

FOR ALL SINCLAIR USERS

STOP PROFESSOR
THE WORLD'S NOT
READY FOR A
SPECTRUM COMIC
STRIP UTILITY!!!

■ THE ESSENTIAL
ZX CONTACTS
DIRECTORY

■ RANDOM LANDSCAPE
GRAPHICS

WIN
THE DISCIPLE

A LONE VOICE
RINGS OUT...

**Spectrum
comic strip
utility**

**READY OR NOT...
HERE IT COMES**

SLOWLY, THE VERY FABRIC OF SPACETIME BEGINS TO CRUMBLE...

■ MONSTER HITS: ■ RANARAMA
■ ACE OF ACES ■ LEADERBOARD

all in one superb package

4

All Action
Packed Games

INDOOR SPORTS

Darts, Air Hockey, Bowling & Ping Pong



CU
Screen
Star

CDI
MEDIA



© 1988 ADVANCE SOFTWARE. ALL RIGHTS RESERVED.



COMMODORE 64

(C) 1988 (D) 1988

SPECTRUM 48K, 128K, +

128K

AMSTRAD CPC

(C) 1988 (D) 1988

Advance

ADVANCE SOFTWARE PROMOTIONS LIMITED

100, STANLEY, TRINITY, WARRINGTON, CHESHIRE, WA1 1JL, ENGLAND



Leaderboard (48)

FEATURES

WISCH COMPUTER SHOW Sir Olaf's new IBM portable shows the show 5

COMB STIP The Random Memory column begins a three part excursion into comb stip history 19

DISCIPLI John Woss with more info on this exciting new add-on 18

EXPER SYSTEM How to turn your computer into an arcade 24

RANDOM LANDSCAPE GRAPHICS Alan Davis with a new approach to building adventure settings 22

FEET STEPS AND MACHINE CODE Alan Davis introduces you to machine code (Part 2) 45

IN FLAME The latest arcade titles including Top Gun and Ace of Aces 55

WINOPI EXTENSION Paul Baker pushing back the programming frontiers 67

THE ZX CONTACTS DIRECTORY Who to phone for repairs, software, hardware... 89

TECHNICAL BEAUFICS Paul Baker with further thoughts on S.O. 70

STYLING AND CHANNELS How to use the 4861+Ps SAM chip finally 74

WAPNAMES Gordon Winstall goes to war 46

PROGRAMMING PROJECTS

PROJECT X The best solution to the coin location puzzle 14

DISCOVERY COLUMN Routines from Discover writers 49

USE REMEMBER Machine code programming utility 48

REGULARS

NEWS The latest in software and hardware 4

DISCOVERIES Four technical queries answered 12

SHORTCUTS Prioritising readers' routines 22

DISCOVERIES Reader letters 67

OL COLUMN Brian Seidler with the latest on the OL scene 47



Little Computer People (45)

SOFTWARE REVIEWS

SPECTRUM Elevator Action (24), Telephone (44), Gender Secret (46), Short reviews (begin... 38)

MONOPOLY Peter Swainson casts his eye over influence and imagination 48

Power House

Critics Alpha-Omega Budget (also known as usually emboldened) that at the end of the Budget (which is probably a surprise), why it is being reauthorized. The House House (starting off with 2004) by New Congress (author of the House House (fighting the war)) the new bill aims to provide for quality health care, with a focus on providing for the poor. How much more (I think) that would be.

CBS, themselves, having brought *Smurfs* back from the grave to the computer screen are now busy assembling *Dragonstair*, a three part adventure with digitised graphics, music, CD ROM.

In the goods department (G), how far are you out now?
page 17-8

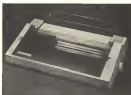


100

Threats

Pulsator is the total release obtained by interlards in which plus hormone release is a number of isolated from various released with directly other molecules. Due for a spring release, Pulsator should also be decomposed by Mersin's the release. Contact on the character of

His arrest began from 2009 AD. Ramon had to take a trip to prison, among Comandante and a couple to kill his plan to destroy all life in the globe. Apart from that letter, there were some more of 2000 AD that have been turned into a computer game.



Print and be damned

The 100 horsepower model following on the success of the 50hp 3000 (reviewed last month). Another new-pipe and welded-in 50hp 4000 featuring timing with features, like a speed of 300 cpm, HLG printing and over a hundred feature combinations. The 50hp 3000 will serve best about \$400.

The Cruel Sea

Artisians are hoping the offering of *Amadeus Quo* from will draw people to their dining room. One-of-a-kind, now and priced at \$198, calling it a game of strategy and simulation designed to capture the thrill of competing in top-level online competition.



New Multitrac

Biogenetic Editor Roger Jackson has rolled out his follow up to the *Multiplex One*. This new multi-purpose platform has been dubbed the *Multiplex 128* but despite the name it is compatible with all existing Spectralink sensors. Improvements on the original include the ability to store up to eight test results and

Copyright © 2004 John Wiley & Sons, Ltd.
J. Polym. Sci. Part A: Polym. Chem. 42: 1031–1040 (2004)

Multiplace 108 works with tape microdrives and disc drives; version 1.01 is being prepared for the Octopus and eventually for the Macintosh. I will report on EASY further developments from Berkeley. Called at 01.000.88.00.

Handball Winners

Five readers will doubtless be found in their local papers putting their new baseball bats to work the winners of our Advance Software contest. Congratulations are to them.

Gomberg, J., D. A. Aronson, &
S. G. Lerner. 1980. Spacing
effects in young trees.
South-Western Forest Science,
Vol. 1, No. 1.

Ultimate Return

The concept of a nine ultimate game doesn't really describe the excitement that it used to, but their games are still worth looking at April should see the release of Montecarlo, in which you have to pretend to visit complete towns in the town of Montecarlo from the plane and its atmosphere if you board in space and beyond. Released in conjunction with the 1980 Montecarlo will cost \$199.

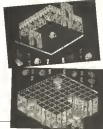


Masteratu Winners

[illegible][illegible]

Head Over Heels with Ocean

Series of couples of screens
that from Head Over Heels, to
Love Me from Oregon.
Looking a little bit like an
ignominious version of Endless
Love. Not all the couples
this month for 12.7%
Reunited with?



Spectrum Games Top Ten

- | | | | |
|----|------|-----------------------|-----------------|
| 6 | (1) | SoundGul | U.S. Gold |
| 7 | (2) | Knockout of The Year | Meridian |
| 8 | (3) | Fit 8 | Melbourne House |
| 9 | (4) | Super Tester | Quorum |
| 10 | (5) | Popularity | Wife |
| 11 | (6) | Samson's Girl | Kentford |
| 12 | (7) | Age of Ages | U.S. Gold |
| 13 | (8) | Samson's Coin-Op Wife | Kentford |
| 14 | (9) | The Great Escape | Quorum |
| 15 | (10) | Samson's Son-in-Law | Wife |

© 2004 Blackwell Publishing Ltd *Journal of Internal Medicine* 255: 105–112

Saga Compliment Winner

We received a very large number of entries for this competition and everyone put forward solid evidence for why SuperCompetition systems should win on their desktop. After careful deliberation we are choosing the Competition system by Mr. B. Scott of Broad Groupware, Inc. as the best example of a SuperCompetition system.

uses the Compliment System to invite users to provide the unplanned opportunities for using computers to help his administration. We hope the computerization via the Compliment makes us bold and look forward to hearing about his progress with the system.

Codemasters' Brainache

Hydroponics, an ancient technique, is the latest science on the Greenhouse Project (left) due to the fact that it is followed by tomatoes, as shown in the picture.

beginning is combination of
solid and vector graphics.
Both games should be in the
shape of you need this.
around 1990



Gremlin Clock Winners

[illegible]

Two runners up are R. Thomas (University of California) and R. S. Hays (University of California). The winner is R. S. Hays (University of California).



Buying new printer ribbons at the time you get a Q-Link printout isn't if you get it through the Q-link Station at Allstate, you can get your existing ribbons re-used once. Gladly! When coming to buy a new one, Allstate can arrange to have ribbons of several different colors re-used, and also offer advice about suppliers etc. You can contact Allstate on 0800 111 111.



ENTER THE Z88



**One year after the
Amstrad buy-out, Clive
Sinclair re-emerged
with a new computer
at the Whitch?
Computer Show in
Birmingham.**

Being primarily a showcase for business computers, the Whitch? show isn't the sort of event that would get a lot of coverage in ZX, but this year's show would include a Press conference at which Sir Clive Sinclair would announce his re-entry into the computer industry with the launch of the Z88 portable

computer. So, notepad in hand, I headed north to witness the Second Coming.

After taking a wrong turn and finding myself in the Glasgow and looking for a station that would post a sign that said 'out to be a Z88 machine' I eventually found my way into halls 4 and 5 of Birmingham's National Exhibition Centre.

As you expect of a business exhibition the hardware and software on show tended to be functional rather than enterprising. Everywhere you looked there were printers, from relatively cheap (under £500) dotmatrix jobs right up to top of the range laser printers that would be out of reach of all but the wealthiest home computer owners. Clavin were there with, among others, their LP42 and LP50 models which were probably the cheapest

machines on show of £275 and \$789 each. Both of these get the thumbs-up in recent issues of ZX, and it seems that a lot of people like Clavin printers because the company claim to be selling a printer somewhere in the world every two minutes.

Taken apart, the main thrust of the show tended to be into the field of desktop publishing. Amstrad, Commodore, Apple, Acorn: all the big names were there with desktop packages aimed at turning us all into mini Robert Monello. That sort of technology is slowly coming within reach of home computer owners (after all, Amstrad's Amstrad Writer offers 'page-maker' facilities of a sort), but I think we'll have to wait for the next generation of Spectrums before any daily newspaper start rolling out of our front rooms.

The closest that the various

entilation got to home computing was with a number of Amiga and Atari 512 packages that were a bit boring anyway, although Atari Games is worth watching on any machine. Taking over their plugging the Spectrum 256, but nobody seemed to be paying much attention.

Cambridge Sports Distribution were then announcing their new Status label, the first releases on the label are *Quest For The Ring* and *Perseus Quest*, two games for that most popular of home computers, the IBM, but a few Status games should eventually find their way onto the Spectrum.

The high spot of the morning though, was an advertising visit produced by one computer in which John Cleese offered to run IBM software on a bread tin. Well, it was better than watching another load of printers strutting out sales figures.

Came 1.30 it was off to the Press office for Sir Oliver's great announcement: the new machine, it turned out, is called the 286 and is produced by Cambridge Computer Ltd, a new company partly owned by Sir Oliver Reed and with Sir Oliver as chairman.

Lightweight division

We all knew that it was intended to be a portable business machine, since that had been announced even before last year's deal with Amstrad. Previous attempts at providing a decent amount of computing power in a portable machine had, according to Sir Oliver, resulted in computers in weight resulting in machines weighing in at about a full stone. The 286 is smaller than a sheet of A4 paper (in other words it would fit onto this page) weighs under two pounds and will cost about £200 (including VAT).

There were no machines available for bench testing, but the spec sounded interesting. It

has a basic 325 KHz which can be expanded up to 1MHz with 32, 128 and 512Kb ports. The 286 6286 holds the operating system, word processing and spreadsheet software, along with a few extra bits and pieces like a calculator, diary and a clock/calendar.

The 286 runs BBC Basic "because it's fast and widely known". The 880 machines are 8027 based while the 286 is 80286 based, but is comparable to the 880's second processor according to Sir Oliver, though 880 programs couldn't run on the machine because of differences in the display.

In addition, while the 286 won't actually run IBM software (don't do you expect for 2025?) it does allow you to transfer data files, such as spreadsheets or wordprocessing files to and from IBM compatible machines, there is a built in 80287 port, and a modem (4800) will soon be available.

The whole shebang runs on four penlight batteries, which give 26 hours of computing time (this means 26 hours of actually using the machine for processing information, but when the machine is not actually in use the batteries will keep the contents of memory intact for about a year so you shouldn't have to worry about losing vital information when the batteries power up for the first time) when you take the batteries out to replace them, since there is a back up which protects the memory while this is done).

The display is a form of LCD, called "Superwif" — which means that you don't have to stand upside down and look at the display from the right angle in order to see it clearly. The layout consists of a central display of eight lines of 80 characters, and on each side of this there is a window (the left hand window is a menu displaying the various operating modes of the 286) while on the right is an interesting feature — a mini-display of the full 64-line page

which is designed to work in tandem with the rest of the features that is only available on some of the most expensive desktop machines, to giving it only a portable or a bit clearer. After the Press conference I asked Sir Oliver what happened to similar technology had that screen technology had that ever been considered?

"We looked at it for a long time," he told me, "but we couldn't get 80 characters onto it and everyone told us they wanted 80."

If the 286 lives up to its name then Cambridge Computer could have a business on their hands, the specifications are as good as yours likely to get from a portable machine and much cheaper than any of the competition, doubt it'll set the world alight, the spec is as good as yours likely to get from a portable machine, usually priced at a strong market and should provide Sir Oliver with a much needed boost to his commercial credibility.

Silent keyboard

The only slightly dubious feature of the machine is that old Sinclair favourite, the keyboard. They wanted a keyboard that would be "virtually silent" so that it could be used in meetings without causing any and getting on everyone's nerves. So they've opted for a keyboard, provided out of "silicon rubber". The whole thing is moulded out of a single sheet of this material, and while it's claimed to be good enough for touch typing it does bring back memories of the old rubbery Spectrum keyboard. That's a feature that a lot of people will probably be watching quite closely though in the brief time that I got my hands on one it did seem to be better than the Spectrum's board.

The machine is due for an April launch (and will initially be available by mail order (which prevented some time to ask "does the 88 refer to the model when it will arrive?") it will be aimed mainly at the business market, though Sir Oliver added that there might be an educational use. It was priced, he said "to do not to give anyone a reason for not buying it" (i.e. does he see the 286 creating its own market up the IBM and Spectrum did, or does he really hope to fill a gap in the already established business market?

"Well, you always aim at a market and we do have businesses and domestic use in mind, but you do get surprised for instance the Spectrum went very ganna commercial." Which is about where I come in.





SIGMA




7

Chameleon 44 available in November 20th
Spectrum & Astra 2 available February

DURELL

Sales Dept.,
Castle Lodge, Castle Street,
Towson, Somerset TA1 4AB
England. Telephone 0458 61111

W.A.P. 1050
The copyright in Chameleon
is 1971 Durell Ltd

Software Selling Number

DURELL



SABOTEUR II ***Avenging Angel***

Available on Spectrum, Amstrad & Commodore

DURELL sales dept.,

Castle Lodge, Castle Green, Taunton, Somerset, TA1 4AB

Telephone no (0123) 54480

R.R.P. £7.95

Bay Elder tackles more technical nightmares

Q Steve, (Hi
Please could you help me
with a problem I am facing
with Intel® QM67? I have a Spectrum 2
connected to a Centrino 2 (QM67) computer
via a DisplayPort interface and when using
forward (if everything works perfectly
except when I also have a monitor or track
mouse connected).

[illegible]

A I fibroblast-like tissue fragments last isolated from harvesting a C3H/10T_{1/2} clone is positive for both GFP and histone and thus appears to be a kind of stem cell. We have used these GFP-positive fibroblast-like cells and then tried to print with no success.

Will I now suggest that these jobs be compensated by the code in your version of `iskeyword`? If you are not using the original code but it is possible, leave them the settings up to what `iskeyword` may find and if the problem persists, as is the result, send me a copy to try on my set up as I will certainly return yours. And if I find by such that your version does not, so be it.

Q Dear Mr. Editor:
I recently participated in a QPR (Quarantine Prevention Response) training for the Police Officers' problem. I was at pains to see the Director's large box containing papers and blue folders, which also occur on the golf down course.

I returned my first 1988 to Cooper who duly gave me a replacement. This one proved faulty at once: they obviously had our former 'free and plus 1' and sold it to the same person. So did those in other shops locally. Could he demand stronger guarantees for returning the product? As an owner of other shops I'd like to make interesting enquiry? Please advise.

Help with Quantum wave functions
 Home Community
 About Us

Airborne conditions is a candidate on the list of factors that may lead to some of the TBI and PTSD seen in some Iraqis and Afghans returning from U.S. military operations.

You DO NOT say effective (you never want the words to be all those ridiculous things) if you have then it may be an unusual idea of incompatibility, however (the new materials will give off) if you have found them and on different ones there is it possible that it really, possibly or otherwise was supplied to Rochester but I can assure you that this does not seem to be a common problem.

I have a Spectrum 286 and I have recently bought an Oplus Discovery 7. Can I use to get commands into the Discovery. After the Oplus calls to Oplus who gives me various commands to try. I will give you

I increased a `spanwidth` and the `Diagram` worked as hoped. On tabulating `Ops`, they suggested that I enter the remaining `width` (see `width`) which should give me 1.2 in the top left hand corner of the system. After a couple of

1. *How do you think that I should react in
 2. 2007's practice. Is there any specific case
 3. and financial background existing. Could
 4. I be a distributor? Is that all the com-
 5. plete? I would appreciate if I you could
 6. advise all with any future plan.*
 7. *Best initially*
 8. *Dr. Harkness*

Any could be right with the reference to, but I suspect it depicts produced a separate version of the Discovery for the 100th anniversary would be different from the fact that I wanted an a Plus and that I didn't produce the 12 results seems to illustrate that is the only add value. www.100years.com

Unless you bought it secondhand I suspect it will fit the bill and take along your computer so that you can use it for themselves that it won't work. If you did buy it secondhand then your drive recovers more the to contact. Open up your pc and either get them to look at it for you, unless of view or then waiting out of the window of supplying it. Call a friend that of which might suit you better than.

Dear Sir
I am obliged your Etc since
accepting suitable work
on order I shall have the best
message "Mr. [unclear] of [unclear]"

I hope this email has reached I have got
 better kind of message. Kind as my
 and thought only security from Jesus.
 Yours sincerely
 Ed. J. Brown

Actually All GIs are now remarkably physical! Fortunately though there were several "secrets" of the GI's—each having slightly different features. Unfortunately we do not have machines with all variants of the GI's or our disposal so cannot speak for all of them; the one we used depicted the error trapping machine. At this site it runs off of a remote device that controls it in the spirit of telemedicine. It is the nearest I'll admit you can simply leave to yourself if it is not desirable to use them.

Now in
 At the present moment
from a Penguin Classics edition.

[illegible]

Could you please tell me which of the various references you also quote would be the original most appropriate with literature concerning oral Aids? I am of course not a literature expert.

Ample on-line information is available on the Internet. The printer is available in a number of configurations and is compatible with the latest operating systems. The printer is available in a number of configurations and is compatible with the latest operating systems. The printer is available in a number of configurations and is compatible with the latest operating systems.

[illegible]

The DOWNING SYSTEM is quite often used to indicate a true local station in order to distinguish it from other local stations.

PROJECT

**Ray Elder presents
readers' solutions to
the first Project X
programming
challenge.**

When I wrote the first of these articles neither I nor the Color had any idea of the volume of response we would receive. In fact there were so many replies that it has taken me some time to read them all and write this follow-up article.

To refresh your memories, the project was to write a program to decode the kind of message which has been encoded by shifting the alphabet.

We had hundreds of solutions, all of which were valid, and the general standard of programming was so high that picking just one winner proved impossible. So my solution is to point a direction of the best and to mention those readers whose letters or programs (particularly notes) stood out for me.

The solutions were essentially based on one of three ideas, a set of loops shifting the three pointers and using the PM function. In between we had variations in style, user friendliness and even a moderately successful attempt of artificial intelligence.

Project leaders

Camel Green lives in Belgium, has a home-made robot as a hobby, and produced the program with which he has been able to check each word decoded but impossible letter combinations and stop to the next shift if it fails. The rule he used is that there are never more than four consonants together and short words contain at least one vowel.

Billy Hax used the PM to help him produce a three-line solution. **Chris Oliver** also used the PM to produce a coder and decoder however both his programs were identical except

for the messages, perhaps the two could be combined, a code/decode option given and the messages presented depending on the option selected?

Tom Mays used the three pointer method on 14 lines, **Karin Geoff** took 9 lines and **Charles Weiss** did it in 5 lines. Not that length matters (this isn't a contest).

Talking of length **Ben Govers**, of Green Software, sent in a one line solution. He also sent in an expanded version which was shown to everybody. Even that could have been shortened by changing the third element of the list to LIST 60-A0.

Mr. J. Woodham created both coder and decoder programs controlled by menu selection and sent a pleasant message — coded of course — **Irene Wittenborg** included a printer option, **Lee Canton** turned all the letters into their codes and manipulated them in a numerical array.

R. S. Bolson, along with many others, used FORB 20000.8 to set GATS LOCK on and around all input was in capital letters then used the conversion to numerical value method. **Herman Strong** added to the discussion by pointing out that time could be saved by looking at a short section of a long message, perhaps 15 letters, as this would be enough to see if it was intelligible or not.

Mr. A. S. Edwards used seven lines and suggested that I was not really trying with my own 23 line effort and **Wava Moody** used codes of 0000 as we can hold good programs should. **Mr. B. Bamberger** used an interesting two dimensional array with the PM function.

Finally a mention for the following readers who produced programs of equal merit but I just couldn't get them all in and a special thanks to **John L. Amphlett** who not only sent in an interesting letter pointing out that we were really processing CHARACT and not CODES (a nice by any other name etc. etc.) but also put forward an idea for a future Project X topic.

Mentions to **Brian Anthony**, **Ion Harris**, **E. M. Sedgwick**, **Sgt. Dave Brooks** I know SAT Sapper will turn about 10 years ago! **Cloran Buttrick** and **John Chapman**.

Small tokens of appreciation are on their way to all who got their programs published and my thanks to all who wrote in to me.

Now I must dash to start looking at the programs sent in for the last Project X and begin planning Project X Mini!

Project X solutions

```
1 REM Camel Green
2 INPUT "Words to decode" :a
3
4 10 INPUT "Please to shift 1-26"
5   OR 0 for computer ch
6   NEXT "a"
7 10 IF a=0 THEN LET a=1 :GOTO 10000
8 10 IF a=26 THEN LET a=1 :GOTO 10000
9 10 FOR i=1 TO LEN a
10 IF a(i)=a THEN LET a(i)=a
11 LET b(i)=CHR$(CODE a(i)+a)
12 IF CODE b(i)=10000 THEN LET b(i)=a(i)
13 IF CODE b(i)=10000 THEN LET b(i)=a(i)
14 IF CODE b(i)=10000 THEN LET b(i)=a(i)
15 IF CODE b(i)=10000 THEN LET b(i)=a(i)
16 IF CODE b(i)=10000 THEN LET b(i)=a(i)
17 IF CODE b(i)=10000 THEN LET b(i)=a(i)
18 IF CODE b(i)=10000 THEN LET b(i)=a(i)
19 IF CODE b(i)=10000 THEN LET b(i)=a(i)
20 IF CODE b(i)=10000 THEN LET b(i)=a(i)
21 IF CODE b(i)=10000 THEN LET b(i)=a(i)
22 IF CODE b(i)=10000 THEN LET b(i)=a(i)
23 IF CODE b(i)=10000 THEN LET b(i)=a(i)
24 IF CODE b(i)=10000 THEN LET b(i)=a(i)
25 IF CODE b(i)=10000 THEN LET b(i)=a(i)
26 IF CODE b(i)=10000 THEN LET b(i)=a(i)
27 IF CODE b(i)=10000 THEN LET b(i)=a(i)
28 IF CODE b(i)=10000 THEN LET b(i)=a(i)
29 IF CODE b(i)=10000 THEN LET b(i)=a(i)
30 IF CODE b(i)=10000 THEN LET b(i)=a(i)
31 IF CODE b(i)=10000 THEN LET b(i)=a(i)
32 IF CODE b(i)=10000 THEN LET b(i)=a(i)
33 IF CODE b(i)=10000 THEN LET b(i)=a(i)
34 IF CODE b(i)=10000 THEN LET b(i)=a(i)
35 IF CODE b(i)=10000 THEN LET b(i)=a(i)
36 IF CODE b(i)=10000 THEN LET b(i)=a(i)
37 IF CODE b(i)=10000 THEN LET b(i)=a(i)
38 IF CODE b(i)=10000 THEN LET b(i)=a(i)
39 IF CODE b(i)=10000 THEN LET b(i)=a(i)
40 IF CODE b(i)=10000 THEN LET b(i)=a(i)
41 IF CODE b(i)=10000 THEN LET b(i)=a(i)
42 IF CODE b(i)=10000 THEN LET b(i)=a(i)
43 IF CODE b(i)=10000 THEN LET b(i)=a(i)
44 IF CODE b(i)=10000 THEN LET b(i)=a(i)
45 IF CODE b(i)=10000 THEN LET b(i)=a(i)
46 IF CODE b(i)=10000 THEN LET b(i)=a(i)
47 IF CODE b(i)=10000 THEN LET b(i)=a(i)
48 IF CODE b(i)=10000 THEN LET b(i)=a(i)
49 IF CODE b(i)=10000 THEN LET b(i)=a(i)
50 IF CODE b(i)=10000 THEN LET b(i)=a(i)
51 IF CODE b(i)=10000 THEN LET b(i)=a(i)
52 IF CODE b(i)=10000 THEN LET b(i)=a(i)
53 IF CODE b(i)=10000 THEN LET b(i)=a(i)
54 IF CODE b(i)=10000 THEN LET b(i)=a(i)
55 IF CODE b(i)=10000 THEN LET b(i)=a(i)
56 IF CODE b(i)=10000 THEN LET b(i)=a(i)
57 IF CODE b(i)=10000 THEN LET b(i)=a(i)
58 IF CODE b(i)=10000 THEN LET b(i)=a(i)
59 IF CODE b(i)=10000 THEN LET b(i)=a(i)
60 IF CODE b(i)=10000 THEN LET b(i)=a(i)
61 IF CODE b(i)=10000 THEN LET b(i)=a(i)
62 IF CODE b(i)=10000 THEN LET b(i)=a(i)
63 IF CODE b(i)=10000 THEN LET b(i)=a(i)
64 IF CODE b(i)=10000 THEN LET b(i)=a(i)
65 IF CODE b(i)=10000 THEN LET b(i)=a(i)
66 IF CODE b(i)=10000 THEN LET b(i)=a(i)
67 IF CODE b(i)=10000 THEN LET b(i)=a(i)
68 IF CODE b(i)=10000 THEN LET b(i)=a(i)
69 IF CODE b(i)=10000 THEN LET b(i)=a(i)
70 IF CODE b(i)=10000 THEN LET b(i)=a(i)
71 IF CODE b(i)=10000 THEN LET b(i)=a(i)
72 IF CODE b(i)=10000 THEN LET b(i)=a(i)
73 IF CODE b(i)=10000 THEN LET b(i)=a(i)
74 IF CODE b(i)=10000 THEN LET b(i)=a(i)
75 IF CODE b(i)=10000 THEN LET b(i)=a(i)
76 IF CODE b(i)=10000 THEN LET b(i)=a(i)
77 IF CODE b(i)=10000 THEN LET b(i)=a(i)
78 IF CODE b(i)=10000 THEN LET b(i)=a(i)
79 IF CODE b(i)=10000 THEN LET b(i)=a(i)
80 IF CODE b(i)=10000 THEN LET b(i)=a(i)
81 IF CODE b(i)=10000 THEN LET b(i)=a(i)
82 IF CODE b(i)=10000 THEN LET b(i)=a(i)
83 IF CODE b(i)=10000 THEN LET b(i)=a(i)
84 IF CODE b(i)=10000 THEN LET b(i)=a(i)
85 IF CODE b(i)=10000 THEN LET b(i)=a(i)
86 IF CODE b(i)=10000 THEN LET b(i)=a(i)
87 IF CODE b(i)=10000 THEN LET b(i)=a(i)
88 IF CODE b(i)=10000 THEN LET b(i)=a(i)
89 IF CODE b(i)=10000 THEN LET b(i)=a(i)
90 IF CODE b(i)=10000 THEN LET b(i)=a(i)
91 IF CODE b(i)=10000 THEN LET b(i)=a(i)
92 IF CODE b(i)=10000 THEN LET b(i)=a(i)
93 IF CODE b(i)=10000 THEN LET b(i)=a(i)
94 IF CODE b(i)=10000 THEN LET b(i)=a(i)
95 IF CODE b(i)=10000 THEN LET b(i)=a(i)
96 IF CODE b(i)=10000 THEN LET b(i)=a(i)
97 IF CODE b(i)=10000 THEN LET b(i)=a(i)
98 IF CODE b(i)=10000 THEN LET b(i)=a(i)
99 IF CODE b(i)=10000 THEN LET b(i)=a(i)
100 IF CODE b(i)=10000 THEN LET b(i)=a(i)
```


DISCIPLE DATA

John Wase documents some new commands for the new interface.

A relative newcomer on the scene, the Disciple, as you probably know is a disc interface device. One of its features is that part of the software loads in from disc on booting up the system. Also, surely it doesn't corrupt your program but gets into paged RAM, a piece of memory which occupies the same location as the Spectrum ROM and can be "tagged" in and out just the same as software (I find that doesn't lose programs). The Disciple has a special command to enable you to get insight into that piece of paged RAM and alter directly the system variables that were set in it: the POKE (p) command, and is extremely powerful. Those who already have Disciple will possibly not be aware of this command, so I have produced a full list in Figure 1 for those who already know about POKE (p) then the same old commands in this list. So make a photocopy of the list and keep it with your Disciple Manual!

Most of the instructions are self-explanatory. For instance POKE (p) 0,n puts down the hexadecimal number n when the disc are in use to store hexadecimal proportions. Some are a bit more subtle though. POKE (p) 1 or 2 are for drives 1 and 2. POKE (p) with 40 or 80 for number of tracks on single sided discs, and 120 for each POKE (p) for double sided discs. POKE (p) 3 sets the step rate in milliseconds, what the value does not mention is that any figure you put in either in the set up program or by POKE (p) 3 will be automatically added to it. POKE (p) 41 turns tones off and is essential for setting up format; the value CHR\$ can then be omitted from printer control commands. POKE (p) 12 and 13 (12 for the low order byte and 13 for the high order byte) alter the length of the interfacing file, normally 49152 bytes, but I can just need to snapshot the screen, then POKE (p) with 49152 should do the trick.

Usually if you already have a Disciple and would like to distribute useful routines or comments of general interest please send them in to ZX Computing, marking your envelopes Disciple.

Figure 1

The Poke (p) Command

This command allows a value between 0 and 255 to be stored in the variable area of the disc operating system (DOS).

POKE (p) 0,n (n=0 to 8) (SECTOR) default 7
The sector output storage is essentially the sector number when reading and writing on the disc. This variable is added with the sector value that output to the screen border. If 0 is chosen, no border colour will occur. If 7 is chosen, all sectors will change the border colour.

POKE (p) 1,n (n=40 or 80 or 120) (TRACKS) default 255
This variable holds the number of tracks and whether double or single sided for disc drive one.

POKE (p) 2,n (n=40 or 80 or 120) (TRACKS) default 255
This variable holds the number of tracks and whether double or single sided for disc drive two.

POKE (p) 3,n (n=0 to 255) (STPAT) default 5
This variable allows you to set the DOS for the different step rates of drives (usually 12 to 30 milliseconds). The step rate is not allowed to go lower than six milliseconds.

POKE (p) 4,n (n=0 or 1) (EXPRT) default 0
If the setting is 0, then DOS uses the settings printer port. If 1, then the printer channel is unoccupied by DOS.

POKE (p) 5,n (n=0 to 255) (WIDTH) default 55
This variable sets the number of characters allowed per line when using PRINT and LIST with the graphics printer port.

POKE (p) 6,n (n=0 or 1) (PCODE) default 0
If the setting is 0, then the printer driver looks for alternate codes such as TAB and AT when PRINT and LIST commands. If the setting is 1, then the printer driver outputs directly to the printer the absolute value of the code 0 to 255. This is essential for sending control codes to the printer. E.g.

POKE (p) 6,1:PRINT CHR\$ 27;CHR\$ 60;POKE (p) 6,0

POKE (p) 7,n (n=0 to 255) (LSPCH) default 12
This variable sets the pointer line feed in increments of n/70 of an inch during graphics printing.

POKE (p) 8,n (n=0 to 255) (LFEED) default 1
This variable sets the number of line feeds automatically executed by the printer driver after a carriage return.

POKE (p) 9,n (n=0 to 255) (MARGIN) default 0
This variable sets the number of spaces inserted automatically by the printer driver after a carriage return. It is used for left hand margin control.

POKE (p) 10,n (n=0 or 1) (GRAPH) default 1
If the setting is 1, the printer driver generates the graphic representation of, and on PRINT and LIST statements. If the setting is 0, the normal printer character is output.

POKE (p) 11,n (n=1 to 64) (NSLAT) default 1
This variable sets the network station number. Be careful not to poke 0.

POKE (p) 12,n
and POKE (p) 13,n (n=0 to 49152) (LDRPT) default 49152
This variable controls the length of file which the unformatted files. POKE (p) 12 with the low order 256 byte and POKE (p) 13 with the high order 256 byte.

POKE (p) 14,n
and POKE (p) 15,n (n=0 to 65536) (JNTRN) default 0
If the setting is 0, then DOS returns to the Spectrum an error which do not relate to disk status or DOS syntax statements. If an address is given into these two locations, it relates a call indirectly to the address. It further avoids checking. Necessary when adding extra commands to the system.

**EXCLUSIVE
3 GAME OFFER
ONLY £9.95**

PLAN YOUR OWN EASTER SPORTS PROGRAMME!

**Peter Shilton's
Handball Maradona**
Normally £6.95, C&M 64,
Spectrum 48K and Amstrad



**NICK FALDO
Open**
Commodore



**Nick Faldo
Plays the Open**
Normally £9.95, C&M 64,
Spectrum 48K and Amstrad

**American Football
Bug-byte Top Hit**
Normally £2.95, C&M 64,
Spectrum 48K and Amstrad



BUG-BYTE

**The R.R.P. of these 3 products
is normally £19.89!**

**LOOK OUT FOR
THE NEW ARGUS
SOFTWARE PRODUCTS**

The Tube — C&M 64, Spectrum

Scorpion — C&M 64, Spectrum,
Amstrad

Romulus — C&M 64

Please send me The Sports Collection @ £9.95 (P&P Free)

Commodore 64 ☐ Spectrum 48K ☐ Amstrad CPC 464 ☐

Name

Address

Cheques/PO to: Argus Press Software
Victory House, Leicester Place, London WC2H 7NB

Credit Card Hotline: 01-439 0666

(Please allow 20 days for delivery) (Limited availability)

CAPTAIN COMPILER'S Cartoon Capers

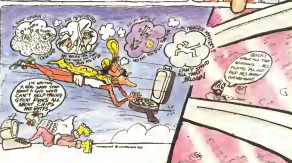
by *Jeffrey*

Captain Compiler, Super hero
and villain, has been seen
only in drawings in the
Savage Super Comics

Over my shoulder
and to the horizon
I just want to say
a big, huge thank you
for my
contributions and efforts.
It's a Compiler's love, more
than in the world of
A. THOMAS FOR THE
THE S-REPRESENT OF
SUPER HEROES.

HUH? I
I FEEL THE NEED TO IMAGE
'GEEK-HOOD' FOR A 'MAN, HAPPY-
DELAID' OF A GEEKY FOLK.

After a surprisingly speedy
transition, Captain Compiler
appears every where in the
info to have furthered
the Savage Super Hero.



Comic Strip



Comic Strip

Drawing

Let's begin with an explanation of the background drawing routine. This, remember, is made up of two elements, the line drawing, and the embellishments (bubbles and text) symbols over the door in the example screen. The line drawing data is held in a series of PLOT or DRAW co-ordinates, both set data with a PLOT pair and continues with a number of DRAW pairs. The DRAW co-ordinates are absolute, i.e. the actual co-ordinates of the pixels you want to draw to, not offset values as in the Basic DRAW. This makes inputting values from a drawing on pixel paper much

of this type of element to be printed, so in Fig. 1 this would be 3 — as there are two tokens and the numeric symbol block is printed. These then follow a series of numbers in blocks of three. The first of these is the length of that data block (maximum 255), which begins at the address given by the next two numbers, so the sequence:

3,120,195,10,190,195,6,200,195

would print three objects. The first is made up of 7 bytes of data beginning at 195+195*256=50000. The remaining numbers would translate in a similar way. *a) Access* — As before a starting 0 means miss out this element. Any



other number is the total number of tokens to print (for Fig. 1 this would be 2 — one for the roll and one for the foreground character). The number is followed by others in blocks of five. The first number in each block is the Print Address, the second the column, the third and fourth the address (held as before) and the fifth is the type of token shape. There are three types of token available. Type 0 is three character squares wide by four high, for small characters like the roll. Type 1 is four wide by five high, and type 2 is a 6x6 grid for close-ups (or two small

PROGRAM 1

```
10 FOR f=from 10 end: INPUT i:
   POKe f,i: PRINT f,i: NEXT f
20 REM alter from & end number
   as required
```

easier. When the routine finds a 1 then next pair starts a new PLOT position. A terminal 0 is needed to return from the routine.

