Vol.5 - Nº 8.   April 1992.

FORMAT

FOR SPECTRUM AND SAM USERS

TYPOGRAPHY
Better Letters By Design

£1.25 [UK R.R.P.]   ISSN 0963-8598

CONTENTS

Vol.5 № 8.   April 1992.

NEWS ON 4

POSTER MAKER FOR SAM

The SAM Coupé can now boast a Poster maker, Card maker, Stationary maker and Graphic designer - all in one package.

SAM PRINT, from new Swansea based company Noesis Software, allows you to turn out A4 posters and Stationary with ease. Greetings cards can be produced in one go by printing on A4 and folding twice.

The package, supplied on two discs, comes complete with several type faces and a very large collection of ready designed graphic images. Your own graphics can be added using the built-in designer.

SAM PRINT costs £14.99 and is available from Noesis Software, Unit 12, Oxwich Court, Fenrod Business Park, Valley Road, Swansea, SA6 8QP.

ROMANIA HOSPITAL APPEAL

A South Wales group of radio amateurs are off to Romania over the next month to help rebuild hospital facilities in the beleaguer country.

They are on the look out for second-hand/surplus building materials to help this rebuilding. They are also looking for medical supplies to take out with them, from simple Asprin and Cough syrup to Insulin and other drugs.

In addition they are looking for second-hand radio and computer equipment that can be sold at radio rallies to raise money for supplies and transport.

If anyone can help then phone Dave or Sheila on Swansea (0792) 653785. But make it quick, the first teams go out in early April.

SAM DISC FILING

Betasoft have released a disc-based filing program for the SAM Coupé called FILE MANAGER. Written to run under MasterDOS and MasterBasic the program allows up to 780k of information to be stored in one file. The program allows variable length records and contains all the features expected of a data filing program these days. However, as the program is in Basic you can tailor it to your exact requirements.

With the SAM 1Meg expansion fitted access to records is very fast. With no expansion the program will work quite happily on a twin drive system or on a single drive 512k Coupé although in that situation the total file size is more limited.

Betasoft are offering £10 off the total price if you order three programs from them at once. They are also now offering a free upgrade service for users of eallier versions of MasterDOS and Master Basic.

FILE MANAGE costs £12.95, see Betasoft's advert in this issue.

PFN - PRINT

Garry Rowland, renowned for his Spectrum Music programs, has launched a new utility for 48k or 128k Spectrum users. Called the PFN Print System it costs you just £1, yes ONLY £1 for tape and instructions.

PFN provides a set of routines that allow proportional spacing of printing on the Spectrum's screen, giving a much more readable and pleasing look. It allows printing in a variety of sizes and type faces.

Also available at the same time is PFN Editor. This costs just £4 and allows you to design your own type fonts or to amend existing ones. If you buy the PFN editor you also get the PFN print system as well.

Write to G.Rowland, PO Box 49, Dagenham, RM9 5NY.

News Credits: Nev Young, Bob Bates.

URGENT we need your news. Anything you think other people should know about. Items printed earn contributor 3 month extra subscription (please claim when renewing).

The Editor Speaks

It only seems like yesterday that I finished writing the last editorial, but in fact I've aged a year since then. Yep two more years and life begins or so they say - I can't wait (provided it doesn't mean going back to baby food and nappies that is).

This month sees the start of a new series entitled "HACKER'S CORNER". In the first few months Mark will be concentrating on showing you the rudiments of transferring multi-load games to disc on the Spectrum. He needs your feed-back though, your letters will be passed on to Mark Lambert if you send them in to the FORMAT office, but please bear in mind that he can only deal with suggestions or queries through Hacker's Corner.

Last month I asked for details of any known bugs in SAM Basic. So far only two reports - still I suppose it is a little early. Please, don't rely on someone else reporting the bug you have found - report it yourself. I will run a list in June's FORMAT if there is room.

On the subject of SAM, there are still many people who are dissatisfied with the Disc Operating Systems available. SAMDOS is only really usable by those who don't program, MasterDOS is much better - but not very user friendly in its syntax. ProDos is rather specialist because of its CP/M roots (a full review of ProDos should be ready for next months issue I hope). So what do SAM readers think. Is there room for another operating system? Are there things you would like to see included? Drop me a line with your comments and suggestions - especially if you feel you could become part of a team of programmers to produce a new system.

Many readers have written to me asking if it would be possible to start a series on Basic programming - similar to the series Carol is so ably running on machine code. I normally assume readers have a certain understanding of Basic when I edit articles for FORMAT, am I wrong in making this assumption? Well, not in general, but FORMAT is the only source of programming information still printed. This means newcomers to computing have only FORMAT to turn to. I would like to hear from readers who feel they need to learn more about Basic, from your letters I can gauge what sort of articles are needed.

Next, news of the second FORMAT's Demo Disc for SAM. This is now planned for launch in July, and I'm still looking for items for the disc. Spectrum users shouldn't feel left out either - I'm looking for items for a Spectrum demo disc as well. In both cases the routines can be small or large, everyone who has material published will receive a copy of the disc and we will also be organizing a readers vote, with cash prizes for the items voted 1st; 2nd and 3rd on each disc. Submissions should be original and sent on disc with instructions. All items will be considered for inclusion on future discs, some may also be selected for printing in FORMAT. Discs will be returned.

Oh, one last thing, Nev sent me the Help Page on IBM disc this month. I knew it was an IBM disc as soon as I opened the envelope - there was no label, so how did I know?

Bob Brenchley. Editor.

SHORT·SPOT

Edited By:- John Wase.

Lots of little goodies for you this month - and some apologies. Firstly; I've had a SAM breakdown. Along with pressures of work, a sick wife and a funeral in Scotland. This means that there are some errors and omissions. Nevertheless, I have managed to look at Peter Williamson's word processing program: indeed, I got the thing on disc, though I have some doubts about its total verification, as you will see later. Actually, I'm sure that Peter won't mind if I mention that it is not really a word processor in the real sense of the word. In effect, he has simulated, rather nicely, the operation of an electric typewriter. In this respect, it's quite unusual. If there is enough interest (write in to the letters page, please; not to me), we will publish it in sections, inviting improvements, additions and alterations from our readers. What do you all think?

Back to SAM. I was testing ProDOS, a program which emulates Digital Research's CP/M operating system, and runs on SAM. So I copied bits and pieces, and sent them down to Brian Gaff of BG Services, the vendor and helpmate, who was puzzled at a lot of the effects. In one case, all the functions were corrupted: the first disc sector was marked as a SAM file sector, but was left blank. Eventually I found that "Wordstar" would run, but that one of its overlays would not load - on my machine, that is. It worked fine on Andy Wright's. We deduced a RAM defect. But RAM is tested on SAM when you boot. No it isn't: only the first byte in each page is checked. So we're checking where the defect is. Beware; check your SAM's memory yourself!

ProDOS was originally conceived because of RS232 problems. Those who have an RS232 and want to use it to

input files might run into similar difficulties - Andy Wright did. Baud rate seemed not to matter: after between one and two hundred characters, he got transmission errors. Here's his hardware fix. He deduced that there was a problem with the electrical supply in one area of the printed circuit board. You remove the board from its case and have it the right way up with the gold plated connector which fits SAM nearest you. In other words, just the same way as it would be in operation. Done that? Good. Now look at the connector. Just above it, soldered to the board, at the left side is a resistor marked R3. Got that, too? And just below the left end of the resistor is a plated hole, right through the board; part of a track. There's plenty of bare lead to the left end of the resistor, and you can solder right through the hole. All we now need is a good fat looking tantulum capacitor, which we can solder across these five and zero volt lines, to smooth the supply in that area. Done it? What, already? Fine.

Here's a sterling effort from one of our regular contributors; L.G.Baumann of Cowies Hill, South Africa. He writes that it is almost an unwritten law that one should save programs or texts at reasonably frequent intervals, as an insurance against accidents. Unless you are very methodical, this can be a tedious (and therefore neglected) process. Here's the easy saving method used by Mr Baumann. It works with microdrives, +D discs or Discovery, and will work with other systems like +3 or SAM with only minor syntax changes. Type in the program, do LIST 9999, increase the variable z by one and RUN 9999. Change the cartridge or disc and RUN 9999 again for the backup. The program name must not exceed eight characters (six for the +3). This is because the

program adds a two-digit number to the file name. So successive saves would look like this:-

```
anewpgm10
anewpgm11
anewpgm12
```

except that the program also unclutters things by erasing earlier saves, leaving only two at any time. Using RUN has the advantage that all the variables are removed. These can sometimes give problems: for instance DIM a$(300,20) uses almost 6k of storage. Worse still, if the Discovery is very tight for room, saving the variables can mean that it won't reload. Sometimes, though, you still want the variables. Then use GO TO 9999. Here's the program.

```
9998 STOP: REM pgmsaver
9999 LET z=10: CLS : LET z$="anewpgm":
     PRINT "saving ";z$;z: SAVE *"m";
     1;z$+ STR$ z: PRINT "verifying ";
     z$;z: VERIFY *"m";1;z$+ STR$ z: I
     F z>11 THEN PRINT "erasing ";z$;z
     -2: ERASE *"m";1; z$+STR$ (z-2)
```

Nice one, Mr Baumann, and thanks for the contribution.

The next few items I typed on the Wednesday night before the funeral and a 300 mile drive the following morning. The Spectrum crashed on the last two lines. Now I'm retyping it on my Z88 in a pub in Scotland! What we do to meet the deadlines.

