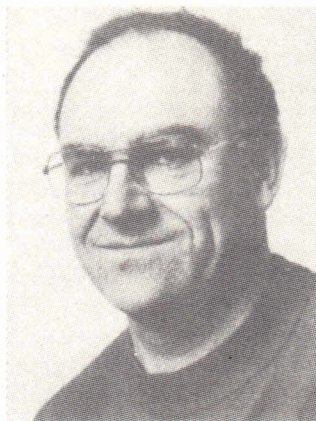
Ian Sinclair started writing technical articles while working with English Electric Valve Co. on TV camera tubes. Moving to technical teaching in 1966, he started to write textbooks for the courses (Electronics and Physics) that he taught. When the first home computers became available, he decided that the manuals could bear improvement and launched into a series of books on this type of topic. This led to becoming a full-time author in 1983 and he now has a total of 96 books published on a variety of technical topics.

Sordid Details

Part Three of our Logo series moves on from pretty pattens to more serious programming

Over the past three issues we've looked at Dr. Logo's commands to control the screen and move the turtle around, introducing some programming techniques along the way. Drawing pretty shapes is all very well, but to be able to write useful logo programs there are some basic programming steps to be mastered.

One of the most attractive features of Logo on the PCW is that it can print high resolution graphics directly on the screen - ie. it can turn individual dots on and off on its grid of dots 720 wide by 520 high. Remember that all BASIC can do is print specific characters on a 90 by 32 grid - 130 times coarser resolution!

Learning how to control this high resolution screen is very useful. If your work involves the need to draw graphs, for instance, you can plot out your results or graph particular functions. To do this you need to know how to calculate arithmetic functions in Logo, how to place a dot at a particular point on the screen, and it helps to be able to read typed inputs from the keyboard.

Logo can do most of the mathematical calculations that BASIC can, with the help of the now-familiar make command:

```
make "x 1 + 1
```
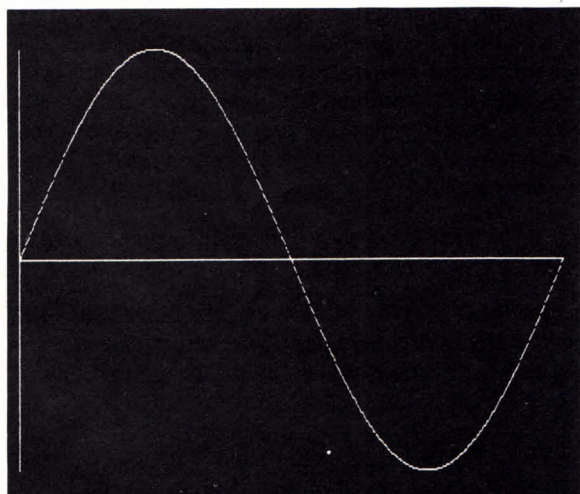

sets up a variable x with the value 2. You can use brackets in expressions that get complicated, so

```
make "x (1 + 2) / (3 * 4)
```

sets x to 0.25. For the mathematically minded, a range of trigonometry functions are available too. Sine (sin), cosine (cos) and inverse tangent (arctan) are all there, so if you wanted to work out the tangent of variable y (sine divided by cosine) it would be

```
make "x (sin :y) / (cos :y)
```

Another useful function is random, which (you'll never guess) generates random numbers. Logo's random numbers are always whole numbers, not decimals, between zero and the number you give, so random 50 generates a number between 0 and 49 inclusive.

Screen test

The next thing to do is to get to grips with the a screen. As you've seen from previous articles, the main way to draw on the screen is to move the turtle to the starting point of the line to be drawn and then to the end of the line - as it moves it leaves the line behind it as a trail. Whenever you move the turtle it leaves its trail, so don't forget to do a pu (pen up) before you move to the start then pd (pen down) before you draw the line. To make a start on the graph drawing program then, it seems like a good idea to draw some axes.

The Logo screen, you will remember, extends from -260 to +260 vertically and -360 to +360 horizontally. Assuming that you only want to cope with positive values of x you would want to draw a y-axis from (-300, -250) to (-300, 250) and an x-axis from (-300, 0) to (300, 0) you give the commands

```
pu setpos [-300 -250] pd fd 500
pu setpos [-300 0] pd rt 90 fd 600
```

Drawing dots as opposed to lines is a little easier - you don't need to mess around lifting the pen up and down before each dot. To put a dot at a co-ordinate (40,50) you just say

```
dot [40 50]
```

dot expects a list of co-ordinates to follow, which makes things slightly trickier if the x and y co-ordinate values are held in variables since you can't say dot [:x :y]. Instead,

you must use the command

```
dot (list :x :y)
```

which converts :x and :y into the form that Logo expects.

The only thing left to do before drawing a graph is to decide what values of x are going to be used. This is a good opportunity to ask the user what values to use when the program is run. Logo has three different commands to read from the keyboard: rc waits for the user to type a single character, rq reads a word ended by [RETURN] (similar to BASIC's INPUT), and rl reads a line of characters (similar to BASIC's LINE INPUT) and makes them into a list. As examples, if you typed key z while Logo was waiting to complete a

```
make "x rc
```

command, variable x would get set to 'z'. If you type 123 in response to

```
make "x rq
```

x would become 123. If you type hello boys and girls to

```
make "x rl
```

x would become [hello boys and girls] - a list.

A graph a minute

So to Listing One. The first thing to do is to decide what graph you want to draw - in the example y=sin(x) is drawn. This is set up by the procedure calc.

draw falls into three parts. First it asks the user what values of x the graph is to span. Then knowing that because the Logo x-axis is 600 dots along it works out how much to increment x by for each dot. Finally, there is a repeat loop to calculate where to put the dot for each of the 600 values of x.

If you just type in the listing shown and run it by saying draw, you can generate the example screen by replying 0 and 360 for the first and last x values. It takes a minute or so to draw the graph.

If you want to draw something a little more exciting than sin(x), you can edit calc to whatever you want - type ed "calc, alter it as you want then [EXIT]. All you have to do is

ensure that for any value of x that is given to calc, it sets up a value of y between -250 and +250 (these being the extremes of the y axis on the screen). So, you could try replacing the guts of calc by

```
make "y (sin :x) * (sin :x) * 250 with x from 0
to 720
make "y :x * :x * (:x - 1) * 50 with x from 0 to
2
```

or any other familiar formula. Don't forget to do a save "draw when you're about to quit Logo.

Listing One

```
to draw
pr [first x value?]
make "first rq
pr [last x value?]
make "last rq
make "increment (:last - :first) / 600
fs cs ht
pu setpos [-300 -250] pd fd 500
pu setpos [-300 0] pd rt 90 fd 600
make "x :first
make "xpos -300
repeat 600 [calc :x dot (list :xpos :y)
        make "x (:x + :increment) make "xpos
        (:xpos + 1)]
end

to calc :x
make "y (sin :x) * 250
end
```

If only...

When writing a program, the three things you have to master are input/output of data, repeating loops of commands and making decisions. This last one is perhaps the most important and is covered by the if primitive, which has been used in previous examples but has never been properly explained.

The basic form of the command is if *condition* [*commands to do if condition is true*] [*commands to do if condition is false*]. You don't have to have the second part if you don't need it. Here are some examples:

```
if rc = "x [pr [you typed x]] [pr [you didn't
type x]]
```

type this directly at Logo's ? prompt and it will wait for you to press a key. If you press x, Logo cleverly tells you that you typed x, otherwise it tells you that you didn't type x. Similarly,

```
if :a = 0 [make "b 0] [make "b 1 / :a]
```

will set up a variable b to be either 1/a or 0 if a is 0 (handy


```
?guess
Think of a number between 1 and 100
I guess 98
Is your number higher, lower or the same (h / l / s) ?
I guess 3
Is your number higher, lower or the same (h / l / s) ?
I guess 83
Is your number higher, lower or the same (h / l / s) ?
I guess 34
Is your number higher, lower or the same (h / l / s) ?
I guess 38
Is your number higher, lower or the same (h / l / s) ?
I guess 61
Is your number higher, lower or the same (h / l / s) ?
Whoop - de - doo!
?
```

since asking it to work out 1/0 would generate an error).

To show the `if` statement in action, Listing Two is a simple game where you think of a number between 1 and 100 and the PCW then guesses it. Whenever it makes a guess, you have to tell it whether the correct number is higher than the number you thought of, lower than it or the same.

Listing Two

```
to guess
pr [Think of a number between 1 and 100]
make "least 1
make "most 100
ask
end

to ask
local "answer
make "best :least + random (1 + :most -
:least)
pr (list "I "guess :best)
pr [Is your number higher, lower or the same
(h / l / s) ?]
make "answer rc
if :answer = "h [make "least :best + 1 ask]
if :answer = "l [make "most :best - 1 ask]
if :answer = "s [pr [Whoop - de - doo!]]
end
```

The program keeps a track of the current highest and lowest limits for the number, so if you say that a guess is lower than the correct number the program knows that the maximum the number could possibly be is 1 less than its guess. If the number is higher than the guess, the lower limit is modified to 1 greater than the guess. Then the next guess is a random number somewhere between the lower and upper limits.

The only thing to be careful with in random numbers is that you cover the range you want to properly. `random 100` will generate a random whole number between 0 and 99, so to make a variable with a random number between 1 and 100 you have to say

```
make "guess 1 + random 100
```

Other points of interest in Listing Two are how the looping is controlled to go on guessing until the right number is found. If the PCW guesses wrong and you type h or l to the guess, the current limits are modified and the asking procedure calls itself - you're using recursion. If you type s, the PCW has guessed right, ask prints out a happy message but doesn't call itself again and the calling chain unwinds naturally. It's important that :answer is a local variable, otherwise 'Whoop-de-doo' will print out over and over again as the recursion unwinds. Try omitting the `local "answer` line and see what happens.

What are lists?

Many times over the last few Logo articles funny things called 'lists' have cropped up. Next month's gripping instalment of the Logo series will cover them properly, but a quick survey will help you to understand how pr and other commands work.

Single names or numbers in Logo are called 'words', "fred and 1 are both words. If you want to pass groups of words around in a Logo program you combine them into a list, which has square brackets around it. So [1 2 3 fred] is a list. Logo never tries to work out what the words in a list mean. This is why you don't need the double quote mark before 'fred' if it is inside the list brackets. If you have a variable x set to 3 then [12:x] will be a list of three 'words' 1,2 and :x. If you want to get x's *value* into the list, you need to use the list-making primitive `list`. The command (list 1 2 :x) works out that x is 3 and generates the list [1 2 3].

In some cases - particularly with the print command pr - you can think of lists just like strings of characters in BASIC. pr [This is a message] will print out 'This is a message' just like the BASIC command PRINT "This is a message". But because Logo thinks of lists as collections of words separated by a single space, if you say pr [1 2 3] you will just get '1 2 3' coming out, single spaced. Also, pr [in/out] will produce 'in/out', since '/' is a 'word' by itself (being Logo's division sign).

If you want to extract words from within lists, there are a variety of commands. For example, if shopping is a list containing [chips chocs tea vegemite], then

pr first :shopping prints the first item of the list, chips

pr last :shopping prints the last item, vegemite

pr bf :shopping prints all but the first item, [chocs tea vegemite]

pr bl :shopping prints all but the last item, [chips chocs tea]

More of lists, and some useful applications next month. Can you wait?

Pretty printing

There is no explicit logo command to send drawings to the printer rather than the screen, but you can do a screen dump by pressing [EXTRA]+[PTR] once the screen display you want is showing (PCW only). While this will put the picture on paper, it is small and will include the status line.

An alternative and a cunning way to get an A4 size printout was explained in Tip-Offs on Page 35 of the November 1987 issue. It's a Basic program which takes data from a saved LOGO picture file (.PIC filetype) and loads it byte by byte into a string, then prints a column of eight dots by LPRINTing the string.

The program is only 27 lines long, and is well worth the small amount of effort typing it in.

Trifold Trifles

Take a nostalgic trip back to the early days of video games when many an evening was spent bouncing a ball against a bat or blasting alien invaders out of the universe.

SQUASH
by W.D. Atkin

Our older readers will shed a nostalgic tear at this little program. The bouncing ball game was a forerunner of all the high technology zap-'em-up arcade games of today. In those days people quite happily fed 10p's into a machine all day just to experience the heady pleasure of batting a tiny computerised ball round the screen. But in those days we thought digital watches were pretty neat too.

The game itself is simplicity to play. You move the bat (the green flat thing at the bottom) from side to side using the cursor keys. All you need to do is to be there when the bouncing ball lands. It even adds up how many times you hit the ball and tells you at the end. What could be simpler? Our best score is six but we never were very co-ordinated.

And for all those who worry about having to type in long listings this one is only 16 lines long. Don your sweatbands, flex your fingers and give it a whirl.

```
1 'PCW SQUASH
2 'BY W.D. Atkin
3 'The Amstrad User Feb 1988
10 PRINT CHR$(27) + "E" + CHR$(27) + "Y":PNT = 0
20 DEF FNAT$(X,Y)= CHR$(27) + "Y" + CHR$(X+32) + C
HR$(Y+32)
30 BAP = 38:LR=1:UD=-1:BLX=20:BLY=21:PRINT FNAT$(B
LX,BLY)"O"
40 GOSUB 150:PRINT FNAT$(1,20);CHR$(150);
50 FOR X = 20 TO 60 :PRINT CHR$(154);:NEXT X:PRINT
 CHR$(156)
60 FOR X = 2 TO 19:Y=20:PRINT FNAT$(X,Y)CHR$(149):
Y=62:PRINT FNAT$(X,Y)CHR$(149)
65 NEXT X
70 PRINT FNAT$(BLX,BLY)" ":BLX=BLX+UD:BLY=BLY+LR:P
RINT FNAT$(BLX,BLY)"O"
80 IF BLY >59 THEN LR=-1 ELSE IF BLY <22 THEN LR=1
90 IF BLX < 3 THEN UD=1:PNT=PNT+1
100 IF BLX <20 THEN GOTO 120
110 IF ABS(BLY-(BAP+3))>1 THEN GOTO 160 ELSE UD=-1
:GOTO 70
120 MOV$ =INKEY$:IF MOV$=CHR$(1) AND BAP >20 THEN
BAP =BAP -2
130 IF MOV$=CHR$(6) AND BAP < 60 THEN BAP=BAP+2
140 GOSUB 150:GOTO 70
150 PRINT FNAT$(20,BAP)" "CHR$(154)CHR$(154)CHR$(1
54)" ":RETURN
160 PRINT FNAT$(25,10)"YOUR SCORE IS "PNT
```

INVADERS
by W. Melbourne

Another trip down memory lane, this time a Space Invaders type clone for the PCW.

Playing is simple. You can move your little gun from side to side with 'z' (left) and 'x' (you guessed it - right) and 'm' fires the bullets. After rigorous testing we suggest you hit the flying saucer before it gets to the bottom.

```
1 'PCW INVADERS
2 'BY W. Melbourne
3 'The Amstrad User Feb 1988
10 PNT=0:A$=CHR$(27):PRINT A$+"f":CLS$=A$+"E"+A$+"
Y"
20 DEF FNAT$(X,Y)=A$+"Y"+CHR$(X+32)+CHR$(Y+32):PRI
NT CLS$
30 FOR P=10 TO 24:PRINT FNAT$(P,18);CHR$(149):PRIN
T FNAT$(P,73);CHR$(149):NEXT P
40 FOR P=18 TO 73:PRINT FNAT$(9,P);CHR$(95):PRINT
FNAT$(25,P);CHR$(95):NEXT P
50 GX=23:GY=19:IX=10:IY=45
60 IY=IY+(INT((RND*2)+1.5))*2:PRINT FNAT$(IX,IY);C
HR$(139)
70 FOR K=1 TO 150:NEXT K:PRINT FNAT$(IX,IY);" ":IF
 IY>61 THEN GOSUB 140
80 FOR J=1 TO INT(RND*110):NEXT J:PRINT FNAT$(GX,G
Y);CHR$(129)
90 MOV$=INKEY$:IF MOV$=CHR$(122) OR MOV$=CHR$(120)
 THEN PRINT FNAT$(GX,GY);" "
100 IF MOV$=CHR$(122) AND GY>19 THEN GY=GY-4
110 IF MOV$=CHR$(120) AND GY<71 THEN GY=GY+4
120 IF MOV$=CHR$(109) THEN GOSUB 170
130 PRINT FNAT$(3,41);"SCORE =";PNT:GOTO 60
140 PRINT FNAT$(IX,IY);" ":IY=21:IX=IX+3
150 IF IX>22 THEN PRINT CLS$,FNAT$(16,35)"YOUR SCO
RE IS";PNT:PRINT A$+"e":END
160 RETURN
170 FOR A=GX TO 12 STEP-1:PRINT FNAT$(A+1,GY);" ",
FNAT$(A,GY);CHR$(148):NEXT A
180 PRINT FNAT$(A+1,GY);" ":IF GY=(IY+2) THEN PRIN
T CHR$(7):PNT=PNT+1:GOTO 50
190 RETURN
```

TELETYPE
by J.D. Sault

Here is an extremely short program which will print any message of your choice on the screen at teletype speed - complete with totally authentic clacking noises. We thought you would have fun trying this out if you had a moment to spare.

```
1 'PCW TELETYPE
2 'BY J.D. SAULT
3 'The Amstrad User Feb 1988
10 READ x$
20 FOR x=1 TO LEN(x$):PRINT MID$(X$,X,1);CHR$(7);:
FOR xx= 1 TO 150
30 NEXT:NEXT
40 DATA This is the message that will be printed o
n the screen
```

Protext and Survive

Ron Ainsley begins a short series on the pros and cons of Arnor's Protext word processor with a LocoScript conversion course.

In LocoScript most of you are probably by now happily writing letters from a letterhead template resplendent with italics, bold and underlines with both hands tied behind your back, but to achieve the same simple objective in Protext may seem to those of us weaned on Loco unpleasantly like an adventure game. In fact there's nothing illogical or unreasonable about Protext; it's just very different from LocoScript.

You'll need your Protext work disc, lots of paper, a packed lunch and a can of beer. Load Protext and put your data disc in. Protext has been designed to accommodate LocoScript converts, so as many of the keys as possible are similar - in LocoScript 1 you press [f1] to change discs; in Protext, to get a directory of the disc you can press [f1] also to get a disc directory.

In edit mode most of the cursor keys, [DEL], [EOL], [PARA] keys etc work just as in Loco, and [ALT] plus the up or down cursor moves by a screenful at a time. However, editing which messes up the paragraph won't be automatically reformatted - you have to manually reformat (from the cursor to the end of the paragraph) by entering [ALT]-F.

Protext doesn't have quite the same concept of Templates and Groups as LocoScript does, but there is a simple way to achieve the same effect. First create your letterhead template by going into edit and typing against left hand margin

```
>sm 10
>pl 55
>cp off
```

These are 'stored commands' - the > tells Protext not to print the line but to obey the command after it, the above three respectively setting the side margin to 10, the page length to 55, continuous printing off (ie. single sheet).

For the next line enter [ALT]-Xe; the 'e' shows in reverse video and is a print command to make the letter come out in 12 pitch, or 'elite', type. Protext thinks 10 pt is miles better and prints everything in that if you don't put your foot down.

Then enter your telephone number and address as on a letterhead, using the [TAB] key to align your address - if you aren't happy with where the tabs are defined, you can alter them as explained in a minute. Back in command mode SAVE the file as TEMPLATE.STD or something equally boring. There's nothing special about the name; in Loco it would be used as a template for every new document, but as far as Protext is concerned it's just another file.

To use this template file in future you would just tell Protext to LOAD TEMPLATE.STD. Move the cursor down to where the body of the letter will start, type in the text and when finished go back to command mode and type SAVE LETTER.MUM or whatever you are going to call the finished file.

Take me to your ruler

The equivalent of Loco's layouts are 'ruler lines'. The line of hyphens and exclamation marks at the top is a ruler line where L shows the left-hand margin, R the right, !s show the positions of the tabs. The hyphens are just padding characters to fill the gaps.

If you want to put a new layout somewhere in a document, to inset a quotation in the middle of your letter for example, you just enter a ruler line like

```
>        L—!————————————R
```

and all the text below this will be indented to the new left margin at L, stopping at R. You may have to do an [ALT]-F to reformat any existing text below that point if you want to. Pressing [ALT]-R will restore the old ruler line. A ruler line in the template will be the default for documents prepared under that template.

To get all those italics, bold and underlinings, you enter more print control codes. The equivalent of (+Italic) or (-Italic) in LocoScript is [ALT]-Xi in Protext; the first [ALT]-Xi turns italics on, then another one turns them off. It shows on the screen as a reversed video 'i', and takes up a space on screen but won't on the printout. Similarly [ALT]-Xb selects or deselects bold and

Chalk and Cheese

LocoScript is a very good wordprocessor. It has enabled literally hundreds of thousands of people to produce neat, impressive, well laid-out letters and manuscripts. It's virtually foolproof; it constantly checks to ensure that all the document is well laid-out and correctly formatted, for example - which tends to slow it down.