To use this element, the data (PLOT, DRAW, is and is) are held in a series of bytes starting at a known address, like 50000, and the routine is called with a line such as:

```
10 POKe 33384, 30: POKe 33397,
   195: RANDOMISE USR 50000
```

The POKe (note the italics) bytes of the Printer buffer) are the low and high values of the data start address (33384+195*256=50000). Look at the beginning of Table 8, the data will use for the demonstration at the end and you'll see the 195 which sets the PLOT position and the 0 to return.

The remainder of the elements are called with a single USR call. Don't try to link these routines with the background in one single call. The routine will crash! This call produces the embellishments, the tokens and the speech bubble. The intervention to control the printing (but not the data itself) is held in a string of numbers as follows —

a) Embellishments — A 0 to start means miss out this element. Any other number is the total number

1 19 35 37 49 41 73 85	2 14 26	3 16 27 29 31 43 75 87
4 10 24 40 52 64 76 88	5 17	6 18 30 42 54 66 78 90
7 19 31 43 55 67 79 91	8 20	9 21 33 45 57 69 81 93
10 22 34 46 58 70 82 94	11 23 95	12 24 36 48 60 72 84 96

Figure 1



characters, which always appear together). Each is printed in such a way as not to obliterate the background where no icon pixels are being filled. The sequence:

```
2,14,62,195,0,2,0,62,195,2
```

would print two icons, the first a solid black with top left corner of row 1 column 4, then data starting at 60000. It's leave you to work out the second. Note that this routine only operates within the top third of the screen. If the bottom of a block comes below row 7, strange things will happen!

Speech Bubble — Again, 8 means into cell otherwise the first (and only) digit refers to the column where the pointer 'V' is printed (in Fig. 1 it's at column 58). The start of such a control string is P0450 (low byte) and P0451 (high byte) before calling the routine with **RANDOMIZE USR P0450**.

Data

Now to the actual data to which the control string refers.

a) **Intellifonts** — This is held at:

```
22 row column, codes of characters to be printed (see Appendix A of your Spectrum manual) referring with 22 row column otherwise a row print position is required. So the data string
```

```
22,1,2,48,6980,22,2,2,6,6
```

(which, incidentally, has a length of 31 — you could the 22 as well) would print '000' at row 1, column 1 and '6F' at row 2, column 5. You can include block graphics or any user designed graphics you have made (20 are available as A is used for the bubble printed). You can also include control characters (see p143 in the manual) — 16, 3 would print 68 to red, 17, 3 would print as yellow paper, 19, 1 would have the colour right and 18, 1 would have them flashing. (In 22 row column, 18, 1, 18, 137 for minor animation.)

b) **Icon data** — The data for these is the actual bit patterns for each icon byte to be used. Each character square on screen is made up of eight lines of eight pixels (one below the other, but held in the computer's memory one after the other) to make pictures worse they are held in a rather strange order. To see what I mean enter the line

```
FOR I=16384 TO 16432: P0451, 255: P0450 20: NEXT I
```

and watch the way the screen fills. What this amounts to is, apparently, nothing less than confusion if isn't really. Refer to Fig. 2 and you'll see that you're told to P0451 in the bit pattern of each byte (using the 68k notation) in the order given for a 384 icon. (If you're still confused well, well, I'm not sure I can supply an editor program to do the job for you.)

So and let's pull all the ideas together with a grand demonstration at the facilities offered. We'll reproduce the top right picture from Fig. 1 but to save us typing we'll just use one icon, the ball, and the speech bubble will remain empty (we'll deal with how to fill that later).

After this RUN the loader program to enter the data from 16384 to 16431. Enter just the numbers, not the text lines, those are to help you find your way around. Now enter the loader again to enter the control string data (Table 2). We address 60000 to 60010.

We're almost ready to roll, but we need a line to call up the routine. Type in Program 2 and RUN for the glory. The 814s will explain what each part does.

Now what about that empty speech bubble? There are eight character squares while it, so (provided you only need that number of spaces as a maximum) you could just PRINT **AT 5, 32 " "** (see the 'RANDOMIZE' in Fig. 1). That doesn't help it, so it makes sense, you need to display more information. Program 3 will come in your aid and it is a basic routine for scrolling text through a window.

Table 1

Designed data

175	175	177	188	232	188	232	127
177	127	177	188	1	232	188	232
175	1	232	127	232	127	1	177
127	176	112	0				

Table 1 (continued) data

22	2	22	128	22	2	19	122
----	---	----	-----	----	---	----	-----

Table 2

0	0	0	40	17	0	95	125
0	15	128	0	0	0	0	04
21	0	95	202	0	15	128	0
7	248	0	95	202	0	147	248
128	10	128	0	7	248	0	95
202	0	77	249	0	15	128	0
0	0	0	95	125	0	15	128
0	15	128	0	7	72	0	95
202	0	15	128	0	7	112	0
15	132	0	95	125	0	15	128
0	15	128	0	37	216	0	95
202	0	15	128	0	27	72	0

It takes the message string, m0 adds eight bytes (spaces) to the front to make the bubble start empty then uses a loop, set to the original length of the string to print successive sections of the new string in the speech bubble. Pressing SPACE at any time during the scroll will reset it. Add to it your own (Program 3) and try it out. Well, that's stage one in the can. Watch this space for further developments.

Table 3

P_topch

1 0 105 195

P_botch

1 2 25 115 195 0

Bubble pointer

28

PROGRAM 2

```
5 REM drawCODE must be on board
```

```
10 FOR 23276,88: FOR 23277,1  
15: RANDOMIZE USR 65177: P0451 64  
1942,195: P0451 64743,204: RANDOMIZE  
10 USR 64736
```

```
20 REM draw background then  
P0451 0
```

PROGRAM 3

```
50 LET str="THIS IS AN EXAMPLE  
OF SCROLLING IN A SPEECH BUBBLE"  
10 LET maxlen=LEN str
```

```
150
```

```
20 FOR i=1 TO maxlen PRINT AT 1,23  
PRINT TO i+7: PAUSE 10: IF INKEY  
V#="" THEN GO TO 30
```

```
70 NEXT i
```

SHORTCUTS

More compact classics from ZX80 readers.

Another mixed bag of programs which prove the old saying about good things coming in small packages. This month we start with a program which will hopefully simplify some of the burden of questions to Commodore.

It is a program which will be

greated warmly by all those who purchased a Commodore printer package, there are still a lot of problems to solve but this is a long and so it gets the coveted Star Cut award



48K R3232

Chris Reynolds of Huddersfield supplies the solution to getting the printer to work in 48K mode and all you have to do is type in his listing and run it.

Provided no error report is produced, SayC the code using **SAVE "R3232" CODE 48000,REL** to use it make sure you have reloaded it each time and type **SAVECODES:USP 48000** and all subsequent **PRINTING** will be done via the R3232.

One problem is that no line feed is sent at the end of a line, so you will have to set your printer's DIP switches to produce an automatic line feed after a carriage return if this is not possible then you are limited to manual advancing the machine at the end of each line and **USING** is likely to become a little messy.

The **REL** code is automatically set to a default of 48000 if you require another setting then one way is to **FORWARD** as usual in 64K mode and make a note of the value of address **200010** by typing them, and then in 48K mode **FORW** them into addresses **200200**.

```
1 REM 48K R3232
2
3 CLEAR 64000
4 LET CPU
5 FOR I=48000 TO 48100
6   REM 48K LET address F000 F,0
7   NEXT I
8 IF CPU=0 THEN PRINT AT
9   "F0000" GOTO 60000:GOTO 3,-10
10  REM 48K 40,70,90,100,70,30,400,
11  20,40,10,200,700,100,100,104,100,1
12  1,0,104,104,70,100,100,100,100,100
13  10
14  REM 48K 240,14,200,200,200,20,1
15  40,44,40,14,300,100,200,70,200,1
16  10,100,100,44,10,200,400,10,100,1
17  10,200,100,200,60,200,200,40,100,4
18  10,200,100
19  REM 48K 240,40,200,70,100,1,100
20  7,400,100,200,200,200,200,200,100
21  10,4,200,4,200,100,200,4,40,100,
22  100,70,200,100,4,4,200,100,70,
23  10,100,200,100
24  REM 48K 40,100,100,100,100,100,100,100
25  10,200,100,200,100,100,100,100
26  10,100,100,100,100,100,100,100
```

Stripes

Berry Breen a resident of Liverpool, sent to one of those programs which tells into the "how-to-use-what-I-can-use-it-for-but-I-might-come-to-hugely-over-kill" category.

The program draws several broad stripes across the screen then resolves them vertically at a speed selected by you and very pretty and possibly of use to hypnotists.

```
1 REM Stripes
2
3 LET CPU=0:FOR I=0 TO 320
4   REM 64K 40,70,90,100,70,30,400,
5   20,40,10,200,700,100,100,104,100,1
6   1,0,104,104,70,100,100,100,100,100
7   10
8   REM 64K 240,14,200,200,200,20,1
9   40,44,40,14,300,100,200,70,200,1
10  10,100,100,44,10,200,400,10,100,1
11  10,200,100,200,60,200,200,40,100,4
12  10,200,100
13  REM 64K 240,40,200,70,100,1,100
14  7,400,100,200,200,200,200,200,100
15  10,4,200,4,200,100,200,4,40,100,
16  100,70,200,100,4,4,200,100,70,
17  10,100,200,100
18  REM 64K 40,100,100,100,100,100,100,100
19  10,200,100,200,100,100,100,100
20  10,100,100,100,100,100,100,100
```

Shadow Effect

Screen presentation is a common preoccupation and this routine produces an unusual and pleasing effect. Gary Franklin of Conner Island writes it and once the machine code listing has been entered, run and saved, then it will produce anything on the screen printed with **PRINT** &.

Experiment with the lines as suggested to produce your own individual effect.

```
1 REM Shadow Effect
2
3 CLEAR 64000:PRINT:LET
4 CPU
5 FOR I=0 TO 320
6   REM 64K 40,70,90,100,70,30,400,
7   20,40,10,200,700,100,100,104,100,1
8   1,0,104,104,70,100,100,100,100,100
9   10
10  REM 64K 240,14,200,200,200,20,1
11  40,44,40,14,300,100,200,70,200,1
12  10,100,100,44,10,200,400,10,100,1
13  10,200,100,200,60,200,200,40,100,4
14  10,200,100
15  REM 64K 240,40,200,70,100,1,100
16  7,400,100,200,200,200,200,200,100
17  10,4,200,4,200,100,200,4,40,100,
18  100,70,200,100,4,4,200,100,70,
19  10,100,200,100
20  REM 64K 40,100,100,100,100,100,100,100
21  10,200,100,200,100,100,100,100
22  10,100,100,100,100,100,100,100
23  REM 64K 40,100,100,100,100,100,100,100
24  10,200,100,200,100,100,100,100
25  10,100,100,100,100,100,100,100
26  REM 64K 40,100,100,100,100,100,100,100
27  10,200,100,200,100,100,100,100
28  10,100,100,100,100,100,100,100
29  REM 64K 40,100,100,100,100,100,100,100
30  10,200,100,200,100,100,100,100
31  10,100,100,100,100,100,100,100
32  REM 64K 40,100,100,100,100,100,100,100
33  10,200,100,200,100,100,100,100
34  10,100,100,100,100,100,100,100
35  REM 64K 40,100,100,100,100,100,100,100
36  10,200,100,200,100,100,100,100
37  10,100,100,100,100,100,100,100
38  REM 64K 40,100,100,100,100,100,100,100
39  10,200,100,200,100,100,100,100
40  10,100,100,100,100,100,100,100
41  REM 64K 40,100,100,100,100,100,100,100
42  10,200,100,200,100,100,100,100
43  10,100,100,100,100,100,100,100
44  REM 64K 40,100,100,100,100,100,100,100
45  10,200,100,200,100,100,100,100
46  10,100,100,100,100,100,100,100
47  REM 64K 40,100,100,100,100,100,100,100
48  10,200,100,200,100,100,100,100
49  10,100,100,100,100,100,100,100
50  REM 64K 40,100,100,100,100,100,100,100
51  10,200,100,200,100,100,100,100
52  10,100,100,100,100,100,100,100
53  REM 64K 40,100,100,100,100,100,100,100
54  10,200,100,200,100,100,100,100
55  10,100,100,100,100,100,100,100
56  REM 64K 40,100,100,100,100,100,100,100
57  10,200,100,200,100,100,100,100
58  10,100,100,100,100,100,100,100
59  REM 64K 40,100,100,100,100,100,100,100
60  10,200,100,200,100,100,100,100
61  10,100,100,100,100,100,100,100
62  REM 64K 40,100,100,100,100,100,100,100
63  10,200,100,200,100,100,100,100
64  10,100,100,100,100,100,100,100
65  REM 64K 40,100,100,100,100,100,100,100
66  10,200,100,200,100,100,100,100
67  10,100,100,100,100,100,100,100
68  REM 64K 40,100,100,100,100,100,100,100
69  10,200,100,200,100,100,100,100
70  10,100,100,100,100,100,100,100
71  REM 64K 40,100,100,100,100,100,100,100
72  10,200,100,200,100,100,100,100
73  10,100,100,100,100,100,100,100
74  REM 64K 40,100,100,100,100,100,100,100
75  10,200,100,200,100,100,100,100
76  10,100,100,100,100,100,100,100
77  REM 64K 40,100,100,100,100,100,100,100
78  10,200,100,200,100,100,100,100
79  10,100,100,100,100,100,100,100
80  REM 64K 40,100,100,100,100,100,100,100
81  10,200,100,200,100,100,100,100
82  10,100,100,100,100,100,100,100
83  REM 64K 40,100,100,100,100,100,100,100
84  10,200,100,200,100,100,100,100
85  10,100,100,100,100,100,100,100
86  REM 64K 40,100,100,100,100,100,100,100
87  10,200,100,200,100,100,100,100
88  10,100,100,100,100,100,100,100
89  REM 64K 40,100,100,100,100,100,100,100
90  10,200,100,200,100,100,100,100
91  10,100,100,100,100,100,100,100
92  REM 64K 40,100,100,100,100,100,100,100
93  10,200,100,200,100,100,100,100
94  10,100,100,100,100,100,100,100
95  REM 64K 40,100,100,100,100,100,100,100
96  10,200,100,200,100,100,100,100
97  10,100,100,100,100,100,100,100
98  REM 64K 40,100,100,100,100,100,100,100
99  10,200,100,200,100,100,100,100
100 10,100,100,100,100,100,100,100
```



Biggies

This is a bit of a mystery which I thought I include as a challenge to computer sophisticates.

The program enlarges and prints characters to the screen in a 3x3 size. The documentation was nonexistent (but the program produced a picture and I can tell you that changing the DATA line will change the message. A=23, B=58)

Hardly for fast, short headings but if you want to extend the message length or change the printed screen position then you have to get your dissembler out

Mini-label

A. Welch from Glasgow sent this small program to produce a small label on the printer when you then cut out and stick to the cassette or PCMC between the edge of the cassette insert and the PCMC card.

The S --- F prompt provides an easy way of formatting the layout of the output line.

```

1 REM Mini-Label
2
30 INPUT T: T=0 TO 9: PRINT T: G
  GOTO 10
40 INPUT S: S=" " TO "F": GOTO 10
50 INPUT F: F=" " TO "S": GOTO 10
60 PRINT "Cassette Printer: A=23, B=58"
70 PRINT S
80 PRINT AT 20,1:PRINT F TO T
  GOTO 40
90 PRINT "COPY: PAPER: A TO T"
  GOTO 40
100 PRINT "
  TO 20:PRINT "ON: 20:PRINT "
  GOTO 40
110 PRINT "
  TO 20:PRINT "
  GOTO 40
120 PRINT "
  TO 20:PRINT "
  GOTO 40
130 PRINT "
  TO 20:PRINT "
  GOTO 40
140 PRINT "
  TO 20:PRINT "
  GOTO 40
150 PRINT "
  TO 20:PRINT "
  GOTO 40
160 PRINT "
  TO 20:PRINT "
  GOTO 40
170 PRINT "
  TO 20:PRINT "
  GOTO 40
180 PRINT "
  TO 20:PRINT "
  GOTO 40
190 PRINT "
  TO 20:PRINT "
  GOTO 40
200 PRINT "
  TO 20:PRINT "
  GOTO 40
210 PRINT "
  TO 20:PRINT "
  GOTO 40
220 PRINT "
  TO 20:PRINT "
  GOTO 40
230 PRINT "
  TO 20:PRINT "
  GOTO 40
240 PRINT "
  TO 20:PRINT "
  GOTO 40
250 PRINT "
  TO 20:PRINT "
  GOTO 40
260 PRINT "
  TO 20:PRINT "
  GOTO 40
270 PRINT "
  TO 20:PRINT "
  GOTO 40
280 PRINT "
  TO 20:PRINT "
  GOTO 40
290 PRINT "
  TO 20:PRINT "
  GOTO 40
300 PRINT "
  TO 20:PRINT "
  GOTO 40
310 PRINT "
  TO 20:PRINT "
  GOTO 40
320 PRINT "
  TO 20:PRINT "
  GOTO 40
330 PRINT "
  TO 20:PRINT "
  GOTO 40
340 PRINT "
  TO 20:PRINT "
  GOTO 40
350 PRINT "
  TO 20:PRINT "
  GOTO 40
360 PRINT "
  TO 20:PRINT "
  GOTO 40
370 PRINT "
  TO 20:PRINT "
  GOTO 40
380 PRINT "
  TO 20:PRINT "
  GOTO 40
390 PRINT "
  TO 20:PRINT "
  GOTO 40
400 PRINT "
  TO 20:PRINT "
  GOTO 40
410 PRINT "
  TO 20:PRINT "
  GOTO 40
420 PRINT "
  TO 20:PRINT "
  GOTO 40
430 PRINT "
  TO 20:PRINT "
  GOTO 40
440 PRINT "
  TO 20:PRINT "
  GOTO 40
450 PRINT "
  TO 20:PRINT "
  GOTO 40
460 PRINT "
  TO 20:PRINT "
  GOTO 40
470 PRINT "
  TO 20:PRINT "
  GOTO 40
480 PRINT "
  TO 20:PRINT "
  GOTO 40
490 PRINT "
  TO 20:PRINT "
  GOTO 40
500 PRINT "
  TO 20:PRINT "
  GOTO 40
510 PRINT "
  TO 20:PRINT "
  GOTO 40
520 PRINT "
  TO 20:PRINT "
  GOTO 40
530 PRINT "
  TO 20:PRINT "
  GOTO 40
540 PRINT "
  TO 20:PRINT "
  GOTO 40
550 PRINT "
  TO 20:PRINT "
  GOTO 40
560 PRINT "
  TO 20:PRINT "
  GOTO 40
570 PRINT "
  TO 20:PRINT "
  GOTO 40
580 PRINT "
  TO 20:PRINT "
  GOTO 40
590 PRINT "
  TO 20:PRINT "
  GOTO 40
600 PRINT "
  TO 20:PRINT "
  GOTO 40
610 PRINT "
  TO 20:PRINT "
  GOTO 40
620 PRINT "
  TO 20:PRINT "
  GOTO 40
630 PRINT "
  TO 20:PRINT "
  GOTO 40
640 PRINT "
  TO 20:PRINT "
  GOTO 40
650 PRINT "
  TO 20:PRINT "
  GOTO 40
660 PRINT "
  TO 20:PRINT "
  GOTO 40
670 PRINT "
  TO 20:PRINT "
  GOTO 40
680 PRINT "
  TO 20:PRINT "
  GOTO 40
690 PRINT "
  TO 20:PRINT "
  GOTO 40
700 PRINT "
  TO 20:PRINT "
  GOTO 40
710 PRINT "
  TO 20:PRINT "
  GOTO 40
720 PRINT "
  TO 20:PRINT "
  GOTO 40
730 PRINT "
  TO 20:PRINT "
  GOTO 40
740 PRINT "
  TO 20:PRINT "
  GOTO 40
750 PRINT "
  TO 20:PRINT "
  GOTO 40
760 PRINT "
  TO 20:PRINT "
  GOTO 40
770 PRINT "
  TO 20:PRINT "
  GOTO 40
780 PRINT "
  TO 20:PRINT "
  GOTO 40
790 PRINT "
  TO 20:PRINT "
  GOTO 40
800 PRINT "
  TO 20:PRINT "
  GOTO 40
810 PRINT "
  TO 20:PRINT "
  GOTO 40
820 PRINT "
  TO 20:PRINT "
  GOTO 40
830 PRINT "
  TO 20:PRINT "
  GOTO 40
840 PRINT "
  TO 20:PRINT "
  GOTO 40
850 PRINT "
  TO 20:PRINT "
  GOTO 40
860 PRINT "
  TO 20:PRINT "
  GOTO 40
870 PRINT "
  TO 20:PRINT "
  GOTO 40
880 PRINT "
  TO 20:PRINT "
  GOTO 40
890 PRINT "
  TO 20:PRINT "
  GOTO 40
900 PRINT "
  TO 20:PRINT "
  GOTO 40
910 PRINT "
  TO 20:PRINT "
  GOTO 40
920 PRINT "
  TO 20:PRINT "
  GOTO 40
930 PRINT "
  TO 20:PRINT "
  GOTO 40
940 PRINT "
  TO 20:PRINT "
  GOTO 40
950 PRINT "
  TO 20:PRINT "
  GOTO 40
960 PRINT "
  TO 20:PRINT "
  GOTO 40
970 PRINT "
  TO 20:PRINT "
  GOTO 40
980 PRINT "
  TO 20:PRINT "
  GOTO 40
990 PRINT "
  TO 20:PRINT "
  GOTO 40

```

Attributes

Another IBM specific program: this one from beta programmer D.J. Walker changes the colour of the IBM cell screens. A useful program for those who find the default a bit eye-straining.

The code can be located anywhere except in the top page and the destination address for your code is required to be input at the start of the program.

```

1 REM IBM Attributes
2
30 INPUT "Load Address?" AT100
  GOTO 40
40 INPUT "Main Page" M: M=100
50 INPUT "Main Page" M: M=100
60 INPUT "Main Page" M: M=100
70 INPUT "Main Page" M: M=100
80 INPUT "Main Page" M: M=100
90 INPUT "Main Page" M: M=100
100 INPUT "Main Page" M: M=100
110 INPUT "Main Page" M: M=100
120 INPUT "Main Page" M: M=100
130 INPUT "Main Page" M: M=100
140 INPUT "Main Page" M: M=100
150 INPUT "Main Page" M: M=100
160 INPUT "Main Page" M: M=100
170 INPUT "Main Page" M: M=100
180 INPUT "Main Page" M: M=100
190 INPUT "Main Page" M: M=100
200 INPUT "Main Page" M: M=100
210 INPUT "Main Page" M: M=100
220 INPUT "Main Page" M: M=100
230 INPUT "Main Page" M: M=100
240 INPUT "Main Page" M: M=100
250 INPUT "Main Page" M: M=100
260 INPUT "Main Page" M: M=100
270 INPUT "Main Page" M: M=100
280 INPUT "Main Page" M: M=100
290 INPUT "Main Page" M: M=100
300 INPUT "Main Page" M: M=100
310 INPUT "Main Page" M: M=100
320 INPUT "Main Page" M: M=100
330 INPUT "Main Page" M: M=100
340 INPUT "Main Page" M: M=100
350 INPUT "Main Page" M: M=100
360 INPUT "Main Page" M: M=100
370 INPUT "Main Page" M: M=100
380 INPUT "Main Page" M: M=100
390 INPUT "Main Page" M: M=100
400 INPUT "Main Page" M: M=100
410 INPUT "Main Page" M: M=100
420 INPUT "Main Page" M: M=100
430 INPUT "Main Page" M: M=100
440 INPUT "Main Page" M: M=100
450 INPUT "Main Page" M: M=100
460 INPUT "Main Page" M: M=100
470 INPUT "Main Page" M: M=100
480 INPUT "Main Page" M: M=100
490 INPUT "Main Page" M: M=100
500 INPUT "Main Page" M: M=100
510 INPUT "Main Page" M: M=100
520 INPUT "Main Page" M: M=100
530 INPUT "Main Page" M: M=100
540 INPUT "Main Page" M: M=100
550 INPUT "Main Page" M: M=100
560 INPUT "Main Page" M: M=100
570 INPUT "Main Page" M: M=100
580 INPUT "Main Page" M: M=100
590 INPUT "Main Page" M: M=100
600 INPUT "Main Page" M: M=100
610 INPUT "Main Page" M: M=100
620 INPUT "Main Page" M: M=100
630 INPUT "Main Page" M: M=100
640 INPUT "Main Page" M: M=100
650 INPUT "Main Page" M: M=100
660 INPUT "Main Page" M: M=100
670 INPUT "Main Page" M: M=100
680 INPUT "Main Page" M: M=100
690 INPUT "Main Page" M: M=100
700 INPUT "Main Page" M: M=100
710 INPUT "Main Page" M: M=100
720 INPUT "Main Page" M: M=100
730 INPUT "Main Page" M: M=100
740 INPUT "Main Page" M: M=100
750 INPUT "Main Page" M: M=100
760 INPUT "Main Page" M: M=100
770 INPUT "Main Page" M: M=100
780 INPUT "Main Page" M: M=100
790 INPUT "Main Page" M: M=100
800 INPUT "Main Page" M: M=100
810 INPUT "Main Page" M: M=100
820 INPUT "Main Page" M: M=100
830 INPUT "Main Page" M: M=100
840 INPUT "Main Page" M: M=100
850 INPUT "Main Page" M: M=100
860 INPUT "Main Page" M: M=100
870 INPUT "Main Page" M: M=100
880 INPUT "Main Page" M: M=100
890 INPUT "Main Page" M: M=100
900 INPUT "Main Page" M: M=100
910 INPUT "Main Page" M: M=100
920 INPUT "Main Page" M: M=100
930 INPUT "Main Page" M: M=100
940 INPUT "Main Page" M: M=100
950 INPUT "Main Page" M: M=100
960 INPUT "Main Page" M: M=100
970 INPUT "Main Page" M: M=100
980 INPUT "Main Page" M: M=100
990 INPUT "Main Page" M: M=100

```

1 REM biggies

2

```

10 LET sum=0 FOR f=50000 TO 500
95: READ a: FOR e f,a: LET sum+=a
NEXT f
20 IF sum/9471 THEN PRINT "Det
a error": STOP
30 RANDOMIZE USR 50000
100 DATA 1,162,195,33,36,64,17,
0,41,229,10,38,0,111,41,41,25,
235,225,197,229,6,2,197,6,4,197,
200,138,195,36,200,138,195,36,1
9,193,16,243,1,224,7,237,66,193,
16,233,225,35,35,193,3,203,117,1
92,24,204,229,24,6,4,23,203,22,2
03,22,203,78,40,2,203,198,16,243,
35,203,69,32,236,225,201
110 DATA 58,56,0,35,47,45,48,53,
52,41,46,39

```

Feedback

A nice feature of this section is the lively correspondence we get in response to the programs we print. READ200 in our February issue drew a reply from WE Thomson of Aldershot who comments on the ingenuity of the previous program that offers his version of it (titled of the length).

DEC-HIS makes good use of the RM function, one calling the other and also employs the technique of splitting a program into simple steps.

Alan Knight of Bromley writes that Ben Granger's December routine for producing 64 columns in a line won't work in RM mode. He is because 600 "Y" is used and the 128 also needs it as a play token. He continues that the assumption that the 128's are at the top of memory suffers when you use other code in the gap and have perhaps relocated the 128's. (Or even if you have a 16K machine).

Finally should you attempt to print a character with a code of less than 32 (by mistake) then the program crashes. The test for mode tends to disrupt the stack. However not being one to merely offer negative criticism Alan has provided a revision section which allows for all these things. We print it as HEX dumps and you can replace the code in the December issue with it or enter it with a loader program to address 38000 if it is 16K bytes long.

If you replace the old printed code then you will have to change line 130 to:

130 SAVE W: CODE 38000,154.

1 REM DEC-HIS

2

```

30 DEF FN H28(X)=123456789ABCDEF12345
40 DEF FN H28(X)=123456789ABCDEF12345
50 DEF FN H28(X)=123456789ABCDEF12345
60 DEF FN H28(X)=123456789ABCDEF12345
70 DEF FN H28(X)=123456789ABCDEF12345
80 DEF FN H28(X)=123456789ABCDEF12345
90 DEF FN H28(X)=123456789ABCDEF12345
100 DEF FN H28(X)=123456789ABCDEF12345
110 DEF FN H28(X)=123456789ABCDEF12345
120 DEF FN H28(X)=123456789ABCDEF12345
130 DEF FN H28(X)=123456789ABCDEF12345
140 DEF FN H28(X)=123456789ABCDEF12345
150 DEF FN H28(X)=123456789ABCDEF12345
160 DEF FN H28(X)=123456789ABCDEF12345
170 DEF FN H28(X)=123456789ABCDEF12345
180 DEF FN H28(X)=123456789ABCDEF12345
190 DEF FN H28(X)=123456789ABCDEF12345
200 DEF FN H28(X)=123456789ABCDEF12345
210 DEF FN H28(X)=123456789ABCDEF12345
220 DEF FN H28(X)=123456789ABCDEF12345
230 DEF FN H28(X)=123456789ABCDEF12345
240 DEF FN H28(X)=123456789ABCDEF12345
250 DEF FN H28(X)=123456789ABCDEF12345
260 DEF FN H28(X)=123456789ABCDEF12345
270 DEF FN H28(X)=123456789ABCDEF12345
280 DEF FN H28(X)=123456789ABCDEF12345
290 DEF FN H28(X)=123456789ABCDEF12345
300 DEF FN H28(X)=123456789ABCDEF12345
310 DEF FN H28(X)=123456789ABCDEF12345
320 DEF FN H28(X)=123456789ABCDEF12345
330 DEF FN H28(X)=123456789ABCDEF12345
340 DEF FN H28(X)=123456789ABCDEF12345
350 DEF FN H28(X)=123456789ABCDEF12345
360 DEF FN H28(X)=123456789ABCDEF12345
370 DEF FN H28(X)=123456789ABCDEF12345
380 DEF FN H28(X)=123456789ABCDEF12345
390 DEF FN H28(X)=123456789ABCDEF12345
400 DEF FN H28(X)=123456789ABCDEF12345
410 DEF FN H28(X)=123456789ABCDEF12345
420 DEF FN H28(X)=123456789ABCDEF12345
430 DEF FN H28(X)=123456789ABCDEF12345
440 DEF FN H28(X)=123456789ABCDEF12345
450 DEF FN H28(X)=123456789ABCDEF12345
460 DEF FN H28(X)=123456789ABCDEF12345
470 DEF FN H28(X)=123456789ABCDEF12345
480 DEF FN H28(X)=123456789ABCDEF12345
490 DEF FN H28(X)=123456789ABCDEF12345
500 DEF FN H28(X)=123456789ABCDEF12345
510 DEF FN H28(X)=123456789ABCDEF12345
520 DEF FN H28(X)=123456789ABCDEF12345
530 DEF FN H28(X)=123456789ABCDEF12345
540 DEF FN H28(X)=123456789ABCDEF12345
550 DEF FN H28(X)=123456789ABCDEF12345
560 DEF FN H28(X)=123456789ABCDEF12345
570 DEF FN H28(X)=123456789ABCDEF12345
580 DEF FN H28(X)=123456789ABCDEF12345
590 DEF FN H28(X)=123456789ABCDEF12345
600 DEF FN H28(X)=123456789ABCDEF12345
610 DEF FN H28(X)=123456789ABCDEF12345
620 DEF FN H28(X)=123456789ABCDEF12345
630 DEF FN H28(X)=123456789ABCDEF12345
640 DEF FN H28(X)=123456789ABCDEF12345
650 DEF FN H28(X)=123456789ABCDEF12345
660 DEF FN H28(X)=123456789ABCDEF12345
670 DEF FN H28(X)=123456789ABCDEF12345
680 DEF FN H28(X)=123456789ABCDEF12345
690 DEF FN H28(X)=123456789ABCDEF12345
700 DEF FN H28(X)=123456789ABCDEF12345
710 DEF FN H28(X)=123456789ABCDEF12345
720 DEF FN H28(X)=123456789ABCDEF12345
730 DEF FN H28(X)=123456789ABCDEF12345
740 DEF FN H28(X)=123456789ABCDEF12345
750 DEF FN H28(X)=123456789ABCDEF12345
760 DEF FN H28(X)=123456789ABCDEF12345
770 DEF FN H28(X)=123456789ABCDEF12345
780 DEF FN H28(X)=123456789ABCDEF12345
790 DEF FN H28(X)=123456789ABCDEF12345
800 DEF FN H28(X)=123456789ABCDEF12345
810 DEF FN H28(X)=123456789ABCDEF12345
820 DEF FN H28(X)=123456789ABCDEF12345
830 DEF FN H28(X)=123456789ABCDEF12345
840 DEF FN H28(X)=123456789ABCDEF12345
850 DEF FN H28(X)=123456789ABCDEF12345
860 DEF FN H28(X)=123456789ABCDEF12345
870 DEF FN H28(X)=123456789ABCDEF12345
880 DEF FN H28(X)=123456789ABCDEF12345
890 DEF FN H28(X)=123456789ABCDEF12345
900 DEF FN H28(X)=123456789ABCDEF12345
910 DEF FN H28(X)=123456789ABCDEF12345
920 DEF FN H28(X)=123456789ABCDEF12345
930 DEF FN H28(X)=123456789ABCDEF12345
940 DEF FN H28(X)=123456789ABCDEF12345
950 DEF FN H28(X)=123456789ABCDEF12345
960 DEF FN H28(X)=123456789ABCDEF12345
970 DEF FN H28(X)=123456789ABCDEF12345
980 DEF FN H28(X)=123456789ABCDEF12345
990 DEF FN H28(X)=123456789ABCDEF12345

```

64 Revisited

```

0A 7B 5C 1E 9D 0A 19 65
0B 61 3A 4B 5C 7E FE 03
0C 01 0A 0A 09 5A 23 5A
0D 7A FE 4B 4B 3B 03 1A 0A
0E 7B FE 1A 0A 7E FE 80
0F FE 20 0B 25 05 1B 05
10 61 1A 1B 63 0A 2A 2A
11 0A 0A 0A 03 0B 03 0A 1A
12 FA 11 0A 3D 19 0B 65
13 01 0B 0A 0B 0B 0B 0B
14 0A 0B 7E 4F 17 81 4F
15 0A 0A 0A 0A 0A 0A 0A
16 1F 1F 1F 1F 1F 1F 1F
17 0A 0A 0A 0A 0A 0A 0A
18 0A 0A 0A 0A 0A 0A 0A
19 0A 0A 0A 0A 0A 0A 0A
20 0A 0A 0A 0A 0A 0A 0A
21 0A 0A 0A 0A 0A 0A 0A
22 0A 0A 0A 0A 0A 0A 0A
23 0A 0A 0A 0A 0A 0A 0A
24 0A 0A 0A 0A 0A 0A 0A
25 0A 0A 0A 0A 0A 0A 0A
26 0A 0A 0A 0A 0A 0A 0A
27 0A 0A 0A 0A 0A 0A 0A
28 0A 0A 0A 0A 0A 0A 0A
29 0A 0A 0A 0A 0A 0A 0A
30 0A 0A 0A 0A 0A 0A 0A
31 0A 0A 0A 0A 0A 0A 0A
32 0A 0A 0A 0A 0A 0A 0A
33 0A 0A 0A 0A 0A 0A 0A
34 0A 0A 0A 0A 0A 0A 0A
35 0A 0A 0A 0A 0A 0A 0A
36 0A 0A 0A 0A 0A 0A 0A
37 0A 0A 0A 0A 0A 0A 0A
38 0A 0A 0A 0A 0A 0A 0A
39 0A 0A 0A 0A 0A 0A 0A
40 0A 0A 0A 0A 0A 0A 0A
41 0A 0A 0A 0A 0A 0A 0A
42 0A 0A 0A 0A 0A 0A 0A
43 0A 0A 0A 0A 0A 0A 0A
44 0A 0A 0A 0A 0A 0A 0A
45 0A 0A 0A 0A 0A 0A 0A
46 0A 0A 0A 0A 0A 0A 0A
47 0A 0A 0A 0A 0A 0A 0A
48 0A 0A 0A 0A 0A 0A 0A
49 0A 0A 0A 0A 0A 0A 0A
50 0A 0A 0A 0A 0A 0A 0A
51 0A 0A 0A 0A 0A 0A 0A
52 0A 0A 0A 0A 0A 0A 0A
53 0A 0A 0A 0A 0A 0A 0A
54 0A 0A 0A 0A 0A 0A 0A
55 0A 0A 0A 0A 0A 0A 0A
56 0A 0A 0A 0A 0A 0A 0A
57 0A 0A 0A 0A 0A 0A 0A
58 0A 0A 0A 0A 0A 0A 0A
59 0A 0A 0A 0A 0A 0A 0A
60 0A 0A 0A 0A 0A 0A 0A
61 0A 0A 0A 0A 0A 0A 0A
62 0A 0A 0A 0A 0A 0A 0A
63 0A 0A 0A 0A 0A 0A 0A
64 0A 0A 0A 0A 0A 0A 0A
65 0A 0A 0A 0A 0A 0A 0A
66 0A 0A 0A 0A 0A 0A 0A
67 0A 0A 0A 0A 0A 0A 0A
68 0A 0A 0A 0A 0A 0A 0A
69 0A 0A 0A 0A 0A 0A 0A
70 0A 0A 0A 0A 0A 0A 0A
71 0A 0A 0A 0A 0A 0A 0A
72 0A 0A 0A 0A 0A 0A 0A
73 0A 0A 0A 0A 0A 0A 0A
74 0A 0A 0A 0A 0A 0A 0A
75 0A 0A 0A 0A 0A 0A 0A
76 0A 0A 0A 0A 0A 0A 0A
77 0A 0A 0A 0A 0A 0A 0A
78 0A 0A 0A 0A 0A 0A 0A
79 0A 0A 0A 0A 0A 0A 0A
80 0A 0A 0A 0A 0A 0A 0A
81 0A 0A 0A 0A 0A 0A 0A
82 0A 0A 0A 0A 0A 0A 0A
83 0A 0A 0A 0A 0A 0A 0A
84 0A 0A 0A 0A 0A 0A 0A
85 0A 0A 0A 0A 0A 0A 0A
86 0A 0A 0A 0A 0A 0A 0A
87 0A 0A 0A 0A 0A 0A 0A
88 0A 0A 0A 0A 0A 0A 0A
89 0A 0A 0A 0A 0A 0A 0A
90 0A 0A 0A 0A 0A 0A 0A
91 0A 0A 0A 0A 0A 0A 0A
92 0A 0A 0A 0A 0A 0A 0A
93 0A 0A 0A 0A 0A 0A 0A
94 0A 0A 0A 0A 0A 0A 0A
95 0A 0A 0A 0A 0A 0A 0A
96 0A 0A 0A 0A 0A 0A 0A
97 0A 0A 0A 0A 0A 0A 0A
98 0A 0A 0A 0A 0A 0A 0A
99 0A 0A 0A 0A 0A 0A 0A

```

EXPERT SYSTEMS

Part 3: David Howland shows how you can turn your computer into an 'expert'.