So over to Ettrick Thomson of Aldeburgh; another regular contributor. His first item is a super-subtle correction. Ettrick writes that G.Jackson's "Spectrum clock" in February's "Short spot" made him (Ettrick, that is) look at his "SAM clock" program again. There's a sign wrong. Ettrick sends apologies all round. The correct version with a "greater than" sign in line 10070 follows (incidentally, it's fortunate that the chance of the error in this line giving a wrong answer is very, very small). Nevertheless, here's the correct "SAM clock".

```
10000 DEF FN Lo=DPEEK SVAR 632
```

```
10010 DEF FN hi=65536*PEEK (SVAR 632+2
      )
10020 DEF PROC clock q
10030 IF NOT q
10040 DPOKE SVAR 632,0: POKE SVAR 632+
      2,0
10050 ELSE
10060 LET Lo1=FN Lo,hi1=FN hi,Lo2=FN L
      o
10070 IF Lo1>Lo2 THEN LET t=(Lo2+FN hi
      )/50: ELSE LET t=(Lo1+hi1)/50
10080 PRINT t DIV 60;":";t MOD 60
10090 END IF
10100 END PROC
```

Finally, Ettrick mentions again the problem of the digit "one" and lower case "ell". Neither I nor Bob can win, Ettrick. I have only a short time to put this together; if I type in programs into the word processor, I invariably get transcription errors. If they are on a disc, at any rate there are no mistakes, but it's all too easy to miss things like this. And all Bob has time for is a long wait for the postman, a quick file transfer, fit-cut-and-paste job, to get it to the printers. Again, he is very unlikely to spot these things. So our apologies for when things get fouled up, but please, please avoid lower case "ell". Like the plague!

Back to Alan Cox, who gets exasperated when he sees things like:

```
10 INPUT "Another game? (y/n)";a$
20 IF a$<>"y" AND a$<>"Y" AND a$<>"n
   " AND a$<>"N" THEN GOTO 10
```

For the Spectrum, you can at least simplify this by setting CAPS LOCK on by doing POKE 23617,8. This alters the system variable which tells the Spectrum which cursor mode it is in to 'C' mode - caps mode. For SAM, a more satisfying approach is:-

```
20 IF NOT INSTR("YyNn",a$) THEN GOT
   O 10
```

And this is one of the few useful tricks which Alan learned from Basic on a PC.

A little note on one of Alan's envelopes tells me that "Potty" (Short

Spot Jan '92) was "Pot Builder" by W.Crowther of Horsham, Sussex, for the 16k Spectrum, and first appeared in Sinclair Programs, December 1984. And Ray Burford of Stourbridge has a little modification. He writes that during "ALTER", the new radii were not correctly represented. The problem lies in the use of "OVER1", which does not unplot pixels. The solution? Change line 1005 to:-

```
1005 PRINT AT 10,0;"< 11 SPACES >"
```

The subroutine 6000 now becomes:-

```
6000 PRINT AT INT((175-ST)/8),0;"< 32
     spaces >"
6005 PLOT 127-W(I),ST: DRAW W(I),0: PL
     OT 127,ST-5: DRAW 0,10: PLOT 127,
     ST: DRAW W(I),0
6010 RETURN
```

which works perfectly even when all radii are zero.

Roy also mentions a small problem with David Finch's "Rotating Box" program. The drawings were appearing at the hi-res screen origin, and almost immediately going off-screen and halting the program. Solution - the PLOT 0,0 at the start of lines 90, 91 and 92 were wrong. And the mid-screen co-ordinates xos and yos had been assigned but not used. begin 90, 91 and 92 with "PLOT xos, yos and all will be well. Many thanks Roy.

Several readers have written in about Martyn Bader's contribution. In short, as Alan Cox mentions, these are described completely in a letter from Toni Baker, itself a follow-up of an article. This appeared in "ZX Computing", June 1986. It described an interrupt-driven machine code program that gave the same capabilities. It also used much more rememberable key combinations than the ones from Basic. All this was an emulation of the dreaded keypad. Alan chased this up by writing to Spain, as he would have, at the time, liked one. They were sold by Investronica, but only as a complete deal with a 128 - a Spanish one, to boot! An article in a Spanish computer magazine showed a Sinclair proprietary

ROM in the works, so that was that. If anyone is gifted in electronics and wants to make a keypad, a simple emulator should be possible. If you write, I'll put you in contact.

Next, a little contribution from Mr Southwell of Heaton Chapel, Stockport. Oh; I guess it's for SAM.

```
1 REM ********************
2 REM *** PASSLETTERS!  ***
3 REM *** BY P.SOUTHWELL ***
4 REM ***FOR FORMAT 1992.***
5 REM ********************
10 SCROLL CLEAR : CLS : MODE 1
20 CSIZE 8,16
30 POKE 23361,1: POKE 23264,0,0: RE
   M *THESE POKES ARE TO STOP 'ESC'
   AND 'BREAK'*
40 PRINT AT 6,1; PEN 4;"TO GET TO T
   HE MAIN SCREEN YOU HAVE TO GUESS
   THE PASS LETTERS!!": PRINT AT 9
   ,3; PEN 4; "YOU HAVE ONLY ONE CH
   ANCE."
50 GET a$: GET b$
60 ON ERROR GOTO 10
70 IF a$="*" AND b$="*" THEN GOTO 8
   0: ELSE CSIZE 8,16: CLEAR : PRIN
   T AT 5,3; PAPER 10; PEN 6; FLASH
   1;"W R O N G ! ! PASSLETTERS ! !
   ": PAUSE : NEW
80 CLEAR : CSIZE 8,16: PRINT AT 5,0
   ; PEN 6;"USE PASSLETTERS TO PROT
   ECT YOUR PROGRAMS.": PRINT AT 0,
   0; PEN 10;"TO CHANGE PASSLETTERS
   YOU HAVE TO ALTER 'a$ & b$' AT
   LINE 70"
```

Here's a nice little program from David Stokes of Launceston. He sent in a decimal-binary job, but most of us already have these. So I won't print it unless there's a demand. However, this one's a good little bit more intriguing. It draws bar graphs. It wouldn't take much to draw a border around the graph, push it further upscreen and, by means of a 64 character generator and an enlarging dump, label the axes and give it lots of versatility. Come on, all you Basic programmers; this is a lovely one to modify. It originally came from a mag, but David can't remember which one.

```
1 CLS : PRINT AT 8,29;"100": PRINT
  AT 14,30;"50": PRINT AT 19,30;"10
```

```
  2 PRINT AT 4,0;"JAN          JUN
    DEC"
 10 BORDER 4: INK 0: PAPER 7: PRINT A
    T 4,0;"JAN          JUN
    DEC"
 20 PRINT AT 0,10;"BAR GRAPH"
 30 LET C=0: LET X=0: LET Y=9: LET I=
    0
 40 INPUT "NUMBER OF BARS (1-12) ";B
 50 IF B<=0 OR B>12 THEN GOTO 40
 60 INPUT "PERCENTAGE OF BAR (1-100)
    ";P
 70 IF P<=0 OR P>100 THEN GOTO 60
 80 FOR J=0 TO P: FOR K=X TO Y
 90 PLOT K,J
100 NEXT K: NEXT J
120 LET X=X+20: LET Y=Y+20
140 GOSUB 1000
150 IF c=b THEN STOP
160 GOTO 60
1000 DRAW 5,5: DRAW 0,-P: DRAW -5,-5
1010 DRAW -9,P: DRAW 5,5: DRAW 9,0
1019 LET c=c+1
1020 RETURN
```

An even more intriguing contribution from David is this circle program by P.Berry, published in an unknown magazine a long time ago. This principle is rather nice, and could be applied to a lot of situations. Try it...

```
  5 REM CIRCLE PROGRAM
  6 REM by P.BERRY
  7 PRINT #1;"Use p for right, o for
    left."
 10 FOR f=23296 TO 23296+72
 20 READ a
 30 POKE f,a
 40 NEXT f
 50 DEF FN c(x,y,r)=USR 23296
 55 LET x=127
 60 FOR r=1 TO 87 STEP 1.2
 70 LET x=x+(INKEY$="p")-(INKEY$="o")
 80 LET I=FN c(x,87,INT r)
 90 NEXT r
100 DATA 221,42,11,92,221,110,4,221,1
02,12,221,70,20,14,0,80,203,58,21
3,205,38,91,209,12,122,145,87,48,
4,122,128,87,5,120,185,48,237,201
,205,41,91,120,65,79,205,47,91,12
0,237,68,71,125,129,212,59,91,125
,145,216,197,79,229,124,128,71,25
4,176,220,223,34,22
5,193,201
```
A "quickie" from Ettrick, who has

been very prolific this month. Stuart Hughes wrote in March's FORMAT. He wants to save a string array as a code file, then load it back into a string. This is very easy in SAM.

To save the string array A$ as a code file:-

```
SAVE "name" CODE LENGTH (0,A$), LENGTH
(1,A$)
```

To load a code file into the string array A$: you must know the length of the file, N, say; then:-

```
DIM A$ (N): LOAD "name" CODE LENGTH (0
, A$)
```

You could do the same thing with a number array, but the length of the code file will be 5 times that of the array.

Next, a comment from Kevin Gould. Poor Kevin's been trying to modify his Tasword 2 with two FORMAT programs. One, by Istvan Ordog, appeared in "Short Spot" recently. The other was by Clyde Bish. Well, Kevin, Clyde's programs only get as far as Bob. I'm a contributor from miles away! the best solution is for me to ask Clyde, through this column, to drop me a line with his address on, agreeing that you may have it. Then you can correspond. As soon as I can dig it out, I'll let you have a copy of Istvan's disc. Howzat!

