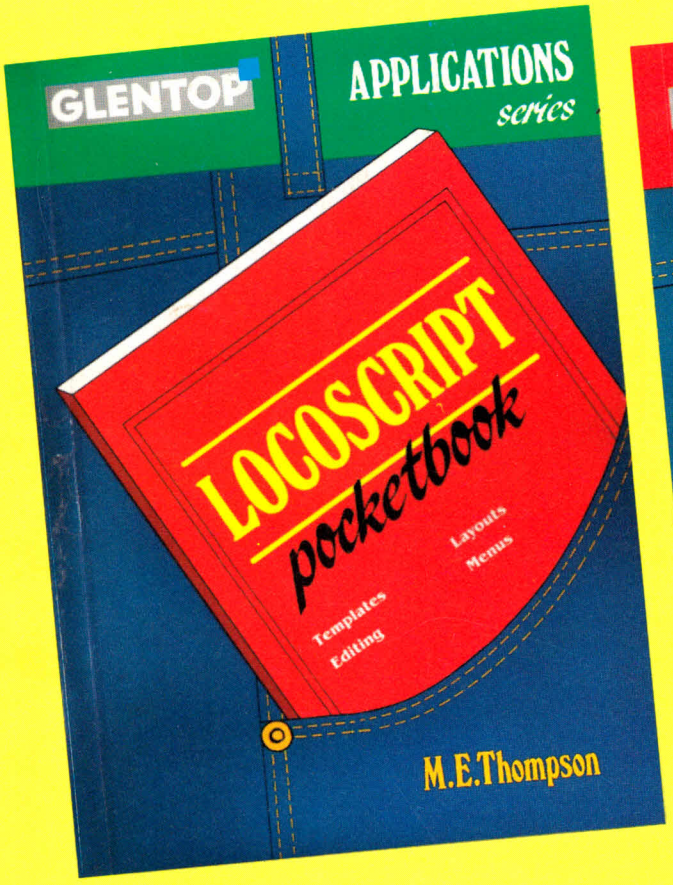
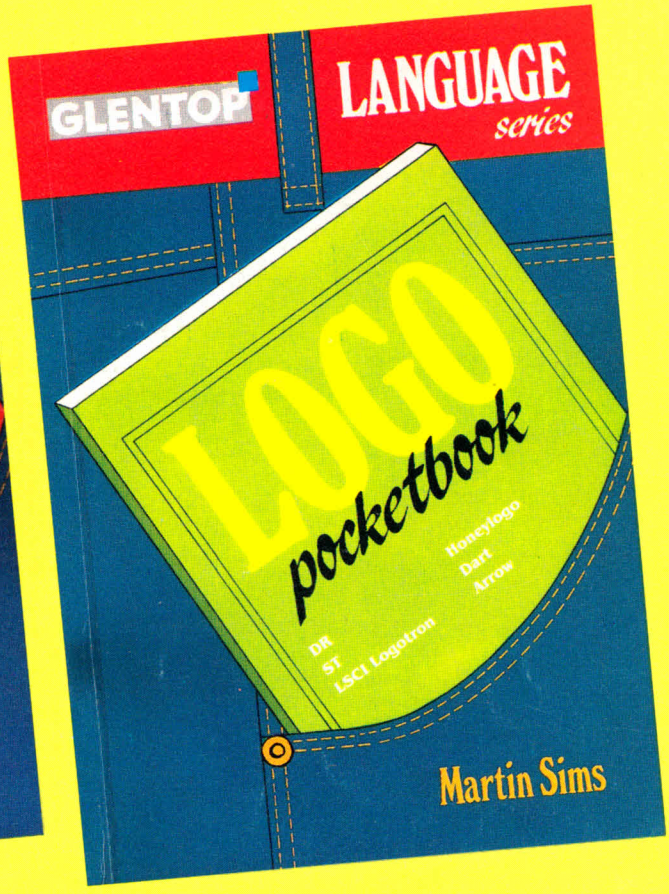
The Aussie Mag for Amstrad owners

THE AMSTRAD USER

Issue No. 33 $3.75 October 1987

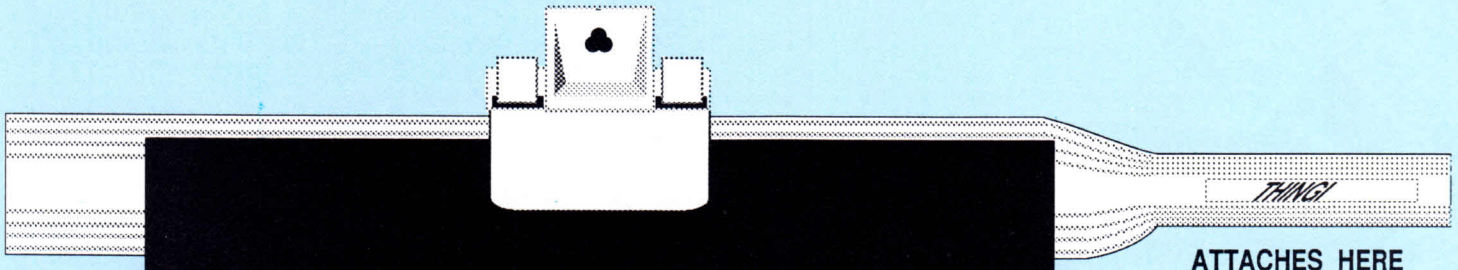




- ***Handling bugs + step by step program tutorial + heaps of type-ins including "Concentration"***
- ***Review of Masterfile 8000 + updated Word Count program + more on LocoMail***
- ***PC Help section and Compatibles Corner***

FOR THE NOVICE & EXPERIENCED USER

Registered by Australia Post – Publication No VBP7017



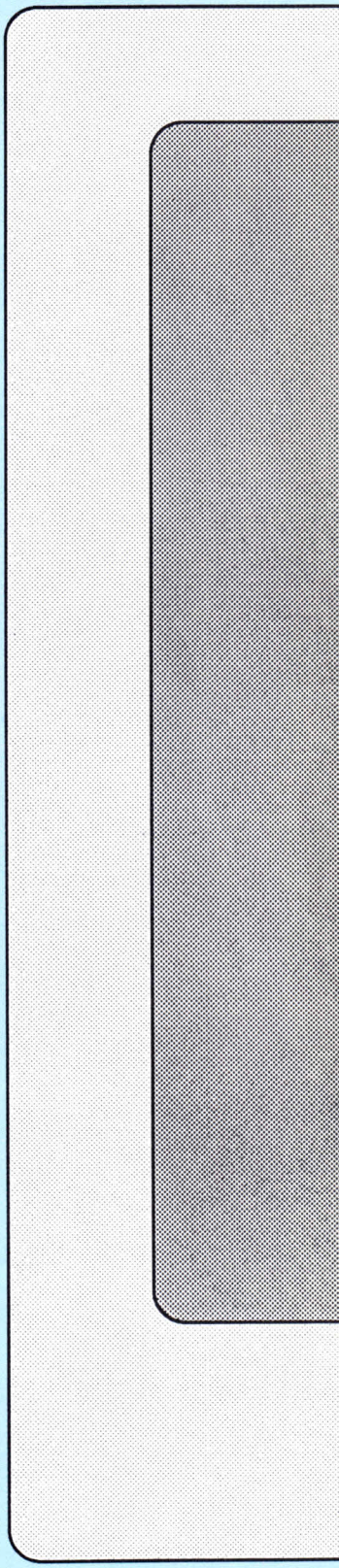
ATTACHES HERE

At last, a simple but effective scientifically designed gadget to reduce neck and eye strain (and even bugs in your program!)

THINGI is very easy to attach to your computer, and will increase productivity by improving accuracy and input speeds.

THINGI will hold single or multiple sheets of paper, even magazines, at the same level as your computer screen.

When you think about it, THINGI has a host of uses outside a computer environment too!



Available through The Amstrad User at $24.95 (including postage)

Please note: Right-hand model will be supplied unless otherwise ordered

Phone orders on (03) 233 9661 or mail to:
The Amstrad User, 1/245 Springvale Road, Glen Waverley, Victoria 3150
Bankcard, Mastercard or Visa accepted.

CONTENTS

THE AMSTRAD USER

Issue No. 33
October 1987

ADVERTISER'S INDEX

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

Side 1:	AMSMATCH	- 11	WEEKDAY	- 46	WEEKFIX	- 59
	RANFILE	- 70	ANDIGCLK	- 122		
Side 2:	ADLIST10	- 9	MINIBUG	- 22	SCROLLRT	- 33
	BOLDCHAR	- 43	ITALCHAR	- 52	THINCHAR	- 61
	CPATTS	- 71	STARDODG	- 86	FUZZY	- 96
	SINEWAVE	- 106				

All enquiries and contacts concerning this Publication should be made in the first instance by writing to The Amstrad User, Suite 1, 245 Springvale Road, Glen Waverley, Victoria 3150, Australia. Urgent matters can be phoned through on (03) 233 9661.

The Amstrad User is normally published on the first working day of each month. The single copy price of $3.75 is the recommended retail price only.

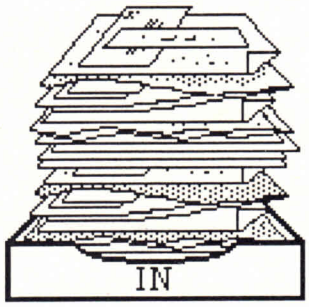
The subscription rate (for Australia) is $37.50 for 12 issues of the magazine only, or $80.00 for 12 issues of the magazine plus tape (for CPC range only) containing programs appearing in that issue. Postage is included in the above prices. For subscriptions to New Zealand, PNG, Solomon Islands or Vanuatu please add $21 airmail. Other overseas prices available upon application.

Please note that whilst every effort is made to ensure the accuracy of all features and listings herein, we cannot accept any liability whatsoever for any mistakes or misprints. Contributions are welcome from readers or other interested parties. In most circumstances the following payments will apply to published material: Letters $5.00, Cartoon $5.00 and a rate of $10.00 per page for programs, articles etc.

Contributions will not be returned unless specifically requested coupled with a suitable stamped and return addressed padded bag (for tapes or discs).

The Amstrad User is an independent Australian magazine and is not affiliated in any way with Amstrad or their Australian distributors Mitsubishi Electric AWA Pty Ltd., or any other dealer in either software or hardware.

Letters



This is being typed on an old-fashioned electric typewriter because of my feeling I could beat the high price of replacement PCW 8256 ribbons by re-inking my old one.

I had read all the warnings about using RP7 and suchlike, but it seemed that stamp-pad ink might be the answer. So I took the ribbon assembly to my carpentry bench, engaged the ribbon feed knob in the chuck of a hand drill, and by dint of turning the hand drill to expose a fresh section each time, carefully painted the ribbon from end to end within 15 minutes.

Back in the printer, the ribbon worked beautifully - nice and dark, even in draft mode. But I noticed that the starting section dried out very quickly if left to stand for an hour or so, but winding the ribbon on a short way overcame that, so it still seemed a good idea.

Twenty-four hours later the printer gave up the ghost - nothing resulted from its efforts. A brand-new ribbon still gave no sign of any print, so out I took the whole thing to the wilds of Heidelberg, and there it waits the ministrations of the experts while I try to go back to pressing the right key first time on one of these things.

John S. Talbot, Ashburton, Vic.

A cheque for $25 to John towards his repair costs and for saving other would-be re-inkers from the same fate.

Looking at K Partridge's letter in the August issue of the magazine I remember how I at first thought it was an easy task to translate, with limited knowledge, programs from other computers.

I have a conversion chart published in the UK, which helps, to a degree, but the biggest problem that confronts you is that the majority of programmers are looking to show off, or use to the best advantage the particular Basic on their machine.

For example in Locomotive Basic the Locate command is great for text display and is not found on other computers. With the VZ200 I had first, the duplicate command was Print@n, where n was a location on the screen.

So the task in front of the converter is easier said than done.

The other problem is that most good and fast programs use memory locations through Peek and Poke. Needless to say these are far from uniform.

So for those with limited programming knowledge, do as I did, FORGET IT!

Although only to a degree. Fortunately Locomotive Basic is the nearest basic to standard Microsoft that I have found.

This means that there is a wealth of program listings in magazines and books which can be readily converted for the Amstrad.

One of the best authors to watch is Tim Hartnell who heads (I think he still does) Interface Publications. They produced the Amstrad Omnibus, and Pentacle adventure creature.

Using one of Hartnell's early books (no reference to the Amstrad), I was able to produce a few modest programs, including - surprise surprise!!! a poker machine game. These listings are skeletons only, using ASCII or keyboard characters instead of graphics. Once they are typed in and you can see how they work, without the intricacies of graphics or special commands, then there's no stopping you.

In fact doing things this way means that you can concentrate on how your own machine works, not how another copes with things.

I am presently compiling an ASCII file of this conversion chart (a long job) and hope to send it to TAU for use later.

Nick Van Kempen

I have been a reader of your magazine for a year now and look forward to receiving it each month. Two things of interest in your August issue were the text dump program and also your request as regards a classified section.

I enclose a basic subroutine I use in a family tree program I have written. This uses the COPYCHR$ command of the 6128 to achieve a very effective ASCII screen dump, in this case in Mode 2. The loop in line 9330 would need to be changed for another mode. Line 9310 sets the printer into condensed mode with reduced line spacing (this looks better) and line 9380 resets the printer after use.

```
110 'acsii screen dump
120 PRINT#8,CHR$(27);CHR$(64):
PRINT#8(27);CHR$(77);CHR$(27);
```


All letter for the Mailbag section should be addressed to:
The Editor
The Amstrad User
1/245 Springvale Road
Glen Waverley, Vic 3150
We regret that we cannot enter into any personal correspondence.

```
CHR$(48);CHR$(13)
130 FOR y=1 TO 25
140 FOR x=1 TO 80
150 LOCATE
x,y:PRINT#8,COPYCHR$(#0)
160 NEXT x
170 PRINT#8,CHR$(13)
180 NEXT y
190 PRINT#8,CHR$(27);CHR$(64)
200 RETURN
```

I would be very interested to see a classified section in your magazine, this is one part I always like to read in the overseas magazines, most of which are now too expensive. I would agree that commercial software should not be sold through such advertising, but I for one would like to advertise the above mentioned family tree program in such a column. I have previously sold my Spectrum version of this program through user magazines.

Finally, could you or a reader please tell me the significance of the character "@" when used with strings and variables? I notice it is used in the Recipe program I received on your Year Disc 1 and have seen it in some listings but not in others. I find no mention of it in the 6128 manual except in the BANKMAN commands and even then its use is not explained.

Jeff Raphael, Turramurra, NSW.

The "@" character before a variable (number, string or name) identifies the location in memory of that particular variable. For example, type v$="HERE" and return. Then type PRINT @v$ - this will result in a number being displayed and is the location of v$.

The response to our request for feelings on a classified section in the mag has fallen mainly on deaf ears. There were a handful of readers who expressed agreement (a 'thank you' to them) but not enough, we feel, to go ahead.

Finally, we are currently evaluating a Genealogical database program from the UK, suitable for CPCs, PCWs and PCs. It can store all the standard details - name, sex, date and place of birth/baptism and death/burial, dates and places of marriages, names and birthdates of up to 19 children and so on. Each file can hold 1000 people. CPC users with single drives are restricted to two files, but for others the sky's the limit. The facilities provide to automatically update associated records, search for specified individuals, report of occurrences, construct lines of descent, produce indented chart of descendants over four generations, browse .... and the list goes on.

Watch future issues for this and/or Jeff's version.

I have another method of drawing bingo numbers which I think almost resembles the 'real life' method of drawing numbers.

My method removes the number drawn from the list of possible choices entirely, thus eliminating the possibility of that number being drawn again, therefore no check is required. As a result the time taken to draw each of the 100 numbers remains constant throughout. Peter Campbell's method (TAU May '87) however becomes slower as the number of numbers drawn increases.

```
10 CLS:DIM a(100),b(100)
```

```
20 FOR k=1 TO 100
30 a(k)=k
40 NEXT k
50 FOR j=1 TO 100
60 x=100-(j-1)
70 bnum=INT(RND*x)+1
80 PRINT a(bnum)
90 b(j)=a(bnum)
100 GOSUB 1000
110 NEXT j
120 END
1000 y=a(bnum)
1010 a(bnum)=a(x)
1020 a(x)=y
1030 RETURN
```

As a result of the swapping, it is possible to read the list of numbers drawn and therefore the order of drawing by working backwards through the array. This enables the second array b(100) to be removed.

James Williams, Ravenswood, Tas.

Please cansul my subscription to your magazine. You dont put enuff games in it. If you like you can by my amstrad. I will use the muney to get a skateboard.

Ben Church, Raglan, (?).

How about a spell checker first before you go down hill? 'Nuff said!

I am writing to complain about the packaging of those cheap tape based games.

If you've picked one up before, you would have noticed a colourful fantasy picture on the front and a small hole on the back through which some blurb about the game can be read. Some inlays may even have a picture of the actual game as well (often not).

Now that's all very well and good if you've about three friends with Amstrads or you belong to a user club and have seen it before. However, if you don't - then tough!

You see, with the more expensive disc/tape games (which I naturally prefer) the guy/girl behind the counter will generally be only too pleased to demonstrate the thing. However, with these 'little beggars' it's "Oh, sorry . . .We can't break the seal". It's really annoying. Sure, I've guessed a few times, sometimes for the best, but $9.98 isn't cheap when the games fail to satisfy.

We need an alternative - I believe better packaging would make life better for all concerned.

Peter Sims, Footscray, Vic

I could not let the letter from K.A. Partridge of Port Pirie (SA), published in the August issue, go without a reply from at least one PCW owner.

I applaud your reply re. the complaint from this reader concerning the content of the magazine. I think it is great that you take the trouble to cater for all types of Amstrad computers. Perhaps if K.A. Partridge read some of the articles for other models he/she might broaden their knowledge. I find the articles very informative and entertaining even if they do not cover the PCW. As to the remark that the PCW machine is a business machine, yes, it can be used for business but so can the CPCs using business type programs.

I bought my PCW late last year to assist me with my studies so I could work smarter and save time from writing and re-writing by hand. The PCW has done this for me but it has also opened up a whole new interest in computers. I do not only use it for study but also for pleasure. I think the PCW is greatly underrated by people who do not know its capabilities. Just two examples of alternatives to business type programs that work well on the PCW are "Printmaster" and its companion "Art Gallery" that can give hours of pleasure and fun. Also the desktop publishing programs. I'm learning to use Fleet Street Editor Plus at the moment - a program that would surprise people who thought that the PCW had no graphics capabilities.

I'm not into games, but I don't put down those who are and I am glad we can all share a magazine that caters for everyone.

David Breach, Hallett Cove Est., SA

Yes, we are in an era where violence resides alongside us. I agree with what S.A. Mah of Willerton said in his letter to your Mailbag. This is so plainly shown by your recent burglary you notified readers about. Violent games probably do have an effect on the subconscious!

Arcade games ARE violence based so if someone can inform me of the ones that aren't, I'll be delighted to know about them. So do some adventure games. I don't believe games that don't have a bit of violence at the least would sell very widely, or exist for very long.

This, I believe, is basically because human nature is violent (read the Bible if you don't believe me) and the frustrations of our society cause people to have to channel their frustrations somehow, else they'll go mad. Games with violence are one of these channels, as are football etc..

I am a Year 10 student at High School intending to do my HSC and continue on to University and either be a Barrister, Journalist or Artist. Therefore I need to obtain good results and work hard. The frustrations and stress of my school work and study need to be released and I find Arcade and/or Adventure Games the perfect outlet (eg. when feeling upset or angry I go and 'KILL ORC WITH AXE'). I am sure other do the same. I very much doubt that games can exist without a degree of violence. The only way they can is when people stop needing to release frustration and stress of day to day life and the only way this can be achieved is by having a society that runs at an unstressful, gentler pace.

Karla Slack, Springwood, NSW.

What a lovely thought! No more magazine deadlines, let the phone ring, find the golf clubs - goodbye TAU until whenever!

I have come across an interesting machine code subroutine for the CPC6128. It can be used in controlling the relay normally used as a remote switch for cassette decks. Although this only gives the effect of a simple toggle switch, it is by far the easiest way to activate devices external to the computer, as it does not require an electrical interface of any kind.

After the following lines have been run, the relay may be energised by CALLing 30000 and then released again with CALL 30009. By quickly alternating between these commands, an annoying buzzing noise can be produced.

```
10 DATA 1,224,246,62,16,237,121,
201,1,224,246,62,239,237,121,201
20 MEMORY 30000
30 FOR count=0 TO 15:READ value:
POKE 30001+count,value:NEXT
count
```

The remote switch may be used to switch on and off other LOW POWER equipment (such as a small light or radio), via the tape port, which is a multi-pinned circular socket on the left hand side of the 6128. The two pins we are concerned with are the top two (disregarding the ring around the outside), numbered as 1 and 3 in the manual.

Brad Taylor, Girrawheen, WA

We didn't try Brad's discovery - we have enough buzzing noises in the office without adding any more!

Prior to buying my PCW I had a Commodore Pet and twin disc drive (5.25"), and over a long period I collected many discs and games etc., all of course quite useless to me.

I am a retired person and use my PCW purely as a word processor with LocoScript. I am not into CP/M and I have no desire to venture into it. (I'm too old).

However, I am asking for help in one matter. On my belated Commodore Pet I did have, amongst others, a game called "Yahtzee" which my wife and I spent many hours playing, and we sadly miss the game. Is there any one of your readers who could help me? Surely among all those many PCW owners out there, there must be someone who can help.

If there is, I will happily send them a formatted disc for a copy of it and pay any expenses involved.

Alternatively, is there any way of converting my Commodore Basic 3 discs for use on the PCW? If there is, I have lots of programs that perhaps other could use and I would willingly give them if I could get copies of some of them that I could use.

G. Bostock, Port Macquarie, NSW

An Amstrad CPC version of Yahtzee adapted by Alf Azzopardi was published in issue 15 (April '86). If any PCW owners have converted it, we would like to here from you.

Nationwide User Groups

There are quite a few changes in the listing this month - thanks for letting us know. Belatedly, we mention that Mrs. Beryl Schramm can be contacted on (079) 738035 for starting a group in **Boyne Island** (Qld), Don Cottrell for a group in **Bribie Island** (Qld) on (075) 488158, Nick Rogers on (068) 641170 for a group in **Bogan Gate** in NSW, R. Kernebone on (050) 233708 for a group in **Mildura** (Vic), and Neville Eriksen on (079) 782418 for a group in **Gladstone**.

WESTERN AUSTRALIA

ALBANY AMSTRAD USER GROUP

President: Gerry Barr (098 41 6884)
Secretary: Steven Hands (098 41 5183)
Treasurer: Gavern Grose
Venue: Priess Street Centre, 14 Priess Street, Albany on the first and third Mondays of each month at 7.00 pm.

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 further 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 Avenue, 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: Wandarrah Hall, Edgeware Street, Lynwood every 2nd and 4th Wednesday of each month from 7.00 pm.
Mail: The Secretary, 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: Peter & Carolyn 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: Geoff Martin
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: Chris Sowden (08 295 5923)
Vice Pres: Frank Matzka (08 382 2101)
Treasurer: Les Jamieson (08 356 9612)
Venue: The Church Hall, 15 Clayton Avenue, Plympton between 6.30 p.m. and 9.00 p.m. each Tues.
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 8.00 pm. 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 Centre, 370 The Terrace, Port Pirie every 2nd and 4th Monday from 7.30 pm.
Mail: The Port Pirie Amstrad User Group, c/o D.T. Green, 207 Senate Rd., Port 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 1820)
Venue: Casuarina Library, Darwin at 8.00 p.m. every 2nd and 4th Monday.

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: John Holmes (03 434 1607)
Venue: Hall at the corner of Church and Somerset Streets, Richmond on the first Sunday of each month commencing at 1.00 p.m. and generally twelve days later on a Friday evening starting at 7.00 p.m.

EASTERN AMSTRAD USER GROUP Inc.

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 Centre, 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

MACEDON RANGES AMSTRAD USER GROUP

Contacts: Wayne Urmston (03 744 2719)
Ken McMaster (054 22 2620)
Venue: Contact above for more details.

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 pm. 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 Secretary, PO Box 100, Seaford, Vic 3198.

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.

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: Neale Yardley
Secretary: Steven Walker (062 58 2323)
Editor: John Ault (062 47 5747)
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 fourth Wednesday of each month at 8.00 p.m.

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 pm. on the first Friday of each month.

FAIRFIELD MICRO USER GROUP
Contact: Arthur Pittard (02 72 2881)
Venue: Room 65, Canley Vale High School, Prospect Road, Canley Vale every third Wed. from 7.00.

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

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 pm. evey third Saturday.

LISMORE DISTRICT AMSTRAD COMPUTER CLUB
President: Max Muller (066 337 113)
Vice Pres: Nick Van Kempen (066 874 579)
Sec/Treas: Chris Rosolen (066 219 754)
Venue: Goonellabah Public School, Ballina St. on the last Tuesday of each month from 6.30 pm.
Mail: PO Box 88, South Lismore, NSW 2480

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 7.30 pm.

NAMOI AMSTRAD USERS GROUP
Contact: Martin P. Clift, JP (067 92 1333) B/H (067 92 3077) A/H
Venue: Narrabri Technical College, Barwan Street, Narrabri on the first Saturday of each month at 2.00 p.m.

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: John Joseph (02 331 2717)
Treasurer: David Springett (02 660 4515)
Venue: Auburn Public School, Adderley St., Auburn every second Tuesday of the month at 7.30 pm.
Mail: PO Box 1879, North Sydney, NSW 2060.

PORT MACQUARIE AMSTRAD USERS GROUP
Mail: Craig Tollis, Box 584, Port 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
Secretary: Bob Ashe (07 355 5699)
Treasurer: Ivan Dowling (07 269 8795)
Tech. Editor: Franz Hendrickx (07 356 0633)
Tech. Lib.: John Wotton
Venue: Main meetings at in Room 15a of Junction ParkState School, Waldheim St., Annerley starting at 7.30p.m. Another is held at Wynnum Central State School, Florence Street, Wynnum Central on the first Saturday of each month at 1.00p.m. The co-ordinator is Warren Kennedy (07 351 4232). A third is held at Newmarket State School, Banks St., Newmarket on the second Saturday of each month at 1.30p.m. The co-ordinator is Cherry Shrier (07 351 6179).

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
Contact: Graeme Annabell (079 27 4915)
Venue: Waraburra State School, Gracemere on the first Friday of each month at 7.00 pm.

COMPUTER USER GROUPS OF AUSTRALIA
Pittsworth Branch
President: David Siebuhr
Contact: Ron Langton (076 931 690)
Venue: Every 1st Tuesday of each 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 Maclaren
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 pm, alternating between the Hervey Bay Senior College and Maryborough TAFE College. Contact the above for more details.
Mail: Les Patford, PO Box 24, Torquay, Qld 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: Pricilla Thompson (076 35 5092)
Secretary: Adrian Dunsmore (076 91 1561)
Treasurer: Edwin Gerlach (076 33 1054)
Junior Pres: Troy Achilles (076 34 3460)
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: Allan Maddison (077 79 2607)
Secretary: S. Crawshaw (077 73 3933)
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
Publicity 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: Paul Gerard (003 34 0441)
Treasurer: Russell Lockett (003 44 8972)
Secretary: Andrew Blazely (003 93 1687)
Publicity: Marie Griffiths (003 93 6568)
Venue: Launceston Community College (opposite Park Street) in Room 11 on the first Saturday of the month at 5.00 p.m.

N.W. COAST AMSTRAD USER'S CLUB
President: Rick Ferguson (004 31 6280)
Treasurer: Robert Simpson
Secretary: Karen Stevenson
Venue: Burnie Technical College, Mooreville Road, Burnie on the third Friday of each month at 6.30 p.m.

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: PO Box 23.082 Bishopdale, Christchurch, NZ.

WELLINGTON AMSTRAD USER GROUP
Contact: Tony Tebbs 791 072 (evgs)
Venue: Room 718, Kirk Block, Victoria Univ. on the last Wednesday of each month from 7.30 pm.
Mail: PO Box 2575, Wellington, New Zealand.

The closing date on amendments to this list for Issue 36 (December 1987) is 21st October 1987

The PCW9512 - It's official

All the rumours and speculation about this addition to the PCW range came to a halt with the official release in the UK last month.

If you read last month's issue, you would have seen our report giving the most plausible rumours abounding at the time. It was almost right - the surprise came with the announcement of the disc drive configuration. The double density drive B (which was expected to be removed) has been promoted to an A drive.

So, what else is fact? For a start, out goes the green screen and in comes a paper white high resolution monochrome monitor - the same resolution as the present PCW's but white on black instead of green on black.

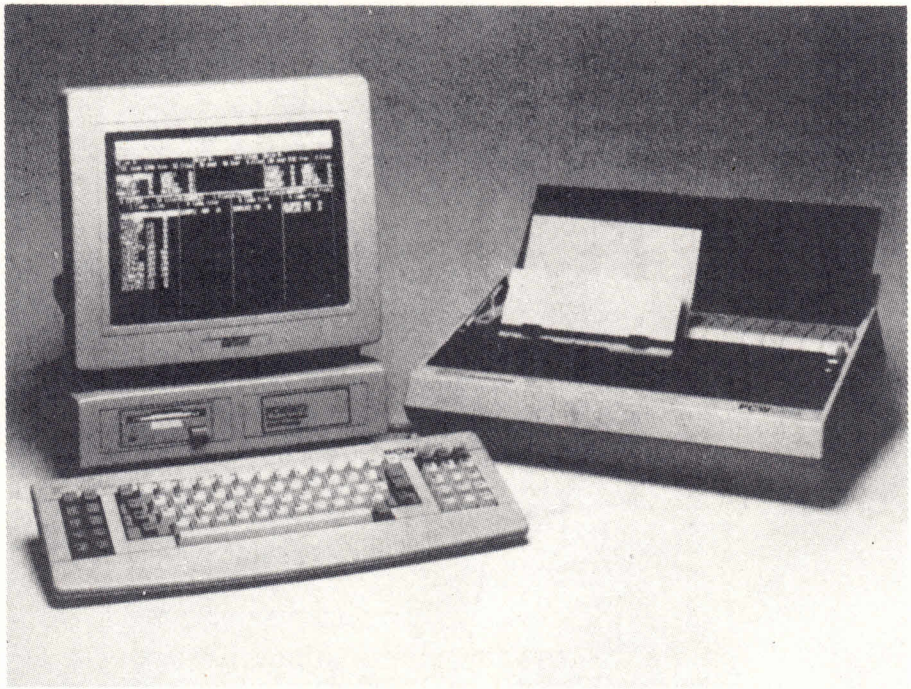
The disc drive, as already mentioned, will store 720k on the main drive into which your start of day disc goes. The standard PCW9512 will only have one disc drive, but there is an option of a second identical drive, which with the standard 512k M drive makes for a powerful machine with a lot of storage capacity - 4 times as much as the IBM PC! Amstrad state that it will be compatible with software on the present PCW A-drive single density discs in much the same way as the present B drive on an 8512 can read an A-drive disc. So anyone wanting to upgrade and carry on using their old data files can breath a sigh of relief. The drive(s) are housed 'PC style' in a unit under the monitor.

Another change towards the PC style is the keyboard. It's more spread out and the function keys [+] [-] [ALT] [EXTRA] [PTR] and [CAN] are divided off into a separate section at the left.

Amstrad have tackled the problem of the PCW printer by including what they describe as a 'high quality daisywheel printer' with bold, double strike, underline, superscript and subscript. It is claimed to print at 20 characters per second, about equivalent to the Near Letter Quality speed of the 8256/8512 dot matrix printer. One advantage will be the 15 inch printer which will allow for paper up to 15.5" wide, such as A4 on its side - a feature that will suit accounting and legal departments. The printer fits into the PCW in the same way the old PCW dot matrix does and depends on the hardware in the PCW - in other words you cannot use it with another computer. But if you cannot live without a fast dot matrix printer for drafts, there is an inbuilt Centronics parallel interface into which you can plug your favourite printer.

The PCW9512 will come bundled with LocoScript2 plus the LocoSpell spell checker and the LocoMail mail merger - obviously aiming more directly at the office environment.

As reported last month, importers Mitsubishi Electric AWA advise that the 9512 will not be available in Australia until early in 1988. No Australian price has been announced.



The PCW 9512 - released at the London Personal Computer World Show in September

SAGE/PCS on show

Personal Computer Software, exclusive distributors of Sagesoft products, were available for discussions and demonstrations of the Sage PC software range at the ACE '87 Exhibition in Melbourne at the beginning of September.

Among the products on show was a new range of low-cost PC learning software. The competitively priced software covers:

- PC Foundation (MSDOS & PC Basic)
- PC Accounts
- PC Spreadsheets
- PC Database
- PC Telecommunications
- PC Wordprocessing
- PC Touch Typist

Each product is designed to provide an interactive self-teaching software package related to the relevant Sage PC software program. Each package is presented in an A5 plastic folder, with

disc and slim workbook. They will retail at $89 (including tax).

Also on display, a world premiere in fact, was SAGENET - a local area networking package. Consisting of both hardware and software, this product is aimed at the low cost business computer market. The starter pack includes 2 boards, cables and software. It runs all Sage Networked Accounting Systems, is NetBios compatible and can handle up to eight users with no speed degradation. The proposed recommended retail price is $1150 (including tax).

Accountant, Accountant Plus and Financial Controller are now available in networking versions, and run on SAGENET or any other MSDOS compatible network. Their recommended retail prices (including tax) are $900, $1200 and $2500 respectively. More details can be obtained from PCS on (02) 923 2899

Campaign against child abuse

In 1986 the UK Software industry, through Softaid, raised over $700,000 for the now well publicised Live Aid movement.

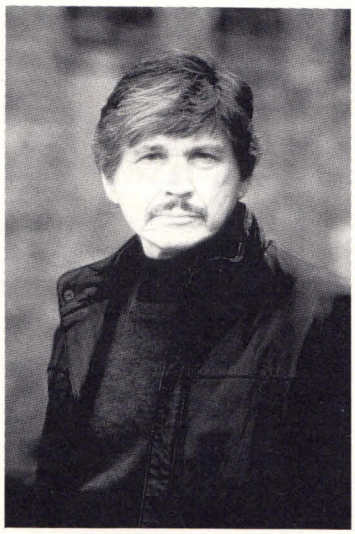
This year, by way of a campaign called BACK (Battle Against Cruelty to Kids), the organisation hopes to raise money to assist in the establishment of a network of special child protection teams under the supervision of both the National Society for the Prevention of Cruelty to Children and the Royal Society for the Prevention of Cruelty to Children. The money will be raised from sales of donated entertainment software which will be available as a BackPack series for different machines.

The Amstrad CPC compilation, which will contain Xeno, Deactivators, Marsport, Monty on the Run, Night Gunner, Knightshare, Nomad, Tempest, Starion and Starstrike, will sell for under £10.

Whether the pack will make Australia is not known, but it may sow the seeds of a similar worthwhile campaign to benefit less fortunate young Aussies.

Forthcoming attractions from Gremlin for CPCs

Death Wish 3



The computer software rights to the blockbuster film epic DEATH WISH, starring Charles Bronson, have been snapped up by top software development house GREMLIN.

Bronson plays Paul Kersey, New York's own unique brand of justice-fighter who's out to rid New York of the punks and creeps who infect the streets. Convinced that the Chief of Police is powerless against the underworld, Kersey adopts his own attack strategy. Armed with his famous 475 Wildey Magnum, a pump-action shotgun, a machine gun and a rocket launcher (I'm surprised he can move around - Ed), Kersey means business, and Gremlin's development team are currently in the process of capturing this outstanding story on screen.

Gremlin say "this is one of the most exciting ventures the we have been lucky enough to work upon so far. It offers our creative team a fantastic base from which to produce and outstanding, challenging game, featuring incredibly lifelike animation".

We wait with bated breath.

Coconut Capers

Another game from Gremlin, this time starring "Jack the Nipper". Jack and his family have been deported to Australia, but fortunately for us, he doesn't like the idea so parachutes from the plane, quickly followed by his father, into a tropical jungle.