It is reassuring to know that most of us, in our own right, are experts. If you are working, and if you've been doing that job for some time, then you are an expert. Anyone starting to do the same job as you should regard you as an expert, and they should, if they have any sense, be tapping your experience to enable them to become experts too. And for those who are too young to be experts, then you are probably training to become skilled in one or more areas: you are the experts of the future.

If you've been following this series, then you are learning to become an expert in expert systems. At present, I hope that you regard me as an expert in this area, because I am able to pass on to you experience and knowledge to enable you to gain some expertise in this area. But, expert systems are a long way from my principle area of expertise, and there are now many expert system professionals here when I should learn a great deal.

All of which goes to show that expertise is relative: someone who knows a little about a subject will appear an expert to someone who knows nothing at all about that area. Experts should devote less learning more about their subject to become better experts. This is even true to 'high-powered' experts, rules and procedures will change, knowledge in any field is constantly expanding, and all experts, to maintain their expertise, must learn these new rules and knowledge.

The same is also true of computer expert systems. With the help of a human expert they must be capable of learning new rules and knowledge to become better experts, and adjust to changing circumstances. How expert systems can learn is the subject of this, the third part of my expert system series.

Tools

One of the principle ways in which we learn is simply to take in tools. Experts give us facts through books, TV, newspapers,

(and IT Computing, of course!) etc, as well as fact-to-fact or teachers, lecturers and tutors. Given facts in this way, our brain then files them away for future use.

In the expert systems I gave you last month, I gave those programs facts, as well as the

means of storing and recalling those facts. I 'taught' the computer these facts by including that information in DATA lines in the BASIC programs. These expert systems could be taught more tools through a programmer adding more DATA lines containing



Fig. 3 Learning Example in Serial Decision-making

```

1000 REM ***** Serial Expert System *****
1010 REM ***** Learning New Rules *****
1020 REM ***** (See Fig. 3) *****
1030 REM *****
1040 REM ***** Serial Decision-making *****
1050 REM ***** (See Fig. 3) *****
1060 REM *****
1070 REM ***** Serial Decision-making *****
1080 REM ***** (See Fig. 3) *****
1090 REM *****
1100 REM ***** Serial Decision-making *****
1110 REM ***** (See Fig. 3) *****
1120 REM *****
1130 REM ***** Serial Decision-making *****
1140 REM ***** (See Fig. 3) *****
1150 REM *****
1160 REM ***** Serial Decision-making *****
1170 REM ***** (See Fig. 3) *****
1180 REM *****
1190 REM ***** Serial Decision-making *****
1200 REM ***** (See Fig. 3) *****
1210 REM *****
1220 REM ***** Serial Decision-making *****
1230 REM ***** (See Fig. 3) *****
1240 REM *****
1250 REM ***** Serial Decision-making *****
1260 REM ***** (See Fig. 3) *****
1270 REM *****
1280 REM ***** Serial Decision-making *****
1290 REM ***** (See Fig. 3) *****
1300 REM *****
1310 REM ***** Serial Decision-making *****
1320 REM ***** (See Fig. 3) *****
1330 REM *****
1340 REM ***** Serial Decision-making *****
1350 REM ***** (See Fig. 3) *****
1360 REM *****
1370 REM ***** Serial Decision-making *****
1380 REM ***** (See Fig. 3) *****
1390 REM *****
1400 REM ***** Serial Decision-making *****
1410 REM ***** (See Fig. 3) *****
1420 REM *****
1430 REM ***** Serial Decision-making *****
1440 REM ***** (See Fig. 3) *****
1450 REM *****
1460 REM ***** Serial Decision-making *****
1470 REM ***** (See Fig. 3) *****
1480 REM *****
1490 REM ***** Serial Decision-making *****
1500 REM ***** (See Fig. 3) *****
1510 REM *****
1520 REM ***** Serial Decision-making *****
1530 REM ***** (See Fig. 3) *****
1540 REM *****
1550 REM ***** Serial Decision-making *****
1560 REM ***** (See Fig. 3) *****
1570 REM *****
1580 REM ***** Serial Decision-making *****
1590 REM ***** (See Fig. 3) *****
1600 REM *****
1610 REM ***** Serial Decision-making *****
1620 REM ***** (See Fig. 3) *****
1630 REM *****
1640 REM ***** Serial Decision-making *****
1650 REM ***** (See Fig. 3) *****
1660 REM *****
1670 REM ***** Serial Decision-making *****
1680 REM ***** (See Fig. 3) *****
1690 REM *****
1700 REM ***** Serial Decision-making *****
1710 REM ***** (See Fig. 3) *****
1720 REM *****
1730 REM ***** Serial Decision-making *****
1740 REM ***** (See Fig. 3) *****
1750 REM *****
1760 REM ***** Serial Decision-making *****
1770 REM ***** (See Fig. 3) *****
1780 REM *****
1790 REM ***** Serial Decision-making *****
1800 REM ***** (See Fig. 3) *****
1810 REM *****
1820 REM ***** Serial Decision-making *****
1830 REM ***** (See Fig. 3) *****
1840 REM *****
1850 REM ***** Serial Decision-making *****
1860 REM ***** (See Fig. 3) *****
1870 REM *****
1880 REM ***** Serial Decision-making *****
1890 REM ***** (See Fig. 3) *****
1900 REM *****
1910 REM ***** Serial Decision-making *****
1920 REM ***** (See Fig. 3) *****
1930 REM *****
1940 REM ***** Serial Decision-making *****
1950 REM ***** (See Fig. 3) *****
1960 REM *****
1970 REM ***** Serial Decision-making *****
1980 REM ***** (See Fig. 3) *****
1990 REM *****
2000 REM ***** Serial Decision-making *****
2010 REM ***** (See Fig. 3) *****
2020 REM *****
2030 REM ***** Serial Decision-making *****
2040 REM ***** (See Fig. 3) *****
2050 REM *****
2060 REM ***** Serial Decision-making *****
2070 REM ***** (See Fig. 3) *****
2080 REM *****
2090 REM ***** Serial Decision-making *****
2100 REM ***** (See Fig. 3) *****
2110 REM *****
2120 REM ***** Serial Decision-making *****
2130 REM ***** (See Fig. 3) *****
2140 REM *****
2150 REM ***** Serial Decision-making *****
2160 REM ***** (See Fig. 3) *****
2170 REM *****
2180 REM ***** Serial Decision-making *****
2190 REM ***** (See Fig. 3) *****
2200 REM *****
2210 REM ***** Serial Decision-making *****
2220 REM ***** (See Fig. 3) *****
2230 REM *****
2240 REM ***** Serial Decision-making *****
2250 REM ***** (See Fig. 3) *****
2260 REM *****
2270 REM ***** Serial Decision-making *****
2280 REM ***** (See Fig. 3) *****
2290 REM *****
2300 REM ***** Serial Decision-making *****
2310 REM ***** (See Fig. 3) *****
2320 REM *****
2330 REM ***** Serial Decision-making *****
2340 REM ***** (See Fig. 3) *****
2350 REM *****
2360 REM ***** Serial Decision-making *****
2370 REM ***** (See Fig. 3) *****
2380 REM *****
2390 REM ***** Serial Decision-making *****
2400 REM ***** (See Fig. 3) *****
2410 REM *****
2420 REM ***** Serial Decision-making *****
2430 REM ***** (See Fig. 3) *****
2440 REM *****
2450 REM ***** Serial Decision-making *****
2460 REM ***** (See Fig. 3) *****
2470 REM *****
2480 REM ***** Serial Decision-making *****
2490 REM ***** (See Fig. 3) *****
2500 REM *****
2510 REM ***** Serial Decision-making *****
2520 REM ***** (See Fig. 3) *****
2530 REM *****
2540 REM ***** Serial Decision-making *****
2550 REM ***** (See Fig. 3) *****
2560 REM *****
2570 REM ***** Serial Decision-making *****
2580 REM ***** (See Fig. 3) *****
2590 REM *****
2600 REM ***** Serial Decision-making *****
2610 REM ***** (See Fig. 3) *****
2620 REM *****
2630 REM ***** Serial Decision-making *****
2640 REM ***** (See Fig. 3) *****
2650 REM *****
2660 REM ***** Serial Decision-making *****
2670 REM ***** (See Fig. 3) *****
2680 REM *****
2690 REM ***** Serial Decision-making *****
2700 REM ***** (See Fig. 3) *****
2710 REM *****
2720 REM ***** Serial Decision-making *****
2730 REM ***** (See Fig. 3) *****
2740 REM *****
2750 REM ***** Serial Decision-making *****
2760 REM ***** (See Fig. 3) *****
2770 REM *****
2780 REM ***** Serial Decision-making *****
2790 REM ***** (See Fig. 3) *****
2800 REM *****
2810 REM ***** Serial Decision-making *****
2820 REM ***** (See Fig. 3) *****
2830 REM *****
2840 REM ***** Serial Decision-making *****
2850 REM ***** (See Fig. 3) *****
2860 REM *****
2870 REM ***** Serial Decision-making *****
2880 REM ***** (See Fig. 3) *****
2890 REM *****
2900 REM ***** Serial Decision-making *****
2910 REM ***** (See Fig. 3) *****
2920 REM *****
2930 REM ***** Serial Decision-making *****
2940 REM ***** (See Fig. 3) *****
2950 REM *****
2960 REM ***** Serial Decision-making *****
2970 REM ***** (See Fig. 3) *****
2980 REM *****
2990 REM ***** Serial Decision-making *****
3000 REM ***** (See Fig. 3) *****
3010 REM *****
3020 REM ***** Serial Decision-making *****
3030 REM ***** (See Fig. 3) *****
3040 REM *****
3050 REM ***** Serial Decision-making *****
3060 REM ***** (See Fig. 3) *****
3070 REM *****
3080 REM ***** Serial Decision-making *****
3090 REM ***** (See Fig. 3) *****
3100 REM *****
3110 REM ***** Serial Decision-making *****
3120 REM ***** (See Fig. 3) *****
3130 REM *****
3140 REM ***** Serial Decision-making *****
3150 REM ***** (See Fig. 3) *****
3160 REM *****
3170 REM ***** Serial Decision-making *****
3180 REM ***** (See Fig. 3) *****
3190 REM *****
3200 REM ***** Serial Decision-making *****
3210 REM ***** (See Fig. 3) *****
3220 REM *****
3230 REM ***** Serial Decision-making *****
3240 REM ***** (See Fig. 3) *****
3250 REM *****
3260 REM ***** Serial Decision-making *****
3270 REM ***** (See Fig. 3) *****
3280 REM *****
3290 REM ***** Serial Decision-making *****
3300 REM ***** (See Fig. 3) *****
3310 REM *****
3320 REM ***** Serial Decision-making *****
3330 REM ***** (See Fig. 3) *****
3340 REM *****
3350 REM ***** Serial Decision-making *****
3360 REM ***** (See Fig. 3) *****
3370 REM *****
3380 REM ***** Serial Decision-making *****
3390 REM ***** (See Fig. 3) *****
3400 REM *****
3410 REM ***** Serial Decision-making *****
3420 REM ***** (See Fig. 3) *****
3430 REM *****
3440 REM ***** Serial Decision-making *****
3450 REM ***** (See Fig. 3) *****
3460 REM *****
3470 REM ***** Serial Decision-making *****
3480 REM ***** (See Fig. 3) *****
3490 REM *****
3500 REM ***** Serial Decision-making *****
3510 REM ***** (See Fig. 3) *****
3520 REM *****
3530 REM ***** Serial Decision-making *****
3540 REM ***** (See Fig. 3) *****
3550 REM *****
3560 REM ***** Serial Decision-making *****
3570 REM ***** (See Fig. 3) *****
3580 REM *****
3590 REM ***** Serial Decision-making *****
3600 REM ***** (See Fig. 3) *****
3610 REM *****
3620 REM ***** Serial Decision-making *****
3630 REM ***** (See Fig. 3) *****
3640 REM *****
3650 REM ***** Serial Decision-making *****
3660 REM ***** (See Fig. 3) *****
3670 REM *****
3680 REM ***** Serial Decision-making *****
3690 REM ***** (See Fig. 3) *****
3700 REM *****
3710 REM ***** Serial Decision-making *****
3720 REM ***** (See Fig. 3) *****
3730 REM *****
3740 REM ***** Serial Decision-making *****
3750 REM ***** (See Fig. 3) *****
3760 REM *****
3770 REM ***** Serial Decision-making *****
3780 REM ***** (See Fig. 3) *****
3790 REM *****
3800 REM ***** Serial Decision-making *****
3810 REM ***** (See Fig. 3) *****
3820 REM *****
3830 REM ***** Serial Decision-making *****
3840 REM ***** (See Fig. 3) *****
3850 REM *****
3860 REM ***** Serial Decision-making *****
3870 REM ***** (See Fig. 3) *****
3880 REM *****
3890 REM ***** Serial Decision-making *****
3900 REM ***** (See Fig. 3) *****
3910 REM *****
3920 REM ***** Serial Decision-making *****
3930 REM ***** (See Fig. 3) *****
3940 REM *****
3950 REM ***** Serial Decision-making *****
3960 REM ***** (See Fig. 3) *****
3970 REM *****
3980 REM ***** Serial Decision-making *****
3990 REM ***** (See Fig. 3) *****
4000 REM *****

```

information and rules. This is not particularly convenient. It assumes that the expert teaching the system new knowledge is also a programmer. Not only has he to type in data lines containing facts, he has to understand how the program works, and the structure of the rules. He may even have to modify or update some of the existing rules or some program lines to allow the system to work with the new rules and knowledge.

So, why have a computer if it can't do things automatically for you. And that's what the first two programs this month will do. One is serial decision (Fig. 3) and one for parallel decisions (Fig. 4). Use last month, they were written on the QL, but tested on both Spectrum and QL. They are a little light on the interrogation side: the theory and practice of that was covered last month. These programs aim to show you how to teach facts and rules to an expert system.

Let's start with the serial decision maker in fig. 4. Type in the listing as shown if you have a QL. If you are using a

Spectrum, then enter line 3000 to 3160 and 4000 to the end.

All the expert systems this month deal with rules. The idea is that you specify a set of features, and the expert system will select a computer for you which meets your specifications.

For the serial decision examples, a small knowledge base is included which allows you to select one of the Spectrum computers. The decision tree for this very small expert system is shown in fig. 2(b). When you RUN this program, the knowledge and rule arrays are created (line 180). The first array (lines 190 to 250) the questions and answers strings are entered into array q5 (line 240). Memory array 7 (line 250) is the rule array which contains the pointers to positions in array q5 depending on 'yes' or 'no' answers. A zero in the rule array indicates that an answer has been reached.

Rules

The rule and knowledge arrays are dimensioned to 62 rules and

questions/answers, to allow for a reasonable expansion to the seven included (lines 270 to 330 and fig. 3(b) is the program On Running, since the arrays are set up, the program's basic menu is shown, a choice of interrogating the knowledge base (5), expanding the knowledge base (2) or stopping (3) is given. Using the knowledge base follows much the same logic as last month's program. You give yes/no answers to each question until an answer is given, then you go back to the menu. Before you try teaching the knowledge base crafting new try using the system a few times.

Once you have the feel of the system (lines 70 on the main menu), apart from the introductory message (lines 620 to 640), the first part of adding to the knowledge base is just the using it. The purpose is to get to the appropriate part of the knowledge base that you wish to expand.

If the program in fig. 2(b) feels like an update/down line to you, then that's exactly what it is — a decision tree. And something like a real tree, this

one grows by spreading new branches from its tips. Once you teach a tip (i.e. an answer), you can replace that answer with a question. One answer to that question should be the answer you have just displayed; the other answer, for this expert system, another name of a computer. So the question should be carefully chosen (by you, the expert) to discriminate between the old answer (computer system) and the new one.

Fig. 2a gives you one example. Start from the main menu, press option 2, then select 'Y/N' to the three questions, four and up being told that the answer is a Spectrum 128, and the questions "Do you want to add to this point?" A 'Y' reply would take you back to the main menu (an escape route if you've taken the wrong branch); you should reply 'N'. So for we have a home computer with 128K of RAM, which does not have a built-in cassette deck. This fits a Spectrum 128. It's also true for a QL, and that's the extra answer that I want to insert at this point. So what yes/no question will discriminate between a QL and Spectrum 128? We choose "Does

it have built-in microdrive?"

So when the message "Enter new question" appears on the screen, type in the above question (with the question mark); then enter the new answer "QL" on request, and finally tell the computer whether the answer given in the yes or no response to the new question. The program then sorts out the rule (old knowledge base (lines 1100 to 1160) to insert the new question and two answers in the right place, when all that is done is an input the program returns you to the main menu.

Knowledge base

In the same way you can add further knowledge and rules to the system, up to a maximum of 80. If you had answered 'N' to the last question "Is it a home computer?", you would get the response "IBM PC". Now, while that isn't too far from the truth, I'm sure that those computer experts amongst you would want to add a great deal more!

Once you've expanded the knowledge base, the last thing you'll want to do is load all that work when you turn off your computer. So, when you select 'stop' from the main menu, you

get the option to save the knowledge base. If you say yes, then, for the first time, there is a difference between QL and Spectrum versions.

On the Spectrum, (lines 4030 to 4035) the program is simply saved (with all its data), and set to restore the next time you load, without re-dimensioning the variables.

You can do that on the QL, so the routine (lines 3030 to 3100) is used instead. Place a carriage in `move%_before` saying 'Y' to "Do you want to save the knowledge base?" A file called `expert_base` is saved as `move%_data` on the data. Next time you want to run this knowledge base on the QL, load the program, and type in LD instead of RUN. This activates the routine (lines 4030 to 4100) to load in the data from `expert_base`, before you use the program.

It may have occurred to you in using this program that a serial approach is probably not the best way to select a computer. You use more likely to have a list of features that you want, and match those against various computer profiles for the best fit — in other words, use a parallel approach to decide on your

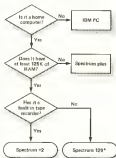


Fig. 2a: Decision tree for serial learning example

EXPERT SYSTEMS



Fig. 2b: Adding a new branch to the tree



preferred computer. So, the next preferred program does just that.

Fig 3 contains the listing for this program. The program can be entered as listed on the GL. On the Spectrum, do not type in lines 1040 to 1070 and 1220 to 1250. Also on the Spectrum, the string ending & in lines 1240 and 1250 should be +.

Just to remind you, in parallel decisions you are asked a series of questions and your replies are coded 1 and 0, respectively. This string of zeros and ones is then matched against the zeros and ones in the rule base for all possible answers. If a perfect match is found, then the computer should report this, if the match isn't perfect, the computer should search for the closest match.

In running this program,

you'll see it has several features the same as the serial example. The thing is exactly the same. By selecting 1, you get to try out the system, which starts with only two Simulac computers in its knowledge base though there is room for fifty.

This expert system will ask fifteen questions which I have decided to differentiate between various computers. Since 200 to 340, if you enter your computer, experts can think of more more, but mine should come as a suitable example. So, you will always need 49 '0' or 0 replies to elicit a response.

In teaching the expert system some new computer (option 2), you are first asked to give the name of the new computer. A check is made to see that the name doesn't already exist (lines

600 to 640). Then it will ask you to enter your yes/no replies to all fifteen questions (lines 650 to 690). Checking to see that exactly the same profile doesn't already exist in the rule base (see 700). If this is satisfied, then your expert rules are stored in the rule array and the name array (see 720).

I list the serial example, the knowledge base can be stored when the stop option (0) is selected. Having found a matching work is exactly the same way as in the serial learning program example.

D.I.Y.

So far you've had to help your expert system every step of the way in the final example this month, well try and get the

Fig 3 Example Learning Program - Parallel Decisions

```

100 REMARK "Simplest Expert System"
110 REMARK "Learning new profiles"
120 REMARK "Knowledge base array"
130 REMARK
140 REMARK "New computer"
150 REMARK "Knowledge base"
160 REMARK
170 REMARK "Knowledge base"
180 REMARK "New computer"
190 REMARK "Knowledge base"
200 REMARK "New computer"
210 REMARK "Knowledge base"
220 REMARK "New computer"
230 REMARK "Knowledge base"
240 REMARK "New computer"
250 REMARK "Knowledge base"
260 REMARK "New computer"
270 REMARK "Knowledge base"
280 REMARK "New computer"
290 REMARK "Knowledge base"
300 REMARK "New computer"
310 REMARK "Knowledge base"
320 REMARK "New computer"
330 REMARK "Knowledge base"
340 REMARK "New computer"
350 REMARK "Knowledge base"
360 REMARK "New computer"
370 REMARK "Knowledge base"
380 REMARK "New computer"
390 REMARK "Knowledge base"
400 REMARK "New computer"
410 REMARK "Knowledge base"
420 REMARK "New computer"
430 REMARK "Knowledge base"
440 REMARK "New computer"
450 REMARK "Knowledge base"
460 REMARK "New computer"
470 REMARK "Knowledge base"
480 REMARK "New computer"
490 REMARK "Knowledge base"
500 REMARK "New computer"
510 REMARK "Knowledge base"
520 REMARK "New computer"
530 REMARK "Knowledge base"
540 REMARK "New computer"
550 REMARK "Knowledge base"
560 REMARK "New computer"
570 REMARK "Knowledge base"
580 REMARK "New computer"
590 REMARK "Knowledge base"
600 REMARK "New computer"
610 REMARK "Knowledge base"
620 REMARK "New computer"
630 REMARK "Knowledge base"
640 REMARK "New computer"
650 REMARK "Knowledge base"
660 REMARK "New computer"
670 REMARK "Knowledge base"
680 REMARK "New computer"
690 REMARK "Knowledge base"
700 REMARK "New computer"
710 REMARK "Knowledge base"
720 REMARK "New computer"
730 REMARK "Knowledge base"
740 REMARK "New computer"
750 REMARK "Knowledge base"
760 REMARK "New computer"
770 REMARK "Knowledge base"
780 REMARK "New computer"
790 REMARK "Knowledge base"
800 REMARK "New computer"
810 REMARK "Knowledge base"
820 REMARK "New computer"
830 REMARK "Knowledge base"
840 REMARK "New computer"
850 REMARK "Knowledge base"
860 REMARK "New computer"
870 REMARK "Knowledge base"
880 REMARK "New computer"
890 REMARK "Knowledge base"
900 REMARK "New computer"
910 REMARK "Knowledge base"
920 REMARK "New computer"
930 REMARK "Knowledge base"
940 REMARK "New computer"
950 REMARK "Knowledge base"
960 REMARK "New computer"
970 REMARK "Knowledge base"
980 REMARK "New computer"
990 REMARK "Knowledge base"

```

expert system to do a lot more work for itself. The program for this example is in fig 4. It again uses the parallel decision approach, but this time, it tries to develop the rule base so that you find it's very inefficient, but the program coding will give you an idea how the expert system will automatically adjust its rules — well, almost automatically. It still needs a little help from you.

To keep things fairly simple, there are just five answers in this self-learning program, with seven features. Line 180 dimension's seven arrays. As before, if you **q1** contain the names of the features and answers, respectively, the rules are now in feature array, **ans** is the answer array, **ans** is the rule base which the computer will create by self-learning. Naming array 'Y' is used in decision making. It keeps the score of each possible answer when comparing entered features against the rule base.

When you **RUN** the program you again get a menu with three options. Option 1 just tests the self-learning rule base for you. It sets up through each answer in turn, and enter the stored features from the correct rule array (1) then pass the computer to make a decision based on those features using the rule array (1) is determining (2). When you hit **END** the program, and asked the test option, you'll note that the rule array will always select "Spectrum 48K" — it still has a lot to learn!

You can start teaching the system to allow it to adjust the rule array either manually or automatically. Option 2 is the manual approach. You will be asked five times before returning to the main menu, to fill up one of the five answers, and provide the correct yes/no

response to each of the seven features. The computer will make a guess. If the guess is right, then the rule does not change, but if the guess is wrong, then the rules are altered, and the updated rule is displayed.

At this point, some explanation is required as to what is happening during adjustment of the self-learning rule array (1). That array is initially filled with zeros (line 490) that's why the answer was always the same on the very first test. When you enter the profile of any one computer system (lines 2000 to 2090) a string of ones and zeros is set up, representing your sequence of yes/no answers. In lines 2000 to 2090 every element in that string is compared with the corresponding rule (in array 1) for each possible answer. If a match is found, then the score (2) is incremented (line 2000), and the total for each possible answer stored in numeric array 'Y'. The highest score indicates the best fit of the entered profile and the rule base, and this is how the answer is selected (line 2490).

If the computer's answer is wrong, then you tell it the correct answer (lines 2420 to 2450), and the self-learning rule array is adjusted. Each rule element in the correct answer is incremented if a 'yes' response was given to a particular feature (lines 2770 to 2790). For the answer given incorrectly (and other possible answers which scored the same as the incorrect answer) the rule element is decremented by one. For each feature given 'yes' is the profile supplied by you (line 2710 to 2760).

All that may seem rather complicated, but if you work through it a few times, I hope you will start to see the logic of what is going on. When the rule array is printed out on the screen, you'll get strings of

numbers, these are not just restricted to ones and zeros, you'll get higher numbers, and negative numbers too. Typing in feature time (and time again is rather boring, so there is an automatic feature entry to speed things up, option 3 from the main menu. It picks a possible answer at random (lines 2100 to 2170 — please note the different structures in the 2170 depending on which machine, Spectrum or GC, you are using), and feeds the yes/no answers automatically into string 1. The computer will then select an answer as before, using its determining rule array (1). As the computer has found the correct answer, has it selected the best? It tells you if the answer is right or wrong, and updates the rule array if it is wrong.

Whether you use the manual or automatic profile entry procedure you may be surprised by the number of times this expert system has to 'proceed' to pass the test (option 1) and get the answer right every time for every time. In teaching an expert system, it is always much easier to be able to give the computer the correct rules to enter into its rule base. But any modification of the rules may well be necessary sometimes, for example, a manufacturer may change some of the features of one or more of his computers. Self learning may be the easiest way a user not familiar with the inner workings of the system, can update the rule base to adjust to these changes.

So let us now consider all our examples as capable of dealing with yes/no responses only. What happens if the user doesn't know the answer? And what happens if the response cannot be as simple as yes or no? The mathematics of our expert systems get a little more complex as we consider these items in the next part of this series.

Fig 4. Example of Rule Learning/Modification

```

100 REMARK ***** Expert System *****
200 REMARK: Learning new rules
300 REMARK: no character
400 REMARK:
500 REMARK: State of Learning
600 REMARK: Feature array, ans
700 REMARK:
800 REMARK:
900 REMARK: Initialization
1000 REMARK: Feature array, ans, ans, ans, ans, ans, ans, ans
1100 REMARK:
1200 REMARK:
1300 REMARK:
1400 REMARK:
1500 REMARK:
1600 REMARK:
1700 REMARK:
1800 REMARK:
1900 REMARK:
2000 REMARK: Feature array, ans, ans, ans, ans, ans, ans, ans
2100 REMARK:
2200 REMARK:
2300 REMARK:
2400 REMARK:
2500 REMARK:
2600 REMARK:
2700 REMARK:
2800 REMARK:
2900 REMARK:
3000 REMARK:
3100 REMARK:
3200 REMARK:
3300 REMARK:
3400 REMARK:
3500 REMARK:
3600 REMARK:
3700 REMARK:
3800 REMARK:
3900 REMARK:
4000 REMARK:
4100 REMARK:
4200 REMARK:
4300 REMARK:
4400 REMARK:
4500 REMARK:
4600 REMARK:
4700 REMARK:
4800 REMARK:
4900 REMARK:
5000 REMARK:
5100 REMARK:
5200 REMARK:
5300 REMARK:
5400 REMARK:
5500 REMARK:
5600 REMARK:
5700 REMARK:
5800 REMARK:
5900 REMARK:
6000 REMARK:
6100 REMARK:
6200 REMARK:
6300 REMARK:
6400 REMARK:
6500 REMARK:
6600 REMARK:
6700 REMARK:
6800 REMARK:
6900 REMARK:
7000 REMARK:
7100 REMARK:
7200 REMARK:
7300 REMARK:
7400 REMARK:
7500 REMARK:
7600 REMARK:
7700 REMARK:
7800 REMARK:
7900 REMARK:
8000 REMARK:
8100 REMARK:
8200 REMARK:
8300 REMARK:
8400 REMARK:
8500 REMARK:
8600 REMARK:
8700 REMARK:
8800 REMARK:
8900 REMARK:
9000 REMARK:
9100 REMARK:
9200 REMARK:
9300 REMARK:
9400 REMARK:
9500 REMARK:
9600 REMARK:
9700 REMARK:
9800 REMARK:
9900 REMARK:
10000 REMARK:

```

[illegible]

CONCRETE

the disciple

COMPETITION

Your chance to win the most exciting Spectrum add-on to appear for a very long time

The Disciple is an immensely versatile multi-purpose interface that functions as a disk, printer, dual joystick and network.

Interface for all computers in the Spectrum range.

Conveniently designed to replicate the Interface 1, the Disciple uses the same rated connector leaving the Spectrum's input and output ports free to use. With features such as strophol buttons allowing up to 16 individual bits to be added when working via double density the Disciple has found favour among serious Spectrum users since its recent introduction.

We are offering two Disciples in this competition. All you have to do is select the right answers to the questions below and ring

them on the entry coupon. Give to the person who has issued in John Watts' Disciple review in last month's ZX and perhaps on this page. Or to avoid awkward questions perhaps Rockport Products will send you a Disciple if you ring them on the number to be found elsewhere in this issue.

The first two correct entries picked out of the hat will receive a Disciple.

The Questions

1. How many 48K commercial games could save on a double density disk using the Disciple with an 88 track disk drive?

- a) 1
- b) 12
- c) 16

2. Loading a file from disk with the Disciple is approximately

- a) 20 times
 - b) 40 times
 - c) 60 times
- faster than loading the same file from tape.

3. To copy a screen from a commercial program to your printer with the Disciple you simply:

- a) Press Cap Shift and the strophol button
- b) Enter 12345
- c) Press F and Strohol button

4. Which statement is incorrect?

- a) The Disciple uses hook codes similar to Interface 1
- b) To transfer most commercial programs from tape at 1500 bauds to disk press the strophol button

c) The joystick ports are Seraphim Interface II and Proxi

The competition is open to all ZX readers residing in players of Argus Specialist Publications, Cress Web, Rockport Products, and Mass System Technology. The editors' decision is final and no correspondence can be entered into.

Send your entries to: Disciple Competition, ZX Computing, No 1 Golden Square, London W1R 3AB.

Please remember to put your address on your entry envelope.

The closing date for entries is May 31 1987.



Disciple Competition

Please ring the appropriate letter:

- | | | | |
|---|---|---|---|
| 1 | A | B | C |
| 2 | A | B | C |
| 3 | A | B | C |
| 4 | A | B | C |

Name _____

Address _____

Send your entries to Disciple Competition, ZX Computing, No 1 Golden Square, London W1R 3AB.



A game of ups and downs from Quickliva

Elevator Action
Quickliva
£8.95

After a series of pretty unexciting titles it looks like Quickliva are finally starting to show signs of life again. Their recent Glider Rider wasn't quite plying but it did have some nice touches, and now Quickliva have joined the rest of the industry by turning to consumers of successful arcade games.

I've never played the original (the 80 version of Elevator Action as I don't know how faithful a conversion this is, but after it was sort of a dud but out to be simple, if underwhelming, fun).

The game is set inside a tall building where all the floors are interconnected, both horizontally and vertically by numerous lifts and escalators. On each floor there are a number of doors, painted either red or yellow and behind the red doors are test papers that you have to collect before you can leave the

building through the basement. You enter through the roof and head downwards, though you might find that you have to do a bit of shuffling up and down as you go after all the papers.

While you own several service chaps who go up to compare shoppers in the street and offer them bags full of official secrets, the owners of these papers have sensibly charged for them to be protected by armed guards who patrol each floor and some peeping out of the locked, yellow doors of the legal department's apartment. When you come close to face with the guards you can either run for the nearest lift or, if there isn't one around you can kick or fire at them with your own gun. The trouble here is that the guards often manage to get off a few shots of their own before they bite the dust, but if you fire on a lift so it you can jump or slide out of the line of fire.

At the top of the building there's just the one lift, and no guards to avoid, but the further down you go the more complicated the interconnections become and the more guards there are wandering around. Fortunately, once you're in a lift you can control its movements with the joystick, and this allows you to go straight to the doors with the red doors, rather than having to

climb your way through each floor individually.

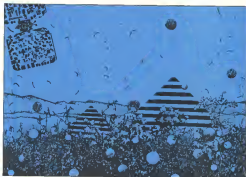
For some reason, when you get to the end of the bottom of the building there are some weird bonus clothes, but these don't affect the game itself and you then go on to the start of the next game.

It's not the fastest, most action-packed arcade game I've ever come across but for some reason it does seem to be a bit addictive (possibly because it is so simple for once — it makes a change not to have to read a great wodge of instructions before getting into the game). It would be even better if the layout of the building changed at the end of each section, instead of using the same building with more doors and guards — as it stands I think the game's life is probably limited by the number of times that you're prepared to go over the same ground. The game is also a bit overpriced at £8.95, but if you want your collection of arcade conversions to be complete then Elevator Action is worth a look.