Finally, a couple more from Alan Cox. Roy Burford wrote in last month, saying that he liked Alan's "Block delete" from his archives and asked for a "Block renumber". The problem is that there are plenty in code. Alan's solution is to use a pair of programs. One deals with all the line numbers. The second finds all the GO TO and GOSUB references, and thus easily lets you change these by hand. These are the two programs...

```
 10 REM Basic renumber program on Spe
    ctrum
 20 REM by F W White
 30 REM  Published in Popular Computi
    ng Weekly 10-16 April 1986
```

```
 40 REM To use program GOTO 9990, the
    n respond to the 'Increment' quer
    y
 45 REM Note - the variable in lines
    9994 and 9996 would be more expla
    natory if called New - but SAM wi
    ll not accept this as the name fo
    r a variable:the Spectrum will
 50 REM note - line 9995 - that the p
    rogram will not renumber itself
9990 LET s=PEEK 23635+256*PEEK 23636
9991 INPUT "Increment?";I
9992 LET n=1
9993 BEEP .01,50
9994 LET ne=I*n
9995 IF 256*PEEK s+PEEK (s+1)>9989 THE
     N STOP
9996 POKE s,INT (ne/256): POKE s+1,256
     *(ne/256-INT (ne/256))
9997 LET s=PEEK (s+2)+256*PEEK (s+3)+4
     +s
9998 LET n=n+1
9999 GOTO 9993
```

```
 10 REM Seeker program for the Spectr
    um
 20 REM by R Luxton
 30 REM  published in Popular Computi
    ng Weekly 17-23 April 1986
 40 REM  The program can find the lin
    es in which any tokens/keywords a
    re used.  However, in the presen
    t context it can be used in parti
    cular to find GOTOs, GOSUBs, REST
    OREs and LISTs
 50 REM The program should be merged
    with program of interest, after w
    hich RUN 9993
 60 REM Spectrum BASIC does not allow
    a token/keyword to be entered di
    rectly in response to INPUT, so a
    t the prompt enter THEN (ie symbo
    l shift +G) followed by the token
    /keyword key of interest, backspa
    ce to the gap between the two and
    DELETE - then ENTER
9993 GOSUB 9999: CLS : INPUT "Token so
     ught ? ";a$: LET q=CODE a$
9994 IF FN c()=9993 THEN STOP
9995 FOR j=FN a() TO FN b(): IF PEEK j
     <165 THEN GOTO 9997
9996 IF PEEK j=q THEN GOTO 9998
9997 NEXT j: LET s=FN b()+1: GOTO 9994
9998 BEEP .25,30: PRINT STR$ FN c(); T
     AB 5; PAPER 6;CHR$ PEEK j: GOTO 9
     997
9999 LET s=23755: DEF FN a()=s+4: DEF
     FN b()=s+3+PEEK (s+2)+256*PEEK (s
```

```
+3): DEF FN c()=PEEK (s+1)+256*PE
EK s: RETURN
```

And that's really all for this month, folks. I have had a lot of contributions, but they are from only a very few contributors. Please, if you have any favourite snippet culled from an old mag years ago, it's worth sending in. Even more so if it's a new program. See you next month. Keep 'em coming to John Wase, Green Leys Cottage, Bishampton, Pershore, Worcs, WR10 2LX. And thank you all.

TAILPIECE. Some of you have sent in discs with the envelope marked with things like "fragile". "Care". "Magnetic media". "Don't touch or you'll get Computer pox". Don't bend, roll, jump on, heat, cool or get wet". And so on. Suitably scared, our post lady has been carefully putting these items in the coal house (which floods in the rain) instead of through the letter box in case they should break on hitting the floor. Just goes to show - the power of communication!

YOUR LETTERS

Dear Editor,
On the front of Your Sinclair is a game demo for SAM called Batz & Ball. To save a copy to SAM disc:-

```
 5 DEVICE T
10 MERGE " " CODE
15 DEVICE D1
20 SAVE "Batz&Balls" CODE 212992,38
   342,212992
```

Yours sincerely, J.Begg.

Thanks for the info, there is also a copy of the Batz and Balls demo on the latest Samco news disc. Ed.

Dear Editor,
Referring to Tom Fisher's request to connect an Amstrad CPC 464 monitor to a Spectrum +2. If the +2 (grey) has the same connections on the back as the +2a (black) RGB/Peritel then the connections are as follows.
The CPC 464 video output connector (6 pin DIN) this is the socket marked on the back on the computer as MONITOR. Looking at the back of the 464 you have the following:-

```
1 RED
2 GREEN
3 BLUE
4 SYNC
5 GND
6 LUM
```

The +2a socket, marked RGB/PERITEL is as follows (again from the back):-

```
1 +12V
2 GND
3 AUDIO OUT
4 COMPOSITE SYNC
5 +12V
6 GREEN
7 RED
8 BLUE
```

I hope this information is of use.

With the exception of Tasman products what else is there on the market for the +2a to make use of the 128k memory. My local computer shop has hardly any Spectrum software, either 48 or 128k.

Yours sincerely, Kevin Gould.

As far as I know the +2a is the same as the +2 so thanks for the CPC details Kevin. Ed.

Dear Editor,
Nice to speak to you the other day, thank you very much for maintaining the hot-line. I am sure it must be a nuisance to you and your family at times. For Disciple/Spectrum + users like myself, the service is an incredible help.
Your advice concerning my Citizen 120D printer was bang on the button!
As I said, was getting no descenders in draft mode, p q g's etc., being printed as o's.
As you suggested, I cleaned the printhead and got bags of RED ink on cotton buds!!! Over Xmas I did some cards with the Spectrum/Art Studio and a RED ribbon. I normally use CITIZEN ink ribbon cartridges, they are much the same price and more reliable than "compatibles." but this wasn't a CITIZEN; serves me right! SO!!!! I NOW HAVE DESCENDERS, what a saving, what a relief.
Incidentally, I took off the cover, lubricated and generally cleaned up the machine, it wasn't too dirty but some fluff and paper bits were there. I have lost a squeak and it runs smoother. What I couldn't do was remove the printhead ribbon connector. The manual was no help, two arrows moulded on the connector pointed UP. In the event I was feared something would break with the resistance I found, so I gave up and carefully cleaned the head with it connected. So much for CITIZEN'S "It is an easy job

13

for owners."

Best wishes and my thanks,

Yours sincerely, C.Leslie Pollard.

It is amazing what a good clean-up will do for a printer, now mine has been serviced it is just like new. Ed.

Dear Editor,

In the Vol 2 issue 9 edition of FORMAT you published an article by Bill Long called TOTAL RECALL, which gave and un-erase program for the Disciple/PLUS D. I duly typed in the program and used it whenever I needed it until I converted to SAM. Since it then did work, of course, I put it to one side, promising myself that I would convert it one day.

This I have now done. I cannot recall that you have published a SAM un-erase program - but my memory is not very good these days - and I wondered whether you would be interested in a short article up-dating Bill's program.

As I am sure you know better than I, the slightly tricky point is that whereas the Disciple/Plus D stored two copies of the indication of file type, and only overwrote one of them when it erased a file, SAMDOS does not have this second copy, and it is necessary to do a bit of calculation to find out what the erased file type was.

If I have done my sums correctly, you can find out from bytes 221 to 223 in the appropriate directory entry. (Incidentally, I ignore the ZX SNP file type, since I have no interest in it, and I only handle the other 5 file types set out in the SAM Technical Manual, edition 3.0 viz BASIC, Numerical array, String array, CODE and SCREEN$). If byte 221 is equal to 32 then the file type is CODE, if it is greater than 32 then it is an array, with bit 5 or 6 set for numerical or string respectively, if it is less than 32 then it is either SCREEN$ or BASIC, and if byte 223 is greater than 127 it is BASIC ELSE it is SCREEN$ - because the MSB of the file length MOD 16384 for a BASIC file has 128 added to it - since the unit above is a page, this MSB can never be greater than 64.

Other fairly trivial changes to the original program are (i) use READ AT and WRITE AT instead of LOAD @ and SAVE @, (ii) only five file types need to be covered, with different numbers indicating them, and (iii) SAM does not like NEXT as part of a variable name - so use NXT.

I know that there are various un-erase programs available commercially, so you may take the view that to publish anything more on the topic is unnecessary. I should be glad to learn what you feel.

Yours sincerely, Alan D.Cox.

TOTAL RECALL is one of the most useful programs we ever published, I use it at least once a week. I'm sure readers would like to see the listing for a SAM version if you send it in. Ed.

Dear Editor,

I have been meaning to put finger to key for a while to say thank you for FORMAT. Without this superb publication my usage of the Spectrum would have been far poorer. Throughout the pages of each month's issue I have learnt much about computing in general besides that specific to this lovely little machine. This is particularly so in the last couple of years when availability of both hardware and software has been nigh on absent in my locality (and in fact it is not so different countrywide here in South Africa).

This problem was one that FORMAT was responsible for solving. My main supplier in Pretoria moved on to other business leaving me high and dry as regards anything for both Spectrum and QL. One of FORMAT's regular contributors has been Len Baumann from Cowies Hill. On the off chance I looked him up in the telephone directory and was able to contact him. Unfortunately I was not able to help him with his problem of converting OCP Finance Manager to PLUS D usage but he was able to put me on to an active supplier and repairer of things Spectral. (I am glad to see that Len's problem has also been sorted out via the pages of FORMAT - Nev's Help Page

of April 1991).

Now I hope I am able to do some return helping. G.New in June's issue was enquiring about HiSoft C on disc. I have recently purchased a copy direct from Microsnips (25-29, Grange Road West, Birkenhead, Merseyside L41 4BY) at a price of £25.54. And again thanks to FORMAT! If it had not been for that article in Vol2 No8 I would not have been able to make use of the disc that was supplied. Apart from the standard instruction manual there was no specific information as to how to set up a working copy from the master disc. This was very well explained in the aforementioned article.