Protext is much, much faster but not as straightforward - it's rather like the difference between a point-and-shoot compact camera and a motor driven SLR. Which you prefer is a matter of taste, both have their fanatical adherents. Protext comes with a spelling checker, an instant word counter, full mail merge facilities and a battery of features like a calculator, simultaneous two-file editing, two column printing etc. There is also a stripped-down version 'Pocket Protext' (reviewed last month), without the spelling checker and mailmerger.

[ALT]-Xu underlines.

To get a printout of the current file you are working on, give the command PRINT [RETURN] (or just P will do), say 'Y' to background printing, and out will come your letter. P always prints in draft mode, and PQ will print in high quality if you want that instead. You can stop printing by entering STOP (if you want to resume later, which is done by typing CO) or AB (to abandon printing completely, if you get a bad paper jam for instance).

Course in management

All LocoScript's disc management functions - copying, renaming, erasing etc - are done in Protext's command mode. To copy a file, enter the command COPY[RETURN] - you're asked for the old filename, ie. the name of the file to be copied, and the file to copy it into.

To make a backup of your letter on a disc in the B drive, for example, you would enter the old filename as LETTER.MUM and on hitting [RETURN] up comes the prompt 'New name or drive', so type B[RETURN] (on an 8256 you'll be prompted to put the disc for B into the drive). Typing a new name will make a copy of the file on the same disc under the specified name.

Renaming a file is done by RENAME or just REN, and again you're asked for the old and new filenames. Erasing is done with ERASE or ERA. Whenever you save a document, Protext obligingly renames the previous version of it with the extension .BAK as a backup file. This corresponds to Loco's idea of 'Limbo' files, where if you edit a document the old version is saved as a Limbo file in case you need to go back to it.

Score with headers

Now, having finished a letter to your literary agent, you can get down to printing an extract from your new book to send with it. Suppose you have already typed the extract in and you want to print it out on continuous paper with headers, footers and page numbers.

First you want to set up a template as before, to be saved as say TEMPLATE.MS, consisting of the stored commands:

```
>ls 2
>cp on
>sm 10
>pn 1
>fo              Page %
e
>h           The Importance of Ron
```

The first four commands set the line spacing to two, continuous

Loco note

To convert a document from LocoScript 1 to Protext, you'll have to make an ASCII file way back in LocoScript via [f7]=Modes in the main menu (take the 'simple text file' option) making sure to put it in group 0 of the disc and LOAD the document. You will have to reformat it with the FORMAT command but otherwise it'll now be a Protext document. In making an ASCII file though you lose all the fancy formats, italics, bolds and pitch commands.

printing on, side margin to ten, and set the first page number to 1. The next one he sets the footer text which puts the page number at the bottom of every page; trial and error will get the centring right. The % is a signal to Protext to 'print page number here', as per Loco's (+PageNo).

The header is defined after the embedded e for 12 pitch text, ie after the start of the first page, to ensure the header is printed out from page 2 onwards. If you want the header on page one as well you can put it before the start of the first page (before the embedded codes) but it will be printed out in 10 pitch as the codes haven't taken effect yet. You can turn off headers or footers at any point in the text by putting >he on or >fo off somewhere on the page before the one at which the change is to take effect.

May all merge

As the extract you're sending to your agent has probably already been written, it would be easier to somehow lay it on top of the template just defined rather than have to type in the whole thing again.

You can do this with a MERGE command. If your document RON.DOC has been prepared in Protext, all you have to do is LOAD RON.DOC and, without leaving command mode, MERGE TEMPLATE.MS. The template will be inserted at the position of the cursor, ie. at the start of the document. Save the result as RON.MS (or RON.DOC if you don't need to keep the original). Reformat if necessary by entering FORMAT, load the continuous paper in the printer, and print out as above. Now drink the beer.

Justifying decisions

'Justification' in Protext means having a smooth right hand margin on the page. Pressing [ALT]-J while editing will turn justification on or off - you'll see if it's currently selected by a status message on the information lines at the top of the edit screen. By default it's on; if you want to unjustify a justified document, or vice versa, select [ALT]-J to change justification state, and reformat the entire document from command mode by typing FORMAT.

Protext normally justifies its text by inserting whole space characters between words in the line, rather than LocoScript's nicer way of inserting a fraction of a space between every word so giving an even spacing. If you put at the top of your template >ms on - this turns 'microspacing' on and makes Protext do proper justification. Protext isn't very good at this and it takes a long time to print out a microspaced document on the standard PCW printer.

You can 'right justify' a line (ie. align it with the right margin but leave the left end ragged) by pressing [TAB] after the final tab stop you have defined on the current ruler line. This will take the cursor to the extreme right edge of the screen.

Count your blessings

Merely by entering COUNT in command mode, the number of words in the working document will appear almost instantaneously. You can then use the built-in calculator to multiply by your rate per word to work out how much to charge for your article. Many people would gladly sell their house and all its contents for such a mindboggling useful facility.

Finance Program

from John Dunsmuir

Being a father of five grown up children I still get asked for advice on money matters, not the "can you lend me" kind, but more serious ones like buying houses or investing savings. To enable me to help them I wrote the following program which I have found extremely useful. More experienced readers will no doubt find improvements to my simple method - but it works.

The Saga of Jane

Jane had just turned 18 and got her driving licence. She was passing 'Honest Joe's' car yard when she spotted a bright blue continental sports car. Pausing to admire it, Joe approached her and asked if she liked it. She said she did but couldn't afford a car at $5999. Joe, perceiving that she was keen on the car, suggested that they talk about it as he was sure he could make a deal that she would find attractive. He asked her how much she could put up as a deposit. She said that she only had $500. Joe said "how much can you afford to pay a month - would $150 be within your reach?" She thought for a moment and said yes. Joe said "I will do a special deal for you - I will give you the car for $5500, you can pay $500 down and $150 a month and it is yours."

Jane thought for a moment and then started to ask some questions. How long will it take to pay off? Five years said Joe. What is the interest on the loan? Just over 15% said Joe then looking up in his book he said it was actually 16% simple interest. She said she would have to get the $500 out of her savings account and would be back.

When she got home she told her father who ran Joe's figures through his Amstrad computer choosing the "money borrowed option" with $5000 loan for 5 years at 12 payments per annum, 0 (zero) for interest as it is an unknown element and finally the monthly payments of $150. The result showed a true compound interest of 26.1%. Her father then ran up the investment option and told her that if she could spare $150 per month and invest it at 13.5% it would grow to $4106.25 in two years. She could then afford to buy a good car and still have money left over to buy petrol and she would have proved that she could save $150 per month without getting into the hands of the repossessors. He also entered the "money borrowed option" - $5000 over 5 years, 12 payments per annum at 26.1% interest - to see what Joe's deal would have cost. The result? A shade short of $9000!

[With humble apologies to all honest car traders]

```
1 'PCW FINANCE PROGRAM
2 'BY John Dunsmuir, Ocean Grove
3 'The Amstrad User Feb 1988
10 ON ERROR GOTO 20
20   GOTO 50
30  C = (P*(1 +J/100/F)^(Y*F)*J/100/F)/((1+ J/100/
F)^(Y*F)-1):RETURN
40 X= P*(1+I/F/100)^(J + 1)-R*((1+I/F/100)^(J + 1)
-1) /(I/F/100):RETURN
50 PRINT CHR$(27) + "E" + CHR$(27) + "H"
60 PRINT"FOR MONEY BORROWED OR INVESTED FOLLOW INS
TRUCTIONS BELOW":PRINT
70 PRINT"FOR MONEY BORROWED TYPE 'B'":PRINT
80 PRINT"FOR MONEY INVESTED IN A LUMP SUM TYPE 'L'
":PRINT
90 PRINT"FOR MONEY INVESTED AT REGULAR INTERVALS T
YPE 'R'"
100 PRINT:PRINT
110  Z$ = INKEY$
120 IF Z$ = "B" THEN 150 ELSE IF Z$ = "b" THEN 150

130 IF Z$ = "L" THEN 1530 ELSE IF Z$ = "l" THEN 15
30
140 IF Z$ = "R" THEN 2660 ELSE IF Z$ = "r" THEN 26
60 ELSE 110
150 PRINT CHR$(27) + "E" + CHR$(27) + "H"
160 PRINT"THIS PROGRAM CALCALCULATES REPAYMENTS IN
TEREST & PAYOUT VALUE FOR LOANS"
170 PRINT:PRINT"            ANSWER ALL QUES
TIONS - PRESS RETURN"
180 PRINT"            TYPE (0) ZERO  FOR UN
KNOWN QUANTITY"
190 PRINT:PRINT:INPUT"AMOUNT OF LOAN
 $ ",P
200 PRINT:PRINT:PRINT:INPUT"YEARS TO REPAY
       ",Y
210 PRINT:PRINT:PRINT:INPUT"NUMBER OF PAYMENTS PER
 ANNUM        ",F
220 IF F = 0 GOTO 210
230 PRINT:PRINT:PRINT:INPUT"INTEREST PERCENT PER A
NNUM        ",I
240 PRINT:PRINT:PRINT:INPUT"PAYMENTS per PERIOD
     $ ",B
250 IF B=0 THEN 290
260 IF P=0 THEN 770
270 IF Y=0 THEN 980
280 IF I=0 THEN 1230
290 PRINT CHR$(27) + "E" + CHR$(27) + "H"
300 R=(P*(1+I/100/F)^(Y*F)*I/100/F)/((1+I/100/F)^(
Y*F)-1)
310 PRINT "AMOUNT DUE EVERY ";:PRINT INT(364/F);:P
RINT" DAYS = $";
320 PRINT USING"####.##";R
330 PRINT"YEAR";:PRINT TAB(6)"WEEK";:PRINT TAB(13)
"AMOUNT";
340 PRINT TAB(22)"INTEREST";:PRINT TAB(34)"PAYOUT"
350 FOR J = 0 TO (Y*F-1)
360 PRINT TAB(2) FIX(J/F+1);
370 PRINT TAB(7) USING"##"; (((J/F+1)-INT(J/F+1))*5
2)+INT(52/F);
380 PRINT TAB(11)USING"######.##"; (R*J+R);
390 GOSUB 40
400 PRINT TAB(22)USING"######.##"; (R*(J+1)+X-P);
410 GOSUB 40
420 PRINT TAB(32)USING"######.##";X
430 NEXT
```

```
440 PRINT"YEAR";:PRINT TAB(6)"WEEK";:PRINT TAB(13)
"AMOUNT";
450 PRINT TAB(22)"INTEREST";:PRINT TAB(34)"PAYOUT"
460 PRINT "AMOUNT DUE EVERY ";:PRINT INT(364/F);:P
RINT" DAYS = $";
470 PRINT USING"####.##";R:PRINT
480 PRINT
490 PRINT"TO PRINT RESULTS TYPE 'Y' - TO RERUN PRE
SS SPACE BAR OR 'X' TO QUIT"
500 A$ = INKEY$
510 IF A$="y"THEN 540 ELSE IF A$="Y" THEN 540 ELSE
 IF A$ = CHR$(32) THEN 150
520 IF A$ = "x" THEN 20 ELSE IF A$ = "X" THEN 20 E
LSE 500
530 END
540 LPRINT"AMOUNT OF LOAN                     $ ";:LP
RINT USING"#####.##";P
550 LPRINT"YEARS TO REPAY                        ";
:LPRINT USING"###.#";Y
560 LPRINT"NUMBER OF PAYMENTS PER ANNUM           ";
:LPRINT USING"###";F
570 LPRINT"INTEREST PERCENT PER ANNUM             ";
:LPRINT USING"###.##";I
580 LPRINT "AMOUNT DUE EVERY ";:LPRINT INT(364/F);
:LPRINT" DAYS = $";
590 LPRINT USING"####.##";R
600 LPRINT"YEAR";:LPRINT TAB(6)"WEEK";:LPRINT TAB(
13)"AMOUNT";
610 LPRINT TAB(22)"INTEREST";:LPRINT TAB(34)"PAYOU
T"
620 FOR J = 0 TO (Y*F-1)
630 LPRINT TAB(2) FIX(J/F+1);
640 LPRINT TAB(7) USING"##";(((J/F+1)-INT(J/F+1))*
52)+INT(52/F);
650 LPRINT TAB(11)USING"######.##";(R*J+R);
660  GOSUB 40
670 LPRINT TAB(22)USING"######.##";(R*(J+1)+X-P);
680 GOSUB 40
690 LPRINT TAB(32)USING"######.##";X
700  NEXT
710 LPRINT"YEAR";:LPRINT TAB(6)"WEEK";:LPRINT TAB(
13)"AMOUNT";
720 LPRINT TAB(22)"INTEREST";:LPRINT TAB(34)"PAYOU
T"
730 LPRINT "AMOUNT DUE EVERY ";:LPRINT INT(364/F);
:LPRINT" DAYS = $";
740 LPRINT USING"####.##";R
750 GOTO 500
760 END
770 PRINT CHR$(27) + "E" + CHR$(27) + "H"
780 P=B * ((1+I/F/100)^(Y*F)-1)/(1+I/F/100)^(Y*F)/
(I/F/100)
790 PRINT"AMOUNT OF LOAN                     $ ";
:PRINT USING"#####.##";P
800 PRINT"YEARS TO REPAY                       ";
:PRINT USING"##"; Y
810 PRINT"NUMBER OF PAYMENTS PER ANNUM           ";
:PRINT USING"##"; F
820 PRINT"INTEREST PERCENT PER ANNUM             ";
:PRINT USING"##.##"; I
830 PRINT "AMOUNT DUE EVERY               ";:PRINT IN
T(364/F);
840 PRINT " DAYS = $";:PRINT USING"###.##"; B:PRIN
T
850 PRINT"TO PRINT RESULTS TYPE 'Y' - TO RERUN PRE
SS SPACE BAR OR 'X' TO QUIT"
860 A$ = INKEY$
870 IF A$="y" THEN 900 ELSE IF A$="Y" THEN 900 ELS
E IF A$=CHR$(32) THEN 150
880 IF A$ = "x" THEN 20 ELSE IF A$ = "X" THEN 20 E
LSE 860
890 END
900 LPRINT"AMOUNT OF LOAN                     $ "
;:LPRINT USING"######.##";P
910 LPRINT"YEARS TO REPAY                      ";:LPRINT USING"##"; Y
920 LPRINT"NUMBER OF PAYMENTS PER ANNUM         ";:LPRINT USING"##"; F
930 LPRINT"INTEREST PERCENT PER ANNUM           ";:LPRINT USING"##.##"; I
940 LPRINT "AMOUNT DUE EVERY             ";:LPRINT
INT(364/F);
950 LPRINT " DAYS = $";:LPRINT USING"###.##"; B
960 LPRINT " "
970 GOTO 860
980 PRINT CHR$(27) + "H" + CHR$(27) + "E"
990 K=LOG10(B/(B-P*I/100/F))/LOG10(1+I/100/F)
1000 Y = INT(K/F)
1010 W = INT((K/F - Y)*52)
1020 PRINT"AMOUNT OF LOAN                    $ "
;:PRINT USING"######.##";P
1030 PRINT"YEARS TO REPAY                      ";::PRINT USING"##";Y;
1040 PRINT" &";:PRINT W;:PRINT" WEEKS"
1050 PRINT"NUMBER OF PAYMENTS PER ANNUM         ";:PRINT USING"##"; F
1060 PRINT"INTEREST PERCENT PER ANNUM           ";:PRINT USING"##.##";I
1070 PRINT "AMOUNT DUE EVERY          ";:PRINT I
NT(364/F);
1080 PRINT " DAYS = $";:PRINT USING"###.##"; B:PRI
NT
1090 PRINT"TO PRINT RESULTS TYPE 'Y' - TO RERUN PR
ESS SPACE BAR OR 'X' TO QUIT"
1100 A$ = INKEY$
1110 IF A$="Y"THEN 1140 ELSE IF A$ ="y" THEN 1140
ELSE IF A$=CHR$(32) THEN 150
1120 IF A$ = "x" THEN 20 ELSE IF A$ = "X" THEN 20
ELSE 1100
1130 END
1140 LPRINT"AMOUNT OF LOAN                     $
";:LPRINT USING"######.##";P
1150 LPRINT"YEARS TO REPAY                     ";:LPRINT USING"##"; Y;
1160 LPRINT" &";:LPRINT INT((Y-INT(Y))*52);:LPRINT
" WEEKS"
1170 LPRINT"NUMBER OF PAYMENTS PER ANNUM        ";:LPRINT USING"##"; F
1180 LPRINT"INTEREST PERCENT PER ANNUM          ";:LPRINT USING"##.##"; I
1190 LPRINT "AMOUNT DUE EVERY         ";:LPRINT IN
T(364/F);
1200 LPRINT " DAYS = $";:LPRINT USING"######.##";
B
1210 LPRINT " "
1220 GOTO 1100
1230 PRINT CHR$(27) + "E" + CHR$(27) + "H"
1240 A=255:D=1.9
1250 J=(A+D)/2:GOSUB 30
1260 IF J < 2 THEN 1290 ELSE IF J > 250 THEN 1290
1270 IF C>B THEN A = J ELSE IF C<B THEN D = J
1280 IF INT(C*100) = INT(B*100) THEN 1320 ELSE GOT
```

```
O 1250
1290 PRINT "                   INPUT DATA OUTSIDE OF PR
OGRAMME RANGE"
1300 PRINT"                            WAIT AND RE-NTER
 DATA"
1310 FOR J = 1 TO 5000 : NEXT:GOTO 150
1320 PRINT"AMOUNT OF LOAN                             $ "
;:PRINT USING"######.##";P
1330 PRINT"YEARS TO REPAY
";:PRINT USING"##"; Y
1340 PRINT"NUMBER OF PAYMENTS PER ANNUM
";:PRINT USING"###"; F .
1350 PRINT"INTEREST PERCENT PER ANNUM
";:PRINT USING"##.##";J;
1360 PRINT" %"
1370 PRINT "AMOUNT DUE EVERY           ";:PRINT INT
(364/F);
1380 PRINT " DAYS = $";:PRINT USING"#####.##"; B:P
RINT
1390 PRINT"TO PRINT RESULTS TYPE 'Y' - TO RERUN PR
ESS SPACE BAR OR 'X' TO QUIT"
1400 A$ = INKEY$
1410 IF A$="y" THEN 1440 ELSE IF A$="Y" THEN 1440
ELSE IF A$=CHR$(32) THEN 150
1420 IF A$ = "x" THEN 20 ELSE IF A$ = "X" THEN 20
ELSE 1400
1430 END
1440 LPRINT"AMOUNT OF LOAN                          $
";:LPRINT USING"######.##";P
1450 LPRINT"YEARS TO REPAY
";:LPRINT USING"##"; Y
1460 LPRINT"NUMBER OF PAYMENTS PER ANNUM
";:LPRINT USING"###"; F
1470 LPRINT"INTEREST PERCENT PER ANNUM
";:LPRINT USING"##.##";J;
1480 LPRINT " %"
1490 LPRINT"AMOUNT DUE EVERY            ";:LPRINT I
NT(364/F);
1500 LPRINT "DAYS = $";:LPRINT USING"#####.##"; B
1510 LPRINT" "
1520 GOTO 1400
1530 PRINT CHR$(27) + "E" + CHR$(27) + "H"
1540 PRINT"              ANSWER ALL QUESTIONS
- PRESS RETURN"
1550 PRINT"              TYPE (0) ZERO  FOR U
NKNOWN QUANTITY":PRINT:PRINT
1560 PRINT:PRINT:PRINT
1570 PRINT:PRINT:INPUT"AMOUNT DEPOSITED
 $",P
1580 PRINT:PRINT:INPUT"FOR HOW MANY YEARS
    ",Y
1590 PRINT:PRINT:PRINT"NUMBER OF INTEREST PAYMENTS
"
1600 INPUT"PER ANNUM                       ",F
1610 IF F = 0 THEN 1590 ELSE 1620
1620 PRINT:PRINT:INPUT"ANUAL INTEREST PER CENT
   ",I
1630 PRINT:PRINT:INPUT"ACCRUED AMOUNT
$",A
1640 IF P = 0 THEN 1660 ELSE IF Y = 0 THEN 1910 EL
SE 1650
1650 IF I = 0 THEN 2150 ELSE IF A = 0 THEN 2400 EL
SE 1570
1660 PRINT CHR$(27) + "E" + CHR$(27) + "H"
1670 P = A/(1+I/100/F)^(Y*F):PRINT:PRINT:PRINT
1680 PRINT"AMOUNT INVESTED                 $";:PR
INT USING"######.##";P
1690 PRINT:PRINT:
1700 PRINT"YEARS TO ACCRUE                  "
;:PRINT USING"##";Y
1710 PRINT:PRINT:
1720 PRINT"NUMBER OF INTEREST PAYMENTS"
1730 PRINT"PER ANNUM                        ";
:PRINT USING"###";F
1740 PRINT:PRINT:
1750 PRINT"INTEREST PERCENT PER ANNUM        ";
:PRINT USING"###.##";I
1760 PRINT:PRINT:
1770 PRINT"ACCRUED AMOUNT                $";:PRI
NT USING"#######.##";A
1780 PRINT:PRINT:
1790 PRINT"TO PRINT RESULTS TYPE 'Y' - TO RERUN PR
ESS SPACE BAR OR 'X' TO QUIT"
1800 A$ = INKEY$
1810 IF A$="y"THEN 1840 ELSE IF A$="Y" THEN 1840 E
LSE IF A$=CHR$(32) THEN 1530
1820 IF A$ = "x" THEN 20 ELSE IF A$ = "X" THEN 20
ELSE 1800
1830 END
1840 LPRINT"AMOUNT INVESTED                $";:LPR
INT USING"######.##";P
1850 LPRINT"YEARS TO ACCRUE                    ";
:LPRINT USING"##";Y
1860 LPRINT"NUMBER OF INTEREST PAYMENTS p/a      ";
:LPRINT USING"##";F
1870 LPRINT"INTEREST PERCENT PER ANNUM        ";:
LPRINT USING"###.##";I
1880 LPRINT"ACCRUED AMOUNT                $";:LPRI
NT USING"#######.##";A
1890 LPRINT " "
1900 GOTO 1800
1910 PRINT CHR$(27) + "E" + CHR$(27) + "H"
1920 Y = ((LOG(A)-LOG(P))/LOG(1+I/100/F))/F
1930 PRINT"AMOUNT INVESTED                 $";:PRINT
USING"######.##";P
1940 PRINT:PRINT:
1950 PRINT"YEARS TO ACCRUE                   ";:PR
INT USING"##.##";Y
1960 PRINT:PRINT:
1970 PRINT"NUMBER OF INTEREST PAYMENTS"
1980 PRINT"PER ANNUM                        ";:PRI
NT USING"###";F
1990 PRINT:PRINT:
2000 PRINT"INTEREST PERCENT per ANNUM        ";:PRI
NT USING"###.##";I
2010 PRINT:PRINT:
2020 PRINT"ACCRUED AMOUNT                $";:PRINT U
SING"#######.##";A
2030 PRINT:PRINT:
2040 PRINT"TO PRINT RESULTS TYPE 'Y' - TO RERUN PR
ESS SPACE BAR OR 'X' TO QUIT"
2050 A$ = INKEY$
2060 IF A$="y"THEN 2090 ELSE IF A$="Y" THEN 2090 E
LSE IF A$=CHR$(32) THEN 1530
2070 IF A$ = "x" THEN 20 ELSE IF A$ = "X" THEN 20
ELSE 1800
2080 END
2090 LPRINT"AMOUNT INVESTED                 $";:LP
RINT USING"######.##";P
2100 LPRINT"YEARS TO ACCRUE                    "
;:LPRINT USING"##.##";Y
```

```
2110 LPRINT"NUMBER OF INTEREST PAYMENTS p/a        "
;:LPRINT F
2115 LPRINT"INTEREST PERCENT per ANNUM            ";
:LPRINT USING"###.##";I
2120 LPRINT"ACCRUED AMOUNT                   $";:LPR
INT USING"#######.##";A
2130 LPRINT " "
2140 GOTO 2060
2150 PRINT CHR$(27) + "E" + CHR$(27) + "H"
2160 I=((A/P)^(1/Y/F)-1)*100*F
2170 PRINT"AMOUNT INVESTED              $";:PRIN
T USING"######.##";P
2180 PRINT:PRINT:
2190 PRINT"YEARS TO ACCRUE                    ";:
PRINT USING"##.##";Y
2200 PRINT:PRINT:
2210 PRINT"NUMBER OF INTEREST PAYMENTS"
2220 PRINT"PER ANNUM                        ";:P
RINT USING"###";F
2230 PRINT:PRINT:
2240 PRINT"INTEREST PERCENT PER ANNUM       ";:P
RINT USING"###.##";I
2250 PRINT:PRINT:
2260 PRINT"ACCRUED AMOUNT              $";:PRINT
 USING"#######.##";A
2270 PRINT:PRINT:
2280 PRINT"TO PRINT RESULTS TYPE 'Y' - TO RERUN PR
ESS SPACE BAR OR 'X' TO QUIT"
2290 A$ = INKEY$
2300 IF A$="y"THEN 2330 ELSE IF A$="Y" THEN 2330 E
LSE IF A$=CHR$(32) THEN 1530
2310 IF A$ = "x" THEN 20 ELSE IF A$ = "X" THEN 20
ELSE 2290
2320 END
2330 LPRINT"AMOUNT INVESTED             $";:LP
RINT USING"######.##";P
2340 LPRINT"YEARS TO ACCRUE                    "
;:LPRINT USING"##.##";Y
2350 LPRINT"NUMBER OF INTEREST PAYMENTS p/a     ";
:LPRINT USING"###";F
2360 LPRINT"INTEREST PERCENT PER ANNUM         ";
:LPRINT USING"###.##";I
2370 LPRINT"ACCRUED AMOUNT              $";:LPR
INT USING"######.##";A
2380 LPRINT " "
2390 GOTO 2290
2400 PRINT CHR$(27) + "E" + CHR$(27) + "H"
2410 A=P*(1+I/100/F)^(Y*F)
2420 PRINT:PRINT:PRINT
2430 PRINT"AMOUNT INVESTED              $";:PRIN
T USING"######.##";P
2440 PRINT:PRINT:
2450 PRINT"YEARS TO ACCRUE                    ";:
PRINT USING"##.##";Y
2460 PRINT:PRINT:
2470 PRINT"NUMBER OF INTEREST PAYMENTS"
2480 PRINT"PER ANNUM                        ";:P
RINT USING"###";F
2490 PRINT:PRINT
2500 PRINT"INTEREST PERCENT PER ANNUM       ";:P
RINT USING"###.##";I
2510 PRINT:PRINT:
2520 PRINT"ACCRUED AMOUNT              $";:PRINT
 USING"#######.##";A
2530 PRINT:PRINT:
2540 PRINT"TO PRINT RESULTS TYPE 'Y' - TO RERUN PR
ESS SPACE BAR OR 'X' TO QUIT"
2550 A$=INKEY$
2560 IF A$="y" THEN 2590 ELSE IF A$="Y" THEN 2590
ELSE IF A$=CHR$(32) THEN 1530
2570 IF A$ = "x" THEN 20 ELSE IF A$ = "X" THEN 20
ELSE 2550
2580 END
2590 LPRINT"AMOUNT INVESTED             $";:LP
RINT USING"#####.##";P
2600 LPRINT"YEARS TO ACCRUE                    "
;:LPRINT USING"##.##";Y
2610 LPRINT"NUMBER OF INTEREST PAYMENTS p/a     ";
:LPRINT USING"###";F
2620 LPRINT"INTEREST PERCENT PER ANNUM         ";
:LPRINT USING"###.##";I
2630 LPRINT"ACCRUED AMOUNT              $";:LPR
INT USING"######.##";A
2640 LPRINT " "
2650 GOTO 2550
2660 PRINT CHR$(27) + "E" + CHR$(27) + "H"
2670 PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRI
NT:PRINT:PRINT:PRINT:PRINT
2680 PRINT"            IF PERIOD PAYMENTS ARE UN
KNOWN TYPE ONE   (1)"
2690 PRINT"            IF ACCRUED AMOUNT IS UNKN
OWN TYPE ZERO      (0)"
2700 A$=INKEY$
2710 IF A$= "1" THEN 2730 ELSE IF A$ = "!" THEN 27
30
2720 IF A$ = "0" THEN 3090 ELSE IF A$ = ")" THEN 3
090 ELSE 2700
2730 PRINT CHR$(27) + "E" + CHR$(27) + "H"
2740 PRINT"              ANSWER ALL QUESTIONS
- PRESS RETURN"
2750 PRINT:PRINT
2760 INPUT"NUMBER OF DEPOSITS p.a.    ",P
2770 PRINT:PRINT
2780 INPUT"YEARS TO ACCRUE           ",Y
2790 PRINT:PRINT: IF Y = 0 THEN 2780
2800 INPUT"INTEREST CALCS. p.a.      ",M
2810 PRINT:PRINT: IF m = 0 THEN 2800
2820 INPUT" INTEREST % p.a.          ",I
2830 PRINT:PRINT
2840 INPUT"ACCRUED AMOUNT            $",S

2850 IF S = 0 THEN 2840
2860 N = Y * P
2870 R = S * ((1 + I/100/M)^(M/P)-1)/((1 + I/100/M
)^(M*N/P) -1)
2880 PRINT CHR$(27) + "E" + CHR$(27) + "H"
2890 PRINT"PERIOD DEPOSITS         $";:PRINT USIN
G"######.##";R
2900 PRINT"NUMBER OF DEPOSITS p.a.      ";:PRINT U
SING"###";P
2910 PRINT"YEARS TO ACCRUE          ";:PRINT
USING"##";Y
2920 PRINT"INTEREST CALCS. p.a.        ";:PRINT U
SING"###";M
2930 PRINT"INTEREST % p.a.            ";:PRINT U
SING"###.##";I
2940 PRINT"ACCRUED AMOUNT          $";:PRINT USIN
G"######.##";S
2950 PRINT"TO PRINT RESULTS TYPE 'Y' - TO RERUN PR
ESS SPACE BAR OR 'X' TO QUIT"
2960 A$ = INKEY$
```

```
2970 IF A$="y"THEN 3000 ELSE IF A$="Y" THEN 3000 E
LSE IF A$=CHR$(32) THEN 2730
2980 IF A$ = "x" THEN 20 ELSE IF A$ = "X" THEN 20
ELSE 2960
2990 END
3000 PRINT CHR$(27) + "E" + CHR$(27) + "H"
3010 LPRINT"PERIOD DEPOSITS             $";:LPRINT US
ING"######.##";R
3020 LPRINT"NUMBER OF DEPOSITS p.a.        ";:LPRINT
 USING"###";P
3030 LPRINT"YEARS TO ACCRUE              ";:LPRIN
T USING"##";Y
3040 LPRINT"INTEREST CALCS. p.a.         ";:LPRINT
 USING"###";M
3050 LPRINT"INTEREST % p.a.              ";:LPRINT
 USING"###.##";I
3060 LPRINT"ACCRUED AMOUNT            $";:LPRINT USI
NG"######.##";S
3070 LPRINT " "
3080 GOTO 2950
3090 PRINT CHR$(27) + "E" + CHR$(27) + "H"
3100 PRINT"             ANSWER ALL QUESTIONS
 - PRESS RETURN"
3110 PRINT:PRINT
3120 INPUT"NUMBER OF DEPOSITS p.a.     ",P
3130 IF P = 0 THEN 3120
3140 PRINT:PRINT
3150 INPUT"PERIOD DEPOSITS          $",R
3160 IF R = 0 THEN 3150
3170 PRINT:PRINT
3180 INPUT"YEARS TO ACCRUE           ",Y
3190 PRINT:PRINT:IF Y = 0 THEN 3180
3200 INPUT"INTEREST CALCS. p.a.      ",M
3210 PRINT:PRINT:IF M = 0 THEN 3200
3220 INPUT"INTEREST % p.a.           ",I
3230 N = Y * P
3240 S = R * ((1 + I/100/M)^(M*N/P)-1)/((1 + I/100
/M)^(M/P) - 1)
3250 PRINT CHR$(27) + "E" + CHR$(27) + "H"
3260 PRINT"PERIOD DEPOSITS            $";:PRINT USIN
G"######.##";R
3270 PRINT"NUMBER OF DEPOSITS p.a.        ";:PRINT U
SING"###";P
3280 PRINT"YEARS TO ACCRUE               ";:PRINT
USING"##.##";Y
3290 PRINT"INTEREST CALCS. p.a.          ";:PRINT U
SING"###";M
3300 PRINT"INTEREST % p.a.              ";:PRINT U
SING"###.##";I
3310 PRINT"ACCRUED AMOUNT            $";:PRINT USING"
#######.##";S
3320 PRINT"TO PRINT RESULTS TYPE 'Y' - TO RERUN PR
ESS SPACE BAR OR 'X' TO QUIT"
3330 A$ = INKEY$
3340 IF A$="y"THEN 3370 ELSE IF A$="Y" THEN 3370 E
LSE IF A$=CHR$(32) THEN 3090
3350 IF A$ = "x" THEN 20 ELSE IF A$ = "X" THEN 20
ELSE 3330
3360 END
3370 LPRINT"NUMBER OF DEPOSITS p.a.        ";:LPRIN
T USING"###";P
3380 LPRINT"PERIOD DEPOSITS            $";:LPRINT U
SING"######.##";R
3390 LPRINT"INTEREST CALCS. p.a.         ";:LPRIN
T USING"###";M
3400 LPRINT"YEARS TO ACCRUE              ";:LPRI
```

```
NT USING"##";Y
3410 LPRINT"INTEREST CALCS. p.a.         ";:LPRIN
T USING"###";M
3420 LPRINT"INTEREST % p.a.              ";:LPRIN
T USING"###.##";I
3430 LPRINT"ACCRUED AMOUNT           $";:LPRINT USI
NG"#######.##";S
3440 LPRINT " "
3450 GOTO 3330
```

