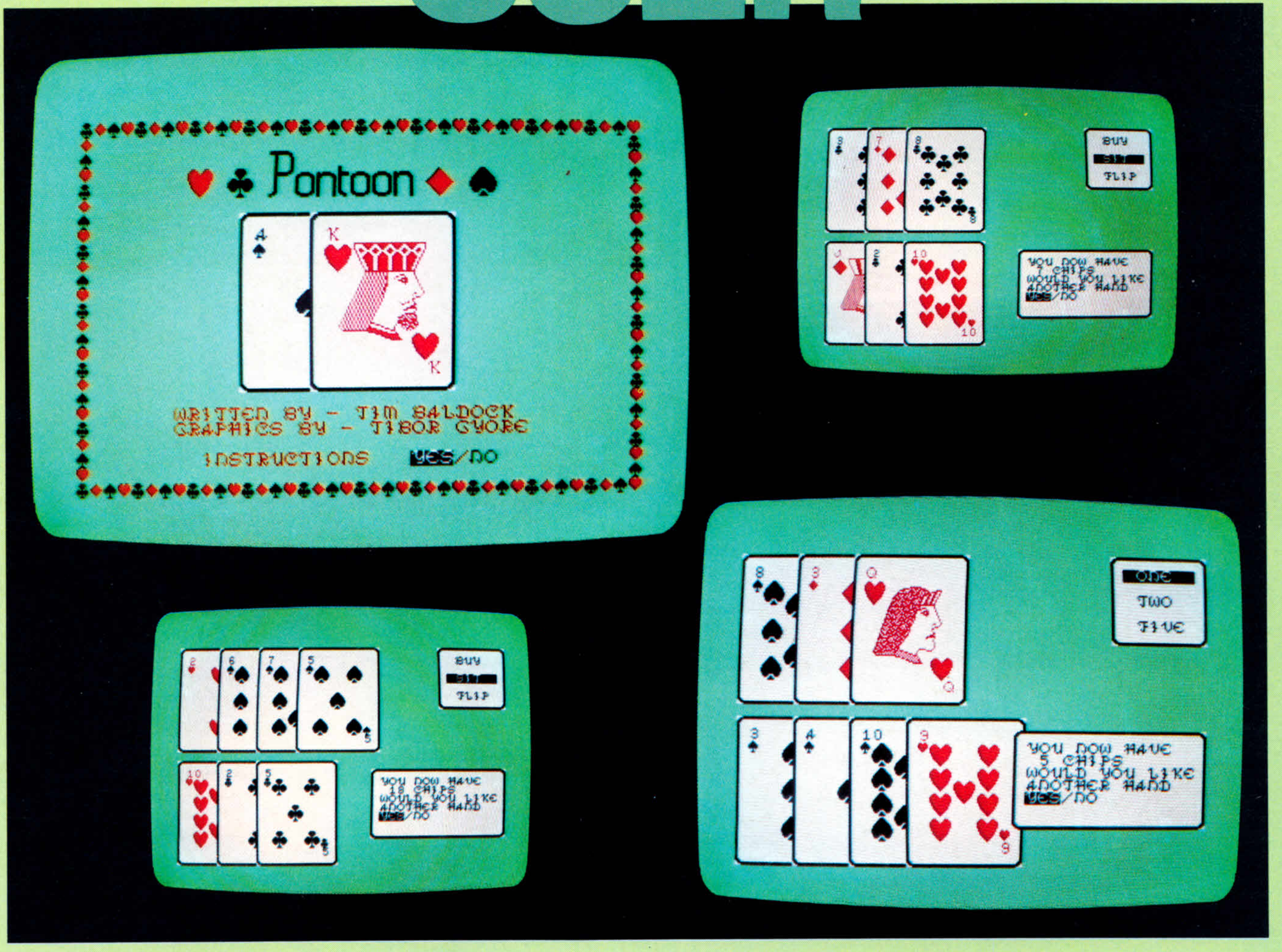
e18

THE AMSTRAD USER

Issue No. 18 $3.50 July 1986

PCW Range Software Supplement



YOU NOW HAVE 7 CHIPS WOULD YOU LIKE ANOTHER HAND YES/NO

YOU NOW HAVE 18 CHIPS WOULD YOU LIKE ANOTHER HAND YES/NO

YOU NOW HAVE 5 CHIPS WOULD YOU LIKE ANOTHER HAND YES/NO

- THE ULTIMATE VERSION OF PONTOON
- TRADING GAME FOR YOUNGSTERS
- BUILD A SPRITE - MORE ON CP/M
- USER GROUP INFORMATION

FOR THE NOVICE & EXPERIENCED USER

Registered by Australia Post – Publication No VBP7017

bBBBBBOOKSSSSs

Title	Subscriber Price	Normal Price
Advanced User Guide	$ 19.75	$ 21.95
Adventure Games for the Amstrad	$ 25.15	$ 27.95
Amstrad Compendium	$ 20.65	$ 22.95
Amstrad Computing	$ 21.55	$ 23.95
Amstrad Games Book (Pitman/Ramshaw)	$ 14.35	$ 15.95
Amstrad Games Book (Melbourne House)	$ 20.65	$ 22.95
Amstrad Pentacle Adventure Creator	$ 8.05	$ 8.95
Amstrad Users Omnibus - 464/664/6128	$ 17.95	$ 19.95
Basic BASIC	$ 11.45	$ 12.75
Basic Programming on the Amstrad	$ 22.45	$ 24.95
Brainteasers for the Amstrad	$ 19.75	$ 21.95
Childs' Guide to the Amstrad Micro	$ 9.85	$ 10.95
Disc System, The Amstrad CPC 464	$ 25.15	$ 27.95
Dynamic Games for the Amstrad	$ 17.95	$ 19.95
Exploring Adventures on the Amstrad	$ 21.55	$ 23.95
Filing Sytems And Databases for the Amstrad	$ 26.95	$ 29.95
Games and Graphics Programming - 464/664/6128	$ 26.95	$ 29.95
Graphics Programming Techniques	$ 22.45	$ 24.95
Ins and Outs of the Amstrad	$ 20.65	$ 22.95
Machine Code for Beginners on the Amstrad	$ 17.95	$ 19.95
Machine Language for the Absolute Beginner	$ 20.65	$ 22.95
Making Music on the 464/664	$ 19.75	$ 21.95
Master Machine Code on your 464/664	$ 19.75	$ 21.95
On the road to Artificial Intelligence	$ 17.95	$ 19.95
Pitman's First Book of Amstrad Games	$ 11.65	$ 12.95
Practical Programs for the 464	$ 21.55	$ 23.95
Programming the Amstrad CPC 464	$ 17.95	$ 19.95
Ready made Machine Language routines - 464/664	$ 20.65	$ 22.95
Structured Programming on the 464/664/6128	$ 26.95	$ 29.95
Whole Memory Guide	$ 26.95	$ 29.95
Working Amstrad	$ 17.95	$ 19.95
Writing Adventure Games on the 464/664	$ 20.65	$ 22.95
Your first Amstrad Program	$ 20.65	$ 22.95
40 Educational Games for the Amstrad	$ 19.75	$ 21.95
60 Programs for your Amstrad	$ 19.75	$ 21.95

How to Order

Send a list of the titles and quantities you require along with a cheque for the total **plus** $5.00 postage and packing (regardless of the quantity you order) to:

STRATEGY PUBLICATIONS
Suite 1 , 245 Springvale Road, Glen Waverley, Victoria, 3150
Bankcard or Mastercard accepted (not Visa) on (03) 233 9661

THE AMSTRAD USER

Issue No. 18
July 1986

Editorial 2

Letters 3

Nationwide User Groups 6

Orangeade Stall
A trading game for youngsters
Adapted by A.F. Ryan 9

The Learning Centre
CP/M Explored - Part Four
by Shane Kelly 15

Build a Sprite
by P.T. Crowe (SCAUG) 17

Newcastle Database
from the Newcastle Amstrad User's Group 20

Adventurer's Attic
(including some answers for 'lost souls')
by Philip Riley 24

Characters in Colour
by Peter and Michael Doutch 26

Using ams-FORTH with disc drive(s)
by Petr Lukes 29

A definitive Black-Jack
from Tibor Gyore and Tim Baldock 31

Impressions of PC '86
from Julian Tipper 39

For Tape Subscribers, the programs can be found at these approximate positions:
Side 1: ORANGEADE - 11, APPLE1 - 108, APPLE2 - 123
Side 2: NEWCASDB - 11, SPRITE - 82

All enquiries and contacts concerning this Publication should be made to The Amstrad User, Suite 1, 245 Springvale Road, Glen Waverley, Victoria 3150, Australia. [Telephone: (03) 233 9661].

The Amstrad User is published each month by Strategy Publications. Reprinting of articles published in The Amstrad User is strictly forbidden without written permission. All rights reserved. Copyright 1986 by Strategy Publications.

The single copy price of $3.50 is the recommended retail price only.

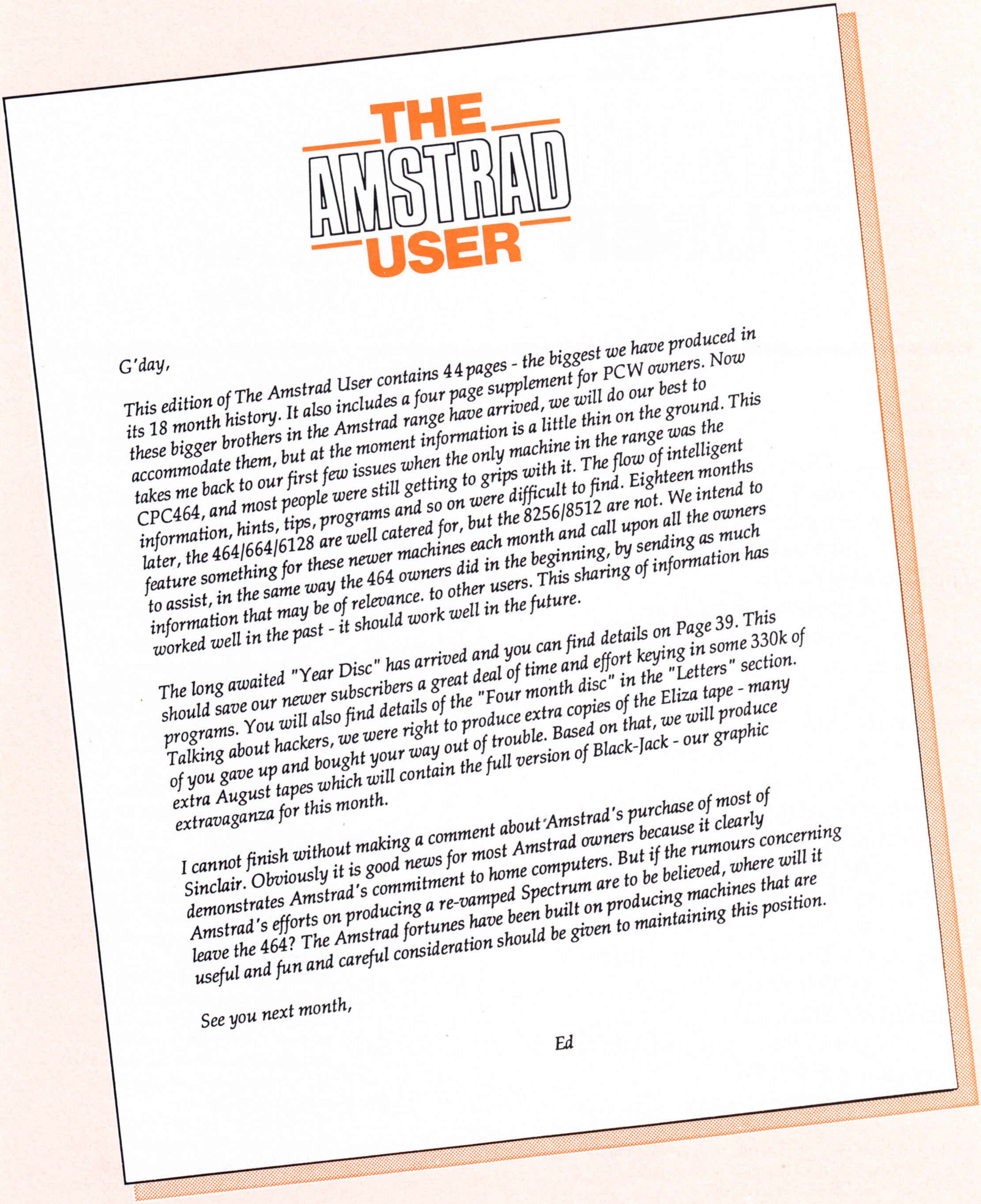
The subscription rate (for Australia only) is $35.00 for 12 issues of the magazine only, or $75.00 for 12 issues of the magazine plus tape containing all programs appearing in that issue. Postage is included in the above prices. Overseas prices available upon application. (New Zealand, PNG, Solomon Is, Vanuatu add $12 airmail)

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, Cartoons-$5.00 and a rate of $10.00 per page for programs, articles etc.

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



THE AMSTRAD USER

G'day,

This edition of The Amstrad User contains 44 pages - the biggest we have produced in its 18 month history. It also includes a four page supplement for PCW owners. Now these bigger brothers in the Amstrad range have arrived, we will do our best to accommodate them, but at the moment information is a little thin on the ground. This takes me back to our first few issues when the only machine in the range was the CPC464, and most people were still getting to grips with it. The flow of intelligent information, hints, tips, programs and so on were difficult to find. Eighteen months later, the 464/664/6128 are well catered for, but the 8256/8512 are not. We intend to feature something for these newer machines each month and call upon all the owners to assist, in the same way the 464 owners did in the beginning, by sending as much information that may be of relevance. to other users. This sharing of information has worked well in the past - it should work well in the future.

The long awaited "Year Disc" has arrived and you can find details on Page 39. This should save our newer subscribers a great deal of time and effort keying in some 330k of programs. You will also find details of the "Four month disc" in the "Letters" section. Talking about hackers, we were right to produce extra copies of the Eliza tape - many of you gave up and bought your way out of trouble. Based on that, we will produce extra August tapes which will contain the full version of Black-Jack - our graphic extravaganza for this month.

I cannot finish without making a comment about Amstrad's purchase of most of Sinclair. Obviously it is good news for most Amstrad owners because it clearly demonstrates Amstrad's commitment to home computers. But if the rumours concerning Amstrad's efforts on producing a re-vamped Spectrum are to be believed, where will it leave the 464? The Amstrad fortunes have been built on producing machines that are useful and fun and careful consideration should be given to maintaining this position.

See you next month,

Ed

Letters

All letters published in this section attract a payment of five dollars. Any correspondence should be addressed to The Editor, The Amstrad User, Suite 1, 245 Springvale Road, Glen Waverley, Victoria 3150.

I have recently become the owner of a PCW 8256, and an excellent machine it is for word processing and general computing. On reading through the two manuals provided, I now find myself frustrated by being unable to program it to produce graphs, as it does not recognise the various plot commands such as "LOCATE", "MOVE", "DRAW", etc.. I know that the machine will produce graphics. One only has to try DR LOGO to see that. Are there any simple, or even complicated, commands or routines that can be entered to make use of the GSX graphics support utility? Alternatively, is it possible to use a different Basic that will provide these facilities?

A feature that is not mentioned in the instruction manuals is how to obtain a printout of the pictures generated by DR LOGO. At least, I cannot find any reference to it. I have now discovered that by pressing EXTRA + PTR, whatever is on the screen appears on the paper in the printer. This may be of some use to readers who are still looking for a way out, other than trying some of the screen dumps that have been published recently. By the way, do any of them work on the PCW8256?

The main drawback of this simple built in screen dump appears to be that the size of the screen appears on paper as a block 120mm x 80mm. I am still working on that problem.

Arnold Goldman, North Dandenong, Vic

As you will be aware, I am a NSW contact point, but as yet have not been successful in interesting sufficient owners to form a group. They are very thin on the ground in my area!

Most of those who have contacted me are 'long distance' telephone callers so the necessary numbers of eye-ball to eye-ball attendees (5 or 6 at least) at meetings, without which the group would be still-born, are not in sight.

However a point I would like to make concerns HELP. Your Editorial in the April issue advised (quite understandably) that such help would be of necessity restricted in the future. May I suggest that the preamble to the User Group Contact list less forcefully indicate that such contacts "should NOT be viewed as a problem solving service".

I, for one, would be most happy to help (and in fact do) within the limits of my knowledge and experience. Callers have been somewhat "off-put" with the existing qualification - and approach me rather diffidently as a result. Perhaps an asterisk and a footnote may fill the bill if you feel others may not wish to HELP. Surely this can only enhance the possibility of effective Group information.

Chas Fletcher, Toongabbie, NSW

Point taken, but the reason for making the 'forceful' statement is due to a number of requests from people whose names appear in the list. You see, most are (or were) *pretty new to computers and feel that by helping to establish a group in their area will save them and other locals much 'wheel-inventing'. Unfortunately, many calls did not relate to setting up a group - one I recall was received at 2.00 am concerning a Syntax Error - so the abuse had to stop. We will certainly consider your point about identifying those people who, like yourself, are happy to answer queries. Perhaps they would make themselves know to me - Ed.*

My wife and I are keen players of AMTHELLO and would very much appreciate a copy of the program for two players as mentioned on Page 3 of Issue 16.

Eric Tytherleigh, Kelmscott, WA

Mike Nicolai provided only one copy of a complete two player program and this was passed on to G.J. de Vos. If you write to Mike, though these offices, we will pass the letter on to him.

I am the owner of a CPC464, one of the best computer decisions I have made, but I have a problem with the printer I have chosen to keep. I say 'keep', because I previously owned a Dick Smith VZ200 which performed very well but the cost became a major factor.

To update from the VZ200 to the VZ300 and get 64k when it originally came on the market was only marginally cheaper than the CPC464. When buying the CPC464, I retained my Seikosha GP100 printer. Although not the fastest or best printer on the market, it is adequate.

I have no problems with Tasword and can use the special printer control characters for extended printing (CHR$(14)-CHR$(15)). The printer also works well with other program such as databases.

I have been told the GP100 is the base for many printers including the DMP1 and in fact they are the same and should perform the same functions. If this is true, can someone help with graphics on the GP100 and through this column enlighten me further?

N. van Kempen, Wardell, NSW

I was elated to find in the March issue just what I had been looking for - a system of keeping track of my disc files. I am, of course, referring to the disc cataloguer by Petr Lukes. However, after typing it in, and double and triple checking, it would not work correctly. I have two defects. It keeps giving a READ FAIL message when trying to create a log. When using drive A only, the disc drive keeps rattling when the log create prompt is answered and the only way to stop this is to switch off. When drive B is used it does not happen. Nevertheless, regardless of which drive is used, the READ FAIL message is always there. I thought my problems were solved when, in the May issue, I found an error correction. Sorry, but the extra full-stop did not help. To add further to my dilemma the correction showed (2,8) and (2,9) whereas in the original it showed (z,8) and (z,9). Which is correct?

I was also very interested to read on Page 4 of Issue 14, a reference to supplying all TAU listings on disc. I would certainly be a candidate for that. However, if this project fails, could I change my subscription to include tapes at this time and receive back copies, including last year's?

G. Hauer, Morphett Vale, SA

Hopefully your problems have now been solved with further corrections to Disc Cataloguer published in the June issue. The first correction should have read (z,8) and (z,9).

Petr Lukes writes - "I am very grateful to Mr. Webber of Galston, NSW (Letters, June '86) for pointing out two further bugs. An empty file is certainly a valid situation and should be catered for. I now remember crashing on a non-executing inner loop (as in line 470:next f,s) at some stage. Instead of correcting all my programs, I hastened to write a letter to the Editor, detailing it as yet another Basic bug (which it is since it is a legal construct, and the interpreter should be able to handle all possible cases). I never got round to fixing my masters. There is another correction, which will avoid losing the file name (stored in x$) during SEARCH:

```
650  PRINT#pr,MID$(z,2,8)"."
MID$(z,10,3)RIGHT$("  "+STR$
(VAL("&"+MID$(z,13)))+"k",5)
,:RETURN
```

Unfortunately, I never did get around to cataloguing my discs after the first test-run, so the program was not adequately tested. I regret any inconvenience I have caused."

Elsewhere in this issue you will find that we have taken the plunge and are now offering a disc full of the programs/utilities etc. which appeared in Issues 1 to 12 inclusive. To buy all the back copies on tape would have cost 12 x $5 = $60. Buying the disc at $50 will save you $10.

We will also be making available three discs a year, each containing the previous four months published programs with, perhaps, some free public domain CP/M software. If all goes to plan this will happen in October when two discs will be released, one containing Issue 13 to Issue 16 and the second Issue 17 to Issue 20. A third disc will be released in February 1987 covering Issue 21 to 24. The cost of each disc will be $22. A disc subscription is much more difficult to operate as the start or renewal would have to be in line with the production times of the discs. We are still trying to think this one out and if we come to any firm conclusions we will, of course, provide full details.

I am hoping that someone can help me in relation to using my Amstrad 6128 to access Viatel and other bulletin boards. The advertising brochure for Amstrad computers states "the Amstrad RS232C interface provides a complete solution to your communications needs. Built into the unit is a ROM which contains the software you need to connect your CPC to other devices" etc. Further down it goes on to say "using the RS232 interface you can convert your CPC into a Viewdata terminal. With a suitable modem and phone, you can enter the Prestel database" etc. which I understand to be similar to our Viatel.

The first question I have is, will that interface be adequate to access Viatel and if not, from where can the appropriate software be purchased?

Before buying the computer I was told that I could use the Amstrad for just such a purpose, and because I thought the Amstrad 6128 was such good value for money, this was the final deciding factor. If I had been told that it couldn't access Viatel, I probably wouldn't have bought one, due to my occupation and the use to which I wish to put the computer.

Finally, I think your User Magazine is great, keep up the good work.

N. McMartin, Nhill, Vic

Thank you for the compliment. We asked our resident 'technocrat' Shane Kelly to provide an answer. He says that "the Amstrad RS232 interface has software in ROM to access the British Prestel service which is almost identical to our Viatel system. After the baud rate has been set (see Page 18 of the RS232 manual) you need to type | *PRESTEL. The screen will clear to black. Dial Viatel and wait until answered. Key in your access code and then use the numeric keypad for all functions. If you are using Viatel to download software, you may be unsuccessful as some British software only allows downloading of text source files. I don't know if this is the case with Amstrad software. Perhaps someone out there can inform us?"*

I wonder if you can assist me? In February 1986 I despatched my copy of Expense Manager on tape to Gameworx Software who were offering to convert it to disc for $10. I also ordered a game at the same time. It took several letters and three telephone calls to finally get an acknowledgement that my second letter had been received and that the order had just been sent. It all sounds a bit suspicious as I am still waiting for the software and believe you have mentioned a problem with Gameworx in a previous issue.

Phil Maynard, Katherine, NT

Our previous reference to Gameworx was in Adventurer's Attic (Issue 15) where Patrick Cahill had sent $1 for a coded hint book for King Solomon's Mines - Part 1 and had not received a reply. In fact he should have sent $2 and a stamped and return addressed envelope. When we contacted Gameworx they advised that the code book forKSM - Pt.1 had always been $2 for the Amstrad version. Older versions for other machines were $1 but these have not been available for some time. A check has been instigated to see if Patrick's order was received. On the other point, they advise that Phil's second letter arrived before the first, and in fact two copies of the required software had now been despatched. Just to make sure there were no other problems, the second despatch was sent by certified mail. Hopefully, Phil should have received it by now.

NATIONWIDE USER GROUPS

This month we welcome another three groups to the ever growing list of established Amstrad user groups.

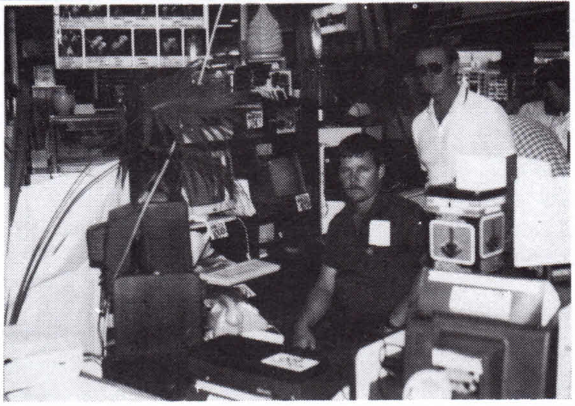
The **Townsville Amstrad User Group** who we knew were in existence have now revealed themselves. In fact the club was formed in October last year and now has over 50 members. A club day was held in May (in conjunction with Mitsubishi-Electric - AWA and Chandlers K-Mart Plaza store) from 8.00 a.m. to 9.00 p.m. Advice and demonstrations were provided on Viatel, the AMX Mouse, the PCW 8256 and the new DMP 2000. The day was such a success that they have been invited to repeat the display next year.

As an aside, the Weipa Amstrad User's Club also held a 'fun day' recently and raffled a green screen CPC464 quite profitably it appears.

Am-User's is now established in North Ryde. Actually, we did make a small mention in the October edition only to say that we had lost the piece of paper with their details! We are happy to re-establish contact. This group has negotiated discounts with several stores for members of the group.