ELEVATOR ACTION





RANDOM LANDSCAPE GRAPHICS

Alan Davis presents a technique for creating random adventure backgrounds

Despite the arrival of the 128K Spectrum, many of us are still fondling along with our old 48K machines, and are generally fairly content I suspect. Speaking purely personally, and confining my remarks to the adventure game genre, I don't believe that the potential of 48K has been exhausted yet (or anywhere near it). But having said that, I must admit that there are aspects of adventure programming where the limitations of 48K are very evident. I'm talking about graphics, of course. They gobble memory mindlessly, and the adventure writer is always on the look-out for programming methods which can help to ease this problem.

One possible solution is to use some method of "streamlining"

the production of at least part of each graphic illustration, perhaps by using a small number of subroutines to set up a limited number of backgrounds onto which different shapes can be superimposed. In this way you could arrange for each location in your adventure to have some sort of illustration — though there's a price to pay of course. Inevitably there'd be a certain sameness about them, even though they might differ in detail.

One program which used something like this technique was the tremendously successful "Archon", which has been given a fine lease of life recently by being released as a budget game. Again, speaking purely for myself, I guess I felt that the game itself was rightly less interesting than good top pudding, but I must admit that I do find the illustrations highly atmospheric. So the method clearly can work, and yet oddly enough, nothing quite like it seems to have been attempted since. Shall we have a go?

Background work

I don't propose to do more in this article than to tackle the job of creating backgrounds. Because this game generated quite enough interesting little problems to keep us occupied for a month! At the outset I decided it would be a rather neat idea to let the Spectrum random generator take over the creation of the picture, and began (as I so often do) by trying out a few simple ideas in BASIC. Listing 1 will show you the sort of thing I was aiming for, and if you type it in and run it you'll get some idea of the ideas that can be achieved even with these simple routines (it's obviously slow of course, but that doesn't matter for the moment). The screen shot which should be lurking somewhere about these pages shows an application of the same principles, extended to use the entire screen, and with some graphics for buildings superimposed to show how the ideas could be used in practice.

As you'll see, this little

program uses the RND function to determine the operation of the various FUD and GEAR routines, and its capable of producing some rather effective mountains, rippled water, clouds, and so on. But what makes the system of potential interest to the adventure programmer is the fact that as you know, the RND function doesn't actually produce random numbers at all, it produces pseudo-random numbers. This means that we get a strictly predetermined sequence which repeats itself only after 65536 numbers have been generated. The starting point of the sequence is set by using RANDOMIZE n, and you'll see this used on line 1000. There's one more point to keep in mind: the value of n is increased so that the sequence begins next time from a different point and produces a different picture. The changes are most noticeable in the mountain profiles, mainly because these are the most complex shapes (being the ones I spent the most time tinkering with).

I think you'll see now what I'm getting at. If we let n represent the location number in our adventure, and provide every call to the graphics subroutine by RANDOMIZE n, we'll always get the same set of background graphics for the same location, yet the background for each location will be different. Of course you could go much further than I've done here. Other subroutines can be added (and called) continuously on the value of n) just by changing the ink and paper the title in the middle distance can be made to turn into a marsh, an ice sheet, a sandy desert, or whatever.

Speed trap

Unfortunately we still have the problem of speed — or rather lack of it. Any, you say. No problem. FUD, GEAR, and RND can easily be coded with in-machine code by making suitable calls to the ROM. Not an easy job with an assembler and we're home and dry. Well, don't let me stop you, but I'm afraid you'll find the exercise very disappointing. The short story (there always is one, isn't there?) is that the correct Spectrum ROM routine is **NOT** because also does using it from machine code doesn't actually make much difference. We'd better think again.

Slightly speaking, we don't actually need all the possible permutations in the RND sequence for the job in hand. Nor do we need 65536 different pictures, come to that. Wouldn't 256 be enough? Well then, why don't we simply store 256 bytes

Listing 1

```

1 DEF FN STAT=INT (RANDOM)
2 NUMBER 0: FOR M=1 TO 40: GO TO 1000
3 END
4 END
5 DEFN RANDOM RECTANGLE AND TALL WITH BACKGROUND THICKEN
6 END
7 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
8 RANDOMIZE L: GOTO 40: M, N
9 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
10 RANDOMIZE L: GOTO 40: M, N
11 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
12 RANDOMIZE L: GOTO 40: M, N
13 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
14 RANDOMIZE L: GOTO 40: M, N
15 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
16 RANDOMIZE L: GOTO 40: M, N
17 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
18 RANDOMIZE L: GOTO 40: M, N
19 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
20 RANDOMIZE L: GOTO 40: M, N
21 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
22 RANDOMIZE L: GOTO 40: M, N
23 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
24 RANDOMIZE L: GOTO 40: M, N
25 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
26 RANDOMIZE L: GOTO 40: M, N
27 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
28 RANDOMIZE L: GOTO 40: M, N
29 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
30 RANDOMIZE L: GOTO 40: M, N
31 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
32 RANDOMIZE L: GOTO 40: M, N
33 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
34 RANDOMIZE L: GOTO 40: M, N
35 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
36 RANDOMIZE L: GOTO 40: M, N
37 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
38 RANDOMIZE L: GOTO 40: M, N
39 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
40 RANDOMIZE L: GOTO 40: M, N
41 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
42 RANDOMIZE L: GOTO 40: M, N
43 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
44 RANDOMIZE L: GOTO 40: M, N
45 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
46 RANDOMIZE L: GOTO 40: M, N
47 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
48 RANDOMIZE L: GOTO 40: M, N
49 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
50 RANDOMIZE L: GOTO 40: M, N
51 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
52 RANDOMIZE L: GOTO 40: M, N
53 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
54 RANDOMIZE L: GOTO 40: M, N
55 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
56 RANDOMIZE L: GOTO 40: M, N
57 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
58 RANDOMIZE L: GOTO 40: M, N
59 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
60 RANDOMIZE L: GOTO 40: M, N
61 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
62 RANDOMIZE L: GOTO 40: M, N
63 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
64 RANDOMIZE L: GOTO 40: M, N
65 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
66 RANDOMIZE L: GOTO 40: M, N
67 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
68 RANDOMIZE L: GOTO 40: M, N
69 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
70 RANDOMIZE L: GOTO 40: M, N
71 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
72 RANDOMIZE L: GOTO 40: M, N
73 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
74 RANDOMIZE L: GOTO 40: M, N
75 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
76 RANDOMIZE L: GOTO 40: M, N
77 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
78 RANDOMIZE L: GOTO 40: M, N
79 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
80 RANDOMIZE L: GOTO 40: M, N
81 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
82 RANDOMIZE L: GOTO 40: M, N
83 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
84 RANDOMIZE L: GOTO 40: M, N
85 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
86 RANDOMIZE L: GOTO 40: M, N
87 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
88 RANDOMIZE L: GOTO 40: M, N
89 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
90 RANDOMIZE L: GOTO 40: M, N
91 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
92 RANDOMIZE L: GOTO 40: M, N
93 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
94 RANDOMIZE L: GOTO 40: M, N
95 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
96 RANDOMIZE L: GOTO 40: M, N
97 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
98 RANDOMIZE L: GOTO 40: M, N
99 FOR L=1 TO 100: FOR M=1 TO 100: FOR N=1 TO 100: PRINT AT L,17;"TIT"
100 RANDOMIZE L: GOTO 40: M, N

```

memory a table of 256 pseudo-random integers, and then extract our random numbers from the table on and when they're needed?

Let me explain in a little more detail. First we reserve 256 bytes

of memory (in fact I used address 84384 to 84999 inclusive) and fill each of those bytes with a random integer like this:

```
FOR I = 84714 TO 84999: FOR J = INT(RND*256): NEXT J
```



Just one background effect achieved with this listing



Now imagine that we have a pointer which can be moved along the table. Whenever we want a random number we just collect the entry in the table indicated by the current position of the pointer and move the pointer along one notch ready for the next random number collection (when the pointer gets to the end of the table we simply arrange for it to move back to the start again). This way we can get 256 different pointers according to the initial position of the pointer (I suppose you'll rightly point out that during the drawing of one complete profile the pointer will cycle many times through the table, repeating itself many times. Indeed it will — but oddly enough it doesn't matter much in practice. A test of regularity in things like the 'gross' or 'needs' spreadsheet will hardly be abusive anyway and with more complicated shapes like mountains the use of the random number by the graphic routines is complex enough to make almost completely the underlying repetitions. The technique would suit a mathematician — but it'll do for us.

Translation

We have to do a lot of things in a position to attempt what we set out to do — that is, to 'translate' the slow bits of our original BASIC program into machine code. In order to do so we need 4 essential subroutines, and these are given in Listing 2. The first two take the well-known methods of using the ROM routines for drawing a line and plotting a point to get the equivalent of PLOT as we load the C register with 'X', the B register with 'Y', and then CALL PLOT to get the equivalent of DRAW as we load the B and C registers similarly and then CALL DRAW.

The interesting stuff comes with the third routine (called SAND in Listing 2). I've translated that in some detail so that if I need to do here I explain how it's used. It produces the equivalent of INT (RND*256) using a table of 256 random integers as we discussed above. The routine is entered with the A register holding the value of 'x' and on return you'll find the A register holding INT (RND*256). In other words, it does exactly the same job as the defined function in Listing 1. Now I need to fill the table from BASIC. Well, of course as I described earlier — but since you've seen that the table can be saved along with the rest of the code as a single block, you control the starting point of the SAND sequence by PENDING SAND? (PENDING with the current location number in your spreadsheet — this is effectively the 'pointer' we were talking about before.

Listing 2

WISDOT (RANDOM) SUBROUTINE
IN SPECTRUM

THIS ROUTINE GETS WISDOT INTO A
AND PENDING CONTAINED

DATA: 1. RANDOM NO.

```

1000 1000
1010 1000
1020 1000
1030 1000
1040 1000
1050 1000
1060 1000
1070 1000
1080 1000
1090 1000
1100 1000
1110 1000
1120 1000
1130 1000
1140 1000
1150 1000
1160 1000
1170 1000
1180 1000
1190 1000
1200 1000
1210 1000
1220 1000
1230 1000
1240 1000
1250 1000
1260 1000
1270 1000
1280 1000
1290 1000
1300 1000
1310 1000
1320 1000
1330 1000
1340 1000
1350 1000
1360 1000
1370 1000
1380 1000
1390 1000
1400 1000
1410 1000
1420 1000
1430 1000
1440 1000
1450 1000
1460 1000
1470 1000
1480 1000
1490 1000
1500 1000
1510 1000
1520 1000
1530 1000
1540 1000
1550 1000
1560 1000
1570 1000
1580 1000
1590 1000
1600 1000
1610 1000
1620 1000
1630 1000
1640 1000
1650 1000
1660 1000
1670 1000
1680 1000
1690 1000
1700 1000
1710 1000
1720 1000
1730 1000
1740 1000
1750 1000
1760 1000
1770 1000
1780 1000
1790 1000
1800 1000
1810 1000
1820 1000
1830 1000
1840 1000
1850 1000
1860 1000
1870 1000
1880 1000
1890 1000
1900 1000
1910 1000
1920 1000
1930 1000
1940 1000
1950 1000
1960 1000
1970 1000
1980 1000
1990 1000
2000 1000

```

```

2010 1000
2020 1000
2030 1000
2040 1000
2050 1000
2060 1000
2070 1000
2080 1000
2090 1000
2100 1000
2110 1000
2120 1000
2130 1000
2140 1000
2150 1000
2160 1000
2170 1000
2180 1000
2190 1000
2200 1000
2210 1000
2220 1000
2230 1000
2240 1000
2250 1000
2260 1000
2270 1000
2280 1000
2290 1000
2300 1000
2310 1000
2320 1000
2330 1000
2340 1000
2350 1000
2360 1000
2370 1000
2380 1000
2390 1000
2400 1000
2410 1000
2420 1000
2430 1000
2440 1000
2450 1000
2460 1000
2470 1000
2480 1000
2490 1000
2500 1000
2510 1000
2520 1000
2530 1000
2540 1000
2550 1000
2560 1000
2570 1000
2580 1000
2590 1000
2600 1000
2610 1000
2620 1000
2630 1000
2640 1000
2650 1000
2660 1000
2670 1000
2680 1000
2690 1000
2700 1000
2710 1000
2720 1000
2730 1000
2740 1000
2750 1000
2760 1000
2770 1000
2780 1000
2790 1000
2800 1000
2810 1000
2820 1000
2830 1000
2840 1000
2850 1000
2860 1000
2870 1000
2880 1000
2890 1000
2900 1000
2910 1000
2920 1000
2930 1000
2940 1000
2950 1000
2960 1000
2970 1000
2980 1000
2990 1000
3000 1000

```


example, if you want your picture to appear as though light had fallen you could POKE 65537. This would change the paper to the top of a strip (the "top" from open to close, to turn the "light" into a "flash," you might POKE 65534) — and so on.

All that now remains to be done is to "translate" each of the separate graphics routines into its assembly language equivalent, and then done. This is in Listing 3. Although this isn't annotated (if it was it would take up far more space than it's worth) I've done as closely as I can to the BASIC in Listing 1 so that you should be able to follow it without too much difficulty. In any case you'll almost certainly want to tinker with these, adding your own extra routines (and possibly making the existing ones look more interesting on screen if you want to run the whole thing just all at once, then simply assemble Listing 3 at the end of Listing 2, leaving out the LOAD command and the POKE DRAW and DRAW label definitions of the end.

Picture demo

Once you've assembled the code and saved it (and don't forget that BASIC takes whatever you do), you'll need a little BASIC program to drive it. Listing

4 will do a demonstration job, cycling through the 256 pictures as you press a key. You'll find that production of the pictures is now quite fast enough for the system to be usable — and very considerably faster in fact, than we'd have managed if we'd stuck with the BASIC routine (initially used). The price of this improvement in speed is just the 256 bytes needed to store the random number table, which I think you'll agree is a very acceptable bargain.

As you'll see, the graphics routines themselves take up very little memory and you can afford to add to them without

much anxiety. Of course these backgrounds won't do on their own, but changing a few bits in routines to print up the shapes of letters, huts, and the like, in blocks of character squares isn't difficult. I have a little design program to deal with this, which stores the graphics for buildings in an alternative character set so that they can be printed up using ASCII, but that might be a job for some future article, and in the meantime you'll doubtless have your own ideas about ways of using the random back grounds. Could you be the first to advertise in *BYTE*?

BANDSCARVED, page 7

Listing 4

```

1 DEF FN RAND(100) = INT(100 * RND)
2 DEF FN RAND(100) = INT(100 * RND)
3 DEF FN RAND(100) = INT(100 * RND)
4 DEF FN RAND(100) = INT(100 * RND)
5 DEF FN RAND(100) = INT(100 * RND)
6 DEF FN RAND(100) = INT(100 * RND)
7 DEF FN RAND(100) = INT(100 * RND)
8 DEF FN RAND(100) = INT(100 * RND)
9 DEF FN RAND(100) = INT(100 * RND)
10 DEF FN RAND(100) = INT(100 * RND)
11 DEF FN RAND(100) = INT(100 * RND)
12 DEF FN RAND(100) = INT(100 * RND)
13 DEF FN RAND(100) = INT(100 * RND)
14 DEF FN RAND(100) = INT(100 * RND)
15 DEF FN RAND(100) = INT(100 * RND)
16 DEF FN RAND(100) = INT(100 * RND)
17 DEF FN RAND(100) = INT(100 * RND)
18 DEF FN RAND(100) = INT(100 * RND)
19 DEF FN RAND(100) = INT(100 * RND)
20 DEF FN RAND(100) = INT(100 * RND)
21 DEF FN RAND(100) = INT(100 * RND)
22 DEF FN RAND(100) = INT(100 * RND)
23 DEF FN RAND(100) = INT(100 * RND)
24 DEF FN RAND(100) = INT(100 * RND)
25 DEF FN RAND(100) = INT(100 * RND)
26 DEF FN RAND(100) = INT(100 * RND)
27 DEF FN RAND(100) = INT(100 * RND)
28 DEF FN RAND(100) = INT(100 * RND)
29 DEF FN RAND(100) = INT(100 * RND)
30 DEF FN RAND(100) = INT(100 * RND)
31 DEF FN RAND(100) = INT(100 * RND)
32 DEF FN RAND(100) = INT(100 * RND)
33 DEF FN RAND(100) = INT(100 * RND)
34 DEF FN RAND(100) = INT(100 * RND)
35 DEF FN RAND(100) = INT(100 * RND)
36 DEF FN RAND(100) = INT(100 * RND)
37 DEF FN RAND(100) = INT(100 * RND)
38 DEF FN RAND(100) = INT(100 * RND)
39 DEF FN RAND(100) = INT(100 * RND)
40 DEF FN RAND(100) = INT(100 * RND)
41 DEF FN RAND(100) = INT(100 * RND)
42 DEF FN RAND(100) = INT(100 * RND)
43 DEF FN RAND(100) = INT(100 * RND)
44 DEF FN RAND(100) = INT(100 * RND)
45 DEF FN RAND(100) = INT(100 * RND)
46 DEF FN RAND(100) = INT(100 * RND)
47 DEF FN RAND(100) = INT(100 * RND)
48 DEF FN RAND(100) = INT(100 * RND)
49 DEF FN RAND(100) = INT(100 * RND)
50 DEF FN RAND(100) = INT(100 * RND)
51 DEF FN RAND(100) = INT(100 * RND)
52 DEF FN RAND(100) = INT(100 * RND)
53 DEF FN RAND(100) = INT(100 * RND)
54 DEF FN RAND(100) = INT(100 * RND)
55 DEF FN RAND(100) = INT(100 * RND)
56 DEF FN RAND(100) = INT(100 * RND)
57 DEF FN RAND(100) = INT(100 * RND)
58 DEF FN RAND(100) = INT(100 * RND)
59 DEF FN RAND(100) = INT(100 * RND)
60 DEF FN RAND(100) = INT(100 * RND)
61 DEF FN RAND(100) = INT(100 * RND)
62 DEF FN RAND(100) = INT(100 * RND)
63 DEF FN RAND(100) = INT(100 * RND)
64 DEF FN RAND(100) = INT(100 * RND)
65 DEF FN RAND(100) = INT(100 * RND)
66 DEF FN RAND(100) = INT(100 * RND)
67 DEF FN RAND(100) = INT(100 * RND)
68 DEF FN RAND(100) = INT(100 * RND)
69 DEF FN RAND(100) = INT(100 * RND)
70 DEF FN RAND(100) = INT(100 * RND)
71 DEF FN RAND(100) = INT(100 * RND)
72 DEF FN RAND(100) = INT(100 * RND)
73 DEF FN RAND(100) = INT(100 * RND)
74 DEF FN RAND(100) = INT(100 * RND)
75 DEF FN RAND(100) = INT(100 * RND)
76 DEF FN RAND(100) = INT(100 * RND)
77 DEF FN RAND(100) = INT(100 * RND)
78 DEF FN RAND(100) = INT(100 * RND)
79 DEF FN RAND(100) = INT(100 * RND)
80 DEF FN RAND(100) = INT(100 * RND)
81 DEF FN RAND(100) = INT(100 * RND)
82 DEF FN RAND(100) = INT(100 * RND)
83 DEF FN RAND(100) = INT(100 * RND)
84 DEF FN RAND(100) = INT(100 * RND)
85 DEF FN RAND(100) = INT(100 * RND)
86 DEF FN RAND(100) = INT(100 * RND)
87 DEF FN RAND(100) = INT(100 * RND)
88 DEF FN RAND(100) = INT(100 * RND)
89 DEF FN RAND(100) = INT(100 * RND)
90 DEF FN RAND(100) = INT(100 * RND)
91 DEF FN RAND(100) = INT(100 * RND)
92 DEF FN RAND(100) = INT(100 * RND)
93 DEF FN RAND(100) = INT(100 * RND)
94 DEF FN RAND(100) = INT(100 * RND)
95 DEF FN RAND(100) = INT(100 * RND)
96 DEF FN RAND(100) = INT(100 * RND)
97 DEF FN RAND(100) = INT(100 * RND)
98 DEF FN RAND(100) = INT(100 * RND)
99 DEF FN RAND(100) = INT(100 * RND)
100 DEF FN RAND(100) = INT(100 * RND)

```

ARE YOU A BUDDING PROGRAMMER?

It's always looking for high-quality games and utilities for publication. If you have a top-notch game or a useful utility for the Spectrum or QL, why not send it to us for appraisal on cassette, or microdrive, or computer with a listing if possible.

There is also our new Picture Shot Cuts to show you your practical, novel or imaginative software utilities with crash prices for published listings. For longer programs we pay competitive rates, and if you have an idea for an article or series for ZX — drop us a line or phone Bryan or Gill on 04-437 3626 to talk it over.



SPORTS SIMULATION OFFER



Three games for the price of one from Argus Press Software

Sports simulation fans can pick up three games for the special price of £9.95 in the 28 modern offer requested in conjunction with Argus Press Software.

The most recent release is **Felix Falck's Handball**. **Moradono** is a football simulation with a difference as it puts you in the boots of the goalkeeper with the challenge of keeping a clean sheet in practice games and league matches at a variety of skill levels.

Nick Falck's Golf gives you the fun of 900 screens and 18 holes as you play round a faithful simulation of the course at Royal St George's. The package comes complete with a fully illustrated guide to the course.

American Football is an oriented grid line simulation for one or two players featuring all the elements of the real game — passing, defence, running, defence and kicking to create a blend of on field action and strategy.

Now these games are available by mail order for just £9.95 (including postage and packing). Do not send your cheque or postal order to the 28 address but to the address on the coupon on this page. Make cheques and postal orders payable to **Argus Press Software**.

But if sports simulations don't appeal to you, don't despair! This is just the first in a series of special offers featuring Argus Press Software games and forthcoming ones will feature war, space and arcade games.



Argus Press Software Sports Offer

Please send me **Handball Moradono**, **Nick Falck's Golf** and **American Football**

Name

Address

I enclose a cheque/postal order for £9.95 including p&g payable to Argus Press Software.

Send your remittance to: **Argus Mail Order (28), Units 182, Garden Developments, Watery Lane, Gorness, Lancashire BB5 2PL**



TERRA CRESTA

Atari 2600
C148

Another decade warrens from Imaginix, this time it's space rather than the old friends, and a lot of spaceship lightness. The control is spaceship flying over the scrolling surface of a planet painted diagonally. If you've heard the one (laugh) by now you know of aliens who will swoop-out or fire, so don't try to pierce you off over the place. Plus, at the same time as dodging and firing back at the aliens, you've also got to manage the spaceship on the ground while you're going to the bus, by kicking barrels into the sky.

Needless to say, all this sky-bus action keeps you very busy but is you pick off the 100 you are rewarded with occasional weapons that help you blast your way through to the final stages of the game where you'll probably be killed by a huge boss ship.

The old Atari standard don't turn up action, professionally presented but not nearly so long with originality for some reason (possibly to keep the competition as faithful as possible to the arcade original). The playing area is squeezed into the right hand side of the screen while the rest is occupied by the Imaginix logo and score

tables. As a result the graphics in the actual playing area though occasionally well drawn are badly scaled and require close attention to keep on eye on what's happening. I know a slight diversion or less faithful can seem, but it might have made a slightly better game if the programmers had used the whole

of the screen for playing it in as it stands it looks a bit like you're only getting half a game and one half of the screen is actually wasted.

There are no major faults in Terra Cresta but without a bit more originality it does seem to take some less important features that turn up.

GOOD



GAME'S

10TH FRAME

Atari 2600
C148

When it comes to sports simulations, collections can be a drawback. This ten pin bowling game replaces the tedious format of the game while still making up for the weakness of the actual game. Features mentioned include a pointer, a

long wait while the machine at the end of the lane moves away, knocked pins, pins, ball, and so on. It could be said to have been designed to avoid the pinball similarity, its meaning for the ball is the same as in the real game.

Controls are keyboard only and there are no computer support as you can either go to be entered on importing your own score, or the three difficulty levels or programs some friends into coming in just to eight points.

The ball control comes in during the rolling and throwing of the ball. Rotation is indicated with a laser grid around and measures for testing your rotation on a moving ball. Diagrams of the bottom left of the screen. The indicator also determines the amount of "hook" you put on the ball. A ball hook of 100 percent means that the ball will generate spin into the gutter about a third of the way down the alley.

Modifying the refinements of the game takes some time but

once you've become pinball and it becomes very much like the real thing. It's a game of surprise when you see them when you hit the target, then when you get a strike in very low key - no flashing lights or whistles.

The animation, such as it is, is good and the graphics are fast and although it will suit those who are pin bowling, maybe the impulse buyer may find it redundant and over priced.



GRIM





FIRST STEPS INTO

machine code routine to do the same job instantly? After all, the Spectrum screen is distributed area is just a section of memory, like any other, it occupies the 768 bytes from address 28000 onwards, and so we know moving numbers around like this is something that the 680 is very good at. So this is what we're going to attempt.

Registers

In the previous article we dealt with the 680's registers only one at a time, but a single register can hold only numbers up to 255, and to write our "rain" routine we'll need to be able to handle numbers larger than this. Fortunately there's a simple solution: We can store larger numbers by using TWO registers, and the 680 has in fact a lot of instructions for handling pairs of registers for just this purpose. Two registers which are paired together are the H and L registers (either common pointers are bit labels H() and L() and an instruction for loading the H() pair with a number takes the form

LD HL, #N

What's actually happening here? Well, the number N is in fact being split into two separate bytes. One of these — the "High byte" — stores multiples of 256, and the other — the "Low byte" — stores whatever remains. So in this case the H register stores the number 3 (256x3=768) and the L register stores 3 (i.e. 512-768) — it's easy to discover the biggest number that can be stored in this way: If both registers contain 255, then we have

255x256+255=65535. However, we don't really need to cope with this in detail at present, it's enough just to appreciate that one number of 65535 can be loaded into the H, L register pair.

Now we're in a position to take a look at Listing 2, which is an assembly program for changing the attributes of the first character square on the screen (top left hand corner). We'll be doing the machine code in the user defined graphics area, and the `clbg` directive at line 10 tells the assembler to do this. (Line 20 loads the number 255255 which is the address for the start of the attributes area, remember) into the H, L register pair, and line 30 loads the H register with the contents of the byte labelled `COUNT` (see the `dl`, where the defined value of 7 corresponds to white ink and black PAUSE). This much is fairly obvious — but

now comes the clever bit, and a new 280 instruction, `LD HL, A`. This means "put the number held in the A register into the byte of memory whose address is held in the H, L register pair". In our case, this results in the number 7 being stored in the address 255255. Then, finally, line 40 returns us to BASIC.

Now you might be thinking "But why don't we just use the instruction `LD (255255), A` instead of all this messing about with H, L? That doesn't worry about that for now — all will be revealed in due time. Meanwhile, you might like to try out this little routine for yourself: To do so, determine the code and return to BASIC, now the Spectrum (which will still have the machine code stored in the 1000 area) and enter the command `PRINT A` to get the letter 'A' in the top left character square. Now type `RANDOMISE USR 40000`, and `PRINT A` — the 'A' will still be there, but in white ink on black PAUSE. By poking a different value (`RANDOMISE USR 40000`) into `COUNT` and do `RANDOMISE USR 40000` again, it really does work, doesn't it? So all we need now is to repeat the process 768 times to change the colour for the whole of the screen. In other words, we need a program loop.

Loops

I kept put present problems to the moment, and left a consider a simpler one. Now can we make the 680 simply count a specific number of cycles (say 255), and then return to BASIC? In other words, can we write the equivalent of something like

FOR i=0 TO 255: NEXT i

Well yes we can. There are several ways of doing it, in fact — and Listing 3 is one of them. There are better methods than this one, but this method perhaps may seem to be the simplest we already know, actually, a better equivalent as a BASIC subroutine would be

```
10 LET COUNT=0
20 IF COUNT=255 THEN RETURN
30 LET COUNT=COUNT+1
40 GOTO 20
```

What I've done in Listing 3 is to set up a single byte (`COUNT`) which will store the game progress on the counting variable in the little BASIC loop above. The program proper starts at line 20. By clearing out the contents of the A register, and line 30 then dumps the result back into the address labelled

COUNT. This is just to make sure that the counting of the loops does actually start at zero. At line 40 we enter the loop itself — and I've used the label `LOOP` to mark this point in the program, we'll see why in a moment. Here we get the current contents of the `COUNT` byte into the A register (and on this first pass, of course), and line 50 then compares it with another new instruction, `CP 255`. (Think of this as meaning "Compare the number in the A register with 255". Now obviously the 255 can't actually compare two numbers in any intelligent sense, but it can do a simple subtraction of one from the other — and if the result of the subtraction is zero, then clearly the numbers are the same.) For the outcome of the `CP 255` instruction will be "remembered" by the 680 as either `NC` or `MC` (255).

Incidentally, this result is stored in a single bit of the F register, called the "Zero Flag", but you don't really need to worry about that here. The contents of the A register are unchanged by all this.

I think you'll see that here we have the possibility of conditional actions. We can make the program perform an action in the event of a 2550 result, and another if we get `MC` (255). Which brings us to line 60 — and you can probably guess what "B" means. "B" means "B" if the result is zero. In other words, is the number held in the byte `COUNT` turn out to be equal to 255, the program will return to BASIC. On this first pass, of course, `COUNT` contains zero — it isn't equal to 255, so the

program moves on to the next instruction, line 70. Here the A register is incremented by 1 (we deal with `INC A` later on), and then line 80 causes the new number to be deposited in `COUNT`. After the last pass, then, `COUNT` will no longer contain zero, but 1. Finally, at line 90, we encounter a "jump" instruction: `JF LOOP`. This tells the 680 to go back to the instruction labelled `LOOP` and carry on from there. I think you can follow the rest yourself. The program will cycle round and round, increasing the contents of `COUNT` by one each time, until eventually it contains 255. When this happens, line 60 will cause a return to BASIC.

This is the most complicated bit of programming we've attempted so far, and it's worth making sure that you understand just what's going on before proceeding further. If you get lost, refer back to the little 4-line

MACHINE CODE

BRAC loop counter for help. The parallel between the first line (waiter) and the first line (waiter) is very close. Happy! Good — because we have very nearly home and dry. If we combine Listing 3 and Listing 4 with just a very slight (but crucial) addition, we have a routine (Listing 4) which will perform a "wash" over the first 256 character squares of the screen (lines 20 to 40) set up the initial conditions (put the address of attributes into HL, and zero the counter) and then the loop is entered. The A register picks up the "colour" (line 40) and transfers it to the address which HL points to (line 40) then we check the counter (lines 40 to 40) increase it by one (line 40) and close the loop count (line 40). Now comes the crucial addition I mentioned. It is a new instruction, but you'll know exactly what it means. INC H, (line 40). This increases the number held in H, by 1 — so that H now points to the next address in the attributes file. Line 40 covers it jump back to the instruction labelled LOOP and the whole process is repeated over and over again until 256 cycles have been completed (jump a return to BASIC at line 40).

Can you see now why we used the apparently roundabout routine in Listing 3? In closing the current attribute address in H, we can simply "point" the routine to the next address by using the INC H instruction repeatedly as we go round the loop.

Listing 4 will indeed work — but of course it will only wash the top third of the screen. What about the other two thirds? No problem. All we need to do is to

use the routine three times in succession with H initially containing 22528 for the first run, 22784 (22528+256) for the second, and 23040 (22784+256) for the third (and final) run. But we don't need to type in the whole thing three times, because the JNC allows us to use the equivalent of a BASIC subroutine (using the CALL instruction). Listing 5 — which is, of last our ultimate goal — shows how it can be done fairly pointlessly. What I've done here is to take the whole of Listing 4 (with the initialisation of H removed, and attach the label "WASH" to it — so you can see at line 40 of Listing 3. This now forms a subroutine which can be called using CALL WASH.

The main assembler program then becomes the sequence of instructions from lines 20 to 40 in which H is given the appropriate initial value and the subroutine WASH called three times before the return to BASIC is made at line 40. Just one further point may puzzle you here: why is it that previously the RET Z instruction has occurred a return to BASIC whereas now it no longer does? The short answer — which is all we have space for — is that in this case it was preceded by a CALL instruction, and the 256 "remember" where to every other it returns sort of subroutines. Incidentally just as every BASIC subroutine must end with a RETURN sooner or later, so every machine code subroutine must conclude its code with a RET.

Run

Now let's make it all work. Assemble Listing 5, and if you

Listing 4

"Washing" the top third of the screen.

```
00368 10      ORG 00368
00368 20      LD HL,22528
00371 30      LD A,0
00374 40      LD (COUNT),A
00377 50      LD A,(COLOUR)
00380 60      LD (HL),A
00383 70      LD A,(COUNT)
00386 80      CP 255
00389 90      SET Z
00392 100     INC A
00395 110     LD (COUNT),A
00398 120     INC HL
00401 130     JP LOOP
00404 140     COLOUR DEFS 7
00407 150     COUNT DEFS 6
```

like, save the machine code (after this, produced using "wash" CODE 22528-44). Note the Spectrum (which loads the machine code still intact in the 1000 space), then in Listing 1 again — and just for fun, run it once more to remind yourself what the code is like before you use a machine code editor. Now, edit the code, delete lines 40, 40 and 40, and add the following BASIC lines:

```
10 FOR I=22528 TO 23040
20 RANDOMIZE I:GOTO 40
```

Now RUN the program. Wait for the pattern to be drawn, choose your paper colour, hold your breath — then choose the ink. Magic!

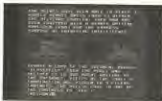
Listing 5

Complete screen "wash".

```
00368 10      ORG 00368
00368 20      LD HL,22528
00371 30      CALL WASH
00374 40      LD HL,22784
00377 50      CALL WASH
00380 60      LD HL,23040
00383 70      CALL WASH
00386 80      SET Z
00389 90      LD A,0
00392 100     LD (COUNT),A
00395 110     ORG 00395
00395 120     LD (COUNT),A
00398 130     LD A,(COLOUR)
00401 140     LD (HL),A
00404 150     LD A,(COUNT)
00407 160     CP 255
00410 170     SET Z
00413 180     INC A
00416 190     LD (COUNT),A
00419 200     INC HL
00422 210     JP LOOP
00425 220     COLOUR DEFS 7
00428 230     COUNT DEFS 6
```

Activision's latest casts
the hacker as hero.

HACKER 2



Hacker 2: The Doomsday Papers Activision \$9.95

As cassette labeled *Activision*, an incomprehensible technical manual and no instructions whatsoever, you will never see the yellow beginnings for a great game. But to the great surprise, these ingredients make *Hacker 2* one of the most sophisticated releases I have played for ages. It's American, of course.

In this games' present, you hooded your way through business strategies to prove yourself one of the world's great hackers, but your endeavor did not go unnoticed. Leading the *Activision* tape, you log in to what seems a normal enough Microsoft style service — until the music is interrupted by a message from the CIA. As the leading expert on breaking security systems, they have a mission for you. The Russians have developed a simple but devastating plan to topple the US government (and those scheming conspire) the details are enclosed in a document — "The Doomsday Paper" (sort of thing the Russians would call such devious plans). Your task is to retrieve these papers from the military complex in Siberia (where else? As you may have guessed, I do not entirely approve of the typically paranoid American scenario). How can you do this, if you're sitting in front of a screen back home? Via the miracles of modern technology that's how: hidden within the code are three Module Remotes (with GRS), disks of your choice) and the Multi-Functioning Magic Matrix, which controls the MSUs,

the MSU is linked by satellite to CIA's central computer which is connected by modem to your computer. On your screen you see a map up of the MSU.

Security at the base is not so strict that MSUs can wander around undetected. Guards patrol the corridors, and there are thirty-eight security cameras scattered around. Two electronic robots float through these corridors, checking that no unauthorized objects are in the picture. With the MSU, you can view what any of the 38 cameras see, what other monitor is registering or a plan of the immediate area surrounding the MSU. The clever bit is, the MSU also includes 32 voice channels. If you synchronize correctly, you can listen in on conversations or recording of what it should be

seeing, and speak your MSU in (whichever). You have to check several things at once, so the MSU has four screens.

Sounds complicated? I've simplified it considerably. *Hacker 2* is an extremely complex piece of software. It's difficult to see, and challenging, though not especially intellectual. Quick thinking and intelligent reasoning are required. What makes it so unusual is the incredible attention to detail. For example, you have to set the vertical field on each of the TV screens. The video simulation is so authentic you even get noise bars when you fast forward. And the CIA Manual is phenomenally containing (not that I've ever seen a CIA manual). Real efforts have been made with this game to create realism, something very rare with British software.