How to type in a listing

The first thing is to load Mallard BASIC. Turn on your PCW, or reset it by pressing [SHIFT][EXTRA][EXIT], and put a copy of the CP/M master disc into drive A:.

When the A> prompt appears, type BASIC and press [RE-TURN]. After s few seconds a message about Mallard BASIC will appear on the screen, ending with the prompt 'OK'.

Type in each line of the listing very carefully, starting with the line number and finishing with [RETURN]. Be careful not to mix up capital I with the number 1, capital O with number 0 and colons with semi-colons. During a long listing, it's important to save your work every 15 minutes or so, and you should always save any listing before printing it. To do this, find a work disc with space on it, put it in the drive and type SAVE"PROGRAM"[RETURN]. Of course, you can choose any name of up to eight characters instead of 'PROGRAM'. When you've finished, type LIST[RETURN] and the whole program will appear on the screen. Check it, and if any lines are wrong, correct them with the line editor. For example, if there's a mistake in line 100, type EDIT 100[RETURN]. Use the cursor and delete keys to correct the line and press [RETURN] when you've finished. You can delete a whole line by typing its number and pressing [RETURN].

To run the program, simply type RUN[RETURN] and yes, the program goes wrong.

It's more than likely, no matter how carefully you typed in the listing, that it won't work properly the first time you run it. You may get an error message such as 'Syntax error in 100'. List the program out and check the screen against the original in the magazine.

Don't forget that the line number in the error message isn't necessarily where the error is - it's simply the point at which the PCW gets stuck. You may have to look elsewhere for the error.

When you find an error, either retype the complete line or use the line editor (described earlier) to correct it. Run the program again, and hopefully this time it will work. If it doesn't you have to go through the correction process again. Once the program is running correctly, save it again. To leave BASIC and return to CP/M, type SYSTEM.

Each time you want to run the program, you must load BASIC, type LOAD followed by the name of the file you saved it in, and type RUN once it's loaded.

Big Pipsqueak

One of CP/M's most useful and powerful commands is PIP, but is is also the command newcomers fear most. Alec Rae investigates.

If you are starting to branch out from LocoScript into the big world of CP/M, you are probably accruing piles of discs and backup discs with all the files you need spread over about ten of them. One of the first utilities you have to get to grips with in CP/M is the Peripheral Interchange Program (PIP to its friends) which allows you to copy files from disc to disc. And yet PIP can do so much more to help you control your disc files, it seems a crime to only use the barest and simplest facilities that this clever little program can provide.

Starting from basics, PIP is a way of copying a file. When CP/M addicts talk about 'pipping' a file they mean copying it - it's an extension of the old adage, "There ain't no noun that can't be verbed". To copy a file, give the command PIP, then the name where the file is to be copied to (the 'destination'), and finally the name of the file to copy (the 'source').

As per the usual CP/M convention, disc drives are referred to by their letter and a colon (A:, B: or M:), then the filename follows. If you are copying a file from one disc to another and want the copy to be called the same as the original then you needn't specify the new filename - PIP assumes the copy is to be called the same as the original unless you tell it differently.

You need to have the file PIP.COM on your current disc drive, or somewhere

on your PCW anyway. Here are a few examples to clarify things:

```
PIP M:=ZEBEDEE.DOC
```
(Make a copy of the current disc's ZEBEDEE.DOC file in the M drive)

```
PIP M:FLORENCE.DOC=ZEBEDEE.DOC
```
(Ditto, but call the M drive's copy FLORENCE.DOC instead of ZEBEDEE.DOC)

```
B:PIP M:DYLAN.DOC=A:DOUGAL.DOC
```
(PIP.COM is on the B drive, and use it to copy DOUGAL.DOC from the A drive to DYLAN.DOC on the M drive) Just remember, PIP destination=source - 'd' comes before 's'.

There are two ways to run PIP. Either type PIP before every line of files to be copied as in the above three examples, or you can go into PIP's 'multiple command mode'. Type PIP [RETURN] and you will see its asterisk prompt appear. You can now type just the command M:=ZEBEDEE.DOC or whatever. This has the virtue, if you are copying lots of files, of saving you typing three letters each time and also making PIP much faster. Normally whenever you type PIP CP/M has to look for PIP.COM, load it from disc and then run it. In multiple command mode it needs to do this only once. You can leave multiple command mode by pressing [RETURN] or [STOP] - if you are copying one file it is simpler just to precede your command with PIP which returns you to the CP/M A> prompt when the operation is completed.

You can use PIP to merge files while you are copying them. If you had written your grand meganovel as one chapter per file, you could merge them into one by
```
PIP
```

```
BOOK.DOC=CHAPTER.001,CHAPTER.002,CHAPTER.003
...
```

Just put all the files to be strung together one after the other on the 'source' side of the equals sign. Bear in mind that PIP doesn't like spaces in command lines except immediately after the PIP itself.

Go wild in the country
One of the most useful features of PIP's file copying powers is the use of 'wildcards' - those strange symbols that allow you to move any number of files of similar name at the same time.

The two wild card symbols are * and ?. If you use these symbols in a filename that CP/M is expecting, ? is taken to stand for any letter and * for any number of letters. Use the wildcard symbols instead of the filename and all the files that fit the criteria you have set down will be moved. So if you have 10 files ending '.COM' you can move them all to M drive by typing PIP M:=A:*.COM. If you want to move all the files named DOCUMENT.001 to DOCUMENT.009 to B drive type

```
PIP M:=A:DOCUMENT.00?
```

and the files will move. MYFILE.DOC, MYPROG.COM and MYPROG.BAS could all be moved to M: with the command PIP M:=A:MY*.*.
  PIP M:=A:MY???.* would move MYDOC.BAS but not MYFILE.BAS - because you used just three?s and there are four letters in FILE. You have to provide the correct number of ?s. The most useful is obviously PIP M:=A:*.* which will move every file to M drive.

What are your options?
PIP has a hundred and one hidden features which you get to by using its 'options.' Options are extra commands

to PIP which you put in square brackets immediately after the command line (no spaces between the file names and the opening square bracket). For instance the letter [C] will make PIP ask for confirmation before copying files. If you want the majority of files on your disc copied you type PIP M:=A:*.*[C] and then you will be asked MYFILE.DOC (Y/N?) and so on for each file on the disc so you can decide which files go and which don't. Just press Y to copy it, N not to copy it, and [STOP] to abort the whole copying process.

Other useful options are [A], designed to help you 'Archive' (ie back up) your files. PIP marks files that have already been archived, and if you use the [A] option when copying files with PIP only files that have been altered will be copied. So at the end of each day PIP B:=A:*.*[A] will back up all files you have changed that day without copying the entire disc.

The [G] option lets you read and write files to specific user groups. LocoScript, as you know, stores files in groups 0 to 7, so you could take a LocoScript disc and copy a document from group 3 to group 0 by

PIP A:[G0]=A:MYFILE.DOC[G3]

The [Z] option is useful to prepare a WordStar or Protext document for another word processor like LocoScript. It pays to make a copy of the document first of all with the [Z] option - PIP MYFILE.NEW=MYFILE.DOC[Z] - and load the new copy instead of the original. This removes all the strange formatting characters that Word Star etc. keep in their files (for the technical, is strips the eighth bit of the byte, or 'Zeroes the parity bit').

PIP to anything

PIP is not only used to copy files to discs. You can use the printer, the screen or the

Merging note

Don't be tempted to try the merging trick with LocoScript document files as it messes up completely, although you could do it with ASCII files created from LocoScript.

Optional extras

There seem to be as many different parameters you can set on PIP as there are letters in the alphabet although some of them become so obscure it is difficult to think up practical uses for them. PIP parameters can be combined together, with the one set of square brackets, eg.
PIP NEWFILE=AUX: [ZT8RV].
Here is a complete checklist of all PIP options:

[A]	Archive. Only copies files that have been changed since the last back-up.
[B]	'Block mode transfer'. We don't know what this one is for!
[C]	Confirm. Asks for confirmation before copying each file.
[Dnum]	Delete. Deletes all characters past the column specified in number. Use for files where the lines are too long for a device such as an 80 character printer.
[E]	Echo. Will type out a text file on screen as it is transferred, eg. PIP LST:=CON: [E].
[F]	Filter Formfeeds Takes out all the form-feeds in the document. To change the page lengths in a document see [P]
[Gnum]	Get. Will copy a file to or from the group number specified in the source or the destination file.
[H]	Hex. Checks data for hexadecimal format file. Reports any errors.
[I]	Ignore. Ignores :00 when transferring hexadecimal format files.
[K]	Kill. Suppresses the list of file names on screen when carrying out multiple operations like wildcard transfers.
[L]	Lower Case. Changes all capital letters in a file into lower case as it copies.
[N]	Number. Adds line numbers starting at 1 and increasing by one as the

file is copied. Using [N2] adds line nos. starting at 000001 and increasing by one.