Finally, the Wagga Wagga Amstrad User Group has been launched which we are sure will please the users in that area especially as the organiser is a relatively old hand at Amstrad computers!

All the details of the new and existing groups can be found in the next few pages.



Two members of the Townsville Amstrad User Group ready to lick the opposition (andthe Ice Cream!)

WESTERN AUSTRALIA

AMSWEST, Perth

President:	Tony Clitheroe	(09 275 1257)
Vice President:	Steve Cushnahan	(09 445 2062)
Secretary:	Mrs. P.T. Ardron	(09 361 8975)
Treasurer:	John Firth	

Regular meetings take place at a venue in Shenton Park on the first and third Tuesdays of each month starting at 7.30p.m.

SOUTHSIDE AMSTRAD USER CLUB

President:	John Marshall	(09 390 7335)
Secretary:	Debbie Hoffman	(09 459 8702)
Treasurer:	Eric Tytherleigh	(09 390 8865)
Librarian:	Roy Depurouzel	(09 457 9026)

SAUC meets from 7.00 p.m. every 2nd and 4th Tuesday of each month at Gosnells Scout Hall on the corner of Verra and Corfield Streets, Gosnells. All meetings are socially orientated with a minimum of business matters and can include software and hardware demonstrations, discussions or lectures which all prove to be helpful for beginners or advanced users alike. The club has an extensive library of tapes, discs, magazines etc. and discounts have been obtained from most local dealers and are available to financial members. Contact can be made with any of the above officers or by writing to The Secretary, Southside Amstrad Users Club, PO Box 324, Gosnells, WA 6110.

ROCKINGHAM-KWINANA AMSTRAD USER GROUP

President	Bob Harwood	
Vice-President	Keith Gaisford	
Treasurer	Rob Macilroy	
Secretary	Keith Saw	(095 27 6519)

This new group meets at 7.30 p.m. at the Cooloongup Primary School, Westerly Way, Cooloongup (Rockingham), every second Wednesday. Further details can be obtained from Keith Saw on the above number or by writing to 29 Millgrove Avenue, Cooloongup, WA 6168.

SOUTH AUSTRALIA

AMSTRAD COMPUTER CLUB INC. (SA)

President:	Chris Sowden	(08 295 5923)
Vice Pres:	Frank Matzka	(08 382 2101)
Tréasurer:	Les Jamieson	(08 356 9612)

The group now meets each Tuesday at the Church Hall, 15 Clayton Avenue, Plympton between 6.30 p.m. and 9.00 p.m. Any of the above officers can be contacted for further details and correspondence can be addressed to PO Box 210, Parkholme, 5043.

NORTHERN AMSTRAD COMPUTER CLUB (SA)

Annexed to the Amstrad Computer Club Inc. (SA), this group meets every second Wednesday of the month at the Scout Hall, Bagsters Road, Salisbury from 7.00 p.m. Further details can be obtained from Richard McVey on (08) 258 3400 or any officer of the main club above.

PORT PIRIE AMSTRAD USER GROUP

President:	Rick Cable	(086 32 5967)
Treasurer:	Dave Green	(086 32 6834)
Secretary:	Keith Partridge	(086 32 3919)

The group meets at 7.30 p.m. every first and third Monday of each month at the Way Inn Coffee Lounge, Ellen Street, Port Pirie City Centre. Meetings are well attended with members from Pt. Broughton, Warnertown and even Burra. For further details contact Rick Cable who will advise on the benefits of belonging to this group.

API COMPUTER GROUP

Contact:	Mike Denieuwe	(08 225 5995)

The Australian Post-Tel Institute has a number of computer groups, almost entirely dedicated to Amstrads. Monthly meetings are held in:

Blair Athol	- 2nd Tuesday of the month at 5.30 p.m.
Elizabeth	- last Tuesday of the month at 5.30 p.m.
St. Marys	- 3rd Tuesday of the month at 5.15 p.m.

with a central meeting place in the City at various times during the month. Membership is $15 per year. For more details, contact Mike Denieuwe during office hours on the above telephone number, or on 08-297 8500 after hours.

AMSOUTH AMSTRAD USERS GROUP

President	Geoff Martin	(08 384 4796)
Treasurer	Bob Bleachmore	(085 56 2048)
Secretary	Andrew Chapman	(08 382 1716)

As the name suggests, this group has been established to cater for Amstrad users living south of Adelaide with the emphasis on family involvement. They meet every second Wednesday of each month at the Seaford Anglican Church Hall which is on the corner of Oldham and Commercial Road, Seaford. Meetings commence at 7.30 pm.

VICTORIA

WESTERN AMSTRAD USER GROUP

The meetings are held on alternate Tuesdays (to allow for shift workers) from 6 p.m. at the Fairburn Kindergarten, Fairburn Road, Sunshine.

CENTRAL AMSTRAD USER SOCIETY

President:	Rimon Russo	(03 749 4365)
Vice-Pres:	Dennis Whelan	(03 367 6614)
Treasurer:	Fred Gillan	(03 598 5780)
PR Officer:	John Holmes	(03 434 1607)

Meetings are held twice a month in the 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. All meetings are conducted in a friendly atmosphere - families are welcome.

EASTERN AMSTRAD USER GROUP

President:	Tony Blakemore	(03 878 6212)
Secretary:	Barry Fredrickson	(03 846 1340)
Treasurer:	Ron Dunn	(03 277 7868)

Regular meetings are held on the first Sunday of every month at St. Ninian's Church Hall, cnr. McCracken Avenue and Orchard Grove, South Blackburn. The group organises tutorials for beginners as well as lectures and demonstrations. Proceedings commence at 2.00 p.m. Their postal address is PO Box 279, Heidelberg, Vic 3084.

SOUTHERN AMSTRAD USER GROUP

President:	Mike Prezens	(03 781 2158)
Secretary:	Martin Scragg	(059 78 6949)
Treasurer:	Herman Schutte	(03 799 2482)

Meetings are held on the third Tuesday of every month (except December when it is the first Tuesday) from 7.30 p.m. to 10.30 p.m. The venue is the Senior Campus at John Paul College, Frankston.

NORTHERN AMSTRAD USER GROUP

Contact:	Brian Ellis	(03 469 4425)

This group caters for users in the Preston/Coburg areas. Meetings are devoted to learning more about computers and consist of lectures, demonstrations and practical workshops of projects such as modems, expansion busses etc. Arcade games are banned from meetings. The Group is privately funded and there are no membership fees.

SALE AMSTRAD GROUP

Organiser:	Alan Harris	(051 44 1454)

The Group meets informally every Thursday night from 7.00p.m. at the Sale Neighbourhood House in Leslie Street. In addition, small group tutorials are held twice a month. Contact Alan Harris for further details.

GEELONG AMSTRAD USER CLUB

President:	Ron Butterfield	(052 50 2251)
Vice-President:	Arthur Pounsett	(052 78 2160)
Secretary:	Ross Bennett	(052 44 1556)

This club now meets at South Barwon Community Services Centre, 33 Mount Pleasant Road, Belmont on the first Wednesday of every month, starting at 7.30 p.m.

MACEDON RANGES AMSTRAD USER GROUP

Contacts:	Wayne Urmston	(03 744 2719)
	Ken McMaster	(054 22 2620)

Covering the Gisborne, Sunbury, Kyneton and Woodend areas, this group meets in the Admin. Building of Flexdrive Industries on the second Wednesday of each month from 7.30 p.m.

MARYBOROUGH AMSTRAD USER CLUB

President:	Chad Banfield	(054 68 1351)
Treasurer:	Brendan Severino	(054 61 3191)
Secretary:	Paul Clark	(054 61 2135)

This group consists of a number of students and teachers from Maryborough CCC. Meeting take place each Friday from 12.10 p.m. to 12.45 p.m.

ACT

CANBERRA AMSTRAD USER'S GROUP

Convenor:	Arthur McGuffin	(062 31 9437)
Secretary:	Peter Stehn	(062 81 0258)
Treasurer:	Phil Rogers	(062 41 3039)

The group meets at 7.30 p.m. on the first Wednesday of each month in the Seminar Room of the Oliphant Building at the Research School of Physical Science, Australian National University.

NEW SOUTH WALES

JUBOL AMSTRAD USER GROUP of COFFS HARBOUR and DISTRICT

Contacts:	Bruce Jones	(066 52 8334)
	Jim Owen	(066 55 6190)

The "JUBOL" User Group is currently a small group covering the Coffs Harbour area. They meet informally on the first Friday of each month. Some members attend external DP courses and it is hoped that the group as a whole will shortly be embarking upon internal machine code tutorials.

SYDNEY AMSTRAD COMPUTER CLUB

President:	Raja Vijayenthiran	(02 519 4106)
Secretary:	Reed Walters	(02 560 9487)
Treasurer:	Jim Chryss	(02 327 7872)

Junior Rep:	Daniel Story

This club, which services the inner City area, now meets in the Alexandria on the first Saturday of every month from 2.30 p.m for a normal club meeting and on the third Saturday of each month for the purposes of programming tutorials only. A permanent meeting place has not yet been established so prospective members or visitors are advised to contact one of the officers listed above.

NEWCASTLE AMSTRAD USER GROUP

President	John Harwood	(049 48 5337)
Sec/Treasurer	Erica Harwood	(049 48 5337)

Meetings take place on the first Tuesday of each month at Kotara Public School, Park Avenue, Kotara. A Newsletter is produced each month for members. Interested parties should contact John or Erica on the above number.

AM-USER's (North Ryde)

Contact:	Lawrence Walters	(02 888 1898)

The primary objective of the group is to enhance member's knowledge on both hardware and software available and on the Amstrad computers themselves. Meetings are held in the Meeting Room at 2 Leisure Close, North Ryde from 7.30 p.m. on the first Tueday of each month. Discounts for members have been established at several stores in the area.

WAGGA WAGGA AMSTRAD USER GROUP

Contact:	J.D. Tipper	(069 21 3697)

This recently formed group meets very informally with all interests catered for. Meetings take place on Saturday afternoons - all are welcome. For more information contact Julian Tipper on the above phone number or write to 36 Wills Place, Kapooka, NSW 2661.

QUEENSLAND

BRISBANE AMSTRAD COMPUTER CLUB

President:	Paul Witsen	(07 371 9259)
Secretary:	John Roberts	(07 283 3349)
Tech. Editor:	Peter Walker	(07 371 4286)
Tech. Librarian:	Peter Golledge	(07 376 1651)

Meetings are held on the first Tuesday of each month at Junction Park State School, Annerley starting at 7.30 p.m. in Room 15a.

SOUTHSIDE AMSTRAD USER GROUP (QLD)

President:	Michael Toussaint	(07 200 5414)
Secretary:	Sylvia Wilson	(07 209 1947)
Treasurer:	Col Liebke	(07 200 5555)

Meetings take place every third Saturday of the month at Waterford West State School, Waterford West starting at 2.00 p.m. The group was formed to service the southern outskirts of Brisbane and membership consists of beginners to advanced programmers. Demonstrations of various hardware and software packages are given at meetings and the formation of smaller instruction groups for personal attention is under way. A BASIC programming instruction course is held fortnightly.

WEIPA AMSTRAD USERS CLUB

President	Andrew Seaborn	
Vice-President	Dave Wootton	
Treasurer	Frances Casey	
Secretary	Gary Chippendale	(070 69 7448)

This new group has already had a few meetings at Noola Court in Weipa. Prospective members should contact Gary on the above telephone number or write to 15 Noola Court, Weipa, 4874.

PENINSULA AMSTRAD CLUB

President	Ivan Dowling	(07 269 8795)
Treasurer	Keith Johnston	(07 203 2339)
Secretary	Tracie Payne	(07 267 6645)

The aims and objectives of this new club are to "further the knowledge, understanding and enjoyment of the Amstrad computer in its entirety". Meetings are held every third Tuesday of each month at 7.30 p.m. in the Kippa-Ring State School Library, Elizabeth Avenue.

THE WARWICK AMSTRAD USER GROUP

President:	Adrian Christensen	
Secretary:	John Wode	(076 61 5176)
Treasurer:	Neville Christensen	

Meetings take place at the Warwick Education Centre on the first Saturday of each month from 7.30 p.m. Discounts for members have been obtained from two local dealers.

TOWNSVILLE AMSTRAD USER GROUP

President:	Allan Maddison	(077 79 2607)
Vice President:	Brett Kettle	(077 78 6915)
Treasurer:	Shirley Paull	(077 78 2318)
Secretary:	Alister Buckingham	(077 73 3955)

The club meets at 7.30 p.m. on the first and third (only) Tuesdays in each month in the Science Block of the Kirwan High School in Thuringowa Drive. Activities include Basic programming and CP/M sessions, new product reviews, problem solving, competitions and social outings. Club members also receive discounts on software and stationery from selected stores.

TASMANIA

SOUTHERN TASMANIAN AMSTRAD USER CLUB

President	Peter Campbell	
Secretary	Rosemarie Parkinson	(002 43 8101)
Publicity Officer	Danny Brittain	(002 47 7070)

Meetings generally take place on the first Wednesday of each month, commencing at 7.30 p.m. Enquiries should be made to Graham or Jenese West - (002) 34 5817.

LAUNCESTON AMSTRAD USER GROUP

This is a new group which has provisionally organised meetings at the Launceston Workingman's Club on the last Saturday of each month from 1.30 p.m. onwards. For more details contact Andrew Banfield on (003) 44 3181 after 6.00 p.m.

NEW ZEALAND

AMSTRAD CANTERBURY

Contact:	Christine Linfoot	459 132
	Ian Orchard	524 064

The club meets on the fourth Wednesday of each month at Four Avenues School, cnr. Madras Street and Edgeware Road, Christchurch 1. They are currently running tutorials in Basic for beginners, machine code and general purpose debugging, with ad hoc discussion groups on the working utilities such as Tasword, Masterfile etc. The postal address of the group is: PO Box 23.079 Templeton, Christchurch, NZ.

Orangeade Stall

A trading game for youngsters

Adapted by A.F. Ryan

Having a grandchild about 8 years old, I have been interested, ever since buying my CPC464, in the possibilities of writing some programs for her amusement and enlightenment. So far I have only found time and inspiration to write one such, the subject of this one.

I read "Parents, Kids and Computers" by Lynne Alper and Meg Holmberg, and they described "Lemonade", many versions of which I am told exist - almost as many as there are micros to run it on.

It sounded a simple and attractive idea and obviously offered scope for plenty of "local colour", so I set down to write my own version.

The "Scenario" calls for the player to run a roadside Saturday morning stall, selling orange drinks to the passers-by "to earn pocket money". A kindly parent provides a stake to start with, enough to purchase initial supplies, and the player then has to try and make money over a series of "weekends".

Prices of drink crystals and paper cups are given (these can easily be adjusted to the local rates) and the "ground rules" state that a decision must be made on each day as to how many drinks to prepare, as (a) no further drinks can be made up once the day has started; and (b) any drinks left over at the end of the day must be thrown away. In addition to this decision, the player has to decide what price to charge for each drink (high prices will definitely suppress sales!!). Of course the player must also decide how much supplies to buy, and how much to spend on "advertising" (posters in local shop windows, at a small charge by the shopkeeper).

A whole range of random factors decide how may people will come down the street on the day, and some of these will affect the proportion who are willing to stop and buy a drink. An attempt has been made to make this model behave in a reasonably lifelike way (e.g. while the weather will vary from day to day, there will be a "bias" in any one season towards generally-better or generally-worse than average).

After the decisions have been made for each "trading day", the results both financial and in terms of weather conditions, unexpected events etc. are displayed, and then the pattern is repeated for the "next weekend". After ten "selling days", the overall results for the season are displayed.

I would not pretend that the program can't be improved, functionally it works fine, but it could have much better sound effects and graphic displays if you work it up. I work with a green screen myself, so have not attempted fancy colour effects.

My granddaughter and a friend of hers (ages 8 and 9) play the game with pleasure, but 10-12 would probably be a better age range. The player needs to have some appreciation of money values, and to be able to work out the implications of costs and selling prices at a simple level.

