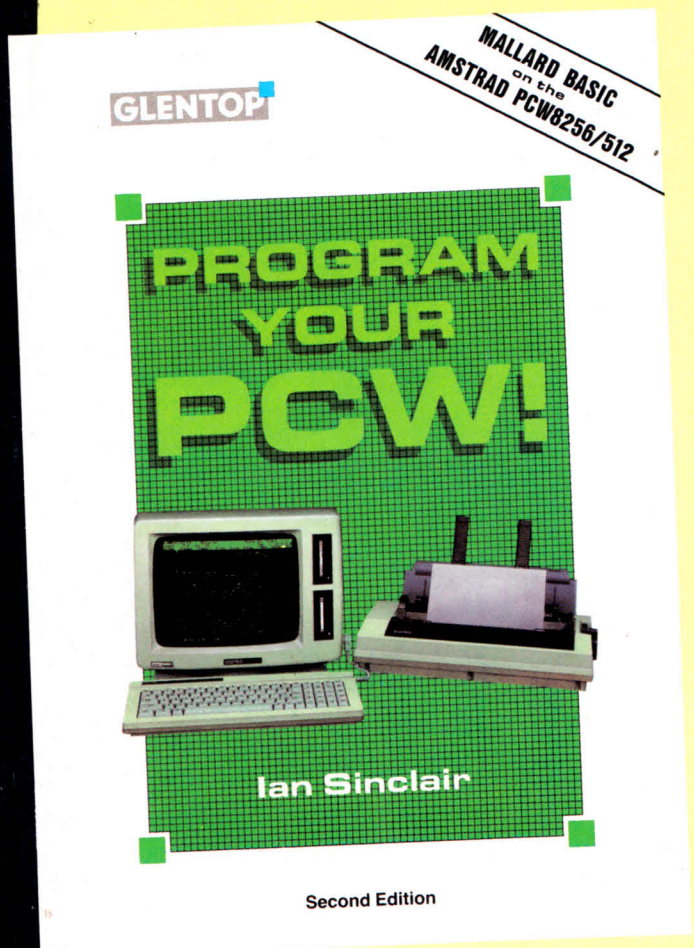
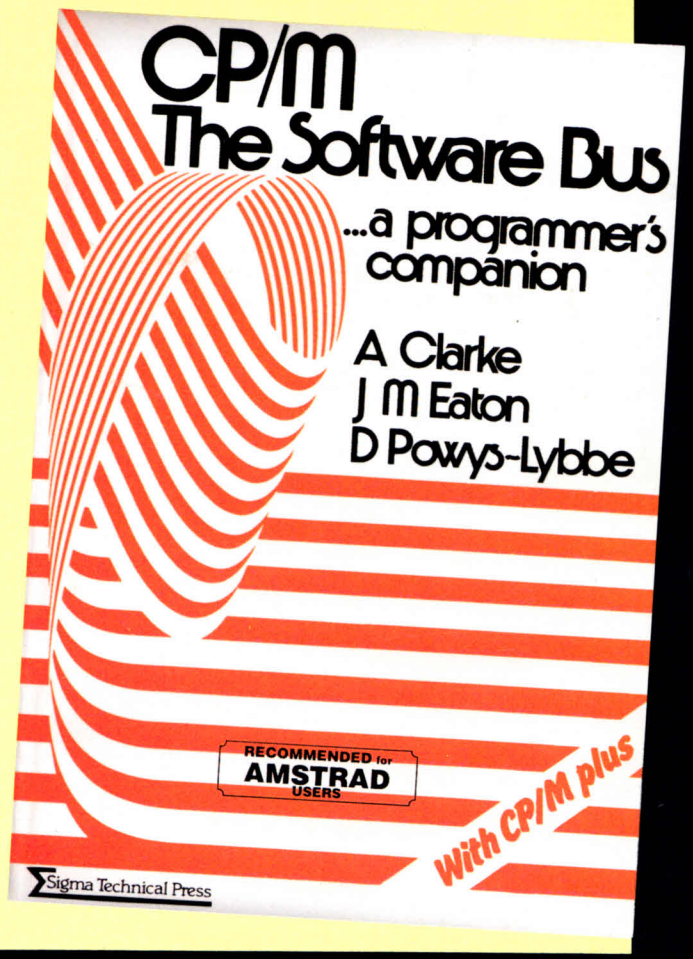
The Aussie Mag for Amstrad owners

THE AMSTRAD USER

Issue No. 31 $3.75 August 1987





- *A cheap Word Processor using CPC Basic rom + a card trick tutorial to help produce better programs*
- *Connect Four type-in for PCWs + review of Desktop Publisher with Mouse + more on LocoMail*
- *Review of Tasprint PC and Fleet Street Editor*

FOR THE NOVICE & EXPERIENCED USER

Registered by Australia Post – Publication No VBP7017

the SENTINEL

FIREBIRD

From The Amstrad User
Cassette: £14.95
Disc: £19.95
(Postage included)

CONTENTS

THE AMSTRAD USER

Issue No. 31
August 1987

ADVERTISER'S INDEX

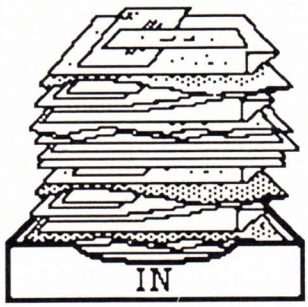
For Tape Subscribers, CPC programs appearing in this month's magazine can be found at the following approximate positions:

Side 1:	SIDEWAY2	- 9	CWSU	- 30	CHEAPW	- 38
	CTRICK	- 56	TDUMPBAS	- 76	TDUMPBIN	- 86
	SERPENT	- 92	TRIANGL	- 100	ADLIST1	- 110
	ADLIST2	- 122				
Side 2:	CRIBBAGE	- 10 (18 blocks)				

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. Reprinting of articles published in The Amstrad User is strictly forbidden without written permission. 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 issue of The Amstrad User so nearly became a victim of a senseless act of burgulary perpetrated by some brainless, criminal morons.
Without any respect of private property and the consequences of their actions, and clearly lacking even a modicum of morality, a number of persons broke into these offices and stole various pieces of computer equipment, discs and manuals.
Even with back-ups, it has still taken quite a while to re-create your magazine and other files.
If you get offered any discs containing data relevant to The Amstrad User, ring us or the Waverley CIB on (03) 566 1535.

I am not one who condones violence but the recent spate of, shall we say, unpleasant news about the air has prompted me to this. We are now residing in an era where violence masquerades beyond the realms of notification, that is, we are unaware of the apparent troubles that share a place in society today. Television and other forms of mass media seem to thrive on the exploits of the violent, thus leading me to the main concern of this letter, the violence in computer games.

Since the birth of Space Invaders, game players have found themselves immersed in the action of blasting aliens, stabbing enemy soldiers, randomly firing at poor defenceless prisoners etc. And along with the constant exposure to the violence that comes in other forms, this could very well inevitably affect the subconscious

Don't get me wrong. I do not think badly of computer games. As a matter of fact, I regard them as a highly advanced form of entertainment available to those elite few who can comprehend the utterly complex world or arcade, adventure and strategy. I find it hard to refuse a good arcade adventure regardless of the degree of violence involved.

Many arcade games seem to rely on violence as the major source of entertainment, and may I be as bold to state that ALL arcade games do so, from the insane massacre of Commando to the medieval carnage of Gauntlet. Though most of them dwell in the Good versus Evil genre, the element of "hack and slay" is still strong. And unless a "cheat mode" is available, there is very little chance that good will triumph over evil.

Even the strategically involving adventure games depend, to a certain extent, on violence. As is evident with "The Forest at Worlds End" and "Runestone", the command KILL is quite highly emphasised. The gripping world of pure strategy programs such as PSS' "Theatre Europe" and "Tobruk" both use the background of a violent situation as a solid base. Though, fortunately for mankind, the former actually tries to instigate the prevention of a nuclear holocaust rather than to encourage it.

And this finally boils down to one question; can a computer game exist without the influence of violence? Though seemingly rhetorical, this question can be answered. It is all up to the individual to actually interpret the concept behind this violence. And this draws my conclusion.

I would be most interested in the views of other readers regarding this rather controversial issue. One publisher of an Australian computer magazine even went as far as to state that games condoning violence would not be given favourable review in his magazine.

Personally, I think that to condemn a piece of brilliant entertainment software because of its inhumanity is wrong.

In the February 1987 issue of The Amstrad User, it was stated that "Dracula" from CRL was censored due to its highly violent portrayal of Bram Stoker's novel. Does this mean that it is not available in Australia? The adventure was highly acclaimed in the UK and I would not want to see a good adventure go to waste because of some censor board of narrow minded "chaps".

S.A. Mah, Willerton, WA

As mentioned last month, we have decided to pay $25 to the writer of "the chosen letter" rather than $5 to all. This is the chosen one!

I have been a subscriber to TAU since its inception.

Up until the introduction of the PCW I really looked forward to receiving my copies and a month seemed to be too long between issues.

I have got a bit browned off lately as I do not think it right that the PCW which is basically a business machine should have usurped about 30% of TAU

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.

at the expense of those of us who have supported the magazine from the word go.

Will the PC1512 also steal 30%?

Now that I've had my winge I'll ask a favour . . .

Could it be possible for you to translate and print some of the commands used in Microbee, VZ200, Commodore etc. programs to assist in adapting these foreign programs to Amstrad?

K.A. Partridge, Pt. Pirie, SA

Thank you for your support. As a "founder reader" you will remember that The Amstrad User started in February 1985 as a 32 page magazine for the CPC464 - that was the only machine available at the time. These days you should find, on average, there are at least that number of pages in each issue relevant to the CPC machines. Don't forget that TAU is for all Amstrads - we can't exclude one because it's not a games machine!

I've been a subscriber to TAU for twelve months now and I thought I'd let you know how much I've enjoyed the mag. I think I speak for most of your readers when I propose a vote of thanks to you in the editorial office and to all those contributors who take the time and trouble to present us with interesting programs and hints and tips on programming the beasts.

Without doubt I've learnt more about my 6128 through the mag than I'd ever have learnt through the manual. A couple of books have been helpful, but TAU is numero uno. Keep up the good work folks!

I made a small discovery with the 6128 which might help newcomers. If you write a two or three line program to redefine the f-keys (as per previous issues) and save it to disc, then on start-up RUN your program and then type in DELETE 1-3 (or whatever your last program line number is) then press ENTER. This wipes the program lines from memory and prevents them from being part of the next program you type in, but the keys still operate as defined - I love it!

Now my turn to ask for help. I typed in the Blackjack program (phew!!) and after much burning of midnight candles the program is now up and running - except that all the instructions etc. are unreadable (save for the letters that have been formed by re-defined keys). Can anyone tell me how to correct this state of affairs? The typing has been checked line by line (another phew!!) and I'd say it is word perfect.

Lastly, anyone considering a word processing program should seriously consider Tasword. It is good.

Rod Dent, Iluka, NSW

I have a couple of comments about material in the July 1987 issue.

POSTFIX: CPC Basic v1.1 will not allow the names following the line number of a GOSUB, so owners of a 6128 will need to delete them. I wonder whether anybody has compiled a list of differences between the two versions?

I think I have come across one (but did not have time to confirm it definitely): while v1.0 leaves string constants in the program area, v1.1 appears to move them to the string pool. Unfortunately this invalidates the idea of storing

machine language in string constants, and the way I found the start of Basic in "What makes Locomotive run" (Issue 15, April '86). As a Hot Tip points out (Issue 30, July '87), the start of Basic can be re-located, and the low memory could be used for storage of machine language routines.
MEMORIES: KL_ROM_WALK is called as part of the power-up initialization, and calling it again from within a program will cause the machine to lose its place. However this problem can be overcome by calling KL_CHOKE_OFF first: this clears all RSXs, event queues, etc. KL_ROM_WALK or KL_INIT_BACK can then be called (with the correct addresses in the registers) without upsetting the machine.

Petr Lukes, Toowoomba, QLD

Does anyone have a screen dump patch for Logo v1.1 running under CP/M 2.2? Why aren't there any "Cheat Modes" pokes for "Tomahawk"?

D.C. MacKinnon, Lake Hts, NSW

The answer to your second question is that nobody has given us any yet. Perhaps it is just too difficult to get into? How about it hackers (and don't forget the PCW).

I am a Grade 6 student and I was wondering if you could send me all or samples of the programs because I want something to do and also I've shown all my friends all that there is in the Amstrad CPC6128 manual.

My friends used to come over to my place and see all the programs. Now they are getting fed up with me showing them the same thing over and over again. Please send programs to me *(address supplied)* by the 20th July.

Matthew Stafford, Rosewood, QLD

What programs are you talking about? Have you tried entering some of the programs that appear in The Amstrad User? There are more than enough to keep you occupied during school holidays and that way you will learn how to use your Amstrad as a programming tool which could help you in the future.

Request from The Editor

I'd like to hear the views of our readers to the idea of printing a classified ad section. This could include anything with the exception of software for sale or swapping as I do not want to promote 'pirates' as undoubtedly they would abuse it.

Please send your comments direct to The Editor, The Amstrad User, 1/245 Springvale Road, Glen Waverley, Vic 3150.

Thank you.

Doo Dah disc shuffle

Nemesis is meddling with discs again - Pat McDonald tries its new utilities.

Bonzo Doo Dah: Disc only - $36.00

"Roll up, ladies and gentlemen! For your entertainment, Bonzo Doo Dah will balance a stick of rhubarb on his nose while juggling with burning tabloids!"

I'm sorry about that, but this program has tickled my funnybone. Bonzo Doo Dah is a disc utility. Its primary purpose is to give you more room on each disc - 230k, no less, which is a sizeable chunk over the standard 178k. It will also allow you to discover (sorry!) just what is on it, and change it to what you want, such as highscore tables and programmers' credits. This is known as hacking discs.

When run,the program presents a menu. Every word from now on is true. If it sounds rather strange, it is because it is strange.

Big K

Bigbonzy is the first option on the menu. From it, you can format discs in any standard format - IBM, Data, System and Vendor - and also in the new, larger format, called Bigbonzo. Once this has been set up, a file of the same name is written to the disc. To use the disc, just run this file first, and all should be well. However, I am not sure that it caters for CPM; the manual has nothing to say. So if you want to use CPM as well, be wary of trying to use Bigbonzo format.

There is also a rather effective disc clone option. I have little to say, since Nemesis claims this is a byproduct and not a design intent - the idea was to be able to copy the Bigbonzo format only. The fact that it can also copy various commercial games is a twist of fate. Personally I think people will have a field day ripping off disc software, then get bored of having 250 different games to play.

Bonzedit is the second choice on the main menu. From here you can interrogate standard, Bigbonzo and various (though not all) types of commercially protected discs. As a rule of thumb, the older the disc, the more chance Bonzo Doo Dah has of coping with it.

Good editing tools

The editing functions are really very good. First off you can get a map of the disc, showing which tracks and sectors are in use. This and many of the functions can be put out to a printer.

Next you can get a file-location map. This is a list of the files on the disc, and each filename has after it a series of double numbers, like this: 03/41. The first is a track number, and refers to which of the 40 tracks or concentric rings the data is stored on. The second is the sector number, and this indicates where on the track the data block is stored. So even if a program has been split into many pieces and scattered all over the disc - you would be very confused if you just studied sectors in sequence - you can use this feature and know just where the different blocks of the program have been placed.

There is also a function to alter the contents of the directory of the disc. This may sound dangerous, but it need not be with a little care. If you have accidentally erased a file, you can recover it again with this program if you have not written anything else on top of it. You can also set your programs to "System"; they will be invisible to anyone doing a standard "cat" or "dir" of a disc. These functions are well implemented and the approach is that you can make any correction or deletion you want to on the screen: the program will not change anything on the disc until you tell it to.

When you want to edit the data on the disc, Bonzo Doo Dah again provides you with a safety net. You can select the track you want to edit, and the program remembers as much of that track as you want to change. This means that you can quickly restore a track to its original state. It's a good feature, but if you find that you have made a mistake after you have switched to another track, that's too bad, because the program remembers only one track at a time. The moral: be careful despite the safety net.

Search warrant

The last editing option is to search the disc. You type in the string you want to search for - up to 128 bytes long, in Ascii or Hex. If the machine finds that particular sequence on the disc, it gives you the option of going to it under the Track and Sector editor, or to ignore it and search on. You can also jump to Bigbonzy or to the copying routine.

Bonzcopy is a copying program, rather like Filecopy on CPM 2.2. From a list of files you select the ones you want to copy, up to ten at one time. It can handle the maximum standard size of file under Amsdos. It will copy programs onto the Bigbonzo format.

On the rear of the disc is an example of a Bigbonzo format disc, including some very pretty pictures and a free adventure game. Some of the wit in the graphics is pretty sharp - I especially liked the Amiga lookalike! (Hasn't someone done that before?) Alluding to earlier Nemesis titles, these are labelled "Meddler diversions", and needless to say, they are.

The program's presentation is flawed only by the manual: one piece of paper typed on both sides. It contains everything relevant to the program, and if you know what you are about I doubt if you'll look at it much. Most of the program is self-explanatory - to the informed.

If you want to explore deeper into disc software, you will need a good piece of kit to help you. As commercial disc editors go, this is the best I've seen in a long while. It can't do everything, and will no doubt be superseded someday, but for its user-friendly approach, its sensible price, and its humour, I recommend it.

For more information on the above and other Bonzo products contact Pipeline-Computer Accessories on (049) 73 2754

Nationwide User Groups

Apart from a number of amendments that were received too late to go into last month's magazine so were held over, we are happy to welcome two new groups to our list. First, the **Hawkesbury Amstrad User Group** has at last taken off thanks to Terry Webb who has been patiently waiting in the Contact list for quite some time. Second, the **Amsnorth Amstrad User's Group** in SA is about to cause a power dip in Gepps Cross. Welcome to you both.

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 7.30p.m.

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: Tom Bird (09 457 5614)
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.

TOM PRICE AMSTRAD USER CLUB

President: Peter Hoffman (091 89 1608)
Secretary: Colin Smith
Treasurer: Mark Hedley-Smith
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 Tuesday.
Mail: PO Box 210, Parkholme, SA 5043

NORTHERN COMPUTING SOCIETY INC.

President: Grant Wilson (08 250 2760)
Treasurer: Percy Cook (08 248 1065)
Secretary: Judith Thamm (085 20 2377)
Venue: Salisbury North Primary School, cnr Bagster and Woodyates Rds every Wednesday from 7.00 pm.
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

NORTHERN TERRITORY AMSTRAD USER GROUP

President: Kevin Bateman (089 32 1463)
Treasurer: Greg Heron (089 27 8814)
Secretary: Colin Gorton (089 84 4655)
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.

President: Tony Blakemore (03 890 3116)
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.

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 details of venue and meeting times.

MARYBOROUGH AMSTRAD USER CLUB

President: Chad Banfield (054 68 1351)
Treasurer: Brendan Severino (054 61 3191)
Secretary: Paul Clark (054 61 2135)
Venue: Maryborough CCC each week on Friday from 12.10 p.m.

MOUNTAIN DISTRICT AMSTRAD USER GROUP

President: Ian Poli (03 758 5282)
Treasurer: Lindsay Bell (03 758 9921)
Secretary: Wayne Darvell (03 221 2182)
Venue: Country Womens Association Hall, 4 Sundew Avenue, Boronia from 7.00 pm. every second Monday of the month.

NORTHERN AMSTRAD USER GROUP

Contact: Brian Ellis (03 469 4425)
Venue: Preston every second Sunday. Contact above for more details.

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 7.30 to 10.30 pm.
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 each month at 3.00 pm.

WESTERN COMPUTER CLUB

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

ACT

CANBERRA AMSTRAD USER'S GROUP

Convenor: 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 Wednesday from 7.00 pm.

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 either 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: Paul Witsen (07 393 4555)
Secretary: John Roberts (07 283 3349)
Treasurer: John O'Connor (07 271 3350)
Librarian: Peter Golledge (07 376 1651)
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: Ron Simkin
Treasurer: Sheila Cole (071 72 3884)
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.

GOLD COAST AMSTRAD USER GROUP
Contact: Mark Abbott (075 31 2114)
Venue: Ashmore Health and Medical Centre, Cotlew St. on the first Saturday of each month at 2.00.
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

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
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: Ken Henry (07 208 8730)
Treasurer: Wayne Stephens (07 287 2459)
Librarian: Brian Moore (07 209 1488)
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.

TOOWOOMBA AMSTRAD USERS GROUP
President: Stephen Gale (076 35 5001)
Vice-Pres: Robert Nisbet (076 35 7025)
Secretary: Malcolm Woodside (076 32 8867)
Treasurer: Peter Fraser (076 34 7032)
Venue: Toowoomba Education Centre, Baker Street, Toowoomba on the 4th Monday of each month.

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: Peter Gibson (004 24 7586)
Treasurer: Philip Reid (004 31 6560)
Secretary: Rick Ferguson (004 31 6280)
Venue: Burnie Technical College, Mooreville Road, Burnie on the third Friday of each month at 6.30 p.m.

NEW ZEALAND

AMSTRAD CANTERBURY
Contact: Christine Linfoot 459 132
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 23.079 Templeton, 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 34 (October 1987) is 21st August 1987

Gossip from the UK

✧ Sales of entertainment software for the CPCs appears to be on the increase, matched only by the decline in sales for the Spectrum and Commodore. Despite the fact that the CPCs have only been around a short time compared with the other two, a recent survey indicated that the CPC market has risen from 10% to 15% over the last year. Some of the credit for the increase is given to the drop in disc prices - £5 to £2.99 - which has encouraged software publishing houses to release their titles on this medium as well as tape.

✧ Across the Channel, Infogrames will soon have two French products called *Passengers on the Wind* and *Murders on the Atlantic*. French software is ever increasing in originality and quality.

Passengers on the Wind is an adventure-type game based on mainland Europe's largest selling cartoon book. It is set in the 18th century. The story centres on a young maiden who attempts to regain her birthright. The game features stunning graphics and pull-down menus.

Murders on the Atlantic is a crime thriller aboard a transatlantic cruiser - that's all the information I have.

✧ There's been an explosion in the release of "compilations". Virgin have released *Now Games 4* consisting of five games in the one pack - *Dan Dare, Back to the Future, Hacker, Mission Omega* and *Jonah Barrington's Squash*.

Beau Jolly have also released some 'five packs'. *Computer Hits 5* contains *2112AD, Wizard's Lair, Contraption, Attack of the killer Tomatoes* and *Kettle* while *Five-Star Games II* has *Cauldron II, Alien Highway, Dandy, Doomsday Blues* and *Frost Byte*.

Durell and Mikro-Gen have entered the compilation fray with selections of their own games. Durell has released the *Big 4 - Combat Lynx, Turbo Esprit, Saboteur* and the previously unreleased *Critical Mass*. Four of the Micro-Gen games make up the Classic Collection No 1. They are *Stainless Steel, Frost Byte, Pyjamarama* and *Battle of the Planets*.

Last but not least is Elite's Hit Pak containing six games and an unreleased bonus title. The seven games are *Scooby Doo, Fighting Warrior, 1942, Sacred Armour of Antiriad, Jet Set Willy, Split Personalities* and *Duet*.

✧ Although the prices of the compilations are extremely attractive and will no doubt please many users, some controversy has bubbled to the surface. In the first instance, some users have complained that they paid full price for some of the titles just six months previously. (Sacred Armour of Antiriad, Frost Byte and Kettle were released about six months ago).

This is also a bone of contention with distributors who can be left with stocks of old games that are practically worthless once they appear on a compilations.

It has been suggested that the industry should agree not to put games on compilations until nine months after their original release to save angering both consumers and distributors.

✧ Psst! want any banned software? You'll have to go to West Germany to get it - although it is generally available in other countries. Confused? Well, a World War II submarine simulation game called Silent Service (from MicroProse) can apparently only be bought from "regulated outlets" such as sex shops within the federal republic. As the package is so accurate, it has been deemed material "thought likely to incite aggressive behaviour" and placed under the Youth Dangerous Publications List along with pornographic products.

✧ For the first time since the UK Gallup Top 40 Software charts were published a business software package has been featured at position 25. *Mini Office II* with 26 man-years of programming development has gained the accolade, all the more significant because the authoritative chart includes all machine formats. Its rise to higher positions is expected now that it is also available for the Commodore market.

✧ While on the subject of success stories, British Telecom, responsible for Rainbird, Firebird and Beyond labels has scooped some 31 awards from reader's polls and journalists' votes throughout Europe and the United States.

✧ Having just launched its *NewStar 4* word processing package for PCs, NewStar Software is continuing a busy schedule with distribution rights to Paperback's *Executive Writer* and *Executive Filer*. It has also announced the release of *Smartkey Plus*. This package allows the user to reconfigure the keyboard to reduce keystrokes and aid control of the operating system.

Newstar also has *Popspell* - a combined 'spell corrector and thesaurus' which is suitable for various word processors.

✧ The latest issue of the revamped Computing with the Amstrad (the English version) has surprisingly dropped from 90 pages including cover in June to 74 in July in spite of the fact that it now incorporates the ex-Newsfield flop Amtix. A quick scan reveals that at least 40% of the pages are devoted to adverts. A local straw poll indicates that they will have to do better than that in a market as big as the UK if they are to avoid going the same way as Amtix.

Rumour has it that another magazine, Amstrad Professional Computing is going to split into two magazines, one catering for the PC and the other for the PCW.

✧ Electric Studio is just about to release (10th July) a new video digitiser for the Amstrad PC. This means that owners of this IBM compatible can input pictures from a video camera or recorder and store them for later tinkering with the software provided with the unit. Features will include the card and software; 768 x 576 pixel resolution; two, four or sixteen shade display and printout; 256 shade data file; 2 to 1 and 3 to 1 zoom, saving to disc in GEM or Postscript format; laser printer driver and on-line print out.

Tas+....
a plus for PCs

Four products imported from the United States by Business Tools Australasia in SA are available for the PC1512. They come under the Tas+ banner and consist of Tas+ Modifiable Accounting; Tas+ Advanced Accounting; Tas+Relational Database and Tas+ Relational Database Developer's Version.

Business Tools Australasia have exclusive Australia and New Zealand rights to the US sourced products, and already ships large volumes to New Zealand.

Tas+ Modifiable Accounting: is a fully integrated package and includes general ledger, accounts receivable, accounts payable and Tas+ Relational Database Developers Version - all for $999. It is fully menu-driven and can be modified for all types of business, product marketers and service organisations.

Tas+ Advanced Accounting: this is a complete accounting system, fully integrated, costing $1429. It contains the same modules as Tas+ Modifiable Accounting plus inventory, sales order entry, purchase order entry and payroll.

If you start with Modifiable Accounting and want to upgrade, you merely pay the difference between the two packages.

Both packages require MS-DOS/PC-DOS 2.0 or higher. Multi-user versions are also available for local area networks that support MS-DOS/Netbios 3.1 standards. You'll also need two 5.25" drives (or a hard disc and a floppy drive) and a minimum of 512k RAM.

Tas+ Relational Database: essentially for a first time user, it provides all the power of a relational database and a 4th Generation language at just $199. You also get a source code editor, a run-time compiler, a screen painter, a program generator, a database browser and a report generator.

Database: Up to 65000 records can be held in one file with an access time of 3 seconds to any record. Can have up to 254 characters per field, 16 files open at one time, 16 key fields per file, and up to 10254 characters per record. Dates are compressed into 4 characters and numeric fields are stored in BCD which means that a 10 digit number uses only 5 characters of storage. dbase III files can be converted to Tas+ and vice-versa.

Procedural Language: can accommodate up to 4500 command lines per program with up to 255 named fields per program. A binary tree file structure allows you to search on any one of 16 key fields per record without sorting. Gosubs or structures may be nested 10 deep. Date and Time arithmetic is supported.

Compiler: Tas+ compiler converts Tas+ applications into fast running pseudocode (executable by Tas+ at run time). It automatically checks a program for syntax errors and command usage problems as it is compiled at the same time compacting the compacting the object program.

Screen Painter: creates a screen that will look identical to your applications. Colour or graphic characters are easily added by making menu choices. Automatically creates programs by 'painting' the screen and allowing Tas+ to write the program.

Source Code Editor: Displays all Tas+ commands in plain English and all options and makes sure you make all the required entries.

Report Writer: creates and runs reports quickly and easily with the option of totalling columns. Up to 10 different fields may be used as a selection criteria in each report.

Other Utilities: Browse utility lets you display 10 records at once, choosing which fields and in which order to display them. Maintain database lets you add, change or delete records in a file.

Tas+ Relational Database Developer's Version: this computer professional's package claims to be 3 times more powerful that dbase III. It offers eveything that the standard Relational Database package has plus a programmer's toolkit giving access to features not normally available on a database/language. You can directly control the memory stack and have up to 17 million records per file with 32 indexes per record. This package also includes a Trace Utility that lets you place break points in your source code. If you are using a standard ASCII editor, the Developer's Version provides a complete cross listing of Tas+ commands, plus a straight forward method of manually writing commands.

Using Tas+ Relational Database Developer's version, BTA have developed their own **Business Tools Australasia Accounting Software** package which is marketed throughout Australia and New Zealand. Currently available in MS-DOS/PC-DOS version, Unix/Zenix versions should be available later this year. **Tas Books** is another home produced accounting based package due shortly from BTA and represents the best performance/value ratio available today. At $612.00 ex-tax, Tas Books gives debtors, creditors, general ledger and inventory modules, lightning fast operation and hundreds of dollars below Sybiz Book Worker and Attache 4. Unlike these products however, Tas Books has much higher limitations at 65000 records.

Rather than rely on maintaining a high profile, BTA prefer to outperform opposition products by fully supporting all their Tas+ products. So sure of their products, they offer a 30-day money back guarantee which means if you don't like the product you send it back. Each product also gets automatic 30-day free support.

Details of Tas+ products can be obtained from:

Nationally	-	Triumph Adler
South Australia	-	John Trafford & Assoc (271 2077)
Victoria	-	Compak Computer Centre (596 7222)
New South Wales	-	Access Business Machines (818 3931)
Tasmania	-	Sleebs Computer Centre (34 6477)
Queensland	-	Greater Brisbane Optical (209 9296)
Rockhampton	-	Xanthos Electronic Products (27 8952)

or BTA themselves on (08) 211 7922 for all the Tas+ products and their own customised Tas+ software products.

News from Amsnet International

Some exciting products have arrived at Amsnet. There's something for everyone this month. Have a look at these . . .

BEAVER: this easy to use, but a very powerful and flexible PCW program with unique features for business users. It can be used solely for Estimating/Quoting, solely for Stock Management or for both simultaneously. You can create and maintain catalogues of parts, materials and labour costs with up to 700 cost elements in each catalogue. You can incorporate the clients name and address into an estimate and then save/print or both. For Stock management, Beaver can quickly locate items, it will build a receipt and save to disc or print. It will print catalogues of stock items by bin or location, adjust re-order levels and produce purchase orders. Beaver sells for $199.00.

EASIBREED: a terrific Farm Management program that allows the farmer to keep a full electronic stud/herd book on up to 500 animals. It will give reports such as - number of animals in the herd, females available for breeding, selling or auction list, family tree of up to 7 generations, relationship between any two animals to avoid cross-breeding, full birth details and weight for age at all critical periods. Easibreed runs on the PC1512 or any IBM compatible and costs just $734.00.

SUPERMIX: from the same stable as Easibreed, this program will let you have four different animal types or poultry and up to 40 different raw foods. With dairy cows, for example, you can monitor cost per cow, milk/cream yield, cost per litre and adjust the rations to get the best output with best nutrient at the lowest cost. It will also monitor growth per day in kilos. Supermix 400 is a more extensive program handling up to 400 raw materials and will take into account future prices of raw materials and will calculate the least cost formula based on those prices. Supermix 400 sells for $1977 and runs on the PCs.

