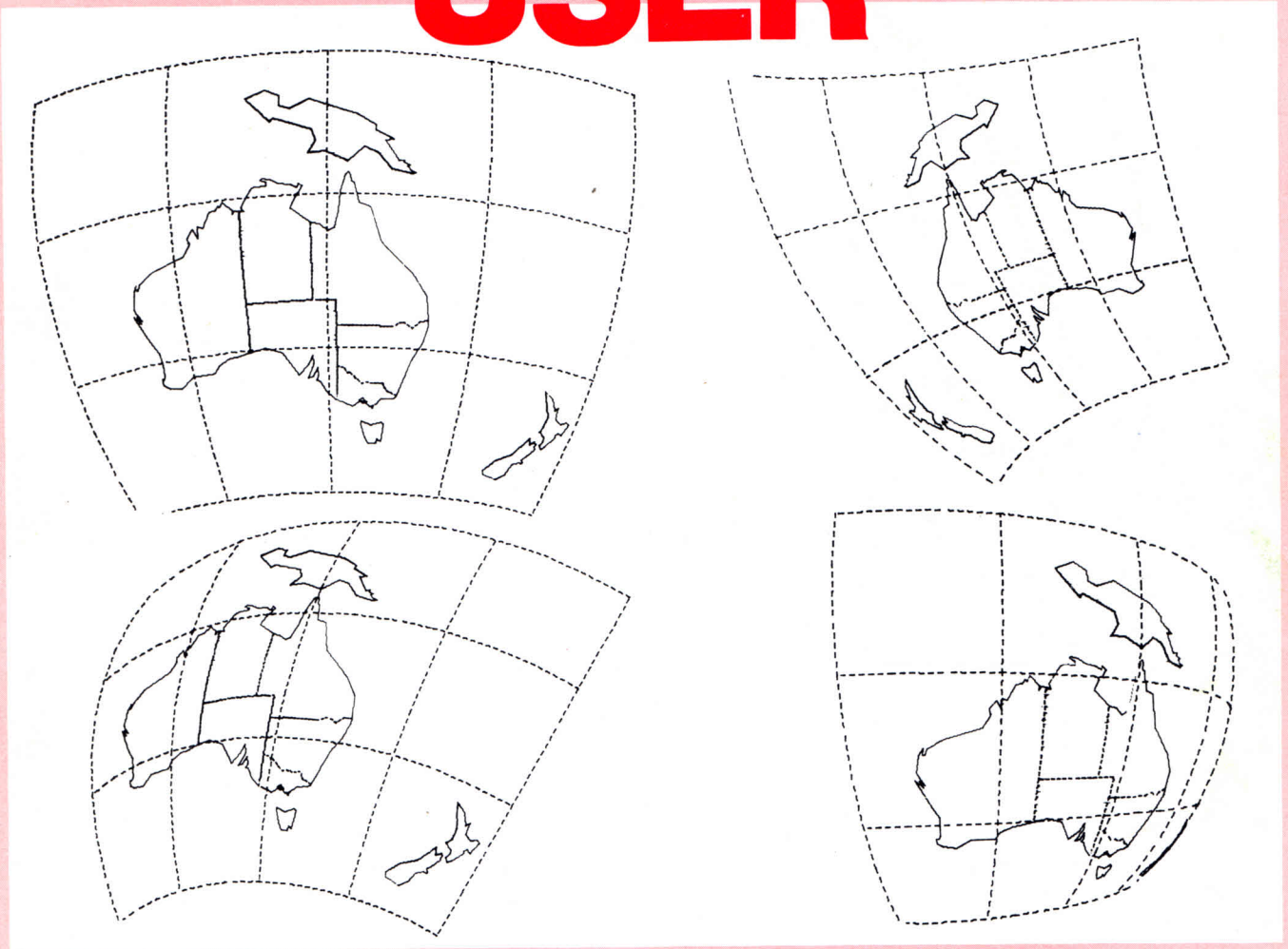


THE **AMSTRAD** **USER**

Issue no. 5

\$3.00

June 1985



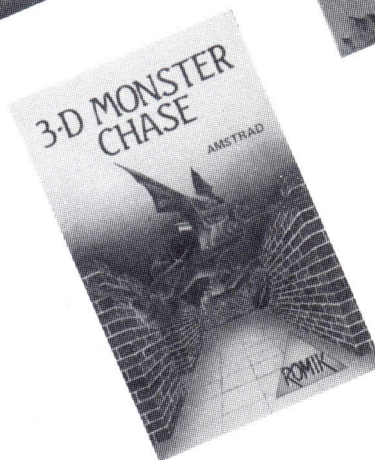
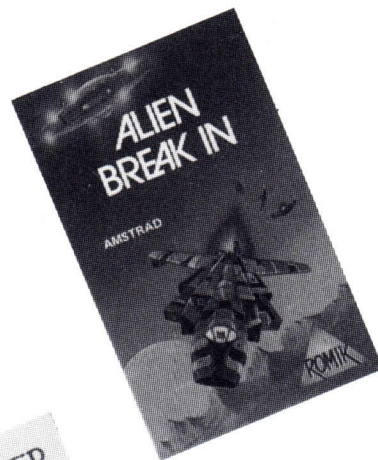
- **GRAPHICS – AUSTRALIA FROM SPACE**
- **SOUND EXPERIMENT SOFTWARE**
- **USER GROUP REPORTS AND CONTACTS**
- **LETTERS PLUS \$2500 COMPETITION**

FOR THE NOVICE & EXPERIENCED USER

NEW

AMSTRAD ARCADE ALLEY

**Five Exciting Quality Games from Strategy
at only \$19.95**



OFFER EXTENDED TO END OF JUNE 1985

Any one - \$19.95 Any three - \$39.95

All five - \$59.95

(Prices include postage and packing)

Name

Address

..... Postcode

Creditcard

or cheque for \$

(Tick which required)

- 'ER-BERT
- DRAGONS GOLD
- ALIEN BREAK-IN
- ATOM SMASHER
- 3D MONSTER CHASE

Enquiries to: Strategy Software, Shop 2, 33 The Centreway,
Blackburn Road, Mt. Waverley, Victoria 3149 Tel: (03) 233 9227

THE AMSTRAD USER

Issue No. 5 June 1985

CONTENTS

Editorial	2
Letters	3
User Group Information	6
A Graphic view of Australia from Len Heitman	9
The Learning Centre plus utility program	14
Review of FORTH by Shane Kelly	20
The Trials of Tony Blakemore	21
Sounds Good — an experiment from P. Campbell	22
Review of Ghostbusters by Simon Anthony	30
A Plug for the Wireless Institute	31
Another chance at the Competition	32

All enquiries and contacts should be made to The Amstrad User, Suite 4a, 33-45 The Centreway, Blackburn Road, Mt. Waverley 3149, Australia. Tel: (03) 233 9227.

The Amstrad User is published each month by Strategy Publications. Reprinting of articles in The Amstrad User is strictly forbidden without written permission. All rights reserved. Copyright 1985 by Strategy Publications. The single copy price of \$3.00 is the recommended retail price only. Also

available on subscription for 12 months.

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 welcomed from readers or other interested parties. If you want them returned, then please send a large SAE with all submissions.

THE AMSTRAD USER

G'day

Yes - the rumours are true! I did pay nearly \$300 to contributors last month. This will probably explain why we have had so many contributions this month. Not that I am complaining - on the contrary - I doubt that I shall ever be happy until I have too much to handle, which may sound a little convoluted, but in reality, would allow me to produce a magazine with a greater degree of forward planning. So the message is: Keep 'em rollin' in (and maybe earn some dollars at the same time).

You will remember that in March this year, we attended the APC Show in Sydney. A similar Show will be held in Melbourne at the World Trade Centre from 17-20 July. On this occasion, The Amstrad User will have its' own stand, and hopefully a camera that works to record events. (Unfortunately we lost the pictures from Sydney due to an operator error. This time we will check that the lens cover has been removed first).

I don't often make acknowledgements in the editorial, but it would be remiss of me not to congratulate Len Heitman for his mammoth effort over many months, in developing the software to produce the amazing graphic effects you can see on the front cover, and that you can create yourself from the program listing and instructions printed in this edition.

Following my request last month for emerging User Groups to keep me informed of their developments, it was certainly a surprise to hear that a group at Innaloo in Western Australia, calling themselves AMSWEST Amstrad User Group, had formed and after just two meetings had an attendance of almost 80 people. A full report can be found in the User Group Information section.

Finally, a reminder that we are looking for people to form a panel of software reviewers who, for their trouble, will be able to keep the article of software they are reviewing. We have had a few responses so far, and if you feel that you could be eligible to join the panel, drop us a line giving the reasons why you should be considered. It won't be a large panel, so it would be wise to move quickly.

See you next month,

Ed.

Letters

A selection of letters received this month for which the authors will earn \$5. We remind you that if you require a personal reply, please send a stamped and self-addressed envelope.

In chapter 8 page 33, of the Users Instruction Manual the Peek Command contains a printing error. I have corrected this error and added other lines so that the Peek command will show blocks of 8 and these will be in hexadecimal.

I enclose a listing for the above changes. Hoping these will be of some use to other Amstrad users.

```
10 ZONE 10
20 CLS
30 MODE 2
40 BORDER 26:INK 0,20:INK 1,0
50 LOCATE 10,12:PRINT"Put prefix '&' before both addresses to show hex values":PRINT:PRINT
60 INPUT "start address";first
70 INPUT "end address";last
80 FOR n= first TO last
90 value$=HEX$(PEEK(n),2)
100 PRINT value$;
110 PRINT"->";HEX$(n),
120 NEXT
```

Edward Plunkett, Eugowra, NSW

I have flown racing pigeons for the past 30 years and I would be very interested in hearing from another pigeon flying Amstrad user who knows of a program to work out race velocities on the Amstrad.

I would like to be able to enter the loft number or the flyer's name, his clock in time and the clock's loss or gain. After all times and clock variations have been entered I would like to run the results i.e. Flyer's position and velocity, his flying time and for a check only his clock in time with it's loss or gain. Would this be possible without a disk drive?

I would also like any Amstrad User living in Pt. Pirie to ring me on (086) 32 3919 so a club may be formed or information exchanged.

Congratulations on the magazine but I have one suggestion. You have two order forms in this month's issue. One is back to back with the Animal Jumble program and the other one on the back cover which would, upon being cut out, ruin the copy of the magazine. Could a tear out order form be included?

Keith Partridge, Pt. Pirie, SA

We could provide tear-out forms, but extras of this kind inevitably add to production costs. Normally, a photo-copy of an order form is quite acceptable if you need to keep your magazine intact.

This letter is in relation to those people who have purchased a computer without any previous knowledge of their workings.

The first thing I did was to get it all set up and working so as I could justify to the wife the reason for purchasing a computer. This is where my first of many problems started, I didn't even understand the word PRINT. Six months later and many hours into the mornings (chasing those things the wife hates (BUGS)) I've managed to get my programming skills to what they are today.

One evening in frustration I nearly sent the lot out the window, but decided help would be cheaper. That's where the local user group in Frankston came in, they were more than helpful and most importantly friendly. So I would urge anyone with similar experienced to seek out such groups (It's better than losing one's hair or go grey). Don't do what I did for six months, and that's try and work

it out by one's self to the point of frustration. Now that I've joined the local group the wife might be a little more well happier.

Steve Issell, Carrum Downs, Vic

I found the "Kingdoms" game to be quite good except that it did not appear to do as it was supposed to. I found that the acreage available expanded at an exponential rate whether I cut back on plantings of corn or not. Also, there appeared to be no limit on population growth as mentioned in the instructions.

I modified the program to cut back on acreage when plantings were reduced, and to limit the population to three times the initial one. For your information, the changes were:

```
85 mx=0 : REM sets initial value to zero
310 GOTO 85
420 e=0:b1=b
990 IF aa/10>a*0.6 THEN a=INT(a+aa/20) ELSE a=INT(a-aa/20)
995 IF b>3*b1 THEN b=b1*3:mx=b1:GOTO 1010
1000 IF b>mx THEN mx=b
1005 b=bb/6
1010 IF b<0.7*mx THEN af=1 ELSE af=0 DELETE 1020
```

R. Chapman, Glenbrook, NSW

I congratulate you on the presentation and layout of your magazine. It seems that from a small and perhaps shaky start you have developed into a very informative and professionally styled user magazine. There is never a night, when programming, that either one or more issues are open for reference.

I have a grumble though. At those enthusiasts who submit their programs for publication --- Try this one-liner before the final save or print out.

```
10 LOAD "PROGRAM NAME":  
RENUM:LIST #8
```

This will save many hours and the hair on those of us who are devout 'keyboard bashers' especially in the early hours of the morning. There is nothing worse than finding that The Amstrad User has published a reasonable program but having to type in the line numbers. After all BASIC is complicated enough for us without having to worry about line numbers, or is Amstrad hiding dark secrets about the Basic commands 'AUTO' and 'RENUM'.

Please don't get me wrong, I can appreciate that in program development one may have to insert extra lines or delete others, but why make it hard for type-setters and us Keyboarders?

And to all you EXPERIENCED PROGRAMMERS, how about some contributions. Need some ideas?

1. Redefining the cursor to any of the character set, and making it flash. This will make it much easier to spot on a screen full of

text in mode 2.

2. A keyword entry routine obtained by e.g. shift/A for AUTO to shift/Z for LEFT\$, etc. with a key toggle to switch on/off allowing normal keyboard use without complicated redefining keys (Machine Code?).

Bob Harwood, Cooloongup, WA

Perhaps somebody will come up with the answer.

I am running Tasword and have a Silver Reed EXP 500 daisy wheel printer. I am having no success in getting the printer to underline. It is compatible with Diablo 1610 and Tasword is Epson based. I am sure I am not the only person who is having this problem and would be happy to hear if anybody has solved it.

Sydney Jackson, St. Kilda, Vic

I purchased a colour monitor with the Amstrad. Later, as the need arises I would like to use a green screen for word processing etc. Is it possible to purchase the Amstrad Green Screen Monitor separately?

If not, is it possible to use any brand of monitor through the modulator, and would there be any modifications necessary in the circuiting?

Michael Toussaint, Loganlea, QLD

At the moment it is not possible to purchase a green screen separately. AWA advise that if there is sufficient demand they would re-consider. It is possible to use most modern monitors along with the modulator without modifications.

The Fifth Australian Personal Computer Show

'PC 85', to be staged at the World Trade Centre in Melbourne from 17-20 July 1985, is apparently sold out in terms of exhibitor space.

While all types of computer shows have proliferated throughout Australia in recent years, the PC shows, as well as attracting strong exhibitor support, have also been a magnet for more than 100,000 visitors. It is an opportunity to see and compare, which creates so much interest in a show where the most advanced technology is on display.

Naturally, AWA - Thorn will be exhibiting the Amstrad range of computer equipment and staff will be on hand to answer questions from the thousands of visitors expected.

The Amstrad User will also have a stand, and will be happy to meet subscribers, contributors or anyone who has an interest in the magazine.

For people unfamiliar with personal computers, a feature of the show is Computermat which offers instructions on how various equipment operates from working models installed by the major exhibitors.

JONES! YOU WONT BELIEVE WHAT
THE COMPUTER MAKES OF FORMULA X2



Amstrad family computer with monitor, from around \$500

No other system adds up.



For around the same price as a basic computer, Amstrad gives you a complete, ready-to-go system which includes a high definition green screen monitor. Plus you get a full 12 month warranty on the computer, monitor and keyboard.