True to form, Jack tries hard to be as naughty as possible which lasts for as long as he can keep away from his pursuing father. A special surprise awaits Jack if he reaches 100% on the Naughtyometer.

The Plonker

A rather strange name for a new product which appears to have taken the UK by storm. The Plonker is a small device which holds 4 discs and can be easily attached to the side of your computer keyboard or monitor. The idea is to raise them above the area of possible damage due to coffee spills and the like. Within the first three days of its release at the Amstrad Computer Show (don't they have a lot in the UK!) over 1000 units were sold.

An Australian distributor has been selected but to date has not made themselves known.

The Thingi

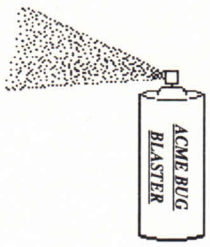
Another innovative idea which allows you to suspend any paperwork from the side of your monitor.

We've been using them in the office here at The Amstrad User now for a couple of months and can honestly say we can see things in a different light. Even neck-aches seem to have disappeared.

So convinced are we that everyone should have one (even for Mum as a 'memory board' attached to the fridge), we have purchased a number to be supplied, post free, at $24.95 each. See the inside front cover for details.

Bug-Off

How to handle common programming errors



You've typed in the listing and it doesn't work. You've spent hours scanning it and can't find any errors. You've had the whole family read the listing to you while typing it in. It still doesn't work. You've put a rope around your neck and are about to kick the stool.... Stop! First take a look at this article. It gives you hints on finding those elusive bugs and points out ways to avoid making mistakes in the first place. Why won't it work?

When you get "Syntax error" or similar message the first thing you tend to do is blame the listing, saying it is incorrect. More often than not you'll find, if you re-check your typing, that the error was of your making. True, there are occasions when printed listings are incorrect, but very rarely.

If you are copying a program from a magazine or book then here are some points to beware:

- copy exactly what you see
- take care with your spelling
- watch punctuation
- don't leave lines out
- make sure spaces are in the correct places
- assign variables correctly
- don't confuse letters with lookalike numbers.

COPY CORRECTLY

It is easy to type in commands, numbers, variables, data or anything else incorrectly. Arnold understands only a limited vocabulary of commands. If you invent new words you'll get the reply "syntax error." You get similar rude remarks if you splice two commands together or leave out spaces or hit a semicolon instead of a colon. It is so easy to fool you computer that you must be extremely careful what you type in. For example, the innocent-looking line:

```
10 IF B=0 THEN PRINT a$;" ";
   ELSE GOTO 80
```

First point: don't alter the line number (or for that matter any part of the program). If you do, then there's every chance that the program will not operate properly.

Notice the statement B=0. Certain letters and numbers, on some printers, look deceivingly similar. Confusing Bs and 8s, Os and 0s, Ls and 1s is very common. Type SLOWLY, and if you're not sure whether a character should be, say, A or B then mark this down on a piece of paper. First try one character. If an error results use the other.

Use spaces. It is better to have too many than too few. This is the only way the computer can distinguish between one command and another. Remember every command or statement you use must have a space either side of it. If you leave out spaces, for example, and type IFB=0 then, you've guessed it, "Syntax error" will pop in for tea. The same happens when you join two commands together, such as, THENPRINT. This is not true for all dialects of Basic. It is with Locomotive Basic, though, so take care.

If your spelling is as good as mine, beware. The computer can be very cruel. You must be word-perfect with the commands you use. If you enter PRIT instead of PRINT the computer will be most unforgiving. You are quite lucky with Locomotive Basic, as it displays command words in capitals even if you typed them lower-case. This makes spotting the mis-spelt command much simpler.

PERNICIOUS PUNCTUATION

You'd be surprised how easy it is to use a semicolon (;) instead of a colon(:) and vice-versa. Semicolons are common in PRINT and INPUT commands. Their main use is to join strings of characters together on the same line. For instance, to print "this that" with a change of border colour between the words:

PRINT "this ";: BORDER 0: PRINT "that"

Leave out the semicolon and "that" will be printed on the line below "this".

Colons have an altogether different function. Leaving them out is more likely to induce an error message. They tell the computer that one set of commands has finished and another follows. In general you'll find a command followed by a colon, another command then a colon... Only a few commands can be coexist in one line without colons: IF, AND, OR and THEN are examples.

Other characters that can be confused are the fullstop (.) and comma(,); the minus sign (-) and underline (_); the apostrophe(!) and grave accent or open-quote (`)

MEANINGFUL MESSAGES

Often the computer throws up a message that can be confusing or, worse, misleading. Sometimes it computer will say the error is in line such-and-such when quite often it's elsewhere. Here's a guide to the more usual errors.

SYNTAX ERROR

The commonest of errors. This is probably the one message that tells you the actual line the error occurred in. If you have read earlier paragraphs then you should be in a position to judge why the error occurred.

UNEXPECTED NEXT

You have placed a NEXT too few or a FOR too many in the listing. Each FOR a must have a NEXT a to compliment it. Note that it is not actually necessary to specify the variable like a after the NEXT - this can cause problems when searching for the error. Take care that you don't nest loops incorrectly. If you have loops within loops, the first one you entered

must be the last you leave.

Correct nesting:

```
FOR a=1 TO 5: FOR b=2 TO 6: NEXT b:
NEXT a
```

Incorrect nesting:

```
FOR a=1 TO 5: FOR b=2 TO 6:  NEXT a:
NEXT b
```

UNEXPECTED RETURN

Similarly, if you have used a GOSUB in your program then there should be a RETURN to match it. However, if you use a RETURN without previously defining a GOSUB then this message will appear.

DATA EXHAUSTED

You have tried to READ data that does not exist. Or the computer cannot find any more data to read. Check your DATA statements carefully. If there are too few or you have placed a fullstop instead of a comma to separate items then you'll get an error.

IMPROPER ARGUMENT

This if the message that can bring tears to your eyes. It is the most difficult to find and cure. Rarely does the line thrown at you contain the error. Reading data, it can happen when you try to define characters while converting strings to numerical values... The only way to rid yourself of this one is to be cautious when copying DATA lines or assigning values to variables.

TYPE MISMATCH

Your Amstrad has encountered a string when it was expecting a number or vice-versa. This error occurs commonly when reading data. You'll generally get a message like "Type mismatch in line 20." There is no error in line 20; that's just where the program was when it tried to use the data. The mistake will be within a DATA statement.

WHAT NEXT?

This little lot should help you when debugging. Of course, there are numerous other errors that can creep into a listing, but this article should point you in the right direction. Next month (but no promises) we'll look into DATA statements. Most errors are made there. They are also the hardest to find.

Program made Plain

Every so often, I shall look closely at one program - at its merits, at how it can be improved, and at what certain commands do. The listing need not be lengthy, just well structured, interesting and simple to follow. Remember the Card Trick program in the August issue - I said that logic plays an important part in computer programing? Well, this month's listing, WEEKDAY, is riddled with ANDs, ORs and IFs. Enough waffle - let's find out how the program works.

WEEKDAY

A clever listing from Gary Nugent of Churchdown determines the weekday of any given date. This works with any year - BC as well as AD. Enter years BC as negative numbers, although the day and month remain positive.

There was no year 0. The mathematical concept of zero displaced Roman numerals many centuries after Constantine's decree that years were to be numbered from the birth of Christ - which was about 300 years ago by then, and they were out by at least four years. Anyway the years jumped from 1 BC to 1 AD. The program accounts for this and leapyears. It also works for both Gregorian and Julian calendars.

Enter the date in the form DAY, MONTH, YEAR. For example, to find out the weekday of June 23, 1986, then enter 23,6,1986 at the prompt. Year must be in full: typing 86 will land you near Nero.

REMEMBER TO REM

```
1 ' Weekday
2 ' by Gary Nugent
3 ' The Amstrad User Sep 87
```

Label your listing. Identify what each section does. It makes it easier for everyone concerned. Gary's program is short and therefore does not need many REMs. Indeed the majority here are the credits at the top of the listing - which will remind you six months from now where to find all these explanations.

```
110 DIM day$(6)
120 FOR i=0 TO 6: READ day$(i): NEXT i
130 MODE 1: BORDER 0:INK 0,0:INK 1,26:INK 2,24:INK 3,6:CLS
140 WINDOW #0,1,40,2,25: WINDOW #1,1,40,1,1
150 PAPER #1,1: PEN #1,3: CLS#1: PRINT#1,SPC(16);"Weekday"
160 LOCATE 3,7: PAPER 3: PEN 1: PRINT"Enter date: ";
```

RESTORING YOUR WORK

The first line of Gary's program, DIM day$(6), would be necessary on many home micros. Not, however, the Amstrad. Why? Well, as you'll remember from out previous discussion of arrays, Arnold automatically dimensions a 10-item array. There is no need for you to do this. Gary has dimensioned a string array capable of holding seven items - superfluous, although it does make the listing more intelligible.

Line 120 loops from 0 to 6. As it does this, it READs in an item of DATA and places it in one of the free spaces provided by the array, day$. The data it reads is held in lines 270 and 280. The computer automatically searches out the data so you don't have to

specify where it is. The only occasions when you'd need to tell the computer where the data lies (with the command RESTORE line-number) is when you have lots of it scattered throughout the program, or when some of it is to be used more than once in the same run. The RESTORE command can also simplify debugging. Imagine you had loads of READs and DATAs in your listing. How could you tell which set of data was being read? You couldn't. Use RESTORE line number. In this case it would be RESTORE 270. If you peer at line 270 you'll notice "Sunday,Monday,Tuesday..." We now know what day$ does - it contains the names of the days of the week.

All the other lines in the section deal with setting up the colours and screen mode. Windows are also defined to highlight certain bits of text. They act like individual screens. The parameters following WINDOW are window number (between 0 and 7), left side of window, right side, top of window and bottom. You must place a hash(#) before the window number. Once a window has been defined, you can PRINT #windownumber, INPUT #windownumber, PAPER #windownumber, CLS #windownumber... Very useful.

LOGICALLY SPEAKING

```
170 PAPER 0: PEN 2: INPUT" ", day, month, year
180 IF month>2 THEN yy=year: mm=month ELSE yy=yedar-1:mm=m
onth+12
190 y1=year-INT(year/4)*4: y2=year-INT(year/100)*100:y3=ye
ar-INT(year/400)*400: ly=(y1=0 AND y2<>0 OR y3=0)
200 IF year=0 OR month=0 OR month>12 OR day=0 OR day>31 OR
 ((month=4 OR month=6 OR month=9 OR month=11) AND day>30)
OR (month=2 AND day>28-ly) THEN LOCATE 3,9: PAPER 2: PEN 0
: PRINT" Invalid Date ";CHR$(7): GOTO 260
210 IF day>4 AND day<15 AND year=1582 AND month=10 THEN LO
CATE 3,9: PAPER 2: PEN 0: PRINT" Date did not exist ";CHR$
(7): GOTO 260
220 IF year>1582 OR (year=1582 AND month>10) OR (year=1582
 AND month=10 AND day>14) THEN a=INT(yy/100):b=2-a+INT(a/4
) ELSE b=0
```

After you have told it the day, month and year with INPUT, the program, starting from line 180, will set up some variables according to your input. The expression y1=year-INT(year/4)*4 in line 190 can be simplified to y1=year MOD 4. The other formulae in the line could similarly be reduced in size - and, by the way, in execution speed. All do the same thing: the answer is the remainder after dividing year by a certain number. if you see, say 7 MOD 2 or 7-INT(7/2)*2 you should be able to determine the result, 1, without too much trouble.

The reason for all this kerfuffle is to check if year is a leap year. Leap years occur if the year divides by four. But there are exceptions (and this is the correction the Gregorian calendar made on the Julian, because it was found that the year is not 365 1/4 days long but 365 days, 5 hours, 48 minutes and 46 seconds). Years that divide by 100 are not leapyears unless they also divide by 400.

That's the meaning of the last statement in line 190:

```
ly = (y1=0 AND y2<>0 OR y3=0)
```

Looks a rather odd equation at first, doesn't it? Puzzle through the right-hand side by thinking "true" or "false" for each part. Then convert a final result to a number: "true" is -1. "false" is 0. So ly ends up as 0 or -1 depending if year was normal or leap. In other words, there are two possibilities that produce a leapyear.

- y1 (which is year MOD 4) equals zero at the same time as y2 (which is year MOD 100) does not equal zero
- y3 (which is year MOD 400) equals zero.

In these cases the result for ly will be -1 (a leapyear). If any of these results prove to be opposite then YEAR is normal. Phew!!

As you can see from the last paragraph, it is much easier to manipulate logic by using mathematical shorthand than by waffling on in English.

THE POPE'S PATCH

Lines 200 and 210 aren't much easier to follow. They both check that the date you entered exists and is legal. The first line checks that neither the year nor month equal zero or the month is greater than 12. It then looks to see that the date is not equal to zero or greater than 31 or greater than 30 if the month happens to be 4, 6, 9 or 11. If any of these possibilities fall true or the month is 2 (February) and the day greater than 28 (29 for a leapyear) then the program throws up the message "Invalid Date."

Pope Gregory XIII made life interesting by decreeing that in 1582 ten days would be skipped to get the calendar back into step with the seasons. Some poor sod will have missed his birthday, because 15 October came immediately after the 4th that year. Line 210 makes sure that you haven't typed one of these dates in.

Complication. Lines 210 and 220 above are correct for Catholic countries. But Protestant and Orthodox countries protested. This resulted in considerable calendar confusion for centuries. Not until September 1752 did England follow suit - and by then had to lose 11 days to match up. Russia stuck to the old calendar till after 1917, which is why the October Revolution took place in November. If you wish the listing to work for historical British dates then alter as follows:

```
205 REM Calendar correction in England
210 IF day>2 AND day<14 AND year=1752 AND month=9 THEN LOC
ATE 3,9: PAPER 2: PEN 0: PRINT" DAte did not exist "; CHR$
(7): GOTO 260
220 IF year>1752 OR (year=1752 AND month>9) OR (year=1752
AND month=9 AND day>14) THEN a=INT(yy/100): b=2-a+INT(a/4)
 ELSE b=0
230 jd=INT(365.25*yy-0.75*(yy<0))+INT(30.6001*(mm+1))+INT(
day)+1720996+b
240 dd=ROUND((jd/7-INT(jd/7))*7)
250 LOCATE 3,10: PAPER 2: PEN 0: PRINT" That was a ";day$(
dd);" "
260 PAPER 0: PEN 1: LOCATE 1,23: END
270 DATA Sunday, Monday, Tuesday, Wednesday
280 DATA Thursday, Friday, Saturday
```

WHAT DAY?

Finally, lines 230 and 240 calculate the weekday from the data you typed in earlier. Line 250 prints the weekday and line 260 returns you to the Ready prompt.

Riley's"believe it or not" Syntax Errors

Philip Riley adds to the discussion

It was late one night at about 12.30, I had been on the computer for several hours. I ran the program I was working on, it ran for a while then suddenly it stopped and the message SYNTAX ERROR IN 400 appeared on the screen. It was at this precise time that I am sure I heard a ghost of a laugh, well maybe not a laugh, more a nasty little giggle. I looked over my shoulder but no one was there.

To this day I don't know where that laugh came from in the dead of night in that empty house but I never go near my Amstrad after midnight anymore. I am of the strong opinion that when the computer prints SYNTAX ERROR IN 400 it is not only telling you of an error it is also laughing at you deep down in the CPU, and if you listen closely on a dark night you will hear it chuckle as it prints those terrible words that have sent many a programmer totally mad.

It was this experience that has led to the writing of this article (and hopefully others in the future) to help you to cut down on your computers enjoyment a little. So just what is the computer telling you. Basically it is saying Ha Ha Ha you have typed something wrong in your program. Not only does it tell you the line number that the error is on it then defies you to find the error by printing that line onto the screen all ready for you to edit.

Well you know it's in there somewhere so in you go undeterred. You look along the line but you see no error. You look again, still nothing. This is when your patience starts to run out (I might add that at this point the computer is chuckling to itself down there).

Don't give up, we have only just started. The first thing to look at is the various commands in the line (eg. GOTO, RETURN, IF-THEN). These should all be in upper case characters, if any are in lower case then this is probably the error. You may have left out a space or colon (:). Correct this and rerun the proggy.

If all of the commands are alright you can now check to see if you have left out any colons or maybe typed a semicolon instead of a colon. This is always a good one and can keep your computer amused for ages. Another good one is to count up any brackets in the line, first count all the opening brackets and then the closing brackets. The two numbers should be the same, if not you will have to bung one in somewhere (where you bung it in is up to you , but more often than not it's one on the end that's been missed).

If it is a particularly long line with many commands and colons and other assorted junk in it, you will need to use a different approach. Look along the line for the first colon. Type in STOP: after the first colon and run the proggy. If it is a long program that will take a long time to reach the line in question don't run the program just type in GOTO <line number>. If the statement BREAK IN <line number> appears on the screen then you know that the problem does not lie in that part of the line. If, however you get SYNTAX ERROR IN <line number> you will know that the error is in that part of the line.

If the error is not in the first part of the line take out the STOP: and insert it back in after the next colon. Type in run or GOTO and see what happens. In this way you will be able to track down the section of the program that error lies in.

If even this fails you, try listing the line onto the screen and retyping it into the computer underneath. You will quite often find that you will either a) find the mistake as you are typing the line in or b) when you run the program the line strangely seems to work. Either way it will shut your computer up and show it that you really are a force to be reckoned with.

Now as a little afterthought on the subject of errors. You have probably heard of the American Star Wars project. It entails shooting enemy missiles out of the sky from a satellite based laser. Did you know however, that the massive computer program that is being designed to operate Star Wars can only be tested in the event of a nuclear war taking place. I would therefore like you to picture this scene.

Big Ronnie Reagan has gone on national television and said that the Ruskies are nothing more than stupid hairy animals with big cold noses and bad breath. Now, how was the Russian leader supposed to know that Big Ronnie had got things mixed up again and was actually talking about Huskies. Well the Russians decide to declare war on America.

We declare war on you capitalistic pigs, says the Russian leader (there what did I tell you).

Enough thermo-nuclear warheads are launched from Russia and Eastern Europe to send the whole of America into orbit along with Starwars. Deep in the heart of the Rockies the strike is detected and the President informed. Make ready Starwars says big Ronnie on the phone. The message is relayed to a little man at a computer terminal deep down in the Rockies. He takes out a disc and puts it into the drive. He then types in RUN"disc". The disc drive jumps to life and a nice loading screen depicting a satellite shooting down a missile, with the words Starwars is now loading, please wait. Eventually the program is loaded and the message "PRESS SPACE BAR WHEN READY TO CONTINUE" appears on the screen.

The little man sits at the computer terminal drumming his fingers on the desk. Suddenly the phone rings, he answers and big Ronnie's voice says "GO STARWARS". The man presses the space bar, the screen clears. For what seems an eternity nothing appears to happen, until suddenly the words SYNTAX ERROR IN 220 appear on the screen.

And computer technology has the last laugh!

Filling a gap in Amsdos

from Martin Kimber

As a teacher who has been working in the field of microcomputers for some 7 years, and an owner of an Amstrad 6128 since their introduction, I would firstly like to say that I find this machine to be very good value for money, and a computer I recommend highly to my students.

However, I have always felt that Amsdos has one or two omissions without which the disc drives are not much better than high speed cassettes. Perhaps the most serious of these is the lack of random access files. Here is a list of my program 'RANFILE" which is an attempt to remedy this situation, a Basic program to generate the binary file "RANFILE.BIN". The program resides at &9000 and is loaded as follows:

```
MEMORY &8FFF
RUN "ranfile.bin"
CALL &9000
```

Thereafter the program can be invoked via 5 RSX commands. These are:

|OPENRAN,"filename" - this command opens a 32k file of the given name for use. Data in the file is not lost and if the file does not exist it will be created. Note that a file extension is not required - indeed, if one is supplied, it will be overwritten with the file extension .RAN

|FIELD,x - this command divides the file into 'fields' of a size x. The value of x must be a factor of 512, ie. 1,2,4,8,16....512 so that there is an integral number of fields per sector.

|PUT,string,x - this command puts the string into field x. Note that the length of the string should be of the length given in the field command, padded as required.

|GET,stringvar,x - this command gets the string from field x. Note that the length of the string should be preset to the length given in the field command.

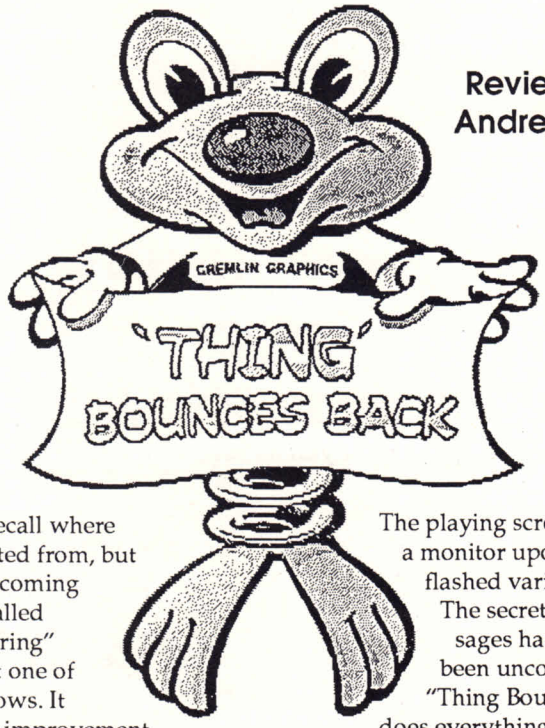
|CLOSERAN - this will save the last PUT data to disc and tidy up the 'house -keeping'. Note that the file name can be given if desired, but is not necessary.

I have incorporated some error trapping into the routine, so that common errors such as FIELDing an unOPENed file, or GETting from an unFIELDed file, for example, can be trapped, as can using the wrong number of parameters.

```
10 REM ranfile
20 REM by M H Kimber B Ed (Sc)
30 REM "BINGUNGUNYA", rmb 2262b
40 REM MARYBOROUGH, 3465
50 REM
60 loc=&9000
70 ln=990:c=&FE:mc=&DC
80 FOR i=&9000 TO &963C STEP 8
90 ln=ln+10
100 FOR j=i TO i+7
110 READ a$:a=VAL("&"+a$)
120 c=c XOR a:POKE loc,a:loc=loc+1
130 LOCATE 1,1:PRINT ln;"&"HEX$(loc)
140 NEXT:READ ch:IF c<>ch THEN MODE 2:PRINT"Chechsum ERROR
 in LINE";ln:END
150 mc=mc XOR c
160 NEXT:READ mch:IF mc<>mch THEN MODE 2:PRINT"Mastersum e
rror":END
170 SAVE "ranfile",b,&9000,&63D,&9000
1000 DATA 01,09,90,21,00,90,C3,D1,&C5
1010 DATA BC,1A,90,C3,35,90,C3,CD,&9B
1020 DATA 91,C3,FA,91,C3,00,92,C3,&30
1030 DATA 07,92,4F,50,45,4E,52,41,&A2
1040 DATA CE,46,49,45,4C,C4,50,55,&AB
1050 DATA D4,47,45,D4,43,4C,4F,53,&BA
1060 DATA 45,52,41,CE,00,3D,28,0D,&3A
1070 DATA 21,4F,93,C3,6F,92,E1,21,&39
1080 DATA 82,93,C3,6F,92,DD,56,01,&9C
1090 DATA DD,5E,00,1A,B7,28,E9,32,&41
1100 DATA B3,95,13,EB,5E,23,56,EB,&5F
1110 DATA 11,A1,93,E5,2A,B6,95,7C,&EC
1120 DATA B5,20,DB,D5,21,A1,93,36,&52
1130 DATA 20,11,A2,93,01,07,00,ED,&B9
1140 DATA B0,CD,34,92,D1,E1,7E,FE,&D2
1150 DATA 2E,28,11,E6,5F,12,13,23,&5E
1160 DATA 04,3E,08,B8,28,06,3A,B3,&73
1170 DATA 95,B8,20,EA,3E,81,CD,79,&9F
1180 DATA 92,3E,FF,DF,B1,95,3E,89,&80
1190 DATA CD,79,92,3E,03,DF,B1,95,&60
1200 DATA 3E,84,CD,79,92,DD,21,9D,&9D
1210 DATA 93,21,AD,93,11,00,00,DD,&DD
1220 DATA 7E,00,B7,CA,B2,91,4F,DF,&6D
1230 DATA B1,95,38,04,DD,23,18,E9,&7A
1240 DATA DD,7E,00,32,B8,95,FE,41,&79
1250 DATA 20,04,3E,02,18,05,FE,C1,&43
```

```
1260 DATA 20,01,AF,32,3C,96,3A,B8,&D7
1270 DATA 95,32,BB,95,21,BC,95,22,&74
1280 DATA AF,95,3E,02,32,AE,95,CD,&B6
1290 DATA 59,92,D2,B2,91,21,AD,93,&93
1300 DATA 23,7E,FE,E5,28,25,11,BB,&72
1310 DATA 95,37,EB,ED,52,38,10,EB,&47
1320 DATA 2B,7E,FE,E5,C4,40,92,38,&27
1330 DATA 4A,01,20,00,09,18,E1,3A,&86
1340 DATA BB,95,3C,32,BB,95,E6,07,&69
1350 DATA C2,E7,90,2A,B6,95,7C,B5,&1C
1360 DATA 20,34,7D,21,A9,93,36,2E,&76
1370 DATA 23,36,52,23,36,41,21,A1,&E5
1380 DATA 93,11,AD,93,06,0C,CD,BC,&12
1390 DATA BC,21,00,00,11,00,7F,CD,&2C
1400 DATA 98,BC,CD,8F,BC,CD,34,92,&9D
1410 DATA 3A,BB,95,E6,F0,3C,32,BB,&2A
1420 DATA 95,18,89,22,B6,95,11,10,&2E
1430 DATA 00,19,06,10,DD,2A,AF,95,&EC
1440 DATA C5,4E,23,E5,26,00,69,29,&C7
1450 DATA E5,7C,4D,11,09,00,CD,8A,&4C
1460 DATA 92,3A,B8,95,85,DD,71,00,&E0
1470 DATA DD,77,01,DD,23,DD,23,C1,&8A
1480 DATA 03,78,11,09,00,CD,8A,92,&3C
1490 DATA 3A,B8,95,85,DD,71,00,DD,&DF
1500 DATA 77,01,DD,23,DD,23,E1,C1,&89
1510 DATA 10,C6,E5,2A,AF,95,11,40,&FB
1520 DATA 00,19,22,AF,95,E1,3A,AE,&8F
1530 DATA 95,3D,32,AE,95,C2,F0,90,&8C
1540 DATA 18,06,21,77,93,CD,6F,92,&67
1550 DATA 3E,81,CD,79,92,3E,00,DF,&1F
1560 DATA B1,95,3E,89,CD,79,92,3E,&94
1570 DATA 10,DF,B1,95,C9,47,3A,B3,&78
1580 DATA 95,B7,20,06,21,5D,93,C3,&50
1590 DATA 6F,92,05,28,06,21,6B,93,&5F
1600 DATA C3,6F,92,DD,5E,00,DD,56,&69
1610 DATA 01,3E,02,0E,00,CD,8A,92,&8F
1620 DATA 47,7C,B5,20,E8,ED,43,B4,&D3
1630 DATA 95,C9,CD,BD,92,ED,B0,C9,&F9
1640 DATA CD,BD,92,EB,ED,B0,C9,2A,&4E
1650 DATA B6,95,7C,B5,20,05,21,5D,&FD
1660 DATA 93,18,5C,ED,5B,B9,95,7A,&CA
1670 DATA B3,28,0B,7A,32,3C,96,7B,&C3
1680 DATA 32,BB,95,CD,55,92,21,AD,&59
1690 DATA 93,36,00,11,AE,93,01,90,&41
1700 DATA 02,ED,B0,C9,21,A9,93,36,&FA
1710 DATA 52,23,36,41,23,36,4E,C9,&6E
1720 DATA E5,23,06,0A,11,A1,93,1A,&9D
1730 DATA BE,20,07,23,13,10,F8,37,&EB
1740 DATA 18,01,B7,E1,C9,3E,85,18,&CE
1750 DATA 02,3E,84,CD,79,92,21,AD,&DC
1760 DATA 93,1E,00,3A,3C,96,57,3A,&AC
1770 DATA BB,95,4F,DF,B1,95,C9,7E,&81
1780 DATA FE,FF,C8,CD,5A,BB,23,18,&5F
1790 DATA F6,32,AD,95,21,AD,95,CD,&77
1800 DATA D4,BC,22,B1,95,79,32,B3,&E1
1810 DATA 95,C9,21,00,00,06,10,CB,&41
1820 DATA 11,17,ED,6A,ED,52,30,01,&4E
1830 DATA 19,3F,10,F3,CB,11,17,C9,&8F
1840 DATA AF,ED,52,19,30,01,EB,B2,&EE
1850 DATA 37,C0,B3,5A,20,07,EB,C9,&F5
1860 DATA EB,19,EB,29,D8,1F,30,FB,&C9
1870 DATA B7,20,F5,19,C9,FE,02,28,&AF
1880 DATA 06,E1,21,8C,93,18,A8,3A,&FC
1890 DATA B3,95,B7,20,06,E1,21,5D,&D6
1900 DATA 93,18,9C,DD,4E,00,DD,7E,&F1
1910 DATA 01,ED,5B,B4,95,CD,8A,92,&B2
1920 DATA 47,79,FE,41,30,DB,7C,B5,&11
1930 DATA 20,04,0B,2A,B4,95,E5,21,&F1
1940 DATA BC,95,09,09,46,23,4E,E1,&12
1950 DATA ED,5B,B9,95,7A,B3,28,17,&7E
1960 DATA 7B,B9,20,04,7A,B8,28,20,&52
1970 DATA E5,C5,7A,32,3C,96,7B,32,&D9
1980 DATA BB,95,CD,55,92,C1,E1,E5,&38
1990 DATA ED,43,B9,95,78,32,3C,96,&5A
2000 DATA 79,32,BB,95,CD,59,92,E1,&D8
2010 DATA E5,ED,5B,B4,95,01,00,02,&A9
2020 DATA 78,CD,8A,92,67,69,D1,E5,&3E
2030 DATA 1B,CD,A0,92,11,AD,93,19,&EC
2040 DATA EB,DD,46,03,DD,4E,02,03,&0D
2050 DATA 0A,6F,03,0A,67,C1,C9,4F,&41
2060 DATA 50,45,4E,52,41,4E,20,65,&02
2070 DATA 72,72,6F,72,FF,46,49,4C,&A3
2080 DATA 45,20,4E,4F,54,20,4F,50,&AC
2090 DATA 45,4E,FF,46,49,45,4C,44,&1A
2100 DATA 20,65,72,72,6F,72,FF,44,&F9
2110 DATA 49,53,43,20,65,72,72,6F,&8A
2120 DATA 72,FF,46,49,4C,45,20,4F,&6E
2130 DATA 50,45,4E,FF,50,41,52,52,&DB
2140 DATA 41,4D,45,54,45,52,20,65,&94
2150 DATA 72,72,6F,72,FF,C1,41,01,&F7
2160 DATA 00,20,20,20,20,20,20,20,&D7
2170 DATA 20,52,41,4E,4E,00,00,00,&E4
2180 DATA 00,00,00,00,00,00,00,00,&E4
2190 DATA 00,00,00,00,00,00,00,00,&E4
2200 DATA 00,00,00,00,00,00,00,00,&E4
2210 DATA 00,00,00,00,00,00,00,00,&E4
2220 DATA 00,00,00,00,00,00,00,00,&E4
2230 DATA 00,00,00,00,00,00,00,00,&E4
2240 DATA 00,00,00,00,00,00,00,00,&E4
2250 DATA 00,00,00,00,00,00,00,00,&E4
2260 DATA 00,00,00,00,00,00,00,00,&E4
2270 DATA 00,00,00,00,00,00,00,00,&E4
2280 DATA 00,00,00,00,00,00,00,00,&E4
2290 DATA 00,00,00,00,00,00,00,00,&E4
2300 DATA 00,00,00,00,00,00,00,00,&E4
2310 DATA 00,00,00,00,00,00,00,00,&E4
2320 DATA 00,00,00,00,00,00,00,00,&E4
2330 DATA 00,00,00,00,00,00,00,00,&E4
2340 DATA 00,00,00,00,00,00,00,00,&E4
2350 DATA 00,00,00,00,00,00,00,00,&E4
2360 DATA 00,00,00,00,00,00,00,00,&E4
2370 DATA 00,00,00,00,00,00,00,00,&E4
2380 DATA 00,00,00,00,00,00,00,00,&E4
2390 DATA 00,00,00,00,00,00,00,00,&E4
2400 DATA 00,00,00,00,00,00,00,00,&E4
2410 DATA 00,00,00,00,00,00,00,00,&E4
2420 DATA 00,00,00,00,00,00,00,00,&E4
2430 DATA 00,00,00,00,00,00,00,00,&E4
2440 DATA 00,00,00,00,00,00,00,00,&E4
2450 DATA 00,00,00,00,00,00,00,00,&E4
2460 DATA 00,00,00,00,00,00,00,00,&E4
2470 DATA 00,00,00,00,00,00,00,00,&E4
2480 DATA 00,00,00,00,00,00,00,00,&E4
2490 DATA 00,00,00,00,00,00,00,00,&E4
2500 DATA 00,00,00,00,00,00,00,00,&E4
2510 DATA 00,00,00,00,00,00,00,00,&E4
2520 DATA 00,00,00,00,00,00,00,00,&E4
2530 DATA 00,00,00,00,00,00,00,00,&E4
2540 DATA 00,00,00,00,00,00,00,00,&E4
2550 DATA 00,00,00,00,00,00,00,00,&E4
2560 DATA 00,00,00,00,00,00,00,00,&E4
2570 DATA 00,00,00,00,00,00,00,00,&E4
2580 DATA 00,00,00,00,00,00,00,00,&E4
2590 DATA 00,00,00,00,00,00,00,00,&E4
2600 DATA 00,00,00,00,00,00,00,00,&E4
2610 DATA 00,00,00,00,00,00,00,00,&E4
2620 DATA 00,00,00,00,00,00,00,00,&E4
2630 DATA 00,00,00,00,00,00,00,00,&E4
2640 DATA 00,00,00,00,00,00,00,00,&E4
2650 DATA 00,00,00,00,00,00,00,00,&E4
2660 DATA 00,00,00,00,00,00,00,00,&E4
2670 DATA 00,00,00,00,00,00,00,00,&E4
2680 DATA 00,00,00,00,00,00,00,00,&E4
2690 DATA 00,00,00,00,00,00,00,00,&E4
2700 DATA 00,00,00,00,00,00,00,00,&E4
2710 DATA 00,00,00,00,00,00,00,00,&E4
2720 DATA 00,00,00,00,00,00,00,00,&E4
2730 DATA 00,00,00,00,00,00,00,00,&E4
2740 DATA 00,00,00,00,00,00,00,00,&E4
2750 DATA 00,00,00,00,00,00,00,00,&E4
2760 DATA 00,00,00,00,00,00,00,00,&E4
2770 DATA 00,00,00,00,00,00,00,00,&E4
2780 DATA 00,00,00,00,00,00,00,00,&E4
2790 DATA 00,00,00,00,00,00,00,00,&E4
2800 DATA 00,00,00,00,00,00,00,00,&E4
2810 DATA 00,00,00,00,00,00,00,00,&E4
2820 DATA 00,00,00,00,00,00,00,00,&E4
2830 DATA 00,00,00,00,00,00,00,00,&E4
2840 DATA 00,00,00,00,00,00,00,00,&E4
2850 DATA 00,00,00,00,00,00,00,00,&E4
2860 DATA 00,00,00,00,00,00,00,00,&E4
2870 DATA 00,00,00,00,00,00,00,00,&E4
2880 DATA 00,00,00,00,00,00,00,00,&E4
2890 DATA 00,00,00,00,00,00,00,00,&E4
2900 DATA 00,00,00,00,00,00,00,00,&E4
2910 DATA 00,00,00,00,00,00,00,00,&E4
2920 DATA 00,00,00,00,00,00,00,00,&E4
2930 DATA 00,00,00,00,00,00,00,00,&E4
2940 DATA 00,00,00,00,00,00,00,00,&E4
2950 DATA 00,00,00,00,00,00,00,00,&E4
2960 DATA 00,00,00,00,00,00,00,00,&E4
2970 DATA 00,00,00,00,00,00,00,00,&E4
2980 DATA 00,00,00,00,00,00,00,00,&E4
2990 DATA 00,00,00,00,00,00,00,00,&E4
3000 DATA &D8
```

Martin also sent us an assembly listing of his program which runs to nine pages - too long to publish!
However, if you would like a copy, please drop us a line enclosing a self-addressed envelope of a size capable of taking 9 A4 sheets, duly stamped (68¢ for within Victoria, 78¢ other states or 95¢ airmail). Send your request to: The Amstrad User, 1/245 Springvale Road, Glen Waverley, Victoria 3150.