[O]	Object file transfer. For transferring machine code files. Greatest advantage is that it will ignore any 'Control Z' in the file which would normally be taken as the end-of-file mark and stop copying of the file.
[P]	Page Inserts form feeds (end of page) every 60 lines. Use [Pnum] to set the form feeds at a specific number. Normally used with [F] to take out the form feeds already in file.
[Qstring^Z]	Quit. Stop copying the file at the word or words in the string. The string must be ended by ^Z, ie. [ALT]-Z. Can be used with [S] start option.
[R]	Read system files (as set up by SET filename [SYS]). Normally system files are ignored by PIP.
[Sstring^Z]	Start copying the file at word or words in the string. The end of the string must be marked by ^Z as with [Q].
[Tnum]	Tabs. Changes the tab characters into the number of spaces specified in number as it copies.
[U]	Upper case. Changes all alphabetic characters into capitals as it copies.
[V]	Verify. Checks that file has been transferred correctly by comparing the source and destination files.
[W]	Write over. Writes over files set to Read Only at destination without asking. Normally if you use PIP to write over a R/O file without [W] it will ask if you really want to do it.
[Z]	Zero the parity bit. Sets the 8th bit in each char. to zero in a character file effectively stripping out dot commands etc.

serial interface using this trusty utility. What does this mean? Well for a start if you wanted to quickly print out a text file all you need to do is type PIP LST:=MTFILE.DOC and MYFILE.DOC will be printed out. LST stands for listing device', what we in the computer world now call in our new fangled jargon 'a printer'.

Also note that if you are sending anything to or from a device you need a colon after it as though it were a disc drive - as in LST:.

Why not turn your PCW into an instant typewriter with PIP LST:=CON:? 'CON' is short for console, meaning either the keyboard (in the context of inputting data) or the screen (for output). Then anything you type will be printed out when you press [RETURN] just like Direct Printing in LocoScript. You will need to press [ALT]-J after each [RETURN] for a new line. To finish press [ALT]-Z.

PIP CON:=MYFILE.DOC[Z] is a useful command which prints the named file to the screen, much as CP/M's TYPE command does. The virtue of PIP used with its [Z[option is that if the document is a WordStar or Protext file it will be displayed correctly on the screen, whereas TYPE would list it out with strange line breaks and hieroglyphics at the end of each word.

PIP is also the first utility to try in passing files back and forward through the serial port. If you are linked to another computer through a serial interface you can use PIP files=AUX: to accept files or PIP AUX:=files to send them.

PIP at your fingertips

PIP is a program that you will find yourself using a lot, but it is tedious to have to remember to put PIP.COM on virtually every disc you have. If you want to run PIP and it isn't on your current disc you have to swap discs, find your CP/M disc, type PIP[RETURN] to get into Multiple Command Mode, put your old disc back in and then do the command you wanted all along.

A better way is to make sure your CP/M startup disc has a PROFILE.SUB file on it with these lines somewhere in it:

```
SETDEF M:,*
PIP M:=PIP.COM
```

This copies PIP.COM into the M drive so

that no matter what discs you have in A or B it is always available. The first line of the pair makes CP/M look in the M drive for any command you give so that,

if your current drive were A, if PIP.COM is not on the A disc CP/M will find it and run it from drive M without you having to remember to type M:PIP.

Quick filing

After copying files, arguably the most useful function of PIP is the ability to create short files without having to use an editor.

Say you want to create a Submit file that automatically loads your BASIC program GERBILS.BAS. you just type PIP PROFILE.SUB=CON:
The cursor moves the next line and you can type in BASIC GERBILS[RETURN]. Finish the operation with an [ALT]-Z. Then assuming you have SUBMIT.COM,BASIC.COM and the CP/M.EMS file on your disc the machine will automatically load BASIC and load GERBILS.BAS each time you start up using the disc.
There are problems in that you can't edit any mistakes that you make while typing - you can't even use the delete key. In fact it is easier to give up and start again if you make a mistake. Also, you must press [RETURN] then [ALT]-J for each new line - [RETURN] moves

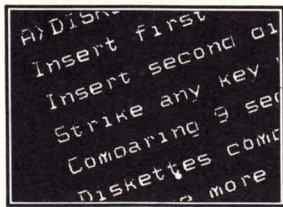the cursor back to column 1 and [ALT-J] to the next line.
Perhaps the most important advantage of creating files using PIP is that you can put 'Escape Codes' directly into a file. Escape Codes are the instructions that you can give the computer to do something special to the screen or printer.
The most useful example is the simple clear screen routine. You can clear the screen by typing directly at the keyboard [EXIT]E[EXIT]H. To make life easier you can create a file that has this written in it. Type PIP CLEARSCR=CON: so that anything you write on the screen will be written to a file called CLEARSCR. Type in [EXIT]E[EXIT]H and press [ALT]-Z to leave PIP. Any time you do TYPE CLEARSCR now the screen will be cleared.
All the 'Escape sequences' that the PCW recognises are listed on page 139 of the PCW CP/M manual.

TIP-OFFS

More tips than a colliery's back yard

Multiple Printer copies from LocoScript

It seems despite all of its fine qualities, LocoScript lacks a couple of fundamental functions. One in particular is the (apparent) non-existence of a multiple-copy print command.

As an example; you have just prepared a newsletter (on LocoScript of course) for your Local Chapter of the "I've Got Nowhere To Ride My Horsey" Association. The newsletter is a single page long, and some sixty copies are required.

At this point, it would seem that you have to bring up the DISC MANAGEMENT menu, and repeatedly select the relevant document, press [P] for printer, press [ENTER], wait for the printer action to finish, and repeat the sequence until sufficient copies have been created.

The alternative is to use the available functions of LocoScript to create a document consisting of many copies of the original. Understand that???

Practical experience is the best teacher, so boot up LocoScript on your trusty PCW, and select the document of which you wish to have multiple copies.

1. Press [F3] (copy) and copy the document into one of the groups in DRIVE M:. If you are unsure of this step consult page 43 of the LocoScript manual.
2. With the copy completed move the cursor onto the (copied) document in DRIVE M:.
3. Press [E] for EDIT followed by [ENTER].
4. Once in the EDIT mode, move to the end of the document ([SHIFT] & [DOC]), and press [ALT] & [RETURN]. This will force the cursor to the next page.
5. Press [EXIT] and select "SAVE AND CONTINUE", followed by [ENTER].
6. Now ensure the cursor is either at the TOP or BOTTOM of the document (it doesn't matter which).
7. Press [F7] and select "INSERT TEXT" followed by [ENTER].
8. When the disc manager comes up, the cursor should be on the current document. ENSURE THAT IT IS, AND PRESS [ENTER] TWICE. The document selected from the disc manager will be copied into the one being edited. As the edited and inserted documents are one and the same, the copy in memory now contains two of the original.

On the first pass you will have created TWO copies. Now repeat steps 5,6,7 and 8. The document's size is doubled to FOUR copies.

Repeating the sequence from step 5 onwards will keep on doubling the number of copies. That sequence is 1,2,4,8,16,32,64,128 etc. copies. I would suggest that 64 single page copies will see you run out of memory.

Having made the multiple copies, all that remains is that you [EXIT] to the disc manager in the normal manner, set the printer for continuous paper, and print out the new file while you have a cup of coffee.

Whilst the above may not be the perfect solution, it is infinitely better than printing a multitude of copies one at a time.

In closing, you do not have to use DRIVE M: to carry out this action. However, there are some benefits of speed to be had until such time as the computer's memory starts to fill up. Thirty-two pages are quite practical, after which your printer may want a rest!
Extracted from an article by Pierre A. Du Parté in an Illawarra Amstrad User Club Newsletter.

BASIC CORNER

Basically faster

Some tips to make your BASIC programs run faster:

1. When using variables, use integers whenever possible, especially as loop counters. For example,
```
10 for i=1 to 10000:next i
```
takes 9.0 seconds, but
```
10 for i%=1 to 10000:next i%
```
takes 6.8 seconds.
2. Use variables to store all numbers, even when the number doesn't change. For example,
```
10 for i%=1 to 10000:a= 3.14159265*3.14159265:next i%
```
takes only 3.7 seconds.

3. Write the program so that the most commonly used variables in the running of the program are set up first.

4. Put the most commonly called subroutines at the top of the program text, with the lowest line numbers.

5. Avoid writing to the disc or the screen as far as possible. If you want to store intermediate results, use arrays.
If your program is still very slow in running, you could think of buying a 'compiler' such as CBASIC or ZBASIC. These are similar to the PCW's Mallard BASIC but run as a compiled language rather than an interpreted language - instead of each being turned into machine code (interpreted) as the program runs, as in BASIC, the whole program is turned into code at the beginning (compiled) making the program run much faster. Some of the Mallard BASIC commands (in particular the Jetsam commands) won't work with these alternative systems since they weren't originally designed for PCW use.
L.K. Waymont

Do not collect $200

When you're working on a BASIC program you can use GOSUB or GOTO commands directly, not just as part of a listing. This means you can test subroutines before incorporation within a program just by entering GOSUB 3000, or whatever the line number is, directly after the 'OK' prompt. You can continue a program which has stopped somewhere by entering GOTO 190 or some other suitable point to recommence.

A direct GOSUB like this will return control to the keyboard as soon as a RETURN is encountered in the program, but a program entered using GOTO runs until it stops.
M.G. Surl

Key sera sera

Frequent users of BASIC may find it helpful to use the SETKEYS facility to assign to the function keys strings such as LIST, RUN [RETURN], EDIT, and SAVE ". For the last command remember you don't need the closing quote on the filename before hitting

return. You put a quotation mark into a setkeys file by typing ^".
M.G. Surl

Freak OUT

An unusual effect can be had in a BASIC listing by the command OUT 246, followed by a number between 0 and 255 - this changes the vertical origin of the screen. You can return to normal by the command OUT 246,255. Another interesting effect comes from entering OUT 245, and a number between 0 and 255, and you can return to

normal by typing OUT 245,91 (don't worry if you can't see this last line as you type it - it's going in!).

You can use these to make a fake 'malfunction' in your listings to deter the curious from finding out what happens when they press this or that key!
James Roskell

Black OUT

If your BASIC listing involves building up complex screen layouts you can make things look much slicker by putting the command OUT 248,8

before the procedure and OUT 248,7 after it has finished. This blanks out the screen while the PRINT statements run and turns it back afterwards, presenting the completed picture apparently instantly.

What these commands do is effectively turn the ink colour to black and then back to green again. The user has to look at a blank screen for a second or two, but the effect is quite neat and professional looking.
Andrew Porter

AtLast - more space

I have only recently started messing around with the CP/M Plus facilities of my PCW8256. I discovered that I could edit the file PROFILE.ENG renamed as PROFILE.SUB to also include the transfer of the ED.COM and DISCKIT.COM commands to the M: drive when auto-booting with CP/M. Doing this helped me when I was learning to use my newly bought At Last Database Manager software package.

The new PROFILE.SUB now reads:

```
 1:  setdef m:,* [order = (sub, com,) temporary = m:]
 2:  pip
 3:  <m:basic.com[o]
 4:  <m:dir.com[o]
 5:  <m:erase.com[o]
 6:  <m:paper.com[o]
 7:  <m:pip.com[o]
 8:  <m:rename.com[o]
 9:  <m:setkeys.com[o]
10:  <m:show.com[o]
11:  <m:submit.com[o]
12:  <m:type.com[o]
13:  <
14:  pip m:=ed.com[o]
15:  pip m:=disckit.com[o]
```

With a PCW8256, the At Last package will only work well if the command and other necessary files lie on the memory disc and your database data disc in the A: drive. The nasty catch is that the M: drive in a PCW8256 is not large enough to hold both the DBDEF and DBUSE files. The following provides a reasonable solution by taking the suggestions in the At Last manual one step further.

I examined the ATLAST.SUB file which is part of the At Last package, then edited and renamed copies of it to suit myself. Firstly, I created my own SUB file for just DBUSE files, then I realised I could do the same for the DBDEF files. The first thing to do, as always, is to copy the master discs with DISCKIT on to a spare disc. (This is where my "new"

PROFILE.SUB came in handy with DISCKIT already in the M: drive.)

The master was then also copied onto the reverse side of the master back-up disc. On the second copy of the master disc I deleted all the "example" files, namely the ones for the Club and Cashbook databases. The files remaining on this disc should then be ATLAST.SUB, ATLAST.KEY, DBDEF.000, DBDEF.001, DBDEF.COM, DBUSE.000, DBUSE.001, DBUSE.002, DBUSE.003 and DBUSE.com. The next thing to do is to copy (using PIP in drive M:) the ATLAST.SUB file as ATLASTA.SUB (or any other name you want). The ATLASTA.SUB file is now edited (again the "new" PROFILE.SUB came in handy with ED.COM already in the M: drive) to read as follows:

```
 1:  era m:basic.com      {include here all the
 2:  era m:dir.com        {
 3:  era m:disckit.com    {files you have in your
 4:  era m:ed.com         {
 5:  era m:rename.com     {PROFILE.SUB except
 6:  era m:show.com       {
 7:  era m:type.com       {ERASE.COM
 8:  era m:dbdef.com
 9:  era m:dbdef.000
10:  era m:dbdef.001
11:  era m:erase.com
12:  pip m:=a:dbuse.*
13:  setkeys a:atlast.key
14:  paper a4             (note: set to whatever
                          paper you may want)
15:  m:
16:  dbuse $1
```

This ATLAST.SUB file will then erase all the unwanted files in the M: drive, will leave the PAPER.COM, PIP.COM, SETKEYS.COM and SUBMIT.COM files in the M: drive and transfer all the DBUSE files to the M: drive and auto boot DBUSE. It is most important that the ERASE.COM file is the last in the list of files to be erased from the M: drive, other-

wise any further "era" lines cannot be executed.

This procedure is beaut, but still left me with the problem of having to use my disc in the A: drive for both DBDEF and data. I then thought, why not create another SUB file for DBDEF. I named it ATLASTDF.SUB and it reads as follows (again the "erase" file is the last to be listed):

```
 1:   era m:basic.com        {include here all the
 2:   era m:dir.com          {
 3:   era m:disckit.com      {files you have in your
 4:   era m:ed.com           {
 5:   era m:rename.com       {PROFILE.SUB except
 6:   era m:show.com         {
 7:   era m:type.com         {ERASE.COM
 8:   era m:dbuse.com
 9:   era m:dbuse.000
10:   era m:dbuse.001
11:   era m:dbuse.002
12:   era m:dbuse.003
13:   era m:erase.com
14:   pip m:=a:dbdef.*
15:   setkeys a:atlast.key
16:   paper a4               (note: set to whatever paper
                             you may want)
17:   m:
18:   dbdef $1
```

Note: each time a " *.SUB" file is edited by ED.COM, a copy of the original "*.SUB" file is created as a "*.BAK" file. These can be deleted when you are satisfied that you have your *.SUB files correct.

When using these SUB files to erase unwanted files from the M: drive, it may often happen that a file has already been erased. This causes no problem, the computer just automatically goes on to the next step within the SUB file. If you get an error message straight after the computer tries to PIP the DBDEF.COM or the DBUSE.COM to the M: drive while executing the SUB file, you should start again from scratch.

My whole procedure to use At Last is now as follows:

1. Load CP/M;
2. Remove CP/M disc and place the special database booting disc in the A: drive;
3. "SUBMIT" ATLASTDF.SUB (to autoboot DBDEF).
4. Remove special database booting disc and place an empty formatted data disc in the A: drive and start creating your new database;
5. When you have finished designing your definition files, index entries and forms, close the database to the stage where you have an empty screen with M> only showing, change the drive to the A: drive, and "SUBMIT" ATLASTA.SUB (to autoboot DBUSE). Enter data into your database and follow the computer's instructions.
6. If you find when entering data in a new database with DBUSE that either the definition file, index or form is not quite right, close the database to the stage where you have an empty screen with M> only showing, change drive to the A: drive, remove data disc, insert special database booting disc, and "SUBMIT" ATLASTDF.SUB (to autoboot DBDEF again)

then start redesigning the database files, index entries and forms. (As the At Last manual states, you have to be very careful with any redesigning as it may cause you to lose any or all data entries already entered (these are stored in "*.DAT files and *.IDX files); so do plenty of experimenting while you still have only a small number of entries in a new database.)

I have tried to explain this procedure in the simplest English possible so that new chums like me could understand it.

With this procedure, you get the full use of all the space on your data discs and save yourself the time and frustration of trying to remember or work out what is where etc. It has also saved me the cost of an expansion module, which I first thought was the only way out for my predicament.
Elisabeth L Light, Weston, ACT

PCW 8256/8512 & PC Software

GAMES for PCWs

Title	Price
Air Combat Emulator	39.99
'Colossus' Bridge + book	49.95
'Cyrus' Chess	44.50
Distractions	SOLD OUT
Gm. Gooch Cricket	SOLD OUT
Guild of Thieves	TEMP. O/S
Knight Orc	TEMP. O/S
Monsters of Murdac	TEMP. O/S
The Pawn	89.95
Silicon Dreams	59.99
Starglider	69.99
Strike Force Harrier	SOLD OUT
Tomahawk	57.95

SPREADSHEET

Scratchpad Plus	149.99

DATABASES

Cardbox	179.99
Condor One	199.99
Database Manager	99.95
Masterfile 8000	119.00

DESKTOP PUBLISHING

Fleet Street Editor	249.95
Font 'n' Graphics for FSE	69.99

Newsdesk International PLUS Light Pen pack

Newsdesk International is an extremely powerful program controlled normally by cursor keys but now comes with an Electric Studio Light Pen. Newsdesk International can be used as a stand-alone system with its own word processing editor or accept input from a word processor such as LocoScript, or even digitized input from the Electric Studio Video Digitizer.

This pack will give you the ability to mix the text files created on the word processor with the graphics created using Newsdesk International. Headlines and text can be sized easily according to your requirements.

NDI + E/S Light Pen	249.95

Fonts for Newsdesk	49.99
Snip Art for Newsdesk	49.99

GENNY - Genealogical Database on a 3'' disc (PCW)	$119.00

GRAPHICS & PRESENTATION

Title	Price
DR Draw	169.95
DR Graph	169.95
Font Module (to use with Elec. Studio light pen)	69.96
Rotate	89.99
Snip Art (to use with Elec. Studio light pen)	54.95
Super Type	89.99

DIGITIZER

Input from standard video camera to Newsdesk Int. or Fleet Street Editor.	349.95

UTILITIES

Catalogue (HiSoft)	39.95
C-Basic Compiler	169.95
Devpac 80	125.00
Disc Mate	94.95
The Knife (HiSoft)	39.95
The Torch (HiSoft)	39.95
Catalogue, Knife and Torch pack	99.99
Typing Crash Course	79.50

WORD PROCESSORS

LocoScript 2	72.95
NewWord2	275.00
Tasword 8000	79.95
Wordstar PCW	199.00

LANGUAGES

Pascal MT/+	149.00
Nevada Pilot	125.00

TASMAN PRODUCTS

Tasword 8000	79.95
Tas-spell 8000	56.00
Tasprint 8000	52.00
Tas-sign 8000	89.95

Please note that items on this page are Mail Order only. Prices may change without notice - check first!

GAMES for PCs

Title	Price
Air Combat Emulator	34.99
Beyond Wolfenstein	19.99
Casino Parlour Games	19.99
Crossfire	19.99
Factactics Trivia	19.99
Games Pack 1 (13 games)	29.99
Games Pack 2 (4 games)	29.99
Gamma Games	49.99
Guild of Thieves	TEMP. O/S
Jewels of Darkness	TEMP. O/S
Knight Orc	TEMP. O/S
The Pawn	TEMP. O/S
Saboteur II	29.99
Silicon Dreams	TEMP. O/S
Skyrunner	34.99
Starglider	TEMP. O/S
Tracker	TEMP. O/S

BUSINESS

ABC Business Pack	499.99
Ability Plus	450.00
Ability	225.00
Brainstorm	149.99
Cardbox	179.00
Condor 1 Jnr	199.99
Delta 4	399.99
In-House Accountant	339.99
Money Manager	139.99
Portex (with PVC binder)	299.99
Portex (with leather binder)	349.99
Scratchpad Plus	149.99

GEM SERIES

Business Library	149.99
GEM First Word Plus	299.99
Fonts and Drivers	139.99
GEM Font Editor	349.99
GEM Wordchart	349.99

GENNY - Genealogical Database on one 5.25'' disc (PC)	$119.00

Mail your orders to:

**THE AMSTRAD USER
1/245 Springvale Road
Glen Waverley, Vic 3150**

**Phone your firm orders to:
(03) 233 9661**

Bankcard, Mastercard or Visa accepted.