The few criticized glitches are adequate rather than crushing (slightly chunky, otherwise *Hacker 2* is perfectly executed). Patience is required, but it does pay off. Usability is questionable, but I reckon this will provide value for money for most people. *Activision's* Spectrum conversion are usually look-alike affairs (remember the memorable *Chouten*?) so congratulations are in order, at last they have created a winner!



GREAT





Little Computer People



Little Computer People
Activision
\$9.95
(128 only)

While you're sighted inside the Commodore 64, it now appears that the Little Computer People have started to step up in other places as well. Sometime 1988, version 4 of the Spectrum (apparently there's not enough room inside the 486 models — they may be Little Computer People, but there's not that little) for what are these little beings (LCPs as they are sometimes called)? The first LCPs were observed by American programmer David Gagne, who had suspected the presence of tiny human beings within his computer for some time when he created what he called a "house on a cassette" loading this cassette into the recorder

created a small home inside it empty and just waiting for someone to come along and live inside.

When you load your own "house" into your 128 or Plus, too you are first of all presented with a rolemodel into which you enter your name, live date, and time of day. This, and other information about your LCP is continually updated and carried over from session to session.

Once we had gotten our own Plus Two hooked up, we did find ourselves sitting in front of the Spectrum staring at the empty House on the cassette. After a couple of minutes a small head stuck its way beyond the front door, then an LCP entered to give the place the once over.

This was Dr. D, a cool looking LCP sporting sunglasses and bow tie, who proceeded to take up residence (accompanied by his Little Computer Dog). Once your LCP is in, you've got a choice: you can sit back and

Now you can invite
Little Computer People
 to take up residence in
 your 128.

do nothing and just watch him yes, but you get to hear all sorts of sightings of LCPsness which, by implication, gives rise to all sorts of interesting questions) wander around, or you can send messages, or perform various actions via the keyboard.

Left to himself, Dr. D seemed perfectly happy to go about the house peering, listening to his stereo, tinkering with his computer or even doing aerobics. You can sit and watch for ages and your LCP will get on with things on his own. Watching this is a bit like playing with empty heads — ultimately pointless, but quite calming somehow.

However, like all intelligent beings, your LCP has certain needs which have to be met — food and drink have to be supplied by you, via the keyboard. (And when he gets a bit or lonely its up to you to cheer him up as best you can (perhaps by playing cards or anagrams with him).

There are certain actions which you can always rely upon, such as giving him a new look or record to play, but there is enough scope within the program to allow you to experiment and discover how best to get on with your LCP just as you would do with any other person.

Remember though, that if you neglect your LCP he's likely to go green and take to his bed with a terminal case of the sulks, to fly your responsibility to look after him.

Little Computer People is thoroughly ingenious and a genuine oddity. It's an entertainment at its best, though it's not a game, in a way it can almost be described as the computer equivalent of background music, except that it often has the opportunity to get involved with what's going on. It can't be recommended to a specific audience in the same way that an adventure or arcade game can be, but if you've got a 128 or a Plus 2 why not say hello to a Little Computer Person sometime. Who knows, it could be the start of a beautiful friendship.



GREAT



Amphibian daring in the dungeon as Hanson unveils a sprawling magical epic

**Ranama
Hanson
\$7.95**

When you're a wizard, the only way to win is not to play. That's the motto of the wizard who's been hired to slay the dragon. But when the dragon turns out to be a friendly, intelligent creature, the wizard's job is no longer so simple. In this book, the wizard is a young man named Ranama, and the dragon is a young woman named Ranama. They are both wizards, and they are both in love. The book is a fantasy novel, and it's a very good one. It's a story of love, magic, and adventure. It's a story that will keep you reading late into the night.

The book is a fantasy novel, and it's a very good one. It's a story of love, magic, and adventure. It's a story that will keep you reading late into the night. The book is a fantasy novel, and it's a very good one. It's a story of love, magic, and adventure. It's a story that will keep you reading late into the night.

intentional letters of "Ranama" books to the original work. This is a relief, but that calls for a new way and good to reflect on with the legend. On the other hand, the book is a very good one. It's a story of love, magic, and adventure. It's a story that will keep you reading late into the night.

The book is a fantasy novel, and it's a very good one. It's a story of love, magic, and adventure. It's a story that will keep you reading late into the night. The book is a fantasy novel, and it's a very good one. It's a story of love, magic, and adventure. It's a story that will keep you reading late into the night. The book is a fantasy novel, and it's a very good one. It's a story of love, magic, and adventure. It's a story that will keep you reading late into the night.

Seeing the light

It is a story of love, magic, and adventure. It's a story that will keep you reading late into the night. The book is a fantasy novel, and it's a very good one. It's a story of love, magic, and adventure. It's a story that will keep you reading late into the night.

The book is a fantasy novel, and it's a very good one. It's a story of love, magic, and adventure. It's a story that will keep you reading late into the night. The book is a fantasy novel, and it's a very good one. It's a story of love, magic, and adventure. It's a story that will keep you reading late into the night. The book is a fantasy novel, and it's a very good one. It's a story of love, magic, and adventure. It's a story that will keep you reading late into the night.

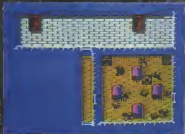
The book is a fantasy novel, and it's a very good one. It's a story of love, magic, and adventure. It's a story that will keep you reading late into the night. The book is a fantasy novel, and it's a very good one. It's a story of love, magic, and adventure. It's a story that will keep you reading late into the night. The book is a fantasy novel, and it's a very good one. It's a story of love, magic, and adventure. It's a story that will keep you reading late into the night.

The book is a fantasy novel, and it's a very good one. It's a story of love, magic, and adventure. It's a story that will keep you reading late into the night. The book is a fantasy novel, and it's a very good one. It's a story of love, magic, and adventure. It's a story that will keep you reading late into the night.





you will need to know some of your names by looking at the spots that will result. There is a hint

[illegible][illegible]

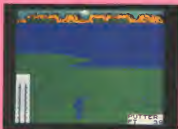
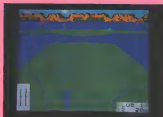


LEADER BOARD

The most authentic golf simulation the Spectrum has ever seen.

Leaderboard
US Gold
£9.95

Sports enthusiasts have seen in the doldrums recently to the arrival of Leaderboard to chess-typing Commodore games on the Spectrum is cause for celebration especially as it's in a different league to any existing golf game.



The list of options is undeniably impressive — there are four 18 hole courses to play on plus a practice driving range. Up to four players can take part in a round and there are three levels of difficulty, novice, amateur and professional (where subtle shifts in wind direction affect the flight of the ball). Each player has a choice of clubs (from an available 14) and can adjust the power, size and hook in his shot by timing the drive against a moving bar display.

The joy of the game is the

high responsiveness to subtle changes in your play. Select the wrong club or power too much and you pay dearly as your score threatens to expand off the scoreboard. On a lot of holes, water hazards dominate and if this was real the yard of spending a fortune in lost balls. Pinpoint accuracy is essential all the way down to the hole. You are kept up to date with the exact distance to the hole, even down to the last few inches so that even a tap in can be a problem especially on most green's have a vicious slope. An

authentic touch familiar to poor golfers occurs when you pull too hard but on the right line and the ball hops over the rim of the hole.

Graphically, Leaderboard is expert with three block colours and a smooth, animated playing flag. The perspective of the ball in flight as it swings through the air is real concerning and golf buffs will uncover a lot of small details that have been overlooked by previous golfing games — such as hitting the flag in the hole from an approach shot.

Leaderboard is an exceptional game simulation which will be topped up by anyone with an interest in golf and by those who enjoy a well put together game.

The ultimate advantage for a sporting simulation would be that it was preferable to playing the game. Real and although Leaderboard can't quite manage that, it's a very close thing.



**MONSTER
HIT**

THE DISCOVERY COLUMN



John Wase presents another selection of useful routines for Discovery owners.

One month ago, a correspondent was asking about random access files and how to handle them. He thought that perhaps the best solution was to set up a relatively short file of key numbers to be used as an index to manipulate larger files. When I wrote back, I pointed out that this was a general disc filing problem rather than a Discovery problem, but that he was on the right track. Now here's a short program that illustrates the principle perfectly.

The Dreaded Telephone List

Yet it is, once again, I've got a telephone list. This is by Chris Oliver of Tottenham. Needless to say, I took my eye because it illustrated beautifully random access filing. It was clear. It had clear accompanying notes and it came with a disc. So I left out of the program. Chris uses two random access files, a main listing file having names, addresses and numbers, and a file of key numbers, used as an index. The index number list must be filed as string arrays, discussed so that, as in all random access files, each entry in a particular file is the same length as the others.

Listing 1
This program creates and reserves space on the disc

for two files. Line 1 creates a random access file with a record length of four, that is three digits, plus one for the \$M\$ used as a separator and automatically inserted each time we print to the file; this gives 999 of these records plus an extra two at the end which are used as slots for storing counters. Line 3 creates a further file of record length 50 (plus one for the separator) because -4 is stipulated for the file size. It uses one half of the space available on the disc, which is why the index file was created first. Next the positions used for storing the counters are printed (lines 5-6). As Chris says, this is a good illustration of the method. Discovery uses to write to a random access file, this part of the program is then no longer needed.

Listing 2
The first thing this program must do is fetch the current values of the counters before manipulating or adding further records, and this is done in lines 10-16 which illustrate clearly the method of reading from a random access file. And this now covers the basis of the program; the rest is merely manipulation.

As Chris says, there are one or two things mentioned only briefly in the Oplus manual which are worth emphasizing. Firstly you cannot get a record to flow in two ways at once.

Since there are plenty of streams available, Chris finds it sensible to reserve some for writing to and others for reading from files. Chris also warns, too, of using **CLASH** when useful on occasions for closing all streams. I don't what I say, but, if I clear out everything in all open streams, doing a **Roll Control** on the record you were trying to send to the file, so in routine "find", line 690 **CLASH** could be used, but in the "delete record" routine which moves key-numbers in the index file, each stream must be closed (line 760) to send the latest to the file.

The manual does not mention that if you **POINT** it you find, it is not a valid number; you get the "excess in basic" report. And unless you are a very fast reader you will need to control the scrolling when printing the contents of a file to the screen. A simple variable is good in the "list-of" and "find" routines of the example program.

Improvements

The example program has been deliberately kept simple. Improvements could include use of subroutines for reading from or writing to random access files and use of a "inserting search" routine instead of the simple progressive search could speed things up. In addition the crude "change record" routine could be replaced with an editing routine, and an "add-more" option in the add item routine would be helpful. Use of machine code input and find routines using the RAM disc (particularly the big 128K one) would also make things run faster. Finally I had a glitch in the disc (which got on a bit of a perpetual repair) or something while the program was running, so I suspect, I RAM it again. That does the file, this could be avoided by having listing 1 as an outprinting subroutine.

Listing 1

```
1 1000 Create two random access files. (also file discovery units
2 1010 and print headers) with record size taken up to 4 characters
3 1020 Print header for file numbers on file disc (line 1000) to the
4 1030 separator)
5 1040 Print header for file numbers on file disc (line 1000) to the
6 1050 separator)
7 1060 Print header for file numbers on file disc (line 1000) to the
8 1070 separator)
9 1080 Print header for file numbers on file disc (line 1000) to the
10 1090 separator)
11 1100 Print header for file numbers on file disc (line 1000) to the
12 1110 separator)
13 1120 Print header for file numbers on file disc (line 1000) to the
14 1130 separator)
15 1140 Print header for file numbers on file disc (line 1000) to the
16 1150 separator)
17 1160 Print header for file numbers on file disc (line 1000) to the
18 1170 separator)
19 1180 Print header for file numbers on file disc (line 1000) to the
20 1190 separator)
21 1200 Print header for file numbers on file disc (line 1000) to the
22 1210 separator)
23 1220 Print header for file numbers on file disc (line 1000) to the
24 1230 separator)
25 1240 Print header for file numbers on file disc (line 1000) to the
26 1250 separator)
27 1260 Print header for file numbers on file disc (line 1000) to the
28 1270 separator)
29 1280 Print header for file numbers on file disc (line 1000) to the
30 1290 separator)
31 1300 Print header for file numbers on file disc (line 1000) to the
32 1310 separator)
33 1320 Print header for file numbers on file disc (line 1000) to the
34 1330 separator)
35 1340 Print header for file numbers on file disc (line 1000) to the
36 1350 separator)
37 1360 Print header for file numbers on file disc (line 1000) to the
38 1370 separator)
39 1380 Print header for file numbers on file disc (line 1000) to the
40 1390 separator)
41 1400 Print header for file numbers on file disc (line 1000) to the
42 1410 separator)
43 1420 Print header for file numbers on file disc (line 1000) to the
44 1430 separator)
45 1440 Print header for file numbers on file disc (line 1000) to the
46 1450 separator)
47 1460 Print header for file numbers on file disc (line 1000) to the
48 1470 separator)
49 1480 Print header for file numbers on file disc (line 1000) to the
50 1490 separator)
51 1500 Print header for file numbers on file disc (line 1000) to the
52 1510 separator)
53 1520 Print header for file numbers on file disc (line 1000) to the
54 1530 separator)
55 1540 Print header for file numbers on file disc (line 1000) to the
56 1550 separator)
57 1560 Print header for file numbers on file disc (line 1000) to the
58 1570 separator)
59 1580 Print header for file numbers on file disc (line 1000) to the
60 1590 separator)
61 1600 Print header for file numbers on file disc (line 1000) to the
62 1610 separator)
63 1620 Print header for file numbers on file disc (line 1000) to the
64 1630 separator)
65 1640 Print header for file numbers on file disc (line 1000) to the
66 1650 separator)
67 1660 Print header for file numbers on file disc (line 1000) to the
68 1670 separator)
69 1680 Print header for file numbers on file disc (line 1000) to the
70 1690 separator)
71 1700 Print header for file numbers on file disc (line 1000) to the
72 1710 separator)
73 1720 Print header for file numbers on file disc (line 1000) to the
74 1730 separator)
75 1740 Print header for file numbers on file disc (line 1000) to the
76 1750 separator)
77 1760 Print header for file numbers on file disc (line 1000) to the
78 1770 separator)
79 1780 Print header for file numbers on file disc (line 1000) to the
80 1790 separator)
81 1800 Print header for file numbers on file disc (line 1000) to the
82 1810 separator)
83 1820 Print header for file numbers on file disc (line 1000) to the
84 1830 separator)
85 1840 Print header for file numbers on file disc (line 1000) to the
86 1850 separator)
87 1860 Print header for file numbers on file disc (line 1000) to the
88 1870 separator)
89 1880 Print header for file numbers on file disc (line 1000) to the
90 1890 separator)
91 1900 Print header for file numbers on file disc (line 1000) to the
92 1910 separator)
93 1920 Print header for file numbers on file disc (line 1000) to the
94 1930 separator)
95 1940 Print header for file numbers on file disc (line 1000) to the
96 1950 separator)
97 1960 Print header for file numbers on file disc (line 1000) to the
98 1970 separator)
99 1980 Print header for file numbers on file disc (line 1000) to the
100 1990 Print header for file numbers on file disc (line 1000) to the
```

jumped over in later program reports. Nevertheless, the program works, works satisfactorily and displays well random screen data.

An Alternative

Chris writes that he has used machine code to reduce the six minutes of John Barnhart's BASIC screen dump, but does not know how to pass non-ASCII data to the printer without returning to a BASIC "hand" statement. He would also like to know the addresses of the ports used and how to interface them. Can anyone help him?

Several other readers' comments might also be of interest. For instance, David Colson of Winnipeg mentions that if he wants a program to die as follows:

```
SAVE "filename" LINE 10
```

and when he gets to the question mark, changes the line to 7 (and adds, instead, a couple of lines before changing the line back to 5, then this protects the program. He says that you must have line 10 deleted or the computer will accept it. If you try and load by typing in the whole name after a C&I, you usually get the "file not found" error. It also prevents accidental erases. Perhaps even more important. Oh, and if you suffer from insomnia, then just change paper to another value before you C&I and all is revealed.

RAM Chip

A query from a computer club concerned 5 1/4 inch drives. Is it possible to run two of these drives from a Discovery unit and what is the RAM chip? The answer is that the RAM chip is pretty standard and not difficult to get. It is an 18 018 chip and yes, you can run two 5 1/4 inch drives if you so wish, don't forget to unplug the 3 1/2" floppy. Discovery drives from the printed circuit board. Mr D. D. M. Cherry of Fort St. Vrain, United Rep. of Sweden, also offers comments and also has about disc drive recommendations. Almost any modern disc drive will do, but it needs to be Schugart compatible and you need to add a power supply unit if it hasn't got one.

Errors

S. Comb of Wetzigen, Switzerland, mentioned that he had read the reference about line 100 being in the Discoverer issue. He had had continual errors of this sort and noticed that they often appeared through interference from other peripherals, for instance, switching on a printer. He recommended that if you are

Listing 2

```

100 REM ***** DISCOVERY *****
101 REM *****
102 REM *****
103 REM *****
104 REM *****
105 REM *****
106 REM *****
107 REM *****
108 REM *****
109 REM *****
110 REM *****
111 REM *****
112 REM *****
113 REM *****
114 REM *****
115 REM *****
116 REM *****
117 REM *****
118 REM *****
119 REM *****
120 REM *****
121 REM *****
122 REM *****
123 REM *****
124 REM *****
125 REM *****
126 REM *****
127 REM *****
128 REM *****
129 REM *****
130 REM *****
131 REM *****
132 REM *****
133 REM *****
134 REM *****
135 REM *****
136 REM *****
137 REM *****
138 REM *****
139 REM *****
140 REM *****
141 REM *****
142 REM *****
143 REM *****
144 REM *****
145 REM *****
146 REM *****
147 REM *****
148 REM *****
149 REM *****
150 REM *****
151 REM *****
152 REM *****
153 REM *****
154 REM *****
155 REM *****
156 REM *****
157 REM *****
158 REM *****
159 REM *****
160 REM *****
161 REM *****
162 REM *****
163 REM *****
164 REM *****
165 REM *****
166 REM *****
167 REM *****
168 REM *****
169 REM *****
170 REM *****
171 REM *****
172 REM *****
173 REM *****
174 REM *****
175 REM *****
176 REM *****
177 REM *****
178 REM *****
179 REM *****
180 REM *****
181 REM *****
182 REM *****
183 REM *****
184 REM *****
185 REM *****
186 REM *****
187 REM *****
188 REM *****
189 REM *****
190 REM *****
191 REM *****
192 REM *****
193 REM *****
194 REM *****
195 REM *****
196 REM *****
197 REM *****
198 REM *****
199 REM *****
200 REM *****
201 REM *****
202 REM *****
203 REM *****
204 REM *****
205 REM *****
206 REM *****
207 REM *****
208 REM *****
209 REM *****
210 REM *****
211 REM *****
212 REM *****
213 REM *****
214 REM *****
215 REM *****
216 REM *****
217 REM *****
218 REM *****
219 REM *****
220 REM *****
221 REM *****
222 REM *****
223 REM *****
224 REM *****
225 REM *****
226 REM *****
227 REM *****
228 REM *****
229 REM *****
230 REM *****
231 REM *****
232 REM *****
233 REM *****
234 REM *****
235 REM *****
236 REM *****
237 REM *****
238 REM *****
239 REM *****
240 REM *****
241 REM *****
242 REM *****
243 REM *****
244 REM *****
245 REM *****
246 REM *****
247 REM *****
248 REM *****
249 REM *****
250 REM *****
251 REM *****
252 REM *****
253 REM *****
254 REM *****
255 REM *****
256 REM *****
257 REM *****
258 REM *****
259 REM *****
260 REM *****
261 REM *****
262 REM *****
263 REM *****
264 REM *****
265 REM *****
266 REM *****
267 REM *****
268 REM *****
269 REM *****
270 REM *****
271 REM *****
272 REM *****
273 REM *****
274 REM *****
275 REM *****
276 REM *****
277 REM *****
278 REM *****
279 REM *****
280 REM *****
281 REM *****
282 REM *****
283 REM *****
284 REM *****
285 REM *****
286 REM *****
287 REM *****
288 REM *****
289 REM *****
290 REM *****
291 REM *****
292 REM *****
293 REM *****
294 REM *****
295 REM *****
296 REM *****
297 REM *****
298 REM *****
299 REM *****
300 REM *****
301 REM *****
302 REM *****
303 REM *****
304 REM *****
305 REM *****
306 REM *****
307 REM *****
308 REM *****
309 REM *****
310 REM *****
311 REM *****
312 REM *****
313 REM *****
314 REM *****
315 REM *****
316 REM *****
317 REM *****
318 REM *****
319 REM *****
320 REM *****
321 REM *****
322 REM *****
323 REM *****
324 REM *****
325 REM *****
326 REM *****
327 REM *****
328 REM *****
329 REM *****
330 REM *****
331 REM *****
332 REM *****
333 REM *****
334 REM *****
335 REM *****
336 REM *****
337 REM *****
338 REM *****
339 REM *****
340 REM *****
341 REM *****
342 REM *****
343 REM *****
344 REM *****
345 REM *****
346 REM *****
347 REM *****
348 REM *****
349 REM *****
350 REM *****
351 REM *****
352 REM *****
353 REM *****
354 REM *****
355 REM *****
356 REM *****
357 REM *****
358 REM *****
359 REM *****
360 REM *****
361 REM *****
362 REM *****
363 REM *****
364 REM *****
365 REM *****
366 REM *****
367 REM *****
368 REM *****
369 REM *****
370 REM *****
371 REM *****
372 REM *****
373 REM *****
374 REM *****
375 REM *****
376 REM *****
377 REM *****
378 REM *****
379 REM *****
380 REM *****
381 REM *****
382 REM *****
383 REM *****
384 REM *****
385 REM *****
386 REM *****
387 REM *****
388 REM *****
389 REM *****
390 REM *****
391 REM *****
392 REM *****
393 REM *****
394 REM *****
395 REM *****
396 REM *****
397 REM *****
398 REM *****
399 REM *****
400 REM *****
401 REM *****
402 REM *****
403 REM *****
404 REM *****
405 REM *****
406 REM *****
407 REM *****
408 REM *****
409 REM *****
410 REM *****
411 REM *****
412 REM *****
413 REM *****
414 REM *****
415 REM *****
416 REM *****
417 REM *****
418 REM *****
419 REM *****
420 REM *****
421 REM *****
422 REM *****
423 REM *****
424 REM *****
425 REM *****
426 REM *****
427 REM *****
428 REM *****
429 REM *****
430 REM *****
431 REM *****
432 REM *****
433 REM *****
434 REM *****
435 REM *****
436 REM *****
437 REM *****
438 REM *****
439 REM *****
440 REM *****
441 REM *****
442 REM *****
443 REM *****
444 REM *****
445 REM *****
446 REM *****
447 REM *****
448 REM *****
449 REM *****
450 REM *****
451 REM *****
452 REM *****
453 REM *****
454 REM *****
455 REM *****
456 REM *****
457 REM *****
458 REM *****
459 REM *****
460 REM *****
461 REM *****
462 REM *****
463 REM *****
464 REM *****
465 REM *****
466 REM *****
467 REM *****
468 REM *****
469 REM *****
470 REM *****
471 REM *****
472 REM *****
473 REM *****
474 REM *****
475 REM *****
476 REM *****
477 REM *****
478 REM *****
479 REM *****
480 REM *****
481 REM *****
482 REM *****
483 REM *****
484 REM *****
485 REM *****
486 REM *****
487 REM *****
488 REM *****
489 REM *****
490 REM *****
491 REM *****
492 REM *****
493 REM *****
494 REM *****
495 REM *****
496 REM *****
497 REM *****
498 REM *****
499 REM *****
500 REM *****
501 REM *****
502 REM *****
503 REM *****
504 REM *****
505 REM *****
506 REM *****
507 REM *****
508 REM *****
509 REM *****
510 REM *****
511 REM *****
512 REM *****
513 REM *****
514 REM *****
515 REM *****
516 REM *****
517 REM *****
518 REM *****
519 REM *****
520 REM *****
521 REM *****
522 REM *****
523 REM *****
524 REM *****
525 REM *****
526 REM *****
527 REM *****
528 REM *****
529 REM *****
530 REM *****
531 REM *****
532 REM *****
533 REM *****
534 REM *****
535 REM *****
536 REM *****
537 REM *****
538 REM *****
539 REM *****
540 REM *****
541 REM *****
542 REM *****
543 REM *****
544 REM *****
545 REM *****
546 REM *****
547 REM *****
548 REM *****
549 REM *****
550 REM *****
551 REM *****
552 REM *****
553 REM *****
554 REM *****
555 REM *****
556 REM *****
557 REM *****
558 REM *****
559 REM *****
560 REM *****
561 REM *****
562 REM *****
563 REM *****
564 REM *****
565 REM *****
566 REM *****
567 REM *****
568 REM *****
569 REM *****
570 REM *****
571 REM *****
572 REM *****
573 REM *****
574 REM *****
575 REM *****
576 REM *****
577 REM *****
578 REM *****
579 REM *****
580 REM *****
581 REM *****
582 REM *****
583 REM *****
584 REM *****
585 REM *****
586 REM *****
587 REM *****
588 REM *****
589 REM *****
590 REM *****
591 REM *****
592 REM *****
593 REM *****
594 REM *****
595 REM *****
596 REM *****
597 REM *****
598 REM *****
599 REM *****
600 REM *****
601 REM *****
602 REM *****
603 REM *****
604 REM *****
605 REM *****
606 REM *****
607 REM *****
608 REM *****
609 REM *****
610 REM *****
611 REM *****
612 REM *****
613 REM *****
614 REM *****
615 REM *****
616 REM *****
617 REM *****
618 REM *****
619 REM *****
620 REM *****
621 REM *****
622 REM *****
623 REM *****
624 REM *****
625 REM *****
626 REM *****
627 REM *****
628 REM *****
629 REM *****
630 REM *****
631 REM *****
632 REM *****
633 REM *****
634 REM *****
635 REM *****
636 REM *****
637 REM *****
638 REM *****
639 REM *****
640 REM *****
641 REM *****
642 REM *****
643 REM *****
644 REM *****
645 REM *****
646 REM *****
647 REM *****
648 REM *****
649 REM *****
650 REM *****
651 REM *****
652 REM *****
653 REM *****
654 REM *****
655 REM *****
656 REM *****
657 REM *****
658 REM *****
659 REM *****
660 REM *****
661 REM *****
662 REM *****
663 REM *****
664 REM *****
665 REM *****
666 REM *****
667 REM *****
668 REM *****
669 REM *****
670 REM *****
671 REM *****
672 REM *****
673 REM *****
674 REM *****
675 REM *****
676 REM *****
677 REM *****
678 REM *****
679 REM *****
680 REM *****
681 REM *****
682 REM *****
683 REM *****
684 REM *****
685 REM *****
686 REM *****
687 REM *****
688 REM *****
689 REM *****
690 REM *****
691 REM *****
692 REM *****
693 REM *****
694 REM *****
695 REM *****
696 REM *****
697 REM *****
698 REM *****
699 REM *****
700 REM *****
701 REM *****
702 REM *****
703 REM *****
704 REM *****
705 REM *****
706 REM *****
707 REM *****
708 REM *****
709 REM *****
710 REM *****
711 REM *****
712 REM *****
713 REM *****
714 REM *****
715 REM *****
716 REM *****
717 REM *****
718 REM *****
719 REM *****
720 REM *****
721 REM *****
722 REM *****
723 REM *****
724 REM *****
725 REM *****
726 REM *****
727 REM *****
728 REM *****
729 REM *****
730 REM *****
731 REM *****
732 REM *****
733 REM *****
734 REM *****
735 REM *****
736 REM *****
737 REM *****
738 REM *****
739 REM *****
740 REM *****
741 REM *****
742 REM *****
743 REM *****
744 REM *****
745 REM *****
746 REM *****
747 REM *****
748 REM *****
749 REM *****
750 REM *****
751 REM *****
752 REM *****
753 REM *****
754 REM *****
755 REM *****
756 REM *****
757 REM *****
758 REM *****
759 REM *****
760 REM *****
761 REM *****
762 REM *****
763 REM *****
764 REM *****
765 REM *****
766 REM *****
767 REM *****
768 REM *****
769 REM *****
770 REM *****
771 REM *****
772 REM *****
773 REM *****
774 REM *****
775 REM *****
776 REM *****
777 REM *****
778 REM *****
779 REM *****
780 REM *****
781 REM *****
782 REM *****
783 REM *****
784 REM *****
785 REM *****
786 REM *****
787 REM *****
788 REM *****
789 REM *****
790 REM *****
791 REM *****
792 REM *****
793 REM *****
794 REM *****
795 REM *****
796 REM *****
797 REM *****
798 REM *****
799 REM *****
800 REM *****
801 REM *****
802 REM *****
803 REM *****
804 REM *****
805 REM *****
806 REM *****
807 REM *****
808 REM *****
809 REM *****
810 REM *****
811 REM *****
812 REM *****
813 REM *****
814 REM *****
815 REM *****
816 REM *****
817 REM *****
818 REM *****
819 REM *****
820 REM *****
821 REM *****
822 REM *****
823 REM *****
824 REM *****
825 REM *****
826 REM *****
827 REM *****
828 REM *****
829 REM *****
830 REM *****
831 REM *****
832 REM *****
833 REM *****
834 REM *****
835 REM *****
836 REM *****
837 REM *****
838 REM *****
839 REM *****
840 REM *****
841 REM *****
842 REM *****
843 REM *****
844 REM *****
845 REM *****
846 REM *****
847 REM *****
848 REM *****
849 REM *****
850 REM *****
851 REM *****
852 REM *****
853 REM *****
854 REM *****
855 REM *****
856 REM *****
857 REM *****
858 REM *****
859 REM *****
860 REM *****
861 REM *****
862 REM *****
863 REM *****
864 REM *****
865 REM *****
866 REM *****
867 REM *****
868 REM *****
869 REM *****
870 REM *****
871 REM *****
872 REM *****
873 REM *****
874 REM *****
875 REM *****
876 REM *****
877 REM *****
878 REM *****
879 REM *****
880 REM *****
881 REM *****
882 REM *****
883 REM *****
884 REM *****
885 REM *****
886 REM *****
887 REM *****
888 REM *****
889 REM *****
890 REM *****
891 REM *****
892 REM *****
893 REM *****
894 REM *****
895 REM *****
896 REM *****
897 REM *****
898 REM *****
899 REM *****
900 REM *****
901 REM *****
902 REM *****
903 REM *****
904 REM *****
905 REM *****
906 REM *****
907 REM *****
908 REM *****
909 REM *****
910 REM *****
911 REM *****
912 REM *****
913 REM *****
914 REM *****
915 REM *****
916 REM *****
917 REM *****
918 REM *****
919 REM *****
920 REM *****
921 REM *****
922 REM *****
923 REM *****
924 REM *****
925 REM *****
926 REM *****
927 REM *****
928 REM *****
929 REM *****
930 REM *****
931 REM *****
932 REM *****
933 REM *****
934 REM *****
935 REM *****
936 REM *****
937 REM *****
938 REM *****
939 REM *****
940 REM *****
941 REM *****
942 REM *****
943 REM *****
944 REM *****
945 REM *****
946 REM *****
947 REM *****
948 REM *****
949 REM *****
950 REM *****
951 REM *****
952 REM *****
953 REM *****
954 REM *****
955 REM *****
956 REM *****
957 REM *****
958 REM *****
959 REM *****
960 REM *****
961 REM *****
962 REM *****
963 REM *****
964 REM *****
965 REM *****
966 REM *****
967 REM *****
968 REM *****
969 REM *****
970 REM *****
971 REM *****
972 REM *****
973 REM *****
974 REM *****
975 REM *****
976 REM *****
977 REM *****
978 REM *****
979 REM *****
980 REM *****
981 REM *****
982 REM *****
983 REM *****
984 REM *****
985 REM *****
986 REM *****
987 REM *****
988 REM *****
989 REM *****
990 REM *****
991 REM *****
992 REM *****
993 REM *****
994 REM *****
995 REM *****
996 REM *****
997 REM *****
998 REM *****
999 REM *****
1000 REM *****

```

[illegible]

played like he was and landing your Discovery to Open means you postage but don't change the problem, it's solved on Discovery but

He also reminds that his own files often get rid of this error by formatting an empty disk, but warns you to make sure that it is empty because "format" will

© 2004 Blackwell Publishing Ltd *Journal of Internal Medicine* 255: 111–118

[illegible]

and research and development
information can be the best

Abstract

Jim McMenis of BIRD 42 says he has trouble with an Alphadom printer and Discovery and asks if this is due to Discovery's port being non-standard. The answer is that the Alphadom is non-standard, some models more than others, but that they all contain their own certain say of interpreting the signals from the Spectrum and the Discovery interface with this. So you can use both effectively together.

1000

```

1  # Import the necessary libraries
2  import pandas as pd
3  import numpy as np
4  from sklearn.preprocessing import StandardScaler
5  from sklearn.model_selection import train_test_split
6  from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
7  from sklearn.svm import SVC
8  from sklearn.ensemble import RandomForestClassifier
9  from sklearn.linear_model import LogisticRegression
10
11 # Load the dataset
12 data = pd.read_csv('data.csv')
13
14 # Split the data into training and testing sets
15 X_train, X_test, y_train, y_test = train_test_split(data, data['category'],
16                                                    test_size=0.2,
17                                                    random_state=42)
18
19 # Standardize the features
20 scaler = StandardScaler()
21 X_train = scaler.fit_transform(X_train)
22 X_test = scaler.transform(X_test)
23
24 # Train the models
25 svm = SVC(kernel='rbf')
26 svm.fit(X_train, y_train)
27
28 rf = RandomForestClassifier(n_estimators=100)
29 rf.fit(X_train, y_train)
30
31 lr = LogisticRegression()
32 lr.fit(X_train, y_train)
33
34 # Evaluate the models
35 svm_pred = svm.predict(X_test)
36 rf_pred = rf.predict(X_test)
37 lr_pred = lr.predict(X_test)
38
39 svm_acc = accuracy_score(y_test, svm_pred)
40 rf_acc = accuracy_score(y_test, rf_pred)
41 lr_acc = accuracy_score(y_test, lr_pred)
42
43 print(f'SVM Accuracy: {svm_acc}')
44 print(f'Random Forest Accuracy: {rf_acc}')
45 print(f'Logistic Regression Accuracy: {lr_acc}')
46
47 # Confusion Matrix and Classification Report for SVM
48 svm_cm = confusion_matrix(y_test, svm_pred)
49 svm_report = classification_report(y_test, svm_pred)
50
51 print('SVM Confusion Matrix:')
52 print(svm_cm)
53
54 print('SVM Classification Report:')
55 print(svm_report)
56
57 # Confusion Matrix and Classification Report for Random Forest
58 rf_cm = confusion_matrix(y_test, rf_pred)
59 rf_report = classification_report(y_test, rf_pred)
60
61 print('Random Forest Confusion Matrix:')
62 print(rf_cm)
63
64 print('Random Forest Classification Report:')
65 print(rf_report)
66
67 # Confusion Matrix and Classification Report for Logistic Regression
68 lr_cm = confusion_matrix(y_test, lr_pred)
69 lr_report = classification_report(y_test, lr_pred)
70
69 print('Logistic Regression Confusion Matrix:')
70 print(lr_cm)
71
71 print('Logistic Regression Classification Report:')
72 print(lr_report)
73
74 # Feature Importance for Random Forest
75 rf_feature_importance = rf.feature_importances_
76
77 print('Random Forest Feature Importance:')
78 print(rf_feature_importance)
79
80 # Feature Importance for Logistic Regression
81 lr_feature_importance = lr.coef_
82
83 print('Logistic Regression Feature Importance:')
84 print(lr_feature_importance)
85
86 # Feature Importance for SVM
87 svm_feature_importance = svm.support_vectors_
88
89 print('SVM Feature Importance:')
90 print(svm_feature_importance)
91
92 # Feature Importance for SVM
93 svm_feature_importance = svm.support_vectors_
94
95 print('SVM Feature Importance:')
96 print(svm_feature_importance)
97
98 # Feature Importance for SVM
99 svm_feature_importance = svm.support_vectors_
100
101 print('SVM Feature Importance:')
102 print(svm_feature_importance)

```

WESTINGHOUSE **UD-500**

An anonymous reader from New Milford, Conn., mentions that Discoveries printer port has an interesting feature, namely that of sending question marks instead of user defined graphics. He points out that if program by our own Roy Lick (21 April 1984, p.63), originally written for that old computer interface world, equally well with Discoveries here it is, again (page 5) it merely replaces all the user defined graphics with appropriate letters, as follows:

Dume is an Amateur

the A and B of the first line are used in a program to send a jump to an Address (GPR 3000) in the BASIC listing (Figure 2), line 340 sets up the first loop and line 400 increments this by a small amount. Line 490 sets the printer for multiple density graphics and also to prepare to print 256*4, the number of graphic lines. The code is shown in Figure 3. The way that due to its relative inexperience, the only used the G, R, L and A registers, the program is not as short as shorter alternatives. Its also sets about peripherals, being forced to use the 394-4300M as the everything else will be used for

all right, particularly if one is using a 486 Spectrum, as the power supply is simple even for a 108 and most peripherals, provided that there is no clash, particularly over the XTAL line which is not continued through the Opus port.