Some of the outstanding

features of the Amstrad system include:

- 64K RAM, 27 colours, 8 text windows.
- Green screen monitor included, optional colour monitor available.
- 20, 40 and 80 column modes.
- Built-in data recorder.
- QWERTY typewriter keyboard.
- Separate numeric keypad.
- 32 user defined keys.
- Full edit facility including copy cursor.
- Stereo sound — 3 channels, 7 octaves.
- 4 real time clocks.
- Fast and extended BASIC.

The optional Amstrad Starter Kit is also available, with a

selection of programmes and accessories to get you into computing fast. Then, there's a wide range of programmes for business, education and family entertainment. Some of the



many titles available are featured here — your Amstrad dealer can give you details of the full range.

Compare the Amstrad CPC 464 colour personal computer with any other comparably priced system and you'll agree — no other system adds up to Amstrad. Ask for a demonstration today, at leading computer stores and retailers throughout Australia.

For the location of your nearest Amstrad dealer, please telephone the head office of AWA-Thorn in your state.

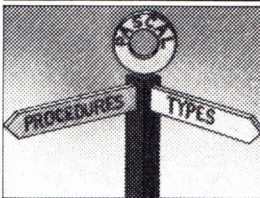
AMSTRAD

Distributed and guaranteed throughout Australia by

AWA-THORN

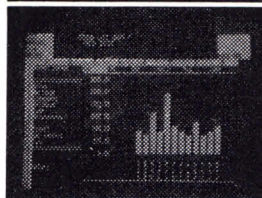
TPI 501

PROGRAMMING



From the most widely used computer language, BASIC, through to more advanced languages like PASCAL, Hi-Soft PASCAL and FORTH, Amstrad offers a range of easily understood programmes.

BUSINESS



Amstrad programmes are available for many business applications, including Spreadsheet, Word Processing, Data Base, Sales Invoicing and Stock Control — all designed to help your business.

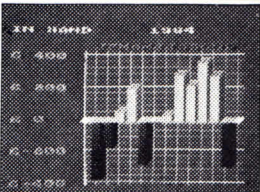
BRAIN POWER

Product name	Disc's	SR	RP
1. Project Planner	1	55	25
2. Decision Maker	1	45	24
3. Entrepreneur	1	45	24
4. Star Watcher	2	65	28
5. Project Planner	1	55	25
6. Decision Maker	1	45	24
7. Entrepreneur	1	45	24
8. Star Watcher	2	65	28
9. Project Planner	1	55	25
10. Decision Maker	1	45	24

Press any key to exit track

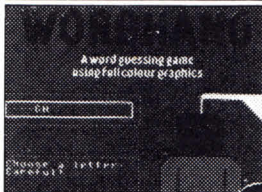
Learn the art of time, money and business management with programmes like Project Planner, Decision Maker, Entrepreneur and Star Watcher — they're all available to suit your Amstrad computer.

ELECTRONIC HOME



The Amstrad range of home programmes includes Home Budget, Diary, Typing Tutor and Bank Manager. Use them to plan your cashflow and manage your household affairs efficiently.

EDUCATION



Interest and involvement are the keys to this popular family of programmes covering Numbers, Telling the Time, Maths, Spelling, Geography, and more. They're all fun, all valuable learning aids.

GAMES



Amstrad offers some of the most exciting games ever devised for the home computer, including favourites like Ghostbusters, The Hobbit, Harrier Attack, Master Chess, Amsgolf and many, more.

User Group Information

Western Australia features pretty strongly this month with news on the formation of the AMSWEST User Group and the near-formation of the Rockingham-Kwinana Users Club. We also have news from Victoria, South Australia and Canberra.

AMSWEST, PERTH

I have been asked by the President of AMSWEST, the Amstrad User Group in Perth, to write and inform you of the fact that we have started a Group, which at the moment meets at Innaloo, W.A.

This group covers a wide range of age groups, from 13 years old to 70 years old.

I am enclosing a copy of a leaflet passed out to members at our meeting, as this will give you an idea of what the Group intends doing. At a later date we will probably send you some more information about what we are actually achieving. At the moment we have only just elected a Committee to start things rolling, and we hope to get our first educational programmes off the ground at the next meeting. Our membership, after only two meetings, is approximately 80.

The President is Mr. Tony Clitheroe of 17 Ash Way, Morley, (09) 275 1257, and I can be contacted at 6 Weston Street, Carlisle, (09) 361 8975 or during business hours on (09) 386 2282.

Mrs. P.T. Ardron (Secretary)

ROCKINGHAM-KWINANA USERS CLUB

I have today forwarded a letter to those people who have indicated interest in a Users' Club, and I am awaiting their reply with bated breath. We have problems in that half the prospective members are shift workers, so it could be difficult in meeting everyone at the same time. We hope that we will be able to iron this out, and I will keep you advised of developments.

Come on all you Amstrad 'gamers'. How about hanging up your joysticks and find out what the CPC464 can really do. See the user group page for details and contact me for meeting times.

Contact Bob Harwood on (095) 27 1777

AMSWEST

AMSTRAD USER GROUP

Welcome to AMSWEST the Amstrad User Group of Perth metro area.

As a member of the user group, meeting other users with the same hopes and problems and being able to discuss these will make your computing easier and more fun.

It is hoped that with your membership and support the user group will be able to offer you many services and facilities some of these are:-

EDUCATION:- It is the intention of the group to provide courses in programming in the following languages BASIC, PASCAL, Z80 machine code, CP/M. Also basic Business applications.

LIBRARY:- The library will be structured for input from the members and may offer public domain program listings or cassettes. Books or magazines on the Amstrad and computing in general.

MAGAZINE:- The magazine will provide helpful hints, tips, reviews on hardware and software, listings and general interest articles.

TECHNICAL ASSISTANCE:- Technical assistance in the development of add on peripherals maybe provided by technicians in the club.

DISPLAYS:- With the support of AWA and many computer houses displays and discussions of new products and software.

We trust some of these things are what you are looking for to make computing easy and pleasant and will encourage your membership.

HAPPY COMPUTING

WESTERN AMSTRAD USER CLUB

Tentative arrangements have been made to use a convenient meeting place in Braybrook (Victoria) for the establishment of the Western Amstrad User Club. In order to ensure that we don't miss out, I need to confirm a booking by 15th June.

Anyone interested in joining the club should contact me on 312 5594.

Mike McQueen

THE CAUS – RICHMOND (VIC)

The Central Amstrad User Society has been formed with its base and meeting place in the Hall at the corner of Church and Somerset Sts, Richmond.

A Management Committee was elected from the Members present and a constitution agreed upon. The President is Rimon Russo who can be contacted on 428 4281. At present the meetings are being held fortnightly due to the intense interest and rapidly growing membership. The meetings are held on a Sunday afternoon and commence at 1 p.m. There is a short Business Session followed by a Predetermined Topic and concludes with a friendly group session on the Amstrads. There are usually at least six Amstrad computers, Disc Drives and Printers in use at each meeting.

The meetings are a good chance for both the Novice and Experienced User to gain further knowledge on all aspects of the Amstrad, be it Software, Programming, Game Playing, Business Applications, etc.

Don Leith
Publicity Officer C.A.U.S.

AMSTRAD SOUTHERN USER GROUP

The Amstrad Southern User Group (Victoria) was formally established on 16th April 1985. Meetings take place at the John Paul College, Senior Campus, Frankston from 7.30 p.m. to 10.30 p.m. on the third Tuesday of each month (except December).

The aim of the group is to promote the advancement of software usage and the development of hardware add-ons. An annual subscription of \$30 has been fixed.

Please feel free to bring your computer with you.

Mike Prezons

AMSTRAD EASTERN USER GROUP

On the first Sunday in every month, the Amstrad Eastern User Group meets between 1 p.m. and 5 p.m. Approximately 25 people attended the first meeting held at Tony Blakemore's residence.

In addition to the monthly meetings, special 'beginners' meetings will be held once a fortnight. For details of these and the main meetings, contact Tony on 878 6212.

ACT AMSTRAD USERS GROUP

This group meets on the first Wednesday of each month at the Research School of Physical Sciences at the Australian National University. To be precise, in the Seminar Room of the Oliphant Building.

About 40 people attended the first meeting at which interim officers were elected until a formal constitution has been drawn up. All interested persons are very welcome to attend the meetings. Further details can be obtained from Kevin Loughrey on (062) 312 991.

The Micro-Processor Special interest group will be holding an exhibition at the Canberra TAFE College, Reid on Saturday 6th July from 10a.m. to 5p.m. The ACT Amstrad Users Group will have a stand at this show, and members will be available to answer questions concerning the CPC464 and Software.

The SA Report

As expected Amstrad interest is growing in S.A. which is being reflected in our club. A need is obvious for further clubs to be formed in the northern and southern suburbs. We keep a register of Amstrad users which we would be happy to make available to anybody else interested in starting a club.

Our club still meets weekly at the Grange Primary School between 6.30 – 9.00 p.m. each Tuesday.

We have found by conducting a raffle advertising our existence on each ticket is proving worthwhile. Thanks to all the S.A. retailers and club members who have helped us on our way by selling our raffle tickets. First prize is an Amstrad computer with green screen or a Disk Drive or \$400 cash. Tickets are \$1 each.

One obvious advantage of a club is the exchange of information. In this regard most retailers and of course AWA-Thorn have been most helpful. A.P.I. (Australian Post-Tel Institute) are conducting on-going courses on the Amstrad starting in June. I believe classes are filling fast. If anybody requires further information ring Mike Denieuwe on (08) 225 5995.

Maurice Van der Hoeven has taken up the pen and is bringing out a S.A. User group magazine which will work towards bringing owners and potential owners together. For more information contact Maurice on (08) 258 1722 after hours.

Now for the bad news. A good test for any club is its survival when members move on. Unfortunately I have been transferred to Perth in my job so I will be leaving Adelaide and the club in mid June.

We are hoping one of our regular members will be able to continue the running of the club at the Grange Primary School but this matter still has to be confirmed in the coming weeks.

As for myself I will be sorry to go but I am looking forward to meeting fellow Amstrad owners in W.A. and would like to join or form a club once I get settled in the West.

Don Buchanan.

Good luck with your transfer to Perth, and many thanks Don for getting the ball rolling in Grange – Ed.

Please note that any reports or information to be included in the User Group pages must reach THE AMSTRAD USER no later than the end of the first week of the month prior to publication.

User Group Contact List

NSW

John Patterson	Lismore	(066) 21 3345
Chris Craven	Canowindra	(063) 44 1150
Paul Wilson	Moruya	(044) 74 3160
Frank Humphreys	Mummulgum	(066) 64 7920
R. Vijayenthiran	Newtown	(02) 519 4106
Hans Hill	Blacktown	(02) 671 2929
Martin Clift	Narrabri	(067) 92 3077
Mrs. D. Sparks	East Gosford	(043) 24 3342
Jim Owen	Urunga	(066) 55 6190

ACT

John Payne	Phillip	(062) 82 2277
Arthur McGuffin	Kambah	(062) 31 9437
C.R. Rogers	Fraser	(062) 58 5749

Vic

R.A. Russo	Richmond	(03) 428 4281
Tony Blakemore	Nunawading	(03) 878 6212
Martin Scragg	Pearcedale	(059) 78 6949
Don Leith	Brunswick	(03) 383 1498
Michael Prezems	Frankston	(03) 781 2158
Mrs. G. Chapman	South Clayton	(03) 551 4897
Mike McQueen	Braybrook	(03) 312 5594
Paul Walker	Heathmont	(03) 729 8657
David Carbone	Burwood	(03) 29 4135
Sue Kelly	Manangatang	(050) 35 1402
Alan Harris	Sale	(051) 44 1454

QLD

Paul Witsen	Bulimba	(07) 371 9259
Mick O'Regan	Gladstone	(079) 79 2548
R.C. Watterton	Toowoomba	(076) 35 4305
D.F. Read	Ingham	(077) 77 8576
Kylie Telford	Goondiwindi	Calingunee 246 (weekends only)
Michael Toussaint	Loganlea	(07) 200 5414

SA

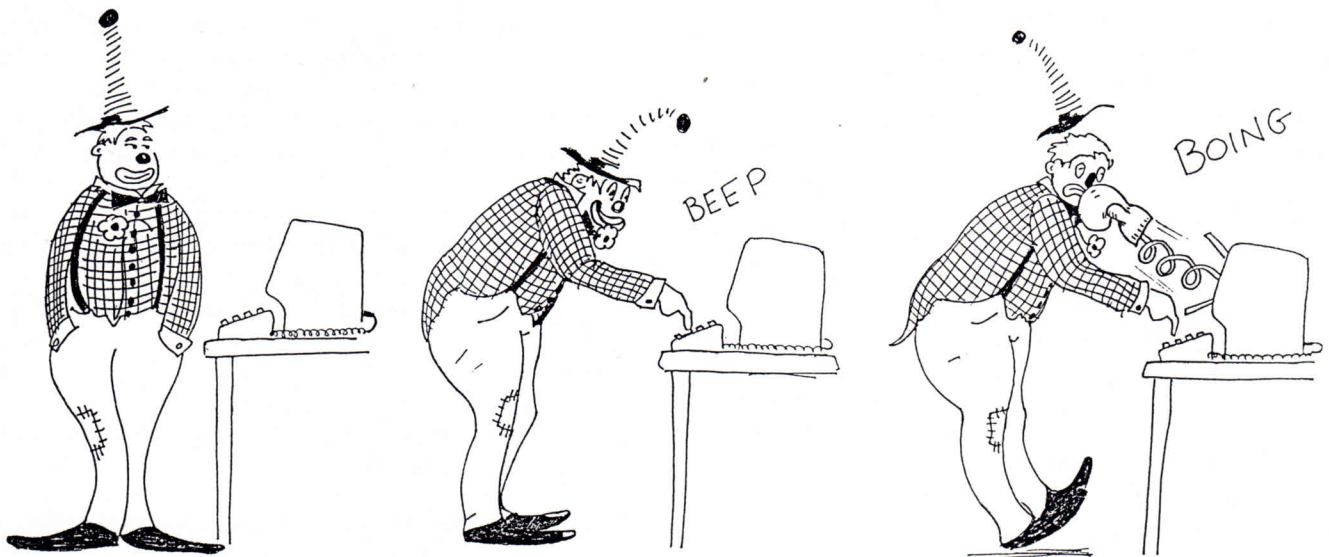
Don Buchanan	Grange	(08) 356 2664
Lindsay Allen	Murray Bridge	(085) 32 2340
Rick Cable	Pt. Pirie	(086) 5967

WA

Bob Harwood	Cooloongup	(095) 27 1777
Dave Andersen	6 Kitchener Rd Merredin, 6415	
Graeme Worth	Scarborough	(09) 341 5211
Warren Pynt	Dianella	(09) 276 9883
Tony Clitheroe	Morley	(09) 275 1257
Mrs. P. Ardron	Carlisle	(09) 361 8975

TAS

Conal McClure	Scottsdale	(003) 52 2514
---------------	------------	---------------



A Graphic View of Australia

Each night on television you see a weather map of Australia taken from a satellite in a geo-stationary orbit over the equator at a height of 35814 kms. MAP DRAW and MAP CODE programs written by Len Heitman allow you to draw your own map from any point in space or even from the centre of the earth.

The combination of MAP DRAW and MAP CODE programs will draw a map in perspective, fill it with colour, draw lines of longitude and latitude and present a picture of a globe being viewed from space, or from within the globe, and at various angles.