Available from The Amstrad User while stocks last

CONFIG.SYS explained and more Public Domain games on offer

from Chris Collins

Welcome once again to Compatibles Corner, and welcome to the New Year. It appears that we have made it into the new year without blowing up our little planet, so maybe things in general aren't too bad. However, this is not meant to be a discussion on the status and conditions of our world, but an article to help you get the best out of your PC1512s and 1640s. With this in mind, I suppose we'd better get on with it. But first, a couple of notices that should interest most of you.

It appears that the gremlin got into my games discs. Games Disc 4 does have problems. It appears that my master copy that I made went really screwy, and subsequent copies of it were causing people problems. Any faulty copies that were returned to me have been recopied off a new master disc and appear to give no trouble at all. The things to look for are as follows:

1. Check the disc label for space free. If it is equal to 0 bytes free, chances are your copy of the disc is faulty, and

2. When you try to unarchive the files, DOS reports a SECTOR NOT FOUND error. If your disc is doing this, please return it to me and I will forward you a new copy of the disc.

My apologies to all concerned. If you do get one of my discs that doesn't work, please feel free to return it to me to check. If it is indeed faulty, I will replace it without any qualms.

Whilst I am talking about discs, I must make mention of Fas-Type. It appears that when I first checked the program on my clone, it functioned perfectly. However, on upgrading my machine, I've since found that the help screens will not show up. This also happens on the PC1512. To counter this problem a program was supplied on the disc that would allow you to print out the help screens and correct the problem. This was mentioned in a README.1ST file on

the disc. It appears that a lot of people didn't.

Unfortunately I don't really have time to answer all the letters that you write to me, I do attempt, where possible to answer them. If you must have an answer to something, please give me a phone number to contact you (preferably during business hours). However, whenever you receive any discs from anyone, you should always check for any README files on the disc. These will usually give you some last minute information that you might need to know.

This month I am pleased to announce that the prices of any future diskettes will be lower than at present. Due to the increased demand for diskettes, I have been able to purchase supplies slightly cheaper so from here on, the first diskette will still cost $10, and any others ordered at the same time will cost you only a further $6. This means that an order of 5 diskettes will cost $34 ($10 + (4 X $6)). This will apply to all purchases from this month. Think of it as a late Christmas present.

The first diskette that I want to look at this month is Hangman. This is one of the all time favourite word games that children play. It has been converted to computer, and is presented for your pleasure. There are two versions of this game on the diskette. The first plays only in English, the second in both English and Spanish. The two versions are both very similar in structure and contain 4 further versions of each game.

The first is what they call BASIC Hangman. This is the game as we all know it, where one is meant to guess a word and the program fills in the letters. The next type is the EDUCATIONAL version. In this game, a definition of the word is shown on screen to help you learn. Of course, the words are also

harder. This is very good to help children learn and increase their vocabulary. LITERATE is the next version you can play. This game revolves around phrases. The author's name is provided, and you must guess the phrase that has been spoken by this person. This is very difficult, and often beats me. The last type is the PERSONALISED game. This can have two or more players, and the players enter the phrases. This is an incredible version of Hangman that I had never previously encountered. I found it most satisfying.

Whilst all these versions are available, there are also two different skill levels. This adds another twist to the game. To make things even more adjustable, you can play with a timer (or without). If enabled the computer will make necessary sounds to let you know what is happening. This can be disabled if you wish to play a quiet game late at night. The computer will also keep track of the hanging of your man. This is done in colour, and is quite a reasonable rendition of the gallows and man. At any time in the game, F1 will get you a new word or phrase to try, and F10 will return you to the main menu.

The SPANISH game is very similar except as follows. The choice of games is limited to WORDS, PHRASES and PERSONALISED. The first two can be further subdivided. You can have an ENGLISH WORD or PHRASE to work out, with SPANISH clues, or a SPANISH WORD or PHRASE with ENGLISH clues.

Both versions of the game are written by NORLAND SOFTWARE, and are available on the one diskette called HANGMAN.

The next program we will look at is called THE TRIVIA MACHINE. This program occupies two diskettes. Diskette 1 is the program files, and diskette 2 contains the question files. The Trivia machine is a multiple choice game about trivia (hence it's name). Upon loading the program proper, you will be presented with a configuration menu. This will allow you to setup the program as you wish. This menu asks if you require sounds (say yes), what your responses will be to the questions (use 1,2,3 & 4) and which drives you will place the program and question files on. This will then create a small data file so that you won't

have to configure the program each time you play it.

Into the game proper, and we are presented with the title screen. This looks rather good in colour, and asks whether you would like instructions or not. After this you will be asked which category you would like to play. The question file loads, and you are asked which level you wish to play at. After this we get into the questions proper. These are all of the multiple choice type, and require that you press an answer key before a timer runs out. That sounds simple enough until you realise that if you get an answer wrong, the timer deducts 20, beeps at you and keeps counting. The only basic problem that I had with the game is that it uses American questions, so it can be difficult.

Good fun, but I'm afraid that I don't quite know my trivia well enough so I didn't score too well in any category. As always, if you require any of the diskettes, please send your cheques or money orders to me, C/- The Amstrad User.

The command that I would like to look at this month is a small file called CONFIG.SYS. A config.sys file is a list of commands that set up your system. Each time you start your system, DOS searches the root directory or it's default drive for this file. If the file is found, DOS reads the file and carries out the commands it finds. If the file is not found, DOS will assign default values for the configuration commands.

When setting up your config.sys file, please remember that the changes that are available include the following options:

* Set extended checking of CONTROL-BREAK (BREAK)
* Specify the number of disc buffers for DOS to use (BUFFERS)
* Specify the country whose date and time format you require (COUNTRY)
* Install device drivers (DEVICE)
* Specify the number of files that can be open at one time (FILES), and
* Specify the maximum drive letter you may access (LASTDRIVE)

I will go through the options separately and explain them as best I can. The BREAK command is set to either ON or OFF as follows; BREAK=[ON/OFF]

This instructs DOS to check for CONTROL-BREAK at any time it is requested to perform any functions. The default

here is BREAK=OFF, which means that DOS will only check for the break signal during certain operations. However, if your program gets into a loop and doesn't use a lot of DOS functions, sometimes the only way out is a reboot.

The BUFFERS command allows you to set up the number of disc buffers that DOS will allocate in memory. A disc buffer is a block of memory that DOS uses to store data being read from or written to a disc. If DOS is then required to read from the disc, it will first check the disc buffers before going to the disc if it is not available. This is much faster than always reading from the disc. The command is set as follows; BUFFER=x where x is a number between 1 and 99. The default value for a PC1512 is 2, which isn't really enough. A better setting in most cases is 25. This is about the upper logical limit. Once the buffers get too big, DOS will spend too much time checking the buffers to make the system slower than without buffers.

The next option on our list is COUNTRY. This is used to tell DOS what format you want to use for the date and time commands. It also controls the currency symbols. The command is as follows; COUNTRY=xxx where xxx is a three digit international country code for the telephone system. The default is 001 which is the U.S. code. The countries available to you number eighteen, but the three that you probably need to know are; Australia (061), United Kingdom (044) and the United States (001). Keyboards can also be set.

Device drivers are the next thing I would like to look at. A device driver is a program that installs into your system to let you fix a shortcoming that DOS has. For example, my config.sys file has the line DEVICE=HARDRIVE.SYS. This is a small program that allows me to split my 30mb hard disc into two parts. DOS will not allow me to do this, but with the device driver installed it will recognise both my C: and D: drives.

The device drivers that are available to you on your DOS disc include the following; ANSI.SYS, DRIVER.SYS and RAMDRIVE.SYS.

Your system should always call up ANSI.SYS when it boots, as this allows access to the extended screen and keyboard control. The command line is as follows;

DEVICE=[d:][\path]ANSI.SYS. This allows you to keep the file out of your root directory and keep it neat.

DRIVER.SYS is used to set up other logical drives on your system (apart from the physical ones). The proper command line for this is very confusing so I will attempt to shorten it a bit, but here goes;

DEVICE=[d:][\path]DRIVER.SYS /D:ddd [/T:ttt] [/S:ss] [/H:hh] [/F:ff]

As always, switches inside a square bracket are optional, but they allow for a great deal more variety. This command line needs to be broken down more for it to make sense. So lets get to it. /D:ddd specifies the physical disc drive and is referred to as Drive A on the DOS line. Drive 1 is the next physical disc drive (B:). The first physical fixed disc drive must be referred to as drive 128.

/T:ttt specifies the number of tracks per side. This can be between 1 and 999. The default setting is 80 tracks.

/S:ss specifies the number of sectors per track. This can be a number between 1 and 99. The default is 9 sectors per track.

/H:hh is the maximum number of heads in the drive. The default is 2, and the number can be between 1 and 99.

/F:f is the device type. The default is type 2, but can be any of the following;
0 160/180kb
0 320/360kb
1 1.2mb
2 720kb

As you can see, DRIVER.SYS will allow you to connect up almost any type of disc drive to your system.

RAMDRIVE.SYS is used to setup a virtual disc in memory. This functions as a very fast disc drive in your system. This can be useful to try and speed up programs that read a lot of overlay files from disc. You can install more than 1 virtual disc, and refer to each by it's drive letter. RAMDRIVE.SYS has a command line as follows;

DEVICE=[d:][/path]RAMDRIVE.SYS [bbb] [sss] [ddd]

[bbb] refers to the size of the RAMDRIVE that you wish to create. 64kb is the default, the range of values is between 1kb and the memory capacity of the computer. If the installation of a RAMDRIVE leaves less than the 64kb of free memory, DOS will adjust the size of the RAMDRIVE downward to leave 64kb free memory.

[sss] is the sector size that you require (in bytes). Allowable sizes are 128, 256 and 512. If a sector size is omitted, DOS will use 128 bytes.

[ddd] refers to the number of directory entries available on the RAMDRIVE. The default value is 64, but it can be anywhere between 2 and 512.

FILES is the next command line option available to you. This allows you to specify the maximum number of files that DOS can have open concurrently. The command is FILES=x where x is a number between 8 and 255. The default value for this is 8. I have found over time that FILES=25 suits almost all applications.

The last command that we will look at is LASTDRIVE. This is set as follows; LASTDRIVE=x where x is an alphabetic character. The default at any time is E:. If this number is set at less than the physical number of drives attached to your system, the command is ignored.

The other two commands that are available include SHELL and FCBS, but I've never found a use for either of these and accordingly, I can't see the point in bothering you with them. If you need to use them, it's time that you read your DOS manual.

A sample config.sys file could be as follows;

    DEVICE=ANSI.SYS
    BREAK=ON
    FILES=25
    BUFFERS=25
    COUNTRY=061

This would be a quite satisfactory file for most of you to use. To create such a file you could either use a word-processor that creates ASCII files, or simply use the following command; COPY CON: CONFIG.SYS.

This will then copy what you type on your keyboard, into a file called CONFIG.SYS. Simply type in the lines above, pressing ENTER at the end of each and make sure that you don't make any mistakes. When you have typed in all the command lines that you require, press the F6 key and ENTER again and this will close the file and copy it to the disc. When you press F6, you will see ^Z appear on your screen. This is CONTROL-Z and simply tells DOS that you are finished. Reboot your system for the changes to take effect, and you are in business.

Unfortunately, space dictates that I don't get into this too much further. I hope that you've enjoyed the column this month. It did get a bit deep but there is no other way to go when you need to talk about some of the options that are available for you to configure your system. As you can see, your PC1512 or PC1640 can be customized to what you require.

I hope you've learnt a bit more about your computer this month. Next month, we will be looking at AUTOEXEC.BAT. This will allow us to set up our systems even more to what we like. Until then, BBFN. (Bye Bye for now).

> PC HELP

Where each tip is worth at least the cost of the magazine! We're still waiting for a flood of tips - where are they?

Happier Reset

Undocumented ways of guaranteeing a system reset include; working for three hours on a spreadsheet without periodically saving it to disc; letting a Basic2 Mandlebrot curve program run all night until it's almost ready for you to photograph; and putting the PC Help editor's document disc in the drive and asking for a spell check.

Happily however, there are less traumatic means to achieve the same result. Apart from pressing [ALT][CTRL][DEL], you can also use [ALT][CTRL][LEFT SHIFT KEY][CAPS LOCK].

Word Junior Help

If you are having problems setting up work discs for use with Microsoft's Word Junior, these hints from Microsoft themselves may provide the answers.
1. The instruction in 'SETUP' to copy DOS onto the program disc should not in fact be there, as it was not originally intended for DOS to reside or to be copied on to the Word Junior disc. Unfortunately there is simply insufficient memory for this to function correctly.

This came about due to an error when the wrong version of SETUP was shipped. Future releases of the product will not include an option to copy DOS.
2. The fact that DOS is not resident will not cause any problems in the operation of the product, with the sole exception of using the LIBRARY RUN command as it requires COMMAND.COM. You can however copy this onto your program disc through the following method:
Place MS-DOS disc in Drive A

Place Word Junior disc in Drive B
Enter COPY COMMAND.COM B:
If you chose not to carry this operation out you will simply be prompted to insert a disk containing COMMAND.COM when you choose LIBRARY RUN.
3. When you start the Word program from floppy disc, it needs to create a temporary file on the program disc, therefore you need to ensure that there is no write protect tab on the disc otherwise you will get the message, 'Enter Y to retry access to MWxxxxxx.tmp'.

Top Form

By creating a file called FF.BAT with RPED, it is possible to send a form feed command to the printer at the A-prompt.
FF.BAT should contain the following lines:

```
echo off
echo [CTRL][L] >PRN
```

The [CTRL][L] (hold down [CTRL] and press [L]) will produce a biological symbol, and entering FF at the A-prompt will send a form feed command to the printer.

This is a good example of example of using ECHO to display control characters which can't be generated directly from the keyboard; and of redirection (the >PRN bit) to send what would normally be screen output to another device.

The ECHO technique is a much neater way of sending escape sequences to ANSI.SYS (to change screen colours and so on) than the tricky PROMPT method - as long as you can get the control characters into the ECHO statement in the first place. Some editors and word processors are very fussy about putting 'non-displayable' characters into text, and most are very awkward about getting the ESCAPE character (decimal 27) into a document. (See next hint).

Great Escape

So, how do you get the Escape character (ASCII 27 or HEX1B) into a batch file whilst using RPED? Both EDLIN (with Ctrl-V plus left square bracket) and DEBUG can handle [Esc], after which RPED shows it as a left pointing arrow. When it comes to embedding special

characters in text documents, you are very much in the hands of your editor - either it lets you or it doesn't! RPED doesn't as it always interprets the [Esc] character as a 'write the file to disc' command.

For the relatively small number of [Esc] characters likely to be needed, a file can be created with RPED, marking the places where [Esc] is required with some other little-used character (a tild [~] for example). Then use DEBUG to replace them with ASCII 27 before use.

Using DEBUG, first enter DEBUG filename at the A-prompt. This loads the file into memory ready for you to get to work. If you now enter d100 at the DEBUG prompt it will dump the first 128 bytes of your file to the screen. Use the instruction s,100,ffff,7E to find the address of any occurrences of the [~] character - this instruction means 'search from address 100 to address ffff (well beyond the end of the file) for any occurrences of hex 7E'. To change these to, say, the Escape character, you need to enter E address. This displays the contents of that address - 7E - followed by a stop. Enter 1B (Hex for Escape) and the change is made. When you have finished enter w to write the changes to disc, and q to quit DEBUG.

1-2-3 crash!

Lotus 1-2-3 sometimes gives a "Fatal Stack Error" or similar message. This usually happens if you have done a couple of hours work without saving, and then try to save it to disc! As the message implies the error is fatal and there is nothing you can do to retrieve the work lost since your last save.

This message is not usually caused by bugs in the software being run: it occurs because Microsoft decided to save a bit of memory by cutting down on the stack space reserved by MS-DOS version 3.2, as compared to earlier version. The solution is to add an additional line to your CONFIG.SYS file which says STACKS=64,128. This undocumented command will restore the stack space to the value used in earlier versions of DOS. Even with this increased memory you can still eventually generate the error on some PC1512's by turning the volume control down to zero and leaning a book on the keyboard so that it depresses a key.

Keeping tabs on Tabbies

A Pedigree Database

from Lloyd Cherry

At some time or other just about everyone has dealings of some sort with pedigrees. Be it fancy breeds of poultry, blood stock, homing pigeons or in my case pedigreed cats which I have bred for many years. There is always a need to store the ancestral details in a manner which allows easy and reliable access. My answer to this problem is PEDIGREE à la 464.

In conjunction with the DDI-1 disc drive, my Amstrad CPC464 provides me with an accurate storage system for my pedigrees. It also gives me a fast print-out in an acceptable format to either three or four generations.

All instructions are incorporated in the program and appear as prompts where necessary.

Lines 70 to 110 tailor-make the program to your own individual needs.

The print-out has been designed for an 80 column format and the printer codes are for a Star Gemini-10X.

The "Combine Two Procedures" routine is especially useful in projecting matings and previewing the resultant pedigree. It also removes the necessity of re-entering the whole ancestral details when both parents are already on file. All offspring is retained as future breeding stock.