RANDOM DATABASE: from Minerva, at last all the CPC fans have a full random access database. This was not possible before because the ROM Basic does not support R/A files. What's Random Access? It's the ability to quickly and accurately pluck files from your disc in a minimum of time. Unlike sequential files which have to scan through the entire file until it finds the one required, R/A breaks the file into two parts and then again until a match is found. On a file of around 1000 items, R/A will usually find the item in about five comparisons compared with up to 1000 comparisons for sequential files. The other BIG feature is that you are no longer restricted by the size of your memory as R/A Database uses your disc to its maximum. This means that a disc drive 464 is now as powerful as a CPC6128 in file handling. And the screen is fully designed by the user. R/A also means that you can have a full relational database system, again something previously lacking on the CPCs. *(More on Page 12)*

The simple secret of making a fortune

By Alan Sugar

Alan Sugar is one of the British success stories of the 80's. Now 40, the East Ender who left school at 16 and sold car aerials off the back of a van, has built up Amstrad into Britain's most successful computer firm.

It is Sugar's aggressive leadership and marketing genius that have taken his firm to the heights. In an address to London's City University Business School, he revealed his formula for success. This article is based on his speech...

For some reason I've been called a barrow boy. Maybe I'm not smooth enough for everybody, but what does it matter? The turnover of my business has doubled every year since 1980. It was £300 million in the last six months so they can call me what they like.

'Barrow boy', incidentally, I take as a compliment. I would only take exception to it if my ambition in life was to be seen nightly at Annabel's in the celebrated company of Lord and Lady Beeseenwith. It isn't.

My ambition is continued success and growth. I want to give the punter exactly what he wants - that's why he'll buy what I have to sell.

And that's why marketing is just like a stall in Petticoat Lane. No different at all, really. The stallholder is offering his or her wares. The sales pitch may be loud, seeking attention, but is that any different from a high-cost national advertising campaign? Not really, if you think about it, you just reach a wider audience that's all.

My philosophy is all about aggression, energy, realism, instinct, not conforming to the standards that are written down in books, but using innovative ways to cut corners and achieve objectives.

I sometimes break the commercial rules of self-preservation on sheer gut feeling.

Realism

I remember when two Spanish lads who had sold a small number of my computers came back and asked for a big shipment. The only problem was money. This highly undercapitalised pair couldn't pay for them.

The traditional response would have been a thudding door. The financial advice was not to touch them. But I liked them. Frankly, they reminded me of myself - urgent, aggressive and hungry as I was once when I went around selling car aerials from the back of a van.

So I gave them the business and now they are a £100 million Spanish company mopping up 55 per cent of the Spanish computer market with Amstrad. Which is good for them. Good for me. And at the end of the day good for Britain.

My philosophy is all about realism: Swift thinking and decision-making without committees. Pan-Am takes good care of you. Marks and Spencer loves you. Securicor cares. IBM says the customer is King. At Amstrad 'we want your money'.

At Amstrad the staff start early and finish late. Nobody takes lunches - they may get a sandwich slung on their desk - and there's no small talk. It's all action, the atmosphere is amazing and the esprit de corps is terrific. Working hard is fun.

They're a bunch of Alan Sugar clones, learning to work like me. Amstrad bought the Sinclair computer business when I was out of the country and out of touch. The board did just as I would have done if I'd been there.

There's no room for people who are not aggressive and hungry. If they are not that,they're out. Everybody earns their pay every day.

We attract people who either catch on quick or are out in two minutes flat. It's all about swift thinking and decision-making without committees, rising or falling by your own decision or getting out, possessing a sense of urgency to get to the point. There isn't time for waffle.

The longer they are with us the more they assume the corporate identity of the company. With IBM it's blue suit, button-down white shirt. At Amstrad it's confrontational, speedy, incisive, powerful and definitely no bull.

The Japanese use the phrase 'value analysis'. We would call that 'knocking the cost down'.

We pay attention to detail, too. So do the Japanese, of course, but in a different way. When I was still in the design stage of my first computer I went to a Japanese casing manufacturer and told him I wanted white. Immediately he showed me 25 different shades of white.

Indeed, real forceful marketing is coming from the Japanese and the rest of the Far EAst - it certainly isn't coming from the States any more.

The Japanese salesman knows everything about his product. His American counterpart is a relatively empty fellow who can usually be found saying, 'Have a nice day,' moving from job to job either inside his massive corporation or jumping from company to company - there's such a big turnover of staff there. And we swallow all this bull thinking they really are great salesmen. They are not.

Restless

So we at Amstrad follow the Japanese way in creating a corporate personality that sustains all the personnel. It may be brash, but it works. Like the big Japanese companies, we feel like a family.

We encourage a restless spirit in our company. If a product is selling well we instantly begin to examine ways of reducing the cost to make it even more profitable while the product is selling.

And when our managers walk away with a big order from Dixon's (a large UK electrical retail chain) for say a new computer, instead of celebrating they are worrying and thinking about the instruction books, anxiously making sure it is ready and readable. Without it you don't have the whole unit. One screw not ready and you're done for - you haven't made a sale.

If a product isn't selling I don't cut it's selling price to shift it, I'll wipe my face - my East End jargon for getting out. I got out of videos when they weren't selling. I'm back in now.

In the UK we have achieved market shares of up to 70 per cent in some product areas, so the next move is to plant our philosophy in other world markets.

Accent

Frankly, if we got the same market share in all the products we sell now in all the countries in the world, we'd be bigger than General Motors. So we still have a long way to go.

It's highly competitive, just like a bunch of street traders. Whatever your accent, we're all barrow boys in the marketing business. It's a hell of a battle.

There will always be people who are happy to be non-achievers in their life, who get buried in a big corporation or flit from job to job, never growing, never changing their ways.

There are others who are like me. They never expect anything for nothing. They know only how to put their head down and get on with the job.

That's all it is. No great secret. Just intelligence and hard work. Britain is beginning to realise this. When it becomes part of our national philosophy we'll be on top of the world again.

There's no reason why we shouldn't be.

This article was originally published in the Daily Mail (UK) on 2nd May 1987 and has been reproduced with kind permission from the Mail Newspapers PLC.

News from Amsnet International

(continued from Page 10)

TEMPDISC.AMS: a real beauty from Thurston Brown and Associates for the PCW. Tempdisc is a collection of preset Templates on disc. Dozens of very useful layouts for dozens of different uses. It is probably one of the most useful layouts a LocoScript user can buy. We expect it to be around $60 after we have it "Australianised".

THE DESKTOP PUBLISHER: from Database. Amstrad made the PCW the most powerful 8 bit machine in the world, but up to now users had to grind their teeth in envy at the CPC users with programs such as Pagemaker. Now there is the Desktop Publisher to put things right! It comes with full features like pull-down menus, Text Editor, Graphics, Lines, Boxes, Ellipses, Triangles, Fill & Paint, Zoom, a built-in font editor, plus Clip-Art, Extra fonts and a Tutorial. We have only 10 at an introductory price of $275 complete with a high quality mouse, a saving of $50 on the usual price of $325. The program is available for just $125.00.

TRANS-NET: a new networking system for the PC. The basic configuration allows two PCs to be linked together or it can be linked up to 255 PCs. The basic system is completely transparent to the user - hence the name. The options allowed are; Log on to the system; See who else is on the system; Decide which devices you wish to share or grant limited access; Connects you to other shared peripherals; Lets you send messages to other users; Saves the current network configuration to disc; Signs you off; Returns you to the DOS prompt level.
There is a full range of five different software packs such as Net-Bios which emulates the IBM PC-Net. Then there is E-Mail. DMS provides a high level network file handling capability with modules available for all commonly used languages. The last is The Classrooom Monitor where the tutor can communicate with individuals, selected groups or the total class. Trans-Net a low cost way of joining PC's together in various configurations. Prices on application.

All enquiries on the above products should be made to: Amsnet International, PO Box 1319, Southport, QLD 4215 or phone (075) 325464/321465.

Extracts from Games Press Releases

Pneumatic Hammers - Firebird Silver
Strategy/Arcade - Andromeda Software
A lever controlling the pneumatic hammers which are pounding bridge pillars into the riverbed has snapped off. The hammers are now out of control causing rockslides which will eventually destroy the whole research base. Your task in the game is to skillfully control the movements of Red O'Blair in his search for gold, in order to cast a new lever to switch off the power supply.
The level of difficulty can be selected by determining the frequency of hammer strokes, weight of switch lever and average time between crumbles.

Gunstar - Firebird Silver
Shoot 'em up - Software Creations
In the game you are the last hope for civilisation to free the human race from eternal slavery. Your task is to destroy the alien forces which are terrorising the earth, by taking command of the newly developed Gunstar fleet and completing a series of five tasks. Each task gets progressively more difficult and if you are successful in completing a game, the next becomes faster and more furious.

Parabola - Telecomsoft Silver
3 Dimensional arcade - J. Bond
In Parabola your aim in the game is to help poor little 'Bouncing Bruce' escape from the cosmic energy grid. Carefully planned bounces are required to guide Bruce across each square of the grid to reach the exit, collecting the spinning 'energy discs' en route. If Bruce makes a false bounce he could be destroyed by an 'ejection disc'.

Kinetik - Firebird
Arcade Adventure - Jasden Joerges
Kinetik takes you on a quest for the Kinemator, a mysterious entity who lives on a distant screen and waits for a single word to release him from his slumber. Your spaceship travels on waves of pure force, but some areas of this alien world reverse all the laws of natural motion and inertia making travel a strange and obscure experience. The game features 50 screens of crazy colourful action and original gameplay that will keep you on your toes.

Thing bounces Back - Gremlin
Arcade Adventure
In this sequel, the super-fit Thing must bounce round a factory complex (housing a computer auto-producing toys) in his search for software to aid him in his re-programming venture. He can alter the layout of the rooms through which he travels to suit his purposes, and has access to the factory pipe network. Weird and wonderful goblins lurk in every location, anxious to deplete Thing's oil stocks. A quick hand and dogged determination are needed if you're to get anywhere with Thing in his latest escapade.

Samurai Trilogy - Gremlin
Martial Arts
To prove that you are worthy of the coveted title 'Samurai', you must demonstrate your fighting skills and mental agility whilst using three different combat techniques, Karate, Kendo and Samurai. You must emerge victorious over a top level expert in each of the techniques before you are allowed to progress to the next one. Tactics play a big part in this game which if you manage to complete, you are honoured by the inscription of your name on the walls of the Chopemup Temple.

Another slant on Sideways

from Petr Lukes

I was very interested in Mr. Walker's SIDEWAYS program (TAU, June 1987). However, I do not think I could leave the printer running and go to sleep (I distrust anything that moves), so I had a look if the printing could be speeded up.

First of all, his line 1190 does an enormous amount of processing for each character, and this could be minimized by storing the pattern in a string rather than in a numeric array. A string can contain codes 0 to 255, not only the printables. SIDEWAY0 shows the method, and the timings. Sending the pattern as a string requires only one fifth the time needed to assemble each character just before sending it to the printer. This does not mean that the overall printing time will be reduced to one fifth, but it should produce a significant saving in time and leave more memory for text.

Most printers now have an area of memory which can be initialized with custom character patterns (the jargon is to "download characters"). When selected by a command, this set can be accessed in the same manner as the usual set, i.e. by sending the printer just the number of the character rather than the 9 bit matrix. I have used this facility in SIDEWAY1, but at this point you will need to consult your printer manual to see whether your printer can be used in this manner, and if so, what the relevant control code sequences are.

Rather than typing in 94 DATA lines with 8 values each, I used the computer character set. The set is transferred to known memory location (just above HIMEM) by invoking SYMBOL AFTER 33 and can be sent to the printer by PEEK and PRINT#8, after the printer is advised to treat the next 9 bytes as the download character of the given number. The computer characters are stored row-by-row whereas the printer prints column-by-column, so that the characters are already sideways so far as the printer is concerned.

Different printers process the columns differently, so you will need to experiment to get the characters facing the right way (not upside-down or left-to-right). The lack of the 8th bit on the printer port may require the character to be shifted up or down (by c=c/2 or c=c+c). The remarks on the relevant lines should help, but patience will be required.

Some of the copied characters may not be suitable, but they can be replaced by your own design. The downloaded set will remain in the printer memory so long as it is switched on, selected or de-selected by the relevant control sequences, so the set-up program is not needed after initialization. The printer will operate at its normal speed whichever set is selected.

```
10 PRINT"Sideway0 LKS 870605"
20 PRINT"Timing of printing methods for SIDEWAYS, A. Walk
er, TAU Jun 1987"
30 PRINT"Printer stream is sent to a 'null' printer, i.e.
 printer not connected but BUSY line (pin 11) is earthed.
"
40 DEFINT a-z:pr=8:count=1000
50 DIM aln(0,7),al$(0)
60 DATA 0,65,66,67,68,69,70,71,72
70 RESTORE 60:READ b:FOR a=0 TO 7:READ c:aln(b,a)=c:NEXT
a'values into a numeric array, equivalent to original lin
e 1490
80 RESTORE 60:READ b:al$(b)="":FOR a=0 TO 7:READ c:al$(b)
=al$(b)+CHR$(c):NEXT a'values assembled into a string, co
uld be substituted for above
90 PRINT FRE("")"bytes free"'tidy up string space
100 PRINT"Method 1":ti1!=TIME
110 FOR a=1 TO count
120 PRINT#pr,CHR$(aln(0,0))+CHR$(aln(0,1))+CHR$(aln(0,2))
+CHR$(aln(0,3))+CHR$(aln(0,4))+CHR$(aln(0,5))+CHR$(aln(0,
6))+CHR$(aln(0,7))
130 NEXT a
140 ti1!=TIME-ti1!
150 'line 120 is equivalent to original line 1190
160 PRINT"Method 2":ti2!=TIME
170 FOR a=1 TO count
180 PRINT#pr,CHR$(aln(0,0))CHR$(aln(0,1))CHR$(aln(0,2))CH
R$(aln(0,3))CHR$(aln(0,4))CHR$(aln(0,5))CHR$(aln(0,6))CHR
$(aln(0,7))
190 NEXT a
200 ti2!=TIME-ti2!
210 PRINT"Method 3":ti3!=TIME
220 FOR a=1 TO count
230 PRINT#pr,al$(0)
240 NEXT a
250 ti3!=TIME-ti3!
260 'line 230 could be substituted for original line 1190
270 PRINT"Timings for count"count":"ti1! ti2! ti3!
280 PRINT"My timings for count 1000 to null printer : 837
7 8006 1757"
290 END
300 PRINT"Sideway1 LKS 870605"
310 PRINT"Possible speed-up for A. Walker's SIDEWAYS, TAU
```

```
 Jun 1987"
320 PRINT"Copies computer character set to printer RAM"
330 PRINT"Control sequences are for STAR gemini-10x print
er":WIDTH 255
340 DEFINT a-z:symb=33:pr=8:marg=2
350 SYMBOL AFTER symb:me=UNT(HIMEM+1)'Move computer chara
cter set from ROM to RAM so that we can get at it. Each c
haracter is encoded in a matrix of 8 bytes. The computer
matrix is at 90 deg to the normal printer matrix.
360 PRINT"Sending characters to printer"
370 PRINT#pr,CHR$(27)CHR$(66)CHR$(2);'ELITE pitch for 96
char line
380 PRINT#pr,CHR$(27)CHR$(36)CHR$(1);'select RAM set
390 FOR char=symb TO 126:e=me+(char-symb)*8
400 PRINT#pr,CHR$(27)CHR$(42)CHR$(1)CHR$(char)CHR$(0);'in
struct printer to accept character (9 bytes) into RAM
410 FOR b=0 TO 7:c=PEEK(e+7-b)'peek the matrix; some prin
ters may need peek(e+b)
420 c=c'some printers may need c/2 or c=c+c
430 a$=BIN$(c,8):b$="":FOR c=0 TO 7:b$=b$+MID$(a$,8-c,1):
NEXT c:c=VAL("&x"+b$)'mirror image, not needed by some pr
inters
440 PRINT#pr,CHR$(c);:NEXT b:PRINT#pr,CHR$(0);'send proce
ssed bytes to printer
450 PRINT#pr,CHR$(char);'each rotated character to printe
r
460 NEXT char:PRINT#pr,CHR$(27)"@"'reset printer
470 PRINT"The printer is now set up and the programme cou
ld be removed":SYMBOL AFTER 256'reclaim memory used by SY
MBOL
480 PRINT"Demonstration"
490 DIM z$(4)'store for text
500 z$(0)="the quick brown fox jumps over the lazy dog"
510 z$(1)="!"+CHR$(34)+"#$%&'()*+,-./"
520 z$(2)="0123456789"
530 z$(3)=":;<=>?@ [\]^_ `{|}~"
540 z$(4)=UPPER$(z$(0))
550 mx=0:FOR a=0 TO 4:mx=MAX(mx,LEN(z$(a))):PRINT z$(a):N
EXT a'find length of longest string
560 PRINT#pr,CHR$(27)CHR$(36)CHR$(1);'select RAM set
570 PRINT#pr,CHR$(27)CHR$(66)CHR$(2);'ELITE pitch for 96
char line
580 PRINT#pr,CHR$(27)CHR$(48);'line feed 1/8 inch
590 FOR b=1 TO mx:PRINT#pr,SPC(marg);
600 FOR a=4 TO 0 STEP-1:IF b>LEN(z$(a))THEN a$=SPACE$(1)E
LSE a$=MID$(z$(a),b,1)
610 PRINT#pr,a$;:NEXT a:PRINT#pr
620 NEXT b
630 PRINT#pr,CHR$(27)"@"'reset printer
640 END
```

Word Processing on the cheap

by Ian Wallace

Why spend a lot of money on a word processor when a powerful editor is available in ROM, absolutely free? This article presents a simple method of using the editor to prepare text files.

The editor assumes that everything written after a line number is part of a basic program, but as long as you don't RUN it, anything can be written. If you type AUTO or a line number, then start entering text, the first problem you will notice is that as soon as you press ENTER all basic keywords are converted to upper case. This problem is solved by placing an apostrophe at the beginning of each line, so that the line is a REMark.

You can also then use AUTO and type in your text line by line. Use DELETE, EDIT, the COPY key, etc., to add and remove text as required. (The User Instructions Manual explains the editing functions quite well.) The setup program below (Program 1) redefines some of the keys to simplify the text manipulation functions. It changes the large ENTER key to ENTER + ', so that an apostrophe will appear at the beginning of each new line in auto. Because this is the only key change on the main keyboard, typists can type exactly the same keys in exactly the same way as they would on a normal typewriter. If you need the normal ENTER (eg. during editing) use the small ENTER key.]

```
50 'CHEAPWORD Program 1 (I.L.Wallace)
60 'Setup keyboard
70 '
80 KEY 0,"SAVE"+CHR$(34)
90 KEY 140,"LOAD"+CHR$(34)
100 KEY 1,"AUTO "
110 KEY 2,"EDIT "
120 KEY 3,"LIST "
130 KEY 4,"DELETE "
140 KEY 5,"MERGE"+CHR$(34)
150 KEY 6,"LIST#8"
160 KEY 7,CHR$(39)
170 KEY 8,"RENUM"
180 KEY 9,CHR$(34)+",A"
190 KEY DEF 18,1,141
200 KEY 141,CHR$(13)+CHR$(39)
210 MODE 2
220 NEW
```

The numeric keypad is then given some useful commands, all available at a single keystroke, an extra apostrophe (not requiring the SHIFT key) is assigned to the 7, since this character is likely to be used often, and the characters used at the end of an ASCII save command are given to the 9. CTRL-ENTER is changed from RUN" to LOAD", the 80 column mode is invoked, since most printers are of this size, and finally the program destroys itself to make room for your text.

Now you are probably wondering what use are a lot of numbered lines: they could ruin your letters when you are trying to impress someone. The second program solves that problem. After you have completed whatever you are writing and you are reasonably satisfied with it, save it as a text file with SAVE"filename",A (use the numeric keypad 0 and 9 to save keystrokes) and then load it again using the second program.

Choose item 2, Numbered lines, when asked the type of file to load, and the program will remove the line numbers and the apostrophe as each line is input. Program 2 will display, save, print, or merge your files, but it cannot edit them; so always keep a copy of any text in the form of a Basic program so that it can be changed later if necessary.

Both of these programs can be altered and improved if desired to suit your needs. The second program in particular could be given more sophisticated functions, such as some simple editing facilities or the ability to change a normal file back to numbered lines. The program is quite straightforward, with the possible exception of the string manipulation subroutines at lines 670 to 870, which have been extensively REMed. Nevertheless, an explanation of the main variables is given below:

a$()	stores the lines of text
i	line number
n	total number of lines
j	character number, used during searches
w$	stores extra words from too-long lines
l	string length (various strings)
lm	max. line length

Naturally, this article was typed using this method, and I found several disadvantages compared with a word processor. Firstly the editor will not justify lines, so line ends look ragged, but this shouldn't be too difficult to live with. The subroutine at line 770 checks for lines that are too long, cuts them and then adds the extra words to the next line. This makes insertion and deletion of words extremely easy, since Program 2 will automatically make your lines less than the set length, (the variable lm, line 330) as long as they were greater than the set length to begin with. However, you must remember that while lines less than the maximum length are often processed in this way (because words from previous lines are added), there is no guarantee of clean looking text unless all lines except the last in paragraphs are approximately equal to or greater than the set length.

Even when a line is much less than the maximum length, the program might not assume that it is the last line in a paragraph, and you might have to insert a blank line to get it to start a new paragraph I did not find this a problem, but if you do, you could modify the program so that all lines, short or

long, are added together to form the right length unless the next line starts a new paragraph (ie the first character is a space).

When saving or loading, this program sometimes falls foul of the problem where the filename entered becomes corrupted, despite the forced garbage collection in each case. I have found that simply entering CONT after this occurs rectifies the problem. Finally, control codes cannot be embedded in Basic REM statements, but modifying program 2 to insert control codes for your printer should not be too difficult. Despite the above disadvantages, I found Cheapword quite easy to use,with only a few idiosyncrasies that take a matter of minutes to become used to.

As well as the disadvantages, there are even some advantages compared with a word processor such as AMSWORD. The obvious advantage is cost,particularly when the two programs here are quite short and will not take much time to type in. A second advantage is the vastly increased memory available for text. Even AMSWORD users might find program 2 useful for merging two large text files or (tape users particularly) for merging or printing without having to load the long AMSWORD program.

```
100 'CHEAPWORD Program 2  (by Ian Wallace)
110 '(load / save / print)
120 MODE 2:DEFINT a-z:DIM a$(200) 'increase No of lines i
f desired
130 PRINT:PRINT"      CHEAPWORD"
140 PRINT:PRINT"  1. Load file"
150 PRINT"  2. Merge file"
160 PRINT"  3. Save file"
170 PRINT"  4. Print to screen"
180 PRINT"  5. Print to printer"
190 PRINT:PRINT"   (ENTER CHOICE)"
200 a$=INKEY$:i=VAL(a$)
210 ON i GOTO 250,260,470,590,600
220 GOTO 200
230 '
240 'Load and Merge
250 i=-1:GOTO 270 'Load
260 i=n 'Merge
270 PRINT:PRINT"  What type of file?"
280 PRINT"  1. Normal text file"
290 PRINT"  2. Numbered lines (from Basic program)"
300 PRINT:PRINT"   (ENTER CHOICE)"
310 a$=INKEY$:t=VAL(a$):IF t<1 OR t>2 THEN 310
320 w$="" 'extra words from end of lines
330 lm=79 'max chars/line
340 PRINT" Free memory =";FRE("") 'collect garbage
350 INPUT "Enter filename";n$
360 OPENIN n$
370 i=i+1
380 LINE INPUT #9,a$(i)
390 IF t=2 THEN GOSUB 670
400 PRINT a$(i)
410 IF EOF=0 THEN 370
420 CLOSEIN
430 n=i
440 GOTO 900
450 '
460 'Save
470 PRINT" Free memory =";FRE("")
480 INPUT "Enter filename";n$
490 OPENOUT n$
500 FOR i=0 TO n
510 PRINT #9,a$(i)
520 PRINT a$(i)
530 NEXT
540 CLOSEOUT
550 n=i
560 GOTO 900
570 '
580 'Print
590 s=0:GOTO 610 'Screen
600 s=8 'Printer
610 FOR i=0 TO n
620 PRINT#s,a$(i)
630 NEXT
640 GOTO 900
650 '
660 'Remove numbers and ' (rem)
670 j=0
680 j=j+1:IF MID$(a$(i),j,1)<>" " THEN 680  'search for s
pace
690 IF MID$(a$(i),j+1,1)=CHR$(39) THEN j=j+1 'followed by
 ' ?
700 l=LEN(a$(i))-j
710 a$(i)=w$+RIGHT$(a$(i),l) 'remove numbers, add words f
rom previous lines
720 l=LEN(a$(i)):w$=""
730 IF l>lm THEN GOSUB 770 'if line too long
740 RETURN
750 '
760 'Check line length
770 j=lm
780 j=j-1
790 IF j=0 THEN j=lm:GOTO 810 'no spaces found
800 IF MID$(a$(i),j,1)<>" " THEN 780 'search for space
810 w$=RIGHT$(a$(i),l-j)+" " 'store extra words in w$
820 a$(i)=LEFT$(a$(i),j-1) 'new (shortened) line
830 IF LEN(w$)<lm THEN RETURN 'is w$ too long?
840 PRINT a$(i)
850 i=i+1
860 a$(i)=w$:l=LEN(w$) 'repeat process if yes
870 GOTO 770
880 '
890 'Hold
900 PRINT:PRINT"**********  end of file  **********"
910 PRINT" (press any key for menu)"
920 a$=INKEY$:IF a$="" THEN 920
930 GOTO 130
```

A Tricky Program

A guided tour of Basic: how to improve your programming, avoid pitfalls and stay sane.

If you own a microcomputer then at one point or another you will enter a command in Basic. Basic - it's supposed to stand for Beginners' All-purpose Symbolic Instruction Code, but the acronym was obviously thought up first - is the simplest method the home user has for getting the computer to perform a task. It attempts to be straight-forward English and often is easy to follow - but sometimes it's double-Dutch.

This article explains how you can start programming, clarifies a few Basic terms and puts them to good use, tries to banish the dreaded "syntax error" and takes you by the hand through a relatively large and complex listing.

You've bought it. Unpacked it. Run the demo tape. Played games. Used it as a word processor. But what next? Try selling it! But before that final solution, why not have a go at Basic programming? There are numerous advantages and it sounds impressive when you tell Granny Smith that you can program a computer.

Programming develops the way in which your mind works. You start to think logically - you have no choice, for Basic is riddled with ANDs, IFs, NOTs and ORs. Formal logic is sadly a topic that is rushed through in mathematics classes at school. Computers can work only in logic: yes or no, right or wrong, true or false - that is all the computer understands. It is worth understanding this area of maths. It's not only a necessity if you wish to program confidently, but it's also easy.

Neatness, structure and economy will become second nature - that's what programming does for you: it helps you become organised. All the hassle of learning it and struggling with its complexities and difficulties are worth it for the thrill of producing a working program, even if you typed it out from a magazine.

Card trick

Simon Capp is the man responsible for this listing. It appears in this article as it is neat, simple-to-follow and, at the end of the day, is fun to use. It's main function will be to show you how to improve and develop your programming techniques. Card trick is going to be pulled apart and areas of it cordoned off. Each section is going to be explained: what it does, why does it do it, why it is of particular merit, and how it can be improved.

I'm sure the first thing you'll want to know is what does it do? Simon claims it to be "a world first" on the computer. As the title suggests it is a computerised card trick. The initial screen shows several rows of playing cards. You must choose one, taking pains not to reveal its identity to the computer. You've now chosen the card. Tell the computer which row it's in. Your Amstrad will then proceed to shuffle the cards, displaying them in rows when finished. Again you must inform Arnold in which row your card now appears. This will happen several times, after which your CPC will delight in telling you the card you picked.

```
1 ' Card trick
2 ' by Simon Capp
3 ' The Amstrad User Aug 87
```

It's important to keep track of what the program does, when it was written and by whom. There is no better way of doing this than by putting the lot in Remark statements. Placing a REM in a Basic line will cause the computer to ignore the rest of the line. Anything can be placed after a REM except for "|" in Basic 1.0 (one of Locomotive Basic's few bugs: odd things occur). Enough! We are straying. Amstrad Basic has two ways of labelling these comment lines: REM and a simple apostrophe, "'" (which shares a key with "7"). They both do the same job. The only difference is that one takes a third the time to type in.

Remind yourself

If you load a program that you typed in six months ago you likely won't remember what it does, which keys it uses, where you got it from and why you have it. REM it and you won't forget. Don't just put REMs at the start of the program. Label the beginning of subroutines too. That way, you know what's going on.

```
50 GOSUB 130: REM Initialize
60 GOSUB 540: REM Shuffle
70 GOSUB 620: REM Select 21 cards
80 GOSUB 690: REM Ask for row and deal
90 GOSUB 1110: REM Select magic card
100 MODE 1: PAPER 0: PEN 1
110 END
```

A structured program will make extensive use of GOSUBS. A GOSUB tells the computer to execute a subroutine (or another

part of the program if you like) before performing the instructions following it. When it encounters the command RETURN it will go back to where it left off. Why bother using a GOSUB when a GOTO would work and is shorter to type in? True, a GOTO would work. However, if you jumped to this subroutine several times throughout the program and from different line numbers then how would you know which line to return to? You wouldn't! The only thing you could do is to write the routine several times. Wasteful!
Follow Simon's layout when using GOSUBs. Place a REM statement after each one to describe the routine's function. believe me, it saves a lot of headaches in the long run.

```
120 '
130 REM initialize
140 '
150 MODE 0
160 SYMBOL AFTER 239
170 SYMBOL 239,76,82,82,82,82,82,76,0
180 INK 0,0: INK 1,14: INK 2,6: INK 3,26: INK 4,15
190 PAPER 0: PEN 1: BORDER 0
200 GOSUB 350: REM Double-height code
210 RANDOMIZE TIME
220 DEFINT a-z
230 DIM pack$(51),h$(20),temp$(20)
240 FOR s=1 TO 4
250 suit$=CHR$(225+s)
260 RESTORE 330
270 FOR c=0 TO 12
280 READ card
290 pack$(c+((s-1)*13)) = CHR$(card) + suit$
300 NEXT c
310 NEXT s
320 RETURN
330 DATA 65,50,51,52,53,54,55,56,57,239,74,81,75
```

Try to keep the number of commands per line down to a minimum. The listing will look neat and prevent you making mistakes. Most of the lines in Simon's program consist of four or fewer commands.
Computers are incapable of generating a random number. What they produce is pseudo-random. And the way in which they get this pseudo-random number is by working on another number. The number following RANDOMIZE is the seed or number that the computer uses to produce the pseudo-random number. TIME holds the elapsed time since the computer was switched on or reset. Therefore, using RANDOMIZE TIME will ensure a reasonably random number.
DEFINT a-z sets all the variables to integer-type. In other words everything following the decimal point will be discarded.