First a map outline is drawn using the longitude and latitude data points. Then using a fast machine code routine, the coastal outline is filled. As this fill routine increases the program running time, there is an option to cut it out by entering 1 at the NOFill input stage. The graph window has two spaces on the left — the top holds the map centre and camera angles used — the bottom records the initial input. Tests are performed including a check on the mode number which is used to control the ink colour for mode 2 and to stop the #1 window from printing in mode 0.

The program should be entered in two parts, first MAP CODE followed by MAP DRAW. Once saved, MAP CODE should be loaded which will automatically call MAP DRAW — type RUN.

You will be asked for the longitude and latitude at the centre of the map. This allows you to move the map up, down and sideways — like moving a camera to centre the picture.

The radius is preset at 100. If you wish to use any other radius, take out the word REM in line 70 and input your choice when asked. Radius and Size control how big the map is to be. It is all relative — a greater radius needs less size input. With Radius=100 and Size=6 all will be shown, but experiment with different sizes.

The camera angles and distance are input next. Zero azimuth and elevation is directly over the map centre. Think of a globe and you are directly above the map centre. If you move your head further to the east while still looking at the centre, say, at a 30 deg. angle, that equals 30 azm. If you look from the west side, that equals -30 azm. The same with elevation — if you move your head further north and look back, that equals 30 deg. Looking from the south of the centre would be -30 deg. Again, experiment with different angles (+ and -). Remember that these are zenith angles from the map centre not magnetic or geographic azimuth angles.

Now let's look at camera distance. 100 is equal to the earth radius. If you wish to use a true relative height then the distance equals HEIGHTkms/6378*100). At the equator a geostationary satellite is 562 distance units. Distance controls perspective, so going close does not alter the size only the perspective from the centre. Under about 20, with a large area, the centre enlarges and the sides shrink in and reverse below the horizon. Again, think of a globe with your eye one inch above it — looking sideways the sides curve below the edge. With a transparent globe they would curve under towards the centre, the lines of longitude and latitude show this clearly.

Remember, if you choose size 40 to take a close look at, say, Victoria, a low height such as 5 to 10 from the side will distort but 600 is like the weather map. Try map centre Long:145, Lat:-38, Camera azm:20, elv:44, dist:590, size:40. Don't try to fill it as it will take a while before any lines are drawn. This is because the program is trying to draw the rest of the map.

If you wish to draw only New Zealand, put the centre at about 173/-39, azm:0, elv:0, dist:100, size:12.

Distances greater than 999 can be used but will not show completely in the reserved space at the top left of the screen. The perspective does not change much over 500. In respect of the size — too big and you lose the edges, too small and the pixels may touch making the fill jump to the next item or filling outside the coast.

At NOFill, merely press enter to fill the map with colour, or enter 1 to inhibit the fill. Now sit back and watch in wonder!

To get input values from a particular place in space, a GOTO 900 will give a routine to get the camera values. Note that the radius of the satellite is from the earth centre (earth radius=6378kms), and for geo-stationary orbit the radius equals 42192kms (that is 662 relative to 100).

Now let's take a closer look at the coding and isolate some of the areas with which you may wish to experiment.

GRID LINES: referring to line 2340 as an example and in particular the first three data values — 3002,4105185,5660020 — which can be represented by

the following format — T0CC,Suuuyyy,Vwwwxxx where

T = Type = 3000
 CC = Pen number
 S = No. of long. lines
 uuu = Left side long. value
 yyy = Right side long. value
 V = No. of lat. lines
 www = Bottom lat. value
 xxx = Top lat. value

Note that if Lon. is west of Greenwich then Lon.=abs(lon)+600. Similarly, if lat. is south latitude then lat.=abs(lat)+600.

LONG. AND LAT. POSITION

Take a look at the first few data values in line 2010 (Australian coast). You will find 1001,140.97,-38.05. These values can be represented by T0CC,HLLL,Kill where

T = 1000 = Move pen to
 2000 = Draw pen to
 CC = Pen number
 H = + or - height for mountain or sea depth contours (optional)
 LLL = 0 to read the next 2 nos.
 1 to read the next 3 nos.
 2 to go to END routine
 Ill = Latitude

Note that the program will always read three DATA values first. The value in K will determine whether the

next 2 or 3 values are read or a GOTO END routine should be executed.

FILL

This routine starts from given longitude and latitude points in the middle of an island area. Then it fills the hollow, recording any missed areas to be filled later. It is better not to fill if a large azimuth angle or a small size is used.

For an example, refer to line 2290 — 4003,142.3,-7.8 — which has the format of T0CC,Long.,Lat., where

T = Type = 4000
 CC = Pen No.
 Long = Longitude
 Lat = Latitude

If filling starts in the wrong place, press Escape and Enter to undo the wrong part and jump to the next.

WORD OF WARNING

For people without a tape subscription, be very careful to get the number on the end of each data line correct when entering MAP-CODE. This is a checksum and helps to display the number of the line that has wrong machine code bytes in it.

SAMPLE ANGLES

Just to whet your appetite, try these entries:

	Lon.	Lat.	azm.	elv.	dist	size
1.	143	-24	0	0	100	5
2.	152	-32	32	-15	160	6
3.	152	-16	12	18	568	4.2
4.	142	-24	193	30	169	3.5

```

10 REM MAP-CODE: CODE to FILL MAP.26/4/85
20 CLS:PRINT" MAP-CODE: Loading machine code":PRINT:PRINT
   Leave PLAY key down on tape":PRINT
30 aa=0:DEF FN hex(h$)=VAL("&"+LEFT$(h$,2)):start=39000:addr=s
  tart:ln=130:MEMORY start-1
40 total=0
50 READ h$:h$=UPPER$(h$):aa=aa+1:PRINT aa,:IF h$="END" THEN 14
  0
60 IF ASC(h$)<>ASC("@") THEN 100
70 h$=RIGHT$(h$,LEN(h$)-1):Lb=FN hex(h$):total=total+Lb:h$=RIG
  HT$(h$,LEN(h$)-2):IF h$="" THEN 130
80 hb=FN hex(h$):total=total+hb:h$=RIGHT$(h$,LEN(h$)-2):w=256*
  hb+Lb-30000+start
90 POKE addr,w-256*INT(w/256):addr=addr+1:POKE addr,INT(w/256
  ):GOTO 110
100 b=FN hex(h$):total=total+b:h$=RIGHT$(h$,LEN(h$)-2):POKE add
  r,b
110 addr=addr+1:IF h$<>" THEN 60
120 READ check:PRINT check,total:IF check=total THEN ln=ln+10:G
  OTO 40
130 PRINT"error in line",ln:END
140 CLS:PRINT,TAB(10)"MAP-CODE Loaded":PRINT:PRINT,TAB(9)"Now l
  oading MAP-DRAW":PRINT:PRINT,TAB(8)"If PLAY key still down"
  :PRINT:PRINT CHR$(7):LOAD"!map-draw"
150 PRINT
160 PRINT
170 DATA cde1bb32@ec7647cde7bb32@ed76b8c8cd@6876c0ed73@ee76cd11
  bc4704,4316
180 DATA 3e08cb3f10fc32@dc7621@f27e22@f07621@f68622@f47e2100002
  2@e276,3013
190 DATA 22@e47622@e676cd09bbda@9076ed5b@f07621@e776010a00edb8,
  3143

```

```

200 DATA ed53@f076cd@6876204fcd@e8752a@e27622@e6762a@e67622@de7
6,3302
210 DATA 2a@f0762323235e23561313ed53@e076cd@6d752a@e67622@de762
a@f076,3015
220 DATA 2323235e23561b1bed53@e076cd@6d752a@e676ed4b@dc760922@e
676,2898
230 DATA ed4b@e476a7ed4238bc28ball@de762a@f07623010a00eab02b22@
f076c9,3450
240 DATA 2a@dc7622@ea76cd@34762a@e876ed4b@dc76a7ed4222@e476eb2a
@f47e,3680
250 DATA 722b73e5210000ed4b@dc76a7ed4222@ea76cd@34762a@e876ed4b
@dc76,3457
260 DATA 0922@e276ebel2b722b732bed5b@e076722b732b22@f47ec9ed5b@
de76,3463
270 DATA ed53@e8762a@e076ed5b@e876cdeabb2a@e876ed4b@ea760922@e8
76545d,3984
280 DATA 018002a7ed42d02a@e076cdf0bb473a@ed76b828d5c92a@de76018
002af,3635
290 DATA ed42301a2a@e076018f0led4230102a@e076ed5b@de76cdf0bb473
a@ed76,3441
300 DATA b8c9fe01c92a@f47e11@f6867dbb20037cbac83a@ed76cd@ebb2a@
f47e23,3981
310 DATA 5e2356ed53@e076234e2346e5eb5059cdc0bbel235e2356e52a@e0
76,3400
320 DATA cd6bbell1@f6867dbb20d97cba20d53a@ec76cd@ebb2a@e76f9c
90000,4250
330 DATA end

```

```

10 REM MAP-DRAW Len Heitman 3/5/85
20 REM Test for Mode : M=Mode number
30 h=HMEM:MEMORY h-9:b=HMEM+1:hi=INT(b/256):lo=b-hi*256:FOR
n=b+2 TO b+8:READ d:POKE n,d:NEXT:POKE b+6,lo:POKE b+7,hi:C
ALL b+2
40 M=PEEK(b):MEMORY h:RESTORE 50:DATA 205,17,188,50,0,0,201
50 REM For clear window #0 goto 999--For camera angles from
space goto 900
60 BORDER 2:INK 0,14:INK 1,6:INK 2,20:INK 3,24
70 WINDOW #0,1,14,9,25:WINDOW #1,1,6,1,8:PEN 1:PAPER #0,2:PAPE
R #1,2:CLS #0
80 PRINT:PRINT"Centre of Map:Long.® Lat.® Radius = 100":PRIN
T
90 INPUT "Centre Long.= ";B4:INPUT "Centre Lat.= ";B5
100 R=100:REM input "centre radius=";R
110 DEG:Z2=SIN(B5)*R:BB=SQR(R*R-z2*z2):X2=COS(B4)*BB:Y2=SIN(B4)
*BB
120 CLS:PRINT"CAMERA ANGLES ® DISTANCE":PRINT: INPUT " Camera
Azm.= ";xx0:X0=xx0+B4
130 INPUT" Camera ELV.= ";zz0:Z0=zz0+b5:INPUT" Camera Dis.= "
;A5:INPUT " Draw Size = ";A6
140 REM Map turns clockwise with + deg. If Not filling map
press 1 enter
150 INPUT "Rotate Ang";RT:PRINT:INPUT "Fill Enter OR
NOfill 1 Enter";NOF
160 CLS:A1=SIN(X0):A2=COS(X0):A3=SIN(Z0):A4=COS(Z0):Z3=A3*A5:B0
=SQR(A5*A5-Z3*Z3):X3=A2*B0:Y3=A1+B0
170 REM Set Origin
180 ORIGIN 360,199,81,639,399,0:CLG 0:WINDOW #0,1,6,9,25:GOTO 3
00:CLS
190 REM "RT Rotate"
200 Z=SQR(x*x+y*y):AZ=X/Z:aa=AZ:1F ABS(az)>=1 THEN AZ=0.9999999
210 ZZ=(90-ATN(aa/SQR(1-AZ*AZ)))*SGN(y):DZ=ZZ-RT:x=COS(dz)*z:y=
SIN(dz)*z:RETURN
220 REM "Transform X4-Y4-Z4 > X-Y"
230 X5=X4-X2:Y5=Y4-Y2:Z5=Z4-Z2
240 X6=x5*a2+a4+y5*a1*a4+z5*a3:Y6=y5*a2-x5*a1:Z6=z5*a4-x5*a2*a3
-y5*a1*a3
250 x=a5*y6/(a5-x6)*a6: y=a5*z6/(a5-x6)*a6
260 1F rt<>0 THEN GOSUB 200 REM rotate
270 X=CINT(X):Y=CINT(Y)
280 1F t=2 OR t=3 THEN DRAW x,y,c ELSE 1F t=1 THEN MOVE x,y ELS
E 1F t=4 OR t=5 THEN RETURN
290 ON K GOTO 370,310,620,530,570

```

```

300 REM READ 3
310 READ X4,Y4,Z4:T=INT(x4/1000):x4=x4-t*1000:ON T GOTO 330,330
,470,590,590
320 REM "Move-Draw"
330 C=x4:f1=SGN(y4):y4=ABS(y4):HE=INT(y4/1000):LON=y4-HE*1000:HE=HE*f1
340 f1=SGN(z4):z4=ABS(z4):K=INT(z4/1000):LA=(z4-k*1000)*f1
350 K=K+1:GOSUB 410:GOTO 230: REM To transform

360 REM "Read 2"
370 READ LON,LA:f1=SGN(LON):LON=ABS(LON):HE=INT(LON/1000):LON=LON-he*1000:he=he*f1
380 f1=SGN(LA):LA=ABS(LA):k=INT(LA/1000):LA=(LA-k*1000)*f1
390 k=k+1:t=2:GOSUB 410:GOTO 230
400 REM LON,LA -> x4-y4-z4
410 IF LON>600 THEN LON=360-LON+600
420 IF LA>600 THEN LA=360-LA+600
430 rr=r+he:z4=SLN(la)*rr:aa=SQR(rr*rr-z4*z4):x4=COS(LON)*aa:y4=SLN(LON)*aa:RETURN
440 REM "V VV check negative Long.and Lat.data"
450 IF V>600 THEN VV=360-V+600 ELSE VV=V
460 RETURN:REM "Grid Lines T=3"
470 he=0:GL1=INT(y4/1000000):y4=y4-GL1*1000000:V=INT(y4/1000):GOSUB 450:LN1=VV:V=y4-V*1000:GOSUB 450:LN2=VV:IF LN1>LN2 THEN LN2=LN2+360
480 c=x4:IF M=2 THEN c=1
490 GL2=INT(z4/1000000):z4=z4-GL2*1000000:V=INT(z4/1000):GOSUB 450:LA1=VV:V=z4-V*1000:GOSUB 450:LA2=VV:IF LA1>LA2 THEN LA2=LA2+360
500 k=4:s1=(LN2-LN1)/(GL1-1):s2=(GL1-1)*3:s3=(LA2-LA1)/(GL2-1):st=s1/3:LON=LN1:LA=LA1:FOR NL=1 TO GL2:t=1
510 FOR n=1 TO s2+1:GOSUB 400:GOTO 230
520 REM k4 return
530 t=2:LON=LON+st:NEXT n:LON=LN1:LA=LA+s3:NEXT NL
540 k=5:s4=(GL2-1)*3:st=s3/3:LON=LN1:LA=LA1:FOR NL=1 TO GL1:t=1
550 FOR n=1 TO s4+1:GOSUB 400:GOTO 230
560 REM k5 return
570 t=2:LA=LA+st:NEXT n:LA=LA1:LON=LON+s1:NEXT NL:GOTO 310
580 REM --FILL-- If NOFill=1 jump direct to border lines
590 IF Nof=1 THEN RESTORE 2310:GOTO 300 ELSE p=x4:lon=y4:la=z4:GOSUB 400:GOSUB 230:x=x+360:y=y+199:IF x<81 OR x>639 OR y<0 OR y>399 THEN GOTO 300 ELSE ORIGIN 0,0,0,639,399,0
600 start=h+1:xa=&76DE-30000+start:ya=&76E0-30000+start:PLOT 800,800,p:POKE xa,x-256*INT(x/256):POKE xa+1,INT(x/256):POKE ya,y-256*INT(y/256):POKE ya+1,INT(y/256)
610 CALL start:ORIGIN 360,199,81,639,399,0:GOTO 300
620 REM "End"
630 IF M=0 THEN 660
640 RESTORE:PEN #1,1:CLS #1:LOCATE #1,1,1:PRINT#1,b4:LOCATE #1,1,2:PRINT#1,b5:LOCATE #1,1,3:PRINT#1,"A"xx0:LOCATE #1,1,4:PRINT#1,"E"zz0
650 LOCATE #1,1,5:PRINT#1,"D"a5:LOCATE #1,1,6:PRINT#1,"S"a6:LOCATE #1,1,7:PRINT#1,"R"rt
660 WINDOW #0,1,9,8,25:PRINT CHR$(7):PRINT CHR$(7):END
900 REM Camera Angles to Satellite
910 PRINT"To get Camera angles and distance from Map Centre to special place in Space Geostationary satellite radius 42192 earth radius 6378 proportion 6.62 times earth radius. Radius=662"
920 INPUT"Camera Long.=";LON:INPUT"Camera LAT.=";LAT:INPUT"radius=";r:DEG:GOSUB 960:x1=x0:y1=y0:z1=z0
930 INPUT"Centre LONG.=";LON:INPUT"centre LAT=";LAT:r=100:GOSUB 960:x2=x1-x0:y2=y1-y0:z2=z1-z0:b5=SQR(x2*x2+y2*y2):a5=SQR(b5*b5+z2*z2)
940 LO=x2/b5:LO=(90-ATN(LO/SQR(1-LO*LO)))*SGN(y2):LA=b5/a5:LA=(90-ATN(LA/SQR(1-LA*LA)))*SGN(z2):IF lo<0 THEN lo=lo+360
950 x0=LO-LON:z0=LA-LAT:PRINT"AZIM.=",USING"####.##";x0:PRINT"ELV.=",USING"####.##";z0:PRINT"Distance=",USING"#####.##";a5:END
960 z0=SLN(LAT)*r:ab=SQR(r*r-z0*z0):x0=COS(LON)*ab:y0=SLN(LON)*ab:RETURN
990 REM RESET #0 WINDOW
999 WINDOW 1,80,1,25:CLS:END