I hope you find this little program as useful as I do.

```
10 '
20 ' PEDIGREE PROGRAM
22 ' The Amstrad User, Feb(88).
40 '
50 ' INITIATE DETAILS OF BREEDING
60 ' PLEASE INSERT YOUR OWN RELEVANT INFORMATION
70 STUDNAME$="COLISSA CATTERY"
80 BREEDER$="L.J "
90 ADDRESS$="McKENNER ROAD, LESLIE VALE, TAS. 7100"
100 BREED$="CAT"
```

```
110 YOUNG$="KITTEN"
120 MODE 2
130 DIM A$(31)
140 DIM B$(31)
150 DIM S$(31)
160 DIM D$(31)
170 '
180 ' MENU ROUTINE
190 '
200 CLS
210 PRINT"THE FOLLOWING OPTIONS ARE AVAILABLE TO YOU"
220 PRINT
230 PRINT:PRINT TAB(10),"E - ENTER NEW PEDIGREE DETAILS"
240 PRINT:PRINT TAB(10),"L - CHECK LISTING OF PEDIGREE"
250 PRINT:PRINT TAB(10),"A - ALTER DETAILS"
260 PRINT:PRINT TAB(10),"D - DISPLAY PEDIGREE ON SCREEN"

270 PRINT:PRINT TAB(10),"P - PRINT COPY OF PEDIGREE"
280 PRINT:PRINT TAB(10),"R - RETRIEVE PEDIGREE FROM FILE"
290 PRINT:PRINT TAB(10),"F - FILE PEDIGREE TO MEMORY"
300 PRINT:PRINT TAB(10),"C - COMBINE TWO PEDIGREES"
310 PRINT:PRINT TAB(10),"T - TERMINATE PROGRAM"
320 PRINT: PRINT:PRINT"PLEASE INDICATE YOUR CHOICE"
330 SOUND 1,90
340 FOR A=1 TO 10:A$=INKEY$:NEXT
350 O$=INKEY$: IF O$=""THEN 350
360 IF UPPER$(O$)="E"THEN 460
370 IF UPPER$(O$)="L"THEN 630
380 IF UPPER$(O$)="A"THEN 790
390 IF UPPER$(O$)="D"THEN 1530
400 IF UPPER$(O$)="P"THEN 1530
410 IF UPPER$(O$)="R"THEN 3550
420 IF UPPER$(O$)="F"THEN 3690
430 IF UPPER$(O$)="C"THEN 3820
440 IF UPPER$(O$)="T"THEN END
450 GOTO 170
460 '
470 ' NEW PEDIGREE ROUTINE
480 '
490 CLS
500 PRINT"ENTER 31 NAMES STARTING WITH THE ";BREED$
510 PRINT"FOLLOWED BY PARENTS, GRAND PARENTS"
520 PRINT"GREAT GRAND PARENTS,AND GREAT GREAT GRAND PARENT
S"
```

```
530 PRINT
540 PRINT"NOW INPUT FROM 1: - 31:
550 PRINT:PRINT
560 FOR LOOP =1 TO 31
570 PRINT LOOP;
580 SOUND 1,90
590 INPUT A$(LOOP)
600 IF LEN (A$(LOOP)) > 41 THEN PRINT CHR$(7): PRINT"NAME
IS";LEN(A$(LOOP))-41;"LETTERS TOO LONG..PLEASE ENTER AGAIN
":PRINT:GOTO 570
610 NEXT LOOP
620 GOTO 170
630 '
640 ' DISPLAY OF ENTRIES
650 '
660 CLS
670 D=0
680 PRINT:PRINT"THE FOLLOWING ";BREED$;"S HAVE BEEN ENTERE
D INTO MEMORY"
690 PRINT:PRINT
700  FOR A=1 TO 31
710  PRINT TAB(15)A;TAB(20)A$(A)
720 NEXT
730 IF D>0 THEN 850
740 PRINT:PRINT"PRESS ANY KEY TO RETURN TO MENU"
750 SOUND 1,90
760 FOR A=1 TO 10:A$=INKEY$:NEXT
770 A$=INKEY$:IF A$=""THEN 770
780 GOTO 170
790 '
800  ' ALTERATION ROUTINE
810  '
820 D=1
830 CLS
840 GOTO 680
850 IF D=2 THEN 102
860  PRINT:PRINT: PRINT"WHIC  ENTRY W  LD YO  LIKE TO ALTE
R?"
870 SOUND 1,90
880 PRINT
890  INPUT B
900 CLS
910 PRINT TAB(3);A$(B)
920 PRINT
930  PRINT"PLEASE ENTER THE CORRECT NAME"
940 PRINT
950 SOUND 1,90
960  INPUT B$(B)
970 IF LEN (B$(B)) > 41 THEN PRINT CHR$(7): PRINT"NAME IS"
;LEN(B$(B))-41;"LETTERS TOO LONG..PLEASE ENTER AGAIN":PRIN
T:GOTO 930
980 PRINT
990  LET A$(B)=B$(B)
1000  PRINT:PRINT
1010 D=2:CLS:GOTO 680
1020  PRINT:PRINT:PRINT"ARE THERE ANY MORE ALTERATIONS? <Y
/N>"
1030 SOUND 1,90
1040 FOR A=1 TO 10:A$=INKEY$:NEXT
1050 A$=INKEY$:IF A$=""THEN 1050
1060 IF UPPER$(A$)="Y"THEN 860
1070 IF UPPER$(A$)="N"THEN 170
1080 GOTO 1020
1090 '
1100 ' PRINT ROUTINE (3 GENERATIONS)
1110 '
1120 CLS
1130 Z=0
1140 PRINT #Z.TAB(39)A$(8)
1150 PRINT #Z,TAB(38)"/"
1160 PRINT #Z,TAB(26)A$(4)
1170 PRINT #Z,TAB(25)"/";TAB(38)"\"
1180 PRINT #Z,TAB(24)"/";TAB(39)A$(9)
1190 PRINT #Z,TAB(13)A$(2)
1200 PRINT #Z,TAB(12)"/";TAB(24)"\";TAB(39)A$(10)
1210 PRINT #Z,TAB(11)"/";TAB(25)"\";TAB(38)"/"
1220 PRINT #Z,TAB(10)"/";TAB(26)A$(5)
1230 PRINT #Z,TAB(9)"/";TAB(38)"\"
1240 PRINT #Z,TAB(8)"/";TAB(39)A$(11)
1250 PRINT #Z,A$(1)
1260 PRINT #Z,TAB(8)"\";TAB(39)A$(12)
1270 PRINT #Z,TAB(9)"\";TAB(38)"/"
1280 PRINT #Z,TAB(10)"\";TAB(26)A$(6)
1290 PRINT #Z,TAB(11)"\";TAB(25)"/";TAB(38)"\"
1300 PRINT #Z,TAB(12)"\";TAB(24)"/";TAB(39)A$(13)
1310 PRINT #Z,TAB(13)A$(3)
1320 PRINT #Z,TAB(24)"\";TAB(39)A$(14)
1330 PRINT #Z,TAB(25)"\";TAB(38)"/"
1340 PRINT #Z,TAB(26)A$(7)
1350 PRINT #Z,TAB(38)"\"
1360 PRINT #Z,TAB(39)A$(15)
1370 IF Z=8 THEN 1510
1380 PRINT:PRINT"PRESS ANY KEY TO RETURN TO MENU"
1390 SOUND 1,90
1400 FOR A=1 TO 10:A$=INKEY$:NEXT
1410 A$=INKEY$:IF A$=""THEN 1410
1420 GOTO 170
1430 '
1440 ' PRINT COPY ROUTINE - 3 GENERATIONS
1450 '
1460 CLS:PRINT"COPY OF PEDIGREE NOW BEING PRINTED -  3 GE
NERATIONS"
1470 PRINT#8,CHR$(27);CHR$(65);CHR$(24);
1480 Z=8
1490 PRINT#8:PRINT#8
1500 GOTO 1140
1510 PRINT#8,CHR$(27);CHR$(50)
1520 GOTO 170
1530 '
```

```
1540 'GENERATION STATUS ROUTINE
1550 '
1560 CLS
1570 IF UPPER$(O$)="P"THEN GOSUB 1720
1580 CLS
1590 PRINT:PRINT:PRINT"DO YOU REQUIRE   3   GENERATIONS OR
     4
1600 SOUND 1,90
1610 PRINT:PRINT:PRINT
1620 FOR A=1 TO 10:A$=INKEY$:NEXT
1630 Q$=INKEY$:IF Q$=""THEN 1630
1640 IF Q$="3"THEN 1670
1650 IF Q$="4"THEN 1690
1660 GOTO 1570
1670 IF UPPER$(O$)="D"THEN 1090
1680 IF UPPER$(O$)="P"THEN 1430
1690 IF UPPER$(O$)="D"THEN 2950
1700 IF UPPER$(O$)="P"THEN 2320
1710 GOTO 1580
1720 '
1730 'MESSAGE ROUTINE FOR PEDIGREE PRINTOUT
1740 '
1750 PRINT#8,CHR$(27);CHR$(87);CHR$(1);CHR$(27);CHR$(69)

1760 PRINT#8,"            ";STUDNAME$
1770 PRINT#8.CHR$(27);CHR$(87):CHR$(0):CHR$(27):CHR$(70)
1780 PRINT#8
1790 CLS:PRINT:PRINT:PRINT
1800 PRINT"IS THIS PEDIGREE FOR :-
1810 PRINT:PRINT:PRINT"      S - STUD SERVICE COPY ONLY

1820 PRINT:PRINT:PRINT"      I - INFORMATION COPY ONLY

1830 PRINT:PRINT:PRINT"      X - SELLING COPY
1840 SOUND 1,90
1850 FOR A=1 TO 10:A$=INKEY$:NEXT
1860 Q$=INKEY$:IF Q$=""THEN 1860
1870 IF UPPER$(Q$)="S"THEN 1920
1880 IF UPPER$(Q$)="I"THEN 1960
1890 IF UPPER$(Q$)="X"THEN 2020
1900 GOTO 1790
1910 FOR A=1 TO 10:A$=INKEY$:NEXT
1920 PRINT#8:PRINT#8,"STUD SERVICE PEDIGREE OF:-     ";CH
R$(27);CHR$(69);CHR$(27);CHR$(45);CHR$(1);A$(1);CHR$(27);C
HR$(45);CHR$(0);CHR$(27);CHR$(70)
1930 PRINT#8
1940 PRINT#8,"OWNED BY:- ";BREEDER$;", ";ADDRESS$
1950 RETURN
1960 PRINT#8:PRINT#8,"COPY OF PEDIGREE OF:-    ";CHR$(27);C
HR$(69);CHR$(27);CHR$(45);CHR$(1);A$(1);CHR$(27);CHR$(45);
CHR$(0);CHR$(27);CHR$(70)
1970 PRINT#8,"FOR INFORMATION ONLY"
1980 CLS
1990 PRINT#8
2000 PRINT#8,"OWNED BY:- ";BREEDER$;", ";ADDRESS$
```

```
2010 RETURN
2020 CLS
2030 INPUT"WHAT IS THE COLOUR AND VARIETY";CV$
2040 SOUND 1,90
2050 PRINT:PRINT
2060 INPUT"WHAT IS THE BREED NUMBER";BN$
2070 SOUND 1,90
2080 PRINT:PRINT
2090 INPUT"WHAT WAS THE DATE OF BIRTH";DOB$
2100 SOUND 1,90
2110 PRINT:PRINT
2120 PRINT"WHAT IS THE SEX OF THE ";YOUNG$:INPUT SEX$
2130 SOUND 1,90
2140 PRINT:PRINT
2150 PRINT"WHO WAS THE ";YOUNG$;" SOLD TO":INPUT NO$
2160 SOUND 1,90
2170 PRINT:PRINT
2180 INPUT"WHAT WAS THE DATE OF SALE";DS$
2190 SOUND 1,90
2200 PRINT#8
2210 IF A$(1)<>(STUDNAME$+" "+YOUNG$)THEN 2240
2220 PRINT#8,"PEDIGREE OF "CHR$(27);CHR$(69);CHR$(27);CHR$
(45);CHR$(1);SPACE$(56);CHR$(27);CHR$(45);CHR$(0)
2230 PRINT #8:GOTO 2260
2240 PRINT#8,"PEDIGREE OF "CHR$(27);CHR$(69);CHR$(27);CHR$
(45);CHR$(1);A$(1);CHR$(27);CHR$(45);CHR$(0)
2250 PRINT#8
2260 PRINT#8,SPACE$(12);CHR$(27);CHR$(69);SEX$;" ";CV$;" "
;"(";BN$;")"; CHR$(27);CHR$(70);" born on ";DOB$
2270 PRINT#8
2280 PRINT#8,"BRED BY:- ";BREEDER$;", ";ADDRESS$
2290 PRINT#8
2300 PRINT#8,YOUNG$;" SOLD TO:- ";NO$;" on ";DS$
2310 RETURN
2320 '
2330 ' PRINT COPY ROUTINE (4 GENERATIONS)
2340 '

2350 CLS
2360 PRINT"COPY OF PEDIGREE NOW BEING PRINTED  -  4 GENERA
TIONS"
2370 PRINT#8:PRINT#8
2380 PRINT#8:PRINT#8
2390 PRINT #8,TAB(39)A$(16)
2400 PRINT #8,TAB(38)"/"
2410 PRINT #8,TAB(26)A$(8)
2420 PRINT #8,TAB(25)"/";TAB(38)"\"
2430 PRINT #8,TAB(24)"/";TAB(39)A$(17)
2440 PRINT #8,TAB(13)A$(4)
2450 PRINT #8,TAB(12)"/";TAB(24)"\";TAB(39)A$(18)
2460 PRINT #8,TAB(11)"/";TAB(25)"\";TAB(38)"/"
2470 PRINT #8,TAB(10)"/";TAB(26)A$(9)
2480 PRINT#8,CHR$(27);CHR$(69);
2490 PRINT#8,CHR$(27);CHR$(45);CHR$(1);
2500 PRINT#8,"SIRE";
```

```
2510 PRINT#8,CHR$(27);CHR$(70);CHR$(27);CHR$(45);CHR$(0);
2520 PRINT#8,"        /                            \"
2530 PRINT #8,TAB(8)"/";TAB(39)A$(19)
2540 PRINT #8,A$(2)
2550 PRINT #8,TAB(8)"\";TAB(39)A$(20)
2560 PRINT #8,TAB(9)"\";TAB(38)"/"
2570 PRINT #8,TAB(10)"\";TAB(26)A$(10)
2580 PRINT #8,TAB(11)"\";TAB(25)"/";TAB(38)"\"
2590 PRINT #8,TAB(12)"\";TAB(24)"/";TAB(39)A$(21)
2600 PRINT #8,TAB(13)A$(5)
2610 PRINT #8,TAB(24)"\";TAB(39)A$(22)
2620 PRINT #8,TAB(25)"\";TAB(38)"/"
2630 PRINT #8,TAB(26)A$(11)
2640 PRINT #8,TAB(38)"\"
2650 PRINT #8,TAB(39)A$(23)
2660 PRINT#8:PRINT#8:PRINT#8
2670 PRINT #8,TAB(39)A$(24)
2680 PRINT #8,TAB(38)"/"
2690 PRINT #8,TAB(26)A$(12)
2700 PRINT #8,TAB(25)"/";TAB(38)"\"
2710 PRINT #8,TAB(24)"/";TAB(39)A$(25)
2720 PRINT #8,TAB(13)A$(6)
2730 PRINT #8,TAB(12)"/";TAB(24)"\";TAB(39)A$(26)
2740 PRINT #8,TAB(11)"/";TAB(25)"\";TAB(38)"/"
2750 PRINT #8,TAB(10)"/";TAB(26)A$(13)
2760 PRINT#8,CHR$(27);CHR$(69);
2770 PRINT#8,CHR$(27);CHR$(45);CHR$(1);
2780 PRINT#8,"DAM";
2790 PRINT#8,CHR$(27);CHR$(70);CHR$(27);CHR$(45);CHR$(0);

2800 PRINT#8,"        /                            \"
2810 PRINT #8,TAB(8)"/";TAB(39)A$(27)
2820 PRINT #8,A$(3)
2830 PRINT #8,TAB(8)"\";TAB(39)A$(28)
2840 PRINT #8,TAB(9)"\";TAB(38)"/"
2850 PRINT #8,TAB(10)"\";TAB(26)A$(14)
2860 PRINT #8,TAB(11)"\";TAB(25)"/";TAB(38)"\"
2870 PRINT #8,TAB(12)"\";TAB(24)"/";TAB(39)A$(29)
2880 PRINT #8,TAB(13)A$(7)
2890 PRINT #8,TAB(24)"\";TAB(39)A$(30)
2900 PRINT #8,TAB(25)"\";TAB(38)"/"
2910 PRINT #8,TAB(26)A$(15)
2920 PRINT #8,TAB(38)"\"
2930 PRINT #8,TAB(39)A$(31)
2940 GOTO 170
2950 '
2960 ' DISPLAY PEDIGREE ROUTINE (4 GENERATIONS)
2970 '
2980 CLS
2990 PRINT TAB(39)A$(16)
3000 PRINT TAB(38)"/"
3010 PRINT TAB(26)A$(8)
3020 PRINT TAB(25)"/";TAB(38)"\"
3030 PRINT TAB(24)"/";TAB(39)A$(17)
3040 PRINT TAB(13)A$(4)
3050 PRINT TAB(12)"/";TAB(24)"\";TAB(39)A$(18)
3060 PRINT TAB(11)"/";TAB(25)"\";TAB(38)"/"
3070 PRINT TAB(10)"/";TAB(26)A$(9)
3080 PRINT TAB(9)"/";TAB(38)"\"
3090 PRINT TAB(8)"/";TAB(39)A$(19)
3100 PRINT A$(2)
3110 PRINT TAB(8)"\";TAB(39)A$(20)
3120 PRINT TAB(9)"\";TAB(38)"/"
3130 PRINT TAB(10)"\";TAB(26)A$(10)
3140 PRINT TAB(11)"\";TAB(25)"/";TAB(38)"\"
3150 PRINT TAB(12)"\";TAB(24)"/";TAB(39)A$(21)
3160 PRINT TAB(13)A$(5)
3170 PRINT TAB(24)"\";TAB(39)A$(22)
3180 PRINT TAB(25)"\";TAB(38)"/"
3190 PRINT TAB(26)A$(11)
3200 PRINT TAB(38)"\"
3210 PRINT TAB(39)A$(23)
3220 PRINT:PRINT"PRESS ANY KEY TO DISPLAY DAM'S SIDE"
3230 SOUND 1,90
3240 FOR A=1 TO 10:A$=INKEY$:NEXT
3250 A$=INKEY$:IF A$=""THEN 3250
3260 CLS
3270 PRINT TAB(39)A$(24)
3280 PRINT TAB(38)"/"
3290 PRINT TAB(26)A$(12)
3300 PRINT TAB(25)"/";TAB(38)"\"
3310 PRINT TAB(24)"/";TAB(39)A$(25)
3320 PRINT TAB(13)A$(6)
3330 PRINT TAB(12)"/";TAB(24)"\";TAB(39)A$(26)
3340 PRINT TAB(11)"/";TAB(25)"\";TAB(38)"/"
3350 PRINT TAB(10)"/";TAB(26)A$(13)
3360 PRINT TAB(9)"/";TAB(38)"\"
3370 PRINT TAB(8)"/";TAB(39)A$(27)
3380 PRINT A$(3)
3390 PRINT TAB(8)"\";TAB(39)A$(28)
3400 PRINT TAB(9)"\";TAB(38)"/"
3410 PRINT TAB(10)"\";TAB(26)A$(14)
3420 PRINT TAB(11)"\";TAB(25)"/";TAB(38)"\"
3430 PRINT TAB(12)"\";TAB(24)"/";TAB(39)A$(29)
3440 PRINT TAB(13)A$(7)
3450 PRINT TAB(24)"\";TAB(39)A$(30)
3460 PRINT TAB(25)"\";TAB(38)"/"
3470 PRINT TAB(26)A$(15)
3480 PRINT TAB(38)"\"
3490 PRINT TAB(39)A$(31)
3500 PRINT:PRINT"PRESS ANY KEY TO RETURN TO MENU"
3510 SOUND 1,90
3520 FOR A=1 TO 10:A$=INKEY$:NEXT
3530 A$=INKEY$:IF A$=""THEN 3530
3540 GOTO 170
3550 '
3560 ' RETRIEVE FROM MEMORY ROUTINE
3570 '
```

```
3580 CLS
3590 PRINT"RETRIEVE PEDIGREE FROM FILE"
3600 PRINT
3610 PRINT"HERE IS THE CATALOGUE OF PEDIGREES ON FILE":PRI
NT:CAT
3620 SOUND 1,90:INPUT"ENTER NAME REQUIRED";G$
3630 PRINT:PRINT
3640 SOUND 1,90:PRINT"INSERT PEDIGREE FILE THEN PRESS ANY
KEY"
3650 IF INKEY$=""THEN 3650
3660 OPENIN G$
3670 FOR X=1 TO 31:INPUT #9,A$(X):NEXT:CLOSEIN
3680 GOTO 170
3690 '
3700 ' FILE PEDIGREE TO MEMORY ROUTINE
3710 '
3720 CLS
3730 PRINT"FILE PEDIGREE TO MEMORY"
3740 PRINT:PRINT
3750 SOUND 1,90:INPUT"ENTER NAME";G$
3760 PRINT:PRINT
3770 SOUND 1,90:PRINT"INSERT PEDIGREE FILE THEN PRESS ANY
KEY"
3780 IF INKEY$=""THEN 3780
3790 OPENOUT G$
3800 FOR X=1 TO 31:WRITE #9,A$(X):NEXT:CLOSEOUT
3810 GOTO 170
3820 '
3930 ' COMBINE PEDIGREES ROUTINE
3840 '
3850 CLS
3860 PRINT"COMBINE TWO PEDIGREES FOR NEW PROGENY PEDIGREE"
3870 PRINT"HERE IS THE CATALOGUE OF PEDIGREES ON FILE":PRI
NT:CAT
3880 SOUND 1,90:INPUT"ENTER NAME OF THE SIRE";S$
3890 PRINT:PRINT
3900 SOUND 1,90:PRINT"INSERT PEDIGREE FILE THEN PRESS ANY
KEY"
3910 IF INKEY$=""THEN 3910
3920 OPENIN S$
3930 FOR X=1 TO 31:INPUT #9,A$(X):S$(X)=A$(X):NEXT:CLOSEIN
3940 PRINT:PRINT
3950 SOUND 1,90:INPUT"ENTER NAME OF THE DAM";D$
3960 PRINT:PRINT
3970 SOUND 1,90:PRINT"INSERT PEDIGREE FILE THEN PRESS ANY
KEY"
3980 IF INKEY$=""THEN 3980
3990 OPENIN D$
4000 FOR X=1 TO 15:INPUT #9,A$(X):D$(X)=A$(X):NEXT:CLOSEIN
4010 PRINT:PRINT:PRINT"HAS THE ";YOUNG$;" BEEN NAMED? <Y/N
>"
4020 FOR A=1 TO 10:A$=INKEY$:NEXT
4030 KN$=INKEY$:IF KN$=""THEN 4030
4040 IF UPPER$(KN$)="N"THEN A$(1)=STUDNAME$+" "+YOUNG$:GOT
O 4120
4050 IF UPPER$(KN$)="Y"THEN 4070
4060 GOTO 4010
4070 PRINT:PRINT"ENTER NAME, VARIETY AND COLOUR OF ";YOUNG
$
4080 SOUND 1,90
4090 PRINT
4100 INPUT A$(1)
4110 IF LEN (A$(1)) > 41 THEN PRINT CHR$(7): PRINT"ENTRY I
S";LEN(A$(1))-41;"LETTERS TOO LONG..PLEASE ENTER AGAIN":GO
TO 4070
4120 A$(2)=S$(1)
4130 A$(3)=D$(1)
4140 A$(4)=S$(2)
4150 A$(5)=S$(3)
4160 A$(6)=D$(2)
4170 A$(7)=D$(3)
4180 A$(8)=S$(4)
4190 A$(9)=S$(5)
4200 A$(10)=S$(6)
4210 A$(11)=S$(7)
4220 A$(12)=D$(4)
4230 A$(13)=D$(5)
4240 A$(14)=D$(6)
4250 A$(15)=D$(7)
4260 A$(16)=S$(8)
4270 A$(17)=S$(9)
4280 A$(18)=S$(10)
4290 A$(19)=S$(11)
4300 A$(20)=S$(12)
4310 A$(21)=S$(13)
4320 A$(22)=S$(14)
4330 A$(23)=S$(15)
4340 A$(24)=D$(8)
4350 A$(25)=D$(9)
4360 A$(26)=D$(10)
4370 A$(27)=D$(11)
4380 A$(28)=D$(12)
4390 A$(29)=D$(13)
4400 A$(30)=D$(14)
4410 A$(31)=D$(15)
4420 GOTO 170
```

Games Reviews

Mag Max
From Imagine
(Joystick or keys)

Great games come in many forms. Although I've criticized shoot-em-ups in the past as repetitive and predictable, when they're done well they are wildly addictive. Classy shoot-em-ups have more action than a singles bar and this one guarantees you a severe case of Knackerus Fingeroid Firitus.