Those willing to experiment and "see what happens - " will have the most fun, and also learn the most.

```
10 REM ****
*******************************
20 REM    SMALL BUSINESS Versn. 1.1

30 REM *********************************
**
40 REM     by Tony Ryan, June 85
50 REM *********************************
**
60 REM A simulation game for children,
70 REM (ages about 10-14). Demonstrates

80 REM effects of decisions on a
90 REM     "Small Business".
100 REM ********************************
***
110 REM This program is in the public do
main.
120 REM All prices are in statement 230,
 so
130 REM these can easily be kept up to d
ate.
140 REM ********************************
***
150 REM When starting a second or subseq
uent game,
160 REM reply "Y" will result in a new r
andomisation,
170 REM reply "R" will restore the previ
ous one, this
180 REM allows two players to compare th
eir results
190 REM under identical circumstances.

200 REM ********************************
***
210 REM initialisation. intro, ask wheth
er instructions required.
220 REM prices and stake money in dollar
s.cents
230 REM prcrys=price of drink crystals:
prcup10=packet of 10 cups
240 REM prcup100=packet of 100 cups: sta
ke=starter/grubstake
250 REM pradv=price of one advert for on
e week
260 prcrys=3.3:prcup10=0.61:prcup100=4.8
5:stake=15:pradv=0.5
270 DIM evt(20):DIM evt$(20)
280 best=0
290 GOSUB 910
300 RANDOMIZE TIME : 'initial set of ran
domizer
310 randr=(RND) : 'hold this for restart
s
320 RANDOMIZE randr : 'reset to start fi
gure
330 MODE 1:BORDER 1:PAPER 0:PEN 1:INK 0,
1:INK 1,24
340 LOCATE 4,10:PRINT "Hullo! What is yo
ur name?"
350 PRINT "  Or names, if there's more t
han one of you!"
360 LOCATE 1,14:PRINT"Type name(s), then
"
```

```
370 INPUT ;"press ENTER ",name$
380 IF LEN(name$)<27 THEN GOTO 430
390 PRINT"Sorry, name(s) has to be less
than 27   letters long (counting spaces
too)"
400 GOSUB 3520 :'siren noise
410 FOR a=1 TO 4000:NEXT a
420 CLS:GOTO 360
430 CLS:PRINT "Hullo ";name$;"!":PRINT C
HR$(7):PRINT
440 PRINT"You're going to run an orange-
drink     stall for ten Saturdays this s
ummer to  try and earn some spending mon
ey."
450 PRINT
460 DEFINT w
470 PRINT "Do you need instructions? (y
or n)"
480 a$=INKEY$:IF a$="" GOTO 480
490 IF a$="y" OR a$="Y" GOTO 500 ELSE GO
TO 650
500 REM Instructions.
510 CLS:MODE 2:INK 0,9:INK 1,20
520 LOCATE 1,1
530 PRINT"So you can get started, Mum or
 Dad will lend you"
540 LOCATE 58,1:PRINT"You will have to p
ay":LOCATE 50,1:PRINT USING "$$##.##";st
ake
550 PRINT"this back at the end of your t
en-week season. You can borrow a table a
nd a      couple of chairs for your stal
l, but you will have to paint a big noti
ce to     attach to the front of the sta
ll. "
560 PRINT:PRINT"Prices of the materials
you will need (paper cups and orangeade
crystals) will  appear at the start of e
ach week. You have to decide how much to
 buy and also   how many drinks to prepa
re for each Saturday. You have to make u
p all your "
570 PRINT"drinks in cups before business
 starts on Saturday, you can't make up m
ore if yourun out during the day."
580 PRINT"Made-up drinks left over at th
e end of the day cannot be kept for next
 week and are wasted, but unused cups an
d crystals will keep and can be used ano
ther week."
590 PRINT"You can set up your stall by a
 fairly busy road, but all sorts of thin
gs may    affect the number of people wh
o walk along it on Saturday. Some of the
se people will stop to buy a drink from
you, but not all. The number who stop wi
ll depend"
600 PRINT"on what sort of day it is and
also on the price you are charging for y
our       drinks.  You can change this p
rice for each Saturday, it is one of the
 things   you have to decide."
610 PRINT"You can try to encourage more
people to come by making up advertising
posters   and putting them in local shop
```

```
s. Some shopkeepers are willing to put y
our          posters up, but they charge yo
u a small amount per week for use of the
ir space."
620 PRINT:PRINT"      Press the SPACE bar
 when you are ready to start, and GOOD L
UCK!"
630 REM set up tables -
640 a$=INKEY$:IF a$="" GOTO 640
650 REM mainline
660 MODE 1:INK 0,1:INK 1,24
670 totdk=0: 'total drinks made up
680 totc=0:totcp=0: 'totals of cups boug
ht & cost (for avg)
690 wadvt=0: 'total advertising cost (fo
r avg)
700 crys=0:cups=0:cash=stake:totw=0:totp
=0:totb1=0:totb2=0:wadv=0
710 REM reset accumulators for 1st or la
ter game
720 bsw=0: wk=0: 'reset bankruptcy switc
h and week counter
730 wbias=INT(40+(RND*20)): 'set weather
 bias for season
740 FOR week=1 TO 10
750 IF bsw=1 THEN GOTO 800
760 wk=wk+1
770 GOSUB 980: 'prices and orders
780 GOSUB 1980: 'calculate factors
790 GOSUB 2340: 'display results
800 NEXT week
810 GOSUB 3190: 'final results
820 IF a$="y" OR a$="Y" THEN RANDOMIZE T
IME :randr=(RND):GOTO 860
830 IF a$="R" OR a$="r" THEN RANDOMIZE r
andr : GOTO 860
840 REM reply="r" allowed for.' This repe
ats exact conditions of last game
850 STOP
860 CLS:LOCATE 5,5
870 PRINT"Is this ";name$;" again (y or
n)?"
880 a$=INKEY$:IF a$="" GOTO 880
890 IF a$="y" OR a$="Y" GOTO 650 ELSE GO
TO 330
900 IF a$="y" OR a$="Y" GOTO 650 ELSE ST
OP
910 REM rest of initialisation
920 FOR a=1 TO 20
930 READ evt(a)
940 READ evt$(a)
950 NEXT a
960 cstp=100*cstp : 'convert to cents
970 RETURN
980 REM start of a week.
990 REM list week number, prices and sto
cks-in-hand.
1000 REM
1010 MODE 2
1020 INK 0,6:INK 1,18
1030 LOCATE 18,1:PRINT"WEEK ";week
1040 LOCATE 1,3: PRINT"PRICES OF MATERIA
LS ARE:-"
1050 LOCATE 5,4:PRINT"Orange-drink cryst
als (one tin makes 30 glasses)"
1060 LOCATE 5,5:PRINT"Paper cups (10)"

1070 LOCATE 5,6:PRINT"paper cups (100)"

1080 LOCATE 5,7:PRINT"Advertising (per p
oster)"
1090 LOCATE 55,4:PRINT USING "$$##.##";p
rcrys
1100 LOCATE 55,5:PRINT USING "$$##.##";p
rcup10
1110 LOCATE 55,6:PRINT USING "$$##.##";p
rcup100
1120 LOCATE 55,7:PRINT USING "$$##.##";p
radv
1130 LOCATE 1,9:PRINT"YOUR PRESENT STOCK
S ARE:-"
1140 LOCATE 5,10:PRINT"Orange-drink crys
tals"
1150 LOCATE 5,11:PRINT"Paper cups"
1160 LOCATE 5,12:PRINT"CASH"
1170 LOCATE 40,10: PRINT USING "###.##";
crys
1180 cups=INT(cups)
1190 LOCATE 40,11: PRINT USING "#,###";c
ups
1200 LOCATE 37,12: PRINT USING "$$####.#
#";cash
1210 LOCATE 1,14: PRINT "YOU DECIDE TO B
UY (press ENTER when ready):-"
1220 LOCATE 10,15:INPUT ;"Tins of orange
 crystals            - ";cr
1230 crp=cr*prcrys
1240 IF crp>cash GOTO 1250 ELSE GOTO 131
0
1250 LOCATE 10,16:PRINT"You don't have e
nough money for that."
1260 GOSUB 3520
1270 FOR a=1 TO 2000:NEXT a
1280 LOCATE 47,15:PRINT SPACE$(10)
1290 LOCATE 10,16:PRINT SPACE$(45):GOTO
1220
1300 GOTO 1220
1310 REM Enough money to buy crystals, p
roceed
1320 cash=cash-crp:crys=crys+cr
1330 PRINT CHR$(7):LOCATE 55,15:PRINT"OK
, cash now = ";
1340 PRINT USING "$$####.##";cash
1350 LOCATE 10,16:INPUT ;"paper cups
                     - ";cp
1360 REM must work out cost of cups acco
rding to qty
1370 IF cp>99 GOTO 1410
1380 REM buying cups in units of 10 only
1390 cppr=cp*prcup10/10
1400 GOTO 1510
1410 REM split orde. of cups into units
of 100s and 10s
1420 cp5=0:cpr=cp
1430 REM cp5 = no. of units of 100
1440 cp5=cp5+1
1450 cpr=cpr-100
1460 IF cpr>99 GOTO 1440
1470 cppr=cp5*prcup100
1480 REM (units of 100) * (price of 100)
```

```
 = cppr
1490 IF cpr<10 GOTO 1510
1500 cpr=INT(cpr/10):cppr=cppr+(cpr*prcu
p10)
1510 REM now check whether overspent
1520 IF cash>cppr GOTO 1580
1530 LOCATE 10,17:PRINT"you haven't enou
gh cash for that many cups"
1540 GOSUB 3520
1550 FOR a=1 TO 2000:NEXT a
1560 LOCATE 47,16:PRINT SPACE$(10)
1570 LOCATE 10,17:PRINT SPACE$(45):GOTO
1350
1580 REM cups purchase is ok
1590 LOCATE 55,16:PRINT"OK";CHR$(7)
1600 totc=totc+cp:totcp=totcp+cppr: 'acc
umulate cost/price of cups
1610 cash=cash-cppr:cups=INT(cups+cp)
1620 LOCATE 55,16:PRINT"OK, cash now = "
;
1630 PRINT USING "$$####.##";cash
1640 REM advertising -
1650 LOCATE 10,17:INPUT"How many adverti
sing posters this week - ",ca
1660 cap=ca*pradv
1670 IF cash>cap GOTO 1720
1680 LOCATE 10,17:PRINT SPACE$(45):LOCAT
E 10,17:PRINT"Not enough cash left"
1690 GOSUB 3520
1700 FOR a=1 TO 2000:NEXT a
1710 LOCATE 10,17:PRINT SPACE$(25):GOTO
1650
1720 REM budget is ok
1730 IF ca>2 THEN wadv=wadv+ca ELSE wadv
=wadv+ca-2
1740 IF wadv<0 THEN wadv=0
1750 REM effect of advertising is cumula
tive
1760 cash=cash-cap
1770 LOCATE 55,17:PRINT "OK, cash now =
";
1780 PRINT USING "$$####.##";cash:PRINT
CHR$(7)
1790 LOCATE 10,19:INPUT"What is your pri
ce this week for a drink. In cents - ";d
p
1800 dp=dp/100
1810 PRINT CHR$(7)
1820 LOCATE 10,20:INPUT"How many drinks
will you make up this week";dq
1830 IF (crys+0.005)>(dq/30) GOTO 1870
1840 LOCATE 10,20:PRINT"You don't have e
nough crystals left            "
1850 GOSUB 3520
1860 FOR a=1 TO 2000:NEXT a:GOTO 1820
1870 IF cups>(dq-1) GOTO 1900
1880 LOCATE 10,20:PRINT"You don't have e
nough cups left                "
1890 GOTO 1850: 'delay loop, then repeat
 question
1900 PRINT CHR$(7):crys=crys-(dq/30): cu
ps=cups-dq: 'decrement stocks
1910 MODE 0:INK 1,15:LOCATE 1,10:PRINT".
..I'M THINKING..."
1920 BORDER 4,11
1930 FOR a=1 TO 1000:NEXT a
1940 totdk=totdk+dq: 'running total of d
rinks made up
1950 IF totdk>0 THEN cstp=(prcrys/30)+(p
rcup10/10)+(wadvt/totdk) ELSE cstp=(prcr
ys/30)+(prcup10/10)
1960 REM calc cost price of a drink
1970 RETURN
1980 REM calculation of operations this
week
1990 REM get an initial number for  peop
le on street
2000 peop=30+(RND*100)
2010 peop=peop+(peop*wadv/50): 'advertis
ing may increase people
2020 REM calc weather using approx "norm
al distribution" + bias
2030 x=((RND*20)-10): y=(RND*20)-10
2040 wea=wbias+(x*y) : '(RND)*(RND) appr
oximates normal curve
2050 IF wea<1 THEN GOTO 2030
2060 IF wea>100 THEN GOTO 2030 : 'reject
 values outside range
2070 IF wea<6 THEN peop=0 : byr=0 : GOTO
 2220 'disastrous weather - no sales
2080 peop=peop*wea/60: ' people on stree
t propn to weather
2090 event=INT(RND*100) : 'special event
s, select only when <21
2100 IF event<21 THEN peop=peop*evt(even
t)
2110 totbl=totbl+peop
2120 REM buyers = people/5+weather-facto
r
2130 IF wea<20 THEN byr=(peop/10): GOTO
2210
2140 IF wea<40 THEN byr=(peop/5): GOTO 2
210
2150 IF wea<70 THEN byr=(peop*0.3): GOTO
 2210
2160 IF wea<90 THEN byr=(peop*0.4): GOTO
 2210
2170 IF wea>89 THEN byr=(peop*0.6): GOTO
 2210
2180 REM advertising can increase % of b
uyers
2190 IF wadv>2 THEN byr=byr+(byr*wadv/10
0)
2200 IF byr>peop THEN byr=peop: ' can't
exceed popn
2210 REM take price into account
2220 REM no effect between cost and 1.5*
cost
2230 IF dp>cstp THEN GOTO 2270
2240 REM price below cost increases buye
rs!
2250 byr=byr*cstp/dp : 'this is rather s
teep, may need adjustment
2260 GOTO 2300
2270 IF dp>(1.5*cstp) THEN GOTO 2290 ELS
E GOTO 2310
2280 REM Higher prices have exponential
effect
2290 dp.cstp=(dp-cstp)*100:byr=byr*(100-
(dp.cstp)^1.01)/100
```

```
2300 IF byr<0 THEN byr=0
2310 IF byr>peop THEN byr=peop
2320 byr=INT(byr)
2330 RETURN
2340 REM picture of stall and drinks
2350 MODE 1:INK 1,26:INK 0,9:INK 2,15: B
ORDER 11
2360 PLOT 98,60: 'draw stall outline
2370 DRAW 98,360,1:MOVER 1,0:DRAW 99,60:
MOVER 1,0:DRAW 100,360
2380 DRAW 540,360:DRAW 540,60:MOVER 1,0:
DRAW 541,60
2390 MOVER 1,0:DRAW 542,360:MOVE 100,150
:DRAW 540,150
2400 MOVER 0,1:DRAW 100,151:MOVER 0,1:DR
AW 150,152
2410 DRAW 130,150:MOVER 1,0:DRAW 131,100
:MOVER 1,0:DRAW 132,150
2420 MOVE 510,100
2430 DRAW 510,150:MOVER -1,0:DRAW 509,10
0:MOVER -1,0:DRAW 508,150
2440 xlt=100:xrt=540:posn=152
2450 FOR a=1 TO 30
2460 xlt=xlt+1:xrt=xrt-1:posn=posn+1
2470 MOVE xlt,posn
2480 DRAW xrt,posn
2490 NEXT a
2500 MOVE 100,280
2510 DRAW 540,280
2520 REM now the banner
2530 n=LEN(name$)
2540 n=INT(21-(n/2))
2550 LOCATE n,4
2560 PRINT name$
2570 LOCATE 11,6
2580 PRINT"ORANGE DRINKS ";
2590 PRINT USING "$$###.##";dp
2600 REM Draw glasses on counter
2610 xlt=130
2620 IF dq>10 THEN wglass=10 ELSE wglass
=(dq)
2630 FOR a=1 TO wglass
2640 posn=161
2650 MOVE xlt,210
2660 DRAW xlt,160,2
2670 DRAW xlt+20,160
2680 DRAW xlt+20,210
2690 FOR b=1 TO 40
2700 MOVE xlt+1,posn
2710 DRAW xlt+19,posn
2720 posn=posn+1
2730 NEXT b
2740 xlt=xlt+40
2750 NEXT a
2760 FOR a=1 TO 1000:NEXT a
2770 BORDER 1
2780 REM present results of week's tradi
ng
2790 MODE 1:INK 0,1:INK 1,19
2800 LOCATE 7,5:PRINT"RESULTS OF TRADING
, WEEK ";week
2810 IF wea<5 THEN wn=wea*10:wea$="Worst
 storm for ":GOTO 2880
2820 IF wea<10 THEN wea$="Thunderstorm,
heavy rain":GOTO 2880
```

```
2830 IF wea<20 THEN wea$="Heavy rain all
 day":GOTO 2880
2840 IF wea<40 THEN wea$="Dull weather,
rather cool":GOTO 2880
2850 IF wea<70 THEN wea$="Average summer
 weather":GOTO 2880
2860 IF wea<90 THEN wea$="Fine, sunny da
y":GOTO 2880
2870 wea$="Very hot, everyone thirsty!":
GOTO 2880
2880 LOCATE 3,7:PRINT"The weather this S
aturday was -"
2890 IF wea<5 GOTO 2900 ELSE GOTO 2910
2900 LOCATE 5,8:PRINT wea$;INT(wn);" yea
rs"
2910 LOCATE 5,8:PRINT wea$
2920 IF event>20 GOTO 2950
2930 LOCATE 3,10:PRINT"On this Saturday
-"
2940 LOCATE 5,11:PRINT evt$(event)
2950 LOCATE 3,13:PRINT INT(peop);" peopl
e came along the street"
2960 LOCATE 3,14:PRINT INT(byr);" were c
ustomers"
2970 IF byr>dq THEN 2980 ELSE 3020
2980 LOCATE 5,15:PRINT"but you only had
"dq; " drinks made up"
2990 totb2=totb2+(byr-dq): 'keep tally o
f missed buyers
3000 byr=dq: 'reduce buyer numbers to dr
inks available
3010 LOCATE 5,16
3020 IF dq>byr THEN 3030 ELSE 3040
3030 ta=dq-byr:PRINT"Left-over drinks to
 throw away ";INT(ta):PRINT
3040 cupc=(dq*totcp/totc)+(dq*prcrys/30)
+cap: 'cost of materials used
3050 PRINT"Cost of stocks used today ";

3060 PRINT USING "$$###.##";cupc
3070 PRINT"Your gross takings were ";

3080 take=byr*dp:PRINT USING "$$####.##"
;take
3090 cash=cash+take
3100 REM test for bankruptcy
3110 IF cash<prcup10 AND cups<1 THEN GOT
O 3130
3120 IF cash<prcrys AND crys<0.04 THEN G
OTO 3130 ELSE GOTO 3140
3130 bsw=1: 'set switch=bankrupt
3140 LOCATE 1,20
3150 PRINT"Press space bar when ready"

3160 a$=INKEY$:IF a$="" GOTO 3160
3170 totw=totw+wea:totp=totp+peop
3180 RETURN
3190 REM report results of season
3200 MODE 1:INK 0,1:INK 1,24
3210 LOCATE 5,1:PRINT"RESULTS FOR SUMMER
 SEASON"
3220 IF bsw=0 THEN GOTO 3260: 'OK, norma
l ending
3230 PRINT:PRINT"You haven't enough cash
 left to carry on"
```

```
3240 PRINT"Your trading only lasted ";wk
;" weeks"
3250 GOTO 3490
3260 LOCATE 1,3:PRINT"The weather was ";

3270 IF totw<200 THEN PRINT"perfectly fr
ightful":GOTO 3340
3280 IF totw<300 THEN PRINT"very bad":GO
TO 3340
3290 IF totw<400 THEN PRINT"poor":GOTO 3
340
3300 IF totw<600 THEN PRINT"about averag
e":GOTO 3340
3310 IF totw<700 THEN PRINT"good":GOTO 3
340
3320 IF totw<800 THEN PRINT"very good":G
OTO 3340
3330 IF totw<900 THEN PRINT"just beautif
ul" ELSE PRINT"unbelievably perfect!"
3340 LOCATE 1,5:PRINT"The average number
 of people on the      street on one day
was ";INT(totb1/10)
3350 LOCATE 1,8:PRINT"The number of buye
rs you missed through  running out of dr
inks was ";totb2
3360 LOCATE 1,11:PRINT"You ended up with
 cash ";
3370 PRINT USING "$$####.##";cash
3380 PRINT"But you have to repay your gr
ubstake of ";
3390 PRINT USING "$$###.##";stake
3400 LOCATE 1,14:PRINT"So you are left w
ith cash ";
3410 cash=cash-stake
3420 PRINT USING "$$####.##";cash:PRINT
3430 PRINT"Plus orange crystals ";INT(cr
ys)
3440 PRINT"And paper cups       ";INT(cu
ps)
3450 PRINT
3460 IF cash>best THEN best=cash: 'best
score to date
3470 PRINT"best result so far today is "
;
3480 PRINT USING "$$####.##";best
3490 LOCATE 5,20:PRINT"Would you like to
 try again (y or n)?";
3500 a$=INKEY$:IF a$="" GOTO 3500
3510 RETURN
3520 REM hee-haw-siren noise rtne
3530 ENT 1,2,17,70
3540 SOUND 1,142,140,15,0,1
3550 SOUND 1,142,140,15,0,1
3560 RETURN
3570 DATA 0.7,Test Match on TV,0.4,Test
Match in town,2.0,Local Sports Day,0.7,S
ports Day next town,0.5,Car crash closes
 road,2.5,House fire - many people,1.3,C
ouncil workmen nearby,3.0,Local paper fe
atures you,1.3,Garage Sale 2 doors away
3580 DATA 0.6,Kite flying contest elsewh
ere,0.7,Airshow at airfield,0.8,Dog Show
 in town,1.2,Tramping Club go past,1.3,P
arachutist lands nearby,1.4,Bus strike &
 people walking,0.5,Bus strike & people
stay home,0.7,House fire elsewhere
3590 DATA 1.2,Street trees in bloom,1.7,
Gorse fire on hills,0.4,Smoke from fire
across stall,2.2,search for lost child
```

The Learning Centre

CP/M Explored - Part Four

by Shane Kelly

In the previous articles I have concentrated on how to use the various .COM programs that are supplied with CP/M 2.2 on the AMSTRAD. In response to the very few letters that have arrived with comments on this series of articles, I will now chronicle the differences between CP/M 2.2. and CP/M 3.0. The depth of detail will not be great but should enable you to assess the probability of public domain software running under your particular system.

CP/M 3.0 was released to take advantage of the 'new' generation Z80 based computers that could access more than 64k of memory at one time. This is done by 'bank switching' or swapping in and out of the address range different 'banks' of memory chips. The advantages of this are less disc accesses, larger TPA and therefore larger programs. At the same time Digital Research made some radical changes to CP/M to enable the operating system to perform in commercial environments with the features that commerce wanted. these included password protection at disk and file level and date and time stamping of files along with 'drive train searching' which allows you to have a program on drive b: while logged onto drive a: and run the program on drive b:. This was a great boon to commercial users as there was no need for operators to see the (sometimes) frightening A> prompt and not knowing what to do next! These features were added to CP/M by adding further service calls to the BDOS and enhancing the BIOS. The beauty of this is that compatibility with CP/M 2.2 was maintained (in the majority of cases!). In general then any program that runs under AMSTRAD CP/M 2.2 'straight off the disc' will almost certainly run under AMSTRAD CP/M 3.0. I must be vague on this because there are so may programs written for CP/M 2.2 and it would be impossible for me to try them all! However, I can say that all the public domain programs that I run under CP/M 2.2 run with no problems under CP/M 3.0. Be warned that the converse is not true! Programs written for CP/M 3.0 explicitly will not run under CP/M 2.2 because of the extended features that will have been used in the program.

The transient programs supplied with CP/M 3.0 are more numerous than those supplied with CP/M 2.2 and take advantage of all the facilities available in the system. As a brief rundown, here are the main transient commands supplied with all CP/M 3.0 systems. The specialised commands supplied with AMSTRAD are explained in more detail in the manual.

DATE: This sets or displays the date. The AMSTRAD clock is not battery operated so the date and time is lost at switchoff and must be set every time the computer is powered up. The clock is not updated when disc accesses are performed but his will not unduly affect the time over the average stint at the keyboard.

DEVICE: This displays and assigns the device and is analogous to STAT for this purpose. Also included are various options for setting communications protocols.

GET: This command redirects input to a program from a disk file. This is particularly handy for programs that require the same information each time they are run. Put the information into a file and direct the program to take it's input from your file via GET.

PUT: This is the converse of GET. All program output may be put to a disk file.

SHOW: This takes the functions of STAT that device left out, mainly those statistics relating to disc drive characteristics.

Those are the main transient commands supplied with CP/M 3.0. AMSTRAD have given several more that pertain to CP/M 3.0 as supplied but their use is specialised to this system.

We mentioned in passing Public Domain software. Briefly, PD Software are programs written by CP/M users that perform tasks that someone had a need for and have been donated to the CP/M user group library to be used free of copyright by anyone. There are no copyright restrictions on these programs and may be freely distributed to whomever you like. There are many megabytes of this software available and while there are many worthwhile programs contained in the library there is some rubbish as well! It is worthwhile to obtain a catalogue of the software as there are many interesting utilities and implementations of computer languages available for the price of a disc.

Of all the public domain software that I use, my favourite is NUSWEEP (sometimes called SWEEP or SWEEP38). This is probably the most useful file transfer utility that I have seen. In use NUSWEEP is friendly and generally is fail safe i.e. it does not crash out to A>, but handles the errors inside itself. There are a host of operations that can be carried out on one file at a time or by 'tagging' a group of files they can be transferred in mass from user area to user area or from one drive to another with or without CRC checking. (CRC is cyclic redundancy checking and is a protocol for error checking a stream of bytes). With NUSWEEP you can transfer a file and rename it at the same time, view ASCII files on the screen (Squeezed or unsqueezed files, it makes no difference!), change discs any time you like, squeeze or unsqueeze files and many more less than every day type of operations. If someone in your user group has a copy of NUSWEEP then get it!

My next most used piece of software is CAT. This is a cataloguing utility that requires you to name each disk with something like-WORK.001 or whatever (the hyphen makes the name first in a sorted directory) and then by using CAT you will build up a file of the programs on that disc. You may then print the catalogue and so have a hard copy of what each disc contains. That beats putting fifteen discs in the drive and typeing DIR fifteen times! To find a particular file in your catalogue you simply type CAT filename.ext *.* and CAT will search your file and tell you what disc your filename.ext is on. So much easier than continually re-inserting the discs in the drive and less wear and tear also!

For recreation purposes, I like playing Colossal caves (the ORIGINAL adventure). This is a game that can take many months to get through and it's textual description still leaves some of the modern day adventures in the shade. This is not the only game that appears in the public domain but most of the rest require MBASIC or its equivalent to run. If your feeling lucky you may even try to convert these games to run under locomotive basic. Those of you who have 8256's can run these programs under Mallard basic which I am reliably informed is MBASIC compatible.

One of the 'personalities' of public domain software is Ward Christiansen, who wrote such stars as MODEM7 and CAT which have stood the test of time and proved their usefulness to many hobbyists over the years. One question that had me foxed for a long time was why should anyone write a program and then give it away? I found out that many of these programs were written by professional programmers in the USA who, under the terms of their employment, could not sell anything they produced except through their employer so these programs were donated to the CP/M user group library. Their loss, our gain.

Also included in the PD software are assemblers and disassemblers, editors and file creating utilities, and many more programs of such obscure use that figuring them out will take you a lifetime. The public domain software will keep you occupied for many years and is available from many sources. the difficulty for us AMSTRAD users is to obtain the software in the appropriate disc format. Select Software of Sydney has the software in AMSTRAD format (on 5 1/4 inch disc ONLY!) and can supply by mail. Select Software advertise in APC and YOUR COMPUTER and will take phone orders and bankcard. If you know someone who has an AMSTRAD running a 5 1/4 inch disc drive then transfer to a 3 inch is straight forward.

I hope this has cleared some of the fog surrounding CP/M 2.2 and CP/M 3.0 and what will or will not run on these systems. For the next article we will be looking at bulletin boards and Viatel as this appears to be another area causing confusion to the average user.

As always any comments regarding the points covered (or not covered) in this series of articles are gratefully received via this magazine.

Build a Sprite

from P.T. Crowe (SCAUG)

This program was developed mainly because of the need to design new sprites for the "Roland on the Run" game which recently appeared in a number of parts in the English Amstrad Computer User, and to find out how the screen memory is used and addresses when using multi-coloured sprites. As 'Roland' is in Mode 1, this program is also designed to build sprites in that Mode.

In Mode 1, one byte controls the colour of four pixels. Each pixel is controlled by two bits. Bits 7 and 3 control the leftmost pixel; the next pixel across is controlled by bits 6 and 2; the next by bits 5 and 1; and finally the rightmost pixel by bits 4 and 0. For example, take the binary number 00000001 and look at bits 4 and 0 as a pair. Putting them side by side (01) the two bits can now be read as a normal binary number. In this case the answer is 1 which represents INK 1 for the rightmost pixel. If the binary number was 00010000, then bits 4 and 0 would read 10, which, when read as a binary number gives 2, making the rightmost pixel INK 2. The monsters in 'Roland' use 64 bytes - this program builds sprites of the same size for use in the 'Roland' game.

Inside the Program

When a pen number is keyed in, it is given a predetermined decimal number and will produce the correct binary number for the bit. When the 'P' (Place in memory) is pressed a FOR...NEXT loop is set up producing 256 bits, which in turn are combined in fours to produce 64 bytes. The bytes are converted to a hex number and stored as the data.

Running the Program

On running the program, you will notice a flashing cursor in the top left hand corner of a large square. The square is the working area in which the sprite is built. The cursor is moved about the screen via the cursor keys. There is no need to follow a pattern - just move the cursor around and input the pen number you wish a pixel to be at a chosen point.(0=Black, 1=White, 2=Red and 3=Blue). With the background already black there is no need to fill it. Build your sprite with the other three colours using 0 (black) to wipe out any errors.

At any time during the building of your sprite 'P' can be pressed. This will pole the information into screen memory producing an actual size sprite in the small square. The cursor then returns to the work area allowing alterations. Continue building your sprite and when happy with it, press 'S' which will save the data to disc or tape. If you find that you are not happy with the sprite, then press 'L' and the data will be loaded allowing further alterations. Pressing the control and 'P' keys sends the hex data to a printer.

The hex data produced by this program (representing your sprite) can now be placed into the unassembled listing of 'Roland', or it can be poked in after the Roland program is running.

The size of the sprite is two characters high by two characters wide. We suggest that you make a drawing of your sprite in a 16 x 16 grid prior to running the program - it will make the building process easier.