Reviewed by Andre Urankar

I don't exactly recall where "Thing" originated from, but I can remember coming across a game called "Thing On A Spring" early last year at one of the Amstrad Shows. It seemed to be an improvement on the "ladders and ramps" concept and offered a new mode of game play.

Thing Bounces Back (TBB) continues in this genre, but also adds some new twists: slippery slides, air streams and vacuum tubes. These move our little hero (Thing) to other parts of the playing area. Thing can also hop and somersault to move from location to location.

Thing's objective is to collect the four parts of a computer program scattered around the factory. As usual, life was not meant to be easy, and there are a whole bunch of nasties just waiting to drain the oil out of Thing. Additional quantities of oil are hidden under stepping stones painted with a question mark. These are used as stepping stones when travelling over them, but when head-butted they burst to reveal either scoring points, quantities of oil, weights (which tend to deplete the oil reserves of Thing), or even an additional "heart". Thing has three lives at the start of the quest, and these are indicated as red hearts at the bottom right hand side of the screen. During each life the heart beats (changing colour) until the reserve of oil, indicated by the graduated scale on the right hand side of the screen, reaches zero. The heart then breaks, and the next life comes into play.

The playing screen also depicts a monitor upon which are flashed various messages. The secret of these mes sages has as yet not been uncovered.

"Thing Bounces Back" does everything almost right:

- a bouncy introduction theme
- a demonstration mode
- good and appropriate sound effects
- excellent movement in all directions
- action at a speed that both dad and the children can enjoy
- a PAUSE button
- a measure of strategy

BUT it lacks clear instructions as to the aim of the game. The enclosed instruction sheet indicates the various objects, nasties and collectables, but does not mention what needs to be done when those items have been collected. (Perhaps this is part of the strategy?)

The game can be played by both juniors as a simple "run around and collect whatever is available" type game, or by seniors as a "what do we do next " game. It deserves to do well.

Excuse me now while I go back to the game, and find out where the Listing Paper is hidden.

Thing Bounces Back is imported by ISD. It is available on tape at $29.99 or Disc at $39.99

Cheat Mode

Where invincibility is the name of the game.

TRIO FROM TWO WELLS

Martin West (Two Well, SA) has provided a couple of pokes and tips.
The first poke is for the disc version of ***Ghost and Goblins*** and will give you up to 255 lives.

```
10 MEMORY &17FF: LOAD "CODE
20 INPUT "How many lives";L
30 POKE &50A8,L
40 CALL &5000
```

The second is another disc poke, this time for ***Galaxia.*** It gives joystick control and 255 lives.

```
10 MEMORY &32FF: LOAD
"15:GALAXIA.BIN
20 FOR T=1 TO 6: READ A$,B:POKE
VAL("&"+A$),B:NEXT
30 CALL &3300,379:CALL &7A587
40 DATA 38DE,73,38AB,72,39EE,76,
3911,75,5A2A,47,35E0,255
```

Martin's first tip concerns ***Defend or Die.*** Apparently, if you get into a bit of trouble, slamming down the control, shift and caps lock keys together will make the aliens on the screen explode (never mind what it may do to your keyboard!). But you can only use this about three or four times.
His second tip is not really "Cheat Mode" material, but we couldn't think of anywhere else to put it. He advises that only 16 user numbers can be accessed through the |USER command, but by entering `POKE &A701,user number` you can access all 256. So, saving a program in a nominated user number would save anyone coming along and accidentally erasing or viewing it (eg.little brothers or sisters).

TERMINUS

A Mastertronic money-minimal Method 1 poke is up for grabs. Steve Hooper is the man to blame. Infinite lives, infinite energy and no more crunching crushers.

```
1 ' Terminus - tape
2 ' by Steve Hooper
3 ' The Amstrad User Oct 87
10 MEMORY 4863: MODE 0
20 INK 0,0: INK 1,0
30 BORDER 0: FOR a=0 TO 3
40 READ b,c: OUT &BC00,b
50 OUT &BD00,c: NEXT a
60 FOR a=0 TO 150: NEXT
70 PRINT"Please wait"
80 INK 1,26: LOAD"!bot"
90 LOAD"!": LOAD"!",4864
100 POKE 8354,0: POKE 10555,0
110 POKE 17347,&C9: CALL 4864
120 DATA 1,32,2,42,6,1,7,6
```

THRUST II

Phil "remember Thrust I" Howard has poked Firebird's follow-up. This poke is very similar to the poke he created for the original Thrust: press the Control key and the ship stops dead in its tracks. Phil has been improving his techniques somewhat. Not only does your ship halt, but so does the ball. Use Method 1.

```
1 ' Thrust II - tape
2 ' by Phil Howard
3 ' The Amstrad User Oct 87
10 DATA 21,38,bd,36,1b,21
20 DATA bc,bc,36,c3,23,36
30 DATA 16,23,36,be,c3,00
40 DATA a3,cf,88,88,21,2d
50 DATA 29,36,17,21,27,be
60 DATA 22,35,29,cd,13,be
70 DATA c3,28,23,21,00,00
80 DATA 22,48,69,22,4a,69
90 DATA 22,4c,69,c3,6a,27
100 y=0: MEMORY &5000
110 FOR x=&BE00 TO &BE35
120 READ a$: a=VAL("&"+a$)
130 POKE x,a: y=y+a: NEXT
140 IF y<>&11AE THEN 170
150 LOAD"thrust",&A300
160 CALL &BE00
170 PRINT"Data error"
```

BRUCE LEE

Phil Howard has been attacking a few oldies. The first of the bunch is US Gold's beat-em-up. A Method 1 entry provides you with a lot of Lees.

```
1 " Bruce Lee
2 ' by Phil Howard
3 ' The Amstrad User Oct 87
10 DATA 21,e2,39,36,c3,23
20 DATA 36,16,23,36,be,21
30 DATA 40,00,e5,21,00,bb
40 DATA e5,c3,b7,39,e5,21
50 DATA 74,02,36,2c,23,36
60 DATA 85,e1,f1,f3,c9,21
70 DATA f1,65,36,a7,c3,00
80 DATA 58
90 y=0: MEMORY &2000
100 FOR x=&BE00 TO &BE2A
110 READ a$: a=VAL("&"+a$)
120 POKE x,a: y=y+a: NEXT
130 IF y<>&122E THEN 160
140 LOAD"bruce lee",&A300
150 CALL &BE00
160 PRINT"Data error"
```

HUNCHBACK II

Another of Phil Howard's pokes that provides you with infinite lives. Method 1 is the way.

```
1 ' Hunchback II - tape
2 ' by Phil Howard
3 ' The Amstrad User Oct 87
10 DATA 21,e2,39,36,c3,23
20 DATA 36,16,23,36,be,21
30 DATA 40,00,e5,21,00,bb
40 DATA e5,c3,b7,39,e5,21
50 DATA 74,02,36,2c,23,36
60 DATA 85,e1,f1,f3,c9,21
70 DATA f6,84,36,a7,c3,68
80 DATA 42
90 y=0: MEMORY &2000
100 FOR x=&BE00 TO &BE2A
110 READ a$: a=VAL("&"+a$)
120 POKE x,a: y=y+a: NEXT
130 IF y<>&12A4 THEN 160
140 LOAD"hunchback II",&A300
150 CALL &BE00
160 PRINT"Data error"
```

ELITE

We have had such a comprehensive set of pokes for the disc version of Elite that we thought there was nothing else you could do to it.
Stephen Basford has proved us wrong. Add these two pokes as lines 151 (instant hyper space) and 152 (difficult game).

```
151 POKE &4C56,24
152 POKE &507F,0
```

BOULDER DASH

Alte de Boer has been at it again. This time he has redesigned Rockford's playing ground. Cave A has been completely rearranged. It is like nothing you have seen before. You are given a long time to complete the cave. You'll need it. The cave is tough. Enter the poke using Method 1 and get Rockford digging.

```
1 ' Boulder Dash - tape
2 ' by Alte de Boer
3 ' The Amstrad User Oct 87
50 FOR a=&A100 TO &A119: READ a$
60 POKE a,VAL("&"+a$): NEXT
70 MEMORY &2FFF: LOAD "!",&3000
75 CALL &A100
80 a=&8ED0: b=6: c=0: GOSUB 170
90 a=&8F01: b=5: c=&5F: GOSUB
170
100 a=&8F06: b=5: c=0: GOSUB 170
110 a=&8F0F: b=3: c=0: GOSUB 170
120 POKE &8EFA,2: POKE &8EFB,4
130 c=0: FOR a=&8F13 TO &8F8A
140 READ a$: b=VAL("&"+a$):
c=c+b
145 POKE a,b: NEXT
150 IF c<>&BFC THEN PRINT"Error
in data. Please check.": END
160 MODE 1: CALL &A10C
170 FOR x=a TO a+b: POKE x,c
180 NEXT: RETURN
190 ' Loading/relocating data
200 '
210 DATA 21,00,30,11,25,70,3E
215 DATA 87,cd,a1,bc,c9,21,00
220 DATA 30,11,00,02,01,25,70
225 DATA ed,b0,c3,52,1f
230 '
240 ' Boulder Dash cave data
250 '
260 DATA 25,01,03,90,03,03,24
270 DATA 0c,10,43,03,10,24,02
280 DATA 40,26,12,24,06,48
290 DATA 03,12,0f,02,42,26,10
300 DATA 03,06,02,03,10,3A,01
310 DATA 16,50,01,14,04,02
320 DATA 82,11,13,04,04,30,04
330 DATA 27,16,10,26,13,46,25
340 DATA 13,03,04,50,24,15
350 DATA 05,06,14,22,16,14,26
360 DATA 14,01,25,14,06,24,14
370 DATA 40,17,15,05,02,40
380 DATA 1D,14,02,02,00,15,14
390 DATA 40,26,05,05,05,14,22
400 DATA 07,14,24,05,54,15
410 DATA 0B,07,07,48,0B,0B,05
420 DATA 06,00,09,0a,40,12,0b
430 DATA 06,00,14,12,06,FF
```

OH MUMMY

The old Amsoft classic that features on many Amstrad advertisements has finally been poked. Owen Cunningham has done the dirty and supplied you with infinite lives. Just type in the listing and follow the Method 1 guidelines.

```
1 ' Oh Mummy
2 ' by Owen Cunningham
3 ' The Amstrad User Oct 87
10 MEMORY 15000
20 LOAD"mummy 1",&6000
30 POKE &760E,0:CALL &6000
```

VAMPIRE

Many lives are available for Codemaster's cheapie. Enter it using Method 1 and send your thanks to Phil Howard.

```
1 ' Vampire
2 ' by Phil Howard
3 ' The Amstrad User Oct 87
10 DATA 21,4f,bf,36,0b,23
20 DATA 36,be,c3,00,bf,21
30 DATA 5b,2d,36,34,c3,88
40 DATA 13,21,e2,39,36,c3
50 DATA 23,36,29,23,36,be
60 DATA 21,40,00,e5,21,00
70 DATA bb,e5,c3,b7,39,e5
80 DATA 21,91,01,36,d0,23
90 DATA 36,6f,23,36,cd,e1
100 DATA f1,f3,c9
110 y=0: MEMORY &2000
120 FOR x=&BE00 TO &BE38
130 READ a$: a=VAL("&"+a$)
140 POKE x,a: y=y+a: NEXT
150 IF y<>&171C THEN 170
160 LOAD"vampire": CALL &BE13
170 PRINT"Data error"
```

SUPERCYCLE

Another lengthy poke from Richard Monteiro. This time it's for the disc version of US Gold's Epyx game. Data is not written to the disc so you can write-protect it if you wish. The poke gives you an infinite time to complete each course - you should be able to visit all the tracks with this one.

```
1 ' Super Cycle - disc
2 ' by Richard Monteiro
3 ' The Amstrad User Oct 87
10 a=PEEK(&BE42): b=PEEK(&BE43)
15 c=256*b+a: FOR d=c TO c+24
20 POKE d,0: NEXT d: CAT
25 FOR t=&A000 TO &A153
30 READ a$: b=b+VAL("&"+a$)
35 POKE t,VAL("&"+a$): NEXT
40 IF b=34777 THEN CALL &A000
50 PRINT"ERROR IN DATA"
60 DATA AF,CD,0E,BC,F3,DD
70 DATA 21,33,A1,21,00,C0
80 DATA 22,A3,A0,3E,15,32
90 DATA E4,A0,3E,12,32,36
100 DATA A1,32,3C,A1,CD,78
110 DATA A0,DD,21,33,A1,21
120 DATA 40,00,22,A3,A0,3E
130 DATA 0A,32,36,A1,32,3C
140 DATA A1,3E,11,32,E4,A0
150 DATA CD,78,A0,21,40,00
160 DATA 11,00,80,3E,E5,AE
170 DATA 77,23,1B,7A,B3,3E
180 DATA E5,20,F6,3E,32,32
190 DATA CD,51,21,61,A0,11
200 DATA A1,52,01,17,00,ED
210 DATA B0,31,38,00,C3,8B
220 DATA 3E,20,44,49,53,4B
230 DATA 20,50,4F,4B,45,20
240 DATA 20,00,20,20,42,59
250 DATA 20,52,50,4D,20,20
260 DATA 3E,01,CD,1E,A1,01
270 DATA 7E,FB,11,30,A1,CD
280 DATA EC,A0,7E,B7,20,F6
290 DATA 11,33,A1,CD,EC,A0
300 DATA CD,24,A1,CB,6E,28
310 DATA F9,3A,4B,A1,DD,BE
320 DATA 03,20,EB,11,39,A1
330 DATA 21,00,50,CD,EF,A0
340 DATA 21,F9,FF,19,7E,E6
350 DATA 38,20,08,23,7E,23
360 DATA B6,E6,7F,28,16,CD
370 DATA 24,A1,11,43,A1,CD
380 DATA EC,A0,CD,24,A1,CB
390 DATA 66,20,F0,CB,6E,28
400 DATA F5,18,BB,2A,A3,A0
410 DATA 11,00,10,19,22,A3
420 DATA A0,DD,34,03,DD,7E
430 DATA 03,32,3C,A1,3D,FE
440 DATA 00,38,A3,AF,CD,1E
450 DATA A1,C9,21,4A,A1,E5
460 DATA 1A,08,13,ED,78,87
470 DATA 30,FB,87,38,05,1A
480 DATA 0C,ED,79,0D,3E,06
490 DATA 3D,20,FD,08,3D,20
500 DATA E8,ED,78,CB,67,28
510 DATA 0C,FE,C0,38,F6,0C
520 DATA ED,78,77,23,0D,18
530 DATA EE,EB,E1,C9,01,7E
540 DATA FA,ED,79,C9,11,37
550 DATA A1,CD,EC,A0,CB,5E
560 DATA C8,F1,18,89,02,4A
570 DATA 00,03,0F,00,00,01
```

```
580 DATA 08,09,46,00,00,00
590 DATA 41,02,49,2A,FF,02
600 DATA 07,00,FF,FF,FF,FF
610 DATA 00,00,00,00,00,00
620 DATA 00,00,00,00
```

HEAD OVER HEELS

Tony Hoyle has worked hard to give you infinite everything. There should be no excuse for not completing the game now. Use Method 1. You will find when you run the poke that the horn, bag and doughnuts are already in your possession.

Tony has fixed it so you have 255 of everything. You'll be treated to an odd display - all the counters will show ";;" - but don't worry, it's HOH's way of showing 255. As if that weren't enough to be going on with, the poke activates all the bonuses permanently. This means you'll be able to jump further, go faster and be invulnerable.

```
1 ' Head Over Heels - tape
2 ' by Tony Hoyle
3 ' The Amstrad User Oct 87
30 MEMORY &39AE
40 x=&BE80
50 READ a$
60 WHILE a$<>"xx"
70 POKE x,VAL("&"+a$)
80 x=x+1
90 READ a$
100 WEND
110 LOAD"",&39AF
120 CALL &3A6A
130 LOAD"!",&4000
140 CALL &BE80
160 DATA f3,21,00,40
170 DATA 11,00,bb,01
180 DATA f6,01,3e,8d
190 DATA cd,df,be,21
200 DATA 46,bb,11,46
210 DATA bb,01,b0,01
220 DATA 3e,c4,cd,df
230 DATA be,21,00,01
240 DATA 22,d6,bc,21
250 DATA 00,9f,22,d9
260 DATA bc,21,00,0a
270 DATA 22,e0,bc,21
280 DATA c0,0d,22,e3
290 DATA bc,21,00,00
300 DATA22,4c,bc,21
310 DATA c4,be,22,f4
320 DATA bc,c3,c0,bc
330 DATA 21,72,24,11
340 DATA 73,24,01,08
350 DATA 00,36,ff,ed
360 DATA B0,21,FD,BE
370 DATA 11,6E,41,01
380 DATA 06,00,ED,B0
390 DATA C3,00,01,32
400 DATA FC,BE,7E,E5
410 DATA 21,FC,BE,AE
420 DATA E1,12,23,13
430 DATA 0B,3A,FC,BE
440 DATA C6,0A,CB,FF
450 DATA 32,FC,BE,78
460 DATA B1,20,E7,C9
470 DATA 00,3E,FF,77
480 DATA F6,FF,C9,xx
```

DISC

Protection systems are getting trickier, which means the length of poke-listings is inevitably going to increase. Still, this disc poke from Richard Monteiro gives you all you'll need to complete Ocean's blockbuster.

You can write-protect the disc; the poke doesn't write to it. It does, though, give 255 of everything, invulnerability, super-jump and faster movement. You'll also have in your mitts the horn, bag and doughnuts.

```
1 ' Head Over Heels - disc
2 ' by Richard Monteiro
3 ' The Amstrad User Oct 87
10 a=PEEK(&BE42): b=PEEK(&BE43)
20 c=256*b+a: FOR d=c TO c+24
30 POKE d,0: NEXT d: CAT
40 FOR t=&BE80 TO &BF40
50 READ a$: e=e+VAL("&"+a$)
60 POKE t,VAL("&"+a$): NEXT
70 IF e=17454 THEN CALL &BE80
80 PRINT"ERROR IN DATA"
100 DATA 06,0C,21,29,BF,11
105 DATA 00,C0,CD,77,BC,21
110 DATA 00,01,CD,83,BC,CD
120 DATA 7A,BC,06,0C,21,35
130 DATA BF,11,00,C0,CD,77
140 DATA BC,21,00,C0,CD,83
150 DATA BC,CD,7A,BC,21,00
160 DATA 01,11,00,9F,3E,76
170 DATA AE,77,23,1B,7A,B3
180 DATA 3E,76,20,F6,CD,17
190 DATA BF,21,00,C0,11,00
200 DATA A0,01,C0,0D,ED,B0
210 DATA 21,72,24,11,73,24
220 DATA 01,08,00,36,FF,ED
230 DATA B0,21,11,BF,11,6E
240 DATA 41,01,06,00,ED,B0
250 DATA 11,89,33,21,0D,BF
260 DATA 01,04,00,ED,B0,11
270 DATA 93,33,21,08,BF,01
280 DATA 05,00,ED,B0,11,9E
290 DATA 33,21,04,BF,01,04
300 DATA 00,ED,B0,C3,00,01
310 DATA 20,20,42,59,20,20
320 DATA 52,50,4D,50,4F,4B
330 DATA 45,3E,FF,77,F6,FF
340 DATA C9,F3,AF,18,02,3E
350 DATA 28,CD,19,BD,01,01
360 DATA BC,ED,49,04,ED,79
370 DATA C9,41,48,45,41,44
380 DATA 31,20,20,2E,53,42
390 DATA 46,41,48,45,41,44
400 DATA 32,20,20,2E,42,49
410 DATA 4e
```

"I LIKE GAMES WITH A GOOD ENDING..."

SHOCKWAY RIDER

FTL's game has been given a good poking by Tony Hoyle and Richard Monteiro. Tony produced the cassette poke and Richard the disc version. On tape (method 1) you can now go Full Circle without losing your head. Rails, street-punks, balls, bricks and even grannies are incapable of harming you.

```
1 ' Shockway Rider - tape
2 ' by Tony Hoyle
3 ' The Amstrad User Oct 87
10 OPENOUT"d"
20 MEMORY &7CF
30 LOAD"!"
40 POKE &868,&80
50 POKE &869,&BE
60 x=&BE80
70 READ a$
80 IF a$="xx" THEN CALL &7D0
```

```
90 POKE x,VAL("&"+a$)
100 x=x+1
110 GOTO 70
120 DATA af,32,d2,76,32,97
130 DATA 96,c3,a4,73,xx
```

DISC

The disc poke for Shockway Rider makes you invulnerable. Type in the poke, run and go Full Circle.

```
1 ' Shockway Rider - disc
2 ' by Richard Monteiro
3 ' The Amstrad User Oct 87
40 OPENOUT"d": MEMORY &7CF
50 LOAD"DISC.BIN"
60 POKE &88D,&CD: POKE &88E,0
70 POKE &88F,&BF
80 FOR t=&BF00 TO &BF07
90 READ a$: POKE t,VAL("&"+a$)
95 NEXT: CALL &7D0
100 DATA af,32,d2,76,32,97
110 DATA 96,c9
```

JET SET WILLY

It's a long time since a poke for Software Projects' classic appeared. And never has a disc poke for this wonderful game alighted upon these pages. Stephen Basford has changed all that. You can choose the number of lives you desire by altering the value held in line 60. This poke is for the "Amstrad CPC Four-Pack" disc compilation and probably won't work on any other version.

```
1 ' Jet Set Willy - disc
2 ' by Stephen Basford
3 ' The Amstrad User Oct 87
10 INPUT "Are you using a 464
(Y/N): ",a$
20 MEMORY &1FFF
30 LOAD"jet2",&3100
40 LOAD"jet3",&7100
50 IF a$="n" THEN POKE
&8D4E,&49: POKE &8D4F,&B6
60 POKE &81F0,7: ' No. of lives
70 POKE &82A8,0: ' Inf. lives
80 RUN"jet1"
```

FEUD

Here are another lot of pokes for Bulldog's epic wizard game. This bunch is from Stephen Basford. A Method 1 entry gives you options for infinite energy (both wizards) and protection from the gardener, and it lets the wizard jump.

```
1 ' Feud
2 ' by Stephen Basford
3 ' The Amstrad User Oct 87
10 INPUT"Gardener harmful (Y/
N):",a$
20 INPUT"Inf.energy for BOTH (Y/
N): ",b$
30 INPUT "Wizards jump (Y/N):
",c$
40 MEMORY &4FFF: LOAD"!feud"
50 POKE &5007,&24: POKE
&5008,&5E
60 FOR c=&5E24 TO &5E35
70 READ d: POKE c,d
80 NEXT c
90 IF a$="n" THEN POKE &5E25,53
100 IF b$="n" THEN POKE &5E2A,53
110 IF c$="n" THEN POKE
&5E2F,200
120 CALL &5000
130 DATA 62,0,50,205,31
140 DATA 62,0,50,98,14
150 DATA 62,0,50,93,14
160 DATA 195,0,4
```

GLIDER RIDER

Alte de Boer certainly knows how to pull a game to pieces. A very classy poke for Quicksilva's musical motorcycle and hang-glider destroy-the-plastic-island game. The poke works on both cassette and disc versions.
This poke removes those annoying laser turrets and replaces them with something much more useful: bombs.

```
1 ' Glider Rider- tape or disc
2 ' Remove laser turrets
3 ' by Alte de Boer
4 ' The Amstrad User Oct 87
40 OPENOUT"d": MEMORY &3FF
50 LOAD"glider.bin": DEFINT i
60 PRINT"Please wait..."
70 FOR i=&8300 TO &9F20
80 IF PEEK(i)=&29 THEN POKE
i,&2A
90 IF PEEK(i)=&A9 THEN POKE
i,&AA
100 NEXT: CALL &400
```

Jan-Mirko Maczewski didn't want to get left out of the Glider Rider mayhem. Indeed, he has sent in pokes to disable the lasers and supply you with infinite bombs, time and energy. Disc or cassette users can benefit from this one. Method 1 if you have it on tape.

```
1 ' Glider Rider
2 ' by Jan-Mirko Maczewski
3 ' The Amstrad User Oct 87
10 OPENOUT"x": MEMORY 1023
20 LOAD"glider.bin"
30 POKE &12DF,&BF ' infinite
bombs
40 POKE &1314,&B7 ' infinite
energy
45 ' disable lasers
50 POKE &334D,24: POKE &44E4,&C9
60 POKE &11BB,&C9 ' infinite
time
```

HIVE

Infinite energy for Firebird's buzzing, but not budget, game has been hacked by poker extraordinary Phil Howard. Use Method 1.

```
1 ' Hive - tape
2 ' by Phil Howard
3 ' The Amstrad User Oct 87
10 DATA cf,ca,8a,21,38,bd
20 DATA 36,1b,21,0e,bc,36
30 DATA c3,23,36,16,23,36
40 DATA be,c3,00,3c,cd,00
50 DATA be,21,20,be,22,ac
60 DATA 01,c9,af,32,c0,34
70 DATA 32,59,3c,c3,82,12
80 y=0: MEMORY &3000
90 FOR x=&BE00 TO &BE29
100 READ a$: a=VAL("&"+a$)
110 POKE x,a: y=y+a: NEXT
120 IF y<>&1008 THEN 140
130 LOAD"hive1": CALL &BE03
140 PRINT"Data error"
```

DEEP STRIKE

In TAU 30 was the cassette poke for Durrell's chocks-away-chaps game. Stephen Basford has kindly sent in pokes for the disc version. By altering the value (3) in line 140 you can select the number of bombers you want. Remove line 150 if you don't care for invulnerability.

```
1 ' Deep Strike - disc
2 ' by S Basford
3 ' The Amstrad User Oct 87
10 MODE 1: MEMORY 5000
20 LOAD"dopic": BORDER 0
30 INK 0,18: INK 1,26
40 INK 2,6: INK 3,2
50 OUT &BC00,1: OUT &BD00,32
60 OUT &BC00,2: OUT &BD00,42
70 OUT &BC00,6: OUT &BD00,24
80 CALL 32768; LOAD"frame",&9E00
90 LOAD"deep.bin":
```

```
LOAD"tables",&C700
100 LOAD"hills",&CE00:
LOAD"moved"
110 LOAD"graphics",&D600
120 POKE &57E0,201: POKE
&5777,201
130 POKE &5729,0: ' Protecton
140 POKE &5BDF,3: ' No. of
Bombers
150 POKE &7055,0: CALL &8600
```

DOOMSDAY BLUES

A Method 2 poke (fast-forward the cassette until you reach the binary file EDEN, which is the first block after the picture) from Stephen Basford. You are given infinite strength, bravery and fitness. If endlessness equals mindlessness in your books then you can alter the amount of strength etc that you have - must be in the range 1 to 100.

```
1 ' Eden Blues - tape
2 ' by Stephen Basford
3 ' The Amstrad User Oct 87
10 MODE 1
20 INPUT " Remain value (70):
",r
30 INPUT "Use value (10): ",u
40 INPUT "inf.strength (y/n):
",a$
50 INPUT "inf.bravery (y/n):
",b$
60 INPUT "inf.fitness (y/n):
",c$
70 CLS: LOCATE 1,3
80 sum=0
90 FOR c=49152 TO 49219
100 READ d$: d=VAL("&"+d$)
110 POKE c,d: sum=sum+d
120 NEXT c
130 IF sum<>6286 THEN STOP
140 POKE &C016,r: POKE &C01B,u
150 IF a$="n" THEN POKE &C020,53
160 IF b$="n" THEN POKE &C039,53
170 IF c$="n" THEN POKE &C02B,53
180 CALL 49152
190 DATA F3,06,04,21,40,C0,11,40
200 DATA 00,CD,77,BC,21,40,00,CD
210 DATA 83,BC,CD,7A,BC,3E,46,32
220 DATA 86,71,3E,0A,32,8B,71,3E
230 DATA 00,32,5D,8D,32,BC,95,32
240 DATA 12,96,3E,00,32,97,8F,32
250 DATA 98,8F,32,5B,90,32,28,98
260 DATA 3E,00,32,87,91,C3,00,70
270 DATA 45,44,45,4E
```

AVENGER

Richard Monteiro has had a hectic poking session this month. Gremlin's disc version of the beat-em-up game is his latest victim. Kwon will never get tired of replenishing your energy with this poke.

```
1 ' Avenger - disc
2 ' by Richard Monteiro
3 ' The Amstrad User Oct 87
10 DATA 00,3E,01,CD,0E,BC
20 DATA 21,C4,00,CD,D4,BC
30 DATA 22,C5,00,11,00,00
40 DATA 21,00,01,0E,41,DF
50 DATA C5,00,21,BD,00,22
60 DATA 6E,01,C3,00,01,AF
70 DATA 32,C1,6C,C3,DB,65
80 DATA 84,00,00,07
90 FOR t=&9A TO &C7
100 READ a$: z=z+VAL("&"+a$)
110 POKE t,VAL("&"+a$): NEXT
120 IF z=3828 THEN CALL &9A
130 PRINT"error in DATA"
```

ARKANOID

Imagine's coin-op goes infinite. Richard Monteiro has supplied the game with a never-ending supply of bats. Disc only.

```
1 ' Arkanoid - disc
2 ' by Richard Monteiro
3 ' The Amstrad User Oct 87
10 DATA 06,0A,21,75,90,11
20 DATA 00,C0,CD,77,BC,21
30 DATA 40,00,CD,83,BC,CD
40 DATA 7A,BC,21,40,00,11,
50 DATA C0,75,3E,55,AE,77
60 DATA 23,13,7A,B3,3E,55
70 DATA C2,1C,90,AF,32,F3
80 DATA 02,11,7C,41,06,06
90 DATA C5,21,47,90,01,1E
100 DATA 00,ED,B0,C1,10,F4
110 DATA 21,65,90,01,10,00
120 DATA ED,B0,C3,52,46,54
130 DATA 48,49,53,20,44,49
140 DATA 53,4B,20,50,4F,4B
150 DATA 45,20,42,59,20,52
160 DATA 50,4D,20,31,39,38
170 DATA 37,20,20,20,20,48
180 DATA 49,20,41,4E,44,20
190 DATA 42,49,20,46,52,4F
200 DATA 4D,20,49,47,41,4D
210 DATA 45,20,20,2E,53,42
220 DATA 46,00,00
230 a= PEEK(&BE42): b=PEEK(BE43)
240 c=256*b+a: FOR d=c TO c+24
250 POKE d,0: NEXT d: CAT
260 FOR t=&9000 TO &9080
270 READ a$: b=b+VAL("&"+a$)
280 POKE t,VAL("&"+a$): NEXT t
290 IF b=10411 THEN CALL &9000
300 PRINT"ERROR IN DATA"
```

Send your pokes and tips to: The Editor (Cheat Mode) 1/245 Springvale Road, Glen Waverley, Vic 3150 but PLEASE, PLEASE make sure that they are your own work and not just a copy from another publication. Unless you have permission, this is breaking copyright.

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 |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.

Firing-up CP/M - 1

A new series to help you down the CP/M path

We have run a number of articles on CP/M over the last couple of years or so relating primarily to CP/M 2.2 for the CPC464.

Owners of CPC6128s and PCWs have another version - CP/M Plus which allows the use of the larger memory capacities. For the benefit of all the new users and those still struggling with CP/M, regardless of which machine they have, we present this new series which hopefully, once and for all, will put you on the right course.

We don't intend to go right back to step-by-step first principles - you already know that CP/M is an operating system written by Digital Research which gives you the means to communicate with your computer, use peripherals and manipulate files. We'll assume that you also know how to load CP/M on your particular machine and have seen the familiar A-prompt which indicates that CP/M is waiting for you to tell it to do something.
Let's do just that!

Turn your Arnold or PCW on and load up CP/M. On the A-prompt (the A> symbol) type DIR and hit Return. You will be presented with the directory listing, but let's look at it more closely.

Notice that most of the files listed end with the extension (another name for file type) .COM. This indicates that the files concerned are program files, summoned by issuing the COMmand that is the first part of the filename. For example, if you have a CPC machine and are running CP/M PLUS, try entering DATE at the prompt. A date and time will be displayed on the screen, followed by the good old A-prompt. What you have done is run the program file called DATE.COM. The time and date are probably totally wrong, but don't worry about that at this stage.

If you have a PCW you probably won't have the file DATE.COM on your directory. However you will have BASIC.COM, so try typing BASIC instead. A message will come up showing you are in Mallard Basic, followed by the Basic prompt 'OK'. You could now start writing Basic programs if you felt like it, as what you have done is run the program BASIC.COM, which is the programming language Mallard Basic. Instead enter System which takes you out of Basic and back into CP/M PLUS.

Finally to those of you with CP/M 2.2, who might be feeling left out in the cold by now as neither of the above files are on your system disc - try entering LOGO2. This will load up the Logo programming language, stored in the file labelled LOGO2.COM, and you will be presented with the Logo prompt '?'. To exit back to CP/M 2.2, simply type bye and the A-prompt reappears.

COM files are probably the most important that you will find on commercial software discs, but there are other file types that are generally reserved for particular types of files, and best avoided when naming your own efforts. These are listed in a panel on these pages, together with other types that are in common usage. Another file type that might be important is .BAS, used for Basic programs. Programs written under Basic are saved with this extension, but these do not operate in the same way as COM files - they cannot be run by simply typing the first part of the program name. Instead you must load Basic and summon the program from there.

MORE ON DIRECTORIES

The basic directory listing, obtained using the straight DIR command, is of limited use. It tells you what files are on the disc in question, but it does not tell you anything about them. In particular it does not tell you how long they are, or how much room you have left on the disc - both important information for housekeeping.

As usual, CP/M PLUS and 2.2 handle this in very different ways. If you are using CP/M PLUS you will notice a file listed on the directory called DIR.COM. As you know this is a program file, summoned by the command DIR. It contains extensions to the basic DIR command, which are listed in the panel on these pages, and increase the power of DIRectory listing. Try entering this after the A-prompt:

```
DIR [size]
```

Now you can see how long those files are in kilobytes. At the bottom left of the screen you can also see that they take a total of

```
A>dir [size]

Scanning Directory...

Sorting  Directory...

Directory For Drive A:  User  0

A: BASIC    COM   28k : DIR      COM   15k : DISCKIT  COM    7k
A: ED       COM   10k : ERASE    COM    4k : J11CPM3  EMS   40k
A: KEYS     WP     1k : LANGUAGE COM    1k : PALETTE  COM    1k
A: PAPER    COM    2k : PIP      COM    9k : PROFILE  ENG    1k
A: RENAME   COM    3k : RPED     BAS    7k : RPED     SUB    1k
A: SET      COM   11k : SET24X80 COM    1k : SETDEF   COM    4k
A: SETKEYS  COM    2k : SETLST   COM    2k : SETSIO   COM    2k
A: SHOW     COM    9k : SUBMIT   COM    6k : TYPE     COM    3k

Total Bytes     =    170k  Total Records =    1305  Files Found =   24
Total 1k Blocks =    170   Used/Max Dir Entries For Drive A:   27/  64
```

170K of the disc space. Amstrad discs can hold upto178K of data, so simple arithmetic tells you that you have approximately 8K of empty disc capacity left.