Hooray for arrays

To allocate memory for numbers or strings you must dimension an array. An array is a collection of data under one name whose members are individually identified by subscripts. When you hear the words "dimension array" it means the amount of space the array will allocate for itself. The command follows this format: DIM name(size or amount of space to reserve). In line 230, Simon has dimensioned pack$(51), which is an array capable of holding 52 elements: arrays start from zero. Array h$ will hold the 21 cards that are finally printed on the screen. I can't prove it yet. Just believe me.
Lines 240 to 310 are responsible for filling up pack$. The command RESTORE 330 tells the computer that the data it should READ starts at line 330. Indeed, if you look at line 330 there is a DATA statement. This holds the values 65, 50,.... If you look at the back of your User Guide you will see a table that converts these numbers into their Ascii equivalents. You'll find these numbers represent A, 1, 2, 3,..., J, Q, K: the cards found in a suit of playing cards. We have now discovered the function of pack$. It holds the values of all the cards in a pack.

```
340 '
350 REM Double-height routine
360 '
370 RESTORE 420:FOR a=&8000 TO &806D: READ a$
380 b=b+VAL("&"+a$)
390 POKE a,VAL("&"+a$):NEXT
400 IF b<>&3670 THEN PRINT "Error in data!": STOP
410 RETURN
420 DATA DD,66,01,DD,6E,00,7E,F5,C1,E5
430 DATA DD,E1,DD,23,DD,66,01,DD,6E,00
440 DATA 2B,C5,23,E5,7E,47,CD,06,B9,F5
450 DATA 78,CD,A5,BB,DD,21,68,80,06,08
460 DATA 7E,DD,77,00,DD,23,DD,77,00,DD
470 DATA 23,23,10,F2,F1,CD,0C,B9,3E,FE
480 DATA 21,68,80,CD,A8,BB,3E,FF,21,70
490 DATA 80,CD,A8,BB,3E,FE,CD,5A,BB,3E
500 DATA 0A,CD,5A,BB,3E,08,CD,5A,BB,3E
510 DATA FF,CD,5A,BB,3E,0B,CD,5A,BB,E1
520 DATA C1,10,AE,C9,00,00,00,00,00,00
```

This routine pokes machine-code numerals into specified memory locations. The command follows this format: POKE address,value. The address can be between 0 and 65535 (in hexadecimal, &FFFF). This amounts to 64k (65536 divided by 1024) - the total amount of space or memory your computer can handle at one go. The value must be between 0 and 255, which is the largest an eight-bit computer can hold: 255 = 2^8 - 1 = &FF hex = 1111 1111 binary.

Basic itself is a large machine-code program. Compared to raw machine-code it is slow: every line must be interpreted and then the relevant action must be taken. Machine-code does not suffer from this. This is what the computer works and thinks in: numbers. Often it is necessary to revert to machine-code to perform a function many times faster than the equivalent Basic routine. Simon's routine prints characters

at double their normal height. Starting at 32768 (&8000), it finishes at 32877 (&806D). You can incorporate this into your own programs. Use lines 340 to 520. To get a character printed in double height follow this procedure: a$ = "This will be printed in double height": CALL &8000,@a$. Simple! Don't worry if you don't understand how or why this works. Master Basic before leaping into code.

```
530 '
540 REM Shuffle cards
550 '
560 FOR f=0 TO 51
570 rn=RND*51
580 t$=pack$(f): pack$(f)=pack$(rn): pack$(rn)=t$
590 NEXT f
600 RETURN
```

As you might gather, this short subroutine shuffles the cards. A loop f is set up to be done 52 times (from 0 to 51) - the number of cards in a pack. A variable rn holds a random value between 0 and 51. This is determined by RND: rn=RND*51 - let rn equal any value between 0 and 51. The card value held in array f is then swapped with the value held in array rn.

```
610 '
620 REM Select hand of 21 cards
630 '
640 FOR f=0 TO 20
650 h$(f)=pack$(f)
660 NEXT f
670 RETURN
```

Pick a card, any card

We have a shuffled pack of cards. We must now select 21 cards. These will, later, be printed on screen. And from there you will have to select one of them. Simon has taken the first 21 values from array pack$ and placed them in h$. You can now see why h$ was dimensioned.

```
680 '
690 REM Ask for row & deal
700 '
710 FOR deal=1 TO 3
720 GOSUB 980: REM deal 21 cards
730 LOCATE 6,16: PEN 4: PAPER 0
740 d$="Which  row ": CALL 32768, @d$
750 LOCATE 2,19
760 d$="is your  card in ?": CALL 32768, @d$
770 LOCATE 8,23: PEN 2: PRINT "1=Top"
780 LOCATE 1,25: PEN 2: PRINT "2=Middle   3=Bottom"
790 CALL &BB03
800 i$=INKEY$: IF i$="" THEN 800
810 IF ASC(i$)<49 OR ASC(i$)>51 THEN 800
820 row=VAL(i$)
830 IF row=1 THEN row=4
840 st=row-2
850 count=0
860 FOR f=1 TO 3
870 FOR c=st TO 20 STEP 3
880 temp$(count)=h$(c)
890 count=count+1
900 NEXT c
910 st=st+1: IF st=3 THEN st=0
920 NEXT f
930 ERASE h$: DIM h$(20)
940 FOR f=0 TO 20: h$(f)=temp$(f): NEXT f
950 NEXT deal
960 RETURN
```

From now on you'll see some real action - things will appear on screen. Line 710 sets up a three-pass loop. This loop (labelled deal) jumps to a subroutine that prints the 21 cards that were chosen (at random) by the computer.

After that, the computer will ask you in which row your card is held: 1= Top, 2 = Middle, 3 = Bottom. To print things in certain positions on the screen, you must use the commands LOCATE and PRINT. Every screen mode has a maximum of 25 lines from top to bottom of the screen. Mode 0 has 20 characters across, Mode 1 has twice this and Mode 2 twice this amount again. Therefore, if you wished the word "position" to appear on row two, line five use: LOCATE 2,5: PRINT "position". The command PRINT will do just that. It will print whatever is held within the quotes, print numerals directly or the value of a variable.

The cards are printed. The computer has now reached line 790: CALL &BB03. This jumps to one of Arnold's native machine-code routines held high in memory. Its function is to clear the input-buffer. In other words if you pressed a key before the computer reached line 790 it would forget it. 664 and 6128 users have a basic command to simulate this call: CLEAR INPUT.

Line 800 is waiting for a keypress. INKEY$ is the command that tests for this: IF INKEY$="a" THEN .... Using INKEY$, you can wait until a certain key has been pressed and then carry out an action accordingly. For example in a game it could test if you wanted to move left or right. Going back to line 800: assuming you haven't hit the keyboard, the computer will keep jumping to line 800 until you do. Once you've pressed a key, your Amstrad checks that it's between 1 and 3. If not, then back to 800 you go.

The secret revealed

After telling the computer in which row your card lies, they will be re-drawn. Once again inform the computer in which row your card appears. This happens three times in all. Each time the computer draws up the cards, in a seemingly random order. Lines 830 to 920 hold the secret to the trick.

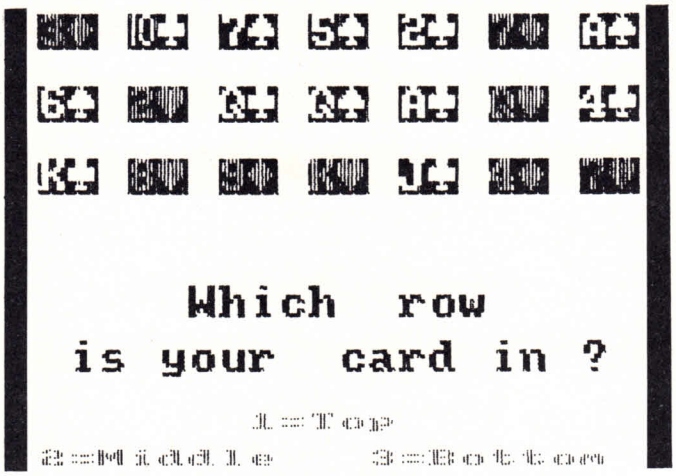
There are three rows of cards. You tell the computer which row the card is in. Suppose it lies in row one. The computer then starts "picking up" the cards two rows down (row three). It then goes to the next row down (four). However, a row four does not exist it jumps to the first row (row one), picks up those, moves to the next row (two) and picks up those. When it lays the cards out the next time, they are placed down in columns not rows. Every time you choose a row, the computer whizzes through this cycle (which happens three times). Assuming you haven't cheated - or more kindly, been incon-

sistent with your card choice - the computer will be able to reveal the chosen card.

```
970 '
980 REM deal 21 cards
990 '
1000 CLS
1010 p=0
1020 FOR across=1 TO 20 STEP 3
1030 FOR down=1 TO 9 STEP 4
1040 IF RIGHT$(h$(p),1) = CHR$(227) OR RIGHT$(h$(p),1) =
CHR$(228) THEN PAPER 3: PEN 2 ELSE PAPER 3:PEN 0
1050 LOCATE across, down: CALL 32768, @h$(p)
1060 p=p+1
1070 NEXT down
1080 NEXT across
1090 RETURN
1100 '
1110 REM Select magic card
1120 '
1130 CLS
1140 LOCATE 4,2:PAPER 0:PEN 2
1150 d$="Your card was":CALL 32768,@D$
1160 IF RIGHT$(h$(10),1) = CHR$(227) OR RIGHT$(h$(10),1)=
CHR$(228) THEN PAPER 3: PEN 2 ELSE PAPER 3: PEN 0
1170 LOCATE 10,7: CALL 32768, @h$(10)
1180 PEN 4: PAPER 0: LOCATE 2,18: PRINT "Another go? (Y/N
)"
1190 i$=INKEY$: IF i$="" THEN 1190
1200 IF UPPER$(i$)="Y" THEN 60
1210 IF UPPER$(i$)="N" THEN RETURN
1220 GOTO 1190
```

It is here that the computer tells you the card you selected. After "picking up" the cards for the fourth time it is guaranteed that the card you chose will be sitting in position eleven of the h$ array. If you don't believe me, do the trick with a real pack of cards. It works every time.

Finally you are asked whether you want another go. Yes and you jump to line 60 where it begins all over again. No and you'll get the Ready sign.



It doesn't work

When things don't work, it is frustrating. Here are some tips that might help.

The best thing to do is split up long lines: one line for one command. You can do this at any colon, ":", except those following an IF command - lines 400, 1040 and 1160. True, the listing may be double the original length, but you'll get a much better picture of what's going on. If an error message now appears, you'll be able to pinpoint it with much more ease.

Some other common pitfalls:

Confusing the letters o or O with the numeral 0, or the small letter l or capital I with the number 1. We do ask authors to avoid using these letters as variables.

Confusing a colon (:) with a semicolon (;).

DATA statements are particular traps for the unwary, because a blunder in one of these will cause an error message when the computer attempts to execute some other line. Watch these points:

Data entries must be separated by commas (,) - have you typed a fullstop (.) by mistake?

Have you typed anything other than the numbers 0 to 9 and A to F? (Lower case a to f are also permitted on the Amstrad). These are the sixteen digits of the hexadecimal system. Data statements should contain nothing else when they are used to define characters or poke machine-code; you can tell their purpose from the READ statement that uses the data.

Don't spend hours and hours on a listing that won't work. Go on to something else. Perhaps later, or even the next day, you can return to it. Chances are the mistake will stick out like a sore thumb.

If all else fails, wait till next month, when we shall look further into debugging.

Gallimaufry V

Two short graphics programs to demonstrate the power of your CPC, and one to dump your screen to a printer.

SERPENT

This routine moves several lines around the screen simultaneously. The listing works only on 664 and 6128 machines. However, if you have a 464, you can remove the last two commas and the 1 from line 60 - you won't get the same effect but it's better than a syntax error.

```
1 'Serpent
2 'T Magee
3 'The Amstrad User Aug 87
10 MODE 1
20 ORIGIN 320,200
30 GOSUB 60:a=a-2
40 IF a>-0.1 THEN GOSUB 60
50 a=a+2.1:GOTO 30
60 MOVE 220*SIN(a/2),98*COS(a),,1
70 DRAW 200*COS(a/2),198*SIN(a)
80 RETURN
```

TRIANGLES

A short and interesting listing which draws a series of triangles in a spiral form - typical of something produced using Logo. Once drawn, you are treated to a tunnel-like effect.

```
1 ' Triangles
2 ' by Dale Clinton
3 ' Amstrad User Aug 87
10 DEG:MODE 1:q=1:CALL &BC02
40 DEG:MODE 1:q=1:CALL &BC02
50 FOR numb=1 TO 440 STEP 5
60 FOR count=1 TO 4:p=count*120
70 c=c+1:IF c>3 THEN c=1
80 x=SIN(numb+p)*numb+320
90 y=COS(numb+p)*numb+200
100 IF count=1 THEN PLOT x,y
110 DRAW x,y,c:NEXT count,numb
120 INK 0,0:BORDER 0:p=0
130 FOR numb=1 TO 300
140 p=p+q:IF p>3 THEN p=1 ELSE IF p<1 THEN p=3
150 INK p,26:FOR count=1 TO 50:NEXT count
160 INK p,0:NEXT numb:q=-q:GOTO 130
```

TEXTDUMP

If you want to dump whatever is on your screen to a printer, this handy listing is just what you need. Type in the program and run it. If everything goes according to plan then the message `New Command: |TDUMP` will appear. Then you can save Textdump as a binary file:

```
SAVE "TEXTDUMP.BIN",B,41000,100
```

To load it again, type the following commands:

```
MEMORY 40999
LOAD"TEXTDUMP.BIN",41000
CALL 41000
```

Alternatively you can save this as a Basic file

```
SAVE "TEXTDUMP"
```

When you have a screen of text, issue the command |TDUMP to send its contents to the printer. If your printer has a dipswitch allowing you to toggle between carriage return (CR) and carriage return and linefeed (CR & LF) set it for CR & LF. On the Amstrad DMP-2000 printer, this is dipswitch 4; move it to the on position.

Tape subscribers will find both the Basic (TDUMPBAS.BAS) and Binary (TDUMPBIN.BIN) on this month's tape.

```
10 'Textdump
20 'by Nigel Magowan
30 'The Amstrad User Aug 87
50 MODE 1
60 MEMORY &A027
70 checksum=0
80 PRINT "Please wait -- POKEing in progress."
90 LOCATE 1,3:PRINT "New Command: "
100 LOCATE 1,5:PRINT "|TDUMP"
110 FOR a=41000 TO 41099
120 READ x$
130 checksum=checksum+VAL("&"+x$)
140 POKE a,VAL("&"+x$)
150 NEXT a
160 IF checksum <>11470 THEN CLS:PRINT "ERROR in DATA":ST
OP
170 CALL 41000
180 LOCATE 1,7:PRINT "RSX set up."
190 DATA 01,36,a0,21,32,a0,cd,d1
200 DATA bc,c9,00,00,00,00,3b,a0
210 DATA c3,42,a0,54,44,55,4d,d0
220 DATA 00,00,cd,11,bc,fe,00,28
230 DATA 36,fe,01,28,36,fe,02,28
240 DATA 36,cd,06,b9,2e,ff,26,ff
250 DATA 3e,0d,cd,74,a0,2c,7d,fe
260 DATA 19,28,28,24,7c,b8,28,ee
270 DATA e5,c5,cd,d6,bd,c1,e1,cd
280 DATA 74,a0,18,ef,cd,2b,bd,d8
290 DATA cd,2e,bd,30,f7,18,f9,06
300 DATA 14,18,ce,06,28,18,ca,06
310 DATA 50,18,c6,c9
```

Hot Tips

The new section strictly for Hackers and the like

A new firmware routine?

I think I have found some previously undocumented entries in the jumpblock (I use a 6128). They are all used by the Basic rom but reside in the Kernel.

The only useful one I have found so far is located at &BD5E (see notes below). It is the routine used by Basic to read lines of input from the keyboard.

The other entries from &BD61 onwards seem to be associated with floating-point arithmetic. For example, the entry at &BD64 converts a number in HL with a sign in A to a 5-byte floating-point number.

Notes:
- the line is input using all the normal editing keys.
- the buffer is a string terminated by CHR$(0). If it has characters in it when the routine is called then the cursor is placed at the start of the line and the current line is echoed to the screen.
- all control codes are echoed to the screen as graphic characters.
- the line is a maximum of 255 characters. When the routine exits the last character will be followed by CHR(0), ready to call the routine again.

202: KL INPUT ??? BD5E
Get a string from the keyboard.
Action: read the keyboard into a buffer until return or escape is pressed.
Entry conditions: HL contains the address of the buffer (must be above #4000)
Exit conditions:
- if the line was terminated by return
  A contains #D
  Carry true
- if the line was terminated by escape
  A contains #FC
  Carry false
- always
  other flags corrupt
  all other registered preserved

Tony Hoyle

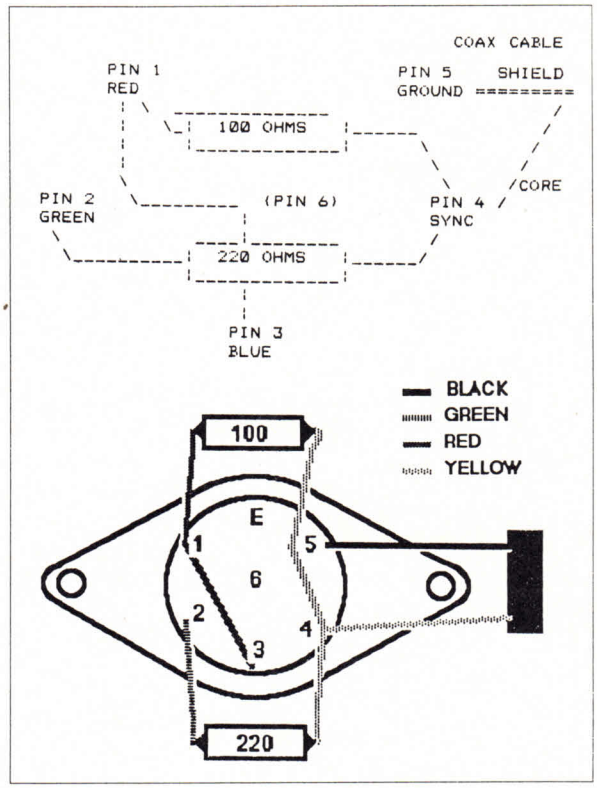
The routine you are describing can be considered as a Basic vector rather than a firmware routine - its actual name is edit. You can be assured that firmware routines will always be in the same place in the 464, 664 and 6128. This isn't true with Basic vectors, which are located in different areas of the three machines - unfortunate.

EDIT, as you correctly state, is located at &BD5E in the 6128. 664 owners can find it at &BD5B and 464 users at &BD3A. You have documented the routine impeccably.
Using this vector can open up some intriguing possibilities. For example, you could redirect the vector to your own routine which checked for certain commands. If it found this command then it could branch off and do its bit, otherwise it could pass the contents of the buffer to the Basic system. The perfect way of adding Basic commands: no need for the " | ".

For Amstrad read Spectrum

This little hardware widget allows you to use the output from a Spectrum and display it on an Amstrad monitor. Follow the instructions carefully and you should have no problems.
First, go and buy a six-pin DIN socket (similar to the one pictured in your manual), and a couple of resistors - 100 ohms and 200 ohms will do. Connect wire from the 'Signal out' of the Spectrum to the 'Signal in' (or aerial socket) of the television set, then cut off the 'Aerial in' socket and carry out the following steps:
1. Remove centre pin (6)
2. Link pins 1 and 3, using one tail of the 100 ohm resistor, keeping the wire close to the plastic.
3. Push the free end of this resistor through the hole in pin 4.
4. Wire the 220 ohm resistor across pins 2 and 4.
5. Connect the core of the coaxial cable to pin 4.
6. Connect the shielding to pin 5.

Bryan Archer



If you can hack your way through to a pen and some paper - write your *Hot Tips* down and send them to The Amstrad User, 1/245 Springvale Raod, Glen Waverley, Vic 3150.

Fleet Street Editor

You too could become a newspaper tycoon - Dave Prakel looks at Desktop Publishing on your PC.

You might think from its title that this is a word processor. It is not. Fleet Street Editor is a 'personal publishing package'. Mirrorsoft already sells a program with the same name for the BBC micro, but the PC-compatible package reviewed here is no relation - it would appear that Fleet Street Editor is more a catch-all title for Mirrorsoft's range of desktop software.

This program is a conversion of the American ClickArt Personal Publisher Package from T/Maker. A word processor (WP) assembles and formats text for output to a printer. The finished result looks a lot like a typed report or letter, and though some WP packages allow you to integrate graphics written in related programs the text is still conventional 'typing'.

Personal Publishing software takes you one stage further into layout, typographical design and illustration. Working on already 'word processed' texts, Fleet Street Editor can design a booklet, letter or report around a range of typefaces and styles, and allows considerable flexibility over layout. It also lets you incorporate pictures from other programs or from a graphics library into your text.

Getting started

Fleet Street Editor has certain hardware requirements. The single disc Amstrad PC cannot run the program but the twin-disc version will. Although the package looks like a GEM program on screen it does not in fact run under GEM Desktop. A mouse is not essential as a NOMOUSE program is supplied which puts the mouse click function (essential for dragging blocks of copy or art about) onto the [f10] key on the standard IBM keyboard, but you have to remember to run NOMOUSE before you run PUBLISH.EXE. That said a mouse certainly makes things easier, despite the good conversion for keyboard.

For output you need a dot-matrix printer supporting IBM graphics. Better image quality will be achieved at a much higher cost with a laser printer - laser printer drivers come in an optional package: the Apple LaserWriter option available soon from ISD and drivers for the Hewlett Packard laserjet are currently available at $259.95. It is expected that the Canon Laserwriter drivers will be available later in the year.

The three discs that make up Fleet Street Editor are not copy protected. You get a system disc, a Font disc and a Graphics Library disc. Installation is straightforward on either hard disc or twin floppy systems.

Printer Installation - adapting the program to the idiosyncrasies of your particular make of printer - is also easy but there are only 28 printer options which may leave some users attempting to write their own printer definition file. Luckily the printer installation software allows you to do just this if you have the time, understanding and patience to plough through your printer manual. Mirrorsoft's Marketing Manager Pat Bitton felt that the existing drivers would cover most people's needs but agreed that some printer manufacturers 'claim the earth for compatibility and leave the software houses to take the consequences'.

The manual supplied gives general tips as well as specific program information. It is presented as three 'Lessons' - Text, Art and Layout - followed by an alphabetically arranged section covering individual features; with Art and Font examples and a good section on planning and producing finished documents.

In use

The first thing you'll meet loading Fleet Street Editor is a tiny clock face. This means 'Wait'. You'll get quite used to looking at this clock as the program does spend a great deal of time in thought.

The top of the opening screen shows the Top Menu, with the Tools and Elevator Bar to the right. The Top Menu consists of a series of pull-down menus which open over the working area and give options which can be selected with the Mouse, the keyboard [Return], or often an Alt-Key combination. The Elevator Bar allows you to move the working window around on the document you are creating - you simply drag the page marker up or down the bar thereby sliding the page 'under' the window.

The Tools are selected with the [f9] key. Initially set to Text entry, Tools also gives you a selection of Graphics Text (text which can be manipulated using the Art functions and which is not subject to the grid layout for the real text), the Art Hand (the pointer/cursor for art manipulation), Outlining (for moving art and wrapping text around art areas), Line drawing (in three widths), plus a free-hand pencil, eraser, and box outline.

Though you can enter text directly on-screen this is not the ideal method - it is also very slow as the screen reformats

between keystrokes. You can use the Insert command to open up existing text which overcomes this problem, but really you are expected to import text from existing ASCII files saved with the TXT extension.

Planning a document

The ideal procedure is to plan your document ahead. Let's assume that text from a word processor is ready, and you know where your graphics elements are coming from. You must next design the grid for the document. This is the most powerful aspect of Fleet Street Editor.

Bringing down the Baselines menu and accepting the [Alt-L] Layout option gives you a decision box (one of many throughout the program). This gives you the opportunity to set up the bare bones of your page: margins, line spacing, even the number of columns. These grids can be stored - a very powerful feature if you need layouts quickly: month by month for two different newsletters for example. At this point you can even choose between cm and inches for your measurements!

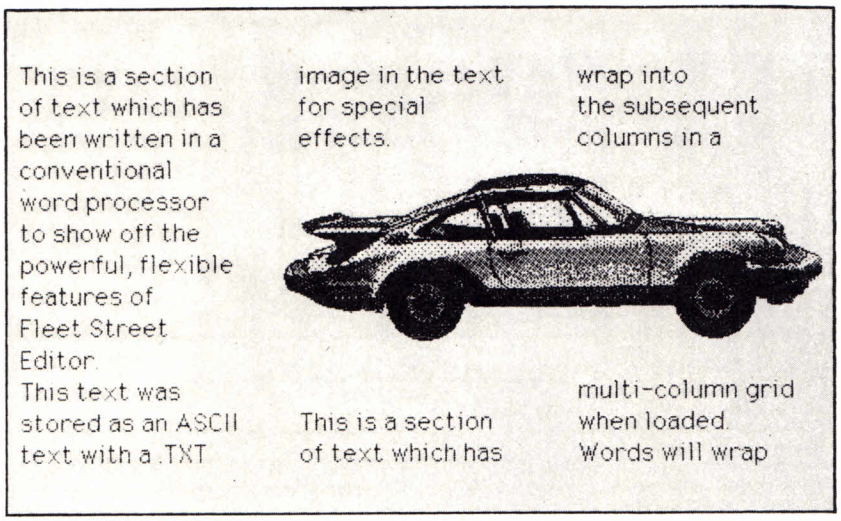
Taking the baselines' Adjust Below option switches on a series of baselines (underlining on the screen that shows where the text will go). Using the mouse to 'click' on one of the lines causes a set of 'handles' to appear which allow you to drag the left and right sides of the text column or to move it vertically so that you can take, say, two lines up for a banner headline. You need some idea at this stage of the size of typeface you intend to use as this will affect the number of lines you can get on the page.

Once satisfied with the baselines layout you use the Get Text option under Edit to read your prepared text file into the clipboard buffer and then to paste it into the existing grid. The Get Text option can only deal with ASCII files with a TXT extension up to 5000 characters in length. Any TXT file loads into the clipboard for later pasting. Alternatively you can label your text with a PUB extension if you want more space and these load text automatically onto the page, but this is a less flexible method.

At this point you can select the required typeface, or change the size or style (Bold, Normal or Italic). The Font and Style options are both pull down menus. Any type size or style not available under a particular typeface comes up in low intensity video and cannot be selected. Using the Edit Paste function you load the text into the grid and the text wraps round and fills up the layout.

Putting in pictures

Adding artwork is as simple. Loading from one of the MAC extension libraries you outline and save the piece of artwork you require. Then use the [f7] Art menu to load the image into the clipboard and the Art hand to slide it around on the document. Once in place you can Paste it in and then, using the



An example showing what can be done using one of the sample pictures supplied with Fleet Street Editor

[f6] Align function,wrap the text around the image. With a small typeface and big artwork this is very impressive.

The Art function allows you to duplicate, even Flip the image both vertically and horizontally or to 'invert' it, making the image negative. You can't change the size of the image within the document (though you could conceivably do this externally using SNAPSHOT).

If you've noticed a spelling mistake in the text you can use the Edit function to change a letter even at this late stage. Editing allows you to cut, copy and paste text. Just as an Art Editor in the real publishing world wouldn't be expected to make massive re-writes of copy, so too Fleet Street Editor is not set up for real word-processing - after all that's what word processors are for.

The finished page can then be seen in miniature using the File [Alt-Z] function. Printing is as simple with the option of Draft or Final quality. Documents of up to 99 pages can be handled but Fleet Street Editor is more at home working with a dozen or so pages. The finished document can be saved with a PUB extension; Fleet Street Editor giving you choice of drive and sub-directory for all such loading and saving functions.

Perhaps the most powerful feature is SNAPSHOT; this is an external program, as is the related SNAP2ART program. SNAPSHOT is a Ram resident utility that will take a screen shot of anything, including other programs, at any time simply by using the [Shift-PrtScr] key (the computer beeps when you press the 'shutter').

The SNAP2ART program develops your pictures and lets you read about half the screen into an ART file for manipulation within the program. Using these simple utilities you can capture graphics from any of your programs and put them where you like in your document. SNAP2ART even allows scaling and magnification.

Good value

You do get a lot for your money. Mirrorsoft's Pat Bitton said on pricing: 'We were worried we had underpriced the package for the PC market and that FSE would be perceived as a toy'. It's certainly no toy, but neither is it full-blown typesetting software. Fleet Street Editor lives in a peculiar no-man's land between integrated packages like GEM Write/Paint and specialised typesetting programs like Wordsmiths' Typefit or Book Machine from Prefis. Unlike Pagemaker on the Apple Mac, Fleet Street Editor cannot, at present, drive 'real' typesetting machines.

Its specification does however fit the needs of many individuals, small businesses, charities, clubs etc. who have the need for specially presented documents at low cost.

The current price of Fleet Street Editor is $499.95. For more information contact your local dealer or ISD on (03) 222 2288

BASIC2 Basics

Gets to grips with Locomotive's powerful version of the BASIC programming language for the PC1512

As a new Amstrad PC owner you may wonder what the disc labelled Basic2 disc is for. If it's your first micro you'll be pleased to know that Basic2 is a very easy-to-use programming language. As a beginner you'll find that many of the horrific tales of obscure commands, of PEEKs and POKEs and of strange numbers before each program line are things of the past as far as this new version of BASIC is concerned. If you've used other BASICs, on earlier Amstrad micros or other makes of computer, you'll be pleasantly surprised by Basic2.

The single most interesting feature of Basic2 is that it runs under the GEM system. It is the first full version of the BASIC language to do this, and allows you to write your own programs which can make use of windows, icons, a mouse and the screen pointer. Basic2 has powerful graphics commands to let you produce a wide variety of graphs, charts or other pictures. It can also deal with files on disc, giving you the opportunity to write programs to handle name and address lists, stock control or simple accounts.

Using Basic2.

You run Basic2 by inserting a copy of the GEM desktop and Basic2 disc, running GEM and selecting the big 'B' icon in the top GEM window. Once Basic2 is loaded it displays three windows, labelled 'Dialogue', 'Edit' and 'Results-I', with a menu bar along the top. If you don't understand any of these terms then have a look at the article on GEM published in Issue 24 - January 1987 of The Amstrad User.

The Edit and Results windows are blank to start with, but the Dialogue window contains the copyright notice and BASIC's 'Ready' prompt. This is the window where you type commands such as SAVE and LOAD to move programs to and from disc, and RUN to set them going.

When you load a program, such as DEMO (supplied on the Basic2 disc), the program itself appears in the Edit window. You can examine it more closely by typing EDIT, followed by hitting the [RETURN] key. The cursor (the small black rectangle) moves to the pointer's position. If you need more room to see the program, click on the diamond in the corner of the window and it will enlarge to fill the screen. Click on the diamond again and the window shrinks back to its previous size. To run the DEMO program, leave the Edit window by pointing anywhere in the Dialogue window and clicking the mouse button. When the cursor returns to the Dialogue window type RUN and the [RETURN] key.

The results of DEMO, the wording and graphics appear in three Results windows: the original 'Results-I' window and two others the program opens. As you can see if you run the DEMO program, you can display text in several different sizes (even upside down), draw shapes very quickly, fill them with many different patterns and calculate the results of complex formulae.

Over to you.

On this page is a simple program in Basic2. Type it in and see what it does. Pull down the Program menu from the menu bar at the top of the screen and select the 'New' option, this clears the DEMO program from the computer. Move to the Edit window and type in each line of the example program printed here. You can type everything in lower case, as Basic2 recognises all its own 'keywords' (such as PRINT, CIRCLE and REPEAT) and automatically changes them to capitals when you press [RETURN] at the end of each line. Once you've typed the program in you can see what it does by returning to the Dialogue window and entering RUN.