(Program overleaf)

```
10 '*** BUILD A MONSTER BY SCAUG     P.T.
CROWE. ***
20 ON BREAK GOSUB 1550
30 MODE 1:DIM d(16,16),h$(64),pix(64),dd
(255):pixloc=&C2A6
40 INK 0,0:INK 1,26:INK 2,6:INK 3,11:BOR
DER 0:PEN#1,3
50 GOSUB 480:GOSUB 1080
60 '***  set up selecter  ***
70 menu$=CHR$(16)+"slp"+CHR$(240)+CHR$(2
41)+CHR$(242)+CHR$(243)
80 WHILE r=0
90 GOSUB 140
100 z=INSTR(menu$,a$)
110 ON z GOSUB 710,800,920,540,350,380,4
10,440
120 WEND
130 RETURN
140 '***  display cursor  ***
150 LOCATE x+1,y+1:
160 IF a$="0" THEN PEN 0:PRINT CHR$(255)
;:d(x,y)=p0
170 IF a$="1" THEN PEN 1:PRINT CHR$(255)
;:d(x,y)=p1
180 IF a$="2" THEN PEN 2:PRINT CHR$(255)
;:d(x,y)=p2
190 IF a$="3" THEN PEN 3:PRINT CHR$(255)
;:d(x,y)=p3
200 a$=""
210 cur=&BB84
220 WHILE a$=""
230 LOCATE x+1,y+1:PEN 1
240 cur=cur XOR 5:CALL cur
250 FOR flash=0 TO 300:NEXT flash
260 a$=LOWER$(INKEY$)
270 WEND
280 CALL &BB84
290 LOCATE x+1,y+1:
300 IF a$="0" THEN PEN 0:PRINT CHR$(255)
;
310 IF a$="1" THEN PEN 1:PRINT CHR$(255)
;
320 IF a$="2" THEN PEN 2:PRINT CHR$(255)
;
330 IF a$="3" THEN PEN 3:PRINT CHR$(255)
;
340 RETURN
350 '***  up  ***
360 IF y>0 THEN y=y-1
370 RETURN
380 '***  down  ***
390 IF y<15 THEN y=y+1
400 RETURN
410 '***  left  ***
420 IF x>0 THEN x=x-1:GOSUB 480
430 RETURN
440 '***  right ***
450 IF x<15 THEN x=x+1:GOSUB 480
460 RETURN
470 '***  set pix values  ***
480 IF x=0 OR x=4 OR x=8 OR x=12 THEN p0
=0:p1=128:p2=8:p3=136
490 IF x=1 OR x=5 OR x=9 OR x=13 THEN p0
=0:p1=64:p2=4:p3=68
500 IF x=2 OR x=6 OR x=10 OR x=14 THEN p
0=0:p1=32:p2=2:p3=34
510 IF x=3 OR x=7 OR x=11 OR x=15 THEN p
0=0:p1=16:p2=1:p3=17
520 RETURN
530 '***  calculation  ***
540 FOR xx=0 TO 15:dd(n)=d(xx,yy):td=td+
d(xx,yy)
550 IF a=3 THEN nn=nn+1:h$(nn)=HEX$(td):
a=0:td=0:GOTO 570
560 a=a+1
570 n=n+1:IF LEN(h$)>2 THEN n=0 xx=0:yy=
0:nn=0:td=0:goto540
580 NEXT xx
590 yy=yy+1:IF yy>15 THEN n=0:xx=0:yy=0:
nn=0:GOTO 620
600 GOTO 540
610 '***  print monster  ***
620 CLS#2:FOR zz=1 TO 64:h=VAL("&"+h$(zz
)):POKE pixloc,h:pixloc=pixloc+1:mpd=mpd
+1:IF mpd=4 THEN mpd=0:pixloc=pixloc+204
4:count=count+1:'NEXT zz:h=0:zz=0:RETURN
630 IF count=8 AND mpd=0 THEN pixloc=&C2
F6
640 IF count=16 AND mpd=0 THEN pixloc=&C
2A6:count=0:mpd=0
650 NEXT zz:h=0:zz=0
660 CLS#2:'***  print data to screen  **
*
670 FOR nn=1 TO 64
680 PRINT#2,USING"\\";h$(nn);" ";
690 NEXT nn
700 nn=0:RETURN
710 '***  to  printer  ***
720 GOSUB 790:IF pp=64 THEN CLS#2:PRINT#
2:PRINT#2:PRINT#2,"Printer Off Line":FOR
 td=1 TO 800:NEXT td:PRINT#2:PRINT#2::PR
INT#2,"Turn it on !!":FOR td=1 TO 2000:N
EXT td:CLS#2:GOTO 670
730 PRINT#2:PRINT#2:CLS#2:INPUT#2,"name
:",name$
740 PRINT#8,name$
750 FOR pp=1 TO 16:PRINT#8,h$(pp);",";:N
EXT pp:PRINT#8
760 FOR pp=17 TO 32:PRINT#8,h$(pp);",";:
NEXT pp:PRINT#8
770 FOR pp=33 TO 48:PRINT#8,h$(pp);",";:
NEXT pp:PRINT#8
780 FOR pp=49 TO 64:PRINT#8,h$(pp);",";:
NEXT pp:PRINT#8:GOTO 670
790 pp=INP(&F500) AND 64:RETURN
800 '***  SAVE TO DISC  ***
810 ON ERROR GOTO 1060
820 GOSUB 530:MODE 2:CAT:LOCATE 22,22:IN
PUT " NAME DATA        : ",namedata$:MODE
1:GOSUB 1080:PRINT#2:PRINT#2,"PLEASE WAI
T."
830 OPENOUT namedata$
840 FOR ND=1 TO 64
850 PRINT#9,h$(ND)
860 NEXT ND
870 FOR ND=0 TO 255
880 PRINT#9,dd(ND)
890 NEXT ND
900 CLOSEOUT
910 ND=0:GOTO 620
```

```
920 '***  LOAD FROM DISC  ***
930 ON ERROR GOTO 1060
940 MODE 2:CAT:LOCATE 22,22:INPUT "FILE
NAME       :",namedata$:MODE 1:GOSUB 108
0:PRINT#2,"PLEASE WAIT."
950 OPENIN namedata$
960 FOR ND=1 TO 64
970 INPUT #9,h$(ND)
980 NEXT ND
990 FOR ND=0 TO 255
1000 INPUT #9,dd(ND)
1010 NEXT ND
1020 CLOSEIN
1030 ND=0
1040 GOSUB 1360:GOTO 620
1050 '***  error trap  ***
1060 CLS#2:PRINT#2," ERROR ";ERR:FOR TD=
1 TO 20:SOUND 1,200,15:NEXT:RESUME  140
1070 '********   set up screen *******
1080 WINDOW#0,2,17,2,17:WINDOW#2,25,40,3
,18:WINDOW#1,1,21,20,25:WINDOW#3,23,40,2
0,25:WINDOW#4,20,40,1,1:WINDOW#5,1,40,1,
25:PEN#5,3:PEN#4,2:PEN#3,2:PEN#2,1
1090 PRINT#4,"BUILD  A  SPRITE":PEN#2,1
1100 SYMBOL 255,24,126,126,255,126,126,2
4
1110 LOCATE#5,18,7:PRINT#5,"Sprite"
1120 LOCATE#5,19,3:PRINT#5,"Data :"
1130 PLOT 14,127:DRAW 14,384:DRAW 272,38
4:DRAW 272,127:DRAW 14,127
1140 PLOT 3,254:DRAW 13,254:DRAW 13,256:
DRAW 3,256
1150 PLOT 143,396:DRAW 143,385:DRAW 145,
385:DRAW 145,396
1160 PLOT 273,257:DRAW 285,257 :DRAW 285
,255 :DRAW 273,255
1170 PLOT 143,126:DRAW 143,114:DRAW 145,
114 :DRAW 145,126
1180 PLOT 285,230:DRAW 285,280:DRAW 355,
280:DRAW 355,230:DRAW 285,230
1190 PLOT 300,380:DRAW 560,380:PLOT 300,
376:DRAW 560,376
1200 PEN#1,2:PRINT#1," MENU ":PEN#1,3
1210 PRINT#1,"Move via cursor keys.";
1220 PRINT#1,"'P'  place in memory.";
1230 PRINT#1,"'S'  to save to disc.";
1240 PRINT#1,"'L'   load from disc.";
1250 PRINT#1,"'CTRL-P'  to printer.";
1260 PRINT#3,"   COLOURS"
1270 PRINT#3
1280 PEN#3,1:PRINT#3,"Press 0 for black"
;
1290 PRINT#3," : :  1 for white":PEN#3,2
1300 PRINT#3," : :  2 for red":PEN#3,3
1310 PRINT#3," : :  3 for blue"
1320 PLOT 0,0:DRAW 639,0:DRAW 639,111:DR
AW 0,111:DRAW 0,0
1330 PLOT 390,80:DRAW 525,80
1340 RETURN
1350 '***  print grid after loaded  ***
1360 FOR xx=0 TO 15:d(xx,yy)=dd(n):n=n+1
:NEXT xx
1370 yy=yy+1:IF yy>15 THEN n=0:xx=0:yy=0
:GOTO 1390
1380 GOTO 1360
1390 FOR n=0 TO 255
1400 IF dd(n)=0 THEN dd(n)=0
1410 IF dd(n)=128 OR dd(n)=64 OR dd(n)=3
2 OR dd(n)=16 THEN dd(n)=1:GOTO 1440
1420 IF dd(n)=8 OR dd(n)=4 OR dd(n)=2 OR
 dd(n)=1 THEN dd(n)=2
1430 IF dd(n)=136 OR dd(n)=68 OR dd(n)=3
4 OR dd(n)=17 THEN dd(n)=3
1440 NEXT n:n=0
1450 FOR xx=0 TO 15
1460 IF dd(n)=0 THEN PEN 0:GOTO 1500
1470 IF dd(n)=1 THEN PEN 1:GOTO 1500
1480 IF dd(n)=2 THEN PEN 2:GOTO 1500
1490 IF dd(n)=3 THEN PEN 3:GOTO 1500
1500 LOCATE xx+1,yy+1:PRINT#0,CHR$(255);
1510 n=n+1
1520 NEXT xx
1530 yy=yy+1:IF yy>15 THEN n=0:xx=0:yy=0
:RETURN
1540 GOTO 1450
1550 MODE 2:LIST
1560 '
1570 '
1580 ' When programme is not used with d
isc,then insert and replace these
1590 ' lines below.
1600 'line No 820 = GOSUB 530:WINDOW SWA
P 0,2:CLS#0:INPUT#0, " NAME DATA       :
",namedata$:PRINT#0,"PLEASE WAIT."
1610 'line No 905 = WINDOW SWAP 0,2
1620 'line No 940 = WINDOW SWAP 0,2:CLS#
0:INPUT#0,"FILE NAME        :",namedata$:
PRINT#0,"PLEASE WAIT."
1630 'line No 1025 = WINDOW SWAP 0,2
```

Newcastle Database

from the Newcastle Amstrad User's Group

"Here is a program our group has been working on for a period of time and was written by J.J. Vinopal in collaboration with J. and E. Harwood.
We hope your magazine can publish our work to show other groups what can be done with co-operation and helping each other. We have worked long and hard on this program."

How it Works

10 - 130	Main Program
140 - 250	Instructions
260 - 410	Menu
420 - 520	Delete a record
530 - 890	Sort in alpha order
900 - 1010	Load records from disc
1020 - 1250	Print
1260 - 1340	Graphics
1350 - 1550	Logo
1560 - 2100	Data
2110 - 2290	Graphics
2300 - 2390	Add a record
2400 - 2530	Find a record
2540 - 2630	Save a record
2640 - 2790	Rename fields

```
10 '*******"NEW'CLE DATABASE"*******
20 'This program must be typed exactly
30 'as it is *************************
40 'Otherwise it will not work *******
50 REM (C) J.J.Vinopal + J&E.Harwood
60 REM Newcastle Amstrad Users Group
70 ON BREAK GOSUB 260:ON ERROR GOTO 260
80 REM Protected by COPYRIGHT
90 CLS:CLEAR:d$="J.J.Vinopal":SYMBOL AFT
ER 32:GOSUB 2270
100 RESTORE 1560:p$=" J&E.Harwood"
110 READ r$(0),s$(0),t$(0),u$(0),v$(0),w
$(0)
120 MODE 1:BORDER 0:INK 0,5:INK 1,0:INK
2,31:INK 3,0,23
130 PAPER 0:PEN 2:b5=9*44:b6=6*59:GOSUB
1470
140 m$="  "+CHR$(164)+" "+d$+p$+" ":GOSU
B 1260:GOSUB 2180
150 WINDOW 3,39,6,24:BORDER 9:mn=ASC(o$)
160 PRINT k$:PRINT" This program will st
ore up to 200x6"
170 PRINT k$:PRINT"titles from  your col
lection, store,"
180 PRINT k$:PRINT"stock,address, music,
software,plants"
190 PRINT k$:PRINT",books,etc. Fields ca
n be renamed to"
200 PRINT k$:PRINT"suit  your   storage
 needs  and  be"
210 PRINT k$:PRINT"saved with file on  d
isc. For bigger"
220 PRINT k$:PRINT"storage then 1200 ite
ms you can  use"
230 PRINT k$:PRINT"more files like box1
box2,etc. "i$" JJV "i$
240 v=mm+um+200:WINDOW 1,40,1,25
250 c$="":mn=ABS(mn-111):GOSUB 1230:IF v
<>1 THEN CALL &3000,n
260 m$="  200 x 6 File Database  ":GOSUB
 1260
270 GOSUB 2180:WINDOW 3,39,6,25
280 PRINT:PRINT i$"CREATE"i$" a record..
........ADD.......";:PEN 3:PRINT"C":PEN
1
290 PRINT:PRINT i$"DELETE"i$" a record..
........ERASE.....";:PEN 3:PRINT"D":PEN
1
300 PRINT:PRINT i$"FIND  "i$" a record..
........SELECT....";:PEN 3:PRINT"F":PEN
1
310 PRINT:PRINT i$"RENAME"i$" the record
 fields.CHANGE....";:PEN 3:PRINT"R":PEN
1
320 PRINT:PRINT i$"PRINT "i$" records to
 the... PRINTer...";:PEN 3:PRINT"P":PEN
1
330 PRINT:PRINT i$"LOAD  "i$" records fr
om disc.READ......";:PEN 3:PRINT"L":PEN
1
```

```
340 PRINT:PRINT i$"ALPHA "i$" order reco
rds.....SORT......";:PEN 3:PRINT"A":PEN
1
350 PRINT:PRINT i$"SAVE  "i$" records to
 disc...WRITE.....";:PEN 3:PRINT"S":PEN
1
360 WINDOW 1,40,1,25
370 WHILE op$>"":op$=INKEY$:WEND
380 WHILE op$="":op$=INKEY$:WEND
390 o=INSTR("CDFRPLAS",UPPER$(OP$)):IF o
<=0 OR o>=9 THEN 370
400 ON o GOSUB 2300,420,2400,2640,1020,9
00,530,2540
410 PRINT k$;:GOTO 260
420 m$="  Delete a record  ":GOSUB 1260
430 PRINT i$"       Delete which record";
:INPUT de:PRINT i$:'6 spaces
440 IF de>=nb THEN PRINT"Does not exist
record No.":GOTO 520
450 dt=de:ss=0:GOSUB 1120
460 INPUT"Press Y to confirm deletion :"
,L$:PRINT
470 IF UPPER$(L$)<>"Y"THEN PRINT TAB(15)
;"Not deleted":GOTO 520
480 PRINT k$:FOR j=dt+1 TO nb:r$(j-1)=r$
(j):s$(j-1)=s$(j)
490 t$(j-1)=t$(j):u$(j-1)=u$(j)
500 v$(j-1)=v$(j):w$(j-1)=w$(j):NEXT
510 nb=nb-1:PRINT TAB(18);k$"Deleted"k$
520 GOSUB 1230:RETURN
530 m$="  Alpha sort records  ":GOSUB 12
60
540 PRINT i$:PRINT"        To MENU just [
ENTER]        ":' 7 speces,8 spaces
550 PRINT:INPUT"       Select   FIELD  [
1 - 6]";so:PRINT i$:'7 speces
560 IF so<=0 THEN 260
570 IF so>=7 THEN 550
580 PRINT TAB(8);"Sorting by FIELD  ";so
590 tn=TIME:li%=nb-1:r%=li%:WHILE r%>1
600 r%=r%/2:s%=li%-r%:g%=1:WHILE g%=1:g%
=0
610 ON so GOTO 720,750,780,810,840,870
620 WEND:WEND:tw=TIME-tn:tw=INT(tw/30)/1
0
630 LOCATE 8,12:PRINT"Sort done in";tw;"
seconds"
640 GOSUB 1230:RETURN
650 f$=r$(t%):r$(t%)=r$(u%):r$(u%)=f$
660 f$=s$(t%):s$(t%)=s$(u%):s$(u%)=f$
670 f$=t$(t%):t$(t%)=t$(u%):t$(u%)=f$
680 f$=u$(t%):u$(t%)=u$(u%):u$(u%)=f$
690 f$=v$(t%):v$(t%)=v$(u%):v$(u%)=f$
700 f$=w$(t%):w$(t%)=w$(u%):w$(u%)=f$
710 g%=1:RETURN
720 FOR t%=1 TO s%:u%=r%+t%
730 IF r$(t%)>r$(u%)THEN GOSUB 650
740 NEXT t%:GOTO 620
750 FOR t%=1 TO s%:u%=r%+t%
760 IF s$(t%)>s$(u%)THEN GOSUB 650
770 NEXT t%:GOTO 620
780 FOR t%=1 TO s%:u%=r%+t%
790 IF t$(t%)>t$(u%)THEN GOSUB 650
800 NEXT t%:GOTO 620
810 FOR t%=1 TO s%:u%=r%+t%
820 IF u$(t%)>u$(u%)THEN GOSUB 650
830 NEXT t%:GOTO 620
840 FOR t%=1 TO s%:u%=r%+t%
850 IF v$(t%)>v$(u%)THEN GOSUB 650
860 NEXT t%:GOTO 620
870 FOR t%=1 TO s%:u%=r%+t%
880 IF w$(t%)>w$(u%)THEN GOSUB 650
890 NEXT t%:GOTO 620
900 m$="  Load records from disc  ":GOSU
B 1260
910 PRINT i$" Insert disc press [SPACE]
for catalog."i$;k$;:GOSUB 1240
920 CAT:ON ERROR GOTO 260
930 PRINT i$;" Back to MENU just [ENTER]
 "
940 PRINT " To Load [FILENAME-ENTER] ";i
$;:PEN 3:INPUT nm$:PEN 1
950 IF LEN(nm$)<=0 THEN 260
960 PRINT k$:p=9:OPENIN nm$:INPUT#p,nb
970 FOR j=0 TO nb-1:INPUT#p,r$(j):INPUT#
p,s$(j)
980 INPUT#p,t$(j):INPUT#p,u$(j):INPUT#p,
v$(j)
990 INPUT#p,w$(j):NEXT:CLOSEIN
1000 PRINT"This FILE is ";:PEN 3:PRINT n
b-1;"x6";:PEN 1:PRINT" fields."
1010 GOSUB 1230:RETURN
1020 m$="  Print out ***file***  ":GOSUB
 1260
1030 PRINT"This FILE has ";nb-1;"x6 fiel
ds to print."
1040 PRINT "Is "i$"the printer"i$" conne
cted (Y/N) "i$"[enter]"i$;:INPUT pr$
1050 PRINT i$" STOP printer press [enter
] for second."i$
1060 IF UPPER$(pr$)<>"Y" THEN CLS:GOTO 2
60:ss=0 ELSE ss=8:PEN 1
1070 PRINT#8,CHR$(27);"3";CHR$(28);CHR$(
15);:REM this is the code for most print
ers
1080 PRINT#ss,"NEW-DB":REM for  example
BRORHER-EPSON-SEIKOSHA
1090 PRINT#ss:PRINT #ss:FOR dt=1 TO nb-1
1100 c$=INKEY$:IF c$=""THEN 1110 ELSE 26
0
1110 GOSUB 1120:NEXT:PRINT #ss:PRINT #ss
:GOSUB 1230:RETURN
1120 PEN 3:PRINT#ss,"Record";dt;"is:"
1130 PRINT#ss,"-----------":PEN 1
1140 PRINT#ss,r$(0);TAB(n);":";r$(dt)
1150 c$=INKEY$:IF c$=""THEN 1160 ELSE 26
0
1160 PRINT#ss,s$(0);TAB(n);":";s$(dt)
1170 PRINT#ss,t$(0);TAB(n);":";t$(dt)
1180 PRINT#ss,u$(0);TAB(n);":";u$(dt)
1190 c$=INKEY$:IF c$=""THEN 1200 ELSE 26
0
1200 PRINT#ss,v$(0);TAB(n);":";v$(dt)
1210 PRINT#ss,w$(0);TAB(n);":";w$(dt)
1220 PRINT#ss:RETURN
1230 PEN 3:LOCATE 9,25:PRINT i$" Press S
PACE to continue "i$;k$;
1240 WHILE INKEY$<>" ":WEND
1250 PEN 1:SOUND 1,27,30,15:RETURN
1260 b1=LEN(m$)+4:b2=INT((42-b1)/2)
```

```
1270 CLS:PAPER 1:PEN 2:REM screem titie
plate + printing
1280 LOCATE b2,1:PRINT CHR$(217);STRING$
(b1-2,32);CHR$(219)
1290 LOCATE b2,2:PRINT CHR$(217);" ";m$;
" ";CHR$(219)
1300 LOCATE b2,3:PRINT CHR$(217);STRING$
(b1-2,32);CHR$(219)
1310 PAPER 0:PEN 1:PRINT
1320 b3=b2*16-12:b4=(b2+b1)*16-20
1330 MOVE b3,b5:DRAW b4,b5,3:REM screen
title frame
1340 DRAW b4,b6:DRAW b3,b6:DRAW b3,b5:RE
TURN
1350 GOSUB 1590:'(c) Newcastle Amstrad U
sers Group
1360 LOCATE 18,10:PEN 1:FOR s=65 TO 72:P
RINT CHR$(s);:NEXT
1370 LOCATE 18,11:PRINT CHR$(81);CHR$(82
);CHR$(83);CHR$(84);CHR$(85);CHR$(97);CH
R$(98);CHR$(99)
1380 LOCATE 18,12:PRINT CHR$(119);CHR$(7
3);CHR$(74);CHR$(75);CHR$(76);CHR$(77);C
HR$(78);CHR$(89)
1390 LOCATE 18,13:PRINT CHR$(121);CHR$(2
34);CHR$(79);CHR$(80);CHR$(140);CHR$(100
);CHR$(101);CHR$(89)
1400 LOCATE 18,14:PRINT CHR$(86)+CHR$(86
)+CHR$(87)+CHR$(143)+CHR$(143)+CHR$(143)
+CHR$(88)+CHR$(118)
1410 LOCATE 18,15:um=-53:FOR s=102 TO 10
9:PRINT CHR$(s);:NEXT
1420 LOCATE 18,16:mm=-146:FOR s=110 TO 1
17:PRINT CHR$(s);:NEXT
1430 LOCATE 19,17:PRINT CHR$(213);CHR$(4
9);CHR$(57);CHR$(56);CHR$(54);CHR$(212)
1440 LOCATE 20,18:PRINT CHR$(213);CHR$(1
43);CHR$(143);CHR$(212):LOCATE 21,19
1450 PRINT CHR$(213);CHR$(212):SYMBOL AF
TER 0:RETURN
1460 GOTO 260
1470 LOCATE 2,2:PRINT i$;" ";CHR$(164);"
 ";d$;STRING$(11,32);p$;" ";i$
1480 PEN 3:LOCATE 33,22:PRINT i$"[space]
"i$:PEN 1
1490 LOCATE 17,5:PRINT i$"  NEW-DB  "i$:
LOCATE 17,6:PRINT i$;STRING$(10,32);i$
1500 LOCATE 17,7:PRINT i$" 200  x 6 "i$:
LOCATE 17,8:PRINT i$;STRING$(10,32);i$
1510 LOCATE 17,4:PRINT i$;STRING$(10,32)
;i$
1520 LOCATE 33,21:PRINT"[press]":PEN 1
1530 GOSUB 1350:GOSUB 2100
1540 WHILE INKEY$="":WEND:INK 1,20:INK 0
,9:INK 2,5:INK 3,15,5
1550 RETURN
1560 DATA Field A1,.......2,.......3,Fie
ld B1,.......2,.......3
1570 DATA record,video,books,sport,files
,memory,photos,address
1580 DATA diary,copies,members,movies,mu
sic,software,plants
1590 SYMBOL 119,143,128,128,128,128,128,
128,128
1600 SYMBOL 89,1,1,1,1,1,1,1,1
```

```
1610 SYMBOL 87,3,7,15,7,7,199,199,199
1620 SYMBOL 88,240,240,240,255,255,255,2
55,255
1630 SYMBOL 118,227,227,227,255,255,255,
255,255
1640 SYMBOL 117,7,254,252,248,240,224,19
2,128
1650 SYMBOL 234,254,218,202,210,218,254,
128,128
1660 SYMBOL 86,128,128,128,128,128,227,2
27,227
1670 SYMBOL 116,193,255,255,255,255,255,
255,255
1680 SYMBOL 115,251,255,255,255,255,255,
255,0
1690 SYMBOL 111,130,255,255,255,255,255,
255,255
1700 SYMBOL 112,251,255,255,255,255,255,
255,0
1710 SYMBOL 113,29,255,255,255,255,255,2
55,0
1720 SYMBOL 114,216,255,255,255,255,255,
255,0
1730 SYMBOL 74,210,0,0,0,0,0,0,0:b7=38
1740 SYMBOL 75,224,0,0,0,0,0,0,0:b8=42
1750 SYMBOL 76,25,0,0,0,0,0,0,0:b9=315
1760 SYMBOL 77,47,2,2,7,15,31,63,127
1770 SYMBOL 78,189,0,0,0,128,192,224,240

1780 SYMBOL 79,0,0,0,0,0,0,0,1:c2=326
1790 SYMBOL 80,0,0,0,0,63,127,255,255
1800 SYMBOL 102,255,255,222,206,214,218,
220,222
1810 SYMBOL 103,255,255,130,190,190,190,
134,190
1820 SYMBOL 104,255,255,251,250,250,218,
170,114
1830 SYMBOL 105,255,255,31,238,253,253,2
52,237
1840 SYMBOL 106,255,255,120,183,215,216,
31,215
1850 SYMBOL 107,255,255,192,123,251,251,
123,123
1860 SYMBOL 108,255,255,95,223,223,223,2
23,223
1870 SYMBOL 109,255,255,7,127,127,31,127
,127
1880 SYMBOL 65,255,128,132,138,145,160,1
60,191
1890 SYMBOL 66,255,0,16,24,20,147,144,14
4
1900 SYMBOL 67,255,0,39,104,168,39,32,32
1910 SYMBOL 68,255,0,227,16,0,0,224,16
1920 SYMBOL 69,255,0,251,66,66,66,66,67
1930 SYMBOL 70,255,0,240,8,9,10,18,227
1940 SYMBOL 71,255,0,67,162,18,10,10,250
1950 SYMBOL 72,255,1,241,9,9,9,9,9
1960 SYMBOL 73,115,0,0,0,0,0,0,0
1970 SYMBOL 81,160,160,128,128,137,137,1
37,137
1980 SYMBOL 82,144,144,0,0,59,66,51,10
1990 SYMBOL 83,40,39,0,0,220,18,146,28
2000 SYMBOL 84,16,224,0,0,112,128,96,16
2010 SYMBOL 140,0,1,0,0,255,255,255,255
2020 SYMBOL 85,66,66,0,0,29,33,45,37
```

```
2030 SYMBOL 97,18,10,0,0,207,40,42,200
2040 SYMBOL 98,10,11,0,0,165,165,165,165
2050 SYMBOL 99,9,241,1,1,193,33,193,1
2060 SYMBOL 121,128,128,128,128,128,128,
128,128
2070 SYMBOL 100,255,255,127,127,255,255,
255,255
2080 SYMBOL 101,248,252,240,240,240,240,
240,240
2090 SYMBOL 110,222,127,63,31,15,7,3,1:R
ETURN
2100 FOR sp=1 TO 2:RESTORE 1570:FOR i=1
TO 14:READ kw$
2110 k1=INT(RND*3)+1:PEN k1:SOUND 1,RND*
43,10,15
2120 IF sp=1 THEN kx=INT(RND*9)+2:ky=INT
(RND*17)+3
2130 IF sp=2 THEN kx=INT(RND*9)+27:ky=IN
T(RND*17)+3
2140 LOCATE kx,ky:PRINT kw$:kw$="":NEXT
i,sp
2150 PEN 2:LOCATE 2,24:PRINT i$;STRING$(
20,32)"A U S T R A L I A "
2160 LOCATE 2,22:PRINT STRING$(31,32);i$

2170 CALL &BD25,&F2:FOR w=1 TO 6*&CD:NEX
T:RETURN
2180 MOVE 4,b9+1:DRAW 634,b9+1,2:DRAW 63
4,b8+8:DRAW 4,b8+8
2190 DRAW 4,b9+1:MOVE 10,b7:DRAW 10,330,
3:DRAW 2*b9,330
2200 DRAW 2*b9,b7:DRAW 10,b7:MOVE 14,b8:
DRAW 14,c2,1:DRAW 626,c2
2210 DRAW 626,b8:DRAW 14,b8:RETURN
2220 LOCATE 2,25:PEN 3:PRINT i$" F for M
ore [F]ile or M for [M]enu "i$;k$
2230 a$=INKEY$:IF a$=""THEN 2230

2240 IF a$="f"THEN PEN 1:RETURN
2250 IF a$="m"THEN SOUND 1,22,100,15:GOT
O 260
2260 GOTO 2230
2270 DI:SPEED WRITE 1:ar=&CD:n=9:i$=CHR$
(24):k$=CHR$(7)
2280 DIM r$(ar+1),s$(ar+1):nb=1:o$=(MID$
(d$,9,1))
2290 DIM t$(ar+1),u$(ar+1),v$(ar+1),w$(a
r+1):RETURN
2300 m$="  Create a record  ":GOSUB 1260
2310 IF nb>ar THEN LOCATE 14,8:PRINT"Dat
abase full":GOTO 1230
2320 PRINT i$"Create record number";k$;i
$;nb:PRINT
2330 PRINT r$(0);TAB(n);:INPUT":",r$(nb)
2340 PRINT s$(0);TAB(n);:INPUT":",s$(nb)
2350 PRINT t$(0);TAB(n);:INPUT":",t$(nb)
2360 PRINT u$(0);TAB(n);:INPUT":",u$(nb)
2370 PRINT v$(0);TAB(n);:INPUT":",v$(nb)
2380 PRINT w$(0);TAB(n);:INPUT":",w$(nb)
2390 nb=nb+1:GOSUB 1230:RETURN
2400 m$="  Find a record  ":GOSUB 1260
2410 LOCATE 3,6:PRINT"To"i$" see"i$" all
"nb-1"x6 records"i$" just [ENTER]"k$;i$
2420 LOCATE 3,8:PRINT"OR"i$" Letters "i$
"to be"i$" SEARCHed"i$" for";:INPUT mu$
```

```
2430 mu$=UPPER$(mu$):IF mn<>1 THEN  CALL
 &4006
2440 PEN 1:PRINT k$:FOR se=1 TO nb-mn:L$
=mu$
2450 IF INSTR(UPPER$(r$(se)),L$)THEN GOS
UB 2520
2460 IF INSTR(UPPER$(s$(se)),L$)THEN GOS
UB 2520
2470 IF INSTR(UPPER$(t$(se)),L$)THEN GOS
UB 2520
2480 IF INSTR(UPPER$(u$(se)),L$)THEN GOS
UB 2520
2490 IF INSTR(UPPER$(v$(se)),L$)THEN GOS
UB 2520
2500 IF INSTR(UPPER$(w$(se)),L$)THEN GOS
UB 2520
2510 NEXT:CLS:GOSUB 2220:RETURN
2520 CLS:PRINT:PRINT:dt=se:ss=0:GOSUB 11
20
2530 GOSUB 2220:L$=CHR$(0):RETURN
2540 m$="  Save records to disc  ":GOSUB
 1260
2550 PRINT"   Save file with ";nb-1;"x6
records"
2560 PRINT:PRINT"   Put a DISC in the dr
ive and"
2570 PRINT:PRINT"    "i$;" Back to MENU
just [ENTER] "
2580 LOCATE 5,10: INPUT " To Save [FILEN
AME-ENTER]",nm$:PRINT i$;
2590 IF LEN(nm$)<=0 THEN 260
2600 OPENOUT nm$:PRINT#9,nb:FOR j=0 TO n
b-1:PRINT#9,r$(j)
2610 PRINT#9,s$(j):PRINT#9,t$(j):PRINT#9
,u$(j)
2620 PRINT #9,v$(j):PRINT #9,w$(j):NEXT

2630 CLOSEOUT:GOSUB 1230:RETURN
2640 m$="  Rename record fields  ":GOSUB
 1260
2650 e$="FIELD ":h$=" Change to:"
2660 PRINT:PRINT e$;r$(0);h$;:INPUT c$
2670 IF c$>"" THEN r$(0)=c$
2680 PRINT:PRINT e$;s$(0);h$;:INPUT c$
2690 IF c$>"" THEN s$(0)=c$
2700 PRINT:PRINT e$;t$(0);h$;:INPUT c$
2710 IF c$>"" THEN t$(0)=c$
2720 PRINT:PRINT e$;u$(0);h$;:INPUT c$
2730 IF c$>"" THEN u$(0)=c$
2740 PRINT:PRINT e$;v$(0);h$;:INPUT c$
2750 IF c$>"" THEN v$(0)=c$
2760 PRINT:PRINT e$;w$(0);h$;:INPUT c$
2770 IF c$>"" THEN w$(0)=c$
2780 GOSUB 1230:RETURN
2790 RUN
```