Auto-run select

And last, I shall break my rule of not publishing any more C64 routines, because this is one with a difference: you remember Mr Nutting's program, Supercat, in last October's *IX* Computing? Well, by using his 'auto' code and modifying the BASIC, Alan Hopper has made a loader as simple that even his child of five can use it. The BASIC, listed in Figure 4, is saved as 'run' UNL 0000 when run, the program automatically fits on the BASIC program on the disc and assigns a number to each, saving the number of the required program then loads it and starts it.

Tailpiece

Opus has clearly abandoned the Discovery, but we will continue to keep this column going as long as there is sufficient interest. Keep the programs coming. See you next month.

Figure 3. Assembly listing for screen dumps.

40000	LD	A	,0
40001	LD	A	,0
40002	LD	A	,0
40003	LD	A	,0
40004	LD	A	,0
40005	LD	A	,0
40006	LD	A	,0
40007	LD	A	,0
40008	LD	A	,0
40009	LD	A	,0
40010	LD	A	,0
40011	LD	A	,0
40012	LD	A	,0
40013	LD	A	,0
40014	LD	A	,0
40015	LD	A	,0
40016	LD	A	,0
40017	LD	A	,0
40018	LD	A	,0
40019	LD	A	,0
40020	LD	A	,0
40021	LD	A	,0
40022	LD	A	,0
40023	LD	A	,0
40024	LD	A	,0
40025	LD	A	,0
40026	LD	A	,0
40027	LD	A	,0
40028	LD	A	,0
40029	LD	A	,0
40030	LD	A	,0
40031	LD	A	,0
40032	LD	A	,0
40033	LD	A	,0
40034	LD	A	,0
40035	LD	A	,0
40036	LD	A	,0
40037	LD	A	,0
40038	LD	A	,0
40039	LD	A	,0
40040	LD	A	,0
40041	LD	A	,0
40042	LD	A	,0
40043	LD	A	,0
40044	LD	A	,0
40045	LD	A	,0
40046	LD	A	,0
40047	LD	A	,0
40048	LD	A	,0
40049	LD	A	,0
40050	LD	A	,0
40051	LD	A	,0
40052	LD	A	,0
40053	LD	A	,0
40054	LD	A	,0
40055	LD	A	,0
40056	LD	A	,0
40057	LD	A	,0
40058	LD	A	,0
40059	LD	A	,0
40060	LD	A	,0
40061	LD	A	,0
40062	LD	A	,0
40063	LD	A	,0
40064	LD	A	,0
40065	LD	A	,0
40066	LD	A	,0
40067	LD	A	,0
40068	LD	A	,0
40069	LD	A	,0
40070	LD	A	,0
40071	LD	A	,0
40072	LD	A	,0
40073	LD	A	,0
40074	LD	A	,0
40075	LD	A	,0
40076	LD	A	,0
40077	LD	A	,0
40078	LD	A	,0
40079	LD	A	,0
40080	LD	A	,0
40081	LD	A	,0
40082	LD	A	,0
40083	LD	A	,0
40084	LD	A	,0
40085	LD	A	,0
40086	LD	A	,0
40087	LD	A	,0
40088	LD	A	,0
40089	LD	A	,0
40090	LD	A	,0
40091	LD	A	,0
40092	LD	A	,0
40093	LD	A	,0
40094	LD	A	,0
40095	LD	A	,0
40096	LD	A	,0
40097	LD	A	,0
40098	LD	A	,0
40099	LD	A	,0
40100	LD	A	,0

Figure 4. Assembly program.

1	LD	A	,0
2	LD	A	,0
3	LD	A	,0
4	LD	A	,0
5	LD	A	,0
6	LD	A	,0
7	LD	A	,0
8	LD	A	,0
9	LD	A	,0
10	LD	A	,0
11	LD	A	,0
12	LD	A	,0
13	LD	A	,0
14	LD	A	,0
15	LD	A	,0
16	LD	A	,0
17	LD	A	,0
18	LD	A	,0
19	LD	A	,0
20	LD	A	,0
21	LD	A	,0
22	LD	A	,0
23	LD	A	,0
24	LD	A	,0
25	LD	A	,0
26	LD	A	,0
27	LD	A	,0
28	LD	A	,0
29	LD	A	,0
30	LD	A	,0
31	LD	A	,0
32	LD	A	,0
33	LD	A	,0
34	LD	A	,0
35	LD	A	,0
36	LD	A	,0
37	LD	A	,0
38	LD	A	,0
39	LD	A	,0
40	LD	A	,0
41	LD	A	,0
42	LD	A	,0
43	LD	A	,0
44	LD	A	,0
45	LD	A	,0
46	LD	A	,0
47	LD	A	,0
48	LD	A	,0
49	LD	A	,0
50	LD	A	,0
51	LD	A	,0
52	LD	A	,0
53	LD	A	,0
54	LD	A	,0
55	LD	A	,0
56	LD	A	,0
57	LD	A	,0
58	LD	A	,0
59	LD	A	,0
60	LD	A	,0
61	LD	A	,0
62	LD	A	,0
63	LD	A	,0
64	LD	A	,0
65	LD	A	,0
66	LD	A	,0
67	LD	A	,0
68	LD	A	,0
69	LD	A	,0
70	LD	A	,0
71	LD	A	,0
72	LD	A	,0
73	LD	A	,0
74	LD	A	,0
75	LD	A	,0
76	LD	A	,0
77	LD	A	,0
78	LD	A	,0
79	LD	A	,0
80	LD	A	,0
81	LD	A	,0
82	LD	A	,0
83	LD	A	,0
84	LD	A	,0
85	LD	A	,0
86	LD	A	,0
87	LD	A	,0
88	LD	A	,0
89	LD	A	,0
90	LD	A	,0
91	LD	A	,0
92	LD	A	,0
93	LD	A	,0
94	LD	A	,0
95	LD	A	,0
96	LD	A	,0
97	LD	A	,0
98	LD	A	,0
99	LD	A	,0
100	LD	A	,0

Dear Sir,
I'd like to complain
about this advertisement
because

Most advertisements are legal, decent, honest and truthful. A few are not, and, like you, we want them stopped. If you would like to know more about how to make complaints, please send for our booklet 'The Do's and Don'ts of Complaining.' It's free.

The Advertising Standards Authority.
We're here to put it right.

ASA Ltd., Dept. Z, Brook House, Torrington Place, London WC1E 7HN

This notice is directed to the interests of high standards of advertising.

IN-FLIGHT



TOP GUN

Score
1746

Games based on top film have often been top flop! but this high flyer is set to reverse the trend and is soaring up the charts as a game in its own right.

You start in the game that pits you in a high speed aerial duel against either a human opponent or a series of computer opponents. The action takes place on a vertical split screen display with vector graphics representing the F4 Phantom fighters. I found the vertical split screen display worked exceptionally well and made the game a lot easier than the normal horizontal display (one player's screen on top of the other's) particularly for players hunched either side of the keyboard. The rate of course set (adjust if you lose them but single stick games may opt for the keyboard to make the battles fun if you're up against the computer) you'll need all the help you can get.

Control of the F4s is a mixture of the standard climb, bank and dive controls with additional keys to control the throttle and

change weapons while reading the instruments that report your speed, altitude, attitude, spinning or decelerating and plot your opponent's position on your radar. Unlike these displays one way to read even when travelling at speed is that's the way this games played.

The contest begins as both fighters take-off from their respective carriers and head straight for each other thanks to the simple flight controls you can concentrate on the combat and decisions on whether to attack with your machine guns or fire a sidewinder homing missile. A missile will wipe out your opponent in a single shot but you will have to keep him in your sights while your missile options lock on. To add to your problems and also your chances of survival you can also fire a flare to distract enemy fire but when you can only select one system at a time you have to anticipate your opponent's actions almost before he knows them himself. Even the simple

Tony Hetherington takes a spin in four of the latest Spectrum flight simulators.

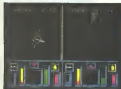
machine guns can cause you problems as they can overheat and jam before you can score enough hits to destroy him.

The two player game is a simple head to head contest where both players have a stock of three planes, the last plane flying wins the contest.

Against the computer your three planes (at first) must tactical a series of computer plots that increase in skill. You don't score any points, just survive to fight the next level or group of three fighters. The last two levels are relatively easy as the enemy mainly attacks with its machine guns and tail cone spray to your sidewinders but look them on if you get a lot harder as you learn a succession of missile firing goes, to win you will really have to be Top Gun.



GREAT





IN-FLIGHT

ACE OF ACES

US Gold
£9.95



ACE OF ACES



"The British aircraft will ever bomb Berlin," was the slogan of Luftwaffe chief Herman Goering (now, thanks to US Gold, you can join the ranks of RAF Mosquito pilots that not only bombed Berlin but also rescued trains, gunwheeled planes, sank submarines and even scored victories against the dreaded V-1 buzz bombs).

Your game begins in the briefing room with your possible targets marked on the Squadron leaders' blackboard. You choose whichever ones you like to go on a (possible mission or attempt the Ace of Aces grand slam) and go for the tail. However, before you can bomb submarines and trains while fighting Me109s, London-bound bombers and V's you'll need to learn how to fly this highly versatile fighter bomber.

The mechanics of flying are similar to those experienced by Lancaster bomber pilots in the excellent Panzerbuster game and consists of a series of screens each with their own set of controls. The plane is viewed from the pilot's screen that shows the view through the front of the aircraft as well as instruments to monitor the plane's height, speed and down and direction and height of any enemy "targets". The Mosquito has two Rolls Royce engines that are controlled directly from the port and starboard engine controls. These monitor the throttle and boost given to each engine and also contain a fire extinguisher in case the engines overheat. There are also flaps and landing gear controls that can be used in evasive tactics as they'll cause a sudden loss of speed (although you don't need them to land or take-off as the computer does that for you).

On course

Selecting the navigator's screen

punches up the mission's tactical map that plots the relative positions of you, the targets and any fast weather that can make flying even more hazardous and, of course, the dangerous Me109 fighters. Once you reach your target (on a pending mission) it's all up to the skill, accuracy and timing of the bombardier. Opening the bomb doors will hopefully reveal the target that can be destroyed by your bombs. This screen is also used to dump empty fuel tanks to lighten the load and to switch the forward gunner (pilot screen) from screen to screen.

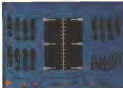
Naturally each mission is different and requires specialist loads, so you can customise your weapons load before you set off (adding extra bombs, rockets or cannon ammunition and extra fuel tanks for longer flights).

To finish your training you should fly a few practice missions to tune up your timing as you bomb the train before you to hit only the packages that don't contain POWs and submarines before they slip to safety.

In the real bombing mission it is vital that you get everything right as you only get one chance. The ideal approach is at 10,000 ft or anything under 10,000 feet but this might not be always possible if you've been damaged by Me109 attacks.

After the first get back to briefly in time for another mission. Your performance is based on a points system with high scores awarded for getting back as well as air targets hit and even bombs, fuel and rocket saved.

After a few successful sorties you'll be ready for a day's mission, perhaps defending London against V-1 and bomber attacks until finally you go for the grand slam and become one of the Ace of Aces.



MicroJet

ACROJET

Microprocess
27 19

More high flying action as man and machine are again driven to their limits, only this time it's a battle against the clock since you're competing in an extreme decoupage — jet events that require supreme control of your MicroJet 200mph jet aircraft if you are to take the championship and enter the Hall of Fame. Before you write your name into the record books you'll have to learn how to fly, this is undoubtedly the most difficult of this month's simulations to keep off the ground let alone trying to compete in the aerobically event. This test made my editor when you consider that the instructions were written for the Commodore 64 with a brief and extremely vague single sheet to explain the Spectrum level. Use joystick with Interface 2 or large 475K for controlling plane, guarantee initial decoupage flight, as you sort out the controls, and if dip switches out the most popular joystick interfaces.

The screen display is custom built for sport flying and contains a bank of instruments and a 3D view that also includes your aircraft as it loops and turns around the circuit. A plan of the course which also doubles as a radar combines with a compass to keep you on course as you negotiate your way around the hazards.

The first event is a relatively simple task to enter four points before returning to the starting take off strip. In the Slalom race things are a little trickier as you have to enter the same points but in a set order and in the correct direction.



If flying is still more luck than judgement then you should quit now and be grateful for the points you've already got because the next event involves cutting a 3" ribbon string between two poles, to show that much as there you have to cut more ribbons while flying upside down. Then in the ribbon roll you must cut one ribbon roll over and get another without touching your wings on the ground. If that was too easy why not try the ribbon low level race, the loop and Cuban eagle which contains everything you've liked so far and the Income spot landing in which you must deliberately stall your engine at 5000 then land with pinpoint accuracy if you turn on.

your engine you lose points but if you don't you might lose your plane.

Finally you're awarded points even if you don't finish the course so even beginners can compete with each other.

Up to now I've perfected my own events including the crash of the engine, the runway, the under the pylons and head-to-head event. I'll take a lot more practice before I can take on Microprocess's own Major Six Sixty.



DEEP STRIKE

Event
27 19

World War II dogfights were extremely hazardous with planes often shooting off their own propellers in the heat of the action. Add to that the unreliability of such early aircraft and you realise the danger of the mission you've been sent on. Your mission is to fight off hordes of Red Bombs, Black Bombs and Blue Bombs to protect the Bombers that fly in front of you.

This aerial conveyor has its own problems in that you have to dog the bombs while trying not to shoot the Bombers that spend

most of the game in front of your gunsight.

The screen shows your view over the scrolling landscape and your controls that are dominated by two joysticks which show the deteriorating condition of your plane and a pointer that keeps you on course.

The best results seem to be achieved if you fly so that you're just below the bomber's undercarriage but be prepared to react out to intercept enemy fighters and gun down enemy balloons before they destroy your Bomber. As each bomber is destroyed another takes its place until you either get a chance to bomb your main target (the enemy fuel dump) or you're left on your own to take out as many of their planes as you can before they get you.





MEMORY EXPANSION PROGRAM

Toni Baker shows how to squeeze some extra memory out of 16 and 48K Spectrums.

PROGRAM

```
10 CLEAR 32767
20 LET X=02768
30 FOR I=0 TO 14
40 LET C=0
50 FOR J=1 TO 4
60 READ A
70 FORS X,A
80 LET X=X+4
90 LET C=C+8
100 NEXT J
110 IF A=C THEN PRINT "ERROR"
120 LINE "110"=0:148: STOP
130 NEXT I
140 RANDOMIZE USR 32768
150 STOP
160 DATA 22,8,8,17,7,14,62
170 DATA 3,63,88,62,73,76,379
180 DATA 32,78,79,79,76,32,344
190 DATA 62,2,285,1,32,1,273
200 DATA 7,8,17,8,138,285,337
210 DATA 68,72,4,44,177,1,748
220 DATA 11,8,17,7,128,285,748
230 DATA 68,32,192,16,243,17,54
240 DATA 3,7,285,84,31,218,544
250 DATA 123,27,118,118,26,281,81
260 DATA 134,38,283,135,213,122
270
280 DATA 285,255,54,238,248,179
290
300 DATA 33,8,88,17,1,32,337
310 DATA 1,255,2,119,237,176,79
320
330 DATA 288,24,219,8,8,8,452
```

One upon a time, when I was a lot younger than I am now, I was privileged enough to own a calculator. It wasn't anything flash, you understand, because calculators had only just come onto the market — but it was good enough for me at the time. It did add and subtract, multiply and divide. All very simple stuff I grant you, but fun at the time.

A friend of mine also had a calculator. But was a more expensive version. It was made by the same company, but was the real one up in the range. It had an extra column of buttons on the right, and was coloured black instead of white. The extra buttons were things like M+, M—, MR and so on — all memory functions, allowing you to store and recall numbers and so on. Then, one day while we were playing a casual game of throwing calculators around-the-house (as children do), the case of my friend's calculator came open, revealing a panel of little rubber buttons on one half, and a grid of wires on the other half. It became

immediately apparent to me that pushing down one of the rubber buttons would make contact between a vertical wire and a horizontal wire, and hence register the key depression. It was then that I realised an inconsistency — there were more vertical wires than there were columns of keys, giving an extra six possible contacts.

Surprise surprise!

Imagine my surprise when, with the case still open, I pressed on the extra contacts to see what happened — only to find that they performed the M+,

M—, MR (and functions of the more expensive version). It seemed that the internal hardware of the two calculators was identical, but they were put into two different external cases so that they could sell at two different prices for differing sections of the market. I suppose it must have been cheaper for them to do that than to produce two entirely different machines.

With that little anecdote in mind, we return to the present. Calculators have long gone out of fashion — today, computers are the thing. I began to wonder, however, whether what was true of calculators in the early days, might also be true of computers now. Could it be possible that inside the various different boxes there was basically one, and only one, type of Spectrum?

Literally speaking, of course, this cannot be true. The parts seem to give different results for different sizes of Spectrum, but the central problem remains, it is not possible that the advanced features of the Spectrum 128 (extra memory, full screen editor, and so on) are, and have always been, available on the old 48K and 16K Spectrums?

It turns out that it is not the case. Both 48K and 48K machines, now in fact already got 128K of RAM built in — though accessing it has not been easy. On the Spectrum 128, outputting a byte to port 1770 was sufficient to page in the extra memory. On the 48K and 16K machines this is not so.

It is possible, however, to devise a machine code program which will give genuine access to the extra

RAM and ROM memory. Such a program is included in this article. It is a BASIC program with the machine code hidden DATA statements. The BASIC program — essentially a machine code loader, will ROM the machine code into memory and then, by means of a GOSUB statement, run the machine code itself. The last item in each DATA statement is a checksum to ensure that the line has been typed in correctly. If the checksum does not match with the rest of the line then an error message is generated, indicating the line number of which the DATA was incorrect.

Once the program has been typed in, with each DATA statement error free, then the program may be RUN. There will be a short delay whilst the machine code is loaded into memory and then the computer will effectively turn itself into a Spectrum 128, and will remain so until the power is switched off.

Reset

The effect of hitting this program is equivalent to pressing the RESET button on a normal Spectrum 128. The screen goes black with a white border for a few seconds

before you are presented with the main 128 menu. A cursor bar outlines the first item on the menu, DATA LOADER. Ignore this for the moment! Using the CURSOR DOWN function (CAPS-SENTE) if you don't have a Spectrum+ with a separate CURSOR DOWN key) move the cursor to the second item, 128 BASIC, and your machine is every bit as good as the 128 itself.

I would like to stress at this point that the program will convert a Spectrum to a Spectrum 128 by software means alone! This is important, because it means that the guarantee is not invalidated!

Of course the main problem with converting a Spectrum by this means, is that it doesn't happen on instruction book. It is not within the scope of this article to give detailed information on 128 BASIC, and furthermore, neither Sinclair nor Amstrad will provide an instruction book without supplying a computer as well. The only thing I can suggest is that you find a friend who's got a Spectrum 128, and borrow their instruction book. If you don't know anyone who owns a Spectrum 128 then it might be worth your while joining a local computer club. Most of the extra money in a

Spectrum 128 is locked up in a software down-loading system called alpha disc, or 128alpha. It is worthwhile getting hold of an alpha version book, as the simple reason that without it you won't know how to access the extra memory in 128 BASIC.

MIC

Now the machine code program works is surprisingly simple. As has been previously stated, an output to port 3470 alone is not sufficient. The Spectrum contains two new registers, which to my knowledge have never been properly explained. They are referred to as Accumulator Programming Registers, and it is the first of these (APR1 for short) which coordinates between the various different hardware models of the Spectrum. Access to the chip is by a complex sequence of OUT instructions. This sequence is performed automatically by all Spectrum whenever the machine is either switched on or reset so that the machine knows what kind of Spectrum it's supposed to be.

It takes it to you now to deftly load in the program and try it out. Good luck, and I'll see you again next month.

It's easy to complain about advertisements. But which ones?

Every week, millions of advertisements appear in print, on posters or in the cinema.

Most of them comply with the rules contained in the British Code of Advertising Practice.

But some of them break the rules and warrant your complaints.

If you're not sure about which ones they are, however, drop us a line and we'll send you an abridged copy of the Advertising Code.

Then, if an advertisement bothers you, you'll be justified in bothering us.

The Advertising Standards Authority ✓
If an advertisement is wrong, we'll have to put it right.

ASA Ltd Dept 2 Stock House, Tottenham Place, London WC1E 6HN

TIDYBINDERS

SIMPLE, SMART
AND TIDY

£5.30 (including postage
and packing)

Build up your Wham Video reference library and be sure of finding your head quickly on your back issues of Wham Video by filling them in a Tidybinder, designed to hold 12 issues and smartly blocked with the Wham Video logo.

ASP Products Services, PO Box 26,
Walley House, Walley Road,
Hemel Hempstead, Herts HP1 1BB
(0442-89511)



Order Form 2X Compulsory

Please supply: ☐ Order is for year(s):

1 issue(s) £ (please make cheque payable to ASP Ltd)

Name

Address

Date

I am enclosing the necessary payment. Account No.

Check by postcard

ZXCONTACTS DIRECTORY

Who to call if you want
your Spectrum
repaired... the latest
game... a new
peripheral... It's all in
this indispensable
contacts guide

Before we start talking
telephone numbers a brief word
about the services which are
available through ZX, the try to
provide as full a service to our
readers as possible and to get
the most from us follow the
guidelines that are outlined
here.

Back issues

Back issues are handled by a
company called Inform, Times
House, 179 The Marshes, Ham-
mington, Hants HP1 9BB. If you
have missed an issue copies
may still be available from
Inform, price £1.50+50p towards
postage and packing. Copies
are still available for all ZX
Computing Members but copies
of the special bi-monthly issues
are only limited.
Inform 0442 48428

Subscriptions

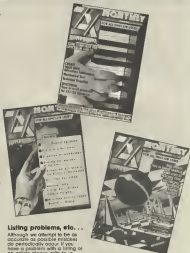
All letters relating to
subscriptions are handled by
Inform. If you want to subscribe
at a special discount price (£3
off the usual annual sub price)
write to the subscriptions savings
department at the end of this
directory but hurry as this offer
ends on April 30th 1987.

Photocopies

If you want to get hold of an
article from a past ZX there is a
photocopy service available,
price £1.50 per article. Send
request to the Golden Square
address marking your envelope
clearly 'Photocopy service'.
Cheques payable to ASP
Limited.

Binders

ZX binders are available price
£5.20 inc p&p from ASP Binders
Services, PO Box 35, Watney
Road, Watney Road, (Hemel)
Hempstead, Herts HP2 4BA. 0442
47021. Cheques payable to ASP
Limited.



Listing problems, etc...

Although we attempt to be as
accurate as possible mistakes
do periodically occur. If you
have a problem with a listing or
an article please write to us
rather than phoning. If a
mistake has been spotted then
we will send you a correction
sheet. If not we'll do our best to
help. If you wish to correspond
with a contributor over a
problem with an article, write to
ZX Computing, marking your
envelope clearly with the
contributor name. In all cases
please enclose a SAE and mark
your envelope clearly with an
indication of its content, to the
title of the listing. It's simple
easier to process that way and
you will receive a reply sooner.
We try to be prompt in
responding to letters but please
understand if you suffer a delay
in receiving a reply.

Future services

In the near future we hope to
launch a Software Service
offering ZX readers the chance
to purchase tapes at cost that
have appeared in ZX. This is still
in planning stage but if there
are listings or series from ZX that
you would particularly like to
buy in tape form drop us a line,
and like to hear from you — it
would help to decide the first
few releases.

Ryan Ralph, Editor.

ZX Computing Monthly, Hot
Golden Square, London W9B
3AA (or 427 0624).

REPAIRS

When your Spectrum decides to go on the blink, who do you turn to? ZX focuses on professional repair companies.

■ If you're one of the people who got a Spectrum this Christmas then it's quite likely that you're still at the stage where you're totally infatuated with the thing. Well it's time I put a stop to that.

It's one of those inevitable facts of life that your home computer like any other household gadget may one day blow a gasket and leave you in the lurch just when you need it. That's why we've listed all those repair companies in this issue but just what are the facilities that the Spectrum is heir to?

We spoke to some repair services and it seems that all sorts of things have happened to machines at one time or another from the fairly common case of coffee getting knocked over them, to a TV aerial getting struck by lightning and the shock going right through the Spectrum where it blew out fairly close to the magazine.

Self-inflicted problems

However odd it may sound, it does seem that a lot of what goes wrong with Spectrums is, as one company told us, "repeatedly self-inflicted". There are certain problems, such as keys ceasing to function that just creep up from time to time and they're plain bad luck, but there is one cardinal sin that you can (and should) always avoid. **Never plug anything into, or remove anything from, the Spectrum's peripheral port while the power is switched on.**

I know that sounds a bit obvious but everyone who's been to a repair shop will agree that misuse of the edge connector is still the number one cause of dead Spectrums. Barry Clayton of Microcare in Bristol told us:

"There are some problems that are very Spectrum specific. They're caused mostly because the Spectrum was built along a low-cost specification. It often things that could be avoided by having a few more of them, as



on the cost." As a result, the Spectrum doesn't have some of the safeguards that are built into more expensive computers and other types of electrical appliances, and there are no buttons protecting the chips, which are the heart of any computer from the raw electrical supply. On the other hand, as Barry Clayton added, "because it's a low-cost machine it's that much easier to repair, so it's a case of things and roundabouts."

Another similar problem that can be avoided is when the alignment key, an in-socket bolt on, retaining a Bad connection when the interface is plugged in (if you take off the Spectrum's edge connector you'll see a little notch cut out of

the left-hand side, this matches up with a little ball on any interface that is designed to use with the machine. This is the alignment key, a fairly cheap sort of ensuring that the proper electrical connection is made when using any peripherals).

Having a go

Then there is what's known as "problems having a go". The Quantum repair service told us that quite a lot of things go wrong the moment people start to take the cover off the machine. "Don't touch it — a computer in complete is 50 times more complex than a car engine and there's very few people who could take one apart."

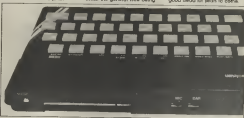
Keyboard problems crop up from time to time, and the quality of the keyboard was always one of the things that Sinclair machines were criticised for in the past. Instead of using "proper" typewriter style keyboards, Sinclair opted for a cheap, pressure sensitive membrane. The trouble with that was that if you constantly chose a particular key that got used more than the others (i.e. for

example, you always chose the same key as the 'Enter' key in arcade shoot 'em ups) then you could literally wear the membrane out. This led to the large market in add-on keyboards. When people getting their machines repaired decided that for a few pounds extra they might as well go the whole hog and buy themselves an entirely new board.

That about covers the problems to which the Spectrum is especially prone. Peripherals like the microdrive and Interface 1 are a different kettle of fish altogether, although some of you might be surprised to hear that repair services don't seem to have too much trouble with these, the general view being

that they're OK "if treated properly".

So, despite the past reputation that Sinclair machines had for reliability in their early days, the repair companies seem to agree with Sir Clive's own claim. Not much of the problem was due to the wear and tear inflicted on the machines by their owners ("blowdown" was the word one company used), rather than poor quality control. The Spectrum may not have that heavy safeguard built into it and any mess produced here will throw up faulty units long time to time, but as long as you don't blow it up yourself your Spectrum should stand you in good stead for years to come.



The edge connector: the most vulnerable part of your Spectrum

UK Repair Services

AL COMPUTERS AND ELECTRONICS	01-528-7149	GAUWCE COMPUTERS	0295-35240	NORTH DEVON MICRO CENTRE	0371-440269
AGE SURF	0413-398266	GATER COMPUTERS	0379-63327	VIC CD/AS	01-403-1988
BUILDING COMPUTERS	0748-368327	H.L. COMPUTER SERVICES	0772-632666	VIDEO WALT	081-236-0376
G. BUNGE AND SON	0628-661996	IT WESTERN (ELECTRONICS)	0225-702077	ONE STOP BYRONO	0803-663796
CAPITAL COMPUTER SERVICES	0203-641801	MANCOMP LTD	0472-241884	RIAC	0429-233189
CHILDREN COMPUTER CENTRE	0582-455648	MCE SERVICES	0623-39022	FROMPT	01-836-7166
COMPUTER HOSPITAL	0584-43354	MICRODOT	0213-382649	QUANTUM	0203-24602
COMBINATION COMPUTERS	0203-57679	MICRODARE	0273-46876	R.A. ELECTRONICS	0803-66285
FAST BYTE	0332-365280	MICROMIND	0733-247016	SCREEN SCENE	0242-626779
SHAD COMPUTING	0772-561952	MICRO POWER LTD	0532-458800	SPEARHEAD COMPUTERS	0208-536209
		MICROSEN	0226-737110	SPECTRUM HOSPITAL	0602-751163
		MICRO SERV	0236-730716	TECH-INFORM SERVICES	0734-543460
		MICROTECH	0204-396136	THEFORD MICRO	0776-66266
		MICROWORKSHOP	0410-654974	TV SERVICES OF CAMBRIDGE	0223-319071
				ZEEM COMPUTERS	0234-213646

HARDWARE & PERIPHERALS

Where to find the right
odd-one and
accessories.

ADAPT ELECTRONICS 01-554-2840
Only the 485 versions of the Spectrum don't have an RGB output that allows you to plug it into a monitor. So if you want to use one you'll need Adapt's RGB interface.

ADVANCED MEMORY SYSTEMS 0703-412602
Makers of the AMX Mouse.

CHETAN 0323-777327
System interfaces, connectors, you name it. Chetani make it but they are best known for their range of musical odd-ones such as the Speedyrum, Sound Sampler and their new MXS Midi Keyboard.

CLASSIFIED PRODUCTS

0225-345434
Suppliers of all sorts of cables and extension leads.

DEAN ELECTRONICS 0344-883444
Produces the solo supporters of external paper for the Aphorism 32 printer.

DESIGN DATA 0208-880004
As well as producing but offering every sort of interface that you Spectrum is ever likely to need, they also do a neat little mouse.

EAN ELECTRONICS 0202-880024
Have started out by making joy sticks interfaces but they've recently branched out with the innovative RomWrite printer interface (with built-in word processor) and the Music Machine 32bit interface that doubles as a sound sampler and drum synth.

ROCKSTAR PRODUCTS

01-223-0491
Newcomer to the Spectrum market with their Disciple unit which is a combination disc, joystick and networking interface.

ROMANTIC BOBOT 01-224-8870
Producers of the Muntio One unit and now the Muntio 128.

SAGA SYSTEMS 0443-22972
Now the odd-one keyboard market covered with a whole range of keyboards at different prices. They also do a nice line in printers and wordprocessing equipment.



St. Oliver's new 128 package.

TECHNOLOGY RESEARCH

01-876-3847

TRANSFORM 01-876-4300
Spectrum don't actually produce anything themselves, but they are one of the best suppliers of printers, monitors, utilities and such like, and they are also one of the few places that can still supply microdrive cartridges.

TROJAN 0703-225491

Makes one of the better Spectrum lightpens.

VIDEO TALK 044-224-8231

General repair company who also supply the Spectrum's Spectravite+ upgrade kit and the old 128 monitor keyboard — neither of which are generally available these days.

Where are they now?

AMSTEAD CONSUMER ELECTRONICS 0237-228888

They changed the name of the Spectrum with the 128+2, will they change direction with the +3?

CAMBRIDGE COMPUTERS 0223-112216

Followers of St. Oliver will undoubtedly want this number now that he's re-emerged with the new 288 package which will initially be available by mail order only.



The Spectrum 128+2.



SOFTWARE

A directory of British software house contact numbers.

ACTIVISION 01-431-1131

The UK branch of a large American outfit so most of their stuff is extensions of old games such as RoboRaid, Harker and their all-time buggy Ghostbusters.

ADGITIVE 0203-286604

ALLIANCE 0743-255796

ADVANCE SOFTWARE 0279-412641

New software Focus made impressive debut with Hareball.

ADRIUS FREE SOFTWARE 0543-755796

Software conglomerate that makes a habit out of acquiring up smaller companies. Quixote, Luthien, Bug Byte and A's F are all members of the AFS stable.

ANGLASOFT 01-638-3411

BUNSLAVE 0710-250462

COS 01-638-0363
Strategy/Wargame genre particularly big business in the UK market, but COS have managed to specialise in them without going bust. Their latest release, Vulcan is reviewed elsewhere this issue.



CODEMASTERS 0276-68426

Providing new budget label formed by ex Mastertronic programmers.

COS 01-638-0363

Strategic last year on publisher of good quality adventures such as the Bege and Crusade.

CSD 0232-433300

DOMARK 01-947-9426

Masters of the licensing deal. Domark have been responsible for such software classics as Friday the 13th, View to A Kill and the Christmas success, Devil Hunt.

DURELL 0823-24489

THE EDGE 01-631-1801

ELECTRIC DELTA 0703-226444

Another outfit that's keen on licensing deals. Electric Delas tend to concentrate on film licenses (Look to The Future, Howard the Duck etc), though only the recent Alien really did justice to the original film.

ELITE 0623-68662

FIL 021-6572661

The arcade label launched by Gorgyle Games who had previously concentrated on adventure games. There's only been one FIL release so far the cracking Lightforce, but there's more to come.



GARIBOLDI GAMES 0215-573981

GAULTEP 0444-732765
Gaultep's entire reputation and business is based on the success of their adventure creating system The Gault, which has spawned a whole mini industry of budget adventures. The Gaults gaining a bit of notoriety, so watch out for the arrival of the new Professional Writing System.

GETHUR 0245-759403
The software house that Monty Mole built.

HEWSON 0235-833716
Andrew Hewson's outfit has built up a solid reputation based on good games. The latest is Steve James' Barbarians.

IN SOFT 0655-794491
Used product is the ZX Beta Compiler.

INCENTIVE 0734-591678
Mothers of the Grapho Adventure Creator.

LEVEL 9 0734-594789
With no competition in sight, level 9 took out top software category their Middle Earth and Silicon Dragon trilogies are landmarks in the field. These days they're spreading their wings by doing software with other companies, such as Knight One, due from Rainbow in April.



MADEIRONIC 01-377-6890
The UK's top budget software house, so successful in fact that they've just gone and bought out the ones around it.

MELBOURNE HOUSE 04-543-0791

MIRROSOFT 0344-427917

MICROSPHERE 01-663-6441

MIRROSOFT 01-377-6830

OCEAN 0419-853688
High output of coin op conversions and spin-offs. Notable games recently Coded and Great Escape.

PALACE 01-378-0784
A subsidiary of the Palace Films/Voice outfit, Palace have tended to stick to a small number of good releases such as the Countess games and the recent Antares, though they may be expanding in the future.

PERMANE 01-338-6633
An offshoot of the Macmillans publishing company. Best game to date is the quality Rogues.

PIS 023-5447864

SOFTWARE PROJECTS 091-3389393

Since making Matthew Smith's feature with Martin Slater and Jan Set Wiley, they've had what is kindly referred to as a low profile, though their Dragon Isle conversions show that there's life in them yet.

TRISHAM 0223-438508
Masters of multiplatform too — unless such as the Copy for Pans, but most notably forward now in its third incarnation.

TELECOMSOFT 01-245-4838
Telecomsoft is the little bit of British Telecom that owns or controls Data, Regional, National and Rainbow (who are conveniently omitted in this issue).

U.S. GOLD 021-3442188
Close runners to Ocean in the big league, U.S. Gold have got a variety of titles under their wing, including marketing deals with Vortex, Ultimate and Adventure as well as a seemingly endless flow of US software waiting to be converted onto the Spectrum.

VIRAGE 01-634-8348
The software arm of the Barmen empire. Last week snapped up Leisure Genius and thereby the rights to titles such as Scud Race, Monopoly and Scudwin.