```

```

2000 REM AUSTRALIAN COAST
2010 DATA 1001,140.97,-38.05,143.5,-38.85,144.53,-38.27,144.62,-
38.28,144.72,-38.18,144.63,-38.11,144.5,-38.16,144.35,-38.1
5,144.38,-38.08,144.5,-38.08,144.92,-37.85,145.13,-38.15,14
5,-38.3,144.82,-38.37,144.66,-38.32,144.9,-38.5,145.22
2020 DATA -38.38,145.27,-38.22,145.5,-38.26,145.55,-38.42,145.37
,-38.55,145.92,-38.9,146.07,-38.82,146.43,-39.15,146.48,-38
.78,146.3,-38.9,146.18,-38.75,146.92,-38.57,147.72,-37.92,1
49.5,-37.77,150,-37.5,151.72,-33,153.03,-31,153.6,-28.87
2030 DATA 153.5,-28.17,152.9,-25.3,150.8,-22.7,146.75,-19.18,145
.2,-14.83,143.78,-14.45,143.2,-12,142.85,-11.85,142.75,-11,
142.53,-10.7,142.13,-11,141.58,-13
2040 DATA 141.28,-15.5,140.8,-17.47,140,-17.72,139,-16.9,138,-16
.74,135.43,-14.92,136.07,-13.67,135.85,-13.42,136.47,-13.25
,136.98,-12.35,136.4,-12,136.22,-12.48
2050 DATA 135.5,-12.07,134,-11.87,132.92,-11.33,132,-11.15,131.7
8,-11.32,132.58,-11.65,132.57,-12.1,131,-12.1,130.57,-12.4,
129.37,-14.42,129.62,-15,129,-14.88
2060 DATA 128.38,-14.78,128.12,-15.2,128.18,-14.72,126.92,-13.58
,126.62,-14.18,126.37,-14,126,-14.5,125.25,-14.58,124.24,-1
6.42,123.58,-16.16,123.58,-17.5,122.92
2070 DATA -16.4,122.16,-17.35,122.22,-17.95,122.42,-18,121.35,-1
9.5,116.66,-20.72,114.66,-21.83,114.37,-22.5,114.13,-22.5,1
14.17,-21.8,113.67,-22.57,113.15
2080 DATA -24.63,114.2,-26.2,113.47,-25.6,113.87,-26.5,112.92,-2
5.5,114.9,-29.25,115.7,-30.5,115.75,-31.88,115.66,-33.2,115
.3,-33.61,115.02,-33.51,115.05,-34.27
2090 DATA 116,-34.81,117.93,-35.12,120,-33.92,123.5,-33.95,124.1
8,-33,126,-32.27,127,-32.28,129,-31.78,131.17,-31.45,133.65
,-32.17,135.63,-34.92,137.2,-33.67
2100 DATA 137.77,-32.5,137.92,-33.6,136.87,-35.3,137.75,-35.12,1
38.4,-34.15,138.5,-35.07,138.2,-35.62,138.8,-35.7,139.6,-36
.95,140.97,-1038.05
2110 REM TASMANIA COAST
2120 DATA 1001,148,-40.75,148.33,-41,148,-43.25,147.33,-42.85,14
7.3,-43.5,146.05,-43.55,145.3,-42.6,144.63,-41,144.75,-40.6
8,146.37,-41.17,148,-1040.75
2130 REM NEW ZEALAND south coast
2140 DATA 1003,169.35,-46.6,166.69,-46.23,166.47,-45.66,168.36,-
44,170.77,-42.9,172.73,-40.52,173.2,-41.35,173.95,-40.86,17
4.2,-41.1,174.05,-41.46,174.28,-41.76,173.22,-43.01,172.71,
-43.4,173.13,-43.72,173.03,-43.91,172.46,-43.83
2150 DATA 171.25,-44.37,170.64,-45.83,169.35,-1046.6
2160 REM NEW ZEALAND north coast
2170 DATA 1003,178.54,-37.7,178,-37.53,177.08,-38,176,-37.61,175
.8,-36.11,175.3,-36.6,174.28,-35.27,172.68,-34.49,173.05,-3
5.21,174.4,-37.5,174.33,-38.86,173.81,-39.52,174,-39.57,175
.06,-40
2180 DATA 175.2,-40.56,174.8,-41.2,175.33,-41.63,176,-41.18,177.
08,-39.64,176.92,-39.42,177.25,-39.08,178,-39.28,178.54,-10
37.7
2190 REM NEW GUINEA
2200 DATA 1003,150.78,-10.52,150.33,-10.4,150.09,-10.27,149.7,-9
.82,150.05,-9.68,149.2,-9.35,149.32,-9.03,148.6,-9.1,146.98
,-6.72,148.02,-6.59,145.78,-5.43,145.78,-4.83,143.62,-3.45,
137.84,-1.51,135.3,-2.6,134.1,-2.1,134.05,-.86
2210 DATA 132.41,-.04,130.07,-1.03,133.1,-4.22,133.5,-3.7,138.1,
-5.43,138.62,-6.72,137.59,-8.32,139.74,-8.1,142.59,-9.31,14
4.48,-7.59,146.12,-8.19,147.93,-10.17,150.34,-10.73,150.78,
-1010.52
2220 REM Fill Australia
2230 DATA 4001,144,-30
2240 REM Fill Tasmania
2250 DATA 4001,146.8,-42
2260 REM Fill N.S.New Zealand
2270 DATA 4003,168.8,-45.2,4003,175.8,-38.8
2280 REM New Guinea Fill
2290 DATA 4003,142.3,-7.8
2300 REM STATE BORDERS
2310 DATA 1000,141,-38.05,141,-34.03,141,-29,141,-26,138,-26,134
,-26,129,-1026,1000,129,-31.68,129,-29,129,-26,129,-22.5,12
9,-19,129,-1014.88,1000,138,-16.75,138,-20,138,-23,138,-102
6,1000,141,-29,145,-29,149,-29,149.5,-28.6,150.33,-28.53
2320 DATA 151.4,-29.18,152.4,-28.35,153.5,-1028.17,1000,150,-37.
5,148.2,-36.8,147.7,-35.93,146.55,-36.05,145,-35.87,144.75,
-36.15,142.77,-34.6,142.63,-34.8,142.08,-34.12,141,-1034
2330 REM LONGITUDE-LATITUDE lines. END
2340 DATA 3002,4105185,5660020,1000,132,-2027

```

The Learning Centre

In previous articles, we have discussed redefining certain keyboard characters to assist your programming efforts. This month we go the whole way with a lengthy contribution from C. Rogers of Fraser who, with the aid of a program called KEY, has converted most of the keyboard characters into predefined strings.

How often have you slogged away entering program code, character by character, and thinking "There must be an easier way!". Well short of getting someone else to enter the program for you, this is the nearest thing to it. Once you have run the KEY program, then by merely pressing the green CTRL key and simultaneously pressing some other key (see the Keyboard Table), your computer will instantly interpret this as a request for a predefined string to be used instead of the actual character you pressed.

This is all possible due to the Amstrad CPC464 having a completely software controlled keyboard. But before you enter the program, you should read the following explanation. Don't just use the program blindly. Try to understand how and why it does what it does.

We've all read the User Manual haven't we. Well I hope so! In doing so you will have encountered the <KEY> and <KEY DEF> commands. If you were confused, I'm not at all surprised. Let me explain.

THE AMSTRAD CHARACTER SET

There are 256 characters (or symbols) available for use on your Amstrad CPC464. This is commonly called the ASCII character set – American nation Standard Code for Information Interchange. In truth, ASCII normally applies to a character set of only 128 characters, but Amstrad like many other manufacturers has stretched their character set to a full 256 characters. These characters are numbered 0 to 255 (see your User Manual Appendix III.1 & III.2).

So why start at zero and finish at an odd number like 255? It's because of

the way in which each character is stored in memory. I'll explain.

BYTES

A 'byte' is the most common way to measure the size of a computer's memory. It is easiest to think of a byte being equivalent to one character, where a character may be either a letter of the alphabet, or a number, or any other sort of symbol. The CPC464 memory is 64K bytes long – ie 64 x 1024 characters.

Each character or byte is composed of eight 'bits'. A bit is the smallest unit of information that a computer can handle. Believe it or not, your computer does just about everything 'bit' by 'bit'. This is because it is a digital computer, just like 99.9% of the worlds computers. The other 0.1% are called analogue computers, and are used solely for specialist scientific applications.

BITS

A bit has only two possible states, just like a light switch, either ON or OFF. These states are usually represented by <1> for ON and <0> for OFF. This is the realm of binary maths as used by all digital computers. Fortunately we don't have to program directly into our computers in binary, though it would result in ultra fast programs, if you ever managed to get them to work. Even Assembler programmers only rarely deal with binary numbers.

In the decimal numbering system that we are all so accustomed to, we use the symbols zero through nine for each digit in a number. Thus the decimal number 164 can be represented as:

$$1 \times 10^2 + 6 \times 10^1 + 4 \times 10^0$$

Now try entering:

```
PRINT BIN$ (164)
```

You should get <10100100> displayed. So decimal 164 is equivalent to binary 10100100 which can be represented as:

$$1 \times 2^7 + 0 \times 2^6 + 1 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0$$

On the Amstrad CPC464 a decimal (base 10) number is entered as you would expect, but a hexadecimal (base 16) number and a binary (base 2) number have to be entered with a prefix to distinguish them from one another. Look at the following example:

```
Decimal ..... 164
```

```
Hexadecimal .... &HA4
```

```
Binary ..... &X1010010
```

We are now ready to answer the question why our 'eight bit based' micro has 256 characters, number 0 through 255. Lets have the computer answer for us. Now enter the following:

```
PRINT &X00000000,&X11111111
```

These binary numbers represent the smallest and largest single values possible for identifying a particular character – only eight bits per character remember.

Voila! You get the (decimal) numbers 0 and 255 displayed! All a bit cumbersome, but if you can understand a little about binary numbers it will help you with understanding how your computer works.

CHARACTERS = NUMBERS

Lets look at the character 'A'. We can find out what ASCII number is assigned to a character by using the <ASC> command. So for the letter capital 'A' we would enter the following:

```
ASC("A")
```


The value displayed is the number 65. This means that the Amstrad CPC464 character set identifies the letter uppercase 'A' with the number 65. Now repeat the above step substituting with lowercase 'a'. Lo and behold the value displayed is the number 97. Every symbol that the computer understands, be it an uppercase letter, a lowercase letter, a decimal digit, a comma, a 'stick man' or whatever, has been allocated a number between 0 and 255. This is known as its ASCII number.

This number NEVER changes, though you can change what you want to happen when you press a particular key on the keyboard. Pressing the 'A' key need not necessarily tell the computer that you want it to use the letter 'a' or 'A'. This is the basis of the KEY program.

KEYS VERSUS CHARACTERS

Every key on the keyboard has a number associated with it. This is not the ASCII number for the symbol on the key itself, but rather it is the 'key' number. Key numbers range between 0 and 79. See the User Manual Appendix III page 16 for a diagrammatic presentation of the Key and Joystick, KEY numbers. Try not to get confused between the ASCII numbers which identify characters, and the key numbers which identify the actual buttons (keys) that you press.

Incidentally, it is the key number that is the 'variable' used in the <INKEY> command. Thus to detect in a program if someone is pressing the 'A' key - key number 67 - we would code:

```
IF INKEY(67) > -1 THEN . . . .
do routine when 'A' key pressed
regardless of any other keys.
```

Now, what happens when you switch on your computer is quite terrific. Every one of the keys is allocated four values.

1) The first is the 'repeat' switch with a value of either zero or one. One indicates that the key is self repeating. If you keep the key depressed, then the computer is

made to think that you are pressing the key many times. Zero indicates that it is not self repeating. This same feature is common to many typewriters.

2) The second is a number between 0 and 255. You guessed it! It's the number of the ASCII character that will be used when you press that key by itself.

The INKEY command to detect this where 'nn' is the key number is:

```
IF INKEY(nn) = 0 THEN . . . .
do routine when key 'nn' is
pressed by itself.
```

3) The third is similar to the second, except that it is the number of the ASCII character used when you press that key while simultaneously pressing the Shift key.

The INKEY command to detect this is:

```
IF INKEY(nn) = 32 THEN . . . .
do routine when key 'nn' is
pressed and Shift key is also
pressed.
```

4) Finally, the fourth value, like the second and third, is the number of the ASCII character used when you press that key while simultaneously pressing both the Shift and Control (CTRL) keys.

The INKEY command to detect this is:

```
IF INKEY(nn) = 128 THEN . . . .
do routine when key 'nn' is
pressed and Shift and Control
also pressed.
```

The INKEY command can be used to detect ANY combination of keys being pressed. See the User Manual Ch.8 page 20 for further details.

Now before I explain how to alter the ASCII values for your keys, look at the User Manual Appendix III page 14 for the 'Default ASCII values'. Notice how the numeric keypad has not been covered on page 14, but rather on page 15. I'll explain this first.

EXPANSION STRINGS

Now although I earlier stated that ASCII number 65 is uppercase 'A;

and that this never changes, there are actually thirty two ASCII values that you can change. These are ASCII 128 through 159. It is these values that can be changed or 'expanded' and as such they are known as the 'Expansion Values' and the data linked to them as the 'Expansion Strings'.

This means that when you press a key assigned an ASCII value in the Expansion Value range, you can get more characters than just the default single character associated with that ASCII number. These characters, the Expansion String, can be set to none, one or many characters.