Adventurer's Attic

by Philip Riley

This month Philip turns his attentions to planning an Adventure. Plus we have some answers for some 'lost souls'

If you ever attempt to write your own adventure games, there are two golden rules which you must always remember.

1) Plan your adventure
2) Save memory

In fact planning your game well at the beginning will help to save you memory later on.

So what is the best method for planning a game? Simple - draw a map! This inevitably helps you later on when writing the game itself. I use the system drawn below. (Figure 1)

As you can see we have a block of nine squares each with a number and a description. Each one of these squares is a location in the game. The dotted line is the direction that you can move from one location to the next. You will also notice that location 4 has a key and location 6 has a gun. These are items that can be picked up and used, but more on this later. Always make sure that you mark everything on the map including all of the traps and obstacles that must be overcome to complete the game.

The next problem is, how will the game know which way you can move, and which way you cannot. The easiest answer to this is read the information in from data. All you will need are 16 numbers to cover every possible combinations moves. And just to make things easier I have drawn them all below. (See Figure 2)

To read the appropriate data for the nine locations on our map you will need the following program.

```
10 DIM YP(9)
20 FOR T=1 TO 9:READ
YP(T):NEXT
30 DATA 1,3,15,5,9,1,12,14,9
```

YP(N) is the variable that will be used throughout the program to keep tabs on your position. As you can see we have 9 numbers in the data statement - one for each location.

All that you need to do is check this number against the direction

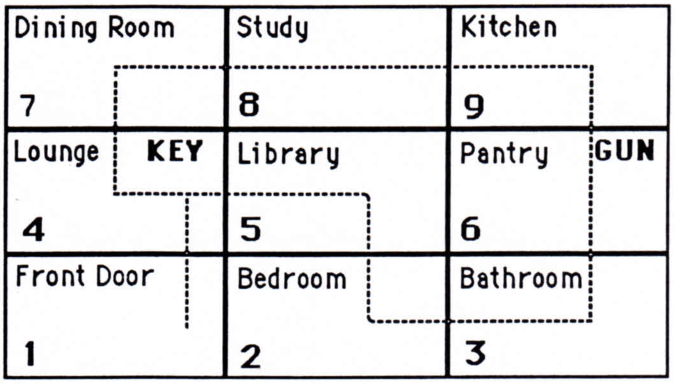


Figure 1

that you wish to move. To do this you will need to use 4 IF...THEN statements like the one below. This checks to see if you can move north or not.

```
 100 IF(YP(N)70) AND (YP(N)<9)
THEN N=N+3:GOTO500
```

Please note in the above line you must also check the direction that you are trying to move to. If you use the up cursor key to move north you would add -

```
AND(A$=CHR$(240))
```

The variable N will correspond with the numbers on the map. For instance if N=4 then YP(N)=5 . So in the above line you can move north. Thus N=N+3 and we are now at location 7. Where we cannot move north after the 4 IF...THEN statements you must add a line something like this:

```
 140 PRINT"YOU CANNOT GO THAT
WAY":GOTO xxx
```

This is the default line if you try to move in a direction that you are not allowed. Another point to remember is if you have an obstacle that stops you from moving in a certain direction, (for instance a locked door in location 8 that stops you moving into location 9) you must put the appropriate number into the data statement. In this case the 8th number would not be a 14 it would be a 15. When the door has been unlocked you simply say YP(8)=14 and you will now be able to move from location 8 to 9.

The last point to remember is that when moving up or down on your own map the variable N will not always be plus or minus 3. The amount that is added or subtracted depends on the width of your map.

Next month we will take a look at picking up and dropping items.

Don't forget that this column is also here to help you write adventures not just to solve them. If you have any problems writing your own game let me know.

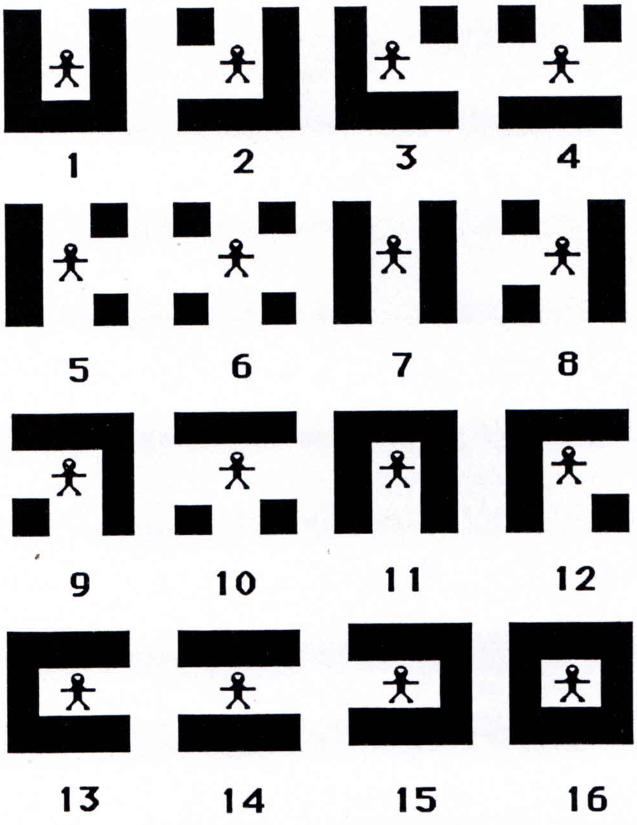


Figure 2

For the 'lost souls' . . . .

Kevin Cryer of Chisholm, ACT writes:

Let's get this column (Adventurer's Attic) going so that us adventurers get looked after as well. In answer to R. Coggins of Morayfield who is stuck in 'Jewels of Babylon', all you have to do is type CLIMB OUT OF BOAT.

As of this date, I have almost proceeded as far as Anthony Eden of Kincumber, and when I solve his dilemma I will let you know how.

Darren Robinson of Reservoir will go a lot further in 'The Trial of Arnold Blackwood' by following one or the other of two possible solutions to get past the dog.

1. Knowing that Lord Erebus *hates* dogs he can 'SHOOT THE DOG', or

2. He can go in another direction and by-pass the dog - both directions will lead to the same part of Lord Erebus' estate.

In answer to Jason Clark's problem in 'Classic Adventure', Barry Klein of Bulleen, Vic provides the following information - to get past the troll you must throw (or possibly drop) him a treasure. If you throw the right treasure, you can get it back later. Something on the other side of the bridge will let you return past the troll without surrendering another treasure. You must drop ALL the treasure (including the pirate's treasure chest) in the building before you can complete the game. Note: there is a rather nasty one point puzzle to solve somewhere in the game.

Characters in Colour

by Peter and Michael Doutch

In last month's issue I presented a Programmable Character Generator (PCG) Designer. This month I shall take things a little further by adding colour to a picture.

Figure 1.1 shows a picture of an apple made up of four squares.

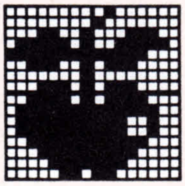


Figure 1.1

If you were to re-define it, using the PCG Designer I hope, it would consist of one colour. For green screen owners this would probably not cause a problem, but others may wish to take advantage of their colour monitors.

Before we start adding colours, you must first be aware of some special control characters.

Number	Symbol	Key	Function
08	←	Ctrl H	Back Space
09	→	Ctrl I	Move Forward
10	↓	Ctrl J	Move Cursor Down
11	↑	Ctrl K	Move Cursor Up
14	⊗	Ctrl N	As PAPER command
15	⊙	Ctrl O	As PEN command
22	⊓	Ctrl V	Transparent Mode
25	⬢	Ctrl Y	As SYMBOL command
26	ʕ	Ctrl Z	As WINDOW command

The above codes are those you are most likely to use. Page 3-8, Chapter 7 of the 664 Instruction manual has all the codes and their respective keys, but doesn't tell you what they look like.

Let's investigate the character codes more closely.

First, character code 22 - transparent mode. If we type a character on the screen then press the DEL key, it will erase the character which was behind the cursor. By putting the computer in transparent mode, we can write over the top of the character keeping the old character underneath. (This is explained on Page 51, Chapter 8 of the 664 Instruction manual). Transparent mode is the trick to adding different colours to a character.

Second, character 08 - equivalent of pressing the DEL key. When in transparent mode, the character behind the cursor will not disappear.

Third, and the most important code in this exercise is character 15 - equivalent to the PEN command. By following this character with a colour number will change the current colour.

So how do you use these codes? Taking a line from Telly Tennis in the Appendix of the manual provides a good example.

```
270  bat$="|"+CHR$(8)+CHR$(10)+"|"
```

Breaking this line down it means - draw the character "|", go back one space so that the cursor is over the "|", then go down one line and draw another "|". All this is kept in the string "bat$".

However, the line can be shortened as:

```
270 bat$="|←↓|"
```

It has exactly the same effect, and the two "arrows" are created by holding down the control key while pressing another key. This method is especially useful when developing long programs and saving space may be important.

Now back to control codes in respect of multi-coloured characters. In Basic, it is not possible to use a single character to specify more than one colour. So to specify three different colours in one character we need to re-define three characters.

Taking the top two squares of the apple (Figure 1.2), we could use three different colours - Green (colour 7) for the stem, Bright Green (colour 18) for the leaves and Bright Red (colour 6) for the small amount of apple at the bottom of each square.

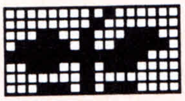
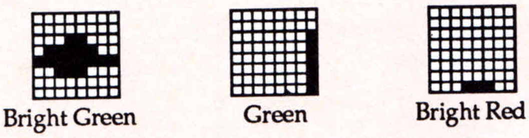


Figure 1.2

Now looking at just the first square, there are three elements to be allocated different colours. Figure 2.1 identifies those elements.



Bright Green

Green

Bright Red

Figure 2.1

The coding to define these is:

```
10   SYMBOL AFTER 90
20   SYMBOL 91,0,0,24,124,255, 56,0,0
:REM LEAVES (Bright Green - 10)
30   SYMBOL 92,0,0,1,1,1,1,1,1
:REM STEM (Green - 9)
40   SYMBOL  93,0,0,0,0,0,0,0,28
:REM APPLE (Bright Red - 6)
```

and to check what you have done :

```
100   PRINT " [ \] "
```

If it looks the same as Figure 2.1, you can go onto the next step.

At the start of this article I talked about transparent mode which was activated by using control code 22 (Ctrl V). Delete line 100 and add this line to your program.

```
100   TRANS$=CHR$(22)+CHR$(1):
 OFF$=CHR$(22)+CHR$(0)
```

When you run the program again this will put the screen in transparent mode. Now identify the leaves, stem and part of the apple as follows:

```
110 LEAVES$="["
120 STEM$="\"
130 APPLE$="]"
```

Then print them:

```
140 PRINT LEAVES$;STEM$;APPLE$
```

This is all very well, but we still have three different squares. Type in the following lines:

```
5 INK 1,6:INK 2,9:INK 3,18:INK 0,0: REM
 RED, GREEN, B/GREEN, BLACK
7 PAPER 0:BORDER 0
8 RED$=CHR$(15)+CHR$(1): GREEN$=
 CHR$(15)+CHR$(2)
9 BGREEN$=CHR$(15)+CHR$(3): BACKSPACE$
 =CHR$(8)
```

As you can see I have used control codes (15) to establish the colours in their strings and (8) to put a backspace in its own string.

If you run the program and typed PRINT RED$ the computer will set the ink colour to Red. If you type PRINT "a";BACKSPACE$ the computer will print the "a" then backspace over it - probably too fast for you to notice it.

Delete line 140 and type the following:

```
140 FIRSTCOLOUR$=BGREEN$+LEAVES$
 +BACKSPACE$
150 SECONDCOLOUR$=GREEN$+STEM$
 +BACKSPACE$
160 THIRDCOLOUR$=RED$+APPLE$
170 PRINT TRANS$;FIRSTCOLOUR$;
 SECONDCOLOUR$;THIRDCOLOUR$;OFF$
```

When you run the finished program you will get a single square displayed on the screen in the three different coloured shapes.

To save memory (and time) there is an even shorter way of doing the same thing as above. Instead of using the CHR$( ) command, we use their symbols which can be found at the beginning of this article.

Delete the following lines: 8, 9, 110, 120, 130, 140, 150, 160 and 170. Type in the following:

```
110 APPLE$="⊙3[←⊙2\←⊙1]"
120 PRINT TRANS$;APPLE$;OFF$
```

Much shorter isn't it, but if you attempt to list this to a printer you may well get some peculiar results. (See 'Listing Control Codes embedded in strings' by Petr Lukes - Issue 17 June 1986).

With this article I have included two listings which will draw the complete apple with all its colours. The first listing uses the CHR$( ) commands. The second uses their picture equivalents (what they look like on the screen when you press the right keys). Notice the difference in length between the two programs.

Listing One

```
5 INK 1,6:INK 2,9:INK 3,18:INK 0,0: REM
 RED, GREEN, B/GREEN, BLACK
7 PAPER 0:BORDER 0
8 RED$=CHR$(15)+CHR$(1): GREEN$=
 CHR$(15)+CHR$(2)
9 BGREEN$=CHR$(15)+CHR$(3): BACKSPACE$
 =CHR$(8)
10 SYMBOL AFTER 40
20 SYMBOL 91,0,0,24,124,255,56,0,0: REM
 LEAVES  -  BRIGHT GREEN 18
30 SYMBOL 92,0,0,1,1,1,1,1,1: REM STEM
  -  GREEN 9
40 SYMBOL 93,0,0,0,0,0,0,0,28: REM APPLE
-  BRIGHT RED 6
50 SYMBOL 94,0,0,28,62,252,112,0,0: REM
LEAVES  -  BRIGHT GREEN 18
60 SYMBOL 95,64,128,0,0,0,0,0,0: REM
STEM  -  GREEN 9
70 SYMBOL 96,0,0,0,0,0,0,0,112: REM
APPLE  -  BRIGHT RED 6
80 SYMBOL 97,61,127,127,127,63,31,15,6:
REM APPLE  -  RED 6
90 SYMBOL 98,120,252,204,204,248,240,
224,192: REM APPLE  -  RED 6
100  TRANS$=CHR$(22)+CHR$(1):
 OFF$=CHR$(22)+CHR$(0)
110 LEAVES$="[":LEAVES2$="↑"
120 STEM$="\":STEM2$="_"
130 APPLE$="]":APPLE2$="`"
140 BOTTOMAPPLE$="ab"
150 FIRSTCOLOUR$=BGREEN$+LEAVES$
 +BACKSPACE$
160 SECONDCOLOUR$=GREEN$+STEM$
 +BACKSPACE$
170 THIRDCOLOUR$=RED$+APPLE$
180 FIRSTCOLOUR2$=BGREEN$+LEAVES2$
 +BACKSPACE$
190 SECONDCOLOUR2$=GREEN$+STEM2$
 +BACKSPACE$
200 THIRDCOLOUR2$=RED$+APPLE2$
210 PRINT TRANS$;FIRSTCOLOUR$;
 SECONDCOLOUR$;THIRDCOLOUR$;
220 PRINT FIRSTCOLOUR2$;
 SECONDCOLOUR2$;THIRDCOLOUR2$;OFF$
230 PRINT RED$;BOTTOMAPPLE$
240 REM
250 REM END OF LISTING ONE
260 REM
```

Listing Two

```
10 MODE 1
20 INK 1,6:INK 2,9:INK 3,18:INK 0,0: REM
RED, GREEN, B/GREEN, BLACK
30 SYMBOL AFTER 40
40 SYMBOL 91,0,0,24,124,255,56,0,0: REM
 LEAVES  -  BRIGHT GREEN 18
50 SYMBOL 92,0,0,1,1,1,1,1,1: REM STEM
  -  GREEN 9
60 SYMBOL 93,0,0,0,0,0,0,0,28: REM APPLE
-  BRIGHT RED 6
70 SYMBOL 94,0,0,28,62,252,112,0,0: REM
 LEAVES  -  BRIGHT GREEN 18
80 SYMBOL 95,64,128,0,0,0,0,0,0: REM
 STEM  -  GREEN 9
90 SYMBOL 96,0,0,0,0,0,0,0,112: REM
 APPLE  -  BRIGHT RED 6
100 SYMBOL 97,61,127,127,127,63,31,15,6:
REM APPLE  -  RED 6
110 SYMBOL 98,120,252,204,204,248,240,
224,192: REM APPLE  -  RED 6
120 TRANS$=" Л1":OFF$=" ЛO"
130 APPLE$=" ⊙3[←⊙2\←⊙1]" :APPLE2$="
 ⊙3↑←⊙2_←⊙1`"
140 BOTTOMAPPLE$=" 1"+"ab"
150 PRINT TRANS$;APPLE$;APPLE2$
160 PRINT BOTTOMAPPLE$;OFF$
170 REM
180 REM END OF LISTING 2
190 REM
```

If you have any questions regarding this article or last month's PCG Designer, Peter would be happy to answer them. Address your letters to Peter Doutch, c/o The Amstrad User.

Using ams-FORTH with disc drive(s)

by Petr Lukes

ams-Forth provides an opportunity to try out FORTH without spending too much money or trying to install a system from the very beginning. It is fig-FORTH, similar to the '4TH' which was used in the tutorial series in the UK User, March to August 1985.

It is supplied on a protected tape, but it requires minimal effort to make it usable with disc, without violating he protection. The program is supplied with a routine and full instructions on how to copy the system, indeed an admonishment to make a working copy and preserve the original (although the label states that copying is not authorised).

FORTH is designed for customization, so it quite easy to incorporate a routine to access external (bar |) commands. It is limited to commands without parameters (it cannot deal with |ERA, |REN, etc.), but it will allow switching between disc and tape or between drives. When compiled and saved with SYS-SAVE, it will become part of the system and it will be impossible to FORGET it.

The program auto-starts, which causes a shut-down of all ROMS. The routine re-initialises them by testing the disc command 'A': if it is not found, all the background ROMs are initialised by calling KL ROM WALK, which is then disabled by placing RETURN in the first byte of its jump block; the system would get lost if the initialisation was performed more than once.

KL FIND COMMAND is used to return the ROM number in reg. C and the routine address in reg. HL; on entry, HL must contain the address of the command name whose last character must have the bit 7 set (128 added to it). KL FAR PCHL will then execute the routine if it exists.