All this will be music to the ears of anyone who's been waiting for the game. It has taken quite a while to convert. I can't attest to its authenticity in terms of the arcade version, but I can tell you it's a great game. The scenario, thankfully short, tells you the human race has met its ultimate end, leaving just a robot to battle the alien invaders.

The robot is in pieces: one of them is under your control; the others you have to find. You have to progress through scrolling levels of mayhem, both above and below ground, against an ever increasing number of aliens.

The game is split into levels, each one having both above- and below-ground sections. You can get between them using craters, so that you can travel through whichever one you find easier to deal with. In both cases the screen scrolls from right to left, but the perspectives are different.

Above ground you have a 3D view with the ship skimming over the surface. In the caverns this switches to a 2D view. In both instances the ship can move freely about the screen, but can't stop the steady scrolling that reveals the enemy forces.

As you make your way across the screens you can find the bits of the robot which will give him additional firepower and a new weapon. However,
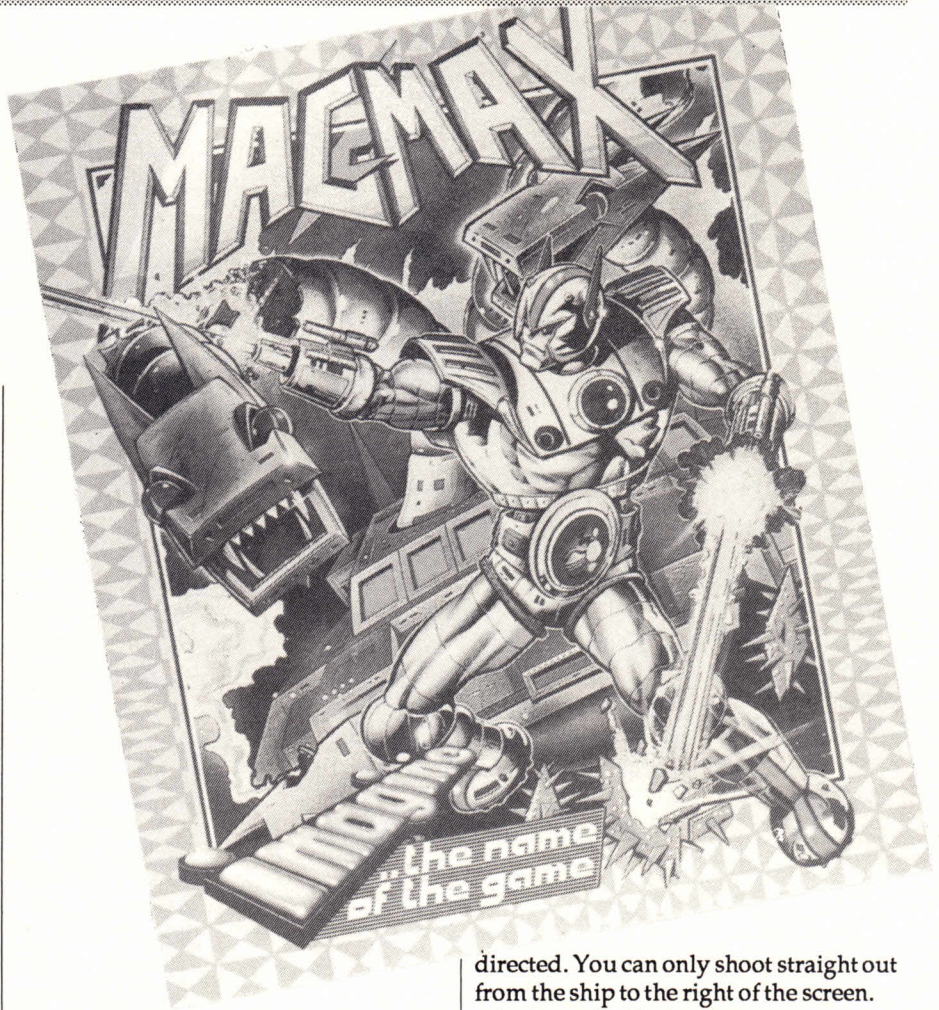
hanging on to these for long can be a problem. The first add-on module looks like a teddy bear and provides the upper half of the body. When the legs are added it gives you a wider field of fire - but also a bigger target for the aliens. A lance can also be added to the body section to destroy otherwise indestructible objects.

The surface aliens come in all sorts of shapes and behaviour patterns. Some are static and blast away at you; others dodge around trying to zap you. Different types of each are revealed as you get further through the game. If you run into them or their bullets you will lose a piece of the robot. When there are no pieces to lose, a life will go down the swannee.

Underground the aliens all fly and move in patterns. Like the surface ones they tend to come in waves. All the bullets are aimed at your current location when fired, so you have to be constantly on the move around the screen to avoid their deadly accuracy. Your own firepower also needs to be accurately

directed. You can only shoot straight out from the ship to the right of the screen.

At the end of the level you are supposed to run into a dragon that has to be shot bit by bit, but I didn't notice one - just more and more waves of aliens. The background graphics change between levels, and of course the aliens get meaner and more plentiful.

What with constant moving, no auto-repeat on the Fire button, hordes of the enemy and needing a high degree of accuracy, this is a testing shoot-em-up indeed. Fortunately the aliens do appear in the same formations every game so that you can learn the patterns and progress further. There are also regular "restart" spots where you will begin after losing a life.

The graphics are detailed, colourful and varied. Combined with a reasonably smooth, slow scroll, it makes an excellent game to look at. The gameplay is demanding but very addictive. The crucial aspect is that of variety; there are so many different aliens and attack patterns, a 3D and 2D view, weapons and parts to collect, different levels to move on to.

This is the sort of game that rewards hard work. The longer you play it the better you'll get. You'll certainly come away with a much fitter fire finger and a throbbing hand, a sure sign of a good shoot-em-up. It will take you to your game playing limit and still keep you hooked.

Stormbringer
from Mastertronic
(Joystick or keys)

There's not much left to be said about the brilliant series of games starring Magic Knight. He first came to light in Finders Keepers but really achieved stardom in Spellbound and Knight Tyme with the introduction of the "Windimation" system of window menus. The last in the series once more uses the same system to good effect.

Having clanked around in the 25th century in Knight Tyme, Magic Knight has travelled back to his own time. However, on the way he's been split into two, leaving you as the good Magic Knight and the Off-White Knight, your dark side. Your task is to merge the two halves of MK to prevent the Off-White Knight, nicknamed Stormbringer, from wreaking havoc in the land.

If you've already played the previous games in the series you'll be right at home with this one. It features the same interaction between characters, Windimation and weird humour. However, don't expect that to make it any easier because the puzzles David Jones has concocted are more devious than ever.

If you're unfamiliar with Windimation here's how it works. The Fire button is used to call up a window menu from which all the other menus can be accessed. This main menu contains five standard commands: pick up an object, drop an object, take object (from character), give object and examine. It can also feature a number of additional commands depending on what objects you are carrying. These allow you to cast spells, read objects, command characters, wear things, smell, drink and throw things.

Most of these main commands will produce submenus that contain either information or a further list of choices. For instance, if you want to examine something you have to decide whether it's an object, character or yourself, which object or character, and finally see the information relating to your choice. All this information flicks up on a new window and is superbly presented.

There are other characters in the game who move around independently of you. You can give them objects or take them, depending on whether the character is co-operative. You've also got to look after their welfare and can request help from them if you think there's a task they can perform which you can't.

Watch out for Grunter the Bearwoolf, who's an enormous, mean-looking monster and can end the game with one swipe. Another deadly problem is a cloud that floats around, produced by the Stormbringer: it will zap you with lightning if you stand around too long.

There are no significant improvements in this latest MK adventure, but then, quite frankly, it didn't need any. Fans of his previous games will be deliriously happy with the new puzzles, characters and humour, while newcomers to the series will probably rush out and buy the previous games as well.

The graphics and animation are once again adequate and a pleasant in-game tune is provided. As ever you don't just get a tune though, you have to pick up and wear a personal stereo in order to hear it - cute huh? I'm still full of admiration for this series of arcade adventures and it's fitting that they should exit on this high note.

Ball Crazy
From Mastertronic
(Joystick or keys)

Authors keep coming up with new ideas for bouncing games and The Firm has produced one that will once again try to persuade you that you're a yo-yo. It's a one-screen challenge but it's addictive and has plenty of depth.

At the bottom of the screen is a row of five coloured blocks. A happy, smiling, cheerful, grinning, I'm-glad-to-be-alive face appears on the blocks and bounces up and down as if there's absolutely nothing wrong with the world. Near the top of the screen is another coloured block and just above that is a square orifice (this is a happy, smiling, have-a-nice-day term for a hole).

The idea is that by bouncing on the blocks you cycle through a series of colours until the block is the same colour as the one at the top of the screen. Once all five blocks are the right colour another layer of blocks will appear on top of them and they will have to be changed to a new colour.

Now this wouldn't be much fun unless there were some problems along the way, and there are. The gap in the fabric of the game screen's backdrop (hole) will produce some strange alien creatures. These fly around the screen bouncing off the walls, ceiling and floor until they either hit you or get hit by a bullet. Of course, it's not healthy to be hit by these guys and it's you that can shoot the bullets.

If you shoot an alien another one will soon appear. On later levels more than one alien will appear at once, making it harder and harder to avoid them. When your dementedly placid visage gets punctured it will deflate on the ground still grinning its little heart out.

You aren't totally vulnerable. Other objects come tumbling down the screen and you collect them. There are extra lives in the form of air bottles, extra bullet canisters and a spinning green tick that will automatically turn all the blocks to the right colour. There's also a satellite that gives you some weird headgear. It provides invulnerability but also stops you collecting other objects. However, I don't recommend this because the face doesn't look half as pleased with itself as usual.

The layers of blocks aren't always flat but come in steps as well. This can make life particularly awkward when the blocks have built up to three or four layers. After four layers have been completed you move onto the next stage with more aliens. After every eight layers you get bonus points.

Despite being very simple in concept it's surprisingly addictive. You've got to anticipate where the aliens are going and try to avoid them by jumping,

staying on the ground or shooting them. You've also got to grab all the extra lives and bonuses going to get as far as possible. One tactic is to hang around on the first level, build up plenty of lives and see how far you can get.

The game is well programmed using plenty of colour and detail. I'm convinced the smiling face is some sort of psycho-analytical tool to make the user feel placid and well-meaning, so you're bound to like him - aren't you? The sort of game you can play for hours and keep coming back to.

Barbarian
From Palace
(Joystick or keys)

The advertising was sexist and it's yet another combat game, but surprisingly enough there's a lot of class about it. It's nearly two years since "Way of the Exploding Fist" started off the constant stream of combat games and it's possible that this could revive the genre.

The concept is simple enough and hasn't changed since Fist. Two guys fight it out in 2D until one of them gets hit enough times. In this fight the combatants are armed with swords and you've got to battle to release a fair princess (yawn). There are three forms of fight: one player against the computer in practice, one player against the computer for real, and two players.

There are two different backgrounds for the practice fighting and two for the real thing. You can only play with two players in the practice mode, where the rules are also different. The four backgrounds are all superbly drawn and detailed, adding a nice atmospheric touch.

The fighters are all the same muscle-bound warriors but with varying flesh hues and different-coloured jerkins. They are stunningly animated through 16 possible moves with many intervening frames. The moves are split into two types: movement without the Fire button pressed and attacking strokes with the button pressed.

The more defensive movement commands include forward and backward rolls, crouching, jumping and protecting the head and body with the sword. The attacks are very varied including a head-butt, web of death (spinning sword) and overhead chop.

Whenever an attack contacts the opponent a red flash indicates the hit. This will reduce the strength of the opponent - it starts off at six and is reduced by a half for every hit. When no strength is left the fighter will collapse and get dragged off screen by a little green monster looking like a cross between a toad and ET.

There is an even more spectacular way to dispatch opponents, which is not for the faint-hearted. If you successfully time a flying neck chop it will end the fight immediately: your opponent's head flies into the air and his torso flops to the ground spurting blood. This gruesome sight isn't easy to achieve except against the early computer opponents. It's really hilarious when it happens. The green monster will appear once again. As well as dragging away the body he unceremoniously boots the bonce off screen. (Yuk - Ed).

There are seven computer opponents of increasing cunning and skill. The harder ones require a lot of moves rather than just relying on two or three. The two-player game can obviously be a lot more fun and unpredictable. Each bout has a time limit. If you leave a fighter unmoved he'll turn to you and shrug his shoulders.

The graphics and animation are unsurpassed, the music excellent, the action difficult, and the sense of humour great. However, let's face it: it's a combat game pure and simple. The basic concept and action hasn't changed at all since the days of Fist, and I find it hard to get excited about the format. What the game has going for it is that it combines the graphics, animation, speed and gameplay of all the other combat games to make a very playable one. You'll be delighted at first, but with all the tarting-up in the world you can't disguise the basic gameplay.

Two Chesty models are helping to promote Palace Software's recently released game, *Barbarian*. The only information we have to go on is Michael Vanwigk's (the guy with the hair) 53-inch chest and Maria Whittaker's softer one. You'll pleased to know (as we were) these two delightfully moulded models will appear on the title screen of the game.

YEAR DISC 7

From Issue 33
ANDIGCLK
ADLIST10
AMSMATCH
BOLDCHAR
CPATTS
FUZZY
ITALCHAR
MINIBUG
RANFILE
SCROLLRT
SINEWAVE
STARDODG
THINCHAR
WEEKDAY
WEEKFIX

From Issue 34
579MAGIC
NEUTMAZE
QUIZZ
TEXTGEN
ZYCLOID

From Issue 35
BHPSCROL
MISER
RALLY
REACT
RNDXMAS

From Issue 36
BLACK464
BLACK128
CHBYCHR
DEGREES
POUND#
THERAPY
UNSCRAM
VARICAT
WEAVING
XPANDOCP
XPANDPIC

Plus this FREE Basic and CP/M Public Domain Software:

BAGELS - A "guess the number game" for kids.

CHASE - See if you can out-smart the guards with this game.

CHOCHIK - Without getting zapped, try to eat all the worms.

DIR22 - Extended directory.

LABELMKR - Always know what is on your disc with label maker.

POET - Sit and watch while your computer creates amusing random poems.

SUNRISE - Very accurately calculate at what time the sun will rise.

TEMPCONV - Check the temperature in every scale

TINYPRNT - Configure your printer to print IN TINY CHARACTERS

VFILER - Similar to Newsweep but easier to use.

WUMPUS - A simple text adventure for all ages.

Still only $25
including postage and packing

Send your order to:
The Amstrad User
1/245 Springvale Road
Glen Waverley, Vic 3150

Credit Card telephone orders taken on (03) 233 9661

Gallimaufry IX

Three more reasonable type-ins to spin, plot and scroll your CPC into action.

Spinning Diamond

By switching Inks (or palette switching as it is known) can make some interesting effects. Giles Park has experimented and come up with this program. Type it in and see the wonderful effect.

```
1 ' Spinning Diamond
2 ' by Giles Park
3 ' The Amstrad User, Feb(88)
10 ' ****set up screen + colours****
20 INK 11,26: FOR t=0 TO 15: INK t,t: NEXT: INK 11,26
30 ' ****draw stars****
40 MODE 0: FOR r=1 TO 20: FOR e=10 TO 5 STEP -2
50 FOR w=1 TO 3: MOVE -100+RND*640,1+RND*400
60 FOR q=1 TO 10: PLOTR e*w,0,q: NEXT q,w,e,r
70 ' ****draw diamond****
80 coltri=1: FOR r=0 TO 90 STEP 9
90 FOR t=0 TO 360 STEP 90: DEG
100 MOVE 320+200*COS(t+r),200+100*SIN(t+r)
110 DRAW 320+200*COS(t+90+r),200+100*SIN(t+90+r),coltri
120 DRAW 320,0: MOVE 320+200*COS(t+r),200+100*SIN(t+r)
130 DRAW 320+150*COS(t+r),300+75*SIN(t+r)
140 DRAW 320+150*COS(t+r+90),300+75*SIN(t+r+90): NEXT
150 ' ****spinning part****
160 coltri=coltri+1
170 IF coltri>10 THEN coltri=1
180 NEXT: coltri=26
190 FOR t=1 TO 10: INK t,coltri: INK t-1,0
200 CALL &BD19: INK 10,0: NEXT t: GOTO 190
```

Pattern Plotter

Here's a handy program for anyone with a plotter: it generates patterns and gives you the option of sending it to a Tandy plotter. Paul Bridal is the man to thank for the code. If you have a different brand of plotter, a breakdown of the codes may help you to write your own.

- CHR$(17) Sets text mode.
- CHR$(18) Sets graphics mode.
- S1 Gives 40 columns.
- P Lets you mix text and graphics.
- M Means move pen.
- I Sets the origin.
- D Means draw.
- A To return to text mode.

Below is a demo of one of the pictures you can create. This dump was performed on a TP-40 plotter.

Step Value: 2
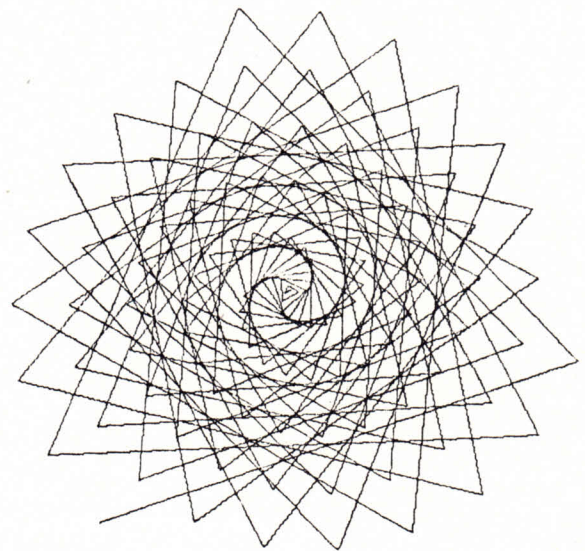
```
100 ' Pattern Plotter
110 ' by Paul Bridle
120 ' The Amstrad User, Feb(88)
130 ' FOR tandy CGP-115 plotters & Compatibles.
140 MODE 2:INPUT "Step Value (0.1 to 4 are best) ",stp
150 ' screen draw
160 ORIGIN 320,200: FOR a=0 TO 230 STEP stp
170 DRAW a*SIN(a),a*COS(a): NEXT
180 INPUT "Do you want to plot-out this ? ",plt$
190 IF UPPER$(plt$)="Y" THEN 210 ELSE 140
200 ' plot
210 PRINT#8,CHR$(17):PRINT#8:PRINT#8,CHR$(18)
220 PRINT#8,"S1":PRINT#8,"PStep Value: ";stp
230 PRINT#8,"M240,-260":PRINT#8,"I"
240 FOR a=0 TO 230 STEP stp
250 PRINT#8,"D";a*SIN(a);",";a*COS(a): NEXT a
260 PRINT#8,"A": GOTO 140
```

Message Scroller

How many different routines have you seen that will scroll text ? We've seen many. Daren Vernon's scroller however is slightly different. The characters are larger than standard and whiz by very smoothly. As it stands, the routine will scroll messages in red on a black background The colours can be changed by poking &9005 with the border colour, &8FD0 with paper colour and &8fd1 with the ink colour.If you want, you can bypass the Mode and colour setup with CALL &901D. Scroller works only in Mode 0.

```
1 ' Message scroller
2 ' by Daren Vernon
3 ' Amstrad Action     October 87
10 ' *Data for machine-code*
15 DATA 00,06,55,FF,FF,AA,55,55,AA,AA,00,55,AA,00,00,FF,6F
F
20 DATA FF,AA,00,55,AA,FF,00,55,AA,55,55,AA,55,AA,55,7F
8
25 DATA AA,55,AA,55,AA,FF,00,FF,FF,AA,00,00,00,00,00,00,64
F
30 DATA AF,CD,0E,BC,01,00,00,CD,38,BC,AF,21,D0,8F,E5,F5,81
1
35 DATA 4E,41,CD,32,BC,F1,E1,23,3C,FE,10,20,F1,21,A2,90,7E
D
40 DATA 22,9D,90;3E,03,32,9F,90,CD,19,BD,CD,3C,90,3E,2F,69
A
45 DATA CD,1B,BB,30,F3,AF,CD,0E,BC,C9,62,42,3A,9F,90,3C,81
E
50 DATA E6,03,32,9F,90,20,1E,2A,9D,90,7E,23,FE,FF,20,05,6A
2
55 DATA 21,A2,90,7E,23,22,9D,90,CD,A5,BB,11,DF,8F,ED,53,82
F
60 DATA A0,90,CD,53,BC,21,50,C5,ED,5B,A0,90,13,ED,53,A0,8A
D
65 DATA 90,1B,01,08,03,C5,1A,E6,C0,D5,C5,E5,5D,54,23,01,69
0
70 DATA 4F,00,ED,B0,12,E1,C1,7C,C6,08,67,30,04,11,50,C0,6A
6
75 DATA 19,D1,10,E2,13,13,13,13,C1,0D,20,D9,C9,B7,90,03,60
2
80 DATA E3,8F,20,20,20,48,6F,77,27,73,20,61,62,6F,75,74,5D
5
85 DATA 20,74,68,69,73,20,66,6F,72,20,61,20,6E,65,61,74,58
8
90 DATA 20,73,63,72,6F,6C,6C,69,6E,67,20,6D,65,73,73,61,62
6
95 DATA 67,65,2C,20,65,68,20,3F,20,20,20,20,20.20,20,FF,42
3
100 ' *Poke in machine-code*
110 h=HIMEM
120 addr=&8FD0: MEMORY addr-1: FOR x=0 TO 16: FOR y=1 TO 1
6
130 READ byte$: byte=VAL("&"+byte$): POKE addr,byte
150 addr=addr+1: checksum=checksum+byte: NEXT
160 READ chk$: chk=VAL("&"+chk$)
170 IF checksum<>chk THEN PRINT"Data error in line";15+15*
x: END
180 checksum=0: NEXT
190 ' *Input message*
200 INPUT "Message to scroll ";message$
210 IF message$="" THEN GOTO 250
220 FOR x=1 TO LEN(message$): byte=ASC(MID$(message$,x,1))
230 POKE &90A2+x,byte: NEXT:POKE &90A2+LEN(message$)+1,&FF
240 ' *CALL machine-code scroll*
250 CALL &9000
```