If you want to know all the facts about the files on a disc, try the following command after the A-prompt:

```
DIR [full]
```

As you can see, in addition to the size in kilobytes a number of other parameters are listed as well. Next comes 'Records', which is a more accurate measure of the file size. One Record is 128 bytes of data. If you look at the above directory listing you will see six files listed as 1K long, but containing anything from one to eight 128-byte records. The total number of records on

```
A>dir [full]

Scanning Directory...

Sorting  Directory...

Directory For Drive A:  User  0

     Name      Bytes   Recs   Attributes      Name      Bytes   Recs   Attributes
------------ ------ ------ ------------ ------------ ------ ------ ------------
BASIC    COM    28k    224 Dir RW       DIR      COM    15k    114 Dir RW
DISCKIT  COM     7k     56 Dir RW       ED       COM    10k     73 Dir RW
ERASE    COM     4k     29 Dir RW       J11CPM3  EMS    40k    320 Dir RW
KEYS     WP      1k      7 Dir RW       LANGUAGE COM     1k      8 Dir RW
PALETTE  COM     1k      8 Dir RW       PAPER    COM     2k     16 Dir RW
PIP      COM     9k     68 Dir RW       PROFILE  ENG     1k      2 Dir RW
RENAME   COM     3k     23 Dir RW       RPED     BAS     7k     55 Dir RW
RPED     SUB     1k      1 Dir RW       SET      COM    11k     81 Dir RW
SET24X80 COM     1k      8 Dir RW       SETDEF   COM     4k     32 Dir RW
SETKEYS  COM     2k     16 Dir RW       SETLST   COM     2k     16 Dir RW
SETSIO   COM     2k     16 Dir RW       SHOW     COM     9k     66 Dir RW
SUBMIT   COM     6k     42 Dir RW       TYPE     COM     3k     24 Dir RW

Total Bytes     =    170k  Total Records =    1305  Files Found =   24
Total 1k Blocks =    170   Used/Max Dir Entries For Drive A:   27/  64
```

the disc is recorded as 1305, which seems to indicate that only 163K of the available space is occupied, and not the 170K listed. Unfortunately CP/M takes the latter figure and not the former, so the extra 7K is effectively unavailable.

The last two parameters are under the heading of 'Attributes'. Files can either be 'Directory' or 'System', which are hidden from DIRectory listings. Files can also be 'Read/Write', which means they can be changed or deleted by the user as well as read, or 'Read Only' files, that you can run or look at but cannot change or delete. Setting files to 'RO' is a useful way of preventing unfortunate accidents - fear not, we will be covering attributes later.

BACK TO CP/M 2.2

Unfortunately, none of these work with CP/M 2.2. Instead you have to use a new command, STAT, which gives you control over the STATus of your system. This program is contained in the file STAT.COM which you can see on the directory listing, and is a powerful command in its own right which we will examine more fully next month; but for now let's use STAT to examine the contents of the disc. First of all, try entering STAT by itself:

```
A>stat
A: R/W, Space: 7k
```

This does at least tell you how much space you have left on the disc, but does not give a complete file list. STAT operates in much the same way as DIR, using wildcards . So to get a full file listing, try entering STAT*.* after the A-prompt.

```
A)stat *.*

 Recs  Bytes  Ext Acc
    8     1k    1 R/W A:AMSDOS.COM
   10     2k    1 R/W A:BOOTGEN.COM
   15     2k    1 R/W A:CLOAD.COM
   14     2k    1 R/W A:CSAVE.COM
   38     5k    1 R/W A:DDT.COM
  168    21k    2 R/W A:DISC.BAS
   48     6k    1 R/W A:DISCKIT2.COM
    2     1k    1 R/W A:DRLKEYS.COM
    4     1k    1 R/W A:DUMP.COM
   52     7k    1 R/W A:ED.COM
   22     3k    1 R/W A:FILECOPY.COM
    1     1k    1 R/W A:FWRESET.COM
   14     2k    1 R/W A:LOAD.COM
  256    32k    2 R/W A:LOGO2.COM
    1     1k    1 R/W A:LOGO2.SUB
   76    10k    1 R/W A:MOVCPM.COM
   58     8k    1 R/W A:PIP.COM
   89    12k    1 R/W A:RITDEMO.BIN
  208    26k    2 R/W A:ROINTIME.DEM
   61     8k    1 R/W A:SETUP.COM
   41     6k    1 R/W A:STAT.COM
   10     2k    1 R/W A:SUBMIT.COM
   12     2k    1 R/W A:SYSGEN.COM
    6     1k    1 R/W A:XSUB.COM
Bytes Remaining On A: 7k
```

Although organised differently, this gives much the same information as the CP/M PLUS command DIR [FULL]. The STATus command has many uses, and will be covered more fully in next month's episode of this Series.

DIRECTORY OPTIONS

CP/M 2.2 users are stuck with the simple command DIR, but CP/M 3.1, otherwise known as CP/M PLUS, allows several options that extend the power of DIR. With the exception of DIRS, which is a command in its own right, these are enclosed by square brackets after a space - for example DIR[FULL]. In practice you need only type the first square bracket and the first two letters of the extension, so DIR[FU would have the same effect:

DIRS This is the flipside of the conventional DIR command, displaying only the SYStem files that are hidden from DIR.
[FULL] Shows the full details of the directory, including size in kilobytes and all attributes. The list is sorted by filename and type.
[SIZE] Displays the files with their size in kilobytes.
[DATA] Displays the date and time stamps of the files.
[ATT] Displays the attributes of the files, such as R/O for Read Only files, R/W for files that can be written to as well, and SYS if this option is used.
[DRIVE=ALL] Displays the files on all the disc drives in use.
[DRIVE=A] Displays the files on the disc in drive A. [DRIVE=B] would display those on drive B.
[EXCLUDE] Excludes files with the specification following the command from the directory listing, so DIR [EX] *.COM would show all except the Command files.
[NOSORT] Displays the file names in the order that they are found on the disc, rather than sorted by file name and type. This option requires at least the first three letters [NOS to run properly.
[RO] Only shows files with the R/O attribute.
[RW] Only shows R/W files.
[SYS] Only shows SYS files - rather like DIRS.
[USER=1] Only shows files for User One (or whatever number is used).
[USER=ALL] Displays all files, regardless of user.

With the exception of DIRS, all the options listed above are 'transient', and require the presence of the file DIR.COM on the logged drive. This file is on your CP/M PLUS system disc, side one.

FILE TYPES

These are the most common filetypes or 'extensions' that you are likely to find on a program disc, and are generally best avoided in your file names:

COM A 'Command' file, indicating an executable program.
BAS A Basic program. If you save a program written from Basic it will automatically be saved with the BAS extension, unless you specify otherwise.
BIN A Binary file. Programs written in machine code will usually be saved with this extension.
BAK A Backup file. When you save a text file from Wordstar, or some other word processors, the previously saved file is given this extension. If you need to edit the Backup file it has to be renamed as most word processors won't edit files with this extension.
OVR An Overlay file. These are special program files which act as extensions to the main program. An Overlay is called into memory by the main program when it is needed.
SUB A Submit file. We will be looking at how you create these later on in this series.
$$$ A Temporary file. These are sometimes created by programs for their own purposes.

There are other file types which have become accepted for certain types of file. Whether you use them or not is up to you, but it is generally a good idea if your discs are likely to be used by anyone else:
DAT A Data file, for data that might be needed by a program.
DOC A Document file. Usually a text file containing information about a program on the disc, or just some erudite prose.
TXT A Text file. Usually created by a word processor, and much the same as a DOC file.

PC HELP

We've gathered together some snippets of information to help you drive your PC a little better.
The continuation of PC HELP into future issues will depend largely upon your contributions.
So, however simple or obvious your tip may seem to you - it may be a God-send to someone else. Share and help others by sending your revelations to:

PC HELP
The Amstrad User
1/245 Springvale Road
Glen Waverley
Vic 3150.

BASIC2 BUGS

Do you remember reading a letter from Doug Fortescue (Shelley, WA) in the July '87 issue of TAU? He was not happy with the calculation abilities of Basic2. The simple calculation a*b + c*d where a is a negative integer and b is equal to 0 (zero), or vice versa, produced an inaccurate result. We faxed Locomotive Software with the problem, but they chose to ignore us.

We don't know if Doug has a relation in the UK, but a P.W. Fortescue of Eastleigh in Hampshire also had the same problem. He (we assume a male) wrote to his local PC mag who also took it up with Locomotive Software. They had more luck being 'just down the road' and came back with the following answer:

"We spoke to Locomotive about your problem. They are now advertising an improved version of 1.21 of Basic2 for £5.95 including postage. This does not suffer from your problem."

"No computer software is completely bug-free, and the final bugs tend to get iron out when customers complain. Like most other software, Basic2 has gone through a number of versions since it first appeared. The version first issued was version 1.12, which is the version you have. The version being shipped with the Amstrad PC now is version 1.14. Locomotive have given Amstrad a copy of the latest version (1.21) and what Amstrad does with it has yet to be seen."

"Locomotive also sent us a Basic2 program that fixes two bugs in version 1.12, upgrading it to version 1.12B. The other bug occurs when UPPER$ or LOWER$ of a constant string is evaluated - the wrong result is given and the machine can crash."

"Make a copy of your Basic2 before running this, as it patches directly to the main files. You should see the message CORRECT if the program has run successfully. BREAK indicates a mistake somewhere."

FOOLHARDY DISCS

PC owners with hard discs could be interested in the batch file (shown on next page) which prevents accidental reformatting of their hard disc. The basis of the program appeared in the March '87 issue of Byte on its BIX pages.

1. From DOS enter: ren format.exe formatx.exe
2. Use RPED (or similar) to create format.bat below.

For instructions on how to use the batch file type FORMAT and the file

```
REM Program to patch BASIC2 Version 1.12 to 1.12B
p$="a:\basic2\basic2.":REM Path & filename

REPEAT:READ c$:FOR i=1 TO LEN(c$)
 s=(s+ASC(c$(i)))*2:s=(s+s\1000)MOD 1000
NEXT:UNTIL c$="e":IF s<>182 THEN STOP
RESTORE:RECORD b;a [0 TO 127]UBYTE
REPEAT:READ c$
 IF c$="o" THEN READ f$:OPEN #9 OLD RANDOM p$+f$
 IF c$="c" THEN CLOSE
 IF c$="g" THEN READ r:GET #9,r$ AT r
 IF c$="p" THEN PUT #9,r$
 IF c$="x" THEN READ f,l:FOR o=f TO l:READ r$.b.a[o]:NEXT
UNTIL c$="e"
DATA c,o,app,g,1,x,18,19,13,175,p
DATA g,310,x,85,86,64,4,p
DATA g,319,x,23,28,232,198,250,233,73,251,p
DATA g,324,x,73,73,65,x,75,78,84,2,137,4
DATA x,87,90,138,236,50,232,x,92,92,196
DATA x,94,97,3,233,19,7,x,99,103,30,7,233,92,110
DATA x,105,116,247,229,93,137,84
DATA 2,247,218,112,14,127,20
DATA x,120,120,212,x,122,127,216,131,218,0,235,202,p
DATA g,325,x,1,13,11,192,117,4,11,201
DATA 120,196,247,218,232,188,255
DATA x,61,61,171,x,87,87,147,p,c
DATA o,rsc,g,13,x,127,127,66,p
DATA g,14,x,2,3,50,57,x,5,7,77,97,121
DATA x,10,10,55,p,c,e

REM Program to checksum BASIC 2 Version 1.12B
p$="a:\basic2\basic2.":REM Path and filename

f$="app":GOSUB check:IF s<>&28e0d839 THEN STOP
f$="rsc":GOSUB check:IF s<>&10babee0 THEN STOP
?"CORRECT":END

LABEL check
 CLOSE:OPEN #9 OLD RANDOM p$+f$
 RECORD w;a[0 TO 63]UWORD:s=0
 WHILE NOT EOF(#9):GET #9,r$:FOR i=0 TO 63
 s=(s+r$.w.a[i])*2:s=(s+s\&3fff0000)MOD &3fff0000
 NEXT i:POSITION #9,NEXT:WEND:CLOSE
RETURN
```

will prompt you. To format a hard disc, use the formatx.exe file as normal.

NEWWORD TIP

If you use NewWord3 (and it might work with NewWord2 as well) and have been bugged by hard carriage returns that appear at the end of ASCII files when you try to edit them - here's the answer. Try keying [Ctrl][N] and you will find them transformed to soft carriage returns. One snag though - it's not compatible with WordStar.

```
ECHO OFF
ECHO %1 %2
REM Used to prevent accidental re-format of hard disk
REM Uses FORMAT.EXE renamed to FORMATX.EXE
ECHO DISK FORMAT:
IF %1 @==@ GOTO :START
IF %1 == a: GOTO :A
IF %1 == A: GOTO :A
IF %1 == b: GOTO :B
IF %1 == B: GOTO :B
:START
ECHO ERROR!
ECHO YOU MAY ONLY FORMAT A DISK IN DRIVE A: OR B:
ECHO PLEASE PLACE A DISK IN DRIVE A: OR B: AND RETRY
ECHO USE THE FORM:- FORMAT [drive:] [/option]....[/option]
ECHO i.e. FORMAT A: /S /V (the space is obligatory)
GOTO :END
:A
:B
FORMATX %1 %2 %3 %4 %5
:END
```

FLIGHT OF FANCY

Before rushing out and buying a copy of Microsoft's Flight Simulator, you may like to know that the colours you'll get are somewhat strange. Reports of a grey ground, turquoise sky and purple mountains and instrument panel when running this software on a 1512 colour monitor have been confirmed by Microsoft themselves. You see, Flight Simulator runs in IBM Colour Graphics Adaptor (CGA) mode which gives four colours in medium resolution (320 x 200 pixels). There are four colours chosen for this program, namely, Black, White, Cyan and Magenta (purple and turquoise on your screen) which correspond to 'Palette 1' of the 1512 medium resolution display, which in turn matches the CGA standard. However, in America there are two sets of colours - those occurring naturally (natural colours) and a second set appearing on the NTSC television system they use, the latter apparently being designed for people who are used to seeing orange-faced presenters and purple mountains.

Whilst the IBM 'personal colour display' doesn't use the NTSC system, displays which always look a bit muddy on a genuine IBM screen take on an exaggerated look in a clone designed to reproduce natural colours.

As IBM is the only standard to work on, games have to be written 'down' to that standard. As you know, the pc1512 can display 16 colours in a 640 pixel resolution, but as this is non-standard only programs written specifically for it will use it.

Sub-directories, Pathways and more Public Domain Software

From Chris Collins

As I write this, it is late July and I have just finished sending out all of the copies of PC File III that were requested. I hope you are finding it useful. Just a little word of warning. Hang on to your original PC File disc as I believe that Jim Button has released a newer version called PC File+, and I will be offering an upgrade (at a special price) to people that ordered PC File the first time around.

As I was processing the orders, I was also reading the comments that people were writing. Most of them seemed to be very impressed with the column and wanted it to keep going. Remarks like that make it worthwhile for me to write, as I feel that I am helping other people to get the best out of their computers.

Also in the letters were a few questions regarding the operation of your PC1512s. Unfortunately, I cannot answer each question individually, but I will attempt to answer the questions in the column. The first question relates to pathnames. Most people seem to have no trouble at all with drive names, but pathnames and sub-directories have them beat. Don't be annoyed with yourselves as I and a lot of others have also had a lot of trouble with them when first we got our IBM compatibles.

To understand pathnames and sub-directories, you need to think of a disc as a single drawer filing cabinet. If you put all your information into the drawer willy-nilly, it will all be there, but it will be difficult to find. However, if you first divide all your information into smaller groups, put them into folders and then store them in the filing cabinet, your information will then be a lot easier to find. That is the easiest way that I know of to explain sub-directories. As to pathnames, these are simply a way to tell DOS how to find a certain file in one of your folders. For example, a little while ago my hard disc went haywire, and I was returned to the world of floppy discs. To keep my Tasword workdisc neat, I created a sub-directory called DATA. This was where I wanted to put all my documents, as distinct from the Tasword program files. I simply told Tasword that all my documents would be in a directory called \DATA, and Tasword could find them. The correct pathname for this is A:\TASWORD\DATA.

To explain even further, a document called A:\TASWORD\DATA\amuser.aug would be found on drive A:, in a sub-directory of TASWORD called DATA. Sub-directories and pathnames are somewhat redundant on a floppy disc, but don't let that stop you from using them to keep things neat. However, on a hard disc they are absolutely essential, so you had best learn how to use them, if you ever plan to upgrade to a hard disc. A directory that is often forgotten is the ROOT DIRECTORY. This is the first directory on a disc and is often referred to as A:\. This simply refers to the root directory on A: drive. I hope that this has made things easier.

Now, onto this month's software surprise. For those of you that are still building up your software collection, this month will be a beauty. For October, I have managed to scrape together a collection of the best utilities and useful programs that I could find. These occupy 5 discs. All of these files have been archived (a method of saving space and keeping files together) to save space, and include any necessary documentation. These will be available for $40 including postage and handling. Please send your cheques or money orders to me, C/- The Amstrad User, and please allow 14 days for delivery.

As I don't have the space available in my column to explain every program on the 5 discs, I have just gone through the list and picked out a couple off each disc. Let's get on with the goodies.

Off the first disc is a program called NEWKEY. NEWKEY is a keyboard definition program similar to PROKEY and SUPERKEY. These types of programs allow you to redefine your keyboard to produce any set of commands, or string of text from a single key. Key definitions can be loaded in from disc, or they can be created on the fly. If these definitions are not then saved however, they will be lost.

This is useful if you need to create a definition, that you only wish to use a couple of times. The key definition files that are supplied with the program include a file for Wordstar users, another for assembly language programmers and a further file that will create a Dvorak keyboard.

Documentation for the program runs to 20 pages, and this explains very easily how to use the program. Because NEWKEY is memory resident, I tried it with my other memory resident program, and I didn't come across any clashes.

CWEEP is a program that provides the same capability for MS-DOS users that SWEEP does for CP/M users. CWEEP provides the ability to manipulate a sorted list of files, which can then be viewed, printed, deleted and copied. This can be done either individually or in groups.

When the documentation is printed out, it runs to 8 pages although this is not really required to operate the program. CWEEP works on a system of tagged and untagged files. These will then be operated on by the next command called. This program is a "must" in any good MS-DOS library. Another program of this type is also included on the discs. This is called VFILER and is on disc 4. If you don't like CWEEP, try VFILER. It is the same type of program, but it has a totally different feel to it. I have tried both programs and I find CWEEP better, but the choice is totally up to you.

The last program off the first disc is a little beauty that I only recently found. It is called FOGFIND, and it measures the readability of a piece of text. By counting the numbers of words and

sentences, and then comparing it to a modified Gunning Fog index, this will then give you a figure relating to the complexity of a piece of text. It doesn't tell you whether a piece of text is good or bad, only how difficult or easy it is to read. I tried FOGFIND out on the previous Compatibles Corner that have been written, and find that they are running at an average 9 1/2. This is equivalent to a 9th or 10th grader and is reasonably difficult, but not so hard that you won't want to read it.

On to the second disc, and we come across three absolutely necessary programs. The first is called ST, and is a replacement for the DOS TYPE command. ST has three major enhancements to the type command. These are as follows:

1. ST pages the display, rather than scrolling,
2. ST removes the funny characters that occur when you try to read a Wordstar file, and
3. Up to 64 screens of text file are kept in memory.

ST is much easier to use than the TYPE command, and much more powerful. ST can be called from the DOS prompt with a filename, or if you forget, it will prompt you for a filename.

CLEAN2 is another simple program for the PC that will make full use of the cleaning discs that are available. Because the cleaning discs are paper, DOS will attempt to read the directory and then stop and produce an error message. Therefore, you will probably never use more than 5% of the surface area of the cleaning disc. CLEAN2 forces DOS to ignore the error created above, and moves the heads all around the cleaning discs. Although cleaning your drives is a good preventative measure, it should not be done to excess. Once every second month is more than enough, unless your machine is being used all day long.

SCRNSAVE is a small machine language program that will blank your screen, if you don't touch a key for three minutes. SCRNSAVE was written back when the monitors were not of the quality that we have today. Too bright a static image for too long a period would burn into the screen itself and damage the screen. Not really a worry today, but it does pay to be safe, especially if your computer will do all the work for you. *(It also acts as a security from prying eyes if you are running a program displaying confidential figures and have to leave the screen unattended for a period of time - Ed)*

The third disc contains two programs that allow you to control your printer. The first is called PRINTCTL and this is the short name for PC Print Control. This program (and SETPRINT) allow you to control your printer from the keyboard, rather than from the printer itself. PRINTCTL allows you to set print style (emphasized, double strike etc.), number of lines per page, direction of printing and many other commands that are required to operate your printer effectively.

SETPRINT is the other program that I mentioned. The major difference between the two programs is that SETPRINT is memory resident, and the minor difference is that it supports A4 paper. SETPRINT is called by pressing the ALT and RIGHT-SHIFT keys together. This will popup a window, occupying most of the screen. From this window you will be allowed to control print size, print quality, character style and form size. After setting your preferences, the window will disappear and all will be well. Whether you use SETPRINT or PRINTCTL is again up to you. Both are fine programs, but SETPRINT does not offer the same range of options as PRINTCTL.

PCCHECK is another program off the third disc, and this is a menu driven cheque book balancing program. This program's full title is PC Check Manager, and is written by the same person that wrote PRINTCTL. That is not meant to be totally derogatory, however both programs contain the same garish opening screen and it is this I object to more than anything else.

Working through the program is very simple because of the menus, but some points are rather difficult to understand. The biggest problem that I found related to the date format.

As this is an American program (cheque vs. check), the date format expected is MM/DD/YY, and it takes a bit of getting used to. The other problem that I came across whilst testing PCCHECK was inserting a starting balance into the account. A quick read of the documentation soon fixed that. The program is meant to be a cheque book balancer, but I had no trouble at all using it to balance my savings account. Overall, I didn't really find a need for it, but if you have trouble keeping your account in order, give it a look.

Moving on to the fourth disc in the series, we come across a program called BRADFORD. This is a gem of a program, and I think that most of you will find it the same. It is a similar type of program to Tasprint PC, but it is infinitely better. It allows you to print out your documents in NLQ, even if your printer does not have that option. BRADFORD loads 5 different fonts into memory, and allows you to print your documents in any of these 5 typefaces. The printer options at the command menu are necessarily sparse, but a bit of trial and error will get you one that works with your printer.

It took me a couple of tries to find a printer option that worked with my printer, but after that I tried out all five fonts, and was quite impressed with the results. Vanilla font is exceptional! The documentation (BRADFORD.INF) is very sparse, but it is more than enough to get the program working properly, and the author does promise information about imbedded commands and the like if you register. At US$15 (approx. AUS$25) it is cheaper than Tasprint PC and infinitely better.

P is another program that purports to replace a DOS command, in this case the PRINT command. According to the documentation, P is a super printing program and to date it has found a place on my hard disc. With the addition of some switches on the DOS command line, P will also do much more than PRINT was ever able to do. Some of the options that are available include:

* current date and time at the top of the page,
* pause for single sheet paper,
* line numbering,
* slashed zeros (ø),
* optional left margin settings, and
* typewriter mode.

As you can see, P is definitely more powerful than PRINT is ever likely to be, and should also have a place with your DOS disc.

Onto the fifth disc, we come across the final two programs that we will be looking at this month. The first of these is DOG.ARC. One of the problems that can occur with disc files is that they will become fragmented on the disc. This occurs because you use a file, update it and resave it, then erase the first copy. The next time that you go to save a file, DOS will attempt to put it in that area. If the file is too large, DOS will put what it can there, and save the rest in another area. Remembering that the smallest area on a disc is 512 bytes, it is possible for a 15 kilobyte file to occupy 30 small areas. This is not likely, but it does show the extremes.

You might wonder what difference this would make. So long as DOS can find the bits of the file, and in the right order, who cares. Well, the end effect of all this is that it will make all your loading operations much longer than they have to be, because DOS is searching all over the disc trying to find the components of a file.

This can be remedied quite easily on a floppy disc. Simply format another disc and copy all the files across. Dos will unfragment the files as it copies and all will be well. However, on a hard disc, this is not really a practical proposition, so Disc OrGanizer was created. DOG will go through your hard disc (and also your floppies, if you wish) and unfragment your files for you. The documentation for DOG explains the four ways that it is possible for DOG to organise your files. Fortunately, it is easy to understand and follow, so you shouldn't have too much trouble operating DOG. The only other program of this type that I know is called Disc Optimizer and it retails for US$50.

The last program to look at is the all singing, all dancing PKX34A20. Like all other programs on the discs, it is archived. This is a method of collecting all the necessary files for a program, compressing them and then making one file for the result. However, unlike all the other files on the discs, you will not need PKXARC to get PKX34A20 working. The reason for this is that PKX34A20 is self extracting. Otherwise how would you ever get PKXARC to unarchive it. The two basic files in this archive are PKXARC and PKARC.

Both of these have documentation with them. PKXARC lets you unarchive files that have been created into an archive, and PKARC is the program that creates the archive. Both are very easy to use, and should give you no trouble at all to run. If you find that all your backup files are occupying too many discs, simply use PKARC to create an archive of them and then replace your backups with this archive. This will save you anywhere between 10% and 70% of your disc space. Also on this disc is ARC51. This is also an archiver, but it is a lot slower than PKARC. Try both of them, and use whichever one you feel is better.

Well I think that this will be all for this month. I hope that you are all enjoying your PC's, and I hope to get some suggestions from you for future topics for Compatibles Corner. Just as a teaser, and so I can't be blamed for being too businesslike, next month will include a special 4 disc pack of games. Until then, Happy Computing!

TIP-OFFS

Revelations to make you reel and revel

Pulling out the stops

While the TAB stops appear straight forward in their use, there is one position where it is impossible to see that you have inserted a TAB. This is on the right hand margin.
If you try the position before this, the word gets split, while the position after the margin does strange things to your document. The use of the combined tab and margin is of most use when using the Paragraph TAB. The use of a carriage return cancels the paragraph tab but the combined margin and tab allows the continuation of the paragraph even if only one word or a set of tabs is present on that line.
J.G. Boers, North Yunderup

Getting the measure of Newsdesk International

One problem with the otherwise excellent Newsdesk International desktop publishing program is that it is hard to judge from the screen what the size of an object will be on the printout.
Just remember this simple paper-to-screen conversion: on the 'x'-axis, each 1/10th of an inch on paper corresponds to 12 pixels on the screen. On the 'y'-axis, 1/6th of an inch on paper corresponds to 12 pixels on the screen. These are useful figures because in LocoScript you would typically use 10 characters per inch text ([+]LP10), and 6 lines to the inch line spacing ([+]LS6).
Therefore what you need to do is make up a grid showing the screen co-ordinates required for a given plan on paper.
From the MAIN MENU, select WINDOWS and then SET TEXT. With the cursor keys and [ALT] expand the text WINDOW to span the full width of the page and up the page as far as 'y' co-ordinate 504. Return to the main menu, and select graphics→draw→ pen. Press [C] to give a co-ordinate display, and draw two parallel lines across the foot of the page 4 pixels apart. Draw a second pair 8 pixels apart up the left hand edge. Stop the top end of these lines when the cursor reaches 504 pixels up the screen.
These horizontal and vertical lines will be your'x' and 'y' scales. Divide each into 12 pixels units.
Every 60 pixels on the x-scale, and every 36 on the y-scale, extend the unit marker 10 pixels into the page. These are the 1/2 inch position markers.
From the main menu move to fonts→write font→keyboard input. Use the default font to calibrate your x and y scales in pixels, and add any other text you feel like. Now print the page out in high density.
Take your printed template to your local Copy Shop, and ask them to photocopy it onto A4 overhead transparency film.
Now from your rough paper plan for a page you can work out where the text and lines should go on the screen to match it.
Ronald Macdonnell

Tel all

A list of telephone numbers which can be cut to fit into any slim diary or Filofax can be made in the following way. Create a document with these layouts:

Layout 1:	Layout 2:
Pitch	Pitch PS
Line spacing $^1/_2$	Line spacing $^1/_2$
Line pitch 6	Line pitch 6
Left margin 10	Left margin 48
Tabs 12,31,45	Tabs 50,69,83
Right margin 43	Right marg. 81

Pagination: remove all headers and footers and enter the following in the pages menu:

Page length 38
Header zone 0
header position 1
Footer zone 0
Footer position 38

Now set up your headings with:
+LayouT1+Bold[RETURN]
Name [TAB] Telephone
[RETURN][RETURN]
-Bold+Pitch17[ENTER]
+SupeR[RETURN][RETURN].
Then enter the names, [TAB], and the numbers each on a new line - the page will take up to 70. Start each section with a capital letter heading set up by a phrase such as:
(-SupeR)(+Bold)(+Pitch17D)
A(+Pitch17)(-Bold)(+SupeR)
followed by two RETURNS.
At the end of the page change to Layout 2 by typing [f6] [ENTER]+LayouT2, set up the Name and Address heading as above, and continue entering names. Names can be up to 28 characters long and numbers up to 18 digits. You can change these lengths to your taste by altering the tab settings.
As you add more names later on some might spill from page 1 to page 2. You can paste them into the right place quite easily. Print both pages on the same sheet of paper being careful to align the paper in the same position both times.
David Mason

Going for a scroll

One of the most frustrating things about using the [STOP] key to stop a BASIC listing scrolling is that it is not

Connect Four

I enjoy playing it (Type-in Issue 31) but I have made the following alterations which stop the board from being re-drawn with every move. My amendment inserts only the new move into the game but still draws a new board for a new game. The alterations are in bold characters as follows:

```
320 PRINT cls$: FOR w=1 to 1000:NEXT w: win=1
650 PRINT S$;S$;S$;home$;
790 PRINT:PRINT"        MY MOVE. . ."
795 PRINT "     "
```

J.G. Boers, North Yunderup

possible to restart from that point. CONT only works if the program is being run, not listed.
However the listing can be stopped and re-started by pressing [ALT]+S - very useful if long listings need examination.
[ALT]+A is also useful. It recalls the last line you typed in. Very handy for those occasions when you type in a lengthy command line press [RETURN] and find you have the wrong disc in or have printed one or two letters wrong. Instead of laboriously typing everything in again you type [ALT]+A and the last line appears again as if by magic. You can then correct it if necessary with the cursor keys and delete keys before pressing [RETURN].
[ALT]+W does the same thing in CP/M.
John Sorsby

Towards edible hardware

If you put a box of A4 listing paper behind the printer, the paper feeds in nicely but the output does not stack well on top of the box, unless a) the back support is at a lower angle, ie reversed b) a tube of Rowntree's Fruit Pastilles is placed in the trough between the platen and the back support.
A full tube seems to have just the right mass to make the fanfolding work properly. The problem is watching the pastilles go round without taking any out to eat.
G.E.B Russell

We advise readers not to attempt to eat other items of hardware except under strict medical supervision - Ed

Down by numbers

It is usual to label lines in BASIC programs in multiples of ten to leave space for the inevitable insertions later on. However, using the RENUM function, you can add lines anywhere you like and renumber automatically. Suppose you want to insert 20 lines between lines 100 and 110. With the usual increment of 10 for each line, line 110 should now be 310. Simply type RENUM 310,110 [RETURN]. When the desired lines are in and working then type RENUM [RETURN] and all the lines are listed in conventional order. And you don't have to worry about the effect on GOTOs, GOSUBs and the like - RENUM remembers them and adjusts them accordingly.
Gerald Lewis

LocoScript database

Here is a simple method of exporting information from a database (in ASCII form) into LocoScript which allows you to easily convert the information into a table in your LocoScript document.
An extract from a club membership list, when exported in ASCII format, might look like:

```
0101
Smith, J
Sydney
$10.00
0102  (End first record)
Brown, N
Perth
      (End second record)
0103
Jones, E
Darwin
$10.00  (End third record)
```

where each line ends with a [RETURN]. This can be pasted into a LocoScript document which has a template set with the required tabs and headings for your table. By editing the file to replace some [RETURN]s with tabs, and

adding appropriate headings, a neatly formatted table of data is produced. With a short document replacing the [RETURNS]s is a easy enough. But with longer files it can be time consuming. However, by including in the original database a field with only one character - for example Z - this character can be exported to the ASCII file between each field in the above list. Just miss out one between the last field of one record and the first of the next. The exported ASCII file then looks like

0101
Z
Smith, J
Z
Sydney
Z
$10.00
0102
Brown, N
Z
Perth
Z

Using the [EXCH] feature of LocoScript to 'Find', [RETURN]Z[RETURN] and 'Exchange' this throughout the document for [TAB] the document relays itself and to look like:

Member Number	Name	Town or City	Fees paid
0101	Smith, J	Sydney	$10.00
0102	Brown, N	Perth	
0103	Jones, E	Darwin	$10.00

Using the Find/Exchange feature in this way is slow, but it is automatic - allowing time for the usual cup of coffee or two.
I.M. Wyles

Protext's split personality

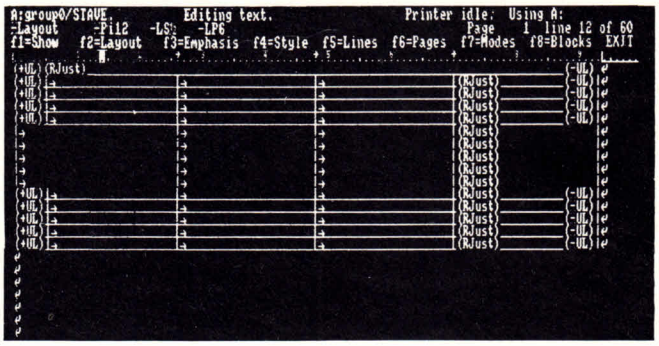
One of the assets of Protext, apart from its ease of use, is the fact that it can hold and work on two different documents in memory at once. This could be valuable if you want to load data, swap over to the the other document to write your article, and pull out information as required. Open a file on the required subject and then type in the information as your research progresses. If, for example, you are writing an article about cats, your data document can be created first. Into this you write the data in clearly defined blocks leaving blank lines between items, and under sub-headings such as Persians, Tabbies etc.
A reasonable size data file can be built up by this method and if an index on sub-headings is included at the beginning then by using the fast FIND facility the items required can be quickly located.
The command to swap to the other working document is [ALT]+Y - if you're unused to Protext and suddenly find your entire text has disappeared for no reason, you've probably hit these keys by mistake. Your precious 50k document has not been erased! Just press [ALT]+Y again (or SW for SWAP in command mode for the same effect) and the original document will reload. To copy a block across from one document to another, first mark the block in the normal way using SHIFT++ at the beginning and end, swap to the other document, press [ALT]+O and the required block will appear at the cursor position.
Another application for the two-file editing facility is just to use one of the files as jotting space.
When an idea occurs to you, swap to the second file, type it in and swap back. At the end of the session, print out the results.
P. Stephenson

At last... ManuScript 2

You can print your own music manuscript paper with bar lines by setting up a file in LocoScript by using:

Pitch	12
Line	6
Line Spacing	1/2
LH Margin at	0
RH Margin at	93
Tabs at	26,48,70
Page length	70
Body	60
Header zone	6
Position	7
Footer zone	4
Position	66

Then type
Line 1: (+UL)(+Rjust)(-UL)[RETURN]
Lines 2-5: (+UL) then | at beginning, tabs, and end (+UL)[RETURN]
(The | is produced by [EXTRA]+ full stop)
Lines 6-10: | at beginning, tabs, and end [RETURN]
Lines 11-15: (+UL) then | at beginning, tabs, and end (-UL)[RETURN]
Lines 16-22: [RETURN]
This can be done much quicker by copying lines 2 and 6 as phrases.
This gives you one musical stave with four bars per line, the first being slightly wider to take the clefs, key signature and time signature. You can adjust the spacing of bar lines or increase the number of bars per line by altering the Tab settings in the header.
You then copy lines 1-22 as a block and paste repeatedly to make up to six staves to a page.
Les Gutteridge



Special K

Users of Protext can get an extra 5k on their data discs by formatting them with DFORMD, Protext's own formatting function used for copying the spell checker dictionary around. In fact this formats discs in the same way the Amstrad CPC6128 computer uses them. Instead of the usual 173k your disc will now take up to 178k! You can't use them as start of day discs though, and DISCKIT won't copy them either - you will have to use PIP to back up the files on such discs.