If you now look at your User Manual Appendix III page 15, you will see which keys already have an Expansion String assigned to them. They are the twelve keys that comprise the Numeric keypad.

Note that the small ENTER key has two strings associated with it. One is for normal or shifted use. This is CHR\$(13) or quite simply ENTER. The second is for control use and is the only default expansion with more than one character assigned to it. Thus when you press the small ENTER key and the control (CTRL) key simultaneously you get an expansion string of <RUN"+chr\$(13) }. It is this combination that you use to load and run the majority of purchased software.

Notice then, how in using the default expansion of the CTRLed small ENTER key, we can convert two 'keystrokes' into five characters. Think of the possibilities!

THE 'KEY' COMMAND.

It is the <KEY> command that we must use in order to create our own 'expansion strings'. Lets have a go ourselves. Try entering:

```
KEY 128,CHR$(34)+"This is
my OWN expansion"+CHR$(34)
If it didn't work, then reset your
computer (press CTRL/SHIFT/ESC)
and try again.
```

If you now press the zero key on the Numeric keypad you will not get a zero displayed. Instead you will get

the expansion string as entered above. Although this is only a simple example, very complex strings can be set up. Now lets take it one step further and introduce the <KEY DEF> command.

THE 'KEY DEF' COMMAND

It is the <KEY DEF> command that we use to associate the hardware keyboard buttons with the software ASCII character set. Remember how in KEYS VERSUS CHARACTERS I explained the four values assigned to each key when the computer is switched on? The <KEY DEF> command is there to allow us to alter these default settings.

Now without resetting the keyboard enter the following:

```
KEY DEF 69,0,63,128,13
```

Now press the 'A' key:

firstly, by itself

secondly, with the shift key

lastly, with the control key.

Delightful isn't it. I'll explain what you just did.

In the KEY DEF command just entered:

- 69 is the KEYboard number for the 'A' key,
- 0 indicates a non-repeating key,
- 63 is the ASCII value of '?' – the shortened form for PRINT,
- 128 is the ASCII value now linked to the expansion string we entered earlier,
- 13 is the ASCII value for ENTER – ie chr\$(13).

So you have just entered a direct mode BASIC command – via the 'A' key – to request the computer to print the expansion string entered previously. Notice how the screen displayed the command as you entered it, followed on the next line by the result of executing that command. We could conceivably have also combined the '?' and the ENTER into the expansion string itself, so as to further reduce the keystrokes required.

This isn't a very practical example as it leaves your keyboard without the

ability to enter either an upper or lower case 'A'. Reset your computer now before you forget!

RESTRICTIONS

By now you can appreciate the enormous possibilities! There are six restrictions to keep in mind:

- 1) There are only 32 Expansion Strings available.
- 2) Pressing CTRL/SHIFT/ESC or switching your computer off, will reset all the KEY and KEY DEF settings to their default values.
- 3) You can only allocate a maximum of 32 characters to each of the 32 expansion strings.
- 4) Despite what the User Manual says in Ch.8 page 23, there is actually a full 120 characters set aside for the 32 expansion

*Don't pass the buck
– earn it*

LETTERS published in The Amstrad User attract a payment of \$5. We don't mind whether you want to pass on a tip, have a moan, make an observation or whatever – merely put pen to paper and direct your comments to the Editor.

CARTOONS can also earn you \$5. Just make sure they are drawn in black ink on white paper and not folded when you submit them to the Editor, who may well be folded when he sees them.

Please send your letters or cartoons to: The Editor, The Amstrad User Suite 4a, 33-45 The Centreway, Blackburn Road, Mt. Waverley 3149.

strings.

- 5) Although it may seem possible to change the key defaults for any key to any ASCII value, it is in fact not that flexible. Use of ASCII values in the range 0 to 31 – other than 13 (ENTER) – is not recommended. These are screen control codes. Using them may cause unexpected results! Values 224 to 254 will also pose some problems. These should also be avoided as some of them are reserved for the Escape, Edit (cursor control) and Copy keys. Value 255 will be ignored so don't waste your time. Experiment if you like. You can't hurt the memory of your computer through any software you run. At worst you will confuse it.
- 6) The Escape, Shift and Control keys (key numbers 66, 21 and 23) may not be altered. Just think of the havoc if you could!

We can't change points 1, 2, 3, 5 or 6, but we can change point 4 by allocating another (larger) buffer for the computer to use in which to hold the new expansion strings. Look at the machine code routine generated in lines 1000-1090 in the KEY program. This allocates an ample buffer area outside of the restrictive 120 character buffer area. (*Hey – some of us are not into machine code yet – Ed.*)

IN CONCLUSION

If you have understood even part of what you have read then Bravo! You are well on your way to appreciating the power and flexibility of your Amstrad CPC464.

Practice will make it all the more clear to you. So now enter the KEY program exactly as it is listed. Be sure to SAVE it **before** you RUN it. When RUN, the program will first set all the keys, then line 150 will clear the program from memory! Once entered and saved, you may wish to try your own variations. Reload the program then change the DATA statements for either the expansion strings or the key allocations or maybe both.

APOLOGIES

In the KEY program, many of the expansion strings begin with a colon – a statement separator. Although this isn't necessary at the start of a new line, it won't hurt. They are included because of their need in multi-statement lines.

If their presence offends you, or you somehow become a bit squeezed for memory, then after entering all the code for a line but before pressing the ENTER key:

- 1) Simultaneously press the CTRL and the UP cursor-positioning key. This will place the cursor at the first character in the line, even if the line is the full 255 characters long.
- 2) Press the CLR key once. The colon will disappear.
- 3) Press the ENTER key. Your corrected line is now entered.

program is to reset the keyboard to facilitate easier entry of program code, you could quite conceivably adjust the expansion strings to suit many other applications.

An example would be for an Adventure game where you could setup the cursor positioning keys to equate to the four most common Adventure game commands – 'GO NORTH' ; 'GO EAST' ; etcetera.

The possibilities are only limited by your imagination.

SUGGESTIONS

Although the purpose of the KEY

```
1 GOTO 9
2 SAVE "KEY.BAS" : STOP
3 ' SOFTWARE TYPE : Utility
4 ' FIRST ENTERED : 29/04/85
5 ' LAST UPDATED :
6 ' COMMON NAME : Set "Key Defaults" Utility
7 ' DESCRIPTION : The purpose of this program is to reset t
  he keyboard, so as           to provide for maxim
  um efficiency while entering programs.
  ie. One KEY-press gives MANY characters !
8 '
9 CALL &BBFF : PAPER 0 : PEN 1 : REM - General Purpose Initia
  lisation
10 GOSUB 1010
20 GOSUB 2010
30 GOSUB 3010
40 BORDER 10
50 PAPER 0
60 INK 0,10
70 PEN 1
80 INK 1,0
90 MODE 2
100 CLS
110 PRINT CHR$(24)
120 PRINT SPACE$(80)+" Your entire keyboard has been reset, an
  d is ready for entering BASIC code... "+SPACE$(80)
130 PRINT CHR$(24)
140 PRINT CHR$(7)
150 NEW : REM **** This ensures memory is clear of this code
160 END
999 '
1000 REM **** Machine Code Routine to increase buffer size to ar
  ound 1000 chars
1010 RESTORE 1090
1020 MEMORY &8FFD:base%=&9000
1030 FOR shift%=0 TO 9
1040 READ mcode%
1050 POKE base%+shift%,mcode%
1060 NEXT shift%
1070 CALL &9000
```

```

1080 RETURN
1090 DATA 17,10,144,33,0,4,205,21,187,201
1999 '
2000 REM **** Setup the Expansion Strings
2010 RESTORE 2150
2020 FOR expankey%=128 TO 159
2030 READ expanval$,enterstart%,enterend%
2040 IF enterstart%=1 THEN expanval$=CHR$(13)+expanval$
2050 IF enterend%=1 THEN expanval$=expanval$+CHR$(13)
2060 KEY expankey%,expanval$
2070 NEXT expankey%
2080 RETURN
2090 '
2100 ' The DATA statements should not be concatenated. They hav
e been
2110 ' intentionally set out as 32 separate Data statements so a
s to
2120 ' easily allow you to alter those particular Expansion Stri
ngs
2130 ' that you may feel are currently inappropriate for your ne
eds.
2140 '
2150 DATA "ink 0,10:ink 1,0:border 10:list ",1,0
2160 DATA " then ",0,0
2170 DATA ":goto ",0,0
2180 DATA " else ",0,0
2190 DATA ":locate ",0,0
2200 DATA ":gosub ",0,0
2210 DATA ":return",0,0
2220 DATA "auto ",1,0
2230 DATA "renum 10,10,10",1,0
2240 DATA ": rem **** ",0,0
2250 DATA "call &bbff:paper 0:pen 1:mode 2",1,1
2260 DATA "run ",1,1
2270 DATA "mode 2:cls",1,1
2280 DATA ":asc()",0,0
2290 DATA ":border ",0,0
2300 DATA "chr$()",0,0
2310 DATA ":draw ",0,0
2320 DATA "=inkey$:if ",0,0
2330 DATA ":if inkey()>-1 then ",0,0
2340 DATA "lower$()",0,0
2350 DATA "mid$()",0,0
2360 DATA ":next ",0,0
2370 DATA ":origin ",0,0
2380 DATA ":plot ",0,0
2390 DATA ":print #ps%",0,0
2400 DATA ":return",0,0
2410 DATA "space$()",0,0
2420 DATA "upper$()",0,0
2430 DATA "val()",0,0
2440 DATA ":ink ",0,0
2450 DATA ":for x=",0,0
2460 DATA ":for wait%=0 to 1782 :next wait%",0,0
2999 '

```

```

3000 REM **** Reset the KEY DEFaults to allow more effective pro
      gramming
3010 RESTORE 3090
3020 pushkey$="start"
3030 WHILE pushkey$<>" "
3040 READ pushkey$,keynum%,repeat%,normal%,shifted%,control%
3050 KEY DEF keynum%,repeat%,normal%,shifted%,control%
3060 WEND
3070 RETURN
3080 ' Alphabetic Keys
3090 DATA "A",69,1,97,65,141
3100 DATA "B",54,1,98,66,142
3110 DATA "C",62,1,99,67,143
3120 DATA "D",61,1,100,68,144
3130 DATA "E",58,1,101,69,131
3140 DATA "F",53,1,102,70,158
3150 DATA "G",52,1,103,71,133
3160 DATA "H",44,1,104,72,130
3170 DATA "I",35,1,105,73,145
3180 DATA "J",45,1,106,74,146
3190 DATA "K",37,1,107,75,157
3200 DATA "L",36,1,108,76,147
3210 DATA "M",38,1,109,77,148
3220 DATA "N",46,1,110,78,149
3230 DATA "O",34,1,111,79,150
3240 DATA "P",27,1,112,80,151
3250 DATA "Q",67,1,113,81,0
3260 DATA "R",50,1,114,82,153
3270 DATA "S",60,1,115,83,154
3280 DATA "T",51,1,116,84,129
3290 DATA "U",42,1,117,85,155
3300 DATA "V",55,1,118,86,156
3310 DATA "W",59,1,119,87,159
3320 DATA "X",63,1,120,88,158
3330 DATA "Y",43,1,121,89,0
3340 DATA "Z",71,1,122,90,159
3350 ' Odds & Sods
3360 DATA "?",30,1,47,63,152
3370 DATA "CLR",16,1,16,16,140
3380 ' Numeric Keypad
3390 DATA "little ENTER",6,0,13,0,139
3400 DATA "decimal point",7,0,46,0,138
3410 DATA "0",15,0,48,0,128
3420 DATA "1",13,0,49,0,129
3430 DATA "2",14,0,50,0,130
3440 DATA "3",5,0,51,0,131
3450 DATA "4",20,0,52,0,132
3460 DATA "5",12,0,53,0,133
3470 DATA "6",4,0,54,0,134
3480 DATA "7",10,0,55,0,135
3490 DATA "8",11,0,56,0,136
3500 DATA "9",3,0,57,0,137
3510 ' This must be the last DATA statement
3520 DATA "",76,0,0,0,0

```

A Review of ams-FORTH

Is FORTH a new dimension in programming? Should FORTH be your first choice in systems development? Shane Kelly takes a more serious look at this relatively new 'language' available for the CPC464

FORTH is a programming language that has its beginnings in astrological observatories. The language was used to program computers to operate the giant telescopes of these observatories. In actuality, FORTH is a programming system rather than just a language. FORTH systems that are true to the model of FORTH as laid down by the FORTH INTEREST GROUP, a body of dedicated users who decide on the standard FORTH, contain a compiler, an interpreter and a virtual memory system.

So how close is ams-FORTH to the model? About as close as it is possible to come on a cassette based system. For your money you get the ams-FORTH cassette and a user manual. The documentation justifies the title of user manual in this instance because after reading it you will be able to use ams-FORTH with some degree of confidence. The manual takes you briefly but clearly through an introduction of the FORTH language (Note – not a tutorial, but an introduction), how to install it on your system and where it all fits in your AMSTRAD, and then goes into quite a bit of detail on how to use the special sound and graphics commands built into ams-FORTH. The section on using the editor is a bit bare bones and relies heavily on the FIG FORTH MODEL editor which no-one will ever claim is sophisticated. There is a section on using the printer and a most welcome addition, a glossary of the FORTH language – once again borrowed from the FIG-FORTH documentation and software is perfectly acceptable as it is in the public domain. What you pay for is the

convenience of having it integrated into your system. As one who has brought a FORTH system up from scratch on a cassette based system, let me tell you that having someone else do the hard work for you is worth twice the price you pay for the program!

So, why would anyone buy ams-FORTH? And what can you do with it when you buy it? There is a theory that goes something like this:- the more programming languages you become familiar with the easier it becomes to envisage solutions to problems in terms of the computer. Simply put, it means that the more languages you know, the easier it is to program. FORTH is a language that fits in well the philosophy of the above statement because FORTH IS ANY LANGUAGE THAT YOU WANT IT TO BE! How can this be? FORTH is an extensible language – briefly this means that it is possible to add new commands to the language by defining new commands (called words in FORTH) in terms of old commands. The possibilities are tremendous. Allied to this is the fact that FORTH is much faster than an interpreted language like BASIC and only a little bit slower than machine language, and you can begin to grasp the possibilities of this rather unusual language. Because FORTH is extensible you can write your own graphics language or a peripheral control program that can act in real time. Literally anything is possible with FORTH given the usual constraints of storage capacity and your imagination.

Now a few words on the storage system as implemented in ams-FORTH. At this time cassette only is

the norm, but on the front of the user manual is the statement:- This software is a prelude to the release of CPC-464 Multi-FORTH 83, a Multi-tasking FORTH 83 Sideways ROM for the AMSTRAD. So it looks like there will be further enhancements to the system as time goes on. If the updates do come to pass, then I would assume that they would be for Disc-based machines. Since the output from the ams-FORTH EDITOR is an ASCII file I would venture to say that it would not be impossible to transfer your files to the Disc-based version should it become available.

So far this has been a review that is in danger of being over positive. There IS a negative side to FORTH and this is that it is extremely difficult to become proficient in, due mainly to its propensity for obscure symbols and Reverse Polish Notation and it's reliance on the stack concept. If you have the time to spend, it will be well worth your while to acquire more than a nodding acquaintance with this versatile and rewarding programming system.

To sum up:-

1. ams-FORTH is an excellent implementation of FIG FORTH 79 for the AMSTRAD
2. The documentation actually helps you understand the program instead of obscuring it, as some do.
3. The price is pitched at a level that means you can buy it even if you are only mildly interested (Be warned – FORTH makes fervent converts!!)