You can, of course, ignore Basic2 and run only ready-made applications on your Amstrad PC. But it would be a great shame to ignore the potential of this modern, well designed programming language.

```
CLS
WINDOW OPEN
text$="Welcome to Basic2"
FOR row=2 to 18 STEP 2
PRINT POINTS(20) AT(6;row) text$
NEXT
PRINT AT(10;20)"Press any key to end program"
REPEAT
FOR pattern=1 TO 38
FOR radius=50 TO 1000 STEP 50
CIRCLE 2500;3000, radius FILL WITH pattern
IF INKEY$<>"" THEN STOP
NEXT radius
NEXT pattern
UNTIL FALSE
```

Deskmate accessory and Fas-type typing tutor

by Chris Collins

Welcome once again to Compatible's Corner. Bringing this month's column to you has been fraught with problems. I had intended to bring you a review of PC Write Version 2.70. but as I rewrite this, it has yet to arrive. So this month I have decided to talk to you about two totally different types of programs that are available for your PC1512. These include a typing tutor and a program called Deskmate. I will start with the hardest one first.

DESKMATE

This program is the type that is usually referred to as a desktop accessories program. Deskmate is another shareware program from a company called Alternative Decision Software. The best known program of this type is probably Borland's SideKick. Deskmate can operate in one of two modes. it can be a stand alone program that you call up at the DOS prompt, or it can be memory resident. For it to be memory resident, you simply insert the command DESKMATE /M in your AUTOEXEC.BAT file and Deskmate will be loaded into your computer's memory when you start your system.

It will then remain in memory, just sitting there watching your keyboard and waiting for a certain combination of keys that it recognises. When it sees it's callup combination, it will suspend operation of the program that you are using, you then use Deskmate to do whatever jobs you require, and when you exit Deskmate, control will be returned to the interrupted program, and it will restart from where you left it. At the moment, I have Deskmate running behind Tasword PC.

If I press ALT-M (Deskmate's callup combination), Deskmate will suspend Tasword and popup it's main menu window. This is why programs of this type are often called popup programs.

Deskmate consists of B accessories that are more or less useful whilst using your computer. These include the following; Alarm Clock, Calculator, Calender, DOS Commands, Notepad, Phone Dialler, Printer Control and Typewriter. I will progress slowly through the list and explain each as best I can.

Alarm Clock

This can be accessed from the Deskmate main menu by pressing F2. A small window will popup in the top right corner of the screen. You can set the system time, date, alarm time, whether the clock is on or off, ditto for the alarm and also for the chime. It will display the time on the monitor every 15 seconds, designate a time for the alarm to ring and display a note on the monitor, and it can also execute a program or batch file if you wish. To exit from the alarm clock, press F1 to go back to the main menu or ESC to return to your program. This works with all options.

Calculator

This provides a full function printing calculator which will allow you to read your results back into your interrupted program. Pressing F3 from the main menu will draw a picture of a calculator on the right hand side of the screen. This can be changed to the left hand side if you wish. It provides all the functions that you are likely to require in a calculator including addition, subtraction, division, multiplication and percentages.

Calendar

This function provides a calendar for the current month and a small notepad. Pressing F4 at the main menu will draw two windows on the right side of the screen. On top is the calender, underneath it is the notepad. The calender will show you the month that the system date is set to, haven't set the date correctly, the calender will not be for the current month. You can move the two windows to the opposite side of the screen if you wish. You can also setup a list of dates for your appointments and then save them to disk. If saved with the name CALENDAR.PAD. it will be automatically loaded when you call up the calendar.

DOS

DOS commands are also available from within your program by using F5 at the menu.

All of the usual commands are available including chkdsk, chdir, copy, dir, erase, print, rename and type. I won't explain the commands as the syntax is the same as if you executed the commands from the DOS command line.

The notepad is the next option that is available to you from the main menu. Press F6 and you will be placed in the notepad window. A window will occupy the right hand side of the screen, although it can be moved to the opposite side if you wish. Whilst in the notepad, pressing F2 will bring up a list of help screens that relate to your current commands that you can use. As with any of the options available to you, pressing F1 will get you back to the main menu, and pressing ESC will get you back to your interrupted program. The information in your notepad can be saved to disk, and can be recalled at a later date and printed out. Think of it as a very small word processor. All of the options that offer a notepad (calender, phone dialler, and the notepad itself), are controlled by the same commands.

Phone Dialler

A Hayes compatible modem is required to use this option. If you don't know what a Hayes compatible modem is, I shall explain. The Hayes command set is an American system for controlling modems. It requires that a modem respond in a certain way to a certain command. It is not a standard in Australia, but it is necessary to have a Hayes compatible modem if you wish to use any of the American communication programs that are in the public domain. The Phone Dialler is available by

pressing F7 at the main menu, and opens two windows on the right hand side of the screen. The top window is the phone dialler itself, and the lower window is the ever present notepad. When you call up the phone dialler, Deskmate will load a file called PHONE.PAD if it is present on the current drive. This can contain all the phone numbers that you require. It is possible to setup more than one list of phone numbers. The program can also change the settings on the modem, setting it as either Com1 or Com2, tone or pulse dialling, and also supports prefix dialling for STD and ISD.

Printer Controller

Option F8 is the Printer Controlmenu. This allows you to disable the PrtSc key, setup a test pattern to be sent to your printer, change the printing characteristics, and also change the port that the printer is attached to. The commands work well with either an IBM compatible printer or an Epson compatible printer. The configuration file can be changed to use another printer if you wished.

Typewriter

This option allows your PC1512 and printer to become an easy to use electronic typewriter. Those documents such as envelopes, labels and forms which cannot easily be produced by a word processor, can now by typed on your PC1512 in a manner similar to a typewriter. Press F9 and a window will appear, and clear the top half of the screen. It will display the current ruler line and a list of commands that are available. The window can also be shifted to the bottom of the screen if required.

Other Options

There are only two remaining options on the main menu that I haven't told you about. These are F1 and F10. F1 will return you to your interrupted program and F10 will put you into setup mode. Setup will allow you to change the foreground colour, the background colour and to save and recall the configuration file. Whilst in setup mode, F2 will cycle you through the foreground colours, and F3 will do the same for the background colour. Simply pick the colours that you require. When saving the configuration file to disk, a lot of information about the state of the other accessories is also saved. This is explained in the documentation.

Documentation

This is disc based and runs to a full 60 pages. This requires 66 line paper and can be printed out by simply running a batch file. The documentation is very easy to read and understand. The only thing to remember is that F10 is referred to as F0 in all the documentation and on the screen.

There is also a couple of appendices that deal with the installation program, and errors that are likely to occur, and the best explanation of the shareware principle that I have ever read. All in all, the documentation is very good.

Summary

Deskmate is a program that I myself have started to find a lot of use for. I often wondered what type of person would find any use for a program of this kind with their computer. Having read all the documentation and played around with the program for more than a week, I find a lot of useful options in there. The alarm clock and clock are very useful, if like me you sit at your computer and work, and suddenly find that three hours have passed. I think that when I get my modem, I will find a lot more use for the phone dialler, as I hate dialling phone numbers.

I feel that Alternative Decision Software is deserving of praise for the program that they have created. In comparison with SideKick, the only advantage that SideKick offers is the fact that it operates in many colours. However, Deskmate is for all intents and purposes free, and the best recommendation that I can give is the fact that I will be registering my copy!

FAS-TYPE

The second program that we will look at this month is a typing tutor called Fas-Type. Is produced by a company known as TrendTech Corporation and is specifically designed for the IBM-PC keyboard. Fas-Type is a menu driven interactive typing tutor that lets you progress at your own speed. It is designed for people who cannot type, and also for people that can but who wish to learn the uniquely designed PC keyboard.

Fas-Type has the capabilities to drill you in the home keys, numbers, capital letters and special symbols. These can be mixed in almost any combination. The program operates in medium resolution colour, and is very easy to use. Fas-Type operates on a system of menus to let you know what is happening

The function keys are used to operate the program. They are used to select the part of the program that you are going to use. All of the exercises that you do have help screens available, that will explain what you are supposed to do. Pressing F1 when at any of the major menu will bring up these help screens.

In the beginner's menu, you can select from letters, numbers, special symbols, capital letters only and also all of the above. If you then progress to the beginner's exercises, you will be able to exercise on three letter words, four letter words and phrase groups. As you get better, you then progress to the advanced exercises. This contains easy sentences, difficult sentences and short paragraphs.

The documentation runs to 47 pages, but in some ways I found it rather difficult to read. Fortunately that doesn't really matter, as once you start the program you don't require it. The appendices at the rear of the documentation explain how to printout the help screens, modify the exercise files, list the files on the disk and also explain any program messages that are likely to appear. Also at the rear of the documentation is a glossary of terms that are used throughout the program.

Unfortunately, it is rather difficult to explain the whys and wherefores of a typing tutor. They are a very personal type of program and what suits one will not suit another. However, after saying that, I must say that I found it very easy to use and also very easy to understand. Even after only using the program for a week, I found that my typing was improving.

Well those are the two programs that I was going to look at this month. If you have any questions regarding the PC1512 or any of the programs that are available for it please let me know. The same applies if you wish to see a write-up on a certain type of program.

Tasprint PC
- the style writer

Reviewed by Chris Collins

Well, the Editor has given me a major panic job to review Tasprint PC for this issue of the magazine. So this is necessarily brief but fortunately, because Tasprint PC is used in association with another program, it shouldn't be too much trouble.

Tasprint PC is another memory resident program that allows users of PC compatibles to print documents in differing styles of type. I must be careful with these memory resident programs, otherwise I will run out of memory before I have loaded my application program Tasprint must be loaded before any program that you wish to use with it.

The first thing that you must do, after making a working copy of your disc, is to configure Tasprint to your computer and printer. This can easily be done by running a program called Tpconfig. This will ask you which type of printer you are using. If you are using an Amstrad printer, you will have no trouble at all because the Amstrad printers are on the list. If you don't find your printer on the list, you can create a printer driver to suit. You do this by simply following the prompts and answering the questions.

After you have configured Tasprint, you can load it into memory for use. Tasprint is designed to be used with Tasword, but it can be used with other programs as well. When you load Tasprint, you must tell it where to find it's font files. To keep the font disc in drive B, simply Tasprint as follows;

```
Tasprint b:
```

However, you must have Tasprint.exe and a couple of other files on the current drive.

The fonts that are available to you at any one time number only seventeen. However, on the disk are twenty-five fonts. The fonts that are available can be changed when you load Tasprint by changing the Tpfonts.dat file. When used with Tasword PC,Tasprint also changes a couple of the printing commands. This is all explained in the documentation.

The fonts that are available include the following;
Roman, Ranchero, Scroll, Slimline, Typerite, Anglican, Upright, Heraldic, Upright Bold, Artwork, Palace, Broadway, Median, Supastar, Compacta, Lecture, Outline, Breaker, Block, Datarun, and Cloister.
Also available are four of the above fonts but in an italics mode.

On the disc is a file called Taswrite. This is a direct typing program that will allow you to use your PC1512 and printer as an electric typewriter. Taswrite will not work if Tasprint is loaded into memory and can only be used with one font at a time.

Another of the files on the disc, is a font editor called Tpfed. This will allow you to change existing fonts or to create entirely new ones. It can also be used to create logos. Very powerful.

There are also a couple of demonstration files on the disc. Two of these will print calendar (one for this year and one for next), and a demo text file that demonstrates some of the fonts.

The documentation that comes with Tasprint PC is as always up to Tasman's usual standard. If anything, it is better than normal. This manual also has a list of files that are on disk, and also four appendices. Appendix 1 deals with configuring Tasprint for printers that are not on it's list. Appendix 3 deals with modifications to the Tpfonts.dat file as I explained before. Appendix 4 lists all of the fonts that are available on the disc, gives the pixel width of a character in that font and also shows how each font will look.

All in all, I think Tasprint PC could be a useful program if you have a need for it. It could be used, with Tasword to produce newsletters for a small club or organisation. It would take a lot of organising to set it up correctly, but the effect would be worthwhile. This would be a lot cheaper than buying a publishing program.

The program is very simple to use, the documentation is very good, and the ability to create new fonts means that you should never outgrow the program. If you feel that you have a use for a program like Tasprint PC, by all means go for it!!

CASH TRADER

Designed for non Accountants, you use CASH TRADER like a 140 column Cash Book to maintain complete control over your business, produce Profit & Loss A/c's and Balance Sheets whilst constantly monitoring Bank Balances. With Analyser, you can produce a wide variety of additional 'tailored' reports. Commended for Special Merit by PCW/ Evening Standard as "....an outstanding and original contribution to business profitability and efficiency". Will run on single drive systems.

SHOEBOX

A proven Incomplete Records program in use in many 100's of practices, Shoebox will pay for itself over and over again. Randomly batched documents are entered rapidly with 20 character descriptions and, optionally, automatic extraction. The optional File Expander increases the standard 400 account/2000 entries to 999/32000. Personalised Final Accounts with Notes and Director's Reports can be produced on a number of Word Processing programs which can 'read' data from ShoeBox. Twin Drives required.

QUEST BUSINESS CONTROL SYSTEM

With over 35,000 copies sold, the proven QBCS accounting system will suit every business. The Combined Business Pack has an open-item Sales Ledger, powerful stock control and flexible Invoicing routines - using plain or pre-printed stationery. Separate Purchase & Nominal Ledgers are available. With File Expander up to 9,999 Sales and Purchase Ledger A/cs, 32,000 stock lines and 999 Nominals can be handled.
"For a basic .....easy-to-use program which includes accounting, stock control and invoicing the QBCS will take some beating" Micro Decision June 1986.

CASHIER

A complete Legal Accounting system conforming to Law Society rules with an optional Time Recorder at an 'Amstrad' price. Cashier harnesses the power of the computer to speed up posting, eliminate errors and produce meaningful, up to date information and will handle 650 open-item matters (expandable to 5000). 12 Fee Earners, 16 Office/Client Bank A/cs. 60 Designated Deposit A/cs and 80 Nominal Accounts. Requires twin floppy disk drives or a Hard Disk. $1899

TIME RECORDER

Suitable for all professional offices, to rapidly and simply record Time and Expenses (at 150 rates) over 200 clients. Time Recorder produces detailed reports on Work in Progress, identifies non-productive time and will prove invaluable in ensuring that Clients are invoiced early, accurately and profitably for all work undertaken. File Expander allows up to 9999 clients. Twin drives required.

IT Marketing publish a range of Business programs which rapidly pay for themselves in 'time saved' and improved management information. Our programs come with comprehensive manuals which include key by key examples. Periodic newsletters keep users informed of up-dates and new developments.
Our unique Evaluation Packs, which are actually the real program with a restriction on the number of entries, gives you an unrivalled opportunity to study manuals and use the software with your own data at leisure - once you have completed the evaluation, and paid the balance we'll give you a password to remove the restriction.

Designed for Amstrad PC1512, IBM PC and true IBM compatibles, Apricot PC, Amstrad CPC6128, PCW 8256 and 8512 (Memory disk can be used) and will work with most Epson Compatible Parallel Printers.

IT
INFORMATION
TECHNOLOGY
IT MARKETING

Cash Trader	275
Cash Trader with Analyser	375
Evaluation Cash Trader	80
QBC Combined Business Pack	275
QBC Purchase Ledger	135
QBC Nominal Ledger	135
QBC File Expander	184
Evaluation QBC suite	80
ShoeBox	405
Time Recorder	184
File Expander	184
Evaluation ShoeBox/Time Recorder	80
Cashier - Evaluation Copy	80

Send me more information & review copies ▲
Send me your Dealer Pack (✓)

I enclose my cheque
Please charge my Visa/Bankcard/Mastercard

No _____ Exp. Date _____

Signature _____

Address _____

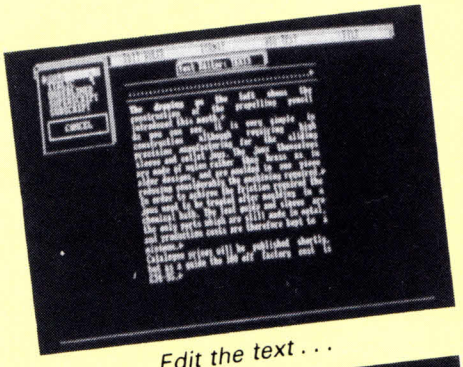
_____

Computer _____

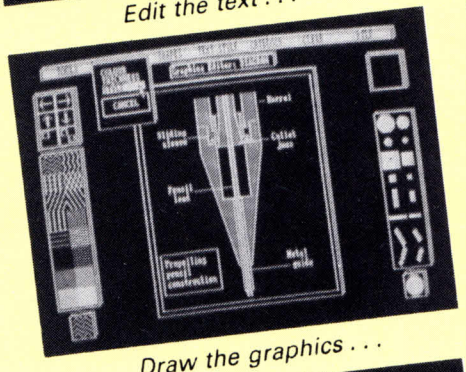
AMSNET INTERNATIONAL PTY LTD
106 Scarborough St. Southport, Q 4215
PO Box 1319 Ph: 075-321465

Make headline news ...on your PCW

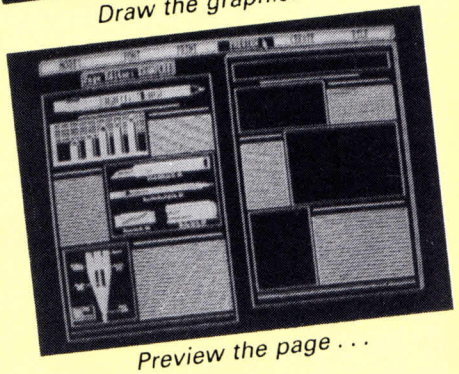
THE revolution which has sent a storm through the printing industry now comes to your computer. With The Desktop Publisher you can unleash the graphics potential of your PCW to design newsletters, flyposters, adverts, letterheads and company reports -then print out the final artwork with headlines, text and graphics exactly where you want them.



Edit the text . . .



Draw the graphics . . .



Preview the page . . .

User friendly

And it's so easy to use. Simply move the pointer around the screen, then click on the option you require using the keyboard or the AMX, Electric Studio or Kempston mouse.

Drop-down menus give you split-second access to all the many facilities available.

How it works

Start by planning your layout: Decide the number and width of columns, and position windows for graphics, captions and headlines.

Then load in your text written in LocoScript, or any other word processor that can output an ASCII file. The built-in text editor can produce perfectly fully-justified text with bold and italic characters shown on the screen just as they will be printed out.

You can design artwork using the feature packed graphics program that it included. You can draw lines, boxes, triangles and ellipses, fill areas or paint with one of 16 patterns, and zoom in for precise pixel editing. There's even a built-in font editor so you can design your own custom character sets.

When you are ready you can preview your masterpiece in miniature, then print it out in top quality mode or use the faster draft mode.

What you get

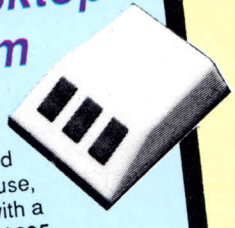
The double-sided disc contains the three main programs, a selection of useful clip art, extra fonts and a tutorial page.

There's also an extremely comprehensive, carefully-written manual with a hands on guide that will ensure you are up and running in no time.

And all for the incredible price of only $125

The finished product — hot off your own printer!

The complete Desktop Publishing System



This package includes The Desktop Publisher software, plus an AMX Mouse and serial interface. It gives you the added flexibility of the precision-engineered Mouse, which plugs into the back of your PCW with a serial interface.

$325

STOP PRESS . . . special launch price of only $275. Offer closes August 30, 1987 or when stocks run out.

The Desktop Publisher

from DATABASE SOFTWARE

AMSNET International

Pty. Ltd. (Aust.) Limited (U.K.)

Australian Microcomputer and Software NETwork

106 Scarborough Street, Southport, QLD 4125

FROM YOUR USUAL DEALER OR DIRECT FROM ORDER HOTLINE: 075-325464 or 075-531734

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.

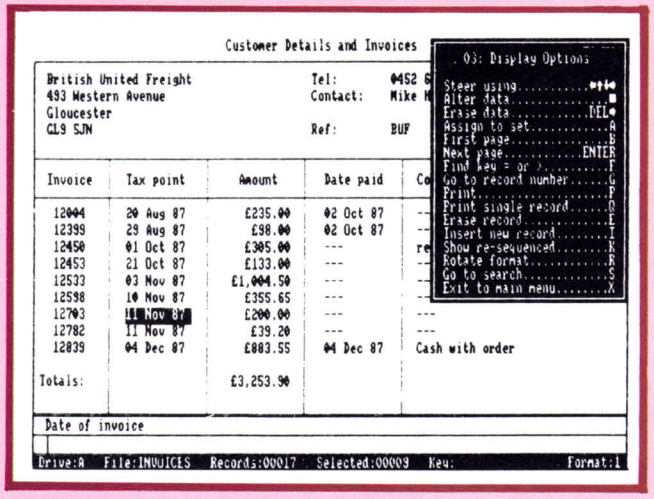
Other products offer a choice between fast but limited capacity RAM files, and large capacity but cumbersome fixed-length, direct access disc files. **MASTERFILE 8000** and the PCW RAM disc combine to offer high capacity with fast access to variable-length data. File capacity is limited only by the size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

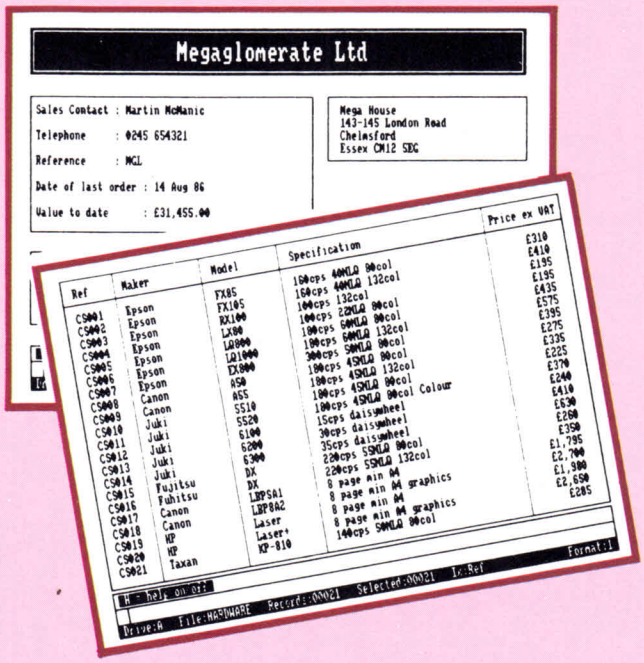


Keyed files are maintained automatically in key sequence, with never any need to sort. You can have unkeyed files too, where records can be inserted at any point in the file.

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:

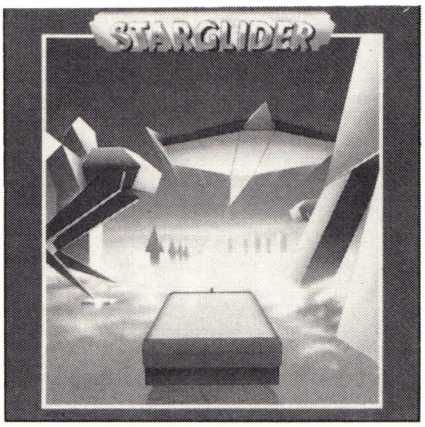
THE AMSTRAD USER
Suite 1/245 Springvale Road,
Glen Waverley,
Victoria 3150 **Tel: (03) 233 9661**

A Shoot 'em up and a Cue 'em up

Two PCW games to hit your dealers soon

STARGLIDER - Rainbird

Isn't it peculiar how planets are always being invaded by psychopathic aliens? Personally, I've never seen an alien, though I have had to look closely at my next door neighbour's children once or twice. In Starglider, the planet Novenia has been invaded by the Eghrons who, as well as having the worst name in science fiction history, are bent on destruction. Well, who isn't these days?



As usual it's your task to sort these psychos out. Consequently, in this vector graphic flight simulation, you assume the role of pilot of an Airborne Ground Attack Vehicle (AGAV). This means that you can battle with both airborne and ground-based enemy craft.

These take a number of forms. Some look like birds of prey, others like the Skywalkers in the Star Wars films, but there are at least another fifteen to contend with. All are well animated vector graphics of the type you might have seen in the Battle Zone tank game in the arcades. Most awesome of all, however, is Starglider One which must be located and destroyed as you fly over Novenia's barren landscape.

Not surprisingly, the AGAV is supplied with three types of weapon: laser, missile and super-missile. The last of these is the most powerful but it is only available when specific missions have been completed. Laser sights can be fixed or movable.

The AGAV can be rearmed by docking with rotating silos. This procedure is very tricky to say the least and requires excellent judgement and timing. Fortunately, the AGAV is very manoeuverable. Once inside, the silo's computer will provide extra information on enemy craft.

The instrument panel includes altimeter, radar scanner, energy level, shield status, velocity indicator and so on. The top of the screen details the AGAV's direction and your score, depending on how many of the enemy you have destroyed.

As with other Rainbird packages, this one includes a novella and a comprehensive playguide, and there is also a keyboard chart. The novella is not just cosmetic - it includes information about the range of enemy craft you have to contend with and gives you precise details on how to dock with silos.

As a shoot 'em up (with a hint of strategy) this game is the best around. Excellent!

STEVE DAVIS SNOOKER - CDS

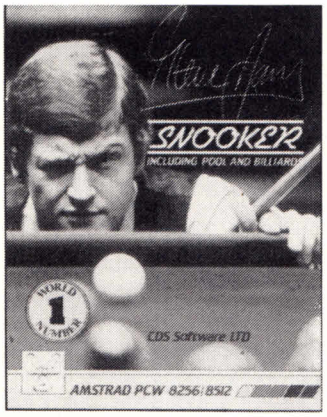
I'm more of an Alex Higgins myself. There's something too clinically perfect about Steve Davis which always infuriates me. Didn't he once say that snooker was more important than sex? Perhaps he's got a point...

The game includes Snooker, Pool and Billiards and all three can be played either alone or with an opponent. They're all accompanied by simple but comprehensive playing instructions, covering the rules of each game and the controls.

In each of the three games, the table, balls and additional features are all neatly presented. The position of the cue ball and strikes are represented by a cross, which is moved by using the cursor keys or, if you prefer, by joystick.

At the bottom of the screen a bar indicates the power of each shot, which can be varied by pressing the appropriate key. There is also the option of putting a spin on the cue ball, a magnified version of which is shown at bottom right of the screen. Again, by moving a little black cross you can choose exactly where you want the cue-tip to hit the cue ball. This enables you to screw back, up or indeed in any direction you like.

The screen also details the number of



visits each player has made to the table, the colour of the ball he is attempting to hit and the break. At the top of the screen is a scoreboard which also indicates any foul shots.

The game's playability is in no way marred by the lack of colour. In the snooker game the reds are clearly distinguishable from the colours (which are numbered in accordance with their value). In pool, 'stripes' and 'spots' are clearly different; in billiards, different shades distinguish the three balls on the table.

As a pool player myself, I found the play realistic, demanding the same precise judgement required in the real game. The movement of the balls is as authentic as one can realistically expect - indeed, they're the roundest balls I've seen in a computer game of this kind.

If you fancy yourself as a snooker ace but haven't room in your garret for a full-sized table, then Steve Davis Snooker should prove a pretty good substitute.

TIP-OFFS

More devious delvings into the PCW to digest

Boxed in

When creating question forms or survey sheets with multiple choice answers, wouldn't it be nice to have a box to tick or cross with each appropriate choice? Here's a method all the way from Wewak in Papua New Guinea.

From within your document:

1. Press f1 - Delete [-] Codes and Effectors and select [+] Rulers Blanks and Spaces. Confirm Single Line Spacing is selected and proportional spacing is not selected. This will allow proper alignment of the top and bottom of the box.
2. To create the box, press the desired number of spaces but at least one space followed by a single stroke of the underline key (shift hyphen) then press return.
Press the space bar until positioned one character under and to the left of the previous character and press EXTRA and [.] together. This produces the | character.
3. Press the underline key followed by EXTRA and [. } together.
4. To check the box alignment, press f1 - Delete [-] Rulers and Blank Spaces.

For a more even looking box try this slightly more complicated method:

1. Insert Layout with Pitch 12 line spacing 1 and Line Pitch of 8
2. Pitch 17 ; ONE space; pitch 10; underline key; pitch 12; return
3. Pitch 17; EXTRA and [.]; pitch 10; underline key; EXTRA and [.]; pitch 17; space; pitch 12

Whitehead Family, PNG

RPED's hidden depths

Users of the BASIC editor, RPED, trying to switch between insert and overwrite mode will discover that the normal '+' key doesn't work. The correct key is the + key to the left of the space bar.
Two unpublicised RPED features are [ALT]+[DEL>], which deletes from the cursor position to the end of the line, and [ALT]+[<DEL], which deletes from the beginning of the line to the character before the cursor.

D. McCallum

Print while you work

When printing a document from LocoScript you can only edit or create another document if it is on the same disc as the one being printed. This can be inconvenient when printing a long document from, say, your Letters disc

Customised print from Protext

Arnor's Protext word processor allows documents to control howthey are printed out by 'stored commands' in the document itself. Any line beginning with a > sign is assumed to be a Protext command rather than part of the document.
A standard 'template' file would therefore contain the stored commands for the line spacing, pitch and so on that you want to use. This has the disadvantage that if you want to do a double spaced draft of a document which is to end up single spaced, you will have to edit the document to alter the stored print commands before printing it, and then undo the changes afterwards. However, because Protext's stored commands are virtually a programming language in their own right, you can write a document file which will allow you greater flexibility. The printout shown is a standard Protext file containing stored commands. The idea is to use the >IN command to merge this file into your document as it prints.
This special file (which is about 1k large) is best stored on your Protext startup disc and copied onto the M drive. If you call it SET.PTR then the existing startup procedure as supplied by Arnor will automatically copy it for you (it copies all files ending in .PTR).
Make sure that the first line of each document contains the line:

```
>IN M:SET.PTR
```

When you print a file with this line in it, you will be asked whether you want to use default values. These are set up in line three of SET.PTR so you could alter them to suit you better. Otherwise you are asked a string of questions to set each option individually - if you press [RETURN] rather than replying properly, then Protext's default value is used. Unless you have reconfigured your Protext (with the CONFIG program), these will be 66 single spaced lines per page, with printing starting at page 1.
You could define a function key to do the printing for you from within an editing session. If you add the command:

```
KEY V " ↑ 254 ↑ pq ↑ 13 ↑ "
```

to your STARTUP file, then whenever you press [EXTRA]+V or [f3/f4], the current document you are working on will print with the new process.