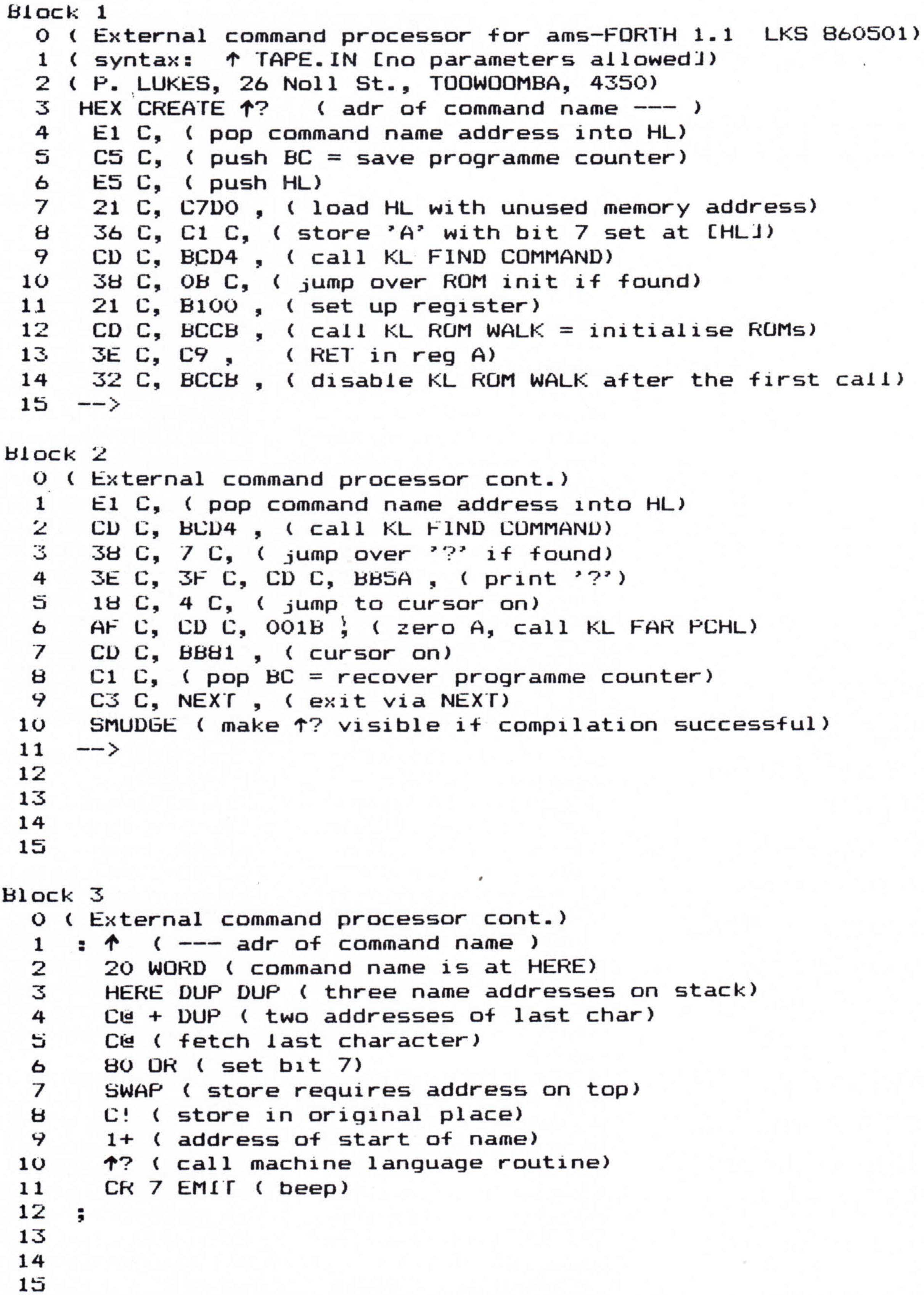
As it stands, the routine will initialise all background ROMs, if any, but it could be altered to initialise only selected ROMs by using KL INIT BACK. From my reading, it appears that the routine should work equally well with the tape '4TH' (if it is not already built in). The recommended procedure is to take the original ams-FORTH, make a backup copy, and put both tapes away. With the program still in memory, carefully type in the three screens of code, omitting the remarks (it is advisable to n CLEAR each screen first) and save them with SCR-SAVE. Then compile the code (by 1 LOAD): if there are any errors, correct them and SCR-SAVE again, then re-compile by entering the line

'SP! FORTH SMUDGE FORGET ↑? 1 LOAD'.

This will tidy up the stack and delete the wrong words. Only when compilation is successful, try a command: make sure that a disc is in drive A, enter '↑DISC', then 'SYS-SAVE'; to the prompt 'Filename?' type 'AMSFORTH' and hold down the space bar. Writing to disc should start as soon as the 16 character name buffer is filled up (this also applies to SCR-SAVE/LOAD).

If all went well, reset the computer by CTRL+SHIFT+ESC, then 'run"amsforth' to bring the program back from disc with the command processor now incorporated in the system. The auto-start would have shut down the disc ROM, so enter '↑DISC' after the sign-on message to re-activate it.

(See next page)



```
Block 1
 ( External command processor for ams-FORTH 1.1  LKS 860501)
 ( syntax:  ↑ TAPE.IN [no parameters allowed])
 ( P. LUKES, 26 Noll St., TOOWOOMBA, 4350)
 HEX CREATE ↑?    ( adr of command name --- )
  E1 C, ( pop command name address into HL)
  C5 C, ( push BC = save programme counter)
  E5 C, ( push HL)
  21 C, C7D0 , ( load HL with unused memory address)
  36 C, C1 C, ( store 'A' with bit 7 set at [HL])
  CD C, BCD4 , ( call KL FIND COMMAND)
  38 C, 0B C, ( jump over ROM init if found)
  21 C, B100 , ( set up register)
  CD C, BCCB , ( call KL ROM WALK = initialise ROMs)
  3E C, C9 ,    ( RET in reg A)
  32 C, BCCB , ( disable KL ROM WALK after the first call)
 -->

Block 2
 ( External command processor cont.)
  E1 C, ( pop command name address into HL)
  CD C, BCD4 , ( call KL FIND COMMAND)
  38 C, 7 C, ( jump over '?' if found)
  3E C, 3F C, CD C, BB5A , ( print '?')
  18 C, 4 C, ( jump to cursor on)
  AF C, CD C, 001B , ( zero A, call KL FAR PCHL)
  CD C, BB81 , ( cursor on)
  C1 C, ( pop BC = recover programme counter)
  C3 C, NEXT , ( exit via NEXT)
  SMUDGE ( make ↑? visible if compilation successful)
 -->




Block 3
 ( External command processor cont.)
 : ↑  ( --- adr of command name )
   20 WORD ( command name is at HERE)
   HERE DUP DUP ( three name addresses on stack)
   C@ + DUP ( two addresses of last char)
   C@ ( fetch last character)
   80 OR ( set bit 7)
   SWAP ( store requires address on top)
   C! ( store in original place)
   1+ ( address of start of name)
   ↑? ( call machine language routine)
   CR 7 EMIT ( beep)
 ;
```

A definitive Black-Jack

from Tibor Gyore and Tim Baldock

Not another Black-Jack I hear you say! I'm afraid it is, after all , how many ways can you play this game. Black-Jack or not you are going to love this version. Our Pontoon will automatically reshuffle the deck of cards just like the real thing and the cards look about as authentic as they come.

We have incorporated some features which other versions sadly lack - the option to use a joystick or keyboard to operate the game, the program does not mind if you use either or both during play. This program uses a system of menus to make it child's play to control, and in the event that you manage to select the wrong option, you can always return to the previous menu or confirm your selection before the function is carried out.

The other feature you'll appreciate is knowing that the Banker cannot cheat. He has no knowledge whatsoever of the contents of your hand.

Our version of Pontoon shuffles the pack of cards when you reach the end of the deck. Of course the cards on the table at the time are not put back into the pack to be reshuffled.

But the best thing about this program that sets it apart from all others is the way in which we have presented it. While realistic cards are nothing special, our Pontoon is crammed with goodies. The most noticeable is the new character set, really putting you in the old time atmosphere.

Here is a brief description of the program. You will notice that the program has been assembled in a logical fashion so that you will be able to key in the program by sections.

200 - 300	Initialise the pens etc.
340	Load redefined symbols
380	Goto the instructions before the game
400 - 470	Setting up the deck of cards
490 - 530	Initialise to playing table
550 - 1550	All this just to deal the player his (or her) cards
1570 - 1640	Defines the suit for the card being dealt
1660 - 1770	Subroutine to select a card from the pack
1790 - 1960	Routine to reshuffle the deck of cards
1980 - 2140	Drawing up of the card
2160 - 2330	Pattern on the back of the cards
2530 - 3400	The data for the symbol redefinition
3420 - 5000	Draw up each card from the Ace to the Ten
5020 - 5540	Setup of the Jack card and the symbol data for his picture
5560 - 6060	Symbol data for the Queen's picture and card setup
6080 - 6550	Drawing up of the King's card and the symbol data for his picture
6570 - 7140	Subroutine for the information and menu windows system
7160 - 7570	Logic to drive the main menu window
7590 - 7910	Logic for the betting menu
7930 - 8160	Logic to drive the Ace menu
8180 - 8360	Logic for confirmation menu
8380 - 8560	Logic for Yes/No options
8570 - 9330	The Banker's cards are dealt in this section
9350 - 10590	This section decides who wins the hand and displays a message in the information window
10610 - 10760	Routine to ask you for another hand
10780 - 11290	Front screen and border around the screen
11310 - 12760	Instructions for the game
12780 - 12970	Setup of the Cashier's window
12990 - 13140	Subroutine to buy more chips
13160 - 13330	Reporting to the Cashier at the end of the game

Throughout the program, we have used easy to understand variables. They usually describe the variable's function or for what it is used. The variable declarations are self explanatory. The game also has a comprehensive set of rules on how to play.

