VOL.1
NO. 4

STATUS 1500

MAY
1983

* Europe: add 30% - Outside Europe: add 60% *
* 6 Issues £4.50 * 12 issues £8.50 * Single Copies £1.50 * Published Monthly *

* edited by RONALD COHEN, 62 BLENHEIM CRESCENT, LONDON W.11 *

SERVICE

Over Easter my printer packed up again - not SHARP's fault: I was involved in a slight accident - and this is why the magazine is late. I sent it off for repair on the Tuesday after Easter, and got it back 4 days later. The results of my enquiries about service seem to indicate that SHARP are genuinely anxious to help customers, and do their best: but when they are under pressure, or it is difficult to help in a particular case, they are not always able to implement their good intentions. It would seem that the PC 1500 very seldom goes wrong, unless misused: so perhaps their unexpanded repair facilities are a tribute to the sturdiness and precision of the machine.

TANDY are more impersonal. My cassette-recorder was perhaps damaged at the same time, and has gone to them for repair. I took it in to their Centrepoint Computer Centre last Wednesday, and confidently expect it back a week later. Apparently all repairs are collected from the branch each Thursday, and are normally returned the following Thursday. I am keeping my fingers crossed!

* * * * * * * *

It is sad to have to relate that SHARP have shown themselves extremely allergic to criticism of them that has appeared in this magazine. They do not appear to appreciate the very great support this magazine gives them. Let me take this opportunity to make it clear both to SHARP and to subscribers that no considerations either personal or commercial will deter STATUS 1500 from printing what it believes to be the truth: we shall continue to criticise where any criticism is due: and to praise where praise is due. We do not intend to print in every issue what an excellent machine the PC 1500 is. We all know this. It is because we like the machine that we are sensitive to any imperfections in back-up.

* * * *CONTENTS* * * *

MAILING LIST

This is the program I use for printing out the labels when sending out the newsletter, and controlling the subscription list. It has advantages and disadvantages compared with a conventional database. All data must be programmed in; it is designed for my own use, would have to be adapted for any other use. However keeping all data in program form saves much time - just a CLOAD, and its ready to run: no need to spend time with INPUT # . Being only about ½K long leaves plenty of room for data. Any editing is done by altering the program: any 'search' application must be added as a subroutine, and deleted after.

Lines 1 - 5 DIMension the variables, and run through the names on the screen to check on correct formatting of data: on error it will halt on or before the wrongly formatted entry. (Must be RUN before the rest of the program can be used).

Lines 6 - 120 print out the labels. If you want a single label you can specify it by number: BREAK to stop further labels being printed. "repeat" refers to the number of times it overwrites the same label: useful with pens getting dry : "repeat 2 " gives a good impression.

Lines 1010 (which is entry no.1) onwards hold data. Note that the subscription information and name are always on a line ending in 0, thus allowing lines 10000 to 10002 tc LLIST subscription information and names only, without addresses. The second number (single digit) gives the number of string entries including the name. This number must be correct.

Line 13000 primes the Reserve keys for the symbols used most in programming in new names and addresses.