In your letters of May 1991, A.J.Erasmus complains of delays in obtaining items from the UK. (I assume he must have the same problems in obtaining locally as do I). The HiSoft C mentioned above took four days to get from Liverpool to my desk here in Bedford. Even the local Post Office commented on the unprecedented speed! I haven't dealt with either Lerm or Enigma but have done business with many other of FORMAT's advertisers and, on average using airmail, find that from sending to receiving orders is of the matter of three to four weeks. I don't reckon that to be too bad. I second your statement praising the companies that advertise in FORMAT - fast, friendly and helpful. Professional journals airmailed from the UK arrive here before those published on the same day and posted in Jo'berg. In some 120 journal/months only three have gone missing. Seamail is not only slower but less reliable: a lot of non-appearances when I used that method! Customs only seems to be a problem with larger parcels: single cassettes or discs and two PLUS D's have slipped through unnoticed. The only time I got lumbered with duty was a packet of four programs from MGT which set me back the equivalent of some £70, which, if I remember correctly, was more than the actual cost. It didn't delay arrival - more's the pity with that bill!!

Please keep FORMAT rolling! I and my four Spectrums are highly dependent on it. Without FORMAT I would be in total isolation. For the last four years all

my correspondence has been done using either Tasword+2, Fastword or, more recently, Word Master. The quality of the product of these word processors has been commented on by many of my correspondents. At present I have two PLUS D's attached via a Twoface enabling me to run GDOS, BetaDos or Unidos. My original rubber key 48K is awaiting my daughter to take an interest: she is just starting the Macmillan "Learn to Read" program on it. Keep up the good work!

Yours sincerely, Mike Bennett.

Thanks for your comments regarding air versus sea mail Mike. We are looking at ways in which we can narrow the gap between the costs of sea and air so we can persuade more readers onto that service.

* - * - * - * - * - *

Letters may be shortened or edited to fit on these pages.

This is YOUR letters page so it is up to you, our readers, to fill it. Send your letters, on any subject you feel would interest other readers. Send to our usual address, keep them as short as you can so we can fit in as many as possible.

HACKER'S CORNER

By:- Mark Lambert.

While I am talking about Spectrum programs at the moment, much that I cover will be of interest to SAM owners as well. I am starting with the problem of converting multi-load tape games to disc because the editor tells me that is what many readers have been asking for. I will cover other aspects of hacking, and turn some attention to the SAM Coupé, in later issues.

Software pirates please note, these articles are not intended to help you rip off other people's hard work - it's illegal! There will be no information on making unprotected tape copies. However, if you are a disc user, have bought a game and want to have a more convenient copy for yourself then this should interest you. Remember though, if you part with the original tape copy of the program you must erase ALL disc copies and of course you must not pass on any disc copies to anyone (not even your best mate).

Although the snapshot button is very useful for transferring many games to disc, modern "multiload" games still rely on tape to load the levels. These articles will show you how to transfer whole multiloaders to disc, and give examples of a few games.

The things you will need are:-

1) A disc drive and interface (obviously). These conversions should work on the Plus D and Disciple, owners of other interfaces may like to modify the code to work with their machines.

2) A working knowledge of machine code.

3) An assembler.

4) A snapshot hacker (such as S.D.Software's HACKER'S WORKBENCH) or perhaps a Multiface with a monitor program.

5) A work disc.

Remember to always work on an otherwise empty disc, and write protect it whenever you test your code. Using the drives from machine code could erase files if you make a mistake. Never save on the tape that you bought your game on. Keeping backups of important files is always a good idea.

THE TECHNIQUE

You must have a system of naming files, so that you can identify what they are when you need them later. I give files the same name as the game, with a suffix to tell me what the file is.

To convert the games to disc I do the following operations. These steps are just guidelines, and how to do them will be explained more as I give examples.

1) Snapshot the game to disc (when it is not loading a level). I give this the suffix "0".

2) Get the game into a position where it is loading and use the snapshot hacker to find the level-loading routine (the "loader"), and the calls made to it.

3) Save the loader to disc with the suffix "L".

4) Create some method of transferring levels to disc. This is usually a machine code program to use the loader, and a BASIC program which calls it and saves the levels to disc. The assembler source has the

suffix "TS", and after it is assembled, the machine code is saved with suffix "TO". The BASIC program has suffix "TB" and the levels are usually given suffixes "1", "2" etc.

5) Make a new disc loader to replace the tape loader in the game, my suffixes are usually "NS" for the assembler source and "NO" for the machine code produced. 6) Replace the tape loader inside the game, replace the tape loader with the disc loader.

It is important to remember that the aim is to make the game believe it is still loading from tape. If it is not "convinced" (the disc routine is not thorough enough in its imitation) then the game will crash.

To perform your own conversions, you will need to know a little about tape and disc loading routines. For Plus D / Disciple loaders see the article on Hook Codes in Format volume 2 issue 4.

ABOUT THE SPECTRUM ROM LOAD ROUTINE

Spectrum tape files consist of two blocks separated by silence, a header and a data block.

Each block has a leader tone, followed by a "flag" byte (0 means the block is a header, 255 means it is a data block), and then the bytes which make up the block.

A header contains 17 bytes:-

1 byte: Gives the type of file,
   0=BASIC,
   1=numeric array,
   2=string array,
   3=code.

10 bytes: Filename as ASCII, padded to 10 characters with spaces.

2 bytes: Length of the file.

2 bytes: Start address of a code file, or autorun line number for BASIC.

2 bytes: Length of the program not including variables for BASIC.

The bytes contained in the data block form the BASIC program and variables, code file or array.

To load a block, you would set up the Z80 registers in this way: A=Flag byte, IX=Address in memory to load the block at, DE=Number of bytes to load (not including the flag byte, which is checked rather than loaded) set the Carry flag to indicate loading (if reset the block is verified), then call the ROM loading routine at address 1366.

The ROM loading routine returns with the Carry flag set if the load was successful, or reset if there was an error. If the flag byte of the block on the tape did not match the byte in A then L will contain the flag byte from the tape.

TURBOLOADERS

Most (almost all) turboloaders are copies of the ROM loading routine with timing changes and alterations to the border colour or whatever, and as such they have certain similarities, particularly the use of IX and DE. This makes loading calls quite easy to spot, as IX is a little used register. Also, by comparing program code with the ROM it is not difficult to spot the end of a loading routine by the sequence: OUT (254),A SCF RET which is an uncommon fragment of code.

The other alteration a programmer may make when creating a turboloader is to remove the instructions which control the interrupts, so before calling a turboloader it is important to perform a DI to disable interrupts, otherwise the timing will be upset and the file will not load. Also use EI to enable interrupts again after returning from the load (to restart the key-scanning and so on).

Next month we will look at our first real conversion, so make sure you send off your renewal if it is due so you wont miss the next instalment.

Z80-SUBS
MACHINE CODE SUBROUTINES.

Edited By:- Bill Nicholls.

It's me again, back with a few Z80 machine code routines which I hope many of you will find useful. Sorry I was missing last month, Bob really should get his priorities right, fancy cutting your favourite column.

A big thank you to all who have written to me, your comments have been very interesting but I wish a few more of you had included routines I could print in this column.

Still we will start this months foray into the world of the Z80 with an offering from Geoff Eglinton. You will remember I printed a routine a while ago to allow a 'Relative Call', well Geoff was the first person to send in a version that allows a two byte displacement instead of the one byte displacement in my routine. Here is his version:-

```
RCALL2B EX (SP),HL ;Stack HL & get RET address
        PUSH DE ;Stack DE
        LD E,(HL)
        INC HL
        LD D,(HL) ;DE now holds displacement
        INC HL ;HL is now return address
        EX DE,HL
        ADD HL,DE ;Routine address now in HL
        EX (SP),HL ;DE recovered and
        EX DE,HL ;routine address stacked
        INC SP
        INC SP
        EX (SP),HL
        DEC SP
        DEC SP
        RET ;Indirect jump to routine.
```

As with my original routine you need to store RCALL2B at a fixed location in memory but you can then implement relative calls by doing:-

```
        CALL RCALL2B
        DEFW Address of routine
```

Geoff also points out that on SAM

you can store the address of the routine in the system variable RST30V at location 23282 (5AF2 hex) you then only need to use a RST 48 followed by the offset, that will save two bytes.

Next, a little routine from Jos Vervloet in Belgium. The routine provides a simple INPUT routine which calls the ROM key scanning routine and then prints the resulting character to the lower screen.

```
INPUT LD (IY-50),0 ;Zero LAST K
      RST 56 ;Restart to read keys
      LD A,1 ;1=printinh to lower screen
      CALL 5633 ;open stream for printing
      LD A,(IY-50) ;recover character
      AND A ;set flags
      JR Z,INPUT ;loop if no key press
      CP 12 ;is delete key pressed?
      JR Z,END ;if Z flag then goto end
      RST 16 ;print character
END   RET ; exit routine
```

Returning to your comments received in recent letters. Several of you have asked which assembler I would suggest they buy. Well that is really like asking which make/model of car should you buy. It is, so much, a matter of personal taste. I use three different assemblers on the Spectrum, each has its good and bad points and each I use in a different situation. Over the years I have used many more, some very crude and simple - others so full of bells and whistles that you needed to keep the manual in front of you at all times. When talking about any Spectrum software one thing I am always quick to point out is that it is all so cheap (when you compare it to software for other machines) that you can afford to buy several programs and play around with them until [you] decide which is best for you.

A FIRST LOOK AT typography

By:- Ian Ross.

The appearance of this article in Format is perhaps an anachronism as it really has nothing to do with computing. It is destined for owners of PCG Wordmaster-Typeliner-Headliner who feel tempted to have a go at using the font editor for their own purposes - altering fonts, adding foreign accents, inventing logos or producing whole new fonts - but who feel the need of some background knowledge about typography.