BIG

SUBSCRIPTION SAVINGS FOR YOU!

Take advantage of these fantastic money saving subscription offers to these magazines. Hurry, this amazing offer is for a limited period only.

UNITED KINGDOM

OVERSEAS

(Distributed outside only)

	Normal Price	Save Price	Please Tick	Normal Price	Save Price	Please Tick
ASB Computing	£24.00*	£18.00	☐	£28.00*	£20.00	☐
Computer Gamer	£18.00*	£13.00	☐	£24.00*	£17.00	☐
Your Commodore	£16.00*	£11.00	☐	£24.00*	£17.00	☐
ZX Computing Monthly	£18.00*	£13.00	☐	£18.00*	£14.00	☐
Citizens' Band	£18.00*	£13.00	☐	£22.00*	£17.00	☐
Ham Radio Today	£13.00*	£14.00	☐	£21.00*	£16.00	☐
Electronics Digest	£27.00*	£ 8.00	☐	£14.00*	£11.00	☐
Electronics Today International	£24.10*	£14.00	☐	£22.00*	£18.00	☐
Video Today	£24.00*	£13.00	☐	£24.00*	£18.00	☐
Which Video?	£18.00*	£15.00	☐	£24.00*	£18.00	☐
Photography	£18.00*	£12.00	☐	£24.00*	£18.00	☐
Photoplay	£14.00*	£14.00	☐	£24.00*	£17.00	☐
Clocks	£20.00*	£24.00	☐	£24.00*	£20.00	☐
Woodworker	£18.00*	£13.00	☐	£20.00*	£18.00	☐
Popular Crafts	£22.00*	£14.00	☐	£24.00*	£17.00	☐
Woodsman & Bowyer	£18.00*	£10.00	☐	£22.00*	£18.00	☐
Aeromodeller	£24.10*	£20.00	☐	£20.00*	£22.00	☐
Model Modelling	£18.00*	£13.00	☐	£24.00*	£18.00	☐
Model Boats	£24.10*	£13.00	☐	£20.00*	£18.00	☐
Radio Control Model Cars	£24.10*	£14.00	☐	£24.00*	£17.00	☐
Model Engineer	£27.00*	£22.00	☐	£24.00*	£20.00	☐
Radio Control Boat Modeller	£ 8.00*	£ 7.00	☐	£13.00*	£ 8.00	☐
R C M & E	£18.00*	£12.00	☐	£20.00*	£18.00	☐
Radio Control Scale Aircraft			☐			☐
Quarterly	£ 8.00*	£ 8.00	☐	£13.00*	£ 8.00	☐
Radio Modeller	£24.10*	£13.00	☐	£20.00*	£18.00	☐
Sea Classics International	£10.00*	£ 8.00	☐	£20.00*	£10.00	☐
Scale Models International	£18.00*	£13.00	☐	£20.00*	£18.00	☐
Your Model Railway	£14.00*	£12.00	☐	£20.00*	£20.00	☐

(Offer ends 30th April 1987)

Please counter my subscription(s) with the

cheapest my cheapest order for 8

Made payable to Argus Specialist Publications Ltd

to date

from my account/banking card to

valid from

Name

Address

Send this form with your remittance for: **Subscription Savings Offer (S.O.S.)**

1980/87 178, Times House, 178 The Marston, Royal Wootton Bassett, Wilt. SP1 1BB.

Brian Becker looks at the latest QL software.

IBM Microline continue to produce new products, although it seems as if there are less sales each time. The only coverage to come out of this is the slight note of walking around to see what is on offer without all the pushing and shoving through hordes of game nuts. There wasn't much new in the way of the QL either at the most recent Microline, but one new is good new these days and it is impressive how new products keep coming despite America having officially ended the competition.

Digital Precision worked all night and had the long awaited **Turbo-charge**, son-of-Super charge image compiler out on the day. Although my review copy arrived a day or so after the show I am sadly unable to say much more about Turbo than I said six months or so unfortunately I was sent the wrong size disk.

I did, however, get Digital's first benchmarks for Turbo and can pass them on as hot-off-the-press news. According to Digital, average speed-up factors for compiled code are (1) Integer benchmarks: 441 for Turbo compared to 211 for Supercharge; (2) Floating point benchmarks: 212 for Turbo compared to 173 for Supercharge; Digital's estimates for its computer generators are 212 and 21 respectively for a "Superbasic" language (Project Planner); Digital's estimates are 441/200 for Supercharge 212 and Turbo-charge 212. For comparing compiler speeds, Digital takes 441/200 for Supercharge 212 and Turbo 212. Digital was over the moon about the results and — up to the last minute — the management and designers did know how good a product they had developed. They expected Turbo to be as fast or faster than Supercharge, but confessed to pure delight when the results came in. Turbo looks an excellent piece of software and the company deserves every success with it.

Making headlines

The **Front Page** is not just the title of a classic play and a good movie but also a package of QL software offering desktop publishing for a mere £22.50. Produced by G&P Software, the **Front Page** is a desktop program that is fully menu-driven and enables the user to produce text and graphics layouts freely and easily. It comes on one

microdrive and works with an unexpanded QL, but is designed to take advantage of disc, disk drives, and so on if you have the facilities. The **Front Page** is designed to offer a reasonably powerful desktop publishing at a bargain basement price to those who want to produce their own news letters, handouts or wallpaper.

G&P's MB had sent my QL, obviously a few months back and told me that — while he would defend to the death my right to say it — I ought to be taken out and shot. He had noted a growing QL enthusiasm among British groups, school groups, and such like and I pointed out that this could be due to the fact that the QL had reached rock-bottom prices in the last few months. This coincided with some of the truly excellent

sales plus the typewriter-like keyboard. You just take off the old QL top and in the new MB when (the keyboard comes with a clear instruction sheet) and the new keyboard falls. After replacing the screen, you're ready to go. It is an impressive product and a good bargain. There is nothing to worry about — the only tool you need is a small Phillips-head screwdriver — and the result gives you a QL with a keyboard of professional quality. The image drawback I found with Sir Charles keyboard was the tendency to hit two keys at the same time but Sir Charles' replacement quids this and you can type with an ease that is a pleasure by comparison. The desktop layout is exactly the same (the function keys are in red but the rest is in black that we all know and love) and

Q COLUMN

software example (including the **Front Page**) makes the QL an attractive proposition to low-budget groups and others outside the IBM style. At the end of the day he began to think for being a journalist — especially after I said that I was fully behind all small QL companies and users — and we parted as friends. But the QL is dead and — unless somebody pulls off a minor miracle with Amstrad — it's going to stay that way. This doesn't mean that the machine won't be useful to a lot of people for a long time or that its portability to produce any more new QL products but only that it's out of the race in the marketplace.

Speaking of new products, the keyboard-driven are getting better. There was one selling for around £30 at the Microline. It just gives the old days a better feel and, frankly, I don't think its worth £22 so I don't plan to say any more about it. Schön has won the race to get their first and its replacement boards are now available. They will be just over £40 and you get a new QL housing with a new set of LED

the board is compatible with all peripherals and software.

QL Casino

If you're running a small casino for your family, neighbours and anybody with a few quid to lose, Pyramid has just released **Super Crapsler** at £15. You play against the QL, but there's nothing stopping you from letting the computer be the house backed up by your cash — but this would be illegal and — as QL centres are on the wane — the point is of academic interest only. There are its partner on the program suite: **Blackjack**, **Baccarat**, **Craps**, **21**, **21**, **21**, **21** and **21** machine. Fortunately I like the Poker and the **Baccarat** because I've never had any luck with **Blackjack** or **Reunion** in real casinos. **Baccarat** is a very enjoyable game of can also be a costly one and it's nice to have a computer program to play against in a very good package (you even get Snappy running around the screen to help out and we'll write the money).

gold Line RememBer program is without doubt one of the most useful programs that anyone could wish to add to their collection. However, the problem with most utilities is that they demand a fixed position in memory and usually require a small BASIC program to operate them. This is not the case with this program — because it can be loaded to any convenient position in memory at any time it also contains routines to tell itself where it is located in memory and to remember its own variables accordingly. It also has its own input routines, dispensing with the need for a BASIC control program.

The routines in the main part of the program have been designed to cater for every imaginable situation, all forms of BASIC (SPECTRUM, BASIC 80000) and SWE. Line statements can be unnumbered (indirect), indirective, even what in complex conditional form.

Add to this the data for

various registers, and the original remembrance routine, which is a mere thirty-three bytes in length, goes into a convenient larger 1540 bytes.

Before we can go one further, however, it will be necessary to enter the HEX 00000 program and the machine code. The 00000 program has been designed to wipe out the process of entering the machine code in error-free, selecting any lines that are not correctly entered, or in the wrong order.

The HEX 00000 is arranged in 101 lines of eleven pairs. The first nine pairs represent the actual machine code instructions, whilst the last two pairs contain a checksum of the values. In the particular line, if any other method of input is used you should remember to ignore these last two pairs.

Each line of eleven pairs should be entered as a continuous string, with one space between each pair. The lines are linked in convenient blocks to assist with this. Besides during input the address to be inserted in Line 4 of the hex loader must be the location of which you have decided to enter the machine code, say 40000 for 48K machines. Line 5 sets RAMTOP to one less than this address. When loading has finished the program should be saved on tape by using **SAVE "rememBer" 00000, 1540**.

You can save experiment with the program by reentering the HEX 00000 program (see RANDOMISE USE 10000) for whatever address you have

chosen and, if everything is in order, a prompt will appear at the lower part of the screen requesting you to **ENTER STARTING LINE, STOP SIZE**.

This requires you to enter two values requested by a comma, followed by ENTER. The values entered must be such that the STOP SIZE is not greater than the STARTING LINE, although the two can be equal. Otherwise you can enter any values you like — 0.1 — 99999 — 99999 etc. If you type in rubbish, illegal values, or 00000 with no value at all, the system will automatically use the values 0000.

Directly the renumbering process has been completed (in a fraction of a second with an average program) the program list will appear on the screen, where it will remain until a key is pressed. The only restrictions concerning the use of the program are that the first line of a program to be renumbered must contain a REM statement, and corresponding addresses must exist for any commands involving line numbers. A statement like GOTO 100 would be ignored if Line 100 does not exist.

Normally, assuming that the program is totally devoid on outside, it can be loaded to any position in memory by using the following sequence:

1. Choose a suitable address
2. **CLEAR** address
3. **LOAD** — **CODE** address
4. Make sure the first line of program is a REM
5. **RANDOMISE USE** address
6. **ENTER STARTING LINE, STOP SIZE**

Machine code buffs will detect

```

10 REM *****
20 REM **   REM LOADER   **
30 REM *****
40 CLEAR 0000
50 FOR I=00000 TO 40000 STEP 5
70 PRINT I;
80 LET CHECK=0
90 INPUT "Enter bytes?": LINE
100 IF LEN LINE<1 THEN PRINT "
Error in length": GOTO 50
110
120 PRINT TAB 100;

```

A programming utility
from Kenneth Baker

line

Instantly that the same address has been used for all three commands this is particularly in this case because the line number in the program has the same value as the RAMTOP marker. Of A.M. RAMTOP can of course be set at any address you desire, provided that it is somewhere below the concerned

Display file

It was mentioned earlier that this program could be entered anywhere — it can also be stored in the Display file for address 10000 in cases where renumbering is an afterthought or where you only intend to use the program once. You will not need to use the CLEAR command with this method, but you must use GOTO first to try

```

GOTO 10000
LOAD = CODE 10000
RANDOMISE USE 10000

```

and the program will execute in the normal way. You will have to select the program each time you use this method, as the program will not directly you enter a command, but the method does have many advantages — principally that because CLEAR is not used, all of the variables are left intact, and you are not left with the unedited renumber program in the computer once it has performed its brief task.

If you do wish with to remove the code from the computer the following command is suggested:

```

FOR I=address TO
address+1540: POKE I,0: NEXT I

```

```

K70)
270 IF BIST/1000 < 0 THEN PRINT
275 "Error in string = input again"
280 BEEP :Z=12: GO TO 20
285 NEXT Z
290 PRINT FLASH : "READY TAPE"
FOR SAVING"
295 SAVE "reducer"CODE $$$$.L
140
300 CLS
305 PRINT FLASH : "READY TAPE"
TO VERIFY"
310 VERIFY "reducer"CODE $$$$.
140

```

SPECTRUM PROGRAMING

[illegible]

technical graphics

Part 3: Toni Baker goes wandering into some 3D landscapes.

We return to the final part of the series. Take a look at the screen dump included in this article — this is just one possible output of the program listed here. As you can see, it has a number of interesting properties. Firstly it is a three-dimensional graph. Specifically it is a graph of the function

$$Z = \sin(X) \sin(Y)$$

Where $X = 32800 \cdot Y + Y^2$

The function, as well as the boundaries and scales of the graph, can be easily changed: the program is therefore capable of handling almost any three-dimensional graph. There are of course restrictions on what is possible: the function must be of the form $Z = f(X, Y)$ — that is to say there must exist an algorithm to convert given values of X and Y to a single value of Z .

But notice something else about the graph: hidden deep, easily is hidden. Anything which falls behind a peak or ridge is hidden from view, and is not shown on the screen. Of course — this isn't a true hidden line algorithm: it cannot draw solid figures — it is restricted to drawing one very special kind of three-dimensional landscape, a graph of a function of X and Y .

The last subroutine in the machine code listing (the last labelled DEMO, JM of address 829A) is not actually part of the main program of all: it merely the algorithm for calculating Z given X and Y for the function $Z = \sin(X) \sin(Y)$ with $X = 32800 \cdot Y + Y^2$. You can replace, or supplement, this subroutine with similar functions of your own, provided you follow the SUBS, which I shall now explain.

```
10 RANDOMIZE FN B(-10,10,10,10
,1,1,40,3348C)
20 STOP
30 DEF FN B(X,X,X,X,X,X,X,X)=0
BR 33300
```

Figure 1

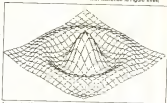
Rule one — the subroutine must ensure that on entry the calculator mode is empty containing no entries whatsoever.

Rule two — you are allowed to make use of calculator memories M0 to M9, but memories M0 upwards are strictly out of bounds, and must not be corrupted.

Rule three — sixteen memories together are used by the main program, numbered M0 to M9. You will need to read at least two of them — memory M0 contains the variable X , and memory M1 contains the variable Y . Your algorithm must read these memories whenever X and Y are required.

Such a subroutine may be placed at any address, and likewise it is possible to contain a series of such functions in memory simultaneously.

Once such a function subroutine exists in memory it is then easy to draw the graph of it. Figure One shows the BASIC program required to draw the screen image shown. More generally Figure Two shows the BASIC line required to draw any graph: it is a use of the BASIC function FN B, requiring eight parameters. The last of these parameters must be the address of the machine code subroutine for calculating the Z values. Figure Two explains the purpose of each of the eight parameters, with reference to Figure Three.



An example of the 3D landscapes that the program generates.

Figure 2

```
DEF FN S(X,X1,X2,X3,X4,X5) = 100/33300
```

```
RANDOMIZE FN S(X1,Y1,X2,Y2,X3,Y3,X4,Y4,X5,Y5,Factor)
```

Factor = address of machine code subroutine to calculate S values of function.

Scale = factor by which S coordinates are to be multiplied.

See Figure Three.

Listing

```

10 DEF FN S(X,X1,X2,X3,X4,X5) = 100/33300
20 RANDOMIZE FN S(X1,Y1,X2,Y2,X3,Y3,X4,Y4,X5,Y5,Factor)
30
40
50
60
70
80
90
100
110
120
130
140
150
160
170
180
190
200
210
220
230
240
250
260
270
280
290
300
310
320
330
340
350
360
370
380
390
400
410
420
430
440
450
460
470
480
490
500
510
520
530
540
550
560
570
580
590
600
610
620
630
640
650
660
670
680
690
700
710
720
730
740
750
760
770
780
790
800
810
820
830
840
850
860
870
880
890
900
910
920
930
940
950
960
970
980
990

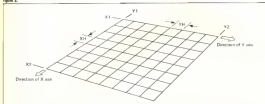
```

12 Computing Models June 2007



10	screen	140	range	100	line to be drawn is on
20	0:000	150	range	100	0
30	0:000	160	range	100	0
40	0:000	170	range	100	0
50	0:000	180	range	100	0
60	0:000	190	range	100	0
70	0:000	200	range	100	0
80	0:000	210	range	100	0
90	0:000	220	range	100	0
100	0:000	230	range	100	0
110	0:000	240	range	100	0
120	0:000	250	range	100	0
130	0:000	260	range	100	0
140	0:000	270	range	100	0
150	0:000	280	range	100	0
160	0:000	290	range	100	0
170	0:000	300	range	100	0
180	0:000	310	range	100	0
190	0:000	320	range	100	0
200	0:000	330	range	100	0
210	0:000	340	range	100	0
220	0:000	350	range	100	0
230	0:000	360	range	100	0
240	0:000	370	range	100	0
250	0:000	380	range	100	0
260	0:000	390	range	100	0
270	0:000	400	range	100	0
280	0:000	410	range	100	0
290	0:000	420	range	100	0
300	0:000	430	range	100	0
310	0:000	440	range	100	0
320	0:000	450	range	100	0
330	0:000	460	range	100	0
340	0:000	470	range	100	0
350	0:000	480	range	100	0
360	0:000	490	range	100	0
370	0:000	500	range	100	0
380	0:000	510	range	100	0
390	0:000	520	range	100	0
400	0:000	530	range	100	0
410	0:000	540	range	100	0
420	0:000	550	range	100	0
430	0:000	560	range	100	0
440	0:000	570	range	100	0
450	0:000	580	range	100	0
460	0:000	590	range	100	0
470	0:000	600	range	100	0
480	0:000	610	range	100	0
490	0:000	620	range	100	0
500	0:000	630	range	100	0
510	0:000	640	range	100	0
520	0:000	650	range	100	0
530	0:000	660	range	100	0
540	0:000	670	range	100	0
550	0:000	680	range	100	0
560	0:000	690	range	100	0
570	0:000	700	range	100	0
580	0:000	710	range	100	0
590	0:000	720	range	100	0
600	0:000	730	range	100	0
610	0:000	740	range	100	0
620	0:000	750	range	100	0
630	0:000	760	range	100	0
640	0:000	770	range	100	0
650	0:000	780	range	100	0
660	0:000	790	range	100	0
670	0:000	800	range	100	0
680	0:000	810	range	100	0
690	0:000	820	range	100	0
700	0:000	830	range	100	0
710	0:000	840	range	100	0
720	0:000	850	range	100	0
730	0:000	860	range	100	0
740	0:000	870	range	100	0
750	0:000	880	range	100	0
760	0:000	890	range	100	0
770	0:000	900	range	100	0
780	0:000	910	range	100	0
790	0:000	920	range	100	0
800	0:000	930	range	100	0
810	0:000	940	range	100	0
820	0:000	950	range	100	0
830	0:000	960	range	100	0
840	0:000	970	range	100	0
850	0:000	980	range	100	0
860	0:000	990	range	100	0
870	0:000	1000	range	100	0
880	0:000	1010	range	100	0
890	0:000	1020	range	100	0
900	0:000	1030	range	100	0
910	0:000	1040	range	100	0
920	0:000	1050	range	100	0
930	0:000	1060	range	100	0
940	0:000	1070	range	100	0
950	0:000	1080	range	100	0
960	0:000	1090	range	100	0
970	0:000	1100	range	100	0
980	0:000	1110	range	100	0
990	0:000	1120	range	100	0
1000	0:000	1130	range	100	0

Figure 1.



which explains all of them diagrammatically.

Figure Three in text shows a case grid, without a function imposed on it. (You could argue of course that it shows a graph of the function $Z = 0$, but that would be trivial.) That grid is converted into a 3D graph (such as the one whose screen dump is shown) by elevating (or perhaps lowering) each point on the grid by the Z value calculated for that point. The grid will therefore "pop" into a three dimensional graph.

Although the rules for the Z axis must be included by hand as one of the parameters for TH of the scales for the X and Y axes are worked out automatically by the program the scales are chosen so that the graph fits exactly across the

full width of the screen ensuring maximum possible resolution.

All you actually have to worry about is the range of values allowed by X , and the range of values allowed by Y . In this program the X values range from $X1$ to $X2$, in steps of SH , while the Y values range from $Y1$ to $Y2$ in steps of TH , so you don't necessarily have to start at the origin (although of course you can if you want to).

Algorithms

Let's have a look at the main program itself. Basically how its algorithm works. The first thing to note is that the graph is drawn in two passes. In pass one the lines parallel to the X axis are drawn, and in pass two the lines

parallel to the Y axis are drawn. In each case the lines (or to be more precise, curves) are drawn from front to back. It is because the curves are drawn in this order that the elimination of hidden detail is possible.

If any part of a line falls behind already existing detail, then the hidden part must not be drawn. By waiting from front to back this decision is easy to make. If a line is intended to be drawn above all other parallel lines, then the line will be visible, and must be drawn. On the other hand, if a line is intended to be drawn below any of the parallel lines, then it is hidden (and must not be drawn).

In practice, however, whole lines are not always either completely above, or completely below, other detail. In practice

technical graphics

only part of a line must be drawn, this means that the DRAW algorithm in the ROM is no good to us, and we must write one of our own. The decision process of whether an individual dot comprising part of a line is to be plotted or not is still, however, tremendously easy.

What we do is to construct a table of address 800H of 400n bytes, each entry in the table corresponding to a pixel-column from the screen. Whenever we plot or attempt to plot a pixel in the screen, then we must examine the table entry corresponding to the column number of the point. The table contains the height. In pixels, of the uppermost plotted point on the screen in the given column. If the point to be plotted lies above the point then it may be safely plotted and the table entry updated. If the point to be plotted lies below this point then it is hidden, and should not be plotted. If we now take a look at the machine code that we shall use this algorithm, in detail I shall explain the workings of each subroutine as we come to it.

Routines

The first subroutine is called `G_PLOT` (address #100). This is the subroutine which attempts to PLOT a point (whose screen

coordinates are held in memories M4 and M5). It compares the coordinates given with the column table, in the manner described above, and PLOTS the point only if it is not hidden.

The second subroutine is `G_DRAW` (address #104). This is my own version of the DRAW algorithm. It will draw a line beginning at the point whose screen coordinates are held in memories M4 and M5, and ending at the point whose screen coordinates are at the top of the calculator stack, in the order M7 instead of actually plotting every point on the line. However, it will update each point to the `G_PLOT` algorithm above to decide whether or not it is hidden.

Next we have a nice easy subroutine `G_CLEAR` (address #108). All this does is to empty the column table ready for each pass.

The next subroutine is called `G_CONVERT` (address #110). Its purpose is to convert three-dimensional graph coordinates to two-dimensional screen coordinates. It takes the graph coordinates X and Y from memories M6 and M7 and immediately calculates the graph coordinate Z using the subroutine provided by the user, and will then proceed to calculate the screen

coordinates `plotx` and `ploty` from the graph coordinates X, Y and Z, using an arbitrary projection algorithm similar to that in Part Two of this series. On exit, the screen coordinates `plotx`, `ploty` are left on the calculator stack, in that order.

Then we have the subroutine `G_PLOT_3D` (address #104) which is used to plot the endpoints of the lines. On entry the graph coordinates X? must be stored in memories M6 and M7. The subroutine will convert these to screen coordinates before attempting to plot the point.

`G_DRAW_3D` (address #100) is very similar. It takes the graph coordinates X and Y from memories M6 and M7, and converts them to screen coordinates, and then proceeds to draw a line from screen coordinates M6 and M7 to the point extrapolated ending any points which need to be hidden. The subroutine also tests whether or not `ENDAC` is pressed, as that you can break out the program (before it's finished by jumping `BRAC` or `CANF-GETSWAC`).

The subroutine `G_GRID` (address #100) is the main algorithm for drawing the grid. It is in three parts. The first part is concerned with initialization. It declares the screen memories required by the program and calculates memories M6 to M7. The values for M6 to M7 are taken from the calculator stack, and are the last three coordinates supplied by `PR` (i.e. the value for M5 has to be calculated), and this is done here. The final two parts are the first and second pass of grid drawing. On every pass an outer loop varies Y (or X) for each curve, while an inner loop varies X (or Y) for different point along that curve. Finally the calculator memories are returned to normal. Note that on entry it must contain the function subroutine address.

The third subroutine in the program is called `PR_GRID` (address #200), and it is the subroutine which handles the parameter (supplied by `PR`) G and the calculator mode (used on the last one into M4) before jumping into `G_DRAW` to draw the actual graph.

The subroutine `GRID_IN` (address #204) is not a part of the main program. It is, rather, an example program, intended to be replaced by your own efforts, or described earlier.

This program concludes the TECHNICAL GRAPHICS series. We shall give graphics a break, for a while, for graphics enthusiasts have to wait a new series entitled `3D-ANALYSIS` is planned for the not too-distant future. Until then, there'll be some surprises. Good programming everyone.

Figure Four

These are the calculator memories used by the program. Note that the first six of these variables are multipurpose, so that different variables can stored there at different times.

Any line segment drawn by the program is defined by being either 'horizontal-ish' or 'vertical-ish' depending on its slope. For 'horizontal-ish' lines, the `POINTX` coordinate is always the X coordinate, and the `POINTY` coordinate is always the Y coordinate. For 'vertical-ish' lines the reverse is true, with the `POINTX` coordinate being the Y coordinate, and the `POINTY` coordinate being the X coordinate.

M0	drawn	DRAW displacement in X direction
M1	drawy	DRAW displacement in Y direction
M2	drawx	DRAW displacement in horizontal direction
M3	drawy	The table <code>drawx/drawy</code>
M4	line	X coordinate of next point on line
M5	line	Forward coordinate of next point on line
M6	line	X coordinate of next point on line
M7	line	Forward coordinate of next point on line
M8	plotx	X coordinate of point to PLOT
M9	ploty	Forward coordinate of point to PLOT
M10	plotx	Forward coordinate of point to PLOT
M11	ploty	Forward coordinate of point to PLOT
M12	xi	Lower bound for X (see Figure 2)
M13	xi	Lower bound for Y (see Figure 2)
M14	xi	Upper bound for X (see Figure 3)
M15	xi	Upper bound for Y (see Figure 3)
M16	xi	Step size dividing X (see Figure 2)
M17	xi	Step size dividing Y (see Figure 2)
M18	xi	Amount by which Z-coords must be multiplied
M19	xi	Amount by which X and Y coordinates must be multiplied
M20	xi	General X coordinate (R0)
M21	xi	General Y coordinate (R1)

Let'shead+Dumpy **Brookway Software** **\$1.50/CP.OO**

Recently, Good Housekeeping gave some beautiful examples of letter dumps, posters and notes, with hints on how to produce them (24 Feb 83). One of the difficulties of this sort of thing is that you need a good collection of design routines to fit your particular needs. Combination 4. As there are lots of interfaces and printers, you can be left with quite a problem to get your ready-made printed sometimes suitable. It's true, but only if you are prepared to spend a lot of time.

Brookway Software have been producing programs that are, firstly, just the ticket. Two programs, in particular, sort out virtually all these sorts of problems, enabling you to produce a variety of machine-readable dumps in an enormous range of printer/interface combinations.

Let'shead+Dumpy is designed primarily to allow you to send unformatted labels or small notes. A special feature lets you produce labels, these is a new if necessary and a variant of this gives you ready-optimized cassette labels.

Cassette microdrive or Qbus also versions of the programs are available for an additional \$1.50. Typical of the care taken is level of combinations in the inclusion of a Wordwise version on side B of the cassette: this is because the Wordwise users much more of the Spectrum RAM for its system variables, buffers and discovers than a microdrive, and it was therefore impossible to include the Wordwise printer interface version within the main program.

The program has codes with a choice of five assorted print styles already available, and a further twenty replacements which can be selected from a menu and loaded: as you can load in your own favourite fonts or use a font-writing/designing program (also supplied) to create your own designs, the program enables you to print to any part of the screen in one of these fonts, put a box round it, insert an arrow round any part of the screen in any direction (green) if you want to centre things, insert an arrow on area and put in a grid to help positioning. The off-subroutines are pretty substantial (plot, draw, etc, delete and load fill), this is not a substitute for a graphics program, nor was it intended to be so. Any ROMable font such a program can, however be loaded, edited and used as part of the display: a scratch pad/memory is provided to help

you with this and allow you to merge over ROMable with another.

A strength and a weakness of the program is that a large amount is in BASIC, which means that it is very user friendly (although system users have not been triggered), readily adapted, and most important of all, will be restored on crashing without losing the picture. It also means that cursor movement, erasing and getting to the screen are all fairly slow, and one would not want to use this program for printing out more pages of busy text.

The best feature is the way the program will deal with a huge range of printer/interface combinations. In addition there are hints and tips for dealing with 'real quality laser/word' printers including the brother and Centronics GUP and even the 'not nearly 16pin standard' Jetris Casio Laser 40, for which a special machine code patch is included. All in all, I found this to be a utility program which has had a great deal of thought put into it, is very well documented and, within its limitations, does what it was intended to do with a minimum of user hassle, and I would recommend it.

Unlike Let'shead+Dumpy does not have built-in type fonts and design capabilities, but it does let you assemble fonts in compa-

ctive libraries of short routines a range of word-processor oriented code machine code screen dump programs, which will tell you printer/interface combinations. Dumpex allows you to choose between format or wordwrap dumps, plain or shaded dumps, and there is a whole range of widths, heights and panel densities to choose from. In addition, you can specify an area of screen and use an auto scroll to skip over blocks from the start or finish of the picture.

If you have done something like this before, to locate the machine code in the middle of the BASIC program area, a large black warning appears on screen, but you are still allowed to pass over. In other words it is designed both for the novice and those who find it with caution. Finally you get the option of saving the code as a text file, for which a coloured screen is loaded.

Again the program has been carefully put together with a comprehensive 12-page handbook neatly printed in jet good power and including a helpfile for those in need.

Both programs are available from Brookway Software on cassette at \$1.50 for Dumps 3 and \$2.00 for Let'shead+ makes one worth buying or Qbus disc at \$1.50 extra, including post and packing.

UTILITIES = current status

Halderbay Kempston Tasman Morex cent. Qbus \$120 R5202	Interface 1 Kempston II ZX I/Print III di'ronics Passage Indescomp
Carriage return code CR Linefeed code LF Program Linefeeds ON ION to print >1 label wide	
Save microdrive back-up copy &P/ Lash/E th	
Design Print Loading Label Interfaces	Save Set Adv Reload Press Adv File Fonts Reload DO
to quit	



STREAMS AND CHANNELS

This month, Toni Baker opens up a new channel to the 128's RAMdisc facility.

This is the final part in the Streams and Channels series, and this is an article for people who own either a Spectrum 128 or a Spectrum 128+2. This month's new channel introduces **SRAM**, **FRS** to the 128's machines.

People who have Interdata One's and Micrograves will already be used to serial files on microcassette cartridge. In this case you open a microcassette file either as a **READ** file (if the file already exists on microcassette) or a **WRITE** file (in which case the file must be created on microcassette). You can print text to a **WRITE** file and then **CLOSE** and **OPEN** it so that it becomes a **READ** file and you may then input your text as either strings or numbers into a **BASIC** variable.

RAMdisc

RAMdisc files work in exactly the same way except that you don't need an Interdata One. When you first **OPEN** a **RAMdisc** file for writing, a file with a given name is created on the Spectrum's so-called 'silicon disc'. Text or numbers may then be printed into this file. Once the channel is closed, no more printing to the file is possible. The file may, however, be re-**OPENED** as a read file, in which case whatever is in the file may be input into a **BASIC** variable.

As with microcassette files, **RAMdisc** files must be **CLOSED** even if the data has been printed to, or input from, the file. If a **WRITE** file is not closed then some or all of the data may be lost, as it will not be cleared from a special buffer. If a **READ** file is not closed then the consequences are less serious, however, since **RAMdisc** channel requests more than 100, which may only be explained by closing the file. You should always close such a file once you have finished with it.

Once a file has been opened, it will appear in the **RAMdisc** catalogue, which you can verify by typing **CAT** in **BASIC**. It is impossible to **UNDO** a **RAMdisc** serial file using a **LOAD** command, however, if it is possible to **ERASE** a **RAMdisc** serial file in the usual manner by entering **ERASE** 'filename'. You should never **ERASE** a **RAMdisc** serial file which is still in use (ie, which still has a stream attached to it); the machine code program does in fact protect itself from this, so naturally so such an error would not be fatal, however — you will certainly get spurious results if you break this rule.

Silicon disc

The key to how the machine code program works is the reorganisation of the memory organisation known as the silicon disc, or **RAMdisc**, which is normally used to store programs, data or machine code for as long as the machine is switched on. **RAMdisc** files are much faster than microcassette files but the whole of **RAMdisc** is closed when the machine is switched off. **RAMdisc** serial files will, of course, suffer from precisely these advantages and disadvantages.

The **RAMdisc** memory itself is primarily organised by the **CARDROM**, which is an index to all files stored in **RAMdisc**. The **CARDROM** resides in **RAM** page seven. It is effectively a stack, which begins at address 7680 and grows downwards, with each entry taking twenty bytes. Figure 2nd shows the meanings of these twenty bytes, with X pointing to the first of these bytes. At the end of the catalogue stack is a twenty byte 'level of catalogue' marker, only three bytes of which are used. The system variable **(RAMC03)** points to this marker and is effectively the stack pointer for this catalogue stack.

So long as we keep the catalogue in its required format we can manipulate the **RAMdisc** organisation itself from machine code. This is the aim of this issue's program.

The **RAMdisc** files themselves begin in **RAM** page 1 and grow upwards through **RAM** pages 3, 4 and 7 (page 5 is taken to ensure that **RAMdisc** files do not collide with the catalogue stack) to avoid any problems with this strange page-numbering. All addresses in **RAMdisc** are 'page-coded'. This means that one register will hold a page-code, while another register will hold an actual physical address. These page codes are 0, 1, 2, 3 and 4, sequentially for **RAMdisc** files, with page-code 5 being the conventional notation for normal (NR) **RAM**.

Channel 'R'

We shall call our new channel 'R', which stands for **RAMdisc** serialised. This is not to be confused with the **SRAM** 'internal' 'R' channel, which is used for inserting bytes into normal (NR) dynamic **RAM**.

R_CHANNELS: DISPROPORTIONATE SPACE

IX + 00:	R_OUT	Address of RAMdisc file output routine (=802B).
IX + 02:	R_IN	Address of RAMdisc file input routine (=8024).
IX + 04:	R_CHANF	Name of channel (= "C").
IX + 05:	R_IDEN	New channel identifier (= 12345).
IX + 07:	R_CLOSE	Address of RAMdisc file close routine (= 8060).
IX + 09:	R_LEN	Length of channel information block (= 020E).
IX + 0B:	R_BUFFER	Pointer into buffer.
IX + 0D:	R_RECORD	Record number within file.
IX + 0F:	R_FILENAME	Filename.
IX + 10:	R_FLAGS	Various flags, defined as follows:
Bits 7 to 0:		Not used.
Bit 1:		Set if End-Of-File at end of record, reset otherwise.
Bit 0:		Reset for a READ file; set for a WRITE file.
IX + 1B:	R_RECORDS	Length of record within buffer.
IX + 1D:	R_BUFFER	Buffer storing current record.

Figure 1

channel-channel?" is quite interesting in fact it is permanently attached to stream number one, so to use it from machine code all you have to do is select stream number one as the current stream by loading A with FF and calling address 1000. It is impossible to use from BASIC. Prior to selecting stream number one the system variable R_CUR must be made to point somewhere into dynamic RAM. Pointing to stream 0 will then insert characters into dynamic memory at the point indicated by R_CUR.

Our new channel is also called "C" but in use it might more correctly — and of course it may be used from BASIC.

We require a channel information block over 10K in size. Most of this is in fact a huge 4000 byte buffer for reasons of speed, the buffer is used most of the time with RAMdisc itself only being accessed when the end of the buffer is reached. It is of course important to realise that

the location of a RAMdisc file is not constant — it may move either if another file is erased, or if more bytes are inserted into another RAMdisc serial file. This means that the file has to be re-located every time we wish to read or write into it. To save time we make use of a temporary buffer virtually all of the time.

Figure One shows the actual organisation of the channel information block to use it. channel. Note that R_RECORD, R_RECORDS, and for one of R_FLAGS are used only if the file is a READ file. The remaining variables are used for both types of file.