Your home-grown Tip-Offs should be sent for sharing with other PCW users to:

The Editor
The Amstrad User
1/245 Springvale Rd
Glen Waverley
Vic 3150

Using SID to alter SuperCalc printer codes

by Helen Bradley

This article is being written in part as a solution to the problems of getting SuperCalc to print in other than condensed typestyle and in part as an example of how computer problems can be solved step by step in the hopes that this can remove some of the mystery which people less familiar with computers find frightening.

The problem I set out to solve was to cure SuperCalc's annoying habit of printing all spreadsheets in condensed text. This can, I understand be changed by resetting the printer default codes using the SC2 INSTALL program but I haven't tried that. Putting printer commands in when using the setup functions available in /Output do not work for reasons outlined below.

The way I solved the problem was to change my SC2 program itself. SC2 sends an ESC @ code to the printer when it loads and this, of course has the effect of resetting the printer (see CP/M section Book 1 page 130) and any codes inserted using the setup options under /Output are effectively ignored. I thought that if I found this code then I could disable it so the printer settings could be loaded before loading SC2 and would not be reset by it. As it happened once I was closer to solving the problem I opted for another solution but that was the idea I started out with.

The first step was to find the elusive ESC @ code. Because I knew that SC2 sent the ESC @ code to the printer it seemed reasonable to assume that somewhere in SC2 the hexadecimal code representing ESC @ would appear. From the manual page 122 & 113-118 I obtained the hex numbers representing ESC and @ which are 1B and 40 respectively. So I was looking for the numbers 1B 40 in that order somewhere in the SC2 program.

To find this code using the method I used you should do the following:

1. Load CP/M by putting side 2 of your LocoScript disc in drive A. If you have done as the manual suggests you will have renamed PROFILE.ENG to PROFILE.SUB and CP/M will boot automatically.
2. Remove the LocoScript disc from drive A and in its place insert side 1 of the other CP/M disc - the one with the programming utilities on it. Use pip to copy across onto drive M the file SID.COM by typing:

```
pip M:=A:SID.COM
```

3. Remove the CP/M disc from the drive and type **M:** to change the working drive to drive M then place your SC2 disc in drive A.

At this stage it might be appropriate to suggest that you should not be using your original copy of SC2 to do this to. You should be using a backup copy only just in case something goes wrong as the designers of SC2 probably would not like you doing what you about to do to their program and are unlikely to replace your original if it gets damaged doing this.

4. Check now that your copy of SC2 is able to be written on by ensuring that drive M contains a copy of DIR.COM and then with the SC2 disc in drive A type:

```
DIR A:[full]
```

If the SC2.COM file has RW opposite it in the 'attributes column' then it is able to be written on. If not you will need to PIP a copy of SET.COM from side 2 of your LocoScript disc onto drive M and with SC2 in drive A at the M> prompt type:

```
SET A:SC2.COM [RW]
```

This will enable you to write onto your SC2 program.

5. Next enter SC2 into SID (the Symbolic Instruction Debugger) so the changes can be made by typing:

```
SID A:SC2.COM
```

The computer will then provide you with a reply something like this:

```
NEXT MSZE PC END
7080 7080 0100 DAFF
```

These are values of various memory locations and the program counter. The prompt here is # so don't be surprised to see it on the left of your screen.

6. To see what SC2 really looks like now simply type: **d0100** and you will see the screen fill up with 16 lines of numbers. Look at this for a while. On the far left of the screen you have a column which gives you the memory location of the first hex number in the middle block of numbers which are the actual hex numbers representing the bytes in the memory locations. On the far right is the ASCII value of each byte where the hex numbers are able to be represented in ASCII format.

To find the printer control code either look for the hex values 1B 40 in that order in the hex columns or (a far easier method) look for an '@' symbol in the right hand column and then check whether the number 40 which will appear in the hex column everywhere a '@' appears in the ASCII column has a 1B in the same row and one column to the left of it. (The hex numbers read left to right and from top to bottom of the screen).

With luck and good eyesight you should not need to look far as it is in the first screenful of information in the line starting with memory location 01A0. If you like you may want to have a look at the rest of the program to see what it looks like. You cannot do any harm to it so press 'd' and then the <RETURN> key to see each subsequent set of 16 lines and explore away.

When you are ready to go on look again at memory location 01A0 by typing d01AC <RETURN>. Look at this byte and the next seven bytes. If you are unsure how to count using hexadecimal notation then study the hex columns of the ASCII table in your manual starting at page 113 and familiarise yourself with the sequence 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F, 10, 11, 12...etc.

The eight bytes numbers that we are looking at are the eight

printer control codes which the program sends to the printer and in my installation of SC2 these are:

1B 40	1B 43 42	0F	1B 63
ESC @ resets the printer	ESC C (dec)66 sets page length to to 66 lines	(dec)15 condensed text	ESC c continuous paper

These codes have been translated using the printer control codes from the manual page 122 - 135 and using the ASCII chart to convert hexadecimal numbers to decimals (manual page 113 onwards).

Your SC2 may be set up differently to this but remember you are only looking at the eight bytes including and following the ESC @ ie. 1B 40.

Now having found out what it was that SC2 was telling my printer to do each time I wanted to print I set about altering SC2 so it would print the way I wanted it to. Remembering that I only had a maximum of 8 codes that could be sent to the printer I set about deciding what variables to use. At this stage I had decided against disabling the ESC @ code and instead decided to leave that there but to alter some of the other codes instead.

Using the printer control codes from page 130 of the manual I decided to:

1. leave the ESC @ code in the program and also the ESC c code for continuous paper;
2. change the other codes for condensed text and 66 lines.

The default settings for the printer are set out on page 130 of the manual so it is only necessary to change those that you do not want. This is because by leaving the ESC @ code in the SC2 program the printer will automatically reset to the default settings. I use paper of length 29.5 cm (11.75 inch) so the default page length of 70 lines at 6 lines to the inch suits me fine. I do, however, like my SC2 printouts to be at 12 pitch so I want 'elite' type not the default 'pica'.

I also want my printouts to be quick and the best quality available so I use bold type and draft printing. Draft printing can be selected by using the PTR key whilst in SC2 so I did not want to waste precious codes doing something I can do elsewhere but I did need to set up bold printing from within SC2.

The changes I required came out as follows:

1B 40	1B 45
ESC M Elite type	ESC E Bold type

and the codes to be changed and their replacements looked like this:

1B 43 42 0F (existing codes)
1B 4D 1B 45 (new codes)

as you can see one of the codes - the first 1B was already there so in fact I only needed to make three changes. Looking at the DUMP screen I located the hex numbers to be replaced at 01AF, 01B0 and 01B1.

I then wanted to use the SID program (the Symbolic Instruction Debugger) to change the hex codes so I experimented a lot and finally came up with something that worked. I found some information on SID in the HELP utility on side 2 of the second CP/M disc supplied with my machine (see page 102 of the manual on how to use it) and some in the book The Amstrad CP/M Plus by David Powys-Lybbe and Andrew R.M. Clarke.

To change the codes I typed the memory location of the first byte to change preceded by a "s" as follows:

s01AF <RETURN>

The computer then repeated this memory location for me and also gave me the hex number that at present filled that location thus I received the following:

01AF 43

and the cursor was placed one space away. To do the foul deed I then typed the replacement hex code and pressed return so the whole line looked like this (my input was simply the last two digit number) and pressing the RETURN key

01AF 43 4D <RETURN>

The computer then gave me the contents of the next memory location and again I typed the required code and pressed return and then I did it for the third time to replace the last piece of code. When the computer gave me the details of memory location 01B2 I simply hit the SPACE BAR once and then RETURN and the computer returned to the hex prompt after putting a "?" on the screen first - not the neatest way to stop the substitution - but it works. So what I ended up with on my screen looked like this (of course <RETURN> and <SPACE> merely show that the RETURN key or SPACE BAR was pressed - it won't appear like this on the screen):

s01AF <RETURN>
01AF 43 4D <RETURN>
01B0 42 1B <RETURN>
01B1 0F 45 <RETURN>
01B2 1B <SPACE><RETURN>

That was almost it. I then checked that it all looked ok by using the dump again and typing:

d01AF

and the bytes were all changed as I wanted them. Before doing cartwheels over the living room floor with excitement at having made the thing work I saved the changes onto my copy of SC2 by typing:

W A:SC2.COM

and the disc whirred a little and then I received a message stating the number of records that had been written. To exit to CP/M I typed <RETURN>.

All that was now required was to check that SC2 would respond as required and of course it did.

I hope that this article will be of use to users other than those simply having difficulty with printing SC2 spreadsheets. I hope it encourages others to look around the utilities on their discs and to try them and see how they operate. I hope also that if you find them useful you write an article or letter to this magazine and share your experiences with us all so we can all learn too.

Best of luck and successful computing.

Amstrad Retailers...

A guaranteed way to success.

Do you sell only on price?

.....or.....

Can you give real back-up and support?
Can you give real advice?
Are you supported by the largest importer of Amstrad software?
Is your importer "on-line" to you and your client for help?
Can you supply virtually any product for the Amstrad?

Let Amsnet International help you.

Our current price list has over 400 items on it. Some for every Amstrad model. At Amsnet International we are committed to the Amstrad. We supply only Amstrad related products. Each of our staff is an expert on many items. There's one of us who can answer your questions on virtually any product.. To help you sell more effectively, this is vital.

Real Advice is backed with experience. Collectively, our three consultants have had over 15 years experience working with home and business computers. All of us have "hands-on" experience with all Amstrad models, particularly the PCW and PC1512. Collectively, we have over 50 years of real business experience.

On Line Help means just that. We have a phone line dedicated to help on products that we supply. The number at present is 075-325464. If that is engaged then try 075-321465. But please; phone only during business hours.

Support. We give limited support on most products. Obviously, with the low cost items, support is minimal. However, with higher priced software, we will always help you or your client to get the program up and running. We have found that most problems occur through not reading the instruction manual thoroughly. We expect that you will help your client with that. If you need to phone, then please have the problem clearly defined, as that will help both of us. If, for some reason, we cannot answer your question, we will refer it back to the author of the program for clarification. In twelve months, we have had to do this only three times!

General Importing. Because we have our own resident agent in the UK, we can supply virtually any product for your Amstrad. Amsnet International does stock a good variety of programs and accessories, but because of the vast number of items available, we cannot guarantee that we will have any item in stock at any given time. If it is not in stock, then delivery is usually less than four weeks, depending on availability. Our import policy is very easy to understand. You pay immediately on receipt of the goods!

AMSNET .... Amstrad specialists. Help to retailers, consultants and users across Australia.

What is available?
Will it do the job?
How much does it cost?
Which machine needed?
Is it easy to use?
Where do I buy it?

Every bit of information you need to be a successful Amstrad retailer is available from AMSNET, a goldmine of information for retailers, consultants and the users.

Aimed primarily at the business and add-market, over the next few months we will be featuring items on accounting, desktop publishing, utilities, add-ons, word-processing, databases, specialist software for tradesmen, dairy farmers, solicitors, accountants, hire companies, small business, investors, repair firms, motor garages.... and the list goes on. This information is available to retailers by writing and asking to be put on our mailing list.

From phoning around, we have found that most retailers have no idea of what really is available for the Amstrad. Consequently, everyone tries to sell a spreadsheet or a database with disastrous results. How many of your business clients really wants to learn to program the computer.

Our vertical market software can solve all your problems with easy to use, ready to run software tools. No programming experience needed. Now you can set up your client with hardware and software and because you're the expert, you won't have to discount to get the sale. Isn't that a better way to do business?

AMSNET helps you make profitable sales. And because AMSNET supply only for the Amstrad, you won't have to worry if the other brand computer store is in opposition. He won't be able to buy the product unless he buys it from you.

If you're an Amstrad User and you would like your Retailer to have more information and help available to you, then show this advertisement to him.

GUARANTEED PRODUCTS ...
GUARANTEED SUPPORT ...
GUARANTEED SALES ...

... GUARANTEES PROFITS

Amsnet International

49, Riversdale Road, Oxenford.
All postal orders to P.O. Box 1319
Southport 4215

Retail phone orders to 075-321465
Trade phone orders to 075-531734
Telex AA43470.

AMSNET .... Amstrad specialists. Help to retailers, consultants and users across Australia.

More on auto-start discs for Basic

by J.G. Boers

In our July '87 issue (page 49), Mr. J.G. Boers gave us the information necessary to create an auto-start disc for Basic.
He now goes a stage further, incorporating the M drive and achieving a switch between CP/M and Basic without losing key definitions.

While I have so far omitted the use of the memory disk, 'M' drive, the following section makes extensive use of it. The 'BASIC' disk has all the command files used so far, but it now needs further files in order to perform functions that will simplify the use of BASIC and these files need to be transferred with the same care as before.

The following ones are required :-

PIP.COM
SETKEYS.COM

The "PIP' file will be used in conjunction with the 'PROFILE' file to transfer other files to 'M' drive. The 'SETKEYS' is used to give four of the function keys the ability to simplify operations in Basic. But before making these modifications the first thing to do is to modify the 'RPED.BAS' file that is on the 'BASIC' disk.

This is easy due to a tip-off by T.F. Potten in the June issue. RPED is an editor written in Basic but with the annoying feature, when working in Basic, that it returns to CP/M. To modify RPED, start up basic, type NEW to ensure that memory is empty then type "save "temp",a". This creates an empty file. Now type " load "rped " as you would to load any other file. Then type " merge "temp " in order to make the protected file readable by removing the protection.

Next is a simple alteration to give the editor the ability to return to Basic. Type in "edit 19" and you will find this line on your screen. Move the cursor to the word SYSTEM at the end of the line. Delete just this word, then type in NEW in its place and press return. Now type "save "RPED",P" in order to save the new file and protect it at the same time. Now, whenever this version is used it will return to Basic.

One thing to remember is that the editor needs an Ascii file so, if you want to edit a Basic file in Rped you will have to save the file first with " filename,a" to make it into an ascii file unless it was created in the editor as this would cause it to be an ascii file. The other catch is that the editor's text layout is not as wide as the normal screen so that, on long lines, you may have to check that the lines are complete.

The next step is to modify PROFILE.SUB in order to run SETKEYS and to transfer BASIC.COM, RPED.BAS and any other files you wish to M drive. The reason for this is to speed up the running of Rped and to allow you to go to CP/M mode and back again without having to re-boot Basic.

The following text is the actual file on my PCW:-

```
setkeys fnkeys.bsk
pip
<m:=basic.com[o]
<m:=rped.com[o]
<
m:basic
```

The first line causes the setkeys command file to set certain keys as to the definitions in fnkeys.bsk file. The next line activates PIP, followed by two pip command lines and a line,<,that finishes the running of PIP. Any other files you want to transfer can be put on new lines. Always start the command line for PIP with < as this causes a carriage return command to be read by PIP.

The last line causes Basic to run. While Profile is running you see all commands echoed on screen. Don't worry about the apparent errors while the keys are being set. These are caused by the difference in command structure between Basic and CP/M but do not interfere with anything.

Next up is the creation of the file FNKEYS.BSK using RPED to do so. Under most conditions the file we are creating would be longer but by using keys that are unused in CP/M or RPED it becomes very simple and has the following text:-

```
E #82 "run ↑'#22'm:rped ↑M"

E #84 "run ↑'#22'"

E #86 "load ↑'#22'"

E #88 "edit "
```

The use of ↑'#22' is because this is the only way that a key definition can use a double quote within itself. The ↑ is found by "extra u" or "extra ;". The keys that are re-defined are F2,F4,F6 and F8 in that sequence. Thus F2 , E #82, causes Rped to run without having to use a carriage return. This is caused by the ↑M at the end of the command line.

The run and load commands only need a filename when used, without needing to type the quote, by pressing F4 or F6.

Finally F8 will allow a quick entering of edit line, but don't forget to put the

space between the t and " as that would mean you have to put it in every time the command is used.

One final file that is handy is: BAS. SUB . It's text is simply :- m:basic. This will allow you, once the basic disk has been booted, to return to Basic from CP/M. To do so make sure that the auto-start basic disk is in the drive and type submit bas.

With the disk you now can continually change between CP/M and BASIC without losing key definitions while Basic loads much faster from the memory disk when changing back to it.

A final item that is often overlooked is that, while writing a program in Basic, you can inspect any disk directory. The directory listing is only put to the screen and not to the listing you are working on. But remember, this can not be done if you are using rped or if the auto-line-numbering is on.

And the winners were . . . .

The saying "out of sight - out of mind" was never so true when we discovered that we had not published the results of our PCW competition in the August issue as promised. The results were in one of those small text files we had not bothered to back-up which had been stolen in the break-in. Of course, the winners know as they received their prizes towards the end of July. They were:

First Prize:
J.M Bruce, Burnie, Tas — **At Last Database Manager**

Second Prize:
B. Brotherston, St. Kilda, Vic — **Money Manager**

Third Prize:
T. Tomlinson, Port Pirie, SA — **Companion**

Our thank to all those who entered, and to International Software Distributors for providing the prizes.

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.

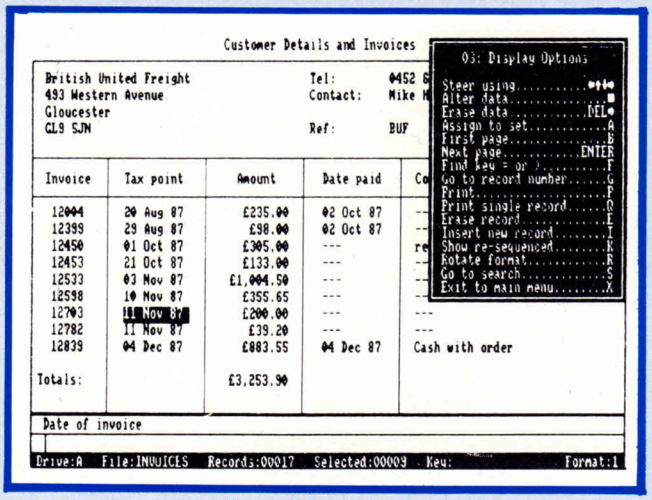
Other products offer a choice between fast but limited capacity RAM files, and large capacity but cumbersome fixed-length, direct access disc files. **MASTERFILE 8000** and the PCW RAM disc combine to offer high capacity with fast access to variable-length data. File capacity is limited only by the size 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.

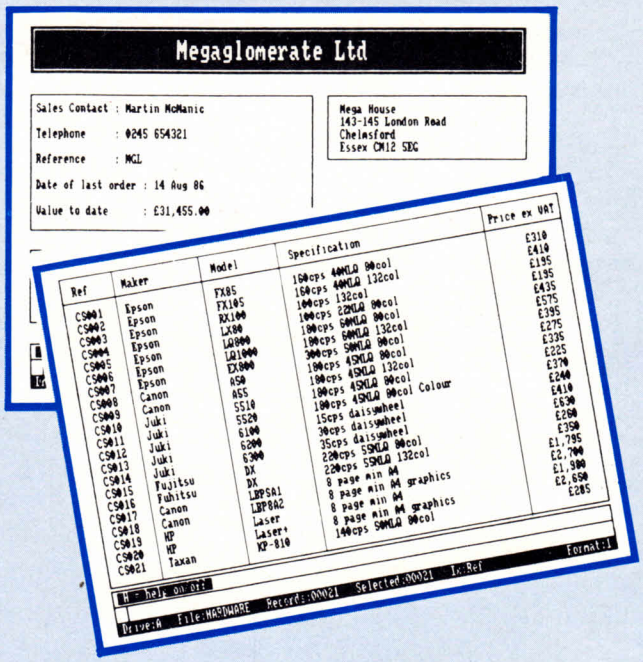


Keyed files are maintained automatically in key sequence, with never any need to sort. You can have unkeyed files too, where records can be inserted at any point in the file.

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 $109.00 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**

Special Relations

Databases come and go, but Masterfile lives on. What does this latest incarnation have to offer?

Masterfile 8000 is proving to be a very popular package, judging by the number of copies sold through The Amstrad User. To answer some of the questions prospective users may have, we are reprinting an article that appeared in 8000 Plus a few months ago.

Masterfile 8000 is the latest version of this relational filing system, and is written especially for the PCW. It takes full advantage of the micro's best features, including the enlarged screen display, graphic capability and RAM disc. The program is completely written in machine code, which means it's fast in operation, and builds on earlier versions of the program. It started life on the Sinclair Spectrum, and progressed through the Amstrad CPC range to where it stands today.

What you get

Masterfile comes in the ubiquitous black plastic folder, which contains a single disc and a 64 page manual. The documentation covers file creation and use, and includes sections on the theory of data files and relational links. It's quite clear, but a little sketchy in places and not written for the absolute beginner (but then, Masterfile is not for the novice, either). Overall, it provides a lot of information.

As well as the manual, extra information is provided in a number of data files on the disc. Once you've worked out how to load and display a file, you can use the files themselves to teach you more about the system. As well as providing some ready made applications (record collections, customer files etc.), these files include ample demonstration of the more advanced features of Masterfile. Even the index and glossary are provided as files, though some of the definitions seem a bit off-beam, viz 'Disc - A quoit thrown by ancient Greek athletes; any flat thin circular body or structure'.

Masterfile face-lift

The most obvious difference between Masterfile 8000 and other databases on the PCW is the graphical content of the screen. Gone is the 'This is a database and shouldn't be fun' syndrome. With Masterfile you can draw little books to illustrate your book records, railway lines to run across your train-spotting records and doric columns to surround your classical record records. They're actually surprisingly easy to draw, as well. You can add lines, boxes and panels to the screen at almost any point.

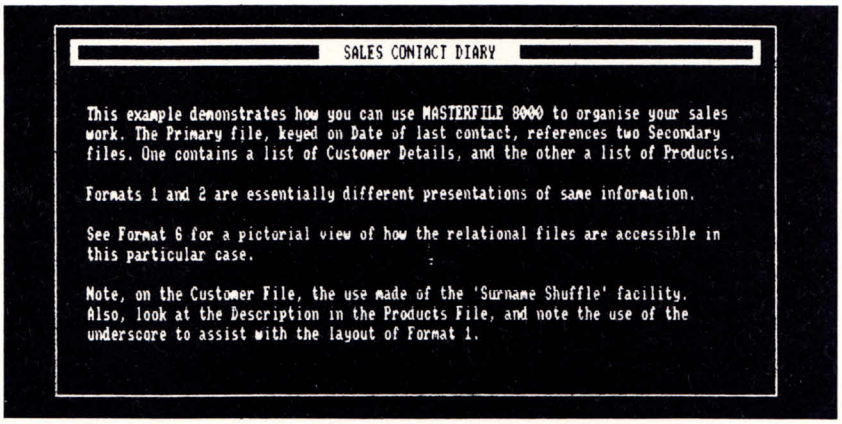
These elaborations are not just gimmicks, but do help to highlight particular areas of the record, and draw attention to specific fields of information.

Filing it down

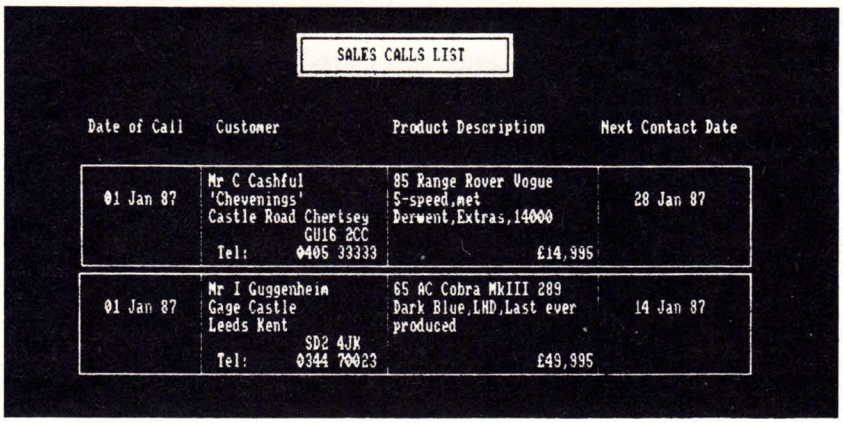
A Masterfile file is created in two main steps. The first is to specify the number and content of the fields themselves, and the second is to design a format for each record in the file. The second step is actually optional, as defining the fields sets up a basic display format, known as format O. This format consists of all the field headings you've typed in and space next to each of them for the associated data.

To define a file, you first enter the number of different pieces of information (the number of fields) you want to put on each record. It's important to specify a large enough number, as once you've started to define your record you can't add extra fields. This is an odd restriction, as most databases allow you to add extra fields as long as there's no data in your file.

Once you've settled on the number of fields, you type a heading for each one, which may be up to 22 characters long, and specify whether the field is to contain a number, a date or simple alphabetical characters. If you don't want to bother with other formats, that's all you need do - you can start



▲ A format screen used soley for displaying a description of one of the sample files

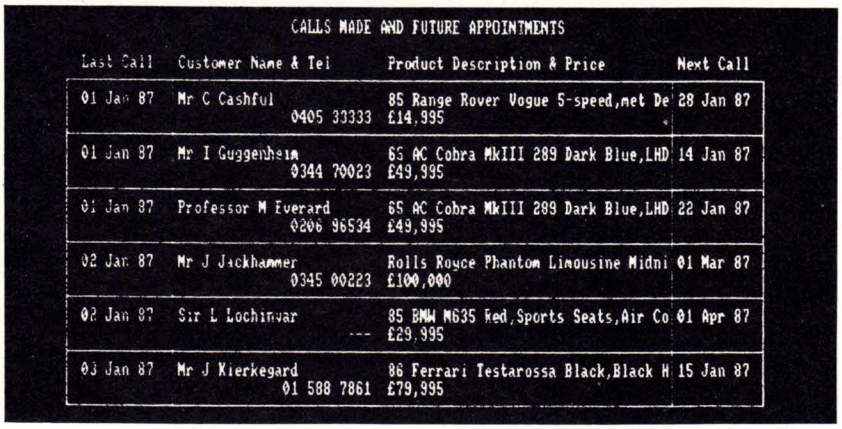


SALES CALLS LIST

Date of Call	Customer	Product Description	Next Contact Date
01 Jan 87	Mr C Cashful 'Chevenings' Castle Road Chertsey GU16 2CC Tel: 0405 33333	85 Range Rover Vogue 5-speed,met Derwent,Extras,14000 £14,995	28 Jan 87
01 Jan 87	Mr I Guggenheim Gage Castle Leeds Kent SD2 4JK Tel: 0344 70023	65 AC Cobra MkIII 289 Dark Blue,LHD,Last ever produced £49,995	14 Jan 87

▲ One format from a car sales database. Note : two records are displayed at once

▼ A different view of the same information. Here six records are displayed.



CALLS MADE AND FUTURE APPOINTMENTS

Last Call	Customer Name & Tel	Product Description & Price	Next Call
01 Jan 87	Mr C Cashful 0405 33333	85 Range Rover Vogue 5-speed,met De £14,995	28 Jan 87
01 Jan 87	Mr I Guggenheim 0344 70023	65 AC Cobra MkIII 289 Dark Blue,LHD £49,995	14 Jan 87
01 Jan 87	Professor M Everard 0206 96534	65 AC Cobra MkIII 289 Dark Blue,LHD £49,995	22 Jan 87
02 Jan 87	Mr J Jackhammer 0345 00223	Rolls Royce Phantom Limousine Midni £100,000	01 Mar 87
02 Jan 87	Sir L Lochinvar ---	85 BMW M635 Red,Sports Seats,Air Co £29,995	01 Apr 87
03 Jan 87	Mr J Kierkegard 01 588 7861	86 Ferrari Testarossa Black,Black H £79,995	15 Jan 87

typing in your data.

You don't need to specify the length of each field, as Masterfile 8000 uses variable length records. This means that each record that is saved on disc is only the length of the data that it contains (plus a few extra bytes which the program uses to locate it). Most databases use fixed length records, which nearly always take up more room on disc, but are easier for the program to alter or delete.

Drawing the line

If you want to make more of the visual display of your records, though, you can define up to nine other formats. These are effectively screen layouts which may contain some or all of the fields you defined in the base format, 0.

You can specify the position of each field, give it a different title from that in the base format, and mark out a display area for the data it will contain.

Each of these functions is controlled from a series of menus which appear on the screen. Although this system guides you through quite a maze of different facilities, it takes a while to get used to the techniques involved, and the menus do tend to get in the way.

When you are adding headings to your layout, it's a little disconcerting to discover that the cursor used for positioning them wipes out anything it passes over. This is rectified when the screen is redrawn at the end of each new addition, but if you're lining up a series of headings you have to be careful not to wipe out others by mistake.

One of the powerful features of Masterfile is that the program will automatically assess the depth of the layout you produce, and display more than one record on the screen at once, if it can. In the extreme, you can have up to 28 records on the screen at once, assuming each takes up only one line of the display.

Working from memory

Unlike many databases, Masterfile 8000 loads the file you're working from into the memory drive before starting. This speeds up many of the programs functions, but does mean you have to remember to save your file back to disc before switching off. Masterfile reminds you to do this, but if you're the victim of a power-cut, you can lose a lot more than the last record you changed.

The other disadvantage is that you're limited to the space on the M: drive, rather than the larger capacity of the A: and B: drives. To take a typical example, suppose you define a membership list, as in the illustration. There are 12 fields in the record, and you might expect the following average lengths for each:

Name	12
Company	16
Address 1	20
Address 2	20
Town	8
State	8
Post Code	10
Tel No	10
Total Subs	5
Inst Paid	5
Outstanding	5
Last Payment	3

If you allow an overhead per record of 8 bytes, plus one byte extra for each field (as suggested in the manual), this gives you a record length of 142 bytes. You should allow 2K for formats and other overheads on the disc drive, so you could expect to store 110x1K/142=793 records of this length on an unexpanded PCW8256, and 366x1K/142=2639 records on a PC8512. If Masterfile worked with the 3" drives, though, you could have stored 1283 on the A: drive and 5177 on the B: drive.

File features

When you come to use your Masterfile database, you display one of the formats you've created and start typing your information into the data fields. Character and number fields accept data in any form, but date fields do some elementary checking. As long as the month isn't more than 31 days long, though, it'll probably get through. There are no calculation facilities in Masterfile, other than a simple totalling,

so numbers aren't checked for size or form.

UPDATE: Field calculation facilities are now available. This means that you can amend existing fields or display a new one being a result of a calculation on other fields.

When you have entered a few records, you'll probably want to browse through them. You can move forward record by record, and return to the start of the file, but oddly you can't move backwards through a file. Records are normally displayed in the order of the field you defined as a key when you set the file up, but you can temporarily define an alternative key so that they will appear in another order.

The main facilities within Masterfile 8000 are 'Find' and 'Search'. Find allows you to specify a search word in the key field, and Masterfile then hunts through the records until it finds a match.

Search is rather more complex, as it allows a 'rule' to be set up, which may contain several conditions referring to a number of fields. You could, for instance, define a rule which said 'Name is Johnson, Town is Margate and Total Subs is less than $100'. All records which match that rule would then be 'selected', and only these would be displayed, until you selected the whole file again.

You can define a number of different formats for the printed documents which Masterfile can produce. They are based on the screen formats, and since you can define up to nine of these, you could set up several of them specifically for printed output. Again, Masterfile will output as many records as it can per printed page.

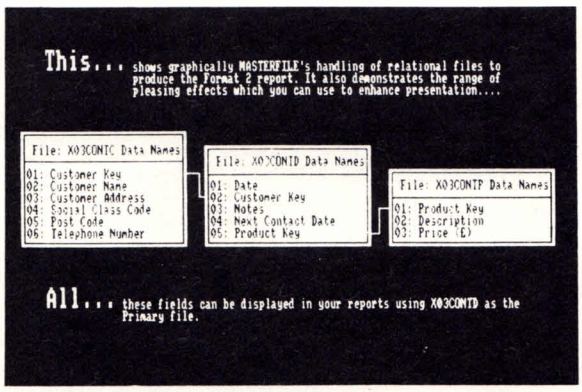
One of the Campbell System's proud boasts is that Masterfile has always been a relational database, which means it can refer to data in more than one file at once, and these files can be linked by the use of common fields. This relational facility saves disc space and entry time. It won't always be of immediate use, but it's nice to know it's there if you need it.

Verdict

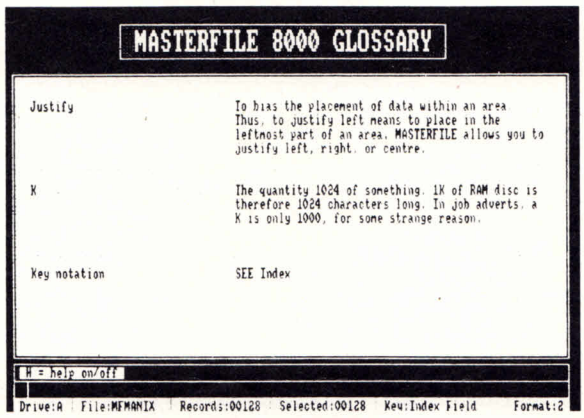
Masterfile 8000 shows its sound ancestry, being from a series of good, value for money databases. It is quick and versatile, has plenty of scope for flexible screen and print layouts, and has good selection and sorting facilities.

It works fast by dealing exclusively with the PCW's RAM drive, but this does limit its capacity. If your application is large, and you own a PCW8256, you may have to think of upgrading its memory before installing Masterfile.

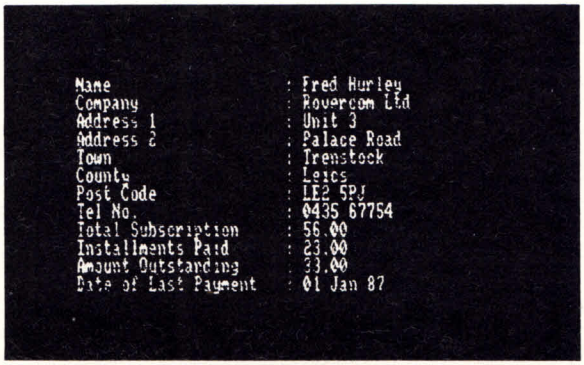
UPDATE: A PC version of Masterfile is under development and may be released at the end of this year.



▲ This format explains how three files are related



▲ A page from the Glossary file



▲ A typical record display in format 0 of a membership file

Relational files

Imagine you are running a database to hold a file of books supplied by the Marmoset Fanciers Book Club you run. Each record in the file might contain the book's title, it's author, it's publisher, marmosets covered by the book, the date of publication and the price.

Another file contains details of customers. Each has a name and address, and details of the books that have been supplied to them. One of the uses of this second file is to prepare invoices. To do this, an essential piece of information is the price of each book. You could hold these prices on each record of this file, but it would save space on the disc to be able to read them directly from the other one. It would also mean that you only have to enter the process once.

A relational database, such as Masterfile 8000, can do just this. By making the title the 'key' field (the one on which it's indexed) in the book file, Masterfile could extract any of the information in that file. None of the book details can be altered, but they can be viewed and printed out just as if they were part of the customer file itself.

Masterfile 8000 can link up to eight files together in this way.

Re-count

An update to our original Word Counter program - it now reads LocoScript files!

from Robert Gardner

"Count.bas" is a word counter for Locoscript. It originally developed from the program printed in the November 1986 issue of the Amstrad User, however the current version bears no relationship to this program, as an inspection of the listing will show. My program has two important advantages over the original, namely that it works on Locoscript documents without the need for them to be made into an ASCII file, and also the fact that it will locate the file to be counted from any one of the eight user-areas that Locoscript divides the disc into. As a result, there is no need to take any special steps before the program can be run.

The listing can be divided into several main areas, some of which may require further explanation:

Lines 50 - 90: check all the user areas (1-8) for the file; if the file is not found the computer beeps and returns to the beginning of the program. It should be noted that this step is very important, since if the file does not exist the computer does not worry but continues to happily count the non-existent file ad infinitum.

170 - 220: find the start of the text. Locoscript actually stores the information for the inspection of the document (f2) before the document itself, and if this check is not carried out the number of words counted will include those in this area. Start$ contains a group of numbers that appears before the text in every document I have ever encountered.

260 - 310: count the number of words that occur before the next 256 characters are fetched and the main routine can begin.

330 - 410: is the main routine. EOF detects when the last record is reached (which is actually the end of the file).

420 - 480: simply tidy up the end of the program, setting all values back to their default, and stating how many words have been counted.