This product is recommended with confidence.

The Trials of Tony Blakemore

A column dedicated to the absolute beginner

I had intended this month to discuss character and key redefining. But with the activity and rapid growth of user groups I think we should spend some time discussing the groups and how valuable they are for helping new users.

I will never forget the first time that we held a user group meeting. Over a period of time a group of dedicated users of a computer that is best left nameless, had grown to the point where holding the meetings in our homes had become out of the question. It was decided that a hall should be rented and yours truly was elected to organise it. A hall was found and all the members of the group contacted and all promised to turn up on the elected night.

The night of the meeting arrived and what a night it was. I had no sooner loaded all the gear into the car when the heavens opened and one of Melbourne's famous showers arrived. Within minutes I could

hardly see the road in front of me. Still everyone had said they would turn up and what was a drop of rain!! If anything the weather got worse, but undaunted I drove on. When I arrived at the hall I was the first. Still it was early yet. One hour later I was still the only one there. Then I heard footsteps. Another body had arrived. I did not know the person but then the members had said they would try and bring new people with them. Determined to get the meeting underway I said "It's very good of you to turn up on such a dreadful night, how about we get the meeting under way". The answer I received was short and to the point. "Meeting be damned. I am the caretaker and did not think anyone would be silly enough to turn up on a night like this. If you have finished mucking about why don't you go home so that I can do the same."

So much for the first meeting. Happily to report within a few months we had a very strong user group that is still in existence and I often have a chuckle to myself when I think about it. One of the hardest things to find when you first start learning is someone to help with problems which at the time seem insurmountable. Everyone that you knew was only starting and really could not help much. We have all gone through it, and what a relief to find a group of users that have formed a club and to quickly find that the huge problem was not such a big problem after all.

One of the criticisms of computers that is often heard is that it is a very insular hobby. One person huddled round a computer to the exclusion of everything and everyone else. Nothing could be further from the truth. I have made more lasting friends from contacts through the various groups than I ever thought possible. We all are beginners at different levels and it does not matter that you have only a few weeks experience. You may have

other skills that the group will find invaluable. After all someone has to be president or secretary, treasurer or committee man. (Whoops, 'Committee person'). I have not yet found anyone who is not prepared to sit down with a brand new user and spend as much time as is necessary to solve whatever problem that they may have.

And what a variety of people that you do meet. Everyone from University professors, very clever in their own field, who are learning the basics from someone who has never had any formal education. Retired people who, looking for an interest, have started to learn Basic well into their sixties. School children younger than your own whose knowledge puts you back into the beginners class. Adventure fanatics who love nothing more than to spend two or three weeks looking for obscure clues designed to send most people round the bend. 'Zap 'em' types who accumulate huge scores shooting down the baddies. Serious programmers who would never be seen dead playing games and who spend weeks peering through roms and rams to discover some more helpful routines that have everyone amazed. All types and all prepared to sit down and very patiently explain the basics to someone who has just started.

Now you may say that I am labouring the point a bit but I cannot think of anyway that is better to learn the basics than to talk to someone who has been there and done that!! You can spend a fortune on books (more than twenty now available) and magazines and not learn as much as you can by ten minutes with someone who can show you where you may be going wrong.

Support your local groups. If you want a contact in your area ring me at the weekend on (03) 878 6212 and I will put you in contact with the organiser for your area. Next month we will get back to basics.

NEXT MONTH

We take the covers off the new CPC664

This exciting addition to the AMSTRAD range will be reviewed in full, including information on new instructions available with this machine.

Sounds Good

Up to now, we have not investigated the sound potential of the CPC464. Fortunately, Peter Campbell of Hobart has corrected this with an impressive Sound Experiment Program which provides all the parameters once you have selected your required sound.

Program Commentary:

This "Sound Experiment Program" is intended to be read in conjunction with Chapter 6 of the CPC464 User Instructions. It displays the parameters on screen, while playing the associated sound. One volume envelope is provided, limited to four segments, and the single tone envelope is restricted to two segments. The channel parameter is limited to a maximum value of 5.

As it is theoretically possible to set up a sound with a duration exceeding 20 months(!), the program contains error traps to make long durations difficult to set accidentally. Should you succeed in setting up such a sound, the controls will appear to be "frozen" and you will probably be tempted to hit ESC twice and then RUN the program again to recover from the situation. However, if you hold down "E" patiently, you will eventually be given the option of continuing with the duration reset to "10".

The program is operated by pressing a cursor key and, while holding it down, one of the keys on the main keyboard, or on the numeric keypad. UP and DOWN cursor keys give unitary movement through the parameters (units of ten in tone period) and LEFT and RIGHT cursor keys give ten times faster movement, where appropriate. Where the permissible settings are confined to "0" and "1" the cursor keys act as toggles.

The "heart" of the program is contained in line 210, which converts the detected key into a GOSUB instruction. The called subroutine changes the parameters and calls further subroutines to print out the parameter on screen and to sound the associated note. Where appropriate error trapping subroutines are called.

Variables are generally those used in Chapter 6 of the User Instructions and the key to press is generally the same as the variable. Exceptions were necessary in the case of the volume and tone envelopes. The volume envelope is set from the numeric keypad whilst the tone envelope uses not only T,V and W, but also the nearby F,C and A.

Provision has been made for playing Octaves -1, 0 and 1, both playing all the notes and playing a major scale over two octaves. If you can plug Arnold into a stereo amplifier, the base notes, in particular, will be much improved. You will also be able to compare Channel C (4) where half the sound is fed to each speaker and channel

value 5 where the whole sound is fed to each speaker. The difference is quite pronounced!

The program could be enhanced by adding a graphics screen to show the envelopes in graphic form, but I suggest you first get out pencil and graph paper, sketch some envelope shapes, convert them into their numerical parameters and use the program to play the sound. This will teach you far more about the relationship between envelope shape and resultant sound than just looking at the shape on screen. Besides, with the shape on paper and the numerical parameters on screen, you can look at both simultaneously!

How it Works

LINES 540, 1120, 1600 etc.

The inverted commas in the remarks on these lines contained an imbedded CTRL L symbol to trigger form-feed on a Mitsui 2100 printer.

LINES 60-80

Call setting up routine (3380).

LINES 100-160

Detect E and wait for and detect other key input. Note the nested WHILE WEND loops. Except when E is pressed, the program is controlled by these loops.

LINES 200-230

This is the heart of the program and selects the appropriate subroutine when a key is pressed.

LINES 250-300

This routine is invoked when you opt to end the experiment, resetting the small ENTER key and choosing suitable colours for mode 2, which is, however, not automatically chosen.

LINES 320-400

Because of the repetitious structure of most of the subroutines which adjust the parameters this general routine is invoked after converting the individual variables into the general form, "n", with limits, "lim1" and "lim2" and variations, "var1" and "var2". Lines 380 and 390 prevent the variable exceeding the limits.

LINES 420-440

Sounds the note or noise.

LINES 460-530

Subroutine invoking both the tone and volume envelopes.

LINES 550-1110

Subroutines for setting the various parameters of the sound command.

LINES 1130-2070

Subroutines for setting the four segments of the volume envelope.

LINES 2090-2550

Subroutines for printing the screen parameters.

LINES 3080-3200

Subroutine for detection of, and recovery from, excessively long duration sounds.

LINES 3220-3340

Subroutine giving the option of ending the experiment completely or continuing with the duration reset to 10.

LINES 3360-3670

Subroutines setting up the original screen layout. Lines 3560-3680 are also called to restore the screen after warning messages etc. have been displayed.

LINES 3690-4490

Subroutines which add the option of playing "true" musical notes from octaves -1, 0 and 1 either all in succession or proceeding through them playing a major scale. The sub-program works by READING DATA into arrays (see line 3870), from whence it is used as required. An alternative approach can be devised using a mathematical formula, which will give a more compact program. The correct formula is $\text{frequency} = 440 * (2^{(\text{octave} + ((n-100)/12)))$, which IS what is shown in Appendix 7 of the SECOND EDITION of the User Instructions, but NOT what Jeremy Vine gives in the April issue of the English "Amstrad CPC464 User", despite his insistence that the User Instructions are wrong! Whilst the coding is shorter, would the introduction of a power cause it to run slower?

Main Variables Used

NB: All variables unless otherwise designated by "\$" are integers (line 3390).

In addition to the variables used in the User Instructions, Chapter 6, the following are used:

a\$,b\$

These accept the key input (i.e INKEY\$) at various points in the program.

n, lim1, lim2, var1, var2

(See notes on lines 320-400)

f,c,a

Additional variables used to match the keys which set the second section of the tone envelope.

key.str\$

The string of acceptable key inputs.

note\$,octave,tone,scaleup,scaledown

Arrays used in the musical note/scale option, whose names indicate use.

x,y,z,zz

Used to designate positions in the arrays and as control variables in the FOR..NEXT loops which load the arrays. "zz" is the value of "z" at the start of a scale.

startnote\$

Contains the name (e.g C #) of tone (zz) at the start of a scale.

```

10 REM *****
20 REM SOUND EXPERIMENT PROGRAMME FOR
   AMSTRAD CPC 464
30 REM   by PETER N CAMPBELL 1985
40 REM *****
50 '
60 REM **** Set Up Routine ****
70 '
80 GOSUB 3380
90 '
100 REM **** Keypress Detection ****
110 '
120 WHILE a$(<)"e"
130   WHILE a$=""
140     a$=INKEY$
150     a$=LOWER$(a$)
160   WEND
170 '
180 REM *** Branch To Subroutines ***
190 '
200 IF a$="z" THEN GOSUB 4000
210 ON INSTR(keystr$,a$) GOSUB 440,570,
   650,730,820,900,990,1070,2110,2190,
   2270,1150,1230,1310,1390,1470,1550,
   1630,1710,1790,1870,1950,2030,3710,
   2350,2430,2510
220 IF a$(<)"e" THEN a$=""
230 WEND
240 '
250 REM **** End Option ****
260 '
270 GOSUB 3240
280 IF a$="" THEN 120
290 KEY DEF 6,1,13 'Resets enter key
300 INK 0,1:INK 1,24:CLS:BORDER 1:END
310 '
320 REM *** Parameter Adjustment ***
330 '
340 IF INKEY(0)=0 THEN n=n+var1
350 IF INKEY(2)=0 THEN n=n-var1
360 IF INKEY(1)=0 THEN n=n+var2
370 IF INKEY(8)=0 THEN n=n-var2
380 IF n<lim1 THEN n=lim2
390 IF n>lim2 THEN n=lim1
400 RETURN
410 '
420 REM **** Sound Command ****
430 '
440 SOUND g,h,i,j,k,l,m
450 '
460 REM **** Tone Envelope ****
470 '
480 ENT -1,t,v,w,f,c,a
490 '
500 REM **** Volume Envelope ****
510 '
520 ENV 1,p1,q1,r1,p2,q2,r2,p3,q3,r3,p4,
   q4,r4
530 RETURN
540 '""
550 REM **** Sound Channel ****
560 '
570 WHILE INKEY(52)=0
580   n=g:lim1=1:lim2=5:var1=1:var2=1
590   GOSUB 340:g=n
600   GOSUB 2590

```

```

610 WEND:RETURN
620 '
630 REM **** Tone Period ****
640 '
650 WHILE INKEY(44)=0
660 n=h:lim1=0:lim2=4095:var1=10:
var2=100
670 GOSUB 340:h=n
680 GOSUB 2590
690 WEND:RETURN
700 '
710 REM **** Duration ****
720 '
730 WHILE INKEY(35)=0
740 n=i:lim1=-32768:lim2=32767:var1=1:
var2=10
750 GOSUB 340:i=n
760 IF i<-2 OR i>500 THEN GOSUB 3100
770 GOSUB 2590
780 WEND:RETURN
790 '
800 REM **** Volume ****
810 '
820 WHILE INKEY(45)=0
830 n=j:lim1=0:lim2=15:var1=1:var2=1
840 GOSUB 340:j=n
850 GOSUB 2590
860 WEND:RETURN
870 '
880 REM *** Volume Envelope Number ***
890 '
900 WHILE INKEY(37)=0
910 n=k:lim1=0:lim2=1:var1=1:var2=1
920 GOSUB 340:k=n
930 IF i=0 AND k=1 THEN GOSUB 3100
940 GOSUB 2590
950 WEND:RETURN
960 '
970 REM **** Tone Envelope Number ****
980 '
990 WHILE INKEY(36)=0
1000 n=l:lim1=0:lim2=1:var1=1:var2=1
1010 GOSUB 340:l=n
1020 GOSUB 2590
1030 WEND:RETURN
1040 '
1050 REM **** Noise ****
1060 '
1070 WHILE INKEY(38)=0
1080 n=m:lim1=0:lim2=31:var1=1:var2=10
1090 GOSUB 340:m=n
1100 GOSUB 2590
1110 WEND:RETURN
1120 '""
1130 REM **** Env Step Count ****
1140 '
1150 WHILE INKEY(10)=0
1160 n=p1:lim1=0:lim2=127:var1=1:
var2=10
1170 GOSUB 340:p1=n
1180 GOSUB 2730
1190 WEND:RETURN
1200 '
1210 REM **** Env Step Size ****
1220 '
1230 WHILE INKEY(11)=0
1240 n=q1:lim1=-128:lim2=127:var1=1:
var2=10
1250 GOSUB 340:q1=n
1260 GOSUB 2730
1270 WEND:RETURN
1280 '
1290 REM **** Env Pause Time ****
1300 '
1310 WHILE INKEY(3)=0
1320 n=r1:lim1=0:lim2=255:var1=1:
var2=10
1330 GOSUB 340:r1=n
1340 GOSUB 2730
1350 WEND:RETURN
1360 '
1370 REM **** Env Step Count ****
1380 '
1390 WHILE INKEY(20)=0
1400 n=p2:lim1=0:lim2=127:var1=1:
var2=10
1410 GOSUB 340:p2=n
1420 GOSUB 2730
1430 WEND:RETURN
1440 '
1450 REM **** Env Step Size ****
1460 '
1470 WHILE INKEY(12)=0
1480 n=q2:lim1=-128:lim2=127:var1=1:
var2=10
1490 GOSUB 340:q2=n
1500 GOSUB 2730
1510 WEND:RETURN
1520 '
1530 REM **** Env Pause Time ****
1540 '
1550 WHILE INKEY(4)=0
1560 n=r2:lim1=0:lim2=255:var1=1:
var2=10
1570 GOSUB 340:r2=n
1580 GOSUB 2730
1590 WEND:RETURN
1600 '""
1610 REM **** Env Step Count ****
1620 '
1630 WHILE INKEY(13)=0
1640 n=p3:lim1=0:lim2=127:var1=1:
var2=10
1650 GOSUB 340:p3=n
1660 GOSUB 2730
1670 WEND:RETURN
1680 '
1690 REM **** Env Step Size ****
1700 '
1710 WHILE INKEY(14)=0
1720 n=q3:lim1=-128:lim2=127:var1=1:
var2=10
1730 GOSUB 340:q3=n
1740 GOSUB 2730
1750 WEND:RETURN
1760 '
1770 REM **** Env Pause Time ****
1780 '
1790 WHILE INKEY(5)=0
1800 n=r3:lim1=0:lim2=255:var1=1:
var2=10