HISTORICAL BACKGROUND

In prehistoric times, the scenes of hunting and war painted in caves - ideograms - represent the first known method of written communication. Next came pictography, the use of simple drawings (a bit like our signs for ladies or gents toilets) in succession to tell a story. Egyptian hiero-glyphics introduced the use of signs to represent sounds, an important stage in the invention of an alphabet. Subsequently the Phoenicians produced the first phonetic alphabet consisting of 22 signs representing consonants only and read from right to left. About 1000 BC the Greeks used an alphabet containing both consonants and vowels and read from left to right. The Greek alphabet universally taught today dates from 400 BC. The great monument to the alphabet as we know it today is Trajan's column in Rome; it contains an inscription of extraordinary beauty and clarity and equal to anything that might have come from, say, the Oxford University Press. This can be considered the prototype of Western typography. At first there were capitals (square shaped or rustic and used for inscriptions) and cursives, the originals of our lower case characters and used for correspondence.

It is just about now that we start to run into elements of typographic design - lines of different thickness, circles and curves of varying thickness and serifs. The earliest stone cut Roman capitals were without thick and thin strokes. Growing popularity of monumental inscriptions led to greater sophistication. Painting of letters before chiselling involved the handling of a bush which, in its turn, affected the shape curves. Serifs were a finishing stroke presumably ued to optically strengthen the ends of strokes.

Uncial writing, developed out of square written capitals, became common from the fall of the Roman Empire in the fourth century. There followed half uncials and the medieval scripts, merovingian, caroline, gothic etc. The Renaissance revived classical roman lettering - capitals like those on Trajan's column and lower case letters somewhat resembling present day italics or calligraphic scripts. The invention of printing in the fifteenth century produced at first Gothic typefaces including Fraktur but later, the upper and lower case letters familiar to us today. The great pioneering names are Bembo (1495), Jenson (France, 1460), Garamont (France 1544), Caslon (England, 1720), Baskerville (England, 1757) and Bodoni (Italy, 1780) - to name just a few.

Finally, modern types came in with improved technology and better paper surfaces. A greater contrast was possible between thick and thin strokes. The early nineteenth century saw the introduction of "sans serif" or "grotesque" typefaces, the popularity of which spread rapidly and which is now a commonplace. Sans serif simply means the absence of serifs; the type looks simpler and more austere. the most famous example is perhaps Bauer's Futura (1927).

For this historical summary, I have drawn heavily on James Hutchison's "LETTERS" (Herbert Press, London, 1983, £7). This small attractively produced book is most entertaining and informative both in the way it is written and in its illustrations; it is strongly recommended.

Why bother with a historical summary at all? The reason is that typefaces differ in character as well as in form - as do also their constituent parts, strokes, curves, dots, serifs. Making up a font needs an appreciation of its character and this is helped by an awareness of the context of typefaces as a whole, including the historical context - the growth of sophistication both cultural and technological, changes in demand and changes in fashion. Choosing a typeface depends on the use for which it is intended; for example, one would not use the capitals of a stately Caslon or Baskerville typeface to advertise a children's picnic party.

WORK WITH THE FONT EDITOR

How best to make a start? Drawing characters is a painstaking and fiddly business and a succession of false starts and unsatisfactory results can be very frustrating for a beginner. The production of a complete font is long and laborious. Probably the best thing is to set oneself limited goals at the outset, to make things as easy as possible for oneself and to remember that the object of the whole exercise is to have fun. A first objective could be the production of single characters, symbols or logos of the sort contained in the PCG Dingbats font - if font it can be called. Other examples may be found in Rudolf Morley's "Handbook of Pictorial Symbols", Dover Publications, New York, obtainable in the UK at £6. Other ploys are the addition of foreign accents to an existing font and making up scientific symbols.

MAKING UP A NEW FONT

Try and find source material. The most obvious for PCG users are PCG's own fonts, not only those originally produced but the later and somewhat larger ones. Then use the font editor to see how to draw such things as lines sloping at different degrees of steepness and curves which vary in thickness. A search round bookshops may produce one or two modestly priced publications but I have found it necessary to spend £27.50 on Jaspert, Berry & Johnson: "Encyclopedia of Type Faces", Blandford, London. Admittedly, such expenditure is only for the very keen or for those who enjoy browsing around type faces of many different kinds.

On getting down to the actual work, remember to make things easy for oneself. Acquiring a new font usually means cribbing from somebody else's work or from one's own previous efforts. The copying or creation of a new font can be helped by means of a number of short cuts: 1) Start off by adapting a font in your possession which bears a resemblance to the desired one e.g. thick, thin, italic, ornate, serif or sanserif etc. 2) Handle together "kindred characters" which resemble one another in respect of curves, slanting lines etc., e.g. c-e-o, b-d-p-q, h-k, i-j-l, v-w, B-D, C-G-O-Q, P-R, H-I-K-L, E-F, V-W, 3-8, 6-9, 1-4 etc. S,x,y and z, whether upper or lower case, usually need special attention. M and N in upper case are not always so closely related as might appear. X and Y do not necessarily have the same slant as the v and w in the same typeface. 3) Where practicable, one might wish to use the rescaling facility in a programme like Artstudio to draw key characters against a magnification grid - the characters being of a chosen size - and then copy them, pixel by pixel, on to a font editor grid. 4) It is essential to decide at the outset on the dimensions of your characters, both upper and lower case, as, if they are not consistent, they will look hopelessly wrong when seen in succession.

A sample text should be kept at hand for displaying and assessing newly completed work. Are the characters

correctly aligned with one another and are they evenly spaced? Look also at the general appearance of the new collection of characters. Does the result correspond with one's hopes? Here is the sample text that I use:-

```
!"#$&'()*+,-./0123456789:;<=>?@ABCDEFG
HIJKLMNOPQRSTUVWXYZ[\]^_£abcdefghijklm
nopqrstuvwxyz{ }~
```

The quick brown fox jumps over the lazy dog. PARIS, FRANCE.

As far as technique is concerned, much of the work amounts to laborious and painstaking copying; a certain amount of artistic judgment together with an eye for proportion is needed, also an ability to compromise from time to time in deciding on the final shape of a character. It will be found, for example, that the Typeliner grid is not deep enough to permit ascenders and descenders of the length that conveys to many type faces much of their charm and essential character; it may therefore be necessary to give up altogether or to make to do with less than the ideal. Some study of shape rendering is useful - drawing steep or shallow slopes, ovals and circles which vary in thickness, the choice of thickness itself. Possibly through lack of practice, I find it particularly tiresome to draw curves of varying thickness and slants of varying steepness. But one dare not ignore them; otherwise the letters will look wrong and one will not achieve the typeface which appeared so seductive originally. In this connection, do not forget the importance of serifs as their presence or absence can completely transform the whole appearance and character of a type face. Some may feel that they are not artistic enough for this pursuit; this is not necessarily true and one could be agreeably surprised. After all, pleasure in the shape of a car or in the cut, material and colour of clothing - which a lot of us have - is proof of some degree of artistic sensibility.

A reading of this article will doubtless convince some that typography is not for them. To those who may feel tempted to have a go, I wish good luck and enjoyment.

CARLTON

The PCG font which most closely resembles the old style classical Roman lettering.
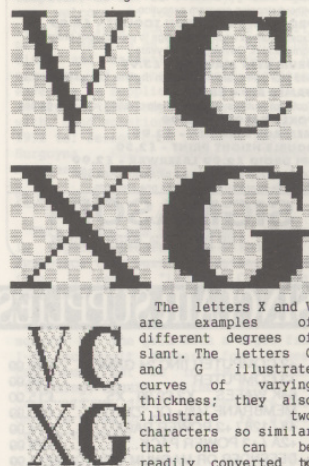
The letters X and V are examples of different degrees of slant. The letters C and G illustrate curves of varying thickness; they also illustrate two characters so similar that one can be readily converted to the other, i.e. the C modified so as to form the letter G.

BRITISH RAIL ANNOUNCES DEARER TRAIN DELAYS

THOUGHT SPOT.

By:- Jeremy Cook.

Welcome to you all. Come in, come in, there's plenty of room for everybody. Are you sitting comfortably? Then I'll begin.

I'm afraid I've let you all down again and the hanoi solution will not be found here this month. No real excuses, but when I get a lot of assignments I tend to neglect Thought Spot, as has been the case. Sorry. I now, however, have a short break in which I hope to make up the shortfall.

This month we do have the famous prize puzzle, for which the prize is (as if you didn't know by now!) ONE YEARS FREE SUBSCRIPTION to FORMAT. In addition we also have some ordinary, everyday puzzles to wrap your intellects around. So, without further ado, let thinking commence.

PRIZE PUZZLE No.15: JEALOUS HUSBANDS

A group of married couples are out strolling one day when they come to a river. Luckily they find a small two-person boat moored on their side. In the middle of the river is an island. The party decide to cross the river, but the husbands are as jealous as they are young and handsome (this is supposed to mean that they are very jealous), and so a woman can only be in male company if her husband is present. How does the group get across to the other bank?

You may have noticed that I did not tell you how many couples there are. That is your problem this month. Write a program to work out the sequence of journeys required to get all the couples from one side of the river to the other, via the island, for any number of couples. If there are n couples, then the least number of journeys needed is 8(n-1).

This is another classic problem (though not usually solved by computer, as far as I know), and you are welcome to look it up in a book. Your solutions should be as short, neat, readable as possible. Note that discs and cassettes will only be returned if a stamped self-addressed envelope is supplied. As always I am interested in anything you come up with. Send solutions in by 1st July 1992 to: Jeremy Cook (Thought Spot), 6 Burgoyne Road, Sunbury-on-Thames, Middlesex, TW16 7PW.

FIGURE IT OUT

This column has been rather lacking in logical problems, so I though I'd reproduce this puzzle I rather liked from a recent "Logical Challenge" magazine.

Use the clues below to fill each square of the grid with a digit from 1-9. Each digit appears four times, but must not be in two adjacent (touching) squares either across or down. If a digit is used more than once in a row or column it is stated in the relevant clue.