As with previous long programs (Black-Jack consumes 36k), this program listing has been divided into two sections. The final part will be published in next month's issue. Tape subscribers please note that the complete program will appear on a twenty minute tape issued with the August 1986 edition.

```
10 '        (C) COPYRIGHT 1986
20 '    BY - INOVATIVE SOFTWARE INC.
30 '
40 ' THIS PROGRAM MAY NOT BE REPRODUCED
50 ' OR CHANGED INCLUDING ALL GRAPHICS
60 ' IN WHOLE OR PART WITHOUT PRIOR
70 ' CONSENT OF -
80 ' INOVATIVE SOFTWARE INC. AND
90 ' STRATEGY PUBLICATIONS.
100 '
110 'NOTE - ANY VIOLATION OF THIS
120 '       COPYRIGHT WILL RESULT IN
130 '       FUTURE TITLES BEING
140 '       ABANDONED !
150 '
160 '
170 'THIS PROGRAM REMAINS THE PROPERTY
180 'OF INOVATIVE SOFTWARE INC.
190 '
200 '--INITIALIZE--
210 '
220 '
230 CLEAR
240 DEFINT A-Z
250 INK 0,0
260 INK 1,26
270 INK 2,22
280 INK 3,6
290 BORDER 22
300 RANDOMIZE TIME
310 '
320 '--LOAD GRAPHICS--
330 '
340 GOSUB 2370
350 '
360 ' INTRODUCTION & INSTRUCTIONS
370 '
380 GOSUB 10800:T=0
390 '
400 '--SET UP DECK--
410 '
420 MONEY=20:DIM PACK(52):CARDSLEFT=52:X
4=33:Y4=1
430 PLAYTOT=0:BANKTOT=0:BETTOT=0
440 FOR Q=1 TO 52
450 PACK(Q)=0
460 NEXT
470 X=1:Y=1:X1=32:Y1=2:BPONTOON=0:BFIVEU
NDER=0
480 '
490 '--SET UP PLAYING SCREEN--
500 '
510 PAPER 2
520 CLS
530 TOGGLE1=1:R=0:GOSUB 6590:TOGGLE1=0
540 '
550 ' --PLAYERS CARDS--
560 '
570 '--DEAL PLAYERS CARD NO. (1)--
580 '
590 GOSUB 2000
600 FOR DELAY=1 TO 1000:NEXT
610 GOSUB 1680
620 GOSUB 1600
630 GOSUB 3470
```

```
640 A=CARD
650 IF F>10 THEN F=10
660 IF F<>1 THEN 710
670 GOSUB 7960
680 PLAYTOT=PLAYTOT+ACE
690 ACE=0
700 GOTO 720
710 PLAYTOT=PLAYTOT+F
720 T=1:GOSUB 7610
730 GOSUB 8620
740 Y=Y-13
750 '
760 '--DEAL PLAYERS CARD NO. (2)--
770 '
780 X=X+5
790 R=0:GOSUB 2000
800 FOR DELAY=1 TO 1000:NEXT
810 GOSUB 8740
820 Y=Y-13
830 GOSUB 1680
840 GOSUB 1600
850 GOSUB 3470
860 B=CARD
870 IF F>10 THEN F=10
880 IF F<>1 THEN 930
890 GOSUB 7960
900 PLAYTOT=PLAYTOT+ACE
910 ACE=0
920 GOTO 940
930 PLAYTOT=PLAYTOT+F
940 IF PLAYTOT=21 THEN PONTOON=1:GOTO 87
90
950 Y1=2:R=1:GOSUB 7180
960 '
970 '--DEAL PLAYERS CARD NO. (3)--
980 '
990 X=X+5
1000 R=0:GOSUB 2000
1010 FOR DELAY=1 TO 1000:NEXT
1020 GOSUB 1680
1030 GOSUB 1600
1040 GOSUB 3470
1050 C=CARD
1060 IF F>10 THEN F=10
1070 IF F<>1 THEN 1120
1080 GOSUB 7960
1090 PLAYTOT=PLAYTOT+ACE
1100 ACE=0
1110 GOTO 1140
1120 PLAYTOT=PLAYTOT+F
1130 IF PLAYTOT>21 THEN 9320
1140 IF PLAYTOT=21 THEN 8790
1150 Y1=2:R=1:GOSUB 7180
1160 '
1170 '--DEAL PLAYERS CARD NO. (4)--
1180 '
1190 X=X+5
1200 R=0:GOSUB 2000
1210 FOR DELAY=1 TO 1000:NEXT
1220 GOSUB 1680
1230 GOSUB 1600
1240 GOSUB 3470
1250 D1=CARD
1260 IF F>10 THEN F=10
1270 IF F<>1 THEN 1320
```

```
1280 GOSUB 7960
1290 PLAYTOT=PLAYTOT+ACE
1300 ACE=0
1310 GOTO 1330
1320 PLAYTOT=PLAYTOT+F
1330 IF FLAYTOT=21 THEN 8790
1340 IF PLAYTOT>21 THEN 9320
1350 Y1=2:R=1:GOSUB 7180
1360 '
1370 '--DEAL PLAYERS CARD NO. (5)--
1380 '
1390 X=X+5
1400 R=0:GOSUB 2000
1410 FOR DELAY=1 TO 1000:NEXT
1420 GOSUB 1680
1430 GOSUB 1600
1440 GOSUB 3470
1450 E=CARD
1460 IF F>10 THEN F=10
1470 IF F<>1 THEN 1520
1480 GOSUB 7960
1490 PLAYTOT=PLAYTOT+ACE
1500 ACE=0
1510 GOTO 1530
1520 PLAYTOT=PLAYTOT+F
1530 IF PLAYTOT>21 THEN 9320
1540 IF PLAYTOT<=21 THEN FIVEUNDER=1 ELS
E FIVEUNDER=0
1550 GOTO 8790
1560 '
1570 '--SET A$,B$,C$,D$ TO GRAPHICS--
1580 '--OF SUIT OF CARD CHOSEN      --
1590 '
1600 IF SUIT=1 THEN PEN 3:A1$=CHR$(228):
A$=CHR$(137):B$=CHR$(138):C$=CHR$(139):D
$=CHR$(140)
1610 IF SUIT=2 THEN PEN 0:A1$=CHR$(229):
A$=CHR$(141):B$=CHR$(142):C$=CHR$(143):D
$=CHR$(144)
1620 IF SUIT=3 THEN PEN 0:A1$=CHR$(226):
A$=CHR$(145):B$=CHR$(146):C$=CHR$(147):D
$=CHR$(148)
1630 IF SUIT=4 THEN PEN 3:A1$=CHR$(227):
A$=CHR$(133):B$=CHR$(134):C$=CHR$(135):D
$=CHR$(136)
1640 RETURN
1650 '
1660 '--PICK CARD FROM PACK--
1670 '
1680 CARD=INT(RND(1)*52+1)
1690 IF PACK(CARD)=1 THEN 1680
1700 PACK(CARD)=1
1710 F=CARD-13*INT(CARD/13)
1720 IF F=0 THEN F=13
1730 SUIT=(INT((CARD-1)/13)+1)
1740 IF SUIT=0 THEN 1730
1750 CARDSLEFT=CARDSLEFT-1
1760 IF CARDSLEFT<1 THEN 1820
1770 RETURN
1780 '
1790 '--RESHUFFLE DECK WHEN ALL CARDS--
1800 '--USED EXCEPT CARDS ON THE TABLE--
1810 '
1820 FOR I=1 TO 52
1830 IF A=I AND PACK(A)=1 THEN U=U+1:GOT
O 1940
1840 IF B=I AND PACK(B)=1 THEN U=U+1:GOT
O 1940
1850 IF C=I AND PACK(C)=1 THEN U=U+1:GOT
O 1940
1860 IF D1=I AND PACK(D1)=1 THEN U=U+1:G
OTO 1940
1870 IF E=I AND PACK(E)=1 THEN U=U+1:GOT
O 1940
1880 IF F1=I AND PACK(F1)=1 THEN U=U+1:G
OTO 1940
1890 IF G=I AND PACK(G)=1 THEN U=U+1:GOT
O 1940
1900 IF H=I AND PACK(H)=1 THEN U=U+1:GOT
O 1940
1910 IF I1=I AND PACK(I1)=1 THEN U=U+1:G
OTO 1940
1920 IF J=I AND PACK(J)=1 THEN U=U+1:GOT
O 1940
1930 PACK(I)=0
1940 NEXT
1950 CARDSLEFT=52-U:U=0
1960 RETURN
1970 '
1980 '--DRAW MAIN BORDER OF CARD--
1990 '
2000 WINDOW #4,X,X+9,Y,Y+11:PAPER #4,1
2010 PEN 0:PAPER 1:CLS#4
2020 LOCATE X,Y:PRINT CHR$(149);
2030 FOR I=1 TO 8
2040 LOCATE X+I,Y:PRINT CHR$(150);
2050 NEXT
2060 LOCATE X+9,Y:PRINT CHR$(151)
2070 FOR I=1 TO 10
2080 LOCATE X,Y+I:PRINT CHR$(152):LOCATE
 X+9,Y+I:PRINT CHR$(153)
2090 NEXT
2100 LOCATE X,Y+11:PRINT CHR$(154)
2110 FOR I=1 TO 8
2120 LOCATE X+I,Y+11:PRINT CHR$(156)
2130 NEXT
2140 LOCATE X+9,Y+11:PRINT CHR$(155)
2150 '
2160 '-- DRAW FACE DECORATIONS--
2170 '
2180 PEN 3
2190 LOCATE X+1,Y+1:PRINT CHR$(157)
2200 FOR I=2 TO 7
2210 LOCATE X+I,Y+1:PRINT CHR$(158)
2220 NEXT
2230 LOCATE X+8,Y+1:PRINT CHR$(161)
2240 FOR I=2 TO 9
2250 LOCATE X+1,Y+I
2260 PRINT CHR$(163);CHR$(160);CHR$(160)
;CHR$(160);CHR$(160);CHR$(160);CHR$(160)
;CHR$(162)
2270 NEXT
2280 LOCATE X+1,Y+10:PRINT CHR$(166)
2290 FOR I=2 TO 7
2300 LOCATE X+I,Y+10:PRINT CHR$(165)
2310 NEXT
2320 LOCATE X+8,Y+10:PRINT CHR$(167)
2330 RETURN
2340 '
2350 '--MAIN DATA FOR GRAPHICS--
```

```
2360 '
2370 SYMBOL AFTER 35
2380 SYMBOL 36,20,63,84,62,21,126,20,0
2390 SYMBOL 38,56,68,68,56,69,66,61,0
2400 SYMBOL 40,4,8,16,16,16,8,4,0
2410 SYMBOL 41,32,16,8,8,8,16,32,0
2420 SYMBOL 47,1,2,4,8,16,32,64,0
2430 SYMBOL 48,28,34,34,34,34,34,28,0
2440 SYMBOL 49,8,24,40,8,8,8,62,0
2450 SYMBOL 50,28,34,2,28,32,32,62,0
2460 SYMBOL 51,28,34,2,28,2,34,28,0
2470 SYMBOL 52,4,12,20,36,62,4,4,0
2480 SYMBOL 53,62,32,60,2,2,34,28,0
2490 SYMBOL 54,28,34,32,60,34,34,28,0
2500 SYMBOL 55,62,2,4,8,8,8,8,0
2510 SYMBOL 56,28,34,34,28,34,34,28,0
2520 SYMBOL 57,28,34,34,30,2,34,28,0
2530 SYMBOL 65,4,12,20,36,61,102,36,0
2540 SYMBOL 66,46,81,33,62,33,81,46,0
2550 SYMBOL 67,30,33,74,68,64,33,30,0
2560 SYMBOL 68,60,82,17,17,17,81,62,0
2570 SYMBOL 69,30,33,64,120,64,33,30,0
2580 SYMBOL 70,63,68,36,7,4,20,12,0
2590 SYMBOL 71,30,33,64,68,66,33,30,0
2600 SYMBOL 72,51,85,17,63,81,17,17,0
2610 SYMBOL 73,24,40,8,24,12,8,24,0
2620 SYMBOL 74,3,5,1,33,65,34,28,0
2630 SYMBOL 75,51,82,20,24,20,18,19,0
2640 SYMBOL 76,48,80,16,16,18,17,30,0
2650 SYMBOL 77,116,74,73,73,73,73,41,0
2660 SYMBOL 78,60,34,33,33,33,101,34,0
2670 SYMBOL 79,28,34,65,65,65,34,28,0
2680 SYMBOL 80,60,18,17,30,16,80,48,0
2690 SYMBOL 81,28,34,65,65,69,34,29,0
2700 SYMBOL 82,60,18,17,30,24,85,50,0
2710 SYMBOL 83,30,33,64,62,1,66,60,0
2720 SYMBOL 84,63,68,36,4,4,20,12,0
2730 SYMBOL 85,17,49,81,17,17,19,13,0
2740 SYMBOL 86,17,49,81,17,17,10,4,0
2750 SYMBOL 87,34,65,73,73,73,85,34,0
2760 SYMBOL 88,65,34,20,8,20,34,65,0
2770 SYMBOL 89,49,81,17,19,13,1,14,0
2780 SYMBOL 97,255,255,24,24,24,24,24,24
2790 SYMBOL 98,224,248,24,12,12,12,12,12
2800 SYMBOL 99,24,24,24,31,31,24,24,24
2810 SYMBOL 100,12,12,28,248,224,1,1,1
2820 SYMBOL 101,0,0,0,62,255,193,128,128
2830 SYMBOL 102,0,0,0,27,159,220,216,216
2840 SYMBOL 103,1,1,1,225,249,29,13,13
2850 SYMBOL 104,128,128,128,248,249,131,
131,131
2860 SYMBOL 105,0,0,0,124,255,131,1,1
2870 SYMBOL 106,0,0,0,7,31,184,176,176
2880 SYMBOL 107,0,0,0,195,243,59,27,27
2890 SYMBOL 108,0,0,0,126,255,131,1,1
2900 SYMBOL 109,0,0,0,0,0,128,128,128
2910 SYMBOL 110,24,24,24,24,24,56,240;19
2
2920 SYMBOL 111,1,1,1,1,1,1,0,0
2930 SYMBOL 112,128,128,128,128,128,193,
255,62
2940 SYMBOL 113,216,216,216,216,216,216,
152,24
2950 SYMBOL 114,13,13,13,13,13,13,12,12
2960 SYMBOL 115,131,131,131,131,131,195,
241,48
2970 SYMBOL 116,1,1,1,1,1,131,255,124
2980 SYMBOL 117,176,176,176,176,176,184,
31,7
2990 SYMBOL 118,27,27,27,27,27,59,243,19
5
3000 SYMBOL 119,1,1,1,1,1,1,1,1
3010 SYMBOL 120,128,128,128,128,128,128,
128,128
3020 SYMBOL 133,0,1,3,7,15,31,63,127
3030 SYMBOL 134,0,128,192,224,240,248,25
2,254
3040 SYMBOL 134,0,0,128,192,224,240,248,
252
3050 SYMBOL 135,255,127,63,31,15,7,3,1
3060 SYMBOL 136,254,252,248,240,224,192,
128,0
3070 SYMBOL 137,0,56,124,254,255,255,255
,127
3080 SYMBOL 138,0,56,124,254,254,254,254
,252
3090 SYMBOL 139,127,63,63,31,15,7,3,1
3100 SYMBOL 140,252,248,248,240,224,192,
128,0
3110 SYMBOL 141,0,3,7,15,31,63,127,127
3120 SYMBOL 142,0,128,192,224,240,248,25
2,252
3130 SYMBOL 143,255,255,255,127,124,57,3
,7
3140 SYMBOL 144,254,254,254,252,124,56,1
28,192
3150 SYMBOL 145,0,7,15,15,7,3,57,127
3160 SYMBOL 146,0,192,224,224,192,128,56
,252
3170 SYMBOL 147,255,255,255,125,57,1,3,7
3180 SYMBOL 148,254,254,254,124,56,0,128
,192
3190 SYMBOL 149,15,63,112,224,192,192,19
2,192
3200 SYMBOL 150,255,255,0,0,0,0,0,0
3210 SYMBOL 151,240,252,14,7,3,3,3,3
3220 SYMBOL 152,192,192,192,192,192,192,
192,192
3230 SYMBOL 153,3,3,3,3,3,3,3,3
3240 SYMBOL 154,192,192,192,192,224,112,
63,15
3250 SYMBOL 155,3,3,3,3,7,14,252,240
3260 SYMBOL 156,0,0,0,0,0,0,255,255
3270 SYMBOL 157,255,128,191,160,160,161,
162,164
3280 SYMBOL 158,129,66,60,129,129,60,86,
171
3290 SYMBOL 159,153,66,37,38,37,38,67,15
3
3300 SYMBOL 160,85,170,85,170,85,170,85,
170
3310 SYMBOL 161,255,1,253,5,5,133,69,37
3320 SYMBOL 162,153,194,100,164,100,164,
66,153
3330 SYMBOL 163,153,66,37,38,37,38,67,15
3
3340 SYMBOL 164,153,66,37,38,37,38,67,15
3
3350 SYMBOL 165,213,106,60,129,129,60,66
,129
```

```
3360 SYMBOL 166,164,162,161,160,160,191,
128,255
3370 SYMBOL 167,37,69,133,5,5,253,1,255
3380 SYMBOL 226,56,124,56,84,254,254,84,
56
3390 SYMBOL 229,16,56,124,254,254,84,16,
56
3400 RETURN
3410 '
3420 '--DRAW FACE OF THE CARD--
3430 '
3440 '
3450 '-- FACE OF THE ACE--
3460 '
3470 WINDOW #5,X+1,X+8,Y+1,Y+10:PAPER #5
,1
3480 CLS#5
3490 IF F<>1 THEN 3600
3500 LOCATE X+1,Y+1:PRINT"A"
3510 LOCATE X+1,Y+2:PRINT A1$
3520 LOCATE X+4,Y+5:PRINT A$;B$
3530 LOCATE X+4,Y+6:PRINT C$;D$
3540 LOCATE X+8,Y+9:PRINT"A"
3550 LOCATE X+8,Y+10:PRINT A1$
3560 RETURN
3570 '
3580 '--FACE OF NO. (2)--
3590 '
3600 IF F<>2 THEN 3730
3610 LOCATE X+1,Y+1:PRINT"2"
3620 LOCATE X+1,Y+2:PRINT A1$
3630 LOCATE X+4,Y+2:PRINT A$;B$
3640 LOCATE X+4,Y+3:PRINT C$;D$
3650 LOCATE X+4,Y+8:PRINT A$;B$
3660 LOCATE X+4,Y+9:PRINT C$;D$
3670 LOCATE X+8,Y+9:PRINT A1$
3680 LOCATE X+8,Y+10:PRINT"2"
3690 RETURN
3700 '
3710 '--FACE OF NO. (3)--
3720 '
3730 IF F<>3 THEN 3880
3740 LOCATE X+1,Y+1:PRINT"3"
3750 LOCATE X+1,Y+2:PRINT A1$
3760 LOCATE X+4,Y+2:PRINT A$;B$
3770 LOCATE X+4,Y+3:PRINT C$;D$
3780 LOCATE X+4,Y+5:PRINT A$;B$
3790 LOCATE X+4,Y+6:PRINT C$;D$
3800 LOCATE X+4,Y+8:PRINT A$;B$
3810 LOCATE X+4,Y+9:PRINT C$;D$
3820 LOCATE X+8,Y+9:PRINT A1$
3830 LOCATE X+8,Y+10:PRINT"3"
3840 RETURN
3850 '
3860 '--FACE OF NO. (4)--
3870 '
3880 IF F<>4 THEN 4010
3890 LOCATE X+1,Y+1:PRINT"4"
3900 LOCATE X+1,Y+2:PRINT A1$
3910 LOCATE X+2,Y+2:PRINT A$;B$;SPC(2);A
$;B$
3920 LOCATE X+2,Y+3:PRINT C$;D$;SPC(2);C
$;D$
3930 LOCATE X+2,Y+8:PRINT A$;B$;SPC(2);A
$;B$
```

```
3940 LOCATE X+2,Y+9:PRINT C$;D$;SPC(2);C
$;D$
3950 LOCATE X+8,Y+9:PRINT A1$
3960 LOCATE X+8,Y+10:PRINT"4"
3970 RETURN
3980 '
3990 '--FACE OF NO. (5)--
4000 '
4010 IF F<>5 THEN 4160
4020 LOCATE X+1,Y+1:PRINT"5"
4030 LOCATE X+1,Y+2:PRINT A1$
4040 LOCATE X+2,Y+2:PRINT A$;B$;SPC(2);A
$;B$
4050 LOCATE X+2,Y+3:PRINT C$;D$;SPC(2);C
$;D$
4060 LOCATE X+4,Y+5:PRINT A$;B$
4070 LOCATE X+4,Y+6:PRINT C$;D$
4080 LOCATE X+2,Y+8:PRINT A$;B$;SPC(2);A
$;B$
4090 LOCATE X+2,Y+9:PRINT C$;D$;SPC(2);C
$;D$
4100 LOCATE X+8,Y+9:PRINT A1$
4110 LOCATE X+8,Y+10:PRINT"5"
4120 RETURN
4130 '
4140 '--FACE OF NO. (6)--
4150 '
4160 IF F<>6 THEN 4310
4170 LOCATE X+1,Y+1:PRINT"6"
4180 LOCATE X+1,Y+2:PRINT A1$
4190 LOCATE X+2,Y+2:PRINT A$;B$;SPC(2);A
$;B$
4200 LOCATE X+2,Y+3:PRINT C$;D$;SPC(2);C
$;D$
4210 LOCATE X+2,Y+5:PRINT A$;B$;SPC(2);A
$;B$
4220 LOCATE X+2,Y+6:PRINT C$;D$;SPC(2);C
$;D$
4230 LOCATE X+2,Y+8:PRINT A$;B$;SPC(2);A
$;B$
4240 LOCATE X+2,Y+9:PRINT C$;D$;SPC(2);C
$;D$
4250 LOCATE X+8,Y+9:PRINT A1$
4260 LOCATE X+8,Y+10:PRINT"6"
4270 RETURN
4280 '
4290 '--FACE OF NO. (7)--
4300 '
4310 IF F<>7 THEN 4480
4320 LOCATE X+1,Y+1:PRINT"7"
4330 LOCATE X+1,Y+2:PRINT A1$
4340 LOCATE X+2,Y+2:PRINT A$;B$;SPC(2);A
$;B$
4350 LOCATE X+2,Y+3:PRINT C$;D$;SPC(2);C
$;D$
4360 LOCATE X+2,Y+5:PRINT A$;B$;SPC(2);A
$;B$
4370 LOCATE X+2,Y+6:PRINT C$;D$;SPC(2);C
$;D$
4380 LOCATE X+2,Y+8:PRINT A$;B$;SPC(2);A
$;B$
4390 LOCATE X+2,Y+9:PRINT C$;D$;SPC(2);C
$;D$
4400 LOCATE X+4,Y+7:PRINT A$;B$
4410 LOCATE X+4,Y+8:PRINT C$;D$
```

```
4420 LOCATE X+8,Y+9:PRINT A1$
4430 LOCATE X+8,Y+10:PRINT"7"
4440 RETURN
4450 '
4460 '--FAVE OF NO. (8)--
4470 '
4480 IF F<>8 THEN 4670
4490 LOCATE X+1,Y+1:PRINT"8"
4500 LOCATE X+1,Y+2:PRINT A1$
4510 LOCATE X+2,Y+2:PRINT A$;B$;SPC(2);A
$;B$
4520 LOCATE X+2,Y+3:PRINT C$;D$;SPC(2);C
$;D$
4530 LOCATE X+2,Y+5:PRINT A$;B$;SPC(2);A
$;B$
4540 LOCATE X+2,Y+6:PRINT C$;D$;SPC(2);C
$;D$
4550 LOCATE X+2,Y+8:PRINT A$;B$;SPC(2);A
$;B$
4560 LOCATE X+2,Y+9:PRINT C$;D$;SPC(2);C
$;D$
4570 LOCATE X+4,Y+3:PRINT A$;B$
4580 LOCATE X+4,Y+4:PRINT C$;D$
4590 LOCATE X+4,Y+7:PRINT A$;B$
4600 LOCATE X+4,Y+8:PRINT C$;D$
4610 LOCATE X+8,Y+9:PRINT A1$
4620 LOCATE X+8,Y+10:PRINT"8"
4630 RETURN
4640 '
4650 '--FACE OF NO. (9)--
4660 '
4670 IF F<>9 THEN 4840
4680 LOCATE X+1,Y+1:PRINT"9"
4690 LOCATE X+1,Y+2:PRINT A1$
4700 FOR I=2 TO 8 STEP 2
4710 LOCATE X+2,Y+I:PRINT A$;B$;SPC(2);A
$;B$
4720 NEXT
4730 FOR I=3 TO 9 STEP 2
4740 LOCATE X+2,Y+I:PRINT C$;D$;SPC(2);C
$;D$
4750 NEXT
4760 LOCATE X+4,Y+5:PRINT A$;B$
4770 LOCATE X+4,Y+6:PRINT C$;D$
4780 LOCATE X+8,Y+9:PRINT A1$
4790 LOCATE X+8,Y+10:PRINT"9"
4800 RETURN
4810 '
4820 '--FACE OF NO. (10)--
4830 '
4840 IF F<>10 THEN 5040
4850 LOCATE X+1,Y+1:PRINT"10"
4860 LOCATE X+1,Y+2:PRINT A1$
4870 FOR I=2 TO 8 STEP 2
4880 LOCATE X+8,Y+9:PRINT A1$
4890 LOCATE X+2,Y+I:PRINT A$;B$;SPC(2);A
$;B$
4900 NEXT
4910 FOR I=3 TO 9 STEP 2
4920 LOCATE X+2,Y+I:PRINT C$;D$;SPC(2);C
$;D$
4930 NEXT
4940 LOCATE X+4,Y+3:PRINT A$;B$
4950 LOCATE X+4,Y+4:PRINT C$;D$
4960 LOCATE X+4,Y+7:PRINT A$;B$
```

```
4970 LOCATE X+4,Y+8:PRINT C$;D$
4980 LOCATE X+8,Y+9:PRINT A1$
4990 LOCATE X+7,Y+10:PRINT"10"
5000 RETURN
5010 '
5020 '--FACE OF THE JACK--
5030 '
5040 IF F<>11 THEN 5580
5050 '
5060 '--GRAPHICS FOR JACK REPLACES--
5070 '--KING OR QUEENS WHEN DEALT --
5080 '
5090 SYMBOL 97,255,129,129,129,64,64,32,
32
5100 SYMBOL 98,255,240,240,240,248,248,2
48,120
5110 SYMBOL 99,255,255,255,255,255,126,1
26,126
5120 SYMBOL 100,255,15,15,15,31,31,31,30
5130 SYMBOL 101,255,129,129,129,2,2,4,4
5140 SYMBOL 102,16,16,8,8,4,4,2,3
5150 SYMBOL 103,120,120,60,60,60,28,28,2
55
5160 SYMBOL 104,126,126,60,60,60,60,60,2
55
5170 SYMBOL 105,30,30,60,60,60,56,56,255
5180 SYMBOL 106,8,8,16,16,32,32,64,192
5190 SYMBOL 107,2,2,5,5,10,10,21,21
5200 SYMBOL 108,170,170,85,85,170,170,85
,85
5210 SYMBOL 109,160,160,97,98,160,160,96
,96
5220 SYMBOL 110,0,112,136,60,236,60,8,0
5230 SYMBOL 111,0,0,0,0,0,0,1,1
5240 SYMBOL 112,64,64,32,32,16,16,8,8
5250 SYMBOL 113,42,42,85,85,170,170,85,8
5
5260 SYMBOL 114,192,192,64,128,128,128,0
,0
5270 SYMBOL 115,0,0,0,0,0,0,0,2
5280 SYMBOL 116,4,4,2,98,132.56,192,128
5290 SYMBOL 117,170,170,84,84,168,168,80
,80
5300 SYMBOL 118,78,48,1,7,12,0,0,0
5310 SYMBOL 119,64,96,192,224,32,64,64,3
2
5320 SYMBOL 120,42,42,85,85,170,170,85,6
3
5330 SYMBOL 121,170,170,85,85,170,170,85
,255
5340 SYMBOL 122,160,160,64,112,142,193,2
48,255
5350 SYMBOL 123,0,0,0,0,1,130,114,14
5360 SYMBOL 124,0,0,0,248,7,0,0,0
5370 SYMBOL 125,16,16,16,32,192,0,0,0
5380 SYMBOL 126,255,15,1,0,0,0,0,0
5390 SYMBOL 127,225,255,255,127,15,0,0,0
5400 SYMBOL 128,0,128,128,192,224,0,0,0
5410 LOCATE X+1,Y+1:PRINT"J"
5420 LOCATE X+1,Y+2:PRINT A$;B$
5430 LOCATE X+1,Y+3:PRINT C$;D$
5440 LOCATE X+3,Y+2:PRINT"abcde"
5450 LOCATE X+3,Y+3:PRINT"fghij"
5460 LOCATE X+2,Y+4:PRINT" klmnp"
5470 LOCATE X+2,Y+5:PRINT"oqlrst"
5480 LOCATE X+2,Y+6:PRINT"klu vw"
```

```
5490 LOCATE X+2,Y+7:PRINT"xyz{|}"
5500 LOCATE X+4,Y+8:PRINT CHR$(126);CHR$
(127);CHR$(128)
5510 LOCATE X+7,Y+8:PRINT A$;B$
5520 LOCATE X+7,Y+9:PRINT C$;D$
5530 LOCATE X+8,Y+10:PRINT"J"
5540 RETURN
5550 '
5560 '--FACE OF THE QUEEN--
5570 '
5580 IF F<>12 THEN 6070
5590 '
5600 '--GRAPHICS OF QUEEN REPLACES--
5610 '--KING OR JACKS WHEN DEALT  --
5620 '
5630 SYMBOL 97,0,0,0,0,0,31,120,149
5640 SYMBOL 98,0,0,0,0,255,93,235,246
5650 SYMBOL 99,0,0,0,0,0,224,120,173
5660 SYMBOL 100,0,0,0,0,0,0,60,228
5670 SYMBOL 101,1,3,6,15,11,14,29,27
5680 SYMBOL 102,255,173,218,111,215,189,
86,187
5690 SYMBOL 103,110,125,215,61,242,107,2
45,191
5700 SYMBOL 104,190,118,204,205,153,155,
59,255
5710 SYMBOL 105,104,200,208,144,160,32,6
4,192
5720 SYMBOL 106,63,50,109,127,94,235,181
,221
5730 SYMBOL 107,171,189,247,156,236,88,2
48,184
5740 SYMBOL 108,0,112,136,60,236,60,8,0
5750 SYMBOL 109,64,64,32,32,16,16,8,8
5760 SYMBOL 110,1,1,1,3,3,2,7,5
5770 SYMBOL 111,232,176,80,224,224,192,6
4,192
5780 SYMBOL 112,4,4,2,98,132,56,192,128
5790 SYMBOL 113,6,11,13,14,31,26,21,63
5800 SYMBOL 114,239,190,221,119,218,188,
244,88
5810 SYMBOL 115,64,128,128,0,0,0,0,0
5820 SYMBOL 116,0,0,1,7,12,0,0,0
5830 SYMBOL 117,64,96,192,224,32,64,64,3
2
5840 SYMBOL 118,54,59,109,119,222,255,10
6,63
5850 SYMBOL 119,117,222,237,87,253,171,8
5,255
5860 SYMBOL 120,176,224,224,96,220,171,8
5,170
5870 SYMBOL 121,0,0,0,0,1,130,114,174
5880 SYMBOL 122,0,0,0,248,7,0,0,0
5890 SYMBOL 123,32,32,32,64,128,0,0,0
5900 SYMBOL 124,245,14,1,0,0,0,0,0
5910 SYMBOL 125,85,170,213,122,15,0,0,0
5920 SYMBOL 126,0,128,128,192,224,0,0,0
5930 LOCATE X+1,Y+1:PRINT"Q"
5940 LOCATE X+1,Y+2:PRINT A$;B$
5950 LOCATE X+1,Y+3:PRINT C$;D$
5960 LOCATE X+4,Y+2:PRINT"abcd"
5970 LOCATE X+3,Y+3:PRINT"efghi"
5980 LOCATE X+3,Y+4:PRINT"jfklm"
5990 LOCATE X+2,Y+5:PRINT"nfgo p"
6000 LOCATE X+2,Y+6:PRINT"qgrstu"
```

```
6010 LOCATE X+2,Y+7:PRINT"vwxyz{"
6020 LOCATE X+4,Y+8:PRINT"|}";CHR$(126)

6030 LOCATE X+7,Y+8:PRINT A$;B$
6040 LOCATE X+7,Y+9:PRINT C$;D$
6050 LOCATE X+8,Y+10:PRINT"Q"
6060 RETURN
6070 '
6080 '--FACE OF THE KING--
6090 '
6100 '
6110 '--GRAPHICS OF KING REPLACES--
6120 '--QUEEN OR JACKS WHEN DEALT--
6130 '
6140 SYMBOL 97,255,135,131,153,92,73,35,
39
6150 SYMBOL 98,255,231,195,153,60,153,19
5,231
6160 SYMBOL 99,255,225,193,153,58,140.19
6,228
6170 SYMBOL 100,23,19,11,9,5,4,2,3
6180 SYMBOL 101,231,231,231,231,231,231,
231,255
6190 SYMBOL 102,232,200,208,144,160,32,6
4,192
6200 SYMBOL 103,2,2,5,5,10,10,21,21
6210 SYMBOL 104,170,170,85,85,170,170,85
,85
6220 SYMBOL 105,160,160,97,98,160,160,96
,96
6230 SYMBOL 106,0,112,136,60,236,60,8,0
6240 SYMBOL 107,0,0,0,0,0,0,1,1
6250 SYMBOL 108,64,64,32,32,16,16,8,8
6260 SYMBOL 109,42,42,85,85,170,170,85,8
5
6270 SYMBOL 110,192,192,64,128,128,128,0
,0
6280 SYMBOL 111,0,0,0,0,0,0,0,2
6290 SYMBOL 112,4,4,2,98,132,56,192,128
6300 SYMBOL 113,170,170,84,84,168,168,80
,80
6310 SYMBOL 114,78,48,1,7,28,0,15,53
6320 SYMBOL 115,64,96,192,224,32,64,192,
96
6330 SYMBOL 116,42,42,85,85,170,170,85,6
3
6340 SYMBOL 117,170,170,85,85,170,170,85
,255
6350 SYMBOL 118,160,160,64,112,142,193,2
48,255
6360 SYMBOL 119,0,0,0,0,1,130,114,14
6370 SYMBOL 120,234,93,254,235,61,43,22,
5
6380 SYMBOL 121,160,224,224,208,112,160,
192,0
6390 SYMBOL 122,255,15,1,0,0,0,0,0
6400 SYMBOL 123,225,255,255,127,15,0,0,0
6410 SYMBOL 124,0,128,128,192,224,0,0,0
6420 LOCATE X+1,Y+1:PRINT"K"
6430 LOCATE X+1,Y+2:PRINT A$;B$
6440 LOCATE X+1,Y+3:PRINT C$;D$
6450 LOCATE X+3,Y+2:PRINT"abbbc"
6460 LOCATE X+3,Y+3:PRINT"deeef"
6470 LOCATE X+2,Y+4:PRINT" ghijl"
6480 LOCATE X+2,Y+5:PRINT"kmhnop"
```

```
6490 LOCATE X+2,Y+6:PRINT"ghq rs"
6500 LOCATE X+2,Y+7:PRINT"tuvwxy"
6510 LOCATE X+2,Y+8:PRINT"  z{¦"
6520 LOCATE X+7,Y+8:PRINT A$;B$
6530 LOCATE X+7,Y+9:PRINT C$;D$
6540 LOCATE X+8,Y+10:PRINT"K"
6550 RETURN
6560 '
6570 '--SET UP MENU & INFORMATION WINDOW
--
6580 '
6590 WINDOW #7,X4,X4+7,Y4,Y4+6
6600 PAPER #7,1
6610 PAPER 1:PEN 0
6620 CLS #7
6630 LOCATE X4,Y4:PRINT CHR$(149)
6640 FOR I=X4+1 TO X4+6
6650 LOCATE I,Y4:PRINT CHR$(150)
6660 NEXT
6670 LOCATE X4+7,Y4:PRINT CHR$(151)
6680 FOR I=Y4+1 TO Y4+5
6690 LOCATE X4,I:PRINT CHR$(152):LOCATE
X4+7,I:PRINT CHR$(153)
6700 NEXT
6710 LOCATE X4,Y4+6:PRINT CHR$(154)
6720 LOCATE X4+7,Y4+6:PRINT CHR$(155)
6730 FOR I=X4+1 TO X4+6
6740 LOCATE I,Y4+6:PRINT CHR$(156)
6750 NEXT
6760 IF TOGGLE=1 THEN RETURN
6770 IF TOGGLE1=1 THEN 6970
6780 WINDOW #6,X4,X4+7,Y4+8,Y4+12
6790 PAPER #6,1:PEN #6,0
6800 CLS #6
6810 PEN 0:PAPER 1
6820 LOCATE X4,Y4+8:PRINT CHR$(149)
6830 FOR I=X4+1 TO X4+6
6840 LOCATE I,Y4+8:PRINT CHR$(150)
6850 NEXT
6860 LOCATE X4+7,Y4+8:PRINT CHR$(151)
6870 FOR I=Y4+9 TO Y4+11
6880 LOCATE X4,I:PRINT CHR$(152):LOCATE
X4+7,I:PRINT CHR$(153)
6890 NEXT
6900 LOCATE X4,Y4+12:PRINT CHR$(154)
6910 LOCATE X4+7,Y4+12:PRINT CHR$(155)
6920 FOR I=X4+1 TO X4+6
6930 LOCATE I,Y4+12:PRINT CHR$(156)
6940 NEXT
6950 PEN 0:PAPER 1
6960 IF TOGGLE=1 THEN RETURN
6970 WINDOW #3,25,40,15,22:PAPER #3,1
6980 CLS #3
6990 LOCATE 25,15:PRINT CHR$(149)
7000 FOR I=26 TO 39
7010 LOCATE I,15:PRINT CHR$(150)
7020 NEXT
7030 LOCATE 40,15:PRINT CHR$(151)
7040 FOR I=16 TO 21
7050 LOCATE 25,I:PRINT CHR$(152):LOCATE
40,I:PRINT CHR$(153)
7060 NEXT
7070 LOCATE 25,22:PRINT CHR$(154)
7080 FOR I=26 TO 39
7090 LOCATE I,22:PRINT CHR$(156)
7100 NEXT
7110 LOCATE 40,22:PRINT CHR$(155)
7120 IF R=2 THEN RETURN
7130 PEN 0:PAPER 1
7140 LOCATE 26,21:PRINT"CHIPS";MONEY
7150 '
7160 '--MAIN MENU DRIVER--
7170 '
7180 IF Y1>6 THEN Y1=2
7190 IF Y1<2 THEN Y1=6
7200 IF Y1=2 THEN PEN 1 ELSE PEN 0
7210 IF Y1=2 THEN PAPER 0 ELSE PAPER 1
7220 LOCATE X4+1,Y4+1:PRINT SPC(1);"BUY"
;SPC(2)
7230 IF Y1=4 THEN PEN 1 ELSE PEN 0
7240 IF Y1=4 THEN PAPER 0 ELSE PAPER 1
7250 LOCATE X4+1,Y4+3:PRINT SPC(1);"SIT"
;SPC(2)
7260 PEN 0:PAPER 1
7270 IF Y1=6 THEN PEN 1 ELSE PEN 0
7280 IF Y1=6 THEN PAPER 0 ELSE PAPER 1
7290 LOCATE X4+1,Y4+5:PRINT SPC(1);"FLIP
";SPC(1)
7300 IF R=0 OR TOGGLE=1 OR TOGGLE1=1 THE
N RETURN
7310 IF (INKEY(0)=0 OR JOY(0)=1) AND Y1>
1 THEN FOR DELAY=1 TO 50:NEXT:Y1=Y1-2
7320 IF (INKEY(2)=0 OR JOY(0)=2)AND Y1<7
 THEN FOR DELAY=1 TO 50:NEXT:Y1=Y1+2
7330 IF INKEY(9)=0 OR JOY(0)=16 THEN FOR
 DELAY=1 TO 300:NEXT:GOTO 7360
7340 GOTO 7180
7350 RETURN
7360 PAPER 1:PEN 0
7370 Y2=9:TOGGLE=1:GOSUB 6780:TOGGLE=0:G
OSUB 8200
7380 IF CANCEL=1 THEN CANCEL=0:Y1=2:GOTO
 7180
7390 IF Y1=2 THEN 7610
7400 IF Y1=4 THEN 7420
7410 IF Y1=6 THEN 7500
7420 IF PLAYTOT>=16 THEN 8790
7430 LOCATE 26,16:PRINT"YOU CAN'T SIT"
7440 LOCATE 26,17:PRINT"UNDER"
7450 PEN 3:LOCATE 32,17:PRINT"16":PEN 0
7460 FOR DELAY=1 TO 3000:NEXT
7470 LOCATE 26,16:PRINT SPC(13)
7480 LOCATE 26,17:PRINT SPC(8)
7490 GOTO 7180
7500 IF PLAYTOT>=12 THEN RETURN
7510 LOCATE 26,16:PRINT"YOU CAN'T FLIP"
7520 LOCATE 26,17:PRINT"UNDER"
7530 PEN 3:LOCATE 32,17:PRINT"12":PEN 0
7540 FOR DELAY=1 TO 3000:NEXT
7550 LOCATE 26,16:PRINT SPC(14)
7560 LOCATE 26,17:PRINT SPC(8)
7570 GOTO 7180
```

Quite a deal to type in, isn't it ?!!
The last six pages of the program will be presented next month - so don't throw your hand in yet

Impressions of PC '86

from Julian Tipper

PC '86 was held in the Exhibition Buildings, Melbourne from June 1 - 4. It was an event combining the office technology and the communications Show, resulting in an interesting spectacle of the technological age in which we live.