Execute by RUN, DEF X , DEF L , DEF V

```
 1:RESTORE :INPUT
   "SPEED";S:WAIT
   S
 2:DIM A$(9)*24
 3:ON ERROR GOTO
   5
 4:READ A:READ B:
   FOR J=1TO B:
   READ A$(J):
   NEXT J:PRINT A
   ;" ";A$(1):
   GOTO 4
 5:WAIT 333:PRINT
   A;" ";A$(1):
   END
 6:"X"R1=2:ON
   ERROR GOTO 7
 7:INPUT "start a
   t ";X:RESTORE
   (X*10+1000):
   INPUT "repeat
   ";R1
10:GRAPH :ROTATE
   1:RESTORE X*10
   +1000
15:X=180:Y=0
20:READ S:READ L
30:GLCURSOR (0,0)
   :CSIZE 2:
   LPRINT S
40:U=(L<7):K=20+1
   0*U
60:CSIZE 2+U
70:FOR F=1TO L
80:READ A$(F)
89:FOR G=1TO R1
90:GLCURSOR (X,Y)
91:LPRINT A$(F)
92:NEXT G
100:X=X-K:Y=Y-K
110:NEXT F
120:TEXT :CSIZE 2:
   LF 13:GRAPH :
   ROTATE 1:GOTO
   15
1010:DATA 111001,
     5,"M.THATCHE
     R"
1011:DATA "10 DOW
     NING STREET"
     ,"WHITEHALL"
     ,"LONDON","S
     W1"
1020:DATA 222002,
     5,"PRESIDENT
      REAGAN"
1021:DATA "THE WH
     ITE HOUSE","
     PENNSYLVANIA
      AVENUE"
1022:DATA "WASHIN
     GTON D.C","U
     .S.A"
1030:DATA 333003,
     4,"Y.ANDROPO
     V"
1031:DATA "THE KR
     EMLIN","MOSC
     OW","USSR"
9998:DATA 0,1,"EN
     D"
9999:STOP
10000:"L"FOR F=101
     0TO 1030STEP
     10:CSIZE 1
10001:LLIST F:LF -
     4
10002:NEXT F
13000:"V"POKE 1442
     2,1,58,3,34,
     44,34,4,241,
     141,5,59,6,4
     4,2,34,0
13001:BEEP 1
```

DISTRESS SIGNALS

JAMES LOTHIAN asks "What is the purpose of the commands PEEK#, POKE#, and OPN ?"

OPN is used in connection with the RS232 interface. PEEK# and POKE# access the "ALTERNATIVE MEMORY BUFFER", which is used mainly by the CPU . The series "PEEK,POKE & MEMORY" will eventually deal with these in greater detail.

CHRISTOPHER LEDSAM asks if I can recommend a good book on programming in machine code in general.

I do not know of one: can any reader help? The difficulty is that most information published is with reference to a particular chip, such as the 6502. Obviously one must use whatever code the chip understands. The SHARP chip in the PC 1500 is custom-made for them, and therefore I cannot see information , except from Sharp, being of much use.

ALLAN THOMAS writes from New Zealand that he tried my 'RENUMBER' program, and it renumbered every line as 10. Do I know what went wrong?

No, I do NOT know what went wrong, since you did not bother to enclose a listing of what you were trying to run. Your confidence in my ability to debug your program without even seeing it is excessively flattering. Very probably you wrote Sl instead of SI in line 60000 or line 60130. The following are very easily confused, particularly in small sizes:

letter I and number 1
letter U and letter V
number 0 and letter O and symbol @

If this advice does not solve your problem, please write again, and ENCLOSE A LISTING.

FRANCIS AKINWUNMI is trying to write a subroutine which will make a word (A$) disappear off the left-hand side of the screen, as it can do on the right-hand side.

Imagine that you are standing at a terminus: a train disappears into a tunnel in the distance. But it cannot disappear from view into the buffers where you are standing! You must create the illusion that it does so, by successively chopping bits off the front. Try something like:

```
101: N = LEN A$: WAIT 0
102: CURSOR 0: PRINT   RIGHT$ (A$,N):CLS
103: N=N-1: GOTO 102
```

In general, many problems seem insoluble because they are formulated too narrowly. Look through the Instruction Manual, and back numbers of this newsletter, and see if any of the techniques could be relevant to your problem: then reformulate the problem in terms of that technique. It very often works.

TIM LANDON writes "Here is a better program than yours for Demodulating Infuriants. Please publish it quick before someone else markets it." Unfortunately I have not been able to make it work in the way intended, and since his letter is dated "*April Fools Day, in MidChannel*" I question whether I am meant to . More news - perhaps - next month.

The USING instruction is cumbersome to use. It must be written out in full every time; if you wish to make use of the "&&&.." facility for formatting words, it automatically cancels the USING instruction for numbers. However by the use of PEEK & POKE this facility can be addressed indirectly, and words and numbers can both be formatted simultaneously.