	1	2	3	4	5	6
1						
2						
3						
4						
5						
6						

Across
1. Two twos. Total twenty-eight.
2. Consecutive numbers in ascending order.

3. Two ones. Two sixes. No seven.
4. Two eights. Two nines.
5. Two fours. Three is the only odd number.
6. Two fives. No even numbers.

Down
1. Two eights. Two sixes. Total thirty-three.
2. Two ones. Four is the only even number.
3. Two twos. Five is the only odd number.
4. Two nines. Three is the lowest number.
5. Six is the highest number.
6. Two sevens. Eight is the only even number.

UNDER ATTACK

What is the smallest number of chess pieces that can be placed on a chess board so that every square is either occupied or being attacked, if the pieces are all (i) pawns, (ii) knights, (iii) bishops, (iv) rooks, (v) queens, or (vi) kings.

LANGUAGES

Find the languages. Only alternate letters have been given. For example, -N-L-S- is ENGLISH. (This puzzle was sent in by Robert Brady. Thanks also to everyone who has sent me puzzles- they haven't been forgotten).

1. -A-T-S-   7. -U-C-
2. -R-N-H   8. -U-M-S-
3. -E-G-L-   9. -R-E-
4. -A-T-N-S-   10. -C-L-N-I-
5. -W-H-L-   11. -F-I-A-N-
6. -U-K-S-   12. -R-U
(and now some from me)
13. -A-L-C   15. -L-M-S-
14. -A-L-O-   16. -A-A-O-

1984

Use the ten digits to make an improper fraction equal to 1984, with the numerator a seven digit number and the denominator a three digit number.

Once again we must apply the brakes, stop the engine and turn off the headlights of this not-very-well-oiled

machine called Thought Spot. As we close the garage doors behind us you may like to ponder this: everybody has heard of the Seven Deadly Sins, but how many can you name?

Just before we go, you may have wondered what "B..G..B..B.." was at the end of last months column. It was in fact a poor attempt at giving the impression of Porky pig (of cartoon fame) saying "Thats all folks", but it was printed in the wrong place. So now you know. Byeee.

----oOOo----

SOLUTIONS TO MARCH'S PUZZLES

Calendice:

Die 1- 0,1,2,3,4,5
Die 2- 0,1,2,6,7,8

The trick is that the 6 upside-down doubles as a 9.

View to a ?:

You may have your own building, but this is probably better-

There must be a word for it:

Marsupials, Jetsam, Hyperbole, Sericulture, Numismatics, Mnemonic.

Rebus:

this was a bit silly, but the answer is- hors d'oeuvre (oar, derv)

----ooOOoo----

MACHINE CODE
WITHOUT THE TEARS

Part 8.   By:- Carol Brooksbank.

So far, when we have wanted to print something on screen, whether a character or a graphic byte, we have made use of RST 16. We have not needed to know anything about the screen layout - the way the file of screen bytes is arranged in memory - because the ROM routine called by RST 16 has taken care of it all for us. Now we are going to take a look at that.

This is where things become rather different for Spectrum and SAM users. Spectrum users have one screen file, always in the same place in memory, and always using the same layout. SAM users have four different screen modes, and can have numerous screens operating at once, so there can be several screen files in memory, and they need not have the same layout.

Fortunately, SAM mode 1 screens have the same layout as Spectrum screens. I want to go on giving you instructions and programs which are useful for both machines, so I am not going to deal with SAM screen modes 2, 3 and 4. There are articles in Format from time to time about the various modes, and the SAM technical manual describes all four modes in full. Mode 1 is by far the most complex of the four - the others will seem plain sailing once you have got to grips with mode 1. So I want SAM users to enter the BASIC command: MODE 1 in future, before attempting to use the routines.

SAM users will also have to have a couple of extra routines in most of the programs, for bringing the screen file into the memory area between 0 and 65535 - where the Z80 can get at it - and for paging it out again when we have finished with it, because SAM memory is organized in 'pages' of 16384 bytes. Four of these can be in use at once, and if we want to use higher pages, we must swap them with

the ones in use. This is called paging. The screens are all held well above 65535, each screen at the start of a page. When we work in BASIC, the ROM looks after the paging automatically, but in machine code we must take care of it. We will look at those routines next month.

I want Spectrum users always to look at the SAM operations too, because there will be techniques and commands involved in them that you will find useful, even though the complete routines themselves may be no use to you. So please don't skip bits because you think they are not for you.

The Spectrum screen file always starts at byte 16384, and is 6912 bytes long. If you enter:-

SAVE "picture" SCREEN$
or SAVE "picture" CODE 16384,6912

exactly the same bytes are saved. SCREEN$ is shorthand for CODE 16384,6912.

The current SAM screen file starting address is not fixed, because it depends on which screen is in use, and whether you have a 256K or 512K SAM, but fortunately there is an infallible way of discovering its address from BASIC:-

LET X=IN 252 BAND 31:LET Y=(A+1)*16384

will always leave you with Y holding the address of the current screen.

The BASIC command IN is just like the machine code instruction IN. It reads a byte from one of the ports of the data address bus, which is used in both machines to communicate with peripherals, including the keyboard, the sound chip and the monitor. SAM's port 252 is the VIDEO MEMORY PAGE

REGISTER, which governs which screen file, in which mode, is to be displayed on the monitor, and which is also used to communicate with the midi interface. LET X=IN 252 gives us the byte it holds in the X variable.

BAND is the SAM BASIC equivalent of the machine code instruction AND. You met AND last month. Look back to Fig 2 from last month to remind you of what it does. In BASIC and in machine code, we often use AND to isolate a number held in just a few of the bits of a byte, rather than in all of them. In the Coupé's VMPR byte, the screen page is in bits 0-4, so to isolate that number we need to AND with BIN 00011111, which will always reset bits 5-7 and leave bits 0-4 unchanged. BIN 00011111 is 31.

Fig 1 shows how, by using AND 31 (BAND 31 in SAM BASIC), the midi and screen mode bits are reset to 0, whatever they were to start off with, leaving us with just (screen page number minus 1) in the byte. By adding 1 to get the actual page number and multiplying by 16384, we have the starting address of the screen file.
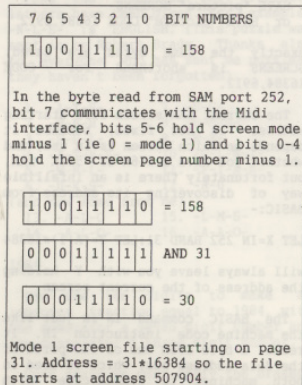
```
 7 6 5 4 3 2 1 0   BIT NUMBERS

 1 0 0 1 1 1 1 0  = 158
```

In the byte read from SAM port 252, bit 7 communicates with the Midi interface, bits 5-6 hold screen mode minus 1 (ie 0 = mode 1) and bits 0-4 hold the screen page number minus 1.

```
 1 0 0 1 1 1 1 0  = 158

 0 0 0 1 1 1 1 1   AND 31

 0 0 0 1 1 1 1 0  = 30
```

Mode 1 screen file starting on page 31. Address = 31*16384 so the file starts at address 507904.

SAM mode 1 screen files are longer than Spectrum screen files because they have extra bytes at the end holding the list of 16 colours being used. Spectrum users have no choice of colours. But if SAM users enter PALETTE as a direct command, the colours will be very like the Spectrum ones, and we can then ignore those extra bytes. The first 6912 bytes are exactly like those on the Spectrum.

So, how does the screen file work?. The first 6144 bytes hold the set/reset pattern of all the pixels in the screen, 1 bit per pixel. The other 768 bytes hold the attribute information for the 768 colour cells, 24 rows of 32 cells. In each cell, the reset pixels are in PAPER colour and the set pixels in INK/PEN colour. The FLASH and BRIGHT status affect the whole cell.

An attribute byte holds the information in the following way:-

Bit 7 Flash status   0=FLASH 0
                     1=FLASH 1
Bit 6 Bright status 0=BRIGHT 0
                     1=BRIGHT 1
Bits 3-5 PAPER colour
       (Can be colours 0-7 which only
        need up to 3 binary digits.
        7 is BIN 00000111)
Bits 0-2 INK/PEN colour (Also 0-7)

There are actually 16 colours, numbers 0-15, with 0 and 8 both holding black, but on the Spectrum colours 0-7 are called BRIGHT 0, and colours 8-15 are called colours 0-7, BRIGHT 1. The reason is that you need 4 bits to hold numbers between 8 and 15, 15 is BIN 00001111. If we used two lots of 4 bits for the PAPER/INK colours, we should have run out of bits in the byte and have nothing left for FLASH. So, to find the colour number, the BRIGHT bit is treated as bit 3 and the PAPER or INK bits as bits 0-2. If BRIGHT is reset, the resulting number is between 0 and 7, but if it is set, it adds 8 to the colour and the ones between 8 and 15, the bright colours, are used:
       BIN 0110 = 6 (Yellow)
       BIN 1110 = 14 (Bright Yellow)

Now, enter this BASIC command:-

(Spectrum users) CLS : POKE 16384,255
(SAM users) MODE 1: CLS : LET X=IN 252
     BAND 31,Y=(X+1)*16384: POKE Y,255

You will see a short line at the top left corner of the screen. Why? Because you have set the first 8 pixels of the screen file, and they have changed from PAPER to INK. (255=BIN 11111111)

You would expect that if you set all 6144 pixel status bytes in order, that the screen pixels would all be set in order from the top left to the bottom right. But, in the Spectrum and SAM mode 1 screens, things are not so simple.