Denzil Millichap

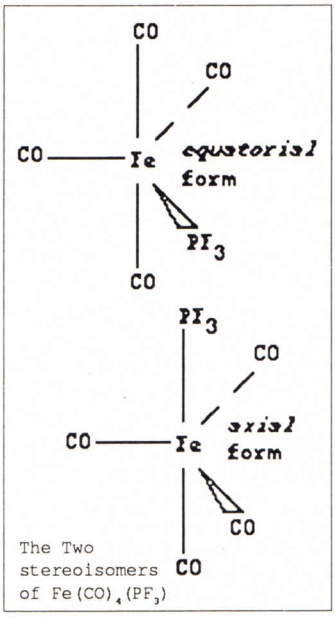
```
>AV "Default values for printout? (Y/N) " def
>IF def="Y"
>SV nc=1 ls=1 sa=1 ea=1000 pl=66 cp="ON"
>EI
>IU cp
>AV "Continuous stationery? (Y/N) " cp
>IF "Y" IN cp
>SV cp="ON"
>EL
>SV cp="OFF"
>EI
>EI
>CP &cp&
>IU nc
>AV "Number of copies? (1) " nc
>EI
>NC &nc&
>IU ls
>AV "Line spacing? (1) " ls
>EI
>LS &ls&
>IU pl
>AV "Page length? (66) " pl
>EI
>PL &pl&
>IU sa
>AV "Start at page? (1) " sa
>EI
>SA &sa&
>IU ea
>AV "End at page? " ea
>EI
>EA &ea&
```

The contents of SET.PTR

when you have a 1000 word report waiting to be typed or edited on your Manuscript disc.
However, by copying the document to be printed onto the M drive (using the [f3] key) you can then change the discs as you like while the document happily prints out from the M drive and you get on with the dreaded report!
Lorraine Forrest

LocoScript Graphics

Did you know that you can use LocoScript to draw simple line diagrams such as the ones shown?
The secret is to realise that the strokes/, | and \ can be in any of eight different pitches (10, 12, 15, 17, 10D, 12D, 15D and 17D, where D means double width), half height and full height. Each different pitch produces strokes at a different angle to the horizontal. For example, the left hand side of the wedge was done in 10 pitch, and the right hand side in 12D pitch.
Similarly, spaces of eight different widths are available. If you want to place a character an exact number of centimetres in from the margin then you can either do an elaborate series of calculations on the character widths or use trial and error and a lot of printouts.
Varying the line spacing (including using zero spacing to overprint) and using subscripts and superscripts opens up a wide range of possibilities. To get a continuous vertical line, you will need to use the | character, which is [EXTRA]+fullstop, and reduce the line spacing to 1/2. For horizontal lines, you can use the underscore character, but you might find hyphens better since they are printed in the middle of a line. You will need to resort to overprinting with more hyphens to make it a continuous line.
It hardly needs to be said that what you see on the screen is not what you get on the paper, and the LocoScript codes need to be hidden otherwise everything becomes a confusing mess. You can store regularly used shapes as phrases.
E. Keith Lloyd



The Two stereoisomers of $Fe(CO)_4(PF_3)$

Undoing SETKEYS

If you use a program which customises the PCW keyboard with SETKEYS, this can be troublesome for other programs. For instance, after using WordStar, the cursor keys have been altered and will no longer work when trying to edit the lines of a BASIC program in RPED.
What you need to do is to restore the original key definitions (which are listed out in the back of the Amstrad manual, on page 112). You need to type the following lines into a file: call it KEYS.CPM for reference. To get the | symbol, type [EXTRA]+ semicolon, and to get \ type [EXTRA]+1/2.
With this all typed in, the command
SETKEYS KEYS.CPM [RETURN]
will restore your keyboard to its virgin state - of course you will need to have the file SETKEYS.COM on the work disc with your KEYS.CPM file.

```
66 N " ↑ C "
02 N S " ↑ Z "
00 N S " ↑ Q "
73 N S " ↑ S "
77 N S " ↑ P "
16 N " ↑ G "
75 N " ↑ H "
10 N " ↑ U "
03 N " ↑ W "
20 N " ↑ ] "
E#8F " ↑ F ↑ B ↑ B "
13 N " ↑ '#8F ' "
E #90 " ↑ F ↑ B "
13 S " ↑ ' #90' "
14 N " ↑ _"
23 N " ↑ V "
15 N " ↑ A "
06 N " ↑ F "
05 N " ↑ F "
01 N " ↑ R "
79 N " ↑ '30' "
16 A " ↑ K "
76 N " ↑ \ "
79 A " ↑ E "
72 A " ↑ X "
```

Getting the most from your M drive

Are you a Fat 8256 owner? Not, of course, anything to do with your girth, but meaning have you installed extra memory chips in your M drive to up it to the 8512 capacity?
The advantage of the extra memory is that you can hold all the files from two sides of a standard A-drive disc on the M disc at once. In the 8256's unexpanded form, you can't even fit all the files from one side into it.
Most people probably have program discs that start up automatically with a PROFILE.SUB. Write yourself a new version, as shown below, which one you use depends on whether all the files you need are on the side A of your startup disc or not.
The version for when the files are not all on side A of your startup disc will PIP all the files on the first side of the disc onto the memory drive, then present you with an asterisk which means you should turn the disc over (or put in another disc) and press [RETURN]. The PCW then runs off the memory to load up the remaining files from the other disc, and runs the program.
Variations on this theme are possible if you don't want to transfer all the files on a disc, just insert the specific file name in place of the *.* of the PIP commands.
"So what?" you say. This means you can now take out

```
SETDEF M:,* [ORDER=(COM,SUB) TEMPORARY=M:]
PIP
<M:=*.*[O]
<
M:
name of program to run, eg. SC2
```
PROFILE.SUB when all files are on the same side of the disc

```
SETDEF M:,* [ORDER=(COM,SUB) TEMPORARY=M:]
PIP
<M:=*.*[O]
M:
PIP M:=A:*.*[O]
name of program to run eg. SC2
```
PROFILE.SUB when all the files are spread over both sides of the disc

the program disc altogether and use the A drive solely for your data discs. You need never swap discs, and it saves wear and tear on the drive. Most importantly,it dramatically speeds up programs that write to and from discs while they work. DR GRAPH is totally rejuvenated on the M drive.
The first line of the PROFILE.SUB is quite cunning. This tells CP/M to look on the M drive for all files before looking on the default drive. Thus, if you have PIP.COM on the M drive, as is quite likely, and you are logged on to A drive (ie. your CP/M prompt is A>), you can run PIP just by typing PIP[RETURN], as opposed to M:PIP[RETURN] as you would normally do.
Matthew Tod

Letter lists

Many people name letters by the date that they were written, as a quick reference guide. For instance, you might call a letter written on 17th May 1987 to A.N. Elk, '170587.ANE'.
LocoScript kindly presents you with a sorted directory of your files. Unfortunately, 011287 (the 1st of December) will be listed before 020187 (the 2nd of January) for obvious reasons. The obvious solution is to write dates YYMMDD rather than DDMMYY.
This way of writing dates is now the international standard, being part of the ISO system of international codes precisely because it is directly computer sortable. Readers in South Africa will be familiar with dates like these as it has been the official standard there since 1974.
LocoScript sorts files by their first letter,in the order

" # $ % ' 0...9 A...Z ...{ }

So, if you want to force some files to be listed on the directory before others, rename them to have a " in front of them.
G.E.B. Russell

Logo listings

Everybody agrees that typing listings without mistakes is a very difficult business, and it is very frustrating to be typing in the eighth line of a Logo procedure when you notice a typing error in line two. You can't get back to correct it, so you either have to start typing again from scratch or use the editor to edit the procedure afterwards.
You can avoid this situation very easily. Whenever you want to type in a procedure, go straight into the editing screen. Type ed [RETURN] with no procedure name, and you will be given a blank edit screen to work on. The normal cursor key movements work, allowing you to type and correct as often as necessary to get it right.
You can define more than one procedure at a time in the edit screen,and when you press [EXIT] you will see Logo confirm each definition as it reads it.
Be sure the name you choose for a procedure doesn't clash with a Logo built-in primitive. In the normal typing mode, Logo will tell you of a name clash as soon as you start defining it, but using the editor you won't find out until after you've typed the whole lot in.
J.A. Coleman

Shampooing your printer

Some printers seem prone to developing problems printing the bottoms of letters, and the descenders of g, j,p,q,y start disappearing.
One solution is to apply a little WD40 lubricant to the printer head to clean it. The printer head can be easily removed by gently prising off the metal retaining plate lying over the flat ribbon cable behind the head assembly. The head is then slipped away from the roller and lifted out. Spray on the WD40, wipe off the excess with a tissue and reassemble the print head.
Ricardo Maragna

Money Manager and disc drives

Those who use the Money Manager package and either own an 8512 or an expanded 8256 may be disappointed to find that Drive B: is not supported by the software. This may be overcome by simple editing of the programs MONEY.BAS and MONEYX.BAS.
Be warned that this modification is made at your own risk, and you can't expect Connect Systems to help you out if things go awry.
Make sure you only modify a copy of the program, on a new disc, and that you keep an original version of the program around to go back to if necessary.
First you have to unprotect the two program files. You load up BASIC and type SAVE "M:TEMP",A
This has saved a blank file to the M disc. Now put the new copy of your Money Manager disc in the drive and type:

```
LOAD "MONEY"
MERGE "M:TEMP"
SAVE "MONEY"
LOAD "MONEYX"
MERGE "M:TEMP"
```

Both MONEY.BAS and MONEYX.BAS may now be edited like any other BASIC program, and you can alter it to access drive B instead of A. Since MONEYX is still in memory, start with that. List out the first few lines with the command LIST - 100 and you will see that line 40 reads OPTION FILES "A". This is the BASIC command that restricts the files to the A drive. To access B instead, change this to B retyping the whole line including its line number:

```
40 OPTION FILES "B"
```

and then save the edited version and load MONEY.BAS into memory by:

```
SAVE "MONEYX"
LOAD "MONEY"
```

You will find on line 130 (list out a few lines to check, with the command LIST 100-150) the same command, OPTION FILES "A". Change this to OPTION FILES "B" as well by retyping it with its line number, and save it back to disc with:

```
SAVE "MONEY"
```

You now have Money Manager configured to read and write its data files from drive B instead of A - this means you will need a separate B-format data disc to use now. The program is still started and run in exactly the same way.
Donald Stuart-Smith

Your Dollar earning Tip-Offs should be sent to:
The Editor
1/245 Springvale Rd,
Glen Waverley,
Vic 3150

Dear Sir or Madam

LocoScript users who don't have LocoMail could be spending a lot of time doing unnecessary retyping. How tired are your fingers?

If you do any kind of business correspondence, from invoicing runs to bulk mailshots, you are bound to benefit from a mailmerge program. For LocoScript users, this comes in the shape of LocoMail. Although it can be used very simply (for mailing lists for instance), LocoMail is in fact surprisingly sophisticated. Following the brief review last month, Ben Taylor looks at a few applications for LocoMail with a quick guide on how to get the most from its lesser known features.

A statistic very popular with pub philosophers is that humans only use around 10% of the processing capacity of their brains. Something of this sort is probably also true of LocoMail - most people who use it only get as far as running off mailshots from an address list. In fact, LocoMail has many powerful features. It can print out invoices for you, calculate discounts, rephrase paragraphs to suit particular friends or enemies, print out multiple copies of LocoScript documents, and much more.

Ask me a question

Let's take a simple mailshot-producing system and add a few bells and whistles to it. Suppose that you have a list of customers and the amount they owe you. For use with LocoMail, you will be storing this data in a document such as the one shown. This contains a greeting for the "Dear ...." part, a statement of the amount due, and the full address.

Producing a simple form file to print out the address, greeting and amount in a letter is straightforward, as per the many examples in the LocoMail manual. The next question you have to think about is how to get the date into the letter. The obvious solution is to use LocoScript to edit the form letter to include the current date, but this has the disadvantage that you would have to re-edit that file every day that you wanted to print out a letter to make sure that the address was right.

The best way is to make LocoMail ask you what the current date is. As you know, LocoMail stores all its working data in 'variables'. Suppose you want to store the date in a variable called, startlingly, 'date', then the way to get LocoMail to ask you what the date should be is to put a line at the start of the document saying:

```
(+Mail)date=? ;Today's date?(-Mail)
```

This says to LocoMail, "when you read this line, stop and wait for the user to type in what the date is." The text after the semicolon is the prompt that is to be used, so you will see displayed in reverse video 'Today's date?' when LocoMail is ready for you to type the date. The date can be entered in any format you like - '17/5/87', or '17th May 1987' - and then press [ENTER] to continue.

When you want to print out the date, you can insert it into the document with the LocoMail command:

```
(+Mail)date(-Mail)
```

This is fine, but has the major problem that when you are doing a run of letters you are asked to type in the date for every single one, even though it hasn't changed. There is a way of telling LocoMail to only ask you the first time, and then to remember and reuse that date in subsequent letters: simply put and exclamation mark between the (+Mail) and the 'date=?'. Don't ask why you use an exclamation mark, you just do, that's all:

```
(+Mail)! date=?(-Mail)
```

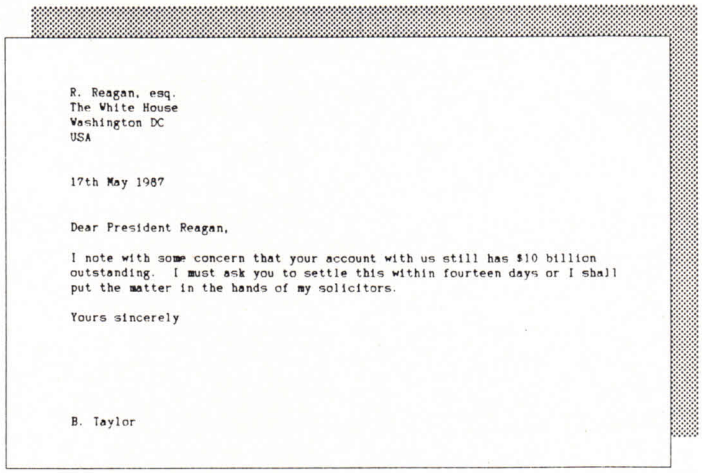
Conditioning yourself

The most powerful feature of LocoMail is undoubtedly its 'conditional printing' capabilities. This means that you can include, omit or modify any of the text in a letter depending on the details of the person you are writing to.

Since LocoMail stores all its details in variables, you can inspect these variables, compare them to other values and take appropriate action. If the date was April 1st, you could print 'April Fool's Day' instead. Carrying on with the debt collection mailing list example, suppose that some people on the list you are on friendly terms with, but relations are a little strained with others.

What you want to do is to phrase the letter differently for each category. A nice chummy letter for your friends, and something a little more menacing for the rest of the population. Who you like and who you don't varies from day to day, so it would be best if LocoMail asked you "is this a friendly letter or not?" on each run through.

What is going to happen is this: when LocoMail starts processing each person's letter, it will ask you "Is this friendly?" and wait for you to type y or n, (meaning yes or no respectively suitably enough). For argument's sake, we'll store this in a variable called 'friendly'. After printing the address and "Dear Fred" greeting normally, we'll then print a brief paragraph



R. Reagan, esq.
The White House
Washington DC
USA

17th May 1987

Dear President Reagan,

I note with some concern that your account with us still has $10 billion outstanding. I must ask you to settle this within fourteen days or I shall put the matter in the hands of my solicitors.

Yours sincerely

B. Taylor

phrased according to whether variable 'friendly' is "y" or "n".

The LocoMail way of saying 'if this is a friendly letter then print a friendly paragraph' is a little obscure, but here goes:

```
(+Mail)# friendly="y" : <(-Mail)
friendly paragraph
(+Mail)>(-Mail)
```

When you read through this, think of '#' as meaning 'if', the ':' as meaning 'then', and the '<' and '>' as brackets enclosing the text to be printed if the condition is true. So, in English, that block of LocoMail commands reads "if the variable 'friendly' contains 'y', then print the following text."

If the #'s,:'s,<'s and >'s frighten you, you could write them into your LocoMail template as IF, THEN, PRINT and END respectively, and use LocoScript's [EXCH] facility to convert them all to the appropriate symbols just before you do the mailshot run. It's best to use upper case for the words inside the Mail command, otherwise you will find that occurrences of the word 'if' in the main body of your letter also get changed, which isn't very useful.

To bring it all together, look at the screen shot for the form letter template. This starts off by asking you for the date, then inserts the name and address of the person (so you can see on the screen who it is you are writing to). It then asks you whether this letter is to be friendly or not. Type y or n at this point. The greeting name is inserted after the "Dear", and then the conditional text paragraphs start.

The first paragraph is printed only if 'friendly' holds "y", and the second paragraph only if it holds "n" - this ensures they never both appear. Notice how two consecutive conditional tests have been used to get this effect, both testing what 'friendly' contains and doing different things.

Finally, the signature block is tailored to fit. Depending on what 'friendly' contains again, it is signed either 'Ben Taylor' for friendly letters, or 'B. Taylor' for more formal ones. Look carefully at where the [RETURN] codes are - [RETURN]s inside the Mail commands are not printed on the final documents, but the ones outside are printed, and could give you unwanted blank lines.

Much ado about nothing

"If you do not immediately pay us $0.00, we will clap you in jail." is the kind of letter all too easily generated with mailmergers. With the form letter as it stands, if someone's account has been paid off and their amount owing is 0, they will still get a letter from you.

There is a special LocoMail command which can save you the embarrassment of sending out this sort of thing. Put as the very first line of the form letter document:

```
(+Mail)# amount="0" : < :
 * : >(-Mail)
```

You can see that this is the same kind of command as the conditional test to include or exclude text, but has an asterisk inside the Mail command where normally you would have ordinary text outside it. It says 'if the amount owing is zero, abandon the current person's data and go onto the next one'. The asterisk tells LocoMail to abandon the current person.

One thing to watch is that when editing your data file with the names, addresses and amounts in it, if you want to set someone's balance to zero you must use just a 0, not $0 or 0¢. $0 is not the same thing to LocoMail as a simple 0.

• *Next month we will finish off Locomail's conditional printing facilities, and look at doing arithmetic inside documents - how you can get it to do all your numbering and discounting for you.*

What is LocoMail?

If you've never heard of a mailmerger, you might be wondering what one does. Very simply, a mailmerger is a program to produce personalised form letters given a list of names and addresses, and a sample of the desired letter. You are bound to have received such mailshots yourself - perhaps from Reader's Digest offering you a guaranteed fortune in their prize draw, or from your credit card company offering you bigger debts.

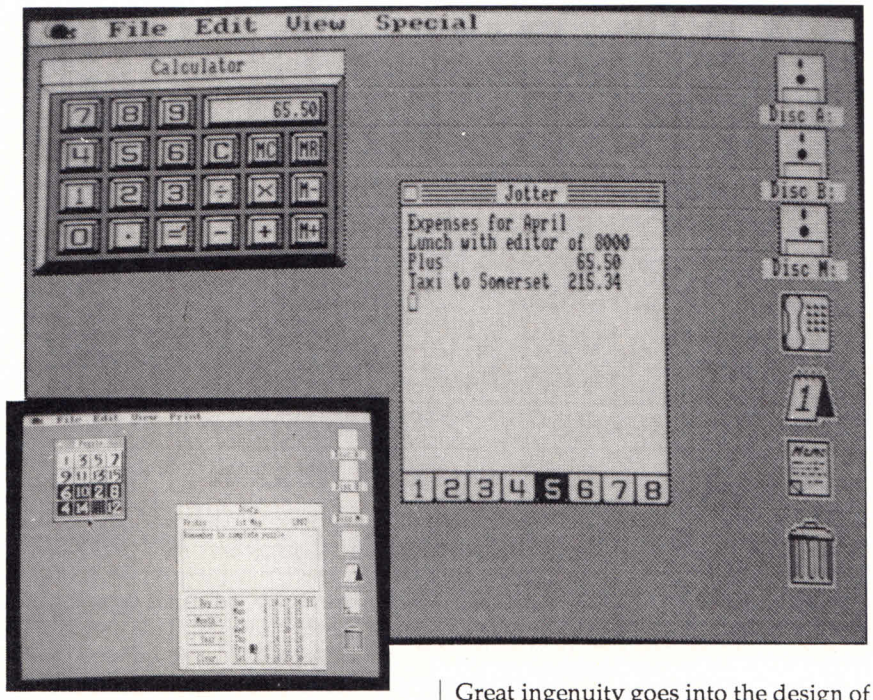
As well as filling in the address and "Dear Mr Snurd" part at the top of the letter, mailmergers can also customise the body of the text. If you have a customer list with a record of what people have bought from you, then you could insert a special paragraph into each person's letter saying "As an owner of our luminous toaster...", or whatever that person's particular last purchase was.

LocoMail, a quite sophisticated program, offers a lot more than this. You can insert numbers into the document, do calculations on them, and even go as far as writing a complete invoicing system.

The Mouse finds a home

A review of the AMX Mouse & Desktop

Is a 'mouse' a real breakthrough for the PCW or is it a rather extravagant extra to clutter up your desk? Last month Advanced Memory Systems launched the AMX desk top package, where the mouse can at last show its paces in the proper 'WIMP' environment. Alex Rae decides whether it could prove a useful pet for the PCW user.



Desk top packages are commodities that always sound good in the adverts. In the world of the 'paperless office' it seems almost blasphemy still to be taking notes on the back of used envelopes or looking up a phone number in a dog-eared address book. But often the practicalities of the average desk top package makes it too bothersome to use regularly.

The new AMX package goes a long way to overcome these problems and presents its services in such a way as to make them attractive and easy to use. The package is an 'organiser' like the Gem desk top for the PC and is a reasonable facsimile of the famous Apple Macintosh set-up.

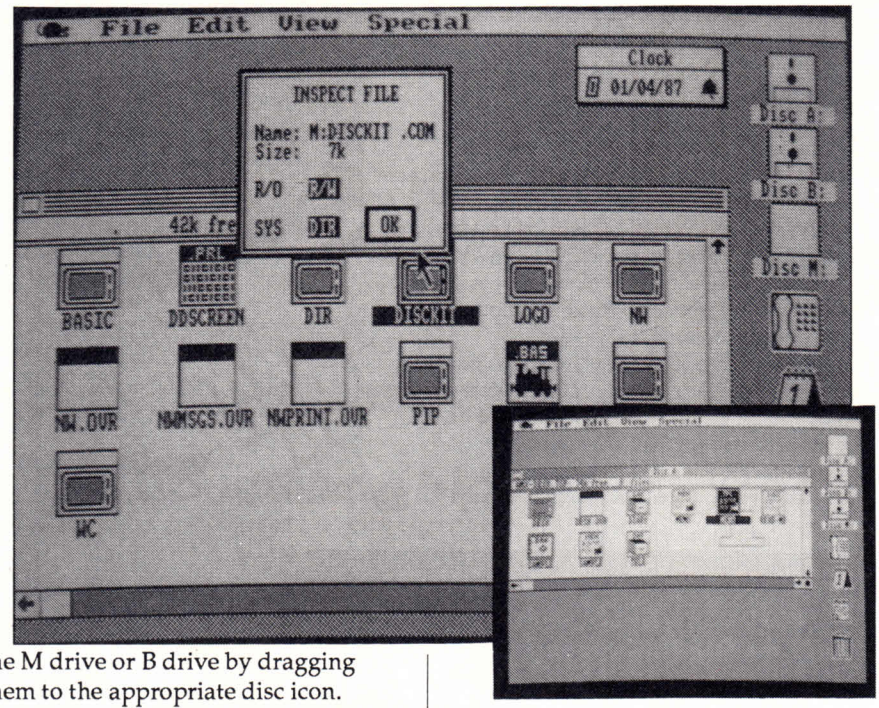
For those of who are wary of using CPM it allows you to move, copy and erase files and run programs without even bothering about whether it is PIPM:=A: or A:+M:. It has a diary, telephone book and a note pad that are real practical propositions and it allows you to create any number of memos (now that is real organisation). You have a clock with an alarm, a calculator that works well and even a puzzle that is annoying enough to keep you going for days. All this without touching your keyboard in a genuine WIMP environment.

Great ingenuity goes into the design of the screen icons, as if it was a serious possibility that a large number of dyslexic people would go out and buy word processors. When you are given the choice of keyboard repeat rate, Desktop shows a snail (for slowness) at one end of the scale and a car (for speed) at the other.

You can view the directory on any drive either in boring old text or by icon. Program files are shown by a little picture of a screen while the text file icon has writing on the front. Most interesting are Mallard Basic files which are each depicted by a little Locomotive.

File it in the bin

Once you display this directory, the world is your oyster. You can 'drag' files around by putting the pointer onto the file, pressing the operating button and pulling the file to the correct place. You can erase files by dragging them into a little bin icon, and move files to



the M drive or B drive by dragging them to the appropriate disc icon.

You can also run a program from any drive by clicking on its icon. Desktop doesn't affect any key assignments you've made with SETKEYS, so you can safely leave your word-processing program in drive M and call it up when you need it.

Desktop takes over 80k of your M drive, an important consideration for 8256 users, but as it carries out all the functions of PIP.COM, DIR.COM and SHOW.COM you don't need these utilities cluttering up the drive.

Here we are again

The neat part of Desktop is that when you leave your chosen program you are automatically taken back to AMX Desk Top, loaded from your M drive. This overcomes the basic problem with 'organisers' - remembering to use them. You still need the AMX disc in the A drive when you want information from your diary or telephone book, but it is still easier and more convenient than many of its competitors. When you eventually leave Desktop you are asked if you want to retain any diary or phone book changes and, if so, you have to replace your Desk Top disc for updating.

This automatic linking back to the AMX program can cause a few minor problems. Should you try to run a text file, the program thoughtfully informs you that you cannot open it (you can only run applications like Disckit, WordStar or SuperCalc). You can, however, print it or 'type' it - display it on the screen as with the TYPE command in CP/M. It does this easily but then immediately returns you to the Desk Top before you have a chance to read it, unless you are quick to press [PTR].

You can't use the pull-down menus from within other programs, but this is often a rather unsuccessful facility in other desk top programs. The note pad only provides limited space, but if you want to write great swathes of text you would probably use your word processor anyway.

These are minor irritations when compared with the potential of the package. It is easy and pleasant to use and does everything that it sets out to do. The telephone book is quick and efficient and has an effective 'Find' facility, that could actually make it quicker to use than a 'manual' telephone book.

Dear diary

The diary, although perhaps limited for space, seems practical enough to use. If there is an entry against a given date it is highlighted on the calendar so you have no excuse for forgetting to look. You can pass back and forward through the months with ease and can use the calendar feature to work out the day your birthday is on up to the year 1999 (sensibly avoiding the leap year problem - Ed). After that you have to guess.

The ultimate feature for all aspiring bureaucrats is a special memo-writing option which neatly lays out all your memos and then stores the result to disc. The editing here is made slightly more complicated by the use of the mouse. Most of the effects you would want are available (paste, cut, clear, justify centre and inserting or deleting lines) but you have to refer to an option at the top of the screen to carry out these simple tasks. You pick out the text or move the cursor around the page with the mouse, and this takes a moment or two to get used to. Still it is

To a mouse

Nowadays Rabbie Burns wouldn't recognise the "wee sleekit courin' timrous beastie" that he spoke of in his poem. The modern mouse has dropped the verminous reputation of his rodent predecessors and is making a home for himself with friendly computer users.

The mouse is now a hand-sized box of plastic with buttons on the front and a long tail of cable joined to an interface which fits into the expansion slot in the back of your PCW. The AMX version comes with its own expansion slot for 'piggy-back' additions. (Amazing the number of animals joining the world of high technology.)

The mouse runs on a metal ball that allows you to move it around your desk (or special 'mouse' mat). As you move the mouse the cursor moves on the screen. When you reach the relevant menu or icon you simply press the button on the mouse to make your choice.

The AMX mouse has three buttons - one for choosing, one for cancelling and one to use on special occasions (like when you are moving a number of files at the same time). Until now there has been little software written to take full advantage of the mouse but the recent launch of various mice and the announcement of a flood of suitable software means that soon no self-respecting PCW freak could afford to be seen without a long-tailed friend.

a facility that no red-blooded executive should live without.

Notes are taken under an option disarmingly called the 'Jotter'. This gives you a potential of five pages of notes and more importantly you can move from page to page easily and quickly. You can even transfer text from one facility to another using the paste routine.

A four function calculator is available and worked either from the mouse or the keyboard - it's quite convenient to use the mouse.

You can set the time and date in a small clock window and perhaps more interesting you can set an alarm that bleeps away merrily, as long as you have Desktop on screen. As you mainly use the program as a link between other programs this is less useful than it might seem at first. It is invaluable, though, if you set your mind to solving the 'simple' sliding block puzzle.

This puzzle is a faithful copy of the trend setting Apple Macintosh one and doesn't seem to be any easier than the original. The manual gives you nine possible solutions (including one marked ominously 'Impossible') which you can work towards until senility sets in or the alarm goes off to point out that you have just wasted two or three hours of your life.

One at a time

The only problem with pulling out the many windows is that you cannot use a window that is overlapped by another one. Every so often you get a message window telling you to close all the others that overlap before you can use a facility.

The best way round this is to adjust the size of the windows, a simple exercise using the mouse. For instance, the directory window can be expanded or contracted at will - you can still use it at its reduced size to find the file you want.

The other possibility is to actually move the windows around the screen to suit your needs or artistic sensibilities. This system is not infallible, though. Try as you like it seems impossible to show the jotter and the 'phone book at the same time.

Verdict

The mouse itself is not a thing of beauty in grey with 'something bright at night' red buttons. The action is not as smooth as some mice but it works well enough. It soon becomes such a natural extension to your hand that you have to think what to do when you eventually have to use the keyboard again.

The program at last provides a desk-top organiser that you would really use to 'organise' your life with the PCW. It would also be very useful to those who are wary of taking the plunge into the twilight world of CP/M.

AMX Mouse and Desktop is imported by Amsnet International and retails for $275.00. For more information ring (075) 325464 or 321465.

CHAIN LETTERS

In his final article, John Hughes ties the strings together and puts them into loops.

Last month we saw it was possible to join string variables together - to concatenate them - by putting a plus sign, '+', in between the variable names, turning "Good" and "morning" into "Good-morning", for example. In this article we shall look at some of the other things which can be done with strings, and the different techniques which Locomotive BASIC offers for handling them.

To begin with, we shall look at the sort of comparisons which BASIC can perform on strings. Program 1 shows how a very simple comparison can be performed; two strings are input and are then compared with each other, and the program prints a suitable message to say whether they match.

The actual comparison is done in Line 60, using a simple IF command, extended with the ELSE keyword, which means 'otherwise' - that is, if the first string is the same as the second string, then the program will print 'They're the same'; otherwise it will print 'They're different'.

Type in the program and try it out to see how it works. There's actually no need to restrict yourself to single words, because a string can be of any length up to 255 characters. After a few tries, you'll realise that in BASIC, strings are 'equal' only if they are precisely the same in every way. The slightest difference, such as an extra space, or even using lower case letters instead of

Program 1

```
10 REM Program to compare strings
20 PRINT "Type in a word"
30 INPUT FIRST$
40 PRINT "Type another word"
50 INPUT SECOND$
60 IF FIRST$=SECOND$ THEN PRINT "They're the same" ELSE
 PRINT "They're different"
70 END


60 IF FIRST$<SECOND$ THEN PRINT "The first string is less
 than the second string"
```

capitals, and BASIC will see them as different.

"TO BE" is less than "to be"

If strings are equal, can they also be 'less than' or 'greater than' each other? The answer is that they can, as you will see if you modify Line 60 of Program 1 to read as follows:

```
60 IF FIRST$<SECOND$ THEN PRINT
"The first string is less than
the second"
```

(You may remember that the symbol '<' means 'less than'.)

Try the modified program out with a few pairs of single letters, such as 'a' and 'A', and 'A' and 'B', and then with pairs of words such as FRED and FREDA. The more you experiment with this, the more you will find out about how BASIC handles strings, so don't be tempted to rush things.

When you have finished your explorations, you will probably have discovered that BASIC has three main rules for comparing characters: basically, capital letters are always 'less than' lower case letters; between pairs of letters which are both capitals or both lower case the earlier the letter comes in the alphabet, the lower its value; and between words such as FRED and FREDA, the shorter word is 'less than' the longer one.

The explanation for all these rules is simple enough: every character is assigned a special number called its ASCII number; ASCII (pronounced 'Askey') stands for the American Standard Code for Information Interchange, and a copy of the code is printed here.

What BASIC really means by a character being 'less' than another character is that its ASCII Code is less than the ASCII Code of the second character. A string is equal to another string if every ASCII Code in each string is the same.