The USING counters are from 30869 to 30872.

30869 switches the USING format on.(1 for On, or 0 for Off).
30870 holds the format for numbers. (add 1 for the sign,visible or not)
30871 formats strings, as by "&&&&&..."
30872 is for numbers after the decimal point(add 1 for the point)

Try this:
```
99: INPUT A$: POKE 30869,1,LEN A$-1 : LPRINT A$: LPRINT.
    " "; LEN A$
```
and you will find that the length of the string is printed immediately under the last letter of the word. With a single digit this could be done by an LCURSOR instruction: but if the number is 2 digits, LCURSOR would print the first digit under the last letter. Try making A$ a 10 or 11 letter word, and you will see the difference.

If into 30869 you POKE 65 instead of 1, you will get the 'asterisk fill'. If you POKE 30869, 33, you will get the ± sign where appropriate. POKE 30869, 97 will combine both formats. (97 = 64 + 32 + 1)

Try this:
```
11: J$="HEAVEN": J= 234.567: POKE 30869,65,6,5,2
12: LPRINT J$; J
```

Of greater importance are the Main System Pointers, which run from 30821 to 30826. Readers who have tried any of the 'utility' programs in this newsletter may recognise these numbers. They are in pairs, working in '256ary'

30821 and 30822 indicate the start of program space. 256 *(PEEK 30821) + PEEK 30822 will give the same as STATUS 2 - STATUS 1

30823 and 30824 give the end of program: and correspond to STATUS 2.
30825 and 30826 give the start of the SEARCH/EDIT area. You may remember they were used in the MERGE process in last month's DISTRESS SIGNALS, and in SUPERMERGE.

All these pointers can be manipulated by POKEing different values into them, but generally only in co-ordination with other code. For instance, you cannot merely delete the end of a program by altering the values in 30825 and 30826: they must coincide with the End-of-Program marker, which is always 255. Altering the 30821 and 30822 values may delete the beginning of your program, if you find the right address for your new start.But experiment with care: and PEEK the values you are altering, so that that you can POKE them back if things go wrong. Study the utility programs, and see how they are used. There are no short cuts.

You will note how 30821 and 30822 normally hold 56(or 64, or 32) and 197 The first figure corresponds to a number of K (256 is a quarter K) and 197 represents the RESERVE area - 9 bytes for pointers, 78 for templates, and 110 for RESERVE programs.

[To be continued]

AIRCRAFT CARRIER

A miniature 'landing' program in 581 bytes. Execute by Cursor movements. Crashes if you go too high or too low. Two options: 'Easy' is hard, 'Hard' is horrible.

```
  1:E=0:H=1:INPUT
    "EASY(E) OR HA
    RD(H)? ";U:CLS
  5:BEEP 1:BEEP 30
    ,0,0:BEEP 1
  9:X=1+RND 5:Y=9+
    RND 110:GOSUB
    555:P=12+RND 1
    00
 13:ON ERROR GOTO
    700
 99:A$=INKEY$ :
    GOTO 100+ASC A
    $
100:X=X+2-RND 3:Y=
    Y+3-RND 5:GOTO
    500
108:Y=Y-1-U*(3-RND
    3):GOTO 500
110:X=X+1+U*(2-RND
    2):GOTO 500
111:X=X-1-U*(2-RND
    2):GOTO 500
112:Y=Y+1+U*(3-RND
    3):GOTO 500
500:Z=2^X:ZZ=2*Z:
    IF Z>63LET ZZ=
    Z/2
501:GCURSOR P-1:
    GPRINT 0;96;64
    ;64;64;0;64;64
    ;64;96;0;
502:GCURSOR Y-1:
    GPRINT 0;0;Z;Z
    ;Z;ZZ;Z;Z;Z;0;
    0;
503:IF Z>63AND Y=P
    GOTO 900
504:IF (Z>63OR Z<1
    )AND Y<>PGOTO
    700
507:R=RND 5:IF R=3
    *ULET P=P+2-
    RND 3
508:GOSUB 555:GOTO
    99
555:WAIT (RND 100/
    RND 40)*U
556:RETURN
700:WAIT 60:CLS
701:GCURSOR Y:
    GPRINT 2;4;8;2
    1;8;4;2:CURSOR
    0:PRINT "CRASH
    ED!)";Y-P:GOTO
    501
900:IF Z=64AND Y=P
    CLS :WAIT :
    PRINT "LANDED!
    "
```