Try the following and you will see for yourself the order in which the bytes are held in the screen file:-

(Spectrum users)
FOR X=0 TO 6143: POKE (16384+X),255: NEXT X

(SAM users)
MODE 1: LET X=IN 252 BAND 31,Y=(X+1)*16384: FOR X=0 TO 6143: POKE (Y+X),255: NEXT X

The screen is divided into three bands, each containing 8 rows of colour cells. The first 32 bytes set the top line of bytes. The next 32 set the top line of the second row of colour cells. When the top line of each row of cells in the top third of the screen have been set, the second lines of each row are set, then the third and so on until the whole of the top third is finished. Then the middle third is done, in the same order, and finally the bottom third.

Fig 2 shows this layout in more detail. Remember that the Spectrum screen always starts at 16384. The SAM screen file, wherever it resides in memory normally, has to be paged into the area between 0 and 65535 before we can manipulate it from machine code - when we poke from BASIC, the ROM pages the screen in and out without us being aware of it. It

would make life nice and easy if we could page the SAM screen to 16384 so that everybody could use the same addresses. But, sadly, that is not practical, so we shall usually be paging the screen to address 32768.

SPECTRUM & SAM (MODE 1) SCREEN FILE LAYOUT			
0000	LINE 1	0031	BYTES IN
0256	LINE 2	0287	THE TOP
0512	LINE 3	0543	ROW OF
0768	LINE 4	0799	CELLS
1024	LINE 5	1055	
1280	LINE 6	1311	
1536	LINE 7	1567	
1792	LINE 8	1823	
0032	LINE 1	0063	
1824	LINE 8	1855	
0064	LINE 1	0095	
1856	LINE 8	1887	
0096	LINE 1	0127	BYTES IN
1888	LINE 8	1919	THE OTHER
			ROWS IN
0128	LINE 1	0159	THE TOP
1920	LINE 8	1951	THIRD OF
			THE
0160	LINE 1	0191	SCREEN
1952	LINE 8	1983	
0192	LINE 1	0223	
1984	LINE 8	2015	
0224	LINE 1	0255	
2016	LINE 8	2047	
2048	ROW 9 LN 1	2079	BYTES IN
4064	ROW 16 LN 8	4095	MIDDLE THIRD
4096	ROW 17 LN 1	4127	BYTES IN
6112	ROW 24 LN 8	6143	LOWER THIRD
ATTRIBUTE BYTES			
6144	ROW 1	6175	BYTES IN
6368	ROW 8	6399	TOP THIRD
6400	ROW 9	6431	BYTES IN
6624	ROW 16	6655	MIDDLE THIRD
6656	ROW 17	6687	BYTES IN
6880	ROW 24	6911	LOWER THIRD

As you will see there are 32 bytes in a line and 8 lines in a row of colour cells, so a row consists of 32 8x8 pixel colour cells. There are 8 rows in a screen third and 24 rows in the full screen. In fig 2 the byte on the left is the first, and the one on the right the last, in a line. The byte numbers must be added to 16384 on the Spectrum, and to the start address of a SAM screen, to find the actual address in memory.

If you have a screen byte in HL, INC HL will move to the next one on the right provided you stay in one line of bytes. If you reach the end of a line, INC HL will move to the corresponding byte in the next cell, provided you stay in the same screen third.

This handy little subroutine will find the corresponding byte of the next cell wherever you are on screen, even if you cross from one third to the next.

```
NXCELL  RR  H
        RR  H
        RR  H
        INC HL
        RL  H
        RL  H
        RL  H
        RET
```

To find the corresponding byte in the next colour cell down, you add 32 to HL, provided you stay in the same third of the screen. Here is another useful subroutine, which we shall use a lot, it will find the next cell down wherever you are on screen, whether or not you cross a 'third' boundary.

```
NXDOWN  RR  H
        RR  H
        RR  H
        LD  BC,32
        ADD HL,BC
        RL  H
        RL  H
        RL  H
        RET
```

If you have a screen byte in HL, INC H, which adds 256 to HL, will find the next byte below, provided you stay in

the same colour cell.

The 768 attribute bytes run in one sequence from start to finish, without doing anything strange at 'third' boundaries.

This subroutine, entered with HL holding the address of a screen file byte, returns with HL holding the address of the attribute byte for the cell in which the screen byte lives.

```
FINDATTR LD A,H
         RRCA
         RRCA
         RRCA
         AND 3
         OR  88 ;Spectrum only. SAM
                 users paging screen
                 to 32768 use OR 152
         LD  H,A
         RET
```

You can see that this routine only alters the MSB of the address. This is because all the bytes in a colour cell have the same LSB. You can see that this must be true because INC H will find the next byte down - the byte in L remains the same. The LSB of the attribute byte for a cell is the same as the LSB for all the screen file bytes in the cell. So all we have to do to get the right attribute byte is find the right MSB.

You have met all the commands in these subroutines before except RRCA. This is a special form of RRC which is used only with the A register. It works in the same way as RRC A, but RRCA needs only one byte of object code, whereas RRC A needs two. It also works faster than RRC A. All the rotating instructions have these special versions for A - RLA RLCA and RRA also exist - all one-byte instructions and all faster working than the ones which treat A like all the other registers.

I am going to leave it there for for this month, because this has been the most complicated information I have had to give you so far, and you will probably need to read it several times to be sure you have grasped it all.

NEV'S
HELP PAGE

By:- Nev Young.

I think that I shall have to get John Wase's recipe for humble pie, as it's my turn to have some this month. I have found in the past that if I put anything that is wrong in the help page then I will soon be told by a torrent of letters. I suppose this is good as it does mean that I am not wasting my hard earned leisure hours writing this stuff and also that there are many other very knowledgeable people reading FORMAT. So what have I done wrong this time you may ask. Last month I said that the fonts in the FORMAT font library could not be used with the DTP package from PGC. I was wrong, you can.

The procedure is as follows. Create separate font files following the instructions that come with the font library. Then load Wordmaster. Next load the extension program CONVERT! and get it from the Wordmaster main menu. Give it the name of the font you want to use and it will be read into memory and converted into, I believe, a FNT-1 file. This can then be saved from Wordmaster as normal. To use this font you have to load the extension HEADLINER! and get it. You can then use the font within that program. The files that are produced can then be used within TYPELINER! If you do not have the DTP then you won't have a clue what all that is about.

Next a little problem that crops up from time to time. How do you get the RS232 printer port to work with a PLUS D. Normally this is no problem at all as when you get your PLUS D you will specify in the setup program that you do not want to use the PLUS D printer port. In this way the RS232 is left enabled. But what if you want to use both or if you have loaded a system file with the wrong settings.

Firstly I must emphasize that the

best solution is to have the correct setup defined in your system file (sys-3d on DISCiPLE. +sys-2a on PLUS D, or Uni-Dos). Which printer you use is defined by the POKE @11 command. On GDOS POKE @11,0 signals the use of the parallel printer port. POKE @11,1 signals use of the Spectrum's built in printer port. With Uni-Dos, just to be awkward, these are reversed. The way GDos and Uni-Dos work are slightly different but the basics are the same.

When you boot your system file, if the system file is set up not to use the parallel printer then nothing happens, as far as printers are concerned, and the RS232 is left functioning. If you now do POKE @11,0 (or 1 on Uni-Dos) the parallel printer is enabled and the channel information for the printer is overwritten with new values. The parallel printer will now be operating, assuming all the other printer variables are set up correctly. However if you now do a POKE @11,1 (or 0 on Uni-Dos) the RS232 printer is NOT re-enabled. This is because the channel area is not restored until you reset or NEW the machine. You could poke the RS232 values into the channel area your self if you wish and then it would be working, but in general you can not switch back without resetting. Of course if you only have one printer then just make sure that all the system files on all of your discs are setup to use that printer. Then when ever you boot your system it will be ready to print.

Also be aware of the fact that some programs do a POKE @11 when they are run and this could ruin your setup. These programs are normally easy to spot, the printer will work OK before they are run but stops once you try and run the program.

Something that I have been quite surprised about is the number of people who not only have a SAM or Spectrum but who also use an IBM PC. I do get quite a lot of letters asking if SAM can read and/or write PC discs. Well, as it comes, no. But that wonderful little company S.D.Software does have a program for SAM that enables it to both read and write IBM PC discs. Both the 3.5" 720K and the 5.25" 360K (if you have an external 40 track drive fitted). But remember SAM WILL NOT run an IBM PC program and the IBM PC WILL NOT run a SAM program. All that you can do is transfer data between the two systems.

Oh oh. You may remember the story I told about when is a PC not a PC a couple of months back. Well in much the same vein I have had a query regarding the software adverts in a computer magazine. I won't mention any names to save embarrassment. But ads like:-

SOFTWARE
   DOS 5 UPGRADE £49
   GRAPH-IN-A-BOX £49

HARDWARE
   EPSON EX 1000 COLOUR PRINTER £249
   5.25" FIXING KIT FOR 3.5" DRIVE £8

BOOKS
   DOS & BIOS FUNCTIONS £7.95
   MS-DOS TRICKS & TIPS £16.45

The question is what can he make use of. As he only has a Spectrum and a DISCiPLE and already has a printer then I would say none of it. Apart from the printer in the above list everything is for the IBM PC. To answer one specific question the fixing kit allows you to fix a 3.5" disc drive into a hole meant for a 5.25" disc drive. I think that it is fairly safe to say that if you've never seen it in FORMAT then it is not much use to a serious Spectrum or SAM user. (Yes I know there will be exceptions but I am trying to be general).

It is now March as I write this and I have this week received a post card from Luca Giuaudi in Italy with a post mark of 1991! Still what is on the card was worth waiting for. He gives a routine to let the OCP ART STUDIO program use the DISCiPLE/PLUS D printer port. I have not tried it but here it is.

Poke the following values in the main code block from 34660 upward:-

100,135,110,135,111,135,113,135,16,0, 201,175,201,207,57,201.