ASC and CHR$

There are two related keywords which are very useful in handling characters; they are ASC (which is obviously an abbreviation for ASCII) and CHR$.

If you look at the listing of the ASCII printable characters, you will see that it was produced by using a FOR-NEXT loop and CHR$. CHR$ is almost always used with PRINT, and the function outputs the character whose ASCII value is put in brackets after the keyword.

For example, PRINT CHR$(65) will put a capital 'A' onto the screen; PRINT CHR$(42) will output an asterisk, and so on. You can output 'non-printable' characters in the same way; PRINT CHR$(7) will make the speaker beep, for example, and PRINT CHR$(13) has the same effect as pressing [RETURN].

Before you experiment too much with this, make sure that you have saved any program you have been working on onto disc, because some of these non-printable characters have special meanings and can cause some odd side-effects.

ASC is used to find out what the ASCII code of any specified character is; it is thus the complement of CHR$. The form of it is PRINT ASC(string) where string is any character or group of characters. What will be printed out is the ASCII Code of the character in the brackets - or value of the first character, if there are more than one.

For example, to find out the ASCII value of '&' you would type PRINT ASC("&") and to find the code of 'A' you would type PRINT ASC("A"). It is very easy to get ASC and CHR$ mixed up with each other, so don't worry too much if you find that you confuse them at first.

Going up to the Capital

We humans treat upper and lower case characters like "b" and "B" as substantially the same and the fact that BASIC does not handle them in this way can be rather a nuisance at times.

We have already seen one example of this sort of problem in Program 4 of the fourth part of this series. You may remember that we asked whether a particular list was to be output on the printer or not, and then tested whether either "y" or "Y" had been pressed. This rather clumsy approach is necessary because a user of the program might press "Y" and have their entry misunderstood if the program was arranged only to accept "y".

Fortunately, BASIC has a pair of commands with which strings can easily be changed into either upper or lower case letters. The two commands are UPPER$ and LOWER$. The first converts a string to capital letters, and the second converts a string to lower case; characters other than letters are not affected by these commands.

The way they work is shown in Program 2; this asks for an input in lower case letters, and promptly converts this into capitals before printing it. Any characters which were in upper case to begin with are unaffected.

Look at the syntax of the UPPER$ command to see how it works - the name of the string to be converted is enclosed in brackets immediately following the keyword UPPER$. LOWER$ is used in exactly the same way, so you should find it easy to rewrite the program to use it.

UPPER$ and LOWER$ are particularly useful in alphabetical sorts - perhaps in a payroll program or perhaps as a preparation to carrying out a spelling check on a document. You can also use them when testing for a response of the

Program 2

```
10 PRINT "Input a string in small letters"
20 INPUT YOUR$
30 MY$=UPPER$(YOUR$)
40 PRINT "You typed ";MY$
50 END
```

Program 3

```
50 ANSWER$=UPPER$(INKEY$): IF ANSWER$="" THEN 50
60 IF ANSWER$="Y" THEN 100
```

Program 4

```
10 REM A Sample Menu
20 REM The Amstrad User
30 CLS$=CHR$(27)+"E"+CHR$(27)+"H"
40 PRINT CLS$
50 FOR POSITION% = 1 TO 10
60 PRINT
70 NEXT
80 PRINT "                              E      Enter Data"
90 PRINT
100 PRINT"                               L      Leave Program"
110 PRINT
120 PRINT
130 PRINT"          Press the appropriate letter to make
 your choice"
140 answer$=UPPER$(INKEY$):IF ANSWER$="" THEN 140
150 IF ANSWER$="E" THEN 200:REM section of program for
 entering data.
160 IF ANSWER$="L" THEN 1000:REM leaving program.
170 GOTO 140
```

'Press 'Y' to print the document' type. Program 3 shows what a segment of such a program might look like.

On the menu

Menus, as you will certainly be aware if you have used LocoScript, are not just semi-fictional descriptions in dubious French of the dishes a restaurant is trying to unload on its clientele. Rather, they are lists of all the options available to you at some point in a computer program.

There are two ways of making a choice from a menu. Either a cursor bar is moved up and down the menu to rest on the desired option (as in LocoScript), or the user is asked to enter a letter or number corresponding to the required choice.

Program 4 shows how a menu might be used as part of a program; don't let the extreme simplicity of this menu - it only has two choices! - blind you to the way it works.

First of all, a FOR-NEXT loop has been used to print ten blank lines in order to position the menu tidily in the middle of the screen. This may seem rather unnecessary, but a tidy screen display is one of the hallmarks of professional programming. In the same way, the two options have been tidily spaced in from the left edge of the screen. And note the prompt at the bottom, so that you are in no doubt what you are meant to do.

Only single-key responses are required, so we have used INKEY$; this simplifies things for the non-expert, who doesn't need to remember to press [RETURN]. If no key is pressed, the program will circle round to Line 140 waiting for a key. When a key is finally pressed, the value of it is converted to upper case by UPPER$; this makes the comparisons in Lines 150 and 160 easier to program.

There is also a 'mug trap' in Line 170 to catch the wise guy who presses neither the "E" nor the "L"; if neither of these keys is pressed, the control goes back to Line 140, and another key press can be made.

This program won't work properly as it stands, simply because Lines 200 and 1000 are missing, so there is nowhere for it to go if "E" or "L" is pressed; but apart from that it is quite OK, and should give you a sensible structure for your own menus.

Incidentally, the Leave Program option could just have an END command in Line 160. However, it makes a program much easier to follow if the END statement is placed at the end of the program.

ASCII nicely

The ASCII Code is one of the few real international standards in computing; big computers sometimes go their own way, but micros like the PCW almost invariably follow it.

Every printable character is assigned its own number between 32 and 127, and numbers below 32 are allotted to 'control characters' such as LF (Line Feed), CR (Carriage Return) and BEL (the Bell). You can get to some of these directly from BASIC; try PRINT CHR$(7) and the Bell (actually the bleeper) will sound; 7 is the ASCII Code for BEL.

Many systems use codes between 127 and 256 for their own purposes; these aren't true ASCII, so they vary between different makes of computer; they are often used for accented characters, Greek letters and the like; even the Pound sign is a non-standard character, because ASCII is an American code, and they never bothered to incorporate foreign currency symbols!

Character 127 is something of a maverick too; many computers print it either blank, a black rectangle or a chequer pattern, but the PCW's print it as a zero, as you can see.

Round Up

In the last five parts of this tutorial we've started from scratch with loading Mallard BASIC and using it as a calculator in 'immediate' mode. Since then we've covered the following topics:

- The BASIC line editor
- The use of variables, and how to display their contents on the screen
- Taking input from the user of your program
- Loading, running and saving a BASIC program
- Remarks, and how to make them unobtrusively
- Branching from one place in a program to another
- The difference between strings of characters and numbers, both integer and real
- Passing through a section of program more than once - looping
- Comparing and manipulating strings
- The ASCII code system for computer characters
- Creating a 'menu' within a program to handle multiple choices

This completes John Hughes' series on Mallard Basic for the PCWs.

Connect Four

A PCW version of this popular game

Adapted by Stephen Reid

This is my version of the popular game played with plastic chips and a rack to drop your chips into one at a time. The object of the game is to get four chips in a row (horizontally, diagonally or vertically) before the other player. You can also block the other person by dropping one of your chips next to, or on top of, his/hers.

Changes

If you are not satisfied with the board size you can increase it by altering the spaces between the inverted commas in lines 340, 400, 520 and 880. Similarly, by reducing the spaces, the board will get smaller. To get the desired height of the board, simply add or take away a ":PRINT" command at the end of line 660.

You can also change the player's symbols by amending C$ (Computer) and H$ (Human) in line 340. To select your new symbol you will have to look up the complete character set on pages 113 - 118 in the last half of book 1 supplies with your PCW, then replace the number in the brackets with one from the decimal value column.

It is also possible to shorten the program by leaving out the lines from line 1210 to the end of the listing, but you would also have to change line 125 to read 125 S$ = CHR$(7). This will remove the introductory screen and message when the computer beats you.

The way the computer moves

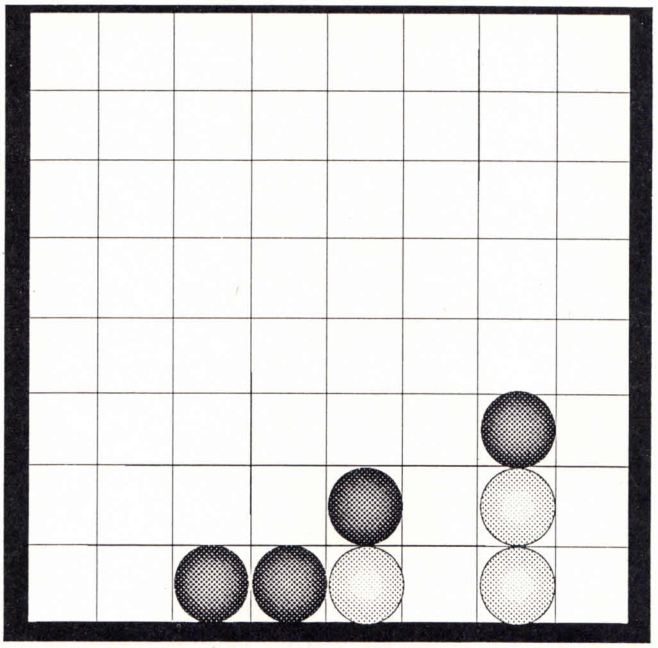
Every move that is made by the computer is the result of it looking at all the possible moves and giving each a value. The move that will stop the human opponent from winning or help the computer to win is the one that will be executed.

Symbols used

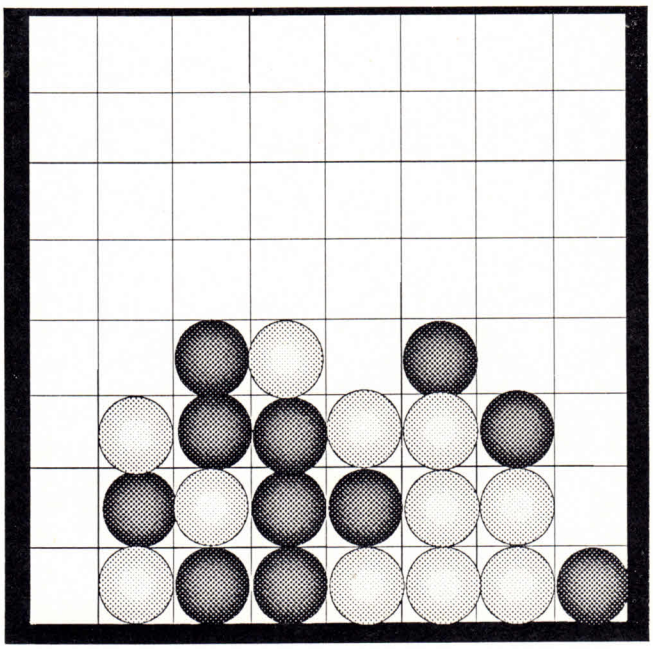
CHR$(143)	Computer's playing symbol
CHR$(159)	Human's playing symbol
CHR$(144)	Represents a blank space on the game board.
s$	[CHR$(7)] is what makes the beep when the screen is cleared, and when the Connect Four name is displayed in the introductory screen.

```
10 REM ************************
20 REM ******* 1/2/87 ********
30 REM ************************
40 REM **** CONNECT FOUR ****
50 REM ************************
60 REM ** BY STEPHEN REID ***
70 REM ************************
80 REM ************************
90 REM ************************
95 DIM A$(10,10),B(10,2)
100 escape$=CHR$(27)
110 home$=escape$+"H"
120 cls$=escape$+"E"+home$
125 GOTO 1210
130 REM ************************
140 REM ******* RULES ********
150 REM ************************
170 PRINT cls$;
180 PRINT:PRINT:PRINT TAB(37)"> > RULES < <"
190 PRINT:PRINT:PRINT " Enter your move as a numbe
r between 1 & 8, enter 0 for a new game, if you wa
nt to see the"
195 PRINT "rules enter 9."
200 PRINT:PRINT:PRINT " The objective of the game
is to get four of your pieces to form a row in any
 direction."
210 PRINT:PRINT:PRINT " Like this."TAB(20)"<"; TAB
(47)"<"; TAB(73)"<"
220 PRINT TAB(20)"<"; TAB(49)"<"; TAB(71)"<"
230 PRINT TAB(20)"<"; TAB(30)"< < < <"; TAB(51)"<"
; TAB(69)"<"
240 PRINT TAB(20)"<"; TAB(53)"<"; TAB(67)"<"
250 PRINT:PRINT:PRINT " If you can win in one move
 you can stop the game and start a new one by ente
ring 10."
260 PRINT "In the original game if you don't see a
 win it isn't counted as a win."
270 :PRINT:PRINT:PRINT " When you make your move,
your piece will appear in the lowest available pos
ition in"
280 PRINT "that column, but if the column is full
you will get another go."
285 PRINT:PRINT:PRINT "To stop at any time press [
STOP].          / / / PLEASE WAIT ) ) )"
290 REM ************************
300 FOR F=1 TO 30000:NEXT F
310 GOTO 460
320 FOR W=1 TO 1000:NEXT W:WIN=0
330 REM CHANGE NEXT LINE FOR YOUR OWN CHOICE OF SY
MBOLS
340 C$=CHR$(143)+"   ":H$=CHR$(159)+"   ": REM *3
SPACES BETWEEN INVERTED COMMAS*
350 FOR F=1 TO 8
```

```
360 B(F,1)=6
370 NEXT F
380 FOR F=1 TO 6
390 FOR G=1 TO 8
400 A$(F,G)=CHR$(144)+"    ": REM #3 SPACES BETWEEN
 INVERTED COMMAS#
410 NEXT G
420 NEXT F
430 REM ************************
440 REM * ACCEPT HUMAN MOVE **
450 REM ************************
460 GOSUB 650
470 PRINT:PRINT "         YOUR MOVE..."
480 INPUT A
485 IF A=9 THEN 170
490 IF A=0 THEN 320
495 IF A=10 THEN PRINT "Congratulations if you hav
e won!": GOTO 320
500 IF A<1 OR A>8 THEN 480
510 L=0
520 IF A$(L+1,A)<>CHR$(144)+"    " OR L=6 THEN 550:
 REM #3 SPACES BETWEEN INVERTED COMMAS#
530 L=L+1
540 GOTO 520
550 IF L=0 THEN 480
560 A$(L,A)=H$
570 B(A,1)=B(A,1)-1
580 GOSUB 650
590 GOSUB 790
600 GOSUB 650
610 GOTO 470
620 REM ************************
630 REM **** PRINT BOARD *****
640 REM ************************
650 PRINT S$;S$;S$;cls$;
660 FOR F=1 TO 6:PRINT
670 FOR G=1 TO 8
680 PRINT A$(F,G);
690 NEXT G
700 PRINT
710 NEXT F
720 PRINT:PRINT "1   2   3   4   5   6   7   8"
730 PRINT
740 IF WIN=1 THEN PRINT "         I HAVE WON": FOR
T = 1 TO 1000: NEXT T: GOTO 1210
750 RETURN
760 REM ************************
770 REM *** COMPUTER MOVES ***
780 REM ************************
790 PRINT "         MY MOVE..."
800 MV=0
810 FOR F=1 TO 8
820 B(F,2)=0
830 NEXT F
840 FOR F=1 TO 8
850 FOR X=-1 TO 1
860 FOR Y=-1 TO 1
870 IF B(F,1)=0 THEN 910
880 IF A$(B(F,1)+X,F+Y)="" OR A$(B(F,1)+X,F+Y)=CHR
$(144)+"    " THEN 910: REM #3 SPACES BETWEEN INVER
TED COMMAS#
890 IF A$(B(F,1)+X,F+Y)=H$ THEN GOSUB 1040
900 IF A$(B(F,1)+X,F+Y)=C$ THEN GOSUB 1140
910 NEXT Y
920 NEXT X
930 NEXT F
940 P=0
950 FOR F=1 TO 8
960 IF B(F,2)>P THEN P=B(F,2):N=F
970 NEXT F
980 A$(B(N,1),N)=C$
990 B(N,1)=B(N,1)-1
1000 N=0
1010 P=0
1020 RETURN
1030 REM ************************
1040 MV=2
1050 M1=MV
1060 IF A$(B(F,1)+(X#2),F+(Y#2))=H$ THEN MV=MV+10
1065 IF A$(B(F,1)-X,F-Y)=C$ THEN MV=MV+20
1070 IF A$(B(F,1)-X,F-Y)=H$ THEN MV=MV+20
1080 IF MV<>M1+10 THEN 1100
1090 IF A$(B(F,1)+(X#3),F+(Y#3))=H$ THEN MV=MV+100
0
1100 B(F,2)=B(F,2)+MV
1110 M1=0
1120 RETURN
1130 REM ************************
1140 MV=2
1150 M1=MV
1160 IF A$(B(F,1)+(X#2),F+(Y#2))=C$ THEN MV=MV+9
1170 IF MV<>M1+9 THEN 1190
1180 IF A$(B(F,1)+(X#3),F+(Y#3))=C$ THEN MV=MV+200
0:WIN=1
1190 B(F,2)=B(F,2)+MV
1200 RETURN
1210 REM ************************
1220 REM **** FIRST SCREEN ****
1230 REM ************************
1235 S$ = CHR$(7)
1240 REM ***** R *****
1250 R1$ = CHR$(134)+CHR$(138)+CHR$(138)+CHR$(138)
+CHR$(140)+" "
1260 R2$ = CHR$(133)+"    "+CHR$(133)+" "
1270 R3$ = CHR$(133)+"    "+CHR$(133)+" "
1280 R4$ = CHR$(130)+CHR$(138)+CHR$(132)+CHR$(138)
```



```
+CHR$(137)+" "
1290 R5$ = CHR$(133)+" "+CHR$(131)+CHR$(140)+"  "
1300 R6$ = CHR$(133)+"  "+CHR$(131)+CHR$(140)+" "
1310 R7$ = CHR$(139)+"   "+CHR$(139)+" "
1320 REM ***** T *****
1330 T1$ = CHR$(135)+CHR$(138)+CHR$(132)+CHR$(138)
+CHR$(136)+" "
1340 T2$ = "  "+CHR$(133)+"   "
1350 T3$ = "  "+CHR$(133)+"   "
1360 T4$ = "  "+CHR$(133)+"   "
1370 T5$ = "  "+CHR$(133)+"   "
1380 T6$ = "  "+CHR$(133)+"   "
1390 T7$ = "  "+CHR$(139)+"   "
1400 REM ***** C *****
1410 C1$ = CHR$(134)+CHR$(138)+CHR$(138)+CHR$(138)
+CHR$(136)+" "
1420 C2$ = CHR$(133)+"     "
1430 C3$ = CHR$(133)+"     "
1440 C4$ = CHR$(133)+"     "
1450 C5$ = CHR$(133)+"     "
1460 C6$ = CHR$(133)+"     "
1470 C7$ = CHR$(131)+CHR$(138)+CHR$(138)+CHR$(138)
+CHR$(136)+" "
1480 REM ***** O *****
1490 O1$ = CHR$(134)+CHR$(138)+CHR$(138)+CHR$(138)
+CHR$(140)+" "
1500 O2$ = CHR$(133)+"   "+CHR$(133)+" "
1510 O3$ = CHR$(133)+"   "+CHR$(133)+" "
1520 O4$ = CHR$(133)+"   "+CHR$(133)+" "
1530 O5$ = CHR$(133)+"   "+CHR$(133)+" "
1540 O6$ = CHR$(133)+"   "+CHR$(133)+" "
1550 O7$ = CHR$(131)+CHR$(138)+CHR$(138)+CHR$(138)
+CHR$(137)+" "
1560 REM ***** N *****
1570 N1$ = CHR$(134)+CHR$(138)+CHR$(140)+" "+CHR$(
132)+" "
1580 N2$ = CHR$(133)+" "+CHR$(133)+" "+CHR$(133)+"
 "
1590 N3$ = CHR$(133)+" "+CHR$(133)+" "+CHR$(133)+"
 "
1600 N4$ = CHR$(133)+" "+CHR$(133)+" "+CHR$(133)+"
 "
1610 N5$ = CHR$(133)+" "+CHR$(133)+" "+CHR$(133)+"
 "
1620 N6$ = CHR$(133)+" "+CHR$(133)+" "+CHR$(133)+"
 "
1630 N7$ = CHR$(139)+" "+CHR$(131)+CHR$(138)+CHR$(
137)+" "
1640 REM ***** E *****
1650 E1$ = CHR$(134)+CHR$(138)+CHR$(138)+CHR$(138)
+CHR$(136)+" "
1660 E2$ = CHR$(133)+"     "
1670 E3$ = CHR$(133)+"     "
1680 E4$ = CHR$(135)+CHR$(138)+CHR$(138)+CHR$(136)
+"  "
1690 E5$ = CHR$(133)+"     "
1700 E6$ = CHR$(133)+"     "
1710 E7$ = CHR$(131)+CHR$(138)+CHR$(138)+CHR$(138)
+CHR$(136)+" "
1720 REM ***** F *****
1730 F1$ = CHR$(134)+CHR$(138)+CHR$(138)+CHR$(138)
+CHR$(136)+" "
1740 F2$ = CHR$(133)+"     "
1750 F3$ = CHR$(133)+"     "
1760 F4$ = CHR$(130)+CHR$(138)+CHR$(138)+CHR$(136)
+"  "
1770 F5$ = CHR$(133)+"     "
1780 F6$ = CHR$(133)+"     "
1790 F7$ = CHR$(139)+"     "
1800 REM ***** U *****
1810 U1$ = CHR$(132)+"   "+CHR$(132)+" "
1820 U2$ = CHR$(133)+"   "+CHR$(133)+" "
1830 U3$ = CHR$(133)+"   "+CHR$(133)+" "
1840 U4$ = CHR$(133)+"   "+CHR$(133)+" "
1850 U5$ = CHR$(133)+"   "+CHR$(133)+" "
1860 U6$ = CHR$(133)+"   "+CHR$(133)+" "
1870 U7$ = CHR$(131)+CHR$(138)+CHR$(138)+CHR$(138)
+CHR$(137)+" "
1880 REM ***** NAME *****
1885 PRINT cls$;
1890 FOR Z = 1 TO 8
1900 PRINT "      ";C1$;O1$;N1$;N1$;E1$;C1$;T1$;"
       ";F1$;O1$;U1$;R1$;S$
1910 PRINT "      ";C2$;O2$;N2$;N2$;E2$;C2$;T2$;"
       ";F2$;O2$;U2$;R2$;S$
1920 PRINT "      ";C3$;O3$;N3$;N3$;E3$;C3$;T3$;"
       ";F3$;O3$;U3$;R3$;S$;S$
1930 PRINT "      ";C4$;O4$;N4$;N4$;E4$;C4$;T4$;"
       ";F4$;O4$;U4$;R4$;S$
1940 PRINT "      ";C5$;O5$;N5$;N5$;E5$;C5$;T5$;"
       ";F5$;O5$;U5$;R5$;S$
1950 PRINT "      ";C6$;O6$;N6$;N6$;E6$;C6$;T6$;"
       ";F6$;O6$;U6$;R6$;S$;S$
1960 PRINT "      ";C7$;O7$;N7$;N7$;E7$;C7$;T7$;"
       ";F7$;O7$;U7$;R7$;S$
1970 PRINT S$;S$
1980 NEXT Z
1990 PRINT "To see the rules enter 9."
2000 GOTO 320
```



A message from SID

Don't you like the way your PCW talks to you? Mostyn Davies explains how to change it with the help of CP/M's SID program.

Every PCW program has its quota of unhelpful prompts and error messages. Take PIP for example: rather than say 'sorry, your disc is full' it says 'ERROR: DISC WRITE NO DATA BLOCK'. Wouldn't it be nice if you could change these messages to suit your own tastes - humorous, obscure, or just helpful. With a little ingenuity, and the help of SID, you can.

Picture the scene: there you are, good old SuperCalc clattering away like a good 'un on drive M, doing wonders on next year's budget. Midnight strikes, you sigh happily, save the file, switch off and go to bed. Next morning you restart to find your data disc blank - you had saved it all to drive M instead of A and lost the lot.

What you need is for the prompt you get after the '/Save' command to remind you to stick that all-important 'A:' in front of the filename. In theory, it isn't difficult. All you have to do is get into the program, find where the error message is stored, change it and save the new version. Let's take the SuperCalc example and follow it through, but you should be able to use this technique with any program you like - even LocoScript.

The first job is to start your PCW up using the CP/M+ disc in drive A. Now you need to get the necessary utility program into drive M - SID.COM from side 3 of the PCW master discs. To do this, when faced with the A> prompt and your copy of the CP/M startup disc in the drive, run PIP with the command PIP[RETURN] -you will see the asterisk prompt. Now put your copy of side 3 of the PCW discs in drive A and type:
M:=SID.COM [RETURN]
[RETURN]

Now you are back at the A> prompt, and ready for the clever bit.

Down in the dumps

To do the SuperCalc example, you need to work on the file SC2.COM which contains the program. Copy SC2.COM from your SuperCalc work disc to a spare disc - never use SID on working copies of programs since if things go wrong you will need to go back to the old version.

With this new scratch copy on the disc in the A drive, type:
M:=SID SC2.COM

This will automatically put SID to work on your program and leave you with a neat but cryptic display that says NEXT MSIZE PC END, with mystery numbers underneath and a # prompt sitting there waiting. Write down the hex number under NEXT (see the box for what hex numbers mean) for future reference - this tells you where your program ends.

SID is not a very friendly character. He was designed as a programmers' tool, and takes very obscure commands. The Amstrad manual says almost nothing useful, so if you want to learn more than is printed here you will need to buy a book like 'The Amstrad CP/M Plus' by Clarke & Powys-Lybbe. If you give a wrong command, SID only says '?' by way of an error message.

The Open Sesame command d0100[RETURN] will amaze and delight you with a screenful of code numbers and, on the right, gobbledegook characters mixed up with the odd word that makes sense. The d stands for 'dump', and the code is just that: machine code written in 'hex bytes' - see the box for details. The section on the right of the screen does its best to print the text characters corresponding to these codes. When the code numbers represent real words, you can read them, otherwise you are looking at the guts of the program itself.

You are now faced with the # prompt again. If you type d [RETURN] the next section of memory is displayed for you, going on from where the last block finished. Every time you do this, you step through the memory, block by block starting from address 0100. Keep using the d command and keep your eyes open, scanning the right hand side of the screen for the message you want to alter. Be patient, it's in there someplace!

The leftmost column shows you the memory addresses, and if you go on long enough you'll finally pass the number under NEXT that you noted down earlier. That's the end of the program, and there's no point going any further. You can go back to the beginning just by typing d0100[RETURN], which will start dumping again from 0100. Of course, you can use any address you like after the d - d1F34 dumps a block of bytes from address 1F34.

Bring on the substitutes

For the customised SuperCalc, you want to change the screen message that comes up when you give the /Save command. Normally this says 'Enter File Name (or <RETURN> for directory)'. So, looking through the file SC2.COM with repeated d's, this text finally appears in the block starting at address 1B40. How about changing the text in the brackets to read '(don't forget the A:)' instead?

How programs are stored

Many of the more complex programs you can buy are stored in several files on the disc, so it might be difficult to find out where a message is that you want to change.

The first place to look is always the .COM file. If you run a program (like SuperCalc) by typing 'SC2' at the A> prompt, then that program will be stored on your disc in a file called SC2.COM. If you list out the disc directory, you will see that file. However, there may also be other files which end '.OVR'. These too contain program code, and you may well find the message you are looking for hidden in these files instead of the .COM file.

LocoScript is, of course, different in all major respects. If you want to alter any of the text that appears in the menus, you will need to look through the files J*something*LOCO.EMS (the *something* is a two digit number which depends on which version of LocoScript you have) and SCRIPT.JOY. Even then you may not find the text of the menus verbatim, since LocoScript stores its internals in a fairly obscure way.

Look back to the screen to find the start address of the text you want to change: looking at the left hand column of addresses, pick up the four character address number on the relevant line, and count across, in hex, until you reach the first letter to be changed. In this case, the opening bracket is at address 1B5F.

Now to substitute the new text. This is done with the 's' command. The one really important thing to remember is that the new text must be no longer than the old text. If it is any shorter, you must pad out the end with spaces to make it the same length. Type s 1B5F [RETURN], and you will see the address redisplayed with the code it currently contains beside it.

To enter the new text, just type it in preceded by a double-quote mark:
"(don't forget the A:)
type six spaces afterwards to match lengths, and press [RETURN]. SID thinks you might want to change more text, so it prompts you with the next address for you to alter. Since the modifications are now all done, type a full stop then [RETURN] to leave substitution mode. You now see the # prompt again.

To check the changes, dump the new block to the screen again with d1B40 - remember 1B40 is just the address used in this example, you should use whatever one you noted down for the text you are changing. You can see the changed message in place. If it looks wrong, use the substitute process again to modify it until it is OK.

The only thing left to do is to save the new version to disc and get back to CP/M. This is very simple, using SID's 'w' (for 'write') command. Type w followed by the name of the file to hold the new version. For this example, write the file back to SC2.COM, which will erase the old version on the disc. So:
wSC2.COM [RETURN]
does the trick. To leave SID, press the [STOP] key and you are back at CP/M's friendly A> prompt.

You use the new version of SuperCalc quite normally. Typing SC2 [RETURN] runs the modified program, and you can see the prompt has been changed in the /Save command.

Health warning

The methods described in this article are quite reliable, but there are a couple of potential pitfalls.

If, when you are modifying a program, you accidentally change part of it which is not a simple error message, you could find that when you run the modified version the machine crashes. Before you use SID, always make sure you keep a copy of the unchanged file on a backup disc in case anything goes wrong. A serious program crash can only lose you the contents of the M drive.

Don't forget when you are substituting some text of your choice for an old prompt or error message, the new version must be exactly the same length as the old one. You must add trailing spaces to match the lengths if the new text would be shorter.

Hacker Note

Another two interesting commands of SID's are 'L' and 'A'. L0100 Lists out the memory from address 0100 as 8080 assembler mnemonics, and A0100 allows you to enter and Assemble 8080 mnemonics into that address.

A.E.T.

Acronym Explanation Time: PIP stands for 'Peripheral Interchange Program'; and SID for 'Symbolic Instruction Debugger'. Are you any the wiser?

Putting on the HEX

If you run SID, you can't have failed to have noticed that the number pairs it prints on the screen are a bit strange. Some of the numbers are conventional digits, but some are letters of the alphabet. This is a sure sign of 'hexadecimal' numbers - 'hex' for short, or 'base 16' to all students of modern maths.

If you can count in 10's you can count in 16's just as easily, and that's really all that hexadecimal is about. In hex, you use the digits 0 to 9 as normal, and then the letters A to F to correspond to the numbers 10 to 15 respectively.

In a decimal, a number like 234 means 2x100+3x10+4x1. In other words, the digits stand for ones, tens, hundreds, thousands and so on. In hexadecimal, the digits stand for ones, 16's, 256's, 4096's and so on, with each column 16 times bigger than the last one. Thus 234 in hex is the number2x256+3x16+4x1, or 8244 in decimal.

So, 10 in hex is 16 in decimal. 1A (hex) is 26 (decimal), and FF (hex) is 255 (decimal). Hex may seem awkward to humans, but computers love it because just two digits can count to 255, where decimal can only get to 99.Two hex digits are called a byte. One hex digit is called a nybble (honestly!).