"SHOPPING LIST" by *DAVE JACKSON*

The main interest in this miniature is the use of PAUSE L$; in line 15 - PAUSE followed by a semi-colon is a technique I have not met before.

```
  5:"SHOP":CSIZE 2: WAIT
 10:CLS:READ L$: Q$=""
 15:PAUSE L$;
 20:INPUT Q$
 25:IF Q$ = "" THEN 20
 30:IF ASC Q$ = 48 THEN 10
 40:LPRINT Q$+" "+L$
 50:GOTO 10
900:DATA "xxxxx","yyyyy" etc.etc.
```

PROGRAM EXCHANGE

ANGUS CRAWFORD of *ORION AIRWAYS* has an AIR NAVIGATION program which loads the Latitude, Longtitude & Magnetic Variation of Fixed Navigation Points and reads out bearings. This is a practical program which he uses constantly while flying.

J.K.GAUTON has composed a STAR FINDER program which will give the position in the sky of the major stars at any time or place.

* EASIFILE * DBASE * EXECUTIVE *

I am not fond of general database systems; I would prefer to sit up half the night writing a specialised program for a particular application - and perhaps the next five nights debugging it! But the average business user has neither the time nor the energy to do this. He wants something that works first time. He is prepared to spare a limited amount of his time learning the use of the database: after which he wishes to forget about the mechanics of it, and just use it. It is in this light we must consider the three database systems named above. EASIFILE and DBASE are conventional databases, whereas EXECUTIVE is more of a document-handling system, with some number-crunching facility, so it is not entirely fair to compare it directly with the others. But then I have no intention of being fair: the average business user is not particularly tolerant either.

EASIFILE is the most elaborate of the three,and has the most facilities. It comes with an excellent manual, written in 'tutorial' form. Nevertheless being impatient I got into a muddle with it, not understanding initially the need to enter at least one "Report Format" before anything could print out. The program is largely menu-driven, and I did not like the flow from menu to menu. The main faults are faults of the PC 1500, rather than of the program; I feel the large number of facilities offered are top-heavy for maximum 10K, of which over 4K is occupied by the program itself. Also, the slowness of CLOADing and INPUT# data must limit its use.(Same for all). The author very fairly points out some limitations , and gives a most useful explanation of what a Database is.

DBASE is the one I found simplest. It has less facilities than EASIFILE, - it only searches on one level, instead of three - but I think this is adequate considering the limited number of records that the program will hold. It is accompanied by a listing. The manual is less elegant, being in broadsheet form. Perversely, I found this easier to work with. The program is only faintly marred by instructions on how to prime a function key in Reserve mode, to give GOTO 1 @ . Quite unnecessary : the following simple line of program could have taken care of this , without taking the user out of RUN mode.

```
9999: "S" POKE STATUS 2 - STATUS 1 -111, 1, 241, 146, 49, 64, 0
```

and then DEF S would have done the priming.

EXECUTIVE is somewhat different. It is designed for document handling. All entries occupy exactly 26 bytes, neither more nor less. Records are not subdivided into fields: indeed the divisions between records disappear whenever you examine a different record, and must be re-erected, which I found irritating and confusing. The number-crunching facilities could be useful. I also found it difficult to clear data.except item by item. It has been pointed out that adding a subroutine would do this: but the coding is extremely sophisticated: and I would not find this easy to do. However the purpose of this program, as I have mentioned, is not exactly that of a conventional database, and no doubt there are uses for which it has greater advantages than indicated in this review.

Overleaf various features of the three are compared in tabular form.