(Vortac Listing continued from Page 21).

```
550 '{------------------------------------}
560 'Procedure Perform_Calculations;
570 DEG:a2=d-e
580   s2=SQR(((s3^2)+(s1^2))-(2*s3*s1*COS(a2))):'USING COS
INE RULE
590   sina3=(s3*SIN(a2))/s2 :'USING SINE RULE
600   cosa3=SQR(1-(sina3^2)):'USING TRIGONOMETRY
610   a3=ATN(sina3/cosA3):'USING ARCTANGENTS
620   x=e+180-a3:'USING QUADRANT LAWS
630 RETURN
640 'End of Perform_Calculations........
650 '{------------------------------------}
660 'Procedure Print_Out_Results;
670   PEN 1:PAPER 0:CLS
680   LOCATE 16,2 : PRINT "According to the figures suppli
ed to me,";
690   LOCATE 16,4 : PRINT "if you follow the following dir
ections";
700   LOCATE 16,6 : PRINT "you should reach the destitanat
ion air-";
710   LOCATE 16,8 : PRINT "port.";
720   LOCATE 16,11: PRINT "Steer a course of";FIX(x);" deg
rees.";
730   LOCATE 16,13: PRINT "Travel a distance of";FIX(s2);"
 Km.";
740   LOCATE 9,15 : PRINT "Do You Wish To Calculate More D
istances (Y/N) ? ";:INPUT "",ans$
750 RETURN
760 'End of Print_Out_Results........
770 '{------------------------------------}
780 '{         MAIN PROGRAM          }
790 'START HERE..
800   BORDER 26:INK 1,0:INK 0,26:ans$="Y"
810   GOSUB 450 :' SET_UP_SCREEN.......
820   WHILE UPPER$(RIGHT$(ans$,1))="Y"
830    GOSUB 340:GOSUB 570:GOSUB 660:' COLLECT_INPUTS, PER
FORM_CALCULATIONS, PRINT_OUT_RESULTS.
840 WEND:INK 0,0:INK 1,26:PAPER 0:PEN 1:MODE 2:PRINT "Bye
.....":BORDER 0:END
```

Missing link

The first of a new series by Kevin Mclean introducing VIATEL, the national telephone database

As there are enough Amstrad Viatel Users now this column maybe of use to those connected to the service. Even if you are not connected there may be some news from time to time to warrant reading this article. An expert on Viatel, I am not, but have been using it for various reasons since early 1986.

Viatel is essentially an electronic database that can be accessed via the telephone. In order to do so you need a computer or keyboard some connecting equipment and some software. Telecom runs Viatel and acts as traffic policeman to the various service providers. We will explore all this equipment later.

The Viatel database has a wealth of information and these include everything from home banking to TAB betting. Electronic mail is probably the major advantage with the facility to send text long distances instantly and receive a reply instantly (provided the other person checks his mail-box). Up to date computer info and prices can be obtained from the various service providers as well as ordering and purchasing via bankcard or the like.

Business people have a pandora's box of services including banking and stockmarket telebroking as well as agricultural and economic info. Some of these services require an additional subscription, for example Money Watch, which monitors financial info. We will investigate some of these services in depth as we look further into Viatel.

Recreational services include sports news, betting and entertainment guides. Air fares and most transport schedules are available, although I think a phone call to the airline or bus service would do the job a lot easier. There are also real time talk sessions (or close enough to it) as well as games, jokes etc.

Connection equipment needed to access viatel has been a hassle for the AMSTRAD but appears to have improved by the look of some of the latest Amstrad magazines. By the way much info can be gained from the several magazines available, the UK version and the Australian as well as a few publications I have not sighted as yet.

Equipment:

RS232 Add-on interface with power supply.
Modem (1200/75)
Interface to Modem lead.
Telephone.
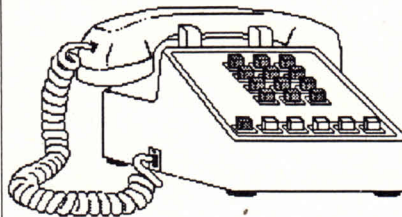
The next item is software and as you may have gathered this is the one causing all the problems. There are some commercial programs available from Sage at one end of the scale (in the hundreds) to Honeysoft at the other end of the scale (in the forties). Some software has auto log-on as well as the ability to transmit a text file. This saves time (and money) as these can be composed off-line, then transmitted when on-line.

Now, down to crunch time, meaning COSTS for all these services and equipment. I won't go into computer equipment as a quick glance through the available magazines cover these. (A modem and RS232 can be bought for $359 or constructed cheaply by the more knowledgeable).

VIATEL RATES

Subscription:
Business	$12.50/month.
Home	$3.50/month.
Local call fee	18c.
Connect time charge	Mon-Fri 8 to 6 9c/min, other 6c/min.
Frame charge	Set by indiv. ser. prov. average 10c - 20c.

So....I hear you say, is it all worth it. I think it depends on each user and what he or she obtains from the services. To quote the Telecom magazine "the standard of most SP databases is lower than can be reasonably expected" and sometimes I feel Telecom themselves are only out for the quick buck but if you have a use for Viatel and intend to continue you should be comforted by the fact that Viatel users are increasing by the rate of 1,000 per month, a staggering figure considering there are already 12,000 users.

Computer info for the Amstrad is limited at the moment but appears to be on the increase. Other computer developments can be monitored by a number of SP's including some of the big names in computers (Nashua, IBM, Commodores etc.)

Well that should be enough to digest for one session, we can have a look at getting to use some of the various services such as banking and stockmarket info next time.....see you then.

Adventurer's Attic

While we wait for Philip Riley to return from his adventure in S.A. Tony Flanagan initiates beginners in the gentle art of adventuring with a look at the classic Hitchiker's Guide to the Galaxy

Have you ever looked at the games review pages in magazines and wondered what all this fuss about 'adventure' games is? Maybe you've actually gone and bought one and given up after a few frustrating hours. You need to penetrate the adventurer's mentality to really start to enjoy these games; you need to know a few basic tips and tricks on how to get answers you want. Without giving too much of the fun away, here's a gentle tour through The Hitchhiker's Guide to the Galaxy for the newcomer.

So, anyway, it's like this. One day you wake up with one hell of a throbber which feels as if there's a pnuematic drill attacking pieces of your skull. If only you hadn't overloaded your puny organism with rather efficacious poisons of the alcholic kind.

To make matters worse, you soon discover that the council is outside with a big yellow bulldozer heading towards your albeit humble abode. Perhaps you should have paid the arrears. Still, bulldozers? That's heavy. It's not until you open the junk mail that you discover that a demolition order has been served on your house to make way for a bypass. Things can't get worse.

Well, as one might expect, things can and do. That well-known body The Galactic Hyperspace Planning Council has apparently served a demolition order on the Earth itself. This is to make way for a hyperspatial express route which, even from a mere earthling's point of view, does not seem wholly unjustified. After all, why should a fractious little planet like ours stand in the way of progress?

And so begins one of the best adventure games you're ever likely to encounter: The Hitchhiker's Guide to the Galaxy. This, as most earthlings are no doubt aware, began life as a radio series, was transcribed to the printed page, then later appeared on TV which, all in all presumably made its progenitor, Douglas Adams, nauseatingly rich. Having read the various books helps at first, but the plot of the game soon takes its own route.

Like all good adventures, Hitchhiker's is an experiment in sustained masochism. As the player you submit yourself to days of ceaseless searching as you try, often in vain, to crack the most absurd puzzles and converse with characters whose manner is strange and whose responses can be far from helpful. Still, there are pleasurable moments - like when the stupidest solution you can think of turns out to be - well - stupid.

Still in the dark

Even though Infocom games are the best, this doesn't mean they're the easiest to play, particularly if you're new to adventuring. There's nothing more demoralising than forking out lots of dollars only to find that the game is unplayable or, to put it more correctly, you can't play the game.

By incorporating a few simple techniques into your play, the whole enterprise will suddenly seem far more enjoyable (unless you really are a masochist of course). Once you make any significant progress in a game it's essential that you save your position. This means that if your progress is suddenly thwarted (a flying piece of masonry perhaps?) then you won't have to start all over again from the beginning. This is simply a matter of typing in SAVE, then inserting a formatted disc with some free space on it, keying in a file name and pressing [RETURN]. Should you wish to restore a saved position, type in RESTORE, insert the disc on which the position is stored, type in a file name.

Another time-saving strategy is to make a map, identifying the name of each scene you enter and its geographical/spatial position in relation to others so that eventually you have in front of you a matrix of squares, all signifying a particular scene in the unfolding drama. Such cartography, does not have to be very complicated. In this way you will save yourself eons of time which otherwise you would spend wandering (and wondering) aimlessly.

To keep track of where you've been and how you got there, Infocom adventures enable you to make a transcript of anything that takes place on screen. This is done by typing in SCRIPT (to begin the transcript) and UNSCRIPT (to end transcript). This is a means of reviewing your progress and making sure that you haven't missed an important clue or failed to pick up an essential object. In Hitchhiker's, for example, it's very easy to forget your towel or even your gown, both of which are indispensable. Infocom adventures admit 'verbose', 'brief', superbrief' modes depending on how detailed you prefer the descriptions. Unless you know a room well it's better to stick with brief or verbose as you might miss vital information.

Like all respectable adventures, Infocom games have the following commands : LOOK (describes a location), EXAMINE (describes an object) and INVENTORY (lists the objects being carried). The command WAIT (or Z for short) is often useful as it gives you time to see what a character is doing or what the result of an action might be. AGAIN (abbreviated to G) repeats the previous command you have made and saves you retyping lengthy commands. In Hitchhiker's, the command DIAGNOSE gives you a

current report on the state of your health. Try this one at the very beginning of the game or later in the Vogon Hold.

Hitchhiker's guide

At the very start of the game you find yourself in darkness. Switching on the light might help but then there's the hangover to contend with. Perhaps an aspirin might help but then of course you've got to find one and with a hangover like yours it's not going to be easy. Perhaps for decency's sake you should put something on.

Having discovered that your house is about to be demolished it's time to step outside and sort this council fellah out! Mr. Prosser's implacability is only matched by his singleness of purpose. No amount of pleading will stop him carrying out what his superiors have ordered him to do. Still, you're not going to take it lying down...are you?

If you do the right thing and wait long enough, Ford Prefect (lately of Guildford, formerly of a small planet in the vicinity of Betelgeuse) will appear to save your house and whisk you off to the Horse and Groom for a quick pint or three before the world ends. The number of pints one drinks is of great importance. Beer, by the way, is a great muscle relaxant, essential for matter transference. Drink too much, and you'll be sent back to the beginning suitably chastised. Drink too little and your body just won't be able to dematerialise and hitch your lift to safety when the moment calls. Back outside, there are only a few minutes left before the Vogons energise their demolition beams. Ford will follow and provide you with the Sub-Etha Sens-O-Matic, an electronic thumb, which is your passport to further adventuring in the cosmic void.

Having successfully hitched a ride you find that, in a manner of speaking, you don't really exist. All that dematerialisation has disorientated you. To continue in the game you must pull yourself together and come to your senses. By waiting and examining the senses the text mentions (or more importantly the one it omits), you'll realise that one sense is a little more important than the others. Use this sense and you'll soon come round but only to pit your wits

with a babel fish dispenser. Ha Ha! Never seen one of those in the works canteen now have you!

Now the babel fish is a canny beast in that it allows you to understand any language, including Vogonese. Getting the fish in your ear, therefore is the main object of this episode and this, one would think, is simply a matter of pressing the dispenser button and bending your ear over the slot. Don't you believe it! It would probably be a good idea to block up any Babel fish-sized holes with appropriate objects - gown, towel and satchel in the right places - then you just might possibly succeed. Don't tell me you left the junk mail behind? Oh dear, you're just going to have to go all the way back to the beginning.....

Now whether you get the fish in yer waxworks or not, you won't escape the Vogon poetry reading. By consulting the Guide which Ford has given you, you can find out much about Vogon poetry and lots of other useful facts. Vogon verse does have a rather rugged vitality which clearly sets it apart from the Elizabethan sonnet. The line 'I will rend thee in the gobberwarts with my blurglecruncheon, see if I don't!' has a directness few modern poets could match.

The end of the world

It's clear that Hitchhiker's is a masterpiece of comic science fiction. Like other Infocom games it's one that can be immensely enjoyable providing it is approached in the right way. There's a great deal to explore, a wealth of puzzles and many fascinating charac-

ters. The only problem is that such games can be too absorbing - you forget the kids' appointment with the social worker, your appointment with the psychiatrist, the psychiatrist's appointment with the social worker.... seen any Sub-Etha Sens-O-Matics around lately?

Expect the unexpected

Now one of the naughty things about Hitchhiker's is that responses to commands are not always totally reliable. In fact, occasionally you'll find that your computer actually tells you teeny-weeny, well whacking great whoppers, so don't take all the responses at face value.

Talking to the other characters can also provide useful tips as well as amusement, though again, don't treat what they say as gospel. They're there to be challenged just as in different ways they challenge you. If you want to talk to a character, type in the name followed by a comma and then your message. For example, Prosser, stop the bulldozer (as if he would). Whether you believe his response or not it does provide you with a valuable clue.

It is also important to remember that every problem in HitchHiker's has a perfectly illogical solution. This is not surprising in an absurd universe, only a little less absurd than our own. Intelligence, for example, is 'the ability to reconcile totally contradictory situations without going completely bonkers - for example, having a stomach ache and not having a stomach ache at the same time' (from the Sirius Cybernetics Corporation android manuals).

Hint Sheet

THE HOBBIT

Hint Sheet From Karla Slack

General Tips :-

1/ Save often - it will save you having to retrace steps.

2/ Draw a map. (especially for the Goblin's Dungeon !)

3/ Keep sentences short. The Hobbit has a few bugs in it which Melbourne House have not removed. Short sentances will reduce the risk of the screen freezing.

The Aim: The aim of the Hobbit is to kill the evil dragon, Smaug, and look for wealth and return with it. This is easier said than done as you will find !

Troll's Clearing: Stay too long and you will find all about the nature of trolls ! It will help if you go ahead and sleep till the day dawns-trolls have a habit of becoming hard like their characters. You may pick up something if you return.

Cave: This is locked. A key is obviously needed - go back a location. When you're in the cave, two objects will be found.

Rope: For throwing across the river E, E, E and E from Beorns.

Shord Sword: What use do you think it has, neophyte ?

Elrond: Is a good reader and seems to carry food around his person.

Beron's House: Open the curtains.

Golden Key: Found in the Deep Misty Valley if you explore well. It's only a treasure.

Goblin's Dungeons: You will probably get captured. If you do, you should have Thorin with you who is tall and will help carry you out the window. Then go SW, D, W, N, E, SW at the lake and go N, SW. You'll be in a place where there should be a ring. You should have found it on your way. Take it. If Gollum is alive, kill him before he kills you. To escape the dungeons from here, go HW, N, D(or is it up ?), E and you'll be in a cave. Go out of the fissure.

Eyes: Wait twice at each location before quietly moving.

Wood Elf:: Will probably capture and place you in Dungeons.

River: Throw rope across to boat..

Boat: Pull, get in and out when at over side.

Web: Smash with sword.

Magic Door: It takes a while for an elf to come out. Wear the ring to slip inside unnoticed.

ElvenKing Dungeons: If there's a warg, kill it. Open the door when someone unlocks it. The only way out is via the trapdoor.

Butler: He has the red key which will open your own cell door. If you have the ring, wear it so he won't see you. If not, kill him.

Barrels: Contain wine. Drink and watch the slurs ! Either pick it up yourself, throw it through the trapdoor and jump in/on it or (if Gandalf is here) tell him to throw it out the trapdoor (when you've climbed in !).

Long lake: Go to the bard.

Bard: Carry him.

Dragon: Shoot him. If you can't aim, get the Bard to shoot.

Side Door: You can't go down it.

Hall Where Dragon Sleeps: Look and you will see...

Treasure: Take it silly !!
(Return To Your Home...)

Chest: What normally goes in a chest ?

Note: You won't lose points for the amount of time you take to finish this adventure and saving your position won't take a toll on your point score which, incidentally, is out of 100 %.

This is our fourth Hint Sheet and earns Karla a cheque for $25. Now that you have the format the sheet should take, why not cash in your experience and send your solutions (don't give whole game away) to:
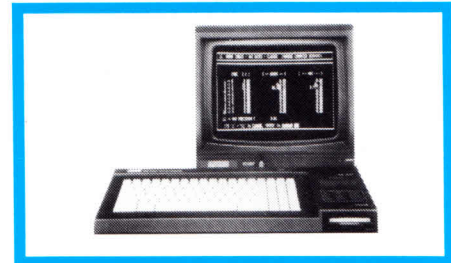
Whether your investment was $500 or $2500 ...

... you can't afford to be without The Amstrad User

The Amstrad User is the only Australian magazine which supports the Amstrad range of computers exclusively and has grown to become the leading national magazine for your computer.

It makes sense to keep up to date with what's happening on your doorstep, to learn with advice from other users and our regular articles, to have fun with the many published games or to have assistance in your business software decisions through our independent software reviews.

The Amstrad User is available through most newsagents (if not yours ask for it - it's distributed nationally by Gordon and Gotch) or take out a Subscription using the form below and ensure regular delivery to your home each month.

MASTERFILE 8000

FOR ALL AMSTRAD PCW COMPUTERS

MASTERFILE 8000, the subject of so many enquiries, is now available through The Amstrad User from Campbell Systems in the UK.

MASTERFILE 8000 is a totally new database product. While drawing on the best features of the CPC versions, it has been designed specifically for the PCW range. The resulting combination of control and power is a delight to use.

Other products offer a choice between fast but limited capacity RAM files, and large capacity but cumbersome fixed-length, direct access disc files. **MASTERFILE 8000** and the PCW RAM disc combine to offer high capacity with fast access to variable-length data. File capacity is limited only by the size of your RAM disc.

A MASTERFILE hallmark is the provision of multiple, user-designed display formats. This flexibility remains, but now it's even easier. With **MASTERFILE 8000** you design your formats "live"; no more questionnaires, just move your format effects around the screen using the cursor keys!

Record updating is even easier than before - just steer your cursor to any field on the screen and then insert/erase/alter as required.

Special options are provided for handling dates and surnames, and column totals can be generated.

All screen work is done graphically - and hence we offer unique panel, box, and ruled line options. Choose the line spacing at pixel resolution - you will be amazed how much clearer 9-pixel lines are than the usual 8-pixels. (Study the picture.) And all this faster than CP/M normally lets you paint the screen! PCW printer functions, under menu control, are provided.

Keyed files are maintained automatically in key sequence, with never any need to sort. You can have unkeyed files too, where records can be inserted at any point in the file.

Any file can make RELATIONAL references to up to EIGHT read-only keyed files, the linkage being effected purely by the use of matching file and data names.

You can import/merge ASCII files (e.g. from MASTERFILE III), or export any data (e.g. to a word processor), and merge files. For keyed files this is a true merge, not just an append operation. By virtue of export and re-import you can make a copy of a file in another key sequence. New data fields can be added at any time.

File searches combine flexibility with speed. (**MASTERFILE 8000** usually waits for you, not the other way around.) You can even assign subsets of a file into one or more of seven pigeon-holes for subsequent reference or further manipulation.

MASTERFILE 8000 is totally menu-driven, fully machine-coded, and comes with example files and a detailed manual. We claim (modestly) that you will not find another filing system with such power, flexibility, and friendliness.

MASTERFILE 8000 costs $119.00 including postage and packing, and if you request air-mail within Australia, we'll do that at no extra charge too! (If you live outside Australia please add $4.00 for air-mail cost.)

Bankcard, Mastercard or Visa orders are welcome, written or telephoned, quoting the card expiry date.

Send your order now to:
THE AMSTRAD USER
Suite 1/245 Springvale Road,
Glen Waverley,
Victoria 3150   Tel: (03) 233 9661