***NEXT MONTH:** an article by Helen Bradley (Sale, Vic) on using SID to alter SuperCalc2 printer codes and experimenting with CP/M utilities.*

Cheat Mode

The only pages of Pokes and Tips read by real cheats to beat an Amstrad at its own game.

BRUCE LEE

This tip from Jamie Dimech in Riverstone, NSW saves you having to do the first 14 screens. He writes "at the beginning, move to the far right hand side of the screen then go down to the bottom. Make sure you are right up against the wall, then press down on the joystick. You will enter a different screen - missing the first 14.

BOULDER DASH

Philip Green (and Mum) in Rathmines, NSW advise that to get out of K3 you need to press the fire button (shift) and the key for the direction of the obstruction. Be careful that you are not standing under a boulder.
They also discovered (as no operating instructions were received with the game) that the following keys are the important ones:

Shift	-	START
Z	-	LEFT
X	-	RIGHT
/	-	DOWN
+	-	UP

DEFEND OR DIE

This poke from Martin West (Two Wells, SA) allows you to move the joystick up instead of pressing the space bar (line 20). The second poke adds on lives as you progress through the game. Don't worry if they don't come in the first game, sometimes it takes them a while. You may also get some strange scores.

```
1 'Defend or Die Disc Poke
2 ' Martin West
3 'The Amstrad User Aug 87
10 MEMORY &3AAA: LOAD "DEFEND",&4000
20 POKE &5764,72:POKE &5155,100
30 CALL &4025
```

BATTLEFIELD GERMANY

Jaranpat Khaejornbut has sent tips for the PSS war game. Among them he has spotted an omission from the instructions and a bug.
1. Something unusual happens with the cursor. If you leave it over a hex occupied by a unit at the end of your turn, it will have disappeared after the computer's turn. So before you press X to finish your turn make sure that the cursor hasn't been left over any of your units. You could place the cursor over the enemy's units and make your task that much easier.
2. The R key has been omitted from the instructions but is very useful. It centres the cursor on the screen.
3. Don't rush into combat with enemies that have a higher efficiency rating. Try to cut their supplies first. When they're out of supplies, fatigue increases and affects efficiency. The efficiency display stays the same but its effect on the combat calculations will have altered and has considerable influence on the result.

To cut the enemy supply route find the weakest point in the line and attack it. Once you've made a gap, units can move through it with less movement points. Use the air mobile units and drop paratroopers behind enemy lines to help the attack. These units may have to be sacrificed but it's worth it.

TEMPEST

TC Spanner has a quick tip for the Electric Dreams shoot-em-up. Hold down the delete key and type in the following in uppercase: bess, tony, credits, dates, capri, ferrari. Each one gets a different response.

DYNAMITE DAN II

Dan, Dan the Mirrorsoft man has been given a good poking by Paul Robson of Middlesbrough. Entered using Method 1, the poke supplies Dan with infinite energy.

```
1 'Dynamite Dan II
2 'by Paul Robson
3 'The Amstrad User Aug 87
10 MEMORY &3FFF
20 MODE 1:LOAD"!"
30 FOR f=&A7F8 TO &A80E
40 READ a$:POKE f,VAL("&"+a$):NEXT
50 DATA 21,a8,01,22,d5,46,c3,05
55 DATA 40,f5,cd,19,bd,3e,00,32
60 DATA c5,15,32,cf,15,f1,c9
70 CALL &A7F8
```

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.

EQUINOX

L Bonnet and G Wilson know how to get infinite lives on the Mikro-Gen game. All you do is get the Pete object and return to the start screen. Go to the top left corner and press the letters RNC and you have infinite lives. However, the time limit still runs out so it's not all plain sailing.

GET DEXTER

A correction to Andrew Mansell's pause for the PSS game. The keys should be the D and X keys.

MR FREEZE

Paul Taylor of Frome has sent in the tip that de-icing number six twice in the Firebird game will give you infinite lives.

THRUST

Colin Ward has supplied you with an excellent poke for Firebird's cheapie. Entered via Method 1, it gives 9999 fuel points and infinite lives - what more could you want?

```
10 DATA 2a,00,9f,22,38,bd,cd,37
20 DATA bd,3e,f7,32,c1,3c,3e,c3
40 DATA 21,1f,9e,32,30,00,22,31
60 DATA 00,cd,00,4f,c3,00,3c,3e
80 DATA 99,32,66,71,21,3e,04,22
90 DATA 3f,43,c3,00,70,c9,2a,38
110 DATA bd,22,00,9f,21,ee,99,22
120 DATA 38,bd,3e,c3,21,00,9e,32
140 DATA 19,bd,22,1a,bd,c3,00,36
150 DATA c3,00,9e
160 MEMORY &2000:BORDER 0:MODE 1
180 n=0:FOR x=&9E00 to &9E4A
200 READ a$:POKE x,VAL("&"+a$)
210 n=n+VAL("&"+a$):NEXT x
230 IF n<>6737 THEN PRINT"error in
data":END
240 LOAD"thrust 1":CALL &9E2E
```

SUPER PIPELINE II

Yet another of the Computer Hits 2 games poked. Again it's Richard Hodges' doing. Richard has found a routine incorporated in the Pipeline code, and a couple of his pokes re-enables it. The s key starts again, w or l proceed to the next pipe and A increments your high score (going into slow-motion as it does). Use Method 1 to run the poke. Then load the game and press the required key. An infinite lives poke is included.

```
1 ' Super Pipeline II
2 ' by Richard Hodges
3 ' The Amstrad User Aug 87
10 OPENOUT"a":MEMORY &3FF
20 LOAD"!sup-II"
30 ' enable S,A,W,L keys
40 POKE &773,0:POKE &798,&FF
50 ' infinite lives
60 POKE &6CC,0:POKE &69D,0
70 CALL &400
80 GOTO 70
```

ZUB

Another Mastertronic game bites the dust thanks to Richard Hodges. Endless (makes a change from infinite) lives, number-of-platforms choice and removal of droids are the capabilities of this Method 1 poke.

```
1 ' Zub
2 ' by Richard Hodges
3 ' The Amstrad User Aug 87
10 MODE 1:FOR p=&BE00 TO &BE11
20 READ p$:POKE p,VAL("&"+p$)
30 NEXT:INPUT"Endless lives ";i$
40 IF UPPER$(i$)="Y" THEN POKE
&BE01,0
50 INPUT"Remove droids ";i$
60 IF UPPER$(i$)="Y" THEN POKE
&BE06,&C9
70 INPUT"Enter number of platforms
per planet (normally 18) ";s
80 IF s<1 OR s>18 THEN 70
90 POKE &BE0B,(19-s)
100 MEMORY &2000:LOAD"",&2040
110 POKE &2042,&20:POKE &2058,&C9
120 MODE 0:CALL &2040
130 POKE &4025,&BE:CALL &4000
140 DATA 3e,3c,32,a2,43,3e,3e,32
150 DATA 75,45,3e,01,32,af,44,c3
160 DATA 00,9d
```

FEUD

Mastertronic's cheapie has been given a good going over by Robert Brooks. He has kindly provided you with a choice of infinite spells and the possibility of skipping the herb-collecting stage. Enter the poke using Method 1.

```
1 ' Feud
2 ' by Robert Brooks
3 ' The Amstrad User Aug 87
10 addr=&5400
20 PRINT"No need to collect herbs (Y/
N) ?"
30 GOSUB 140: IF a THEN RESTORE
100:FOR n=0 TO 3:READ a$:POKE
n+addr,VAL("&"+a$):NEXT:addr=addr+n
40 PRINT"All spells infinite (Y/N) ?"
50 GOSUB 140:IF a THEN RESTORE
110:FOR n=0 TO 12:READ a$:POKE
n+addr,VAL("&"+a$):NEXT:addr=addr+n
60 PRINT"All corners act as cauldrons
(Y/N) ?"
70 GOSUB 140:IF a THEN RESTORE
120:FOR n=0 TO 3:READ a$:POKE
n+addr,VAL("&"+a$):NEXT:addr=addr+n
80 FOR n=0 TO 2:READ a$:POKE
n+addr,VAL("&"+a$):NEXT
90 MEMORY &4FFF:LOAD"FEUD":POKE
&5008,&54:MODE 1:CALL &5000
100 DATA af,32,84,09
110 DATA 21,c5,15,11,c6,15,36,ff,01,
0b,00,ed,b0
120 DATA af,32,7a,09
130 DATA c3,00,04
140 a$="":a=0:WHILE a$<>"n" AND
a$<>"Y":a$=LOWER$(INKEY$):WEND
150 IF a$="Y" THEN a=1
160 RETURN
```

ACADEMY

We've received two sets of tips for the CRL Mastergame, some general ones from Guy Blackburn and detailed ones on the first four missions from S. Reeves.

Lasers are the most useful weapon. Missiles are best used to keep the pressure off. The mission information section will give you a guide to the conditions on the planet. A skimmer is provided for each mission and it's best to use that at first. Once you realise the equipment required it's best to design your own. Missions can't always be completed if you choose the wrong skimmer, and there's nothing worse than battling a horde of aliens for 20 minutes only to find you've forgotten a vital piece of equipment.

In any mission where the robots are spread out, a compass and scanner are vital equipment. It's also a good idea to go for the most powerful lasers and shields. The high-power steering unit can be oversensitive so go for a medium-power one instead. Don't waste space on flares and an infrared unit unless they're essential.

Each set of four missions needs an average score of only 90%. This means that if you score 100% in the first three you need only get 60% on the fourth to move onto the next level. If you can't get 90% in one of the earlier missions skip it and come back to it if you have to after tackling the other three.

Level 2

Cipher: collect and assemble the codes from nearby reactors.

At the OK Corral: one for budding John Waynes.

Where to, guv?: launch to a solitary jump pad. Unfortunately the jump network

has been taken over by pirates. You can always jump out of trouble but watch the fuel.
Hide and Seek: Involves eliminating several solar-disc complexes. You'll need a suppressor droid.

Level 3
Laserium: more straight-ahead blasting.
Hades II: very poorly lit - infrared and flares needed.
The sands of time: a whole network of reactors to eliminate. The best way is to sneak missile at point-blank range.
Mission improbable: collect and assemble codes. Suppressor droids come in handy here.

Level 4
Ceti revisited: a bit of nostalgia here on Tau Ceti III.
Out of the frying pan: shoot-em-up with a nasty welcoming committee.
Don't panic: solar discs and reactor complexes mean delay bombs are needed.
Needle in a haystack: navigational skills needed, plus finding a path through a minefield.

Level 5
The coalmine: planet in permanent darkness, so infrared is vital.
Paz!: watch out for indestructible supermissiles. Average survival time 35 seconds.
Protector: chase the tracker units. You need a fast skimmer, but this mission is quite easy once you get the hang of it.
The shepherd: find and bring back, intact, watch towers.

Level 1
If it moves: your base is surrounded by four sets of beacons in twos. Don't destroy them as they are necessary for navigation. Design your ship with good lasers, scanner, 4 bombs, 8 missiles, anti-missile missiles (amms), compass and good main drive. First clear anything from around the GalCorp Lander (GLV) that isn't a beacon.
Return to the GLV and re-equip. Fly out along the line of one of the sets of beacons. When you find a ship steer towards it but very slowly. There are waves of ships which thrash you in close combat, so stop and blast them one by one. If they don't approach you at first edge a bit nearer. Use all missiles, turn and head for the GLV. Change to rear view, shooting dropping bombs the whole time. Be careful not to crash into the GLV.
Repeat this for each set of beacon directions and you should easily get a pass. If it gets dark, return to the GLV and wait five minutes.
Red dawn: use the same ship as in mission one,but with a jump unit, infrared or flares, and 8 amms. Destroy enemy near the GLV, then re-equip and use the jump pad.
Beacons play an important part so don't blast them straight away. A line of beacons will point you in the direction of something interesting - blast them on the way back. If a mission involves jump pads check each destination for fuel centres because the pads are in a one-way network and it may be a long way back to the lander.
If the mission involves docking with reactors be very careful not to shoot at them. One shot and the door will close.
Fortresses will often be tucked in next to a reactor so circle it until you can get a clean shot at the fortress.
Locate the robot factories and keep well away from them until the last minute or else they'll slaughter you. Blast everything else then fly between the factories and bomb them. Use amms against the missiles that the factories lob at you. If you run out of bombs or amms head for home - you should have enough fuel. Refuel, re-equip and head back for the other factories.
Meltdown: this is probably the hardest of the first four missions. Get a ship with high shield, high lasers, 8 amms, 4 bombs, scanner, compass and missiles. Destroy the mines shown as four dots close together. Every so often slow down and check the scanner for anything moving - blast it. Avoid or bomb the fortresses. The volcanoes are indestructible and not the reactor. Time is the most important thing, so you can't wait for the sun to come up but must persevere.
Softly, softly: this is dead easy. Use a ship with good lasers, good main drive, high shield, scanner and compass. Follow the corridor of mines and repeat the process of stopping and checking the scanner for anything moving - kill it. Use missiles on ships attacking you, for speed.
Scoring is on time but dock carefully at the end or you'll.....

MONTY ON THE RUN

Richard Hodges is responsible for poking Gremlin's furry character around. Entered using Method 1, it works only on 664 and 6128 machines. There is a comprehensive list of cheats you can supply: infinite lives, crusher-stopper, remove collision detection. Delete the data you don't require and get playing.

```
1 ' Monty on the Run -664/6128
2 ' by Richard Hodges
3 ' The Amstrad User Aug 87
10 GOSUB 20:CALL &BE00
20 DATA 21,80,be,11,e9,02,01,20
30 data 00,ed,b0,00,00,00,00,00
40 data 21,40,20,11,40,00,01,f7
50 data 01,ed,b0,21,40,00,e5,21
60 data 89,03,e5,21,f7,01,e5,21
70 data 1f,b1,e5,21,bb,02,e5,f1
75 data 21,e4,b1,11,f7,01,f3,c9
80 MEMORY &2000:LOAD"":CALL &42F3
90 LOAD"!",&2040:POKE &2236,5
100 FOR x=&BE00 TO &BE37:READ a$
110 POKE x,VAL("&"+a$):NEXT
120 RESTORE 240:p=&BE80
130 WHILE b$<>"end":READ b$
140 POKE p,VAL("&"+b$):p=p+1:WEND
150 RETURN
200 ' infinite lives
240 DATA 3e,00,32,bc,9c
250 ' stop crushers
260 DATA 3e,c9,32,e3,9b
270 ' remove collision detection
280 DATA 3e,c9,32,a6,b6
290 ' leave the following lines
300 DATA c3,e9,82,end
```

BOMBJACK

Kojo Ellimah has a revised poke which supplies you infinite lives and three bonus coins at the start of each screen, plus a power ball after collecting the first 10 bombs. Use Method 1.

```
1 ' Bombjack
2 'by Kojo Ellimah
3 ' The Amstrad User Aug 87
10 MEMORY 5999:BORDER 0:MODE 0:INK
0,1
20 FOR f=0 TO 15:READ a:INK f,a:NEXT
f
30 LOAD "!bjscreen.bin",&C000
40 LOAD"!bjcode.bin",6000
50 POKE 6653,0
60 POKE 12222,12
70 CALL 6000
80 DATA 1,0,26,8,24,13,11,6
90 DATA 15,16,5,2,6,3,20,10
```