EASIFILE by Ian Traynor is distributed by ELKAN ELECTRONICS
DBASE by Wadsworth is distributed by MICROS FOR MANAGERS
EXECUTIVE by MiCROL distributed by MICROMAIL, 38 Burleigh ST.Cambridge.

DATABASES COMPARED

	Easifile	Dbase	Executive
SORT	YES	YES	NO
by specified field	YES	YES	NO
specified range of records	NO	YES	NO
SORT NUMBERS	YES	must be equal length	NO
RECORDS FORMATTED	YES	YES	NO
FLEXIBLE LENGTH FIELDS	YES	YES	must be 26 chrs.
PRINT SIZE OPTIONS	YES	NO	one on each side
FLEXIBLE SIZE RECORDS	NO	NO	YES
FIELDS NAMEABLE	YES	YES	NO
RECORDS NAMEABLE	NO	NO	YES
SIMPLICITY OF MENU	NO	YES	YES
MENU/COMMAND DRIVEN	MENU	MENU	COMMAND
SIMPLE TO CLOAD	YES	YES	NO
SEARCH LEVELS	3 (including comparisons)	1	start of lines only
NEED TO UNDERSTAND PROGRAMMING	NO	V.SLIGHT	YES
EDITING BY.....	FIELD	RECORD	LINE
Manipulate NUMBERS SEPARATELY	NO	NO	YES
LISTING PROVIDED	NO	YES	NO
CLEAR ALL DATA	simple	simple	tedious, unless subroutine added
DELETE DATA + SAVE FORMAT	NO	YES	no formats
WORKS ON 4K	limited	NO	NO
----------	---------- with 8K :	----------	----------
MAX FIELDS PER RECORD	10	20	255
MAX RECORDS	256	255	255
MAX CHRS. PER FIELD	16	16	26
MAX CHRS. PER RECORD	80	80	6630
MAX RECORDS, EACH 80 CHRS.	61	63	85
MAX RECORDS, EACH 10 FIELDS OF 8 CHRS	61	63	25
DISABLES 'BREAK' KEY	NO	NO	YES
TOTAL COMMANDS/MENU CHOICES	26	23	30
GETTING STARTED: ASPIRINS NEEDED	2	0	4
LENGTH	4½K	4K	2½K

SAFECRACKER

In last month's PEEK & POKE I described how programs could be given a measure of protection, by concealing lines from LLISTING. Here is a brief program for stripping that protection. MERGE it, or add it by hand to the program to be stripped. It is so short that it can be deleted manually after use.

```
10:"="P1=STATUS 2
   -STATUS 1-1:
   WAIT 0
20:P2=PEEK P1
30:POKE P1,13
40:E1=STATUS 2-239

50:K=1:E1=E1-1:IF
   E1<P1POKE P1,P
   2:STOP :LLIST:END
60:E2=PEEK E1:
   PRINT E1
70:IF E2<>13LET K
   =K+1:E1=E1-1:
   GOTO 60
80:IF PEEK (E1-3)
   =13BEEP 3:E1=E
   1-3:K=K+3
81:IF PEEK (E1-2)
   =13BEEP 2:E1=E1
   -2:K=K+2
90:POKE E1+3,K-3
100:GOTO 50
```

FROM THE KEYBOARD

When changing pens the colour you want to change is always the last one to turn up. In fact if you call the previous colour before you key for changing , the one you want will appear for disposal. For instance, if you wish to change pen 3, call COLOR 2 before going into 'change' position. If you want pen 1, key COLOR 3 first.

ROUNDING OFF negative numbers sometimes requires thought. ABS INT (-3.4) and INT ABS (-3.4) do not give the same result. In the first case the INT is done first. INT 3.4 gives the next lower whole number, which is 3, but the next whole number *lower* than -3.4 would be 4. If you wanted the result 3 you would need to key

```
X= -3.4
X= (INT ABS X)* SGN X
```

To round off to the *nearest* whole number just add 0.5. Thus INT (3.3 +0.5) gives 3 , whereas INT (3.8 + 0.5) gives 4 . For negative numbers, work it out for yourself.