```

```

1810 GOSUB 340:r3=n
1820 GOSUB 2730
1830 WEND:RETURN
1840 '
1850 REM **** Env Step Count ****
1860 '
1870 WHILE INKEY(15)=0
1880 n=p4:lim1=0:lim2=127:var1=1:
var2=10
1890 GOSUB 340:p4=n
1900 GOSUB 2730
1910 WEND:RETURN
1920 '
1930 REM **** Env Step Size ****
1940 '
1950 WHILE INKEY(7)=0
1960 n=q4:lim1=-128:lim2=127:var1=1:
var2=10
1970 GOSUB 340:q4=n
1980 GOSUB 2730
1990 WEND:RETURN
2000 '
2010 REM **** Env Pause Time ****
2020 '
2030 WHILE INKEY(6)=0
2040 n=r4:lim1=0:lim2=255:var1=1:
var2=10
2050 GOSUB 340:r4=n
2060 GOSUB 2730
2070 WEND:RETURN
2080 '
2090 REM **** Ent Step Count ****
2100 '
2110 WHILE INKEY(51)=0
2120 n=t:lim1=0:lim2=239:var1=1:
var2=10
2130 GOSUB 340:t=n
2140 GOSUB 2890
2150 WEND:RETURN
2160 '
2170 REM **** Ent Step Size ****
2180 '
2190 WHILE INKEY(55)=0
2200 n=v:lim1=-127:lim2=127:var1=1:
var2=10
2210 GOSUB 340:v=n
2220 GOSUB 2890
2230 WEND:RETURN
2240 '
2250 REM **** Ent Pause Time ****
2260 '
2270 WHILE INKEY(59)=0
2280 n=w:lim1=0:lim2=255:var1=1:
var2=10
2290 GOSUB 340:w=n
2300 GOSUB 2890
2310 WEND:RETURN
2320 '
2330 REM **** Ent Step Count ****
2340 '
2350 WHILE INKEY(53)=0
2360 n=f:lim1=0:lim2=239:var1=1:
var2=10
2370 GOSUB 340:f=n
2380 GOSUB 2890

```

```

2390 WEND:RETURN
2400 '
2410 REM **** Ent Step Size ****
2420 '
2430 WHILE INKEY(62)=0
2440 n=c:lim1=-127:lim2=127:var1=1:
var2=10
2450 GOSUB 340:c=n
2460 GOSUB 2890
2470 WEND:RETURN
2480 '
2490 REM **** Ent Pause Time ****
2500 '
2510 WHILE INKEY(69)=0
2520 n=a:lim1=0:lim2=255:var1=1:
var2=10
2530 GOSUB 340:a=n
2540 GOSUB 2890
2550 WEND:RETURN
2560 '
2570 REM *** Print Sound Parameters ***
2580 '
2590 LOCATE#1,1,1:PRINT#1," SOUND:-"
2600 PRINT#1," G: Channel ";PRINT#1,
USING "#####";g
2610 PRINT#1," H: Tone ";PRINT#1,
USING "#####";h
2620 PRINT#1," I: Duration ";PRINT#1,
USING "#####";i
2630 PRINT#1," J: Volume ";PRINT#1,
USING "#####";j
2640 PRINT#1," K: Vol env ";PRINT#1,
USING "#####";k
2650 PRINT#1," L: Tone env ";PRINT#1,
USING "#####";l
2660 PRINT#1," M: Noise ";PRINT#1,
USING "#####";m
2670 GOSUB 440:RETURN
2680 '
2690 REM * Print Volume Env Controls *
2700 '
2710 LOCATE#4,2,2:PRINT#4,
"VOLUME ENVELOPE"
2720 LOCATE#4,2,3:PRINT#4,
"Num Keypad controls these parameters"
2730 LOCATE#4,2,4:PRINT#4,"P1:"
USING "#####";p1
2740 LOCATE#4,16,4:PRINT#4,"Q1:"
USING "#####";q1
2750 LOCATE#4,30,4:PRINT#4,"R1:"
USING "#####";r1
2760 LOCATE#4,2,5:PRINT#4,"P2:"
USING "#####";p2
2770 LOCATE#4,16,5:PRINT#4,"Q2:"
USING "#####";q2
2780 LOCATE#4,30,5:PRINT#4,"R2:"
USING "#####";r2
2790 LOCATE#4,2,6:PRINT#4,"P3:"
USING "#####";p3
2800 LOCATE#4,16,6:PRINT#4,"Q3:"
USING "#####";q3
2810 LOCATE#4,30,6:PRINT#4,"R3:"
USING "#####";r3
2820 LOCATE#4,2,7:PRINT#4,"P4:"
USING "#####";p4

```

```

2830 LOCATE#4,16,7:PRINT#4,"Q4:"
      USING "####";q4
2840 LOCATE#4,30,7:PRINT#4,"R4:"
      USING "####";r4
2850 GOSUB 440:RETURN
2860 '""
2870 REM **** Print Ent Parameters ****
2880 '
2890 LOCATE#2,2,2:PRINT#2,
      " TONE ENVELOPE:-"
2900 PRINT#2," T: Step count";:PRINT#2,
      USING "####";t
2910 PRINT#2," V: Step size ";:PRINT#2,
      USING "####";v
2920 PRINT#2," W: Pause time";:PRINT#2,
      USING "####";w
2930 PRINT#2," F: Step count";:PRINT#2,
      USING "####";f
2940 PRINT#2," C: Step size ";:PRINT#2,
      USING "####";c
2950 PRINT#2," A: Pause time";:PRINT#2,
      USING "####";a
2960 GOSUB 440:RETURN
2970 '
2980 REM * Print Keybrd Instructions *
2990 '
3000 LOCATE#3,6,1:PRINT#3,"CONTROLS"
3010 PRINT#3,"< Z > - Scale"
3020 PRINT#3,"<SPACE> - Sound note"
3030 PRINT#3,"< E > - End"
3040 PRINT#3,"< "CHR$(240)+CHR$(128)
      +CHR$(243)" > - Increase
3050 PRINT#3,"< "CHR$(242)+CHR$(128)
      +CHR$(241)" > - Decrease
3060 RETURN
3070 '
3080 REM **** Recovery Routine ****
3090 '
3100 IF i=0 AND k=1 AND p1*r1+p2*r2
      +p3*r3+p4*r4>500 THEN 3110 ELSE
      IF i<0 THEN 3110 ELSE RETURN
3110 CLS#5:PRINT#5,"WARNING!"
3120 BORDER 1,5 'Flashes border
3130 PRINT#5,"Long duration sound"
3140 PRINT#5,
      "Do you wish to continue - y/n?";
3150 b$=INKEY$:IF b$=""THEN 3150
3160 b$=LOWER$(b$)
3170 IF b$="y" THEN 3180 ELSE IF b$="n"
      THEN 3240 ELSE 3150
3180 CLS#5:BORDER 1
3190 GOSUB 3640:GOSUB 2590
3200 RETURN
3210 '""
3220 REM **** End Option Routine ****
3230 '
3240 SOUND 135,0,1,0:CLS#3:LOCATE#3,1,3
3250 PRINT#3,"Do you wish to end";
      "the experiment - y/n?"
3260 b$=INKEY$:IF b$=""THEN 3260
3270 b$=LOWER$(b$)
3280 IF b$="y" THEN 290 ELSE IF b$="n"
      THEN 3290 ELSE 3260
3290 a$="":i=10:SOUND 135,0,1,0
3300 CLS#3:LOCATE#3,1,2:PRINT#3,
      "Duration reset to 10"
3310 PRINT#3,"Press any key";
      "to continue"
3320 IF INKEY$=""THEN 3320
3330 CLS#3:CLS#5:BORDER 8:GOSUB 3590
3340 RETURN
3350 '
3360 REM **** Set Up Basics ****
3370 '
3380 KEY DEF 6,1,93
3390 DEFINT a-z
3400 keystr$=
      " ghijklmtvw7894561230.lzfc"
3410 MODE 1:BORDER 8:INK 0,0:INK 1,3:
      INK 2,13:INK 3,21
3420 g=1:h=200:j=7:i=10:z=28
3430 GOSUB 3870 'Scale option
3440 '
3450 REM **** Set Up Windows ****
3460 '
3470 WINDOW#6,1,10,10,14:PEN#6,0:
      PAPER#6,2:CLS#6
3480 WINDOW#7,32,40,10,14:PEN#7,0:
      PAPER#7,2:CLS#7
3490 WINDOW#0,1,40,1,25:PEN#0,3:
      PAPER#0,0
3500 WINDOW#1,1,20,1,9:PEN#1,3:
      PAPER#1,0:CLS#1
3510 WINDOW#2,21,40,1,9:PEN#2,3:
      PAPER#2,1:CLS#2
3520 WINDOW#4,1,40,15,22:PEN#4,3:
      PAPER#4,1:CLS#4
3530 WINDOW#3,11,31,9,14:PEN#3,0:
      PAPER#3,3:CLS#3
3540 WINDOW#5,5,35,22,25:PEN#5,0:
      PAPER#5,3:CLS#5
3550 '
3560 REM **** Print Menu ****
3570 '
3580 IF a$="e"THEN a$=""
3590 GOSUB 2590:GOSUB 2890:GOSUB 3000:
      GOSUB 2710:GOSUB 3640:GOSUB 3810:
      GOSUB 4000
3600 RETURN
3610 '""
3620 REM **** Print Titles ****
3630 '
3640 LOCATE#5,4,2:PRINT#5,
      "SOUND EXPERIMENT PROGRAM"
3650 LOCATE#5,4,3:PRINT#5,CHR$(164);
3660 LOCATE#5,7,3:PRINT#5,
      "P N Campbell 1985"
3670 RETURN
3680 '
3690 REM **** Scale Option ****
3700 '
3710 WHILE INKEY(71)=0
3720 IF INKEY(0)=0 THEN GOSUB 4050
3730 IF INKEY(2)=0 THEN GOSUB 4150
3740 IF INKEY(1)=0 THEN GOSUB 4250
3750 IF INKEY(8)=0 THEN GOSUB 4390
3760 GOSUB 440:GOSUB 3810
3770 WEND:RETURN
3780 '

```

```

3790 REM * Print Scale Parameters *
3800 '
3810 LOCATE#6,2,2:PRINT#6,"SCALE":
LOCATE#6,2,3:PRINT#6,"Note ";
note$(z);" ":LOCATE#6,2,4:PRINT#6,
"Tone";:LOCATE#6,7,4:IF h<1000 THEN
PRINT#6,USING "###";h
3820 LOCATE#7,2,2:PRINT#7,"PARAS.":
LOCATE#7,2,3:PRINT#7,"Octav";
octave(z);:LOCATE#7,2,4:PRINT#7,
"Start ";startnote$
3830 RETURN
3840 '
3850 REM **** Read Scale Data ****
3860 '
3870 DIM note$(37),octave(37),tone(37),
scaleup(25),scaledown(25)
3880 FOR x=1 TO 37
3890 READ note$(x),octave(x),tone(x)
3900 NEXT
3910 FOR x=1 TO 25
3920 READ scaleup(x),scaledown(x)
3930 NEXT
3940 DATA C,-1,956,C#,-1,902,D,-1,851,
D#,-1,804,E,-1,758,F,-1,716,F#,-1,
676,G,-1,638,G#,-1,602,A,-1,568,A#,
-1,536,B,-1,506,C,0,478,C#,0,451,D,
0,426,D#,0,402,E,0,379,F,0,358,F#,
0,338,G,0,319,G#,0,301,A,0,284
3950 DATA A#,0,268,B,0,253,C,1,239,C#,1,
225,D,1,213,D#,1,201,E,1,190,F,1,
179,F#,1,169,G,1,159,G#,1,150,A,1,
142,A#,1,134,B,1,127,C,2,119
3960 DATA 2,2,2,2,1,2,2,1,2,2,2,2,1,1,2,
2,2,2,1,2,2,1,2,2,2,2,1,1,2,2,2,2,1,
2,2,1,2,2,2,2,1,1,2,2,2,2,2,2,1,1
3970 '""
3980 REM **** Clear Scale Option ****
3990 '
4000 LOCATE#6,8,3:PRINT#6," "":
LOCATE#6,7,4:PRINT#6," "":
LOCATE#7,8,4:PRINT#7," "":
LOCATE#7,7,3:PRINT#7," " "
4010 RETURN
4020 '
4030 **** REM Play Notes Up ****
4040 '
4050 WHILE INKEY(0)<>-1
4060 startnote$=" "
4070 z=z+1
4080 IF z>37 THEN z=1
4090 h=tone(z)
4100 GOSUB 3810:GOSUB 2590
4110 WEND:RETURN
4120 '
4130 REM **** Play Notes Down ****
4140 '
4150 WHILE INKEY(2)<>-1
4160 startnote$=" "
4170 z=z-1
4180 IF z<1 THEN z=37
4190 h=tone(z)
4200 GOSUB 3810:GOSUB 2590
4210 WEND:RETURN
4220 '

```

```

4230 REM *** Play Scale Up Routine ***
4240 '
4250 IF z>24 THEN z=z-24
4260 IF z>12 THEN z=z-12
4270 zz=z:start=tone(z):startnote$=
note$(zz):y=1
4280 WHILE tone(z)>start/4.1 AND
INKEY(1)=0
4290 h=tone(z)
4300 GOSUB 3810:GOSUB 2590
4310 z=z+scaleup(y)
4320 IF z>37 THEN z=zz
4330 y=y+1:IF y>25 THEN y=1
4340 WEND
4350 z=zz:IF INKEY(1)=-1 THEN RETURN
ELSE 4250
4360 '""
4370 REM ** Play Scale Down Routine **
4380 '
4390 IF z<12 THEN z=z+24
4400 IF z<24 THEN z=z+12
4410 zz=z:start=tone(zz):startnote$=
note$(zz):y=25
4420 WHILE tone(z)<start*4.1 AND
INKEY(8)=0
4430 h=tone(z)
4440 GOSUB 3810:GOSUB 2590
4450 z=z-scaledown(y)
4460 IF z<1 THEN z=zz
4470 y=y-1:IF y<1 THEN y=25
4480 WEND
4490 z=zz:IF INKEY(1)=-1 THEN RETURN
ELSE 4390

```



Discounted Books for subscribers of The Amstrad User

The prices of titles below have been reduced by 10% for the benefit of subscribers. We are constantly reviewing the availability of books suitable for the CPC464, and only offer those which we consider worthwhile to users.

The Amstrad Games Book

A collection of exciting games for your Amstrad, including arcade-style space and combat games, maze games, impossible and dangerous missions for superheroes and many more **\$13.45**

Amstrad Computing

Takes you step by step, in detail, through the commands on your CPC464. You will find how business calculations are worked out, how vivid displays are obtained and how attention-catching sounds can be arranged. **\$14.35**

Basic Programming on the Amstrad

The first two chapters are for the complete beginner, the rest introduces the majority of the BASIC commands available, and the last few deal with topics of more specialised interest such as the use of sound, and how to create files and read data from them. **\$17.95**

The Working Amstrad

This book is a collection of solid application programs. The areas covered include home finance and tax, information storage and retrieval, household and diary management, creative graphics, and effective display techniques. **\$17.95**

40 Educational Games for the Amstrad CPC464.

The programs in this book have been designed to help the younger members of a family to handle the Amstrad and to increase their general knowledge – whilst enjoying themselves. Subjects include languages, mathematics and science **\$14.35**

Machine Code for Beginners on The Amstrad

As the title suggests, this book is intended for the beginner wishing to learn how to use Machine code on the CPC464. Extensive use is made of the machine operating system allowing results from programs to be seen immediately **\$16.15**

The Amstrad CPC464 Advanced User Guide

This book assumes you have got your CPC464 working and have already done some simple programming. It provides explanations on how the CPC464 communicates with external devices, strings and characters, data structures, Graphics and Sound. **\$15.65**

60 Programs for your Amstrad CPC464

A massive software library for the price of a single cassette. Explosive games, dynamic graphics and invaluable utilities, this specially commissioned collection takes BASIC to the limits and beyond **\$17.10**

An Amstrad CPC464 Compendium

A fascinating selection of thirty games, each presented with a complete listing in LOCOMOTIVE BASIC, a helpful introductory description and a screenshot. There is something to appeal to the traditional and ultramodern computer gamers alike **\$20.65**

The CP/M Handbook with MP/M

If your future in computing goes beyond playing games – you’d better have a copy of this book on your shelf. It will satisfy the requirements of first time users as well as veteran CP/M experts. **\$26.95**

CP/M User Guide.