500 -560: is the main count routine and is basically similar to that of the original program. Line 510 checks that there are no 'return' characters in the group. Line 540 ensures that characters used by Locoscript as identifiers are not counted as part of a word.

600 - 690: allow for return characters ending words instead of the normal space. I found that this was necessary in order to make the number of words counted more accurate.

As a point of interest, I found it necessary to open the document as a random access file because Locoscript uses Control-Z characters throughout the document, and these indicate that the original word count program did actually count words in a Locoscript document, but it always seemed to considerably underestimate the number of words. This was a result of the first Control-Z character appearing immediately after the inspect document information, and this is what was actually counted.

From running these, I have found that the word counter overestimates the number of words by about 1%. This error is associated with the headers and footers for each page, and cannot be easily fixed, but I am sure you will agree that 1% error is not excessive.

```
10 MEMORY ...256
20 INPUT "File name to be counted";file$
30 check$=FIND$(file$)
40 w%=1
50 WHILE check$=""
60 OPTION FILES(STR$(w%))
70 check$=FIND$(file$)
80 w%=w%+1:IF w%>8 THEN PRINT CHR$(7);"File not fo
und":GOTO 20
90 WEND
100 check$=STRIP$(check$)
110 start$=CHR$(0)+CHR$(136)+CHR$(8)+CHR$(134)
120 OPEN "R",1,check$,256
130 FIELD 1,16 AS c$
140 wds%=0:ret$=CHR$(136)+CHR$(2)
150 PRINT
160 PRINT "Counting.....Please Wait"
170 GET 1
180 FOR a=1 TO 16
190 LSET c$=INPUT$(16,#1)
200 IF INSTR(c$,start$) THEN 230: REM Search for s
```

```
tart of text
210 NEXT a
220 GOTO 170
230 inwd%=0
240 line$=RIGHT$(c$,11)
250 GOSUB 500
260 WHILE a<16
270 LSET c$=INPUT$(16,£1)
280 line$=c$
290 GOSUB 500
300 a=a+1
310 WEND
320 w%=0
330 WHILE NOT EOF(1)
340 GET 1
350 FOR a=1 TO 16
360 LSET c$=INPUT$(16,£1)
370 line$=c$
380 IF ASC(line$)=0 THEN w%=w%+1
390 GOSUB 500
400 NEXT a
410 WEND
420 PRINT:PRINT
430 PRINT wds% "words counted in ";file$
440 CLOSE
450 OPTION FILES "0":REM return to default user nu
mber
460 PRINT:LINE INPUT "Do you wish to count another
 file (Y/N) ",check$:PRINT
470 IF UPPER$(check$)="Y" THEN 20
480 END
500 REM ------ count subroutine ----->
510 IF INSTR(line$,ret$) THEN 610 ELSE IF RIGHT$(l
ine$,1)=CHR$(136) AND inwd% THEN wds%=wds%+1
520 FOR i=1 TO LEN(line$)
530 e$=MID$(line$,i,1)
540 IF e$ =CHR$(129) AND inwd% THEN wds%=wds% +1 :
inwd%=0 ELSE IF ASC(e$)<=128 OR ASC(e$)>=159 THEN
inwd%=-1
550 NEXT i
560 RETURN
600 ------- RETURN subroutine ------->
610 b=INSTR(line$,ret$)
620 FOR i=1 TO b
630 e$=MID$(line$,i,1)
640 IF e$ =CHR$(129) AND inwd% THEN wds%=wds% +1 :
inwd%=0 ELSE IF ASC(e$)<=128 OR ASC(e$)>=159 THEN
inwd%=-1
650 NEXT i
660 IF inwd% THEN wds%=wds% +1:inwd%=0
670 IF (LEN(line$))-b<=2 THEN RETURN
680 line$=RIGHT$(line$,(LEN(line$)-b-2))
690 GOTO 500
```


Many thanks to readers who have sent us their updated versions of Word Count, Mastermind, Connect4 and others. We have looked at most of them (those that were on disc at least - we can't make time to key the others in) and clearly the originals have seeded some good ideas. In some cases, as no clues were given, a comparison between the original and the update was necessary to work out the changes that had been made.

However, it is unlikely that any will be published as we feel other readers may get a little bored with seeing a string of "variations on a theme".
If you think you've come up with something really clever, and it's not too long, put the line amendments in a letter and we will get the message across that way.
NEW PROGRAMS: Please remember to send your program on a disc - we will not re-key it if it's too long - and don't forget to include adequate documentation.

Making it count

Continuing our series opening up the dreaded 'Advanced Utilities' section of the LocoMail manual.

In the first episode of this mini-series we looked at the conditional printing facilities - how to include or omit chunks of text depending on who you are writing to. This month we add a few frills to the process and delve into making LocoMail do your arithmetic for you.

The LocoMail manual is strange. The first half takes you gently through the simple process of running off a form letter, then just as you think you have really sorted it out it suddenly hits you with all the weird symbols like #, <, ;, and *.

Last month we covered the commands which let you customise the text of a letter depending on who it is to and ended up with a letter something like this:

If this is a friendly letter:
print this friendly text.
If this is not a friendly letter:
print this other text.

Or, to recap on LocoMail's 'simplified' way of doing things:

```
(+Mail)# friendly="y" : <(-Mail)
friendly text
(+Mail)>(-Mail)
(+Mail)# friendly="n" : <(-Mail)
other text
(+Mail)>(-Mail)
```

There are still a couple of refinements to make life easier. The first thing to note is that split over the third and fourth lines are a pair of adjacent (+Mail) and (-Mail) signs. These are redundant, since the one effectively cancels out the other and you would be better to omit them both.

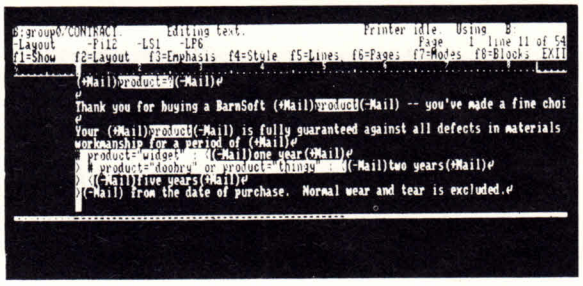
Any character outside a Mail command will always appear on the final printout but those inside the command are not printed. So, once the redundant (+Mail)/(-Mail) are eliminated, you have the advantage that the [RETURN] character at the end of line three is now inside a Mail command. This means it does not appear in the final printed letter, which probably suits your page layout better.

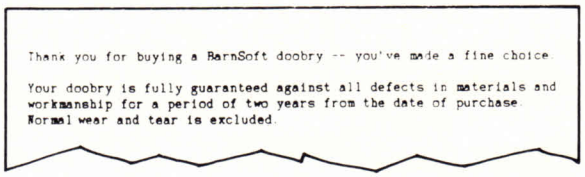
Now for the clever stuff. Consider the problem where you are churning out a standard contract of sale for one of your products. Normally you give a five year guarantee, but widgets only get one year, and doobries and thingies two each.

The way of doing this is shown in the screen shot, and if you can master these commands then you have total control of conditional printing. In particular, there are three new things here.

First remember, as ever, that any [RETURN]s which you use outside the (Mail) symbols (ie. which don't appear in reverse video on the PCW screen) will appear in your final document and may mess up your page layout.

Second, see how the text for the 'doobries' and 'thingies' has been dealt with in the same go by combining them with an 'OR'. The way this works is fairly obvious if you read the command as English - "if the product is a doobry OR a thingy, print this." You could use 'AND' in the same way - "if the product is a thingy AND there is a maintenance contract, print this."





▲ Try out this example in LocoMail's "Fill" mode, and set 'product' up to be a 'doobry', 'thingy', 'widget' or 'wotsit' and see what happens

Finally, and most importantly, the case for the common or garden products comes at the end, without any conditions. LocoMail will print this only if none of the previous conditions apply. So the most general form of conditional printing is:

```
(+Mail)# first condition : <(-Mail) first text
(+Mail)> # second condition : <(-Mail) second text
```

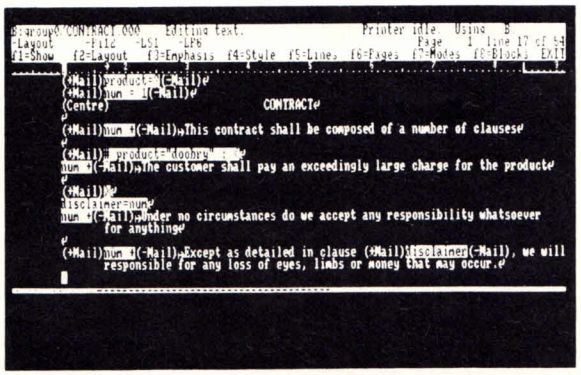
```
(+Mail)> # third condition : <(-Mail) third text
... other conditions
(+Mail)> <(-Mail) text if none of the above conditions
apply
(+Mail)>(-Mail)
```

One, two, three...

And now for something completely different - getting LocoMail to do your sums for you.

If you write the kind of letters in which you number the paragraphs, perhaps for a contract or other formal document, then you can get into trouble when you start playing around with the conditional printing commands. The problem is that you may have clauses that only apply to certain kinds of client, and therefore you either want to leave out parts of the document or include specialised ones.

This may present no problem if you can arrange all these special clauses after all the standard ones that are always there. However, if you want to include an extra clause for someone after clause 3, not unreasonably numbered 4, then all the subsequent numbers will have to be shunted up. Fortunately LocoMail provides a special way of doing this with its variables.



▲ This contract allows you to insert an extra clause if the product is a 'doobry', while keeping track of the numbers. Try it out in 'Fill' mode setting product to doobry, and then something else.

In your standard document you would actually type in 1,2, 3 and so on beside the appropriate paragraphs. The trick is instead to use a LocoMail variable to hold the number of the current clause. If this variable is called 'num', for argument's sake, then you will recall that the command

```
(+Mail)num(-Mail)
```

anywhere in a LocoMail document will cause the current value of 'num' to be printed at that point.

Here's the clever part. Whenever you want to put a number down for a clause, use the command

```
(+Mail)num +(-Mail)
```

which means, "print the number held in 'num' here, and then add one to 'num' so that it is right for the next clause."

All will become clear (don't laugh) by looking at the example in the screen shot. Of course you must tell LocoMail what the number of the first clause will be by explicitly setting the variable 'num' up. Most people find that a good point to start is with clause number one.

```
(+Mail)num = 1(-Mail)
```

Fine, you say. But what happens if you want to refer to a specific clause, by saying "as specified in clause 3" or something? If the clauses are being numbered by the automatic method you won't know when you type up the LocoMail template what number a particular clause will have when it comes to be printed out.

The solution is to use yet another variable. You have to store the number of the clause you wish to refer to in a temporary variable, and then you insert this variable in the text later on when you need to know it. In the example, the line

```
(+Mail)disclaimer = num(-Mail)
```

saves the current value of the variable 'num' in another variable called 'disclaimer'. Then, when referring back to the disclaimer clause in the text, you could say something like "all eventualities are covered except as detailed in clause (+Mail)disclaimer(-Mail)".

Go forth and multiply

This business of LocoMail being able to add one on to a clause number each time that you print it leads into the larger area of arithmetic.

If your document is an invoice, for instance, then you can get LocoMail to do all the calculations for you. LocoMail can take numbers (or variables that contain numbers) and add, subtract, multiply or divide them This means that you can automatically work out discounts, add up bills and so on.

Remembering back to your school maths days textbooks were full of questions like 'what is 123 (divide symbol) 3', 'what is 789X1.15', and so on. LocoMail can understand sums written down in this form too. Computers use special symbols for division and multiplication: because there is no division key, the slash symbol, /, is used for division instead, and to avoid confusion between lower case x and the standard multiplication sign an asterisk, *, is used for multiplication.

To tell LocoMail that there is a calculation coming up, you must enclose the commands in a pair of square brackets, and of course the inevitable (+Mail)/(-Mail) commands. So a line in a document reading

```
(+Mail)[69.95*1.15](-Mail)
```

would insert 80.44 at that point in the final printed document. This isn't very useful in itself, since you might just as well work out what 69.95X1.15 is, and type 80.44 into the document straight away. The power of LocoMail's arithmetic is that you can work on the variables that you have set up in your mailmerge data.

Rather than printing the result out straight away, you could

store it for use in a calculation later on. To do this you need to pick a variable name (good old 'fred', for argument's sake in this example) to hold the result, and use the command

```
(+Mail)fred=[69.95*1.15](-Mail)
```

One important difference between working something out and storing it in a variable is that when you store it the value is not printed out - you have to do it explicitly in a separate command. You can now use 'fred' in a later calculation like

```
(+Mail)[fred+10](-Mail)
```

This is relevant when dealing with tax. If you are doing an invoice run and your data records consist of customer's names, addresses and amounts owed, you can print an invoice showing the tax exclusive amount, the tax, and the total. A new variable called, say, "tax", is set up to hold the amount of tax, and that is used again when working out the total amount.

Next month, in the final exciting instalment of the LocoMail story, we'll go through a complete example application showing LocoMail organising an order book.

When is a number not a number?

'Variables' are LocoMail's way of remembering things. You can tell it to store numbers or text in them. This means that by choosing a suitable name you can re-use the same information in several parts of the same document.

However, now that you have read this month's article and discovered the beauty of arithmetic in LocoMail, you ought to think carefully about what goes on behind the scenes. This may save you much heartache when LocoMail comes up with one of its friendly yet concise error messages like 'command type mismatch'.

It may sound like stating the obvious but numbers and letters are not the same thing. Whereas you can ask LocoMail to work out 2+2, it isn't too happy when asked to work out "peas"x"carrots".

More realistically, you can work out 19.95*1.15 - to add 15% sales tax on to a $19.95 bill - but not $19.95*1.15. This is because 19.95 is a number, but the presence of the $ sign means that $19.95 is treated as text.

The moral is that whenever you are setting up a Loco-Mail data document with lists of numbers in them make sure that all the numbers are typed as numbers, ie that they don't have any characters other than the digits 0 to 9 and a full stop. Otherwise any arithmetic you do is likely to go wrong.

Quick Pick Type-in

J. Hellis takes Raffle Draws into the 21st Century

We know probably the easiest thing in the world to do it to pick raffle tickets out of a hat, but things are always more impressive when computers do them.

This simple program just asks you the two important questions, "How many tickets have been sold" and "How many prizes will there be". Assuming your tickets are numbered from 1 to the total number sold, the program then produces a list of which ticket number has won which prize.

Think how flash it would look down at the club, everyone clustered around the PCW waiting to find out whether they've won the box of biscuits donated by the President's wife. It certainly would speed up the agony of most local raffle draws. The other important feature is that it actually prints out a list of winning numbers so that you can remember who has won what when you sober up the next day. That must be worth the effort alone.

You may have seen the faithful old randomise facility before (see line 110) and then there's INKEY$ to make the whole proceeding flow. All this in only 19 simple lines.

Why wait for the next raffle - start listing now.

```
10 PRINT CHR$(27)+"E"+CHR$(27)+"H":d%=0
20 INPUT "How many tickets have been sold -";t%
30 PRINT: INPUT "How many prizes do you have to pr
esent -";p%
40 IF p%>t% THEN PRINT:PRINT "SELL MORE TICKETS TH
EN!":GOTO 20
50 PRINT:PRINT"Do you want the prize numbers on th
e (S)creen or (P)rinter."
60 INPUT z$: IF UPPER$(z$)="P" THEN d%=1
70 PRINT: PRINT "Press the <Space Bar> to select t
he winners."
80 PRINT:PRINT
90 FOR b%=1 TO p%
100 WHILE INKEY$=""
110 a%=INT(RND*t%+1)
120 a$=STR$(a%)
130 FOR c%=1 TO b%
140 IF INSTR(e$,a$)<>0 THEN GOTO 110
150 NEXT c%: WEND
160 e$=e$+a$
170 IF d%=0 THEN PRINT"Prize no ";b%;" goes to tic
ket number ";a% :GOTO 190
180 LPRINT "Prize number ";b%;" goes to ticket num
ber ";a%
190 NEXT b%
```

Wizard Games

A delve into pursuits of fantasy and fable

Lord of the Rings - Melbourne House

Hobbits as we all know, are furry little creatures whose favourite pastime it seems is going off on long journeys which last at least 3,000 pages. If only Tolkien had been better at precis...

Now the computer game is not quite that long, covering as it does only the first part of Tolkien's trilogy - The Fellowship of the Ring. Having said that, it will probably take you as long to complete the game as it would to read all three volumes three times over!

For anybody who's been in an Outer Mongolian monastery for the last 50 years, here's a little background. Lord of the Rings is set in Middle Earth, a strange place inhabited by all manner of weird and wonderful creatures, indeed, not that much different from the House of Parliament.

Balrogs, dwarves, elves, orcs, trolls, wargs and wolves, you name it, they're all there. The orcs are particularly charming. It seems they can't say even hello without slicing someone's head off.

The scenario, at its most simple, is this. Sauron, the baddie, has developed an unhealthy fetish for rings, particularly magical ones. Now it just so happens that Frodo, the goodie, has inherited one from his eccentric uncle Bilbo (a name to be read very carefully if you're dyslexic). Slightly peeved by this, Sauron has despatched his Black Riders to lop off little Frodo's head and take the ring. Consequently, Frodo must leg it as quick as he can to Rivendell where Elrond, the Elf-King, will advise him what to do.

You assume the role of Frodo or one of his three friends, Sam, Merry and Pippin, who accompany him on his dangerous quest. However, it is possible to control all four characters and to switch from one to the other by simply using one of three simple commands.

Ironically, all four hobbits are fairly human. They get tired, hungry and usually die when a sword or some other such weapon pierces their vitals. This means that you have to allow your characters plenty of food and rest, particularly if they've recently been engaged in anything strenuous.

The game comes in two parts and there is also a beginner's game for those unfamiliar with adventuring. Although it is advisable that you complete part one before adventuring on to part two this is not compulsory.

Knowing Tolkien's work might prove useful but it is not essential and might even be misleading. There are no pretty graphics screens, but this compelling adventure game triumphs in its own right and will delight and baffle in equal quantities.

Available through The Amstrad User at a pre-Christmas price of $39.95

Distractions - Design Design

Design Design like Duran Duran have a slight problem. It seems two is always better than one. I said two is always better than one. Now they've gone one step further by putting three games onto one disc, which should add up to good value for money.

Mutants, mazes and mushrooms just about sum up the first game in this rather incongruous trilogy. In On the Run, replete with protective suit and jetpack, you find yourself in a maze with only an hour in which to find seven flasks. These are not ordinary flasks, of course, but contain toxic chemicals.

Out to stop you are a host of malicious mutants from giant jellyfish to noisome gnashers. Contact with any of these sap your energy, which must be continually replenished by picking up the various objects that are scattered about the maze. This game is simple but very playable. Though the graphics are large and for the most part quite detailed, the flip screen is very irritating. Moreover, the title seems a bit of a misnomer - shouldn't we see Ronald Biggs or some such personage trotting across the green screen?

The second game, 2112AD, is a little dry. A computer has assumed a mind of its own, a scenario that anyone with a PCW can relate to. Accompanied by your faithful robo-hound Poddy, it's your task to collect ten code pieces (no, not codpieces), scattered throughout the computer complex.

Along the way you must avoid a number of mechanical monsters, who quite rightly assume the computers are superior to humans. Gather the code pieces together, insert them in the right order and the game is won - simple!

Initially, this game is quite difficult to get into. First, there seem an infinite number of icons (OK twenty or so) and second, the windows which display them are not exactly huge. Once you get the hang of the game, however, it turns out to be quite intriguing.

The final game is a 3D arcade adventure entitled NEXOR, which stands for Nemesis (sic) EXperimental and Operational Research, whatever that means. The idea of the game is to stop the Nemesis device and its blueprints being captured by the Andromedans (they sound quite friendly don't they?).

Finding yourself in a multi-level complex, surrounded by buildings that look not unlike New York sky-scrapers, you must scamper about as fast as you can. It wouldn't be so bad perhaps if the character you control didn't jump as if he were suffering from terminal diarrhoea. Dilly-dally too long and the Andromedans will beat you to it, seize the parts of the NEXOR device and the valuable documents that accompany them.

Distractions didn't drive me to distraction (fortunately) but all three games were enjoyable in their own way and for the money you can't really complain.

Distractions is not yet available in Australia, but may well be through the pages of next month's magazine. It is likely to cost around $60 for the three games on one disc.

YEAR DISCS

(Not suitable for PCW or PC)

1 Issues 1 to 12 $50.00

Side 1

23MATCH	3DBALL	ADBOOK
AMTHELLO	ANJUMBLE	BEAMHEAD
BINTOHEX	BUZZLINE	COLORDEM
DAYDATE	DIYTYPER	DOWNMINE
FINDUTIL	FLASH	FLOWERS
KABOOM	KEY	KINGDOMS
MAP-CODE	MAP-DRAW	MASTERMD
MATHTABL	MCODE1	MCODE2
MEMRYDMP	MOWER	OWLFABET
PONTOON	RUNNER	SCHEDULE
SHORTWP	SORTDEMO	SOUNDEXP
SUPRESS	TEXTFILL	XHATCH

Side 2

AMSFILE1	AULDLANG	BUDGET
BUGHUNT	CALENDAR	CLOCKPT1
CLOCKPT2	DRAW	FFLOADER
FFORTS	GRAFLOAD	GRAFPLOT
HEADREAD	JJBARCH	JJPICS
MENU10	MENU13	MUSLST1
MUSLST2	NOTEPLAY	PAINTOUT
PERCENTS	QUIZ	RECIPE
RSXGEN	SCNDMP	SNIGHT1
SNIGHT3	SORTS	SPACELDR

2 Issues 13 to 16 $22.50

Side 1 (11k Free)

AMSFILE2 (13)	2K	JJPIC2 (15)	1K
AMSFILE3 (15)	3K	LOADTEST (13)	1K
BANKCAT (14)	3K	LOCODEM (15)	1K
CASSLAB (13)	7K	LPVII (15)	1K
DIGITANT (16)	11K	MLIST45 (14)	6K
DISKCAT (14)	7K	MLIST67 (14)	5K
GETRSX (13)	2K	POKERSX (13)	2K
GUNFIGHT (13)	9K	RANDNOS (13)	4K
HANOI1 (14)	1K	SCA-1920 (16)	3K
HANOI4 (14)	3K	SCA-ANTS (16)	2K
JJPIC1 (15)	1K	YAHTZEE (15)	17K

**Figures in Brackets denote Issue Number*

PLUS FREE PUBLIC DOMAIN SOFTWARE:

SD.COM	4K	SQUEEZE.DOC	24K
SD.DOC	12K	USQ.COM	10K
SQ.COM	14K		

3 Issues 17 to 20 $25.00

Side 1 (9k free)

BLACKJAC (19)	36K	MONSTERP (19)	3K
DOODLE (20)	7K	NEWCASDB (18)	12K
ELIZA (17)	37K	ORANGEAD (18)	15K
ELIZA-IN (17)	1K	PCG (17)	5K
GREMLINS (20)	12K	PROGEDIT (19)	5K
HEDGEHOG (19)	13K	REEDPROG (20)	1K
HILBERT (17)	2K	SPRITE (18)	6K
LONGX (20)	5K		

Side 2 (132k Free)

AMSFILE4 (17)	3K	CATS (17)	3K
APPLE1 (18)	2K	COPYCAT (19)	3K
APPLE2 (18)	1K	CTLCODES (17)	2K

**Figures in Brackets denote Issue Number*

PLUS FREE PUBLIC DOMAIN SOFTWARE:

UNSPOOL.COM	2K	WCOUNT.COM	6K
UNSPOOL.DOC	14K	WCOUNT.DOC	1K

4 Issues 21 to 24 $25.00

Side 1 (73k free)

ADENCODE (24)	3K	PERSONAL (24)	15K
AMCUP128 (23)	7K	PETRCALC (21)	15K
AMCUP464 (23)	7K	PETRLOAD (21)	2K
CAPTURE (23)	8K	SCRNECHO (23)	3K
FIVEROW (21)	6K	SOUNDEX (22)	2K
GIGO (22)	6K	SUPER (22)	2K
LOCO2 (24)	2K	TWISTCUB (21)	8K
MENU464 (24)	2K	WORDSLEU (22)	6K
MENU6128 (24)	2K		

**Figures in Brackets denote Issue Number*

Side 2 (46k Free)

123K FREE PUBLIC DOMAIN SOFTWARE:

CPMPOWER.COM	8K	MAST.CAT	1K
CPMPOWER.DOC	8K	MCAT.COM	7K
LRUN2.COM	2K	MCAT.DOC	7K
LU3.COM	19K	NSWEEP.COM	12K
LU3.DOC	35K	NSWEEP.DOC	7K
LU3.TXT	2K	XCAT.COM	3K
LUDEF3.DOC	8K	XCAT.DOC	4K

5 Issues 25 to 28 $25.00

Side 1 (79k free)

ADVBASE (25)	3K	PYRAMID (25)	1K
AMCUP464 (24/5)	7K	RESRAM1 (28)	1K
AMSCOM (26)	3K	RESRAM2 (28)	1K
BATSHIPS (28)	11K	RESRAM3 (28)	2K
DOUBLEHT (26)	1K	SHUFFLE (26)	2K
GROWWORM (27)	12K	SNOW (25)	1K
HYPNOTIC (25)	1K	SPACE (26)	2K
KALEIDO (26)	2K	SPEEDCHK (28)	4K
KEYEXPND (25)	2K	STAINGLS (25)	1K
LIGHTHO (26)	29K	TEXTDEM1 (27)	1K
PATTERNS (26)	1K	TEXTDEM2 (27)	2K

**Figures in Brackets denote Issue Number*

Side 2 (89k Free)

FREE PUBLIC DOMAIN SOFTWARE:

BITMAP.COM	1K	FIND.COM	1K	MFT.DOC	4K
BITMAP.DOC	2K	FIND.DOC	2K	SWITCH.COM	1K
COMPARE.COM	2K	FINDBAD.COM	2K	SWITCH.DOC	2K
COMPARE.DOC	1K	FINDBAD.DOC	3K	TRANSLAT.COM	3K
CRC.COM	3K	FMAP.COM	2K	TRANSLAT.DOC	1K
CRC.DOC	1K	FMAP.DOC	2K	UKM7.COM	8K
DIRCHK.COM	2K	HELP.COM	3K	UKM7.DOC	10K
DIRCHK.DOC	1K	HELP.DOC	2K	UNERASE.COM	1K
DUMPX.COM	3K	HELP.HLP	7K	UNERASE.DOC	2K

6 Issues 29 to 32 $25.00

Side 1 (36k free)

ADLIST1	(31)	2K	CRIBBAGE	(30/1)	36K	RESTORE.COM	(32) 1K
ADLIST2	(31)	1K	CTRICK	(31)	3K	RESTORE.PRN	(32) 9K
ADPROG1	(32)	2K	CURVE	(30)	1K	SCROLL	(30) 2K
ADPROG2	(32)	1K	CWSU	(31)	1K	SERPENT	(31) 1K
ADPROG3	(32)	1K	DAZZLE	(30)	1K	SIDEWAY2	(31) 4K
ADPROG4	(32)	1K	DOTTY	(29)	1K	SIDEWAYS	(29) 6K
ADPROG5	(32)	1K	DOUBLE	(29)	2K	STROBE	(30) 2K
ADPROG6	(32)	1K	DRUMKIT	(30)	4K	TDUMPBAS	(31) 1K
BLITTER	(30)	2K	GEN1.TXT	(29)	1K	TDUMPBIN.BIN	(31) 1K
BORDER	(29)	1K	GEN2.TXT	(29)	4K	TILES	(29) 3K
BOX	(29)	1K	HANOI-A	(32)	4K	TRACKS	(30) 2K
CHEAPWP	(31)	3K	HANOI-B	(32)	5K	TRAIN	(30) 3K
CIRC1	(32)	1K	MODE210	(30)	1K	TRAINDEM	(30) 4K
CIRC2	(32)	1K	POKE1	(29)	1K	TRIANGL	(31) 1K
CIRC3	(32)	1K	POKE2	(29)	1K	TRIANGLE	(30) 2K
CIRC4	(32)	1K	POKE3	(29)	1K	**Figures in brackets denote Issue Number*	
CODECONV	(30)	2K	POSTFIX	(30)	3K		

Side 2 (71k free)

98K OF FREE PUBLIC DOMAIN SOFTWARE

CAT.COM	2K	LISTT.COM	2K	QK.COM	4K
CAT.DOC	1K	LISTT.DOC	4K	QK.DOC	9K
DUU.COM	7K	LSWEEP.COM	16K	QK.TXT	1K
DUU.DOC	13K	LSWEEP.DOC	4K	WASH.COM	3K
FINREP.COM	3K	MAKE.COM	2K	WASH.DOC	16K
FINREP.DOC	9K	MAKE.DOC	2K		

What you get in the free Public Domain CP/M Software:

CAT.COM	-	**sorted Directory lister (like CAT)**
DUU.COM	-	**comprehensive disc utility from Ward Christensen**
FINREP.COM	-	**search and replace program**
LISTT.COM	-	**prints any requested file**
LSWEEP.COM	-	**library utility**
MAKE.COM	-	**extended version of STAT.COM**
QK.COM	-	**key definer**
WASH.COM	-	**the ultimate CP/M disc directory**

The prices of all Year Discs include postage.
Send your order to: The Amstrad User, 1/245 Springvale Road Glen Waverley, Victoria 3150 or phone (03) 233 9661
Mastercard, Bankcard or Visa accepted.

Account for yourself

Part two of Ian Berry's teach-yourself-accounts trilogy discovers the beauty of paying on credit.

Last month's article revealed that basic book-keeping is just keeping a diary, and it left off asking, "Will we ever get round to using the computer?" This month you can finally blow the dust off your computer, plug in and start to streamline your accounts with the help of a spreadsheet.

Last month's cashbook accounting system is not a very good information base since all the payments are jumbled up together, as are the receipts and you need to be able to answer questions like "How much did I spend on goods for resale in this week?" or "How much did it cost to run the van in that week?" To do this you would need to sit down and analyse the cash book. If it were already analysed these answers would be readily available. The payments side of the cash book would have to look like the example shown.

date	Narrative	Total	Purch	Van	Wages	Rent
Mon	Goods	60	60			
	Rent	20				20
Tues	Petrol	15		15		
	Bags	7	7			
	Goods	110	110			
Wed	Repairs	25		25		
Thu	Wages	55			55	
		292	267	40	55	20

The first three columns are exactly the same as the old 'diary' system (date, narrative, payment), but then the amount is repeated in the appropriate 'analysis' column so that by totalling any column you find out how much you have spent in that area.

This is where the computer starts to be helpful.

Now you are into analysis you need an analysis program. These are called 'spreadsheets', because they are like a large sheet of paper spread out on a desk. There are umpteen spreadsheet programs available, and it would be unfair to single out any particular one because most of them can do everything you need.

You need to design your layout with columns for all the headings you will use. However, it would be impossible to have a column for every heading you would ever need, since there will always be unexpected categories cropping up. It is useful to reserve the last two columns of a page under the heading 'others'. One of these holds a subsidiary narrative, where you write a description of the transaction, and the other holds the amount. Then as you go along you can enter your payments, keep the accumulating spreadsheet in memory, and when you need to know how much you spent on the van one week, you use the program to add up the relevant column for you.

You can double-check your figures by comparing the 'total' column with the sum of all individual analysis columns. If they match, you can be sure that there are no errors, and you can just read the values you want off the columns. Rather than keep the whole year on one sheet, you would total up and print reports out weekly or monthly, keep the printouts for future reference, and start the next period's sheet with the totals from the last one. This avoids your spreadsheet becoming unmanageably large.

As an example, look at the spreadsheet shown for a mythical fruit and veg business. This shows the analysed receipts on the left and the analysed payments on the right. For detailed transactions you may be better off keeping receipt and payment details in separate files, otherwise things can get confusing.

The takings from the shop and the stall have been shown separately, but cannot be totalled separately without some difficulty. In this case it might be advisable to increase the number of columns on the receipts side so that the four columns shown here are duplicated, with one set used for the shop and the other for the stall.

Having got a total for the week, you can now think about how you are going to use it. You could either just use it as the first line of the next week's sheet, or you could start another spreadsheet file of weekly totals, or you could carry it forward to the end of the month and then have a sheet of monthly totals. In short, do whatever you think will give you the most useful information.

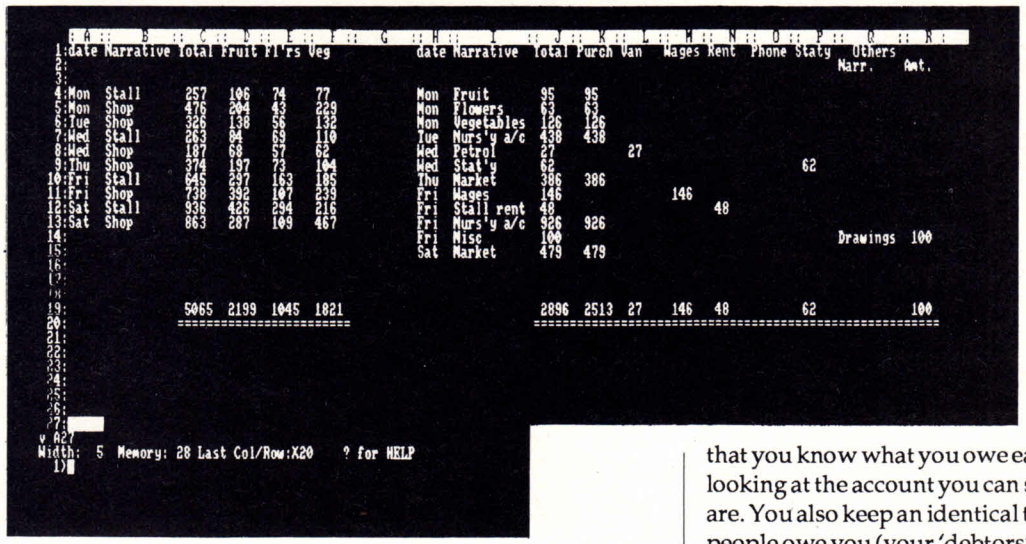
'On account' trading

When we set out on the diary system which grew into a spreadsheet, I explained that it would work for any cash business. What if you buy and sell goods on credit? As far as the cash side of the business goes these systems will still work, but you need something more to keep tabs on what you are owed and what you owe other people.

Think back to the cash book diary from last month. What you need to do now is to adjust your "Duplicate Receipt Book" and "File of Paid Bills" boxes to allow for unpaid as well as paid items. This means that you need an extra file for each one to hold the unpaid items, so your system will look like the one five-box diagram shown.

Now that you have sorted out the configuration, what do you do differently? Apart from looking after the two extra files - nothing.

Is this still a 'perfectly good' record of your business? Well, not quite, because you don't know how much you have spent in the period, but only what you have paid for. The difference



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	date	Narrative	Total	Fruit	Fl'rs	Veg		date	Narrative	Total	Purch	Van	Wages	Rent	Phone	Stat'y	Others Narr.	Amt.
2																		
3																		
4	Mon	Stall	257	106	74	77		Mon	Fruit	95	95							
5	Mon	Shop	476	204	43	229		Mon	Flowers	63	63							
6	Tue	Shop	326	138	56	132		Mon	Vegetables	126	126							
7	Wed	Stall	263	94	62	110		Tue	Murs'y a/c	438	438							
8	Wed	Shop	187	68	57	62		Wed	Petrol	27		27						
9	Thu	Shop	374	197	73	104		Wed	Stat'y	62						62		
10	Fri	Stall	645	297	163	185		Thu	Market	386	386							
11	Fri	Shop	738	392	107	239		Fri	Wages	146			146					
12	Sat	Stall	936	426	294	216		Fri	Stall rent	48				48				
13	Sat	Shop	863	287	109	467		Fri	Murs'y a/c	926	926							
14								Fri	Misc	100							Drawings	100
15								Sat	Market	479	479							
16																		
17																		
18																		
19			5065	2199	1045	1821				2896	2513	27	146	48		62		100
20																		
21																		
22																		
23																		
24																		
25																		
26																		
27																		

v A27
Width: 5 Memory: 28 Last Col/Row:X20 ? for HELP
1)

the 'unpleasantness' of full accounting.