SPACE-SAVING. If you wished to DIMension 250 numbers , by DIM A(250), at 8 bytes per number, this would occupy 2000 bytes. But if they were all not more than 4-figure numbers , you could DIM A$(250)*4, and operate on A= VAL A$(n) whenever you wished to use them. To put a number back into variable A$(n), merely : A$(n)= STR$ A . You would then have reserved only 1000 bytes instead of 2000.

SOFTWARE RECEIVED

EASITREND *a statistical forecasting program by IAN TRAYNOR*

PROCOS *a "number-crunching" program from MiCROL*

Both these very useful programs will be reviewed next month , or soon after.

MINDBOGGLE CORNER

First, a quickie. No prizes. On the back of your PC 1500 it is described by two of the following words:
POCKET/ELECTRIC/ELECTRONIC/MICROCOMPUTER/COMPUTER/CALCULATOR
DON'T LOOK YET! Which two words? Write down your answer before you look.

The main competition this month is to COMPOSE A MINDBOGGLER FOR THIS CORNER. Closing date: August 1st. The entries I find suitable - quite arbitrarily - will be printed on this page, and the problem that attracts the most entries will be the winner. *Prize:* £10 book-token.

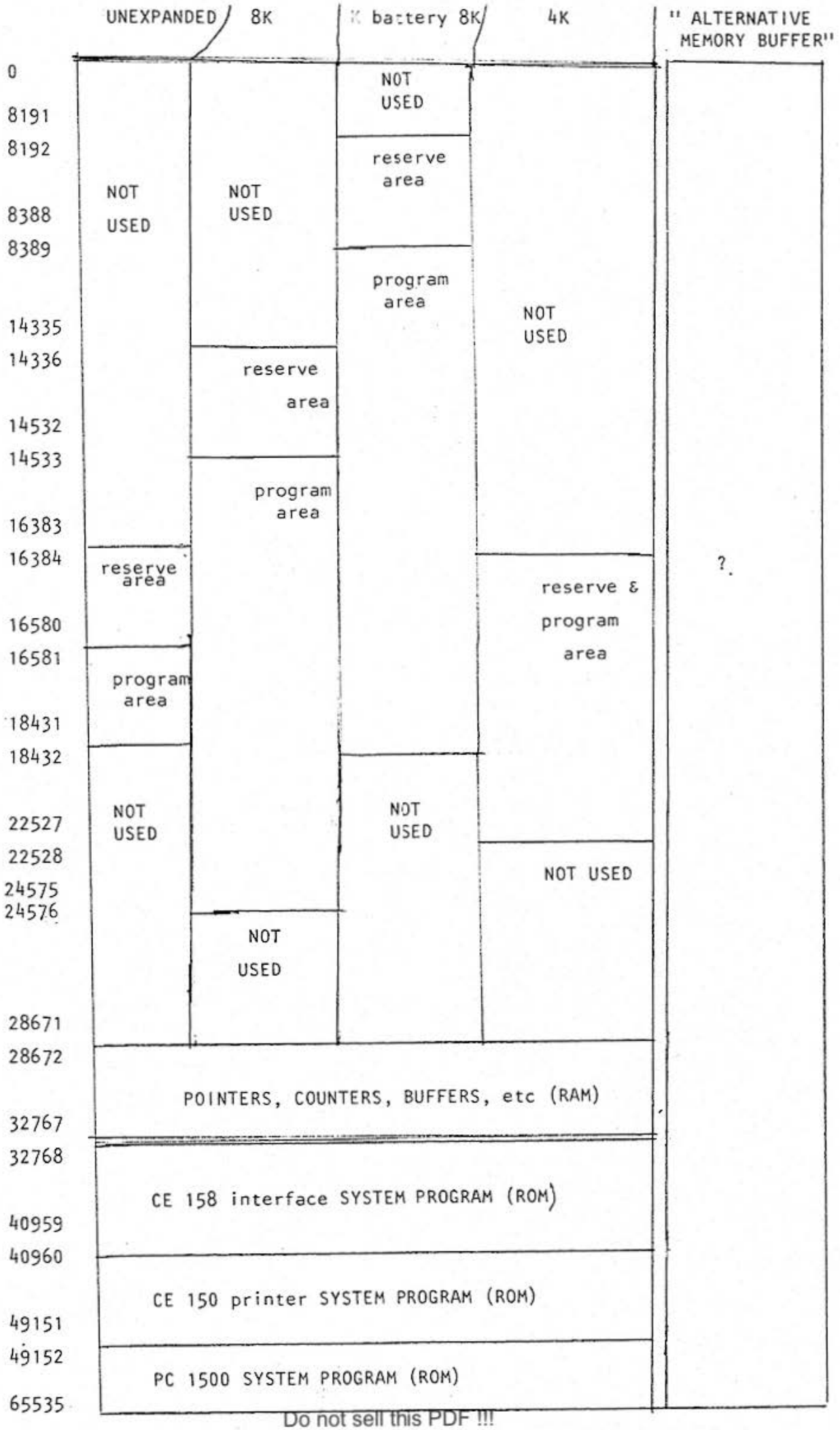
APRIL MINDBOGGLER received disappointingly few entries. Whether subscribers cannot write a program to design a multicolored elliptical spiral, or merely cannot be bothered to, I shall never know.