Cribbage: Part 2 - the card game

from Nick Herrick

Here's the final part of Nick Herrick's version of Cribbage. To remind you, it's a card game which has an element of luck as you don't know which cards are going to be dealt. On the other hand (no pun intended!), it is also a game of skill. You must "peg" (score) as many points as possible based on the cards provided at each hand.

```
4070 IF k<=21 THEN 4110
4080 PRINT #1,"Can you go? (Y/N)"
4090 i$=UPPER$(INKEY$):IF i$<>"Y" AND i$<>"N" THEN 4090
4100 IF i$="N" THEN yplay=yplay+1:q=0:RETURN
4110 INPUT #1,"Please enter the number of the card you wi
sh to play ",gg
4120 IF gg=g1 OR gg=g2 OR gg=g3 THEN PRINT #1,"You've alr
eady played it! Don't cheat":GOTO 4110
4130 IF gg<1 OR gg>4 THEN PRINT #1,"You haven't got that
card: try again":GOTO 4110
4140 IF k+yhn(gg)>31 THEN PRINT #1,"You can't count":GOTO
 4080
4150 GOSUB 5000
4160 IF g1=0 THEN g1=gg ELSE IF g2=0 THEN g2=gg ELSE IF g
3=0 THEN g3=gg ELSE g4=gg
4170 r1=yhnd(gg):r2=xhnd(v1):r3=xhnd(v2):r4=xhnd(v3):r5=x
hnd(v4)
4180 xhnd(v)=yhnd(gg)
4190 IF r1=r2 AND r1=r3 AND r1=r4 THEN yscore=yscore+12:G
OTO 4240
4200 IF r1=r2 AND r1=r3 THEN yscore=yscore+6:GOTO 4240
4210 IF r1=r2 THEN yscore=yscore+2:GOTO 4240
4220 GOSUB 5300
4230 yscore=yscore+rscore
4240 k=k+yhn(gg):IF k=31 OR k=15 THEN yscore=yscore+2
4250 IF k=31 THEN yplay=yplay+1:cplay=cplay+1
4260 GOSUB 5650:FOR t=1 TO 999:NEXT t
4270 IF k=31 THEN k=0
4280 q=0:RETURN
4290 ' ******** COMPUTER PLAY *********
4300 IF f4>0 THEN q=1:cplay=cplay+1:RETURN
4310 v=v+1
4320 v1=v-1:v2=v-2:v3=v-3:v4=v-4
4330 IF v=1 THEN v2=0:v3=0:v4=0
4340 IF v=2 THEN v3=0:v4=0
4350 IF v=3 THEN v4=0
4360 f=0:ff=0:WHILE f=0 AND ff<4
4370 ff=ff+1:IF ff>4 THEN 4390
4380 IF ff=f1 OR ff=f2 OR ff=f3 OR k+chn(ff)>31 THEN 4370
 ELSE f=1
4390 WEND
4400 IF f=0 THEN PRINT #1,"I can't go":cplay=cplay+1:FOR
t=1 TO 600:NEXT t:q=1:RETURN
4410 r2=xhnd(v1):r3=xhnd(v2):r4=xhnd(v3):r5=xhnd(v4)
4420 f=0:ff=0:WHILE ff<4 AND f=0
4430 ff=ff+1:IF ff>4 THEN 4500
4440 IF ff=f1 OR ff=f2 OR ff=f3 OR k+chn(ff)>31 THEN 4430
4450 r1=chnd(ff)
4460 IF r1=r2 AND r1=r3 AND r1=r4 THEN cscore=cscore+12:f
=1:GOTO 4500
4470 IF r1=r2 AND r1=r3 THEN cscore=cscore+6:f=1:GOTO 450
0
4480 GOSUB 5300
4490 IF rscore>0 THEN cscore=cscore+rscore:f=1
4500 WEND
4510 IF f=1 THEN 4670
4520 f=0:ff=0:WHILE ff<4 AND f=0
4530 ff=ff+1:IF ff>4 THEN 4580
4540 IF ff=f1 OR ff=f2 OR ff=f3 OR k+chn(ff)>31 THEN 4530
4550 IF k+chn(ff)=15 THEN cscore=cscore+2:f=1
4560 r1=chnd(ff):IF r1=r2 THEN cscore=cscore+2:f=1
4570 IF k+chn(ff)=31 THEN cscore=cscore+2:f=1
4580 WEND
4590 IF f=1 THEN 4670
4600 IF f3>0 AND ff<5 THEN 4670
4610 IF level=3 THEN 4720
4620 t=0
4630 ff=INT(RND*4+1)
4640 t=t+1
4650 IF k=0 AND chnd(ff)=5 AND t<6 THEN 4630
4660 IF ff=f1 OR ff=f2 OR ff=f3 OR chn(ff)+k>31 THEN 4630
4670 IF f1=0 THEN f1=ff ELSE IF f2=0 THEN f2=ff ELSE IF f
3=0 THEN f3=ff ELSE f4=ff
4680 xhnd(v)=chnd(ff):k=k+chn(ff)
4690 GOSUB 4930:GOSUB 5650:FOR t=1 TO 999:NEXT t
4700 IF k=31 THEN k=0:yplay=yplay+1:cplay=cplay+1
4710 q=1:RETURN
```

```
4720 ' ***** LEVEL 3 COMPUTER PLAY ****
4730 FOR z=0 TO 6:s(z)=0:NEXT z
4740 FOR z=1 TO 4
4750 IF z=f1 OR z=f2 OR z=f3 OR k+chn(z)>31 THEN s(z)=100
:GOTO 4840
4760 r1=chnd(z)
4770 IF r1=r2+1 OR r1=r2+2 OR r1=r3+1 OR r1=r3+2 OR r1=r2
-1 OR r1=r2-2 OR r1=r3-1 OR r1=r3-2 THEN s(z)=s(z)+4
4780 IF r1=5 THEN s(z)=s(z)+5
4790 IF k+chn(z)*2=15 THEN s(z)=s(z)+6
4800 IF k=0 AND r1>5 THEN s(z)=s(z)+2
4810 IF k+chn(z)<15 THEN s(z)=s(z)+2
4820 IF k+chn(z)=21 THEN s(z)=s(z)+3
4830 IF r1=1 THEN s(z)=s(z)+1
4840 NEXT z
4850 ff=0:a=0
4860 WHILE ff=0
4870 FOR t=1 TO 4
4880 IF s(t)=a THEN ff=t:GOTO 4910
4890 NEXT t
4900 a=a+1
4910 WEND
4920 GOTO 4670
4930 ' *** COMPUTER HAND CARD PRINT ***
4940 IF box=1 THEN y=1 ELSE y=10
4950 IF ff=1 THEN x=1 ELSE IF ff=2 THEN x=7 ELSE IF ff=3
THEN x=13 ELSE x=19
4960 GOSUB 5530
4970 IF box=1 THEN y=1 ELSE y=10
4980 x=28:z=ff:GOSUB 5600
4990 RETURN
5000 ' ***** YOUR HAND CARD PRINT *****
5010 IF box=0 THEN y=1 ELSE y=10
5020 IF gg=1 THEN x=1 ELSE IF gg=2 THEN x=7 ELSE IF gg=3
THEN x=13 ELSE x=19
5030 GOSUB 5530
5040 IF box=0 THEN y=1 ELSE y=10
5050 x=28:z=gg:GOSUB 5700
5060 RETURN
5070 ' ** PRINT BACKS FOR NEXT PLAY **
5080 x=28:y=1:GOSUB 5230
5090 y=10:GOSUB 5230
5100 RETURN
5110 ' ********** PRINT CARD *********
5120 PEN p
5130 IF n(z)<>10 THEN 5150
5140 LOCATE x,y:PRINT c$(z):GOTO 5160
5150 LOCATE x,y:PRINT c$(z);" "
5160 LOCATE x,y+1:PRINT "     "
5170 GOSUB 7110
5180 LOCATE x,y+6:PRINT "     "
5190 IF n(z)<>10 THEN 5210
5200 LOCATE x,y+7:PRINT c$(z):GOTO 5220
5210 LOCATE x,y+7:PRINT c$(z);" "
5220 RETURN
5230 ' ******** PRINT BACKS **********
5240 PEN 1
5250 FOR t=1 TO 8
5260 LOCATE x,y:PRINT back$
5270 y=y+1
5280 NEXT t
5290 RETURN
5300 ' ******** CALCULATE RUNS *******
5310 FOR z=0 TO 8:xhn(z)=0:NEXT z
5320 rscore=0:IF r3=0 THEN RETURN
5330 IF r2=0 THEN RETURN
5340 xhn(1)=r1:xhn(2)=r2:xhn(3)=r3:xhn(4)=r4:xhn(5)=r5
5350 FOR z=1 TO 4
5360 FOR a=z+1 TO 5
5370 IF xhn(a)=xhn(z) THEN xhn(a)=-5
5380 IF xhn(a)<xhn(z) THEN 5420
5390 t=xhn(a)
5400 xhn(a)=xhn(z)
5410 xhn(z)=t
5420 NEXT a
5430 NEXT z
5440 IF r5=0 THEN 5460
5450 IF xhn(1)=xhn(2)+1 AND xhn(1)=xhn(3)+2 AND xhn(1)=xh
n(4)+3 AND xhn(1)=xhn(4)+3 THEN rscore=rscore+5:RETURN
5460 IF r4=0 THEN 5490
5470 IF xhn(1)=xhn(2)+1 AND xhn(1)=xhn(3)+2 AND xhn(1)=xh
n(4)+3 AND r1+r2+r3+r4=xhn(1)+xhn(2)+xhn(3)+xhn(4) THEN r
score=rscore+4:RETURN
5480 IF xhn(2)=xhn(3)+1 AND xhn(2)=xhn(4)+2 AND xhn(2)=xh
n(5)+3 AND r1+r2+r3+r4=xhn(2)+xhn(3)+xhn(4)+xhn(5) THEN r
score=rscore+4:RETURN
5490 IF xhn(1)=xhn(2)+1 AND xhn(1)=xhn(3)+2 AND r1+r2+r3=
xhn(1)+xhn(2)+xhn(3) THEN rscore=rscore+3:RETURN
5500 IF xhn(2)=xhn(3)+1 AND xhn(2)=xhn(4)+2 AND r1+r2+r3=
xhn(2)+xhn(3)+xhn(4) THEN rscore=rscore+3:RETURN
5510 IF xhn(3)=xhn(4)+1 AND xhn(3)=xhn(5)+2 AND r1+r2+r3=
xhn(3)+xhn(4)+xhn(5) THEN rscore=rscore+3
5520 RETURN
5530 ' ******** DELETE CARD **********
5540 PAPER 0
5550 FOR t=1 TO 8
5560 LOCATE x,y:PRINT "     "
5570 y=y+1
5580 NEXT t
5590 RETURN
5600 ' **** PRINT COMPUTER'S PLAY ****
5610 IF cs$(z)=heart$ OR cs$(z)=diam$ THEN p=2 ELSE p=0
5620 PAPER 3:c$(z)=chand$(z):n(z)=chnd(z):h$=cs$(z)
5630 GOSUB 5110
5640 RETURN
5650 ' **** PRINT RESULTS SO FAR *****
5660 CLS #1:PRINT #1,"count=";k
5670 PRINT #1,"Your score=";yscore
5680 PRINT #1,"My score=";cscore
```

```
5690 RETURN
5700 ' ******* PRINT YOUR PLAY *******
5710 PAPER 3
5720 IF ys$(z)=heart$ OR ys$(z)=diam$ THEN p=2 ELSE p=0
5730 c$(z)=yhand$(z):n(z)=yhnd(z):h$=ys$(z)
5740 GOSUB 5110
5750 RETURN
5760 ' ************ WIN **************
5770 IF cscore>=121 THEN cwin=cwin+1
5780 IF yscore>=121 THEN ywin=ywin+1
5790 BORDER 0:PAPER 0:PEN 3:CLS
5800 PRINT:PRINT:PRINT
5810 IF cscore>=121 THEN PRINT "          HOORAY!!!  I wi
n!"
5820 IF yscore>=121 THEN PRINT "      Congratulations, yo
u win."
5830 PRINT:PRINT:PRINT "         You have won";ywin;"game
s"
5840 PRINT:PRINT "           I have won";cwin;"games"
5850 PRINT:PRINT:PRINT "    Do you want another game? (Y/
N)"
5860 cscore=0:yscore=0
5870 i$=UPPER$(INKEY$):IF i$<>"N" AND i$<>"Y" THEN 5870
5880 IF i$="Y" THEN RETURN
5890 PRINT:PRINT "Thank you for a very enjoyable game.":P
RINT
5900 IF cwin>ywin THEN PRINT "I am delighted to beat you.
 Better luck next time."
5910 IF ywin>cwin THEN PRINT "Well played. I will try har
der next timewe meet."
5920 IF cwin=ywin THEN PRINT "Evens! We must meet again t
o see who's  better."
5930 PRINT:END
5940 ' ******* INSTRUCTIONS **********
5950 BORDER 0:PAPER 0:PEN 2
5960 CLS:PRINT:PRINT
5970 PRINT "              CRIBBAGE":PRINT:PRINT
5980 PRINT " This game follows standard six card    cribb
age rules and scoring."
5990 PRINT "You have the option of the initial deal;after
 this the deal alternates.":PRINT
6000 PRINT " The computer shuffles and deals for youor it
self. The shuffle is complete, fairand random."
6010 PRINT:PRINT " The scores and count of cards during
 play are automatically tallied."
6020 PRINT:PRINT " The computer will play at three skill
 levels:"
6030 PRINT "1. Beginner"
6040 PRINT "2. Experienced"
6050 PRINT "3. Expert"
6060 PRINT:INPUT "Which level, 1, 2 or 3, do you want";le
vel
6070 IF level<1 OR level>3 THEN PRINT "You can't have tha
t!":GOTO 6060
6080 RETURN
6090 ' ******* CALCULATE HANDS *******
6100 q=0:IF box=1 THEN GOSUB 6160 ELSE GOSUB 6220
6110 IF yscore>=121 OR cscore>=121 THEN RETURN
6120 q=1:IF box=0 THEN GOSUB 6160 ELSE GOSUB 6220
6130 IF yscore>=121 OR cscore>=121 THEN RETURN
6140 q=2:GOSUB 6280
6150 RETURN
6160 ' ******** COMPUTER HAND ********
6170 FOR z=1 TO 4
6180 xhand$(z)=chand$(z):xhnd(z)=chnd(z):xhn(z)=chn(z):xs
$(z)=cs$(z)
6190 NEXT z
6200 GOSUB 6390
6210 RETURN
6220 ' ********** YOUR HAND **********
6230 FOR z=1 TO 4
6240 xhand$(z)=yhand$(z):xhnd(z)=yhnd(z):xhn(z)=yhn(z):xs
$(z)=ys$(z)
6250 NEXT z
6260 GOSUB 6390
6270 RETURN
6280 ' ************* BOX *************
6290 xhand$(1)=ybox1$:xhnd(1)=ybx1:xs$(1)=ys1$
6300 IF ybx1=11 OR ybx1=12 OR ybx1=13 THEN xhn(1)=10 ELSE
 xhn(1)=ybx1
6310 xhand$(2)=ybox2$:xhnd(2)=ybx2:xs$(2)=ys2$
6320 IF ybx2=11 OR ybx2=12 OR ybx2=13 THEN xhn(2)=10 ELSE
 xhn(2)=ybx2
6330 xhand$(3)=cbox1$:xhnd(3)=cbx1:xs$(3)=cs1$
6340 IF cbx1=11 OR cbx1=12 OR cbx1=13 THEN xhn(3)=10 ELSE
 xhn(3)=cbx1
6350 xhand$(4)=cbox2$:xhnd(4)=cbx2:xs$(4)=cs2$
6360 IF cbx2=11 OR cbx2=12 OR cbx2=13 THEN xhn(4)=10 ELSE
 xhn(4)=cbx2
6370 GOSUB 6390
6380 RETURN
6390 ' ******* BOX & HAND SCORE ******
6400 rscore=0
6410 PAPER 0:CLS
6420 PAPER 3
6430 FOR z=1 TO 4
6440 c$(z)=xhand$(z):n(z)=xhnd(z):h$=xs$(z)
6450 IF xs$(z)=heart$ OR xs$(z)=diam$ THEN p=2 ELSE p=0
6460 y=5:READ x
6470 GOSUB 5110
6480 NEXT z
6490 c$(z)=lift$:n(z)=lft:x=36:y=5:h$=ls$
6500 IF ls$=heart$ OR ls$=diam$ THEN p=2 ELSE p=0
6510 RESTORE 8590:GOSUB 5110
6520 IF xs$(1)=xs$(2) AND xs$(1)=xs$(3)AND xs$(1)=xs$(4)
AND xs$(1)=ls$ THEN rscore=rscore+5
6530 IF q=2 OR rscore=5 THEN 6550
6540 IF xs$(1)=xs$(2) AND xs$(1)=xs$(3)AND xs$(1)=xs$(4)
```

```
THEN rscore=rscore+4
6550 xhand$(5)=lift$:xhnd(5)=lft
6560 IF lft=11 OR lft=12 OR lft=13 THEN xhn(5)=10 ELSE xh
n(5)=lft
6570 FOR z=1 TO 4
6580 IF xhnd(z)=11 AND xs$(z)=ls$ THEN rscore=rscore+1
6590 NEXT z
6600 FOR z=1 TO 4
6610 FOR a=z+1 TO 5
6620 IF xhnd(a)>xhnd(z) THEN 6660
6630 t=xhnd(a)
6640 xhnd(a)=xhnd(z)
6650 xhnd(z)=t
6660 NEXT a
6670 NEXT z
6680 FOR z=1 TO 4
6690 FOR a=z+1 TO 5
6700 IF xhn(z)+xhn(a)=15 THEN rscore=rscore+2
6710 IF xhnd(z)=xhnd(a) THEN rscore=rscore+2
6720 NEXT a
6730 NEXT z
6740 k=xhn(1)+xhn(2)+xhn(3)+xhn(4)+xhn(5)
6750 IF k=15 THEN rscore=rscore+2
6760 FOR z=1 TO 5
6770 IF k-xhn(z)=15 THEN rscore=rscore+2
6780 NEXT z
6790 FOR z=1 TO 3
6800 FOR a=z+1 TO 4
6810 FOR t=a+1 TO 5
6820 IF xhn(z)+xhn(a)+xhn(t)=15 THEN rscore=rscore+2
6830 NEXT t
6840 NEXT a
6850 NEXT z
6860 IF xhnd(1)=xhnd(2)-1 AND xhnd(1)=xhnd(3)-2 AND xhnd(
1)=xhnd(4)-3 AND xhnd(1)=xhnd(5)-4 THEN rscore=rscore+5:G
OTO 7040
6870 f=0:FOR z=1 TO 2
6880 FOR a=z+1 TO 3
6890 FOR t=a+1 TO 4
6900 FOR z1=t+1 TO 5
6910 IF xhnd(z)=xhnd(a)-1 AND xhnd(z)=xhnd(t)-2 AND xhnd(
z)=xhnd(z1)-3 THEN rscore=rscore+4:f=1
6920 NEXT z1
6930 NEXT t
6940 NEXT a
6950 NEXT z
6960 IF f=1 THEN 7040
6970 FOR z=1 TO 3
6980 FOR a=z+1 TO 4
6990 FOR t=a+1 TO 5
7000 IF xhnd(z)=xhnd(a)-1 AND xhnd(z)=xhnd(t)-2 THEN rsco
re=rscore+3
7010 NEXT t
7020 NEXT a
7030 NEXT z
7040 IF (q=0 AND box=0) OR (q=1 AND box=1) THEN yscore=ys
core+rscore:CLS #1:PRINT #1,"Your hand score=";rscore
7050 IF (q=0 AND box=1) OR (q=1 AND box=0) THEN cscore=cs
core+rscore:CLS #1:PRINT #1,"My hand score=";rscore
7060 IF q=2 AND box=0 THEN cscore=cscore+rscore:CLS #1:PR
INT #1,"My box score=";rscore
7070 IF q=2 AND box=1 THEN yscore=yscore+rscore:CLS #1:PR
INT #1,"Your box score=";rscore
7080 GOSUB 5670
7090 FOR t=1 TO 8000:NEXT t
7100 RETURN
7110 ' **** PRINT INDIVIDUAL CARDS ****
7120 IF n(z)=1 OR n(z)=11 OR n(z)=12 OR n(z)=13 THEN 7710
7130 IF h$=spade$ THEN pipt$=spt$:pipb$=spb$
7140 IF h$=heart$ THEN pipt$=htt$:pipb$=htb$
7150 IF h$=diam$ THEN pipt$=ddt$:pipb$=ddb$
7160 IF h$=club$ THEN pipt$=clt$:pipb$=clb$
7170 IF n(z)=2 THEN 7260
7180 IF n(z)=3 THEN 7310
7190 IF n(z)=4 THEN 7360
7200 IF n(z)=5 THEN 7410
7210 IF n(z)=6 THEN 7460
7220 IF n(z)=7 THEN 7510
7230 IF n(z)=8 THEN 7560
7240 IF n(z)=9 THEN 7610
7250 IF n(z)=10 THEN 7660
7260 LOCATE x,y+2:PRINT "  ";h$;"  "
7270 LOCATE x,y+3:PRINT "     "
7280 LOCATE x,y+4:PRINT "     "
7290 LOCATE x,y+5:PRINT "  ";h$;"  "
7300 RETURN
7310 LOCATE x,y+2:PRINT "  ";h$;"  "
7320 LOCATE x,y+3:PRINT "  ";pipt$;"  "
7330 LOCATE x,y+4:PRINT "  ";pipb$;"  "
7340 LOCATE x,y+5:PRINT "  ";h$;"  "
7350 RETURN
7360 LOCATE x,y+2:PRINT " ";h$;" ";h$;" "
7370 LOCATE x,y+3:PRINT "     "
7380 LOCATE x,y+4:PRINT "     "
7390 LOCATE x,y+5:PRINT " ";h$;" ";h$;" "
7400 RETURN
7410 LOCATE x,y+2:PRINT " ";h$;" ";h$;" "
7420 LOCATE x,y+3:PRINT "  ";pipt$;"  "
7430 LOCATE x,y+4:PRINT "  ";pipb$;"  "
7440 LOCATE x,y+5:PRINT " ";h$;" ";h$;" "
7450 RETURN
7460 LOCATE x,y+2:PRINT " ";h$;" ";h$;" "
7470 LOCATE x,y+3:PRINT " ";pipt$;" ";pipt$;" "
7480 LOCATE x,y+4:PRINT " ";pipb$;" ";pipb$;" "
7490 LOCATE x,y+5:PRINT " ";h$;" ";h$;" "
7500 RETURN
7510 LOCATE x,y+2:PRINT " ";h$;pipt$;h$;" "
7520 LOCATE x,y+3:PRINT " ";pipt$;pipb$;pipt$;" "
```

```
7530 LOCATE x,y+4:PRINT " ";pipb$;" ";pipb$;" "
7540 LOCATE x,y+5:PRINT " ";h$;" ";h$;" "
7550 RETURN
7560 LOCATE x,y+2:PRINT " ";h$;pipt$;h$;" "
7570 LOCATE x,y+3:PRINT " ";pipt$;pipb$;pipt$;" "
7580 LOCATE x,y+4:PRINT " ";pipb$;pipt$;pipb$;" "
7590 LOCATE x,y+5:PRINT " ";h$;pipb$;h$;" "
7600 RETURN
7610 LOCATE x,y+2:PRINT " ";h$;" ";h$;" "
7620 LOCATE x,y+3:PRINT " ";h$;pipt$;h$;" "
7630 LOCATE x,y+4:PRINT " ";h$;pipb$;h$;" "
7640 LOCATE x,y+5:PRINT " ";h$;" ";h$;" "
7650 RETURN
7660 LOCATE x,y+2:PRINT " ";h$;pipt$;h$;" "
7670 LOCATE x,y+3:PRINT " ";h$;pipb$;h$;" "
7680 LOCATE x,y+4:PRINT " ";h$;pipt$;h$;" "
7690 LOCATE x,y+5:PRINT " ";h$;pipb$;h$;" "
7700 RETURN
7710 IF n(z)=1 AND h$=spade$ THEN xt$=st$:xb$=sb$
7720 IF n(z)=1 AND h$=heart$ THEN xt$=ht$:xb$=hb$
7730 IF n(z)=1 AND h$=diam$ THEN xt$=dt$:xb$=db$
7740 IF n(z)=1 AND h$=club$ THEN xt$=ct$:xb$=cb$
7750 IF n(z)=11 THEN xt$=jt$:xb$=jb$
7760 IF n(z)=12 THEN xt$=qt$:xb$=cb$
7770 IF n(z)=13 THEN xt$=kt$:xb$=kb$
7780 LOCATE x,y+2:PRINT "     "
7790 LOCATE x,y+3:PRINT " ";xt$;" "
7800 LOCATE x,y+4:PRINT " ";xb$;" "
7810 LOCATE x,y+5:PRINT "     "
7820 RETURN
7830 ' ************ TITLE **************
7840 BORDER 0:PAPER 0:CLS
7850 PAPER 3
7860 PEN 2:PRINT " "+heart$+heart$+"  ";:PEN 0:PRINT spad
e$+spade$+spade$+"  ";:PEN 2:PRINT diam$+diam$+diam$+"  "
;:PEN 0:PRINT club$+club$+club$+"  ";'1 L
7870 PEN 2:PRINT heart$+heart$+heart$+"    ";:PEN 0:PRINT
 spade$+"    ";:PEN 2:PRINT diam$+diam$+"  ";:PEN 0:PRINT
 club$+club$+club$+club$;'1 R
7880 PEN 2:PRINT heart$+"  "+heart$+" ";:PEN 0:PRINT spad
e$+"  "+spade$+"  ";:PEN 2:PRINT diam$+"   ";:PEN 0:PRINT
 club$+"  "+club$+" ";' 2 L
7890 PEN 2:PRINT heart$+"  "+heart$+"  ";:PEN 0:PRINT spa
de$+" "+spade$+"  ";:PEN 2: PRINT diam$+"  "+diam$+" ";:P
EN 0:PRINT club$+"   ";'2 R
7900 PEN 2:PRINT heart$+"    ";:PEN 0:PRINT spade$+"  "+s
pade$+"  ";:PEN 2:PRINT diam$+"   ";:PEN 0:PRINT club$+"
 "+club$+" ";' 3 L
7910 PEN 2:PRINT heart$+"  "+heart$+" ";:PEN 0:PRINT spad
e$+"   "+spade$+" ";:PEN 2: PRINT diam$+"    ";:PEN 0:PRI
NT club$+"   "; '3 R
7920 PEN 2:PRINT heart$+"    ";:PEN 0:PRINT spade$+spade$
+spade$+"   ";:PEN 2:PRINT diam$+"   ";:PEN 0:PRINT club$
+club$+club$+"  "; ' 4 L
7930 PEN 2:PRINT heart$+heart$+heart$+"  ";:PEN 0:PRINT s
pade$+spade$+spade$+spade$+spade$+" ";:PEN 2:PRINT diam$+
"    ";:PEN 0:PRINT club$+club$+club$+club$;' 4 R
7940 PEN 2:PRINT heart$+"    ";:PEN 0:PRINT spade$+spade$
+"    ";:PEN 2:PRINT diam$+"   ";:PEN 0:PRINT club$+"  "+
club$+" ";' 5 L
7950 PEN 2:PRINT heart$+"  "+heart$+" ";:PEN 0:PRINT spad
e$+"   "+spade$+" ";:PEN 2:PRINT diam$+" "+diam$+diam$+"
";:PEN 0:PRINT club$+"   ";' 5 R
7960 PEN 2:PRINT heart$+"  "+heart$+" ";:PEN 0:PRINT spad
e$+" "+spade$+"   ";:PEN 2:PRINT diam$+"   ";:PEN 0:PRINT
 club$+"  "+club$+" ";' 6 L
7970 PEN 2:PRINT heart$+"  "+heart$+" ";:PEN 0:PRINT spad
e$+"   "+spade$+" ";:PEN 2:PRINT diam$+"  "+diam$+" ";:PE
N 0:PRINT club$+"   ";' 6 R
7980 PEN 2:PRINT " "+heart$+heart$+"  ";:PEN 0:PRINT spad
e$+"  "+spade$+" ";:PEN 2:PRINT diam$+diam$+diam$+"  ";:P
EN 0:PRINT club$+club$+club$+"  ";' 7 L
7990 PEN 2:PRINT heart$+heart$+heart$+"  ";:PEN 0:PRINT s
pade$+"   "+spade$+"  ";:PEN 2:PRINT diam$+diam$+"  ";:PE
N 0:PRINT club$+club$+club$+club$ '7 R
8000 PAPER 0:PEN 2
8010 PRINT "         By Nick Herrick # 1986"
8020 c$(1)=" A "+spade$:n(1)=1:xs$(1)=spade$
8030 c$(2)=" J "+club$:n(2)=11:xs$(2)=club$
8040 c$(3)=" 4 "+diam$:n(3)=4:xs$(3)=diam$
8050 c$(4)=" K "+spade$:n(4)=13:xs$(4)=spade$
8060 c$(5)=" A "+club$:n(5)=1:xs$(5)=club$
8070 c$(6)=" Q "+heart$:n(6)=12:xs$(6)=heart$
8080 PAPER 3
8090 FOR z=1 TO 6
8100 IF xs$(z)=diam$ OR xs$(z)=heart$ THEN p=2 ELSE p=0
8110 h$=xs$(z):y=11:READ x
8120 GOSUB 5110
8130 NEXT z
8140 PAPER 0:PRINT:PRINT:PRINT:PRINT "         PRESS ANY
KEY TO PLAY"
8150 WHILE INKEY$="":WEND
8160 RETURN
8170 ' ********** RULES **************
8180 CLS:LOCATE 1,12:PRINT "Do you want instructions on h
ow to play cribbage? (Y/N)"
8190 i$=UPPER$(INKEY$):IF i$<>"Y" AND i$<>"N" THEN 8190
8200 IF i$="N" THEN RETURN
8210 CLS:PRINT:PRINT:PRINT "        RULES OF CRIBBAGE":PR
INT
8220 PRINT " 1. The pack is shuffled and six cards  are d
ealt to each player."
8230 PRINT:PRINT " 2. Each player has to choose two cards
 to put into the crib or 'box', which at the end of play
counts as an extra      scoring hand for the dealer. Much
 of theskill of playing cribbage lies in       selecting
the box."
8240 PRINT:PRINT" 3. The non-dealer then lifts part of
```

```
the remaining pack. The dealer then     turns over the to
p card of the remainderThis is NOT used in play, but coun
ts    towards the hand score."
8250 PRINT:PRINT:PRINT "     Press any key to continue"
8260 WHILE INKEY$="":WEND
8270 CLS:PRINT " 4. The hands are then played: each     h
and is kept separate for later scoring.Non dealer leads.
During play scoring   combinations are scored and a runni
ng   total of the count of the cards is kept.";
8280 PRINT " This count must not exceed 31 and, if  you c
an play, you must. If you get      exactly 31 you score t
wo points; if you play 'last card' under 31 you score one
 point."
8290 PRINT " If the player cannot go in turn the    other
 player goes if he/she can and goeson playing until 31 is
 reached or she/heruns out of cards."
8300 PRINT " After 31 (or nearest) is reached the   last
to play leads again (or the player with cards left) until
 all cards are    played. After 31 the cards are turned
 over and the count starts from 0 again."
8310 PRINT:PRINT "5. Scoring: press any key to continue."
8320 WHILE INKEY$="":WEND
8330 CLS:PRINT"            SCORE"
8340 PRINT "1. Ace always counts as one. Cards countpip v
alue; Jack, Queen and King count   10 in scoring 'fifteen
s' otherwise as   11, 12 and 13."
8350 PRINT:PRINT "2. Score 'fifteens': ALL combinations o
fcards adding up to 15 score 2 points,   e.g. 10, J and 5
 count 4 points: two    points for 10 and 5, 2 points for
 J and 5. Another 5 would score again twice    totalling
8 for 'fifteens'."
8360 PRINT " Any number of cards may be used to     score
 'fifteen'."
8370 PRINT:PRINT "3. Likewise all combinations of pairs
 score 2 points. This means three of a   kind scores 6 an
d 4 of a kind scores 12.Court cards DO NOT equal 10 for t
his."
8380 PRINT:PRINT "4. Runs: any three or more cards in
 sequence score one for each card. Yet   again all combin
ations count."
8390 PRINT:PRINT "    Press any key to continue"
8400 WHILE INKEY$="":WEND
8410 CLS:PRINT "5. Bonus of 'One for his nob' is scored b
y any hand containing the Jack of the  same suit as the c
ard exposed by the    lift of pack; 'Two for his heels' i
s    scored by dealer if Jack exposed."
8420 PRINT:PRINT "6. During play the score counts on card
splayed. Thus if the computer plays 8 andyou play 7 you s
core 2 for 'fifteen'."
8430 PRINT " Runs count, but the order of play does not;
e.g. cards as follows count as run for last player: 2,4,A
,3 who scores 4   points."
8440 PRINT:PRINT "7. After play the hands are scored in
 the same manner ('One for his nob' is   scored now). The
 turned up card is now  used to add to the score."
8450 PRINT:PRINT "8. The game is won by the first player
 to reach 121 points."
8460 PRINT:PRINT"    Press any key to continue"
8470 WHILE INKEY$="":WEND
8480 CLS:PRINT " Non dealers hand is scored first: if   t
his gives him game in a closely fought battle she/he wins
! The dealer then     scores his/her hand and then the 'b
ox'  as an extra hand."
8490 PRINT:PRINT " If all 4 cards in hand are the same
 suit then score 4 points. If the turn upis also the same
 suit 5 points are      scored. In the box all 5 cards mu
st be  the same suit to score a flush."
8500 PRINT:PRINT " The computer handles all dealing,
 counting and scoring in this game. It   also plays its h
and. It does not cheat  or 'know' what cards you have."
8510 PRINT " When non dealer you will be asked to   lift
the pack by selecting a number: thecomputer exposes the n
ext card in the   pack in the correct way."
8520 PRINT:PRINT "  Phew! That's a lot to take in. Do you
 want to read it again? (Y/N)"
8530 i$=UPPER$(INKEY$):IF i$<>"N" AND i$<>"Y" THEN 8530
8540 IF i$="Y" THEN 8210
8550 RETURN
8560 DATA 3,9,15,21,27,33
8570 DATA 1,1,1,10,7,1,7,10,13,1,13,10,19,1,19,10,25,1,25
,10,31,1,31,10
8580 DATA 1,1,1,10,7,1,7,10,13,1,13,10,19,1,19,10
8590 DATA 1,7,13,19
```

Adventurer's Attic

by Philip Riley

ADVENTURE BASES

As promised in last months mag we will be looking at bases this month. Yes, I know we looked at one a couple of months ago but it is always good to get other ideas on the subject.

So, what should you include in your own base? The answer is simple, just about anything you want to throw in there. But as a guide most bases include the parser and routines for movement. The Listing 1 shows you my standard base for two word input using the cursor keys to move and allowing you to abbreviate your inputs to the first three letters of each word. Now lets take a look at it one step at a time.

Line 90 is the games vocabulary. a1$ is the first words of the input e.g. get, take, leave, drop and use. a2$ are the second words of the inputs and also the single word inputs. These read axe, sword, rope, help and quit. As you can see we have only got the first three letters of each word as this is all we need.

Line 100 sets all the variables used later on to zeros and asks the question "WHAT NEXT".

Line 110 is a loop that asks for an input from the keyboard. If no key is pressed then it loops back on itself.

Line 120 assigns a to equal the ASCII code of the key that was pressed.

Line 130 checks to see if the enter key was pressed. If so then the program jumps to line 170.

Line 140 checks to see if a cursor key was pressed if so then the program jumps to line 280.

Line 150 is the delete routine. It checks to see if you have pressed the delete key and if so deletes the last character from b$.

Line 160 adds the character you typed in onto the end of b$ and prints that character onto the screen.

Lines 170-180 search through b$ to find a space. It then assigns x$ to the first three characters of the first word and z$ the first three letters of the second word. If no space is found the computer then knows that only one word has been typed in and the program jumps to line 220.

Lines 190-220 compares x$ to the words contained in a1$.

Line 210 tells you that it does not understand the word if no match is found.

Lines 220-230 compare z$ against the words in a2$. If no match is found the program jumps to line 210.

Line 240 sends the program to line 260 if only one word was typed in.

Line 250 is an ON-GOTO command that is used to send the program off to the answers to the inputs. If x equal 1,2 then the computer knows that you wish to pick something up so you would send the program off to that particular section.

Line 260 checks to see that the single word input that you have typed in is a word that it will except on its own. In this case z must equal 4 or 5.

Line 270 words the same as line 250 except that it is for single word inputs.

The lines from here on have already been explained in a previous issue so I will not go into them again.

A few things to note about this little proggy. First you don't have to use just the first three letters of the words. You could use just the first two letters of the word, just change all of the STEP commands to STEP 2 and only put the first two letters of each word into a1$ and a2$.

You can add more words to the vocabulary just by tacking them onto the end of the vocabulary strings (a1$,a2$), no other changes are required.

This is all very well if you only want to type in abbreviated words, what happens if you want to type in the full word. Well of course you could not do this with the first listing, but the second listing would be ideal.

As you can see this listing is much shorter but that is mainly

Listing 1

```
90 a1$="gettakleadrouse":a2$="axesworophelqui"
100 a$="":b$="":x$="":z$="":PRINT"What next?"
110 a$=LOWER$(INKEY$):IF a$=""THEN 110
120 a=ASC(a$)
130 IF a=13 THEN 170
140 IF (a>239)AND(a<244)THEN 280
150 IF (a=127)AND(LEN(b$)>0) THEN b$=LEFT$(b$,LEN(b$)-1):
PRINT";:GOTO 110
160 b$=b$+a$:PRINT a$;:GOTO 110
170 x$=LEFT$(b$,3):m=0:FOR t=1 TO LEN(b$):IF MID$(b$,t,1)
=" "THEN m=t
180 NEXT:z$=MID$(b$,m+1,3):IF m=0 THEN 220
190 x=0:FOR t=1 TO LEN(a1$) STEP 3:IF x$=MID$(a1$,t,3)THE
N x=INT(t/3)+1
200 NEXT
210 IF x=0 THEN PRINT"I don't understand try something el
se.":GOTO 100
220 z=0:FOR t=1 TO LEN(a2$) STEP 3:IF z$=MID$(a2$,t,3) TH
EN z=INT(t/3)+1
230 NEXT:IF z=0 THEN x=0:GOTO 210
240 IF m=0 THEN 260
250 PRINT:PRINT:ON x GOTO
260 IF(z<4)OR(z>5)THEN PRINT"I don't understand you.":GOT
O 100
270 ON z-3 GOTO
280 IF (a=240)AND((yp(n)>0)AND(yp(n)<9))THEN n=n+8:GOTO 3
30
290 IF (a=241)AND((yp(n)>4)AND(yp(n)<13))THEN n=n-8:GOTO
330
300 IF (a=242)AND((yp(n)=2)OR(yp(n)=4)OR(yp(n)=6)OR(yp(n)
=8)OR(yp(n)=9)OR(yp(n)=10)OR(yp(n)=14)OR(yp(n)=15))THEN n
=n-1:GOTO 330
310 IF (a=243)AND(((yp(n)>2)AND(yp(n)<7))OR(yp(n)=10)OR((
yp(n)>11)AND(yp(n)<15)))THEN n=n+1:GOTO 330
320 PRINT"You cannot move in that direction.":GOTO 100
330 ON n GOTO
```

Listing 2

```
100 RESTORE:b$="":c$="":b=0:c=0:d=0:z=0:INPUT"WHAT NEXT";
a$
110 FOR t=1 TO LEN(a$):IF MID$(a$,t,1)=" "THEN z=t
120 NEXT:c$=RIGHT$(a$,LEN(a$)-z):IF z=0 THEN 140
130 b$=LEFT$(a$,z-1)
140 FOR t=1 TO 9:READ a$:IF b$=a$ THEN b=t:d=d+1
150 IF c$=a$ THEN c=t:d=d+1
160 NEXT:IF d=1 AND z=0 THEN 180 ELSE IF d=1 OR d=0 THEN
190
170 ON b GOTO
180 ON c GOTO
190 PRINT"I don't understand try something else.":GOTO 10
0
200 DATA help,quit,get,take,drop,leave,axe,sword,rope
```

because I have not included the lengthy input routine or movement routine in this proggy. Not that you cannot include them if you wanted to. Now for a look at the listing a little at a time.

Line 100 restores the data, sets all the variable to zero and asks for an input.

Lines 110-130 check for spaces and sort out the two words (or one as the case may be).

Lines 140-160 reads the data one piece at a time and compares the two words (or one) to see if they match. When all the data has been read the program then decides if one word or two was typed in and sends you off to the appropriate ON-GOTO statement on lines 170 and 180.

Line 190 is the default line if the word or words are not understood.

Line 200 is the data containing the vocabulary.

Now a few points about this little proggy. It may look good but unfortunately it uses a lot of string variables which is bad. Those of you who read my December article on encode/decode will know what I am talking about. For those of you who didn't, using lots of strings in your Amstrad programs is bad as it fills up memory and garbage collections will have to be performed. If you want to know more you will just have to look up your back copies of the Amstrad User and read my December '86 article.

That is all that we have room for this month. Next month I will show you a routine that can do exactly the same as LISTING TWO but it only uses one string variable. Impossible I hear someone shout. Nothing is impossible I reply. We will also be taking a look at sentence input (e.g. take the sword and kill the dragon with it then move north). Clever eh!! I just hope I can get the proggy running by next month now. Until next month keep adventuring and don't give in to the bad guys.

QUESTIONS & ANSWERS

Philip has toiled through the last few weeks collating all the question and answers that have been received in these offices over the last year. He's knocked out the duplicates, rejected the silly ones and now presents his summary. Unfortunately, the list is quite long, so this month we are printing the questions which are still outstanding and will follow up next month with a summary of answers.

QUESTIONS

First of all we have some questions from Ian Byrne who is having trouble with the ***Classic adventure.***

Is it possible to retrieve the trident from the pirate after he has stolen it?

Is it possible to prevent the pirate from stealing anything from you?

What purpose does the second maze serve?

Where is the pirates treasure located?

After you have collected all of the treasures where is the entry to the next stage?

And finally, how many treasures are there (He has collected 14 at present)?

Karla Slack is having trouble finding the running river in the *Hobbit* and cannot get past the spider on the way to the wood elf.

Karla is also offering help on *Neverending story* and *Adventure Quest.* You can write to her at the following address (Of course it would be nice to include a postage stamp with your letter). The address is P.O. Box 201, Springwood 2777, N.S.W.

Next we have two readers who are in trouble in *Seabase Delta.* Jason Houlihan and Chris Pile would like to know. How to make a lens cover and a pancake?

How do you get the disc from the bottom of the lift shaft? and how do you climb upto the platform to disarm the missile?

Steve Alatakis would like to know how to get past the bear in *Heroes of Khan*?

We have had a few questions for Heroes and no answers, how about it out there?

Where is the study in ***Bastow Manor***? cries Jurgen Peltz.

When you are ***Hitch-hicking*** around the galaxy how do you open the hatch when you have landed on Magrathea (I thought this planet was only legendary but it would appear that David Watt has found it).

Another problem from Steve, is it possible to operate the teleportation device in ***Message from Andromeda*** and if so, how?

John Dawson would like to launch the tyme machine in ***Knight Tyme*** if anyone can help?

Allan Mearns is having trouble giving the answers to the questions asked by Graunch (I think that is the right name) in ***Return to Eden.***

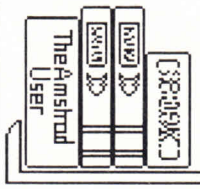
He would also like to know how to get to other levels besides the reception and engineering levels in ***Necris Dome.***

Jason Houlihan is having trouble accumulating money in ***Bugsy.*** It appears that no matter what he does he always gets caught by the bunny in blue.

If you have the answer(s)- it doesn't matter how simple you think it is (or was when you solved it) - please drop a line to Philip. He needs a daily "fix" of letters to keep him from going into a loop!

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 6128 Computing	Out of Print	
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 464/664/6128 Handbook	$ 9.25	$ 9.95
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
Programming the Amstrad CPC 464	$17.95	$19.95
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	$25.95	$27.95
Mastering the Amstrad PCW 8256/8512	$29.50	$32.25
Practical Amstrad Word Processing	$25.95	$27.95

CP/M TITLES

CP/M - The Software Bus: a programmer's companion: with CP/M+	$29.95	$32.25
Intro. to CP/M Assembly Language	$32.35	$35.95

NEW TITLES TO LIST

For CPCs

Using DR Logo on the Amstrad (under CP/M 2.2)	$35.95	$37.95
LOGO Pocketbook	$16.95	$17.95

For PCWs

Practical Logo on the Amstrad (under CP/M+ so OK for 6128 too)	$25.95	$27.95
Program your PCW	$25.95	$27.95
LocoScript Pocketbook	$16.95	$17.95

For PC1512s

Basic 2 - 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

Please add $5.00 post and packing to all book orders

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

For CPCs	Tapes	Discs
Aftershock	$39.95	$57.50
Air Combat Emulator	$39.95	$49.95
Activator	$37.50	$49.50
Bombscare	$9.98	-
Booty	$9.98	-
Dr Who	$34.95	$54.95
Druid	$29.95	$49.95
Elevator Action	$32.50	$44.95
Family Fun Pack 1	$19.95	$29.95
Formula 1 Simulator	$11.98	-
Future Knights	$34.95	$49.95
Hyperbowl	$11.98	-
Lord of the Rings	-	$44.95
Magic Maths	$9.98	-
Mini Office II	-	$79.95
Necris Dome	$9.98	
Sentinel	$34.95	-
Sport of Kings	$11.98	-
Storm	$9.98	-
Terra Cognita	$9.98	-
Tomahawk	-	$49.95
Trailblazers	$34.95	$49.95
For PCWs		
Aftershock	-	$57.50
Tomahawk	-	$57.50
Lord of the Rings	-	$44.95
MASTERFILE 8000		$109.00

All prices include postage.

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 30 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

Tapes or Discs not suitable for PCWs

THE AMSTRAD COMPUTER USER (English)

Issues 3 to 7 each	$5.00
Issues 12 to 19 each	$5.50
Issues 21 to 31 each	$6.00

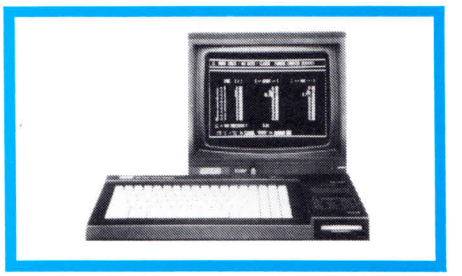
Note: Issues 8 to 11 (Jul '85 to Oct '85) and Issue 20 (Jul '86) are 'Out of Print'

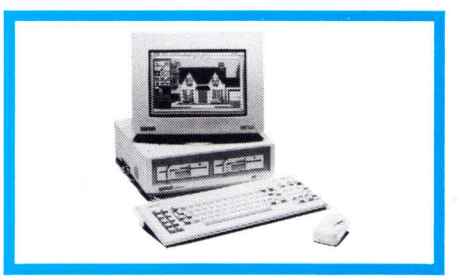
Prices include postage on all Back copies, discs and tapes.

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.

DMP3000 PRINTER

PC Compatible Dot Matrix Printer

The DMP3000 is yet another milestone in the AMSTRAD range of low-cost high-performance computer products.

It combines the versatility of an industry standard software instruction set with AMSTRAD's expertise in quality engineering.

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 DMP3000 to operate directly with most software, including word processors, spreadsheets and graphics programs.

Features

Front loading of paper

Low noise

Low power consumption

IBM graphics character set

Usable with friction or tractor feed paper

Epson compatible control codes

Over 100 different typeface combinations including NLQ (Near Letter Quality)

Computer peripherals by AMSTRAD