To interface with BASIC, an example machine code program is appended to the end. Essentially a RAMdisc serial file may be opened by loading the A register with the stream number of the stream to be opened, whilst the default filename is stored in the system variable M_PATH of address 5047, and then calling the label

R_OPEN. Four additional entry points are included, labelled OPEN_A, OPEN_B, CLOSE_A and CLOSE_B. Calling OPEN_A will open a RAMdisc serial file called "FILE" and attach it to stream four. Similarly calling OPEN_B will open a RAMdisc serial file called "FILE2" and attach it to stream five, the routines CLOSE_A and CLOSE_B will of course close these new channels.

This means that the new channel may easily be used in BASIC. Take a look at figure three. It contains a BASIC program which demonstrates the RAMdisc serial files at work, first on BASIC files, then on READ files. If you may be surprised at how fast it all works.

Of course you won't always want your files to be called "FILE" or "FILE2" and you won't always want to use streams four or five. That is why the more general entry point R_OPEN is included, which, as has already been noted, requires that A

CATALOGUE: USER INFORMATION

IX + 00:	FF_NAME	Filename.
IX + 02:	FF_START	Page-coded address of start of file.
IX + 04:	FF_LEN	Total number of bytes in file, including header info.
IX + 10:	FF_END	Page-coded address of byte beyond end of file.
IX + 12:	FF_FLAG	Reset unless catalogue information incomplete, (ie reset normally).

Figure 2

STREAMS AND CHANNELS

contains the stream number and the 3000 contains the channel (with trailing spaces if required). You will have to write your own machine code to join any other combination into BASIC, utilizing this routine.

The potential for streams and channels is limitless. It is theoretically possible, for instance, to have a channel which utilizes a RAMDISC ACCESS FILE, or a RAMDISC ACCESS FILE — though the program would have to be much more complicated. I have shown you enough of the potential use for streams and channels to whet your appetite a little, and I hope I shall leave you, if demand is high enough, I stay around with more. Good programming everyone! QED may the force be with you.

```
1000 REM WRITE FILE INFO
1010 RANDOMIZE USR 47713: REM OPEN #4,"B_FILE"
1020 RANDOMIZE USR 47720: REM OPEN #5,"B_FILE"
1030 FOR I = 1 TO 512
1040 INPUT #4,IN1
1050 PRINT #4,IN1
1060 PRINT #5,IN1
1070 NEXT I
1080 RANDOMIZE USR 47736: REM CLOSE #4
1090 RANDOMIZE USR 47740: REM CLOSE #5
1100 REM READ FILE INFO
1110 RANDOMIZE USR 47713: REM OPEN #4,"B_FILE"
1120 RANDOMIZE USR 47720: REM OPEN #5,"B_FILE"
1130 FOR I = 1 TO 512
1140 INPUT #4,IN
1150 INPUT #5,IN
1160 PRINT #4,IN
1170 NEXT I
1180 RANDOMIZE USR 47736: REM CLOSE #4
1190 RANDOMIZE USR 47740: REM CLOSE #5
1200 STOP
```

Figure 1

NOTES

In `CPRM_NBR` at address 304D (Streams and Channels Part Two, January Issue, page 66) there is an instruction missing. The instruction is `POP BC` (hex code C7). It should be the eleventh instruction of the routine, occurring between `OR C` and `JR 5,CPRM_NBR,2`. Sorry about that.

Figure 4

This is a listing of the "BASIC" code for the program. It is a high-level language code but contains the low-level code for the hardware. The code is written in a high-level language but contains the low-level code for the hardware. The code is written in a high-level language but contains the low-level code for the hardware.

```
1000 REM WRITE FILE INFO
1010 RANDOMIZE USR 47713: REM OPEN #4,"B_FILE"
1020 RANDOMIZE USR 47720: REM OPEN #5,"B_FILE"
1030 FOR I = 1 TO 512
1040 INPUT #4,IN1
1050 PRINT #4,IN1
1060 PRINT #5,IN1
1070 NEXT I
1080 RANDOMIZE USR 47736: REM CLOSE #4
1090 RANDOMIZE USR 47740: REM CLOSE #5
1100 REM READ FILE INFO
1110 RANDOMIZE USR 47713: REM OPEN #4,"B_FILE"
1120 RANDOMIZE USR 47720: REM OPEN #5,"B_FILE"
1130 FOR I = 1 TO 512
1140 INPUT #4,IN
1150 INPUT #5,IN
1160 PRINT #4,IN
1170 NEXT I
1180 RANDOMIZE USR 47736: REM CLOSE #4
1190 RANDOMIZE USR 47740: REM CLOSE #5
1200 STOP
```

The following instructions are in the code. It is a high-level language code but contains the low-level code for the hardware. The code is written in a high-level language but contains the low-level code for the hardware.

```
1000 REM WRITE FILE INFO
1010 RANDOMIZE USR 47713: REM OPEN #4,"B_FILE"
1020 RANDOMIZE USR 47720: REM OPEN #5,"B_FILE"
1030 FOR I = 1 TO 512
1040 INPUT #4,IN1
1050 PRINT #4,IN1
1060 PRINT #5,IN1
1070 NEXT I
1080 RANDOMIZE USR 47736: REM CLOSE #4
1090 RANDOMIZE USR 47740: REM CLOSE #5
1100 REM READ FILE INFO
1110 RANDOMIZE USR 47713: REM OPEN #4,"B_FILE"
1120 RANDOMIZE USR 47720: REM OPEN #5,"B_FILE"
1130 FOR I = 1 TO 512
1140 INPUT #4,IN
1150 INPUT #5,IN
1160 PRINT #4,IN
1170 NEXT I
1180 RANDOMIZE USR 47736: REM CLOSE #4
1190 RANDOMIZE USR 47740: REM CLOSE #5
1200 STOP
```

The following instructions are in the code. It is a high-level language code but contains the low-level code for the hardware. The code is written in a high-level language but contains the low-level code for the hardware.

The following instructions are in the code. It is a high-level language code but contains the low-level code for the hardware. The code is written in a high-level language but contains the low-level code for the hardware.

The following instructions are in the code. It is a high-level language code but contains the low-level code for the hardware. The code is written in a high-level language but contains the low-level code for the hardware.

The following instructions are in the code. It is a high-level language code but contains the low-level code for the hardware. The code is written in a high-level language but contains the low-level code for the hardware.

WAR GAMES

Gordon Hamlett reports from the front . . .

The number of wargames available for the home computer has expanded considerably over the last couple of years. There are several reasons why the wider section of a game is succeeding whereas its traditional "board game" counterpart had only limited appeal.

The obvious benefit that a novice offers is that you don't need a second person if you want to play a game. Then, there is no need to set up the boards or work out complex results from a set of complicated tables; the strategy is all handled for you. On the debit side, computer wargames are not yet as sophisticated as their original counterparts although the balance is shifting and, as I will point out later, there are certain advantages in playability to be gained on the micro.

Three games have appeared for the Spectrum in recent weeks and all three offer the budding strategist something different. The first of these games is *Samurai* from GSI (£9.95). This is a one player game with three different scenarios and three difficulty levels.

You take the role of a small band of warriors, trying to wipe out an enemy force of assorted temple guards. The game is turn driven and starts with you selecting your initial forces. You move to three points to "spend" and each type of warrior costs a varying amount. There are four types to choose from — Ashigaru (all lightly armoured troops, the traditional Japanese warrior), the Samurai, a mounted Samurai and the deadly Ninja.

The mechanics of the game are very simple and are basically move and fight. Ninja must be carefully managed as



they are the only troops who can attack from a distance and should be used very much in hit-and-run tactics.

Samurai is by far the simplest of the three games and will

appeal more to the novice. However, thought to all intents and purposes, *Samurai* is the same game as *Swords of Samurai* (GSI) and it is probably not worth getting both.

Keith de Borch





Moving forward is a few hundred yards brings us to **Heat** and **PS3's Battle of Britain** (3-95) which had concentrated lightning in its head of the Luftwaffe, to destroy the RAF prior to a planned invasion of Britain. Due to the RAF's meagre resources, just about any tactic would have worked provided the Germans had indulged in 38 profiles. Goring decided that his methods weren't working and so stretched his forces to night bombing major cities. However, through the toll was, there isn't any day I was ever going to destroy the British. Thanks for the simple reason that the Spitfires and Hurricanes never flew at night.

There are three main scenarios for you to try the training game gets you used to commanding the forces of your disposal by simulating a light raid in Britain, the Germans there everything they have or you can only for a period of one day. Finally, there is the much larger campaign which is played over thirty days. There are also optional update scenarios in which you can try something along the lines of the constant of your own main room. If you choose this option, how well you do in your own personal combat directly affects the outcome of a particular battle. This is fine for arcade fans but those of you only interested in the strategy elements should leave well alone.

The gameplay depends on you making a lot of very fast decisions. As the German forces start to appear, you must scramble squadrons to intercept them. After combat or a prolonged period, a squadron must be landed in order that it can refuel and rearm. Failure to monitor the status of your squadrons will result in them becoming depleted and is available to you for a period of time. Airfield planning because of the weather add to your problems.

Again, a fairly simple game to play but for one who has never tried to juggle will tell you it is very soon to throw a lot of balls up into the air at once. It is a different matter to keep them there.

The final game this month is **Wolcen** (3-95) from DOS. This simulates the Russian campaign of World War II by far the most detailed of the three games on offer. Control however remains straightforward and is as simple as driving.

There are five different scenarios designed to last anywhere from half an hour to sixteen hours. You can choose to play either the Axis or the Allied forces and the game can be played against either a computer or human opponents. The DOS version offers several other options including a

declining mode, several different "weather" variations and no need to reload any data after every game (necessary on the 486 spectrum due to memory limitations).

One feature not available on board war games for obvious reasons is that of hidden movement by the enemy. Enemy units are only discovered directly when you bump into them. There are several ways of moving your troops. Normal movement around and level. Assault means that a unit will go all out to gain its objective whereas retreat is very deliberate — you can move while doing so, but double speed. You can also choose to hold a position or fortify it.

Terrain plays an important part in the game, not only does it affect the rate of movement, but also how well a unit can attack from or defend a position. Control is depicted by many factors including the strength of a unit, how well it is supplied and the weather. Supplying your units is a vital element in your strategy and you should also make good use of any air power that you have, either to deliver an air strike or to support as you try to discover the enemy's positions.

Wolcen is very well presented with an excellent instruction booklet giving players hints and the historical background to the campaign. Highly recommended to serious strategists.

Wolcen



Wolcen



Battle of Britain



Samurai



Battle of Britain



Wolcen





MINDPLAY



Peter Sweasay having some problems pigeon-holing this month's unusual adventures.

■ Every old month, some of the games for review is entirely conventional. Imagination — another cheapie from prolific author Peter Horne — is four mini-games in one — yet is the closest to conventional.

The Growing Pains Of Action Mike is another Level 9 multi-choice adventure. In the past there have been doubts as to how Mike would fit into the game, comes courtesy of Edward Ransom Virgin Games — Mike's father-in-law, 'producer'. I hope this constitutes as a partnership, while Mike has been responsible for some commercially unusual licensing deals over the past few years, their advertising has been far less subtle for the games to achieve the high sales they deserved.

Third curiosity is The Inheritance from the coast-to-coast software retailers at Infotronics. It is so hard to classify, I almost reviewed it elsewhere in the magazine, but while it's not what you need (no text input, not much to read) nor is it really an arcade game (no finger motion's needed) in the end it

described categorizing games is a bit like a survey — so I reviewed it here.

Setting of Ransom's first, Infotronics game is very interesting, structured lined up for release soon. Murder On The High Seas is a complex adventure, including all elements, including all aspects to question, but what especially appeals to me is the packaging, included with the game will be evidence — such as cartridge cases, book, newspaper cuttings, ink — which you can examine when it's referred to in the game. Such

packaging adds greatly to the atmosphere of an adventure, and is standard with games from Infotronics (the watch making adventure writers — unless only for the (word machine) and quite common among American companies, but Spectrum adventures have virtually never been anything like this. The closest attempt was the original release of Solitaire II, where you were given some documents, a yellow badge and duplicate messages, containing information vital to the game's completion. Runestone has a



Infotronics providing the clues

map and start going, while Soranika's recent adventures included short quests, but these are nowhere near as well-crafted as what Informan normally offers (with their recent release *Leather Goddesses of Frodo*, for example, is scratch and sniff card sets enclosed) for reference at certain points in the game, along with 3D glasses to read a

special comic, left's hope videogames left a mark. Users of the *Graphic Adventures Creator* will be interested to know that Informan has published *The GAO Adventure Writers Handbook* as a supplement to the manual; it resembles a club magazine, but contains useful information, there are clearer explanations of the

programs' more complicated functions, guides on how to use the program's tool and notes on some of its idiosyncrasies.

There's also a short but excellent introduction on how to write a good adventure, covering minor points in the ones I often expand on these very pages. The handbook costs £1.25 with the incentive address.

HELPLINE

During the month with Soranika highly recommended our *Index* **Barrow of Barrow**. Ben Nise Pulpier and Keith White are being loved by the *Kingdoms*. Simply follow your country code. **CHOOSE GALT** behind you. To catch the dog, Keith, you must **PAID** the proffer. Make one take the rock to the side of the temple, and **PUT** **SAGE** INTO **HOLE**. Then make the other enter the temple, **CHOOSE** **DOOR** and attempt to **CHOOSE** **DOOR**. The multi should run through the hole in the wall — straight into the door! As for the wall, **WALK** **LAUNDRY** down it to see the range. Get labor to take the basket there and drop it down the wall. When **SCORE** climbs down he will find it has broken into planks, which can be used to build a bridge across the river.

Box is not the sort of noise you should make when reading the column, but it is the title of a somewhat unplayable *Master* *tronic* (sheep). Miss I. Y. Smith from Clapton cannot board the bus having passed the cocoons, even though she has the bus stop sign. A question of precision vocabulary, I suspect. **DRIFT** **BOX**, **STOP** **BOX** and **DRIFT** **BOX**.

Stephen Lawrence cannot find the magic potion or the key in part three of *Based on the Rings*. **ENTER** **DOOR** to find the same while the farmer gets found in the box. Is from the headless statue (you'll have to call the *Kingdom* first).

Dead End Quest

Neil Taitton is frustrated by *Medieval* House's rather good *Medieval Quest* for it is tedious to find in the deserted beach he wants to stay under water. Head out and **CHOOSE** **DOOR** go down and to you understand you'll find an *adventure* then to prevent confusion, **WALK** **DOOR** and **ALL** **ADVENTURE** progress in this sector will find you the spray point you need in the future you'll also find someone who wants that newspaper you asked

about (though how it doesn't go all soggy is unexplained). The *Key* *combination*, *Indoors* only is **ENTER**. There's no need to unlock the door in the *barrow* *ville*, indeed there's no need to enter it, at least not to complete the game.

Andrew McMaster wishes to pass the television screen in *Seaside* *Beats*. You need to there's a *poncho* of it. How do you make a *poncho*, you say? No! I need the *herb* *egg* *blow* a *chewing* *gun* *button* to *write* the *poetry* and the *bow*, which is on the *doorway* (well it's remember right). *Examine* the *broken* *shell* and *take* the *four* *stone* it, then *open* the *ridge* to find the *multi*, **MADE** **FRAGILE**, **CHOOSE** **FRAGILE**, **USE** **FRAGILE** (now you manage to carry it around without it falling to bits or sticking to your hands is not explained — realism is not a strong point in this game).

Marous Beer wishes to kill *Slime* in his *gremlin* *adventure*, *Gremlin*, having already blown up the store. As the gas exploded, the *note* *card* *one* should have rushed away and *struck* you over. So **POUNDER** **SLIME** and *kill* *lead* you to the swimming pool. You should have drained this earlier, so when *Slime* died in it reproduced he will have *erected* himself out. **GET** the *unconscious* *bird*, **DRIFT** **UP** then *go* *out* *Down* *will* *break* — did you know how *gremlin* *hole* *light* to *rescue* *Ammon* in *Red* *Hulk*, **PUT** **FLUID** **SARS**, **HOLD** **NOSE** and **CHOOSE** **YES**, then **GET** **ANS**, north twice to the underground *room*, go through the *crack* and they will *attack* *Slime*. As for *Wetness* *IT*, if you go to the *Corporation* in the *box*, you won't be able to pass the *guard* *however* where to *reply* *go* in the *box*, it answered by *stealing* the *other* *program* *Marous* *take* about — how to use the *telescope*, you need to **INSERT** **WINDUP** in the *use* *Pen* **TYPE** **TELESCOPE** for your *destination*. The *request* *comes* from the *con* in the *tourism* — it's the *talk* with the *liquid*.

Gremlin Grips

Help for Simon Fuller with the

some *game*, to *steal* the *lands*, just *take* it, then *give* the *butcher* that *small* *bag* *box* from the *room* *side* (you know, the *one* with the *glowing* *star* in it). *Simon* *also* *calls* for *help* with about a *million* *other* *adventures*, so *here* *we* *go* in *Project* *X*, the *information* *is* *control* *escape* the *car* **CLIMB** **SEAT** **SEAT**, **WIND** **RANGE** and *out* *take* the *lighter* *box* in *Dracula* **Get** **Two** (I hope you *re-view* *it*, *Simon* *progress* from the *last* *location* is *stagnant* the *first* **LOOK** **AROUND** **LOOK** **CLUB** **WOMAN** and **LOOK** **IT** **IS**, take what she *offer* you, but *take* it in *The* *Final* *Mission*, *Simon* *wants* to *avoid* the *time*, in *fact* *dealing* with the *green* *stuff* will *solve* *you* *another* *problem* if you *prepare* *properly*, *take* the *star* from the *bottom* *corner* of the *large* *hall* and *use* it to *fill* the *one* *inch* *gap* at the *bottom* of the *door*, *making* *sure* you *have* the *door* from *both* of the *great* *pillared* *hall*, go to the *chamber* with the *rising* *door*, *doors* *beams*, and *head* up, into the *cubby* *hole*. *Wait* until you *hear* it *springing* *sound*, then **DROP** **SCAP**. Not only will the *time* have been *cleared* up, but it will have *crashed* the *door* for you. As for the *native* *worker* in *happiness* *land* (a *game* *written* by *W.H. Smith* for one of its *reporters*), you must *give* her the *books*.

Simon *also* *many* *more* *questions*, but *space* is *running* *out* so *well* *hope* *those* *he* *need* *time*.

Simon *also* *many* *more* *questions*, but *space* is *running* *out* so *well* *hope* *those* *he* *need* *time*.

Peter Swasey for Poel Laureate

If you're stuck in a game And your brain's all torn out Then remember my name, Just give me a shout. Fill in the *despatch* It's as easy as pie. Send it to *Minipage* (well you try thinking of a rhyme for *swasey*) And to solve it will try.

If I think you better stick to *adventure*, *Swasey* — Ed. *Remember* A few rules: I wish *respondents*, please *enclose* a *stamped* *envelope* *envelope* if you want a personal reply rather than wait some months for



the magazine to come out if you are willing to be blamed, just because you are a nerd — I'll add the postage. I try to be fair, but within two months I can take longer for the other hand, you might receive an immediate and I CAN'T DEAL WITH ADVENTURES. Not arcade games, but technical problems (write to Cromwell about these), not arcade adventures (Gunguisle games included), not even Harry on the Mopeds. Please put the name of the game you're writing about on the back of the envelope. And Manual don't write letters asking for general solutions. — I just don't have time to write those out for everyone. (They can be obtained however from good computer clubs.) Finally I don't advise you write to me with least a problem. I have a schedule but I really much better than anything I could offer to any level. I questions come why down my list of relatives.

Free estimates for producers,
retailers, associations and others.



Title _____

Company _____

Problem _____

I can help solve _____

Name _____

Address _____

See publishing rights to *Applications
in Computing*, No 1 (October)
Source: London 1990 348, 350

many other magazines join
your problems like INs AND keep
you a little entertained!

THE GROWING PAINS OF ADRIAN MOLE



What if the first Atlanta game (probably the one that inspired the film) had been programmed by a team of specialists in very strong style, in the same sense used for the Atlanta 1996? Just in case you've been wondering, the positive voices with some help in it to buck the Atlanta movie in 1994 caught the nation's attention. "The Atlanta movie is a serious misreading of what's really going on," says O'Neil. "It's not about the unemployed parents and inner ghettoes of Atlanta. The phenomenon is a successful blend of Jewish and African-American culture. It's about the children of the ghetto, not the ghetto itself." O'Neil says the film is a "very serious" look at the ghetto, not a "documentary" about it.

It should be noted that the results of the first two studies were not replicated in the third study. The reason for this may be that the first two studies were conducted in a laboratory setting, while the third study was conducted in a field setting. This suggests that the results of the first two studies may be specific to the laboratory environment.

[illegible]

national, or you could try the old
the national

I hope I think you'll understand the game if you know nothing about Action Mike (as there's no explanation in the instructions about what the character you'll encounter is). You don't have to let a Mike fanfic to appreciate the game though. I'm seriously not yet a writer.

[illegible]

As I sat behind the music stand with the signet ring from home in front of me, I felt close to home. I was in the same room, the same place, the same city as the first album was recorded. I was sitting in the same room, the same place, the same city as the first album was recorded. I was sitting in the same room, the same place, the same city as the first album was recorded.

[illegible]

As with *Miss Amyberg*, *Goodnight, Bebe* is a well-written, well-plotted story that shows a young woman's growth and development. The story is a good example of the author's skill in writing for young adults.



600



IMAGINATION

100

[illegible]

With a title like *That's How It Goes*, I was setting the bar high for the album. I needed to be going to that full house by selling one new effort. I liked this product and I felt original and creative. I thought — but then I discovered an underlying and unique feature, which made the title of this movie just what I needed.

The game starts with you sitting in front of your computer. You must do whatever the screen-talking wanderer with the lantern has to say (mind you, played here is a harpian box and you presented with a choice of four games. Choosing one of these — a space adventure, war game, flight game, adventure, or a board

[illegible]

By the way, if you're looking for a place to play, the city has several indoor soccer fields. And if you're looking for a place to play, the city has several indoor soccer fields. And if you're looking for a place to play, the city has several indoor soccer fields.

[illegible]

INVENTORY — virtually useless, often, and what you can do is limited — largely only those offers necessary to complete the game involve a response, which I've said many times before is poor optimizing. But not only are you unable to do much the manager which led you to be optimizing and long windows, doing the likes of "let's you can manage that" but you must do it.

The first two weeks of the year are the best time to visit. The weather is perfect, the crowds are small, and the prices are low. The third week is also a good time to visit, but the crowds are a bit larger. The fourth week is the worst time to visit, as the weather is hot and the crowds are huge.

Intelligence is change and but its self — but its true power the family spectrum achieves once you be more sophisticated than this — each of these is a new world.

THE INHERITANCE

1000

things are looking bleak for Las Vegas, where gambling is a major draw. The city is in a better shape than many other Nevada cities, but it has lost its main market. The city is in a better shape than many other Nevada cities, but it has lost its main market. The city is in a better shape than many other Nevada cities, but it has lost its main market.

There is in Las Vegas (Feb. 8) where you will participate in several gambling games (slot machines, craps, etc.) in your second day. See, however, *page 10*.

Control is why it's missing from graphics. Sticking your nose to the box won't put up physical, spirit, doubt, or all three. The game is misleading since all that happens is the graphics are large, colorful, and generally rather good. The situation is worse, however.

Still well known in traditional style, with other ethnic influences (the traditionalists have French shapes, untraditional), their models are considered — impressively — to show the changing style back toward their traditions.

The reference has many generously sized touches, and it is cleverly designed to encourage the collector to read. The Chronos research team says to make sure you've already read an entire issue before you begin. The book is divided into two sections: the first section is a general overview of the industry, and the second section is a more detailed look at the various aspects of the industry.

required by all those good old walking for over 20 minutes. The game system is the radio control for great potential — but still the interface doesn't use it like the



600



PRICES FOR ACCESS AND BROWSE ARE:

All advertisements in this section must be prepaid. Advertisements are accepted subject to the terms and conditions printed on the advertisement card and insertion on request.



Send your requirements to:
NICOLA BATY
ASP Ltd, 1 Golden Square,
London W1

D.D.8.

PROF.
PROFESSOR
PROFESSOR

[illegible][illegible][illegible]

THE COMBINATION OF
the other programs in each
set

The above article contains part of the text of a letter to the editor published in the 1990s. The text is as follows:

THE RECORDING OF THE PAGE
All four programs on this
cassette in a large video box
Retail Price ONLY \$14.95

All our programs are totally self-paced
and can be done in your own home.
Phone phone 1-800-855-8884
Please send money in advance.
P.O. #123-0, 400 Mountain Road,
Barnstable, Mass 02536-0123
\$500 is per student. Call for: 401-234-5678

[illegible]

BRADWAY SOFTWARE 128

ALL INFORMATION CONTAINED HEREIN IS UNCLASSIFIED EXCEPT WHERE SHOWN OTHERWISE
DATE 08-26-2010 BY 60322 UCBAW

Abstract Following the 1991 Los Angeles earthquake, a large number of new and existing structures were damaged and some of them were found to be unsafe. This paper describes the damage to a building and the reasons for the damage. The building was a three-story office building. The damage was caused by the earthquake. The reasons for the damage were the poor construction and the lack of seismic retrofits. The paper also discusses the importance of seismic retrofits for existing structures.

© 1999 The McGraw-Hill Companies. All rights reserved. Printed in the United States of America. This publication is intended to provide accurate and authoritative information in regard to the subject matter covered. It is sold with the understanding that the publisher is not engaged in rendering legal, accounting, or other professional service. If legal advice or other expert assistance is required, the services of a competent professional person should be sought. This publication is not intended to constitute an offer of insurance or any other financial product. It is not to be construed as a contract or any other legal document. It is to be used only as a guide. The McGraw-Hill Companies, Inc. 1221 Avenue of the Americas, New York, NY 10020-1399. ISBN 0-07-057444-4. Printed in the United States of America. 1999. Printed on acid-free paper. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or by any information storage and retrieval system, without prior written permission from the McGraw-Hill Companies, Inc.

© 2000 Blackwell Science Ltd *Journal of Internal Medicine* 247: 399–406

Business Gateway program is an award-winning HR & business solution. Let's change careers together. Visit www.bgateway.com today.

Abstract

Individuals with a history of a seizure disorder should be monitored closely during treatment with this drug.

ALABAMA (Sept 1981)
A Market (Growth) Strategy
Marketing: Ford Ltd
1981-1982
1981-1982

Abstract—The purpose of this study was to determine the effect of a 10-week training program on the heart rate (HR) and energy expenditure (EE) of sedentary, middle-aged women. The subjects were 15 women, 40 to 50 years of age, who were sedentary and had no cardiovascular disease. They were randomly assigned to a 10-week training program or a control group. The training program consisted of three sessions per week of aerobic exercise at 60% of maximum HR. The control group did not exercise. The HR and EE were measured at rest and during exercise at the beginning and end of the 10-week period. The results showed that the training program significantly increased the HR and EE of the subjects. The HR increased from 68 to 72 beats per minute at rest and from 145 to 155 beats per minute during exercise. The EE increased from 1.2 to 1.4 kcal per minute at rest and from 2.5 to 3.0 kcal per minute during exercise. The results suggest that a 10-week training program can improve the cardiovascular fitness of sedentary, middle-aged women.

the 1970s, the 1980s, and the 1990s. The 1970s were a time of great change for the world, and the 1980s were a time of great change for the United States. The 1990s were a time of great change for the world, and the 2000s were a time of great change for the United States.

1. **Business Model:** The company operates as a **Software-as-a-Service (SaaS)** provider, offering a cloud-based platform for **document management and collaboration**.
 2. **Market Segment:** The primary target market is **mid-sized businesses** across various industries, including **retail, healthcare, and financial services**.
 3. **Competitive Advantage:** The company's key differentiator is its **intuitive user interface and robust security features**, which ensure **data integrity and compliance**.
 4. **Growth Strategy:** The company plans to achieve growth through **organic expansion** and **strategic partnerships** with **technology integrators and system integrators**.
 5. **Financial Projections:** The company expects **revenue growth of approximately 25% annually** over the next five years, driven by **increased adoption and upselling of premium features**.

1. The first step is to identify the problem. In this case, the problem is that the system is not working properly.

WANTED: TEACHERS for teaching
Communications, Computers, and
you attend for your assignments.
Inquire: Chatter (800) 455-4555

**CALL 81-437 5000 FOR
DETAILS OF OUR
SPECIAL OFFERS**

Services

**TO ADVERTISE
IN OUR NEWEST
SECTION RING
NICOLA BATY
ON
01-437 0699**

**STOP
PLAYING GAMES**
Don't waste a minute today. You may need
100 "I need more money" signs. 100 and 100 are
guaranteed to look like all the other signs on the
100 money sign market. Sign is the number
one way to get more money for your business.
Write this. Post this. 100
100 George Street
Baltimore, MD 21201

WARNING NOTICE

Advertisements placed in this magazine are to be in strict compliance with the standard conditions printed at which conditions are available on request and on the clear understanding that the advertiser warrants that his advertisement(s) does not infringe any copyright or condition of sale of any interested party in the advertised product.

Further this advertiser indemnifies the proprietors of this magazine in respect of losses, damages or any other claims brought against them as a result of legal action arising from the publication of the advertisement.

Any breach of these terms or the said conditions may result in prosecution of the advertiser by the proprietors.

SOFTWARE

HUMPHREY SOFTWARE TWO PROGRAMS

OPEN ADDRESS The comprehensive file management system for IBM PCs and compatibles. **FILE SORT**

FILE SORT Sorts, lists and sorts your data according to many key fields. Both search and sort. This is a real time in use program. **PRICE** £14.95

OPEN ADDRESS £14.95 **FILE SORT** £14.95

HUMPHREY SOFTWARE c/o Robert Anderson Esq. Worcester Road, Lymington, Hants. SO41 1LL

CLASICO SOFTWARE

CLASICO SOFTWARE c/o Robert Anderson Esq. Worcester Road, Lymington, Hants. SO41 1LL

CLASICO SOFTWARE c/o Robert Anderson Esq. Worcester Road, Lymington, Hants. SO41 1LL

CLASICO SOFTWARE c/o Robert Anderson Esq. Worcester Road, Lymington, Hants. SO41 1LL

OLDSPECTRUM SYSTEMS

is the range of business and personal programs available for IBM PCs and compatibles. **PRICE** £14.95

OLDSPECTRUM SYSTEMS c/o Robert Anderson Esq. Worcester Road, Lymington, Hants. SO41 1LL

WD Software

FOR THE QL

2000 This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

2000 This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

2000 This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

2000 This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

2000 This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

2000 This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

2000 This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

2000 This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

2000 This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

2000 This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

2000 This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

2000 This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

FORTRAN FILE COMPILER for 486 Spectrum for and 274 Speed. This is a comprehensive business management system for the QL. It includes a full range of business management programs. **PRICE** £14.95

CLASSIFIED COUPON

ALL CLASSIFIED ADVERTISEMENTS MUST BE PRE-PAID. There are no reimbursements for cancellations.

I enclose my Cheque/Postal Order for £ to purchase the following advertisement(s) for publication in the following issue(s) of the following publication(s).

PLEASE PRINT MY ADVERTISEMENT NO.

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100

FOR INSERTION COPY DATE

IF YOU DO NOT WISH TO CUT YOUR MAGAZINE, PHOTOCOPY THIS FORM

Name

Address

POST CODE

DAYTIME TEL NO

Signature Date



Tom Cat (The Special Edition)
 CBL
 \$9.95
 1234-5678

[illegible]

GREAT



THE OFFICIAL



FA Cup 87

FOOTBALL



This new guide is essential for anyone who wants to follow the FA Cup 87. It contains all the information you need to know about the competition, from the first round to the final, and includes details of all the clubs and players involved.

It also includes a full list of all the clubs that are competing in the competition, and a list of all the players who are taking part. It is a must-have for any football fan who wants to follow the FA Cup 87.

The guide is written by the author of the Official FA Cup 87, and is the only book of its kind. It contains all the information you need to know about the competition, from the first round to the final, and includes details of all the clubs and players involved.

Available for Spectator, MFC, and Arsenal CPC. The guide is written by the author of the Official FA Cup 87, and is the only book of its kind. It contains all the information you need to know about the competition, from the first round to the final, and includes details of all the clubs and players involved.

Spectator MFC £1.95
 Carrington MFC £1.95
 Arsenal CPC £1.95

Please tick the number you require and send this form off to: Virgin Games Mail Order, Ltd, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265, 266, 267, 268, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362, 363, 364, 365, 366, 367, 368, 369, 370, 371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 414, 415, 416, 417, 418, 419, 420, 421, 422, 423, 424, 425, 426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442, 443, 444, 445, 446, 447, 448, 449, 450, 451, 452, 453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465, 466, 467, 468, 469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 488, 489, 490, 491, 492, 493, 494, 495, 496, 497, 498, 499, 500, 501, 502, 503, 504, 505, 506, 507, 508, 509, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 522, 523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 540, 541, 542, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562, 563, 564, 565, 566, 567, 568, 569, 570, 571, 572, 573, 574, 575, 576, 577, 578, 579, 580, 581, 582, 583, 584, 585, 586, 587, 588, 589, 590, 591, 592, 593, 594, 595, 596, 597, 598, 599, 600, 601, 602, 603, 604, 605, 606, 607, 608, 609, 610, 611, 612, 613, 614, 615, 616, 617, 618, 619, 620, 621, 622, 623, 624, 625, 626, 627, 628, 629, 630, 631, 632, 633, 634, 635, 636, 637, 638, 639, 640, 641, 642, 643, 644, 645, 646, 647, 648, 649, 650, 651, 652, 653, 654, 655, 656, 657, 658, 659, 660, 661, 662, 663, 664, 665, 666, 667, 668, 669, 670, 671, 672, 673, 674, 675, 676, 677, 678, 679, 680, 681, 682, 683, 684, 685, 686, 687, 688, 689, 690, 691, 692, 693, 694, 695, 696, 697, 698, 699, 700, 701, 702, 703, 704, 705, 706, 707, 708, 709, 710, 711, 712, 713, 714, 715, 716, 717, 718, 719, 720, 721, 722, 723, 724, 725, 726, 727, 728, 729, 730, 731, 732, 733, 734, 735, 736, 737, 738, 739, 740, 741, 742, 743, 744, 745, 746, 747, 748, 749, 750, 751, 752, 753, 754, 755, 756, 757, 758, 759, 760, 761, 762, 763, 764, 765, 766, 767, 768, 769, 770, 771, 772, 773, 774, 775, 776, 777, 778, 779, 780, 781, 782, 783, 784, 785, 786, 787, 788, 789, 790, 791, 792, 793, 794, 795, 796, 797, 798, 799, 800, 801, 802, 803, 804, 805, 806, 807, 808, 809, 810, 811, 812, 813, 814, 815, 816, 817, 818, 819, 820, 821, 822, 823, 824, 825, 826, 827, 828, 829, 830, 831, 832, 833, 834, 835, 836, 837, 838, 839, 840, 841, 842, 843, 844, 845, 846, 847, 848, 849, 850, 851, 852, 853, 854, 855, 856, 857, 858, 859, 860, 861, 862, 863, 864, 865, 866, 867, 868, 869, 870, 871, 872, 873, 874, 875, 876, 877, 878, 879, 880, 881, 882, 883, 884, 885, 886, 887, 888, 889, 890, 891, 892, 893, 894, 895, 896, 897, 898, 899, 900, 901, 902, 903, 904, 905, 906, 907, 908, 909, 910, 911, 912, 913, 914, 915, 916, 917, 918, 919, 920, 921, 922, 923, 924, 925, 926, 927, 928, 929, 930, 931, 932, 933, 934, 935, 936, 937, 938, 939, 940, 941, 942, 943, 944, 945, 946, 947, 948, 949, 950, 951, 952, 953, 954, 955, 956, 957, 958, 959, 960, 961, 962, 963, 964, 965, 966, 967, 968, 969, 970, 971, 972, 973, 974, 975, 976, 977, 978, 979, 980, 981, 982, 983, 984, 985, 986, 987, 988, 989, 990, 991, 992, 993, 994, 995, 996, 997, 998, 999, 1000.

NAME _____
 ADDRESS _____

WORTHY ENCLOSURE



GRANGE HILL – THE GAME



Victory Media,
Caledonian Place,
London, W1 2AL
Telephone 01-439 1464

Available for Spectrum,
Commodore 64
and Amstrad: **£9.95**

Grange Hill ©/TM Redmond (Europe) Ltd