The winner was C.P.UNDERWOOD , *home in 128 bytes, closely followed by JOHN MACK with 135 bytes. A brilliant entry which took only one line of 80 bytes was unfortunately disqualified.*

```
10:ARUN :GRAPH :
   INPUT "r1";A,"
   r2";B,"d";D:
   GLCURSOR (A,B)
   :SORGN :
   GLCURSOR (0,B)
```

```
20:FOR C=0TO 360:
   X=A*SIN C:Y=B*
   COS C-D/360*C:
   LINE -(X,Y),,C
   /91:NEXT C
30:A=A-D:B=B-D:
   GOTO 20
```

OUTLINE MEMORY MAP



11 Bury New Road, Prestwich, Manchester M25 8JZ Telephone 061-798 7613 (24-hour service)

ELKAN ELECTRONICS

ef EASI-FILE 1500

Data base management program that acts as a "card-index system in your pocket!" Program features rapid sorting techniques searching on 3 levels or on key words. Max. no. of records per tape 256. Ideal for mailing lists, small stock control systems, etc.

ELKAN PRICE £19.95

ec EASI-CALC 1500

"Our most popular program" A "spreadsheet" program. Ideal for problem solving and "What if...?" types of questions. Up to 26cols. or 99 rows. Needs 8K RAM, CE150 and tape recorder. Data quality tape and 30page manual.

ELKAN PRICE £19.95

Coming soon!!

EASI-TREND 1500

A statistical forecasting program

Send cash, cheque, Visa or Access card number for quick delivery. All ELKAN PRICES include V.A.T. (where applicable) and U.K. post and packing charges

Just published!!

Write or phone for our very latest FREE POCKET ELKAN FILE with full details of all our products and services for pocket computer users!!

MM

Your partner in computing

Micros for Managers

PC1500 with FREE DBASE data base management software worth £24.95
£169.95 each Call for quantity discounts.

SOFTWARE for PC1500 (also available for PC1211/1251)

3-D VIEW Design package for architects and kitchen planners. £24.95

PERT Complete set of programs identify critical path, allocate staff £24.95

APD AIR PIPE DUCT All IHVE Formulae 18 design programs for air, water flow in rectangular circular pipes, ducts. £24.95

BOOKS

"ALL ABOUT INTEREST" McElwain
Home Finance, Consumer Loan costs, Disc. Cash Flow 186 pages, Flexbound
Publ. at £9.95, Now only £4.95 +50p P&P

To order, enclose your remittance and post to Micros for Managers, 149 Gloucester Road, London, SW7 4TH.