This updated best-seller covers everything you need to know about the CP/M operating system from Digital Research. Organised for both beginner and advanced users, the information you’ll need most often is presented first, followed by more in-depth technical data. **\$24.45**

Logo Programming

This entertaining book will give you the knowledge and experience required to make the most of LOGO. Learn to use this ingenious medium and soon you will be constructing your own micro-worlds, simulations and demonstrations **\$11.65**

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, Shop 2,
33 The Centreway, Blackburn Road,
Mount Waverley, Victoria, 3149*

Bankcard or Mastercard orders accepted by phone on (03) 233 9227

Review of Ghostbusters

Following the release of the film and the record, it was inevitable that a game would be produced to cash in on the success of the first two. But does the game live up to expectations? Simon Anthony provides a spirited review.

The idea behind the game is based, somewhat loosely, on the film of the same name, and the aim is to successfully reach the top of the Temple of Zuul by sneaking two Ghostbusters into the entrance when the Gatekeeper and the Keymaster have joined forces. But if you haven't seen the film these characters may mean nothing to you, which might take the edge off the game a little.

However, let me explain the format. First, it is necessary to set up a Ghostbusters franchise. This is achieved by opening a \$10,000 start-up account with which can be purchased any one of four cars, each capable of travelling at a predetermined speed and of carrying a predetermined quantity of Ghostbusting equipment. These include a PK Energy Detector, an Image Intensifier, a Marshmallow Sensor, Ghost Bait, Traps and so on.

Having made your selection, the program moves to the MAP screen, a representation of the city roads and buildings with, of course, Zuul's Temple and your GHQ. With the aid of a Joystick, your car is guided to buildings with 'Slimers'. As you go, passing Roamers should be vacuumed to keep them from getting to the Temple. Any missed add 100 to the city's PK energy reading.

The next screen, the building with the Slimer, gives you an opportunity to earn some money in trapping the Slimer by manipulating two Ghostbusters with their negative ionizer backpacks. Earning money enables you to re-equip.

The 'nasty' in this game is the monolith of marshmallow, a dumpy

guy with a sweet tooth, who if not immediately lured by some bait will flatten any buildings in its path.

As I mentioned before, unless you have seen the film, it may take a while to come to grips with the game. You will need to study the instructions carefully, and while doing so, can listen to the Ghostbusters theme music which plays (including a 'bouncing ball' over the lyrics) until you are ready



to start. The music is quite impressive through stereo speakers and continues to play during the game. (Thank heavens for the volume control).

The graphics are cleverly structured, but I found the 'car on the streets' screen a bit superfluous, having already driven from A to B on the street map only to find you have to do it again (albeit with a bird's eye view of the car) just to suck up the Roamers.

You need to persevere to create sufficient funds to re-equip your car and to ensure that you have more money at the end of the game than

you started with — you lose if you haven't. This is where I found Ghostbusters to be slow and is probably my main objection. In fairness, it cannot be classified an arcade game, the sort to get the adrenaline running, it is more a strategy/business/arcade combination which I believe doesn't quite work. It's an interesting concept though.

To summarize, by using THE AMSTRAD USER weighting method (Issue 1, Page 8), Ghostbusters scores 38 points out of a possible 56 points. This is broken down as follows, with each element carrying a maximum of 8 points:

Ease of Use	6
Speed	4
Entertainment value	5
Documentation	5
Originality	7
Use of Graphics	7
Ability to hold interest	4

Overall, I was a little disappointed with the game, but I suppose this may be because I didn't think much of the film either.

One 'el of a problem!

It has been pointed out, much to our embarrassment, that the 1's and l's printed in the April edition of The Amstrad User, particularly the Computerised Address Book, are identical. Well, er... yes they do, but in most cases it is fairly simple to decide which one it should be. But, just in case you can't, we will put the problem right in next month's issue.

A Plug for the Wireless Institute

We couldn't resist the heading, but are grateful to Fred Robertson-Mudie for providing this article. It will be of particular interest to Amateur Radio enthusiasts of whom there appear to be a number of CPC464 users.

Amateur Radio is a fascinating hobby which can be, and often is, an area where computer buffs can be involved. Amateurs use computers for various aspects of the hobby, including RTTY, AMTOR, CW, SSTV, FAX, Packet Radio and even for controlling their antennas when working Amateur satellites. If any readers would like further details of Amateur Radio they can write to: The Federal Secretary, Wireless Institute of Australia, PO Box 300, South Caulfield, Vic 3162.

I obtained, via a third party, suitable substitution formulae for ARCSIN and ARCCOSINE for the CPC464. These functions can be very useful and are as follows:

```
10 REM *****
20 REM * BEAM HEADINGS *
30 REM *****
40 REM * de VK1MM *
50 REM *****
60 REM A=your lat.* L1=your lon.
70 REM C=bearing angle
80 REM D=degrees of arc
90 REM E=intermediate value
100 REM K=conversion constant:arc to km
110 REM L=difference in lon.
120 REM B=dx lat.* L2=dx lon.
130 REM M=conversion constant:degrees to
radians
140 REM N=conversion constant:arc to nm
150 REM S=conversion constant:arc to sta
tute miles
160 REM R=reciprocal bearing
170 CLS
180 D=1:K=111.12:M=57.2957795:N=60:S=69.
06
190 PRINT"GREAT CIRCLE DISTANCE AND BEAR
ING"
200 LOCATE 1,4:PRINT"Enter values in who
le degrees and decimals. Use -ve prefix
for south latitudes and east longitudes"
210 IF D<>1 THEN 280
220 A=-35.3:A=A/M
230 L1=-149.133
240 LOCATE 1,8:PRINT"Enter dx qth"
250 SOUND 1,75,10,10
260 INPUT a$
270 SOUND 1,75,10,10
280 INPUT "LATITUDE";B:B=B/M
290 SOUND 1,75,10,10
300 INPUT "LONGITUDE";L2
```

$$\text{ASN}(X) = \text{ATN}(X/\text{SQR}(1-X*X))$$

$$\text{ACS}(X) = 90-\text{ATN}(X/\text{SQR}(1-X*X))$$

and are certainly worth a mention in your magazine.

Next, I am sending you a short program for calculating Great Circle bearings and distances which, as you will appreciate, is primarily of interest to Amateur Radio Operators. I have modified the program so that it is suitable for the CPC464. I should point out that the Amstrad is my first computer and that I have no previous experience so the methodology followed by me may well leave a lot to be desired. In any event the program works, which is the main thing.

```
310 SOUND 1,75,10,10
320 L=(L1-L2)/M
330 E=SIN(A)*SIN(B)+COS(A)*COS(B)*COS(L)
340 D=-ATN(E/SQR(1-E*E))+1.57079
350 C=(SIN(B)-SIN(A)*E)/(COS(A)*SIN(D))
360 IF C>1 THEN C=0:GOTO 380 ELSE IF C<=
-1 THEN C=180/M:GOTO 380
370 C=-ATN(C/SQR(1-C*C))+1.57079
380 C=INT(C*M)
390 IF SIN(L)<0 THEN C=360-C
400 R=180+C
410 IF R>=360 THEN R=R-360
420 R=INT(R)
430 CLS
440 LOCATE 4,1:PRINT"GREAT CIRCLE BEARIN
G AND DISTANCE"
450 LOCATE 8,8:PRINT "DX QTH = "a$
460 LOCATE 8,10:PRINT"Bearing "C" degree
s(short path)"
470 LOCATE 8,12:PRINT" "R" degree
s(long path)
480 LOCATE 8,14:PRINT"Distance" INT(N*D*
M)"NM"
490 LOCATE 8,16:PRINT" "INT(S*D*M
)"Miles"
500 LOCATE 8,18:PRINT" "INT(K*D*M
)"Km"
510 LOCATE 4,22:PRINT"Press any key to c
ontinue..."
520 SOUND 1,75,10,10
530 IF INKEY$="" THEN 530
540 CLS
550 GOTO 170
560 END
570 REM ** Apart from text, there are no
lower case L's in this program **
```

• COMPETITION •

\$2500 worth of prizes to be won over four classes

Class 1
Best overall program
Wins an AWA Video Recorder

Class 2
Best amusement/adventure
Wins a new DD1 disk drive

Class 3
Best educational software
Wins a new DD1 disk drive

Class 4
Best business software
Wins a new DD1 disk drive

How to enter

Think about your program and map it out in a series of events or features. Write the program onto cassette based around these events and check that the program runs as intended. Once you are satisfied, send a copy of the cassette in a suitable envelope along with the following:

- 1 A brief summary of the program in 500 words or less.
- 2 A clear program listing if available.
- 3 A stamped, self addressed envelope of adequate dimensions if you would like your entry returned.
- 4 Your name and address.

You may make as many submissions as you want, but no entrant may win more than one prize.

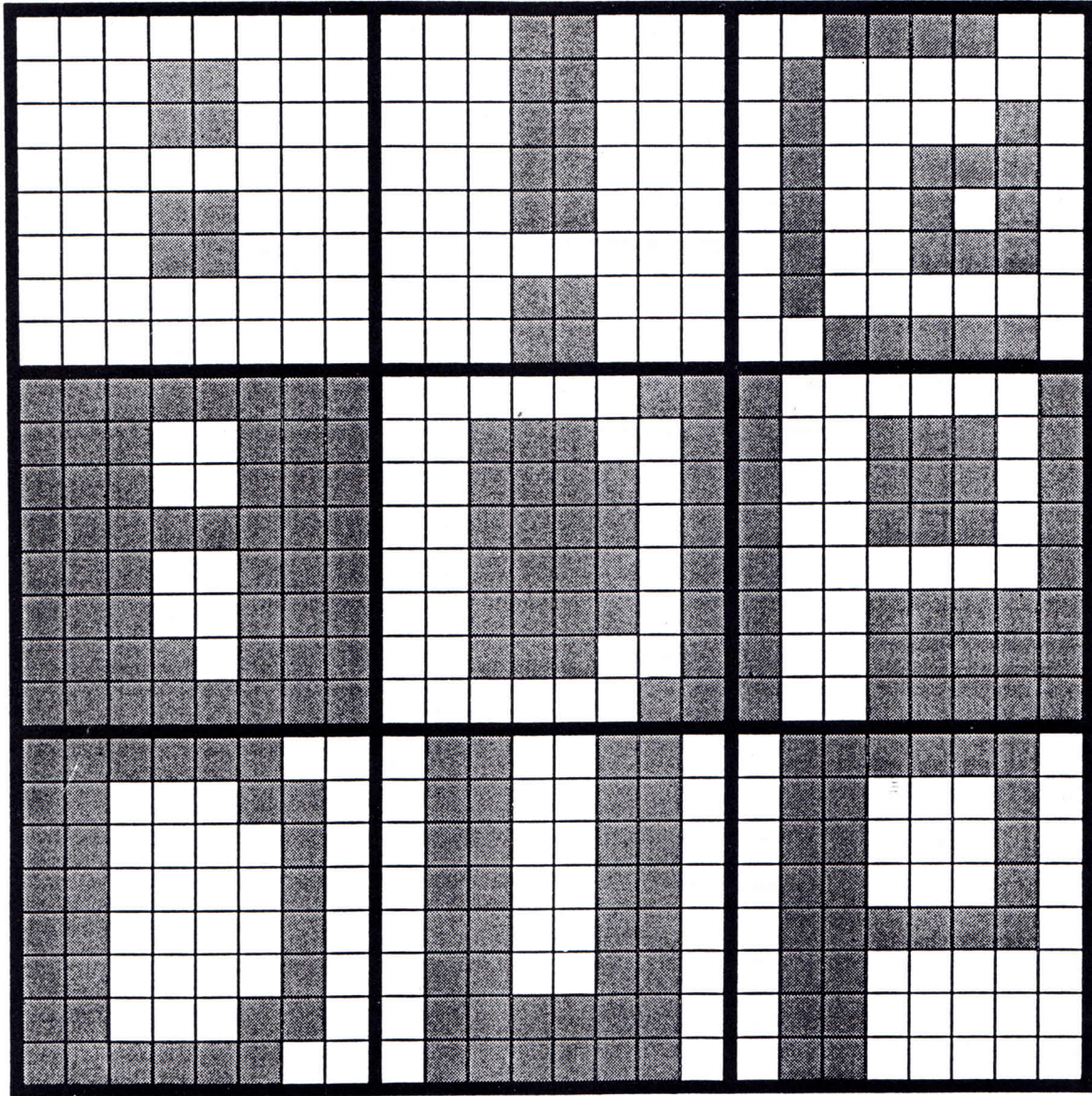
Conditions of Entry

- 1 All entries must run on a CPC464, and must include a cassette copy of the program (plus loading instructions where necessary), a brief summary of the program and its purpose and, if possible, a full listing.
- 2 All entries must arrive by 15th August 1985, and winners will be printed in the October edition of The Amstrad User.
- 3 The decision of the judges is final.
- 4 It is a condition of entry that all entrants have exclusive ownership of the copyright of the material submitted, and the winners agree to assign all copyright in the winning submissions to The Amstrad User. Where the entrant is more than one individual, then one person must be nominated and empowered to act on behalf

of the entire group. All entrants must undertake not to submit the same or a similar program to any other magazine, publisher or organisation until after the announcement of the winning entries.

- 5 We, The Amstrad User, may offer to publish programs other than the winners in the magazine or as commercial software, in which case we will agree terms on an individual basis with the author(s) concerned. We reserve the right to amend, alter or revise any program we publish.
- 6 No employees of The Amstrad User or Strategy Publications, or their relatives may enter this competition.
- 7 The Amstrad User cannot be held responsible for any loss or damage to any submission.
- 8 No entrant may win more than one prize.

ams-Forth 1.1



for the **AMSTRAD**

Enquiries to: Strategy Software
Shop 2, 33 The Centreway, Blackburn Road,
Mt. Waverley, Vic. 3149
Telephone: (03) 233 9227

SAVE \$450

Q.

How many computer magazines would I have to buy to get at least 30 pages of information and program listings for the new Amstrad CPC464?

A.

On average, most popular magazines will devote 2½ pages to the AMSTRAD CPC464. This means you will need to buy 12 magazines at a cost of around \$40 per month, or **\$480 per year!**

Q.

Surely there must be a more sensible and cheaper way of getting the information I need?

A.

There is.
THE AMSTRAD USER is a brand new monthly publication packed with articles, reviews, listings, hints etc. for Amstrad users only, and costs just **\$30 a year.**

Q.

How can I get a copy delivered to my home each month with an optional cassette containing all the program listings?

A.

By completing and returning this order.

Please send me THE AMSTRAD USER for 12 months

Magazine only: \$30 Magazine and cassette: \$70

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 4a, 33-45 The Centreway
Blackburn Road, Mt. Waverley, Vic 3149 Tel 03-233 9227

(OVERSEAS PRICES ON APPLICATION TO ABOVE
ADDRESS)