Creditable performance

The difference between the cash accounting system that we have been looking at and what should perhaps be called 'expenditure' accounting is that we now need to build into the system the effect of future commitments on day-to-day business results.

Since an obligation to pay for something arises as soon as it has been accepted you must start your accounting at the point where the obligation arises. When we were looking at cash accounting the obligation was settled immediately so you did not need to separate the obligation from its settlement.

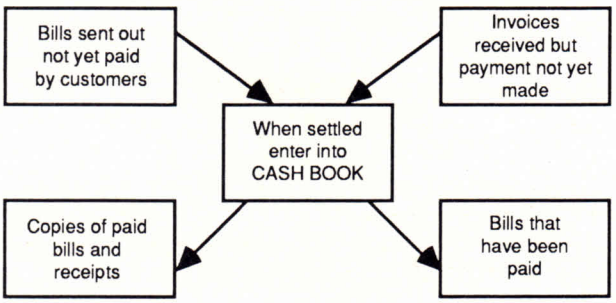
This brings you to the point where you have to start talking about 'diaries' of goods received and sold as well as the cash diary you have already mastered. You also need a more sophisticated way of keeping tabs on what you owe and are owed. All this means that you must shift your focus of attention from the journal (book of original entry remember?) to the 'ledger' which is 'the big book of all the accounts'.

A ledger is simply a book with lots of individual accounts in it, and an account is a record of all the transactions to do with a specific person or 'thing'. When you started analysing your cash book you actually started a sort of ledger because you used each separate column to collect all items for a specific category. The drive towards ledger accounting now, though, is the need to keep tabs on what is owed.

Instead of having a file of unpaid invoices which you have to go through every time you need to find out what you owe people, you can have one book, with the details for each person (the 'creditors') kept separately. This book can take many forms, but in essence it shows, for each creditor, a list of the items you have bought and a list of payments made. When you make a purchase or payment, it is recorded so that you know what you owe each creditor at any time, and by looking at the account you can see how old the various items are. You also keep an identical type of record for what other people owe you (your 'debtors').

You need to understand more fully the nature of all transactions you are merrily recording. Every transaction, whether a purchase, a sale, or a service involves two aspects - goods or services move in one direction and a debt is incurred (and subsequently settled, you hope) in the other. You need to record both of these aspects. When you buy something, you record what you have spent your money on as well as to whom you owe the money. The 'Bought' and 'Sales' Ledgers record the personal side of transactions and are known collectively as the Personal Ledgers, and the impersonal side is recorded in the Nominal or Impersonal Ledger.

The cliff-hanger on which this month's instalment closes is "How do you cope with all these new books?" The answer, as you will see next month, is "Quite easily" - especially with the help of your trusty computer."



Nothing matters

Although book-keeping of some sort has been practised since 3500BC to our certain knowledge, it was pretty rudimentary for most of the time. Initially the problem was in how to record things, and then it was how to calculate, but the invention of 'zero' (in accounting this was at least as important as the invention of the wheel!) set in train two distinct developments.

First, the idea of a number for 'nothing' opened the concept of abstract numbering with ramifications well outside the scope of this article, and secondly, and more relevant to accountants, it encouraged the adoption of Arabic numbers (as opposed to Roman numerals) and positioned notation which enable them to do calculations that had been impossible before.

By the 13th century this number system had become fairly common in Europe and by the late 1400s book-keeping had developed to the point where an Italian monk, Fra Luca Paccioli, was able to write a treatise which included a detailed description of what we would clearly recognise today as 'double-entry book-keeping'. In fact you could almost say that there has been no change at all apart from certain fairly minor developments to deal with changes in the commercial world.

Analogue and digital Clock

A utility to run on your Amstrad when other work needs doing . . . .

from Ian Abbott

If you're anything like me, even after all the serious work that can be accomplished on the computer has been finished, there's still more work, sadly enough, that is not in the slightest way computer-orientated. In the past your beloved possession sat there displaying "READY" until you remembered to pull it's plug. Now you can turn your Amstrad into an attractive combined digital/analogue display.

Features

The program is a "no-frills" application so that people can add their own embellishments as they wish. After input of current time, the screen clears, draws a clock face and puts in the hands at the input time. A digital display is also given.

The digital readout is the hours:minutes:seconds type and has a leading zeros format so that times such as 4:05:34, for example, do not become 4: 5:34. The appropriate am or pm suffix is appended.

The clock face is round (corrections have been incorporated allowing for the Amstrad's vertical distortion) with dashes at every 5 minute mark and dots at each minute mark. There are three hands; the seconds hand is updated each second, the other two updating each minute.

Recent amendments to the program include a video reverse function and a dimmer (which can be used to manipulate colour on a colour monitor).

Operation

Three responses are required. To "hours?" type in the time according to a 24 hour clock (this is how am or pm is selected). Then to "Minute?" give the number of minutes after the hour and to "Seconds?" the number of seconds after the minute. It takes some time to set up the screen so allow about four seconds for total accuracy. Default values are 24 for hours and 0 for minutes and seconds (midnight).

The clock looks best in a darkened room with the brightness and contrast controls turned to a minimum (on a green screen). The background and display colours can be reversed by pressing 'R'. Use the up/down cursor keys to dim or brighten the clock face.

Problems and Program details

Essentially the program loops inside nested WHILE loops. The seconds-counter is incremented by an AFTER 50 interrupt. This is why the loops appear as never-ending which would be the case if there wasn't an interrupt. The minutes counter advances upon the seconds counter reaching 60 (which is reset to 0). In the next time-bracket the same happens for the hours-counter.

Screen preparation is performed by lines 510-610. The hands are drawn in by lines 620-660. Note that XOR ink mode is used (line 630) so that the seconds hand doesn't wipe out the hours and minutes hands in the first revolution. Animation is achieved by drawing over the top of the previous hand position. The first pass draws; the second pass erases.

This method comes with some problems however. Initially the program enters the last subroutine at line 640 (as there is no previous hand to erase). So if the time is 4:33:33 initially, the 33 position is drawn 3 times: once each for seconds and minutes and once more when the second hand advances. This leaves the minute hand "dead". The same problem (but less obvious) occurs when hands pass over each other in the course of the program, but the situation is righted as soon as the faster hand moves ahead.

One other problem was a language fault. BASIC does not support a leading zeros format directly. For this reason bulky IF statements were required to check the length of mins and secs and channel the print out through the required format (initially lines 240-260).

I have kept GOTO statements to a minimum to keep the program reasonably structured, and have indented looped segments (a practice more people should make use of) to improve readability).

Variables

j	-	hand designator (secs, mins, hrs)
ln (1 to 3)	-	hand length
inc (1 to 3)	-	degree increment at each movement
ps (1 to 3)	-	position of hand
last	-	previous position of hand
hrs	-	hours
mins	-	minutes
secs	-	am/pm suffix
ct	-	counter to draw clock face dashes
lit	-	counter to draw dots
innerx	-	inner x co-ord for dashes and dots
outerx	-	outer x co-ord for dashes
innery	-	inner y co-ord for dashes and dots
outery	-	outer y co-ord for dashes

Note: This program was written for a 6128 and will require some small modifications to run on a 464. For example lines 70 and 80 use GRAPHICS - delete this word. Also look at the DRAW command in line 630.

```
10 REM ********************************
20 REM
30 REM ***  ANDIGCLK.BAS - IAN R ABBOTT
40 REM
50 REM ********************************
60 ON BREAK GOSUB 470
70 MODE 1:GRAPHICS PAPER 0:col=26:backcol=0:INK 1,col:INK
0,backcol:BORDER          backcol
80 GRAPHICS PEN 1
90 INPUT "Hour (0 to 24)";hrs
100 IF hrs=0 THEN hrs=24 ELSE IF hrs<1 OR hrs>24 THEN PRIN
T CHR$(7):GOTO 90
110 INPUT "Minute";mins
120 IF mins<0 OR mins>59 THEN PRINT CHR$(7):GOTO 110
130 INPUT "Secs";secs
140 IF secs<0 OR secs>59 THEN PRINT CHR$(7):GOTO 130
150 GOSUB 510
160 j=1:ln(j)=125:ps(j)=secs:inc(j)=6
170 j=2:ln(j)=105 :ps(j)=mins:inc(j)=6
180 j=3:ln(j)=60 :ps(j)=(hrs+mins/60)*60:inc(j)=0.5
190 FOR j=1 TO 3
200  GOSUB 640
210 NEXT j
220 LOCATE 5,5:IF (hrs+mins/60+secs/3600)>12 AND NOT hrs=1
2 THEN suf$="pm" ELSE      suf$="am"
230 IF NOT hrs <13 THEN hrs=hrs-12
240 IF mins<10 AND secs<10 THEN PRINT USING "##:0#:0#";hrs
,mins,secs:GOTO 280
250 IF secs>9 AND mins<10 THEN PRINT USING "##:0#:##";hrs,
mins,secs:GOTO 280
260 IF secs<10 AND mins>9 THEN PRINT USING "##:##:0#";hrs,
mins,secs:GOTO 280
270 PRINT USING "##:##:##";hrs,mins,secs
280 EVERY 50 GOSUB 480
290 WHILE hrs<12
300   WHILE mins<60
310     WHILE secs<60
320        LOCATE 11,5:IF secs<10 THEN PRINT USING "0# ";
secs;:PRINT suf$ ELSE              PRINT USING "## ";secs;
:PRINT suf$
325        IF NOT INKEY(0)=-1 THEN col=col+1 ELSE IF NOT I
NKEY(2)=-1 THEN col=              col-1
326        IF col>26 THEN col=26 ELSE IF col<1 THEN col=1
327        INK 1,col:INK 0,backcol:BORDER backcol
328      IF NOT INKEY(50)=-1 THEN AFTER 20,2 GOSUB 1000
330     WEND
340     SOUND 1,100,5
350     secs=0:mins=mins+1
360     LOCATE 8,5:IF mins>9 THEN PRINT USING "##:0#";mins
,secs ELSE PRINT              USING "0#:0#";mins,secs
370     FOR j=2 TO 3
380      ps(j)=ps(j)+1
390      GOSUB 620
400     NEXT j
410   WEND
420   mins=0:hrs=hrs+1
430   LOCATE 5,5:PRINT USING "##:0#:0#";hrs,mins,secs
440 WEND
450 hrs=0:IF suf$="am" THEN suf$="pm" ELSE suf$="am"
460 GOTO 290
470 INK 1,24:INK 0,0:BORDER 0:PEN 1:PAPER 0:MODE 2:END
480 secs=secs+1
490 j=1:ps(j)=ps(j)+1:GOSUB 620
500 RETURN
510 MODE 1:DEG:ORIGIN 400,170
520 FOR ct=0 TO 360 STEP 30
530   innerx=155*COS(ct):innery=150*SIN(ct)
540   outerx=170*COS(ct):outery=165*SIN(ct)
550   PLOT innerx,innery:DRAW outerx,outery
560   FOR lit=ct+6 TO ct+24 STEP 6
570     innerx=155*COS(lit):innery=150*SIN(lit)
580     PLOT innerx,innery
590   NEXT lit
600 NEXT ct
610 RETURN
620 last=ps(j)-1:x=ln(j)*COS(90-last*inc(j)):y=ln(j)*SIN(9
0-last*inc(j))
630 MOVE 0,0:DRAW x,y,,1
640 x=ln(j)*COS(90-ps(j)*inc(j)):y=ln(j)*SIN(90-ps(j)*inc(
j))
650 MOVE 0,0:DRAW x,y
660 RETURN
1000 temp=col:col=backcol:backcol=temp
1010 RETURN
```

Spaced-out Trio

If you're planning to save the universe read these three reviews from Nick van Kempen first

THE FINAL MATRIX - Gremlin

Three-D with a difference is the Final matrix's new gimmick. No more juggling the joystick to determine whether up is right, or left is down.

Final Matrix gives a new three-d looking down,rather than in, in the tradition of Batman and Head over Heels. But this too has its small problems because Nimrod, the salt and pepper shaker-type character you control can often disappear behind walls or bump into an unseen foe hidden by the shadows.

Nimrod's task is to rescue the Bioptons captured by the hostile Cratons. The Bioptons are held captive in the Craton's Matrix network, a three-dimensional maze guarded by booby traps, guards and other traps.

There are a few innovations with this tape-based game, such as credits and instructions while the game loads. A timer counts down the seconds while a panel shows information about the traps you may encounter and some advice, such as avoid the guards, and the like.

With each instruction is an illustration of the bad or in some cases the good guy.

Manoeuvering Nimrod is a little tricky and without propulsion he relies on certain squares to give him a boost to the next level. A thrust pack can be picked up as can weapons and a TV monitor which shows an overview of the matrix level Nimrod is on.

In this mode the game resembles Gauntlet and I suppose that is what it is, a Gauntlet-Batman-Knightlore clone with a difference.

As with most mazes, gameplay is a challenge of wits and reflexes with puzzles and the ever frustrating need to travel through a few screens to reach another level and then back to where you started.

After Nimrod has located and rescued a Biopton hostage he must make his way back to the take-off site and back to his ship. A pointer display shows the general direction of the take-off site throughout the game. Once back on his ship it's back to another matrix to rescue another hostage or two.

Gameplay is complex and interesting, although the new view of the maze is an interesting challenge.

Although The Final Matrix gives a new slant on an old theme it doesn't, unless you're a dedicated maze fan, have the fast and furious action of Gauntlet to make it addictive. A short play of the game means you only have a small piece to judge, but it does promise to be an interesting game with many levels of fun or frustration.

ULTIMA RATIO - Firebird (Silver Range - cheap!)

It seems that the universe will always need the help of the computer game player. Take Ultima Ratio, from Firebird. The instructions start 'In the 33rd year of entrenched warfare, a new forbidding threat looms out of the blackness of deep space.'

Forget all the 'save the universe' hype and play the game.

Ultima Ratio is a pretty basic space shoot'em up but the graphics and game play make it great value from the budget racks. A while ago another budget game, Cerberus, gave us splendid graphics with pretty ordinary game play. For those who enjoyed Cerberus, Ultima Ratio goes much better with a playable game and good use of the Amstrad colour capabilities.

Ultima Ratio is a new nine-platform battle station with plenty of nasties and you, armed only with your wits and your trusty ship, must clear each platform before moving on.

Scattered around each platform are the assorted bad guys but there are some supplies for you such as shields and fuel. Some areas have the nasty habit of reversing your directions.

You have a limited time for each platform and five lives before the GAME OVER message lights up. Play is by joystick or keyboard with definable keys.

Although cheap, Ultima Ratio is hardly a nasty.

REALM - Firebird (Silver Range - cheap!)

It's a shame that Realm will share shelf space with a game like Ultima Ratio (see above). While Ultima Ratio appears to make the best use of the Amstrad, Realm takes us back almost to the days of Roland and his bits and pieces. Sorry to all Roland fans, but that was a while ago and we should be able to expect good quality, even at the budget end of the market.

Realm uses big chunky sprites and is essentially a maze game. You control a rocket (which doesn't look like a rocket) and travel around the Inner Co-Ordination System (which I guess looks like one). You have to locate and assemble each of the nine solar planets, then position it in its correct position around the sun. Although this sounds grand, you could be forgiven for thinking you were playing a cheap kiddies maze game with pretty colours.

If Realm is given to you as a birthday present you might play it.

If not, think about something else.

Adventurer's Attic

In which the listless Philip Riley, having now found it, presents the long awaited 'sentence input' routine

Yes folks here it is at last, sentence input, the proggy you have been waiting for. But before we get to the listing lets look at a few of the problems that are faced when trying to decipher sentences that are inputted into the game.

You may not have really thought about it too much but the English language is extremely complex. You can say the same thing in so many ways and of course you really don't won't to have to check all of these whenever a sentence is typed into the game, so we have to cut out words that are not really important. These words are called null words. What sort of words are they? Well one classic example of a null word is "the", I always cut this out of any sentence that is inputted into one of my games.

So "KILL THE DRAGON" would become "KILL DRAGON", but it goes much further than this. Most of the words that describe things can be classed as null words eg. small, large, green, square etc.

But of course there may be a time when you want to use some of these words, for instance, if you have two bottles in a game, one red and one green. You will want to know which bottle the player is referring to, so in this case you would include the two colours red and green in words that are to be used and would not class them as null words.

In fact it all boils down to you in the end. Of course the less words that you reject as null words the more variation of input will be possible. Another word that needs consideration is "with", not as a null word on its own but as a null word with the rest of the words after it. You could accept "FILL BOTTLE WITH WATER", but if you are next to a stream then obviously you are going to fill the bottle with water and so you could cut the sentence down to "FILL BOTTLE". Again this is entirely up to you and the situation in your game, if you have more than one type of liquid at a specific location then you may need to include the 'with' as a normal word and not as a null word.

So what about some of the uses of sentence input. Well, of course, it allows for a far greater scope in your games and makes the games far better visually, but it can also be of use in another way. With the normal two word input you can only input one command at a time. With sentence input you are able to input two or more commands in one go. How is this better? Well you can now include a feature into your games which makes it possible to pass a particular problem only if you input two commands together. In this month's proggy I have included an example of this.

Part of the program is a test to show you just how it works. It is a small problem that you must overcome. You must get the sword, kill the dragon and escape to the North. The problem is, if you kill the dragon he falls onto you and squashes you, so you must "KILL DRAGON AND RUN NORTH". In this way you escape unharmed.

And so on with the proggy.

```
10 CLS:DIM a$(20),c$(10)
20 RESTORE:FOR t=1 TO 10:c$(t)="":NEXT:FOR t=1 TO 20:a$(t)
="":NEXT:a=1:INPUT"Input command ",a$:FOR t=1 TO LEN(a$):I
F MID$(a$,t,1)=" "THEN t=t+1:a=a+1
30 a$(a)=a$(a)+MID$(a$,t,1):NEXT
40 FOR t=1 TO 20:IF a$(t)="and" OR a$(t)="then" THEN a$(t)
="&"
50 READ b$:IF b$="end"THEN RESTORE:GOTO 80
60 IF a$(t)=b$ THEN a$(t)="":RESTORE:GOTO 80
70 GOTO 50
80 NEXT:t=1:c=1:q=0
90 READ b$:IF b$="end"THEN 100 ELSE 90
100 IF t=21 THEN 160
110 IF t<20 THEN IF a$(t)="&" AND a$(t+1)="&" THEN t=t+1:G
OTO 100
120 IF a$(t)="&" THEN c=c+1:t=t+1:GOTO 100
130 IF a$(t)=""THEN t=t+1:GOTO 100
140 q=q+1:READ b$:IF b$="end"THEN PRINT"I don't understand
 this word.....";a$(t):INPUT"Please type in a similar wor
d";a$(t):RESTORE 1010:q=0:GOTO 100
150 IF a$(t)=b$ THEN c$(c)=c$(c)+CHR$(q+31):q=0:t=t+1:REST
ORE 1010:GOTO 100 ELSE 140
160 FOR a=1 TO c
170 IF c$(a)=" &"THEN PRINT"You take the sword.":sw=1
180 IF c$(a)="'(" AND sw<>1 THEN PRINT"You have nothing to
 kill the dragon with"
190 IF sw=1 AND c$(a)="'(" AND c$(a+1)<>".)" THEN PRINT"Yo
u kill the dragon but it falls on top of you, killing you
instantly.":END
200 IF sw=1 AND c$(a)="'(" AND c$(a+1)=".)" THEN PRINT"You
 kill the dragon and run to the NorthWell done you have be
aten the dragon.":END
210 NEXT:GOTO 20
1000 DATA the,to,large,small,red,green,end
1010 DATA get,take,leave,drop,axe,rope,sword,kill,dragon,n
orth,south,east,west,go,run,end
```

Breakdown

Line 10 is the setup for the arrays (rather obvious really).

Line 20 restores the data (please note that if you have other data within your program you will have to put the data for the game after it and then restore <line number>). After the restore we clear the array a$(t) and ask for an input. We then start a loop that checks along your input for spaces.

Line 30 adds each character found into a$(a). When a space is found in line 20 'a' is incremented by one and so we start building up the second word.

Line 40 starts the main loop that checks for null words. This loop also checks for the words "and" and "then" and replaces them with the character "&".

Line 50 reads the first of the null words into b$. If this word is "end" then it knows that it has come to the end of the data containing null words and so this word has past the first test; the data is restored (again this would be restore <line number> if you have other data before this).

Line 60 checks the word against the null words. If the word is a null word then the data is restored (same rule applies to this restore as before) and the word is zapped out of existence.

Line 70 loops the program back to read the next null word from data.

Line 80 is the next command for the first loop, but also sets up the variables for the second loop.

Line 90 reads whatever remaining null words remain to be read so that we can move onto the next lot of data.

Line 100 checks to see if we have checked all of the words. In this case we can have 20 words so when t=21 we know we have finished.

Line 110 makes sure that we do not have two "&" characters in a row. This would be the result of someone typing in "AND THEN".

Line 120 checks for just one "&" character. When it finds one it adds one onto the variable c (the reason for this will be explained later).

Line 130 checks to see if the word in a$(t) actually exists (it could have been a null word and got zapped out earlier on in the proggy).

Line 140 adds one onto the variable q. q will tell us how many words we have read so far. Each time we restore this section of data q is set to zero. We now read the first word into b$ again, if it equals "end" then we know we have reached the end of the data and a message is displayed saying that the word that is being checked is not understood.

Line 150 finally checks the word that you have inputted against the word that has been read in from data. If a match is found then we start building up c$(c) with ASCII characters. If a loop is found we restore the data and go back to line 100 to start the whole process again with the next word in the input. If no match is found then we go back to line 140 and read the next piece of data.

Line 160 starts a for-next loop. The size of the loop depends on how many "and" and "then" statements you have in your input. The rest of the program up to line 200 is a test and does not form part of the program (in other words don't put this bit into your proggy).

Lines 170 to 200 will be explained later.

Line 210 is the next line for the loop that starts on line 160. After the loop has been completed the program jumps back to line 20 and the whole thing is repeated with your next input.

Line 1000 contains the data for the null words in this particular test program.

Line 1010 contains the data for words understood by this test program.

Now, how does it work? In much the same way as my encode/decode program in the Dec86/Jan87 editions. We build up a coded string as we find the words that have been inputted. We then check these strings in line 170 to 200 and if they are the same then we carry out the command that has been given.

The commands are split up and are not looked at collectively (this is where the variable c comes in). After we find each "&" character 'c' is incremented by one so moving onto a new string to be coded. As a test try changing line 160 to 160 PRINT c$(1):END

Now run the program. If you have typed in some words that the game understands then you should now be presented with a small row of ASCII characters on the screen. This is your input in coded terms. If you typed in an "and" somewhere in your input, try typing in PRINT c$(2). You will now be presented with another row of ASCII characters. Each command is then taken care of one at a time in the for-next loop in line 160 to 210. You can of course deal with more than one command at a time as demonstrated in line 200 where we kill the dragon and run north.

Now I know what you are thinking, this proggy uses a lot of strings, and you are quite right it does, and I know that you are now thinking that this is not good, and again you are right. So don't worry I will be working on a numeric system to do the same (in fact, it wouldn't hurt some of you lot to do this. Why not give it a go, it's not as hard as you might think). Also on the drwaing board is a numeric system for encode/decode as well as a machine code version, although these are rather a long way off at the moment.

Well that is enough of programming for a while, I think that next month we will take a look at some odds and ends. I also have a few more tips for saving memory.

We have a couple of problems from Boaz Kogon to look at (sorry about the delay Boaz but somehow your letter got mixed up in the Editor's office and I have only recently received it). If letters are addressed directly to Adventurer's Attic, then they get put in the right pigeon hole straight away rather than going through the normal cycle.

This month I will end the column with some words from one of our Queensland readers. In his own words he likes to bug me with things, the latest being a chain letter. Well David Watt, your words are about to become famous. This is how David always ends his letters. MAY THE FORCES OF EVIL BECOME CONFUSED ON THE WAY TO YOUR HOUSE.

I think they have because they were not there last night when I got home!

Hint Sheet

SORCERER

Hint sheet from Michael Shephard.

The aim of this story is to rescue your mentor and friend Belboz who has been captured by a demon called Jeaar. To help you, you are given some spells and have to collect additional spells to further your quest. To obtain help, look up the keyboard or phrase and then read the hint associated with that keyword.

Sorcerer Hints:

Morgia Plant - No-use.

Parrot - What it says can be of use, but not really useful.

Desk - The key to journal is hidden in wall hanging.

Gaspar - Resurrects you when you die (resurrects you at the last place you cast the Gaspar spell.

Food and Water - Vial in store room "Berzia Potion". Drinking it solves your problem.

Matchbook - Read it! Put it in Mail Receptacle before mailman arrives.

Dream - Nothing you have to do in dream, it can't last more than 7 moves.

Hellhound - Read infotator from game package, it won't pursue you away from its territory - so run!

Trunk - Note code word in Belboz's journal and find out colour combination from your infotator. Then press the buttons in the correct order.

Aimfiz - Cast Aimfiz on Belboz or Jeaar (ie. "Aimfiz Jeaar") to get to near where they are.

Minefield - You can't cross the minefield.

Moat - You can't go down into the moat and live!

Indigo Vial - No use in this game (but drink it anyhow).

River - Pulver river - NW and SE directions useless. Note: river stays pulvered for a few moves only.

Amber Vial - no use in this adventure (take the Guano though).

Fort - Lower and examine the flag. The flag itself is of no use.

Cannon - Read about Yipples in your Infotator. (And remember the bat Guano at the hidden cave).

Chasm - Izyuk across.

Zorkmid Tree - You can get only 1 coin. Tree then disappears.

Hall of Carvings - try Malyon Dragon. But! You need to "turbo-charge" the Malyon spell. So cast the Yonk spell on the Malyon spell.

Toll Gate - Give Gnome the coin then search him to get it back.

Floor Waxer - Useless.

Statue - No help at all (Malyon the statue though).

Maze - Cast Fweep spell on yourself. This way you can negotiate the maze. It has 27 rooms, stacked like cubes in a Rubic's Cube.

Store Hut - No use but the fireplace comes in handy when you've solved the glass maze.

Hollows - Drop the scroll down the structure (it falls into the fireplace of the store hut).

Dorn Beast - Must return through the glass maze. Fweep yourself and fly back. When you get to a room with no floor, the Dorn Beast falls to its death. (Carcass is of no use.)

Amusement Park - Pay admission price. If you passed the Toll gate and have no coin read the toll gate hint above.

Flume Ride - No use.

Rollercoaster Ride - No significance.

Haunted House - Zip.

The Game with the Rabbit and the Ball - Drink the potion found in the flag to increase your dexterity.

Casino - After a number of goes, the machine pays out 1 coin, which is useful if you have not entered the toll gate as yet.

Coal MIne - Drink Vilstu potion so you don't have to breathe.

Older Self - Give him your spell book.

Coal Mine Maze - You have limited moves, as the Vilstu potion wears off, so in the moves you have to tie the rope to the timber and put the timber across the opening, then drop the rope down the opening. (Climb down.)

Orange Flash - To get to the room where the orange light is you have to climb down the rope carrying nothing.

Younger Self - Tell him the combination and if you (as the younger self) gave you the spell book, it will be given to you.

Lagoon - Enter Lagoon and dive down. (Drop spell book before doing this, as it is destroyed by water.) Meef the spense weeds.

Vines - Meef vines.

Grues - From box in lagoon; you either wear the grue suit or rub the repellant on you.

Machinery - No use or importance.

Eternal Pain (The Room of) - You can't ever get out of this room.

Eternal Death (The Room of) - See hint to room of Eternal Pain.

Dreams - The dreams in this adventure have no meaning.

Belboz - Vardik Belboz to exorcise the Demon Jeaar. But you first must protect your own mind with the Swanzo spell.

This first Hint Sheet earns Michael $25. It gave our spell-checker a battering and we have our fingers crossed that we have understood Michael's handwriting - a typed copy would have been better!

You too could cash if you have solved an adventure. Send them to The Editor, The Amstrad User, 1/245 Springvale Rd, Glen Waverley, 3150

QUESTIONS & ANSWERS

QUESTIONS

First of all a couple of points concerning letters. Due to the deadline of the mag, letters generally appear eight or ten weeks after I receive them. The minimum deadline is six weeks before the next magazine (ie. I need them about the middle of each month). Secondly I would ask you to address any letters for this column to Adventurers Attic, this helps the Editor sort them out (he sometimes gets a little confused).

Finally, true to hos word, the Editor has published the first Hint Sheet mentioned a few months ago. If you've got one, send it in, but don't give the whole game away otherwise it won't get printed.

Now onto the questions.

Firstly we have four questions from Albert Flawes concerning Heroes of Khan. They are:

How do you get past the pheonix?

How do you get the silver crucifix from the giant bat and how can you get the jade flower, and finally how do you get rid of the spider or the vampire in the treasury.

Jamie Coppack is another hitch-hiker who is having trouble hitch-hiking around the galaxy. He would like to know if there is any way of leaving the VI'HURG and G'GUGVENT spaceship without removing the particle and when you eat the fruit of foreknowledge how do you find Marvin to give him what he asks for.

ANSWERS

Albert Flawes has given us some answers for Bastow Manor. To get into the Manor you must take the ladder to the second floor window (located East of the front door) and then drop it. This will enable you to get back out of the house again. You must then return to the shed and move the case to reveal a trapdoor. Look into the case and get the apple, a key is inside it. Light the torch before you go down. You will enter a dim room that has a door with a beam across it, remove the beam and go to the door and you should now be in the hallway. You can also move South from the dim room where you will be confronted by a snake pit. Lay the branch across the pit to proceed.

We have another one for The Lord of the Rings from Karla Slack this month. She has two ways to get past the barrow-wright. You can either put on the ring, go East and wait until all of your friends have arrived (you should only have to wait for two or three times) and give them all swords or if you have an elfstone throw it, but make sure once again that you all have swords.

TIPS OF THE MONTH

Albert Flawes could not help us with Lord of the Rings but he knows of a barrow-wright in Heroes of Khan. To get past this barrow-wright you attack him with the bible.

This tip is also from Albert Flawes and concerns Bastow Manor. If you enter "1" and not "LOOK" you get clues for each room.

Hot Tips

Hard cats, talking bugs and other distorted characters revealed by you, the Amstrad experts.

Reset no more

I have worked out several pokes that I hope will be of use to some readers:

`POKE &BDEE,&C9`	turns reset function off
`POKE &BDEE,&C3`	turns reset function on
`POKE &B4E7,0`	turns shiftlock off
`POKE &B4E7,&FF`	turns shiftlock on
`POKE &B4E8,0`	turns capslock off
`POKE &B4E8,&FF`	turns capslock on

Note that the shiftlock and capslock pokes are for 464 users only.

Mark Swinson

A bugs story

Did you know that Mini Office II (disc version) contains a secret story within the word processor? Press the keys BUGS simultaneously while loading the word processor from the main menu. I include a poke to send the story to your screen or printer.

But that is not all. There is another hidden story on the disc. To access this hidden story you must alter a few lines of the poke listing:

```
1 'Message Finder
2 'The Amstrad User Oct 87
10 INPUT"Screen or Printer (S/P): ";a$
20 IF a$="P" OR a$="p" THEN j=8
30 MEMORY &4000
40 GOSUB 240
50 INK 1,26:MODE 2
60 INK 0,0:BORDER 0
70 b=&4188
80 d=0
90 FOR t=2 TO 5
100 FOR s=&B TO &13
110 er%=0
120 CALL &A000,b,d,t,s,@er%
130 b=b+&200:NEXT s
140 NEXT t
150 d=&420D
160 a=PEEK(d)
170 IF d=&4A90 THEN END
180 IF a>127 THEN 210
190 IF a=4 THEN GOSUB 220
```

```
200 PRINT#j, CHR$(a);
210 d=d+1: GOTO 160
220 WHILE INKEY$="": WEND
230 RETURN
240 WHILE p$<>"end"
250 READ p$
260 p=VAL("&"+p$)
270 POKE &A000+a,p
280 y=y+p
290 a=a+1:WEND
300 IF y<>&1E29 THEN PRINT "DATA ERROR":END
310 RETURN
320 DATA fe,05,c0,dd,4e,02,dd,56
330 DATA 04,dd,5e,06,dd,6e,08,dd
340 DATA 66,09,dd,e5,d5,e5,c5,21
350 DATA 3c,a0,cd,d4,bc,22,3d,a0
360 DATA 79,32,3f,a0,c1,e1,d1,df
370 DATA 3d,a0,30,03,21,00,00,dd
380 DATA e1,dd,56,01,dd,5e,00,eb
390 DATA 73,23,72,c9,84,00,00,00
400 DATA end
```

The above, as it stands will find and print the message you get when holding down the B U G S keys. Alter the lines below to get the other hidden story from the disc:

```
 90 FOR t=38 TO 39
 100 FOR s=&C1 TO &C9
 150 d=&5300
 170 IF d=&6600 THEN END
```

Richard Hodges

Hard Cat

Here is an easier way to get the same result from the CAT#8 program.

First boot up CPM. Insert the disc you want to catalogue. Press Control-P followed by Return. Type D I R, again followed by Return. When the printer has finished its business press Control-P. Et voila, you have your hard-copy cat.

Andrew Sharp

Tasword revisited

More useful locations for Tasword 6128:
POKE 6632, page number header space
POKE 6633, header text space
POKE 6634, footer text space
POKE 6635, page number footer space

Graham Bennett

Smoothly scrolling

The other day I was playing about with some of the Amstrad's Out commands and discovered some interesting results. Reset the computer before typing in each of the following:

```
 1 OUT &BC00,4: OUT &BD00,255
 2 OUT &BC00,4: OUT &BD00,1
 3 OUT &BC00,5: OUT &BD00,120
 4 OUT &BC00,9: OUT &BD00,1
 5 OUT &BC00,9: OUT &BD00,0
```

Enter this listing and watch that screen fly:

```
10 MODE 1:PAPER 3:CLS:BORDER 16
20 PEN 1:LOCATE 4,13
30 PRINT"SCROLLING THE SCREEN ";:PEN 0
40 PRINT" (literally). ":t=45:PEN 2
50 PRINT SPACE$(52)"By David Hall"
60 OUT &BC00,2:OUT &BD00,t:FOR d=1 TO 3
70 NEXT:m=t*9:t=t-1:IF t=0 THEN t=63
80 SOUND 1,m,1:IF t=45 THEN 90 ELSE 60
90 FOR t=0 TO 2000:NEXT:GOTO 20
```

On the subject of minus signs, try this:

```
PRINT -(114 minuses)- (return)
```

David Hall

Cavorting with characters

Three tiny listings capable of transforming your character set. Roll up, roll up: italic, thin and bold are here for the taking:

```
10 ' Bold characters
20 CLS:SYMBOL AFTER 32
30 FOR A=HIMEN TO HIMEN+767
40 B=PEEK(A):POKE A,B AND B\2
50 NEXT A
```

```
10 ' Italic characters
20 CLS:SYMBOL AFTER 32
30 FOR A=HIMEN TO HIMEN+767 STEP 8
40 FOR B=1 TO 4
50 A$=BIN$(PEEK(A+B),8)
60 B$=RIGHT$(A$,1)+LEFT$(A$,7)
70 POKE A+B,VAL("&X"+B$)
80 NEXT B,A
```

```
10 ' Thin characters
20 CLS:SYMBOL AFTER 32
30 FOR A=HIMEN TO HIMEN+767
40 B=PEEK(A):POKE A,B OR B\2
50 NEXT A
```

To turn off these new fonts you must enter: SYMBOL AFTER 256.

Alastair Scott

Amsmatch

A memory game

by Roy Lundquest

We've held on to this one for rather a long time (sorry Roy), but at last it surfaces.

It's a memory game for one player, based on the old card game of Concentration, and is played against the computer at any of five levels of difficulty. At level one, the computer will store just two of the previous cards that have been turned up, and at level five (the most difficult) it will store ten. The card symbols come straight from the CPC's extra character set.