Amstrad was represented by Mitsubishi Electric-AWA and also by The Amstrad User which brought together volunteers from the various clubs around Melbourne. The pace was pretty hectic on Sunday, and at times it resembled a cattle market! The star of the show was the Amiga, but Amstrad got their money's worth. It wasn't until Tuesday that the pace on the Mitsubishi/AWA stand settled.

The Amstrads on display numbered about 16; with the 464's, 6128's and 8256/8512 'Joyces' sprawled around the whole of the stand. I was a little disappointed as this must be the first show I have been to where Amstrad have revealed nothing new.

Meanwhile, at the Amstrad User stand, it was on for young and old. People flocked to this small but important stand. We received questions from young kids to people in their sixties and seventies, all anxious to get their answers from the user group members. Copies of the Australian and English user magazines were for sale and these sold at a good pace thoughout the four days of the show.

In summing up, I couldn't help but get the impression that people in Victoria were not happy with the service they were *not* getting. The culprit was a major retailer who, until very recently, sold Amstrads exclusively. We had numerous complaints through the Show, so I hope this retailer takes note. It was interesting to observe how many people were keen to join user groups for the 8256 and 8512 models. (Any tips or information about the business Amstrads would be welcomed by the Editor as there are a lot of people out there thirsting for knowledge).

Well, that was it for this year - see you at PC '87.

Back Copies

The Amstrad User (Australian)

		Magazine	Tape
Issue 1	February 1985	$3.50	$5.00
Issue 2	March 1985	$3.50	$5.00
Issue 3	April 1985	$3.50	$5.00
Issue 4	May 1985	$3.50	$5.00
Issue 5	June 1985	Out of Print	$5.00
Issue 6	July 1985	$3.50	$5.00
Issue 7	August 1985	$3.50	$5.00
Issue 8	September 1985	$3.50	$5.00
Issue 9	October 1985 **(20 copies only)**	$3.50	$5.00
Issue 10	November 1985	$4.00	$5.00
Issue 11	December 1985	$4.00	$5.00
Issue 12	January 1986	$4.00	$5.00
Issue 13	February 1986	$4.00	$5.00
Issue 14	March 1986	$4.00	$5.00
Issue 15	April 1986	$4.00	$5.00
Issue 16	May 1986	$4.00	$5.00
Issue 17	June 1986	$4.00	$5.00

Twelve issue subscription: magazine only	$35.00
Twelve issue subscription: magazine plus tape	$75.00

Year Disc No. 1 - containing all software published on The Amstrad User tapes from Issue 1 to Issue 12 inclusive $50.00
(Not suitable for the PCW 8256/8512)

The Amstrad Computer User (English)

Issue 3	January/February 1985	$4.00
Issue 4	March 1985	$4.00
Issue 5	April 1985	$4.00
Issue 6	May 1985	$4.00
Issue 7	June 1985	$4.00
Issue 8	July 1985	Out of Stock
Issue 9	August 1985	Out of Stock
Issue 10	September 1985	Out of Stock
Issue 11	October 1985	Out of Stock
Issue 12	November 1985	$4.50
Issue 13	December 1985	$4.50
Issue 14	January 1986	$4.50
Issue 15	February 1986	$4.50
Issue 16	March 1986	$4.50
Issue 17	April 1986	$4.50
Issue 18	May 1986	$4.50
Issue 19	June 1986	$4.50

Twelve issue subscription: magazine only	$47.50

ALL THE ABOVE PRICES INCLUDE POSTAGE

The 40cent Programs

by Ivor Joystick

Once upon a time, a sensible man bought an Amstrad computer.

Thirsting for information, he took out a subscription to The Amstrad User, the largest selling magazine for the Amstrad range in Australia.

He didn't need his machine to work out that, for less than $3* per month for 32 pages (and quite often more) of solid Amstrad information, he was getting programs for less than 40 cents each!

This made him live very happily ever after.

. . . but that's not the end of the story

Every month The Amstrad User is packed with a range of articles, programs, hardware and software reviews, utilities, User Group Information, Letters, Tutorials, Hints and Tips - all devoted entirely to the Amstrad computers.

And to save you the trouble of keying-in and correcting your typing errors, a cassette containing the programs appearing in the magazine each month is also available.

Make your Amstrad a success story by subscribing to The Amstrad User now.

* Based on yearly subscription of $35.

- -

Please send me THE AMSTRAD USER for 12 months

Magazine only: $35 □ Magazine and cassette: $75 □ (PNG and NZ add $12 airmail)

Payment by: Cheque □ Bankcard or Mastercard □

Card number_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _Expiry date.........

Name .... Phone......................

Address ....

.... Postcode ....

Signed .... **Please start with Issue No.**......

Return to THE AMSTRAD USER, Suite 1, 245 Springvale Road,
Glen Waverley Victoria, 3150 Telephone (03) 233 9661

(OVERSEAS PRICES ON APPLICATION TO ABOVE ADDRESS)

Do you have the drive to succeed in business?

Running any business successfully means first running it efficiently.

And no matter how businesslike you think you are, you can't beat the power and convenience of a random access CP/M* disc system.

Now, with Amstrad's CPC 464 computer, you can enjoy all the advantages of a 3″ disc drive complete with an integral power supply and plug-in interface controller.

Naturally, you can add a second drive (FD-1) to double the on-line storage capacity, speed up copying files and producing back-up discs.

Then, you have a choice of more than 200 disc-based business programmes — with many more titles on the way.

It all adds up to a better business, the Amstrad way.

AMSDOS and CP/M*

CP/M* is the standard disc operating system for 8-bit microcomputers.

We also supply a special version of DR. LOGO* for CPC 464 users.

Our disc drive system includes a number of extensions to BASIC in the AMSDOS operating system that allow the disc to use files originally created for cassette, and vice versa.

Software also takes care of the necessary file management so that CP/M* files can exist alongside these AMSDOS files.

Data files may even be shared bet-ween AMSDOS and CP/M*.

Word processing, data management, accounting, ledgers — you'll find an AMSOFT program to suit whatever kind of business you want to do on your CPC 464.

There's even a range of programmes that teach you professional business practice.

CPC 464. Really gets down to business.

At the heart of it all, of course, is the incredible Amstrad CPC 464 computer.

The CPC 464 has a typewriter-style keyboard, large ENTRY key, sensibly positioned cursor keys, numeric keypad for fast data entry and a full 8-bit character set.

It provides high resolution graphics, 80 column text display, up to 8 text windows plus a graphics window and a palette of 27 colours.

There's also a built-in Centronics standard 7-bit parallel printer interface. So you can enjoy high performance word processing with the printer of your choice.

AMSTRAD

*Registered Trade Mark of Digital Research.

TPB/383

Software for the PCW's

Simon Anthony presents his views on two recent releases of software for the PCW 8256 and 8512, namely THE CRACKER and the POLY series.

As with many computers when they first come onto the market, good software is generally a little thin on the ground. It always seems to take a while before the software developers take the plunge. The PCW range suffered a similar problem when the 8256 was first released but things appear to be on the improve now that the 8512 is available.

Two pieces of software arrived on my desk recently - The Cracker and a 'Poly' series. To be accurate, I should count it as six pieces of software as the 'Poly' series contained four programs on the one disc. Both packages are what should be termed "serious software".

THE CRACKER

This is suitable for the PCW8256/8512 and the CPC6123. I had heard that The Cracker was a little different from the normal spreadsheet so I tackled the manual first. It is reasonably well presented and written in a clear style falling into two major sections - a tutorial and a reference. There are heaps of screen diagrams and examples in the tutorial which help to keep a clear mind on understanding how it all works. Once you are up and running, the reference section scores well with its layout and content.

The rumours were right - it is a spreadsheet - but with a difference. Yes, it can do the normal budget type calculations you would expect from a normal spreadsheet but the difference is that it goes much further than that. It extends into maths, scientific and statistics applications. In fact the manual suggests using it to solve simultaneous equations and similar problems iteratively (*that means repeatedly - Ed*). But I'll come back to this later.

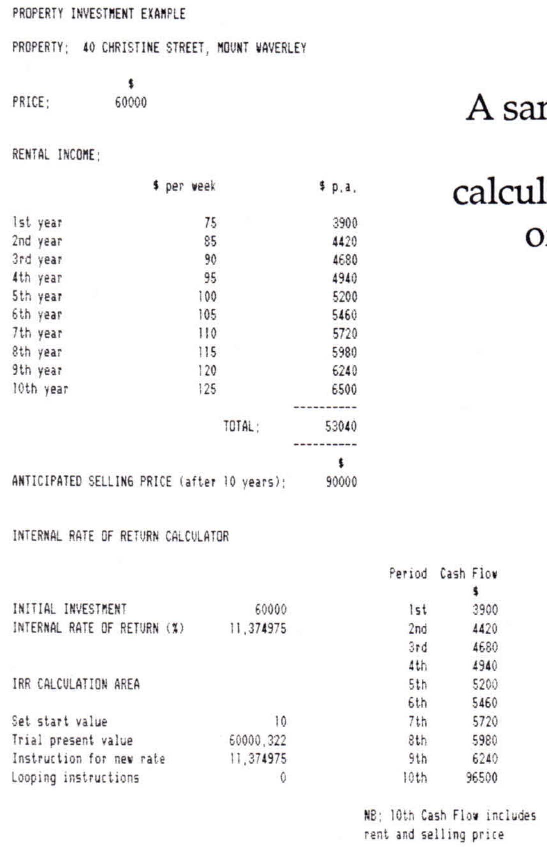
Unlike other spreadsheets when the program is first loaded, a sheet has to be created from scratch. When adding columns, you have to specify the width you want and the default data you want each cell to contain. Naturally, this gives a fair degree of control over the layout but takes a while if you are just using it for a simple budgeting exercise.

What I did find a useful feature was the response in plain English to entering single letter commands which are used to control and manipulate the sheet. For example, typing NF4E will produce a display command line of "New format to be 4 sig figs Exponent". The display appears as you enter the commands, so you have the opportunity to cancel before hitting return. This feature appears at every stage of the command process with all possible key depressions being displayed above the command line. It's also useful when entering formulae as errors are reported immediately - again before the return key has been hit.

Error messages are generally explicit and helpful. For example, I tried to enter "6/+4" but got a beep when I typed the plus sign with a message "Separate these operators".

The Cracker is streets ahead of any spreadsheet I have seen in terms of its originality. Whilst most spreadsheets can usually handle numeric operators, trig functions and statistics with some allowing IF...THEN construction, Cracker formulae can include pi, e, logs (natural or base 10), perms and combs, random numbers, Booleans, interpolations and almost anything else in the mathematics line and alter the contents of other cells quite freely. In addition, the DO....WHILE construct gives true program status.

The Cracker also has the ability to sort rows and columns either numerically or alphabetically.



```
PROPERTY INVESTMENT EXAMPLE

PROPERTY;  40 CHRISTINE STREET, MOUNT WAVERLEY

                 $
PRICE;         60000


RENTAL INCOME;

                       $ per week              $ p.a.

1st year                   75                   3900
2nd year                   85                   4420
3rd year                   90                   4680
4th year                   95                   4940
5th year                  100                   5200
6th year                  105                   5460
7th year                  110                   5720
8th year                  115                   5980
9th year                  120                   6240
10th year                 125                   6500
                                          ----------
                              TOTAL;           53040
                                          ----------
                                                   $
ANTICIPATED SELLING PRICE (after 10 years);    90000


INTERNAL RATE OF RETURN CALCULATOR

                                                    Period  Cash Flow
                                                               $
INITIAL INVESTMENT                  60000              1st      3900
INTERNAL RATE OF RETURN (%)     11,374975              2nd      4420
                                                       3rd      4680
                                                       4th      4940
IRR CALCULATION AREA                                   5th      5200
                                                       6th      5460
Set start value                        10              7th      5720
Trial present value             60000,322              8th      5980
Instruction for new rate        11,374975              9th      6240
Looping instructions                    0             10th     96500

                                                  NB; 10th Cash Flow includes
                                                  rent and selling price
```

A sample printout showing the calculation abilities of The Cracker

This opens up the area of adapting it for use as a small database.

Simultaneous equations can be done either by forced recalculation or by the use of DO...WHILE loops. I suppose one could argue a weakness in the limitation to plus or minus 1E38, but accuracy to 12 decimal places is quite adequate.

There is not a lot of memory left once The Cracker has been loaded (17k on both the PCW's and the 6128), but in reality there is plenty for its intended application. Most spreadsheets perform number crunching with vast amounts of data whereas The Cracker performs far more impressive things with smaller quantities.

The program is fast as it works entirely in memory, and when it comes to the more mundane functions, it can be left in a DO...WHILE loop while you nip out to the Milk Bar.

To summarise, The Cracker is clearly intended for the scientific, engineering and mathematical market. It is full of powerful expressions and on-screen help, but non-programmers may have trouble in getting the full use out of it.

The Cracker retails at $122.50.

THE POLY SERIES

This package, for the PCW's, consists of four pieces of software on one disc. Polyword, a word processor; Polymail, a mail merging database; Polyprint, a printing program with numerous typefaces; and Polyplot, a graph plotting program. I am not sure whether each program can be purchased separately, but my sample came in one packet with the disc and four small manuals.

POLYWORD seems to get over the problem faced by PCW owners in that normal Locoscript produced files cannot be used by any other programs outside of the system because they are not strictly ASCII files. Polyword does, however, and under CP/M. Of course, you can't expect Polyword to have as many features as Locoscript, but the designers have clearly had the experienced Locoscript user in mind as they have kept to the same keys on the numeric keypad.

I was pleased to see that, when the program was loaded, I was still able to access my constant companion PIP. I was not pleased when I discovered that it can only handle files of up to 200 lines with a maximum line length of 80 characters. That test went down the tubes, but in fairness, this limit wouldn't normally be reached in everyday use.

The 'Choose File' option creates a new file or edits an old one and takes you straight into Edit mode. The 32 lines on the screen are allocated to a single status line at the top of the screen and 31 lines for the text. Word wrapping is automatic. The designers have also had Wordstar in mind by providing certain functions and defaults found in that package and by allowing alternative key entries in Polyword that one would use in either Locoscript or Wordstar. Foreign characters are reasonably well catered for, with characters 127 to 186 being 'Polyprintable'.

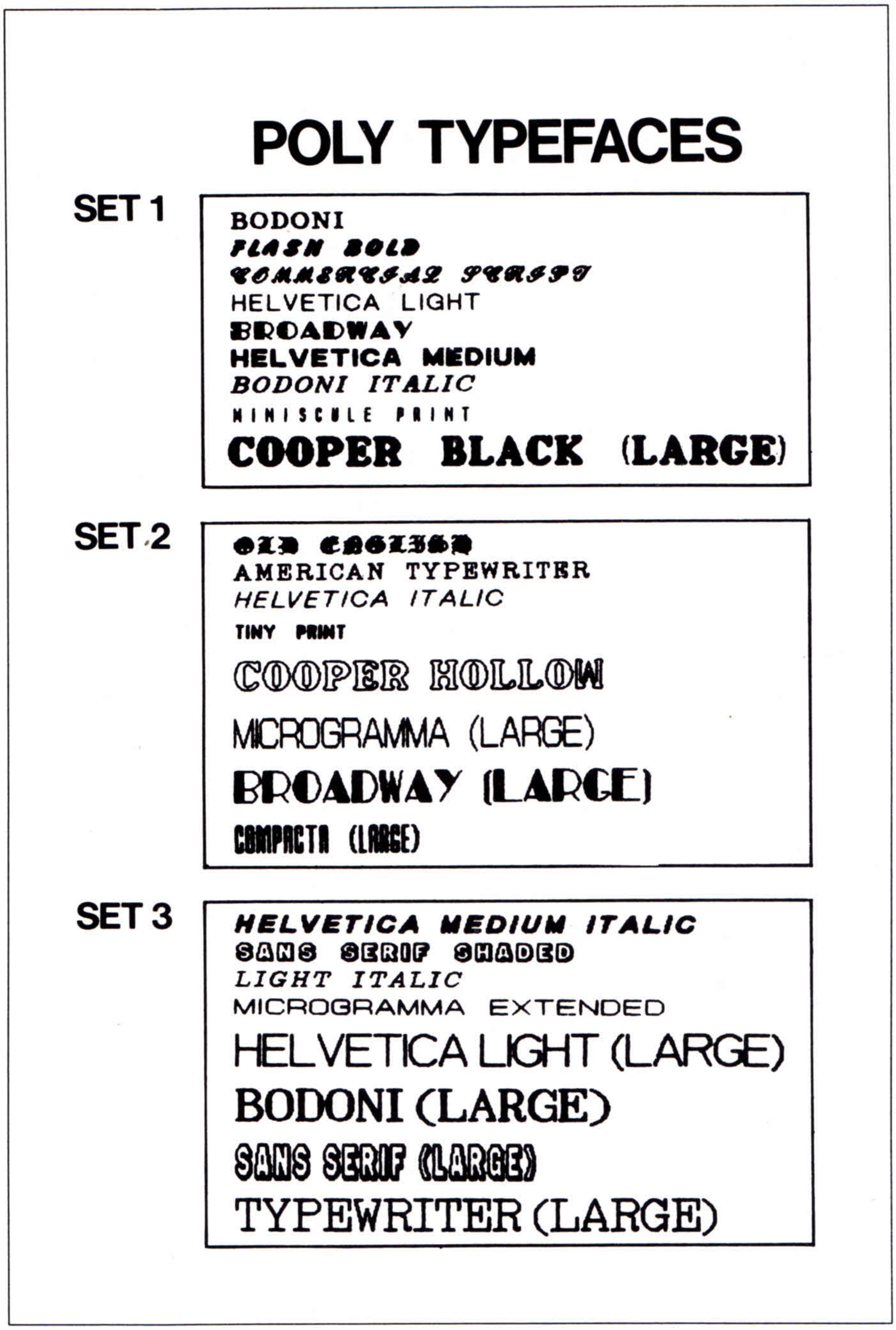
POLYMAIL is the mail-merging database module useful for preparing address labels and circulars. But if you think you can print them in the special Polyprint typefaces - forget it!

It operates in a very similar fashion to Polyword, but again there is a limitation, this time it's the number of records that a Polymail file can hold - just 100. For small businesses with few clients this is probably adequate. I didn't have reams of data to play with, but I guess that if one has more than 100 clients/debtors/creditors or whatever, they could be split alphabetically between a series of files - A to H; I to P; and Q to Z in three files.

Polymail is used to create and edit the records while Polyword takes care of the actual merging of the addresses to the circular, if that is what you are doing.

POLYPRINT lets you turn your basic PCW printer into a powerful tool capable of printing in some 25 typefaces. As the matrix of most of the Polyprint characters are 14 x 18 as opposed to 6 x 8 in draft format on the Amstrad, the quality of the print is much improved. The larger typefaces go to 20 x 32! To get a better idea of the range, take a look at the example.

In the same manner as the other Poly programs, it is Menu driven and easy to operate. Simple one letter commands following an asterisk embedded in your text perform the usual centring, underlining etc. functions. The only time the single letter command differs is when you need to change the typeface. Incidentally, (here comes another limitation), you can only change a complete line to another typeface, not just a single word. Nevertheless, it does produce attractive circulars and mini-posters.



POLY TYPEFACES

SET 1

BODONI
FLASH BOLD
COMMERCIAL SCRIPT
HELVETICA LIGHT
BROADWAY
HELVETICA MEDIUM
BODONI ITALIC
MINISCULE PRINT
COOPER BLACK (LARGE)

SET 2

OLD ENGLISH
AMERICAN TYPEWRITER
HELVETICA ITALIC
TINY PRINT
COOPER HOLLOW
MICROGRAMMA (LARGE)
BROADWAY (LARGE)
COMPACTA (LARGE)

SET 3

HELVETICA MEDIUM ITALIC
SANS SERIF SHADED
LIGHT ITALIC
MICROGRAMMA EXTENDED
HELVETICA LIGHT (LARGE)
BODONI (LARGE)
SANS SERIF (LARGE)
TYPEWRITER (LARGE)

POLYPLOT is used for drawing line graphs, bar graphs and pie charts. Again, it is menu driven with the facility to save the output for later editing. The good news is that you can use the Polyprint typefaces. The bad news is that Polyplot will not access data from other files. You have to key in all the data, you know, the number of sets, the number of points, the values and so on.

I found that it took quite a while to internally prepare the data and print - another chance for a trip down to the Milk Bar! I haven't quite worked out why, but guess that it is something to do with the volume of pixels with which it is dealing. The results are worth waiting for - I thought the output looked quite professional.

To summarise, taken individually, each piece has its advantages and disadvantages. Taken as a whole, the package must appeal to many in providing a low cost yet relatively good quality system. It is certainly not meant for the 'big boys'.

The Poly series retails at $179.50.

Simon Anthony

More PCW Software

WORD PROCESSING		
Newword		172.50
Wordstar	*	P.O.A
Microword/Microfile	*	105.00
Star Index	*	290.00
FINANCIAL MODELLING		
Supercalc 2	*	103.95
The Cracker	*	122.50
Planner Calc	*	97.50
Master Planner		175.00
UTILITIES		
Devpac	*	89.00
PROGRAMMING LANGUAGES		
HiSoft C	*	89.00
Nevada Fortran	*	89.00
Nevada Cobol	*	89.00
Nevada Pilot	*	89.00
Microprolog	*	105.00
Pascal/MT+	*	124.50
HiSoft Basic compiler	*	99.00
Mallard Basic	*	245.00
CBasic compiler	*	124.50
GRAPHICS		
Dr. Draw	*	124.50
Dr. Graph	*	124.50
Polyprint/Polyword/ Polyplot/Polymail		179.50
Polytype		120.00
DATABASE MANAGEMENT		
Cambase	*	97.50
dBase 2	*	910.00
Cardbox	*	295.95
Microword/Microfile	*	105.00
Sage Retrieve	*	295.95
COMMUNICATIONS		
Sage Chit Chat Combo Pack		295.95
ACCOUNTS		
Sage Popular Accounts	*	295.95
Sage Invoicing	*	239.50
Cash Book		88.50
Integrity Accounting 1		695.00
TRAINING		
Guide to Locoscript		24.50
Touch 'n' Go	*	62.50
Two fingers	*	65.50
Typing crash course	*	65.50
PROJECT PLANNING		
Brainstorm	*	125.00
Target Task		245.00
GAMES		
Classic Adventure plus Mordens Quest (1 disc)	*	63.00

We have been advised that items marked with an asterisk can also be run on a 6128.

Software Gossip from the UK

We have managed to establish a link with "an informed source" in the UK, who has promised to supply us with as much gossip and information he can lay his hands (or ears) on. Here is his first report.

Swords and Sorcery - PSS have completed the Amstrad conversion of this fantasy roleplaying adventure. Starting as a novice, skills are built in magic and swordplay as the game progresses.

Nexus - this all-in-one kung-fu arcade adventure is soon to be released for the Amstrads. The aim in this game is to smash an evil drug ring hidden in a secret underground complex.

The Very Big Cave Adventure - a loose adaptation of the original Colossal Caves should be available before the end of the year.

The Comet Game - an arcade strategy game with three phases is being developed for the Amstrads. The aim is to destroy the comet's germbags before they infect the Earth. As Halley disappears for another 76 years perhaps this game will be too late.

Colossal Trilogy - soon to be available for the PCW's on disc. It is essentially an enhanced version of Level 9's Colossal Adventure, Adventure Quest and Dungeon Quest.

Bounces - from Beyond Software has been released. It's a combat type game for two players which won a couple of awards last year when available on other machines. Catch it if you can!

Tasword 8000 - is being developed for the PCW's by Tasman. It is not a CP/M program but will use much of the PCW's large capacity and features. Will include a built-in mail-merge and use the full 90 x 32 screen area.

Delays - US Gold are having trouble keeping to their schedules. The original version of **Impossible Mission** was scrapped and is being rewritten some six months after it was advertised as being "out now". **Dambusters** has also experienced long delays. Infocom seem to be having problems with **Wishbringer** and **Sorceror**, not forgetting the **Hitchhikers Guide to the Galaxy** for the 464/664. Perhaps the latter it thumbed a lift with Halley!

Note: Release dates of software in the UK quite often change and availability in Australia obviously depends upon these and, of course, how quickly the importers can bring them into this country.