You must also edit the BASIC loader using the dreaded POKE @6,1 before the RANDOMIZE USR statement.

Mr Day of PORTCAWL reports that on his SAM, whenever he starts to use the printer, the first line is one position to the left with respect to all other lines. I don't know if I'm just lucky but it doesn't happen to me. Has any body out there seen this and if so is it a problem with the SAM, the interface, the printer or the software.

A tricky little problem with screen dumps now. R.Barton writes "how can I invert the screen dump to give black on white instead of white on black?" The first thing to realize is that snapshot option 3 will save the screen only to the disc. This can then later be reloaded and printed using SAVE SCREEN$ 1. However you can play about with it before you print it. To invert the screen colours all you need to to is swap the INK and PAPER colours for each character cell on the screen. The colour attributes for each cell are made up of 8 bits where the lowest three bits give the INK colour the next three bits give the PAPER colour and the last two are for BRIGHT and FLASH. So to invert the colours all you need to do is swap the lower 6 bits of each byte in the attributes area. The only problem is the Spectrum doesn't have bitwise operators. The following simple! program will load a screen file, print it, invert it, and print it again.

```
10 INPUT "FILE NAME "; F$
20 LOAD D1;F$ SCREEN$
```

```
30 SAVE SCREEN$ 1
40 FOR A=22528 TO 23296
50 LET ATT1=64*(INT((PEEK A)/64))
60 LET ATT2=((8*(INT((PEEK A)/8)))-A
   TT1)/8
70 LET ATT3=(8*((PEEK A/8)- INT (PEE
   K A/8)))*8
80 LET ATT4=ATT1+ATT2+ATT3
90 POKE A,ATT4
100 NEXT A
110 SAVE SCREEN$ 1
```

OK I think thats enough for this month so I will close with the usual comment that if you have any problems then send them to:- Mr N.Lamont, 11 Downing Street, London, W1.

Or, if you really must, to:-

          Nev Young,
          Format Help Page,
          70, Rainhall Road,
          Barnoldswick,
          Colne,
          Lancashire,
          BB8 6AB.
          England.

giving as much detail as you can.

* * * * * * *

Z80 SUBS - Continued from page 19.

As I've said before I don't have a SAM at the moment, although I will have soon, so I can't comment on the assemblers available for SAM users at the moment. However, having looked at a couple of manuals that Bob sent me I think my comment on choosing a Spectrum assembler will also apply to those for SAM just as well.

OK, that is it for this month, off to the post box with this disc, then down to work getting next month's column ready for your stimulation.

I look forward with anticipation to receiving many more contributions from readers - you will send them, wont you? Just address them to me at the FORMAT office and they will reach me.

B.W.M.N.M

34

PERSONAL BANKING SYSTEM

Reviewed By:- Peter Quanstrom.

Hilton Computer Services have produced the subject program for the SAM Coupé. I must confess that it gives me pleasure to use this quality Banking analysis menu driven system, first notified to Format readers in July 1991 under "News on 4". This program by means of multiple analysis can rearrange the banking entries into various information modules either for the screen or the printer.

The capabilities of the Personal Banking System (PBS) will hopefully be expounded as we progress but as you go deeper you find further facets to surprise you. For instance:-

1. Entries out of date order are automatically resorted.

2. Mistakes and Errors of omission can be corrected.

3. PBS can be tailored to your requirements, by loading the Initial Program Load utility (PBSIPL) after exiting from the PBS program.

4. You can open up more Bank Accounts, Credit Card accounts, Building Societies and Cash Accounts than you are ever able to need.

5. The original disc will contain demonstration entries, but these will be erased on your working copies when you "Initialise Directory".

6. The capability to look for things either by date range or expense category, when entering the enquiries mode.

The growing up stages proceed from the beta test version to a change from Samdos to the superior and faster combined MasterDOS and MasterBASIC,

not yet provided by Hilton, but coming I am sure.

The Personal Banking System - User Manual is completely comprehensive (User friendly or People compatible - whichever you prefer) and should explain all queries that arise, is easy to read, and contains diagrams of the on screen menus. The Reconciliation module included with PBS and the forthcoming General Ledger and Budget Modules are portrayed together with further products being released early 1992, these are the "Personal Filing system", and "Personal Portfolio System" (they will not be long in arriving).

The program is ideal for a club, small business or anybody wishing to maintain a control of their Bank/Cash and multi Bank transactions where transfers between accounts occur. Transfers between bank accounts are automatic, but you should beware if you have related Bank accounts of differing currencies where individual transactions are advised to be generated at their respective currency exchange rates.

Under the Bank Statement Menu it is possible to select a variety of 5 Formats in which to present your bank entries, these include 'Date, Reference, Details' followed by one of the following:-

Analysis, Debit, Credit, Balance

Analysis, Debit, Credit, VAT

Account, Debit, Credit, Balance

Gen/Ledger Code, Debit,Credit, Balance

Reconciliation, Debit, Credit, Balance

The Analysis of transactions are

35

particularly interesting where for example, motor expenses can be extracted from a multitude of entries and either listed on screen or dumped to printer.

Analysis appears under "System codes" together with Account, and GL codes, all of which are analysable. That is to say, under Account, for example, there appears any (all) Supplier/ Merchant(s) name(s) you wish, and can, under this program, list and total all purchases for the period/year, assisting you in your approach to that supplier for reciprocation of business!. PBS allows up to 64 codes and descriptions each for "Analysis", "Accounts", and "GL Codes". Should you require reminding at any time then a zoom facility will window and scroll your codes on screen.

Of particular concern is the second format above, where VAT can be extracted either by screen or for printout, but it depends on your own original VAT input (gross or net) how your end result materialises. On the other hand if you do not require VAT then don't use it.

Standing Orders/Direct Debits can also be entered on a weekly 28day, monthly, quarterly, six monthly or annual basis. Provided the date of inception of the order is subsequent to the commencement of regular entries, they will be automatically booked on due date.

The Reconciliation module enables you to check your PBS Bank transactions against your Bank statement, i.e. press Y for items that match, and R for those items not cleared by the Bank. When all items have been so marked, press L to list, enabling you to compare the book balance, adjusted by the uncleared (R) total, against the Bank balance entered.

There is also a password controlled utility called PBSIPL, able to be switched on or off. It is advisable to store this on a separate disc for increased security. You can select "change all passwords", in which case each file is raised in turn, and respective passwords changed.

Forward Projections is an option on the Main menu, that when selected, can forecast your (in mode) Bank Balance based upon Standing orders and Direct Debits to a date specified. This result can be displayed, scrolled or printed. Upon returning to the Main menu your normal bank account will resume.

Working copies are able to be produced as explained in the PBS Manual. Additionally you are also advised to back up your working copies on a "parent/child" progressive basis. The frequency depending on the volumes handled, to avoid losing too much information on the possibility of an unavoidable breakdown.

Hopefully this coverage of PBS will assist in understanding a very useful program for handling Bank, Credit and Cash transactions, and if you are interested in maintaining a control of your day to day finances, then you need not look any further, don't hesitate, this is for you, as you have value for money with PBS.

'Great – that's just the sort of aggression we're looking for! You'll make a perfect traffic warden'

'The way my dad plays, you shut your eyes just before you hit the ball, and the game is to find it in the bushes'

36

EXPANDING
SPECFILE+

By:- Ken Elston.

At the end of my recent review of SPECFILE+ from S.D.Software I promised you an article (or two) on adding new features to this excellent filing program.

SPECFILE+ allows you to write user extentions very easily. It reserves lines 9000 to 9999 for us to use and there is over 3k bytes still free. I know there are a lot of other lines free as well but some may be used for future expansion by SD themselves, whereas lines 9000 plus are ours for the use of.

The routine I am going to outline here adds a count feature to the program. SPECFILE+ users will already be familiar with the 'GROUP FUNCTIONS' which allow operations to be carried out automatically on all records that satisfy the match. My first venture into coding adds a new Group Function - \C - to count records. It is a very small routine, but if you look at how it changes SPECFILE+ it will help you to understand how to make your own enhancements.

As you will know from the SPECFILE+ manual the Group Function codes A-Z are accessed by a jump table starting at line 8901. This means the entry for \C is at 8903 so the first thing we need to do is to replace that line with the following.

```
8903 GOTO 9100: REM COUNT
```

Now add the following lines.

```
  25 LET COUNT=0
 310 PRINT AT 21,0;X$;: IF COUNT TH
     EN PRINT INVERSE 1;"COUNT"; IN
     VERSE 0;"=";COUNT
9100 LET COUNT=COUNT+1: LET Y$=J$:
     GOTO 280
```

The first line sets the count to zero, I placed this at line 25 because it was a convenient place - the count will always be zero before you start a search. The second line prints out a running count provided the count is not zero. This is done just after the matched record is displayed.

The last line, the one that gets executed each time a record is found, adds 1 to the count and loops round to continue the search. You could put any extra testing you want at this point.

You could add a call to the printing routine if you wanted to but I will leave you to expand things beyond this point.

Before I go I just want to cover one small facit of SPECFILE+ that is not covered very well in the manual. This is the \A Group Function. \A jumps to a routine at line 7000 with the selected record in E$() so you can Alter the record. What is not explained is that the record has already been deleted from the file D$(), so even if you don't want to update/alter a record you must still file it away again by doing the LET Y$=" STEP ": GOTO 2000 at the end.

Okay, that's it, I hope you like what I've done, if you do I might be back someday with another routine.

Since I got my first pre-production copy of SPECFILE+, from FORMAT to do the review, it has taken over. I now think in terms of SPECFILE+ whenever I have any data to store - my wife has even started using it to store her cooking recipies (good job I have a spare computer). If you don't already have a copy of SPECFILE+ then order one today - I find it the most usable filing program I have ever come across, and I wish I had shares in the company.

37