The game was written for a 6128. 464 owners will notice the COPYCHR$ command in line 1140 which won't work on their machine. To get a playable version delete everything on that line after the "q" (ie. :pe=ASC(COPYCHR$(#0)).

```
1 '************************************
2 '**        AMSMATCH  by Roy Lundquest
3 '************************************
4 '
10 GOSUB 2400:MODE 1:INK 1,24:INK 2,20:INK 3,16:PEN 1
20 LOCATE 18,11:PEN 2:PRINT"AMSMATCH"
30 LOCATE 12,14:PRINT"by R.Lundquest,1986"
40 MOVE 100,100:DRAW 540,100,3:DRAW 540,300,3:DRAW 100,300
,3:DRAW 100,100,3:PEN 1
50 DEFSTR g:xx=7:yy=8:nc=14
60 DIM g(xx,yy),r(nc),c$(nc)
70 DATA 163,224,225,203,248,227,228,229,233,234,235,236,23
7,239
80 FOR x=1 TO nc:READ a:c$(x)=CHR$(a):NEXT:b$="    "
98 '**       randomize cards          **
100 FOR x=1 TO nc:r(x)=0:NEXT
110 ps=0:cs=0:FOR x=1 TO xx:FOR y=1 TO yy
120 r=INT(RND(1)*nc+1):IF r(r)=4 THEN 120
130 g(x,y)=c$(r):r(r)=r(r)+1
140 NEXT y,x
150 LOCATE 1,22:PRINT"What difficulty (1-5) 1=easy, 5=hard
";:INPUT d
160 IF d<1 OR d>5 THEN 150 ELSE mx=2*d:FOR x=1 TO mx:m(x)=
0:m$(x)="":NEXT:mc=0
170 LOCATE 1,24:INPUT"Your name";na$
198 '**       set up board            **
200 CLS:GOSUB 1600
210 FOR y=1 TO yy:FOR x=1 TO xx
220 LOCATE 5*x-2,y*2+4
230 q=(y-1)*xx+x
240 PRINT q;
250 NEXT x,y
260 FOR y=1 TO 9:y1=330-32*(y-1):MOVE 18,y1:DRAW 578,y1,3:
NEXT
270 FOR x=1 TO 8:x1=18+5*16*(x-1):MOVE x1,330:DRAW x1,74,3
::NEXT
298 '**       player's turn           **
300 mf=0:FOR t=1 TO 2:x(t)=0:NEXT
310 FOR t=1 TO 2
320 GOSUB 700:LOCATE 1,23:PRINT"Which number ";na$;:INPUT
x(t)
330 IF x(t)<1 OR x(t)>xx*yy THEN 320
340 GOSUB 1100:IF pe=32 THEN 320
350 IF x(1)=x(2) THEN 320 ELSE GOSUB 1400
360 NEXT:GOSUB 800:IF t$(1)=t$(2) THEN GOSUB 900:GOSUB 193
0:GOTO 380
370 GOSUB 1500:GOTO 390
380 mf=1:ps=ps+1:GOSUB 1700:IF en=1 THEN 400
390 IF mf=1 THEN 300 ELSE 500
398 '**       end of game             **
400 CLS:PRINT na$;"....";ps;TAB(20)"Arnold....";cs
410 LOCATE 10,12:IF ps>cs THEN PRINT"YOU WIN!!!" ELSE PRIN
T"I WIN !!!"
420 LOCATE 5,18:INPUT"Like another game (y/n)";ag$:IF LOWE
R$(ag$)<>"y" THEN 440
430 CLS:LOCATE 10,12:PRINT"Shuffling";xx*yy;"cards...":GOT
O 100
440 CLS:LOCATE 10,12:PRINT"bye for now ";na$:END
498 '**       Arnold's turn           **
500 x(1)=0:x(2)=0:GOSUB 2000
510 IF mf=1 THEN 590
520 t=1:GOSUB 1200:d1=500:GOSUB 1000
530 t=2:GOSUB 2300:IF mf=1 THEN 560
540 GOSUB 1200:d1=500 :GOSUB 1000:IF t$(1)=t$(2) THEN GOSU
B 900:cs=cs+1:GOSUB 1700:IF en =1 THEN 400 ELSE GOSUB 1000
:GOTO 520
550 d1=1000:GOSUB 1000:GOSUB 1500:GOTO 300
560 d1=500:GOSUB 700:GOSUB 1300:GOSUB 1000:GOSUB 1100:GOSU
B 1400:GOSUB 1000:GOSUB 900
570 cs=cs+1:GOSUB 1700:IF en=1 THEN 400
580 GOSUB 1900:GOTO 500
590 FOR t=1 TO 2:GOSUB 700:GOSUB 1300:d1=500:GOSUB 1000
600 GOSUB 1100:GOSUB 1400:NEXT:d1=500:GOSUB 1000
610 GOSUB 900:cs=cs+1:GOSUB 1700: IF en=1 THEN 400
620 m(d1)=0:m(d2)=0:GOSUB 2100:GOTO 520
698 '**       erase                   **
700 LOCATE 1,23:PRINT SPACE$(38);:RETURN
798 '**       delay loop              **
800 FOR d1=1 TO 800:NEXT:RETURN
898 '**       SUCCESS !!!             **
900 GOSUB 2410:GOSUB 700:LOCATE 1,23:INK 2,15,21:PEN 2:PRI
NT"That's a match!!!"
910 GOSUB 1000:INK 2,20:PEN 1:RETURN
998 '**       delay loop              **
1000 FOR d=1 TO d1*2:NEXT:RETURN
```

```
1098 '**     convert number to co-ordinates  **
1100 z=x(t)
1110 x=((z-1) MOD xx)+1:y=INT((z-0.5)/xx+1)
1120 p=x*5-2:q=y*2+4
1130 p(t)=p:q(t)=q
1140 LOCATE p+1,q:pe=ASC(COPYCHR$(#0))
1150 t$(t)=g(x,y):RETURN
1198 '**     Arnold chooses randomly **
1200 x(t)=INT(RND(1)*xx*yy+1)
1210 IF x(1)=x(2) THEN 1200
1220 GOSUB 1100:IF pe=32 THEN 1200
1230 FOR i=1 TO mc:IF x(t)=m(i) THEN 1200
1240 NEXT
1250 GOSUB 700:GOSUB 1300:GOSUB 1400:RETURN
1298 '**     Arnold's Choice         **
1300 LOCATE 1,23:PRINT"I choose...";x(t)
1310 RETURN
1398 '**     put character in place of number **
1400 LOCATE p+1,q:PRINT"  ";:LOCATE p+1,q:PEN 2:PRINT g(x,
y);:PEN 1
1410 t$(t)=g(x,y):RETURN
1498 '**     put numbers back        **
1500 FOR t=1 TO 2:LOCATE p(t),q(t):PRINT x(t):NEXT
1510 n=2:GOSUB 2200:RETURN
1598 '**     scores                  **
1600 LOCATE 1,1:PRINT na$;"...";ps;TAB(20)"Arnold...";cs:R
ETURN
1698 '**     end of game ??          **
1700 en=0:m=m+1:IF m=xx*yy/2-1 THEN en=1
1710 GOSUB 1600:FOR t=1 TO 2:LOCATE p(t),q(t):PRINT b$:NEX
T
1720 IF en=0 THEN RETURN
1798 '**     do last 2 remaining squares **
1800 z=0:count=1:WHILE count<3
1810 z=z+1:t=count:x(t)=z:GOSUB 1100
1820 IF pe=32 THEN 1840
1830 GOSUB 1400:count=count+1
1840 WEND
1850 d1=500:GOSUB 1000
1860 RETURN
1898 '**     check for numbers after match  **
1900 IF mc=0 THEN RETURN ELSE FOR t=1 TO 2:FOR j=1 TO mc
1910 IF m(j)=x(t) THEN m(j)=0
1920 NEXT j,t:GOSUB 2100:RETURN
1998 '**     check memory for match  **
2000 mf=0:IF mc<2 THEN RETURN ELSE i=1:j=2
2010 IF m$(i)=m$(j) THEN 2050
2020 j=j+1:IF j<=mc THEN 2010
2030 i=i+1:IF i<mc THEN j=i+1:GOTO 2010
2040 RETURN
2050 mf=1:x(1)=m(i):x(2)=m(j):d1=i:d2=j:RETURN
2098 '**     update memory after match **
2100 x=1
2110 IF m(x)<>0 THEN 2150
2120 IF x=mc THEN 2140
2130 FOR y=x TO mc-1:m(y)=m(y+1):m$(y)=m$(y+1):NEXT
2140 m(mc)=0:m$(mc)="":mc=mc-1
2150 x=x+1:IF x>mc THEN RETURN ELSE 2110
2198 '**     add n items x(j),t$(j) to memory **
2200 FOR j=1 TO n
2210 i=1:IF mc=0 THEN 2230
2220 IF x(j)=m(i) THEN 2280 ELSE i=i+1:IF i<=mc THEN 2220
2230 IF mc=mx THEN 2250
2240 mc=mc+1:m(mc)=x(j):m$(mc)=t$(j):GOTO 2280
2250 FOR x=1 TO mc-1
2260 m(x)=m(x+1):m$(x)=m$(x+1)
2270 NEXT:m(mc)=x(j):m$(mc)=t$(j)
2280 NEXT:RETURN
2298 '**     check 1 item against memory **
2300 mf=0:IF mc=0 THEN RETURN ELSE FOR j=1 TO mc
2310 IF t$(1)=m$(j) AND m(j)>0 AND x(1)<>m(j) THEN x(2)=m(
j):mf=1:RETURN
2320 NEXT:RETURN
2398 '**     sound effects           **
2400 FOR so=1 TO 5:SOUND 1,30*2^so,30:NEXT:RETURN
2410 SOUND 1,239,20:SOUND 1,179,50:RETURN
```

Controlling Patterns

Here is a gem by Alex Gough. It draws multi-coloured lines whose direction you control with the cursor cluster. There are five different line patterns to choose from - even a mirror effect, done by pressing the spacebar.

Palette switching (or ink swirling as Alex puts it) is put into effect by pressing R. It's very impressive and soothing.

```
1 ' Controlling patterns
2 ' by Alex Gough
3 ' Amstrad User Oct 87
10 BORDER 0: DEG: DIM b(16): MODE 0: INK 0,0
20 REM set ink colours
30 FOR a=1 TO 15: READ b(a): NEXT
40 FOR a=1 TO 15: INK a,b(a): NEXT
50 DATA 1,3,6,9,13,17,21,24,26,2,4,15,22,11,19
60 x=150: y=100: i=1: m=0
70 ORIGIN 320,200: PP=1: REM patttern type
80 REM plot coordinates and draw mathematical functions
90 IF pp=1 THEN PLOT x+50*SIN(y),y+50*COS(x),i: DRAW x,y
100 IF pp=2 THEN PLOT x+1/2*y+1/2*x,i:DRAW x,y
110 IF pp=3 THEN PLOT x+50*SIN(y),y+50*SIN(x),i: DRAW x,y:
 PLOT x+50*COS(y),y+50*SIN(x),i:DRAW x,y
120 IF pp=4 THEN PLOT x+50*SIN(y),y+50*COS(x),i: DRAW x,y:
 PLOT -x-50*SIN(y),-y-50*COS(x): DRAW-x,-y
130 IF pp=4 THEN PLOT -x-50*SIN(y),y+50*COS(x),i: DRAW -x,
y: PLOT x+50*SIN(y),-y-50*COS(x),i: DRAW x,-y
140 IF pp=5 THEN PLOT -x-50*SIN(y),y+50*COS(x),i: PLOT x+5
0*SIN(y),-y-50*COS(x)
150 IF pp=5 THEN PLOT x+50*SIN(y),y+50*COS(x),i: PLOT -x-5
0*SIN(y),-y-50*COS(x)
160 IF INKEY(0) AND y>-200 THEN y=y-4
170 IF INKEY(1) AND x>-300 THEN x=x-4
180 IF INKEY(2) AND Y<200 THEN y=y+4
190 IF INKEY(8) AND x<300 THEN x=x+4
200 a$=LOWER$(INKEY$)
210 IF a$=" " THEN pp=pp+1: IF pp=6 THEN pp=1
220 IF a$="c" THEN CLS
230 IF a$="r" THEN 280
240 IF a$="m" THEN m=m+1: GOSUB 350
250 i=i+1: IF i>13 THEN i=1
260 GOTO 90
270 REM rotate inks
280 FOR g=1 TO 15
290 FOR h=1 TO 15
300 IF INKEY$=" " THEN GOTO 90
310 INK h,b((g+h)MOD 15)
320 NEXT: NEXT
330 GOTO 280
340 REM change graphical drawing mode
350 IF m=2 THEN m=3
360 IF m=4 THEN m=0: SOUND 1,100,50,5
370 PRINT CHR$(23);CHR$(m)
380 RETURN
```

Star-dodging

A one-liner game. Yes, it's true! Rajiv Gatha has managed to squeeze all the elements of a hot arcade game onto one line (not including the Rem statements). Type in the listing and dodge those stars using keys Z and X to move the space ship.

```
1 'Star-Dodging
2 'The Amstrad User Oct 87
3 'by Rajiv Gatha
10 MODE 0:s=306:WHILE(TEST(s+8,18))=0:TAG:MOVE s,14:PRINT
CHR$(239);:MOVE RND(1)*620,399:PRINT CHR$(46);:TAGOFF:LOCA
TE 1,1:PRINT CHR$(11);CHR$(11);:s=s-((INKEY(71)+1) AND s>=
0)*4:s=s+((INKEY(63)+1) AND s<=612)*4:WEND:PRINT"WHOOPS":F
OR t=0 TO 1000:NEXT:RUN
```

Fuzzy

Here's an interesting routine from David Hudson. It simulates the fuzzy picture seen when your television set goes haywire. How many people can you kid into believing their screen is on the blink.

```
1 ' Fuzzy
2 ' by David Hudson
3 ' The Amstrad User Oct 87
10 ON BREAK GOSUB 60
20 MODE 1:BORDER 0:INK 0,0:INK 1,26
30 INK 2,13:INK 3,0:FOR p=&C000 TO &FFFF
40 POKE p,INT(RND*256):NEXT p
50 OUT &BD00,INT(RND*256):GOTO 50
60 MODE 2:END
```

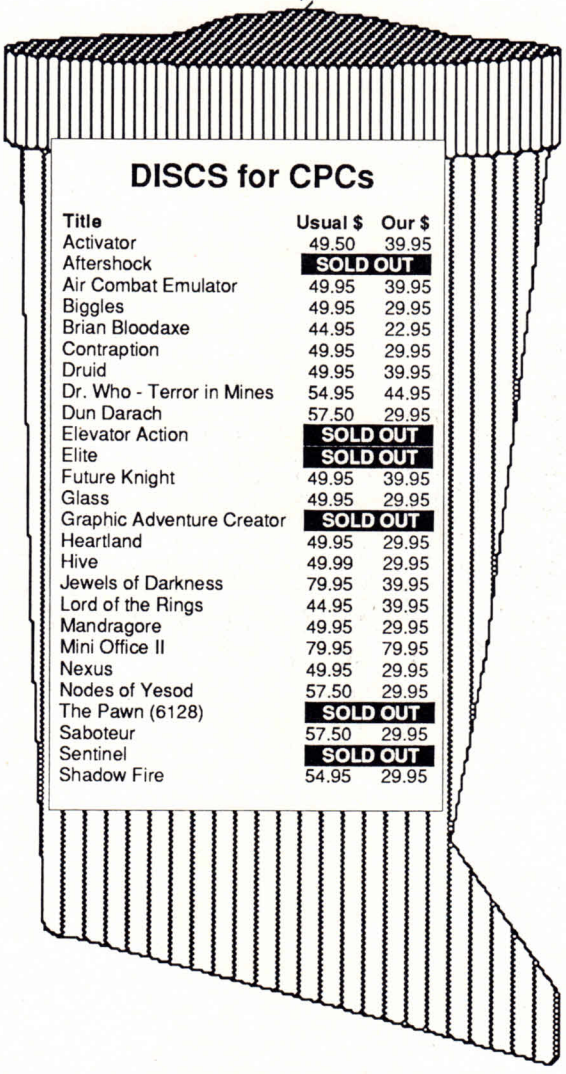
Sine-wave writing

Andrew Perry and Andrew Crawford have clubbed together to produce an interesting listing that move a message across the screen in the form of a sine-wave. The program asks you to input the message you wish moved around the screen. Best results are obtained by using short sentences or a single word.

```
1 ' Sine-wave writing
2 ' by Andrew and Andrew
3 ' Amstrad User Oct 87
10 MODE 1: BORDER 0: INK 0,0: INK 1,26: DEG
20 TAGOFF: INPUT"Enter message ";a$: CLS: TAG
30 FOR n=-100 TO 740 STEP INT (RND*15)+3
40 PLOT -1000,-1000,INT(RND*3)+1
50 MOVE n+4*COS(n),200+150*SIN(n+RND(20))
60 PRINT a$;: x=n: x=x-100
70 MOVE x+4*SIN(x),200+150*COS(n+RND(20))
80 PRINT a$;: NEXT: GOTO 20
90 GOTO 10
```

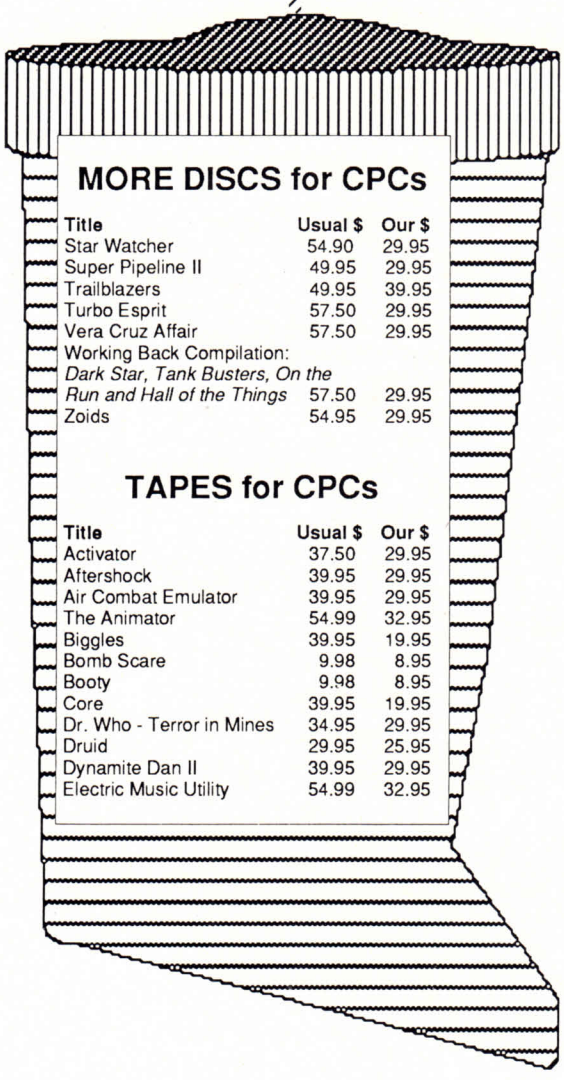
Send your Type-Ins to: The Amstrad User, 1/245 Springvale Road, Glen Waverley, Vic 3150

How can you
at this tim
By snapping up th
bargains* and



DISCS for CPCs

Title	Usual $	Our $
Activator	49.50	39.95
Aftershock	SOLD OUT	
Air Combat Emulator	49.95	39.95
Biggles	49.95	29.95
Brian Bloodaxe	44.95	22.95
Contraption	49.95	29.95
Druid	49.95	39.95
Dr. Who - Terror in Mines	54.95	44.95
Dun Darach	57.50	29.95
Elevator Action	SOLD OUT	
Elite	SOLD OUT	
Future Knight	49.95	39.95
Glass	49.95	29.95
Graphic Adventure Creator	SOLD OUT	
Heartland	49.95	29.95
Hive	49.99	29.95
Jewels of Darkness	79.95	39.95
Lord of the Rings	44.95	39.95
Mandragore	49.95	29.95
Mini Office II	79.95	79.95
Nexus	49.95	29.95
Nodes of Yesod	57.50	29.95
The Pawn (6128)	SOLD OUT	
Saboteur	57.50	29.95
Sentinel	SOLD OUT	
Shadow Fire	54.95	29.95



MORE DISCS for CPCs

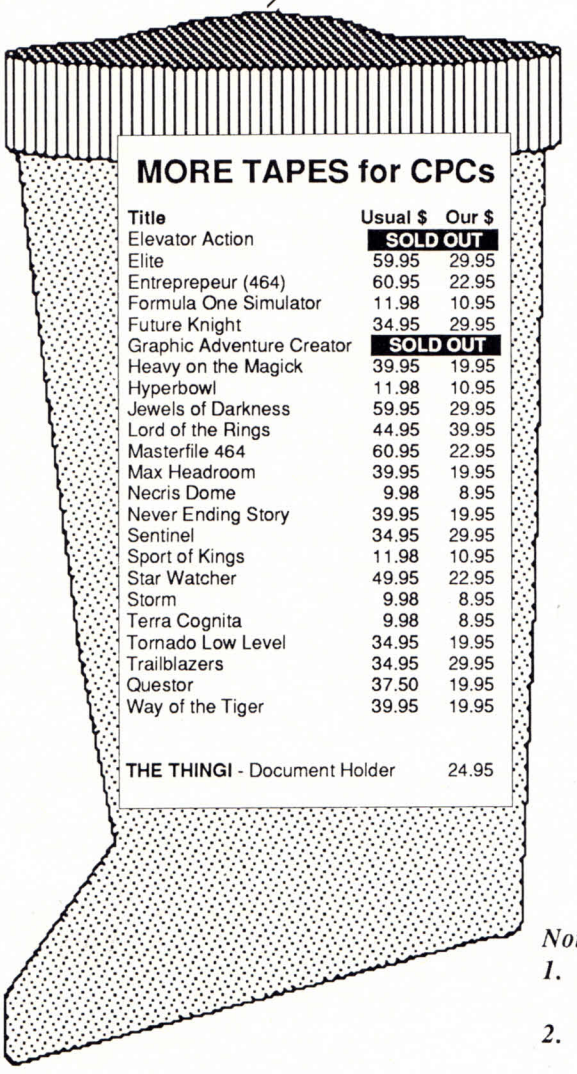
Title	Usual $	Our $
Star Watcher	54.90	29.95
Super Pipeline II	49.95	29.95
Trailblazers	49.95	39.95
Turbo Esprit	57.50	29.95
Vera Cruz Affair	57.50	29.95
Working Back Compilation: *Dark Star, Tank Busters, On the Run and Hall of the Things*	57.50	29.95
Zoids	54.95	29.95

TAPES for CPCs

Title	Usual $	Our $
Activator	37.50	29.95
Aftershock	39.95	29.95
Air Combat Emulator	39.95	29.95
The Animator	54.99	32.95
Biggles	39.95	19.95
Bomb Scare	9.98	8.95
Booty	9.98	8.95
Core	39.95	19.95
Dr. Who - Terror in Mines	34.95	29.95
Druid	29.95	25.95
Dynamite Dan II	39.95	29.95
Electric Music Utility	54.99	32.95

**** Nearly 70% off original RRP on some items***

think of Xmas
e of year?
ese stocking-filler
saving NOW!



MORE TAPES for CPCs

Title	Usual $	Our $
Elevator Action	SOLD OUT	
Elite	59.95	29.95
Entreprepeur (464)	60.95	22.95
Formula One Simulator	11.98	10.95
Future Knight	34.95	29.95
Graphic Adventure Creator	SOLD OUT	
Heavy on the Magick	39.95	19.95
Hyperbowl	11.98	10.95
Jewels of Darkness	59.95	29.95
Lord of the Rings	44.95	39.95
Masterfile 464	60.95	22.95
Max Headroom	39.95	19.95
Necris Dome	9.98	8.95
Never Ending Story	39.95	19.95
Sentinel	34.95	29.95
Sport of Kings	11.98	10.95
Star Watcher	49.95	22.95
Storm	9.98	8.95
Terra Cognita	9.98	8.95
Tornado Low Level	34.95	19.95
Trailblazers	34.95	29.95
Questor	37.50	19.95
Way of the Tiger	39.95	19.95

THE THINGI - Document Holder 24.95

DISCS for PCWs

Title	Usual $	Our $
Aftershock	57.50	47.50
Jewels of Darkness	79.96	39.95
Lord of the Rings	44.95	39.95
Masterfile 8000	109.00	109.00

EXPANSIONS

dk'Tronics 256k Silicon Disc (464/664 only)	309.95	94.95
dk'Tronics 256k Memory Expansion (6128 only)	329.95	94.95

DISC DRIVES

Amstrad DDI-1 plus interface to suit a CPC464	499.00	399.00
Amstrad FD-1 plus cable to suit CPC 664 0r 6128	449.00	349.00

Mail your orders to:
THE AMSTRAD USER
1/245 Springvale Road
Glen Waverley, Vic 3150

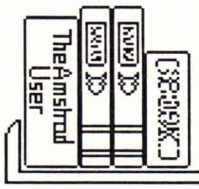
Phone your firm orders to:
(03) 233 9661
Bankcard, Mastercard or Visa accepted.

Note:

1. *The items listed on this page are available until our stocks run out. When ordering, please provide an alternative choice, just in case.*
2. *We cannot reserve items - only firm orders will be supplied and on a first come - first served basis.*
3. *The special prices shown for DISC DRIVES will be held until 30th October and then reviewed.*
4. *All prices include postage and packing within Australia.*

The Amstrad User
MAIL ORDER SHOPPING

Send orders to The Amstrad User
1/ 245 Springvale Road,
Glen Waverley, Victoria 3150



BOOKS

CPC TITLES

	Subscriber Price	Normal Price
Advanced Amstrad Programming Techniques	$22.95	$24.95
Advanced User Guide	$19.75	$21.95
Adventure Games for the Amstrad CPC464	$25.15	$27.95
Amstrad Compendium	$20.65	$22.95
Amstrad Games Book (Pitman)	$14.35	$15.95
Amstrad Games Book (Melb. House)	$20.65	$22.95
Amstrad Pentacle Adventure Creator	$ 8.05	$ 8.95
Basic BASIC - *New Ed.*	$14.95	$15.95
Childs' Guide to the Amstrad Micro	$11.65	$12.95
Disc System, The Amstrad CPC 464	$25.15	$27.95
Dynamic Games for the Amstrad	$17.95	$19.95
Filing Systems and Data Bases for the CPC464	$27.95	$29.95
Graphics Programming Techniques	$22.45	$24.95
High Energy Programs for the Amstrad (Post Free)	$ 9.95	$ 9.95
Ins and Outs of the Amstrad	$20.65	$22.95
Machine Code for Beginners	$17.95	$19.95
Machine Lang. for the Absolute Beginner	$20.65	$22.95
Making Music on the 464/664 (OK for 6128 too)	$19.75	$21.95
Master Machine Code on your 464/664	$19.75	$21.95
Pitman's First Book of Amstrad Games	$11.65	$12.95
Powerful Programming for Amstrad 464/664/6128	$22.55	$25.05
Practical "C"	$25.80	$28.65
Practical Programs for the 464	$21.55	$23.95
Ready made Machine Language routines	$20.65	$22.95
Structured Programming on 464/664/6128	$26.95	$29.95
Whole Memory Guide - 464	$26.95	$29.95
Working Amstrad	$17.95	$19.95
Writing Adventure Games on 464/664	$20.65	$22.95

PCW TITLES

Amstrad Companion *Due in this month*	$25.95	$27.95
LocoScript Pocketbook	$16.95	$17.95
Mastering the Amstrad PCW 8256/8512	$29.50	$32.25
Program your PCW	$29.95	$32.95

CP/M TITLES

CP/M - The Software Bus: a programmer's companion: with CP/M+	$29.95	$32.25

LOGO TITLES

LOGO Pocketbook	$16.95	$17.95
Practical Logo on the Amstrad (under CP/M+)	$25.95	$27.95
Using DR Logo on the Amstrad (under CP/M 2.2)	$35.95	$37.95

PC1512 TITLES

Basic2 - Bk. 2 (Soft 50012)	$37.50	$39.95
Business Presentation Graphics on the PC1512	$50.00	$55.00
Using DOS Plus on the Amstrad PC1512	$37.95	$39.95
Using GEM on the Amstrad PC1512	$50.00	$55.00

*All book orders **under $20 are sent free** of any postage or packing charges. For all book orders **over $20 please add $5.00** for postage and packing (regardless of quantity ordered).*

As most of the above books are imported, the price and availability may change without notice.

BINDERS

In white vinyl with silver logos of "THE AMSTRAD USER" on the front and the spine. Protect your collection of The Amstrad User. Each binder holds 12 issues. Will also take the English "Amstrad Computer User" but sometimes not so many issues. Price, including postage, is $12.95.

SOFTWARE

See previous two pages for great savings

PCW OWNERS

We have ONE Electric Studio Mouse left. Comes complete with interface and Graphics package. Normally $429.95 - our price just **$299.95** including postage.

We also have just ONE Video Digitiser. All you need is a video camera! Normally priced at $349.95 - our price just **$244.95** including postage.

Please confirm availability before ordering.

BACK COPIES

THE AMSTRAD USER

Issues 1 to 9 each	$4.00
Note: Mar '85, Jun '85 and Sep '85 are 'Out of Print'	
Issues 10 to 21 each	$4.50
Issues 22 to 32 each	$4.75
Tapes for any of the above including those issues Out of Print are available, each at	$5.00
Year Disc 1 - Issues 1 to 12	$50.00
Year Disc 2 - Issues 13 to 16	$22.50
Year Disc 3 - Issues 17 to 20	$25.00
Year Disc 4 - Issues 21 to 24	$25.00
Year Disc 5 - Issues 25 to 28	$25.00
Year Disc 6 - Issues 29 to 32	$25.00

Tapes or Discs not suitable for PCWs or PCs

THE AMSTRAD COMPUTER USER (English)

Issues 3 to 7 each	$5.00
Issues 12 to 19 each	$5.50
Issues 21 to 33 each	$6.00

Note: Issues 8 to 11 (Jul '85 to Oct '85), 20 (Jul '86) and 31 (Jun '87) are not available.

Prices include postage on all Back copies, discs and tapes.

Bankcard, Mastercard or Visa accepted on all orders over $5.00

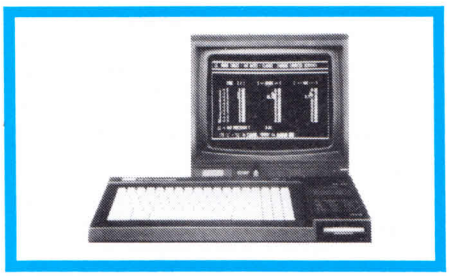
Whether your investment was $500 or $2500 ...

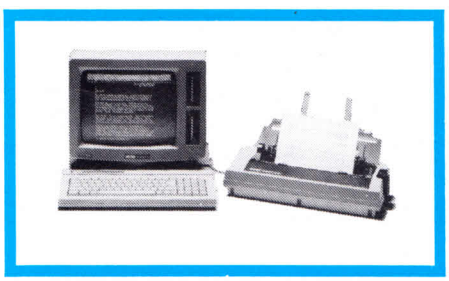
... you can't afford to be without The Amstrad User

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.





Please send me 12 monthly issues of The Amstrad User. I own/intend to own a _____

I would like: the Magazine only at $37.50 ❑ Magazine plus cassette of programs appearing in that issue at $80.00 (tapes are not suitable for PCW or PC owners) ❑

I wish my subscription to start with the current issue ❑ or Please start at Issue No ....

I enclose a cheque ❑ or please charge my Bankcard, Mastercard or Visa ❑ for $ _____

The number is _____ The card expires on _____

Name _____

Address _____

_____ State _____ Post Code _____

Return this form to: THE AMSTRAD USER, Suite 1, 245 Springvale Road, Glen Waverley, Victoria 3150 or ring (03) 233 9661 for further information.

For subscriptions to Papua New Guinea, New Zealand, Solomon Islands, Vanuatu or New Caledonia please add $21 airmail. For Fiji, Brunei, French Polynesia, Indonesia, Kiribati, Malaysia, Nauru, Niue, Samoa, Singapore, Tokelau Islands or Tonga please add $27 airmail.

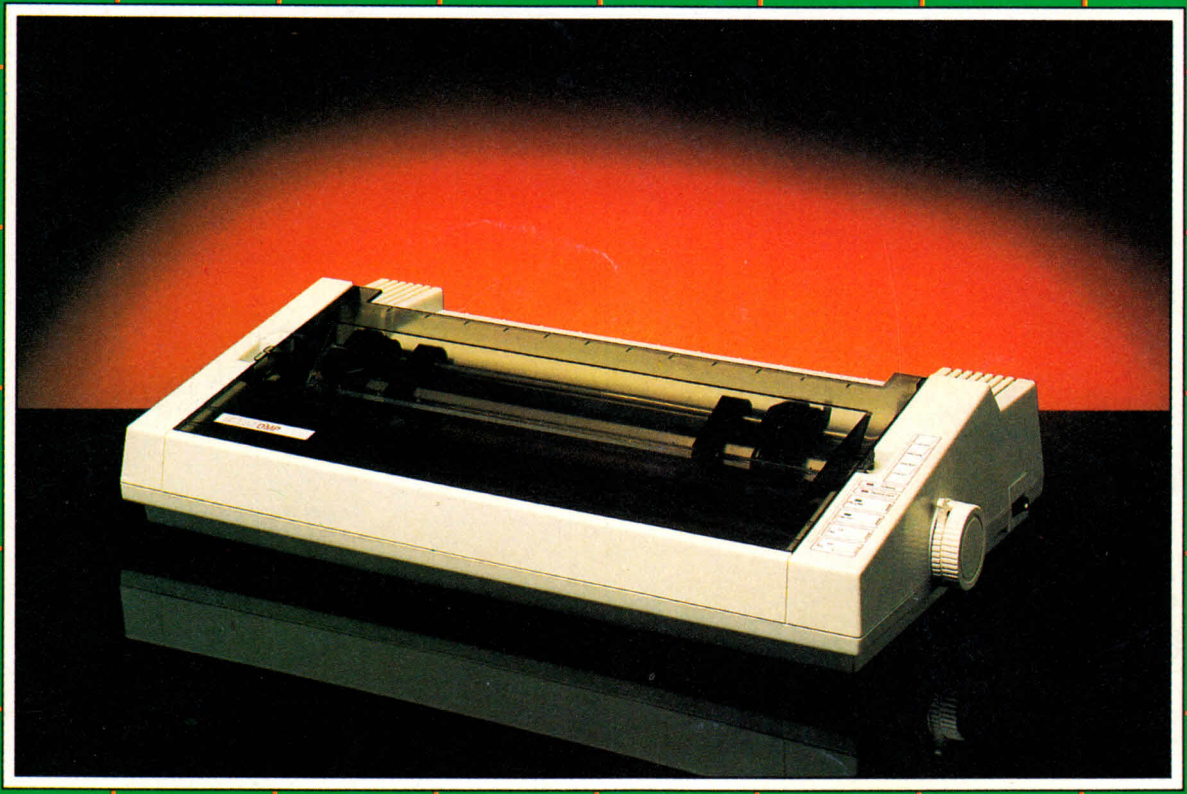
DMP4000 PRINTER

15" Wide Carriage PC Compatible Dot Matrix Printer

Introducing the DMP4000 - another superb performer in the AMSTRAD range of computer products.

The DMP4000 boasts industry standard software instructions and characters (including graphics), and is capable of handling a wide range of paper sizes -between 2-15 1/2 inches (50-395 mm) for single cut sheet or roll paper, and between 3-16 1/2 inches (75-420 mm) for continuous tractor feed paper or labels, all at printing speeds of up to 200 characters per second.

The extremely wide choice of sizes and typefaces coupled with a complete ASCII, international and IBM graphics character set should provide a solution to any printing problem. In addition, the implementation of dot addressable graphics and standard Epson compatible command codes will allow the DMP4000 to operate directly with most software, including word processors, spreadsheets and graphics programs.



Features

Handles 15 inch paper
High speed (up to 200 characters per second)
Low power consumption
Usable with friction or tractor feed paper
Epson compatible control codes
IBM graphics character set
Prints up to 233 characters per line
Over 100 different typeface combinations including NLQ (Near Letter Quality)

Computer peripherals by AMSTRAD